

ARTISAN®
TECHNOLOGY GROUP

Your **definitive** source for quality pre-owned equipment.

Artisan Technology Group
(217) 352-9330 | sales@artisantg.com | artisantg.com

Full-service, independent repair center
with experienced engineers and technicians on staff.

We buy your excess, underutilized, and idle equipment
along with credit for buybacks and trade-ins.

Custom engineering
so your equipment works exactly as you specify.

- Critical and expedited services
- In stock / Ready-to-ship
- Leasing / Rentals / Demos
- ITAR-certified secure asset solutions

Expert team | Trust guarantee | 100% satisfaction

All trademarks, brand names, and brands appearing herein are the property of their respective owners.

Visit our website - *Click HERE*

LeCroy

SERIAL DATA ANALYZER



OPERATOR'S MANUAL

December, 2007

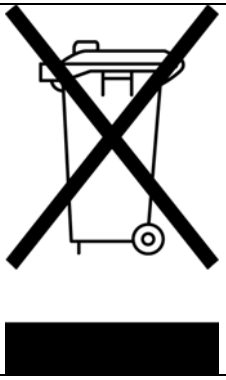
LeCroy Corporation
700 Chestnut Ridge Road
Chestnut Ridge, NY 10977–6499
Tel: (845) 578 6020, Fax: (845) 578 5985

Internet: www.lecroy.com

© 2007 by LeCroy Corporation. All rights reserved.

LeCroy, ActiveDSO, WaveLink, JitterTrack, WavePro, WaveMaster, WaveSurfer, WaveExpert, WaveRunner and WaveJet are registered trademarks of LeCroy Corporation. Other product or brand names are trademarks or requested trademarks of their respective holders. Information in this publication supersedes all earlier versions. Specifications subject to change without notice.

Manufactured under an ISO 9000 Registered Quality Management System
Visit www.lecroy.com to view the certificate.



This electronic product is subject to disposal and recycling regulations that vary by country and region. Many countries prohibit the disposal of waste electronic equipment in standard waste receptacles. For more information about proper disposal and recycling of your LeCroy product, please visit www.lecroy.com/recycle..

SDA-OM-E Rev H
915657-00 Rev A

Introduction 18
How to Use Online Help 18
Type Styles 18
Instrument Help 18
Windows Help 19
Returning a Product for Service or Repair 19
Technical Support 19
Staying Up-to-Date 20
Warranty 20
Windows License Agreement 20
End-user License Agreement For LeCroy X-Stream Software 21
Virus Protection 27
SDA Specifications 28
Vertical System 28
Horizontal System 29
Jitter Noise Floor 29
Acquisition System 29
Memory 30
SDA 18000/13000/11000/9000 30
Acquisition Modes 30
Acquisition Processing 30
Triggering System 30
Automatic Setup 31
Probes 31
Basic Triggers 31
SMART Triggers 31
SMART Triggers with Exclusion Technology 31
Serial Trigger 32
Clock Recovery 32
Jitter Analysis 32
Math Tools – Standard 33
Automated Measure Tools – Standard 33
Pass/Fail Testing 34
Advanced Math Package (XMATH) – Standard 34
Jitter and Timing Analysis Package (JTA2) – Standard 34
ASDA-J - Standard 35
8B/10B Protocol Decoding - Standard 35
Advanced Customization Package (XDEV) – Optional 35

Color Waveform Display 35
Analog Persistence Display 36
Zoom Expansion Traces 36
CPU 36
Internal Waveform Memory 36
Setup Storage 36
Interface 36
Auxiliary Output 36
Auxiliary Input 36
General 37
Environmental Characteristics 37
Certifications 37
CE Declaration of Conformity 37
Physical Dimensions 38
Warranty and Service 39
Optical Reference Receiver (with OE425/OE455) 39
Software Clock Recovery System 39
Communications Mask Testing 40
Jitter Testing 40
Bit Error Rate (optional) 41
Safety 42
Safety Requirements 42
Safety Symbols 42
Operating Environment 43
Cooling 44
AC Power Source 44
Power Consumption 45
Power and Ground Connections 45
Standby (Power) Switch and Scope Operational States 46
Fuse Replacement 46
Calibration 47
Cleaning 47
Abnormal Conditions 47
Basic Controls 48
Alternate Access Methods 48
Mouse and Keyboard Operation 48
Toolbar Buttons 48
Trace Descriptors 49

Trace Annotation 50
Annotating a Waveform 51
Turning on a Channel Trace Label 52
Installation 53
Software 53
Checking the Scope Status 53
Default Settings 53
DDA, SDA, and WaveRunner DSOs 53
Adding a New Option 53
Restoring Software 54
Restarting the Application 54
Restarting the Operating System 55
Removable Hard Drive 55
Connecting to a Signal 56
ProBus Interface 56
ProLink Interface 56
Connecting the Adapters 58
AP-1M Hi-Z Adapter 59
Auxiliary Output Signals 60
Auxiliary Output Setup 60
Sampling Modes 61
Sampling Modes 61
Sampling Mode Selection 61
Single-shot Sampling Mode 61
Basic Capture Technique 61
Sequence Sampling Mode Working With Segments 62
Sequence Mode Setup 63
Sequence Display Modes 64
Displaying Individual Segments 65
Viewing Time Stamps 66
RIS SAMPLING Mode - For Higher Sample Rates 66
Roll Mode 67
Vertical Settings and Channel Controls 67
Adjusting Sensitivity and Position 67
Adjusting Sensitivity 67
Adjusting the Waveform's Position 68
Coupling 68
Overload Protection 68
Coupling Setup 68

Probe Attenuation 68
Setting Probe Attenuation 68
Bandwidth Limit 69
Linear and (SinX)/X Interpolation 69
Interpolation Setup 69
Inverting Waveforms 69
QuickZoom 69
Turning On a Zoom 69
Finding Scale 70
Using Find Scale 70
Variable Gain 70
Enabling Variable Gain 70
Channel Deskew 70
Channel Deskew Setup 70
Group Delay Compensation 71
Timebase and Acquisition System 72
Timebase Setup and Control 72
Autosetup 72
SDA DBI Controls 72
SDA18000 DBI Controls 72
SDA 13000/11000/9000 DBI Controls 73
Smart Memory 73
SMART Memory Setup 74
External Timebase vs. External Clock 74
Triggering 74
TRIGGER SETUP CONSIDERATIONS 74
Trigger Modes 74
Trigger Types 75
Determining Trigger Level, Slope, Source, and Coupling 76
Trigger Source 77
Level 77
Holdoff by Time or Events 77
Hold Off by Time 78
Hold Off by Events 79
Simple Triggers 79
Edge Trigger on Simple Signals 79
Control Edge Triggering 80
Edge Trigger Setup 80

SMART Triggers 83
Width Trigger 83
Glitch Trigger 84
Interval Trigger 85
Qualified Trigger 89
State Trigger 91
Dropout Trigger 92
Logic Trigger 94
Serial Trigger 95
Aux Input Trigger 95
Aux Input Setup 95
Display Formats 96
Display Setup 96
Sequence Mode Display 97
Persistence Setup 97
Saturation Level 97
3-Dimensional Persistence 98
Show Last Trace 99
Persistence Time 100
Locking of Traces 100
Persistence Setup 100
Screen Saver 101
Moving Traces from Grid to Grid 101
Zooming Waveforms 102
Zooming a Single Channel 103
Zooming by Touch-and-Drag 103
Zooming Multiple Waveforms Quickly 104
Multi-Zoom 104
XY Display 105
XY Display Setup 106
Save and Recall 106
Saving and Recalling Scope Settings 106
Saving Scope Settings 106
Recalling Scope Settings 107
Recalling Default Settings 107
Saving Screen Images 107
Saving and Recalling Waveforms 108
Saving Waveforms 108
Recalling Waveforms 110

Disk Utilities 110
Deleting a Single File 110
Deleting All Files in a Folder 111
Creating a Folder 111
Printing and File Management 111
Print, Plot, or Copy 111
Printing 111
Printer Setup 111
Printing 112
Adding Printers and Drivers 112
Changing the Default Printer 112
Managing Files 112
Hard Disk Partitions 112
100Base-T Ethernet Connection 113
Connecting to a Network 113
Communicating over the Network 113
Windows Setups 113
System Restore 114
Track Views 114
Creating and Viewing a Trend 114
Creating a Track View 115
Histograms 116
Creating and Viewing a Histogram 116
Single Parameter Histogram Setup 116
Viewing Thumbnail Histograms 118
Persistence Histogram 118
Persistence Trace Range 118
Persistence Sigma 118
Histogram Parameters 119
Histogram Theory of Operation 132
DSO Process 133
Parameter Buffer 133
Capture of Parameter Events 134
Histogram Parameters (XMAP and JTA2 Options) 134
Histogram Peaks 135
Binning and Measurement Accuracy 135
Waveform Measurements 136
Measuring with Cursors 136

Cursor Measurement Icons 136
Cursors Setup 137
Quick Display 137
Full Setup 137
Overview of Parameters 137
Turning On Parameters 137
Quick Access to Parameter Setup Dialogs 138
Status Symbols 139
Using X-Stream Browser to Obtain Status Information 139
Statistics 141
Applying a Measure Mode 141
Measure Modes 141
Standard Vertical Parameters 141
My Measure 142
Parameter Math (XMath or XMAP option required) 142
Logarithmic Parameters 142
Excluded Parameters 142
Parameter Script Parameter Math 143
Param Script vs. P Script 143
Parameter Math Setup 144
Parameter Script Math Setup 144
Measure Gate 145
Measure Gate Setup 146
Help Markers 147
Help Marker Setup 149
Turning Off Help Markers 149
Customizing a Parameter 149
From the Measure Dialog 149
From a Vertical Setup Dialog 150
From a Math Setup Dialog 150
Parameter Calculations 150
Parameters and How They Work 150
Determining Time Parameters 152
Determining Differential Time Measurements 154
Level and Slope 154
List of Parameters 155
Qualified Parameters 177
Range Limited Parameters 177
Waveform Gated Parameters 178

Waveform Qualifier Setup 178
Waveform Math 178
Introduction to Math Traces and Functions 178
MATH MADE EASY 179
Setting Up a Math Function 179
Resampling To Deskew 180
Resampling 180
Rescaling and Assigning Units 180
Rescaling Setup 182
Averaging Waveforms 182
Summed vs. Continuous Averaging 182
Continuous Averaging Setup 184
Summed Averaging Setup 184
Enhanced Resolution 184
How the Instrument Enhances Resolution 184
Enhanced Resolution (ERES) Setup 187
Waveform Copy 187
Waveform Sparser 187
Waveform Sparser Setup 188
Interpolation 188
Interpolation Setup 188
Fast Wave Port 188
Fast Wave Port Setup - Initial 189
Setup - Case 1 190
Setup - Case 2 190
Setup - Case 3 190
Operational Notes 191
Data Length Limitations 191
Performance 191
Choice of Programming Language 191
Example Application 191
Header Description 194
Data Length Limitations 195
Performance 195
Choice of Programming Language 195
FFT 195
Why Use FFT? 195
Improving Dynamic Range 198
Record Length 198

FFT Algorithms 199
Glossary 200
FFT Setup 203
Analysis 204
Pass/Fail Testing 204
Comparing Parameters 204
Mask Tests 205
Actions 205
Setting Up Pass/Fail Testing 206
Initial Setup 206
Comparing a Single Parameter 206
Comparing Dual Parameters 208
Mask Testing 209
Utilities 210
Status 210
Status Dialog Access 210
Remote communication 210
Remote Communication Setup 210
Configuring the Remote Control Assistant Event Log 211
Hardcopy 211
Printing 211
Clipboard 211
File 211
E-Mail 212
Aux Output 212
Date & Time 212
Setting the Time and Date Manually 212
Setting the Time and Date from the Internet 213
Setting the Time and Date from Windows 213
Options 214
Preferences 214
Audible Feedback 214
Auto-calibration 214
Offset Control 214
Delay Control 215
Trigger Counter 215
Performance Optimization 215
E-mail 216
Acquisition Status 216

Service 217
Show Windows Desktop 217
Touch-Screen Calibration 217
Customization 218
Customizing Your Instrument 218
Introduction 218
Solutions 219
Examples 219
What is Excel? 224
What is Mathcad? 224
What is MATLAB? 224
What is VBS? 224
What can you do with a customized instrument? 226
Number of Samples 227
Calling Excel from Your Instrument 227
How to Select a Math Function Call 227
How to Select a Parameter Function Call 227
The Excel Control Dialog 228
Entering a File Name 228
Organizing Excel Spreadsheets 229
Setting the Vertical Scale 230
Trace Descriptors 230
Multiple Inputs and Outputs 230
Simple Excel Example 1 231
Simple Excel Example 2 234
Excel Example 1: Exponential Decay Time Constant Excel Parameter 238
Excel Example 2: Gated Parameter Using Excel 239
How Does this Work? 241
Excel Example 3: Correlation Excel Waveform Function 242
Excel Example 4: Multiple Traces on One Grid 244
Excel Example 5: Using a Surface Plot 247
Writing VBScripts 248
Types of Scripts in VBS 248
Loading and Saving VBScripts 248
The default parameter function script: explanatory notes 251
Scripting with VBScript 252
Variable Types 252
Variable Names 253

Arithmetic Operators 254
VBS Controls 255
IF . . . Then . . . Else . . . End If 256
Summary of If . . . . Then . . . . Else 257
Select Case 258
Summary of Select Case . . . . End Select 258
Do . . . Loop 258
While . . . Wend 259
For . . . Next 259
VBS keywords and functions 260
Other VBS Words 261
Functions 262
Hints and Tips for VBScripting 263
Errors 264
Error Handling 266
Speed of Execution 266
Scripting Ideas 267
Debugging Scripts 267
Horizontal Control Variables 268
Vertical Control Variables 268
List of Variables Available to Scripts 268
Communicating with Excel from a VBScript 269
Calling MATLAB from the Instrument 270
How to Select a Waveform Function Call 271
The MATLAB Waveform Control Panel 272
MATLAB Waveform Function Editor - Example 272
MATLAB Example Waveform Plot 274
How to Select a MATLAB Parameter Call 274
The MATLAB Parameter Control Panel 275
The MATLAB Parameter Editor 275
MATLAB Example Parameter Panel 276
More Examples of MATLAB Waveform Functions 277
Creating Your Own MATLAB Function 280
CustomDSO 281
What is CustomDSO? 281
Invoking CustomDSO 281
CustomDSO Basic Mode 282
Editing a CustomDSO Setup File 282

Creating a CustomDSO Setup File 283
CustomDSO PlugIn Mode 284
Creating a CustomDSO PlugIn 284
Properties of the Control and its Objects 285
Removing a PlugIn 288
PlugIn Example 1: Exchanging Two Traces on the Grids 289
Second Example PlugIn: Log-Log FFT Plot 291
Control Variables in CustomDSO 293
LabNotebook 293
Introduction to LabNotebook 293
Preferences 293
Miscellaneous Settings 293
Hardcopy Setup 294
E-mail Setup 294
Creating a Notebook Entry 294
Recalling Notebook Entries 298
Creating a Report 299
Previewing a Report 299
Locating a Notebook Entry 299
Creating the Report 299
Formatting the Report 300
Managing Notebook Entry Data 301
Adding Annotations 301
Deleting Notebook Entries 301
Saving Notebook Entries to a Folder 301
Managing the Database 301
Starting a New Database 302
Processing Web Option 302
Using the Web Editor 303
Adding Parameters 305
Adding Previews 305
Exiting the Web Editor 305
Viewing the Output 306
Serial Data Analyzer 306
Serial Data Analyzer Standard and Optional Capabilities 306
SDA Capabilities 306
SDM Capabilities 307
ASDA-J Capabilities 307

Jitter Wizard 309
SDA Basic Setup 312
PLL Setup 315
Summary 317
Mask Test 317
Eye Setup 318
Mask Margin 319
Testing 320
Bit Error Rate - ASDA-J Option Only 320
Serial Trigger 321
Serial Trigger Setup 322
PLL Locking 323
Setting the Data Pattern 323
Storing and Recalling Serial Data Patterns into the Trigger 325
Using the Serial Pattern Trigger 325
Jitter Setup 325
SDA DBI Controls 329
SDA 13000/11000/9000 DBI Controls 329
Vertical Noise Calibration 330
Running the Script 330
Using Noise Compensation 331
Jitter Measurements 332
Pj Breakdown 333
Alternate Jitter Breakdown Methods (option ASDA-J only) 333
Effective Jitter 333
MJSQ Jitter 335
Bathtub Curve 336
Jitter Filter 337
TIE Histogram 337
DDJ (Synchronous N-Cycle Plot) 338
DDj (ISI plot) – ASDA-J Option Only 339
Edge-to-Edge Basic Setup 341
Edge-Edge Jitter Measurements 342
Edge-Edge Jitter Measurement Controls 345
Adjust Rj 345
Async N Cycle Plot 345
Edge-Edge Spacing Controls 345

SDA Function Reference 346
AltNcycle 347
AltNcycle Control Summary 348
Htie to BER 350
Jitter Filter Function 352
Slice2Persist 353
Multi-Eye Measurements 355
Overview of Multi-Eye Measurement Tools 355
Setup and Installation 356
Multi-Eye Setup and Installation 356
Example Setups 357
Front Side Bus (FSB) 359
Introduction to FSB 359
FSB Theory of Operation 360
Transition/Non-Transition Eye Diagram 363
Gated (Qualified) Eye Diagram 364
SDA Theory 365
Clock recovery 365
Eye Pattern 368
Eye Violation Locator (ASDA Option) 369
Eye Pattern Measurements 370
Eye Amplitude 370
Eye Height 371
Eye Width 371
Extinction Ratio 371
Eye Crossing 371
Average Power 371
Q Factor or BER 372
eyeBER 372
Jitter Measurement 373
Bit Error Rate and Jitter 375
Total Jitter 375
Extrapolating the PDF 377
Separating Rj and Dj – Two Methods 379
Effective Random and Deterministic Jitter 379
Direct Measurement of Deterministic Jitter 380
Comparing Models 382
Bit Error Rate 383
Bit Error Map 383

Vertical Noise Compensation 385
Introduction 385
Signal Slew Rate and How Vertical Noise Converts to Jitter 385
The Relative Impact of Noise on Jitter (Quadratic Addition of Noise) 386
How the Measurement System Noise Is Subtracted from the Jitter Measurement 386
Q-scale Theory 386
Introduction 386
Interpretation of TIE Histogram – the Distribution of Edge Transition Times vs. Ideal (Expected) Transition Times 387
Relationship between Histograms and PDF 387
Integrating the PDFs 388
Extrapolation of the Distribution Tails (Extremes) 389
The Error Function erf(x), Inverse Error Function $erf^{-1}(x)$ and Related Functions 389
The Relationship between the Inverse Error Function and Total Jitter 390
Application of Error Function to Measured Jitter CDF (on Q-scale) 391
Automatic Renormalization of the Q-scale 393
Obtaining Deterministic and Random (Gaussian) Components from the Normalized Q-scale Diagram 396

INTRODUCTION

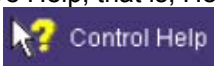
How to Use Online Help

Type Styles

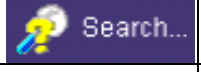
Activators of pop-up text and images appear as green, underlined, italic: *Pop-up*. To close pop-up text and images after opening them, touch the pop-up text again.
Link text appears blue and underlined: Link. Links jump you to other topics, URLs, or images; or to another location within the same Help window. After making a jump, you can touch the **Back** 

icon in the toolbar at the top of the Help window to return to the Help screen you just left. With each touch of the **Back** icon, you return to the preceding Help screen.

Instrument Help

When you press the front panel **HELP** button (if available), or touch the on-screen **Help** button, you will be presented with a menu: you can choose either to have information found for you automatically or to search for information yourself.
If you want context-sensitive Help, that is, Help related to what was displayed on the screen when you requested Help, touch 

in the drop-down menu, then touch the on-screen control (or front panel button or knob) that you need information about. The instrument will automatically display Help about that control.
If you want information about something not displayed on the screen, touch one of the buttons inside the drop-down menu to display the online Help manual:

	**Contents** displays the Table of Contents.
	**Index** displays an alphabetical listing of keywords.
	**Search** locates every occurrence of the keyword that you enter.
	**www.lecroy.com** connects you to LeCroy's Web site where you can find Lab Briefs, Application Notes, and other useful information. This feature requires that the instrument be connected to the internet through the Ethernet port on the scope's rear panel. Refer to Remote Communication for setup instructions.
	**About** opens the Utilities "Status" dialog, which shows software version and other system information.

Once opened, the Help window will display its navigation pane: the part of the window that shows the Table of Contents and Index. When you touch anywhere outside of the Help window, this navigation pane will disappear to reveal more of your signal. To make it return, touch the **Show** 

icon at the top of the Help window or touch anywhere inside the Help information pane.

Windows Help

In addition to instrument Help, you can also access online Help for Microsoft® Windows®. This help is accessible by minimizing the scope application, then touching the **Start** button in the Windows task bar at the bottom of the screen and selecting **Help**.

Returning a Product for Service or Repair

If you need to return a LeCroy product, identify it by its model and serial numbers. Describe the defect or failure, and give us your name and telephone number.
For factory returns, use a Return Authorization Number (RAN), which you can get from customer service. Write the number clearly on the outside of the shipping carton.
Return products requiring only maintenance to your local customer service center.
If you need to return your scope for any reason, use the original shipping carton. If this is not possible, be sure to use a rigid carton. The scope should be packed so that it is surrounded by a minimum of four inches (10 cm) of shock absorbent material.
Within the warranty period, transportation charges to the factory will be your responsibility. Products under warranty will be returned to you with transport prepaid by LeCroy. Outside the warranty period, you will have to provide us with a purchase order number before the work can be done. You will be billed for parts and labor related to the repair work, as well as for shipping.
You should prepay return shipments. LeCroy cannot accept COD (Cash On Delivery) or Collect Return shipments. We recommend using air freight.

Technical Support

You can get assistance with installation, calibration, and a full range of software applications from your customer service center. Visit the LeCroy Web site at www.lecroy.com or call 1-800-553-2769 for the center nearest you.

Staying Up-to-Date

To maintain your instrument's performance within specifications, have us calibrate it at least once a year. LeCroy offers state-of-the-art performance by continually refining and improving the instrument's capabilities and operation. We frequently update both firmware and software during service, free of charge during warranty.
You can also install new purchased software options in your scope yourself, without having to return it to the factory. Simply provide us with your instrument serial number and ID, and the version number of instrument software installed. We will provide you with a unique option key that consists of a code to be entered through the Utilities' Options dialog to load the software option.

Warranty

The instrument is warranted for normal use and operation, within specifications, for a period of three years from shipment. LeCroy will either repair or, at our option, replace any product returned to one of our authorized service centers within this period. However, in order to do this we must first examine the product and find that it is defective due to workmanship or materials and not due to misuse, neglect, accident, or abnormal conditions or operation.
LeCroy shall not be responsible for any defect, damage, or failure caused by any of the following: a) attempted repairs or installations by personnel other than LeCroy representatives, or b) improper connection to incompatible equipment or c) for any damage or malfunction caused by the use of non-LeCroy supplies. Furthermore, LeCroy shall not be obligated to service a product that has been modified or integrated where the modification or integration increases the task duration or difficulty of servicing the oscilloscope. Spare and replacement parts, and repairs, all have a 90-day warranty.
The oscilloscope's firmware has been thoroughly tested and is presumed to be functional. Nevertheless, it is supplied without warranty of any kind covering detailed performance. Products not made by LeCroy are covered solely by the warranty of the original equipment manufacturer.

Windows License Agreement

LeCroy's agreement with Microsoft prohibits users from running software on LeCroy X-Stream oscilloscopes that is not relevant to measuring, analyzing, or documenting waveforms.

End-user License Agreement For LeCroy X-Stream Software

IMPORTANT-READ CAREFULLY: THIS END-USER LICENSE AGREEMENT ("EULA") IS A LEGAL AGREEMENT BETWEEN THE INDIVIDUAL OR ENTITY LICENSING THE SOFTWARE PRODUCT ("YOU" OR "YOUR") AND LECROY CORPORATION ("LECROY") FOR THE SOFTWARE PRODUCT(S) ACCOMPANYING THIS EULA, WHICH INCLUDE(S): COMPUTER PROGRAMS; ANY "ONLINE" OR ELECTRONIC DOCUMENTATION AND PRINTED MATERIALS PROVIDED BY LECROY HEREWITH ("DOCUMENTATION"); ASSOCIATED MEDIA; AND ANY UPDATES (AS DEFINED BELOW) (COLLECTIVELY, THE "SOFTWARE PRODUCT"). BY USING AN INSTRUMENT TOGETHER WITH OR CONTAINING THE SOFTWARE PRODUCT, OR BY INSTALLING, COPYING, OR OTHERWISE USING THE SOFTWARE PRODUCT, IN WHOLE OR IN PART, YOU AGREE TO BE BOUND BY THE TERMS OF THIS EULA. IF YOU DO NOT AGREE TO THE TERMS OF THIS EULA, DO NOT INSTALL, COPY, OR OTHERWISE USE THE SOFTWARE PRODUCT; YOU MAY RETURN THE SOFTWARE PRODUCT TO YOUR PLACE OF PURCHASE FOR A FULL REFUND. IN ADDITION, BY INSTALLING, COPYING, OR OTHERWISE USING ANY MODIFICATIONS, ENHANCEMENTS, NEW VERSIONS, BUG FIXES, OR OTHER COMPONENTS OF THE SOFTWARE PRODUCT THAT LECROY PROVIDES TO YOU SEPARATELY AS PART OF THE SOFTWARE PRODUCT ("UPDATES"), YOU AGREE TO BE BOUND BY ANY ADDITIONAL LICENSE TERMS THAT ACCOMPANY SUCH UPDATES. IF YOU DO NOT AGREE TO SUCH ADDITIONAL LICENSE TERMS, YOU MAY NOT INSTALL, COPY, OR OTHERWISE USE SUCH UPDATES.

THE PARTIES CONFIRM THAT THIS AGREEMENT AND ALL RELATED DOCUMENTATION ARE AND WILL BE DRAFTED IN ENGLISH. LES PARTIES AUX PRÉSENTÉS CONFIRMENT LEUR VOLONTÉ QUE CETTE CONVENTION DE MÊME QUE TOUS LES DOCUMENTS Y COMPRIS TOUT AVIS QUI S'Y RATTACHÉ, SOIENT REDIGÉS EN LANGUE ANGLAISE.

1. GRANT OF LICENSE.

1.1 License Grant. Subject to the terms and conditions of this EULA and payment of all applicable fees, LeCroy grants to you a nonexclusive, nontransferable license (the "License") to: (a) operate the Software Product as provided or installed, in object code form, for your own internal business purposes, (i) for use in or with an instrument provided or manufactured by LeCroy (an "Instrument"), (ii) for testing your software product(s) (to be used solely by you) that are designed to operate in conjunction with an Instrument ("Your Software"), and (iii) make one copy for archival and back-up purposes; (b) make and use copies of the Documentation; provided that such copies will be used only in connection with your licensed use of the Software Product, and such copies may not be republished or distributed (either in hard copy or electronic form) to any third party; and (c) copy, modify, enhance and prepare derivative works ("Derivatives") of the source code version of those portions of the Software Product set forth in and identified in the Documentation as "Samples" ("Sample Code") for the sole purposes of designing, developing, and testing Your Software. If you are an entity, only one designated individual within your organization, as designated by you, may exercise the License; provided that additional individuals within your organization may assist with respect to reproducing and distributing Sample Code as permitted under Section 1.1(c)(ii). LeCroy reserves all rights not expressly granted to you. No license is granted hereunder for any use other than that specified herein, and no license is granted for any use in combination or in connection with other products or services (other than Instruments and Your Software) without the express prior written consent of LeCroy. The

Software Product is licensed as a single product. Its component parts may not be separated for use by more than one user. This EULA does not grant you any rights in connection with any trademarks or service marks of LeCroy. The Software Product is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties. The Software Product is licensed, not sold. The terms of this printed, paper EULA supersede the terms of any on-screen license agreement found within the Software Product.

1.2 Upgrades. If the Software Product is labeled as an "upgrade," (or other similar designation) the License will not take effect, and you will have no right to use or access the Software Product unless you are properly licensed to use a product identified by LeCroy as being eligible for the upgrade ("Underlying Product"). A Software Product labeled as an "upgrade" replaces and/or supplements the Underlying Product. You may use the resulting upgraded product only in accordance with the terms of this EULA. If the Software Product is an upgrade of a component of a package of software programs that you licensed as a single product, the Software Product may be used and transferred only as part of that single product package and may not be separated for use on more than one computer.

1.3. Limitations. Except as specifically permitted in this EULA, you will not directly or indirectly (a) use any Confidential Information to create any software or documentation that is similar to any of the Software Product or Documentation; (b) encumber, transfer, rent, lease, time-share or use the Software Product in any service bureau arrangement; (c) copy (except for archival purposes), distribute, manufacture, adapt, create derivative works of, translate, localize, port or otherwise modify the Software Product or the Documentation; (d) permit access to the Software Product by any party developing, marketing or planning to develop or market any product having functionality similar to or competitive with the Software Product; (e) publish benchmark results relating to the Software Product, nor disclose Software Product features, errors or bugs to third parties; or (f) permit any third party to engage in any of the acts proscribed in clauses (a) through (e). In jurisdictions in which transfer is permitted, notwithstanding the foregoing prohibition, transfers will only be effective if you transfer a copy of this EULA, as well as all copies of the Software Product, whereupon your right to use the Software product will terminate. Except as described in this Section 1.3, You are not permitted (i) to decompile, disassemble, reverse compile, reverse assemble, reverse translate or otherwise reverse engineer the Software Product, (ii) to use any similar means to discover the source code of the Software Product or to discover the trade secrets in the Software Product, or (iii) to otherwise circumvent any technological measure that controls access to the Software Product. You may reverse engineer or otherwise circumvent the technological measures protecting the Software Product for the sole purpose of identifying and analyzing those elements that are necessary to achieve Interoperability (the "Permitted Objective") only if: (A) doing so is necessary to achieve the Permitted Objective and it does not constitute infringement under Title 17 of the United States Code; (B) such circumvention is confined to those parts of the Software Product and to such acts as are necessary to achieve the Permitted Objective; (C) the information to be gained thereby has not already been made readily available to you or has not been provided by LeCroy within a reasonable time after a written request by you to LeCroy to provide such information; (D) the information gained is not used for any purpose other than the Permitted Objective and is not disclosed to any other person except as may be necessary to achieve the Permitted Objective; and (E) the information obtained is not used (1) to create a computer program substantially similar in its expression to the Software Product including, but not limited to, expressions of the Software Product in other computer languages, or (2) for any other act restricted by LeCroy's intellectual property rights in the

Software Product. "Interoperability" will have the same meaning in this EULA as defined in the Digital Millennium Copyright Act, 17 U.S.C. §1201(f), the ability of computer programs to exchange information and of such programs mutually to use the information which has been exchanged.

1.4 PRERELEASE CODE. Portions of the Software Product may be identified as prerelease code ("Prerelease Code"). Prerelease Code is not at the level of performance and compatibility of the final, generally available product offering. The Prerelease Code may not operate correctly and may be substantially modified prior to first commercial shipment. LeCroy is not obligated to make this or any later version of the Prerelease Code commercially available. The License with respect to the Prerelease Code terminates upon availability of a commercial release of the Prerelease Code from LeCroy.

2. SUPPORT SERVICES.

At LeCroy's sole discretion, from time to time, LeCroy may provide Updates to the Software Product. LeCroy shall have no obligation to revise or update the Software Product or to support any version of the Software Product. At LeCroy's sole discretion, upon your request, LeCroy may provide you with support services related to the Software Product ("Support Services") pursuant to the LeCroy policies and programs described in the Documentation or otherwise then in effect, and such Support Services will be subject to LeCroy's then-current fees therefor, if any. Any Update or other supplemental software code provided to you pursuant to the Support Services will be considered part of the Software Product and will be subject to the terms and conditions of this EULA. LeCroy may use any technical information you provide to LeCroy during LeCroy's provision of Support Services, for LeCroy's business purposes, including for product support and development. LeCroy will not utilize such technical information in a form that personally identifies you.

3. PROPRIETARY RIGHTS.

3.1 Right and Title. All right, title and interest in and to the Software Product and Documentation (including but not limited to any intellectual property or other proprietary rights, images, icons, photographs, text, and "applets" embodied in or incorporated into the Software Product, collectively, "Content"), and all Derivatives, and any copies thereof are owned by LeCroy and/or its licensors or third-party suppliers, and is protected by applicable copyright or other intellectual property laws and treaties. You will not take any action inconsistent with such title and ownership. This EULA grants you no rights to use such Content outside of the proper exercise of the license granted hereunder, and LeCroy will not be responsible or liable therefor.

3.2 Intellectual Property Protection. You may not alter or remove any printed or on-screen copyright, trade secret, proprietary or other legal notices contained on or in copies of the Software Product or Documentation.

3.3 Confidentiality. Except for the specific rights granted by this EULA, neither party shall use or disclose any Confidential Information (as defined below) of the other party without the written consent of the disclosing party. A party receiving Confidential Information from the other shall use the highest commercially reasonable degree of care to protect the Confidential Information, including ensuring that its employees and consultants with access to such Confidential Information have agreed in writing not to disclose the Confidential Information.

You shall bear the responsibility for any breaches of confidentiality by your employees and consultants. Within ten (10) days after request of the disclosing party, and in the disclosing party's sole discretion, the receiving party shall either return to the disclosing party originals and copies of any Confidential Information and all information, records and materials developed therefrom by the receiving party, or destroy the same, other than such Confidential Information as to which this EULA expressly provides a continuing right to the receiving party to retain at the time of the request. Either party may only disclose the general nature, but not the specific financial terms, of this EULA without the prior consent of the other party; provided either party may provide a copy of this EULA to any finance provider in conjunction with a financing transaction, if such provider agrees to keep this EULA confidential. Nothing herein shall prevent a receiving party from disclosing all or part of the Confidential Information as necessary pursuant to the lawful requirement of a governmental agency or when disclosure is required by operation of law; provided that prior to any such disclosure, the receiving party shall use reasonable efforts to (a) promptly notify the disclosing party in writing of such requirement to disclose, and (b) cooperate fully with the disclosing party in protecting against any such disclosure or obtaining a protective order. Money damages will not be an adequate remedy if this Section 4.3 is breached and, therefore, either party shall, in addition to any other legal or equitable remedies, be entitled to seek an injunction or similar equitable relief against such breach or threatened breach without the necessity of posting any bond. As used herein, “Confidential Information” means LeCroy pricing or information concerning new LeCroy products, trade secrets (including without limitation all internal header information contained in or created by the Software Product, all benchmark and performance test results and all Documentation) and other proprietary information of LeCroy; and any business, marketing or technical information disclosed by LeCroy, or its representatives, or you in relation to this EULA, and either (i) disclosed in writing and marked as confidential at the time of disclosure or (ii) disclosed in any other manner such that a reasonable person would understand the nature and confidentiality of the information. Confidential Information does not include information (A) already in the possession of the receiving party without an obligation of confidentiality to the disclosing party, (B) hereafter rightfully furnished to the receiving party by a third party without a breach of any separate nondisclosure obligation to the disclosing party, (C) publicly known without breach of this EULA, (d) furnished by the disclosing party to a third party without restriction on subsequent disclosure, or (e) independently developed by the receiving party without reference to or reliance on the Confidential Information.

4. TERMINATION.

This EULA will remain in force until termination pursuant to the terms hereof. You may terminate this EULA at any time. This EULA will also terminate if you breach any of the terms or conditions of this EULA. You agree that if this EULA terminates for any reason, the License will immediately terminate and you will destroy all copies of the Software Product (and all Derivatives), installed or otherwise, the Documentation, and the Confidential Information (and all derivatives of any of the foregoing) that are in your possession or under your control. The provisions of Sections 1.3, 4, 6, 7, 8, and 9 will survive any termination or expiration hereof.

5. U.S. GOVERNMENT RESTRICTED RIGHTS.

If any Software Product or Documentation is acquired by or on behalf of a unit or agency of the United States Government (any such unit or agency, the "Government"), the Government agrees that the Software Product or Documentation is "commercial computer software" or "commercial computer software documentation" and that, absent a written agreement to the contrary, the Government's rights with respect to the Software Product or Documentation are, in the case of civilian agency use, Restricted Rights, as defined in FAR §52.227.19, and if for Department of Defense use, limited by the terms of this EULA, pursuant to DFARS §227.7202. The use of the Software Product or Documentation by the Government constitutes acknowledgment of LeCroy's proprietary rights in the Software Product and Documentation. Manufacturer is LeCroy Corporation, 700 Chestnut Ridge Road, Chestnut Ridge, NY 10977 USA.

6. EXPORT RESTRICTIONS.

You agree that you will not export or re-export the Software Product, any part thereof, or any process or service that is the direct product of the Software Product (the foregoing collectively referred to as the "Restricted Components"), to any country, person, entity or end user subject to U.S. export restrictions. You specifically agree not to export or re-export any of the Restricted Components (a) to any country to which the U.S. has embargoed or restricted the export of goods or services, which currently include, but are not necessarily limited to Cuba, Iran, Iraq, Libya, North Korea, Sudan and Syria, or to any national of any such country, wherever located, who intends to transmit or transport the Restricted Components back to such country; (b) to any end user who you know or have reason to know will utilize the Restricted Components in the design, development or production of nuclear, chemical or biological weapons; or (c) to any end-user who has been prohibited from participating in U.S. export transactions by any federal agency of the U.S. government. You warrant and represent that neither the BXA nor any other U.S. federal agency has suspended, revoked or denied your export privileges. It is your responsibility to comply with the latest United States export regulations, and you will defend and indemnify LeCroy from and against any damages, fines, penalties, assessments, liabilities, costs and expenses (including reasonable attorneys' fees and court costs) arising out of any claim that the Software Product, Documentation, or other information or materials provided by LeCroy hereunder were exported or otherwise accessed, shipped or transported in violation of applicable laws and regulations.

7. RISK ALLOCATION.

7.1 No Warranty. THE SOFTWARE PRODUCT IS NOT ERROR-FREE AND THE SOFTWARE PRODUCT AND SUPPORT SERVICES IS/ARE BEING PROVIDED "AS IS" WITHOUT WARRANTY OF ANY KIND. LECROY, FOR ITSELF AND ITS SUPPLIERS, HEREBY DISCLAIMS ALL WARRANTIES, WHETHER EXPRESS OR IMPLIED, ORAL OR WRITTEN, WITH RESPECT TO THE SOFTWARE PRODUCT OR ANY SUPPORT SERVICES INCLUDING, WITHOUT LIMITATION, ALL IMPLIED WARRANTIES OF TITLE OR NON-INFRINGEMENT, MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, ACCURACY, INTEGRATION, VALIDITY, EXCLUSIVITY, MERCHANTABILITY, NON-INTERFERENCE WITH ENJOYMENT, FITNESS FOR ANY PARTICULAR PURPOSE, AND ALL WARRANTIES IMPLIED FROM ANY COURSE OF DEALING OR USAGE OF TRADE. YOU ACKNOWLEDGE THAT NO WARRANTIES HAVE BEEN MADE TO YOU BY OR ON BEHALF OF LECROY OR OTHERWISE FORM THE BASIS FOR THE BARGAIN BETWEEN THE PARTIES.

7.2. Limitation of Liability. LECROY'S LIABILITY FOR DAMAGES FOR ANY CAUSE WHATSOEVER, REGARDLESS OF THE FORM OF ANY CLAIM OR ACTION, SHALL NOT EXCEED THE GREATER OF THE AMOUNT ACTUALLY PAID BY YOU FOR THE SOFTWARE PRODUCT OR U.S.$5.00; PROVIDED THAT IF YOU HAVE ENTERED INTO A SUPPORT SERVICES AGREEMENT WITH LECROY, LECROY'S ENTIRE LIABILITY REGARDING SUPPORT SERVICES WILL BE GOVERNED BY THE TERMS OF THAT AGREEMENT. LECROY SHALL NOT BE LIABLE FOR ANY LOSS OF PROFITS, LOSS OF USE, LOSS OF DATA, INTERRUPTION OF BUSINESS, NOR FOR INDIRECT, SPECIAL, INCIDENTAL, CONSEQUENTIAL OR EXEMPLARY DAMAGES OF ANY KIND, WHETHER UNDER THIS EULA OR OTHERWISE ARISING IN ANY WAY IN CONNECTION WITH THE SOFTWARE PRODUCT, THE DOCUMENTATION OR THIS EULA. SOME JURISDICTIONS DO NOT ALLOW THE EXCLUSION OR LIMITATION OF INCIDENTAL OR CONSEQUENTIAL DAMAGES, SO THE ABOVE EXCLUSION OR LIMITATION MAY NOT APPLY TO YOU. THESE LIMITATIONS ARE INDEPENDENT FROM ALL OTHER PROVISIONS OF THIS EULA AND SHALL APPLY NOTWITHSTANDING THE FAILURE OF ANY REMEDY PROVIDED HEREIN.

7.3 Indemnification. You will defend, indemnify and hold harmless LeCroy and its officers, directors, affiliates, contractors, agents, and employees from, against and in respect of any and all assessments, damages, deficiencies, judgments, losses, obligations and liabilities (including costs of collection and reasonable attorneys' fees, expert witness fees and expenses) imposed upon or suffered or incurred by them arising from or related to your use of the Software Product.

8. GENERAL PROVISIONS.

8.1 Compliance with Laws. You will comply with all laws, legislation, rules, regulations, and governmental requirements with respect to the Software Product, and the performance by you of your obligations hereunder, of any jurisdiction in or from which you directly or indirectly cause the Software Product to be used or accessed.

8.2 No Agency. Nothing contained in this EULA will be deemed to constitute either party as the agent or representative of the other party, or both parties as joint venturers or partners for any purpose.

8.3 Entire Agreement; Waiver; Severability. This EULA constitutes the entire agreement between the parties with regard to the subject matter hereof. No provision of, right, power or privilege under this EULA will be deemed to have been waived by any act, delay, omission or acquiescence by LeCroy, its agents, or employees, but only by an instrument in writing signed by an authorized officer of LeCroy. No waiver by LeCroy of any breach or default of any provision of this EULA by you will be effective as to any other breach or default, whether of the same or any other provision and whether occurring prior to, concurrent with, or subsequent to the date of such waiver. If any provision of this EULA is declared by a court of competent jurisdiction to be invalid, illegal or unenforceable, such provision will be severed from this EULA and all the other provisions will remain in full force and effect.

8.4 Governing Law; Jurisdiction; Venue. This EULA will be governed by and construed in accordance with the laws of the State of New York, USA, without regard to its choice of law provisions. The United Nations Convention on Contracts for the International Sale of Goods will not apply to this EULA. Exclusive jurisdiction and venue for any litigation arising under this EULA is in the federal and state courts located in New York, New York, USA and both parties hereby consent to such jurisdiction and venue for this purpose.

8.5 Assignment. This EULA and the rights and obligations hereunder, may not be assigned, in whole or in part by you, except to a successor to the whole of your business, without the prior written consent of LeCroy. In the case of any permitted assignment or transfer of or under this EULA, this EULA or the relevant provisions will be binding upon, and inure to the benefit of, the successors, executors, heirs, representatives, administrators and assigns of the parties hereto.

