Mentor®
A Siemens Business

EZwave™ User's and Reference Manual

Release 2020.4

Document Revision 14

Support Center: support.sw.siemens.com
Send Feedback on Documentation: support.sw.siemens.com/doc_feedback_form

Revision History ISO-26262

Revision	Changes	Status/ Date
14	Modifications to title page to reflect the latest product version supported. Approved by Jonathan Kirkham. No technical enhancements, changes, and fixes listed in the AMS Release Notes for this product are reflected in this document. Approved by Franck Durand.	Released October 2020
13	Modifications to title page to reflect the latest product version supported. Approved by Jonathan Kirkham. No technical enhancements, changes, and fixes listed in the AMS Release Notes for this product are reflected in this document. Approved by Franck Durand.	Released July 2020
12	Modifications to title page to reflect the latest product version supported. Approved by Jonathan Kirkham. No technical enhancements, changes, and fixes listed in the AMS Release Notes for this product are reflected in this document. Approved by Franck Durand.	Released April 2020
11	Modifications to title page to reflect the latest product version supported. Approved by Jonathan Kirkham. No technical enhancements, changes, and fixes listed in the AMS Release Notes for this product are reflected in this document. Approved by Franck Durand.	Released January 2020

Author: In-house procedures and working practices require multiple authors for documents. All associated authors for each topic within this document are tracked within the Mentor Graphics Technical Publication’s source. For specific topic authors, contact Mentor Graphics Technical Publication department.

Revision History: Released documents maintain a revision history of up to four revisions. For earlier revision history, refer to earlier releases of documentation which are available on *http://support.mentor.com*.

Table of Contents

Revision History ISO-26262

Chapter 1
Introduction. 37

EZwave Overview 38
EZwave Process Flow. 38
Joint Waveform Database (JWDB) 39
Invoking EZwave. 41
Invoking EZwave in Batch Mode 45
Supported File Types 45
Visual Tour of EZwave 48

Chapter 2
Setting Up EZwave 53

EZwave Environment Variables 54
Environment Variables Set at Installation 54
Environment Variables Set at Run Time 54
Changing Default Environment Variables 56
EZwave Format Preferences 58
Configuring Waveform Names. 58
Configuring the Color Scheme 59
EZwave Display Preferences 61
Configuring Display Preferences 61
Configuring Colors, Fonts, and Styles 62
Changing Waveform Colors 63
Configuring Keyboard Shortcuts 64

Chapter 3
EZwave GUI Overview 67

EZwave Viewer Interface 68
Menu Bar 70
File Menu 70
Edit Menu 71
View Menu 74
Format Menu 75
Tools Menu. 76
Cursor Menu. 78
Window Menu 79
Help Menu 80
Toolbar 81
Waveform List Panel 83
Graph Window 90

Creating a New Graph Window 90
Graph Window Title Bar 91
Rows 91
Waveform Names Display 91
Graph Window Status Bar 92
Workspace 92
Workspace Taskbar. 93
Application Window Status Bar 93
Graph Window Popup Menus 95
Graph Window Popup Menu 96
Row Popup Menu 96
Waveform Popup Menu 98
Axis Popup Menu 99
Cursor Popup Menus 102
Vertical Cursors 102
Horizontal Cursor 104
Cursor Value Popup Menu 104
Waveform List Panel Popup Menus 106
Waveform List Popup Menu. 106
Database Popup Menu 107
Folder Popup Menu 109
Hierarchy Popup Menu. 110
Waveform Name Popup Menu 111
Selected Waveforms Popup Menu 112
Workspace Popup Menu 113
Keyboard and Mouse 115
Keyboard Commands 115
Mouse Strokes. 115
Waveform Calculator GUI. 116

Chapter 4
Add Waveforms 121

Waveform Basics 122
Starting EZwave and Loading a Database 122
Single Waveform Plots 123
Plotting a Single Waveform 123
Moving a Plotted Waveform 124
Copying a Plotted Waveform 124
Multiple Waveform Plots 125
Plotting Multiple Waveforms 125
Plotting Waveforms With the Same Name Using Tandem Mode 126
Managing Database Groups for Tandem Mode 128
Plotting the Difference Between Two Waveforms 130
Creating an XY Plot 130
Managing Waveform Groups 131
Grouping Waveforms 131
Adding a Waveform to a Group 132
Removing a Waveform From a Group 133

Creating an Empty Waveform Group. 133
Ungrouping Waveforms 134
Rules When Plotting Analog and Digital Waveforms 134
Modifying Axis Settings 136
Aligning Y Axes With Different Scales 136
Creating Multiple Y Axes. 137
Changing Y-Axis Settings 138
Modifying Waveforms. 139
Hiding Waveforms 139
Unhiding Hidden Waveforms. 139
Modifying Waveform Properties 140
Modifying the Appearance of Row Titles 141
Specialized Plotting 142
Plotting Complex-Valued Waveforms 142
Plotting wreal Waveforms 143
Plotting Multiple Bit Waveforms as a Bus 143
Displaying Bus Values as a String 144
Plotting Assertions 145
Plotting Histograms 147
Plotting CDFs 148
Compound Waveforms 151
Displaying Compound Waveforms 151
Displaying Compound Waveforms as Single Elements 152
Merging Waveforms Into a Compound Waveform Using an Index File 154
Digital Compound Waveforms. 155
Index File Format 156
Updating Waveform Data 158
Configuring Automatic Reloading of Waveform Data. 158
Manually Updating Waveforms 160
Waveform Update Shortcuts 160
Updating Waveforms With Options. 160

Chapter 5
Analysis 163

Working with Cursors 165
Adding the Base Cursor 165
Changing the Base Cursor 166
Adding Reference Cursors 167
Adding Relative Reference Cursors 168
Adding a Horizontal Cursor 169
Copying Horizontal Cursors 171
Moving Cursors 172
Setting the Visibility of Cursor Values. 174
Moving Cursor Value Flags 174
Hiding a Cursor Value for a Single Waveform 174
Hiding a Cursor Value for All Waveforms 175
Showing Cursor Values for Selected Waveforms 175
Setting Default Cursor Value Visibility When Selecting Waveforms 175

Sharing Cursor Value Visibility Settings . . . 177
Setting Cursor Value Visibility in the Cursor Values Table . . . 178
Displaying Cursor Values in the Reserved Area . . . 178
Working with Y-Level Lines . . . 180
Adding Y-Level Lines . . . 180
Choosing How Delta-Y is Calculated . . . 181
Using the Cursor Value Table With Compound Waveforms . . . 182
Changing the Visibility of Graph Window Elements . . . 183
Using Pick Points to Take Measurements . . . 184
Taking Measurements With Pick Points . . . 184
Adding Delta Markers With Pick Points . . . 187
Using Text Annotations . . . 189
Adding Text Annotations . . . 189
Global Display Options for Text Annotations . . . 192
Using the Event Search Tool . . . 194
Performing an Event Search . . . 194
Performing an Expression Event Search . . . 195
Working with Eye Diagrams . . . 197
Creating an Eye Diagram . . . 197
Adding Additional Waveforms to Eye Diagrams . . . 203
Editing an Eye Mask . . . 203
Eye Diagram Measurement Calculations . . . 206
NRZ (PAM2) Cross Eye Calculation . . . 206
PAM3 and PAM4 Eye Calculations . . . 208
C-Phy Eye Calculation . . . 213
Working With Smith Charts . . . 219
Scattering Parameters (S-Parameters) . . . 219
Creating a Smith Chart . . . 220
Impedance and Admittance Displays . . . 220
Smith Chart and Polar Displays . . . 223
Cursors in the Smith Chart . . . 224
Circles in the Smith Chart . . . 226
Changing the Visibility of Elements . . . 228
Comparing Waveforms . . . 230
Support for Different Types of Waveform . . . 231
Using the Waveform Compare Wizard . . . 231
Manually Comparing Waveforms . . . 236
Starting a Comparison . . . 236
Selecting Waveforms for Comparison . . . 238
Running a Comparison . . . 239
Viewing Waveform Comparison Results . . . 240
Stepping Through Differences With a Cursor . . . 240
Displaying the Tolerance Tube for Analog Comparisons . . . 241
Waveform Comparison Reports . . . 244
Generating a Waveform Comparison Report . . . 244
Digital to Digital Comparison Reports . . . 244
Analog to Analog Comparison Reports . . . 245
Spectral Comparison Report . . . 245
Viewing and Saving Comparison Rules . . . 246

Saving a Comparison Session 246
Setting Comparison Options 248
Setting General Comparison Options 248
Setting Comparison Method Options 250
Setting Tolerance Options 253
Setting Conversion Options 257
Analog Waveform Comparison Algorithm 258
wreal Waveform Comparison 259
Analyzing Waveform Parameters to Generate Pivot Waveforms 261
Analyzing Current Consumption 262
Analyzing Power Consumption 268

Chapter 6
Post-Processing 271

Working With a Bus 273
Creating a Bus 273
Editing Bus Contents 274
Transforming Analog Waveforms to Digital 275
Transforming Digital Waveforms to Analog 275
Bus and Bit Transformations 277
Bus Transformation Setup 277
Bit Transformation Setup 278
Measurement Tool 279
Taking a Measurement 279
Taking DC Hysteresis Measurements 281
Taking Eye Diagram Measurements 284
Frequency Domain Measurements 286
Measuring Bandpass 286
Measuring Gain Margin 287
Measuring Phase Margin 288
General Measurements 289
Measuring Average Value 289
Measuring Y Level Crossing 290
Measuring Intersect 291
Measuring Local Max 291
Measuring Local Min 292
Measuring Maximum Value 292
Measuring Minimum Value 293
Measuring Peak to Peak Value 293
Measuring Slope 294
Measuring Slope Intersect 295
Measuring YVal 295
Taking Statistical Measurements 296
Time Domain Measurements 299
Measuring Delay 299
Measuring Duty Cycle 301
Measuring Falltime 302
Measuring Frequency 303

Measuring Overshoot . . . 304
Measuring Period . . . 305
Measuring Pulse Width . . . 306
Measuring Risetime . . . 307
Measuring Settle Time . . . 308
Measuring Slew Rate . . . 309
Measuring Undershoot . . . 310
Waveform Calculator. . . . 312
Using Expressions in the Waveform Calculator. . . . 313
Waveform Calculator Shell Commands . . . 314
Using Buttons in the Waveform Calculator . . . 315
Using Built-In Functions in the Waveform Calculator. . . . 317
Using and Editing User-Defined Functions in the Waveform Calculator . . . 319
Creating a Tcl Script From the Waveform Calculator History. . . . 320
Updating User-Defined Function Help Documentation . . . 323
Using the Measurement Tool Functions in the Waveform Calculator . . . 325
Methods for Entering Measurement Tool Functions . . . 325
Available Measurement Tool Functions . . . 326
option and param Arguments . . . 330
Choosing the Occurrence of the Result . . . 331
Built-In Functions . . . 332
Complex Functions. . . . 332
Jitter Functions . . . 333
Logic Functions . . . 334
Mathematical Functions . . . 335
Measurement Functions . . . 336
Miscellaneous Functions . . . 338
Phase Noise Functions . . . 341
RF Functions . . . 342
Signal Processing Functions. . . . 343
Statistical Functions . . . 345
Trigonometric Functions . . . 346
Special Functions . . . 347
Calculator Buttons . . . 349
Complex Buttons . . . 349
Jitter Buttons. . . . 351
Logical Buttons . . . 352
Phase Noise Buttons. . . . 357
RF Buttons . . . 359
Signal Processing Buttons . . . 361
Statistical Buttons. . . . 363
Trigonometric Buttons . . . 365
Jitter Tool. . . . 368
Using the Jitter Tool . . . 368
Jitter Measurement Types. . . . 370
Absolute Jitter (Time-Domain) . . . 370
Period Jitter (Time-Domain) . . . 370
Half-Period Jitter (Time Domain) . . . 373
Frequency Jitter (Time Domain) . . . 375

Cycle-to-Cycle Jitter (Time-Domain) 377
Long-Term Jitter (Time-Domain) 379
Time Interval Error (TIE) 380
Jitter Confidence Interval 381
Absolute Jitter (Phase-Noise) 381
Period Jitter (Phase-Noise) 382
Cycle-to-Cycle Jitter (Phase-Noise) 382
Long-Term Jitter (Phase-Noise) 382
Extracting Outputs from a Database 382
Signal Processing Functions 384
Using the Fast Fourier Transform Tool 384
Using the Inverse Fast Fourier Transform Tool 386
Using the Spectrum Measurement Tool 388
Autocorrelation Function and Power Spectral Density 391
Correlogram Method 391
Periodogram Method 392
Convolution Function 392
Harmonic Distortion Function 394
Signal to Noise Function. 394
Signal Processing Function Window Shapes 395
Windowing Transforms 396
Using the DNA Advisor Tool 399

Chapter 7
Save and Output Data 403

Saving and Restoring Graph Windows 404
Saving Graph Windows 404
Restoring Graph Windows 405
Printing Graph Windows 405
Exporting Graph Windows 407
Exporting Graph Windows as a PDF 407
Using a Different PDF Driver. 408
Exporting Graph Windows as an Image 408
Saving a Waveform Database 410
Saving a Single Waveform Database 410
Saving Post-Processed Waveforms 411
Save File Types. 411
Saving Multiple Databases 412
Recovering Database Files. 414
Recovering From Incomplete Simulations 415
Recovering Incomplete Savefiles 416
Outputting a JWDB as an ASCII File 417
Saving a JWDB as an ASCII File. 417
Converting a JWDB to an ASCII File (Batch Mode). 418

Chapter 8
Dialog Box and Field Reference 419

Add Clock Dialog Box 422

Analog to Digital Conversion Dialog Box 423
Annotation Properties Dialog Box. 424
Axis Properties Dialog Box 427
Comparison Options Dialog Box. 429
General Options Tab 430
Comparison Methods Tab. 433
Tolerances Tab 435
Conversions Tab 437
Create Bus Dialog Box 439
Current Analysis Dialog Box. 441
Cursor Properties Dialog Box 445
Data Values Dialog Box 446
Database Properties Dialog Box 448
Digital to Analog Dialog Box 450
Event Search Tool Dialog Box 453
Edit Digital Transformation Dialog Box 455
Eye Diagram Tool Dialog Box 457
Eye Diagram Tool - Settings Tab 458
Eye Diagram Tool - Measurement Results Tab 465
Eye Mask Dialog Box. 470
Extract Outputs From Database Dialog Box 472
EZwave Display Preference Options. 474
EZwave Display Preferences Dialog Box 476
Automatic Reload Options 478
Cursor Options 480
Data Format Options. 482
Foreign Databases Options 484
General Options 485
Grid Options 488
Layout Options 490
Look and Feel Options 491
Mouse Pointer Options 492
Multiple Run Options 493
Pick Points Options. 495
RF Options 498
Row Options 499
Save Data Options 501
Save Window Options 503
Text Annotation Options 504
Transformation Options 505
Waveform Options 507
Waveform Compare Options 508
Waveform List Options 509
Workspace Options. 511
CDF Plot Options 512
CDF Measures Options. 514
CDF Legend Options 516
Axis Title Options. 518
Axis Values Options 519

Axis Values (Smith Chart) Options 520
Calculator Entry Options 521
Cursor/Marker Options 522
Eye Mask Options 524
Grid Options (Fonts and Colors) 525
Header Text Options 526
Histogram Options 527
Measurement Annotation Options 528
Pick Points Options (Fonts and Colors) 529
Row Title Options 530
Text Annotation Options (Fonts and Colors) 531
Waveform Colors Options 532
Waveform Display Options 533
Waveform Name Options 534
Waveform Selection Options 536
Window Background Options 537
Zero-Level Line Options 538
Histogram Plot Options 539
Histogram Measures Options 541
Histogram Legend Options 543
Waveform Calculator Calculation Options 545
Waveform Calculator General Options 547
Waveform Calculator View Options 549
Filter Dialog Box 550
Find Tool Dialog Box 553
Jitter Tool Dialog Box 555
Measurement Tool Dialog Box 560
Parameter Analyzer Dialog Boxes 563
Parameter Analyzer Tool Dialog Box 564
Filters Setup Dialog Box 567
Parameter Table Dialog Box 568
Pick Points Dialog Box 570
Power Analysis Dialog Box 576
Run Filter Dialog Box 580
Save As Dialog Box 583
Save Windows Dialog Box 586
Select Hierarchy Dialog Box 587
Select Waveforms Dialog Box 589
Shortcuts Manager Dialog Box 590
Update Waveforms Dialog Box 592
Waveform Names Display Dialog Box 594
Waveform Properties Dialog Box (For Analog Waveforms) 596
Waveform Properties Dialog Box (Digital Waveforms) 599
Waveform Calculator Dialog Boxes 601
Auto Correlation Dialog Box 602
Chirp Transform Dialog Box 605
Constellation Diagram Dialog Box 609
Convolution Dialog Box 611
Cross Correlation Dialog Box 613

DNA Advisor Dialog Box 615
Error Vector Magnitude and Bit Error Rate Dialog Box 617
Eye Diagram Dialog Box 619
Fast Fourier Transform Tool Dialog Box 621
Harmonic Distortion Dialog Box 626
Histogram Dialog Box 628
Inverse Fast Fourier Transform Dialog Box 630
Phase Noise Dialog Box 632
Power Spectral Density Dialog Box 634
PSS Residue Dialog Box 637
Signal to Noise Ratio Dialog Box 639
Spectrum Measurement Tool Dialog Box 642
Tcl File Viewer Dialog Box 648
Top Noise Dialog Box 650
Windowing Transform Dialog Box 652

Appendix A
Waveform Calculator Functions 655

Syntax Conventions 660
Function Descriptions 662
abs 666
absolutejitter 667
absolutejitterbyintegration 669
acos 670
acosh 671
acot 672
acoth 673
add. 674
allandeviation 676
analysisattributes. 677
analysisattributevalue 678
asin 679
asinh 680
assign 681
atan 682
atan2 683
atanh 684
atod 685
autocor 687
avg. 690
bandpass 691
bandwidth 692
baseline 693
calcvartype 694
cdf 695
ceil. 696
chirp 697
complex. 700

compress . . . 701
compresscompound . . . 702
concat . . . 703
conjugate . . . 704
constellationdiagram . . . 705
continuous . . . 706
convolution . . . 707
cos . . . 709
cosh . . . 710
cot . . . 711
coth . . . 712
cphase . . . 713
cphytrigger . . . 714
crosscorrelation . . . 716
crossing . . . 717
cycle2cyclejitter . . . 718
cycletocyclejitter . . . 720
cycletocyclejitterbyintegration . . . 721
datatocomplex . . . 722
datatodig . . . 723
datatowf . . . 725
dchysteresis . . . 726
db . . . 727
db10 . . . 728
deg . . . 729
delay . . . 730
derive . . . 732
drv . . . 733
dtoa . . . 734
dtoaonbit . . . 735
dutycycle . . . 737
edgephasenoise . . . 739
enob . . . 740
evmber . . . 742
exp . . . 743
exportcsv . . . 744
exportvcd . . . 745
eye . . . 747
eyeamplitude . . . 749
eyecphy . . . 750
eyecrossingamplitude . . . 752
eyedelay . . . 753
eyediagram . . . 754
eyefalltime . . . 756
eyeheight . . . 757
eyeheightatx . . . 758
eyejitter . . . 759
eyemeasures . . . 760
eyerisetime . . . 762

eyesetmask . . . 763
eyesnr . . . 765
eyewidth . . . 766
eyewidthaty . . . 767
eyewithtrigger . . . 768
falltime . . . 770
fft . . . 772
filterdupreal . . . 775
filterempty . . . 776
first . . . 777
firstdiff . . . 778
floor . . . 780
fmod . . . 781
frequency . . . 782
frequencyjitter . . . 784
frexp . . . 786
gainmargin . . . 787
gaussiandistribution . . . 788
gendecade . . . 789
genlinear . . . 790
genoctave . . . 791
getelementat . . . 792
getrunindices . . . 793
getrunparameters . . . 794
getrunparametervalue . . . 795
gettype . . . 796
gmargin . . . 797
gptocomplex . . . 798
groupdelay . . . 799
halfperiodjitter . . . 800
harm . . . 802
harmonicdistortion . . . 803
harmonics . . . 804
harmonicsmeter . . . 805
hdist . . . 806
histogram . . . 807
hypot . . . 808
idb . . . 809
idb10 . . . 810
ifft . . . 811
iipx . . . 813
imag . . . 814
integ . . . 815
integnoise . . . 816
integral . . . 817
intersect . . . 818
ipn . . . 820
jc . . . 823
jcc . . . 824

jee 825
join 826
larger 828
ldexp 829
last 830
length 831
lesser 832
ln 833
localmax 834
localmin 836
log 838
longtermjitter 839
mag 841
max 842
maxdiff 843
maxspectrumdiff 845
mean 847
meanminus3std 848
meanplus3std 849
min 850
modf 851
mptocomplex 852
nand 853
noisetosignaldbc 854
noisetrantophasenoise 855
nor 856
oipx 857
overshoot 858
peaktopeak 859
period 860
periodjitter 862
periodjitterbyintegration 864
phase 865
phasemargin 866
phasenoise 867
phasenoiseoutdbc 868
phnoisebydlm 869
phnoisebymixer 871
phmargin 872
pivot 873
plotjitterconfidenceinterval 874
pow10 875
psd 876
pssresidue 879
pulsewidth 880
rad 882
real 883
reglin 884
relation 885

removepts 886
risetime 887
ritocomplex 889
rms 890
rms_ac 891
rms_accurate 892
rms_noise 893
rms_tran 894
rol 895
ror 896
round 897
sample 898
samplelog 899
samplepsd 900
setAngleUnits 904
setNotation 905
setTemperatureUnits 906
settlingtime 907
sfdr 908
shift 910
shiftedmaxdiff 911
sin 913
sinad 914
sinh 916
size 917
sla 918
slewrate 919
slope 921
slopeintersect 922
sndr 923
snr 925
sorty 927
spectrummeasurement 928
sphibyjitter 931
sphifilter 933
sqr 934
sqrt 935
sra 936
stddev 937
sum 938
tan 939
tanh 940
thd 941
tiejitter 943
timeabsolutejitter 945
timelongtermjitter 947
timeperiodjitter 949
timestep 951
todchysteresis 952

topline 953
topnoise 954
trunc 955
undershoot 956
var 957
wavevswave 958
wf 959
wfattributes 961
wfname 962
wftoascii 963
wftodata 964
windavg 965
window 966
windowing 967
xcompress 969
xdown 970
xnor 971
xofmax 972
xofmin 973
xup 974
xval 975
xwave 976
xytowf 977
yval 978

Appendix B
Eldo Simulation 981

Run Eldo With EZwave 981
Complete Eldo Simulation and View Simulation Data Later 982
Manual Status Update 983
Marching Update 984
Joint Waveform Database Read API 984
AMS Results Browser 985

Appendix C
Tcl Scripting Support 987

Tcl Syntax 991
Tcl Scripting Example 991
Tcl Command Syntax Rules 991
If Command Syntax 995
Set Command Syntax 996
Command Substitution 996
Variable Substitution 997
Tcl References 997
Passing Tcl Parameters From the Command Line 999
Tcl List Processing 1000
Supported Tcl Commands 1003
Tcl Syntax Conventions 1003

Tcl Command Short Descriptions. 1004
Specifying Waveforms in Tcl 1012
Selecting Waveforms in Tcl 1014
Tcl Command Detailed Descriptions. 1018
add wave 1025
add workspace. 1032
batch_mode 1033
bloc 1034
compare add 1035
compare clock 1040
compare configure 1042
compare end 1046
compare info 1047
compare list. 1048
compare options 1049
compare run 1055
compare savelog 1056
compare saverules. 1057
compare start. 1058
dataset alias 1060
dataset analysis 1061
dataset clear. 1062
dataset close 1063
dataset extract 1064
dataset info 1065
dataset list 1066
dataset merge 1067
dataset mergewaveforms. 1069
dataset open. 1071
dataset ovd 1072
dataset power analysis 1073
dataset rename. 1076
dataset save 1077
dataset savescalar 1079
dataset savewaveforms 1080
dataset scalar 1082
dataset statistics. 1084
dataset supported. 1086
dataset topnoise. 1087
delete wave 1089
dofile 1090
environment 1091
evalExpression 1092
examine 1093
exit 1095
find analogs. 1096
find currents 1097
find digitals 1098
find nets | signals. 1099

getactivecursortime 1100
precision 1101
printenv 1102
quit 1103
radix 1104
radix define 1106
radix delete 1108
radix list 1109
radix names 1110
radix signal 1111
save 1112
setenv 1113
unsetenv 1114
wave activecursor 1115
wave activewindow 1116
wave activeworkspace 1117
wave addannotation 1118
wave addaxisdeltamarker 1120
wave addcursor 1122
wave adddeltamarker 1124
wave addline 1126
wave addmarker 1128
wave addproperty 1129
wave addwindow 1130
wave addworkspace 1131
wave cdf 1132
wave clean 1136
wave closewindow 1137
wave closeworkspace 1138
wave colortheme 1139
wave compressvri 1140
wave createbus 1142
wave cursortime 1143
wave deletecursor 1144
wave difference 1145
wave displayed 1146
wave exists 1147
wave ezwave_title 1149
wave gettype 1150
wave histogram 1151
wave ipnvri 1155
wave jitter 1158
wave launchfolder 1163
wave listworkspace 1164
wave loadbindings 1165
wave lockcursor 1166
wave names 1167
wave plotcompress 1168
wave plotpacipn 1170

wave refresh 1172
wave rowfit 1173
wave runindexlist 1174
wave runparameters 1175
wave runparametervalue 1176
wave show. 1177
wave showgridlines. 1178
wave showzerolevels 1179
wave tandem mode 1180
wave tile 1181
wave windowdecoration 1182
wave windowlist 1183
wave xaxis. 1184
wave yaxis. 1185
wave zoomfull. 1187
wave zoomin 1188
wave zoomlast. 1189
wave zoomout 1190
wave zoomrange 1191
wfc 1192
write jpeg 1194
write png 1195
write wave. 1196
External Tcl Command Support 1197
Tcl Scripting Examples 1205
Tcl Waveform Calculator Batch Commands Example. 1206
Additional User-Defined Procedures 1208
Opening a Database File. 1208
Setting Global Parameters 1208
Taking Waveform Measurements 1209
ASCII File Output 1210
Tcl User-Defined Functions 1211
Creating a User-Defined Tcl Function 1211
Loading User-Defined Tcl Functions Automatically. 1211
Waveform Comparison With Tcl Examples. 1213
Compare All Waveforms With Default Options 1213
Compare All Waveforms and Scalars 1213
Compare All Terminal Waveforms From Transient Analysis With Default Options . . . 1214
Compare All Waveforms Using a Clocked Comparison 1214
Compare Specific Waveforms With Modified Tolerances, Write a Report 1215
Compare Waveforms Using All Available Comparison Commands. 1215
Delaying Reference Waveforms During Comparison 1215
Delaying Some Result Waveforms (Not All) During Comparison 1216
Compare Waveforms Using the -label Argument 1217
Compare Waveforms Using the -start and -end Arguments 1218
Export a Report of Comparison 1219
Using find Commands 1220
Waveform Calculator Example Tcl Scripts 1224
adc_sndr 1226

adc_sndr_parallel4 1227
clock_jitter 1228
compressvri 1229
IPnVRI 1230
pll_jitter 1233
pll_jitter_parallel4 1235
pll_phasenoise 1236
pll_phasenoise_parallel4 1238
plotcompress 1240
plotpacipn 1242

Appendix D
Supported Net Representation Components 1245

Representing the Signal as a Waveform 1246
Extended Options for Selecting Waveforms 1246
representation 1248
analysis 1248
discipline 1248
physic 1250
Examples of wave show and -show usage 1251

Glossary

Appendix F
Troubleshooting 1261

Known Problems and Workarounds 1261
Linux Printing Issues 1264
Resolving Linux Printing Issues 1264
Printing Issues When Using LPRNG 1264
Printing Issues When Using CUPS 1265
Troubleshooting Memory Issues 1267
Configuring Memory Capacity 1267
Resolving Out-Of-Memory Problems 1268
Resolving Incorrect Estimate the Disk Space Needed to Save A Database 1270
Resolving Why the Simulator Fails to Start EZwave 1270
Increasing the Memory Stack Limit 1271
Loading .fsdb and .tr0 Files 1272
Troubleshooting EZwave Launch Issues 1273
Failure to Load EZwave Dynamic Libraries 1273
Failure to Launch EZwave in Questa ADMS GUI Context 1273
Displaying Results Between Different AMS Versions 1274
Troubleshooting Display Issues 1275
Fonts 1275
Troubleshooting Logfiles 1276
Logfile Locations 1276
Troubleshooting Waveform Comparison 1277
System Error Codes 1277
Contacting the Customer Support Center 1282

Training Classes 1282

Index

Third-Party Information

**End-User License Agreement
with EDA Software Supplemental Terms**

List of Figures

Figure 1-1. EZwave Viewer Graphical User Interface 49
Figure 1-2. Cursors 50
Figure 3-1. EZwave Main Window 68
Figure 3-2. Waveform List Panel 84
Figure 3-3. Database Sort Buttons 86
Figure 3-4. Waveform List Panel Showing Measures Sorted by Value 87
Figure 3-5. Waveform List Panel Showing Waveform Sorted by Mode 88
Figure 3-6. EZwave Graph Window 91
Figure 3-7. Workspace Tabs and Popup Menu 92
Figure 3-8. Workspace Taskbar. 93
Figure 3-9. Popup Menus. 95
Figure 3-10. X and Y Axis Popup Menus 100
Figure 3-11. Vertical Cursor Popup Menu 102
Figure 3-12. Folder Popup Menu. 109
Figure 3-13. Hierarchy Popup Menu 111
Figure 3-14. Waveform Name Popup Menu 112
Figure 3-15. Selected Waveforms Popup Menu 113
Figure 3-16. Waveform Calculator with Button and Function Help Panels 117
Figure 4-1. Expanded Waveform Group 132
Figure 4-2. Overlaid Waveforms With Different Y-Axes Scales 136
Figure 4-3. Realigned Y Axes 137
Figure 4-4. Step Waveform 143
Figure 4-5. Railroad Waveform. 143
Figure 4-6. Multiple Bit Waveforms as a Bus 144
Figure 4-7. Bus Display with Value Strings 144
Figure 4-8. Graph Window With a VHDL-AMS Assertion 146
Figure 4-9. Graph Window With an SOA Assertion. 147
Figure 4-10. Example CDF Plot 149
Figure 4-11. Expanded Compound Waveform Names With Run Parameters 152
Figure 4-12. Compound Waveforms as Single Elements With Run Parameters 153
Figure 4-13. Multiple Overlaid Waveforms Following a Waveform Update 162
Figure 5-1. The Base Cursor 166
Figure 5-2. Reference Cursors 168
Figure 5-3. Horizontal Cursor 170
Figure 5-4. Show/Hide Crossing Points in the Data Values Dialog Box 170
Figure 5-5. EZwave Display Preferences - Horizontal Cursor in Y Axis 172
Figure 5-6. Cursor Values Displayed in the Reserved Area 179
Figure 5-7. Y-Level Lines 180
Figure 5-8. Waveform-Based Delta Ys 181
Figure 5-9. Cursor-Based Delta Ys 182

Figure 5-10. Cursor Value Table 183
Figure 5-11. Adding Deltas From the Pick Points Dialog Box 188
Figure 5-12. Add Annotation Dialog Box 190
Figure 5-13. Text Annotation Examples 191
Figure 5-14. Annotation Properties 192
Figure 5-15. Rising Edge Dialog 194
Figure 5-16. C-Phy Eye Diagram Example 200
Figure 5-17. NRZ Eye Diagram Example 201
Figure 5-18. PAM4 Eye Diagram Example. 202
Figure 5-19. Measurement Results Dialog Box - PAM4 Example 202
Figure 5-20. Left and Right Crossing Points and Vertical Band 206
Figure 5-21. PAM4 Waveform 209
Figure 5-22. PAM4 Eye Diagram 209
Figure 5-23. PAM4 Eye Levels 210
Figure 5-24. PAM4 Height and Width 210
Figure 5-25. PAM 4 Eye Linearity and RLM 212
Figure 5-26. Differential Lines and Zero Level. 214
Figure 5-27. Differential Lines, Zero Level, and Trigger (Rising Edges). 214
Figure 5-28. Differential Lines, UI jitter and Transition Jitter 215
Figure 5-29. C-Phy Eye Diagram with Eye Mask 216
Figure 5-30. C-Phy Eye Diagram and Triggered Eyes with Other Alignments 217
Figure 5-31. Smith Chart Impedance Display 221
Figure 5-32. Smith Chart Admittance Display 221
Figure 5-33. Smith Chart Impedance and Admittance Display. 222
Figure 5-34. Smith Chart with Negative Real Values 223
Figure 5-35. Polar Chart Display. 224
Figure 5-36. Cursor in a Smith Chart 225
Figure 5-37. Setting Data Values. 225
Figure 5-38. Multiple-Circle Plot and Circle Visibility Option. 227
Figure 5-39. Circle Visibility Table. 227
Figure 5-40. Highlighted Circle. 228
Figure 5-41. The Waveform Compare Wizard Process Flow 233
Figure 5-42. Waveform Comparison Results 234
Figure 5-43. Waveform Comparison Results 240
Figure 5-44. Stepping Through Waveform Differences With a Cursor 241
Figure 5-45. Display Tolerance Tube Menu Item 242
Figure 5-46. Graph Window Showing Tolerance Tube. 242
Figure 5-47. Waveform Compare Showing Tolerance 254
Figure 5-48. Calculations on the Reference Waveform. 258
Figure 5-49. Tolerance “Tube” 259
Figure 5-50. Data Points Outside of the Tolerance Tube 259
Figure 5-51. Display of the Tolerance Tube 260
Figure 6-1. Bus Transformation Setup Options. 277
Figure 6-2. Bit Transformation Setup Options 278
Figure 6-3. DC Hysteresis Measurement Results 282

Figure 6-4. Hysteresis Width 283
Figure 6-5. Left Threshold. 283
Figure 6-6. Right Threshold 283
Figure 6-7. Example of Min and Max width (multiple runs) 284
Figure 6-8. Hysteresis Width versus run parameter RVAL 284
Figure 6-9. Calculation of the Average Value of a Waveform 290
Figure 6-10. Mean Value of a Waveform Calculation 297
Figure 6-11. The RMS AC Calculation: 297
Figure 6-12. The RMS Noise Calculation: 297
Figure 6-13. The RMS Tran Calculation: 298
Figure 6-14. Waveform Calculator Functions List 318
Figure 6-15. Waveform Calculator Function Fields 318
Figure 6-16. Create Script Dialog Box 321
Figure 6-17. Update UDF Help Dialog Box 324
Figure 6-18. Example of the Period Jitter Waveform 372
Figure 6-19. Example of the Period Jitter Waveform with Applied Statistical Measurements373
Figure 6-20. Example of the Period Jitter Histogram 373
Figure 6-21. Example of the Long-Term Jitter Waveform 380
Figure 6-22. Correlogram Method. 391
Figure 6-23. Periodogram Method. 392
Figure 6-24. Symmetric Window 396
Figure 6-25. Periodic Window. 396
Figure 6-26. Available Windowing Transforms 397
Figure 6-27. DNA Advisor Waveform Plot. 400
Figure 6-28. DNA Advisor Frequency Plot. 401
Figure 6-29. DNA Advisor Report 401
Figure 7-1. Editing the Command: Field 408
Figure 8-1. Annotation Properties Dialog Box - Font and Color Tab 424
Figure 8-2. Annotation Properties Dialog Box - Appearance Tab 425
Figure 8-3. Axis Properties Dialog Box 427
Figure 8-4. Comparison Options Dialog Box - General Options Tab 430
Figure 8-5. Comparison Options Dialog Box - Comparison Methods Tab. 433
Figure 8-6. Comparison Options Dialog Box - Tolerances Tab 435
Figure 8-7. Comparison Options Dialog Box - Conversions Tab 437
Figure 8-8. Create Bus Dialog Box 439
Figure 8-9. Current Analysis Dialog Box - Detailed View Tab 441
Figure 8-10. Data Values Dialog Box 446
Figure 8-11. Database Properties Dialog Box - Statistics Tab 448
Figure 8-12. Database Properties Dialog Box - Properties Tab. 449
Figure 8-13. Digital to Analog Dialog Box 450
Figure 8-14. Event Search Tool Dialog Box 453
Figure 8-15. Edit Digital Transformation Dialog Box 455
Figure 8-16. Eye Diagram Tool Dialog Box - Settings Tab for C-Phy Eye Type. 458
Figure 8-17. Eye Diagram Tool Dialog Box - Settings Tab for NRZ Eye Type 459
Figure 8-18. Eye Diagram Tool Dialog Box - Settings Tab for PAM 3 and PAM 4 Eye Types

460
Figure 8-19. Eye Diagram Tool Dialog Box - Measurement Results Tab for NRZ 465
Figure 8-20. Eye Diagram Tool Dialog Box - Measurement Results Tab for PAM3 466
Figure 8-21. Eye Diagram Tool Dialog Box - Measurement Results Tab for PAM4 467
Figure 8-22. Eye Diagram Tool Dialog Box - Measurement Results Tab for C-Phy 468
Figure 8-23. Eye Mask Dialog Box 470
Figure 8-24. Extract Outputs From Database Dialog Box. 472
Figure 8-25. EZwave Display Preferences Dialog Box. 476
Figure 8-26. EZwave Display Preferences - Calculation Options 545
Figure 8-27. Filter Dialog Box With List Option Selected 550
Figure 8-28. Filter Dialog Box With Condition Option Selected 551
Figure 8-29. Find Tool Dialog Box 553
Figure 8-30. Jitter Tool Dialog Box. 555
Figure 8-31. Jitter Tool Dialog Box - Phase Noise Tab 556
Figure 8-32. Measurement Tool Dialog Box. 560
Figure 8-33. Parameter Analyzer Dialog Box - Plot Setup Tab 564
Figure 8-34. Parameter Analyzer Dialog Box - Advanced Tab. 565
Figure 8-35. Filters Setup Dialog Box. 567
Figure 8-36. Parameter Table Dialog Box. 568
Figure 8-37. Pick Points Dialog Box 570
Figure 8-38. Power Analysis Dialog Box 576
Figure 8-39. Run Filter Dialog Box. 580
Figure 8-40. Save As Dialog Box 583
Figure 8-41. Select Hierarchy Dialog Box 587
Figure 8-42. Select Waveforms Dialog Box 589
Figure 8-43. Shortcuts Manager Dialog Box. 590
Figure 8-44. Update Waveforms Dialog Box 592
Figure 8-45. Waveform Names Display Dialog Box 594
Figure 8-46. Waveform Properties Dialog Box - Appearance Tab for Analog Waveforms. 596
Figure 8-47. Waveform Properties Dialog Box - Appearance Tab for Digital Waveforms . 599
Figure 8-48. Auto Correlation Dialog Box 602
Figure 8-49. Chirp Transform Dialog Box 605
Figure 8-50. Constellation Diagram Dialog Box. 609
Figure 8-51. Convolution Dialog Box 611
Figure 8-52. Cross Correlation Dialog Box. 613
Figure 8-53. DNA Advisor Dialog Box. 615
Figure 8-54. Error Vector Magnitude and Bit Error Rate Dialog Box 617
Figure 8-55. Eye Diagram Dialog Box 619
Figure 8-56. Fast Fourier Transform Tool Dialog Box 622
Figure 8-57. Harmonic Distortion Dialog Box 626
Figure 8-58. Histogram Dialog Box 628
Figure 8-59. Inverse Fast Fourier Transform Dialog Box 630
Figure 8-60. Phase Noise Dialog Box 632
Figure 8-61. Power Spectral Density Dialog Box 634
Figure 8-62. PSS Residue Dialog Box. 637

Figure 8-63. Signal to Noise Ratio Dialog Box 639
Figure 8-64. Spectrum Measurement Tool Dialog Box- Settings Tab 643
Figure 8-65. Spectrum Measurement Tool Dialog Box - Measurement Results Tab 644
Figure 8-66. Tcl File Viewer Dialog Box 648
Figure 8-67. Top Noise Dialog Box 650
Figure 8-68. Windowing Transform Dialog Box 652
Figure A-1. Example of add() and join() Waveform Calculator Functions. 675
Figure A-2. Example of add() and join() Waveform Calculator Functions. 827
Figure A-3. Calculation of SamplePSD. 901
Figure A-4. Example Output Noise Spectrum Comparison 902
Figure C-1. Graphical Representation of tolLead and tolTrail 1036
Figure C-2. Graphical Representation of tolLead and tolTrail 1043
Figure C-3. Graphical Representation of tolLead and tolTrail 1050
Figure C-4. Example CDF with Upper and Lower Confidence Bounds 1133
Figure C-5. Example Plot for Left Tail 1134
Figure C-6. Example CDF Plot 1135
Figure C-7. Example Plot showing wave histogram Probabilities 1153

List of Tables

Table 1-1. Supported File Types 45
Table 2-1. Environment Variables 54
Table 2-2. EZwave Display Preferences 61
Table 2-3. Fonts and Colors Preferences 62
Table 3-1. File Menu Items 70
Table 3-2. Edit Menu Items 71
Table 3-3. View Menu Items 74
Table 3-4. Format Menu Items 75
Table 3-5. Tools Menu Items 76
Table 3-6. Cursor Menu Items 78
Table 3-7. Window Menu Items 79
Table 3-8. Help Menu Items 80
Table 3-9. Toolbar Icons 81
Table 3-10. Waveform List Element Icons 85
Table 3-11. Waveform List Filter Options 88
Table 3-12. Graph Window Popup Menus 95
Table 3-13. Graph Window Popup Menu Items 96
Table 3-14. Row Popup Menu Items 96
Table 3-15. Waveform Popup Menu Items 98
Table 3-16. Axis Popup Menu Items 100
Table 3-17. Vertical Cursor Popup Menu Items 102
Table 3-18. Horizontal Cursor Popup Menu Items 104
Table 3-19. Cursor Value Popup Menu Items 105
Table 3-20. Cursor Value Popup Menu Items for Compound Waveforms 105
Table 3-21. Right-Click Popup Menus 106
Table 3-22. Waveform List Popup Menu Items 106
Table 3-23. Database Popup Menu Items 108
Table 3-24. Folder Popup Menu Items 110
Table 3-25. Hierarchy Popup Menu Items 111
Table 3-26. Waveform Name Popup Menu Items 112
Table 3-27. Selected Waveforms Popup Menu Items 113
Table 3-28. Workspace Popup Menu Items 113
Table 3-29. Frequently Used Keyboard Commands 115
Table 3-30. Calculator Shell Key Binds 118
Table 3-31. Calculator Shell Mouse Operations 119
Table 4-1. Graphic Elements for Assertions 145
Table 5-1. NRZ Eye Diagram Measurements 207
Table 5-2. PAM4 Eye Diagram Measurements 212
Table 5-3. C-Phy Eye Mask 217
Table 6-1. DC Hysteresis Measurements 281

Table 6-2. Statistical Measurements 297
Table 6-3. Arguments for Setting Occurrence 331
Table 6-4. Built-In Complex Functions 332
Table 6-5. Built-In Jitter Functions 333
Table 6-6. Built-In Logic Functions 334
Table 6-7. Built-In Mathematical Functions 335
Table 6-8. Built-In Measurement Functions 337
Table 6-9. Built-In Miscellaneous Functions 338
Table 6-10. Built-In Phase Noise Functions 342
Table 6-11. Built-In RF Functions 342
Table 6-12. Built-In Signal Processing Functions 344
Table 6-13. Built-In Statistical Functions 345
Table 6-14. Built-In Trigonometric Functions 346
Table 6-15. Built-In Statistical Functions 347
Table 6-16. Complex Function Buttons 349
Table 6-17. Jitter Buttons 351
Table 6-18. Logical Buttons 352
Table 6-19. Phase Noise Buttons 358
Table 6-20. RF Buttons 359
Table 6-21. Signal Processing Buttons 361
Table 6-22. Statistical Buttons 363
Table 6-23. Trigonometric Buttons 365
Table 6-24. Zero Padding 393
Table 6-25. Multiply FFT 393
Table 6-26. Inverse FFT 393
Table 8-1. Add Clock Dialog Box Contents 422
Table 8-2. Analog to Digital Conversion Dialog Box Contents 423
Table 8-3. Annotation Properties Dialog Box - Font and Color Tab Contents 424
Table 8-4. Annotation Properties Dialog Box - Appearance Tab Contents 426
Table 8-5. Axis Properties Dialog Box Contents 427
Table 8-6. Comparison Options Dialog Box - General Options Tab Contents 430
Table 8-7. Comparison Options Dialog Box - Comparison Methods Tab Contents 433
Table 8-8. Comparison Options Dialog Box - Tolerances Tab Contents 435
Table 8-9. Comparison Options Dialog Box - Conversions Tab Contents 437
Table 8-10. Create Bus Dialog Box Contents 439
Table 8-11. Current Analysis Dialog Box Contents 442
Table 8-12. Cursor Properties 445
Table 8-13. Data Values Dialog Box Contents 446
Table 8-14. Digital to Analog Dialog Box Contents 450
Table 8-15. Bit Transformation Digital and Analog Default Values 451
Table 8-16. Event Search Tool Dialog Box Contents 454
Table 8-17. Edit Digital Transformation Dialog Box Contents 455
Table 8-18. Eye Diagram Tool Dialog Box - Settings Tab Contents 460
Table 8-19. Eye Mask Dialog Box Contents 470
Table 8-20. Extract Outputs From Database Dialog Box Contents 472

Table 8-21. EZwave Preferences 476
Table 8-22. CDF Preferences 477
Table 8-23. Fonts and Colors 477
Table 8-24. Histogram Preferences 477
Table 8-25. Waveform Calculator Preferences 477
Table 8-26. EZwave Display Preferences - Automatic Reload Options 478
Table 8-27. EZwave Display Preferences - Cursor Options 480
Table 8-28. EZwave Display Preferences - Data Format Options 482
Table 8-29. EZwave Display Preferences - General Options 485
Table 8-30. EZwave Display Preferences - Grid Options 488
Table 8-31. EZwave Display Preferences - Layout Options 490
Table 8-32. EZwave Display Preferences - Look and Feel Options 491
Table 8-33. EZwave Display Preferences - Mouse Pointer Options 492
Table 8-34. EZwave Display Preferences - Multiple Run Options 493
Table 8-35. EZwave Display Preferences - Pick Points Options 495
Table 8-36. EZwave Display Preferences - RF Options 498
Table 8-37. EZwave Display Preferences - Row Options 499
Table 8-38. EZwave Display Preferences - Save Data Options 501
Table 8-39. EZwave Display Preferences - Save Window Options 503
Table 8-40. EZwave Display Preferences - Text Annotation Options 504
Table 8-41. EZwave Display Preferences - Transformations - Transformations Tab 505
Table 8-42. EZwave Display Preferences - Transformations - Conversion Tab 506
Table 8-43. EZwave Display Preferences - Waveform Options 507
Table 8-44. EZwave Display Preferences - Waveform Compare Options 508
Table 8-45. EZwave Display Preferences - Waveform List Options 509
Table 8-46. EZwave Display Preferences - Workspace Options 511
Table 8-47. EZwave Display Preferences - CDF Plot Options Dialog Box Contents 512
Table 8-48. EZwave Display Preferences - CDF Measures Options Dialog Box Contents . 514
Table 8-49. EZwave Display Preferences - CDF Legend Options Dialog Box Contents . . 516
Table 8-50. EZwave Display Preferences - Axis Title Options 518
Table 8-51. Fonts and Colors - Axis Values Options 519
Table 8-52. EZwave Display Preferences - Axis Values (Smith Chart) Options 520
Table 8-53. EZwave Display Preferences - Calculator Entry Options 521
Table 8-54. EZwave Display Preferences - Cursor/Marker Options 522
Table 8-55. EZwave Display Preferences - Eye Mask Options 524
Table 8-56. EZwave Display Preferences - Grid Options 525
Table 8-57. EZwave Display Preferences - Header Text Options 526
Table 8-58. EZwave Display Preferences - Histogram Options 527
Table 8-59. EZwave Display Preferences - Measurement Annotation Options 528
Table 8-60. EZwave Display Preferences - Pick Points Options (Fonts and Colors) 529
Table 8-61. Fonts and Colors - Row Title Dialog Box Contents 530
Table 8-62. EZwave Display Preferences - Text Annotation Options (Fonts and Colors) . 531
Table 8-63. EZwave Display Preferences - Waveform Colors Options 532
Table 8-64. Fonts and Colors - Waveform Display Dialog Box Contents 533
Table 8-65. Fonts and Colors - Waveform Name Dialog Box Contents 534

Table 8-66. EZwave Display Preferences - Waveform Selection Options 536
Table 8-67. Fonts and Colors - Window Background Dialog Box Contents 537
Table 8-68. Fonts and Colors - Zero-Level Line Dialog Box Contents 538
Table 8-69. EZwave Display Preferences - Histogram Plot Options Dialog Box Contents . 539
Table 8-70. EZwave Display Preferences - Histogram Measures Options Dialog Box Contents 541
Table 8-71. EZwave Display Preferences - Histogram Legend Options Dialog Box Contents 543
Table 8-72. EZwave Display Preferences - Waveform Calculation Options 545
Table 8-73. EZwave Display Preferences - Waveform Calculator General Options 547
Table 8-74. EZwave Display Preferences - Waveform Calculator View Options 549
Table 8-75. Filter Dialog Box Contents 551
Table 8-76. Find Tool Dialog Box Contents 553
Table 8-77. Jitter Tool Dialog Box Contents 556
Table 8-78. Measurement Tool Dialog Box Contents 560
Table 8-79. Parameter Table Dialog Box Contents 568
Table 8-80. Pick Points Dialog Box Contents 570
Table 8-81. Expression Calculations 574
Table 8-82. Power Analysis Dialog Box Contents 577
Table 8-83. Run Filter Dialog Box Contents 580
Table 8-84. Save As Dialog Box Contents 583
Table 8-85. Save Windows Dialog Box Contents 586
Table 8-86. Select Hierarchy Dialog Box Contents 587
Table 8-87. Select Waveforms Dialog Box Contents 589
Table 8-88. Shortcuts Manager Dialog Box Contents 591
Table 8-89. Update Waveforms Dialog Box Contents 592
Table 8-90. Waveform Names Display Dialog Box Contents 594
Table 8-91. Appearance Tab for Analog Waveforms Contents 596
Table 8-92. Appearance Tab for Digital Waveform Contents 599
Table 8-93. Auto Correlation Dialog Box Contents 603
Table 8-94. Chirp Transform Dialog Box Contents 606
Table 8-95. Constellation Diagram Dialog Box Contents 609
Table 8-96. Convolution Dialog Box Contents 611
Table 8-97. Cross Correlation Dialog Box Contents 613
Table 8-98. DNA Advisor Dialog Box Contents 615
Table 8-99. Error Vector Magnitude and Bit Error Rate Dialog Box Contents 618
Table 8-100. Eye Diagram Dialog Box Contents 619
Table 8-101. Fast Fourier Transform Tool Dialog Box Contents 623
Table 8-102. Harmonic Distortion Dialog Box Contents 626
Table 8-103. Histogram Dialog Box Contents 628
Table 8-104. Inverse Fast Fourier Transform Dialog Box Contents 630
Table 8-105. Phase Noise Dialog Box Contents 632
Table 8-106. Power Spectral Density Dialog Box Contents 635
Table 8-107. PSS Residue Dialog Box Contents 637
Table 8-108. Signal to Noise Dialog Box Contents 640

Table 8-109. Spectrum Measurement Tool Dialog Box Contents 644
Table 8-110. Tcl File Viewer Dialog Box Contents 648
Table 8-111. Top Noise Dialog Box Contents 650
Table 8-112. Windowing Transform Dialog Box Contents 652
Table A-1. Complex Functions 662
Table A-2. Jitter Functions 662
Table A-3. Logic Functions 662
Table A-4. Mathematical Functions 662
Table A-5. Measurement Functions 663
Table A-6. Miscellaneous Functions 663
Table A-7. Phase Noise Functions 663
Table A-8. RF Functions 664
Table A-9. Signal Processing Functions 664
Table A-10. Statistical Functions 664
Table A-11. Trigonometric Functions 664
Table A-12. Special Functions 665
Table A-13. NAND Truth Table 853
Table A-14. NOR Truth Table 856
Table A-15. XNOR Truth Table 971
Table C-1. Tcl Backslash Sequences 993
Table C-2. Tcl List Processing Commands 1000
Table C-3. Tcl Command Topics 1003
Table C-4. Supported Tcl Commands 1004
Table C-5. Commands that Access Waveforms 1017
Table C-6. Tcl Commands 1018
Table C-7. wave difference Error Messages 1145
Table C-8. wave runindexlist Error Messages 1174
Table C-9. wave runparameters Error Messages 1175
Table C-10. wave runparametervalue Error Messages 1176
Table C-11. Questa SIM Command Support 1197
Table C-12. Questa ADMS Command Support 1203
Table C-13. pllphasenoise Output Variables 1237
Table D-1. Commands that Access Waveforms 1247
Table D-2. Supported analysis Values 1248
Table D-3. Supported discipline Values 1248
Table D-4. Supported primary_physic Values 1250
Table F-1. JVM Memory Heap Error Messages 1271
Table F-2. Characters to Avoid in Logfiles 1276
Table F-3. System Error Codes 1277

Chapter 1
Introduction

This section presents an overview of the functionality and capability of the EZwave™ viewer, as well as detailing basic concepts relating to the tool and this manual.

EZwave Overview **38**
EZwave Process Flow 38
Joint Waveform Database (JWDB) 39
Invoking EZwave **41**
Invoking EZwave in Batch Mode **45**
Supported File Types **45**
Visual Tour of EZwave **48**

EZwave Overview

The EZwave viewer is an advanced graphical user interface (GUI) that supports viewing of multiple databases of analog, digital and mixed-signal waveforms. It provides a dynamic graphical display of data produced by a variety of Mentor Graphics applications.

The EZwave viewer can be run from the command line or from within host applications, such as the Eldo® and Questa® ADMS™ simulators, Analog FastSPICE™ (AFS), a schematic capture tool such as Pyxis®, or other design environment tools.

Once you have loaded your simulation results into the EZwave viewer, you can view, analyze, and post-process them using a range of utilities, such as the Waveform Measurement Tool and the Waveform Calculator. These utilities enable you to perform sophisticated calculations with a combination of built-in or user-defined arithmetic (such as log, sin, cos) and logical (such as AND, OR, XOR) functions. The calculation results can be waveforms, vectors, or scalar values.

A separate manual, *Getting Started with EZwave*, is available, which also links to video tutorials. In addition, there is a short *EZwave Quick Reference*, which provides a summary of the key functions of EZwave, including a tour of the main window features and tools, actions available in the Wave windows, and keyboard shortcuts.

Tip

You can also view and analyze simulation results and associated files in tabular form using the AMS Results Browser. Refer to the *AMS Results Browser User's Manual*.

EZwave Process Flow **38**
Joint Waveform Database (JWDB) **39**

EZwave Process Flow

The EZwave viewer usage is organized into five distinct stages, described here.

1. **Set Up and Load Data** — Set up the EZwave viewer and open a database containing waveform data.

   See "Setting Up EZwave" on page 53.

2. **Add Waveforms** — Add or select specific waveforms for viewing and analysis. These waveforms can be stacked, overlaid, zoomed, and annotated through the use of keyboard shortcuts, drag and drop, mouse clicks and strokes, and menu items.

   See "Add Waveforms" on page 121.

3. **Analysis** — Measure, analyze, and annotate datapoints or logic units that are represented in the waveforms. You can save data from multiple simulations for

additional analysis and reuse. Add cursors to show data points and interpolated values between data points.

See "Analysis" on page 163.

4. **Post-Processing** — After analyzing the simulator output data, the EZwave viewer provides a number of powerful utilities for processing the data and transforming the raw data to specific characteristic information. Using the Measurement Tool and the Waveform Calculator, you can perform sophisticated calculations with a combination of built-in or user-defined arithmetic (such as log, sin, cos) and logical (such as AND, OR, XOR) functions. The calculation results can be waveforms, vectors, or scalar values.

   See "Post-Processing" on page 271.

5. **Save and Output Data** — Save and/or output results to a disk file.

   See "Save and Output Data" on page 403.

Related Topics

Setting Up EZwave

Add Waveforms

Analysis

Post-Processing

Save and Output Data

Joint Waveform Database (JWDB)

The EZwave viewer obtains waveform data by loading a database. By default, the EZwave viewer uses the Joint Waveform DataBase (*.wdb* file) as its input format. Waveform data that is generated by an analog/mixed-signal (AMS) simulation is stored in the JWDB. From here it can be loaded into the EZwave viewer to view a single database or multiple databases in a single session.

JWDB is a true mixed-signal waveform database. It can hold many different waveform types, including:

- analog (float, double, or complex)
- histogram
- spectral
- scatter
- Verilog
- standard logic

- VHDL char
- buses and records
- bit
- boolean
- string
- integer (16, 32, or 64 bits)
- user-defined enumerated types

X values can either be 64-bit integers or double-precision floating-point numbers. It can contain signals from the time and frequency domains, or any other required domain.

The JWDB is a multi-run database. Waveforms and buses are stored, managed, and analyzed as Compound Waveforms. In addition to compound waveforms, the JWDB has hierarchies which allow waveforms to be placed in folders for further data management.

Tip

Refer to the *Joint Waveform Database Read API Reference Manual*.

Related Topics

Visual Tour of EZwave

EZwave Process Flow

Invoking EZwave

To invoke the EZwave viewer, type **ezwave** at the command line.

This topic describes the **ezwave** command, used to invoke EZwave. Lists all supported files types that can be opened in EZwave.

Tip

The EZwave viewer can also be invoked from host applications such as the Questa ADMS simulator, the Pyxis® schematic capture tool, or other design environment tools. For more information, refer to the documentation provided with the host application.

Syntax

ezwave [*arguments*] [*file1* ...]

ezwave [-nos | -nosplash]
[-v | -version]
[-hei <#> | -height <#>]
[-w <#> | -width <#>]
[-loc <x# y#> | -location <x# y#>]
[-title <"title">]
[--laf <light | gray | dark>]
[-c]
[-hel | -help | -u | -usage]
[-do <filename>]
[-args <args>]
[-assign <variable=value>]
[-i <path>]
[-o <path>]
[-s]
[-tcl | -tclprompt]
[-bigmem]
[-nobigmem]
[-maxwnd | -maximizefirstinnerwindow]
[-now | -nowindow]
[-notree | -forcewfchoosernotvisible]
[-singlewk | -forceworkspacenotallowmultiple]
[-skipexprerror]
[-log <path> | -logfile <path>]
[-nologging]
[-nor | -norestore]
[<file1> ...]

Arguments

- -nos | -nosplash

  Prevents the splash screen from opening when EZwave launches. If specified, this must be the first argument.

- -v | -version

  Displays the EZwave version number.

- -hei <#> | -height <#>

  Sets the default height of the application windows.

- -w <#> | -width <#>

  Sets the default width of the application windows.

- -loc <x# y#> | -location <x# y#>

  Sets the location of the initial window.

- -title <"title">

  Adds a title to the top of the EZwave main display. If the option *-title "My Custom Title"* is set, the displayed name will be "My Custom Title (EZwave #Version#)".

- --laf <light | gray | dark>

  Note

  Early-access feature. This feature has had limited production-level testing.

  Changes the look and feel of EZwave, using light, gray, or dark themes for the GUI.

- -c

  Runs EZwave in batch mode. There is no graphical user interface or display. Refer to "Invoking EZwave in Batch Mode" on page 45.

- -hel | -help | -u | -usage

  Displays this help text.

- -do <filename>

  Indicates Tcl filename and location to be executed by the EZwave viewer.

- -args <args>

  Arguments that follow -ARGS are defined for the Tcl script (and are ignored by EZwave).

- -assign <variable=value>

  Assigns a Tcl variable to a value.

- -i <path>

  Specifies the input simulation results file or directory name (optional).

- -o <path>

  Specifies the output file name for the measurement results generated by the Tcl script.
- -s

  A comma-separated list of variable names for which to print out measurements.
- -tcl | -tclprompt

  Opens a prompt in the terminal, enabling single line Tcl commands to be entered directly.
- -bigmem

  Enables the possibility to use up to 75% of machine memory. Enabled by default.
- -nobigmem

  Limits the memory size to 4 GB. Equivalent to setting the environment variable AMS_JAVA_NO_BIG_MEMORY=1.

  Refer to "EZwave Environment Variables" on page 54 for further details.
- -maxwnd | -maximizefirstinnerwindow

  Maximizes the first inner window at startup.
- -now | -nowindow

  Prevents opening the initial empty window.
- -notree | -forcewfchoosernotvisible

  Hides the waveform list at startup.
- -singlewk | -forceworkspacenotallowmultiple

  Disables multi-workspace management.
- -skipexprerror

  Skips errors when executing Tcl commands or scripts.
- -log <path> | -logfile <path>

  Specifies the location for the session log file.
- -nologging

  Prevents session activity from being logged to a file.
- -nor | -norestore

  Prevents settings from a previous session from being restored.
- <file1>

  The name of the file to open. Refer to "Supported File Types" on page 45 for a list of the file types that can be opened in EZwave.

Related Topics

Invoking EZwave in Batch Mode

Visual Tour of EZwave

EZwave Display Preferences

Add Waveforms

Invoking EZwave in Batch Mode

To invoke EZwave in Batch Mode, type **ezwave** -c at the command line.

This topic describes the **ezwave** -c command, used to invoke EZwave in batch mode.

Syntax

ezwave -c …

Arguments

- -c

  Runs EZwave in batch mode. There is no graphical user interface or display.

Description

When running EZwave in batch-mode, the environment variable $DISPLAY is ignored. There is no graphical user interface or display.

When all connections are closed and script evaluations complete, EZwave batch exits.

Note

Running *.swd* file in batch mode is not supported. Tcl graphical commands are ignored in batch mode). There is no log, -log <path> is ignored.

Tip

You can check whether EZwave is in batch mode using the Tcl command "batch_mode" on page 1033. This can be useful inside scripts if you want them to work differently in batch mode, for example to make graphical operations conditional on mode.

Related Topics

Invoking EZwave

Passing Tcl Parameters From the Command Line

batch_mode

Supported File Types

This topic lists the file types that can be opened in EZwave.

The table lists of the file types that are supported in EZwave.

Table 1-1. Supported File Types

Icon	File Type	Extension	Notes
	MGC Database Files	**.wdb*	Joint Waveform Database (JWDB)

Table 1-1. Supported File Types (cont.)

Icon	File Type	Extension	Notes
	ICX Charter waveforms	*.cht*	
	Nutmeg Files	*.out*	Analog FastSPICE™ Nutmeg format files (includes nutmeg, nutbin, and nutbinf).
	CSV (Comma delimited)	*.csv*	
	HSPICE Graph Data Files	*.tr%* *.ac%* *.ft% *.mt%* *.sw%*	The EZwave viewer can only read files with extension *.tr%* that have been generated by HSPICE. Files with this extension that were generated by other tools may appear in the list of available files but will fail to load. Refer to "Loading .fsdb and .tr0 Files" on page 1272.
	OVD Files	*.ovd*	Open Verification Database format file. See also "Foreign Databases Options" on page 484 and "dataset ovd" on page 1072.
	Cou Files	*.cou*	
	Wlf Files	*.wlf*	Questa SIM Wave Log Format file. The EZwave viewer can only read WLF files that have been fully saved to disk where the simulation has completed and the simulator exited. The list of waveforms in the database is loaded first, then the default hierarchical name used within the waveform. Waveforms are created in the TRAN folder. Unsupported WLF types are ignored and a dialog will point to the log file.[1]
	VCD (Value Change Dump)	*.vcd*	Real signals can be defined as a 64-bit bus with the value provided as a floating-point number. Real 64-bit elements can still be defined using standard 64-bit binary descriptions. The state "U" is accepted as a correct value.
	PSF Files	runObjFile logFile *	PSF files do not have a predefined extension. EZwave will open files with the name *runObjFile* or *logFile* or unrecognized files that contain the string "PSFversion" in the first 100 bytes.

Table 1-1. Supported File Types (cont.)

Icon	File Type	Extension	Notes
	QWAVEDB Files	*.dB*	A qwavedb comes with a separate design.bin file, but this second file is ignored by EZwave. EZwave reads a QWAVEDB file after the simulation completes. In other words, marching waveform is not enabled.[2] Buses having 1000 bits or more are not loaded. You can change this behavior through the *settings.properties* file, for example: `qwavedbOption=-busSizeThreshold 25000`
	HSPICE/HyperLynx Output \ \file	*.lis*	
	ADiT Outputs	*.SD%* *.TR%* *.TB%* *.AC%* *.DC%*	The ADiT output reader is not available on the Windows platform. The “TR0” format generated by ADiT is different from the “tr0” format generated by HSPICE. The file extension is case sensitive to differentiate between formats. There are no marching waveforms with the ADiT formats
	Fsdb Files	*.fsdb* *.fsdb%*	Fast Signal Database File. Refer to Utility to Convert a Waveform Database in the *Eldo User’s Manual* for conversion details. The FSDB reader is not available on the Windows platform. Refer to “Loading .fsdb and .tr0 Files” on page 1272. FSDB file, compatible version is v5.8.
	Spice pwl	*.sti*	Only time-based waveforms can be saved as *.sti*.
	Raw Ascii Files	*.tab*	
	Saved Windows	*.swd*	
	DO-File	*.dofile* *.dof* *.do*	

Table 1-1. Supported File Types (cont.)

Icon	File Type	Extension	Notes
	TCL Files	*.rul *.tcl	
	Compressed Files	*.z *.Z *.gz	Compressed files require the gunzip tool and the compressed file should follow the convention, *<original_file_name>.gz* (or *.z* or *.Z*). The EZwave viewer will not modify the compressed file; it is uncompressed in the *MGC_TMPDIR* temporary directory and is removed from disk after you exit the EZwave viewer.
	Parameter Files	*.wdbparam	

1. **Supported WLF types:** wlfTypeArray, wlfTypeReg, wlfTypeNet, wlfTypeEnum, wlfTypeScalar, wlfTypeReal, wlfTypeBit, wlfTypeVlogEnum, wlfTypeTime, wlfTypeEvent, wlfTypeVlogReal, wlfTypeVlogWReal. **Unsupported WLF types:** bwlfTypeRecord, wlfTypeAccess, wlfTypeFile, wlfTypeMemElem, wlfTypeMemBits, wlfTypeClassRef, wlfTypeString, wlfTypePhysical, wlfTypeInteger.
2. **Supported QWAVEDB file types:** Verilog wire, Verilog reg, Verilog bit, Verilog real, System Verilog unpacked structure of type reg, System Verilog packed structure. From a VPI standpoint: vpiNet, vpiRealVar, vpiReg. **Unsupported QWAVEDB file types:** wreal, Verilog logic (256 states), Enumerations, EZwave ignores unsupported waveforms and does not load them.

Visual Tour of EZwave

This topic provides a tour of the EZwave viewer's main graphical user interface.

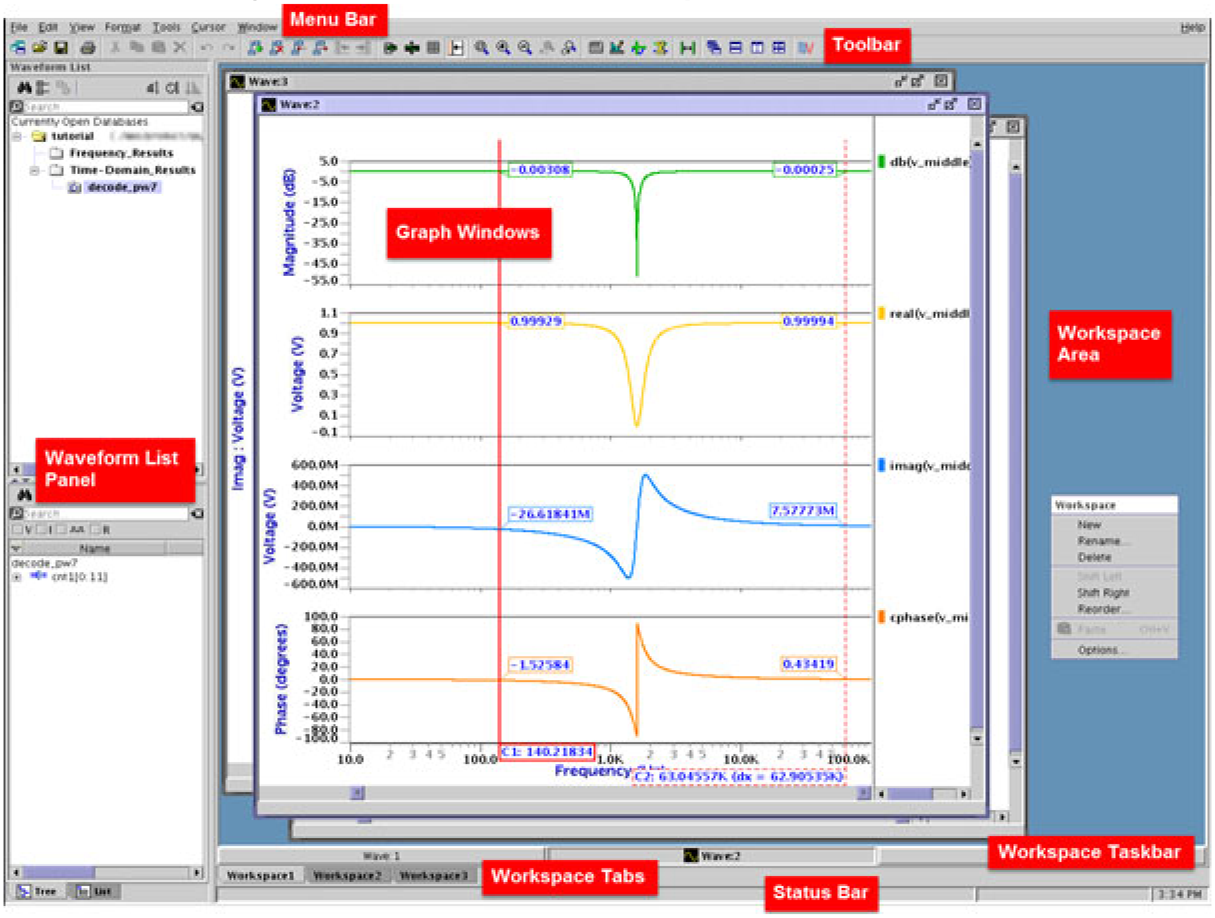
The Waveform List Panel displays all of the currently open databases as folders with folders or the individual waveforms listed underneath. Each waveform list element is associated with an icon indicating how the waveform will be displayed within the graph window.

Use the tabs on the Waveform List Panel to switch between a flat display (**List** tab) or a hierarchical format (**Tree** tab).

The Workspace is the area where the graph windows are displayed. You can move, resize, minimize, and restore each graph window that is displayed on each Workspace. From the **Window** menu, you can manage the windows by tiling or cascading them within the Workspace. You can create multiple Workspaces by right-clicking an open workspace and choosing New from the Workspace Popup Menu. All Workspaces can be accessed using the tabs at the bottom of the pane.

The Graph Window is used to plot and view waveforms. They are created when you drag waveform icons from the left Waveform List panel into the EZwave Workspace. You can have a single waveform in a graph window, multiple waveforms overlaid in a graph window, or multiple rows of waveforms in a single graph window.

Figure 1-1. EZwave Viewer Graphical User Interface



Digital waveforms displayed on a graph window are called trace rows. Digital waveforms display only logic states (on/off, hi/low, and so on). Analog waveforms are displayed in graph rows. Each point on an analog waveform represents a specifically graphed data point (for instance, showing voltage versus time).

Waveforms can be dragged up and down within the graph window or overlaid for comparison display. Drag your mouse pointer on an axis to zoom in on a waveform. To reverse the action, click the Undo Zoom icon on the main toolbar. You can arrange graph windows within a workspace into a cascade or a variety of tilings for easier viewing and printing.

The status bar at the bottom of the graph window displays the current X and Y coordinates of your mouse pointer (analog waveforms can be displayed with dual Y axes). If you click on a waveform, the status bar displays the distance and the slope between two successive click locations (shown as deltaX, deltaY, and slope).

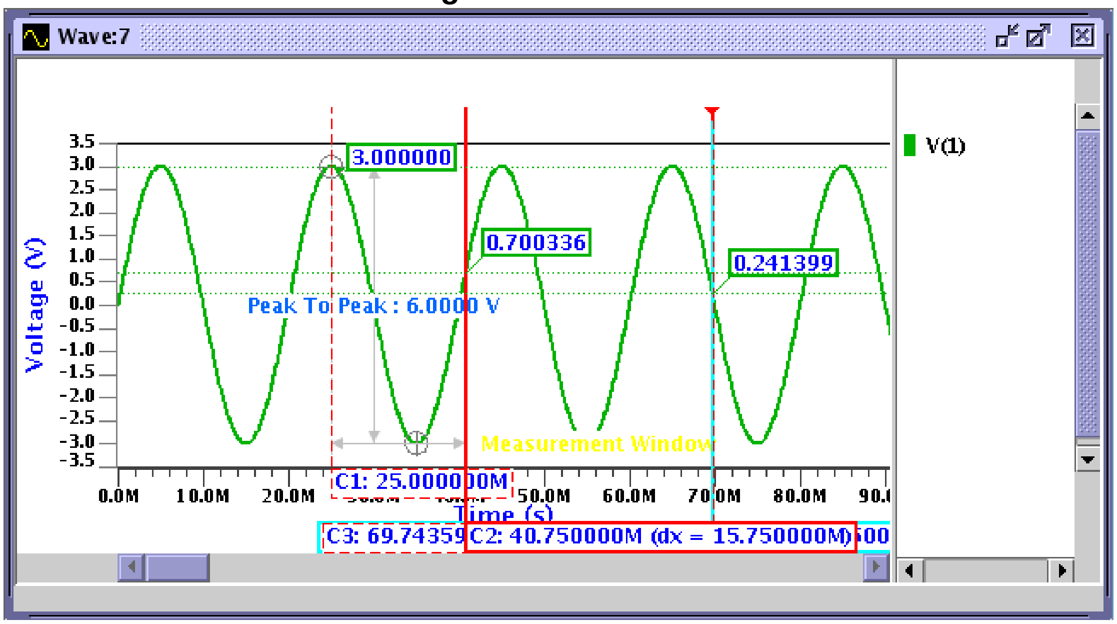
You can add cursors to plotted waveforms, to measure points or lengths of a waveform. A cursor is an on-screen indicator, drawn in the graph window waveform display area to identify locations on the X or Y axes, to create a point for measurement. Cursors are displayed as vertical or horizontal lines, each having a label and an X or Y value.

To add a cursor, right-click a waveform and select **Add Cursor** or **Add Horizontal Cursor**.

The first cursor created is known as the base (reference) cursor. You can add multiple cursors to show data points, as well as interpolated values between data points, and the delta between the base cursor. You can set a base cursor, then add additional cursors to measure values in between using the Measurement Tool (see "Measurement Tool" on page 279). The current cursor is highlighted, while others are set as dotted lines.

For more details, refer to "Working with Cursors" on page 165.

Figure 1-2. Cursors



Choose **Tools** > **Search** to invoke the Event Search Tool. This tool enables you to locate occurrences of simulation events interactively. An event is a definition of specific states (or values) for a single waveform or a collection of waveforms. To define an event, you need to select a set of waveforms and specify the states (or values) you want them to have. You can place markers (indicated by a red triangle) on cursors, enabling you to anchor locations to "jump" between. For details on how to use the Event Search Tool, refer to "Using the Event Search Tool" on page 194.

Choose **Tools** > **Measurement Tool** to invoke the Measurement Tool. This tool enables you to perform a range of analog and mixed-signal measurement operations on waveforms displayed in the graph window. You can annotate the measurement results in the graph window along with the measured waveforms. Measurement operations are divided into categories, such as general, time-domain, frequency-domain, and statistical.

In a graph window, insert two cursors on a waveform, then choose **Tools** > **Measurement Tool** from the EZwave menu. In the Measurement Tool, select the waveforms, then choose what type of measurement you need, and select the waveform area in which to apply the measurement. The results appear highlighted in the graph window. The results of some measurements produce other waveforms. The EZwave Measurement Tool enables you the option of creating and plotting the result waveform in the active graph window. For more details on how to use the Measurement Tool, refer to "Measurement Tool" on page 279.

Choose **Tools** > **Waveform Calculator** to invoke the Waveform Calculator. This tool is an integral part of post-processing and viewing the analog, digital, and mixed-signal simulation results. It can optimize the time it takes to analyze large amounts of simulation data, and supports a range of charting and analysis features. Refer to "Waveform Calculator" on page 312 and "Waveform Calculator GUI" on page 116 for more details.

Choose **Tools** > **Waveform Compare** > **Comparison Wizard** to invoke the Waveform Comparison Wizard. This tool enables you to compare waveforms from a reference simulation to a new result simulation when analyzing the analog, digital, and mixed-signal simulation results. The set of differences can be reported either graphically, or in report files. Refer to "Comparing Waveforms" on page 230 for more information.

Related Topics

EZwave GUI Overview

Chapter 2
Setting Up EZwave

This section describes how to set up EZwave. Installation is platform-dependent.

For information on AMS installation, see the AMS Installation Guide.

EZwave Environment Variables 54
- Environment Variables Set at Installation 54
- Environment Variables Set at Run Time 54
- Changing Default Environment Variables 56

EZwave Format Preferences. 58
- Configuring Waveform Names. 58
- Configuring the Color Scheme 59

EZwave Display Preferences. 61
- Configuring Display Preferences 61
- Configuring Colors, Fonts, and Styles 62
- Changing Waveform Colors 63

Configuring Keyboard Shortcuts 64

EZwave Environment Variables

This section describes the key environment variables that are set during the installation process and during run time, and how to change them.

Environment Variables Set at Installation 54
Environment Variables Set at Run Time 54
Changing Default Environment Variables 56

Environment Variables Set at Installation

This topic describes the key environment variables that are set to default path locations during the installation process. You should verify that these locations are correct.

- MGLS_LICENSE_FILE or LM_LICENSE_FILE

  Points to your Mentor Graphics license file or license server.

- MGC_AMS_HOME

  Points to the root installation tree.

On UNIX®, you can use the following command:

```
echo $environment_variable_name
```

On Windows®, see **Control Panel** > **System** > **Advanced tab** > **Environment Variables**.

Environment Variables Set at Run Time

This topic describes the key environment variables that are set at run time.

See also "Changing Default Environment Variables" on page 56.

The environment variables listed in the table are set during run time.

Table 2-1. Environment Variables

Environment Variable	Description
AMS_VIEWER_SETUP_HOME	Specifies the location of the *.ezwave* directory where the EZwave viewer keeps its setup files. The default location is *$HOME/.ezwave*.
AMS_VIEWER_WFC_NOTATION	Forces the Waveform Calculator to use the specified notation. Takes priority over notation settings specified in "Waveform Calculator General Options" on page 547. Specify value "spice" | "ieee".

Table 2-1. Environment Variables (cont.)

Environment Variable	Description
AMS_EZDO_ROOT	Specifies a root directory for relative paths when saving window content in TCL or SWD (*.swd*) format. The path entered in the File Name field on the Save Windows Dialog Box will be relative to AMS_EZDO_ROOT if the Use Relative Paths option is set to Relative to AMS_EZDO_ROOT in the Save Window Options.
AMS_EZWAVE_CAPACITY	Specify low | medium | high | versatile. Limits the available memory for EZwave. Low limit is 4 GB (suitable for Virtual Machines), medium limit is 32 GB, and high limit is 75% of available memory if the machine's memory is greater that 48 GB (otherwise high is equivalent to medium). Versatile is similar to high but also configures the Java™ Virtual Machine (JVM) minimum heap size -Xms to 1 GB and the maximum heap size -Xmx to 75% of the available memory. Overrides the default memory settings. For more information, refer to "Configuring Memory Capacity" on page 1267.
AMS_JAVA_MEMORY_HEAP	Specifies the minimum and maximum heap memory allocation for the JVM. The maximum amount of memory you can allocate depends on the system. You only need to modify this if you encounter out-of-memory errors. Refer to "Troubleshooting Memory Issues" on page 1267 and "Resolving Out-Of-Memory Problems" on page 1268.
AMS_JAVA_MEMORY_STACK	Removes the 10240 kB maximum stack memory allocation for the JVM. Generally used for troubleshooting. Refer to "Troubleshooting Memory Issues" on page 1267.
AMS_JAVA_NO_BIG_MEMORY	Limits the memory size to 4 GB. Equivalent to using the argument -nobigmem (see "Invoking EZwave" on page 41). Refer to "Troubleshooting Memory Issues" on page 1267.
MGC_TMPDIR	Specifies a directory to store temporary data files. If not set, the default directory for temporary files is */tmp*.
AMS_LSF_NOCHECK	When LSF is used, EZwave performs an automatic reload of a waveform database at the end of a simulation run, to release LSF licenses. This variable disables the automatic reload, in cases where unexpected waveform display issues are encountered.

Table 2-1. Environment Variables (cont.)

Environment Variable	Description
AMS_EDITOR	Specifies an external third-party text editor. This may be used instead of the built-in editor. See "Using and Editing User-Defined Functions in the Waveform Calculator" on page 319.
AMS_UDF_LOAD	Specifies that user-defined *.tcl* functions are loaded at startup, and the path to the user extension files directory. It also accepts multiple paths separated by ":" You can also use the Load User Extension Files At Startup option on the dialog box "Waveform Calculator General Options" on page 547. See also "Using and Editing User-Defined Functions in the Waveform Calculator" on page 319. If both AMS_UDF_LOAD and Load User Extension Files At Startup are used then functions from both locations are loaded.

Changing Default Environment Variables

The environment variables that are set during run time may be changed from their default values, if required, as described here.

The environment variables are listed in "Environment Variables Set at Run Time" on page 54.

Procedure

1. Type the following command once:

```
setenv AMS_USE_ENV 1
```

2. Type the setenv command that sets the environment variable:

```
setenv <environment_variable_name> <value>
```

For example, if you want to change the location of the *.ezwave* directory, type the following:

```
setenv AMS_USE_ENV 1
setenv AMS_VIEWER_SETUP_HOME $HOME/my_directory
```

This creates the directory, *$HOME/my_directory/.ezwave*.

Note

Do not use the **`echo $environment_variable_name`** command for these environment variables.

Related Topics

EZwave Display Preferences

Invoking EZwave

EZwave Format Preferences

You can customize global settings for the EZwave viewer environment to your preference using the many options.

Configuring Waveform Names 58
Configuring the Color Scheme 59

Configuring Waveform Names

You can define the global settings for how waveform names are displayed in graph windows.

Procedure

1. Choose **Format > Waveform Names Display**.

   The Waveform Names Display dialog box opens.

2. Select the required Waveform Hierarchy option to control the display of the waveform name hierarchy:

If you want to...	Do the following:
Display the full waveform name hierarchy	Select Full Hierarchy (default) For example: `:test:u1:x1:clk`
Display only the leaf name	Select No Hierarchy (Leaf Name Only) Use this option when waveforms can be identified without ambiguity by only the leaf name. For example: `clk`
Display a specified number of hierarchy levels	Select Display <N> Levels Use this option when waveforms can be identified without ambiguity by limiting the number of hierarchy levels displayed. For example, if N=2, two levels of hierarchy display: `u1:x1:clk`

3. Select a Justify Value option to specify Left (default) or Right justification of the waveform name display.

Tip
Use Right justification when information for identifying waveforms is at leaf level.

4. Select whether to display the Database Name as part of the waveform name. For example:

```
<adc12test_mixed_eldo_ms/TRAN>clk
```

Related Topics

Waveform Names Display Dialog Box

Changing Waveform Colors

Configuring the Color Scheme

Configuring Display Preferences

Configuring Colors, Fonts, and Styles

Waveform List Panel

Configuring the Color Scheme

You can define a global setting to specify the color scheme used when drawing the graph windows.

Note

The required Color Scheme must be selected on the Page Setup dialog box to affect printed or exported graph windows. See Printing Graph Windows and Exporting Graph Windows.

Procedure

Select the required color scheme:

If you want to...	Do the following:
Use a black background, with colors for the graph window objects.	Select **Format** > **Color Scheme** > **Black Background**
Use a white background, with colors for the graph window objects. This is the default color scheme.	Select **Format** > **Color Scheme** > **White Background**
Use a white background, with black for the graph window objects. Instead of waveforms being displayed in different colors, different line styles are used.	Select **Format** > **Color Scheme** > **Monochrome**
Use settings for items such as waveform line width, fonts, cursors and axes, which have been optimized for printing or exporting an image of the graph window(s).	Select **Format** > **Color Scheme** > **Documentation**

Caution

The Documentation color scheme is primarily intended for printing or exporting, and may significantly reduce the waveform display performance. For best performance, use a waveform line width of 1 pixel and a plain line style, for example, color schemes Black Background or White Background.

Tip

When using the Documentation color scheme for exported pages, ensure this option is selected prior to organizing and annotating your window. This will ensure that the page layout appears as expected in the export.

Related Topics

Configuring Colors, Fonts, and Styles

Configuring Display Preferences

Workspace

Waveform Colors Options

Changing Waveform Colors

EZwave Display Preferences

This section describes the many options for EZwave features and display that can be customized to your preference.

Configuring Display Preferences 61
Configuring Colors, Fonts, and Styles 62
Changing Waveform Colors 63

Configuring Display Preferences

You can customize graphical elements of the waveform display to your preference.

Procedure

1. Choose **Edit** > **Options**.

   The EZwave Display Preferences dialog box opens.

2. The list on the left is expanded by default, and contains the EZwave display preference categories as described in the following table.

Table 2-2. EZwave Display Preferences

General Display Preference Page	Description
Automatic Reload Options	Specifies whether previous results are kept when a new simulation is run.
Cursor Options	Specifies cursor options.
Data Format Options	Specifies the scaling and format of the axis tick labels.
Foreign Databases Options	Specifies load and display options when using foreign databases.
General Options	Specifies settings that affect the overall use of the application.
Grid Options	Specifies the appearance and behavior of the grid in graph windows.
Layout Options	Specifies the layout of the graph windows.
Look and Feel Options	Specifies the look and feel of EZwave and its dialog boxes.
Mouse Pointer Options	Specifies mouse pointer and mouse strokes properties.
Multiple Run Options	Specifies how compound waveforms from multiple runs are displayed.
Pick Points Options	Specifies settings related to pick points.

Table 2-2. EZwave Display Preferences (cont.)

General Display Preference Page	Description
RF Options	Specifies options for RF calculations in the Waveform Calculator.
Row Options	Specifies default row heights.
Save Data Options	Specifies global settings related to saving waveform data.
Save Window Options	Specifies global settings related to saving windows.
Text Annotation Options	Specifies the annotation display options.
Transformation Options	Specifies which data transformations are used with complex waveforms.
Waveform Options	Specifies how waveforms are displayed.
Waveform Compare Options	Specifies options for waveform comparison.
Waveform List Options	Specifies how the waveform list is displayed.
Workspace Options	Specifies workspace related operations.

Related Topics

EZwave Display Preferences Dialog Box

Configuring Colors, Fonts, and Styles

You can modify the fonts and colors of many of the elements in the EZwave viewer to your preference.

Procedure

1. Choose **Edit** > **Options**.

   The EZwave Display Preferences dialog box opens.

2. Expand the Fonts and Colors folder in the preferences list on the left. The following table lists the preferences available for modifying fonts and colors for graphical items.

Table 2-3. Fonts and Colors Preferences

Axis Title Options	Axis Values Options
Axis Values (Smith Chart) Options	Calculator Entry Options
Cursor/Marker Options	Eye Mask Options
Grid Options (Fonts and Colors)	Header Text Options
Histogram Options	Measurement Annotation Options

Table 2-3. Fonts and Colors Preferences (cont.)

Pick Points Options (Fonts and Colors)	Row Title Options
Text Annotation Options (Fonts and Colors)	Waveform Colors Options
Waveform Display Options	Waveform Name Options
Waveform Selection Options	Window Background Options
Zero-Level Line Options	

You can set different preferences for each of the options on the **Format** > **Color Scheme** menu. For Black and White Background color schemes only, you can set individual color settings. All other settings (such as fonts and line styles) are common for Black,.White, and Monochrome schemes. However, for the Documentation scheme, all settings are individual (except as described in the note).

Note

The Waveform Name options for Waveform Hierarchy, Justify Value, and Database Name options are common to all four color schemes.

Related Topics

EZwave Display Preferences Dialog Box

EZwave Viewer Interface

Configuring the Color Scheme

Changing Waveform Colors

You can change the size of the color palette and the colors within it.

Procedure

1. Choose **Edit** > **Options**.

   The EZwave Display Preferences dialog box opens.

2. Expand the Fonts and Colors folder in the list on the left side, and choose **Waveform Colors**.

   For analog waveforms, the default is to use the same color palette for both white and black backgrounds. To define a different palette for the white background, disable the option White Background Color Scheme shares the Black Background Waveform Colors Palette. For digital waves, color palette settings are always set separately for white and black backgrounds.

Note

Even if the color palettes are different, the color palette size is shared between White and Black color schemes.

When the Monochrome color scheme is active, the color palette cannot be set.

The Documentation color scheme uses its own color palette settings.

Related Topics

EZwave Display Preferences Dialog Box

Configuring Waveform Names

Configuring Colors, Fonts, and Styles

Configuring Display Preferences

Workspace

Waveform Colors Options

Waveform Properties Dialog Box (For Analog Waveforms)

Waveform Properties Dialog Box (Digital Waveforms)

Configuring Keyboard Shortcuts

You can modify the default EZwave keyboard shortcuts by assigning key strokes to actions. You can also select alternative key schemes, and import or export key schemes using the **.ezkey* file format.

Procedure

1. Choose **Edit** > **Shortcuts Manager**.

   The Shortcuts Manager dialog box opens.

2. Load an existing keyboard shortcuts scheme or save a new one, and do one of the following:

If you want to ...	Do the following:
Select an existing keyboard shortcuts scheme	Select a scheme from the Scheme dropdown list. Choose **Default** to use the standard EZwave keyboard shortcuts.
Import a **.ezkey* keyboard schema file	Click **Load**, navigate to the required file, and click **Open** on the Import Keyboard Shortcuts From File dialog box.
Apply changes and export an **.ezkey* keyboard schema file	Click **Apply and Save**, specify the required File Name and folder to save the file in, and click **Save** on the Export Keyboard Shortcuts To File dialog box.

3. View the available actions and associated keyboard shortcuts in the Shortcuts Manager, and filter the actions you are looking for in one of the following ways:

If you want to ...	Do the following:
Filter available actions by the name of the action	Type the name of an action into the Filter Actions text box. The table updates as you type, showing the matching actions. Click  to clear th e filter.
Filter available actions by the main menu category	Click **Load**, navigate to the required file, and click **Open** on the Import Keyboard Shortcuts From File dialog box.
Filter available actions by the category	Select the required menu from the By Category dropdown list, for example **Tools**.
Filter available actions by the context in which they are used	Select the required context from the By Context dropdown list, for example **Waveform Calculator**.
View actions that do not have an associated keystroke	Select the Show Empty Key Strokes checkbox.

4. Assign keyboard shortcuts to actions and modify existing assignments (called *binding*) using the Shortcuts Manager.

   To modify bindings, click the required row in the Actions table and then do any of the following:

If you want to ...	Do the following:
Remove the associated key stroke from the selected action	Click **Unbind Selection**.
Restore the last saved key stroke to the selected action	Click **Restore Selected To Last Saved**.
Modify the keyboard binding for the selected action	Specify the required shortcut in the Key Stroke field and ensure that the Description field is correct in the Edit pane. For example: 1. Click the Key Stroke field, hold down the Ctrl key, and press S. 2. Click the **Update** button.

   If you encounter any issues, for example, conflicts with the same shortcuts used for actions in the same context, a message appears in the Status Messages field. Review the items in the Issues table.

5. Click a row in the Issues table and specify different keystrokes for the affected actions in the Edit pane.
6. Click the **Update** button and review the Status Messages area to check that the issue has been resolved.
7. Click **OK** or **Apply** to update your changes.
8. Click **Close** if you do not want to save any changes.

Examples

You can also manually edit a **.ezkey* keyboard schema file or create a new file to define your own keyboard shortcut preferences for EZwave.

The format of the **.ezkey* file is as follows:

```
v1.0
<action> <context> <category> ["]<binding>["] ["]<description>["]
...
<action_N> <context_N> <category_N> ["]<binding_N>["]
["]<description_N>["]
```

You can also load a keyboard schema file using the Tcl command wave loadbindings:

```
wave loadbindings <file.ezkey>
```

Related Topics

Shortcuts Manager Dialog Box

wave loadbindings

Chapter 3
EZwave GUI Overview

This section provides an overview of the EZwave graphical user interface (GUI).

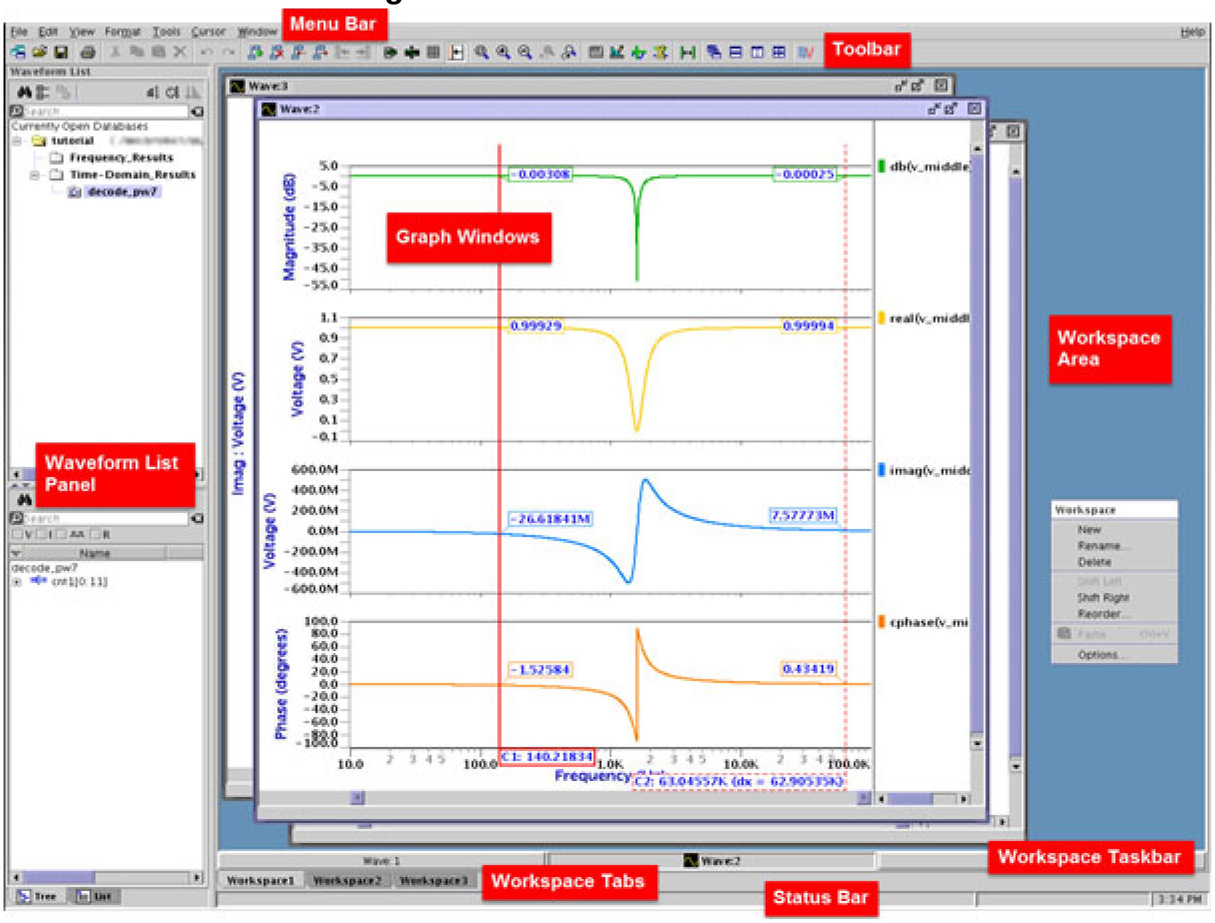
For a more in-depth description of the dialog boxes, see “Dialog Box and Field Reference” on page 419.

EZwave Viewer Interface **68**
Menu Bar 70
Toolbar 81
Waveform List Panel 83
Graph Window 90
Workspace 92
Workspace Taskbar. 93
Application Window Status Bar 93
Graph Window Popup Menus **95**
Graph Window Popup Menu 96
Row Popup Menu 96
Waveform Popup Menu 98
Axis Popup Menu 99
Cursor Popup Menus 102
Cursor Value Popup Menu 104
Waveform List Panel Popup Menus **106**
Waveform List Popup Menu. 106
Database Popup Menu 107
Folder Popup Menu 109
Hierarchy Popup Menu. 110
Waveform Name Popup Menu 111
Selected Waveforms Popup Menu 112
Workspace Popup Menu. **113**
Keyboard and Mouse. **115**
Keyboard Commands 115
Mouse Strokes. 115
Waveform Calculator GUI **116**

EZwave Viewer Interface

The EZwave Main Window provides access to all of the viewer features.

Figure 3-1. EZwave Main Window



The components of the Main Window are described in the following topics:

Menu Bar **70**
File Menu 70
Edit Menu 71
View Menu 74
Format Menu. 75
Tools Menu 76
Cursor Menu 78
Window Menu 79
Help Menu 80
Toolbar **81**
Waveform List Panel **83**
Graph Window **90**
Creating a New Graph Window 90

Graph Window Title Bar 91
Rows 91
Waveform Names Display 91
Graph Window Status Bar 92
Workspace 92
Workspace Taskbar 93
Application Window Status Bar 93

Menu Bar

You can use the menu bar to access all of the EZwave menus.

This section describes the options available under each of the following menus in EZwave:

File Menu 70
Edit Menu 71
View Menu 74
Format Menu 75
Tools Menu 76
Cursor Menu 78
Window Menu 79
Help Menu 80

File Menu

You can use the File menu to access to Load, Save and Print operations.

It contains the following items:

Table 3-1. File Menu Items

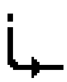
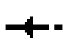
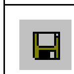
Icon	Mouse Stroke	Key Shortcut	Item	Description
		Ctrl+N	New	Creates a new graph window.
		Ctrl+O	Open	Opens any supported file type. For details refer to "Supported File Types" on page 45.
		Ctrl+W	Close	Closes the active graph window.
		Ctrl+R	Reload	Updates a waveform with new simulator information.
		Ctrl+S	Save	Saves the active graph window as a Saved Graph Window (*.swd*) file. Refer to "Save Windows Dialog Box" on page 586.

Table 3-1. File Menu Items (cont.)

			Export	Saves the active graph window in JPEG (*.jpg*) or PNG (*.png*) format. Refer to “Exporting Graph Windows as an Image” on page 408 for options available when exporting a window to a graphics file.
			Page Setup	Specifies the text displayed at the top of the graph window when printing and the color scheme.
		Ctrl+P	Print	Prints the visible data in the active graph window. If you are having difficulty using */usr/sbin/lpc* to print from a Linux system, refer to “Linux Printing Issues” on page 1264.
			Database List	Opens a recently used waveform database. Provides a listing of all recently used waveform databases. Select a waveform database from the list to open it.
			Exit	Closes all graph windows, databases, and the application. You are prompted to confirm the exit, and save databases with unsaved changes before the application closes. You may control the display of the exit confirmation box through General Options.

Edit Menu

EZwave has an Edit menu for access to Edit, Group, Find, Shortcuts Manager and EZwave Options features.

The Edit menu contains the following items:

Table 3-2. Edit Menu Items

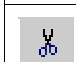
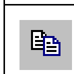
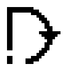
Icon	**Mouse Stroke**	**Key Shortcut**	**Item**	**Description**
			Undo/Redo	Undoes the previous graphical operation, such as plotting a waveform or splitting a row. Removes the effect of the previous undo operation, performing the operation again.
		Ctrl+X	Cut	Removes the currently selected objects from the active graph window and stores them on the *clipboard*.
		Ctrl+C	Copy	Makes a duplicate of the currently selected objects and stores them to the *clipboard*.

Table 3-2. Edit Menu Items (cont.)

			Copy Window to Clipboard	Copies the visible data as a bitmap image from the active window to the Windows clipboard. Only available on the Windows platform.
		Ctrl+V	Paste	Inserts the contents from the *clipboard* into the active graph window.
	Groups	Ctrl+G	Groups	Creates a waveform group, such that multiple waveforms can be manipulated together. Only available when 2 or more waveforms are selected.
			Delete	Removes the currently selected objects from the active graph window.
		Ctrl+A	Select All	Selects all objects in the active graph window for further action. Click a blank area of the active graph window to clear the selection
			Unselect All	Unselects all currently selected objects in the active graph window.
			Find	Opens the Find Tool Dialog Box that displays a filtered list of waveforms to be plotted in the active window.
			Copy Window to Clipboard	Copies the visible data in the active graph window to the *Clipboard (Windows)* as a bitmap image. (Available only on Windows).
			Shortcuts Manager	Opens the Shortcuts Manager Dialog Box that enables you to modify the default EZwave keyboard shortcuts, assigning key strokes to actions. Refer to "Configuring Keyboard Shortcuts" on page 64.
			Options	Opens the EZwave Options dialog box. Refer to "Configuring Display Preferences" on page 61.

Undo and Redo

You can use the **Edit** > **Undo** menu item to undo previous graphical operations within EZwave.

The Undo command is able to undo operations such as adding, moving, and deleting waveforms and cursors, which occur graphically within EZwave. It is not able to operate on simulation-related actions, such as creating waveforms in the database or plotting waveforms interactively from the simulator graphical interface.

The undo buffer can hold multiple commands, so that you can undo the last several actions. However, some actions—particularly those that would cause undo to become unsafe—clear the undo buffer. These commands include the following:

- Opening a *.swd*, *.do*, or *.tcl* file
- Commands sent by a simulator or client that could corrupt the undo buffer
- Updating compound structure (usually sent by a simulator)
- Calculator operations that create or manipulate hidden objects
- Deleting an object from a database
- Working With a Bus
- Converting a digital wave to analog or an analog wave to digital
- Working with Eye Diagrams
- Adding a measurement
- Performing a measurement in the calculator with a command that plots a result waveform
- Deleting a workspace
- Renaming a database or a waveform within a database
- Issuing an Update Waveforms command from the Database Popup Menu

Additionally, the following commands cannot currently be undone, but do not clear the undo buffer:

- Adding or renaming a workspace
- Showing or hiding all annotations
- Moving an annotation or annotation text
- Turning on or off cursor locking
- Turning on or off snapping to data points
- Executing Hide Value or Y-Level Line from the Cursor Value Popup Menu
- Changing the visibility of the cursor values
- Resizing or scrolling a graphical window

The **Edit** > **Redo** command repeats the commands that were undone with the **Edit** > **Undo** command, starting with the most recent one. This functionality can be disabled through an option in the General Options.

View Menu

EZwave has a View menu for access to Zoom, Gridlines, Waveform and Toolbar features.

The View menu contains the following items:

Table 3-3. View Menu Items

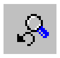
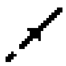
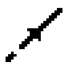
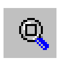
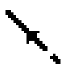
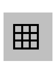
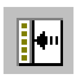
Icon	Mouse Stroke	Key Shortcut	Item	Description
		U	Zoom Undo	Undoes the previous zoom operation in the active graph window
		R	Zoom Redo	Redoes the last action undone by Zoom Undo
		I	Zoom In	Zooms in to view more detail in the x direction of the active graph window.
		O	Zoom Out	Zooms the display out to view less detail in the x axis of the active graph window.
		A	View All	Reset magnification to view all of the data in all of the rows within the active graph window. This operation does not change the height of any of the rows. This behavior may be modified using the View All Activates Auto-Range checkbox on the EZwave Display Preferences General Options dialog box.
			Min Row Height Plot Mode	Activates Minimum Row Height Plot Mode. When on, all Waveforms are plotted using the Minimum Size specified on the EZwave Display Preferences Row Options dialog box. Useful when displaying many waveforms.
		Ctrl+H	Fit Row Heights	Changes the row heights in the active graph window to see as many rows as possible.
			Grid Lines	Toggles the display of grid lines within the active graph window and sets the mode for all future rows. Only rows showing analog waveforms support grid lines. Grid lines on an individual axis can be controlled by using the Axis Popup Menu. The color of grid lines can be change by using Measurement Tool Dialog Box.

Table 3-3. View Menu Items (cont.)

		Ctrl+T	Tandem Mode	Activates Tandem Mode. This enables you to plot waveforms with the same name from different databases. Refer to Plotting Waveforms With the Same Name Using Tandem Mode.
			Zero-Level Lines	Toggles the visibility of lines showing X=0 and Y=0 in the graph window. Only rows showing analog waveforms support zero-level lines. Zero-level lines can be controlled using the Axis Popup Menu. If the zero-level line for an x axis is selected, it turns on the zero-level lines for all rows. For a y axis it activates on a per-row basis.
			Waveform List	Toggles the display of the Waveform List panel.
			Toolbar	Toggles the display of the toolbar.
			Status Bar	Toggles the display of the status bar within the application window.
		Ctrl+L	Refresh	Refreshes the display of the active graph window.

Format Menu

EZwave has a Format menu for access to Waveform Names and Color Scheme features.

The Format menu contains the following items:

Table 3-4. Format Menu Items

Item	Description
Waveform Names Display	Provides format options for waveform names within the graph windows. This is a global setting. **Show Full Hierarchy** - Controls whether the waveform name contains the full design path name, or just the waveform leaf name. **Show Database Names** - Controls whether the displayed waveform name contains the database name. For example, if there is a waveform named “clock” in a database named “results” then checking this option would include the text “results” when the “clock” waveform name was displayed in the graph windows.

Table 3-4. Format Menu Items (cont.)

Item	Description
Color Scheme	Provides control over the color scheme used within the graph windows. This is a global setting. **Black Background** - Uses a black background, with colors for the graph window objects. **White Background** - Uses a white background, with colors for the graph window objects. **Monochrome** - Uses a white background, and uses black for the graph window objects. Instead of using multiple colors for displaying waveforms, this uses different line styles. **Documentation** - Uses a white background and other settings for items such as waveform line width, fonts, cursors and axes, that have been optimized for printing or exporting an image of the graph window(s).

Tools Menu

EZwave has a Tools menu for access to all of the main analysis tools.

The Tools menu contains the following items:

Table 3-5. Tools Menu Items

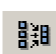
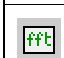
Icon	Item	Description
	Waveform Compare	Contains submenus to set up and perform waveform comparison. Refer to "Comparing Waveforms" on page 230.
	Create Bus	Creates user-defined buses. Refer to "Working With a Bus" on page 273.
	Eye Diagram	Creates an eye diagram based on a period of a waveform. Refer to "Working with Eye Diagrams" on page 197.
	Extract	Extracts outputs from a database. Refer to "Extracting Outputs from a Database" on page 382.
	FFT	Opens the Fast Fourier Transform Tool Dialog Box. Refer to "Using the Fast Fourier Transform Tool" on page 384.

Table 3-5. Tools Menu Items (cont.)

	Jitter	Analyzes clock jitter between any digital and analog target signal with respect to a reference period or frequency, or calculates phase noise jitter. Refer to "Jitter Tool" on page 368.
	Parameter Analyzer	Generates "pivot" waveforms. Changes the running variable of one or more compound waveform(s) to one of the waveform's parameter values, or to the y values of a different compound waveform, and then plots the pivot waveform. Refer to "Analyzing Waveform Parameters to Generate Pivot Waveforms" on page 261.
	Current Analysis	Analyzes power consumption of a circuit and its components over time. Refer to "Analyzing Current Consumption" on page 262.
	Power Analysis	Analyzes power consumption of a circuit and its components over time. Refer to "Analyzing Power Consumption" on page 268.
	Search	Defines and searches for events. Refer to "Using the Event Search Tool" on page 194.
	Spectrum Measurement	Opens the Spectrum Measurement Tool Dialog Box. Refer to "Using the Spectrum Measurement Tool" on page 388.
	DNA Advisor	Opens the DNA Advisor Dialog Box, used to analyze transient or PSS waveforms, and recommend the analysis parameters required for an accurate simulation. Refer to "Using the DNA Advisor Tool" on page 399.
	Waveform Calculator	Post-processes mixed-signal simulation results. Refer to "Waveform Calculator" on page 312.
	Measurement Tool	Measures and verifies analog and mixed-signal simulation results. Refer to "Measurement Tool" on page 279.

Table 3-5. Tools Menu Items (cont.)

	Pick Points	Pick points are temporary markers that enable you to perform quick measurements on selected waveforms. Refer to "Using Pick Points to Take Measurements" on page 184.

Cursor Menu

EZwave has a Cursor menu for access to the various cursor operations.

The Cursor menu contains the following items:

Table 3-6. Cursor Menu Items

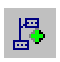
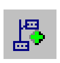
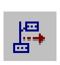
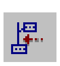
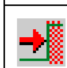
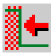
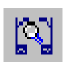
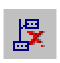
Icon	Keyboard Shortcut	Item	Description
	F5	Add	Creates a new data cursor in the center of the active graph window.
		Add Horizontal Cursor	Creates a horizontal cursor on the selected y axis. By default, the first crossing point with the waveform and its measurement is shown
		Add with Y-Level	Creates a new data cursor with a Y-Level line in the center of the active graph window.
		Add Cursor Relatively To > <cursor>	Opens a dialog box where one or more X values can be specified for creating new cursors relative to the selected cursor.
	Tab	Next Data Point	Moves the cursor to the next data point in the active graph window. First select the waveform whose data should be used.
	Shift +Tab	Previous Data Point	Moves the cursor to the previous data point. First select the waveform whose data should be used.
		Next Error	Moves the cursor to the next error in the waveform, based on time. When applied to a waveform comparison result, the cursor is moved to the next difference between the compared waveforms.

Table 3-6. Cursor Menu Items (cont.)

Icon	Keyboard Shortcut	Item	Description
		Previous Error	Moves the cursor to the previous error in the active graph window. When applied to a waveform comparison result, the cursor is moved to the next difference between the compared waveforms.
		Lock Together When Dragging	Connects all cursors so that when one is moved, all other cursors are moved the same amount. When cursors are locked, a checkmark displays next to the item.
		Share Config of Active Cursor	Configure several cursors to share the same visibility settings for cursor values.
		Zoom Between Cursors	Zooms between the two cursors that are furthest apart.
		Delete Active	Deletes the currently active cursor from the active graph window. The cursor closest to the one being deleted becomes active.
		Delete All	Deletes all annotations.
		Cursor List	Provides a listing of all cursors currently in the active graph window. Select the cursor from the list to move from cursor to cursor.

Window Menu

EZwave has a Window menu for access to the various window features.

The Window menu contains the following items:

Table 3-7. Window Menu Items

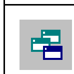
Icon	Item	Description
	New	Creates a new graph window.
	Title	Specify a title for the active graph window.
	Cascade	Arranges the graph windows in an overlapped (cascade) fashion.

Table 3-7. Window Menu Items (cont.)

	Tile Horizontally	Arranges the graph windows in a tiled (left/right) fashion.
	Tile Vertically	Arranges the graph windows in a tiled (top/bottom) fashion.
	Tile in a Grid	Arranges the graph windows in a tiled (left/right and top/bottom) fashion.
	Restore All	Restore all currently minimized graph windows.
	Minimize All	Minimizes all graph windows on the active workspace.
	Close All	Closes all graph windows in the active workspace.
	Window List	At the bottom of the menu is a list of all graph windows on all workspaces. Selecting one of these will switch to the workspace containing the window, and restore the window if it was minimized.
	More Windows	Provides a secondary list for selection.

Help Menu

EZwave has a Help menu for access to the on-line help system and tutorials.

The Help menu contains the following items:

Table 3-8. Help Menu Items

Item	Description
Quick Start	Launches the Setting Up EZwave section of this manual.
Tutorial	Launches the Getting Started with EZwave tutorial manual. The Tutorial Data menu item on the sub-menu provides a small database for use.
Contents and Index	Launches the online help system.
Quick Reference	Provides a short overview of EZwave's key functionality and lists of mouse and keyboard operations and shortcuts.
Release Notes	Launches the AMS Release Notes.
AMS Documentation- InfoHub	Launches the InfoHub, which contains all of the AMS documentation.

Table 3-8. Help Menu Items (cont.)

Item	Description
About EZwave	Provides information about the application for Contacting the Customer Support Center.

Toolbar

The toolbar contains buttons (displayed with icons) for accessing frequently used operations.

To access: Choose **View** > **Toolbar**.

Place the mouse pointer over the toolbar icon to display the *tooltip* for that item.

Table 3-9. Toolbar Icons

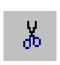
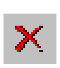
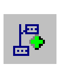
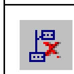
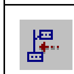
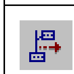
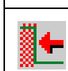
Item	Description
	Creates a new graph window.
	Opens any supported file.
	Saves the active graph window.
	Prints the active graph window.
	Cuts the selection from the active graph window and store it in the viewer clipboard.
	Copies the selection from the active graph window to the viewer clipboard.
	Pastes the contents of the clipboard in a graph window.
	Deletes the selection from the active graph window.
	Undoes/Redoes the previous graphical operation.
	Adds a new data cursor
	Deletes the active data cursor.
	Moves the data cursor to the previous data point or transition.
	Moves the data cursor to the next data point or transition.
	Moves the data cursor to the previous error or difference point.

Table 3-9. Toolbar Icons (cont.)

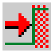
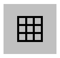
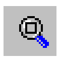
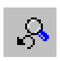
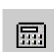
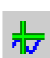
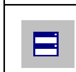
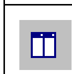
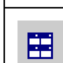
	Moves the data cursor to the next error or difference point.
	Toggles Minimum Row Height Plot Mode on or off. When on, all Waveforms are plotted using the Minimum Size specified on the EZwave Display Preferences Row Options dialog box. Useful when displaying many waveforms.
	Fits row heights in the active graph window.
	Toggles the active graph window grid lines on or off.
	Toggles the Waveform List panel on or off.
	Views all of the active graph window so that all the data is visible.
	Zooms in to increase magnification in the X direction.
	Zooms out to decrease magnification in the X direction.
	Undoes the previous zoom operation. (Multiple operations supported)
	Redoes last undone zoom.
	Invokes the Waveform Calculator.
	Invokes the Measurement Tool.
	Invokes the Pick Points Tool.
	Invokes Tandem Mode, enabling automatic plotting of all waveforms with the same name from different databases.
	Adds a delta marker to a waveform or between two waveforms.
	Updates the graph window with new data from a running simulation. This icon is only available when the application is started from the Questa ADMS tool, or when the EZwave viewer is connected to an Eldo simulation.
	Arranges multiple graph windows in an overlapped (cascade) fashion.
	Arranges multiple graph windows in a stacked (horizontal) fashion.
	Arranges multiple graph windows in a side by side (vertical) fashion.
	Arranges multiple graph windows in a tiled (horizontal and vertical) fashion.

Table 3-9. Toolbar Icons (cont.)

	Toggles the decoration (title, buttons and scrollbars) of wave windows on or off.
	Toggles boundary elements visibility on or off. Only used with .wtcl databases.

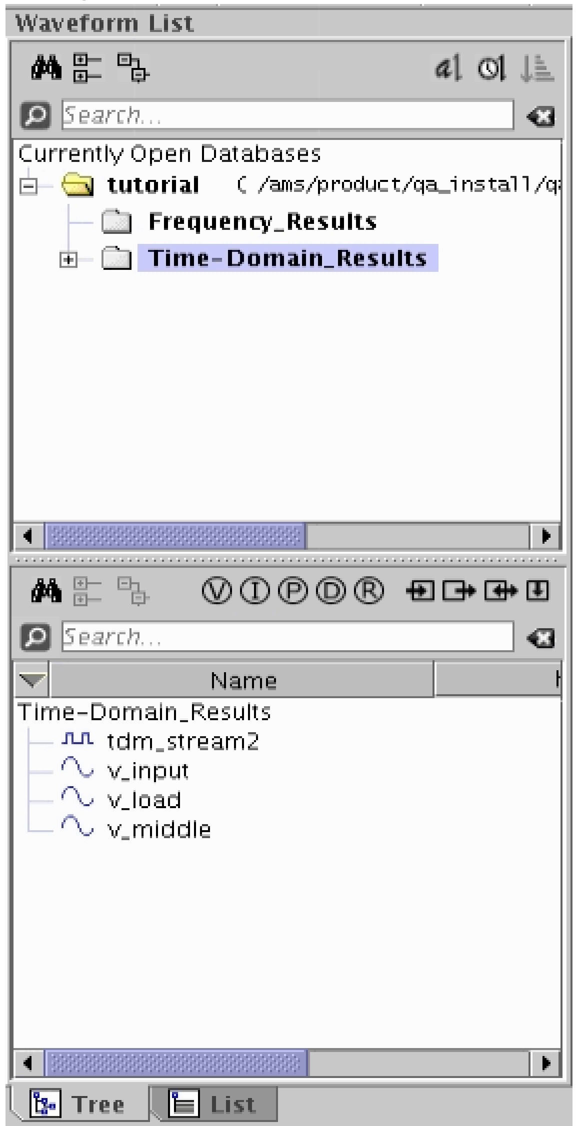
Waveform List Panel

The Waveform List panel shows all of the waveform databases that are currently open in EZwave.

Overview

The top panel lists the databases in folders, and the bottom panel lists the individual waveforms contained within them. Use the tabs to switch between a flat display (**List** tab) or a hierarchical format (**Tree** tab).

Figure 3-2. Waveform List Panel



The **Tree** tab (by default) splits the panel into a Structure view (top) and a waveform list (bottom). To change this, go to the Waveform List Options and clear the Separate Tree View Into Structure and Waveform List Views option

Note

For databases loaded from *.fsdb* files, only loaded waveforms are displayed in the list panel. Loaded waveforms are waveforms that have been displayed once, or waveforms from a hierarchy selected once in the Tree View of the panel. Refer to "Loading .fsdb and .tr0 Files" on page 1272.

Waveform List Popup Menus

Right-click an item in the waveform list panel to access various functions via the Waveform List Panel Popup Menus.

Each item in the waveform list is associated with an icon indicating how the waveform will be displayed when plotted in the graph window.

Table 3-10. Waveform List Element Icons

Icon	Waveform Database Content	Description
	Saved Waveform Database	Indicates a waveform database that exists on disk, and does not contain any unsaved changes. The full path to the database displays to the right of the name.
	Unsaved database	Indicates a waveform database that contains unsaved changes. An attempt to close the database will result in a confirmation box being shown to prevent accidental loss of the changes.
	Folder	Indicates a folder within a waveform database.
	Design Hierarchy Folder	Indicates a folder within a waveform database that contains waveforms in a design hierarchy.
	Analog or complex waveform	Indicates a waveform containing analog data.
	Digital waveform	Indicates a waveform containing digital state data.
	Step waveform	Indicates a waveform whose Y data values are displayed as discrete levels.
	Enumerated waveform	Indicates a waveform containing Y data values from an enumerated list.
	Compound waveform	Indicates a compound waveform containing the results of several simulations for the same node.
	Bus	Indicates that the waveform is a bus, which means it contains one or more bus bit waveforms.
	Measure	Indicates a measure. The Name and Value of the scalar are displayed by default.
	Variable	Indicates a variable. The Name and Value of the variable are displayed by default.

Sorting Databases in the Waveform List

By default, EZwave displays waveform databases in the order in which they were added.You can sort the databases by name or date, using the following buttons:

Figure 3-3. Database Sort Buttons

The buttons are as follows:

- Sort by Name

  Select to sort databases by name.

- Sort by Date

  Select to sort databases by date modified.

- Toggle Sort Direction

  Toggle to change the sort order.

Searching for Databases and Waveforms in the Waveform List

You can perform a quick search across the open waveform databases, by entering a search term into the search text boxes above each panel in the Waveform List panel.

All matching waveform names associated with that panel are highlighted. The search includes an implied * wildcard character at the start and end of the term entered.

The search term can include * and ? wildcard characters. If the search string does not contain either, the search returns waveform names that match the search term at any position in the waveform name. If you include a * or ? wildcard character, the search returns waveform names that match the search term from the first character of the waveform name.

The search highlights waveforms that match the string (or partial string) as you type it into the search field. If automatic searching is disabled, you must press Enter to perform the search. Automatic searching can be disabled using the Waveform List Options.

After you press Enter to perform a search, the search is added to the search history list. Click the  icon to access the search history. The history maintains up to ten entries. Select the **Clear History** option to reset the history.

Note

If more than 100 search results are found, EZwave displays the number of results and asks if you want to display the search results.

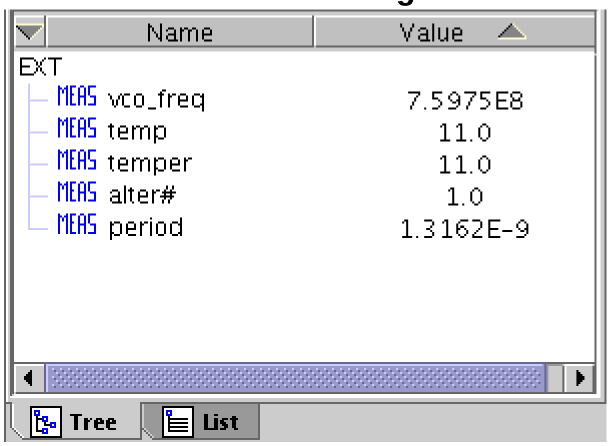
Tip

For more search options, use the Find Tool Dialog Box by clicking .

Waveform List Columns

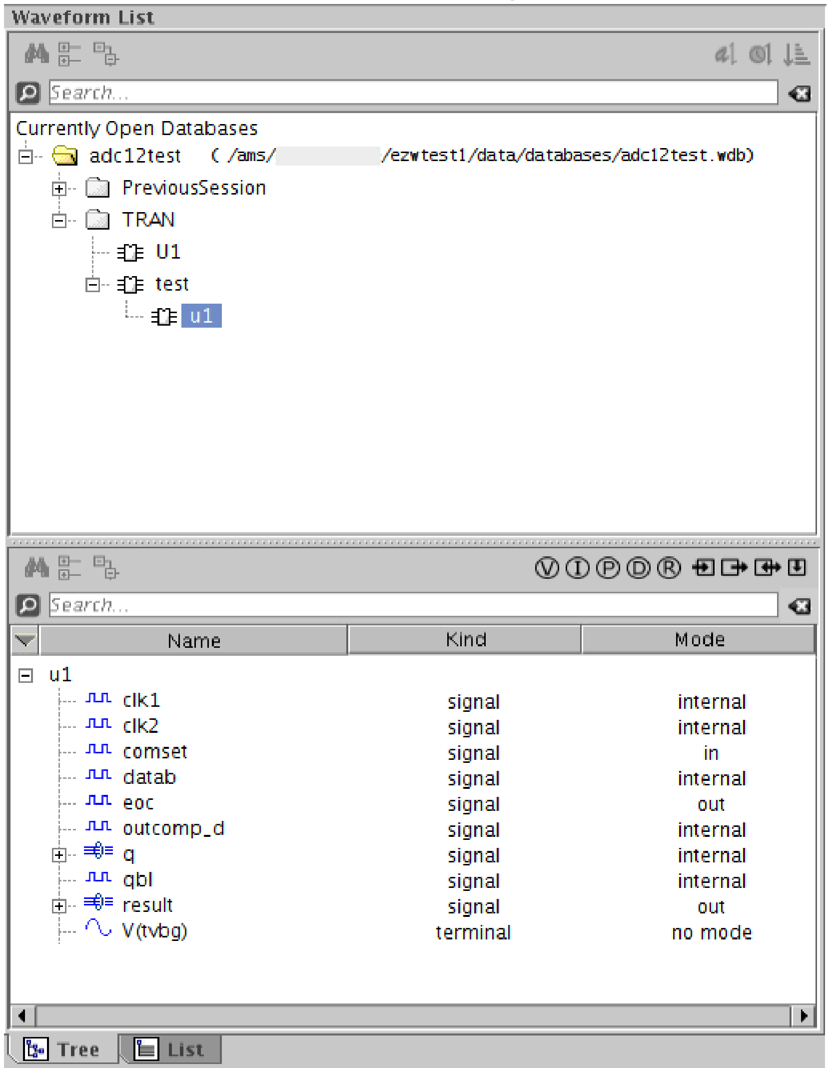
The Waveform List displays the Waveform Name. You can add or remove the Kind, Mode, X, Y, and Value columns; right-click the column title bar and select the appropriate option. Right-clicking the column title bar also displays mode-filtering options. To sort the list, left-click the column title bar that you want to sort by, as shown in the following figure:

Figure 3-4. Waveform List Panel Showing Measures Sorted by Value



In the following figure, the waveform is sorted by Mode. In this example, the Mode corresponds to the pin direction.

Figure 3-5. Waveform List Panel Showing Waveform Sorted by Mode



Filtering Waveforms in the Waveform List Panel

You can filter the waveforms that are displayed in the Waveform List by their type, by clicking on the Waveform List filter options:

Table 3-11. Waveform List Filter Options

Icon	Description
Ⓥ	Analog Voltage
Ⓘ	Analog Current
Ⓟ	Power Signals
Ⓓ	Digital Signals
Ⓡ	Real and WReal Signals

Table 3-11. Waveform List Filter Options (cont.)

Icon	Description
Pin Direction Filters *	
	Input Signals
	Output Signals
	InOut Signals
	Internal Signals

Note

* To show the Pin Direction Filters in the Waveform List panel, choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box and choose Waveform List. Then check the Show Pin Direction Filters Panel box.

Graph Window

The Graph Window is used to plot and view waveforms. They are created when you drag waveform icons from the left Waveform List panel into the EZwave Workspace. You can have a single waveform in a graph window, multiple waveforms overlaid in a graph window, or multiple rows of waveforms in a single graph window.

Waveforms can be dragged up and down within the graph window or overlaid for comparison display.

Creating a New Graph Window **90**
Graph Window Title Bar **91**
Rows **91**
Waveform Names Display **91**
Graph Window Status Bar **92**

Creating a New Graph Window

This topic describes how you can create new graph windows.

Procedure

1. Choose **File** > **New**.

   A new graph window is created.

2. You may also select waveforms from the Waveform List panel and use the Plot (Overlaid) or Plot (Stacked) menu item from the "Database Popup Menu" on page 107.

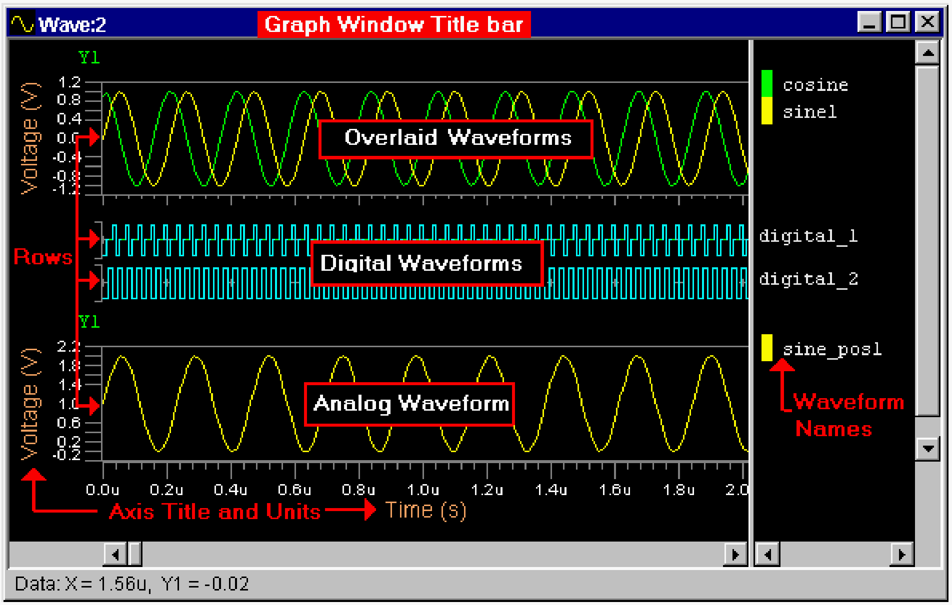
   You can create a large number of graph windows, but only one graph window may be active at any time. The active graph window is always on the top of the workspace area.

Results

As each new graph window is created, a graph window button is added to the workspace taskbar at the bottom of the application window. You can use these buttons to bring the graph window to the top of your workspace.

The grid displayed in the graph window is user-configurable; see the "Grid Options" on page 488 topic.

Figure 3-6. EZwave Graph Window



Graph Window Title Bar

Each graph window has a unique name of the form **Wave:#**.

You can add a custom title string that will appear in the title bar by using the **Window** > **Title** menu item or the Graph Window Popup Menu.

Rows

A graph window can have multiple graph rows. You can quickly resize the height of an analog graph row by dragging the x axis within the row.

Additional viewing options are available from the Row Popup Menu. Right-click within the row to display this menu.

Waveform Names Display

Waveform Names are shown next to the plot of the waveform. The location of the names, relative to the plotted waveforms, and information displayed can be changed.

Procedure

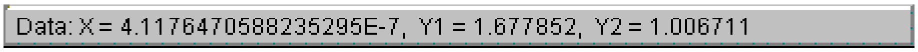
1. Use the **Edit** > **Options** menu item to access the Layout Options.
2. Use the **Format** > **Waveform Names Display** menu item to control how much information is shown for each waveform name.

3. Right-click the waveform name or the plotted waveform to activate the Waveform Name Popup Menu.

   You can select multiple waveforms by selecting each one individually with Ctrl+click, or selecting a range with Shift+click.

Graph Window Status Bar

The status bar at the bottom of the graph window displays the current X and Y coordinates of the mouse pointer.



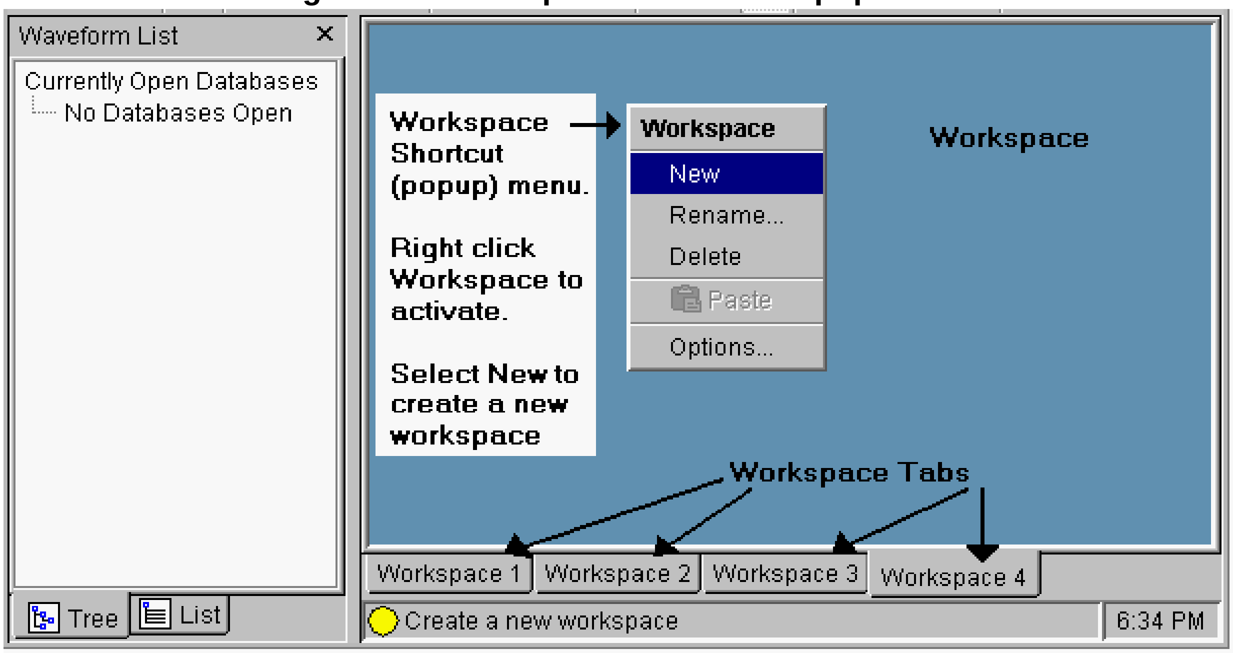
When there are multiple y axes within a single row, there will be one Y data value for each of the axes. If you click a waveform, it also shows the distance between the point where you clicked and the point where you had previously clicked before that as deltaX and deltaY.

Workspace

The workspace is the area where the graph windows are displayed. It is located directly below the toolbar on the application window.

If the Waveform List Panel is displayed, the workspace displays to the right of the waveform list.

Figure 3-7. Workspace Tabs and Popup Menu

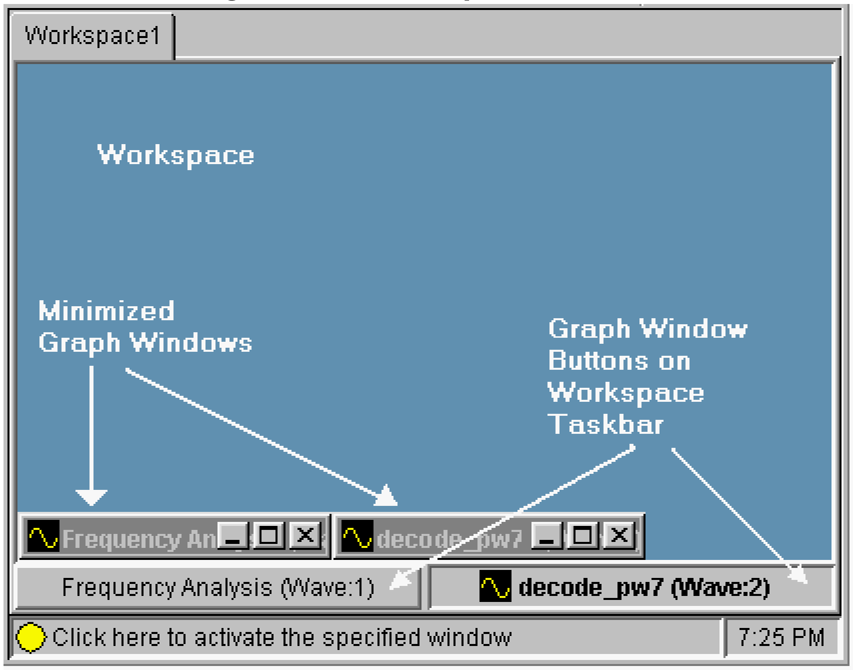


The application supports multiple workspaces for flexibility in organizing graph windows. Multiple workspaces are accessed through workspace tabs. Click the **Workspace** tab to bring that workspace to the top of your application window.

Workspace Taskbar

Above the status bar, a taskbar is available to easily select graph windows. As each new graph window is added to the workspace, a graph window button is added to this taskbar.

Figure 3-8. Workspace Taskbar



When graph windows are minimized, they appear as icons on the workspace. Click the graph window button on the workspace taskbar to make that window the active graph window. When you click the taskbar, if the window is minimized, it is automatically restored and brought to the front of the workspace.

The taskbar can be turned off through the Workspace Options.

Application Window Status Bar

The status bar is located at the bottom of the EZwave Viewer Interface.

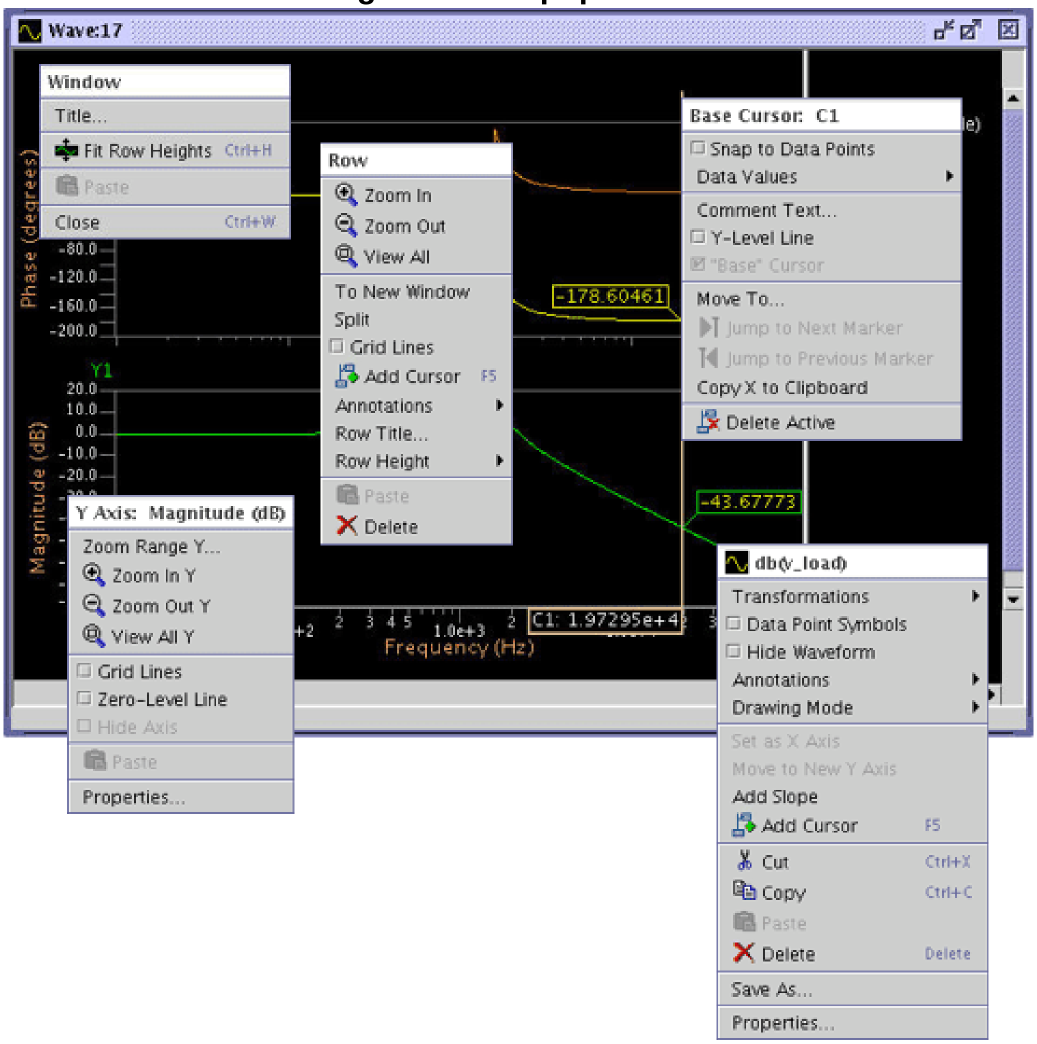
The left area of the status bar describes actions of menu items as you navigate through menus. This area also shows messages that describe the actions of toolbar buttons as you place the mouse pointer over the buttons.

The current time on your computer clock is displayed in the field to the right of the status bar.

Graph Window Popup Menus

The Graph Window provides a variety of popup menus. Right-click any waveform to display relevant Workspace popup menu specific for that waveform.

Figure 3-9. Popup Menus



To activate the popup menus in the graph window, right-click window objects as follows:

Table 3-12. Graph Window Popup Menus

Right-Click Object	Popup Menu
Graph Window	Graph Window Popup Menu
Row	Row Popup Menu
Waveform or Waveform Name	Waveform Popup Menu
X Axis or Y Axis	Axis Popup Menu
Cursor Value	Cursor Value Popup Menu

Table 3-12. Graph Window Popup Menus (cont.)

Right-Click Object	Popup Menu
Cursor	Cursor Popup Menus

Graph Window Popup Menu

The Graph Window popup menu is accessed by right-clicking on the graph window.

Table 3-13. Graph Window Popup Menu Items

Icon	Item	Description
	Title	Specify a title for the active graph window. Same as **Window** > **Title** menu item.
	Fit Row Heights	Changes the row heights in the active graph window to see as many rows as possible. Item also available from the View Menu.
	Copy Window to Clipboard	Copies the graph window to the Clipboard (Windows) as a bitmap for paste as a bitmap image file. Item also available from the Edit Menu. (Available only on Windows platform.)
	Paste	Inserts the contents from the clipboard into the active graph window at the location of the mouse pointer when the menu was activated. Same as **Edit** > **Paste** menu item.
	Close	Closes the active graph window. Same as **File** > **Close** menu item.

Row Popup Menu

The Row popup menu is accessed by right-clicking on a row in the graph window.

Table 3-14. Row Popup Menu Items

Icon	Item	Description
	Smith Chart	Available only if the row contains a Smith Chart. Offers display options for changing the coordinates: • **Impedance** — Along the Z-parameter axis. • **Admittance** — Along the Y-parameter axis. • **Both** — Overlays the impedance and admittance grids.

Table 3-14. Row Popup Menu Items (cont.)

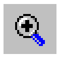
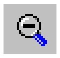
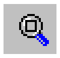
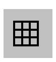
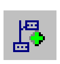
	Zoom In	Zooms in to view more detail in the X direction. Same as the **View** > **Zoom In** menu item.
	Zoom Out	Zooms out to view less detail in the X direction. Same as the **View** > **Zoom Out** menu item.
	View All	Adjusts the display to view all of the X and Y waveform data for all waveforms contained in the row. Item also available from the View Menu. See also the View All Activates Auto-Range checkbox on the EZwave Display Preferences General Options dialog box.
	To New Window	Opens a new window displaying the selected row.
	Overlay Window Waveforms	Plots all waveforms overlaid in a single row. See also the Overlay by ordinate data type checkbox on the EZwave Display Preferences General Options dialog box. Opposite of **Split** menu item.
	Split	Available only if the row contains multiple waveforms. Splits the waveforms into one individual row per waveform.
	Grid Lines	Available only if the row contains analog waveforms. Toggles the display of horizontal and vertical grid lines for all axes within the row. Same as the **View** > **Grid Lines** menu item.
	Add Cursor	Creates a new data cursor at the location of the mouse pointer. Same as the **Cursor** > **Add** menu item.
	Add Horizontal Cursor	Creates a new horizontal cursor at the location of the mouse pointer. Same as the **Cursor** > **Add Horizontal Cursor** menu item.
	Annotations	Accesses the Add Text Annotations option, enabling you to attach a text annotation to the row.
	Row Title...	Enables you to type a title that is displayed above the row. The title has a maximum length of 32 characters.

Table 3-14. Row Popup Menu Items (cont.)

	Row Height	Available only if the row contains analog waveforms. Offers options for changing the height of the row: • **Minimum** — Smallest allowed height for rows containing analog waveforms. • **Maximum** — Adjusts height so the row fills available vertical space in the window. • **Default** — Adjusts height back to the default.
	Paste	Inserts the contents from the clipboard into the row. Same as the **Edit** > **Paste** menu item.

Note

To set options related to row displays, use the Row Options.

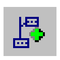
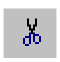
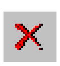
Waveform Popup Menu

The Waveform popup menu is accessed by right-clicking on a waveform or waveform name in the graph window. This popup displays different menu items based on the type of waveform selected, or if multiple waveforms are selected.

Table 3-15. Waveform Popup Menu Items

Icon	Item	Description
	Transformation	Available only if the waveform contains complex data (for example, from a frequency domain analysis). Opens the Transformation Options to enable changes the transformation that is applied to the underlying complex data.
	Radix	Available only if the waveform supports radix display, which would be digital buses or waveforms containing integer data. Changes the radix that is used when displaying the waveform. If multiple buses are selected at once, this change applies to all selected buses.
	Data Point Symbols	Available only for analog waveforms. Toggles the visibility of graphic symbols, shown at each data point.
	Hide Waveform	Toggles the visibility of the waveform. The waveform name remains visible.
	Analog to Digital / Digital to Analog	Used to convert the waveform between the analog and digital domains. Opens a dialog box to specify conversion options.
	Overlay Selected Waveforms	Overlays the selected waveforms in a single row. Only available when 2 or more waveforms are selected.

Table 3-15. Waveform Popup Menu Items (cont.)

	Groups	Creates a waveform group, such that multiple waveforms can be manipulated together. Only available when 2 or more waveforms are selected.
	Annotations	Options to create, show, or hide text annotations.
	Set as X Axis	Sets currently selected waveform as a new x axis.
	Move to New Y Axis	Available only for analog waveforms. The item is disabled if the waveform is the only one on the y axis containing it. The result of this operation is the creation of a new y axis within the row, with the waveform moved to the new axis.
	Circle Visibility...	Available only for Smith Chart circle plots. Opens the Circle Visibility dialog box for selection of which circles are visible.
	Drawing Mode	Displays the waveform in one of five modes: Continuous, Sampled, Scattered, Spectral, or Railroad.
	Add Cursor	Adds a new data cursor at the mouse pointer location on the x axis.
	Add Horizontal Cursor	Creates a new horizontal cursor at the mouse pointer location on the y axis.
	Cut	Removes the waveform from the graph window and places it on the clipboard.
	Copy	Places a duplicate of the waveform on the clipboard.
	Paste	Inserts the contents of the clipboard into the active graph window within the current row.
	Delete	Removes the waveform from the active graph window.
	Y Axes Properties	Opens the Axis Properties Dialog Box.
	Properties	Opens the Waveform Properties Dialog Box (For Analog Waveforms).

Axis Popup Menu

The Axis popup menu is accessed by right-clicking on an x axis or y axis in the graph window.

Figure 3-10. X and Y Axis Popup Menus

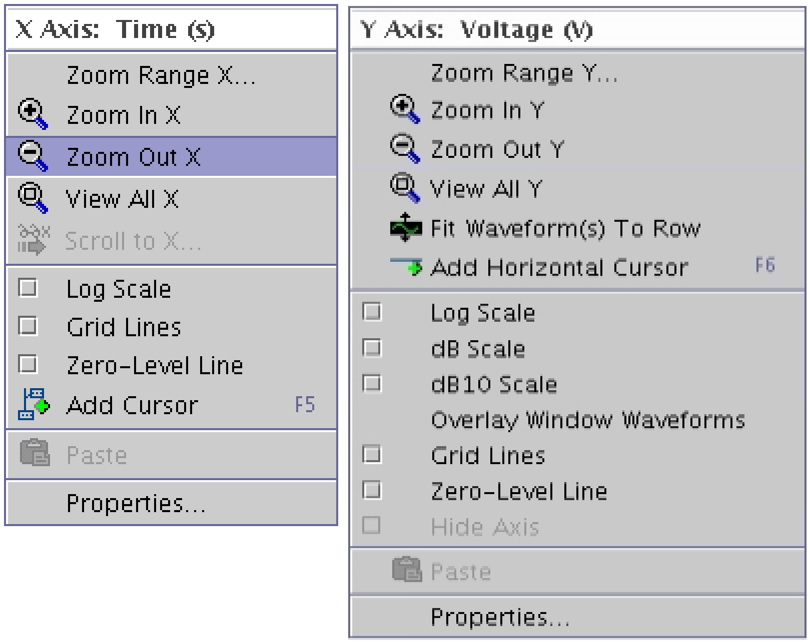
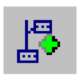


Table 3-16. Axis Popup Menu Items

Icon	Item	Description
	Zoom Range X/Y	Opens the X / Y Axis - Min/Max Settings dialog box. Use to zoom between two locations on the axis.
	Zoom In X/Y	Zooms in to view more detail for the waveforms on the axis.
	Zoom Out X/Y	Zooms out to view less detail for the waveforms on the axis.
	View All X/Y	Adjusts the display to view all of the waveform data for all waveforms on the axis.
	Fit Waveform(s) To Row (y axis only)	Adjusts the y axis only, to fit all of the waveforms.
	Add Horizontal Cursor (y axis only)	Creates a horizontal cursor at that point on the y axis. By default, the first crossing point with the waveform is shown with its measurement.
	Scroll to X (x axis only)	Moves the view to a new X location, using the Scroll X dialog box.
	Log Scale	Toggles between linear and log base 10 scaling.
	dB Scale	(y axis only) Changes the y axis to decibels ($20\log_{10}$ for amplitude ratios).

Table 3-16. Axis Popup Menu Items (cont.)

	dB10 Scale	(y axis only) Changes the y axis to decibels ($10\log_{10}$ for power ratios).
	Overlay Window Waveforms (y axis only)	Plots all waveforms overlaid in a single row. See also the Overlay by ordinate data type checkbox on the EZwave Display Preferences General Options dialog box.
	Grid Lines	Available only if the row contains analog waveforms. Toggles the display of either the horizontal (X) or vertical (Y) grid lines for the axis. Same as the **View** > **Grid Lines** menu item.
	Add Cursor (x axis only)	Creates a new vertical data cursor at that point on the x axis in the active graph window.
	Zero-Level Line	(y axis only) Toggles the visibility of a reference line which is shown at the zero-level location.
	Hide Axis	Toggles the visibility of the axis. When an axis is hidden, it is temporarily removed from the graph window, and all waveforms contained on the axis are also temporarily removed. This item is only enabled for rows containing multiple y axes.
	Paste	Inserts the clipboard contents onto the axis. The result of the paste differs based on the type of axis: **For Y axes:** the paste will use this y axis if the data type matches the waveforms being pasted; otherwise new y axes will be created within the same row. **For X axes:** the paste is only permitted if the data type of the waveforms being pasted matches the X data type for the graph window.
	Properties	Opens the Axis Properties Dialog Box.

Cursor Popup Menus

The Cursor popup menu is accessed by right-clicking on a cursor in the graph window. Different options are available for vertical and horizontal cursors.

Vertical Cursors 102
Horizontal Cursor 104

Vertical Cursors

The Vertical cursor popup menu is accessed by right-clicking on a vertical cursor in the graph window.

Figure 3-11. Vertical Cursor Popup Menu

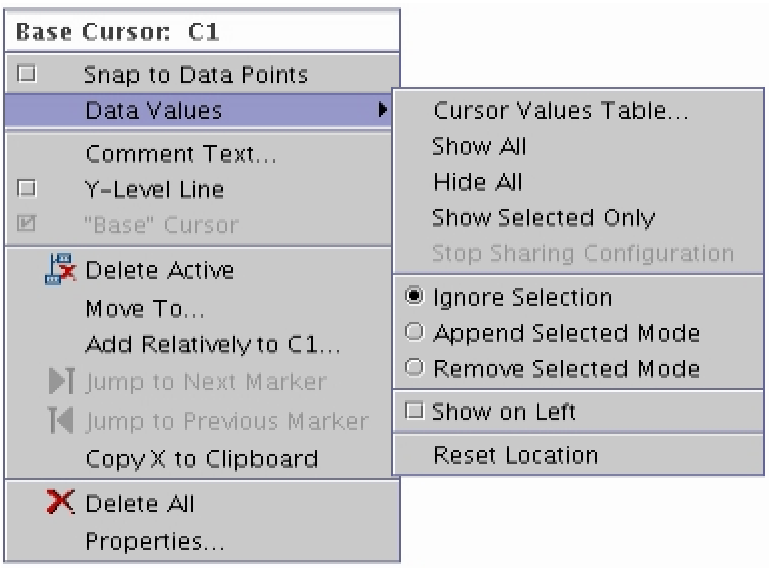


Table 3-17. Vertical Cursor Popup Menu Items

Icon	Item	Description
	Snap to Data Points	Specifies that the cursor is snapped to data points.
	Data Values	Specifies which waveform data values are visible.
	Data Values > Cursor Values Table	Specifies which waveform data values are visible.
	Data Values > Show All	Specifies all waveform data values are visible.
	Data Values > Hide All	Specifies no waveform data values are visible.
	Data Values > Show Selected Only	Specifies only selected waveform data values are visible.
	Data Values > Stop Sharing Configuration	Specifies that the set of visible waveform data values is no longer shared with another cursor.

Table 3-17. Vertical Cursor Popup Menu Items (cont.)

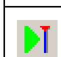
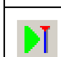
Icon	Item	Description
	Data Values > Ignore Selection	Specifies that the values which are visible does not depend on which waveforms are selected.
	Data Values > Show Selected Mode	When **Edit > Options > Cursors > “Show/Hide Dynamically Cursor Values Based on Waveform Selection”** is set, specifies that only selected waveforms are visible.
	Data Values > Hide Selected Mode	When **Edit > Options > Cursors > “Show/Hide Dynamically Cursor Values Based on Waveform Selection”** is set, specifies that selected waveforms are not visible.
	Data Values > Append Selected Mode	When **Edit > Options > Cursors > “Append/ Remove Dynamically Cursor Values Based on Waveform Selection”** is set, specifies that when a waveform is selected it should be added to the list of visible values.
	Data Values > Remove Selected Mode	When **Edit > Options > Cursors > “Append/ Remove Dynamically Cursor Values Based on Waveform Selection”** is set, specifies that when a waveform is selected it should be removed from the list of visible values.
	Data Values > Show On Left	Specifies that data values are shown on the left by default.
	Comment Text	Opens a dialog box to specify comment text.
	Y-Level Line	Specifies that a Y-level line is displayed.
	“Base” Cursor	Specifies that the active cursor is the base cursor. Available only when the active cursor is not the base cursor.
	Move To	Opens a dialog box to specify a new X location to move the cursor to.
	Add Relatively to <cursor>	Opens a dialog box where one or more X values can be specified for creating new cursors relative to the selected cursor.
	**Jump to Next Marker**	Enables you to jump to the next marker. Available only if markers have been set Using the Event Search Tool.
	**Jump to Previous Marker**	Enables you to jump to the previous marker. Available only if markers have been set Using the Event Search Tool.

Table 3-17. Vertical Cursor Popup Menu Items (cont.)

Icon	Item	Description
	Copy X to Clipboard	Copies the X value of the cursor to the clipboard.
	**Delete All**	Deletes all annotations.
	Properties	Opens the Cursor Properties Dialog Box, which enables you to specify the font, color and line style for the cursor.

Horizontal Cursor

The Horizontal cursor popup menu is accessed by right-clicking on a horizontal cursor in the graph window.

Table 3-18. Horizontal Cursor Popup Menu Items

Icon	Item	Description
	Comment Text	Enables a note of up to 64 characters to be attached to the cursor.
	“Base” Cursor	Specifies this cursor as the base cursor. Available only when the active cursor is not already the base cursor.
	Show in All Compatible Y-Axes	Duplicates the cursor in other graphs that contain the same waveform.
	Data Values	Displays the Data Values Dialog Box, for the selected waveform. This dialog box contains information about the x-axis value at each crossing point, and enables you to save the results to text format, or copy them to the clipboard.
	Move To	Enables a new Y location for the cursor to be entered.
	Add Relatively to <cursor>	Opens a dialog box where one or more Y values can be specified for creating new cursors relative to the selected cursor.
	Copy Y to Clipboard	Copies the Y value of the cursor to the clipboard.
	**Delete All**	Deletes all annotations.
	Properties	Opens the Cursor Properties Dialog Box, which enables you to specify the font, color, and line style for the cursor.

Cursor Value Popup Menu

The Cursor Value popup menu is accessed by right-clicking on a cursor value in the graph window.

Table 3-19. Cursor Value Popup Menu Items

Item	Description
Hide Value	Temporarily hides the cursor data value. To show the cursor data value, right-click the cursor and select **Data Values** > **Show All**.
Y-Level Line	Displays a line across the graph window showing the Y value.
"Base" Y-Level Line	Selects the current Y-level line as the "base." Delta-y values for waveforms that intersect this cursor are based on this line.
Copy Value to Clipboard	Copies the full precision value to the clipboard.

The Cursor Value popup menu for compound waveforms contains the following additional items:

Table 3-20. Cursor Value Popup Menu Items for Compound Waveforms

Item	Description
Hide Other Values	Hides values for other elements of the waveform.
Show Other Values	Displays values for other elements of the waveform.
Show/Hide...	Opens the Cursor Value dialog box.

Waveform List Panel Popup Menus

The Waveform List Panel has several popup menus that provide quick access to a variety of database and waveform operations.

To activate the popup menus in the graph window, right-click window objects in the Waveform List Panel as follows:

Table 3-21. Right-Click Popup Menus

Right-Click Object	Popup Menu
Waveform List	Waveform List Popup Menu
Database Name	Database Popup Menu
Folder	Folder Popup Menu
Hierarchy	Hierarchy Popup Menu
Waveform Name	Waveform Name Popup Menu
Multiple Waveform Names	Selected Waveforms Popup Menu

Waveform List Popup Menu

To access the Waveform List popup menu, right-click the **Currently Open Databases** text (or anywhere within the panel except over a waveform database name or waveform name).

Note

The Database list only displays waveform database files. To open the other supported file types, use the **File** > **Open** menu item.

Table 3-22. Waveform List Popup Menu Items

Icon	Item	Description
	Manage Database Groups	Only visible when at least one database is loaded. Opens the Manage Database Groups dialog box, used to assign databases to groups for use with Tandem Mode. . Refer to "Managing Database Groups for Tandem Mode" on page 128.
	Exclude All Databases from Group	Only visible when at least one database is loaded and at least one database group exists. Removes all databases from their groups and deletes the groups. Refer to "Managing Database Groups for Tandem Mode" on page 128.

Table 3-22. Waveform List Popup Menu Items (cont.)

	Include All Databases in Tandem	Only visible if no database groups have been created. Includes all databases in the Tandem Mode. Refer to "Plotting Waveforms With the Same Name Using Tandem Mode" on page 126.
	Exclude All Databases from Tandem	Only visible when Tandem Mode is selected and no database groups have been created. Excludes all databases in the Tandem Mode. Refer to "Plotting Waveforms With the Same Name Using Tandem Mode" on page 126.
	Open	Same as the **File** > **Open** menu item to open any supported file type.
	Expand All	Expands all currently open databases.
	Collapse All	Collapses all currently expanded waveform databases.
	Close All Databases	Closes all currently loaded databases.
	Save All Databases	Opens the Save As Dialog Box to enable saving all databases that contain unsaved changes.
	Most Recently Used (1,2,3,4,0) Database List	Same as the **File** > **Database List** menu item for opening previously opened waveform databases. Click **0 More Databases** to open an additional window with older databases for selection.
	Hide Waveform List	Temporarily hides the Waveform List panel. To show the panel again, choose **View** > **Waveform List**.
	Find	Opens the Find Tool Dialog Box to enable searching in the structure list.

Tip

You can control the visibility of the Database list items by choosing General Options.

Database Popup Menu

To access the Database popup menu, right-click any database name in the waveform list panel.

Table 3-23. Database Popup Menu Items

Item	Description
Update Waveforms	Opens the "Update Waveforms Dialog Box" on page 592 to update the currently plotted waveforms in this database. Only waveforms that match the names within the database are updated. The update can either replace the existing waveforms, or they can be shown overlaid with the matching waveforms from the database. All waveforms, those in the active graph window, or a specific set of waveforms can be selected to update.
Run Filter	Only visible when the database contains multiple runs. Opens the "Parameter Table Dialog Box" on page 568 that enables you to choose which runs are to be shown, highlighted or grouped before you plot them. The icon for the database changes to  indicating that a filter is currently applied.
Plot (Overlaid)	Plots all waveforms at this level within a single row at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot (Stacked)	Plots all waveforms at this level in separate rows at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot All (Overlaid)	Plots all waveforms in the database within a single row at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot All (Stacked)	Plots all waveforms in the database in separate rows at the bottom of the active graph window. A new graph window is created if one does not already exist.
Include in Tandem	Choose Include from Tandem to include the selected database from Tandem Mode.
Exclude in Tandem	Only visible when Tandem Mode is selected. Choose Exclude from Tandem to exclude the selected database from Tandem Mode. The icon adjacent to the database name changes to indicate the database is excluded.
Manage Database Groups	Only visible if at least one database group exists. Opens the Manage Database Groups dialog box, used to assign databases to groups for use with Tandem Mode. Refer to "Managing Database Groups for Tandem Mode" on page 128.
Exclude from Group	Removes the selected database from its group.
Rename	Opens a dialog box to rename the database.
Descriptive Label...	Opens a dialog box to add a descriptive text label. This label displays next to the database name.

Table 3-23. Database Popup Menu Items (cont.)

Item	Description
Save As...	Opens a "Save As Dialog Box" on page 583 for saving the selected databases to disk.
Close	Closes the selected databases. If a database contains unsaved changes or has waveforms currently plotted in one or more graph windows, a confirmation box is displayed.
Reload	Reloads the selected databases from the disk file if the simulation has finished, or updates the plotted waveforms for ongoing simulations using JWDB format.
Show Boundary Elements	Turns on boundary elements visibility for the selected database. Only accessible for .wtcl databases.
Expand Selected	Expands the selected database.
Collapse Selected	Collapses the selected database.
Properties	Opens the "Database Properties Dialog Box" on page 448 displaying properties and statistics associated with the waveform database.

Folder Popup Menu

To access the Folder popup menu, right-click a folder in a database.

Figure 3-12. Folder Popup Menu

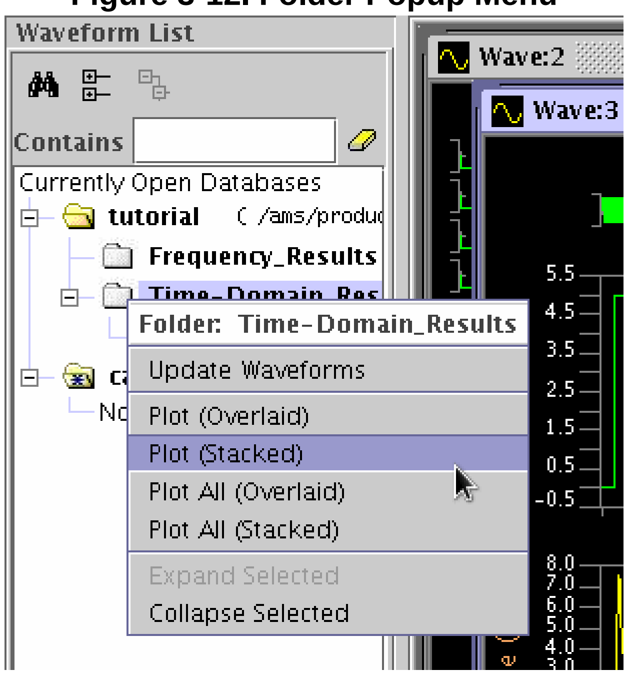


Table 3-24. Folder Popup Menu Items

Item	Description
Update Waveforms	Updates the active graph window using the waveforms from this database. Only waveforms in the graph window that match the names within the database are updated. The update can either replace the existing waveforms, or they can be shown overlaid with the matching waveforms from the database.
Run Filter	Only visible when the analysis folder contains multiple runs. Opens the Parameter Table Dialog Box that enables you to choose which runs are to be shown, highlighted or grouped before you plot them. The icon for the folder changes to  indicating that a filter is currently applied.
Plot (Overlaid)	Plots all waveforms at this level within a single row at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot (Stacked)	Plots all waveforms at this level in separate rows at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot All (Overlaid)	Plots all waveforms in the database within a single row at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot All (Stacked)	Plots all waveforms in the database in separate rows at the bottom of the active graph window. A new graph window is created if one does not already exist.
Expand Selected	Expands the selected database.
Collapse Selected	Collapses the selected database.

Hierarchy Popup Menu

To access the Folder popup menu, right-click a folder in a database.

Figure 3-13. Hierarchy Popup Menu

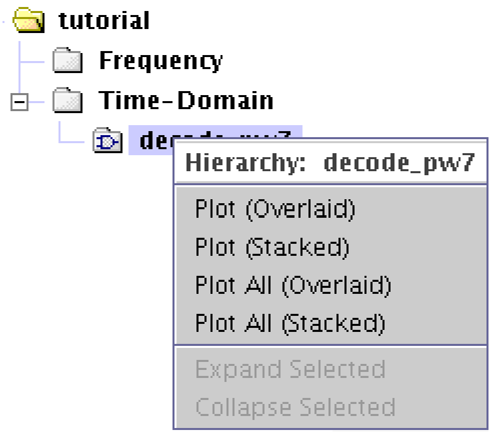


Table 3-25. Hierarchy Popup Menu Items

Item	Description
Plot (Overlaid)	Plots all waveforms at this level within a single row at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot (Stacked)	Plots all waveforms at this level in separate rows at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot All (Overlaid)	Plots all waveforms in the database within a single row at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot All (Stacked)	Plots all waveforms in the database in separate rows at the bottom of the active graph window. A new graph window is created if one does not already exist.
Expand Selected	Expands the selected database.
Collapse Selected	Collapses the selected database.

Waveform Name Popup Menu

To access the Waveform Name popup menu, right-click any waveform name in the Waveform List panel.

Figure 3-14. Waveform Name Popup Menu

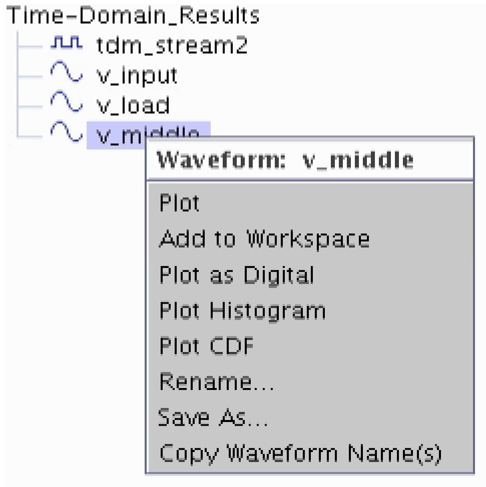


Table 3-26. Waveform Name Popup Menu Items

Item	Description
Plot	Plots the waveform in a new row at the bottom of the active graph window. A new graph window is created if one does not already exist.
Add to Workspace	Adds the selected waveforms to the Waveform Calculator Workspace.
Plot as Digital	Plots analog waveforms as digital waveforms, using default conversion parameters. See also "Transforming Analog Waveforms to Digital" on page 275.
Plot Histogram	Plots a histogram of the waveform data.
Plot CDF	Plots a cumulative distribution function of the waveform data.
Rename	Opens a dialog box to rename the waveform.
Save As	Opens the Save As Dialog Box to save a copy of the waveform (with options to overwrite the current waveform).
Copy Waveform Name(s)	Copies the selected waveform's name to the clipboard.

Selected Waveforms Popup Menu

To access the Selected Waveforms popup menu, select multiple waveforms in the Waveform List panel and right-click.

Figure 3-15. Selected Waveforms Popup Menu

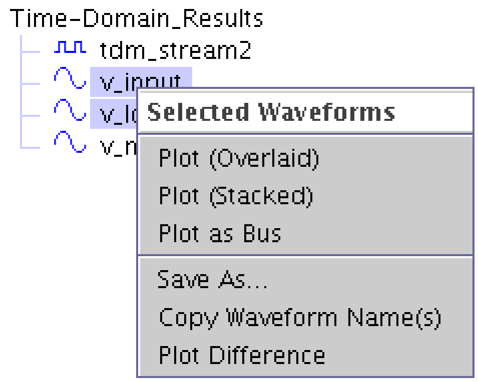


Table 3-27. Selected Waveforms Popup Menu Items

Item	Description
Plot (Overlaid)	Plots the selected waveforms within a single row at the bottom of the active graph window. A new graph window is created if one does not already exist.
Plot (Stacked)	Plots the selected waveforms in separate rows at the bottom of the active graph window. A new graph window is created if one does not already exist.
Save As	Opens the Save As Dialog Box to save a copy of the waveform (with options to overwrite the current waveform).
Copy Waveform Name(s)	Copies the selected waveform's name to the clipboard.
Plot Difference	Plots the difference between the two selected waveforms. Only available when two waveforms are selected. The second waveform selected is subtracted from the first waveform selected.

Workspace Popup Menu

To access the Workspace popup menu, right-click in the workspace area (the area behind the Wave window(s) which is blue by default).

Table 3-28. Workspace Popup Menu Items

Item	Description
New	Creates a new workspace.
Rename	Opens a dialog box to give the active workspace a user defined name.

Table 3-28. Workspace Popup Menu Items (cont.)

Item	Description
Delete	Removes the currently active workspace. This item is unavailable if there is only one workspace.
Shift Left	Moves the currently active **Workspace** tab one position to the left. This item is unavailable if the currently active workspace's tab is the leftmost tab.
Shift Right	Moves the **Workspace** tab one position to the right.
This item is unavailable if the currently active workspace's tab is the rightmost tab.	Reorder
Opens the Reorder Workspaces dialog box, from which you can interactively set the order of all workspaces.	Paste
Creates a new graph window in the workspace and pastes the contents of the clipboard in the graph window.	This item is unavailable if the clipboard is empty.
Options...	Displays the Workspace Options. Use this to define how the workspaces are displayed in the application window. You may also set the application to support a single or multiple workspaces from this dialog box.

Keyboard and Mouse

EZwave has a number of keyboard and mouse shortcut and access features.

Keyboard Commands 115
Mouse Strokes 115

Keyboard Commands

Frequently used EZwave keyboard commands are described here.

Table 3-29. Frequently Used Keyboard Commands

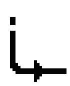
Press	To
F2	Unselect All
F5	Add Cursor at Mouse Pointer Location
Ctrl + A	Select All
Ctrl + C	Copy
Ctrl + N	Open New Graph Window
Ctrl + V	Paste
Ctrl + W	Close Window
Ctrl + X	Cut
ESC	Dismiss a popup menu, abort an in-progress mouse drag operation or close a dialog box without applying selections.

Mouse Strokes

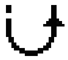
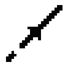
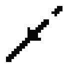
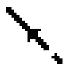
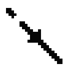
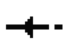
Mouse Strokes provide a convenient way to perform common tasks by simply drawing shapes using the mouse. For example, drawing the letter “D” deletes the current set of selected objects.

By default, mouse strokes are executed with the middle mouse button. If you do not have a middle mouse button, use the **Edit** > **Options** menu item and then select a keyboard modifier from the dialog box. Use these keyboard modifiers together with the left mouse button to draw strokes.

Mouse strokes available with this application are as follows:



File > **Open**

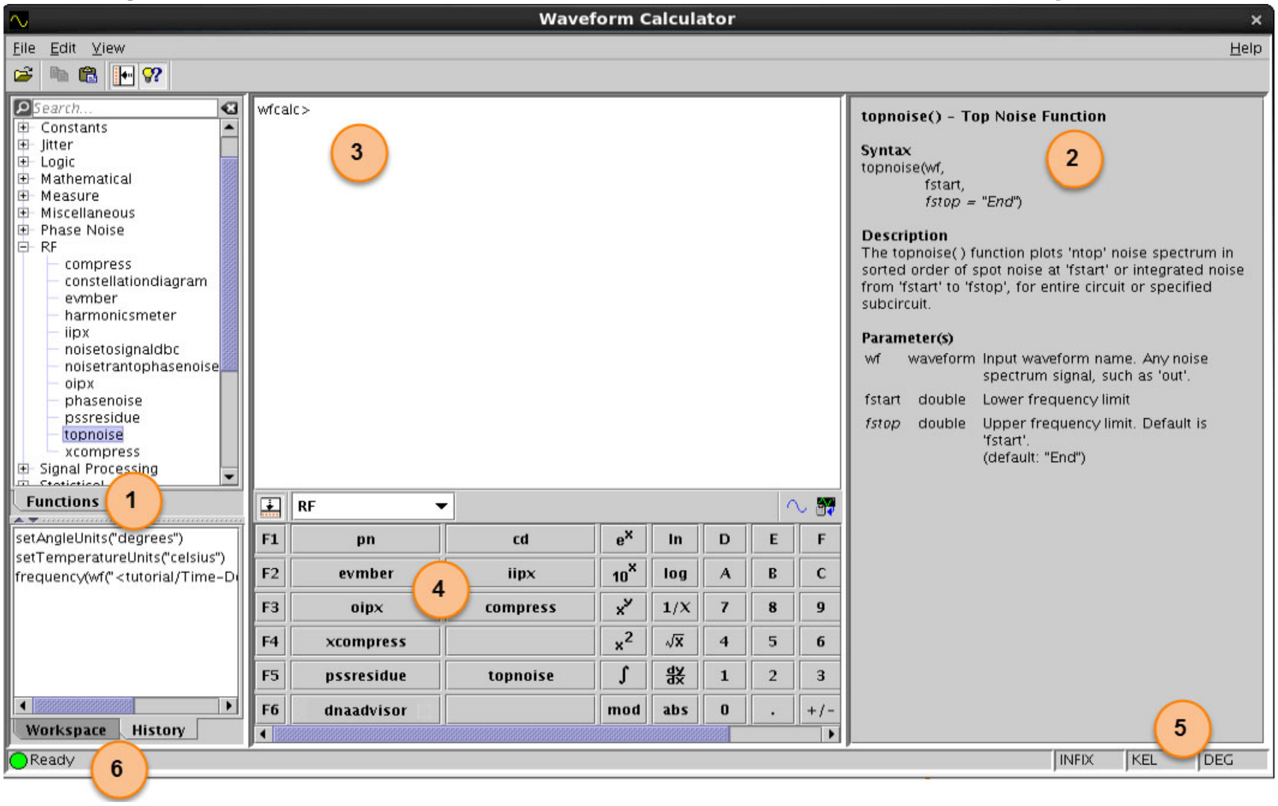
	**Edit** > **Delete**
	**Edit** > **Unselect All**
	**View** > **Zoom Out**
	**View** > **Zoom In**
	**View** > **View All**
	View Area - zooms in to view the area covered by the stroke.
	Scroll X Location to Center of Graph Window
	**File** > **Close** if done within an active graph window or **File** > **Exit** if done on the workspace
	Displays all available mouse strokes

Waveform Calculator GUI

The Waveform Calculator enables you to post-process waveforms for advanced analyses or debugging.

Overview

Figure 3-16. Waveform Calculator with Button and Function Help Panels



The user interface of the Waveform Calculator includes a number of components, described in the following sections.

1 - Chooser Panel

To show or hide the Chooser panel, click the button. The Chooser panel appears on the left-hand side of the Waveform Calculator and has three tabbed panes:

- The **Functions** tab contains Built-In Functions and constants, and also any User-Defined Functions that you have registered with the calculator. Double-clicking a function name in the **Functions** tab adds it to the Shell panel. Double-clicking the function will display a new Function panel (temporarily replacing the Button panel), enabling you to enter the parameters for the function. Click **OK** to add the expression to the Shell panel.

Tip

Some functions have dedicated dialog boxes that enable you to specify additional options. If available, click the **Advanced** button in the Function panel to display the dedicated dialog box.

- The **Workspace** tab contains the results generated after you press Enter (or click the **Eval** button) to evaluate an expression. Results can be scalars, vectors, and waveforms. If you right-click an item, a popup menu enables you to copy or plot the item, or clear all items.
- The **History** tab displays a list expressions that you have evaluated in the calculator (when you are in **INFIX** mode). If you right-click an item, a popup menu enables you to copy or evaluate the item, or clear all items.

2 - Function Help

Display Waveform Calculator function help with the **View** > **Function Help** menu item or toolbar button. The Waveform Calculator displays the description, syntax, and parameters for the selected function.

3 - Shell Panel

When you select buttons on the calculator, edit function parameters, or drag and drop waveform names, the Shell panel is updated. You can also edit the text directly. Once you have the desired entries, press Enter (or click the **Eval** button). The results display in the **Workspace** tab of the Choose panel and in the Shell panel.

Expressions may be entered using IEEE notation (for example, 5T, 3.2u), SPICE notation (for example, 7.1MEG, 0.2N), or engineering notation (for example, 1.1e+3, 6e-2).

Table 3-30. Calculator Shell Key Binds

ENTER	Evaluate expression
SHIFT + ENTER	Evaluate expression and then display prompt with same expression
SHIFT + CTRL +V	Paste and keep line jump
MIDDLE MOUSE BUTTON	Paste and keep line jump
UP	Browse up command history
DOWN	Browse down command history
CTRL + L	Clear terminal
CTRL + C	Copy
CTRL + V	Paste
CTRL + X	Cut
CTRL + A	Select all
HOME	Go to the beginning of the expression
END	Go to the end of the expression

Table 3-30. Calculator Shell Key Binds (cont.)

BACKSPACE	Delete characters or selection to the left
CTRL + BACKSPACE	Delete word or selection to the left
DEL	Delete characters or selection to the right
CTRL + DEL	Delete word or selection to the right

Table 3-31. Calculator Shell Mouse Operations

Double left-click	Select word or function
Triple left-click	Select line
CTRL + left-click	Trigger generic auto-form mode
Right-click	Contextual menu

Tip

See also "Waveform Calculator Shell Commands" on page 314.

4 - Button Panel

To show or hide the Button panel, click the button. There are functions located on the buttons of the Waveform Calculator and also in the functions on the **Functions** tab one the Chooser panel of the calculator. The Button panel does not include a dedicated button for every function or operator. The Button panel changes based on the level you choose from the dropdown list:

- Complex Buttons
- Jitter Buttons
- Logical Buttons
- Phase Noise Buttons
- RF Buttons
- Signal Processing Buttons
- Statistical Buttons
- Trigonometric Buttons

5 - Evaluation Mode Bar

On the bottom-right corner of the Waveform Calculator, the Evaluation Mode Bar shows the input and evaluation mode (e.g., **INFIX** for Infix mode) and the trigonometric angle mode (e.g., **DEG** for Degrees).

6 - Waveform Calculator Status Bar

The Status Bar of the Waveform Calculator shows whether the calculator is in busy or ready state. It also shows error messages if the input expression has a syntax error.

Use the **Edit** > **Options** menu item of the Waveform Calculator application window to control the calculator settings.

Related Topics

Waveform Calculator

Waveform Calculator Dialog Boxes

bloc

Chapter 4
Add Waveforms

This section describes how to plot and manage different types of waveforms.

Waveform Basics **122**
Starting EZwave and Loading a Database 122
Single Waveform Plots 123
Multiple Waveform Plots 125
Managing Waveform Groups 131
Rules When Plotting Analog and Digital Waveforms 134

Modifying Axis Settings **136**
Aligning Y Axes With Different Scales 136
Creating Multiple Y Axes 137
Changing Y-Axis Settings 138

Modifying Waveforms **139**
Hiding Waveforms 139
Unhiding Hidden Waveforms 139
Modifying Waveform Properties 140
Modifying the Appearance of Row Titles 141

Specialized Plotting **142**
Plotting Complex-Valued Waveforms 142
Plotting wreal Waveforms 143
Plotting Multiple Bit Waveforms as a Bus 143
Displaying Bus Values as a String 144
Plotting Assertions 145
Plotting Histograms 147
Plotting CDFs 148

Compound Waveforms **151**
Displaying Compound Waveforms 151
Displaying Compound Waveforms as Single Elements 152
Merging Waveforms Into a Compound Waveform Using an Index File 154
Digital Compound Waveforms 155
Index File Format 156

Updating Waveform Data **158**
Configuring Automatic Reloading of Waveform Data 158
Manually Updating Waveforms 160

Waveform Basics

You can plot and manage single or multiple waveforms.

Starting EZwave and Loading a Database 122
Single Waveform Plots 123
Plotting a Single Waveform 123
Moving a Plotted Waveform. 124
Copying a Plotted Waveform 124
Multiple Waveform Plots 125
Plotting Multiple Waveforms 125
Plotting Waveforms With the Same Name Using Tandem Mode 126
Managing Database Groups for Tandem Mode 128
Plotting the Difference Between Two Waveforms. 130
Creating an XY Plot 130
Managing Waveform Groups. 131
Grouping Waveforms 131
Adding a Waveform to a Group 132
Removing a Waveform From a Group 133
Creating an Empty Waveform Group. 133
Ungrouping Waveforms 134
Rules When Plotting Analog and Digital Waveforms 134

Starting EZwave and Loading a Database

This topic escribes how you launch EZwave and load a database containing waveform data.

Procedure

1. Launch the EZwave Viewer by entering the following command at the command line:

```
ezwave
```

2. Choose > **Open** or click the **Open** toolbar  icon, and select a database to load into EZwave.

   It must be one of the supported file types, as listed in the topic "Supported File Types" on page 45.

Single Waveform Plots

You can plot a single waveform, and move and copy a plotted waveform.

Plotting a Single Waveform 123
Moving a Plotted Waveform 124
Copying a Plotted Waveform 124

Plotting a Single Waveform

You can plot a single waveform in a graph window.

Procedure

1. Find the name of the waveform you want to plot in the lower section of the Waveform List Panel and use one of the following methods to plot it:

   - Double-click the waveform name.

     If an active graph window is open, the waveform is plotted in that window. Otherwise, the waveform is plotted in a new graph window.

   - Right-click the waveform name and choose **Plot** from the Waveform Name popup menu.

     If an active graph window is open, the waveform is plotted in that window. Otherwise, a new graph window opens and the waveform is plotted in it.

2. You can also do one of the following drag and drop operations:

If you want to...	Do the following:
Plot the waveform in a new graph window in the current workspace.	Drag the waveform onto the workspace.
Plot the waveform in a new row at the top of an existing graph window.	Drag the waveform to the top of the graph window.
Overlay the waveform onto an existing waveform in the same row in a graph window.	Drag the waveform onto the existing waveform or waveform label in the graph window.
Plot the waveforms in a new row above an existing waveform in a graph window.	Drag the waveform above the existing waveform in the graph window.
Plot the waveforms in a new row between existing rows in a graph window.	Drag the waveform between the existing rows in the graph window.
Plot the waveforms in a new row at the bottom of a graph window.	Drag the waveform to the bottom of the graph window.
Overlay the waveform onto the y axis.	Drag the waveform onto the y axis.

If you want to...	Do the following:
Plot the waveform on a new graph window in an existing workspace.	Drag the waveform onto the **Workspace** tab.
Plot the waveforms in graph window in a new workspace.	Drag the waveform onto an empty part of the workspace tabbed area.

Related Topics

Waveform List Panel

Waveform Name Popup Menu

Plotting Multiple Waveforms

Moving a Plotted Waveform

You can move a plotted waveform from one row to another row within a graph window or from one graph window to another graph window.

Procedure

1. Select the waveform label in the graph window. This may be to the right, left, above or below the plotted waveform, depending on layout option.
2. Drag the waveform to a different row in the same graph window or a row in a different graph window.

Copying a Plotted Waveform

You can copy a plotted waveform from one row to another row within a graph window or from one graph window to another graph window.

Procedure

1. Select the waveform label in the graph window. This may be to the right, left, above or below the plotted waveform, depending on layout option.
2. Do one of the following:
    - Press and hold down the Ctrl key while dragging the waveform to a different row in the same graph window, or a row in a different graph window.
    - Right-click the waveform, choose **Copy**, then place the cursor in a different row in the same graph window, or a row in a different graph window, and choose **Paste**.

Multiple Waveform Plots

You can plot a multiple waveforms, plot the difference between two waveforms, and create XY plots.

Plotting Multiple Waveforms 125
Plotting Waveforms With the Same Name Using Tandem Mode 126
Managing Database Groups for Tandem Mode 128
Plotting the Difference Between Two Waveforms 130
Creating an XY Plot 130

Plotting Multiple Waveforms

You can plot multiple waveforms in one procedure.

Restrictions and Limitations

- Waveforms with incompatible x-axis domains cannot appear together in the same graph window. X-axis domains are compatible if they share the same units, even if they have different unit titles, for example Frequency(Hz) and DIFF(Hz).
- For databases loaded from *.fsdb* files, only loaded waveforms are displayed in the Waveform List Panel. Loaded waveforms are waveforms that have been plotted once or waveforms from a hierarchy selected once in the tree view of the Waveform List Panel. Refer to "Loading .fsdb and .tr0 Files" on page 1272.

Procedure

1. To select the waveforms to be plotted in the Waveform List Panel, press the Ctrl key and click the name of each waveform; or press the Shift key and click the first and last waveforms in a sequence.
2. Use one of the following methods to plot them:
   - Drag the waveforms onto the workspace or a graph window.

     Digital waveforms are plotted stacked in separate rows. Analog waveforms are plotted overlaid in a single row regardless of the order of selection.
   - Right-click and choose one of the following from the Selected Waveforms popup menu:

If you want to...	Do the following:
Plot all the waveforms together in the same row.	Choose **Plot (Overlaid)**
Plot all the waveforms separately in different rows.	Choose **Plot (Stacked)**
Plot the waveforms as a bus.	Choose **Plot as Bus**

Note

When creating a bus using **Plot as Bus** and a conversion from analog to digital is needed (at least one of the bits is analog), a default threshold is calculated as *(min_y + max_y)/2* from the first analog waveform that is not flat *(min_y == max_y == 0)* by iterating from MSB to LSB. Alternatively, a single threshold, or two thresholds, may be specified. Refer to the **Conversion** tab in "Transformation Options" on page 505.

If an active graph window is open, the newly added waveforms are added in new rows at the bottom of the window. Otherwise, the waveforms are added to a new graph window.

Related Topics

Waveform List Panel

Selected Waveforms Popup Menu

Transformation Options

Plotting Waveforms With the Same Name Using Tandem Mode

Plotting Waveforms With the Same Name Using Tandem Mode

You can automatically plot waveforms with the same name from different databases, by activating tandem mode. This is useful, for example, when you want to visually compare waveforms from multiple simulation runs. When tandem mode is activated and source waveforms are plotted using the EZwave GUI, a search is made for waveforms with the same names in different databases and plotted with the source waveforms.

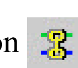
Tip

You can create database groups, and then only databases in the same group as the source waveform's database are searched for waveforms with the same name. If no database groups have been created, all opened databases are searched. Refer to "Managing Database Groups for Tandem Mode" on page 128.

Restrictions and Limitations

- Some graphical plots are not considered for tandem mode: Plot CDF, Plot Histogram, Plot as Bus, Plot Difference.
- Measures and expressions are not considered, regardless of whether they are computed from the Waveform Calculator, Measurement Tool, or any other dialog box from the **Tools** menu.

Procedure

1. Choose **View** > **Tandem Mode** or click the **Tandem Mode** icon  from the toolbar.

Tandem mode activates.

2. Use one of the following methods to plot a waveform:
    - Double-click a waveform in the Waveform List.
    - Drag a waveform onto the workspace or a graph window.
    - Right-click a folder, hierarchy or waveform, and choose one of the following from the popup menu:

If you want to...	Do the following:
Plot (when a single waveform is selected).	Choose **Plot**
Plot all the waveforms together in the same row.	Choose **Plot (Overlaid)**
Plot all the waveforms separately in different rows.	Choose **Plot (Stacked)**
Plot all the waveforms in the database or folder in the same row.	Choose **Plot All (Overlaid)**
Plot the waveforms in the database or folder separately in different rows.	Choose **Plot All (Stacked)**

3. Optionally:

If you want to...	Do the following:
Always display the Database Name for each waveform, when using tandem mode.	Choose **Format** > **Waveform Names Display** to open the Waveform Names Display Dialog Box and select Show Database Name in Tandem Mode. Alternatively, choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Waveform Name. Select the option Show Database Name in Tandem Mode on the Waveform Name Options dialog box.
Plot waveforms in the same database with the same color, when using tandem mode.	Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences dialog box, then expand the Fonts and Colors folder on the left side, and choose Waveform Colors. Select the option Use Same Waveform Color Per Database in Tandem Mode on the Waveform Colors Options dialog box. Enabled by default.

If you want to...	Do the following:
Apply tandem mode on a subset of loaded databases.	Right-click a database in the Waveform List and choose **Exclude from Tandem** from the popup menu. The icon adjacent to the database name changes to indicate the database is excluded. Choose **Include in Tandem** to include the database again.
Apply tandem mode to a group of databases.	Choose **Edit** > **Groups** > **Group Databases** and use the dialog box to create groups and assign databases to groups. For more information, refer to "Managing Database Groups for Tandem Mode" on page 128. Only databases in the same group as the source waveform are considered in tandem mode.

4. Click the **Tandem Mode** icon from the toolbar again when you are ready to deactivate tandem mode.

Results

Waveforms with the same name from different databases are plotted in a graph window.

Note

Tandem mode may only be used from within the GUI, and is not supported when using TCL or internal commands.

Related Topics

Waveform List Panel

Selected Waveforms Popup Menu

Manually Updating Waveforms

Plotting Multiple Waveforms

Managing Database Groups for Tandem Mode

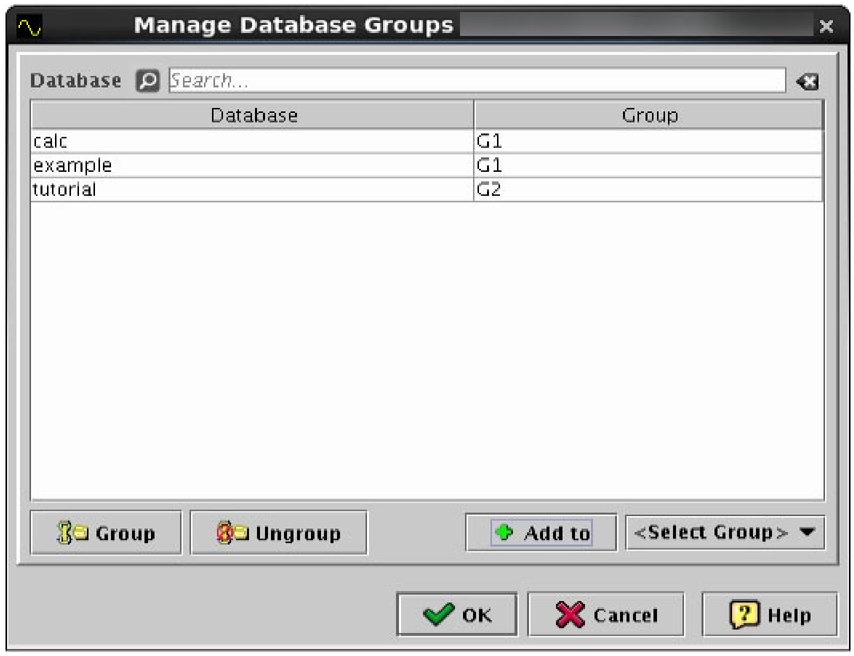
Managing Database Groups for Tandem Mode

You can create and manage groups containing multiple databases for use with tandem mode.

Procedure

1. Choose **Edit** > **Groups** > **Group Databases** from the main menu.

   The Manage Database Groups dialog box opens, listing the currently open databases.



2. To add a database to a new group, select the database from the table and click the **Group** button.

   A new group is created and the database is assigned to that group.

3. To add a database to an existing group, select the database from the table, choose the required group from the Select Group dropdown list and click the **Add to** button. The database is assigned to the selected group.

   The database is assigned to the selected group.

4. To remove a database from a group, select the database from the table and click the **Ungroup** button.

   Tip
   You can use the Database Search field to find databases. This is useful when many databases are open.

5. Click **OK** to finalize the group-database associations and close the dialog box.

   When Tandem Mode is activated, the database group number is displayed adjacent to the each database in the Currently Open Databases list. Only databases in the same group as the source waveform's database are considered in tandem mode.

Related Topics

Plotting Waveforms With the Same Name Using Tandem Mode

Plotting the Difference Between Two Waveforms

You can subtract two waveforms and plot the difference between them.

Procedure

1. Hold down the Ctrl key and select two waveforms from the Waveform List Panel.

   The order you select the waveforms is important—the second waveform selected is subtracted from the first.

2. Right-click and choose **Plot Difference** from the Selected Waveforms popup menu.

Results

The resulting waveform is plotted in a graph window.

Related Topics

Waveform List Panel

Selected Waveforms Popup Menu

Creating an XY Plot

You can plot a waveform as a function of another waveform.

Procedure

1. Plot the desired waveforms in the same graph row.
2. Right-click the waveform that is to be the x axis and choose **Set as X Axis** from the Waveform popup menu.

Results

The resulting XY plot waveform displays in a new graph window.

Related Topics

Waveform Popup Menu

Managing Waveform Groups

You can organize waveforms by creating and using waveform groups, so that they can be manipulated together.

Grouping Waveforms 131
Adding a Waveform to a Group 132
Removing a Waveform From a Group 133
Creating an Empty Waveform Group 133
Ungrouping Waveforms 134

Grouping Waveforms

You can organize waveforms by grouping them together within a graph window so they can be manipulated together.

A group is shown as an expandable label in the graph window, with the waveforms it contains collapsed, in other words hidden from view, beneath it. Modifying properties such as color or line style of a group applies to all waveforms in the group of the appropriate type (analog, digital, or bus).

All waveforms with a compatible domain axis can be grouped, except complex waveforms. Transformed complex waveforms can be grouped, however.

Groups can themselves be included in a group. Subgroups can be expanded and collapsed in the graph window. Bus and record waveforms can also be expanded and collapsed within a group.

When compound waveforms are grouped, each element is treated as an independent waveform within the group.

Restrictions and Limitations

- Expression and measurements cannot be performed directly on a group. They can only be performed on individual waveforms within the group.
- Saving grouped waveforms or a *calc* database containing such groups is not supported in conjunction with any kind of X Range Setup or Data Sampling fields on the Save As dialog box.

Procedure

1. Select the required waveforms in the graph window, and right-click and choose **Groups**. Alternatively, you can press Ctrl-G or choose **Edit** > **Groups** > **Group Waveforms**.
2. Type a name for the group and click **OK**.

By default, the group name is *Group0*. Subsequent groups are named *Group1*, *Group2*, and so on. If you enter a name for the group, for example *My_Group*, then subsequent groups are named *My_Group0*, *My_Group1* and so on.

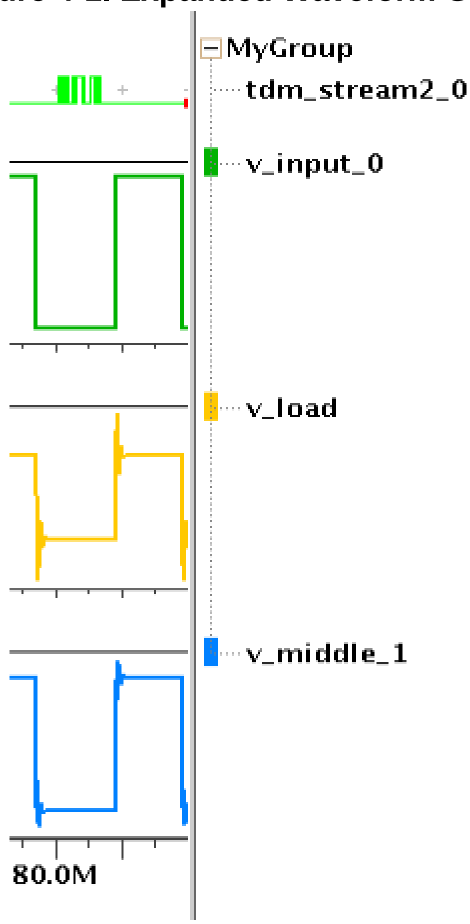
Once a group is created, the waveform labels appear as a hierarchical tree.

Note

When specifying a name for a new group, any existing group of the same name is overwritten without warning.

3. Expand and collapse the tree to show and hide the waveforms within the group, by clicking the + and - icons.

Figure 4-1. Expanded Waveform Group



Adding a Waveform to a Group

You can add a waveform to an existing waveform group.

Procedure

Drag the waveform name from the waveform list, any window, or from within a group onto the destination group's label.

Note

When moving a plotted waveform in or out of a group, or from one group to another, any measurements attached to it are kept in the group. However, multiple waveform measurements such as Intersect or Add Delta are not kept when the related source waveforms are moved into a group.

Results

The waveform is added to the destination from the group.

Removing a Waveform From a Group

You can delete a waveform from a group.

Procedure

Do one of the following:

- Select the waveform's label and press the Delete key.
- Right-click the label and choose **Delete**.

Results

The waveform is removed from the group, and from the graph window.

Creating an Empty Waveform Group

You can create an empty waveform group.

Procedure

1. Clear any current waveform selection.
2. Do one of the following:
    - Choose **Edit** > **Groups** > **Group Waveforms** from the main menu.
    - Use the keyboard shortcut Ctrl-G.
3. (Optional) Modify the suggested group name on the Create Waveform Group dialog box.
4. Click **OK**.

Results

An empty waveform group is created.

Related Topics

add wave

Ungrouping Waveforms

Waveforms can be removed from a waveform group, and the group deleted.

Restrictions and Limitations

- After performing an ungroup action on grouped waveforms, although the group is removed from the graph window, it is not removed from the database.
- There is no Tcl command to ungroup waveforms.

Procedure

Right-click the group and choose **Ungroup Waveforms**.

> **Note**
> It is possible to select multiple waveform groups and ungroup them in a single action, as long as no non-grouped waveforms are selected at the time.

Related Topics

add wave

Rules When Plotting Analog and Digital Waveforms

When analog and digital waveforms are plotted, the order in which the waveforms are selected will dictate whether the plotted waveforms appear together in a single row (overlaid) or in new rows (stacked).

- Digital waveforms cannot be plotted overlaid over other digital waveforms.
- Digital waveforms can be plotted over analog waveforms (overlaid).
- Analog waveforms can be plotted over analog and digital waveforms (overlaid).
- When plotting multiple (analog and digital) waveforms, an analog waveform must be selected first from the Waveform List Panel if both analog and digital waveforms are to be plotted in the same row.
- Digital waveforms are plotted in separate rows until an analog waveform is selected. Subsequent waveforms are plotted overlaid on that waveform.

- When using drag and drop to plot or move multiple waveforms, digital waveforms are plotted stacked in separate rows and analog waveforms are plotted overlaid in a single row regardless of the order of selection. To avoid this default behavior, use the Selected Waveforms Popup Menu to specify plotting overlaid or stacked.

Related Topics

Plotting Multiple Waveforms

Modifying Axis Settings

You can adjust the scale and alignment settings for the x and y axes.

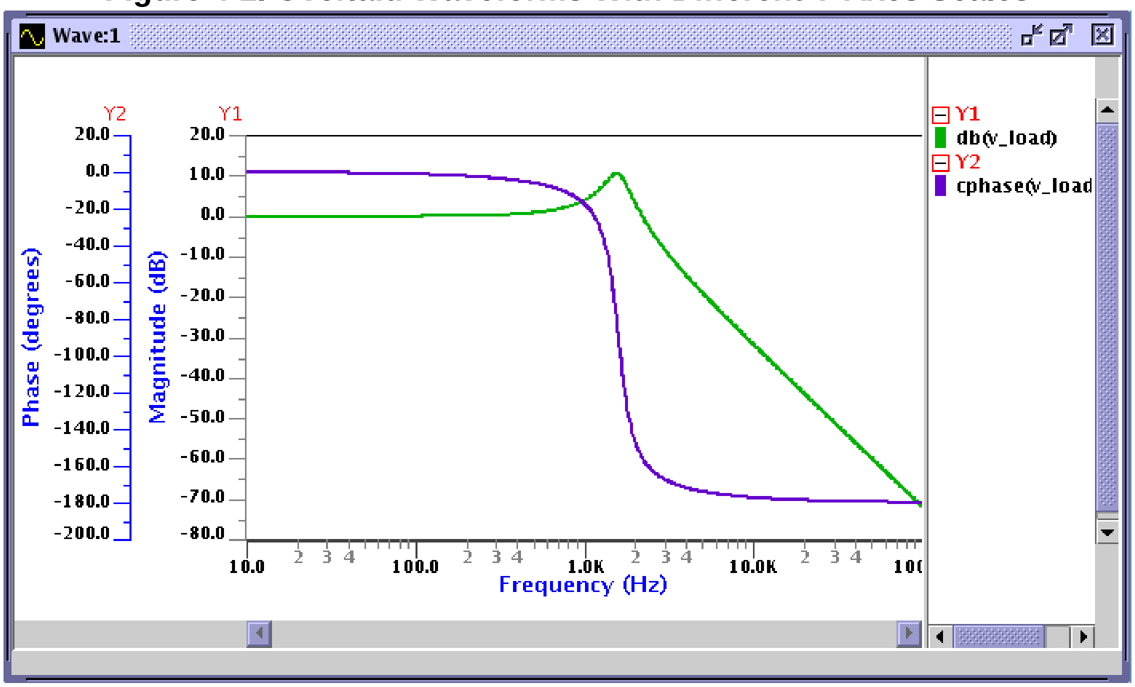
Aligning Y Axes With Different Scales 136
Creating Multiple Y Axes 137
Changing Y-Axis Settings 138

Aligning Y Axes With Different Scales

Overlaid waveforms may have different y axes with different scales. You can specify the axes' marker spacing and how the axes are aligned.

For example, the following figure shows two waveforms overlaid with different y axes. The y axes are lined up at y=20, but have different scales and so do not line up elsewhere.

Figure 4-2. Overlaid Waveforms With Different Y-Axes Scales



Procedure

1. In a graph window showing two or more y axes, right-click the y axis that is to be set as the reference axis and choose **Set As Reference Axis** to open the Set Reference Y Axis dialog box.

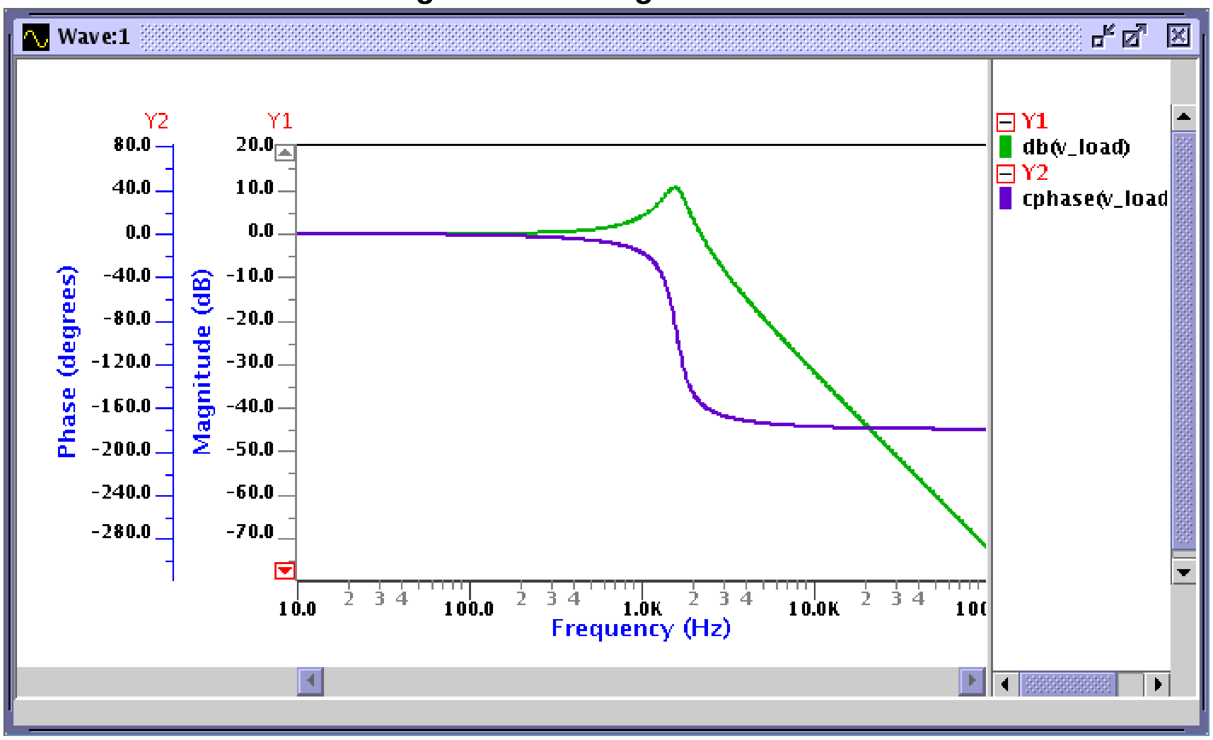
2. In the Reference Y Axis section, in the field labeled with the y axis that is to be the reference y axis specify the value that the other axis will align to.

3. In the Reference Y Axis section, in the Spacing field specify the value for the spacing of the reference axis.
4. In the Y Axis section, specify the alignment value and spacing for the non-reference axis (or axes).
5. Click **OK** to apply the changes and close the dialog box.

Results

In this example, the y axes are realigned to 0 and the spacing of axis Y2 is changed to 40.

Figure 4-3. Realigned Y Axes



To reset the axes to their original scaling and alignment, right-click the reference axis, choose **Unset Reference Axis**, and click the **View All** icon from the toolbar.

Related Topics

wave yaxis

Creating Multiple Y Axes

Plotting complex waveforms and using overlaid transformations will automatically create multiple y axes, as different types of results use different units (for example, db or phase). For regular waveforms, you can create additional y axes by specifying the -newyaxis argument on the add wave Tcl command.

Procedure

Use the following command to create a single row with two axes named Y1 and Y2:

```
add wave -overlay -terminals tvin -newyaxis tvbg
```

The first axis is created by default, and is named Y1. A second axis is created and is named Y2, according to the default naming convention for new axes, Y(n+1).

Related Topics

add wave

Changing Y-Axis Settings

You can change the y-axis settings using the Tcl command wave yaxis.