8.6 Notices. All notices or other communications between LeCroy and you under this EULA will be in writing and delivered personally, sent by confirmed fax, by confirmed e-mail, by certified mail, postage prepaid and return receipt requested, or by a nationally recognized express delivery service. All notices will be in English and will be effective upon receipt.

8.7 Headings. The headings used in this EULA are intended for convenience only and will not be deemed to supersede or modify any provisions.

8.8 Acknowledgment. Licensee acknowledges that (a) it has read and understands this EULA, (b) it has had an opportunity to have its legal counsel review this EULA, (c) this EULA has the same force and effect as a signed agreement, and (d) issuance of this EULA does not constitute general publication of the Software Product or other Confidential Information.

Virus Protection

Because your scope runs on a Windows-based PC platform, it must be protected from viruses, as with any PC on a corporate network. It is crucial that the scope be kept up to date with Windows Critical Updates, and that anti-virus software be installed and continually updated.
Visit www.lecroy.com/dsosecurity for more information regarding Windows Service Pack compatibility with LeCroy operating software, and related matters.

SDA SPECIFICATIONS

Note: Specifications are subject to change without notice.

Vertical System

	SDA 6000/6000A/6020	SDA 5000/5000A	SDA 4000A/4020	SDA 3010
Analog Bandwitdth at 50 Ω (-3 dB)	6 GHz	5 GHz	4 GHz	3 GHz
Rise Time (10-90% Typical)	75 ps	90 ps	105 ps	150 ps
Rise Time (20-80% Typical)	NA	NA	NA	NA
Number of Input Channels	4	4	4	4
Bandwidth Limiters	20 MHz, 200 MHz, 1 GHz, 3 GHz, 4 GHz	20 MHz, 200 MHz, 1 GHz, 3 GHz, 4 GHz	20 MHz, 200 MHz, 1 GHz, 3 GHz	20 MHz, 200 MHz, 1 GHz

	SDA 18000			SDA 13000		SDA 11000		SDA 9000	
Analog Bandwidth at 50 Ω (-3 dB)	18 GHz	11 GHz	6 GHz	13 GHz	6 GHz	11 GHz	6 GHz	9 GHz	6 GHz
Rise Time (10-90% Typical)	27 ps	40 ps	75 ps	35 ps	75 ps	40 ps	75 ps	49 ps	75 ps
Rise Time (20-80% Typical)	19 ps	28 ps	NA	25 ps	NA	28 ps	NA	NA	NA
Number of Input Channels	1	1	4, 2, or 1	2 or 1	4 or 2 or 1	2 or 1	4 or 2	2 or 1	4 or 2
Bandwidth Limiters	Full	Full	20 MHz, 200 MHz, 1 GHz, 3 GHz, 4 GHz	Full	20 MHz, 200 MHz, 1 GHz, 3 GHz, 4 GHz	Full	20 MHz, 200 MHz, 1 GHz, 3 GHz, 4 GHz	Full	20 MHz, 200 MHz, 1 GHz, 3 GHz, 4 GHz

- **Input Impedance:** 50 Ω ±2%
- **Input Coupling:** DC, GND (SDA 3010: DC, GND, DC 1MΩ, AC 1MΩ)
- **Maximum Input:** ±4 V_{peak} (±5 V_{rms})
- **Vertical Resolution:** 8 bits; up to 11 bits with enhanced resolution (ERES)
- **Sensitivity:** 2 mV to 1 V/div, < 10 mV/div through zoom
- **DC Gain Accuracy:** ±1.5% of full-scale
- **Offset Range (excluding SDA 3010):** ±750 mV (or more) @ 2 mV to 195 mV/div; ±4 V @ 195 mV to 1 V/div
- **Offset Range (SDA 3010):** ±700 mV @ 2-4.99 mV/div, ±1.5 V @ 5-100 mV/div, ±10 V @ 0.102-1 V/div (@ 50 Ω), ±20 V @ 0.102-2 V/div (@ 1 MΩ)

Horizontal System

- **Timebases:** Internal timebase common to 4 input channels; an external clock may be applied at the Auxiliary Input (SDA 6020 only)
- **Math & Zoom Traces:** 8 math/zoom traces standard
- **Clock Accuracy: ≤** 1 ppm @ 0 to 40 degrees C (SDA 3010 ≤ 10 ppm @ 0 to 40 degrees C)
- **Time Interval Accuracy:** ≤ 0.06/SR + (1 ppm x Reading) (rms);
  SDA 3010: ≤ 0.06/SR + (10 ppm x Reading) (rms)
- **Sample Rate + Delay Time Accuracy:** ±1 ppm @ 25 degrees C (SDA 3010: ±5 ppm ≤ 10 s interval)

Jitter Noise Floor

SDA 18000			SDA 13000/11000		SDA 9000	
18 GHz/Ch Mode	**11 GHz/Ch Mode**	**6 GHz/Ch Mode**	**13/11 GHz/Ch Mode**	**6 GHz/Ch Mode**	**9 GHz/Ch Mode**	**6 GHz/Ch mode**
< 350 fs rms measured with 35 ps rise time (typical)		1 ps rms (typical)	< 350 fs rms measured with 35 ps rise time (typical)	1 ps rms (typical)	< 350 fs rms measured with 35 ps rise time (typical)	1 ps rms (typical)

- **SDA 6020, 6000x, 5000x, 4020, 4000x, 3010:** 1 ps rms (typical)
- **External Clock Frequency:** 30 MHz to 2 GHz / 50 Ω impedance applied at the auxiliary input (except SDA X000A) 30 MHz to 1 GHz / 50 Ω impedance applied at the auxiliary input (SDA 3010)
- **External Reference Clock Frequency:** 100 MHz / 50 Ω (except SDA 11000 and 9000)
- **Trigger and Interpolator Jitter:** ≤ 2.5 ps rms
- **Time per Division Range:** 20 ps to 1000 s/div (10 s/div in Auto-trigger mode)

SDA 18000			SDA 13000/11000	
18 GHz/Ch Mode	**11 GHz/Ch Mode**	**6 GHz/Ch Mode**	**13/11 GHz/Ch Mode**	**6 GHz/Ch Mode**
10 ps/div to 50 µs/div (Std. memory) 10 ps/div to 500 µs/div (-XL memory)		20 ps/div to 10 s/div	10 ps/div to 50 µs/div (Std memory)	20 ps/div to 10 s/div (Std memory)

Acquisition System

Single-Shot Sample Rate/Ch: 10 GS/s (SDA 6020 and 4020: 20 GS/s)

SDA 18000			SDA 13000/11000		SDA 9000	
18 GHz/Ch Mode	**11 GHz/Ch Mode**	**6 GHz/Ch Mode**	**11 GHz/Ch Mode**	**6 GHz/Ch Mode**	**9 GHz/Ch Mode**	**6 GHz/Ch mode**
60 GS/s on 1 Ch	40 GS/s on 1 Ch	20 GS/s on 4, 2, or 1 Ch	40 GS/s on 4 or 2 Ch	20 GS/s on 2 or 1 Ch	40 GS/s on 2 or 1 Ch	20 GS/s on 4 or 2 Ch

Memory

- **SDA 6020/4020:** 20 Mpts/Ch. Option SDA-XL: 25 Mpts/Ch, 50 Mpts/Ch for 2 or 1
- **Ch.SDA6000A XXL/SDA4000A XXL:** 50 Mpts/Ch, 100 Mpts/Ch for 2 or 1 Ch
- **SDA3010:** 10 Mpts/Ch, 20 Mpts/Ch for 2 or 1 Ch. Option SDA-XL: 25 Mpts/Ch, 50 Mpts.Ch for 2 or 1 Ch.

SDA 18000/13000/11000/9000

	SDA 18000			SDA 13000/11000		SDA 9000	
	18 GHz/Ch Mode	11 GHz/Ch Mode	6 GHz/Ch Mode	13/11 GHz/Ch Mode	6 GHz/Ch Mode	9 GHz/Ch Mode	6 GHz/Ch mode
Channels	1	1	4	2	4	2	4
Std Memory	60M	40M	20M	40M	20M	40M	20M
XL Memory Option	150M	100M	50M	100M	50M	100M	50M

Acquisition Modes

- **Random Interleaved Sampling (RIS):** 200 GS/s for repetitive signals (SDA 18000/13000/11000/9000: RIS mode is available in 6 GHz mode only)
- **Single-Shot:** For transient and repetitive signals: 20 ps/div to 1000 s/div; (SDA 18000/13000/11000/9000 in 6 GHz mode only): 20 ps/div to 1 µs/div
- **Sequence:** 2 to 40,000 segments (number of segments depends upon memory options)
- **Intersegment Time:** 6 µs (typical)

Acquisition Processing

- **Averaging:** Summed averaging to 1 million sweeps; Continuous averaging to 1 million sweeps
- **Enhanced Resolution (ERES):** From 8.5 to 11 bits vertical resolution
- **Envelope (Extrema):** Envelope, floor, roof for up to 1 million sweeps

Triggering System

- **Modes:** Normal, Auto, Single, and Stop
- **Sources:** Any input channel, External, EXT X10, EXT/10, or line; slope and level unique to each source (except line trigger; disabled channels cannot be used as a trigger source in SDA 18000, SDA13000, SDA 11000, or SDA 9000 when operating in 18 GHz, 13 GHz, 11 GHz, or 9 GHz modes.)
- **Coupling Modes:** DC
- **Pre-trigger delay:** 0 to 100% of memory size (adjustable in 1% increments)
- **Post-trigger delay:** smaller of 0 to 10,000 divisions or 86400 seconds
- **Hold-off by time or events:** Up to 20 s or from 1 to 99,999,999 events
- **Internal trigger range:** ±4.1 div from center
- **Max. trigger frequency:** Up to 5 GHz with Edge Trigger; 750 MHz with SMART Trigger

-  **External trigger input range:** AUX (±0.4 V); AUX X10 (±0.04 V); AUX /10 (±4 V) Not available on SDA 6000A, SDA 5000A, SDA 4000A

Automatic Setup

- **Auto Setup:** Automatically sets timebase, trigger, and sensitivity to display a wide range of repetitive signals.
- **Vertical Find Scale:** Automatically sets vertical sensitivity and offset for the selected channels to display a waveform with maximum dynamic range.

Probes

- **Probes:** A variety of optional passive and active probes are available.
- **Probe System:** ProLink with ProBus automatically detects and supports a variety of compatible probes; Supports ProLink SMA or BNC input adapters.
- **Scale Factors:** Automatically or manually selected depending on probe used.

Basic Triggers

Edge/Slope/Line: Triggers when signal meets slope and level condition.

SMART Triggers

- **State or Edge Qualified:** Triggers on any input source only if a defined state or edge occurred on another input source. Delay between sources is selectable by time or events
- **Dropout:** Triggers if signal drops out for longer than selected time between 2 ns and 20 s
- **Pattern Logic:** Combination (AND, NAND, OR, NOR) of 5 inputs (maximum of 3 or 4 inputs; External Trigger not available on the SDA 6000A, SDA 5000A, and SDA 4000A; 4, 3, or 2 Ch + External Trigger on SDA 18000/13000/11000/9000). Each source can be high, low, or don't care. Triggers at the start or end of the pattern.

SMART Triggers with Exclusion Technology

- **Glitch:** Triggers on positive or negative glitches with widths selectable from 600 ps to 20 s or on intermittent faults.
- **Signal or Pattern Width:** Triggers on positive or negative pulse widths selectable from 600 ps to 20 s or on intermittent faults.
- **Signal or Pattern Interval:** Triggers on intervals selectable between 2 ns and 20 s.

Serial Trigger

Available on SDA 6000A (XXL), SDA 5000A (XXL), SDA 4000A (XXL)

Data Type	NRZ encoded for clock extraction (needs edge density > 20%)
Sensitivity	1 division minimum and at least 10 mV_{p-p}
Min. Frequency	50 MHz
Max. Frequency	2.7 GHz
Serial Trigger Length	up to 32 bits
Clock/Data Output Connector Type	SMA
Clock/Data Output	1/2 amplitude, AC coupled LVPCL
Clock/Data Output Voltage Swing (into 50 Ω)	400 mV typical
Clock /Data Output Rise/Fall Time	200 ps typical
Recovered CLK and DATA Jitter	0.015 UI rms typical
Recovered CLK and DATA Phase Relationship	Data is centered on the rising edge of the clock
Source	Channel 4 only
Trigger Sensitivity	1 division minimum and at least 10 mV_{p-p}

Clock Recovery

custom filter settings
damping factor
natural frequency
number of poles
Standard PLL settings
(FC, Golden, PCIe, DVI, Custom)

Jitter Analysis

advanced (peak-peak, rms)
basic (Tj, Rj, Dj)
bathtub curve
conventional
cycle-cycle jitter
Dj breakdown
edge to edge (data to data)
edge to reference (data to clock)
effective
filtered jitter
half-period jitter
ISI plot
jitter histogram
jitter wizard
MJSQ
period jitter
periodic jitter (Pj) with peak frequency listing
synchronous N-cycle with bit pattern display
TIE clock jitter
TIE jitter

Math Tools – Standard

Display up to eight math function traces (F1 to F8). The easy-to-use graphical interface simplifies setup of up to two operations on each function trace. Function traces can be chained together to perform math-on-math.

- absolute value
- average (summed)
- average (continuous)
- derivative
- deskew (resample)
- difference ()
- enhanced resolution (to 11 bits vertical)
- envelope
- exp (base e)
- exp (base 10)
- fft (power spectrum, magnitude phase up to 25 kpts)
- floor
- histogram of 1,000 events
- integral
- invert (negate)
- log (base e)
- log (base 10)
- product (x)
- ratio (/)
- reciprocal
- rescale (with units)
- roof
- (sin x)/x
- square
- square root
- sum (+)
- trend (datalog) of 1,000 events
- zoom (identity)

Automated Measure Tools – Standard

These standard tools allow the display of any 8 parameters together with statistics, including their average, high, low, and standard deviations. Histicons provide a fast, dynamic view of parameters and wave shape characteristics.

- amplitude
- area
- base
- cycles
- data
- delay
- delay
- duty cycle
- duration
- falltime (90-10%, 80-20%, @ level)
- frequency
- first
- last
- level@x
- maximum
- mean
- median
- minimum
- number of points
- +overshoot
- overshoot
- peak-to-peak
- period
- phase
- risetime (10-90%, 20-80%, @ level)
- rms
- std. deviation
- top
- width
- time @ minimum (min.)
- time @ maximum (max.)
- time @ level
- time @ level from trigger
- x @ max
- x @ min

Pass/Fail Testing

Simultaneously test multiple parameters against selectable parameter limits or pre-defined masks. Pass or fail conditions can initiate actions including document to local or networked files, email the image of the failure, save waveforms, send a pulse out of the front panel auxiliary BNC output, or (with the GPIB option) send a GPIB SRQ.

Advanced Math Package (XMATH) – Standard

This package provides a comprehensive set of signal WaveShape Analysis tools providing insight into the wave shape of complex signals. Additional capability provided by XMATH includes:

- 8 math traces total (4 additional)
- Parameter math add, subtract, multiply, or divide two different parameters
- Histograms expanded with 19 histogram parameters and up to 2 billion events
- Trend (datalog) of up to 20,000 events
- Track graphs of any measurement parameter
- FFT capability added to include: power averaging, power density, real and imaginary components, frequency domain parameters, and FFT on up to 25 Mpts
- Narrow-band power measurements
- Auto-correlation function
- Sparse function
- Cubic and Quadratic Interpolation

Jitter and Timing Analysis Package (JTA2) – Standard

This package provides jitter timing and analysis using time, frequency, and statistical views for common timing parameters, and also includes other useful tools.

- Jitter and timing parameters, with Track graphs for the following:

Cycle-Cycle Jitter	Period	Time Interval Error	Skew
N-Cycle	Half Period	Setup	Duty Cycle
w/ start selection Frequency	Width	Hold	Duty Cycle Error

- Edge@lv parameter (counts edges)
- Histograms expanded with 19 histogram parameters and up to 2 billion events
- Trend (datalog) of up to 20,000 events
- Track graphs of all parameters
- Persistence histogram, persistence trace (mean, range, sigma)

ASDA-J - Standard

This package includes:

- Edge-to-edge jitter
- Clock jitter
- Filtered jitter
- Effective and MJSQ jitter
- ISI plot of data dependent jitter
- N-cycle jitter plot
- Bit error rate analysis
- Mask violation

8B/10B Protocol Decoding - Standard

Simultaneously translates up to 4 lanes of 8B/10B encoded Serial Data waveforms into symbol views to allow easier troubleshooting. This allows you to quickly correlate protocol events with the physical serial data waveform. The decoder operates with 8B/10B encoded data at rates up to 6.25 Gb/s.

Advanced Customization Package (XDEV) – Optional

This package provides a set of tools to modify the analyzer and customize it to meet your unique needs. Additional capability provided by XDEV includes:

- The ability to create your own measurement parameter or math function using third-party software packages and display the result in the scope. Supported third-party software packages include: VBScript (Visual Basic), Excel, MATLAB, Mathcad
- CustomDSO creates your own user interface in a scope dialog box.
- Supports plug-ins

Color Waveform Display

- **Type:** Color 10.4" flat-panel TFT-LCD with high-resolution touch-screen
- **Resolution:** SVGA; 800 x 600 pixels
- **Realtime Clock:** Dates, hours, minutes, seconds displayed with waveform, SNTP support to synchronize to precision internet clocks
- **Number of Traces:** Display a maximum of 8 traces. Simultaneously display channel, zoom, memory, and math traces.
- **Grid Styles:** Auto, Single, Dual, Quad, Octal, XY, Single + XY, Dual + XY, or Auto
- **Waveform Styles:** Sample dots joined or dots only

Analog Persistence Display

- **Analog & Color-Graded Persistence:** Variable saturation levels; stores each traces persistence data in memory.
- **Persistence Selections:** Select analog, color, or three dimensional.
- **Trace Selection:** Activate Analog Persistence on all or any combination of traces.
- **Persistence Aging Time:** Select from 500 ms to infinity.
- **Sweeps Displayed:** All accumulated, or all accumulated with last trace, highlighted

Zoom Expansion Traces

Display up to 8 Math/Zoom traces.

CPU

- **Processor:** Intel® Pentium 4 @ 2.53 GHz (or better) with MS Windows® XP Platform
- **Processing Memory:** Up to 2 Gbytes

Internal Waveform Memory

M1, M2, M3, M4 Internal Waveform Memory (Store full-length waveforms with 16 bits/data point.) Or store to any number of files limited only by data storage media.

Setup Storage

Front Panel and Instrument Status Save to the internal hard drive, floppy drive or to a USB-connected peripheral device.

Interface

- **Remote Control:** Via Windows Automation, or via LeCroy GPIB command set
- **GPIB Port:** (Optional) Supports IEEE-488.2
- **Ethernet Port"** 10/100Base-T Ethernet interface
- **USB Ports:** 4 USB ports support Windows compatible devices.
- **External Monitor Port:** Standard 15-pin D-Type SVGA-compatible
- **Parallel Port:** 1 standard

Auxiliary Output

- **Signal Types:** Select from calibrator or control signals output on front panel.
- **Calibrator Signal:** 5 Hz to 5 MHz square wave or DC Level; 0.0 to +0.5 Volts into 50 Ω (0 to 1 V into 1 M), or TTL Volts (selectable)
- **Control Signals:** Trigger enabled, trigger out, pass/fail status

Auxiliary Input

Signal Types: Select from External Trigger or External Clock input on front panel.

General

- **Auto Calibration:** Ensures specified DC and timing accuracy is maintained for 1 year minimum.
- **Power Requirements:** 100 to 120 VAC at 50/60/400 Hz; 200 to 240 VAC at 50/60 Hz; Power consumption: 650 VA, 650 W max. (SDA 3010/4000A/5000A/6000A); 800 VA, 800 W max. (SDA 4020/6020/9000/11000/13000/18000)

Environmental Characteristics

Temperature

- **Operating:** 5 to 40 °C
- **Storage (non-operating):** -20 to +60 °C

Humidity

- **Operating:** Maximum relative humidity 80% for temperatures up to 31 °C decreasing linearly to 50% relative humidity at 40 °C
- **Storage (non-operating):** 5 to 95% RH (non-condensing) as tested per MIL-PRF-28800F

Altitude

- **Operating:** Up to 2,000 m
- **Storage (non-operating):** 12,192 m (40,000 ft)

Certifications

CE Compliant, UL and cUL Listed

CE Declaration of Conformity

The oscilloscope meets requirements of EMC Directive 89/336/EEC for Electromagnetic Compatibility and Low Voltage Directive 73/23/EEC for Product Safety.

EMC Directive:	EN 61326/A3:2003 EMC requirements for electrical equipment for measurement, control, and laboratory use.
Electromagnetic Emissions:	EN 55011/A2:2002, Class A Radiated and conducted emissions (Class A)*
	EN 61000-3-2/A2:2005 Harmonic Current Emissions (Class A)
	EN 61000-3-3/A2:2005 Voltage Fluctuations and Flickers (Pst = 1)

* To conform to Radiated Emissions standard, use properly shielded cables on all I/O terminals.

Warning

This is a Class A product. In a domestic environment this product may cause radio interference, in which case the user may be required to take appropriate measures.

Electromagnetic Immunity:	EN 61000-4-2/A2:2001* Electrostatic Discharge (4 kV contact, 8 kV air, 4 kV vertical/horizontal coupling planes)
	EN 61000-4-3/A1:2003* RF Radiated Electromagnetic Field (3 V/m, 80-1000 MHz)
	EN 61000-4-4:2004* Electrical Fast Transient/Burst (1 kV AC Mains, 0.5 kV I/O signal/control)
	EN 61000-4-5/A1:2001* Surges (1 kV AC Mains, 0.5 kV I/O signal/control)
	EN 61000-4-6/A1:2001* RF Conducted Electromagnetic Field (1 kV / 0.5 kV common mode / differential mode - AC Mains)
	EN 61000-4-11:2004** Mains Dips and Interruptions (1 cycle voltage dip, 100% short interruption)

* Meets Performance Criteria "B" limits during the disturbance; product undergoes a temporary degradation or loss of function of performance which is self recoverable.

** Meets Performance Criteria "C" limits during the disturbance; product undergoes a temporary degradation or loss of function of performance which requires operator intervention or system reset.

Low Voltage Directive:	EN 61010-1:2001
	Safety requirements for electrical equipment for measurement, control, and laboratory use.
	The oscilloscope has been qualified to the following EN 61010-1 limits: Installation Categories II (Mains Supply Connector) & I (Measuring Terminals) Pollution Degree 2 (Normally only dry non-conductive pollution occurs. Occasionally a temporary conductivity caused by condensation must be expected.) Protection Class I (Provided with terminal for protective ground)

UL and cUL Certifications:	UL Standard: UL 61010-1 2nd Edition
	Canadian Standard: CSA-C22.2 No. 61010-1-04

Physical Dimensions

- **Dimensions (HWD):** 264 mm x 397 mm x 491 mm; 10.4" x 15.6" x 19.3" (height excludes feet); SDA 18000/13000/11000/9000: 310 mm x 447 mm x 500 mm, 12.2" x 17.6" x 19.7
- **Weight:** 18 kg; 39 lbs. (SDA 6020: 23 kg; 49 lbs.)
- **Shipping Weight:** 24 kg (53 lbs.); SDA 6020, 4020: 29 kg (63 lbs.); SDA 18000/13000/11000/9000: 36 kg, (80 lbs.)

Warranty and Service

3-year warranty; calibration recommended annually.
Optional service programs include extended warranty, upgrades, and calibration services.

Optical Reference Receiver (with OE425/OE455)

4th order Bessel-Thompson filter response at the following data rates:

	SDA 3010/4000A/4020/5000A/6020/ 6000A/9000/11000/13000/18000
SONET/SDH	OC-1/STM0 (51.84 Mb/s) OC-3/STM1 (155.52 Mb/s) OC-12/STM4 (622.08 Mb/s) OC-48/STM16 (2488.3 Mb/s)
Fibrechannel	FC133 (132.7 Mb/s) FC266 (265.6 Mb/s) FC531 (531.2 Mb/s) FC1063 (1063.5 Mb/s) FC2125 (2127 Mb/s)
Gigabit Ethernet	1.25 Gb/s
InfiniBand	2.25 Gb/s (2127 Mb/s)
User-Defined	Any rate up to 3.5 Gb/s

Software Clock Recovery System

	SDA 4000A/SDA 5000A/SDA 6000A
Channel	Software-based clock recovery using golden PLL for data rates up to 3.5 Gb/s
PLL Bandwidth	Single pole w/ adjustable cut-off frequency
Jitter	Same as time base: 1ps rms

Communications Mask Testing

	SDA 3010/4000A/4020/5000A/6020/6000A/9000/11000/13000/18000
SONET/SDH	OC-1/STM0 OC-3/STM1 Optical OC-12/STM4 Optical OC-48/STM16 Optical
Ethernet IEEE Std 802.3 and ANSI X3.263-1995	1000Base-SX Short Wave Optical 1000Base-LX Long Wave Optical
Fibre Channel Electrical (ANSI X3.303-1997)	FC133E, FC266E, FC531E, FC1063E
IEEE 1394b (draft)	S400 Optical S400b T1 S400b T2 S800 Optical S800b T1 S800b T2 S1600 Optical S1600b T1 S1600b T2
Serial ATA	G1, G1 Rx, G1 Tx G2, G2 Rx, G2 Tx
DVI (rev.1.0)	Transmit normalized, Receiver Low/high
InfiniBand (draft)	2.5 Gb/s Optical 2.5 Gb/s Electrical

Jitter Testing

- **Data Measurements:** Period, Frequency, TIE
- **Clock Measurements:** Frequency, Period, Half Period, Cycle-Cycle, N-Cycle, Positive Cy-Cy Duty, Negative Cy-Cy Duty, Clock TIE
- **Analysis:** Tj, DCD, DDj, Pj, Rj, Dj, Rje, Dje
- **Delta Time Accuracy:** 1 ps rms at 20 GS/s

Bit Error Rate (optional)

	SDA 4000A/SDA 5000A/SDA 6000A
Data Rate	≤ 2.7 Gb/s
Maximum capture buffer size (bits)	20 GS/s at 2.5 Gb/s data rate (8 samples/bit)
Std. **VL (32M)** **XL (50M)** **XXL (100M)**	1M 3M 5M 10M
Minimum Detectable BER (single acquisition)	1×10^{-7}

SAFETY

Safety Requirements

This section contains information and warnings that must be observed to keep the instrument operating in a correct and safe condition. You are required to follow generally accepted safety procedures in addition to the safety precautions specified in this section.

Safety Symbols

Where the following symbols appear on the instrument's front or rear panels, or in this manual, they alert you to important safety considerations.



This symbol is used where caution is required. Refer to the accompanying information or documents in order to protect against personal injury or damage to the instrument.



This symbol warns of a potential risk of shock hazard.



This symbol is used to denote the measurement ground connection.



This symbol is used to denote a safety ground connection.



This symbol is used to denote a grounded frame or chassis terminal.



This symbol shows that the switch is a Standby (power) switch. When it is pressed, the scope's state toggles between operating and Standby mode. This switch is not a disconnect device. The instrument can only be placed in a complete Power Off state by unplugging the power cord from the AC supply.



This symbol is used to denote Alternating Current.

CAUTION The CAUTION sign indicates a potential hazard. It calls attention to a procedure, practice or condition which, if not followed, could possibly cause damage to equipment. If a CAUTION is indicated, do not proceed until its conditions are fully understood and met.

WARNING The WARNING sign indicates a potential hazard. It calls attention to a procedure, practice or condition which, if not followed, could possibly cause bodily injury or death. If a WARNING is indicated, do not proceed until its conditions are fully understood and met.

CAT I Installation (Overvoltage) Category rating per EN 61010-1 safety standard and is applicable for the oscilloscope front panel measuring terminals. CAT I rated terminals must only be connected to source circuits in which measures are taken to limit transient voltages to an appropriately low level.

Operating Environment

The instrument is intended for indoor use and should be operated in a clean, dry environment. Before using this product, ensure that its operating environment will be maintained within these parameters:
Temperature: 5 to 40 °C
Humidity: Maximum relative humidity 80% for temperatures up to 31 °C decreasing linearly to 50% relative humidity at 40 °C.
Altitude: Up to 2,000 m

Note: Direct sunlight, radiators, and other heat sources should be taken into account when assessing the ambient temperature.



WARNING

The scope must not be operated in explosive, dusty, or wet atmospheres.



CAUTION

Protect the scope's display touch-screen from excessive impacts with foreign objects.



CAUTION

Do not exceed the maximum specified front panel terminal (CH1, CH1, CH2, CH3, CH4, AUX IN) voltage levels. Refer to Specifications for more details.

Installation (Overvoltage) Category II refers to local distribution level, which is applicable to equipment connected to the mains supply (AC power source).
Installation (Overvoltage) Category I refers to signal level, which is applicable to equipment measuring terminals that are connected to source circuits in which measures are taken to limit transient voltages to an appropriately low level.
Pollution Degree 2 refers to an operating environment where normally only dry non-conductive pollution occurs. Occasionally a temporary conductivity caused by condensation must be expected.
Protection Class 1 refers to a grounded equipment, in which protection against electric shock is achieved by Basic Insulation and by means of a connection to the protective ground conductor in the building wiring.

Note:

The design of the instrument has been verified to conform to
EN 61010-1 safety standard per the following limits:

Installation (Overvoltage) Categories II (Mains Supply Connector) & I (Measuring Terminals)

Pollution Degree 2

Protection Class I

Cooling

The instrument relies on forced air cooling with internal fans and ventilation openings. Care must be taken to avoid restricting the airflow around the apertures (fan holes) at the sides and rear of the scope. To ensure adequate ventilation it is required to leave a 10 cm (4 inch) minimum gap around the sides and rear of the instrument.
The instrument also has internal fan control circuitry that regulates the fan speed based on the ambient temperature. This is performed automatically after start-up with no manual intervention required.

CAUTION
Do not block the ventilation holes located on both sides and rear of the scope.

CAUTION
Do not allow any foreign matter to enter the scope through the ventilation holes, etc.

AC Power Source

100 to 120 VAC (±10%) AC at 50/60/400 Hz;
200 to 240 VAC (±10%) AC at 50/60 Hz; Automatic AC voltage selection; Installation Category: 300V CAT II
No manual voltage selection is required because the instrument automatically adapts to line voltage.

Note:
The instrument automatically adapts itself to the AC line input within the following ranges:

Voltage Range
90 to 132 VAC
180 to 264 VAC
Frequency Range
45 to 440 Hz
45 to 66 Hz

Power Consumption

SDA 6020, 4020, 9000, 11000, 13000, 18000: ≤ 800 watts (800 VA) depending on accessories installed (internal printer, probes, PC port plug-ins, etc.)
SDA 4000A, 5000A, 6000A: ≤ 650 watts (650 VA)
The power supply of the scope is protected against short circuit and overload by a 5x20 mm fuse (T10.0 A/250 V). See "Fuse Replacement" section for replacement instructions.

Power and Ground Connections

The instrument is provided with a grounded cord set containing a molded three-terminal polarized plug and a standard IEC320 (Type C13) connector for making line voltage and safety ground connection. The AC inlet ground terminal is connected directly to the frame of the instrument. For adequate protection against electrical shock hazard, the power cord plug must be inserted into a mating AC outlet containing a safety ground contact.



WARNING

Electrical Shock Hazard!

Any interruption of the protective conductor inside or outside of the scope, or disconnection of the safety ground terminal creates a hazardous situation.

Intentional interruption is prohibited.

In Standby mode the scope is still connected to the AC supply. The instrument can only be placed in a complete Power Off state by physically disconnecting the power cord from the AC supply.
The scope should be positioned to allow easy access to the socket-outlet. To disconnect the scope from the AC supply, unplug the instrument's power cord from the AC outlet after the scope is placed in Standby state.
See "Standby (Power) Switch and Scope Operational States" section for more information.



CAUTION

The outer shells of the front panel terminals (CH1, CH2, CH3, CH4, AUX IN, AUX OUT) are connected to the instrument's chassis and therefore to the safety ground.

Standby (Power) Switch and Scope Operational States

The front Standby (Power) switch controls the operational state of the scope. This toggle switch is activated by momentarily pressing and releasing it. The color of the LED below the switch indicates the status of the scope as follows:

- **On (LED Green)*** - scope is fully-powered and operational
- **Standby (LED Off)*** - scope is powered off (except for some "housekeeping" circuits)
- **Standby (LED Red)** - scope's computer subsystems (hard drive, etc.) are in Standby (reduced power) mode. All other scope subsystems are fully-powered.

* Factory Settings

The scope's factory settings result in two basic scope states: On (LED Green) or Standby (LED Off). In Standby (LED Off), the scope is powered off with the exception of some "housekeeping" circuitry (approximately 12 watts of dissipation). The scope can only be placed in a complete power off state by unplugging the instrument's power cord from the primary power source (AC outlet). It is recommended that the power cord be unplugged from the AC outlet when the scope is not used for extended periods of time.

The scope can be changed back to the original factory settings using the "Power Options Properties" menu in Windows at the following the path: Settings ➔ Power ➔ Options. It is important to note that the Windows Power Option named "Standby" only provides control of the scope's computer subsystems (CPU, hard drive, etc.) and does not affect the other subsystems within the scope. In general, these other subsystems remain fully-powered. For additional information on setting these Power Options, see the Windows Help menu or other related technical documentation. In terms of control buttons, this scope uses only a power button/switch. Therefore, sleep button references are not applicable.

The scope can always be put in the Standby state (LED Off) Power Off (except for some "housekeeping" circuits) by pressing and holding the Standby toggle switch for approximately 5 seconds.

Fuse Replacement

Set the scope's Standby (power) switch to Standby mode (LED off) and disconnect the power cord before inspecting or replacing the fuse. Open the black fuse holder (located at the rear of the instrument directly to the right of the AC inlet) using a small, flat-bladed screwdriver. Remove the old fuse, replace it with a new 5x20 mm "T" rated 10.0 A/250 V fuse, and reinstall the fuse holder.



WARNING

For continued fire protection at all line voltages, replace fuse with the specified type and rating only. Disconnect the power cord before replacing fuse.

Calibration

The recommended calibration interval is one year and should be performed by qualified personnel only.

Cleaning

Clean only the exterior of the instrument, using a damp, soft cloth. Do not use chemicals or abrasive elements. Under no circumstances allow moisture to penetrate the instrument. To avoid electrical shock, unplug the power cord from the AC outlet before cleaning.



WARNING
Electrical Shock Hazard! No operator serviceable parts inside. Do not remove covers.
Refer servicing to qualified personnel.

Abnormal Conditions

Operate the instrument only as intended by the manufacturer.
If you suspect the scope's protection has been impaired, disconnect the power cord and secure the instrument against any unintended operation.
The scope's protection is likely to be impaired if, for example, the instrument shows visible damage or has been subjected to severe transport stresses.
Proper use of the instrument depends on careful reading of all instructions and labels.



WARNING
Any use of the scope in a manner not specified by the manufacturer may impair the instrument's safety protection. The instrument and related accessories should not be directly connected to human subjects or used for patient monitoring.

BASIC CONTROLS

Alternate Access Methods

Note: The instrument often provides more than one way to access dialogs and menus.

Mouse and Keyboard Operation

Touch-screen operation is used for most procedures explained in our manuals. However, if you have a mouse connected to the instrument, you can also be used to click on objects. Also, if you have a keyboard connected, it can be used instead of the virtual keyboard provided in the instrument.
Use only a USB connection for a mouse (if used).

Toolbar Buttons

The procedures also focus on the use of the menu bar at the top of the screen to access dialogs and menus. However, on several dialogs common functions are accessible from a row of buttons that save you a step or two in accessing their dialogs. For example, at the bottom of the Channel Setup dialog, these buttons perform the following functions:



	Calls up the Measure menu. You can then select a parameter from this menu without leaving the Channel Setup dialog. The parameter automatically appears below the grid.
	Creates a zoom trace of the channel trace whose dialog is currently displayed.
	Calls up the Math menu. You can then select a math function from this menu without leaving the Channel Setup dialog. A math trace of the channel whose dialog is currently open is automatically displayed.
	Loads the channel trace into the next available memory location (M1 to M4).
	Automatically performs a vertical scaling that fits the waveform into the grid.
	Automatically moves the channel trace whose dialog is currently open onto the next grid. If you have only one grid displayed, a new grid will be created automatically, and the trace moved.
	Enables you to attach identifying labels to your waveforms. The labels are preserved when the waveform is saved as a LabNotebook entry and when saved to file.

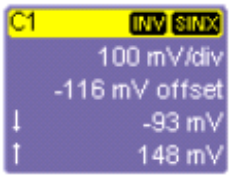
Another example includes the buttons appearing at the bottom of the Measure **Px** dialogs. Each button opens a menu from which to choose a math trace (F1 to Fx - The number of math traces available depends on the software options loaded on your scope, so please refer to your scope specifications) to display the functions named in the following buttons:



By using these buttons you can remain in the Measure dialog and set up other options.

Trace Descriptors

Vertical and horizontal trace descriptors (labels) are displayed below the grid. They provide a summary of your channel, timebase, and trigger settings. To make adjustments to these settings, touch the respective label to display the setup dialog for that function.



Channel trace labels show the vertical settings for the trace, as well as cursor information if cursors are in use. In the title bar of the label are also included indicators for (SinX)/X interpolation, waveform inversion (INV), deskew (DSQ), coupling (DC/GND), bandwidth limiting (BWL), and averaging (AVG). These indicators have a long and short form.

AVG	A
SINX	S
INV	I
DSQ	DQ
DC	D
Gnd	G
BwL	B

Besides channel traces, math and parameter measurement labels are also displayed. Labels are displayed only for traces that are turned on.



The title bar of the **TimeBase** label shows the trigger delay setting. Time per division and sampling information is given below the title bar.



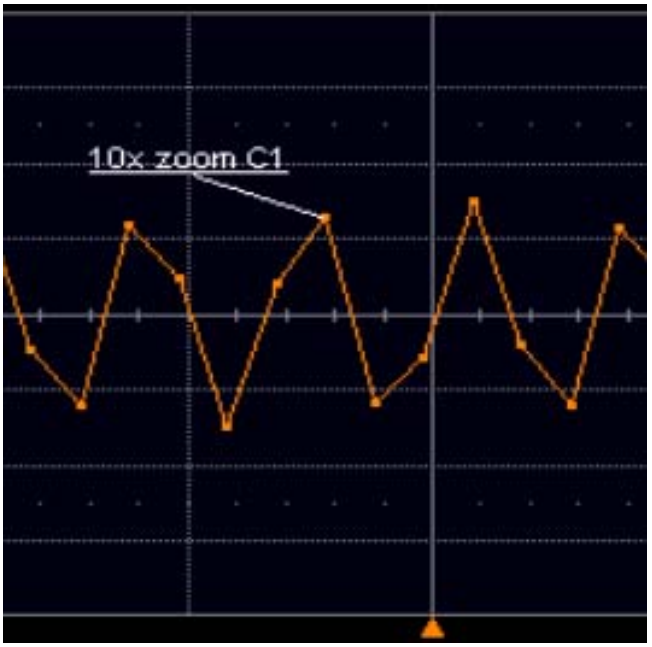
The title bar of the **Trigger** label shows the trigger mode: Auto, Normal, or Stopped. Below the title bar is given the coupling (DC), trigger type (Edge), source (C1), level (0 mV), and slope (Positive).



Shown (as follows) the TimeBase and Trigger labels are setup information for horizontal cursors, including the time between cursors and the frequency.

Trace Annotation

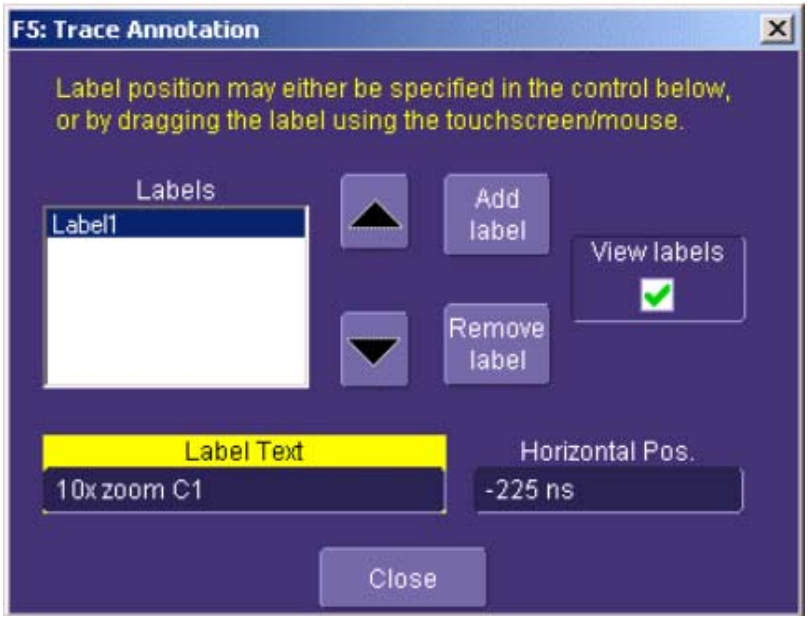
The instrument gives you the ability to add an identifying label, bearing your own text, to a waveform display:



For each waveform, you can create multiple labels and turn them all on or all off. Also, you can position them on the waveform by dragging or by specifying an exact horizontal position.

Annotating a Waveform

1. Touch the waveform you want to annotate, and then **Set label...** in the pop-up menu. A dialog box opens in which to create the label. If you are creating a label for the first time for this waveform, **Label1** is displayed with default text. If you are modifying an existing label, under **Labels** touch the label you want to change.



Please Note the Following:



- If the dialog for the trace you want to annotate is currently displayed, you can touch the label button at the bottom to display the Trace Annotation setup dialog.
- You may place a label anywhere you want on the waveform. Labels are numbered sequentially according to the order in which they are added, and not according to their placement on the waveform.

2. If you want to change the label's text, touch inside the **Label Text** field. A pop-up keyboard appears for you to enter your text. Touch **O.K.** on the keyboard when you are done. Your edited text will automatically appear in the label on the waveform.
3. To place the label precisely, touch inside the **Horizontal Pos.** field and enter a horizontal value, using the pop-up numeric keypad.
4. To add another label, touch the **Add label** button. To delete a label, select the label from the list, then touch the **Remove label** button.
5. To make the labels visible, touch the **View labels** checkbox.

Turning on a Channel Trace Label

Note: If you want to display each trace on its own grid automatically, enable Autogrid by touching **Display** in the menu bar, then **Autogrid** in the drop-down menu.

- On the front panel, press a channel select button, such as , to display the trace label for that input channel and turn on the channel.
- To turn on a math function trace, touch **Math** in the menu bar, then **Math Setup...** in the drop-down menu. Touch the **On** checkbox for the trace you want to activate.
- You can also quickly create traces (and turn on the trace label) for math functions and memory traces, without leaving the Vertical Adjust dialog, by touching the icons at the bottom of the Vertical Adjust dialog:



Whenever you turn on a channel, math, or memory trace via the menu bar, the dialog at the bottom of the screen automatically switches to the vertical setup or math setup dialog for that selection.



You can configure your traces from here, including math setups.
The channel number appears in the Vertical Adjust tab of the "Vertical Adjust" dialog, signifying that all controls and data entry fields are dedicated to the selected trace.



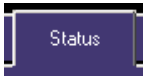
INSTALLATION

Software

Checking the Scope Status

To find out the scope's software and hardware configuration, including software version and installed options, proceed as follows:

- In the menu bar, touch 

.
- Touch the 

tab.

You can find information related to hard drive memory, etc. as follows:

1. Minimize the instrument application by touching 

, then selecting **Minimize** in the drop-down menu.
2. Touch the **Start** taskbar button and, per usual Windows® operation, open Windows Explorer.

Default Settings

DDA, SDA, and WaveRunner DSOs

On your front panel, the **DEFAULT SETUP** push button does not exist. For these instruments, therefore, to recall a default setup

1. Press the **SAVE/RECALL** push button to the left of the **DRIVE ANALYSIS** push button.

Note: You can also touch **File** in the menu bar, then **Recall Setup...** in the drop-down menu.

2. Touch the "Recall Setup" tab in the dialog.
3. Then touch the on-screen **Recall Default** button:



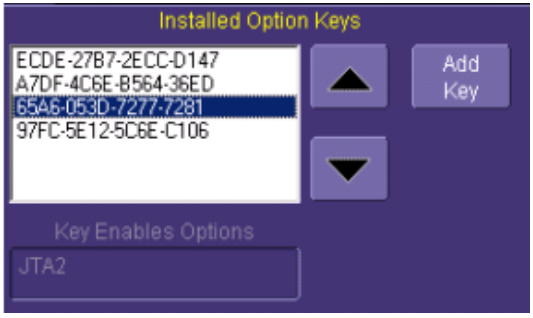
Adding a New Option

To add a software option you need a key code to enable the option. Call LeCroy Customer Support to place an order and receive the code.
To add the software option do the following:

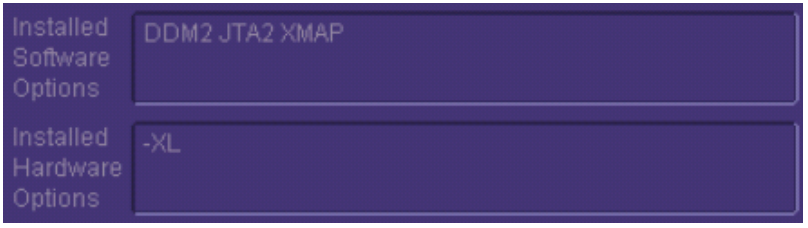
1. In the menu bar, touch 

.

2. In the dialog area, touch the Options tab.
3. Touch Add Key.
4. Use the pop-up keyboard to type the key code. Touch **O.K.** on the keyboard to enter the information.
5. The name of the feature you just installed is shown below the list of key codes. You can use the scroll buttons to see the name of the option installed with each key code listed:



The full array of installed software and hardware options is displayed on the left side of the dialog:



Restoring Software

Restarting the Application

Upon initial power-up, the scope will load the instrument application software automatically. If you exit the application and want to reload it, touch the shortcut icon on the desktop:



If you minimize the application, touch the appropriate task bar or desktop icon to maximize it:



Restarting the Operating System

To restart the Windows® operating system, reboot the scope by pressing and holding in the power switch for 10 seconds (until the system shuts down). Press the power button again to start the scope.

Removable Hard Drive

The removable hard drive option replaces the standard internal hard drive with a removable hard drive that is installed at the rear of the scope, in the slot normally occupied by the CD-ROM drive. The kit includes two hard drives, which can be used interchangeably. It also includes a USB CD-ROM for loading of new software.

Caution! The Removable Hard Drive Is Not Hot-swappable
To avoid damage to the drive or the oscilloscope, shut off power to the oscilloscope before you insert or remove the hard drive. Ensure that the protective cover is installed over the drive at all times.



Proper Orientation of Drive



Protective Cover

CONNECTING TO A SIGNAL

ProBus Interface

LeCroy's ProBus® probe system provides a complete measurement solution from probe tip to oscilloscope display. ProBus allows you to control transparent gain and offset directly from your front panel. It is particularly useful for voltage, differential, and current active probes. It uploads gain and offset correction factors from the ProBus EPROMs and automatically compensates to achieve fully calibrated measurements.
This intelligent interconnection between your instrument and a wide range of accessories offers important advantages over standard BNC and probe ring connections. ProBus ensures correct input coupling by auto-sensing the probe type, thereby eliminating the guesswork and errors that occur when attenuation or amplification factors are set manually.

ProLink Interface

LeCroy's ProLink Adapters (LPA) give you the ability to connect your signal in one of three ways:

- BMA connector
- SMA using the BMA-to-SMA adapter
- BNC using BMA-to-BNC adapter



(1) BMA-to-SMA Adapter; (2) BMA-to-BNC Adapter



(1) BMA Female Connector, (2) ProLink BMA-to-SMA Adapter Installed, (3) ProLink BMA-to-BNC Adapter Installed

Note: When connecting an active probe to the instrument, be sure to use a ProLink BMA-to-BNC adapter (item 3 in the figure). Do not plug the probe directly into the front panel connector (item 1) without an adapter.

Connecting the Adapters

The mating end of the ProLink adapter has four fastening clips, as shown here:



When installing an adapter on the instrument's connector panel, align the male 6-pin connector with the female connector and push the adapter straight in. There will be some resistance and you'll hear clicks as the four clips slide into place. Then tighten the captive screws.
When removing an adapter, loosen the two captive screws. Push down on the adapter to unseat the clips. This will require some force and will be initially noisy, but no damage will result to the connector, the floating female BMA connector, or the pins, which can be 15 degrees off axis when being mated or unmated.

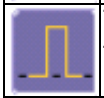
AP-1M Hi-Z Adapter



The AP-1M adapter provides a means to connect a high-impedance input to your instrument. In order to achieve high bandwidth with excellent signal integrity, these instruments have a ±4 V dynamic range and 50 Ω termination to ground. However, for applications that combine one or more high-speed signals with slower, higher-voltage signals, the AP-1M provides a 1 M input impedance path and a full-scale range of ±8 V. It is also suitable as an interface for current probes that require a 1 M input path.
In addition to acting as a ProLink-to-ProBus adapter, the AP-1M also enables a much larger offset voltage range (up to ±50 V).
The AP-1M is supplied with a PP005A passive probe.

Auxiliary Output Signals

In addition to a calibration signal, the following signals can be output through the AUX OUTPUT connector:

	Square Wave
	Trigger Out - can be used to trigger another scope
	DC level - a reference level
	Trigger Enabled - can be used as a gating function to trigger another instrument when the scope is ready
	Pass/Fail - allows you to set a pulse duration from 1 ms to 500 ms; generates a pulse when pass/fail testing is active and conditions are met.
	Aux Output Off - turns off the auxiliary output signal

Auxiliary Output Setup

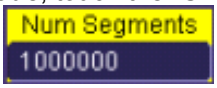
1. In the menu bar, touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
2. Touch the **Aux Output** tab.
3. If you simply want a 1 kHz, 1 V square wave, touch the button so labeled.
4. If you want a specialized output, touch one of the buttons under **Use Auxiliary Output For**.
5. Touch inside the **Amplitude** data entry field and enter a value, using the pop-up numeric keypad. If you want a TTL level signal, touch the **TTL Level** checkbox. The **Amplitude** field will accordingly become unavailable.
6. If you selected Square Wave, touch inside the **Frequency** data entry field and enter a value, using the pop-up keypad. You can set a value from 5.0 Hz to 5 MHz.
7. If you selected Pass/Fail, touch inside the **Pulse Duration** field and enter a value from 1 ms to 500 ms, using the pop-up numeric keypad.

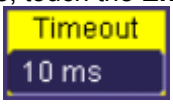
SAMPLING MODES

Sampling Modes

Depending on your timebase, you can choose either Single-shot (Real Time) , Sequence , or RIS  mode sampling.

Sampling Mode Selection

1. In the menu bar, touch **Timebase**, then **Horizontal Setup...** in the drop-down menu.
2. In the "Horizontal" dialog, touch a **Sample Mode** button.
3. If you chose **Sequence Mode**, touch the "Smart Memory" tab, then touch inside the **Num Segments** data entry field 

and enter a value using the pop-up numeric keypad.
4. If you want to use a timeout condition for Sequence mode, touch the **Enable Timeout** checkbox; then touch inside the **Timeout** data entry field 

and enter a value using the pop-up numeric keypad.

Single-shot Sampling Mode

Basic Capture Technique

A single-shot acquisition is a series of digitized voltage values sampled on the input signal at a uniform rate. It is also a series of measured data values associated with a single trigger event. The acquisition is typically stopped a defined number of samples after this event occurs: a number determined by the selected trigger delay and measured by the timebase. The waveform's horizontal position (and waveform display in general) is determined using the trigger event as the definition of time zero.

You can choose either a pre- or post-trigger delay. Pre-trigger delay is the time from the left-hand edge of the display grid forward to the trigger event, while post-trigger delay is the time back to the event. You can sample the waveform in a range starting well before the trigger event up to the moment the event occurs. This is 100% pre-trigger, and it allows you to see the waveform leading up to the point at which the trigger condition was met and the trigger occurred. (The instrument offers up to the maximum record length of points of pre-trigger information.) Post-trigger delay, on the other hand, allows you to sample the waveform starting at the equivalent of 10,000 divisions after the event occurred.

Because each instrument input channel has a dedicated ADC (Analog-to-Digital Converter), the voltage on each is sampled and measured at the same instant. This allows very reliable time measurements between the channels.
On fast timebase settings, the maximum single-shot sampling rate is used. But for slower timebases, the sampling rate is decreased and the number of data samples maintained.
The relationship between sample rate, memory, and time can be simply defined as:

$$Capture\ Interval = \frac{1}{Sample\ Rate} \times Memory$$

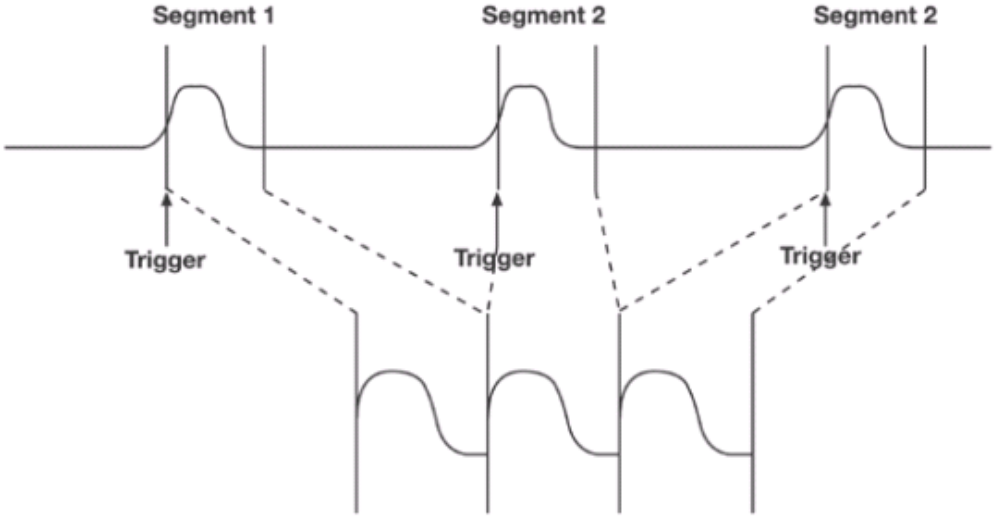
and

$$\frac{Capture\ Interval}{10} = Time\ Per\ Division$$

Sequence Sampling Mode Working With Segments

In sequence mode, the complete waveform consists of a number of fixed-size segments acquired in single-shot mode (see the instrument specifications for the limits). Select the number of segments to be captured, then select each segment individually and use it for processing with math and measure tools.
Sequence mode offers a number of unique capabilities. With it, you can limit dead time between trigger events for consecutive segments. The instrument can capture in fine detail complicated sequences of events over large time intervals, while ignoring the uninteresting periods between the events. You can also make time measurements between events on selected segments using the full precision of the acquisition timebase.
Each individual segment can be zoomed or used as input to math functions.
The instrument uses the sequence timebase setting to determine the capture duration of each segment: 10 x time/div. Along with this setting, the scope uses the desired number of segments, maximum segment length, and total available memory to determine the actual number of samples or segments, and time or points. However, the display of the complete waveform with all its segments may not entirely fill the screen.

You can also use Sequence mode in remote operation to take full advantage of the instrument's high data-transfer capability.



How the instrument captures segments

Sequence Mode Setup

Note: Once a single acquisition has started, you can interrupt the acquisition at any time by pressing the Single front panel button a second time. In this case, the segments already acquired will be retained in memory.

1. In the menu bar, touch **Timebase**, then touch **Horizontal Setup...** in the drop-down menu.
2. Touch the **Smart Memory** tab, then touch the **Sequence** mode button .
3. Under **Sequence Options**, touch inside the **Num Segments** data entry field and enter the number of segments you want to display, using the pop-up keypad.
4. Touch inside the **Timeout** data entry field and enter a timeout value.

Note: The timeout period accounts for instances when a **Num Segments** miscount occurs for some reason and the scope waits indefinitely for an unforthcoming segment. During that time, no scope functions are accessible. By means of a timeout value, however, the acquisition will be completed, the waveform displayed, and control of the scope returned to the user after the timeout has elapsed.

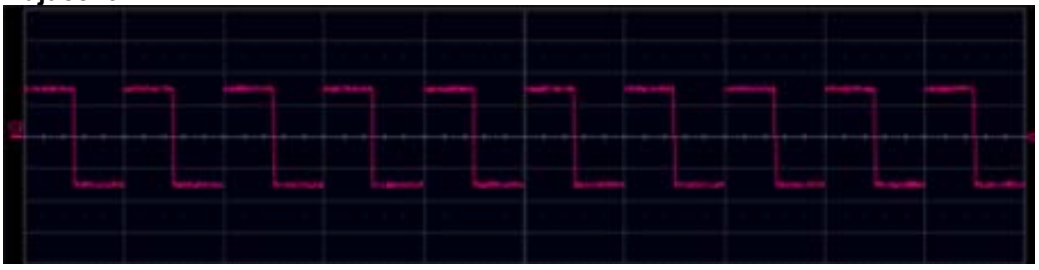
5. Touch the **Enable Timeout** checkbox.
6. In the menu bar, touch **Display**, then **Display Setup...** in the drop-down menu.

7. At the far right of the "Display" dialog, touch inside the **Display mode** field, and make a selection from the pop-up menu.
8. Touch inside the **Num seg displayed** field and enter a value from 1 to 80, using the pop-up numeric keypad.
9. Touch inside the **Starting at** field and enter a starting segment number, using the pop-up numeric keypad.

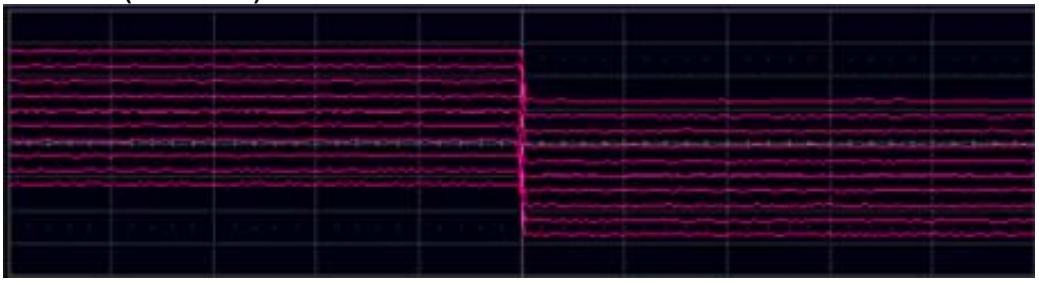
Sequence Display Modes

The instrument gives you a choice of five ways to display your segments:

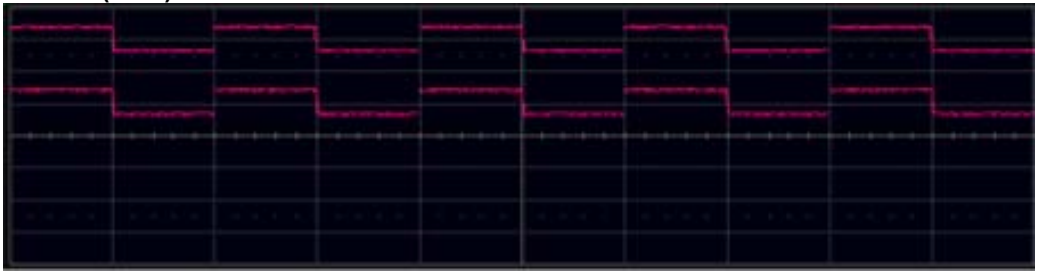
Adjacent



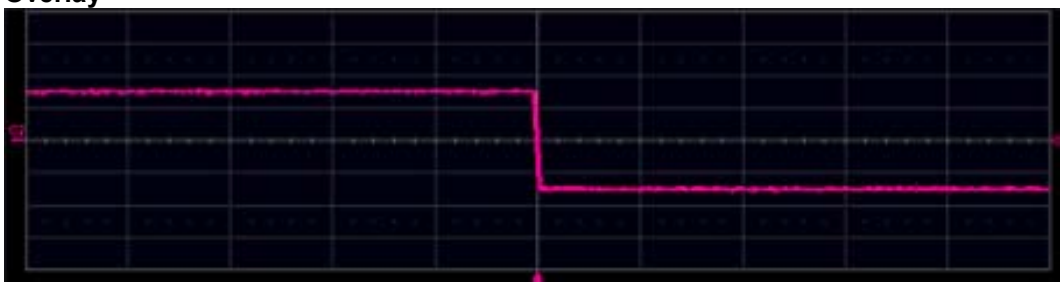
Waterfall (cascaded)



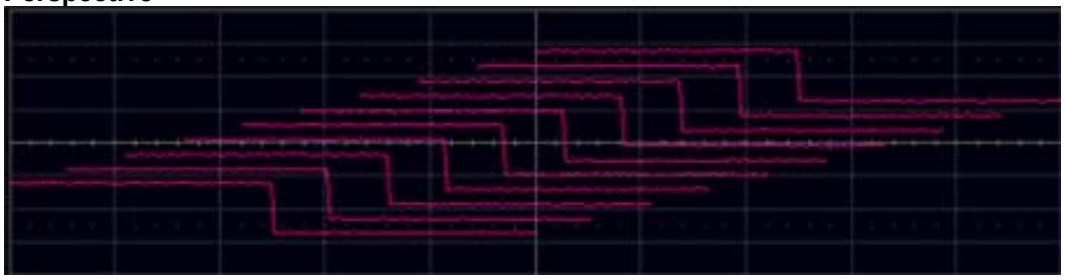
Mosaic (tiled)



Overlay



Perspective



The number of segments you choose to display (80 maximum) can be less than the total number of segments in the waveform. For example, in the pop-up images above, the number of display segments is 10, but the total number of segments entered in the timebase dialog's **Num Segments** field is 100.

Displaying Individual Segments

1. Touch **Math** in the menu bar, then **Math Setup...** in the drop-down menu.
2. Touch a function tab (**F1** to **Fx** The number of math traces available depends on the software options loaded on your scope. See specifications.).
3. Touch inside the **Operator1** field and select **Segment**  from the pop-up menu.
4. In the right-hand dialog, touch the **Select** tab.
5. Touch inside the **Select** data entry field and use the pop-up numeric keypad to select the segment you want to display.

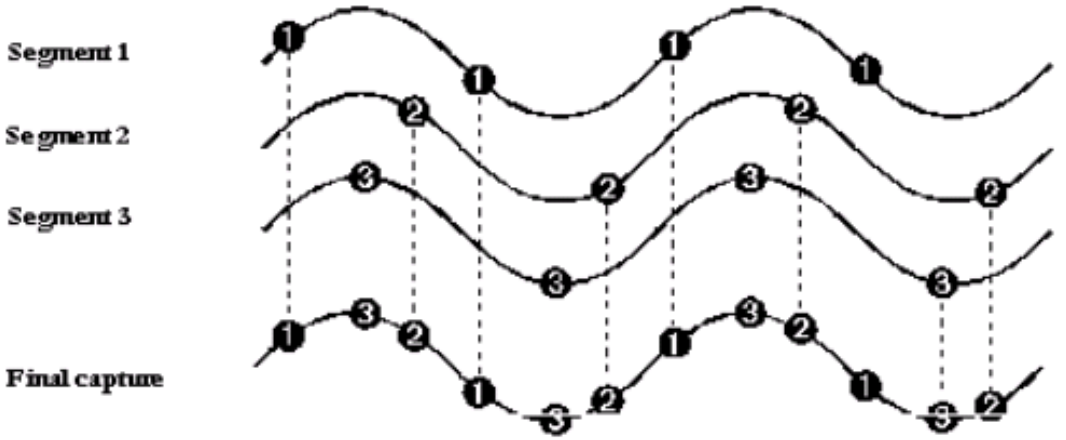
Note: In Persistence mode, the segments are automatically overlaid one on top of the other in the display. In non-Persistence mode, they appear separately on the grid.

Viewing Time Stamps

1. In the menu bar, touch **Timebase**, then touch **Acquisition Status...** in the drop-down menu.
2. Touch the **Time** tab.
3. Touch one of the channel buttons under **Select Waveform**.
4. Touch inside the **Select Segment** field and enter a segment number, using the pop-up keypad.

RIS SAMPLING Mode - For Higher Sample Rates

RIS (Random Interleaved Sampling) is an acquisition technique that allows effective sampling rates higher than the maximum single-shot sampling rate. It is used on repetitive waveforms with a stable trigger. The maximum effective sampling rate of 50 GS/s can be achieved with RIS by making 100 single-shot acquisitions at 500 MS/s. The bins thus acquired are positioned approximately 20 ps apart. The process of acquiring these bins and satisfying the time constraint is a random one. The relative time between ADC sampling instants and the event trigger provides the necessary variation, measured by the timebase to 5 ps resolution.
The instrument requires multiple triggers to complete an acquisition. The number depends on the sample rate: the higher the sample rate, the more triggers are required. It then interleaves these segments (see figure) to provide a waveform covering a time interval that is a multiple of the maximum single-shot sampling rate. However, the real-time interval over which the instrument collects the waveform data is much longer, and depends on the trigger rate and the amount of interleaving required. The oscilloscope is capable of acquiring approximately 40,000 RIS segments per second.



Note: RIS mode is not available when the scope is operating in Fixed Sample Rate mode. When the SDA 18/13/11/9000 is operating in 18/13/11/9 GHz mode, RIS mode sampling is not available.

Roll Mode

Roll mode applies only to WavePro 7000 and WaveRunner 6000 series scopes. It is invoked automatically when the time per division is 500 ms/div or greater. However, you can cancel Roll Mode and return to Real Time mode at any time.
Roll mode displays, in real time, incoming points in single-shot acquisitions that have a sufficiently low data rate. The oscilloscope rolls the incoming data continuously across the screen until a trigger event is detected and the acquisition is complete. The parameters or math functions connected to each channel are updated every time the roll mode buffer is updated, as if new data is available. This resets statistics on every step of Roll mode that is valid because of new data.

Note: If the processing time is greater than the acquire time, the data in memory gets overwritten. In this case, the instrument issues the warning: **Channel data is not continuous in ROLL mode!!!** and rolling will start over again.

VERTICAL SETTINGS AND CHANNEL CONTROLS

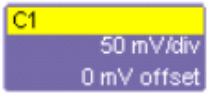
Adjusting Sensitivity and Position

Adjusting Sensitivity

1. Press the appropriate channel push button, for example 

to turn on channel 1. Or touch **Vertical** in the menu bar, then **Channel 1** in the drop-down menu.
2. Touch inside the **Trace On** checkbox 

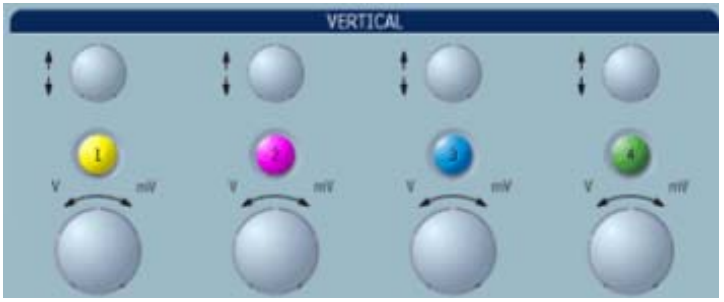
to display the trace.
3. Turn the volts per division knob 

for the selected channel. Or you can touch inside the **Volts/Div** field and type in a value using the pop-up keypad, or use the up/down arrows.
4. The voltage that you set is displayed in the trace descriptor label 

and in the **Volts/Div** field.

Adjusting the Waveform's Position

Turn the vertical offset adjust knob directly above the channel button whose waveform you want to move vertically.



Or, you can touch inside the **Offset** field and type in a value on the pop-up keypad. To set the vertical offset to zero, touch the **Zero Offset** button directly below the **Offset** field.

Coupling

The choices of coupling are as follows:

- DC 50 (all instruments)
- GROUND (all instruments)
- DC 1 M (WavePro & WaveRunner instruments)
- AC 1 M (WavePro & WaveRunner instruments)

Overload Protection

The maximum input voltage is 4 V peak. Whenever the voltage exceeds this limit, the coupling mode automatically switches from DC 50 Ω to GROUND. You will then have to manually reset the coupling to DC 50 Ω, as described next.

Coupling Setup

1. In the menu bar, touch the **Vertical** button, then **Channel X Setup...** in the drop-down menu.
2. Touch inside the **Coupling** field and select a coupling mode from the pop-up menu.

Probe Attenuation

Setting Probe Attenuation

LeCroy's ProBus system automatically senses probes and sets their attenuation for you. If you want to set the attenuation manually,

1. In the menu bar, touch **Vertical**, then select a channel from the drop-down menu.
2. Touch inside the **Probe Atten.** data entry field 

. Touch a divide-by menu selection or touch **Var** (variable). If you choose **Var**, type in a value using the pop-up numeric keypad.

Bandwidth Limit

Reducing the bandwidth also reduces the signal and system noise, and prevents high frequency aliasing.

To set bandwidth limiting

1. In the menu bar, touch **Vertical**, then select a channel from the drop-down menu.
2. Touch inside the **Bandwidth** field and select a bandwidth limit value from the pop-up menu. The options are
   - Full (all X-Stream scopes)
   - 4 GHz (WaveMaster 8600A/8500A, DDA-5005A, SDA)
   - 3 GHz (WaveMaster 8600A/8500A/4000A/4020, DDA-5005A, SDA)
   - 1 GHz (WaveMaster DSOs, DDA-5005A, SDA)
   - 200 MHz (all X-Stream scopes)
   - 20 MHz (all X-Stream scopes)

Linear and (SinX)/X Interpolation

Linear interpolation, which inserts a straight line between sample points, is best used to reconstruct straight-edged signals such as square waves. (Sinx)/x interpolation, on the other hand, is suitable for reconstructing curved or irregular wave shapes, especially when the sample rate is 3 to 5 times the system bandwidth.

Interpolation Setup

1. Touch the button for the channel you want to set up, 

for example.
2. In the dialog area, touch inside the **Interpolation** data entry field under **Pre-Processing**. "Pre-Processing" means before Math processing.
3. Touch inside the **Interpolation** data entry field. A pop-up menu appears offering **Linear** or **Sinx/x** interpolation.
4. Touch the button for the type of interpolation you want.

Inverting Waveforms

Touch the **Invert** checkbox to invert the waveform for the selected channel.

QuickZoom

QuickZoom automatically displays a zoom of the channel or trace on a new grid.

Turning On a Zoom

Touch the **Zoom** button 

in the channel dialog.

Finding Scale

You can access the **Find Scale** button from the channel setup dialog. This feature automatically calculates peak-to-peak voltage, and chooses an appropriate Volts/Div scale to fully display the waveform.

Using Find Scale

1. Touch the trace label for the waveform you desire.
2. Touch the **Find Scale** icon.

Variable Gain

Variable Gain lets you change the granularity with which the gain is incremented. For example, when **Variable Gain** is disabled, the gain will increase or decrease in preset increments of 10 or 100 mV each time you touch the **Up/Down** buttons.
However, when **Variable Gain** is enabled, you can increase or decrease the gain in increments as small as 1 mV, depending on the scale of the waveform.

Enabling Variable Gain

1. Touch the descriptor label for the waveform whose gain you want to vary.
2. Touch the **Variable Gain** check box.

Channel Deskew

Unlike the Deskew math function, channel Deskew does no resampling, but instead adjusts the horizontal offset by the amount that you enter. The valid range is dependent on the current timebase ± 9 divisions.

Channel Deskew Setup

1. In the menu bar, touch **Vertical**; from the drop-down menu, select a channel to set up.
2. Touch inside the **Deskew** data entry field and enter a value using the pop-up numeric keypad.

Group Delay Compensation

Group delay is defined as the rate of change of the total phase shift with respect to angular frequency through a device or transmission medium. An unavoidable consequence of correcting for group delay is that preshoot and overshoot will be added to the signal.

The SDA 18000/13000/11000/9000 provides a control in the channel "Vertical Adjust" dialog that allows you to choose between two group delay compensations.



Pulse Response - minimizes preshoot.
This selection is preferred in physics applications: no "prediction" of the future event.

Eye Diagram - less overshoot.
This selection improves the accuracy of serial eye diagrams.

Flatness – like eye diagram.
This selection provides a flat frequency response.

TIMEBASE AND ACQUISITION SYSTEM

Timebase Setup and Control

Set up the timebase by using the front panel **Horizontal** controls, just as for analog scopes.
For additional timebase setups

1. Touch **Timebase** in the menu bar, then **Horizontal Setup...** in the drop-down menu. The "Horizontal" dialog appears.
2. Touch inside the **Time/Division** data entry field and enter a value using the pop-up numeric keypad, or use the up/down arrows to adjust the value.
3. Touch inside the **Delay** data entry field and type in a value, using the pop-up keypad. Touch the **Set To Zero** button to set the delay to zero.
4. Touch the **SMART Memory** button or tab and adjust the memory as needed.

Autosetup

When channels are turned on, Autosetup operates only on those turned-on channels. If no channels are turned on, all channels are affected. When more than one channel is turned on, the first channel in numerical order with a signal applied to it is automatically set up for edge triggering.

You can perform an autosetup of all these functions together by simply pressing 

on the front panel, or by touching **Autosetup** 

in the Vertical, Timebase, or Trigger drop-down menu.

SDA DBI Controls

The SDA 18000/13000/11000/9000 oscilloscopes achieve higher bandwidth and sample rate using an innovative LeCroy technology called Digital Bandwidth Interleaving (DBI). This technology allows resources to be borrowed from unused channels to multiply not only sample rate but also bandwidth.

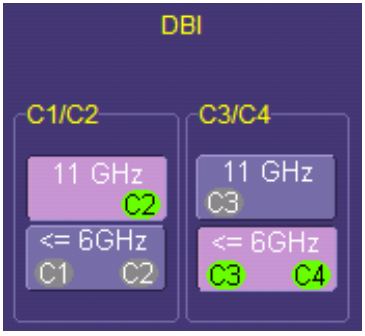
SDA18000 DBI Controls



When 18 GHz is selected, the active channels are either C1 or C4 or both, providing 60 GS/s sampling on the 18 GHz channel, and 20 GS/s sampling on the 6 GHz channel.
When 11 GHz is selected, the active channels are C1, C3 and C4. The 11 GHz channel provides 40 GS/s sampling, while the 6 GHz channels provide 20 GS/s sampling.
When 6 GHz is selected, all 4 channels are active, each providing 20 GS/s.

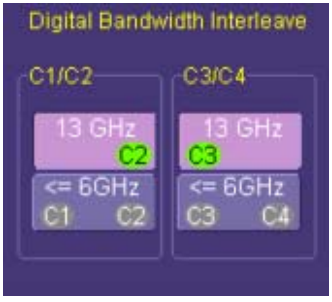
	18 GHz mode		11 GHz mode		6 GHz mode
Bandwidth	18 GHz	6 GHz	11 GHz	6 GHz	6 GHz
Number of Channels	1	1	1	2	4
Sample Rate	60 GS/s	20 GS/s	40 GS/s	20 GS/s	20 GS/s

SDA 13000/11000/9000 DBI Controls



When 13/11/9 GHz is selected, the active channels are either C2 or C3 or both, providing 40 GS/s sampling on each channel. For 6 GHz bandwidth, all four channels can be used, providing 20 GS/s sampling on each channel. Channels can be set to allow 13/11/9 GHz and 6 GHz bandwidths at the same time, as shown at left.

Bandwidth	13/11/9 GHz	6 GHz
Number of Channels	2	4
Sample Rate	40 GS/s	20 GS/s



Smart Memory

Note: When the SDA 18000/13000/11000/9000 is in DBI mode, only Fixed Sample Rate sampling is available.

LeCroy's SMART Memory feature ensures the highest time resolution for the time window displayed, without aliasing. SMART Memory provides these advantages:

- Acquisition memory is automatically allocated as needed.
- Memory size optimization: Set Maximum Memory optimizes memory to obtain highest sampling rate, reducing the risk of aliasing. You can set a maximum memory up to 48 Mpts.
- Fixed Sample Rate allows setting of a specific sample rate, with the scope calculating the amount of memory needed for a timebase setting.
- The entire acquisition is displayed on the screen.
- High-speed compaction shows all significant features of your waveform.
- You can set a maximum memory up to 48 Mpts.

SMART Memory Setup

1. Touch **Timebase** in the menu bar, then **SMART Mem Setup...** in the drop-down menu.
2. Touch the **SMART Memory** tab.
3. Under **Timebase Mode**, touch the **Set Maximum Memory** or **Fixed Sample Rate** button. Information about your choice appears below the buttons. The calculated memory length and time per sample point appear below the scroll buttons.
4. Touch inside the **Time/Division** data entry field and set a time per division.

Note: If you are currently acquiring waveforms, you will notice a change in sampling rate as you select different modes.

5. If you selected **Sequence** mode, touch inside the **Num Segments** data entry field and enter a value using the pop-up numeric keypad. If you want to use a timeout period, touch the **Enable Timeout** checkbox; then touch inside the **Timeout** data entry field and enter a value.

External Timebase vs. External Clock

An external timebase reference is used to synchronize the scope's internal timebase to an external frequency source. This allows multiple instruments to lock their timebases to a common source. The external timebase reference frequencies are model dependent. For example, the WaveMaster, SDA, and DDA series instruments use 100 MHz ±0.001%, and the voltage level required is 650 $mV_{pk\text{-}pk}$ or 0.0 dBm ±30%. On the other hand, the WavePro 7000A series uses an external timebase reference of 10 MHz ±0.01% with an amplitude of 650 $mV_{pk\text{-}pk}$ or 0.0 dBm ±30%. The external timebase reference is applied to the rear panel BNC connector.

Note: An external timebase reference cannot be applied to the SDA 18000/13000/11000/9000 when operating in DBI mode.

An external sampling clock, applied via the auxiliary input, replaces the scope's internal timebase as the sampling clock. This means that the external sampling clock controls when the scope's digitizers sample the input waveforms. The frequency range of the external sampling clock is model dependent: WaveMaster, SDA, DDA, and WavePro 7000A series scopes require a signal frequency between 30 MHz and 2 GHz. However, the WaveRunner 6000A series scopes accept an external sampling clock between DC and 1 GHz. In all cases, the external sampling clock levels are TTL, ECL, or 0 volt crossing 650 $mV_{pk\text{-}pk}$ or 0.0 dBm ±30%. Since the external sampling clock uses the auxiliary input, an external trigger cannot be used when the external sampling clock is in use.

TRIGGERING

TRIGGER SETUP CONSIDERATIONS

Trigger Modes

Auto mode causes the scope to sweep even without a trigger. An internal timer triggers the sweep so that the display remains, even when the signal does not cause a trigger.

In **Normal** mode, the scope sweeps only if the input signal reaches the set trigger point. Otherwise it continues to display the last acquired waveform.
In **Single** mode, only one sweep occurs each time you press the button.
Stop mode inhibits all sweeps until you select one of the other three modes.

Trigger Types

The triggers available to you are defined as follows:



A simple trigger, Edge trigger is activated by basic waveform features or conditions such as positive or negative slope, and holdoff.



One of LeCroy's SMART Triggers®, Width trigger allows you to define a positive- or negative-going pulse width bounded by a voltage level, above or below which a trigger will occur. Or you can specify a pulse width and voltage range, within or outside of which a trigger will occur.



Another of the SMART Triggers, Glitch trigger is a simpler form of Width trigger. Use Glitch trigger when you want to define a fixed pulse-width time or time range only. Glitch trigger makes no provision for voltage levels or ranges.



While Glitch trigger performs over the width of a pulse, Interval trigger performs over the width of an interval the signal duration (the period) separating two consecutive edges of the same polarity: positive to positive or negative to negative. Use interval trigger to capture intervals that fall short of, or exceed, a given time limit. In addition, you can define a width range to capture any interval that is itself inside or outside the specified range an Exclusion trigger by interval.



The Qualify trigger is an edge-qualified SMART Trigger that allows you to use one signal's positive or negative transition to qualify a second signal, which is the trigger source. For Qualify trigger, you specify the time or number of events after the transition when you want the trigger to occur.



The State trigger is a level-qualified SMART Trigger which requires that the qualifying signal remain above or below a specified voltage level for a trigger to occur. For Sate trigger, you specify the time or number of events after the signal has gone above or below the voltage level when you want the trigger to occur.



Used primarily in single-shot applications, and usually with a pre-trigger delay, Dropout trigger can detect lost signals. The trigger is generated at the end of the timeout period following the last trigger source transition. You can select a timeout period from 2 ns to 20 s.



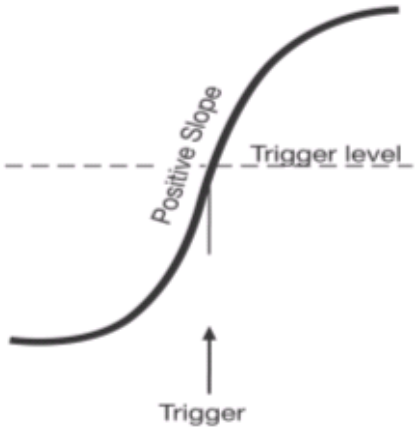
Logic trigger enables triggering on a logical combination (pattern) of five inputs: CH1, CH2, CH3, CH4, EXT. You have a choice of four Boolean operators (AND, NAND, OR, NOR), and you can stipulate the high or low voltage logic level for each input independently.

Determining Trigger Level, Slope, Source, and Coupling

Level defines the source voltage at which the trigger circuit will generate an event: a change in the input signal that satisfies the trigger conditions. The selected trigger level is associated with the chosen trigger source.
Trigger level is specified in volts and normally remains unchanged when you change the vertical gain or offset. The amplitude and range of the trigger level are limited as follows:

- ±4.1 screen divisions with a channel as the trigger source
- ±400 mV with EXT as the trigger source
- ±4 V with EXT/10 as the trigger source
- ±40 mV with EXT*10 as the trigger source
- None with LINE as the trigger source (zero crossing is used).

Coupling refers to the type of signal coupling at the input of the trigger circuit. Because of the instrument's very high bandwidth, there is only one choice of trigger coupling: DC 50 Ω. However, as a visual check of where ground is, you may switch the channel to ground coupling at any time while testing.
With DC coupling, all of the signal's frequency components are coupled to the trigger circuit for high-frequency bursts.
Slope determines the direction of the trigger voltage transition used for generating a particular trigger event. You can choose a positive or negative slope. Like coupling, the selected slope is associated with the chosen trigger source.



Edge trigger works on the selected edge at the chosen level. The slope (positive or negative) is specified in the Trigger label permanently displayed below-right of the grid.

Trigger Source

The **Trigger On** source may be one of the following:

- The acquisition channel signal (CH 1, CH 2, CH 3 or CH 4) conditioned for the overall voltage gain, coupling, and bandwidth.
- The line voltage that powers the oscilloscope (LINE). This can be used to provide a stable display of signals synchronous with the power line. Coupling and level are not relevant for this selection.
- The signal applied to the EXT BNC connector (EXT). This can be used to trigger the oscilloscope within a range of ±400 mV on EXT, ±4 V with EXT/10 as the trigger source, or ±40 mV with EXT*10 as the trigger source.
- A logic pattern.

Level

Level defines the source voltage at which the trigger circuit will generate an event (a change in the input signal that satisfies the trigger conditions). The selected trigger level is associated with the chosen trigger source. Note that the trigger level is specified in volts and normally remains unchanged when the vertical gain or offset is modified.
The Amplitude and Range of the trigger level are limited as follows:

- ±4.1 screen divisions with a channel as the trigger source
- ±400 mV with EXT as the trigger source
- ±4 V with EXT/10 as the trigger source
- ±40 mV with EXT*10 as the trigger source
- none with LINE as the trigger source (zero crossing is used)

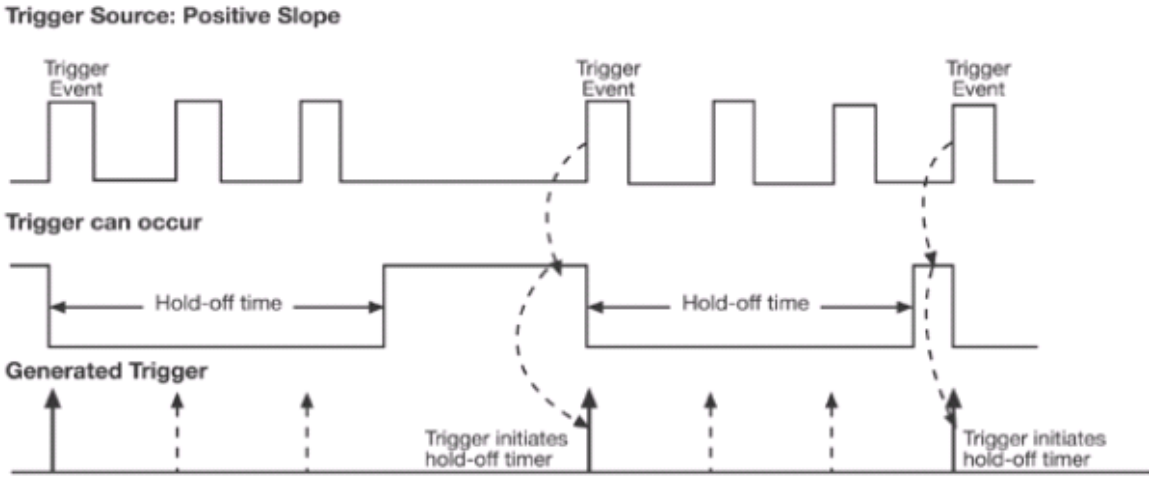
Note: Once specified, Trigger Level and Coupling are the only parameters that pass unchanged from trigger mode to trigger mode for each trigger source.