Procedure

1. When a row contains a single y axis, you can use the wave yaxis command to set the scale of the axis in the first row in the active graph window. For example, to set the scale of the axis to log2, use the following:

   ```
   wave yaxis -row 1 -scale log2
   ```

2. When a row contains multiple y axes, any changes to the axis settings affect the first axis in the row, Y1. To make changes to a different axis in the row, use the -axis argument to specify it. For example, to set the scale of the axis Y2, located in the second row, to log10, use the following:

   ```
   wave yaxis -row 2 -axis Y2 -scale log10
   ```

   Note

   Plotting complex waveforms and using overlaid transformations will automatically create multiple y axes, as different types of results will use different units (for example, db or phase).

   The first axis, created by default, is named Y1. Subsequent axes are named according to the default naming convention, Y(n+1).

Related Topics

add wave

wave yaxis

Modifying Waveforms

You can modify existing waveforms in various ways once they are plotted in the graph window.

Hiding Waveforms 139
Unhiding Hidden Waveforms 139
Modifying Waveform Properties 140
Modifying the Appearance of Row Titles 141

Hiding Waveforms

When several waveforms are displayed in a graph window, you can temporarily hide some of them from view to make the graph window easier to interpret.

Procedure

Select the waveform in the active graph window, right-click and choose **Hide Waveform** from the popup menu.

Results

The waveform label is displayed in the active graph window but the waveform is hidden. The label is dimmed indicating the hide condition. The **Hide Waveform** action differs from **Edit** > **Delete** as the waveform is still within the window and available for later viewing.

Related Topics

Modifying Waveform Properties

Waveform Popup Menu

Unhiding Hidden Waveforms

You can unhide a hidden waveform.

Procedure

Right-click the label for the hidden waveform and uncheck **Hide Waveform** from the popup menu.

Results

The waveform is re-displayed in the graph window. The label is undimmed.

Related Topics

Modifying Waveform Properties

Waveform Popup Menu

Modifying Waveform Properties

You can modify the appearance, display parameters, and transformations applied to waveforms. The properties available to modify depend on whether you want to modify the properties of one waveform or multiple waveforms, and whether it is analog, digital, or a bus.

Procedure

1. In the graph window, right-click the waveform or the label of the waveform whose properties you want to modify, and choose **Properties**. For multiple waveforms, **Ctrl-click** or **Shift-click** the waveforms whose properties you want to modify, then right-click the label of one of the highlighted waveforms, or right-click one of the highlighted waveforms. This opens the Waveform Properties dialog box (For Analog Waveforms).
2. Do one of the following operations:

If you want to...	Do the following:
For a single selected waveform, modify its name or a visual effect such as color, line style, line width, or the symbol used for data points.	Modify the appropriate property in the **Appearance** tab. The options vary depending on the type of waveform.
For a single selected waveform, apply transformations to it.	Choose the appropriate transformations in the **Transformations** tab. You can choose different kinds of transformations to apply if the waveform is analog.
For multiple analog waveforms, modify their visual effects.	Modify the appropriate properties in the **Analog Waveform Properties** tab.
For multiple digital waveforms, modify their visual effects.	Modify the appropriate properties in the **Digital Waveform Properties** tab.
For multiple waveforms, select the radix used for displaying waveform state values.	In the **Radix Waveform Properties** tab, select the radix used for displaying waveform state values. This tab is only available for digital buses or analog waveforms containing integer data.

3. Click **OK** to close the Waveform Properties dialog box.

Related Topics

Waveform Popup Menu

Waveform Properties Dialog Box (For Analog Waveforms)

Modifying the Appearance of Row Titles

When you add a title to a newly plotted waveform it is formatted according to the global defaults. You can override the appearance of each row title individually.

The global defaults are set in the Row Title Options.

Note

Once custom properties are applied to a row title, it will not be affected by any changes to global row title settings.

Procedure

1. Add a row title to a waveform by right-clicking the waveform header, choosing **Row Title**, and specifying a label in the Row Title dialog box.
2. Right-click the row title to modify and choose **Properties**.
3. Edit the font or color, or choose to hide the title outline as required in the Row Title Properties dialog box, then click **OK** to apply the change.

   Note

   Clicking the **Default** button resets the row title to the global default.

Specialized Plotting

There are particular rules and procedures for plotting specialized kinds of waveforms and specialized kinds of plots.

Plotting Complex-Valued Waveforms **142**
Plotting wreal Waveforms **143**
Plotting Multiple Bit Waveforms as a Bus **143**
Displaying Bus Values as a String **144**
Plotting Assertions **145**
Plotting Histograms **147**
Plotting CDFs **148**

Plotting Complex-Valued Waveforms

Complex-valued waveforms are represented in two dimensions, and so require a transformation to be applied to their data before they can be displayed in a graph window.

Use these transformations:

- To separate the waveform parts:
  - Real and imaginary parts
  - Gain (dB) and phase parts
  - Magnitude and phase parts
  - Any combination or single element of these
- To plot value pairs of complex-values against each other in charts:
  - Complex plane plots
  - Polar charts
  - Smith charts

This enables the waveform to be displayed in a variety of ways, showing the information required about the individual points.

Related Topics

Plotting a Single Waveform

Plotting Multiple Waveforms

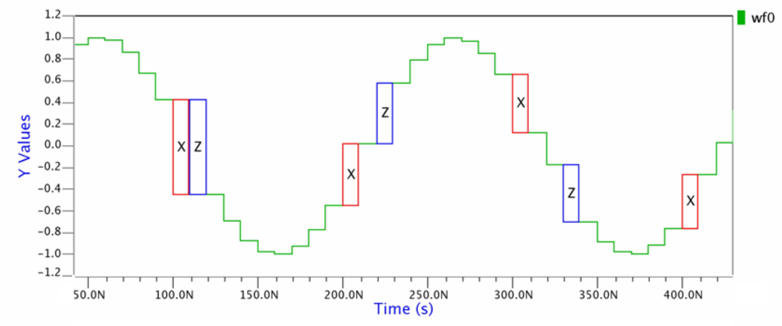
Transformation Options

Plotting wreal Waveforms

You can display wreal waveforms as step waveforms or railroad waveforms.

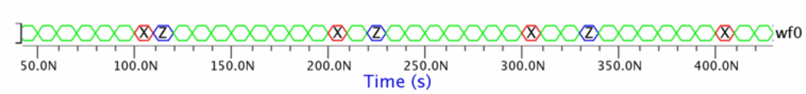
- **Step Waveforms —** Here wreal waveforms are displayed similar to real waveforms. With step waveforms the "X" and "Z" states are displayed as colored rectangles. By default, EZwave displays wreal waveforms as step waveforms.

Figure 4-4. Step Waveform



- **Railroad Waveforms —** Here wreal waveforms are displayed as a railroad waveform where the "X" and "Z" states as displayed as colored circles along a timeline.

Figure 4-5. Railroad Waveform



You can control how wreal waveforms are displayed using the Waveform Options.

Related Topics

Specialized Plotting

Plotting Multiple Bit Waveforms as a Bus

You can create a bus from multiple digital waveforms (also known as bits).

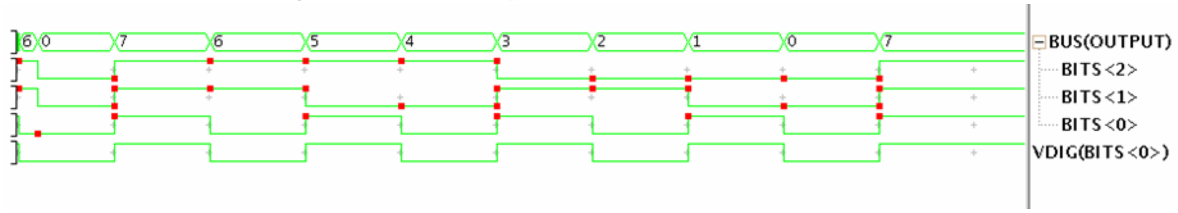
Procedure

1. In the Waveform List Panel, select the waveforms to plot as a bus.
2. Right-click and choose **Plot as Bus** from the Selected Waveforms popup menu.
3. Click the plus sign (+) next to the bus name, to show the individual digital waveforms. Red squares indicate when there is an 'event' on a waveform but it does not change state.

Note

MSB to LSB order is determined through the order of selection before choosing **Plot as Bus**. The first selected waveform is MSB and the last is LSB.

Figure 4-6. Multiple Bit Waveforms as a Bus



Note

When creating a bus using **Plot as Bus** and a conversion from analog to digital is needed (at least one of the bits is analog), a default threshold is calculated as *(min_y + max_y)/2* from the first analog waveform that is not flat *(min_y == max_y)* by iterating from MSB to LSB. Alternatively, a single threshold, or two thresholds, may be specified.

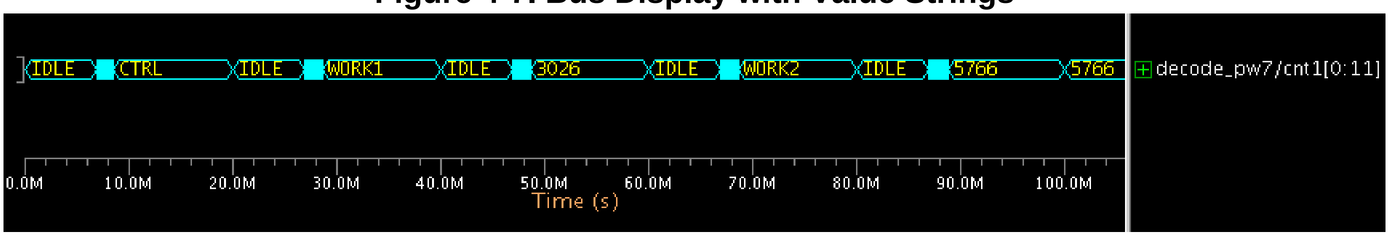
Related Topics

Transformation Options

Displaying Bus Values as a String

You can display bus values as strings, by setting up a file with mappings between the values and strings.

Figure 4-7. Bus Display with Value Strings



Prerequisites

There must be a file with a *.tcl* extension containing a Tcl command radix define specifying the strings that are displayed in the place of specific numerical values in the bus. For example, a radix named States might be defined as follows:

radix define States {

12'b000000000000, IDLE

12'b111000000110, CTRL

12'b000100100110, WORK1

12'b110000110110, WORK2

}

Procedure

1. Open the radix definition *.tcl* file by choosing **File** > **Open**.
2. Select the required waveforms, right-click and choose **Plot as Bus**.
3. Right-click the plotted bus and choose **Radix** from the Waveform popup menu.
4. Select the name of the radix defined in the Tcl file from the options available.

Related Topics

Working With a Bus

Tcl Scripting Support

Plotting Assertions

This topic describes the assertions you can display.

- VHDL-AMS assertions
- SPICE Safe Operating Area (SOA) assertions
- Condition Coverage for VHDL-AMS if-use and case-use.

These are displayed as a waveform group.

Note

Assertions and SOA are supported for display purposes only. They are not supported in post-processing or waveform comparison.

The following table summarizes the graphic elements used in the graph window for assertions:

Table 4-1. Graphic Elements for Assertions

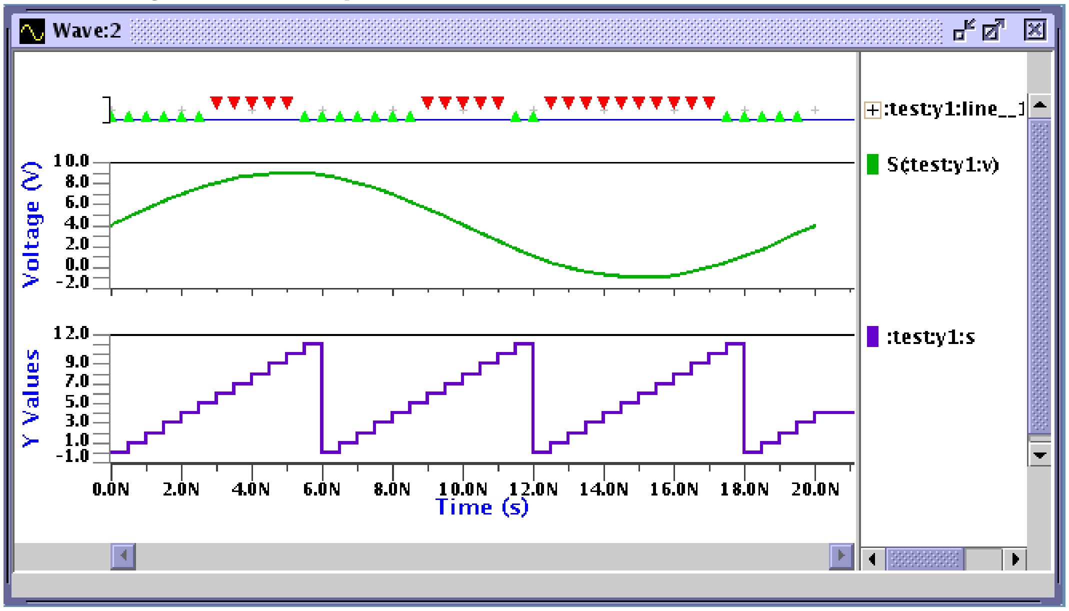
Graphic Element	Meaning
Blue line	Assertion is inactive
Green line	Assertion is active
Red line[1]	Assertion failed
Blue square	Assertion starts
Green triangle	Assertion passed
Red triangle	Assertion failed
Red triangle with horizontal red line below[1]	Assertion failed, crossed the upper boundary

Table 4-1. Graphic Elements for Assertions (cont.)

Graphic Element	Meaning
Inverted red triangle with horizontal red line above[1]	Assertion failed, crossed the lower boundary
Yellow triangle	Antecedent match occurred in assertion

1. Applies only to SPICE SOA assertions.

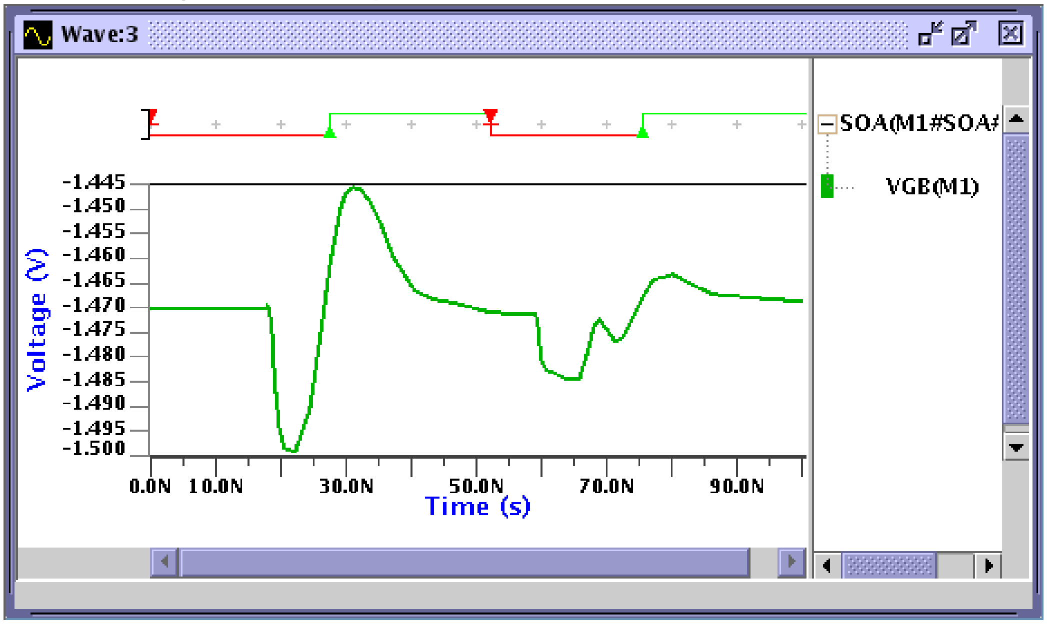
Figure 4-8. Graph Window With a VHDL-AMS Assertion



For SPICE SOA assertions, a blue line means that the SOA check is off, or inactive. Green and red lines indicate that the SOA checks are being made. Green means that the check is passing. Red means it is failing.

The triangles, pointing either up or down, indicate if the target waveform is passing up into a fail-or-pass region or down into a fail-or-pass region. Green indicates when the assertion is becoming true, and red indicates when it is becoming false.

Figure 4-9. Graph Window With an SOA Assertion



Tip

See also: Integration with Questa SIM Verification Methodology in the *Questa ADMS User's Manual*, .CHECKSOA command in the *Eldo Reference Manual*, Plotting Safe Operating Area Limits in the *Eldo User's Manual*.

Plotting Histograms

You can plot waveform data as a histogram, including a number of statistical measures such as mean, the Gaussian curve matching the mean and standard deviation, and the lower and upper bounds. The measures to display are user-defined, as is the number of bins. When plotting a histogram, you can either accept the default histogram preferences and generate the histogram in a single step, or you can configure each histogram as it is plotted.

There are two kinds of histogram:

- General Purpose Histogram

  This histogram is derived from a waveform with additional, user-defined measures.

- Histogram from a Monte Carlo Analysis

  This histogram is based on measures provided by Eldo, which enables an expanded set of measures to be displayed over those available in the general purpose histogram. The default histogram plot settings are ignored, as the measures to display are dictated by the *.wdb* file generated by Eldo.

Tip
You can configure the default appearance of histograms using the Histogram Options.

Procedure

1. Right-click a waveform in the Waveform List Panel, and choose **Plot**.

   The waveform is added to the graph window.

2. Right-click the waveform or waveform name and choose **Plot Histogram**.

   The histogram is plotted, with measures displayed based on the histogram preferences set, or the contents of the *.wdb* file, as applicable.

3. Once you have plotted a waveform, you can edit the options used to configure its display—overriding the default preferences—by right-clicking a histogram and selecting **Histogram**.

   The dialog box that opens offers the same options available in the Histogram Plot Options, but any changes are only applied to the current histogram; the global preferences are unchanged.

Tip
See also .HISTOGRAM in the *Eldo Reference Manual.*

Related Topics

histogram

wave histogram

Histogram Plot Options

Histogram Measures Options

Histogram Legend Options

Histogram Options

Plotting CDFs

You can plot waveform data as a CDF (Cumulative Density Function), including a number of statistical measures such as mean, the Gaussian curve matching the mean and standard deviation, and the lower and upper bounds. When plotting a CDF, you can either accept the default CDF preferences and generate the CDF in a single step, or you can configure each CDF as it is plotted.

There are two kinds of CDF:

- General Purpose CDF

This CDF is derived from a waveform with additional, user-defined measures.

- CDF from a Monte Carlo Analysis

  This CDF is based on measures provided by Eldo, which enables an expanded set of measures to be displayed over those available in the general purpose CDF. The default CDF plot settings are ignored, as the measures to display are dictated by the *.wdb* file generated by Eldo.

Procedure

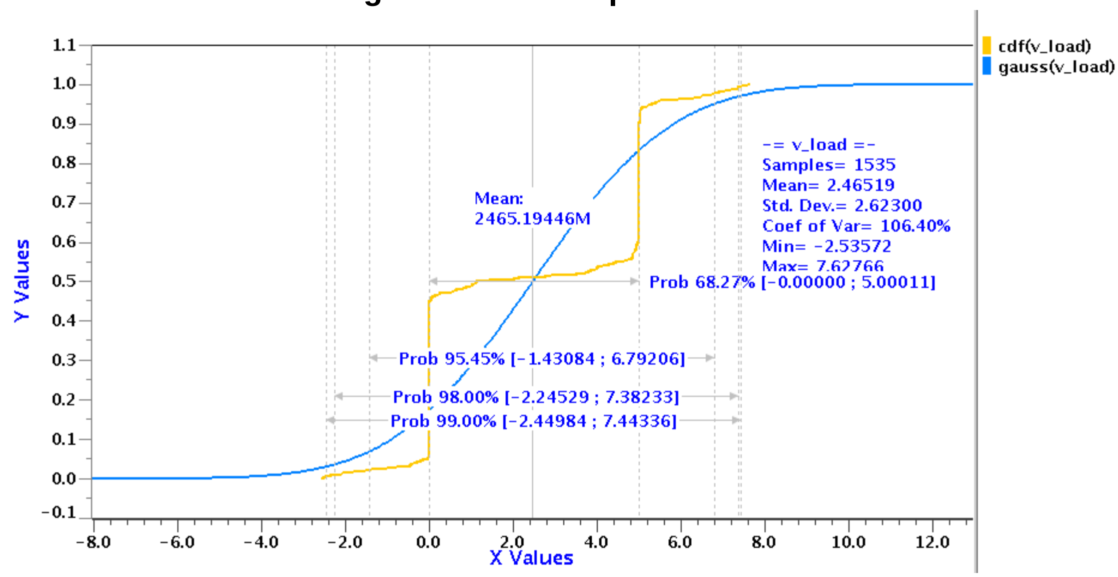
1. Right-click a waveform in the Waveform List Panel, and choose **Plot**.

   The waveform is added to the graph window.

2. Right-click the waveform or waveform name and choose **Plot CDF**.

   The CDF is plotted, with measures displayed based on the CDF preferences set, or the contents of the *.wdb* file, as applicable.

Figure 4-10. Example CDF Plot



3. Once you have plotted a waveform, you can edit the options used to configure its display—overriding the default preferences—by right-clicking a CDF and selecting **CDF**.

   The dialog box that opens offers the same options available in the CDF Plot Options, but any changes are only applied to the current CDF; the global preferences are unchanged.

Tip

See also the Waveform Calculator Function cdf and Tcl command wave cdf, and also .CDF in the *Eldo Reference Manual.*

Related Topics

cdf

wave cdf

CDF Plot Options

CDF Measures Options

CDF Legend Options

Compound Waveforms

A compound waveform is a group of waveforms containing results of several simulations for the same node. They can be generated from multiple-run sweep analyses specified for example by .STEP or .ALTER simulation commands. This section describes the tasks related to compound waveforms.

Displaying Compound Waveforms 151
Displaying Compound Waveforms as Single Elements 152
Merging Waveforms Into a Compound Waveform Using an Index File 154
Digital Compound Waveforms 155
Index File Format 156

Displaying Compound Waveforms

By default compound waveforms are displayed using the same color for all runs. You can expand the display to show the individual element names and run parameters, and show the elements waveforms displayed in different colors.

Note

When multiple runs are displayed as a compound waveform, any measurements made are performed on the entire set of runs. To make measurements on individual runs, you first need to plot the individual elements, as described in "Displaying Compound Waveforms as Single Elements" on page 152.

Procedure

1. Choose **Edit** > **Options**.

   The EZwave Display Preferences dialog box opens.

2. Choose **Multiple Run** from the list on the left-hand side of the EZwave Display Preferences dialog box.

   The Multiple Run Options panel is displayed.

3. Choose **Display as Compound**.

4. To display the names of each element, choose **Expand Single Element Names**.

5. To display the run parameters associated with each component waveform in the graph windows and also in the Waveform List panel (Tree view), choose **Show Single Element Names With Run Parameters**.

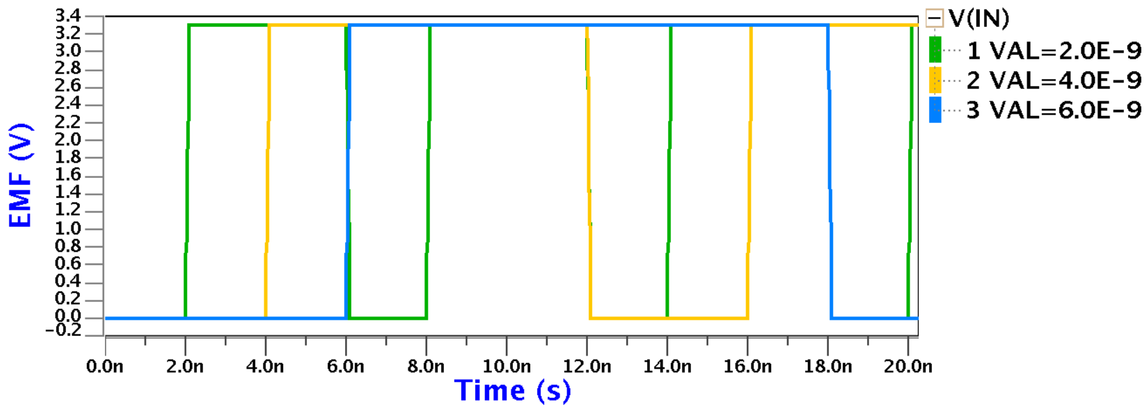
6. To display a single color for all waveforms from the same simulation run, check **Color-Code Each Single Element**.

7. Click **OK** to accept the changes.

Results

When you plot a compound waveform, the Multiple Run preferences you have set are used. An example is shown in the following figure:

Figure 4-11. Expanded Compound Waveform Names With Run Parameters



Tip

You can click the [+] or [-] symbol that appears to the left of the displayed compound waveform name to individually expand or hide the element names.

Related Topics

EZwave Display Preferences Dialog Box

Multiple Run Options

Displaying Compound Waveforms as Single Elements

You can separate the individual elements of a compound waveform. Each element is displayed in a different color.

Note

When multiple runs are displayed as a single elements, you can make measurements on individual runs. To make measurements on the entire compound waveform, you need to plot the runs as a compound waveform. See "Displaying Compound Waveforms" on page 151.

Procedure

1. Choose **Edit** > **Options**.

   The EZwave Display Preferences dialog box opens.

2. Choose **Multiple Run** from the list on the left-hand side of the EZwave Display Preferences dialog box.

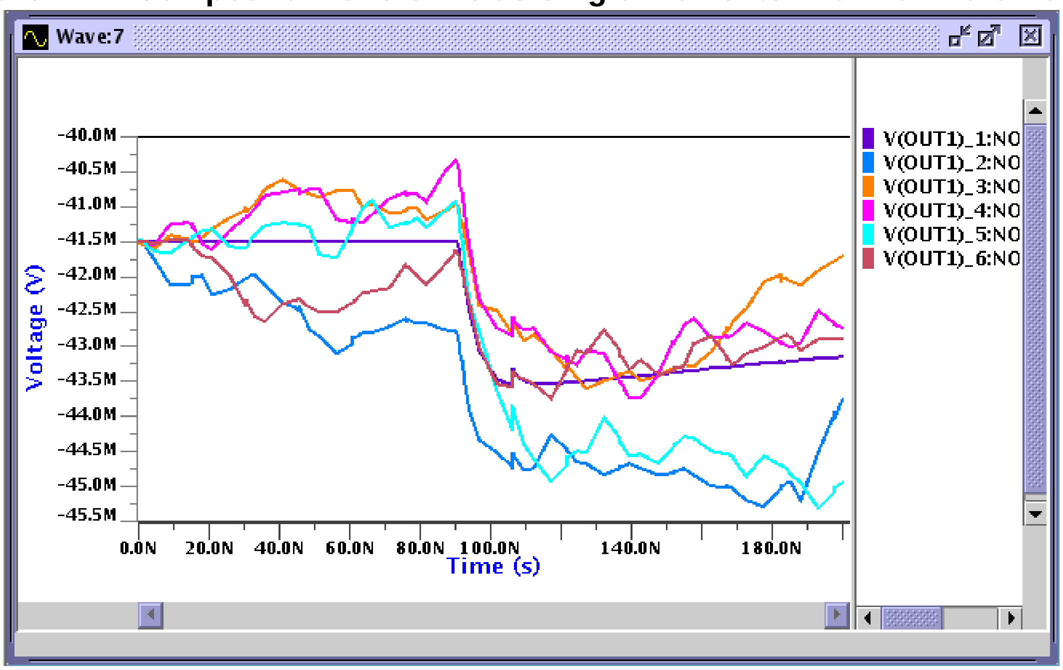
The Multiple Run Options panel is displayed.

3. Choose **Display as Single Elements.**
4. To display the run parameters associated with each component waveform in the graph windows and also in the Waveform List panel (Tree view), choose **Show Single Element Names with Run Parameters**.
5. To display a single color for all waveforms from the same simulation run, check **Color-Code Each Single Element**.
6. Click **OK** to accept the changes.

Results

The elements that make up the compound waveform are displayed with different colors for each element. If you chose to display the run parameters, they are displayed next to the waveform names.

Figure 4-12. Compound Waveforms as Single Elements With Run Parameters



Note

When a multiple run is displayed as single elements, if a measurement is taken on a run, this measurement will not be taken automatically for subsequent simulated runs. Moreover, if a user display is set for a run, it is not applied to subsequent simulated runs.

Related Topics

EZwave Display Preferences Dialog Box

Multiple Run Options

Displaying Compound Waveforms

Merging Waveforms Into a Compound Waveform Using an Index File

When running multiple corner simulations, the output may be saved as individual waveform databases, each one referencing a specific corner. You can use an index file to define both the path to each waveform database file and the parameters used to define the corner. You can then use this file in EZwave to load all of the waveforms as a compound waveform, and to interrogate the parameters in a single interface.

Note

The creation of a compound waveform using Tcl/SWD scripts is supported via the dataset merge command. See the Tcl Scripting Support section for more information.

Prerequisites

- You must have completed all simulation runs, and have access to the waveform database files.
- You must have created an index file; see the Index File Format for full details.
- To inspect the parameters for each corner, the option Compound Waveform Display must be set to Display as Compound in the Multiple Run Options.

Procedure

1. Choose **File** > **Open**, browse to the index file, and click **Open**.

   The waveform data from each file is merged into a compound waveform. The new database is named "merge_" and suffixed with an incrementing number. The new database is shown in the Waveform List panel.

   Note

   The new waveform database is created in an unsaved state.

2. Click the compound waveform database to view the waveforms, and then plot any/all as required.
3. Right-click a waveform name and choose **Parameter Table**. The Parameter Table Dialog Box opens, enabling you to view the parameters used to generate each corner.

Related Topics

Index File Format

Digital Compound Waveforms

The runs of a digital compound waveform may be displayed individually as single elements, or all together in the form of a bus.

Note

When all runs of a digital compound waveform are displayed all together (in the form of a bus), there is the limitation that you cannot right-click the bus to open the Parameter Table to view the parameters. The workaround is to right-click the database or analysis folder and choose **Run Filter** from the popup menu to open the Parameter Table Dialog Box. Set up the filter before plotting the waveform.

Index File Format

Input for: EZwave Compound Waveform feature. Generated by: user.

An Index file is a user-generated file which references multiple waveform databases, each containing the waveform data from a single run of a multiple-run simulation. When loaded into EZwave, it enables all referenced waveforms to be automatically combined into a compound waveform, along with all parameter data.

Index files have the extension *.wdbparam*.

Format

An Index file must conform to the following formatting and syntax rules:

- Each RUN statement must include spaces between the RUN keyword, run index, database path, and parameter definitions.
- RUN statements are separated by carriage returns.
- To use a quotation mark (") or equals sign (=) in the database path or parameter name/value, prefix it with the backslash character (\), as these two characters are used as separators in the index file. If any special character, including a quotation mark or equals sign with a backslash prefix, is used in the database path or parameter name/value, the entire database path or parameter name/value pair should be enclosed between quotation marks.

```
RUN <run_index> ["]<database_path>["] ["]<param_definition>["]
{["]<param_definition>["]}
```

Parameters

- run_index

  Specifies the run order and is used as the waveform label to identify it within the compound waveform data. Numeric.

- database_path

  Specifies the path and filename to the waveform database containing the waveform data for this run. This can be a Windows or Linux path. Database names cannot contain the following characters: /*?":<>|

- param_definition

  Specifies a parameter name (string) and value (string or numeric) pair separated by the equals (=) character and optionally followed by a unit definition in the following form: UNIT=<unit_name>.

Note

FSDB files are not supported in index file (*.wdbparam*) format. You must convert the files to WDB format (*.wdb*) before writing to *.wdbparam* format.

ffcv <file.fsdb> -jwdb <file.wdb>

Examples

```
v1.0
RUN 1 "extract_of_extract.wdb" VTH=10u UNIT=V
RUN 2 extract_of_extract_1.wdb VTH=0.05e3
RUN 3 extract_of_extract_2.wdb  VTH=2
RUN 4 extract_of_extract_3.wdb  VTH=0
RUN 5 extract_of_extract_4.wdb  VTH=3
…
```

Related Topics

Merging Waveforms Into a Compound Waveform Using an Index File

Utility to Convert a Waveform Database [Eldo Platform User's Manual]

Updating Waveform Data

You can update waveform data automatically or manually.

Configuring Automatic Reloading of Waveform Data 158
Manually Updating Waveforms 160
Waveform Update Shortcuts. 160
Updating Waveforms With Options 160

Configuring Automatic Reloading of Waveform Data

When running a series of iterative simulations, EZwave automates the display of waveforms, so that new data is displayed automatically when a simulation is re-run, overwriting the previous database. You can configure certain aspects of this process.

Procedure

1. Choose **Edit** > **Options**.

   The EZwave Display Preferences dialog box opens.

2. Choose **Automatic Reload** from the list on the left-hand side of the EZwave Display Preferences dialog box.

   The Automatic Reload panel is displayed.

3. Set any of the following options:

If you want to...	Do the following:
Have EZwave overwrite the previous results when a new simulation is run.	Choose **Replace Previous Result**. When this option is selected, you can specify whether a warning is reported before overwriting takes place.
Have EZwave preserve the previous results when a new simulation is run.	Choose **Keep Previous Result**, and type the number of previous databases you want to keep. EZwave will automatically save the previous simulation results, and rename each database as: *<filename>_sim1.wdb*, *<filename>_sim2.wdb*, ... *<filename>_sim<n>.wdb*.

If you want to...	Do the following:
Have EZwave ask whether to Keep, Close or Delete previous databases when a new simulation is run. • Keep means that the database file <filename> is renamed to <filename_sim#> and remains open. • Close means that the database file <filename> is renamed to <filename_sim#> and closed. All waveforms from this database are removed from the display, but the database is saved. • Delete means that the database file <filename> is renamed to <filename_sim#>, and is closed. All waveforms from this database are removed from the display and the file is deleted.	Choose **Ask Me**.
Have EZwave display new simulation results on top of the previous simulation results.	Choose **Display New Results Automatically**. This is set by default.
Have EZwave preserve previous results databases when a new simulation is run by just closing them, rather than deleting them once the specified number of previous databases is reached.	Choose **Only Close Old Result** (do not delete file).
Have EZwave preserve the display of all previously loaded waveforms.	Choose **Keep All Waveforms**. This can be useful for waveforms that are not immediately created when simulation starts, for example, extraction waveforms.

Related Topics

Manually Updating Waveforms

Plotting Waveforms With the Same Name Using Tandem Mode

Manually Updating Waveforms

When viewing plotted waveforms from a database in EZwave, you can update the waveform display with the results of another database. Matching waveform data is updated automatically based on names. This is useful when running a series of iterative simulations with differing parameters between runs, where results are stored in separate *.wdb* files.

When updating a database manually, you can control which waveforms to update, and whether to replace the previous waveforms, or to plot the new waveforms separately.

Waveform Update Shortcuts **160**
Updating Waveforms With Options **160**

Waveform Update Shortcuts

You can quickly access to the latest waveform data from subsequent runs of the same simulation.

Procedure

Right-click an open database and select one of the following options:

- **Update Waveforms > Replace Existing with New**

  This finds matches for the waveform names in the database in all open graph windows and replaces those waveforms with the new data. You can also select the new database and use the keyboard shortcut Alt-R.

- **Update Waveforms > Show Existing and New Together**

  This finds matches for the waveform names in the database in all open graph windows and plots the updated waveforms alongside the existing matching waveforms (see Figure 4-13 on page 162). You can also select the new database and use the keyboard shortcut Alt-S.

  Tip

  You can enable tandem mode to automatically plot waveforms with the same name from different databases. See "Plotting Waveforms With the Same Name Using Tandem Mode" on page 126.

Updating Waveforms With Options

You can update waveform data from a new database to take full control of the process.

This is done using the "Update Waveforms Dialog Box" on page 592.

Procedure

1. Open a database and plot one or more waveforms.
2. Open another database containing different results for the same simulation. Right-click this database in the Waveform List and choose **Update Waveforms** > **More Options** (alternatively, press Alt-O).

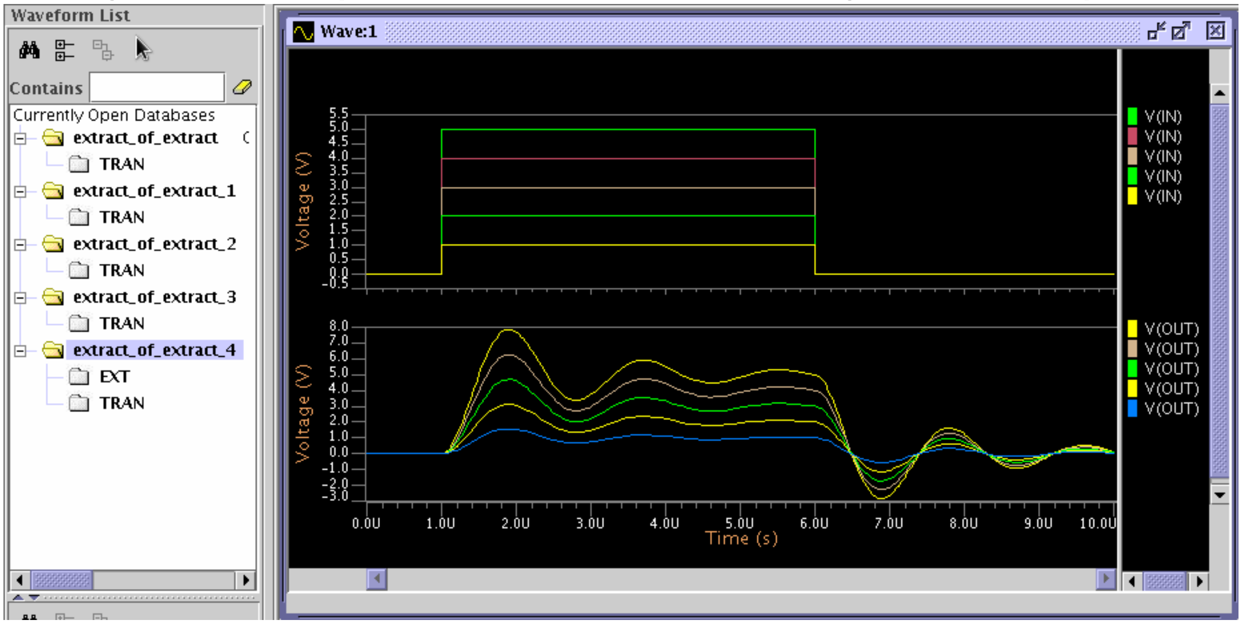
   Tip

   You can update multiple databases in one step by holding down the Shift or Ctrl keys when making selections from the Waveform List. Once all the required databases are highlighted, right-click the multiple selection and choose **Update Waveforms** > **More Options**.

3. The Update Waveforms dialog box is displayed. Choose to either **Replace** the currently plotted waveforms with the matching waveforms from the new database, or to **Show Existing & New Together**, which will plot the new waveforms overlaid.
4. Choose the **Update Target** to define which waveforms to update:
   - All Windows—Waveforms plotted in all open windows are updated.
   - Active Window Only—Only the waveforms plotted in the currently active window are updated.
   - Plotted Waveforms—Only the waveforms added to the **Plotted Waveforms List** are updated. To define specific waveforms in this list, do the following:
     - Click **Add Waveforms** to select which of the currently plotted waveforms to add to the **Plotted Waveforms List**.
     - To remove a waveform from the list, select it and click **Remove Selected**.
     - Use the **Move Up** and **Move Down** buttons to re-order the **Plotted Waveforms List**. This is for organizational purposes only — the order of waveforms in the list has no impact on the update.
5. Click **OK** to apply the update and close the dialog box. Waveforms from the newer database are matched with those currently plotted in the target windows.

   The following figure shows the results from multiple databases plotted overlaid, using the **Show Existing & New Together** menu option.

Figure 4-13. Multiple Overlaid Waveforms Following a Waveform Update



Related Topics

[Automatic Reload Options]

[Update Waveforms Dialog Box]

Chapter 5
Analysis

This section describes how to measure, analyze, and annotate data points or logic units that are represented in the waveforms.

Working with Cursors . . . 165
- Adding the Base Cursor . . . 165
- Changing the Base Cursor . . . 166
- Adding Reference Cursors . . . 167
- Adding Relative Reference Cursors . . . 168
- Adding a Horizontal Cursor . . . 169
- Copying Horizontal Cursors . . . 171
- Moving Cursors . . . 172
- Setting the Visibility of Cursor Values . . . 174
- Working with Y-Level Lines . . . 180
- Using the Cursor Value Table With Compound Waveforms . . . 182
- Changing the Visibility of Graph Window Elements . . . 183

Using Pick Points to Take Measurements . . . 184
- Taking Measurements With Pick Points . . . 184
- Adding Delta Markers With Pick Points . . . 187

Using Text Annotations . . . 189
- Adding Text Annotations . . . 189
- Global Display Options for Text Annotations . . . 192

Using the Event Search Tool . . . 194
- Performing an Event Search . . . 194
- Performing an Expression Event Search . . . 195

Working with Eye Diagrams . . . 197
- Creating an Eye Diagram . . . 197
- Adding Additional Waveforms to Eye Diagrams . . . 203
- Editing an Eye Mask . . . 203
- Eye Diagram Measurement Calculations . . . 206

Working With Smith Charts . . . 219
- Scattering Parameters (S-Parameters) . . . 219
- Creating a Smith Chart . . . 220
- Impedance and Admittance Displays . . . 220
- Smith Chart and Polar Displays . . . 223
- Cursors in the Smith Chart . . . 224
- Circles in the Smith Chart . . . 226
- Changing the Visibility of Elements . . . 228

Comparing Waveforms . . . 230

Support for Different Types of Waveform 231
Using the Waveform Compare Wizard 231
Manually Comparing Waveforms 236
Viewing Waveform Comparison Results 240
Waveform Comparison Reports 244
Viewing and Saving Comparison Rules 246
Saving a Comparison Session 246
Setting Comparison Options 248
Analog Waveform Comparison Algorithm 258
wreal Waveform Comparison 259

Analyzing Waveform Parameters to Generate Pivot Waveforms 261

Analyzing Current Consumption 262

Analyzing Power Consumption 268

Working with Cursors

This section explains how to use cursors, for example, adding and moving cursors.

Adding the Base Cursor **165**
Changing the Base Cursor **166**
Adding Reference Cursors **167**
Adding Relative Reference Cursors **168**
Adding a Horizontal Cursor **169**
Copying Horizontal Cursors **171**
Moving Cursors **172**
Setting the Visibility of Cursor Values **174**
Moving Cursor Value Flags 174
Hiding a Cursor Value for a Single Waveform 174
Hiding a Cursor Value for All Waveforms. 175
Showing Cursor Values for Selected Waveforms 175
Setting Default Cursor Value Visibility When Selecting Waveforms 175
Sharing Cursor Value Visibility Settings 177
Setting Cursor Value Visibility in the Cursor Values Table 178
Displaying Cursor Values in the Reserved Area 178
Working with Y-Level Lines **180**
Adding Y-Level Lines 180
Choosing How Delta-Y is Calculated. 181
Using the Cursor Value Table With Compound Waveforms **182**
Changing the Visibility of Graph Window Elements **183**

Adding the Base Cursor

The first cursor you set on a waveform is known as the base cursor. The base cursor is used as a reference for measurements. Only one base cursor is permitted in any graph window.

Procedure

1. Use one of the following methods to add a vertical cursor:

   - Choose **Cursor** > **Add**.
   - Use the keyboard shortcut **F5**.
   - Right-click in a row and choose **Add Cursor** from the Waveform Popup Menu.
   - Use the wave addcursor Tcl command:

     ```
     wave addcursor [-window window_name] [-time time | X_value]
     [-name cursor_name]
     ```

2. You can move the cursor to the required position by dragging it, or by using the Move cursor buttons on the toolbar.

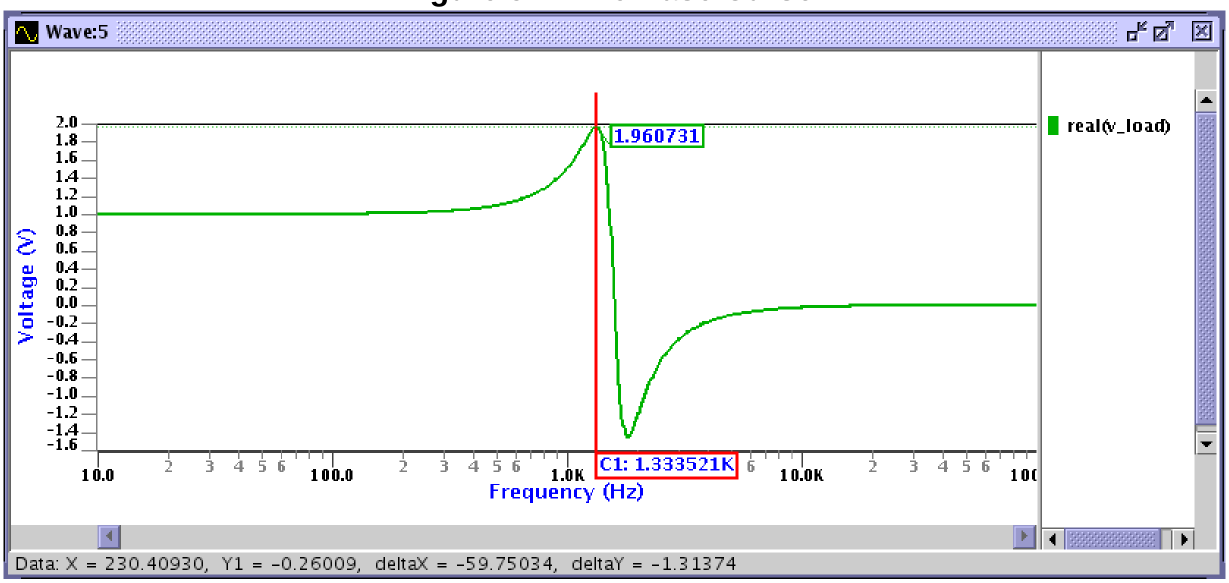
3. If you need more precision, use the snap feature to move the cursor into place. Right-click the cursor and choose **Snap to Data Points** from the Vertical Cursors popup menu. If necessary, zoom in tightly and drag the cursor to see the cursor snap to the closest point.

Tip

Use the **View All** icon on the toolbar to see the entire waveform.

4. A label showing the current X value for the base cursor is displayed beneath the cursor, flush with the x axis. When you move the cursor, by dragging, the X value updates.

Figure 5-1. The Base Cursor



Changing the Base Cursor

When subsequent cursors are added, by default, the first cursor placed is the base cursor. You can change the base cursor.

Procedure

To define another cursor as the base cursor, right-click the desired cursor and choose “**Base Cursor**.

Results

Once this section is made, all delta-X measurements are updated to reflect the new base cursor, and the previous base cursor becomes a regular cursor.

Related Topics

Adding Reference Cursors

Adding Y-Level Lines

Choosing How Delta-Y is Calculated

Adding Reference Cursors

After you have adding an initial base cursor to the waveform, all subsequent cursors are reference cursors. Value boxes for reference cursors are displayed beneath the unit labeling for the x axis and contain the current X value, and a delta-X value, which is the difference between the current X value and the X value of the base cursor.

Procedure

1. Add two vertical cursors to the graph window using one of the following methods:
   - Choose **Cursor** > **Add**.
   - Use the keyboard shortcut **F5**.
   - Right-click in a row and select **Add Cursor** from the Waveform Popup Menu.
   - Use the wave addcursor Tcl command:

     ```
     wave addcursor [-window window_name] [-time time | X_value]
     [-name cursor_name]
     ```

   The first is the base cursor C1, and the second a reference cursor, C2.

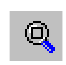
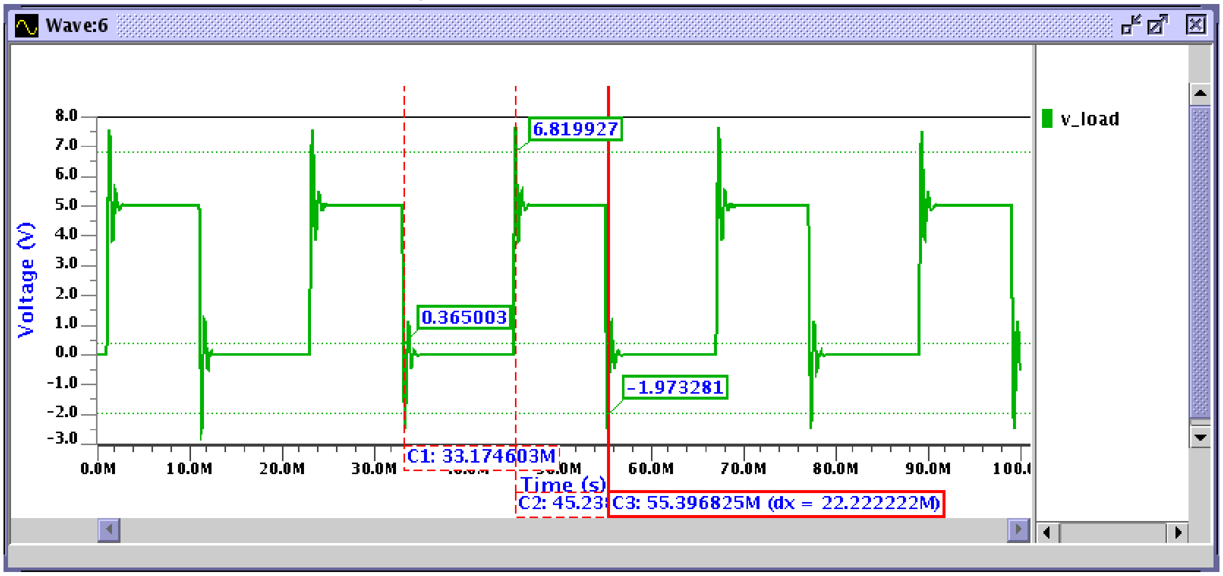
2. With the new cursor in position, use the **View All** icon  to see the entire waveform with the two cursors.
3. Right-click the reference cursor C2 and select **Data Values** > **Show on Left** from the Vertical Cursors popup menu. This will flip the values to the other side, so the axis area can be seen clearly.
4. Now add further cursors to the waveform. This cursor will appear as C3, C4, C5... and so on.

Figure 5-2. Reference Cursors



5. When a new cursor is added, it can be accessed from the **Cursor** menu. This can be useful to quickly view a cursor that is not currently in the visible region of the active graph window.

Adding Relative Reference Cursors

You can place a cursor relative to the current cursor. This is particularly useful for quickly verifying the duty cycle occurrence on some waveforms.

Use the following procedure to add one or more reference cursors relative to an existing cursor.

Procedure

1. Right-click the waveform, select **Add Cursor Relative to** > and choose the required cursor. Alternatively, select **Cursor** > **Add Cursor Relative to** > from the main menu.
2. Type one or more X values separated by spaces in the Enter Relative X/Y Location(s) field. These values define the location of new cursors, relative to the chosen cursor.
3. Select Lock Together When Dragging to specify that all cursors are locked together so that when you move one cursor the others are dragged the same distance. This option is equivalent to choosing **Cursor** > **Lock Together When Dragging** from the main menu.
4. Click **Apply**. The new cursors appear on the waveform.

Related Topics

Adding the Base Cursor

Adding Reference Cursors

Moving Cursors

Adding a Horizontal Cursor

Adding a Horizontal Cursor

You can add an x-axis cursor, that behaves in the same way as the y-axis cursor, but has a horizontal display.

Note

The horizontal cursor is available only for analog or step waveforms, excluding wreal, digital, digital bus and digital bits. You cannot add a horizontal cursor to Smith charts, Polar charts or eye diagrams.

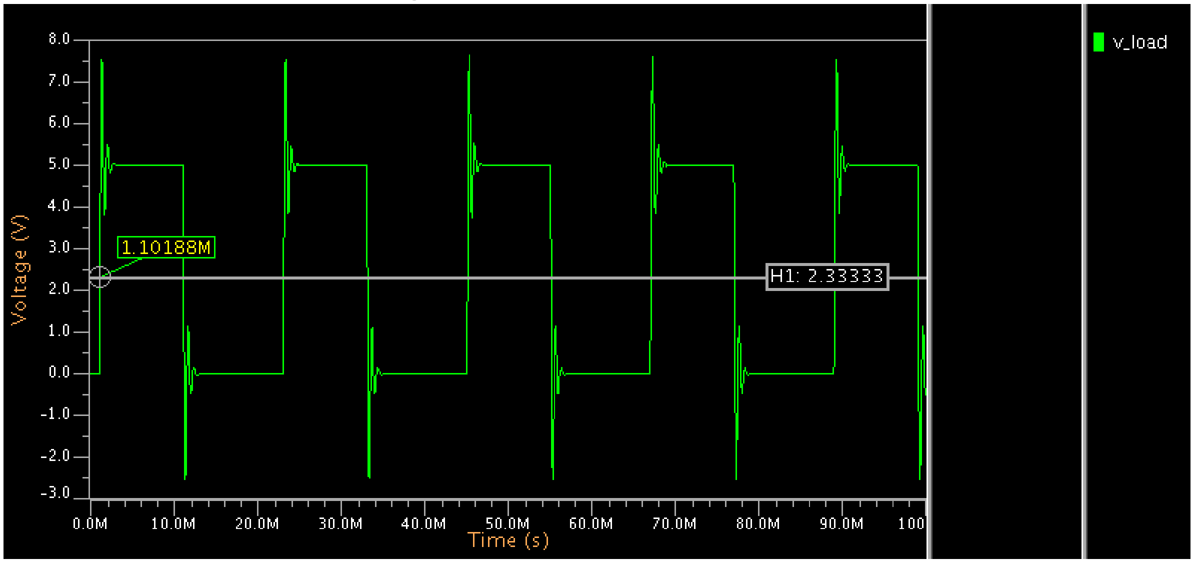
Procedure

1. Horizontal cursors may be added to graph windows using any of the following methods:
   - Choose **Cursor** > **Add Horizontal Cursor**.
   - Use the keyboard shortcut **F6**.
   - Right-click a row, or the y axis, and select **Add Horizontal Cursor**.

     In this case, the horizontal cursor is created at the Y value point where the right-click occurred.
   - Use the wave addcursor Tcl command:

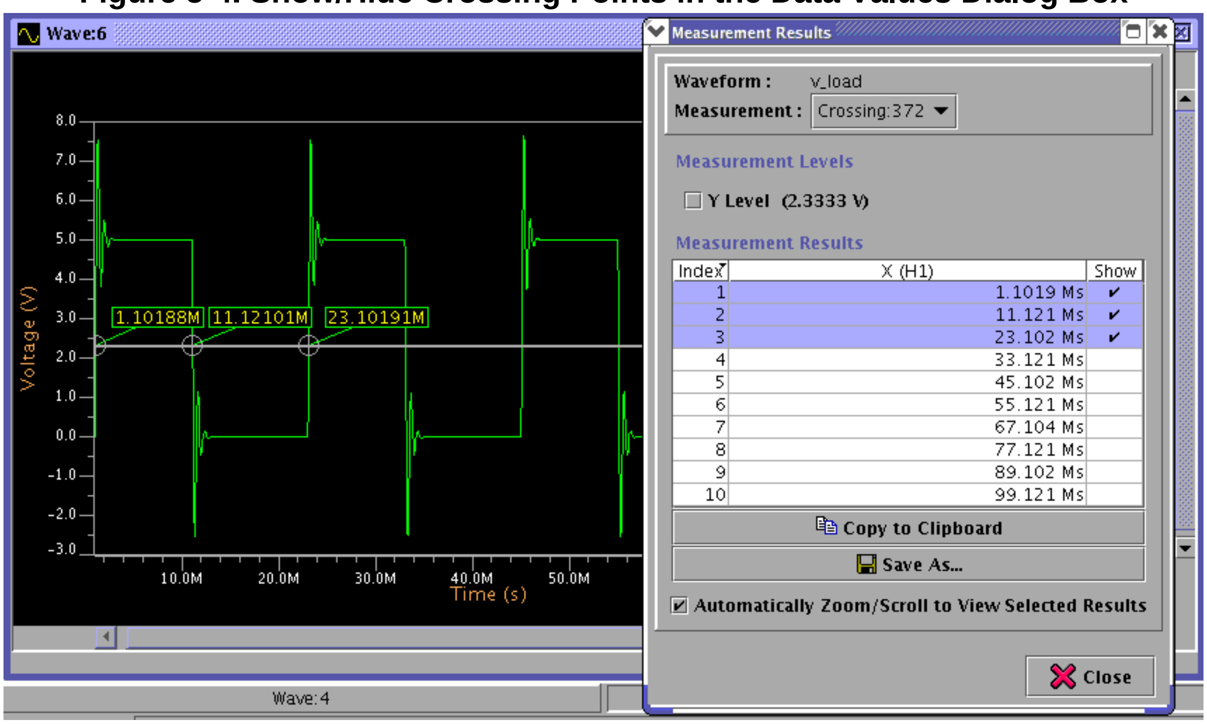
     ```
     wave addcursor [-window window_name] -horizontal [-row row_index]
     [-axis axis_name] [-name cursor_name]
     ```
2. When a horizontal cursor is added, the first crossing point with the waveform is shown, with its measurement. When the cursor is dragged up and down, the Y value and the value at the crossing point will update.

Figure 5-3. Horizontal Cursor



3. To change the number of crossing points that are displayed, right-click the cursor, select **Data Values** and select the waveform name. In the Data Values Dialog Box, use **Shift + click** to add or remove ticks in the **Show** column to show or hide the crossing points, as shown in the following figure:

Figure 5-4. Show/Hide Crossing Points in the Data Values Dialog Box



4. You can move crossing points to a different location by selecting them with the left mouse button and dragging them to a different place on the waveform.

Note

If you change the unit of a degree/radian/gradient y axis to be different from the Trigonometric Angle defined in **Edit** > **Options** > **Waveform Calculator** > **General**, the horizontal cursor crossing values and position are incorrect. If you need another Y unit type, be sure to align the trigonometric angle value accordingly.

Copying Horizontal Cursors

You can duplicate an existing horizontal cursor across all unit-compatible waveform rows in the display.

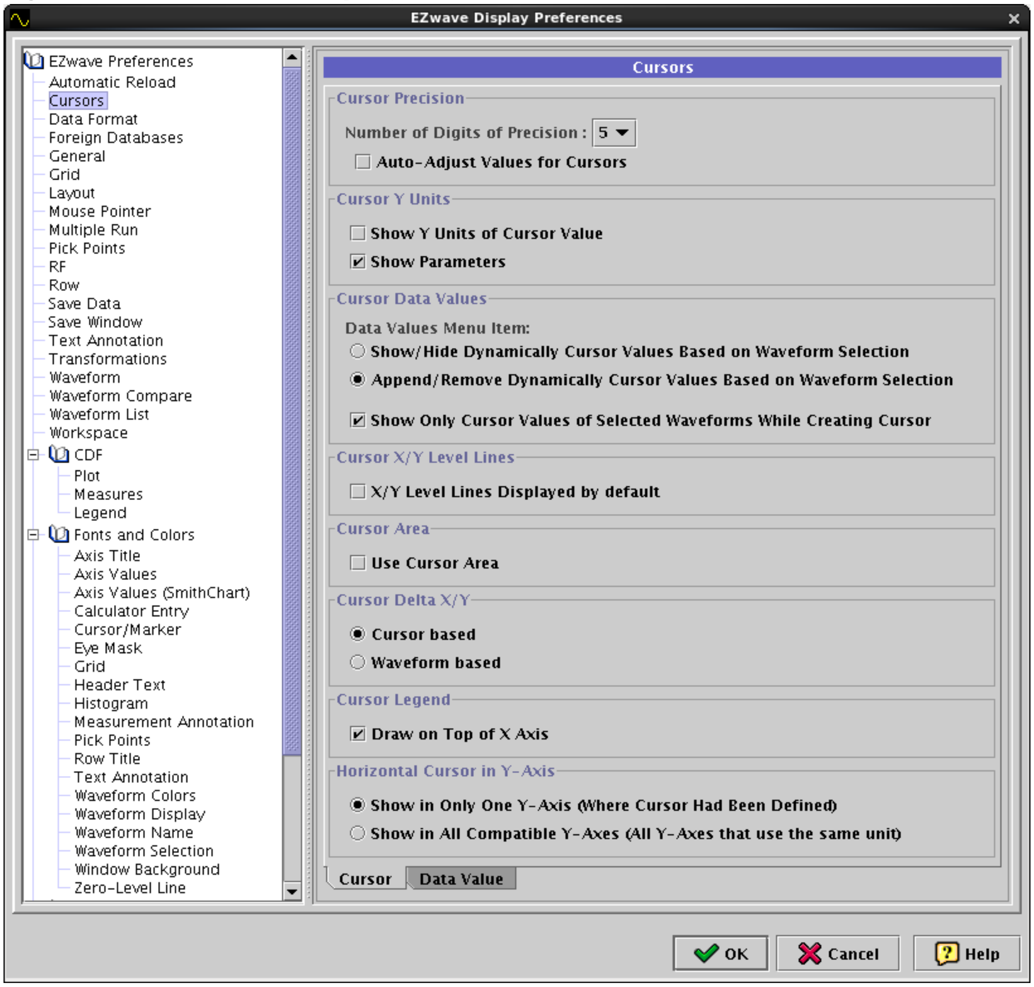
Procedure

1. Right-click the horizontal cursor and select Show in All Compatible Y Axes.

   The horizontal cursor is copied to all waveform rows that use the same y-axis units. When the horizontal cursor is moved, all duplicated cursors also move to the new position. Crossing point markers and measurements will update accordingly

2. The Horizontal Cursor in Y Axis setting under **Edit** > **Options** > **Cursors** enables you to control whether new horizontal cursors are added to the currently selected waveform row only, or to every row in the current display.

Figure 5-5. EZwave Display Preferences - Horizontal Cursor in Y Axis



Related Topics

Adding the Base Cursor

Adding Reference Cursors

Moving Cursors

Moving Cursors

You can move cursors by specifying a value, or by dragging and dropping a cursor to a new location. The corresponding value will update accordingly.

Use the following procedure to move a cursor to a specific point on the waveform:

Procedure

1. Right-click the cursor and select **Move To**. The Move Cursor dialog box displays.

2. Type a new X or Y location (or locations, separated by spaces) in the Enter New X/Y Location(s) field.
3. Select the unit of measurement from the pulldown menu. Choose from hertz or femto-, pico-, nano-, micro-, milli-, kilo-, mega-, giga-, or terahertz.
4. If more than one cursor is added, the first entered becomes the active cursor.
5. Click **OK** and the cursor moves to the specified location.

Tip

Cursors can be locked together so that when you move one cursor, the other is moved the same distance, relative to it. To lock cursors together, choose **Cursor** > **Lock Together When Dragging** from the main menu.

Related Topics

Adding the Base Cursor

Adding Reference Cursors

Adding a Horizontal Cursor

Setting the Visibility of Cursor Values

When more than one cursor is displayed in a graph window it is often useful to temporarily move or hide some of the cursor values from view to make the graph window easier to read.

Moving Cursor Value Flags 174
Hiding a Cursor Value for a Single Waveform 174
Hiding a Cursor Value for All Waveforms 175
Showing Cursor Values for Selected Waveforms 175
Setting Default Cursor Value Visibility When Selecting Waveforms 175
Sharing Cursor Value Visibility Settings 177
Setting Cursor Value Visibility in the Cursor Values Table 178
Displaying Cursor Values in the Reserved Area 178

Moving Cursor Value Flags

You can move the positions of vertical cursor value flags freely. The relative position of the value flag is maintained if a cursor is moved.

Note

If the locations of cursor value flags are modified and the window is saved to TCL/SWD, the positions of the flags are not saved. Flags will return to their default (automatic) positions after a reload.

Procedure

1. Drag the value flag (the box on the cursor that displays the waveform value at that point) to the required location.

   If you move the cursor, the relative position of the value flag is maintained (as much as possible within the confines of the plot window).

   Tip

   If a cursor has multiple flags (for example with overlaid plots), you can select and move all of its flags at the same time. Hold down the Ctrl key and use the left mouse button to select the required flags. Then drag one of the flags and the other selected flags will also move.

2. To return the value flag to its default position, right-click the cursor and choose **Data Values** > **Reset Location**.

Hiding a Cursor Value for a Single Waveform

You can hide the cursor values for a single waveform.

Procedure

1. Right-click the specific Value Flag (the box on the cursor that displays the waveform value at that point) on the cursor you wish to hide.
2. Select **Hide Value** from the Cursor Value Popup Menu.

> **Note**
> Other Value Flags on different waveforms using the same cursor do not disappear.

Hiding a Cursor Value for All Waveforms

You can hide cursor values for more than one waveform in a graph window.

Procedure

1. Select the cursor whose values are to be hidden.
2. Right mouse-click the cursor and select **Data Values** > **Hide All**.

   The Value Flags disappear from the selected cursor for all waveforms displayed.
3. Repeat these steps to make the cursors visible again, selecting **Data Values** > **Show All**.

Showing Cursor Values for Selected Waveforms

You can control the visibility of cursor values for selected or unselected waveforms in the graph window.

Procedure

1. Select one or more waveforms in the graph window.
2. Right-click a cursor and choose **Data Values** > **Show Selected Only Set from Waveform Selection**.

Results

Only the cursor values for the selected waveforms will be shown on the cursor.

Setting Default Cursor Value Visibility When Selecting Waveforms

You can change the default visibility settings for cursor values when selecting waveforms.

Procedure

1. Choose **Edit** > **Options**.

   The EZwave Display Preferences dialog box opens.

2. Choose **Cursors** from the list on the left-hand side of the EZwave Display Preferences dialog box.

   The Cursors panel is displayed.

3. Locate the Cursor Data Values section and choose one of the following actions:

If you want to...	Do the following:
Configure a cursor to show cursor values only for waveforms that are selected.	1. Check the “Show/Hide Dynamically Cursor Values Based on Waveform Selection” box. 2. Click OK. 3. Select the cursor in the graph window. 4. Right-click and choose Data Values. 5. Check Show Selected Mode. 6. In the graph window select one or more waveforms. The cursor value for those waveform are shown.
Configure a cursor to show cursor values only for the waveforms that are not selected.	1. Check the “Show/Hide Dynamically Cursor Values Based on Waveform Selection” box. 2. Click OK. 3. Select the cursor in the graph window. 4. Right-click and choose Data Values. 5. Check Hide Selected Mode. 6. In the graph window select one or more waveforms. The cursor value for those waveform are hidden.
Configure a cursor so that when a waveform is selected it is automatically added to the set of values shown at that cursor.	1. Check the “Append/Remove Dynamically Cursor Values Based on Waveform Selection” box. 2. Click OK. 3. Select the cursor in the graph window. 4. Right-click and choose Data Values. 5. Check Append Selected Mode. 6. In the graph window select a waveform. A cursor value for that waveform is added to the set of values at that cursor.

If you want to...	Do the following:
Configure a cursor so that when a waveform is selected it is automatically removed from the set of values shown at that cursor.	1. Check the “Append/Remove Dynamically Cursor Values Based on Waveform Selection” box. 2. Click OK. 3. Select the cursor in the graph window. 4. Right-click and choose Data Values. 5. Check Remove Selected Mode. 6. In the graph window select a waveform. The cursor value for that waveform is removed from the set of values at that cursor.
Configure cursor creation so that when a new cursor is created, only the values of selected waveforms are shown, or all the values if no waveforms are selected.	1. Check the Show Only Cursor Values Of Selected Waveforms While Creating Cursor box. 2. Click OK. 3. Select one or more waveforms on the Graph Window. 4. Choose **Cursor** > **Add** from the main menu or press F5. A new cursor is created in the graph window. Only the cursor values of selected waveforms are shown on the cursor. If no waveform is selected, all waveforms are considered to be selected. You cannot add a cursor without displaying a waveform value.

Related Topics

EZwave Display Preferences Dialog Box

Cursor Options

Sharing Cursor Value Visibility Settings

You can configure several cursors to share the same visibility settings for cursor values.

Procedure

1. From the main menu, check the **Cursor** > **Share Config of Active Cursor box**.
2. Choose **Cursor** > **Add** from the main menu or press F5. A new cursor is created in the Graph Window.

Results

The new cursor and the previously active cursor now share the same cursor value visibility settings. If the settings for one cursor are changed, the settings for the other cursor change too. To stop sharing, right-click the cursor and choose **Data Values** > **Stop Sharing Configuration**.

Setting Cursor Value Visibility in the Cursor Values Table

You can set the visibility of cursor values for particular waveforms on a selected cursor using the Cursor Values table.

Procedure

1. Right-click the selected cursor and choose **Data Values** > **Cursor Values Table**. The Cursor Values Table opens showing a list of all the waveforms.
2. In the Show column, check the waveforms you want to show, or uncheck the waveforms you want to hide.

   Or click **Show All** to show all waveforms.

   Or click **Hide All** to hide all waveforms.

Results

The cursor values for the selected waveforms are shown on the cursor in the Graph Window.

Displaying Cursor Values in the Reserved Area

You can display cursor values in a reserved area near the **Waveform Names Area** rather than in cursor data flags.

Procedure

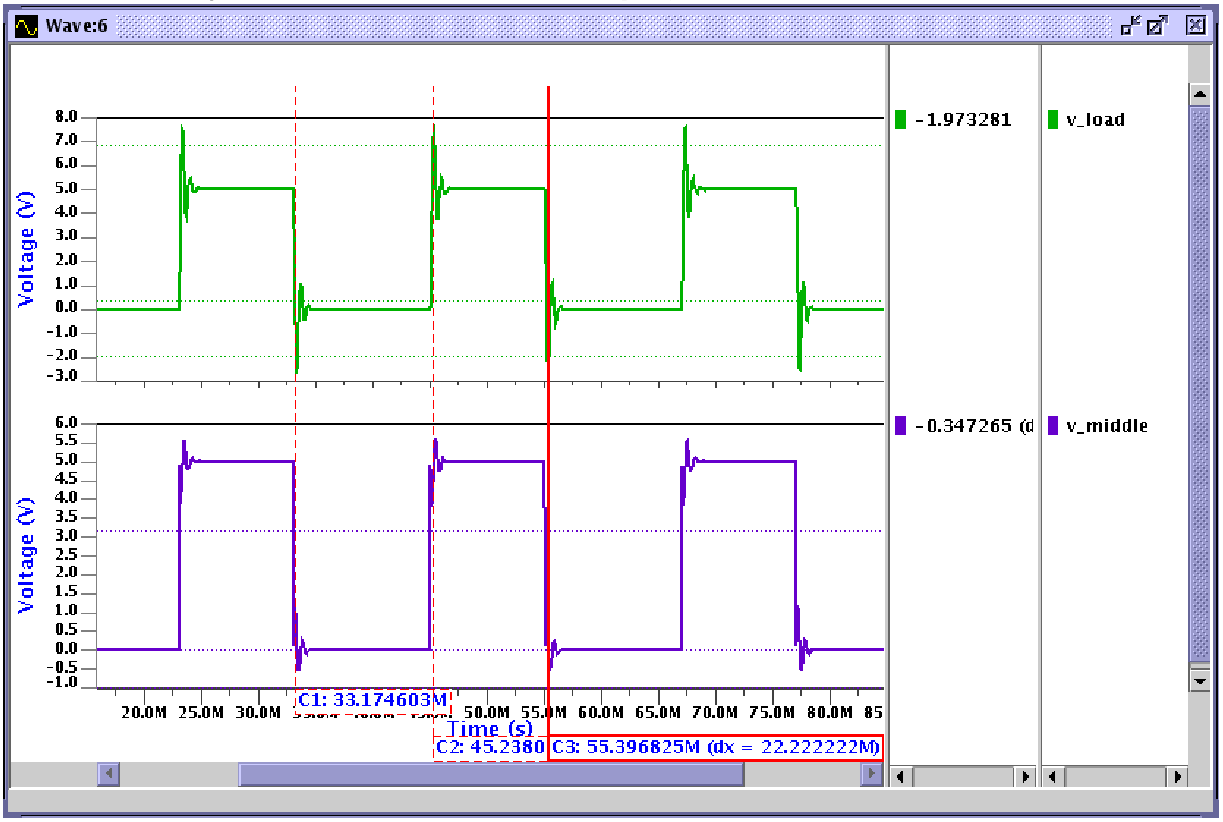
1. Choose **Edit** > **Options**.

   The EZwave Display Preferences dialog box opens.
2. Choose **Cursors** from the list on the left-hand side of the EZwave Display Preferences dialog box.

   The Cursors panel is displayed.
3. Check the Use Cursor Area option.

   The cursor values are displayed in the reserved area.

Figure 5-6. Cursor Values Displayed in the Reserved Area



Related Topics

EZwave Display Preferences Dialog Box

Cursor Options

Adding the Base Cursor

Adding Reference Cursors

Adding a Horizontal Cursor

Working with Y-Level Lines

You can add a Y-level line and measure the difference between Y-level lines.

Adding Y-Level Lines 180
Choosing How Delta-Y is Calculated 181

Adding Y-Level Lines

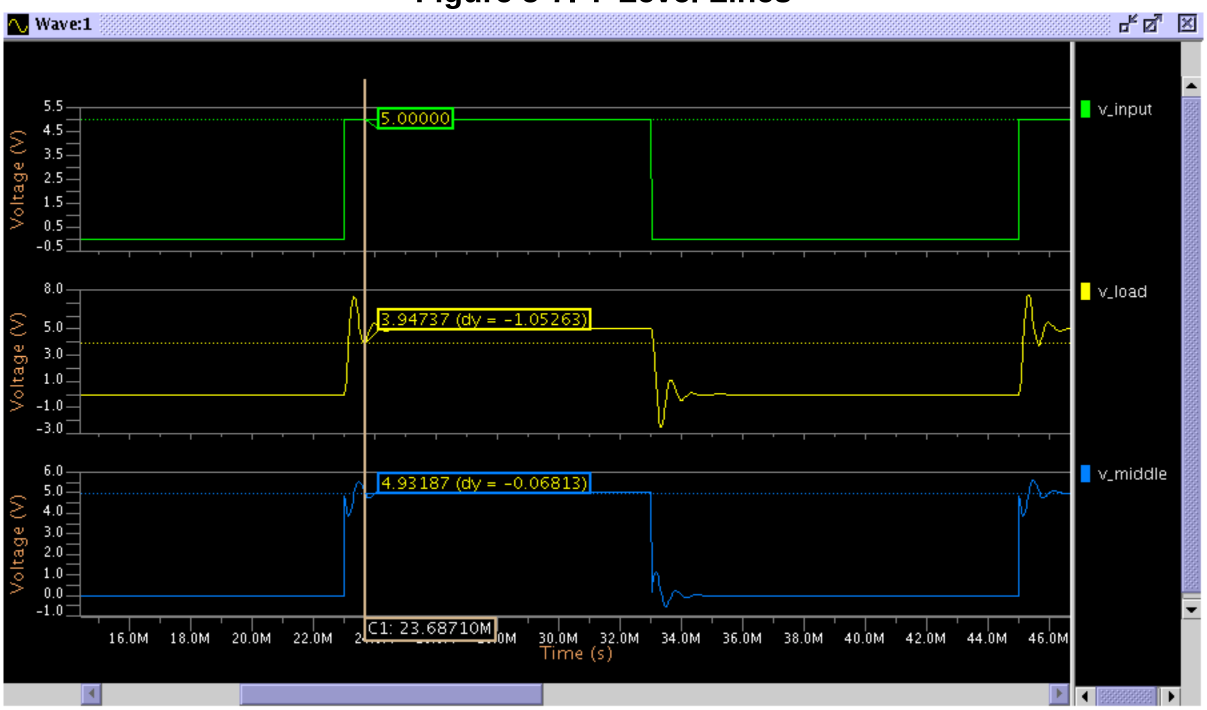
You can add a Y-level line. This is a dotted horizontal line that marks the intersection of a cursor and a waveform.

Procedure

1. Right-click a cursor and select **Y-Level Line** from the popup menu to show the Y-level line.
2. Alternatively, use the menu item **Cursor** > **Add with Y-Level** to add a new vertical cursor with Y-level lines activated.

   A dotted line is now shown on the y axis at the point where the cursor crosses the waveform.

Figure 5-7. Y-Level Lines



3. To hide Y-level lines, right-click the cursor and deselect **Y-Level Lines**.

Choosing How Delta-Y is Calculated

The delta-Y relative to the base Y-level line is displayed on every non-base Y-level line. You can control how the delta-Y values are measured.

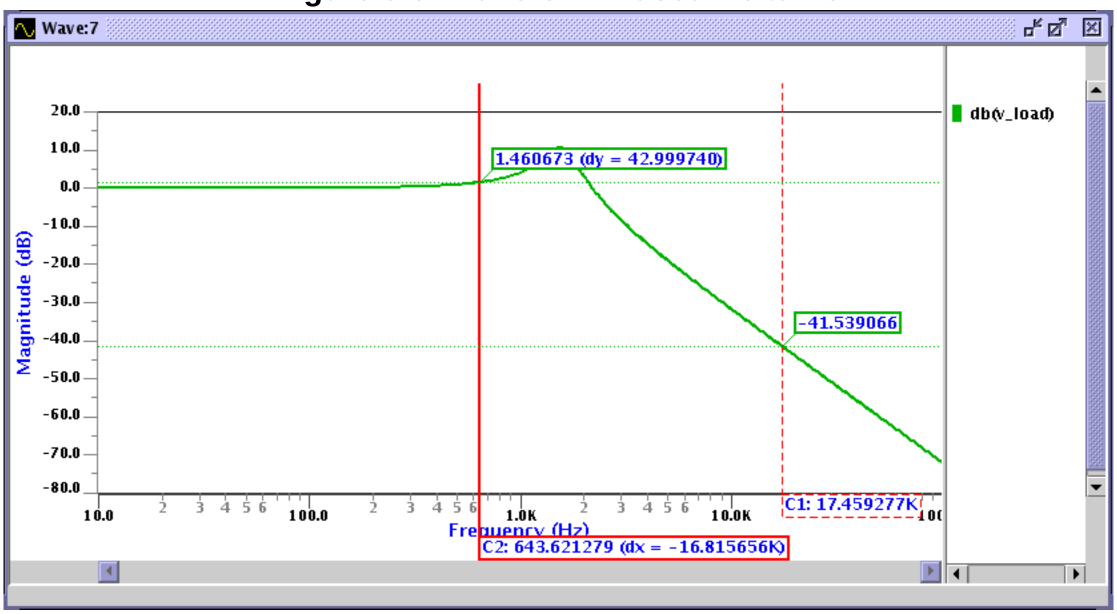
This is determined by the Cursor Delta Y setting in the Cursor Options.

The EZwave viewer can measure the delta-Y in the following ways:

- Waveform based (default)

  Measures the difference between Y-level lines on one waveform. The Y-level line of the reference cursor displays the delta between it and the base Y-level line.

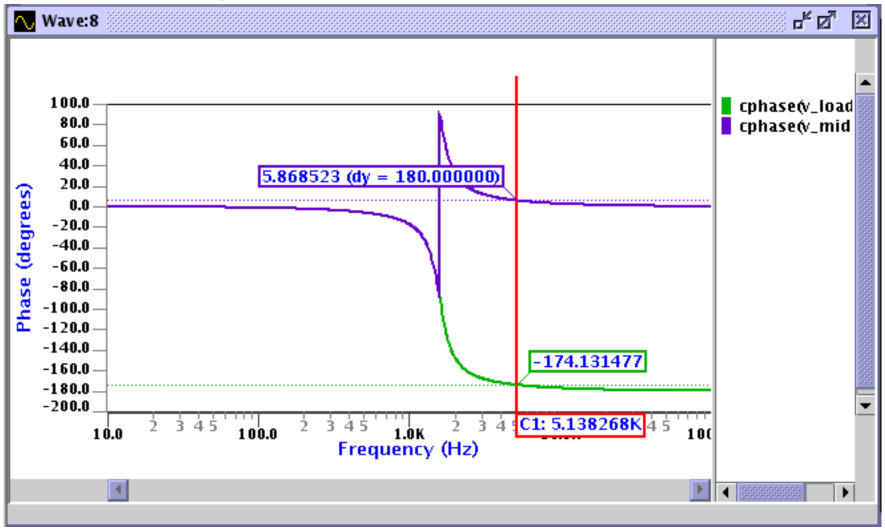
Figure 5-8. Waveform-Based Delta Ys



- Cursor based

  Measures the differences between Y-level lines on the same cursor. The delta-Y value of the reference Y-level line is the difference between it and the base Y-level line on the same cursor.

Figure 5-9. Cursor-Based Delta Ys



Related Topics

Adding the Base Cursor

Adding Reference Cursors

Adding a Horizontal Cursor

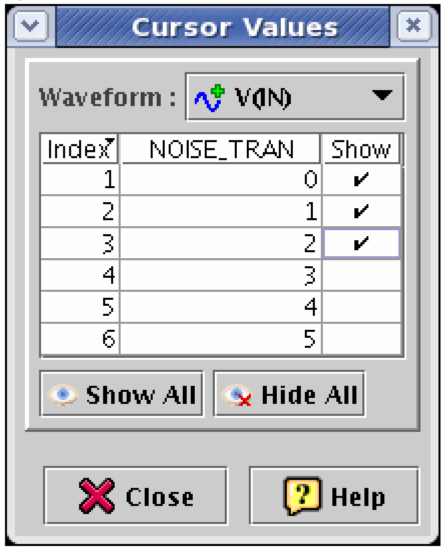
Using the Cursor Value Table With Compound Waveforms

You can view the Cursor Value Table to see information about one or more elements of a compound wave.

Procedure

1. To access the Cursor Value Table, first select an element of a compound waveform displayed in a graph window.
2. Right-click the element cursor value.
3. From the Cursor Value Popup Menu, select the **Show/Hide** menu item.

Figure 5-10. Cursor Value Table



4. The Cursor Value Table displays the following information about individual elements of a compound waveform.

    - **Index** - A sequential number assigned to each element of the compound waveform. The first element is assigned the number one. By default, the numbers are listed in ascending order. To reverse this order, click the small triangle.
    - **Parameter** - The parameter column shows the value of a parameter specified at simulation time. To invert the order of this column, click the small triangle.

Related Topics

Cursor Value Popup Menu

Changing the Visibility of Graph Window Elements

You can choose which elements of a compound waveform are visible in the graph window.

Procedure

1. Click the box or boxes in the **Show** column corresponding to the element number or numbers.
2. If multiple elements, you can group them together at either the top or bottom of the table by clicking the small arrow at the top of the column.

Related Topics

Compound Waveforms

Using Pick Points to Take Measurements

Pick points are temporary markers that enable you to perform quick measurements on selected waveforms. They can be used as an alternative to placing cursors.

Use the Pick Points dialog box to place and manage pick points on single or multiple waveforms. When this dialog boxs is active, clicking on a waveform or in the waveform row, will add a pick point marker to the waveform(s). Pick points are not persistent, meaning that they do not remain on the waveform once the dialog box is closed. To mark the point permanently, you can use this dialog box to add a cursor, slope or delta marker, or a waveform-based annotation.

Each graph window or workspace can have its own dedicated pick points. If you switch between different graph windows or workspaces without closing the Pick Points dialog box, the pick point data for each updates accordingly.

Pick Point colors can be configured on the EZwave Display Preferences dialog box—Pick Points Options (Fonts and Colors).

Taking Measurements With Pick Points **184**
Adding Delta Markers With Pick Points **187**

Taking Measurements With Pick Points

You can use the Pick Points dialog box to place and manage pick points on single or multiple waveforms and perform quick measurements.

Procedure

1. Choose **Tools** > **Pick Points** or use the icon in the main toolbar  to activate Pick Points mode.

   The Pick Points Dialog Box opens.

   The mouse pointer is replaced by a pick pointer, displayed as vertical and/or horizontal crossed bars. As you move the pick pointer in the waveform row area, the current X and Y data coordinates values are shown at the top of the Pick Points dialog box.

2. Click anywhere in the waveform row (either on waveform or on the “blank” area) to add the first pick point. The position at which the pick point marker is added on the waveform(s) depends on the current pick mode: Interpolate, Snap to Data Points or Free. To change the mode, click Show Options on the Pick Points dialog box to expose the Pick Point Mode controls.

   You can only add pick point markers to the waveform value. You cannot add pick point markers outside the waveform area, between waveform rows, or on top of an existing marker or cursor.

3. Click to add a second pick point.

   The Picked Points Values table contains information about the previous and current picked values including the difference between them for each displayed axis.

   Note

   The difference between the “picked point” and “picked value” is, that the picked point is attached to the waveform in the graph window, and is displayed as the marker, but the picked value is the X/Y data coordinates of the original mouse-click (which may not be necessarily on the waveform, but could be elsewhere in the waveform area). Picked values are only visible in the Pick Points dialog box and are not displayed in the active wave window.

   The Picked Points Waveforms table lists information about the most recently added pick point on each waveform in the graph window.

   By default, the maximum number of simultaneously visible pick points per plotted waveform is 2. If you add a third pick point, the first pick point you created is removed and the new one is added. You can change the maximum number of pick points on the EZwave Display Preferences dialog box—Pick Points options.

4. Click the Expression column header in the Picked Points Waveforms list to assign an expression to all waveforms in the list. See Table 8-81 on page 574 for details on the expression calculations available. Choose Expression from the list to disable expressions on all waveforms.

5. Click the  or  buttons to add vertical or horizontal cursors at the location of all pick points in the active graph window. Alternatively, to add cursors to one or more specific waveforms (with pick points) select them in the Picked Points Waveforms list, right-click and choose **Add Cursor(s)** or **Add Horizontal Cursor(s)**.

   Tip

   Mouse over a pick point in the graph window and use the Add Cursor keyboard shortcuts F5 and F6 to add cursors at the position of the pick point. The cursors will snap to the pick point.

6. Click the  button to add a waveform-based text annotation at the position of each pick point in the active graph window. The annotation text contains the x and y coordinates of the pick point, and the slope, parameter and expression value (if present).

   Alternatively, to add a text annotation to the most recently added pick point on a waveform, right-click the required waveform in the Picked Points Waveforms list, and choose **Add Text Annotation(s)**.

To edit the annotation text, right-click the annotation and choose **Edit**. The Waveform Annotation dialog box that opens is pre-filled with:

```
X = %x
Y = %y
SlopeParam = %s
Param = %p
<expression_name> = %e
```

Where %x, %y, %s, %p and %e are placeholders for the corresponding values from the Picked Points Waveforms list on the Pick Points dialog box.

7. Use the and buttons to add slopes and delta measurements between pick points. See “Adding Delta Markers With Pick Points” on page 187.

Tip

You can also right-click a pick point in the graph window, or on a waveform in the Picked Points Waveforms list to access these functions.

8. Use the Delete button to remove all annotations, slope markers, cursors and delta markers currently attached to existing pick points in the graph window. Pick point markers themselves are not removed. This button cannot be used to remove annotations, markers or cursors once their associated pick point has been removed, or if their annotation, slope marker, cursor and delta marker were moved outside the picked point marker area.

9. You can filter the Picked Points Waveforms list by entering a regular expression into the Filter Names field. Only waveforms that match the specified term will be listed.

10. Right-click a current or previous value in the Picked Values table and select **Copy Value(s) to Clipboard** to copy the value for pasting elsewhere. If multiple cells are selected, copied values will be pasted as a comma-separated list. Use the **Send Value(s) to Calc** menu item to send it to the Waveform Calculator.

    You can also right-click an item in the Picked Points Waveforms list to access functions to copy the waveform name and send it to the Waveform Calculator.

Note

All popup menu actions in the Picked Points Waveforms list are applied to the last pick point of the selected waveform(s).

11. Click **Save As** to save the current Picked Points Waveforms list to *.csv* or *.txt* format.

12. Information can be cleared from the Picked Values table and the Picked Points Waveforms list using the Clear Values and Clear Waveforms buttons, respectively. Use the **Clear All** button to clear all listed values from both tables, and remove all pick points from waveforms in the graph window.

13. To deactivate pick points mode, close the Pick Points dialog box either by clicking **Close**, or by pressing the Esc key. All pick points in all graph windows will be removed.

Related Topics

Pick Points Dialog Box

Pick Points Options

Adding Delta Markers With Pick Points

Adding Delta Markers With Pick Points

You can add a delta marker between the pick points of each waveform in the current active graph window.

Procedure

Click the button on the Pick Points dialog box to add a delta marker between the pick points of each waveform in the current active graph window (where the addition is possible). Delta markers are added in a “chain” meaning that they are added between all pick points of the waveform starting from the first one (by creation order).

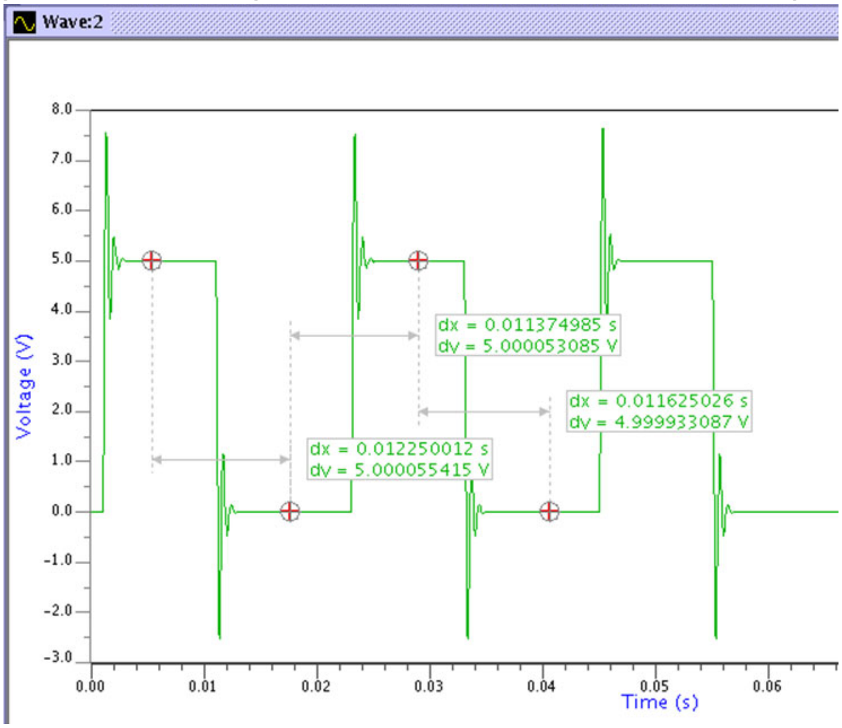
Tip

You can right-click on a delta marker text box and select **Edit** to change the annotation text.

Examples

The waveform in the following figure has four pick points. The **Add Delta** button will add delta markers between the 1st and 2nd, 2nd and 3rd, and the 3rd and 4th pick points.

Figure 5-11. Adding Deltas From the Pick Points Dialog Box



Related Topics

Pick Points Dialog Box

Pick Points Options

Using Text Annotations

Working with Cursors

wave adddeltamarker

Using Text Annotations

A text annotation is a text box that you can use to add labels, or other textual information, to waveforms or to the graph window.

A text annotation that is attached to a waveform is bound to that waveform's row. Once placed, the text annotation's anchor can only be moved within its current row but its text box may be moved outside the row.

A text annotation that is added to the graph window (not attached to a waveform) can be moved freely within the window, and is not associated with any row.

Adding Text Annotations 189
Global Display Options for Text Annotations. 192

Adding Text Annotations

You can use text annotations to add labels, or other textual information, to waveforms or to the graph window.

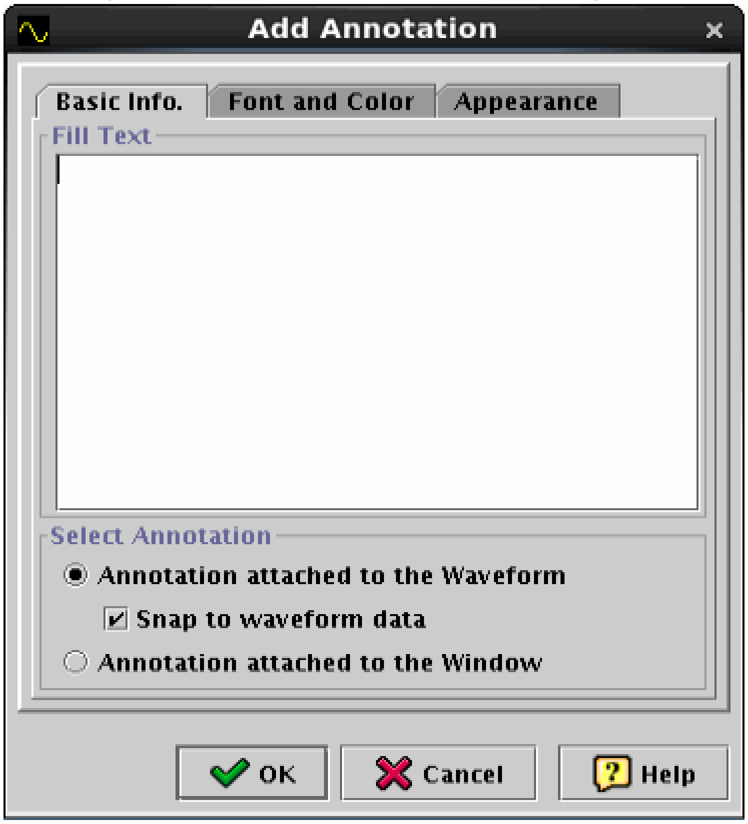
Procedure

1. To add a text annotation to a waveform, select a waveform and right-click to display the Waveform Popup Menu.

   To add a text annotation to the graph window, right-click anywhere in the graph window to display the Row Popup Menu.

2. Choose **Annotation** > **Add Text Annotation** from the Row popup menu. The Add Annotation dialog box opens.

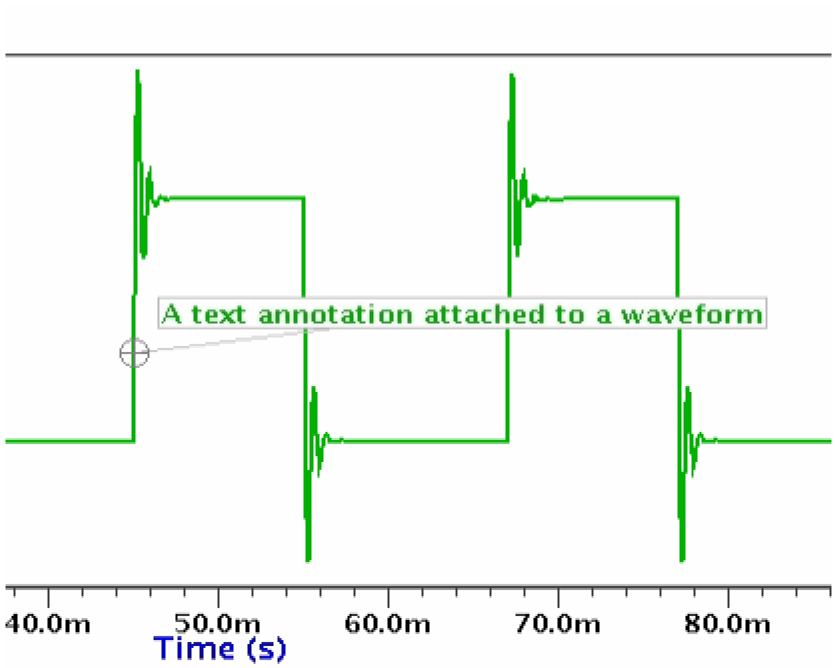
Figure 5-12. Add Annotation Dialog Box



3. Type the desired text for the annotation in the Fill Text field. You can also select here whether to attach the text annotation to the waveform or the graph window.

   When text annotations are attached to waveforms, an anchor symbol is shown at the point where the anchor line from the text annotation box touches the waveform.If you attach the annotation to a waveform, you can also choose Snap to waveform data to snap to the nearest data point after a mouse drag and drop of the anchor. Click **OK**.

Figure 5-13. Text Annotation Examples



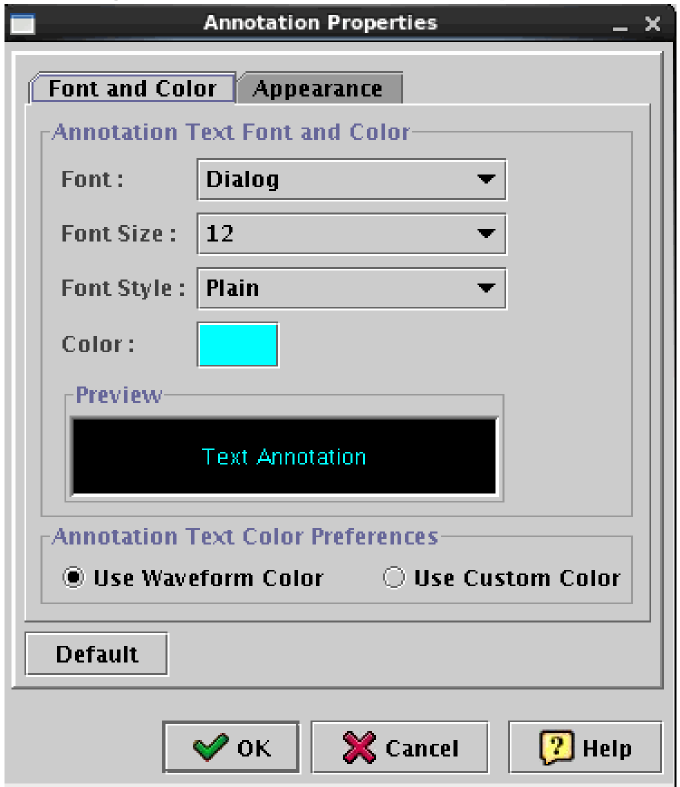
By default, text annotations that are attached to waveforms use the same color setting as the waveform.

4. Optionally, you can attribute custom font and color settings specific to a text annotation.

   To do this whilst adding a new annotation: use the **Font and Color** and **Appearance** tabs on the Add Annotation dialog box.

   To do this for an existing annotation: right-click the text annotation and select **Properties**. The Annotation Properties Dialog Box opens, enabling you to make changes.

Figure 5-14. Annotation Properties



Note

Once custom properties are applied to a text annotation, it will not be affected by any changes to global text annotation settings (those specified on the Text Annotation Options (Fonts and Colors) and Text Annotation Options pages of the EZwave Display Preferences dialog box).

Tip

You can right-click an annotation to show the Annotation popup menu. From here you can **Edit** the text, **Delete** the annotation, **Delete All Annotations**, **Copy Value to Clipboard**, and edit the annotation **Properties**.

Global Display Options for Text Annotations

You can define the default font and color settings used when adding text annotations to waveforms.

Procedure

1. Choose **Edit** > **Options**.

   The EZwave Display Preferences dialog box opens.

2. Choose **Text Annotations** from the list on the left-hand side of the EZwave Display Preferences dialog box.

   The Text Annotations panel is displayed.

3. Use the Text Annotations options to control visibility of text annotation elements, such as the anchor point symbol or the anchor line.

4. Expand the Fonts and Colors list, and choose Text Annotation Options (Fonts and Colors).

5. Select the required font, style and size from the dropdown lists.

6. In the Waveform Annotation Color section, select **Use Custom Color**, and choose a color.

   The preview pane shows a sample of what the annotation will look like in the current color scheme.

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

8. All new text annotations will inherit these new settings. Existing text annotations will not be affected. You can still apply custom settings for individual annotations, by right-clicking on the annotation and selecting **Properties**.

Related Topics

EZwave Display Preferences Dialog Box

Text Annotation Options (Fonts and Colors)

wave addannotation

Using the Event Search Tool

You can use the Event Search Tool to locate occurrences of simulation events interactively. An event is a specific state (or value) for a single or collection of waveforms.

Performing an Event Search **194**
Performing an Expression Event Search **195**

Performing an Event Search

You can use the Event Search Tool to perform a basic event search.

Procedure

1. Choose **Tools** > **Search**.

   The Event Search Tool Dialog Box opens.

2. Name the event in the text window of the Event field. Click **Save** if to save the event definition for later use, if required.

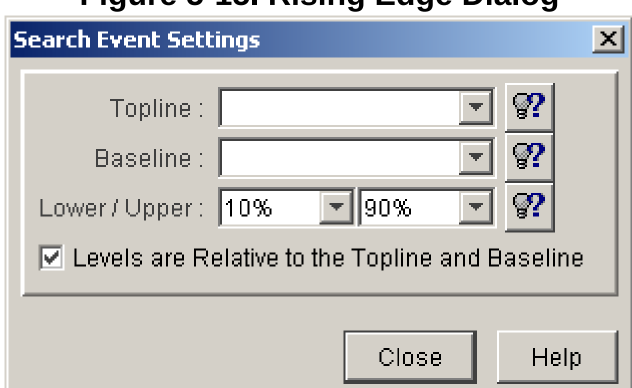
3. To specify the waveform to search by, select a waveform in the active graph window and then click on  to add it to the Source Waveform field.

4. Choose an event type from the following four options:

   - Any Event — Any type of event.
   - Rising Edge — Only rising edge occurrences.

     Opens the Search Event Settings dialog box.

Figure 5-15. Rising Edge Dialog



     i. Check the Levels are Relative to the Topline and Baseline box to automatically select the Top and Base line.

     ii. Select the lower and upper percentages through the pulldown menu.

iii. To manually set the Top and Base lines, deselect the Levels are Relative to the Topline and Baseline box and type appropriate lower and upper values.

iv. Close the Search Event Settings dialog box and click to show the lines in the graph window.

- Falling Edge — Only falling edge occurrences. Opens the Search Event Settings dialog box. Specify values as for Rising Edge.
- Waveform Value — Only waveform values. A tolerance needs to be specified when searching for data points generated by an analog simulator.

5. Move the cursor along the specified waveform to search for the specified events by clicking to go in the decreasing time value direction or to go in an increasing time value direction.
6. To mark an event, in the Markers section of the Event Search Tool select the Add a Marker icon . Right-click the marker to rename it, move it, copy it to the clipboard, or delete it, as desired.
7. Jump to a marker by selecting the marker name from the drop-down list or by using the previous marker icon and next marker icon .

Related Topics

Event Search Tool Dialog Box

Performing an Expression Event Search

Performing an Expression Event Search

In addition to using the Event Search Tool to locate occurrences of simulation events interactively, you can perform an expression event search.

Procedure

1. Choose **Tools** > **Search**.

   The Event Search Tool dialog box opens.
2. Name the event in the text window of the Event field. Click **Save** if to save the event definition for later use, if required.
3. Select the Logic Expression option.
4. Select the expression by:
   - Invoking the Waveform Calculator to create or select an existing expression. Click the Eval button in the calculator to import the expression from the calculator to the Event Search expression area.

- Using the **Add Selected Waveform** icon and type the logical function names or operators in the expression area.

5. Move the cursor along the specified waveform to search for the specified events by clicking to go in the decreasing time value direction or to go in an increasing time value direction.
6. To mark an event, in the Markers section of the Event Search Tool select the Add a Marker icon . Right-click the marker to rename it, move it, copy it to the clipboard, or delete it, as desired.
7. Jump to a marker by selecting the marker name from the drop-down list or by using the previous marker icon and next marker icon .

Related Topics

Event Search Tool Dialog Box

Waveform Calculator

Using the Event Search Tool

Performing an Event Search

Working with Eye Diagrams

This section describes the procedures for creating eye diagrams and editing industry standard eye masks, and details the measurements available within an eye diagram.

Tip

See also the eye diagram functions listed in the Statistical Functions table in Waveform Calculator "Function Descriptions" on page 662.

Creating an Eye Diagram **197**
Adding Additional Waveforms to Eye Diagrams **203**
Editing an Eye Mask **203**
Eye Diagram Measurement Calculations **206**
NRZ (PAM2) Cross Eye Calculation 206
PAM3 and PAM4 Eye Calculations 208
C-Phy Eye Calculation 213

Creating an Eye Diagram

You can plot eye diagrams based on a period of waveform data.

Procedure

1. Choose **Tools** > **Eye Diagram**.

   The Eye Diagram Tool Dialog Box opens.

2. From the Eye Type dropdown list on the **Settings** tab, specify either:

   - C-Phy (3 source waveforms)
   - NRZ (2 amplitude levels)
   - PAM 3 (3 amplitude levels)
   - PAM 4 (4 amplitude levels)

   The Eye Diagram Tool GUI fields change depending on the eye type selected.

3. Plot and select the required Source Waveform and click the **Add Selected Waveforms**  icon. For C-Phy eye type only, exactly three waveforms must be selected (in order).

   The Eye Parameters values reflect the selected source waveform:

   - Eye Period — The eye diagram is generated by overlaying a semi-periodical waveform signal on an interval in X (usually a time interval). The interval is defined as the eye period. A default eye period is calculated based on period divided by 2.

- Period Tolerance — (C-Phy only) Specifies the period tolerance to detect a transition.
- Offset — (NRZ, PAM 3, PAM 4 only) The open part of the eye is not always at the center of the axes. Specify a different offset value to generate an eye diagram with the open part at the desired location. You can press the keyboard Enter key each time you change the offset value, as an alternative to clicking **Apply**.

  Tip

  It is sometimes not easy to setup the offset parameter. After you have set up the eye diagram parameter and plotted the waveform, you can right-click the waveform at the desired location and choose **Set Offset** > **Align Here** or **Set Offset** > **Center Here** from the popup menu. **Align Here** sets the X offset at this position, that is, the eye becomes left and right aligned to this location. **Center Here** sets the offset so that this location becomes the center of the eye, that is, the offset is set to the Period/2.

- Minimum X value and Maximum X value — Defines the range of waveform data used for generating the eye diagram.

Note

If the selected waveform is a compound waveform, all Eye Post-processing and Eye Mask options will be deselected. These may be reselected. Refer to the Eye Parameters Calculation options accessed via **Edit** > **Options** > **Multiple Run**. This provides options for calculating parameters and plotting.

Tip

You can type expressions into any numeric fields. For example, to calculate the Eye Period value of a waveform with frequency 1 MHz, type the expression 1/(2*1meg) into the Eye Period field. You can also incorporate a variable that you have calculated using the Waveform Calculator. For example, use the Waveform Calculator to calculate the variable freq_mean=mean(frequency(wf(…))).

Then type the expression **1/(2*freq_mean)** into the Eye Period field.

4. (Optional) In the Eye Post-Processing section, click the Apply Measurements checkbox and specify the required information in the Eye Measurements section. This section controls how the measurements are displayed and updated on the selected source waveform:

   - For C-Phy signals only, you can specify the X value and Y value where the measurements are to be made.
   - For NRZ signals only, select Apply Width and Height to have the measurements automatically updated and displayed as the values are modified in this dialog box. This updates the values in the eye diagram as well as all of the values in the

Measurement Results tab. Select Apply Inner/Outer to have the inner or outer height or width displayed at the specified X or Y.

- For PAM 3 and PAM 4 signals only, specify the X value where the measurements are to be made. Measurements for each of the three eyes are performed. Once calculated, you can independently drag the measurements to different positions along the X axis. You can also specify the Reference Amplitude Ratio. When set to "Automatic", toplines and baselines begin at 10% and increase in steps of 5% until all points are either below the baseline and above the topline (the Measured Edge Percentage reaches 100%).

5. (Optional) In the Eye Post-Processing section, for C-Phy and NRZ signals only, click the Apply Mask checkbox and specify the required information in the Eye Mask section. This section relates to the mask displayed with the eye diagram. The setup values include:

   - Specifications— Choose an industry standard mask from the dropdown list.

     Select the **Eye Mask** icon  to the right of the dropdown list to open the Eye Mask dialog box displaying the shape and values for the corresponding mask.

   - Margin — Enables scaling of the eye mask. The X margin is the horizontal margin and the Y margin is the vertical margin, the values entered representing a percentage of the total displayed range.

     Select the Automatic Fit checkbox for X and/or Y to make the mask fit to the horizontal and/or vertical inner contours of the eye diagram respectively.

   - Offset — Enables shifting of the eye mask. The X offset is the horizontal offset and the Y offset is the vertical offset.

     Select the Automatic Fit checkbox for X to automatically determine the horizontal offset.

6. When you have the desired setup, click the **Measurement Results** tab. This is where to view the results of the eye diagram measurements, once calculated.

7. Click **Apply**. You can also press the keyboard **Enter** key each time you change the value, without closing the dialog box, as an alternative to clicking **Apply**. The eye diagram is displayed in a new graph window. Click **OK** or **Cancel** to close the Eye Diagram Tool dialog box.

Figure 5-16. C-Phy Eye Diagram Example

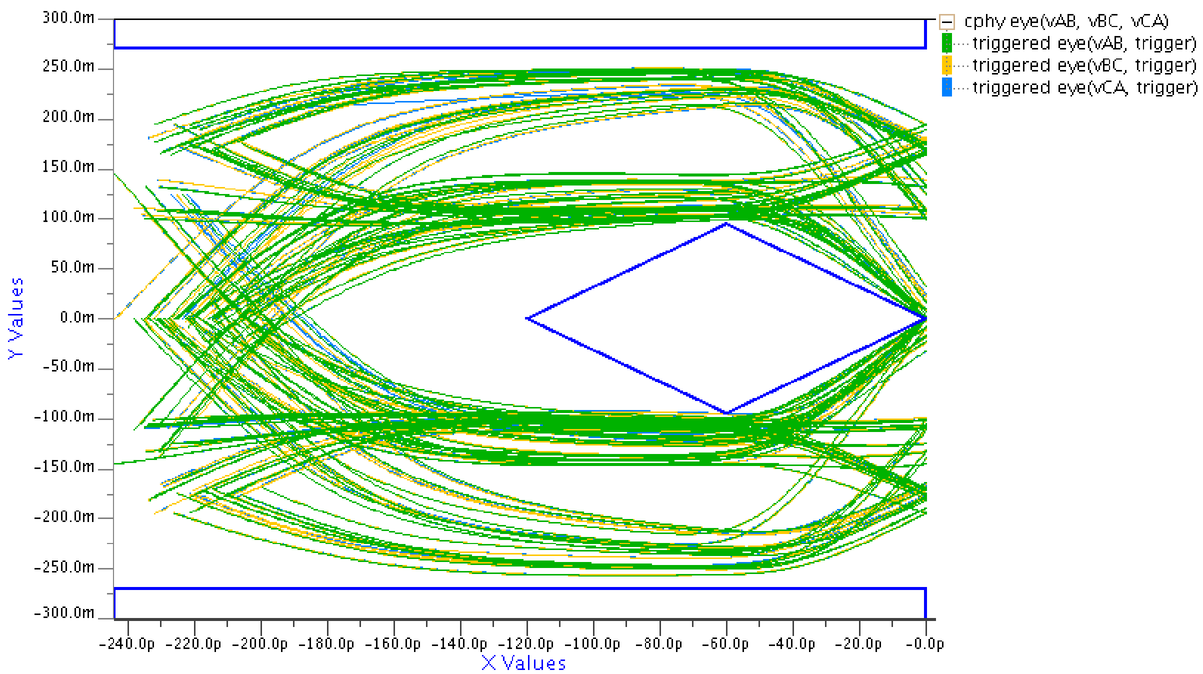
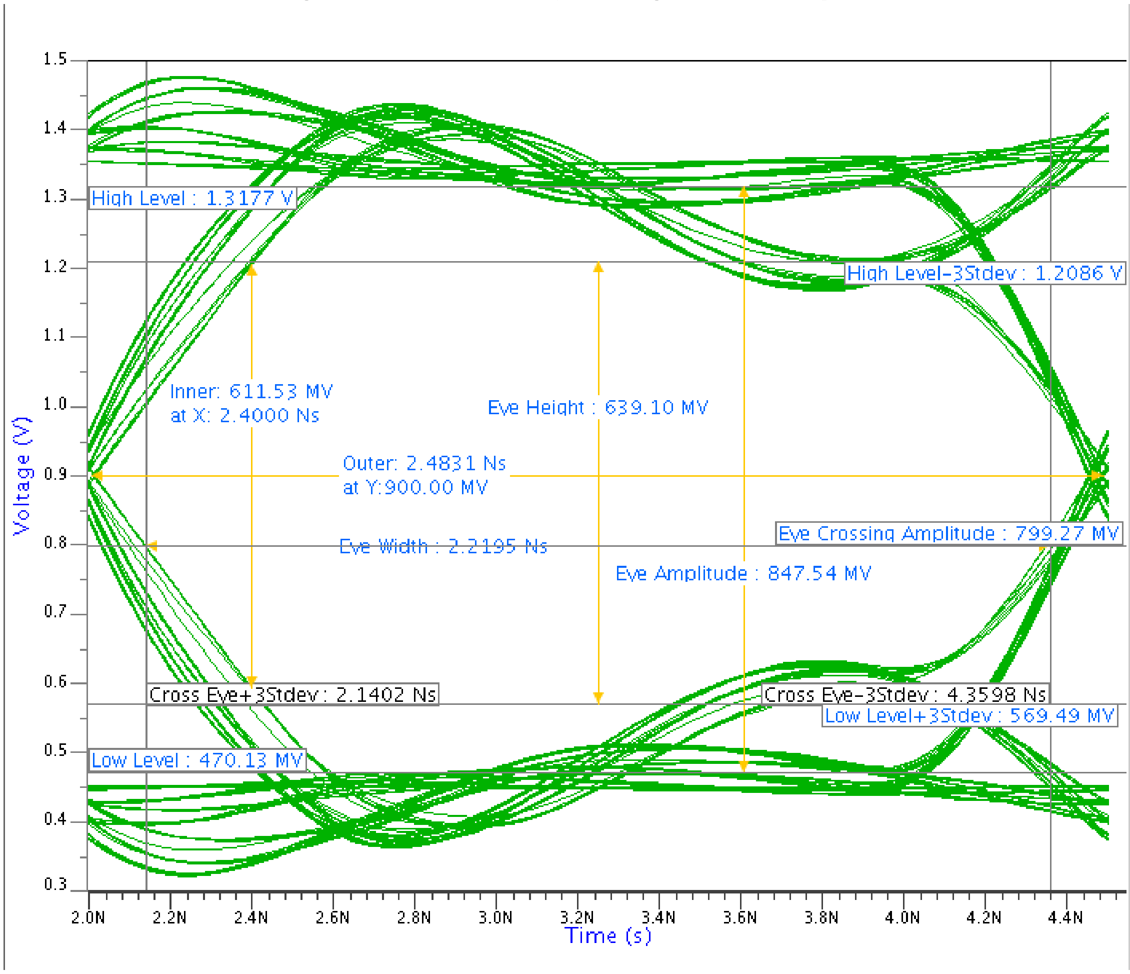
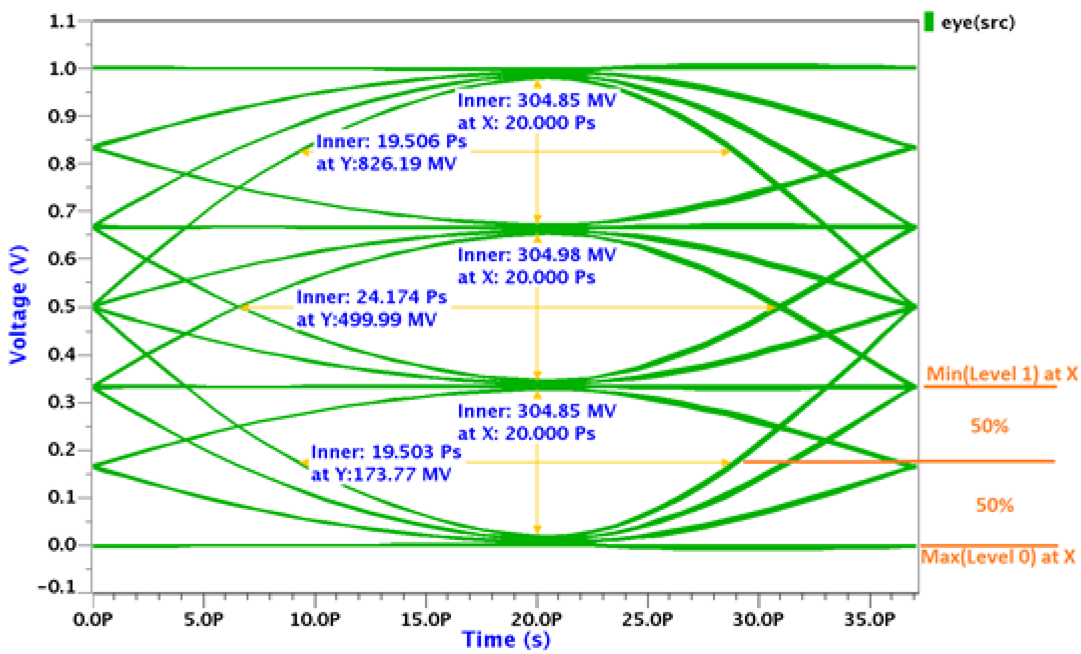


Figure 5-17. NRZ Eye Diagram Example



When a NRZ eye diagram is displayed, right-click anywhere on the eye diagram and choose **Add Cursor** to place a cursor enabling the exact values to be seen.

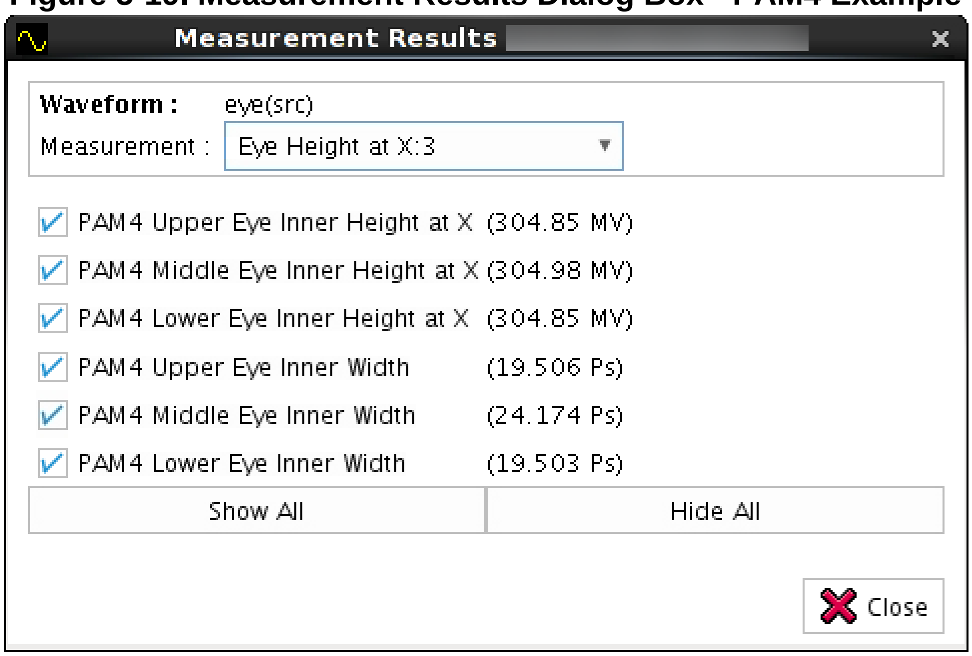
Figure 5-18. PAM4 Eye Diagram Example



You can drag and drop the measurement arrows to see measurements at different locations.

Right-click a PAM 3 or PAM4 measure, and select **Measurement Results** to display the inner height and inner width measures on the Measurement Results Dialog box.

Figure 5-19. Measurement Results Dialog Box - PAM4 Example



Related Topics

Eye Diagram Tool Dialog Box

Eye Mask Dialog Box

Adding Additional Waveforms to Eye Diagrams

You can drag and drop new waveforms from the Waveform List, onto an existing eye diagram, plotted in the graph window.

When you drag and drop new waveforms onto an existing eye diagram, the behavior depends on whether the waveform is a simple waveform or a compound waveform, as described here.

Simple Waveforms

The new waveform is plotted using the same offset, period and X range parameters, and the same graphical measurements and masks are applied.

If a graph window contains several eye diagrams, new waveforms can be placed inside a specific row, above or below all existing rows, or between rows.

When dropping a new waveform inside an existing row, the new eye diagram is plotted using the same parameters as the occupying waveform.

When dropping a new waveform above, below or between existing rows, a new row is created to contain the new eye diagram, which uses the same parameters as the first eye diagram in the graph window.

You can drag and drop multiple waveforms simultaneously into the graph window. In this case, the dragged waveforms are plotted in the same row.

Compound Waveforms

For compound waveforms, the options for calculating parameters and plotting are controlled by the Eye Parameters Calculation options. These are accessed via **Edit** > **Options** > **Multiple Run**.

Related Topics

Eye Diagram Measurement Calculations

Editing an Eye Mask

Editing an Eye Mask

You can edit an eye mask using the Eye Diagram Tool.

Procedure

1. Select a mask from the dropdown list in the Eye Mask frame of the Eye Diagram Tool dialog box.
2. Click the Eye Mask icon .

   The Eye Mask dialog box opens, displaying the shape and values for the corresponding mask.
3. To specify an alternative source eye mask, select a built in eye mask from the drop down list or use the Open Folder icon to load a user defined eye mask from disk.
4. Click the Edit New Mask icon to edit the mask.
5. Type a name for the new mask in the Create a New Eye Mask dialog box. Editing is enabled and the grayed out Save and Delete buttons and Edit frame become available.
6. Edit the eye mask:
   - Click to make the handles available so that the mask can be dragged to form a new shape.
   - Click to add new handles or delete existing ones. Click an existing handle to remove it. Click an area without a handle to add one.
     - Point Handles enable the mask to be dragged in any direction.
     - Vertical Handles enable the mask to be dragged horizontally.
     - Horizontal Handles enable the mask to be dragged vertically.
     - Barycenter handles enable the mask to be moved as a whole. They do not alter the high and low levels, which can be adjusted manually.
     - The Coordinate Flags display the X and Y values that relate to the handles. The X coordinate flags show a percentage of the eye diagram X range. The Y coordinate flags show absolute values.
   - Check X (Y) Symmetric to also modify the symmetric handles in the mask when a handle is dragged, ensuring the mask remains symmetric. This is only possible in masks with X (Y) symmetry. When unchecked, only the dragged handle is modified.
7. Click **Save** to save the new eye mask.

Related Topics

Eye Diagram Tool Dialog Box

Eye Mask Dialog Box

Creating an Eye Diagram

Eye Diagram Measurement Calculations

Eye Diagram Measurement Calculations

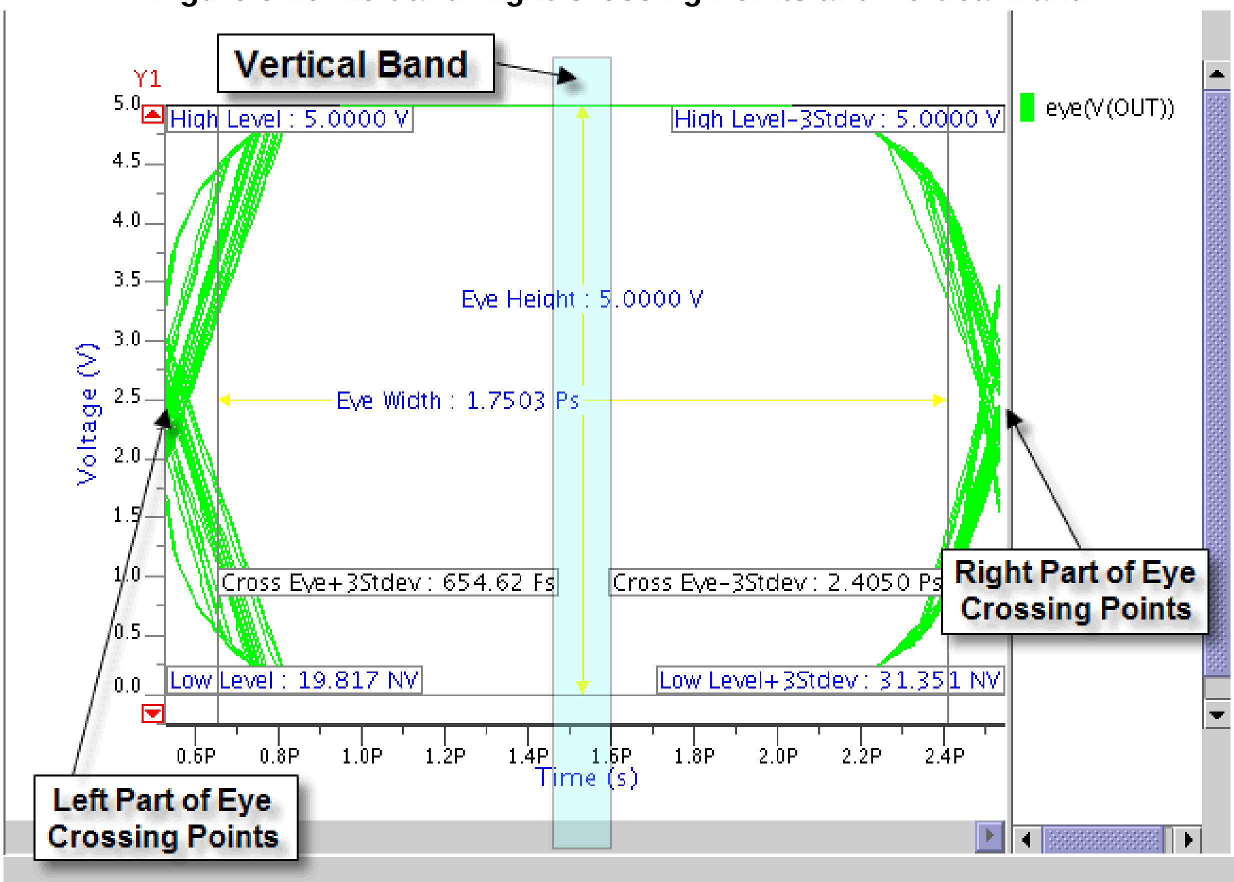
This section describes how each of the eye diagram measurements is calculated

NRZ (PAM2) Cross Eye Calculation 206
PAM3 and PAM4 Eye Calculations 208
C-Phy Eye Calculation 213

NRZ (PAM2) Cross Eye Calculation

The NRZ (also know as PAM2) Cross Eye measurement is calculated as the middle of the High Level and the Low Level. The points of the eye diagram cross this horizontal level ("eye crossings") on the left and right of the diagram. The distance between the left- and right-crossing is called the Cross Eye.

Figure 5-20. Left and Right Crossing Points and Vertical Band



The Vertical Band is calculated as follows. Considering the Left Part (Xmin) and Right Part (Xmax) of the eye crossing points, the midpoint of the vertical band is calculated as (Xmax-Xmin)/2. The Vertical Band left width is then set to 10% less than the midpoint, and its right width is set to 10% more than the midpoint.

The following table describes each eye diagram measurement:

Table 5-1. NRZ Eye Diagram Measurements

Measurement	Description	Notes
Eye Height	Calculated as the difference between the High Level -3stdev and the Low Level +3stdev calculations.	
Eye Amplitude	Calculated by subtracting the Low Level value from the High Level value.	
Eye Width	Calculated as the difference between the Cross Eye - 3stdev and the Cross Eye + 3stdev calculations.	
Eye Crossing Amplitude	The amplitude level at which the eye crossings occur, based on the middle of the High Level and Low Level.	
High Level	The mean of the point distribution in the upper part of a “Vertical Band” around the midpoint of the Cross Eye. The High Level standard deviation (high_level_stdev) is computed within this band.	
High Level - 3stdev	This is calculated by subtracting (3 * high_level_stdev) from the High Level value.	
Low Level	The mean of the point distribution in the lower part of a “Vertical Band” around the midpoint of the Cross Eye. The Low Level standard deviation (low_level_stdev) is computed within this band.	
Low Level + 3stdev	Calculated by adding (3 * low_level_stdev) to the Low Level value.	
Risetime	The mean time between the low and high threshold values, calculated from 10% to 90% of the eye amplitude.	**Results** tab only
Falltime	The mean time between the high and low threshold values, calculated from 10% to 90% of the eye amplitude.	**Results** tab only
Cross Eye + 3stdev	Calculated by adding (3 * cross_eye_stdev) to the left value of the Cross Eye.	
Cross Eye - 3stdev	Calculated by subtracting (3 * cross_eye_stdev) to the right value of the Cross Eye.	
Eye Delay	The distance from the midpoint of the eye to the time origin, measured in seconds.	**Results** tab only
Eye Jitter	The width of the eye crossing points.	**Results** tab only
Eye SNR	Calculated by the following formula: $$\frac{High\_Level - Low\_Level}{high\_level\_stdev + low\_level\_stdev}$$	**Results** tab only

Table 5-1. NRZ Eye Diagram Measurements (cont.)

Measurement	Description	Notes
Inner Height	The inner eye height.	
Outer Height	The outer eye height.	
Inner Height at X Setup	The inner Eye Height at a specified X. Drag the results line within the eye diagram to move to a new X location. Right-click the results within the eye diagram to access the Eye Height at X dialog box with options including the ability to switch between inner and outer, and to move to another X.	
Inner Width	The inner eye width.	
Outer Width	The outer eye width.	
Inner Width at Y Setup	The inner Eye Width at a specified Y. Drag the results line within the eye diagram to move to a new Y location. Right-click the results within the eye diagram to access the Eye Width at Y dialog box with options including the ability to switch between inner and outer, and to move to another Y.	

Tip

See also "Eye Diagram Tool - Measurement Results Tab" on page 465 and the eye diagram functions listed in the Statistical Functions table in Waveform Calculator "Function Descriptions" on page 662.

Related Topics

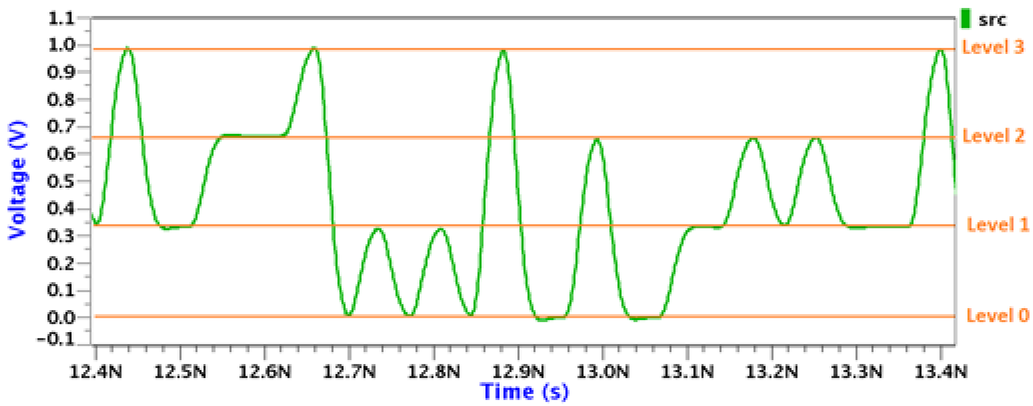
Editing an Eye Mask

PAM3 and PAM4 Eye Calculations

A PAM3 waveform is a Pulse Amplitude Modulation waveform with three levels, and contains 2 eyes. A PAM4 waveform has four levels and contains 3 eyes. In this section, we will use the PAM4 waveform as the example.

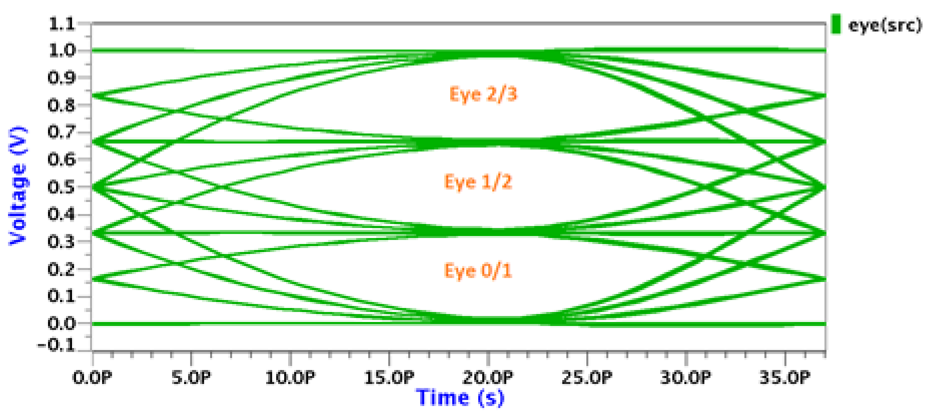
A PAM4 waveform has four levels that are identified by their number in range 0..3, starting from bottom: "Level .0" to "Level 3".

Figure 5-21. PAM4 Waveform



A PAM4 eye contains 3 eyes. They are identified by the levels they are in between: "Eye 0/1" (bottom or lower eye), "Eye 1/2" (middle eye) and "Eye 2/3" (top or upper eye).

Figure 5-22. PAM4 Eye Diagram

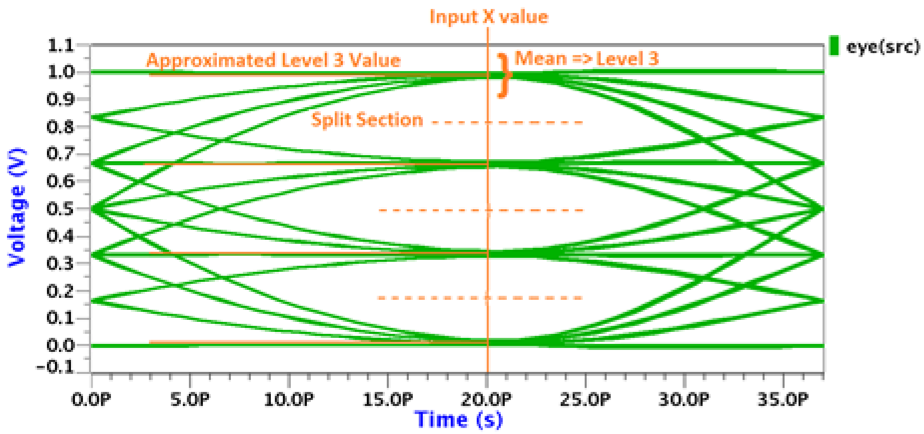


To compute measures on a PAM4 eye diagram, you have to provide an input X value (in the eye-diagram X-range) specifying where to compute the sampling

The levels are computed as follows:

- Approximated level value are computed for each level, using a histogram made of values of the eye at the given input X value
- The eye diagram is split horizontally in 4 parts, half-way between each approximated level values
- The Level Value for each eye-split is then the mean of the eye Y value at the given input X value

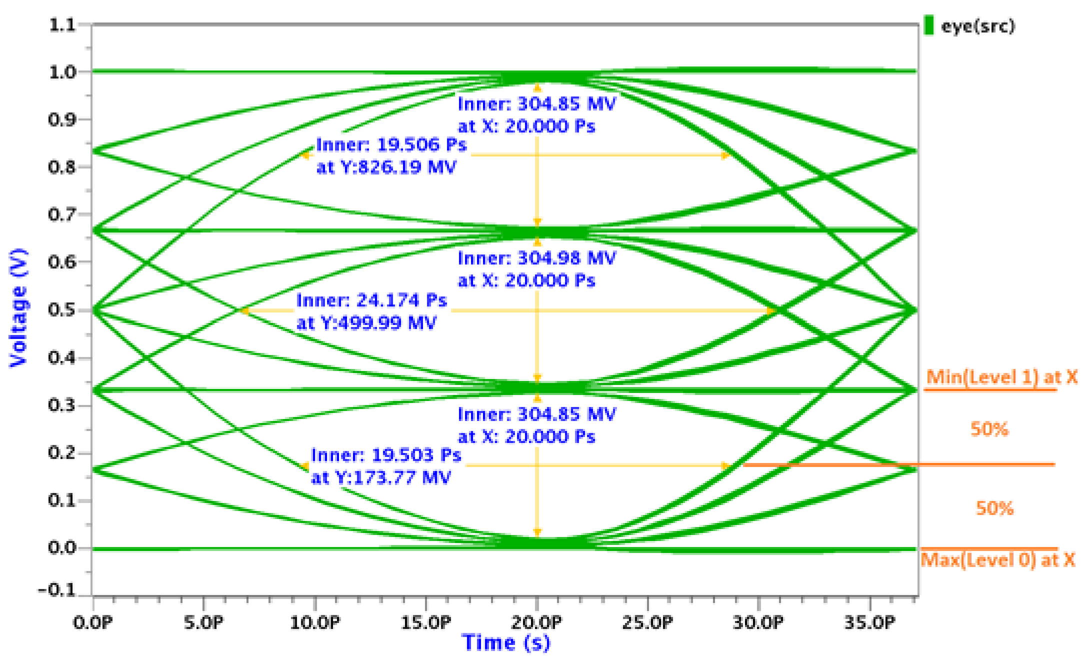
Figure 5-23. PAM4 Eye Levels



The Eye Inner Height is the distance at the given input X value between the upper point of the level below and the lowest point of the level above.

The Eye Inner Width is the distance, measured half way of an inner height, of the right most point on the left to the left most point on the right.

Figure 5-24. PAM4 Height and Width



The rise time and fall time are computed as follows:

- When the waveform rises from a low level to an upper one, the rise time is the time between (lower level + delta) to (upper level - delta)
- When the waveform falls from a low level to a lower one, the fall time is the time between (upper level - delta) to (lower level +delta)
- The delta is 10% of the level amplitude, i.e. (Level_2 - Level_0) * 0.1. The delta is NOT fixed, for example, at 10% of the max level difference.
- Level 0 to Level 1, mean for transition 0-2, 0-3, 1-2, 1-3, 2-3 => 6 rise times).

The eye distortion (Ratio of Level Mismatch) and eye linearity are computed as follows:

- RLM = min((3*ES1), (3*ES2), (2 - 3*ES1), (2 - 3*ES2))

  where:

  - $V_{mid} = (V_0 + V_3)/2$
  - $ES1 = (V_1 - V_{mid}) / (V_0 - V_{mid})$
  - $ES2 = (V_2 - V_{mid}) / (V_3 - V_{mid})$

  Note

  There is no RLM for PAM3 eyes.

- Eye Linearity = $\min(AV_{upp}, AV_{mid}, AV_{low}) / \max(AV_{upp}, AV_{mid}, AV_{low})$

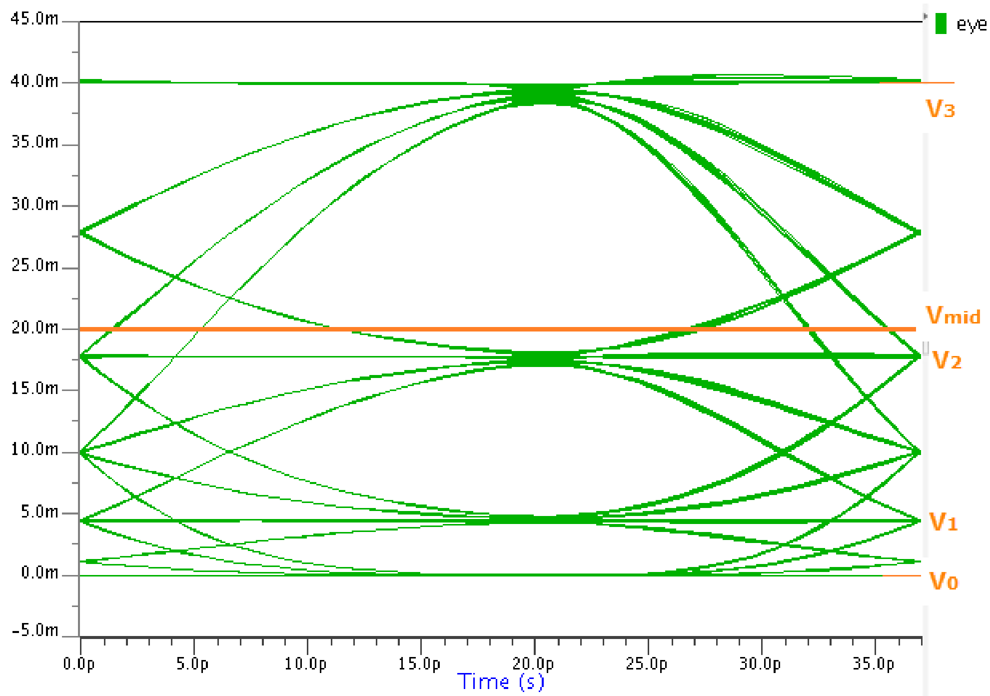
  where:

  - $AV_{upp} = V3 - V_2$
  - $AV_{mid} = V_2 - V_1$
  - $AV_{low} = V_1 - V_0$

  Note

  There is no AV_{mid} for PAM3 eyes.

Figure 5-25. PAM 4 Eye Linearity and RLM



The following table describes each eye diagram measurement:

Table 5-2. PAM4 Eye Diagram Measurements

Measurement	Description
at "X"	The first row (at "X") result is the measurement setup value, entered by user in Measurements Setup section of the Eye Diagram Tool - Settings Tab.
Eye Inner Heights	The upper, middle and lower inner eye heights at X.
Eye Inner Widths	The upper, middle and lower inner eye widths.
Levels	The four eye levels.
RLM	Measure of eye distortion (Ratio of Level Mismatch).
Eye Linearity	Measure of eye linearity.
Reference Amplitude Percentage	Indicates the percentage of amplitudes (difference between top-level and low-level) used to compute baselines and toplines. Risetimes are the times from baselines to toplines. Falltimes are times from toplines to baselines.
Measured Edge Percentage	The number measured risetimes + falltimes, divided by the total number of (rising + falling) edges, and multiplied by 100.
Risetimes	The risetimes between each of the different levels.

Table 5-2. PAM4 Eye Diagram Measurements (cont.)

Measurement	Description
Falltimes	The falltimes between each of the different levels.

Tip

See also “Eye Diagram Tool - Measurement Results Tab” on page 465and the eye diagram functions listed in the Statistical Functions table in Waveform Calculator “Function Descriptions” on page 662.

C-Phy Eye Calculation

A MIPI C-Phy eye-diagram is made of the 3 differential lines. It is triggered by the first 0 crossing at each transition, and is right aligned.

Waveform Calculator Functions

The following Waveform Calculator Statistical functions support the MIPI C-Phy V1.2 eye diagram measurement method:

- cphytrigger

  Creates first zero-crossing trigger waveform. Also used to compute UI jitter and transition jitter.

- eyecphy

  Constructs a self-triggered C-Phy eye diagram from three input waveforms.

- eyesetmask

  Adds a mask to an eye diagram.

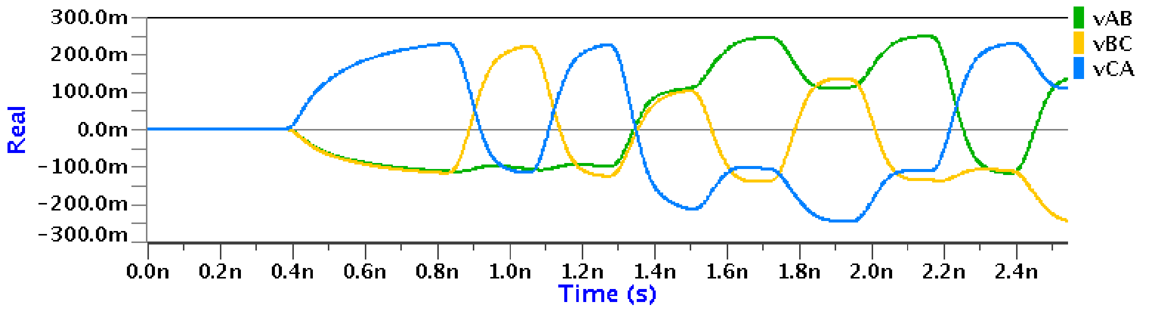
- eyewithtrigger

  Constructs a triggered eye diagram from a waveform.

Trigger

The Waveform Calculator functions may be used to analyze and display C-Phy Eye Diagram waveforms. The following figure shows an example where at least one differential line crosses zero at each symbol transition. At most the three differential lines cross 0. The trigger for the eye diagram is the first zero crossing at each symbol transition.

Figure 5-26. Differential Lines and Zero Level



For each differential line:

- The minimum of the risetime and falltime is computed.
- All zero crossings are computed.

The maximum of the minimum risetimes and falltimes is a validity-threshold:

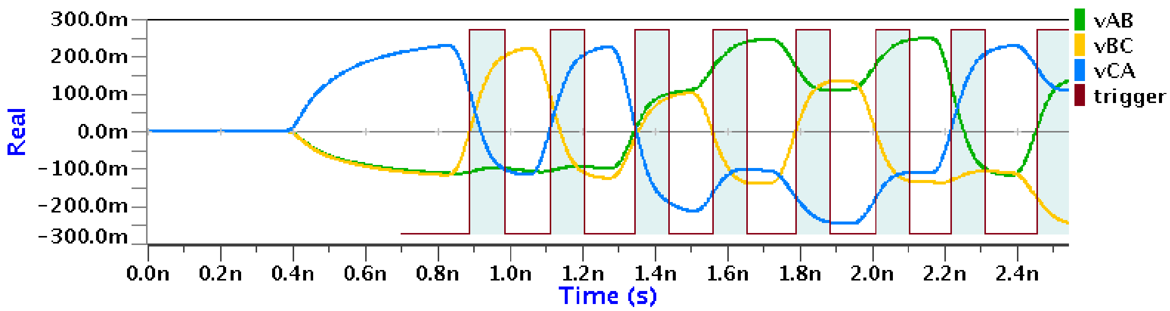
- All zero crossings with time-delta less than the validity-threshold are discarded.
- In Figure 5-26, before 0.5ns: waveforms are around 0V, but zero crossings in this region are not used to build the eye.

For each differential line individually:

- UI_{min}, the minimum delta-time between two crossings, is computed.
- The minimum of these three minimum delta-times (one per differential lines), minus a tolerance (default is 20%), is a transition-threshold.

All valid zero-crossings from the three lines are merged and sorted. The resulting triggers are zero crossings, from time min to time max, whose distance with previous zero crossing is more than the transition-threshold, as shown in the following figure:

Figure 5-27. Differential Lines, Zero Level, and Trigger (Rising Edges)



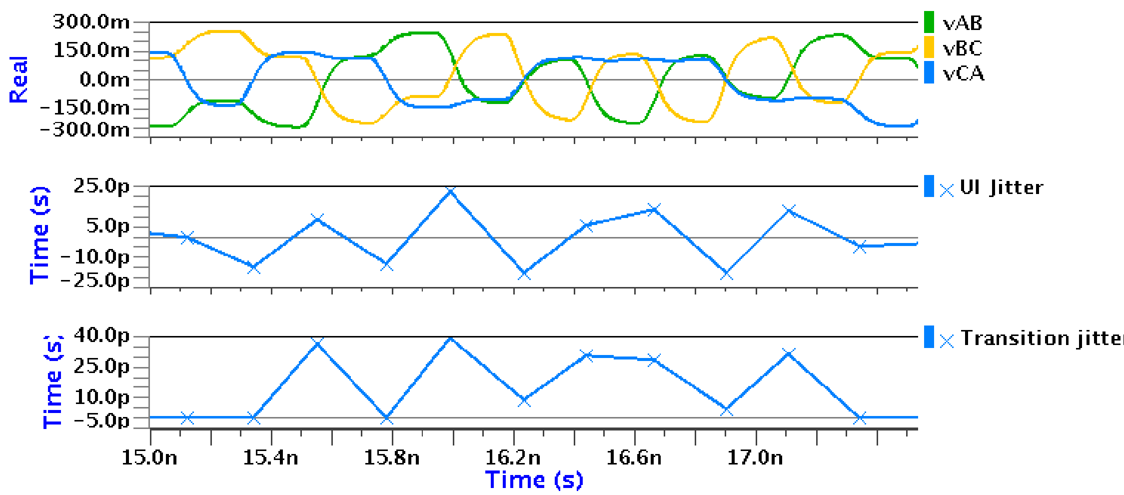
Because the eye-diagram is right aligned, the time range of the crossing:

- Is from one $UI_{average}$ before the first trigger; to display data before the first crossing.
- Is to the last trigger. There is no trigger after to anchor the remaining of the waveforms.

Jitter

The next figure shows the differential lines, UI jitter and transition jitter for a C-Phy waveform.

Figure 5-28. Differential Lines, UI jitter and Transition Jitter



UI Jitter

The UI jitter is the jitter of the period of the trigger. For each time $t_{(i)}$ of the trigger rising edge:

$$UI\_jitter_{(i)} = (t_{(i+1)} - t_{(i)}) - UI_{average}$$

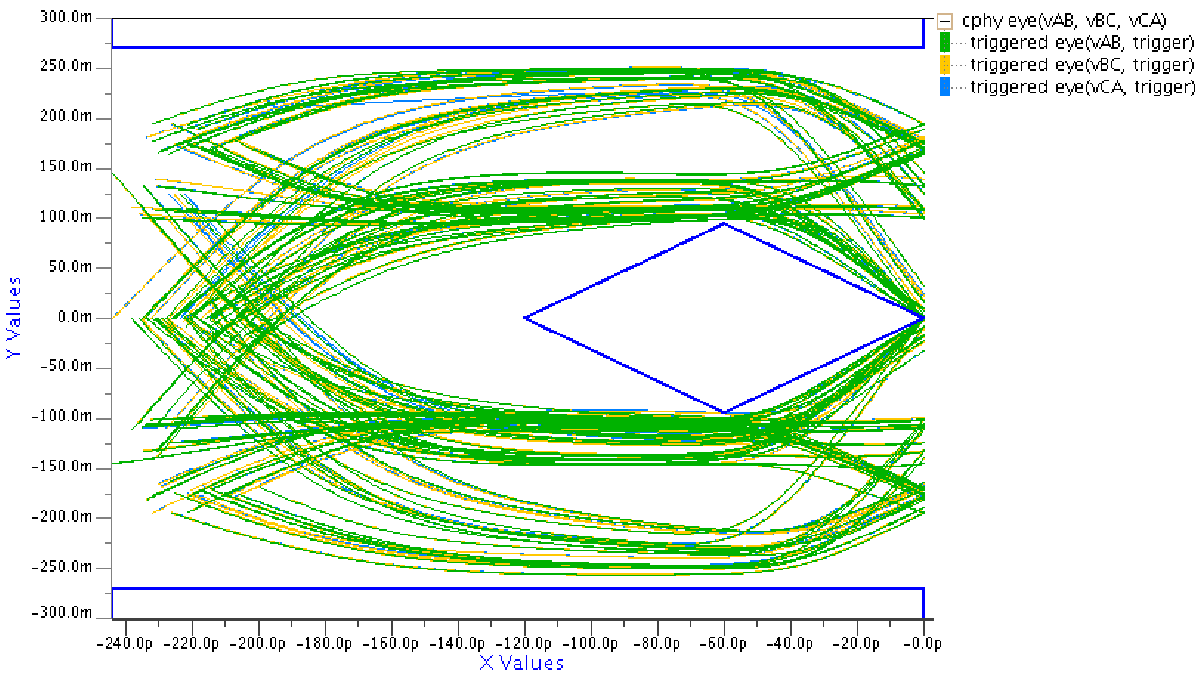
Transition Jitter

Transition jitter is the delay, for a symbol transition, from the first zero-crossing to the last zero-crossing. The transition jitter of a single zero crossing is always 0. The transition jitter is always positive.

Eye Diagram

The C-Phy eye-diagram is made of the three differential lines, based on the trigger, right aligned, and superposed. It is triggered by the first 0 crossing at each transition, as shown in the following figure:

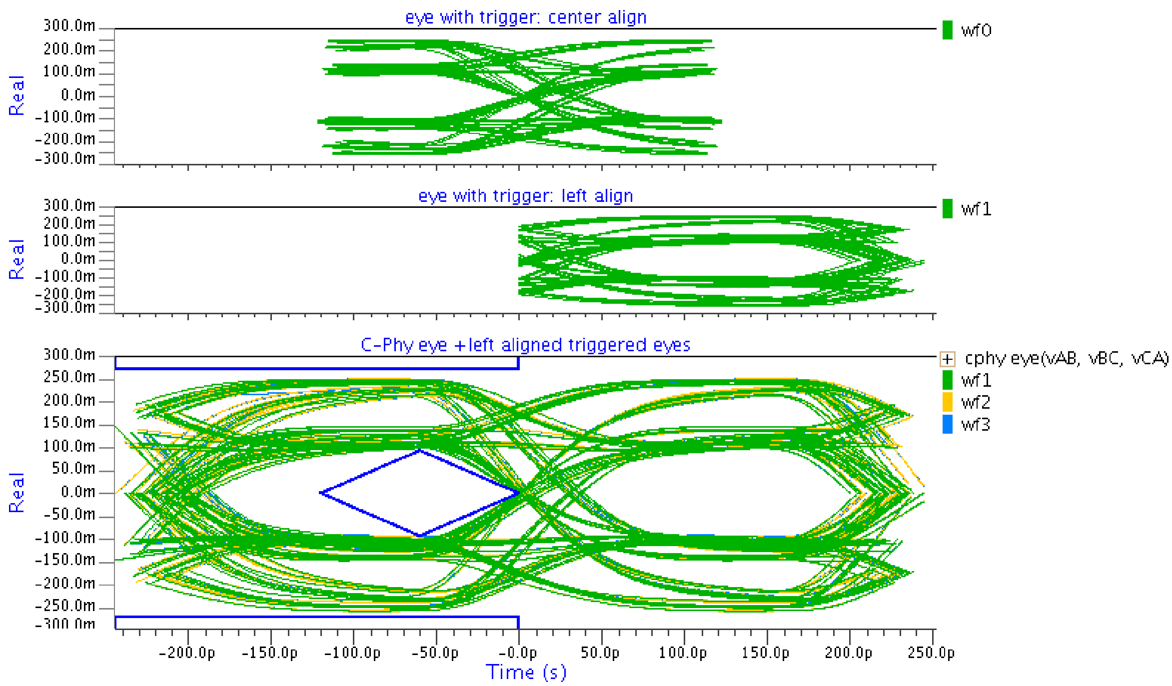
Figure 5-29. C-Phy Eye Diagram with Eye Mask



The domain range (x axis) is right-aligned with the trigger: all data are negative for a right-aligned eye.

General triggered eye diagrams can be left aligned or center aligned. C-Phy eye diagrams are always right-aligned. The next figure shows an example of C-Phy eye diagram (right aligned) with left-aligned eye diagrams made from the same differential lines and C-Phy trigger.

Figure 5-30. C-Phy Eye Diagram and Triggered Eyes with Other Alignments



Eye Mask

A mask on a C-Phy eye diagram is always right-aligned. Example masks are show in the following table:

Table 5-3. C-Phy Eye Mask

Diamond Shape	Hexagon Shape
{MASK_FILE} {VERSION = 2.0} {MASK = CPhy-Diamond (PTOPMAX = .270) (PBASEMIN = -.270) (50;0) (75;.1) (99.99;0) (75;-.1) }	{MASK_FILE} {VERSION = 2.0} {MASK = CPhy-Hexagon (PTOPMAX = .270) (PBASEMIN = -.270) (25;0) (50;.1) (75;.1) (99.99;0) (75;-.1) (50;-.1) }

The masks have Y values min and max at -1 and 1. These example masks require the automatic Y fit option.

You can also build your own eye-masks based on industry standards.

Note

A CPhy.mask file may be loaded using menu **Tools > Eye Diagram** with Eye Type set to "C-Phy", select Apply Mask, click the **Eye Mask** icon to open the Eye Mask Dialog Box. Then use the **Open Folder** icon to load a *CPhy.mask* file from disk.

Alternatively, add the following in *$HOME/settings.properties* the following line:
userEyeMaskFiles=<path to the directory>/CPhy.mask

Working With Smith Charts

A Smith Chart displays a sequence of impedance (Z-parameters), admittance (Y-parameters), scatter parameter (S-parameters) or reflection coefficient data, plotted as curves on a grid. The Smith Chart enables all possible impedances to be found on the domain of existence of the reflection coefficient.

Mathematically, the Smith Chart represents the following relationship for all values of Z (impedance) in the reflection coefficient plane, also called the G plane:

$$\Gamma = (Z - Z_O) / (Z + Z_O)$$

Z_O represents the "characteristic impedance" of a transmission line. If you define normalized impedance as z, then the relationship is as follows:

$$z = Z / Z_O = (R + jX) / Z_O = r + jx$$

which translates to the form:

$$\Gamma = (z - {}_1) / (z + 1)$$

With a Smith Chart, you can plot impedance values using curves and then read reflection-coefficient values from the grid.

Scattering Parameters (S-Parameters) **219**
Creating a Smith Chart **220**
Impedance and Admittance Displays **220**
Smith Chart and Polar Displays **223**
Cursors in the Smith Chart **224**
Circles in the Smith Chart **226**
Changing the Visibility of Elements **228**

Scattering Parameters (S-Parameters)

When the amplitude and phase of an incoming wave on a transmission line moves through a circuit, its energy "scatters" and is partitioned between all outgoing waves on all other transmission lines connected to the circuit. Scattering parameters (also known as S-parameters) are the fixed properties of the circuit which describe how the energy couples between each pair of ports or transmission lines connected to the circuit. S-parameters are plotted as curves on a Smith Chart grid.

Creating a Smith Chart

You can create a Smith Chart for both frequency-domain waveforms and complex-valued frequency-domain waveforms.

Procedure

1. For frequency-domain waveforms, right-click the waveform name in the Waveform List panel and choose **Plot as** > **smith_chart**.
2. For complex-valued frequency-domain waveforms, select **Transformations** > **smith_chart** from the Waveform Popup Menu. If a waveform is not an S11 or S22 parameter waveform, it is displayed as a complex-plane plot over a Smith Chart grid. By default, S11 and S22 parameters (scattering parameters) are displayed in a Smith Chart when they are brought up for viewing.

Related Topics

Working With Smith Charts

Impedance and Admittance Displays

You can change the coordinates, switching between displaying by Impedance (along the Z-parameter axis) or Admittance (along the Y-parameter axis). You can also display both impedance and admittance overlaid on a single chart (ImpedAdmit).

Use the following procedure to change between impedance and admittance displays:

Procedure

1. Right-click the row containing the Smith Chart.

   The Row Popup Menu opens.
2. Choose **Smith Chart** > **Impedance** or **Smith Chart** > **Admittance** or **Smith Chart** > **Both**.

Figure 5-31. Smith Chart Impedance Display

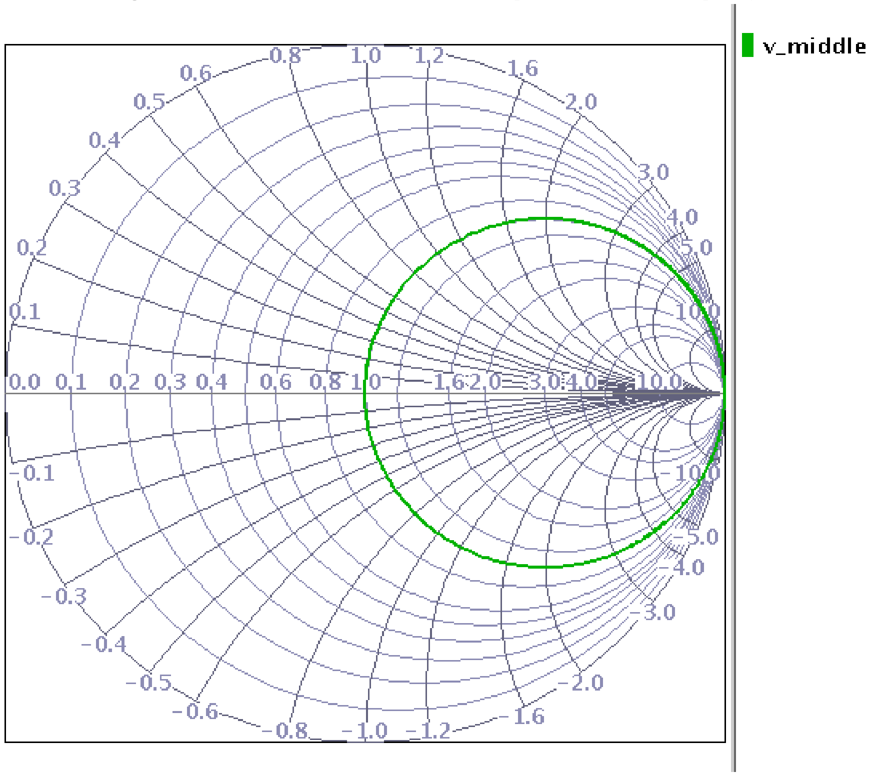


Figure 5-32. Smith Chart Admittance Display

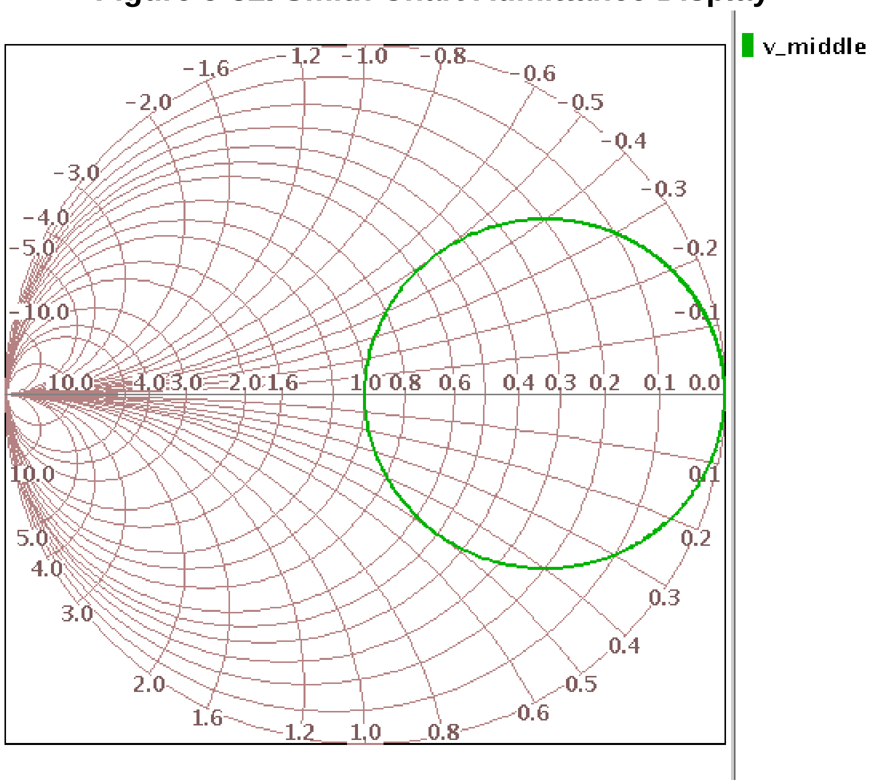
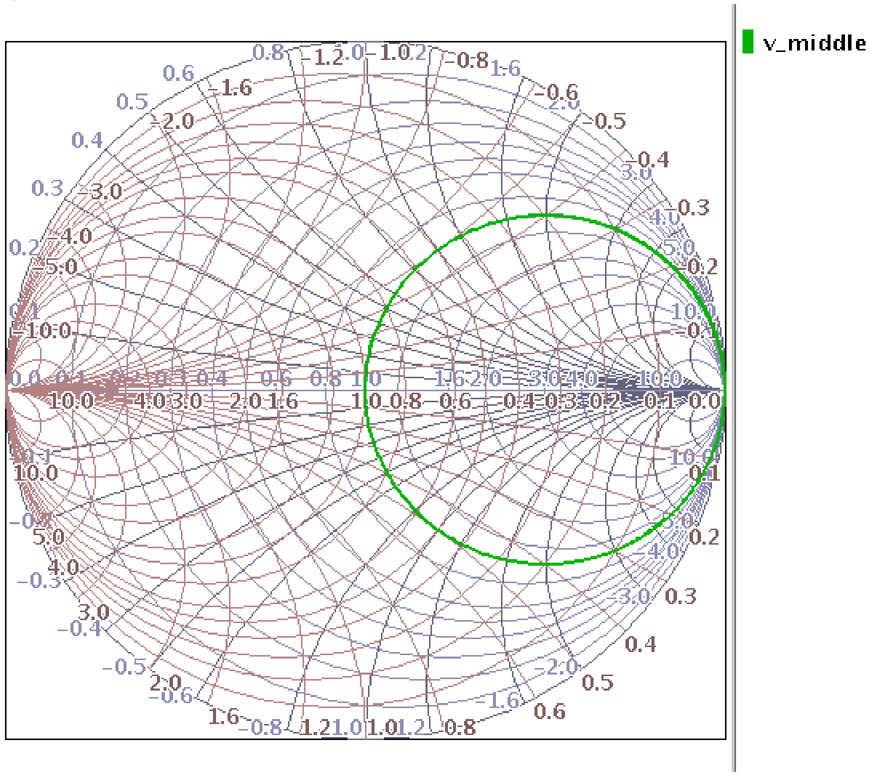


Figure 5-33. Smith Chart Impedance and Admittance Display



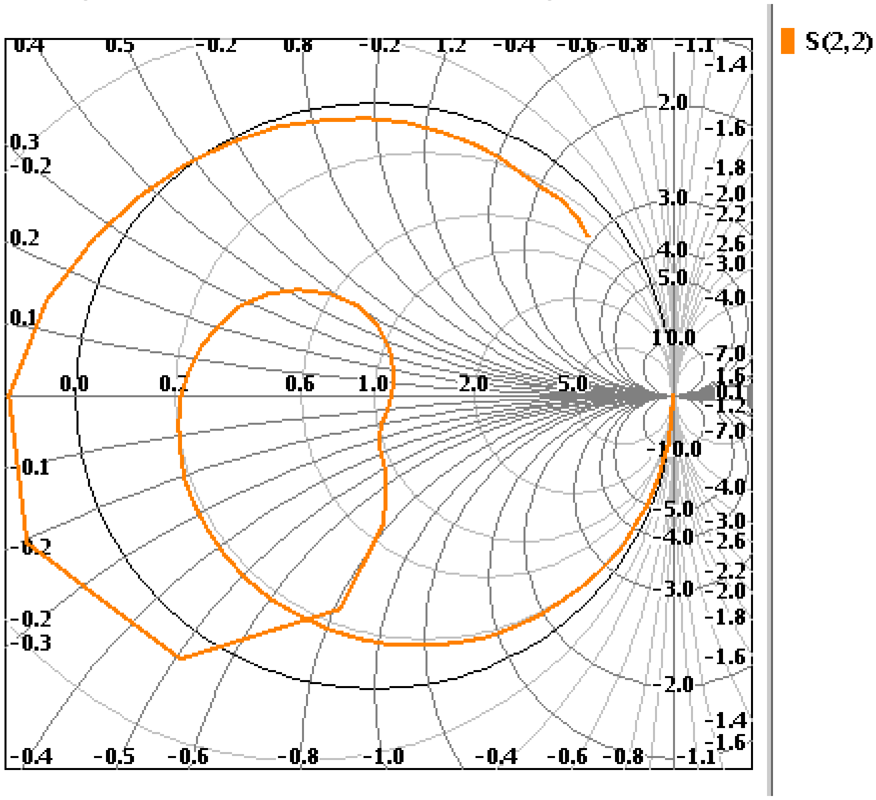
3. Ordinarily, the Smith Chart displays only positive real values. If, however, values of S11- or S22-parameter waveforms extend beyond the reaches of this display, the Smith Chart is automatically extended to display values outside this range. Maximum negative real values:

    - -0.8 for the left part of the chart
    - -1.2 for the right part of the chart

4. Maximum imaginary values:

    - 0.2 for the top part of the chart
    - -0.2 for the bottom part of the chart

Figure 5-34. Smith Chart with Negative Real Values



Related Topics

[Working With Smith Charts]

[Smith Chart and Polar Displays]

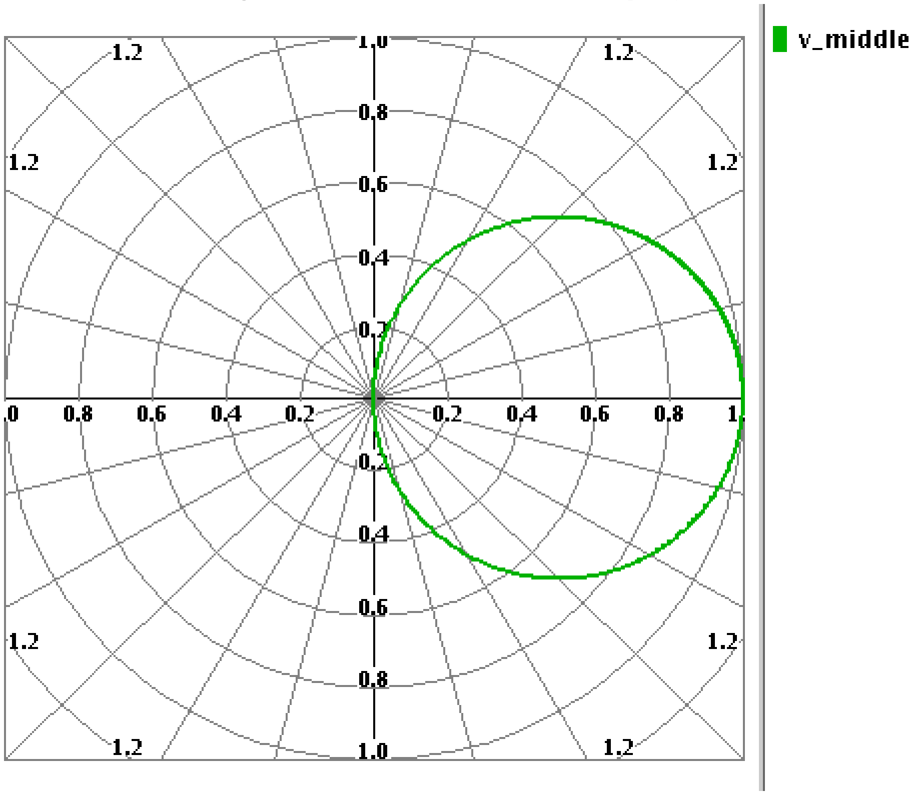
Smith Chart and Polar Displays

You can switch between a Smith Chart display and a polar display.

Procedure

1. Right-click an element on the chart.
2. To switch to a polar display select **Transformations** > **polar_chart** from the Waveform Popup Menu.
3. To switch back select or **Transformations** > **smith_chart**.

Figure 5-35. Polar Chart Display



Related Topics

Working With Smith Charts

Impedance and Admittance Displays

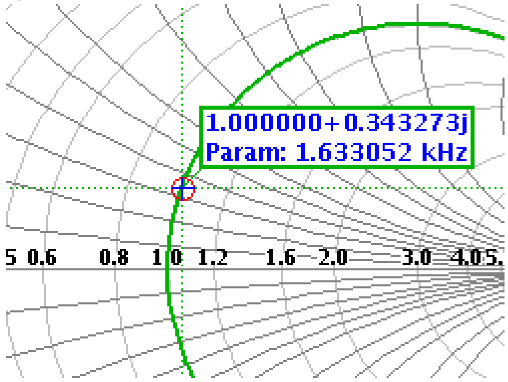
Cursors in the Smith Chart

You can add a cursor to a Smith Chart, change the data value, and select which values to display.

Procedure

1. To add a cursor to a Smith Chart select **Cursor** > **Add** from the main menu or click the **Add Cursor** icon in the toolbar. The Smith Chart does not use bar cursors. Instead, cursors are represented by marks on the plotted waves, and the circles that correspond to that point on the graph.

Figure 5-36. Cursor in a Smith Chart



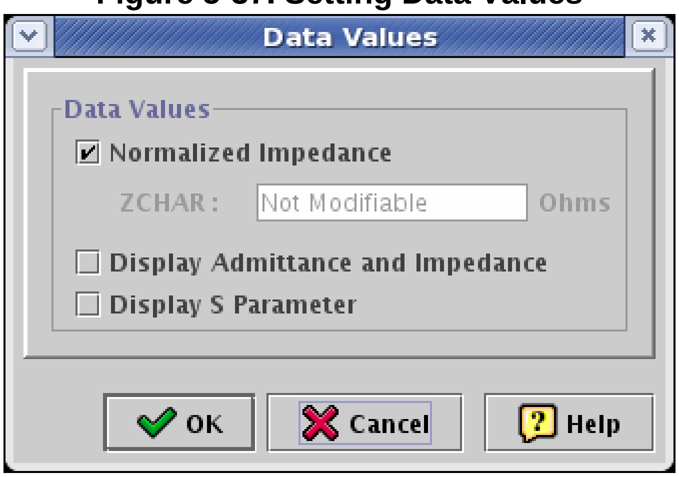
2. Drag the cursor along the path of any curve to determine the impedance at that point. Change the data value to take into account the characteristic impedance. The F (frequency) and Z values are displayed. The S parameters can be displayed by selecting the appropriate option from the Data Values dialog box.

Note

You can automatically snap the cursor to the nearest data point. Choose **Edit** > **Options** and select **RF** from the EZwave Preferences list and then check Snap to Data.

3. Use the following procedure to change the data value and select which values to display:
4. Add a cursor to a Smith Chart.
5. Right-click the value flag.
6. Choose **Data Values** > **Set** from the Cursor Value Popup Menu.

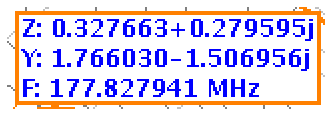
Figure 5-37. Setting Data Values



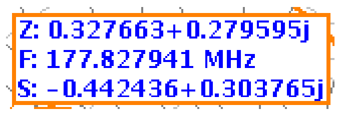
- Select Normalized Impedance, Normalized Admittance or Normalized Admittance/ Impedance for the displayed value to be normalized. This option varies depending on whether impedance, admittance, or both are displayed in the value flag.

If the value is not normalized, the display in the value flag reflects this. If this option is not selected, type a ZCHAR value, in ohms.

- Select Display Admittance and Impedance to display both admittance and impedance values in the value flag.



- Select Display S Parameter to display the S parameter in the value flag.



Note

If more than one waveform is displayed on a Smith Chart, dragging the cursor along one wave causes the cursor to move along the others, as well.

Related Topics

Working With Smith Charts

Scattering Parameters (S-Parameters)

Impedance and Admittance Displays

Circles in the Smith Chart

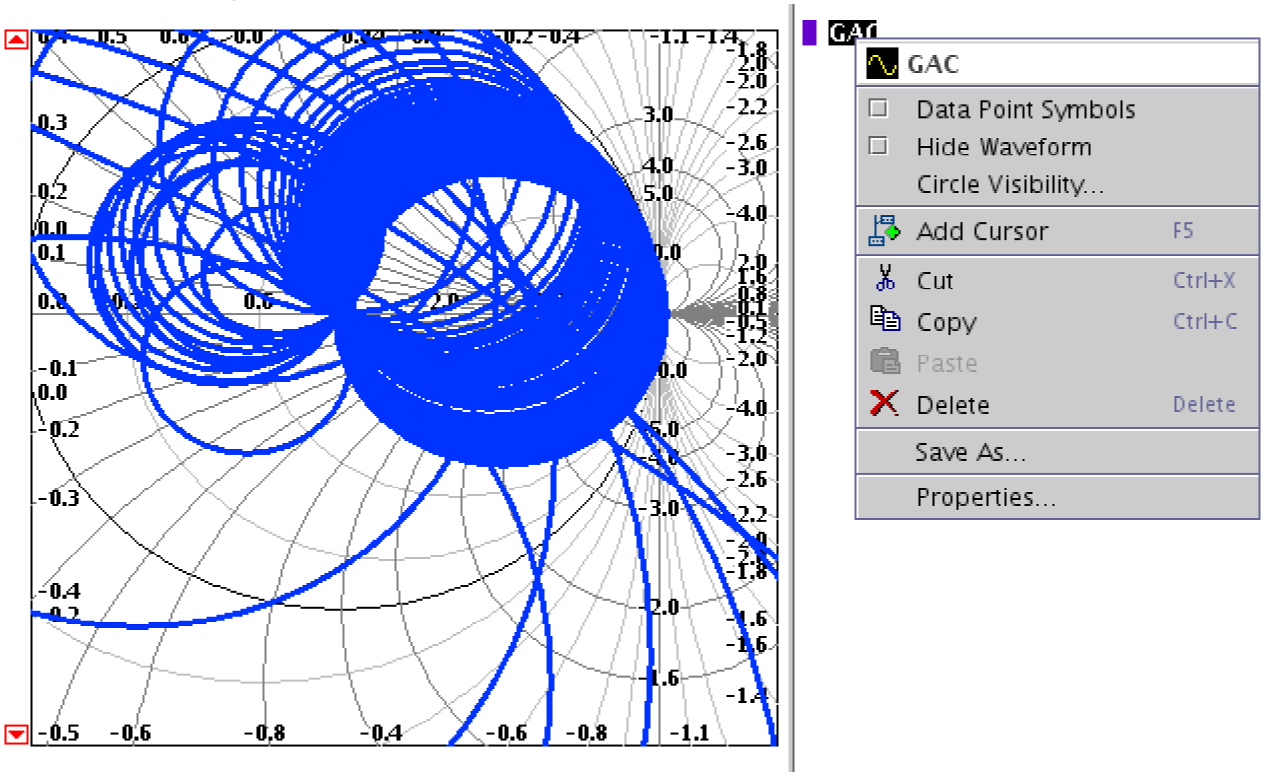
You can plot Constant circles and Stability circles on a Smith Chart. Usually, a circle plot consists of a family of circles.

Use the following procedure to invoke the Circle Visibility dialog box which enables you to set which circles are visible in a circle plot:

Procedure

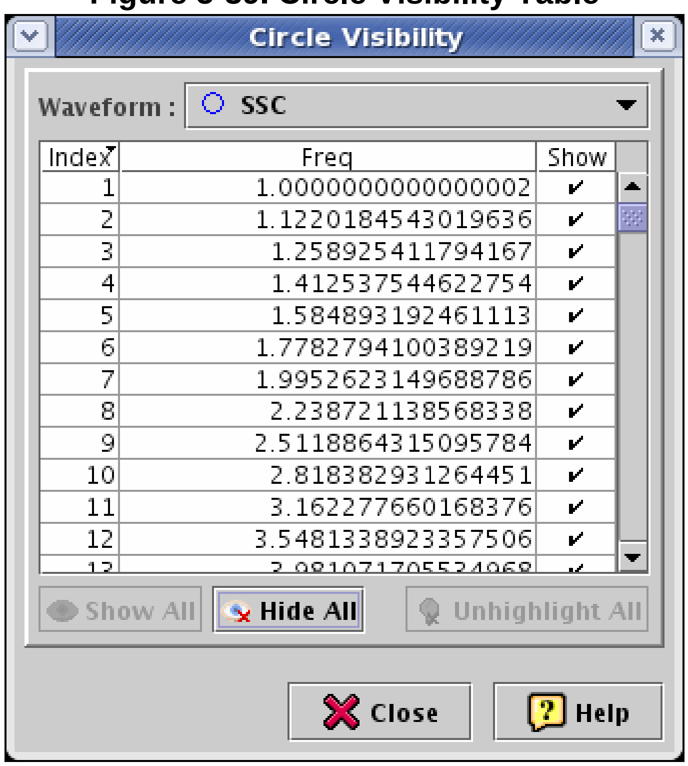
1. Right-click a Smith Chart circle plot or circle plot name in a graph window.
2. From the Waveform Popup Menu, select **Circle Visibility**.

Figure 5-38. Multiple-Circle Plot and Circle Visibility Option



Results

Figure 5-39. Circle Visibility Table



The Circle Visibility Table displays the following information about individual elements of a circle plot.

- Index - A sequential number assigned to each element of the circle plot. The first element is assigned the number one. By default, the numbers are listed in ascending order. To reverse this order, click the small triangle.
- Frequency - The frequency column shows the frequency value of the element. To invert the order of this column, click the small triangle.

Changing the Visibility of Elements

You can select which Smith Chart elements are visible in the graph window.

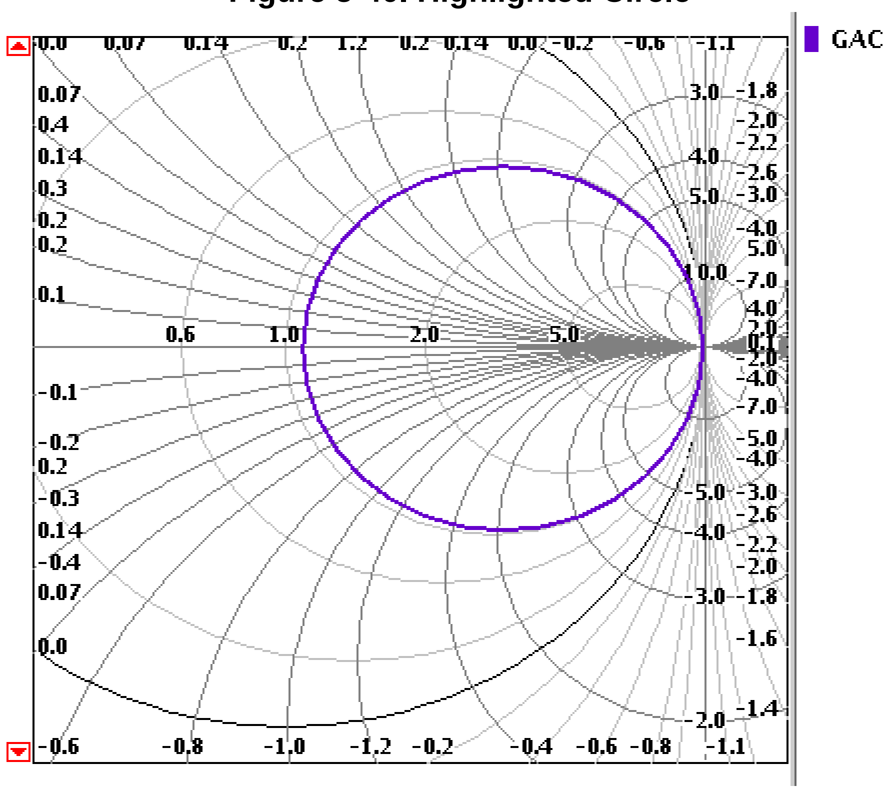
Procedure

1. Click the box or boxes in the Show column corresponding to the element number or numbers.
2. If multiple elements, you can group them together at either the top or bottom of the table by clicking the small arrow at the top of the column.

Results

The following figure shows the results after selecting a single circle to highlight.

Figure 5-40. Highlighted Circle



Related Topics

Working With Smith Charts

Comparing Waveforms

The EZwave Waveform Compare Tool enables waveforms from a reference and a new simulation to be compared. It can report the set of differences graphically or in report files.

This tool can be accessed through the graphical user interface or through a set of Tcl commands (see "Waveform Comparison With Tcl Examples" on page 1213).

To watch a tutorial that shows you how to use the Waveform Compare Tool, using an example that compares the results of periodic steady state analysis with time-domain analysis results, see the video:

This section describes the following tasks relating to waveform comparison and the waveform comparison algorithm:

Support for Different Types of Waveform 231
Using the Waveform Compare Wizard 231
Manually Comparing Waveforms 236
Starting a Comparison 236
Selecting Waveforms for Comparison 238
Running a Comparison 239
Viewing Waveform Comparison Results. 240
Stepping Through Differences With a Cursor 240
Displaying the Tolerance Tube for Analog Comparisons 241
Waveform Comparison Reports. 244
Generating a Waveform Comparison Report 244
Digital to Digital Comparison Reports 244
Analog to Analog Comparison Reports 245
Spectral Comparison Report. 245
Viewing and Saving Comparison Rules. 246
Saving a Comparison Session 246
Setting Comparison Options. 248
Setting General Comparison Options 248
Setting Comparison Method Options 250
Setting Tolerance Options 253

Setting Conversion Options 257
Analog Waveform Comparison Algorithm 258
wreal Waveform Comparison. 259

Support for Different Types of Waveform

You can apply the Waveform Compare Tool to digital, analog and mixed-signal simulation results.

Supported waveforms:

- Verilog and VHDL digital waveforms.
- Analog continuous, wreal and real waveforms.
- Enum, char and string waveforms.

For buses, records and compound waveforms:

- Digital buses are compared directly providing they have the same number of bits. If this is not the case, EZwave's behavior cannot be guaranteed.
- Analog buses and records are exploded and bits compared individually. Bits are matched by index rather than by name. If the number of bits differs between the test and the reference object, bits are compared until the smaller number is reached. An error message is then displayed.
- Compound waveforms are exploded and elements compared individually. Elements are matched by index rather than by name. If the number of elements differs between the test and the reference waveform, elements are compared until the smaller number is reached. An error message is then displayed.

Note

Assertions, SOA and complex waveforms are not supported in the Waveform Comparison Tool.

Related Topics

Using the Waveform Compare Wizard

Manually Comparing Waveforms

Using the Waveform Compare Wizard

You can use the Waveform Compare Wizard to help step you through each stage of the waveform comparison process.

Note

The waveform compare tool automatically applies a set of default settings to the comparison. To change these defaults refer to “Setting Comparison Options” on page 248.

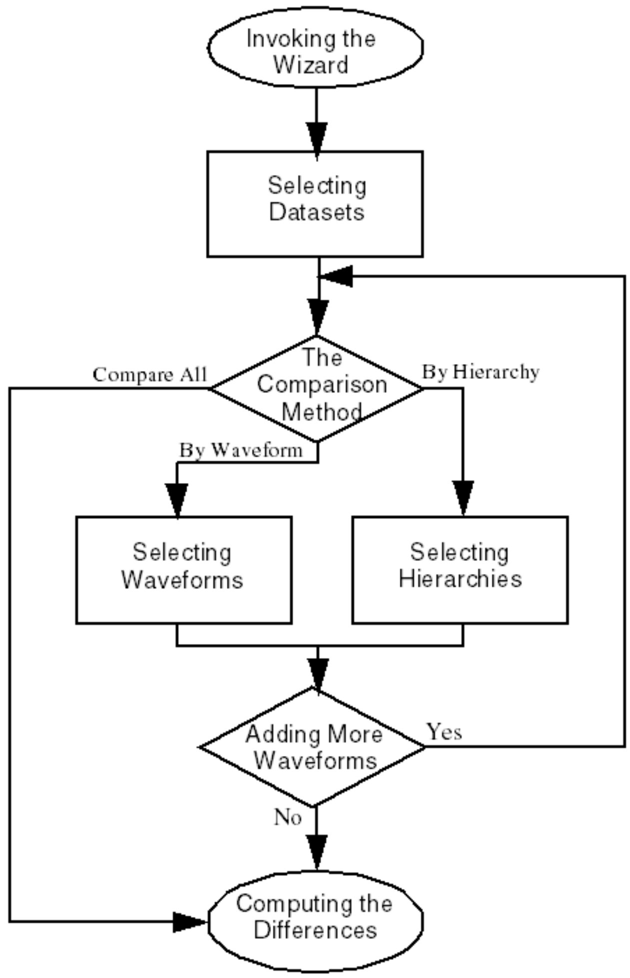
To watch a tutorial that shows you how to use the Waveform Compare Tool, using an example that compares the results of periodic steady state analysis with time-domain analysis results, see the video:

Note

For details on error messages that may be encountered during a waveform comparison, see “Troubleshooting Waveform Comparison” on page 1277.

The process flow is shown in the following figure:

Figure 5-41. The Waveform Compare Wizard Process Flow



Procedure

Tip

The Waveform Compare Wizard dialog box can only open *.wdb* files. For other database file formats such as *.wlf*, first open the required databases by selecting **File** > **Open** from the EZwave main menu. You can then choose an opened database from the dropdown list using the Waveform Compare Wizard dialog box, as described here.

1. From the EZwave main menu, select **Tools** > **Waveform Campare** > **Comparison Wizard**. This opens the first page of the Waveform Compare Wizard.
2. Specify the reference dataset in the Choose Reference Dataset From List or Disk field. Use the dropdown list to choose from recently opened databases, or click the folder icon to use a file browser to select the reference database.

3. Specify the test dataset in the Choose Test Dataset From List or Disk field.

   By default, the current open database is used as the test dataset (indicated by the Use Current Database (*database_name*) option). Specify a different database if required.

   Click **Next** to move on to the Comparison Method page of the Waveform Compare Wizard.

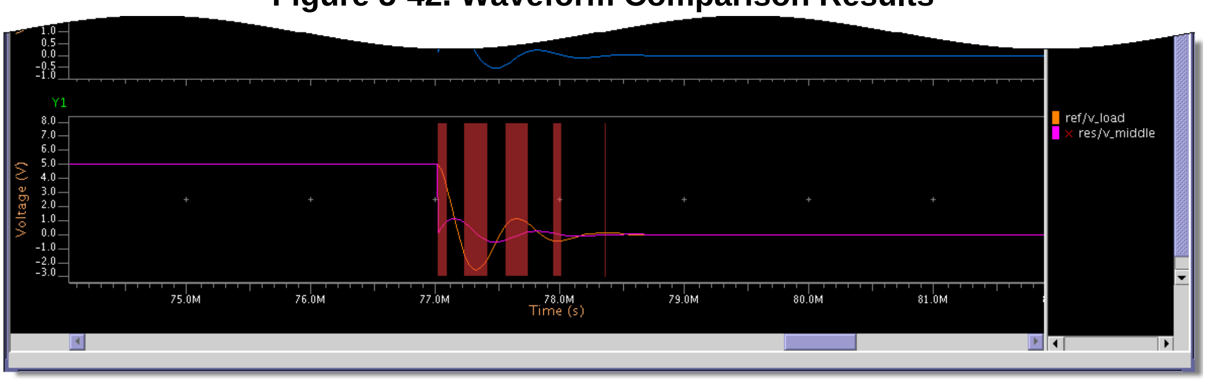
4. Select from the following three comparison methods:

   - Compare All Waveforms

     All available waveforms in the test dataset are compared against the same waveforms in the reference dataset. Select this option and click Next to begin the comparison.

     The results of the waveform comparison are displayed in the graph window. Waveforms that contain differences from the reference waveform are marked by a red X by their name and those difference portions of the waveform are highlighted.

Figure 5-42. Waveform Comparison Results



   - Specify Comparison By Waveform

     Select this option to specify the waveforms to compare on the Select Waveforms Dialog Box:

     i. Click Add on the Add Waveforms From Reference Database side of the window. This opens the Add Waveforms dialog box.

     ii. Select one or more waveforms and click **OK**. The selected waveforms are added to the Reference side of the Waveform Compare Wizard.

Note

For databases loaded from *.fsdb* files, only loaded waveforms are displayed in the **Waveform Compare By Waveform** > **Add Waveform** dialog box. Loaded waveforms are waveforms that have been displayed once or waveforms located at a hierarchy level that has been selected once in the Waveform List Panel. Refer to “Loading .fsdb and .tr0 Files” on page 1272.

iii. By default, the Take Corresponding Waveforms in Test Database option is selected, indicating that the tool will search the test dataset for waveforms with names matching those in the reference dataset.

iv. To compare waveforms with different names, uncheck Take Corresponding Waveforms in Test Database, and click Add to add test waveforms.

When comparing waveforms with different names, the first waveform in the list of reference waveforms is compared with the first waveform in the list of test waveforms, and so on, regardless of their names.

- Specify Comparison By Hierarchy Level

  Select this to specify a hierarchy level for comparison using the Select Hierarchy Dialog Box:

  i. Select the reference waveform hierarchy from the Waveform List panel and click the Add Selected Hierarchy  icon. This populates the field with the selected waveform hierarchy.

  ii. By default, the comparison searches the same hierarchy level name in the test database. To specify a different hierarchy level, select Specify a different name for test Hierarchy Level and choose the test hierarchy level.

  iii. In the Compare Waveforms of Type area, select the types of waveforms to include in the comparison and whether to search the hierarchies recursively.

5. After defining the waveform or hierarchy for comparison, you can select more waveforms by specifying **Yes** on the **Would you like to add more waveforms to the comparison** page of the Waveform Import Wizard.

6. Click **Next** and **Finish** to begin computation of the waveform differences.

Related Topics

Waveform Comparison Reports

Selecting Waveforms for Comparison

Manually Comparing Waveforms

You can manually compare waveforms without the use of the Waveform Compare Wizard.

Note

The waveform compare tool automatically applies a set of default settings to the comparison. By default, the waveform compare tool looks for waveforms with the same name in the reference and test simulation result databases. To change these defaults refer to "Setting Comparison Options" on page 248.

Note

For details on error messages that may be encountered during a waveform comparison, see "Troubleshooting Waveform Comparison" on page 1277.

Use the following steps:

Starting a Comparison **236**
Selecting Waveforms for Comparison **238**
Compare All Waveforms 238
Compare by Waveforms 238
Compare by Hierarchy 239
Running a Comparison **239**

Starting a Comparison

You can manually compare waveforms using the Waveform Compare Tool.

Use the following procedure to begin a manual waveform comparison:

Procedure

Tip

The Select Datasets dialog box can only open *.wdb* files. For other database file formats such as *.wlf*, first open the required databases by selecting **File** > **Open** from the EZwave main menu. You can then choose an opened database from the dropdown list using the Select Datasets dialog box, as described here.

1. Choose **Tools** > **Waveform Compare** > **Start Comparison**. This opens the Select Datasets dialog box.

2. Select the reference dataset in the Choose Reference Dataset From List or Disk field. Use the dropdown list to choose from recently opened databases, or click the folder icon to use a file browser to select the reference database.

3. Select the test dataset in the Choose Test Dataset From List or Disk field. By default, the currently open database is used as the test dataset (indicated by the Use Current Database *(database_name)* option). To specify a different database than the current one, use the dropdown list or the file browser to choose the test database.
4. Click **OK** to close the Select Datasets dialog box.

Related Topics

Selecting Waveforms for Comparison

Setting Comparison Options

Troubleshooting Waveform Comparison

Selecting Waveforms for Comparison

Once you have started a waveform comparison, you can begin to add waveforms.

You can choose the waveforms for comparison with the following three methods.

Compare All Waveforms 238
Compare by Waveforms 238
Compare by Hierarchy 239

Compare All Waveforms

You can specify the comparison of all available waveforms in the test dataset against the same waveforms in the reference dataset.

Procedure

Choose **Tools** > **Waveform Compare** > **Add** > **Compare All Waveforms**.

Compare by Waveforms

You can specify which waveforms are used for the comparison.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Add** > **Compare by Waveforms**.

   This opens the Select Waveforms dialog box.

2. Click Add on the Add Waveforms From Reference Database side of the window. This opens the Add Waveforms dialog box.

3. In the Add Waveforms dialog box, select one or more waveforms.

4. Click **OK**. The selected waveforms are added to the Reference side of the Waveform Compare Tool.

5. By default, the “Take Corresponding Waveforms in Test Database” option is selected, indicating that the Waveform Compare Tool will search for waveforms in the test dataset whose names match the specified reference waveforms.

   To compare waveforms with different names, uncheck “Take Corresponding Waveforms in Test Database”, and add test waveforms in the same way reference waveforms were added.

   When comparing waveforms with different names, the first waveform defined in the list of reference waveforms is compared with the first waveform defined in the list of test waveforms, and so on, regardless of their names.

6. Click **OK** to close the Select Waveforms dialog box.

Related Topics

Select Waveforms Dialog Box

Compare by Hierarchy

You can specify which waveforms are used for the comparison by hierarchy.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Add** > **Compare by Hierarchy**.

   The Select Hierarchy dialog box opens.

2. Select the reference waveform hierarchy from the EZwave waveform list.

3. In the Reference Hierarchy Level field, click the **Add Selected Hierarchy**  icon. This populates the field with the selected waveform hierarchy.

4. By default, the same hierarchy level name is searched in the test database for the comparison. If you want to specify a different hierarchy level for comparison, select "Specify a different name for test Hierarchy Level". You can then choose the test hierarchy level for comparison.

5. In the "Compare Waveforms of Type" area, select the types of waveforms to include in the comparison and whether to search the hierarchies recursively.

6. Click **OK** to close the Select Hierarchy dialog box.

Related Topics

Select Hierarchy Dialog Box

Running a Comparison

Once you have specified the waveforms you want for comparison, you can run the comparison.

Procedure

1. Select **Tools** > **Waveform Compare** > **Run Comparison**.

2. You can continue to add waveforms and run comparisons until you select **Tools** > **Waveform Compare** > **End Comparison**.

Related Topics

Select Hierarchy Dialog Box

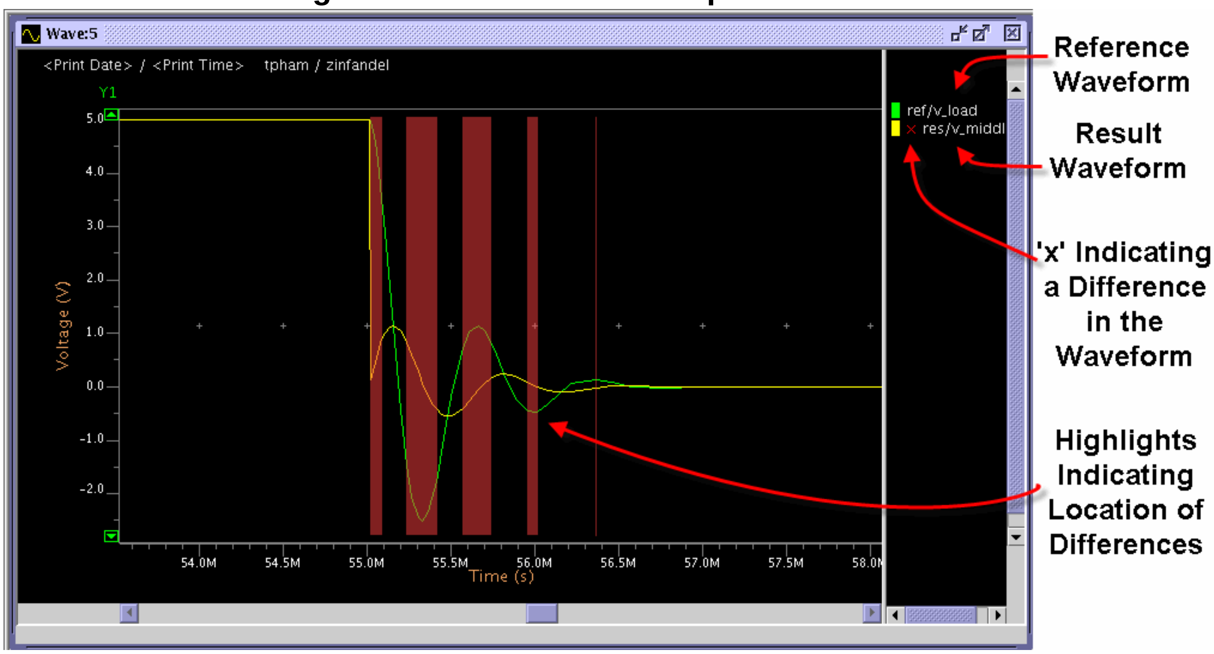
Select Waveforms Dialog Box

Viewing Waveform Comparison Results

After waveform comparison, the results are displayed in a graph window. Waveforms that contain differences with respect to the reference waveform are marked by a red 'x' by its name.

Those difference portions of the waveform are highlighted in the graph window:

Figure 5-43. Waveform Comparison Results



You can also generate a comparison report. See Waveform Comparison Reports.

Stepping Through Differences With a Cursor **240**
Displaying the Tolerance Tube for Analog Comparisons **241**

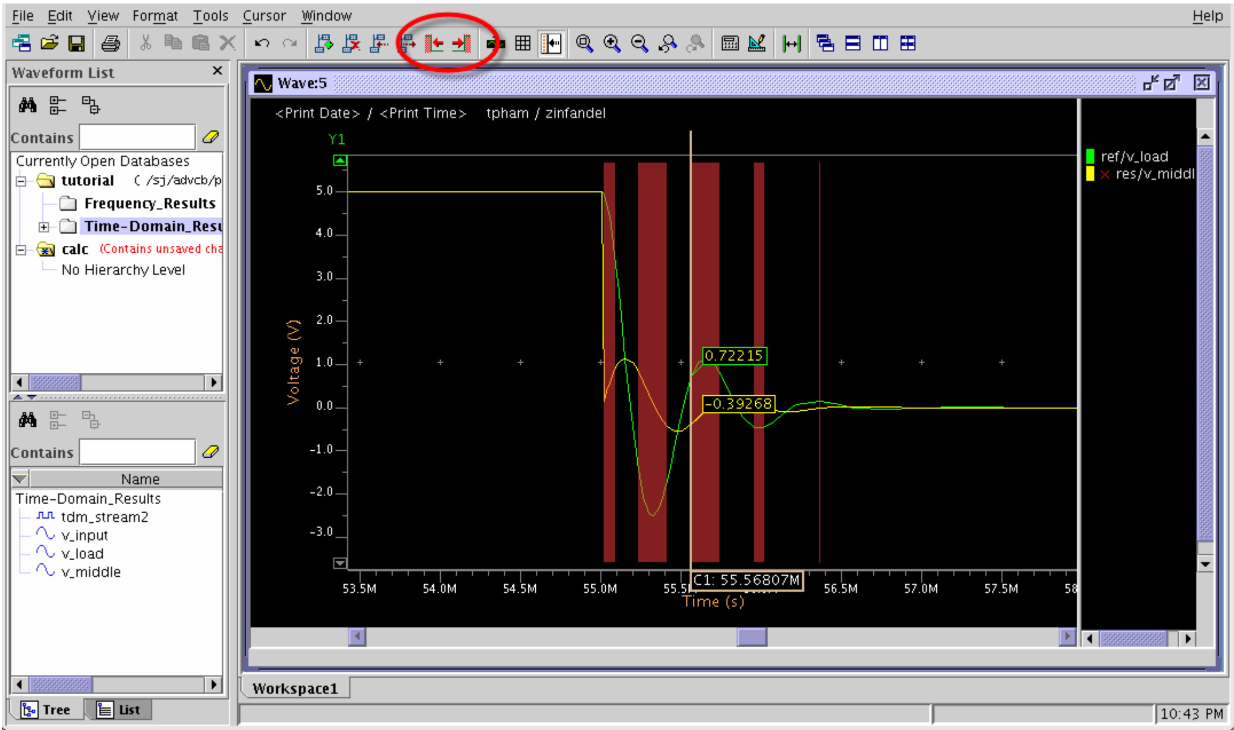
Stepping Through Differences With a Cursor

You can add a cursor to the waveform comparison results and easily step through each difference between the reference and result waveforms.

Procedure

1. Add a cursor by pressing **F5**. A cursor is added to the waveform comparison result graph window.
2. In the toolbar, click the **Move Cursor to Next Error** icon. The icon advances the cursor to the "next" difference between the reference and result waveforms.
3. To step the cursor to the "previous" difference between the reference and result waveforms, click the **Move Cursor to Previous Error** icon.

Figure 5-44. Stepping Through Waveform Differences With a Cursor



Related Topics

Displaying the Tolerance Tube for Analog Comparisons

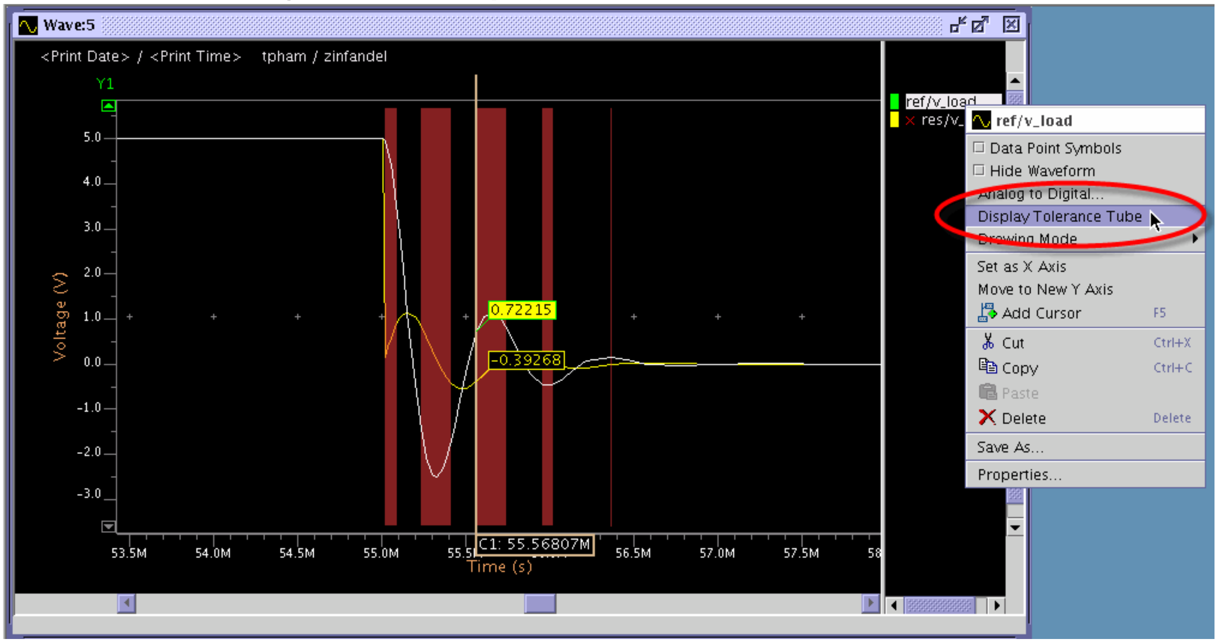
Displaying the Tolerance Tube for Analog Comparisons

The tolerance “tube” is the virtual area along the reference waveform that defines the limits where the result waveform can deviate from the reference waveform before being considered “different.”

For more information on analog waveform tolerances, see “Analog Waveform Comparison Algorithm” on page 258.