Holdoff by Time or Events

Holdoff is an additional condition of Edge trigger. It can be expressed either as a period of time or an event count. Holdoff disables the trigger circuit for a given period of time or number of events after the last trigger occurred. Events are the number of occasions on which the trigger condition is met. The trigger will again occur when the holdoff has elapsed and the trigger's other conditions are met.
Use holdoff to obtain a stable trigger for repetitive, composite waveforms. For example, if the number or duration of sub-signals is known you can disable them by choosing an appropriate holdoff value. Qualified triggers operate using conditions similar to holdoff.

Hold Off by Time

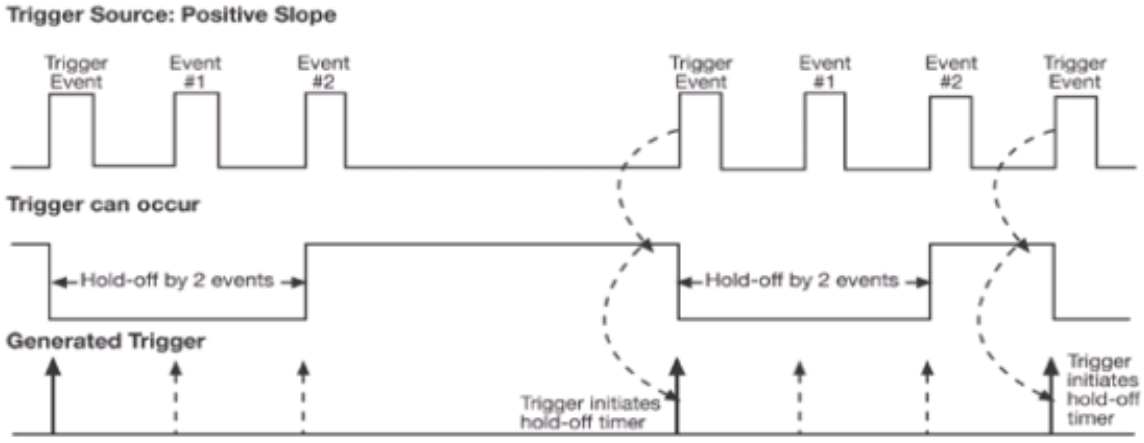
Sometimes you can achieve a stable display of complex, repetitive waveforms by placing a condition on the time between each successive trigger event. This time would otherwise be limited only by the input signal, the coupling, and the instrument's bandwidth. Select a positive or negative slope, and a minimum time between triggers. The trigger is generated when the condition is met after the selected holdoff time, counted from the last trigger. Any time between 2 ns and 20 s can be selected. The delay is initialized and started on each trigger.



Edge Trigger with Holdoff by Time. The bold edges on the trigger source indicate that a positive slope has been selected. The broken upward-pointing arrows indicate potential triggers, which would occur if other conditions are met. The bold arrows indicate where the triggers actually occur when the holdoff time has been exceeded.

Hold Off by Events

Select a positive or negative slope and a number of events. An event is the number of times the trigger condition is met after the last trigger. A trigger is generated when the condition is met after this number, counted from the last trigger. The count is restarted on each trigger. For example, if the event number is two, the trigger will occur on the third event. From one to 1,000,000,000 events can be selected.



Edge Trigger with Holdoff by Events (in this example, two events). The bold edges on the trigger source indicate that a positive slope has been selected. The broken, upward-pointing arrows indicate potential triggers, while the bold ones show where triggers actually occur after the holdoff expires.

Simple Triggers

Edge Trigger on Simple Signals

The instrument uses many waveform capture techniques that trigger on features and conditions that you define. These triggers fall into two major categories:

- **Edge** activated by basic waveform features or conditions such as a positive or negative slope, and hold-off
- **SMART Trigger®** sophisticated triggers that enable you to use basic or complex conditions for triggering.

Use Edge Triggers for simple signals, and the SMART Triggers for signals with rare features, like glitches.

Control Edge Triggering

Horizontal: Turn the Delay knob in the HORIZONTAL control group to adjust the trigger's horizontal position. Or, touch inside the **Delay** field in the timebase setup dialog and enter a value, using the pop-up keypad.

The trigger location is shown by a marker under the grid:



Post-trigger delay is indicated by a left-pointing arrow below-left of the grid:



The time value is given in the title line of the **TimeBase** label below-right of the grid:



Vertical: Turn the **LEVEL** knob 

in the TRIGGER control group to adjust the trigger's vertical threshold.
Turn this knob to adjust the level of the trigger source or the highlighted trace. Level defines the source voltage at which the trigger will generate an event a change in the input signal that satisfies the trigger conditions.
Alternatively, in the "Trigger" dialog, you can touch inside the **Level** field and type in a value, using the pop-up numeric keypad. To quickly set a level of zero volts, touch the **Zero Level** button directly below the **Coupling** field.
An arrow on the left side of the grid shows the threshold position. This arrow is only visible if the trigger source is displayed.

Edge Trigger Setup

Channel Setup

1. In the menu bar, touch **Trigger**, and then select **Trigger Setup...** from the drop-down menu.
2. Touch the **Edge** trigger button 

under the **Trigger** tab.

3. Touch inside the **Trigger On** data entry field and select an input from the pop-up menu:



4. Touch inside the **Level** data entry field 

. In the pop-up numeric keypad, enter a value in millivolts or use the up/down buttons  to increase or decrease the value in increments of 1 mV. Or, touch one of the preset value buttons:



Max.	1.000 V
Default	0 mV
Min.	1.000 V

5. Select the holdoff by touching the **Time** or **Events** buttons 

. Using the pop-up numeric keypad, enter a value and specify the unit of time:



- ***Or***, use the up/down buttons  to increase or decrease the time value in increments of 200 ps. Or, touch one of the preset value buttons:



The preset **Time** values are as follows:

Max.	20.0 s
Default	50.0 ns
Min.	2 ns

The preset **Events** values are as follows:

Max.	1,000,000,000 events
Default	1 event
Min.	1 event

6. Choose **Positive** or **Negative** slope



SMART Triggers

Width Trigger

How Width Trigger Works

Width trigger allows you to define a positive- or negative-going pulse width bounded by a voltage level, above or below which a trigger will occur. You can specify a pulse width and voltage range, within or outside of which a trigger will occur.

Width Trigger Setup

1. In the menu bar, touch **Trigger**, then **Trigger Setup...** in the drop-down menu.
2. Touch the **Width** trigger button 

3. Touch inside the **Trigger On** data entry field and select a source on which to trigger:
4. Touch inside the **Level** data entry field and enter a value using the pop-up numeric keypad.
5. Select positive or negative slope.
6. Touch the **LessThan** button and enter a pulse-width value in the **Upper Limit** data entry field. Or touch the **GreaterThan** button and enter a pulse-width value in the **Lower Limit** data entry field. Or touch the **InRange** button. Touch the **Delta** button 

to set up a nominal range, plus or minus a delta value in seconds. Touch inside the **Nominal Width** and **Delta** data entry fields and enter values using the pop-up numeric keypads. Alternately, touch the **Limits** button 

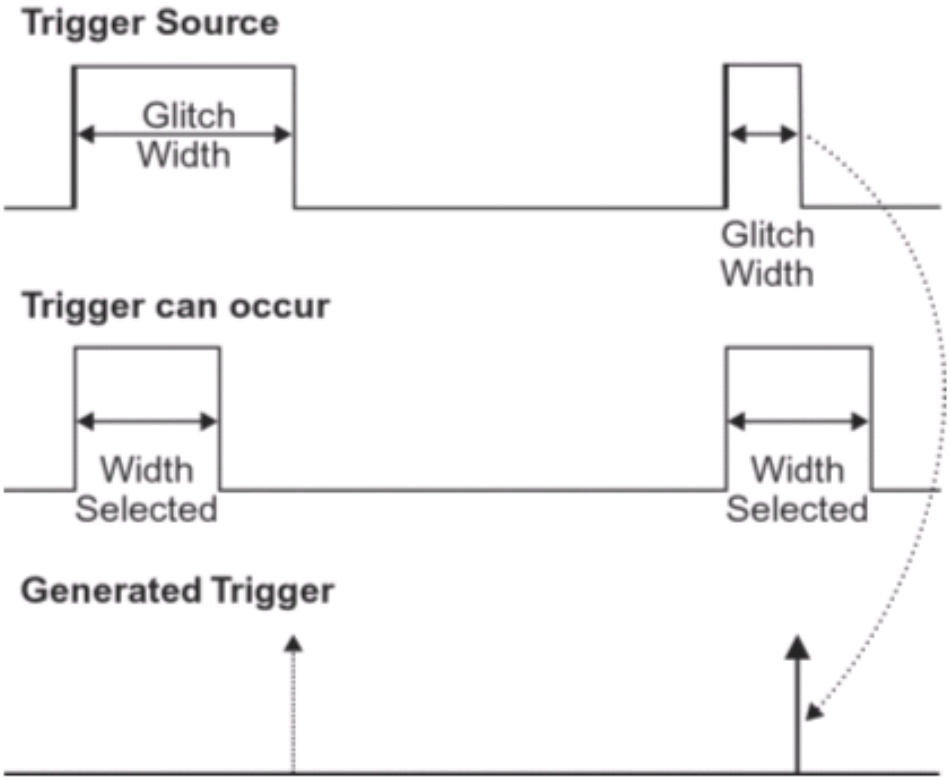
to set up a precise pulse-width range. Touch inside the **Lower Limit** and **Upper Limit** data entry fields and enter values using the pop-up keypads. Or touch the **OutOfRange** button and perform the same range setups as for InRange triggering.

Glitch Trigger

How Glitch Trigger Works

Glitch trigger can be used to catch glitches. You can specify a pulse width or a pulse width range.
Pulse smaller than selected pulse width: Set a maximum pulse width. This glitch trigger is generated on the selected edge (positive or negative) when the pulse width is less than or equal to the set width.
The timing for the width is initialized and restarted on the opposite slope to that selected. You can set widths from 600 ps to 20 s.

Note: If the glitch's width is narrower than the signal's width, set the trigger to a narrower width than that of the signal. The signal's width, as determined by the instrument trigger comparator, depends on the DC trigger level. If that level were to be set at the middle of a sine wave, for example, the width could then be considered as the half period. But if the level were higher, the signal's width would be considered to be less than the half period.



Glitch Trigger: In this example triggering on a pulse width less than or equal to the width selected. The broken upward-pointing arrow indicates a potential trigger, while the bold one shows where the actual trigger occurs.

Glitch Trigger Setup

1. In the menu bar, touch **Trigger**, then **Trigger Setup...** in the drop-down menu.
2. Touch the **Glitch** trigger button 

.
3. Touch inside the **Trigger On** data entry field and select a source on which to trigger.
4. Touch inside the **Level** data entry field and enter a value using the pop-up numeric keypad.
5. Select positive or negative slope.
6. Define the width of the glitch you are looking for. You can trigger on any glitch less than a chosen pulse-width (**Upper Limit**); or you can trigger on a chosen range (**InRange**). Touch the **LessThan** button; the **Upper Limit** data entry field alone is displayed. Touch the **InRange** button; the **Upper Limit** and **Lower Limit** fields are displayed.
7. Touch inside the limit field or fields and enter a time value using the pop-up numeric keypad.

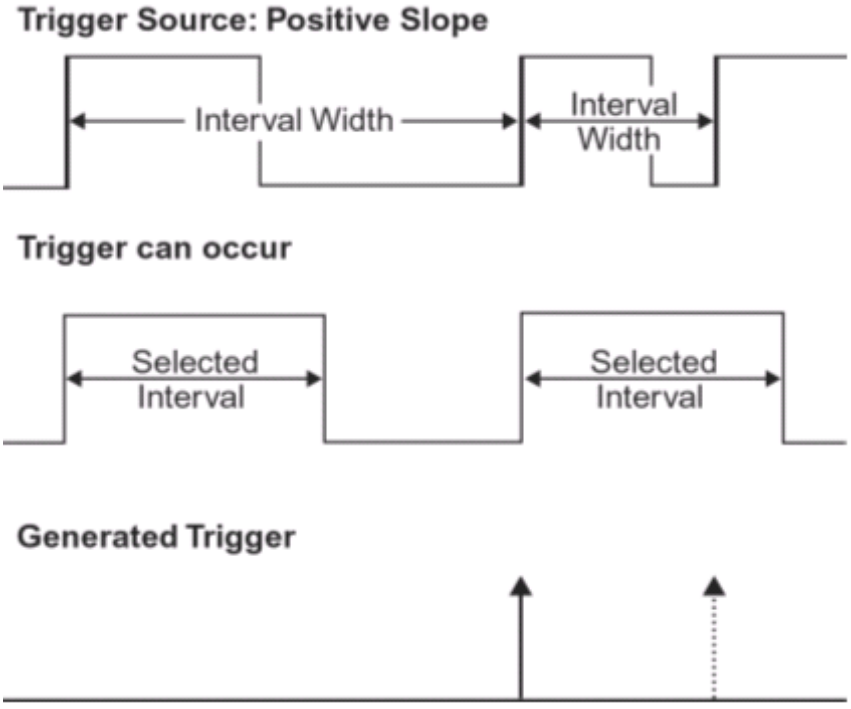
Interval Trigger

How Interval Triggers Work

While Glitch trigger performs over the width of a pulse, Interval trigger performs over the width of an interval, with the signal duration (period) separating two consecutive edges of the same polarity: positive to positive or negative to negative. Use Interval trigger to capture intervals that fall short of, or exceed, a given time limit. In addition, you can define a width range to capture any interval that is itself inside or outside the specified range: an exclusion trigger by interval.

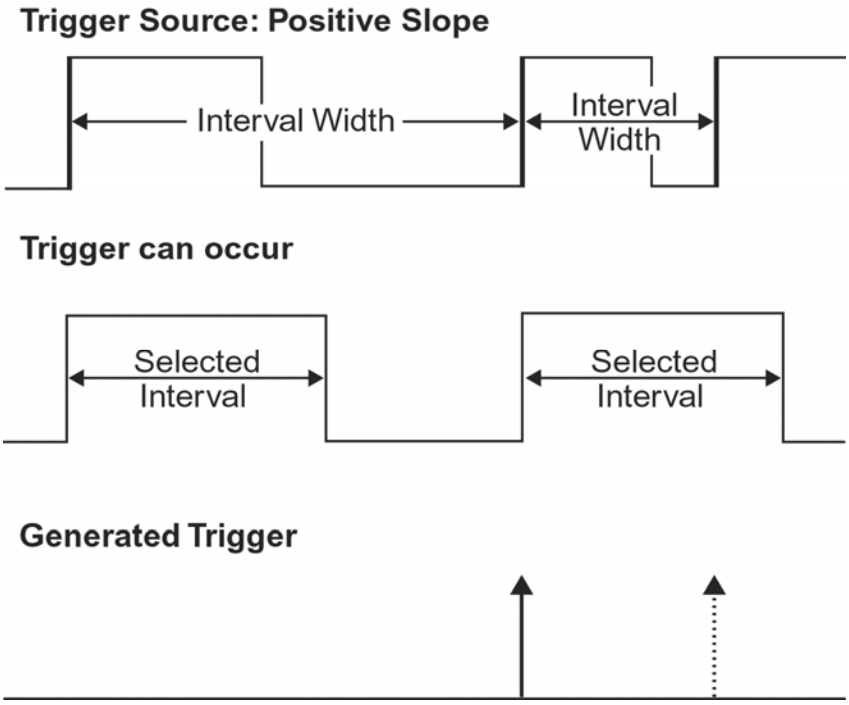
Interval Less Than: For this Interval Trigger, generated on a time interval smaller than the one that you set, choose a maximum interval between two like edges of the same slope (positive, for example).

The trigger is generated on the second (positive) edge if it occurs within the set interval. The instrument initializes and restarts the timing for the interval whenever the selected edge occurs. You can set an interval from 2 ns to 20 s.



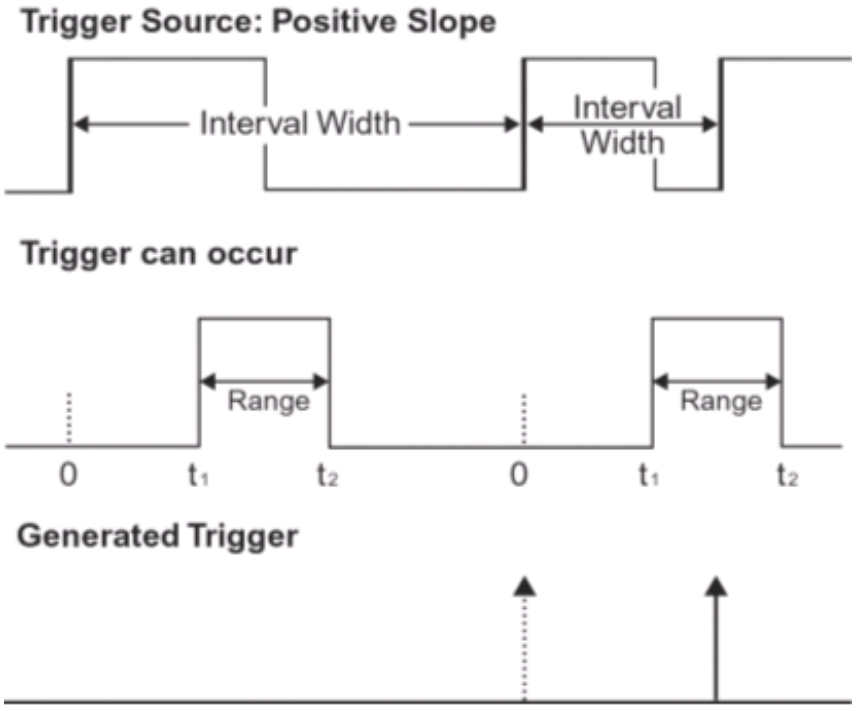
Interval Trigger that triggers when the interval width is smaller than the selected interval. The broken, upward-pointing arrow indicates a potential trigger, while the bold one shows where the actual trigger occurs on the positive edge within the selected interval.

Interval Greater Than: For this Interval Trigger, generated on an interval larger than the one that you set, select a minimum interval between two edges of the same slope. The instrument generates the trigger on the second edge if it occurs after the set interval. The timing for the interval is initialized and restarted whenever the selected edge occurs. You can set an interval from 2 ns to 20 s.



Interval Trigger that triggers when the interval width is larger than the set interval. The broken upward-pointing arrow indicates a potential trigger, while the bold one shows where the actual trigger occurs on the positive edge after the selected interval.

Interval In Range: This Interval Trigger is generated whenever an interval between two edges of the same slope falls within a selected range. The instrument initializes and restarts the timing for the interval whenever the selected edge occurs. You can set an interval from 2 ns to 20 s.



Interval Trigger that triggers when the interval falls within the selected range:
t1 = range's lower time limit; t2 = range's upper limit. The broken upward-pointing arrow indicates a potential trigger, while the bold one indicates where the actual trigger occurs on the positive edge within the selected range.

Interval Trigger Setup

1. In the menu bar, touch **Trigger**, then **Trigger Setup...** in the drop-down menu.
2. Touch the **Interval** trigger button 

.
3. Touch inside the **Trigger On** data entry field and select a source on which to trigger.
   Touch inside the **Level** data entry field and enter a value using the pop-up numeric keypad.
4. Select positive or negative slope.
5. Touch the **LessThan** button and enter a pulse-width value in the **Upper Limit** data entry field.

   Or touch the **GreaterThan** button and enter a value in the **Lower Limit** data entry field.

   Or touch the **InRange** button.

6. Touch the **Delta** button 

to set up a nominal range, plus or minus a delta value in seconds. Touch inside the **Nominal Width** and **Delta** data entry fields and enter values using the pop-up numeric keypads.

7. Touch the **Limits** button 

to set up a precise range. Touch inside the **Lower Limit** and **Upper Limit** data entry fields and enter values using the pop-up numeric keypads.

   Or touch the **OutOfRange** button and perform the same Delta or Limits setup as for InRange triggering.

Qualified Trigger

How Qualified Triggers Work

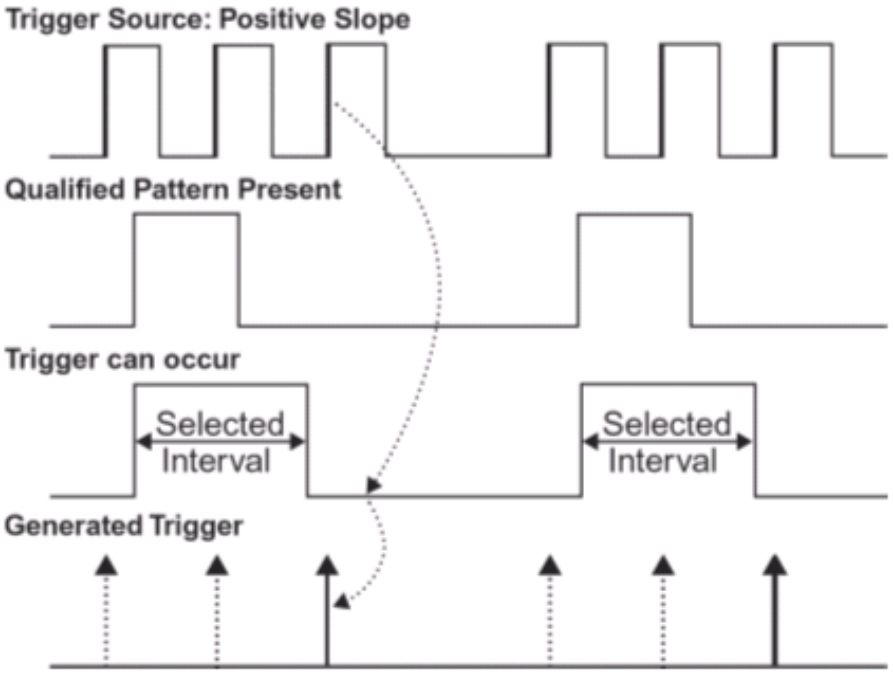
Use a signals transition above or below a given level (its validation) as an enabling (qualifying) condition for a second signal that is the trigger source. These are Qualified triggers. For Edge Qualified triggers (the default) the transition is sufficient and no additional requirement is placed on the first signal. For State Qualified triggers the amplitude of the first signal must remain in the desired state until the trigger occurs. A qualified trigger can occur immediately after the validation, or following a predetermined time delay or number of potential trigger events. The time delay or trigger count is restarted with every validation.


Within Time creates a time window within which a trigger can occur.


Wait Time determines a delay from the start of the desired pattern. After the delay (timeout) and while the pattern is present, a trigger can occur. The timing for the delay is restarted when the selected pattern begins.



Events determines a minimum number of events of the trigger source. An event is generated when a trigger source meets its trigger conditions. On the selected event of the trigger source and while the pattern is present, a trigger can occur. The count is initialized and started whenever the selected pattern begins, and continues while the pattern remains. When the selected count is reached, the trigger occurs.



Edge Qualified and Wait: Trigger after timeout. The broken upward-pointing arrows indicate potential triggers, while the bold ones show where the actual triggers occur.

Qualified First Trigger

Qualified First trigger is intended to be used exclusively in Sequence Mode to speed up the trigger rate. With Qualified First trigger, a single valid trigger is sufficient to acquire a full sequence. Other than in Sequence Mode, Qualified First is identical to the Qualified triggers. In data storage applications, the index pulse can be defined as the qualifier signal and the servo gate signal as the trigger source.

Edge Qualified Trigger Setup

1. In the menu bar, touch **Trigger**, then **Trigger Setup...** in the drop-down menu.
2. Touch the **Qualify** trigger button 

.
3. Touch inside the **Trigger On** data entry field and select a source on which to trigger. Select **Positive** or **Negative** slope.

4. Touch inside the **After** data entry field and select the qualifying signal source from the pop-up menu. If you select an input channel or external source, touch inside the **has gone** data entry field and select a logic level: **Above** or **Below**. Then touch inside the **Level** field and set a voltage level using the pop-up numeric keypad. If you select **Pattern** from the pop-up menu, touch the **Pattern** tab and choose a logic gate. Then touch inside the **State** field for each channel input you want to use in the pattern and select a logic condition: **High** or **Low**. Select **Don't Care** for unused inputs. For the inputs to be used, touch inside each **Level** field and enter a voltage threshold using the pop-up numeric keypad. Then touch the **Trigger** tab again.
5. If you want to set a holdoff in time or events, touch one of the **Qualify by:** buttons:

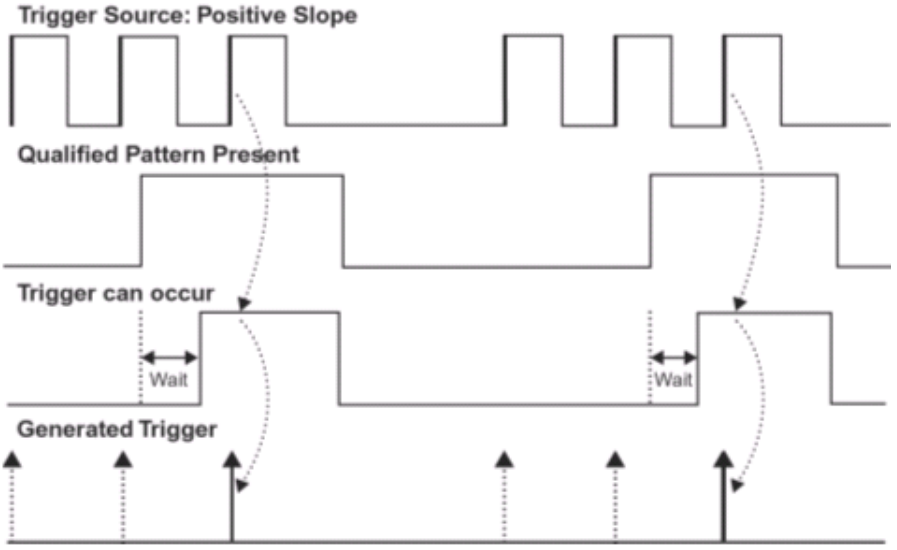


6. Touch inside the field below the **Qualify by:** buttons and enter a value using the numeric keypad.
7. To set up a Qualified First trigger, touch the **Qualify first segment only** checkbox if you are in Sequence mode.

State Trigger

State trigger is another Qualified trigger; however, instead of using the edges of the qualifying inputs, State trigger uses the logic state of the inputs to qualify the trigger. Therefore, the pattern must become true and remain true (for a period of time or number of events that you specify) to qualify the trigger.

See also the How Qualified Triggers Work topic (previous).



State Qualified and Wait: Trigger after timeout. The broken upward-pointing arrows indicate potential triggers, while the bold arrows show where the actual triggers occur.

State Qualified Trigger Setup

1. In the menu bar, touch **Trigger**, then **Trigger Setup...** in the drop-down menu.
2. Touch the **State** trigger button 

.
3. Touch inside the **Trigger On** data entry field and select a source on which to trigger:



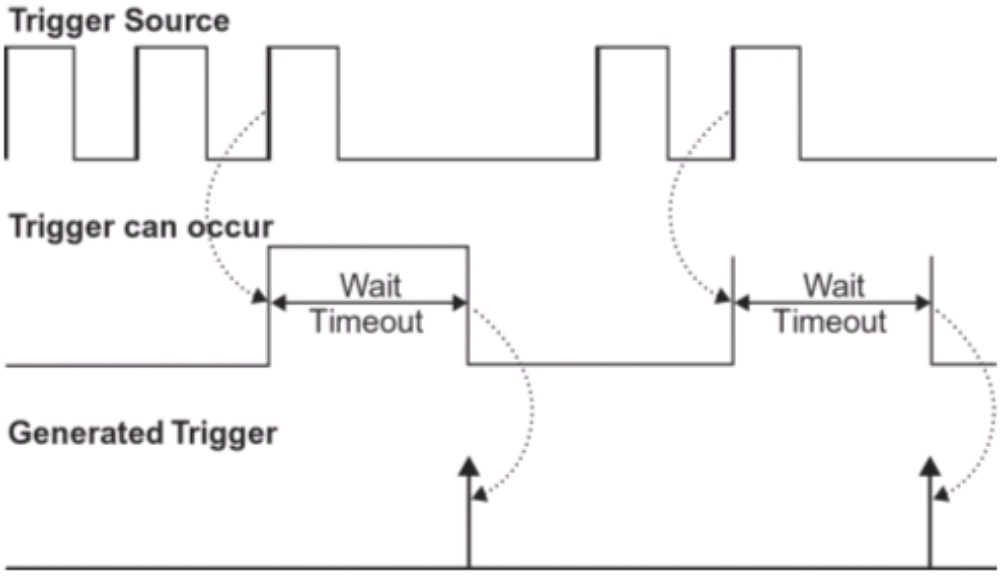
4. Select **Positive** or **Negative** slope.
5. Touch inside the **has gone** data entry field and select the qualifying signal source from the pop-up menu. If you select an input channel or external source, touch inside the **has gone** data entry field and select a logic level: **Above** or **Below**. Then touch inside the **Level** field and set a voltage level using the pop-up numeric keypad. If you want to set a holdoff in time or events, touch one of the holdoff buttons: 

6. Touch inside the field below the holdoff buttons and set a value using the numeric keypad.

Dropout Trigger

Used primarily in single-shot applications, and usually with a pre-trigger delay, Dropout trigger can detect lost signals. The trigger is generated at the end of the timeout period following the last trigger source transition. You can set a timeout period from 2 ns to 20 s.

How Dropout Trigger Works



Dropout Trigger: occurs when the timeout has expired. The bold upward-pointing arrows show where the trigger occurs.

Dropout Trigger Setup

1. In the menu bar, touch **Trigger**, then **Trigger Setup...** in the drop-down menu.
2. Touch the **Dropout** trigger button.



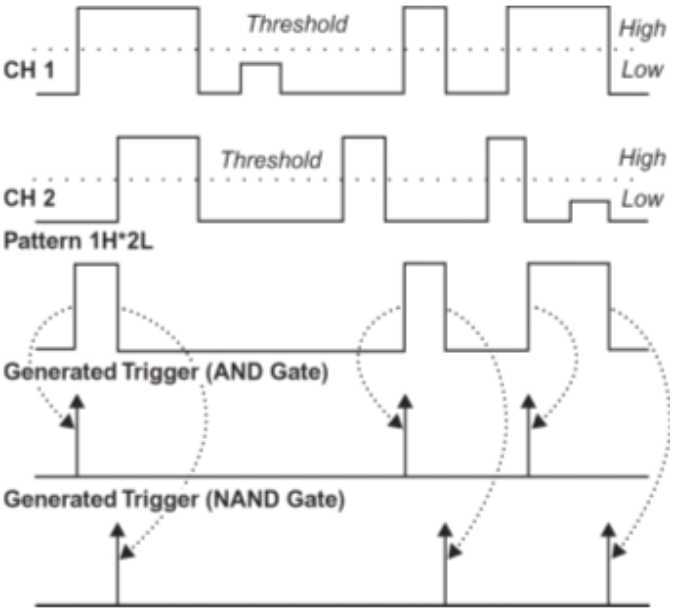
3. Select **Positive** or **Negative** slope.
4. Touch inside the **Trigger after timeout** data entry field and enter a time window using the pop-up numeric keypad.

Logic Trigger

How Logic Trigger Works

Logic Trigger enables triggering on a logical combination of up to five inputs: CH 1, CH 2, CH 3, CH 4, and EXT. The combination of inputs is referred to as a pattern. There are four logic gates available: AND, NAND, OR, NOR.
A trigger state is either high or low: high when a trigger source is greater than the trigger level (threshold) and low when less than it. For example, an AND pattern could be defined as true when the trigger state for CH 1 is high, CH 2 is low, and EXT is irrelevant (X or don't care). If any one of these conditions is not met, the pattern state is considered false. You can set holdoff limits from 2 ns to 20 s or from 1 to 1,000,000,000 events.

Logic Applications



Logic Trigger can be used in digital design for the testing of complex logic inputs or data transmission buses.

Logic Trigger Setup

1. In the menu bar, touch **Trigger**, then **Trigger Setup...** in the drop-down menu.
2. Touch the **Logic** trigger button.



3. Touch the Pattern tab.
4. For each input you want to include in the logic pattern, touch inside the **State** data entry field and select a logic state: Low or High. Select **Don't Care** for all other inputs.

5. Touch inside the **Level** data entry field for each input included in the pattern and enter a voltage level threshold using the pop-up numeric keypad.
6. Touch the **Trigger** tab.
7. If you want to hold off the trigger (either in time or events) when the pattern becomes true, touch one of the holdoff buttons:



8. Touch inside the holdoff data entry field and enter a value using the pop-up numeric keypad.

Serial Trigger

Serial Trigger is only available on SDA "A" model scopes.

Aux Input Trigger

Some instrument models give you the capability to trigger on an auxiliary input. When you select this option, the auxiliary trigger setup is routed to channel 3, and an information icon appears in the Channel 3 descriptor label.



CAUTION

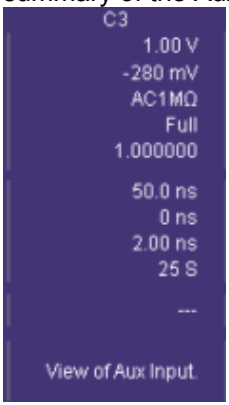
If you select Aux Input trigger on a WavePro 7000 Series scope, but do not input an external signal, the scope will not operate.

Aux Input Setup

1. Touch **Trigger** in the menu bar, then **Trigger Setup...** in the drop-down menu.
2. Touch inside the **View AUX IN on Channel 3** check-box 

. The **Coupling** field is then disabled.
3. Press the Channel 3 front panel button to turn on Channel 3 and display the setup dialog.
4. Perform vertical setups for your auxiliary input in the Channel 3 dialog.

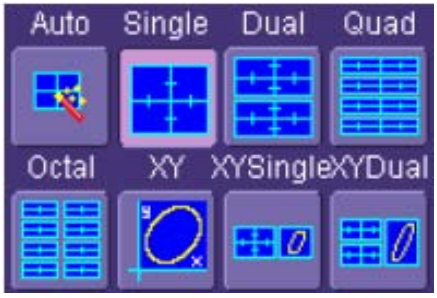
5. Touch **Vertical** in the menu bar, then **Channels Status...** in the drop-down menu to view a summary of the Aux Input setup:



DISPLAY FORMATS

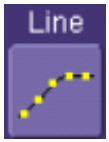
Display Setup

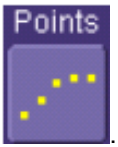
1. In the menu bar, touch **Display;** then touch **Display Setup** in the drop-down menu.
2. Touch one of the **Grid** combination buttons:



As selections are made, **Autogrid** automatically shows or removes grids from the scope display.

3. Touch inside the grid **Intensity** data entry field and enter a value from 0 to 100 using the pop-up keypad.
4. Touch the **Grid on top** checkbox if you want to superimpose the grid over the waveform. Depending on the grid intensity, some of your waveform may be hidden from view when the grid is placed on top. To undo, simply uncheck **Grid on top**.
5. Touch the **Axis labels** checkbox to permanently display the values of the top and bottom grid lines (calculated from volts/div) and the extreme left and right grid lines (calculated from the timebase).

6. Choose a line style for your trace: solid **Line** 

or **Points** 

.

Sequence Mode Display

To a set up Sequence Mode display, you must first have selected **Sequence** trigger mode in the Timebase "Horizontal" dialog. You must also have entered a **Num Segments** value.

1. In the menu bar, touch **Display;** then touch **Display Setup** in the drop-down menu.
2. Touch inside the **Display Mode** field and select a display mode from the pop-up menu.
3. Touch inside the **Num seg displayed** field and enter a value, using the pop-up keypad. The maximum number of segments that can be displayed is 80.
4. Touch inside the **Starting at** field and enter a value.

Note: The maximum value that you can enter for **Starting at** depends on the **Num Segments** value you entered in the "Timebase" dialog. It also depends on the **Num seg displayed** value you entered here in the "Display" dialog. For example, if you had entered a value of 500 in **Num Segments**, and a value of 10 in **Num seg displayed**, the maximum value you can enter as a starting segment is 491so that 10 segments can be seen.

Persistence Setup

The analog Persistence feature helps you display your waveform and reveal its idiosyncrasies or anomalies for a repetitive signal. Use Persistence to accumulate on-screen points from many acquisitions to see your signal change over time. The instrument persistence modes show the most frequent signal path "three-dimensionally" in intensities of the same color, or graded in a spectrum of colors.
You can show persistence for up to eight inputs for any channel, math function, or memory location (M1 to M4).

Saturation Level

The Persistence display is generated by repeated sampling of the amplitudes of events over time, and the accumulation of the sampled data into "3-dimensional" display maps. These maps create an analog-style display. User-definable persistence duration can be used to view how the maps evolve proportionally over time. Statistical integrity is preserved because the duration (decay) is proportional to the persistence population for each amplitude or time combination in the data. In addition, the instrument gives you post-acquisition saturation control for a more detailed display.

When you select 

mode from the Persistence dialog (with **All Locked** selected), each channel is assigned a single color. As a persistence data map develops, different intensities of that color are assigned to the range between a minimum and a maximum population. The maximum population automatically gets the highest intensity, the minimum population gets the lowest intensity, and intermediate populations get intensities in between these extremes.
The information in the lower populations (for example, down at the noise level) could be of greater interest to you than the rest. The Analog persistence view highlights the distribution of data so that you can examine it in detail.

You can select a saturation level as a percentage of the maximum population. All populations above the saturation population are then assigned the highest color intensity: that is, they are saturated. At the same time, all populations below the saturation level are assigned the remaining intensities. Data populations are dynamically updated as data from new acquisitions is accumulated.

Color mode persistence, selected by touching 

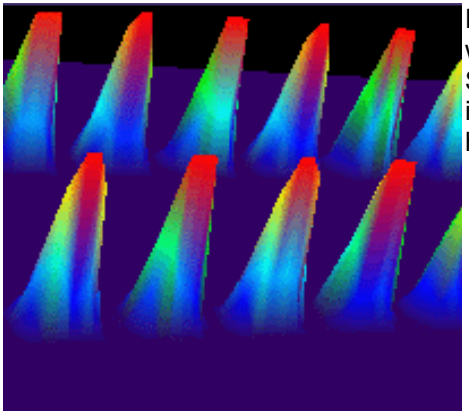
, works on the same principle as the Analog persistence feature, but instead uses the entire color spectrum to map signal intensity: violet for minimum population, red for maximum population. A saturation level of 100% spreads the intensity variation across the entire distribution; at lower saturation levels the intensity will saturate (become the brightest color) at the percentage value specified. Lowering this percentage causes the pixels to be saturated at a lower population, and makes visible those rarely hit pixels not seen at higher percentages.

3-Dimensional Persistence

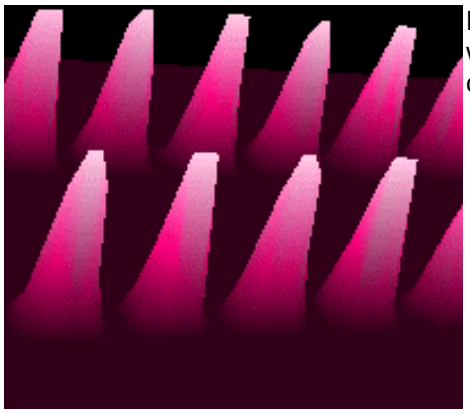
By selecting **3d** 

, you can create a topographical view of your waveform from a selection of shadings, textures, and hues. The advantage of the topographical view is that areas of highest and lowest intensity are shown as peaks and valleys, in addition to color or brightness. The shape of the peaks (pointed or flat) can reveal further information about the frequency of occurrences in your waveform.

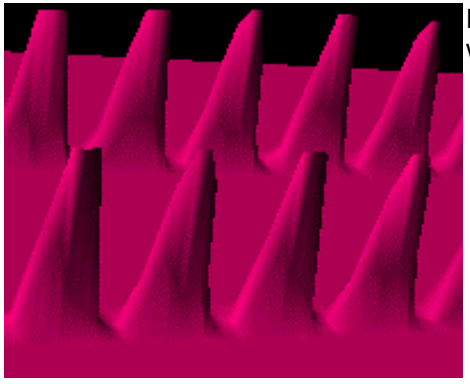
The instrument also gives you the ability to turn the X and Y axes of the waveform through 180° of rotation from -90° to +90°.



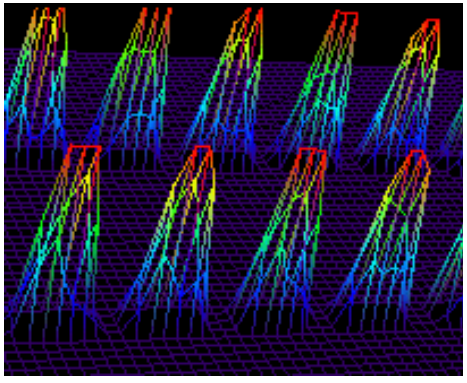
Here is an example of a 3-dimensional view of a square wave using the **solid** view of color-graded persistence. Saturation is set at 50%, with red areas indicating highest intensity. The X-axis has been rotated 60%; the Y-axis has been rotated 15%.



Here is a **monochrome** (analog) view of the same waveform. The lightest areas indicate highest intensity, corresponding to the red areas in the solid view.



Here is a **shaded** (projected light) view of the same waveform. This view emphasizes the shape of the pulses.



Here is a **wire frame** view of the same waveform in which lines of equal intensity are used to construct the persistence map.

Show Last Trace

For most applications, you may not want to show the last trace because it will be superimposed on top of your persistence display. In those cases turn off **Show Last Trace** by touching the checkbox. However, if you are doing mask testing and want to see where the last trace is falling, turn **Show Last Trace** on.

Persistence Time

You can control the duration of persistence by setting a time limit, in seconds, after which persistence data will be erased: 0.5 s, 1 s, 2 s, 5 s, 10 s, 20 s, or infinity.

Locking of Traces

The instrument gives you the choice of constraining all input channels to the same mode, saturation level, persistence time, and last trace display, or setting these for each input channel individually. Choose 

to constrain input channels. Choose 

to set up input channels individually.

Persistence Setup

1. In the menu bar touch **Display**, and then touch **Persistence Setup...** in the drop-down menu.
2. Touch the **Persistence On** checkbox. If **Per Trace** is selected, touch the **Reset All** button 

to return all input channel setups to their default settings.
3. Touch the **All Locked** button 

if you want to set the same mode, saturation level, persistence time, and last trace display for all input channels. Touch the **Per Trace** button 

to set these for each input channel individually.
4. If you selected **All Locked**, touch one of the mode buttons 

.
5. Then touch the **Show last trace** checkbox if you want the last trace displayed.
6. Touch inside the **Saturation** data entry field and enter a whole number integer, using the pop-up numeric keypad.
7. Touch inside the **Persistence time** data entry field and make a selection from the pop-up menu.
8. If you selected **Per Trace**, for each input channel touch its tab, then make selections of mode, saturation level, persistence time, and last trace display in the same way as for **All Locked**.
9. To create a 3-dimensional view, touch the 3d button 

. Then
10. Touch inside the **Saturation** data entry field and enter a whole number integer, using the pop-up numeric keypad.
11. Touch inside the **Persistence time** data entry field and make a selection from the pop-up menu.