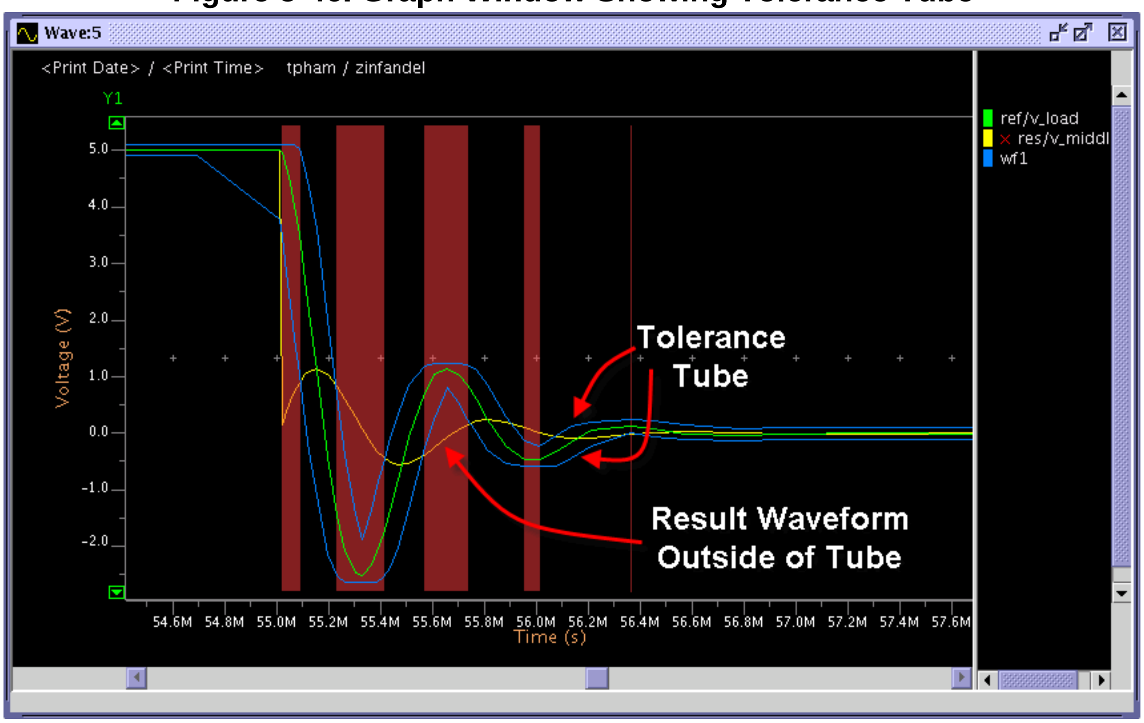
In the waveform comparison results, you can display the tolerance tube by right-clicking on one of the waveform names in the graph window that contains the comparison results and selecting **Display Tolerance Tube**.

Figure 5-45. Display Tolerance Tube Menu Item



In the following figure, the green waveform is the reference waveform. The two blue "waveforms" represent the tolerance tube. When the yellow result waveform goes "outside" of this tube is where it is "different" from the reference waveform, given the tolerances.

Figure 5-46. Graph Window Showing Tolerance Tube



Related Topics

Analog Waveform Comparison Algorithm

Waveform Comparison Reports

After you run a manual waveform comparison, you can generate a report of the differences in a text file.

Generating a Waveform Comparison Report 244
Digital to Digital Comparison Reports 244
Analog to Analog Comparison Reports 245
Spectral Comparison Report 245

Generating a Waveform Comparison Report

This topic describes how to generate a waveform comparison report using the Waveform Compare Tool.

Procedure

1. To generate a waveform comparison report, choose **Tools** > **Waveform Compare** > **Differences** > **Write Report**. This opens a file browser where you can choose the name and location of the report.
2. In the Save Comparison Report file browser, choose a location and a name for the report and click **Save**.

Digital to Digital Comparison Reports

For digital-digital reporting, the syntax is the same as the Questa SIM report files. The difference is that Questa SIM reports differences organized around time, but the EZwave reports are based on each waveform.

Example

The following is an example of a digital to digital comparison report:

```
Total differences = 100
Diff number 1, From time 2.000000e-8 s to time 4.000000e-8 s.
<vsimref>clk = '1'
<vsimres>clk = '0'
Diff number 2, From time 6.000000e-8 s to time 8.000000e-8 s.
<vsimref>clk = '1'
<vsimres>clk = '0'
Diff number 3, From time 1.200000e-7 s to time 1.400000e-7 s.
<vsimref>clk = '0'
<vsimres>clk = '1'
Diff number 4, From time 1.600000e-7 s to time 1.800000e-7 s.
<vsimref>clk = '0'
<vsimres>clk = '1'
...
```

Related Topics

Viewing Waveform Comparison Results

Waveform Comparison Reports

Analog to Analog Comparison Reports

For analog to analog comparisons, the report indicates the number of differences found, and for each difference, it indicates the beginning and end times where the waveforms differ. The report also indicates the time at which the maximum Y difference is encountered, and the Y values of reference and test waveforms.

Example

The following is an example of an analog to analog comparison report:

```
Total differences = 1
Diff number 1, From time 9.329983e-5 s to time 1.198995e-4 s.
Maximum difference at time 9.329983e-5 s : deltaY = 1.172866e-4 A
<REF>I(vv1) = -1.498311e-4
<RES>I(vv1) = -3.254448e-5
```

Related Topics

Viewing Waveform Comparison Results

Waveform Comparison Reports

Spectral Comparison Report

For spectral comparisons, the report indicates the three types of possible differences.

Description

|Possible comparison differences:

- A reference without a result with a matching frequency
- A result without a reference with a matching frequency
- A reference matching a result’s frequency but not close enough in amplitude. In this case, the comparison tracks the result closest to the reference and reports the distance outside the tolerance.

Example

The following is an example of a spectral comparison report:

```
Total differences between <refDB> and <resDB> = 3
Diff number 1
   <refDB> frequency 14.321MHz, magnitude 3.23 dB
   <resDB> frequency 14.320MHz, magnitude 4.51 dB (1.362 dB outside
amplitude tolerance)
Diff number 2
   <refDB> no matching frequency within tolerance
   <resDB> frequency 19.438MHz, magnitude 2.03 dB
Diff number 3
   <refDB> frequency 23.127MHz, magnitude 4.03 dB
   <resDB> no matching frequency within tolerance
```

Viewing and Saving Comparison Rules

When you set up a waveform comparison, the settings (or “rules”) you choose for the comparison can be viewed and saved in a Tcl file for later use.

Saving comparison rules only saves comparison options, clock definitions, and region and signal selections.

Procedure

1. To view and save the comparison rules for a session, start a waveform comparison session, either manually (see “Manually Comparing Waveforms” on page 236) or with the Waveform Compare Wizard (see “Using the Waveform Compare Wizard” on page 231).
2. Make your waveform comparison rules.
3. Before you end the comparison, select **Tools** > **Waveform Compare** > **Rules** > **Show**. The file that contains the current rules for the waveform comparison is displayed.
4. To save the comparison rules to a Tcl file, select **Tools** > **Waveform Compare** > **Rules** > **Save** to open the Save Comparison Rules dialog box.
5. Navigate to the directory where you want to save the rules Tcl file, give the file a name, and click **Save**.

Saving a Comparison Session

You can save a copy of the comparison session in a Tcl file for direct replay later.

Procedure

1. To save the session, start a waveform comparison session, either manually (see “Manually Comparing Waveforms” on page 236) or with the Waveform Compare Wizard (see “Using the Waveform Compare Wizard” on page 231).

2. Make your waveform comparison rules.
3. To save the comparison session to a Tcl file, select **Tools** > **Waveform Compare** > **Save Session** to open the Save Comparison Session dialog box.
4. Navigate to the directory where you want to save the rules Tcl file, give the file a name, and click **Save**. The file is called *session.tcl* by default.

Setting Comparison Options

You can set various comparison options for the Waveform Compare Tool.

Select **Tools** > **Waveform Compare** > **Options** to open the four-tabbed Comparison Options Dialog Box.

Use the four tabs to do the following:

Setting General Comparison Options **248**
Setting Comparison Method Options **250**
Continuous Comparison 250
Sampled Comparison 250
Clocked Comparison. 251
Setting Tolerance Options. **253**
Analog Comparison 253
Digital Comparison. 254
Mixed Comparison 255
Spectral Comparison. 255
Setting Conversion Options **257**
Digital Conversion 257
Analog to Digital Conversion. 257

Setting General Comparison Options

You can set global general options for Waveform Compare Tool.

Procedure

1. Select **Tools** > **Waveform Compare** > **Options** and select the **General Options** tab.
2. In the Comparison Limit Count fields, specify the maximum number of differences (Total Limit) and differences per signal (Per Signal Limit) allowed before the comparison terminates.
3. If you want to begin the comparison from a point other than the start time (or start frequency) of the selected waves, select Use Start in the Domain Selection and type in a new value.

   If you want to end the comparison at a point other than the stop time (or stop frequency) of the selected waves, select Use Stop and type in a new value.
4. If you want to specify a delay to shift all added waveforms from the reference dataset, select Use Reference Delay in the Delay Definition section and type in the delay time.

   If you want to specify a delay to shift all added waveforms from the test dataset, select Use Test Delay and type in the delay time.

5. When making comparisons you can choose how the results are treated. Select from the following Plot Options:

    - Automatically Add Comparison to Wave Window — This option adds the comparison results to the current wave window, including both correct (waveforms with no differences) and erroneous (waveforms with differences) waveforms.
    - Automatically Add Comparisons With Errors to Wave Window — This option adds only comparison results that contain errors (waveforms with differences) to the current wave window.
    - Do Not Plot Results of Comparison — This option does not add the comparison results to the current wave window.

6. You can sort the comparison results. Click the **Sort By** checkbox and choose from the following options:

    - First Difference — This option finds the first time when any signal in one run begins to differ from the corresponding signal in another run, and plots the signals in sorted order.
    - Maximum Difference — This option finds the first time when any signal in one run differs most from the corresponding signal in another run, and plots the signals in sorted order.
    - Maximum Spectrum Difference — This option finds the frequency when any signal in one run differs the most from the corresponding signal in another run, after phase shift, and plots the signals in sorted order.
    - Plot Top Results — This option specifies the number of top results to be plotted. Default is 8.
    - Annotate Result Waveforms With Sorting Parameter — If checked, the sorting parameter value will be plotted as annotation (or other marker, for example delta) over the plotted comparison results (waveforms).

Related Topics

General Options Tab

Setting Comparison Method Options

You can set options to define how the Waveform Compare Tool compares digital waveforms. Choose **Tools** > **Waveform Compare** > **Options** and select the **Comparison Methods** tab.

There are three methods for comparing digital waveforms.

Continuous Comparison **250**
Sampled Comparison **250**
Clocked Comparison **251**

Continuous Comparison

The Waveform Compare Tool uses the Continuous Comparison method by Default. In this method, test signals are compared to reference signals at each transition (for digital-digital comparisons) or at each simulated data point (for analog-analog and mixed signal comparisons) of the reference waveform.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Options** and select the **Comparison Methods** tab.
2. Select **Continuous Comparison**.
3. Optionally:

If you want to...	Do the following:
Perform a “digital” edge comparison where the source and reference signals (one, or both) are analog. **Tip:** This is useful, for example, where the waveforms represent a clock but have different amplitudes. You can compare the delay between the waveforms.	Select **Convert Analog To Digital** in the Edge Comparison section. An Analog to Digital conversion is applied automatically during the comparison. Refer to additional settings for “Analog to Digital Conversion” on page 257 and “Digital Comparison” on page 254.

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

Sampled Comparison

Sampled comparison enables you to define either a Frequency or a Period to use for the comparison session. A comparison between a reference and a test waveform is made for each point sampled.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Options** and select the **Comparison Methods** tab.
2. Select Sampled Comparison.
3. Type either a Frequency or a Period to use for the comparison session.
4. Optionally:

If you want to...	Do the following:
Perform a “digital” edge comparison where the source and reference signals (one, or both) are analog. **Tip:** This is useful, for example, where the waveforms represent a clock but have different amplitudes. You can compare the delay between the waveforms.	Select **Convert Analog To Digital** in the Edge Comparison section. An Analog to Digital conversion is applied automatically during the comparison. Refer to additional settings for “Analog to Digital Conversion” on page 257 and “Digital Comparison” on page 254.

5. Click **OK** to save the settings.

Clocked Comparison

In the Clocked Comparison method, you define a clock to use as a trigger waveform for the comparison. Signals are compared only at, or just after, an edge on some signal.

In this mode, you define one or more clocks. The test signal is compared to a reference signal and both are sampled relative to the defined clock. The clock can be defined as the rising or falling edge (or either edge) of a particular signal plus a user-specified delay. The design need not have any events occurring at the specified clock time.

The clocked comparison settings here will also apply to analog-analog and analog-digital (mixed) comparisons.

When you define the clock, it can also apply to an analog waveform. The edges, rising or falling, are automatically calculated using functions available in the EZwave Measurement Tool.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Options** and select the **Comparison Methods** tab.
2. Select Clocked Comparison.
3. Select a waveform from the dropdown list or use the Use the **Add Selected Waveform** icon  to add the currently selected waveform to the list.

4. Click **Clocks** to open the Clocks dialog box. This dialog box lists all the clocks you've defined.
5. Click **Add** to open the Add Clock Dialog Box. If you want to modify a clock you've previously added, select it from the list of clocks and click **Modify**.
6. Specify a name for this clock definition in the Clock Name field.
7. If you want to set the Delay Signal Offset, type a value in the field. By default, this is 0.
8. In the Based on Waveform field, select the waveform whose edges are to be used as the strobe trigger.
9. In the Compare Strobe Edge field, select which edge to use in the clocked comparison. By default, only the rising edge is used.
10. Click **OK** to close the Add Clocks dialog box.
11. Click **OK** to close the Clocks dialog box.
12. Optionally:

If you want to...	Do the following:
Perform a "digital" edge comparison where the source and reference signals (one, or both) are analog. **Tip:** This is useful, for example, where the waveforms represent a clock but have different amplitudes. You can compare the delay between the waveforms.	Select **Convert Analog To Digital** in the Edge Comparison section. An Analog to Digital conversion is applied automatically during the comparison. Refer to additional settings for "Analog to Digital Conversion" on page 257 and "Digital Comparison" on page 254.

13. Click **OK** to save the settings.

Setting Tolerance Options

You can set tolerance options for analog-analog, digital-digital, mixed-signal and spectral comparisons.

These options are an extension of the Continuous Comparison of digital-digital compare options. They are set using the Tolerances Tab. Choose **Tools** > **Waveform Compare** > **Options** and select the **Tolerances** tab.

The tolerance options are described in the following topics:

Analog Comparison **253**
Digital Comparison **254**
Mixed Comparison **255**
Spectral Comparison **255**

Analog Comparison

The analog comparison options enable you to specify the X and Y tolerances, or use the default tolerances.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Options** and select the **Tolerances** tab.
2. Type in percentage or numerical values for X Tolerance and Y Tolerance.

   Alternatively, select Use Automatic X Tolerance or Use Automatic Y Tolerance, the Waveform Compare Tool uses the following formulas to calculate the tolerances:

   - X Tolerance

     The minimum of *((x_max - x_min)/number_of_points)* or *(0.01 * (x_max - x_min*))
   - Y Tolerance

     *0.01 * (y_max - y_min)*

   You can also specify both Absolute and Relative Y tolerance at the same time, for a point relative Y tolerance. The Y tolerance used is then:

   *absolute_y_tolerance + (relative_y_tolerance * y_value_of_the_sampling_point)*
3. Optionally, you can also filter differences by specifying a minimum delta X value. This is usually time - differences are ignored if they have a shorter duration than the specified value. Type a value in the By Minimum Delta X field.

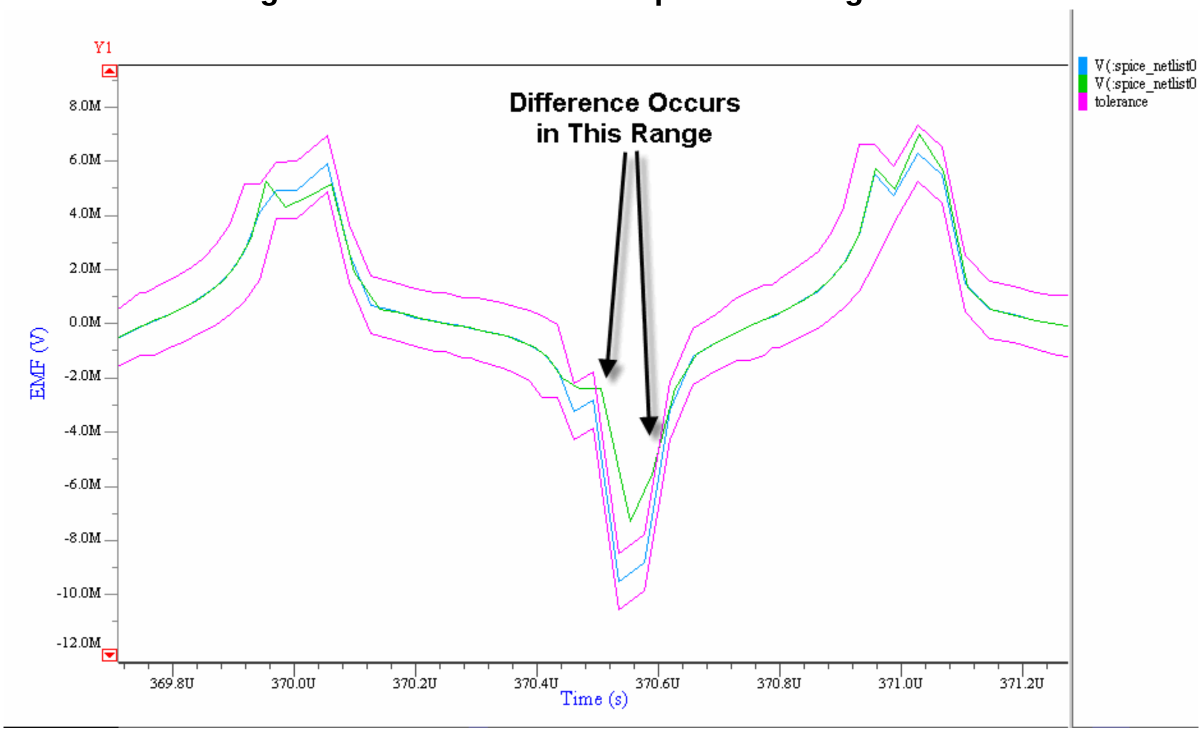
Results

The analog-analog comparison is based on waveform data points provided by the simulator, for both the reference and the test waveform.

When you choose to specify the tolerances, you can use percentage or numerical values. If you specify a percentage value (for example, 0.1%), it defines a tolerance value relative to the minimum and maximum of the waveform (in this example, 0.1% of (x_max – x_min) or 0.1% of (y_max – y_min)). If you instead specify a numerical tolerance value (for example, 0.1), it defines an absolute tolerance to be used around the X or Y values.

In the following figure, the blue waveform represents the reference waveform, and the green waveform represents the test waveform. The magenta "waveforms" are the virtual tubes that represent the tolerances.

Figure 5-47. Waveform Compare Showing Tolerance



For details on how the waveform compare algorithm uses tolerances, see "Analog Waveform Comparison Algorithm" on page 258.

Digital Comparison

The digital-digital waveform comparison options enable you to specify the leading and trailing edge tolerances.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Options** and select the **Tolerances** tab.

2. The Digital Comparison option contains the following settings:
    - Leading Tolerance

      Here you can specify a leading edge tolerance. This specifies the maximum time a test signal edge is allowed to lead a reference edge in an asynchronous comparison of digital signals. Defaults is 0.
    - Trailing Tolerance

      Here you can specify a trailing edge tolerance. This specifies the maximum time a test signal edge is allowed to trail a reference edge in an asynchronous comparison of digital signals. Defaults is 0.

Mixed Comparison

You can specify leading and trailing edge tolerances for mixed signal comparison.

When comparing mixed-signal waveforms, the Waveform Compare Tool must first convert the analog waveform into a digital waveform (refer to Analog to Digital Conversion for conversion options). Then it applies a digital-digital comparison (the comparison function is based on the least accurate waveform). Refer to Analog to Digital Conversion.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Options** and select the **Tolerances** tab.
2. The Mixed Comparison option contains the following settings:
    - Leading Tolerance

      Here you can specify a leading edge tolerance. This specifies the maximum time a test signal edge is allowed to lead a reference edge in an asynchronous comparison of digital signals. Defaults is 0.
    - Trailing Tolerance

      Here you can specify a trailing edge tolerance. This specifies the maximum time a test signal edge is allowed to trail a reference edge in an asynchronous comparison of digital signals. Defaults is 0.

Spectral Comparison

You can specify Frequency Tolerance, Amplitude Tolerance and Noise Floor for spectral comparison.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Options** and select the **Tolerances** tab.

Note

The **Comparison Methods** tab options do not affect spectral comparison.

2. The Spectral Comparison option contains the following settings:
    - Frequency Tolerance

      Here you can specify a frequency tolerance. Defaults is 0.
    - Amplitude Tolerance

      Here you can specify an amplitude tolerance. Defaults is 0.
    - Noise Floor

      Here you can specify a noise floor level. If the Use Noise Floor option is enabled, errors that occur below the specified noise floor amplitude are ignored in the comparison.

Setting Conversion Options

You can use these options for Digital Conversion and Analog to Digital Conversion.

Digital Conversion 257
Analog to Digital Conversion 257

Digital Conversion

You can specify the VHDL and Verilog signal value matching rules.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Options** and select the **Conversions** tab on the Comparison Options dialog box.
2. Specify VHDL and Verilog signal value matching rules in VHDL Matching and Verilog Matching areas. The D in these matching options represent the '-' "don't care" std_logic value.

Tip
For more information on the VHDL and Verilog matching options, refer to the Mapping Data Types section of the *Questa SIM User's Manual* (if the host application installs this manual).

Related Topics

Conversions Tab

Comparison Options Dialog Box

Analog to Digital Conversion

You can specify thresholds for the analog to digital conversion.

Procedure

1. Choose **Tools** > **Waveform Compare** > **Options** and select the **Conversions** tab on the Comparison Options dialog box.
2. Specify whether you want to apply a Single Threshold (default) or Two Thresholds for the analog to digital conversion. By default, the "Use Automatic Thresholds" option is checked, specifying that the thresholds are deduced from the input waveforms. Uncheck this to specify fixed values for each of the thresholds.

Related Topics

Conversions Tab

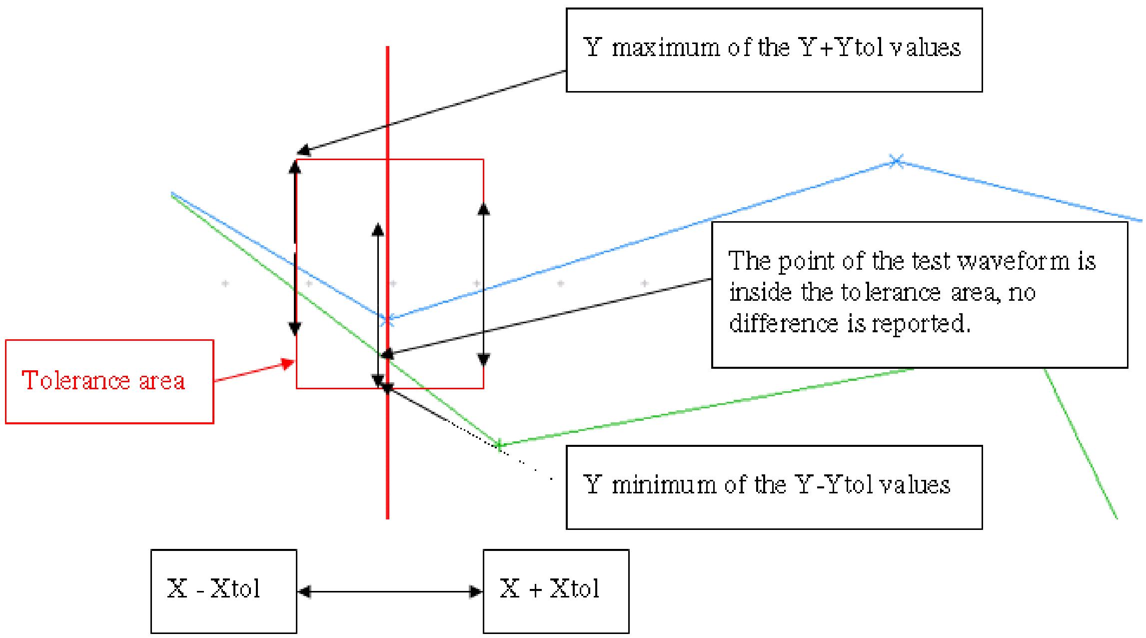
Comparison Options Dialog Box

Analog Waveform Comparison Algorithm

The EZwave waveform comparison algorithm is based only on waveform data points of both the reference and result waveforms. At each X value on either the reference or result waveform, the calculations are executed on the reference waveform.

This is illustrated in the following figure. The blue line is the reference waveform and the green line is the result waveform. "Xtol" and "Ytol" are the tolerances for X and Y, respectively.

Figure 5-48. Calculations on the Reference Waveform



Using the default or defined X tolerances (see "Analog Comparison" on page 253), the algorithm calculates (X – Xtol) and (X + Xtol). These values are used with the X value to deduce (x, y) corresponding data points to the reference waveform.

At each (x, y) data point, the algorithm calculates (Y – Ytol) and (Y + Ytol). These values are used to deduce the maximum and minimum Y values.

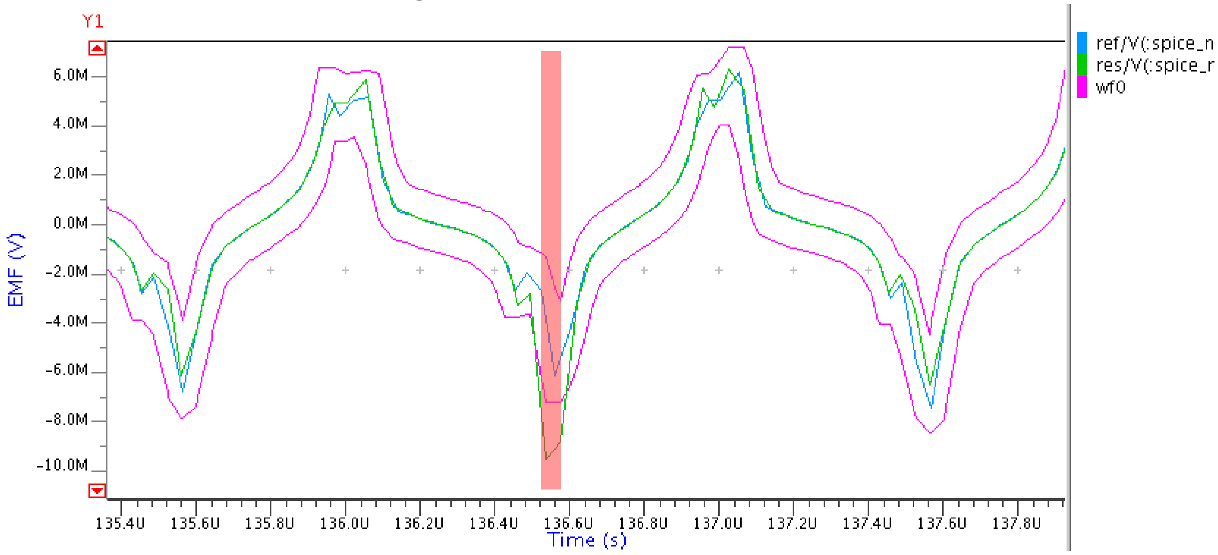
You can specify both Absolute and Relative Y tolerance at the same time, for a point relative Y tolerance. The Y tolerance used is then:

```
absolute_y_tolerance + (relative_y_tolerance *
y_value_of_the_sampling_point)
```

These calculations form the tolerance area (the red rectangle in Figure 5-48). During comparison, the result waveform is checked to see if it lies in the tolerance area at the original X data point.

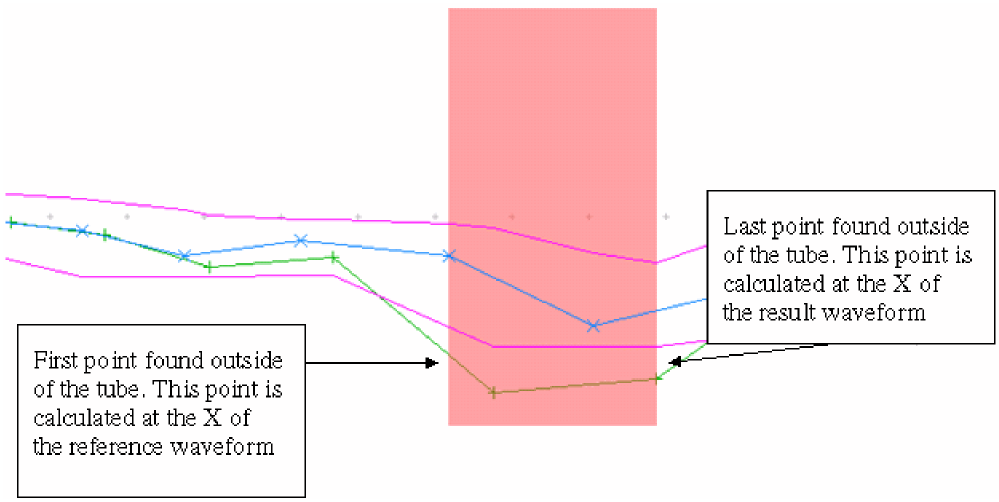
The tolerances define a virtual “tube” around the reference waveform. The Waveform Compare Tool checks to see if the test waveform is “inside” the tube, and reports a difference if the test waveform is outside the tube. In the next figure, the blue reference waveform is “surrounded” by two magenta waveforms that represent the tolerance tube. The green result waveform is only highlighted (red vertical highlight) at the area where it is “outside” the tube.

Figure 5-49. Tolerance “Tube”



The “difference” highlight is calculated from the first data point found outside of the tube to the last data point found outside of the tube, as shown in te following figure:

Figure 5-50. Data Points Outside of the Tolerance Tube



wreal Waveform Comparison

The comparison of “wreal” waveforms follows the same method and tolerance values as comparison of “real” waveforms.

When ‘X’ or ‘Z’ is encountered, only the ‘X’ tolerance is considered. Within the ‘X’ tolerance, ‘X’ or ‘Z’ may match any other values, including numerical ones.

Outside of the ‘X’ tolerance, a strict ‘Y value’ comparison is applied, meaning that ‘X’ must match ‘X’ and the ‘Y value’ must match the ‘Y value’.

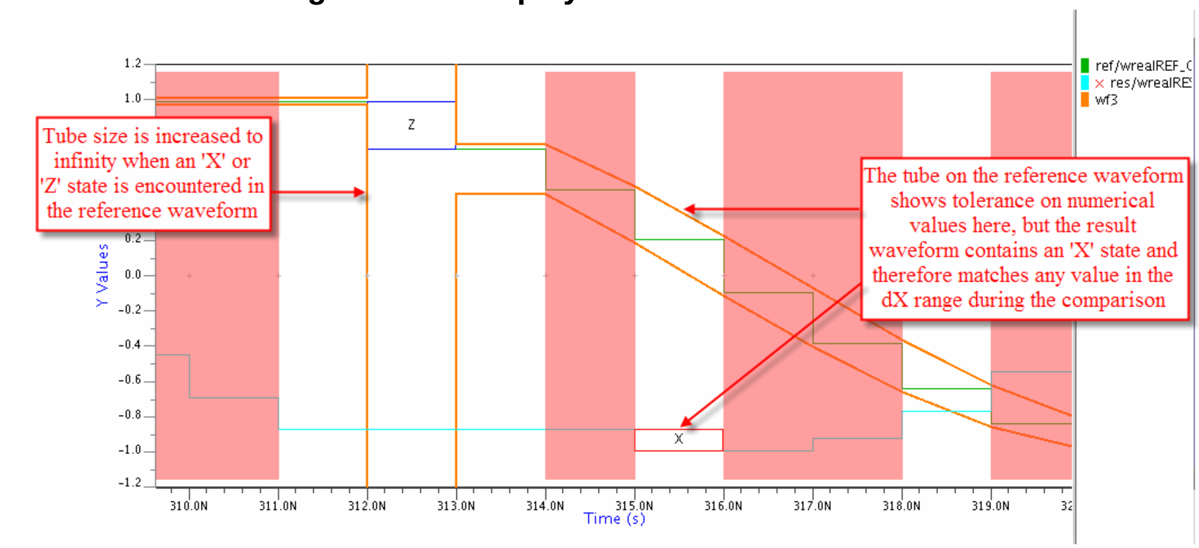
The display of the tolerance tube is similar to that used for the “real” waveform, except when the ‘X’ or ‘Z’ state is reached. The tolerance tube is increased to infinity here to enable ‘X’ and ‘Z’ values to match any others.

The following figure shows the tolerance tube increased to infinity when a ‘Z’ state is encountered in the *reference* waveform. However, when an ‘X’ state is encountered in the *result* waveform, the tolerance tube does not increase to infinity. Values are matched within the ‘X’ tolerance range.

Note

In the comparison report, the location of the ‘X’ and ‘Z’ state is indicated instead of the location of the ‘Y value’ maximum difference.

Figure 5-51. Display of the Tolerance Tube



Note

This comparison will produce different results to Questa SIM. In Questa SIM, the digital comparison algorithm (digital comparison with default tolerance of 0) applies to wreal waveforms. In Questa ADMS the tolerance is considered around the X domain for ‘X’ and ‘Z’ states, and elsewhere the ‘real’ waveforms comparison algorithm (analog comparison with dX and dY tolerance) applies.

Analyzing Waveform Parameters to Generate Pivot Waveforms

This topic describes how to use the Parameter Analyzer Tool to generate "pivot" waveforms. You can change the running variable (x axis) of one or more compound waveform(s) to one of the waveform's parameter values, or to the y values of a different compound waveform, and then plot. You can also filter (exclude) individual parameter values and group parameters by color.

Restrictions and Limitations

- Only compound waveforms may be selected to use as the y axis of new pivot waveforms.

Procedure

1. Open the required waveform database and expand the set of waveforms in the Waveform List Panel.

    a. Choose **Tools** > **Parameter Analyzer**.

    The Parameter Analyzer Tool Dialog Box opens

    b. Drag and drop the required waveform(s) into the Waveform List area. Alternatively use **Add Selected Waveform** icon to add the currently selected waveform(s).

    Tip

    You can also right-click a compound waveform in the Waveform List Panel and choose **Plot versus** > **Parameter Analyzer** to open the Parameter Analyzer Tool dialog box populated with the selected source waveform.

    c. Select the y-axis checkbox adjacent to one or more waveforms in the Waveform List from which you want to create new pivot waveforms.

    d. Choose the parameter or waveform (y values) that you want to use as the new x axis from the X-Axis dropdown list.

    e. In the Display Options area, choose whether you want to plot the resultant pivot waveforms **Stacked** or **Overlaid**, and whether you want the plot results in a **New Window**.

    f. Click **Apply** or **OK** to create the pivot waveform.

2. (Optional) To set up filters to exclude parameter values from the pivot waveforms:

    a. Click the **Advanced** tab of the Parameter Analyzer Tool and then click **New**.

    The Filters Setup dialog box opens.

b. Choose a Parameter from the menu. Select one or more values from the Values to Hide list. When you have finished selecting values, click **OK**. The parameter name and excluded values are added to the Add Filter list.

c. Repeat the steps above if you want to add another parameter name and values to the Add Filters list.

d. If you want to change any of the filter values, select a row in the Run Filters table and click **Edit** or click **Delete** to remove the selected row from the Run Filters table.

e. Click **Apply** or **OK** to create the pivot waveform.

3. (Optional) Waveforms may be colored depending on the selected parameters. All waveforms that share the same parameter value are displayed with the same color. To set up color groupings:

   a. Click the **Advanced** tab of the Parameter Analyzer Tool and choose one or more parameters from the Group Parameters dropdown list.

   b. Click **Apply** or **OK** to create the pivot waveform.

4. (Optional) You can quickly create a simple pivot waveform without the need to open the Parameter Analyzer Tool dialog box:

   a. Open the required waveform database and expand the set of waveforms in the Waveform List Panel.

   b. Right-click the required compound waveform in the Waveform List Panel and choose **Plot versus** > **Select X-Axis Param** and select an x-axis parameter from the list. A pivot waveform is then created using the selected parameter as the x axis.

Results

A pivot waveform is plotted for each of the compound source waveforms selected. The x axis of the source waveforms has been changed to use the specified x-axis parameter or waveform instead.

Related Topics

Parameter Analyzer Tool Dialog Box

Filters Setup Dialog Box

Analyzing Current Consumption

When your simulation results include a current analysis, you can use the Current Analysis dialog box to analyze current consumption of a circuit and its components over time. You can define time ranges and thresholds to filter results to locate a problem within a subcircuit or device.

The current analysis is fully hierarchical, which means that you can display all the circuit components, showing the hierarchy, and filter accordingly down through the hierarchy.

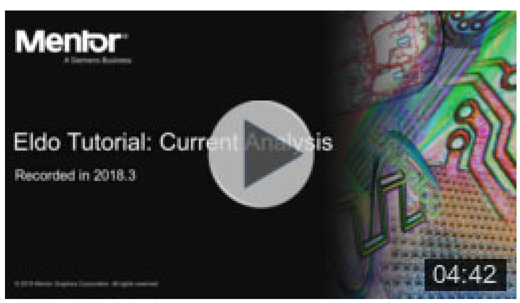
The main purpose of the current analysis is to help you understand where and when the current consumption is inside a large circuit, or which part of a circuit is consuming more than a specified threshold, or which subcircuits are consuming what current. You can analyze the current without having to resimulate, everything is manipulated in the Current Analysis dialog box and you perform your debugging tasks accordingly there.This analysis provides an effective way to report current consumption. For designs operating on battery supplies, leakage issues are important. Advanced process nodes tend to make these problems worse. The power supply voltages tend to decrease, and the leakage currents tend to increase exponentially at each new node. This means the monitoring of current variables (that relate directly to power) is as important, if not more, than the monitoring of voltages.

When a simulation has been run with current analysis requested, current waveforms are saved in the waveform database in folders named *CURRENT_<analysis_type>* depending upon the types of analyses requested during simulation. You can plot waveforms from these folders as normal.

Note

See also .CURRENT_ANALYSIS in the *Eldo Reference Manual*.

A Tutorial—Using Current Analysis is in the *Eldo User's Manual*. A video is available that shows the tasks described in this tutorial.



Prerequisites

You must have requested a current analysis during simulation with Eldo, by including the .CURRENT_ANALYSIS SPICE command in your netlist or command file. Refer to ".CURRENT_ANALYSIS" in the *Eldo Reference Manual*.

Procedure

1. Open a waveform database containing results from a current analysis.
2. Expand the required set of current analysis results in the Waveform List Panel, right-click the folder at the required level of hierarchy and choose **Current Analysis** from the popup menu.

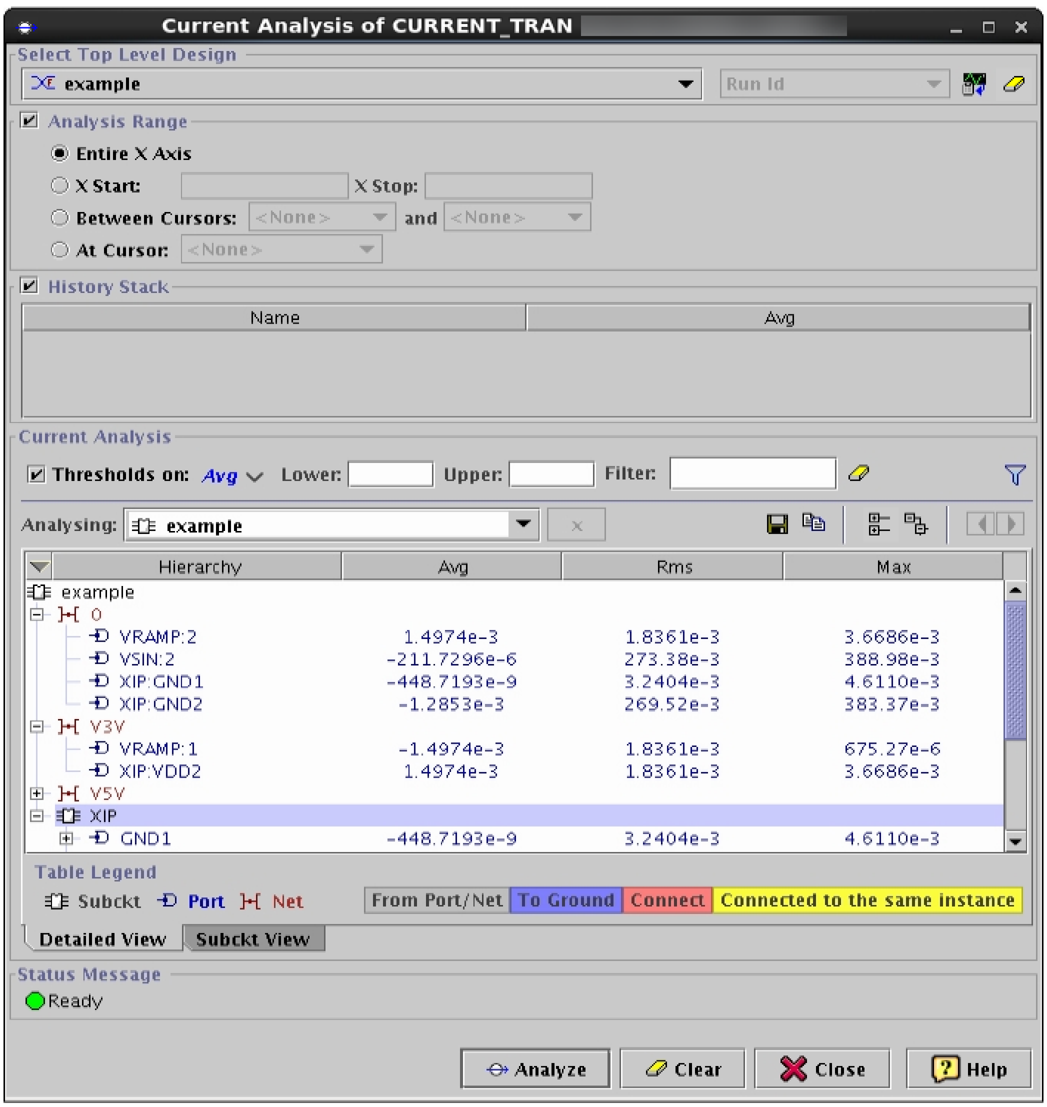
The Current Analysis Dialog Box opens.

3. Optionally, you can specify values for X Start and X Stop for the analysis, or specify one or two cursors, to only list results that occur within a specified time range or at a point.

Tip

You can specify the EZwave analysis range using the Eldo command .CURRENT_ANALYSIS, parameters TSTART and TSTOP. The X Start and X Stop values are then automatically set as the analysis range on the dialog box.

4. Click **Analyze**. The Current Analysis table is populated with the current results from the design. You can click to expand all of the results.



Tip

You can display current values with different notations. Choose **Edit** > **Options** > **Data Format** to specify the display notation. Notation does not apply to exported values.

5. Click the button to choose which results to display in the table. You can click on a column heading to sort the results in ascending or descending order. You can expand the Hierarchy by clicking on the + symbol adjacent to a Subckt, Port or Net item.

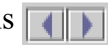
Tip

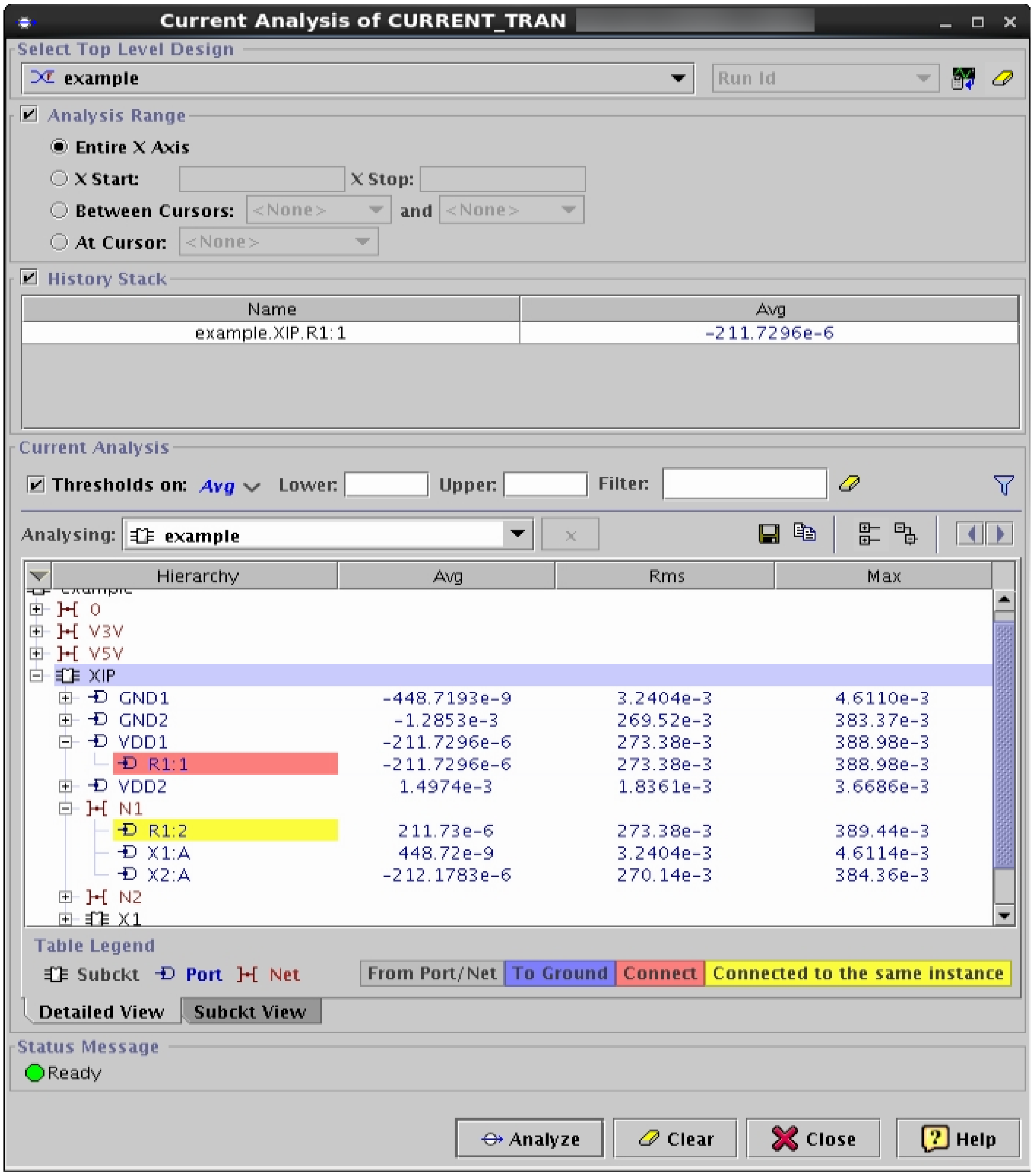
You can specify a list of columns that are displayed in the table, using the Eldo command .CURRENT_ANALYSIS parameter MEAS_TYPE.

6. To filter the table, you can do any of the following:

   - Select **Thresholds on**, choose a column from the adjacent dropdown list, specify Lower, Upper, or both, current thresholds and click . Filtering is performed on the absolute values of the data (minus sign indicates direction of flow).
   - Type text in the Filter field and click .

7. To highlight all connected ports for a device, right-click an item in the Hierarchy column in the Current Analysis table and choose **Connected to the same instance** from the popup menu. The item is then highlighted in red and its connected ports are

highlighted in yellow. Use the left and right arrow buttons  to move between highlighted items.



Tip

To identify the source of any unexpectedly high current consumption, review the average current results in the list, select the item of interest in the hierarchy and choose **Connected to the same instance** from the popup menu. Repeat until you have drilled down the hierarchy sufficiently to identify the offending item(s).

You can also right-click an item in the Hierarchy column in the Current Analysis list and choose **Step In** to open the corresponding instance and select the port and expand it, or **Step Out** to highlight the parent subcircuit (or the top level).

8. If you want to see the results for a single subcircuit, right-click a subcircuit in the table and select **See in Subckt View**. Alternatively, to see a simplified view showing results for all subcircuits, click the **Subckt View** tab. In the subcircuit view, the available columns show the hierarchy, abs(sum(avg_node)), and sum(abs(avg_node)), where "avg_node" is the average current value for the ports of the subcircuit.

9. If you want to export the current results for a single subcircuit as a *.csv* file, right-click a subcircuit in the table and select **Export currents**. The Export Currents dialog box opens. Specify the file Path and the required subcircuit hierarchy Depth and click **OK**. Example output:

```
.HEADER
..NAMES
Hierarchy Level 1,Current Value,Current Value Sum
.DATA
X1,0.0,8.967999240248334E-7
X2,0.0,4.2440985658298493E-4
X3,0.0,0.0029948591015101465
```

10. To plot a current waveform, double-click a Port in the Hierarchy column. Alternatively, right-click a Port in the Hierarchy column and choose **Plot** from the popup menu.

    The Port and its average current is automatically added to the History Stack.

11. To copy a waveform name to the clipboard or send the name to the Waveform Calculator, right-click an item in the Hierarchy column in the Current Analysis list and choose either **Copy Waveform Name** or **Send to Calc**.

12. Select a row in the History Stack; the tree automatically expands to show the element in the Current Analysis list.

13. To save the History Stack to a file, right-click in the History Stack and choose **Save As** from the popup menu. You can also copy the History Stack to the clipboard, right-click in the History Stack and choose **Save As** from the popup menu.

14. To edit the History Stack, right-click in a row and choose **Remove** from the popup menu. Alternatively, choose **Remove From Here** to remove the currently selected row and all rows below it, or choose **Clear** to remove all rows from the History Stack.

15. The Current Analysis dialog box is an independent window and may be minimized, maximized and iconified. You can also show or hide the Analysis Range and History Stack sections using the checkboxes, to free more space to display the Current Analysis table.

Tip

To make investigating current leakage issues easier, you can open as many Current Analysis dialog boxes as you need, either on the same, or different, databases. This enables you, for example, to compare results before and after making design changes.

Related Topics

Current Analysis Dialog Box

Analyzing Power Consumption

When your simulation results include a power analysis, you can use the Power Analysis dialog box to analyze power consumption of a circuit and its components over time. You can define time ranges and thresholds to filter results to locate a problem within a subcircuit or device.

The Tutorial—Using Power Analysis is in the *Eldo User's Manual*. A video is available that shows the tasks described in this tutorial.

When a simulation has been run with power analysis requested, power waveforms are saved in the waveform database in folders named *POWER_<analysis_type>* depending upon the types of analyses requested during simulation. You can plot waveforms from these folders as normal.

Prerequisites

You must have requested a power analysis during simulation with Eldo or Questa ADMS, by including the .POWER_ANALYSIS Spice command in your netlist or command file.

Procedure

1. Open a waveform database containing results from a power analysis.
2. Expand the required set of power analysis results in the Waveform List Panel, right-click the folder at the required level of hierarchy and choose **Power Analysis** from the popup menu.

   The Power Analysis Dialog Box opens.
3. Click **Analyze**. The tree list is populated with the power waveform results from the design. The power consumption for each circuit element is shown in the Sum column, together with additional results.

   For RF Power Analysis, table columns display the Sum and power for individual harmonics.

4. Click the button to choose which results to display in the table. You can click on a column heading to sort the results in ascending or descending order. You can expand the hierarchy by clicking on the + symbol adjacent to a circuit element.

5. If the results are numerous and difficult to view, you can filter them such that you see only the results of interest.

   - Use the depth control to limit the analysis to the specified number of levels from the top down. Results coming from parts of the circuit located at a deeper level of hierarchy will not be listed.
   - Define a minimum and/or maximum power consumption threshold for the listed results. Results with power consumption values below or above these thresholds will not be listed.
   - Define lower and upper time bounds of the analysis, to only list results that occur within the specified time range.

6. The Power Analysis dialog box is an independent window and may be minimized, maximized and iconified. You can also show or hide the Analysis section using the checkbox, to free more space to display the Power Analysis table.

Note

The Eldo simulator, working in conjunction with EZwave, provides a (inst=) parameter to specify subcircuits in the .POWER_ANALYSIS spice command. Power analysis is then only applied to the specified subcircuits. If this parameter is used, some of the hierarchies in the resulting *POWER_<analysis_type>* folder may be empty (when they are located on a path to a specified instance). The Power Analysis dialog box will show a "-" symbol instead of a value for the power, to indicate when a subcircuit has not been analyzed by Eldo.

Tip

See also .POWER_ANALYSIS in the *Eldo Reference Manual* and Tutorial—Using Power Analysis for Static Leakage Analysis of a PLL Circuit in the *Eldo User's Manual*.

Related Topics

Power Analysis Dialog Box

Chapter 6
Post-Processing

This section describes how to use EZwave to process and transform data after simulation.

Working With a Bus 273
Creating a Bus. 273
Editing Bus Contents 274
Transforming Analog Waveforms to Digital. 275
Transforming Digital Waveforms to Analog. 275
Bus and Bit Transformations 277
Bus Transformation Setup 277
Bit Transformation Setup 278
Measurement Tool 279
Taking a Measurement 279
Taking DC Hysteresis Measurements. 281
Taking Eye Diagram Measurements. 284
Frequency Domain Measurements 286
General Measurements 289
Taking Statistical Measurements 296
Time Domain Measurements 299
Waveform Calculator 312
Using Expressions in the Waveform Calculator. 313
Waveform Calculator Shell Commands 314
Using Buttons in the Waveform Calculator 315
Using Built-In Functions in the Waveform Calculator. 317
Using and Editing User-Defined Functions in the Waveform Calculator 319
Creating a Tcl Script From the Waveform Calculator History. 320
Updating User-Defined Function Help Documentation 323
Using the Measurement Tool Functions in the Waveform Calculator 325
Built-In Functions. 332
Calculator Buttons 349
Jitter Tool 368
Using the Jitter Tool 368
Jitter Measurement Types. 370
Extracting Outputs from a Database 382
Signal Processing Functions 384
Using the Fast Fourier Transform Tool 384
Using the Inverse Fast Fourier Transform Tool 386
Using the Spectrum Measurement Tool 388

Autocorrelation Function and Power Spectral Density 391
Convolution Function 392
Harmonic Distortion Function 394
Signal to Noise Function. 394
Signal Processing Function Window Shapes 395
Windowing Transforms 396

Using the DNA Advisor Tool 399

Working With a Bus

You can create a bus from selected digital waveforms (bits) or modify a bus after it is created.

Creating a Bus 273
Editing Bus Contents... 274

Creating a Bus

Create your own bus from selected digital waveforms (bits).

Procedure

1. Select bits to add to the bus from the active graph window or from the Waveform List panel.

   Use Shift-click or Ctrl-click to select multiple waveforms.

   Note

   For databases loaded from *.fsdb* files, only loaded waveforms display in the Waveform List panel. Waveforms are loaded when they are displayed for the first time or when they are located at hierarchy level. The Waveform List panel only displays waveforms that have already been loaded.

2. Choose **Tools** > **Create Bus**.

   The Create Bus Dialog Box opens.

   EZwave identifies the longest prefix common to all bits of the bus and adds it as the default bus name in the Name field.

3. Click the text window to rename the bus and type the bus name.

   If no common prefix is found, the default bus name is Bus1. Subsequent buses with no common prefix are named Bus2, Bus 3, and so on.

4. Click the dropdown list next to Radix and select the required radix.

   The Bits in Bus window lists all of the selected waveform names. The waveform selected first is used as the Most Significant Bit (MSB) of the bus, and the waveform selected last is used as the Least Significant Bit (LSB).

   EZwave attempts to identify the Most Significant Bit (MSB) and the Least Significant Bit (LSB) indexes of each bit, and add a range (MSB:LSB) between bus index delimiters at the end of the common bus name prefix.

5. You can specify bus index delimiters in "Data Format Options" on page 482 of the "EZwave Display Preferences Dialog Box" on page 476.

If the bus is an analog or hybrid bus, the analog signals are automatically transformed to digital.

6. Select Auto-Update Threshold and the threshold calculation is automatically updated when the bit order is modified.

   This option is enabled by default.

7. Select Single Threshold or Two Thresholds to digitize the input waveform.
8. Click the text windows of the Transformation Setup field and change the default to the threshold values you want.
9. Click **OK** to apply or click **Cancel** to abort bus creation.

Related Topics

Loading .fsdb and .tr0 Files

Create Bus Dialog Box

Waveform List Panel

Data Format Options

Editing Bus Contents

The content of a bus can be modified after it is created. You can add and remove bits, change MSB and LSB definitions and invert the bits of the bus.

Procedure

1. To add bits to the bus, select the bus, right-click and choose **Create Bus**.

   The Create Bus dialog box opens.

2. Click **Add**.

   A dialog box opens, which lists all of the digital waveforms in the database not already included in the bus.

3. Select one or multiple waveforms and click **OK** to add these to the bus.

> **Note**
> For databases loaded from *.fsdb* files, only loaded waveforms are displayed in the **Create Bus** > **Add Bits to Bus** dialog box. Waveforms are loaded when they are displayed for the first time or when they are located at hierarchy level. Refer to "Loading .fsdb and .tr0 Files" on page 1272.

- Click **Invert** to invert the bits of the bus.
- Use the **Move Up** and **Move Down** buttons to re-order the bits in the list.

Related Topics

Create Bus Dialog Box

Transforming Analog Waveforms to Digital

You can convert an analog waveform to a digital waveform.

Tip

You can quickly plot analog waveforms as digital waveforms, using default conversion parameters. Select waveform(s) in the Waveform List chooser panel, right-click and choose **Plot as Digital** from the Waveform Name Popup Menu. For more options for the analog to digital conversion process, use the procedure described here.

Procedure

1. Select the waveform in the active graph window, right-click and select **Analog to Digital** from the Waveform Popup Menu.

   The Analog to Digital Conversion dialog box opens.

2. Type a name for the new waveform in the Digitized Waveform(s) field and changing the default name.
3. Choose whether to plot the digitized waveform Stacked in a new row or Overlaid with the original waveform, or clear the Plot Result Waveform(s) if you do not want to plot the digitized waveform in the active graph window.
4. Set the required threshold(s) in the Transformation Setup fields, to define how the waveform is digitized.
5. Click **OK** to perform the transformation.

Related Topics

Analog to Digital Conversion Dialog Box

Transforming Digital Waveforms to Analog

Transforming Digital Waveforms to Analog

You can create an analog waveform from a digital signal or bus by transforming the digital waveforms to analog.

All analog waveforms created in this manner are created in the “calc” database.

Procedure

1. Select the digital signal or bus in the active graph window, right-click and select **Digital to Analog** from the Waveform Popup Menu.

   The Digital to Analog Conversion dialog box opens.

   Depending on whether you have selected an individual signal or signals, a bus, or both, the dialog box may include a section for Bit Transformation Setup, Bus Transformation Setup, or both.

2. In the Name text box, change the default name to a desired waveform name. If multiple waveforms are selected, the Name text window is dimmed out and the default names are used.

3. Select whether to plot the resulting waveform Stacked in a new Graph Row or Overlaid with the original waveform, or if you do not want to plot the resulting waveform to the active graph window, clear the Plot Result Waveform(s) checkbox.

Related Topics

Digital to Analog Dialog Box

Transforming Digital Waveforms to Analog

Bit Transformation Setup

Bus Transformation Setup

Bus and Bit Transformations

You can specify the transformation setup for a bus or digital signal.

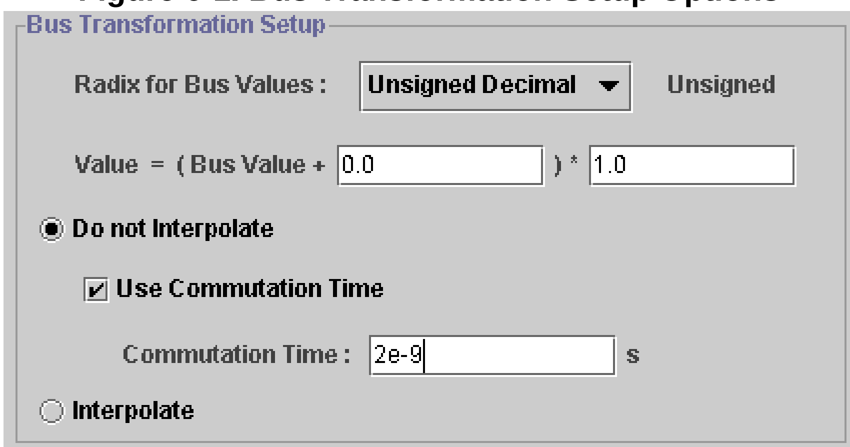
Bus Transformation Setup 277
Bit Transformation Setup 278

Bus Transformation Setup

You can specify the transformation setup for a bus.

The options are displayed in the following figure:

Figure 6-1. Bus Transformation Setup Options



Procedure

1. Specify the Radix by using the dropdown list and selecting one of the following values: **Two's Complement** (signed), **Binary**, **Hexadecimal**, **Octal**, **Unsigned Decimal**, (all unsigned), or **Ascii**.
2. Specify the Value by entering an addition value and then a multiplication value. If the bus value is one of the standard logic states, then the analog value is the same as the previous analog value, or 0.0 in the case of the initial value.
3. Select either Do Not Interpolate or Interpolate.

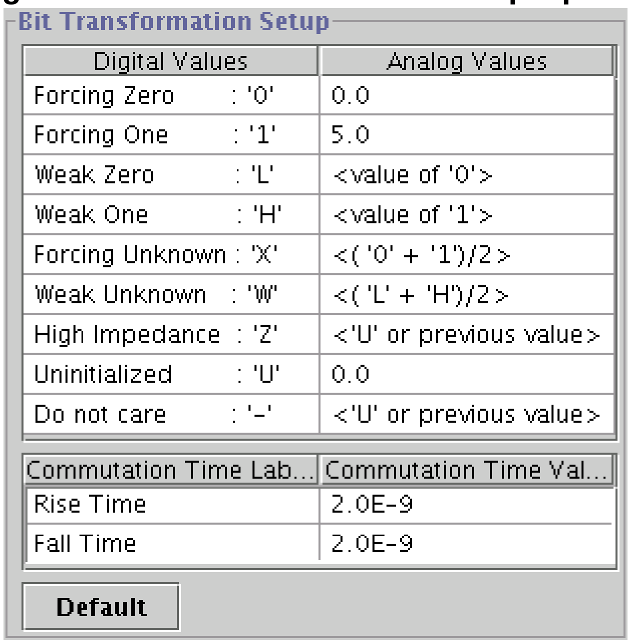
   If you select Do Not Interpolate, specify if you want to Use Commutation Time and type the desired time. This is the time necessary to switch from the previous bus value to the new bus value.
4. Click **OK** to perform the transformation, or click **Cancel** to abort and close the dialog box.

Bit Transformation Setup

You can specify the bit transformation for a digital signal:

The options are displayed in the following figure:

Figure 6-2. Bit Transformation Setup Options



Bit Transformation Setup

Digital Values	Analog Values
Forcing Zero : '0'	0.0
Forcing One : '1'	5.0
Weak Zero : 'L'	<value of '0'>
Weak One : 'H'	<value of '1'>
Forcing Unknown : 'X'	<('0' + '1')/2 >
Weak Unknown : 'W'	<('L' + 'H')/2 >
High Impedance : 'Z'	<'U' or previous value>
Uninitialized : 'U'	0.0
Do not care : '-'	<'U' or previous value>

Commutation Time Lab...	Commutation Time Val...
Rise Time	2.0E-9
Fall Time	2.0E-9

Default

Procedure

1. Each digital value has a default real (analog) assigned. To use a different value, type the corresponding analog value as a numeric figure or engineering value. The possible digital values are Forcing Zero, Forcing One, Weak Zero, Weak One, Forcing Unknown, Weak Unknown, High Impedance, Uninitialized, and Do not care. Values that have been used previously in a session are automatically remembered for future transformations.

2. Under Commutation Time Values, type in the Rise Time (the necessary time to go from '0' to '1') and Fall Time (the necessary time to go from '1' to '0').

3. Click **OK** to perform the transformation, or click **Cancel** to abort and close the dialog box.

   For bit transformations, this process is only implemented for “standard logic” waveforms and other digital types that can be internally converted to standard logic waveforms (bit, Boolean, and Verilog logic).

Measurement Tool

The Measurement Tool can be used to take a wide variety of measurements of the elements of a compound waveform.

The Measurement Tool can be used for the following:

Taking a Measurement 279
Taking DC Hysteresis Measurements 281
Taking Eye Diagram Measurements 284
Frequency Domain Measurements 286
General Measurements 289
Taking Statistical Measurements 296
Time Domain Measurements 299

Taking a Measurement

You can make various measurements using the Measurement Tool.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select a measurement category and type from the Measurement dropdown lists.

3. Select one or more waveform elements or waveform labels in the active graph window.

   Click the **Add Selected Waveforms** icon to add the selected waveforms or elements to the Waveform List. Use Shift-click or Ctrl-click to select multiple waveforms, or use the Delete key to remove them.

   Tip

   If you do not want the Measurement Tool Source Waveform(s) list to be automatically populated with currently selected waveforms, choose **Edit** > **Options** to open the EZwave Display Preferences dialog box, choose **General** from the EZwave Preferences list on the left, and uncheck “Use Automatic Waveform Selection for Measurement Tool”.

   Note

   The measurements Delay, Intersect and Slope Intersect can only be applied to a single waveform at any one time.

4. Depending on the type of measurement selected, specify additional information such as topline/baseline, edge trigger, and measurement results settings in the Measurement Setup section. Refer to the specific section on the selected measurement for details on these settings.

   Where the value "Automatic" is specified for an option in the Measurement Setup section, a parameter is applied individually to each waveform. If a value is entered instead, this value is applied to all waveforms.

5. Some measurement types allow you to specify how the results should be displayed in the Measurement Results section (This section is not available for all measurements.):

   a. Where available, specify either to Annotate Waveform(s) with Measurement Results or Plot New Waveform of "<measurement type>" vs <parameter>.

      If there is more than one parameter to choose from, a dropdown list is available for specifying the required parameter. When more than one waveform has been selected for the measurement, only parameters common to all waveforms are available in the list.

6. To apply the measurement to the Entire Waveform, Visible X Region, or Between Two Cursors, click the Apply Measurement to: dropdown list and make a selection.

   Note

   You cannot use horizontal cursors to define parts of a waveform for use with the Measurement Tool.

7. To remove all previous measurements, check Remove All Previous Results.

8. Click **Apply** to perform the measurement, or click **Cancel** to exit the Measurement Tool.

Results

A waveform is displayed, annotated with the measurement result values as appropriate.

Tip

You can specify the precision of the annotated results. This is the same as the Cursor Precision. Refer to "Cursor Options" on page 480.

- To view the detailed results or choose which results to display, right-click a measurement annotation and choose **Measurement Results** from the dropdown list. The Measurement Results dialog box displays the results, with checkboxes to show or hide each result.
- To copy a result, right-click a measurement annotation and choose **Copy Value to Clipboard** from the dropdown list.
- To send the expression used to calculate the results to the Waveform Calculator shell, right-click the result waveform or the measurement annotation, and choose **Send**

Expression to Calc Shell from the dropdown list. The function and all its parameter values are written to the Waveform Calculator shell panel.

Related Topics

Measurement Tool Dialog Box

Taking DC Hysteresis Measurements

You can perform measurements of DC hysteresis width, and lower and upper threshold, using the Measurement Tool.

Table 6-1. DC Hysteresis Measurements

DC Hyst. Left Thres.	The left crossing value.
DC Hyst. Right Thres.	The right crossing value.
DC Hysteresis Width	The difference between the left and right thresholds.

To access: Select **DC Hysteresis** from the left dropdown list, and select **DC Hyst. Left Thres.**, **DC Hyst. Right Thres.** or **DC Hysteresis Width** from the right dropdown list in the Measurement Tool.

Prerequisites

- You must have a *.wdb* file containing the result of a DC Hysteresis computed by Eldo, using ".DC ... HYST" keywords.

  Tip
  See ".DC" in the *Eldo Reference Manual*.

- You must know the Y level of interest, where the thresholds and width are computed.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select DC Hysteresis from the left dropdown list.

3. Select either **DC Hyst. Left Thres., DC Hyst. Right Thres.** or **DC Hysteresis Width** from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - Specify the Y axis to use if multiple different Y axes are present in the graph window.

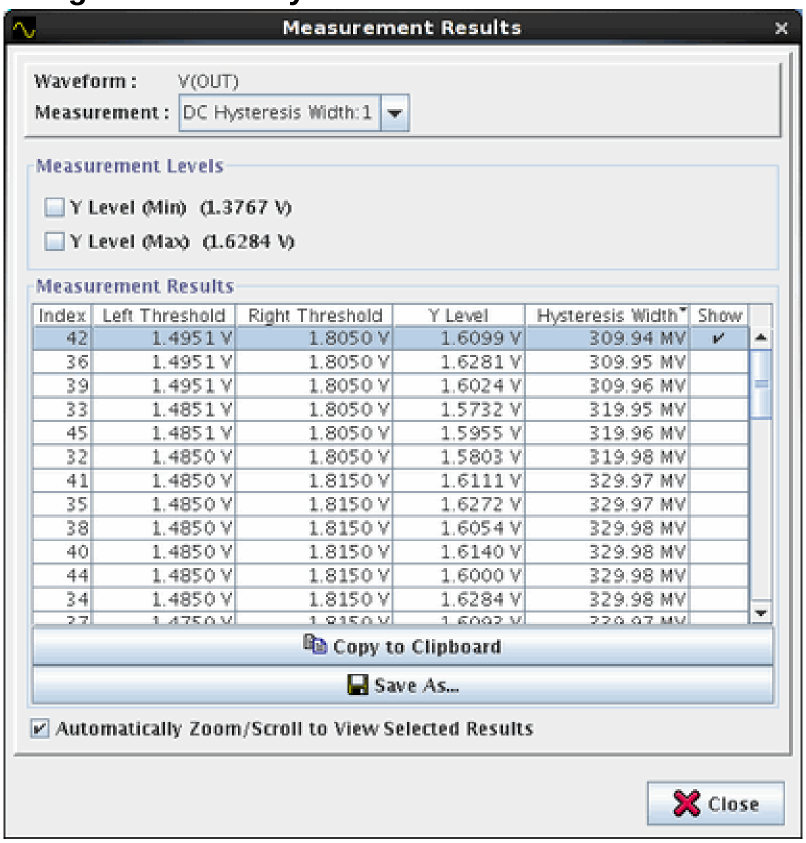
- Specify a Y Level or use the default value. The default value is automatically calculated with (baseline+topline)/2. Click the preview button to display the Topline level on the specified waveform.

Results

The waveforms are annotated with the hysteresis values.

Right-click the measurement annotation and select Measurement Results. The Measurement Results dialog box is displayed.

Figure 6-3. DC Hysteresis Measurement Results



Click on a column name to sort the column. Click on a row to show or hide the corresponding annotation. Ctrl-click to select several rows. The Y level is the value set in the Measurement Tool when not set to <Automatic>, or the individual computed value.

Examples

Example waveform annotations are shown here:

Figure 6-4. Hysteresis Width

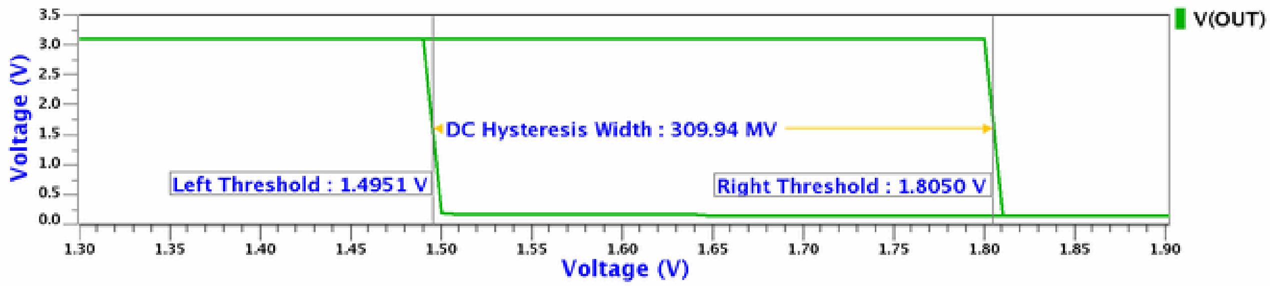


Figure 6-5. Left Threshold

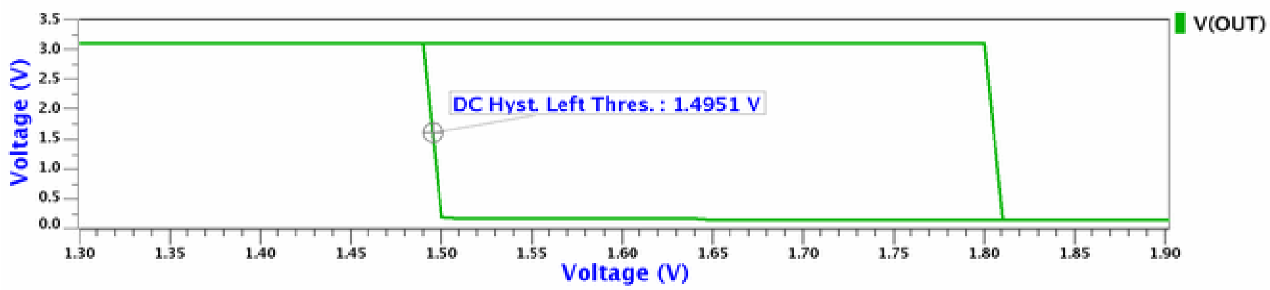
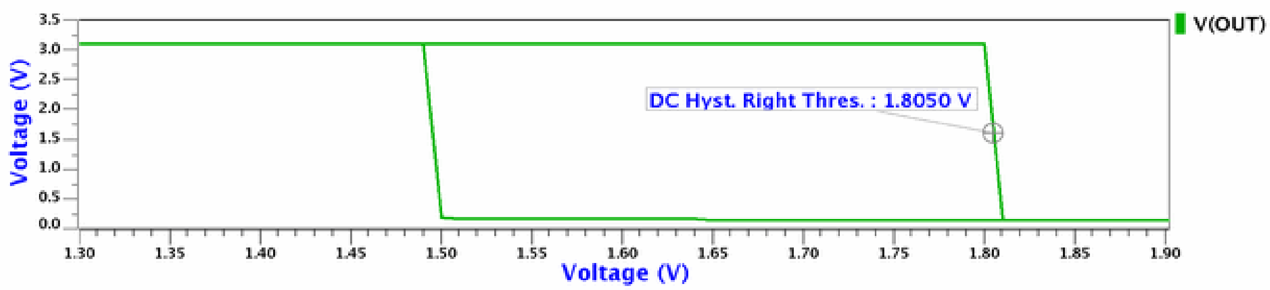
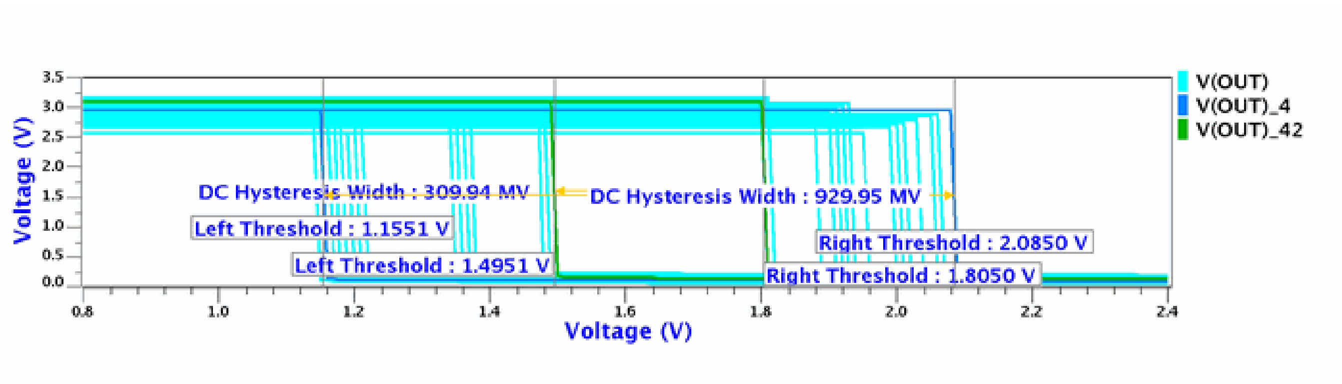


Figure 6-6. Right Threshold



For a compound waveform, EZwave displays two waveforms results: the min and max of the desired measurement (width, left or right threshold).

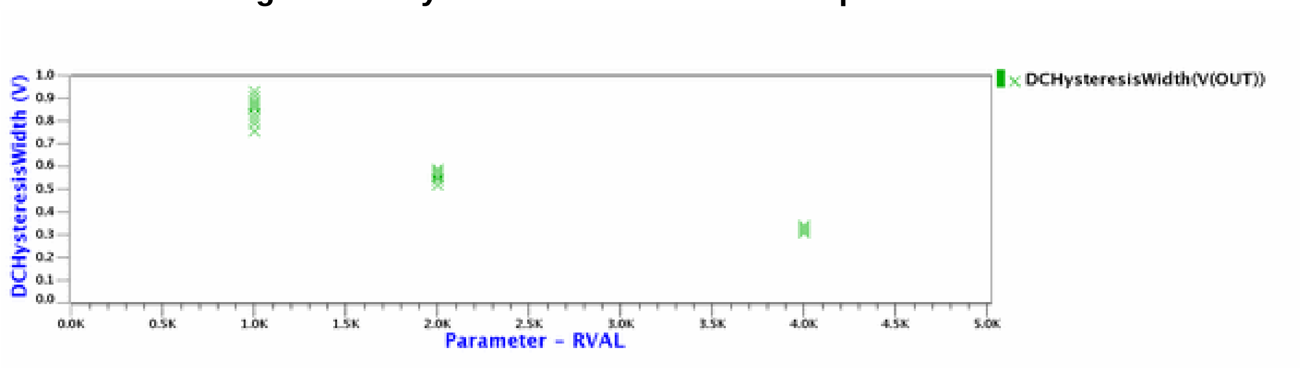
Figure 6-7. Example of Min and Max width (multiple runs)



In this example, the min and max are plotted above the whole compound waveform, to highlight these waveforms.

For a compound waveform, you can plot a new waveform, selecting a simulation parameter as the x axis.

Figure 6-8. Hysteresis Width versus run parameter RVAL



Related Topics

Measurement Tool Dialog Box

Taking Eye Diagram Measurements

This measurement finds and displays the eye width and height of the eye diagram.

The Eye Height is calculated as the difference between the High Level - 3 stdev and the Low Level + 3 stdev.

The Eye Width is calculated as the difference between the Cross Eye + 3 stdev and the Cross Eye - 3 stdev.

Refer to "Eye Diagram Measurement Calculations" on page 206 for details on all measurements relating to the eye diagram.

Tip

A dedicated Eye Diagram Tool is available in EZwave that offers additional functionality. Refer to "Working with Eye Diagrams" on page 197.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Eye Diagram from the left dropdown list.
3. Select Eye Height, Eye Height at X, Eye Width or Eye Width at Y from the right dropdown list in the Measurement Tool.
4. The measurements require specific information in the Measurement Setup section of the Measurement Tool dialog box:

   Select one or more of the available measurements:

   - Eye Height
   - Eye Width
   - X - specify eye width at X (for PAM4 signals, check the Source is PAM box)
   - Y - specify eye width at Y

5. Click **Apply**. The measurement is plotted. You can drag the measurement to display the results at other positions.

Related Topics

Measurement Tool Dialog Box

Frequency Domain Measurements

You can perform a number of frequency domain measurements using the Measurement Tool.

The measurements are described in the following topics:

Measuring Bandpass **286**
Measuring Gain Margin **287**
Measuring Phase Margin **288**

Measuring Bandpass

This measurement finds and displays the bandwidth, the lower band edge, upper band edge, center frequency and quality factor, and the level at which the measurement is made for a bandpass-shaped waveform.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Frequency Domain from the left dropdown list.

3. Select Bandpass from the right dropdown list in the Measurement Tool.

4. This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.

   - Specify a Topline value or use the default value. The default value is automatically calculated. Click the **Preview** button  to display the Topline level on the specified waveform.

   - Specify an Offset value to be applied relative to the Topline value. The Offset is always in dB, and you must also include a sign, minus(-) or plus(+), along with the specified level.

Results

The measurement level, either "Topline - Offset" or "Topline + Offset", is used to determine the following bandpass measurement:

- Lower band cutoff frequency (F-low): the frequency that the frequency response falls crossing the measurement level before the maximum point.

- Higher band cutoff frequency (F-high): the frequency that the frequency response falls crossing the measurement level after the maximum point.

- Center frequency (F-center): calculated as sqrt(F-high * F-low)
- Bandwidth: calculated as (F-high - F-low)
- Quality factor (Q): calculated as (F-center / Bandwidth).

Related Topics

Measurement Tool Dialog Box

Measuring Gain Margin

This measurement finds and displays the gain margin in decibels (dB) and the associated crossover frequencies of a complex waveform. The gain margin is defined as the difference between the gain of the measured waveform and 0 dB (unity gain) at the frequency where the phase shift is -180 degrees (Phase Crossover Frequency). The frequency that gives a gain of 0 dB is the Gain Crossover Frequency.

The gain margin is found by first finding the X value at which the phase is -180 degrees. X is most likely frequency, but it does not have to be. The difference between 1.0 (or 0 dB) and the gain at that frequency is the gain margin. Interpolation between data points is used to find the exact crossing points with the margin values.

The gain margin is the amount of gain increase required to make the loop gain unity at the frequency where the phase angle is -180 degrees. In other words, the gain margin is 1/g if g is the gain at the -180 degrees phase frequency. The frequency at which the phase is -180 degrees is called the Phase Crossover Frequency.

It is generally found that gain margins of 3 dB or more combined with phase margins between 30 and 60 degrees result in a reasonable trade-off between bandwidth and stability.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.
2. Select Frequency Domain from the left dropdown list.
3. Select Gain Margin from the right dropdown list in the Measurement Tool.

   This measurement requires no specific information in the Measurement Setup section of the Measurement Tool dialog box.

Related Topics

Measurement Tool Dialog Box

Measuring Phase Margin

This measurement finds and displays the phase margin of a complex waveform in degrees or radians. The phase margin is defined as the difference in phase between the measured waveform and -180 degrees at the point corresponding to the frequency that gives us a gain of 0 dB (the Gain Crossover Frequency). The frequency where the phase shift is -180 degrees is the Phase Crossover Frequency.

The phase margin is found by first finding the X value (X will most likely be frequency but it does not have to be) at which the magnitude is 1.0 (or 0 dB). The difference between the phase of the response and -180 degrees when the loop gain is 1.0 is phase margin. Interpolation between data points is used to find the exact crossing points with the margin values.

The frequency at which the magnitude is 1.0 is called the Unity-Gain Frequency or Crossover Frequency.

It is generally found that gain margins of 3 dB or more combined with phase margins between 30 and 60 degrees result in a reasonable trade-off between bandwidth and stability.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Frequency Domain from the left dropdown list.
3. Select Phase Margin from the right dropdown list in the Measurement Tool.

   This measurement requires no specific information in the Measurement Setup section of the Measurement Tool dialog box.

Related Topics

Measurement Tool Dialog Box

General Measurements

You can perform a number of general measurements using the Measurement Tool.

The measurements are described in the following topics:

Measuring Average Value 289
Measuring Y Level Crossing 290
Measuring Intersect 291
Measuring Local Max 291
Measuring Local Min 292
Measuring Maximum Value 292
Measuring Minimum Value 293
Measuring Peak to Peak Value 293
Measuring Slope 294
Measuring Slope Intersect 295
Measuring YVal 295

Measuring Average Value

This measurement finds and displays the average value of the specified waveform.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select General from the left dropdown list.

3. Select Average from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - You can select Average, Peak to Peak, Minimum, Maximum, or any combination as desired measurement types in the Measurement Setup area.

Results

The average value of a waveform is calculated as follows:

Figure 6-9. Calculation of the Average Value of a Waveform

$$\frac{1}{X_{max} - X_{min}} \int_{X_{min}}^{X_{max}} (W \cdot dx)$$

where W represents the waveform, and X_{min} and X_{max} are the beginning and the end points for the waveform.

Related Topics

Measurement Tool Dialog Box

Measuring Y Level Crossing

This measurement finds and displays the intersection points between a waveform and a reference Y level. The level is determined as Y = <value>.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select General from the left dropdown list.

3. Select Crossing from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.
   - Specify a Y Level or use the default value. The default value is automatically calculated with (baseline+topline)/2. Click the preview button  to display the Topline level on the specified waveform.
   - Specify a Slope Trigger. Select from Positive and Negative, Positive Only, or Negative Only.

Related Topics

Measurement Tool Dialog Box

Measuring Intersect

This measurement finds and displays the intersection points between two waveforms. Interpolation between data points is used to find the exact intersection points between the two waveforms. The result could be either X data or (X, Y) data.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select General from the left dropdown list.

3. Select Intersect from the right dropdown list in the Measurement Tool.

   This measurement requires two waveforms to be selected, and the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - For Waveform (#1), click to specify the Slope Type - Either Positive or Negative Slope, Positive Slope, or Negative Slope.
   - For Reference Waveform (#2), specify the Slope Relationship. Click the dropdown list next to Slope Relationship and select between Non-Inverting and Inverting.
   - To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of "Intersect" vs "Time".

     To view the other measurement results after the measurement is performed, select the displayed result marker, right-click the mouse to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

Related Topics

Measurement Tool Dialog Box

Measuring Local Max

This measurement finds and displays the local maxima of the waveform.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select General from the left dropdown list.

3. Select Local Max from the right dropdown list in the Measurement Tool.

This measurement requires the following specific information in the Measurement Results section of the Measurement Tool dialog box:

- Select Annotate Waveform(s) with Measurement Results to display the local maximum for the specified region.

  Or:

- Select Plot New Waveform of “Local Max” vs “Time” to create a new waveform in the active graph window that shows how the local maximum changes with time.

Related Topics

Measurement Tool Dialog Box

Measuring Local Min

This measurement finds and displays the local minima of the waveform.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select General from the left dropdown list.
3. Select Local Min from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Results section of the Measurement Tool dialog box:

   - Select Annotate Waveform(s) with Measurement Results to display the local maximum for the specified region.

     Or:

   - Select Plot New Waveform of “Local Min” vs “Time” to create a new waveform in the active graph window that shows how the local maximum changes with time.

Related Topics

Measurement Tool Dialog Box

Measuring Maximum Value

This measurement finds and displays the maximum value of the waveform.

Procedure

1. Choose **Tools** > **Measurement Tool**.

The Measurement Tool Dialog Box opens.

2. Select General from the left dropdown list.
3. Select Maximum from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - In addition to the Maximum measurement, you can select additional measurement types in the Measurement Setup area.

Related Topics

Measurement Tool Dialog Box

Measuring Minimum Value

This measurement finds and displays the minimum value of the waveform.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select General from the left dropdown list.
3. Select Minimum from the right dropdown list in the Measurement Tool.
4. This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - In addition to the Minimum measurement, you can select additional measurement types in the Measurement Setup area.

Related Topics

Measurement Tool Dialog Box

Measuring Peak to Peak Value

This measurement finds and displays the peak-to-peak value of the waveform.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select **General** from the left dropdown list.
3. Select **Peak to Peak** from the right dropdown list in the Measurement Tool.

This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

- In addition to the Maximum measurement, you can select additional measurement types in the Measurement Setup area.

Related Topics

Measurement Tool Dialog Box

Measuring Slope

This measurement finds and displays the slope value of the waveform at a specified X value. For frequency domain waveforms, this measurement can be displayed as a value per decade or a value per octave.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select General from the left dropdown list.

3. Select Slope from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - Provide an absolute value as the X value and the measurement returns the Slope value at that coordinate.

     You may utilize the *cursor* to input the X value: move the mouse pointer close to the active cursor, right-click to display the Cursor Popup Menus, and select **Copy X to Clipboard** from the popup menu. Click in the text window of X in the Measurement Setup field, right-click to display the popup menu and select **Paste** to enter the X value.

   - Click to specify the Slope Option. This field is visible only in the Frequency Domain category. You may choose to display the Slope measurement result normally (None), display the Slope per decade (Decade), or display the Slope per octave (Octave).

Related Topics

Measurement Tool Dialog Box

Measuring Slope Intersect

This measurement finds and displays the slope intersect of two waveforms at specified X values.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select **General** from the left dropdown list.

3. Select **Slope Intersect** from the right dropdown list in the Measurement Tool.

   This measurement requires two waveforms to be selected and the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - For each waveform, provide an absolute value as the X value and the measurement returns the slope value at that coordinate and the slope intersect value between the two slopes.

     You may utilize the cursor to input the X value: move the mouse pointer close to the active cursor, right-click to display the Cursor Popup Menus, and select **Copy X to Clipboard** from the popup menu. Click in the text window of X in the Measurement Setup field, right-click to display the popup menu and select **Paste** to enter the X value.

   - Once applied, markers on each of the specified waveforms show the slopes at the selected points, and a marker for the slope intersect shows where the two tangent lines to the selected points intersect.

   - To view the slopes and slope intersects at other points along the curves, click on a slope marker and drag it along the waveform. The displayed slope measurement, tangent lines, and slope intersect measurement change according to the new position along the waveform.

Related Topics

Measurement Tool Dialog Box

slopeintersect

Measuring YVal

This measurement finds the Y value(s) at a given X coordinate of a waveform.

For single waveforms, this measurement places a marker at the specified X coordinate that indicates its Y value.

For compound waveforms, you can also plot the results of the Y-value measurements against a swept parameter or index.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select General from the left dropdown list.

3. Select Yval from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - Provide an absolute value as the X value, and the measurement returns the Y value(s) at that coordinate.

     Utilize the cursor to input the X value: move the mouse pointer close to the active cursor, right-click to display the Cursor Popup Menus, and select **Copy X to Clipboard** from the popup menu. Click in the text window of X in the Measurement Setup field, right-click to display the popup menu and select **Paste** to enter the X value.

   - The following options are available when applying the measurement to a compound waveform:

   a. To add a marker that displays the Y value of the waveform at the specified X coordinate, select Annotate Waveform(s) with Measurement Results.

   b. To plot the Y values against a swept parameter, select the name of the parameter from the dropdown list next to **Plot New Waveform of** “**Y**”**vs**. When you click **Apply**, this will display the results in a new graph window with the parameter values along the x axis and the Y values along the y axis.

   c. To plot the Y values against the index of the swept parameters, select **Index** from the dropdown list next to **Plot New Waveform of** “**Y**”**vs**. When you click **Apply**, this will display the results in a new graph window with the parameter index along the x axis and the Y values along the y axis.

Related Topics

Measurement Tool Dialog Box

Taking Statistical Measurements

You can perform a number of statistical measurements using the Measurement Tool.

Table 6-2. Statistical Measurements

Maximum	Mean	Mean + 3 Std Dev	Mean - 3 Std Dev
Minimum	RMS	RMS AC	RMS Accurate
RMS Noise	RMS Tran	Standard Deviation	

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Measurement from the left dropdown list.

3. Select either **Maximum, Mean, Mean +3 Std Dev, Mean -3 Std Dev, Minimum, Rms, Rms AC, RMS Accurate, Rms Noise, Rms Tran,** or**Standard Dev** from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - Select one or more desired measurement types in the Measurement Setup area.

Results

The mean value of a waveform is calculated as follows:

Figure 6-10. Mean Value of a Waveform Calculation

:

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

where N is the number of data points, and W_j represents the individual data points of the waveform.

The RMS value of a waveform *v* is calculated as follows:

Figure 6-11. The RMS AC Calculation:

$$\sqrt{\int(v \times v)}$$

Figure 6-12. The RMS Noise Calculation:

$$\sqrt{\Sigma(v \times v)}$$

Figure 6-13. The RMS Tran Calculation:

$$\sqrt{\frac{\int(v \times v)}{\text{time interval}}}$$

Related Topics

Measurement Tool Dialog Box

Time Domain Measurements

You can perform a number of time domain measurements using the Measurement Tool.

The measurements are described in the following topics:

Measuring Delay 299
Measuring Duty Cycle 301
Measuring Falltime 302
Measuring Frequency 303
Measuring Overshoot 304
Measuring Period 305
Measuring Pulse Width 306
Measuring Risetime 307
Measuring Settle Time 308
Measuring Slew Rate 309
Measuring Undershoot 310

Measuring Delay

This measurement finds and displays the delay between the edges on one or two waveforms relative to default (automatically calculated) or user-specified topline and baseline levels for both the measured waveform and the reference waveform.

The delay is calculated as the difference in time between two edges on one or two waveforms. These two waveforms used for the measurement do not have to be the same waveform type. For analog waveforms, the delay can be measured at any percentage level relative to the Topline level and Baseline level of either waveform. For example, to measure the delay from the 50% level of one waveform to the 90% level of the other waveform. The analog waveforms can be assumed to be voltage waveforms only.

It is assumed that the rising or falling edge on the reference waveform (#2) causes the corresponding (rising or falling) edge on the measured waveform (#1) so that the reference edge occurs prior to the measured edge.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.

3. Select Delay from the right dropdown list in the Measurement Tool.

This measurement requires two waveforms to be selected and the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

- For Waveform (#1):

a. Specify a Topline value or use the default value. The default value is automatically calculated.

b. Specify a Baseline value or use the default value. The default value is automatically calculated.

Tip

Click the preview button to display the topline or baseline level on the waveform.

c. Click the dropdown list next to Delay Level to make your selection of percentage relative to the Topline and Baseline value.

d. Click to specify the Edge Trigger that the measurement starts from: Either Rising or Falling Edge (depending on which comes first in the specified measurement window), Rising Edge, or Falling Edge.

- For Reference Waveform (#2):

a. Specify a reference Topline value or use the default value. The default value is automatically calculated.

b. Specify a reference Baseline value or use the default value. The default value is automatically calculated.

c. Click the dropdown list next to Delay Level to make your selection of percentage relative to the Topline and Baseline value of reference waveform.

d. The measurement may starts on the reference waveform at the previous edge with the same polarity (Non-Inverting) as the measured waveform (#1) or the opposite polarity (Inverting). Click the dropdown list next to Edge Relationship and make your selection accordingly.

e. Select Find the Closest Reference Edge to display the reference edge nearest to the measured edge.

- To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of “Delay” vs “Time”.

  To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

Related Topics

Measurement Tool Dialog Box

Measuring Duty Cycle

This measurement finds and displays the duty cycle of a periodic waveform relative to default (automatic calculated) or user-specified topline and baseline levels. The duty cycle of the periodic waveform is the ratio of the “high” portion of the waveform to the length of the period. The high portion of a cycle is the duration of the positive pulse measured at the middle level.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.

3. Select Duty Cycle from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.
   - Specify a Topline value or use the default value. The default value is automatically calculated.
   - Specify a Baseline value or use the default value. The default value is automatically calculated.

     Tip

     Click the preview button to display the topline or baseline level on the specified waveform.

   - Click to specify the Edge Trigger that the measurement starts from: Either Rising or Falling Edge (depending on which comes first in the specified measurement window), Rising Edge, or Falling Edge.
   - To present the multiple measurement results, check either Annotate Waveform with Result Marker(s) or Plot New Waveform of “Duty Cycle” vs “Time”.

     To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

Related Topics

Measurement Tool Dialog Box

Measuring Falltime

This measurement finds and displays the falltime between specified upper and lower levels of a waveform. The falltime is calculated as the difference in time when the waveform falls from the upper level to the lower level.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.

3. Select Falltime from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.
   - Specify a Topline value or use the default value. The default value is automatically calculated.
   - Specify a Baseline value or use the default value. The default value is automatically calculated.

   Tip

   Click the preview button to display the Topline or Baseline level on the specified waveform.

   - Click the dropdown lists next to Lower / Upper to make your selection of percentage relative to the Topline and Baseline value of the specified waveform. The left dropdown list specifies the Lower level and the right dropdown list specifies the Upper level, while the following Levels are Relative to the Topline and Baseline remains checked.
   - Clear the Levels are Relative to the Topline and Baseline checkbox if you want to specify absolute values as the Lower and Upper levels.
   - To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of “Falltime” vs “Time”.

To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

Related Topics

Measurement Tool Dialog Box

Measuring Frequency

This measurement finds and displays the frequency of a periodic waveform relative to default or specified topline and baseline levels.

The frequency is calculated as the reciprocal of the period (refer to "Measuring Period" on page 305).

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.

3. Select Frequency from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.

   - Specify a **Topline** value or use the default value. The default value is automatically calculated.

   - Specify a **Baseline** value or use the default value. The default value is automatically calculated.

     Tip

     Click the **Preview** button to display the Topline or Baseline level on the waveform.

   - Click to specify the Edge Trigger that the measurement starts from: Either Rising or Falling Edge (depending on which comes first in the specified measurement window), Rising Edge, or Falling Edge.

   - To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of "Frequency" vs "Time".

To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

Related Topics

Measurement Tool Dialog Box

Measuring Overshoot

This measurement finds and displays the overshoot value of a waveform. The overshoot value is calculated as the difference between the maximum point and the topline level of the waveform.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.

3. Select Overshoot from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.
   - Specify a Topline value or use the default value. The default value is automatically calculated.
   - Specify a Baseline value or use the default value. The default value is automatically calculated.

   Tip

   Click the **Preview** button to display the Topline or Baseline level on the specified waveform.

   - To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of “Overshoot” vs “Time”.

     To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

Note

Overshoot uses the topline or baseline so that there is just one single measure per upper or lower state. This is to avoid many measures in situations where, for example, the waveform is ringing before reaching its steady state.

Related Topics

Measurement Tool Dialog Box

Measuring Period

This measurement finds and displays the period of a periodic waveform relative to default or specified topline and baseline levels.

The period is calculated as the difference in time between two consecutive edges of the waveform of the same polarity (rising edge to rising edge or falling edge to falling edge).

For analog waveforms, the period is always measured from the middle level of one edge to the middle level of the next edge of the same polarity. The waveform shape is not necessarily square.

For digital waveforms, the period is measured from the beginning X value of one edge to the beginning X value of the next edge of the same polarity.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.

3. Select Period from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.
   - Specify a Topline value or use the default value. The default value is automatically calculated.
   - Specify a Baseline value or use the default value. The default value is automatically calculated.

Tip

Click the **Preview** button to display the Topline or Baseline level on the specified waveform.

- Click to specify the Edge Trigger that the measurement starts from: Either Rising or Falling Edge (depending on which comes first in the specified measurement window), Rising Edge, or Falling Edge.
- To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of “Period” vs “Time”.

  To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

Related Topics

Measurement Tool Dialog Box

Measuring Pulse Width

This measurement finds and displays the pulse width of a waveform relative to default or specified topline and baseline levels.

The pulse width, for a “positive” pulse, is the difference in time between the middle level of a rising edge and the middle level of the next falling edge on the waveform. For a “negative” pulse, the pulse is the time difference between the middle level of a falling edge and the middle level of the next rising edge.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.

3. Select Pulse Width from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.
   - Specify a Topline value or use the default value. The default value is automatically calculated.
   - Specify a Baseline value or use the default value. The default value is automatically calculated.

Tip

Click the **Preview** button to display the Topline or Baseline level on the specified waveform.

- Click to specify the Pulse Type from Either Positive or Negative Pulse, Positive Pulse, or Negative Pulse.
- To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of “Pulse Width” vs “Time”.

  To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

Related Topics

Measurement Tool Dialog Box

Measuring Risetime

This measurement finds and displays the risetime between selected upper/lower levels of a waveform.

The risetime is the difference in time when the waveform rises from the lower level to the upper level.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.
3. Select Risetime from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.
   - Specify a Topline value or use the default value. The default value is automatically calculated.
   - Specify a Baseline value or use the default value. The default value is automatically calculated.

Tip

Click the **Preview** button to display the Topline or Baseline level on the specified waveform.

- Click the dropdown lists next to Lower / Upper to make your selection of percentage relative to the Topline and Baseline value of the specified waveform. The left dropdown list specifies the Lower level and the right dropdown list specifies the Upper level, while the following Levels are Relative to the Topline and Baseline remains checked.
- Clear Levels are Relative to the Topline and Baseline if you want to specify absolute values as the Lower and Upper levels.
- To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of "Risetime" vs "Time".

  To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Results Window.

Related Topics

Measurement Tool Dialog Box

Measuring Settle Time

This measurement finds and displays the settle time of a waveform with respect to default or specified steady state level and a specified tolerance.

The size of settle band is specified as the tolerance level on either side of steady state level. The settle time is the last time point that the waveform crosses the settle band, either the positive level or the negative level of tolerance, from out of bound to inner bound.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.
3. Select Settle Time from the right dropdown list in the Measurement Tool.

   This measurement requires two waveforms to be selected and the following specific information in the Measurement Setup section of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.

- Specify a Steady State Level value or use the default value. The default value is automatically calculated. Click the **Preview** button to display the Steady State level on the specified waveform.
- Click the dropdown lists next to Tolerance to make your selection of percentage relative to the amplitude of the specified waveform, while the following Tolerance is Percentage of Waveform Amplitude remains checked.
- Clear Tolerance is Percentage of Waveform Amplitude if you want to specify an absolute value as the Tolerance.

Related Topics

Measurement Tool Dialog Box

Measuring Slew Rate

This measurement finds and displays the slew rate of the waveform. The slew rate is the difference between the upper and lower levels of the waveform divided by the risetime of the rising edge (or the falltime of the falling edge).

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.
3. Select Slew Rate from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.
   - Specify a Topline value or use the default value. The default value is automatically calculated.
   - Specify a Baseline value or use the default value. The default value is automatically calculated.

     Tip

     Click the **Preview** button to display the Topline or Baseline level on the specified waveform.

   - Click the dropdown lists next to Lower / Upper to make your selection of percentage relative to the Topline and Baseline value of the specified waveform. The left dropdown list specifies the Lower level and the right dropdown list specifies the

Upper level, while the following Levels are Relative to the Topline and Baseline remains checked.

- Clear the Levels are Relative to the Topline and Baseline checkbox if you want to specify the Lower and Upper levels as absolute values.
- Click to specify the Edge Trigger that the measurement starts from: Either Rising or Falling Edge (depending on which comes first in the specified measurement window), Rising Edge, or Falling Edge.
- To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of “Slew Rate” vs “Time”.

  To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

Related Topics

Measurement Tool Dialog Box

Measuring Undershoot

This measurement finds and displays the undershoot value of a waveform. The undershoot value is calculated as the difference between the minimum point and the baseline level of the waveform.

Procedure

1. Choose **Tools** > **Measurement Tool**.

   The Measurement Tool Dialog Box opens.

2. Select Time Domain from the left dropdown list.
3. Select Undershoot from the right dropdown list in the Measurement Tool.

   This measurement requires the following specific information in the Measurement Setup and Measurement Results sections of the Measurement Tool dialog box:

   - Specify the Y Axis to use if multiple different Y axes are present in the graph window.
   - Specify a Topline value or use the default value. The default value is automatically calculated.
   - Specify a Baseline value or use the default value. The default value is automatically calculated.

Note

Click the **Preview** button to display the Topline or Baseline level on the specified waveform.

- To present the multiple measurement results, check either Annotate Waveform(s) with Result Marker(s) or Plot New Waveform of “Undershoot” vs “Time”.

  To view the other measurement results after the measurement is performed, select the displayed result marker, right-click to display the popup menu, and select **Measurement Results** from the menu to display the Measurement Result Window.

  Note

  Undershoot uses the baseline or topline so that there is just one single measure per lower or upper state. This is to avoid many measures in situations where, for example, the waveform is ringing before reaching its steady state.

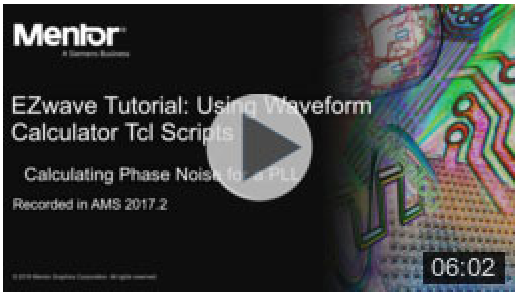
Related Topics

Measurement Tool Dialog Box

Waveform Calculator

The Waveform Calculator enables you to post-process waveforms for advanced analyses or debugging. The Calculator has a comprehensive graphical interface.

To watch a tutorial that shows you how to use the Waveform Calculator Tcl scripts to calculate phase noise for a PLL, see the video:



Restriction

wreal waveforms, assertions, and Safe Operating Area assertions are not supported in the Waveform Calculator.

Note

The Waveform Calculator supports either IEEE notation (the default) or SPICE notation. You can choose which notation to use. Refer to "Waveform Calculator General Options" on page 547.

Waveform Calculator GUI

Using Expressions in the Waveform Calculator 313

Waveform Calculator Shell Commands 314

Using Buttons in the Waveform Calculator 315

Using Built-In Functions in the Waveform Calculator 317

Using and Editing User-Defined Functions in the Waveform Calculator 319

Creating a Tcl Script From the Waveform Calculator History 320

Updating User-Defined Function Help Documentation 323

Using the Measurement Tool Functions in the Waveform Calculator 325

Built-In Functions 332

Calculator Buttons 349

Using Expressions in the Waveform Calculator

This topic describes how you can use expressions in the waveform calculator to post-process waveforms for advanced analyses or debugging.

Procedure

1. Open the Waveform Calculator application window with the calculator button from the toolbar or select **Tools** > **Waveform Calculator**.
2. Add entries to the Shell panel of the calculator using one of the following methods, and then click the **Eval** button or press Enter.
   - Add waveform names using the **Add Selected Waveforms** icon
   - Add operations using the buttons (refer to "Using Buttons in the Waveform Calculator" on page 315)
   - Add functions using the built-in functions (refer to "Using Built-In Functions in the Waveform Calculator" on page 317)
   - Add functions using user-defined functions (refer to "Using and Editing User-Defined Functions in the Waveform Calculator" on page 319)
   - Add functions from the **Measurement Tool** (refer to "Using the Measurement Tool Functions in the Waveform Calculator" on page 325).

   Tip

   To view function help, click . You can also use the following command in the shell window:

   ```
   help [syntax|description|desc|parameters|params] <functionname>
   ```

   The results are displayed in the **Workspace** tab of the Chooser panel (along with an automatically generated Alias) and also in the Shell panel.
3. (Optional) To use a previous result in a new calculation:
   a. Right-click a result in the **Workspace** tab.
   b. Select Copy Alias or Copy Value to copy the result.
   c. Paste this into a new expression.
4. If the results include a waveform, click the **Plot** icon to plot the resulting waveform in the waveform viewer display.

   You need to save any resulting waveforms. Refer to "Saving Post-Processed Waveforms" on page 411 for more information.

5. (Optional) If you want to store an expression as a Tcl file for later use, click the **Store** button.

   This opens the Create Script dialog box. Refer to "Creating a Tcl Script From the Waveform Calculator History" on page 320 for more information on how to use this dialog box. You can open expressions previously stored in a Tcl file using the **Recall** button. Restored expressions are listed in the **History** tab of the Chooser panel.

6. (Optional) Copy or evaluate any expression listed in the **History** tab of the Chooser panel:

   a. Right-click an expression in the **History** tab.

   b. Select **Copy** or **Evaluate**.

Waveform Calculator Shell Commands

This topic describes the commands that you can use in the Waveform Calculator shell panel.

help

```
help [syntax|description|desc|parameters|params|return] <functionname>
```

Displays the help of a function in the shell. Use the options to limit the displayed information for the function to either Syntax, Description, Parameters or Return. For example:

```
help syntax fft

help description fft

help parameters fft
```

doc

```
doc <functionname>
```

Displays the help of a function in the help panel.

search

```
search [all|syntax|name|title|description|desc|parameters|params|return]
<string>
```

Searches the function help documentation. Use the options to limit the search to any combination of function fields. For example:

```
search title fft

search description value
```

```
search syntax|title|desc multple word search

search title|description "multiple word search"
```

Note

The *all* option does not return the parameters (to reduce the size of the results). To return everything including the parameters, use *all|params*.

Tip

To display the function help for one of the search results, hold down the Ctrl key, mouse over the function name to highlight it and click the left mouse button.

history

```
history
```

Displays all entered expressions, and a number associated with each expression.

```
history N
```

Displays the last "N" entered expressions.

Tip

To recall an expression from the history, type !N where N is the number of the expression.

clear

Clears the calculator shell panel.

clear history

Clears the user history.

clear workspace

Clears the user workspace.

clear all

Clears the calculator shell, history and workspace.

Using Buttons in the Waveform Calculator

You can use the buttons in the Waveform Calculator to enter expressions into the Waveform Calculator.

Procedure

1. Open the Waveform Calculator application window with the calculator button  from the toolbar or select **Tools > Waveform Calculator**.

2. If the Button panel is not shown, click the  button. Select the function category from the dropdown list. The Button panel changes based on the level you choose from the dropdown list:

    - Complex Buttons
    - Jitter Buttons
    - Logic Buttons
    - Phase Noise Buttons
    - RF Buttons
    - Signal Processing Buttons
    - Statistical Buttons
    - Trigonometric Buttons

    Tip

    The Button panel does not include a button for every function or operator. If a button is not available for a particular function, double-click the function from the Functions list of the Chooser panel. Refer to "Using Built-In Functions in the Waveform Calculator" on page 317.

3. Click on a button in the Button panel. One of the following actions occurs, depending on the particular button:

    a. A dialog box is displayed, specifically designed for the function, with an associated **Help** button. Complete the required fields and click **OK** to evaluate the function.

    b. The function appears in the Shell panel. Place the cursor in between the parentheses of the function and type in the required parameter(s). To specify a waveform, do one of the following:

    Right-Click Method

    i. Select the waveform or waveform label in the active graph window.

    ii. Right-click to display the Waveform popup menu, and select **Copy**.

    iii. Place the cursor in between the parentheses of the function in the Shell panel, right-click to display the popup menu, and select **Paste** to add the waveform name in between the parentheses of the function.

    Drag and Drop Method

i. Select the waveform label in the *active* graph window.

ii. Hold the left mouse button down, drag the label between the parentheses of the function in the Shell panel, and release the mouse button.

Tip

If you hold down Ctrl, move the cursor over a function name in the Shell panel until it is highlighted and click the left mouse button, a panel appears showing the parameters for the selected function for easy data entry.

Press Enter (or click the **Eval** button) to evaluate the function.

Related Topics

Waveform Calculator GUI

Using Built-In Functions in the Waveform Calculator

You can use a large number of built-in functions in the Waveform Calculator.

Procedure

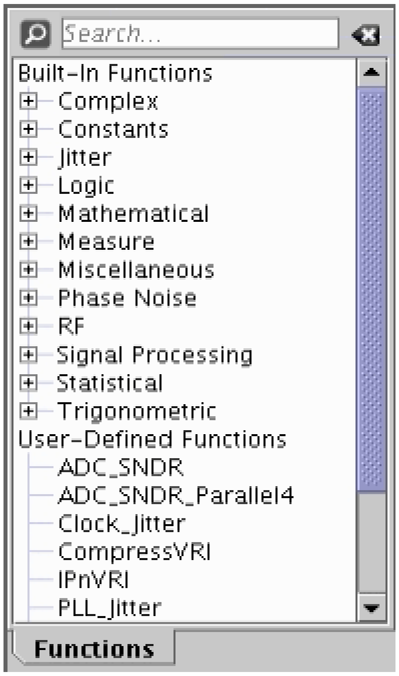
1. Open the Waveform Calculator application window with the calculator button from the toolbar or select **Tools** > **Waveform Calculator**.
2. Choose a function from the Functions list in the Chooser panel, shown in Figure 6-14. You can manually locate a function by clicking on the + next to each of the types of functions to expand the list.

   Alternatively, you can type the first few letters of the function you are looking for into the Search text field. The function lists automatically expand to display any functions that match your typing.

   Tip

   If you press Enter to perform a search, the search is added to the search history list. Click the icon to access the search history. The history maintains up to ten entries. Select the **Clear History** option to reset the history.

Figure 6-14. Waveform Calculator Functions List

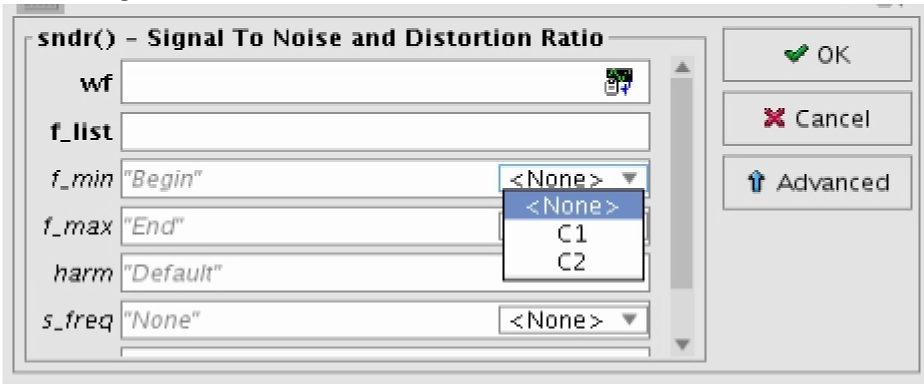


3. Double-click a function from the Functions list of the Chooser panel. The Button panel (if visible) is replaced by a panel showing the parameters for the selected function for easy data entry, as shown in Figure 6-15. The default values for each of the parameter are shown in the fields as gray text.

   Where a parameter has several pre-defined values, you can use the dropdown list displayed to the right of the field to select one of the available options.

   If you use cursors on a plotted waveform, you can use the cursor values to complete appropriate fields. A dropdown list is displayed to the right of appropriate fields, showing the cursors that may be selected.

Figure 6-15. Waveform Calculator Function Fields



Note

Some functions have a dedicated dialog box that provides additional options tailored specifically to the function. In this case a button, **Advanced**, is visible. Click **Advanced**, complete the required fields, and then click **OK**.

4. Complete the required fields and click **OK** to add the command into the Shell panel. You can also edit the function parameters directly in the Shell panel. Press Enter (or click the **Eval** button) to evaluate the function.

Using and Editing User-Defined Functions in the Waveform Calculator

The waveform calculator enables you to open, display and edit functions you have written in Tcl scripts.

The waveform calculator lists these functions under User-Defined Functions in the Functions list of the Chooser panel.

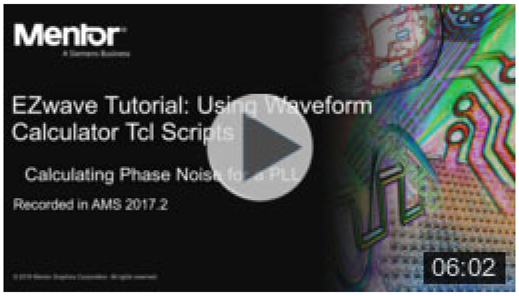
Tip

It is possible to load user-defined functions automatically. See the Load User Extension Files at Startup option in "Waveform Calculator General Options".

Note

Some example Tcl scripts that provide equivalent functions to legacy AFS WaveCrave CalcPAD scripts are available in the EZwave examples directory. Refer to "Waveform Calculator Example Tcl Scripts" on page 1224.

To watch a tutorial that shows you how to use the Waveform Calculator Tcl scripts to calculate phase noise for a PLL, see the video:



Procedure

1. Open the Waveform Calculator dialog box with the **Calculator** button from the toolbar or select **Tools** > **Waveform Calculator**.

2. On the Waveform Calculator dialog box choose **File** > **Open Custom File Function**. A browser window displays.
3. Select the desired Tcl file. The waveform calculator lists the function under User-Defined Functions in the Functions list.
4. Use the user-defined function in the same manner as a built-in function.

   Refer to "Using Built-In Functions in the Waveform Calculator" on page 317.
5. To search for a specific function, type the first few letters of the function you are looking for into the Search text field. The function lists automatically expand to display any functions that match your typing.

   Tip

   If you press Enter to perform a search, the search is added to the search history list. Click on the icon to access the search history. The history maintains up to ten entries. Select the **Clear History** option to reset the history.
6. You can view or edit a user-defined function. Right-click the function and select **Edit** to access the TCL File Viewer.

   Refer to the "Tcl File Viewer Dialog Box" on page 648.
7. You can right-click a user-defined function and select **Edit In System Editor** if you want to use a third-party editor tool (Linux only), specified using the AMS_EDITOR system variable.
8. You can right-click a user-defined function and select **Update UDF Help** to open the Update UDF Help dialog box to add help documentation to the function.

   Refer to "Updating User-Defined Function Help Documentation" on page 323.

Creating a Tcl Script From the Waveform Calculator History

This topic describes how you can create a Tcl scripts from expressions listed in the Waveform Calculator **History** tab.

Prerequisites

- You must have evaluated at least one expression using the waveform calculator so that it appears in the Chooser panel **History** tab.

Procedure

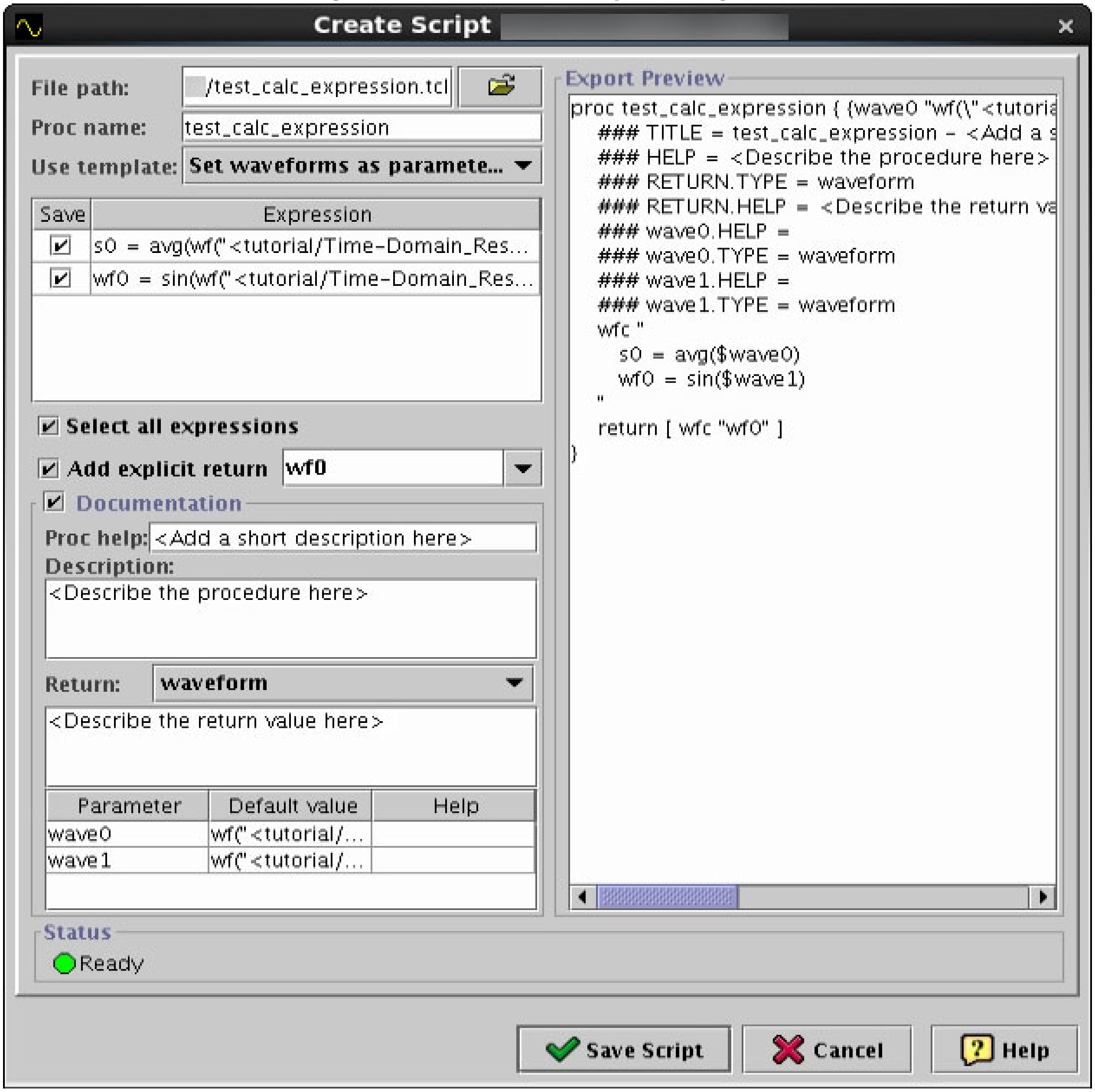
1. On the Waveform Calculator dialog box, select **File** > **Save Evaluated Expressions**. Alternatively, click the **Store** button.

The Create Script dialog box opens, with all of the evaluated expressions listed in the Expression column of the table and the script displayed in the Export Preview panel.

Tip

To load a subset of the available expressions, select the required expressions in the **History** tab, right click and choose **Create Script**.

Figure 6-16. Create Script Dialog Box



2. In the Proc name field, type the name for the procedure.

Tip
If you do not want the procedure to appear in the Waveform Calculator tree, prefix the procedure name with two underscores "__".

If you want to...	Do the following:
Choose the expressions to use in the script.	Select the expressions in the Save column of the table. Use the Select all expressions checkbox to select or deselect all the listed expressions.
Specify an explicit return value in the expression.	Click the Add explicit return checkbox and select the required return value from the dropdown list. A line to return the specified value is added to the script, and shown in the Export Preview panel.
Make the database name generic.	Choose Set generic database name from the Use template dropdown list. The database name associated with each waveform is replaced by "sim" and the script may then be used with any database that contains the same waveform names.
Make the script generic.	Choose Set waveforms as parameters from the Use template dropdown list.
Run the script on a series of databases.	Choose Generate database loop from the Use template dropdown list. The script is modified and a loop is inserted into the code to run the script on a list of databases. The default list contains the names of all databases used in the selected expressions.
Add documentation help text.	Click the Documentation checkbox and specify the Proc help text (a short description) and a (longer) Description for the Tcl Script. You can also document the return value; choose the type from the Return dropdown list (for example waveform, double) and type a description for the return value. If you have selected Set waveforms as parameters from the Use template dropdown list, the parameters appear in the Parameter column of the table, where you can also edit their names. The Default value column of the table initially lists the default waveforms assigned to each parameter, but may be edited. You can also add a description for each parameter in the Help column. The details are added to the script in the Export Preview panel

3. In the File path field, specify a path and filename for the new script. Alternatively, click  to open the Save Script dialog box. Browse to the required path and type a File Name, and then click **Save** to create the new file. If you update the script, click the **Save Script** button on the Create Script dialog box.

   When you subsequently reload the script, using **File** > **Open Custom Function File** or the **Recall** button, it appears in the User-Defined Functions section of the Functions list and the documentation you have entered is visible when you view the Function Help .

Updating User-Defined Function Help Documentation

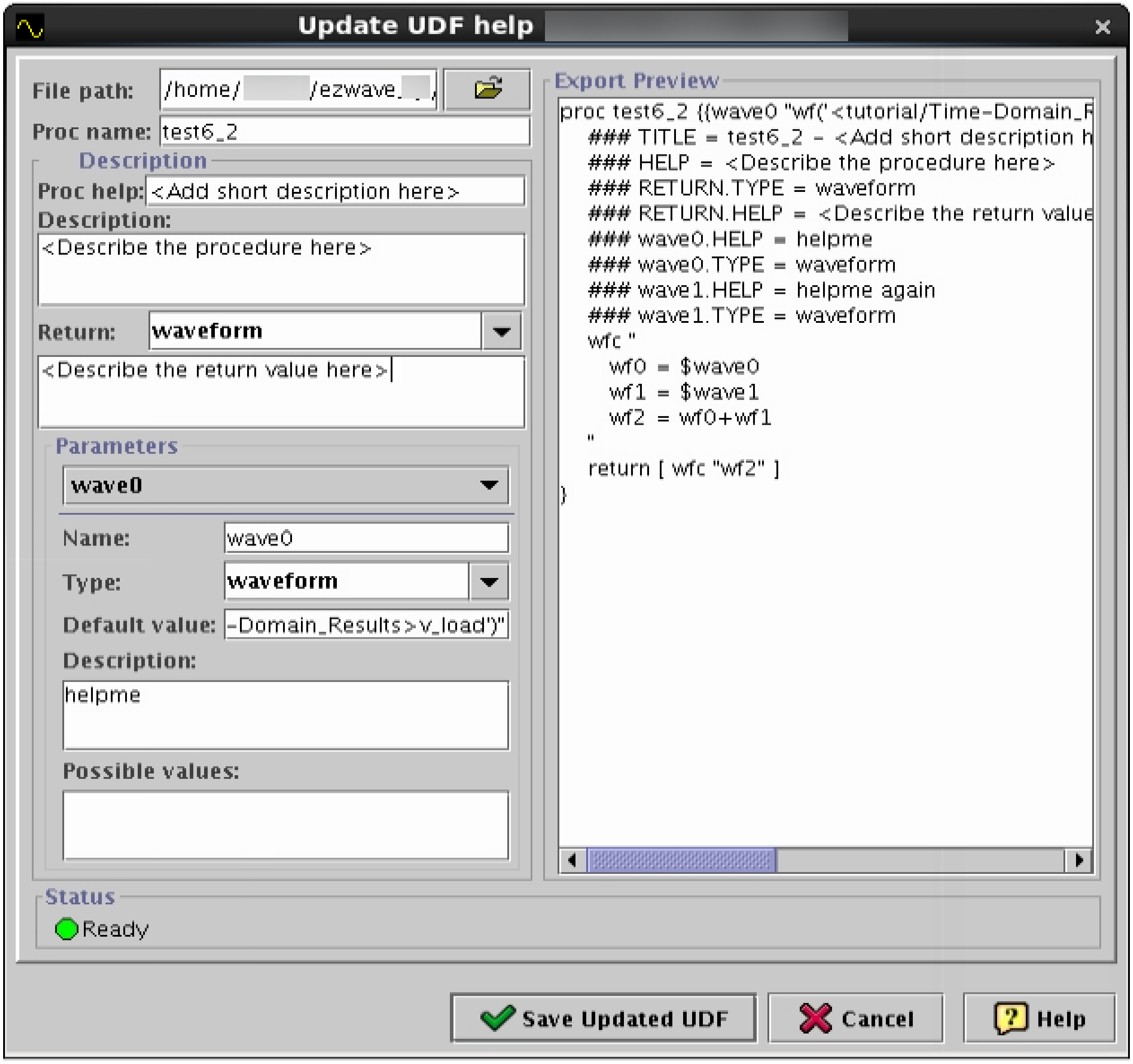
This topic describes how you can add help documentation to user-defined functions listed in the Waveform Calculator Functions list.

Procedure

1. On the Waveform Calculator dialog box, right-click the required user-defined function in the Functions list, and select **Update UDF Help**.

   The Update UDF Help dialog box opens, with the script displayed in the Export Preview panel.

Figure 6-17. Update UDF Help Dialog Box



2. Specify the Proc help text (a short description) and a (longer) Description for the user-defined function. You can also document the return value; choose the type from the Return dropdown list (for example waveform, double) and type a Description for the return value. The details are added to the script in the Export Preview panel.

3. (Optional) Specify a Name, Type (using the dropdown menu), Default value, Description and Possible values for each of the parameters listed in the Parameters dropdown list. The details are added to the script in the Export Preview panel.

4. Click **Save Updated UDF** to save the help documentation.

Using the Measurement Tool Functions in the Waveform Calculator

The functions from the Measurement Tool can be used to evaluate waveforms in the Waveform Calculator or in a Tcl script file.

Refer to "Using and Editing User-Defined Functions in the Waveform Calculator" on page 319.

Note

wreal waveforms and assertions are not supported in the waveform calculator.

Using the risetime measurement as an example:

```
risetime(wf, topline = "Automatic", baseline = "Automatic",
low = "10%", mid = "50%", up = "90%", x_start = "Begin",
x_end = "End", option = "WF", param = "parameter_name")
```

Note

All of the parameters have a corresponding field text entry, pulldown list, check box, or radio button in the Measurement Tool Dialog Box.

Methods for Entering Measurement Tool Functions 325
Available Measurement Tool Functions 326
option and param Arguments 330
Choosing the Occurrence of the Result 331

Methods for Entering Measurement Tool Functions

You can use several methods for entering Measurement Tool functions in the Waveform Calculator or in a Tcl script file.

Examples

The following are examples:

```
risetime(wf("<tutorial/Time-Domain_Results>v_middle"))
```

If only the waveform name is specified in the function, the application uses all the default parameters.

```
risetime(wf("<tutorial/Time-Domain_Results>v_middle"), baseline=0,
topline=5.0)
```

All the default parameters are used except those that are specified. The parameters can be in any order, and the reference levels can be either percentages or values.

```
risetime(wf("<tutorial/Time-Domain_Results>v_middle"), "Automatic", 0,
10%, 50%, 90%, "Begin", "End", "WF", "parameter_name")
```

A complete list of parameter values are specified. Without the parameter identifier specified, all the parameters have to be in the right sequence.

Available Measurement Tool Functions

You can use a number of Measurement Tool functions to evaluate waveforms in the Waveform Calculator or in a Tcl script file.

Description

Measurement Tool functions:

- **average**

  ```
  average(wf, x_start = "Begin", x_end = "End",
  option = "Value")
  ```

- **bandpass**

  ```
  bandpass(wf, topline = "Automatic", offset = -3,
  x_start = "Begin", x_end = "End", option = "Value")
  ```

- **crossing**

  ```
  crossing(wf, ylevel = "Automatic",
  slopetrigger = "Either",
  x_start = "Begin", x_end = "End", option = "WF",
  param = "parameter_name")
  ```

- **dchysteresis**

  ```
  dchysteresis(wf, ylevel = "Automatic",
  measure = "Width | Left | Right",
  x_start = "Begin", x_end = "End",
  option = "WF | VALUE | ANNOTATION",
  param = "parameter_name")
  ```

- **delay**

  ```
  delay(wf1, wf2, topline1 = "Automatic", baseline1 = "Automatic",
  dlev1 = "50%", topline2 = "Automatic", baseline2 = "Automatic",
  dlev2 = "50%", edgetrigger="Either", inverting = 0, closestedge = 0,
  x_start = "Begin", x_end = "End", option = "WF",
  param = "parameter_name")
  ```

- **duty cycle**

  ```
  dutycycle(wf, topline = "Automatic", baseline = "Automatic",
  edgetrigger="Either", x_start = "Begin", x_end = "End",
  option = "WF", param = "parameter_name")
  ```

- **eye height**

```
eyeheight(wf)
```

- **eye height at X**

```
eyeheightatx(wf, x_value, {"inner"|"outer"})
```

- **eye width**

```
eyewidth(wf)
```

- **eye width at Y**

```
eyewidthaty(wf, y_value, {"inner"|"outer"})
```

- **fall time**

```
falltime(wf, topline = "Automatic", baseline = "Automatic",
low = "10%", mid = "50%", up = "90%", x_start = "Begin",
x_end = "End", option = "WF", param = "parameter_name",
fall = "all")
```

- **frequency**

```
frequency(wf, topline = "Automatic", baseline = "Automatic",
edgetrigger="Either", x_start = "Begin", x_end = "End",
option = "WF", param = "parameter_name")
```

- **gain margin**

```
gainmargin(wf, option = "Value")
```

- **intersection**

```
intersection(wf1, wf2, slopetrigger = "Either", inverting = 0,
x_start = "Begin", x_end = "End",
option = "WF", param = "parameter_name")
```

- **local max**

```
localmax(wf, x_start = "Begin", x_end = "End",
option = "WF", param = "parameter_name")
```

- **local min**

```
localmin(wf, x_start = "Begin", x_end = "End",
option = "WF", param = "parameter_name")
```

- **maximum**

```
max(wf, x_value="no", x_start = "Begin", x_end = "End",
option = "Value")
```

- **mean**

```
mean(wf, x_start = "Begin", x_end = "End", option = "Value")
```

- **mean +3 standard deviation**

```
meanplus3std(wf, x_start = "Begin", x_end = "End", option = "Value")
```

- **mean -3 standard deviation**

```
meanminus3std(wf, x_start = "Begin", x_end = "End",
option = "Value")
```

- **minimum**

```
min(wf, x_value="no", x_start = "Begin", x_end = "End",
option = "Value")
```

- **overshoot**

```
overshoot(wf, topline = "Automatic", baseline = "Automatic",
x_start = "Begin", x_end = "End", option = "WF",
param = "parameter_name", overshoot = "all")
```

- **peak to peak**

```
peaktopeak(wf, x_start = "Begin", x_end ="End", x_value = "no",
option = "Value")
```

- **period**

```
period(wf, topline = "Automatic", baseline = "Automatic",
edgetrigger="Either", x_start = "Begin", x_end = "End",
option = "WF", param = "parameter_name")
```

- **phase margin**

```
phasemargin(wf, option = "Value")
```

- **pulse width**

```
pulsewidth(wf, topline = "Automatic", baseline = "Automatic",
pulsetype="Either", x_start = "Begin", x_end = "End",
option = "WF", param = "parameter_name")
```

- **rise time**

```
risetime(wf, topline = "Automatic", baseline = "Automatic",
low = "10%", mid = "50%", up = "90%", x_start = "Begin",
x_end = "End", option = "WF", param = "parameter_name",
rise = "all")
```

- **rms**

```
rms(wf, x_start = "Begin", x_end = "End", option = "Value")
```

- **rms ac**

```
rms_ac(wf, x_start = "Begin", x_end = "End", option = "Value")
```

- **rms noise**

```
rms_noise(wf, x_start = "Begin", x_end = "End", option = "Value")
```

- **rms tran**

```
rms_tran(wf, x_start = "Begin", x_end = "End", option = "Value")
```

- **settle time**

```
settletime(wf, steadystate = "Automatic", tolerance = "5%",
x_start = "Begin", x_end = "End", option = "Value")
```

- **slew rate**

```
slewrate(wf, topline = "Automatic", baseline = "Automatic",
low = "10%", mid = "50%", up = "90%", edgetrigger="Either",
x_start = "Begin", x_end = "End", option = "WF",
param = "parameter_name", slewrate = "all")
```

- **slope**

```
slope(wf, x, slopetype = "None", option = "Value")
```

- **slope intersect**

```
slopeintersect(wf1, wf2, x1, x2, option = "Value")
```

- **standard deviation**

```
stddev(wf, x_start = "Begin", x_end = "End", option = "Value")
```

- **undershoot**

```
undershoot(wf, topline = "Automatic", baseline = "Automatic",
x_start = "Begin", x_end = "End", option = "WF",
param = "parameter_name", undershoot = "all")
```

- **y value**

```
yval(wf, x1, option = "Value")
```

option and param Arguments

You can set *option* and *param* arguments.

Usage

Some Measurement Tool functions have *option* and *param* arguments that can be set.

Arguments

- option

  (Optional) Specifies the output type. Legal values:

  - "VALUE" — Output is a numerical value or array of numerical values.
  - "WF" — Output is a waveform.
  - "ANNOTATION" — Adds annotation to the input waveform.

  Note

  If the occurrence of the result is anything other than "all" (refer to "Choosing the Occurrence of the Result" on page 331), option="VALUE" is forced.

- param

  (Optional) Used with option="WF". Specifies the simulation parameter to be used to generate the result waveform.

  For compound waveforms the parameters can be seen in the Parameter Table.

Choosing the Occurrence of the Result

You can use *optional* arguments with some measurement tool functions for specifying the occurrence of the result.

Usage

Some Measurement Tool functions have *optional* arguments available for specifying the occurrence of the result.

Arguments

- The argument differs with each function, as shown in the table:

Table 6-3. Arguments for Setting Occurrence

Function	Argument
risetime	rise
falltime	fall
slewrate	slewrate
overshoot	overshoot
undershoot	undershoot

The arguments can all take the following values:

- *occurrence argument*

  (Optional) Specifies the occurrence of the result that the measurement will return.

  For compound waveforms it applies to each element individually.

  Legal values:

  - "first" — Specifies the first occurrence of the result.
  - "all" — Specifies all occurrences of the result. Default.
  - "last" — Specifies the last occurrence of the result.
  - *n* or *"n"* — Specifies the *n*th occurrence of the result.

Built-In Functions

The Waveform Calculator contains a number of built-in functions.

The built-in functions are briefly summarized in the following sections. See "Waveform Calculator Functions" on page 655 for more details on the functions.

In the Waveform Calculator, select the **Functions** tab of the Chooser panel to view the available built-in functions.

Tip

To view detailed information directly in the Waveform Calculator about each function, including the syntax and parameters, choose **View** > **Function Help**.

The following categories are available from the dropdown list in the Waveform Calculator:

Complex Functions 332
Jitter Functions 333
Logic Functions 334
Mathematical Functions 335
Measurement Functions 336
Miscellaneous Functions 338
Phase Noise Functions 341
RF Functions 342
Signal Processing Functions 343
Statistical Functions 345
Trigonometric Functions 346
Special Functions 347

Complex Functions

The Waveform Calculator contains a number of built-in complex functions.

Table 6-4. Built-In Complex Functions

Item	Description
complex	Constructs a complex waveform from two input waveforms
conjugate	Constructs the conjugate of the source complex waveform.
cphase	Returns the phase of the input complex waveform in radians.

Table 6-4. Built-In Complex Functions (cont.)

Item	Description
db	Converts the magnitude data of a waveform to decibels: 20*log10(x)
db10	Converts the magnitude data of a waveform to decibels: 10*log10(x)
gptocomplex	Constructs a complex waveform from a waveform of gain in decibels (the first) and a waveform of phase in radians (the second).
idb	Computes the inverse decibel function for the input waveform: 10(v/20).
idb10	Computes the inverse decibel function for the input waveform: 10(v/10).
imag	Returns the imaginary part of a complex waveform.
mag	Returns the absolute magnitude of a complex waveform.
mptocomplex	Constructs a complex waveform from a waveform of magnitude (the first) and a waveform of phase in radians (the second).
phase	Returns the phase of a complex waveform.
real	Returns the real part of a complex waveform.
ritocomplex	Constructs a complex waveform from a waveform of real part (the first) and a waveform of imaginary part (the second).

Jitter Functions

The Waveform Calculator contains a number of built-in jitter functions.

Table 6-5. Built-In Jitter Functions

Item	Description
cycle2cyclejitter	Calculates the variation in time of the difference between the duration of two adjacent cycles over part of, or the whole waveform.
frequencyjitter	Calculates the variation in time of the frequency of the source waveform relative to its average frequency or to the reference waveform (or ideal clock) frequency over part of, or the whole waveform.

Table 6-5. Built-In Jitter Functions (cont.)

Item	Description
halfperiodjitter	Calculates the variation in time of the half-period of a source waveform relative to its average half-period, or to a reference waveform (or ideal clock) half-period, over part of, or the whole waveform.
plotjitterconfidenceinterval	Plots approximate confidence interval for rms jitter vs. number of Gaussian samples (N).
tiejitter	Calculates the variation in time of the source waveform edges relative to the reference (or ideal clock) waveform edges over part of, or the whole waveform.
timeabsolutejitter	Calculates absolute jitter (time domain).
timelongtermjitter	Calculates the variation in time of the cumulative period of adjacent N-cycle samples. This type of jitter is sometimes called “accumulated” jitter.
timeperiodjitter	Calculates the variation in time of the period of a source waveform relative to its average period or to a reference waveform (or ideal clock) period, over part of, or the whole waveform.

Logic Functions

The Waveform Calculator contains a number of built-in logic functions.

Table 6-6. Built-In Logic Functions

Item	Description
nand	Can be applied to any two digital waveforms whose data types are either bit or boolean.
nor	Can be applied to any two digital waveforms whose data types are either bit or boolean.
rol	Returns a value that is L rotated left by R index positions. That is, if the value of its leftmost argument is referred to as 'L' and the value of its rightmost argument is referred to as 'R', rol() replaces L with a value that is the result of a concatenation whose left argument is the rightmost (L'Length-1) elements of L and whose right argument is L(L'Left).

Table 6-6. Built-In Logic Functions (cont.)

Item	Description
ror	Returns a value that is L rotated right by R index positions. That is, if the value of its leftmost argument is referred to as 'L' and the value of its rightmost argument is referred to as 'R', ror() replaces L with a value that is the result of a concatenation whose right argument is the leftmost (L'Length-1) elements of L and whose left argument is L(L'Right).
sla	Returns a value that is L arithmetically shifted left by R index positions. That is, if the value of its leftmost argument is referred to as 'L' and the value of its rightmost argument is referred to as 'R', sla() replaces L with a value that is the result of a concatenation whose left argument is the rightmost (L'Length-1) elements of L and whose right argument is L(L'Right).
sra	Returns a value that is L arithmetically shifted right by R index positions. That is, if the value of its leftmost argument is referred to as 'L' and the value of its rightmost argument is referred to as 'R', sra() replaces L with a value that is the result of a concatenation whose right argument is the leftmost (L'Length-1) elements of L and whose left argument is L(L'Left).
xnor	Can be applied to any two digital waveforms whose data types are either bit or boolean.

Mathematical Functions

The Waveform Calculator contains a number of built-in mathematical functions.

Table 6-7. Built-In Mathematical Functions

Item	Description
abs	Returns the absolute value of its input argument.
ceil	Computes the smallest integral value not less than each data point of the waveform.
derive	Computes the derivative of the input waveform at the given point.
drv	Computes the derivative of the input waveform.
exp	Computes the value of e raised to the power of the input value.

Table 6-7. Built-In Mathematical Functions (cont.)

Item	Description
floor	Computes the largest integral value not greater than each data point of the waveform.
fmod	Returns the floating-point remainder of the division of x by y.
frexp	Breaks a floating-point number into a normalized fraction and an integral power of 2.
hypot	Computes the length of the hypotenuse of a right-angled triangle: sqrt(x*x+y*y).
integ	Returns the definite integral of a waveform.
integral	Computes the indefinite integral of the input waveform.
ldexp	Computes the quantity x * 2^y.
ln	Computes the natural logarithm of the input argument.
log	Computes the base-10 logarithm of the input argument.
modf	Breaks the argument x into integral and fractional parts, each of which has the same sign as the argument.
pow10	Computes the value of 10 raised to the input argument.
relation	Generates a wave from two input waveforms and a point-by-point relational expression.
round	Computes the closest integer value to each data point of the waveform.
sqr	Computes the square of the input argument.
sqrt	Computes the square root of the input argument.
trunc	Computes the floor for positive data points and ceiling for negative data points of the waveform.
xofmax	Returns all the x value(s) at maximum of a waveform.
xofmin	Returns all the x value(s) at minimum of a waveform.
xwave	Creates a new waveform with y values identical to the x values.

Measurement Functions

The Waveform Calculator contains a number of built-in measurement functions.

Table 6-8. Built-In Measurement Functions

Item	Description
bandpass	Calculates the bandwidth at the level at which the measurement is made, for a bandpass-shaped waveform.
bandwidth	Returns the bandwidth of the input waveform based on the calculation of its topline.
baseline	Returns the baseline of the input waveform based on the calculation of histograms.
crossing	Measures the crossing of a waveform relative to default (automatically calculated) or user-specified Y level.
delay	Measures the delay between edges of the source waveform and a reference waveform.
dutycycle	Measures the duty cycle of a periodic waveform relative to default (automatically calculated) or user-specified topline and baseline levels.
falltime	Measures the difference in time when the waveform falls from the upper level to the lower level.
frequency	Measures the frequency of a periodic waveform relative to default (automatically calculated) or user-specified topline and baseline levels.
gainmargin	Calculates the gain margin in decibels (dB) of a complex waveform. The gain margin is defined as the difference between the gain of the measured waveform and 0 dB (unity gain) at the frequency where the phase shift is -180 degrees (Phase Crossover Frequency).
localmax	Finds the local maxima of a waveform.
localmin	Finds the local minima of a waveform.
mean	Calculates the mean value of a waveform.
meanminus3std	Calculates the mean minus 3 standard deviation value of a waveform.
meanplus3std	Calculates the mean plus 3 standard deviation value of a waveform.
overshoot	Calculates the overshoot value of a waveform. The overshoot value is calculated as the difference between the maximum point and the topline level of the waveform.
peaktopeak	Calculates the peak-to-peak value of a waveform.
period	Measures the period of a periodic waveform relative to default (automatically calculated) or user-specified topline and baseline levels.

Table 6-8. Built-In Measurement Functions (cont.)

Item	Description
phasemargin	Calculates the phase margin of a complex waveform (in degrees or radians). The phase margin is defined as the difference in phase between the measured waveform and -180 degrees at the point corresponding to the frequency that gives a gain of 0 dB (the Gain Crossover Frequency).
pulsewidth	Measures the pulse width of a periodic waveform relative to default (automatically calculated) or user-specified topline and baseline levels.
risetime	Measures the difference in time when the waveform rises from the lower level to the upper level.
settlingtime	Calculates the settle time of a waveform with respect to default or specified steady state level and a specified tolerance.
slewrate	Measures the slew rate of a periodic waveform relative to default (automatically calculated) or user-specified topline and baseline levels.
slope	Returns the slope value of a waveform at a specified x value.
slopeintersect	Finds the slope intersection of two waveforms at specified x values.
stddev	Calculates the standard deviation of the specified waveform.
topline	Returns the topline of the input waveform based on the calculation of histograms.
undershoot	Calculates the undershoot value of a waveform. The undershoot value is calculated as the difference between the minimum point and the baseline level of the waveform.

Miscellaneous Functions

The Waveform Calculator contains a number of built-in miscellaneous functions.

Table 6-9. Built-In Miscellaneous Functions

Item	Description
add	Adds two overlapping waveforms, retaining non-overlapping parts.
analysisattributes	Returns analysis attribute names and values from the PSF header of the specified waveform.
analysisattributevalue	Returns the value of the specified analysis attribute from the PSF header of the specified waveform.

Table 6-9. Built-In Miscellaneous Functions (cont.)

Item	Description
assign	Uses “best efforts” to assign a specified name to an expression, without overwriting existing expressions.
atod	Transforms an analog waveform to a digital waveform.
calcvartype	Returns the type of a Waveform Calculator variable.
concat	Computes the concatenation of two input waveforms.
continuous	Converts any analog waveform to a continuous analog waveform.
datatocomplex	Creates a complex waveform based on one or two arrays of data points.
datatodig	Creates a digital waveform based on one or two arrays of data events.
datatowf	Creates an analog waveform based on one or two arrays of data points.
dchysteresis	Measures the thresholds and width of a DC Hysteresis waveform relative to a default, or specified, Y level.
dtoa	Constructs an analog waveform from a digital bus.
dtoaonbit	Constructs an analog waveform from a bit.
exportcsv	Exports the specified waveforms to a *.csv* file (without header), on the x-axis slice on which all waveforms are defined.
exportvcd	Exports the specified (time domain) waveforms to a 4-state *.vcd* file. This is useful, for example, to generate a *.vcd* file from an analog simulation to use as stimuli in a digital simulation.
filterdupreal	Removes glitches on real (step or wreal) waveforms.
filterempty	Removes all empty waveforms from a compound waveform.
first	Returns the first point of a waveform or a list.
firstdiff	Finds the first time when any signal in one run begins to differ from the corresponding signal in another run, and plots *ntop* signals in sorted order.
gendecade	Returns a list that contains numbers sweeping from a start value to a stop value with a specified number of points per decade.
genlinear	Returns a list that contains numbers sweeping from a start value to a stop value with a specified step.

Table 6-9. Built-In Miscellaneous Functions (cont.)

Item	Description
genoctave	Returns a list that contains numbers sweeping from a start value to a stop value with a specified number of points per octave.
getelementat	Returns the element at a given index in a bus or compound waveform.
getrunindices	Returns a list of run indices for the specified compound waveform.
getrunparameters	Returns a list of run parameters for the specified compound waveform.
getrunparametervalue	Returns the run parameter value for the specified compound bit waveform.
gmargin	Computes the gain margin of a complex waveform or a magnitude waveform and a phase waveform.
groupdelay	Computes the Group Delay (GD), in seconds, from an input signal from an AC or SP analysis. GD is defined as the derivative of the phase with respect to the frequency.
harm	Returns the voltage on a net at a specified harmonic.
intersect	Returns an array with all the intersection points of the two input waveforms.
join	Joins two non-overlapping waveforms together into a single waveform.
last	Returns the last point of a waveform or a list.
length	Returns the number of elements of a given bus or compound waveform.
maxdiff	Finds the time or the frequency when any signal in one run differs the most from the corresponding signal in another run, and plots *ntop* signals in sorted order.
maxspectrumdiff	Finds the frequency when any signal in one run differs the most from the corresponding signal in another run, after *phase* shift, and plots *ntop* signals in sorted order.
phmargin	Computes the phase margin of a complex waveform or a magnitude waveform and a phase waveform.
pivot	Generates a pivot waveform using the input compound waveform as the y axis and a second waveform, or sweep parameter, as the x axis.
reglin	Performs the linear regression between a par_value and a char_value.

Table 6-9. Built-In Miscellaneous Functions (cont.)

Item	Description
removepts	Removes points from a waveform where the states of a second (digital) waveform are “0”.
shift	Creates a shifted waveform.
shiftedmaxdiff	Finds the time when any signal in one run, shifted by *tdelay*, differs the most from the corresponding signal in another run, and plots *ntop* signals in sorted order.
timestep	Returns a waveform of timesteps vs. time of a waveform variable.
todchysteresis	Creates a DC Hysteresis waveform from a two step DC simulation.
var	Enables database variable values to be used in Waveform Calculator expressions.
wfattributes	Returns a waveform built from the input waveform with modified units, scales or display type.
wfname	Returns the waveform name, or a specified part of the name, as a string.
wftoascii	Dumps the input waveform to a text (ASCII) file.
wftodata	Returns an array with the data points of the input waveform
window	Creates a new waveform between specified lower and upper bounds out of the input wavefrm, and interpolates the points at the interval bounds.
xdown	Returns all the x value(s) where the waveform crosses a given y value with negative slope.
xup	Returns all the x value(s) where the waveform crosses a given y value with positive slope.
xval	Returns all x values for a given y value.
xytowf	Creates a new waveform using the Y values from wfX as the x axis and the Y values from wfY as y axis.
yval	Returns the y values for a given x value.

Phase Noise Functions

The Waveform Calculator contains a number of built-in Phase Noise functions.

Table 6-10. Built-In Phase Noise Functions

Item	Description
absolutejitter	Computes the value representing the absolute jitter of the waveform, applied to SST Noise Analysis results.
absolutejitterbyintegration	Calculates the rms absolute jitter by Sphi integration.
allandeviation	Returns the Allan Deviation vs. Time waveform from SST Noise Analysis results.
cycletocyclejitter	Returns the cycle-to- cycle jitter, applied to SST Noise Analysis results.
cycletocyclejitterbyintegration	Calculates rms cycle-to-cycle or adjacent k-period jitter by Sphi integration.
longtermjitter	Computes the waveform representing the long term jitter.
periodjitter	Computes the value representing the period jitter of the waveform.
periodjitterbyintegration	Calculates the rms k-period jitter by Sphi integration.
phasenoiseoutdbc	Returns a phase noise dBc/Hz waveform from a V/sqrtHz output and determines dBc scalar values at two selected frequency points.
phnoisebydlm	Computes the phase noise in dBc/Hz by delay line measurement, where the oscillation waveform is delayed and mixed with itself.
phnoisebymixer	Computes the phase noise in dBc/Hz by mixer measurement, where the oscillation waveform is mixed with a pure tone.
sphibyjitter	Computes two-sided Sphi density in dB rad^2/Hz by jitter measurement. The positive side of Sphi is returned.
sphifilter	Applies a high pass filter to frequency-domain Sphi density.

RF Functions

The Waveform Calculator contains a number of built-in RF functions.

Table 6-11. Built-In RF Functions

Item	Description
compress	Extracts the y-axis value of the wave at the point where the difference between the actual value of the wave and the linear extrapolation of the wave based on the computed slope value becomes greater than the supplied value.

Table 6-11. Built-In RF Functions (cont.)

Item	Description
compresscompound	Computes the N^{th} compression point for the specified harmonic.
constellationdiagram	Computes the constellation diagram of the input waveform.
edgephasenoise	Computes an edge phasenoise waveform from pnoise (or hb) sampled(jitter) analysis.
evmber	Computes the Error Vector Magnitude and Bit Error Rate of the two input constellation diagrams.
harmonicsmeter	Computes the maximum Fourier frequency value for a periodic waveform.
iipx	Computes the input inferred intercept point of order x from the values of the circuit input and output.
integnoise	Computes the integrated noise from *f_start* to *f_stop* from a pnoise sampled(jitter) "out" (V/sqrt(Hz)) noise waveform.
ipn	Computes the N^{th} order input-referred intercept point, output-referred intercept point, or both.
jc	Computes cycle jitter.
jcc	Computes cycle-to-cycle jitter.
jee	Computes edge-to-edge jitter.
noisetosignaldbc	Calculates the ratio of noise to signal amplitude, in dBc/Hz.
noisetrantophasenoise	Computes the Phase Noise Spectrum (Power Spectral Density) of a (noisy) periodic transient waveform.
oipx	Computes the output inferred intercept point of order x from the value of the circuit output wave.
phasenoise	Computes the phase noise of a transient analysis.
pssresidue	Computes the difference of every signal at *tstop* and *tstop-tperiod*. Returns the maximum pss residue.
topnoise	Computes noise spectrum of spot noise at *fstart* or integrated noise from *fstart* to *fstop*.
xcompress	Extracts the x-axis value of the wave at the point where the actual value of the wave and the extrapolated linear value of the wave exceeds a certain value.

Signal Processing Functions

The Waveform Calculator contains a number of built-in signal processing functions.

Table 6-12. Built-In Signal Processing Functions

Item	Description
autocor	Computes the Auto Correlation of the input waveform.
chirp	Computes the Chirp Transformation of the input waveform.
convolution	Computes the Convolution of the two input waveforms.
crosscorrelation	Computes the Cross Correlation of the two input waveforms.
deg	Converts the waveform into degrees.
enob	Computes the effective number of bits (ENOB) for the input waveform (by a conversion of the sinad result).
fft	Computes the Discrete Fourier Transform of the input waveform using the Fast Fourier Transform method.
harmonicdistortion	Computes the Harmonic Distortion of the input waveform.
harmonics	Computes the Harmonic Distortion of the input waveform.
hdist	Computes the total harmonic distortion of the input waveform.
ifft	Computes the inverse Fast Fourier Transform of the input waveform.
psd	Computes the Power Spectral Density of the input waveform.
rad	Converts the waveform into radians.
sample	Creates a sampled waveform with equidistant points with respect to the x-axis variable.
samplelog	Creates a sampled waveform with data points logarithmically spaced.
samplepsd	Samples a continuous waveform, and returns its power spectral density
sfdr	Computes the spurious free dynamic range of the input waveform.
sinad	Computes the signal to noise and distortion ratio (SINAD) of the input waveform.
sndr	Computes the signal to noise and distortion ratio (SINAD) of the input waveform.
snr	Computes the Signal to Noise Ratio of the input waveform.
spectrummeasurement	Computes spectrum measurements on the input waveform; “snr”, “sndr”, “sfdr”, “enob”, or “thd”.

Table 6-12. Built-In Signal Processing Functions (cont.)

Item	Description
thd	Computes the total harmonic distortion of the input waveform.
windowing	Calculates the Windowing of the input waveform.

Statistical Functions

The Waveform Calculator contains a number of built-in statistical functions.

Table 6-13. Built-In Statistical Functions

Item	Description
avg	Returns the average of the waveform.
cphytrigger	Creates first zero-crossing C-Phi trigger waveform. Also used to compute UI jitter and transition jitter.
eye	Constructs an eye diagram from the input waveform.
eyeamplitude	Returns the amplitude of an eye diagram.
eyecphy	Constructs a self-triggered C-Phy eye diagram from three input waveforms.
eyecrossingamplitude	Returns the amplitude level at which the eye crossings occur on an eye diagram.
eyedelay	Returns the distance from the midpoint of the eye to the time origin.
eyediagram	Constructs an eye diagram of a waveform.
eyefalltime	Returns the falltime of an eye diagram.
eyeheight	Returns the height of an eye diagram.
eyeheightatx	Returns the eye height at X.
eyejitter	Returns the jitter of an eye diagram.
eyemeasures	Returns a list of pairs of named measures with their values.
eyerisetime	Returns the risetime of an eye diagram.
eyesetmask	Adds a mask to an eye diagram.
eyesnr	Returns the SNR of an eye diagram.
eyewidth	Returns the width of an eye diagram.
eyewidthaty	Returns the eye width at Y.
eyewithtrigger	Constructs a triggered eye diagram from a waveform.

Table 6-13. Built-In Statistical Functions (cont.)

Item	Description
histogram	Creates a histogram of the input waveform showing the magnitude probability density distribution of the waveform.
larger	Creates a new waveform based on the larger of two data points at any given time of the input waveforms.
lesser	Creates a new waveform based on the lesser of two data points at any given time of the input waveforms.
max	Returns the maximum value of a waveform.
min	Returns the minimum value of a waveform.
rms	Returns the root mean square value of a waveform, depending on its x-axis scale. If the x-axis scale is in Hz, it uses the rms_ac function, otherwise it uses rms_tran, to calculate the root mean square.
rms_ac	Returns the root mean square value of a waveform for frequency analysis using: sqrt(integ(v*v))
rms_accurate	Returns the integrated root mean square value of a continuous waveform that is accurate even if the waveform is noisy.
rms_noise	Returns the root mean square value of a waveform for frequency noise analysis: sqrt(sum(v*v))
rms_tran	Returns the root mean square value of a waveform for time analysis: sqrt(integ(v*v)/time_interval)
size	Returns the number of points (or transitions) in the waveform.
sum	Returns a sum of all of the Y-values of the input waveform.
windavg	Returns an average value for each X-value of the input waveform.

Trigonometric Functions

The Waveform Calculator contains a number of built-in trigonometric functions.

Table 6-14. Built-In Trigonometric Functions

Item	Description
acos	Computes the principal value of the arc cosine of the argument.
acosh	Computes the inverse hyperbolic cosine of the argument.

Table 6-14. Built-In Trigonometric Functions (cont.)

Item	Description
acot	Computes the arc cotangent of the input waveform.
acoth	Computes the hyperbolic arc cotangent of the input waveform.
asin	Computes the principal value of the arc sine of the argument.
asinh	Computes the inverse hyperbolic sine of the argument.
atan	Computes the principal value of the arc tangent of the argument.
atan2	Computes the principal value of the arc tangent of y/x, using the signs of both arguments to determine the quadrant of the return value.
atanh	Computes the inverse hyperbolic tangent of the argument.
cos	Computes the cosine of the argument, in degrees.
cosh	Computes the hyperbolic cosine of the argument.
cot	Computes and returns the cotangent of the argument, in degrees.
coth	Computes and returns the hyperbolic cotangent of the waveform.
sin	Computes the sine of the argument, in degrees.
sinh	Computes the hyperbolic sine of the argument.
tan	Computes the tangent of the argument, in degrees.
tanh	Computes the hyperbolic tangent of the argument.

Special Functions

The Waveform Calculator contains a number of special functions.

Note

Waveform Calculator Special functions do not appear in the Waveform Calculator Functions list but may be used for scripting.

Table 6-15. Built-In Statistical Functions

Item	Description
setAngleUnits	Specifies the trigonometrical angle unit.

Table 6-15. Built-In Statistical Functions (cont.)

Item	Description
setNotation	Specifies the expression evaluation logic (IEEE or SPICE).
setTemperatureUnits	Specifies the temperature unit.
wf	References a waveform.

Calculator Buttons

You can use the dropdown list in the Waveform Calculator toolbar to select different types of calculator buttons to display.

Listed in the following tables are the categories available and their associated buttons. These categories are available from the dropdown list in the Waveform Calculator:

Complex Buttons 349
Jitter Buttons 351
Logical Buttons 352
Phase Noise Buttons 357
RF Buttons 359
Signal Processing Buttons 361
Statistical Buttons 363
Trigonometric Buttons 365

Complex Buttons

The Complex dropdown list item changes the Waveform Calculator buttons to functions and operators for complex number operations/calculations.

Table 6-16. Complex Function Buttons

Icon	Item	Description
1/X	1 / x	Returns the reciprocal value of a scalar or a waveform.
10^x	pow(10,x)	Returns 10 to the x*th* power.
abs	abs()	Absolute value of a scalar is the scalar without its sign. The absolute values of a waveform is calculated as the absolute value of an argument. abs(x) = x abs(-x) = x
complex	complex(,)	Constructs a complex waveform from a pair of input waveforms. The pairs include: real part and imaginary part waveforms, magnitude and phase waveforms, or gain and phase waveforms.
cphase	cphase()	Extracts the continuous phase from a complex waveform. The returned waveform is the continuous phase from the input complex waveform.

Table 6-16. Complex Function Buttons (cont.)

dB	db()	Converts the waveform in dB. A transformation setting for complex waveforms that shows the magnitude of each point of the complex waveform calculated in decibels (20 * log (|waveform|))
dB10	db10()	Converts the waveform in dB. A transformation setting for complex waveforms that shows the magnitude of each point of the complex waveform calculated in decibels (10 * log (|waveform|)
$\frac{dy}{dx}$	drv()	Returns the derivative of a waveform.
Eval	Evaluate	Evaluates the expression specified in the textbox.
e^x	exp()	Exponential function is defined by exp(x) = ex, where e is the constant 2.71828…
j		Imaginary unit
imag	imag()	Returns the imaginary values of a complex waveform.
ln	ln()	Returns the natural logarithm of a waveform.
log	log()	Returns the base 10 logarithm of a waveform.
mag	mag()	Returns the magnitude of a waveform. The transformation applied to a complex waveform that shows the square root of ($real^2$ + $imag^2$) for each point in the complex waveform.
mod	x % y	Returns the modulus of a waveform. The mod of a waveform is calculated as “x % y” is the remainder of the division x/y for integers x and y.
Plot	Plot	Plots the last result waveform to the active graph window. The button is active only if the expression result is a waveform.
phase	phase()	Returns the phase values of a waveform. The phase of each point in the complex waveform in Polar form. All phase angles are restricted between -180 and +180 degrees (-pi radians and +pi radians).
real	real()	Returns the real values of a waveform.
Recall	Recall	Enables you to restore the results of a set of expressions saved to a Tcl file with the Store command.
Store	Store	Enables you to save a set of selected expressions in a Tcl file that can be recalled later.

Table 6-16. Complex Function Buttons (cont.)

x^2	sqr()	Returns the squared scalar or waveform. The square of the waveform is calculated as $x^2 = x * x$.
x^y	x ** y	Returns “x to the y^{th} power” or x^y, where x and y can be either a waveform or a scalar.
$\sqrt{x}$	sqrt()	Returns the square root of a scalar or a waveform.
$\int$	integral()	Returns the anti-derivative of a waveform.

Jitter Buttons

The Jitter dropdown list item changes the Waveform Calculator buttons to functions and operators for jitter calculations.

Table 6-17. Jitter Buttons

Icon	Item	Description
1/X	1 / x	Returns the reciprocal value of a scalar or a waveform.
10^x	pow(10,x)	Returns “10 to the x^{th} power”.
abs	abs()	Absolute value of a scalar is the scalar without its sign. The absolute values of a waveform is calculated as the absolute value of an argument abs(x) = x abs(-x) = x
cycle2c...	Cycle-to-Cycle Jitter (Time-Domain)	Opens the Jitter Tool Dialog Box with the Jitter option set to Cycle-to-cycle Jitter.
$\frac{dy}{dx}$	drv()	Returns the derivative of a waveform.
Eval	Evaluate	Evaluates the expression specified in the textbox.
e^x	exp()	Exponential function is defined by exp(x) = ex, where e is the constant 2.71828…
freque...	Frequency Jitter (Time Domain)	Opens the Jitter Tool Dialog Box with the Jitter option set to Frequency Jitter.
halfper...	Half-Period Jitter (Time Domain)	Opens the Jitter Tool Dialog Box with the Jitter option set to Half-period Jitter.
ln	ln()	Returns the natural logarithm of a waveform.

Table 6-17. Jitter Buttons (cont.)

log	log()	Returns the base 10 logarithm of a waveform.
mod	x % y	Returns the modulus of a waveform. The mod of a waveform is calculated as “x % y” is the remainder of the division x/y for integers x and y.
Plot	Plot	Plots the last result waveform to the active graph window. The button is active only if the expression result is a waveform.
plotjitt...	Plot Jitter Confidence Interval	Opens the Jitter Tool Dialog Box with the Jitter option set to Jitter Confidence Interval.
Recall	Recall	Enables you to restore the results of a set of expressions saved to a Tcl file with the Store command.
Store	Store	Enables you to save a set of selected expressions in a Tcl file that can be recalled later.
tiejitter	Time Interval Error (TIE)	Opens the Jitter Tool Dialog Box with the Jitter option set to Time Interval Error.
timeabs...	Time Absolute Jitter	Opens the Jitter Tool Dialog Box with the Jitter option set to Absolute Jitter.
timelo...	Long-Term Jitter (Time-Domain)	Opens the Jitter Tool Dialog Box with tthe Jitter option set to Long-term Jitter.
timepe...	Period Jitter (Time-Domain)	Opens the Jitter Tool Dialog Box with the Jitter option set to Period Jitter.
x^2	sqr()	Returns the squared scalar or waveform. The square of the waveform is calculated as $x^2 = x * x$
x^y	x ** y	Returns “x to the y*th* power” or x^y, where x and y can be either a waveform or a scalar.
$\sqrt{x}$	sqrt()	Returns the square root of a scalar or a waveform.
$\int$	integral()	Returns the anti-derivative of a waveform.

Logical Buttons

The Logic dropdown list item changes the Waveform Calculator buttons to functions and operators for logic number operations/calculations.

Table 6-18. Logical Buttons

Icon	Item	Description

Table 6-18. Logical Buttons (cont.)

0x	0x	Hexadecimal based notation.
<	<	Lesser than operator can be applied to any two digital waveforms, or one waveform and one scalar value. If the input is two digital waveforms whose data types are either bit or boolean, then the result of the operator is defined in the following table: The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit. A B A<B T T F T F F F T T F F F
>	>	Greater than operator can be applied to any two digital waveforms, or one waveform and one scalar value. If the input is two digital waveforms whose data types are either bit or boolean, then the result of the operator is defined in the following table: The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit. A B A>B T T F T F T F T F F F F

Table 6-18. Logical Buttons (cont.)

<=	<=	Less than or equal to operator can be applied to any two digital waveforms, or one waveform and one scalar value. If the input is two digital waveforms whose data types are either bit or boolean, then the result of the operator is defined in the following table: The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit. A B A<=B T T T T F F F T T F F T
>=	>=	Greater than or equal to operator can be applied to any two digital waveforms, or one waveform and one scalar value. If the input is two digital waveforms whose data types are either bit or boolean, then the result of the operator is defined in the following table: The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit. A B A >= B T T T T F T F T F F F T
!=	!=	Not equal operator can be applied to any two digital waveforms, or one waveform and one scalar value. If the input is two digital waveforms whose data types are either bit or boolean, then the result of the operator is defined in the following table: The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit. A B A!=B T T F T F T F T T F F F

Table 6-18. Logical Buttons (cont.)

=	=	Equal operator can be applied to any two digital waveforms, or one waveform and one scalar value. If the input is two digital waveforms whose data types are either bit or boolean, then the result of the operator is defined in the following table: The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit. A B A==B T T T T F F F T F F F T
' '	''	Single bit notation.
" "	" "	VHDL bit string notation.
and	'&'	And operator can be applied to any two digital waveforms whose data types are either bit or boolean. The result of the operator is defined in the following table: The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit. A B A&B T T T T F F F T F F F F
Eval	Evaluate	Evaluates the expression specified in the textbox.
H	H	Specifies logic state H, weak 1.
L	L	Specifies logic state L, weak 0.

Table 6-18. Logical Buttons (cont.)

<table>
<tr><td>not</td><td>~</td><td>Not operator can be applied to any digital waveform whose data type is either bit or boolean.

The result of the operator is defined in the following table:

The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit.

A ~A

T F

F T</td></tr>
<tr><td>or</td><td>|</td><td>Or operator can be applied to any two digital waveforms whose data types are either bit or boolean.

The result of the operator is defined in the following table:

The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit.

A B A|B

T T T

T F T

F T T

F F F</td></tr>
<tr><td>Plot</td><td>Plot</td><td>Plots the last result waveform to the active graph window. The button is active only if the expression result is a waveform.</td></tr>
<tr><td>Recall</td><td>Recall</td><td>Enables you to restore the results of a set of expressions saved to a Tcl file with the Store command.</td></tr>
<tr><td>sll</td><td><<</td><td>Shift left logical returns a value that is L logically shifted left by R index positions. If the value of its leftmost argument is referred to as 'L' and the value of its rightmost argument is referred to as 'R', sll() replaces L with a value that is the result of a concatenation whose left argument is the rightmost (L'Length-1) elements of L and whose right argument is T'Left, where T is the element type of L.

1.) If R is '0' or if L is a null array, the return value is L;

2.) If R is positive, sll() is repeated R times to form the result.</td></tr>
</table>

Table 6-18. Logical Buttons (cont.)

srl	>>	Shift right logical returns a value that is L logically shifted right by R index positions. If the value of its leftmost argument is referred to as 'L' and the value of its rightmost argument is referred to as 'R', srl() replaces L with a value that is the result of a concatenation whose right argument is the leftmost (L'Length-1) elements of L and whose left argument is T'Left, where T is the element type of L. 1.) If R is '0' or if L is a null array, the return value is L; 2.) If R is positive, srl() is repeated R times to form the result.
Store	Store	Enables you to save a set of selected expressions in a Tcl file that can be recalled later.
U	U	Specifies logic state U, uninitialized.
W	W	Specifies logic state W, weak unknown.
X	X	Specifies logic state X, forcing unknown.
xor	^	The xor operator can be applied to any two digital waveforms whose data types are either bit or boolean. The result of the operator is defined in the following table: The symbol 'T' represents TRUE for type boolean and '1' for type bit; the symbol 'F' represents FALSE for type boolean and '0' for type bit. A B A^B T T F T F T F T T F F F
Z	Z	Specifies logic state Z, high impedance.

Phase Noise Buttons

The Phase Noise dropdown list item changes the Waveform Calculator buttons to functions and operators for phase noise operations/calculations.

Table 6-19. Phase Noise Buttons

Icon	Item	Description
1/X	1 / x	Returns the reciprocal value of a scalar or a waveform.
10^x	pow(10,x)	Returns "10 to the x^{th} power".
abs	abs()	Absolute value of a scalar is the scalar without its sign. The absolute values of a waveform is calculated as the absolute value of an argument abs(x) = x abs(-x) = x
absolut...	Absolute Jitter	Opens the Jitter Tool Dialog Box with the Phase Noise Jitter option set to Absolute Jitter.
cycleto...	Cycle-to-cycle Jitter	Opens the Jitter Tool Dialog Box with the Phase Noise Jitter option set to Cycle-to-cycle Jitter.
$\frac{dy}{dx}$	drv()	Returns the derivative of a waveform.
Eval	Evaluate	Evaluates the expression specified in the textbox.
e^x	exp()	Exponential function is defined by exp(x) = ex, where e is the constant 2.71828…
ln	ln()	Returns the natural logarithm of a waveform.
log	log()	Returns the base 10 logarithm of a waveform.
longter...	Long-term Jitter	Opens the Jitter Tool Dialog Box with the Phase Noise Jitter option set to Long-term Jitter.
mod	x % y	Returns the modulus of a waveform. The mod of a waveform is calculated as "x % y" is the remainder of the division x/y for integers x and y.
periodji...	Period Jitter	Opens the Jitter Tool Dialog Box with the Phase Noise Jitter option set to Period Jitter.
Plot	Plot	Plots the last result waveform to the active graph window. The button is active only if the expression result is a waveform.
Recall	Recall	Enables you to restore the results of a set of expressions saved to a Tcl file with the Store command.
Store	Store	Enables you to save a set of selected expressions in a Tcl file that can be recalled later.
x^2	sqr()	Returns the squared scalar or waveform. The square of the waveform is calculated as $x^2 = x * x$

Table 6-19. Phase Noise Buttons (cont.)

Icon	Item	Description
x^y	x ** y	Returns “x to the y*th* power” or x^y, where x and y can be either a waveform or a scalar.
$\sqrt{x}$	sqrt()	Returns the square root of a scalar or a waveform.
$\int$	integral()	Returns the anti-derivative of a waveform.

RF Buttons

The RF dropdown list item changes the Waveform Calculator buttons to functions and operators for RF operations/calculations.

Table 6-20. RF Buttons

Icon	Item	Description
1/X	1 / x	Returns the reciprocal value of a scalar or a waveform.
10^x	pow(10,x)	Returns “10 to the x*th* power”.
abs	abs()	Absolute value of a scalar is the scalar without its sign. The absolute values of a waveform is calculated as the absolute value of an argument abs(x) = x abs(-x) = x
cd	constellationdiagram()	Creates a Constellation Diagram of a waveform. Opens the Constellation Diagram Dialog Box to prompt for the waveform, the delay, and the symbol period.
compress	compress()	Extracts the y-axis value of the wave at the point where the difference between the actual value of the wave and the linear extrapolation of the wave based on the computed slope value becomes greater than the supplied value.
dnaadvisor	harmonicsmeter()	Opens the DNA Advisor Dialog Box. Plots signals in sorted order of the maximum Fourier frequency. Returns the maximum Fourier frequency value.
$\frac{dy}{dx}$	drv()	Returns the derivative of a waveform.
Eval	Evaluate	Evaluates the expression specified in the textbox.

Table 6-20. RF Buttons (cont.)

evmber	evmber()	Opens the Error Vector Magnitude and Bit Error Rate Dialog Box. Calculates error vector magnitude and bit error ratio from a set of constellation diagrams.
e^x	exp()	Exponential function is defined by exp(x) = ex, where e is the constant 2.71828…
iipx	iipx()	Returns the input referred intercept point of order X from the value of the circuit input and output.
ln	ln()	Returns the natural logarithm of a waveform.
log	log()	Returns the base 10 logarithm of a waveform.
mod	x % y	Returns the modulus of a waveform. The mod of a waveform is calculated as “x % y” is the remainder of the division x/y for integers x and y.
oipx	oipx()	Returns the output referred intercept point of order X from the value of the circuit output wave.
Plot	Plot	Plots the last result waveform to the active graph window. The button is active only if the expression result is a waveform.
pn	phasenoise()	Opens the Phase Noise Dialog Box. Calculates the phase noise of a transient analysis.
pssresidue	pssresidue()	Opens the PSS Residue Dialog Box. Calculates the difference of every signal at *tstop* and *tstop-tperiod* and plots *ntop* signals in sorted order of the maximum difference. Returns the maximum pss residue.
Recall	Recall	Enables you to restore the results of a set of expressions saved to a Tcl file with the Store command.
Store	Store	Enables you to save a set of selected expressions in a Tcl file that can be recalled later.
topnoise	topnoise()	Opens the Top Noise Dialog Box. Plots *ntop* noise spectrum in sorted order of spot noise at *fstart* or integrated noise from *fstart* to *fstop*.
xcompress	xcompress()	Extracts the x-axis value of the wave at the point where the difference between the actual value of the wave and the linear extrapolation of the wave based on the computed slope value becomes greater than the supplied value.
x^2	sqr()	Returns the squared scalar or waveform. The square of the waveform is calculated as $x^2 = x * x$

Table 6-20. RF Buttons (cont.)

x^y	x ** y	Returns "x to the y^{th} power" or x^y, where x and y can be either a waveform or a scalar.
$\sqrt{x}$	sqrt()	Returns the square root of a scalar or a waveform.
$\int$	integral()	Returns the anti-derivative of a waveform.

Signal Processing Buttons

The Signal Processing dropdown list item changes the Waveform Calculator buttons to functions and operators for transform operations/calculations.

Table 6-21. Signal Processing Buttons

Icon	Item	Description
1/X	1 / x	Returns the reciprocal value of a scalar or a waveform.
10^x	pow(10,x)	Returns "10 to the x^{th} power".
abs	abs()	Absolute value of a scalar is the scalar without its sign. The absolute values of a waveform is calculated as the absolute value of an argument: abs(x) = x abs(-x) = x
autocor	autocor()	Calculates the Autocorrelation Function of a signal waveform. Opens the Auto Correlation Dialog Box.
chirp	chirp()	Calculates the chirp Z-transform. Opens the Chirp Transform Dialog Box.
conv	convolution()	Calculates the linear convolutions of two finite data sequences.Opens the Convolution Dialog Box.
$\frac{dy}{dx}$	drv()	Returns the derivative of a waveform.
Eval	Evaluate	Evaluates the expression specified in the textbox.
e^x	exp()	Exponential function is defined by exp(x) = e^x, where e is the constant 2.71828…

Table 6-21. Signal Processing Buttons (cont.)

fft	fft	Determines the frequency content of analog signals encountered in circuit simulation, which deals with sequences of time values. Opens the Fast Fourier Transform Tool Dialog Box. See Statistical Buttons for details.
hd	harmonicdistortion()	Calculates the harmonic distortion of a signal. Opens the Harmonic Distortion Dialog Box.
ifft	ifft	Calculates the inverse fast Fourier transform of the input waveform. Opens the Inverse Fast Fourier Transform Dialog Box. See Using the Inverse Fast Fourier Transform Tool for details.
ln	ln()	Returns the natural logarithm of a waveform.
log	log()	Returns the base 10 logarithm of a waveform.
mod	x % y	Returns the modulus of a waveform. The mod of a waveform is calculated as “x % y” is the remainder of the division x/y for integers x and y.
Plot	Plot	Plots the last result waveform to the active graph window. The button is active only if the expression result is a waveform.
psd	psd()	Calculates the Power Spectral Density of a signal waveform. Opens the Power Spectral Density Dialog Box.
Recall	Recall	Enables you to restore the results of a set of expressions saved to a Tcl file with the Store command.
snr	snr() sndr() enob() sfdr	Opens the Signal to Noise Ratio Dialog Box. Enables you to specify parameters for the four functions listed.

Table 6-21. Signal Processing Buttons (cont.)

spec. meas.	spectrummeasurement()	Computes a specific measurement on the fft of a waveform. Opens the Spectrum Measurement Tool Dialog Box. See Using the Spectrum Measurement Tool for details.
Store	Store	Enables you to save a set of selected expressions in a Tcl file that can be recalled later.
wnd	windowing()	Tapers data near ends of records to avoid abrupt truncation effects. Opens the Windowing Transform Dialog Box.
x^2	sqr()	Returns the squared scalar or waveform. The square of the waveform is calculated as $x^2 = x * x$
x^y	x ** y	Returns “x to the y^{th} power” or x^y, where x and y can be either a waveform or a scalar.
$\sqrt{x}$	sqrt()	Returns the square root of a scalar or a waveform.
xcorr	crosscorrelation()	Calculates the cross correlation between two data sets. Opens the Cross Correlation Dialog Box.
$\int$	integral()	Returns the anti-derivative of a waveform.

Statistical Buttons

The Statistical dropdown list item changes the Waveform Calculator buttons to functions and operators for statistical number operations/calculations.

Table 6-22. Statistical Buttons

Icon	Item	Description
1/X	1 / x	Returns the reciprocal value of a scalar or a waveform.
10^x	pow(10,x)	Returns “10 to the x^{th} power”.

Table 6-22. Statistical Buttons (cont.)

abs	abs()	Absolute value of a scalar is the scalar without its sign. The absolute values of a waveform is calculated as the absolute value of an argument abs(x) = x abs(-x) = x
avg	avg()	Returns the average of the waveform.
$\frac{dy}{dx}$	drv()	Returns the derivative of a waveform.
Eval	Evaluate	Evaluates the expression specified in the textbox.
e^x	exp()	Exponential function is defined by exp(x) = ex, where e is the constant 2.71828…
eye	eyediagram()	Constructs an eye diagram of the waveform.
histogram	histogram()	Creates a histogram of the input waveform showing the magnitude probability density distribution of the waveform. Opens the Histogram Dialog Box to help you specify the parameters.
larger	larger(,)	Creates a new waveform based on the larger of two data points at any given time of the two input waveforms.
lesser	lesser(,)	Creates a new waveform based on the lesser of two data points at any given time of the two input waveforms.
ln	ln()	Returns the natural logarithm of a waveform.
log	log()	Returns the base 10 logarithm of a waveform.
max	max()	Returns the maximum value of a waveform.
min	min()	Returns the minimum value of a waveform.
mod	x % y	Returns the modulus of a waveform. The mod of a waveform is calculated as "x % y" is the remainder of the division x/y for integers x and y.
Plot	Plot	Plots the last result waveform to the active graph window. The button is active only if the expression result is a waveform.
Recall	Recall	Enables you to restore the results of a set of expressions saved to a Tcl file with the Store command.
rms	rms()	Returns the root mean square value of a waveform, depending on the x-axis scale. If the x-axis scale is in Hz then it uses the rms_ac function, otherwise it uses the rms_tran function to calculate the root mean square.

Table 6-22. Statistical Buttons (cont.)

size	size()	Returns the number of data points in an analog waveform or the number of transitions in a digital waveform.
Store	Store	Enables you to save a set of selected expressions in a Tcl file that can be recalled later.
windavg	windavg()	Returns an average value for each x value of the input waveform within a particular "window" based on the current x value.
x^2	sqr()	Returns the squared scalar or waveform. The square of the waveform is calculated as $x^2 = x * x$
x^y	x ** y	Returns "x to the y^{th} power" or x^y, where x and y can be either a waveform or a scalar.
$\sqrt{x}$	sqrt()	Returns the square root of a scalar or a waveform.
$\int$	integral()	Returns the anti-derivative of a waveform.

Trigonometric Buttons

The Trigonometric dropdown list item changes the Waveform Calculator buttons to functions and operators for trigonometric operations/calculations.

Table 6-23. Trigonometric Buttons

Icon	Item	Description
1/X	1 / x	Returns the reciprocal value of a scalar or a waveform.
10^x	pow(10,x)	Returns "10 to the x^{th} power".
abs	abs()	Absolute value of a scalar is the scalar without its sign. The absolute values of a waveform is calculated as the absolute value of an argument abs(x) = x abs(-x) = x
acos	acos()	Trigonometric wave function returns the arc cosine of the waveform.
acosh	acosh()	Trigonometric wave function returns the hyperbolic arc cosine of the waveform.
asin	asin()	Trigonometric wave function returns the arc sine of the waveform.

Table 6-23. Trigonometric Buttons (cont.)

asinh	asinh()	Trigonometric wave function returns the hyperbolic arc sine of the waveform.
atan	atan()	Trigonometric wave function returns the arc tangent of the waveform.
atanh	atanh()	Trigonometric wave function returns the hyperbolic arc tangent of the waveform.
Clear	Clear	Clears the content in the expression textbox.
cos	cos()	Trigonometric wave function finds the cosine of the waveform.
cosh	cosh()	Trigonometric wave function finds the hyperbolic cosine of the waveform.
$\frac{dy}{dx}$	drv()	Returns the derivative of a waveform.
Eval	Evaluate	Evaluates the expression specified in the textbox.
e^x	exp()	Exponential function is defined by exp(x) = ex, where e is the constant 2.71828…
ln	ln()	Returns the natural logarithm of a waveform.
log	log()	Returns the base 10 logarithm of a waveform.
mod	x % y	Returns the modulus of a waveform. The mod of a waveform is calculated as “x % y” is the remainder of the division x/y for integers x and y.
Plot	Plot	Plots the last result waveform to the active graph window. The button is active only if the expression result is a waveform.
Recall	Recall	Enables you to restore the results of a set of expressions saved to a Tcl file with the Store command.
sin	sin()	Trigonometric function returns the sine of the waveform.
sinh	sinh()	Trigonometric wave function returns the hyperbolic sine of the waveform.
Store	Store	Enables you to save a set of selected expressions in a Tcl file that can be recalled later.
tan	tan()	Trigonometric wave function returns the tangent of the waveform.
tanh	tanh()	Trigonometric wave function returns the hyperbolic arc tangent of the waveform.

Table 6-23. Trigonometric Buttons (cont.)

x^2	sqr()	Returns the squared scalar or waveform. The square of the waveform is calculated as $x^2 = x * x$
x^y	x ** y	Returns "x to the y^{th} power" or x^y, where x and y can be either a waveform or a scalar.
$\sqrt{x}$	sqrt()	Returns the square root of a scalar or a waveform.
$\int$	integral()	Returns the anti-derivative of a waveform.

Jitter Tool

The Jitter Tool enables you to analyze clock jitter between any digital and analog target signal with respect to a reference period or frequency, or to calculate phase noise jitter.

The jitter calculation is represented either as a waveform, which shows the jitter evaluation in time, or as a single value. In addition, you can optionally plot a jitter histogram and the following jitter statistics: max, min, mean, standard deviation, and peak-to-peak.

The tool supports the analysis of periodic signals having the sufficient number of cycles necessary for high jitter calculation precision. In this context, “cycle” refers to one complete oscillation of a repeating waveform, and “period” refers to the time taken for one complete cycle of a repeating waveform.

Jitter can be measured against the rising, falling, or both edges of a reference signal. Time domain waveforms are the targeted waveforms.

Using the Jitter Tool 368
Jitter Measurement Types 370

Using the Jitter Tool

You can use the Jitter Tool to analyze clock jitter between any digital and analog target signal with respect to a reference period or frequency, or to calculate phase noise jitter.

Procedure

1. Open a waveform database containing the signal to analyze.
2. Choose **Tools** > **Jitter**.

   The Jitter Tool Dialog Box opens.
3. Select the required tab, either **Time-Domain** or **Phase Noise**. Select the required Jitter measurement type to use. The options offered in the dialog box change based on these selections.
4. If required, type a unique name for the resulting jitter waveform in the Waveform Output Name field. You can also specify whether to plot the jitter waveform in a new graph window.

   Tip

   If a jitter result waveform with the given name is already plotted in the active graph window and you leave the output waveform name unchanged, the jitter result waveform will be updated accordingly to the jitter setup options when you click **Apply** or **OK**.

5. Select the Source Waveform(s) to analyze. You drag waveforms from the active graph window into the Waveform List area, or you can select them and click the Add Selected Waveforms () icon. Use Shift-click or Ctrl-click to select multiple waveforms, or use the Delete key to remove them.
6. Optionally, you can specify a reference value upon which to base the jitter calculation. This can be the Reference Period or Frequency (**Time-Domain** tab) or the Fundamental Frequency (**Phase Noise** tab).
7. Configure the scope of the analysis using the Jitter Setup options.
8. If you intend to generate a graph based on the Time-Domain analysis results, choose which calculated values to display on the waveforms using the Display Jitter Statistics options.
9. Click **Apply**. The analysis is performed and the results plotted in the graph window.

Related Topics

Jitter Tool Dialog Box

Jitter Measurement Types

The Jitter Tool enables you to calculate jitter based on a number of different measurement types.

You can select these when configuring the analysis in the Jitter Tool Dialog Box.

The following Time-Domain and Phase Noise measurement types are available:

Absolute Jitter (Time-Domain) 370
Period Jitter (Time-Domain) 370
Half-Period Jitter (Time Domain) 373
Frequency Jitter (Time Domain) 375
Cycle-to-Cycle Jitter (Time-Domain) 377
Long-Term Jitter (Time-Domain) 379
Time Interval Error (TIE) 380
Jitter Confidence Interval 381
Absolute Jitter (Phase-Noise) 381
Period Jitter (Phase-Noise) 382
Cycle-to-Cycle Jitter (Phase-Noise) 382
Long-Term Jitter (Phase-Noise) 382

Absolute Jitter (Time-Domain)

Absolute Jitter (Time-Domain) is the accumulated difference between the actual cycle period and the reference or average source waveform period values. This type of jitter is useful when evaluating long-term signal stability.

Period Jitter (Time-Domain)

Period Jitter (Time-Domain) is the variation in time of the period of a source waveform relative to its average period or to a reference period, over part of, or the whole waveform. This type of jitter helps to evaluate the short- and long-term signal stability.

The tool provides two methods of measuring Period Jitter:

- Case 1. Reference period is not set—calculation of the difference between the actual and average periods of the source waveform:

$$P_{jit_N} = P_N^{cycle\ src} - \overline{P}$$

- Case 2. Reference period is set—calculation of the difference between the actual source and reference periods:

$$P_{jit_N} = P_N^{cycle\ src} - P^{ref}$$

where:

- P_{jit_N} is the Period Jitter of the Nth cycle
- $\bar{P}$ is the average period of the source waveform
- $P_N^{cycle\ src}$ is the Nth cycle period of the source waveform
- P^{ref} is the reference period
- *N* is the current waveform cycle index, which participates in the calculation:

  $\{N \in \mathbf{N} | 1 \le N \le total\_nb\_of\_cycles\}$

  The Period Jitter RMS value is displayed over the resulting waveform.

- Case 1:

$$Period\ Jitter\ RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_i^{cycle\ src} - \bar{P})^2}$$

- Case 2:

$$Period\ Jitter\ RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(P_i^{cycle\ src} - P^{ref})^2}$$

where:

- *Period Jitter RMS* is the Period Jitter RMS value
- $\bar{P}$ is the average period of the source waveform
- $P_i^{cycle\ src}$ is the ith cycle period of the source waveform
- P^{ref} is the reference period
- **N** is the total number of cycles in the waveform
- *i* is the index of the current cycle

Note

The standard deviation measurement is different from the jitter RMS value measurement (although these values can be very close) as the standard deviation measurement is applied to the jitter result waveform, which already represents the delta values between periods. The period jitter RMS value is calculated using the period values (and not the deltas between periods) of the source waveform, and the reference period, if present.

$$Std.\ dev. = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\Delta P_i - \overline{\Delta P})^2}$$

$$\Delta P_i = P_{jit_K}$$

where:

- **Std. dev.** is the standard deviation value
- ΔP_i is the ith difference (delta) between the periods, which is equal to the P_{jit_K} (see equations)
- $\overline{\Delta P}$ is the average difference (delta) between periods
- **N** is the number of calculated differences between deltas
- P_{jit_K} is the Period Jitter of the Kth cycle (see equations)
- **K** is the current waveform cycle index, which participates in the calculation:

  $$\{K \in \mathrm{N} | 1 \le K \le total\_nb\_of\_cycles\}$$

- **i** is the index of the Period Jitter value for the current cycle.

Figure 6-18. Example of the Period Jitter Waveform

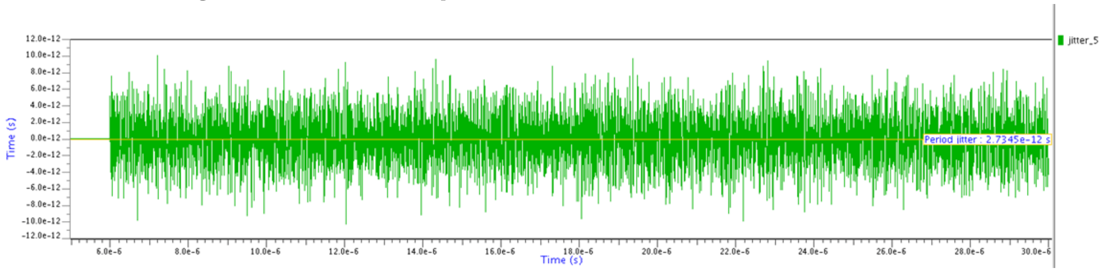


Figure 6-19. Example of the Period Jitter Waveform with Applied Statistical Measurements

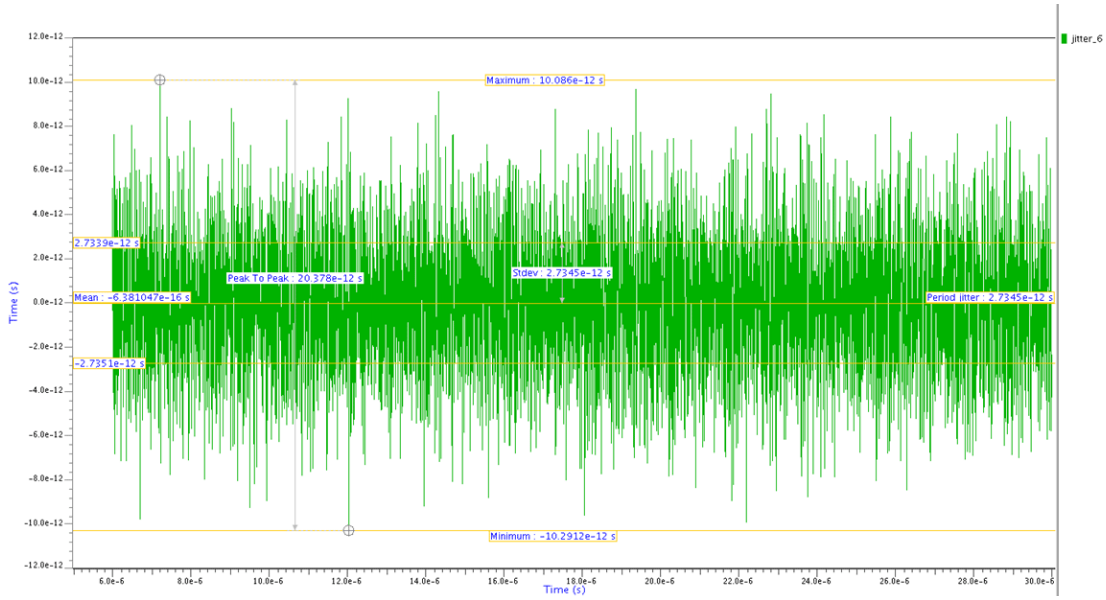
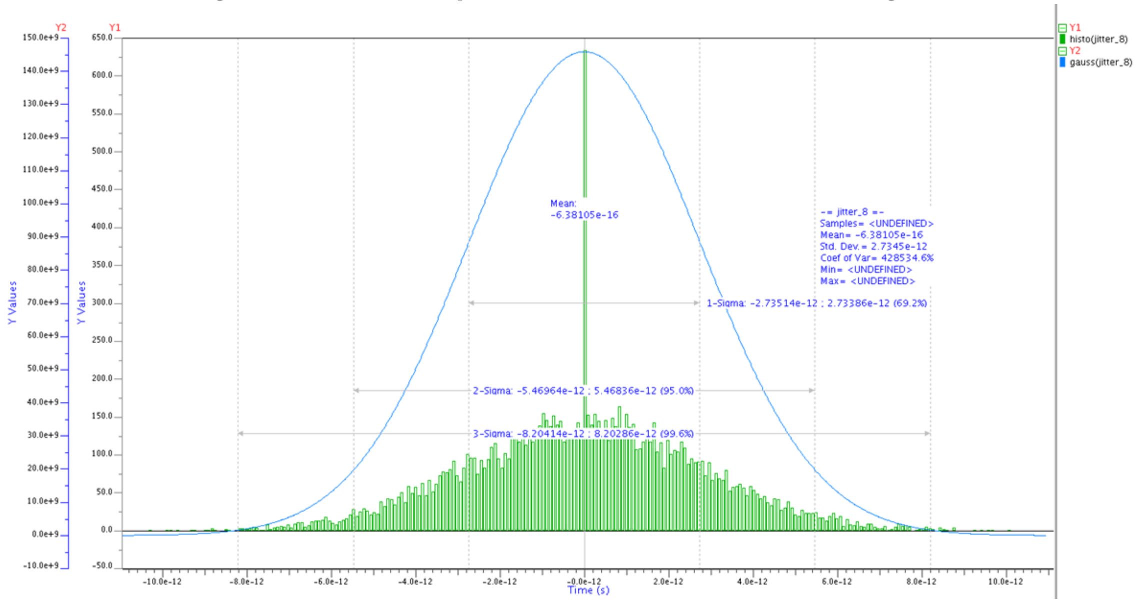


Figure 6-20. Example of the Period Jitter Histogram



Half-Period Jitter (Time Domain)

Half-period Jitter is the variation in time of the half-period of a source waveform relative to its average half-period, or to a reference half-period, over part of, or the whole waveform. This type of jitter helps to evaluate the short- and long-term signal stability.

A waveform's half-period is the period multiplied by its duty cycle.

The tool provides two methods of measuring Half-period Jitter:

- Case 1. Reference period is not set—calculation of the difference between the actual and average half-periods of the source waveform:

$$HP_{jit_N} = HP_N^{cycle\ src} - \overline{HP}$$

- Case 2. Reference period is set—calculation of the difference between the actual source and reference waveform half-periods:

$$HP_{jit_N} = HP_N^{cycle\ src} - HP^{ref}$$

where:

- HP_{jit_N} is the Half-period Jitter of the Nth cycle
- $\overline{HP}$ is the average half-period of the source waveform
- $HP_N^{cycle\ src}$ is the Nth cycle half-period of the source waveform
- HP^{ref} is the reference half-period
- N is the current waveform cycle index, which participates in the calculation:

  $\{N \in \mathbb{N} | 1 \le N \le total\_nb\_of\_cycles\}$

The Half-Period Jitter RMS value is displayed over the resulting waveform.

- Case 1:

$$Half\_period\ jitter\ RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(HP_i^{cycle\ src} - \overline{HP})^2}$$

- Case 2:

$$Half\_period\ jitter\ RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(HP_i^{cycle\ src} - HP^{ref})^2}$$

where:

- **Half_period Jitter RMS** is the Half-period Jitter RMS value
- $\overline{HP}$ is the average half-period of the source waveform

- $HP_i^{cycle\,src}$ is the ith cycle half-period of the source waveform
- HP^{ref} is the reference half-period
- N is the total number of cycles in the waveform
- i is the index of the current cycle

$$Std.\ dev.\ = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\Delta HP_i - \overline{\Delta HP})^2}$$

$$\Delta HP_i = HP_{jit_K}$$

where:

- *Std. dev.* is the standard deviation value
- ΔHP_i is the ith difference (delta) between the half-periods, which is equal to the HP_{jit_K} (see formulas above)
- $\overline{\Delta HP}$ is the average difference (delta) between half-periods
- N is the number of calculated differences between deltas
- HP_{jit_K} is the Half-period Jitter of the Kth cycle (see above for formula)
- K is the current waveform cycle index, which participates in the calculation:

  $$\{K \in \mathrm{N} | 1 \le K \le total\_nb\_of\_cycles\}$$

- i is the index of the Half-period Jitter value for the current cycle.

Frequency Jitter (Time Domain)

Frequency Jitter is the variation in time of the frequency of the source waveform relative to its average frequency or to a reference frequency over part of, or the whole waveform. This type of jitter helps to evaluate the short- and long-term signal stability.

The tool provides two methods of measuring Frequency Jitter:

- Reference frequency is not set—calculation of the difference between the actual and average frequencies of the source waveform:

  $$F_{jit_N} = F_N^{cycle\,src} - \overline{F}$$

- Reference frequency is set—calculation of the difference between the actual source waveform and reference frequencies:

$$F_{jit_N} = F_N^{cycle\ src} - F^{ref}$$

where:

- F_{jit_N} is the Frequency Jitter for the Nth cycle
- $\bar{F}$ is the average frequency of the source waveform
- $F_N^{cycle\ src}$ is the frequency of the Nth cycle of the source waveform
- F^{ref} is the reference frequency
- N is the current waveform cycle index, which participates in the calculation:

$$\{N \in \mathbf{N} | 1 \leq N \leq total\_nb\_of\_cycles\}$$

The Frequency Jitter RMS value is displayed over the result waveform.

- Case 1:

$$Frequency\ jitter\ RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(F_i^{cycle\ src} - \bar{F})^2}$$

- Case 2:

$$Frequency\ jitter\ RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(F_i^{cycle\ src} - F^{ref})^2}$$

where:

- *Frequency jitter RMS* is the Frequency Jitter RMS value
- $\bar{F}$ is the average frequency of the source waveform
- $F_i^{cycle\ src}$ is the ith cycle frequency of the source waveform
- F^{ref} is the reference frequency
- N is the total number of cycles in the waveform
- i is the index of the current cycle

$$Std.\ dev. = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\Delta F_i - \overline{\Delta F})^2}$$

$$\Delta F_i = F_{jit_K}$$

where:

- *Std. dev.* is the standard deviation value
- DF_i is the ith difference (delta) between frequencies, which is equal to F_{jit_K} (see formulas above)
- $\overline{\Delta F}$ is the average difference (delta) between frequencies
- N is the number of the calculated differences between deltas
- F_{jit_K} is the Frequency Jitter value for the Kth cycle (see above for the formula)
- K is the current waveform cycle index, which participates in the calculation:

  $$\{K \in \mathrm{N} | 1 \le K \le total\_nb\_of\_cycles\}$$

- i is the index of the Frequency Jitter value for the current cycle.

Cycle-to-Cycle Jitter (Time-Domain)

Cycle-to-cycle (C2C) Jitter is the variation in time of the difference between the duration of two adjacent cycles over part of, or the whole waveform. This type of jitter is calculated only on the source waveform. C2C helps to evaluate the instantaneous and very short-term signal stability.

Cycle-to-cycle jitter is calculated using the following equation:

$$NC_{jit_K} = \sum_{i=N+1}^{2N} P_i^{cycle} - \sum_{i=1}^{N} P_i^{cycle}$$

where:

- NC_{jit_K} is the N-cycle Jitter for the Kth adjacent N-cycle samples pair
- P_i^{cycle} is the period of the ith cycle in the sample
- N is the number of cycles in the sample
- i is the index of the current cycle in the sample

- K is the current N-cycle sample pair index, which participates in the calculation:

$$\left\{ K \in \mathbf{N} \middle| 1 \le K \le \left\lfloor \frac{total\_nb\_of\_N\_cycle\_samples}{2} \right\rfloor \right\}$$

$$total\_nb\_of\_N\_cycle\_samples = \left\lfloor \frac{total\_nb\_of\_cycles}{N} \right\rfloor$$

The cycle-to-cycle jitter RMS value is displayed over the result waveform.

$$N\_Cycle\,jitter\,RMS = \sqrt{\frac{1}{N}\sum_{z=1}^{N}\left(\sum_{i=K+1}^{2K} P_i^{cycle} - \sum_{i=1}^{K} P_i^{cycle}\right)^2}$$

where:

- $N\_Cycle\ jitter\ RMS$ is the cycle-to-cycle jitter RMS value
- P_i^{cycle} is the period of ith cycle in the sample
- K is the number of cycles in the sample
- N is the number of the calculated differences between adjacent N-cycle samples
- i is the index of the current cycle in the sample

$$Std.\ dev. = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\Delta NC_i - \overline{\Delta NC})^2}$$

$$\Delta NC_i = NC_{jit_K}$$

where:

- $Std.\ dev.$ is the standard deviation value
- ΔNC_i is the ith difference (delta) between the duration of the adjacent N-cycle samples, which is equal to the NC_{jit_K} (see formulas above)
- $\overline{\Delta NC}$ is the average difference (delta) between adjacent N-cycle samples
- N is the number of calculated differences between deltas
- NC_{jit_K} is the N-cycle Jitter of the Kth adjacent N-cycle samples pair (see above for formula)

- K is the current N-cycle sample pair index, which participates in the calculation:

$$\left\{K \in \mathbf{N} \middle| 1 \le K \le \left\lceil \frac{total\_nb\_of\_N\_cycle\_samples}{2} \right\rceil \right\}$$

$$total\_nb\_of\_N\_cycle\_samples = \left\lceil \frac{total\_nb\_of\_cycles}{N} \right\rceil$$

- i is the index of the N-cycle Jitter value for the current N-cycle samples pair.

Long-Term Jitter (Time-Domain)

Long-term Jitter is the variation in time of the cumulative period of adjacent N-cycle samples. This type of jitter is sometimes called "accumulated" jitter. It can help to evaluate the signal's accuracy and stability.

In other words, Long-term Jitter is the ensemble of N-period Jitter RMS values, for N starting from 1 to selected maximum number of cycles in the N-cycle sample. It means that the first value on the Long-term Jitter waveform will correspond to the Period Jitter RMS value (N=1).

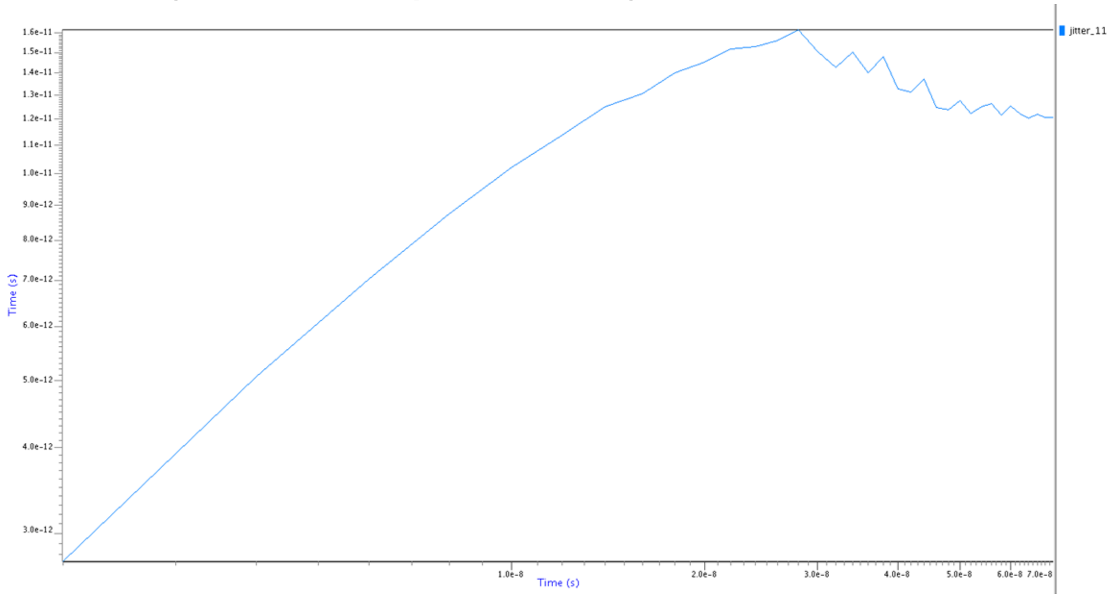
$$LT_{jit_N} = N\_period\ jitter\ RMS_N$$

where:

- LT_{jit_N} is the Long-term Jitter value for N number of cycles in the sample
- $N\_period\ jitter\ RMS_N$ is the N-period Jitter RMS value for the N number of cycles in the sample
- N is the current number of cycles in the N-cycle sample

$$\{N \in \mathbf{N} | 1 \le N \le max\_nb\_of\_cycles\_in\_the\_sample\}$$

Figure 6-21. Example of the Long-Term Jitter Waveform



Time Interval Error (TIE)

Time Interval Error (TIE) is the variation in time of the source waveform edges relative to the reference (or ideal clock) waveform edges over part of, or the whole waveform. TIE helps to evaluate the signal accuracy and the short- and long-term signal stability.

The TIE value is calculated using the equation:

$$TIE_{jit_N} = E_N^{src} - E_N^{ref}$$

where:

- TIE_{jit_N} is the time interval error at the Nth edge
- E_N^{src} is the Nth source waveform edge
- E_N^{ref} is the Nth reference waveform edge

The TIE Jitter RMS value is displayed over the result waveform.

$$TIE\ jitter\ RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(E_i^{src} - E_i^{ref})^2}$$

where:

- *TIE jitter RMS* is the Time Interval Error Jitter RMS value

- E_i^{src} is the ith source waveform edge
- E_i^{ref} is the ith reference waveform edge
- N is the total number of calculated edges in the waveform
- i is the index of the current edge

$$Std.\ dev.\ = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(\Delta TIE_i - \overline{\Delta TIE})^2}$$

$$\Delta TIE_i = TIE_{jit_K}$$

where:

- *Std. dev.* is the standard deviation value
- $DTIE_i$ is the ith difference between edges, which is equal to TIE_{jit_K} (see formulas above)
- $\overline{\Delta TIE}$ is the average difference between edges (the average TIE jitter value)
- N is the total number of the calculated differences between edges (the total number of TIE jitter values)
- TIE_{jit_K} is the TIE jitter for the Kth edge (see above for the formula)
- K is the index of the edge, which participates in the TIE jitter calculation:

  $$\{K \in \mathrm{N} | 1 \le K \le total\_nb\_of\_edges\}$$

- i is the index of the current TIE jitter value.

Jitter Confidence Interval

Jitter Confidence Interval plots the approximate confidence interval for rms jitter vs. number of Gaussian samples.

Absolute Jitter (Phase-Noise)

Absolute Jitter (Phase-Noise) is applied to SST Noise Analysis results. Absolute jitter is calculated as the result of the integration of the *sphi* double side band waveform.

Period Jitter (Phase-Noise)

Period Jitter (Phase Noise) is applied to SST Noise Analysis results. Period jitter is calculated as the result of the integration of the *sphi* double side band waveform.

Cycle-to-Cycle Jitter (Phase-Noise)

Cycle-to cycle (CtoC) Jitter (Phase Noise) is applied to SST Noise Analysis results. Cycle-to cycle (C2C) jitter is calculated as the result of the integration of the *sphi* double side band waveform.

Cycle-to-cycle jitter (phase noise) is calculated using the following equation:

$$\sigma^2{}_{cc} = \frac{4}{(\pi f_0)^2} \int_0^{+\infty} S_\phi(f) sin^4(\pi f T_0) df$$

Long-Term Jitter (Phase-Noise)

Long-term Jitter (Phase Noise) is applied to SST Noise Analysis results. Long-term jitter is calculated as the result of the integration of the *sphi* double side band waveform.

Extracting Outputs from a Database

This tool enables you to run a new simulation on a previously simulated database, returning—and, optionally, plotting—the results of all .extract and .meas commands defined in a new netlist. This enables you to modify and add measurements to a simulation without running the entire simulation again, as well as to apply a single set of measurements to multiple simulations.

Note

It is possible to run the tool in batch mode; see the description of the dataset extract command for details.

Prerequisites

- You must have a database file containing the results of the simulation upon which you want to run the extract.
- You must have a netlist (*.cir*/*.ckt*) file containing the .extract or .meas statements to apply to the simulation results.

Procedure

1. Choose **Tools** > **Extract**.

   The Extract Outputs From Database dialog box opens.

2. Specify the database upon which to perform the extract. Either select an open database by clicking the down-arrow to the right of the Select Source Database field, or click the open icon () and browse to the database file.
3. The Select Netlist field defaults to the netlist with the same name as the database file in the same directory. If you want to run the extract with a different netlist, either edit the path or browse to the file.
4. The AEX Output File defaults to *<netlist_name>.aex*. If you want to specify a different file to which to write the values from the new extract run, either edit the path or browse to the file.
5. By default, waveforms are not generated from the extracted values. To generate them, select the Extract Waveforms option. The Output Wdb File field defaults to a new *.wdb* file named according to the original simulation filename suffixed with *_extract*, but this can be overridden.
6. If required, specify any arguments to pass to Eldo in the Eldo Extra Arguments field.
7. Click **Run**. The simulation runs, with the progress displayed in the transcript window.

Related Topics

Extract Outputs From Database Dialog Box

Signal Processing Functions

A number of signal processing functions are available in EZwave.

Using the Fast Fourier Transform Tool **384**
Using the Inverse Fast Fourier Transform Tool **386**
Using the Spectrum Measurement Tool **388**
Autocorrelation Function and Power Spectral Density **391**
Correlogram Method 391
Periodogram Method 392
Convolution Function **392**
Harmonic Distortion Function **394**
Signal to Noise Function **394**
Signal Processing Function Window Shapes **395**
Windowing Transforms **396**

Using the Fast Fourier Transform Tool

The Discrete Fourier Transform (DFT) computes time and frequency domain expressions for data sampled at discrete intervals. The Fast Fourier Transform (FFT) is the fastest, most efficient algorithm for computing the DFT.

Prerequisites

If you want to replicate Eldo FFT results in the EZwave viewer, verify the following conditions are true:

- The input waveforms must be the same. In the Eldo simulator, you can display the waveform it uses for the FFT calculation by setting display_input = 1 in the .optfour command.
- The parameter and option settings must be the same for both FFT calculations

Procedure

1. Open the Fast Fourier Transform Tool Dialog Box by one of the following methods:
   - Choose **Tools** > **FFT**

   or

   - Access through the Waveform Calculator:
     i. Click the Waveform Calculator  icon, or select it from the **Tools** menu.
     ii. Choose **Signal Processing** from the dropdown list.

iii. Click the **fft** button on the calculator keypad.

2. Specify a source wave in the Source Waveform field. Select a wave from an open graph window and use the Add Selected Waveforms icon to add it to the drop down list. Use the Clear Waveform List icon to remove all waveforms from the list.

Tip

You can perform an FFT on complex waveforms, which is useful when the negative and positive frequency responses are not symmetrical and need to be considered individually. This situation can occur, for example, when designing data converters.

3. Select an FFT input parameter that you want to calculate automatically from the **Input Method** dropdown list.
4. Specify the remaining input parameters. You can click the **Default** button if you want to restore the default values.

   These parameters give information about the sampling of the input data and the range of the input signal which is used for the computation.
5. (Optional) Modify the input signal range:
   - **Start/Stop Time —** specifies the start and stop times for the computation
   - **Points /Frequency —** specifies the number of sampling points and the sampling frequency for the computation.

   These parameters satisfy the following equation:

   ((Points-1)/ Frequency) = Stop Time - Start Time

   The number of points for the FFT results is Points/2.

Note

If the number of points is set to a factor of 2^n (n = 2, 3, ...), the FFT computation is more efficient.

You can type expressions into any numeric fields. For example, to convert a 1µs period to a frequency, type the expression 1/1u into the Sampling Frequency field.

6. (Optional) Select **File** > > **Save Configuration**at the top of the Fast Fourier Transform Tool dialog box if you want to save the FFT configuration.

   All parameter of the FFT Tool GUI are saved, except the source waveform and the output waveform names. You can subsequently load a saved FFT configuration by selecting **File** > **Open Configuration**.
7. Click **Apply**.

If problems in the computation occur, a relevant error or warning message is displayed. The transformation is calculated based on the specified parameters and a new graph window is opened and the results are graphed.

Tip

If you opened the Fast Fourier Transform Dialog Box by using the Waveform Calculator **fft** button and no results are displayed, click the **Eval** button and then the **Plot** button.

Related Topics

Fast Fourier Transform Tool Dialog Box

Waveform Calculator Calculation Options

Windowing Transforms

Using the Inverse Fast Fourier Transform Tool

Using the Inverse Fast Fourier Transform Tool

For any two signals in the frequency domain, it is possible to perform an Inverse Fast Fourier Transform (IFFT).

Procedure

1. To open the Inverse Fast Fourier Transform Tool dialog box:

    a. Click the Waveform Calculator icon.

    b. Choose **Signal Processing** from the dropdown list.

    c. Click the **ifft** button on the calculator keypad.

2. Specify a source wave in the Source Waveform field. Select a wave from an open graph window and use the Add Selected Waveforms icon to add it to the drop down list. Use the Clear Waveform List icon to remove all waveforms from the list.

    The Parameter Setup controls are populated with a default set of parameters. These parameters reflect the sampling of the input data and the range of the input signal which is used for the computation.

3. Select an IFFT input parameter that you want to calculate automatically from the Input Method dropdown list in the Parameter Setup area.

4. Specify the remaining input parameters. You can click the **Default** button if you want to restore the default values.

    These parameters give information about the sampling of the input data and the range of the input signal which is used for the computation.

5. (Optional) Modify the input signal range:
   - **Start/Stop Freq —** specifies the start and stop frequencies for the computation
   - **Points /Time —** specifies the number of sampling points and the sampling time for the computation

   Note

   An IFFT analysis always creates results with an even number of points. Be aware that when calculating results in conjunction with an FFT analysis, an even number of points with the FFT must also be used if the following condition is to be fulfilled:

   IFFT (FFT(signal)) = signal

   You can type expressions into any numeric fields. For example, to convert a 1µs period to a frequency, type the expression 1/1u into the Stop Frequency field.

6. Click the Improve IFFT Results checkbox to display optional controls to set the sampling method:
   - Use Data Points should be used if the input data has equidistant Time Steps; otherwise, use Uniform Sampling. Select Cubic Spline to compute interpolated points using the Cubic Spline method rather than using linear interpolation.
   - Data Padding — Select Pad with Zeros to pad the input data with zeros, before or after the input data set

7. Click **Apply**. If problems in the computation occur, a relevant error or warning message is displayed.

8. Return to the Waveform Calculator to evaluate and plot the results of the transformation.

   a. Click the **Eval** button. The transformation will now be calculated based on the specified parameters.

9. Click the **Plot** button to plot the results. A new graph window is opened and the results are graphed.

Related Topics

Inverse Fast Fourier Transform Dialog Box

Waveform Calculator Calculation Options

Windowing Transforms

Using the Fast Fourier Transform Tool

Using the Spectrum Measurement Tool

You can compute spectrum measurements on waveforms using the Spectrum Measurement Tool. Calculates Signal-to-Noise-and-Distortion Ratio (SNDR), Spurious Free Dynamic Range (SFDR), Effective Number Of Bits (ENOB), Total Harmonic Distortion (THD) and Signal-to-Noise Ratio (SNR) (without distortion) by using a Fast Fourier Transform (FFT) of a real, complex, or transient input signal. The Spectrum Measurement Tool is typically used for characterizing A-to-D converters and is supported for transient simulation data.

Procedure

1. Open the Spectrum Measurement Tool Dialog Box by one of the following methods:
   - Choose **Tools > Spectrum Measurement**

   or

   - Access through the Waveform Calculator:
     i. Click the Waveform Calculator icon, or select it from the **Tools** menu.
     ii. Choose Signal Processing from the dropdown list.
     iii. Click the **spec. meas.** button on the calculator keypad.
2. On the **Settings** tab, use the **Add Selected Waveform** icon to add the currently selected waveform(s). Use the **Clear Waveform List** icon to remove all waveforms in the list. Use the **Delete** icon to remove the specified waveform from the list.
3. In the FFT Setup area, choose one of the FFT input parameters that you want to be calculated automatically using the **Input Method** dropdown list. Then specify the remaining input parameters. These parameters give information about the sampling of the input data and the range of the input signal which is used for the computation. Click the **Default** button to restore the default values.

   The range of the input signal can be modified using the following input parameters:
   - Start/Stop Time — specify the start and stop times for the computation
   - Points /Frequency — specify the number of sampling points and the sampling frequency for the computation.

   Note

   The parameters above satisfy the following equation:

   ((Points-1)/ Frequency) = Stop Time - Start Time

4. Enter the parameters in the Measurement Setup area:

    - Frequency Min/Max — specify the lowest and highest frequencies to used in the spectrum measurement noise integration calculation. Click the button refresh the default minimum and maximum frequencies for the selected waveform(s).
    - Harmonics — specify the number of harmonics of the signal to be considered. Default value is 6 (signal + 5 harmonics).
    - Windowing — select the windowing type to specify the method for computing the sampled data. Select the windowing method from the dropdown list. The default option is Rectangular. Certain windowing methods require additional input, type the appropriate value in the box that appears. Refer to "Windowing Transforms" on page 396 for details on the options available.
    - Bin Size— specifies the number of points to take into account around the fundamental signal. For example, with Bin Size = 0, only the fundamental is taken into account. With Bin Size = 1, the frequencies taken into account correspond to the first point to the left and right for the found fundamental and each of its associated harmonics. The default value depends on the Windowing transform selected.

5. (Optional) Check Display FFT if you want to plot the FFT of the waveform(s). You can also specify the Unit, Spectral or Continuous plot, and Linear or Log scale.
6. (Optional) Select **File > Save Configuration** at the top of the Spectrum Measurement Tool dialog box if you want to save the Spectrum Measurement configuration. All parameter of the Spectrum Measurement Tool GUI are saved, except the source waveform and the output waveform names. You can subsequently load a saved Spectrum Measurement (or FFT) configuration by selecting **File > Open Configuration**.
7. Click **Measure**. If problems occur in the computation, a relevant error or warning message is displayed.

    The spectrum measurements are calculated based on the specified parameters and are displayed in both the **Measurement Results** tab and the Waveform Calculator.

    Tip

    If the input is a compound waveform, a new entry is created in the **Measurement Results** tab for each waveform contained in the compound.

8. (Optional) Right-click a measurement displayed in the **Measurement Results** tab and choose:

    - **Send expr to calc** - to send the expression to the Waveform Calculator, where it is evaluated.
    - **Copy result** - to copy the value to the clipboard.

If you right-click the top level waveform name, you can choose:

- **Send all exprs to calc** - to send all of the expressions as a single command to the Waveform Calculator, where they are evaluated.
- **Remove result** - to remove all of the results for the waveform.

Related Topics

Spectrum Measurement Tool Dialog Box

spectrummeasurement

Autocorrelation Function and Power Spectral Density

Two methods are available for calculating the autocorrelation function (AF) and the power spectral density (PSD) of a signal waveform. The AF is an average measure of its time domain properties; as such, it can be especially relevant when the signal is random.

The Correlogram Method means that the FFT is used directly to compute estimates of the AF *Rxx(n)* for N_{auto} lags, where 2 x N_{auto} is the size of the transform used. The Periodogram Method means that a sliding FFT is used to compute estimates of the PSD directly rather than estimating an AF.

Correlogram Method 391
Periodogram Method 392

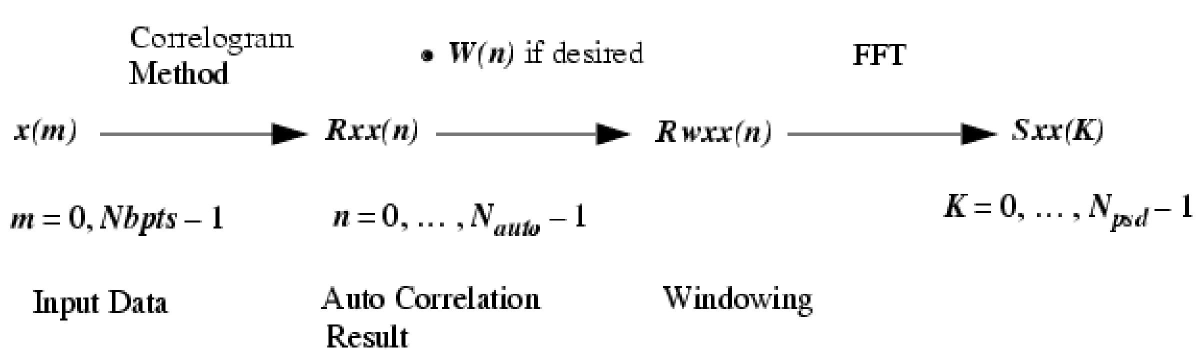
Correlogram Method

This method uses the FFT directly to compute estimates of the autocorrelation function *Rxx(n)* N_{auto} lags, where 2 x N_{auto} is the size of the transform used.

For this, the program divides the sampled input data set into K = ($Nbpts$ + N_{auto}) / (N_{auto} + 1) sections and the FFT routine (2 x (N_{auto} -1) points) is performed once on each section with the appropriate input data. The autocorrelation result is then computed by performing an Inverse FFT (2 x (N_{auto} -1) points) after accumulating the partial results.

At this point, a spectral estimate with any desired degree of frequency resolution can be obtained by augmenting the just computed *R(n)* or the first $Ncorr \leq N_{auto}$ values with zeros and performing a single FFT (2 x (N_{psd} - 1) points).

Figure 6-22. Correlogram Method



To avoid the undesired effects of truncating data records in the PSD result, it is possible to apply different kinds of smoothing windows on the autocorrelation result.

Periodogram Method

This method uses a sliding FFT to compute estimates of the PSD directly rather than estimating an autocorrelation function as in the Correlation method.

The given sequence *X(m)*, *m*=0, ..., *Nbpts* – 1 is first decomposed into subsequences *Xr(m)* of length *Nsect* with overlapping *D* = *Nsect*/2. Each of them is shifted by the arithmetic mean of all the data.[1]

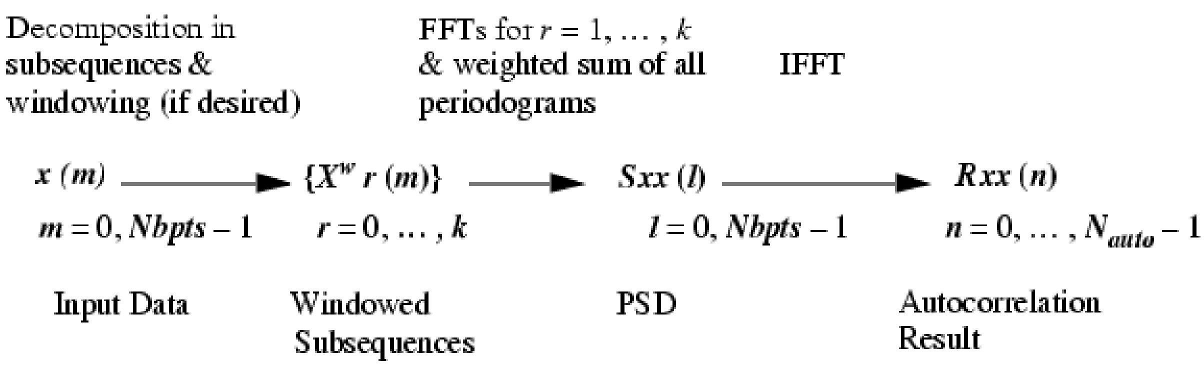
$$Xr(m) = x(m + (r-1)D) - xmean \qquad r = 1, ..., k$$

where $k = (Nbpts - D)/D$ is the number of sequences.

On each section, a window is then applied and the PSD is computed (FFT with 2 x (N_{auto} – 1) points) as a weighted sum of their periodograms (= | $X^W r(k)|^2$ *energy_of_window*).

The inverse FFT (2 x (N_{auto} – 1) points) is now used to estimate the autocorrelation function from the PSD result.

Figure 6-23. Periodogram Method



If the user wishes to have more values for the PSD result than computed by default, an additional FFT (2 x (N_{psd} – 1) points) is executed on a sequence, generated by the first N_{corr} autocorrelation values and a suitable number of zeros.

Convolution Function

The convolution function calculates the discrete linear convolution between two data sets.

1. The subtraction of *xmean* is implemented in order that the autocorrelation result is centered around y = 0. This useful when using non-rectangular waveforms.

For two finite data sequences, $x(n)$, $n = 0, \ldots, N-1$ and $h(n)$, $n = 0, \ldots, M-1$, the discrete linear convolution $x(n) \cdot h(n)$ is defined as follows:

$$y(n) = \sum_{m=-\infty}^{\infty} x(m) \cdot h(n-m)$$

For signals $x(n)$, $h(n)$ which are periodic with period N, the discrete FFT of their periodic convolution is equal to the multiplication of the separate FFT results. This relation is called the *Discrete Convolution Theorem*. Using zero padding of $x(n)$ and $h(n)$ to make circular convolution yields the same result as linear convolution. The following method can then be applied for computing the linear convolutions of two finite data sequences:

$x(n)$, $n = 0, \ldots, N-1$ and

$h(n)$, $n = 0, \ldots, M-1$.

- **Step 1 — Zero Padding**

Table 6-24. Zero Padding

$x'(n) = x(n)$	$n = 0, \ldots, N-1$
$x'(n) = 0$	$n = N, \ldots, N+M-1$
$h'(n) = h(n)$	$n = 0, \ldots, M-1$
$h'(n) = 0$	$n = M, \ldots, N+M-1$

- **Step 2 — Multiply FFTs**

Table 6-25. Multiply FFT

of $x'(n)$ and $h'(n)$
$x'(n) \rightarrow X'(k)$ FFT
$h'(n) \rightarrow H'(k)$ FFT
Multiplication:
$X'(k) \cdot H'(k) = Y'(k)$

- **Step 3 — Inverse FFT**

Table 6-26. Inverse FFT

$Y'(k) \rightarrow y(n) = x(n)*h(n)$ IFFT

Harmonic Distortion Function

This topic describes how the harmonic distortion function computes the harmonics and the total harmonic distortion (THD) of the input waveform signal.

Note

Only a frequency-domain gain from a uniformly sampled FFT result is accepted as a valid source waveform.

The THD of a signal is the ratio of the sum of the powers of all harmonic frequencies above the fundamental frequency to the power of the fundamental frequency. This value is expressed in dB.

The harmonics inside the interval [*Fmin*, *Fmax*] are computed as follows:

$$harmonic(i) = \left|\frac{A(i)}{Ao}\right|$$

where:

- A(i) = amplitudes of the multiples of the fundamental frequency.
- Ao = amplitude of the fundamental frequency.

The Total Harmonic Distortion is given by the following:

$$tot\_harm = \sqrt{\sum \frac{|A(i)|^2}{|Ao|^2}}$$

where the sum is computed over all multiples (equal or greater than 2) of the fundamental frequency in the specified band. If these values are not identical to the sampled data values, then they are computed by interpolation.

Note

Each harmonic frequency is a multiple of the fundamental frequency. Only those harmonic frequencies above the fundamental frequency and inside the [*fmin*, *fmax*] interval are used for the THD computation.

Signal to Noise Function

Computes the signal to noise ratio of the input waveform signal by using the Gain of the FFT result.

Note

Only a complex waveform or a waveform representing a Gain is accepted as a valid source waveform.

Tip

See also snr and Signal to Noise Ratio Dialog Box.

In digital signal analysis, it is often of interest to compute, for a signal composed of the sum of noisy sinusoids, the following relationship:

$$10\log 10\left(\frac{\sum |AS(i)|^2}{\sum |AN(j)|^2}\right)$$

where:

$\sum |AS(i)|^2$ = sum over all squares of amplitudes of the sinusoidal frequencies.

$\sum |AN(j)|^2$ = sum over all squares of noise amplitudes of the signal.

This is called the *Signal to Noise Ratio* of the signal.

The calculation takes into account harmonics (default is 6, signal + 5 harmonics). You can also control the sampling frequency of the source waveform to manage harmonic folding and aliasing (default is none).

Signal Processing Function Window Shapes

This topic describes the two window shapes, Symmetric and Periodic, which you can use with EZwave signal processing functions.

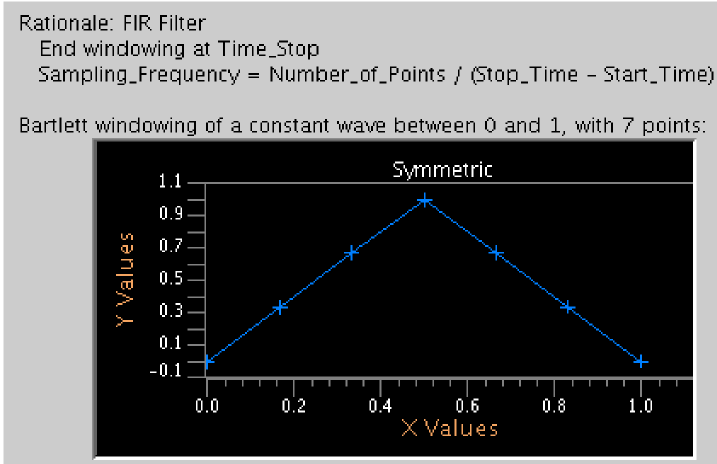
- **Symmetric**

  The default window shape.

  For symmetric windows, the input parameters satisfy the following equation:

  $$sampling\_frequency = (number\_of\_points - 1)/(time\_stop - time\_start)$$

Figure 6-24. Symmetric Window



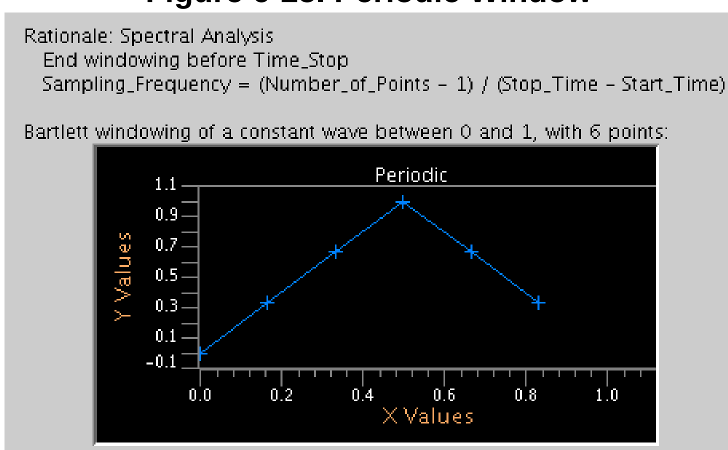
- **Periodic**

  Setup is enhanced for spectral analysis of periodic signals.

  For periodic windows, the input parameters satisfy the following equation

$$\mathrm{sampling\_frequency} = (\mathrm{number\_of\_points})/(\mathrm{time\_stop} - \mathrm{time\_start})$$

Figure 6-25. Periodic Window



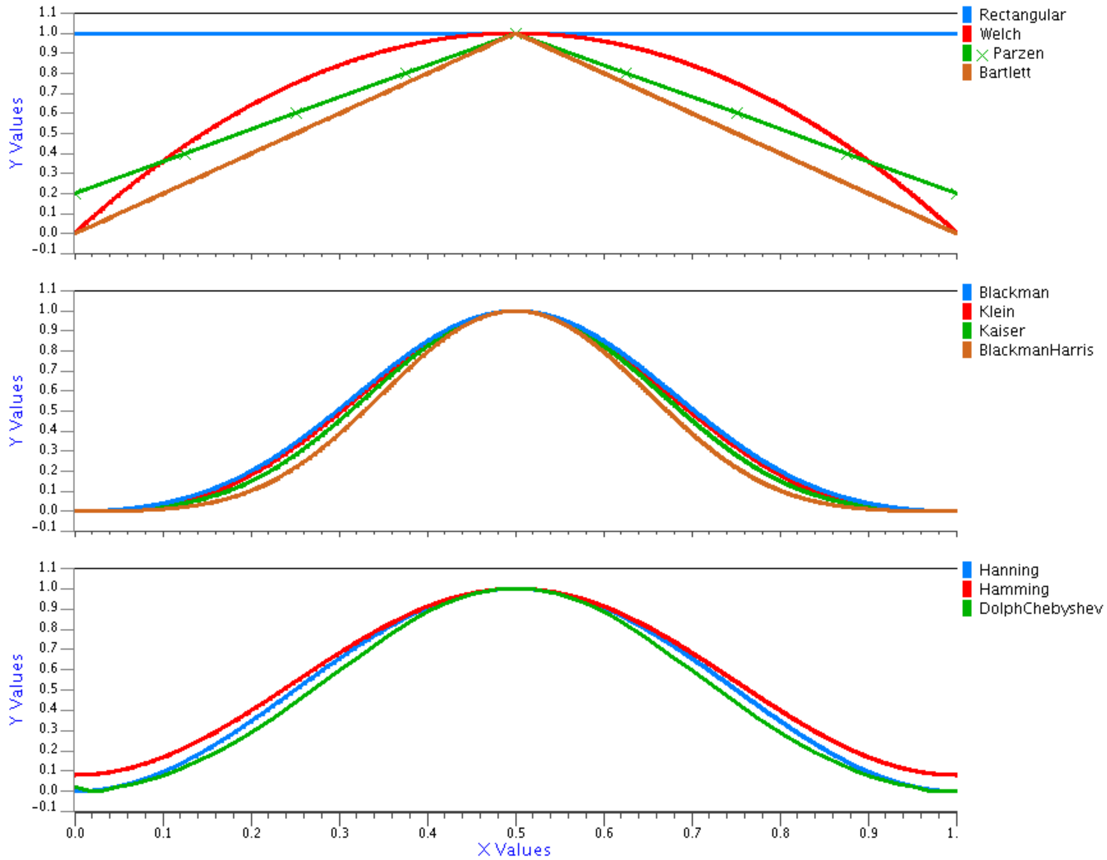
You can control the default window shape setting for FFT windows and functions using the Waveform Calculator Calculation Options.

Windowing Transforms

To estimate the power density spectrum of a random signal, only a finite part of the signal is used in practice even if the signal is of infinite duration. To reduce the undesirable effects of truncating the data records (leakage), it is convenient to apply different types of windows that gradually taper the data near the ends of the record, thereby avoiding the abrupt truncation of a rectangular window.

11 different types of windowing transforms are available:

Figure 6-26. Available Windowing Transforms



For a record consisting of N points indexed from 0 to N - 1, EZwave works with the following equations:

- **Bartlett Window**

$$w(i) = \frac{2i}{N-1} \qquad 0 \le i \le \frac{N-1}{2}$$

$$w(i) = 2 - \frac{2i}{N-1} \qquad \frac{N-1}{2} \le i \le N-1$$

- **Blackman Window**

$$w(i) = 0.42 - 0.5\cos\left(\frac{2\pi \times i}{N-1}\right) + 0.08\cos\left(\frac{4\pi \times i}{N-1}\right)$$

- **Blackman-Harris Window**

$$w(i) = 0.35875 - 0.48829 \cdot \cos\left(\frac{2\pi \cdot i}{N-1}\right) + 0.14128\cos\left(\frac{4\pi \cdot i}{N-1}\right) - 0.01168\cos\left(\frac{6\pi \cdot i}{N-1}\right)$$

- **Dolph-Chebyshev Window**

$$W(i) = \frac{-1^{i} \cdot \cos\left(N \cdot \cos^{-1}\left(B \cdot \cos\left(\frac{\pi \cdot i}{N}\right)\right)\right)}{\cosh(N \cdot \cosh^{-1}(B))}$$

where:

$$B = \cosh\left[\left(\frac{1}{N}\right) \cdot \cosh^{-1}(10^{a})\right] \qquad 0 < |i| < N-1$$

- **Hamming Window**

$$w(i) = 0.54 - 0.46 \cdot \cos\left(\frac{2\pi \times i}{N-1}\right) \qquad i = 0, \ldots, N-1$$

- **Hanning Window**

$$w(i) = \alpha - \alpha\cos\left(\frac{2\pi \times i}{N-1}\right) \qquad 0 \le \alpha \le 1$$

$$\text{Default } \alpha = 0.5$$

Note

The second half of the Hanning window, $M \le i \le N$ where $M = N/2$ for even N and $M = (N+1)/2$ for odd N, is obtained by flipping the first half around M.

Symmetric window shapes are preferred when using a Hanning window in FIR filter design. Periodic window shapes are preferred when using a Hanning window in spectral analysis. This is because the Discrete Fourier Transform assumes periodic extension of the input vector. A periodic Hanning window is obtained by constructing a symmetric window and removing the last sample.

- **Kaiser Window**

$$w(i) = \frac{I_0\left(\beta\sqrt{1-\left(\frac{2i}{N-1}-1\right)^2}\right)}{I_o(\beta)} \qquad 0 \le \beta \le 20$$

Where:

*I*0(x)=Modified zero-order Bessel function.

ß=Constant which specifies a frequency trade-off between the peak height of the side lobe ripples and the width of energy in the main lobe.

- **Klein Window**

$$w(i) = \frac{1}{\pi}\left[\sin\frac{\pi i}{k}\right] + \left(1 - \frac{|i|}{k}\right)\cos\frac{\pi i}{k}$$

When:

$k = \frac{N-1}{2} + 1$ and $-\frac{N-1}{2} \le i \le \frac{N-1}{2}$

Note

$w(i)$ has a minimum amplitude moment and minimizes the truncation error in high resolution computations.

- **Parzen Window**

$$w(i) = 1 - \left|\frac{(i - 0.5(N-1))}{0.5(N+1)}\right|$$

- **Rectangular Window**

$$w(i) = 1 \qquad i = 0, \ldots, N-1$$

- **Welch Window**

$$w(i) = 1 - \left(\frac{(i - 0.5(N-1))}{(0.5(N+1))}\right)^2$$

Using the DNA Advisor Tool

You can use the Device Noise Analysis Advisor (DNA Advisor) to calculate the maximum Fourier frequency for each input waveform. The DNA Advisor analyzes transient or PSS waveforms, recommends the analysis parameters required for an accurate simulation, and highlights analysis methods that are not recommended. The results help you set noise analysis options for both Eldo, AFS and non-Mentor simulators.

Tip

See also the Waveform Calculator harmonicsmeter function.

Procedure

1. Open the DNA Advisor Dialog Box by one of the following methods:
   - Choose **Tools > DNA Advisor**

   or

- Access through the Waveform Calculator:

    i. Click the Waveform Calculator icon, or select it from the **Tools** menu.

    ii. Choose **RF** from the dropdown list.

    iii. Click the **dnaadvisor** button on the calculator keypad.

2. In the Select Hierarchy area, use the **Add Selected Hierarchy** icon to add the currently selected hierarchy. Use the **Clear Hierarchy** icon to remove the hierarchy. You can choose a database, folder or waveform hierarchy as the input.

3. (Optional) In the Select Hierarchy area, you can specify a Search Pattern to consider only waveform names that match the specified pattern. Check the box if you want to perform a Recursive Search and specify a Search Depth (default is All).

4. (Optional) In the Analysis Parameters section, specify the following:

    - Fundamental Frequency — specify the fundamental frequency of the periodic circuit. Default is Automatic.

    - Effort Level — choose the effort level for the Fourier analysis (number of harmonics) from the dropdown list. Default is Standard, corresponding to 505 harmonics.

    - Dynamic Range (dB) — specify the dynamic range required for the circuit. This is the relative tolerance for finding the upper Fourier frequency limit. Default is 60. Range is from 60 to 120 dB.

5. (Optional) In the Plot Results Setup section, check the box to reveal the plot options. Check Plot Top Results and enter a value to specify the number of signals to plot, sorted in order of the maximum difference. Default is 8. You can also check Show Grid Lines.

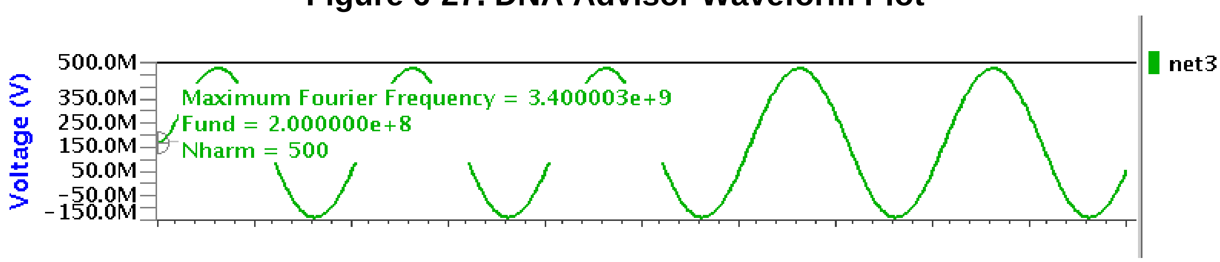
6. Click **Apply**. If any issues occur in the computation, a relevant error or warning message is displayed.

    The results are calculated based on the specified parameters.

Results

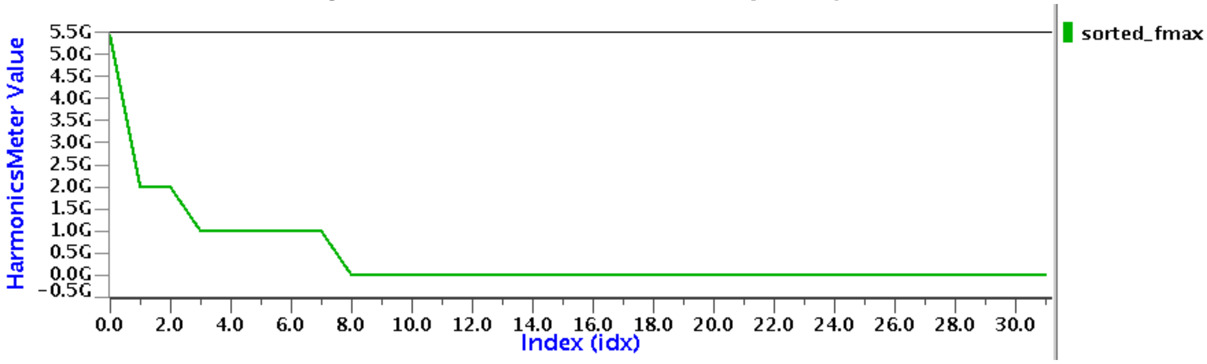
The maximum Fourier frequency is calculated for each input waveform. These values are then sorted from highest to lowest and the waveforms plotted in this order. The following figure shows an example plot of one of the waveforms and its annotations:

Figure 6-27. DNA Advisor Waveform Plot



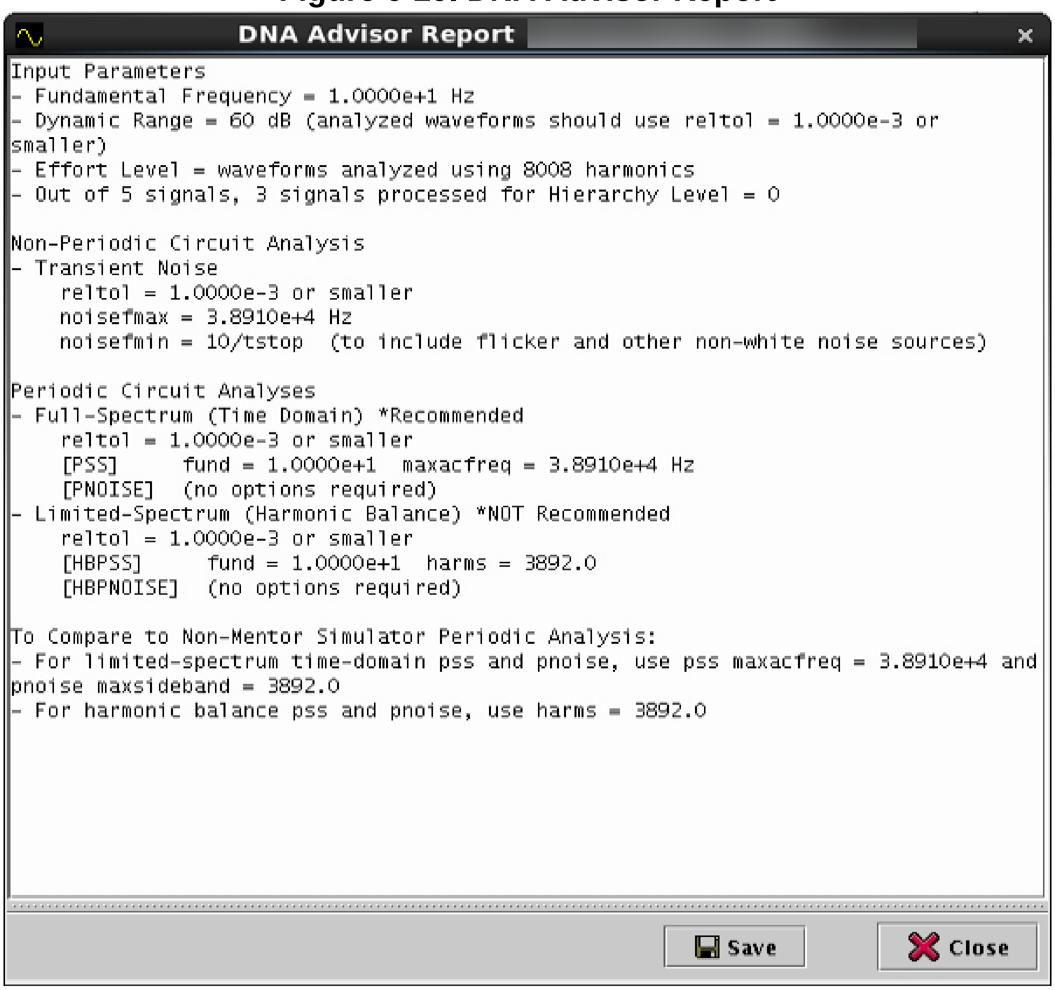
Another plot is generated at the same time, showing the frequency values sorted from highest to lowest (index), as shown in the next figure:

Figure 6-28. DNA Advisor Frequency Plot



A report is also produced, as shown in the following figure, which provides advice and guidelines on settings to obtain or improve the results of the analysis.

Figure 6-29. DNA Advisor Report



Guidelines for interpreting the DNA Advisor reports:

- The reltol recommendation is based on the dynamic range that you specified.
- The noisefmax option sets the maximum noise bandwidth for transient noise analysis. The noisefmax recommendation helps you choose a value that is small enough to yield accurate results, and not so high that time steps become unnecessarily small (which increases simulation time).
- Always set noisefmin to 10/tstop for transient noise analysis.
- The recommended value for fund is either the value that you specified (for transient waveforms) or the calculated value (for pss waveforms).
- The value for maxacfreq determines the number of time points that the simulator takes (the maximum time step). The DNA Advisor's recommendation aims to give you sufficient accuracy in your PSS analysis based on the highest frequency in the output waveforms.
- For a short transient analysis, the circuit may be far from its periodic steady-state. In this situation, the report will be less precise.

Chapter 7
Save and Output Data

This section describes how to save and export simulation results.

Saving and Restoring Graph Windows **404**
Saving Graph Windows 404
Restoring Graph Windows 405

Printing Graph Windows **405**

Exporting Graph Windows **407**
Exporting Graph Windows as a PDF 407
Using a Different PDF Driver 408
Exporting Graph Windows as an Image 408

Saving a Waveform Database **410**
Saving a Single Waveform Database 410
Saving Post-Processed Waveforms 411
Save File Types 411

Saving Multiple Databases **412**

Recovering Database Files **414**
Recovering From Incomplete Simulations 415
Recovering Incomplete Savefiles 416

Outputting a JWDB as an ASCII File **417**
Saving a JWDB as an ASCII File 417
Converting a JWDB to an ASCII File (Batch Mode) 418

Saving and Restoring Graph Windows

EZwave can save graph windows in both SWD and TCL formats.

SWD (*.swd*) and TCL (*.tcl*) formats can store:

- Waveforms associated with the graph window
- Window size, position, axis and background settings
- Complex waveform transformation settings
- Waveform display and cursor settings.
- Optionally, for SWD only, you can save the related waveform database (as a *.wdb* file)

Note

New waveforms computed from the Measurement Tool or using measurement functions from the Waveform Calculator are not currently stored in the TCL or SWD formats.

Saving Graph Windows 404
Restoring Graph Windows 405

Saving Graph Windows

To save your session, you must save each of your graph windows.

Procedure

1. Select the window you wish to save.
2. Choose **File** > **Save**.

   The Save Windows dialog box opens.
3. Select whether you want to save the active window or all windows and whether you want to save the related database.

   Note

   If you have configured your save options to not display this dialog box, the graph window is saved according to the options you have configured.
4. In the text area, type the name you want for the saved window, using *.swd* as the extension, or click the open folder icon to browse for the file you want to save. If the save file already exists, select Overwrite existing file.
5. Click **Save** to save the database and close the dialog box.

6. The system will save the file and also write a new Mentor Graphics database (*.wdb*) file with the same name as the *.swd* file.
7. Do this for each window that you wish to save, issuing a new name for each window, or save all open windows at once.

Related Topics

Save Windows Dialog Box

Restoring Graph Windows

To restore your session, you must open each of your graph windows.

Procedure

1. Choose **File** > **Open**.
2. From the Open dialog box, select the directory you need.
3. Select the Saved Window Databases (*.swd*) filter from the dropdown list.
4. Browse or type the name of the saved window database (*.swd*) file.
5. Click **OK** to open the database and close the dialog box. The system will load the window and the associated *.wdb* file for the window.
6. Do this for each window that you wish to open.

Printing Graph Windows

Before printing multiple graph windows, you can arrange them for better viewing purposes.

Procedure

1. To automatically arrange the graph windows in your workspace, choose one of the following items from the **Window** menu:
    - **Cascade** — Arranges the graph windows in an overlapping fashion proceeding down and to the right in the workspace
    - **Tile Horizontally** — Arranges the graph windows to fill the screen horizontally, fitting several to the screen one above another. If this operation would make some windows too small, they may be stacked on top of each other.
    - **Tile Vertically** — Arranges the graph windows to fill the screen vertically, fitting several to the screen side by side. If this operation would make some windows too small, they may be stacked on top of each other.

- **Tile in a Grid** — Arranges the graph windows to fill the screen in a grid pattern, fitting several to the screen side by side and one above another. If this operation would make some windows too small, they may be stacked on top of each other.

These automatic arrangement options are also available from the toolbar.

You can print the contents of a single graph window, all visible graph windows, or all open graph windows by performing the following:

2. If you want to print a single graph window, make that graph window the active one.
3. Choose **File** > **Print**. This brings up the Print dialog box.
4. Select the options for your print job.
5. Click **Page Setup** to set other options for the print job. On the Page Setup dialog box you can choose to include the database title, and information such as Time and Date (selecting Show Actual Values in Graph Window replaces the Time and Date with actual values). You can also include your User Name and the Machine Name if required. These will be displayed on all windows. You can select a color scheme for the print (the Documentation option is optimized for color printing or exporting). Click **OK** to close the Page Setup dialog box.
6. Click **OK** on the Print dialog box to print the graph windows.

Note

If you are having difficulty using */usr/sbin/lpc* to print from a Linux system, refer to "Linux Printing Issues" on page 1264.

Note

You can print to PDF by choosing **PDF Writer** in the Printer field. See "Exporting Graph Windows" on page 407 for more instructions.

Exporting Graph Windows

You can export a graph windows as a PDF, as described in the following sections.

Exporting Graph Windows as a PDF 407
Using a Different PDF Driver 408
Exporting Graph Windows as an Image 408

Exporting Graph Windows as a PDF

You can export the contents of the active graph window to a PDF file.

Procedure

1. Select the graph window you wish to export.
2. Choose **File** > **Print**.

   The Print dialog box opens.
3. Choose **PDF Writer** in the Printer field.

   The Command field will automatically populate with the required print command for the default PDF tool, and the Use Printing Command option is enabled.
4. In the File field, type the required path and filename. You can use the file browser to navigate to a particular directory and filename. If you don't enter a path, the PDF file is written to the current working directory. The checkbox next to this field should be enabled.
5. If required, modify any paper size and orientation options.
6. Choose which waveforms are to be included; Visible Waveforms or All Waveforms
7. Select whether you want to export only the Active Window, All Windows in Workspace, One per Page, or All Visible Windows, in One Page. If you choose to export all visible windows, the windows must be tiled and visible.
8. By default, any text in the graph window is exported to PDF as a font. You can disable this option by deselecting the Print Text as Font checkbox.
9. Click **Page Setup** to set other options for the PDF output. On the Page Setup dialog box you can choose to include the database title, and information such as Time and Date (selecting Show Actual Values in Graph Window replaces the Time and Date with actual values). You can also include your User Name and the Machine Name if required. These will be displayed on all windows. You can select a color scheme for the image (the Documentation option is optimized for color printing or exporting). Click **OK** to close the Page Setup dialog box.
10. Click **OK** on the Print dialog box to create the PDF.

Using a Different PDF Driver

You can chose a different postscript-to-PDF driver if required.

The ps2pdf tool is usually provided as part of a Linux installation. If preferred, any installed postscript-to-PDF driver may be chosen.

Procedure

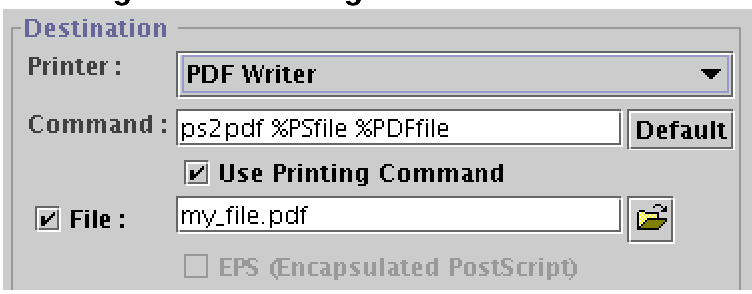
1. Choose **File** > **Print**.

   The Print dialog box opens.

2. Choose **PDF Writer** in the Printer field.

3. Edit the Command field.

Figure 7-1. Editing the Command: Field



- **%PSfile**

  This parameter specifies the temporary*.ps* file.

- **%PDFfile**

  This parameter specifies the destination *.pdf* file.

Exporting Graph Windows as an Image

You can export the contents of the active graph window or all visible graph windows to a JPEG or PNG image file.

Procedure

1. Select the window you wish to export.

2. Choose **File** > **Export**.

   The Export Image dialog box opens.

3. Type the path and filename where you want to save the file. You can use the file browser to navigate to a particular directory and filename. Use an appropriate extension for the filename (*.jpeg*, *.jpg*, or *.png*). If the chosen file already exists, select the Overwrite existing file option to save over the existing file.

4. Select the output format you want from the Save As Type dropdown list.

   Supported output formats are JPEG and PNG.

5. In the options field, select the resolution for the output file. You can select Screen, Printer Low, Printer High or Custom. For Custom, you can specify the required width and, if Lock aspect ratio is unchecked, the height of the image. Additionally, for JPEG images, you can select the image quality by using the slider bar or by entering the percentage into the text field.

6. Select whether you want to export only the Active Window or All Visible Windows. If you are choosing to export all visible windows, the windows must be tiled and visible.

7. Click **Page Setup** to set other options for the image output. On the Page Setup dialog box you can choose to include the database title, and information such as Time and Date (selecting Show Actual Values in Graph Window replaces the Time and Date with actual values). You can also include your User Name and the Machine Name if required. These will be displayed on all windows. You can select a color scheme for the image (the Documentation option is optimized for color printing or exporting). Click **OK** to close the Page Setup dialog box.

8. Click **Export** to export to an image file and close the Export Image dialog box.

Saving a Waveform Database

You need to save the waveform database if you have made changes to a waveform. Waveforms created using the Waveform Calculator also need to be saved.

Saving a Single Waveform Database 410
Saving Post-Processed Waveforms. 411
Save File Types. 411

Saving a Single Waveform Database

After you have made changes to a waveform, you need to save the waveform database. You can also save folders containing scalars, for example noise contributors, to a text file.

Procedure

1. Right-click the database folder and select **Save As**.

   The Save As dialog box opens.

2. Type a name for the file. If the file already exists, you must confirm that you want to overwrite the database.

3. Specify the Save File Types as MGC Database (*.*wdb*), SPICE PWL (*.*sti*), text file (*.*txt*) or comma-separated values (*.*csv*), and GZipped compression (.*gz*) if required.

4. Click **OK** to apply the settings and close the dialog box.

Results

- When saving a database as a SPICE PWL file (.*sti*), only time-based waveforms can be saved. Waveforms that contain voltage or current sources are saved by default. If a database contains waveforms with neither voltage or current sources, those waveforms can still be saved - refer to “Spice Pwl Options” in “Save Data Options” on page 501.

- For databases loaded from *.fsdb* files, only loaded waveforms are saved. Loaded waveforms are waveforms that have been displayed once or waveforms from a hierarchy selected once in the Tree View of the Waveform List Panel. Refer to “Loading .fsdb and .tr0 Files” on page 1272.

- When saving a database in GZipped MGC Database format the original database is not removed. This can be removed manually to reduce the amount of disk space required.

Related Topics

Save As Dialog Box

Saving Post-Processed Waveforms

Waveforms created using the Waveform Calculator will need to be saved.

When using the Waveform Calculator, a *calc* folder is created in the Waveform List panel. This folder contains the created waveforms. A message next to the folder lets you know if the folder contains unsaved changes.

To save the waveforms:

Procedure

1. Right-click the waveform you want to save and select **Rename**.
2. Type the desired name of the waveform.
3. Right-click the *calc* folder and select **Save As**.
4. Navigate to the desired location and type the file name.
5. Specify the file type as either MGC Database (**.wdb*), SPICE PWL (**.sti*), text file (**.txt*) or comma-separated values (**.csv*), and GZipped compression (*.gz*) if required.

   Note
   When compressing a saved file in GZipped format the original file is not removed. This can be removed manually to reduce the amount of disk space required.

Save File Types

In addition to native MGC database files (*.wdb*), the EZwave viewer can output files in several additional formats.

These include:

- Text (*.txt*)

  A text-format database save file includes X and Y values for each point in the waveform separated by tabs, spaces, commas, or a user-chosen delimiter selected from the Save Data Options. You can also select the precision of the saved values from this dialog box.

- Comma-Separated Values (*.csv*)

  A comma-delimited database save file contains value pairs separated by commas. It can be useful for viewing your results in a program such as a spreadsheet. The *.csv* file contains the X and Y values for each point in the waveform, as well as the simulation data (date, title, and so on) and, in the case of multiple-run simulations, the run parameters.

- SPICE PWL (*.sti*)

A SPICE PWL database save file contains value pairs separated by spaces and enclosed in parentheses. A leading plus sign (+) in a line indicates a continuation of the previous line. This file type can be used in Eldo simulation as a stimulus. For more information about the PWL format, refer to the Eldo User's Manual.

Note

When saving a database as a SPICE PWL file (*.sti*), only time-based waveforms can be saved. Waveforms that contain voltage or current sources are saved. If a database contains waveforms with neither voltage or current sources, those waveforms are ignored.

Saving Multiple Databases

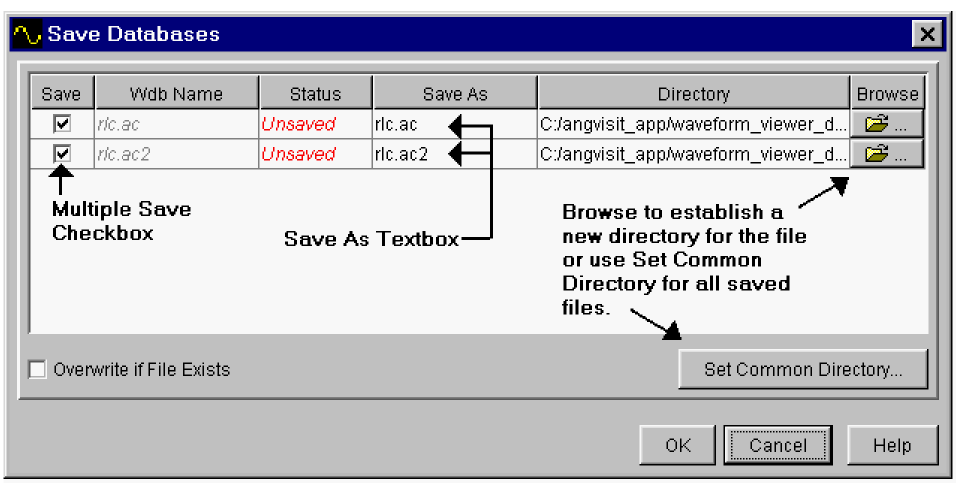
If you have unsaved data, you have the capability to save the data without having to perform separate save operations.

To save multiple databases, perform the following steps:

Procedure

1. Right-click the Waveform List panel to activate the Waveform List Popup Menu.
2. Select the **Save All Databases** menu item.

   The Save Databases dialog box opens.



   The Save Databases dialog box lists all currently unsaved databases or modified databases in the Waveform List panel.

3. Click the checkboxes to select the databases you wish to save. The Wdb Name column shows the current name of the database. The status column displays the status of *Unsaved*, *Saving* or *Saved*.

4. Use the Save As text box to type the new name for the database. If the file already exists, you must confirm the desire to overwrite the database. If you do not wish to be warned about existing databases of the same name, check the Overwrite if File Exists checkbox in the lower left corner of the dialog box.
5. Click **OK** to apply the settings and close the dialog box, or click **Cancel** to close the dialog box without saving any databases.

Recovering Database Files

The application provides two different methods of recovering database files. The first method described here enables you to recover incomplete database files, produced when a simulator that the EZwave viewer was connected to does not exit cleanly. The second method recovers databases that are saved incompletely, such as might be the case during a disk write error or a network shutdown.

Recovering From Incomplete Simulations 415
Recovering Incomplete Savefiles 416

Recovering From Incomplete Simulations

You can use the **recoverjwdb** utility to recover simulation data and save it to a **jwdb** file in the event of an unclean simulator exit.

Usage

During simulation, the simulator may create an information file called **jwdb***PortHostname* containing information about the simulation process. If the EZwave viewer is still running, this utility can retrieve simulation data from memory or from this spill file.

To invoke **recoverjwdb**, use the following syntax:

```
recoverjwdb info_file save_path
```

Arguments

- ***info_file***

  The simulator information file.

- ***save_path***

  The path to the required save location for the file.

Recovering Incomplete Savefiles

You can use the temporary recovery file created by EZwave to recover the databases if required. While the EZwave viewer is running, a temporary file named *<file>.wdb_recoveryKeyFile* is created. This temporary file contains the information needed to recover the databases, and is removed when the EZwave viewer exits normally. If an abnormal exit condition occurs, this recovery key file will remain and can be used to recover any unsaved data from the simulation.

Usage

The following command enables you to use this temporary file to recover incomplete savefiles in this event:

```
ezwave -recovery file.wdb file.wdb_recoveryKeyFile output_file
```

where *file* is the original file you were working with.

Note

If disk space runs out during a simulation, the EZwave viewer will display a notice that this has occurred and will prompt you for an alternate location to save the remainder of the database file. The first part of the savefile is saved in the original disk location as *file_recovery_part1*, and the remainder are saved in whatever location you specify as *file_recovery_part2*. The file is saved in two parts even if you remove files in the original disk partition and elect to save the remainder of the database there.

To recover the incomplete savefile in this case, use the following command format:

```
ezwave -recovery file_recovery_part1 file_recovery_part2 output_file
```

where *file* is the database savefile you were using.

Arguments

- **-norecovery**

  If the recovery key file does not produce satisfactory results, the recovery mechanism can be disabled by using the **-norecovery** command line option.

Outputting a JWDB as an ASCII File

This section describes how you can output a JWDB to an ASCII file, either from the GUI or using the *jwdbtoasc* batch utility.

Saving a JWDB as an ASCII File 417
Converting a JWDB to an ASCII File (Batch Mode) 418

Saving a JWDB as an ASCII File

You can save a JWDB as an ASCII file.

Procedure

1. In the Waveform List panel, right-click the database name and select **Save As**.

   The Save As dialog box opens.

2. Type the name for the database. If the file already exists, you must confirm the desire to overwrite the database.

3. Select an ASCII file type from the Files of Type dropdown list.

   EZwave creates an ASCII file that contains all the signals.

Related Topics

Save As Dialog Box

Save File Types

Converting a JWDB to an ASCII File (Batch Mode)

You can convert a JWDB savefile to ASCII in batch mode (that is, without having to open the file using the EZwave viewer and save it as ASCII) by using the *jwdbtoasc* utility. All signals, both analog and digital, can be converted using this tool. You can also specify a Tcl script to run for post-processing before the converted waveforms are saved to the ASCII file.

Usage

The syntax for this tool is as follows:

```
jwdbtoasc {-i filename.wdb} [-o filename.txt [-do filename.tcl]]
   [-precision value]
```

Arguments

- ***-i***

  Specifies the input JWDB file name.

- ***-o***

  Specifies the output ASCII file name. If the output file is not specified, the new file name is the same as the input file name but with the *.wdb* extension changed to *.txt*.

- ***-do***

  Specifies the post-processing Tcl script name.

- ***-precision***

  Specifies the level of precision. *value* should be between 1 and 16, inclusively. This argument also accepts full as the *value*, setting it to the full 16 digits.

Chapter 8
Dialog Box and Field Reference

This section provides detailed descriptions of the EZwave windows and dialog boxes.

Add Clock Dialog Box 422
Analog to Digital Conversion Dialog Box 423
Annotation Properties Dialog Box 424
Axis Properties Dialog Box 427
Comparison Options Dialog Box 429
General Options Tab 430
Comparison Methods Tab. 433
Tolerances Tab 435
Conversions Tab 437
Create Bus Dialog Box. 439
Current Analysis Dialog Box 441
Cursor Properties Dialog Box. 445
Data Values Dialog Box. 446
Database Properties Dialog Box 448
Digital to Analog Dialog Box. 450
Event Search Tool Dialog Box 453
Edit Digital Transformation Dialog Box 455
Eye Diagram Tool Dialog Box 457
Eye Diagram Tool - Settings Tab 458
Eye Diagram Tool - Measurement Results Tab 465
Eye Mask Dialog Box. 470
Extract Outputs From Database Dialog Box 472
EZwave Display Preference Options 474
EZwave Display Preferences Dialog Box 476
Automatic Reload Options 478
Cursor Options 480
Data Format Options. 482
Foreign Databases Options 484
General Options 485
Grid Options 488
Layout Options 490
Look and Feel Options 491
Mouse Pointer Options 492

Multiple Run Options 493
Pick Points Options. 495
RF Options 498
Row Options 499
Save Data Options 501
Save Window Options 503
Text Annotation Options 504
Transformation Options 505
Waveform Options 507
Waveform Compare Options 508
Waveform List Options 509
Workspace Options. 511
CDF Plot Options 512
CDF Measures Options. 514
CDF Legend Options 516
Axis Title Options. 518
Axis Values Options 519
Axis Values (Smith Chart) Options 520
Calculator Entry Options 521
Cursor/Marker Options 522
Eye Mask Options. 524
Grid Options (Fonts and Colors) 525
Header Text Options 526
Histogram Options 527
Measurement Annotation Options 528
Pick Points Options (Fonts and Colors) 529
Row Title Options. 530
Text Annotation Options (Fonts and Colors) 531
Waveform Colors Options 532
Waveform Display Options 533
Waveform Name Options 534
Waveform Selection Options 536
Window Background Options 537
Zero-Level Line Options 538
Histogram Plot Options 539
Histogram Measures Options 541
Histogram Legend Options. 543
Waveform Calculator Calculation Options. 545
Waveform Calculator General Options 547
Waveform Calculator View Options 549
Filter Dialog Box 550
Find Tool Dialog Box. 553
Jitter Tool Dialog Box 555
Measurement Tool Dialog Box 560

Parameter Analyzer Dialog Boxes **563**
Parameter Analyzer Tool Dialog Box 564
Filters Setup Dialog Box. 567
Parameter Table Dialog Box **568**
Pick Points Dialog Box **570**
Power Analysis Dialog Box **576**
Run Filter Dialog Box **580**
Save As Dialog Box **583**
Save Windows Dialog Box **586**
Select Hierarchy Dialog Box **587**
Select Waveforms Dialog Box **589**
Shortcuts Manager Dialog Box **590**
Update Waveforms Dialog Box **592**
Waveform Names Display Dialog Box **594**
Waveform Properties Dialog Box (For Analog Waveforms) **596**
Waveform Properties Dialog Box (Digital Waveforms) **599**
Waveform Calculator Dialog Boxes **601**
Auto Correlation Dialog Box 602
Chirp Transform Dialog Box 605
Constellation Diagram Dialog Box. 609
Convolution Dialog Box 611
Cross Correlation Dialog Box. 613
DNA Advisor Dialog Box 615
Error Vector Magnitude and Bit Error Rate Dialog Box 617
Eye Diagram Dialog Box 619
Fast Fourier Transform Tool Dialog Box 621
Harmonic Distortion Dialog Box 626
Histogram Dialog Box 628
Inverse Fast Fourier Transform Dialog Box. 630
Phase Noise Dialog Box 632
Power Spectral Density Dialog Box 634
PSS Residue Dialog Box 637
Signal to Noise Ratio Dialog Box. 639
Spectrum Measurement Tool Dialog Box 642
Tcl File Viewer Dialog Box 648
Top Noise Dialog Box 650
Windowing Transform Dialog Box 652

Add Clock Dialog Box

To access: Choose**Tools** > **Waveform Compare** > **Options** and select the **Comparison Methods** tab (see “Comparison Methods Tab” on page 433), then select Clocked Comparison and click the **Clocks** button. Then click **Add** or **Modify**.

Use this dialog box to define a clock that can then be used for clocked-mode comparisons. The clock can be based on an existing waveform, or on an ideal signal.

Objects

Table 8-1. Add Clock Dialog Box Contents

Field	Description
Clock Name	Specifies an identifying name for the clock.
Delay Signal Offset	Specifies a delay in the source signal for comparison. The default is 0 (no delay).
Based on	
Ideal Clock	Specifies an ideal signal based on either a frequency or a period/duty cycle definition. • **Frequency**—specify the frequency of the clock signal. • **Period**—specify the period of the clock signal. Duty Cycle specifies the percentage of time per period that the signal is active.
Waveform	Specifies the waveform upon which to base the clock signal. The Add Selected Waveform icon adds the currently selected waveform in the active graph window or waveform list. The Clear Waveform List icon removes all waveforms in the list.
Compare Strobe Edge	
Rising	Specifies that the rising edge of the specified signal or waveform should be used. Default.
Falling	Specifies that the falling edge of the specified signal or waveform should be used.
Both	Specifies that both the rising and falling edges of the specified signal or waveform should be used.

Analog to Digital Conversion Dialog Box

To access: Right-click an analog waveform and select **Analog to Digital Conversion**.

Use this dialog box to convert analog waveforms to digital ones and specify options for the conversion.

Objects

Table 8-2. Analog to Digital Conversion Dialog Box Contents

Field	Description
Digitized Waveform(s)	
Name	The name of the input analog waveform
Plot Option	Controls how to display the digital waveform: • Stacked Plots the resulting digital waveform in a new row, at the end of the graph window. • Overlaid Plots the resulting digital waveform in the same graph as the input waveform (in the same row).
Plot Result Waveform(s)	Plots the resulting digital waveform using the plot option specified.
Transformation Setup	
Single Threshold	Specifies a threshold for defining the digitization of the input waveform. Values that fall below this threshold are converted to 0, while values above it are converted to 1. The default value is the middle of the waveform: *(min(wave) + max(wave))/2*
Two Thresholds	Converts the input waveform to a 3 state waveform (0, X and 1), defined using the High Threshold and Low Threshold fields. Values that fall between these thresholds are converted to “X”.
High Threshold	Values above high threshold are converted to 1. The default value is 80% of the input waveform amplitude: *min(wave) + 0.8*(max(wave) - min(wave))*
Low Threshold	Values below this threshold are converted to 0. The default value is 20% of the input waveform amplitude: *min(wave) + 0.2*(max(wave) - min(wave))*.

Usage Notes

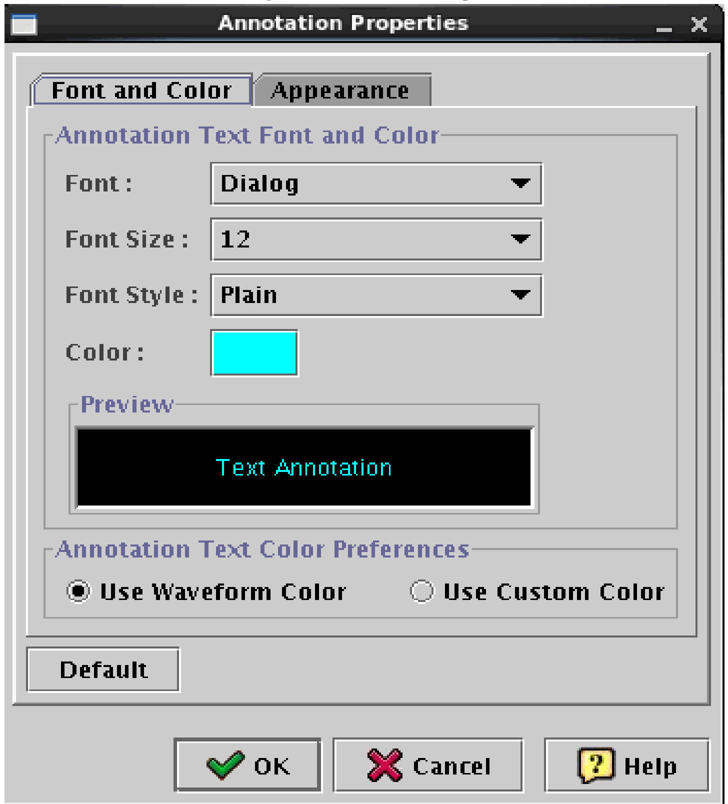
The default values for threshold in each case (80%, 50% and 20%) are not customizable.

Annotation Properties Dialog Box

To access: Right-click a text annotation and select **Properties**.

Used to define custom display properties for individual text annotations. The controls on this dialog box are organized on two tabs, **Font and Color** and **Appearance**.

Figure 8-1. Annotation Properties Dialog Box - Font and Color Tab



Objects

Table 8-3. Annotation Properties Dialog Box - Font and Color Tab Contents

Field	Description
Font	Specifies the font to use for the current text annotation.
Font Size	Specifies the point size of the font to use for the current text annotation.
Font Style	Specifies the style of the font to use for the current text annotation, from Plain, Bold, Italic and Bold/Italic.
Color	Opens a Color Selection dialog box to choose the color to use for the current text annotation. Click **OK** to accept the color and return to the dialog box. This control is not available when the Monochrome theme is active.

Table 8-3. Annotation Properties Dialog Box - Font and Color Tab Contents

Field	Description
Preview	Displays a sample of what the text will look like with the current selections.
Waveform Annotation Color	Controls the default color of text annotations: • Use Waveform Color Uses the same color as the waveform. Available only for annotations that are attached to waveforms. • Use Custom Color Text annotations will use the custom color defined on this dialog box.
Default	Restores the font and color of the current text annotation to the default settings.

- The Annotation Properties - Appearance dialog box is shown here:

Figure 8-2. Annotation Properties Dialog Box - Appearance Tab

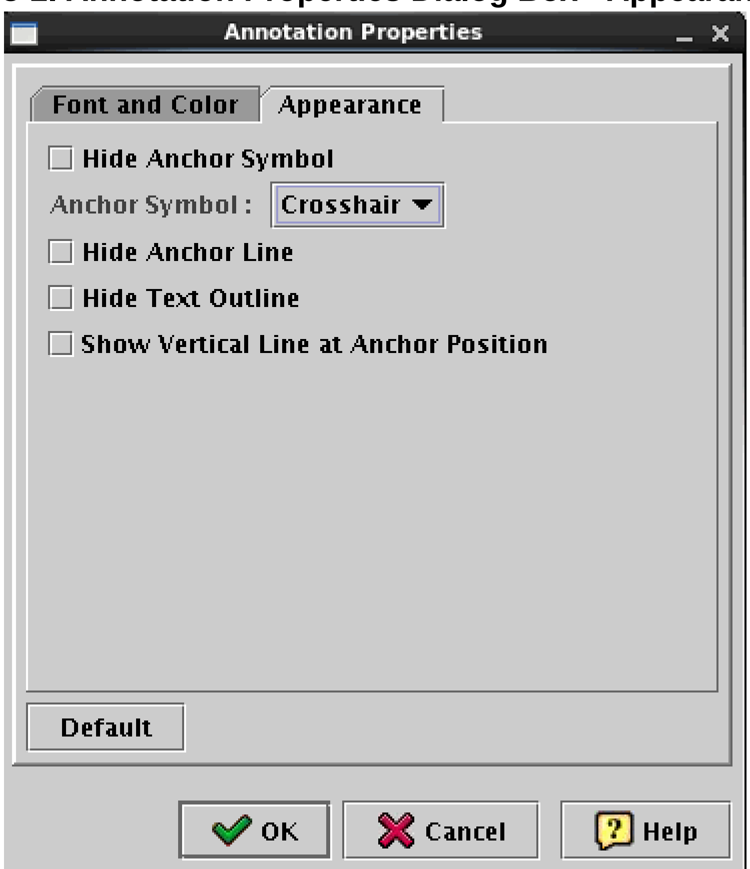


Table 8-4. Annotation Properties Dialog Box - Appearance Tab Contents

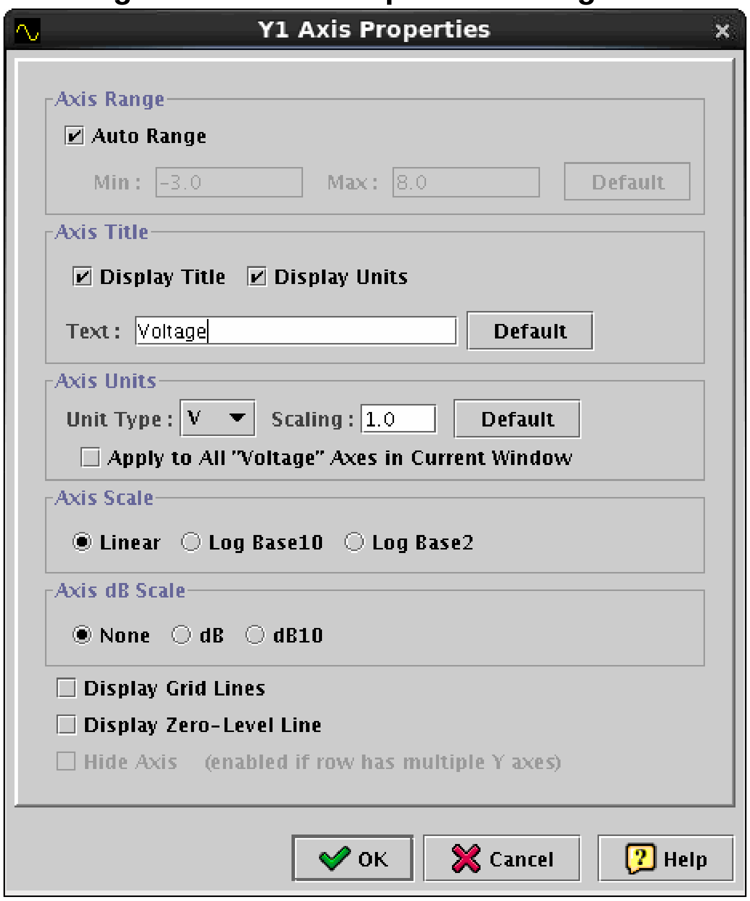
Field	Description
Hide Anchor Symbol	Hides the symbol at the anchor point between the annotation and the waveform (if the annotation is attached to a waveform).
Anchor Symbol	Selects the kind of symbol used at the anchor point of the text annotation (if the annotation is attached to a waveform). Can be either Crosshair or Dot.
Hide Anchor Line	Hides the line attaching the annotation to the waveform (if the annotation is attached to a waveform).
Hide Text Outline	Hides the text box on the current annotation.
Show Vertical Line at Anchor Position	Specifies that a vertical line is displayed across the waveform at the selected position. Available only for annotations that are attached to waveforms.
Default	Restores the appearance of the current text annotation to the default settings.

Axis Properties Dialog Box

To access: Right-click either the x or y axis to display the Axis Popup Menu and select the **Properties** menu item. Alternatively, you can select multiple plotted waveforms and right-click to display the Waveform Popup Menu popup menu and select **Y-Axis Properties** to display the Axis Properties dialog box. This is populated with the y-axis properties of the first waveform in the selection. OK applies the y-axis properties to all selected waveforms.

Use this dialog box to specify the properties of the current axis.

Figure 8-3. Axis Properties Dialog Box



Objects

Table 8-5. Axis Properties Dialog Box Contents

Field	Description
Axis Range	
Auto Range	Specifies that the axis show all of the data.
Min/Max	Specifies the exact minimum and maximum values to be viewed on the axis. Available when Auto Range is not checked.
Axis Title	

Table 8-5. Axis Properties Dialog Box Contents (cont.)

Field	**Description**
Display Title	Specifies that the title is shown on the axis.
Display Units	Specifies that the units are displayed.
Text	A custom text string that specifies the axis title.
Axis Units	
Unit Type	Specifies the available units for the axis. An example of use is a waveform with data that repeats at a regular interval. For example, if a waveform has data that repeats every 11 nanoseconds (11ns, 22ns, 33ns...), then setting the scaling value to “11.0” would result in axis labels of 1, 2, 3.
Scaling	Specifies the scale factor applied to the axis labels.
Default	Sets both the Unit Type and Scaling values to the current settings controlled through the Data Format Options.
Apply to All “Voltage” Axes in Current Window	Specifies that these unit settings are applied to a voltage axes in the current window.
Axis Scale	
	Specifies how the axis is scaled from: • Linear • Log Base 10 • Log Base 2
Axis dB Scale	
	Specifies how the y axis is scaled from: • None • dB - changes the y axis to decibels ($20\log_{10}$ for amplitude ratios). • dB10 - changes the y axis to decibels ($10\log_{10}$ for power ratios).
Display Grid Lines	Specifies that grid lines are displayed.
Display Zero Level Line	Specifies that the zero level line is displayed.
Hide Axis	Specifies that the axis is hidden. Available if the row has multiple y-axes.

Comparison Options Dialog Box

Use this dialog box to specify options for the Waveform Compare Tool.

It contains the following tabs:

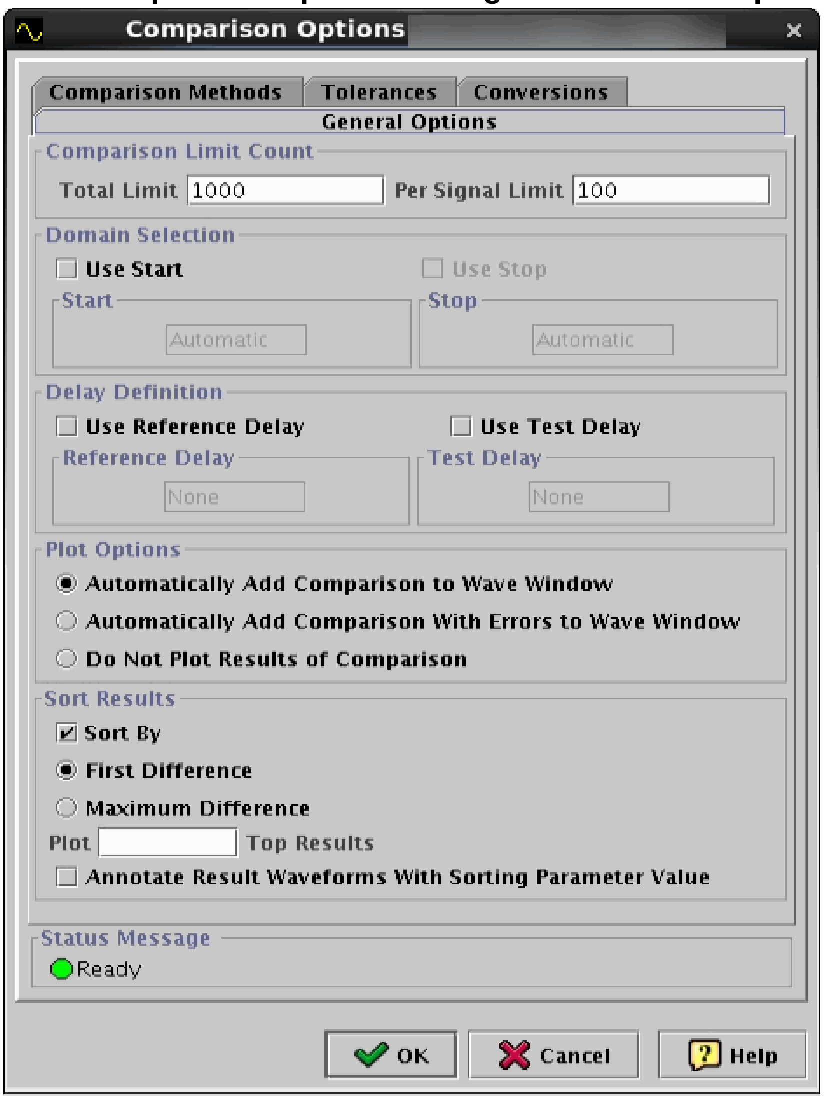
General Options Tab 430
Comparison Methods Tab. 433
Tolerances Tab 435
Conversions Tab 437

General Options Tab

To access: Choose **Tools** > **Waveform Compare** > **Options** from the main menu. The **General Options** tab is selected by default.

Use this tab to set global options for Waveform Comparison, including settings for the maximum number of differences allowed before the comparison terminates, and settings that determine which comparison results to display.

Figure 8-4. Comparison Options Dialog Box - General Options Tab



Objects

Table 8-6. Comparison Options Dialog Box - General Options Tab Contents

Field	Description
Comparison Limit Count	

Table 8-6. Comparison Options Dialog Box - General Options Tab Contents

Field	Description
Total Limit	Specifies the maximum number of differences allowed before the comparison terminates.
Per Signal Limit	Specifies the maximum number of differences per signal allowed before the comparison terminates.
Domain Selection	
Use Start	If you want to begin the comparison from a point other than the start (time or frequency) of the selected waves, select this option and type a new value at which to begin, in the data unit specified in the Data Format Options.
Use Stop	If you want to end the comparison from a point other than the end (time or frequency) of the selected waves, select this option and type a new value at which to finish, in the data unit specified in the Data Format Options.
Delay Definition	
Use Reference Delay	If you want to specify a delay to shift all added waveforms from the reference dataset, select this option and type the delay time in the field.
Use Test Delay	If you want to specify a delay to shift all added waveforms from the test dataset, select this option and type the delay time into the field.
Plot Options	
Automatically Add Comparison to Wave Window	Adds the comparison results to the current wave window, including both correct (waveforms with no differences) and erroneous (waveforms with differences) waveforms.
Automatically Add Comparison With Errors to Wave Window	Adds only comparison results that contain errors (waveforms with differences) to the current wave window.
Do Not Plot Results of Comparison	This option does not add the comparison results to the current wave window.
Sort Results	
Sort By	Click to enable the sort options described here.
First Difference	This option finds the first time when any signal in one run begins to differ from the corresponding signal in another run, and plots the signals in sorted order starting from the highest difference.
Maximum Difference	This option finds the first time when any signal in one run differs most from the corresponding signal in another run, and plots the signals in sorted order starting from the highest difference.
Plot Top Results	This option specifies the number of top results to be plotted.

Table 8-6. Comparison Options Dialog Box - General Options Tab Contents

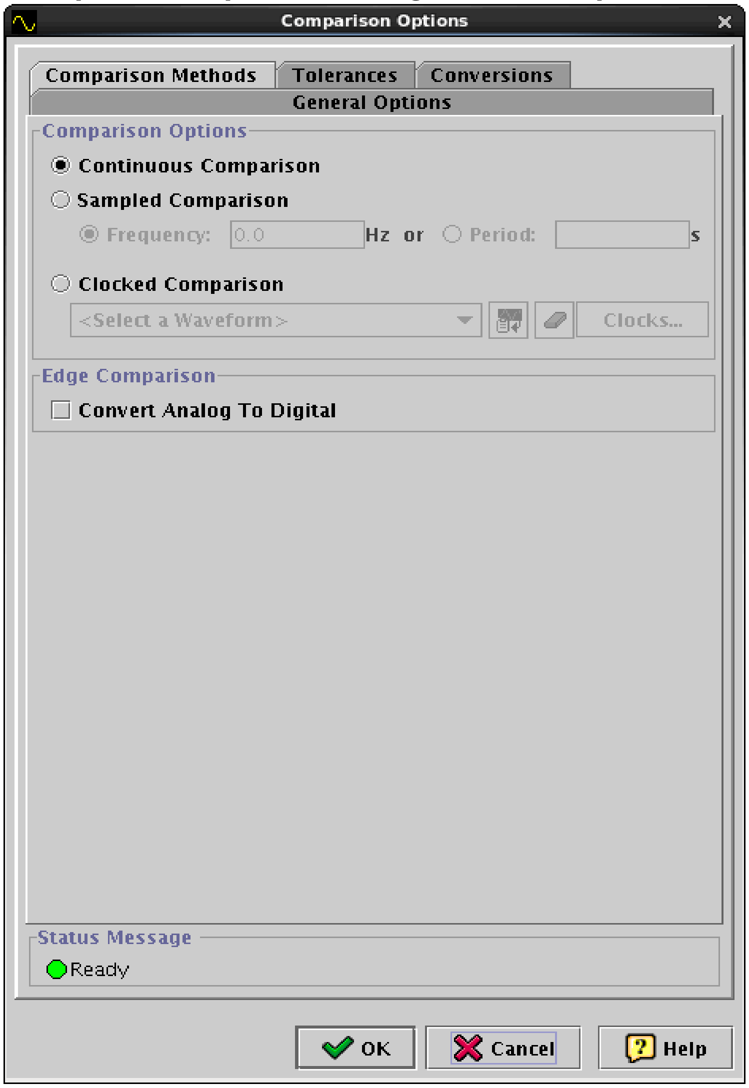
Field	Description
Annotate Result Waveforms With Sorting Parameter	If checked, the sorting parameter value will be plotted as annotation (or other marker, for example delta) over the plotted comparison results (waveforms).

Comparison Methods Tab

To access: Choose **Tools** > **Waveform Compare** > **Options** from the main menu to display the Comparison Options dialog box, and click the **Comparison Methods** tab.

Used to specify how the Waveform Compare Tool compares digital waveforms.

Figure 8-5. Comparison Options Dialog Box - Comparison Methods Tab



Objects

Table 8-7. Comparison Options Dialog Box - Comparison Methods Tab Contents

Field	Description
Comparison Options	
Continuous Comparison	Specifies that test signals are compared to reference signals at each transition of the reference. The default option.

Table 8-7. Comparison Options Dialog Box - Comparison Methods Tab Contents (cont.)

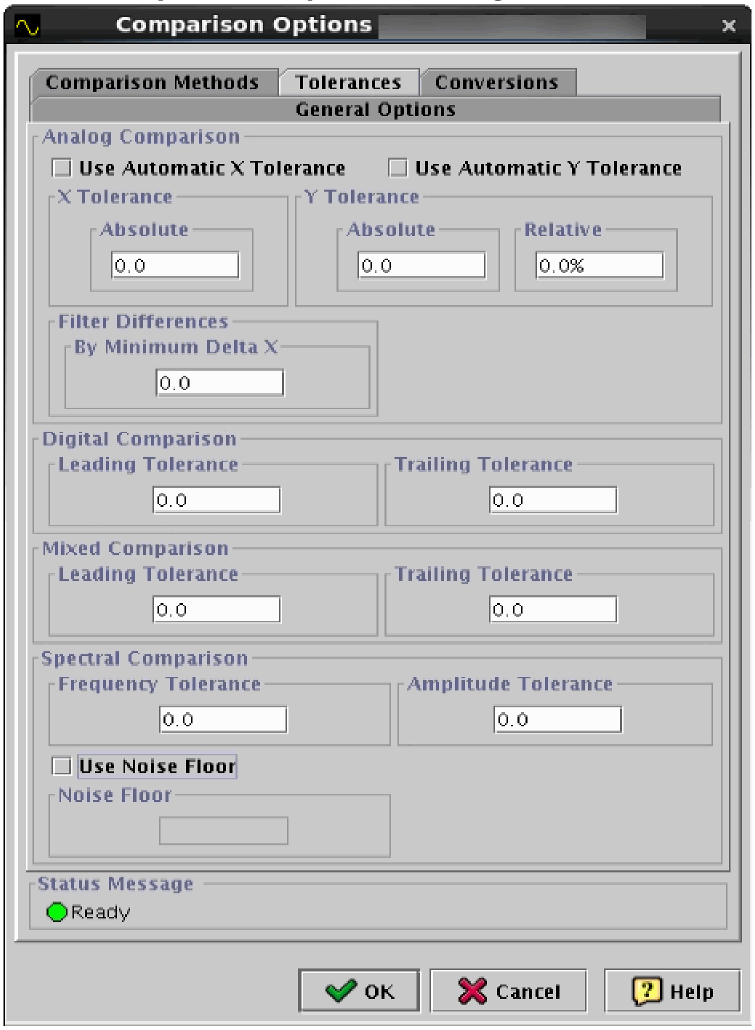
Field	Description
Sampled Comparison	Offers the possibility to define either a frequency or a period to use for the comparison session. A comparison between a reference and a test waveform is made for each point sampled.
Clocked Comparison	Specifies that a clock signal is defined to use as a trigger waveform for the comparison. Either: • Select waveform to use as the clock signal from the active graph window and click the Add Selected Waveforms icon . • Click the **Clocks** icon to access the Clocks dialog box, and from there click **Add** to open the Add Clock Dialog Box and configure a new clock signal.
Convert Analog To Digital	Forces a “digital” edge comparison where both, or one, source and reference signals are analog. An Analog to Digital conversion is applied automatically during the comparison. **Tip:** This is useful, for example, where the waveforms represent a clock but have different amplitudes. You can compare the delay between the waveforms. Refer to additional settings for “Analog to Digital Conversion” on page 257 and “Digital Comparison” on page 254.

Tolerances Tab

To access: Choose **Tools** > **Waveform Compare** > **Options** from the main menu to display the Comparison Options dialog box, and click the **Tolerances** tab.

Used to specify tolerances for analog, digital, mixed-signal and spectral waveform comparison.

Figure 8-6. Comparison Options Dialog Box - Tolerances Tab



Objects

Table 8-8. Comparison Options Dialog Box - Tolerances Tab Contents

Field	Description
Analog Comparison	
Use Automatic X Tolerance	Specifies that an automatic tolerance is used for the X tolerance.
Use Automatic Y Tolerance	Specifies that an automatic tolerances is used for the Y tolerance.

Table 8-8. Comparison Options Dialog Box - Tolerances Tab Contents (cont.)

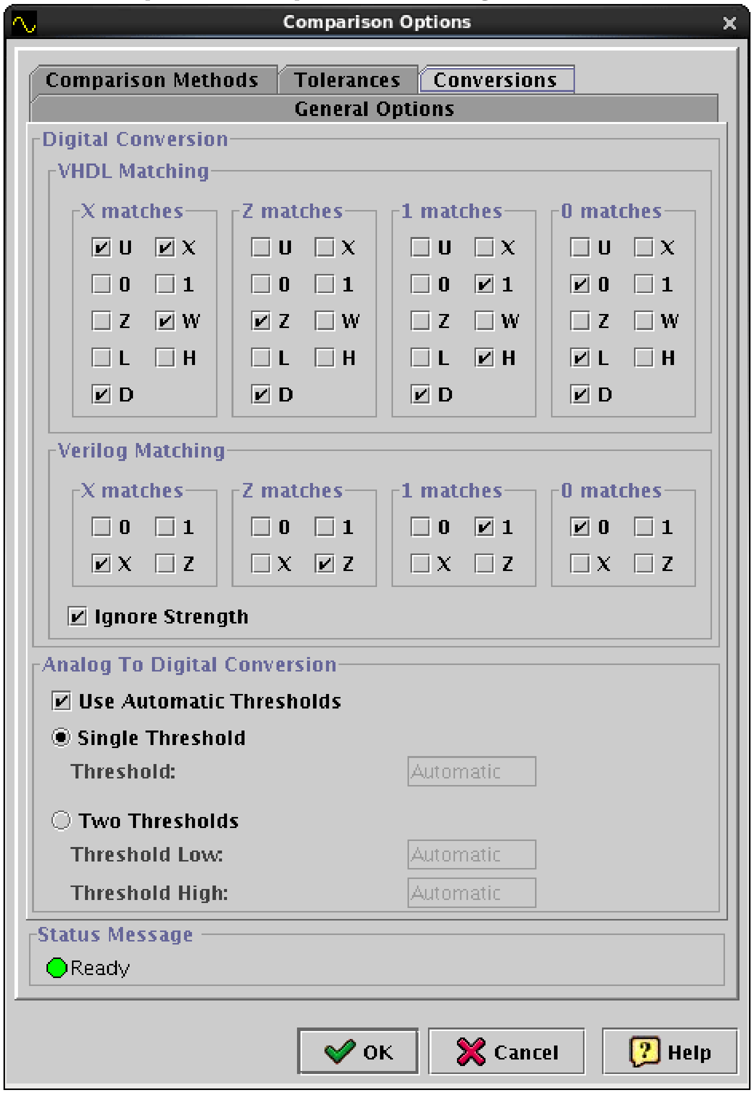
Field	Description
X Tolerance	Specifies the X tolerance. Available if Use Automatic X Tolerance is not selected.
Y Tolerance	Specifies the Y tolerance. Available if Use Automatic Y Tolerance is not selected. You can specify both Absolute and Relative Y Tolerance at the same time. The Y Tolerance used to calculate the error tube is then: `absolute_y_tolerance + (relative_y_tolerance * y_value_of_the_sampling_point)`
Filter Differences	Specifies the minimum delta X tolerance (usually time). Ignores differences that are less than the specified value. Default = 0.
Digital Comparison	
Leading Tolerance	Specifies the maximum time a test signal edge is allowed to lead a reference edge in an asynchronous comparison of digital signals. Default = 0.
Trailing Tolerance	Specifies the maximum time a test signal edge is allowed to trail a reference edge in an asynchronous comparison of digital signals. Default = 0.
Mixed Comparison	
Leading Tolerance	Specifies the maximum time a test signal edge is allowed to lead a reference edge in an asynchronous comparison of both digital and analog signals. Default = 0.
Trailing Tolerance	Specifies the maximum time a test signal edge is allowed to trail a reference edge in an asynchronous comparison of digital and analog signals. Default = 0.
Spectral Comparison	
Frequency Tolerance	Specifies the frequency tolerance. Default = 0.
Amplitude Tolerance	Specifies the amplitude tolerance. Default = 0.
Use Noise Floor	Specifies that the noise floor level is considered in the comparison.
Noise Floor	Specifies the noise floor level. If the Use Noise Floor option is enabled, errors that occur below the specified noise floor amplitude are ignored in the comparison.

Conversions Tab

To access: Choose **Tools** > **Waveform Compare** > **Options** from the main menu to display the Comparison Options dialog box, and click the **Conversions** tab.

Used to specify options for analog and mixed-signal comparisons, including settings for signal value matching rules.

Figure 8-7. Comparison Options Dialog Box - Conversions Tab



Objects

Table 8-9. Comparison Options Dialog Box - Conversions Tab Contents

Field	Description
Digital Conversion	

Table 8-9. Comparison Options Dialog Box - Conversions Tab Contents (cont.)

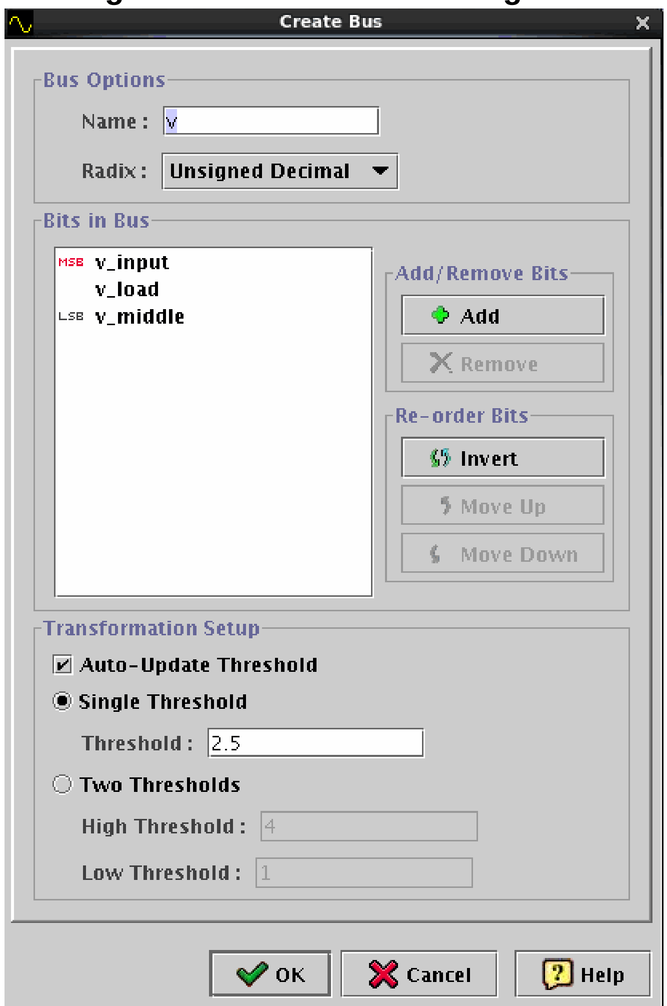
Field	Description
VHDL Matching	Specifies the VHDL signal value matching rules. For more information on these options, refer to the Mapping Data Types section of the *Questa SIM User's Manual*.
Verilog Matching	Specifies the Verilog signal value matching rules. For more information on these options, refer to the Mapping Data Types section of the *Questa SIM User's Manual*. **Note:** The Ignore Strength checkbox is not currently implemented, and has no effect.
Analog to Digital Conversion Options	
Use Automatic Thresholds	Specifies that automatic thresholds are used. Available options: • **Single Threshold** — Specifies a single threshold. The text window specifies the desired threshold value if Use Automatic Thresholds is not selected. • **Two Thresholds** — Specifies a low and high threshold. The text windows specify the desired threshold values if Use Automatic Thresholds is not selected.

Create Bus Dialog Box

To access: Choose **Tools** > **Create Bus** from the main menu.

Use this dialog box to create your own bus from selected digital waveforms (bits).

Figure 8-8. Create Bus Dialog Box



Objects

Table 8-10. Create Bus Dialog Box Contents

Field	Description
Bus Options	
Name	The name of the bus. This field is populated with a proposed name, deduced from the longest prefix common to all bits of the bus.

Table 8-10. Create Bus Dialog Box Contents (cont.)

Field	Description
Radix	Specifies the radix as one of the following: • One’s Complement • Two's Complement • Signed Magnitude • Binary • Hexadecimal • Octal • Unsigned Decimal • Ascii • Fixed Point • Gray Code to Decimal • To Gray Code • Unary (Thermo Code)
Bits in Bus	Lists all of the selected waveform names. The waveform selected first is used as the most significant bit (MSB) of the bus, and the waveform selected last is used as the least significant bit (LSB).
Transformation Setup	
Auto-Update Threshold	Specifies that the threshold calculation is automatically updated when the bit order is modified. Selected by default.
Threshold	If the bus is an analog or hybrid bus, the analog signals are automatically transformed to digital. This option specifies the thresholds to digitize the input waveform(s) as one of the following: • Single Threshold • Two Thresholds Specifies the desired threshold values with each.

Related Topics

Creating a Bus

Current Analysis Dialog Box

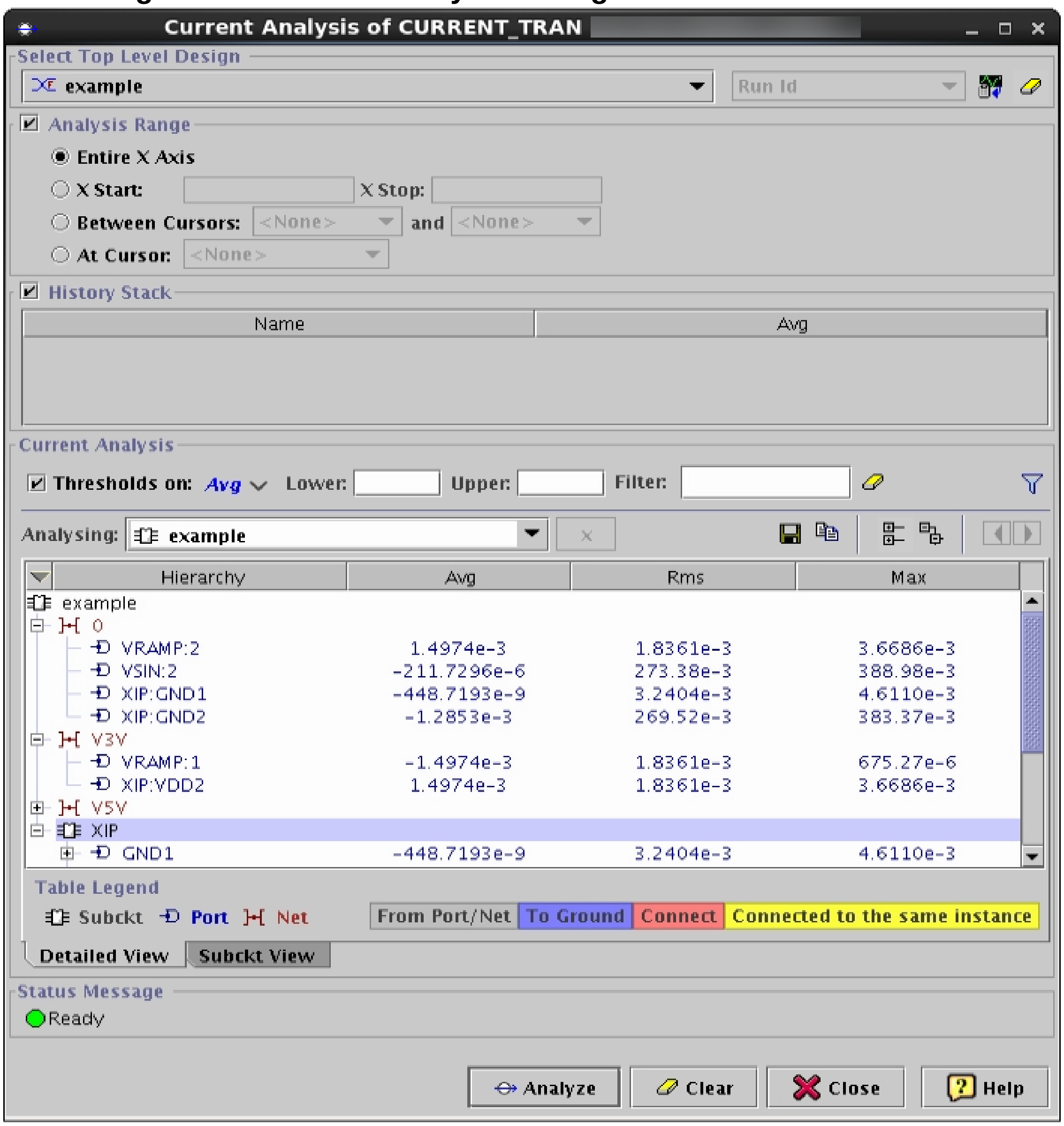
To access: Expand a set of current analysis results (named CURRENT_<analysis_type>) in the Waveform List Panel, right-click the folder at the required level of hierarchy and choose **Current Analysis**. Alternatively, choose **Tools > Current Analysis**.

Used to analyze current consumption of a circuit and its components over time.

Tip

Refer to "Analyzing Current Consumption" on page 262.

Figure 8-9. Current Analysis Dialog Box - Detailed View Tab



Objects

Table 8-11. Current Analysis Dialog Box Contents

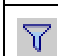
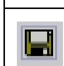
Field	Description
Select Top Level Design	Limits the results to those listed below the selected Top Level Design.
Run Id	For compound waveforms, specify the run.
Analysis Range	
Entire X Axis	Specifies the entire input waveform range to be analyzed.
X Start	Specifies the start of the range for the input waveform to be analyzed. Results outside of this range will not be listed.
X End	Specifies the end of the range for the input waveform to be analyzed. Results outside of this range will not be listed.
Between Cursors	Specifies the range between two cursors to analyze the input waveform. You can drag either cursor to dynamically analyze results for different ranges.
At Cursor	Specifies a single point to analyze the input waveform. You can drag the cursor to dynamically analyze results at different points.
History Stack	
Name	Displays the full hierarchical path to the port.
Avg	Displays the default measure performed on the port on the current time window.
Current Analysis	
Thresholds on: Avg | RMS | Max | Min | Integ | Sum | Peak to Peak	If checked, filters the Hierarchy column using current thresholds on the value of the waveform in the range given by the Lower and/or Upper values.
Lower	Specifies a minimum current value. Results below the specified threshold will not be listed.
Upper	Specifies a maximum current value. Results above the specified threshold will not be listed.
Filter	Specifies the text used to filter the Hierarchy column.
	Click to clear the filter.
	Click to apply the filter.
Analysing	Choose the results to analyze (**Detailed View** tab only).
	Saves the results as a *.csv* file.

Table 8-11. Current Analysis Dialog Box Contents (cont.)

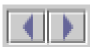
Field	Description
	Copies the results to the clipboard.
	Collapses the hierarchy.
	Expands the hierarchy.
	Steps through previous or next connections.
	Menu specifies which results to display: Avg | RMS | Max | Min | Integ | Sum | Peak to Peak
Hierarchy	Displays the Subckt, Port and Net hierarchy.
Avg	Displays the average current for the port.
Min	Displays the minimum current for the port.
Max	Displays the maximum current for the port.
Integ	Displays the integrated current for the port (continuous integral calculation).
Sum	Displays the total current for the port (discrete integral calculation, summing only Y values of simulation datapoints).
Rms	Displays the RMS current for the port.
Peak to peak	Displays the peak-to-peak current for the port.
abs (Σ current)	(**Subckt View** tab only) Displays the absolute value of the sum of the average values of the currents of the ports of the subcircuit.
Σ abs (current)	(**Subckt View** tab only) Displays the sum of the absolute value of the currents of the ports of the subcircuit.
Table Legend	
Subckt, Port, Net	Indicates the symbol for the item and the connection color legend.
Table Tabs	
Detailed View	Shows details of all results in the current analysis table.
Subckt View	Shows results for subcircuits only in the current analysis table.
Buttons	
Analyze/Pause/Resume	Starts, pauses, or resumes the analysis depending on the current state of the dialog box.

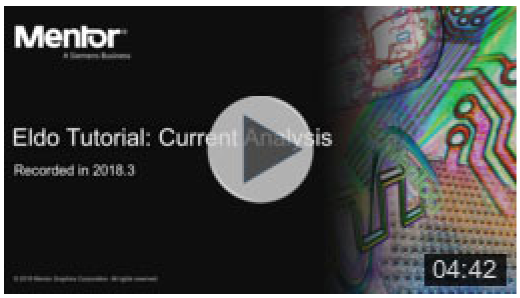
Table 8-11. Current Analysis Dialog Box Contents (cont.)

Field	Description
Clear	Clears the results of the previous analysis.

Usage Notes

Provides an interactive display of results from current analyses, requested by the SPICE command .CURRENT_ANALYSIS in Eldo. Refer to ".CURRENT_ANALYSIS" in the *Eldo Reference Manual*.

A Tutorial—Using Current Analysis is in the *Eldo User's Manual.* A video is available that shows the tasks described in this tutorial.



Related Topics

Analyzing Current Consumption

Cursor Properties Dialog Box

To access: Right-click a cursor and choose **Properties** from the cursor popup menu.

Use this dialog to specify the fonts, line style, and colors for an individual cursor.

Objects

Table 8-12. Cursor Properties

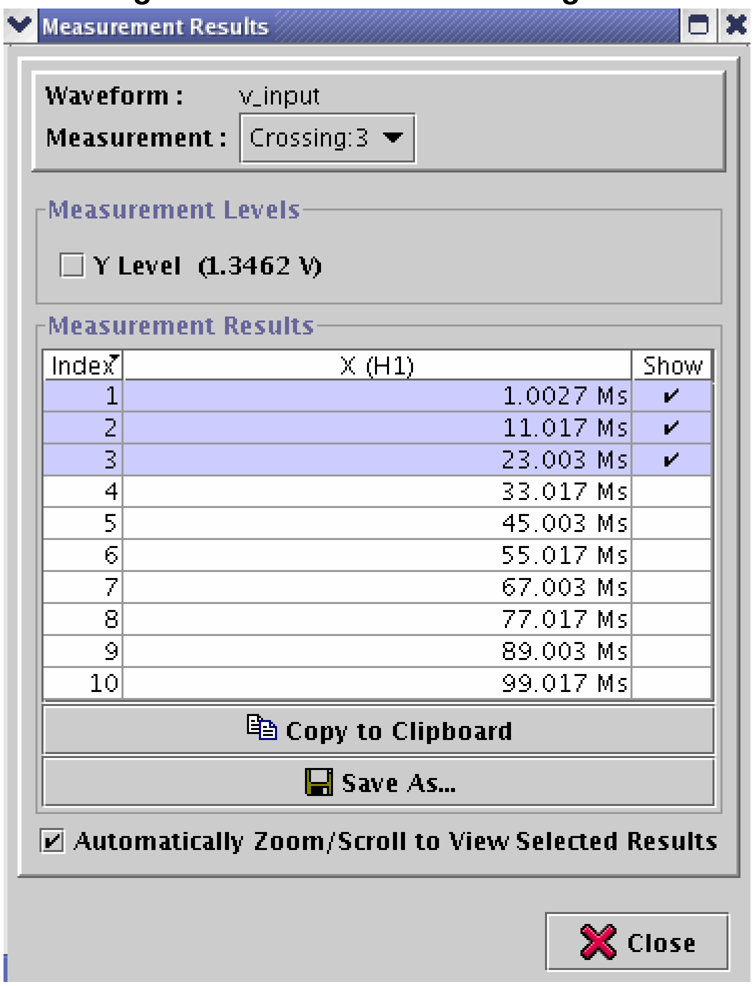
Field	Description
Font	Controls the font of the cursor .
Font Size	Controls the size of the font of the cursor.
Font Style	Controls the font style of cursor.
Y Values	Controls the display color of the Y value associated with the cursor. Click the colored box to open the Color Selection dialog box and make a new color selection.
X Values	Controls the display color of the X value associated with the cursor. Click the colored box to open the Color Selection dialog box and make a new color selection.
Preview	Displays a sample of what the text will look like with the current selections.
Active Cursor	
Cursor Style	Choose a line style from the dropdown list for when the cursor is the active cursor.
Cursor Width	Choose a line width from the dropdown list for when the cursor is the active cursor.
Cursor Color	Controls the color of the cursor line for when the cursor is the active cursor. Click the colored box to open the Color Selection dialog box and make a new color selection.
Cursor	
Cursor Style	Choose a cursor line style from the dropdown list for when the cursor is not the active cursor.
Cursor Width	Choose a cursor line width from the dropdown list for when the cursor is not the active cursor.
Cursor Color	Controls the color of the cursor line for when the cursor is not the active cursor. Click the colored box to open the Color Selection dialog box and make a new color selection.

Data Values Dialog Box

To access: Right-click a horizontal cursor, select **Data Values** then select the required waveform.

Use this dialog box to view the crossing points, where the horizontal cursor intersects with the waveform.

Figure 8-10. Data Values Dialog Box



Objects

Table 8-13. Data Values Dialog Box Contents

Field	Description
Waveform	The name of the currently selected waveform.
Measurement	The index ID of the currently selected crossing point.
Measurement Levels	Specifies whether to show or hide the Y value of the currently selected crossing point.

Table 8-13. Data Values Dialog Box Contents (cont.)

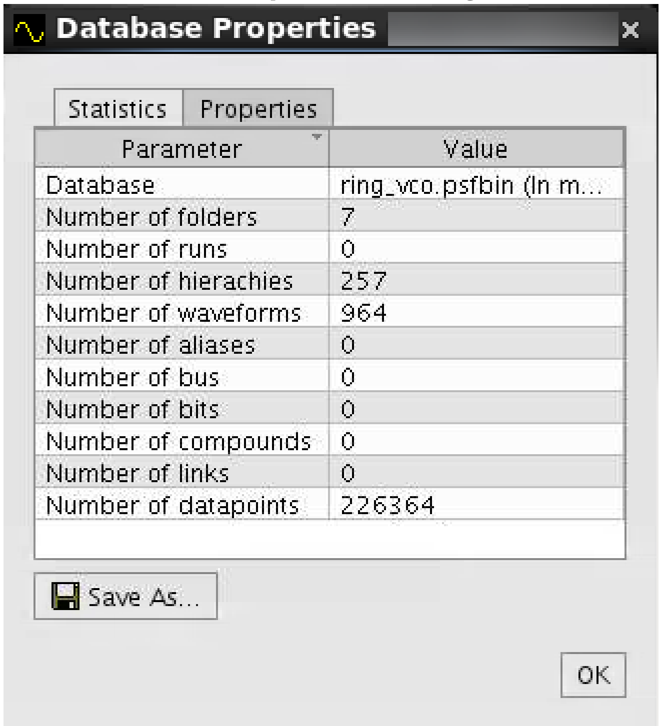
Field	Description
Measurement Results	Lists all of the waveforms crossing points with the horizontal cursor, with their index ID and X value. Specifies whether to show or hide each crossing point.
Copy to Clipboard	Puts the contents of the Measurement Results table on the clipboard.
Save As	Saves the contents of the Measurement Results table to a specified text file (*.csv* or *.txt*).

Database Properties Dialog Box

To access: Right-click a database name and select **Properties**.

Used to display and export the properties and statistics for a waveform database. This dialog box has two tabs, **Statistics** and **Properties**.

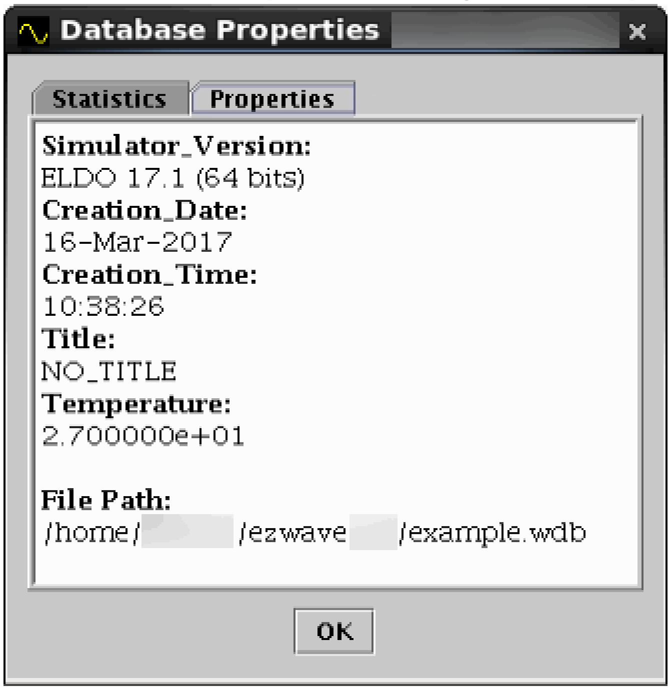
Figure 8-11. Database Properties Dialog Box - Statistics Tab



Note

For *.psf* and *.fsdb* databases, click the **Number of datapoints** button that appears in the Parameter column to determine the number of datapoints. This may take a long time for large databases.

Figure 8-12. Database Properties Dialog Box - Properties Tab



Related Topics

Database Popup Menu

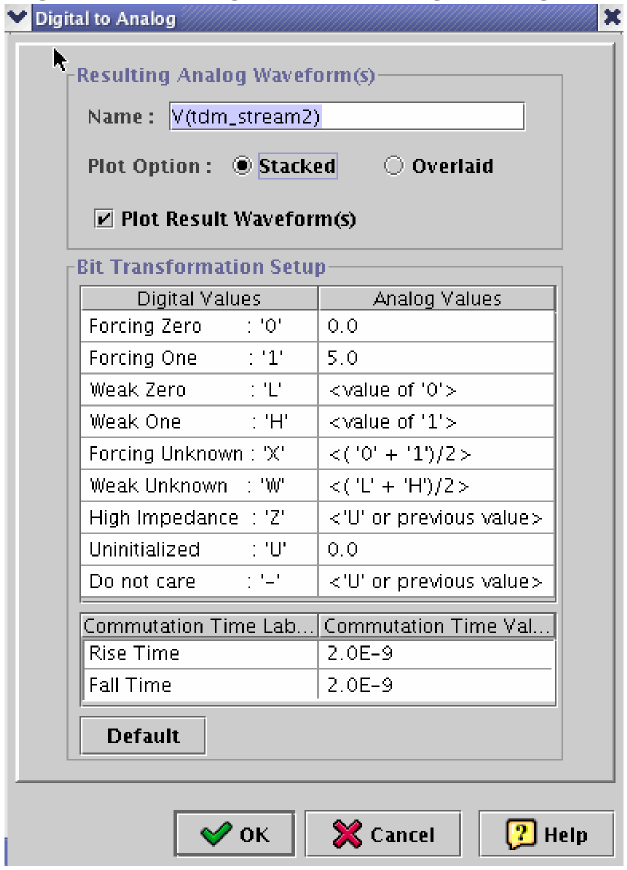
dataset statistics

Digital to Analog Dialog Box

To access: Right-click a digital waveform and select **Digital to Analog**.

Use this dialog box to convert digital waveforms to analog ones and specify options for the conversion.

Figure 8-13. Digital to Analog Dialog Box



Objects

Table 8-14. Digital to Analog Dialog Box Contents

Field	Description
Resulting Analog Waveform(s)	
Name	The name of the input digital waveform

Table 8-14. Digital to Analog Dialog Box Contents (cont.)

Field	Description
Plot Option	Controls how to display the analog waveform: • Stacked Plots the resulting analog waveform in a new row, at the end of the graph window. • Overlaid Plots the resulting analog waveform in the same graph as the input waveform (in the same row).
Plot Result Waveform(s)	Plots the resulting analog waveform using the plot option specified.
Bit Transformation Setup	
Digital Values	Lists the editable digital values
Analog Values	Specifies an analog value for each of the listed Digital Values.
Commutation Time Labels	Rise Time: The time taken to reach value 1 from 0 Fall Time: The time taken to reach value 0 from 1
Commutation Time Values	Specifies a value for Rise Time and Fall Time.
Default	Resets the dialog box, changing all values back to the default setting.

Usage Notes

In some cases, expressions are used to define the default value in the Analog Values field. For example:

- If the default value is the expression <*value_of_0*>, then the analog value is the same value as that specified for Forcing Zero: '0'.
- If the default value is the expression <*'U' or previous value*> then the analog value is the value of U (Uninitialized) if it is the first event (the first point of the waveform; the waveform starts with a 'Z' or a '-') otherwise it keeps the previous value.

You can only alter the value in the Analog Values field, if the value you enter is a double value (with or without suffix) for example, 1, or 1M. If you try to use an expression value, for example <*value_of_0*>, when that expression is not already the default setting for that field, then it will not be accepted, and the default value for that field is used. The following table lists the digital values and their default analog values.

Table 8-15. Bit Transformation Digital and Analog Default Values

Digital Value	Text Value	Default Analog Value
Forcing Zero	0	0.0
Forcing One	1	5.0

Table 8-15. Bit Transformation Digital and Analog Default Values (cont.)

Digital Value	Text Value	Default Analog Value
Weak Zero	L	<value of ‘0’>
Weak One	H	<value of ‘1’>
Forcing Unknown	X	<(‘0’ + ‘1’)/2>
Weak Unknown	W	<(‘L’ + ‘H’)/2>
High Impedance	Z	<‘U’ or previous value>
Uninitialized	U	0.0
Do not care	-	<‘U’ or previous value>

Note

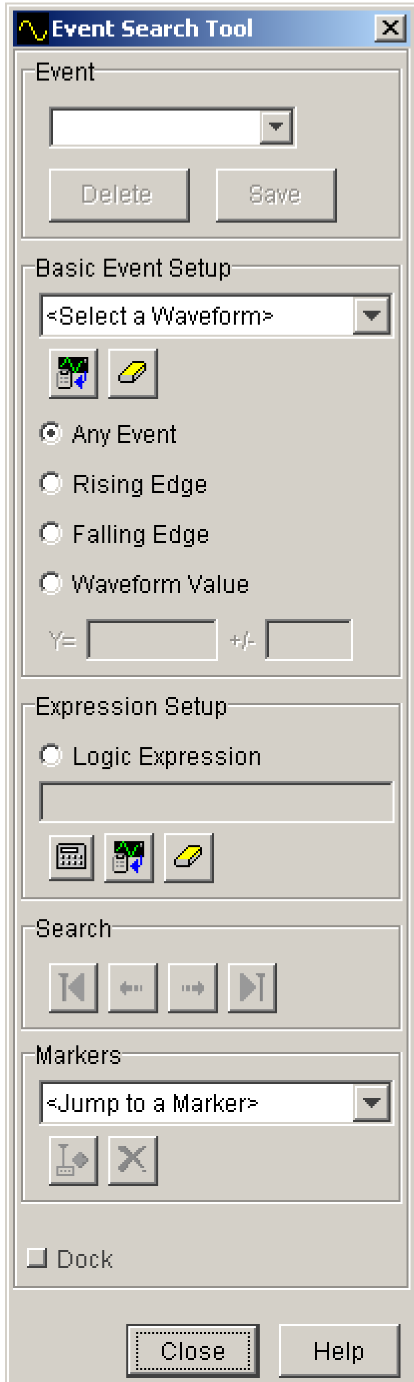
The value for Forcing Zero: '0' must be less than the value for Forcing One: '1'.

Event Search Tool Dialog Box

To access: Choose **Tools** > **Search** from the main menu.

Use this dialog box to locate occurrences of simulation events interactively.

Figure 8-14. Event Search Tool Dialog Box



Objects

Table 8-16. Event Search Tool Dialog Box Contents

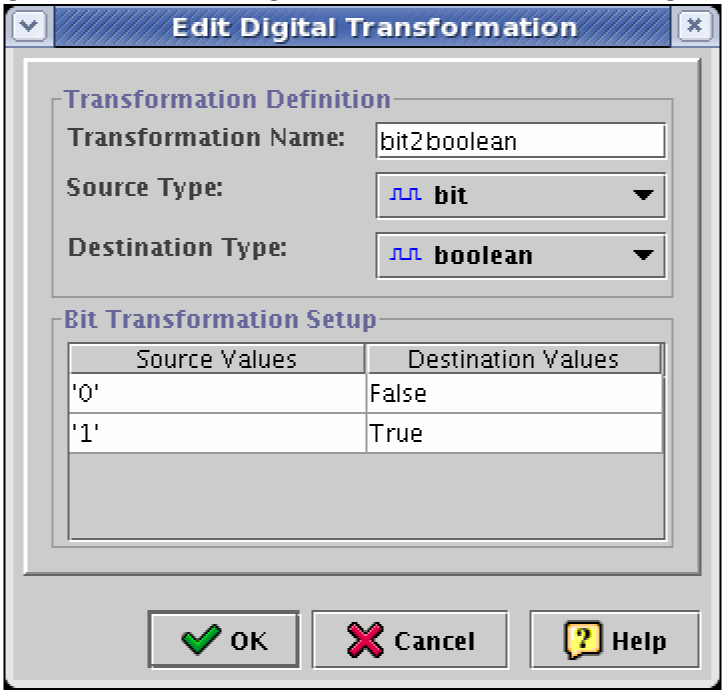
Field	Description
Event	Specifies a name for the event
Basic Event Setup	
Select a Waveform	Specifies the source waveform. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
	Specifies the type of event from: • **Any Event** — Moves from one event to the previous or next. • **Rising Edge** — Moves from one rising edge to the previous or next rising edge. Click the **Rising Edge** icon to open the Search Event Settings dialog box. • **Falling Edge** — Moves from one falling edge to the previous or next rising edge. Click the **Rising Edge** icon to open the Search Event Settings dialog box. • **Waveform Value**— Moves from one event to the previous or next.
Y= +/-	Specify a tolerance when searching for data points generated by an analog simulator Only available when Waveform Value is selected.
Expression Setup	
Logic Expression	Specifies an Expression Event search. The expression can be entered by using either the Waveform Calculator button or the **Add Selected Waveform** icon . Refer to "Performing an Expression Event Search" on page 195.
Cursor	
	Move the cursor along the specified waveform by clicking the arrow buttons. The Previous Marker and Next Marker buttons jump to previous and next markers respectively.
Markers	
Jump to a marker	Specifies a marker to jump to.
	The Add a Marker icon specifies an event to mark.

Edit Digital Transformation Dialog Box

To access: Choose **Edit** > **Options** from the main menu to display the EZwave Display Preferences dialog box. Select **Transformations** from the list on the left to display the Transformation Options. Click the **New** or **Edit** icon.

Use this dialog box to define a new transformation for an enumerated type, or edit an existing one.

Figure 8-15. Edit Digital Transformation Dialog Box



Objects

Table 8-17. Edit Digital Transformation Dialog Box Contents

Field	Description
Transformation Definition	
Transformation Name	Specifies a name for the transformation.
Source Type	Specifies the type of the source. The options available in the dropdown list depend on the enumerated types that exist in the loaded database.

Table 8-17. Edit Digital Transformation Dialog Box Contents (cont.)

Field	Description
Destination Type	Specifies the type of the destination as one of the following: • analogic • bit • Boolean • standard logic • Verilog logic • VHDL characters
Bit Transformation Setup	
Source Values	All possible source values are specified in the Source Values column.
Destination Values	The corresponding cell for each of the Source Values specifies the value of the destination as one of the options in the dropdown list. select a destination value for each source value

Usage Notes

A Destination Value must be selected for each Source Value in the Bit Transformation Setup table.

Eye Diagram Tool Dialog Box

Use this dialog box to create an eye diagram based on a period of waveform data using the Eye Diagram Tool.

Eye Diagram Tool - Settings Tab **458**

Eye Diagram Tool - Measurement Results Tab **465**

Eye Mask Dialog Box **470**

Eye Diagram Tool - Settings Tab

To access: Choose **Tools** > **Eye Diagram** from the main menu and click the **Settings** tab.

Use the **Settings** tab to create an eye diagram based on a period of waveform data using the Eye Diagram Tool.

Figure 8-16. Eye Diagram Tool Dialog Box - Settings Tab for C-Phy Eye Type

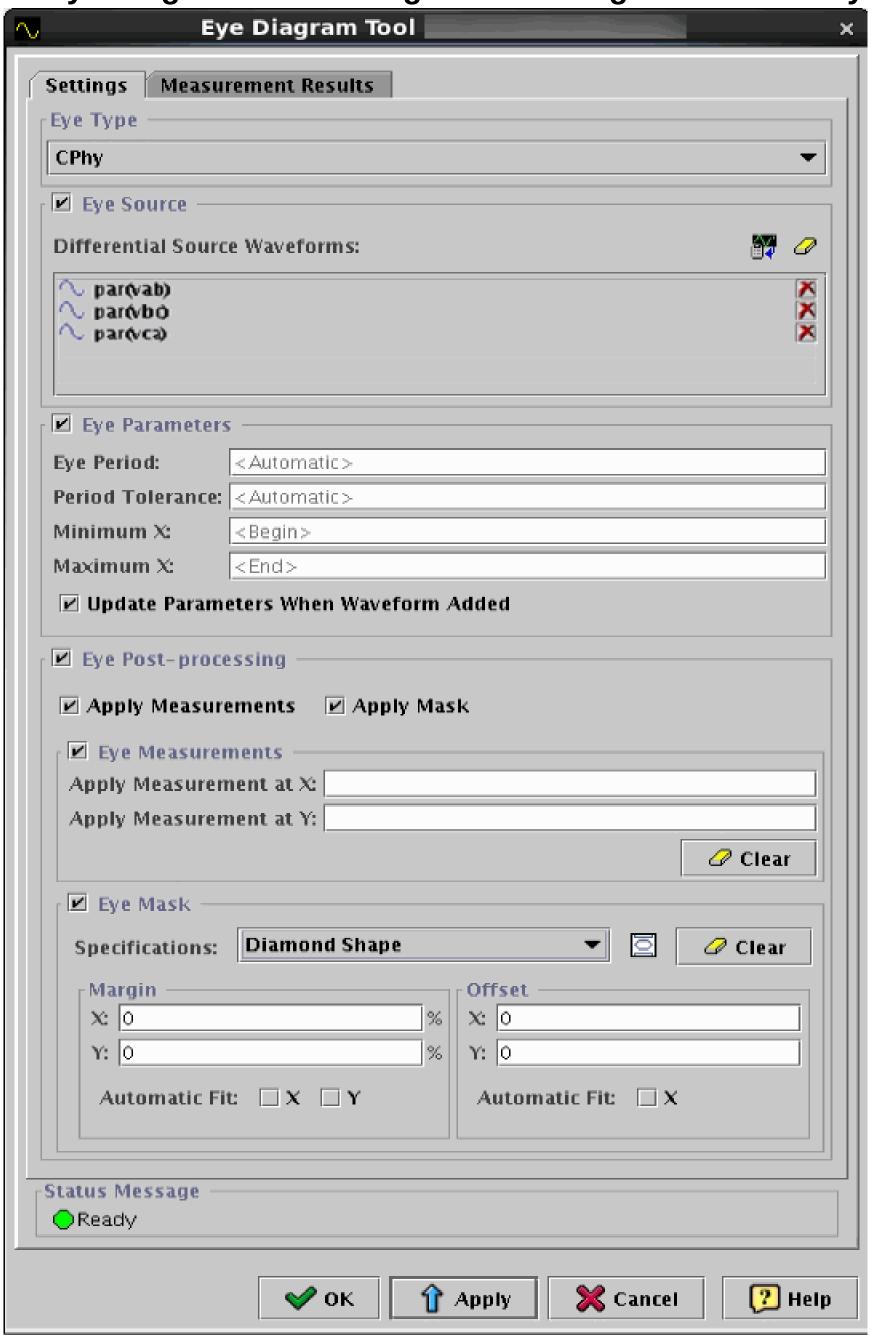


Figure 8-17. Eye Diagram Tool Dialog Box - Settings Tab for NRZ Eye Type

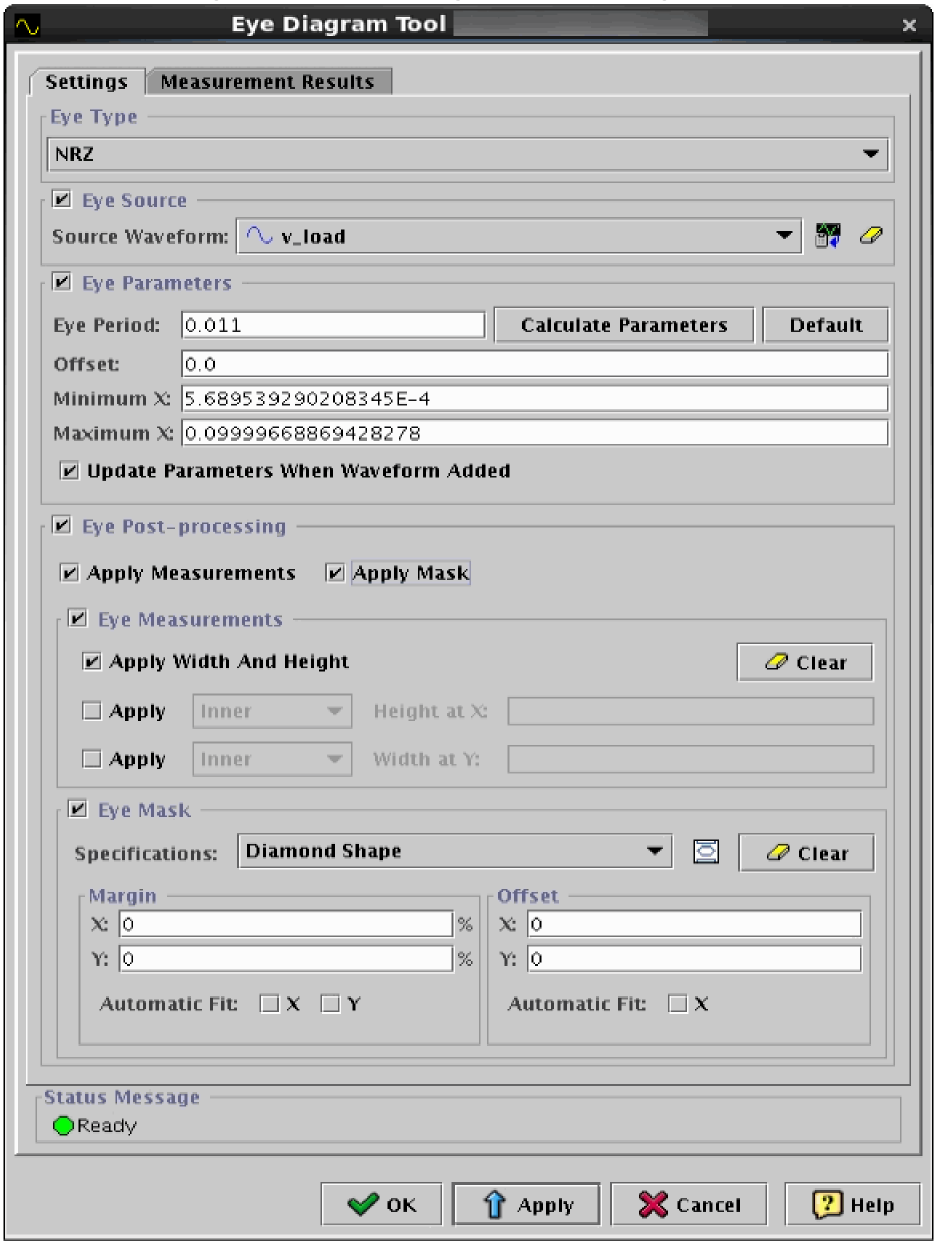
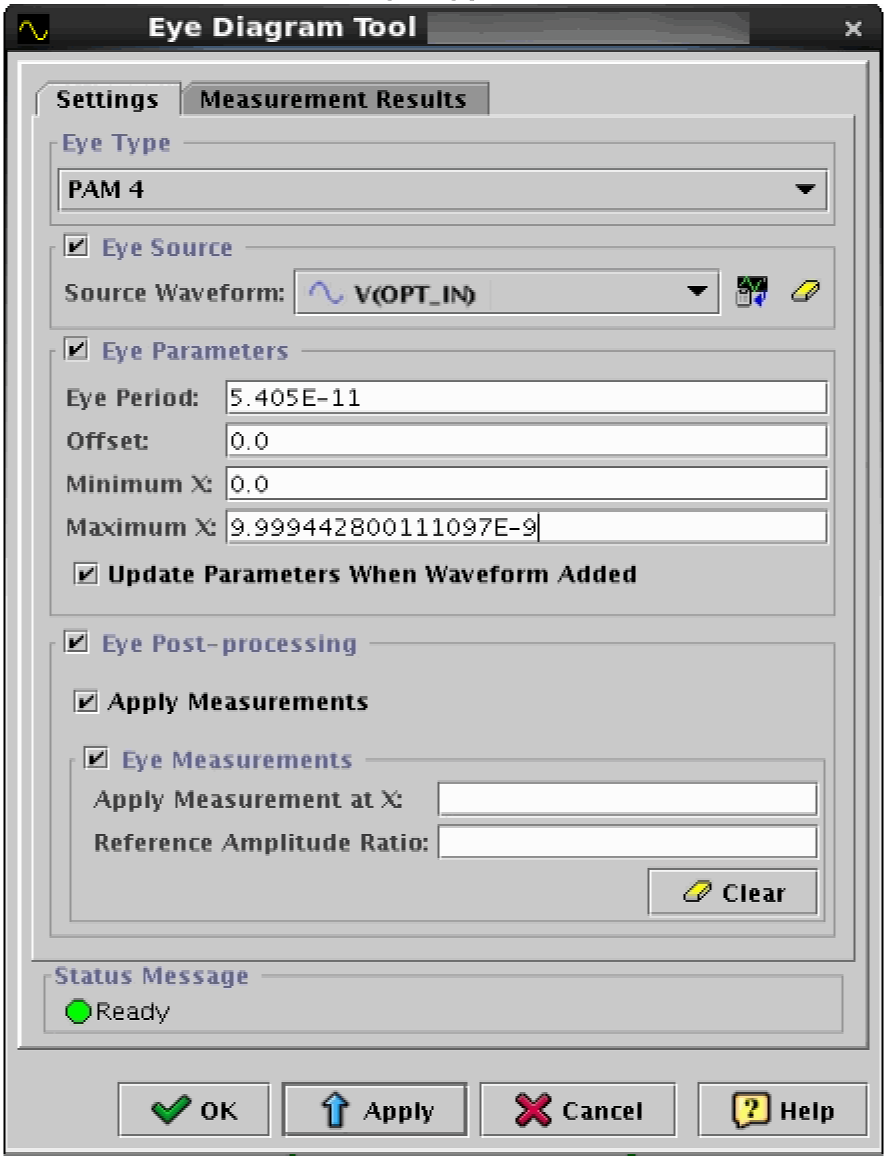


Figure 8-18. Eye Diagram Tool Dialog Box - Settings Tab for PAM 3 and PAM 4 Eye Types



Objects

Table 8-18. Eye Diagram Tool Dialog Box - Settings Tab Contents

Field	Description
Eye Type	
Eye Type	Specifies the eye type, either **C-Phy**, **NRZ**, **PAM 3** or **PAM 4**. The fields on the dialog box change depending on the setting.

Table 8-18. Eye Diagram Tool Dialog Box - Settings Tab Contents (cont.)

Field	Description
Eye Source	
Source Waveform	Specifies the source waveform. For C-Phy you must specify exactly three source waveforms in order (first waveform is C(A,B). The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Eye Parameters	
Eye Period	Specifies the eye period. The eye diagram is generated by overlaying a semi-periodical waveform signal on an interval in X (usually a time interval). The interval is defined as the eye period. A default Eye Period is calculated based on the source waveform period divided by 2.
Offset	Specifies the offset value. Enables shifting of the eye as the open part of the eye is not always at the center of the axes.
Period Tolerance	For C-Phy eye type only. Specifies the tolerance on the period used to detect transition.
Minimum X and Maximum X	Specifies the range of waveform data used for generating the eye diagram.
Update Parameters When Waveform Added	Updates the Eye Parameters when a waveform as added. If the added waveform is not an eye diagram, the Eye Diagram Tool tries to calculate the default eye parameters (except eye measurements and mask). If the added waveform is an eye, the tool tries to load or calculate all possible parameters (including eye measurements and mask).
Calculate Parameters	Calculates the NRZ Offset, Minimum X, and Maximum X using the Eye Period value and automatically populates the fields.
Eye Post-processing	
Apply Measurements	Specifies that the measurements are automatically updated as you modify the values in this dialog box. This updates the appropriate values in the eye diagram as well as all the values in the Eye Diagram Tool - Measurement Results Tab. Checked by default, the measurements are computed when the eye diagram is created.
Apply Mask	For C-Phy and NRZ signals only. Specifies that the selected eye mask is applied.

Table 8-18. Eye Diagram Tool Dialog Box - Settings Tab Contents (cont.)

Field	Description
Eye Measurements	
Apply Width and Height	For NRZ signals only. Specifies the eye width and eye height measurements are displayed on the eye diagram. Unchecked by default.
Apply and Height at X	Specifies an inner or outer height to display at a specific X. Use the drop down box to determine inner or outer and specify the desired X. Unchecked by default.
Apply and Width at Y	Specifies an inner or outer width to display at a specific Y. Use the drop down box to determine inner or outer and specify the desired Y. Unchecked by default.
Apply Measurement at X	For PAM 3, PAM 4 and C-Phy signals only. Specify the X value where the measurements are to be made.
Apply Measurement at Y	For C-Phy signals only. Specify the Y value where the measurements are to be made.
Reference Amplitude Ratio	For PAM 3 and PAM 4 signals only. When set to “Automatic”, toplines and baselines begin at 10% and increase in steps of 5% until all points are either below the baseline and above the topline (the Measured Edge Percentage reaches 100%). Valid values are “Automatic”, a double value between 0.0 and 1.0, or a number followed by % (e.g. “20%”). Default value is “Automatic” (at least 10%, increasing by 5%).

Table 8-18. Eye Diagram Tool Dialog Box - Settings Tab Contents (cont.)

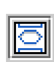
Field	Description
Eye Mask	
Specifications	Eye masks may be applied to C-Phy and NRZ signals. Specifies an industry standard mask is to be applied. When Apply Mask is checked, specifies a mask from: • DDR2 • FC-100DF/SE_RX_beta • FC-100DF/SE_RX_delta • FC-100DF/SE_RX_gamma • FC-100DF/SE_TX_beta • FC-100DF/SE_TX_delta • FC-100DF/SE_TX_gamma • FC-200DF/SE_RX_delta • FC-200DF/SE_RX_gamma • FC-200DF/SE_TX_delta • FC-200DF/SE_TX_gamma • PCIE_RX • PCIE_TX_de-emphasis • PCIE_TX_transition • QDRII • SAS_1.5Gb_RX • SAS_3Gb_RX • SATA1_RX • SATA1_TX • USB2.0-High_Speed_RX • USB2.0-High_Speed_TX • XAUI-Xilinx-rcvr • Diamond Shape • Key Shape • T Shape • Trapezoid Shape The **Eye Mask** icon  opens the Eye Mask Dialog Box.

Table 8-18. Eye Diagram Tool Dialog Box - Settings Tab Contents (cont.)

Field	Description
Margin	Specifies scaling of the eye mask. The X margin is the horizontal margin and the Y margin is the vertical margin. The values entered representing a percentage of the total displayed range. The **Automatic Fit** options specify that the mask is to be fitted to the horizontal (X) and/or vertical (Y) inner contours of the eye diagram. When selected, a homothecy is applied in the X and/or Y direction until one side of the inner contour is reached.
Offset	Specifies the offset value. Enables shifting of the eye mask. The X offset is the horizontal offset. Select the **Automatic Fit** (X) checkbox to automatically determine the horizontal offset.

Usage Notes

Refer to "Working with Eye Diagrams" on page 197.

Eye Diagram Tool - Measurement Results Tab

To access: Choose **Tools** > **Eye Diagram** from the main menu and click the **Measurement Results** tab.

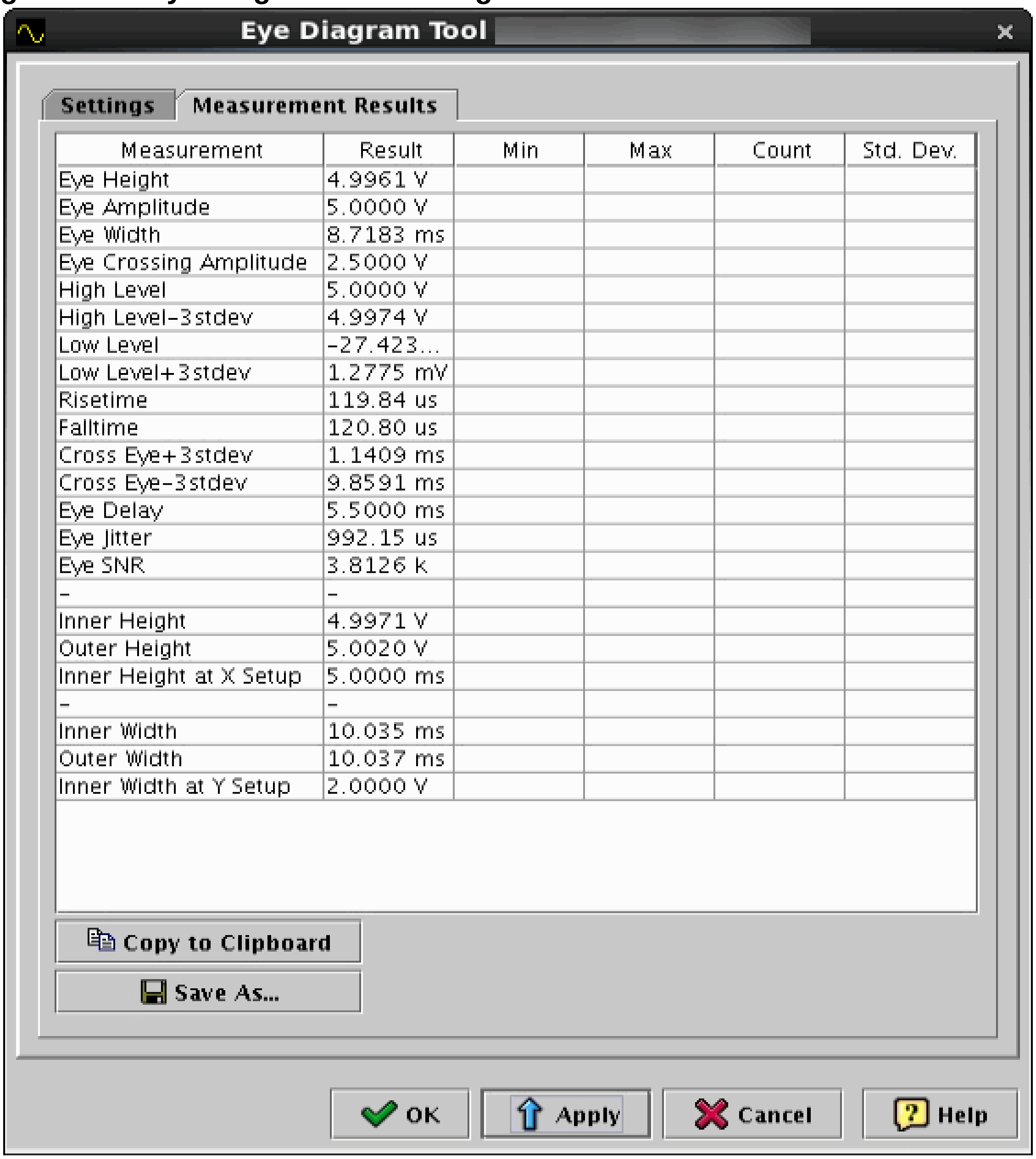
The **Measurement Results** tab displays the Eye Diagram values calculated from the current setup in the **Settings** tab.

Objects

- For details of each measurement refer to:

  "Eye Diagram Measurement Calculations" on page 206.

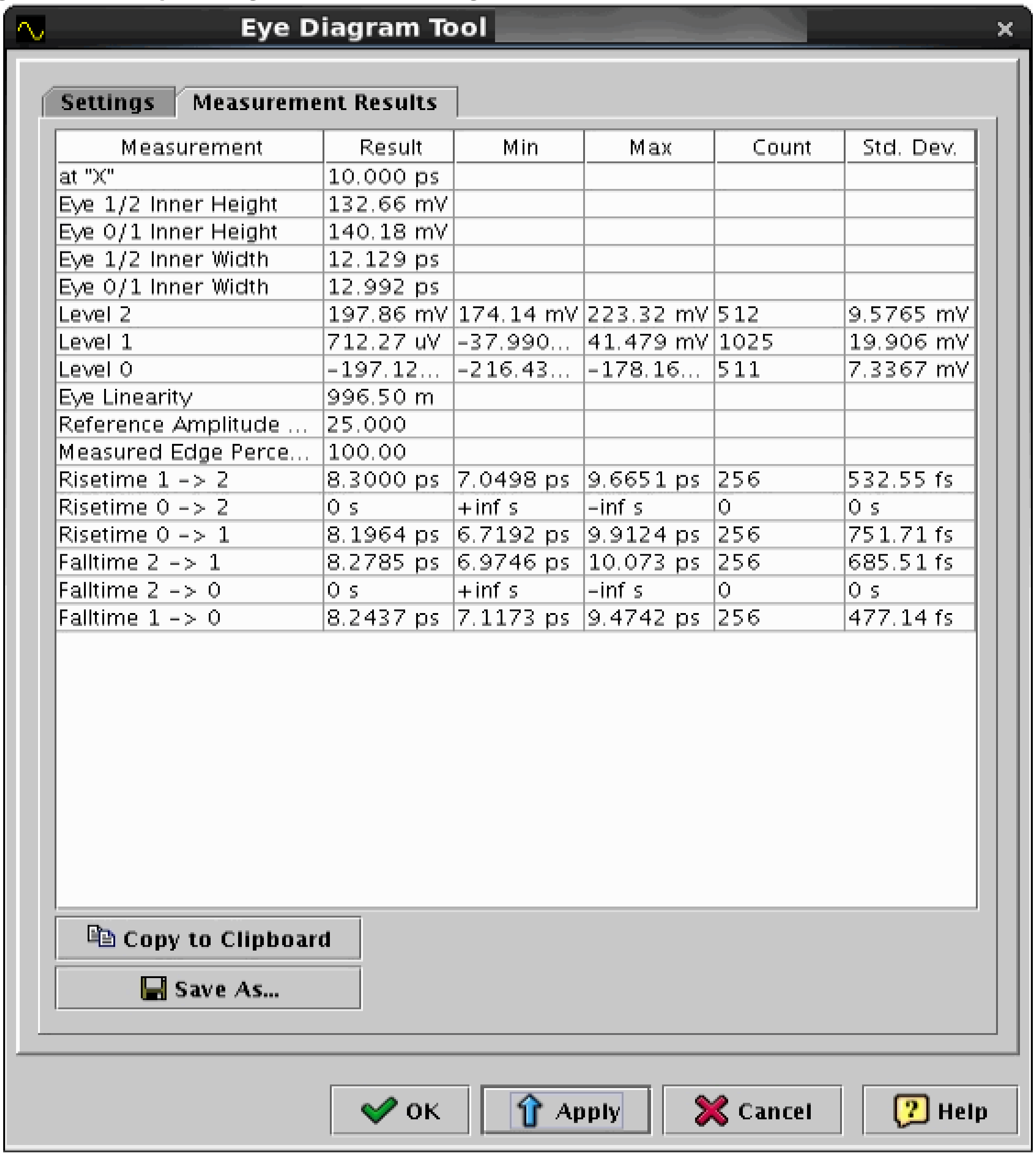
Figure 8-19. Eye Diagram Tool Dialog Box - Measurement Results Tab for NRZ



Measurement	Result	Min	Max	Count	Std. Dev.
Eye Height	4.9961 V				
Eye Amplitude	5.0000 V				
Eye Width	8.7183 ms				
Eye Crossing Amplitude	2.5000 V				
High Level	5.0000 V				
High Level-3stdev	4.9974 V				
Low Level	-27.423...				
Low Level+3stdev	1.2775 mV				
Risetime	119.84 us				
Falltime	120.80 us				
Cross Eye+3stdev	1.1409 ms				
Cross Eye-3stdev	9.8591 ms				
Eye Delay	5.5000 ms				
Eye Jitter	992.15 us				
Eye SNR	3.8126 k				
-	-				
Inner Height	4.9971 V				
Outer Height	5.0020 V				
Inner Height at X Setup	5.0000 ms				
-	-				
Inner Width	10.035 ms				
Outer Width	10.037 ms				
Inner Width at Y Setup	2.0000 V				

Refer to "NRZ (PAM2) Cross Eye Calculation" on page 206.

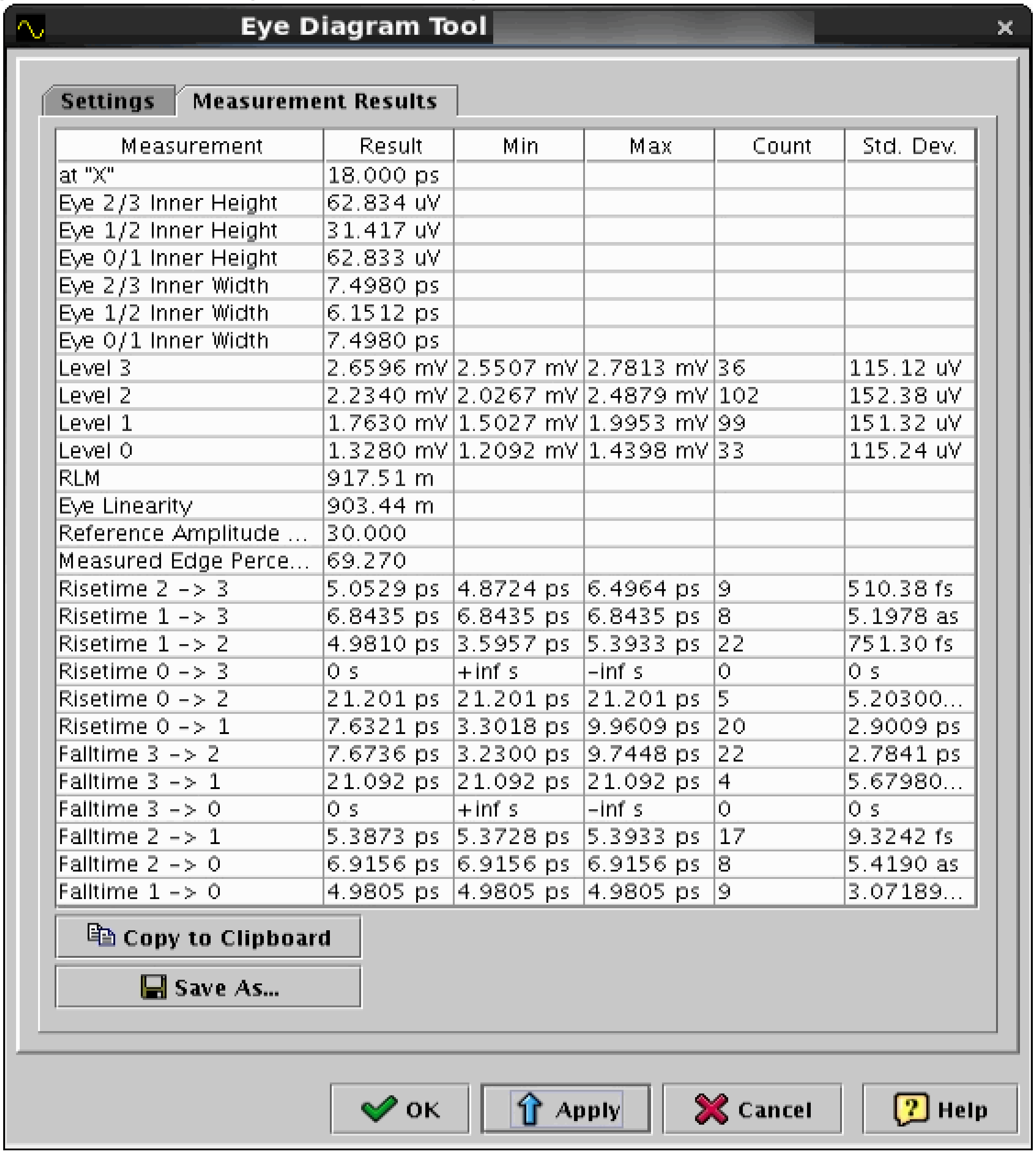
Figure 8-20. Eye Diagram Tool Dialog Box - Measurement Results Tab for PAM3



Measurement	Result	Min	Max	Count	Std. Dev.
at "X"	10.000 ps				
Eye 1/2 Inner Height	132.66 mV				
Eye 0/1 Inner Height	140.18 mV				
Eye 1/2 Inner Width	12.129 ps				
Eye 0/1 Inner Width	12.992 ps				
Level 2	197.86 mV	174.14 mV	223.32 mV	512	9.5765 mV
Level 1	712.27 uV	-37.990...	41.479 mV	1025	19.906 mV
Level 0	-197.12...	-216.43...	-178.16...	511	7.3367 mV
Eye Linearity	996.50 m				
Reference Amplitude ...	25.000				
Measured Edge Perce...	100.00				
Risetime 1 -> 2	8.3000 ps	7.0498 ps	9.6651 ps	256	532.55 fs
Risetime 0 -> 2	0 s	+inf s	-inf s	0	0 s
Risetime 0 -> 1	8.1964 ps	6.7192 ps	9.9124 ps	256	751.71 fs
Falltime 2 -> 1	8.2785 ps	6.9746 ps	10.073 ps	256	685.51 fs
Falltime 2 -> 0	0 s	+inf s	-inf s	0	0 s
Falltime 1 -> 0	8.2437 ps	7.1173 ps	9.4742 ps	256	477.14 fs

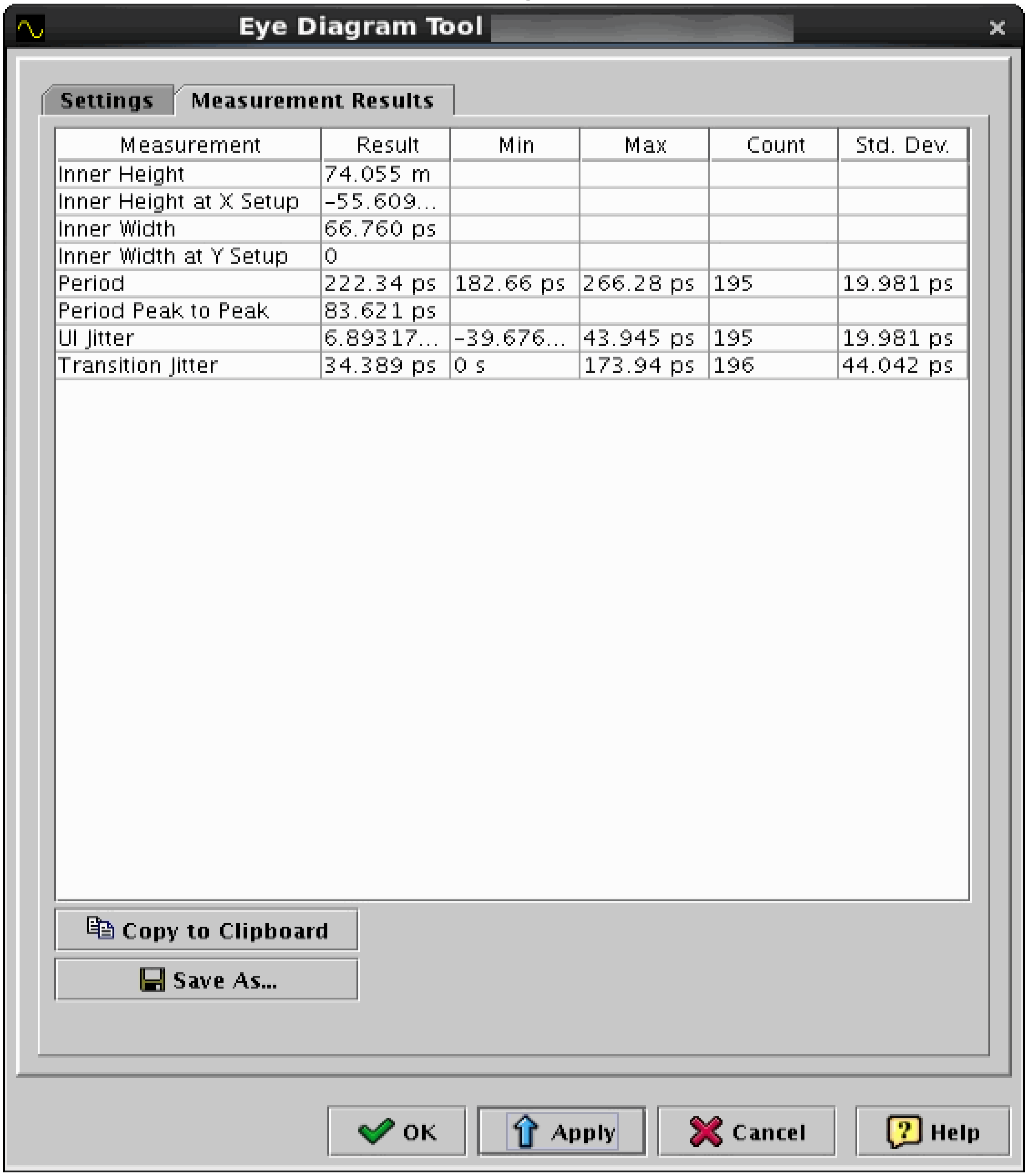
Refer to "PAM3 and PAM4 Eye Calculations" on page 208.

Figure 8-21. Eye Diagram Tool Dialog Box - Measurement Results Tab for PAM4



Refer to "PAM3 and PAM4 Eye Calculations" on page 208.

Figure 8-22. Eye Diagram Tool Dialog Box - Measurement Results Tab for C-Phy



Refer to “C-Phy Eye Calculation” on page 213.

Usage Notes

Refer to “Working with Eye Diagrams” on page 197.

For the PAM3 and PAM4 results, the first row (at “X”) result is the measurement setup value, entered by user in Measurements Setup section of the Eye Diagram Tool - Settings Tab.

For the levels, risetimes and falltimes:

- The Result column is the mean of all individual results
- The Min column is the lower bound
- The Max column is the upper bound
- The Count column is the number of individual results
- The Std. Dev. column is the standard deviation.

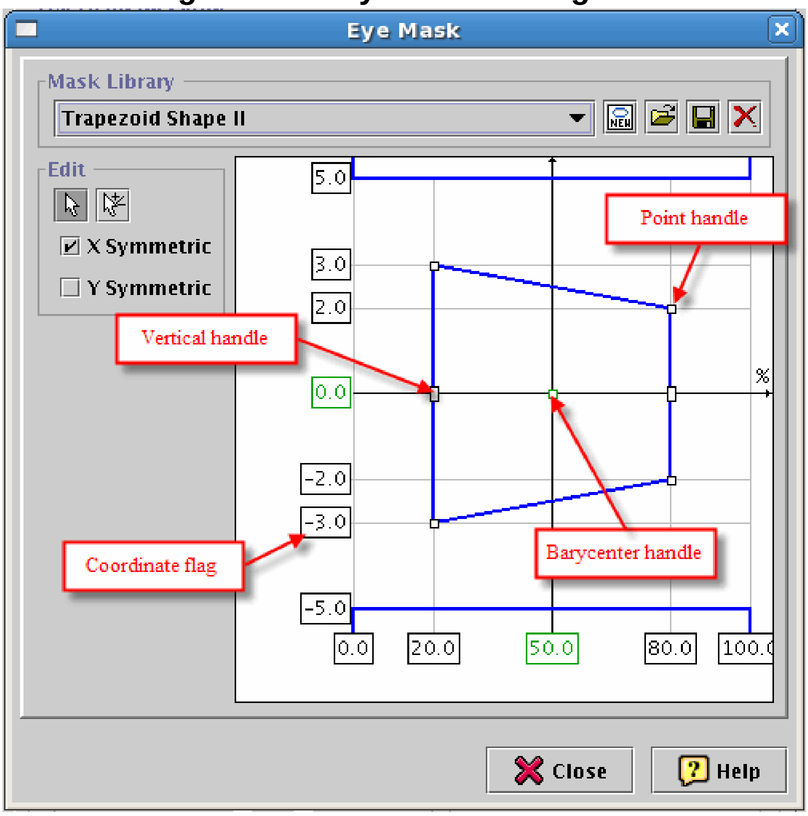
The height of eye 2/3 equals Min("level 3") – Max("level 2").

Eye Mask Dialog Box

To access: Choose **Tools** > **Eye Mask** from the main menu, then click the **Eye Mask** icon in the Eye Diagram Tool Dialog Box.

Use this dialog box to view and edit built-in and user defined eye masks.

Figure 8-23. Eye Mask Dialog Box



Objects

Table 8-19. Eye Mask Dialog Box Contents

Field	Description
Mask Library	Specifies the source eye mask. Select a built in eye mask from the drop down list or alternatively use the **Open Folder** icon to load a user defined eye mask from disk. The **Edit New Mask** icon enables editing of the mask. Once editing is enabled, the grayed out **Save** and **Delete** buttons will become available.
Edit	
	When activated indicates that the mask is in “edit” mode and the handles are available for dragging the mask to form a new shape.

Table 8-19. Eye Mask Dialog Box Contents (cont.)

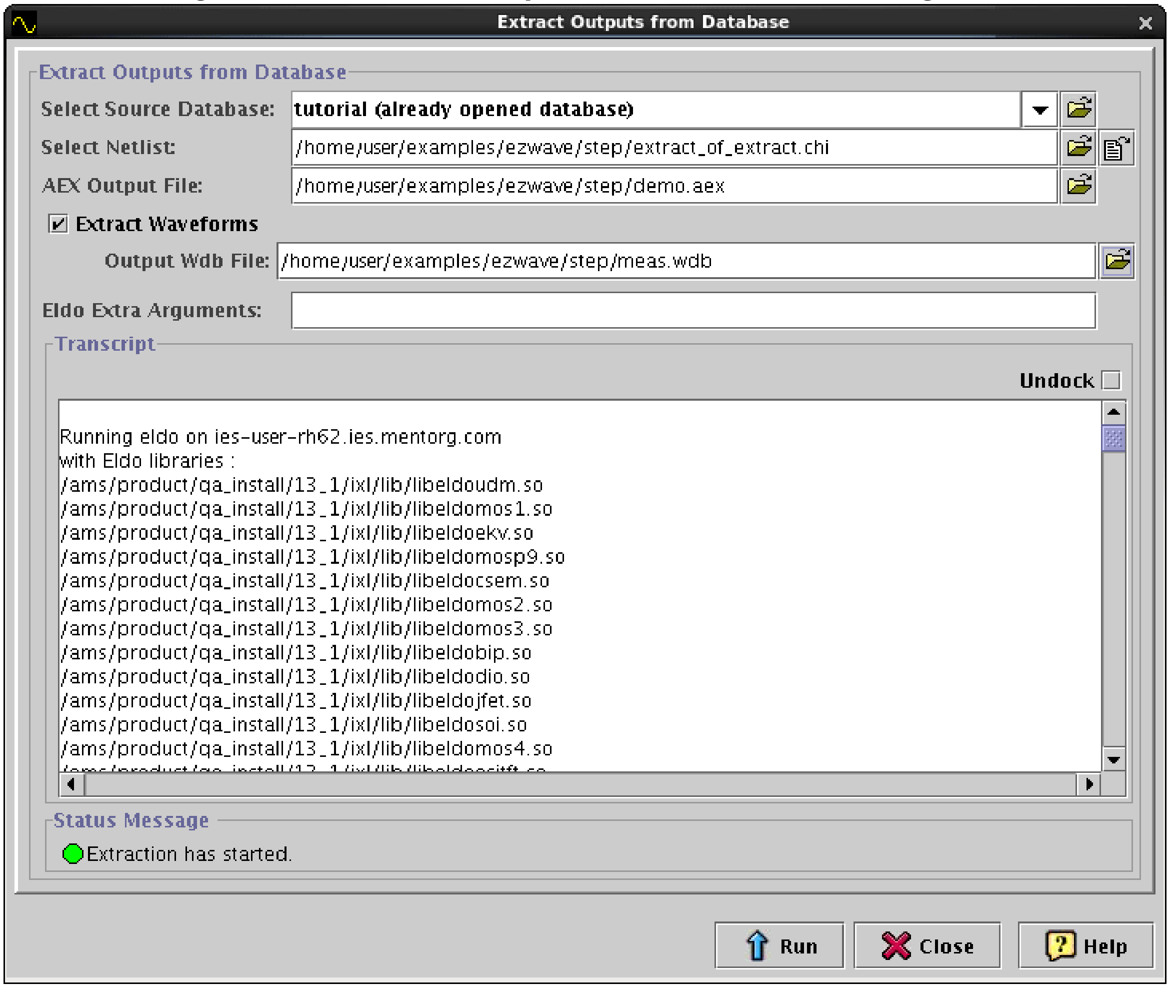
Field	Description
	When activated, handles can be added or removed from the mask, and the mask shape is modified accordingly.
X Symmetric	Specifies that the X symmetric handle should be moved accordingly when a handle is dragged within the mask. This option is only possible in masks with X symmetry.
Y Symmetric	Specifies that the Y symmetric handle should be moved accordingly when a handle is dragged within the mask. This option is only available in masks with Y symmetry.

Extract Outputs From Database Dialog Box

To access: Choose **Tools** > **Extract** from the main menu.

Use this dialog box to perform an extract on a previously simulated database.

Figure 8-24. Extract Outputs From Database Dialog Box



Objects

Table 8-20. Extract Outputs From Database Dialog Box Contents

Field	Description
Select Source Database	Specify the database upon which to run the .extract command. Either type the path into the field or click the folder icon to browse to the database file. The down-arrow button offers a list of recent and open database files.
Select Netlist	By default, this displays the path to a netlist with the same name as the database file selected above, but you can specify a different netlist to use.

Table 8-20. Extract Outputs From Database Dialog Box Contents (cont.)

Field	Description
AEX Output File	By default, this is set to an *.aex* file with the same name as the database file selected above, but you can specify a different *.aex* file if required. **Note:** Extracted scalar values are only written to the file if the AEX option is specified for the .extract command.
Extract Waveforms	With this option enabled, waveforms are generated for the extracted values, and stored in the database file you specify in the Output Wdb File field.
Eldo Extra Arguments	Enter any arguments you want to pass to Eldo.
Transcript	This pane displays the output transcript from the simulation. The **Undock** option enables you to separate the transcript pane from the dialog box. Select **Undock** option to split the panes; clear the option on the Transcript dialog box to re-dock.
Run	Click to start the simulation.

EZwave Display Preference Options

This section describes the EZwave Display Preferences dialog box and the different options.

EZwave Display Preferences Dialog Box 476
Automatic Reload Options 478
Cursor Options 480
Data Format Options 482
Foreign Databases Options 484
General Options 485
Grid Options 488
Layout Options 490
Look and Feel Options 491
Mouse Pointer Options 492
Multiple Run Options 493
Pick Points Options 495
RF Options 498
Row Options 499
Save Data Options 501
Save Window Options 503
Text Annotation Options 504
Transformation Options 505
Waveform Options 507
Waveform Compare Options 508
Waveform List Options 509
Workspace Options 511
CDF Plot Options 512
CDF Measures Options 514
CDF Legend Options 516
Axis Title Options 518
Axis Values Options 519
Axis Values (Smith Chart) Options 520
Calculator Entry Options 521
Cursor/Marker Options 522
Eye Mask Options 524
Grid Options (Fonts and Colors) 525

Header Text Options 526
Histogram Options. 527
Measurement Annotation Options 528
Pick Points Options (Fonts and Colors). 529
Row Title Options 530
Text Annotation Options (Fonts and Colors) 531
Waveform Colors Options 532
Waveform Display Options. 533
Waveform Name Options 534
Waveform Selection Options 536
Window Background Options 537
Zero-Level Line Options 538
Histogram Plot Options. 539
Histogram Measures Options 541
Histogram Legend Options. 543
Waveform Calculator Calculation Options. 545
Waveform Calculator General Options. 547
Waveform Calculator View Options 549

EZwave Display Preferences Dialog Box

To access: Choose **Edit** > **Options** from the main menu.

Sets preferences for EZwave display behavior.

Objects

- EZwave Preferences are grouped into the following categories: EZwave Preferences, CDF Preferences, Fonts and Colors, Histogram Preferences and Waveform Calculator Preferences. Access the required options by selecting from the list on the left of the dialog box.

Figure 8-25. EZwave Display Preferences Dialog Box

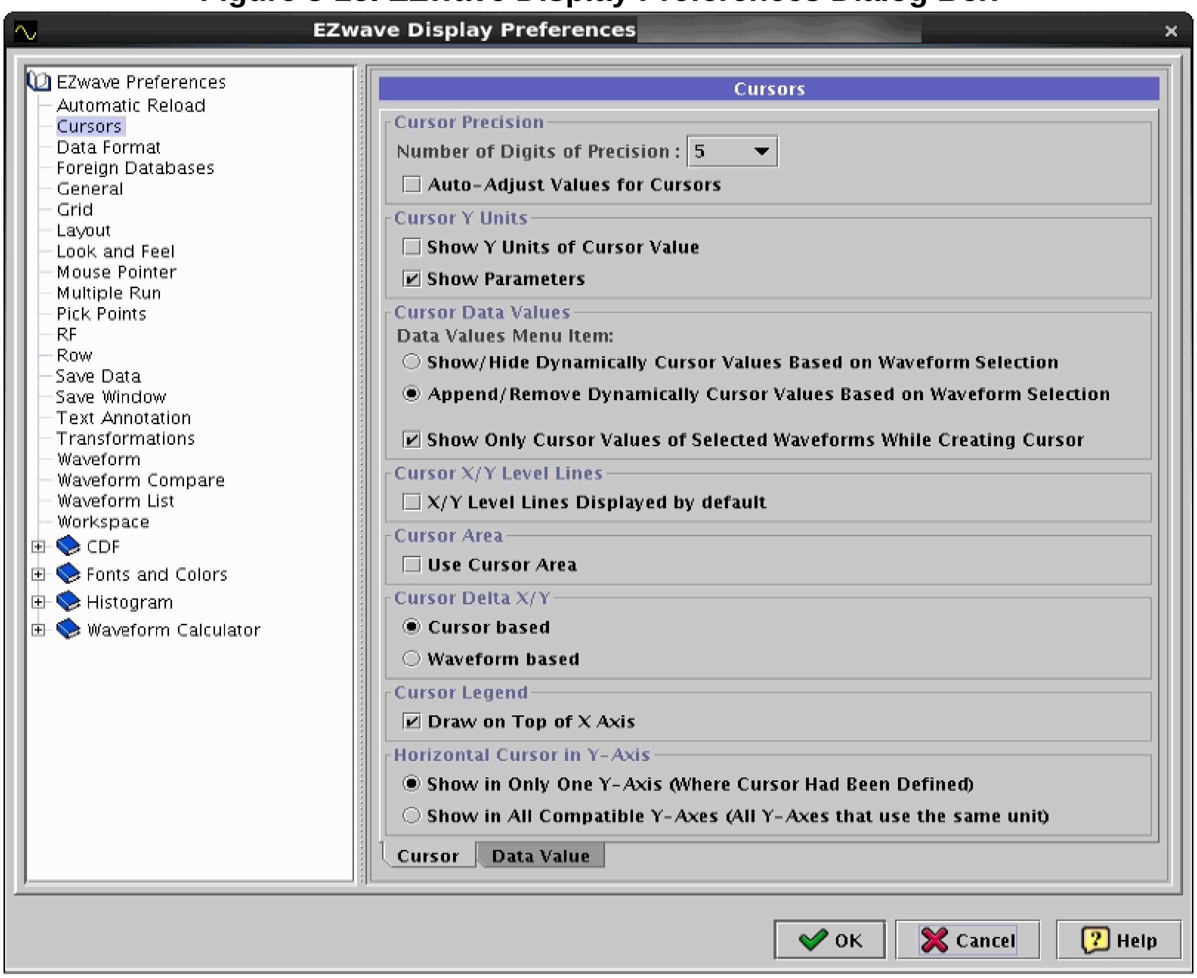


Table 8-21. EZwave Preferences

Automatic Reload Options	Cursor Options
Data Format Options	Foreign Databases Options
General Options	Grid Options
Layout Options	Look and Feel Options

Table 8-21. EZwave Preferences (cont.)

Mouse Pointer Options	Multiple Run Options
Pick Points Options	RF Options
Row Options	Save Data Options
Save Window Options	Text Annotation Options
Transformation Options	Waveform Options
Waveform Compare Options	Waveform List Options
Workspace Options	

Table 8-22. CDF Preferences

CDF Plot Options	CDF Measures Options
CDF Legend Options	

Table 8-23. Fonts and Colors

Axis Title Options	Axis Values Options
Axis Values (Smith Chart) Options	Calculator Entry Options
Cursor/Marker Options	Eye Mask Options
Grid Options (Fonts and Colors)	Header Text Options
Histogram Options	Measurement Annotation Options
Pick Points Options (Fonts and Colors)	Row Title Options
Text Annotation Options (Fonts and Colors)	Waveform Colors Options
Waveform Display Options	Waveform Name Options
Waveform Selection Options	Window Background Options
Zero-Level Line Options	

Table 8-24. Histogram Preferences

Histogram Plot Options	Histogram Measures Options
Histogram Legend Options	

Table 8-25. Waveform Calculator Preferences

Waveform Calculator Calculation Options	Waveform Calculator General Options
Waveform Calculator View Options	

Automatic Reload Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Automatic Reload from the EZwave Preferences list on the left.

Used to specify whether previous results are kept when a new simulation is run.

Note

Automatic Reload is not possible on the Windows operating system.

Objects

Table 8-26. EZwave Display Preferences - Automatic Reload Options

Field	Description
Replace Previous Results	Specifies that previous results are overwritten when a new simulation is run.
Warn User Before Reloading	Specifies that a warning is displayed before each reload.
Only Reload Modified Database	Specifies that only databases that have changed are reloaded. Applies only to JWDB format databases.
Keep Parameter Table Show/Highlight Filters	Specifies that the Show and Highlight filter settings for compound waveforms on the Parameter Table Dialog Box are maintained after a **File** > **Reload** action (from the same database).
Keep Previous Result	The number of previous results specified in the Keep field are kept. The oldest results are automatically deleted when the *N* limit is reached.
Ask Me	EZwave will ask whether you want to keep or delete the previous results when the N limit is reached.
Display New Result Automatically	Specifies that the new result is displayed automatically, overlaid with the previous one. Leave unchecked to automatically load the new results without displaying them.
Only Close Old Result (do not delete file)	Prevents automatic deletion of oldest results when the *N* limit is reached. If this is not checked, when the limit is reached the oldest results are removed from disk and the related data removed from the EZwave display.

Table 8-26. EZwave Display Preferences - Automatic Reload Options (cont.)

Field	Description
Keep All Waveforms	Preserves the display of all previously displayed waveforms. This option is useful if some previously displayed waveforms are not created immediately at the start of a simulation (when several analyses are run successively) or temporarily disappear from the database (due to a netlist modification). With the Keep All Waveforms option checked, their names will remain displayed, and the plotted waveforms will temporarily disappear until they appear again in the database.

Usage Notes

The database will be renamed from *file.wdb* to *file_sim1.wdb*, *file_sim2.wdb*, and so on, with the current simulation always retaining its name. With each simulation the *_sim#* is increased by one.

Caution

This functionality only occurs if EZwave is displaying a simulation result when a simulation is rerun. If this is not the case, previous results are automatically overwritten.

Cursor Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose **Cursors** from the EZwave Preferences list on the left.

Used to specify cursor display options.

Objects

Table 8-27. EZwave Display Preferences - Cursor Options

Field	Description
Cursor Tab:	
Cursor Precision	
Number of Digits of Precision	Specifies the degree of precision in value flags attached to cursors. Select from 1 to 9 digits of precision. Also specifies the precision of displayed Measurement Tool results.
Auto-Adjust Values for Cursors	Specifies whether cursor values are automatically adjusted to show the most significant numbers, regardless of axis scale. For example, 0.000005183 is displayed as 5.183e-6.
Cursor Y Units	
Show Y Units of Cursor Value	Specifies whether units are displayed with the values listed in the cursor flag.
Show Parameters	Specifies whether parameters and their values are listed in the cursor flag.
Cursor Data Values	
Show/Hide Dynamically Cursor Values Based on Waveform Selection	Specifies whether, when a waveform is selected, to show and hide cursor values for that waveform. Whether it is shown or hidden is specified by right-clicking on the cursor and choosing Data Values > Show Selected Mode, or Data Values > Hide Selected Mode.
Append/Remove Dynamically Cursor Values Based on Waveform Selection	Specifies whether, when a waveform is selected, to add and remove that waveform from the set of cursor values shown at a particular cursor. Whether it is added or removed is specified by right-clicking on the cursor and choosing Data Values > Append Selected Mode, or Data Values > Remove Selected Mode.
Show Only Cursor Values Of Selected Waveforms While Creating Cursor	Specifies whether to show only the values of selected waveforms when a new cursor is added. The values for other waveforms are hidden.
Cursor X/Y Level Lines	
X/Y Level Lines Displayed by Default	Specifies whether X and Y level lines are displayed.

Table 8-27. EZwave Display Preferences - Cursor Options (cont.)

Field	Description
Cursor Area	
Use Cursor Area	Specifies whether cursor data is displayed near the waveform names area rather than in cursor data flags.
Cursor Delta X/Y	
Cursor Based	Specifies that the Y deltas are calculated from the Y-level lines of a single cursor. Each Y-level's delta Y value is calculated from the base Y-level line.
Waveform Based	Specifies that the Y deltas are calculated based on the same waveform between the Y-level lines of two or more cursors. Each cursor's delta Y value is calculated from the base cursor.
Cursor Legend	
Draw on Top of X Axis	Displays the X value base cursor below the x axis, instead of on top of it.
Horizontal Cursor in Y Axis	
Show in Only One Y Axis (Where Cursor Had Been Defined)	Specifies that horizontal cursors are added only to the currently selected waveform row.
Show in All Compatible Y-Axes (Y-Axes Have Same Unit)	Specifies that horizontal cursors are added to every row using the same unit, in the current display.
Data Value Tab:	
Behavior of Data Values With Custom Location As The Cursor Moves	
Lock Data Value X Location	When cursor is moved, the X location of the flag remains constant. This option is not selected by default.
Lock Data Value Y Location	When cursor is moved, the Y location of the flag remains constant. This option is selected by default.

Data Format Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Data Format from the EZwave Preferences list on the left.

Used to specify global scaling, notation and formatting for the waveform display.

Objects

Table 8-28. EZwave Display Preferences - Data Format Options

Field	Description
Axis Data Units	
Data Type	List of all known waveform data types.
Scaling	Specifies how many units should be used for the axis labels. For example, if the waveform data has a periodic interval of 33, then the x-axis tick labels might be 33, 66, 99. To have the labels display the number of periods, you would specify the scaling value to be 33. Then the tick labels would be 1, 2, 3.
Units	Specifies the units to be used for each data type.
Y Axis Options	
	Specifies how y-axis units are plotted when waveforms with different y-axis units are plotted in the same Graph row. Available options: • **Overlay All Y Axis Units** — Specifies that certain sets of units are displayed on the same y axis. Select if you want all graph rows to have a single y axis, regardless of the units. • **Overlay Only the Following Y Axis Units** — Specifies groupings of compatible units.
Define Compatible Units	The **New** button enables you to create a new grouping of compatible units. The box in the **Apply It** column specifies whether a compatible unit grouping is on or off. The **Edit** and **Delete** buttons enable you to modify existing groupings of compatible units.
Notation	
IEEE Suffix	Specifies that suffixes defined by IEEE are used to format numerical values in the graph window.
SPICE Suffix	Specifies that suffixes defined by the SPICE simulation syntax are used to format numerical values in the graph window.
Engineering Notation	Specifies that values in the graph window are displayed in exponential format; uses an exponent which is a multiple of three.
None	Specifies that no special formatting will be used for numerical values in the graph window.

Table 8-28. EZwave Display Preferences - Data Format Options (cont.)

Field	Description
Bus Format	
Default Bus Radix	Specifies the default radix for displaying bus values: • One’s Complement • Two's Complement • Signed Magnitude • Binary • Hexadecimal • Octal • Unsigned Decimal • Ascii • Fixed Point • Gray Code to Decimal • To Gray Code • Unary (Thermo Code)
Bus Index Delimiters	Specifies the characters that EZwave will interpret as delimiters of a bit index when a bus is created using **Tools > Create Bus**, or **Plot As Bus**. By default, this field will contain (, [, <, {, _, and -.
Use Separator Character	Specifies the character to separate values in the bus. The default value is ‘,’ (comma). When disabled, values are displayed separately, similar to the way they are displayed in the Questa SIM Viewer.
Double Format	
Double Format Precision	Specifies the level of precision displayed. You can select from 1 to 9 digits of precision, or full precision.

Foreign Databases Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose **Foreign Databases** from the EZwave Display Preferences list on the left.

Use this dialog box to specify loading and display options for foreign databases.

Objects

Field	Description
Foreign Databases	
Load Scalars	Specifies that Scalars (parameters or measurements) are loaded.
Display Measures in Waveform List	Specifies that Measures are displayed in the Waveform List.
PSF Info Files	
Load Info Files	Specifies that PSF *.info* files are loaded from output directories. **Note:** Loading *.info* files may require considerable time and memory. **Note:** The *designParamVal.info* file is always loaded in EZwave.
Identify Info Files: -By File Extension -By analysisType Field in File	When Load Info Files is checked: - all PSF files ending in *.info* are loaded in EZwave or - all PSF files with analysisType field set to info are loaded in EZwave.
Detect DC Hysteresis Waveforms	Enables or disables the DC Hysteresis detection. When disabled, data points are sorted.
Nutmeg Files	
Standardize X-Axis	Specifies how the x axis is handled.
VCD Files	
Read Port Strength from Extended VCD Files	Enables reading of the Verilog port signal state and strength from extended VCD files as verilog_logic. When disabled, port values are converted to std_logic.

General Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose General from the EZwave Display Preferences list on the left.

Used to set options related to opening files, updating waveforms, and exiting the EZwave viewer.

Objects

Table 8-29. EZwave Display Preferences - General Options

Field	Description
General Options	
Overlay by ordinate data type	With this option selected, waveforms are overlaid using a separate row for each different y-axis data type. With this option unchecked, all waveforms are plotted overlaid in a single row with multiple y axes as required. Refer to **Overlay Window Waveforms** in the Row Popup Menu and Axis Popup Menu.
Show Grid Lines in Newly Opened Windows	Specifies whether graph windows display grid lines by default when opened.
Show Zero-Level Lines in Newly Opened Windows	Specifies whether zero-level lines are turned on by default when new graph windows are opened.
Show List of Recent Databases in File Menu	The **Clear List** button clears the MRU information from memory. When turned off, the MRU information is kept in memory, but not displayed.
Activate Undo/Redo	Specifies whether the **Undo** and **Redo** functions from the **Edit** menu are available.
Use AMS Results Browser to open AEX Files	With this option selected, scalar results of extracts are displayed in the AMS Results Browser (see the AMS Results Browser User's Manual for information). With this option not selected, or for users on Windows, the scalar results are shown in EZwave.
Use Automatic Waveform Selection for Measurement Tool	With this option unchecked, selected waveforms are not automatically added to the Source Waveform(s) list on the Measurement Tool dialog box.
Use Source Waveform Properties for Measurement Tool New Waveforms	With this option selected, the source waveform color, point style, line width, and style are used to plot the result waveforms.
Allow Filename Editing in File Browser	With this option unchecked, filenames cannot be edited in the File Browser.

Table 8-29. EZwave Display Preferences - General Options (cont.)

Field	Description
View All Activates Auto-Range	When using View All (accessed via **View** > **View All** or the Row popup menu), the ranges of the axes are automatically set with this option selected. With this option unchecked, **View All** uses the specified Axis Range settings where Auto Range has been unchecked on the Y-Axis Properties or X-Axis Properties dialog boxes.
Require Confirmation Before	
Exiting	Specifies whether a confirmation dialog box is shown when exiting EZwave. Turning off this option will not prevent loss of data, because if there are unsaved waveform databases a separate confirmation dialog box displays.
Closing in-use Database	Specifies whether a confirmation dialog box is shown when attempting to close a database that is currently in use.
Closing Modified Database	Specifies whether a confirmation dialog box is shown when attempting to close a database that has been modified.
Closing Eye Mask Dialog	Specifies whether a confirmation dialog box is shown when attempting to close the Eye Mask Dialog Box.
Deleting All Cursors/ Markers	Specifies whether a confirmation dialog box is shown when selecting **Delete All Annotations** from the **Annotations** popup menu.
Marching Waveforms	
Automatically Update Displayed Waveforms Every *N* Interval	Specifies whether waveforms are automatically updated from a running simulation at the time interval specified. The frequency of the interval will impact the simulators performance. The minimum recommendation is one minute.
Automatically Update displayed Windows Every *N* % of Simulation	Specifies whether waveforms are automatically updated from a running simulation at the percent completed of the simulation.
Open Window	
Break on Error	Specifies that the behavior in case an error is found when loading an swd file. The swd file is displayed if an error is found, and the line containing the error is highlighted. You can then choose to: • Skip All Errors • Skip Error • Abort

Table 8-29. EZwave Display Preferences - General Options (cont.)

Field	Description
Ignore All Errors	Specifies that the behavior in case an error is found when loading an swd file. The file loads completely, regardless of errors. This can cause unexpected behavior in the waveforms.
Graphical Memory Management	
Graphical Memory Threshold	Specifies the maximum amount of memory (in megabytes) that is set aside for the graphics engine. If this limit is reached, a warning dialog box will ask whether you want to allocate more memory.

Grid Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Grid from the EZwave Display Preferences list on the left.

Used to set the appearance and behavior of the grid in graph windows, overriding the automatic settings. The Default button returns the Grid Spacing option to Automatic, and returns all manually entered values to their default values.

Note

These options only apply for linear axis scales; they have no effect on logarithmic axes.

Objects

Table 8-30. EZwave Display Preferences - Grid Options

Field	Description
Grid Spacing	
Automatic	Graph windows use the default system grid behaviors.
User Defined	Graph windows use he settings entered on this dialog box.
Grid Units	
Pixels	Enabled by default. The grid is drawn based on the following two values, measured in pixels from the graph origin: • Grid Offset—defines the distance from the origin to the first grid line. Must be a positive value. • Grid Period—defines the distance between adjacent grid lines. Must be a positive value.
Pixels Predefined Axis Data Units	The grid is drawn based on values specified for the unit type the axis represents. The grid offers each data type configured in the Axis Data Units grid of the Data Format Options, and for each type, you can specify the following values: • Grid Offset—defines the distance from the origin to the first grid line. Must be a positive value. • Grid Period—defines the distance between adjacent grid lines. Must be a positive value.
Axes Minimum Grid Spacing	
X Axis	Specifies the minimum distance between adjacent grid lines for the x axis. Default is 25. Must be a positive value.
Y Axis	Specifies the minimum distance between adjacent grid lines for the y axis. Default is 25. Must be a positive value.
Grid Anchor	

Table 8-30. EZwave Display Preferences - Grid Options (cont.)

Field	Description
Minimum Absolute Axis Value	When enabled, the grid and all of its parameters (such as Grid Offset) are calculated and plotted starting from the minimum possible (absolute) x and/or y-axis values. For example, the absolute range for the x axis is from 0.0M to 100.0M; when the wave window is zoomed, the current (visible) range of the axis changes, but the grid is still calculated using the absolute range (its minimum value, which is 0.0M).
Minimum Visible Axis Value	When enabled, the grid and all of its parameters (such as Grid Offset) are calculated and plotted starting from the currently visible minimum x and/or y-axis values.

Layout Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose **Layout** from the EZwave Display Preferences list on the left.

Use this dialog box to specify layout options for the graph window.

Objects

Table 8-31. EZwave Display Preferences - Layout Options

Field	Description
X-Axis	
Above Waveforms	Specifies that the x axis is displayed above the waveforms.
Below Waveforms	Specifies that the x axis is displayed below the waveforms. Default.
Waveform Names	
Left of Waveforms	Specifies that waveform name labels are displayed on the left of the waveform.
Right of Waveforms	Specifies that waveform name labels are displayed on the right of the waveform.

Look and Feel Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose **Look & Feel** from the EZwave Display Preferences list on the left.

Use this dialog box to specify the look and feel of the EZwave main display and its dialog boxes.

Objects

Table 8-32. EZwave Display Preferences - Look and Feel Options

Field	Description
Light	Specifies the "light" theme.
Gray	Specifies the "gray" theme (default).
Dark	Specifies the "dark" theme.
Classic	Specifies the "classic" theme (as used in images in this manual).

Usage Notes

After changing the look and feel, you need to restart EZwave to apply the new theme.

Mouse Pointer Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Mouse Pointer from the EZwave Display Preferences list on the left.

Use this dialog box to specify options for your mouse pointer and mouse strokes.

Objects

Table 8-33. EZwave Display Preferences - Mouse Pointer Options

Field	Description
Mouse Pointer Options	
Show Tooltips	Specifies whether popup tooltip windows are used.
Change Color of Objects When Under Pointer	Specifies whether graph window objects, such as axes or waveform names, change color when the mouse pointer is over them.
Mouse Strokes	
About Mouse Strokes	Gives a quick overview on mouse strokes without having to refer to the Help system.
Color to Use When Drawing Mouse Strokes	Specifies the color that will appear when drawing mouse strokes. The Select button enables a different color to be chosen.
Choose the Key Modifier to Use with the Left Mouse Button ...	For systems that do not have a middle mouse button, specifies one or more keyboard modifiers to use with the left button instead. Available options: • Alt • Control • Shift
Require Confirmation Before Executing a Close Stroke	Specifies that a confirmation dialog box is shown when the close stroke is executed.
Mouse Drag Options	
Highlight Drop Target Areas	Highlights the target drop area for a waveform as you drag it around.
Highlight Color	Specifies the color of the highlight area rectangle that appears when dragging a waveform. The Select button enables a different color to be chosen.

Multiple Run Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Multiple Run from the EZwave Display Preferences list on the left.

Use this dialog box to specify display, name and color options relating to compound waveforms.

Objects

Table 8-34. EZwave Display Preferences - Multiple Run Options

Field	Description
Compound Waveform Display	
Display as Compound	Specifies that waveforms are plotted as compound waveforms.
Expand Single Element Names	Only available when Display as Compound is selected. Specifies that the name for each individual element (run) may be displayed in the name area.
Display as Single Elements	Specifies that compound waveforms are “unwrapped” into their individual elements (sub-waveforms, each made by a single run) and then plotted showing each individual element overlaid with its own color. The individual element names are also shown in the name area.
Show Single Element Names With Run Parameters	Specifies that run parameter names and values are displayed in the graph windows and also in the Waveform List panel (Tree view).
Color-Code Each Single Element	Specifies that a single color is used for all waveforms from the same simulation run. When not selected: • For Display as Compound, the same color is used for all single element runs of a compound waveform • For Display as Single Elements, an individual color (not related to the simulation run) is used for each element.
Eye Diagram	
Plot	• Overlaid - (Default) Specifies that eye diagrams for each element in the compound waveform are overlaid on the same graph. • Stacked - Specifies that separate eye diagram graphs for each element in the compound waveform are displayed stacked.

Table 8-34. EZwave Display Preferences - Multiple Run Options (cont.)

Field	Description
Eye Parameters	• Calculate From First Single Element - (Default) Specifies that settings are calculated for the first element in the compound waveform and applied for all eye diagrams for other elements. • Calculate For Each Single Element - Specifies that individual default settings are calculated and applied for each element in the compound waveform.

Usage Notes

When a multiple run is displayed as a compound waveform, any measurements made, for example using the Waveform Calculator, are done using the entire set of runs that make up the compound waveform.

When a multiple run is displayed as single elements, be aware of the following:

- If a measurement is taken on a run, this measurement will not be taken automatically for subsequent simulated runs.
- If a user display is set for a run, it is not applied to subsequent simulated runs.

Pick Points Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Pick Points from the EZwave Display Preferences list on the left.

Use this dialog box to control settings related to pick points.

Objects

Table 8-35. EZwave Display Preferences - Pick Points Options

Field	Description
Maximum Number of Visible Pick Points	The maximum number of simultaneously visible pick points per plotted waveform in the graph window. Default is 2.
Pick Point Symbol	Specifies the graphic symbol to use when adding pick point markers to waveforms. Can be: • Cross (default)  • Rounded Cross  • Dot 
Follow Waveform(s)	When enabled, the mouse pointer is attached to the waveform nearest to it, and can only be moved around in the graph window in the selected mode: Snap to Data Points or Interpolate. When not enabled, the mouse pointer can be moved freely within the row. Pick points will be placed at the closest point (real or interpolated) on the waveform(s) in the row, starting with the waveform nearest to the position of the mouse-click.
Snap to Data Points	The pick pointer is attached to the current closest waveform in the waveform row. It can be moved only to the position on the waveform, where a data point is present. Pick points are placed on the waveform(s) at the data point closest to the mouse-click position. This is the default setting. If multiple waveforms are present in the same row, one pick point will be placed on each waveform in the row at the closest to the mouse-click position data point, starting from the nearest waveform to the mouse-click position.

Table 8-35. EZwave Display Preferences - Pick Points Options (cont.)

Field	Description
Interpolate	The pick pointer is attached to the current closest waveform in the waveform row. It can be moved anywhere on the waveform. Pick points are placed on the waveform(s) at the interpolated point closest to the mouse-click position (minimal distance on x and y axes). If multiple waveforms are present in the same row, one pick point will be placed on each waveform in the row at the closest interpolated point to the mouse-click position, starting from the nearest to the mouse-click position on the waveform. This setting is not used when waveforms use scattered or spectral drawing modes. In this case, Snap to Data Points mode is used.
Use Mouse Pointer Axis	Specifies how the mouse pointer is displayed in Pick Points mode and how pick points are added in the graph window: • X&Y (default) Both x and y bars and corresponding coordinates of the mouse position are used to calculate the closest data point and to add the pick point(s). Pick points will be added only on the waveforms plotted in the row which is currently under the mouse pointer position. • X Only the vertical bar (X coordinate) of the mouse pointer is used to calculate the closest data point and to add the pick point(s). If this mode is selected, one pick point will be added at the same X coordinate of each waveform in the active graph window (not just the row under the mouse pointer). If X&Y axes are selected for digital waveforms, the horizontal bar of the mouse pointer will be displayed and moved according to the digital waveform events (high, middle, low) but only the W coordinate will participate in the closest data point calculation.
If Displayed as Compound	Controls how pick point markers are added to compound waveforms: • Add Pick Point to The Closest Element Adds one pick point to the nearest data point on the compound element closest to the mouse-click position. • Add Pick Point to All Elements Adds one pick point to the nearest data point of each compound element starting from the one closest to the mouse-click position.

Table 8-35. EZwave Display Preferences - Pick Points Options (cont.)

Field	Description
Pick Points Dialog Data Precision	Specifies the global data precision to use in the Pick Points dialog box: • Use Cursor Number of Digits of Precision (the default) Uses the precision value set in the Number of Digits of Precision field, under Cursor Options. • Use Double Format Precision Uses the precision value set in the Double Format Precision field, under Data Format Options.
Default	Returns pick point settings to their defaults.

RF Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose RF from the EZwave Display Preferences list on the left.

Use this dialog box to specify RF settings for the waveform display.

Objects

Table 8-36. EZwave Display Preferences - RF Options

Field	Description
Smith Chart	
Plot Sxx-Parameters in a Smith Chart	Specifies that S11 and S22 (scattering) parameters will automatically be displayed in a Smith Chart. Enabled by default.
Always consider waveforms as Sxx-parameters	Specifies that waveforms in the Smith Chart are displayed as the normalized impedance of a Sxx parameter. Otherwise, waveforms are not considered as Sxx parameters and input waveforms are simply displayed in the Smith Chart plane. Enabled by default.
Smith Chart Type	Specifies the default Smith Chart type as Impedance, Admittance, or Impedance and Admittance (both grids overlaid on a single chart).
Smith Chart Cursors	
Snap to Data	Specifies that the cursor automatically snaps to the nearest data point on a Smith Chart.
Impedance / Admittance Display	Specifies the display as normalized (default) or to use the characteristic impedance.
Stability / Noise Circles	
Hide Stability / Noise Circles	Specifies that circles are hidden by default.
Polar Chart	
Plot Sxy-Parameters in a Polar Chart	Specifies that S12 and S21 (scattering) parameters are displayed in a polar display of a Smith Chart. Enabled by default.
Polar Chart	The complex-valued waveform is plotted.
Polar Chart Display	Specifies the polar chart to be displayed in degrees (default) or radians.

Row Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Row from the EZwave Display Preferences list on the left.

Use this dialog box to specify row height and position settings for the waveform display.

Objects

Table 8-37. EZwave Display Preferences - Row Options

Field	Description
Row Height	
Analog Row Height	Specifies the height for the analog rows as one of the following: • Automatic — Specifies that the row height is determined based on the content of the row. Specify the Minimum Size for a row in pixels. • Fixed — Specifies that the rows are set at a fixed height. Specify the fixed Size in pixels.
Digital Row Height	Specifies the height of digital rows in pixels.
Space Between Rows	
Minimize Space Between Rows	Specifies that the distance between rows is reduced. This causes the rows to be more closely spaced to increase the number of rows visible at one time. By default, this is turned off.
Y Axis Range	
Ignore Questa ADMS Uninitialized Values	Specifies that uninitialized Questa ADMS values are ignored when calculating the y-axis range for plots.
Row Title Appearance	
Hide Row Title Outline	Controls the visibility of the text box around the row title.

Usage Notes

Tip

You can force EZwave to use the Minimum Size analog row height set on the Row Options dialog box for all plotted waveforms. Choose **View** > **Min Row Height Plot Mode** before plotting. This option is also available on the Toolbar, and is useful if you need to fit a large number of waveforms in the display.

In VHDL-AMS, un-initialized values default to the following values:

- 1.0e308 for "real" signals
- 2147483648 for "integer" signals

- 9223372036854775807 for "time" signals (femto seconds)

When plots are generated, these uninitialized values at $t = 0$ may cause the plot to have a y-axis range that is too large in relation to the waveform. By selecting Ignore Questa ADMS Uninitialized Values, these values are ignored when calculating the y-axis range for plots.

Note

These uninitialized values are only ignored when calculating the y-axis range of plots. They will still exist in the waveform data (adding a cursor at the location will display the value) as it may be useful to know that there are uninitialized values in the waveform.

Save Data Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Save Data from the EZwave Preferences list on the left.

Use this dialog box to specify global settings related to saving waveform data.

Objects

Table 8-38. EZwave Display Preferences - Save Data Options

Field	Description
Save to Ascii Options	
Field Separator	Specifies the character to use as a field separator when saving a waveform to an ASCII file. Available options: • **Predefined** — A dropdown list of available predefined field delimiters (Tab, Space, or a comma [,]). • **User defined** — A text box to define your own by entering any series of characters.
Number of Digits of Precision	Specifies the size of the output file. Select the level of precision in the save file with the dropdown list. You can select from 1 to 9 digits of precision, or full precision.
Save Bus as Separated Bits	Specifies that buses are saved as separated bits.
Bus Radix	Specifies that the bus value is saved instead of the values of the individual bits composing the bus. This option is only available when the Save Bus as Separated Bits option is unchecked. The radix can be one of the following: • One’s Complement • Two's Complement • Signed Magnitude • Binary • Hexadecimal • Octal • Unsigned Decimal • Ascii • Fixed Point • Gray Code to Decimal • To Gray Code • Unary (Thermo Code)

Table 8-38. EZwave Display Preferences - Save Data Options (cont.)

Field	Description
Use “Re:” and “lm:” for X/Y Plot	Enables the old behavior for X and y-axis labeling when data is saved to an ASCII format. The labels Re: (Real) and Im: (Imaginary) are used instead of the X and y-axis units.
Spice Pwl Options	Sets the default conversion for untyped waveforms when saving a database as a SPICE PWL file (*.sti*). Can be None, Voltage or Current. EZwave only saves time-based current and voltage waveforms in *.sti* files. Choose None to save waveforms only of type voltage or current. For other waveforms, select either Voltage or Current to perform a unit conversion before saving the waveform.
Show Option Dialog Before Saving	The Save As Dialog Box is displayed by default when a database is saved, to enable changes to the save options (for example, X range setup or data sampling) to be specified.

Save Window Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Save Window from the EZwave Preferences list on the left.

Use this dialog box to specify global settings related to saving windows.

Objects

Table 8-39. EZwave Display Preferences - Save Window Options

Field	Description
Save Window Management	
Save Active Window	Specifies that only the active window is saved during **File** > **Save**.
Save All Windows	Specifies all windows are saved during **File** > **Save**.
Save Related Database (Only for *.swd* File)	Specifies that when you save the active graph window to a *.swd* file, EZwave creates and saves a new *.wdb* database file with the waveforms that are plotted in the saved graph window.
Show Option Dialog Before Saving	The Save Windows Dialog Box is displayed by default when a window is saved, to enable changes to the save options to be specified.
Save in TCL	Specifies that windows are saved in TCL format
Save in SWD	Specifies that windows are saved in SWD format.
Save As Paths	
Use Absolute Paths	An absolute path is used to point to dataset (*.wdb*) files within the TCL or SWD scripts that are generated when a window is saved.
Use Relative Paths	A relative path is used to point to dataset (*.wdb*) files within the TCL or SWD scripts that are generated when a window is saved. Paths can be: • Relative to AMS_EZDO_ROOT The AMS_EZDO_ROOT environment variable. This variable must be defined when the TCL or SWD script is executed. • Relative to Script Location

Text Annotation Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Text Annotation from the EZwave Preferences list on the left.

Use this dialog box to control the default display options for text annotations.