12. Under "3D settings," touch inside the **Quality** field and select an image quality from the pop-up menu: wire frame, solid, or shaded.
13. For each axis, touch inside the data entry field and enter a value from -90° to +90°.
14. To turn off persistence for an individual channel, touch the left-most persistence mode button 

. To turn off persistence for all channels, press the front panel Analog Persist button 

. This button toggles Analog Persistence on and off.

Screen Saver

The Windows screen saver is activated the same way as on a standard PC.

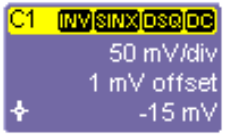
1. Minimize the instrument display by touching **File** in the menu bar, then **Minimize** in the drop-down menu.
2. Touch **Start** down in the task bar.
3. Touch **Settings** in the pop-up menu.
4. Touch **Control Panel**.
5. Touch **Display**.
6. Touch the **Screen Saver** tab.

Moving Traces from Grid to Grid

You can move traces from grid to grid at the touch of a button.

Moving a Channel or Math Trace

1. Touch the descriptor label for the waveform that you want to move.



Example Descriptor Label (previous)

2. Now, touch the **Next Grid** button 

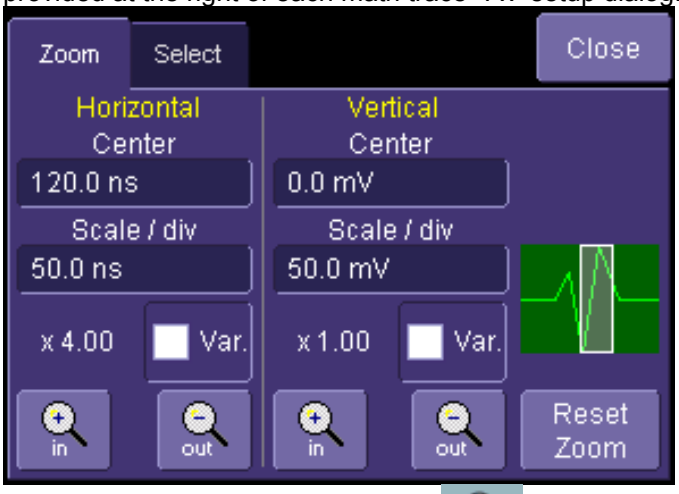
.

Note: If you have more than one waveform displayed on only one grid, a second grid will open automatically when you select **Next Grid**.

Zooming Waveforms

The Zoom button 

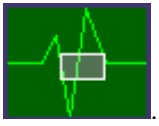
appears as a standard button at the bottom of the channel "Cx Vertical Adjust" setup dialog if you want to create a math function zoom trace of your input waveform. On the other hand, you can zoom a memory or math function non-zoom trace directly without having to create a separate zoom trace. For such traces, a zoom control mini-dialog is provided at the right of each math trace "Fx" setup dialog:



The front panel "QuickZoom" button  creates multiple zooms, one for each displayed input channel.

At any time, you can also zoom a portion of a waveform by touching and dragging a rectangle around any part of the input waveform. The zoom trace will size itself to fit the full width of the grid. The degree of magnification, therefore, will depend on the size of the rectangle that you create.

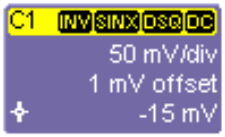
When you zoom a waveform, an approximation of the zoomed area will appear in a thumbnail icon in the "Zoom" dialog:

.

The "Zoom" dialog appears alongside the math setup dialog when Zoom is the math or memory function selected.

Zooming a Single Channel

1. In the menu bar, touch **Vertical**; then touch a channel number in the drop-down menu. Alternatively, you can just touch the channel trace label for a displayed channel.



2. Touch 

at the bottom of the "Cx Vertical Adjust dialog." A zoom math trace (one of F5 to Fx [The number of math traces available depends on the software options loaded on your scope. See specifications.]) will be created of the selected channel.
3. To vary the degree of zoom, touch the newly created Fx trace label. The setup dialog for the math function opens, and the zoom control dialog appears at lower-right. It shows the current horizontal and vertical zoom factors.
4. If you want to increase or decrease your horizontal or vertical zoom in small increments, touch the **Var.** checkbox to enable variable zooming. Now with each touch of the zoom control buttons 

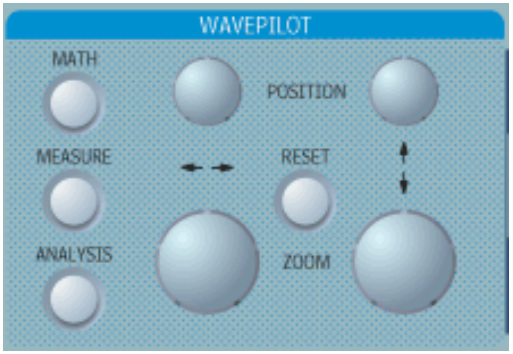
, the degree of magnification will change by a small increment. To zoom in or out in large standard increments with each touch of the zoom control buttons, leave the **Var.** checkbox unchecked. To set exact horizontal or vertical zoom factors, touch inside the Horizontal **Scale/div** data entry field and enter a time-per-div value, using the pop-up numeric keypad. Then touch inside the Vertical **Scale/div** field and enter a voltage value.
5. To reset the zoom to x1 magnification, touch **Reset Zoom** in the dialog or press the front panel zoom button:


.

Zooming by Touch-and-Drag

1. Touch and drag a rectangle around any part of an input channel waveform, math trace, or memory trace. If you have more than one trace displayed, a pop-up "Rectangle Zoom Wizard" will appear.
2. If more than one trace is displayed, touch the "Source" tab and select a trace to act on.
3. Touch the "Action" tab and select **Create a New Zoom Trace**. You will be offered the choice of creating a new zoom trace or modifying the current trace.
4. Touch the **Zoom** tab and select a math function trace to display the zoom.

5. Turn the front panel Wavepilot position knobs to adjust the vertical and horizontal position of the zoom:



6. Turn the front panel Zoom knobs to control the boundaries of the zoom.

Zooming Multiple Waveforms Quickly

Press the QuickZoom button  on the front panel. Math function traces **F5** to **F8** will be used to create a zoom of each displayed input channel waveform. Each zoom will be displayed in its own grid.

Turning Off Zoom

1. Touch the math function trace label for the zoom you want to turn off.
2. Touch the **Trace On** checkbox to delete the check mark and disable the zoom trace.

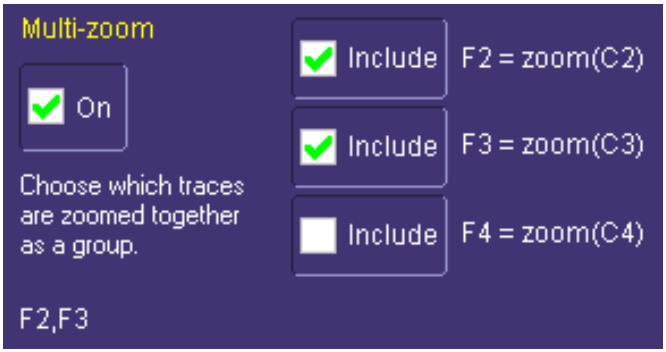
Multi-Zoom

The Multi-zoom feature creates time-locked zoom traces for only the waveforms that you choose to include. The zooms are of the same X-axis section of each waveform. Thus, as you scroll through a waveform, all included zooms scroll in unison.

Setting Up Multi-zoom

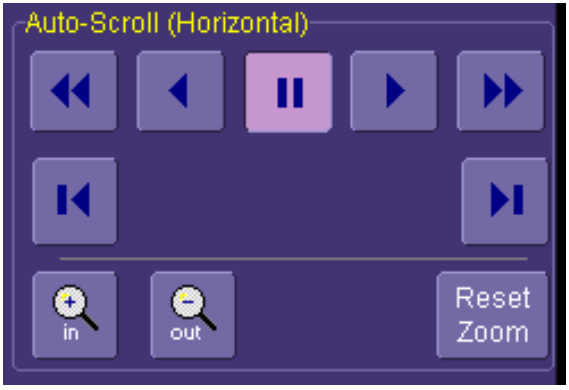
1. In the menu bar, touch **Math**, then **Math Setup...** in the drop-down menu.
2. Verify that the math function selected for each **Fx** position you want to include is **zoom**. If you need to change the math function for any Fx position, simply touch the **Fx** button and select **Zoom** from the **Select Math Operator menu**.
3. Touch the **On** checkbox to display each zoom you want to include in the multi-zoom.
4. Touch the Multi-Zoom **Setup** button. The **Multi-Zoom** dialog opens.

5. Touch the Multi-zoom **On** checkbox to enable Multi-zoom. Then touch the **Include** checkbox for each zoom trace you want to include in the time-locked multi-zoom:



Here the user has chosen to include only F2 and F3 in the Multi-zoom, even though F4 is also a zoom function and is also displayed. Thus, the scrolling feature will not affect zoom F4.

6. Use the **Auto-Scroll** buttons at the right of the Multi-Zoom dialog to control the zoomed section of your waveforms:



Turning Off Multi-Zoom

1. In the menu bar, touch **Math**, then **Math Setup...** in the drop-down menu.
2. Touch the Multi-Zoom **On** checkbox to turn off Multi-zoom.

XY Display

Use XY displays to measure the phase shift between otherwise identical signals. You can display either voltage on both axes or frequency on both axes. The traces must have the same X-axis. The shape of the resulting pattern reveals information about phase difference and frequency ratio.

XY Display Setup

1. In the menu bar, touch **Display**; then touch **Display Setup...** in the drop-down menu.

2. Choose an XY display by touching one of the XY display mode buttons 

. You have the choice of showing the two waveforms on just the XY grid, or you can also show the input waveforms on a single or dual grid.
3. Touch inside the **Input X** and **Input Y** data entry fields and select your input sources from the pop-up menus. The inputs can be any combination of channels, math functions, and memory locations.

SAVE AND RECALL

Saving and Recalling Scope Settings

You can save or recall scope settings to or from hard disk, floppy disk, or LAN location.

Saving Scope Settings

1. In the menu bar, touch **File**; then touch **Save Setup...** in the drop-down menu. Or, press the Save/Recall front panel button, then touch the "Save Setup" tab.
2. To **Save To File**, touch inside the **Save Instrument Settings** data entry field and use the pop-up keyboard to enter the path to the destination folder. Or touch **Browse** to navigate to the destination folder. Then touch 

below the data entry field. To save to folder **Internal Setups** on the scope's hard drive, touch inside a **SetupX** data entry field and use the pop-up keyboard to enter a file name. Touch 

alongside the data entry field. The file is deposited in **D:\Internal Setups**, and the current date is displayed above the field.

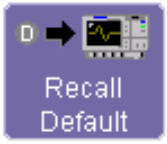
Recalling Scope Settings

1. In the menu bar, touch **File**; then touch **Recall Setup...** in the drop-down menu.
2. To **Recall From File**, touch inside the **Recall panels from file** data entry field and use the pop-up keyboard to enter the path to the source folder. Or touch **Browse** to navigate to the source folder. Then touch 

. To recall settings from folder **D:\ Internal Setups** on the scope's hard drive, touch 

alongside the file you want to recall.

Recalling Default Settings

1. In the menu bar, touch **File**; then touch **Recall Setup...** in the drop-down menu.
2. Touch the button under **Recall Default Setup** 

.

The default settings are as follows:

Vertical	Timebase	Trigger
50 mV/div	50.0 ns/div	DC50 or AC1M (model dependent), C1, 0 mV trigger level
0 V offset	5.0 or 10.0 GS/s (model dependent)	edge trigger positive edge
	0 s delay	Auto trigger mode

Saving Screen Images

You can send images to a hard copy printer or to storage media. Both types of output are done from the same dialog.

1. In the menu bar, touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
2. Touch the **Hardcopy** tab.
3. Touch the **File** button.
4. Touch inside the **File Format** field and select a file type.
5. Under **Colors**, touch the **Use Print Colors** checkbox if you want your waveforms to print in color with a white background. A white background saves printer toner.
6. Touch inside the **Directory** field and type in the path to the directory where you want the image stored, using the pop-up keyboard. Or you can touch the browse button and navigate there.

7. Touch inside the **File Name** field and type in a name for your image, using the pop-up keyboard.
8. Under **Include On Print**, touch the **Grid Area Only** checkbox if you do not want to include the dialog area in the image.
9. Touch the **Print Now** button.

Saving and Recalling Waveforms

Saving Waveforms

1. In the menu bar, touch **File**; then touch **Save Waveform...** in the drop-down menu.
2. In the "Save Waveform" dialog, touch the **Save To** 

or 

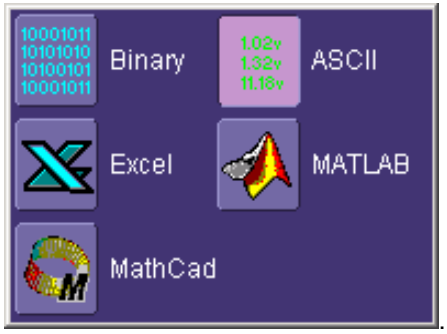
button.
3. Touch inside the **Source** field and select a source from the pop-up menu. The source can be any trace; for example, a channel (C1C4), math function (F1F4), or a waveform stored in non-volatile RAM (M1M4).
4. Touch inside the **Trace Title** data entry field if you want to change the default name of your waveforms. Use the pop-up keyboard to type in the new name.

Note: You can change the name but not the sequence number.

CAUTION

If you use a name that ends in a number instead of a letter, the instrument may truncate the number. This is because, by design, the first waveform is automatically numbered 0, the second 1, etc. For example, if you want to use waveform name "XYZ32" but it is not preceded by waveforms XYZ0 through XYZ31, the waveform will be renumbered with the next available number in the sequence.
If you need to use a number in your waveform's name, it is recommended that you append an alpha character at the end of the number : "XYZ32a" for example.

If you are saving to file, touch the **Data Format** field and select a format type from the pop-up menu:



If you select **ASCII** or **Excel**, also touch the **SubFormat** field and select either **Time Data** or **Time & Ampl**. Then touch the **Delimiter** field and select a delimiter character from the pop-up menu: comma, space, semicolon, or tab.

1. Touch the **Browse** button for the **Save file in directory** field and browse to the location where you want the file saved. The file name is assigned automatically and is shown below the field.
2. Touch 

.

Auto Save

You can also enable Auto Save from this dialog by touching one of the Auto Save buttons:



- **Wrap** (old files overwritten) ***Or***
- **Fill** (no files overwritten).

CAUTION

If you select Fill, you can quickly use up all disk space on your hard disk.

Recalling Waveforms

1. In the menu bar, touch **File**; then touch **Recall Waveform...** in the drop-down menu.
2. In the "Recall Waveform" dialog, touch the **Recall From** 

or 

button.
3. If you selected **Memory**, touch inside the **Source** field and select a memory location: **M1** to **M4**.
4. If you selected **File**, touch inside the **Destination** field and select a memory location in which to store the file.
   a. Touch inside the **Show only files** field and select an area to limit the search to: channels, math functions, or memory.
   b. Touch inside the **Recall files from directory** data entry field and enter the path, using the pop-up keyboard. Or touch the **Browse** button to navigate to the file.
   c. Touch inside the **Next file will be recalled from** data entry field and enter the path, using the pop-up keyboard. Or touch the **Browse** button to navigate to the file.
5. Touch 

.

Disk Utilities

Use the Disk Utilities dialog to delete files or create folders.

Deleting a Single File

1. Touch **File** in the menu bar, then **Disk Utilities...** in the drop-down menu.
2. Touch the **Delete** button  in the "Disk Utilities" dialog.
3. Touch inside the **Current folder** data entry field and use the pop-up keyboard to enter the path to the folder that contains the file you want to delete. Or touch the **Browse** button and navigate to the folder.
4. Touch inside the **File to be deleted** data entry field and use the pop-up keyboard to enter the name of the file. Or touch the **Browse** button and navigate to the file.
5. Once you have located the file, touch the **Delete File** button.

Deleting All Files in a Folder

1. Touch **File** in the menu bar, then **Disk Utilities...** in the drop-down menu.
2. Touch the **Delete** button  in the "Disk Utilities" dialog.
3. Touch inside the **Current folder** data entry field and use the pop-up keyboard to enter the path to the folder that contains the file you want to delete. Or touch the **Browse** button and navigate to the folder.
4. Once you have located the folder, touch the **Empty Folder** button.

Creating a Folder

1. Touch **File** in the menu bar, then **Disk Utilities...** in the drop-down menu.
2. Touch the **Create** button  in the "Disk Utilities" dialog.
3. Touch inside the **Current folder** data entry field and use the pop-up keyboard to enter the path to the directory you want to create the folder in, and the name of the folder.
4. Touch the **Create Folder** button.

PRINTING AND FILE MANAGEMENT

Print, Plot, or Copy

The instrument gives you the ability to output files to a printer or plotter, to print to file, or to e-mail your files. Any WindowsXP supported printer is supported by your instrument.

Printing

Printer Setup

1. In the menu bar, touch **File**, then **Print Setup...** in the drop-down menu. The Utilities **Hardcopy** dialog opens.
2. In the dialog area, touch the **Printer** icon .
3. Under **Colors**, touch the **Use Print Colors** checkbox if you want the traces printed on a white background. A white background saves printer toner. (You can change the printer colors in the Preference dialog;)
4. Touch inside the **Select Printer** field. From the touch pad pop-up choose the printer you want to print to. Touch the **Properties** button to see your printer setup.
5. Touch the icon for the layout **Orientation** you want: portrait or landscape.
6. Touch the **Grid Area Only** checkbox if you do not need to print the dialog area and you only want to show the waveforms and grids.

Printing

You can print in one of three ways:

- Press the printer button on the front panel:
- In the menu bar, touch **File**, then **Print** in the drop-down menu.
- Touch the **Print Now** button in the "Hardcopy" dialog

Adding Printers and Drivers

Note: If you want to add a printer driver, the driver must first be loaded on the scope.

1. In the menu bar, touch **File**, then **Print Setup...** in the drop-down menu. The Utilities **Hardcopy** dialog opens.
2. In the dialog area, touch the **Printer** icon .
3. Touch the **Add Printer** button. An MS Windows® window with which to add a printer will open.
4. Touch the Properties button to change printer properties such as number of copies.

Changing the Default Printer

1. If you want to change the default printer, minimize the instrument application by touching **File** in the menu bar, then **Minimize** in the drop-down menu.
2. Touch the **Start** button in the task bar at the bottom of the screen.
3. Select **Settings**, then **Printers**.
4. Touch the printer you want to set as the default printer, then touch **File, Set as Default Printer**.

Managing Files

Use the instrument's utilities to create waveform files on floppy disk, internal hard drive or network drives. You can copy files from your hard drive to floppy disk. You also can give your files custom names and create directories for them.

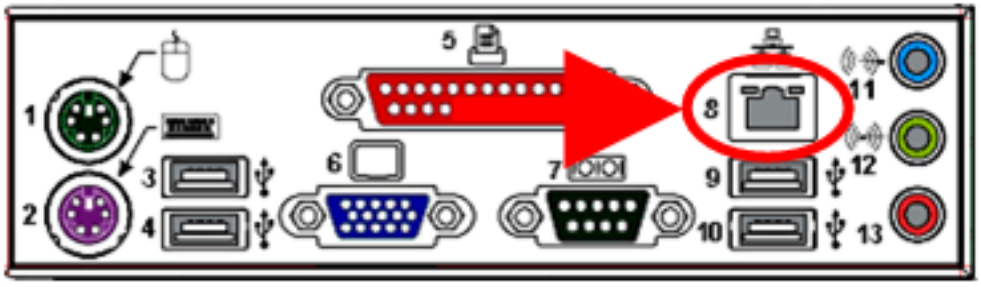
Hard Disk Partitions

The instrument's hard disk is partitioned into drive **C:** and drive **D:**. Drive C: contains the Windows operating system and the instrument application software. Drive D: is intended for data files.

100BASE-T ETHERNET CONNECTION

Connecting to a Network

Use the Ethernet connector (item **8** in the rear panel diagram) to connect the instrument to a network.



Communicating over the Network

In its default configuration the instrument is set up to use the DHCP protocol to retrieve its IP address from the network. In cases where a DHCP server is not available on your network, a static IP address can be configured in the Windows Network Settings dialog.

Windows Setups

Instruments that are required to participate in a Windows Network Domain will need to be "joined" to the domain by a network administrator (the procedure typically requires an administrator username and password).
Domain membership is not required to use the instrument on a network, but will generally make it easier to access network shared drive and printer resources.

Guidelines for Working in Windows

Although the instrument has an open architecture, avoid modifying the Windows operating system, since this may cause problems for the instrument's user interface. Please follow these recommendations:

- Do not load any version of Windows not provided by LeCroy. Windows service packs and critical updates are generally safe, and LeCroy does encourage you to install them to keep your scope safe from network-borne viruses and worms. However, LeCroy cannot guarantee that any update distributed by Microsoft will not adversely affect the operation of your instrument. Any compatibility issues detected by LeCroy will be posted on our DSO Security Web site at www.lecroy.com/dsosecurity. It is advisable to check this site before applying updates.
- If the instrument powers up in Windows Safe Mode, the touch-screen will not function. You may need a mouse or keyboard to restore normal operation.
- Avoid modifying Control Panel settings.
- Do not change the color resolution (24 bit) or screen size (800 x 600 pixel) settings.

- After you load third-party software applications, if your scope does not work properly try reloading the instrument software from the CD shipped with the scope. If your instrument is not equipped with a CD drive, you will need a USB CD-ROM to do this (not supplied by LeCroy). This does not apply to WaveSurfer or WaveRunner 6000A models, which can be recovered from their internal hard drives.
- Do not modify or remove any system fonts; doing so may affect the readability of the dialogs.
- Do not change any display properties like Background, Appearance, Effects, or Settings. Functionality of the scope or screen saver may be affected.
- Do not make any changes to the Windows folder.
- Do not make any changes to the BIOS settings.
- Do not make any changes to the Windows power management system.

System Restore

Although the scope creates regularly scheduled restore points automatically, before you install any hardware or software on your instrument LeCroy strongly recommends that you manually create a restore point. The restore point resides on the scope's hard drive, so no external storage medium (floppy disk, USB memory stick, etc.) is required.

Creating a Restore Point

1. From the **File** menu, minimize or Window the scope display to reveal the task bar.
2. In the task bar, select **Start**, **Programs**, **Accessories**, **System Tools**, **System Restore**.
3. Touch the **Create a restore point** radio button, then touch **Next**.
4. In the **Restore point description** box, indicate what software or hardware is going to be added after the restore point is created, then touch **Next**.
5. The restore point will be created and a confirmation message will be displayed.

TRACK VIEWS

Creating and Viewing a Trend

1. In the menu bar, touch **Measure**, then **Measure Setup** in the drop-down menu.
2. Touch one of parameter tabs **P1** through **Px**.
3. Touch inside the **Source1** data entry field and select an input waveform from the pop-up menu.
4. Touch inside the **Measure** data entry field and select a parameter from the pop-up menu.

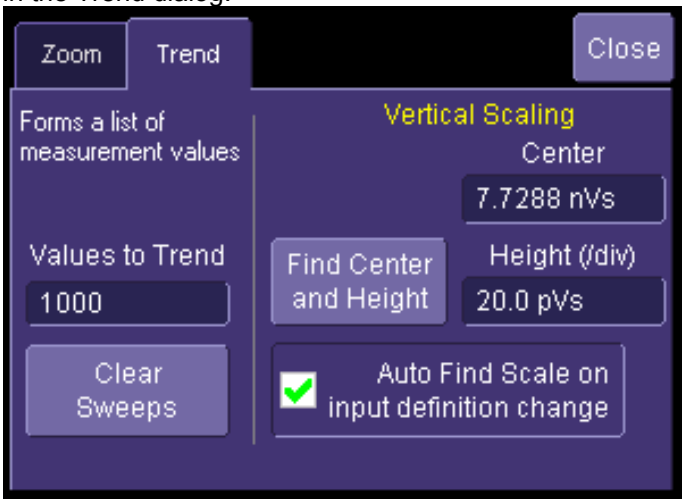
5. Touch the **Trend** button 

at the bottom of the dialog; then, from the **Math selection for Trend** menu, select a math function location (F1 to Fx The number of math traces available depends on the software options loaded on your scope. See specifications.) to store the Trend display. The Trend will be displayed along with the trace label



Example Trend Trace Label for the math function you selected.

6. Touch the newly displayed Trend math function trace label if you want to change any settings in the Trend dialog:



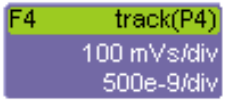
Creating a Track View

This feature is available in the XMAP option.

1. In the menu bar, touch **Measure**, then **Measure Setup** in the drop-down menu.
2. Touch one of parameter tabs **P1** through **Px**.
3. Touch inside the **Source1** data entry field and select an input waveform from the pop-up menu.
4. Touch inside the **Measure** data entry field and select a parameter from the pop-up menu.

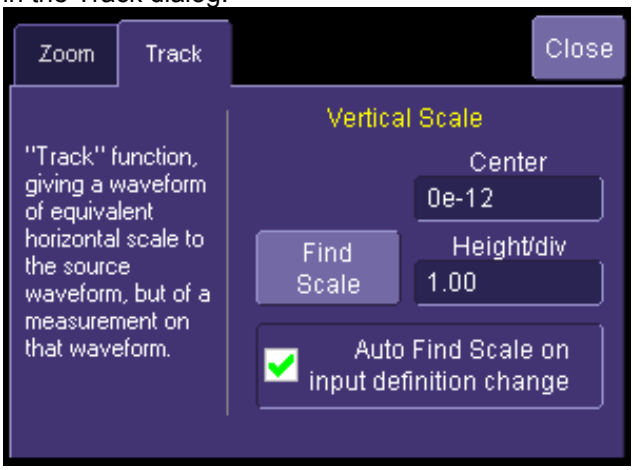
5. Touch the **Track** button 

at the bottom of the dialog; then, from the **Math selection for Track** menu, select a math function location (F1 to Fx The number of math traces available depends on the software options loaded on your scope. See specifications.) to store the Track display. The Track will be displayed along with the trace label



Example Track Trace Label for the math function you selected.

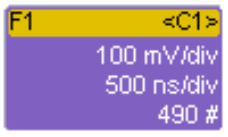
6. Touch the newly displayed Track math function trace label if you want to change any settings in the Track dialog:



HISTOGRAMS

Creating and Viewing a Histogram

Note: The number of sweeps comprising the histogram will be displayed in the bottom line of the trace descriptor label:



Single Parameter Histogram Setup

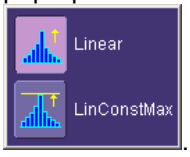
From Measure Dialog

1. In the menu bar, touch **Measure**, then **Measure Setup**.
2. Touch the **My Measure** button.
3. Touch one of tabs **P1** through **Px**.
4. Touch inside the **Source1** field and select an input waveform from the pop-up menu.
5. Touch inside the **Measure** field and select a parameter from the pop-up menu.

6. Touch the **Histogram** button at the bottom of the dialog.
7. Touch a math trace in which to place the resulting histogram, then close the pop-up menu.
8. Touch the math trace label for the math trace you just created.
9. In the dialog to the right, touch the Histogram tab.
10. Under "Buffer," touch inside the #Values data entry field and enter a value.
11. Under "Scaling," touch inside the #Bins data entry field and enter a value from 20 to 2000.
12. Touch the Find Center and Width button to center the histogram. Or touch inside the Center, then the Width, data entry fields and enter a value using the pop-up numeric keypad.

From the Math Dialog

1. In the menu bar, touch **Math**, then **Math Setup**.
2. Touch one of function tabs **F1** through **Fx** The number of math traces available depends on the software options loaded on your scope. See specifications..
3. Touch the **Graph** button .
4. Touch inside the **Source1** field and select a source from the pop-up menu.
5. Touch inside the **Measurement** field and select a parameter from the pop-up menu.
6. Touch inside the **Graph with** field and select **Histogram** from the pop-up menu.
7. In the dialog to the right, touch the **Histogram** tab.
8. Under "Buffer," touch inside the **#Values** data entry field and enter a value from 20 to 1000.
9. Under "Scaling," touch inside the **#Bins** data entry field and enter a value from 20 to 2000.
10. Touch the **Find Center and Width** button to center the histogram. Or touch inside the **Center**, then the **Width**, data entry fields and enter a value using the pop-up numeric keypad.
11. Touch inside the Vertical Scale field and select **Linear** or **Linear Constant Max** from the pop-up menu



Viewing Thumbnail Histograms

Histicons are miniature histograms of parameter measurements that appear below the grid. These thumbnail histograms let you see at a glance the statistical distribution of each parameter.

1. In the menu bar, touch **Measure**, then one of the Measure Mode buttons: **Std Vertical**, **Std Horizontal**, or **My Measure**.
2. Touch the **Histicons** checkbox to display thumbnail histograms below the selected parameters.

Note: For measurements set up in My Measure, you can quickly display an enlarged histogram of a thumbnail histogram by touching the Histicon you want to enlarge. The enlarged histogram will appear superimposed on the trace it describes. This does not apply to "Std Vertical" or "Std Horizontal" measurements.

Persistence Histogram

You can create a histogram of a persistence display also by cutting a horizontal or vertical slice through the waveform. You also decide the width of the slice and its horizontal or vertical placement on the waveform.
This math operation is different than the "Histogram" math operation and is not affected by **Center** and **Width** settings made there.

Persistence Histogram Setup

1. In the menu bar, touch **Math**, then **Math Setup**.
2. Touch one of function tabs **F1** through **Fx** The number of math traces available depends on the software options loaded on your scope. See specifications..
3. Touch inside the **Source1** field and select a source from the pop-up menu.
4. Touch inside the **Operator1** field and select **Phistogram**  from the **Select Math Operator** menu.
5. Touch the "Phistogram" tab, then touch inside the Slice Direction field and select **Horizontal** or **Vertical** slice from the pop-up menu.
6. Touch inside the **Slice Center** field and enter a value, using the pop-up keypad.
7. Touch inside the **Slice Width** field and enter a value, using the pop-up keypad.

Note: You can use the front panel Adjust knobs to move the **Slice Center** line and the **Slice Width** boundary lines.

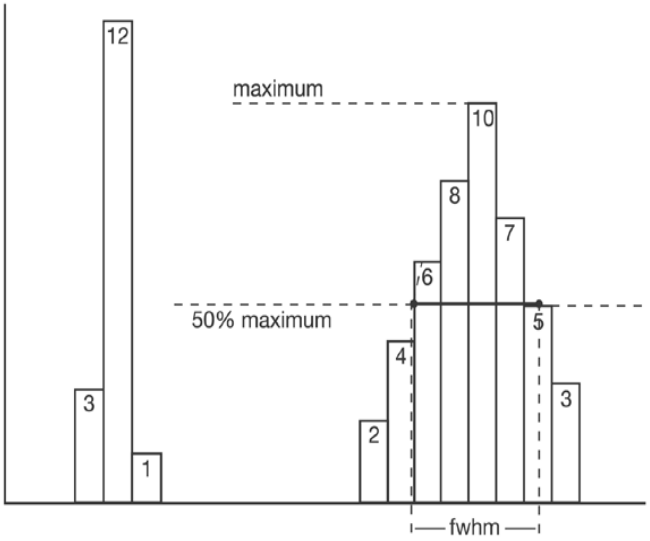
Persistence Trace Range

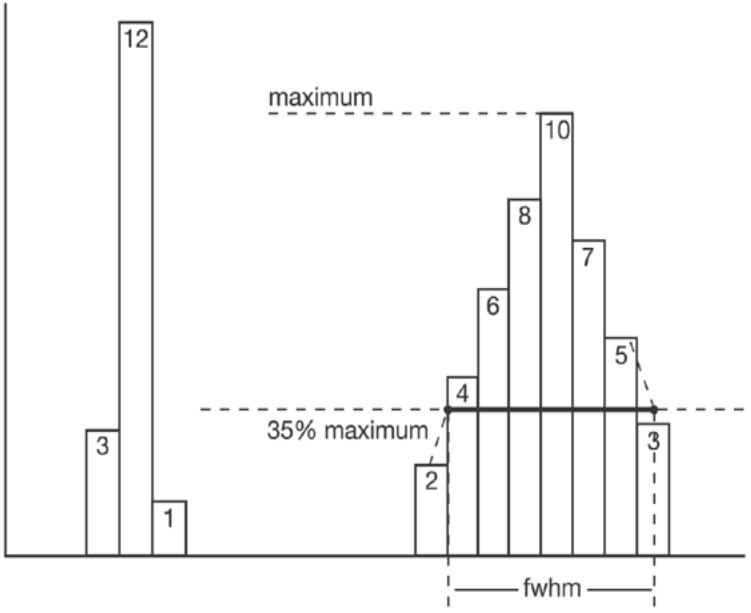
This math operation has a field where you can enter the percent of the persistence trace population to use in creating a new waveform.

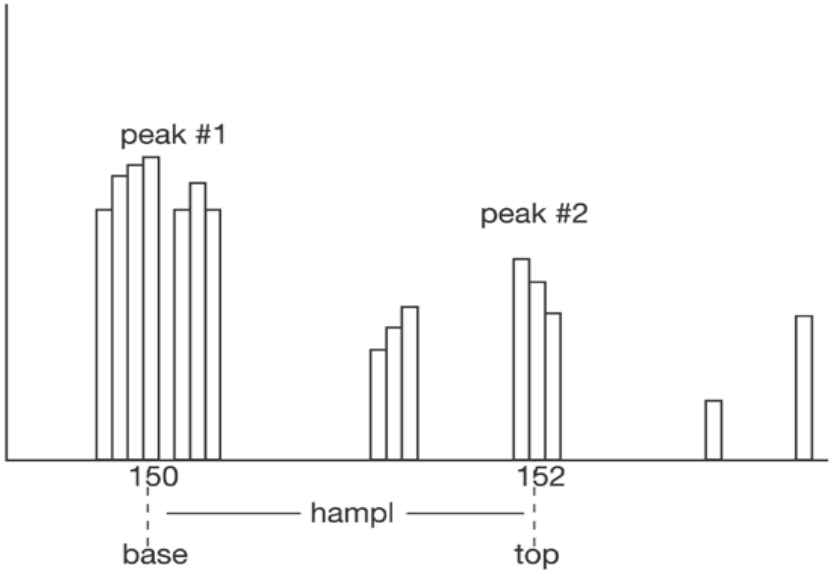
Persistence Sigma

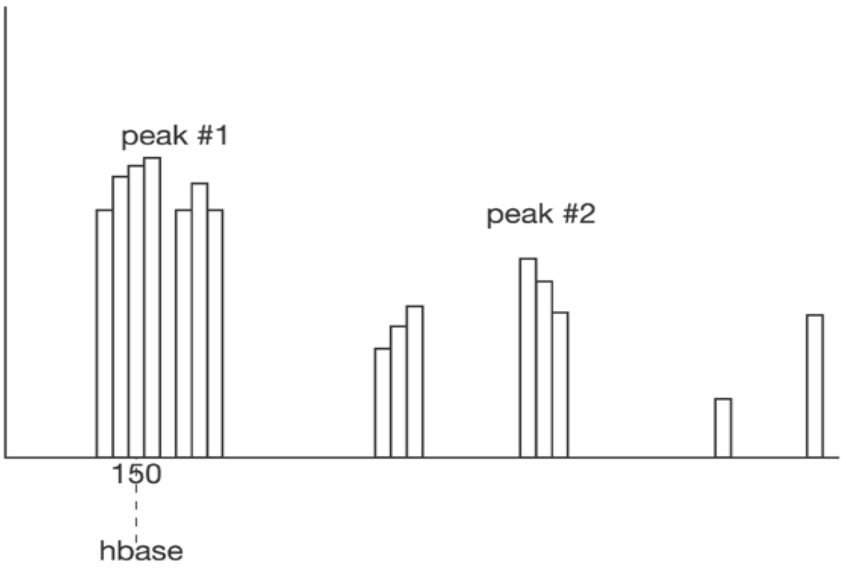
This math operation has a field where you can enter a scale, measured in standard deviations, by which to create a new waveform.

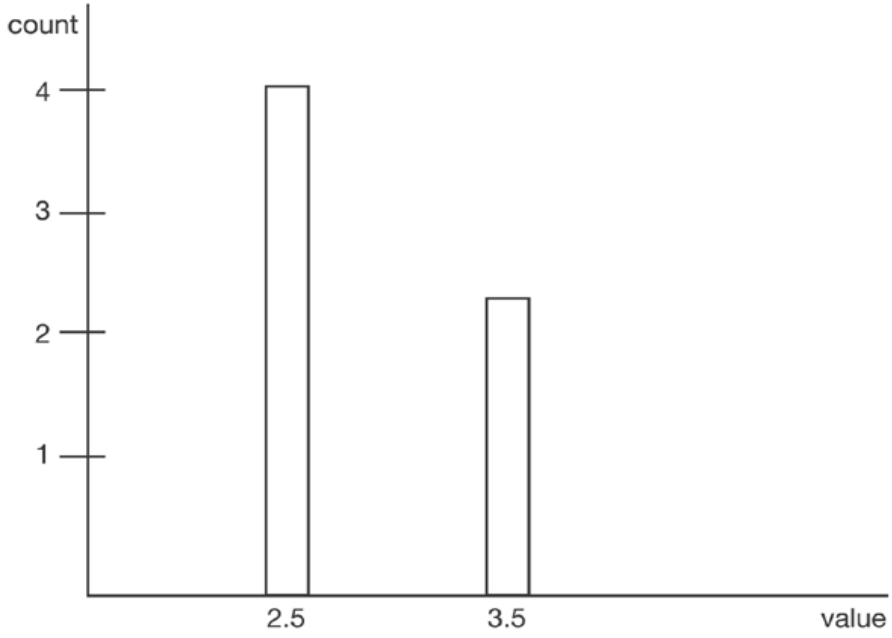
Histogram Parameters

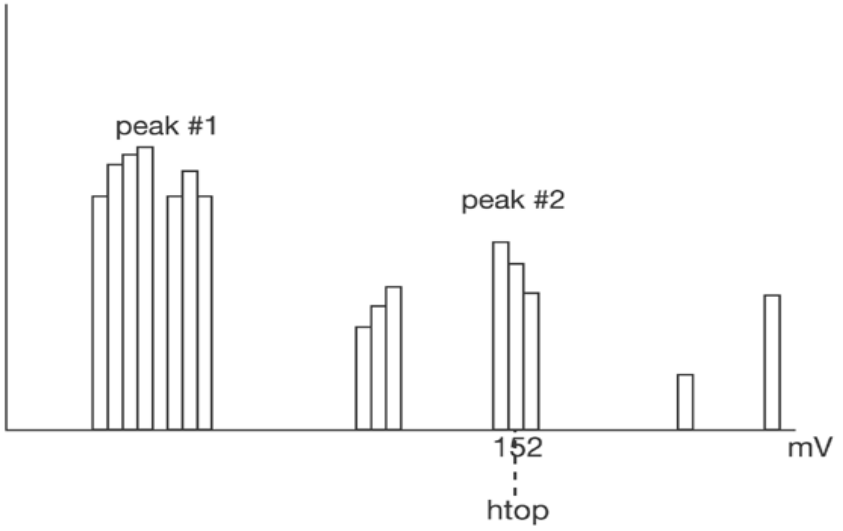
fwhm	Full Width at Half Maximum
Definition:	Determines the width of the largest area peak, measured between bins on either side of the highest bin in the peak that have a population of half the highest's population. If several peaks have an area equal to the maximum population, the leftmost peak is used in the computation.
Description:	First, the highest population peak is identified and the height of its highest bin (population) determined (for a discussion on how peaks are determined see the pks parameter Description:). Next, the populations of bins to the right and left are found, until a bin on each side is found to have a population of less than 50% of that of the highest bin's. A line is calculated on each side, from the center point of the first bin below the 50% population to that of the adjacent bin, towards the highest bin. The intersection points of these lines with the 50% height value is then determined. The length of a line connecting the intersection points is the value for fwhm.
Example:	

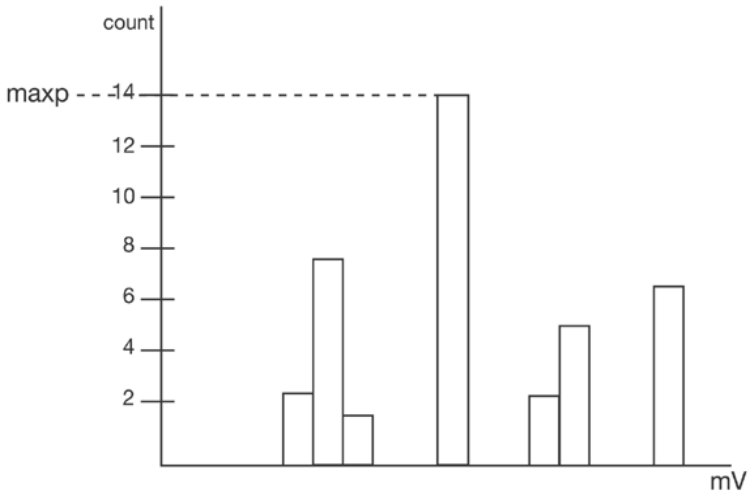
fwxx	Full Width at xx% Maximum
Definition:	Determines the width of the largest area peak, measured between bins on either side of the highest bin in the peak that have a population of xx% of the highest's population. If several peaks have an area equal to the maximum population, the leftmost peak is used in the computation.
Description:	First, the highest population peak is identified and the height of its highest bin (population) determined (see the pks description). Next, the bin populations to the right and left are found until a bin on each side is found to have a population of less than xx% of that of the highest bin. A line is calculated on each side, from the center point of the first bin below the 50% population to that of the adjacent bin, towards the highest bin. The intersection points of these lines with the xx% height value is then determined. The length of a line connecting the intersection points is the value for fwxx.
Example:	fwxx with threshold set to 35%: 

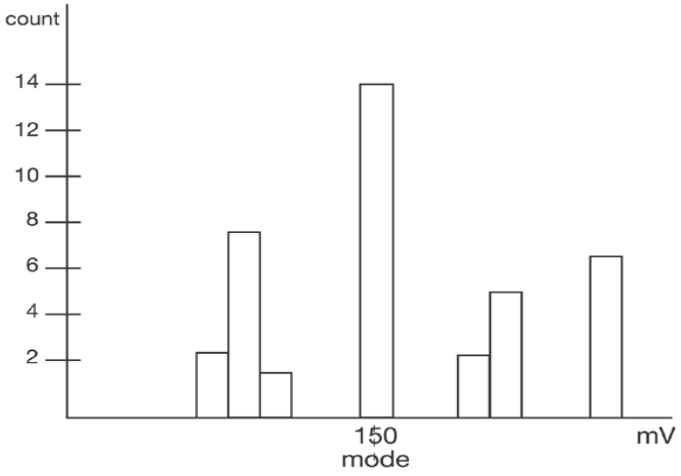
hist ampl	Histogram Amplitude
Definition:	The difference in value of the two most populated peaks in a histogram. This parameter is useful for waveforms with two primary parameter values, such as TTL voltages, where hampl would indicate the difference between the binary `1' and `0' voltage values.
Description:	The values at the center (line dividing the population of peak in half) of the two highest peaks are determined (see pks parameter description:). The value of the leftmost of the two peaks is the histogram base (see hbase). While that of the rightmost is the histogram top (see htop). The parameter is then calculated as:
	hampl = htop - hbase
Example:	 In this histogram, hampl is 152 mV - 150 mV = 2 mV.

hbase	Histogram Base
Definition:	The value of the leftmost of the two most populated peaks in a histogram. This parameter is primarily useful for waveforms with two primary parameter values such as TTL voltages where hbase would indicate the binary `0' voltage value.
Description:	The two highest histogram peaks are determined. If several peaks are of equal height the leftmost peak among these is used (see pks). Then the leftmost of the two identified peaks is selected. This peak's center value (the line that divides the population of the peak in half) is the hbase.
Example:	

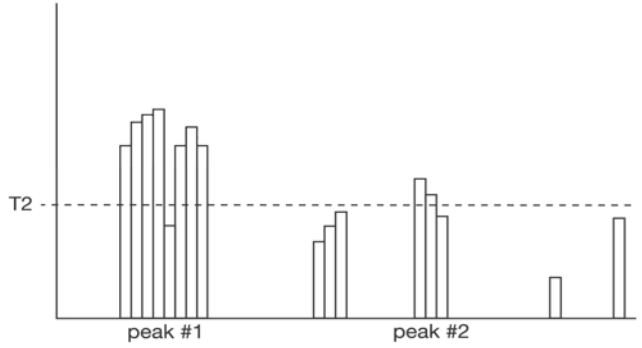
hist rms	Histogram Root Mean Square
Definition:	The rms value of the values in a histogram.
Description:	The center value of each populated bin is squared and multiplied by the population (height) of the bin. All results are summed and the total is divided by the population of all the bins. The square root of the result is returned as hrms.
Example:	Using the histogram shown here, the value for hrms is: $hrms = \sqrt{(3.5^2 * 2 + 2.5^2 * 4)/6} = 2.87$ 

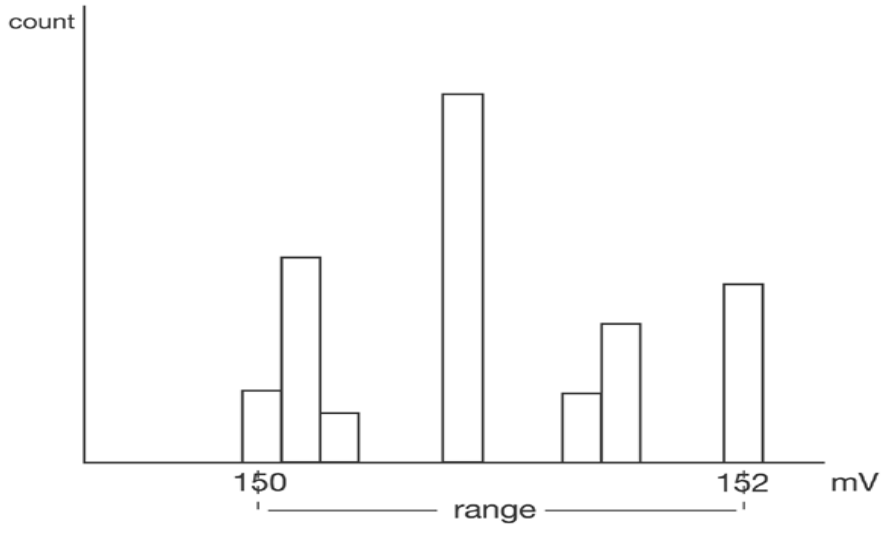
hist top	Histogram Top
Definition:	The value of the rightmost of the two most populated peaks in a histogram. This parameter is useful for waveforms with two primary parameter values, such as TTL voltages, where htop would indicate the binary `1' voltage value.
Description:	The two highest histogram peaks are determined. The rightmost of the two identified peaks is then selected. The center of that peak is htop (center is the horizontal point where the population to the left is equal to the area to the right).
Example:	

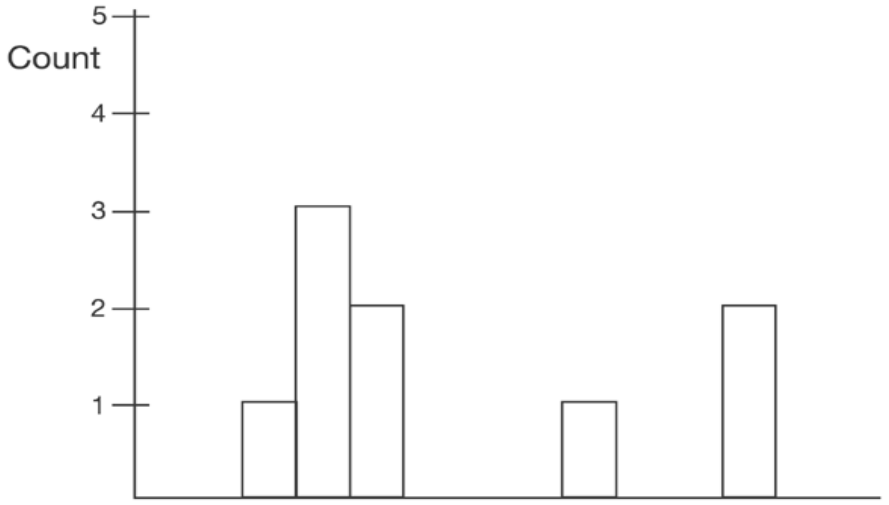
maxp	Maximum Population
Definition:	The count (vertical value) of the highest population bin in a histogram.
Description:	Each bin between the parameter cursors is examined for its count. The highest count is returned as maxp.
Example:	On the following histogram, the maxp is 14. 