Objects

Table 8-40. EZwave Display Preferences - Text Annotation Options

Field	Description
Annotation Display Options	
Hide Anchor Symbol	Hides text annotation anchor symbols when the annotation is attached to a waveform.
Anchor Symbol	Selects the symbol used at the anchor point of the text annotation when the annotation is attached to a waveform. Can be either Crosshair or Dot.
Hide Anchor Line	Hides the line between the anchor point and the text annotation box when the annotation is attached to a waveform.
Hide Text Outline	Hides the outline of the text annotation box.
Show Vertical Line at Anchor Position	Specifies that a vertical line is displayed across the waveform at the selected position.
Auto-Adjust Annotation Text Location When Outside Waveform Area	Changes the location of the waveform text annotation for better visibility, so that it is not truncated if outside the waveform area.
Add Annotations Options	
Expand Digital Row Height	Specifies that the default digital row height is expanded when annotations are added to a digital waveform.

Transformation Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Transformations from the EZwave Display Preferences list on the left.

Use this dialog box to specify the transformations that are applied when waveforms are plotted.The options on this dialog box are organized on two tabs, **Transformations** and **Conversion**.

Objects

Table 8-41. EZwave Display Preferences - Transformations - Transformations Tab

Field	Description
Supported Complex Waveform Transformations	
dB	The magnitude of each point of the complex-valued waveform calculated in decibels (20 * log (waveform))
Magnitude	The square root of (re^2 + im^2) for each point of the complex-valued waveform.
Real	The real component of each point in the complex-valued waveform.
Imaginary	The imaginary component of each point in the complex-valued waveform.
Phase	The phase of each point in the complex-valued waveform. All phase angles are between -180 degrees and 180 degrees (-PI radians and PI radians).
Continuous Phase (Cphase)	The phase of each point in the complex-valued waveform, including an accumulated phase angle from the previous points. As this transformation does not include a phase discontinuity at the 180 degree boundaries, unlike the Phase transformation, this transformation is useful when analyzing waveforms with more than +/- 180 degrees of phase shift.
Complex Plane	The complex-valued waveform of a collection of complex numbers (z=a+ib, where i is the imaginary number) plotted as a curve (not necessarily monotonic) of the unique points (a,b) in the complex plane.
Smith Chart	The complex-valued waveform is plotted on a Smith Chart.
Polar Chart	The complex-valued waveform is plotted in a polar display of a Smith Chart. See “Smith Chart and Polar Displays” on page 223.
Selected Transformation Should be Plotted Overlaid	If more than one transformation is selected, specifies that all of the transformed waveforms are plotted in the same row (overlaid). This option does not apply to complex-plane plots, polar charts and Smith Charts.

Table 8-41. EZwave Display Preferences - Transformations - Transformations Tab (cont.)

Field	Description
Apply Transformation Before "Plot Difference"	This option controls how expressions VX(a,b) and IX(a,b) are computed in AC analysis. X stands for DB (magnitude, in decibels), M (magnitude), P (phase) or GD (group delay). • Selected (Default)—VX(a,b) or IX(a,b) are computed from the complex value v(a)-v(b) or from i(a)-i(b). For example, VDB(1,2) is computed as DB(V(1)-V(2)). • Unselected (Default in -compat mode)—VX(a,b) or IX(a,b) are computed from the complex value VX(a,0)-VX(b,0) or from IX(a,0)-IX(b,0). For example, VDB(1,2) is computed as VDB(1)-VDB(2).
Supported Digital Waveform Transformations	
New	Click to add a new digital waveform transformation. This invokes the Edit Digital Transformation dialog box. Type the transformation name, source type, and destination type in this dialog box.
Edit	Select a transformation and click to change the transformation details. This invokes the Edit Digital Transformation dialog box.
Delete	Select a transformation and click to remove that transformation.

Table 8-42. EZwave Display Preferences - Transformations - Conversion Tab

Field	Description
Auto-Compute Threshold	When selected, the default threshold calculation is used for the **Plot as Bus** and **Plot as Digital** waveform popup menu items. When unchecked, thresholds can be entered manually.
Threshold	If the bus is an analog or hybrid bus, the analog signals are automatically transformed to digital. This option specifies the thresholds to digitize the input waveform(s) as one of the following: • Single Threshold • Two Thresholds Specify the desired threshold values.

Usage Notes

The Apply By Default checkboxes specify one or more transformations that are automatically applied when a complex-valued waveform is plotted.

You can also apply the transformation directly by right-clicking a waveform and choosing a transformation from the **Transformations** menu item.

Waveform Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Waveform from the EZwave Display Preferences list on the left.

Use this dialog box to specify how analog and wreal waveforms are displayed in EZwave graph windows.

Objects

Table 8-43. EZwave Display Preferences - Waveform Options

Field	Description
Analog Waveforms	
Extend last analog waveform data point to infinity	Specifies that the last data point is extended to infinity. Extending would cause the last data point in the result waveform to extend horizontally to infinity (as well as the reference waveform).
Limit the Number of drawn Data Point symbols	Reduces the number of data points drawn on the screen, to speed up drawing time when there are a large number of points. Although not all of the points are drawn, those that are visible are all real data points. The default.
Wreal Waveforms	
Display as Step Waveforms	Displays wreal waveforms as “step” waveforms, similar to real waveforms. With “step” waveforms the ‘X’ and ‘Z’ states are displayed as colored rectangles.
Display as Railroad Waveforms	Displays wreal waveforms as “railroad” waveforms.
Eye Diagram Tool	
Update Eye Parameters When Waveform Added	Updates the Eye Parameters on the Eye Diagram Tool - Settings Tab dialig box when a waveform as added. The default. If the added waveform is not an eye diagram, the Eye Diagram Tool tries to calculate the default eye parameters (except eye measurements and mask). If the added waveform is an eye, the tool tries to load or calculate all possible parameters (including eye measurements and mask).

Waveform Compare Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose **Waveform Compare** from the EZwave Display Preferences list on the left.

Use this dialog box to specify options for waveform comparison.

Objects

Table 8-44. EZwave Display Preferences - Waveform Compare Options

Field	Description
Show Existing & New Together	Updated waveforms are shown overlaid over the previous result.
Replace	The existing plotted waveforms are replaced with the new result.
Update Target	Selects which waveforms to apply the update to: • All Windows The waveforms in all open graph windows are updated • Active Window Only The waveforms in the currently active graph window are updated • Plotted Waveforms Only the waveforms specified in the Plotted Waveforms List are updated.
Allow Partial Match	Specifies that when comparing waveforms, waveforms that partially match the names within the database are updated.

Waveform List Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Waveform List from the EZwave Display Preferences list on the left. See Waveform List Panel.

Use this dialog box to specify options for the Waveform List Panel.

Objects

Table 8-45. EZwave Display Preferences - Waveform List Options

Field	Description
Waveform List Panel	
Use Following View by Default	Specifies the default view for the Waveform List Panel, as either Tree View (default) or List View.
Separate Tree View Into Structure and Waveform List Views	Specifies that the Tree View is divided into two panels. The top panel is called the Structure List and displays the hierarchical structures of the waveform database. The bottom panel is called the Waveform List and displays the list of waveforms that belong to the selected structures. This option is enabled by default.
Plot Hierarchical Nodes Stacked When Dragging and Dropping	Specifies that when you drag and drop a hierarchical node from the Structure List to the workspace or a graph window, the waveforms are stacked. If this option is cleared, plotted nodes are overlaid. By default, waveforms are stacked.
Show Only Databases Related to Active Workspace	Specifies that only the databases that are related to the currently active workspace are listed. By default, all databases are shown.
Sort the Waveform List in a Case-Insensitive Way	Specifies that case is disregarded when sorting the Waveform List. By default, the alphabetically-sorted Waveform List considers case.
Show Hidden Waveforms	Specifies that hidden waveforms are shown in the Waveform List panel. By default, waveforms that are hidden in the databases are not shown in the Waveform List panel. Showing hidden waveforms in the Waveform List panel does not make them visible in the graph windows.
Disable Automatic Search While Typing Text	Specifies whether to search the waveform lists while you type in the search text fields. This text field displays in the Waveform List panel and by default, automatic searching is enabled. While automatic searching is enabled, it will show only waveforms that match the string (or partial string) that you type while you type. If automatic searching is disabled, you must press Enter to perform the search.
Show Pin Direction Filters Panel	Specifies that the pin direction filter icons are shown in the Waveform List panel.

Table 8-45. EZwave Display Preferences - Waveform List Options (cont.)

Field	Description
Highlight Plotted Waveforms on Selection in the Waveform Tree	Specifies that when you select a waveform name from the Waveform Tree panel with a single click, any already plotted waveforms with that name are highlighted.

Workspace Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then choose Workspace from the EZwave Display Preferences list on the left. Or, right-click in your workspace to display the Workspace Popup Menu and choose Options.

Use this dialog box to specify options relating to your workspace.

Objects

Table 8-46. EZwave Display Preferences - Workspace Options

Field	Description
Workspace Options	
Allow Multiple Workspaces	Specifies whether multiple workspaces can be created. When turned off, the Workspace Popup Menu will not show the **New** menu item and the **Workspace** tab area is removed.
Hide Icons for Minimized Windows	Specifies whether minimized graph windows appear as icons on the workspace. When turned off, minimized windows don't appear anywhere on the workspace and you will not be able to double-click the icon to restore the window. Click the window name in the Window menu or the Graph Window button on the workspace taskbar to restore it.
Hide Taskbars	Specifies whether the workspace taskbar is visible. If the taskbar is not visible, you can activate windows either by clicking directly on them, or by clicking the window name in the Window menu.
Show Close Workspace Tab Button	Specifies whether the workspace tabs have a close (X) button. You can also delete a tab by right-clicking on the tab and selecting **Delete**.
Location for Tabs	Specifies the location of the workspace selection tabs. Select the desired location from the dropdown list.
Workspace Background	
Solid Color	The workspace area will be a solid color.
Tiled Image	The workspace area will be a tiled image.
Select	Used to specify the color or image to use for the workspace area.

CDF Plot Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the CDF folder on the left side, and choose Plot.

Used to specify default options when plotting a CDF (Cumulative Density Function).

Objects

Table 8-47. EZwave Display Preferences - CDF Plot Options Dialog Box Contents

Field	Description
CDF Properties	
Plot Predefined cdf (when available in file)	Specifies that the cdf stored in the simulation file (for example, a cdf() waveform generated by Eldo on an extract) should be plotted instead of a new cdf being calculated. With this option selected, all CDF Customization options are ignored.
CDF Customization	
Minimum X Value	Specifies the X value at the lower bound of a window interval.
Maximum X Value	Specifies the X value at the upper bound of a window interval.
Sample Data	When selected, the input waveform is sampled (the number of points remains the same, but are equally distributed; see the histogram function for details). When not selected, the raw Y values of the input waveform are used.
Shift Data so that Mean Equals 0	When selected, the Y-values of the input waveform are decreased by the mean() of these values, so the displayed mean equals 0.
Multiply Data so that Standard-Deviation Equals 1	When selected, the Y-values of the input waveform are decreased by the mean() of these values, and then divided by the standard-deviation. So the displayed mean equals 0, and the displayed standard-deviation equals 1.0. This option is available only if "Shift Data so that Mean Equals 0" is selected, above.
Companion Waveforms	
Plot Gaussian Distribution	When selected, the integral of the Gaussian distribution is plotted based on the mean and standard-deviation of the input waveform. The range for this distribution is between mean +/- N*stddev, with the following constraints: • N is the smallest integer so that the Y min() and max() of the input waveform are in the range • N is greater than or equal to 4 • N is less than or equal to 20

Table 8-47. EZwave Display Preferences - CDF Plot Options Dialog Box Contents (cont.)

Field	Description
Plot PDF (when available in file)	Plots the integral of the PDF (potential density function) associated to the input waveform. This functionality is available in *.wdb* files generated by an Eldo Monte Carlo simulation.
CDF is a Step Waveform	When selected, the CDF is plotted as a step waveform.
Plot Confidence Bounds (when available in file)	When selected, the plots upper and lower confidence bounds, if these have been computed by Eldo.

CDF Measures Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the CDF folder on the left side, and choose Measures.

Use this dialog box to specify the measures to display when plotting a CDF.

Objects

Table 8-48. EZwave Display Preferences - CDF Measures Options Dialog Box Contents

Field	Description
Single Measures	
Draw Average	Select to draw a vertical line at the average location.
Draw Mean	Select to draw a vertical line at the mean location.
Draw Median	Select to draw a vertical line at the median location.
Draw Nominal Value (when available in file)	When selected, and when the nominal value is stored in the waveform database file, this option draws a vertical line at the nominal location. It is available in *.wdb* files generated by an Eldo Monte Carlo simulation.
Twin Measures	
Draw Probability	There are three options: • N-Sigma: Empirical. When selected, the plot is based on the CDF. Specify a comma-separated list of integer N values, between 1 and 7. For probability P of N-Sigma, the range: min point = X of first CDF point where Y ³ P/2 max point = X of last CDF point where Y £ 1 - P/2 • N-Sigma: Theoretical. When selected, the plot is based on the mean and standard deviation of the theoretical Gaussian distribution. Specify a comma-separated list of integer N values, between 1 and 7. A range is drawn for each value, from: mean-N´stddev to mean+N´stddev. • Probability. This is the same computation as N-Sigma: Empirical (above) except that you specify a comma-separated list containing the required probability values, between 0 and 1, or percentage values between 0% and 100%. (For example 0.6827,0.9545,0.98,99%.) The range legend: `Prob <percentage_of_data_in_range>% [<lower_value> ; <upper_value>]`

Table 8-48. EZwave Display Preferences - CDF Measures Options Dialog Box Contents (cont.)

Field	Description
Lower-Bound/Upper-Bound	When either option is selected, a vertical line is drawn, which can be repositioned in the same way as cursors. Depending on the two settings, the bound location is: • As defined in the *.wdb* file, when Predefined Position (when available in file) is selected. It is available in *.wdb* files generated by an Eldo Monte Carlo simulation. • At a user-defined location, if User Defined Position is selected, and the previous option Predefined Position (…) is not selected, above, or the bound value is not stored in the file. The user defined position is relative to mean + <value> * stddev (as N-sigma, but <value> is a real number, not an integer).

Usage Notes

You can edit a plotted CDF, in which case the User Defined Position value is an absolute value, not a relative one.

When both Lower and Upper Bound are displayed, only the Upper Bound legend displays the yield value.

CDF Legend Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the CDF folder on the left side, and choose Legend.

Use this dialog box to configure and display a legend to the statistical measures displayed on a CDF.

Objects

Table 8-49. EZwave Display Preferences - CDF Legend Options Dialog Box Contents

Field	Description
Draw Stat Legend	Enables the display of the legend on a CDF.

Edit Stat Legend	Used to define the values to display in the legend. Type legend items in the following format:
	`<Label>= @(<measure>)`
	For example:
	`Mean= @(mean)`
	All available measures can be added using the Shortcut Buttons. The variable @(name) adds the waveform name to the legend. The large field at the bottom of the dialog box displays a preview of how the legend will be laid out, using placeholder values.

Shortcut Buttons	
Average	Adds the average to the Stat Legend text.
Cp	Adds the process capability to the Stat Legend text, which is equal to: `(upper_bound - lower_bound) / (^6* stddev)` This measure requires that the lower and upper bound are plotted, otherwise it is <undefined>.
CpK	Adds the process capability, assuming that it is not centered. It is equal to: Min(upper_bound – mean) / (3*stddev),(mean – lower_bound) / (3* stddev) This measure requires that the lower or upper bound is plotted, otherwise it is <undefined>.
Kurtosis	Adds the skewness value to the Stat Legend text. This option is available only when the value is available in the wdb file.

Table 8-49. EZwave Display Preferences - CDF Legend Options Dialog Box Contents (cont.)

Field	Description
Name	Adds the name of the input waveform to the Stat Legend text, without database or hierarchy names.
Max	Adds the maximum value to the Stat Legend text.
Mean	Adds the mean value to the Stat Legend text.
Median	Adds the median value to the Stat Legend text.
Min	Adds the minimum value to the Stat Legend text.
Nominal	Adds the nominal value to the Stat Legend text, when available in the waveform database file, otherwise it is <undefined>.
Relative Std. Dev.	Adds the relative standard deviation to the Stat Legend text, which is equal to: • stddev if mean is zero • stddev/mean if stddev is not zero
Size	Adds the number of points of the input waveform to the Stat Legend text.
Skewness	Adds the skewness value to the Stat Legend text. This option is available only when the value is available in the wdb file.
Std. Dev.	Adds the standard deviation to the Stat Legend text.
@	Adds the @ symbol to the Stat Legend text.

Axis Title Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder in the list on the left side, and choose Axis Title.

Used to modify the fonts and colors of axis titles.

Objects

Table 8-50. EZwave Display Preferences - Axis Title Options

Field	Description
Font	Specifies the font used for axis titles.
Font Size	Specifies the font point size used for axis titles.
Font Style	Specifies the font style used for axis titles.
Color	Specifies the color used for axis titles.
Preview	Displays a sample of what axis titles will look like with the current selections.
Default	Restores the font and color of axis titles to the default settings.

Axis Values Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Axis Values.

Use this dialog box to modify the fonts and colors of axis values.

Objects

Table 8-51. Fonts and Colors - Axis Values Options

Field	Description
Font	Used to change the font of axis values.
Font Size	Used to change the font point size of axis values
Font Style	Used to change the font style of axis values.
Color	Used to choose the color of axis values using the Color Selection dialog box.
Preview	Displays a sample of what axis values will look like with the current selections.
Default	Restores the font and color of the axis values to the default settings.

Axis Values (Smith Chart) Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Axis Values (Smith Chart).

Use this dialog box to modify the fonts and colors of Smith Chart axes.

Objects

Table 8-52. EZwave Display Preferences - Axis Values (Smith Chart) Options

Field	Description
Font	
Font Name	To change the font of axis values for Smith Charts, select an available font from this dropdown list.
Font Size	To change the size of axis values for Smith Charts, select a point size from this dropdown list.
Font Style	To change the style of axis values for Smith Charts, select a font style from this dropdown list.
Colors	
Impedance Grids	Controls the color of the impedance grid when displaying a Smith Chart. Click the colored box to open the Color Selection dialog box and make a new color selection.
Admittance Grids	This controls the color of the admittance grid when displaying a Smith chart. Click the colored box to open the Color Selection dialog box and make a new color selection.
Preview	
Preview	Displays a sample of what the Smith Chart grids will look like with the current selections.
Default	Restores the font and color of the axis values for Smith Charts to the default settings.

Calculator Entry Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Calculator Entry.

Use this dialog box to modify the fonts and colors of the Waveform Calculator.

Objects

Table 8-53. EZwave Display Preferences - Calculator Entry Options

Field	Description
Font Size	To change the size of the font of text, select a point size from this dropdown list.
Preview	Displays a sample of what the text will look like with the current selections.
Default	Restores the font size of the calculator entries to the default setting.

Cursor/Marker Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Cursor/Marker.

Use this dialog to specify the default fonts and colors for cursors and markers.

Tip

You can also adjust the fonts, line styles, and colors for each cursor individually. See "Cursor Properties Dialog Box" on page 445.

Objects

Table 8-54. EZwave Display Preferences - Cursor/Marker Options

Field	Description
Font	Controls the font of the values in cursors and markers.
Font Size	Controls the size of the font of values in cursor and markers.
Font Style	Controls the style of cursors and markers.
Y Values	Controls the display color of the Y values associated with cursors and markers. Click the colored box to open the Color Selection dialog box and make a new color selection.
X Values	Controls the display color of the X values associated with cursors and markers. Click the colored box to open the Color Selection dialog box and make a new color selection.
Preview	Displays a sample of what the text will look like with the current selections.
Active Cursor	
Cursor Style	Choose a line style from the dropdown list for the active cursor.
Cursor Width	Choose a line width from the dropdown list for the active cursor.
Horizontal Cursor	Controls the color of the active horizontal cursor line. Click the colored box to open the Color Selection dialog box and make a new color selection.
Vertical Cursor	Controls the color of the active vertical cursor line. Click the colored box to open the Color Selection dialog box and make a new color selection.
Cursor	
Cursor Style	Choose a cursor line style from the dropdown list for cursors when they are not the active cursor.
Cursor Width	Choose a cursor line width from the dropdown list for cursors when they are not the active cursor.

Table 8-54. EZwave Display Preferences - Cursor/Marker Options (cont.)

Field	Description
Horizontal Cursor	Controls the color of horizontal cursor lines for cursors when they are not the active cursor. Click the colored box to open the Color Selection dialog box and make a new color selection.
Vertical Cursor	Controls the color of vertical cursor lines for cursors when they are not the active cursor. Click the colored box to open the Color Selection dialog box and make a new color selection.
Marker Line	Controls the color of marker lines. Click the colored box to open the Color Selection dialog box and make a new color selection.
Default	Restores the font, style, and color of cursors and markers to their default settings.

Eye Mask Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Eye Mask.

Use this dialog box to modify the color and line width of eye masks.

Objects

Table 8-55. EZwave Display Preferences - Eye Mask Options

Field	Description
Pass Color	Controls the color of a pass status of the eye mask specification. Click the colored box to open the Color Selection dialog box and make a new color selection. The default is blue.
Fail Color	Controls the color of a “fail” status of the eye mask specification. Click the colored box to open the Color Selection dialog box and make a new color selection. The default is red.
Line Width	Controls the line width of eye masks. Select a width from this dropdown list. By default the width is the same as the Eye Diagram waveform width.
Default	Restores the font and color of eye masks to the default settings.

Grid Options (Fonts and Colors)

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Grid.

Use this dialog box to modify the color, width, and style of both major and minor grid lines. Minor grid lines are displayed when waveforms are plotted using logarithmic scales.

Objects

Table 8-56. EZwave Display Preferences - Grid Options

Field	Description
Major Grid Lines	
Color selected according to the Y Axis	When enabled, the color of the major grid lines will match the color of the y axis. Enabled by default.
User Color	Used to select a different color for the major grid lines. Click the colored box to open the Color Selection dialog box and make a new color selection.
Line Width	Controls the width of the major grid lines. Select a width from 1 to 5, from the dropdown list. By default the width is 1.
Line Style	Controls the line style for the major grid from solid, dotted or dashed. Select a style from the dropdown list. By default the line style is solid.
Minor Grid Lines	
Color selected according to the Y Axis	When enabled, the color of the minor grid lines will match the color of the y axis. Enabled by default.
User Color	Used to select a different color for the minor grid lines. Click the colored box to open the Color Selection dialog box and make a new color selection.
Line Width	Controls the width of the minor grid lines. Select a width from 1 to 5, from the dropdown list. By default the width is 1.
Line Style	Controls the line style for the minor grid from solid, dotted or dashed. Select a style from the dropdown list. By default the line style is dotted.
Grid	
Default	Restores the color, line width, and line style of both major and minor grid lines to the default settings.

Header Text Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Header Text.

Use this dialog box to modify the fonts and colors of the header text. By default, header text is hidden. Choose **File** > **Page Setup** to select the information to display in the header.

Objects

Table 8-57. EZwave Display Preferences - Header Text Options

Field	Description
Font	Controls the font of header text.
Font Size	Controls the size of the header text font.
Font Style	Controls the style of font for the header text.
Color	Click the colored box to open the Color Selection dialog box and make a new color selection.
Preview	Displays a sample of what the text will look like with the current selections.
Default	Restores the font and color of header text to the default settings.

Histogram Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Histogram.

Use this dialog box to modify the fill properties of histograms.

Objects

Table 8-58. EZwave Display Preferences - Histogram Options

Field	Description
Pattern fill	Specifies that histogram bars use a pattern fill. This is the default setting.
Transparent	Specifies that the histogram bars be transparent, useful for large numbers of histograms. Overlapping histograms are shaded to improve visualization of the distribution.
No fill	Specifies that the histogram bars are not filled with any pattern. This is useful when running the EZwave viewer over a network.
Full	Specifies that the histogram bars are filled with solid color.

Measurement Annotation Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Measurement Annotation.

Used to modify the fonts and colors of measurement annotations, added by the Measurement tool.

Objects

Table 8-59. EZwave Display Preferences - Measurement Annotation Options

Field	Description
Font	Specifies a font for measurement annotation text.
Font Size	Specifies the size of the font for measurement annotation text.
Font Style	Specifies the style of font to use for measurement annotation text.
Color	Click the colored box to open the Color Selection dialog box and make a new color selection.
Preview	Displays a sample of what the text will look like with the current selections.
Measurement Annotation Color	Controls the default color of measurement annotations: • Use Waveform Color Uses the same color as the waveform. Available only for annotations that are attached to a single waveform. • Use Custom Color Measurement annotations will use the custom color defined on this dialog box.
Default	Restores the font and color of measurement annotations to the default settings.

Pick Points Options (Fonts and Colors)

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Pick Points.

Used to modify colors and color mode for pick points.

Objects

Table 8-60. EZwave Display Preferences - Pick Points Options (Fonts and Colors)

Field	Description
Pick Point Color	Specifies the color of pick points. Default colors: • Red for the white and black background color schemes • Black for the monochrome color scheme.
Pick Point Color Mode	Specifies whether to use a custom color for pick points (default) or to have the pick points match the color of the waveform.
Pick Point Selection Color	Specifies the color of a pick point when it is selected. Default colors: • White for the black background color scheme • Black for the white background and monochrome color schemes.

Row Title Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Row Title.

Use this dialog box to modify the default fonts and colors of row titles. The font properties of individual row titles can be edited by right-clicking the title in the graph window and selecting properties.

Objects

Table 8-61. Fonts and Colors - Row Title Dialog Box Contents

Field	Description
Font	To change the font of row title text, select an available font from this dropdown list. The default is Dialog.
Font Size	To change the size of the font, select a point size from this dropdown list. The default is 12.
Font Style	To change the style of font, select a font style from this dropdown list. The default is Plain.
Color	Click the colored box to open the Color Selection dialog box and make a new color selection. The default is Blue.
Preview	Displays a sample of what the text will look like with the current selections.
Default	Restores the font and color of row titles to the default settings.

Text Annotation Options (Fonts and Colors)

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Text Annotation.

Use this dialog box to modify the default fonts and colors of text annotations.

Objects

Table 8-62. EZwave Display Preferences - Text Annotation Options (Fonts and Colors)

Field	Description
Font	Specifies the font to use for text annotations.
Font Size	Specifies the point size of the font to use for text annotations.
Font Style	Specifies the style of the font to use for text annotations: • Plain • Bold • Italic • Bold/Italic
Color	Click the colored box to open the Color Selection dialog box and make a new color selection. This control is not available when the Monochrome theme is active.
Preview	Displays a sample of what the text will look like with the current selections.
Waveform Annotation Color	Controls the default color of text annotations: • Use Waveform Color Text annotations for all waveforms are the same color as the waveform. This option is selected by default for annotations that are attached to waveforms. It is unavailable for annotations that are not attached to a waveform. • Use Custom Color Text annotations will use the custom color defined on this dialog box.
Default	Restores the font and color of text annotations to the default settings.

Waveform Colors Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Waveform Colors.

Use this dialog box to define colors for analog and digital waveforms in the graph windows.

Objects

Table 8-63. EZwave Display Preferences - Waveform Colors Options

Field	Description
Use Same Waveform Color Per Database in Tandem Mode	When Tandem Mode is active, specifies waveforms from the same database are plotted with the same color. Enabled by default.
White Background Color Scheme shares the Black Background Waveform Colors Palette	Only available when the White Background Color Scheme is selected. Use this option to share the waveform color palette between White and Black Background Color Schemes. Enabled by default.
Analog	
Color Palette Size	Controls the size of the color palette used to display waveforms.
Color Palette	Click a color block to select a new color from the Color Selection dialog box.
Digital	
Bus Color	Controls the color of bus waveforms. Click the colored box to open the Color Selection dialog box and make a new color selection. The default color is green.
Bus Value Color	Controls the color of bus values. Click the colored box to open the Color Selection dialog box and make a new color selection.

Usage Notes

In EZwave, the color scheme may be monochrome, or a black or white background. By default the color palette size setting for analog objects is shared between white and black backgrounds. Colors are set when the black color scheme is active. When the white scheme is activated, an option to disable this sharing is visible. Digital object colors can always be different between each scheme.

Waveform Display Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Waveform Display.

Used to modify the line style, width and data point display of waveforms.

Objects

Table 8-64. Fonts and Colors - Waveform Display Dialog Box Contents

Field	Description
Line Style	Controls the default line style of waveforms.
Line Width	Controls the default line width of waveforms.
Data Point Symbol	Controls the default data point symbol of waveforms.

Usage Notes

Caution

The use of a waveform Line Width other than 1 pixel and/or a Line Style other than a plain line may significantly reduce the waveform display performance.

Waveform Name Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Waveform Name.

Use this dialog box to modify fonts, colors and display format of waveform names.

Objects

Table 8-65. Fonts and Colors - Waveform Name Dialog Box Contents

Field	Description
Waveform Font & Color	
Font	Controls the font used for waveform names.
Font Size	Controls the size of the font used for waveform names.
Font Style	Controls the style of font used for waveform names.
Color	Click the colored box to open the Color Selection dialog box and make a new color selection.
Preview	Displays a sample of what the text will look like with the current selections.
Waveform Names Display	
Waveform Hierarchy	Defines global settings for how waveform names are displayed in graph windows. These are common to all Color Schemes. Specifies the hierarchy display as one of the following: • Full Hierarchy — Specifies that the full hierarchy is displayed. Default. • No Hierarchy (Leaf Name Only) — Specifies that only the leaf name is displayed. • Display N Levels — The specified number of levels is displayed.
Justify Value	Specifies how the waveform name is justified when displayed: Left (Default) or Right.

Table 8-65. Fonts and Colors - Waveform Name Dialog Box Contents (cont.)

Field	Description
Database Name	Specifies how the database name is displayed from one of the following: • Always Show Database Name. • Show Name If Two or More Databases. Calculated (<calc>) databases are not included in the count of databases. • Always Hide Database Name. Default. • Show Database Name in Tandem Mode. Database names are always displayed in Tandem mode, irrespective of the three settings above. Default.
Default	Restores the font and color of waveform names to the default settings.

Usage Notes

Some of these settings can also be defined using the Waveform Names Display Dialog Box which can be accessed through the menu item **Format > Waveform Names Display**.

Waveform Selection Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Waveform Selection.

Use this dialog box to modify the line styles and colors of selected and highlighted waveforms.

Objects

Table 8-66. EZwave Display Preferences - Waveform Selection Options

Field	Description
Waveform Selection	
Color	Controls the color of the waveform when selected. Click the colored box to open the Color Selection dialog box and make a new color selection.
Line Width	Controls the line width of a selected waveform.
Waveform Highlight	
Color	Controls whether a highlighted waveform changes color. Click the colored box to open the Color Selection dialog box and make a new color selection.
Line Width	Controls the line width of a highlighted waveform.

Usage Notes

Use the Highlight column within the Parameter Table to highlight waveforms.

If the Color Option is left unchecked and the Line Width is set to Automatic for either Waveform Selection or Waveform Highlight, the selection or highlight will not be differentiable from other waveforms the graph.

Window Background Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Window Background.

Use this dialog box to modify the window background colors.

Objects

Table 8-67. Fonts and Colors - Window Background Dialog Box Contents

Field	Description
Window Background Color	Controls the color of the window background. Click the colored box to open the Color Selection dialog box and make a new color selection. The default background color is white.
Custom Waveform Area Color	Controls the background color of the waveform area (inside the axes). Click the colored box to open the Color Selection dialog box and make a new color selection. The default background color is white.
Default	Restores the window background color to the default setting.

Zero-Level Line Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Zero-Level Line.

Use this dialog box to modify the colors of zero-level lines.

Objects

Table 8-68. Fonts and Colors - Zero-Level Line Dialog Box Contents

Field	Description
Color selected according to the Y Axis	The zero-level line color will match the colors of the appropriate y axes. Enabled by default.
User Color	Specifies a different color to use for zero-level lines. Click the colored box to open the Color Selection dialog box and make a new color selection.

Histogram Plot Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Histogram folder on the left side, and choose Plot.

Used to specify default options when plotting a histogram.

Objects

Table 8-69. EZwave Display Preferences - Histogram Plot Options Dialog Box Contents

Field	Description
Histogram Properties	
Plot Predefined histogram (when available in file)	Specifies that the histogram stored in the simulation file (for example, a HIST() waveform generated by Eldo on an extract) should be plotted instead of a new histogram being calculated. With this option selected, all Histogram Customization options are ignored.
Histogram Customization	
Number of Bins	Specifies the number of waveform divisions (resolution) to use. Type a numerical value or a rule (Scott, Sturge, or Sqrt (the square root of the number of samples)). The default is 10.
Minimum X Value	Specifies the X value at the lower bound of a window interval.
Maximum X Value	Specifies the X value at the upper bound of a window interval.
Gather Values outside Sigmas”	Restricts the histogram width to (mean +-SIGBIN* standard deviation) and specifies the value of SIGBIN, which is equivalent to the SPICE Eldo SIGBIN parameter.
Min and Max Bins Centered on Min and Max Values	When selected, the first (and respectively last) bin is centered on the sample min (resp. max) value. When not selected, the bin is aligned to the right of the min (resp. aligned to the left of the max).
Sample Data	When selected, the input waveform is sampled (the number of points remains the same, but are equally distributed; see the histogram function for details). When not selected, the raw Y values of the input waveform are used.
Normalize Y	When selected, the histogram height is normalized; the sum of the bin height equals 1.0. When not selected, the height of the bins equals the number of hits, and the sum of the bin height equals the total number of points of the input waveform.
Shift Data so that Mean Equals 0	When selected, the Y-values of the input waveform are decreased by the mean() of these values, so the displayed mean equals 0.

Table 8-69. EZwave Display Preferences - Histogram Plot Options Dialog Box Contents (cont.)

Field	Description
Multiply Data so that Standard-Deviation Equals 1	When selected, the Y-values of the input waveform are decreased by the mean() of these values, and then divided by the standard-deviation. So the displayed mean equals 0, and the displayed standard-deviation equals 1.0. This option is available only if "Shift Data so that Mean Equals 0" is selected, above.
Companion Waveforms	
Plot Gaussian Distribution	When selected, the Gaussian distribution is plotted based on the mean and standard-deviation of the input waveform. The range for this distribution is between mean +/- N*stddev, with the following constraints: • N is the smallest integer so that the Y min() and max() of the input waveform are in the range • N is greater than or equal to 4 • N is lesser than or equal to 20
Plot PDF (when available in file)	Plots the PDF (potential density function) associated to the input waveform. This functionality is available in *.wdb* files generated by an Eldo Monte Carlo.
Plot CDF	Plots the CDF (cumulative density function) of the input waveform.
CDF is a Step Waveform	When selected, the CDF is plotted as a step waveform.
Fit Histogram Area (PDF, Gaussian) and Height (CDF)	When selected, the histogram is displayed with only one y axis. The y-axis bounds are derived from the histogram Y-min (0) and Y-max. The PDF and Gaussian distribution are multiplied by the area of the histogram (area fit). The CDF display is multiplied by the Y-max of the histogram so that is has the same height as the histogram (height fit). When unchecked, a second y axis is displayed for Gaussian distribution, PDF and CDF.

Histogram Measures Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Histogram folder on the left side, and choose Measures.

Use this dialog box to specify the measures to display when plotting a histogram.

Objects

Table 8-70. EZwave Display Preferences - Histogram Measures Options Dialog Box Contents

Field	Description
Single Measures	
Draw Average	Select to draw a vertical line at the average location.
Draw Mean	Select to draw a vertical line at the mean location.
Draw Median	Select to draw a vertical line at the median location.
Draw Nominal Value (when available in file)	When selected, and when the nominal value is stored in the waveform database file, this option draws a vertical line at the nominal location. It is available in *.wdb* files generated by an Eldo Monte Carlo simulation.
Twin Measures	
Draw Probability	There are three options: • N-Sigma: Empirical. When selected, the plot is based on the CDF. Specify a comma-separated list of integer N values, between 1 and 7. For probability P of N-Sigma, the range: min point = X of first CDF point where Y ³ P/2 max point = X of last CDF point where Y £ 1 - P/2 • N-Sigma: Theoretical. When selected, the plot is based on the mean and standard deviation of the theoretical Gaussian distribution. Specify a comma-separated list of integer N values, between 1 and 7. A range is drawn for each value, from: mean-N´stddev to mean+N´stddev. • Probability. This is the same computation as N-Sigma: Empirical (above) except that you specify a comma-separated list containing the required probability values, between 0 and 1, or percentage values between 0% and 100%. (For example 0.6827,0.9545,0.98,99%.) The range legend: Prob <percentage_of_data_in_range>% [<lower_value> ; <upper_value>]

Table 8-70. EZwave Display Preferences - Histogram Measures Options Dialog Box Contents (cont.)

Field	Description
Lower-Bound/Upper-Bound	When either option is selected, a vertical line is drawn, which can be repositioned in the same way as cursors. Depending on the two settings, the bound location is: • As defined in the *.wdb* file, when Predefined Position (when available in file) is selected. It is available in *.wdb* files generated by an Eldo Monte Carlo simulation. • At a user-defined location, if User Defined Position is selected, and the previous option Predefined Position (…) is not selected, above, or the bound value is not stored in the file. The user defined position is relative to mean + <value> * stddev (as N-sigma, but <value> is a real number, not an integer).

Usage Notes

You can edit a plotted histogram, in which case the User Defined Position value is an absolute value, not a relative one.

When both Lower and Upper Bound are displayed, only the Upper Bound legend displays the yield value.

Histogram Legend Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Histogram folder on the left side, and choose Legend.

Use this dialog box to configure and display a legend to the statistical measures displayed on a histogram.

Objects

Table 8-71. EZwave Display Preferences - Histogram Legend Options Dialog Box Contents

Field	Description
Draw Stat Legend	Enables the display of the legend on histograms.

Edit Stat Legend	Used to define the values to display in the legend. Type legend items in the following format:
	`<Lable>= @(<measure>)`
	
	For example:
	`Mean= @(mean)`
	
	All available measures can be added using the Shortcut Buttons. The variable @(name) adds the waveform name to the legend. The large field at the bottom of the dialog box displays a preview of how the legend will be laid out, using placeholder values.

Shortcut Buttons	
Average	Adds the average to the Stat Legend text.
Cp	Adds the process capability to the Stat Legend text, which is equal to: `(upper_bound - lower_bound) / (^6* stddev)` This measure requires that the lower and upper bound are plotted, otherwise it is <undefined>.
CpK	Adds the process capability, assuming that it is not centered. It is equal to: `Min(upper_bound - mean) / (3*stddev),(mean - lower_bound) / (3* stddev)` This measure requires that the lower or upper bound is plotted, otherwise it is <undefined>.
Kurtosis	Adds the skewness value to the Stat Legend text. This option is available only when the value is available in the wdb file.

Table 8-71. EZwave Display Preferences - Histogram Legend Options Dialog Box Contents (cont.)

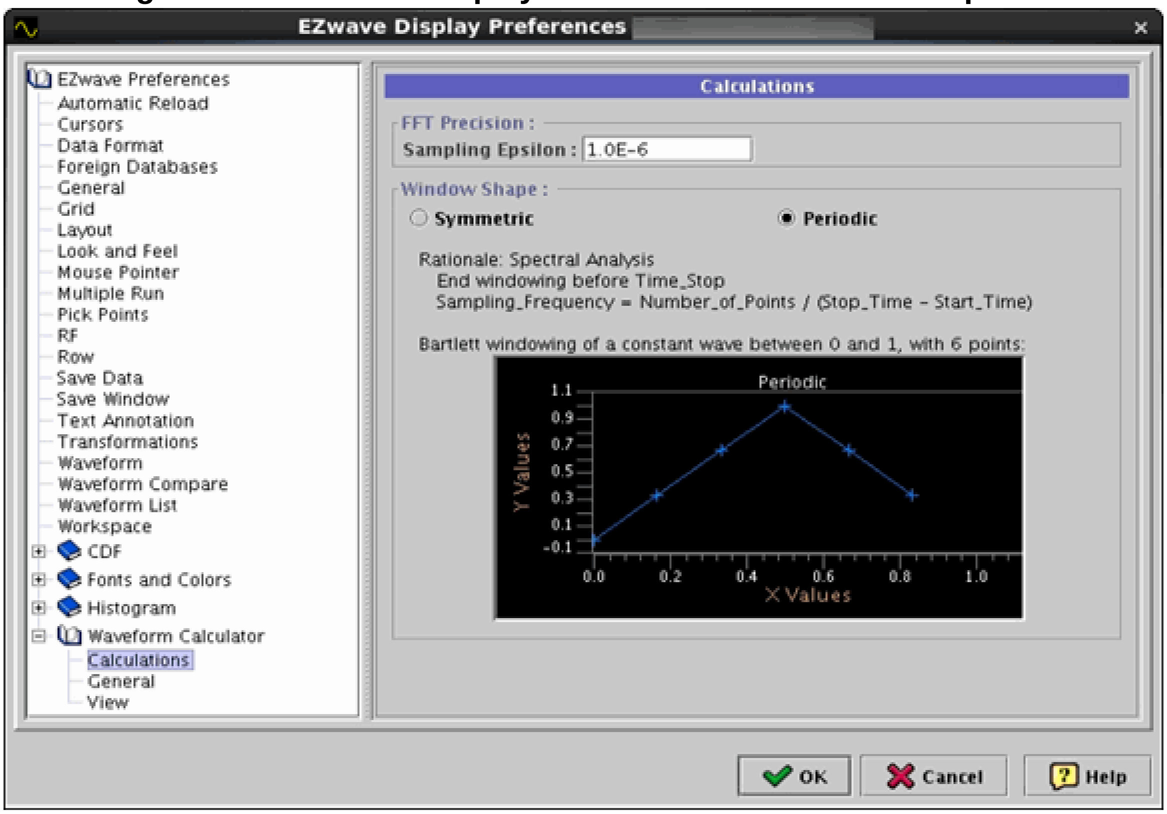
Field	Description
Name	Adds the name of the input waveform to the Stat Legend text, without database or hierarchy names.
Max	Adds the maximum value to the Stat Legend text.
Mean	Adds the mean value to the Stat Legend text.
Median	Adds the median value to the Stat Legend text.
Min	Adds the minimum value to the Stat Legend text.
Nominal	Adds the nominal value to the Stat Legend text, when available in the waveform database file, otherwise it is <undefined>.
Relative Std. Dev.	Adds the relative standard deviation to the Stat Legend text, which is equal to: • stddev if mean is zero • stddev/mean if stddev is not zero
Size	Adds the number of points of the input waveform to the Stat Legend text.
Skewness	Adds the skewness value to the Stat Legend text. This option is available only when the value is available in the wdb file.
Std. Dev.	Adds the standard deviation to the Stat Legend text.
@	Adds the @ symbol to the Stat Legend text.

Waveform Calculator Calculation Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Waveform Calculator folder on the left side, and choose Calculations. Or, select **Edit** > **Options** from within the Waveform Calculator, then choose **Calculations**.

Use this dialog box to specify calculation options for the Waveform Calculator.

Figure 8-26. EZwave Display Preferences - Calculation Options



Objects

Table 8-72. EZwave Display Preferences - Waveform Calculation Options

Field	Description
FFT Precision	
Sampling Epsilon	Specifies the value used in the uniform sampling algorithm of signal processing functions. EZwave uses this value to determine whether it needs to perform interpolation between X values of the waveform at each *t+dt* sampling point. The sampling epsilon value is relative to the scale of the waveform being sampled.

Table 8-72. EZwave Display Preferences - Waveform Calculation Options

Field	Description
Window Shape	Specifies the window shape: • Symmetric — Standard FFT setup. • Periodic — Setup is enhanced for spectral analysis of periodic signals. The default. The Window Shape setting set here is used as the default within FFT windows and functions. For more information see "Signal Processing Function Window Shapes" on page 395.

Waveform Calculator General Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Waveform Calculator folder on the left side, and choose General. Or, select **Edit > Options** from within the Waveform Calculator.

Use this dialog box to specify general Waveform Calculator options.

Objects

Table 8-73. EZwave Display Preferences - Waveform Calculator General Options

Field	Description
Notation Suffix	
IEEE Spice	Specifies the expression evaluation logic. • IEEE — Specifies IEEE notation is used in the Waveform Calculator (the default). • Spice — Specifies SPICE notation is used in the Waveform Calculator. See also AMS_VIEWER_WFC_NOTATION in "EZwave Environment Variables" on page 54 and command "setNotation" on page 905.
Output Suffix	Specifies the output result is displayed using IEEE (SI standard) suffix notation, instead of exponential format. For example, 123.45e-13 becomes 12.345p (where p stands for pico). **Note:** IEEE Suffix or SPICE Suffix must also be selected on the EZwave Display Preferences - Data Format Options dialog box.
Trigonometric Angle	
Degrees Radians Gradians	Specifies the trigonometric angle as one of the following: • Degrees — A unit of angular measure in which the angle of an entire circle is 360 degrees (denoted DEG). • Radians — A unit of angular measure in which the angle of an entire circle is 2 pi radians (denoted RAD). • Gradians — A unit of angular measure in which the angle of an entire circle is 400 gradians (denoted GRAD).
Temperature Unit	
Celsius Kelvin	Specifies the measurement unit for temperature display as Celsius or Kelvin.
Logic Panel Type	
VHDL Verilog	Specifies the logic panel type as VHDL or Verilog, and uses the appropriate notation.

Table 8-73. EZwave Display Preferences - Waveform Calculator General Options (cont.)

Field	Description
Evaluation Result Display	
Clear Last Result	Clears the Expression Entry Area each time after the expression is evaluated. This is equivalent to clicking Eval and Clear.
Always Plot Last Result	When enabled, the resulting waveform is always plotted after the expression is evaluated. This is equivalent to clicking **Eval** and **Plot**. This option is enabled by default.

Automatically Truncate Long Result	Automatically truncates very long results.
	Example without truncation:
	`wfcalc>5.1*4.1`
	`•20.909999999999997`
	Example with truncation:
	`wfcalc>5.1*4.1`
	`•20.91`

Reset Variables	
Reset all calculator variables	Removes all existing variables in the calculator.
User Extension Files	
Load User Extension Files At Startup	Specifies that user-defined functions are loaded at startup. Refer to "Using and Editing User-Defined Functions in the Waveform Calculator" on page 319. You can also use the environment variable AMS_UDF_LOAD to specify a path to load extension files at startup - refer to "EZwave Environment Variables" on page 54.
Directory path	Specifies the default path for loading all of the user extension files. These files must have a *.tcl* extension to load.
Load CalcPAD Scripts At Startup	Specifies that waveform calculator scripts that provide equivalent functionality to legacy AFS WaveCrave CalcPAD scripts are loaded at startup. Refer to "Waveform Calculator Example Tcl Scripts" on page 1224.

Waveform Calculator View Options

To access: Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Waveform Calculator folder on the left side, and choose **View**. Or, select **Edit** > **Options** from within the Waveform Calculator, then choose **View**.

Use this dialog box to specify Waveform Calculator display options.

Objects

Table 8-74. EZwave Display Preferences - Waveform Calculator View Options

Field	Description
Enable Wizard Dialogs (fft, ifft, ...)	Specifies that certain Waveform Calculator functions with more than three arguments will display a specific wizard dialog box to assist in setting up the arguments (fft, ifft, …).
Enable Generic Wizard Forms	Specifies that a generic form is displayed for every Waveform Calculator function to assist with setting up the arguments.
Waveform Calculator in Independent Window	Specifies that the Waveform Calculator will open in an independent window.
Enable Function Name Completion in Entry Field	Specifies that the possible function names are displayed as you type in the Waveform Calculator entry field.
Calculator Layout Selection	Specifies the layout of the shell, button and documentation help panes.

Filter Dialog Box

To access: Click the filter icon at the right of a parameter column heading on the Parameter Table Dialog Box.

Use this dialog box to filter the compound waveform runs that are displayed. You can filter by selecting parameters values from a list or by specifying conditions.

Figure 8-27. Filter Dialog Box With List Option Selected

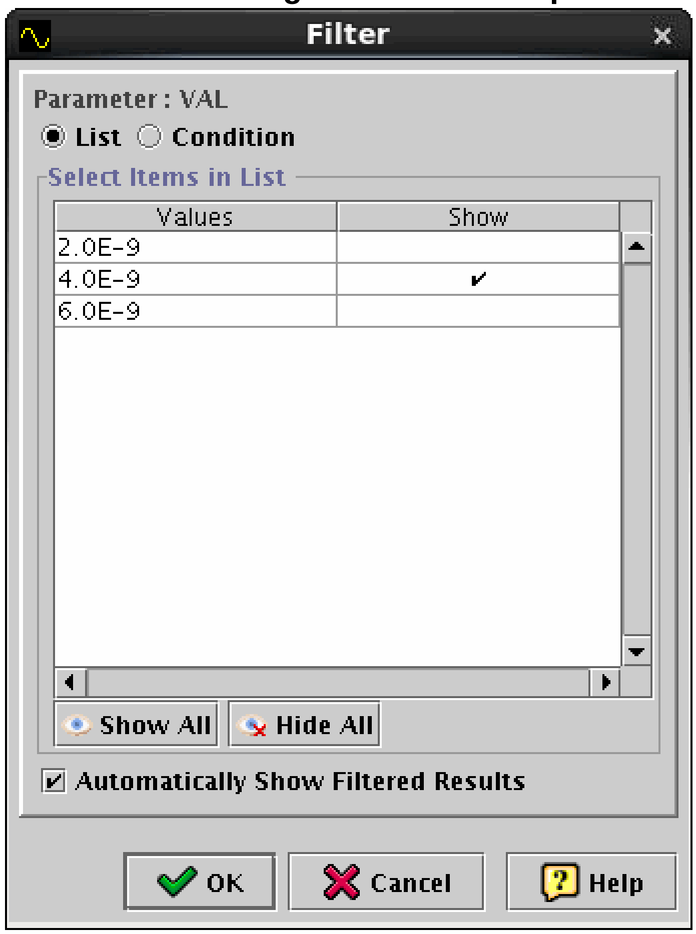
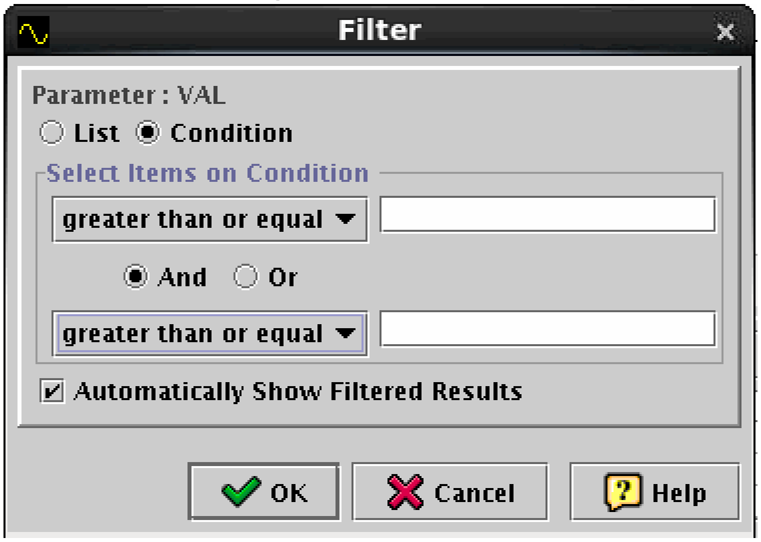


Figure 8-28. Filter Dialog Box With Condition Option Selected



Objects

Table 8-75. Filter Dialog Box Contents

Field	Description
Parameter	Displays the parameter to filter on.
List	Selects the List filter option.
Select Items in List	Select the required parameter value(s) from this list and click **OK** to filter the runs to be displayed on the Parameter Table Dialog Box.
Show All/Hide All	You can show all runs or hide all runs for this parameter.
Condition	Selects the Condition filter option.

Table 8-75. Filter Dialog Box Contents (cont.)

Field	Description
Select Items on Condition	Choose the required condition from the menu and type a parameter value. Available operators on the values: • equals • less than • less than or equal • greater than • greater than or equal • contains • begins with • ends with • in • does not equal • does not contain • does not begin with • does not end with • not in. Optionally, select **And** or **Or** and choose a condition from the second menu and type a second value. Click **OK** to filter the runs to be displayed on the Parameter Table Dialog Box.
Automatically Show Filtered Results	Results that match the filter become visible in the graph area when you click **OK**.

Usage Notes

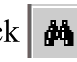
When you click **OK** on the Filter Dialog Box:

- runs in the Parameter Table are filtered out if they do not match the filter:
    - the corresponding table rows are hidden
    - compound items are hidden in the graph area.
- runs in the Parameter Table are filtered in if they match the filter:
    - the corresponding table rows are visible
    - if the Automatically Show Filtered Results option is selected, compound items become visible in the graph area.

Related Topics

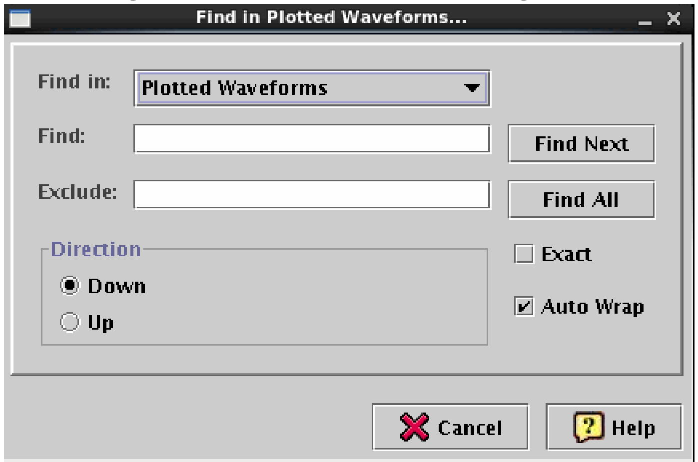
Parameter Table Dialog Box

Find Tool Dialog Box

To access: Choose **Edit** > **Find** from the main menu, or click  in the Waveform Lists.

Use this dialog box to locate specific waveforms by name.

Figure 8-29. Find Tool Dialog Box



Objects

Table 8-76. Find Tool Dialog Box Contents

Field	Description
Find in	Specifies where to search. Choose from: • Plotted Waveforms • Structure List • Waveform List • List View If the Structure View is disabled, the Structure List and Waveform List items are replaced by Tree View.
Find	Used to specify the search string. Wildcards are allowed in the expression: asterisk (*) to match any number of characters; and the question mark (?), to match a single character.
Exclude	Specifies named waveforms to exclude from the search. Wildcards are allowed in the expression: asterisk (*) to match any number of characters; and the question mark (?) to match a single character.
Direction	Specifies the direction of search when searching any of the list views. The default is down.
Exact	Specifies that the search will only return waveforms that match the search string exactly.

Table 8-76. Find Tool Dialog Box Contents (cont.)

Field	Description
Auto-wrap	Specifies that searching continues from the beginning of the list when the search reaches the end of the list. If the search is in the up direction, searching continues at the bottom of the list when it reaches the top of the list.
Find Next	Click to highlight the next waveform name that matches the search expression.
Find All	Click to highlight all the waveform names that matches the search expression.

Jitter Tool Dialog Box

To access: Choose **Tools** > **Jitter** from the main menu.

Use this dialog box to analyze Time Domain clock jitter between any digital and analog target signals with respect to a reference period or frequency, or to calculate Phase Noise jitter. The fields displayed on the Jitter Tool dialog box change according to the Jitter Selection.

Figure 8-30. Jitter Tool Dialog Box

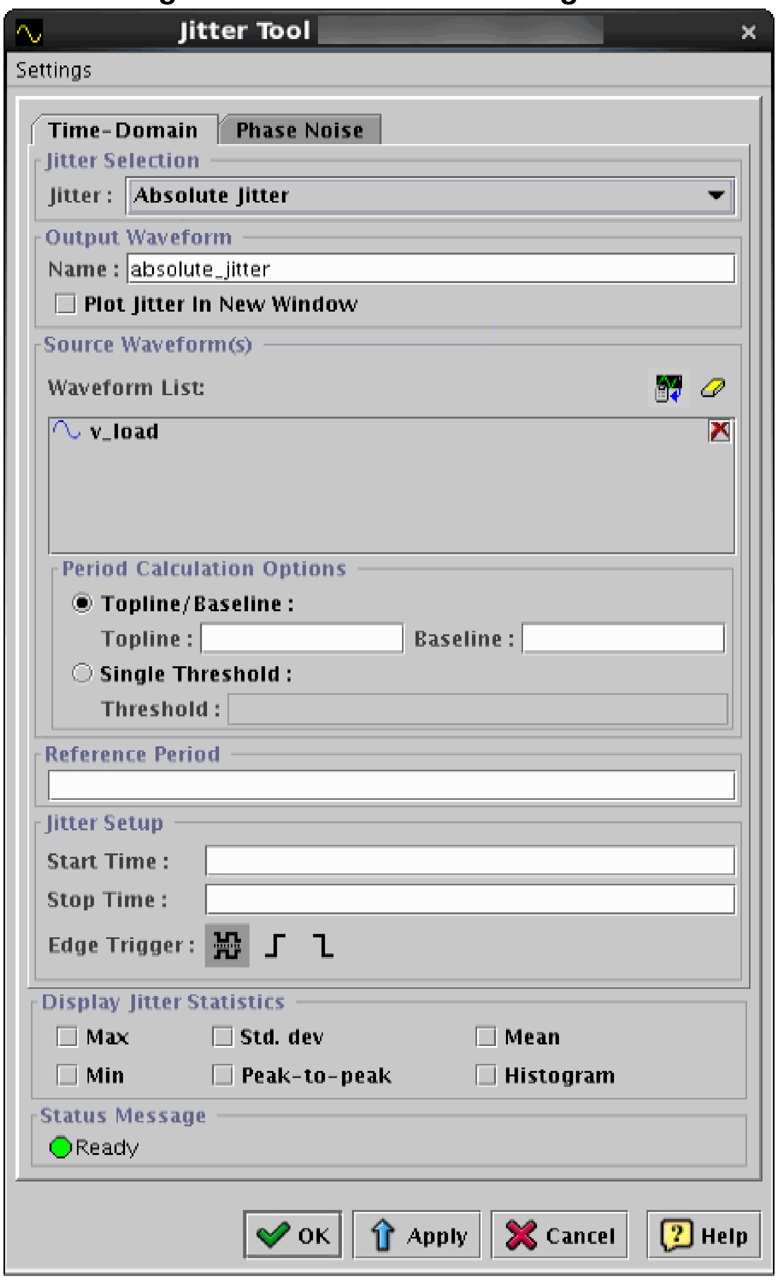


Figure 8-31. Jitter Tool Dialog Box - Phase Noise Tab

Objects

Table 8-77. Jitter Tool Dialog Box Contents

Field	Description
Jitter Selection	Specifies the type of jitter analysis to perform. The options available in the dialog box change based on the jitter type selected.
Output Waveform	Specifies a name for the resulting jitter waveform. The default name is *<jitter_type>_jitter*. If a jitter result waveform with the given name is already plotted in the active graph window and you leave the output waveform name unchanged in this field, the jitter result waveform will be updated according to the jitter setup options when you click **Apply** or **OK**.

Table 8-77. Jitter Tool Dialog Box Contents (cont.)

Field	Description
Plot Jitter in New Window	Plots the resulting jitter waveform in a new graph window.
Source Waveform(s) (Controls available in this section vary by jitter type or dialog box tab)	
Waveform List	Select the source waveform(s) for which jitter is calculated. Either: • Drag waveforms from the active graph window into the Waveform List area, or • Select waveforms in the active graph window click the **Add Selected Waveforms**  icon to add them to the Wavefrom List area.
Period Calculation Options (for Source or Reference Waveforms)	
Reference Waveform	Select a reference waveform for the calculation. Select waveforms in the active graph window or Waveform List and click the **Add Selected Waveforms**  icon. Only available for TIE jitter.
Topline/Baseline	Topline specifies the y value that sets the high threshold of a signal. Baseline specifies the y value that sets the low threshold of a signal. Specify Automatic to have these values computed.
Single Threshold	Specifies the y value that sets a single threshold for a signal. Specify Automatic to have this value computed.
Reference Period or Frequency	
Reference Period or Frequency	Specifies the Reference Period or Reference Frequency.
Fundamental Frequency	Specifies the Fundamental Frequency (**Phase Noise** tab).
Jitter Setup (Controls available in this section vary by jitter type or dialog box tab)	
Duty Cycle	With this option set to Automatic the tool uses the dedicated Measuring Duty Cycle measure from the Measurement Tool to calculate the amount of time the signal is active per period, for use in the jitter calculation (automatic Topline and Baseline parameters are used for Duty Cycle calculation). If you want to specify this value manually, select the User Defined option and type the numeric value in the field to the right (default = 0.5). This value can be specified as a percentage using the percent character (for example, 50%).

Table 8-77. Jitter Tool Dialog Box Contents (cont.)

Field	Description
Number of Cycles in Sample	Specifies the maximum number of waveform cycles in the N-cycle sample, which will participate in the jitter calculation. Available only for Period Jitter, Frequency Jitter, Cycle-to-cycle Jitter, and Long-term Jitter. For Period Jitter, Frequency Jitter, and Cycle-to-cycle Jitter, a default option is available, which sets the Number of Cycles to 100. For Long-term Jitter, the option Automatic is available instead; with this selected, the number of cycles is calculated automatically as: $$max\_of\_cycles\_in\_sample = \left\lceil \frac{total\_nb\_of\_cycles}{50} \right\rceil$$ This is done to obtain at least 50 samples for each Long-term Jitter point calculation, which is considered sufficient for the required level of precision.
Start Time	The source waveform time from which jitter is applied. Optional.
Stop Time	The source waveform time to which jitter is applied. Optional.
Edge Trigger	Specifies the waveform edge(s) that is taken into account during the jitter calculations: • Rising Edge • Falling Edge • Either Rising or Falling Edge, depending on which comes first in the specified time range. (Default)
Jitter Confidence Interval Setup	
RMS Jitter	Specifies the RMS jitter value.
Gaussian Samples Number Sequence	Specifies the Gaussian samples number sequence Start, End, and Step values.
Confidence Level	Specifies the jitter confidence level.
Display Jitter Statistics	
Max	Displays the maximum value of the result (output) waveform.
Min	Displays the minimum value of the result (output) waveform.
Std. dev	Displays the Std. dev between the Max and Mean values of the result (output) waveform.
Peak-to-peak	Displays the peak-to-peak value of the result (output) waveform.
Mean	Displays the mean value of the result (output) waveform.

Table 8-77. Jitter Tool Dialog Box Contents (cont.)

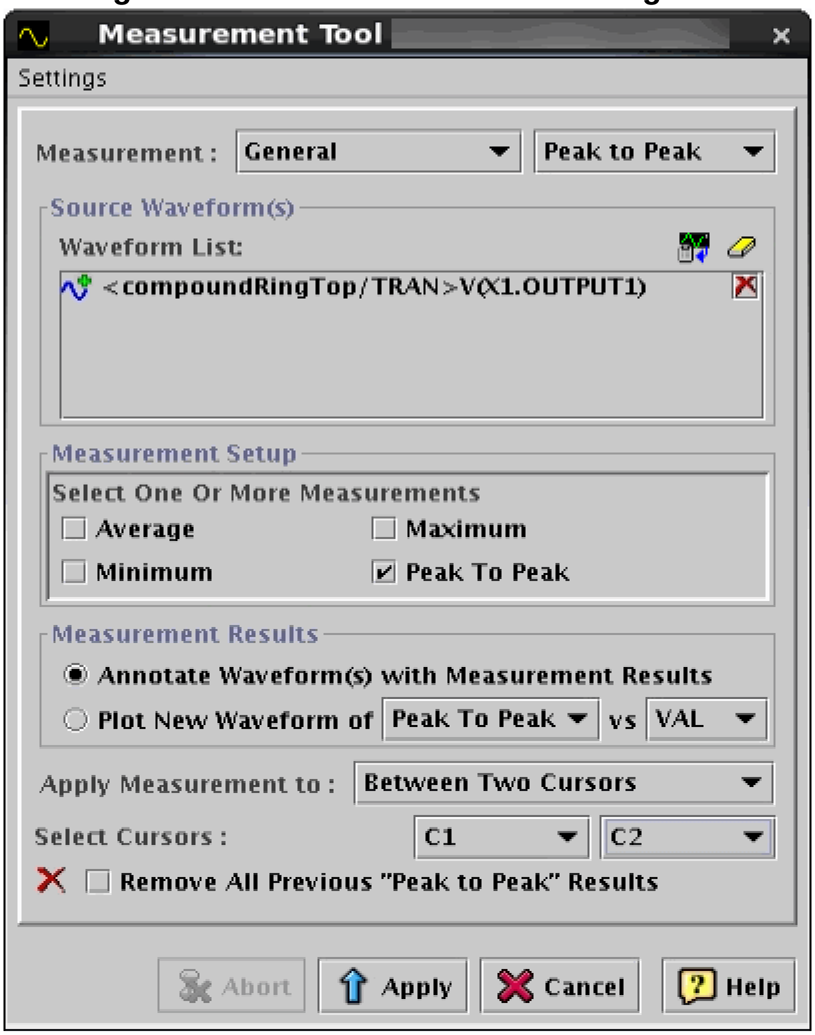
Field	Description
Histogram	Generates and displays a Monte Carlo histogram of the result (output) waveform showing the magnitude probability density distribution of the waveform.

Measurement Tool Dialog Box

To access: Choose **Tools** > **Measurement Tool** from the main menu.

Use this dialog box to specify options for a variety of analog and mixed-signal measurement operations for waveforms displayed in the graph window.

Figure 8-32. Measurement Tool Dialog Box



Objects

Table 8-78. Measurement Tool Dialog Box Contents

Field	Description
Measurement	Specifies the measurement category and type.

Table 8-78. Measurement Tool Dialog Box Contents (cont.)

Field	Description
Waveform List	Specifies the waveforms to apply the measurement to. The Add Selected Waveform icon adds the currently selected waveforms to the Waveform List. The Clear Waveform List icon removes all waveforms in the list. The Delete icon removes the specified waveform from the list.
Measurement Setup	Specifies additional information specific to the measurement selected, such as topline/baseline, edge trigger, and measurement results settings. Refer to the specific section in "Measurement Tool" on page 279 for details on these settings for the selected measurement.
Measurement Results	Specifies how the results should be displayed from one of the following: • Annotate Waveform(s) with Measurement Results. Default. • Plot New Waveform of "<measurement type>" vs. <parameter>. If there is more than one parameter to choose from, a dropdown list is available for specifying the required parameter. When more than one waveform has been selected for the measurement, only parameters common to all waveforms are available to choose from. This section is not available for all measurements. Refer to the specific section in "Measurement Tool" on page 279 for details on availability for the selected measurement.
Apply Measurement To	Specifies where the measurement is to be applied from one of the following: • Entire Waveform • Visible X Region • Between Two Cursors (Vertical cursors only)
Remove All Previous Results	Specified that all previous results of that measurement type are to be removed.

Usage Notes

The measurements Delay, Intersect and Slope Intersect can only be applied to a single waveform at any one time.

Where the value Automatic is specified for a field in the Measurement Setup section, a parameter is applied individually to each waveform. If a value is entered instead, this value is applied to all waveforms.

You cannot use horizontal cursors to define parts of a waveform for use with the Measurement Tool.

Refer to “Measurement Tool” on page 279.

Parameter Analyzer Dialog Boxes

Lists the Parameter Analyzer Tool dialog boxes.

Parameter Analyzer Tool Dialog Box 564
Filters Setup Dialog Box 567

Parameter Analyzer Tool Dialog Box

To access: Choose **Tools** > **Parameter Analyzer** from the main menu.

You can use the Parameter Analyzer Tool to generate “pivot” waveforms. That is, you can change the running variable of one or more compound waveform(s) to one of the waveform’s parameter values or to the y values of a different compound waveform and then plot. You can also filter (exclude) individual parameter values and group parameters by color.

Figure 8-33. Parameter Analyzer Dialog Box - Plot Setup Tab

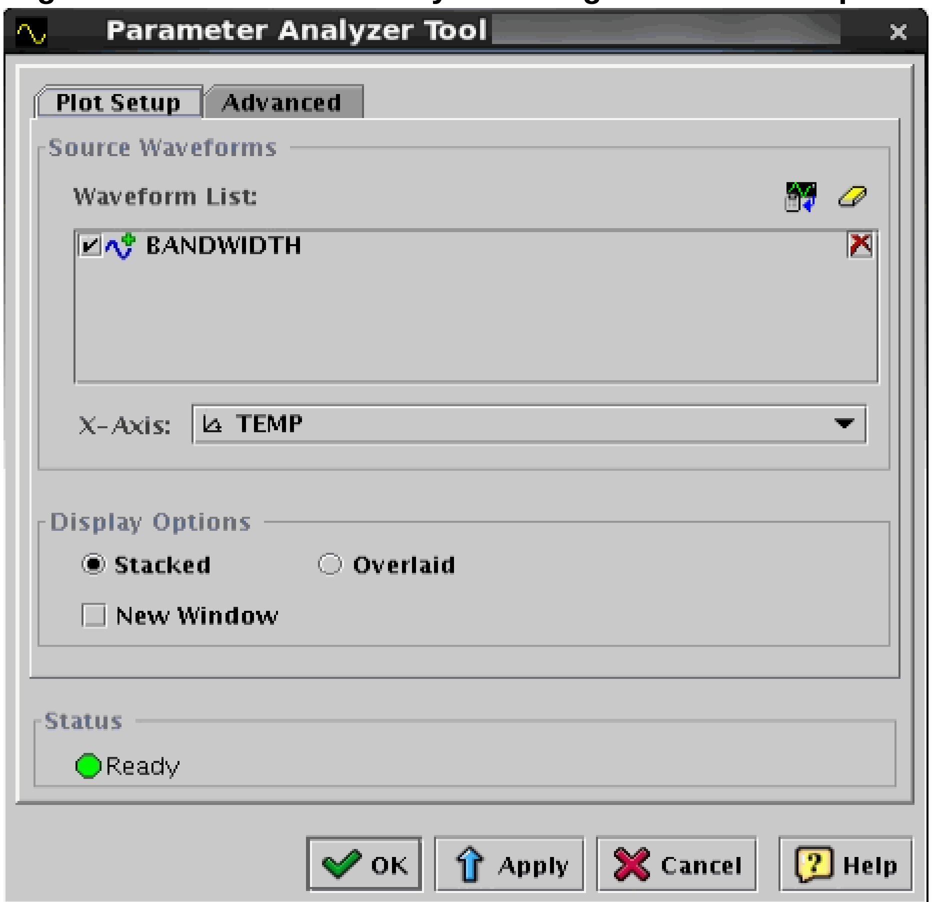
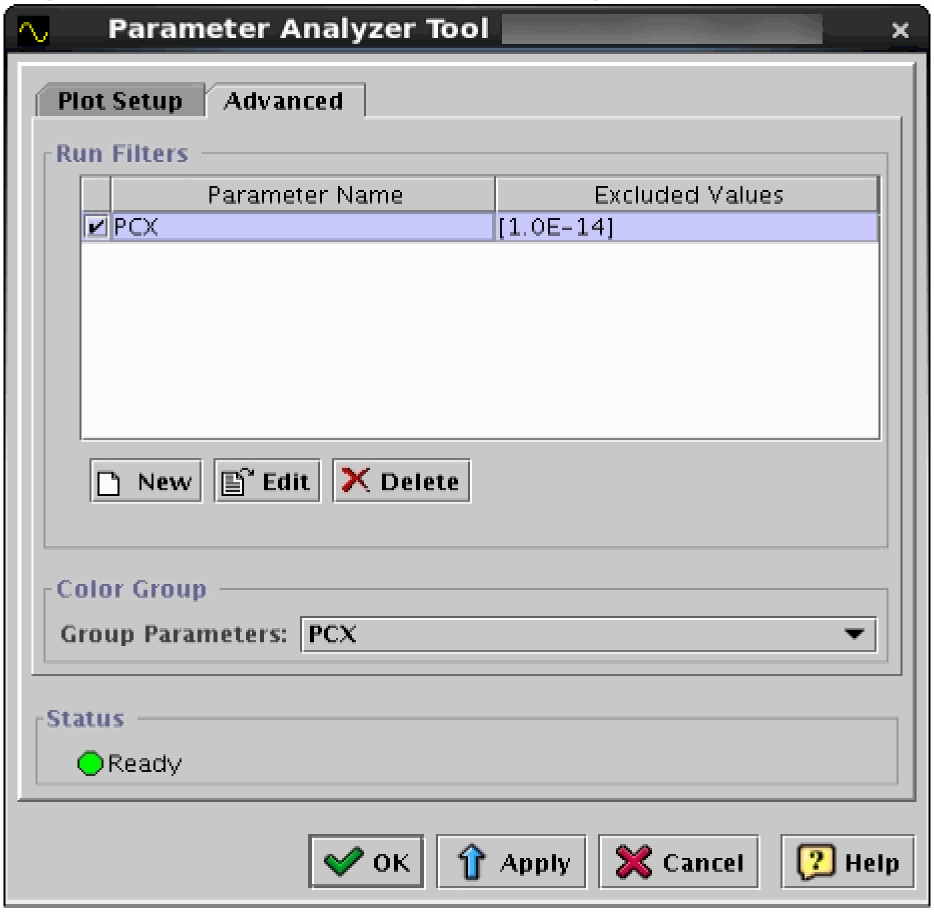


Figure 8-34. Parameter Analyzer Dialog Box - Advanced Tab



Objects

Object	Description
Plot Setup Tab	
Source Waveforms:	
Waveform List	The Add Selected Waveform icon adds the currently selected waveforms to the Waveform List. The Clear Waveform List icon removes all waveforms in the list. The Delete icon removes the adjacent waveform from the list. The checkboxes to the left of each waveform specify whether it will be used as the y axis when a new pivot waveform is generated (in the calc database). Only compound waveforms can be selected for the y axis.

Object	Description
X-Axis	Choose a waveform or parameters to be used as the new x axis for the selected source waveform(s).
Display Options:	
Stacked	Plots all the waveforms separately in different rows.
Overlaid	Plots all the waveforms together in the same row.
New Window	Specifies that the waveforms are plotted in a new window, rather than the current active window.
Advanced Tab	
Run Filters:	
New	Opens the Filters Setup Dialog Box, used to add new filter parameter names and one or more values to be excluded.
Edit	Opens the Filters Setup Dialog Box, used to edit the selected filter parameter and value(s).
Delete	Deletes the selected filter parameter and value(s).
Color Group:	
Group Parameters	Selects a parameter from the list of all the parameters for the current waveform. Waveforms are colored depending on the selected parameter value. All waveforms that share the same value are displayed with same color. See also the -colorgroup parameter of the add wave command.

Usage Notes

Refer to “Analyzing Waveform Parameters to Generate Pivot Waveforms” on page 261.

Related Topics

Filters Setup Dialog Box

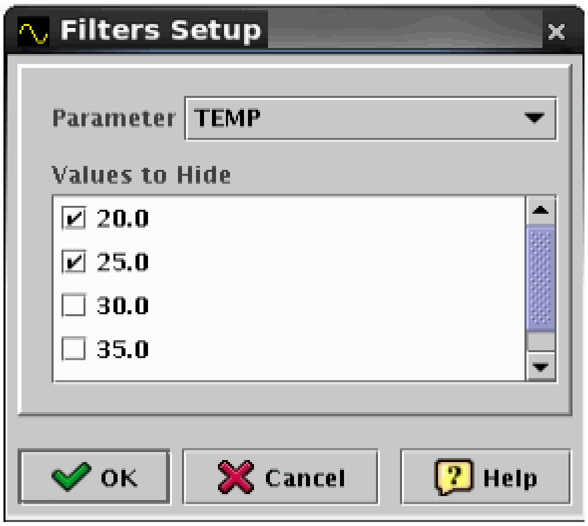
Analyzing Waveform Parameters to Generate Pivot Waveforms

Filters Setup Dialog Box

To access: Choose **Tools** > **Parameter Analyzer** from the main menu and click the **New** button on the **Advanced** tab of the Parameter Analyzer Tool dialog box.

Use this dialog box to select a parameter and specify one or more values to be excluded by the Parameter Analyzer Tool.

Figure 8-35. Filters Setup Dialog Box



Objects

Object	Description
Parameter	Choose the required sweep variable from which to exclude sweep points. Alternatively, choose **Tag** to exclude nested sweep points by reference to the run ID or tag number.
Values to Hide	Choose parameter values to hide.

Usage Notes

Refer to "Analyzing Waveform Parameters to Generate Pivot Waveforms" on page 261 and "Parameter Analyzer Tool Dialog Box" on page 564.

Related Topics

Parameter Analyzer Tool Dialog Box

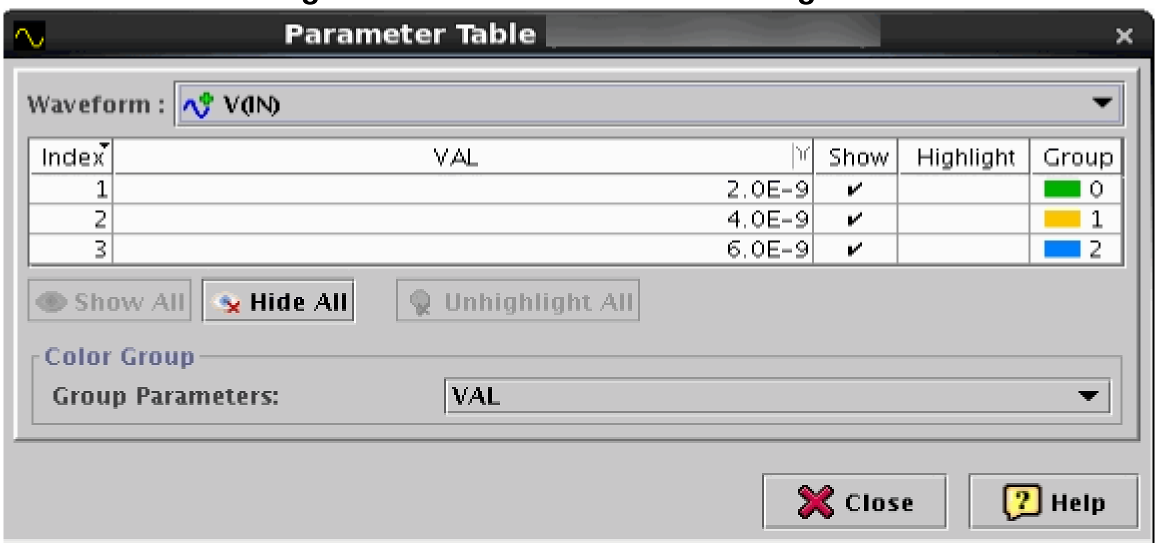
Analyzing Waveform Parameters to Generate Pivot Waveforms

Parameter Table Dialog Box

To access: Right-click a compound waveform name and select **Parameter Table**.

Use this dialog box to filter the runs that are displayed. You can also view the parameter values and runs of a compound waveform, and define a group of parameter names where waveform elements (runs) with identical sets of parameter values will then be plotted using the same color.

Figure 8-36. Parameter Table Dialog Box



Objects

Table 8-79. Parameter Table Dialog Box Contents

Field	Description
Waveform	Dropdown list selects the compound waveform.
Parameter Table	You can sort the contents of the parameter table by clicking on each of the column headings: • Index - shows the parameter run index • Parameter(s) - one column is displayed for each parameter name • Show - enables you display only the selected runs • Highlight - enables you highlight the selected run • Group - Displays the associated run color, based on the color group. You can filter the runs that are displayed by clicking on the icon at the right of a parameter column heading. Refer to the "Filter Dialog Box" on page 550. If a filter is currently in use, the icon changes to .
Show All/Hide All	You can show all runs or hide all runs in the compound waveform.
Unhighlight All	If you have highlighted a run, you can unhighlight it.
Color Group	

Table 8-79. Parameter Table Dialog Box Contents (cont.)

Field	Description
Group Parameters	Selects a parameter from the list of all the parameters for the current waveform. Waveforms are colored depending on the selected parameter value. All waveforms that share the same value are displayed with same color. See also the -colorgroup parameter of the add wave command.

Usage Notes

Show and Highlight

You can choose whether to retain the selected Show or Highlight settings options after a compound waveform **File** > **Reload** (or Ctrl+R) operation. Choose **Edit** > **Options** > **Automatic Reload** and set the Keep Parameter Table Show/Highlight Filters option.

Color Group Example

A compound waveform, Wf, has the following runs and parameters:

1. Wf_1: TEMP = 20, PCX = 1
2. Wf_2: TEMP = 20, PCX = 2
3. Wf_3: TEMP = 20, PCX = 3
4. Wf_4: TEMP = 25, PCX = 1
5. Wf_5: TEMP = 25, PCX = 2
6. Wf_6: TEMP = 25, PCX = 3

If you add TEMP to the Color Group, then waveforms Wf_1, Wf_2 and Wf_3 will use color1 and waveforms Wf_4, Wf_5 and Wf_6 will use color2.

If you add TEMP and PCX to the Color Group, then the waveforms will all have different colors since there are no waveforms that have the same TEMP and PCX values. However, if there is also the following waveform run:

7. Wf_7: TEMP = 20, PCX = 1

then this will be plotted with the same color as Wf_1.

Related Topics

Run Filter Dialog Box

Filter Dialog Box

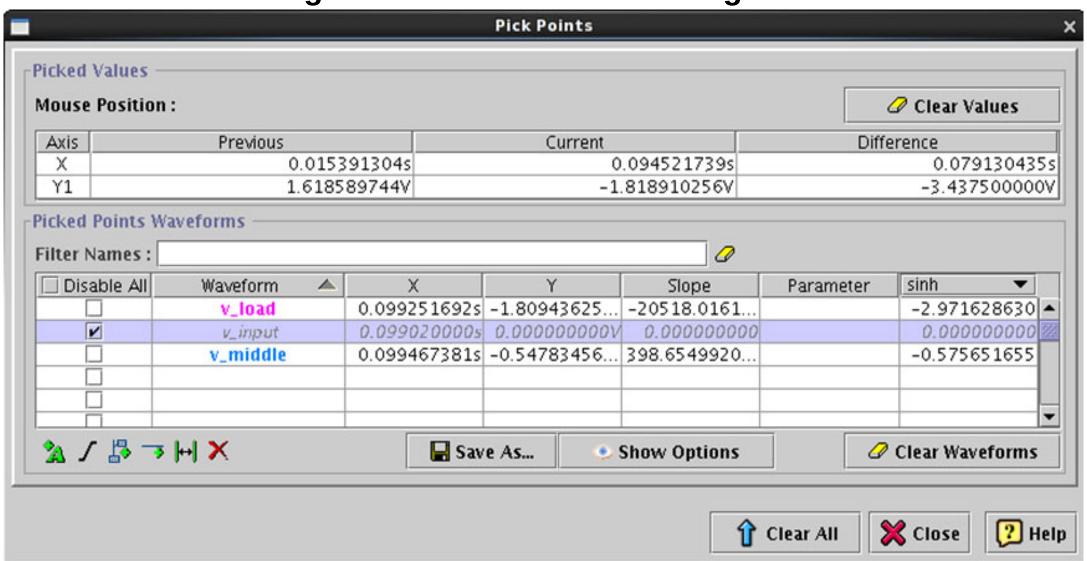
Automatic Reload Options

Pick Points Dialog Box

To access: Choose **Tools** > **Pick Points** or use the icon in the main toolbar  to activate Pick Points mode. You can also use the keyboard shortcut “p”.

Use this dialog box to place and manage pick points on a waveform, enabling quick measurements to be made.

Figure 8-37. Pick Points Dialog Box



Objects

Table 8-80. Pick Points Dialog Box Contents

Field	Description
Picked Values	
Mouse Position	Shows the X and Y coordinates of the current mouse pointer position in the waveform row.
Axis	The axis label. This is dependent upon the waveform type: • X or Y for standard waveforms • Real, Imag for smith charts, polar charts or other complex waveforms.
Previous	The X or Y coordinate of the previous mouse-click.
Current	The X or Y coordinate of the last mouse-click.
Difference	The difference between the current and previous values for each axis.
Clear Values	Clears all listed values from the Picked Values table.
Picked Points Waveforms	

Table 8-80. Pick Points Dialog Box Contents (cont.)

Field	Description
Filter Names	Used to filter the list of results to those matching the specified term.
Disable All	Disables the waveform(s) from pick point mode. The row is grayed out and no new pick points can be added to that waveform. A global button in the column header disables all waveforms in the list.
Waveform	The name of the waveform containing pick points. Click the column header to sort the list by waveform name.
X/Y	The X and Y coordinates of the most recently added pick point on the waveform.
Real/Imag	Values for smith chart or complex waveform types.
Slope	The value of the slope measurement calculated for the most recently added pick point coordinates for each waveform.
Parameter	Value of the parameter, if present, (for X/Y plot or smith chart) calculated for the most recently added pick point coordinates on each waveform.
Expression	Assigned expression value, calculated for the most recently added pick point coordinates for each waveform. Click the Expression column header in the Picked Points Waveforms list to assign an expression to all waveforms in the list. Choose "Expression" from the list to disable expressions on all waveforms. Refer to Table 8-81 on page 574 for a list of available expressions and their calculation methods.
Add Text Annotation	Adds a text annotation to the waveform at the position of each pick point in the current active graph window. The annotation text contains the x and y coordinates of the pick point, and the slope, parameter and expression values (if present). See "Using Text Annotations" on page 189.
Draw Slope	Adds the slope marker to all pick point coordinates on all waveforms in the current active graph window. See "Measuring Slope" on page 294.
Add Cursor(s)	Adds a vertical cursor at the x coordinate of the pick points on all waveforms in the current active graph window.
Add Horizontal Cursor(s)	Adds a horizontal cursor at the y coordinate of the pick points on all waveforms in the current active graph window.
Add Delta	Adds a delta marker between all pick points on all waveform(s) in the order they were created.

Table 8-80. Pick Points Dialog Box Contents (cont.)

Field	Description
 Delete all objects plotted over pick point markers	Removes all annotations, slope markers cursors and delta markers currently attached to pick points in the current active graph window. Pick point markers are not removed.
Clear Waveforms	Clears the Picked Points Waveforms table and removes all pick points from waveforms in the current active graph window. The Filter Names text field is also cleared.
Save As	Saves the contents of the Picked Points Waveforms table to a **.txt* or **.csv* file.
Show Options	Expands the dialog box to expose pick point mode and mouse pointer options.
Clear All	Clears all listed values from the Picked Values table, clears the Picked Points Waveforms table and Filter Names field, and removes all pick points from waveforms in the current active graph window.
Close	Closes the dialog box and deactivates pick points mode. All pick points are removed from waveforms. Pressing the Esc key will also close the dialog box.
Pick Point Mode (Click Show Options to View)	
Follow Waveform(s)	When enabled, the mouse pointer is attached to the waveform nearest to it, and can only be moved around in the graph window in the selected mode: Snap to Data Points or Interpolate. When not enabled, the mouse pointer can be moved freely within the row. Pick points will be placed at the closest point (real or interpolated) on the waveform(s) in the row, starting with the waveform nearest to the position of the mouse-click.
Snap to Data Points	The pick pointer is attached to the current closest waveform in the waveform row. It can be moved only to the position on the waveform, where a data point is present. Pick points are placed on the waveform(s) at the data point closest to the mouse-click position. This is the default setting. If multiple waveforms are present in the same row, one pick point will be placed on each waveform in the row at the closest to the mouse-click position data point, starting from the nearest waveform to the mouse-click position.

Table 8-80. Pick Points Dialog Box Contents (cont.)

Field	Description
Interpolate	The pick pointer is attached to the current closest waveform in the waveform row. It can be moved anywhere on the waveform. Pick points are placed on the waveform(s) at the interpolated point closest to the mouse-click position (minimal distance on x and y axes). If multiple waveforms are present in the same row, one pick point will be placed on each waveform in the row at the closest interpolated point to the mouse-click position, starting from the nearest to the mouse-click position on the waveform. This setting is not used when waveforms use scattered or spectral drawing modes. In this case, Snap to Data Points mode is used.
Pick Point Mouse Pointer Options (Click Show Options to View)	
Use Mouse Pointer Axis	Specifies how the mouse pointer is displayed in Pick Points mode and how pick points are added in the graph window: • X&Y (default) Both x and y bars and corresponding coordinates of the mouse position are used to calculate the closest data point and to add the pick point(s). Pick points will be added only on the waveforms plotted in the row which is currently under the mouse pointer position. • X Only the vertical bar (X coordinate) of the mouse pointer is used to calculate the closest data point and to add the pick point(s). If this mode is selected, one pick point will be added at the same X coordinate of each waveform in the active graph window (not just the row under the mouse pointer). If X&Y axes are selected for digital waveforms, the horizontal bar of the mouse pointer will be displayed and moved according to the digital waveform events (high, middle, low) but only the W coordinate will participate in the closest data point calculation.

Usage Notes

Only the last picked point for each waveform in the waveform row(s) will be present in the Picked Points Waveforms list.

The information in the Pick Points dialog box will reflect only the picked points of the currently active graph window.

When a waveform is hidden, it is disabled in the Pick Points dialog box. Any existing pick points on this waveform are also disabled, and new pick points cannot be added.

If an automatic reload occurs, all existing picked points placed on waveforms which need to be reloaded will be removed. If “Keep N” mode is selected in the EZwave Display Preferences - Automatic Reload Options, any pick points placed on the “kept” waveforms will not be removed. If you choose **File** > **Reload**, all existing pick points are removed in all cases.

You cannot save/reload pick points using TCL or SWD scripts.

The following table lists the available expressions and their methods of calculation. In the calculation method, *wf* is the waveform from the Pick Points Waveforms list for which the expression value needs to be calculated, and *x* and *y* are the x and y coordinates of the last pick point of given waveform.

Table 8-81. Expression Calculations

Expression	Calculation Method
sin	*sin(y)*
cos	*cos (y)*
tan	*tan(y)*
sinh	*sinh(y)*
cosh	*cosh(y)*
tanh	*tanh(y)*
asin	*asin(y)*
acos	*acos(y)*
atan	*atan(y)*
asinh	*asinh(y)*
acosh	*acosh(y)*
atanh	*atanh(y)*
abs	*abs(y)*
drv	*yval(drv(wf), x)*
exp	*exp(y)*
frexp	*frexp(y)*
ln	*ln(y)*
log	*log(y)*
modf	*modf(y)*
pow10	*yval(pow10(wf), x)*

Table 8-81. Expression Calculations (cont.)

Expression	Calculation Method
sqr	*sqr(y)*
sqrt	*sqrt(y)*

Power Analysis Dialog Box

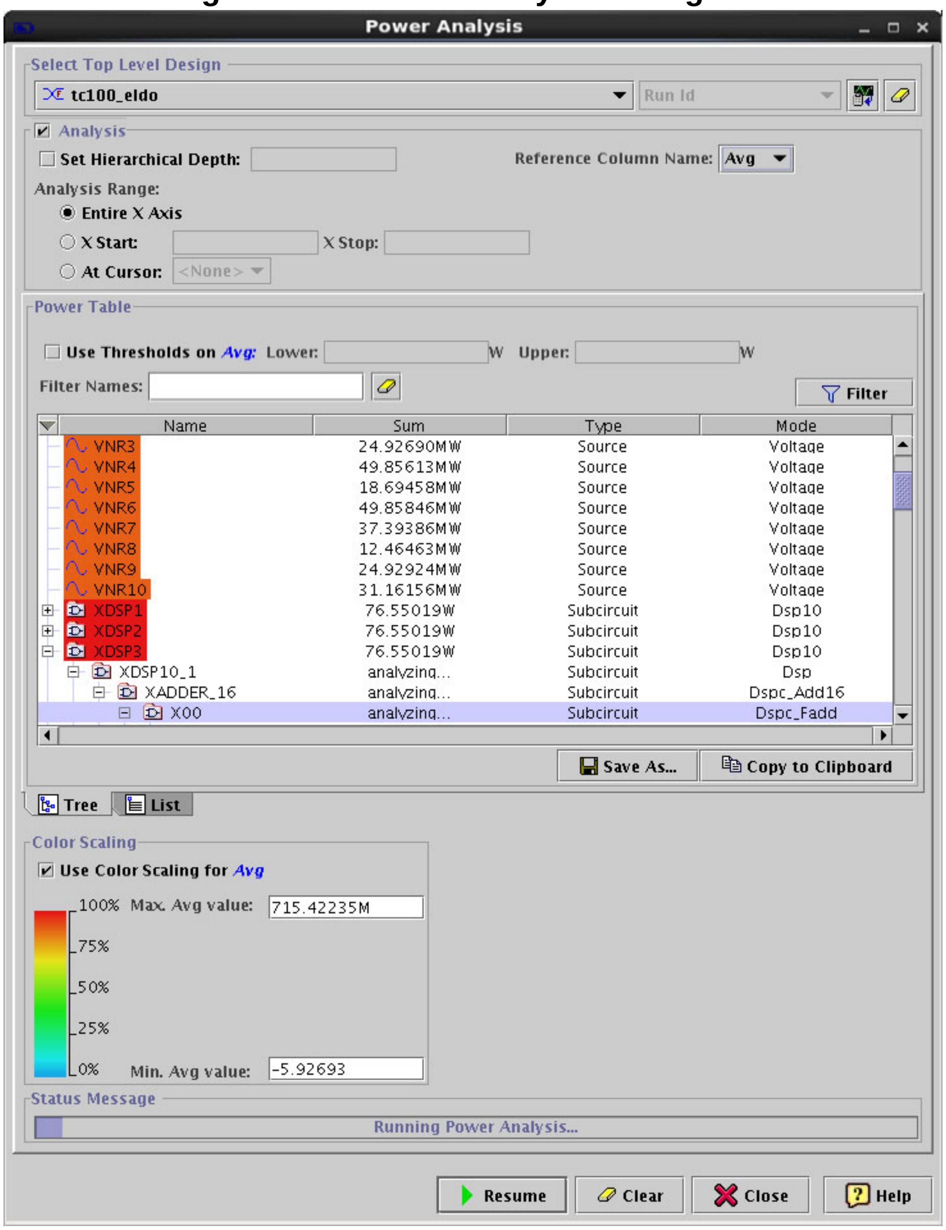
To access: Expand a set of power analysis results (named POWER_<analysis_type>) in the Waveform List Panel, right-click the folder at the required level of hierarchy and choose **Power Analysis**. Alternatively, choose **Tools** > **Power Analysis**.

Used to analyze power consumption of a circuit and its components over time.

Tip

Refer to "Analyzing Power Consumption" on page 268.

Figure 8-38. Power Analysis Dialog Box



Objects

Table 8-82. Power Analysis Dialog Box Contents

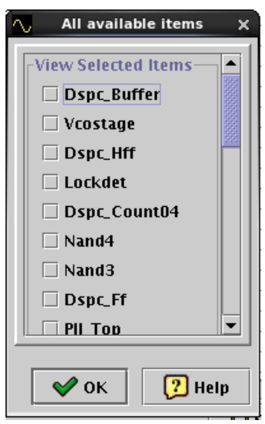
Field	Description
Select Top Level Design	Limits the results to those listed below the selected Top Level Design. The Add Selected Hierarchies icon selects the top level design. The Clear Hierarchy List icon clears the power analysis hierarchy and the analysis data.
Run Id	For compound waveforms, specify the run.
Analysis	
Set Hierarchical Depth	Limits the analysis to the specified number of levels from the top down. Results coming from parts of the circuit located at a deeper level of hierarchy will not be listed.
Reference Column Name	Specifies the reference column name in the Power Table on which you can specify power thresholds and color scaling. Choose from Min, Max, Avg, Sum, and Integ. For RF power analysis, choose Sum or an individual harmonic H(n).
Analysis Range	
Entire X Axis	Specifies the entire input waveform range to be analyzed.
X Start	Specifies the start of the range for the input waveform to be analyzed. Results outside of this range will not be listed.
X End	Specifies the end of the range for the input waveform to be analyzed. Results outside of this range will not be listed.
At Cursor	Specifies a single point to analyze the input waveforms. You can drag the cursor to dynamically analyze results at different points.
Power Table	
Use Thresholds on Min | Max | Avg | Sum | Integ	If checked, specifies using thresholds on the value of the waveform in the range given by Lower and Upper.
Lower	Specifies a minimum power consumption value. Results below the specified threshold will not be listed.
Upper	Specifies a maximum power consumption value. Results above the specified threshold will not be listed.
Filter Names	Used to filter the list of results to those matching the specified term.
Filter	Used to filter the list of results to those matching the specified term.
Save As	Saves the result of the analysis to a text or CSV file.

Table 8-82. Power Analysis Dialog Box Contents (cont.)

Field	Description
Copy to Clipboard	Copies the results to the clipboard.
Color Scaling	
Use Color Scaling for Min | Max | Avg | Sum | Integ	If checked, enables a color scale to be associated with column values in the Power Table. The column is selected from the Reference Column Name dropdown list.
Max. Avg value	Sets the power value representing the highest value of the color scale.
Min. Avg value	Sets the minimum power value representing the lowest value of the color scale.
Analyze/Pause/Resume	Analyze/Pause/Resume depending on the current state of the dialog box.
Clear	Clear the results of the previous analysis.

Usage Notes

- Provides an interactive display of results from power analyses, requested by the SPICE command .POWER_ANALYSIS in Eldo or Questa ADMS.
- In the Power Table, you can filter table entries on particular Type or Mode values by right-clicking on the Type or Mode column heading to open a popup menu, and checking the values you want to show. If there are more than ten different values, a **More** item becomes available. Clicking on this opens the Power Analysis Selection dialog box showing the full set of selectable values for that field.



Tip

See also Tutorial—Using Power Analysis for Static Leakage Analysis of a PLL Circuit in the *Eldo User's Manual* and .POWER_ANALYSIS in the *Eldo Reference Manual.*

Related Topics

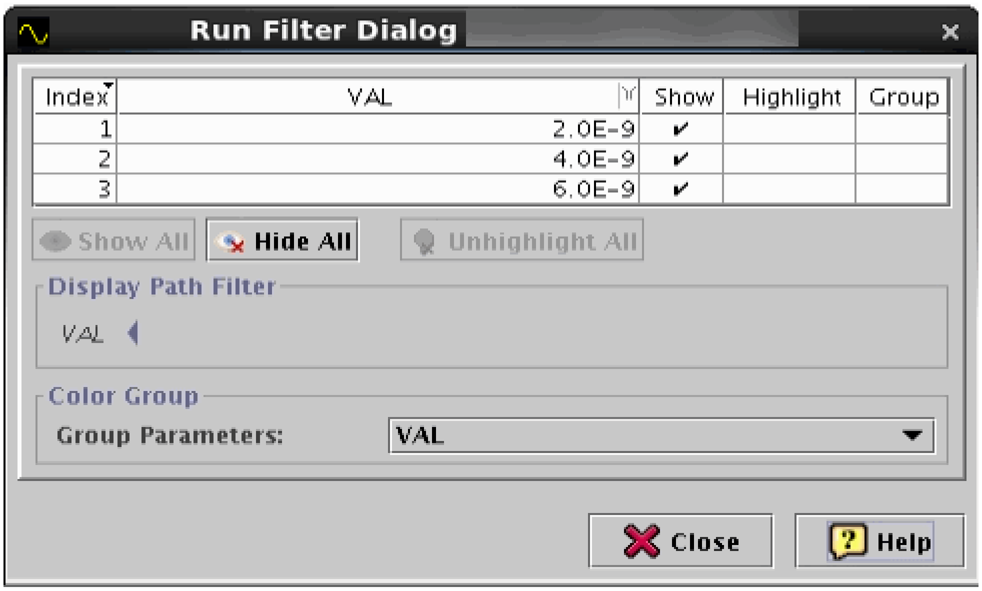
Analyzing Power Consumption

Run Filter Dialog Box

To access: Right-click a database or folder that contains multiple runs and choose **Run Filter** from the popup menu.

Use this dialog box to select which runs are to be shown, highlighted or grouped before you plot them.

Figure 8-39. Run Filter Dialog Box



Objects

Table 8-83. Run Filter Dialog Box Contents

Field	Description
Parameter Table	You can sort the contents of the parameter table by clicking on each of the column headings: • Index - shows the parameter run index • Parameter(s) - one column is displayed for each parameter name • Show - enables you display only the selected runs • Highlight - enables you highlight the selected run • Group - Displays the associated run color, based on the color group. You can filter the runs that are displayed by clicking on the icon at the right of a parameter column heading. Refer to the “Filter Dialog Box” on page 550. If a filter is currently in use, the icon changes to .
Show All/Hide All	You can show all runs or hide all runs in the compound waveform.
Unhighlight All	If you have highlighted a run, you can unhighlight it.
Display Path Filter	

Table 8-83. Run Filter Dialog Box Contents (cont.)

Field	Description
Path Filter	You can remove specific components of the sweep path. Applies to all sweep waveforms plotted from the waveform database.
Color Group	
Group Parameters	Selects a parameter from the list of all the parameters for the current waveform. Waveforms are colored depending on the selected parameter value. All waveforms that share the same value are displayed with same color.See also the -colorgroup parameter of the add wave command.

Usage Notes

The icon for the database or folder changes to indicate that a filter is currently applied. You can also view the parameter values and runs of a compound waveform, and define a group of parameter names where waveform elements (runs) with identical sets of parameter values will then be plotted using the same color.

Show and Highlight

You can choose whether to retain the selected Show or Highlight settings options after a compound waveform **File** > **Reload** (or Ctrl+R) operation. Choose **Edit** > **Options** > **Automatic Reload** and set the Keep Parameter Table Show/Highlight Filters option.

Color Group Example

A compound waveform, Wf, has the following runs and parameters:

1. Wf_1: TEMP = 20, PCX = 1
2. Wf_2: TEMP = 20, PCX = 2
3. Wf_3: TEMP = 20, PCX = 3
4. Wf_4: TEMP = 25, PCX = 1
5. Wf_5: TEMP = 25, PCX = 2
6. Wf_6: TEMP = 25, PCX = 3

If you add TEMP to the Color Group, then waveforms Wf_1, Wf_2 and Wf_3 will use color1 and waveforms Wf_4, Wf_5 and Wf_6 will use color2.

If you add TEMP and PCX to the Color Group, then the waveforms will all have different colors since there are no waveforms that have the same TEMP and PCX values. However, if there is also the following waveform run:

7. Wf_7: TEMP = 20, PCX = 1

then this will be plotted with the same color as Wf_1.

Related Topics

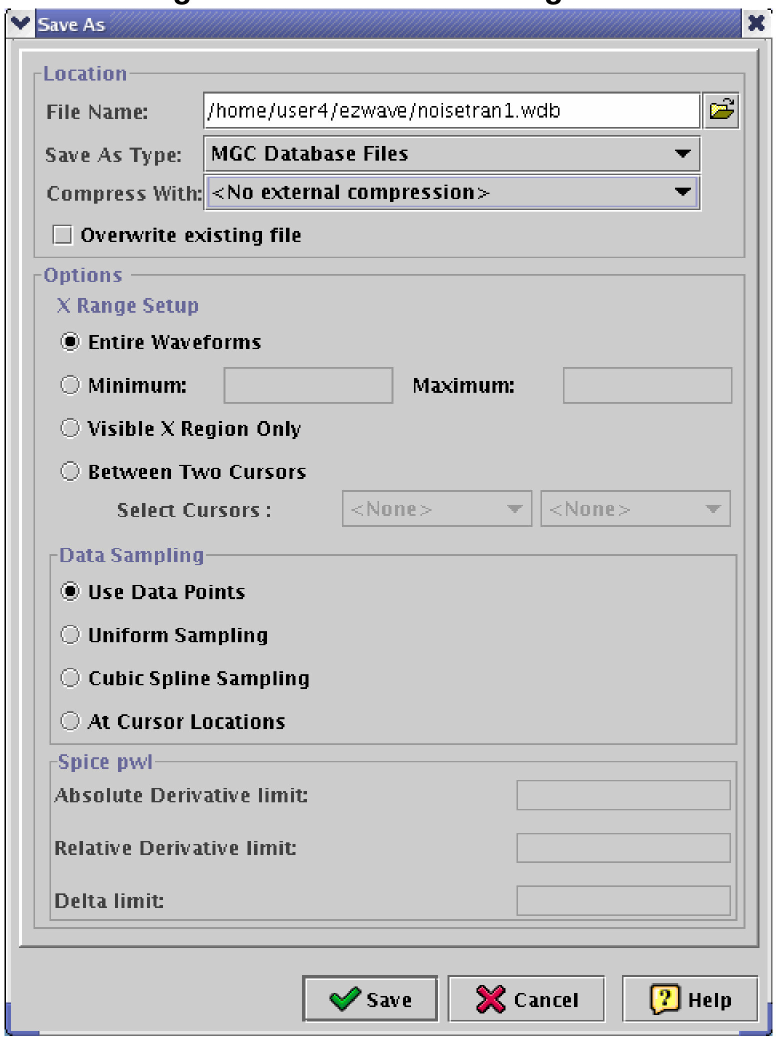
Parameter Table Dialog Box

Save As Dialog Box

To access: Right-click a waveform, waveform name or database, then choose **Save As** from the pop-up menu.

Use this dialog box to specify what portion of the waveform(s) is recorded to the save file, and to set wave-specific options for the save.

Figure 8-40. Save As Dialog Box



Objects

Table 8-84. Save As Dialog Box Contents

Field	Description
Location	
File Name	Specifies he name of the file to save. Type the name, including the extension, or click to browse for a path name and filename.

Table 8-84. Save As Dialog Box Contents (cont.)

Field	Description
Save as Type	Specifies that the file is saved as one of the following: • MGC Database Files (*.wdb*) • SPICE PWL (*.sti*). Only time-based waveforms can be saved. • TXT (Text File) (*.txt*) • CSV (Comma delimited) (*.csv*)
Compress With	Specifies that the save file is compressed in gzip format. For example, a gzipped MGC Database File (*.wdb.gz*).
Overwrite existing file	Specifies that new data is saved over the old file if the file already exists.
Options	
X Range Setup	Specifies the portion of the waveform to save as one of the following: • Entire Waveforms • Minimum / Maximum — The area between two specified X values. Type the X values in the text boxes. If you type a value that is beyond the range of the plotted waveform, the value is set to the minimum or maximum value, as appropriate. • **Visible X Region Only** — Only available for plotted waveforms. • Between Two Cursors — The area between two vertical cursors Only available for plotted waveforms. Use the dropdown lists to select the two cursors to save between. All cursors on the waveform selected are listed in the dropdown lists, even cursors that are not currently visible in the window.

Table 8-84. Save As Dialog Box Contents (cont.)

Field	Description
Data Sampling	Specifies the sampling method as one of the following: • Use Data Points — Specifies that only existing waveform data points are saved. No interpolation or filtering is performed. • Uniform Sampling — Specifies that data points are saved based on a sampling of the waveform. When you select this option, an additional field displays, enabling you to type a sampling value and select a sampling unit from a dropdown list for each x-axis unit being saved. • Cubic Spline Sampling — Specifies that interpolated points are computed using the Cubic Spline method rather than linear interpolation. • At Cursor Locations — Specifies that the cursors on the waveform(s) being saved are used as the data points. When you select this option, an additional field displays, enabling you to select the cursors to use.
Spice pwl	
Absolute Derivative Limit	Specifies the absolute tolerance and enable filtering of output points by performing a slope based comparison. Only available if the file type is Spice PWL (*.sti*).
Relative Derivative Limit	Specifies the relative tolerance and enable filtering of output points by performing a slope based comparison. Only available if the file type is Spice PWL (*.sti*).
Delta Limit	Specifies the threshold the lower limit threshold and enable filtering of output points by performing an absolute variation comparison. Only available if the file type is Spice PWL (*.sti*).

Usage Notes

- For details on customizing the **Save As** options, refer to"Save Data Options" on page 501.
- When saving a database as a SPICE PWL file (*.sti*), only time-based waveforms that contain voltage or current sources are saved. If a database contains waveforms with neither voltage or current sources, those waveforms are ignored.
- When compressing a saved file in gzipped format the original file is not removed. This can be removed manually to reduce the amount of disk space required.

Save Windows Dialog Box

To access: Choose **File** > **Save** from the main menu.

Use this dialog box to save the currently open window(s).

Objects

Table 8-85. Save Windows Dialog Box Contents

Field	Description
File Name	Specifies the path to the file in which to save the window(s)
Overwrite existing file	Specifies that if a file with the same path and name exists, it is overwritten.
Save in TCL	Saves the window in TCL (*.tcl*) format
Save in SWD	Saves the window in SWD (*.swd*) format.
Save Active Window	Only the currently active window is saved.
Save All Windows	All open windows are saved.
Save Related Database	Specifies to also save a database file (*.wdb*) corresponding to the waveforms in the window or windows being saved. This option is only available when Save in SWD is selected.

Usage Notes

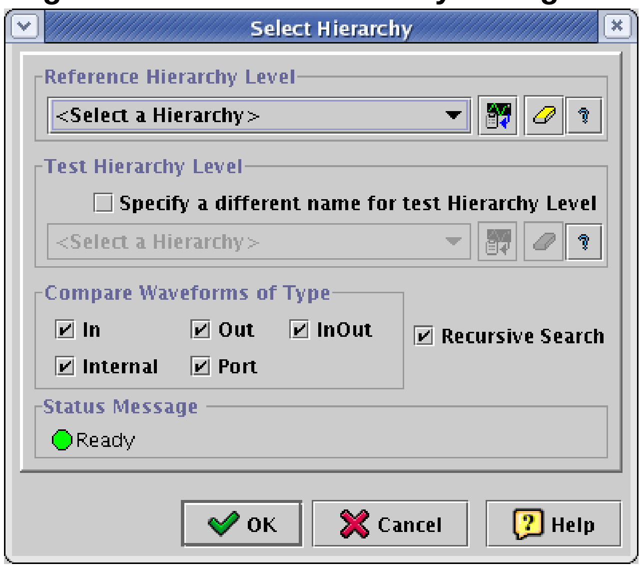
If the window is saved as a TCL script, or a SWD script without the related database, the path to the database (*.wdb* file) within the script may be written as an absolute path, a path relative to the environment variable AMS_EZDO_ROOT or a path relative to the saved script, depending on the Save As Paths settings in Save Data Options.

Select Hierarchy Dialog Box

To access: Choose **Tools** > **Waveform Compare** > **Add** > **Compare by Hierarchy** from the main menu.

This dialog box is part of the Waveform Compare Tool. Use it to specify the reference and test hierarchies for waveform comparison.

Figure 8-41. Select Hierarchy Dialog Box



Objects

Table 8-86. Select Hierarchy Dialog Box Contents

Field	Description
Reference Hierarchy Level	
Select a Hierarchy	Specifies the reference design hierarchy for waveform comparison. Select the hierarchy from the reference database in the upper part of the Waveform List, and click the Add Selected Hierarchies icon .
Test Hierarchy Level	
Specify a different name for the test hierarchy level	Specifies a test hierarchy with a different name to the reference hierarchy. Unchecked by default.
Select a Hierarchy	Specifies the test design hierarchy for waveform comparison. Select the hierarchy from the reference database in the upper part of the Waveform List, and click the Add Selected Hierarchies icon .

Table 8-86. Select Hierarchy Dialog Box Contents (cont.)

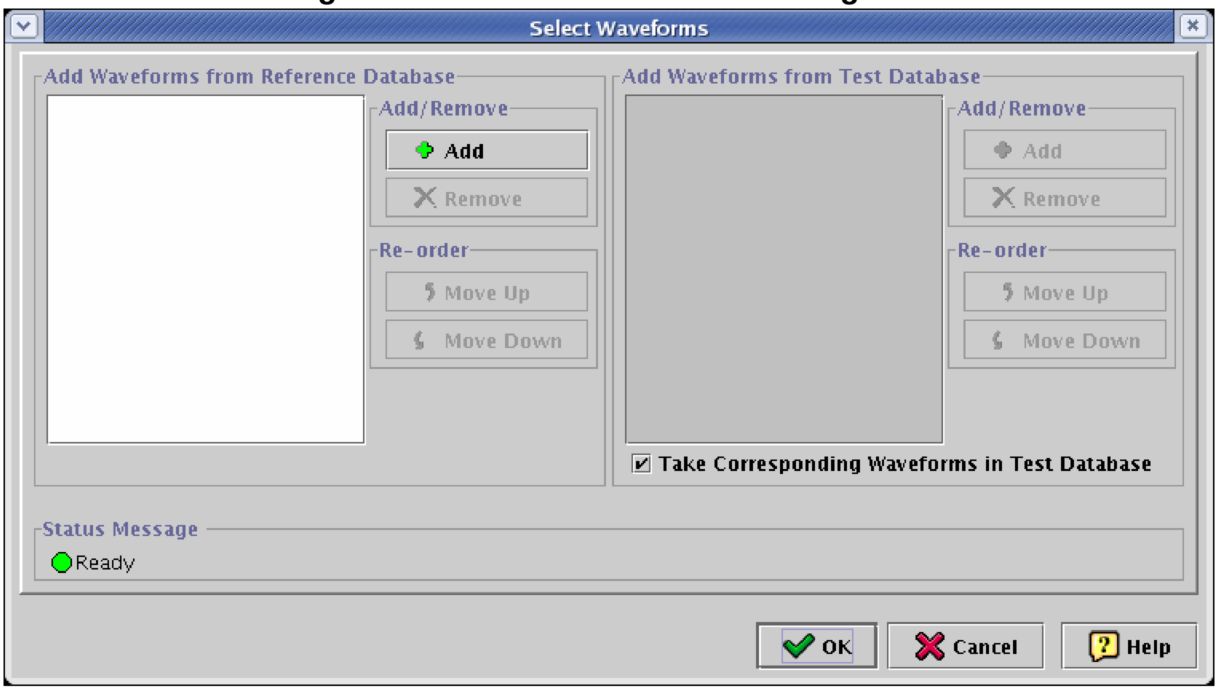
Field	Description
Compare Waveforms of Type	Specifies the types of waveforms to include in the comparison. Select one or more of: • In • Out • InOut • Internal • Port
Recursive Search	Specifies that the Waveform Compare Tool searches the hierarchies recursively during the waveform comparison.

Select Waveforms Dialog Box

To access: Choose **Tools** > **Waveform Compare** > **Add** > **Compare by Waveform** from the main menu.

This dialog box is part of the Waveform Compare Tool. Use it to specify the reference and test waveforms for comparison.

Figure 8-42. Select Waveforms Dialog Box



Objects

Table 8-87. Select Waveforms Dialog Box Contents

Field	Description
Add Waveforms from Reference Database	Specifies the waveforms from the reference database to compare. The reference waveforms to compare are listed on the left. Use the Add, Remove and Re-order buttons to modify this list. The order of the list only matters if the Take Corresponding Waveforms in Test Database options is unchecked.
Add Waveforms from Test Database	Specifies the waveforms from the test database to compare. The white area on the left contains the list of test waveforms to compare. Available if Take Corresponding Waveforms in Test Database is unchecked. Use the Add, Remove and Re-order buttons to modify this list.
Take Corresponding Waveforms in Test Database	Specifies that waveforms with the same name are compared from the reference and test databases. Enabled by default.

Shortcuts Manager Dialog Box

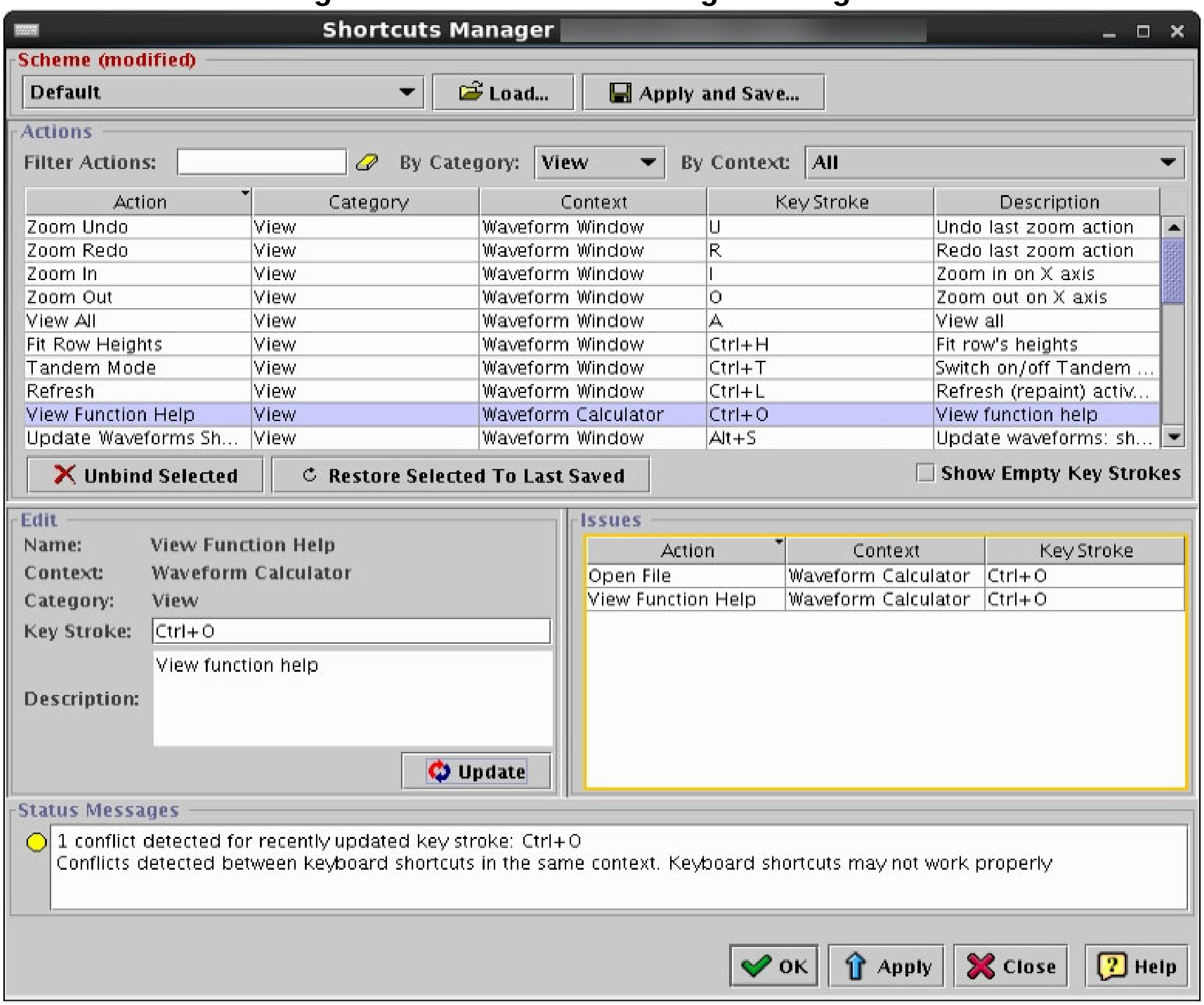
To access: Choose **Edit** > **Shortcuts Manager** from the main menu.

You can modify the default EZwave keyboard shortcuts, assigning key strokes to actions. You can also select alternative key schemes, and import or export key schemes using the **.ezkey* file format.

Tip

Refer to "Configuring Keyboard Shortcuts" on page 64.

Figure 8-43. Shortcuts Manager Dialog Box



Objects

Table 8-88. Shortcuts Manager Dialog Box Contents

Field	Description
Scheme	
Scheme dropdown list	Selects an available keyboard scheme.
Load	Imports a *.*ezkey* keyboard schema file.
Apply and Save	Applies the changes and exports the keyboard schema to a *.*ezkey* file.
Actions	
Filter Actions	Specifies the text used to filter the Action column in the Actions table.
	Click to clear the filter.
By Category dropdown list	Filters the Category column in the Actions table.
By Context dropdown list	Filters the Context column in the Actions table.
Actions table	Displays the Action, its Category and Context, and the associated key stroke together with a Description. Click a column name to sort the table by that column.
Unbind Selection	Removes the associated key stroke from the action selected on the Actions table.
Restore Selected To Last Saved	Reverts the selected Action table row to the last saved value
Show Empty Key Strokes	When checked, filters the Actions table to show only those actions that do not have an associated key stroke.
Edit	
Key Stroke	Specifies the keyboard shortcut for the action. For example, to use Ctrl+S for an action, click in the Key Stroke field, hold down the Ctrl key and press S. Then click the **Update** button.
Description	Specifies the keyboard shortcut description for the action. Click the **Update** button to set.
Issues	
Issues table	Displays the Action, its Context, and the associated Key Stroke for any actions where there are issues that need to be resolved. For example conflicts with the same shortcuts used for actions in the same context. Click a row in the table and correct the issue in the Edit pane.

Related Topics

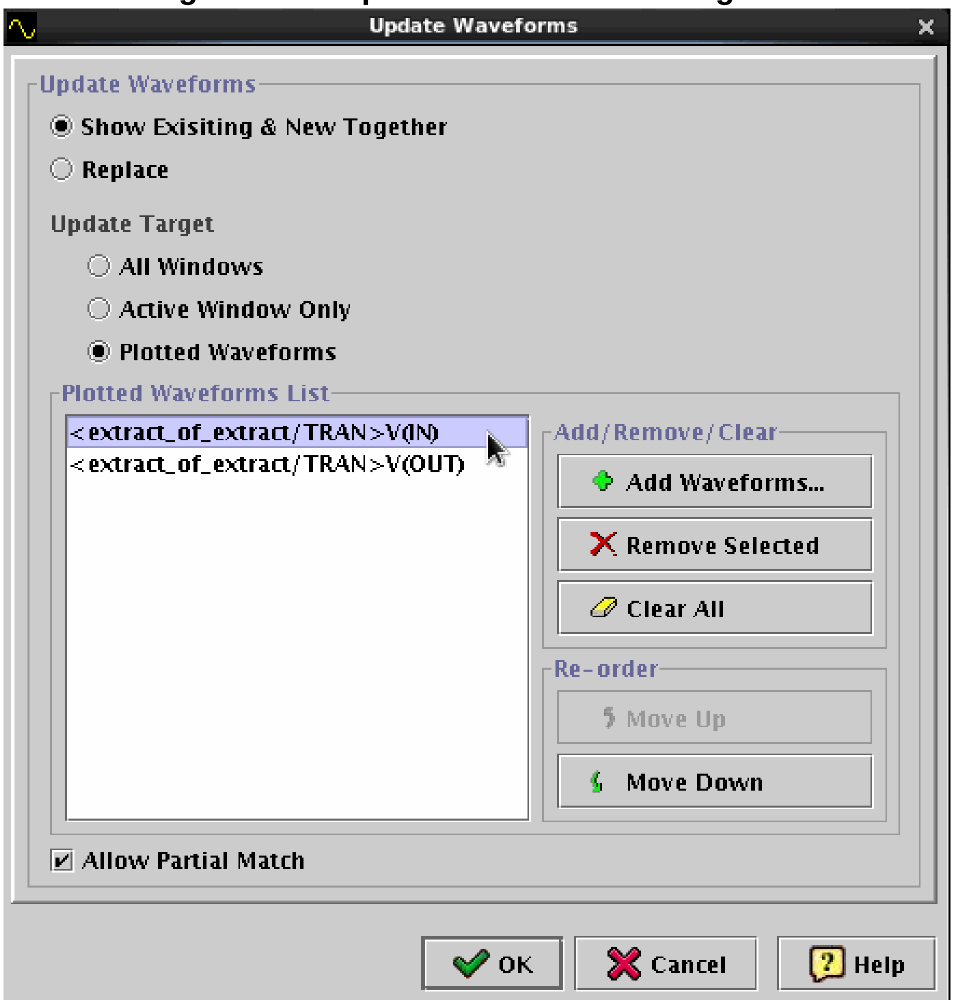
Configuring Keyboard Shortcuts

Update Waveforms Dialog Box

To access: Right-click a waveform database in the Waveform List, and select **Update Waveforms**.

Use this dialog box to update plotted waveforms with new results from subsequent runs. Enables you to compare results from multiple *.wdb* files. Any changes made to the settings on this dialog box are preserved and saved on exit.

Figure 8-44. Update Waveforms Dialog Box



Objects

Table 8-89. Update Waveforms Dialog Box Contents

Field	Description
Show Existing & New Together	Updated waveforms are shown overlaid over the previous result.
Replace	The existing plotted waveforms are replaced with the new result.

Table 8-89. Update Waveforms Dialog Box Contents (cont.)

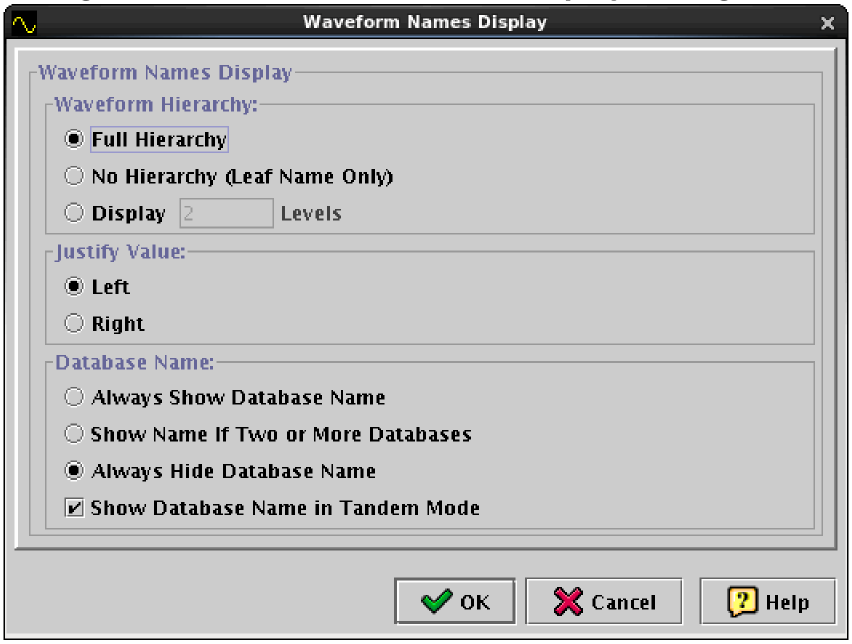
Field	Description
Update Target	Selects which waveforms to apply the update to: • All Windows The waveforms in all open graph windows are updated • Active Window Only The waveforms in the currently active graph window are updated • Plotted Waveforms Only the waveforms specified in the Plotted Waveforms List are updated.
Plotted Waveforms List	Waveforms added to this list are updated when Plotted Waveforms is selected as the Update Target.
Add Waveforms	Opens a dialog box to add waveforms from the currently active Wave windows to the Plotted Waveforms List.
Clear All	Removes all waveforms from the Plotted Waveforms List.
Remove Selected	Removes the currently selected waveforms from the Plotted Waveforms List.
Move Up / Move Down	Used to change the order of waveforms in the Plotted Waveforms List.
Allow Partial Match	Waveforms that partially match the names within the database are updated.

Waveform Names Display Dialog Box

To access: Choose **Format** > **Waveform Names Display** from the main menu.

Use this dialog box to specify how waveform names are displayed. This is a global setting.

Figure 8-45. Waveform Names Display Dialog Box



Objects

Table 8-90. Waveform Names Display Dialog Box Contents

Field	Description
Waveform Names Display	
Waveform Hierarchy	Specifies the hierarchy display as one of the following: • Full Hierarchy — Specifies that the full hierarchy is displayed. Default. • No Hierarchy (Leaf Name Only) — Specifies that only the leaf name is displayed. • Display N Levels — The specified number of levels is displayed.
Justify Value	Specifies how the waveform name is justified when displayed: Left (Default) or Right.

Table 8-90. Waveform Names Display Dialog Box Contents (cont.)

Field	Description
Database Name	Specifies how the database name is displayed from one of the following: • Always Show Database Name. • Show Name If Two or More Databases. Calculated (<calc>) databases are not included in the count of databases. • Always Hide Database Name. Default. • Show Database Name in Tandem Mode. Database names are always displayed in Tandem mode, irrespective of the three settings above. Default.

Usage Notes

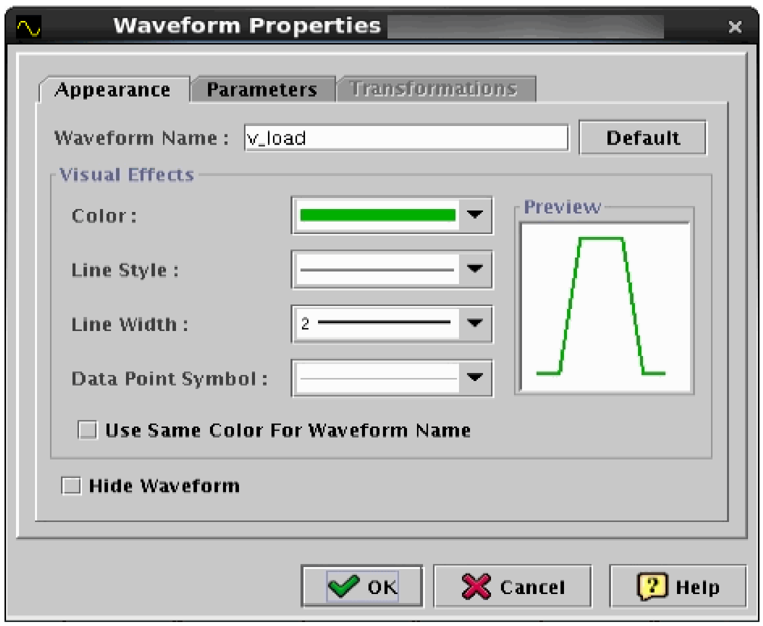
These settings can also be defined as EZwave display preferences. Choose **Edit** > **Options** from the main menu to open the EZwave Display Preferences Dialog Box, then expand the Fonts and Colors folder on the left side, and choose Waveform Name.

Waveform Properties Dialog Box (For Analog Waveforms)

To access: Right-click one or more selected waveforms and select **Properties**.

Use this dialog box to specify the appearance, displays parameters, and transformations applied to waveforms. The Waveform Properties dialog box functions differently depending on whether analog or digital waveform(s), or digital buses or analog waveforms containing integer data, are selected.

Figure 8-46. Waveform Properties Dialog Box - Appearance Tab for Analog Waveforms



Objects

Table 8-91. Appearance Tab for Analog Waveforms Contents

Field	Description
Appearance Tab (Analog Waveform Properties when multiple waveforms are selected)	
Waveform Name	Specifies a unique name for the waveform. The user-defined name will appear in the display and on the tooltip display. Use the **Default** button to reset the waveform name to the full name derived from the database.
Color	Specifies the color of the waveform. Select a color from this dropdown list. Default is Automatic. For compound waveforms, select Match Run to use a single color for all waveforms from the same simulation.

Table 8-91. Appearance Tab for Analog Waveforms Contents (cont.)

Field	Description
Line Style	Specifies the line style of the waveform. Select a style from this dropdown list. Default is Automatic. For compound waveforms, select Match Run to use a single line style for all waveforms from the same simulation. **Note:** Selecting a Line Style other than plain line may significantly reduce waveform display performance.
Line Width	Specifies the default line width of waveforms. Select a width from this dropdown list. **Note:** Selecting a Line Width other than 1 pixel may significantly reduce waveform display performance.
Data Point Symbol	Specifies the data point symbol of the waveform. Select a symbol from this dropdown list.
Use Same Color For Waveform Name	Sets the waveform name label to the same color as the waveform.
Hide Waveform	Temporarily hides the display of the waveform data (the waveform name is not hidden).
Parameters Tab (Not available when multiple waveforms are selected)	
Creation Date	The date that the waveform was created.
Creation Time	The time that the waveform was created.
Transformations Tab (Not available when multiple waveforms are selected)	
dB	The magnitude of each point of the complex waveform calculated in decibels (20 * log (waveform)).
Magnitude	The square root of ($re^2 + im^2$) for each point of the complex waveform.
Real	The real component of each point in the complex waveform.
Imaginary	The imaginary component of each point in the complex waveform.
Phase	The phase of each point in the complex waveform. All phase angles will be between -180 degrees and 180 degrees (-PI radians and PI radians).
Continuous Phase	The phase of each point in the complex waveform that includes an accumulated phase angle from the previous points. Unlike the Phase transformation, this transformation does not include a phase discontinuity at the 180 degree boundaries. This transformation is useful when analyzing waveforms with more than +/- 180 degrees of phase shift.

Table 8-91. Appearance Tab for Analog Waveforms Contents (cont.)

Field	Description
Complex Plane	The complex plane is spanned by the vectors “1” and “i”, where i is the imaginary number. A complex number z=a+ib can be associated to a unique point (a,b) in the complex plane, and a complex waveform of a collection of complex numbers can be represented as a curve (not necessarily monotonic) in the complex plane.
Smith Chart	The complex-valued waveform is plotted on a Smith Chart.
Polar Chart	The complex-valued waveform is plotted in a polar display of a Smith Chart.

Usage Notes

Changing any of the visual effects (line color, width, style or datapoint symbol) in this tab affects all of the selected waveforms. Select Automatic to preserve the settings for each individual waveform. Only the modified properties will apply to all selected waveforms.

If you change the line color, it only applies to that individual display of the wave. Further displays will revert to the default color.

Since color is used to differentiate some states having the same shape, this can result in the loss of visual information in the waveform. This can be addressed by using a cursor to access the value.

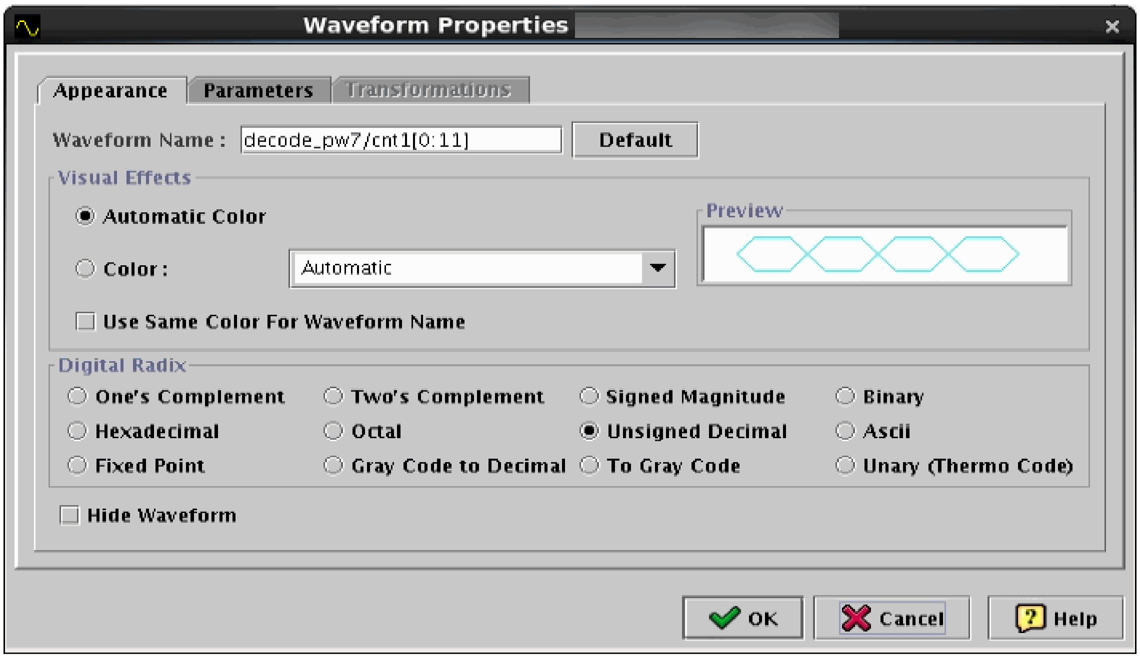
When you use either the Black Background, White Background or Monochrome color scheme (chose **Format** > **Color Scheme**), any changes you make to the visual effects using the Waveform Properties dialog box will apply to all three color schemes. However, a separate set of visual effects is applied when using the Documentation color scheme, enabling you to use different settings for documentation purposes.

Waveform Properties Dialog Box (Digital Waveforms)

To access: Right-click one or more selected waveforms and select **Properties**.

Use this tab to set global options for the appearance of digital waveforms. The Waveform Properties dialog box for digital waveforms contains two tabs: Appearance and Parameters. When more than one waveform is selected, a third tab is available: Digital Waveform Properties.

Figure 8-47. Waveform Properties Dialog Box - Appearance Tab for Digital Waveforms



Objects

Table 8-92. Appearance Tab for Digital Waveform Contents

Field	Description
Appearance Tab (Digital Waveform Properties when multiple waveforms are selected)	
Waveform Name	Specifies a unique name for the waveform. The user-defined name will appear in the display and on the tooltip display. Use the **Default** button to reset the waveform name to the full name derived from the database.
Automatic Color	Specifies the color of the waveform will be set automatically.
Color	Specifies the color of the waveform. Select a color from this dropdown list.

Table 8-92. Appearance Tab for Digital Waveform Contents (cont.)

Field	Description
Use Same Color For Waveform Name	Sets the waveform name label to the same color as the waveform.
Digital Radix (Only available for digital buses)	Specifies the radix used for displaying waveform state values. Choose from: • One's Complement • Two's Complement • Signed Magnitude • Binary • Hexadecimal • Octal • Unsigned Decimal • Ascii • Fixed Point • Gray Code to Decimal • To Gray Code • Unary (Thermo Code)
Hide Waveform	Temporarily hides the display of the waveform data (the waveform name is not hidden).
Parameters Tab (Not available when multiple waveforms are selected)	
Creation Date	The date that the waveform was created.
Creation Time	The time that the waveform was created.

Usage Notes

Changing any of the visual effects in this tab affects all of the selected waveforms. Select Automatic to preserve the settings for each individual waveform. Only the modified properties will apply to all selected waveforms.

If the line color is changed, it only applies to that individual display of the wave. Waveforms added subsequently will use the default color.

Since color is used to differentiate some states having the same shape, this can result in the loss of visual information in the waveform. This can be addressed by using a cursor to access the value or by reverting to the Automatic Color setting.

Waveform Calculator Dialog Boxes

This section describes the dialog boxes accessed from within the Waveform Calculator.

Auto Correlation Dialog Box 602
Chirp Transform Dialog Box 605
Constellation Diagram Dialog Box 609
Convolution Dialog Box 611
Cross Correlation Dialog Box 613
DNA Advisor Dialog Box 615
Error Vector Magnitude and Bit Error Rate Dialog Box 617
Eye Diagram Dialog Box 619
Fast Fourier Transform Tool Dialog Box 621
Harmonic Distortion Dialog Box 626
Histogram Dialog Box 628
Inverse Fast Fourier Transform Dialog Box 630
Phase Noise Dialog Box 632
Power Spectral Density Dialog Box 634
PSS Residue Dialog Box 637
Signal to Noise Ratio Dialog Box 639
Spectrum Measurement Tool Dialog Box 642
Tcl File Viewer Dialog Box 648
Top Noise Dialog Box 650
Windowing Transform Dialog Box 652

Auto Correlation Dialog Box

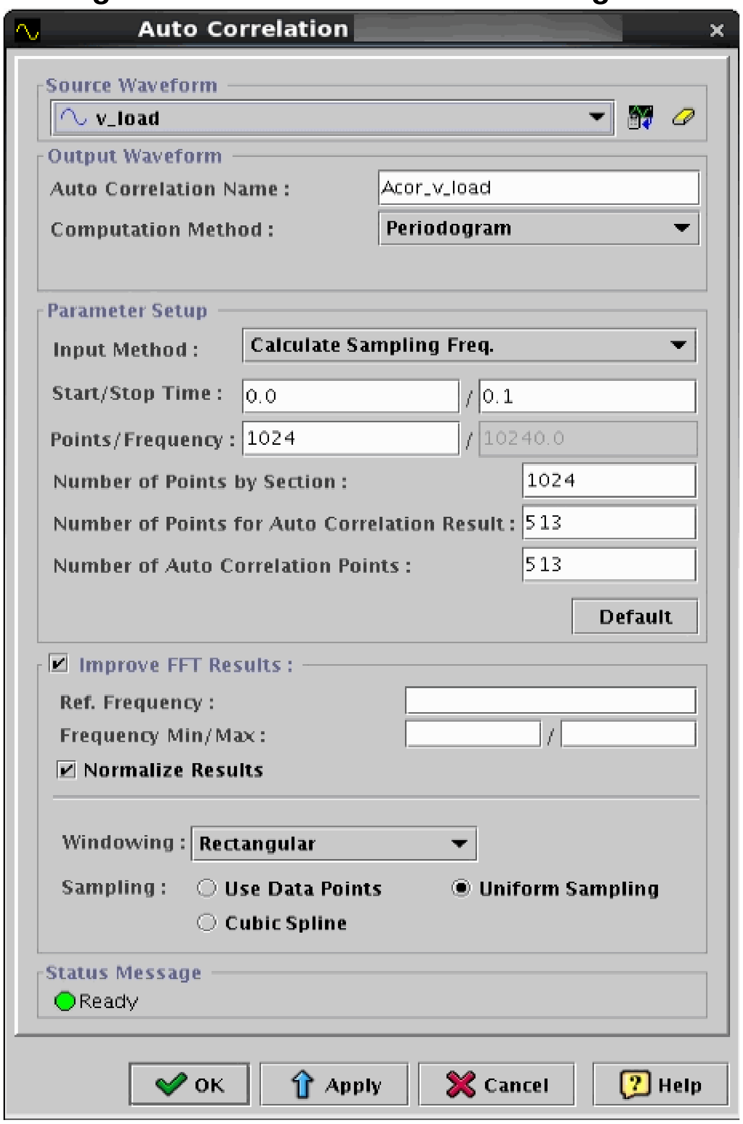
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Signal Processing** from the dropdown list and click the **autocor** button.

Use this dialog box to specify the parameters for the autocorrelation function used in the Waveform Calculator.

Tip

See autocor.

Figure 8-48. Auto Correlation Dialog Box



Objects

Table 8-93. Auto Correlation Dialog Box Contents

Field	Description
Source Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Output Waveform	
Auto Correlation Name	Specifies the name for the AF output waveform.
Computation Method	Specifies the method for calculating the PSD: • Correlogram Method • Periodogram Method Refer to "Autocorrelation Function and Power Spectral Density" on page 391 for more information about these options.
Parameter Setup	
Input Method	Specifies one of the following: • Calculate Time Start • Calculate Time Stop • Calculate Points • Calculate Sampling Freq.
Start /Stop Time	Specifies the start and stop times for the signal.
Points / Frequency	Specifies the number of sampling points and the sampling frequency. The number of sampling points for the FFT results is *(Number of Points)/2*. FFT computations are performed only on those signals having Number of Points set to a factor of 2^n (n = 2, 3, ...). If this is not so, then a slower DFT computation is executed.
Number of Points by Section	Specifies the number of points for the output waveform.
Number of Points for Auto Correlation Result	Specifies the number of points in the AF result (N_{auto}).
Number of Auto Correlation Points	Specifies the number of autocorrelation points used for the PSD computation.
Default	Click to set the default Auto Correlation setup parameters.
Improve FFT Results	
Ref. Frequency	Adjusts results around the y axis so that the point for the specified reference frequency is 0.0.

Table 8-93. Auto Correlation Dialog Box Contents (cont.)

Field	Description
Frequency Min / Max	(Optional) Specifies the start and end frequencies used to display the FFT results. Defaults are "Begin" and "End".
Normalize Results	Specifies that all results are divided by (Number of Points)/2, except for the first point, which is divided by Number of Points.
Windowing	Specifies a windowing transform to apply to the signal from: • Bartlett • Blackman • Blackman-Harris • Dolph-Chebyshev • Hamming • Hanning • Kaiser • Klein • Parzen • Rectangular • Welch Refer to "Windowing Transforms" on page 396 for more details about these options.
Sampling	Specifies the sampling method as one of the following: • Use Data Points — Select if the input data has equidistant Time Steps. The default. • Cubic Spline — Interpolated points are computed using the Cubic Spline method rather than linear interpolation. • Uniform Sampling — Select otherwise.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Chirp Transform Dialog Box

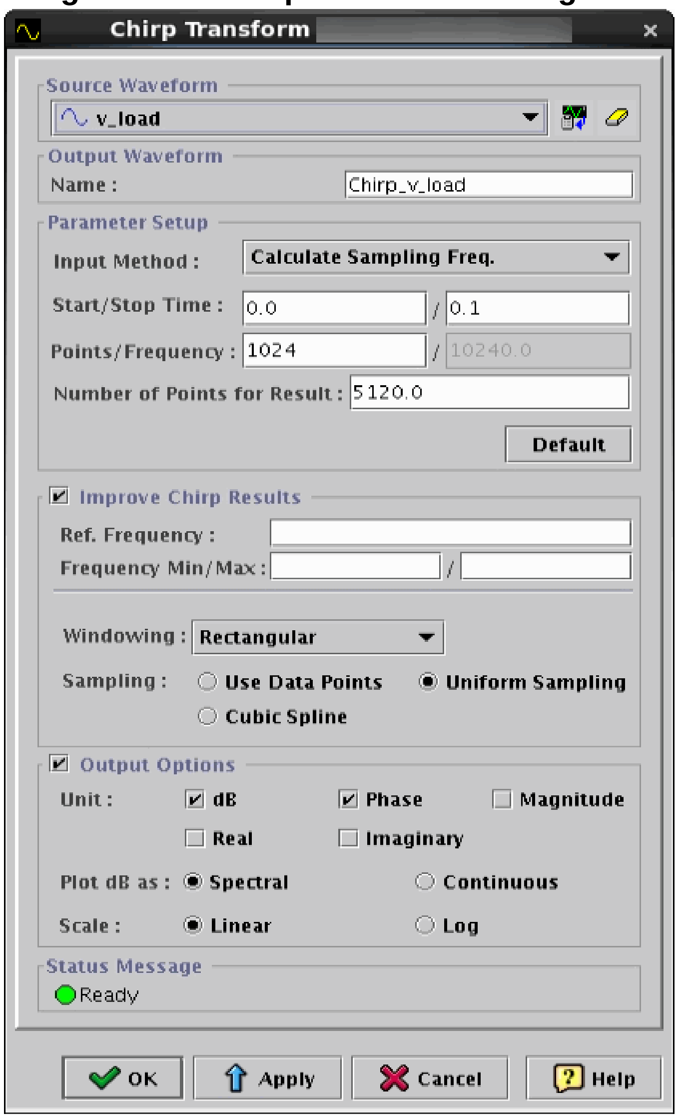
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Signal Processing** from the dropdown list and click the **chirp** button.

Use this dialog box to specify the parameters for the chirp function used in the Waveform Calculator.

Tip

See chirp.

Figure 8-49. Chirp Transform Dialog Box



Objects

Table 8-94. Chirp Transform Dialog Box Contents

Field	Description
Source Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Output Waveform	
Name	Specifies the name for the result waveform.
Parameter Setup	
Input Method	Specifies one of the following: • Calculate Time Start • Calculate Time Stop • Calculate Points • Calculate Sampling Freq.
Start /Stop Time	Specifies the start and stop times for the signal.
Points / Frequency	Specifies the number of sampling points and the sampling frequency. The number of sampling points for the FFT results is *(Number of Points)/2*. FFT computations are performed only on those signals having Number of Points set to a factor of 2^n ($n = 2, 3, ...$). If this is not so, then a slower DFT computation is executed.
Default	Click to set the default Chirp Transform setup parameters.
Number of Points for Result	Specifies the number of points for the output waveform.
Improving Chirp Results	
Ref. Frequency	Adjusts results around the y axis so that the point for the specified reference frequency is 0.0.
Frequency Min / Max	(Optional) Specifies the start and end frequencies used to display the Chirp results. Defaults are “Begin” and “End”.

Table 8-94. Chirp Transform Dialog Box Contents (cont.)

Field	Description
Windowing	Specifies a windowing transform to apply to the output waveform(s) from: • Bartlett • Blackman • Blackman-Harris • Dolph-Chebyshev • Hamming • Hanning • Kaiser • Klein • Parzen • Rectangular • Welch Refer to "Windowing Transforms" on page 396 for more details about these options.
Sampling	Specifies the sampling method as one of the following: • Use Data Points — Select if the input data has equidistant Time Steps. The default. • Cubic Spline — Interpolated points are computed using the Cubic Spline method rather than linear interpolation. • Uniform Sampling — Select otherwise.
Output Options	
Unit	Specifies that the following plot options are enabled: • dB — Specifies result waveform displayed in dB. • Phase — Specifies result waveform for the phase displayed in degrees. • Magnitude — Specifies result waveform for the absolute magnitude of the input waveform displayed. • Real — Displays the real part of the result waveform. • Imaginary — Displays the imaginary part of the result waveform.
Plot dB as	Specifies the drawing mode as one of the following: • Spectral — Specifies the "spectral" drawing mode. The default. • Continuous — Specifies the "continuous" drawing mode.
Scale	Specify the plot scale as either: • Linear • Log

Related Topics

Waveform Calculator

Waveform Calculator GUI

Constellation Diagram Dialog Box

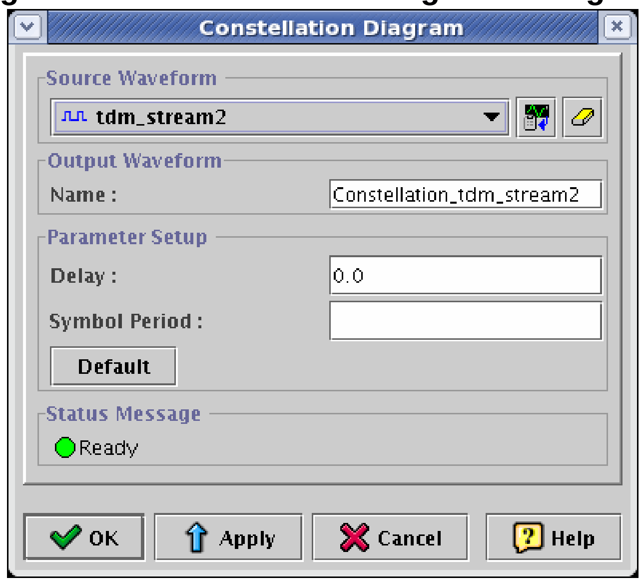
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **RF** from the dropdown list and click the **cd** button.

Use this dialog box to specify the parameters for the constellation diagram function used in the Waveform Calculator.

Tip

See constellationdiagram.

Figure 8-50. Constellation Diagram Dialog Box



Objects

Table 8-95. Constellation Diagram Dialog Box Contents

Field	Description
Source Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Parameter Setup	
Delay	Specifies the initial time delay before the first sampling is taken. This is used as an offset to sample each symbol at the center of each digital data duration.
Symbol Period	Specifies the sampling period for the calculation. This value must be deduced from the circuit.

Usage Notes

The source waveform must be a complex-valued transient waveform. If you have only real-part and imaginary-part (or gain and phase) waveforms, you can reconstruct the complex-valued waveform using the complex function in the Waveform Calculator.

Each scattered dot on a constellation diagram represents a unique symbol, and each symbol represents unique digital data bits. Digital data is parsed into data lengths that encode the symbol waveform. The period between two consecutive symbols can be deduced from the circuit or can be measured from the transient symbol waveform. The Symbol Period is used as a sampling period for generating the constellation diagram.

An inappropriate Delay or Symbol Period selection will result in a bad constellation diagram.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Convolution Dialog Box

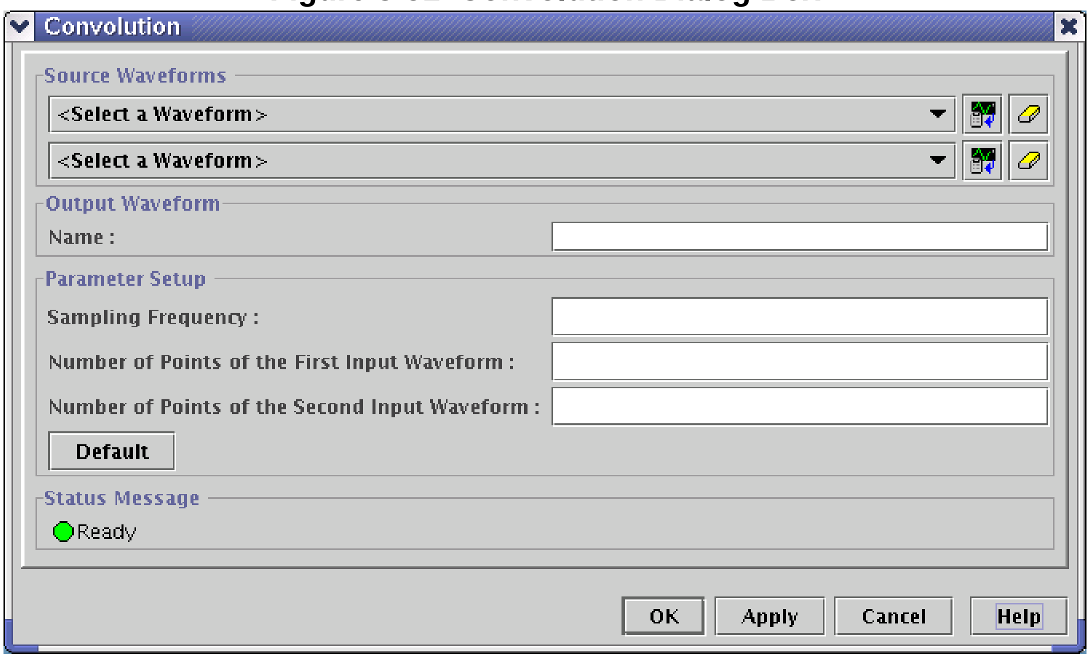
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Signal Processing** from the dropdown list and click the **conv** button.

Use this dialog box to specify the parameters for the convolution function used in the Waveform Calculator.

Tip

Refer to "Convolution Function" on page 392 for details on how the convolution function calculates the discrete linear convolution between two data sets.

Figure 8-51. Convolution Dialog Box



Objects

Table 8-96. Convolution Dialog Box Contents

Field	Description
Source Waveform(s)	Specifies the source waveforms. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Output Waveform	
Name	Specifies the name of the output waveform.
Parameter Setup	
Sampling Frequency	Specifies the sampling frequency of the input datasets.

Table 8-96. Convolution Dialog Box Contents (cont.)

Field	Description
Number of Points of the First Input Waveform	Specifies the number of points used in the first input waveform.
Number of Points of the Second Input Waveform	Specifies the number of points used in the second input waveform.

Usage Notes

All of the data points in the source waveforms must be equidistant. To create a uniformly-sampled waveform data set, use the windowing function in the Waveform Calculator and select the Uniform Sampling option. Refer to "Windowing Transform Dialog Box" on page 652 for details.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Cross Correlation Dialog Box

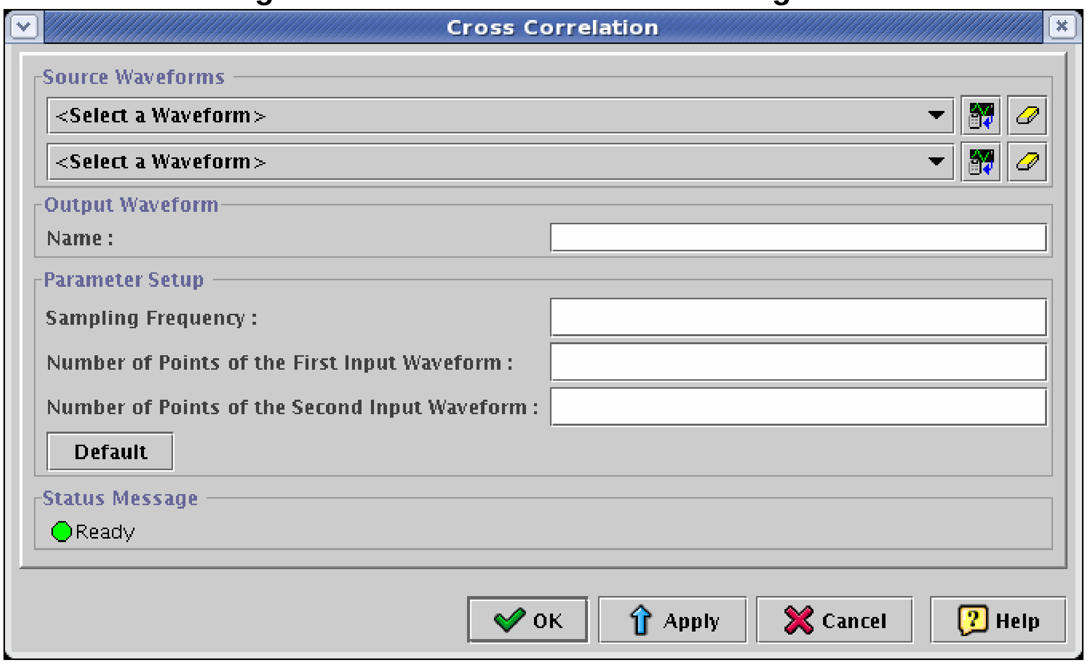
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Signal Processing** from the dropdown list and click the **xcorr** button.

Use this dialog box to specify the parameters for the crosscorrelation function used in the Waveform Calculator.

Tip

See crosscorrelation.

Figure 8-52. Cross Correlation Dialog Box



Objects

Table 8-97. Cross Correlation Dialog Box Contents

Field	Description
Source Waveform(s)	Specifies the source waveforms. The Add Selected Waveform icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Output Waveform	
Name	Specifies the name of the output waveform.
Parameter Setup	
Sampling Frequency	Specifies the sampling frequency of the input datasets.
Number of Points of the First Input Waveform	Specifies the number of points used in the first input waveform.

Table 8-97. Cross Correlation Dialog Box Contents (cont.)

Field	Description
Number of Points of the Second Input Waveform	Specifies the number of points used in the second input waveform.

Usage Notes

All of the data points in the source waveforms must be equidistant. To create a uniformly-sampled waveform data set, use the windowing function in the Waveform Calculator and select the Uniform Sampling option. Refer to "Windowing Transform Dialog Box" on page 652 for details.

Related Topics

Waveform Calculator

Waveform Calculator GUI

DNA Advisor Dialog Box

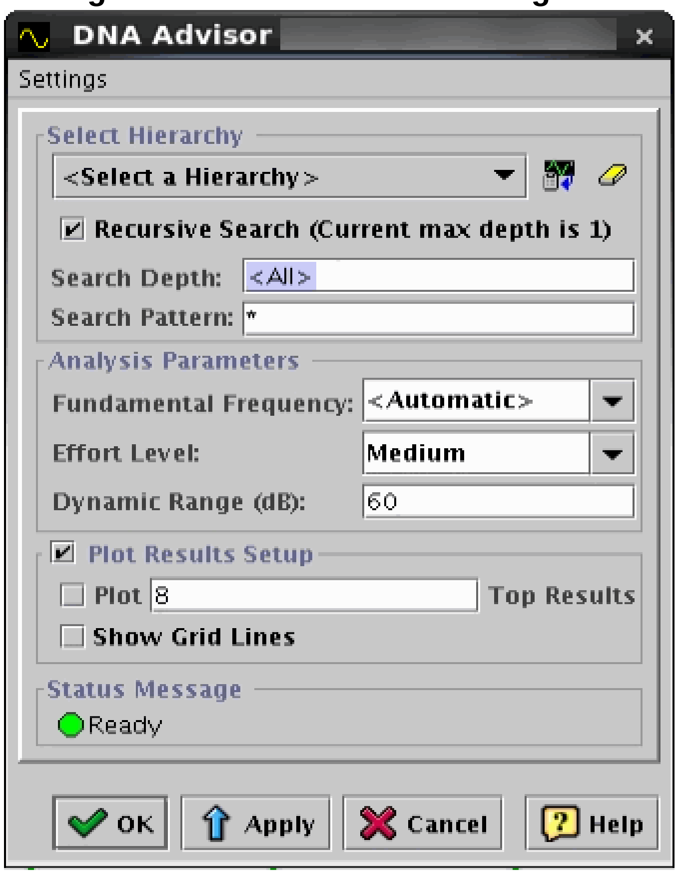
To access: Choose **Tools** > **DNA Advisor** from the main menu. Alternatively, choose **Tools** > **Waveform Calculator** from the main menu, choose **RF** from the dropdown list and click the **dnaadvisor** button.

Use the Device Noise Analysis Advisor (DNA Advisor) to calculate the maximum Fourier frequency for each input waveform. The DNA Advisor analyzes transient or PSS waveforms, recommends the analysis parameters required for an accurate simulation, and highlights analysis methods that are not recommended. The results help you set noise analysis options for both Eldo, AFS and non-Mentor simulators.

Tip

See also the Waveform Calculator harmonicsmeter function.

Figure 8-53. DNA Advisor Dialog Box



Objects

Table 8-98. DNA Advisor Dialog Box Contents

Field	Description
Select Hierarchy	

Table 8-98. DNA Advisor Dialog Box Contents (cont.)

Field	Description
Select a Hierarchy	You can specify a database, folder or waveform hierarchy as the input. Use the **Add Selected Hierarchy** icon to add the currently selected hierarchy or simply drag and drop the required item from the Waveform List into the <Select a Hierarchy> field. The **Clear Hierarchy** icon removes the selected hierarchy.
Recursive Search	Specifies that a recursive waveform search is performed, starting from the currently selected hierarchy. It also displays the maximum depth of the currently selected hierarchy.
Search Depth	Specifies the number of levels deep for the waveform search, starting from the specified hierarchy. Default is <All>, which corresponds to a full recursive search. Use 0 to search only in the selected hierarchy.
Search Pattern	Finds waveform names that match the specified pattern. Default is "*".
Analysis Parameters	
Fundamental Frequency	Specifies the fundamental frequency of the periodic circuit. By default, the entire interval is the fundamental period: 1 / (last(indep(wf)) - first(indep(wf)))
Effort Level	Effort level to run Fourier analysis (number of harmonics). Default is Standard. • Standard = 505 • Medium = 2002 • High = 8008 • Higher = 128000 • Highest = 2048000
Dynamic Range (dB)	Specifies the dynamic range (in dB) required for the circuit (the relative tolerance for finding the upper Fourier frequency limit). Range is 60 to 120 dB inclusive.
Plot Results Setup	
Plot Top Results	Specifies the number of signals to plot, sorted in order of the maximum difference. Default is 8.
Show Grid Lines	Displays grid lines in the results window.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Using the DNA Advisor Tool

Error Vector Magnitude and Bit Error Rate Dialog Box

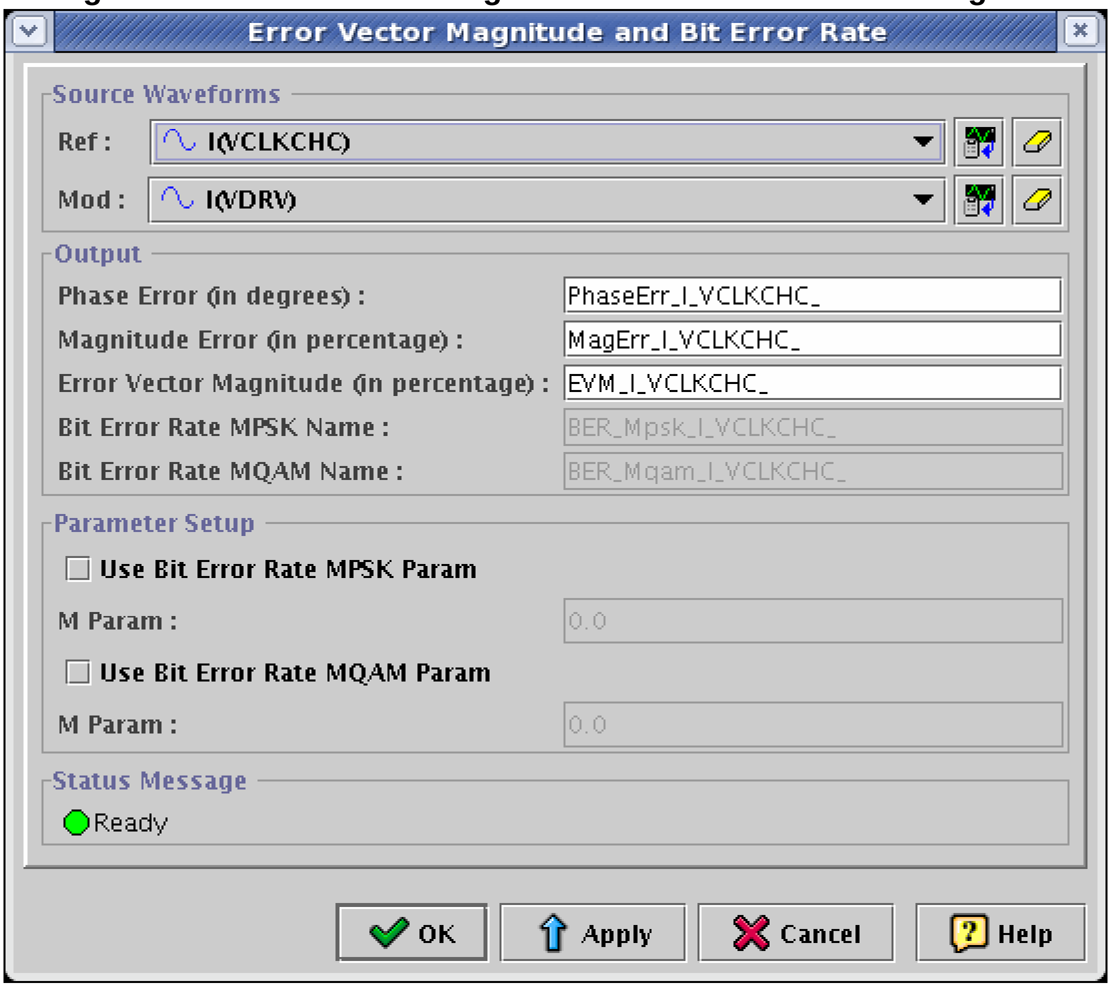
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **RF** from the dropdown list and click the **evmber** button.

Use this dialog box to specify the parameters for the evmber function used in the Waveform Calculator.

Tip

See evmber.

Figure 8-54. Error Vector Magnitude and Bit Error Rate Dialog Box



Objects

Table 8-99. Error Vector Magnitude and Bit Error Rate Dialog Box Contents

Field	Description
Ref / Mod	Specifies the source waveforms. Ref must be the constellation diagram of a reference signal. Mod must be the constellation diagram of a modulated signal. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Output	
Phase Error (in degrees)	Specifies the phase error output variable.
Magnitude Error Name	Specifies the magnitude error output variable.
Error Vector Magnitude Name	Specifies the error vector magnitude output variable.
Bit Error Rate MPSK Name	Specifies the name of the BER MPSK (M-ary Quadrature Amplitude Modulation)-modulated output signal. Only available if Use Bit Error Rate MPSK Param is selected.
Bit Error Rate MQAM Name	Specifies the name of the BER MQAM (M-ary Phase Shift Keying)-modulated output signal. Only available if Use Bit Error Rate MQAM Param is selected.
Parameter Setup	
Use Bit Error Rate MPSK Param	Specifies that the BER for an MPSK (M-ary Quadrature Amplitude Modulation)-modulated signal is estimated. Type the M Param.
Use Bit Error Rate MQAM Param	Specifies that the BER for an MQAM (M-ary Phase Shift Keying)-modulated signal is estimated. Type the M Param.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Eye Diagram Dialog Box

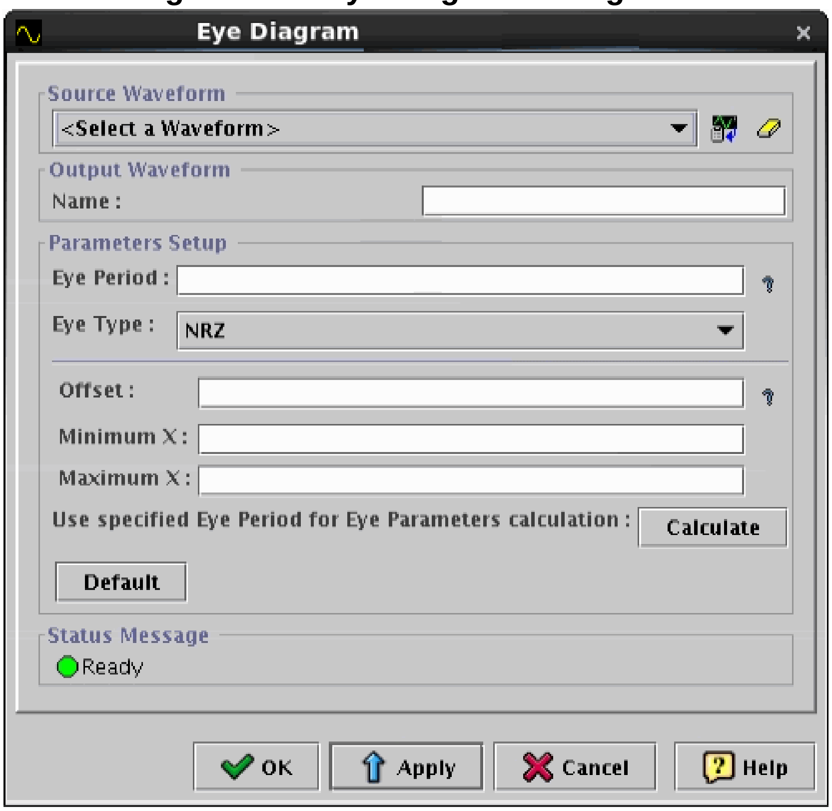
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Statistical** from the dropdown list and click the **eye** button.

Use this dialog box to specify the parameters for the eyediagram function used in the Waveform Calculator.

Tip

See eyediagram.

Figure 8-55. Eye Diagram Dialog Box



Objects

Table 8-100. Eye Diagram Dialog Box Contents

Field	Description
Source Waveform	Specifies the source waveform. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Output Waveform	Specifies the source waveform.
Parameters Setup	

Table 8-100. Eye Diagram Dialog Box Contents (cont.)

Field	Description
Eye Period	Specifies the eye period. The eye diagram is generated by overlaying a semi-periodical waveform signal on an interval in X (usually a time interval). This interval is defined as the eye period. A default Eye Period is calculated based on period divided by 2.
Eye Period	Specifies the eye type, either NRZ or PAM 4.
Offset	Enables shifting of the eye as the open part of the eye is not always at the center of the axes.
Minimum X and Maximum X	Specifies the range of waveform data used for generating the eye diagram.
Calculate	Calculates the Offset, Minimum X, and Maximum X using the Eye Period value and automatically populates the fields.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Fast Fourier Transform Tool Dialog Box

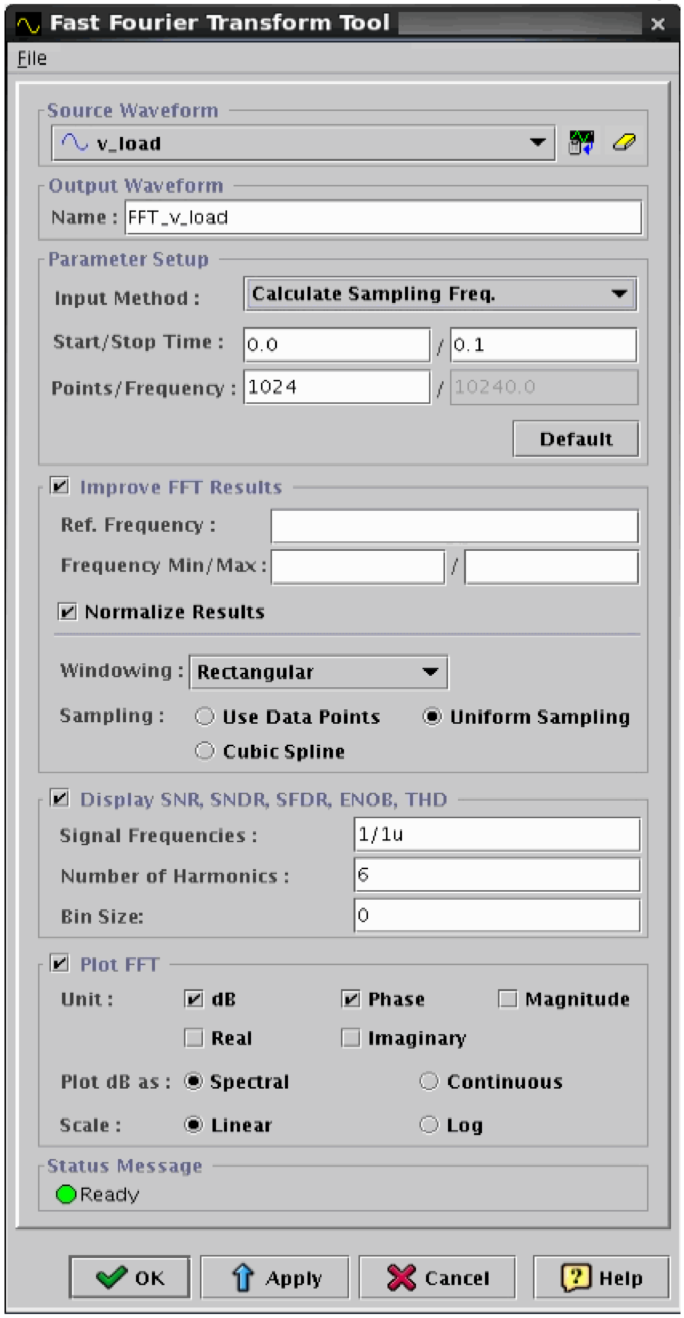
To access: Select **Tools** > **FFT** from the main menu to open the Fast Fourier Transform Tool dialog box. Alternatively, choose **Tools** > **Waveform Calculator** to open the Waveform Calculator, choose **Signal Processing** from the dropdown list and click the **fft** button.

Use this dialog box to specify the parameters for the function used in the Waveform Calculator and FFT tools.

Tip

See "fft" on page 772.

Figure 8-56. Fast Fourier Transform Tool Dialog Box



Objects

Table 8-101. Fast Fourier Transform Tool Dialog Box Contents

Field	Description
File menu	Select **File** > **Open Configuration** to load a FFT configuration. Select **File** > **Save Configuration** to save a FFT configuration. All parameter of the FFT Tool GUI are saved, except the source waveform and the output waveform names, with file extension *.ez_cfg.
Source Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon  adds the currently selected waveform. The **Clear Waveform List** icon  removes all waveforms in the list.
Output Waveform	
Name	Specifies the name of the output waveform.
Parameter Setup	
Input Method	Specifies one of the following: • Calculate Time Start • Calculate Time Stop • Calculate Points • Calculate Sampling Freq.
Start /Stop Time	Specifies the start and stop times for the signal.
Points / Frequency	Specifies the number of sampling points and the sampling frequency. The number of sampling points for the FFT results is *(Points)/2*. If Points is set to a factor of 2^n (n = 2, 3, ...), the FFT computation is more efficient.
Default	Click to set the default FFT setup parameters.
Improve FFT Results	
Ref. Frequency	Adjusts results around the y axis so that the point for the specified reference frequency is 0.0.
Frequency Min / Max	(Optional) Specifies the start and end frequencies used to display the FFT results. Defaults are “Begin” and “End”.
Normalize Results	Specifies that all results are divided by (Number of Points)/2, except for the first point, which is divided by Number of Points.

Table 8-101. Fast Fourier Transform Tool Dialog Box Contents (cont.)

Field	Description
Windowing	Specifies a windowing transform to apply to the signal from: • Bartlett • Blackman • Blackman-Harris • Dolph-Chebyshev • Hamming • Hanning • Kaiser • Klein • Parzen • Rectangular • Welch Refer to “Windowing Transforms” on page 396 for more details about these options.
Sampling	Specifies the sampling method as one of the following: • Use Data Points — Select if the input data has equidistant Time Steps. The default. • Cubic Spline — Interpolated points are computed using the Cubic Spline method rather than linear interpolation. • Uniform Sampling — Select otherwise.
Noise and Distortion Measurements	
Display SNR, SNDR, SFDR, ENOB, THD	Specifies SNR, SNDR (alias for SINAD), SFDR, ENOB, and THD are displayed in the FFT graph area. Applies to non-compound waveforms only.
Signal Frequencies	(Optional) Specifies a comma-separated list of frequencies to be considered as signal for the noise and distortion calculations. Frequencies can be selected outside of Minimum Frequency and Maximum Frequency. Example: `75e6, 200Meg`
Number of Harmonics	(Optional) Specifies the number of harmonics of the signal to be considered. Default value is 6 (signal + 5 harmonics).

Table 8-101. Fast Fourier Transform Tool Dialog Box Contents (cont.)

Field	Description
Bin Size	Specifies the number of points to take into account around the fundamental signal. For example: • with Bin Size = 0, only the fundamental is taken into account • with Bin Size = 1, the frequencies taken into account correspond to the first point to the left and right for the found fundamental and each of its associated harmonics. The default value depends on the Windowing transform selected.
Plot FFT	
Unit	Specifies that the following plot options are enabled: • dB — Specifies result waveform displayed in dB. • Phase — Specifies result waveform for the phase displayed in degrees. • Magnitude — Specifies result waveform for the absolute magnitude of the input waveform displayed. • Real — Displays the real part of the result waveform. • Imaginary — Displays the imaginary part of the result waveform.
Plot as	Specifies the drawing mode as one of the following: • Spectral — Specifies the “spectral” drawing mode. The default. • Continuous — Specifies the “continuous” drawing mode.
Scale	Specify the plot scale as either: • Linear • Log

Usage Notes

To estimate the power density spectrum of a random signal, only a finite part of the signal is used in practice even if the signal is of infinite duration. To reduce the undesirable effects of truncating the data records (leakage), it is convenient to apply different types of Windowing that gradually taper the data near the ends of the record, thereby avoiding the abrupt truncation of a rectangular window.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Using the Fast Fourier Transform Tool

Harmonic Distortion Dialog Box

To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Signal Processing** from the dropdown list and click the **hd** button.

Use this dialog box to specify the parameters for the harmonic distortion function used in the Waveform Calculator.

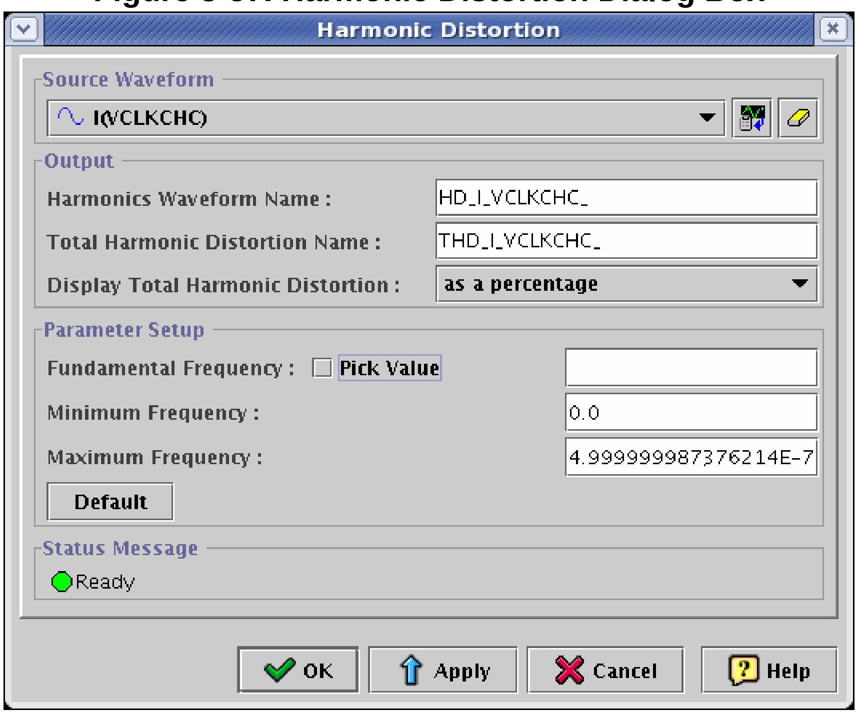
Tip

See harmonicdistortion.

Note

Refer to “Harmonic Distortion Function” on page 394 for details on how the harmonic distortion function computes the harmonics and the total harmonic distortion (THD) of the input waveform signal.

Figure 8-57. Harmonic Distortion Dialog Box



Objects

Table 8-102. Harmonic Distortion Dialog Box Contents

Field	Description
Source Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon  adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list. 

Table 8-102. Harmonic Distortion Dialog Box Contents (cont.)

Field	Description
Output Waveform	
Total Harmonic Distortion Name	Specifies the name of the output variable. This can be used in later computations.
Display Total Harmonic Distortion	Specifies whether THD is shown as a gain (total gain in dB) or as a percentage of the gain of the fundamental signal.
Parameter Setup	
Fundamental Frequency	Specifies the fundamental frequency to be used for the harmonic distortion calculation.
Pick Value	Enables the fundamental frequency to be selected directly from the active waveform. With this option selected, mouse over the active waveform and use the crosshairs that appear along the waveform to select the fundamental frequency. The crosshairs will snap to the nearest data point on the waveform.
Minimum Frequency	Specifies the minimum frequency (fmin) to be used in the calculation.
Maximum Frequency	Specifies the maximum frequency (fmax) to be used in the calculation.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Histogram Dialog Box

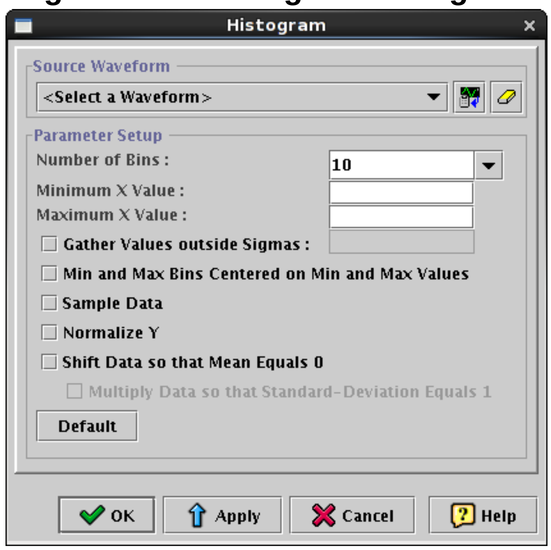
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Statistical** from the dropdown list and click the **histogram** button.

Use this dialog box to specify the parameters for the histogram function used in the Waveform Calculator.

Tip

See histogram.

Figure 8-58. Histogram Dialog Box



Objects

Table 8-103. Histogram Dialog Box Contents

Field	Description
Source Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Parameter Setup	
Number of Bins	Specifies the number of waveform divisions (resolution) to use. The default is 10.
Minimum X Value	Specifies the X value at the lower bound of a window interval.
Maximum X Value	Specifies the X value at the upper bound of a window interval.
Gather Values outside Sigmas”	Restricts the histogram width to (mean +-SIGBIN* standard deviation) and specifies the value of SIGBIN, which is equivalent to the SPICE Eldo SIGBIN parameter.

Table 8-103. Histogram Dialog Box Contents (cont.)

Field	Description
Min and Max Bins Centered on Min and Max Values	When selected, the first (and respectively last) bin is centered on the sample min (resp. max) value. When not selected, the bin is aligned to the right of the min (resp. aligned to the left of the max).
Sample Data	When selected, the input waveform is sampled (the number of points remains the same, but are equally distributed; see the histogram function for details). When not selected, the raw Y values of the input waveform are used.
Normalize Y	When selected, the histogram height is normalized; the sum of the bin height equals 1.0. When not selected, the height of the bins equals the number of hits, and the sum of the bin height equals the total number of points of the input waveform.
Shift Data so the Mean Equals 0	When selected, the Y-values of the input waveform are decreased by the mean() of these values, so the displayed mean equals 0.
Shift and Multiply Data so that Mean Equals 0 and Standard-Deviation Equals 1	When selected, the Y-values of the input waveform are decreased by the mean() of these values, and then divided by the standard-deviation. So the displayed mean equals 0, and the displayed standard-deviation equals 1.0. This option is available only if Shift Data so that Mean Equals 0 is selected, above.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Inverse Fast Fourier Transform Dialog Box

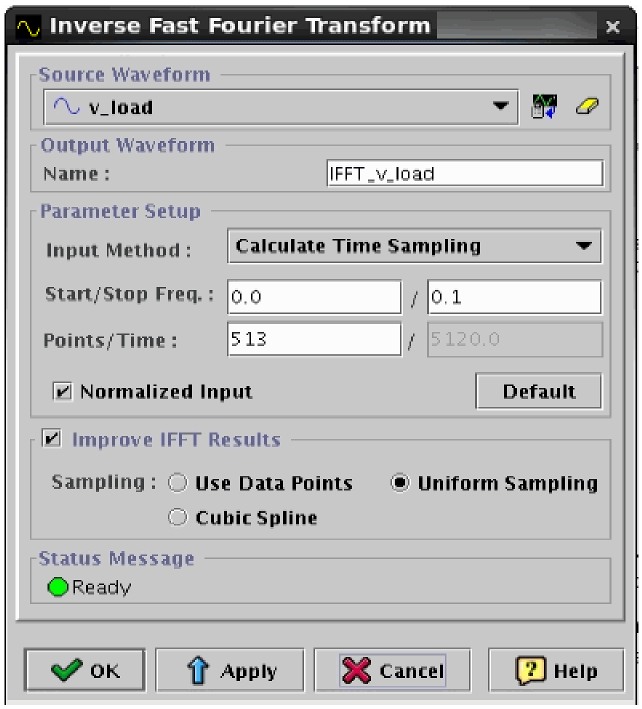
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Signal Processing** from the dropdown list and click the **ifft** button.

Use this dialog box to specify the parameters for the ifft function used in the Waveform Calculator.

Tip

See "ifft" on page 811.

Figure 8-59. Inverse Fast Fourier Transform Dialog Box



Objects

Table 8-104. Inverse Fast Fourier Transform Dialog Box Contents

Field	Description
Source Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Output Waveform	
Name	Specifies the name of the output waveform.
Parameter Setup	

Table 8-104. Inverse Fast Fourier Transform Dialog Box Contents (cont.)

Field	Description
Input Method	Specifies one of the following: • Calculate Start Frequency • Calculate Stop Frequency • Calculate Points • Calculate Time Sampling
Start /Stop Frequency	Specifies the start and stop frequencies for the signal.
Points / Time	Specifies the number of sampling points and the sampling time.
Default	Click to set the default IFFT setup parameters.
Normalized Input	Specifies that all real and imaginary parts of the result are divided by (Number of Points)/2 except for the first point, which is divided by Number of Points.
Improve IFFT Results	
Sampling	Specifies the sampling method as one of the following: • Use Data Points — Select if the input data has equidistant time steps. The default. • Cubic Spline — interpolated points are computed using the Cubic Spline method rather than linear interpolation. • Uniform Sampling — Select otherwise.

Usage Notes

An IFFT analysis always creates results with an even number of points. This means that when calculating results in conjunction with an FFT analysis, an even number of points with the FFT must also be used if the following condition is to be fulfilled:

IFFT(FFT(signal)) = signal

Related Topics

Waveform Calculator

Waveform Calculator GUI

Using the Inverse Fast Fourier Transform Tool

Phase Noise Dialog Box

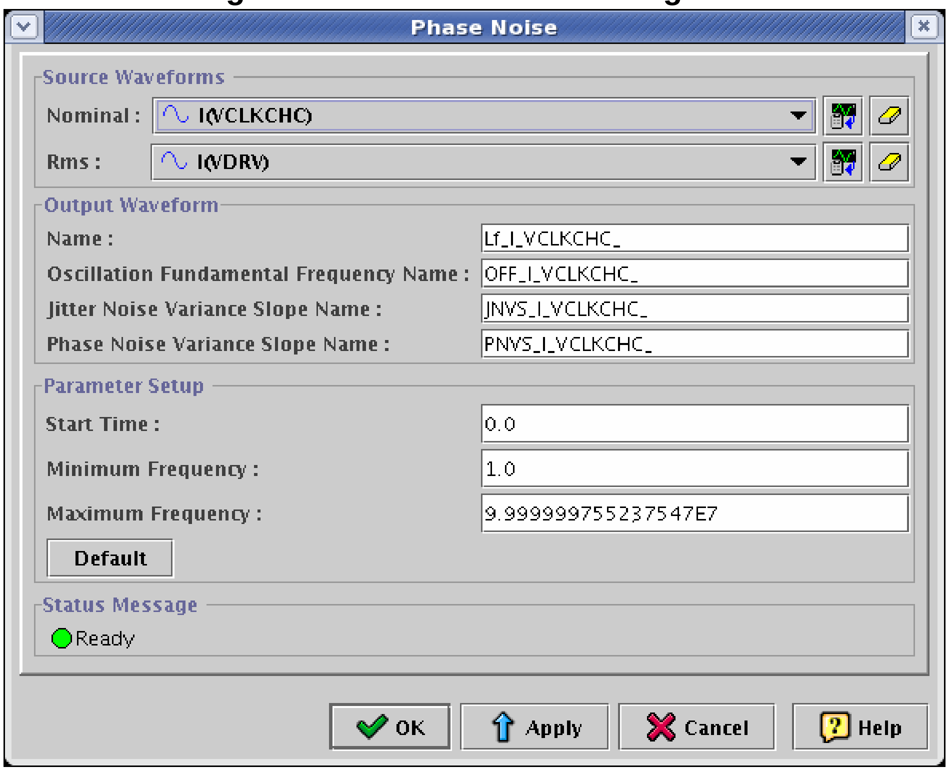
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **RF** from the dropdown list and click the **pn** button.

Use this dialog box to specify the parameters for the phase noise function used in the Waveform Calculator.

Tip

See phasenoise.

Figure 8-60. Phase Noise Dialog Box



Objects

Table 8-105. Phase Noise Dialog Box Contents

Field	Description
Source Waveforms	Specifies the Nominal and Rms source waveforms. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Output	
Waveform Name	Specifies the name of the output waveform. The result waveform has a frequency domain waveform and a dBc range.

Table 8-105. Phase Noise Dialog Box Contents (cont.)

Field	Description
Oscillation Fundamental Frequency Name	Specifies the name of the oscillation fundamental frequency output variable.
Jitter Noise Variance Slope Name	Specifies the name of the jitter noise variance slope output variable.
Phase Noise Variance Slope Name	Specifies the name of the phase noise variance slope output variable.
Parameter Setup	
Start Time	Specifies the first X value of a transient signal.
Minimum Frequency	Specifies the start of the frequency band in which the phase noise spectrum are calculated. Default is 1.0.
Maximum Frequency	Specifies the stop of the frequency band in which the phase noise spectrum is calculated. Default is *(frequency of the derivative signal)/ 2*.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Power Spectral Density Dialog Box

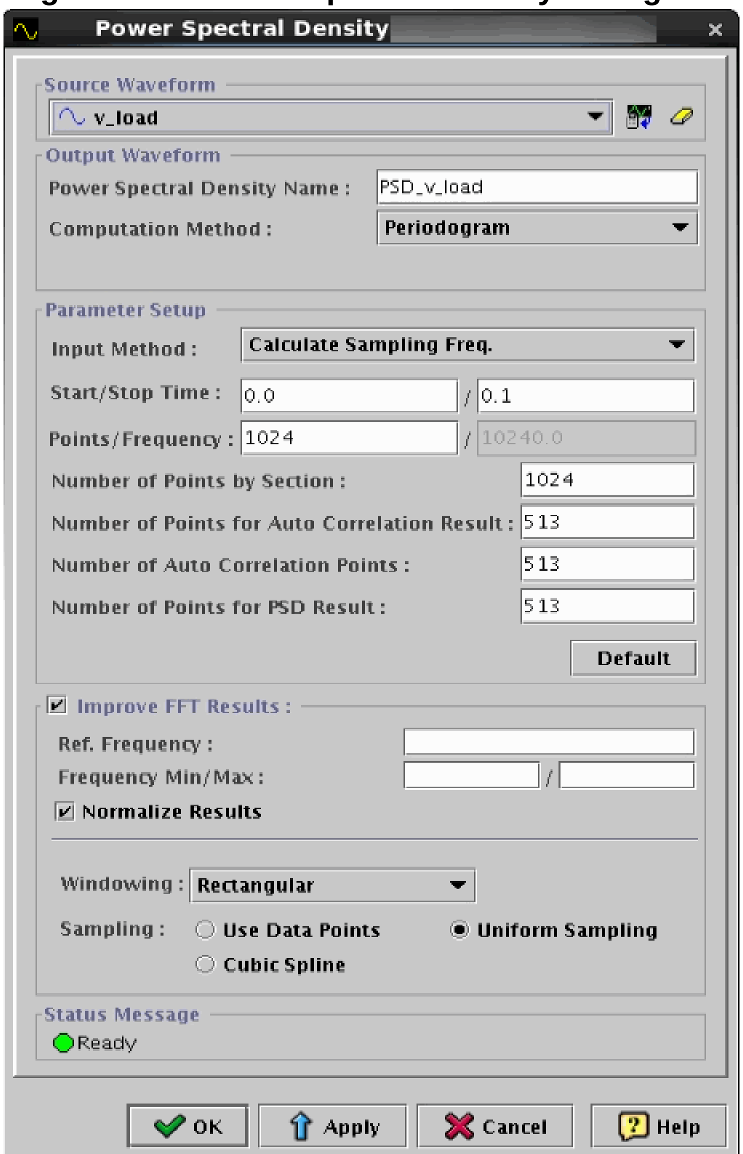
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Signal Processing** from the dropdown list and click the **psd** button.

Use this dialog box to specify the parameters for the power spectral density function used in the Waveform Calculator.

Tip

See psd.

Figure 8-61. Power Spectral Density Dialog Box



Objects

Table 8-106. Power Spectral Density Dialog Box Contents

Field	Description
Source Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon  adds the currently selected waveform. The **Clear Waveform List** icon  removes all waveforms in the list.
Output Waveform	
Power Spectral Density Name	Specifies the name for the PSD output waveform.
Computation Method	Specifies the method for calculating the PSD: • Correlogram Method • Periodogram Method Refer to "Autocorrelation Function and Power Spectral Density" on page 391 for more information about these options.
Parameter Setup	
Input Method	Specifies one of the following: • Calculate Time Start • Calculate Time Stop • Calculate Points • Calculate Sampling Freq.
Start /Stop Time	Specifies the start and stop times for the signal.
Points / Frequency	Specifies the number of sampling points and the sampling frequency. The number of sampling points for the FFT results is *(Number of Points)/2*. FFT computations are performed only on those signals having Number of Points set to a factor of 2^n ($n = 2, 3, ...$). If this is not so, then a slower DFT computation is executed.
Number of Points by Section	Specifies the number of points for the output waveform.
Number of Points for Auto Correlation Result	Specifies the number of points in the AF result (N_{auto}).
Number of Auto Correlation Points	Specifies the number of autocorrelation points used for the PSD computation.
Number of Points for PSD Result	Specifies the number of points for the PSD result.
Default	Click to set the default Auto Correlation setup parameters.

Table 8-106. Power Spectral Density Dialog Box Contents (cont.)

Field	Description
Improve FFT Results	
Ref. Frequency	Adjusts results around the y axis so that the point for the specified reference frequency is 0.0.
Frequency Min / Max	(Optional) Specifies the start and end frequencies used to display the FFT results. Defaults are “Begin” and “End”.
Normalize Results	Specifies that all results are divided by (Number of Points)/2, except for the first point, which is divided by Number of Points.
Windowing	Specifies a windowing transform to apply to the signal from: • Bartlett • Blackman • Blackman-Harris • Dolph-Chebyshev • Hamming • Hanning • Kaiser • Klein • Parzen • Rectangular • Welch Refer to “Windowing Transforms” on page 396 for more details about these options.
Sampling	Specifies the sampling method as one of the following: • Use Data Points — Select if the input data has equidistant Time Steps. The default. • Cubic Spline — Interpolated points are computed using the Cubic Spline method rather than linear interpolation. • Uniform Sampling — Select otherwise.

Usage Notes

This dialog box accesses two different but related functions: calculating the autocorrelation function (AF) of a signal waveform, and calculating the power spectral density (PSD) of a signal waveform. The AF is an average measure of its time domain properties; as such, it can be especially relevant when the signal is random.

Related Topics

Waveform Calculator

Waveform Calculator GUI

PSS Residue Dialog Box

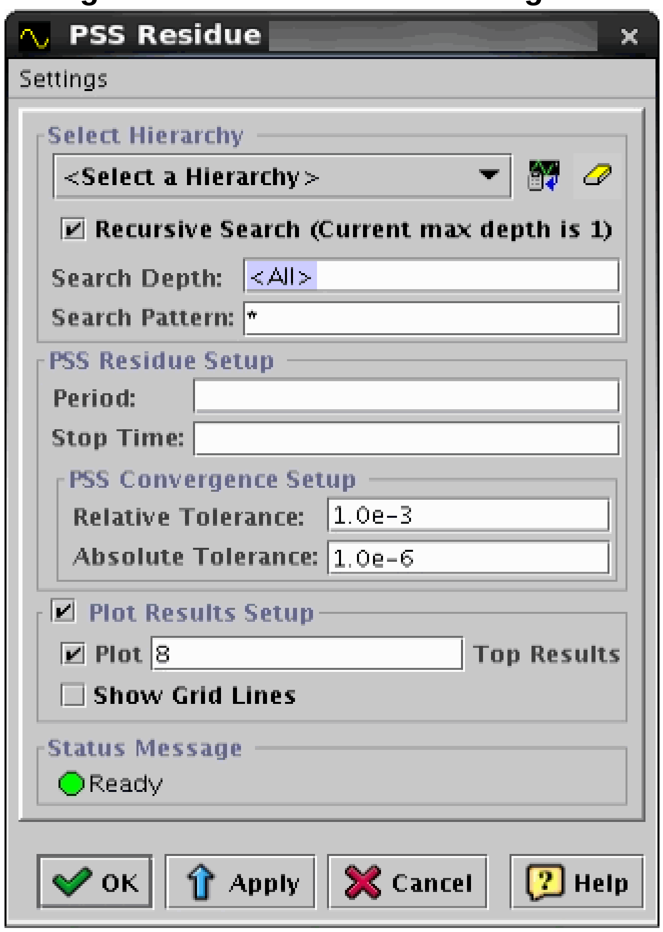
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **RF** from the dropdown list and click the **pssresidue** button.

Use this dialog box to specify the parameters for the pssresidue function used in the Waveform Calculator. This function compares every signal at “Stop Time” and “Stop Time - Period” to calculate the maximum PSS Residue and plots the top signals in sorted order of the maximum difference. A report is generated if non-converged signal are detected during the pssresidue calculation

Tip

See pssresidue.

Figure 8-62. PSS Residue Dialog Box



Objects

Table 8-107. PSS Residue Dialog Box Contents

Field	Description
Select Hierarchy	

Table 8-107. PSS Residue Dialog Box Contents (cont.)

Field	Description
Select a Hierarchy	You can specify a database, folder or waveform hierarchy as the input. Use the **Add Selected Hierarchies** icon to add the currently selected hierarchy or simply drag and drop the required item from the Waveform List into the <Select a Hierarchy> field. The **Clear Hierarchy List** icon removes the hierarchy from the list.
Recursive Search	Specifies that a recursive waveform search is performed, starting from the selected hierarchy.
Search Depth	Specifies the number of levels deep for the waveform search, starting from the specified hierarchy. Default is <All>, which corresponds to a full recursive search. Use 0 to search only in the selected hierarchy.
Search Pattern	Finds waveform names that match the specified pattern. Default is "*".
PSS Residue Setup	
Period	Specifies the fundamental period in PSS analysis. To find slow drifting signals, try specifying integer multiples of the fundamental period.
Stop Time	Specifies the stop time. Default is "End" (maximum X value).
PSS Convergence Setup	
Relative Tolerance	Specifies the relative tolerance for the PSS residue value. Used for PSS convergence detection. PssNorm = PssDiff / (abstol + reltol * max(abs(input_signal))). A node is considered converged when PSSNorm is < 1. When PssNorm is > 100, a node is considered far from converged. Default is 1e-3.
Absolute Tolerance	Specifies the absolute tolerance for the PSS residue value. Used for PSS convergence detection. PssNorm = PssDiff / (abstol + reltol * max(abs(input_signal))). A node is considered converged when PSSNorm is < 1. When PssNorm is > 100, a node is considered far from converged. Default is 1e-6.
Plot Results Setup	
Plot Top Results	Specifies the number of signals to plot, sorted in order of the maximum difference. Default is 8.
Show Grid Lines	Displays grid lines in the results window.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Signal to Noise Ratio Dialog Box

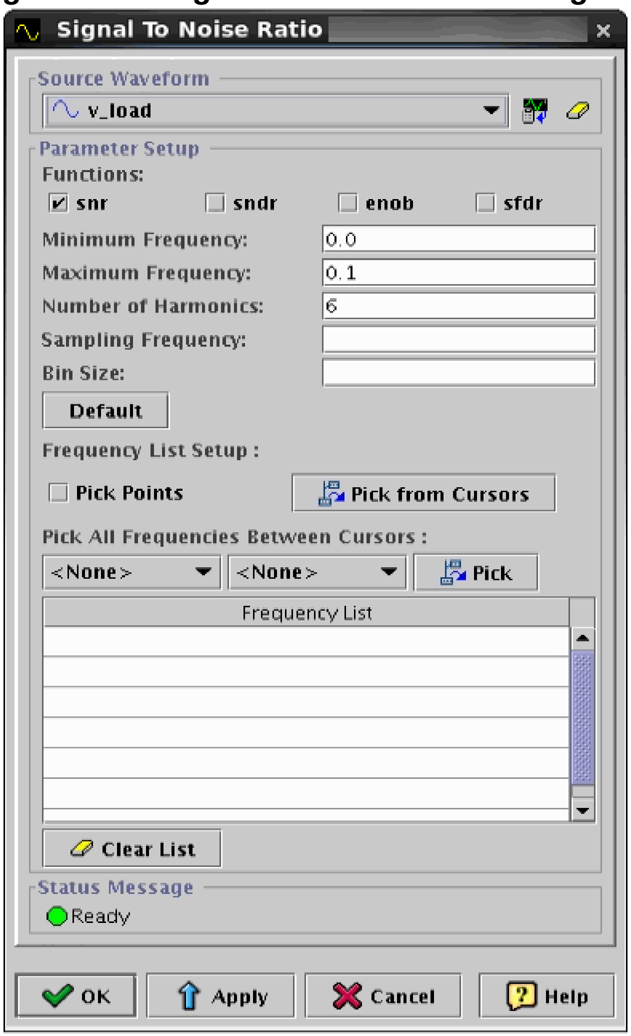
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Signal Processing** from the dropdown list and click the **snr** button.

Use this dialog box to specify the parameters for the snr, sndr, enob, and sfdr signal processing functions used in the Waveform Calculator.

Note

Refer to "Signal to Noise Function" on page 394 for details on how this function computes the signal to noise ratio of the input waveform signal by using the Gain of the FFT result.

Figure 8-63. Signal to Noise Ratio Dialog Box



Objects

Table 8-108. Signal to Noise Dialog Box Contents

Field	Description
Source Waveform	
Select Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon adds the currently selected waveform. The **Clear Waveform List** icon removes all waveforms in the list.
Parameter Setup	
Functions: snr sndr enob sfdr	Select one or more of the functions to be processed.
Minimum Frequency	Specifies the minimum frequency to be used in the calculation.
Maximum Frequency	Specifies the minimum frequency to be used in the calculation.
Number of Harmonics	Number of harmonics of the signal to be considered. Default is 6
Sampling Frequency	Sampling frequency of the source waveform. Used to manage harmonic folding and aliasing. Default is none.
Bin Size	(Optional) Specifies the bin size for the signal to noise measurement, that is, the number of points to take into account around the fundamental signal. For example: • with bin = 0, only the fundamental is taken into account • with bin = 1, the frequencies taken into account correspond to the first point to the left and right of the found fundamental and each of its associated harmonics.
Frequency List	
Pick Points	Specifies that frequencies are added to the list by clicking on the waveform.
Pick from Cursors	Adds the frequencies at the cursors on the specified waveform to the list.
Pick All Frequencies Between Cursors	These dropdown lists specify a pair of cursors to use as boundaries for picking frequencies.
Pick	Populates the frequency list with the frequencies between the two specified cursors.You can also enter the frequencies manually.

Usage Notes

Only a complex waveform or a waveform representing a Gain is accepted as a valid source waveform.

If no minimum and maximum frequencies are specified, the computation is applied over the entire waveform.

The minimum and maximum frequency boundaries do not limit your ability to pick frequencies outside of those boundaries in the Frequency List.

Related Topics

Waveform Calculator

Waveform Calculator GUI

snr

sndr

enob

sfdr

Spectrum Measurement Tool Dialog Box

To access: Select **Tools** > **Spectrum Measurement** from the main menu to open the Spectrum Measurement Tool dialog box. Alternatively, choose **Tools** > **Waveform Calculator** to open the Waveform Calculator, choose **Signal Processing** from the dropdown list and click the **spec. meas.** button.

Use this dialog box to specify the parameters for the spectrummeasurement function used in the Waveform Calculator and Spectrum Measurement Tool. Computes spectrum measurements on the input waveform, including "snr", "sndr", "sfdr", "enob", and "thd".

Tip

See "spectrummeasurement" on page 928 and "Using the Spectrum Measurement Tool" on page 388.

Figure 8-64. Spectrum Measurement Tool Dialog Box- Settings Tab

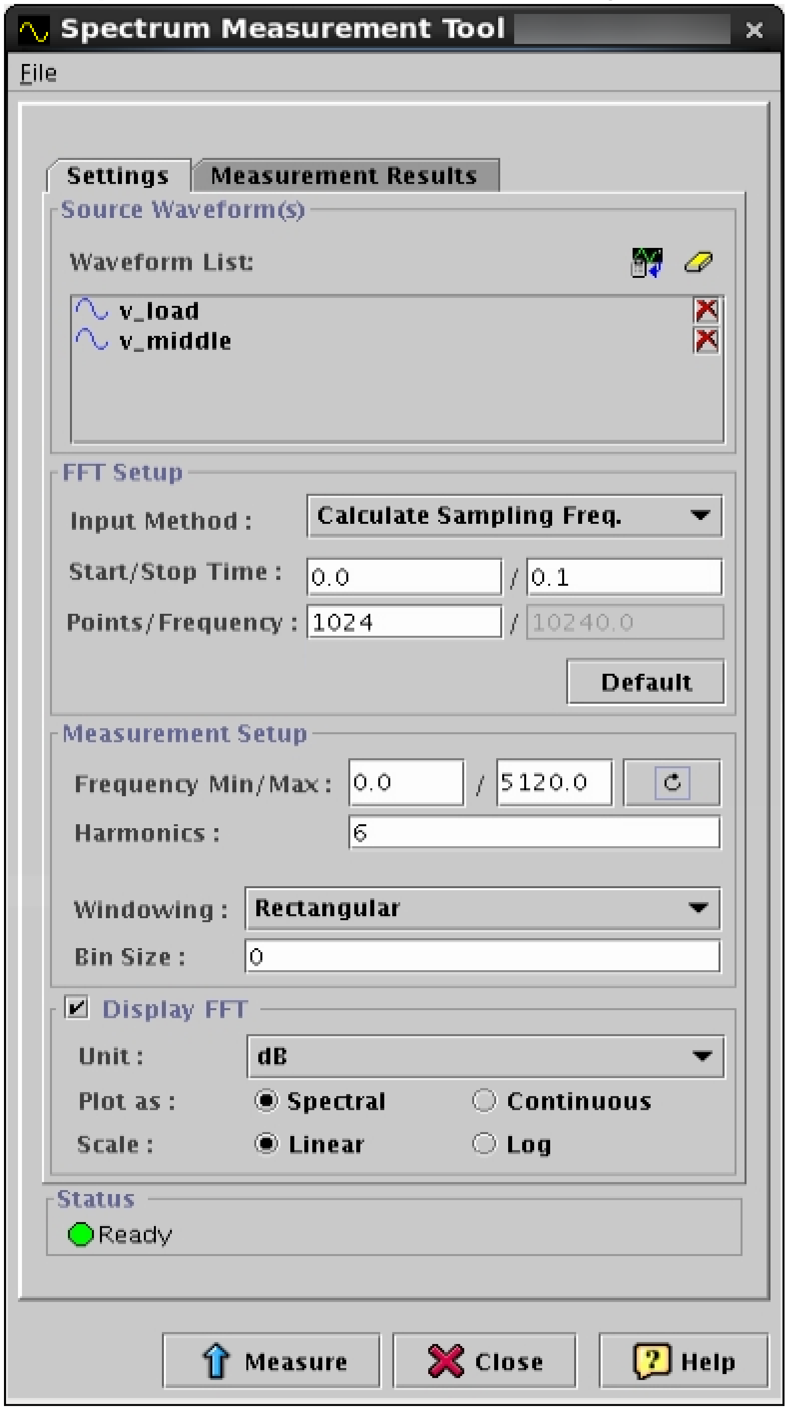
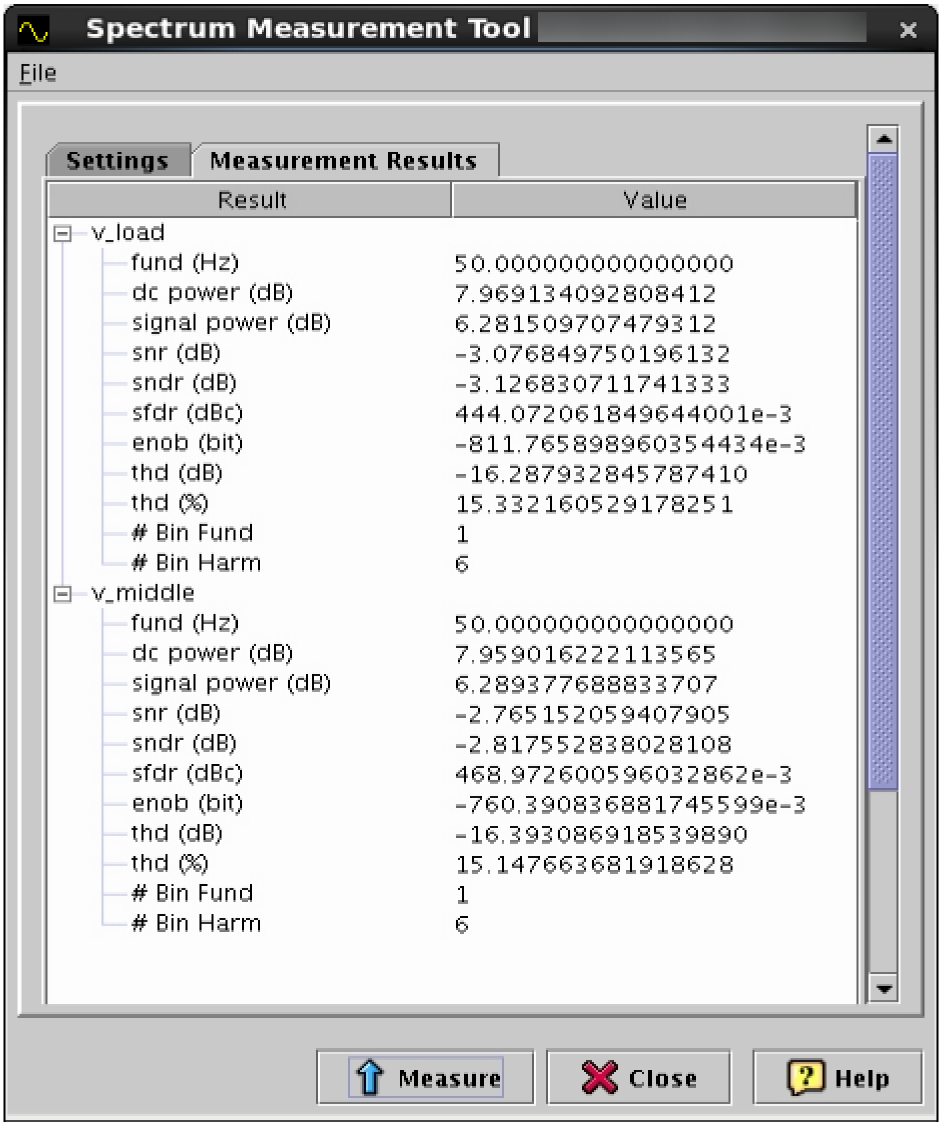


Figure 8-65. Spectrum Measurement Tool Dialog Box - Measurement Results Tab



Objects

Table 8-109. Spectrum Measurement Tool Dialog Box Contents

Field	Description
File menu	Select **File** > **Open Configuration** to load a Spectrum Measurement (or FFT) configuration. Select **File** > **Save Configuration** to save a Spectrum Measurement configuration. All parameter of the Spectrum Measurement Tool GUI are saved, except the source waveform and the output waveform names, with file extension *.ez_cfg.
Source Waveform(s)	

Table 8-109. Spectrum Measurement Tool Dialog Box Contents (cont.)

Field	Description
Waveform List	Specifies the source waveform(s). The **Add Selected Waveform** icon  adds the currently selected waveforms. The **Clear Waveform List** icon  removes all waveforms in the list. The **Delete** icon  removes the specified waveform from the list.
FFT Setup	
Input Method	Specifies one of the following: • Calculate Time Start • Calculate Time Stop • Calculate Points • Calculate Sampling Freq.
Start /Stop Time	Specifies the start and stop times for the signal.
Points / Frequency	Specifies the number of sampling points and the sampling frequency.
Default	Click to set the default FFT setup parameters.
Measurement Setup	
Frequency Min / Max	(Optional) Specifies the start and stop frequency used in the noise integration calculation. Default values are “Begin” and “End”.
	Click to update the minimum and maximum frequencies for the selected waveform(s).
Harmonics	(Optional) Specifies the number of harmonics of the signal to be considered. Default value is 6 (signal + 5 harmonics).
Windowing	Specifies a windowing transform to apply to the signal from: • Bartlett • Blackman • Blackman-Harris • Dolph-Chebyshev • Hamming • Hanning • Kaiser • Klein • Parzen • Rectangular • Welch Refer to “Windowing Transforms” on page 396 for more details about these options.

Table 8-109. Spectrum Measurement Tool Dialog Box Contents (cont.)

Field	Description
Bin Size	Specifies the number of points to take into account around the fundamental signal. For example: • with Bin Size = 0, only the fundamental is taken into account • with Bin Size = 1, the frequencies taken into account correspond to the first point to the left and right for the found fundamental and each of its associated harmonics. The default value depends on the Windowing transform selected.
Display FFT	
Unit	Specifies one of the following: • dB — result waveform displayed in dB. • Magnitude — result waveform for the absolute magnitude of the input waveform displayed. • Phase — result waveform for the phase displayed in degrees. • Real — displays the real part of the result waveform. • Imaginary — displays the imaginary part of the result waveform.
Plot as	Specifies the drawing mode as one of the following: • Spectral — Specifies the “spectral” drawing mode. The default. • Continuous — Specifies the “continuous” drawing mode.
Scale	Specifies either a Linear or Log scale.
Measurement Results	
Result	Lists the waveform name(s) and expands to show the measurements: • fund (fundamental frequency) • dc power (value of the DC amplitude) • signal power (value of the fundamental amplitude) • snr (signal to noise ratio) • sndr (signal to noise and distortion ratio) • sfdr (spurious free dynamic range) • enob (effective number of bits • thd (dB) (total harmonic distortion in dB) • thd (%) (total harmonic distortion in %) • # Bin Fund (number of bins used in fundamental value calculation) • # Bin Harm (number of bins used in harmonic calculation) **Tip:** Right-click waveform names or measurements to see more options.

Table 8-109. Spectrum Measurement Tool Dialog Box Contents (cont.)

Field	Description
Value	Lists the value of each measurement. **Tip:** You can change the number of significant figures displayed. Choose **Edit** > **Options** and select **Data Format** on the EZwave Display Preferences dialog box. Choose the required Double Format Precision from the dropdown list.

Related Topics

Using the Spectrum Measurement Tool

Waveform Calculator

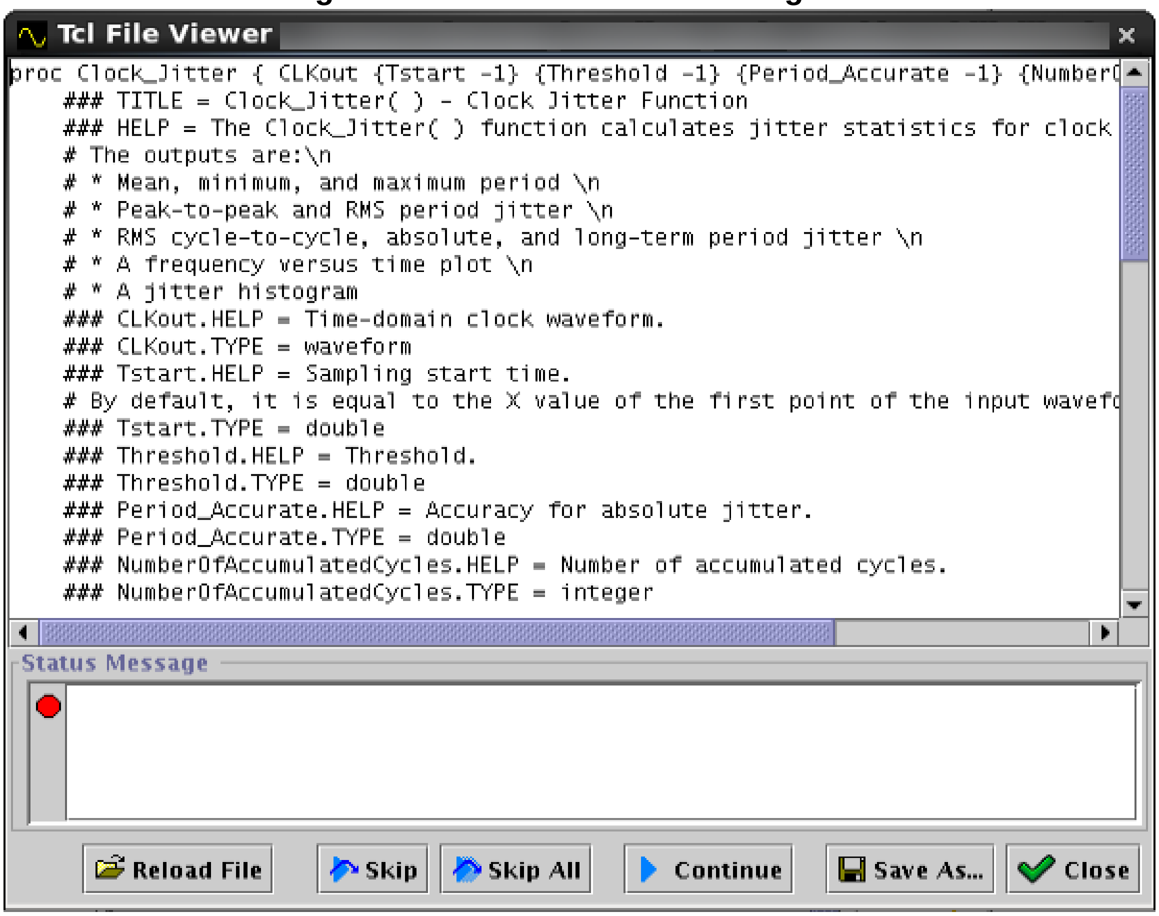
Waveform Calculator GUI

Tcl File Viewer Dialog Box

To access: Choose **Tools** > **Waveform Calculator** from the main menu. Right-click the required function in the Chooser panel User-Defined Functions list and select **Edit**.

You can view, edit, run, save and reload User-Defined Functions in the Waveform Calculator using the built-in Tcl File Viewer.

Figure 8-66. Tcl File Viewer Dialog Box



Objects

Table 8-110. Tcl File Viewer Dialog Box Contents

Field	Description
Text window	Displays the Tcl function. You can edit the text in this window.
Status Message	Displays status messages.
Reload File	Reloads the current user-defined function.
Skip	Skips the last error.
Skip All	Skips all errors.
Continue	Continues running the user-defined function, from the last error.

Table 8-110. Tcl File Viewer Dialog Box Contents (cont.)

Field	Description
Save As	Saves the edited function to a specified file (*.tcl*). **Note:** If you change the name of the function, it is displayed in the User-Defined Functions list for the current session. When a new session is opened, you can reload the function, see "Using and Editing User-Defined Functions in the Waveform Calculator" on page 319.

Usage Notes

You can drag waveform names from the Waveform List into the TCL File Viewer the same way as you can when using the Waveform Calculator. The waveform name and path appears in the text, as shown in the following example:

```
wf("<tutorial/Time-Domain_Results>v_load")
```

You can add documentation information that then appears automatically in the Waveform Calculator Help pane, as shown highlighted in the code example:

```
proc procName { {Param0 "wf(\"<tutorial/Time-Domain_Results>v_middle\")"}
} {
    ### TITLE = procName - ProcTitleHelp
    ### HELP = ProcHelpDesciprion
    ### RETURN.TYPE = object
    ### RETURN.HELP = ProcReturnDescription
    ### Param0.HELP = Param0Help
    ### Param0.TYPE = waveform
    wfc "
       s0 = 1+1
       s1 = 2
       s2 = 3
       wf0 = $Param0
    "
    return [ wfc "wf0" ]
}
```

If the user-defined function has some predefined values for a particular parameter, these can be specified and subsequently selected from a dropdown list on the Waveform Calculator dialog box. Add a line for each of the parameters listing the possible values, as follows:

```
$PARAMNAME.POSSIBLE_VALUES = Value1,Value2,...,ValueN
```

Related Topics

Waveform Calculator

Waveform Calculator GUI

Top Noise Dialog Box

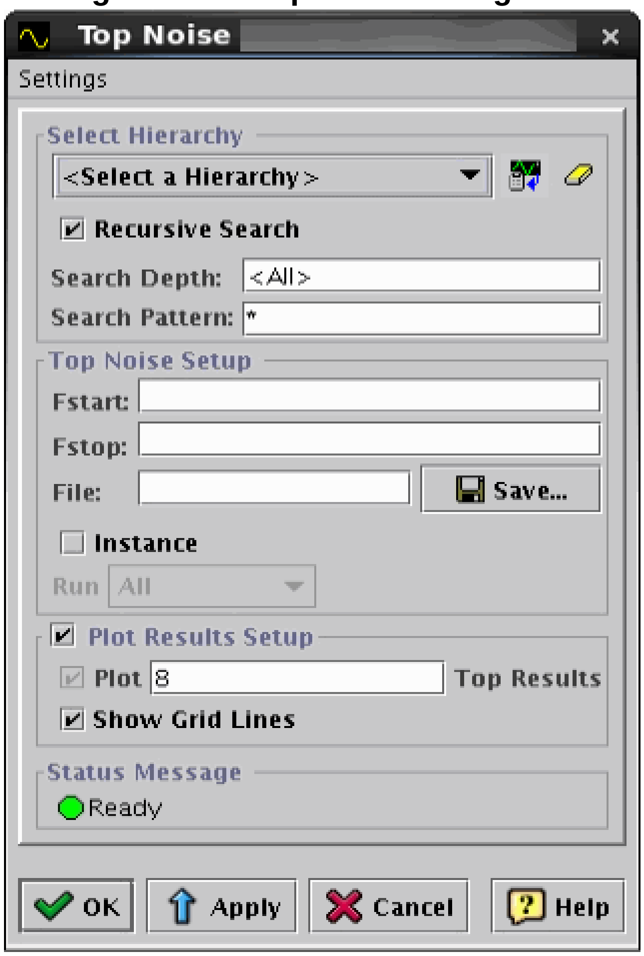
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **RF** from the dropdown list and click the **topnoise** button.

Use this dialog box to specify the parameters for the topnoise function used in the Waveform Calculator. Plots the noise spectrum in order of spot noise if you only specify "Fstart", or plots integrated noise if you specify both "Fstart" and "Fstop".

Tip

See also topnoise and dataset topnoise.

Figure 8-67. Top Noise Dialog Box



Objects

Table 8-111. Top Noise Dialog Box Contents

Field	Description
Select Hierarchy	

Table 8-111. Top Noise Dialog Box Contents (cont.)

Field	Description
Select a Hierarchy	You can specify a database, folder or waveform hierarchy as the input. Use the **Add Selected Hierarchies** icon to add the currently selected hierarchy or simply drag and drop the required item from the Waveform List into the <Select a Hierarchy> field. The **Clear Hierarchy List** icon removes the hierarchy from the list.
Recursive Search	Specifies that a recursive waveform search is performed, starting from the selected hierarchy.
Search Depth	Specifies the number of levels deep for the waveform search, starting from the specified hierarchy. Default is <All>, which corresponds to a full recursive search. Use 0 to search only in the selected hierarchy.
Search Pattern	Finds waveform names that match the specified pattern. Default is "*".
Top Noise Setup	
Fstart	Specifies the lower frequency limit for integrated noise or the single frequency value at which to report spot noise.
Fstop	Specifies the upper frequency limit. If specified, integrated noise is calculated. If not specified, spot noise at "Fstart" is calculated.
File	Specifies a file in which to save a sorted report of all noise contributions.
Save	Click **Save** to browse folders and specify the location for the report file.
Instance	Specifies that report data is generated not only at MOS instance level, but recursively at subckt instance level.
Run	Specifies which runs to analyze for compound waveforms. Choose **All** (default) or specify a run number.
Plot Results Setup	
Plot Top Results	When Plot is enabled, specifies the number of signals to plot, sorted in order of the spot noise. Default is 8.
Show Grid Lines	Displays grid lines in the results window. Unchecking this box may increase plot performance.

Related Topics

Waveform Calculator

Waveform Calculator GUI

topnoise

dataset topnoise

Windowing Transform Dialog Box

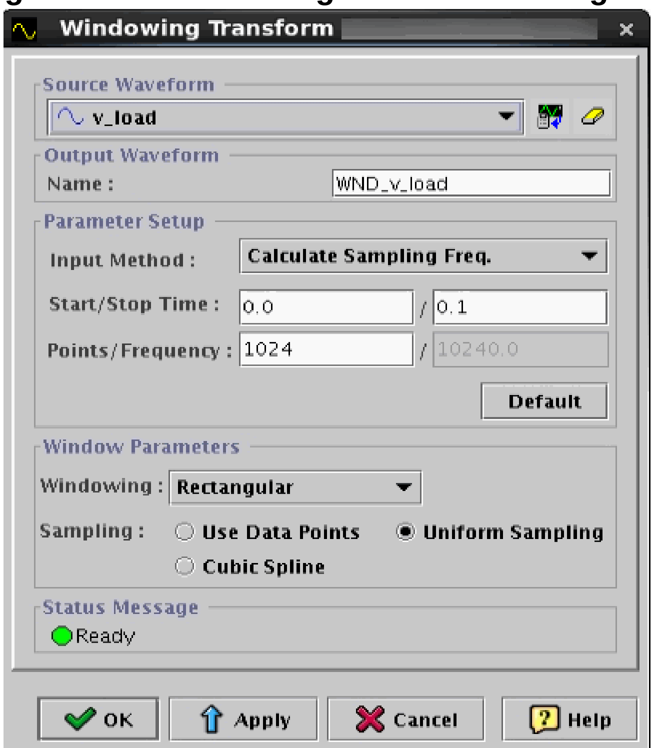
To access: Choose **Tools** > **Waveform Calculator** from the main menu, choose **Signal Processing** from the dropdown list and click the **wnd** button.

Use this dialog box to specify the parameters for the windowing function used in the Waveform Calculator.

Tip

See windowing.

Figure 8-68. Windowing Transform Dialog Box



Objects

Table 8-112. Windowing Transform Dialog Box Contents

Field	Description
Select Waveform	Specifies the source waveforms. The **Add Selected Waveform** icon adds the currently selected waveform. The**Clear Waveform List** icon removes all waveforms in the list.
Output Waveform	
Name	Specifies the name of the output waveform
Parameter Setup	

Table 8-112. Windowing Transform Dialog Box Contents (cont.)

Field	Description
Input Method	Specifies one of the following: • Calculate Time Start • Calculate Time Stop • Calculate Points • Calculate Sampling Freq.
Start /Stop Time	Specifies the start and stop times for the signal.
Points / Frequency	Specifies the number of sampling points and the sampling frequency.
Default	Click to set the default Windowing Transform setup parameters.
Window Parameters	
Windowing	Specifies a windowing transform to apply to the signal from: • Bartlett • Blackman • Blackman-Harris • Dolph-Chebyshev • Hamming • Hanning • Kaiser • Klein • Parzen • Rectangular • Welch Refer to "Windowing Transforms" on page 396 for more details about these options.
Sampling	Specifies the sampling method as one of the following: • Use Data Points — Select if the input data has equidistant Time Steps. The default. • Cubic Spline — Interpolated points are computed using the Cubic Spline method rather than linear interpolation. • Uniform Sampling — Select otherwise.

Related Topics

Waveform Calculator

Waveform Calculator GUI

Appendix A
Waveform Calculator Functions

Detailed descriptions of all the built-in functions available in the Waveform Calculator are listed alphabetically.

Syntax Conventions **660**

Function Descriptions **662**

abs 666
absolutejitter 667
absolutejitterbyintegration 669
acos 670
acosh 671
acot 672
acoth 673
add. 674
allandeviation 676
analysisattributes. 677
analysisattributevalue 678
asin 679
asinh 680
assign 681
atan 682
atan2 683
atanh 684
atod 685
autocor 687
avg. 690
bandpass 691
bandwidth 692
baseline 693
calcvartype 694
cdf 695
ceil. 696
chirp 697
complex. 700
compress 701
compresscompound 702
concat 703
conjugate. 704
constellationdiagram. 705
continuous. 706

convolution 707
cos 709
cosh 710
cot 711
coth 712
cphase 713
cphytrigger 714
crosscorrelation 716
crossing 717
cycle2cyclejitter 718
cycletocyclejitter 720
cycletocyclejitterbyintegration 721
datatocomplex 722
datatodig 723
datatowf 725
dchysteresis 726
db 727
db10 728
deg 729
delay 730
derive 732
drv 733
dtoa 734
dtoaonbit 735
dutycycle 737
edgephasenoise 739
enob 740
evmber 742
exp 743
exportcsv 744
exportvcd 745
eye 747
eyeamplitude 749
eyecphy 750
eyecrossingamplitude 752
eyedelay 753
eyediagram 754
eyefalltime 756
eyeheight 757
eyeheightatx 758
eyejitter 759
eyemeasures 760
eyerisetime 762
eyesetmask 763
eyesnr 765

eyewidth 766
eyewidthaty 767
eyewithtrigger 768
falltime 770
fft 772
filterdupreal 775
filterempty 776
first 777
firstdiff 778
floor 780
fmod 781
frequency 782
frequencyjitter 784
frexp 786
gainmargin 787
gaussiandistribution 788
gendecade 789
genlinear 790
genoctave 791
getelementat 792
getrunindices 793
getrunparameters 794
getrunparametervalue 795
gettype 796
gmargin 797
gptocomplex 798
groupdelay 799
halfperiodjitter 800
harm 802
harmonicdistortion 803
harmonics 804
harmonicsmeter 805
hdist 806
histogram 807
hypot 808
idb 809
idb10 810
ifft 811
iipx 813
imag 814
integ 815
integnoise 816
integral 817
intersect 818
ipn 820

jc 823
jcc 824
jee 825
join 826
larger 828
ldexp 829
last 830
length 831
lesser 832
ln 833
localmax 834
localmin 836
log 838
longtermjitter 839
mag 841
max 842
maxdiff 843
maxspectrumdiff 845
mean 847
meanminus3std 848
meanplus3std 849
min 850
modf 851
mptocomplex 852
nand 853
noisetosignaldbc 854
noisetrantophasenoise 855
nor 856
oipx 857
overshoot 858
peaktopeak 859
period 860
periodjitter 862
periodjitterbyintegration 864
phase 865
phasemargin 866
phasenoise 867
phasenoiseoutdbc 868
phnoisebydlm 869
phnoisebymixer 871
phmargin 872
pivot 873
plotjitterconfidenceinterval 874
pow10 875
psd 876

pssresidue 879
pulsewidth 880
rad 882
real 883
reglin 884
relation 885
removepts 886
risetime 887
ritocomplex 889
rms 890
rms_ac 891
rms_accurate 892
rms_noise 893
rms_tran 894
rol 895
ror 896
round 897
sample 898
samplelog 899
samplepsd 900
setAngleUnits 904
setNotation 905
setTemperatureUnits 906
settlingtime 907
sfdr 908
shift 910
shiftedmaxdiff 911
sin 913
sinad 914
sinh 916
size 917
sla 918
slewrate 919
slope 921
slopeintersect 922
sndr 923
snr 925
sorty 927
spectrummeasurement 928
sphibyjitter 931
sphifilter 933
sqr 934
sqrt 935
sra 936
stddev 937

sum 938
tan 939
tanh 940
thd 941
tiejitter 943
timeabsolutejitter 945
timelongtermjitter 947
timeperiodjitter 949
timestep 951
todchysteresis 952
topline 953
topnoise 954
trunc 955
undershoot 956
var 957
wavevswave 958
wf 959
wfattributes 961
wfname 962
wftoascii 963
wftodata 964
windavg 965
window 966
windowing 967
xcompress 969
xdown 970
xnor 971
xofmax 972
xofmin 973
xup 974
xval 975
xwave 976
xytowf 977
yval 978

Syntax Conventions

This topic describes the syntax conventions used in the Usage sections of the detailed function descriptions for Required, Required Replace and Optional Replace fields.

- **Required** — Function names in **bold** and must be typed exactly as shown.
- ***Required Replace*** — Required argument place holders are in ***bold italics***. You must replace these arguments with a value or string. For example,

  abs(*wf*)

indicates that you must replace the required argument ***wf*** with a value.

- [*Optional Replace*] — Optional argument place holders are surrounded by square brackets [ ] and in *italics*. When using these optional arguments, you must replace them with a value or a string. For example,

  acos(***wf***[, *x_start*, *x_end*])

  indicates that if you want, you can supply start and end values for x by replacing *x_start* and *x_end* with values.

Note

The square brackets [ ] around optional arguments may be omitted in some Usage lines of the function descriptions to improve clarity. In these cases, multiple Usage lines are listed to show the different combinations of optional arguments.

Function Descriptions

All of the functions are available in the Functions list on the Waveform Calculator, and some of them also appear as buttons in the calculator when the appropriate category is selected from the dropdown list in the calculator.

Refer to Waveform Calculator for details.

Tip

Some example Tcl scripts are provided as User-Defined Functions for the Waveform Calculator. Refer to "Waveform Calculator Example Tcl Scripts" on page 1224.

The following tables list the functions by category.

Table A-1. Complex Functions

complex	db10	imag	real
conjugate	gptocomplex	mag	ritocomplex
cphase	idb	mptocomplex	
db	idb10	phase	

Table A-2. Jitter Functions

cycle2cyclejitter	halfperiodjitter	tiejitter	timelongtermjitter
frequencyjitter	plotjitterconfidenceinterval	timeabsolutejitter	timeperiodjitter

Table A-3. Logic Functions

nand	rol	sla	xnor
nor	ror	sra	

Table A-4. Mathematical Functions

abs	fmod	ln	sqrt
ceil	frexp	log	xofmax
derive	hypot	modf	xofmin
drv	integ	pow10	xwave
exp	integral	relation	
floor	ldexp	sqr	

Table A-5. Measurement Functions

bandpass	frequency	overshoot	slewrate
bandwidth	gainmargin	peaktopeak	slope
baseline	localmax	period	slopeintersect
crossing	localmin	phasemargin	stddev
delay	mean	pulsewidth	topline
dutycycle	meanminus3std	risetime	undershoot
falltime	meanplus3std	settlingtime	

Table A-6. Miscellaneous Functions

add	exportvcd	harm	var
analysisattributes	filterdupreal	intersect	wavevswave
analysisattributevalue	filterempty	join	wfattributes
assign	first	last	wfname
atod	firstdiff	length	wftoascii
calcvartype	gendecade	maxdiff	wftodata
concat	genlinear	maxspectrumdiff	window
continuous	genoctave	phmargin	xdown
datatocomplex	getelementat	pivot	xup
datatodig	getrunindices	reglin	xval
datatowf	getrunparameters	removepts	xytowf
dchysteresis	getrunparametervalue	shift	yval
dtoa	gettype	shiftedmaxdiff	
dtoaonbit	gmargin	timestep	
exportcsv	groupdelay	todchysteresis	

Table A-7. Phase Noise Functions

absolutejitter	cycletocyclejitterbyintegration	phasenoiseoutdbc	sphifilter
absolutejitterbyintegration	longtermjitter	phnoisebydlm	
allandeviation	periodjitter	phnoisebymixer	

Table A-7. Phase Noise Functions (cont.)

cycletocyclejitter	periodjitterbyintegration	sphibyjitter	

Table A-8. RF Functions

compress	harmonicsmeter	jcc	phasenoise
compresscompound	iipx	jee	pssresidue
constellationdiagram	integnoise	noisetosignaldbc	topnoise
edgephasenoise	ipn	noisetrantophasenoise	xcompress
evmber	jc	oipx	

Table A-9. Signal Processing Functions

autocor	fft	rad	sndr
chirp	harmonicdistortion	sample	snr
convolution	harmonics	samplelog	spectrummeasurement
crosscorrelation	hdist	samplepsd	thd
deg	ifft	sfdr	windowing
enob	psd	sinad	

Table A-10. Statistical Functions

avg	eyefalltime	eyewidth	rms_ac
cphytrigger	eyeheight	eyewidthaty	rms
eye	eyeheightatx	eyewithtrigger	rms_accurate
eyeamplitude	eyejitter	histogram	rms_noise
eyecphy	eyemeasures	larger	rms_tran
eyecrossingamplitude	eyerisetime	lesser	size
eyedelay	eyesetmask	max	sum
eyediagram	eyesnr	min	windavg

Table A-11. Trigonometric Functions

acos	asinh	cosh	tan
acosh	atan	cot	tanh
acot	atan2	coth	

Table A-11. Trigonometric Functions (cont.)

acoth	atanh	sin	
asin	cos	sinh	

Table A-12. Special Functions

setAngleUnits	setNotation	setTemperatureUnits	wf

Note

Special functions do not appear in the Waveform Calculator Functions list but may be used for scripting.

abs

Waveform Calculator Mathematical function.

Returns the absolute value of the waveform.

Usage

abs(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

Return Values

Waveform. The absolute value of the input waveform.

absolutejitter

Waveform Calculator Phase Noise function.

Returns the absolute jitter value. Absolute jitter corresponds to the jitter (average rms value) with respect to an ideal (jitter-free) reference source. Applied to SST Noise Analysis results.

Usage

absolutejitter(***wf***, ***f0***[, *f_start*, *f_stop*, *sampling_nb_points*])

Arguments

- ***wf***

  (Required) Specifies the name on the input waveform on which the absolute jitter is calculated.

- ***f0***

  (Required) Specifies the fundamental (reference) frequency of SST noise analysis. Default value is 'Automatic'.

  > **Note**
  >
  > If ***f0*** is stored in the database by the Eldo RF simulator, this argument becomes optional.

- *f_start*

  (Optional) Specifies the x value at the beginning of the interval for the absolute jitter calculation. Default value is 'Begin'.

- *f_stop*

  (Optional) Specifies the x value at the end of the interval for the absolute jitter calculation. Default value is 'End'.

- *sampling_nb_points*

  (Optional) Defines whether sampling is applied to the source waveform and specifies the number of sampling points. Default value is 'Automatic', for which 100 sampling points are considered.

Return Values

Double. The absolute jitter value of the input waveform.

Description

In forced circuits, jitter is considered as a random stationary process and its variance can be obtained from the phase noise spectrum. Absolute or synchronous jitter corresponds to the jitter (average rms value) with respect to an ideal (jitter-free) reference source.

$$\sigma_a^{\;2} = \frac{1}{2\pi f_0^{\;2}} \int_0^{+\infty} S_\phi(f).df$$

σ_a corresponds to the absolute jitter value.

Note

The calculation is designed for input waveforms SPHI, DB(SPHI), SPHI_SSB, DB(SPHI_SSB), Lf and DB(PHNOISE), but not restricted to this list.

Examples

```
absolutejitter(wf1)
# This will calculate the absolute jitter for waveform wf1 with an
# automatically detected fundamental frequency that has been stored in
# the database by the Eldo RF simulator. The calculation will use
# the entire waveform. There will be 100 sampling points - the default.

absolutejitter(wf1, 4e6, 3e6, 5e6, 50)
# This will calculate the absolute jitter for waveform wf1
# with fundamental frequency 4.000.000 Hz. The calculation will use
# the waveform from 3.000.000 Hz to 5.000.000 Hz. There will be 50
# sampling points.

absolutejitter(wf1, f0=6e6, sampling_nb_points=200)
# This will calculate the absolute jitter for waveform wf1 with
# fundamental frequency 6.000.000 Hz and 200 sampling points. The entire
# waveform will be used for the calculation.
```

Related Topics

Jitter Measurement Types

absolutejitterbyintegration

Waveform Calculator Phase Noise function.

Calculates the rms absolute jitter by Sphi integration.

Usage

absolutejitterbyintegration(***wf***, ***fund***[, *fstart*, *fstop*])

Arguments

- ***wf***

  (Required) Specifies the two-sided Sphi density waveform (in dB rad^2/Hz vs frequency in Hz) as computed using, for example, sphibyjitter. Only the positive side of Sphi is used.

- ***fund***

  (Required) Specifies the fundamental frequency.

- *fstart*

  (Optional) Specifies the lower frequency limit in integration. Default is ‘Begin’.

- *fstop*

  (Optional) Specifies the upper frequency limit in integration. Default is ‘End’.

Return Values

Double. The absolute jitter by Sphi integration of the input waveform.

acos

Waveform Calculator Trigonometric function.

Computes the principal value of the arccosine.

Usage

acos(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The arccosine of the input waveform.

Description

Computes the principal value of the arccosine of ***wf***. The input value should be in the range of [-1, 1].

acosh

Waveform Calculator Trigonometric function.

Computes the hyperbolic arccosine of the waveform.

Usage

acosh(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The hyperbolic arccosine of the input waveform.

acot

Waveform Calculator Trigonometric function.

Computes and returns the arccotangent of the waveform.

Usage

acot(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The arccotangent of the input waveform.

acoth

Waveform Calculator Trigonometric function.

Computes and returns the hyperbolic arccotangent of the waveform.

Usage

acoth(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The hyperbolic arccotangent of the input waveform.

add

Waveform Calculator Miscellaneous function.

Adds two overlapping waveforms, retaining non-overlapping parts.

Usage

add(*wf1*, *wf2*)

add(*wf1*[, *x_start1*, *x_end1*], ***wf2***[, *x_start2*, *x_end2*])

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- ***wf2***

  (Required) Specifies the second input waveform name.

- *x_start1*

  (Optional) Specifies the x value at the beginning of an interval on wf1.

- *x_end1*

  (Optional) Specifies the x value at the end of an interval on wf1.

- *x_start2*

  (Optional) Specifies the x value at the beginning of an interval on wf2.

- *x_end2*

  (Optional) Specifies the x value at the end of an interval on wf2.

Return Values

Waveform. The summation of two waveforms.

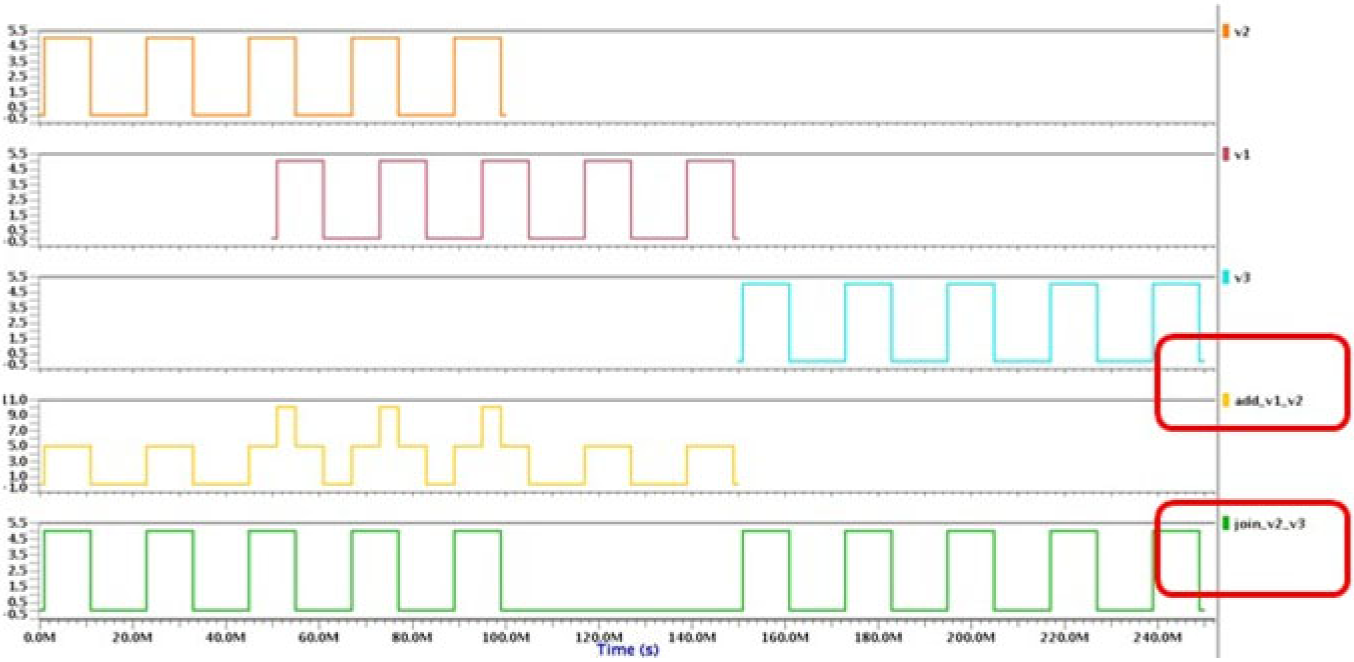
Description

Data points in non-overlapping regions are unchanged in value. Input waveforms must overlap.

Examples

The following figure shows example results when using the add and join commands:

Figure A-1. Example of add() and join() Waveform Calculator Functions



Related Topics

join

allandeviation

Waveform Calculator Phase Noise function.

Returns an Allan Deviation vs. Time waveform from SST Noise Analysis results.

Usage

allandeviation(***wf***, ***f0***[, *f_start*, *f_stop*, *sampling_nb_points*, *compute_variance*])

Arguments

- ***wf***

  (Required) Specifies the name on the input waveform on which the Allan Deviation is calculated.

- ***f0***

  (Required) Specifies the fundamental frequency of SST noise analysis. Default value is 'Automatic'.

  > **Note**
  > If ***f0*** is stored in the database by the Eldo RF simulator, this argument becomes optional.

- *f_start*

  (Optional) Specifies the x value at the beginning of the interval for the Allan Deviation calculation. Default value is 'Begin'.

- *f_stop*

  (Optional) Specifies the x value at the end of the interval for the Allan Deviation calculation. Default value is 'End'.

- *sampling_nb_points*

  (Optional) Defines whether sampling is applied to the source waveform and specifies the number of sampling points. Default value is 'Automatic', for which 100 sampling points are considered.

- *compute_variance*

  (Optional) When set to 'true', calculates Allan Variance rather than Allan Deviation. Default value is 'false', which calculates Allan Deviation.

Return Values

Waveform. The Allan Deviation vs. Time for the input waveform.

Description

Can be applied to SST noise analysis results.

analysisattributes

Waveform Calculator Miscellaneous function.

Returns analysis attribute names and values from the PSF header of the specified waveform.

Usage

analysisattributes(***wf***)

Arguments

- ***wf***

  (Required) Specifies the source waveform.

Return Values

Dictionary. All of the analysis attribute names and values.

Related Topics

analysisattributevalue

analysisattributevalue

Waveform Calculator Miscellaneous function.

Returns the value of the specified analysis attribute from the PSF header of the specified waveform.

Usage

analysisattributevalue(***wf***, ***name***)

Arguments

- ***wf***

  (Required) Specifies the source waveform.

- ***name***

  (Required) Specifies the analysis attribute.

Return Values

Double or String. Value of the specified analysis attribute.

Related Topics

analysisattributes

asin

Waveform Calculator Trigonometric function.

Computes the principal value of the arcsine of the waveform.

Usage

asin(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The arcsine of the input waveform.

Description

Computes the principal value of the arcsine of ***wf***. The input value should be in the range of [-1, 1].

asinh

Waveform Calculator Trigonometric function.

Computes the hyperbolic arcsine of the waveform.

Usage

asinh(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The hyperbolic arcsine of the input waveform.

assign

Waveform Calculator Miscellaneous function.

Uses "best efforts" to assign a specified name to an expression, without overwriting existing expressions. This enables you to overlay previously evaluated plot results for comparison.

Usage

assign(*wf*, *name*)

Arguments

- ***wf***

  (Required) Specifies the expression waveform to name.

- ***name***

  (Required) Specifies the preferred name for the expression.

Return Values

String. Final name assigned to the expression.

atan

Waveform Calculator Trigonometric function.

Computes the principal value of the arctangent of the waveform.

Usage

atan(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The arctangent of the input waveform.

atan2

Waveform Calculator Trigonometric function.

Computes and returns the principal value of the arctangent of ***y***/***x***.

Usage

atan2(*y*, *x*)

Arguments

- ***y***

  (Required) Specifies the numerator of the input.

- ***x***

  (Required) Specifies the denominator of the input.

Return Values

Double. The arctangent of the input numbers.

Description

Computes the principal value of the arctangent of ***y***/***x***, using the signs of both arguments to determine the quadrant of the return value.

Upon successful completion, returns the arctangent of ***y***/***x*** in the range [-180, 180] degrees. If both arguments are 0.0, 0.0 is returned.

Note

This function only accepts scalar numbers as input parameters.

atanh

Waveform Calculator Trigonometric function.

Computes the hyperbolic arctangent of the waveform.

Usage

atanh(*wf*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The hyperbolic arctangent of the input waveform.

atod

Waveform Calculator Miscellaneous function.

Transforms an analog waveform (or analog bus) to a digital waveform.

Usage

atod(***wf***, ***threshold1***)

atod(***wf***, ***threshold1***, ***threshold2***)

atod(***wf***, ***threshold1***, ***threshold2***, ***dtx***)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***threshold1***

  (Required) Specifies the threshold or low threshold.

- ***threshold2***

  (Required) Specifies the high threshold.

- ***dtx***

  (Required) Specifies the duration for the signal to become X after an incomplete transition, for analog to digital conversion with two thresholds.

Return Values

Waveform. The converted waveform.

Description

If only ***threshold1*** is specified, the function generates a binary (bit) waveform, with values 0/1.

If ***threshold1*** and ***threshold2*** are specified, the function generates a digital (std_logic) waveform, with values 0/1/X.

Examples

The following examples use the EZwave *tutorial.wdb* data.

A single threshold for analog to digital conversion is set to 2.5V:

```
atod(wf("<tutorial/Time-Domain_Results>v_load"), 2.5V)
```

The thresholds for analog to digital conversion are set to 1V and 4V:

```
atod(wf("<tutorial/Time-Domain_Results>v_load"), 1, 4)
```

The thresholds for analog to digital conversion are set to 1V and 4V and dtx is set to 0.1ms:

```
atod(wf("<tutorial/Time-Domain_Results>v_load"), 1, 4, 0.1m)
```

autocor

Waveform Calculator Signal Processing function.

Computes the auto-correlation of the input waveform.

Usage

autocor(***wf***[, *t_start*, *t_stop*, *fs*, *points*, *sampling*, *padding*, *normalized*, *windowType*, *alpha*, *nsect*, *nauto*, *ncorr*, *computationMethod, f_ref, f_min, f_max, samplingEpsilon, windowShape*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is 'End'

- *fs*

  (Optional) Specifies the sampling frequency of the signal. Default is 'Automatic'.

- *points*

  (Optional) Specifies the number of sampling points. Default is 'Automatic'.

  For symmetric windows, the parameters above satisfy the following equation:

  $((points\text{-}1)/fs) = t\_stop - t\_start$

  For periodic windows, the parameters above satisfy the following equation:

  $((points)/fs) = t\_stop - t\_start$

- *sampling*

  (Optional) Specifies the method of computing the sampled data. Legal values: 'No Sampling' (the default), 'Interpolation' or 'Spline'.

- *padding*

  (Optional) Activates data padding to pad the input data with zeros, before or after the input data set. Legal values: 'No Padding' (the default), 'Padding Right', 'Padding Left' or 'Padding Left and Right'. The input parameter is verified by the algorithm and changed if necessary.

- *normalized*

  (Optional) Specifies whether you want to take an average on the raw data to reduce noise and smooth the frequency domain waveform. Specify 1 to turn this on, or 0 to not modify the raw data from calculation. Default is 0.

- *windowType*

  (Optional) Applies a windowing function from a selection of windows. Legal values: 'Rectangular' (the default), 'Hamming', 'Hanning', 'Parzen', 'Welch', 'Blackman', 'Blackman-Harris', 'Bartlett', 'Kaiser', 'Klein' or 'Dolph Chebyshev'.

  Note

  For Hanning, symmetric window shapes are preferred when using a Hanning window in FIR filter design.

  Periodic window shapes are preferred when using a Hanning window in spectral analysis. This is because the Discrete Fourier Transform assumes periodic extension of the input vector. A periodic Hanning window is obtained by constructing a symmetric window and removing the last sample.

- *alpha*

  (Optional) Specifies the alpha or beta value that is required by Hanning, Kaiser, and Dolph Chebyshev windows. Default is 'Default'.

- *nsect*

  (Optional) Specifies the number of points by section. Default is 'Automatic'.

- *nauto*

  (Optional) Specifies the number of points for auto-correlation results. Default is 'Automatic'.

- *ncorr*

  (Optional) Specifies the number of auto-correlation points used for power spectral density computation. Default is 'Automatic'.

- *computationMethod*

  (Optional) Specifies the computation method. Legal values: 'PERIODO' (the default) and 'CORRELO'.

- *f_ref*

  (Optional) Adjusts the results around the y axis so that the point for the specified frequency is 0.0. Default is 'Automatic'.

- *f_min*

  (Optional) Specifies the starting frequency used inside the power spectral density result window. Default is 'Automatic'.

- *f_max*

  (Optional) Specifies the last frequency used inside the power spectral density result window. Default is 'Automatic'.

- *samplingEpsilon*

  (Optional) When *sampling* is set to 'Interpolation', specifies that the input waveform data point (Y value) is to be used rather than the exact interpolated value when the X data of the input waveform is close to the computed X value.

  The computed X value corresponds to *t_start* +(*points* * dX) where dX is the sampling interval retrieved from the *fs*. Default is 'Default'.

- *windowShape*

  (Optional) Specifies the shape of the window. Legal values: 'Symmetric' for standard FFT setup or 'Periodic' for enhancing FFT setup for spectral analysis of periodic signals. Defaults to the global setting in the Waveform Calculator Calculation Options. Default is 'Default'.

  Tip

  For comparing Eldo's FFT results with EZwave's FFT results, select the Periodic option, unless EZwave's FFT is executed on an FFT_INPUT waveform. In this case, Eldo has already considered the periodicity of the input signal.

Return Values

Waveform. The auto-correlation of the input waveform.

Description

Computes the auto-correlation of the input waveform.

Performing an FFT analysis on a random signal to extract significant information is relatively unproductive. A far better method is the calculation of the auto-correlation function (AF). The AF of a signal waveform is an average measure of its time domain properties and therefore especially relevant when the signal is random.

avg

Waveform Calculator Statistical function.

Returns the average value of a waveform.

Usage

avg(*wf*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.
- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.
- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

Return Values

Double. The average of the input waveform.

Returns an error if:

- both *x_start* and *x_end* are greater than the maximum value of the domain of ***wf***, or
- both *x_start* and *x_end* are less than the minimum value of the domain of ***wf***.

Description

The average value of a waveform is computed as follows:

integ(***wf***, *x_start*, *x_end*) / (*x_end* - *x_start*)

If *x_end* is greater than the maximum value of the domain of ***wf***, and *x_start* is less than the maximum value, *x_end* is automatically adjusted to the maximum value of the domain of ***wf***.

If *x_start* is less than the minimum value of the domain of ***wf***, and *x_end* is greater than the minimum value, *x_start* is automatically adjusted to the minimum value of the domain of ***wf***.

bandpass

Waveform Calculator Measurement function.

Calculates the bandwidth at the level at which the measurement is made, for a bandpass-shaped waveform.

Usage

bandpass(***wf***, *topline*, *offset*, *x_start*, *x_end*, *option*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold for the source waveform. Specify 'Automatic' to have this value computed. Default is 'Automatic'.

- *offset*

  (Optional) Specifies the offset value applied relative to the Topline value. The offset is always in dB, and you must also include a sign, minus(-) or plus(+), along with the specified level. Default is -3.

- *x_start*

  (Optional) Specifies the start x value for the calculation. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the stop x value for the calculation. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' returns a numerical value of the bandwidth. 'ANNOTATION' plots the source waveform, annotated with the bandwidth, the lower band edge, and the upper band edge. Default is 'VALUE'.

Return Values

The result format depends on the value of the *option* parameter.

bandwidth

Waveform Calculator Measurement function.

Returns the bandwidth of the input waveform based on the calculation of its topline.

Usage

topline(***wf***[, *db*, *mode*, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *db*

  (Optional) Specifies the threshold value below the baseline (in dB) for the bandwidth calculation. Default is 3.0.

- *mode*

  (Optional) Specifies the bandwidth mode. Allowed values: ‘low’ (low pass filter), ‘band’ (bandpass filter), or ‘high’ (high-pass filter). Default is ‘band’.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

Return Values

Double. The bandwidth of the input waveform.

baseline

Waveform Calculator Measurement function.

Returns the baseline of the input waveform based on the calculation of histograms.

Usage

baseline(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

Return Values

Waveform. The baseline of the input waveform.

calcvartype

Waveform Calculator Miscellaneous function.

Returns the type of a Waveform Calculator variable.

Usage

calcvartype(***name***)

Arguments

- ***name***

  (Required) Specifies a variable name, entered between quotes (‘ ’).

Return Values

Integer. Returns the variable type:

- 2 - variable is a waveform
- 1 - variable is a scalar
- 0 - variable does not exist.

Description

Checks whether a specified variable exists in the Waveform Calculator *calc* folder or workspace and, if it exists, determines whether the variable is a waveform or a scalar.

Examples

```
calcvartype(’variable_a’)
```

cdf

Creates a CDF (cumulative density function) from a source waveform.

Usage

cdf(*wf*[, *shape*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *shape*

  (Optional) Legal values are 'continuous' (the default) and 'step'. Using 'step' specifies the CDF is plotted as a step waveform.

Description

Creates a CDF from a source waveform.

If *wf* is histogram data, the computed CDF is based on the input data of the histogram, if available, not on the histogram values. The number of points used is the number of points in the histogram input data and not the number of bins in the histogram.

ceil

Waveform Calculator Mathematical function.

Computes the smallest integral value not less than each data point of the waveform.

Usage

ceil(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

Return Values

Waveform. A waveform containing the ceiling values of the input waveform.

Description

Computes the smallest integral value not less than each data point of ***wf***. Upon successful completion, each returned value is expressed as a type double.

Related Topics

trunc

round

floor

chirp

Waveform Calculator Signal Processing function.

Computes the Chirp Transform of the input waveform.

Usage

chirp(***wf***[, *t_start*, *t_stop*, *fs*, *points*, *sampling*, *padding*, *windowType*, *alpha*, *res_nb_points, f_ref, f_min, f_max*, *samplingEpsilon*, *windowShape*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is 'End'.

- *fs*

  (Optional) Specifies the sampling frequency of the signal. Default is 'Automatic'.

- *points*

  (Optional) Specifies the number of sampling points. Default is 'Automatic'.

  For symmetric windows, the parameters above satisfy the following equation:

  ((*points*)/*fs*) = *t_stop* - *t_start*

  For periodic windows, the parameters above satisfy the following equation:

  ((*points* - 1)/*fs*) = *t_stop* - *t_start*

- *sampling*

  (Optional) Specifies the method of computing the sampled data. Legal values: 'No Sampling' (the default), 'Interpolation' or 'Spline'.

- *padding*

  (Optional) Activates data padding to pad the input data with zeros before or after the input data set. Legal values: 'No Padding' (the default), 'Padding Right', 'Padding Left' or 'Padding Left and Right'. The input parameter is verified by the algorithm and changed if necessary.

- *windowType*

  (Optional) Applies a windowing function from a selection of windows. Legal values: 'Rectangular' (the default), 'Hamming', 'Hanning', 'Parzen', 'Welch', 'Blackman', 'Blackman-Harris', 'Bartlett', 'Kaiser', 'Klein' or 'Dolph Chebyshev'.

- *alpha*

  (Optional) Specifies the alpha or beta value that is required by Hanning, Kaiser, and Dolph Chebyshev windows.

- *res_nb_points*

  (Optional) Specifies the number of points of the result waveform. Default is 'Automatic'.

- *f_ref*

  (Optional) Adjusts the results around the y axis so that the point for the specified frequency is 0.0. Default is 'Automatic'.

- *f_min*

  (Optional) Specifies the starting frequency used inside the chirp result window. Default is 'Automatic'.

- *f_max*

  (Optional) Specifies the last frequency used inside the chirp result window. Default is 'Automatic'.

- *samplingEpsilon*

  (Optional) When *sampling* is set to 'Interpolation', specifies that the input waveform data point (Y value) is to be used rather than the exact interpolated value when the X data of the input waveform is close to the computed X value. Defaults to the global setting in the "Waveform Calculator Calculation Options" on page 545.

  The computed X value corresponds to *t_start* +(*points* * dX) where dX is the sampling interval retrieved from the *fs*.

- *windowShape*

  (Optional) Specifies the shape of the window. Legal values: 'Symmetric' for standard FFT setup or 'Periodic' for enhancing FFT setup for spectral analysis of periodic signals. Defaults to the global setting in the "Waveform Calculator Calculation Options" on page 545.

Tip

For comparing Eldo's FFT results with EZwave's FFT results, select the Periodic option, unless EZwave's FFT is executed on an FFT_INPUT waveform. In this case, Eldo has already considered the periodicity of the input signal.

Return Values

Waveform. The Chirp Transform of the input waveform.

Description

The DFT of a signal can be computed in a very efficient manner using the FFT. Equivalently, this corresponds to computation of samples of the Z-Transform of a finite-length sequence taken at equally spaced points around the unit circle. The most efficient algorithm used for computing the DFT in the Z domain is the Chirp Z-Transform.

All input and output parameters can be selected in the same way as for the FFT option.

Note

A normalization is achieved by dividing all the data by: (*points*)/2.

complex

Waveform Calculator Complex function.

Constructs a complex waveform from two input waveforms.

Usage

complex(***wf1***, [*x_start1*, *x_end1*,] ***wf2***[, *x_start2*, *x_end2*])

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.
- *x_start1*

  (Optional) Specifies the x value at the beginning of an interval on ***wf1***.
- *x_end1*

  (Optional) Specifies the x value at the end of an interval on ***wf1***.
- ***wf2***

  (Required) Specifies the second input waveform name.
- *x_start2*

  (Optional) Specifies the x value at the beginning of an interval on ***wf2***.
- *x_end2*

  (Optional) Specifies the x value at the end of an interval on ***wf2***.

Return Values

Waveform. The complex waveform from two input waveforms.

Description

Constructs a complex waveform from two input waveforms. The input waveforms can be one of the following:

- Gain in decibels (***wf1***) and phase in radians (***wf2***)
- Real part (***wf1***) and imaginary part (***wf2***)

compress

Waveform Calculator RF function.

Extracts the y-axis value of the waveform at the point where the difference between the actual value of the waveform and the linear extrapolation of the waveform based on the computed slope value becomes greater than ***val***.

Usage

compress(***wf***, ***val***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.
- ***val***

  (Required) Specifies the compress value.
- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.
- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double/array. The y-axis value corresponding to the compressed input waveform.

Description

Extracts the y-axis value of the waveform at the point where the difference between the actual value of the waveform and the linear extrapolation of the waveform based on the computed slope value becomes greater than ***val***.

compresscompound

Waveform Calculator RF function.

Computes the N^{th} compression point for the specified harmonic.

Usage

compresscompound(*wf*, *harmonicindex*, [*nDb*, *ePin*, *rload*, *sweepParam*, *measType*])

Arguments

- ***wf***

  (Required) Specifies the input compound waveform name.

- ***harmonicindex***

  (Required) The harmonic index to be used.

- *nDb*

  (Optional) Specifies the delta between the waveform and the ideal slope at the compression point. Default is 1.

- *ePin*

  (Optional) Specifies the input extrapolation point. Default is 'Automatic', the first point in the input power sweep.

- *rload*

  (Optional) Specifies the internal resistance of the source. If specified, this is used to convert the voltage to dBm. Default is 50.

- *sweepParam*

  (Optional) Specifies the sweep parameter used in the input power sweep. Typically, this is in dBm. Default is 'Automatic'.

- *measType*

  (Optional) Specifies the choice of returned result, 'Input', 'Output' or 'Both'. Default is 'Both'.

Return Values

Double. The N^{th} compression point for the specified harmonic.

Related Topics

wave compressvri

compressvri

concat

Waveform Calculator Miscellaneous function.

Computes the concatenation of two waveforms, extending the first waveform by appending the second.

Usage

concat(***wf1***, [*x_start1*, *x_end1*,] ***wf2***[, *x_start2*, *x_end2*])

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- *x_start1*

  (Optional) Specifies the x value at the beginning of an interval on ***wf1***.

- *x_end1*

  (Optional) Specifies the x value at the end of an interval on ***wf1***.

- ***wf2***

  (Required) Specifies the second input waveform name.

- *x_start2*

  (Optional) Specifies the x value at the beginning of an interval on ***wf2***.

- *x_end2*

  (Optional) Specifies the x value at the end of an interval on ***wf2***.

Return Values

Waveform. The concatenated waveforms.

Description

This command computes the concatenation of two waveforms, extending the first waveform by appending the second. The first data point of the second waveform overlays the last data point of the first waveform. The x-axis units of both waveforms must be identical.

conjugate

Waveform Calculator Complex function.

Constructs the conjugate of the source complex waveform.

Usage

conjugate(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

Return Values

Waveform. The conjugate of the input complex waveform

constellationdiagram

Waveform Calculator RF function.

Computes the constellation diagram of the complex input waveform.

Usage

constellationdiagram(***wf***, ***period***, ***delay***)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***period***

  (Required) The Symbol Period can be deduced from the circuit. It is used as a sampling period for the calculation.

- ***delay***

  (Required) Specifies the calculation start time. By default, this is set to 0.

Return Values

Waveform. The constellation diagram of the input waveform.

Description

Computes the constellation diagram of the complex input waveform. The constellation diagram is a sampled view of a trajectory diagram.

continuous

Waveform Calculator Miscellaneous function.

Converts any analog waveform to a continuous analog waveform.

Usage

continuous(***wf***)

Arguments

- ***wf***

  (Required) Specifies the input waveform.

Return Values

Waveform. The input waveform as a continuous waveform.

convolution

Waveform Calculator Signal Processing function.

Computes the convolution of the two input waveforms.

Usage

convolution(***wf1***, ***wf2***[, *points1*, *points2*, *fs, samplingEpsilon*])

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- ***wf2***

  (Required) Specifies the second input waveform name.

- *points1*

  (Optional) Specifies the number of points of the first input waveform. Default is 'Automatic'.

- *points2*

  (Optional) Specifies the number of points of the second input waveform. Default is 'Automatic'.

- *fs*

  (Optional) Specifies the frequency of the signal. Default is 'Automatic'.

- *samplingEpsilon*

  (Optional) When *sampling* is set to 'Interpolation', specifies that the input waveform data point (Y value) is to be used rather than the exact interpolated value when the X data of the input waveform is close to the computed X value. Default is 10^{-6}.

  The computed X value corresponds to *t_start* +(*points* * dX) where dX is the sampling interval retrieved from the *fs*.

Return Values

The convolution of the two input waveforms.

Description

Computes the convolution of the two input waveforms. For two finite data sequences x(*n*),*n*=0,...,N-1 and h(*n*),*n*=0,...,M-1 the discrete convolution is defined as follows:

y(*n*) = Sum(x(*m*).h(*n*-*m*)) with *m* varying from -infinity to +infinity

For signals x(*n*), h(*n*) that are periodic with period N, the discrete FFT of their periodic convolution is equal to the multiplication of the separate FFT results. This relation is called the Discrete Convolution Theorem. Using zero padding of x(*n*) and h(*n*) to make circular convolution yield the same result as linear convolution, the following method can then be applied for computing the linear convolutions of two finite data sequences $x(n), n=0,...,N-1$ and $h(n), n=0,...,M-1$:

1. Zero Padding

   $x'(n) = x(n)$ $n=0,...,N-1$

   $x'(n) = 0$ $n=N,...,N+M-1$

   $h'(n) = h(n)$ $n=0,...,M-1$

   $h'(n) = 0$ $n=M,...,N+M-1$

2. Multiply FFT's of $x'(n)$ and $h'(n)$

   $x'(n)$->$X'(k)$ FFT

   $h'(n)$->$H'(k)$ FFT

   Multiplication:

   $X'(k).H'(k)=Y'(k)$

3. Inverse FFT

   $Y'(k)$->$y(n)=x(n)*h(n)$ IFFT

cos

Waveform Calculator Trigonometric function.

Computes the cosine of the waveform.

Usage

cos(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The cosine of the input waveform.

Description

Computes the cosine of ***wf***, by default measured in degrees.

cosh

Waveform Calculator Trigonometric function.

Computes the hyperbolic cosine of the waveform.

Usage

cosh(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The hyperbolic cosine of the input waveform.

cot

Waveform Calculator Trigonometric function.

Computes and returns the cotangent.

Usage

cot(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The cotangent of the input waveform.

Description

Computes and returns the cotangent of ***wf***. Cotangent is defined as the reciprocal of the tangent; that is, cot(***wf***)=1/tan(***wf***).

coth

Waveform Calculator Trigonometric function.

Computes and returns the hyperbolic cotangent.

Usage

coth(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The hyperbolic cotangent of the input waveform.

Description

Computes and returns the hyperbolic cotangent of wf. Hyperbolic cotangent is defined as the reciprocal of the hyperbolic tangent; that is, coth(***wf***)=1/tanh(***wf***).

cphase

Waveform Calculator Complex function.

Returns the phase of the input complex waveform in radians or degrees.

Usage

cphase(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The cphase of the input complex waveform.

Description

Returns the phase of the input complex waveform with unlimited bounds. In radians or degrees, according to the option set in **Edit** > **Options** > **Axis Data Units** > **Phase**.

cphytrigger

Waveform Calculator Statistical function.

Creates first zero-crossing trigger waveform for C-Phi. Also used to compute UI jitter and transition jitter.

Tip

Used in conjunction with eyecphy function.

Usage

cphytrigger(***wf1***, ***wf2***, ***wf3***, *ui*, *tolerance*, *computation*, *input*, *x_start*, *x_end*)

Arguments

- ***wf1***

  (Required) Specifies the input waveform name for V(A)-V(B) (or V(A), see *input* parameter).

- ***wf2***

  (Required) Specifies the input waveform name for V(B)-V(C) (or V(B), see *input* parameter).

- ***wf3***

  (Required) Specifies the input waveform name for V(C)-V(A) (or V(C), see *input* parameter).

- *ui*

  (Optional) Specifies the minimum Unit Interval for the trigger. Default is ‘Automatic’.

- *tolerance*

  (Optional) Specifies the tolerance on *ui* to detect transition. Valid values are ‘Automatic’, a double value between 0.0 and 1.0, or a number followed by ‘%’ (e.g. ‘20%’).

  The transition threshold = (1 - *tolerance*/100) × *ui*.

  Default is ‘Automatic’ (20%).

- *computation*

  (Optional) Specifies the type of calculation:

  - ‘trigger’ creates a trigger waveform.
  - ‘uijitter’ creates a UI jitter waveform.
  - ‘transitionjitter’ creates a transition jitter waveform.

  Default is ‘trigger’.

- *input*

  (Optional) Specifies the type of input:

  - ‘differentials’ specifies waveforms are differential lines, ***wf1*** is V(A)-V(B), ***wf2*** is V(B)-V(C), and ***wf3*** is V(C)-V(A).
  - ‘lines’ specifies waveforms are lines, ***wf1*** is V(A), ***wf2*** is V(B), and ***wf3*** is V(C).

  Default is ‘differentials’.

- *x_start*

  (Optional) Start time of the input waveforms. Default is ‘Begin’.

- *x_end*

  (Optional) Stop time of the input waveforms. Default is ‘End’.

Return Values

Waveform. Returns a trigger waveform, or a UI jitter waveform, or a transition jitter waveform, according to the setting of the *computation* parameter.

The trigger waveform is the waveform used to trigger the C-Phy eye, which is right aligned to the trigger. Only rising edges are considered as triggering events, falling edges are at time of previous edge + UI. The UI jitter waveform is the variation of the UI, that is, current_period – average_period.The transition jitter waveform is the time for the input differential signals to cross the 0 level line. This equals 0 when there is a single cross.

Related Topics

C-Phy Eye Calculation

eye

eyecphy

eyesetmask

eyewithtrigger

crosscorrelation

Waveform Calculator Signal Processing function.

Computes the cross-correlation of the two input waveforms.

Usage

crosscorrelation(***wf1***, ***wf2***[, *points1*, *points2*, *fs*])

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- ***wf2***

  (Required) Specifies the second input waveform name.

- *points1*

  (Optional) Specifies the number of points of the first input waveform. Default is 'Automatic'.

- *points2*

  (Optional) Specifies the number of points of the second input waveform. Default is 'Automatic'.

- *fs*

  (Optional) Specifies the frequency of the signal. Default is 'Automatic'.

Return Values

Waveform. The cross-correlation of the two input waveforms.

Description

For two finite data sequences x(*n*),*n*=0,...,N-1 and y(*n*),*n*=0,...,M-1 the discrete cross-correlation is defined as follows:

Rxy(*n*) = Sum(x(*m*) * y(*m*-*n*)) with *m* varying from -infinity to +infinity

Because of the following relation:

Rxy(*n*) = x(*n*) * y(-*n*)

the cross-correlation can be computed by performing a Convolution on x(*n*) and y(-*n*), the time-reversed version of y(*n*).

crossing

Waveform Calculator Measurement function.

Measures the crossing of a waveform relative to default (automatically calculated) or user-specified Y level.

Usage

crossing(***wf***)

crossing(***wf***, *ylevel*, *slopetrigger*)

crossing(***wf***, *ylevel*, *slopetrigger*, *x_start*, *x_end*)

crossing(***wf***, *ylevel*, *slopetrigger*, *x_start*, *x_end*, *option*)

crossing(***wf***, *ylevel*, *slopetrigger*, *x_start*, *x_end*, *option*, *param*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *ylevel*

  (Optional) Specifies the y value that sets the threshold of the signal. Specify 'Automatic'to have this value computed.

- *slopetrigger*

  (Optional) Specifies the signal edge from which the measurement begins. Specify 'Rising', 'Falling' or 'Either' (default).

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' indicates numerical values, 'WF' indicates a waveform (default), and 'ANNOTATION' indicates the input waveform plot will be annotated with results.

- *param*

  (Optional) Specifies the simulation parameter to be used to generate the result waveform when option = 'WF'. Default is 'CROSSINGX'.

Return Values

Waveform. The crossing of the input waveform.

cycle2cyclejitter

Waveform Calculator Time-Domain Jitter function.

Computes the variation in time of the difference between the duration of two adjacent N-cycle samples over part of, or the whole waveform.

Usage

cycle2cyclejitter(*wf*[, *t_start*, *t_ stop*, *edgetrigger, nbcycles, single_threshold, topline. baseline, threshold*])

Arguments

- *wf*

  (Required) Specifies the input waveform for which to calculate the jitter.

- *t_start*

  (Optional) Specifies the X (time) value at the beginning of the jitter analysis interval. Possible values are ‘Begin’ or a double value.

- *t_stop*

  (Optional) Specifies the X (time) value at the end of the jitter analysis interval. Possible values are ‘End’ or a double value.

- *edgetrigger*

  (Optional) Defines the waveform edge(s) that are taken into account during the jitter calculation. Possible values are ‘Rising’, ‘Falling’, or ‘Either’.

- *nbcycles*

  (Optional) Specifies the maximum number of waveform cycles in the time interval which will participate in the jitter calculation. Possible values are an integer value or ‘Default’ (1).

- *single_threshold*

  (Optional) If ‘True’, the single *threshold* will be used (*topline*/*baseline* values are ignored). If ‘False’, *topline*/*baseline* will be used. Used for source waveform period calculation. Default is ‘False’.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the source signal, used for period calculation. Default is ‘Automatic’.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the source signal, used for period calculation. Default is ‘Automatic’.

- *threshold*

  (Optional) Specifies the y value that sets a single threshold for the source signal, used for period calculation. Default is ‘Automatic’.

Return Values

Waveform. The N-cycle jitter waveform of the input waveform.

Description

N-cycle jitter can be described as the variation in time of the difference between the duration of two adjacent N-cycle samples (each sample consists of a selected number of adjacent cycles) over the whole, or part of, the waveform.

For N=1, N-cycle jitter equals the cycle-to-cycle jitter. This type of jitter is calculated only on the source waveform. C2C helps to evaluate the instantaneous and very short-term signal stability.

The jitter RMS value horizontal marker is displayed over the result waveform.

Related Topics

Jitter Measurement Types

cycletocyclejitter

Waveform Calculator Phase Noise function.

Returns the cycle-to-cycle jitter, applied to SST Noise Analysis results.

Usage

cycletocyclejitter(***wf***, ***f0***[, *x_start*, *x_end*, *sampling_nb_points*, *nbcycles*])

Arguments

- ***wf***

  (Required) Specifies the name of the input waveform on which the cycle-to-cycle jitter is calculated.

- ***f0***

  (Required) Specifies the Fundamental Frequency of SST Noise Analysis. Default is 'Automatic'.

- *x_start*

  (Optional) Specifies the X (frequency) value at the start of the jitter analysis interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the X (frequency) value at the end of the jitter analysis interval. Default is 'End'.

- *sampling_nb_points*

  (Optional) Defines whether sampling is applied to the source waveform and specifies the number of sampling points. Default value is 'Automatic', for which 100 sampling points are considered.

- *nbcycles*

  (Optional) Specifies the maximum number of waveform cycles which will participate in the jitter calculation. Possible value is an integer value. Default is 'Default', corresponding to 1 cycle.

Return Values

Double. The cycle-to-cycle jitter value of the input waveform.

Related Topics

Jitter Measurement Types

cycletocyclejitterbyintegration

Waveform Calculator Phase Noise function.

Calculates rms cycle-to-cycle or adjacent k-period jitter by Sphi integration.

Usage

cycletocyclejitterbyintegration(***wf***, ***fund***[, *fstart*, *fstop*, *k*])

Arguments

- ***wf***

  (Required) Specifies the two-sided Sphi density waveform (in dB rad^2/Hz vs frequency in Hz) as computed using, for example, sphibyjitter. Only the positive side of Sphi is used.

- ***fund***

  (Required) Specifies the fundamental frequency.

- *fstart*

  (Optional) Specifies the lower frequency limit in integration. Default is ‘Begin’.

- *fstop*

  (Optional) Specifies the upper frequency limit in integration. Default is ‘End’.

- *k*

  (Optional) Specifies the number of accumulated cycles. Default is 1.

Return Values

Double. The rms cycle-to-cycle or adjacent k-period jitter by Sphi integration of the input waveform.

datatocomplex

Waveform Calculator Miscellaneous function.

Creates a complex waveform based on one or two arrays of data points.

Usage

datatocomplex(***array_of_datapoints***)

datatocomplex(***array_of_x***, ***array_of_y***)

Arguments

- ***array_of_datapoints***

  (Required) Specifies the input array of points. For example, [[*x0*, *re0*, im0], …, [*xn*, *ren, imn*]].

- ***array_of_x***

  (Required) Specifies the input array of x values. For example, [*x0*, *x1*, …, *xn*].

- ***array_of_y***

  (Required) Specifies the input array of y values. For example, [*a0 + b0j*, *a1 + b1j*, …, *an + bnj*].

Return Values

Waveform. A new complex waveform from the specified input array(s).

Description

If one array is specified, it must contain data points wrapped between square brackets, for example [x, re, im].

If two arrays are specified, the first array contains the x values, and the second array contains complex values in the form of a + bj.

datatodig

Waveform Calculator Miscellaneous function.

Creates a digital waveform based on one or two arrays of data events.

Usage

datatodig(*array_of_datapoints*)

datatodig(*array_of_x*, *array_of_y*)

Arguments

- ***array_of_datapoints***

  (Required) Specifies the input array of points. For example, [[*x0*, *y0*], …, [*xn*, *yn*]].

- ***array_of_x***

  (Required) Specifies the input array of x values. For example, [*x0*, *x1*, …, *xn*].

- ***array_of_y***

  (Required) Specifies the input array of y transition states. For example, [*y0*, *y1*, …, *yn*].

Return Values

Waveform. A new digital waveform from one or two arrays of data events.

Description

If one array is specified, it must contain data points wrapped between square brackets ([ ]). The first element defines the *x* value, and the second one defines the *y* value. Elements are separated by a comma. For example, [[*x0*, *y0*], [*x1*, *y1*], …, [*xn*, *yn*]].

If two arrays are specified, the first array contains the *x* values, and the second array contains the *y* values. For example, [*x0*, *x1*, …, *xn*], [*y0*, *y1*, …, *yn*].

The *y* values represent digital states. There are two representations supported for the states; the IEEE 1164 Standard Logic states:

{ ‘0’, ‘1’, ‘U’, ’X’ ,’Z’, ‘W’, ‘L’, ‘H’, ‘-’ }

and a subset of the Verilog state set:

{"Hi0", "Hi1", "HiX", "HiZ", "St0", "St1", "StL", "StH", "StZ", "StX", "Sm0", "Sm1", "SmL", "SmH", "SmZ","SmX", "Su0", "Su1", "SuL", "SuH", "SuZ", "SuX", "Me0", "Me1", "MeL", "MeH", "MeZ", "MeX", "We0", "We1", "WeL", "WeH", "WeZ", "WeX", "Pu0", "Pu1", "PuL", "PuH","PuZ", "PuX", "La0", "La1", "LaL", "LaH", "LaZ", "LaX"}.

Note

State entries are case-sensitive and must be in quotes. Mixing state types is not permitted and any invalid or unrecognized state is silently dropped.

datatowf

Waveform Calculator Miscellaneous function.

Creates an analog waveform based on one or two arrays of data points.

Usage

datatowf(***array_of_datapoints***)

datatowf(***array_of_x***, ***array_of_y***)

Arguments

- ***array_of_datapoints***

  (Required) Specifies the input array of points. For example, [[*x0*, *y0*], …, [*xn*, *yn*]].

- ***array_of_x***

  (Required) Specifies the input array of x values. For example, [*x0*, *x1*, …, *xn*].

- ***array_of_y***

  (Required) Specifies the input array of y values. For example, [*y0*, *y1*, …, *yn*].

Return Values

Waveform. A new waveform from the specified input array(s).

Description

If one array is specified, it must contain data points wrapped between square brackets ([ ]). The first element defines the *x* value, and the second one defines the *y* value. Elements are separated by a comma. For example, [[*x0*, *y0*], [*x1*, *y1*], …, [*xn*, *yn*]].

If two arrays are specified, the first array contains the *x* values, and the second array contains the *y* values. For example, [*x0*, *x1*, …, *xn*], [*y0*, *y1*, …, *yn*].

dchysteresis

Waveform Calculator Miscellaneous function.

Measures the thresholds and width of a DC Hysteresis waveform relative to a default, or specified, Y-level.

Usage

dchysteresis(*wf*)

dchysteresis(*wf*, *ylevel***)**

dchysteresis(*wf*, *ylevel*, *measure***)**

dchysteresis(*wf*, *ylevel*, *measure*, *x_start*, *x_end*, *option***)**

dchysteresis(*wf*, *ylevel*, *measure*, *x_start*, *x_end*, *option*, *param***)**

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *ylevel*

  (Optional) Specifies the y level at which the signal is measured. Specify 'Automatic' to have this value computed (as the middle of the top and base lines).

- *measure*

  (Optional) Specify 'Width' (default) to compute the hysteresis width. Specify 'Left' or 'Right' to compute the thresholds.

- *x_start*

  (Optional) Specifies the x value (run number) at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value (run number) at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' indicates numerical values, 'WF' indicates a waveform, and 'ANNOTATION' indicates the input waveform plot will be annotated with results.

- *param*

  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = 'WF'.

Return Values

Waveform. The thresholds and the width of the DC Hysteresis input waveform.

db

Waveform Calculator Complex function.

Converts the magnitude data of the input waveform to decibels.

Usage

db(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The magnitude data of the input waveform in dB.

db10

Waveform Calculator Complex function.

Converts the magnitude data of the input waveform to decibels (log base 10).

Usage

db10(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The scaled input waveform.

Description

Converts the magnitude data of the input waveform to decibels using the following equation:

db10 = 10 * log10 (x)

deg

Waveform Calculator Signal Processing function.

Converts the trigonometric angle of a waveform to degrees.

Usage

deg(*wf*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. A waveform corresponding to the converted input waveform.

delay

Waveform Calculator Measurement function.

Measures the delay between edges of a source waveform and a reference waveform.

Usage

delay(***wf1***, ***wf2***, *topline1*, *baseline1*, *dlev1*, *topline2*, *baseline2*, *dlev2*, *count1*, *count2*, *edgetrigger*, *inverting*, *closestedge*, *x_start*, *x_end*, *option*, *param*)

Arguments

- ***wf1***

  (Required) Specifies the source waveform name.

- ***wf2***

  (Required) Specifies the reference waveform name.

- *topline1*

  (Optional) Specifies the y value that sets the high threshold for the source waveform. Specify 'Automatic' to have this value computed. Default is 'Automatic'.

- *baseline1*

  (Optional) Specifies the y value that sets the low threshold for the source waveform. Specify 'Automatic' to have this value computed. Default is 'Automatic'.

- *dlev1*

  (Optional) Specifies the percentage relative to the topline and baseline values for the source waveform. Default is '50%'.

- *topline2*

  (Optional) Specifies the y value that sets the high threshold for the reference waveform. Specify 'Automatic' to have this value computed. Default is 'Automatic'.

- *baseline2*

  (Optional) Specifies the y value that sets the low threshold for the reference waveform. Specify 'Automatic' to have this value computed. Default is 'Automatic'.

- *dlev2*

  (Optional) Specifies the percentage relative to the topline and baseline values for the reference waveform. Default is '50%'.

- *count1*

  (Optional) Specifies the starting edge to consider on the source waveform. Default is 'All'.

- *count2*

  (Optional) Specifies the starting edge to consider on the reference waveform. Default is 'All'.

- *edgetrigger*

  (Optional) Specifies the signal edge from which the measurement begins. Specify ‘Rising’, ‘Falling’ or ‘Either’ (default).

- *inverting*

  (Optional) If set to 0, starts on the reference waveform at the previous edge with the same polarity (non-inverting) as the source waveform. If set to 1, starts on the reference waveform at the previous edge with the opposite polarity (inverting) as the source waveform. Default is 0.

- *closestedge*

  (Optional) If set to 1, finds the closest reference edge to display the reference edge nearest to the measured edge. Default is 0.

- *x_start*

  (Optional) Specifies the start time for the calculation. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the stop time for the calculation. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ indicates numerical values, ‘WF’ indicates a waveform (default), and ‘ANNOTATION’ indicates the input waveform plot will be annotated with results. Default is ‘WF’.

- *param*

  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = ‘WF’. Default is ‘index’.

Return Values

Returns the delay between edges of the source waveform and a reference waveform. Returns a waveform if *option* is set to ‘WF’.

derive

Waveform Calculator Mathematical function.

Computes the derivative of the input waveform at the single point ***x_val***.

Usage

derive(***wf***, ***x_val***)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***x_val***

  (Required) Specifies the X value of the derivative point.

Return Values

Waveform. The derivative at the specified X value.

drv

Waveform Calculator Mathematical function.

Computes the derivative of the input waveform.

Usage

drv(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The derivative of the input waveform.

dtoa

Waveform Calculator Miscellaneous function.

Constructs an analog waveform from a digital bus.

Usage

dtoa(***wf***, ***coeff_add***, ***coeff_mul***, *t_com*)

dtoa(***wf***, ***coeff_add***, ***coeff_mul***, *use_interpolate*, *use_t_com*, *t_com*, *radix*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***coeff_add***

  (Required) Specifies the additive coefficient in the calculation of the analog values.

- ***coeff_mul***

  (Required) Specifies the multiplicative coefficient in the calculation of the analog values.

- *use_interpolate*

  (Optional) Specifies whether to use the interpolation algorithm.

- *use_t_com*

  (Optional) Specifies whether to use the commutation time in non-interpolate mode.

- *t_com*

  (Optional) The commutation time.

- *radix*

  (Optional) Specifies the radix in which to interpret the bus values.

Return Values

Waveform. The converted waveform.

Examples

```
dtoa(wf("<tutorial/Time-Domain_Results>decode_pw7/cnt1[0:11]"), 0.0, 1.0,
0, 1, 2.0E-9, -10)
```

dtoaonbit

Waveform Calculator Miscellaneous function.

Constructs an analog waveform on bits.

Usage

dtoaonbit(*wf*)

dtoaonbit(*wf*, *s_0, s_1,w_0,*
w_1, s_u, w_u, h_z, u, m, rise, fall)

Arguments

The following arguments define the analog value used when matching a digital state:

- ***wf***

  (Required) Specifies the input bit name.

- *s_0*

  (Optional) '0' Forcing 0 (default 0.0, ‘Automatic’).

- *s_1*

  (Optional) '1' Forcing 1 (default 5.0, ‘Automatic’).

- *w_0*

  (Optional) 'L' Weak 0 (default same value as '0', ‘Automatic’).

- *w_1*

  (Optional) 'H' Weak 1 (default same value as '1', ‘Automatic’).

- *s_u*

  (Optional) 'X' Forcing Unknown (default ('0'+'1')/2, ‘Automatic’).

- *w_u*

  (Optional) 'W' Weak Unknown (default ('L'+'H')/2, ‘Automatic’).

- *h_z*

  (Optional) 'Z' High Impedance (default 'U' or previous value, ‘Automatic’).

- *u*

  (Optional) 'U' Uninitialized (default 0.0, ‘Automatic’).

- *m*

  (Optional) '-' Do not care (default 'U' or previous value, ‘Automatic’).

- *rise*

  (Optional) Commutation Rise Time (default 2e-9, ‘Automatic’).

- *fall*

  (Optional) Commutation Fall Time (default 2e-9, ‘Automatic’).

Return Values

Waveform. The analog waveform corresponding to the digital bit.

dutycycle

Waveform Calculator Measurement function.

Measures the duty cycle of a periodic waveform relative to default (automatically calculated) or user-specified topline and baseline levels.

Usage

dutycycle(*wf*)

dutycycle(*wf*, *topline*, *baseline*)

dutycycle(*wf*, *topline*, *baseline*, *edgetrigger*)

dutycycle(*wf*, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*)

dutycycle(*wf*, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*, *option*)

dutycycle(*wf*, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*, *option*, *param*)

dutycycle(*wf*, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*, *option*, *param*, *threshold*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the signal. Specify 'Automatic' to have this value computed.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the signal. Specify 'Automatic' to have this value computed.

- *edgetrigger*

  (Optional) Specifies the signal edge from which the measurement begins. Specify 'Rising', 'Falling' or 'Either' (default).

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' indicates numerical values, 'WF' indicates a waveform (default), and 'ANNOTATION' indicates the input waveform plot will be annotated with results.

- *param*

  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = 'WF'. Default is 'middleX'.

- *threshold*

  (Optional) Specifies the y value that sets the threshold of the signal. Specify 'Automatic' to have this value computed: mean(wf). Default is 'None'.

  Note

  'topline/baseline' and 'threshold' parameters cannot be used simultaneously: use either 'topline/baseline' or 'threshold'.

Return Values

Waveform. The duty cycle of the input waveform.

edgephasenoise

Waveform Calculator RF function.

Computes an edge phase noise waveform from pnoise (or hb) sampled(jitter) analysis.

Usage

edgephasenoise(*wf*, *fund*, *slew_rate*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *fund*

  (Optional) Specifies the fundamental frequency. Default is ‘Automatic’.

- *slew_rate*

  (Optional) Specifies the slope of the signal at the positive or negative edge. Default is ‘Automatic’.

Return Values

Waveform. The edge phase noise waveform in dBc/Hz.

Related Topics

jc

jcc

jee

enob

Waveform Calculator Signal Processing function.

Computes the effective number of bits for the source waveform.

Usage

enob(***wf***, ***f_list***[, *f_min*, *f_max, harm, s_freq, bin*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. Typically this will be an FFT result calculated in EZwave.

- ***f_list***

  (Required) Specifies the list of frequencies to be considered as signal. Frequencies from *f_list* can be selected outside of [*f_min*, *f_max*].

- *f_min*

  (Optional) Specifies the minimum frequency to be considered in the input waveform. Default is ‘Begin’.

- *f_max*

  (Optional) Specifies the maximum frequency to be considered the input waveform. Default is ‘End’.

- *harm*

  (Optional) Specifies the number of harmonics of the signal to be considered. Default value ‘Default’ is 6 (signal + 5 harmonics). If *harm*=-1, all harmonics within [*f_min*, *f_max*] are considered.

- *s_freq*

  (Optional) The sampling frequency of the source waveform. Default value is ‘None’, no sampling frequency. When specified, the sampling frequency is used to manage harmonic folding.

- *bin*

  (Optional) Specifies the bin width to use as signal on the input wave (single sided). Default is ‘Default’.

Return Values

Double. The effective number of bits of the input waveform.

Description

Computes the effective number of bits, a conversion of the SINAD result. This calculation is given by the following:

$$\frac{SINAD - 1.76}{6.02}$$

where:

$SINAD$= the signal to noise and distortion ratio in dB.

1.76 = the quantization error in an ideal ADC.

6.02 = the converts decibels (a $\log_{10}$ representation) to bits (a $\log_2$ representation).

The input waveform is always assumed to be a linear gain (magnitude).

Examples

```
enob(wf("<calc>FFT_V_OUT_DAC_GOOD_"),[4637700.0],0.0,8.0E7, s_freq=80e6)
= 9.792966993387555
```

Related Topics

Signal to Noise Ratio Dialog Box

evmber

Waveform Calculator RF function.

Computes the Error Vector Magnitude and Bit Error Rate of the two input constellation diagrams.

Usage

evmber(*wf1*, *wf2*)

evmber(*wf1*, *wf2*, *mpsk*)

evmber(*wf1*, *wf2*, *mqam*)

evmber(*wf1*, *wf2*, *mpsk*, *mqam*)

Arguments

- ***wf1***

  (Required) Specifies the first constellation diagram name.

- ***wf2***

  (Required) Specifies the second constellation diagram name.

- *mpsk*

  (Optional) Defines the number of ideal states of the reference constellation diagram. The default is 0.

- *mqam*

  (Optional) Defines the number of ideal states of the reference constellation diagram. The default is 0.

Return Values

Array. The Error Vector Magnitude and Bit Error Rate.

Description

Computes the Error Vector Magnitude and Bit Error Rate of the two input constellation diagrams. The Error Vector Magnitude and Bit Error Rate can be computed from a constellation diagram of a modulated signal and a constellation diagram of a reference signal.

For MPSK and MQUAM, the M parameter specified is the number of ideal states of the reference constellation diagram as specified in the definition of the modulated source in the netlist.

Note

Phase Error, Magnitude Error, and Error Vector Magnitude values are displayed as percentages.

exp

Waveform Calculator Mathematical function.

Computes the value of e^{wf}.

Usage

exp(*wf*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The scaled input waveform.

exportcsv

Waveform Calculator Miscellaneous function.

Exports the specified waveforms to a *.csv* file (without header), on the x-axis slice on which all waveforms are defined.

Usage

exportcsv(*file*, [*wfs*], *period*, *precision*, *override*)

Arguments

- ***file***

  (Required) Specifies the full path to the *.csv* file.

- [***wfs***]

  (Required) Specifies an array of waveforms (comma-separated list of waveforms), in square brackets, to be exported.

- *period*

  (Optional) Specifies the sampling period used. (Default is 'Automatic'.)

- *precision*

  (Optional) Specifies the floating-point precision used to save the Y-coordinates of analog waveforms. (Default is 'Default'.)

- *override*

  (Optional) Specify 'true' to override the file if it already exists. (Default is 'false'.)

Return Values

None. Creates a *.csv* file if the specified file does not already exist.

Examples

```
exportcsv('path/to/result.csv', [wf("<asdasd2/TRAN>V(ADC14)"),
wf("<asdasd2/TRAN>V(ADC15)")])
```

exportvcd

Waveform Calculator Miscellaneous function.

Exports the specified (time domain) waveforms to a 4-state *.vcd* file. Voltage waveforms are converted to std_logic (0/1/X) waveforms, using threshold1 and threshold2.

Usage

exportvcd(*file*, [***wfs***])

exportvcd(*file*, [***wfs***], ***threshold1***)

exportvcd(*file*, [***wfs***], ***threshold1***, ***threshold2***)

exportvcd(*file*, [***wfs***], ***threshold1***, ***threshold2***, ***dtx***)

Arguments

- ***file***

  (Required) Specifies the full path to the *.vcd* file.

- [***wfs***]

  (Required) Specifies a single waveform or an array of waveforms (comma-separated list of waveforms), in square brackets, to be exported. The X axis should be in the time domain. The Y axis should be in voltage, or digital (bit, boolean, std_logic, qsim or verilog). It can also be a database or a "<database/folder>" name.

- ***threshold1***

  (Required) Specifies the threshold, or the low threshold, for analog to digital conversion. The value changes to 0 when a falling edge crosses this threshold.

- ***threshold2***

  (Required) Specifies the high threshold for analog to digital conversion. The value changes to 1 when a rising edge crosses this threshold.

- ***dtx***

  (Required) Specifies the duration for the signal to become X after an incomplete transition, for A2D conversion with two thresholds.

Return Values

None. Creates a *.vcd* file if the specified file does not already exist.

Description

This function is useful, for example, to generate a *.vcd* file from an analog simulation to use as stimuli in a digital simulation.

The VCD (Value Change Dump) file format is defined in IEEE Std1800 - standard for SystemVerilog.

Note

For the analog to digital conversion, exportvcd internally uses the function "atod" on page 685.

Examples

The following example uses the EZwave *tutorial.wdb* data.

wf0 is the result of an expression, stored in the calc database. The other waveforms are from tutorial.wdb. All waveforms are digital except "v_load", which is an analog voltage. Thresholds for analog to digital conversion are set to 1V and 4V. The generated file is *output.vcd* in the current directory.

```
wf0 = wf("<tutorial/Time-Domain_Results>decode_pw7/cnt1[9]") ^
wf("<tutorial/Time-Domain_Results>decode_pw7/cnt1[10]")

wfList = [wf("<tutorial/Time-Domain_Results>v_load"), wf("<tutorial/Time-
Domain_Results>decode_pw7/cnt1[4]"), wf0]

exportvcd('output.vcd', wfList, 1.0, 4.0)
```

eye

Waveform Calculator Statistical function.

Constructs an eye diagram from the input waveform.

Note

Replaces the deprecated eyediagram command.

Usage

eye(***wf***, *period*, *offset*, *x_start*, *x_end*, *mask_name*, *mask_x_offset*, *mask_y_offset*, *mask_x_margin*, *mask_y_margin*, *fit_option*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *period*

  (Optional) Specifies the period for the eye diagram. Default is 'Automatic'.

- *offset*

  (Optional) Specifies the offset for the eye diagram. Default is 'Automatic'.

- *x_start*

  (Optional) Start time of the input waveform. Default is 'Begin'.

- *x_end*

  (Optional) Stop time of the input waveform. Default is 'End'.

- *mask_name*

  (Optional) Specifies the name of the mask. Default is 'None'.

- *mask_x_offset*

  (Optional) Specifies the x-offset of the mask. Default is '0'.

- *mask_y_offset*

  (Optional) Specifies the y-offset of the mask. Default is '0'.

- *mask_x_margin*

  (Optional) Specifies the x-margin of the mask. Default is '0'.

- *mask_y_margin*

  (Optional) Specifies the y-margin of the mask. Default is '0'.

- *fit_option*

  (Optional) Specifies the mask margin and offset fitting. fitOption is a binary OR one of the following:

  - 0: no automatic fit
  - 1: automatic fit for x-margin
  - 2: automatic fit for y-margin
  - 3: automatic fit for margin only
  - 4: automatic fit for x-offset

  Default is ‘0’.

Return Values

The eye diagram corresponding to the input waveform.

Related Topics

eyemeasures

eyeamplitude

Waveform Calculator Statistical function.

Returns the amplitude of an eye diagram.

Usage

eyeamplitude(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

Return Values

Double. The amplitude of the input eye diagram.

Description

Computes the difference between the High Level value and the Low Level value.

eyecphy

Waveform Calculator Statistical function.

Constructs a self-triggered C-Phy eye diagram from three input waveforms.

Tip

Used in conjunction with cphytrigger function.

Usage

eyecphy(***wf1***, ***wf2***, ***wf3***, *ui*, *tolerance*, *input*, *x_start*, *x_end*)

Arguments

- ***wf1***

  (Required) Specifies the input waveform name for V(A)-V(B) (or V(A), see *input* parameter).

- ***wf2***

  (Required) Specifies the input waveform name for V(B)-V(C) (or V(B), see *input* parameter).

- ***wf3***

  (Required) Specifies the input waveform name for V(C)-V(A) (or V(C), see *input* parameter).

- *ui*

  (Optional) Specifies the minimum Unit Interval for the trigger. Default is ‘Automatic’.

- *tolerance*

  (Optional) Specifies the tolerance on *ui* to detect transition.

  Transition threshold = (1 - *tolerance*/100) * *ui*.

  Default is ‘Automatic’ (20%).

- *input*

  (Optional) Specifies the type of input:

  - ‘differentials’ specifies waveforms are differential lines, ***wf1*** is V(A)-V(B), ***wf2*** is V(B)-V(C), and ***wf3*** is V(C)-V(A).
  - ‘lines’ specifies waveforms are lines, ***wf1*** is V(A), ***wf2*** is V(B), and ***wf3*** is V(C).

  Default is ‘differentials’.

- *x_start*

  (Optional) Start time of the input waveform windows. Default is ‘Begin’.

- *x_end*

  (Optional) Finish time of the input waveform windows. Default is ‘End’.

Return Values

Waveform. Returns the C-Phy eye diagram corresponding to the three input waveforms. The eye is right-aligned.

Related Topics

C-Phy Eye Calculation

cphytrigger

eye

eyesetmask

eyewithtrigger

eyecrossingamplitude

Waveform Calculator Statistical function.

Returns the amplitude level at which the eye crossings occur on an eye diagram.

Usage

eyecrossingamplitude(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

Return Values

Double. The amplitude level at which the eye crossings occur on an eye diagram.

Description

Computes the mean value of the vertical histogram at the eye crossing.

eyedelay

Waveform Calculator Statistical function.

Returns the delay of an eye diagram.

Usage

eyedelay(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

Return Values

Double. The delay of an eye diagram

Description

Computes the distance from the midpoint of the eye to the time origin, measured in seconds.

eyediagram

Waveform Calculator Statistical function.

Constructs an eye diagram of the input waveform.

Note

This command is deprecated. It is replaced with the eye command.

Usage

eyediagram(*wf*)

eyediagram(*wf*, *period*)

eyediagram(*wf*, *period*, *offset*, *x_start*, *x_end*)

eyediagram(*wf*, *period*, *offset*, *x_start*, *x_end*, *maskName*, *maskXOffset*, *maskYOffset*, *maskXMargin*, *maskYMargin*, *fitOption*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *period*

  (Optional) Specifies the period of the eye diagram.

- *offset*

  (Optional) Specifies the offset of the eye diagram.

- *x_start*

  (Optional) Specifies the start time of the input waveform.

- *x_stop*

  (Optional) Specifies the stop time of the input waveform.

- maskName

  The name of the mask.

- maskXOffset

  The mask X offset.

- maskYOffset

  The mask Y offset.

- maskXMargin

  The mask X margin.

- maskYMargin

  The mask Y margin.

- fitOption

  Mask margin and offset fitting. fitOption is a binary OR of the following:

  - 0x1: automatic fit for X margin
  - 0x2: automatic fit for Y margin
  - 0x4: automatic fit for X offset

  => 3 stands for automatic fit for margin only ; 0 stands for no automatic fit at all.

Return Values

Waveform. The eye diagram corresponding to the input waveform.

eyefalltime

Waveform Calculator Statistical function.

Returns the falltime of an eye diagram.

Usage

eyefalltime(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

Return Values

Double. The falltime of an eye diagram.

Description

Computes the mean time between the high and low threshold values, calculated from 10% to 90% of the eye amplitude.

eyeheight

Waveform Calculator Statistical function.

Returns the height of an eye diagram.

Usage

eyeheight(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

Return Values

Double. The height of an eye diagram.

Description

Computes the difference between the High Level -3stdev and the Low Level +3stdev on an eye diagram waveform.

eyeheightatx

Waveform Calculator Statistical function.

Returns the eye height at X.

Usage

eyeheightatx(*wf*, *x*, *mode*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

- ***x***

  (Required) Specifies the X coordinate of the eye diagram at which the eye height is to be computed.

- ***mode***

  (Required) Specifies the mode as either ‘Inner’, ‘Outer’, ‘PAM3’ or ‘PAM4’.

Return Values

Double. The eye height at the specified X value.

Description

Computes the inner or outer height of an eye diagram at a specified X coordinate. For a PAM4 signal, it returns a list of three inner heights, from top eye to bottom eye.

Examples

To get the inner height at half the Unit Interval for a C-Phy eye waveform (cphyeye), use the following:

```
eyeheightatx(cphyeye, -UI/2, 'inner')
```

eyejitter

Waveform Calculator Statistical function.

Returns the jitter of an eye diagram.

Usage

eyejitter(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

Return Values

Double. The jitter of an eye diagram.

Description

Computes the width of the eye crossing points.

eyemeasures

Waveform Calculator Statistical function.

Returns a set of named measures with their values for an eye diagram.

Usage

eyemeasures(*wf*, *x*, *y*, *type*, *amplitude_ratio*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

- *x*

  (Optional) Specifies the X coordinate of the eye diagram where to compute the inner/outer or PAM4 height. Default is ‘None’ for *type* = ‘NRZ’. Default is C-Phy eye average Unit Interval (UI) divided by -4 for *type* = ‘CPHY’. Required for *type* = ‘PAM3’ and *type* = ‘PAM4’.

- *y*

  (Optional) Specifies the Y coordinate of the eye diagram where to compute the inner/outer width. Default is ‘None’ for *type* = ‘NRZ’. Default is 0 for *type* = ‘CPGY’. Must be ‘None’ for *type* = ‘PAM3’ and *type* = ‘PAM4’.

- *type*

  (Optional) Specifies the eye computation mode. Either ‘NRZ’ (2 levels), ‘CPHY’ (superimposed eyes), ‘PAM3’ (3 levels) or ‘PAM4’ (4 levels). Default is ‘NRZ’.

  When *type* is ‘NRZ’:

  - The X parameter is used to compute at the same time the inner and outer height at the given x.
  - The Y parameter is used to compute at the same time the inner and outer width at the given y.

  When *type* is ‘CPHY’:

  - The default value for the X parameter is the C-Phy eye average Unit Interval divided by -4. This is a quarter of the eye UI, and negative because the C-Phy eye is right aligned. All eye points have negative X values.
  - The default value for the Y parameter is 0. This is where the zero crossing is computed.

  When *type* is ‘PAM 3’ or ‘PAM4’:

  - The X parameter is mandatory, and used to compute all the PAM measures.
  - The Y parameter should not be set.

- *amplitude_ratio*

  (Optional) Used when *type* is 'PAM3' or 'PAM4'. When set to 'Automatic', toplines and baselines begin at 10% and increase in steps of 5% until all points are either below the baseline and above the topline (the Measured Edge Percentage reaches 100%). Valid values are 'Automatic', a double value between 0.0 and 1.0, or a number followed by '%' (e.g. '20%'). Default value is 'Automatic' (at least 10%, increasing by 5%).

Return Values

Array. Returns a list of pairs of named measures with their values on the input eye diagram.

The keywords are the same as those displayed on the "Eye Diagram Tool - Measurement Results Tab" on page 465.

For details of each measurement, refer to "Eye Diagram Measurement Calculations" on page 206.

Examples

To get the inner width and height of a C-Phy eye waveform (cphy_eye), use the following:

```
eyemeasures(cphy_eye, type='CPHY')
```

You can use the eye() and eyemeasures() functions. You can also call dict() on the result of eyemeasures(), then look for a specific result:

```
myEye=eye(…)
myMeasuresFlat=eyemeasures(myEye, …)
myMeasures=dict(myMeasuresFlat)
eyeInnerWidth=myDict['Inner Width']
```

Related Topics

eye

eyerisetime

Waveform Calculator Statistical function.

Returns the risetime of an eye diagram.

Usage

eyerisetime(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

Return Values

Double. The risetime of the input eye diagram.

Description

Computes the mean time between the low and high threshold values, calculated from 10% to 90% of the eye amplitude.

eyesetmask

Waveform Calculator Statistical function.

Adds a mask to an eye diagram waveform.

Usage

eyesetmask(***wf***, *mask_name*, *mask_x_offset*, *mask_y_offset*, *mask_x_margin*, *mask_y_margin*, *fit_option*, *period*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

- *mask_name*

  (Optional) Specifies the name of the mask. When set to 'none', removes the mask from the eye diagram. Default is 'none'.

- *mask_x_offset*

  (Optional) Specifies the x-offset of the mask. Default is '0'.

- *mask_y_offset*

  (Optional) Specifies the y-offset of the mask. Default is '0'.

- *mask_x_margin*

  (Optional) Specifies the x-margin of the mask. Default is '0'.

- *mask_y_margin*

  (Optional) Specifies the y-margin of the mask. Default is '0'.

- *fit_option*

  (Optional) Specifies the mask margin and offset fitting. fitOption is a binary OR one of the following:

  - 0: no automatic fit
  - 1: automatic fit for x-margin
  - 2: automatic fit for y-margin
  - 3: automatic fit for margin only
  - 4: automatic fit for x-offset

  Default is '0'.

- *period*

  (Optional) Specifies the period of the eye diagram. Default is 'Automatic'.

Return Values

Returns nothing. Modifies the mask in the input eye waveform.

Related Topics

C-Phy Eye Calculation

cphytrigger

eye

eyecphy

eyewithtrigger

eyesnr

Waveform Calculator Statistical function.

Returns the SNR of an eye diagram.

Usage

eyesnr(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

Return Values

Double. The SNR of the input eye diagram.

Description

Computes (High Level - Low Level) / (High Level stdev + Low Level stdev).

eyewidth

Waveform Calculator Statistical function.

Returns the width of an eye diagram.

Usage

eyewidth(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

Return Values

Double. The width of the input eye diagram.

Description

Computes the difference between the Cross Eye -3stdev and the Cross Eye +3stdev on an eye diagram waveform.

eyewidthaty

Waveform Calculator Statistical function.

Returns the eye width at Y.

Usage

eyewidthaty(*wf*, y, *mode*)

Arguments

- ***wf***

  (Required) Specifies the input eye diagram waveform name.

- ***y***

  (Required) Specifies the Y coordinate of the eye diagram at which the eye width is to be computed.

- ***mode***

  (Required) Specifies the mode as either ‘Inner’ or ‘Outer’.

Return Values

Double. The eye width of the input eye diagram at the specified Y value.

Description

Computes the inner or outer width of an eye diagram at a specified Y coordinate.

Examples

To get the inner width at level 0 of a C-Phy eye waveform (cphyeye), use the following:

```
eyewidthaty(cphyeye, 0, ‘Inner’)
```

eyewithtrigger

Waveform Calculator Statistical function.

Constructs a triggered eye diagram from a waveform.

Usage

eyewithtrigger(***wf***, ***trigger***, *edgetrigger*, *ratio*, *x_start*, *x_end*)

Arguments

- ***wf***

  (Required) Specifies the input source waveform name.

- ***trigger***

  (Required) Specifies the input trigger waveform name.

- *edgetrigger*

  (Optional) Specifies the edge used to trigger. Value can be 'risingEdge', 'fallingEdge', 'Either' (same as the first edge kind: rising or falling) or 'Both' (means rising and falling edges). Default is 'either'.

- *ratio*

  (Optional) Specifies the alignment ratio, between 0.0and 1.0:

  - 0.0: left-aligned
  - 0.5: centered
  - 1.0: right-aligned

  Default is '0.5'.

- *x_start*

  (Optional) Start time of the input waveform window. Default is 'Begin'.

- *x_end*

  (Optional) Stop time of the input waveform window. Default is 'End'.

Return Values

Waveform. Creates a triggered eye diagram from the input waveforms.

Related Topics

C-Phy Eye Calculation

cphytrigger

eye

eyecphy

eyesetmask

falltime

Waveform Calculator Measurement function.

Measures the difference in time from when the waveform falls from the upper level to the lower level.

Usage

falltime(***wf***)

falltime(***wf***, *topline*, *baseline*)

falltime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*)

falltime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *x_start*, *x_end*)

falltime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *x_start*, *x_end*, *option*)

falltime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *x_start*, *x_end*, *option*, *param*)

falltime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *x_start*, *x_end*, *option*, *param*, *fall*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of a signal. Specify ‘Automatic’ to have this value computed.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of a signal. Specify ‘Automatic’ to have this value computed.

- *low*

  (Optional) Specifies the percentage of the low threshold. This can range from 0% to mid. By default, this is set to 10%. This parameter is a string (for example, ‘10%’).

- *mid*

  (Optional) Specifies the percentage that sets the limit range for the low and up values. By default, this is set to 50%. This parameter is a string (for example, ‘50%’).

- *up*

  (Optional) Specifies the percentage of the high threshold. This can range from mid to 100%. By default, this is set to 90%. This parameter is a string (for example, ‘90%’).

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. Legal values: 'WF' for waveform (default), 'VALUE' for numerical value or array of numerical values and 'ANNOTATION' for plotting your input waveform with the result annotated on it.

  Note

  If *fall* is anything other than 'all', option = 'VALUE' is forced.

- *param*

  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = 'WF'. Default is 'middleX'.

- *fall*

  (Optional) Specifies the occurrence of the result that the measurement will return. For compound waveforms it applies to each element individually. Legal values:

  - 'first' — Specifies the first occurrence of the result.
  - 'all' — Specifies all occurrences of the result. Default.
  - 'last' — Specifies the last occurrence of the result.
  - *n* or *'n'* — Specifies the *n*th occurrence of the result.

Return Values

Waveform. The difference in time from when the input waveform falls from the upper level to the lower level.

fft

Waveform Calculator Signal Processing function.

Uses the Fast Fourier Transform (FFT) method for calculating the Discrete Fourier Transform (DFT) used to determine the frequency content of analog or complex waveforms encountered in circuit simulation.

Usage

fft(*wf*[, *t_start*, *t_stop*, *fs*, *points, sampling*, *padding*, *normalized*, *windowType*, *alpha*, *f_ref*, *f_min*, *f_max*, *samplingEpsilon*, *windowShape*, *config_file*, *method*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is ‘Begin’.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is ‘End’.

- *fs*

  (Optional) Specifies the sampling frequency of the signal. Default is ‘Automatic’.

- *points*

  (Optional) Specifies the number of sampling points. Default is ‘Automatic’.

  For symmetric windows, the parameters above satisfy the following equation:

  ((*points*-1)/*fs*) = *t_stop* - *t_start*

  For periodic windows, the parameters above satisfy the following equation:

  ((*points*)/*fs*) = *t_stop* - *t_start*

- *sampling*

  (Optional) Specifies the method of computing the sampled data. Legal values are ‘No Sampling’ (the default), ‘Interpolation’ or ‘Spline’.

- *padding*

  (Optional) Activates data padding to pad the input data with zeros, before or after the input data set. Legal values: ‘No Padding’ (the default), ‘Padding Right’, ‘Padding Left’ or ‘Padding Left and Right’. The input parameter is verified by the algorithm and changed if necessary.

- *normalized*

  (Optional) When enabled, an average is taken on the raw data to reduce noise and smooth the frequency domain waveform. Specify 1 to turn this on, or 0 to use the raw data from the calculation unmodified. Default is 1.

- *windowType*

  (Optional) Applies a windowing function from a selection of windows. Legal values: ‘Rectangular’ (the default), ‘Hamming’, ‘Hanning’, ‘Parzen’, ‘Welch’, ‘Blackman’, ‘Blackman-Harris’, ‘Bartlett’, ‘Kaiser’, ‘Klein’ or ‘Dolph Chebyshev’.

  Note

  For Hanning, symmetric window shapes are preferred when using a Hanning window in FIR filter design.

  Periodic window shapes are preferred when using a Hanning window in spectral analysis. This is because the Discrete Fourier Transform assumes periodic extension of the input vector. A periodic Hanning window is obtained by constructing a symmetric window and removing the last sample.

- *alpha*

  (Optional) Specifies the alpha (or beta) value that is required by Hanning, Dolph Chebyshev and Kaiser windows.

  - Hanning : [0;1], default 0.5
  - Dolph Chebyshev : [0:20], default 3.0
  - Kaiser : [0:20], default 10.056.

- *f_ref*

  (Optional) Adjusts the results around the y axis so that the point for the specified frequency is 0.0. Default is ‘Automatic’.

- *f_min*

  (Optional) Specifies the starting frequency used inside the fast fourier transform result window. Default is ‘Automatic’.

- *f_max*

  (Optional) Specifies the last frequency used inside the fast fourier transform result window. Default is ‘Automatic’.

- *samplingEpsilon*

  (Optional) When *sampling* is set to ‘Interpolation’, this option specifies that the input waveform data point (Y value) is to be used rather than the exact interpolated value when the X data of the input waveform is close to the computed X value.

  The computed X value corresponds to *t_start* +(*points* * dX) where dX is the sampling interval retrieved from the *fs*. Default is 1 x 10^{-6}.

- *windowShape*

  (Optional) Specifies the shape of the window. Legal values are ‘Symmetric’ for standard FFT setup or ‘Periodic’ for enhancing FFT setup for spectral analysis of periodic signals. Defaults to the global setting in the “Waveform Calculator Calculation Options” on page 545.

  > **Tip**
  >
  > For comparing Eldo’s FFT results with EZwave’s FFT results, select the Periodic option, unless EZwave’s FFT is executed on an FFT_INPUT waveform. In this case, Eldo has already considered the periodicity of the input signal.

- *method*

  (Optional) Specifies the FFT algorithm (‘Default’ or ‘pack’). Default is ‘Default’.

Return Values

Waveform. The Fourier Transform of the input waveform.

Description

The Discrete Fourier Transform (DFT) is used to determine the frequency content of analog or complex signals encountered in circuit simulation, which deals with sequences of time values. The **fft()** function uses the Fast Fourier Transform (FFT) method for calculating the DFT.

filterdupreal

Waveform Calculator Miscellaneous function.

Removes glitches on a real (step or wreal) waveform.

Usage

filterdupreal(***wf***, *x_start*, *x_end*, *method*)

Arguments

- ***wf***

  (Required) Specifies the input waveform (step or wreal waveform only).

- *x_start*

  (Optional) Specifies the X value at the beginning of an interval. (Default is 'Begin'.)

- *x_end*

  (Optional) Specifies the X value at the end of an interval. (Default is 'End'.)

- *method*

  (Optional) Specify the filtering method, 'First' | 'Last' | 'Automatic'. Specify 'First' to keep the first 'same X' event and discard the others. Specify 'Last' to keep the last 'same X' event and discard the others. Specify 'Automatic' to automatically choose the 'same X' event to keep. (Default is 'Last'.)

Return Values

A new waveform without glitches.

Description

Removes same X value glitches on the specified real (step or wreal) waveform.

The 'Automatic' algorithm for 'same-X' events traces the line between the point before the glitch and the point after the glitch. For each 'same-X' point, EZwave calculates the distance to the traced line. The point with the minimum distance is kept, the other 'same-X' events are discarded.

filterempty

Waveform Calculator Miscellaneous function.

Removes all empty waveforms from a compound waveform.

Usage

filterempty(*wf*)

Arguments

- ***wf***

  (Required) Specifies the compound waveform to filter.

Return Values

A new compound waveform without empty elements.

first

Waveform Calculator Miscellaneous function.

Returns the first point of a waveform or a list.

Usage

first(*wf*)

Arguments

- ***wf***

  (Required) Specifies the source waveform or list.

Return Values

(Double or Complex) Returns the first point of a waveform or a list.

Related Topics

last

firstdiff

Waveform Calculator Miscellaneous function.

Finds the first time when any signal in one run begins to differ from the corresponding signal in another run, and plots *ntop* signals in sorted order.

Usage

firstdiff(*wf1*, *wf2*, [*x_start*, *x_end*, *reltol*, *abstol*, *ntop*, *depth*, *plot*]

Arguments

- ***wf1***

  (Required) Specifies any continuous waveform signal in one run.

- ***wf2***

  (Required) Specifies any continuous waveform signal in another run.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *reltol*

  (Optional) Specifies the relative tolerance. By default, this is set to 1e-2 but you can specify your own value if required.

- *abstol*

  (Optional) Specifies the absolute tolerance. By default, this is set to 1e-5 but you can specify your own value if required.

- *ntop*

  (Optional) Specifies the number of top results to return and plot. Default is 8.

- *depth*

  (Optional) Specifies the hierarchy depth used to search the waveforms. Recursive search by default.

- *plot*

  (Optional) 1 (default) plots the first spectrum difference results. 0 does not plot results.

Return Values

Array. The first difference value for the given input waveforms, in the format:

[[wf1_handle, wf2_handle, first_diff_value], [...]]

Related Topics

maxdiff

maxspectrumdiff

shiftedmaxdiff

floor

Waveform Calculator Mathematical function.

Computes the largest integral value not greater than each data point of the waveform.

Usage

floor(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

Return Values

Waveform. A waveform containing the floor values of the input waveform.

Description

Computes the largest integral value not greater than each data point of ***wf***. Upon successful completion, each returned value is expressed as a type double.

Related Topics

trunc

round

ceil

fmod

Waveform Calculator Mathematical function.

Computes the floating-point remainder of dividing ***x*** by ***y***.

Usage

fmod(***x***, ***y***)

Arguments

- ***x***

  (Required) Specifies the numerator of the input.

- ***y***

  (Required) Specifies the denominator of the input.

Return Values

Double. The floating point remainder.

Description

Computes the remainder of dividing ***x*** by ***y***. It returns ***x*** - (*n* * ***y***), where *n* is the quotient of ***x*** / ***y***, rounded to the greatest integer less than or equal to ***x*** / ***y***.

frequency

Waveform Calculator Measurement function.

Measures the frequency of a periodic waveform relative to default (automatically calculated) or user-specified topline and baseline levels.

Usage

frequency(***wf***)

frequency(***wf***, *topline*, *baseline*)

frequency(***wf***, *topline*, *baseline*, *edgetrigger*)

frequency(***wf***, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*)

frequency(***wf***, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*, *option*)

frequency(***wf***, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*, *option*, *param*)

frequency(***wf***, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*, *option*, *param*, *threshold*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the signal. Specify 'Automatic' to have this value computed.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the signal. Specify 'Automatic' to have this value computed.

- *edgetrigger*

  (Optional) Specifies the signal edge from which the measurement begins. Specify 'Rising', 'Falling' or 'Either' (default).

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' indicates numerical values, 'WF' indicates a waveform (default), and 'ANNOTATION' indicates the input waveform plot will be annotated with results.

- *param*

  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = ‘WF’. Default is ‘middleX’.

- *threshold*

  (Optional) Specifies the y value that sets the threshold of the signal. Specify ‘Automatic’ to have this value computed: mean(wf). Default is ‘None’.

  Note
  ‘topline/baseline’ and ‘threshold’ parameters cannot be used simultaneously: use either ‘topline/baseline’ or ‘threshold’.

Return Values

Waveform. The frequency of the input waveform.

frequencyjitter

Waveform Calculator Time-Domain Jitter function.

Computes the variation in time of the frequency of the source waveform relative to its average frequency or to the reference waveform (or ideal clock) frequency over part of, or the whole waveform.

Usage

frequencyjitter(*wf*[, *ref, t_start*, *t_ stop*, *edgetrigger, nbcycles, single_threshold, topline, baseline, threshold*])

Arguments

- *wf*

  (Required) Specifies the input waveform for which to calculate the jitter.

- *ref*

  (Optional) Specifies a reference period. Default is 'Automatic'.

- *t_start*

  (Optional) Specifies the X (time) value at the beginning of the jitter analysis interval. Possible values are 'Begin' or a double value. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the X (time) value at the end of the jitter analysis interval. Possible values are 'End' or a double value. Default is 'End'.

- *edgetrigger*

  (Optional) Defines the waveform edge(s) that will be taken into account during the jitter calculation. Possible values are 'Rising', 'Falling', or 'Either'. Default is 'Rising'.

- *nbcycles*

  (Optional) Specifies the maximum number of waveform cycles in the N-cycle sample, which will participate in the jitter calculation. Possible values are an integer value or 'Automatic'. Default is 'Default' (1).

- *single_threshold*

  (Optional) If 'True', the single *threshold* will be used (*topline*/*baseline* values are ignored). If 'False', *topline*/*baseline* will be used. Default is 'False'.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the source waveform. Default is 'Automatic'.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the source waveform. Default is 'Automatic'.

- *threshold*

  (Optional) Specifies the y value that sets a single threshold for the source waveform. Default is 'Automatic'.

Return Values

Waveform. The frequency jitter waveform for the input waveform(s).

Description

Frequency jitter is the variation in time of the frequency of the source waveform relative to its average frequency or to the reference waveform (or ideal clock) frequency over part of, or the whole waveform. This type of jitter helps to evaluate the short- and long-term signal stability.

Frequency jitter is calculated as the difference between the actual cycle frequency values of the source and reference waveforms. If the reference waveform is not set, the difference is calculated between the actual and average n-cycle frequency of the source waveform.

The jitter RMS value horizontal marker is displayed over the result waveform.

Related Topics

Jitter Measurement Types

frexp

Waveform Calculator Mathematical function.

Extracts the mantissa and exponent from a double-precision number.

Usage

frexp(***x***)

Arguments

- ***x***

  (Required) Specifies the floating-point number to be decomposed.

Return Values

Array. The mantissa and exponent values of the input number.

Description

Extracts the mantissa and exponent from a double-precision number by breaking the floating-point ***x*** into a normalized fraction and an integral power of 2.

Note

This function only accepts a scalar number as the input parameter.

gainmargin

Waveform Calculator Measurement function.

Calculates the gain margin in decibels (dB) of a complex waveform. The gain margin is defined as the difference between the gain of the measured waveform and 0 dB (unity gain) at the frequency where the phase shift is -180 degrees (Phase Crossover Frequency).

Usage

gainmargin(*wf*, *option*)

Arguments

- ***wf***

  (Required) Specifies the source complex waveform name.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value. ‘ANNOTATION’ plots the source waveform annotated with the results. Default is ‘VALUE’.

Return Values

The result format depends on the value of the *option* parameter.

gaussiandistribution

This function creates a Gaussian distribution.

Usage

gaussiandistribution(**wf)**

gaussiandistribution(**mean**, **stddev**)

Arguments

- **wf**

  (Required) Specifies an input waveform from which the Gaussian distribution is created, based on its mean() and stddev(). Min and max X values are mean+- N*stddev, with N the smallest positive integer so that min and max are inside this range, with the additional constraint that N is greater than or equal to 4, and smaller than or equal to 20.

  If this parameter is not specified, you must specify values for both **mean** and **stddv**.

- **mean**

  (Required) Specifies the mean. Specifying the mean along with the stddev creates a Gaussian distribution in the range of 4-sigma (mean +- 4*stddev).

- **stddev**

  (Required) Specifies the standard deviation.

Description

Creates a Gaussian distribution.

If **wf** is histogram data, the mean and stddev are calculated from the input data of the histogram, if available, not on the histogram values.

gendecade

Waveform Calculator Miscellaneous function.

Returns a list that contains numbers sweeping from ***start_value*** to ***stop_value*** with number of points per decade value equal to ***points_decade***.

Usage

gendecade(***start_value***, ***stop_value***, ***points_decade***)

Arguments

- ***start_value***

  (Required) Specifies the start value.

- ***stop_value***

  (Required) Specifies the stop value.

- ***points_decade***

  (Required) Specifies the number of points per decade.

Return Values

Array. The list of generated numbers.

genlinear

Waveform Calculator Miscellaneous function.

Returns a list that contains numbers sweeping from ***start_value*** to ***stop_value*** with a step equal to ***step_value***.

Usage

genlinear(***start_value***, ***stop_value***, ***step_value***)

Arguments

- ***start_value***

  (Required) Specifies the start value.
- ***stop_value***

  (Required) Specifies the stop value.
- ***step_value***

  (Required) Specifies the step value.

Return Values

Array. The list of generated numbers.

genoctave

Waveform Calculator Miscellaneous function.

Returns a list that contains numbers sweeping from ***start_value*** to ***stop_value*** with number of points per decade value equal to ***points_octave***.

Usage

genoctave(***start_value***, ***stop_value***, ***points_octave***)

Arguments

- ***start_value***

  (Required) Specifies the start value.

- ***stop_value***

  (Required) Specifies the stop value.

- ***points_octave***

  (Required) Specifies the number of points per octave.

Return Values

Array. The list of generated numbers.

getelementat

Waveform Calculator Miscellaneous function.

Returns the element at a given index in a bus or compound waveform.

Usage

getelementat(***wf***, ***index***)

Arguments

- ***wf***

  (Required) Specifies a bus or compound waveform.

- ***index***

  (Required) Specifies the index of the element.

Return Values

Waveform. The waveform at the given index.

Related Topics

length

getrunindices

Waveform Calculator Miscellaneous function.

Returns a list of run indices for the specified compound waveform.

Usage

getrunindices(*wf*)

Arguments

- ***wf***

  (Required) Specifies the name of the compound waveform.

Return Values

List. A list of run indices.

getrunparameters

Waveform Calculator Miscellaneous function.

Returns a list of run parameters for the specified compound waveform.

Usage

getrunparameters(*wf*)

Arguments

- ***wf***

  (Required) Specifies the name of the compound waveform.

Return Values

List. A list of run parameters.

getrunparametervalue

Waveform Calculator Miscellaneous function.

Returns the run parameter value for the specified compound bit waveform.

Usage

getrunparametervalue(***wf***, ***param***)

Arguments

- ***wf***

  (Required) Specifies the name of the compound bit waveform. This is typically the result of using getelementat() on a compound waveform.

- ***param***

  (Required) Specifies the run parameter name.

Return Values

String. The value of the parameter for this run.

gettype

Waveform Calculator Miscellaneous function.

Returns the type of the specified waveform.

Usage

gettype(*wf*)

Parameters

- ***wf***

  (Required) Specifies the name of the waveform of interest.

Return Values

Returns one of the following keywords:

- analog
- voltage
- current
- digital
- real
- complex
- analog_bus
- digital_bus
- compound
- unknown

Note

If the waveform is a compound waveform, two keywords are returned, for example:

```
complex compound
```

gmargin

Waveform Calculator Miscellaneous function.

Computes the difference between the gain of the input waveform and 0 dB (unity gain) at the frequency where the phase shift is -180 degrees (the Phase Crossover Frequency).

Usage

gmargin(***wf***)

gmargin(***wf_db***, ***wf_ph***)

Arguments

- ***wf***

  (Required) Specifies the input waveform name. If the input waveform is specified using ***wf_db*** and ***wf_ph***, this argument is not required.

- ***wf_db***

  (Required) Specifies the gain (dB) of the input waveform when used with ***wf_ph***. If the input waveform is specified with ***wf***, this argument is not required.

- ***wf_ph***

  (Required) Specifies the phase of the input waveform when used with ***wf_db***. If the input waveform is specified with ***wf***, this argument is not required.

Return Values

Double. The difference between the gain of the input waveform and 0dB.

Description

Computes the difference between the gain of the input waveform and 0 dB (unity gain) at the frequency where the phase shift is -180 degrees (the Phase Crossover Frequency). The input ***wf*** can be also be described by a gain (dB) waveform ***wf_db*** and a phase waveform ***wf_ph***. This result is in dB.

gptocomplex

Waveform Calculator Complex function.

Constructs a complex waveform from a waveform of gain in decibels and a waveform of phase in radians.

Usage

gptocomplex(***wf1***, [*x_start1*, *x_end1*,] ***wf2***[, *x_start2*, *x_end2*])

Arguments

- ***wf1***

  (Required) Specifies the first (gain) input waveform name.
- *x_start1*

  (Optional) Specifies the x value at the beginning of an interval on ***wf1***.
- *x_end1*

  (Optional) Specifies the x value at the end of an interval on ***wf1***.
- ***wf2***

  (Required) Specifies the second (phase) input waveform name.
- *x_start2*

  (Optional) Specifies the x value at the beginning of an interval on ***wf2***.
- *x_end2*

  (Optional) Specifies the x value at the end of an interval on ***wf2***.

Return Values

Waveform. A complex waveform combining the two input waveforms as gain and phase.

Description

Constructs a complex waveform from a waveform of gain in decibels (***wf1***) and a waveform of phase in radians (***wf2***).

groupdelay

Waveform Calculator Miscellaneous function.

Computes the Group Delay (GD), in seconds, from an input signal from an AC or SP analysis. GD is defined as the derivative of the phase with respect to the frequency. GD = -drv/df * (phase{input_signal}/360).

Usage

groupdelay(*wf*)

Arguments

- ***wf***

  (Required) Specifies the source AC or SP analysis waveform name.

Return Values

The group delay waveform.

halfperiodjitter

Waveform Calculator Time-Domain Jitter function.

Computes the variation in time of the half-period of a source waveform relative to its average half-period, or to a reference half-period, over part of, or the whole waveform.

Usage

halfperiodjitter(***wf***[, *ref*, *dutycycle*, *t_start*, *t_ stop*, *edgetrigger*, *single_threshold*, *topline*, *baseline*, *threshold*])

Arguments

- ***wf***

  (Required) Specifies the input waveform for which to calculate the jitter.

- *ref*

  (Optional) Specifies the reference period. Default is ‘Automatic’.

- *dutycycle*

  (Optional) Specifies the duty cycle used for the half-period calculation. When ‘Automatic’ is specified, the duty cycle is calculated for each cycle period of the waveform using the automatic topline and baseline parameters. Default value is 0.5.

- *t_start*

  (Optional) Specifies the X (time) value at the beginning of the jitter analysis interval. Possible values are ‘Begin’ or a double value. Default is ‘Begin’.

- *t_stop*

  (Optional) Specifies the X (time) value at the end of the jitter analysis interval. Possible values are ‘End’ or a double value. Default is ‘End’.

- *edgetrigger*

  (Optional) Defines the waveform edge(s) that will be taken into account during the jitter calculation. Possible values are ‘Rising’, ‘Falling’, or ‘Either’. Default is ‘Rising’.

- *single_threshold*

  (Optional) If ‘True’, the single threshold will be used (*topline*/*baseline* values are ignored). If ‘False’, *topline*/*baseline* will be used. Default is ‘False’.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the source signal. Default is ‘Automatic’.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the source signal. Default is ‘Automatic’.

- *threshold*

  (Optional) Specifies the y value that sets a single threshold for the source signal. Default is 'Automatic'.

Return Values

Waveform. The half-period jitter waveform for the input waveform(s).

Description

Calculates the variation in time of the half-period of a source waveform relative to its average half-period, or to a reference waveform (or ideal clock) half-period, over part of, or the whole waveform. This type of jitter helps to evaluate the short- and long-term signal stability.

Half-period jitter is the period multiplied by the duty cycle. It is calculated as the difference between the actual cycle-half-period values of the source and reference waveforms. If the reference period is not set, the difference is calculated between the actual and average cycle half-period of the source waveform.

The jitter RMS value horizontal marker is displayed over the result waveform.

Related Topics

Jitter Measurement Types

harm

Waveform Calculator Miscellaneous function.

Returns the voltage on a net at a specified harmonic.

Usage

harm(*wf*, *harmonic*)

Arguments

- ***wf***

  (Required) Specifies the source waveform.

- ***harmonic***

  (Required) An array of integers (comma-separated list of integers, in square brackets). Specifies the reference harmonic indexes associated with each tone. The order and the number of indexes must match the database data.

Return Values

(Complex) Returns the voltage on a net at a specified harmonic.

Examples

For a single harmonic index (single-tone analysis):

```
harm(wf("<psf/test>mix_out"), 1)
```

For multiple harmonic indexes (multi-tone analysis):

```
harm(wf("<psf/test>mix_out"), [-1, 1, 2])
```

harmonicdistortion

Waveform Calculator Signal Processing function.

Computes the harmonic distortion of the input waveform.

Usage

harmonicdistortion(***wf***[, *f_fund*, *f_min*, *f_max*, *displayTHDoption*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *f_fund*

  (Optional) Specifies the fundamental frequency. Default is 'Automatic'.

- *f_min*

  (Optional) Specifies the minimum frequency of the input waveform. Default is 'Begin'.

- *f_max*

  (Optional) Specifies the maximum frequency of the input waveform. Default is 'End'.

- *displayTHDoption*

  (Optional) Specifies the option for displaying the total harmonic distortion between 'PERCENTAGE' (the default) and 'DB'.

Return Values

Waveform. The harmonic distortion of the input waveform.

Description

Computes the harmonic distortion of the input waveform. This function is computed by using the gain of the FFT result. That means that the input waveform can be either a complex waveform or a waveform representing a gain.

f_min and *f_max* specifies the frequency band that should be taken for the computation.

The output total harmonic distortion is displayed either as a percentage or as a gain.

harmonics

Waveform Calculator Signal Processing function.

Computes the harmonic distortion of the input waveform.

Usage

harmonics(***wf***[,*f_fund*,*f_min*,*f_max*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *f_fund*

  (Optional) Specifies the fundamental frequency. Default is ‘Automatic’.

- *f_min*

  (Optional) Specifies the minimum frequency of the input waveform. Default is ‘Begin’.

- *f_max*

  (Optional) Specifies the maximum frequency of the input waveform. Default is ‘End’.

Return Values

Waveform. The harmonic distortion of the input waveform.

Description

This function is computed by using the gain of the FFT result. That means that the input waveform can be either a complex waveform or a waveform representing a gain.

harmonicsmeter

Waveform Calculator RF function.

Returns the maximum Fourier frequency value for the specified waveform.

Tip

The harmonicsmeter() function returns a scalar value but does not plot any results. You can plot results and see a detailed results report using the DNA Advisor Dialog Box accessed from the Waveform Calculator; choose **RF** from the dropdown list and click the **dnaadvisor** button.

Usage

harmonicsmeter(***wf***, *fund*, *nharm*, *reltol*, *abstol*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name. Signal must be a periodic waveform.

- *fund*

  (Optional) Specifies the fundamental frequency. Default is 'Automatic', where the entire interval is the fundamental period.

- *nharm*

  (Optional) Specifies the number of harmonics in the Fourier analysis. Default is 500.

- *reltol*

  (Optional) Specifies the relative tolerance (or dynamic range) for finding the upper Fourier frequency limit. Default is 1e-3.

- *abstol*

  (Optional) Specifies the absolute tolerance for finding the upper Fourier frequency limit. Default is 1e-3.

Return Values

Double. The maximum Fourier frequency of the input waveform.

hdist

Waveform Calculator Signal Processing function.

Computes the Total Harmonic Distortion of the input waveform.

Usage

hdist(***wf***[, *f_fund*, *f_min*, *f_max, displayTHDoption*])

Arguments

- wf

  (Required) Specifies the input waveform name.

- *f_fund*

  (Optional) Specifies the fundamental frequency. Default is ‘Automatic’.

- *f_min*

  (Optional) Specifies the minimum frequency of the input waveform. Default is ‘Begin’.

- *f_max*

  (Optional) Specifies the maximum frequency of the input waveform. Default is ‘End’.

- *displayTHDoption*

  (Optional) Specifies the option for displaying the total harmonic distortion between ‘PERCENTAGE’ (the default) and ‘DB’.

Return Values

Double. The Total Harmonic Distortion value of the input waveform

Description

This function is computed by using the gain of the FFT result. That means that the input waveform can be either a complex waveform or a waveform representing a gain.

f_min and *f_max* specifies the frequency band that should be taken for the computation.

The output total harmonic distortion is displayed either as a percentage or as a gain.

histogram

Waveform Calculator Statistical function.

Creates a histogram of the input waveform showing the magnitude probability density distribution of the waveform.

Usage

histogram(***wf***, *nb_bins, x_start, x_end, sampling, normalized*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

  The histogram function can also accept a list of values as the source. For example, you can create a histogram from the result of the thd() function, using:

  ```
  histogram(thd(wf, fund))
  ```

- *nb_bins*

  (Optional) Specifies the number of bins. Specify a plain integer or double value to set the number of bins explicitly. Specify 'scott' to use Scott's formula to define the bins. Specify 'sturge' to use Sturge's rule to define the bins. Specify 'sqrt' to use square root of the number of data points to define the bins. Default is 10.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Specify an integer or double value to set the value explicitly. Specify 'Begin' to use the start of the data. The default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. x_end. Specify an integer or double value to set the value explicitly. Specify 'End' to use the start of the data. The default is 'End'.

- *sampling*

  (Optional) Specifies whether or not to use sampling. Specify 0 or 'No Sampling' to create the histogram based on the data points of the input waveform. Specify 1 to first sample the waveform equidistantly and then create the histogram based on the sampled data. The default is 'No Sampling'.

- *normalized*

  (Optional) Specify 0 to use the raw data from the calculation unmodified. Specify 1 to divide each bin value by the total number of points. Default is 1.

Return Values

Waveform. The histogram of the input waveform.

hypot

Waveform Calculator Mathematical function.

Computes the length of the hypotenuse of a right angle triangle using the formula sqrt (x^2+y^2).

Usage

hypot(***x***, ***y***)

Arguments

- ***x***

  (Required) Specifies the x value.

- ***y***

  (Required) Specifies the y value.

Return Values

Double. The resulting value of sqrt (x^2+y^2).

Description

Euclidean distance function.

Note

This function only accepts a scalar number as the input parameter.

idb

Waveform Calculator Complex function.

Performs the inverse decibel function. It converts the input waveform from dB using the following conversion: $10^{(v/20)}$.

Usage

idb(*wf*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The converted input waveform.

idb10

Waveform Calculator Complex function.

Performs the inverse decibel function. It converts the input waveform from dB using the following conversion: $10^{(v/10)}$.

Usage

idb10(*wf*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The converted input waveform.

ifft

Waveform Calculator Signal Processing function.

Computes the inverse Fast Fourier Transform of the input waveform.

Usage

ifft(***wf***[, *f_start*, *f_stop*, *ts*, *points*, *sampling*, *padding*, *normalized, samplingEpsilon, windowShape*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *f_start*

  (Optional) Specifies the start frequency of the signal. Default is 'Begin'.

- *f_stop*

  (Optional) Specifies the stop frequency of the signal. Default is 'End'.

- *ts*

  (Optional) Specifies the sampling time of the signal. Default is 'Automatic'.

- *points*

  (Optional) Specifies the number of sampling points. Default is 'Automatic'.

  For symmetric windows, the parameters above satisfy the following equation:

  $((points)/ts) = f\_stop - f\_start$

  For periodic windows, the parameters above satisfy the following equation:

  $((points - 1)/ts) = f\_stop - f\_start$

- *sampling*

  (Optional) Specifies the method of computing the sampled data. Legal values are 'No Sampling"' (the default), 'Interpolation' or 'Spline'.

- *padding*

  (Optional) Activates data padding to pad the input data with zeros, before or after the input data set. Legal values: 'No Padding' (the default), 'Padding Right', 'Padding Left', and 'Padding Left and Right'. The input parameter is verified by the algorithm and changed if necessary.

- *normalized*

  (Optional) When enabled, an average is taken on the raw data to reduce noise and smooth the frequency domain waveform. Specify 1 to turn this on, or 0 to use the raw data from the calculation unmodified. Default is 1.

- *samplingEpsilon*

  (Optional) When *sampling* is set to 'Interpolation', specifies that the input waveform data point (Y value) is to be used rather than the exact interpolated value when the X data of the input waveform is close to the computed X value. Default is 1e-6.

  The computed X value corresponds to *t_start* +(*points* * dX) where dX is the sampling interval retrieved from the *fs*.

- *windowShape*

  (Optional) Specifies the shape of the window. Legal values are 'Symmetric' for standard FFT setup or 'Periodic' for enhancing FFT setup for spectral analysis of periodic signals. Defaults to the global setting in the "Waveform Calculator Calculation Options" on page 545.

Tip

For comparing Eldo's FFT results with EZwave's FFT results, select the Periodic option, unless EZwave's FFT is executed on an FFT_INPUT waveform. In this case, Eldo has already considered the periodicity of the input signal.

Return Values

Waveform. The inverse Fast Fourier Transform of the input waveform.

iipx

Waveform Calculator RF function.

Returns the input-referred intercept point of order x from the value of the circuit input and output levels.

Usage

iipx(***wave_in***, ***wave_out***, ***freq_1***, ***freq_2***[, *x_start*, *x_end*])

Arguments

- ***wave_in***

  (Required) Specifies the first input waveform name.

- ***wave_out***

  (Required) Specifies the second input waveform name.

- ***freq_1***

  (Required) Specifies the first input frequency.

- ***freq_2***

  (Required) Specifies the second input frequency.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double. The input-referred intercept point.

Description

Returns the input referred intercept point of order x from the value of the circuit input and output: ***wave_in*** and ***wave_out***, respectively. ***wave_in*** and ***wave_out*** must be in dB or dBm. The intercept order is directly calculated from the intermodulation of ***freq_1*** and ***freq_2***. If the input waveform type is complex, the waveforms are automatically converted in dB.

imag

Waveform Calculator Complex function.

Returns the imaginary part of the input complex waveform.

Usage

imag(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The imaginary part of the input complex waveform.

integ

Waveform Calculator Mathematical function.

Returns the definite integral value with upper and lower limits of a waveform.

Usage

integ(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double/array. The integral value of the waveform.

integnoise

Waveform Calculator RF function.

Computes the integrated noise from *f_start* to *f_stop* from a pnoise sampled(jitter) “out” (V/sqrt(Hz)) noise waveform.

Usage

integnoise(***wf***[, *f_start*, *f_stop*])

Parameters

- ***wf***

  (Required) A pnoise sampled(jitter) “out” (V/sqrt(Hz)) noise waveform. Do not select an “out_dBc” waveform.

- *f_start*

  (Optional) The starting frequency for the integration. Default is ‘Begin’.

- *f_stop*

  (Optional) The end frequency for the integration. Default is ‘End’.

Return Values

Waveform. The integrated noise from *f_start* to *f_stop*.

integral

Waveform Calculator Mathematical function.

Computes the indefinite integral (also known as the anti-derivative) of the input waveform.

Usage

integral(***wf***)

integral(***wf***, *y0*)

integral(***wf***, *y0*, *x_start*, *x_end*)

integral(***wf***, *yReal0*, *yImag0*)

integral(***wf***, *yReal0*, *yImag0*, *x_start*, *x_end*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

- *y0*

  (Optional) Specifies the integration constant. If this is not specified, the default is 0.

- *yReal0*

  (Optional) Specifies the real part of the integration constant, if wf is a complex waveform.

- yImag0

  (Optional) Specifies the imaginary part of the integration constant, if wf is a complex waveform.

Return Values

Waveform. The indefinite integral of the input waveform.

Description

Computes the indefinite integral (also known as the anti-derivative) of the input waveform. Upon successful completion, returns a waveform of the indefinite integral of the input. If no integration constant is specified, it is automatically set to 0.

intersect

Waveform Calculator Miscellaneous function.

Returns an array with all of the intersection points of two input waveforms.

Usage

intersect(***wf1***, ***wf2***)

intersect(***wf1***, ***wf2***, *with_x*)

intersect(***wf1***, ***wf2***, *x_start*, *x_end*)

intersect(***wf1***, ***wf2***, *x_start*, *x_end*, *with_x*)

intersect(***wf1***, ***wf2***, *slope_wf1*, *slope_wf2*)

intersect(***wf1***, ***wf2***, *slope_wf1*, *slope_wf2*, *x_start*, *x_end*)

intersect(***wf1***, ***wf2***, *slope_wf1*, *slope_wf2*, *x_start*, *x_end*, *with_x*)

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- ***wf2***

  (Required) Specifies the second input waveform name.

- *with_x*

  (Optional) Specifies whether the corresponding x value is returned. Use 1 or 0 as follows:

  - 1 — Both the y value and the corresponding x value are returned.
  - 0 — Only the y value is returned.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval on ***wf1***.

- *x_end*

  (Optional) Specifies the x value at the end of an interval on ***wf1***.

- *slope_wf1*

  (Optional) Specifies the slope of wf1 as follows:

  - 'neg' — Include only intersects where ***wf1*** is negative.
  - 'pos' — Include only intersects where ***wf1*** is positive.
  - 'Either' — Include all ***wf1*** intersects, regardless of slope.

- *slope_wf2*

  (Optional) Specifies the slope of ***wf2*** as follows:

  - 'neg' — Include only intersects where ***wf2*** is negative.
  - 'pos' — Include only intersects where ***wf2*** is positive.
  - 'Either' — Include all ***wf2*** intersects, regardless of slope.

Return Values

Double/array. The intersection point of the two input waveforms.

ipn

Waveform Calculator RF function.

Computes the N^{th} order input-referred intercept point, output-referred intercept point, or both for the specified input compound waveform.

Usage

ipn(***wf***, [*hb_refHarmonic*, *hb_spurHarmonic*, *pss_refIndex*, *pss_spurIndex*, *inputPower*, *sweepParam*, *nOrder*, *ePin*, *rload*, *measType*])

Arguments

- ***wf***

  (Required) Specifies the input compound waveform name.

- *hb_refHarmonic*

  (Optional) An array of integers (comma-separated list of integers, in square brackets). Specifies the reference harmonic indexes associated with each tone. The order and the number of indexes must match the database data. Used for HB analysis. Default is 'None'.

- *hb_spurHarmonic*

  (Optional) An array of integers (comma-separated list of integers, in square brackets). Specifies the spurious harmonic indexes associated with each tone. The order and the number of indexes must match the database data. Used for HB analysis. Default is 'None'.

- *pss_refIndex*

  (Optional) Specifies the index of the reference frequency. Used for PSS analysis. Default is 'None'.

- *pss_spurIndex*

  (Optional) Specifies the index of the spurious frequency. Used for PSS analysis. Default is 'None'.

- *inputPower*

  (Optional) Specifies the input power. Used for single point IPN. Default is 'None'.

- *sweepParam*

  (Optional) Specifies the sweep parameter used in the input power sweep. Typically, this is in dBm. Default is 'Automatic'.

- *nOrder*

  (Optional) Specifies the N order of the spurious frequency. Default is 3.

- *ePin*

  (Optional) Specifies the input power at which to calculate the intercept point. Default is 'Automatic', the first point in the input power sweep.

- *rload*

  (Optional) Specifies the load resistance. If specified, this is used to convert the voltage to dBm. Default is 50.

- *measType*

  (Optional) Specifies the choice of returned result, 'IIP', 'OIP' or 'Both'. Default is 'Both'.

Return Values

Double or List. The input-referred intercept point, the output-referred intercept point, or both values.

Description

Specifying Frequency Indexes for PSS

For 0Hz, the index=0. For the fundamental frequency (fund), the index=1, where fund is the frequency that is co-periodic and divides evenly into all the other frequencies.

Use the following steps to determine the ipn reference (ref) and spurious (spur) frequencies with their corresponding indexes:

1. Determine the ref frequency from the design.
2. Calculate the spur frequency in the normal manner. For example:

   spur_freq=(n-1)*f1-f2 or (n-1)*f2-f1

   where n=order (and image frequencies if needed).
3. Calculate the ipn indexes by dividing the ref and spur frequencies by fund. For example:

   If f1=2.45GHz and f2=2.65GHz then fund=50MHz, and

   pss_refIndex = 2.45GHz/50MHz = 49 and

   pss_spurIndex = 2.65GHz/50MHz = 53.

Specifying Frequency Indexes for HB

Unlike PSS, indexes are really multiples of each frequency. Then all products are summed to determine the desired reference and spurious frequencies. For example:

If f1=2.45GHz (ref, [reference harmonic]) and f2=2.65GHz (spur, [spurious harmonic]),

enter [0 1] in the *hb_spurHarmonic* field for the spur index at 2.65Hz, and

enter [1 0] in the *hb_refHarmonic* field for the ref index at 2.45GHz.

If there are more than 2 tones (freq), add each additional tone (freq) in the same way.

Examples

Example syntax:

```
ipn(wf("<testdata/test>out"), [-1, -1, 2], [-1, 1, 0], nOrder = 3, rload =
50, measType =  'Both')

ipn(wf("<testdata/test>out"), pss_refIndex = 49, pss_spurIndex = 53)
```

Example from Waveform Calculator showing results:

```
wfcalc>ipn(wf("<psf_spe/hb_fd>outdiff"), [1, 0, -1], [1, 1, -2])
(-1.1509184460706727, 23.249503016356307)
where (IIPn units dBm, OIPn units dBm)
```

Related Topics

IPnVRI

wave ipnvri

jc

Waveform Calculator RF function.

Computes cycle jitter.

Usage

jc(***wf***, *fund*, *fstart*, *fstop*, *k*, *signal*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *fund*

  (Optional) Specifies the fundamental frequency. Default is ‘Automatic’.

- *fstart*

  (Optional) Specifies the lower frequency limit in integration. Default is ‘Begin’.

- *fstop*

  (Optional) Specifies the upper frequency limit in integration. Default is ‘End’.

- *k*

  (Optional) Specifies the number of accumulated cycles. Default is ‘1’.

- *signal*

  (Optional) Specifies the RMS jitter by default, or BER for peak-to-peak jitter. Default is ‘Default’.

Return Values

Double. The cycle jitter.

Description

Computes rms or peak-to-peak k-period jitter by Sphi integration.

Related Topics

edgephasenoise

jcc

jee

jcc

Waveform Calculator RF function.

Computes cycle-to-cycle jitter.

Usage

jcc(***wf***, *fund*, *fstart*, *fstop*, *k*, *signal*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *fund*

  (Optional) Specifies the fundamental frequency. Default is 'Automatic'.

- *fstart*

  (Optional) Specifies the lower frequency limit in integration. Default is 'Begin'.

- *fstop*

  (Optional) Specifies the upper frequency limit in integration. Default is 'End'.

- *k*

  (Optional) Specifies the number of accumulated cycles. Default is '1'.

- *signal*

  (Optional) Specifies the RMS jitter by default, or BER for peak-to-peak jitter. Default is 'Default'.

Return Values

Double. The cycle to cycle jitter.

Description

Computes rms or peak-to-peak cycle-to-cycle or adjacent k-period jitter by Sphi integration.

Related Topics

edgephasenoise

jc

jee

jee

Waveform Calculator RF function.

Computes edge-to-edge jitter (jee) from pnoise sampled jitter edgephasenoise.

Usage

jee(***wf***, ***fo***, *fstart*, *fstop*)

Arguments

- ***wf***

  (Required) Specifies the source pnoise sampled jitter edgephasenoise “out” waveform name (not out_dBc).

- ***fo***

  (Required) Specifies the input frequency.

- *fstart*

  (Optional) Specifies the lower frequency limit in integration. Default is ‘Begin’.

- *fstop*

  (Optional) Specifies the upper frequency limit in integration. Default is ‘End’.

Return Values

Double. The edge to edge jitter (in seconds).

Examples

```
jee(wf("<psf/pnoise_sample>out"), 100k, 10k, 1M)
```

Related Topics

edgephasenoise

jc

jcc

join

Waveform Calculator Miscellaneous function.

Joins two non-overlapping waveforms together into a single waveform.

Usage

join(***wf1***, ***wf2***)

join(***wf1***[, *x_start1*, *x_end1*], ***wf2***[, *x_start2*, *x_end2*])

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- ***wf2***

  (Required) Specifies the second input waveform name.

- *x_start1*

  (Optional) Specifies the x value at the beginning of an interval on wf1.

- *x_end1*

  (Optional) Specifies the x value at the end of an interval on wf1.

- *x_start2*

  (Optional) Specifies the x value at the beginning of an interval on wf2.

- *x_end2*

  (Optional) Specifies the x value at the end of an interval on wf2.

Return Values

Waveform. A waveform representing the two joined waveforms.

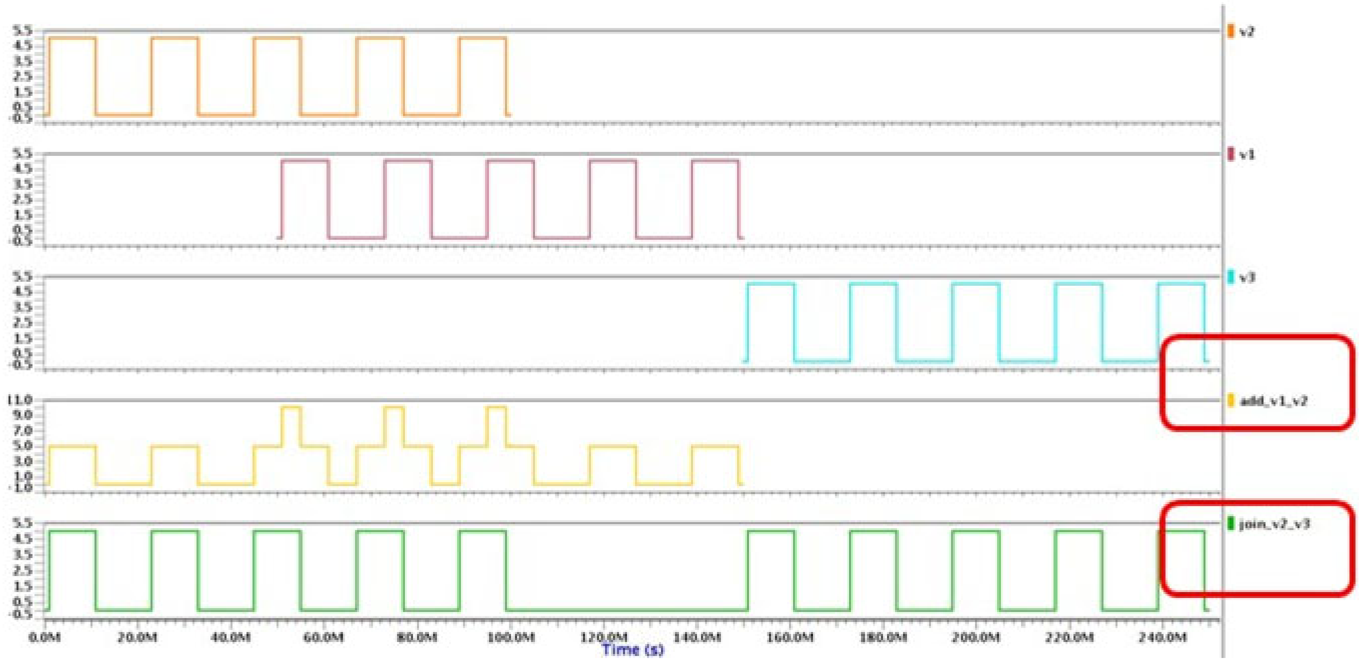
Description

Connects the last point of wf1 to the first point of wf2. No shifting in x takes place. Input waveforms must not overlap.

Examples

The following figure shows example results when using the add and join commands:

Figure A-2. Example of add() and join() Waveform Calculator Functions



Related Topics

add

larger

Waveform Calculator Statistical function.

Creates a new waveform based on the larger of two data points at any given time of the two input waveforms.

Usage

larger(***wf1***, [*x_start1*, *x_end1*,] ***wf2***[, *x_start2*, *x_end2*])

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- *x_start1*

  (Optional) Specifies the x value at the beginning of an interval on ***wf1***.

- *x_end1*

  (Optional) Specifies the x value at the end of an interval on ***wf1***.

- ***wf2***

  (Required) Specifies the second input waveform name.

- *x_start2*

  (Optional) Specifies the x value at the beginning of an interval on ***wf2***.

- *x_end2*

  (Optional) Specifies the x value at the end of an interval on ***wf2***.

Return Values

Waveform. The larger of the two input waveforms.

ldexp

Waveform Calculator Mathematical function.

Computes the load exponent of a floating-point number.

Usage

ldexp(*x*, *y*)

Arguments

- ***x***

  (Required) Specifies the x value.

- ***y***

  (Required) Specifies the y value.

Return Values

Double. The resulting value of $\boldsymbol{x} * 2^{y}$.

Description

Computes the load exponent of a floating-point number using the formula $\boldsymbol{x} * 2^{y}$. Upon successful completion, returns a double representing the value x multiplied by 2 raised to the power y.

Note

This function only accepts scalar numbers as input parameters.

last

Waveform Calculator Miscellaneous function.

Returns the last point of a waveform or a list.

Usage

last(*wf*)

Arguments

- ***wf***

  (Required) Specifies the source waveform or list.

Return Values

(Double or Complex) Returns the last point of a waveform or a list.

Related Topics

first

length

Waveform Calculator Miscellaneous function.

Returns the number of elements of a given bus or compound waveform.

Usage

length(*wf*)

Arguments

- ***wf***

  (Required) Specifies a bus or compound waveform.

Return Values

Integer. The number of elements of a given bus or compound waveform.

Related Topics

getelementat

lesser

Waveform Calculator Statistical function.

Creates a new waveform based on the lesser of two data points at any given time of the two input waveforms.

Usage

lesser(***wf1***, [*x_start1*, *x_end1*,] ***wf2***[, *x_start2*, *x_end2*])

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.
- *x_start1*

  (Optional) Specifies the x value at the beginning of an interval on ***wf1***.
- *x_end1*

  (Optional) Specifies the x value at the end of an interval on ***wf1***.
- ***wf2***

  (Required) Specifies the second input waveform name.
- *x_start2*

  (Optional) Specifies the x value at the beginning of an interval on ***wf2***.
- *x_end2*

  (Optional) Specifies the x value at the end of an interval on ***wf2***.

Return Values

Waveform. The lesser of the two input waveforms.

ln

Waveform Calculator Mathematical function.

Computes the natural logarithm of the input argument waveform.

Usage

ln(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.
- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.
- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The scaled input waveform.

Description

Computes the natural logarithm of the input argument ***wf***. The value of ***wf*** must be positive.

localmax

Waveform Calculator Measurement function.

Finds the local maxima of a waveform.

Usage

localmax(***wf***)

localmax(***wf***, *x_start*, *x_end*)

localmax(***wf***, *x_start*, *x_end*, *option*)

localmax(***wf***, *x_start*, *x_end*, *option*, *param*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ indicates numerical values, ‘WF’ indicates a waveform (default), and ‘ANNOTATION’ indicates the input waveform plot will be annotated with results.

- *param*

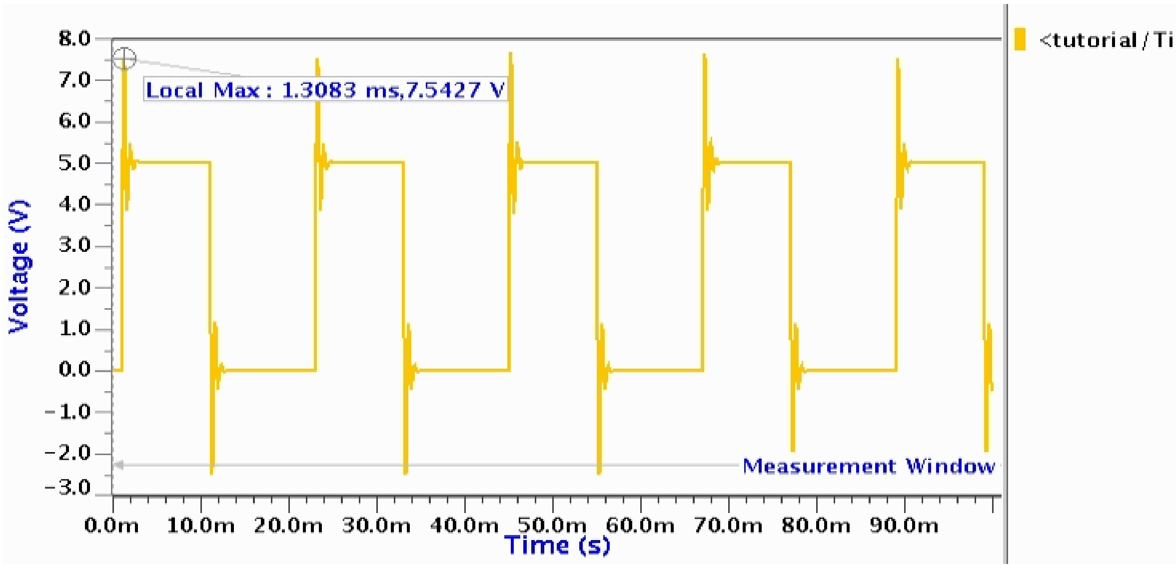
  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = ‘WF’, either ‘resultX’ or ‘Xunit’. Default is ‘resultX’.

Return Values

Waveform. The local maxima of a waveform. The default output type is a waveform, but this can be changed using the *option* argument.

Examples

```
localmax(wf("<tutorial/Time-Domain_Results>v_load"), x_start=0,
   x_end='END', option='ANNOTATION')
```



Related Topics

localmin

localmin

Waveform Calculator Measurement function.

Finds the local minima of a waveform.

Usage

localmin(***wf***)

localmin(***wf***, *x_start*, *x_end*)

localmin(***wf***, *x_start*, *x_end*, *option*)

localmin(***wf***, *x_start*, *x_end*, *option*, *param*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' indicates numerical values, 'WF' indicates a waveform (default), and 'ANNOTATION' indicates the input waveform plot will be annotated with results.

- *param*

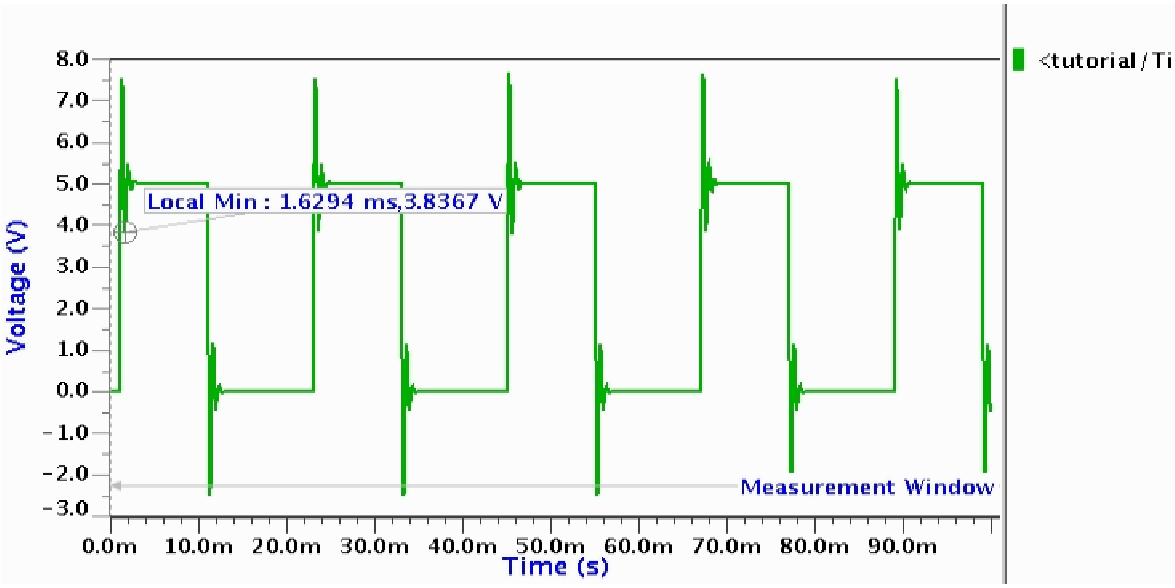
  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = 'WF', either 'resultX' or 'Xunit'. Default is 'resultX'.

Return Values

Waveform. The local minima of a waveform. The default output type is a waveform, but this can be changed using the *option* argument.

Examples

```
localmin(wf("<tutorial/Time-Domain_Results>v_load"), x_start=0,
   x_end='END', option='ANNOTATION')
```



Related Topics

localmax

log

Waveform Calculator Mathematical function.

Computes the base 10 logarithm of the input waveform.

Usage

log(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The scaled input waveform.

Description

Computes the base 10 logarithm of the input ***wf***. The value of ***wf*** must be positive.

longtermjitter

Waveform Calculator Phase Noise function.

Returns the long term jitter waveform.

Usage

longtermjitter(***wf***, ***f0***[, *x_start*, *x_end*, *sampling_nb_points*])

Arguments

- ***wf***

  (Required) Specifies the name on the input waveform on which the long term jitter is calculated.

- ***f0***

  (Required) Specifies the Fundamental Frequency of SST Noise Analysis. Default value is 'Automatic'.

  Note

  If ***f0*** is stored in the database by the Eldo RF simulator, this argument becomes optional.

- *x_start*

  (Optional) Specifies the x value at the beginning of the interval for the long term jitter calculation. Default value is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of the interval for the long term jitter calculation. Default value is 'End'.

- *sampling_nb_points*

  (Optional) Defines whether sampling is applied to the source ce waveform and specifies the number of sampling points. Default value is 'Automatic', for which 100 sampling points are considered.

Return Values

Waveform. The long term jitter waveform.

Description

Calculates and returns the long term jitter waveform as follows:

$$\sigma_t^{\ 2} = \frac{8}{\omega_0^{\ 2}} \int_0^{+\infty} S_\phi(f).\sin^2(\pi f t).df$$

σ_t corresponds to the waveform representing the long term jitter for forced circuits.

Note

The calculation is designed for input waveforms SPHI, DB(SPHI), SPHI_SSB, DB(SPHI_SSB), Lf and DB(PHNOISE), but not restricted to this list.

It is recommended that the waveform is sampled around 100 points (ten points per decade) prior to running the long term jitter calculation to ensure that the calculation does not take too long to run.

Examples

```
longtermjitter(wf1)
# This will calculate the long term jitter for waveform wf1 with an
# automatically detected fundamental frequency that has been stored in
# the database by the Eldo RF simulator. The calculation will use
# the entire waveform. There will be 100 sampling points - the default.

longtermjitter(wf1, 4e6, 3e6, 5e6, 50)
# This will calculate the long term jitter for waveform wf1
# with fundamental frequency 4.000.000 Hz. The calculation will use
# the waveform from 3.000.000 Hz to 5.000.000 Hz. There will be 50
# sampling points.

longtermjitter(wf1, f0=6e6, sampling_nb_points=200)
# This will calculate the long term jitter for waveform wf1 with
# fundamental frequency 6.000.000 Hz and 200 sampling points. The entire
# waveform will be used for the calculation.
```

Related Topics

Jitter Measurement Types

mag

Waveform Calculator Complex function.

Returns the absolute magnitude of the input complex waveform.

Usage

mag(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. Returns the absolute magnitude of the input complex waveform.

max

Waveform Calculator Statistical function.

This function returns the maximum value of a waveform.

Usage

max(***wf***)

max(***wf***, *x_start*, *x_end*, x*_value*)

max(***wf***, *x_start*, *x_end*, *x_value*, *option*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

- *x_value*

  (Optional) Specifies whether to return the x value along with the y value. Specify 0 to return only the maximum y value; specify 1 to return both the maximum y and the corresponding x value at the maximum. Default is ‘no’ (0).

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the result. ‘ANNOTATION’ plots the source waveform, annotated with the result value. Default is ‘VALUE’.

Return Values

Double. The maximum value of the input waveform.

Description

If the input waveform is complex, it returns the largest magnitude of its elements.

maxdiff

Waveform Calculator Miscellaneous function.

Finds the time or the frequency when any signal in one run differs the most from the corresponding signal in another run, and plots *ntop* signals in sorted order.

Usage

maxdiff(*wf1*, *wf2*, [*func*, *x_start*, *x_end*, *reltol*, *abstol*, *ntop*, *depth*, *plot*]

Arguments

- ***wf1***

  (Required) Specifies any continuous waveform signal in one run. Used to identify a hierarchy path in a source folder, plus sub-hierarchies with respect to the *depth* parameter.

- ***wf2***

  (Required) Specifies any continuous waveform signal in another run. Used to identify a test folder only. The name and hierarchy path are the same as those of source folder.

- *func*

  (Optional) Specifies the waveform function. Specify one of:

  - default (check value)
  - slope (check slope)

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *reltol*

  (Optional) Specifies the relative tolerance. By default, this is set to 1e-3 but you can specify your own value if required.

- *abstol*

  (Optional) Specifies the absolute tolerance. By default, this is set to 1e-6 but you can specify your own value if required.

- *ntop*

  (Optional) Specifies the number of top results to return and plot. Default is 8.

- *depth*

  (Optional) Specifies the hierarchy depth used to search the waveforms. Recursive search by default.

- *plot*

  (Optional) 1 (default) plots the maximum difference results. 0 does not plot results.

Return Values

Array. The maximum difference value for the given input waveforms, in the format:

[[wf1_handle, wf2_handle, max_diff_value], [...]]

Examples

If you specify:

```
maxdiff(wf("<db1/folder1>path/to/wf"), wf("<db2/folder2>another/path/to/
wf")),
```

then **maxdiff** is computed on pair of waveforms having the names:

```
{ wf("<db1/folder1>path/to/something"), wf("<db2/folder2>path/to/
something") }
```

Related Topics

firstdiff

maxspectrumdiff

shiftedmaxdiff

maxspectrumdiff

Waveform Calculator Miscellaneous function.

Finds the frequency when any signal in one run differs the most from the corresponding signal in another run, after *phase* shift, and plots *ntop* signals in sorted order.

Usage

maxspectrumdiff(***wf1***, ***wf2***, [*phase*, *f_start*, *f_end*, *reltol*, *abstol*, *ntop*, *depth*, *plot*]

Arguments

- ***wf1***

  (Required) Specifies any pss spectrum signal with large fundamental amplitude, in one run.

- ***wf2***

  (Required) Specifies the same pss spectrum signal, in another run.

- *phase*

  (Optional) Default is ‘Automatic’, which phase shifts the fundamental of the second specified signal to match that of the first. Specify 0 for no phase shift.

- *f_start*

  (Optional) Specifies the f value at the beginning of an interval. Default is ‘Begin’.

- *f_end*

  (Optional) Specifies the f value at the end of an interval. Default is ‘End’.

- *reltol*

  (Optional) Specifies the relative tolerance. By default, this is set to 1e-3 but you can specify your own value if required.

- *abstol*

  (Optional) Specifies the absolute tolerance. By default, this is set to 1e-6 but you can specify your own value if required.

- *ntop*

  (Optional) Specifies the number of top results to return and plot. Default is 8.

- *depth*

  (Optional) Specifies the hierarchy depth used to search the waveforms. Recursive search by default.

- *plot*

  (Optional) 1 (default). Plots the maximum spectrum difference results. 0 does not plot results.

Return Values

Array. The maximum spectrum difference value for the given input waveforms, in the format:

[[spectrum_wf1_handle, spectrum_wf2_handle, max_spectrum_diff_value], [...]]

Related Topics

firstdiff

maxdiff

shiftedmaxdiff

mean

Waveform Calculator Measurement function.

Calculates the mean value of a waveform.

Usage

mean(***wf***, *x_start*, *x_end*, *option*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *x_start*

  (Optional) Specifies the start x value for the calculation. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the stop x value for the calculation. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the mean. ‘ANNOTATION’ plots the source waveform, annotated with the mean value. Default is ‘VALUE’.

Return Values

The result format depends on the value of the *option* parameter.

meanminus3std

Waveform Calculator Measurement function.

Calculates the mean minus 3 standard deviation value of a waveform.

Usage

meanminus3std(***wf***, *x_start*, *x_end*, *option*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *x_start*

  (Optional) Specifies the start x value for the calculation. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the stop x value for the calculation. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' returns a numerical value of the mean minus 3 standard deviation. 'ANNOTATION' plots the source waveform, annotated with the mean minus 3 standard deviation value. Default is 'VALUE'.

Return Values

The result format depends on the value of the *option* parameter.

meanplus3std

Waveform Calculator Measurement function.

Calculates the mean plus 3 standard deviation value of a waveform.

Usage

meanplus3std(***wf***, *x_start*, *x_end*, *option*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *x_start*

  (Optional) Specifies the start x value for the calculation. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the stop x value for the calculation. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' returns a numerical value of the mean plus 3 standard deviation. 'ANNOTATION' plots the source waveform, annotated with the mean plus 3 standard deviation value. Default is 'VALUE'.

Return Values

The result format depends on the value of the *option* parameter.

min

Waveform Calculator Statistical function.

This function returns the minimum value of a waveform.

Usage

min(***wf***)

min(***wf***, *x_start*, *x_end*, *x_value*)

min(***wf***, *x_start*, *x_end*, *x_value*, *option*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

- *x_value*

  (Optional) Specifies whether to return the x value along with the y value. Specify 0 to return only the minimum y value; specify 1 to return both the minimum y and the corresponding x value at the minimum. Default is ‘no’ (0).

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the result. ‘ANNOTATION’ plots the source waveform, annotated with the result value. Default is ‘VALUE’.

Return Values

Double. The minimum value of the input waveform.

Description

If the input waveform is complex, it returns the smallest magnitude of its elements.

modf

Waveform Calculator Mathematical function.

Breaks a floating-point number into integral and fractional parts.

Usage

modf(***x***)

Arguments

- ***x***

  (Required) Specifies the floating-point number to be decomposed.

Return Values

Array. The decomposed input number.

Description

Breaks the argument x into integral and fractional parts, each having the same sign as the argument. The integral part is returned as a type double.

Note

This function only accepts a scalar number as input parameter.

mptocomplex

Waveform Calculator Complex function.

Constructs a complex waveform from a waveform of magnitude and a waveform of phase.

Usage

mptocomplex(***wf1***, [*x_start1*, *x_end1*,] ***wf2***[, *x_start2*, *x_end2*])

Arguments

- ***wf1***

  (Required) Specifies the first (magnitude) input waveform name.
- *x_start1*

  (Optional) Specifies the x value at the beginning of an interval on wf1.
- *x_end1*

  (Optional) Specifies the x value at the end of an interval on wf1.
- ***wf2***

  (Required) Specifies the second (phase) input waveform name.
- *x_start2*

  (Optional) Specifies the x value at the beginning of an interval on wf2.
- *x_end2*

  (Optional) Specifies the x value at the end of an interval on wf2.

Return Values

Waveform. A complex waveform combining the two input waveforms as magnitude and phase.

Description

Constructs a complex waveform from a waveform of magnitude (***wf1***) and a waveform of phase in radians (***wf2***).

nand

Waveform Calculator Logic function.

Applies the NAND function to two input waveforms whose data types are either bit or boolean.

Usage

nand(***wf1***, ***wf2***)

Arguments

- ***wf1***

  (Required) Specifies the first input digital waveform name.

- ***wf2***

  (Required) Specifies the second input digital waveform name.

Return Values

Waveform. The resulting NAND waveform between the two input waveforms.

Description

Applies the NAND function to two input waveforms whose data types are either bit or boolean. The results of the function are described in the following table:

Table A-13. NAND Truth Table

wf1	*wf2*	NAND (*wf1*, *wf2*)
F	F	T
F	T	T
T	F	T
T	T	F

In the table, T represents TRUE for boolean waveforms, and 1 for bit waveforms. F represents FALSE for boolean waveforms, and 0 for bit waveforms.

noisetosignaldbc

Waveform Calculator RF function.

Calculates the ratio of noise to signal amplitude, in dBc/Hz.

Usage

noisetosignaldbc(***noise***, ***amplitude***)

Arguments

- ***noise***

  Noise voltage or current per sqrt(Hz) waveform.

- ***amplitude***

  Amplitude of sinusoidal signal.

Return Values

Waveform. The ratio of noise to signal for the input waveform.

Examples

In the waveform calculator, evaluate:

```
noise=psd(wf("<tutorial/Time-Domain_Results>v_load"))
```

and then:

```
noisetosignaldbc(noise,5.0)
```

noisetrantophasenoise

Waveform Calculator RF function.

Computes the Phase Noise Spectrum (Power Spectral Density) of a periodic (noisy) transient waveform.

Usage

noisetrantophasenoise(***wf***[, *t_start*, *t_stop, f_ref*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. The waveform must be the result of a transient noise analysis. All waveforms in the NOISETRAN directory that are not RMS waveforms may be used.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is 'End'.

- *f_ref*

  (Optional) Specifies the fundamental frequency. If not specified, this is computed automatically from the average period. Default is 'Automatic'.

Return Values

Waveform. The Phase Noise Spectrum of the input waveform.

Description

To provide smoother results, the function iterates several times using small intervals of the input waveform. The interval size is 10% of the *t_start* to *t_stop* range, or larger when several periods are available.

First, this function computes the average period (or average carrier frequency), then it computes the timing and phase jitter with respect to this average period. When the fundamental frequency is provided, the average period is calculated as 1/*f_ref*. Finally, the Phase Noise Spectrum is calculated from the Power Spectral Density of the extracted jitter.

Tip

See also .NOISETRAN in the *Eldo Reference Manual.*

nor

Waveform Calculator Logic function.

Applies the NOR function to two input waveforms whose data types are either bit or boolean.

Usage

nor(***wf1***, ***wf2***)

Arguments

- ***wf1***

  (Required) Specifies the first input digital waveform name.

- ***wf2***

  (Required) Specifies the second input digital waveform name.

Return Values

Waveform. The resulting NOR waveform between the two input waveforms.

Description

Applies the NOR function to two input waveforms whose data types are either bit or boolean. The results of the function are described in the following table:

Table A-14. NOR Truth Table

wf1	***wf2***	**nor (*wf1*, *wf2*)**
F	F	T
F	T	F
T	F	F
T	T	F

In the table, T represents TRUE for boolean waveforms, and 1 for bit waveforms. F represents FALSE for boolean waveforms, and 0 for bit waveforms.

oipx

Waveform Calculator RF function.

Returns the output-referred intercept point of order x from the value of the circuit output waveform.

Usage

oipx(***wf***, ***freq_1***, ***freq_2***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***freq_1***

  (Required) Specifies the first input frequency.

- ***freq_2***

  (Required) Specifies the second input frequency.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double. The output-referred intercept point.

Description

Returns the output-referred intercept point of order x from the value of the circuit output waveform, which must be in dB or dBm. The intercept order is directly calculated from the intermodulation of ***freq_1*** and ***freq_2***. If the input waveform type is complex, the waveform is automatically converted in dB.

overshoot

Waveform Calculator Measurement function.

Calculates the overshoot value of a waveform. The overshoot value is calculated as the difference between the maximum point and the topline level of the waveform.

Usage

overshoot(***wf***, *topline*, *baseline*, *x_start*, *x_end*, *option*, *param*, *overshoot*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold for the source waveform. Specify 'Automatic' to have this value computed. Default is 'Automatic'.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold for the source waveform. Specify 'Automatic' to have this value computed. Default is 'Automatic'.

- *x_start*

  (Optional) Specifies the start x value for the calculation. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the stop x value for the calculation. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' returns a numerical value of the overshoot. 'WF' returns the result waveform. 'ANNOTATION' plots the source waveform, annotated with the overshoot. Default is 'WF'.

- *param*

  (Optional) Specifies the simulation parameter to be used to generate the result waveform for *option* 'WF'. Default is 'overshootX'.

- *overshoot*

  (Optional) Specifies the returned result. Choose 'all', 'first', 'last' or an integer. Default is 'all'.

Return Values

The result format depends on the value of the *option* parameter.

peaktopeak

Waveform Calculator Measurement function.

Calculates the peak-to-peak value of a waveform.

Usage

peaktopeak(***wf***, *x_start*, *x_end*, *option*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *x_start*

  (Optional) Specifies the start x value for the calculation. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the stop x value for the calculation. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' returns a numerical value of the settle time. 'ANNOTATION' plots the source waveform, annotated with the settle time. Default is 'VALUE'.

Return Values

The result format depends on the value of the *option* parameter.

period

Waveform Calculator Measurement function.

Measures the period of a periodic waveform relative to default (automatically calculated) or user-specified topline and baseline levels.

Usage

period(***wf***)

period(***wf***, *topline*, *baseline*)

period(***wf***, *topline*, *baseline*, *edgetrigger*)

period(***wf***, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*)

period(***wf***, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*, *option*)

period(***wf***, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*, *option*, *param*)

period(***wf***, *topline*, *baseline*, *edgetrigger*, *x_start*, *x_end*, *option*, *param*, *threshold*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the signal. Specify 'Automatic' to have this value computed.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the signal. Specify 'Automatic' to have this value computed.

- *edgetrigger*

  (Optional) Specifies the signal edge from which the measurement begins. Specify 'Rising', 'Falling' or 'Either' (default).

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' indicates numerical values, 'WF' indicates a waveform (default), and 'ANNOTATION' indicates the input waveform plot will be annotated with results.

- *param*

  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = ‘WF’. Default is ‘middleX’.

- *threshold*

  (Optional) Specifies the y value that sets the threshold of the signal. Specify Automatic to have this value computed: mean(wf). Default is ‘None’.

  Note
  ‘topline/baseline’ and ‘threshold’ parameters cannot be used simultaneously: use either ‘topline/baseline’ or ‘threshold’.

Return Values

Waveform. The period of the input waveform.

periodjitter

Waveform Calculator Phase Noise function.

Returns the period jitter value. Period jitter measures the deviation of the period from the average period.

Usage

periodjitter(***wf***, ***f0***[, *x_start*, *x_end*, *sampling_nb_points*, *nbcycles*])

Arguments

- ***wf***

  (Required) Specifies the name on the input waveform on which the period jitter is calculated.

- ***f0***

  (Required) Specifies the fundamental frequency of SST noise analysis. Default value is 'Automatic'.

  Note

  If ***f0*** is stored in the database by the Eldo RF simulator, this argument becomes optional.

- *x_start*

  (Optional) Specifies the x value at the beginning of the interval for the period jitter calculation. Default value is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of the interval for the period jitter calculation. Default value is 'End'.

- *sampling_nb_points*

  (Optional) Defines whether sampling is applied to the source ce waveform and specifies the number of sampling points. Default value is 'Automatic', for which 100 sampling points are considered.

- *nbcycles*

  (Optional) (Optional) Specifies the maximum number of waveform cycles which will participate in the jitter calculation. Possible value is an integer value. Default is 'Default', corresponding to 1 cycle.

Return Values

Double. The period jitter value of the input waveform.

Description

Period jitter is calculated as follows:

$$\sigma_{T_0} = \sqrt{\frac{8}{\omega_0^{\ 2}}\int_0^{+\infty} S_\phi(f).\sin^2(\pi f T_0).df}$$

σ_{T_0} corresponds to the period jitter value for forced circuits.

Note

The calculation is designed for input waveforms SPHI, DB(SPHI), SPHI_SSB, DB(SPHI_SSB), Lf and DB(PHNOISE), but not restricted to this list.

Examples

```
periodjitter(wf1)
# This will calculate the period jitter for waveform wf1 with an
# automatically detected fundamental frequency that has been stored in
# the database by the Eldo RF simulator. The calculation will use
# the entire waveform. There will be 100 sampling points - the default.

periodjitter(wf1, 4e6, 3e6, 5e6, 50)
# This will calculate the period jitter for waveform wf1
# with fundamental frequency 4.000.000 Hz. The calculation will use
# the waveform from 3.000.000 Hz to 5.000.000 Hz. There will be 50
# sampling points.

periodjitter(wf1, f0=6e6, sampling_nb_points=200)
# This will calculate the period jitter for waveform wf1 with
# fundamental frequency 6.000.000 Hz and 200 sampling points. The entire
# waveform will be used for the calculation.
```

Related Topics

Jitter Measurement Types

periodjitterbyintegration

Waveform Calculator Phase Noise function.

Calculates the RMS k-period jitter by Sphi integration.

Usage

periodjitterbyintegration(***wf***, ***fund***[, *fstart*, *fstop*, *k*])

Arguments

- ***wf***

  (Required) Specifies the two-sided Sphi density waveform (in dB rad^2/Hz vs frequency in Hz) as computed using, for example, sphibyjitter. Only the positive side of Sphi is used.

- ***fund***

  (Required) Specifies the fundamental frequency.

- *fstart*

  (Optional) Specifies the lower frequency limit in integration. Default is ‘Begin’.

- *fstop*

  (Optional) Specifies the upper frequency limit in integration. Default is ‘End’.

- *k*

  (Optional) Specifies the number of accumulated cycles. Default is 1.

Return Values

Double. The RMS k-period jitter by Sphi integration of the input waveform.

phase

Waveform Calculator Complex function.

This function returns the phase of the input complex waveform.

Usage

phase(*wf*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The phase of the input complex waveform.

Description

Returns the phase of the input complex waveform limited to [-pi, pi] in radians, or limited to [-180, 180] in degrees, according to the option set on the Data Format Options page of the EZwave Display Preferences Dialog Box.

phasemargin

Waveform Calculator Measurement function.

Calculates the phase margin of a complex waveform (in degrees or radians). The phase margin is defined as the difference in phase between the measured waveform and -180 degrees at the point corresponding to the frequency that gives a gain of 0 dB (the Gain Crossover Frequency).

Usage

phasemargin(*wf*, *option*)

Arguments

- ***wf***

  (Required) Specifies the source complex waveform name.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value. ‘ANNOTATION’ plots the source waveform annotated with the results. Default is ‘VALUE’.

Return Values

The result format depends on the value of the *option* parameter.

phasenoise

Waveform Calculator RF function.

Computes the phase noise of a transient analysis.

Usage

phasenoise(***wf1***, ***wf2***[, *t_0*, *f_min*, *f_max*])

Arguments

- ***wf1***

  (Required) Specifies the name of a waveform that is the result of a transient analysis of an oscillator.

- ***wf2***

  (Required) Specifies the name of a waveform that contains RMS values of the noise corresponding to wf1.

- *t_0*

  (Optional) Specifies the start time. Default is 'Begin' (the first x value of the transient signal).

- *f_min*

  (Optional) Specifies the frequency minimum. Default is 'Begin'.

- *f_max*

  (Optional) Specifies the frequency maximum. If not specified, the default value is the derivative signal divided by two. Default is 'End'.

Return Values

Waveform. The phase noise of the input transient analysis.

Description

The first waveform (***wf1***) must be the result of a transient analysis of an oscillator, and the second waveform (***wf2***) must be RMS values of the noise corresponding to the first waveform.

The output units are as follows:

- Oscillation fundamental frequency is displayed in Hz.
- Jitter noise variance slope is displayed in sec^2 * Hz.
- Phase noise variance slope is displayed in rad^2 * Hz.

phasenoiseoutdbc

Waveform Calculator Phase Noise function.

Returns a phase noise dBc/Hz waveform from a V/sqrtHz output and determines dBc scalar values at two selected frequency points.

Restriction

This function is not suitable for edge phase noise calculation from pnoise sampled (jitter).

Note

1. AFS "out" has the Lorentzian corner (roll-off corner) in AFS version 2018.2 and later. AFS also has a default out_dBc output which is non-Lorentzian and can be plotted directly.

2. Spectre "out" does not have the Lorentzian corner unless the Lorentzian option is set (as it is not the default).

Usage

phasenoiseoutdbc(*pss_out*, *phase_out*[, *freq1, freq2, relharm*])

Arguments

- ***pss_out***

  (Required) Specifies the pss_fd large signal output waveform which is needed to determine the magnitude.

- ***pnoise_out***

  (Required) Specifies the pnoise output waveform.

- *freq1*

  (Optional) Specifies the frequency at which a dBc value is returned. Default is '100k'.

- *freq2*

  (Optional) Specifies the frequency at which a dBc value is returned. Default is '1M'.

- *relharm*

  (Optional) Specifies the relative harmonic number. Default is '1', the fundamental frequency.

Return Values

Waveform. A phase noise waveform in dBc/Hz. Also returns dBc scalar values at two selected frequency points.

phnoisebydlm

Waveform Calculator Phase Noise function.

Computes the phase noise in dBc/Hz by delay line measurement, where the oscillation waveform is delayed and mixed with itself. This measurement is useful with circuits like free-running oscillators, which have a fundamental frequency that is stable and varies randomly at the specified value. Harmonic distortion of the waveform reduces the accuracy of this measurement.

Usage

phnoisebydlm(***wf***, ***fund***[, *t_start, t_stop, f_sample, rbw, n_delay*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. The waveform must be the result of a transient noise analysis.

- **fund**

  (Required) Specifies the fundamental or center frequency.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is 'End'.

- *f_sample*

  (Optional) Specifies the sampling frequency. Default value 'Automatic'is the inverse of the maximum dx of the waveform.

- *rbw*

  (Optional) Specifies the frequency resolution bandwidth. Finer RBW increases the variability in the measured phase noise, and reduces the lower offset frequency limit. Default value is 10/(t_stop-t_start).

- *n_delay*

  (Optional) Specifies the number of delay periods. Recommended values are odd quarters: 1/4 3/4 5/4 7/4 9/4. Default value 'Automatic' is 1/4.

Return Values

Waveform. The phase noise by delay line measurement computation.

Examples

```
phnoisebydlm(v0, 3.4G)
```

Tip

See also .NOISETRAN in the *Eldo Reference Manual*.

phnoisebymixer

Waveform Calculator Phase Noise function.

Computes the phase noise in dBc/Hz by mixer measurement, where the oscillation waveform is mixed with a pure tone. This measurement is useful for circuits like PLLs, for which the fundamental frequency is locked.

Usage

phnoisebymixer(***wf***, ***fund***[, *t_start, t_stop, f_sample, rbw*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. The waveform must be the result of a transient noise analysis.

- ***fund***

  (Required) Specifies the fundamental or center frequency.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is 'End'.

- *f_sample*

  (Optional) Specifies the sampling frequency. Default value 'Automatic' is the inverse of the maximum dx of the waveform.

- *rbw*

  (Optional) Specifies the frequency resolution bandwidth. Finer RBW increases the variability in the measured phase noise, and reduces the lower offset frequency limit. Default value 'Automatic' is 10/(t_stop-t_start).

Return Values

Waveform. The phase noise by mixer computation.

Examples

```
phnoisebymixer(v0, 3.4G)
```

Tip

See also .NOISETRAN in the *Eldo Reference Manual.*

phmargin

Waveform Calculator Miscellaneous function.

Computes the difference in phase between the input waveform wf and -180 degrees at the frequency where the gain is 0 dB (the Gain Crossover Frequency).

Usage

phmargin(***wf***)

phmargin(***wf_db***, ***wf_ph***)

Arguments

- ***wf***

  (Required) Specifies the input waveform name. If the input waveform is specified using ***wf_db*** and ***wf_ph***, this argument is not required.

- ***wf_db***

  (Required) Specifies the gain (dB) of the input waveform when used with ***wf_ph***. If the input waveform is specified with ***wf***, this argument is not required.

- ***wf_ph***

  (Required) Specifies the phase of the input waveform when used with ***wf_db***. If the input waveform is specified with ***wf***, this argument is not required.

Return Values

Double. The difference in phase.

Description

The input waveform can be also be described by a gain (dB) waveform ***wf_db*** and a phase waveform ***wf_ph***. This result is either in radians or in degrees according to data format settings.

pivot

Waveform Calculator Miscellaneous function.

Generates a pivot waveform using the input compound waveform as the y axis and a second waveform, or sweep parameter, as the x axis.

Usage

pivot(*wf*,*x_wf_or_param*[,*filter*])

Arguments

- ***wf***

  (Required) Specifies the input compound waveform name (y axis).

- ***wf_or_param***

  (Required) Specifies the waveform (or compound waveform) to be used for the x axis, *or* the sweep parameter name.

- *filter*

  (Optional) Specifies the input array of filtering parameters. Specify sweep parameters and their values to be excluded. Default is 'None'.

Return Values

Waveform. The pivoted compound waveform.

Description

If a filter array is specified, it must contain filter values wrapped between square brackets ([ ]). The first element defines the parameter name and the subsequent elements define the *excluded* values. Elements are separated by a comma. For example, [[*SWEEP0*, *val0*, *val1*,...,*valn*], …, [*SWEEPn* , *val0*, *val1*,...,*valn*]].

plotjitterconfidenceinterval

Waveform Calculator Time-Domain Jitter function.

Plots approximate confidence interval for rms jitter vs. number of Gaussian samples (N).

Usage

plotjitterconfidenceinterval(*rms_jitter*, *nb_samples*[, *confidence_level*])

Arguments

- ***rms_jitter***

  (Required) Sample rms jitter as calculated using the standard deviation. Default is 'Automatic'.

- ***nb_samples***

  (Required) Number of independent Gaussian data samples. This may be a single value or sequence of values (list of values). The function **genlinear**(***start, end, step***) may be used.

  For example:

  [1.0, 10.0, 50.0) or

  genlinear(30.0, 1000.0, 1.0)

  Default is 'Automatic'.

- *confidence_level*

  (Optional) Specifies the confidence level. Default is 'Automatic' (99%).

Return Values

Waveform. The approximate confidence interval.

Related Topics

Jitter Measurement Types

pow10

Waveform Calculator Mathematical function.

Computes the value of 10^{wf}.

Usage

pow10(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

Return Values

Waveform. The scaled input waveform.

psd

Waveform Calculator Signal Processing function.

Computes the Power Spectral Density of the input waveform.

Usage

psd(*wf*[, *t_start*, *t_stop*, *fs*, *points*, *sampling*, *padding*, *normalized*, *windowType*, *alpha*, *nsect*, *nauto*, *ncorr*, *npsd*, *computationMethod*, *f_ref*, *f_min*, *f_max*, *samplingEpsilon*, *windowShape*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is ‘Begin’.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is ‘End’.

- *fs*

  (Optional) Specifies the sampling frequency of the signal. Default is ‘Automatic’.

- *points*

  (Optional) Specifies the number of sampling points. Default is ‘Automatic’.

  For symmetric windows, the parameters above satisfy the following equation:

  $((points)/fs) = t\_stop - t\_start$

  For periodic windows, the parameters above satisfy the following equation:

  $((points - 1)/fs) = t\_stop - t\_start$

- *sampling*

  (Optional) Specifies the method of computing the sampled data. Legal values are ‘No Sampling’ (the default) or ‘Interpolation’.

- *padding*

  (Optional) Activates data padding to pad the input data with zeros, before or after the input data set. Legal values: ‘No Padding’ (the default), ‘Padding Right’, ‘Padding Left’ or ‘Padding Left and Right’. The input parameter is verified by the algorithm and changed if necessary.

- *normalized*

  (Optional) Specifies whether you want to take an average on the raw data to reduce noise and smooth the frequency domain waveform. Specify 1 to turn this on, or 0 to not modify the raw data from calculation. Default is 0.

- *windowType*

  (Optional) Applies a windowing function from a selection of windows. Legal values: 'Rectangular" (the default), 'Hamming', 'Hanning', 'Parzen', 'Welch", 'Blackman', 'Blackman-Harris', 'Bartlett', 'Kaiser', 'Klein' or 'Dolph Chebyshev'.

  Note

  For Hanning, symmetric window shapes are preferred when using a Hanning window in FIR filter design.

  Periodic window shapes are preferred when using a Hanning window in spectral analysis. This is because the Discrete Fourier Transform assumes periodic extension of the input vector. A periodic Hanning window is obtained by constructing a symmetric window and removing the last sample.

- *alpha*

  (Optional) Specifies the alpha or beta value that is required by Hanning, Kaiser, and Dolph Chebyshev windows.

- *nsect*

  (Optional) Specifies the number of points by section. Default is 'Automatic'.

- *nauto*

  (Optional) Specifies the number of points for auto-correlation results. Default is 'Automatic'.

- *ncorr*

  (Optional) Specifies the number of auto-correlation points used for power spectral density computation. Default is 'Automatic'.

- *npsd*

  (Optional) Specifies the number of points for power spectral density results. Default is 'Automatic'.

- *computationMethod*

  (Optional) Specifies the computation method. Legal values: 'PERIODO' (the default) and 'CORRELO'.

- *f_ref*

  (Optional) Adjusts the results around the y axis so that the point for the specified frequency is 0.0.

- *f_min*

  (Optional) Specifies the starting frequency used inside the power spectral density result window.

- *f_max*

  (Optional) Specifies the last frequency used inside the power spectral density result window.

- *samplingEpsilon*

  (Optional) When *sampling* is set to ‘Interpolation’, specifies that the input waveform data point (Y value) is to be used rather than the exact interpolated value when the X data of the input waveform is close to the computed X value.

  The computed X value corresponds to *t_start* +(*points* * dX) where dX is the sampling interval retrieved from the *fs*.

- *windowShape*

  (Optional) Specifies the shape of the window. Legal values are ‘Symmetric’ for standard FFT setup or ‘Periodic’ for enhancing FFT setup for spectral analysis of periodic signals. Defaults to the global setting in the “Waveform Calculator Calculation Options” on page 545.

> **Tip**
> For comparing Eldo’s FFT results with EZwave’s FFT results, select the Periodic option, unless EZwave’s FFT is executed on an FFT_INPUT waveform. In this case, Eldo has already considered the periodicity of the input signal.

Return Values

Waveform. The Power Spectral Density of the input waveform

pssresidue

Waveform Calculator RF function.

Calculates the difference of every signal at *tstop* and *tstop* - *tperiod* and returns signals in sorted order of the maximum difference.

Usage

pssresidue(***wf***, ***tperiod***)

pssresidue(***wf***, ***tperiod***, *tstop*, *reltol*, *abstol*)

Arguments

- ***wf***

  (Required) Specifies the input continuous waveform name.

- ***tperiod***

  (Required) Specifies the fundamental period in PSS analysis. To find slow drifting signals, try specifying integer multiples of the fundamental period.

- *tstop*

  (Optional) Specifies the stop time. Default is 'End' (the maximum X value).

- *reltol*

  (Optional) Specifies the relative tolerance for the PSS residue value. Used for PSS convergence detection. PssNorm = PssDiff / (*abstol* + *reltol* * max(abs(input_signal))). A node is considered converged when PSSNorm is < 1. When PssNorm is > 100, a node is considered far from converged. Default is 1e-3.

- *abstol*

  (Optional) Specifies the absolute tolerance for the PSS residue value. Used for PSS convergence detection. PssNorm = PssDiff / (*abstol* + *reltol* * max(abs(input_signal))). A node is considered converged when PSSNorm is < 1. When PssNorm is > 100, a node is considered far from converged. Default is 1e-3.

Return Values

Double. The maximum PSS residue.

pulsewidth

Waveform Calculator Measurement function.

Measures the pulse width of a waveform that switches at least twice between high and low levels, relative to default (automatically calculated) or user-specified topline and baseline levels.

Usage

pulsewidth(***wf***)

pulsewidth(***wf***, *topline*, *baseline*)

pulsewidth(***wf***, *topline*, *baseline*, *pulsetype*)

pulsewidth(***wf***, *topline*, *baseline*, *pulsetype*, *x_start*, *x_end*)

pulsewidth(***wf***, *topline*, *baseline*, *pulsetype*, *x_start*, *x_end*, *option*)

pulsewidth(***wf***, *topline*, *baseline*, *pulsetype*, *x_start*, *x_end*, *option*, *param*)

pulsewidth(***wf***, *topline*, *baseline*, *pulsetype*, *x_start*, *x_end*, *option*, *param*, *threshold*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the signal. Specify ‘Automatic’ to have this value computed. Default is ‘Automatic’.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the signal. Specify ‘Automatic’ to have this value computed. Default is ‘Automatic’.

- *pulsetype*

  (Optional) Specifies the signal pulse type from which the measurement begins. Specify ‘Rising’ (or ‘+1’), ‘Falling’ (or ‘-1’), or ‘Either’ (or ‘0’). Default is ‘Either’.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ indicates numerical values, ‘WF’ indicates a waveform (default), and ‘ANNOTATION’ indicates the input waveform plot will be annotated with results.

- *param*

  (Optional) Specifies the simulation parameter to be used to generate the result waveform when option = ‘WF’, either ‘middleX’ of ‘Xunit’. Default is ‘middleX’.

- *threshold*

  (Optional) Specifies the y value that sets the threshold of the signal. Specify ‘Automatic’ to have this value computed: (topline + baseline)/2. Default is ‘None’.

Note

‘topline/baseline’ and ‘threshold’ parameters cannot be used simultaneously: use either ‘topline/baseline’ or ‘threshold’.

Return Values

Waveform. The pulse width of the input waveform.

rad

Waveform Calculator Signal Processing function.

Converts the trigonometric angle of a waveform to radians.

Usage

rad(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. A waveform corresponding to the converted input waveform.

real

Waveform Calculator Complex function.

Returns the real part of the input complex waveform.

Usage

real(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The real part of the input complex waveform.

reglin

Waveform Calculator Miscellaneous function.

Performs a linear regression between the values provided, and returns a three-element tuple for the slope, offset and error.

Usage

reglin(*array_of_x*, *array_of_y*)

reglin(*wf*[, *x_start*, *x_end*])

reglin(*array_of_datapoints*)

Arguments

- ***array_of_x***

  (Required) Specifies the input array of x values. For example, [*x0, x1, …, xn*].

- ***array_of_y***

  (Required) Specifies the input array of y values. For example, [*y0, y1, …, yn*].

- ***wf***

  (Required) Specifies the input waveform name.

  x_start

  (Optional) Specifies the x value at the beginning of an interval.

  x_end

  (Optional) Specifies the x value at the end of an interval.

- ***array_of_datapoints***

  Specifies the input array of points. For example, [[*x0*, *y0*], …, [*xn*, *yn*]].

Return Values

A three-element tuple (slope, offset, error).

Description

Performs a linear regression between the values provided, and returns values for the slope, offset and error.

If an array is specified, it must contain data points wrapped between square brackets ([ ]). The first element defines the *x* value, and the second one defines the *y* value. Elements are separated by a comma. For example, [[*x0*, *y0*], [*x1*, *y1*], …, [*xn*, *yn*]].

relation

Waveform Calculator Mathematical function.

Generates a waveform from a point-by-point relational expression.

Usage

relation(***wf1***, ***wf2***, ***operator***, *pt_at_crossing*)

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- ***wf2***

  (Required) Specifies the second input waveform name.

- ***operator***

  (Required) Specifies the relational operator through one of the following legal values:

  - 1 — Represents ‘greater than’.
  - 0 — Represents ‘equals’.
  - -1 — Represents ‘less than’.

- *pt_at_crossing* = ‘Yes’ | ‘No’

  (Optional) Specifies whether points are created at intersections:

  - ‘Yes’ — Points are added to the waveform when ***wf1*** and ***wf2*** cross (default).
  - ‘No’ — Points at crossings are not added to the result.

Return Values

Waveform. The relation between the two input waveforms.

Description

Generates a waveform from a point-by-point relational expression. It returns 1 corresponding to the scalar value operator if any of the following conditions are true:

- ***operator*** = 1 and ***wf1*** > ***wf2***
- ***operator*** = 0 and ***wf1*** = ***wf2***
- ***operator*** = -1 and ***wf1*** < ***wf2***

otherwise, 0 is returned.

removepts

Waveform Calculator Miscellaneous function.

Removes points from a waveform where the states of a second (digital) waveform are ‘0’.

Usage

removepts(*wf1*, *wf2*)

Note

The waveform domains must overlap and ***wf2*** must be digital.

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- ***wf2***

  (Required) Specifies the second (digital) input waveform name.

Return Values

Waveform. A waveform representing ***wf1*** with points removed from the domain where digital ***wf2*** state is ‘0”.

risetime

Waveform Calculator Measurement function.

Measures the difference in time from when the waveform rises from the lower level to the upper level.

Usage

risetime(***wf***)

risetime(***wf***, *topline*, *baseline*)

risetime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*)

risetime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *x_start*, *x_end*)

risetime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *x_start*, *x_end*, *option*)

risetime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *x_start*, *x_end*, *option*, *param*)

risetime(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *x_start*, *x_end*, *option*, *param*, *rise*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of a signal. Specify ‘Automatic’ to have this value computed.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of a signal. Specify ‘Automatic’ to have this value computed.

- *low*

  (Optional) Specifies the percentage of the low threshold. This can range from 0% to mid. By default, this is set to 10%. This parameter is a string (for example, ‘10%’).

- *mid*

  (Optional) Specifies the percentage that sets the limit range for the low and up values. By default, this is set to 50%. This parameter is a string (for example, ‘50%’).

- *up*

  (Optional) Specifies the percentage of the high threshold. This can range from mid to 100%. By default, this is set to 90%. This parameter is a string (for example, ‘90%’).

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. (Optional) Specifies the output type. Legal values are: 'WF' for waveform (default), 'VALUE' for numerical value or array of numerical values and 'ANNOTATION' for plotting your input waveform with the result annotated on it.

  Note

  If *rise* is anything other than 'all', option = 'VALUE' is forced.

- *param*

  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = 'WF'. Default is 'middleX'.

- *rise*

  (Optional) Specifies the occurrence of the result that the measurement will return. For compound waveforms it applies to each element individually. Legal values:

  - 'first' — Specifies the first occurrence of the result.
  - 'all' — Specifies all occurrences of the result. Default.
  - 'last' — Specifies the last occurrence of the result.
  - *n* or *'n'* — Specifies the *n*th occurrence of the result.

Return Values

Waveform. The difference in time from when the input waveform rises from the lower level to the upper level.

ritocomplex

Waveform Calculator Complex function.

Constructs a complex waveform from a waveform of the real part and a waveform of the imaginary part.

Usage

ritocomplex(***wf1***,[*x_start1*, *x_end1*,] ***wf2***[, *x_start2*, *x_end2*])

Arguments

- ***wf1***

  (Required) Specifies the first (real) input waveform name.

- *x_start1*

  (Optional) Specifies the x value at the beginning of an interval on ***wf1***.

- *x_end1*

  (Optional) Specifies the x value at the end of an interval on ***wf1***.

- ***wf2***

  (Required) Specifies the second (imaginary) input waveform name.

- *x_start2*

  (Optional) Specifies x value at the beginning of an interval on ***wf2***.

- *x_end2*

  (Optional) Specifies the x value at the end of an interval on ***wf2***.

Return Values

Waveform. A complex waveform combining two input waveforms as real and imaginary parts.

Description

Constructs a complex waveform from a waveform of the real part (***wf1***) and a waveform of the imaginary part (***wf2***).

rms

Waveform Calculator Statistical function.

Returns the root mean square value of a waveform for transient or AC analysis.

Usage

rms(***wf***[, *x_start*, *x_end*])

rms(***wf***[, *x_start*, *x_end* , *option*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the result. ‘ANNOTATION’ plots the source waveform, annotated with the result value. Default is ‘VALUE’.

Return Values

Double. The root mean square value of the input waveform.

Description

Computes the root mean square value of a waveform for transient or AC analysis, as follows:

sqrt (integ(***wf***^{2}, *x_start*, *x_end*) if the x domain is frequency.

sqrt (integ(***wf***^{2}, *x_start*, *x_end*) / (*x_end* - *x_start*) if the x domain is not frequency.

rms_ac

Waveform Calculator Statistical function.