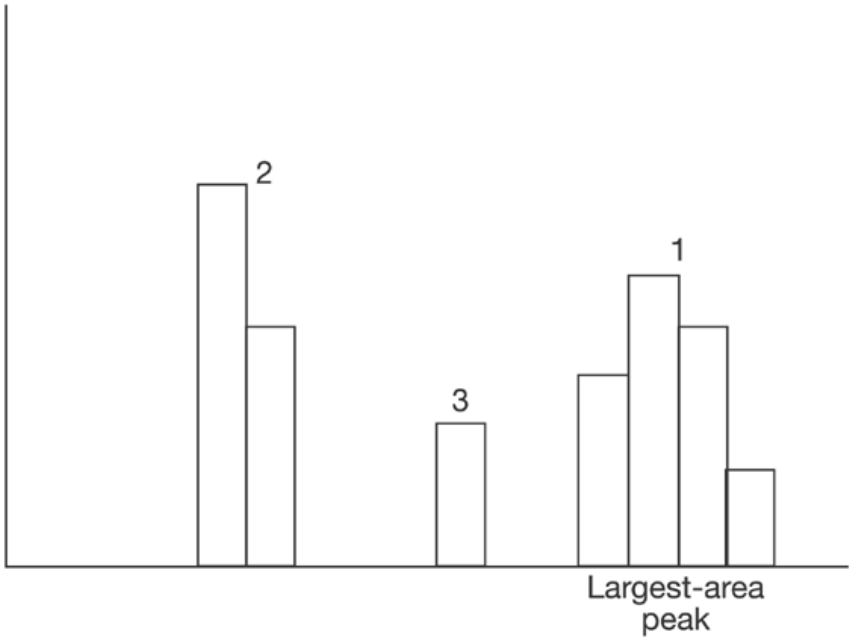
mode	Mode
Definition:	The value of the highest population bin in a histogram.
Description:	Each bin between the parameter cursors is examined for its population count. The leftmost bin with the highest count found is selected. Its center value is returned as mode.
Example:	Here, the mode is 150 mV. 

pctl	Percentile
Definition:	Computes the horizontal data value that separates the data in a histogram such that the population on the left is a specified percentage `xx' of the total population. When the threshold is set to 50%, pctl is the same as hmedian.
Description:	The total population of the histogram is determined. Scanning from left to right, the population of each bin is summed until a bin that causes the sum to equal or exceed `xx'% of the population value is encountered. A ratio of the number of counts needed for `xx'% population/total bin population is then determined for the bin. The horizontal value of the bin at that ratio point of its range is found, and returned as pctl.
Example:	The total population of a histogram is 100. The histogram range is divided into 20 bins and `xx' is set to 25%. The population sum at the sixth bin from the left is 22. The population of the seventh is 9 and its sub-range is 6.1 to 6.4 V. The ratio of counts needed for 25% population to total bin population is:
	3 counts needed / 9 counts = 1/3.
	The value for pctl is:
	6.1 volts + .33 * (6.4 - 6.1) volts = 6.2 volts.

pks	Peaks
Definition:	The number of peaks in a histogram.
Description:	The instrument analyzes histogram data to identify peaks from background noise and histogram binning artifacts such as small gaps.
	Peak identification is a 3-step process:
	The mean height of the histogram is calculated for all populated bins. A threshold (T1) is calculated from this mean, where: T1= mean + 2 sqrt (mean). A second threshold is determined based on all populated bins under T1 in height, where: T2 = mean + 2 * sigma, and where sigma is the standard deviation of all populated bins under T1. Once T2 is defined, the histogram distribution is scanned from left to right. Any bin that crosses above T2 signifies the existence of a peak. Scanning continues to the right until one bin or more crosses below T2. However, if the bins cross below T2 for less than a hundredth of the histogram range, they are ignored, and scanning continues in search of peaks that cross under T2 for more than a hundredth of the histogram range. Scanning goes on over the remainder of the range to identify additional peaks. Additional peaks within a fiftieth of the range of the populated part of a bin from a previous peak are ignored.
	Note: If the number of bins is set too high, a histogram may have many small gaps. This increases sigma and, thereby, T2. In extreme cases, it can prevent determination of a peak, even if one appears to be present to the eye.
Example:	Here the two peaks have been identified. The peak with the highest population is peak #1. 

range	Range
Definition:	Computes the difference between the value of the rightmost and that of the leftmost populated bin.
Description:	The rightmost and leftmost populated bins are identified. The difference in value between the two is returned as the range.
Example:	In this example, the range is 2 mV. 

totp	Total Population
Definition:	Calculates the total population of a histogram between the parameter cursors.
Description:	The count for all populated bins between the parameter cursors is summed.
Example:	The total population of this histogram is 9. 

xapk	X Coordinate of xxth Peak
Definition:	Returns the value of the xxth peak that is the largest by area in a histogram.
Description:	First the peaks in a histogram are determined and ranked in order of total area (for a discussion on how peaks are identified see the description for the pks parameter). The center of the nth ranked peak (the point where the area to the left is equal to the area to the right), where n is selected by you, is then returned as xapk.
Example:	The rightmost peak is the largest, and is thus ranked first in area (1). The leftmost peak, although higher, is ranked second in area (2). The lowest peak is also the smallest in area (3).
	

Histogram Theory of Operation

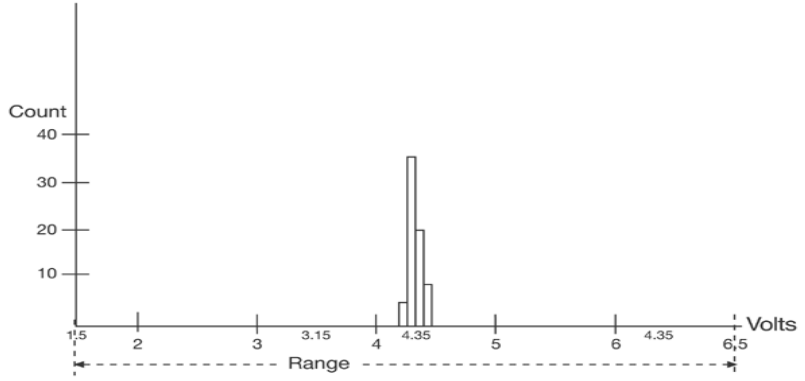
An understanding of statistical variations in parameter values is needed for many waveform parameter measurements. Knowledge of the average, minimum, maximum, and standard deviation of the parameter are often enough. However, in many cases you may need a more detailed understanding of the distribution of a parameter's values.

Histograms allow you to see how a parameter's values are distributed over many measurements. This is done by dividing a range of parameter values into sub-ranges called bins. A count of the number of parameter values (events) falling within ranges of the bin itself is maintained for each bin.

While such a value range can be infinite, for practical purposes it only needs to be defined as large enough to include any realistically possible parameter value. For example, measuring TTL high-voltage values in a range of ±50 V is unnecessarily large, while one of 4 V ±2.5 V is more reasonable. It is the 5 V range that is subdivided into bins. If the number of bins used were 50, each would have a range of 5 V/50 bins or 0.1 V/bin. Events falling into the first bin would be between 1.5 V and 1.6 V, while the next bin would capture all events between 1.6 V and 1.7 V, and so on.

After a process of several thousand events, the bar graph of the count for each bin (its histogram) provides a good understanding of the distribution of values. Histograms generally use the 'x' axis to show a bin's sub-range value and the 'Y' axis for the count of parameter values within each bin. The leftmost bin with a non-zero count shows the lowest parameter value measurements. The highest vertical bin shows the greatest number of events falling within its sub-range.

The number of events in a bin, peak or a histogram is referred to as its population. The following figure shows a histogram's highest population bin as the one with a sub-range of 4.3 to 4.4 V (which is expected from a TTL signal).



The lowest-value bin with events is that with a sub-range of 3.0 to 3.1 V. As TTL high voltages need to be greater than 2.5 V, the lowest bin is within the allowable tolerance. However, because of its proximity to this tolerance and the degree of the bin's separation from all other values, additional investigation may be required.

DSO Process

This instrument can be used to generate histograms of the parameter values of input waveforms.

The follwing items must be defined before doing so:

- The parameter to be histogrammed
- The trace on which the histogram is to be displayed
- The maximum number of parameter measurement values to be used in creating the histogram
- The measurement range of the histogram
- The number of bins to be used

Some of these items are pre-defined, but can be changed. Once defined, the oscilloscope is ready to make the histogram. The sequence for acquiring histogram data is as follows:

1. Trigger
2. Waveform acquisition
3. Parameter calculations
4. Histogram update
5. Trigger re-arm

If you set the timebase for non-segmented mode, a single acquisition occurs prior to parameter calculations. However, in Sequence mode an acquisition for each segment occurs prior to parameter calculations. If the source of histogram data is a memory, saving new data to memory effectively acts as a trigger and acquisition. This only takes place one time per second because updating the screen can take lots of processing. This also minimizes trigger dead time. Under remote control operation, the display can be turned off to maximize measurement speed.

Parameter Buffer

The oscilloscope maintains a circular parameter buffer of the last 20,000 measurements made, including values falling outside the set histogram range. If the maximum number of events to be used for the histogram is a number `N' less than 20,000, the histogram is continuously updated with the last `N' events as new acquisitions occur. If the maximum number is greater than 20,000, the histogram updates until the number of events is equal to `N.' Then, if the number of bins or the histogram range is modified, the scope uses the parameter buffer values to redraw the histogram with either the last `N' or 20,000 values acquired (whichever is the lesser value). This allows the parameter buffer to redisplay histograms using an acquired set of values and settings. This produces a distribution shape with the most useful information.
In cases where the optimal range is not apparent, the scope has a powerful range finding function. If required, the values in the parameter buffer are examined to calculate an optimal range and redisplay the histogram. The instrument also provides a running count of the parameter value amounts falling within, below, or above the range. Values falling below or above the range can then be recalculated by the range finder to include these parameter values (as long as they are still within the buffer).

Capture of Parameter Events

The number of events captured per waveform acquisition or display sweep depends on the parameter type. Acquisitions are initiated by the occurrence of a trigger event. Sweeps are equivalent to the waveform captured and displayed on an input channel (1, 2, or 3 or 4). For non-segmented waveforms an acquisition is identical to a sweep. Whereas for segmented waveforms an acquisition occurs for each segment and a sweep is equivalent to acquisitions for all segments. Only the section of a waveform between the parameter cursors is used in the calculation of parameter values and corresponding histogram events.
The following table provides a summary of the number of histogram events captured per acquisition or sweep for each parameter, and for a waveform section between the parameter cursors.

Parameters	Number of Events Captured
duty, freq, period, width, time@lev, f@level, f80-20%, fall, r@level, r20-80%, rise	All events in the acquisition
ampl, area, base, cmean, cmedian, crms, csdev, cycles, delay, maximum, mean, minimum, nbph, nbpw, over+, over-, pkpk, npts, rms, sdev, dly	One event per acquisition

Histogram Parameters (XMAP and JTA2 Options)

Once a histogram is defined and generated, measurements can be performed on the histogram itself. Typical of these are the histogram's

- average value, standard deviation
- most common value (parameter value of highest count bin)
- leftmost bin position (representing the lowest measured waveform parameter value)
- rightmost bin (representing the highest measured waveform parameter value)

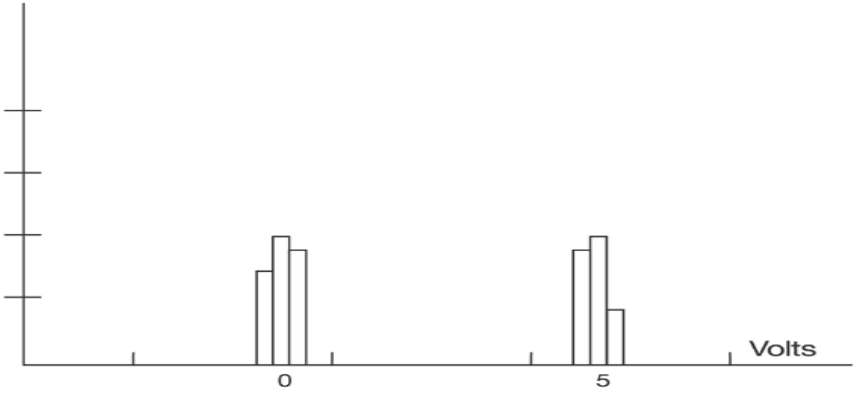
Histogram parameters are provided to enable these measurements. Available through selecting "Statistics" from the "Category" menu, they are calculated for the selected section between the parameter cursors:

- **fwhm** full width (of largest peak) at half the maximum bin
- **fwxx** full width (of largest peak) at xx% the maximum bin
- **hist ampl** histogram amplitude between two largest peaks
- **hist base** histogram base or leftmost of two largest peaks
- **hist max** value of the highest (right-most) populated bin in a histogram
- **hist mean** average or mean value of data in the histogram
- **hist median** value of the x-axis of a histogram that divides the population into two equal halves
- **hist min** value of the lowest (left-most) populated bin in a histogram
- **hist rms** rms value of data in histogram
- **hist sdev** standard deviation of values in a histogram

- **hist top** histogram top or rightmost of two largest peaks
- **max populate** population of most populated bin in histogram
- **mode** data value of most populated bin in histogram
- **percentile** data value in histogram for which specified `x'% of population is smaller
- **peaks** number of peaks in histogram
- **pop @ x** population of bin for specified horizontal coordinate
- **range** difference between highest and lowest data values
- **total pop** total population in histogram
- **x at peak** x-axis position of specified largest peak

Histogram Peaks

Because the shape of histogram distributions is particularly interesting, additional parameter measurements are available for analyzing these distributions. They are generally centered around one of several peak value bins, known, with its associated bins, as a histogram peak.
Example: In the following figure, a histogram of the voltage value of a five-volt amplitude square wave is centered on two peak value bins: 0 V and 5 V. The adjacent bins signify variation due to noise. The graph of the centered bins shows both as peaks.



Determining such peaks is very useful because they indicate dominant values of a signal. However, signal noise and the use of a high number of bins relative to the number of parameter values acquired, can give a jagged and spiky histogram, making meaningful peaks hard to distinguish. The scope analyzes histogram data to identify peaks from background noise and histogram definition artifacts such as small gaps, which are due to very narrow bins.

Binning and Measurement Accuracy

Histogram bins represent a sub-range of waveform parameter values, or events. The events represented by a bin may have a value anywhere within its sub-range. However, parameter measurements of the histogram itself, such as average, assume that all events in a bin have a single value. The scope uses the center value of each bin's sub-range in all its calculations. The greater the number of bins used to subdivide a histogram's range, the less the potential deviation between actual event values and those values assumed in histogram parameter calculations.

Nevertheless, using more bins may require that you perform a greater number of waveform parameter measurements, in order to populate the bins sufficiently for the identification of a characteristic histogram distribution.
In addition, very fine grained binning will result in gaps between populated bins that may make it difficult to determine peaks.
The oscilloscope's 20,000-parameter buffer is very effective for determining the optimal number of bins to be used. An optimal bin number is one where the change in parameter values is insignificant, and the histogram distribution does not have a jagged appearance. With this buffer, a histogram can be dynamically redisplayed as the number of bins is modified by the user. In addition, depending on the number of bins selected, the change in waveform parameter values can be seen.

WAVEFORM MEASUREMENTS

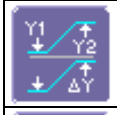
Measuring with Cursors

Cursors are important tools that aid you in measuring signal values. Cursors are markers — lines, cross-hairs, or arrows — that you can move around the grid or the waveform itself. Use cursors to make fast, accurate measurements and to eliminate guesswork. There are two basic types:

- **Horiz(ontal)** (generally Time or Frequency) cursors are markers that you move horizontally along the waveform. Place them at a desired location along the time axis to read the signal's amplitude at the selected time.
- **Vert(ical)** (Voltage) cursors are lines that you move vertically on the grid to measure the amplitude of a signal.

Cursor Measurement Icons

The **Readout** icons depict what is being measured for each measurement mode.

	Each cursor locates a point on the waveform. The cursor values can be read in the descriptor label for the trace. Use the **Position** data entry fields at the right side of the dialog to place the cursors precisely.
	This is the difference in Y values. The value can be read in the descriptor label for the trace.
	Displays absolute and delta cursors together.
	This gives the slope between cursors.

If there are nontime-domain waveforms displayed, there will also be a menu offering choices of x-axis units: **s** or **Hz**, for example.

Cursors Setup

Quick Display

At any time, you can change the display of cursor types (or turn them off) without invoking the "Cursors Setup" dialog as follows:

1. In the menu bar, touch **Cursors**, then **Off**, **Abs Horizontal**, **Rel Horizontal**, **Abs Vertical**, or **Rel Vertical**.
2. The cursors displayed will assume the positions previously set up. If you want to change their position or measurement mode, in the menu bar touch **Cursors**, then **Cursors Setup** in the drop-down menu.

Full Setup

1. In the menu bar, touch **Cursors**, then **Cursors Setup**. The "Standard Cursors" dialog opens.
2. In the dialog area, touch the **Cursors On** check box to display them.
3. Touch one of the **Horizontal** or **Vertical** mode buttons: **Relative** or **Absolute**.
4. If you chose a Relative mode, also touch a readout parameter button: Y position, delta Y, or slope.
5. If you chose a Relative mode, touch inside the **Position 1** and **Position 2** data entry fields and type in a value for each cursor. You can also use the Cursors knobs on the front panel to place the cursors. If you chose an Absolute mode, do the same for your single cursor.
6. If you chose a Relative mode and you would like both cursors to move in unison as you adjust the position, touch the **Track** check box to enable tracking.

Overview of Parameters

Parameters are measurement tools that determine a wide range of waveform properties. Use them to automatically calculate many attributes of your waveform, like rise-time, rms voltage, and peak-to-peak voltage, for example.
There are parameter modes for the amplitude and time domains, custom parameter groups, and parameters for pass and fail testing. You can make common measurements on one or more waveforms.

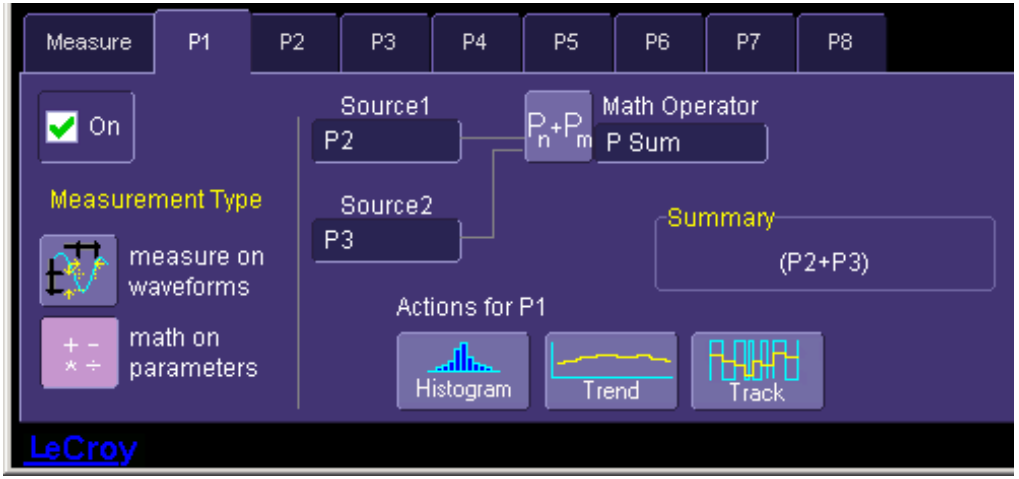
Turning On Parameters

1. Touch **Measure** in the menu bar, then **Measure Setup...** in the drop-down menu.
2. Touch inside the **On** checkbox for each parameter you want to display.

Quick Access to Parameter Setup Dialogs

You can quickly gain access to a parameter setup dialog by touching the parameter list box below the grid. For example, touching within **P1** below the grid displays the setup dialog for P1:





Touching the row titles displays the top Measure dialog.



Status Symbols

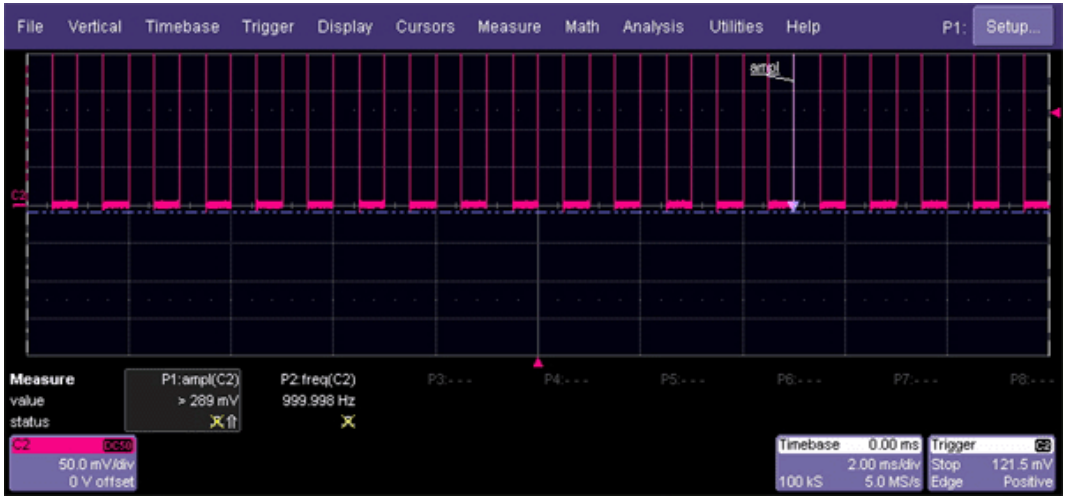
Below each parameter appears a symbol that indicates the status of the parameter, as follows:

⚠	A warning symbol indicates that there is something wrong with the signal or the setup. Touch the parameter list box and read the explanation in the message line at the bottom of the screen.
✓	A green check mark means that the scope is returning a valid value.
⌐	A crossed-out pulse means that the scope is unable to determine top and base; however, the measurement could still be valid.
⇓	A downward pointing arrow indicates an underflow condition.
⇑	An upward pointing arrow indicates an overflow condition.
⇕	An upward-and-downward pointing arrow indicates an underflow and overflow condition.

Using X-Stream Browser to Obtain Status Information

Example:

Here is a case of an overflow condition, in which the amplitude of the waveform cannot be determined:

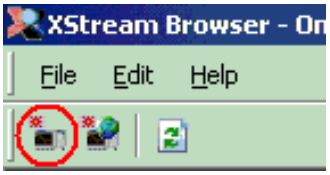


1. Minimize the scope display by selecting **File ➔ Minimize**.

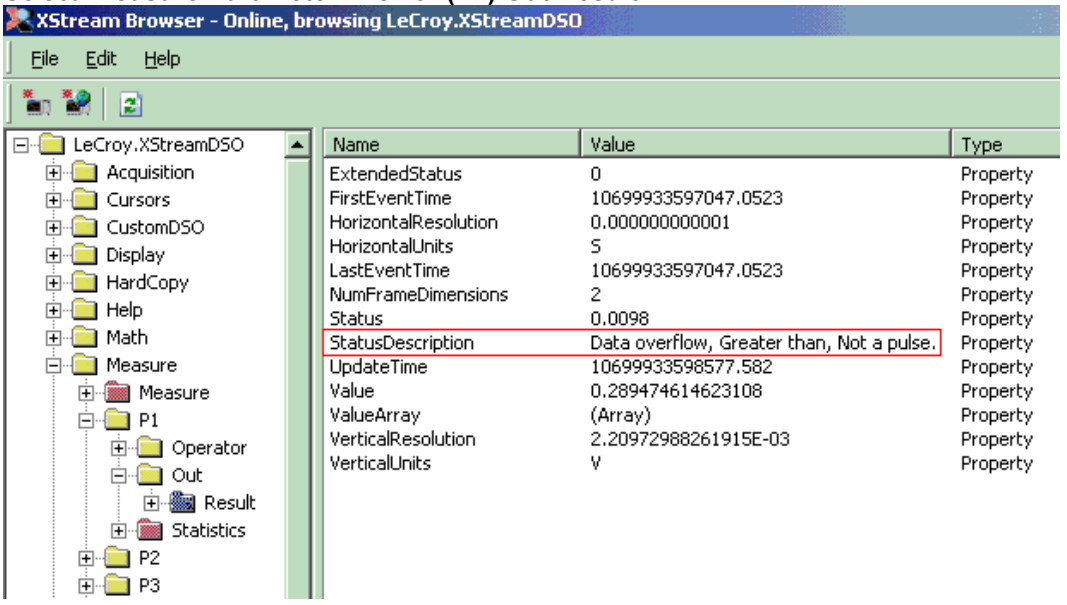
2. Touch the **X-Stream Browser** desktop icon to open the browser.



3. Touch the left scope icon ("Connect to a local X-Stream DSO device") in the X-Stream Browser toolbar:



4. Select **Measure Parameter in error (P1) Out Result**



5. Read the status information in line **StatusDescription**.

Statistics

Touching the **Statistics On** checkbox in the "Measure" dialog displays statistics for standard vertical or horizontal parameters, or for custom parameters. The statistics shown are as follows:

value (last)
mean
min.
max.
sdev
num

The values displayed in the **num** row are the number of measurements computed. For any parameter that computes on an entire waveform (like edge@level, mean, minimum, maximum, etc.) the value displayed represents the number of sweeps.
For any parameter that computes on every event, the value displayed is equal to the number of events per acquired waveform. If *x* waveforms were acquired, the value represents *x* times the number of cycles per waveform. Also, the "value" is equal to the measurement of the last cycle on the last acquisition.

Applying a Measure Mode

1. In the menu bar, touch **Measure**, then **Measure Setup**.
2. Choose a **Measure Mode** from the dialog. The parameters are displayed below the grid.

Measure Modes

The selections for Measure Mode allow you to quickly apply parameters for standard vertical and standard horizontal setups, and custom setups.

Standard Vertical Parameters

These are the default Standard Vertical Parameters:

Vertical
mean
sdev
max.
min.
ampl
pkpk
top
base

Standard Horizontal Parameters
These are the default Standard Horizontal Parameters:

Horizontal
freq
period
width
rise
fall
delay
duty
npoints

My Measure

You can choose to customize up to eight parameters by touching **My Measure**.

Parameter Math (XMath or XMAP option required)

The instrument gives you the ability to perform arithmetic operations (addition, subtraction, multiplication, division) on the results of two parameter measurements. Alternatively, you can apply math to a single parameter (for example, invert). By customizing parameters in this way, you can effectively extend the range of parameter measurements based on your particular needs.

Logarithmic Parameters

The parameter math feature prevents multiplication and division of parameters that return logarithmic values. These parameters are as follows:

- auto-correlation signal-to-noise ratio (ACSN)
- narrow-band power (NBPW)
- media signal-to-noise ratio (MSNR)
- residual signal-to-noise ratio (RSNR)
- top-to-base ratio when the units are in dB (TBR)

Excluded Parameters

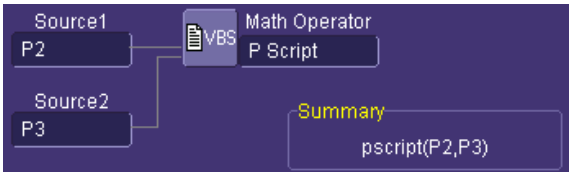
Parameters that are already the result of parameter math operations are excluded. If they are included in a remote control setup command, an error message is generated and the setup canceled.

- Excluded parameters are as follows:
- delta clock-to-data near (DC2D)
- delta clock-to-data next (DC2DPOS)
- delta clock-to-data previous (DC2DNEG)
- delta delay (DDLY)
- delta time at level (DTLEV)
- phase (PHASE)
- resolution (RES)

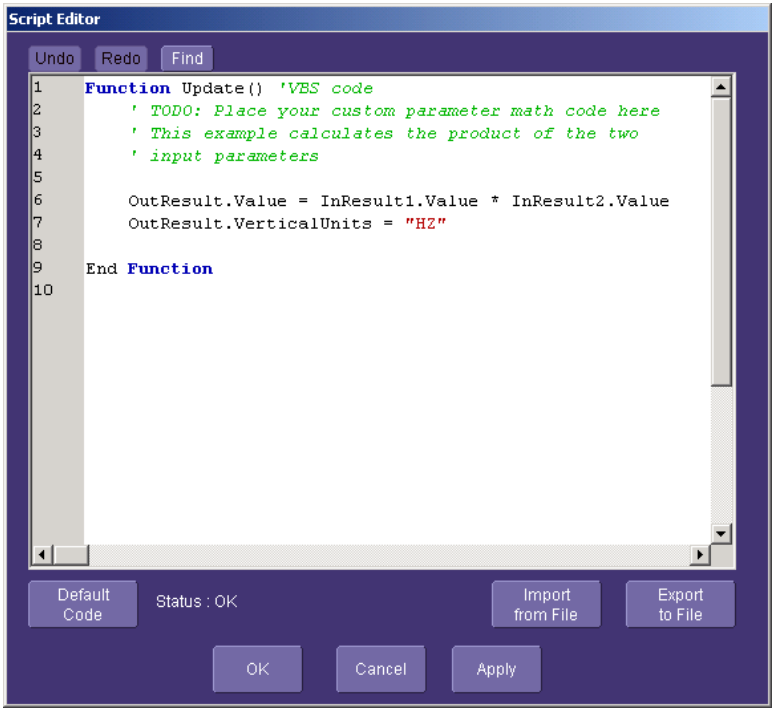
- mTnTmT shift (BEES)
- mTnTmT shift sigma (BEESS)
- mTnTmT shift sigma – list (BEESS)

Parameter Script Parameter Math

In addition to the arithmetic operations, the Parameter Math feature allows you to use VBScript or JavaScript to write your own script for one or two measurements and produce a result that suits your needs.



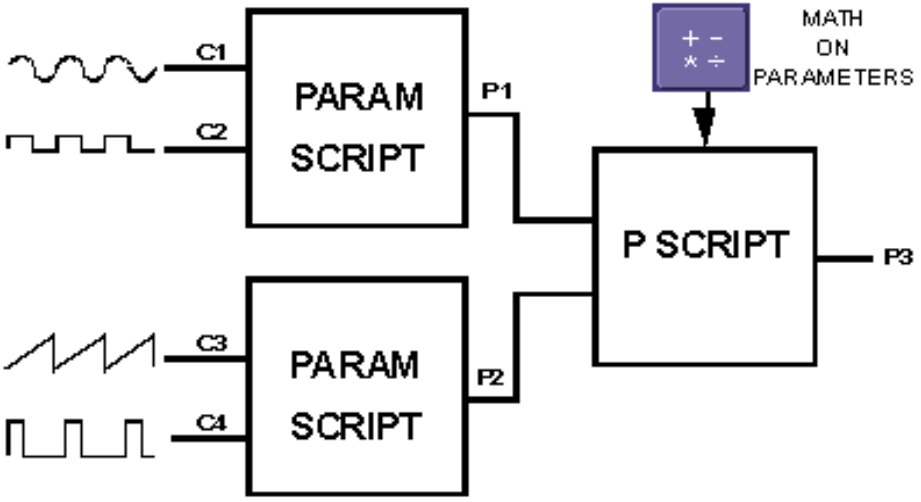
Code entry is done in the Script Editor window directly on the instrument. You can also import an existing script.



Param Script vs. P Script

Param Script is a VBScript or JavaScript that operates on one or two *waveforms* and outputs a parameter measurement, as shown in the figure below. P Script, on the other hand, is another VBScript or JavaScript that takes as input one or two *parameters* and performs a math operation on them to produce another parameter output.

The inputs to Param Script can also be math (F1-Fx) or memory (M1-Mx) traces. The inputs to P Script can be the results of any parameter measurement, not necessarily Param Script.



Parameter Math Setup

1. Touch **Measure** in the menu bar, then **Measure Setup...** in the drop-down menu.
2. Touch the **My Measure** button in the "Measure" dialog.
3. Touch the **Px** tab for the parameter to which you want to apply parameter math.
4. In the "Px" dialog, touch the **math on parameters** button . The **Source** field will expand to two fields.
5. Touch inside the **Source1** and **Source2** fields and select the parameters you want to apply math to (P1 to Px). If you are applying math to a single parameter (for example, invert), just touch inside the **Source1** field and select a parameter (P1 to Px).
6. Touch inside the **Math Operator** field and select a math operation from the **Select Measurement** menu. If you select an operation that requires two input parameters, the **Source** field will expand to two fields.

Parameter Script Math Setup

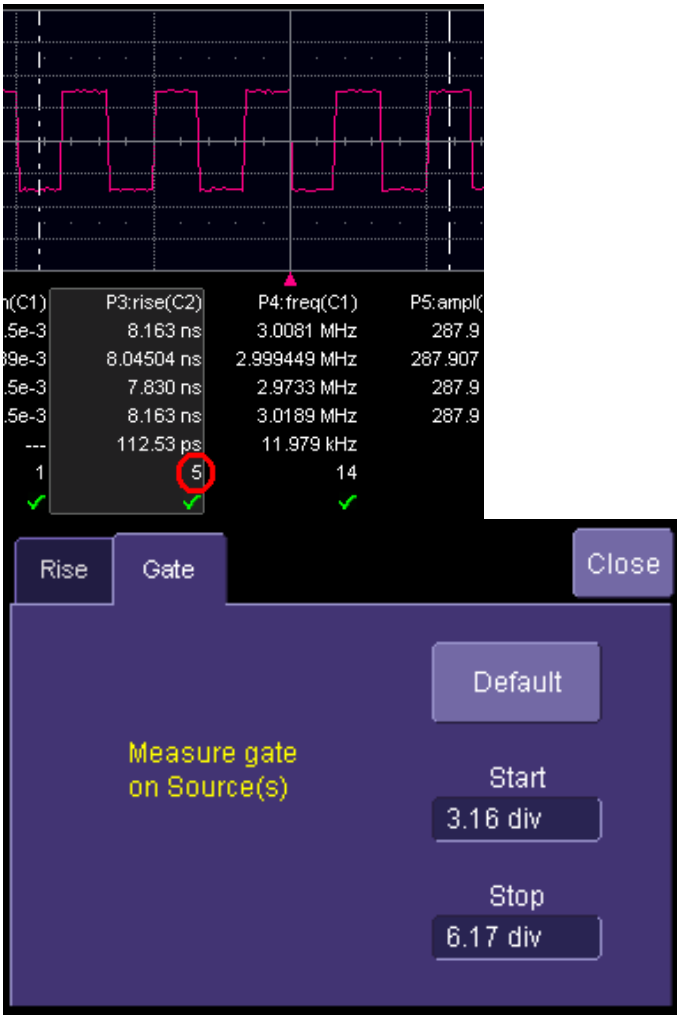
1. Touch **Measure** in the menu bar, then **Measure Setup...** in the drop-down menu.
2. Touch the **My Measure** button in the "Measure" dialog.
3. Touch the **Px** tab for the parameter to which you want to apply parameter math.

4. In the "Px" dialog, touch the **math on parameters** button . The **Source** field will expand to two fields.
5. Touch inside the **Source1** and **Source2** fields and select the parameters you want to apply math to (P1 to Px). If you are applying math to a single parameter (for example, invert), just touch inside the **Source1** field and select a parameter (P1 to Px).
6. Touch inside the **Math Operator** field and select  **P Script** from the **Select Measurement** menu.
7. In the "Script Math" dialog, touch inside the Script Language field and select either **VBScript** or **JScript** from the pop-up menu.
8. Touch the **Edit Code** button; the Script Editor window opens. You can enter code in this window or call up an existing script from a file storage location. If you create your script in this window, you can then export it and save it to file.

Measure Gate

Using Measure Gate, you can narrow the span of the waveform on which to perform parameter measurements, allowing you to focus on the area of greatest interest. You have the option of dragging the gate posts horizontally along the waveform, or specifying a position down to hundredths of a division. The default starting positions of the gate posts are 0 div and 10 div, which coincide with the left and right ends of the grid. The gate, therefore, initially encloses the entire waveform.

Note: If you have Grid On Top enabled, you will not see the gate posts in their default position at each end of the grid. But if you touch either end of the grid, a drag cursor  will appear, indicating that you have control of the post and can now drag it.

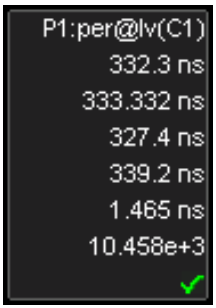


In this example, you can see that the Measure Gate includes only five rising edges. Therefore, parameter calculations for rise time are performed only on the five pulses bounded by the gate posts. The position of the gate posts is shown in the **Start** and **Stop** fields in the accompanying dialog.

Measure Gate Setup

1. In the menu bar, touch **Measure Setup...**
2. Touch the **Px** tab for the parameter you want to gate. A mini-dialog to the right of the main setup dialog opens.

Note: If you already have the parameter of interest set up, you can simply touch the parameter directly below the grid.



Example Parameter Readout

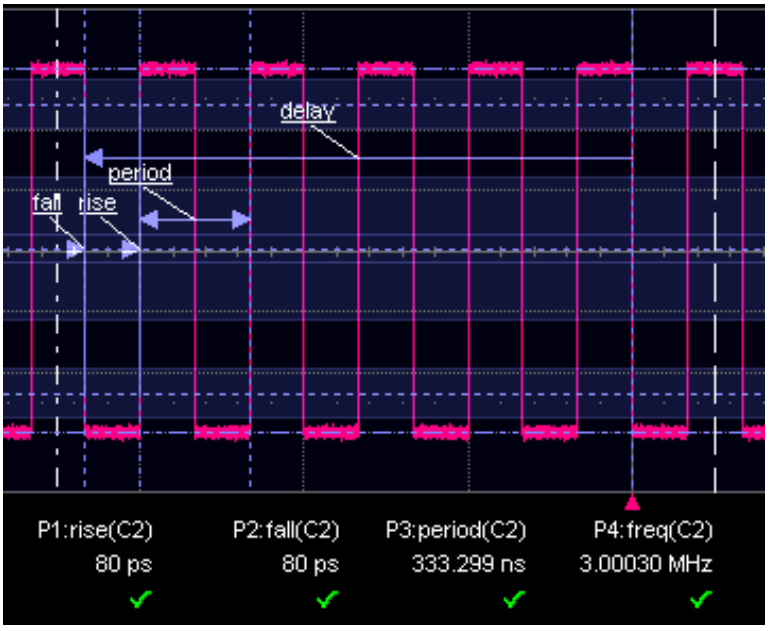
3. Touch inside the **Start** data entry field and enter a value, using the pop-up numeric keypad. Or, you can simply touch the leftmost grid line and drag the gate post to the right.
4. Touch inside the **Stop** data entry field and enter a value, using the pop-up numeric keypad. Or, you can simply touch the rightmost grid line and drag the gate post to the left.

Help Markers

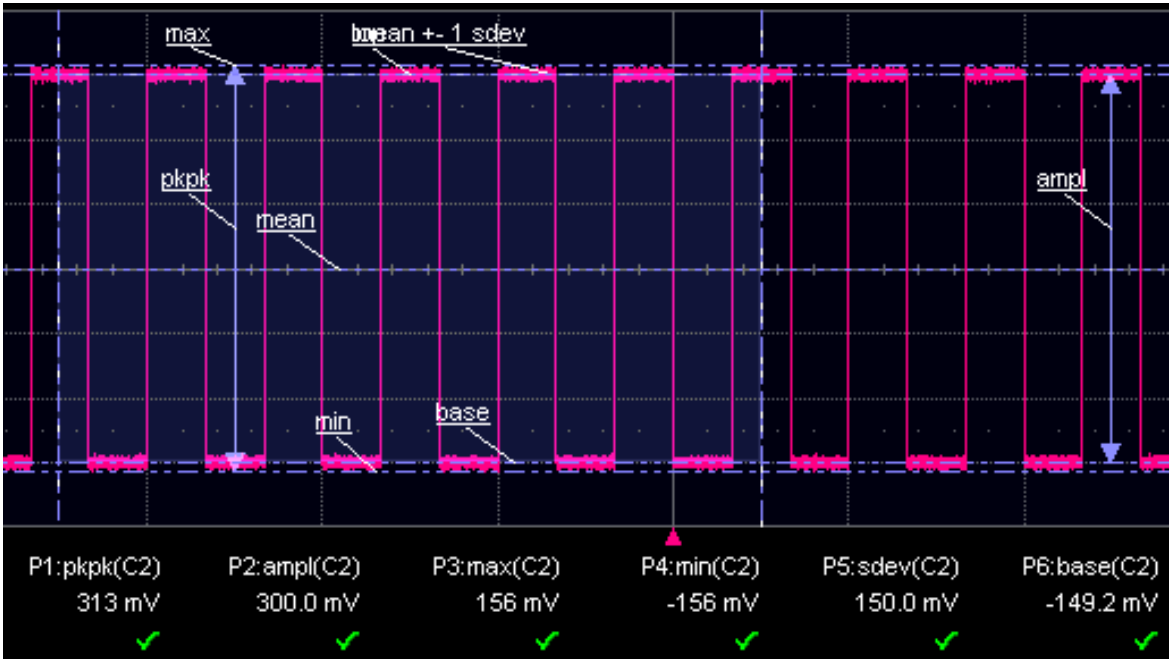
Help Markers clarify parameter measurements by displaying movable cursors and a visual representation of what is being measured. For the "at level" parameters, Help Markers make it easier to see where your waveform intersects the chosen level. This feature also displays the hysteresis band that you have set about that level.
You also have the option, by means of an **Always On** checkbox, to leave the Help Markers displayed after you have closed the Help Markers setup dialog.

You have a choice of **Simple** or **Detailed** views of the markers:

- The **Simple** selection produces cursors and Measure Gate gate posts. The gate posts are independently placeable for each parameter.
- The **Detailed** selection produces cursors, Measure Gate gate posts, a label identifying the parameter being measured, and a level indicator and hysteresis band for "at level" parameters (not part of Standard Horizontal or Standard Vertical parameters).



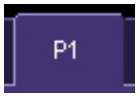
Standard Horizontal Parameter Help Markers



Standard Vertical Parameter Help Markers

Help Marker Setup

1. In the menu bar, touch **Measure Setup...**
2. Select a Measure Mode: **Std Vertical**, **Std Horizontal**, or **My Measure**.
3. Touch the **Show All** button 

to display Help Markers for every parameter being measured on the displayed waveform (C2 in the examples above).
4. Touch a **P1** to **Px** tab 

for any parameter listed; it doesn't matter which. Touch inside the **Help Markers** field and select **Simple** The **Simple** selection produces cursors and Measure Gate gate posts. The gate posts are independently placeable for each parameter. or **Detailed** The **Detailed** selection produces cursors, Measure Gate gate posts, a label identifying the parameter being measured, and a level indicator and hysteresis band for "at level" parameters.

Note: The choice of Simple or Detailed is applied to all parameters at the same time. That is, if you choose Simple markers for one parameter, all parameters will be displayed in this mode.

5. Touch the **Always On** checkbox if you want to continuously display Help Markers for this parameter.

Turning Off Help Markers

1. Touch the **Clear All** button 

to turn off Help Markers for all parameters.
2. To turn off Help Markers for individual parameters, touch the **Px** tab for the parameter in question. Then uncheck the **Always On** checkbox. When you close this dialog, the Help Markers for this parameter will no longer be displayed.

Customizing a Parameter

From the Measure Dialog

1. Touch the **My Measure** button in the "Measure" dialog. The dialog presents you with a panel of eight preset parameters.
2. For each parameter, touch the **On** check box to enable the parameter listed.

3. If you want to change the parameter listed, or a measurement characteristic, touch the parameter button 

(**P1** for example) alongside the check box. A pop-up menu of parameters categorized by type appears. To display parameter icons only, touch the icon button  at the bottom of the menu. To display the icons in list form, along with an explanation of each parameter, touch the list button . Use the Up/Down buttons to scroll through the list of icons.
4. When you make a selection from the parameter icon menu, the setup dialogs for that parameter appear. You can then change the waveform source and other conditions of the parameter.
5. If you are setting up an "@level" parameter, make selections for **Level type** (percent or absolute), **Slope** (positive, negative, both), and **Hysteresis** level.
6. Touch the **Gate** tab, and set the position of the gate posts.

From a Vertical Setup Dialog

1. In the "Cx Vertical Adjust" dialog, touch the Measure button 

.
2. Select a parameter from the pop-up menu. (The **Actions for trace** source defaults to the channel or trace whose dialog is open. If a parameter, it goes into the next "available" parameter, or the last one if all are used.)
3. Select another parameter or touch **Close**.

From a Math Setup Dialog

1. In the "Fx" dialog, touch the Measure button 

.
2. Select a parameter from the pop-up menu. (The **Actions for trace** source defaults to the channel or trace whose dialog is open. If a parameter, it goes into the next "available" parameter, or the last one if all are used.)
3. Select another parameter or touch **Close**.

Parameter Calculations

Parameters and How They Work

Determining Top and Base Lines

Proper determination of the top and base reference lines is fundamental for ensuring correct parameter calculations. Analysis begins by computing a histogram of the waveform data over the time interval spanned by the left and right time cursors. For example, the histogram of a waveform transitioning in two states contains two peaks (see Figure 1 as follows).

The analysis attempts to identify the two clusters containing the largest data density. Then, the most probable state (centroids) associated with these two clusters is computed to determine the top and base reference levels: the top line corresponds to the top and the base line to the bottom centroid.

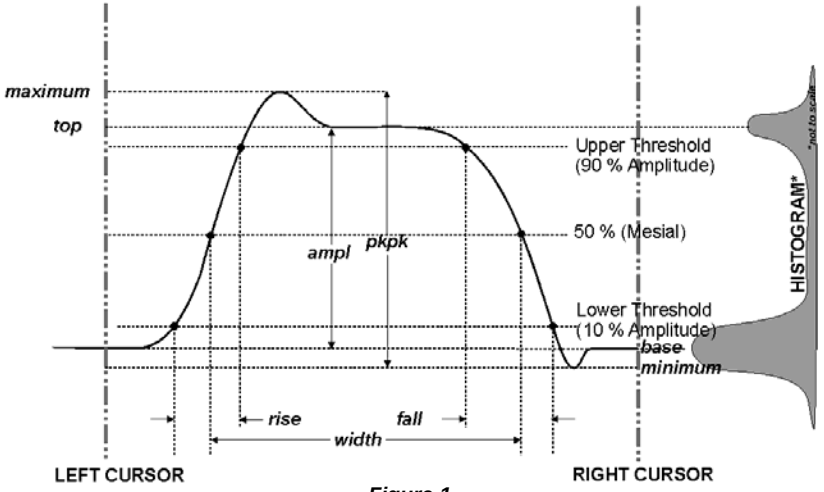


Figure 1

Determining Rise and Fall Times

Once top and base are estimated, calculation of the rise and fall times is easily done (see previous figure). The 90% and 10% threshold levels are automatically determined by the instrument, using the amplitude (ampl) parameter.
Threshold levels for rise or fall time can also be selected using absolute or relative settings (r@level, f@level). If absolute settings are chosen, the rise or fall time is measured as the time interval separating the two crossing points on a rising or falling edge. But when relative settings are chosen, the vertical interval spanned between the base and top lines is subdivided into a percentile scale (base = 0 %, top = 100 %) to determine the vertical position of the crossing points.

The time interval separating the points on the rising or falling edges is then estimated to yield the rise or fall time. These results are averaged over the number of transition edges that occur within the observation window.

Rising Edge Duration	$\frac{1}{Mr}\sum_{i=1}^{Mr}(Tr_i^{90} - Tr_i^{10})$
Falling Edge Duration	$\frac{1}{Mf}\sum_{i=1}^{Mf}(Tf_i^{10} - Tf_i^{90})$
Where *Mr* is the number of leading edges found, *Mf* the number of trailing edges found, Tr_i^x the time when rising edge *i* crosses the x% level, Tf_i^x and the time when falling edge *i* crosses the x% level.	

Determining Time Parameters

Time parameter measurements such as width, period, and delay are carried out with respect to the mesial reference level (see Figure 2 as follows), located halfway (50%) between the top and base reference lines.

Time-parameter estimation depends on the number of cycles included within the observation window. If the number of cycles is not an integer, parameter measurements such as rms or mean are biased. However, only the last value is actually displayed, the mean being available when statistics are enabled. To avoid these bias effects, the instrument uses cyclic parameters, including crms and cmean, which restrict the calculation to an integer number of cycles.

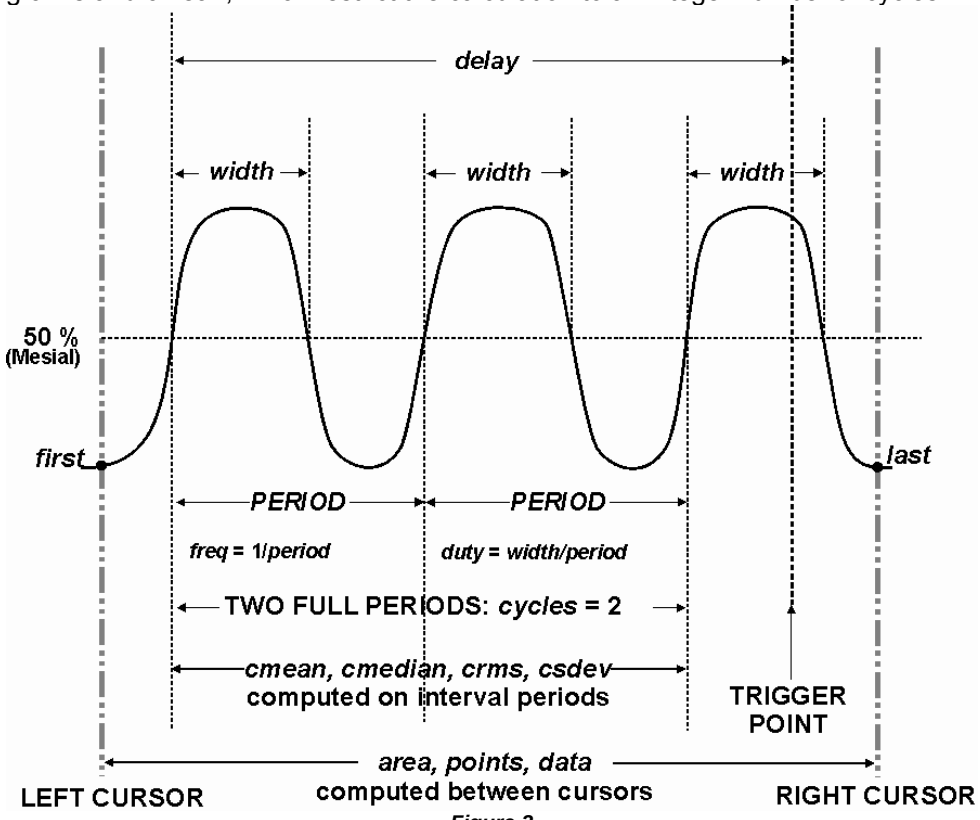


Figure 2

Determining Differential Time Measurements

The instrument enables accurate differential time measurements between two traces: for example, propagation, setup and hold delays (see Figure 3).
Parameters such as Delta c2d± require the transition polarity of the clock and data signals to be specified.

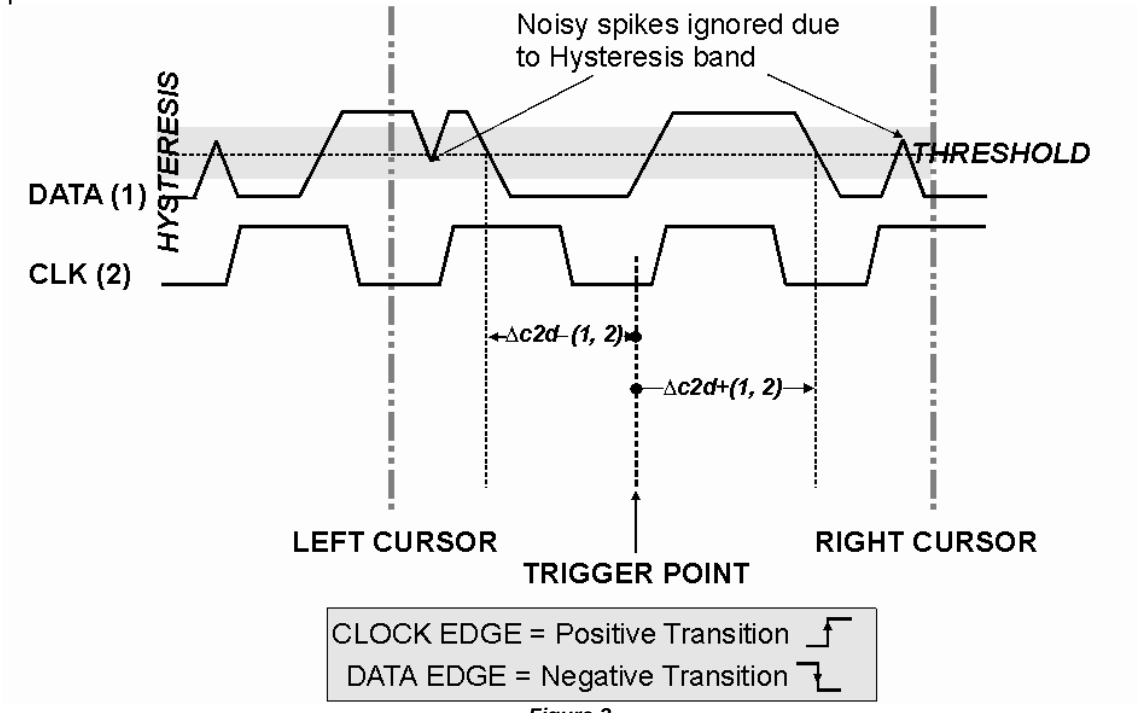


Figure 3

Moreover, a hysteresis range may be specified to ignore any spurious transition that does not exceed the boundaries of the hysteresis interval. In Figure 3, Delta c2d- (1, 2) measures the time interval separating the rising edge of the clock (trigger) from the first negative transition of the data signal. Similarly, Delta c2d+ (1, 2) measures the time interval between the trigger and the next transition of the data signal.

Level and Slope

For several time based measurements, you can choose positive, negative, or both slopes to begin parameter measurements. For two-input parameters, such as Dtime@level, you can specify the slope for each input, as well as the level and type (percent or absolute).

List of Parameters

The following table describes the instrument parameters. Availability of some parameters depends on the options installed. See the comments in the "Notes" column of the table.

Parameter	Description	Definition	Notes
100BT Fall	Fall time between 2 levels (upper-base, base-lower) of a 3-level signal (100BT)		Available with ENET option.
100BT Rise	Rise time between 2 levels (Lower-base, base-upper) of a 3-level signal (100BT)		Available with ENET option.
100BT TIE	Difference between the measured and ideal times at level between base and upper orlower levels of 100BT signal.		Available with ENET option.
100BT Tj	Total jitter from a TIE at level between base and upper or lower levels of 100BTsignal.		Available with ENET option.
ACSN	Auto-correlation Signal-to-Noise provides a signal-to-noise ratio for periodic waveforms.		Available with DDM2 option. Standard in DDA-5005A.
AltNCycle	Alternate N-Cycle Plot. Timing of the transitions in the data waveform is measured for each transition and plotted as a function of the number of unit intervals over which the timing is measured. The N-cycle plot displays the mean or standard deviation of the edge placement in the waveform relative to each other (data to data) or to a reference clock (clock to data).		Available with ASDA option.
Amplitude	Measures the difference between upper and lower levels in two-level signals. Differs from pkpk in that noise, overshoot, undershoot, and ringing do not affect the measurement.	top - base	On signals not having two major levels (such as triangle or saw-tooth waves), returns same value as pkpk. Standard parameter.
Ampl asym	Amplitude asymmetry between taa+ and taa-	1 |(taa+ - taa-)|/(taa+ - taa-)	Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.

Parameter	Description	Definition	Notes
Area	Integral of data: Computes area of waveform between cursors relative to zero level. Values greater than zero contribute positively to the area; values less than zero negatively.	Sum from first to last of data multiplied by horizontal time between points	Standard parameter..
Avg Power	Average power of the waveform		Available with SDA and SDM options. Standard in SDA100G scopes.
Base	Lower of two most probable states (higher is top). Measures lower level in two-level signals. Differs from min in that noise, overshoot, undershoot, and ringing do not affect measurement.	Value of most probable lower state	On signals not having two major levels (triangle or saw-tooth waves, for example), returns same value as min. Standard parameter.
Bit Rate	One over duration of one UI measured on an eye		Available with SDA option. Standard in SDA, SDA100G, and WaveExpert scopes.
CMACp	PCI Express V TX-CM-ACp and V RX-CM-Acp		Available with PCIE option.
Cycles	Determines number of cycles of a periodic waveform lying between cursors. First cycle begins at first transition after the left cursor. Transition may be positive- or negative-going.	Number of cycles of periodic waveform	Standard parameter.
cyclic Mean	Cyclic mean: Computes the average of waveform data. Contrary to mean, computes average over an integral number of cycles, eliminating bias caused by fractional intervals.	Average of data values of an integral number of periods	Choose this parameter by selecting **Mean** from the parameter table, then touching the **Cyclic** checkbox. Standard parameter.
Area	Integral of data: Computes area of waveform between cursors relative to zero level. Values greater than zero contribute positively to the area; values less than zero negatively.	Sum from first to last of data multiplied by horizontal time between points	Standard parameter..
cyclic RMS	Cyclic root mean square: Computes square root of sum of squares of data values divided by number of points. Contrary to rms, calculation is performed over an integral number of cycles, eliminating bias caused by fractional intervals.	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_i)^2}$	Where: v_i denotes measured sample values, and N = number of data points within the periods found. Choose this parameter by selecting **RMS** from the parameter table, then touching the **Cyclic** checkbox. Standard parameter.

Parameter	Description	Definition	Notes
cyclic Std dev	Cyclic standard deviation: Standard deviation of data values from mean value over integral number of periods. Contrary to sdev, calculation is performed over an integral number of cycles, eliminating bias caused by fractional intervals.	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_i - mean)^2}$	Where: v_i denotes measured sample values, and N = number of data points within the periods found. Choose this parameter by selecting **Std dev** from the parameter table, then touching the **Cyclic** checkbox. Standard parameter.
DCD	Amount of jitter due to duty cycle distortion		Available with SDA and ENET options. Standard in SDA and WaveExpert scopes.
DDj	Amount of data dependent jitter in a signal		Available with SDA option.
Delay	Time from trigger to transition: Measures time between trigger and first 50% crossing after left cursor. Can measure propagation delay between two signals by triggering on one and determining delay of other.	Time between trigger and first 50% crossing after left cursor	Standard parameter.
Delta delay	delay: Computes time between 50% level of two sources.	Time between midpoint transition of two sources	Standard parameter.
Dj Effective	Amount of deterministic jitter (estimated) in a signal		Available with SDA option.
DOV	Differential Output Voltage of a 100Base-T signal		Available with ENET option.
Dperiod@level	Adjacent cycle deviation (cycle-to-cycle jitter) of each cycle in a waveform		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2 and XMAP options. Standard in SDA100G scopes.
Droop FG	1000Base-T test mode 1 droop from F to G		Available with ENET option.
Droop HJ	1000Base-T test mode 1 droop from H to J		Available with ENET option.

Parameter	Description	Definition	Notes
Dtime@level	t at level: Computes transition between selected levels or sources.	Time between transition levels of two sources, or from trigger to transition level of a single source This measurement gives the time of the source 2 edge minus the time of the source 1 edge.	Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Standard parameter.
Dtrig Time	Time from last trigger to this trigger		Standard in WaveRunner 6000A, WavePro 7000A, WaveMaster, and sampling scopes.
Duration	For single sweep waveforms, dur is 0; for sequence waveforms: time from first to last segment's trigger; for single segments of sequence waveforms: time from previous segment's to current segment's trigger; for waveforms produced by a history function: time from first to last accumulated waveform's trigger.	Time from first to last acquisition: for average, histogram or sequence waveforms	Standard parameter.
Duty@level	Percent of period for which data are above or below a specified level.		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2 and XMAP options.
Duty cycle	Duty cycle: Width as percentage of period.	width/period	Standard parameter.
Dwidth@level	Difference of adjacent width above or below a specified level.		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2 and XMAP options. Standard in SDA100G scopes.

Parameter	Description	Definition	Notes
Edge@level	Number of edges in waveform.		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2, USB2, SDA, and XMAP options. Standard in SDA100G and WavePro 7000A scopes.
Ext Ratio	Ratio of the power levels of an eye diagram		Available with SDA and SDM options. Standard in SDA, SDA100G, and WaveExpert scopes.
Excel	Performs measurements in Excel by transferring one or two waveforms and reading the resulting parameter value.		Available with XMAP and XDEV options. Standard on DDA-5005A scope. Excel must be loaded on the instrument.
Eye AC RMS	Root mean square of data within one UI		Standard in SDA and WaveExpert scopes.
Eye Amplitude	Difference of the levels of an eye diagram		Available with SDA and SDM options. Standard in SDA, SDA100G, and WaveExpert scopes.
Eye BER	Bit Error Rate estimated from an eye diagram		Available with SDA and SDM options. Standard in SDA, SDA100G, and WaveExpert scopes.
Eye Bit Rate	One over duration of one UI measured on an eye		Standard in SDA and WaveExpert scopes.
Eye Bit Time	Duration of one UI measured on an eye		Standard in SDA and WaveExpert scopes.
Eye Crossing	Level of the crossing in an eye diagram		Available with SDA and SDM options. Standard in SDA, SDA100G, and WaveExpert scopes.
Eye CrossN	Time of first crossing 50% level with negative edge of an eye relative to trigger or eye reference		Standard in SDA and WaveExpert scopes.
Eye CrossP	Time of first crossing 50% level with positive edge of an eye relative to trigger or eye reference		Standard in SDA and WaveExpert scopes.
Eye Cyc Area	The area under the mean persistence trace under first UI		Standard in SDA and WaveExpert scopes.
Eye Delay	Time of first crossing of an eye relative to trigger or eye reference		Standard in SDA and WaveExpert scopes.
Eye Delt Dly	Delay of crossing times between two eyes		Standard in SDA and WaveExpert scopes.

Parameter	Description	Definition	Notes
Eye FallTime	Fall time of the mean of persistence data		Standard in SDA and WaveExpert scopes.
Eye Height	Size of the vertical opening of an eye diagram		Available with SDA and SDM options. Standard in SDA, SDA100G, and WaveExpert scopes.
Eye Mean	Mean level of an eye		Standard in SDA and WaveExpert scopes.
Eye Mean	Mean level of an eye		Standard in SDA and WaveExpert scopes.
Eye OverN	Negative overshoot measured on an eye		Standard in SDA and WaveExpert scopes.
Eye OverP	Positive overshoot measured on an eye		Standard in SDA and WaveExpert scopes.
Eye Pk Noise	Peak-to-peak noise of a level on an eye diagram		Standard in SDA and WaveExpert scopes.
Eye PkPk Jit	Peak-to-peak jitter measured on eye persistence		Standard in SDA and WaveExpert scopes.
Eye Pulse Wid	The width of the eye measured at mid level		Standard in SDA and WaveExpert scopes.
Eye Q Factor	Q factor measured within the eye aperture		Available with SDA and SDM options. Standard in SDA, SDA100G, and WaveExpert scopes.
Eye RiseTime	Rise time of the mean of persistence data		Standard in SDA and WaveExpert scopes.
Eye RMS Jit	Root mean square jitter of an eye		Standard in SDA and WaveExpert scopes.
Eye SD Noise	The standard deviation of data on one eye level		Standard in SDA and WaveExpert scopes.
Eye SgToNoise	Signal to noise of an eye diagram		Standard in SDA and WaveExpert scopes.
Eye SupRatio	Suppression ratio of an eye		Standard in SDA and WaveExpert scopes.
Eye Width	Size of the horizontal opening of an eye diagram		Available with SDA and SDM options. Standard in SDA, SDA100G, and WaveExpert scopes.

<table>
<tr><th>Parameter</th><th>Description</th><th>Definition</th><th>Notes</th></tr>
<tr><td>Fall time</td><td>Fall time: Duration of falling edge from 90-10%.<table><tr><th>Thresh.</th><th>Remote</th><th>Lower Limit</th><th>Upper Limit</th><th>Default</th></tr><tr><td>Lower</td><td>Low</td><td>1%</td><td>45%</td><td>10%</td></tr><tr><td>Upper</td><td>High</td><td>55%</td><td>99%</td><td>90%</td></tr></table>Threshold arguments specify two vertical values on each edge used to compute fall time. Formulas for upper and lower values:
lower = lower thresh. x amp/100 + base
upper = upper thresh. x amp/100 + base</td><td>Time at upper threshold minus Time at lower threshold averaged over each falling edge</td><td>On signals not having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.
Standard parameter.</td></tr>
<tr><td>Fall 80-20%</td><td>Fall 80-20%: Duration of pulse waveform's falling transition from 80% to 20%, averaged for all falling transitions between the cursors.</td><td>Average duration of falling 80-20% transition</td><td>On signals not having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.
Standard parameter.</td></tr>
<tr><td>Fall@level</td><td>Fall at level: Duration of pulse waveform's falling edges between user-specified transition levels. See also Rise@level.</td><td>Transition time for % or absolute levels of all falling edges.
Enhanced version sets measurement calculations to use one of the following:
Base & Top (% or absolute)
Peak-Peak (%)
0V-Min (%)</td><td>On signals not having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.
Standard parameter.
Enhanced parameter available with EMC option.</td></tr>
<tr><td>First</td><td>Indicates value of horizontal axis at left cursor.</td><td>Horizontal axis value at left cursor</td><td>Indicates location of left cursor. Cursors are interchangeable: for example, the left cursor may be moved to the right of the right cursor and first will give the location of the cursor formerly on the right, now on left.
Standard parameter.</td></tr>
</table>

Parameter	Description	Definition	Notes
Frequency	Frequency: Period of cyclic signal measured as time between every other pair of 50% crossings. Starting with first transition after left cursor, the period is measured for each transition pair. Values then averaged and reciprocal used to give frequency.	1/period	Standard parameter.
Freq@level	Frequency at a specific level and slope for every cycle in waveform.		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2 and XMAP options. Standard in SDA100G and WavePro 7000A scopes.
FWHM	Measures the width of the largest area histogram peak at half of the population of the highest peak.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, and WaveExpert scopes.
Half period	Half period of a waveform.		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2, SDA, and XMAP options. Standard in SDA100G scopes.
Hist ampl	Difference in value between the two most populated peaks in a histogram.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.
Hist base	Value of the left-most of the two most populated histogram peaks.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.

Parameter	Description	Definition	Notes
Hist maximum	Value of the highest (right-most) populated bin in a histogram.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.
Hist Max Pop	Peak with maximum population in a histogram.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.
Hist mean	Average or mean value of data in the histogram.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, and WaveExpert scopes.
Hist median	Value of the "X" axis of a histogram that divides the population into two equal halves.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.
Hist minimum	Value of the lowest (left-most) populated bin in a histogram.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, and WaveExpert scopes.
Hist Mode	Position of the highest histogram peak.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.
Hist Pop@X	Population at bin for specified horizontal coordinate. You can place the cursor at any bin and use either Absolute, Reference, or Difference cursor shape.		Available with DDM2, JTA2, SDM, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, and sampling scopes.

Parameter	Description	Definition	Notes
Hist Range	Calculates range (max - min) of a histogram.		Available with DDM2, JTA2, ENET, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.
Hist rms	Root mean square of the values in a histogram.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, and sampling scopes
Hist sdev	Standard deviation of values in a histogram.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.
Hist top	Value of the right-most of the two most populated histogram peaks.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.
Hist X@peak	The value of the nth highest histogram peak.		Applies only to histograms. Available with JTA2, XMATH, XWAV, CAN02, DDM2, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, and WaveExpert scopes.
Hold time	Time from the clock edge to the data edge. You can set levels, slope, and hysteresis independently for Hold Clock and Hold Data. See also Setup parameter.		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2, ENET, USB2, SDA, and XMAP options. Standard in SDA100G scopes.
Hparam Script	Visual Basic (or Java) script which produces a measurement from one or two input histogram results		Available with XMAP, ASDA, and XDEV options. Standard in DDA-5005A.

Parameter	Description	Definition	Notes
Jitter Filter	Jitter in the specified frequency band. Generates a time sequence of jitter measurements that are filtered by the selected band-pass filter.		Available with ASDA option.
Last	Time from trigger to last (rightmost) cursor.	Time from trigger to last cursor	Indicates location of right cursor. Cursors are interchangeable: for example, the right cursor may be moved to the left of the left cursor and first will give the location of the cursor formerly on the left, now on right. Standard parameter.
Level@X	Gives the vertical value at the specified *x* position. If the *x* position is between two points, it gives the interpolated value. When the **Nearest point** checkbox is checked, it gives the vertical value of the nearest data point.		Standard parameter.
Local base	Value of the baseline for a local feature.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local bsep	Local baseline separation, between rising and falling slopes.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local max	Maximum value of a local feature.	If 25% of the way down toward min is more than 10,000 points, the extremal value is used as the local max.	Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local min	Minimum value of a local feature.	If 25% of the way up toward max is more than 10,000 points, the extremal value is used as the local min.	Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.

Parameter	Description	Definition	Notes
Local number	Number of local features (peak/trough pairs).		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local pkpk	Vertical difference between the peak and trough of a local feature (lmax lmin).		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local tbe	Time between events (between local peak and next trough or local trough and next peak).		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local tbp	Time between a local feature peak and the next local peak.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local tbt	Time between a local feature trough and the next local trough.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local tmax	Time of the maximum value of a local feature.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local tmin	Time of the minimum value of a local feature.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local tot	Time a local feature spends over a user specified percentage of its peak-to-trough amplitude.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local tpt	Time between local feature peak and trough.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.

Parameter	Description	Definition	Notes
Local ttp	Time between local feature trough and the next local peak.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Local tut	Time a local feature spends under a user specified percentage of its peak-to-trough amplitude.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
Mathcad	Produces a parameter using a user-specified Mathcad function.		Available with XMAP and XDEV option. Standard in DDA-5005A. Mathcad 2001i or later must be loaded on the instrument.
MATLAB	Produces a parameter using a user-specified MATLAB function.		Available with XMAP and XDEV option. Standard in DDA-5005A, WaveRunner 6000A, WaveMaster, WavePro 7000A, and sampling scopes MATLAB must be loaded on the instrument.
Maximum	Measures highest point in waveform. Unlike top, does not assume waveform has two levels.	Highest value in waveform between cursors	Gives similar result when applied to time domain waveform or histogram of data of same waveform. But with histograms, result may include contributions from more than one acquisition. Computes horizontal axis location of rightmost non-zero bin of histogram - not to be confused with maxp. Standard parameter.
Mean	Average of data for time domain waveform. Computed as centroid of distribution for a histogram.	Average of data	Gives similar result when applied to time domain waveform or histogram of data of same waveform. But with histograms, result may include contributions from more than one acquisition. Standard parameter.
Median	The average of base and top values.	Average of Base and Top.	Standard parameter.

Parameter	Description	Definition	Notes
Minimum	Measures the lowest point in a waveform. Unlike base, does not assume waveform has two levels.	Lowest value in waveform between cursors	Gives similar result when applied to time domain waveform or histogram of data of same waveform. But with histograms, result may include contributions from more than one acquisition. Standard parameter.
Nb phase	Provides a measurement of the phase at a specific frequency of a waveform (narrow band).		Available with DDM2, XMATH, SDA, and XMAP options. Standard in DDA-5005A and SDA100G scopes.
Nb Power	Provides a measurement of the power at a specific frequency of a waveform (narrow band).		Available with DDM2, XMATH, PMA2, SDA, and XMAP options. Standard in DDA-5005A and SDA100G scopes.
N-cycle jitter	Peak-to-peak jitter between edges spaced *n* UI apart.	Compares the expected time to the actual time of leading edges n bits apart.	Available in SDA analyzers.
NLTS	Provides a measurement of the nonlinear transition shift for a prml signal.		Available with DDM2 option. Standard in DDA-5005AA.
Num Points	Number of points in the waveform between the cursors.		Standard parameter.
One Level	One level of an eye diagram		Available with SDA and SDM options. Standard in SDA, SDA100G, and WaveExpert scopes.
Overshoot-	Overshoot negative: Amount of overshoot following a falling edge, as percentage of amplitude.	(base - min.)/ampl x 100	Waveform must contain at least one falling edge. On signals not having two major levels (triangle or saw-tooth waves, for example), may not give predictable results. Standard parameter.
Overshoot+	Overshoot positive: Amount of overshoot following a rising edge specified as percentage of amplitude.	(max. - top)/ampl x 100	Waveform must contain at least one rising edge. On signals not having two major levels (triangle or saw-tooth waves, for example), may not give predictable results. Standard parameter.
Overwrite	Ratio of residual-to-original power of a low frequency waveform overwritten by a higher frequency.		Available with DDM2 option. Standard in DDA-5005A.

Parameter	Description	Definition	Notes
Param Script	Visual Basic or Java script that produces a measurement from one or two input waveforms.		Available with XMAP, XDEV, and ASDA options. Standard in DDA-5005A.
Peak Mag	Peak mag away from a baseline. Note: the measure gate must include more of the baseline than any other single level.		Available with ENET option.
Peaks	Number of peaks in a histogram.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, and WaveExpert scopes.
Peak to peak	Peak-to-peak: Difference between highest and lowest points in waveform. Unlike ampl, does not assume the waveform has two levels.	maximum - minimum	Gives a similar result when applied to time domain waveform or histogram of data of the same waveform. But with histograms, result may include contributions from more than one acquisition. Standard parameter.
Percentile	Horizontal data value that divides a histogram so the population to the left is xx% of the total.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes.
Per DCD	Amount of jitter due to duty cycle distortion		Standard in SDA and WaveExpert scopes.
Per Duty Cyc	Duty cycle measured on a persistence		Standard in SDA and WaveExpert scopes.
Period	Period of a cyclic signal measured as time between every other pair of 50% crossings. Starting with first transition after left cursor, period is measured for each transition pair, with values averaged to give final result.	$\frac{1}{Mr}\sum_{i=1}^{Mr}\left(Tr_i^{50} - Tr_i^{50}\right)$ Where: Mr is the number of leading edges found, Mf the number of trailing edges found, Tr_i^x the time when rising edge i crosses the x% level, and Tf_i^x the time when falling edge i crosses the x% level.	Standard parameter.

Parameter	Description	Definition	Notes
Period@level	Period at a specified level and slope for every cycle in waveform.		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2, AORM, ENET, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, and WavePro 7000A scopes.
Per Pulse Sym	Symmetry of RZ pulse around eye aperture center		Standard in SDA and WaveExpert scopes.
Persist Area	Area under mean persistence trace		Standard in SDA and WaveExpert scopes.
Persist Max	Highest vertical value of input persistence		Standard in SDA and WaveExpert scopes.
Persist Mean	Average of persistence data		Standard in SDA and WaveExpert scopes.
Persist Mid	Mid level between Maximum and Minimum data		Standard in SDA and WaveExpert scopes.
Persist Min	Lowest vertical value of input persistence		Standard in SDA and WaveExpert scopes.
Persist PkPk	Difference between maximum and minimum data values		Standard in SDA and WaveExpert scopes.
Persist RMS	Root mean square of persistence data		Standard in SDA and WaveExpert scopes.
Phase	Phase difference between signal analyzed and signal used as reference. You can set the output type to percent, degrees, or radians. After setting up the reference, touch the **More** tab for signal setups.	Phase difference between signal and reference	Standard parameter.
Pj	Periodic component of jitter		Available with SDA option.
Power Factor	Ratio of real to apparent power		Available with PMA2 option.
PW50	Average pulse width at the 50% point between the local baseline and the local peak or trough.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
PW50-	Average pulse width at the 50% point between the local baseline and the local trough.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.

<table>
<tr><th>Parameter</th><th>Description</th><th>Definition</th><th>Notes</th></tr>
<tr><td>PW50+</td><td>Average pulse width at the 50% point between the local baseline and the local peak.</td><td></td><td>Hysteresis argument used to discriminate levels from noise in data.
Available with DDM2 option.
Standard in DDA-5005A.</td></tr>
<tr><td>Real Power</td><td>Mean of the product of voltage and current waveform (or mean of the instantaneous power)</td><td></td><td>Available with PMA2 option.</td></tr>
<tr><td>Resolution</td><td>Ratio of taa for a high and low frequency waveform</td><td>taa (HF)/mean taa (LF)*100</td><td>Hysteresis argument used to discriminate levels from noise in data.
Available in DDM2.
Standard in DDA-5005A.</td></tr>
<tr><td>Ring</td><td>Ringback (high or low)</td><td></td><td>Available with SDA parameter.</td></tr>
<tr><td>Rj Effective</td><td>Amount of random jitter (estimated) in a signal</td><td></td><td>Available with SDA parameter.</td></tr>
<tr><td>Rise</td><td>Rise time: Duration of rising edge from 10-90%.
<table><tr><th>Thresh.</th><th>Remote</th><th>Lower Limit</th><th>Upper Limit</th><th>Default</th></tr><tr><td>Lower</td><td>Low</td><td>1%</td><td>45%</td><td>10%</td></tr><tr><td>Upper</td><td>High</td><td>55%</td><td>99%</td><td>90%</td></tr></table>Threshold arguments specify two vertical values on each edge used to compute rise time.
Formulas for upper and lower values:
lower = lower thresh. x amp/100 + base
upper = upper thresh. x amp/100 + base</td><td>Time at lower threshold minus Time at upper threshold
averaged over each rising edge</td><td>On signals not having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.
Standard parameter.</td></tr>
<tr><td>Rise 20-80%</td><td>Rise 20% to 80%: Duration of pulse waveform's rising transition from 20% to 80%, averaged for all rising transitions between the cursors.</td><td>Average duration of rising 20-80% transition</td><td>On signals not having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results.
Standard parameter.</td></tr>
</table>

Parameter	Description	Definition	Notes
Rise@level	Rise at level: Duration of pulse waveform's rising edges between transition levels.	Slew rate for % or absolute levels of rising or falling edges. Enhanced version sets measurement calculations to use one of the following: Base & Top (% or absolute) Peak-Peak (%) 0V-Max (%)	On signals not having two major levels (triangle or saw-tooth waves, for example), top and base can default to maximum and minimum, giving, however, less predictable results. Standard parameter. Enhanced parameter available with EMC option.
RMS	Root Mean Square of data between the cursors - about same as sdev for a zero-mean waveform.	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_i)^2}$ Where: v_i denotes measured sample values, and N = number of data points within the periods found up to maximum of 100 periods.	Gives similar result when applied to time domain waveform or histogram of data of same waveform. But with histograms, result may include contributions from more than one acquisition. Standard parameter.
SAS	Signal Amplitude Symmetry of a 100Base-T signal		Available with ENET option.
SD2Skew	Calculates the time skew between 2 serial data lanes		Available with SDA and PCIE options.
Setup	Time from the data edge to the clock edge.		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2 and XMAP options. Standard in SDA and SDA100G scopes.

Parameter	Description	Definition	Notes
Skew	Time of clock1 edge minus time of nearest clock2 edge.		Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Hysteresis on a measurement (if set to 500 mdiv) requires that the signal must transition one way 1/2 division (total swing) across the threshold for the measurement to be valid. Available with JTA2 and XMAP options. Standard in SDA100G and WaveSurfer scopes.
Slew Rate	Slew rate or local dV/dt in a transition zone		Available in SDA and JTA2 options. Standard in SDA100G scopes.
SSC Diff	Calculates difference between average SSC frequencies.		Available with PCIE option.
SSC Frequency	Frequency of Spread Spectrum Clock signal.		Available with PCIE option.
SSC Ratio	Calculates the ratio between the maximum and minimum SSC frequencies.		Available with PCIE option.
SSC Track	Tracks Spread Spectrum Clock. Filtered track of frequency at level.		Available with ASDA option.
Std dev	Standard deviation of the data between the cursors - about the same as rms for a zero-mean waveform.	$\sqrt{\frac{1}{N}\sum_{i=1}^{N}(v_i - mean)^2}$	Gives similar result when applied to time domain waveform or histogram of data of same waveform. But with histograms, result may include contributions from more than one acquisition. Where: v_i denotes measured sample values, and N = number of data points within the periods found up to maximum of 100 periods. Standard parameter.
TAA	Average peak-to-trough amplitude for all local features.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.