Returns the root mean square value of a waveform for AC analysis.

Usage

rms_ac(***wf***[, *x_start*, *x_end*])

rms_ac(***wf***[, *x_start*, *x_end* , *option*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' returns a numerical value of the result. 'ANNOTATION' plots the source waveform, annotated with the result value. Default is 'VALUE'.

Return Values

Double. The root mean square value of the input waveform.

Description

Computes the root mean square value of a waveform for AC analysis, as follows:

sqrt (integ($\boldsymbol{wf}^2$, *x_start*, *x_end*)

rms_accurate

Waveform Calculator Statistical function.

Returns the integrated root mean square value of a continuous waveform that is accurate even if the waveform is noisy.

Usage

rms_accurate(***wf***[, *x_start*, *x_end*, *option*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the result. ‘ANNOTATION’ plots the source waveform, annotated with the result value. Default is ‘VALUE’.

Return Values

Double. Returns the root mean square value of the input waveform.

rms_noise

Waveform Calculator Statistical function.

Returns the root mean square value of a discrete frequency waveform for noise analysis.

Usage

rms_noise(***wf***[, *x_start*, *x_end*])

rms_noise(***wf***[, *x_start*, *x_end*, *option*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *option*

  (Optional) Specifies the output type. 'VALUE' returns a numerical value of the result. 'ANNOTATION' plots the source waveform, annotated with the result value. Default is 'VALUE'.

Return Values

Double. The root mean square of the input waveform for noise analysis.

Description

Computes the root mean square value (also know as the root sum square value) of discrete frequency waveforms such as FOUR or FSST. It may be used, for example, in the calculation of SNR. It is computed as follows:

sqrt (sum($\boldsymbol{wf}^2$, *x_start*, *x_end*))

rms_tran

Waveform Calculator Statistical function.

Returns the root mean square value of a waveform for transient analysis.

Usage

rms_tran(***wf***[, *x_start*, *x_end*])

rms_tran(***wf***[, *x_start*, *x_end*, *option*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the result. ‘ANNOTATION’ plots the source waveform, annotated with the result value. Default is ‘VALUE’.

Return Values

Double. The root mean square value of the input waveform for transient analysis.

Description

Computes the root mean square value of a waveform for transient analysis, as follows:

sqrt (integ($\boldsymbol{wf}^2$, *x_start*, *x_end*)

rol

Waveform Calculator Logic function.

Returns a value that is the waveform rotated left by shift index positions.

Usage

rol(***wf***[, *x_start*, *x_end*], ***shift***)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

- ***shift***

  (Required) Specifies the shifted value.

Return Values

Waveform. A new waveform corresponding to the rotated input waveform.

Description

Returns a value that is the waveform ***wf*** rotated left by shift index positions.

The **rol** function replaces ***wf*** with a value that is the result of a concatenation whose left argument is the rightmost (Length - 1) elements of wf and whose left argument is the leftmost remainder of ***wf***.

The following describe the three possible shifts:

- If the shift is 0 or if ***wf*** is a null array, the return value is ***wf***.
- If the shift is positive, sra() is repeated shift times to form the result.
- If the shift is negative, the return value is the value of the function **ror** (***wf***, -shift)

ror

Waveform Calculator Logic function.

Returns a value that is the waveform rotated right by shift index positions.

Usage

ror(***wf***[, *x_start*, *x_end*], ***shift***)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.
- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.
- *x_end*

  (Optional) Specifies the x value at the end of an interval.
- ***shift***

  (Required) Specifies the shifted value.

Return Values

Waveform. A new waveform corresponding to the rotated input waveform.

Description

Returns a value that is the waveform ***wf*** rotated right by shift index positions.

The **ror** function replaces ***wf*** with a value that is the result of a concatenation whose right argument is the leftmost (Length - 1) elements of ***wf*** and whose left argument is the rightmost remainder of ***wf***.

The following describe the three possible shifts:

- If the ***shift*** is 0 or if ***wf*** is a null array, the return value is ***wf***.
- If the ***shift*** is positive, sra() is repeated shift times to form the result.
- If the ***shift*** is negative, the return value is the value of the function **rol** (***wf***, -***shift***)

round

Waveform Calculator Mathematical function.

Computes the closest integer value to each data point of the waveform.

Usage

round(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

Return Values

Waveform. A waveform containing the rounded values of the input waveform.

Description

Computes the closest integer value to each data point of ***wf***. Upon successful completion, each returned value is expressed as a type double.

Related Topics

trunc

ceil

floor

sample

Waveform Calculator Signal Processing function.

Creates a sampled waveform with equidistant data points.

Usage

sample(***wf***, ***sampling_interval***[, *x_start*, *x_end*, *strict_sampling*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***sampling_interval***

  (Required) Specifies the x interval for sampling.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

- *strict_sampling*

  (Optional) When set to ‘True’, forces equidistant digital events even when the logical state is unchanged. When set to ‘False’, a new digital even is only triggered when the logical state is changed. Default is ‘False’.

Return Values

Waveform. A new sampled waveform.

samplelog

Waveform Calculator Signal Processing function.

Creates a sampled waveform with data points logarithmically spaced.

Usage

samplelog(***wf***, ***number_of_points***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***number of points***

  (Required) Specifies the number of points in the sampled result waveform.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

Return Values

Waveform. A new logarithmically sampled waveform.

samplepsd

Waveform Calculator Signal Processing function.

Samples a continuous waveform variable *wf*, and returns its power spectral density, with unit $(w)^2$/Hz. Returned spectrum is from 0 to f_sample/2, and has frequency bin width of f_sample/nfft.

Usage

samplepsd(***wf***[*, t_start, t_stop, f_sample, nfft, window, detrend*])

Arguments

- ***wf***

  (Required) Name of the input time-based waveform.

- *t_start*

  (Optional) Sampling start time on wf. Default is 'Begin', the first point of wf.

- *t_stop*

  (Optional) Sampling stop time on wf. Default is 'End', the last point of wf.

- *f_sample*

  (Optional) Sampling frequency. Default value 'Automatic' is the inverse of the maximum time-step.

  > **Note**
  >
  > *f_sample* must be specified if *t_start* (or *t_stop*) is specified.

- *nfft*

  (Optional) Number of data samples per windowed FFT segment. If the spectrum has a strong signal at a frequency, let nfft = integer multiple of f_sample / (signal frequency) to minimize signal leakage. Larger nfft improves frequency resolution but increases variability with fewer segments. Default nfft is computed as roundnfft(f_sample/(10/(t_stop - t_start))), where roundnfft() rounds the input to ensure efficient FFT processing. Default is 'Default'.

- *window*

  (Optional) The fft window type. Available types:

  'Rectangular', 'Hamming', 'Hanning', 'Parzen', 'Welch', 'Blackman', 'Blackman-Harris', 'Bartlett', 'Kaiser', 'Klein' or 'Dolph Chebyshev'. Default is 'Hanning'.

- *detrend*

  (Optional) Subtracts the mean value or a linear best-fit line from the output to highlight fluctuations around those values. Specify: 'None', 'Mean' or 'Linear'. Default is 'None'.

Return Values

Waveform. The power spectral density of the input waveform.

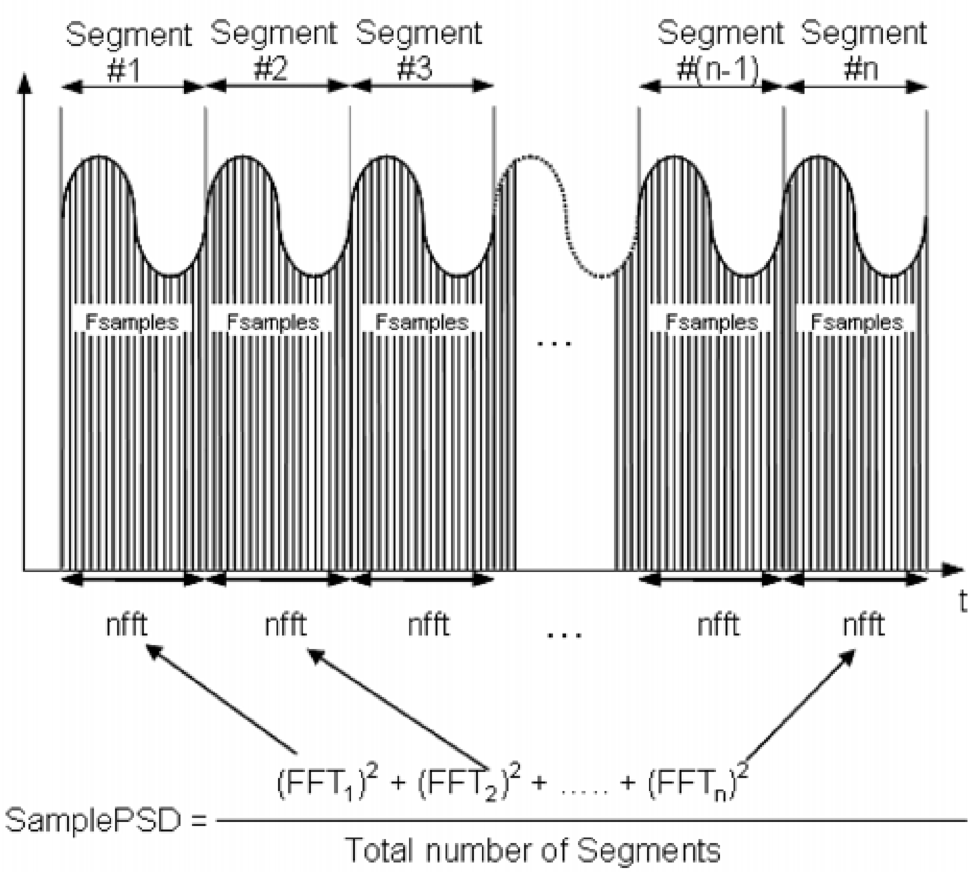
Description

The calculation involves these steps:

1. Sample the waveform at the sampling rate of *f_sample*.
2. Divide the data into segments of *nfft* data points.
3. Compute the Hann-windowed FFT of each segment.
4. Compute the average power.

The following figure shows the calculation:

Figure A-3. Calculation of SamplePSD



Examples

The samplepsd() function can be used on any time-domain waveform. In this example, samplepsd() is used to calculate the power spectral density from transient noise analysis of a RC circuit, and compare it to the result of noise analysis. In this case, the input for the samplepsd() function should not contain any energy from the signal, only noise.

The netlist used is as follows:

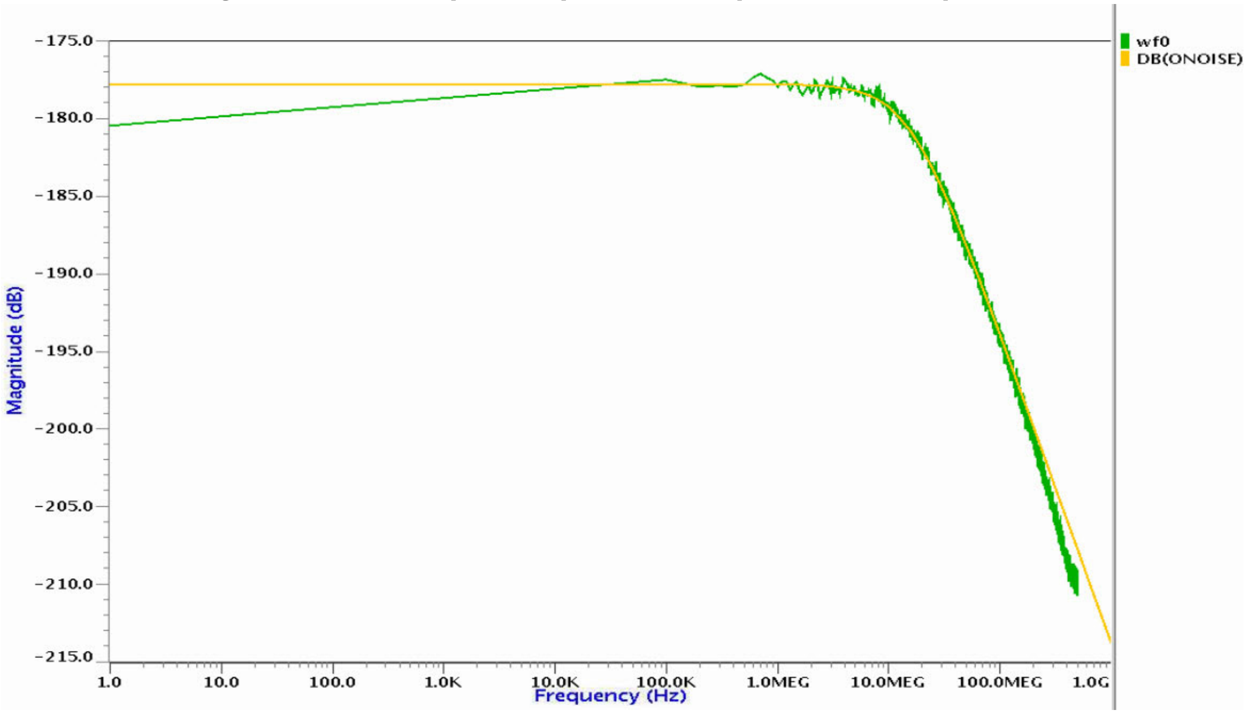
```
*
.option tuning=vhigh
.param vdc=0 vamp=0 per=1u
vin in 0 dc 0 ac 1 0 pulse('vdc-vamp/2' 'vdc+vamp/2' '0.25*per' '0.05*per'
'0.05*per' '0.45*per')
r1 in out 100
c2 out 0 100p
.ac dec 10 1 1G
.noise v(out) vin 10
.plot ac db(onoise)
.tran 1us 1ms
.probe tran v(in) v(out)
.plot ntr v(out)
.param vamp=1
.noisetran fmin=0 fmax=1G nbrun=1
.step param vamp list 0 1
```

The computation,

```
wf0=20*log(sqrt(samplepsd(wf("OUT", " -run 5  -show NOISETRAN.V  -notop
")-wf("OUT", " -run 4  -show NOISETRAN.V  -notop  "), t_start=0.0,
t_stop=0.0010, f_sample=1.0E9, nfft=10000, window='Hanning',
detrend='none,)))
```

produces the following plot:

Figure A-4. Example Output Noise Spectrum Comparison



Tip

See also .NOISETRAN in the *Eldo Reference Manual*.

setAngleUnits

Waveform Calculator Special function.

Specifies the trigonometrical angle unit used in the Waveform Calculator.

Usage

setAngleUnits(‘degrees’|‘radians’|‘gradians’)

Arguments

- ‘degrees’|‘radians’|‘gradians’

  Specifies the trigonometrical angle unit.

Description

Note

This Waveform Calculator Special function does not appear in the Waveform Calculator Functions list but may be used for scripting.

setNotation

Waveform Calculator Special function.

Specifies the expression evaluation logic used in the Waveform Calculator.

Usage

setNotation('IEEE'|'SPICE')

Arguments

- 'IEEE'|'SPICE'

  Specifies the notation.

Description

Note

This Waveform Calculator Special function does not appear in the Waveform Calculator Functions list but may be used for scripting.

setTemperatureUnits

Waveform Calculator Special function.

Specifies the temperature unit used in the Waveform Calculator.

Usage

setTemperatureUnits('celcius'|'kelvin')

Arguments

- 'celcius'|'kelvin'

  Specifies the temperature unit.

Description

Note

This Waveform Calculator Special function does not appear in the Waveform Calculator Functions list but may be used for scripting.

settlingtime

Waveform Calculator Measurement function.

Calculates the settling time of a waveform with respect to default or specified steady state level and a specified tolerance. The size of settle band is specified as the tolerance level on either side of steady state level. The settling time is the last time point that the waveform crosses the settle band, either the positive level or the negative level of tolerance, from out of bound to inner bound.

Usage

settlingtime(***wf***, *steadystate*, *tolerance*, *x_start*, *x_end*, *option*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *steadystate*

  (Optional) Specifies the steady-state y value that for the source waveform. Specify ‘Automatic’ to have this value computed. Default is ‘Automatic’.

- *tolerance*

  (Optional) Specifies the tolerance as percentage relative to the amplitude of the source waveform. Default is ‘5%’.

- *x_start*

  (Optional) Specifies the start x value for the calculation. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the stop x value for the calculation. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the settle time. ‘ANNOTATION’ plots the source waveform, annotated with the settle time. Default is ‘VALUE’.

Return Values

The time required for the input waveform to settle around the final value. The result format depends on the value of the *option* parameter.

sfdr

Waveform Calculator Signal Processing function.

Computes the spurious free dynamic range of the input waveform in dBc.

Usage

sfdr(***wf***, ***f_list***[, *f_min*, *f_max, harm, s_freq, bin*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. Typically this will be an FFT result calculated in EZwave.

- ***f_list***

  (Required) Specifies the list of frequencies to be considered as signal. Frequencies from *f_list* can be selected outside of [*f_min*, *f_max*].

- *f_min*

  (Optional) Specifies the minimum frequency to be considered in the input waveform. Default is 'Begin'.

- *f_max*

  (Optional) Specifies the maximum frequency to be considered the input waveform. Default is 'End'.

- *harm*

  (Optional) Specifies the number of harmonics of the signal to be considered. Default value 'Default' is 6 (signal + 5 harmonics). If *harm*=-1, all harmonics within [*f_min*, *f_max*] are considered.

- *s_freq*

  (Optional) The sampling frequency of the source waveform. Default value is 'None', no sampling frequency. When specified, the sampling frequency is used to manage harmonic folding.

- *bin*

  (Optional) Specifies the bin width to use as signal on the input wave (single sided). Default is 'Default'.

Return Values

Double. The spurious free dynamic range of the input waveform in dBc.

Description

The spurious free dynamic range of the input waveform is given by the following relationship:

$$10\log 10\left(\frac{S}{WorstSpur}\right)$$

where:

S = the signal.

$WorstSpur$ = the highest spectrum which is not the signal (it can be an harmonic or not).

The input waveform is always assumed to be a linear gain (magnitude).

The unit of this ratio is always given in dBc.

Examples

```
sfdr(wf("<calc>FFT_V_OUT_DAC_GOOD_"),[4637700.0],0.0,8.0E7, s_freq=80e6)
= 73.13448176568927
```

Related Topics

Signal to Noise Ratio Dialog Box

shift

Waveform Calculator Miscellaneous function.

Creates a waveform shifted in the x direction by an interval delta.

Usage

shift(***wf***[, *x_start*, *x_end*], ***delta***, *precision*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

- ***delta***

  (Required) Specifies the interval for shifting.

- *precision*

  (Optional) Specifies the floating-point precision used. By default, this is set to 12 (1e-12) but you can specify your own precision if required.

Return Values

Waveform. A new waveform corresponding to the shifted input waveform.

shiftedmaxdiff

Waveform Calculator Miscellaneous function.

Finds the time when any signal in one run, shifted by *tdelay*, differs the most from the corresponding signal in another run, and plots *ntop* signals in sorted order.

Usage

shiftedmaxdiff(*wf1*, *wf2*, [*tdelay*, *x_start*, *x_end*, *reltol*, *abstol*, *ntop*, *depth*, *plot*]

Arguments

- ***wf1***

  (Required) Specifies any continuous waveform signal in one run. Used to identify a hierarchy path in a source folder, plus sub-hierarchies with respect to the *depth* parameter.

- ***wf2***

  (Required) Specifies any continuous waveform signal in another run. Used to identify a test folder only. The name and hierarchy path are the same as those of source folder.

- *tdelay*

  (Optional) Specifies the time delay to be added to the first waveform signal. Default is 'Automatic', the last point in the second waveform minus the interval of the first waveform.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

- *reltol*

  (Optional) Specifies the relative tolerance. By default, this is set to 1e-3 but you can specify your own value if required.

- *abstol*

  (Optional) Specifies the absolute tolerance. By default, this is set to 1e-6 but you can specify your own value if required.

- *ntop*

  (Optional) Specifies the number of top results to return and plot. Default is 8.

- *depth*

  (Optional) Specifies the hierarchy depth used to search the waveforms. Recursive search by default.

- *plot*

  (Optional) 1 (default) plots the shifted maximum difference results. 0 does not plot results.

Return Values

Array. The shifted maximum difference value for the given input waveforms, in the format:

[[wf1_handle, wf2_handle, shifted_max_diff_value], [...]]

Examples

If you specify:

```
shiftedmaxdiff(wf("<db1/folder1>path/to/wf"), wf("<db2/folder2>another/
path/to/wf")),
```

then **shiftedmaxdiff** is computed on pair of waveforms having the names:

```
{ wf("<db1/folder1>path/to/something"), wf("<db2/folder2>path/to/
something") }
```

Related Topics

firstdiff

maxdiff

maxspectrumdiff

sin

Waveform Calculator Trigonometric function.

Computes the sine of a waveform.

Usage

sin(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The sine of the input waveform.

Description

Computes the sine of ***wf***, by default measured in degrees.

sinad

Waveform Calculator Signal Processing function.

Computes the signal to noise and distortion ratio (SINAD) of the input waveform. This is the same as sndr.

Usage

sinad(***wf***, ***f_list***[, *f_min*, *f_max, harm, s_freq, bin*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. Typically this will be an FFT result calculated in EZwave.

- ***f_list***

  (Required) Specifies the list of frequencies to be considered as signal. Frequencies from *f_list* can be selected outside of [*f_min*, *f_max*].

- *f_min*

  (Optional) Specifies the minimum frequency to be considered in the input waveform. Default is ‘Begin’.

- *f_max*

  (Optional) Specifies the maximum frequency to be considered the input waveform. Default is ‘End’.

- *harm*

  (Optional) Specifies the number of harmonics of the signal to be considered. Default value ‘Default’ is 6 (signal + 5 harmonics). If *harm*=-1, all harmonics within [*f_min*, *f_max*] are considered.

- *s_freq*

  (Optional) The sampling frequency of the source waveform. Default value is ‘None’, no sampling frequency. When specified, the sampling frequency is used to manage harmonic folding.

- *bin*

  (Optional) Specifies the bin width to use as signal on the input wave (single sided). Default is ‘Default’.

Return Values

Double. The signal to noise and distortion ratio (SINAD) of the input waveform.

Description

The signal to noise and distortion ratio (SINAD) of the input waveform is given by the following relationship:

$$10\log 10\left(\frac{S}{(N+D)}\right)$$

where:

S = the signal.

N = the noise.

D = the distortion.

The input waveform is always assumed to be a linear gain (magnitude).

The unit of this ratio is always given in dB.

Examples

```
sinad(wf("<calc>FFT_V_OUT_DAC_GOOD_"),[4637700.0],0.0,8.0E7, s_freq=80e6)
= 60.71366130019308
```

sinh

Waveform Calculator Trigonometric function.

Computes the hyperbolic sine of a waveform.

Usage

sinh(*wf*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The hyperbolic sine of the input waveform.

Description

Computes the hyperbolic sine of ***wf***.

size

Waveform Calculator Statistical function.

Returns the number of data points in an analog waveform or the number of transitions in a digital waveform.

Usage

size(*wf*,[*x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Integer. The number of data points of the input waveform

sla

Waveform Calculator Logic function.

Returns a value that is the input waveform arithmetically shifted left by a number of index positions.

Usage

sla(*wf*[, *x_start*, *x_end*], ***shift***)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

- ***shift***

  (Required) Specifies the amount by which to shift wf.

Return Values

Waveform. A new waveform corresponding to the shifted input waveform.

Description

Returns a value that is the input waveform arithmetically shifted left by a number of index positions. That is, if the shift value is 0, the return value is the input waveform. Otherwise, a basic shift operation replaces the waveform with one that is the result of a concatenation whose left argument is the rightmost (length - 1) data points of the waveform and whose right argument is a duplicate of the rightmost data point. If the shift value is positive, this basic shift operation is repeated that number of times to form the result.

slewrate

Waveform Calculator Measurement function.

Measures the slew rate of a periodic waveform relative to default (automatically calculated) or user-specified topline and baseline levels.

Usage

slewrate(***wf***)

slewrate(***wf***, *topline*, *baseline*)

slewrate(***wf***, *topline*, *baseline*, *low*, *mid*, *up*)

slewrate(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *edgetrigger*)

slewrate(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *edgetrigger*, *x_start*, *x_end*)

slewrate(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *edgetrigger*, *x_start*, *x_end*, *option*)

slewrate(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *edgetrigger*, *x_start*, *x_end*, *option*, *param*)

slewrate(***wf***, *topline*, *baseline*, *low*, *mid*, *up*, *edgetrigger*, *x_start*, *x_end*, *option*, *param*, *slewrate*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the signal. Specify ‘Automatic’ to have this value computed.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the signal. Specify ‘Automatic’ to have this value computed.

- *low*

  (Optional) Specifies the percentage of the low threshold. This can range from 0% to mid. By default, this is set to 10%. This parameter is a string (for example, ‘10%’).

- *mid*

  (Optional) Specifies the percentage that sets the limit range for the low and up values. By default, this is set to 50%. This parameter is a string (for example, ‘50%’).

- *up*

  (Optional) Specifies the percentage of the high threshold. This can range from mid to 100%. By default, this is set to 90%. This parameter is a string (for example, ‘90%’).

- *edgetrigger*

  (Optional) Specifies the signal edge from which the measurement begins. Specify ‘Rising’, ‘Falling’ or ‘Either’ (default).

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ indicates numerical values, ‘WF’ indicates a waveform (default), and ‘ANNOTATION’ indicates the input waveform plot will be annotated with results.

- *param*

  (Optional) Specifies the simulation parameters to be used to generate the result waveform when option = ‘WF’. Default is ‘middleX’.

- *slewrate*

  (Optional) Specifies the measurement output type (‘All’ (default), ‘First’, ‘Last’).

Return Values

Waveform. The slew rate of the input waveform.

slope

Waveform Calculator Measurement function.

Returns the slope value of a waveform at a specified x value.

Usage

slope(***wf***, ***x***)

slope(***wf***, ***x***, *slopetype*)

slope(***wf***, ***x***, *slopetype*, *option*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***x***

  (Required) Specifies the x value.

- *slopetype*

  (Optional) Specifies the slope type. Value may be one of the following:

  - None. The slope value is computed normally. Default.
  - Decade. The slope value is computed per decade for waveforms in the frequency domain.
  - Octave. The slope value is computed per octave for waveforms in the frequency domain.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the result. ‘ANNOTATION’ plots the source waveform, annotated with the result value. Default is ‘VALUE’.

Return Values

Waveform. The slope of the input waveform at the given x value.

slopeintersect

Waveform Calculator Measurement function.

Finds the slope intersection of two waveforms at specified x values.

Usage

slopeintersect(*wf1*, *wf2*, *x1*, *x2*, *option*)

Arguments

- ***wf1***

  (Required) Specifies the first input waveform name.

- ***wf2***

  (Required) Specifies the second input waveform name.

- ***x1***

  (Required) Specifies the x value corresponding to the first waveform.

- ***x2***

  (Required) Specifies the x value corresponding to the second waveform.

- *option*

  (Optional) Specifies the output type. 'VALUE' returns a numerical value of the result. 'ANNOTATION' plots the source waveform, annotated with the result value. Default is 'VALUE'.

Return Values

Array. The slope intersections.

Related Topics

Measuring Slope Intersect

sndr

Waveform Calculator Signal Processing function.

Computes the signal to noise and distortion ratio (SINAD) of the input waveform. This is the same as sinad.

Usage

sndr(***wf***, ***f_list***[, *f_min*, *f_max, harm, s_freq, bin*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. Typically this will be an FFT result calculated in EZwave.

- ***f_list***

  (Required) Specifies the list of frequencies to be considered as signal. Frequencies from *f_list* can be selected outside of [*f_min*, *f_max*].

- *f_min*

  (Optional) Specifies the minimum frequency to be considered in the input waveform. Default is ‘Begin’.

- *f_max*

  (Optional) Specifies the maximum frequency to be considered the input waveform. Default is ‘End’.

- *harm*

  (Optional) Specifies the number of harmonics of the signal to be considered. Default value ‘Default’ is 6 (signal + 5 harmonics). If *harm*=-1, all harmonics within [*f_min*, *f_max*] are considered.

- *s_freq*

  (Optional) The sampling frequency of the source waveform. Default value is None, no sampling frequency. When specified, the sampling frequency is used to manage harmonic folding.

- *bin*

  (Optional) Specifies the bin width to use as signal on the input wave (single sided). Default is ‘Default’.

Return Values

Double. The signal to noise and distortion ratio (SINAD) of the input waveform.

Description

The signal to noise and distortion ratio (SINAD) of the input waveform is given by the following relationship:

$$10\log 10\left(\frac{S}{(N+D)}\right)$$

where:

S = the signal.

N = the noise.

D = the distortion.

The input waveform is always assumed to be a linear gain (magnitude).

The unit of this ratio is always given in dB.

Examples

```
sndr(wf("<calc>FFT_V_OUT_DAC_GOOD_"),[4637700.0],0.0,8.0E7, s_freq=80e6)
= 60.71366130019308
```

Related Topics

Signal to Noise Ratio Dialog Box

snr

Waveform Calculator Signal Processing function.

Computes the signal to noise ratio of the input waveform.

Usage

snr(***wf***, ***f_list***[, *f_min*, *f_max, harm, s_freq, bin*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. Typically this will be an FFT result calculated in EZwave.

- ***f_list***

  (Required) Specifies the list of frequencies to be considered as signal. Frequencies from *f_list* can be selected outside of [*f_min*, *f_max*].

- *f_min*

  (Optional) Specifies the minimum frequency to be considered in the input waveform. Default is 'Begin'.

- *f_max*

  (Optional) Specifies the maximum frequency to be considered the input waveform. Default is 'End'.

- *harm*

  (Optional) Specifies the number of harmonics of the signal to be considered. Default value 'Default' is 6 (signal + 5 harmonics). If *harm*=-1, all harmonics within [*f_min*, *f_max*] are considered.

- *s_freq*

  (Optional) The sampling frequency of the source waveform. Default value is None, no sampling frequency. When specified, the sampling frequency is used to manage harmonic folding and aliasing.

- *bin*

  (Optional) Specifies the bin width to use as signal on the input wave (single sided). Default is 'Default'.

Return Values

Double. The signal to noise ratio of the input waveform.

Description

The signal to noise ratio of the input waveform is given by the following relationship:

$$10\log 10\left(\frac{S}{N}\right)$$

where:

S = the signal.

N = the noise.

The input waveform is always assumed to be a linear gain (magnitude).

The unit of this ratio is always given in dB.

Examples

```
snr(wf("<calc>FFT_V_OUT_DAC_GOOD_"),[4637700.0],0.0,8.0E7, s_freq=80e6) =
61.54552999317448
```

Related Topics

Signal to Noise Function

Signal to Noise Ratio Dialog Box

sorty

Creates the same output as the cdf function, but without normalization.

Usage

sorty(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

Description

Creates the same output as the cdf function, but without normalization. The first point of the result is {x=ymin(wf), y=1} and the last point is {x=ymax(wf), y=size(wf)}.

spectrummeasurement

Waveform Calculator Signal Processing function.

Computes spectrum measurements on the input analog or complex waveform, including; "snr", "sndr", "sfdr", "enob", and "thd".

Usage

spectrummeasurement(*wf*[, *t_start*, *t_stop*, *fs* | *points*, *f_min*, *f_max*, *window_type*, *alpha*, *harm*, *bin*, *meas_type*, *config_file*, *windowShape*, *method*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is 'End'.

- *fs*

  (Optional) Specifies the sampling frequency of the signal. Default is 'Automatic'. Specify either *fs* or *points*, not both.

- *points*

  (Optional) Specifies the number of sampling points. Default is 'Automatic'. Specify either *fs* or *points*, not both.

  Note

  The parameters above satisfy the following equation:

  ((*points*-1)/*fs*) = *t_stop* - *t_start*

- *f_min*

  (Optional) Specifies the starting frequency used in the spectrum measurement noise integration calculation. Default is 'Automatic'.

- *f_max*

  (Optional) Specifies the last frequency used in the spectrum measurement noise integration calculation. Default is 'Automatic'.

- *window_type*

  (Optional) Applies a windowing function from a selection of windows. Legal values: 'Rectangular' (the default), 'Hamming', 'Hanning', 'Parzen', 'Welch', 'Blackman', 'Blackman-Harris', 'Bartlett', 'Kaiser', 'Klein' or 'Dolph Chebyshev'.

- *alpha*

  (Optional) Specifies the alpha (or beta) value that is required by Hanning, Dolph Chebyshev and Kaiser windows.

  - Hanning : [0;1], default 0.5
  - Dolph Chebyshev : [0:20], default 3.0
  - Kaiser : [0:20], default 10.056.

- *harm*

  (Optional) Specifies the number of harmonics to be considered. Set to -1 to consider all harmonics. Default is 6, equivalent to signal plus 5 harmonics.

- *bin*

  (Optional) Specifies the bin size for the spectrum measurement, that is, the number of points to take into account around the fundamental signal. For example:

  - with bin = 0, only the fundamental is taken into account
  - with bin = 1, the frequencies taken into account correspond to the first point to the left and right of the found fundamental and each of its associated harmonics.

  The default value depends on the Windowing transform selected.

  - Rectangular: bin = 0
  - Bartlett, Hamming, Hanning, Welch: bin = 1
  - Blackman, Dolph Chebyshev, Kaiser, Klein: bin = 2
  - Blackman-Harris, Parzen: bin = 3.

- *meas_type*

  (Optional) Specifies the type of measurement to be performed. Default value is ‘All’ (all measurement types are calculated). Allowed values:

  - ‘snr’ (Signal to Noise Ratio)
  - ‘sndr’ (Signal to Noise and Distortion Ratio)
  - ‘sfdr’ (Spurious Free Dynamic Range)
  - ‘enob’ (Effective Number Of Bits)
  - ‘thd’ (Total Harmonic Distortion in dB)
  - ‘thd_pc’ (Total Harmonic Distortion in %)
  - ‘fund’ (fundamental frequency)
  - ‘dc’ (value of the dc amplitude)
  - ‘signal’ (value of the fundamental amplitude)

- ‘nb_fund’ (number of bins used in the fund value calculation)
- ‘nb_harm’ (number of bins used in the harmonic calculation)
- ‘all’ (All of the above results).

- *config_file*

  (Optional) Specifies a configuration file that contains input parameters. A configuration file may be generated by entering parameters on the Spectrum Measurement Tool Dialog Box and then selecting **File** > **Save Configuration**. When specified, only the source waveform is mandatory, and any other parameters specified in the function are overridden by those in the configuration file.

- *windowShape*

  (Optional) Specifies the shape of the window. Allowed values are ‘Symmetric’ (for standard FFT setup) or ‘Periodic’ (for enhanced FFT setup for spectral analysis of periodic signals). Default is ‘Periodic’.

- *method*

  (Optional) Specifies the FFT algorithm (‘Default’ or ‘pack’). Default is ‘pack’.

Return Values

List. The spectrum measurement(s) of the input waveform.

Related Topics

Using the Spectrum Measurement Tool

Spectrum Measurement Tool Dialog Box

sphibyjitter

Waveform Calculator Phase Noise function.

Computes two-sided Sphi density in dB rad^2/Hz by jitter measurement. The positive side of Sphi is returned.

Usage

sphibyjitter(*wf*, *fund*[, *t_start, t_stop, rbw, edge, topline, baseline*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. The waveform must be the result of a transient noise analysis.

- ***fund***

  (Required) Specifies the fundamental or center frequency.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is 'End'.

- *rbw*

  (Optional) Specifies the frequency resolution bandwidth. Finer RBW increases the variability in the measured phase noise, and reduces the lower offset frequency limit. Default value 'Automatic' is 10/(t_stop-t_start).

- *edge*

  (Optional) Specifies the edge to consider when calculating periods. Valid values are 'Rising' or 'Falling'. Default value is 'Rising'.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of a signal. Specify 'Automatic' to have this value computed.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of a signal. Specify 'Automatic' to have this value computed.

Return Values

Waveform. The two-sided Sphi density in dB rad^2/Hz by jitter measurement.

Description

The resultant Sphi waveform has a datum “noise” in dBc/Hz whereas the expression noisetrantophasenoise has a datum magnitude in dB/Hz.

Examples

```
sphibyjitter(v0, 3.4G)
```

Tip

See also .NOISETRAN in the *Eldo Reference Manual*.

sphifilter

Waveform Calculator Phase Noise function.

Applies a high pass filter to frequency-domain Sphi density.

Usage

sphifilter(*wf*, *polefreq*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name. The waveform must be a two-sided Sphi density in dB rad^2/Hz vs. frequency in Hz, as computed using sphibyjitter, for example.

- ***polefreq***

  (Required) List of pole frequencies in Hz. Complex poles should be specified as complex conjugate pairs.

Return Values

Waveform. The high pass filter of the input frequency-domain Sphi density.

Examples

```
sphibyfilter(v0, 1e6)
```

Tip

See also .NOISETRAN in the *Eldo Reference Manual*.

sqr

Waveform Calculator Mathematical function.

Computes $\boldsymbol{wf}^2$.

Usage

sqr(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The square of the input waveform.

sqrt

Waveform Calculator Mathematical function.

Computes the square root of ***wf***.

Usage

sqrt(*wf*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The square root of the input waveform.

sra

Waveform Calculator Logic function.

Returns a value that is the input waveform arithmetically shifted right by a number of index positions.

Usage

sra(*wf*[, *x_start*, *x_end*], ***shift***)**

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

- ***shift***

  (Required) Specifies the amount by which to shift ***wf***.

Return Values

Waveform. A new waveform corresponding to the shifted input waveform.

Description

Returns a value that is the input waveform arithmetically shifted right by a number of index positions. That is, if the shift value is 0, the return value is the input waveform. Otherwise, a basic shift operation replaces the waveform with one that is the result of a concatenation whose right argument is the leftmost (length - 1) data points of the waveform and whose left argument is a duplicate of the leftmost data point. If the shift value is positive, this basic shift operation is repeated that number of times to form the result.

stddev

Waveform Calculator Measurement function.

Calculates the standard deviation of the specified waveform.

Usage

stddev(***wf***[, *x_start*, *x_end*, *option*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the result. ‘ANNOTATION’ plots the source waveform, annotated with the result value. Default is ‘VALUE’.

Return Values

Waveform. The standard deviation of the input waveform.

sum

Waveform Calculator Statistical function.

Finds the sum of all the y values of the input waveform.

Usage

sum(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double/complex. The sum of the y values of the input waveform.

Description

If the input waveform is complex then the sum is only calculated on its real part.

tan

Waveform Calculator Trigonometric function.

Computes the tangent of ***wf***.

Usage

tan(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The tangent of the input waveform.

Description

Computes the tangent of ***wf***, by default measured in degrees.

tanh

Waveform Calculator Trigonometric function.

Computes the hyperbolic tangent of the waveform.

Usage

tanh(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. The hyperbolic tangent of the input waveform.

thd

Waveform Calculator Signal Processing function.

The total harmonic distortion function computes the distortion to signal ratio of the input waveform.

Usage

thd(***wf***, ***f_list***[, *f_min*, *f_max, harm, s_freq, bin*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name. Typically this will be an FFT result calculated in EZwave.

- ***f_list***

  (Required) Specifies the list of frequencies to be considered as signal, in the form [freq0, ... , freqN]. Frequencies from *f_list* can be selected outside of [*f_min*, *f_max*].

- *f_min*

  (Optional) Specifies the minimum frequency to be considered in the input waveform. Default is 'Begin'.

- *f_max*

  (Optional) Specifies the maximum frequency to be considered the input waveform. Default is 'End'.

- *harm*

  (Optional) Specifies the number of harmonics of the signal to be considered. Default value is 6 (signal + 5 harmonics). If *harm*=-1, all harmonics within [*f_min*, *f_max*] are considered.

- *s_freq*

  (Optional) The sampling frequency of the source waveform. Default value is 'None', no sampling frequency. When specified, the sampling frequency is used to manage harmonic folding and aliasing.

- *bin*

  (Optional) Specifies the bin width to use as signal on the input wave (single sided). Default is 'Default'.

Return Values

Double. The distortion to signal ratio of the input waveform in dB.

Description

The total harmonic distortion of the input waveform is given by the following relationship:

10 log 10 (D/S)

where:

D = the sum over all squares of distortion amplitudes of the signal.

S = the sum over all squares of amplitudes of the sinusoidal frequencies.

The input waveform is always assumed to be a linear gain (magnitude).

The unit of this ratio is always given in dB.

Examples

```
thd(wf("<calc>FFT_V_OUT_DAC_GOOD_"),[4637700.0],0.0,8.0E7, s_freq=80e6) =
-68.3002538138697
```

tiejitter

Waveform Calculator Time-Domain Jitter function.

Computes the Time Interval Error (TIE) Jitter. TIE Jitter is the variation in time of the source waveform edges relative to the reference waveform edges over part of, or the whole waveform.

Usage

tiejitter(***wf***, ***ref***[*t_start*, *t_ stop*, *edgetrigger*, *single_threshold*, *single_threshold_ref*, *topline*, *baseline*, *threshold*, *topline_ref*, *baseline_ref*, *threshold_ref*])

Arguments

- ***wf***

  (Required) Specifies the input waveform for which to calculate the jitter.

- ***ref***

  (Required) Specifies a reference waveform.

- *t_start*

  (Optional) Specifies the X (time) value at the beginning of the jitter analysis interval. Possible values are 'Begin' or a double value. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the X (time) value at the end of the jitter analysis interval. Possible values are 'End' or a double value. Default is 'End'.

- *edgetrigger*

  (Optional) Defines the waveform edge(s) that will be taken into account during the jitter calculation. Possible values are 'Rising', 'Falling', or 'Either'. Default is 'Either'.

- *single_threshold*

  (Optional) If 'True', the *single threshold* will be used for the source waveform period calculation (*topline*/*baseline* values are ignored). If 'False', *topline*/*baseline* will be used. Default is 'False'.

- *single_threshold_ref*

  (Optional) If 'True', the single *threshold_ref* will be used for the reference waveform period calculation (*topline_ref*/*baseline_ref* values are ignored). If 'False' *topline_ref*/*baseline_ref* will be used. Default is 'False'.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the source waveform. Default is 'Automatic'.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the source waveform. Default is 'Automatic'.

- *threshold*

  (Optional) Specifies the y value that sets a single threshold for the source waveform. Default is ‘Automatic’.
- *topline_ref*

  (Optional) Specifies the y value that sets the high threshold of the reference waveform. Default is ‘Automatic’.
- *baseline_ref*

  (Optional) Specifies the y value that sets the low threshold of the reference waveform. Default is ‘Automatic’.
- *threshold_ref*

  (Optional) Specifies the y value that sets a single threshold for the reference waveform. Default is ‘Automatic’.

Return Values

Waveform. The Time Interval Error jitter waveform for the input waveforms.

Description

TIE helps to evaluate the signal accuracy and the short- and long-term signal stability.

TIE jitter is calculated as the difference between edges (rising, falling, or both) of the source and reference waveforms. The jitter RMS value horizontal marker is displayed over the result waveform.

Related Topics

Jitter Measurement Types

timeabsolutejitter

Waveform Calculator Time-Domain Jitter function.

Returns the absolute jitter waveform for the selected source waveform(s).

Usage

timeabsolutejitter(***wf***[, *ref*, *t_start*, *t_stop*, *edgetrigger*, *single_threshold*, *topline*, *baseline*, *threshold*])

Arguments

- ***wf***

  (Required) Specifies the input waveform(s) for which to calculate the jitter.

- *ref*

  (Optional) Specifies the reference period. Default is 'Automatic'.

- *t_start*

  (Optional) Specifies the X (time) value at the beginning of the jitter analysis interval. Possible values are 'Begin' or a double value. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the X (time) value at the end of the jitter analysis interval. Possible values are 'End' or a double value. Default is 'End'.

- *edgetrigger*

  (Optional) Defines the waveform edge(s) that will be taken into account during the jitter calculation. Possible values are 'Rising', 'Falling', or 'Either'. Default is 'Rising'.

- *single_threshold*

  (Optional) If 'True', the *single threshold* will be used (*topline*/*baseline* values are ignored). If 'False', *topline*/*baseline* will be used. Default is 'False'.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the source waveform. Default is 'Automatic'.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the source waveform. Default is 'Automatic'.

- *threshold*

  (Optional) Specifies the y value that sets a single threshold for the source waveform. Default is 'Automatic'.

Return Values

Waveform. The absolute jitter waveform for the input waveform(s).

Description

Absolute jitter is the accumulated difference between the actual cycle period and the reference or average source waveform period values. This type of jitter helps to evaluate the short- and long-term signal stability.

Absolute jitter is calculated as the accumulated difference between the actual cycle period values of the source waveform and reference period. If the reference period is not set, the accumulated difference is calculated between the actual and average cycle period of the source waveform.

The jitter RMS value horizontal marker is displayed over the result waveform.

Related Topics

Jitter Measurement Types

timelongtermjitter

Waveform Calculator Time-Domain Jitter function.

Returns the long-term jitter waveform for the selected source waveform(s).

Usage

timelongtermjitter(***wf***[, *ref*, *nbcycles*, *t_start*, *t_stop*, *edgetrigger*, *single_threshold*, *topline*, *baseline*, *threshold*])

Arguments

- ***wf***

  (Required) Specifies the input waveform for which to calculate the jitter.

- *ref*

  (Optional) Specifies a reference period. Default is 'Automatic'.

- *nbcycles*

  (Optional) Specifies the maximum number of waveform cycles in the N-cycle sample that will participate in the jitter calculation. The calculation starts from 1 cycle in time interval and stops on *nbcycles* cycles. Possible values are an integer value or 'Automatic'. When 'Automatic' is set, *nbcycles* = total number of cycles in the waveform / 50. This is done for a more accurate calculation of each long-term jitter point (needs at least 50 values in the sample). Default is 'Automatic'.

  > **Tip**
  >
  > By default, the maximum time of the jitter result waveform = maximum time of the source waveform / 50. So a 1 second simulation waveform gives a 20ms jitter result waveform. Increasing the value of *nbcycles* increases the result waveform maximum time, but at the same time it decreases the calculation precision for the jitter waveform, especially for points situated close to the ends of the waveform, due to the lack of statistical data. Possible solutions:
  >
  > - increase the simulation time (by 50x), if possible
  > - try using the Phase Noise longtermjitter function in EZwave. This uses an algorithm based on the integration and does not need the division by 50
  > - use the long-term jitter calculation integrated in Eldo and create a waveform from it.

- *t_start*

  (Optional) Specifies the X (time) value at the beginning of the jitter analysis interval. Possible values are 'Begin' or a double value. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the X (time) value at the end of the jitter analysis interval. Possible values are 'End' or a double value. Default is 'End'.

- *edgetrigger*

  (Optional) Defines the waveform edge(s) that will be taken into account during the jitter calculation. Possible values are 'Rising', 'Falling', or 'Either'. Default is 'Rising'.
- *single_threshold*

  (Optional) Specifies whether *topline*/*baseline* or a single threshold is used for the source waveform period calculation. If True, the single *threshold* will be used (*topline*/*baseline* values are ignored). If 'False', *topline*/*baseline* will be used. Default is 'False'.
- *topline*

  (Optional) Specifies the Y value that sets the high threshold of the source waveform. Default is 'Automatic'.
- *baseline*

  (Optional) Specifies the Y value that sets the low threshold of the source waveform. Default is 'Automatic'.
- *threshold*

  (Optional) Specifies the Y value that sets a single threshold for the source waveform. Default is 'Automatic'.

Return Values

Waveform. The long-term jitter waveform for the input waveform(s).

Description

Long-term jitter can be described as the variation in time of the cumulative period of adjacent N-cycle samples. In other words, long-term jitter is the set of N-period jitter RMS values, for N starting from 1 to the specified maximum number of cycles in the N-cycle sample.

Note

The timelongtermjitter function calculates 1-sigma jitter.

Each point in the result waveform corresponds to the N-period jitter RMS value for the given number of cycles (from 1 to *nbcycles*). This means that the first value on the long-term jitter waveform corresponds to the period jitter RMS value (N = 1).

This type of jitter is sometimes called "accumulated" jitter. It can help to evaluate the signal's long term accuracy and stability.

The jitter RMS value is not calculated for long-term jitter.

Related Topics

Jitter Measurement Types

timeperiodjitter

Waveform Calculator Time-Domain Jitter function.

Returns the N-period jitter waveform for the specified source waveform(s).

Usage

timeperiodjitter(*wf*[, *ref, t_start*, *t_ stop*, *edgetrigger, nbcycles, single_threshold, topline, baseline, threshold]*)

Arguments

- *wf*

  (Required) Specifies the input waveform for which to calculate the jitter.

- ***ref***

  (Optional) Specifies the reference period. Default is 'Automatic'.

- *t_start*

  (Optional) Specifies the X (time) value at the beginning of the jitter analysis interval. Possible values are 'Begin' or a double value. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the X (time) value at the end of the jitter analysis interval. Possible values are 'End' or a double value. Default is 'End'.

- *edgetrigger*

  (Optional) Defines the waveform edge(s) that will be taken into account during the jitter calculation. Possible values are 'Rising', 'Falling', or 'Either'. Default is 'Rising'.

- *nbcycles*

  (Optional) Specifies the maximum number of waveform cycles in the time interval (N-cycle sample), which will participate in the jitter calculation. Default is 1.

- *single_threshold*

  (Optional) If 'True', the single *threshold* will be used (*topline*/*baseline* values are ignored). If 'False', *topline*/*baseline* will be used. Default is 'False'.

- *topline*

  (Optional) Specifies the y value that sets the high threshold of the source signal. Default is 'Automatic'.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold of the source signal. Default is 'Automatic'.

- *threshold*

  (Optional) Specifies the y value that sets a single threshold for the source signal. Default is 'Automatic'.

Return Values

Waveform. The N-period jitter waveform for the input waveform(s).

Description

N-Period jitter is the variation in time of the duration of an N-cycle sample (each sample consists of N adjacent cycles) of the source waveform relative to the average N-cycle sample duration, or relative to the reference (or ideal clock) N-cycle sample duration, over part of, or the whole waveform.

For N = 1, N-cycle jitter equals the Period jitter.

This type of jitter helps to evaluate the short- and long-term signal stability.

Related Topics

Jitter Measurement Types

timestep

Waveform Calculator Miscellaneous function.

Returns a waveform of timesteps vs. times of a waveform variable.

Usage

timestep(*wf*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

Return Values

Waveform. The input waveform's timesteps vs. times of the input waveform variable.

todchysteresis

Waveform Calculator Miscellaneous function.

Creates a DC Hysteresis waveform from a two step DC simulation.

Usage

todchysteresis(*wf*)

todchysteresis(*wf1*, *wf2*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name (compound DC).
- ***wf1***

  (Required) Specifies the input waveform name (DC simulation step forward).
- ***wf2***

  (Required) Specifies the input waveform name (DC simulation step backward).

Return Values

Waveform. The DC Hysteresis waveform corresponding to the input waveform(s).

Description

When a DC Hysteresis simulation is performed using a two-step DC simulation, separate waveforms for the upward and downward simulations are generated. In this case, a transformation is required to process these waveforms before you can use the hysteresis measurement features and dedicated cursors in EZwave.

todchysteresis(wf) creates a compound of DC Hysteresis waveforms from 2*N simple waveforms grouped in a compound of N elements. The compound waveform has a simple waveform indexed from 1 to N, the result <i> is the transformation of the compound element (2*i)-1 with the element 2*i.

todchysteresis(wf1, wf2) creates a single DC Hysteresis waveform from the two simple (not compound) waveforms. For compound waveforms, it creates half the number of hysteresis waveforms compared to the number of input waveforms:

> result_1 is created from input_1 + input_2,
>
> result_2 is created from input_3 + input_4,
>
> …

topline

Waveform Calculator Measurement function.

Returns the topline of the input waveform based on the calculation of histograms.

Usage

topline(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

Return Values

Waveform. The topline of the input waveform.

topnoise

Waveform Calculator RF function.

Returns noise spectrum in sorted order of spot noise if only *fstart* is specified, or integrated noise if both *fstart* and *fstop* are specified.

Usage

topnoise(*wf*)

topnoise(*wf*, *fstart*, *fstop*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name. Any noise spectrum signal, such as “out”.

- ***fstart***

  (Required) Specifies the lower frequency limit for integrated noise or the single frequency value at which to report spot noise.

- *fstop*

  (Optional) Specifies the upper frequency limit. If specified, integrated noise is calculated. If not specified, spot noise at *fstart* is calculated.

Return Values

Double. The top noise.

Related Topics

dataset topnoise

trunc

Waveform Calculator Mathematical function.

Computes the floor for positive data points and ceiling for negative data points of the waveform.

Usage

trunc(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is 'Begin'.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is 'End'.

Return Values

Waveform. A waveform containing the truncated values of the input waveform.

Description

Computes the floor for positive data points and ceiling for negative data points of ***wf***. Upon successful completion, each returned value is expressed as a type double.

Related Topics

round

ceil

floor

undershoot

Waveform Calculator Measurement function.

Calculates the undershoot value of a waveform. The undershoot value is calculated as the difference between the minimum point and the baseline level of the waveform.

Usage

undershoot(***wf***, *topline*, *baseline*, *x_start*, *x_end*, *option*, *param*, *undershoot*)

Arguments

- ***wf***

  (Required) Specifies the source waveform name.

- *topline*

  (Optional) Specifies the y value that sets the high threshold for the source waveform. Specify ‘Automatic’ to have this value computed. Default is ‘Automatic’.

- *baseline*

  (Optional) Specifies the y value that sets the low threshold for the source waveform. Specify ‘Automatic’ to have this value computed. Default is ‘Automatic’.

- *x_start*

  (Optional) Specifies the start x value for the calculation. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the stop x value for the calculation. Default is ‘End’.

- *option*

  (Optional) Specifies the output type. ‘VALUE’ returns a numerical value of the undershoot. ‘WF’ returns the result waveform. ‘ANNOTATION’ plots the source waveform, annotated with the overshoot. Default is ‘WF’.

- *param*

  (Optional) Specifies the simulation parameter to be used to generate the result waveform for *option* ‘WF’. Default is ‘undershootX’.

- *undershoot*

  (Optional) Specifies the returned result. Choose ‘all’, ‘first’, ‘last’ or an integer. Default is ‘all’.

Return Values

The result format depends on the value of the *option* parameter.

var

Waveform Calculator Miscellaneous function.

Enables database variable values (scalars) to be accessed and used in Waveform Calculator expressions.

Usage

var(***name***)

var(***name***, *database*, *folder*)

var(***name***, *database*, *folder*, *run*)

var(***name***, *database*, *folder*, *run*, *option*)

Arguments

- ***name***

  (Required) Specifies the scalar to access.

- *database*

  (Optional) Specifies the source database name.

- *folder*

  (Optional) Specifies the source folder in the database.

- *run*

  (Optional) Specifies the run-id for a multiple run database. Default is 'ALL'.

- *option*

  (Optional) Specifies the output type. 'VALUE' indicates scalar values, 'WF' indicates a waveform (setting values against run variables). Default is 'VALUE'.

Return Values

Object. The accessed value.

Description

var(***name***) looks in the last open or active database for a scalar named ***name***. It reports an error if multiple entries are found.

var(***name***, *database*, *folder*) looks in the *folder* in the *database* for a scalar named ***name***.

var(***name***, *database*, *folder*, *run*) looks in the *folder* in the *database* for a scalar named ***name*** in *run*.

var(***name***, *database*, *folder*, *run*, *option*) looks in the *folder* in the *database* for a scalar named ***name*** in *run*. Returns value or waveform object(s) according to *option*.

wavevswave

Waveform Calculator Miscellaneous function.

Creates a new waveform using the Y values from x_wf as the x axis and the Y values from y_wf as y axis.

Usage

wavevswave(*x_wf*, *y_wf*)

Arguments

- ***x_wf***

  (Required) Specifies the input waveform name. Uses the Y values as the x axis for the new waveform.

- ***y_wf***

  (Required) Specifies the input waveform name. Uses the Y values as the y axis for the new waveform.

Return Values

Waveform. The waveform built from the input waveforms.

Description

Creates a new waveform, taking X values from the first argument waveform, and Y values from the second argument waveform. It performs the same operation as the **Set as X Axis** right-click menu item described in "Creating an XY Plot" on page 130.

Related Topics

Creating an XY Plot

wf

Waveform Calculator Special function.

Enables you to reference a waveform, in the context of the waveform calculator.

Usage

wf(<waveform_full_qualified_name>)

wf(<object_name>, <extended_option_set>)

Arguments

- <waveform_full_qualified_name>

  A waveform fully qualified name. It can take the form:

  ```
  "<database_name>waveform_name" or "<database/folder>waveform_name"
  ```

  For example:

  ```
  <meas/TRAN>V(IN)
  ```

- <object_name>

  An object name, with or without the database identifier. For example:

  ```
  'meas/Q0'
  ```

- <extended_option_set>

  A set of options to define precisely which object is designated.

  Note

  The combination of <object_name> and <extended_option_set> should reference a single waveform only. EZwave returns an error if this is not the case.

  See also "Supported Net Representation Components" on page 1245 and "Extended Options for Selecting Waveforms" on page 1246.

Description

The function wf() enables you to reference a waveform, in the context of the waveform calculator. It can be obtained by selecting a waveform in EZwave, and select **Insert selected waveforms** in the waveform calculator.

Note

This Waveform Calculator Special function does not appear in the Waveform Calculator **Functions** list but may be used for scripting.

Examples

```
wf("<meas/TRAN>V(IN)")*wf("<meas/TRAN>V(Q0)")
```

```
wf("<meas/TRAN>V(IN)")*3

wf("meas/IN", "-show TRAN.V")*wf("meas/Q0", "-show TRAN.V")

wf("meas/IN", "-show TRAN.V")*3
```

Tcl Waveform Calculator Batch Commands Example:

This example illustrates how to use both **wfc** and **wf** commands. The goal is to call a waveform calculator function on a specific waveform.

Here we are going to iterate through all the transient waveforms of a database to count them and retrieve their number of points.

```
## Retrieve the list of all transient waveforms
set wflist [find -show tran.all -r *]
set wfcount 0

## Iterate through the waveforms to count their points
foreach thiswf $wflist {
        foreach wave $thiswf {
               ## Use wf command to designate the waveform object on which
size should be called
                set nb_points [wfc "size(wf(\"$wave\"))"]
                puts "Number of points for waveform $wave is: $nb_points"
                incr wfcount
        }
}
puts "Worked on $wfcount waveforms"
puts "DONE"
```

wfattributes

Waveform Calculator Miscellaneous function.

Returns a waveform built from the input waveform with modified units, scales or display type.

Usage

wfattributes(***wf***[, *x_unit*, *y_unit*, *x_scale*, *y_scale*, *wf_type*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.
- *x_unit*

  (Optional) Specifies the new unit name for the x axis. Default is ‘Default’.
- *y_unit*

  (Optional) Specifies the new unit name for the y axis. Default is ‘Default’.
- *x_scale*

  (Optional) Specifies the new scale for the x axis. Specify ‘Linear’, ‘Log10’ or ‘Log2’. Default is ‘Default’.
- *y_scale*

  (Optional) Specifies the new scale for the y axis. Specify ‘Linear’, ‘Log10’ or ‘Log2’. Default is ‘Default’.
- *wf_type*

  (Optional) Specifies the new waveform display type. Specify ‘Continuous’, ‘Sampled’, ‘Scattered’, ‘Spectral’, or ‘Railroad’. Default is ‘Default’.

Return Values

Waveform. A waveform built from the input waveform and the specified attributes.

wfname

Waveform Calculator Miscellaneous function.

Returns the waveform name, or a specified part of the name, as a string.

Usage

wfname(***wf***, *option*)

Arguments

- ***wf***

  (Required) A waveform.

- *option*

  (Optional) Specifies the part of the waveform name to return. Valid values are ‘Full’, ‘Database’, ‘Folder’, ‘Path’, and ‘Object’. Default is ‘Default’ (full waveform name).

Return Values

String. The waveform name, or a specified part of the name.

wftoascii

Waveform Calculator Miscellaneous function.

Dumps the input waveform in a text file at the specified path location.

Usage

wftoascii(***path***, ***wf***[, *x_start*, *x_end*])

Arguments

- ***path***

  (Required) Specifies the output file path.

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Text file.

wftodata

Waveform Calculator Miscellaneous function.

Returns an array with the data points of the input waveform.

Usage

wftodata(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Array. An array with the data points of the input waveform.

windavg

Waveform Calculator Statistical function.

Returns an average value for each x value of the input waveform.

Usage

windavg(*wf*, *window_size*[, *window_direction*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***window_size***

  (Required) Specifies the window size value.

- *window_direction*

  (Optional) Specifies the window direction. Legal values are 'backward', 'center', or 'forward'. If this is not specified, the default is 'backward'.

Return Values

Waveform. The average value for each x value of the input waveform.

Description

The window average function returns an average value for each x value of the input waveform. The following describe the different *window_direction* options:

- 'backward' — (Default) If the current x value is less than ***window_size*** value, then average will be performed from 0 to current x value, else average will be performed from current x value minus ***window_size*** value to current x value.
- 'center' — If current x value is less than ***window_size*** value divided by 2, then average will be performed from 0 to current x value plus ***window_size*** value divided by 2. Else if current x value is greater than last x value minus ***window_size*** value divided by 2, then average will be performed from current x value minus ***window_size*** value divided by 2 to last x value. Else average will be performed from current x value minus ***window_size*** value divided by 2 to current x value plus ***window_size*** value divided by 2.
- 'forward' — If current x value is greater than last x value minus ***window_size*** value, then average will be performed from current x value to last x value. Else average will be performed from current x value to current x value plus ***window_size*** value.

window

Waveform Calculator Miscellaneous function.

Trims the original waveform and returns a new waveform with the specified lower and upper bounds.

Usage

window(***wf***, *x_start*, *x_end*)

window(***wf***, *x_start*)

window(***wf***, *x_end*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval. Default is ‘Begin’.

- *x_end*

  (Optional) Specifies the x value at the end of an interval. Default is ‘End’.

Return Values

Waveform. A new waveform with the specified lower and upper bounds.

Examples

To determine the time average of a waveform Vout from two to three nanoseconds, use the following:

```
Vavg = avg(window(Vout, 2e-9, 3e-9))
```

windowing

Waveform Calculator Signal Processing function.

Computes the windowing of the input waveform.

Usage

windowing(***wf***[, *t_start*, *t_stop, fs, points, sampling*, *padding*, *windowType*, *alpha*, *samplingEpsilon*, *windowShape*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *t_start*

  (Optional) Specifies the start time of the input waveform. Default is 'Begin'.

- *t_stop*

  (Optional) Specifies the stop time of the input waveform. Default is 'End'.

- *fs*

  (Optional) Specifies the sampling frequency of the signal. Default is 'Automatic'.

- *points*

  (Optional) Specifies the number of sampling points. Default is 'Automatic'.

  For symmetric windows, the parameters above satisfy the following equation:

  $((points)/fs) = t\_stop - t\_start$

  For periodic windows, the parameters above satisfy the following equation:

  $((points-1)/fs) = t\_stop - t\_start$

- *sampling*

  (Optional) Specifies the method of computing the sampled data. Legal values are 'No Sampling' (the default), 'Interpolation' or 'Spline'.

- *padding*

  (Optional) Activates data padding to pad the input data with zeros, before or after the input data set. Legal values: 'No Padding' (the default), 'Padding Right', 'Padding Left' and 'Padding Left and Right'. The input parameter is verified by the algorithm and changed if necessary.

- *windowType*

  (Optional) Applies a windowing function from a selection of windows. Legal values: 'Rectangular' (the default), 'Hamming', 'Hanning', 'Parzen', 'Welch', 'Blackman', 'Blackman-Harris', 'Bartlett', 'Kaiser', 'Klein' or 'Dolph Chebyshev'.

Note

For Hanning, symmetric window shapes are preferred when using a Hanning window in FIR filter design.

Periodic window shapes are preferred when using a Hanning window in spectral analysis. This is because the Discrete Fourier Transform assumes periodic extension of the input vector. A periodic Hanning window is obtained by constructing a symmetric window and removing the last sample.

- *alpha*

  (Optional) Specifies the alpha or beta value that is required by Hanning, Kaiser, and Dolph Chebyshev windows. Default is 'Default'.

- *samplingEpsilon*

  (Optional) When *sampling* is set to Interpolation, specifies that the input waveform data point (Y value) is to be used rather than the exact interpolated value when the X data of the input waveform is close to the computed X value.

  The computed X value corresponds to *time_start* +(*number_of_points* * dX) where dX is the sampling interval retrieved from the *sampling_frequency*.

  Defaults to the global setting in the "Waveform Calculator Calculation Options" on page 545.

- *windowShape*

  (Optional) Specifies the shape of the window. Legal values are Symmetric for standard FFT setup or Periodic for enhancing FFT setup for spectral analysis of periodic signals. Defaults to the global setting in the "Waveform Calculator Calculation Options" on page 545.

Tip

For comparing Eldo's FFT results with EZwave's FFT results, select the Periodic option, unless EZwave's FFT is executed on an FFT_INPUT waveform. In this case, Eldo has already considered the periodicity of the input signal.

Return Values

Waveform. The windowed waveform.

xcompress

Waveform Calculator RF function.

Extracts the x-axis value of the waveform at the point where the difference between the actual value of the wave and the linear extrapolation of the waveform based on the computed slope value becomes greater than ***val***.

Usage

xcompress(*wf*, *val*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.
- ***val***

  (Required) Specifies the Xcompress value.
- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.
- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double/array. The x-axis value corresponding to the compressed input waveform.

xdown

Waveform Calculator Miscellaneous function.

Returns all the x values where the input waveform falls below the given y level with a negative slope.

Usage

xdown(***wf***, ***at_y***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***at_y***

  (Required) Specifies the y value. Only the x values of the waveform falling below the y value with a negative slope are returned.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double/list. All the x values where the input waveform falls below the given y.

xnor

Waveform Calculator Logic function.

Applies the exclusive NOR function to two input waveforms whose data types are either bit or boolean.

Usage

xnor(***wf1***, ***wf2***)

Arguments

- ***wf1***

  (Required) Specifies the first input digital waveform name.

- ***wf2***

  (Required) Specifies the second input digital waveform name.

Return Values

Waveform. The resulting NOR waveform between the two input waveforms.

Description

The results of the NOR function are described in the following table:

Table A-15. XNOR Truth Table

wf1	wf2	xnor (wf1, wf2)
F	F	T
F	T	F
T	F	F
T	T	T

In the table, T represents TRUE for boolean waveforms, and 1 for bit waveforms. F represents FALSE for boolean waveforms, and 0 for bit waveforms.

xofmax

Waveform Calculator Mathematical function.

Returns all the x values at the maximum (or maxima) of a waveform.

Usage

xofmax(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double/list. The X value(s) at the maximum (or maxima) of the input waveform.

xofmin

Waveform Calculator Mathematical function.

Returns all the x values at the minimum (or minima) of a waveform.

Usage

xofmin(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double/list. The X value(s) at the minimum (or minima) of the input waveform.

xup

Waveform Calculator Miscellaneous function.

Returns all x values where the input waveform rises above the specified y level with a positive slope.

Usage

xup(*wf*, *at_y*[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***at_y***

  (Required) Specifies the y value. Only the x values of the waveform rising above the y value with a positive slope are returned.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double/list. All the x values where the input waveform rises above the given y.

xval

Waveform Calculator Miscellaneous function.

Returns all the x values at the specified y level of a waveform.

Usage

xval(***wf***, ***at_y***)

xval(***wf***, ***at_y***, *slope*)

xval(***wf***, ***at_y***, *x_start*, *x_end*)

xval(***wf***, ***at_y***, *slope*, *x_start*, *x_end*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***at_y***

  (Required) Specifies the y value at which to find the x value.

- *slope*

  (Optional) Specifies the type of slope of the input waveform. Results are only collected if the slope at the specified y level matches the specified slope. Values may be any of the following:

  - 'neg' — Negative slope. Only x values at the specified y level with a negative slope are returned.
  - 'pos' — Positive slope. Only x values at the specified y level with a positive slope are returned.
  - 'Either' — Any slope value. Returns the x value at the specified y level regardless of slope.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Double/list. All the x values corresponding to the given Y for the input waveform.

Description

If the slope is specified, returns only x values that matches the specified slope direction at the specified y level. Interpolation is applied.

xwave

Waveform Calculator Mathematical function.

Creates a new waveform with y values identical to the x values.

Usage

xwave(***wf***[, *x_start*, *x_end*])

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- *x_start*

  (Optional) Specifies the x value at the beginning of an interval.

- *x_end*

  (Optional) Specifies the x value at the end of an interval.

Return Values

Waveform. A waveform with x values identical to y values of the input waveform.

xytowf

Waveform Calculator Miscellaneous function.

Creates a new waveform using the Y values from wfX as the x axis and the Y values from wfY as y axis.

Usage

xytowf(*wfX*, *wfY*)

Arguments

- ***wfX***

  (Required) Specifies the input waveform name. Uses the Y values as the x axis for the new waveform. The Y values in the waveform must be increasing.

- ***wfY***

  (Required) Specifies the input waveform name. Uses the Y values as the y axis for the new waveform.

Return Values

Waveform. The waveform built from the input waveforms.

Description

Creates a new waveform, taking X values from the first argument waveform, and Y values from the second argument waveform. Both waveforms must have the same number of points.

Examples

You can stretch the x axis of a waveform. Use the Tcl procedure described here in a Tcl file. It performs the stretching of the x-axis data on a waveform xw obtained with xwave(). It reconstructs the final result waveform with the xytowf() function.

```
proc xstretch { wf k } {
    set xw [ wfc " xwave($wf)*$k " ]
    return [ wfc " xytowf($xw, $wf) " ]
}
```

You can then load the file *xstretch.tcl* into the Waveform Calculator (**File** > **Open Custom File**) and it will be listed in the **Functions** list under User Defined Functions. The first argument is the waveform to be stretched, the second argument is the stretch factor on x axis. You can then call the procedure in the following way in the Waveform Calculator:

```
mystretchedwaveform=xstretch(wf("v_load"), 0.95)
```

Related Topics

wavevswave

yval

Waveform Calculator Miscellaneous function.

Returns the y value at a given x-coordinate of a waveform.

Usage

yval(*wf*, *at_x*, *option*, *param*, *extrapolate*)

Arguments

- ***wf***

  (Required) Specifies the input waveform name.

- ***at_x***

  (Required) Specifies the x value at which to find the y value.

- *option*

  (Optional) Specifies the output type. Legal values for option:

  - 'VALUE' — Output is a numerical value. (Default).
  - 'WF' — Output is a waveform.
  - 'ANNOTATION' — Adds annotation to your input waveform.

- *param*

  (Optional) Used with *option* = 'WF'. Specifies the simulation parameter to be used to generate the result waveform. Default is 'index'.

- *extrapolate*

  (Optional) If set to 'Yes', the function extrapolates the y value if the specified x value is outside the waveform's x-axis range. If set to 'No', an error message is reported if the specified x value is outside the waveform's x-axis range. Default is 'No'.

Return Values

Double | complex | waveform. The y value of the input waveform at the specified x value.

Description

Returns the y value at a given x-coordinate of a waveform. Interpolation is applied.

Note

If multiple data points match the x value, **yval()** returns the y value corresponding to the last x data point.

Examples

In the following array of data points, 2 data points have the same X value and different Y ones:

```
[3[[0.0, 'False'], [1.0, 'True'], [11.0, 'False'], [21.0, 'True'],
[31.000000000000057, 'True'], [31.000000000000057, 'False']]
```

yval(wf*, 31)* will return False, the latest value for *x=31*. (Prior to AMS12.1, the value returned would have been True).

Appendix B
Eldo Simulation

The Eldo simulator outputs waveform data that can be displayed by EZwave. This section covers the different scenarios where EZwave works with the Eldo Simulator.

Run Eldo With EZwave **981**

Complete Eldo Simulation and View Simulation Data Later **982**

Manual Status Update **983**

Marching Update **984**

Joint Waveform Database Read API **984**

AMS Results Browser **985**

Run Eldo With EZwave

In this scenario, the Eldo simulator runs a complete simulation and outputs the data in JWDB format to be directly viewed by EZwave.

Procedure

1. You can use one of the following methods:

    - Invoke Eldo Simulator from the command line as in the following example:

      ```
      eldo test.cir -ezwave &
      ```

      This command invokes the Eldo simulator and directs it to run a complete simulation and output the data to a file.

      The simulator regularly saves incremental data to the disk (by default, each 100Mbs of data). This enables you to run very large simulations without consuming too much memory.

    - Use the -noisaving option to disable incremental saves inside the Eldo simulator.
    - Use a pre-defined configuration:

      ```
      eldo test.cir -ezwave -wdb_config config.swd &
      ```

2. The simulator requests that EZwave display waveforms as defined in the *config.swd* file (this is an EZwave Save Window file) instead of the .PLOT statements defined in the netlist *test.cir*. If some post-processed waveforms were stored through *config.swd*, they are automatically re-computed with new simulation data.

3. When the simulation is completed, the simulator exits and EZwave remains until you exit the program.

Complete Eldo Simulation and View Simulation Data Later

In this scenario, the Eldo Simulator runs a complete simulation and outputs the data in JWDB format to be read by EZwave. In EZwave, the data can be organized and the window contents can then be saved for later viewing.

Procedure

1. Invoke Eldo Simulator as in the following example:

   ```
   eldo test.cir
   ```

   This command invokes the Eldo simulator and directs it to run a complete simulation and output the data to a file.

   Alternatively, if you want to reuse the JWDB server launched by Eldo for other Eldo simulations, use the -jwdb_servermode option as in the following example:

   ```
   eldo test.cir -jwdb_servermode
   ```

   This option specifies that the JWDB server launched by Eldo can be re-used by other simulations. Useful data is stored in a file pointed to by the environment variable AMS_WDBSERVER_INFO, which by default is located in *$HOME/.ezwave/jwdbserver.info*. To modify the default:

   - Set the environment variable AMS_USE_ENV:

     ```
     setenv AMS_USE_ENV 1
     ```

   - AMS_WDBSERVER_INFO must point to a writable file:

     ```
     setenv AMS_WDBSERVER_INFO /my/writable/folder/jwdbserver.info
     ```

   Note
   JWDB Server mode can also be specified in the *eldo.ini* file.

   In this mode, the JWDB server will exit after the time specified by the environment variable AMS_WDBSERVER_TIMEOUT (if Eldo is not using it). Its default value is 60 minutes. To modify this value:

   - Set the environment variable, AMS_USE_ENV:

     ```
     setenv AMS_USE_ENV 1
     ```

   - Set AMS_WDBSERVER_TIMEOUT to the required value in minutes:

     ```
     setenv AMS_WDBSERVER_TIMEOUT 30
     ```

2. To display simulation results, invoke EZwave.
3. In EZwave, use the **File** > **Open** menu to open the JWDB file generated by the simulator. The waveform data displays.

   You can organize data in different graph rows and create some post processing waveforms. You can save the window contents using the **File** > **Save**option and then reuse the “saved window” *.swd* file later on.

Note

For this release, the following limitations apply for this scenario:

- The post-processed waveform is not automatically updated during the simulation.
- If EZwave still displays data at the end of the simulation, the data is “reloaded” from the disk. This can be time-consuming.
- Having EZwave display multiple-run simulation results may lead to internal errors.

Manual Status Update

Waveform data can be manually collected from a running simulation at an interval of your own choosing. This enables you to get a status update on a running simulation.

Procedure

1. Invoke Eldo Simulator using one of the following methods:

   Command line invocation: Invoke with EZwave using the following command:

   ```
   eldo test.cir -ezwave &
   ```

   Output in JWDB format: Invoke the Eldo Simulator and run a complete simulation as in the following example:

   ```
   eldo test.cir &
   ```

   Then invoke EZwave and use the **File** > **Open** option to open the *.wdb* file generated by the simulation.

2. To update the data in the EZwave viewer, click the Update Waveform Data button in the EZwave toolbar. This updates displayed waveforms with new simulation data.

Note

For this release, the following limitation applies for this scenario:

If jwdb_servermode is set (from the command line or in the *eldo.ini* file) when an Eldo simulation is invoked, the simulation output data cannot be accessed until after the simulation completes.

Marching Update

Waveform data can be collected from a running simulation at a pre-defined set interval. This interval is set in EZwave and is run simultaneously with the Eldo Simulator. This automates the process of updating waveform data viewed in EZwave.

Procedure

1. Invoke Eldo Simulator using one of the following methods:

   Command line invocation: Invoke with EZwave with the following command:

   ```
   eldo test.cir -ezwave &
   ```

   Output in JWDB format: Invoke the Eldo Simulator and run a complete simulation as in the following example:

   ```
   eldo test.cir &
   ```

   Then invoke EZwave and use the **File** > **Open** option to open the *.wdb* file generated by the simulation.

2. In EZwave, select **Edit** > **Options** to invoke the EZwave Display Preferences Dialog Box and click General to view the General Options page.

3. In the Marching Waveforms section, set the update interval by either of the following options:

   - Automatically Update Displayed Waveforms Every X time interval: The time interval can be by second, minute, or hour.

   - Automatically Update Displayed Waveforms Every X% of Simulation: This updates based on the percentage completion of the simulation.

   Be careful not to set too small of an interval. Setting a short interval increase the number of updates and then the amount of resources globally used to update the waveform data viewed in EZwave.

   Note

   For this release, the following limitation applies for this scenario:

   If jwdb_servermode is set (from command line or in the *eldo.ini* file) when an Eldo simulation is invoked, the simulation output data cannot be accessed until after the simulation completes.

Joint Waveform Database Read API

You can use the Read API to access the Joint Waveform DataBase (*.wdb* file) generated by Eldo and used by EZwave to obtain waveform data.

This Read API is documented in the *Joint Waveform Database Read API Reference Manual*.

AMS Results Browser

In addition to using EZwave, you can also view and analyze Eldo simulation results and associated files in tabular form using the AMS Results Browser.

Refer to the *AMS Results Browser User's Manual*.

Appendix C
Tcl Scripting Support

EZwave supports Tcl scripting, enabling you to create batch files to execute Tcl commands from within the EZwave viewer.

Tcl Syntax **991**
Tcl Scripting Example 991
Tcl Command Syntax Rules 991
If Command Syntax 995
Set Command Syntax 996
Command Substitution 996
Variable Substitution 997
Tcl References 997

Passing Tcl Parameters From the Command Line **999**

Tcl List Processing **1000**

Supported Tcl Commands **1003**
Tcl Syntax Conventions 1003
Tcl Command Short Descriptions 1004
Specifying Waveforms in Tcl 1012
Selecting Waveforms in Tcl 1014

Tcl Command Detailed Descriptions **1018**
add wave 1025
add workspace 1032
batch_mode 1033
bloc 1034
compare add 1035
compare clock 1040
compare configure 1042
compare end 1046
compare info 1047
compare list 1048
compare options 1049
compare run 1055
compare savelog 1056
compare saverules 1057
compare start 1058
dataset alias 1060
dataset analysis 1061
dataset clear 1062
dataset close 1063

dataset extract 1064
dataset info 1065
dataset list 1066
dataset merge 1067
dataset mergewaveforms. 1069
dataset open. 1071
dataset ovd 1072
dataset power analysis 1073
dataset rename. 1076
dataset save 1077
dataset savescalar 1079
dataset savewaveforms 1080
dataset scalar 1082
dataset statistics. 1084
dataset supported. 1086
dataset topnoise. 1087
delete wave 1089
dofile 1090
environment 1091
evalExpression 1092
examine 1093
exit 1095
find analogs 1096
find currents 1097
find digitals 1098
find nets | signals. 1099
getactivecursortime 1100
precision 1101
printenv 1102
quit 1103
radix 1104
radix define 1106
radix delete 1108
radix list 1109
radix names 1110
radix signal 1111
save 1112
setenv 1113
unsetenv 1114
wave activecursor 1115
wave activewindow. 1116
wave activeworkspace 1117
wave addannotation 1118
wave addaxisdeltamarker 1120
wave addcursor 1122

wave adddeltamarker 1124
wave addline 1126
wave addmarker 1128
wave addproperty 1129
wave addwindow 1130
wave addworkspace 1131
wave cdf 1132
wave clean. 1136
wave closewindow 1137
wave closeworkspace 1138
wave colortheme 1139
wave compressvri 1140
wave createbus 1142
wave cursortime 1143
wave deletecursor 1144
wave difference 1145
wave displayed 1146
wave exists 1147
wave ezwave_title 1149
wave gettype 1150
wave histogram 1151
wave ipnvri 1155
wave jitter 1158
wave launchfolder 1163
wave listworkspace 1164
wave loadbindings 1165
wave lockcursor 1166
wave names 1167
wave plotcompress 1168
wave plotpacipn 1170
wave refresh 1172
wave rowfit 1173
wave runindexlist 1174
wave runparameters 1175
wave runparametervalue 1176
wave show. 1177
wave showgridlines. 1178
wave showzerolevels 1179
wave tandem mode 1180
wave tile 1181
wave windowdecoration 1182
wave windowlist 1183
wave xaxis. 1184
wave yaxis. 1185
wave zoomfull. 1187

wave zoomin 1188
wave zoomlast. 1189
wave zoomout 1190
wave zoomrange 1191
wfc 1192
write jpeg 1194
write png 1195
write wave. 1196

External Tcl Command Support 1197

Tcl Scripting Examples 1205
Tcl Waveform Calculator Batch Commands Example. 1206
Tcl User-Defined Functions 1211
Waveform Comparison With Tcl Examples. 1213
Using find Commands 1220

Waveform Calculator Example Tcl Scripts 1224
adc_sndr 1226
adc_sndr_parallel4 1227
clock_jitter 1228
compressvri. 1229
IPnVRI 1230
pll_jitter. 1233
pll_jitter_parallel4. 1235
pll_phasenoise. 1236
pll_phasenoise_parallel4. 1238
plotcompress 1240
plotpacipn 1242

Tcl Syntax

This section contains information about Tcl syntax.

Tcl Scripting Example	A short example of how to use Tcl commands with EZwave.
Tcl Command Syntax Rules	Lists eleven rules that define the syntax and semantics of the Tcl language.
If Command Syntax	Describes in detail the syntax for the Tcl command, **if**.
Set Command Syntax	Describes in detail the syntax for the Tcl command, **set**.
Command Substitution	Describes command substitution in Tcl.
Variable Substitution	Describes variable substitution in Tcl.
Tcl References	Lists some books and websites on the Tcl language.

Tcl Scripting Example

This topic describes a short example of how to use Tcl commands with EZwave.

Procedure

1. Create a text file *test.tcl* and copy in the following lines:

```
## open the tutorial
dataset open $env(MGC_AMS_HOME)/examples/ezwave/tutorial.wdb
## perform a calculation
set wf [wfc {wf("<tutorial/Time-Domain_Results>v_load")\
-wf("<tutorial/Time-Domain_Results>v_middle")}]
## plot the result add wave
add wave $wf
## print window contents to a PostScript file
write wave -file $env(HOME)/test.ps
## then exit
exit
```

2. Run the following command in your terminal:

```
ezwave -do test.tcl
```

This command invokes EZwave and loads the Tcl commands contained in the *test.tcl* file.

Tcl Command Syntax Rules

Eleven rules define the syntax and semantics of the Tcl language.

1. A Tcl script is a string containing one or more commands. Semi-colons and newlines are command separators unless quoted. Close brackets (**]**) are command terminators during command substitution unless quoted.
2. A command is evaluated in two steps. First, the Tcl interpreter breaks the command into words and performs substitutions. These substitutions are performed in the same way for all commands. The first word is used to locate a command procedure to carry out the command, then all of the words of the command are passed to the command procedure. Different commands interpret their words differently.
3. Words of a command are separated by white space (except for newlines, which are command separators).
4. If the first character of a word is a double-quote (**"**) then the word is terminated by the next double-quote character. If semi-colons (**;**), close brackets (**]**), or white space characters (including newlines) appear between the quotes then they are treated as ordinary characters and included in the word. Command substitution, variable substitution, and backslash substitution are performed on the characters between the quotes. The double-quotes are not retained as part of the word.
5. If the first character of a word is an open brace (**{**) then the word is terminated by the matching close brace (**}**). Braces nest within the word: for each additional open brace there must be an additional close brace. However, if an open brace or close brace within the word is quoted with a backslash then it is not counted in locating the matching close brace. No substitutions are performed on the characters between the braces except for backslash-newline substitutions, nor do semi-colons, newlines, close brackets, or white space receive any special interpretation. The word will consist of exactly the characters between the outer braces, not including the braces themselves.
6. If a word contains an open bracket (**[**) then Tcl performs command substitution. To do this it invokes the Tcl interpreter recursively to process the characters following the open bracket as a Tcl script. The script may contain any number of commands and must be terminated by a close bracket (**]**). The result of the script (the result of its last command) is substituted into the word in place of the brackets and all of the characters between them. There may be any number of command substitutions in a single word. Command substitution is not performed on words enclosed in braces.
7. If a word contains a dollar-sign (**$**) then Tcl performs variable substitution: the dollar-sign and the following characters are replaced in the word by the value of a variable. Variable substitution may take any of the following forms:
   - $**name**: Name is the name of a scalar variable; the name is terminated by any character that isn't a letter, digit, or underscore (**_**).
   - $*name*(*index*): Name gives the name of an array variable and index gives the name of an element within that array. Name must contain only letters, digits, and underscores. Command substitutions, variable substitutions, and backslash substitutions are performed on the characters of index.

- ${*name*}: Name is the name of a scalar variable. It may contain any characters whatsoever except for close braces. There may be any number of variable substitutions in a single word. Variable substitution is not performed on words enclosed in braces.

8. If a backslash (\ ) exists within a word then backslash substitution occurs. In all cases, except those described in the table, the backslash is dropped and the following character is treated as an ordinary character and included in the word. This enables characters such as double quotes, close brackets, and dollar signs to be included in words without triggering special processing. The following table lists the backslash sequences that are handled specially, along with the value that replaces each sequence. Backslash substitution is not performed on words enclosed in braces, except for backslash-newline.

Table C-1. Tcl Backslash Sequences

Sequence	Value
\a	Audible alert (bell) (0x7)
\b	Backspace (0x8)
\f	Form feed (0xc)
\n	Newline (0xa)
\r	Carriage-return (0xd)
\t	Tab (0x9)
\v	Vertical tab (0xb)

\<newline>*white_space*	A single space character replaces the backslash, newline, and all spaces and tabs after the newline. This backslash sequence is unique in that it is replaced in a separate prepass before the command is actually parsed. This means that it will be replaced even when it occurs between braces, and the resulting space is treated as a word separator if it isn't in braces or quotes.
	Example: A backslash at the end of a line of text causes the interpreter to ignore the newline, and treat the text as a single line of text. The interpreter inserts a blank space at the location of the ending backslash.
	`puts "This string comes out\`
	`on a single line"`

\\	Backslash (\ )
ooo	The digits *ooo* (one, two, or three of them) give the octal value of the character.
\x*hh*	The hexadecimal digits *hh* give the hexadecimal value of the character. Any number of digits may be present.

9. If a pound sign (#) exists at a point where Tcl is expecting the first character of the first word of a command, then the pound sign and the characters that follow it, up through the next newline, are treated as a comment and ignored. The # character denotes a comment only when it exists at the beginning of a command.

10. Each character is processed exactly once by the Tcl interpreter as part of creating the words of a command. For example, if variable substitution occurs then no further substitutions are performed on the value of the variable; the value is inserted into the word verbatim. If command substitution occurs then the nested command is processed entirely by the recursive call to the Tcl interpreter; no substitutions are performed before making the recursive call and no additional substitutions are performed on the result of the nested script.

11. Substitutions do not affect the word boundaries of a command. For example, during variable substitution the entire value of the variable becomes part of a single word, even if the variable's value contains spaces.

If Command Syntax

The Tcl if command executes scripts conditionally. The question mark (**?**) indicates an optional argument.

Syntax

if *expr1* ?then? *body1* elseif *expr2* ?then? *body2* elseif … ?else? ?*bodyN*?

Parameters

- The if command evaluates *expr1* as an expression.

  The value of the expression must be a boolean (a numeric value, where 0 is false and anything else is true, or a string value such as “true” or “yes” for true and “false” or “no” for false); if it is true then *body1* is executed by passing it to the Tcl interpreter. Otherwise *expr2* is evaluated as an expression and if it is true then *body2* is executed, and so on. If none of the expressions evaluates to true then *bodyN* is executed. The “then” and “else” arguments are optional “noise words” to make the command easier to read. There may be any number of “elseif” clauses, including zero. *bodyN* may also be omitted as long as else is omitted too. The return value from the command is the result of the body script that was executed, or an empty string if none of the expressions was non-zero and there was no *bodyN*.

Set Command Syntax

The Tcl set command returns or sets the values of variables.

Syntax

set *varName* [*value*]

Parameters

- ***varName***

  (Required) The name of a Tcl variable. If you do not specify a *value*, this command will return the value of the *varName* you specify.

- *value*

  (Optional) The value to be assigned to the variable. When you specify *value* you will change the current state of the *varName* you specify.

Description

Returns the value of variable *varName*. If you specify *value*, the command sets the value of *varName* to value, creating a new variable if one does not already exist, and returns its value. If *varName* contains an open parenthesis and ends with a close parenthesis, then it refers to an array element: the characters before the first open parenthesis are the name of the array, and the characters between the parentheses are the index within the array. Otherwise *varName* refers to a scalar variable. Normally, *varName* is unqualified (does not include the names of any containing namespaces), and the variable of that name in the current namespace is read or written. If *varName* includes namespace qualifiers (in the array name if it refers to an array element), the variable in the specified namespace is read or written.

If no procedure is active, then *varName* refers to a namespace variable (global variable if the current namespace is the global namespace). If a procedure is active, then *varName* refers to a parameter or local variable of the procedure unless the global command was invoked to declare *varName* to be global, or unless a Tcl variable command was invoked to declare *varName* to be a namespace variable.

Command Substitution

Placing a command in square brackets (**[ ]**) will cause that command to be evaluated first and its results returned in place of the command.

For example:

```
set a 25
set b 11
set c 3
echo "the result is [expr ($a + $b)/$c]"
```

This code will output the following:

```
the result is 12
```

Variable Substitution

When a $*var_name* is encountered, the Tcl parser will replace it with the value defined by the variable.

Note

Tcl is case sensitive for variable names.

To access environment variables, use the following construct:

```
$env(var_name)
```

```
echo My user name is $env(USER)
```

Environment variables can also be set using the **env** array:

```
set env(SHELL) /bin/csh
```

Tcl References

You can take advantage of the many excellent books and websites on the Tcl language. The list that follows is provided to give you a place to start in your search for the reference materials that work best for you.

Note

It is not an endorsement of any book or website. While every attempt has been made to ensure that the URLs listed here point to active websites, there is no guarantee of this.

Books

- *Practical Programming in Tcl and Tk*

  Brent B. Welch, Jeffrey Hobbs, Brent Welch, 4th Edition, Prentice Hall PTR (2003)

- *Tcl/Tk in a Nutshell*

  Paul Raines, Jeff Tranter, O’Reilly and Associates, Inc. (1999)

- *Tcl and the Tk Toolkit*

  John K. Ousterhout, Addison-Wesley Professional (1994)

- *Tcl/Tk Tools*

  Mark Harrison, O'Reilly (1997)

Websites

- Tcl Developer Xchange

  `http://www.tcl.tk/doc/`
- Tcl Tutor

  `http://www.msen.com/~clif/TclTutor.html`
- Beginning Tcl

  `http://wiki.tcl.tk/298`
- The Tcler's Wiki

  `http://wiki.tcl.tk`
- ActiveTcl

  `https://www.activestate.com/products/tcl/`

Passing Tcl Parameters From the Command Line

This topic describes how to pass simple parameters from the command line to a Tcl script running in EZwave batch mode or GUI mode.

Syntax

ezwave [-c] [-i <inputFile>] [-o <outputFile>] [-s name [,name]]
[-assign <param>=<value>] [-do <tclScript.tcl>] [-args <arguments>]

Arguments

- -c

  (Optional) Runs EZwave in batch mode. There is no graphical user interface or display. Refer to "Invoking EZwave in Batch Mode" on page 45.

- -i

  (Optional) Specifies the input simulation results file or directory name. For PSF format data, this may be the psf directory.

- -o

  (Optional) Specifies the output file name for the measurement results generated by the Tcl script. The default is standard output. This parameter affects the save() and wfc -save functions:

  When not set:

  Writes to the current OVD iteration if available, otherwise it saves scalars to standard output and waveforms are silently ignored.

  When set:

  Does not write to the current OVD iteration (even if it is available), but writes scalars to *<outputFile>* and waveforms to *<outputFile>.wdb* (removing suffix *.txt*, *.log*, *.out* if *<filename>* ends with such a suffix). When the file already exists, it is replaced without notification.

- -s

  A comma-separated list of Waveform Calculator real or complex scalar variables, short vector variables, or waveform names for which to print out measurements. Mentor does not recommend using this for long variables, because the output gets written to a single line and may be difficult to read.

- -assign

  (Optional) Assigns a Tcl variable to a parameter.

- -do

  (Optional) Specifies the name of a predefined or user-defined Tcl script.

- -args

  (Optional) Specifies a list of arguments for the Tcl script.

Description

You can pass simple parameters from the command line to a Tcl script and run it.

Examples

The command line:

```
ezwave   -i databasename   -assign param1=value1   -assign param2=value2
-do tclScript.tcl   -args  arg1 arg2 arg3
```

executes the following actions:

- Parses the command line parameters until -args <arguments> is reached (it ignores any additional EZwave command line arguments that follow)
- Opens the specified database
- Sets a parameter value for each -assign assignment declaration
- Runs the Tcl script (*tclScript.tcl*) with the specified arguments (arg1 arg2 arg3).

Related Topics

Invoking EZwave in Batch Mode

batch_mode

save

wfc

Tcl List Processing

In Tcl a "list" is a set of strings in braces separated by spaces. Several Tcl commands are available for creating lists, indexing into lists, appending to lists, getting the length of lists and shifting lists.

Tcl List Processing Commands

Table C-2. Tcl List Processing Commands

Command	Syntax	Description
lappend	**lappend** *var_name val1 val2 …*	Appends *val1*, *val2* and so on, to the list *var_name*
lindex	**lindex** *list_name index*	Returns the *index*-th element of *list_name*; the first element is 0

Table C-2. Tcl List Processing Commands (cont.)

Command	Syntax	Description
linsert	**linsert** *list_name index val1 val2 …*	Inserts *val1*, *val2* and so on, just before the *index*-th element of *list_name*
list	**list** *val1, val2 …*	Returns a Tcl list consisting of *val1*, *val2*, ...
llength	**llength** *list_name*	Returns the number of elements in *list_name*
lrange	**lrange** *list_name first last*	Returns a sublist of *list_name*, from index *first* to index *last*; *first* or *last* may be End, which refers to the last element in the list
lreplace	**lreplace** *list_name first last val1, val2, …*	Replaces elements *first* through *last* with *val1*, *val2*, ...
lsearch	**lsearch** *list_name pattern*	Returns the index of the first element in *list_name* that matches *pattern*
lsort	**lsort** *list_name*	Returns *list_name* sorted in increasing ASCII sorting order

Examples

Example 1

This example uses the Tcl while loop to copy a list from variable *a* to variable *b*, reversing the order of the elements:

```
set b [list]
set i [expr {[llength $a] - 1}]
while {$i >= 0} {
   lappend b [lindex $a $i]
   incr i -1
}
```

Example 2

This example uses the Tcl for command to copy a list from variable *a* to variable *b*, reversing the order of the elements:

```
set b [list]
for {set i [expr {[llength $a] - 1}]} {$i >= 0} {incr i -1} {
   lappend b [lindex $a $i]
}
```

Example 3

This example uses the Tcl foreach command to copy a list from variable *a* to variable *b*, reversing the order of the elements. (The foreach command iterates over all of the elements of a list.):

```
set b [list]
foreach i $a { set b [linsert $b 0 $i] }
```

Example 4

This example shows a list reversal as above that aborts on a particular element using the Tcl break command:

```
set b [list]
foreach i $a {
if {$i = "ZZZ"} break
set b [linsert $b 0 $i]
}
```

Example 5

This example is a list reversal that skips a particular element by using the Tcl continue command:

```
set b [list]
foreach i $a {
if {$i = "ZZZ"} continue
set b [linsert $b 0 $i]
}
```

Supported Tcl Commands

This section describes the supported Tcl commands.

Table C-3. Tcl Command Topics

Tcl Syntax Conventions	Describes the conventions used in the short and detailed descriptions of the commands.
Tcl Command Short Descriptions	A quick look at the supported Tcl commands and their short descriptions. Organized in table format and by category.
Specifying Waveforms in Tcl	Details the different ways you can specify one or many waveforms in Tcl.
Selecting Waveforms in Tcl	Details arguments that are used by more than one command.

The EZwave waveform viewer supports the Tcl commands listed in Table C-4 on page 1004.

Tcl Syntax Conventions

This section describes the syntax conventions used in the Tcl command descriptions.

Tcl command descriptions use the following conventions:

- **Required**

  Required command names or arguments are in **bold**.

- ***Required Replaced***

  Required argument place holders are in ***bold italics***. You must replace these arguments with a value or string. For example,

  dofile ***filename***

  indicates that you must replace the required ***filename*** argument with the name of a file.

- [Optional]

  Optional arguments are surrounded by square brackets [ ].

- [*Optional Replace*]

  Optional argument place holders are surrounded by square brackets [ ] and in italics. When using these optional arguments, you must replace it with a value or string. For example,

  compare run [*startTime*]

indicates that *startTime* is an optional argument, but you must replace it with a start time value.

- { a | b }

  Braces { } indicate a choice between the items separated by a pipe |. There can be more than two choices for a set of braces.

Tcl Command Short Descriptions

This section lists the Tcl commands and their short descriptions. In addition, EZwave recognizes Questa SIM and Questa ADMS commands. Depending on the commands, these may either be ignored without error by the EZwave Tcl interpreter or executed by internally applying corresponding actions.

The specific responses are given in the following topics:

- Questa SIM Command Support
- Questa ADMS Command Support

Tip

For commands that require a *waveform_name* argument, you can use special options to select the waveform(s). These are indicated in the table by a footnote at the end of the table.

Note

In addition to the commands described here, some example Tcl scripts that provide equivalent functions to legacy AFS WaveCrave CalcPAD scripts are available in the EZwave examples directory. Refer to "Waveform Calculator Example Tcl Scripts" on page 1224.

Table C-4. Supported Tcl Commands

Command	**Description**
Miscellaneous Commands	
batch_mode	Check if the Tcl program is in batch mode.
bloc	Enables multi-line Waveform Calculator scripts.
dofile ***file_name***	Sources the specified Tcl file.
environment [*path_name*][-dataset | -nodataset]	Displays or changes the current region/ signal environment.
exit [--force] [-discard] quit [-force] [-discard]	Exits the EZwave viewer.

Table C-4. Supported Tcl Commands (cont.)

evalExpression **expression** wfc **expression**	Invoke the EZwave Waveform Calculator and perform operations. **evalExpression** and **wfc** may return different values, depending on the type of output. Refer to the entries on the following pages for more details.
precision [-double | -cursor][*precision_value#*]	Set cursor or double display precision.
write jpeg ***file_name*** [-window *window_name*][-header "dbTitle simuTitle wndTitle fromSimulator time date user machine"] [-blackbackground | -whitebackground | -monochrome | -colorasdisplayed] [-visiblewindows] [-resolution {screen |printerlow |printerhigh}] write png ***file_name*** [-window *window_name*] [-header "dbTitle simuTitle wndTitle fromSimulator time date user machine"] [-blackbackground | -whitebackground | -monochrome | -colorasdisplayed] [-visiblewindows] [-resolution {screen|printerlow|printerhigh}]	Exports the contents of a window as a PNG or JPEG image file.
wave loadbindings ***file.ezkey***	Loads an EZwave keyboard shortcuts schema *.*ezkey* file.
Environment Variable Commands	
printenv [*variable_name*]	Displays name and value of environment variable(s).
setenv ***variable_name*** [*value*]	Sets or displays the value of an environment variable.
unsetenv ***variable_name***	Unsets an environment variable.
Waveform Commands	
add wave (see add wave section for usage)	Add a new waveform.
delete wave [-window *window_name*] **waveform_name**[1].	Removes a waveform.
wave addannotation [*waveform_name*[1]] **-x *x_value*** [-y *y_value*] **-text *text*** [-snap]	Add a text annotation at the specified location in the active window.
wave addaxisdeltamarker **-wf *waveform_name***[1] {**-xdelta** | **-ydelta**} **-X1 *xdata1* -X2** xdata2 [-text*text*]	Adds a delta marker, between two values, on the selected waveform. The marker can be horizontal or vertical.

Table C-4. Supported Tcl Commands (cont.)

wave adddeltamarker {**-xdelta** | **-ydelta**} **-wf1** ***waveform_name1***[1] **-x1** ***x_value1*** **-y1** ***y_value1*** [-wf2 *waveform_name2*] **-x2** ***x_value2*** **-y2** ***y_value2*** [-text*text*]	Adds a delta marker to waveform(s) in the active window.
wave addline {**-x** | **-y**} ***value*** [*waveform_name*] [-text *annotation_text*] [-drag] [-hash {up | down | left | right}]	Adds a horizontal or vertical line to a waveform in the active window.
wave addmarker [-window *window_name*] [-name *name*] [-time {*time* | *x_value*}]	Add a vertical marker at the specified location.
wave addproperty ***waveform_name***[1] **-name** ***property_name*** **-value** ***property_value***	Adds a property to a waveform.
wave addwindow [-title *window_title*] [-divider *ratio*] [-width *width* -height *height*] [-x *xPosition* -y *yPosition*]	Creates a new window.
wave closewindow [*window_name*] | [-all | -workspace]	Closes one or more graph windows.
wave closeworkspace *workspace_name* | all	Closes a specified workspace, or all workspaces.
wave difference ***waveform_name1***[1] ***waveform_name2*** [-name *out_waveform_name*] [-spice]	Generates and displays a waveform corresponding to the difference between the first and second waveforms specified.
wave displayed [-window *window_name* | -allwindows | -activewindow] [-color]	Displays a Tcl list containing waveform names and optionally, waveform colors.
wave exists ***waveform_name***[1]	Returns True or False to indicate whether at least one waveform matching the search string exists.
wave gettype ***waveform_name***[1]	Returns the type of the waveform, for example, real, current, compound.
wave launchfolder	Returns the path to the directory where EZwave was launched from
wave runindexlist ***waveform_name***[1]	Returns a list of run indices for the elements of the compound waveform specified. Useful for iterating through each element individually.
wave runparameters ***waveform_name***[1]	Returns a list of run parameters for the compound waveform specified.
wave runparametervalue **-param** ***param_name*** **-run** ***run_index waveform_name***	Returns the value of the specified run parameter associated with the run index and compound waveform specified.

Table C-4. Supported Tcl Commands (cont.)

wave show *representation* [|*representation2*| ... | *representationN*]	Defines the default net representation that is used with the add wave command.
wave windowlist	Returns a list of open windows.
wave xaxis [-title *title*] [-scale {linear | log2 | log10}] [-window *window_name*]	Sets the x-axis title or scale.
wave yaxis **-row *rowId*** [-axis *axis_name*] [-title *title*] [-scale {linear | log2 | log10}] [-window *window_name*]	Sets the title or scale of the row. The y-axis name must be specified if there are multiple y axes.
write wave [*file*.ps | -file *file*.ps | -printer *printer_name*] [-window *window_name*] [-landscape | -portrait] [-papersize {a3 | a4 | a5 | b4 | b5 | letter | tabloid | ledger | legal | executive}] [-copies *number*] [-allwaveforms | -visiblewaveforms] [-activewindow | -allwindows | -visiblewindows] [-eps] [-header "dbTitle simuTitle wndTitle fromSimulator time date user machine"]	Print a waveform.
Find Commands	
find analogs ***object_name***[1] …	Returns names of all analog objects (quantities and terminals) matching the specified pattern.
find currents ***object_name***[1] …	Returns the names of all the current objects matching the specified pattern.
find digitals ***item_name***[1] …	Returns a list of digital signals matching the specified pattern.
find nets | signals ***item_name***[1] …	Returns the full pathnames of all analog and digital objects that match the name specifications.
Zoom Commands	
wave zoomfull [-window *window_name*]	Zoom out to show entire x axis.
wave zoomin [-window *window_name*] [*factor*]	Zoom in on x axis by specified factor (default: 2x).
wave zoomlast [-window *window_name*]	Undo most recent zoom operation.
wave zoomout [-window *window_name*] [*factor*]	Zoom out on x axis by specified factor (default: 2x)
wave zoomrange [-window *window_name*] [*start*] [*end*]	Zoom in on x xis to range specified by start and/or end poiants.
Cursor Commands	

Table C-4. Supported Tcl Commands (cont.)

getactivecursortime [-window *window_name*]	Return cursor location.
wave activecursor [-window *window_name*] [*cursor_name*]	Specify cursor to be active cursor.
wave addcursor [-window *window_name*] [-time *time* | *X_value*] [-name *cursor_name*]	Create a new cursor.
wave cursortime [-window *window_name*] [-time *time* | *X_value*] [*cursor_name*]	Move cursor to specified position, or return cursor location if position is not specified.
wave deletecursor [-window window_name] [cursor_name | cursor_id] [-all]	Delete the specified cursor, or all cursors, in a window.
wave lockcursor [on | off]	Determines whether the spacing between cursors is fixed.
Display Commands	
wave activewindow [-title window_name]	Returns the name of the currently active window, or sets a specified window to be active.
wave closeworkspace [workspace_name] | [-all]	Closes a specified workspace, or all workspaces.
wave colortheme [black | white | mono] [-print]	Sets the print or graphical display color scheme.
wave ezwave_title new_title	Adds a custom title to the top of the EZwave main display.
wave names [-showdataset {on | off}] [-showhierarchy {on | off}]	Determines whether plotted waveforms show dataset names and (or) complete hierarchical path
wave refresh [-window *window_name*]	Redraw window contents.
wave rowfit [-window *window_name*]	Optimizes row size in a window.
wave showgridlines [on | off]	Determines whether gridlines are displayed in newly created windows
wave showzerolevels {**on** | **off** | **horizontal** | **vertical**}	Determines how zero levels are activated or disabled on all new windows.
wave tandem mode [on | off]	Activates or deactivates Tandem Mode.
wave tile [-cascade | -horizontal | -vertical | -grid]	Arranges windows in one of the four tiling options.
wave windowdecoration [show | hide]	Shows or hides the decoration borders (title, buttons and scrollbars) of wave windows.
Radix Commands	

Table C-4. Supported Tcl Commands (cont.)

radix [-***radix_type*** | ***radix_name***]	Sets the bus radix to one of the following: • onescomplement • decimal • magnitude • binary • octal • hexadecimal • unsigned • ascii • fpoint *n* • tograydec • graytodec • thermo or the name of a user-defined radix.
radix define ***radix_name*** {***radix_value, radix_label*** [*radix_value2, radix_label2 … radix_valueN, radix_labelN*] [-default *radix_type*]}	Defines a new radix.
radix delete ***radix_name***	Removes the radix definition from the named radix.
radix names	Returns a list of all currently defined radix names.
radix list [*radix_name*]	Returns the definition of a radix if a name is supplied, the definition of all currently defined radices otherwise.
radix signal [*waveform_name* [*radix_value*]] [-fpoint *decimal*]	Sets or inspects radix values for the specified signals.
RF Commands	
wave compressvri	Computes the N^{th} compression point for the specified harmonic.
wave ipnvri	Computes the N^{th} order input-referred intercept point, output-referred intercept point, or both for the specified input compound waveform.
wave plotcompress	Plots output power vs input power in dBm and returns n-dB compression point.

Table C-4. Supported Tcl Commands (cont.)

wave plotpacipn	Takes a PAC spectrum signal pair at the desired and unwanted harmonics, within an input power sweep, plots the pair of IPn curves and returns the intercept point.
Dataset Commands	
dataset alias ***dataset_name*** [alias_name [-folder *folder_path*]]	Assigns aliases to datasets.
dataset analysis [*dataset_name*] [-folder *folder_path*]	Returns analysis list for the dataset.
dataset clear	Clear all waveform data from last opened database.
dataset close [*database_name* | -all]	Close specific database, or all databases.
dataset info ***option*** [*database_name*]	Fetch information about a database. ***option*** is one of: • **name** — Database name • **file** — Path- and filename where database is saved • **exists** — Whether database is currently open
dataset list [-long]	List open databases.
dataset merge ***dataset_name file_name1*** … ***file_namen*** [-matchindex | -startindex *start_index*] [-ignorewaveforms]	Merges databases.
dataset mergewaveforms ***dataset_name file_name1*** … ***file_namen*** [-matchindex | -startindex *start_index*] -wf ***waveform1_name***[1] … ***waveform_n_name***	Merges waveforms.
dataset open ***file_name*** [*dataset_name*] [-incl_if_tandem {*on* | *off*}]	Open a database file.
dataset rename ***database_name new_name***	Rename database.
dataset save ***database_name file_name*** [-start *x_start*] [-end *x_stop*] [-sampling *sample*] [-atcursors] [-delta *delta*] [-derivabs *derivabs*] [-derivrel*derivrel*]	Save database to disk.
dataset savescalar **<*wdb*> <*database*>** [<*folder*> ... <*folderN*>] **-output <*report_file*>** [-sort <*name*|*value*> <*asc*|*desc*>]	Save scalars, such as noise contributors, to a TXT file or CSV file.

Table C-4. Supported Tcl Commands (cont.)

dataset savewaveforms ***file_name*** [-start *x_start*] [-end *x_stop*] [-sampling *sample*] [-atcursors] [-delta *delta*] [-derivabs *derivabs*] [-derivrel*derivrel*] ***waveform_name***[1]	Save waveform(s) into the specified database file.
dataset scalar ***ref_dataset test_dataset*** **-output <*report_file*>** [-type <*name*>] [-cs *0*|*1*] [-nameonly|-tol <*value*>]	Compares scalar values of variables stored in two databases (PSF, nutmeg, FSDB, WDB, OVD).
dataset statistics [*dataset_name*] [-requestcompleteload]	Returns statistics on the dataset.
dataset supported ***dataset_path***	Returns "true" if the dataset can be opened by EZwave, and "false" otherwise.
dataset topnoise ***<wdb_obj> -fstart <value>*** [-fstop <value>] [-ntop <value>] [-pattern <value>] [-norecursive] [-depth <value>] [-runs <runID>,<runID1>,...] [-instance] [-file <path>] [-noplot] [-nogrid]	Performs a topnoise calculation on the specified waveform database folder or hierarchy.
Workspace Commands	
wave activeworkspace [*workspace_name*]	Cause a workspace to become active.
wave addworkspace [*workspace_name*] [-active]	Creates a new workspace.
wave listworkspace	Returns a list of workspaces.
Waveform Comparison Commands	
compare add (see compare add section for usage)	Creates the scope for waveform comparison.
compare clock [-delete] [-offset *delay*] [-rising | -falling | -both] ***clock_name waveform_path***	Defines a clock for clocked-mode comparisons.
compare configure [-clock *name*] [-recursive] [-tol *delay*] [-tolLead *delay*] [-tolTrail *delay*] [-mixtolLead *delay*] [-mixtolTrail *delay*] [-freqTol *freqtol*] [-amplTol *ampltol*] [-noiseFloor *noisefloor*] [-matchindex] [-vhdlmatches {*ref-logic-value=test-logic-value:...*}] [-vlogmatches {*ref-logic-value=test-logic-value:...*}] [-xTol *xtolerance*] [-yTol *ytolerance*] [-threshold {1|2}] [-upperthreshold *value*] [-lowerthreshold *value*] [-fixedthreshold value] [-refDelay *delay*] [-testDelay *delay*] [-start *start_value*] [-end *end_value*] [*-edgecompare*] ***comparePath***	Modifies options for compare signals and regions.
compare end	Closes active comparison without saving any information.

Table C-4. Supported Tcl Commands (cont.)

compare info [-start *start_value*] [-end *end_value*] [-write *filename*]	Displays results of the comparison in the main window.
compare list [-expand]	Outputs the Tcl scripts of all the **compare add** commands in effect.
compare options [-maxsignal *n*] [-maxtotal *n*] [-*mode_name*] [-tol *delay*] [-tolLead *delay*] [-tolTrail *delay*] [-vhdlmatches {*ref-logic-value*=*test-logic-value*: …}] [-vlogmatches {*ref-logic-value*=*test-logic-value*: … }] [-xTol *xtolerance*] [-yTol *ytolerance*] [-threshold {1|2}] [-upperthreshold *value*] [-lowerthreshold *value*] [-fixedthreshold *value*] [-wavewin *name*] [-noaddwave] [-addwave] [-adderrorwave]	Sets defaults for various waveform comparison commands.
compare run [*startTime*] [*endTime*]	Runs the difference computation.
compare savelog ***filename***	Saves a log of errors and warnings during comparison.
compare saverules [-session] [-expand] ***rules_filename***	Saves the rules of all **compare add** commands in effect and **compare options** to a file. Can also be used to save the entire comparison session.
compare start [-batch] [-maxsignal*n*] [-maxtotal*n*] [-refDelay *delay*] [-testDelay *delay*] ***reference_dataset*** [*test_dataset*]	Begins a new dataset comparison.

1. You can use extended options for this *waveform_name*. A footnoted *waveform_name* means that you can use the following options to select the waveform(s). See the section for details: [-show *wave_show_expression*][-recursive][-depth <level>][-signals][-quantities][-terminals][-nets][-ports][-in][-out][-inout][-internal][-through][-across][-free][-flow][-i][-boundary][-a2d][-d2a][-bidir][-run {*run_number* | *run_name*}][-session {previous | current}] [-adms | -modelsim]

When saving window contents to a Tcl file, an asterisk (*****) in an extended waveform name is interpreted as a wildcard character.

Specifying Waveforms in Tcl

You can specify one or more waveforms in Tcl.

This is done in the following ways:

- Waveform handles (*handle:* <#:#>) returned by the **wfc** command.

- Fully qualified names in one of the following forms:

  <*database_name*>*waveform_name*

  <*database/folder*>*waveform_name*

  For example:

  <tutorial/Time-Domain_Results>v_middle

- Object names. This method enables you to manipulate objects from the design and add a representation (if necessary). This syntax is consistent with the Questa ADMS **add wave** syntax.

  The object name syntax consists of the following parts:

  - **Dataset name** — The dataset name is the logical name for the JWDB file in which the object exists. Specifying the dataset name is optional when only one dataset is active, or if the command should only apply to the last loaded dataset.

    The following keywords are reserved dataset names:

    - **sim** — This reserved dataset name specifies the currently active simulation.
    - **prev** — This reserved dataset name refers to the PreviousSession folder, if it exists, in the current dataset. The PreviousSession folder must be activated in the modelsim.ini file while running Questa ADMS simulations.

  - **Dataset separator** — The data separator is used to terminate the dataset name string. In the EZwave viewer, this is a forward slash (**/**).
  - **Path separator** — In the EZwave viewer, this is a colon (**:**).
  - **Hierarchical path** — A set of hierarchy instance names, separated by a path separator (**:**).
  - **Object name** — The name of the object in the design.

    Object name syntax examples:

    ```
    add wave :top:clk
    add wave -show ac.vdb :top:out
    add wave adc12/top:x1:out
    ```

Selecting Waveforms in Tcl

Describes the extended options that are available for selecting waveform(s) within the Tcl commands.

Usage

Extended options:

```
[-show wave_show_expression] [-recursive] [-depth <level>]
[-signals] [-quantities] [-separator <separator>] [-terminals]
[-nets] [-notop] [-ports] [-in] [-out] [-inout] [-internal]
[-through] [-across] [-free] [-flow] [-i] [-boundary] [-a2d] [-d2a] [-bidir]
[-run {run_number | run_name | <operator>}] [-session {previous | current}]
[-adms | -modelsim]
```

Parameters

- -a2d

  (Optional) Only a2d nets are added.

- -across

  (Optional) Only across quantities are returned.

- -adms | -modelsim

  (Optional) Filters the list of matching waveforms to match those simulated by the Questa ADMS kernel or those simulated by the Questa SIM (formerly ModelSim) kernel.

- -bidir

  (Optional) Only bidir nets are added.

- -boundary

  (Optional) Only analog, digital, or both sides of boundary nets are returned, depending on your selection during simulation.

- -d2a

  (Optional) Only d2a nets are added.

- -depth *level*

  (Optional) Used with the -recursive option, restricts the recursive search to the specified level of hierarchy.

- -flow

  (Optional) Specifies that current waveforms should be displayed (instead of voltage, displayed by default). This is the same as “-show tran.i”

- -free

  (Optional) Only Free quantities are returned.

- -i

  (Optional) Specifies that current waveforms should be displayed (instead of voltage, displayed by default). This is the same as “-show tran.i”
- -in

  (Optional) Specifies that the scope of the search is to include ports of mode IN.
- -inout

  (Optional) Specifies that the scope of the search is to include ports of mode INOUT.
- -internal

  (Optional) Specifies that the scope of the search is to include internal (non-port) objects.
- -nets

  (Optional) Specifies that the scope of the search includes terminal, signal, and quantity items.
- -notop

  (Optional) Specifies that only waveforms located directly under an analysis folder (for example TRAN or AC) may be selected.
- -out

  (Optional) Specifies that the scope of the search include ports of mode OUT.
- -ports

  (Optional) Specifies that the scope of the search includes ports of modes IN, OUT, or INOUT.
- -quantities

  (Optional) Only quantity nets are added.
- -recursive

  (Optional) You may also use -r as a shortcut. Used with wildcard searches. Specifies that the scope of the search descend recursively into subregions. See also *-depth level* option.
- -run [*run_id* | "*run_id_1 run_id_2 ... run_id_n*" | <*operator*>]

  (Optional) This argument is used to find a specific (or several) run(s) of a compound waveform. *run_id* can be a numerical value or string, depending on the run identifier. The operators available are “=”, “!=”, “>”, “<“, “>=”, and “<=”. The operators “&” or “|” may also be used, but not together in the same -run value.

  Example of usage with add wave command:

  ```
  add wave  -run "TEMP > 0 & TEMP < 120" -show TRAN.v -separator .
     -terminals  XI11.NET4
  ```

- -separator <*separator*>

  (Optional) Specifies the separator used when parsing the object (waveform) name, for example "." or ":".

- -session [previous | current]

  (Optional) Specifies whether to look in a previous session folder or the current database.

- -show *representation* [|*representation2*|...|*representationN*]

  (Optional) Specifies how the signal is to be represented as a waveform. If -show is not specified, the defaults defined by the wave show command are used. If -show is specified, its settings are applied beginning where it exists in the script until another -show is issued, or until the end of the script.

  A *representation* is a complete representation of a net and is composed of an *analysis*, a *discipline*, and a *physic*. A dot (**.**) separates the *analysis* portion from the *discipline* and *physic* portion. There is no separator between the *discipline* and the *physic*, as follows:

  analysis.disciplinephysic

  If more than one *representation* is specified, use a pipe (**|**) to separate them:

  analysis.disciplinephysic|analysis.disciplinephysic|analysis.disciplinephysic

  The reserved keywords, **all** and **none**, can be used as follows:

  - -show all

    (Optional) Requests all objects found in all analyses

  - -show *analysis*.all

    (Optional) Requests all objects found in the specified analysis.

  - -show none

    (Optional) Requests objects directly located in the database in the case where there is no analysis folder.

    Refer to "Supported Net Representation Components" on page 1245 for details.

    Examples of -show usage:

    - add wave -show ac.vdb :test:u1:tvout

      In this example, the *analysis* is AC, the *discipline* is V, and the *physic* is DB. This will only search for the waveform named *database_name*/ AC>VDB(:test:u1:tvout)

    - add wave -show fsst.vdb.h(1) :top:net1

In this example, the *analysis* is FSST, the *discipline* is V, and the *physic* is DB for the *primary_ physic*, and H(1) for the *secondary_physic*. This will search for the specified waveform in the last-opened database:

database_name/FSST>VDB(:top:net1).H(1)

- -signals

  (Optional) Only signal nets are added.

- -terminals

  (Optional) Only reference quantities of terminal nets are added.

- -through

  Specifies that only Through quantities matching the net names are saved

Description

The commands for selecting waveform(s) within the Tcl commands with extended options are listed in the following table:

Table C-5. Commands that Access Waveforms

add wave	dataset savewaveforms	wave addannotation	wave exists
compare add	delete wave	wave adddeltamarker	wave histogram
compare clock	find analogs	wave addproperty	wave runindexlist
compare configure	find digitals	wave cdf	
dataset mergewaveforms	find nets | signals	wave createbus	

Tcl Command Detailed Descriptions

This section details the supported Tcl commands, listed alphabetically.

Table C-6. Tcl Commands

Command	Description
add wave	Adds one or more waveforms to a graphical window.
add workspace	Creates a new workspace in the EZwave session window.
batch_mode	Indicates whether the EZwave viewer or the JWDB server is used, to check whether the Tcl script is currently in batch mode.
bloc	Enables multi-line Waveform Calculator scripts.
compare add	Creates the scope of the comparison (all waveforms, top level ports, …) and supports a set of options, like tolerances, to be applied to the comparison. This command has additional arguments, compared to Questa SIM commands, to enable defining tolerances and settings for analog-analog and analog-digital comparison. The same command is then used in to set up complete mixed-signal comparison. Path of regions is not supported; a global expression (regular expression) should be used instead. For example, :top:level1:* rather than :top:level1.
compare clock	Defines a clock that can be used for clocked-mode comparisons. In clocked-mode comparisons, signals are sampled and compared only at or just after an edge on some signals.
compare configure	Modifies options for compare signals and regions.
compare end	Closes the active comparison without saving any information.
compare info	Lists the results of a waveform comparison in the main window transcript. To save the information to a file, use the -write argument.
compare list	Outputs the Tcl scripts of all the compare add commands in effect.
compare options	Sets defaults for various waveform comparison commands. Those defaults are used when other compare commands are invoked during the current session.
compare run	Runs the difference computation on the signals selected via a compare add command.

Table C-6. Tcl Commands (cont.)

Command	Description
compare savelog	Creates a log file containing error and warning messages generated during waveform comparison.
compare saverules	Saves the rules (all compare add commands in effect and compare options) to a file. Can also be used to save the entire comparison session.
compare start	Begins a new dataset comparison.
dataset alias	Adds additional names (aliases) to a dataset.
dataset analysis	Returns a list of folders (or subfolders) of analysis of the dataset (or folder).
dataset clear	Clears all waveform data from the last opened database.
dataset close	Closes the specified database or all databases.
dataset extract	Runs Eldo -extract on the source database using the source netlist to define the .extract and .meas commands.
dataset info	Returns the specified information about a database.
dataset list	Returns a list of currently opened databases.
dataset merge	Merges databases containing single run simulation results, and generates compound waveforms from found waveforms. When supplied the name of an index file, the command merges the referenced databases and adds parameters as defined in the file.
dataset mergewaveforms	Merges waveforms, grouping waveforms of the same name as if there were multiple simulation runs.
dataset open	Opens a database file.
dataset ovd	Opens an OVD file.
dataset power analysis	Performs a power analysis of the specified waveform database power analysis folder.
dataset rename	Renames the specified database.
dataset save	Saves the specified database to disk as the specified file name.
dataset savescalar	Save scalars, such as noise contributors, to a TXT file or CSV file.
dataset savewaveforms	Saves the specified waveform(s) to disk as the specified file name.
dataset scalar	Compares scalar values of variables stored in two databases (PSF, nutmeg, FSDB, WDB, OVD).

Table C-6. Tcl Commands (cont.)

Command	Description
dataset statistics	Returns statistics on the dataset.
dataset supported	Returns “true” if the dataset can be opened by EZwave, and “false” otherwise.
dataset topnoise	Performs a topnoise calculation on the specified waveform database folder or hierarchy. Returns noise spectrum in sorted order of spot noise if only *fstart* is specified, or integrated noise if both *fstart* and *fstop* are specified.
delete wave	Removes the specified waveform from the indicated window.
dofile	Tells the EZwave Tcl interpreter to source the specified Tcl file.
environment	Displays or changes the region/signal environment.
evalExpression	Invokes the EZwave Waveform Calculator to calculate the expression entered.
examine	Examines an object and displays its current value in the Transcript Window.
exit	Exits the EZwave viewer.
find analogs	Returns the names of all the analog objects (quantities and terminals) matching the specified pattern.
find currents	Returns the names of all the current objects matching the specified pattern.
find digitals	Returns a list of digital signals matching the specified pattern.
find nets | signals	These commands return the full pathnames of all analog and digital objects that match the name specification.
getactivecursortime	Returns the location of the active cursor.
precision	Sets the double or cursor display precision.
printenv	Displays the name and value of environment variables.
quit	Exits the EZwave viewer.
radix	Defines the default bus radix or returns the current default if no radix is specified.
radix define	Defines a new radix.
radix delete	Removes the radix definition from the named radix.
radix list	Returns the complete definition of the named radix.

Table C-6. Tcl Commands (cont.)

Command	Description
radix names	Returns a list of all the defined radices.
radix signal	Sets the radix value for the specified waveforms, or if the radix value is not specified it inspects the radix values.
save	Saves the specified Waveform Calculator measure.
setenv	Sets or displays the value of the specified environment variable.
unsetenv	Unsets an environment variable.
wave activecursor	Sets the specified cursor as the active cursor.
wave activewindow	Returns the name of the currently active window, or sets a specified window to be active.
wave activeworkspace	Causes the specified workspace to become active.
wave addannotation	Adds a text annotation to a waveform at the specified location in the active window.
wave addaxisdeltamarker	Adds a delta marker, between two values, on the selected waveform. The marker can be horizontal or vertical. Optionally text can be added to this marker.
wave addcursor	Creates a new cursor in the specified window.
wave adddeltamarker	Adds a delta marker waveform(s) at the specified location(s) in the active window.
wave addline	Adds a horizontal or vertical line to a waveform in the active window.
wave addmarker	Adds a vertical marker to the specified location.
wave addproperty	Adds a property to a waveform.
wave addwindow	Creates a new graphical window in the EZwave session window and makes it active.
wave addworkspace	Creates a new workspace in the EZwave session window.
wave cdf	Generates and displays a CDF based on the input waveform, or on the saved histogram data in a *.wdb* file generated from an Eldo Monte Carlo simulation.
wave clean	Takes a name and returns a new name with any special characters replaced by "_". This is useful, for example, when scripting inside EZwave and creating waves through the waveform calculator, where the labels chosen to create the waves might cause the script to work incorrectly.

Table C-6. Tcl Commands (cont.)

Command	Description
wave closewindow	Closes a specified window, all windows, or all windows in the workspace.
wave closeworkspace	Closes a specified workspace, or all workspaces.
wave colortheme	Sets the graphical display or print color scheme.
wave compressvri	Computes the *Nth* compression point for the specified harmonic.
wave createbus	Creates a bus containing the specified waveforms, inside the *calc* database.
wave cursortime	Moves a cursor to the specified position.
wave deletecursor	Deletes the specified cursor, or all cursors, in a window.
wave difference	Generates and displays a waveform corresponding to the difference between the first and second waveforms specified.
wave displayed	Retrieves a list of the displayed waveforms.
wave exists	Indicates whether the specified waveform exists.
wave ezwave_title	Adds a custom title to the top of the EZwave main display.
wave gettype	Returns the type of the specified waveform.
wave histogram	Generates and displays a histogram based on the input waveform, or on the saved histogram data generated from an Eldo Monte Carlo simulation.
wave ipnvri	Computes the *Nth* order input-referred intercept point, output-referred intercept point, or both for the specified input compound waveform.
wave jitter	Performs the calculation and plot of the specified jitter and its measurements.
wave launchfolder	Identifies the path to the directory that EZwave was launched from.
wave listworkspace	Returns a list of workspaces.
wave loadbindings	Loads an EZwave keyboard shortcuts schema **.ezkey* file.
wave lockcursor	Specifies whether the distance between cursors are fixed or whether they can move independently of each other.
wave names	Determines whether waveform names within the graphical interface show the dataset names and (or) the complete hierarchical paths.

Table C-6. Tcl Commands (cont.)

Command	Description
wave plotcompress	Plots output power against input power in dBm and returns n-dB compression point. Only supports compound waveforms.
wave plotpacipn	Takes a PAC spectrum signal pair at the desired and unwanted harmonics, within an input power sweep, plots the pair of IPn curves and returns the intercept point. Only supports compound waveforms.
wave refresh	Redraws the contents of the specified window.
wave rowfit	Optimizes the row size in the specified window.
wave runindexlist	Displays a list of run indices for the elements of the compound waveform specified. It is useful for iterating through each element individually.
wave runparameters	Displays a list of run parameters for the compound waveform specified. It is useful for iterating through each element individually.
wave runparametervalue	Displays the value of a specified run parameter associated with the run_index and compound waveform specified.
wave show	Defines the default net representation that is used when the **add wave** command is issued *without* the -show option. If the -show option is specified with the **add wave** command (add wave -show *representation*), it temporarily overrides the **wave show** defaults during the execution of the **add wave** command and then returns to the **wave show** defaults.
wave showgridlines	Turns gridlines on or off in all new windows.
wave showzerolevels	Activates the zero-levels in the rows where the waveform is displayed.
wave tandem mode	Activates or deactivates Tandem Mode.
wave tile	Arranges multiple windows in the specified tiling style.
wave windowdecoration	Shows or hides the decoration of wave windows.
wave windowlist	Returns a list of all open graph windows.
wave xaxis	Sets the x-axis unit title, and the x-axis scale to one of: linear, base 2 log, or base 10 log.
wave yaxis	Used to alter y-axis settings such as the y-axis name, alignment, unit title, scale, and alignment for the specified row.
wave zoomfull	Zooms out to show the entire x axis of the specified graph window.

Table C-6. Tcl Commands (cont.)

Command	Description
wave zoomin	Zooms in on the x axis of the specified graph window, by the specified factor.
wave zoomlast	Undoes the most recent zoom operation in the specified window.
wave zoomout	Zooms out on the x axis of the specified graph window, by the specified factor.
wave zoomrange	Zooms in on the x axis to a range bounded by the specified start and/or end points.
wfc	Invokes the EZwave Waveform Calculator to calculate the expression entered.
write jpeg	Exports the contents of a window to a specified JPEG image file.
write png	Exports the contents of a window to a specified PNG image file.
write wave	Outputs window contents in PostScript format. It can be sent to a printer or a specified location.

add wave

Adds one or more waveforms to a graphical window.

Usage

```
add wave

[-overlay]
[-append]
[-row row_number]
[-position {top | bottom | end | before | after | below | row_number}]
[-rowtitle title]
[-rowtitle_font {name | family} {plain | bold | italic} size]
[-rowtitle_color color]
[-rowtitle_outline {true | false | default}]
[-zerolevel {on | off | horizontal | vertical}]
[-grid {on | off | horizontal | vertical}]
[-point_style id]
[-line_width {1 | 2 | 3 | 4 | 5}]
[-zoomY min_y max_y]
[-height pixels]
[-hide | -visible]
[-group group_name]
[-color standard_color_name]
[-colorgroup param_1,...,param_n]
[-hide_filter runID_1,...,runID_n]
[-highlight_filter runID_1,...,runID_n]


[-label name]
[-literal | -event | -analog-step | -analog-interpolated | -spectral]
[-window window_name]
[-symbolic | -binary | -octal | -decimal | -hexadecimal | -unsigned | -
ascii | -time | -default]
[-enumnumeric | -enumsymbolic]
[-radix {radix_type | radix_name}]
[-collapse]
[-expand]
[-adms | -modelsim]
[-complexplane | -smithchart {-impedance | -admittance} | -polarchart
{-degree | -radian} | -versus x_wf_name]
[-nomode]
```

```
[-newyaxis]
[-hideyaxis]
[-xscale {linear | log10 | log2}]
[-yscale {linear | log10 | log2}]
[-hidexunit]
[-hideyunit]
[-hidextitle]
[-hideytitle]
[-scaleunitx scale_value unit_name]
[-scaleunity scale_value unit_name]
[-tograydec]
[-graytodec]
[-fpoint n]
[wf_name_1 ... wf_name_n] | [-separator separator_char wf_name_1 ...
wf_name_n [-noseparator wf_name_1 ... wf_name_n]]
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- -append

  (Optional) Adds the waveform overlaid on the last available row in the window. If no row exists, it will be created and waveform added. The use of both -append and -row <row_ID> switches together is not recommended because this may lead to unexpected results.

- -color *standard_color_name*

  (Optional) Specifies a color for displaying the waveform. You can use standard color names (for example, 'red' or 'blue') or the RGB color code preceded by a '#'.

  Legal color names: black, blue, cyan, gray, darkgray, lightgray, green, magenta, orange, pink, red, white. yellow.

  RGB color code: The RGB color code is a hexadecimal value preceded by a '#' representing *#RRGGBB*, where *RR* represents the red value, *GG* represents the green value, and *BB* represents the blue value. For example, #0000ff is blue, #000000 is black, and #ffffff is white.

  For compound waveforms, forces the same color to be used for all elements.

- -colorgroup *param_1 . . .param_n*

  (Optional) Specifies a list of compound waveform parameter names in the color group. Waveform runs with identical parameter values will be plotted using the same color.

  See also "Parameter Table Dialog Box" on page 568.

- -collapse

  (Optional) If this option is specified, digital buses and compound waveforms are displayed with all elements collapsed. When not specified, the default setting for compound waveforms (set in **Edit** > **Options** > **Multiple Run**) is used.

- -complexplane | -smithchart [-impedance | -admittance] | -polarchart {-degree | -radian} | -versus *x_wf_name*

  (Optional) Defines a row type to be used for displaying the waveform.

- -enumnumeric | -enumsymbolic

  Accepted for Questa SIM compatibility, but ignored by EZwave.

- -expand

  (Optional) Displays a digital bus with all its bits expanded. Displays a compound waveform expanded to show all single element names. When not specified, the default setting for compound waveforms (set in **Edit** > **Options** > **Multiple Run**) is used.

- -fpoint *n*

  (Optional) Specifies *n* fixed-point radix.

- -graytodec

  (Optional) Coverts gray radix to decimal representation radix.

- -grid [on | off | horizontal | vertical]

  (Optional) Activates the grids in the rows where the waveform is displayed. **on** and **off** enable or disable both horizontal and vertical grids. **horizontal** and **vertical** enable only the appropriate horizontal or vertical grids.

- -group *group_name*

  Creates a waveform group with the name *group_name*, containing the waveforms specified with <*object_name*>. If the specified group already exists, all waveforms specified with <*object_name*> are appended to that group.

  See "Grouping Waveforms" on page 131.

- -height *pixels*

  (Optional) Specifies the height of the row where the waveform is displayed. The value is in pixels.

- -hide | -visible

  (Optional) Specifies whether the waveform should be visible when plotted. When plotting multiple waveforms, -visible cancels out an earlier -hide and visa versa.

- -hideyaxis

  (Optional) Specifies that the current y axis is hidden if more than one axis exists in the row. All waveforms plotted in a hidden y axis are automatically hidden.

- -hidextitle

  (Optional) Specifies that the x-axis title should not be displayed.

- -hideytitle

  (Optional) Specifies that the y-axis title should not be displayed.

- -hidexunit

  (Optional) Specifies that the x-axis units should not be displayed.

- -hideyunit

  (Optional) Specifies that the y-axis units should not be displayed.

- -hide_filter *runID_1 . . .runID_n*

  (Optional) Specifies a list of compound waveform runs to hide.

  See also "Parameter Table Dialog Box" on page 568.

- -highlight_filter *runID_1 . . .runID_n*

  (Optional) Specifies a list of compound waveform runs to highlight.

  See also "Parameter Table Dialog Box" on page 568.

- -label *name*

  (Optional) Specifies a name for the waveform.

- -line_width {1 | 2 | 3 | 4 | 5}

  (Optional) Specifies the line with of the waveform. The numbers map directly to the widths in the Line Width dropdown list in the Waveform Display Options

- -literal | -event | -analog-step | -analog-interpolated | -spectral

  (Optional) Specifies a drawing mode for the waveform that is displayed.

  - -literal - This corresponds to the "railroad" drawing mode
  - -event - This corresponds to the "scattered" drawing mode
  - -analog-step - This corresponds to the "sampled" drawing mode
  - -analog-interpolated - This corresponds to the "continuous" drawing mode
  - -spectral - This corresponds to the "spectral" drawing mode

- -newyaxis

  (Optional) Specifies that the waveform is displayed using a new y axis.

- -nomode

  (Optional) Searches port terminals that do not have IN, OUT, or INOUT modes, and therefore are not included in searches that specify the -in, -out, -inout arguments.

- -noseparator *wf_name_1 ... wf_name_n*

  (Optional) Specifies that the waveform names have no hierarchy.

- -overlay

  (Optional) Causes several waveforms added at once to be added in the same row.

- -point_style *id*

  (Optional) Specifies the point style of the waveform. *id* takes a value between 0 and 16. When 0 is specified, no point style is applied. Setting this value is equivalent to making a selection in the Data Point Symbol field on the Waveform Properties Dialog Box (For Analog Waveforms), accessed by right-clicking on a waveform and selecting **Properties**.

- -position {top | bottom | end | before | after | below | *row_number* }

  (Optional) Specifies the position for the waveform inside the graph window.

  - top - Adds the signal(s) to the beginning of the list of signals
  - bottom | end - Adds the signal(s) to the end of the list of signals
  - before | above - Adds the signal(s) before the selected signal in the graph window
  - after | below - Adds the signal(s) after the selected signal in the graph window
  - *row_number* - Adds the signal(s) to the specified row number. Row numbers start from the top of the window with a value of 1 and increase as you go down the window. Row numbers are dynamic, changing as row are added or removed.

- -radix {*radix_type* | *radix_name* }

  (Optional) Specifies a radix or a user-defined radix. Can be used as an alternative to the options:

  [-symbolic | -onescomplement | -magnitude | -binary | -octal | -decimal | -hexadecimal | -unsigned | -ascii]

  Legal *radix_type* values: symbolic, onescomplement, magnitude, binary, octal, decimal, hexadecimal, unsigned, ascii, time and default.

  Hexadecimal can also be written as hex.

  Options -time and -default are accepted for Questa SIM compatibility, but are ignored by EZwave.

- -row *row_number*

  (Optional) Selects the row in which to display the waveform. The row numbers start at 1 (top-most row). Row numbers dynamically change as rows are added or removed.

- -rowtitle *title*

  (Optional) Specifies the title of the row to be added.

- -rowtitle_font {*name* | *family*} {plain | bold | italic} *size*

  (Optional) Specifies the font characteristics of the row title.

  - *name* | *family* - Specify ether the name of the font to use or the system font family (serif, sans, and so on).
  - plain | bold | italic - Specify the font style.
  - *size* - Specify the font size, in points.

- -rowtitle_color *color*

  Specifies the title text color, using an RGB (hex) value. The RGB color code is a hexadecimal value preceded by a "#" representing #RRGGBB, where RR represents the red value, GG represents the green value, and BB represents the blue value. For example, #0000ff is blue, #000000 is black, and #ffffff is white.

- -rowtitle_outline {true | false | default}

  Controls the visibility of the text box around the row title.

- -scaleunitx *scale_value unit_name*

  (Optional) Specifies the scaling factor and unit type for the x axis.

- -scaleunity *scale_value unit_name*

  (Optional) Specifies the scaling factor and unit type of the y axis.

- -symbolic | -binary | -octal | -decimal | -hexadecimal | -unsigned | -ascii | -time | -default

  (Optional) Specifies a radix for displaying digital bus(es) specified in this command.

  If no radix is specified for an enumerated type, the default radix is used. You can change the default radix for the current simulation using the radix command.

  Note

  -time and -default are accepted for Questa SIM compatibility, but ignored by EZwave.

- -separator *separator_char wf_name_1 ... wf_name_n*

  (Optional) Specifies the hierarchical separator character in waveform names. For example:

  ```
  add wave -overlay -show TRAN.v -separator . XV1.G1
  ```

  In this example, the dot (**.**) is the hierarchical separator in the wave XV1.G1.

- -tograydec

  (Optional) Converts decimal representation radix to gray radix.

- *wf_name_1 . . .wf_name_n*

  (Optional) Specifies the waveform(s) using one of several accepted syntaxes. See Specifying Waveforms in Tcl.

  You must use the object name method for specifying waveforms when using the -show *representation* option.

  You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform(s). You can use more than one option in the same command.

- -window *window_name*

  (Optional) Specifies the window where the waveform should be displayed. If this is not specified, the waveform is displayed in the currently active window.

- -xscale {linear | log10 | log2 }

  (Optional) Specifies the type of scale used for the x axis.
- -yscale {linear | log10 | log2 }

  (Optional) Specifies the type of scale used for the y axis.
- -zerolevel [on | off | horizontal | vertical]

  (Optional) Activates the zero-levels in the rows where the waveform is displayed. On/Off enables or disables both horizontal and vertical levels. Horizontal/Vertical enables only the appropriate horizontal or vertical levels.
- -zoomY *min_ymax_y*

  (Optional) Defines the Y limits while displaying the waveform.

Examples

- This example will plot <*tutorial/Time-Domain_Results*>*v_load* and the result waveform. The resulting waveform is displayed in blue:

```
set result [wfc {wf("<tutorial/Time-Domain_Results>v_load")-
wf("<tutorial/Time-Domain_Results>v_middle")}]

add wave <tutorial/Time-Domain_Results>v_load -color blue $result
```

- This example creates a group named allAnalog that will contain all analog waveforms from the database.

```
add wave -group allAnalog -terminals -r *
```

add workspace

Creates a new workspace in the EZwave session window.

Usage

```
add workspace [workspace_name] [-active]
```

Parameters

- *workspace_name*

  (Optional) Specifies the name of the workspace to be created. If *workspace_name* is not specified, a default name, *Workspace#*, is used, where # is a number.

- -active

  (Optional) Indicates that the workspace created should become the active workspace.

Description

This command is equivalent to wave addworkspace.

Examples

```
add workspace ADC12
```

batch_mode

Indicates whether the EZwave viewer or the JWDB server is used, to check whether the Tcl script is currently in batch mode.

Usage

`batch_mode`

Parameters

None.

Return Values

1 if used with ezwave -c or the JWDB Server (this indicates that the program is in non-graphical batch mode).

0 if used with the EZwave viewer (this indicates that the program is in graphical non-batch mode).

Examples

```
if [batch_mode] {
...
} else { # execute graphical commands
   add wave $wf
}
```

Related Topics

Invoking EZwave in Batch Mode

bloc

Enables multi-line Waveform Calculator scripts.

Usage

```
bloc {script}
```

Parameters

- ***script***

  (Required) Specifies a multi-line Waveform Calculator script.

Examples

```
bloc  {

for x in range(5):
   print x

}
```

In the following example, customceil(3.5) will return 4.0.

```
bloc {

import math

def customceil(x):
   return math.ceil(x)

}
```

Note

Only pure Python modules work with the “import” function. C-Python external modules cannot be loaded into the Waveform Calculator.

Related Topics

Waveform Calculator GUI

compare add

Creates the scope of the comparison (all waveforms, top level ports, …) and supports a set of options, like tolerances, to be applied to the comparison. This command has additional arguments, compared to Questa SIM commands, to enable defining tolerances and settings for analog-analog and analog-digital comparison. The same command is then used in to set up complete mixed-signal comparison. Path of regions is not supported; a global expression (regular expression) should be used instead. For example, :top:level1:* rather than :top:level1.

Usage

```
compare add [-clock clock] [-label label] [-nowin]
[-tol delay] [-tolLead delay] [-tolTrail delay]
[-mixtolLead delay] [-mixtolTrail delay]
[-freqtol freqtol] [-amptol amptol] [-noisefloor noisefloor]
[-matchindex]
[-vhdlmatches {ref-logic-value=test-logic-value:...}]
[-vlogmatches {ref-logic-value=test-logic-value:...}]
[-xTol xtolerance] [-yTol ytolerance] [-mindeltax value]
[-threshold {1|2}] [-upperthreshold value]
[-lowerthreshold value] [-fixedthreshold value] [-refDelay delay]
[-testDelay delay] [-start start_value] [-end end_value]
referencePath [testPath] [-wave] [-win wname] [-edgecompare]
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- -clock*clock*

  (Optional) Specifies the clock definition to use when sampling the specified regions. Required for a clocked comparison; not used for asynchronous comparisons.

- -label *label*

  (Optional) Specifies a name for the comparison when it is displayed in the window.

- -nowin

  (Optional) Specifies that compare signals shouldn't be added to any window. By default, compare signals are added to the default Wave window. See the -wave parameter.

- -recursive

  (Optional) Specifies that signals should also be selected in all nested subregions, and subregions of those, and so on.

- -tol *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to lead or trail a reference edge in an asynchronous comparison. The default is 0. If a unit (for example, ps) is used with the time value, the time must be placed in braces {}.

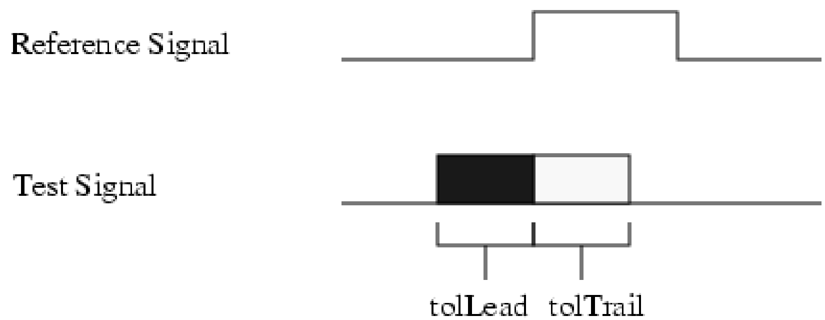
- -tolLead *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to lead a reference edge in an asynchronous comparison. The default is 0. If a unit (for example, ps) is used with the time value, the time must be placed in braces {}.

- -tolTrail *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to trail a reference edge in an asynchronous comparison. The default is 0. If a unit is used (for example, ps) with the time value, the time must be placed in braces {}.

Figure C-1. Graphical Representation of tolLead and tolTrail



- -mixtolLead *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to lead a reference edge in a mixed signal comparison. The default is 0. If a unit (for example, ps) is used with the time value, the time must be placed in braces {}.

- -mixtolTrail *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to trail a reference edge in a mixed signal comparison. The default is 0. If a unit is used (for example, ps) with the time value, the time must be placed in braces {}.

- -freqtol *freqtol*

  (Optional) Specifies the frequency tolerance for a spectral comparison.

- -amptol *amptol*

  (Optional) Specifies the amplitude tolerance for a spectral comparison.

- -noisefloor *noisefloor*

  (Optional) Specifies the noise floor level for a spectral comparison. Errors that occur below this amplitude level are ignored in the comparison.

- -matchindex

  (Optional) Specifies that only compound waveforms (from multiple runs) with matching indexes are compared. This mode is off by default.

  For example, consider a reference dataset containing only a few runs (e.g. runs 1, 3 and 5) and a test dataset with many more runs (e.g. runs 1 to 10). Without the -matchindex option, reference:run_1 is compared with test:run_1, but reference:run_3 is compared with test:run_2, because runs are compared in the order that they are found in the compound waveform. Using the -matchindex option, reference:run_3 would now be compared with the corresponding test:run_3.

- -vhdlmatches {*ref-logic-value*=*test-logic-value*: …}

  (Optional) Specifies how VHDL signal states in the reference dataset should match values in the test dataset. Values are specified in a colon-separated list of match values.

  For example:

  ```
  -vhdlmatches {X=XUD:Z=ZD:1=1HD}
  ```

  Default:

  ```
  {U=UWXD:X=UWXD:0=0LD:1=1HD:Z=ZD:W=UWXD:L=0LD:H=1HD:D=UX01ZWLHD}
  ```

  The 'D' character represents the '-' “don't care” std_logic value.

- -vlogmatches {*ref-logic-value*=*test-logic-value*: …}

  (Optional) Specifies how Verilog signal states in the reference dataset should match values in the test dataset. Values are specified in a colon-separated list of match values.

  For example:

  ```
  -vlogmatches {0=0:1=1:Z=Z}
  ```

  Default:

  ```
  {0=0:1=1:Z=Z:X=X}
  ```

- -xTol *xtolerance*

  (Optional) Specifies the maximum X tolerance (usually time) that test and reference waveforms are allowed to differ.

  Default:

  $$\text{the minimum of } \left(\frac{x\_max - x\_min}{number\_of\_points}\right) or\ (0.01 \times x\_max - x\_min)$$

- -yTol *ytolerance*

  (Optional) Specifies the maximum Y tolerance that test and reference waveforms are allowed to differ. The default is $0.01 \times (y\_max - y\_min)$

- -mindeltax *value*

  (Optional) Specifies the minimum delta X tolerance (usually time). Ignores differences that are less than the specified value. Default is 0.

- -threshold 1 | 2

  (Optional) Specifies if one or two thresholds should be used when converting analog waveform to digital waveform in digital-analog comparison. Default is 1.

- -upperthreshold *value*

  (Optional) Specifies the value of the upper level to be used for converting an analog waveform to a digital waveform, in case of analog-digital comparison using two thresholds. The default is automatically computed based on input waveforms.

- -lowerthreshold *value*

  (Optional) Specifies the value of the lower level to be used for converting an analog waveform to a digital waveform, in case of analog-digital comparison using two thresholds. The default is automatically computed based on input waveforms.

- -fixedthreshold *value*

  (Optional) Specifies the value of the level to be used for converting an analog waveform to a digital waveform, in case of analog-digital comparison using two thresholds. The default is automatically computed based on input waveforms.

- -refDelay *delay*

  (Optional) Specifies a delay to shift all added waveforms from the reference dataset. This setting overrides the delay setting of the **compare start** command.

- -testDelay *delay*

  (Optional) Specifies a delay to shift all added waveforms from the test dataset. This setting overrides the delay setting of the **compare start** command.

- -start *time*

  Specifies the time at which to start the comparison of all added waveforms. This setting overrides the setting from a **compare run** or **info** command).

- -end *time*

  (Optional) Specifies the time at which to end the comparison of all added waveforms. This setting overrides the setting from a **compare run** or **info** command.

- ***referencePath***

  (Optional) Specifies either an absolute or relative path to the reference signal or region, or a glob expression. Relative paths are relative to the current context of the reference dataset. If you specify a glob expression, it will match signals only in the containing context.

  You can use the extended options detailed in Selecting Waveforms in Tcl to match the signals. You can use more than one option in the same command.

Some examples:

```
compare add :top:x1:out

# With wildcard:
compare add :top:x1:*

# With dataset name:
compare add adc12/:top:x1:out

# When a context is defined (see the environment command), then
# the relative path is accepted:
env :top:x1
compare add out
```

- *testPath*

  (Optional) Specifies an absolute or relative path to the test signal or region. Cannot be a glob expression. If omitted, the test path defaults to the same path as *referencePath* except for the dataset name.

- -wave

  (Optional) Specifies that compare signals be added automatically to the default Wave window. Default is true.

- -win *wname*

  (Optional) Specifies a particular window to which to add objects. Used to specify a particular window when multiple instances of that window type exist.

- -edgecompare

  (Optional) Forces a "digital" comparison where the source and reference signals (one, or both) are analog. This applies an automatic analog to digital conversion, enabling an edge comparison to be made. Digital comparison tolerances will apply for the edge comparison, refer to "Digital Comparison" on page 254.

Examples

- Add the waveform name tvin to the comparison using the default -showoptions (see also "Supported Net Representation Components" on page 1245).

  ```
  compare add tvin
  ```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare clock

Defines a clock that can be used for clocked-mode comparisons. In clocked-mode comparisons, signals are sampled and compared only at or just after an edge on some signals.

Usage

```
compare clock [-delete] [-offset <delay>] [-rising | -falling | -both]
<clock_name> [<signal_path> | -frequency <value> [-dutycycle <perc>]]
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- -delete

  (Optional) Deletes an existing compare clock.

- -offset <delay>

  (Optional) Specifies a time value for delaying the sample time beyond the specified signal edge. The default is 0.

- -rising | -falling | -both

  (Optional) Specifies that the rising, falling, or both (rising and falling edges) of the specified signal should be used. The default is rising.

- **clock_name**

  (Required) A name for this clock definition. This name is used with the compare add command when doing a clocked-mode comparison.

- **signal_path**

  (One of **signal_path** or **-frequency** must be specified) A full path to the waveform whose edges are to be used as the strobe trigger.

  You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform. You can use more than one option in the same command.

  Note

  This command currently only supports fully qualified names when specifying the signal path. See Specifying Waveforms in Tcl for details on fully qualified names.

- **-frequency <value>**

  (One of **signal_path** or **-frequency** must be specified) Specifies a perfect clock frequency.

- -dutycycle <perc>

  (Optional) Only applicable if **-frequency <value>** is defined, this specifies the amount of time the signal is active per period, defined as a percentage (0 to 1)

Examples

- Create a clock named myClock that will trigger a comparison on both rising and falling edges of the source waveform <REF/TRAN>:test:eoc.

  ```
  compare clock -both myClock <REF/TRAN>:test:eoc
  ```

- Define a comparison based on an ideal clock named “idealClock” with a frequency of 10000Hz and a duty cycle of 40%. This clock will start at 0.2ns and the comparison will only consider its rising edges.

  ```
  -compare clock -offset 0.2n -rising idealClock -frequency 1e4
  -dutycycle 0.4
  ```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare configure

Modifies options for compare signals and regions.

Usage

```
compare configure [-clock name] [-recursive] [-tol delay] [-tolLead delay]
[-tolTrail delay] [-mixtolLead delay] [-mixtolTrail delay]
[-freqtol  freqtol] [-amptol  amptol] [-noisefloor  noisefloor]
[-matchindex]
[-vhdlmatches {ref-logic-value=test-logic-value:...}]
[-vlogmatches {ref-logic-value=test-logic-value:...}]
[-xTol xtolerance] [-yTol ytolerance] [-mindeltax value]
[-threshold {1|2}] [-upperthreshold value]
[-lowerthreshold value] [-fixedthreshold value] [-refDelay delay]
[-testDelay delay] [-start start_value] [-end end_value] [-edgecompare]
comparePath
```

Parameters

Note
This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- -clock*name*

  (Optional) Changes the strobe signal for the comparison. If the comparison is currently asynchronous, it is changed to clocked. This switch may not be used with the -tol, -tolLead, and -tolTrail options.

- -recursive

  (Optional) Specifies that signals should also be selected in all nested subregions, and subregions of those, and so on.

- -tol *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to lead or trail a reference edge in an asynchronous comparison. The default is 0.

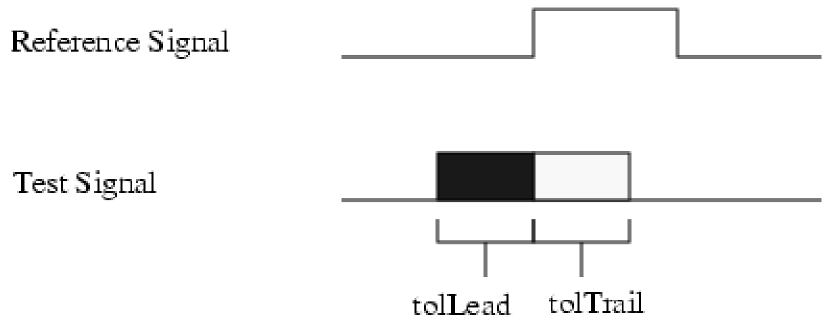
- -tolLead *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to lead a reference edge in an asynchronous comparison. The default is 0.

- -tolTrail *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to trail a reference edge in an asynchronous comparison. The default is 0.

Figure C-2. Graphical Representation of tolLead and tolTrail



- -mixtolLead *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to lead a reference edge in a mixed signal comparison. The default is 0.

- -mixtolTrail *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to trail a reference edge in a mixed signal comparison. The default is 0.

- -freqtol *freqtol*

  (Optional) Specifies the frequency tolerance for a spectral comparison.

- -amptol *amptol*

  (Optional) Specifies the amplitude tolerance for a spectral comparison.

- -noisefloor *noisefloor*

  (Optional) Specifies the noise floor level for a spectral comparison. Errors that occur below this amplitude level are ignored in the comparison.

- -matchindex

  (Optional) Specifies that only compound waveforms (from multiple runs) with matching indexes are compared. This mode is off by default.

  For example, consider a reference dataset containing only a few runs (e.g. runs 1, 3 and 5) and a test dataset with many more runs (e.g. runs 1 to 10). Without the -matchindex option, reference:run_1 is compared with test:run_1, but reference:run_3 is compared with test:run_2, because runs are compared in the order that they are found in the compound waveform. Using the -matchindex option, reference:run_3 would now be compared with the corresponding test:run_3.

- -vhdlmatches {*ref-logic-value*=*test-logic-value*: …}

  (Optional) Specifies how VHDL signal states in the reference dataset should match values in the test dataset. Values are specified in a colon-separated list of match values. For example:

  ```
  -vhdlmatches {X=XUD:Z=ZD:1=1HD}
  ```

Default:

```
{U=UWXD:X=UWXD:0=0LD:1=1HD:Z=ZD:W=UWXD:L=0LD:H=1HD:D=UX01ZWLHD}
```

The 'D' character represents the '-' “don't care” std_logic value.

- -vlogmatches {*ref-logic-value*=*test-logic-value*: …}

  (Optional) Specifies how Verilog signal states in the reference dataset should match values in the test dataset. Values are specified in a colon-separated list of match values. For example:

  ```
  -vlogmatches {0=0:1=1:Z=Z}
  ```

  Default:

  ```
  {0=0:1=1:Z=Z:X=X}
  ```

- -xTol *xtolerance*

  (Optional) Specifies the maximum X tolerance (usually time) that test and reference waveforms are allowed to differ.

  The default:

  $$\text{the minimum of}\left(\frac{x\_max - x\_min}{number\_of\_points}\right) or\ (0.01 \times x\_max - x\_min)$$

- -yTol *ytolerance*

  (Optional) Specifies the maximum Y tolerance that test and reference waveforms are allowed to differ. The default:

  $$\{K \in \mathrm{N} | 1 \leq K \leq total\_nb\_of\_edges\}$$

- -mindeltax *value*

  (Optional) Specifies the minimum delta X tolerance (usually time). Ignores differences that are less than the specified value. Default is 0.

- -threshold 1 | 2

  (Optional) Specifies if one or two thresholds should be used when converting analog waveform to digital waveform in digital-analog comparison. Default is 1.

- -upperthreshold *value*

  (Optional) Specifies the value of the upper level to be used for converting an analog waveform to a digital waveform, in case of analog-digital comparison using two thresholds. The default is automatically computed based on input waveforms.

- -lowerthreshold *value*

  (Optional) Specifies the value of the lower level to be used for converting an analog waveform to a digital waveform, in case of analog-digital comparison using two thresholds. The default is automatically computed based on input waveforms.

- -fixedthreshold *value*

  (Optional) Specifies the value of the level to be used for converting an analog waveform to a digital waveform, in case of analog-digital comparison using two thresholds. The default is automatically computed based on input waveforms.

- -refDelay *delay*

  (Optional) Specifies a delay to shift all configured waveforms from the reference dataset. This setting overrides the delay setting of the **compare start** command.

- -testDelay *delay*

  (Optional) Specifies a delay to shift all configured waveforms from the test dataset. This setting overrides the delay setting of the **compare start** command.

- -start *time*

  Specifies the time at which to start the comparison of all configured waveforms. This setting overrides the setting from a **compare run** or **info** command).

- -end *time*

  (Optional) Specifies the time at which to end the comparison of all configured waveforms. This setting overrides the setting from a **compare run** or **info** command.

- -edgecompare

  (Optional) Forces a "digital" comparison where the source and reference signals (one, or both) are analog. This applies an automatic analog to digital conversion, enabling an edge comparison to be made. Digital comparison tolerances will apply for the edge comparison, refer to "Digital Comparison" on page 254.

- ***comparePath***

  (Required) Identifies the path of a compare signal, region, or glob expression.

  You can use the extended options detailed in Selecting Waveforms in Tcl to select the compare signal, region or glob expression. You can use more than one option in the same command.

Description

The modified options are applied to all objects in the specified compare path. These settings are local to the specified compare path; to set default options for all compare paths, use compare options.

Examples

- Modify the previously added tvin waveforms and use a previously created myClock clock to compare them.

  ```
  compare configure -clock myClock tvin
  ```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare end

Closes the active comparison without saving any information.

Usage

```
compare end
```

Parameters

- None.

Examples

- The comparison is ended, all settings and added waveforms are reset. All clocks that were created and computed comparison waveforms remain present in the <calc> database.

  ```
  compare end
  ```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare info

Lists the results of a waveform comparison in the main window transcript. To save the information to a file, use the -write argument.

Usage

```
compare info [-start start_value] [-end end_value]
[-write [-append] filename]
[-rf pssresidue | topnoise | harmonicsmeter]
```

Parameters

- -start *time*

  (Optional) Specifies the time at which to start the comparison of all configured waveforms.

- -end *time*

  (Optional) Specifies the time at which to end the comparison of all configured waveforms.

- -write

  (Optional) Saves the summary information to a file, specified with *filename*, rather than displaying it in the main window transcript.

- -append

  (Optional) Used with the -write argument when saving the summary information to a file. The report is appended to the file specified with *filename*, rather than overwriting it.

- -rf *pssresidue* | *topnoise* | *harmonicsmeter*

  (Optional) Specifies pssresidue, topnoise or harmonicsmeter waveform compare options.

Examples

- Export the results of the comparison (waveform by waveform differences) to the text file report.txt.

  ```
  compare info -write /path/to/the/report.txt
  ```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare list

Outputs the Tcl scripts of all the compare add commands in effect.

Usage

```
compare list [-expand]
```

Parameters

- -expand

  (Optional) Expands groups of waveforms that were added by related individual waveforms.

Examples

- Replace “compare add -r *” by a list of compare add commands, followed by object names.

  ```
  compare list
  ```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare options

Sets defaults for various waveform comparison commands. Those defaults are used when other compare commands are invoked during the current session.

Usage

```
compare options [-maxsignal n] [-maxtotal n] [-mode_name]
[-tol delay] [-tolLead delay] [-tolTrail delay]
[-mixtolLead delay] [-mixtolTrail delay]
[-freqtol freqtol] [-amptol amptol] [-noisefloor noisefloor]
[-matchindex]
[-vhdlmatches {ref-logic-value=test-logic-value:...}]
[-vlogmatches {ref-logic-value=test-logic-value:...}]
[-xTol xtolerance] [-yTol ytolerance] [-mindeltax value] [-threshold 1|2]
[-upperthreshold value] [-lowerthreshold value] [-fixedthreshold value]
[-wavewin name] [-noaddwave] [-addwave] [-adderrorwave]
[-rf pssresidue | topnoise | harmonicsmeter]
[-tperiod value] [-abstolrf value] [-reltolrf value]
[-f0 value] [-nbharmonics value]
[-sortby firstdiff | maxdiff]
[-ntop value] [-annotate][-edgecompare]
```

Parameters

- -maxsignal *n*

  (Optional) Specifies an upper limit for the total differences encountered on any one signal. When that limit is reached, the EZwave viewer stops computing differences on that signal. The default limit is 100.

- -maxtotal *n*

  (Optional) Specifies an upper limit for the total differences encountered. When that limit is reached, the EZwave viewer stops computing differences. The default limit is 1000.

- *-mode*

  (Optional) Specifies the mode of waveform types that are compared with the compare add command. The actual values the option may take are -in, -out, -inout, -internal, -ports, and -all. You can use more than one mode option in the same command. The following modes available in Questa ADMS are also supported: -across, -free, -through, -a2d, -d2a, -bidir.

- -tol *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to lead or trail a reference edge in an asynchronous comparison. The default is 0.

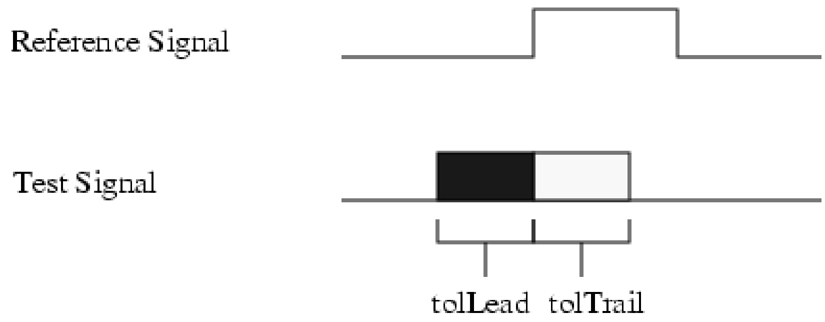
- -tolLead *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to lead a reference edge in an asynchronous comparison. The default is 0.

- -tolTrail *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to trail a reference edge in an asynchronous comparison. The default is 0.

Figure C-3. Graphical Representation of tolLead and tolTrail



- -mixtolLead *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to lead a reference edge in a mixed signal comparison. The default is 0.

- -mixtolTrail *delay*

  (Optional) Specifies the maximum time a test signal edge is allowed to trail a reference edge in a mixed signal comparison. The default is 0.

- -freqtol *freqtol*

  (Optional) Specifies the frequency tolerance for a spectral comparison.

- -amptol *amptol*

  (Optional) Specifies the amplitude tolerance for a spectral comparison.

- -noisefloor *noisefloor*

  (Optional) Specifies the noise floor level for a spectral comparison. Errors that occur below this amplitude level are ignored in the comparison.

- -matchindex

  (Optional) Specifies that only compound waveforms (from multiple runs) with matching indexes are compared. This mode is off by default.

  For example, consider a reference dataset containing only a few runs (for example runs 1, 3 and 5) and a test dataset with many more runs (for example runs 1 to 10). Without the -matchindex option, reference:run_1 is compared with test:run_1, but reference:run_3 is compared with test:run_2, because runs are compared in the order that they are found in the compound waveform. Using the -matchindex option, reference:run_3 would now be compared with the corresponding test:run_3.

- -vhdlmatches {*ref-logic-value*=*test-logic-value*: …}

  (Optional) Specifies how VHDL signal states in the reference dataset should match values in the test dataset. Values are specified in a colon-separated list of match values. For example:

  ```
  -vhdlmatches {X=XUD:Z=ZD:1=1HD}
  ```

Default:

```
{U=UWXD:X=UWXD:0=0LD:1=1HD:Z=ZD:W=UWXD:L=0LD:H=1HD:D=UX01ZWLHD}
```

The 'D' character represents the '-' “don't care” std_logic value.

- -vlogmatches {*ref-logic-value*=*test-logic-value*: …}

  (Optional) Specifies how Verilog signal states in the reference dataset should match values in the test dataset. Values are specified in a colon-separated list of match values. For example:

  ```
  -vlogmatches {0=0:1=1:Z=Z}
  ```

  Default:

  ```
  {0=0:1=1:Z=Z:X=X}
  ```

- -xTol *xtolerance*

  (Optional) Specifies the maximum X tolerance (usually time) that test and reference waveforms are allowed to differ.

  The default:

  $$\text{the minimum of } \left(\frac{\text{x\_max} - \text{x\_min}}{\text{number\_of\_points}}\right) or\ (0.01 \times \text{x\_max} - \text{x\_min})$$

- -yTol *ytolerance*

  (Optional) Specifies the maximum Y tolerance that test and reference waveforms are allowed to differ. The default:

  $$0.01 \times (y\_\text{max} - \text{y\_min})$$

- -mindeltax *value*

  (Optional) Specifies the minimum delta X tolerance (usually time). Ignores differences that are less than the specified value. Default is 0.

- -threshold 1 | 2

  (Optional) Specifies if one or two thresholds should be used when converting analog waveform to digital waveform in digital-analog comparison. Default is 1.

- -upperthreshold *value*

  (Optional) Specifies the value of the upper level to be used for converting an analog waveform to a digital waveform, in case of analog-digital comparison using two thresholds. The default is automatically computed based on input waveforms.

- -lowerthreshold *value*

  (Optional) Specifies the value of the lower level to be used for converting an analog waveform to a digital waveform, in case of analog-digital comparison using two thresholds. The default is automatically computed based on input waveforms.

- -fixedthreshold *value*

  (Optional) Specifies the value of the level to be used for converting an analog waveform to a digital waveform, in case of analog-digital comparison using two thresholds. The default is automatically computed based on input waveforms.

- -wavewin *name*

  (Optional) Specifies the default name of the wave window in which compare differences are viewed.

- -noaddwave

  (Optional) Specifies that the waveform result of the comparison is not displayed.

- -addwave

  (Optional) Plots all compared waveforms.

- -adderrorwave

  (Optional) Plots only erroneous compared waveforms.

- -rf *pssresidue* | *topnoise* | *harmonicsmeter*

  (Optional) Specifies pssresidue, topnoise or harmonicsmeter waveform compare options.

  Note

  Parameters "-sortby" and "-rf" cannot be used simultaneously.

- -tperiod *value*

  (Optional) Specifies the reference time period for the pssresidue waveform compare option.

  Note

  The value is required if "-rf *pssresidue*" is specified in "compare start" or "compare options" commands and ignored for other cases.

- -abstolrf *value*

  (Optional) Specifies the absolute tolerance for the harmonicsmeter and pssresidue waveform compare options.

- -reltolrf *value*

  (Optional) Specifies the relative tolerance for the harmonicsmeter and pssresidue waveform compare options.

- -f0 *value*

  (Optional) Specifies the fundamental frequency for the harmonicsmeter waveform compare option.

  Note

  The value is required if "-rf *harmonicsmeter*" is specified in "compare start" or "compare options" commands and ignored for other cases.

- -nbharmonics *value*

  (Optional) Specifies the number of harmonics for the harmonicsmeter waveform compare option.

  Note
  The value is required if "-rf *harmonicsmeter*" is specified in "compare start" or "compare options" commands and ignored for other cases.

- -sortby *firstdiff* | *maxdiff*

  (Optional) Specifies sorting parameter selection, either sort by first difference or by maximum difference.

  Note
  Parameters "-sortby" and "-rf" cannot be used simultaneously.

- -ntop *value*

  (Optional) Specifies the number of results to plot.

  Note
  The "-ntop" value is only taken into account if the "-rf" or "-sortby" parameter is specified. If "-ntop" is not specified all comparison results are plotted.

- -annotate

  (Optional) Specifies the resulting waveform plots will be annotated with the sorting parameter value.

  Note
  The "-annotate" parameter is only taken into account if the "-rf" or "-sortby" parameter is specified. It is ignored for other cases.

- -edgecompare

  (Optional) Forces a "digital" comparison where the source and reference signals (one, or both) are analog. This applies an automatic analog to digital conversion, allowing an edge comparison to be made. Digital comparison tolerances will apply for the edge comparison, refer to "Digital Comparison" on page 254.

Description

These settings specify the default options for all compare paths. To modify settings local to a specified compare path, use compare configure.

Examples

- Return the current value of all options.

  ```
  compare options
  ```

- Set the maxtotal option to 2000 differences.

```
compare options -maxtotal 2000
```

- Return the current value of the maxtotal option.

```
compare options -maxtotal
```

- Verilog X will now match X, Z, or 0.

```
compare options -vlogmatches {0=0:1=1:Z=Z:X=XZ0}
```

- VHDL std_logic X will now match 'U', 'X', 'W', or 'D'.

```
compare options -vhdlmatches {X=UXWD}
```

- Set the leading tolerance for asynchronous comparisons to 300 picoseconds.

```
compare options -tolLead 300p
```

- Set the trailing tolerance for asynchronous comparisons to 250 picoseconds.

```
compare options -tolTrail 250p
```

- Set the analog tolerances to 0.5% for both x and y axes, do not plot the comparison waveforms and change the number of differences to look for in each waveform to 50.

```
compare options -xTol 0.5% -yTol 0.5% -noaddwave -maxsignal 50
```

- Start the comparison session using comparison switches.

```
set referenceWdb REF.wdb
set RES.wdb

dataset open $referenceWdb
dataset open $testWdb

compare start REF RES
compare options -sortby maxspectrumdiff -phase 1e-6 -annotate
compare add -recursive -show ALL REF/* -show ALL RES/*
compare run
```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare run

Runs the difference computation on the signals selected via a compare add command.

Usage

compare run [*startTime*] [*endTime*]

Parameters

- *startTime*

  (Optional) Specifies when to start computing differences. Optional. Default is zero.

- *endTime*

  (Optional) Specifies when to end computing differences. Optional. Default is the end of the dataset simulation run that ends earliest.

Description

The **compare run** command runs the difference computation on the signals selected via a compare add command. Reports in the transcript pane the total number of errors found.

Examples

- Run the comparison, all waveforms that were added to the comparison will now be compared. For each reference/test pair, a comparison waveform named "compare_ref_test" is created.

  ```
  compare run
  ```

- Run the comparison from 5.3n to 57m

  ```
  compare run 5.3n 57m
  ```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare savelog

Creates a log file containing error and warning messages generated during waveform comparison.

Usage

```
compare savelog filename
```

Parameters

- ***filename***

  (Required) Specifies the name and path of the file in which to save the log file.

Description

Creates a log file containing error and warning messages generated during waveform comparison.

Note

A comparison must have been run or a report had been exported within the same comparison session before this command is called. Otherwise, nothing is generated.

Examples

- Save a log file to a text file named *comparelog.txt*.

  ```
  compare savelog /user/comparelog.txt
  ```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare saverules

Saves the rules (all compare add commands in effect and compare options) to a file. Can also be used to save the entire comparison session.

Usage

compare saverules [-session] [-expand] ***rules_filename***

Parameters

- -session

  (Optional) Saves the entire comparison session, not just the rules.

- -expand

  (Optional) Expands groups of related waveforms into their individual waveforms.

- ***rules filename***

  (Required) Specifies the name and path of the file in which to save the rules file.

Description

Saves the rules (all compare add commands in effect and compare options) to a file. Can also be used to save the entire comparison session.

Examples

- Save rules into a file called *rules.rul*.

  ```
  compare saverules rules.rul
  ```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

compare start

Begins a new dataset comparison.

Usage

```
compare start [-batch] [-maxsignal n] [-maxtotal n] [-refDelay delay]
[-testDelay delay] reference_dataset [test_dataset]
[-rf pssresidue | topnoise | harmonicsmeter]
```

Parameters

- -batch

  (Optional) Specifies that comparisons will not be automatically inserted into the wave window.

- -maxsignal *n*

  (Optional) Specifies an upper limit for the total differences encountered on any one signal. When that limit is reached, the EZwave viewer stops computing differences on that signal. The default limit is 100.

- -maxtotal *n*

  (Optional) Specifies an upper limit for the total differences encountered. When that limit is reached, the EZwave viewer stops computing differences. The default limit is 1000.

- -refDelay *delay*

  (Optional) Specifies a delay to shift all compared waveforms from the reference dataset.

- -testDelay *delay*

  (Optional) Specifies a delay to shift all compared waveforms from the test dataset.

- ***reference_dataset***

  (Required) The reference dataset to be used as the comparison reference.

- *test_dataset*

  (Optional) The dataset to be tested against the reference. If not specified, The EZwave viewer uses the current simulation. The reference and test datasets may be the same.

- -rf *pssresidue* | *topnoise* | *harmonicsmeter*

  (Optional) Specifies pssresidue, topnoise or harmonicsmeter waveform compare options.

Description

The **compare start** command begins a new dataset comparison. The datasets that you will be comparing must already be open.

Examples

- Begin a waveform comparison between a dataset named "gold" and the current simulation. Assumes the gold dataset was already opened.

```
compare start gold
```

- This command sequence opens two datasets and starts a comparison between the two using greater than default limits for total differences encountered.

```
dataset open gold_typ.wdb gold
dataset open bad_typ.wdb test
compare start -maxtotal 5000 -maxsignal 1000 gold test
```

- Start the comparison session and “align” waveforms delaying reference.

```
compare start -refdelay 5n  REF TEST
```

- Start the comparison session and perform pssresidue comparison.

```
set referenceWdb test.wdb
dataset open $referenceWdb

compare start test -rf pssresidue
compare options -tperiod 1e-9 -annotate
compare add -recursive -show TRAN test/*
compare run
```

- For more detailed examples, see Waveform Comparison With Tcl Examples.

dataset alias

Adds additional names (aliases) to a dataset.

Usage

`dataset alias` ***`dataset_name`*** `[`*`alias_name`*` [-folder `*`folder_path`*`]]`

Parameters

- ***dataset_name***

  (Required) Specifies the name of the dataset to which the alias is assigned. Use the root name of the file only. This must be specified as the first argument to the database alias command.

- *alias_name*

  (Optional) Specifies the alias name to assign to the dataset. It also returns a list of all aliases currently assigned to the specified dataset.

- -folder *folder_path*

  (Optional) Specifies the folder name(s) in the dataset where the alias name will be assigned. If several folder levels are encountered, names are separated by a forward slash (/).

Return Values

A list of aliases currently assigned to the dataset.

Examples

- Assign the alias name "REF" to the dataset named "gold".

  ```
  dataset alias gold REF
  ```

- Assigns the alias name "myfolder" to the folder named "saved", which exists in the dataset "adc12". This is equivalent to <adc12/saved> in the EZwave fully qualified name notation.

  ```
  dataset alias adc12 myfolder -folder saved
  ```

dataset analysis

Returns a list of folders (or subfolders) of analysis of the dataset (or folder).

Usage

```
dataset analysis [dataset_name] [-folder folder_path]
```

Parameters

- *dataset_name*

  (Optional) Specifies the name of the dataset on which to list the analysis.

  > **Note**
  > If no argument is supplied, the command is applied to the dataset that was last opened.

- -folder *folder_path*

  (Optional) Specifies the folder name existing in the dataset on which to list the analysis. If several folder levels are encountered, names are separated by a forward slash (/).

Return Values

A list of dataset folders found.

Description

Returns a list of folders (or subfolders) of analysis of the dataset (or folder).

Dataset folders are all considered as analysis by this command, even if they do not always correspond to analysis.

Examples

```
dataset analysis adc12
```

dataset clear

Clears all waveform data from the last opened database.

Usage

`dataset clear`

Parameters

- None.

Examples

```
dataset clear
```

dataset close

Closes the specified database or all databases.

Usage

dataset close [*database_name* | -all]

Parameters

- *database_name*

  (Optional) Specifies the name of the database to be closed.

- -all

  (Optional) Specifies that all open databases should be closed.

Description

Closes the specified database or all databases.

Note
If no argument is supplied, the database that was opened last is closed.

Examples

```
dataset close -all
```

dataset extract

Runs Eldo -extract on the source database using the source netlist to define the .extract and .meas commands.

Usage

```
dataset extract -source <wdb> -netlist <path_to_netlist> -out
<dest_aex_file> [-wave <dest_wdb_file>] [-overwrite] [-arguments
<arguments_string>] [-save <text_or_csv_file>]
```

Parameters

- **-source**

  (Required) Specifies the path to the database file (*.wdb*) containing the simulation results to be processed.

- **-netlist**

  (Required) Specifies the path to the netlist containing the .extract and .meas commands to run.

- **-out**

  (Required) Specifies the path to the aex file into which extracted scalar values are written.

- -wave

  (Optional) Specifies the path to the database file (*.wdb*) into which the waveform data is written.

- -overwrite

  (Optional) Instructs Eldo to overwrite any existing files. When this is not specified, EZwave reports an error if an output file already exists.

- -arguments

  (Optional) Enables you to define parameters to pass to Eldo.

- -save

  (Optional) Saves the content of the scalar extract in a text or CSV file using EZwave standards (headers, separators, and so on).

Examples

```
dataset extract -source test_data.wdb -netlist test_data.cir -out
test_data.aex -wave test_data_extract.wdb -overwrite -save file.txt
```

dataset info

Returns the specified information about a database.

Usage

```
dataset info option [database_name]
```

Parameters

- ***option***

  (Required) Specifies the type of information to be returned. ***option*** can be one of the following:

  - **name** — Returns the name of the database.
  - **file** — Returns the path and filename where the database is saved.
  - **exists** — Returns *1* if the database is currently open; otherwise, returns *0*.

- *database_name*

  (Optional) Specifies the name of the database to get information about. If *database_name* is not specified, the active database is used.

Examples

```
if [dataset info exists adc12 ] {
   # do something
} else {
   # do something else
}
```

dataset list

Returns a list of currently opened databases.

Usage

```
dataset list [-long]
```

Parameters

- -long

  (Optional) If this option is specified, this command also returns the full path of the databases.

Examples

```
dataset list
```

dataset merge

Merges databases containing single run simulation results, and generates compound waveforms from found waveforms. When supplied the name of an index file, the command merges the referenced databases and adds parameters as defined in the file.

Usage

```
dataset merge dataset_name file_name1 [... file_namen] | -wdbparam
<index_file> [-matchindex |-startindex start_index] [-ignorewaveforms]
```

Parameters

- **dataset_name**

  (Required) Specifies the name of the dataset containing the compound waveforms resulting from the merge.

  Caution

  You must choose a *dataset_name* that is different from the names of any of the source databases to prevent overwriting the source with the resulting merged dataset.

- **file_name1** [... file_namen]

  (Required; either a file_name or a -wdparam file must be specified) Specifies the list of databases to be merged. You must specify at least one file. The wildcard character (*) is allowed in file names. It is not necessary to open the databases prior to this command call.

- **-wdbparam** *<index_file>*

  (Required; either a file_name or a *.wdbparam* file must be specified) Specifies the name of the index file to be used in defining the merge. The index file should contain a list of the waveform output files to merge, along with any parameters to associate with the resulting compound waveform. If used, the ***file_name1*** [... file_namen] are ignored.

  When -wdbparam is used and ***dataset_name*** is specified, the merged dataset is named according to that value. If ***dataset_name*** is not specified, the default of "merge_" + number is used for the merged dataset name.

  If -wdbparam is used, -startindex start_index is ignored.

  If -wdbparam is used with -matchindex, EZwave tries to determine the index of the compound element by the database name used in the index file.

  Note

  AC_n folders (or other *<AnalysisName>_n* folders) are considered as different analyses in the merge when AC (or other analysis name) is specified in the *.wdbparam* file (for example CUSTOM_BEGIN FOLDER=AC).

- -matchindex

  (Optional) If this option is specified, the index of the compound element must be determined by its source database name. For example, *test1.wdb* -> index 1, *test.wdb* -> index 0 or even *test.tr2* -> index 2.

- -startindex *start_index*

  (Optional) Specifies the index that the compound elements should start from. For example, merging *test.wdb* and *test2.wdb* using option -startindex 7 will create compounds with element index 7 and 8.

  Note

  This option is ignored when merging databases that already contain compound waveforms. In this case, the original compound element indexes are maintained.

- -ignorewaveforms

  (Optional) This option is *only* used if a user encounters compatibility issues with the updated dataset merge functionality and needs to revert to the previous behavior of dataset merge. See note.

  Note

  In AMS13.2 and earlier versions, EZwave gave priority to compound waveforms in the source databases when merging. For example, for a database containing a TRAN folder with compound waveforms and an EXT folder with regular waveforms, only the TRAN folder was merged. For later versions of AMS, in the same situation, EZwave will merge both the TRAN and EXT folders. In certain situations, this *might* affect a user script that had worked previously, because some folders that used to be ignored are now merged. If this situation should occur, the option -ignorewaveforms may be used to revert to the previous behavior.

Description

This command merges databases containing single run simulation results, and generates compound waveforms from found waveforms. When supplied the name of an index file, the command merges the referenced databases and adds parameters as defined in the file.

Examples

```
dataset merge result /user/simu*.wdb
# Merge waveforms contained in all databases matching
# "/user/simu*.wdb", creating a compound waveform in the dataset
# "result".
```

Note

Merging of FSDB files is not supported by the dataset merge command. You must convert the files to WDB format (*.wdb*) before merging using dataset merge:

ffcv <file.fsdb> -jwdb <file.wdb>

dataset mergewaveforms

Merges waveforms, grouping waveforms of the same name as if there were multiple simulation runs.

Usage

```
dataset mergewaveforms dataset_name file_name1 ... file_namen
[-matchindex | -startindex start_index]
-wf waveform1_name ... waveform_n_name
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- ***dataset_name***

  (Required) Specifies the name of the dataset containing the compound waveforms resulting from the merge.

  Caution

  You must choose a ***dataset_name*** that is different from the names of any of the source databases to prevent overwriting the source with the resulting merged dataset.

- ***file_name1*** … ***file_namen***

  (Required) Specifies the list of databases to be merged. The wildcard character (*) is allowed in file names. It is not necessary to open the databases prior to this command call.

- -matchindex

  (Optional) If this option is specified, the index of the compound element must be determined by its source database name. For example, *test1.wdb* -> index 1, *test.wdb* -> index 0 or even *test.tr2* -> index 2.

- -startindex *start_index*

  (Optional) Specifies the index that the compound elements should start from. For example, merging *test.wdb* and *test2.wdb* using option -startindex 7 will create compounds with element index 7 and 8.

  Note

  This option is ignored when merging databases that already contain compound waveforms. In this case, the original compound element indexes are maintained.

- **-wf *waveform1_name*** … ***waveform_n_name***

  (Required) Specifies the list of waveforms to be merged. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveforms. You can use more than one option in the same command.

Description

This command merges waveforms, grouping waveforms of the same name as if there were multiple simulation runs. The x-axis units of both waveforms must be identical.

Examples

```
dataset mergewaveforms mywdb /user/library/*.wdb -wf IN OUT
```

dataset open

Opens a database file.

Usage

```
dataset open file_name [dataset_name] [-incl_if_tandem {on | off}]
```

Parameters

- ***file_name***

  (Required) The name of a database in a supported format.

- *dataset_name*

  (Optional) Specifies a name for the open dataset. This is a name that will identify the dataset in the current session. If not specified, the default dataset name corresponds to the filename prefix (without file extension).

- -incl_if_tandem {*on* | *off*}

  (Optional) When set to off, excludes a database from tandem mode when loaded.

Return Values

The database name.

Examples

```
dataset open $env(AMS_VIEWER_HOME)/lib/tutorial.wdb
```

dataset ovd

Opens an OVD file.

Usage

```
dataset ovd path_to_ovd_folder [-ovdindex <index_representation>]
```

Parameters

- ***path_to_ovd_folder***

  (Required) Specifies the path to an OVD folder.

- *-ovdindex <index_representation>*

  (Optional) Specifies the iteration that should be opened. Indices are separated by commas, can be a single integer number, or a range of indices separated by “-”. A valid index_representation is 1,3,5-13,17.

Examples

```
dataset ovd netlist.out -ovdindex 12,17,51
```

Related Topics

Foreign Databases Options

dataset power analysis

Performs a power analysis of the specified waveform database power analysis folder.

Usage

dataset power analysis wdb analysis [path [-separator <character>]] [-run <index>] [-list] [-range lower_bound upper_bound] [-upperthreshold <value>] [-lowerthreshold <value>] [-save <file.[txt|csv]> [-overwrite]] [-depth <value>] [-view name1 name2] [-filter name regex] [-sort name <up|down>] [-ignoredefaultfiltering] [-dialog]

Parameters

- wdb

  Name of the waveform database to be analyzed.

- analysis

  Name of the analysis folder to be analyzed.

- path

  (Optional) The path to the subcircuit to be considered as the top for the analysis. When not specified, the default top subcircuit is the one found in the analysis folder.

- -separator <character>

  (Optional) Path separator character.

- -run <index>

  (Optional) Used to specify a run id when performing a power analysis on a multi-run database.

- -list

  (Optional) Lists all names available in the database. Used when setting up the batch analysis.

- -range <lower_bound> <upper_bound>

  (Optional) Defines the lower and upper time bounds of the analysis. It is possible to analyze a single time event by setting the same time value for lower and upper bounds. When not specified, the analysis is performed on the whole simulation time.

- -upperthreshold <value>

  (Optional) Defines the upper threshold to filter devices and subcircuits on power consumption value.

- -lowerthreshold <value>

  (Optional) Defines the lower threshold to filter devices and subcircuit on power consumption value.

- -save <file.[txt|csv]>

  (Optional) Saves the result of the analysis to a text or CSV file. When this switch is not specified in batch mode, the command does nothing.

- -overwrite

  (Optional) Overwrite the output file if it exists. When not specified and the file exists, the command will exit and generate an error.

- -depth <value>

  (Optional) Defines a maximum depth of subcircuit analysis from the top. When not specified, no depth limitation is used and all subcircuits of the top are analyzed.

- -view {"column_name1" "column_name2"…}

  (Optional) Defines the names of the columns to display in the Power Analysis Dialog Box. When not specified, the default display depends on the simulator setup stored on the analysis folder.

- -filter "name" "regex"

  (Optional) Defines a filtering regular expression on a column. This switch can be used multiple times on the same command to define filters on multiple columns.

- -sort "column_name" <up|down>

  (Optional) Defines a sorting order on a column. This switch can be used multiple times on the same command but only the last one defined will be applied.

- -ignoredefaultfiltering

  (Optional) Discard the settings that were set at simulation time and use the default ones. See .POWER_ANALYSIS in the *Eldo Reference Manual*.

- -dialog

  (Optional) When used with the EZwave GUI, opens the Power Analysis dialog box through the TCL command.

Examples

Example 1

```
dataset open tc100_eldo.wdb
dataset power analysis tc100_eldo POWER_TRAN -dialog
```

- This example runs the analysis of the *tc100_eldo* database on the *POWER_TRAN* folder.
- All devices and subcircuits contributions are available in the EZwave Power Analysis Dialog Box.

Example 2

```
dataset open tc100_eldo.wdb
dataset power analysis tc100_eldo POWER_TRAN -dialog tc100_eldo:X1:X2
-separator :
```

- This example runs the analysis of the *tc100_eldo* database on the *POWER_TRAN* folder.
- The top level is not *tc100_eldo* anymore but *tc100_eldo.X1.X2* subcircuit. The separator is not necessarily the same as the one used by EZwave (waveform name).

Example 3

```
dataset power analysis tc100_eldo POWER_TRAN -range 80e-9 100e-9
-lowerthreshold 0.005 -depth 3
```

- This example runs the analysis of the *tc100_eldo* database on the *POWER_TRAN* folder.
- The analysis is performed over the time range 80 to 100 ns.
- All subcircuits from the top one are analyzed up to a depth of 3 levels.
- All devices and subcircuits consuming less than 0.005W will not be shown in the displayed results.

Tip

See also .POWER_ANALYSIS in the *Eldo Reference Manual*.

dataset rename

Renames the specified database.

Usage

`dataset rename` ***`database_name new_name`***

Parameters

- ***database_name***

  (Required) Specifies the database to be renamed.

- ***new_name***

  (Required) Specifies the new name for the database.

Return Values

The database name.

Examples

```
dataset rename adc12 adc12_test
```

dataset save

Saves the specified database to disk as the specified file name.

Usage

```
dataset save database_name file_name [-start x_start] [-end x_stop]
[-sampling sample] [-atcursors] [-delta delta] [-deriv derivabs]
[-derivrel derivrel]
```

Parameters

- ***database_name***

  (Required) Specifies the database to be saved.

- ***file_name***

  (Required) Specifies the path and file name where the database should be saved. The extension used for the file name determines the output format (for example, *myfile.txt* will save to a text file, *myfile.csv* will save to a CSV file).

  Add the *.gz* extension to the filename to save the waveforms to a compressed (gzipped) archive.

- -start *x_start*

  (Optional) Specifies the X value to be considered as the start value of the database to be saved.

- -end *x_stop*

  (Optional) Specifies the X value to be considered as the end value of the database to be saved.

- -sampling *sample*

  (Optional) Specifies the step value to be used for sampling while saving the database.

- -atcursors

  (Optional) Use this option to save only the waveform data at cursor locations.

- -delta *delta*

  (Optional) For file type Spice PWL (.sti) only, use this option to specify the threshold the lower limit threshold and enable filtering of output points by performing an absolute variation comparison.

- -derivabs *derivabs*

  (Optional) For file type Spice PWL (.sti) only, use this option to specify the absolute tolerance and enable filtering of output points by performing a slope based comparison.

- -derivrel *derivrel*

  (Optional) For file type Spice PWL (.sti) only, use this option to specify the relative tolerance and enable filtering of output points by performing a slope based comparison.

Note

Combining switches [-delta *delta*], [-derivabs *derivabs*] and [-derivrel *derivrel*]is not forbidden, but the outcome of doing so is difficult to predict. It is advised that only one switch is used at a time.

Examples

```
dataset save calc /user/ADC12/calc.wdb

dataset save meas /user/db/meas1n.wdb -sampling 1n
# Saves database with a sampling of 1n (second) to the file meas1n.wdb

dataset save dcsweep /user/dc/dcsweep.wdb.gz
# Saves database to the gzipped file dcsweep.wdb.gz
```

dataset savescalar

Save scalars, such as noise contributors, to a TXT file or CSV file.

Usage

```
dataset savescalar <wdb>
[<folder> ... <folderN>] -output <report_file>
[-sort <name|value> <asc|desc>]
```

Parameters

- ***<wdb>***

  (Required) Dataset name, opened in EZwave.

- *<folder> ... <folderN>*

  (Optional) Name(s) of the folder(s) (type(s)) containing the variables to be saved. If not specified, all variable types will be saved.

- **-output *<report_file>***

  (Required) The path to the output file.

- -sort *<name|value> <asc|desc>*

  (Optional) Sorting parameter (name or value) and direction (ascending or descending).

Description

The output file format (value separator) is detected automatically depending on the output file extension. If the extension is *.csv, the format will be CSV (separator is a comma). In other cases the format will be TXT (separator is a whitespace).

Examples

Example of the TXT report:

```
Name Value Y Unit
pss_PhaseNoise_SummaryStatistics
periodJitter 5.542636185984542E-13
periodJitter_flicker 5.2546672272391475E-14
periodJitter_white 5.51767166597425E-13
frequency 8.0E8
period 1.25E-9ss
```

Example of the CSV report:

```
Name,Value,Y Unit
pss_PhaseNoise_SummaryStatistics,,
periodJitter,5.542636185984542E-13,
periodJitter_flicker,5.2546672272391475E-14,
periodJitter_white,5.51767166597425E-13,
frequency,8.0E8,
period,1.25E-9,
```

dataset savewaveforms

Saves the specified waveform(s) to disk as the specified file name.

Usage

```
dataset savewaveforms file_name [-start x_start] [-end x_stop]
[-sampling sample] [-atcursors] [-delta delta] [-derivabs derivabs]
[-derivrel derivrel] waveform_name
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- ***file_name***

  (Required) Specifies the path and file name where the database file should be saved. The extension used for the file name determines the output format. For example, *myfile.txt* will save to a text file, *myfile.csv* will save to a CSV file, and *myfile.sti* will save time-based waveforms only to a SPICE PWL file.

  Add the *.gz* extension to the filename to save the waveforms to a compressed (gzipped) archive.

- -start *x_start*

  (Optional) Specifies the X value to be considered as the start value of the database to be saved.

- -end *x_stop*

  (Optional) Specifies the X value to be considered as the end value of the database to be saved.

- -sampling *sample*

  (Optional) Specifies the step value to be used for sampling while saving the database.

- -atcursors

  (Optional) Use this option to save only the waveform data at cursor locations. This parameter is not supported in batch mode.

- -delta *delta*

  (Optional) For file type Spice PWL (.sti) only, use this option to specify the threshold the lower limit threshold and enable filtering of output points by performing an absolute variation comparison.

- -derivabs *derivabs*

  (Optional) For file type Spice PWL (.sti) only, use this option to specify the absolute tolerance and enable filtering of output points by performing a slope based comparison.

- -derivrel *derivrel*

  (Optional) For file type Spice PWL (.sti) only, use this option to specify the relative tolerance and enable filtering of output points by performing a slope based comparison.

- ***waveform_name*** [*waveform_name2…*]

  (Required) Specifies the waveform(s) to be saved to the database file. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform(s). You can use more than one option in the same command.

> **Note**
> Combining switches [-delta *delta*], [-derivabs *derivabs*] and [-derivrel *derivrel*] is not forbidden, but the outcome of doing so is difficult to predict. It is advised that only one switch is used at a time.

Examples

```
dataset savewaveforms /user/ADC12/calc.wdb v_load

dataset savewaveforms /user/db/wave1n.wdb -sampling 1n Q0 IN
# Saves the waveforms Q0 and IN with a sampling rate of 1n (second) to
# the file wave1n.wdb

dataset savewaveforms /user/db/VQ0.wdb.gz Q0
# Saves the waveform Q0 to the gzipped file VQ0.wdb.gz
```

dataset scalar

Compares scalar values of variables stored in two databases (PSF, nutmeg, FSDB, WDB, OVD).

Usage

```
dataset scalar ref_dataset test_dataset -output <report_file>
   [-type <name>] [-cs 0|1] [-nameonly|-tol <value>]
```

Parameters

- ***ref_dataset***

  (Required) Specify the reference dataset name.

- ***test_dataset***

  (Required) Specify the test dataset name.

- **-output <*report_file*>**

  (Required) Specify the path for the output report file

- -type <*name*>

  (Optional) Name of the scalar’s type (folder to compare). If absent, all scalar types will be compared.

- -cs *0|1*

  (Optional) Case sensitivity for scalar names. Default is 1 (case sensitivity is on).

- -nameonly

  (Optional) If specified, only the presence of scalars in the reference and test datasets will be compared, not the values.

  Note

  -nameonly switch cannot be used simultaneously with the -tol switch.

- -tol <*value*>

  (Optional) Tolerance value (for absolute difference calculation) for scalar comparisons. A percentage value is also supported. If absolute difference (abs(test_value – ref_value)) is greater than the specified tolerance, the difference will be included into the report.

  Note

  -tol switch cannot be used simultaneously with the -nameonly switch.

Return Values

The dataset scalar command generates a text report in the following form:

```
Missing scalars in <test> = <number_of_missing>Folder: <folder_name>
```

```
    <scalar1_path> = <scalar1_value>…<scalarN_path> = <scalarN_value>

Additional scalars in <test> = <number_of_additional>Folder: <folder_name>

    <scalar1_path> = <scalar1_value>…<scalarN_path> = <scalarN_value>

Total differences = <number_of_differences>Folder: <folder_name>

    Diff number 1Reference variable : <ref_scalar1_path> =
    <ref_scalar1_value>Test variable : <test_scalar1_path> =
    <ref_scalar1_value>Absolute difference = <value1>Relative difference =
    <value1>…Diff number NReference variable : <ref_scalarN_path> =
    <ref_scalarN_value>Test variable : <test_scalarN_path> =
    <ref_scalarN_value>Absolute difference = <valueN>Relative difference =
    <valueN>
```

If there are no differences found, the report will contain:

```
Missing scalars in <test> = 0Additional scalars in <ref> = 0Total
differences = 0
```

Description

The function takes the two databases names to compare, a tolerance value and generates a report of differences when variables do not match between the two databases being compared.

Absolute difference is calculated using the formula:

abs_diff = abs(test_value – ref_value)

Relative difference is calculated using the formula:

rel_diff = abs(test_value – ref_value)/min(test_value, ref_value)

If the absolute difference is greater than the tolerance, the difference will be included into the report.

dataset statistics

Returns statistics on the dataset.

Usage

```
dataset statistics [dataset_name] [-requestcompleteload]
```

Parameters

- *dataset_name*

  (Optional) Specifies the name of the dataset on which to return the statistics.

  Note

  If no argument is supplied, the command is applied to the dataset that was last opened.

- -requestcompleteload

  (Optional) If specified, forces EZwave to load an entire database to obtain a complete set of statistics.

  Note

  To improve performance, EZwave does not always load an entire database by default. This depends on the database type (for example PSF and FSDB). Depending on the size of the database, this might take some time and use a large amount of memory.

Return Values

A list of dataset statistics.

Description

Outputs statistics on the dataset.

Examples

```
set myinfo [dataset statistics]
puts $myinfo

Statistics for tutorial (/ams/product/.../lib/lib/tutorial.wdb)
-------------
Number of folders: 2
Number of runs: 0
Number of hierachies: 1
Number of waveforms: 18
Number of aliases: 0
Number of bus: 1
Number of bits: 12
Number of compounds: 0
Number of links: 0
Number of datapoints: 29438
```

In this example, the -requestcompleteload option is not necessary, as EZwave always loads a complete *.wdb* dataset.

Related Topics

Database Properties Dialog Box

dataset supported

Returns “true” if the dataset can be opened by EZwave, and “false” otherwise.

Usage

```
dataset supported dataset_path
```

Parameters

- ***dataset_path***

  (Required) Specifies the path for the dataset to be checked.

dataset topnoise

Performs a topnoise calculation on the specified waveform database folder or hierarchy. Returns noise spectrum in sorted order of spot noise if only *fstart* is specified, or integrated noise if both *fstart* and *fstop* are specified.

Usage

dataset topnoise <*wdb_obj*> *-fstart* <*value*> [-fstop <value>] [-ntop <value>] [-pattern <value>] [-norecursive] [-depth <value>] [-runs <runID>,<runID1>,...] [-instance] [-file <path>] [-noplot] [-nogrid]

Parameters

- <wdb_obj>

  (Required) Name of the waveform database folder or hierarchy. The format is as follows:

  ```
  <database_name>/<folder_or_hierarchy_name1>/<folder_or_hierarchy_name2>/
  ...
  ```

- -fstart <value>

  (Required) Specifies the lower frequency limit for integrated noise or the single frequency value at which to report spot noise.

- -fstop <value>

  (Optional) Specifies the upper frequency limit. If specified, integrated noise is calculated. If not specified, spot noise at *fstart* is calculated.

- -ntop <value>

  (Optional) The number of results to plot. If not specified, the default is 8. Ignored if -noplot switch is used.

- -pattern <value>

  (Optional) The waveform search pattern. Default is "*".

- -norecursive

  (Optional) Disables the recursive waveform search. Cannot be used simultaneously with -depth switch.

- -depth <value>

  (Optional) Waveforms search depth (for a recursive search). If not specified (default) it performs the full depth search. Cannot be used simultaneously with -norecursive switch.

- -runs <runID>,<runID1>,...

  (Optional) A comma-separated list of run ids to process for a multiple run database. If not specified (default) all runs will be processed.

- -instance

  (Optional) Specifies that report data is generated not only at MOS instance level, but recursively at subckt instance level.

- -file <path>

  (Optional) Specifies the path and filename where a sorted report of all noise contributions is saved.

- -noplot

  (Optional) Specifies that no topnoise results are plotted.

- -nogrid

  (Optional) Disables the display of grid lines. Ignored if -noplot switch is used.

Related Topics

topnoise

Top Noise Dialog Box

delete wave

Removes the specified waveform from the indicated window.

Usage

```
delete wave [-window window_name] waveform_name
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- *window_name*

  (Optional) Specifies the window from which to remove the waveform. If *window_name* is not specified, the waveform is removed from the active window.

- ***waveform_name***

  (Required) Specifies name of the waveform to be removed. You can specify multiple waveforms and use wildcards. You can also use the extended options detailed in Selecting Waveforms in Tcl to select the waveform(s). You can use more than one option in the same command.

Examples

```
delete wave -run 1 -session current <tutorial/Time-Domain_Results>v_load
```

dofile

Tells the EZwave Tcl interpreter to source the specified Tcl file.

Usage

```
dofile file_name
```

Parameters

- ***file_name***

  (Required) Specifies the path and file name of the Tcl file to be sourced by the EZwave Tcl interpreter.

Return Values

The string returned by the Tcl file, if any.

Description

This command tells the EZwave Tcl interpreter to source the specified Tcl file. This command differs from the Tcl "source" command in that it enables the Tcl "proc" to register into the EZwave Waveform Calculator as a user-defined function.

Examples

```
dofile /user/adc12.tcl
```

environment

Displays or changes the region/signal environment.

Usage

```
environment [path_name] [-dataset | -nodataset]
env [path_name] [-dataset | -nodataset]
```

Parameters

- *path_name*

  (Optional) Specifies the absolute path name to which the current region/signal environment is to be changed. Relative path names are not allowed.

- -dataset

  (Optional) If this option is specified, the database path is displayed, followed by the design hierarchy, as it would appear in the EZwave waveform list.

- -nodataset

  (Optional) If this option is specified, the path is displayed without any database indications.

Return Values

Returns the path with regards to the hierarchy.

Description

This command displays or changes the region/signal environment. If a path name is specified, the current region/signal environment is change to that path name. Otherwise, the current region/signal environment is displayed.

Note

When the -dataset option is used, this command also displays the database name and hierarchy as it is shown in the database.

Examples

```
env :test:u1

# This changes the current region/signal environment path to :test:u1
# Returns :test:u1

env -dataset

# Returns <adc12_test/TRAN>:test:u1
```

evalExpression

Invokes the EZwave Waveform Calculator to calculate the expression entered.

Usage

```
evalExpression expression
```

Parameters

- ***expression***

  (Required) An expression supported by the EZwave calculator.

Return Values

The result value of the expression. It can be one of the following:

- a list of Y-values corresponding to a waveform, represented as a Tcl list of strings
- a single number
- a list of data represented as a Tcl list of strings for a one-dimensional array
- a list of data pairs represented as a Tcl list of string pairs grouped by parentheses for a two-dimensional array

Description

This command invokes the EZwave Waveform Calculator to calculate the expression entered.

If ***expression*** is enclosed in double quotes (**" "**), value substitution is enabled and all strings beginning with a dollar sign (**$**) are replaced by the variable they name. If an expression is enclosed in braces (**{ }**), value substitution is disabled and the expression is evaluated as presented.

This command differs from wfc in the type of output it returns.

Examples

```
evalExpression {wf_diff = wf("<tutorial/Time-Domain_Results>v_load") -
wf("<tutorial/Time-Domain_Results>v_middle")}
```

```
set diff0 [evalExpression {wftodata(wf_diff)[0][0]}]
```

examine

Examines an object and displays its current value in the Transcript Window.

Usage

```
examine
[-env path]
[-name]
[-symbolic | -onescomplement | -magnitude | -binary |
 -octal | -decimal | -hexadecimal | -unsigned | -ascii ]
[-radix radix_type]
[-showbase]
[-tograydec]
[-graytodec]
[-fpoint n]
[-time time]
[wf_name_1 ... wf_name_n] | [-separator separator_char wf_name_1 ...
 wf_name_n [-noseparator wf_name_1 ... wf_name_n]]
```

Parameters

- -env *path*

  (Optional) *path* is the hierarchical path to the objects. You can use this to avoid repeating it in all object names.

- -name

  (Optional) Causes the name of the object to be returned in a TCL list, as well as its value.

- -symbolic | -onescomplement | -magnitude | -binary | -octal | -decimal | -hexadecimal | -unsigned | -ascii

  (Optional) Specifies a radix for displaying digital bus(es) specified in this command.

  If no radix is specified for an enumerated type, the default radix is used. You can change the default radix for the current simulation using the radix command.

- -radix *radix_type*

  (Optional) Specifies a radix or a user-defined radix. Can be used as an alternative to the options:

  ```
  [-symbolic | -onescomplement | -magnitude | -binary | -octal |
  -decimal | -hexadecimal | -unsigned | -ascii ]
  ```

  Legal *radix_type* values: symbolic, onescomplement, magnitude, binary, octal, decimal, hexadecimal, unsigned, ascii.

- -showbase

  (Optional) Display the length of the bus and its base. It uses the syntax defined in radix.

- -time *time*

  Specifies the time value between 0 and $now at which to examine the objects.

time is a non negative integer. By default it is in seconds. For other units the value and unit must be placed inside braces if there is a space between the time and the unit. For example, the following are equivalent for ps resolution:

```
examine -time {3.6 ns} signal_a
examine -time 3600 signal_a
```

If -time *time* is not specified, the last value of each object is returned.

If an expression is specified, it will be evaluated at that time.

Note

When an object does not have data at the specified time, no error is returned. Instead the command returns “No_Data”.

- -tograydec

  (Optional) Converts decimal representation radix to gray radix.

- -graytodec

  (Optional) Coverts gray radix to decimal representation radix.

- -fpoint *n*

  (Optional) Specifies *n* fixed-point radix.

- *wf_name_1 . . .wf_name_n*

  (Optional) Specifies the waveform(s) using one of several accepted syntaxes. See Specifying Waveforms in Tcl. You must use the object name method for specifying waveforms when using the –show representation option.

  You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform(s). You can use more than one option in the same command.

  The examine command will return “No_Data” for any waveform not of time domain, in other words not having seconds as X.

Examples

```
examine -time 500ps *

examine -time 500ps -name *

examine -time 500ns -showbase -name :top:*

examine -time 500ns -showbase -name *

examine -time 5000000ns -name *
```

Tip

See also examine in the *Questa ADMS Command Reference*.

exit

Exits the EZwave viewer.

Usage

```
exit [-force] [-discard]
```

Parameters

- -force

  (Optional) If specified, the EZwave viewer quits and bypasses the confirmation dialog box. If this option is not specified, the confirmation dialog box displays and requires a manual confirmation.

- -discard

  (Optional) If specified, the EZwave viewer quits and bypasses the unsaved data dialog box (any unsaved data is lost). If this option is not specified, the unsaved data dialog box displays and requires a manual confirmation.

Caution

If the -discard option is specified, all unsaved data is lost.

Return Values

None.

Examples

```
exit
```

```
exit -force
```

```
exit -force -discard
```

find analogs

Returns the names of all the analog objects (quantities and terminals) matching the specified pattern.

Usage

```
find analogs object_name ...
```

Parameters

Note
This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- ***object_name***

  (Required) Specifies the name of an object you want to search. Multiple names and wildcards are supported. You can use the extended options detailed in Selecting Waveforms in Tcl to select the objects(s). You can use more than one option in the same command.

Examples

See the examples in the Using find Commands section.

find currents

Returns the names of all the current objects matching the specified pattern.

Usage

```
find currents object_name ...
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- ***object_name***

  (Required) Specifies the name of a current object you want to search. Multiple names and wildcards are supported. You can use the extended options detailed in Selecting Waveforms in Tcl to select the objects(s). You can use more than one option in the same command.

Description

The find currents and find signals -i commands are synonymous; both commands return the full pathnames of all analog and digital objects that match the name specification.

Examples

See the examples in the Using find Commands section.

find digitals

Returns a list of digital signals matching the specified pattern.

Usage

```
find digitals item_name ...
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- ***item_name***

  (Required) Specifies the name of an item you want to search. Multiple names and wildcards are supported. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform(s). You can use more than one option in the same command.

Examples

See the examples in the Using find Commands section.

find nets | signals

These commands return the full pathnames of all analog and digital objects that match the name specification.

Usage

```
find nets | signals item_name ...
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- ***item_name***

  (Required) Specifies the name of an object you want to search. Multiple names and wildcards are supported. You can use the extended options detailed in Selecting Waveforms in Tcl to select the signal(s). You can use more than one option in the same command.

Description

The **find nets** and **find signals** commands are synonymous; both commands return the full pathnames of all analog and digital objects that match the name specification. Objects listed include signals, quantities, and terminals unless these have been specifically excluded by using the respective argument.

Examples

See the examples in the Using find Commands section.

getactivecursortime

Returns the location of the active cursor.

Usage

```
getactivecursortime [-window window_name]
```

Parameters

- -window *window_name*

  (Optional) Specifies the window from which to return the location of the active cursor. If this is not specified, the location of the active cursor in the active window is returned.

Examples

```
getactivecursortime -window "Wave:2"
```

Note

The **getactivecursortime** command is equivalent to the **wave cursortime** command with no time or cursor name specified.

precision

Sets the double or cursor display precision.

Usage

```
precision [-double | -cursor] [precision_value#]
```

Parameters

- -double | - cursor

  (Optional)

  - -double - Enables the setting of the number of digits displayed when double-precision numbers are displayed in waveforms.
  - -cursor - Enables the setting of the number of digits displayed when in cursor flags.

- *precision_value#*

  Specifies the number of digits to be set.

  Note

  The # character is accepted for Questa SIM compatibility, but ignored by EZwave.

Examples

```
precision -cursor 8#
```

printenv

Displays the name and value of environment variables.

Usage

printenv [*variable_name*]

Parameters

- *variable_name*

  (Optional) Specifies which environment variable name and value to display. If this is not specified, the command will return information on all environment variables.

Return Values

Returns a name and value pair of the specified environment variable (or all the variables if not specified) in the form of:

variable_name = value

Description

This command displays the name and value of environment variables. If no variable name is specified, it returns all environment variable information.

Examples

```
printenv PRINTER

# This returns PRINTER=myprinter
```

quit

Exits the EZwave viewer.

Usage

```
quit [-force ] [-discard]
```

Parameters

- -force

  (Optional) If specified, the EZwave viewer quits and bypasses the confirmation dialog box. If this option is not specified, the confirmation dialog box displays and requires a manual confirmation.

- -discard

  (Optional) If specified, the EZwave viewer quits and bypasses the unsaved data dialog box (any unsaved data is lost). If this option is not specified, the unsaved data dialog box displays and requires a manual confirmation.

Caution

If the -discard option is specified, all unsaved data is lost.

Return Values

None.

Examples

```
quit

quit -force

quit -force -discard
```

radix

Defines the default bus radix or returns the current default if no radix is specified.

Usage

```
radix [-radix_type | radix_name ]

radix [-onescomplement | -decimal | -magnitude |  -binary |
-octal | -hexadecimal | -unsigned | -ascii | [-fpoint n]
-tograydec | -graytodec | -thermo |
-symbolic | -time | -enumnumeric | -enumsymbolic ] |
[radix_name ]
```

Parameters

- -onescomplement

  (Optional) Specifies the default radix to be one's complement.
- -decimal

  (Optional) Specifies the default radix to be decimal (two's complement).
- -magnitude

  (Optional) Specifies the default radix to be signed magnitude.
- -binary

  (Optional) Specifies the default radix to be binary.
- -octal

  (Optional) Specifies the default radix to be octal.
- -hexadecimal

  (Optional) Specifies the default radix to be hexadecimal.
- -unsigned

  (Optional) Specifies the default radix to be unsigned decimal.
- -ascii

  (Optional) Specifies the default radix to be ascii.
- -fpoint *n*

  (Optional) Specifies *n* fixed-point radix.
- -tograydec

  (Optional) Converts decimal representation radix to gray radix.
- -graytodec

  (Optional) Coverts gray radix to decimal representation radix.

- -thermo

  (Optional) Specifies the default radix to be unary (for Eldo **.sigbus** thermo-coded patterns).
- -symbolic

  Ignored; for compatibility purposes only.
- -time

  Ignored; for compatibility purposes only.
- -enumnumeric

  Ignored; for compatibility purposes only.
- -enumsymbolic

  Ignored; for compatibility purposes only.
- *radix_name*

  Specifies the name of a user-defined radix.

Examples

```
radix -binary
```

radix define

Defines a new radix.

Usage

`radix define` ***`radix_name`*** `{`*`radix_value, radix_label`* `[`*`radix_value2, radix_label2 … radix_valueN, radix_labelN`*`] [-default` *`radix_type`*`] [-color` *`color`*`]}`

Parameters

- ***radix_name***

  (Required) Specifies a name for the radix.

- ***radix_value, radix_label*** *[radix_value2, radix_label2 … radix_valueN, radix_labelN*]

  (Required) Specifies a comma-separated list of number pattern and label pairs for the radix.

  Use the wildcard character **?** to specify a range of pattern values. The first matching pattern from the top of the list is used if there is more than one match.

- -default *radix_type*

  (Optional) Specifies the radix to use if a match is not found for a given value.

- -color *color*

  (Optional) Specifies the display color for each state. Supported color keywords:

  - black
  - white
  - gray
  - lightgray
  - darkgray
  - red
  - green
  - blue
  - yellow
  - orange
  - cyan
  - magenta
  - pink

Return Values

None.

Examples

```
# Defines a new radix named States
radix define States {
6'b01??00, WRITE
6'b10??00, READ
-default hex }

# Defines a color for each state
radix define Bin2Gray_dec {
  X"0"  FOO -color red
  X"1"  BAR -color blue
  X"2"  OOF -color yellow
  X"3"  RAB -color green
  X"4"  ABC -color cyan
  X"5"  EFG -color blue
  X"6"  IJK -color magenta
  X"7"  LMN -color pink
...
```

radix delete

Removes the radix definition from the named radix.

Usage

`radix delete` ***`radix_name`***

Parameters

- ***radix_name***

  (Required) Specifies the name of the radix.

Return Values

None.

Examples

```
# Removes the radix definition from the radix named States
radix delete States
```

radix list

Returns the complete definition of the named radix.

Usage

radix list [*radix_name*]

Parameters

- *radix_name*

  (Optional) Specifies the name of the radix.

Return Values

Returns the complete definition of the named radix or the definition of all currently defined radices if a name is not specified.

Description

This command returns the complete definition of the named radix. If no radix name is specified, it returns a list of all the defined radices.

Examples

```
# Define a new radix named States
radix define States {
6'b01??00, WRITE
6'b10??00, READ
-default hex }

radix list States
# This returns:
6'b01??00, WRITE
6'b10??00, READ
-default hex
```

radix names

Returns a list of all the defined radices.

Usage

`radix names`

Parameters

- None.

Return Values

Returns a list of all the radices if not specified in the form.

Examples

```
radix names
```

radix signal

Sets the radix value for the specified waveforms, or if the radix value is not specified it inspects the radix values.

Usage

radix signal [*waveform_name* [*radix_value*]][-fpoint *decimal*]

Parameters

- *waveform_name*

  (Optional) Specifies the name of the waveform(s) for which the radix is set or inspected.

- *radix_value*

  (Optional) Specifies the value of the radix to be set for the specified waveform.

  Use empty quotation marks (" ") to unset the radix for a specified signal.

- -fpoint*decimal*

  (Optional) Ignored.

Return Values

None.

Examples

```
# Sets the radix value for waveform :top:sigout
radix signal :top:sigout States
```

save

Saves the specified Waveform Calculator measure.

Usage

save("***name***")

Arguments

- "***name***"

  (Required) Specifies the name of the Waveform Calculator variable to save. The double quotes (**" "**), are required for the internal function parameter.

Description

The save is effective at the end of the Tcl evaluation, not when the line is executed.

Examples

```
save("mySndr")
```

Related Topics

wfc

setenv

Sets or displays the value of the specified environment variable.

Usage

setenv *variable_name* [*value*]

Parameters

- ***variable_name***

  (Required) Specifies the environment variable name to be set or displayed.

- *value*

  (Optional) The value to which the environment variable is set. If this is not specified, the command will display the current value.

Return Values

The value of the environment variable.

Description

Sets or displays the value of the specified environment variable. If no value is specified, this command displays specified environment variable's value.

Examples

```
setenv MGC_AMS_HOME /home/smith/2007.2
# Sets the value of the environment variable MGC_AMS_HOME to
# /home/smith/2007.2

setenv MGC_AMS_HOME
# Returns /home/smith/2007.2
```

unsetenv

Unsets an environment variable.

Usage

```
unsetenv variable_name
```

Parameters

- ***variable_name***

  (Required) Specifies the environment variable name to be unset.

Return Values

None.

Note

The settings are not persistent and are available only within an EZwave session.

Examples

```
unsetenv MY_TCL_VAR
```

wave activecursor

Sets the specified cursor as the active cursor.

Usage

wave activecursor [-window *window_name*] [*cursor_name*]

Parameters

- -window *window_name*

  (Optional) Specifies the window in which to set the active cursor. If this is not specified, the active cursor is set in the active window.

- *cursor_name*

  (Optional) Specifies the cursor to set as the active cursor. If this is not specified, the name of the active cursor is returned.

Examples

To make cursor C2 the active cursor in the active window:

```
wave activecursor C2
```

wave activewindow

Returns the name of the currently active window, or sets a specified window to be active.

Usage

wave activewindow [-title *window_name*]

Parameters

- -title *window_name*

  (Optional) Specifies the window to be made active. If -title is not specified, returns the name of the currently active window.

Examples

To return the name of the currently active window:

```
wave activewindow
```

To set a window named Wave1 as the active window:

```
wave activewindow -title Wave1
```

wave activeworkspace

Causes the specified workspace to become active.

Usage

`wave activeworkspace` `[`*`workspace_name`*`]`

Parameters

- *workspace_name*

  (Optional) Specifies the name of the workspace to become active. If *workspace_name* is omitted, this command returns the current active workspace name.

Return Values

None if an argument is specified. If no argument is specified, returns the active workspace name.

Examples

```
wave activeworkspace ADC12
```

wave addannotation

Adds a text annotation to a waveform at the specified location in the active window.

Usage

```
wave addannotation -x x_value [-y y_value] -text text [-snap]
[-window | waveform_name ] [-anchor_style <NONE|POINT|DOT>]
[-line_style <NONE|SOLID>] [-outline] [-vertical_line]
[-show_outline <true|false>] [-show_vline <true|false>]
[-text_font <name|family> <PLAIN|BOLD|ITALIC> <size>]
[-text_color #<rgb_value>]
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- **-x *x_value***

  (Required) Specifies the X value at which the annotation is added. If a Y value is specified, the annotation is added at the location, (***x_value***, *y_value*). Otherwise, the command analyzes the waveform data and places the annotation on the waveform at the specified X value.

- -y *y_value*

  (Optional) Specifies the Y value at which the annotation is added, at the location, (***x_value***, *y_value*). If this option is not specified, the command analyzes the waveform data and places the annotation on the waveform at the specified X value.

- **-text *text***

  (Required) Specifies the annotation text. Quotation marks (" ") are required for text that include spaces. Annotation text may also include multiple lines by using **\n**.

- -snap

  (Optional) Specifies that the text annotation snaps to the nearest waveform datapoint.

- -window | *waveform_name*

  (Optional) If *waveform_name* is specified, the annotation is added to that waveform. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform. You can use more than one option in the same command.

  If the waveform is displayed several times in the active window, the last occurrence of the waveform is annotated. If no waveform name is specified, the last displayed waveform in the active window is annotated.

  If -window is specified, the annotation is added to the active window, and not the waveform.

- -anchor_style <NONE|POINT|DOT>

  (Optional). Selects the symbol to use at the anchor point between the waveform and the text annotation. Available only for annotations that are attached to waveforms.

- -line_style <NONE|SOLID>

  (Optional). Controls the visibility of the line between the anchor point and the text annotation box. Default is SOLID. Available only for annotations that are attached to waveforms.

- -outline

  (Optional) Supported for backward compatibility with legacy Tcl scripts.

- -vertical_line

  (Optional) Supported for backward compatibility with legacy Tcl scripts.

- -show_outline <true|false>

  (Optional). Controls the visibility of the text box on the new text annotation. Default is true.

- -show_vline <true|false>

  (Optional). Controls the visibility of a vertical line at the anchor point between the waveform and the text annotation. Available only for annotations that are attached to waveforms. Default is false.

- -text_font <name|family> <PLAIN|BOLD|ITALIC> <size>

  (Optional). Specifies the font, style and point size to use for the new text annotation.

- -text_color #<rgb_value>

  (Optional). Specifies the color to use for the new text annotation.

  The RGB color code is a hexadecimal value preceded by a '#' representing #*RRGGBB*, where *RR* represents the red value, *GG* represents the green value, and *BB* represents the blue value. For example, #0000ff is blue, #000000 is black, and #ffffff is white.

Return Values

The annotation identifier marker **mk:#** where **#** is a number.

Examples

- This example adds a simple text annotation:

  ```
  wave addannotation -x 149.980n -text simple
  ```

- This example adds a text annotation and snaps it to the nearest waveform datapoint:

  ```
  wave addannotation -x 150n -y 3.3 -text "Annotation with snap" -snap
  ```

- This example adds a 2 line text annotation with custom font and color properties.

  ```
  wave addannotation -x 300n -text "Annotation with custom\n font and
  color properties" -text_font Jomolhari ITALIC 14 -text_color #ff4040
  ```

wave addaxisdeltamarker

Adds a delta marker, between two values, on the selected waveform. The marker can be horizontal or vertical. Optionally text can be added to this marker.

Usage

```
wave addaxisdeltamarker -wf waveform_name {-XDelta | -YDelta} -X1 xdata1
-X2 xdata2 [-text text]
```

Parameters

Note
This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- **-wf *waveform_name***

  (Required) Specifies the name of the waveform.

- {**-XDelta**| **-YDelta**}

  (Required)

  - **-XDelta** - specifies that the marker is to show information about an x delta.
  - **-YDelta** - specifies that the marker is to show information about a y delta.

- **-X1 *xdata1***

  (Required) Specifies the co-ordinate of the first point of the delta marker. The x-coordinate is used if -XDelta is set,. If -YDelta is set, -X1 specifies the y-coordinate of the first point to be used.

- **-X2 *xdata2***

  (Required) Specifies the co-ordinate of the second point of the delta marker. The x-coordinate is used if -XDelta is set. If -YDelta is set, -X2 specifies the y-coordinate of the second point to be used.

- -text *text* }

  (Optional) Specifies the text for the delta marker. Quotation marks (" ") are required for text that include spaces. Text may also include multiple lines by using **\n**

Return Values

None.

Examples

```
wave addaxisdeltamarker  -wf -show calc.V -notop  -separator / calc/
SMeas_clk2 -XDelta  -x1 0.0 -x2 146143.06591866582 -text "Measurement
Window"

wave addaxisdeltamarker  -wf -show calc.V -notop  -separator / calc/
SMeas_clk2 -XDelta  -x1 0.0 -x2 10000

wave addaxisdeltamarker  -wf -show calc.V -notop  -separator / calc/
SMeas_clk2 -YDelta  -x1 -130 -x2 -10 -text "Example Text"

wave addaxisdeltamarker  -wf -show calc.V -notop  -separator / calc/
SMeas_clk2 -YDelta  -x1 -100 -x2 -50 -textms
```

wave addcursor

Creates a new cursor in the specified window.

Usage

```
wave addcursor [-window window_name] [-time time | X_value]
[-horizontal [-row row_index] [-axis axis_name]] [-name cursor_name]
[-activeLineStyle active_line_style] [-activeLineWidth active_line_width]
[-lineStyle line_style] [-lineWidth line_width]
[-activeColor active_cursor_color] [-color cursor_color]
[-flagTextColor flag_text_color] [-legendTextColor legend_text_color]
[-textFont text_font text_style text_size]
```

Parameters

- -window *window_name*

  (Optional) Specifies the window in which to create the cursor. If this is not specified, the cursor is created in the active window.

- -time *time*

  (Optional) Specifies the time value where the new cursor should be created.

- *X_value*

  (Optional) Specifies the value along the x axis where the new cursor should be created.

  Note

  If neither a *time* nor an *X_value* is specified, the cursor is created at the beginning of the visible domain area.

- -horizontal

  (Optional) Creates a horizontal cursor at the specified point on the y axis.

- -row *row_index*

  (Optional. For horizontal cursors only) Specifies which row in the graph window to add a horizontal cursor. *row_index* takes a value of 1 to *n*, where the top row has the index value of 1, the second row, 2, and so on. The default value for *row_index* is *n* (the last row in the graph window).

- -axis *axis_name*

  (Optional. For horizontal cursors only) Specifies the name of the axis to add a horizontal cursor to, when there are multiple axes in a row.

- -name *cursor_name*

  (Optional) Specifies a name for the cursor.

- -activeLineStyle *active_line_style*

  (Optional) Specifies the line style when the cursor is active (1-6).

- -activeLineWidth *active_line_width*

  (Optional) Specifies the line width when the cursor is active (1-5).
- -lineStyle *line_style*

  (Optional) Specifies the line style when the cursor is not active (1-6).
- -lineWidth *line_width*

  (Optional) Specifies the line width when the cursor is not active (1-5).
- -activeColor *active_cursor_color*

  (Optional) Specifies the cursor color when the cursor is active (#colorcode).
- -color *cursor_color*

  (Optional) Specifies the cursor color when the cursor is not active (#colorcode).
- -flagTextColor *flag_text_color*

  (Optional) Specifies the color of the flag text (#colorcode).
- -legendTextColor *legend_text_color*

  (Optional) Specifies the color of the legend text (#colorcode).
- -textFont *text_font text_style text_size*

  (Optional) Specifies the text font, style and size.

Examples

To add a cursor in the active window at an X value of 100 ns:

```
wave addcursor 100n
```

To add a horizontal cursor to the second row in the graph window:

```
wave addcursor -horizontal -row 2
```

To add a cursor in the active window at an X value of 100 ns and specify colors, line style, and font:

```
wave addcursor 100n -activeLineStyle 1 -activeLineWidth 1
-lineStyle 2 -lineWidth 3 -activeColor #66ffbb -color #aaaaff
-flagTextColor #ff8800 -legendTextColor #bbbbbb
-textFont Dialog plain 12
```

wave adddeltamarker

Adds a delta marker waveform(s) at the specified location(s) in the active window.

Usage

wave adddeltamarker {**-xdelta** | **-ydelta**} **-wf1** ***waveform_name1*** **-x1** ***x_value1*** **-y1** ***y_value1*** [-wf2 *waveform_name2*] **-x2** ***x_value2*** **-y2** ***y_value2*** [-text *text*]

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- {**-xdelta**| **-ydelta**}

  (Required)

  - **-xdelta** - specifies that the marker is to show information about an x delta.
  - **-ydelta** - specifies that the marker is to show information about a y delta.

- **-wf1** ***waveform_name1***

  (Required) Specifies the name of the waveform to which the first point of the delta marker is to be attached.

- **-x1** ***x_value1***

  (Required) Specifies the x co-ordinate of the first point of the delta marker.

- **-y1** ***y_value1***

  (Required) Specifies the y co-ordinate of the first point of the delta marker.

- -wf2 *waveform_name2*

  (Optional) Specifies the name of the waveform to which the second point of the delta marker is to be attached. If not specified it will default to the same waveform as **-wf1**

- **-x2** ***x_value2***

  (Required) Specifies the x co-ordinate of the second point of the delta marker.

- **-y2** ***y_value2***

  (Required) Specifies the y co-ordinate of the second point of the delta marker.

- -text *text* }

  (Optional) Specifies the text for the delta marker. Quotation marks (" ") are required for text that include spaces. Text may also include multiple lines by using **\n**

Return Values

None.

Examples

```
wave adddeltamarker  -xdelta  -wf1 v_load -x1 0.03851623740539206 -y1
8.505201152235028E-4 -x2 0.051913040161787515 -y2 4.999999302642095 -text
"dx = 13.40M s\ndy = 5.00 V"
# Adds a delta marker with text to two points on the same waveform

wave adddeltamarker  -xdelta  -wf1 v_load -x1 0.06243479893740646 -y1 -
7.579889746381528E-6 -wf2 v_middle -x2 0.07321739137215058 -y2
4.999993280548767 -text "dx = 10.78M s\ndy = 5.00 V"
# Adds a delta marker with text to two points across waveforms
```

Related Topics

Adding Delta Markers With Pick Points

wave addline

Adds a horizontal or vertical line to a waveform in the active window.

Usage

```
wave addline {-x | -y} value [waveform_name] [-text annotation_text]
[-drag] [-hash {up | down | left | right}]
```

Parameters

- { **-x**| **-y**} ***value***

  (Required) Specifies the location and direction of the line to be added. **-x *value*** specifies a vertical line at the specified value on the x axis. **-y *value*** specifies a horizontal line at the specified value on the y axis. The units correspond to the units used in the waveforms.

- *waveform_name*

  (Optional) If this option is specified, the command adds a line marker to that waveform's row. If that waveform is displayed several times it is added to the last occurrence. If not specified it will default to the last displayed waveform.

  > **Note**
  > This command applies only to waveforms that are currently displayed. Use the add wave command to display the relevant waveforms before calling this command.

- -text *annotation_text*

  (Optional) Specifies the text to add as an annotation to the line. Quotations (" ") are only required if there are spaces in the text.

- -drag

  (Optional) If this option is specified, the line marker can be dragged. By default, the added line cannot be dragged.

- -hash { up | down | left | right }

  (Optional) If this option is specified, a hash pattern is placed in the specified location relative to the line: up or down for horizontal lines, and left or right for vertical lines. This may be useful in marking limits in the graph.

Return Values

The line identifier in the form of **mk:#** where # is a number.

Examples

```
wave addline -x 150n -text limit -hash left
# Adds a vertical marker at time 150n with hash on left of line, with text
# 'limit'

wave addline -y 3.3 -text "high level" -hash up
# Adds a horizontal marker at 3.3 volts with hash on top up line, with text
# 'high level'
```

wave addmarker

Adds a vertical marker to the specified location.

Usage

```
wave addmarker [-window window_name] [-name name] [-time {time | x_value}]
```

Parameters

- -window window_name

  (Optional) Specifies the name of the window to which the marker is added. If this option is not specified, the marker is added to the active window.

- -name *name*

  (Optional) Specifies the name to be given to the marker.

- -time { *time* | *x_value* }

  (Optional) Specifies the location of the marker, designated by a time value or an X value. If this option is not specified, the marker is placed at the left-most point in the window.

Return Values

The marker identifier in the form of **m#** where **#** is a number, starting from 1.

Description

This command adds a vertical marker to the specified location. This corresponds to the Add Marker button in the Event Search Tool Dialog Box.

Examples

```
wave addmarker -time 100n
```

wave addproperty

Adds a property to a waveform.

Note

This command modifies the dataset. You may use the “dataset save” command to save the dataset before exiting. You can then see the property name and value when you mouse over the waveform name or waveform data.

Usage

```
wave addproperty waveform_name -name property_name -value property_value
```

Parameters

Note

This command also has extended options, which are described in “Selecting Waveforms in Tcl” on page 1014.

- ***waveform_name***

  (Required) Specifies the waveform to which the property will be added. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform. You can use more than one option in the same command.

- **-name*property_name***

  (Required) Specifies the name of the property to be added to the waveform.

- **-value*property_value***

  (Required) Specifies the value of the property to be added to the waveform. This can be either a number or a string.

Examples

```
wave addproperty OUT -name PRUN -value 1
```

wave addwindow

Creates a new graphical window in the EZwave session window and makes it active.

Usage

```
wave addwindow [-title window_title] [-divider ratio]
[-width width -height height] [-x xPosition -y yPosition]
```

Parameters

- -title *window_title*

  (Optional) Specifies the name of the window to be created. “-title” may be omitted.

- -divider *ratio*

  (Optional) Specifies the position of the divider inside the graph window. The divider is the vertical line that separates the waveform names and the waveforms area. *ratio* must be between 0 and 1. The default value is 0.91.

- -width *width* -height *height*

  (Optional) Specifies the width and height of the new window.

- -x *xPosition* -y *yPosition*

  (Optional) Specifies the x and y position of the new window.

Return Values

The name of the created window (by default, **Wave:#**, where **#** is a number).

wave addworkspace

Creates a new workspace in the EZwave session window.

Usage

```
wave addworkspace [workspace_name] [-active]
```

Parameters

- *workspace_name*

  (Optional) Specifies the name of the workspace to be created. If *workspace_name* is not specified, a default name, Workspace#, is used, where # is a number.

- -active

  (Optional) Indicates that the workspace created should become the active workspace.

Examples

```
wave addworkspace ADC12
```

wave cdf

Generates and displays a CDF based on the input waveform, or on the saved histogram data in a *.wdb* file generated from an Eldo Monte Carlo simulation.

Usage

```
wave cdf <-show …>
[-start <x_start> -end <x_end>]
[-normalize x|xx|y|yx|yxx]
[-sample] [-gauss] [-pdf] [-confidence]
[-bounds | [-lbound <value> | -ubound <value>]]
[-average] [-mean] [-nominal] [probability #(,#)*] [-sigma #(,#)*]
[-stat_legend]  [-stat_legend_text <text_pattern>] [-none]
[-tail  [left|right]]
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- -start <x_start>

  Sets the start of the range for the input waveform (same as window() function). See Note.

- -end <x_end>

  Sets the end of the range for the input waveform (same as window() function). See Note.

  Note

  If the CDF waveform has an associated source waveform (a "HISTO_SRC" property), then -start and -end apply to the associated source waveform. Otherwise -start and -end apply to the CDF waveform directly.

- -normalize <p>

  Normalizes the CDF. <p> is "y" and/or "x" or "xx".

  "y"—normalize Y. Currently has no effect.

  "x"—shift input values so that mean equals 0.

  "xx"—shift input values so that mean equals 0 and stddev equals 1.

  Valid values: "x", "xx", "y", "yx", "yxx".

- -sample

  Samples the input waveform (keep the same number of points, but equally distributed along the x-range of the input waveform).

- -gauss

  Plots the integral of the associated theoretical CDF, calculated as an integral of the Gaussian distribution for the average and standard deviation.

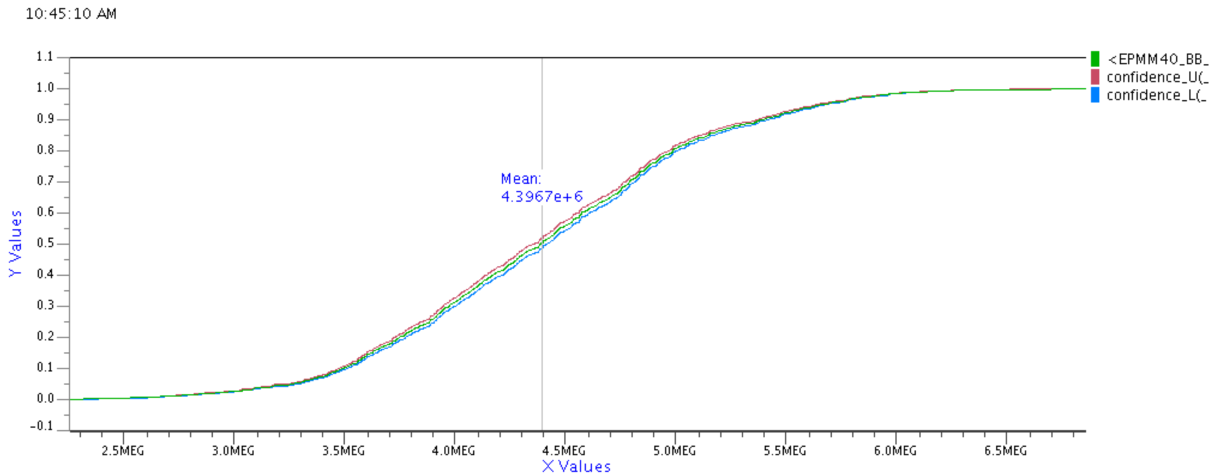
- -pdf

  Plots the integral of the PDF waveform (when available in the waveform database file generated by an Eldo Monte Carlo simulation.

- -confidence

  Plots the upper and lower confidence bounds, when provided by the Eldo simulation, using ".MC dataflow=1" in the netlist. An example is shown in the following figure:

Figure C-4. Example CDF with Upper and Lower Confidence Bounds



- -bounds

  Alias for "-lbound sim -ubound sim".

- -average

  Draws the average (single measure).

- -mean

  Draws the mean (single measure).

- -nominal

  Draws the nominal value (when available in the input file) (single measure).

- -probability <p>

  Draws the probability intervals (twin measures). The plotted probabilities are based on empirical values. <p> is a comma-separated list of probabilities, from 0 to 1, or from 0% to 100%.

  Example of valid value: "0.6827,0.9545,0.98,99%".

- -sigma <p>

  Deprecated. Draws the N-sigma intervals (twin measures). <p> is a comma-separated list of integers, from 1 to 7.

- -stat_legend

  Draws the stat legend. If -stat_legend_text is not provided, EZwave uses the default preferences set in the CDF Legend Options.

- -stat_legend_text

  The text for the stat legend. This is a template (composed of text and @(…) tags). This has no effect until -stat_legend is specified.

- -none

  Does not add the default presentation. When no parameters (except those specifying the waveform) are set for the wave cdf command, then the default settings are used (set from the CDF options on the EZwave Display Preferences Dialog Box).

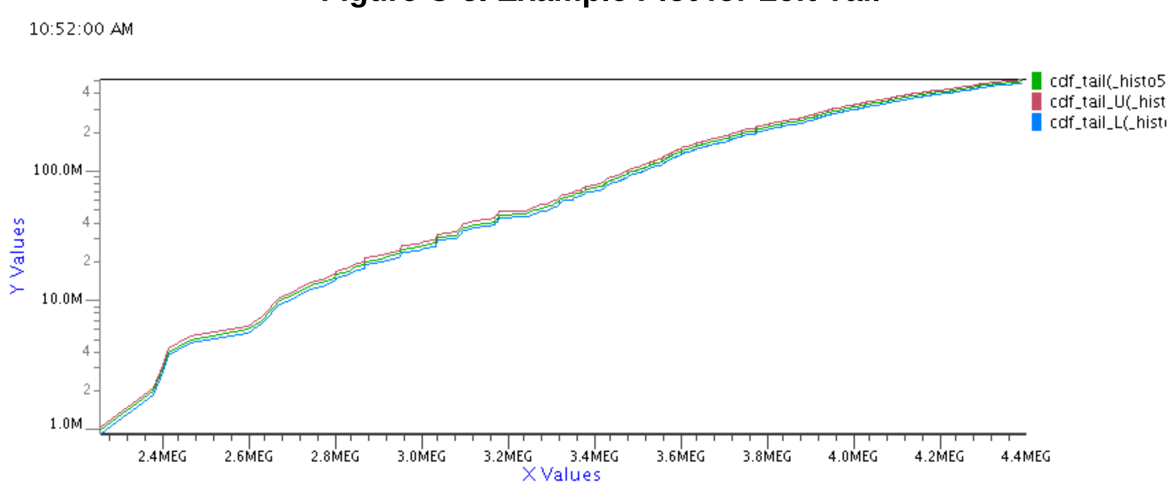
- -tail

  Plots the left or right tail of a transformed CDF with the Y Axis in a log scale. Valid values are “left” and “right”. Gaussian distribution, PDF, all measures, and statistical legends are disabled when the tail is specified. This is not available through the graphical interface. An example is shown in Figure C-5.

Note

The plots of the CDF or its tails are mutually exclusive: you have to use the command three times with different arguments to plot the CDF, the right tail, and the left tail.

Figure C-5. Example Plot for Left Tail



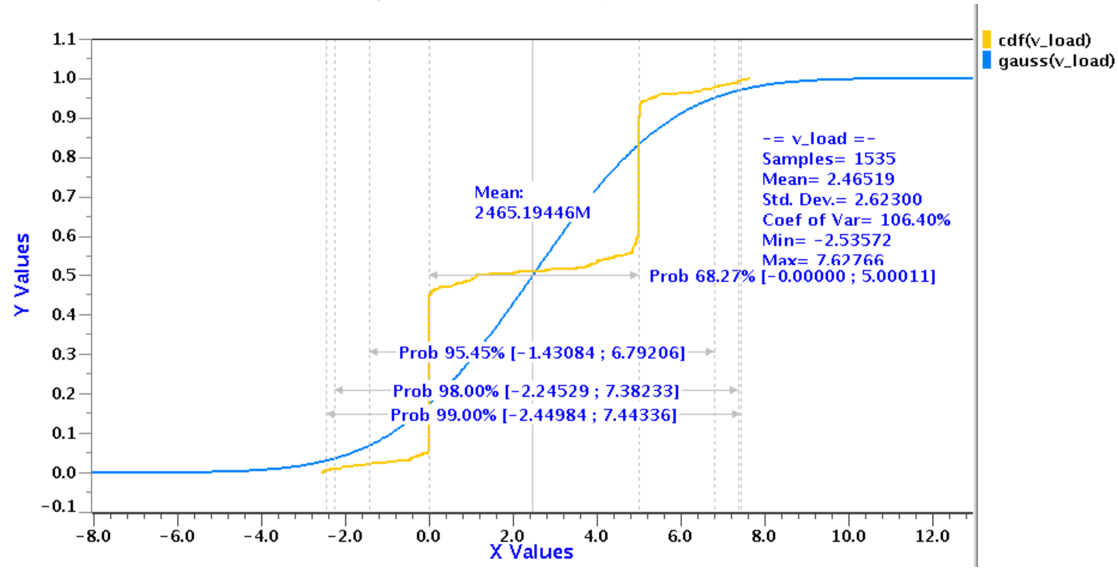
Return Values

None.

Description

Generates and displays a CDF based on the input waveform, or on the saved CDF in a *.wdb* file generated from an Eldo Monte Carlo simulation.

Figure C-6. Example CDF Plot



Tip

See also Cumulative Distribution Functions in the *Eldo User's Manual* and .CDF in the *Eldo Reference Manual.*

Related Topics

Plotting CDFs

wave clean

Takes a name and returns a new name with any special characters replaced by "_". This is useful, for example, when scripting inside EZwave and creating waves through the waveform calculator, where the labels chosen to create the waves might cause the script to work incorrectly.

Usage

wave clean *name*

Parameters

- ***name***

  (Required) Specifies the name to be processed.

Return Values

Returns a new name with any special characters replaced by "_".

Examples

The input name test#inst.hier is returned as test_inst_hier.

wave closewindow

Closes a specified window, all windows, or all windows in the workspace.

Usage

`wave closewindow` `[`*`window_name`*`] | [-all | -workspace]`

Parameters

- *window_name*

  (Optional) Specifies the name of the graph window to close.

- -all | -workspace

  (Optional) Specifies whether to close all currently open windows or to close all windows currently open in the workspace.

Description

Closes a specified window, all windows, or all windows in the workspace. If no argument is specified, this command closes the last window opened.

Examples

```
wave closewindow -workspace
```

wave closeworkspace

Closes a specified workspace, or all workspaces.

Usage

```
wave closeworkspace workspace_name | -all
```

Parameters

- ***workspace_name*** | ***-all***

  Specifies the name of the workspace to close. Alternatively, specify -all to close all workspaces (except the first one).

Examples

```
wave closeworkspace -all
```

wave colortheme

Sets the graphical display or print color scheme.

Usage

```
wave colortheme [black | white | mono | doc] [-print]
```

Parameters

- black | white | mono | doc

  (Optional) Defines the color theme to be set. If no argument is set it defaults to the current color theme.

- -print

  (Optional) If this option is specified the color theme is set for printing only, and does not alter the graphical display.

Examples

```
wave colortheme doc -print
write wave filename.ps
```

wave compressvri

Computes the N^{th} compression point for the specified harmonic.

Usage

```
wave compressvri
-srcwf<source_compound_waveform>
-harmonic <harmonic_index>
[-sweep_param <sweep_param_name>]
[-nDb <ndb_value>]
[-epin <epin_value>]
[-rload <rload_value>]
[-meas_type <returned_result>]
```

Arguments

- ***srcwf***

  (Required) Specifies the input compound waveform name.

- ***harmonic***

  (Required) The harmonic index to be used.

- *sweep_param*

  (Optional) Specifies the sweep parameter used in the input power sweep. Typically, this is in dBm. Default is “Automatic”.

- *nDb*

  (Optional) Specifies the delta between the waveform and the ideal slope at the compression point. Default is 1.

- *epin*

  (Optional) Specifies the input extrapolation point. Default is “Automatic”, the first point in the input power sweep.

- *rload*

  (Optional) Specifies the internal resistance of the source. If specified, this is used to convert the voltage to dBm. Default is 50.

- *meas_type*

  (Optional) Specifies the choice of returned result, “Input”, “Output” or “Both”. Default is “Both”.

Return Values

Double. The N^{th} compression point for the specified harmonic.

Examples

Example Tcl command usage:

```
dataset open /test/psf_data/psf_spe
wave compressvri -srcwf "psf_spe/hb_fi/outdiff" -harmonic 1
```

Related Topics

compressvri

compresscompound

wave createbus

Creates a bus containing the specified waveforms, inside the *calc* database.

Usage

```
wave createbus <bus_name> <wf_name_1> ... <wf_name_n>
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- <bus_name>

  (Required) Specifies a name for the bus.

- <wf_name_1> ... <wf_name_n>

  (Required) Specifies the waveforms to add to the bus.

  You can use the set of options described in "Selecting Waveforms in Tcl" on page 1014 to select waveforms to add to the bus.

Examples

This example adds the waveforms **vip_1** to **vip_8** to a bus called **mybus**:

```
wave createbus mybus vip_1 vip_2 vip_3 vip_4 vip_5 vip_6 vip_7 vip_8
```

wave cursortime

Moves a cursor to the specified position.

Usage

```
wave cursortime [-window window_name] [-time time | X_value ]
[cursor_name]
```

Parameters

- -window *window_name*

  (Optional) Specifies the window in which to move a cursor. If this is not specified, a cursor in the active window is moved.

- -time *time*

  (Optional) Specifies the time value to which the cursor should be moved.

- *X_value*

  Specifies the value along the x axis to which the cursor should be moved.

  Note

  If neither a *time* nor an *X_value* is specified, the cursor's location is returned.

- *cursor_name*

  (Optional) Specifies the cursor to be moved. If *cursor_name* is not specified, the command applies to the active cursor in the specified window.

Examples

To move cursor C2 in the active window to an X value of 100 ns:

```
wave cursortime 100n C2
```

wave deletecursor

Deletes the specified cursor, or all cursors, in a window.

Usage

```
wave deletecursor [-window window_name] [cursor_name | cursor_id] [-all]
```

Parameters

- -window *window_name*

  (Optional) Specifies the window from which to delete the cursor. If this is not specified, a cursor in the active window is deleted.

- *cursor_name* | cursor_id

  (Optional) Specifies the cursor to be deleted. *cursor_name* is the full name of the cursor, for example, C2. *cursor_id* is the cursor's number, for example, 2. If no *cursor_name* or *cursor_id* is specified, the active cursor is deleted.

- -all

  (Optional) Removes all cursors from the window.

Examples

To delete cursor C2 in the active window:

```
wave deletecursor C2
```

wave difference

Generates and displays a waveform corresponding to the difference between the first and second waveforms specified.

Usage

```
wave difference waveform_name1 waveform_name2 [-name out_waveform_name]
[-spice]
```

Parameters

- ***waveform_name1***

  (Required) Specifies the first input waveform. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform. You can use more than one option in the same command.

- ***waveform_name2***

  (Required) Specifies the second input waveform. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform. You can use more than one option in the same command.

- -name *out_waveform_name*

  (Optional) Specifies the name of the output waveform.

- -spice

  (Optional) Specifies that compatibility with Spice is to be ensured for the naming of the output waveform, generating W(<*difference*>).

Return Values

None.

Error Messages:

Table C-7. wave difference Error Messages

Error Message	Description
No object matching	At least one of the input waveforms is not found.
Difference requires 2 objects	Only one input waveform has been specified. Two are required.
Too many objects for difference	At least one of the regular expressions used to describe the input waveform names has identified more than one waveform.

Examples

```
wave difference Q1 Q2
```

wave displayed

Retrieves a list of the displayed waveforms.

Usage

```
wave displayed [-window window_name | -allwindows | -activewindow]
[-color]
```

Parameters

- -window *window_name*

  (Optional) Specifies a window to apply the command to. Only waveforms in the specified window are listed.

- -allwindows

  (Optional) Specifies that displayed waveforms in all windows should be listed.

- -activewindow

  (Optional) Specifies that only the displayed waveform in the currently active window should be listed.

- -color

  (Optional) If this option is specified, RGB color information is included with each displayed waveform listed.

Return Values

A Tcl list containing the displayed waveform names and color (if specified).

Examples

```
wave displayed
# returns:
# <test/tran>v(in)<test/tran>v(out) <test/tran>v(vdd)

wave displayed -color
# returns:
# {<test/tran>v(in) 0,255,0} {<test/tran>v(out) 255,255,0}
# {<test/tran>v(vdd) 0,128,255}
```

wave exists

Indicates whether the specified waveform exists.

Usage

```
wave exists waveform_name
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- ***waveform_name***

  (Required) Specifies name of the waveform of interest. You can specify multiple waveforms and use wildcards. You can also use the extended options detailed in Selecting Waveforms in Tcl to select the waveform(s). You can use more than one option in the same command.

  Note

  This command will apply to the last loaded dataset if the optional dataset name is not specified within the ***waveform_name***. See "Specifying Waveforms in Tcl" on page 1012.

Return Values

True or False indicating whether or not at least one waveform corresponding to the search string is found.

Description

Indicates whether the specified waveform exists. It is used primarily for preventing errors by enabling scripts to suggest an alternative action if a waveform does not exist in the dataset.

Examples

```
## Check if a waveform exists in the database
set wave1 ":adc12test_mixed_eldo_ms:yadc12:x1:clk"
set wave2 ":a:waveform:that:does:not:exist"

if {[wave exists $wave1]} {
   puts "waveform '$wave1' exists"
} else {
   puts "waveform '$wave1' does not exist"
}

if {[wave exists $wave2]} {
   puts "waveform '$wave2' exists"
} else {
   puts "waveform '$wave2' does not exist"
}
```

wave ezwave_title

Adds a custom title to the top of the EZwave main display.

Usage

wave ezwave_title *new-title*

Parameters

- new-title

  (Required) Specifies the new title to be displayed.

Return Values

If the parameter ***new-title*** is set to “My Custom Title”, the displayed name will be “My Custom Title (EZwave #Version#)”.

Examples

```
wave ezwave_title "My Custom Title"
```

wave gettype

Returns the type of the specified waveform.

Usage

```
wave gettype waveform_name
```

Parameters

- ***waveform_name***

  (Required) Specifies the name of the waveform of interest. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform. You can use more than one option in the same command.

Return Values

Returns one of the following keywords:

- analog
- voltage
- current
- digital
- real
- complex
- analog_bus
- digital_bus
- compound
- unknown

Note

If the waveform is a compound waveform, two keywords are returned, for example:

```
complex compound
```

wave histogram

Generates and displays a histogram based on the input waveform, or on the saved histogram data generated from an Eldo Monte Carlo simulation.

Usage

```
wave histogram <-show …> [-bins scott|sqrt|sturge|#] [-binscentered]
[-histosim] [-normalize <norm>] [-sample] [-pdf] [-cdf] [-gauss]
[-average] [-mean] [-nominal] [probability #(,#)*] [sigma #(,#)*]]
[-bounds | [-lbound|-ubound rule_value]] [-stat_legend]
[-stat_legend_text <text>] [-start x start] [-end x end]] [-median]
[-sigbin <gather_sigma>] [-empirical]
```

Parameters

Note

This command also has extended options, which are described in "Selecting Waveforms in Tcl" on page 1014.

- -average

  Draws the average (single measure).

- -bins

  Defines the number of bins. Allowed values:

  scott—apply Scott's rule (see the Eldo User's Manual for details).

  sturge—apply Sturge's rule (see the Eldo User's Manual for details).

  sqrt—the number of bins in the square root of the size of the input waveform.

  Integer value.

- -binscentered

  Extreme bins are centered on the ymin/ymax values of the input waveform.

- -bounds

  Alias for "-lbound sim -ubound sim".

- -cdf

  Plots the CDF (cumulative density function) of the input waveform.

- -gauss

  Plots the Gaussian distribution.

- -histosim

  Uses the histogram stored in the waveform database file (when available in the file), instead of computing a new one.

- -lbound <p>

  Specifies the lower bound position. <p> is composed of a rule and an optional figure, separated by an underscore. Valid rules: “sim”, “simOrUser”, “user”, “no”

  Examples of valid parameters: “sim”, “simoruser_-3”, “user_3.14”, “no”.

  See the description of Lower Bound/Upper Bound in the Histogram Measures Options topic for more information.

- -mean

  Draws the mean (single measure).

- -median

  Draws the median value.

- -nominal

  Draws the nominal value (when available in the input file) (single measure).

- -normalize <p>

  Normalizes the histogram. <p> is “y” and/or “x” or “xx”.

    “y”—normalize Y.

    “x”—shift input values so that mean equals 0.

    “xx”—shift input values so that mean equals 0 and stdev equals 1.

  Valid values: “x”, “xx”, “y”, “yx”, “yxx”.

- -pdf

  Plots the PDF waveform (when available in the waveform database file generated by an Eldo Monte Carlo simulation.stdev

- -sample

  Samples the input waveform (keep the same number of points, but equally distributed along the x-range of the input waveform).

- -sigbin <gather_sigma>

  Truncates the histogram to a certain number of sigmas. Specify 0 for no gathering. Specify a value above 0 to gather all the samples above mean + gather_sigma × sigma in the Above bin, and all the samples below “mean - gather_sigma × sigma in the Below bin. The default is 0.

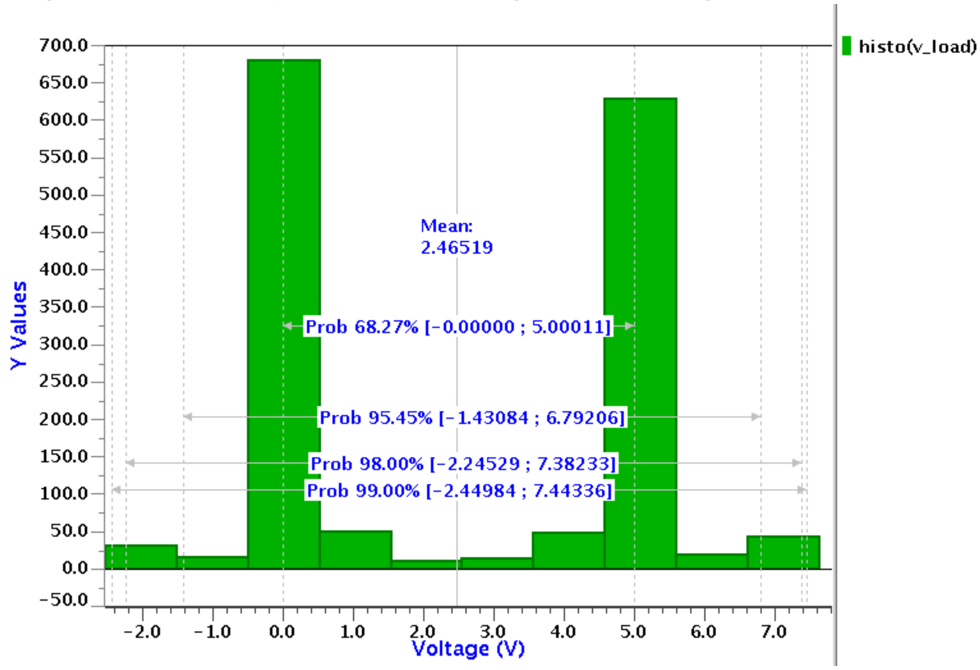
- -probability <p>

  Draws the probability intervals (twin measures). The plotted probabilities are based on empirical values. <p> is a comma-separated list of probabilities, from 0 to 1, or from 0% to 100%.

  Example of valid value: “0.6827,0.9545,0.98,99%”.

  The following figure shows an example histogram plot with probability annotations.

Figure C-7. Example Plot showing wave histogram Probabilities



- -sigma <p>

  Deprecated. Draws the N-sigma intervals (twin measures). The plotted nSigmas are based on the mean and standard deviation of the theoretical Gaussian distribution. <p> is a comma-separated list of integers, from 1 to 7.

- -empirical

  Specifies that the N-Sigma plot is based on the CDF. When not specified, the N-Sigma plot is based on the mean and standard deviation of the theoretical Gaussian distribution.

- -start <x_start>

  Sets the start of the range for the input waveform (same as window() function).

- -end <x_end>

  Sets the end of the range for the input waveform (same as window() function).

- -stat_legend

  Draws the stat legend. If -stat_legend_text is not provided, EZwave uses the default preferences set in the Histogram Legend Options.

- -stat_legend_text

  The text for the stat legend. This is a template (composed of text and @(…) tags). This has no effect until -stat_legend is specified.

- -ubound <p>

  Upper bound position. See -lbound for <p> valid values.

See the description of Lower Bound/Upper Bound in the Histogram Measures Options topic for more information.

- -window

  Specifies the destination window.

- -windowTitle

  Specifies the destination window.

Return Values

None.

Description

Generates and displays a histogram based on the input waveform, or on the saved histogram data in a *.wdb* file generated from an Eldo Monte Carlo simulation.

Note

See also Histograms in the *Eldo User's Manual* and .MC in the *Eldo Reference Manual.*

Related Topics

Plotting Histograms

wave ipnvri

Computes the N^{th} order input-referred intercept point, output-referred intercept point, or both for the specified input compound waveform.

Usage

```
wave ipnvri
-srcwf<source_compound_waveform>
-refharm <reference_harmonic_indexes>
-spurharm <spurious harmonic indexes>
[-sweep_param <sweep_param_name>]
[-norder <norder_value>]
[-epin <epin_value>]
[-rload <rload_value>]
[-meas_type <returned_result>]
```

Arguments

- ***srcwf***

  (Required) Specifies the input compound waveform name.

- ***refharm***

  (Required) An array of integers. Specifies the reference harmonic indexes associated with each tone. The order and the number of indexes must match the database data.

- ***spurharm***

  (Required) An array of integers. Specifies the spurious harmonic indexes associated with each tone. The order and the number of indexes must match the database data.

- *sweep_param*

  (Optional) Specifies the sweep parameter used in the input power sweep. Typically, this is in dBm. Default is “Automatic”.

- *norder*

  (Optional) Specifies the N order intercept. Default is 3.

- *epin*

  (Optional) Specifies the input extrapolation point. Default is “Automatic”, the first point in the input power sweep.

- *rload*

  (Optional) Specifies the internal resistance of the source. If specified, this is used to convert the voltage to dBm. Default is 50.

- *meas_type*

  (Optional) Specifies the choice of returned result, “IIP”, “OIP” or “Both”. Default is “Both”.

Return Values

Float or Tuple (Float, Float). The input-referred intercept point, the output-referred intercept point, or both.

Examples

Example Tcl command usage:

```
dataset open /test/psf_data/psf_spe
wave ipnvri -srcwf "psf_spe/hb_fi/outdiff" -refharm 1 0 -1 -spurharm 1 1 -
2
```

Example User-defined Function:

```
proc IPnVRI { wf refIndexes spurIndexes {sweepParam "Automatic"} {nOrder
3} {ePin "Automatic"} {rload 50} {measType "Both"} } {
    ### TITLE = ipn( ) - N order intersection point
    ### HELP = The ipn() function gives IIP, OIP or both
    ### wf.HELP = Signal Compound waveform
    ### wf.TYPE = waveform
    ### refIndexes.HELP = Array of integers: the reference harmonic
indexes associated to each tone
    ### refIndexes.TYPE = array
    ### spurIndexes.HELP = Array of integers: the spurious harmonic
indexes associated to each tone
    ### spurIndexes.TYPE = array
    ### sweepParam.HELP = The sweep parameter used.
    ### sweepParam.TYPE = string
    ### nOrder.HELP = N order of the spurious frequency. Default is 3.
    ### nOrder.TYPE = integer
    ### ePin.HELP = Extrapolation point. Default is the first point in the
input power sweep.
    ### ePin.TYPE = double
    ### rload.HELP = 50 by default.
    ### rload.TYPE = double
    ### measType.HELP = Choice of the returned result.
    ### measType.TYPE = string

    # here the source waveform is in WFC format: wf(...)
    # we need to parse it (remove "wf()") in order to
    # use it further in the TCL command "wave ipnvri"
    set wf [string trim $wf " "]
    set match [string first "wf(" $wf 0]
    if { $match == 0 } {
        set wf [string replace $wf 0 2 ""]
        set wf [string replace $wf end end ""]
    }

    set value [wave ipnvri -srcwf "$wf" -refharm "$refIndexes" -spurharm
"$spurIndexes" -sweep_param "$sweepParam" -norder $nOrder -epin $ePin -
rload $rload -meas_type "$measType"]

    return $value
}
```

Related Topics

IPnVRI

ipn

wave jitter

Performs the calculation and plot of the specified jitter and its measurements.

Usage

wave jitter

{-Name {ABSOLUTE | ABSOLUTE_INTEG | PERIOD | PERIOD_INTEG | H_PERIOD | FREQUENCY | C2C | C2C_INTEG | LT | LT_INTEG | TIE | CONFIDENCE_INTERVAL}

{-SRC {src_wf_access_options}}

[-REF {ref_wf_access_options | ref_value}]

[-OUT_Name {output_wf_name}]

[-BEGin {t_start_value}]

[-END {t_stop_value}]

[-F0 {f0}]

[-Sampling_Points_NB {sampling_points_number}]

[-RMS_Jitter {rms_jitter_value}]

[-Gaussian_Samples_NB {<seq_start> <seq_end> <seq_step>}]

[-Confidence_Level {confidence_level_value}]

[-Edge {RISING | FALLING | EITHER}]

[-NB_Cycles {nb_cycles_value}]

[-DUTY_Cycle {duty_cycle_value}]

[-Single_Threshold {TRUE | FALSE}]

[-Single_Threshold_Ref {TRUE | FALSE}]

[-Topline {topline_value}]

[-Baseline {baseline_value}]

[-Threshold {threshold_value}]

[-Topline_Ref {topline_ref_value}]

[-Baseline_Ref {baseline_ref_value}]

[-Threshold_ref {threshold_ref_value}]

[-NEW_Window]

[-WINDOW {window_name}]

[-ROW {rowId}]

[-SCale {LINEAR | LOG10 | LOG2}]

[-MIN] [-MAX] [-MEAN] [-STDdev] [-PEAKtopeak] [-HISTOgram]

Parameters

- -Name {ABSOLUTE | ABSOLUTE_INTEG | PERIOD | PERIOD_INTEG | H_PERIOD | FREQUENCY | C2C | C2C_INTEG | LT | LT_INTEG | TIE | CONFIDENCE_INTERVAL}

  (Required) Specifies the short name of the required jitter type for calculation:

  ABSOLUTE—Absolute Jitter (Time-Domain)

  ABSOLUTE_INTEG—Absolute Jitter (Phase-Noise)

  PERIOD—Period Jitter (Time-Domain)

  PERIOD_INTEG—Period Jitter (Phase-Noise)

  H_PERIOD—Half-Period Jitter (Time Domain)

  FREQUENCY—Frequency Jitter (Time Domain)

  C2C—Cycle-to-Cycle Jitter (Time-Domain)

  C2C_INTEG—Cycle-to-Cycle Jitter (Phase-Noise)

  LT—Long-Term Jitter (Time-Domain)

  LT_INTEG—Long-Term Jitter (Phase-Noise)

  TIE—Time Interval Error (TIE)

  CONFIDENCE_INTERVAL—Jitter Confidence Interval

- -SRC {src_wf_access_options}

  (Required) Specifies the source waveform(s) for which jitter is calculated.

  src_wf_access_options

  Source waveform access options (see add wave command).

- -REF {ref_wf_access_options | ref_value}

  (Required for TIE jitter) Specifies a waveform to use as a reference waveform, with respect to which jitter is calculated.

  ref_wf_access_options

  Reference waveform access options (see add wave command).

  ref_value

  Reference period or frequency value.

- -OUT_Name {output_wf_name}

  (Optional) Jitter output waveform name. Default value: *<jitter_type>_jitter*.

- -BEGin {t_start_value}

  (Optional) Specifies a start time for jitter calculation. If not used, the jitter calculation starts at the beginning of the source waveform.

- -END {t_stop_value}

  (Optional) Specifies a stop time for jitter calculation. If not used, the jitter calculation stops at the end of the source waveform.

- -F0 {f0}

  (Required for ABSOLUTE, PERIOD, H_PERIOD, FREQUENCY, and LT jitters, not used for others) Specifies the fundamental or reference frequency.

- -Sampling_Points_NB {sampling_points_number}

  (Optional) Defines if sampling will be applied to the source waveform during jitter calculation and specifies the number of sampling points. Default is Automatic, where sampling is applied with 100 points.

- -RMS_Jitter {rms_jitter_value}

  (Required for CONFIDENCE_INTERVAL, not used for others) Specifies the RMS jitter value, as calculated using the standard deviation.

- -Gaussian_Samples_NB {<seq_start> <seq_end> <seq_step>}

  (Required for CONFIDENCE_INTERVAL, not used for others) Specifies the independent Gaussian samples number sequence Start, End, and Step values.

- -Confidence_Level {confidence_level_value}

  (Required for CONFIDENCE_INTERVAL, not used for others) Specifies the jitter confidence level. Default is 99%.

- -Edge {RISING | FALLING | EITHER}

  (Optional) Edge trigger applied to the waveforms, which participate in the jitter calculation. The default is EITHER, where either the Rising or Falling Edge is used, depending on which comes first in the specified time window.

- -NB_Cycles {nb_cycles_value}

  (Optional) Number of waveform oscillation cycles which participate in the jitter calculation. Used only for PERIOD, FREQUENCY, C2C, and LT jitters. Ignored for all others. If not specified, the default number of cycles is used for PERIOD, FREQUENCY, and C2C jitters and the automatically calculated value is used for LT jitter.

- -DUTY_Cycle {duty_cycle_value}

  (Optional) Specifies the duty cycle applied to waveforms in H_PERIOD jitter calculations, to calculate the amount of time the signal is active per period. Ignored for other jitters. If not specified, the duty cycle value is calculated automatically for each waveform. duty_cycle_value can be given as a percentage, using the percent character (for example, 50%).

- -Single_Threshold {TRUE | FALSE}

  (Optional) If TRUE, the *single threshold* will be used for the source waveform period calculation (*topline*/*baseline* values are ignored). If FALSE, *topline*/*baseline* will be used. Default is FALSE.

- -Single_Threshold_Ref {TRUE | FALSE}

  (Optional) Only used for TIE jitter. If TRUE, the single *threshold_ref* will be used for the reference waveform period calculation (*topline_ref*/*baseline_ref* values are ignored). If False, *topline_ref*/*baseline_ref* will be used. Default is FALSE.
- -Topline {topline_value}

  (Optional) Specifies the y value that sets the high threshold of the source signal. Default is Automatic.
- -Baseline {baseline_value}

  (Optional) Specifies the y value that sets the low threshold of the source signal. Default is Automatic.
- -Threshold {threshold_value}

  (Optional) Specifies the y value that sets a single threshold for the source signal. Default is Automatic.
- -Topline_Ref {topline_ref_value}

  (Optional) Only used for TIE jitter. Specifies the y value that sets the high threshold of the reference signal. Default is Automatic.
- -Baseline_Ref {baseline_ref_value}

  (Optional) Only used for TIE jitter. Specifies the y value that sets the low threshold of the reference signal. Default is Automatic.
- -Threshold_ref {threshold_ref_value}

  (Optional) Only used for TIE Jitter. Specifies the y value that sets a single threshold for the reference signal. Default is Automatic.
- -NEW_Window

  (Optional) Specifies that the jitter result waveform is plotted in a new window. Absent by default.
- -WINDOW {window_name}

  (Optional) Specifies a name of the graph window in which to plot the resulting jitter waveform. window_name is required. Absent by default.
- -ROW {rowId}

  (Optional) Specifies the row in which to plot the resulting jitter waveform. The rowId of the top row is 1, increasing by 1 each row moving downwards. rowID is required. Absent by default.

Note

rowId identifiers are dynamic, with the top row always taking the value of 1.

- -SCale {LINEAR | LOG10 | LOG2}

  (Optional) Specifies the X and Y axes scales (one scale for both axes) for the graph window where the jitter will be plotted. Default value is LOG10 for LT jitter (if the LT jitter resulting waveform is compatible with log scale), LINEAR for other jitter types.

- -MIN

  (Optional) Applies min measurement to the jitter result waveform. The appropriate marker will be plotted over the waveform. Absent by default.

- -MAX

  (Optional) Applies max measurement to the jitter result waveform. The appropriate marker will be plotted over the waveform. Absent by default.

- -MEAN

  (Optional) Applies mean measurement to the jitter result waveform. The appropriate marker will be plotted over the waveform. Absent by default.

- -STDdev

  (Optional) Applies stddev measurement to the jitter result waveform. The appropriate marker will be plotted over the waveform. Absent by default.

- -PEAKtopeak

  (Optional) Applies peak-to-peak measurement to the jitter result waveform. The appropriate marker will be plotted over the waveform. Absent by default.

- -HISTOgram

  (Optional) Apply histogram measurement to the jitter result waveform. The new “Monte Carlo” histogram will be applied. Absent by default.

Examples

This example calculates and plots the period jitter for the specified source and reference waveforms and time interval between 6e-6 sec and 7e-6 sec. The jitter result waveform name is “period_jitter_res”. MIN, MAX and MEAN measurements are applied to the resulting jitter waveform.

```
wave jitter -name period -src <pll_by5_noi/NOISETRAN>V(VCOCLK) -ref 1e-12
-out_name period_jitter_res -begin 6e-6 -end 7e-6 -edge either -min -max
-mean
```

wave launchfolder

Identifies the path to the directory that EZwave was launched from.

Usage

`wave launchfolder`

Parameters

- None.

Return Values

The path to the directory that EZwave was launched from.

Examples

```
wave launchfolder
```

wave listworkspace

Returns a list of workspaces.

Usage

`wave listworkspace`

Parameters

- None

Examples

```
wave listworkspace
```

wave loadbindings

Loads an EZwave keyboard shortcuts schema *.*ezkey* file.

Usage

wave loadbindings *file.ezkey*

Parameters

- file.ezkey

  (Required) Specifies the EZwave keyboard shortcuts schema *.*ezkey* file to be loaded.

Examples

```
wave loadbindings file.ezkey
```

Related Topics

Configuring Keyboard Shortcuts

wave lockcursor

Specifies whether the distance between cursors are fixed or whether they can move independently of each other.

Usage

```
wave lockcursor [ON | OFF]
```

Parameters

- ON

  (Optional) Specifies that the distance between cursors are fixed. Dragging one cursor will move the other cursors, keeping the distance between them the same.

- OFF

  (Optional) Specifies that cursors are able to move independently of each other. Dragging one cursor does not affect the other cursors.

Description

This command specifies whether the distance between cursors are fixed or whether they can move independently of each other. Omit the argument to display the current cursor lock setting.

Examples

```
wave lockcursor ON
```

```
wave lockcursor OFF
```

wave names

Determines whether waveform names within the graphical interface show the dataset names and (or) the complete hierarchical paths.

Usage

wave names [-showdataset {on | off}] [-showhierarchy {on | off}]

Parameters

- -showdataset {on | off}

  (Optional) Determines whether the dataset names are shown.

- -showhierarchy {on | off}

  (Optional) Determines whether the complete hierarchical paths.

Examples

- Show the dataset names within the waveform names.

  ```
  wave names -showdataset on
  ```

- Show the complete hierarchical paths within the waveform names and not the dataset names.

  ```
  wave names -showdataset off -showhierarchy on
  ```

wave plotcompress

Plots output power against input power in dBm and returns n-dB compression point. Only supports compound waveforms.

Usage

```
wave plotcompress
-srcp {srcp_wf_access_options}
[-srcn {srcn_wf_access_options} | 0.0}]
[-sweep_param<sweep_param_name>]
[-fund <fund_value>]
[-ndb <ndb_value>]
[-epin <epin_value>]
[-rload <rload_value>]
[-out_name <output_wf_name>]
[-meas_type {"X"|"Y"|"Both"}]
```

Arguments

- ***srcp***

  (Required) Specifies a compound waveform with PAC scalar signals within an input power sweep.

- *srcn*

  (Optional) Specifies a second compound waveform with PAC scalar signals within an input power sweep.

- *sweep_param*

  (Optional) Specifies the sweep parameter. If not specified, the first sweep parameter found is used.

- *fund*

  (Optional) Specifies the fundamental or harmonic frequency (default nth(indep(src_p), 2)).

- *ndb*

  (Optional) Specifies the number of dB by which the actual curve falls below an ideal linear curve. Use ndb = 1 to extract the 1-db compression point (1 by default).

- *epin*

  (Optional) Specifies the extrapolation point. Default is the first point in the input power sweep.

- *rload*

  (Optional) Specifies the load resistance (50 by default).

- *out_name*

  (Optional) Specifies the output waveform name.

- *meas_type*

  (Optional) Specifies the measurement type to be calculated and plotted. “X” corresponds to the input referred compression. “Y” corresponds to the output referred compression. Default is “Both”.

Return Values

Returns the *ndb* compression point.

Returns a waveform of output power against input power (in dBm).

Related Topics

plotcompress

wave plotpacipn

wave plotpacipn

Takes a PAC spectrum signal pair at the desired and unwanted harmonics, within an input power sweep, plots the pair of IPn curves and returns the intercept point. Only supports compound waveforms.

Usage

```
wave plotpacipn
-srcp {srcp_wf_access_options}
-desired_harm_p<desired_harm_p_value>
-unwanted_harm_p <unwanted_harm_p_value>
[-srcn {srcn_wf_access_options}]
[-desired_harm_n <desired_harm_n_value>]
[-unwanted_harm_n <unwanted_harm_n_value>]
[-sweep_param <sweep_param_name>]
[-norder <norder_value>]
[-rload <rload_value>]
[-out_name <output_wf_name>]
```

Arguments

- ***srcp***

  (Required) Specifies a compound waveform with PAC scalar signals within an input power sweep.

- ***desired_harm_p***

  (Required) Specifies the desired harmonic.

- ***unwanted_harm_p***

  (Required) Specifies the unwanted harmonic.

- *srcn*

  (Optional) Specifies a second compound waveform with PAC scalar signals within an input power sweep.

- *desired_harm_n*

  (Optional) Specifies the desired harmonic, or 0.0 (default).

- *unwanted_harm_n*

  (Optional) Specifies the unwanted harmonic, or 0.0 (default).

- *sweep_param*

  (Optional) Specifies the sweep parameter. If not specified, the first sweep parameter found is used.

- *norder*

  (Optional) Specifies the order of the intercept point (3 by default).

- *rload*

  (Optional) Specifies the load resistance (50 by default).

- *out_name*

  (Optional) Specifies the output waveform name.

Return Values

Returns the *norder* intercept point.

Returns a waveform showing the pair of IPn curves.

Related Topics

plotpacipn

wave plotcompress

wave refresh

Redraws the contents of the specified window.

Usage

```
wave refresh [-window window_name]
```

Parameters

- -window *window_name*

  (Optional) Specifies the window in which to redraw waveforms. If this is not specified, waveforms in the active window are redrawn.

Examples

```
wave refresh
```

wave rowfit

Optimizes the row size in the specified window.

Usage

```
wave rowfit [-window window_name]
```

Parameters

- -window *window_name*

  (Optional) Specifies the window in which to redraw waveforms. If this is not specified, rows are resized in the active window.

Examples

```
wave rowfit
```

wave runindexlist

Displays a list of run indices for the elements of the compound waveform specified. It is useful for iterating through each element individually.

Usage

```
wave runindexlist waveform_name
```

Parameters

- ***waveform_name***

  (Required) Specifies the name of the compound waveform for which the run index list is to be retrieved. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform. You can use more than one option in the same command.

Return Values

List of lists. Returns run indices for the elements of the compound waveform specified.

Error Messages:

Table C-8. wave runindexlist Error Messages

Error Message	Description
No compound waveform found	The input waveform is not compound or the regular expression used has not found any compound waveforms.
Too many waveforms found	The regular expression used to describe the input waveform name has identified more than one compound waveform.

Examples

```
wave runindexlist :top:sigout

wave runindexlist -run TEMP=120 $wave
```

wave runparameters

Displays a list of run parameters for the compound waveform specified. It is useful for iterating through each element individually.

Usage

```
wave runparameters waveform_name
```

Parameters

- ***waveform_name***

  (Required) Specifies the name of the waveform for which the run parameter list is to be retrieved. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform. You can use more than one option in the same command.

Return Values

List. Returns a list of run parameters for the compound waveform specified.

Error Messages:

Table C-9. wave runparameters Error Messages

Error Message	Description
No compound waveform found	The input waveform is not compound or the regular expression used has not found any compound waveforms.
Too many waveforms found	The regular expression used to describe the input waveform name has identified more than one compound waveform.

Examples

```
wave runparameters :top:sigout
```

wave runparametervalue

Displays the value of a specified run parameter associated with the run_index and compound waveform specified.

Usage

wave runparametervalue -param *param_name* -run *run_index* *waveform_name*

Parameters

- **-param *param_name***

  (Required) Specifies the name of the run parameter for which the value is to be retrieved.

- **-run *run_index***

  (Required) Specifies the value of the run index for which the run parameter value is to be retrieved.

- ***waveform_name***

  (Required) Specifies the name of the waveform for which the run parameter value is to be retrieved. You can use the extended options detailed in Selecting Waveforms in Tcl to select the waveform. You can use more than one option in the same command.

Return Values

Returns the value of the specified run parameter associated with the run index and compound waveform specified.

Error Messages:

Table C-10. wave runparametervalue Error Messages

Error Message	Description
No compound waveform found	The input waveform is not compound or the regular expression used has not found any compound waveforms.
Too many waveforms found	The regular expression used to describe the input waveform name has identified more than one compound waveform.
The object ‘<name>’ is not a compound waveform element	The run index does not identify an element of the compound waveform.
Parameter ‘<param_name>’ is not found in compound waveform element ‘<name>’	The parameter specified has not been found in the specified element of the compound waveform.

Examples

```
wave runparametervalue -param CORNER -run 2 :top:sigout
```

wave show

Defines the default net representation that is used when the **add wave** command is issued *without* the -show option. If the -show option is specified with the **add wave** command (add wave -show *representation*), it temporarily overrides the **wave show** defaults during the execution of the **add wave** command and then returns to the **wave show** defaults.

Usage

wave show *representation* [|*representation2*| … |*representationN*]

Parameters

- *representation* [|*representation2* | ...| *representationN*]

  (Optional) Specifies how signals are to be represented as a waveform. A *representation* is a complete representation of a net and is composed of an *analysis*, a *discipline*, and a *physic*. A dot (**.**) separates the *analysis* portion from the *discipline* and *physic* portion. There is no separator between the *discipline* and the *physic*, as follows:

  analysis.disciplinephysic

  If more than one *representation* is specified, use a pipe (**|**) to separate them:

  analysis.disciplinephysic|analysis.disciplinephysic|analysis.disciplinephysic

  Refer to “Supported Net Representation Components” on page 1245 for the supported representations.

Description

The defined representation defaults remains in effect until another **wave show** command is issued or until the end of the script.

If you do not define the waveform representation with this command, the default is defined as follows:

```
tran.v|ac.vdb|ac.vp
```

Examples

```
wave show ac.vdb
# The analysis is AC, the discipline is V, and the physic is DB.

wave show ffst.vdb.h(1)
# The analysis is FFST, the discipline is V, and the physic is DB for
# the primary_ physic, and H(1) for the secondary_physic.
```

wave showgridlines

Turns gridlines on or off in all new windows.

Usage

```
wave showgridlines {on | off}
```

Parameters

- **{on | off}**

  (Required) Specifies whether gridlines should be displayed in all new windows.

Description

Turns gridlines on or off in all new windows. Windows already created will not be modified by this command.

Examples

```
wave showgridlines on
```

wave showzerolevels

Activates the zero-levels in the rows where the waveform is displayed.

Usage

```
wave showzerolevels {on | off | horizontal | vertical}
```

Parameters

- **{on | off | horizontal | vertical}**

  (Required)

  - **on** - Activates the display of level lines at X=0 and Y=0.
  - **off** - Deactivates the display of all zero level lines.
  - **horizontal** - Activates the display of a level line at Y=0.
  - **vertical** - Activates the display of a level line at X=0.

Description

Activates the zero-levels in the rows where the waveform is displayed. On/Off enables or disables both horizontal and vertical levels. Horizontal/Vertical enables only the appropriate horizontal or vertical levels. Windows already created will not be modified by this command.

Examples

```
wave showzerolevels horizontal
```

wave tandem mode

Activates or deactivates Tandem Mode.

Usage

```
wave tandem mode {on | off}
```

Parameters

- {**on** | **off**}

  (Required) Specifies whether Tandem Mode is turned on or off.

Description

Tandem model enables you to plot waveforms with the same name from different databases. Refer to "Plotting Waveforms With the Same Name Using Tandem Mode" on page 126.

Examples

```
wave tandem mode on
```

wave tile

Arranges multiple windows in the specified tiling style.

Usage

```
wave tile [-cascade | -horizontal | -vertical | -grid]
```

Parameters

- -cascade

  (Optional) Arranges the windows in a “cascading” fashion. Windows are overlapped while keeping an area of each window exposed so you can click on an individual window to bring it to the front.

- -horizontal

  (Optional) Arranges the windows in rows. Windows are displayed “above” and “below” each other.

- -vertical

  (Optional) Arranges the windows in columns. Windows are displayed side by side.

- -grid

  (Optional) Arranges the windows in a grid fashion. Windows are displayed side by side in columns as well as “above” and “below” each other in rows.

Examples

```
wave tile -cascade
```

wave windowdecoration

Shows or hides the decoration of wave windows.

Usage

```
wave windowdecoration {show | hide}
```

Parameters

- {**show** | **hide**}

  (Required) Specifies whether decoration should be shown or hidden in all windows.

Description

Shows or hides the decoration borders (title, buttons and scrollbars) of wave windows.

Examples

```
wave windowdecoration hide
```

wave windowlist

Returns a list of all open graph windows.

Usage

`wave windowlist`

Parameters

- None

Examples

```
# Retrieves the x-axis scale type for each open window

foreach {i} [wave windowlist] {
   puts [wave xaxis]
}
```

wave xaxis

Sets the x-axis unit title, and the x-axis scale to one of: linear, base 2 log, or base 10 log.

Usage

```
wave xaxis [-title title] [-scale {linear | log2 | log10}]
[-window window_name]
```

Parameters

- - title title

  (Optional) Specifies the text to be displayed in place of the x-axis unit title.

- -scale { linear | log2 | log10 }

  (Optional) Specifies the type of scale to use for the x axis. Choose from linear, log2, or log10. If this option is not specified, this command returns the current x-axis scale. This option corresponds to the X Axis Properties Dialog Box.

- -window *window_name*

  (Optional) If specified, this option applies the command to the specified window. Otherwise, the command applies to the active window.

Return Values

None, if “-title” or “-scale” argument is specified.

If neither the “-title” or “-scale” argument is specified, this command returns the current x-axis scale.

Examples

```
wave xaxis -title "User Time"
```

```
wave axis -scale log10 -window Wave:2
```

wave yaxis

Used to alter y-axis settings such as the y-axis name, alignment, unit title, scale, and alignment for the specified row.

Usage

```
wave yaxis -row rowId [-axis axis_name] [-title title]
[-scale {linear | log2 | log10}] [-window window_name]
[-align {y1-refLevel,..}] [-spacing {y1-spacing,..}]
```

Parameters

- **-row *rowId***

  (Required) Specifies the row that the command is to be applied to. The ***rowId*** of the top row is 1, increasing by 1 each row moving downwards.

  > **Note**
  > ***rowId*** identifiers are dynamic, with the top row always taking the value of 1.

- -axis *axis_name*

  (Optional) Specifies the name of the y axis to which the command is to be applied. By default the axis is named “Y1”.

- - title title

  (Optional) Specifies the text to be displayed in place of the y-axis unit title.

- -scale { linear | log2 | log10 }

  (Optional) Specifies the type of scale for the y axis. Choose from linear, log base 2, or log base 10. If no value is specified, the current y-axis scale is returned. Specifying this option is equivalent to setting the Axis Scale field on the Axis Properties Dialog Box.

- -window *window_name*

  (Optional) If specified, the changes are applied to the specified window. Otherwise, the command applies to the active window.

- -align {y1-refLevel,.. }

  (Optional) Specifies how the axes are aligned. The -axis argument is the reference axis. Values are specified in a comma-separated list of doubles starting with the first, Y1, y axis. For example:

  -align {0.0,10}

- -spacing {y1-spacing,.. }

  (Optional) Specifies the spacing for the axes. The -axis argument is the reference axis. Values are specified in a comma-separated list of positive doubles starting with the first, Y1, y axis. For example:

  -spacing {10.0,20.0}

Description

Used to alter y-axis settings such as the y-axis name, unit title and the scale, for the specified row. If -title, -scale, -align, or -spacing are not specified, this command returns the current y-axis scale.

Examples

This example applies some changes to the top row (rowId=1). Sets the title of the y-axis unit title to “User Time” and specifies that the y axis uses a logarithmic scale with base 2:

```
wave yaxis -row 1 -title "User Time" -scale log2
```

This example sets the scale of the axis in the first row of window **Wave:2** to log2:

```
wave yaxis -scale log10 -window "Wave:2"
```

When a row contains multiple y axes you can use the -axis argument to specify which axis to apply the scale to. Without the -axis argument, the command will always apply the scale to the first axis, Y1. The following command sets the scale of the axis Y1, located in the second row, to log10:

```
wave yaxis -row 2 -axis Y1 -scale log10
```

In this example alignment and spacing are set for the axes:

```
wave yaxis -row 1 -axis y2 -align {0.0, 5.0} -spacing {10.0, 20.0}
```

wave zoomfull

Zooms out to show the entire x axis of the specified graph window.

Usage

```
wave zoomfull [-window window_name]
```

Parameters

- -window *window_name*

  (Optional) Specifies the window to apply the zoom to. If this is not specified, the zoom applies to the active window.

Examples

```
wave zoomfull -window "Wave:2"
```

wave zoomin

Zooms in on the x axis of the specified graph window, by the specified factor.

Usage

```
wave zoomin [-window window_name] [factor]
```

Parameters

- -window *window_name*

  (Optional) Specifies the window to apply the zoom to. If this is not specified, the zoom applies to the active window.

- *factor*

  (Optional) Specifies the zoom factor to apply. If this is not specified, the default value of 2.0 is used.

Examples

- Zoom x axis of the active window in by a factor of 2.0:

  ```
  wave zoomin
  ```

- Zoom x axis of the window "Wave:2" in by a factor of 2.0:

  ```
  wave zoomin -window "Wave:2"
  ```

- Zoom x axis of the active window in by a factor of 4.0:

  ```
  wave zoomin 4.0
  ```

wave zoomlast

Undoes the most recent zoom operation in the specified window.

Usage

wave zoomlast [-window *window_name*]

Parameters

- -window *window_name*

  (Optional) Specifies the window in which to undo the zoom. If this is not specified, the undo applies to the active window.

Examples

```
# Undo the previous zoom operation in the window "Wave:2":
wave zoomlast -window "Wave:2"
```

wave zoomout

Zooms out on the x axis of the specified graph window, by the specified factor.

Usage

wave zoomout [-window *window_name*] [*factor*]

Parameters

- -window *window_name*

  (Optional) Specifies the window to apply the zoom to. If this is not specified, the zoom applies to the active window.

- *factor*

  (Optional) Specifies the zoom factor to apply. If this is not specified, the default value of 2.0 is used.

Note

Zooming over the y axis is done through an option in the add wave command.

Examples

- Zoom x axis of the active window out by a factor of 2.0:

```
wave zoomout
```

- Zoom x axis of the window "Wave:2" out by a factor of 2.0:

```
wave zoomout -window "Wave:2"
```

- Zoom x axis of the active window out by a factor of 4.0:

```
wave zoomout 4.0
```

wave zoomrange

Zooms in on the x axis to a range bounded by the specified start and/or end points.

Usage

wave zoomrange [-window *window_name*] [*start*] [*end*]

Parameters

- -window *window_name*

  (Optional) Specifies the window in which to apply the zoom. If this is not specified, the zoom applies to the active window.

- *start*

  (Optional) Specifies the start point for the range to zoom to.

- *end*

  (Optional) Specifies the end point for the range to zoom to.

Note

If only one of *start* and *end* is specified, *start* is assumed to be 0 and the specified number is assumed to be the endpoint. If neither *start* nor *end* is specified, the start- and endpoints for the current zoom level are returned.

Examples

Zoom along the x axis in the current window to a start time of 20 ns and an end time of 100 ns:

```
wave zoomrange 20n 100n
```

wfc

Invokes the EZwave Waveform Calculator to calculate the expression entered.

Usage

`wfc` ***`expression`***

`wfc` *`-save`* `{`*`name=<expression>`*`}`

Parameters

- ***expression***

  (Required) An expression supported by the EZwave waveform calculator, where ***expression*** can be name=<subExpression>.

- *-save* {*name=<expression>*}

  (Optional) Saves the variable if it is named.

  When *expression* starts with a name, then a variable with this name is instantiated in the Waveform Calculator, and the name is returned. For waveforms, a default name is proposed (e.g. wf0).

Return Values

The result value of the expression. It can be one of the following:

- a waveform object handle
- a single number
- a list of data represented as a Tcl list of strings for a one-dimensional array
- a list of data pairs represented as a Tcl list of string pairs grouped by parentheses for a two-dimensional array

Description

Invokes the EZwave Waveform Calculator to calculate the expression entered.

If ***expression*** is enclosed in double quotes (**" "**), value substitution is enabled and all strings beginning with a dollar sign (**$**) are replaced by the variable they name. If an expression is enclosed in braces (**{ }**), value substitution is disabled and the expression is evaluated as presented.

This command differs from evalExpression in the type of output it returns.

Examples

```
set wf_diff [wfc {wf("<tutorial/Time-Domain_Results>v_load") -
wf("<tutorial/Time-Domain_Results>v_middle")}]
add wave $wf_diff
```

Related Topics

save

write jpeg

Exports the contents of a window to a specified JPEG image file.

Usage

```
write jpeg file_name [-window window_name]
[-header "dbTitle simuTitle wndTitle fromSimulator time date user
machine"]
[-blackbackground | -whitebackground | -monochrome | -documentation
  |-colorasdisplayed]
[-visiblewindows]
[-resolution {screen | printerlow | printerhigh}]
```

Parameters

- ***file_name***

  (Required) Specifies the file (with optional full path) to which the window contents are to be saved. If no path is specified, it defaults to the location from which EZwave was launched.

- -window *window_name*

  (Optional) If this option is specified, the contents of the specified window is exported. Otherwise, the contents of the active window is exported.

- -header "*dbTitle simuTitle wndTitle fromSimulator time date user machine*"

  (Optional) If this option is specified, include one or more of the keywords to add header text to the output. The first three keywords add the database title, netlist title, and window title to the output. You can also add the time, date, user name, and machine name, optionally specifying *fromSimulator* if you want these values to be the same as the simulator, rather than current values.

- -blackbackground | -whitebackground | -monochrome | -documentation | -colorasdisplayed

  (Optional) These options specify the color scheme for the exported image.

- -visiblewindows

  (Optional) Specifies that the image to be created contains all windows as displayed in the EZwave viewer. This argument is particularly useful when exporting tiled windows.

- -resolution { screen | printerlow | printerhigh }

  (Optional) This option specifies the resolution of the exported image file. Selecting screen sets the output resolution to the screen resolution. printerlow and printerhigh sets the output resolution to approximately 4x and 16x the screen resolution, respectively.

Examples

```
write jpeg /user/adc12.jpg
```

write png

Exports the contents of a window to a specified PNG image file.

Usage

```
write png file_name [-window window_name]
[-header "dbTitle simuTitle wndTitle fromSimulator time date user
machine"]
[-blackbackground | -whitebackground | -monochrome | -documentation
  |-colorasdisplayed]
[-visiblewindows]
[-resolution {screen | printerlow | printerhigh}]
```

Parameters

- ***file_name***

  (Required) Specifies the file (with optional full path) to which the window contents are to be saved. If no path is specified, it defaults to the location from which EZwave was launched.

- -window *window_name*

  (Optional) If this option is specified, the contents of the specified window is exported. Otherwise, the contents of the active window is exported.

- -header *"dbTitle simuTitle wndTitle fromSimulator time date user machine*"

  (Optional) If this option is specified, include one or more of the keywords to add header text to the output. The first three keywords add the database title, netlist title, and window title to the output. You can also add the time, date, user name, and machine name, optionally specifying *fromSimulator* if you want these values to be the same as the simulator, rather than current values.

- -blackbackground | -whitebackground | -monochrome | -documentation | -colorasdisplayed

  (Optional) These options specify the color scheme for the exported image.

- -visiblewindows

  (Optional) Specifies that the image to be created contains all windows as displayed in the EZwave viewer. This argument is particularly useful when exporting tiled windows.

- -resolution { screen | printerlow | printerhigh }

  (Optional) This option specifies the resolution of the exported image file. Selecting screen sets the output resolution to the screen resolution. printerlow and printerhigh sets the output resolution to approximately 4x and 16x the screen resolution, respectively.

Examples

```
write png /user/adc12.png
```

write wave

Outputs window contents in PostScript format. It can be sent to a printer or a specified location.

Usage

write wave [*file*.ps | -file *file*.ps | -printer *printer_name*]
[-window *window_name*] [-landscape | -portrait]
[-papersize {a3 | a4 | a5 | b4 | b5 | letter | tabloid | ledger | legal |
executive}] [-copies number] [-allwaveforms | -visiblewaveforms]
[-activewindow | -allwindows | -visiblewindows] [-eps]
[-header *"dbTitle simuTitle wndTitle fromSimulator time date user machine"*]

Parameters

- -file *file*.ps

  (Optional) Sets the output file name in PostScript format.

- -printer *printer_name*

  (Optional) Sets the printer in the network.

- -window *window_name*

  (Optional) Sets the window to be printed. In case no -window is specified, the active window is printed.

- -landscape

  (Optional) Sets the printing orientation to be landscape (default).

- -portrait

  (Optional) Sets the printing orientation as portrait.

- -papersize { a3 | a4 | a5 | b4 | b5 | letter | tabloid | ledger | legal | executive }

  (Optional) Specifies the paper size for printing. These paper sizes correspond to the sizes found in the **File** > **Print** dialog box.

- -copies *number*

  (Optional) Specifies the number of copies to print.

- -allwaveforms | visiblewaveforms

  (Optional) Specifies whether to print all waveforms or just the visible waveforms. If this option is not specified, only visible waveforms are printed.

- -activewindow | -allwindows | -visiblewindows

  (Optional) Specifies which windows to print. By default, when multiple windows are open, only the active window is printed. If “-allwindows” is specified, all windows are printed, one per page. If “-visiblewindows” is specified, only the visible windows are printed, also one per page.

- -eps

  (Optional) If this option is set, the command generates an Encapsulated PostScript file.

- -header *"dbTitle simuTitle wndTitle fromSimulator time date user machine"*

  (Optional) If this option is specified, include one or more of the keywords to add header text to the output. The first three keywords add the database title, netlist title, and window title to the output. You can also add the time, date, user name, and machine name, optionally specifying *fromSimulator* if you want these values to be the same as the simulator, rather than current values.

Examples

```
write wave dc.ps
```

External Tcl Command Support

The EZwave Tcl interpreter recognizes commands from Questa® SIM (formerly ModelSim®) and Questa® ADMS™. Depending on the command, the interpreter will ignore the command, or the EZwave viewer will execute it by internally applying corresponding actions.

Table C-11 and Table C-12 list the level of support that the EZwave Tcl interpreter maintains for Questa SIM and Questa ADMS commands, respectively.

- Ignore — Indicates that the EZwave Tcl interpreter ignores the command
- Supported — Indicates that the EZwave viewer processes the command internally

Table C-11. Questa SIM Command Support

Command	EZwave Action	Command	EZwave Action
.main clear	Ignore	power on	Ignore
abort	Ignore	power report	Ignore
add atv	Ignore	power reset	Ignore
add button	Ignore	precision	Ignore
add dataflow	Ignore	printenv	Supported
add list	Ignore	process report	Ignore
add log	Ignore	profile clear	Ignore
add memory	Ignore	profile interval	Ignore
add testbrowser	Ignore	profile off	Ignore
add watch	Ignore	profile on	Ignore
add wave	Supported	profile option	Ignore
add_cmdhelp	Ignore	profile reload	Ignore

Table C-11. Questa SIM Command Support (cont.)

Command	EZwave Action	Command	EZwave Action
add_menu	Ignore	profile report	Ignore
add_menucb	Ignore	project	Ignore
add_menuitem	Ignore	pwd	Supported
add_separator	Ignore	questasim	Ignore
add_submenu	Ignore	quietly	Ignore
addtime	Ignore	quit	Supported
alias	Ignore	qverilog	Ignore
assertion active	Ignore	radix	Supported
assertion count	Ignore	radix define	Supported
assertion fail	Ignore	radix names	Supported
assertion pass	Ignore	radix list	Supported
assertion profile	Ignore	radix delete	Supported
atv log	Ignore	readers	Ignore
batch_mode	Supported	realtotime	Ignore
bd	Ignore	record	Ignore
bookmark add wave	Ignore	report	Ignore
bookmark delete wave	Ignore	restart	Ignore
bookmark goto wave	Ignore	restore	Ignore
bookmark list wave	Ignore	resume	Ignore
bp	Ignore	right	Ignore
cd	Supported	run	Ignore
cdbg	Ignore	runstatus	Ignore
change	Ignore	sccom	Ignore
change_menu_cmd	Ignore	scaletime	Ignore
bookmark goto wave	Ignore	sscom	Ignore
bookmark list wave	Ignore	scgenmod	Ignore
check contention config	Ignore	sdfcom	Ignore
check contention off	Ignore	search	Ignore
check float add	Ignore	searchlog	Ignore
check float config	Ignore	see	Ignore

Table C-11. Questa SIM Command Support (cont.)

Command	EZwave Action	Command	EZwave Action
check float off	Ignore	seetime	Ignore
check stable off	Ignore	setenv	Supported
check stable on	Ignore	shift	Ignore
checkpoint	Ignore	show	Ignore
classinfo	Ignore	simstats	Ignore
compare add	Supported	stack	Ignore
compare annotate	Ignore	status	Ignore
compare clock	Supported	step	Ignore
compare configure	Supported	stop	Ignore
compare delete	Ignore	subtime	Ignore
compare end	Supported	suppress	Ignore
compare info	Supported	tb	Ignore
compare list	Supported	tcheck_set	Ignore
compare options	Supported	tcheck_status	Ignore
compare reload	Ignore	toggle add	Ignore
compare reset	Ignore	toggle disable	Ignore
compare run	Supported	toggle enable	Ignore
compare savediffs	Ignore	toggle report	Ignore
compare saverules	Supported	toggle reset	Ignore
compare see	Ignore	tr color	Ignore
compare start	Supported	tr id	Ignore
compare stop	Ignore	tr order	Ignore
compare update	Ignore	transcribe	Ignore
configure	Ignored	transcript	Ignore
context	Ignore	transcript file	Ignore
coverage analyze	Ignore	triage	Ignore
coverage attribute	Ignore	tssi2mti	Ignore
coverage goal	Ignore	typespec	Ignore
coverage ranktest	Ignore	ui_vvmode	Ignore
coverage tag	Ignore	unsetenv	Support

Table C-11. Questa SIM Command Support (cont.)

Command	EZwave Action	Command	EZwave Action
coverage testnames	Ignore	up	Ignore
coverage unlinked	Ignore	validtime	Ignore
coverage weight	Ignore	vcd add	Ignore
coverage clear	Ignore	vcd checkpoint	Ignore
coverage exclude	Ignore	vcd comment	Ignore
coverage reload	Ignore	vcd dumpports	Ignore
coverage report	Ignore	vcd dumpportsall	Ignore
coverage save	Ignore	vcd dumpportsflush	Ignore
dataset alias	Supported	vcd dumpportslimit	Ignore
dataset clear	Supported	vcd dumpportsoff	Ignore
dataset close	Supported	vcd dumpportson	Ignore
dataset config	Ignore	vcd file	Ignore
dataset info	Supported	vcd files	Ignore
dataset list	Supported	vcd flush	Ignore
dataset open	Supported	vcd limit	Ignore
dataset rename	Supported	vcd off	Ignore
dataset restart	Ignore	vcd on	Ignore
dataset save	Supported	vcd2wlf	Ignore
dataset snapshot	Ignore	vcom	Ignore
delete	Supported	vcover attributes	Ignore
describe	Ignore	vcover ranktest	Ignore
disablebp	Ignore	vcover testnames	Ignore
disable_menu	Ignore	vcover convert	Ignore
disable_menuitem	Ignore	vcover merge	Ignore
divtime	Ignore	vcover rank	Ignore
do	Ignore	vcover report	Ignore
down	Ignore	vcover stats	Ignore
drivers	Ignore	vdbg	Ignore
dumplog64	Ignore	vdel	Ignore
echo	Ignore	vdir	Ignore

Table C-11. Questa SIM Command Support (cont.)

Command	EZwave Action	Command	EZwave Action
edit	Ignore	vencrypt	Ignore
enablebp	Ignore	verror	Ignore
enable_menu	Ignore	vgencomp	Ignore
enable_menuitem	Ignore	vhencrypt	Ignore
encoding	Ignore	view	Ignore
environment	Supported	virtual count	Ignore
eqtime	Ignore	virtual define	Ignore
examine	Supported	virtual delete	Ignore
exit	Supported	virtual describe	Ignore
fcover configure	Ignore	virtual expand	Ignore
find	Ignore	virtual function	Ignore
find analogs	Supported	power add	Ignore
find digitals	Supported	power off	Ignore
find nets	Supported	virtual hide	Ignore
find signals	Supported	virtual log	Ignore
find infiles	Ignore	virtual nohide	Ignore
find insource	Ignore	virtual nolog	Ignore
formatTime	Ignore	virtual region	Ignore
force	Ignore	virtual save	Ignore
fsm	Ignore	virtual show	Ignore
gdb dir	Ignore	virtual signal	Ignore
getactivecursortime	Supported	virtual type	Ignore
getactivemarkertime	Ignore	vlib	Ignore
gtetime	Ignore	vlog	Ignore
gttime	Ignore	vmake	Ignore
help	Ignore	vmap	Ignore
history	Supported	vopt	Ignore
inttotime	Ignore	vsim	Ignore
jobspy	Ignore	vsim<info>	Ignore
layout	Ignore	vsim_break	Ignore

Table C-11. Questa SIM Command Support (cont.)

Command	EZwave Action	Command	EZwave Action
lecho	Ignore	vsource	Ignore
left	Ignore	wave	Supported
log	Ignore	wave create	Ignore
lshift	ignore	wave edit	Ignore
lsublist	Ignore	wave export	Supported
ltetime	Ignore	wave exists	Supported
lttime	Ignore	wave modify	Ignore
macro_option	Ignore	when	Ignore
mem compare	Ignore	where	Ignore
mem display	Ignore	wlf2log	Ignore
mem list	Ignore	wlf2vcd	Ignore
mem load	Ignore	wlfman	Ignore
mem save	Ignore	wlfrecover	Ignore
mem search	Ignore	write cell_report	Ignore
messages	Ignore	write format	Ignore
modelsim	Ignore	write list	Ignore
next	Ignore	write preferences	Ignore
neqtime	Ignore	write report	Ignore
noforce	Ignore	write timing	Ignore
nolog	Ignore	write transcript	Ignore
notepad	Ignore	write tssi	Ignore
noview	Ignore	write wave	Supported
nowhen	Ignore	xml2ucdb	Ignore
onbreak	Ignore		
onElabError	Ignore		
onerror	Ignore		
onfinish	Ignore		
pa	Ignore		
pause	Ignore		
play	Ignore		

Table C-11. Questa SIM Command Support (cont.)

Command	EZwave Action	Command	EZwave Action
pop	Ignore		
property list	Ignore		
property wave	Ignore		
push	Ignore		

Table C-12. Questa ADMS Command Support

Command	EZwave Action	Command	EZwave Action
add list	Ignore	vcd add	Ignore
add log	Ignore	vcd checkpoint	Ignore
add wave	Supported	vcd comment	Ignore
batch_mode	Supported	vcd dumpports	Ignore
cd	Supported	vcd dumpportsall	Ignore
change	Ignore	vcd dumpportsflush	Ignore
checkpoint	Ignore	vcd dumpportslimit	Ignore
debuginfo	Ignore	vcd dumpportsoff	Ignore
drivers	Ignore	vcd dumpportson	Ignore
env	Supported	vcd file	Ignore
examine	Supported	vcd files	Ignore
exit	Supported	vcd flush	Ignore
find	Supported	vcd limit	Ignore
force	Ignore	vcd off	Ignore
if	Supported	vcd on	Ignore
import_adms	Ignore	vacom	Ignore
imports_ms	Ignore	vadel	Ignore
isavewdb	Ignore	vadir	Ignore
ms	Ignore	valib	Ignore
noforce	Ignore	valog	Ignore
onRunDone	Ignore	vamake	Ignore
probe	Supported	vamap	Ignore
pwd	Supported	vamatch	Ignore

Table C-12. Questa ADMS Command Support (cont.)

Command	EZwave Action	Command	EZwave Action
quit	Supported	vasetlib	Ignore
restart	Ignore	vasim	Ignore
restore	Ignore	vaspi	Ignore
run	Ignore	vcd2wlf	Ignore
savetranscript	Ignore	vcd add	Ignore
savewaveconfig	Supported	view	Ignore
savewavewindow	Supported	vsimauth	Ignore
savewdb	Supported	vsimdate	Ignore
simparam	Ignore	vsimid	Ignore
splitio	Ignore	vsimversionstring	Ignore
statistics	Ignore	watch	Ignore
stop	Ignore	write list	Ignore
usewaveconfig	Supported		

Tcl Scripting Examples

This section includes some Tcl examples.

Tcl Waveform Calculator Batch Commands Example 1206
Additional User-Defined Procedures 1208
Opening a Database File 1208
Setting Global Parameters 1208
Taking Waveform Measurements 1209
ASCII File Output 1210
Tcl User-Defined Functions 1211
Creating a User-Defined Tcl Function 1211
Loading User-Defined Tcl Functions Automatically 1211
Waveform Comparison With Tcl Examples 1213
Compare All Waveforms With Default Options 1213
Compare All Waveforms and Scalars 1213
Compare All Terminal Waveforms From Transient Analysis With Default Options 1214
Compare All Waveforms Using a Clocked Comparison 1214
Compare Specific Waveforms With Modified Tolerances, Write a Report 1215
Compare Waveforms Using All Available Comparison Commands 1215
Delaying Reference Waveforms During Comparison 1215
Delaying Some Result Waveforms (Not All) During Comparison 1216
Compare Waveforms Using the -label Argument 1217
Compare Waveforms Using the -start and -end Arguments 1218
Export a Report of Comparison 1219
Using find Commands 1220

Tcl Waveform Calculator Batch Commands Example

You can use Tcl scripting to issue batch commands to the EZwave waveform calculator. The wfc function enables full access to all the calculation operations of the waveform calculator.

In the following Tcl script example, *tut1_meas.tcl*, the wfc command is used to calculate several different waveform measurements. The script opens a waveform database file, *meas.wdb*, performs several measurement operations, and sends output to an ASCII file, *meas.out*.

```
#!/usr/local/bin/tclsh
# ###################################################
# additional user defined procedures
proc greater {a b} {
   return [ expr { ($a > $b) ? $a : $b } ]
}
# ###################################################
# open database file
dataset open $env(MGC_AMS_HOME)/examples/ezwave/meas.wdb
puts "\nExtracting from EZwave Post Processing"
# ###################################################
# global parameter settings
# .param vdd=3.3
wfc { vdd = 3.3 }
# ###################################################
# .meas tran TD1 when v(in)='vdd/2' td=5ns rise=1
# .meas tran TD2 when v(in)='vdd/2' td=5ns rise=2
wfc { rt_in_ = risetime(wf("<meas/TRAN>V(IN)"), x_start = 5e-9, baseline =
vdd/4, topline = vdd´3/4, option = "wf") }
set TD1 [ wfc { td1_ = wftodata(rt_in_)[0][0] } ]
set TD2 [ wfc { td2_ = wftodata(rt_in_)[1][0] } ]
# ###################################################
# .meas tran T1 trig at=TD1
# targ v(q0) val='vdd/2' td=5ns rise=1
wfc { rt_q0_ = risetime(wf("<meas/TRAN>V(Q0)"), x_start = greater(td1_,
5e-9), baseline = vdd/4, topline = vdd´3/4, option = "wf") }
set T1 [ wfc { wftodata(rt_q0_)[0][0] - td1_ } ]
# ###################################################
# .meas tran T2 trig at=TD2
# targ v(q0) val='vdd/2' td=TD2 fall=1
wfc { ft_q0_ = falltime(wf("<meas/TRAN>V(Q0)"), x_start = greater(td2_,
td2_), baseline = vdd/4, topline = vdd´3/4, option = "wf") }
set T2 [ wfc { wftodata(ft_q0_)[0][0] - td2_ } ]
# ###################################################
# .meas tran T2_C2 trig at=TD2
# targ v(q0) val='vdd/2' td=5ns cross=2
wfc { cr_q0_ = crossing(wf("<meas/TRAN>V(Q0)"), x_start = greater(td2_,
5e-9), ylevel = vdd/2, slopetrigger = "either", option = "value") }
set T2_C2 [ wfc { cr_q0_[1] - td2_ } ]
# ###################################################
# .meas tran T2_R trig at=TD2
# targ v(q0) val='vdd/2' td=5ns rise=last
wfc { cr1_q0_ = crossing(wf("<meas/TRAN>V(Q0)"), x_start = greater(td2_,
5e-9), ylevel = vdd/2, slopetrigger = "rising", option = "wf") }
wfc { last_ = size(cr1_q0_) - 1 }
set T2_R [ wfc { wftodata(cr1_q0_)[last_][0] - td2_ } ]
# ###################################################
# open an ASCII file for saving measurement results
# results are in nanoseconds
set unit {N}
set factor 1e9
set fileout [ open meas.out w+ ]
puts $fileout "\nExtracted from EZwave Post Processing"
puts $fileout [format " TD1\t= %8.4f%s" [ expr $TD1 * $factor ] $unit ]
puts $fileout [format " T1\t= %8.4f%s" [ expr $T1 * $factor ] $unit ]
puts $fileout [format " TD2\t= %8.4f%s" [ expr $TD2 * $factor ] $unit ]
puts $fileout [format " T2\t= %8.4f%s" [ expr $T2 * $factor ] $unit ]
puts $fileout [format " T2_C2\t= %8.4f%s" [ expr $T2_C2 * $factor ] $unit]
```

```
puts $fileout [format " T2_R\t= %8.4f%s" [ expr $T2_R *$factor ] $unit]
close $fileout
```

This Tcl script can be broken down into several distinct functional areas, described in the following topics.

Additional User-Defined Procedures 1208
Opening a Database File 1208
Setting Global Parameters 1208
Taking Waveform Measurements 1209
ASCII File Output 1210

Additional User-Defined Procedures

You can use any Tcl native functions and define any procedures using Tcl syntax.

This is shown in the following statements:

```
proc greater {a b} {
   return [ expr { ($a > $b) ? $a : $b } ]
}
```

Opening a Database File

You can open a database file using a Tcl statement.

The database file *meas.wdb* is opened using the following Tcl statement:

```
dataset open meas.wdb
```

Setting Global Parameters

This example shows how to set global parameters using a Tcl function.

For the following .param statement in a SPICE netlist file, the value 3.3 is assigned to the parameter variable vdd:

```
.param vdd=3.3
```

The equivalent Tcl function is as follows:

```
wfc { vdd = 3.3 }
```

Taking Waveform Measurements

Electrical specifications (using the .meas command) are computed based on simulation results and typically printed to an ASCII file. You can use Tcl scripting to take the same measurements.

In this example, after a specified time delay (td) of 5 nanoseconds, a measurement is taken to find the exact time when the signal v(in) crosses the voltage level vdd/2 in the first rising event. The result is saved in the variable TD1.

The following example shows how this is implemented as a .meas statement in a SPICE netlist file:

```
.meas tran TD1 when v(in)='vdd/2' td=5ns rise=1
```

In a second .meas statement, after a 5 nanosecond time delay, a measurement is taken to determine the time when the signal v(in) crosses the voltage level vdd/2 in the second rising event. The result is saved in the variable TD2.

```
.meas tran TD2 when v(in)='vdd/2' td=5ns rise=2
```

These two measurements (TD1 and TD2) can be implemented using Tcl scripting as in the following example:

```
wfc { rt_in_ = risetime(wf("<meas/TRAN>V(IN)"), x_start = 5e-9, baseline =
vdd/2, topline = vdd/2, option = "wf") }
set TD1 [ wfc { td1_ = wftodata(rt_in_)[0][0] } ]
set TD2 [ wfc { td2_ = wftodata(rt_in_)[1][0] } ]
```

For a third .meas statement, after a 5 nanoseconds time delay, a measurement is taken to find the difference between the result in TD1 and when the signal v(q0) crosses the voltage level vdd/2 in the first rising event. The result is saved in the variable T1.

```
.meas tran T1 trig at=TD1
targ v(q0) val='vdd/2' td=5ns rise=1
```

The Tcl scripting equivalent to the .meas statement is as follows:

```
wfc { rt_q0_ = risetime(wf("<meas/TRAN>V(Q0)"), x_start = greater(td1_,
5e-9), baseline = vdd/2, topline = vdd/2, option = "wf") }
set T1 [ wfc { wftodata(rt_q0_)[0][0] - td1_ } ]
```

For the fourth .meas statement, after a TD2 time delay, a measurement is taken to determine the difference between TD2 and when the signal v(q0) crosses the voltage level vdd/2 in the first falling event. The result is saved in the variable T2.

```
.meas tran T2 trig at=TD2
targ v(q0) val='vdd/2' td=TD2 fall=1
```

The Tcl scripting equivalent is as follows:

```
wfc { ft_q0_ = falltime(wf("<meas/TRAN>V(Q0)"), x_start = greater(td2_,
td2_), baseline = vdd/2, topline = vdd/2, option = "wf") }
set T2 [ wfc { wftodata(ft_q0_)[0][0] - td2_ } ]
```

For the fifth .meas statement, after a 5 nanosecond time delay, a measurement is taken to find the difference between TD2 and when the signal v(q0) crosses the voltage level vdd/2 in the second crossing event, either rising or falling. The result is saved in the variable T2_C2.

```
.meas tran T2_C2 trig at=TD2
targ v(q0) val='vdd/2' td=5ns cross=2
```

The Tcl scripting equivalent is as follows:

```
wfc { cr_q0_ = crossing(wf("<meas/TRAN>V(Q0)"), x_start = greater(td2_,
5e-9), ylevel = vdd/2, slopetrigger = "either", option = "value") }
set T2_C2 [ wfc { cr_q0_[1] - td2_ } ]
```

In the final .meas statement, after a 5 nanosecond time delay, a measurement is taken to find the difference between the time specified by TD2 and when the signal v(q0) crosses the voltage level vdd/2 in the last rising event. The result is saved in the variable T2_R.

```
.meas tran T2_R trig at=TD2
targ v(q0) val='vdd/2' td=5ns rise=last
```

The Tcl scripting equivalent is as follows:

```
wfc { cr1_q0_ = crossing(wf("<meas/TRAN>V(Q0)"), x_start = greater(td2_,
5e-9), ylevel = vdd/2, slopetrigger = "rising", option = "wf") }
wfc { last_ = size(cr1_q0_) - 1 }
set T2_R [ wfc { wftodata(cr1_q0_)[last_][0] - td2_ } ]
```

ASCII File Output

You can output measurements to an ASCII file using a Tcl command.

In the final section of the example Tcl code, the results of all measurements (TD1, TD2, T2R, and T2_C2) are output to an ASCII file, *meas.out*.

```
# results are in nanoseconds
set unit {N}
set factor 1e9
set fileout [ open meas.out w+ ]
puts $fileout "\nExtracted from EZwave Post Processing"
puts $fileout [format " TD1\t= %8.4f%s" [ expr $TD1 * $factor ] $unit]
puts $fileout [format " T1\t= %8.4f%s" [ expr $T1 * $factor ] $unit]
puts $fileout [format " TD2\t= %8.4f%s" [ expr $TD2 * $factor ] $unit]
puts $fileout [format " T2\t= %8.4f%s" [ expr $T2 * $factor ] $unit]
puts $fileout [format " T2_C2\t= %8.4f%s" [ expr $T2_C2 * $factor ] $unit]
puts $fileout [format " T2_R\t= %8.4f%s" [ expr $T2_R *$factor ] $unit]
close $fileout
```

Tcl User-Defined Functions

It is efficient to use Tcl to define a function that you use regularly.

Creating a User-Defined Tcl Function **1211**
Loading User-Defined Tcl Functions Automatically **1211**

Creating a User-Defined Tcl Function

You can create a user-defined function with Tcl, and then load it manually from the Waveform Calculator.

For example, you need to compute the group delay V(S) of a complex waveform in the Waveform Calculator with the following formula:

```
GD=drv(cphase(wf("<database>/AC/V(S)")))/360
```

Procedure

1. Create a user-defined function with Tcl as follows:

```
proc gd { wave } {
   wfc " drv(cphase($wave))/360 "
}
```

2. Save this definition in a file called *gd.tcl*.
3. In the Waveform Calculator, select **File > Open Custom Function File** to open a file browser.
4. Select the user-defined Tcl file (in this example, the *gd.tcl* file).
5. Click **Open** to load the file into the Waveform Calculator.

Results

The function is now available in User-Defined Functions in the Functions list.

Related Topics

Creating a Tcl Script From the Waveform Calculator History

Loading User-Defined Tcl Functions Automatically

Loading User-Defined Tcl Functions Automatically

You can set the EZwave viewer to automatically load all user-defined functions into the Waveform Calculator at start-up.

Procedure

1. Place all your user-defined Tcl function files in one directory.

2. In the Waveform Calculator, select **Edit** > **Options**.

   The EZwave Display Preferences Dialog Box opens, showing the Waveform Calculator General options.

3. Select **Load User Extension Files at Startup**.

4. Type the path to your user-defined functions directory in the Directory Path field. You can also click **Browse** to use the file browser to select the directory.

   Note

   The user extension files in the specified directory must have a *.tcl* extension to load.

   Tip

   You can also use the environment variable AMS_UDF_LOAD to specify a path to load extension files at startup - refer to "EZwave Environment Variables" on page 54.

5. Click **OK**.

6. Restart the EZwave waveform viewer.

Results

When you open the Waveform Calculator, your user-defined functions are automatically loaded into the User-Defined Functions tree in the **Functions** list.

Tip

If you do not want the function to appear in the Waveform Calculator tree, prefix the procedure name with two underscores "__".

Related Topics

Creating a User-Defined Tcl Function

Waveform Comparison With Tcl Examples

This section contains Tcl examples for waveform comparison.

Compare All Waveforms With Default Options 1213
Compare All Waveforms and Scalars 1213
Compare All Terminal Waveforms From Transient Analysis With Default Options . 1214
Compare All Waveforms Using a Clocked Comparison 1214
Compare Specific Waveforms With Modified Tolerances, Write a Report 1215
Compare Waveforms Using All Available Comparison Commands 1215
Delaying Reference Waveforms During Comparison 1215
Delaying Some Result Waveforms (Not All) During Comparison 1216
Compare Waveforms Using the -label Argument 1217
Compare Waveforms Using the -start and -end Arguments 1218
Export a Report of Comparison 1219

Compare All Waveforms With Default Options

This Tcl script begins by opening two databases and beginning a comparison. To start a comparison, the reference and test databases must be opened first. The next step adds all waveforms recursively.

At this stage the comparison can be run. If errors or warning occur during the comparison, a log file named “comparison.log” is created in the $MGC_TMPDIR directory (see the compare savelog command). Ending the comparison will reset the list of waveforms to be compared.

```
dataset open /my/path/to/ReferenceDatabase.wdb REF
dataset open /my/other/path/to/TestDatabase.wdb TEST

compare start REF TEST
compare add -r *
compare run
compare end
```

Compare All Waveforms and Scalars

This Tcl script begins by opening two databases and beginning a comparison. To start a comparison, the reference and test databases must be opened first. The next step adds all waveforms recursively.

At this stage the comparison can be run. First the waveforms are compared (with default options). In the final step, the scalars are compared. If there are no scalars in the database, the *compareScalarsReport.txt* report will contain only “Nothing to compare”.

```
## Compare all waveform, generic, no specific tolerances
dataset open <path/to/reference/database> REF
dataset open <path/to/test/database> TEST

compare start REF TEST
compare add -r *
compare run
compare end

## Compare all scalars in the databases
dataset scalar REF TEST -output <path/to/compareScalarsReport.txt>
```

Compare All Terminal Waveforms From Transient Analysis With Default Options

This script begins by opening two databases and beginning a comparison. To start a comparison, the reference and test databases must be opened first. The next steps add all terminal waveforms from the TRAN analysis to the comparison recursively.

At this stage the comparison can be run.

```
dataset open /my/path/to/ReferenceDatabase.wdb REF
dataset open /my/other/path/to/TestDatabase.wdb TEST

dataset open REF TEST
compare add -show TRAN.v -terminals -r *
compare run
compare end
```

Compare All Waveforms Using a Clocked Comparison

The basic steps in this script are similar to the preceding script with the addition of a clock creation. This script creates a clock named “myClock” that will trigger a comparison on both rising and falling edges of the its source waveform <REF/TRAN>:test:eoc.

When adding the waveforms to the comparison, the script specifies the clock to use during the computation of the comparison waveforms.

```
dataset open /my/path/to/ReferenceDatabase.wdb REF
dataset open /my/other/path/to/TestDatabase.wdb TEST

compare start REF TEST
compare clock -both myClock <REF/TRAN>:test:eoc
compare add -clock myClock -r *
compare run
compare end
```

Compare Specific Waveforms With Modified Tolerances, Write a Report

This script modifies the default tolerances for custom precision comparison of waveforms.

First, for analog waveforms, the script specifies a tolerance of 0.5% for both x and -axes. Then, for digital waveforms, the leading tolerance is set to 0.1 and trailing tolerance is set to 0.2µ. The script only adds waveforms named "tvin" (recursively, as indicated by **-r**) and then runs the comparison. Finally the script exports the comparison report (waveform by waveform differences) to the text file "report.txt". Running the comparison and exporting the report are two independent steps: you can write a report without running the comparison beforehand; the exporting process will run the comparison for you.

```
dataset open /my/path/to/ReferenceDatabase.wdb REF
dataset open /my/other/path/to/TestDatabase.wdb TEST

compare start REF TEST
compare options -xTol 0.5% -yTol 0.5% -tolLead 0.1e-6 -tolTrail 0.2e-6
compare add -r tvin
compare run
compare info -write /path/to/the/report.txt
compare end
```

Compare Waveforms Using All Available Comparison Commands

This example provides a single Tcl script that uses most available commands within it.

The main difference here is that after adding the waveforms named "tvin" to the comparison, we reconfigure them to use a clocked comparison.

```
compare start REF TEST
compare options -xTol 0.5% -yTol 0.5% -noaddwave -maxsignal 50
compare clock -both myClock <REF/TRAN>:test:eoc
compare add tvin
compare configure -clock myClock tvin
compare run
compare info -write /path/to/the/report.txt
compare end
```

Delaying Reference Waveforms During Comparison

This example Tcl script begins the comparison with a delayed reference waveform.

In this case, the delay alignment is 5n for all the waveforms in the reference dataset.

```
# Open reference and test databases
dataset open ref.wdb REF
dataset open test.wdb TEST

# Start the comparison session and "align" waveforms with a delayed
# reference waveform.
compare start -refdelay 5n  REF TEST

# Add all waveforms to be compared.
compare add -show all -r *

# Run the comparison.
compare run

# Save an error log containing any warnings and errors encountered during
# the comparison.
compare savelog log.txt

# Export a report of this comparison to the report.txt file.
compare info -write report.txt

# End the comparison session.
compare end
```

Delaying Some Result Waveforms (Not All) During Comparison

This Tcl script example aligns the comparison using a 3ns delay for all voltage waveforms in the test dataset.

The VAR waveform comparisons are not aligned with any delay.

```
# Open reference and test databases
dataset open ref.wdb REF
dataset open test.wdb TEST

# Start the comparison session
compare start REF TEST

# Add all voltage waveforms to be compared, use test delay to "align"
# those waveforms for the comparison
compare add -show TRAN.V -r -testdelay 3n *

# Add all VAR waveforms to the comparison, these waveforms won't be
# "aligned".
compare add -show TRAN.VAR -r *

# Run the comparison
compare run

# Save an error log containing any warnings and errors encountered during
# the comparison.
compare savelog log.txt

# Export a report of this comparison to the report.txt file
compare info -write report.txt

# End the comparison session
compare end
```

Compare Waveforms Using the -label Argument

This Tcl script example uses labels to distinguish three comparisons at different intervals along the same waveform.

```
# Open reference and test databases
dataset open Bumpy.wdb REF
dataset open Bumpy_test.wdb TEST

# Start the comparison session
compare start REF TEST

# Here we use a label to compare the same waveforms on different
# intervals. The label will appear in the comparison waveform name to ease
# user recognition of each waveform. Labels avoids name collision and
# waveform overwriting.
# Below are three examples using labels.

# Add all waveforms to the comparison using the label "firstInterval".
# Comparison will be performed between 0 and 1.2n and the reference
# waveform will be shifted 0.3n.
compare add -r -label firstInterval -start 0n -end 1.2n -refdelay .3n *

# Add all waveforms to the comparison using the label "secondInterval".
# Comparison will be performed between 1n and 1.5n and the test waveforms
# will be shifted 0.5n.
compare add -r -label secondInterval -start 1n -end 1.5n -testdelay .5n *

# Add all waveforms to the comparison using the label "thirdInterval".
# Comparison will be performed between 0.75n and 2.4n and the reference
# and test waveforms will be shifted respectively by 0.3n and 0.5n.
compare add -r -label thirdInterval -start .75n -end 2.4n -refdelay .3n
-testdelay .5n *
```

Compare Waveforms Using the -start and -end Arguments

This Tcl script example compares voltage waveforms between 50ns and 100ns. VAR waveforms are compared along their whole domain of definition.

```
# Open reference and test databases.
dataset open ref.wdb REF
dataset open test.wdb TEST

# Start the comparison session.
compare start REF TEST

# Add all voltage waveforms to be compared between 50n and 100n.
compare add -show TRAN.V -r -start 50n -end 100n *

# Add all VAR waveforms to be compared on all their domain of definition.
compare add -show TRAN.VAR -r *

# Run the comparison.
compare run

# Save an error log containing any warnings and errors encountered during
# the comparison.
compare savelog log.txt

# Export a report of this comparison to the report.txt file.
compare info -write report.txt

# End the comparison session.
compare end
```

Export a Report of Comparison

This Tcl script example exports a report of the comparison to a text file and demonstrates the precedence of the compare add command.

```
# Open reference and test databases.
dataset open ref.wdb REF
dataset open test.wdb TEST

# Start the comparison session.
compare start REF TEST

# Add all voltage waveforms to be compared between 50n and 100n. The
# start and end time specified here overrules the ones from the compare
# run command.
compare add -show TRAN.V -r -start 50n -end 100n *

# Add all VAR waveforms to be compared on all their domain of definition.
compare add -show TRAN.VAR -r *

# Run the comparison.
compare run

# Save an error log containing any warnings and errors encountered during
# the comparison.
compare savelog log.txt

# Export a report of this comparison to the report.txt file. The
# starting and ending times will only apply to VAR waveforms as those
# times are overruled for voltage waveforms by the one specified with
# the compare add commands.
compare info -start 0 -end 75n -write report.txt

# End the comparison session
compare end
```

Using find Commands

This example script illustrates the use of the different find commands available in the EZwave viewer.

The final two examples compare the use of the legacy switch “-i” and the use of “-show TRAN.i” syntax for finding currents.

The script will search for waveforms with different criteria (name, kind, mode, and so on) and plot each matching waveform in separate windows. The windows are then sorted for better viewing or printing.

```
# Close all existing windows.
wave closewindow -all

# Create a new window.
wave addwindow -title "Analogs"
# Look for all analog waveforms recursively in the most recently opened
# (or active) database.
set search [find analogs -r *]
# Plot all matching waveforms stacked in the most recently opened window.
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

# Create a new window.
wave addwindow -title "Analogs"
# Look for all analog waveforms recursively in a database named foobar
# even if foobar is not the most recently opened database
set search [find analogs -r foobar/*]
# Plot all matching waveforms stacked in the most recently opened window.
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

# Create a new window.
wave addwindow -title "Analogs 2"
# Look for all terminal waveforms.
set search [find signals -r -terminals *]
# Plot all matching waveforms.
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

## Create an new window
wave addwindow -title "Analogs 3"
## Look for all variables
set search [find signals-show TRAN.var -r *]
## Plot all matching waveforms
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

## Create an new window
wave addwindow -title "AC Analysis"
## Look for all phases
set search [find signals -show AC.vp -r *]
## Plot all matching waveforms
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
```

```
}

# Create a new window.
wave addwindow -title "Digitals"
# Look for all analog waveforms recursively in the most recently opened
# (or active) database
set search [find digitals -r *]
# Plot all matching waveforms stacked in the most recently opened window.
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

# Create a new window.
wave addwindow -title "Digitals 2"
# Look for all signal waveforms.
set search [find nets -r -signals *]
# Plot all matching waveforms.
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

# Create a new window.
wave addwindow -title "Digitals 3"
# Look for all signal waveforms recursively but stop the recursion after
# two levels.
set search [find nets -r -signals -depth 2 *]
# Plot all matching waveforms.
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

# Create a new window.
wave addwindow -title "Digitals 4"
# Look for all signal waveforms recursively whose third element in its
# design path is 'x1'.
set search [find nets -r -signals *:*:x1:*]
# Plot all matching waveforms.
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

# Create a new window.
wave addwindow -title "Out Signals"
# Look for all output signal waveforms recursively.
set search [find signals -r -out *]
# Plot all matching waveforms
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
```

```
}

##
## There are various ways to retrieve currents from a database in TCL.
## Using the legacy switch "-i", using the command find currents, and
## using "-show TRAN.i" syntax.
##
## If more than TRAN analysis is needed,
## you can use -show AC.i or even -show ALL.i.
##

## Create an new window
wave addwindow -title "Current 1"
## Look for all currents using -i switch
set search [find signals -r -i *]
## Plot all matching waveforms
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

## Create a new window
wave addwindow -title "Current 2"
## Look for all currents using the find currents command
set search [find currents -r *]
## Plot all matching waveforms
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}

## Create a new window
wave addwindow -title "Current 3"
## Look for all currents with an appropriate -show option
set search [find signals -show TRAN.i -r *]
## Plot all matching waveforms
foreach list $search {
    foreach wave $list {
        add wave $wave
    }
}
```

Waveform Calculator Example Tcl Scripts

Some of these example waveform calculator Tcl scripts provide equivalent functionality to legacy AFS WaveCrave CalcPAD scripts. The scripts appear in the Waveform Calculator Functions list, under User-Defined Functions.

The examples of waveform calculator Tcl scripts are provided in the EZwave examples folder.

- In the AMS tree, navigate to:

`$MGC_AMS_HOME/examples/ezwave`

- In the AFS tree, navigate to:

`$AFS_ROOT/examples/ezwave`

Tip

You can control whether these scripts are loaded automatically into the Waveform Calculator when EZwave starts up. Refer to "Waveform Calculator General Options" on page 547.

To watch a tutorial that shows you how to use the Waveform Calculator Tcl scripts to calculate phase noise for a PLL, see the video:

The following scripts, when enabled, appear in the Waveform Calculator **Functions** list, under User-Defined Functions:

adc_sndr 1226
adc_sndr_parallel4 1227
clock_jitter 1228
compressvri 1229
IPnVRI 1230
pll_jitter 1233
pll_jitter_parallel4 1235
pll_phasenoise 1236
pll_phasenoise_parallel4 1238

plotcompress **1240**
plotpacipn **1242**

adc_sndr

Waveform Calculator function.

Calculate the Signal to Noise plus Distortion Ratio (SNDR) and the Signal to Noise Ratio (SNR) for a transient noise simulation result for ADCs. The script makes the calculations for a single run.

Tip

See also script adc_sndr_parallel4, which makes the calculations for four parallel runs.

Usage

adc_sndr(*ADCout*, *Fclock*, *Fsignal*)

Parameters

- ***ADCout***

  (Required) A sigma-delta ADC time-domain output waveform.

- ***Fclock***

  (Required) The clock frequency.

- ***Fsignal***

  (Required) The signal frequency.

Return Values

The outputs are as follows:

- Values for SNDR and SNR.
- A power spectral density plot.

adc_sndr_parallel4

Waveform Calculator function.

Calculate the Signal to Noise plus Distortion Ratio (SNDR) and the Signal to Noise Ratio (SNR) for a transient noise simulation result for ADCs. The script makes the calculations for four parallel runs.

Tip

See also script adc_sndr, which makes the calculations for a single run.

Usage

adc_sndr_parallel4(***ADCout***, ***ADCout2***, ***ADCout3***, ***ADCout4***, ***Fclock***, ***Fsignal***)

Parameters

- ***ADCout***

  (Required) First sigma-delta ADC time-domain output waveform.

- ***ADCout2***

  (Required) Second sigma-delta ADC time-domain output waveform.

- ***ADCout3***

  (Required) Third sigma-delta ADC time-domain output waveform.

- ***ADCout4***

  (Required) Fourth sigma-delta ADC time-domain output waveform.

- ***Fclock***

  (Required) The clock frequency.

- ***Fsignal***

  (Required) The signal frequency.

Return Values

The outputs are as follows:

- Values for SNDR and SNR
- A power spectral density plot.

clock_jitter

Waveform Calculator function.

Calculates jitter statistics for clock waveforms.

Usage

clock_jitter(***CLKout***[, *Tstart*, *Threshold*, *Period_Accurate*, *NumberOfAccumulatedCycles*])

Parameters

- ***CLKout***

  (Required) A time-domain clock waveform.

- *Tstart*

  (Optional) The starting time for the jitter calculation. Default is the first point of CLKout.

- *Threshold*

  (Optional) The value used to calculate periods. Default is the mean of CLKout on the observed time domain.

- *Period_Accurate*

  (Optional) A specific period for improved accuracy in absolute jitter. Default is the mean of the periods.

- *NumberOfAccumulatedCycles*

  (Optional) Used for the absolute jitter calculation. Default is 10.

Return Values

- Mean, minimum, and maximum period.
- Peak-to-peak and RMS period jitter.
- RMS cycle-to-cycle, absolute, and long-term period jitter.
- A frequency versus time plot.
- A jitter histogram.

compressvri

Waveform Calculator function.

Computes the N^{th} compression point for the specified harmonic.

Usage

compressvri(*wf*, *harmonicindex*, [*nDb*, *ePin*, *rload*, *sweepParam*, *measType*])

Arguments

- ***wf***

  (Required) Specifies the input compound waveform name.

- ***harmonicindex***

  (Required) The harmonic index to be used.

- *nDb*

  (Optional) Specifies the delta between the waveform and the ideal slope at the compression point. Default is 1.

- *ePin*

  (Optional) Specifies the input extrapolation point. Default is “Automatic”, the first point in the input power sweep.

- *rload*

  (Optional) Specifies the internal resistance of the source. If specified, this is used to convert the voltage to dBm. Default is 50.

- *sweepParam*

  (Optional) Specifies the sweep parameter used in the input power sweep. Typically, this is in dBm. Default is “Automatic”.

- *measType*

  (Optional) Specifies the choice of returned result, “Input”, “Output” or “Both”. Default is “Both”.

Return Values

Double. The N^{th} compression point for the specified harmonic.

Related Topics

wave compressvri

compresscompound

IPnVRI

Waveform Calculator RF function.

Computes the N^{th} order input-referred intercept point (IIP), output-referred intercept point (OIP), or both for the specified compound waveform.

Usage

IPnVRI(***wf***, [*hb_refHarmonic*, *hb_spurHarmonic*, *pss_refIndex*, *pss_spurIndex*, *inputPower*, *sweepParam*, *nOrder*, *ePin*, *rload*, *measType*])

Arguments

- ***wf***

  (Required) Specifies the input compound waveform name.

- *hb_refHarmonic*

  (Optional) An array of integers (comma-separated list of integers, in square brackets). Specifies the reference harmonic indexes associated with each tone. The order and the number of indexes must match the database data. Used for HB analysis. Default is "None".

- *hb_spurHarmonic*

  (Optional) An array of integers (comma-separated list of integers, in square brackets). Specifies the spurious harmonic indexes associated with each tone. The order and the number of indexes must match the database data. Used for HB analysis. Default is "None".

- *pss_refIndex*

  (Optional) Specifies the index of the reference frequency. Used for PSS analysis. Default is "None".

- *pss_spurIndex*

  (Optional) Specifies the index of the spurious frequency. Used for PSS analysis. Default is "None".

- *inputPower*

  (Optional) Specifies the input power. Used for single point IPN. Default is "None".

- *sweepParam*

  (Optional) Specifies the sweep parameter used in the input power sweep. Typically, this is in dBm. Default is "Automatic".

- *nOrder*

  (Optional) Specifies the N order of the spurious frequency. Default is 3.

- *ePin*

  (Optional) Specifies the input power at which to calculate the intercept point. Default is "Automatic", the first point in the input power sweep.

- *rload*

  (Optional) Specifies the load resistance. If specified, this is used to convert the voltage to dBm. Default is 50.

- *measType*

  (Optional) Specifies the choice of returned result, "IIP", "OIP" or "Both". Default is "Both".

Return Values

Double or List. The input-referred intercept point, the output-referred intercept point, or both values.

Description

Specifying Frequency Indexes for PSS

For 0Hz, the index=0. For the fundamental frequency (fund), the index=1, where fund is the frequency that is co-periodic and divides evenly into all the other frequencies.

Use the following steps to determine the ipn reference (ref) and spurious (spur) frequencies with their corresponding indexes:

1. Determine the ref frequency from the design.
2. Calculate the spur frequency in the normal manner. For example:

   spur_freq=(n-1)*f1-f2 or (n-1)*f2-f1

   where n=order (and image frequencies if needed).
3. Calculate the ipn indexes by dividing the ref and spur frequencies by fund. For example:

   If f1=2.45GHz and f2=2.65GHz then fund=50MHz, and

   pss_refIndex = 2.45GHz/50MHz = 49 and

   pss_spurIndex = 2.65GHz/50MHz = 53.

Specifying Frequency Indexes for HB

Unlike PSS, indexes are really multiples of each frequency. Then all products are summed to determine the desired reference and spurious frequencies. For example:

If f1=2.45GHz (ref, [reference harmonic]) and f2=2.65GHz (spur, [spurious harmonic]),

enter [0 1] in the *hb_spurHarmonic* field for the spur index at 2.65Hz, and

enter [1 0] in the *hb_refHarmonic* field for the ref index at 2.45GHz.

If there are more than 2 tones (freq), add each additional tone (freq) in the same way.

Related Topics

ipn

wave ipnvri

pll_jitter

Waveform Calculator function.

Calculates jitter measurements for PLLs for a single run and decomposes jitter into deterministic and random components and reports these results.

Tip

See also script pll_jitter_parallel4, which makes the calculations for four parallel runs.

Usage

pll_jitter(***PLLout***[, *Tstart*, *Threshold*, *Period_Accurate*])

Parameters

- ***PLLout***

  (Required) A PLL time-domain output waveform.

- *Tstart*

  (Optional) The starting time for the jitter calculation. Default is the first point of PLLout.

- *Threshold*

  (Optional) The value used to calculate periods. Default is the mean of PLLout on the observed time domain.

- *Period_Accurate*

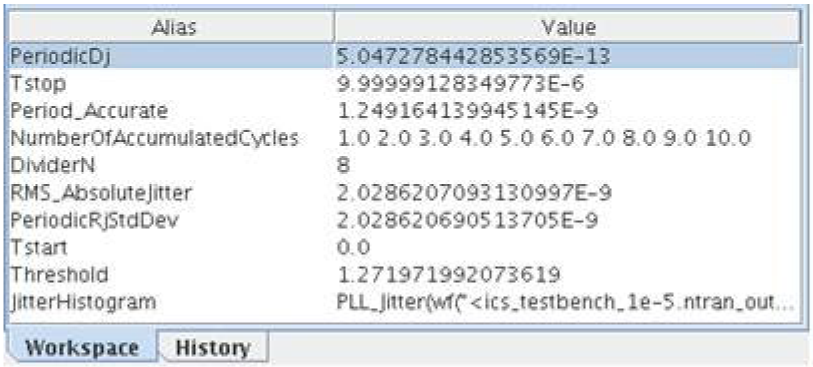
  (Optional) A specific period for improved accuracy in absolute jitter. Default is the mean of the periods.

- *divN*

  (Optional) The number of samples used to decompose jitter into deterministic and random components. By default, jitter is not decomposed.

Return Values

- Absolute jitter.
- RMS period jitter and cycle-to-cycle jitter versus the number of accumulated cycles.
- A jitter histogram.
- Random component of absolute jitter: periodicRjStdDev.
- Deterministic component of absolute jitter: periodicDj.



Alias	Value
PeriodicDj	5.047278442853569E-13
Tstop	9.99999128349773E-6
Period_Accurate	1.249164139945145E-9
NumberOfAccumulatedCycles	1.0 2.0 3.0 4.0 5.0 6.0 7.0 8.0 9.0 10.0
DividerN	8
RMS_AbsoluteJitter	2.0286207093130997E-9
PeriodicRjStdDev	2.028620690513705E-9
Tstart	0.0
Threshold	1.271971992073619
JitterHistogram	PLL_Jitter(wf("<ics_testbench_1e-5.ntran_out...

pll_jitter_parallel4

Waveform Calculator function.

Calculates jitter measurements. The script makes the calculations for four parallel runs.

Tip

See also script pll_jitter, which makes the calculations for a single run.

Usage

pll_jitter_parallel4(*PLLout*, *PLLout2*, *PLLout3*, *PLLout4*[, *Tstart*, *Threshold*, *Period_Accurate*])

Parameters

- ***PLLout***

  (Required) First PLL time-domain output waveform.

- ***PLLout2***

  (Required) Second PLL time-domain output waveform.

- ***PLLout3***

  (Required) Third PLL time-domain output waveform.

- ***PLLout4***

  (Required) Fourth PLL time-domain output waveform.

- *Tstart*

  (Optional) The starting time for the jitter calculation. Default is the first point of PLLout.

- *Threshold*

  (Optional) The value used to calculate periods. Default is the mean of PLLout on the observed time domain.

- *Period_Accurate*

  (Optional) A specific period for improved accuracy in absolute jitter. Default is the mean of the periods.

Return Values

- Absolute jitter.
- RMS period jitter and cycle-to-cycle jitter versus the number of accumulated cycles.
- A jitter histogram.

pll_phasenoise

Waveform Calculator function.

Calculates phase noise measurements from transient noise simulation results for PLLs for a single run.

Video

To watch a tutorial that shows you how to use this Tcl script to calculate phase noise for a PLL, see the video:

Tip

See also script pll_phasenoise_parallel4, which makes the calculations for four parallel runs.

Usage

pll_phasenoise(***PLLout***, ***Fund_Accurate***[, *Tstart*, *SpurFreq*])

Parameters

- ***PLLout***

  (Required) A PLL time-domain output waveform.

- ***Fund_Accurate***

  (Required) The accurate fundamental frequency.

- *Tstart*

  (Optional) The starting time for the phase noise calculation. Default is the first point of PLLout.

- *SpurFreq*

  (Optional) Specify a spur frequency to measure a spur. By default, spurs are not measured. Integer-N PLLs have periodic spurs at the reference frequency.

Return Values

- A frequency versus time plot.
- A phase noise plot.

The following variables are set, and displayed in the Waveform Calculator Workspace tab:

Table C-13. pllphasenoise Output Variables

Alias	Description
Tstop	Sampling stop time (s)
FundAverage	Average fundamental frequency (Hz)
PhaseNoise_upperCL	Phase noise upper confidence limit
PHRange	Confidence limit range
RBW	Frequency resolution bandwidth (Hz)
PhaseNoise_lowerCL	Phase noise lower confidence limit
Ksegments	Calculated as RBW * (Tstop - Tstart)
posX	X value for annotation text
ConfidenceLevel	Phase noise confidence level (0-1)
posY	Y value for annotation text
MCP	Number of multiple parallel simulations
Threshold	Mean signal amplitude (V)

Tip

For more information, right-click on the PLL_PhaseNoise function in the Waveform Calculator **Functions** tab and choose **Edit** to view the code used in the function.

pll_phasenoise_parallel4

Waveform Calculator function.

Calculates phase noise measurements from transient noise simulation results for PLLs. The script makes the calculations for four parallel runs.

Video

To watch a tutorial that shows you how to use this Tcl script to calculate phase noise for a PLL, see the video:

Tip

See also script pll_phasenoise, which makes the calculations for a single run.

Usage

pll_phasenoise_parallel4(*PLLout*, *PLLout2*, *PLLout3*, *PLLout4*, *Fund_Accurate*[, *Tstart*, *SpurFreq*])

Parameters

- ***PLLout***

  (Required) First PLL time-domain output waveform.

- ***PLLout2***

  (Required) Second PLL time-domain output waveform.

- ***PLLout3***

  (Required) Third PLL time-domain output waveform.

- ***PLLout4***

  (Required) Fourth PLL time-domain output waveform.

- ***Fund_Accurate***

  (Required) The accurate fundamental frequency.

- *Tstart*

  (Optional) The starting time for the phase noise calculation. Default is the first point of PLLout.

- *SpurFreq*

  (Optional) Specify a spur frequency to measure a spur. By default, spurs are not measured.

Return Values

- A frequency versus time plot.
- A phase noise plot.

plotcompress

Waveform Calculator function.

Plots output power against input power in dBm and returns n-dB compression point. Only supports compound waveforms.

Usage

plotcompress(***src_p***)

plotcompress(***src_p***, *src_n*)

plotcompress(***src_p***, *src_n*, *sweep_param*)

plotcompress(***src_p***, *src_n*, *sweep_param*, *fund*, *ndb*, *epin*, *rload*, *meas_type*)

Arguments

- ***src_p***

  (Required) Specifies a compound waveform of voltage spectrum signals within an input power sweep.

- *src_n*

  (Optional) Specifies a compound waveform of voltage spectrum signals within an input power sweep, or “None” (equals 0.0). Default is “None”.

- *sweep_param*

  (Optional) Specifies the sweep parameter. If not specified, the first sweep parameter found (except the “harmonic one” is used. Default is “Automatic”.

- *fund*

  (Optional) Specifies the fundamental or harmonic frequency (default nth(indep(src_p), 2)). Default is “Automatic”.

- *ndb*

  (Optional) Specifies the number of dB by which the actual curve falls below an ideal linear curve. Use ndb = 1 to extract the 1-db compression point. Default is 1.

- *epin*

  (Optional) Specifies the extrapolation point. Default is the first point in the input power sweep. Default is “Automatic”.

- *rload*

  (Optional) Specifies the load resistance. Default is 50.

- *meas_type*

  (Optional) Specifies the measurement type. Allowed values are “X”|“Y”|“Both”. “X” corresponds to the input-referred compression. “Y” corresponds to the output-referred compression. Default is “Both”.

Related Topics

wave plotcompress

plotpacipn

plotpacipn

Waveform Calculator function.

Takes a PAC spectrum signal pair at the desired and unwanted harmonics, within an input power sweep, plots the pair of IPn curves and returns the intercept point. Only supports compound waveforms.

Usage

plotpacipn(***src_p***, ***desired_harm_p***, ***unwanted_harm_p***)

plotpacipn(***src_p***, ***desired_harm_p***, ***unwanted_harm_p***, *src_n*, *desired_harm_n*, *unwanted_harm_n*, *sweep_param*)

plotpacipn(***src_p***, ***desired_harm_p***, ***unwanted_harm_p***, *src_n*, *desired_harm_n*, *unwanted_harm_n*, *sweep_param*, *norder*, *rload*)

Arguments

- ***src_p***

  (Required) Specifies a compound waveform with PAC scalar signals within an input power sweep.

- ***desired_harm_p***

  (Required) Specifies the desired harmonic. Default is -1.

- ***unwanted_harm_p***

  (Required) Specifies the unwanted harmonic. Default is -1.

- *src_n*

  (Optional) Specifies a compound waveform with PAC scalar signals within an input power sweep. Default is -1.

- *desired_harm_n*

  (Optional) Specifies the desired harmonic, or 0.0 (default).

- *unwanted_harm_n*

  (Optional) Specifies the unwanted harmonic, or 0.0 (default).

- *sweep_param*

  (Optional) Specifies the sweep parameter. If not specified, the first sweep parameter found is used. Default is -1.

- *norder*

  (Optional) Specifies the order of the intercept point (3 by default).

- *rload*

  (Optional) Specifies the load resistance (50 by default).

Related Topics

wave plotpacipn

plotcompress

Appendix D
Supported Net Representation Components

This section describes the supported net *representation* syntax and values for the wave show Tcl command and the **-show** option of the add wave Tcl command.

Representing the Signal as a Waveform 1246
Extended Options for Selecting Waveforms 1246
representation 1248
Examples of wave show and -show usage 1251

Representing the Signal as a Waveform

The **wave show** Tcl command and the **-show** option of the **add wave** Tcl command specify how the signal is to be represented as a waveform.

The wave show representation is used as the default representation when the add wave command is issued without the **-show** option. If **-show** is specified, it is applied beginning where it appears in the script until another **-show** is issued, or until the end of the script.

If no wave show or add wave -show commands are issued, the default representation is as follows:

```
tran.v|ac.vdb|ac.vp
```

The syntax for the **wave show** command and **-show** option is as follows:

```
wave show representation [ |representation2|...|representationN]
```

```
add wave -show representation [|representation2|...|representationN]
```

The reserved keywords, **all** and **none**, can be used as follows:

- -show all

  Requests all objects found in all analyses

- -show *analysis*.all

  Requests all objects found in the specified analysis.

- -show none

  Requests no objects in the case where there is no analysis folder.

Extended Options for Selecting Waveforms **1246**
representation. **1248**
analysis 1248
discipline. 1248
physic 1250
Examples of wave show and -show usage **1251**

Extended Options for Selecting Waveforms

The **-show** option is available with a number of commands that access waveforms.

These are listed in the following table:.

Table D-1. Commands that Access Waveforms

compare add	dataset savewaveforms	find nets | signals	wave addproperty
compare clock	delete wave	wave addannotation	wave exists
compare configure	find analogs	wave adddeltamarker	wf
dataset mergewaveforms	find digitals	wave addline	

For these commands, along with the **add wave** command, there are a number of extended options available for use with the **-show** option. These options are used for selecting the waveform(s) required. The following extended options are available. See "Selecting Waveforms in Tcl" on page 1014 for details:

```
[-recursive] [-depth <level>] [-signals] [-quantities]
[-separator <separator>] [-terminals]
[-nets] [-ports] [-in] [-out] [-inout] [-internal]
[-through] [-across] [-free] [-flow] [-i]
[-boundary] [-a2d] [-d2a] [-bidir] [-run {run_number | run_name}]
[-session {previous | current}] [-adms | -modelsim]
```

representation

A *representation* is a complete representation of a net and is composed of an *analysis*, a *discipline*, and a *physic*.

A dot (**.**) separates the *analysis* portion from the *discipline* and *physic* portion. There is no separator between the *discipline* and the *physic*, as follows:

analysis.disciplinephysic

If more than one *representation* is specified, use a pipe (**|**) to separate them:

analysis.disciplinephysic|analysis.disciplinephysic|analysis.disciplinephysic

analysis 1248
discipline 1248
physic. 1250

analysis

The *analysis* portion of the *representation* specifies the kind of analysis to use to “wave” a net.

For example, in the same simulation, we can request an AC and a TRAN analysis. And in this case, we may want to see the waveform of one net for the AC analysis and of another net for the TRAN analysis. Supported analyses are shown in the following table. For details on the values in the table, refer to the Eldo manual.

Table D-2. Supported analysis Values

AC	DC	DSP	EXTRACT
LSTB	MEAS	MODSST	NOISE
NOISETRAN	OP	OPFOUR	SST
SSTAC	SSTNOISE	SSTXF	TRAN

discipline

The *discipline* portion of the representation specifies a discipline of an analog net.

For example, current or voltage. Supported disciplines are shown in the following table. For details on the values in the table, refer to the Eldo manual.

Table D-3. Supported discipline Values

A	E	IFNCELL	POW	TABLE
ACC_STRESS	FLKNOISE	IG	POWDYN	TEMP

Table D-3. Supported discipline Values (cont.)

AVG	FLUX	IN	POWSTAT	TGP
B	FLOUR	INX	Q	TGP_
B_OPT	FT	IOUT	QB	THNOISE
BETAAC	FUND_OSC	IPIN	QBD	V
BETADC	G	IPROBE	QBS	VALDIP
BFACTOR	G_OPT	IS	QC	VB
BOPT	GA	ISTAT	QCELL	VBC
C	GA_	ISUB	QD	VBD
CBB	GAC	IW	QE	VBE
CBD	GAC_	IX	QG	VBS
CBDJ	GAM	KFACTOR	QS	VC
CBG	GAM_	LSC	RBB	VCE
CBS	GAMMA_OPT	LSC_	RBNOISE	VCS
CBSX	GAMMA_OPT_	LSTB	RCNOISE	VD
CBX	GASM	LSTB_	RDNOISE	VDIP
CCS	GASM_	LT_JITTER	RENOISE	VDS
CD	GAUM	LV	REP	VDSE
CDD	GAUM_	LX	RGNOISE	VDSS
CDG	GDS	MUFACTOR	RMU	VE
CDF	GMB	MUFACTOR_L	RNEQ	VES
CDS	GMIBD	MUFACTOR_S	RO	VG
CGB	GMIBS	MV	RPI	VGB
CGBO	GOPT	MVBD	RSNOISE	VGS
CGDO	GP_	MVBS	RX	VGSE
CGG	GPC	MVDS	S	VNEG
CGS	GPC_	MVGB	SG	VPOS
CGSO	H	MVGS	SSC	VS
CMU	I	N	SSC_	VT
CPI	IB	NC	SST	VTCELL
CSB	IBDNOISE	NC_	SSTINOISE	VTH
CSD	IBNOISE	NET_POLE	SSTINOISE.H	VTH_D

Table D-3. Supported discipline Values (cont.)

CSG	IBS	NET_POLE_	SSTNOISE	VX
CSS	IBSNOISE	NFMIN	SSTNOISE.H	VXN
CXS	IC	NFMIN_	SSTONOISE	W
DATA	ICNOISE	NOISE	SSTONOISE.H	WOPT
DATA_CTE	ID	OPMODE	SSTSNF	XF
DATA_LIN	IDNOISE	P	STD	Y
DBG	IDS	PDB	STRESS	YOPT
DSP	IE	PHI_OPT	T	Z

physic

The *physic* portion of the *representation* specifies the physical representation of the waveform.

For example, for a frequency signal that is complex, you can choose to see its magnitude in DB, its phase, or just the real portion. It may also include, for example, the harmonic number of a large signal frequency analysis.

The *physic* portion is composed of a *primary_physic* and an optional *secondary_physic*, separated by a dot (**.**).

The primary_physic

The supported *primary_physic* values are shown in the following table. For details of the values in the table, refer to the Eldo manual.

Table D-4. Supported primary_physic Values

D	GD	P	SOL
DB	I	PRED	T
DIG	M	R	
DSOL	MAG	RAD	

The secondary_physic

The *secondary_physic* is in the form of:

h(*index*)

where *index* is an integer literal. If more than one index value is appropriate, separate the index values by commas (**,**).

Examples of wave show and -show usage

Provides examples of **wave show** and **-show** usage.

- wave show ac.vdb

  In this example, the *analysis* is AC, the *discipline* is V, and the *physic* is DB.

- add wave -show ffst.vdb.h(1) :top:net1

  In this example, the *analysis* is FFST, the *discipline* is V, and the *physic* is DB for the *primary_ physic*, and H(1) for the *secondary_physic*. This will search for the specified waveform in the last-opened database: *database_name/FSST>VDB(:top:net1).H(1)*

Glossary

active

The condition when an area of the application window is able to accept data. For example, selecting a graph window makes it *active* to accept waveforms. A plot operation will then display the waveform in the active graph window. When the **File** > **New** menu item is selected, the new window created is automatically the active graph window.

active cursor

The cursor shown using a thick line. Clicking on any cursor will automatically make it the active cursor.

active window

The window where waveforms are plotted when not using drag and drop. This is also the window used for main menu and toolbar commands.

admittance parameters (Y-Parameters)

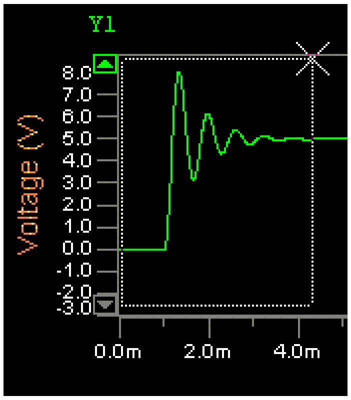
Admittance (y-parameter) is the ease with which an alternating current flows through an electronic circuit or system. For any given circuit element, the admittance is the reciprocal of the impedance (1/Z).

analog waveform

An analog waveform may contain such domains as frequency, voltage and current. Each point on the domain axis corresponds to a value on the waveform.

area zoom

The action when zooming in both the X and Y directions. Drag the mouse over the waveform itself to define the rectangular region to zoom to.



background

Within the viewer, refers to the base color to be used when printing.

base Y-level line

If more than one *Y-level line* is associated with a cursor, a “base” Y-level line is assigned. All other Y-level lines have a delta-Y value calculated from this base line. By default, the waveform whose name appears at the top of the list in the graph window is the base line. To change the base line to another Y-level line, right-click the marker of the desired base line and select "Base" Y-Level Line.

baseline

Baseline is the magnitude reference line at the base magnitude, which is the magnitude of the portion of a pulse waveform that represents the first nominal state of a pulse (usually referred as LOW level).

clipboard

The application's internal storage area. This differs from your computer clipboard area.

clipboard (EZwave)

The internal utility that enables you to move objects (such as waveforms, text and other objects) from one location to another within the viewer using familiar cut, copy and paste operations. The utility stores the information temporarily for exchange between graph windows, workspaces and other areas of the viewer.

Clipboard (Windows)

The Windows Clipboard utility enables you to exchange contents within an application or between shared applications using familiar cut, copy and paste operations. The utility stores the information temporarily for exchange between shared and local applications.

complex waveform

A complex waveform can be defined as any sound wave which is not sinusoidal. By the theorem of Fourier, any complex periodic waveform can be decomposed into a series of simple sinusoids that differ in the three defining attributes of amplitude, frequency, and phase.

compound waveform

A compound waveform can be defined as a waveform that contains the results of several simulations for the same node. This product can perform operations on either the compound waveform or the individual elements that make up the compound waveform.

cursor

A special on-screen indicator, such as a vertical line, drawn in the waveform display area to identify locations or create a point for measurement. The first cursor created is known as the base (reference) cursor.

dB

A transformation setting for complex waveforms that shows the magnitude of each point of the complex waveform calculated in decibels (20 * log (|waveform|))

enumerated type

In VHDL, an enumeration type declaration defines a type that has a set of user-defined values consisting of identifiers and character literals. If a waveform is displayed in an enumerated format, text values are displayed in a box rather than graphical high/low waveform.

export

Within the viewer, the ability to save the active graph window as a .jpg file.

femtosecond (fs)

1/1000 picosecond.

fit row heights

Within the viewer, the ability to change the row heights in the active graph window to see as many rows as possible.

general options

General options control basic functions of the viewer such as background coloring, printing options and display lists of databases.

graph window

graph windows display waveform data. Waveform data can be spectral data, comma-separated value data, or analog and digital data.

hide

The action that temporarily removes a waveform from visual display yet keeps the data within the row. The **Hide Waveform** menu item toggles the display of waveforms in rows carrying overlaid plots.

imaginary

A transformation setting used with complex waveforms. Imaginary transformations display the imaginary component of each point in the complex waveform expressed in Rectangular (Real/Imaginary) form.

impedance parameters (Z-Parameters)

Impedance (z-parameter) is the opposition that an alternate or direct current encounters when it reaches an electronic component, circuit, or system. Transmission lines have a property known as “characteristic impedance” (ZO), which is the square root of the inductance/meter divided by the square root of the capacitance per meter of the line.

The Smith Chart is presented in terms of “normalized impedance”, where the actual impedance is divided by the ZO of the particular line being used. In this way, you can use a single Smith Chart calibration for all possible line characteristic impedances.

input unit

The unit of measure for a waveform. The unit of the result waveform is same as the unit of the input waveform. By default, the unit of measure is set globally, however, setting the unit locally will override the default setting.

jpg

JPEG stands for Joint Photographic Experts Group, which is the name of the committee that created a way to compress the file size of photographic, true-color images without diminishing the quality of the image. JPEG is generally used for photographic images. These image files use the .JPEG and .JPG extensions.

JPEG (jpg)

JPEG stands for Joint Photographic Experts Group which is the name of the committee that created a way to compress the file size of photographic, true-color images without diminishing the quality of the image. JPEG is generally used for photographic images and these image files use the .JPEG and .JPG extensions.

JWDB

JWDB stands for Joint Waveform DataBase. This is the default database format for Mentor Graphics simulation applications.

keyboard accelerators

Keyboard accelerators enable you to select menu items in the Graphical user interface (GUI) without using a mouse. If a keyboard accelerator is available for a menu item, it is shown next to the item on the menu. For example, Ctrl+C is the keyboard accelerator for the Copy menu item and is the keyboard accelerator for the Paste menu item. Keyboard accelerators are different from keyboard shortcuts. Sometimes, keyboard accelerators are referred to as Hot keys.

keyboard access keys

Keyboard shortcuts enable you to select menu items in the Graphical user interface (GUI) without using a mouse. The letter for the keyboard shortcut is underlined within each menu and item. For example, capital F is the keyboard shortcut for the file menu. Typing Alt+F on your keyboard will display the file menu for item selection.

layout

The way the application displays x-axis settings and waveform name displays. Within the EZwave Display Preferences dialog box, click **Layout** to access these settings.

magnitude

The transformation applied to a complex waveform that shows the square root of (re2 + im2) for each point in the complex waveform.

mouse strokes

Mouse Strokes provide you with a convenient way to perform common tasks by enabling you to draw shapes using the mouse. For example, drawing the letter “D” deletes the current set of selected objects. Mouse strokes are usually performed with a three button mouse using the middle mouse button to make the stroke.

new rows

Within the active graph window, new rows are created when a waveform is dragged to any location above, below or between existing rows.

over-axis zooming

The action that uses the mouse pointer to drag over the x or y axis to identify the region for zoom.

overlaid plots

The ability to plot multiple waveforms on top of each other in the same row within the graph window. You may plot analog and digital waveforms overlaid. Existing digital waveforms rows cannot accept overlaid plots.

png

PNG stands for Portable Network Graphics. It is an open, extensible image format with lossless compression. These image files use the .PNG extension

phase

The phase of each point in the complex waveform in Polar form. All phase angles are restricted between -180 and +180 degrees (-pi radians and +pi radians).

property

A property is a name/value pair, where the value can be a number or a string. Waveform Databases and individual waveforms may have property lists containing the individual properties. An example of a waveform database property is *timestep:1*.

question mark

The question mark is a mouse stroke action that opens the About Mouse Strokes Information Box listing the supported mouse strokes for this application.

radix

A quantity whose successive integral powers are the implicit multipliers of the sequence of digits that represent a number of some positional-notation systems. Radix levels used with buses include Octal, Hexadecimal, Binary, Decimal, and Ascii.

right-click

The right mouse key serves an important purpose with this application. A number of context sensitive popup menus appear by pressing the right mouse key. In this help system, references to “right-click” indicate this activity.

real

A transformation setting for complex waveforms. The real component of each point in the complex waveform expressed in Rectangular (Real/Imaginary) form.

refresh

The refresh process forces a complete repaint of all windows within an application.

scale table

A scale table contains a matrix of the unit of measurement used by the waveform. For example, this may be in time units, frequency units, voltage, or other measurement. This table contains the unit name and the suffix of the scale unit to be used as the base. It also contains a list of the units and their scaling relationship to each other.

step waveform

A waveform that, from a viewing perspective, approximates a Heaviside (unit step) function.

taskbar

The area directly above the status bar in the application window that contains Graph Window buttons for selecting the active graph window.

tooltip

A tooltip is a small window that contains descriptive text for the item under the mouse pointer.

topline

Topline is the magnitude reference line at the top magnitude, which is the magnitude of the portion of a pulse waveform that represents the second nominal state of a pulse (usually referred as HIGH level).

true

Having a Boolean value of one (1).

undo zoom

Reverses the previous zoom action. Undo zoom will return the display to the previously selected magnification.

Verilog

Notational conventions used by the Waveform Calculator based on the Verilog hardware description language.

VHDL

Notational conventions used by the Waveform Calculator based on VHDL (VHSIC Hardware Description Language).

view all

The action where the magnification is reset to view all of the data in a window or a row within the active graph window.

waveform

A waveform is a collection of values along a time continuum, frequency, or other domain axis. The axis is referred to as the *domain*, and the values positioned along the axis are the *range*. This is really a logical view, as some waveform events, that is those that are part of a functional waveform, may be generated by a function (e.g. sin(t)).

waveform database

A waveform database contains, data of one or more individual waveforms.

workspace

The tabbed area within the application window that holds graph windows.

X Axis

Commonly the horizontal axis used to display time or distance. In this product, the X Axis is used for *over-axis zooming*.

Y Axis

Used with analog waveforms and complex waveforms. With complex waveforms the Y Axis displays transformations of dB, Magnitude, Real, Imaginary, Phase and Continuous Phase. With multiple analog waveform plots (overlaid plots), you may select to have each waveform displayed on its own y axis within the row.

Y-level line

A Y-level line is a horizontal line that marks the intersection of a cursor and a waveform. Right-click on a cursor and select **Y-Level Line** from the popup menu to show the Y-level line. See also *base Y-level line*.

zooming

The action that enlarges the visual display in the active graph window. For example, zooming between cursors adjusts the display to view between the two cursors that are farthest apart in the active graph window. See also *area zoom* and *over-axis zooming*.

Appendix F
Troubleshooting

This section contains troubleshooting information relating to EZwave.

Known Problems and Workarounds 1261

Linux Printing Issues 1264
Resolving Linux Printing Issues 1264
Printing Issues When Using LPRNG 1264
Printing Issues When Using CUPS 1265

Troubleshooting Memory Issues 1267
Configuring Memory Capacity 1267
Resolving Out-Of-Memory Problems 1268
Resolving Incorrect Estimate the Disk Space Needed to Save A Database 1270
Resolving Why the Simulator Fails to Start EZwave 1270
Increasing the Memory Stack Limit 1271

Loading .fsdb and .tr0 Files 1272

Troubleshooting EZwave Launch Issues 1273
Failure to Load EZwave Dynamic Libraries 1273
Failure to Launch EZwave in Questa ADMS GUI Context 1273
Displaying Results Between Different AMS Versions. 1274

Troubleshooting Display Issues 1275
Fonts 1275

Troubleshooting Logfiles 1276
Logfile Locations 1276

Troubleshooting Waveform Comparison 1277

System Error Codes 1277

Contacting the Customer Support Center 1282

Training Classes 1282

Known Problems and Workarounds

This section describes known problems and their workarounds for EZwave.

- Docking EZwave in Questa ADMS

  By default, in graphical mode, EZwave is not docked in the Questa ADMS GUI. A new Questa ADMS modelsim.ini file variable is available to control this behavior:

  ```
  ; DockEZwaveInQuestaAdms = 1.
  ```

If you dock EZwave back to Questa ADMS, you may encounter drag and drop issues within EZwave.

- Sorting in the Open Dialog

  It is not possible to sort by name, size or modified date within the detailed view of the Open dialog box when EZwave is docked inside the Questa ADMS GUI.

- WLF Reader performance on Windows

  Reading wlf on ixw is slow on Windows.

- Running EZwave fails with the following message:

  ```
  Unable to start get_ezwave_license process: check your path
  ```

  This problem is caused by a Java bug that prevents the execution of a script from Java if the script belongs to a different "user" and "group", even if "read" and "execute" permissions are correctly set. This problem occurs only if the user's "group" is not the "primary group". In this case, execution fails if "read" and "execute" permissions are not granted to "other" (or "public").

  For more information on this Java bug, visit:

  http://bugs.sun.com/bugdatabase/view_bug.do?bug_id=4052517

  Workaround 1: Ensure that the file $MGC_AMS_HOME/bin/ams_command.sh permissions are set to 755. The user in charge of the AMS installation can execute the following command:

  ```
  chmod 755 $MGC_AMS_HOME/bins/ams_command.sh
  ```

  Workaround 2: The issue might be due to an unstable machine. Try starting EZwave with the command ezwave -nobigmem. This limits the memory size to 4 GB. If this resolves the issue, reboot the machine and try launching EZwave normally.

- The EZwave viewer generates the following error message during invocation:

  ```
  Font specified in font.properties not found
  --symbol-medium- r-normal--*-%d-*-*-p-*-adobe-fontspecific]
  ```

  Workaround 1: Install the Symbol font. The font that the Java virtual machine searches for is *symbol.ttf*. This font is available on most machines.

  You may get a copy of the font and copy it to the JRE fonts directory, for example, *$MGC_AMS_HOME/jre/.../lib/fonts*. In the *fonts* directory, there is a file named *fonts.dir* that lists the available, scalable fonts for JRE. Modify this file (you may need to be root) to include the *symbol.ttf* font file as in the following example:

  ```
  symbol.ttf -urw-symbol-medium-r-normal--0-0-0-0-p-0-adobe-
  fontspecific
  ```

After you have added the preceding line, increment the number on the first line of the file by one. The number indicates the total number of fonts listed in the file.

Workaround 2: Alternatively, make the font definitions specified in the *font.properties* file match what are available on your system. You need to change the symbol lines in the file from the following:

```
--symbol-medium-r-normal--*-%d-*-*p-*-adobe-fontspecific
```

to the following:

```
-urw-symbol-medium-r-normal--*-%d-*-*p-*-urw-fontspecific
```

- When loading CSV files through a Tcl script, Supported Net Representation Components are not fully functional.

  Workaround: When opening CSV databases using a Tcl script, use the EZwave fully qualified naming convention.

  For example, you want to display V(OUT) from a time domain analysis in the currently open database named “adc12”. Because of this limitation, you must use

  ```
  add wave <adc12/TRAN>V(OUT)
  # Fully qualified naming syntax, works for CSV and other file
  formats.
  ```

  instead of

  ```
  add wave OUT
  # Supported net representation syntax, currently does not work with
  # CSV files
  ```

- In this release, database compaction is disabled during simulation. For large designs the database size may increase in comparison with earlier releases.

  Workaround: Load the results database into EZwave once the simulation is completed, then use **Save As** to save the database. This save will compact the database and the size is reduced. See “Save As Dialog Box” on page 583 for further information.

- Waveform Group Colors

  Custom colors applied to individual elements of a waveform group are not preserved when a database is saved and reloaded as TCL.

- Progress Bar Display of TCL Script Execution on Windows Platforms

  When running EZwave in Windows, TCL source file execution progress is not monitored in the progress bar. Instead, an activity bar is shown during Tcl script execution.

Linux Printing Issues

This section describes the steps that will help you to troubleshoot your printing process.

Resolving Linux Printing Issues **1264**
Printing Issues When Using LPRNG **1264**
Printing Issues When Using CUPS. **1265**

Resolving Linux Printing Issues

If you are having difficulties using */usr/sbin/lpc* to print from a system running Linux, use the steps described here to troubleshoot your printing process.

Solution

1. Ensure that */usr/sbin/lpc* is available on the machine. The *Java Configuration Requirement for Linux* states, “To print on Linux, the */usr/sbin/lpc* utility must be installed. This is a standard Linux utility. On RedHat 7.1, for example, this utility is in the ‘LPRng-3.7.4-22 RPM’ package.”

   On RedHat 3.0, the CUPS package is sometimes used by default, instead of LPRNG. Use the following commands to check whether LPRNG or CUPS is installed on your system:

   - If the command

     ```
     rpm -qa | grep LPR
     ```

     returns a package name, the LPRNG package is installed. If so, refer to “Printing Issues When Using LPRNG” on page 1264.

   - If the command

     ```
     rpm -qa | grep cups
     ```

     returns a package name, the CUPS package is installed. If this is the case, refer to “Printing Issues When Using CUPS” on page 1265.

Printing Issues When Using LPRNG

This topic describes the steps that will help you to troubleshoot your LPRNG printing process.

Solution

1. Look for the following configuration files:

   - */etc/printcap*

Verify that one or more printers are defined in this file. If this file does not exist, the EZwave viewer is normally unable to print. This file is automatically generated during printing configuration executed by the root admin.

- */etc/lpd.perms*

  This file is optional. If it exists, verify that the following lines are present:

  ```
  # allow anybody to get server, status, and printcap
  ACCEPT SERVICE=C LPC=lpd,status,printcap
  ```

- */etc/lpd.conf*

  This file is mandatory. It may be empty. If it is not empty, it may contain settings that prevent proper printing.

2. If the preceding files are all correct, try the following command:

   ```
   /usr/sbin/lpc status
   ```

   This should return a list of printers with configuration and status details. For example:

   ```
   myprinter:
   printer is on device ‘lpd’ speed -1
   queuing is enabled
   printing is enabled
   no entries
   daemon present
   ```

3. Verify that the LPD daemon is running with the following command:

   ```
   /bin/ps -auxww | grep lpd
   ```

   An **lpd** process must be running; otherwise, printing will fail.

Printing Issues When Using CUPS

This topic describes the steps that will help you to troubleshoot your CUPS printing process.

Solution

1. Execute the following command:

   ```
   /usr/sbin/lpc status
   ```

   This should return a list of printers with configuration and status details. For example:

   ```
   myprinter:
   printer is on device 'lpd' speed -1
   queuing is enabled
   printing is enabled
   no entries
   daemon present
   ```

2. Verify that the cups printer daemon is running with the following command:

   ```
   /bin/ps -auxww | grep cupsd
   ```

   A *cupsd* process must be running, otherwise printing will fail.

3. Verify that the following configuration files exist in the */etc/cups* directory:

   - *classes.conf*
   - *client.conf*
   - *cupsd.conf*
   - *mime.convs*
   - *mime.types*
   - *printers.conf*

   This is the basic CUPS recommended configuration.

4. Log in to the machine as root and execute the following command:

   ```
   lpstat -v
   ```

   This should return address information on printers. Look for the corresponding line for the printer named “myprinter”. For example:

   ```
   device for myprinter: lpd://173.21.21.1/myprinter
   ```

   Register the printer using the command:

   ```
   /usr/sbin/lpadmin -r myprinter -E -v lpd://173.21.21.1/myprinter
    -m myprinter.ppd
   ```

   Then, restart the **cups** daemon:

   ```
   path/cups restart
   ```

   path may be one of the following:

   - */etc/software/init.d*
   - */etc/rc.d/init.d*
   - */etc/init.d*
   - */sbin/init.d*

Troubleshooting Memory Issues

Memory problems may stop EZwave from starting from within simulators such as Questa ADMS or Eldo, and out-of-memory errors may occur when you are working with very large files within the EZwave viewer. This section provides troubleshooting information to help you resolve memory issues.

Configuring Memory Capacity **1267**
Resolving Out-Of-Memory Problems **1268**
Resolving Incorrect Estimate the Disk Space Needed to Save A Database **1270**
Resolving Why the Simulator Fails to Start EZwave **1270**
Increasing the Memory Stack Limit **1271**

Configuring Memory Capacity

This topic describes how to set the EZwave memory capacity using the AMS_EZWAVE_CAPACITY environment variable.

You can use the AMS_EZWAVE_CAPACITY environment variable to override the default settings and specify the memory capacity for EZwave. Allowed values are low | medium | high | versatile.

Note

By default, EZwave is set up as if AMS_EZWAVE_CAPACITY is used, and set to medium.

- low - suitable for virtual machines

  ```
  setenv AMS_EZWAVE_CAPACITY low
  ```

  Limits memory usage to 4 GB.

  Configures the Java Virtual Machine minimum heap size -Xms to 100 MB and the maximum heap size -Xmx to 4000 MB.

- medium - suitable for mainstream machines (<48 GB)

  ```
  setenv AMS_EZWAVE_CAPACITY medium
  ```

  This is the default option. Limits memory usage to 75% of available memory, capped at a maximum of 32 GB. For example, if a machine has 128 GB of RAM, EZwave uses 32 GB.

  The Java Virtual Machine minimum heap size -Xms and maximum heap size -Xmx are set to the same value.

- high - suitable for high-end machines (>48 GB)

```
setenv AMS_EZWAVE_CAPACITY high
```

Limits memory usage to 75% of available memory. This option is only taken into account if the machine has more than 48 GB of RAM (otherwise high is equivalent to medium).

The Java Virtual Machine minimum heap size -Xms and maximum heap size -Xmx are set to the same value.

- versatile - suitable for high-end machines (>48 GB) where a large part of the memory might be used by another process causing out-of-memory problems

  ```
  setenv AMS_EZWAVE_CAPACITY versatile
  ```

  Versatile is similar to high but configures the Java Virtual Machine minimum heap size -Xms to 1 GB and the maximum heap size -Xmx to 75% of the available memory.

Tip

Using the AMS_EZWAVE_CAPACITY environment variable is the preferred method of setting EZwave memory capacity.

However, if you still encounter out-of-memory problems, you may need to further modify the Java Virtual Machine's (JVM) memory heap size. Refer to "Resolving Out-Of-Memory Problems" on page 1268.

Resolving Out-Of-Memory Problems

This topic describes the types of memory shortages detected by EZwave and provides solutions to address them.

Note

Using the AMS_EZWAVE_CAPACITY environment variable is the preferred method of setting EZwave memory capacity. Refer to "Configuring Memory Capacity" on page 1267.

However, if you still encounter out-of-memory problems, you may need to modify the Java Virtual Machine's (JVM) memory heap size, as described in this topic.

Symptoms

The EZwave viewer detects and reports memory shortages in the following cases:

- Machine memory is nearly exhausted.
- Disk is full.
- Maximum allowed EZwave memory is almost reached.

In these cases, information is reported to you, and the simulator performs an automatic save of simulation information (time domain simulation).

Causes

If you encounter out-of-memory problems, you may need to increase the Java Virtual Machine's (JVM) memory heap size. By default, EZwave will automatically determine how much memory to use depending on OS and system physical memory.

Solution

The heap size can grow up to 75% of physical memory with a maximum of 31 GB (the default).

- To disable large memory usage for the JWDB server or EZwave, and limit the memory to 4 GB, modify the value of the environment variable AMS_JAVA_NO_BIG_MEMORY as follows:

  ```
  setenv AMS_JAVA_NO_BIG_MEMORY 1
  ```

- To enable large memory usage above 31 GB, use AMS_JAVA_MEMORY_HEAP to force a specific heap size:

  ```
  setenv AMS_JAVA_MEMORY_HEAP "-Xms100G -Xmx100G"
  ```

  where -Xms sets the minimum heap size and -Xmx sets the maximum heap size.

Caution

You should never take all of the machine memory for the Java heap (-Xmx), because Java uses some memory in addition to the heap (e.g. the stack), and other processes might require some memory. Using 75% of the machine memory as a maximum for the heap is considered a good practice.

Note

The AMS_JAVA_MEMORY_HEAP overwrites any other memory setting, including those set by environment variable AMS_EZWAVE_CAPACITY, by defining exact memory limits. AMS_JAVA_MEMORY_HEAP must be unset when using AMS_JAVA_NO_BIG_MEMORY:

```
unsetenv AMS_JAVA_MEMORY_HEAP
```

Setting Values for Minimum and Maximum Heap Size

When using the environment variable AMS_JAVA_MEMORY_HEAP, the -Xms value sets the minimum heap size and the -Xmx value tells the JVM the maximum size it can increase the memory heap to.

Tip

It is considered a good practice to set -Xms and -Xmx to the same value, as it frees the JVM from having to resize the heap size dynamically.

To determine the maximum heap size that your system can handle, gradually increase the -Xmx value, and start the EZwave viewer (without specifying a *.wdb* file). When you reach a value that is too high, the EZwave viewer will return an error telling you that it cannot handle the specified memory allocation.

When using AMS_JAVA_MEMORY_HEAP, you should also add all the options -XX that EZwave sets by default. For example, to setup AMS_JAVA_MEMORY_HEAP to be equivalent of the setup obtained with setenv AMS_EZWAVE_CAPACITY low:

```
setenv AMS_JAVA_MEMORY_HEAP "-Xms100M -Xmx4000M
-XX:+UseConcMarkSweepGC
-XX:+UseParNewGC
-XX:CMSInitiatingOccupancyFraction=30
-XX:+UseCMSInitiatingOccupancyOnly
-XX:-UseGCOverheadLimit
-XX:+OptimizeStringConcat
```

Resolving Incorrect Estimate the Disk Space Needed to Save A Database

EZwave can sometimes incorrectly estimate the disk space necessary to save a database during a simulation. It can force the simulation to stop, suggesting there is not enough disk space even when there is.

Solution

If you encounter such an issue and you are sure there is enough disk space available, you can set the environment variable AMS_NO_CHECK_DISK_SPACE to 1 to deactivate the EZwave checking mechanism for available disk space.

Resolving Why the Simulator Fails to Start EZwave

If a simulator such as Questa ADMS or Eldo fails to start EZwave, an error message is displayed.

Symptoms

An error message similar to the following is displayed:

```
The JWDB (EZwave) server cannot start the Java Virtual Machine. Refer to
EZwave User's Manual for possible causes.
```

Causes

The most common cause of this is an incorrect setup of memory parameters for EZwave.

Solution

The table lists EZwave error messages relating to the Java Virtual Machine's (JVM) memory heap size and specification of it. It describes a probable cause and suggests a possible solution.

Table F-1. JVM Memory Heap Error Messages

Error Message	Probable Cause	Solution
Could not reserve enough space for object heap.	Xmx value too large in comparison to available physical memory.	Reduce Xmx value (use a maximum of 75% of total memory).
Too small initial heap for new size specified.	Xms value too small to allow EZwave to start.	Increase Xms value (use a minimum of 100M).
Incompatible minimum and maximum heap sizes specified.	Xms value is larger than Xmx value.	Reduce Xms or increase Xmx value. Using the same value for Xms and Xmx is considered good practice.
Invalid maximum (minimum) heap size.	Syntax error in Xmx (Xms) value.	Fix the syntax error. Use -Xmx<value>[m|M|g|G]
The specified size exceeds the maximum representable size.	Xmx value is larger than the addressable space.	Reduce Xmx value (use maximum of 75% of total memory).

Note

If EZwave starts standalone, but fails to run from a simulator such as Questa ADMS or Eldo, try reducing your Xmx value. It may be the JVM cannot reserve enough memory for the thread stacks. The size of the stack influences the maximum amount of memory that EZwave can reserve.

Increasing the Memory Stack Limit

When EZwave or the JWDB server is run from a simulator such as Questa ADMS or Eldo, the thread stack limit is set to 10240 kB. This can be increased.

In earlier releases of EZwave, when EZwave or the JWDB server was run from a simulator such as Questa ADMS or Eldo, the thread stack limit was set to unlimited.

If you require a higher stack limit, for example when troubleshooting, you can set the stack limit back to unlimited by setting the following environment variable:

```
setenv AMS_JAVA_MEMORY_STACK 1
```

Loading .fsdb and .tr0 Files

When EZwave loads .fsdb files or .tr0 HSPICE graph data files, initially only the design hierarchy is loaded, not the waveforms. This improves the speed of loading these types of files, and additionally reduces capacity and memory requirements in comparison to earlier releases of EZwave. Small HSPICE graph data files are loaded in full.

Waveforms are loaded when they are displayed for the first time or when they are located at hierarchy level. The Waveform List panel will only display waveforms that have already been loaded.

EZwave will display warnings in the following scenarios:

- Save As — Only loaded waveforms will be saved to the new database.
- Create Bus — Only loaded waveforms are displayed for selection as bits to be grouped.
- Compare Waveforms — Only loaded waveforms are displayed for selection.

Troubleshooting EZwave Launch Issues

Provides troubleshooting information to help you resolve issues related to launching the EZwave viewer.

Failure to Load EZwave Dynamic Libraries 1273
Failure to Launch EZwave in Questa ADMS GUI Context 1273
Displaying Results Between Different AMS Versions 1274

Failure to Load EZwave Dynamic Libraries

This topic describes an error message regarding failure to load EZwave dynamic libraries.

Symptoms

When launching EZwave, if you encounter the following error message:

```
Cannot load “TclIntegration” or “NativeUtil” library, check user EZwave
user’s manual for possible causes.
```

An *ezwave_error.log* or *jwdb_error.log* file is generated containing more details about this issue.

Causes

Possible causes:

- An incorrect or missing **libstdc++** library. The error log will report this. Either:
  - The AMS installation tree is incorrect
  - The library is missing from your system
  - The environment variable LD_LIBRARY_PATH is not set correctly to access it.
- On Linux, some RPM packages are required to properly run EZwave. They could be missing from your system.

Solution

Use the script *ams_dependency_lookup.sh* from the AMS installation tree and check if it reports any error or missing packages.

Failure to Launch EZwave in Questa ADMS GUI Context

This topic describes an error message regarding failure to launch EZwave in Questa ADMS GUI Context.

Symptoms

When launching Questa ADMS in interactive (GUI) mode, you may encounter the following error messages:

```
-    Error: The JWDB (EZwave) server cannot start the Java Virtual
Machine. Refer to EZwave User's Manual for possible causes

-    Error: Could not load 'NativeUtil' (or TclIntegration) library,
check EZwave User's manual for possible causes.
```

Causes

There may be a problem with your AMS installation tree or packages are missing from your computer. Refer to the section "Failure to Load EZwave Dynamic Libraries" on page 1273 for further details.

Solution

In certain configurations, it is possible that undocking EZwave from Questa ADMS may work around the issue of missing packages.

Displaying Results Between Different AMS Versions

EZwave cannot display results from different AMS versions of a running simulation.

This can happen when you launch an EZwave GUI from one particular environment, for example an AMS12.1 installation tree, then launch an Eldo simulation in an AMS12.2 environment. In this situation, opening a *.wdb* file during the simulation is not supported.

To resolve this issue, you must either:

- Wait until the end of the simulation to open the disk database with a later version of EZwave.
- Ensure that both EZwave's GUI and Server are of the same AMS version.

Troubleshooting Display Issues

This section describes issues with the EZwave display.

Fonts 1275

Fonts

The fonts available will vary from system to system, depending on what is installed. To use a new font in the EZwave viewer, a system administrator must add the font to the system as well as the *font.properties* file of the java package.

Saving EZwave files only preserves font changes within the same operating system. This is because fonts (type, size, and style) vary among different operating systems. Even with the same operating system, font changes may not be preserved if the fonts are not installed on the system. If a saved font is not available on a system, the EZwave viewer uses the default font. Color changes are not affected by different systems and are preserved when saving.

Troubleshooting Logfiles

EZwave creates a logfile on your system when it is active. The logfile keeps track of the commands issued during each viewing session. Should you experience difficulty in using EZwave, you may be asked to locate or provide the logfile from your session when contacting customer support.

You can view the logfile using any standard text editor.

Logfile Locations 1276

Logfile Locations

Logfile locations vary based on the installation directory structure used by the host application.

Because products using EZwave can send the logfile to a location of their choosing, your installation directory specification should not carry unique characters that may prohibit the logfile from being created correctly.

Special characters to be avoided are listed in the table:

Table F-2. Characters to Avoid in Logfiles

Symbol	Character
*	asterisk
^	caret
(	open parentheses
)	close parentheses
:	colon
;	semicolon
`	accent grave
'	single straight quotation marks
"	straight quotation marks
=	equal sign
,	comma
/	forward slash
?	question mark
>	right angle brackets

Troubleshooting Waveform Comparison

This topic describes the error messages that may be encountered during a Waveform Comparison.

Solution

Warning: could not compare '<waveform_name>', an equivalent waveform was not found in the Test Database	Issued when a waveform with the same name as the test dataset waveform cannot be found in the reference dataset.
Warning: could not compare '<waveform_name1>' with '<waveform_name2>', both waveforms are not compound	Issued when trying to compare a compound waveform with one that is not compound
Warning: could not compare '<waveform_name>', both waveforms are not bus	Issued when trying to compare a bus waveform with one that is not a bus
Warning: could not compare '<waveform_name>', applying delay failed	Issued when applying a delay parameter (-**refdelay** or -**testdelay**) within compare start fails
Warning: '<waveform_name>' has been ignored during comparison because its type is not yet supported. Currently supported types are: analog continuous waveform (except complex), digital waveform and bus, records or compound waveform of such waveforms.	Issued when an unsupported waveform type is encountered during the comparison. For more details on currently supported waveform types and how they are compared see "Support for Different Types of Waveform" on page 231.

Note

Assertions and complex waveforms are not supported in the Waveform Comparison.

System Error Codes

Lists the system error codes that are displayed when system-call errors are encountered.

The table lists the error codes for the Linux operating system. For further details on these error codes, check your operating system's user's manual.

Table F-3. System Error Codes

Error Code	Description
1	Operation not permitted
2	No such file or directory
3	No such process

Table F-3. System Error Codes (cont.)

Error Code	Description
4	Interrupted system call
5	I/O error
6	No such device or address
7	Argument list too long
8	Exec format error
9	Bad file number
10	No child processes
11	Try again
12	Out of memory
13	Permission denied
14	Bad address
15	Block device required
16	Device or resource busy
17	File exists
18	Cross-device link
19	No such device
20	Not a directory
21	Is a directory
22	Invalid argument
23	File table overflow
24	Too many open files
25	Not a typewriter
26	Text file busy
27	File too large
28	No space left on device
29	Illegal seek
30	Read-only file system
31	Too many links
32	Broken pipe
33	Math argument out of domain of function

Table F-3. System Error Codes (cont.)

Error Code	Description
34	Math result not representable
35	Resource deadlock would occur
36	File name too long
37	No record locks available
38	Function not implemented
39	Directory not empty
40	Too many symbolic links encountered
41	Operation would block
42	No message of desired type
43	Identifier removed
44	Channel number out of range
45	Level 2 not synchronized
46	Level 3 halted
47	Level 3 reset
48	Link number out of range
49	Protocol driver not attached
50	No CSI structure available
51	Level 2 halted
52	Invalid exchange
53	Invalid request descriptor
54	Exchange full
55	No anode
56	Invalid request code
57	Invalid slot
59	Bad font file format
60	Device not a stream
61	No data available
62	Timer expired
63	Out of streams resources
64	Machine is not on the network

Table F-3. System Error Codes (cont.)

Error Code	Description
65	Package not installed
66	Object is remote
67	Link has been severed
68	Advertise error
69	srmount error
70	Communication error on send
71	Protocol error
72	Multihop attempted
73	RFS specific error
74	Not a data message
75	Value too large for defined data type
76	Name not unique on network
77	File descriptor in bad state
78	Remote address changed
79	Can not access a needed shared library
80	Accessing a corrupted shared library
81	.lib section in a.out corrupted
82	Attempting to link in too many shared libraries
83	Cannot exec a shared library directly
84	Illegal byte sequence
85	Interrupted system call should be restarted
86	Streams pipe error
87	Too many users
88	Socket operation on non-socket
89	Destination address required
90	Message too long
91	Protocol wrong type for socket
92	Protocol not available
93	Protocol not supported
94	Socket type not supported

Table F-3. System Error Codes (cont.)

Error Code	Description
95	Operation not supported on transport endpoint
96	Protocol family not supported
97	Address family not supported by protocol
98	Address already in use
99	Cannot assign requested address
100	Network is down
101	Network is unreachable
102	Network dropped connection because of reset
103	Software caused connection abort
104	Connection reset by peer
105	No buffer space available
106	Transport endpoint is already connected
107	Transport endpoint is not connected
108	Cannot send after transport endpoint shutdown
109	Too many references: cannot splice
110	Connection timed out
111	Connection refused
112	Host is down
113	No route to host
114	Operation already in progress
115	Operation now in progress
116	Stale NFS file handle
117	Structure needs cleaning
118	Not a XENIX named type file
119	No XENIX semaphores available
120	Is a named type file
121	Remote I/O error
122	Quota exceeded
123	No medium found
124	Wrong medium type

Contacting the Customer Support Center

Mentor software support includes software enhancements, technical support and access to comprehensive on-line services with Support Center.

Note

If you are using this product as a component within another Mentor Graphics product (for example, Questa ADMS), refer to the guidelines for support with that product before contacting Mentor Support.

The Support Center is available here:

`https://support.mentor.com`

Before contacting support, gather the following information:

- Product and version (Obtained from the **Help** > **About** menu item on the main application window)
- Any test files
- Exact steps or procedures causing the problem
- The two files that the EZwave viewer and JWDB server generate:
    - *ezwave_error.log*
    - *jwdb_error.log*

  These files are normally generated at the location where the tool is run (EZwave or the simulator) and contain information that may be useful in identifying the root of the problem, particularly if it's difficult to reproduce.

Training Classes

Classes are held regularly at Mentor Graphics and are available on-site at your premises.

Contact your local sales office or consult www.mentor.com for rates and a current class schedule.

Index

— A —

Access keys, 70
Add Clock dialog, 422
Add Clock Dialog Box, 422
add wave Tcl command, 1025
add workspace Tcl command, 1032
Aligning Y-Axes, 136
Application
  application window, 68
  graphic window, 90, 95
  menu bar, 70
  overview
    application window, 68
  status bar, 93
  toolbar, 81
  waveform list panel, 83
  waveform list popup menu, 106
  workspace and workspace tabs, 92, 93, 113
ASCII files, 47
Auto Correlation Dialog, 602
Automatic Reload Dialog, 478
Axis Properties Dialog, 427

— B —

batch_mode Tcl command, 1033

— C —

CDF Legend Options Dialog Box, 516
CDF Measures Options Dialog Box, 514
CDF Plot Options, 512
Chip Transform Dialog box, 605
compare add Tcl command, 1035
compare clock Tcl command, 1040
compare configure Tcl command, 1042
compare end Tcl command, 1046
compare info Tcl command, 1047
compare list Tcl command, 1048
compare options Tcl command, 1049
compare run Tcl command, 1055
compare savelog command, 1056
compare savelog Tcl command, 1056
compare saverules Tcl command, 1057
compare start Tcl command, 1058
Comparison Options Dialog, 429
Comparison rules, 246
Constellation Diagram Dialog, 609
Convolution Dialog, 611
Correlogram Method, 391
COU files, 46
Create bus, 273
Create Bus Dialog, 439
Cross Correlation Dialog, 613
CS V files, 46
Cursors
  menu
    popup, 104
  popup menu, 104
  selecting a base cursor, 166
Cursors Dialog, 480

— D —

Data Format Dialog, 482
dataset alias Tcl command, 1060
dataset analysis Tcl command, 1061
dataset clear Tcl command, 1062
dataset close Tcl command, 1034, 1063
dataset info Tcl command, 1065
dataset list Tcl command, 1066
dataset merge Tcl command, 1067, 1069
dataset mergewaveforms Tcl command, 1069
dataset open Tcl command, 1071
dataset ovd Tcl command, 1072
dataset rename Tcl command, 1076
dataset save Tcl command, 1077, 1080
dataset savewaveforms Tcl command, 1080
dataset statistics Tcl command, 1084
dataset supported Tcl command, 1086
delete wave Tcl command, 1089
Dialog
  Add Clock, 422

Auto Correlation, 602
Automatic Reload, 478
Axis Properties, 427
Chip Transform, 605
Comparison Options, 429
Constellation Diagram, 609
Convolution, 611
Create Bus, 439
Cross Correlation, 613
Cursors, 480
Data Format, 482
DNA Advisor, 615
Edit Digital Transformation, 455
Event Search Tool, 453
EVM and BER, 617
Eye Diagram, 619
Eye Diagram Tool, 457
Eye Mask, 470
Fast Fourier Transform Tool, 621
Filter, 550
Filters Setup, 567
Find Tool, 553
Foreign Databases, 484
General, 430, 485
Harmonic Distortion, 626
Histogram, 628
Inverse Fast Fourier Transform, 630
Layout, 490
Look and Feel, 491
Measurement Tool, 560
Mouse Pointer, 492
Multiple Run, 493
Parameter Analyzer Tool, 564
Parameter Table, 568
Phase Noise, 632
Power Spectral Density, 634
PSS Residue, 637
RF, 498
Row, 499
Run Filter, 580
Save, 501, 503
Save As, 583
Save Windows, 586
Saving Multiple Databases, 412
Select Hierarchy, 587
Select Waveforms, 589
Signal to Noise Ratio, 639
Spectrum Measurement Tool, 642
Tcl File Viewer, 648
Text Annotation, 504, 531
Top Noise, 650
Transformations, 505
Waveform, 507
Waveform Calculator, 545
Waveform Compare, 508
Waveform List, 509
Waveform Names Display, 594
Waveform Properties, 596
Windowing Transform, 652
Workspace, 511
DNA Advisor Dialog, 615
DO files, 47
dofile Tcl command, 1090

— E —

Edit Digital Transformation Dialog, 455
environment Tcl command, 1091
evalExpression Tcl command, 1092
Event search, 194
Event Search Tool Dialog, 453
EVM and BER Dialog, 617
examine Tcl command, 1093
exit Tcl command, 1095
Eye Diagram Dialog, 619
Eye Diagram Tool Dialog, 457
Eye Mask Dialog, 470
EZwave
installation, 54
invocation, 41

— F —

Fast Fourier Transform Tool Dialog, 621
FFT, 363
File types, 45
Filter Dialog Box, 550
Filters Setup Dialog, 567
find analogs Tcl command, 1096, 1097
find digitals Tcl command, 1098
find nets Tcl command, 1099
find signals Tcl command, 1099
Find Tool Dialog, 553

Foreign Databases Dialog, 484
FSDB files, 47
Fully qualified names, 1013

— G —
General Dialog, 430, 485
getactivecursortime Tcl command, 1100
Getting started
  application window, 68
Graph window
  application window, 68
  graph window menus, 95
  graph window popup menu, 96
  graphic window, 90, 95
  row popup menu, 96
  status bar, 92
  waveform shortcut menus, 113
  Xand Y-Axis shortcut menus, 99
Graphical user interface (GUI)
  application window, 68

— H —
Harmonic Distortion Dialog, 626
Help menu, 80
Histogram Dialog, 628
HSPICE files, 46
HSPICE/HyperLynx output file, 47

— I —
ICX Charter waveform files, 46
Installation
  EZwave Environment Variables, 54
Inverse Fast Fourier Transform Dialog, 630
Invocation, 41

— K —
Keyboard shortcuts
  menus, 70

— L —
Layout Dialog, 490
Look and Feel Dialog, 491

— M —
Manually comparing waveforms, 236
Marching waveforms, 485, 984
Measurement Tool Dialog, 560
Menus
  application window, 68
  cursor menu, 78
  edit menu, 71
  file menu, 70
  format menu, 75
  help menu, 80
  menu bar, 70
  tools menu, 76
  view menu, 74
  window menu, 79
MGC database files, 45
Mixed signal
  plotting, 134
Mouse Pointer Dialog, 492
Multiple Run Dialog, 493

— N —
Nutmeg files, 46

— O —
OVD files, 46

— P —
Parameter Analyzer Tool Dialog, 564
Parameter Table Dialog Box, 568
Periodogram Method, 392
Phase Noise Dialog, 632
Plotting mixed signals, 134
Plotting rules, 134
Popup menus
  database, 106
  delete command, 114
  graph window menus, 95
  graph window popup menu, 96
  row shortcut menu, 96
  waveform list popup menus, 106
  waveform list waveform, 106
  waveform popup menus, 113
  workspace, 92, 93, 113
  Xand Y-Axis menu, 99
Power Spectral Density Dialog, 634
precision Tcl command, 1101
printenv Tcl command, 1102
Procedures

adding waveforms to the Graph window, 122
create a bus, 273
perform a FFT, 363
plot a waveform using drag-and-drop, 124
plot analog and digital waveforms, 134
save and restoring graph windows, 404
saving multiple databases, 412
transform analog waveform to digital, 275
use the measurement tool with compound waveforms, 279
PSF files, 46
PSS Residue Dialog, 637

— Q —

quit Tcl command, 1103
QWAVEDB files, 47

— R —

radix define Tcl command, 1106
radix delete Tcl command, 1108
radix list Tcl command, 1109
radix names Tcl command, 1110
radix signal Tcl command, 1111
radix Tcl command, 1104, 1106, 1108, 1109, 1110, 1111
RF Dialog, 498
Row Dialog, 499
Run Filter Dialog Box, 580

— S —

Save As Dialog, 583
Save Dialog, 501
Save Window Dialog, 503
Save Windows Dialog, 586
Saving and restoring sessions, 404
Saving graph windows, 404
Saving post processed waveform, 411
Saving waveform databases, 412
Select Hierarchy Dialog, 587
Select Waveforms Dialog, 589
setenv Tcl command, 1113
Signal to Noise Ratio Dialog, 639
Spectrum Measurement Tool Dialog, 642
SPICE files, 47
Status bar, 93

— T —

Taskbar, 92, 93, 113
Tcl commands, 1003
add wave, 1025
add workspace, 1032
batch_mode, 1033
bloc, 1034
compare add, 1035
compare clock, 1040
compare configure, 1042
compare end, 1046
compare info, 1047
compare list, 1048
compare options, 1049
compare run, 1055
compare savelog, 1056
compare saverules, 1057
compare start, 1058
dataset alias, 1060
dataset analysis, 1061
dataset clear, 1062
dataset close, 1063
dataset info, 1065
dataset list, 1066
dataset merge, 1067, 1069
dataset mergewaveforms, 1069
dataset open, 1071
dataset ovd, 1072
dataset rename, 1076
dataset save, 1077, 1080
dataset savewaveforms, 1080
dataset statistics, 1084
dataset supported, 1086
delete wave, 1089
dofile, 1090
environment, 1091
evalExpression, 1092
examine, 1093
exit, 1095
find analogs, 1096, 1097
find digitals, 1098
find nets, 1099
find signals, 1099
getactivecursortime, 1100
precision, 1101

printenv, 1102
quit, 1103
radix, 1104, 1106, 1108, 1109, 1110, 1111
radix define, 1106
radix delete, 1108
radix list, 1109
radix names, 1110
radix signal, 1111
setenv, 1113
unsetenv, 1114
wave activecursor, 1115
wave activewindow, 1116
wave activeworkspace, 1117
wave addannotation, 1118
wave addaxisdeltamarker, 1120
wave addcursor, 1122
wave adddeltamarker, 1124
wave addline, 1126
wave addmarker, 1128
wave addproperty, 1129
wave addwindow, 1130
wave addworkspace, 1131
wave cdf, 1132
wave closewindow, 1137
wave closeworkspace, 1138
wave colortheme, 1139, 1142
wave compressvri, 1140
wave cursortime, 1143
wave deletecursor, 1144
wave difference, 1145
wave displayed, 1146
wave exists, 1147
wave ezwave_title, 1149
wave ipnvri, 1155
wave launchfolder, 1163
wave listworkspace, 1164
wave loadbindings, 1165
wave lockcursor, 1166
wave names, 1167
wave plotpacipn, 1170
wave refresh, 1172
wave rowfit, 1173
wave runindexlist, 1174
wave runparameters, 1175
wave runparametervalue, 1176
wave show, 1177, 1178, 1179
wave showgridlines, 1178
wave showzerolevels, 1179
wave tandem mode, 1180
wave tile, 1181
wave windecoration, 1182
wave windowlist, 1183
wave xaxis, 1184
wave yaxis, 1185
wave zoomfull, 1187
wave zoomin, 1188
wave zoomlast, 1189
wave zoomout, 1190
wave zoomrange, 1191
wfc, 1192
write jpeg, 1194
write png, 1195
write wave, 1196
Tcl File Viewer Dialog, 648
Tcl files, 48
Text Annotation Dialog, 504, 531
Toolbar, 81
application window, 68
Tools
creating a bus, 273
event search tool, 194
Top Noise Dialog, 650
Transformation
analog to digital, 275
Transformations Dialog, 505

— U —

unsetenv Tcl command, 1114

— V —

VCD files, 46

— W —

wave activecursor Tcl command, 1115
wave activewindow Tcl command, 1116
wave activeworkspace Tcl command, 1117
wave addannotation Tcl command, 1118
wave addaxisdeltamarker Tcl command, 1120
wave addcursor Tcl command, 1122
wave adddeltamarker Tcl command, 1124
wave addline Tcl command, 1126

wave addmarker Tcl command, 1128
wave addproperty Tcl command, 1129
wave addwindow command, 1130
wave addwindow Tcl command, 1130
wave addworkspace Tcl command, 1131
wave cdf Tcl command, 1132
wave closewindow Tcl command, 1137
wave closeworkspace Tcl command, 1138
wave colortheme Tcl command, 1139, 1142
wave compressvri Tcl command, 1140
wave cursortime Tcl command, 1143
wave deletecursor Tcl command, 1144
wave difference Tcl command, 1145
wave displayed Tcl command, 1146
wave exists Tcl command, 1147
wave ezwave_title Tcl command, 1149
wave ipnvri Tcl command, 1155
wave launchfolder Tcl command, 1163
wave listworkspace Tcl command, 1164
wave loadbindings Tcl command, 1165
wave lockcursor Tcl command, 1166
wave names Tcl command, 1167
wave plotpacipn Tcl command, 1170
wave refresh Tcl command, 1172
wave rowfit Tcl command, 1173
wave runindexlist Tcl command, 1174
wave runparameters Tcl command, 1175
wave runparametervalue Tcl command, 1176
wave show Tcl command, 1177, 1178, 1179
wave showgridlines Tcl command, 1178
wave showzerolevels Tcl command, 1179
wave tandem mode Tcl command, 1180
wave tile Tcl command, 1181
wave windecoration Tcl command, 1182
wave windowlist Tcl command, 1183
wave xaxis Tcl command, 1184
wave yaxis Tcl command, 1185
wave zoomfull Tcl command, 1187
wave zoomin Tcl command, 1188
wave zoomlast Tcl command, 1189
wave zoomout Tcl command, 1190
wave zoomrange Tcl command, 1191
Waveform calculator
    ASCII
        saving a post processed waveform, 411
        saving a WDB as an ASCII file, 417
    built-in functions, 411
    histogram, 628
    performing a FFT, 363
    saving post processed waveform, 411
    using the waveform calculator, 116, 312, 563, 601
Waveform Calculator Dialog, 545
Waveform Compare Dialog, 508
Waveform Compare Wizard, 231
Waveform comparison
    viewing results, 240
Waveform Dialog, 507
Waveform List Dialog, 509
Waveform list panel, 83
    application window, 68
    waveform list popup menus, 106
Waveform Names Display Dialog, 594
Waveform plotting rules, 134
Waveform Properties Dialog, 596
Waveforms
    adding waveforms to the graph window, 122
    analog waveform comparison algorithm, 258
    comparing
        manually, 236
        using the wizard, 231
    comparison
        comparison rules, 246
        options, 248
        reports, 244
        Tolerance options, 253
        viewing results, 240
    Waveform Compare Wizard, 231
    waveform list panel, 83
wfc Tcl command, 1192
Window
    application window, 68
    graphic window, 90, 95
Window menu, 79
Windowing Transform Dialog, 652
WLF files, 46
Workspace
    application window, 68

taskbar, 92, 93, 113
workspace and workspace tabs, 92, 93, 113
Workspace Dialog, 511
write jpeg Tcl command, 1194
write png Tcl command, 1195
write wave Tcl command, 1196

— Y —

Y-Axis
setting as reference, 136

Third-Party Information

Details on open source and third-party software that may be included with this product are available in the *<your_software_installation_location>/legal* directory.

End-User License Agreement
with EDA Software Supplemental Terms