Parameter	Description	Definition	Notes
TAA-	Average local baseline-to-trough amplitude for all local features.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
TAA+	Average local baseline-to-peak amplitude for all local features.		Hysteresis argument used to discriminate levels from noise in data. Available with DDM2 option. Standard in DDA-5005A.
TIE@level	Difference between the measured times of crossing a given slope and level and the ideal expected time. For Slope you can choose positive, negative, or both. For output units you can choose time or unit interval (UI). A unit interval equals one clock period. The Virtual Clock setup gives you a choice of Standard (1.544 MHz) or Custom reference clocks. You can also use a mathematically derived Golden PLL to filter low frequency jitter. The cutoff frequency is user selectable.	Cutoff Freq = (1/1.667e3) x Clock Freq	Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2, ENET, SDA, and XMAP options. Standard in SDA100G, WavePro 7000A, WaveExpert, and sampling scopes.
Time@edge	Measures time at each edge on each digital line		Available with MS-32 option.
Time@level	Time at level: Time from trigger (t=0) to crossing at a specified level.	Time from trigger to crossing level	Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Standard parameter.
Top	Higher of two most probable states, the lower being base; it is characteristic of rectangular waveforms and represents the higher most probable state determined from the statistical distribution of data point values in the waveform.	Value of most probable higher state	Gives similar result when applied to time domain waveform or histogram of data of same waveform. But with histograms, result may include contributions from more than one acquisition. Standard parameter.
Total Jitter	Total jitter at a given bit error rate		Available with ENET and SDA options.

Parameter	Description	Definition	Notes
Total Pop	Total population of a histogram.		Available with DDM2, JTA2, XMATH, XWAV, CAN02, SDA, and XMAP options. Standard in DDA-5005A, SDA100G, WaveExpert, and sampling scopes
tUpS	Upsamples a time parameter by nX		Available with SDA and SDM options.
TxCmD	PCI Express: V TX-CM-DC-LINE-DELTA Absolute delta of DC common mode voltage between D+ and D-		Available with PCIE option.
TxFall	Fall2080 and ParamRescale, to get UI		Available with PCIE option.
TxRise	Rise2080 and ParamRescale, to get UI		Available with PCIE option.
Vcross	Voltage at which two signals cross. That is, voltage of either signal at the time when difference is zero.		Available with SDA and PCIE options.
Vdiff	Used for V TX-DIFFp-p and V RX-DIFFp-p for PCI-Express		Available with PCIE option.
VTxDeRatio	Ratio between transition and de-emphasized bits		Available with PCIE option.
Width	Width of cyclic signal determined by examining 50% crossings in data input. If first transition after left cursor is a rising edge, waveform is considered to consist of positive pulses and width the time between adjacent rising and falling edges. Conversely, if falling edge, pulses are considered negative and width the time between adjacent falling and rising edges. For both cases, widths of all waveform pulses are averaged for the final result.	Width of first positive or negative pulse averaged for all similar pulses	Similar to fwhm, though, unlike width, that parameter applies only to histograms. Standard parameter.

Parameter	Description	Definition	Notes
Width@level	Width measured at a specific level.	Time between two transitions of opposite slope at a specified level. (Slope specified for 1st transition.) Enhanced version sets measurement calculations to use one of the following: Base & Top (% or absolute) Peak-Peak (%) 0V-Max (%) 0V-Min (%)	Reference levels and edge-transition polarity can be selected. Hysteresis argument used to discriminate levels from noise in data. Available with JTA2, USB2, EMC, SDA, and XMAP options. Standard in SDA100G and WavePro 7000A scopes. Enhanced parameter available with EMC option.
WidthN	Width measured at the 50% level and negative slope.		Standard parameter.
X@max	Determines the horizontal axis location of the maximum value between the cursors.		Restricted to time and frequency waveforms only.
X@min	Determines the horizontal axis location of the minimum value between the cursors.		Restricted to time and frequency waveforms only.
Zero Level	Zero level of an eye diagram		Available with SDA and SDM options. Standard in SDA, SDA100G, and WaveExpert scopes.

Qualified Parameters

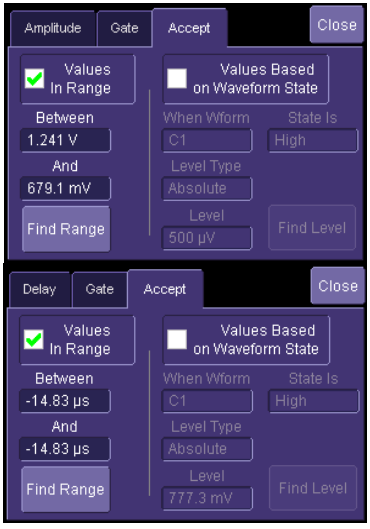
Some LeCroy instruments and software packages give you the ability to constrain parameter measurements to a vertically or horizontally limited range, or to occurrences gated by a second waveform. Furthermore, both constraints can operate together. This capability enables you to exclude unwanted characteristics from your measurements. It is much more restrictive than Measure Gate, which is used only to narrow the span of the waveform for analysis, along the horizontal axis.

Note: Since this feature operates on only a subset of the data, possible alerts or status indicators concerning the measurement (such as " Data range too low") are not displayed.

Range Limited Parameters

Range Qualifier Setup

1. From the menu bar, select **Measure**, then **Measure Setup...** from the drop-down menu.
2. Touch a **Px** tab to open the setup dialog.
3. Touch inside the **Source** field and select a source from the pop-up menu.
4. Touch inside the **Measure** field and select a parameter from the pop-up menu.
5. Touch the **Accept** tab of the right-hand dialog, then touch the **Values In Range** checkbox. Depending on whether you select a vertical or horizontal parameter, the correct units will be automatically displayed (V, s, Hz, dB) in the **Between** and **And** fields. Or, if you select a simple ratio parameter (such as power factor) that yields a dimensionless number, no units will be displayed.
6. Touch the **Find Range** button to quickly display the most recent value of the parameter measurement. From there it is a simple matter to set the desired range.



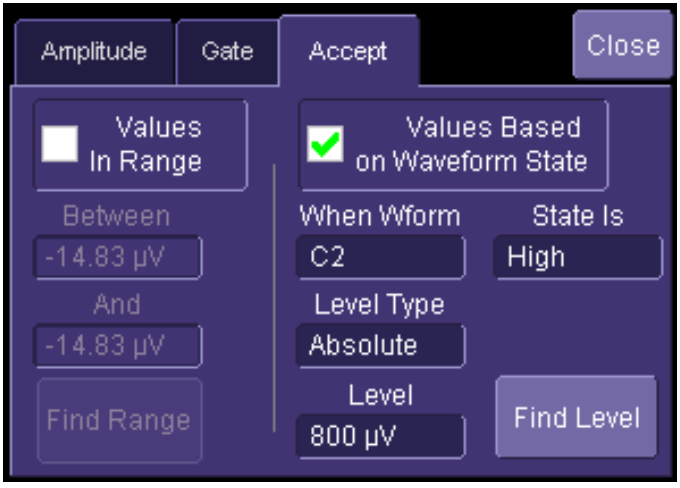
Amplitude Parameter (range set manually)

Delay Parameter (Find Range selected)

Waveform Gated Parameters

Waveform Qualifier Setup

1. From the menu bar, select **Measure**, then **Measure Setup...** from the drop-down menu.
2. Touch a **Px** tab to open the setup dialog.
3. Touch inside the **Source** field and select a source from the pop-up menu.
4. Touch inside the **Measure** field and select a parameter from the pop-up menu.
5. Touch the **Accept** tab of the right-hand dialog, then touch the **Values Based on Waveform State** checkbox.
6. Touch inside the **When Wform** field and select the gating source.
7. Touch inside the **State Is** field and select **High** or **Low** from the pop-up menu. Parameter measurements on the subject waveform will only be taken when the gating waveform is in the selected state.
8. Touch inside the **Level Type** field and select **Absolute** or **Percent** from the pop-up menu.
9. Touch inside the **Level** field and enter the crossing level value at which you want measurements to begin. Alternatively, touch the **Find Level** button to automatically select the 50% level of your gating waveform.



WAVEFORM MATH

Introduction to Math Traces and Functions

With the instrument's math tools you can perform mathematical functions on a waveform displayed on any channel, or recalled from any of the four reference memories M1 to M4. You can also set up traces F1 to Fx [The number of math functions that can be performed at the same time depends on the software options loaded on your scope.] to do math on parameter measurements P1 to Px [The number of parameters that can be measured at the same time depends on the software options loaded on your scope.].

For example: you could set up Trace F1 as the difference between Channels 1 and 2, Trace F2 as the average of F1, and Trace F3 as the integral of F2. You could then display the integral of the averaged difference between Channels 1 and 2. Any trace and function can be chained to another trace and function. For example, you could make Trace F1 an average of Channel 1, Trace F2 an FFT of F1, and Trace F3 a zoom of F2.

Note: Math traces F5-F8 are available only if you have loaded software option package XMATH or XMAP on WaveMaster or WavePro scopes, but are standard on Disk Drive Analyzers and Serial Data Analyzers.

MATH MADE EASY

With the instrument's math tools you can perform mathematical functions on a waveform displayed on any channel C1 to C4, or recalled from any of the four reference memories M1 to M4. To do computations in sequence, you can also use math functions F1 to Fx as a source input waveform. Or you can use Parameters P1 through Px

For example: you could set up F1 as the difference between Channels 1 and 2, F2 as the average of F1, and F3 as the integral of F2. You could then display the integral of the averaged difference between Channels 1 and 2. Any trace and function can be chained to another trace and function. For example, you could make F1 an average of Channel 1, F2 an FFT of F1, and F3 a zoom of F2.

Refer to the Specifications to find out which math tools are available in each optional package.

Setting Up a Math Function

Math Setup

This setup mode allows you to quickly apply frequently used math functions.

1. In the menu bar, touch **Math**, then **Math setup...**
2. If there are math functions already assigned to **F1** through **Fx** [The number of math traces available depends on the software options loaded on your scope. See specifications.], touch the checkbox for the function you want to enable.
3. To assign a new math function to a trace, touch the Fx button for that trace, for example 

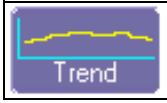
. The math function menu appears.
4. Touch a menu selection; your new function is automatically assigned, with the same setups as were in place for the last function in that Fx position.
5. If you want to change other setup items, like the source waveform, touch the appropriate Fx tab, for example 

. The setup dialog for that Fx position appears.
6. Touch the **Single** function button 

if you want to perform just one math function on the trace, or touch the **Dual** function button 

to perform math on math.

7. Touch the **Graph** button, then touch inside the **Graph with** field to select a graph mode. The Graph modes are as follows:

	Histogram of the values of a parameter
	Track of the values of a parameter
	Trend of the values of a parameter

Resampling To Deskew

Deskew whenever you need to compensate for different lengths of cables, probes, or anything else that might cause timing mismatches between signals. Resample a signal on one channel and adjust it in time relative to a signal on another channel.

Resampling

1. In the menu bar, touch **Math**, then **Math Setup...** in the drop-down menu.
2. Touch a math function trace tab **F1** through **Fx** The number of math traces available depends on the software options loaded on your scope. See Specifications..
3. Touch the **single** function button.
4. Touch inside the **Source1** field and select a source: channel, math trace, memory location.
5. Touch inside the **Operator1** field and select **Deskew** from the **Functions** category.
6. In the dialog on the right, touch the **Deskew** tab.
7. Touch inside the **Delay by** data entry field and type in a time value, using the pop-up keypad.

Rescaling and Assigning Units

This feature allows you to apply a multiplication factor (*a*) and additive constant (*b*) to your waveform: *a*X + *b*. You can do it in the unit of your choice, depending on the type of application. Allowable unit abbreviations are as follows:

(blank)	no units
A	Ampere
C	Coulomb
CYCLE	cycles
DB	Decibel
DBC	Decibel referred to carrier
DBM	Decibel Milliwatt
DBV	Decibel Volts
DBUZ	Decibel Microamp
DEC	Decade
DIV	Divisions

Event	Events
F	Farad
G	Gram
H	Henry
HZ	Hertz
J	Joule
K	Degree Kelvin
CEL	Degree Celsius
FAR	Degree Fahrenheit
L	Liter
M	Meter
FT	Foot
IN	Inch
YARD	yard
MILE	mile
N	Newton
OHM	Ohm
PAL	Pascal
PCT	Percent
POISE	Poise
PPM	parts per million
RAD	Radian
DEG	Degree (of arc)
MNT	Minute (of arc)
SAMPLE	sample
SWEEP	sweeps
SEC	Second (of arc)
S	Second
SIE	Siemens
T	Tesla
UI	Unit interval
V	Volt
VA	Volt amps
W	Watt
WB	Weber
MIN	min
HOUR	Hour
DAY	Day
WEEK	Week

You can also enter combinations of the above units following the SI rules:

- for the quotient of two units, the character **/** should be used
- for the product of two units, the character **.** should be used
- exponents can be represented by a digit appended to the unit without a space

For example,

- acceleration can be entered as M/S2 for meters per second squared
- volts seconds can be entered as V.S

In some cases, the units entered may be converted to simple units. For example entering V.A will display W (watts)

Rescaling Setup

1. In the menu bar, touch **Math**, then **Math Setup...** in the drop-down menu.
2. Touch a math function trace tab **F1** through **Fx** The number of math traces available depends on the software options loaded on your scope. See Specifications..
3. Touch the **single** function button.
4. Touch inside the **Source1** data entry field and select a source: channel, math trace, memory location.
5. Touch inside the **Operator1** data entry field and select **Rescale** from the **Functions** category.
6. In the dialog on the right, touch the **Rescale** tab.
7. Touch inside the **First multiply by** checkbox and enter a value for *a*, the multiplication factor.
8. Touch inside the **then add:** data entry field and enter a value for *b*, the additive constant.
9. Touch inside the **Override units** checkbox to disregard the source waveform's units, using the pop-up keyboard.

Averaging Waveforms

Summed vs. Continuous Averaging

For Summed averaging, you specify the number of acquisitions to be averaged. The averaged data is updated at regular intervals and presented on the screen.
On the other hand, Continuous averaging (the system default) helps to eliminate the effects of noise by continuously acquiring new data and adding the new waveforms into the averaging buffer. You determine the importance of new data vs. old data by assigning a weighting factor. Continuous averaging allows you to make adjustments to a system under test and to see the results immediately.

Note: Continuous Averaging is accessible from the channel "Vertical Adjust" dialog under "Pre-Processing," and from the math function menu.

Summed Averaging

Summed Averaging is the repeated addition, with equal weight, of successive source waveform records. If a stable trigger is available, the resulting average has a random noise component lower than that of a single-shot record. Whenever the maximum number of sweeps is reached, the averaging process stops.
An even larger number of records can be accumulated simply by changing the number in the dialog. However, the other parameters must be left unchanged or a new averaging calculation will be started. You can pause the averaging by changing the trigger mode from NORM/AUTO to STOP. The instrument resumes averaging when you change the trigger mode back to NORM/AUTO.
You can reset the accumulated average by pushing the CLEAR SWEEPS button or by changing an acquisition parameter such as input gain, offset, coupling, trigger condition, timebase, or bandwidth limit. The number of current averaged waveforms of the function, or its zoom, is shown in the acquisition status dialog. When summed averaging is performed, the display is updated at a reduced rate to increase the averaging speed (points and events per second).

Continuous Averaging

Continuous Averaging, the default setting, is the repeated addition, with unequal weight, of successive source waveforms. It is particularly useful for reducing noise on signals that drift very slowly in time or amplitude. The most recently acquired waveform has more weight than all the previously acquired ones: the continuous average is dominated by the statistical fluctuations of the most recently acquired waveform. The weight of 'old' waveforms in the continuous average gradually tends to zero (following an exponential rule) at a rate that decreases as the weight increases.

The formula for continuous averaging is

new average = (new data + weight * old average)/(weight + 1)

This is also the formula used to compute summed averaging. But by setting a "sweeps" value, you establish a fixed weight that is assigned to the old average once the number of "sweeps" is reached. For example, for a sweeps (weight) value of **4**:

1st sweep (no old average yet): new average = (new data +0 * old average)/(0 + 1) = new data only

2nd sweep: new average = (new data + 1*old average)/(1 + 1) = 1/2 new data +1/2 old average

3rd sweep: new average = (new data + 2 * old average)/(2 + 1) = 1/3 new data + 2/3 old average

4th sweep: new average = (new data + 3 * old average)/(3 + 1) = 1/4 new data + 3/4 old average

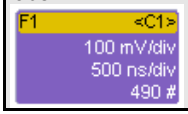
5th sweep: new average = (new data + 4 * old average)/(4 + 1) = 1/5 new data + 4/5 old average

6th sweep: new average = (new data + 4 * old average)/(4 + 1) = 1/5 new data + 4/5 old average

7th sweep: new average = (new data + 4 * old average)/(4 + 1) = 1/5 new data + 4/5 old average

In this way, for sweeps > 4 the importance of the old average begins to decrease exponentially.

Note: The number of sweeps used to compute the average will be displayed in the bottom line of the trace descriptor label:



Continuous Averaging Setup

1. In the menu bar, touch **Math**, then **Math Setup...** in the drop-down menu.
2. Select a function tab from **F1** through **Fx** [The number of math traces available depends on the software options loaded on your scope. See Specifications.].
3. Touch inside the **Source1** field and select a source waveform from the pop-up menu.
4. Touch inside the **Operator1** field and select **Average** from the **Select Math Operator** menu.
5. Touch the **Average** tab in the dialog to the right of the "Fx" dialog, touch the **Continuous** button.
6. Touch inside the **Sweeps** data entry field and enter a value using the pop-up keypad. The valid range is 1 to 1,000,000 sweeps.

Summed Averaging Setup

1. In the menu bar, touch **Math**, then **Math Setup...** in the drop-down menu.
2. Select a function tab from **F1** through **Fx** [The number of math traces available depends on the software options loaded on your scope. See Specifications.].
3. Touch inside the **Source1** field and select a source waveform from the pop-up menu.
4. Touch inside the **Operator1** field and select **Average** from the **Select Math Operator** menu.
5. Touch the **Average** tab in the dialog to the right of the "Fx" dialog, then touch the **Summed** button.
6. Touch inside the **Sweeps** data entry field and type in a value using the pop-up keypad. The valid range is 1 to 1,000,000 sweeps.

Enhanced Resolution

ERES (Enhanced Resolution) filtering increases vertical resolution, allowing you to distinguish closely spaced voltage levels. The functioning of the instrument's ERES is similar to smoothing the signal with a simple, moving-average filter. However, it is more efficient concerning bandwidth and pass-band filtering. Use ERES on single-shot waveforms, or where the data record is slowly repetitive (when you cannot use averaging). Use it to reduce noise when your signal is noticeably noisy, but you do not need to perform noise measurements. Also use it when you perform high-precision voltage measurements: zooming with high vertical gain, for example.

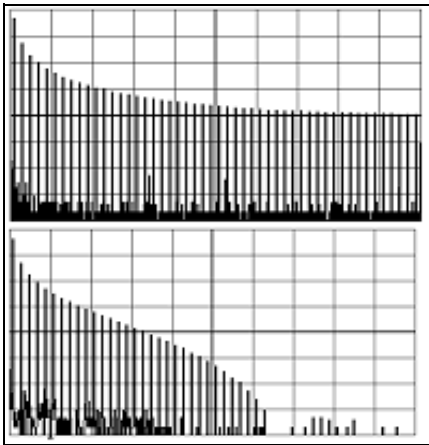
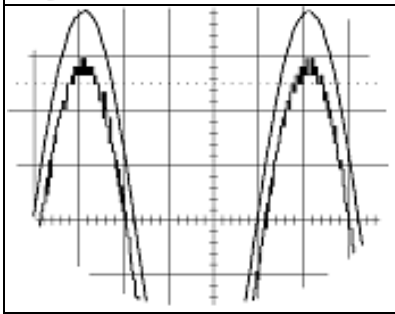
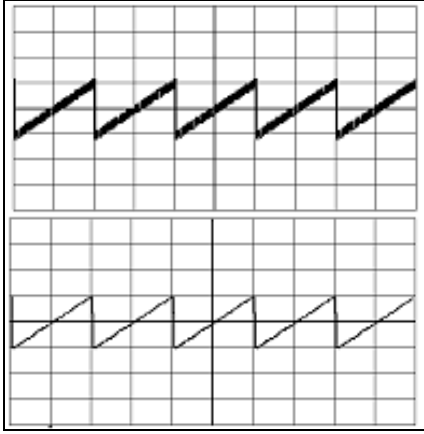
How the Instrument Enhances Resolution

The instrument's enhanced resolution feature improves vertical resolution by a fixed amount for each filter. This real increase in resolution occurs whether or not the signal is noisy, or your signal is single-shot or repetitive. The signal-to-noise ratio (SNR) improvement you gain is dependent on the form of the noise in the original signal. The enhanced resolution filtering decreases the bandwidth of the signal, filtering out some of the noise.

The instrument's constant phase FIR (Finite Impulse Response) filters provide fast computation, excellent step response in 0.5 bit steps, and minimum bandwidth reduction for resolution improvements of between 0.5 and 3 bits. Each step corresponds to a bandwidth reduction factor of two, allowing easy control of the bandwidth resolution trade-off. The parameters of the six filters are given in the following table.

Resolution increased by	-3 dB Bandwidth (× Nyquist)	Filter Length (Samples)
0.5	0.5	2
1.0	0.241	5
1.5	0.121	10
2.0	0.058	24
2.5	0.029	51
3.0	0.016	117

With low-pass filters, the actual SNR increase obtained in any particular situation depends on the power spectral density of the noise on the signal.
The improvement in SNR corresponds to the improvement in resolution if the noise in the signal is white - evenly distributed across the frequency spectrum.
If the noise power is biased towards high frequencies, the SNR improvement will be better than the resolution improvement.
The opposite may be true if the noise is mostly at lower frequencies. SNR improvement due to the removal of coherent noise signals - feed-through of clock signals, for example - is determined by the fall of the dominant frequency components of the signal in the passband. This is easily ascertained using spectral analysis. The filters have a precisely constant zero-phase response. This has two benefits. First, the filters do not distort the relative position of different events in the waveform, even if the events' frequency content is different. Second, because the waveforms are stored, the delay normally associated with filtering (between the input and output waveforms) can be exactly compensated during the computation of the filtered waveform.
The filters have been given exact unity gain at low frequency. Enhanced resolution should therefore not cause overflow if the source data is not overflowed. If part of the source trace were to overflow, filtering would be allowed, but the results in the vicinity of the overflowed data - the filter impulse response length - would be incorrect. This is because in some circumstances an overflow may be a spike of only one or two samples, and the energy in this spike may not be enough to significantly affect the results. It would then be undesirable to disallow the whole trace.
The following examples illustrate how you might use the instrument's enhanced resolution function.

	**In low-pass filtering:** The spectrum of a square signal before (left top) and after (left bottom) enhanced resolution processing. The result clearly illustrates how the filter rejects high-frequency components from the signal. The higher the bit enhancement, the lower the resulting bandwidth.
	**To increase vertical resolution:** In the example at left, the lower ("inner") trace has been significantly enhanced by a three-bit enhanced resolution function.
	**To reduce noise:** The example at left shows enhanced resolution of a noisy signal. The original trace (left top) has been processed by a 2-bit enhanced resolution filter. The result (left bottom) shows a "smooth" trace, where most of the noise has been eliminated.

Note: Enhanced resolution can only improve the resolution of a trace; it cannot improve the accuracy or linearity of the original quantization. The pass-band will cause signal attenuation for signals near the cut-off frequency. The highest frequencies passed may be slightly attenuated. Perform the filtering on finite record lengths. Data will be lost at the start and end of the waveform: the trace will be slightly shorter after filtering. The number of samples lost is exactly equal to the length of the impulse response of the filter used: between 2 and 117 samples. Normally this loss (just 0.2 % of a 50,000 point trace) is not noticed. However, you might filter a record so short there would be no data output. In that case, however, the instrument would not allow you to use the ERES feature.

Enhanced Resolution (ERES) Setup

1. In the menu bar, touch **Math**, then **Math Setup...** in the drop-down menu.
2. Touch a function tab **F1** through **Fx** [The number of math traces available depends on the software options loaded on your scope. See Specifications.].
3. Touch inside the **Operator1** data entry field.
4. Select **ERES** from the **All Functions** or **Filter** group of Math functions.
5. Touch the **Trace On** checkbox.
6. Touch the "ERES" tab in the right-hand dialog, then touch inside the **bits** field and make an "Enhance by" selection from the pop-up menu



Waveform Copy

The **Copy** math function  makes a copy of your present waveform in its unprocessed state. While processing may continue on the original waveform, the copy enables faster throughput in some cases by preserving the original data. That is, no calculations need to be undone on the copy before additional math can be calculated.
This benefit of faster throughput, however, comes at the expense of memory usage.

Waveform Sparser

The Sparse math function 

allows you to thin out an incoming waveform by skipping points at regular intervals, and by starting acquisition at a particular "offset" (point). The **Sparsing factor** specifies the number of sample points to reduce the input waveform by. A sparsing factor of 4, for example, tells the scope to retain only one out of every 4 samples. A **Sparsing offset** of 3, on the other hand, tells the scope to begin on the third sample, then skip the number of samples specified by the sparsing factor (4). In this way, the sample rate is effectively reduced. For the sparsing factor (interval), you can set a value from 1 to 1,000,000 points. For the sparsing offset you can set a value from 0 to 999,999.

Note: The maximum sparsing offset that can be entered for any sparsing factor equals Sparsing Factor 1.

Waveform Sparser Setup

1. In the menu bar, touch **Math**, then **Math setup...** in the drop-down menu.
2. Touch the tab for the function (F1 to Fx [The number of math traces available depends on the software options loaded on your scope. See specifications.]) you want to assign the Sparse operation to.
3. Touch inside the **Source1** field and select an input waveform.
4. Touch inside the **Operator1** field and select **Sparse** from the **Select Math Operator** menu.
5. Touch inside the **Sparsing factor** field and enter a value, using the pop-up keypad.
6. Touch inside the **Sparsing offset** field and enter a value, using the pop-up keypad.

Interpolation

Linear interpolation, which inserts a straight line between sample points, is best used to reconstruct straight-edged signals such as square waves. (Sinx)/x interpolation, on the other hand, is suitable for reconstructing curved or irregular waveshapes, especially when the sampling rate is 3 to 5 times the system bandwidth. The instrument also gives you a choice of Cubic interpolation.
For each method, you can select a factor from 2 to 50 points by which to interpolate (upsample).

Interpolation Setup

1. In the menu bar, touch **Math**, then **Math setup...** in the drop-down menu.
2. Touch the tab for the function (F1 to Fx The number of math traces available depends on the software options loaded on your scope. See Specifications.) you want to assign the Interpolate operation to.
3. Touch inside the **Source1** field and select an input waveform.
4. Touch inside the **Operator1** field, then touch the **Filter** button in the **Select Math Operator** menu.
5. Select **Interpolate** from the Filter submenu.
6. Touch the "Interpolate" tab in the mini setup dialog to the right of the main dialog.
7. Touch inside the **Algorithm** field and select an interpolation type.
8. Touch inside the **Upsample by** [Upsampling is the factor by which sampling is increased.] field and enter a value, using the pop-up numeric keypad, if you want to enter a specific value. Otherwise, use the **Up/Down** buttons to increment the displayed value in a 1-2-5 sequence.

Fast Wave Port

FastWavePort is a processing function for the LeCroy X-Stream family of digital oscilloscopes that enables you to insert your own custom processing algorithm, written in the C/C++ language, into the DSO's processing stream. FastWavePort maximizes data throughput from the acquisition system to your processing function. It also makes it simple to create these custom processing functions.

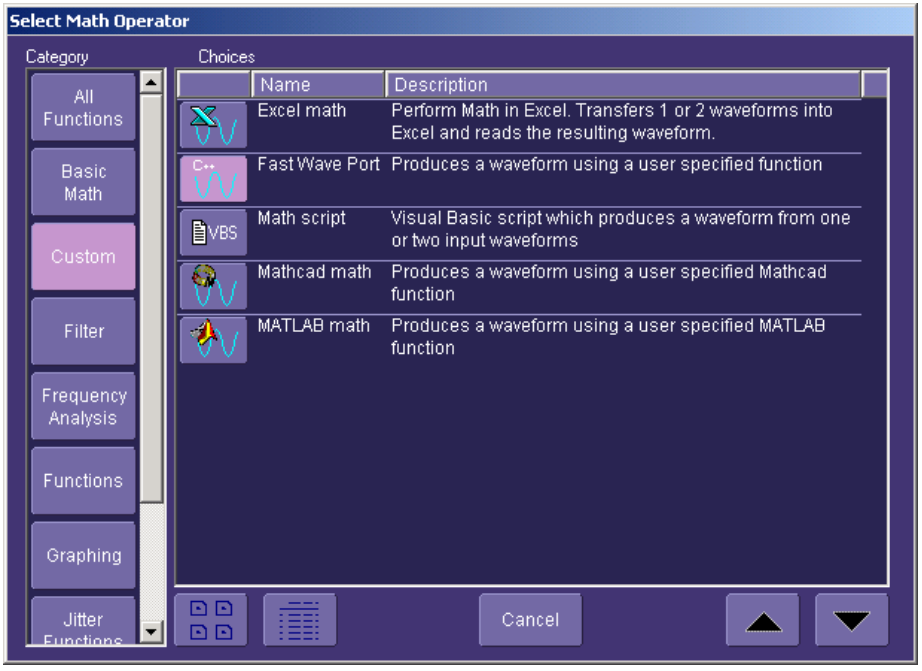
The technology that makes this system possible is the ability of two processes in a Windows system to share a region of memory. This enables the transfer of data at high-speed between the acquisition software and the custom processing function, which runs in a separate process from the DSO application. A major benefit of FastWavePort is that your application may be implemented and, more importantly, debugged independently of the main application.
It is important to note that the transfer of the results of your processing function back into the X-Stream processing stream is optional. If performance is the primary goal, and display or further processing of the results within the DSO software is not required, then this may be skipped.

Fast Wave Port Setup - Initial

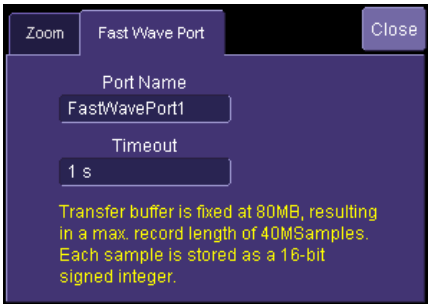
1. In the menu bar, touch **Math**, then **Math Setup...** in the drop-down menu.
2. Touch one of the Math function tabs, 

for example.
3. Touch inside the "Source1" field, and select a signal source from the pop-up menu. The source can be a channel waveform, math or memory trace, or a parameter.
4. Touch inside the "Operator"1 field and select **FastWavePort** 

from the **Custom** menu:



5. In the right-hand mini-dialog, touch the **Fast Wave Port** tab:



6. Touch inside the "Timeout" field and enter a suitable value.

Setup - Case 1

This scenario assumes you have developed your application on a PC.

1. Compile your application on your PC.
2. Copy the compiled file onto the scope, using a memory stick or network drive.
3. Open the Command Prompt window (Start ➔ Programs ➔ Accessories ➔ Command Prompt) and run your application.

Setup - Case 2

This scenario assumes you have Visual C++ loaded on the scope.

- Use the Visual C++ editor to develop and run your application.

Setup - Case 3

This scenario assumes you are using a compiler other than Visual C++ (such as GNU's MinGW). It should be noted that the optimizer in the GNU C Compiler (GCC) is less efficient than that in Visual C++ and results in lower performance.

1. Save your application in a text file, and copy it onto memory stick or network drive.
2. Using Windows Explorer, copy the text file to the scope.
3. Download and install the compiler onto the scope.
4. Configure Environmental Variables as follows:
   A. Open Start ➔ Settings ➔ Control Panel ➔ System.
   B. Click the "Advanced" tab, then the **Environmental Variables** button.
   C. In the "System variables" window, click **Path**, then the **Edit** button.
   D. At the end of the "Variable Value" string, append **;C:\MinGW\bin** for the case of the GNU C Compiler (GCC) for example.
   E. Click OK.
5. Open the Command Prompt window (Start ➔ Programs ➔ Accessories ➔ Command Prompt) and compile your application.
6. Run your application.

Operational Notes

Once FastWavePort is selected, the right-hand dialog shows the current settings. The first of these is critical, and indicates the base name of the memory window and the two events, which are global within the Windows O/S. This should be left at its default value and only changed if multiple FastWavePort functions are used in parallel. Note that this name must match the base name used in the client application.

The full names of these global objects are:

Memory Mapped File	"FastWavePort1File"
Data Available Event	"FastWavePort1MutexDataAvailable"
Processing Complete Event	"FastWavePort1MutexProcessingComplete"

The "Timeout" field specifies the amount of time that the DSO will wait for the custom processing function to complete. This prevents the DSO from waiting indefinitely for a potentially unforthcoming custom processing function. Be careful to set this value to something reasonable, which means a time that is longer by a reasonable margin than the custom processing is ever expected to take.

Data Length Limitations

The size of the memory window is fixed at 80 Mbytes, which equates to 40 MSamples.

Performance

This is by far the fastest way to process data using a user-defined algorithm on an X-Stream DSO.

Choice of Programming Language

The system was designed for use with the C/C++ programming language. However, it is theoretically possible for the processing to be implemented in any language that supports Windows named events (Mutex) and can open a named memory-mapped file. No guarantee can be given, however, as to the behaviour of the system using anything but C/C++.

Example Application

This simple C++ application may be used as a starting point for a custom processing function. It demonstrates the following:

- How to create handles to the global objects (the memory window and the two events)
- How to read data from the memory window when the DSO flags that it's available
- How to scale the data into units of volts using data in the header that's stored at the beginning of the memory window.
- How to perform a simple processing function (in this case the absolute value)
- How to define the physical units of the output of the processing function (in this case 'Amps')
- How to flag to the DSO application that processing is complete

```
//-------------------------------------------------------------------------------------
// FastWavePortClient.cpp :
//
// Prototype C++ client application for "Fast Wave Port' Math Processor
//
// Compatibility:
//      Microsoft Visual C++ 6.0, 7.1
//      MinGW 'gcc' based compiler (free download from www.mingw.org)
//          Compile with: mingw32-c++ -o fastWavePortClient.exe fastwaveportclient.cpp
//-------------------------------------------------------------------------------------
#include "windows.h"
#include <stdio.h>
//-------------------------------------------------------------------------------------
// FastWavePort header, describes various properties of the waveform passed to the user-
processing
// function. Also used to carry the properties of the processed waveform back to the DSO.
//
#define FLAGS_OUTPUT_VALID      0x01
typedef unsigned __int64 lecTimeStamp;
lecTimeStamp lecTimeStampOneSecond = 1000000000;        // 1 ns units in a second
#pragma pack(push, 4)           // pack on 4-byte boundaries (Important!)
struct CDescHeader
{
   int descVersion;            // header version number
   int flags;                 // misc. flags indicating the status of input, and how to treat the output
   int headerSize;             // size of the header, data starts immediately after the hdr.
   int windowSize;             // total size of the window (header + data)
   int numSamples;               // total number of samples in the input waveform
   int segmentIndex;            // index of this segment, usually zero when input waveform is not a
sequence
   int numSweeps;
   int _dummy1;                // not used
   double verGain;              // scale factor that relates integer sample data values to the vertical
units of the waveform.
   double verOffset;            // vertical offset (in vertical units, e.g. Volts) of the waveform
   double verResolution;        // vertical resolution of the measurement system (also in vertical
units)
   double horInterval;          // scale factor that relates integer sample indices to the horizontal units
of the waveform.
   double horOffset;            // horizontal offset (in horizontal units, e.g. seconds) of the waveform
   double horResolution;        // horizontal resolution of the measurement system (also in
horizontal units)
   lecTimeStamp trigTime;       // trigger time, units of 1ns since 00:00:00 on Jan 1st 2000, 64-bit
unsigned integer
   char verUnit[48];            // vertical units of the waveform ("V" for example)
   char horUnit[48];            // horizontal units of the waveform ("s" for example)
```

```
};
#pragma pack(pop)              // restore packing

//---------------------------------------------------------------------------------------
// The buffer size is 80MB (40,000,000 samples, stored as short integers) plus 0x1000 bytes for
the header.
const unsigned long HEADER_SIZE = 0x1000
const unsigned long MEM_MAP_FILE_SIZE = 80000000 + HEADER_SIZE;     // = 40MSamples,
or 80MBytes
int main(int argc, char* argv[])
{
   // names based on 'FastWavePort1' name defined in Processor setup.
   char szMapFileName[]                  = "FastWavePort1File";
   char szMutexDataAvailableName[]       = "FastWavePort1MutexDataAvailable";
   char szMutexProcessingCompleteName[]  = "FastWavePort1MutexProcessingComplete";
   // Associate shared memory file handle value.
      HANDLE m_hMMFile = CreateFileMapping ((HANDLE)0xffffffff, NULL,
PAGE_READWRITE, 0,  MEM_MAP_FILE_SIZE, szMapFileName);
      if(m_hMMFile == 0)
      {
         printf("Unable to create file mapping\n");
         return 0;
      }

   // Map a view of this file for writing.
      short *m_lpMMFile = (short *)MapViewOfFile (m_hMMFile, FILE_MAP_ALL_ACCESS, 0, 0,
0);
      if(m_lpMMFile == 0)
      {
         printf("Unable to map view of file\n");
         return 0;
      }
   // create/open events used for synchronization
   // if the client app. was run before the scope then these events will be created, if the scope was
run first then these events
   // will just be opened
      HANDLE m_hDataAvailable = CreateEvent(NULL, FALSE, FALSE /* initial state */,
szMutexDataAvailableName);
      HANDLE m_hProcessingComplete = CreateEvent(NULL, FALSE, FALSE /* initial state */,
szMutexProcessingCompleteName);
      if(m_hDataAvailable == 0 || m_hProcessingComplete == 0)
      {
         printf("Unable to open events\n");
         return 0;
      }
   // main loop
```

```
      while(1)
      {
         int i = 0;
         printf("Waiting for new data...\n");
         // wait an infinite amount of time for data to be available
            DWORD waitSuccess = WaitForSingleObject(m_hDataAvailable, INFINITE);
         // print the first few bytes of the input waveform
            CDescHeader *descHeader = (CDescHeader *) &m_lpMMFile[0];
            short *m_lpWaveform = &m_lpMMFile[descHeader->headerSize / sizeof(short)];
            for(i = 0; i < 4; ++i)
               printf("%f ", (m_lpWaveform[i] * descHeader->verGain) + descHeader->verOffset);
         // compute the mean of all data values, while computing the abs value of the waveform in-
place
            double sum = 0.0;
            for(i = 0; i < descHeader->numSamples; ++i)
            {
               sum += (m_lpWaveform[i] * descHeader->verGain) + descHeader->verOffset;
               m_lpWaveform[i] = abs(m_lpWaveform[i]);
            }
            sum /= descHeader->numSamples;
         // modify the output units, set to Amps
            strcpy(descHeader->verUnit, "A");
         // print the mean, numer of samples, trigger time in seconds, and the segment index
            printf("    (%f) %d %d %d\n", sum, descHeader->numSamples, (int) (descHeader-
>trigTime / lecTimeStampOneSecond), descHeader->segmentIndex);
         // use to flag that the output is not valid, increasing performance when
         // it is not necessary to read data back into the DSO
            //descHeader->flags &= ~FLAGS_OUTPUT_VALID;
         // flag that processing is complete
            SetEvent(m_hProcessingComplete);
      }
   return 0;
}
```

Header Description

int descVersion; // header version number
int flags; // misc. flags indicating the status of input, and how to treat the output
int headerSize; // size of the header, data starts immediately after the hdr.
int windowSize; // total size of the window (header + data)
int numSamples; // total number of samples in the input waveform
int segmentIndex; // index of this segment, usually zero when input waveform is not a sequence
int numSweeps;
int _dummy1; // not used
double verGain; // scale factor that relates integer sample data values to the vertical units of the waveform.
double verOffset; // vertical offset (in vertical units, e.g. Volts) of the waveform

double verResolution; // vertical resolution of the measurement system (also in vertical units)
double horInterval; // scale factor that relates integer sample indices to the horizontal units of the waveform.
double horOffset; // horizontal offset (in horizontal units, e.g. seconds) of the waveform
double horResolution; // horizontal resolution of the measurement system (also in horizontal units)
lecTimeStamp trigTime; // trigger time, units of 1ns since 00:00:00 on Jan 1st 2000, 64-bit unsigned integer
char verUnit[48]; // vertical units of the waveform ("V" for example)
char horUnit[48]; // horizontal units of the waveform ("s" for example)

Data Length Limitations

The size of the memory window is fixed at 80 Mbytes, which equates to 40M samples.

Performance

Under optimal conditions, on a DSO with a 1.7 GHz Celeron processor, rates of up to 75 MS/s have been observed. Due to the differences between the acquisition and processing hardware in each of the X-Stream DSOs, this value may vary significantly and therefore cannot be guaranteed. However, this is by far the fastest way to process data using a user-defined algorithm on an X-Stream DSO.

Choice of Programming Language

The system was designed for use with the C/C++ programming language, and all furnished examples use this language. It is theoretically possible, however, for the processing to be implemented in any language that supports Windows named events (Mutex) and can open a named memory-mapped file. Nevertheless, no guarantee can be given as to the behavior of the system using anything but C/C++.

FFT

Why Use FFT?

For a large class of signals, you can gain greater insight by looking at spectral representation rather than time description. Signals encountered in the frequency response of amplifiers, oscillator phase noise and those in mechanical vibration analysis, for example, are easier to observe in the frequency domain.
If sampling is done at a rate fast enough to faithfully approximate the original waveform (usually five times the highest frequency component in the signal), the resulting discrete data series will uniquely describe the analog signal. This is of particular value when dealing with transient signals because, unlike FFT, conventional swept spectrum analyzers cannot handle them.
Spectral analysis theory assumes that the signal for transformation is of infinite duration. Since no physical signal can meet this condition, a useful assumption for reconciling theory and practice is to view the signal as consisting of an infinite series of replicas of itself. These replicas are multiplied by a rectangular window (the display grid) that is zero outside of the observation grid.

An FFT operation on an N-point time domain signal can be compared to passing the signal through a comb filter consisting of a bank of N/2 filters. All the filters have the same shape and width and are centered at N/2 discrete frequencies. Each filter collects the signal energy that falls into the immediate neighborhood of its center frequency. Thus it can be said that there are N/2 frequency bins. The distance in Hz between the center frequencies of two neighboring bins is always the same: Delta f.

Power (Density) Spectrum

Because of the linear scale used to show magnitudes, lower amplitude components are often hidden by larger components. In addition to the functions offering magnitude and phase representations, the FFT option offers power density and power spectrum density functions. These latter functions are even better suited for characterizing spectra. The power spectrum (V^2) is the square of the magnitude spectrum (0 dBm corresponds to voltage equivalent to 1 mW into 50 Ω). This is the representation of choice for signals containing isolated peaks — periodic signals, for instance.
The power density spectrum (V^2/Hz) is the power spectrum divided by the equivalent noise bandwidth of the filter associated with the FFT calculation. This is best employed for characterizing broadband signals such as noise.

Memory for FFT

The amount of acquisition memory available will determine the maximum range (Nyquist frequency) over which signal components can be observed. Consider the problem of determining the length of the observation window and the size of the acquisition buffer if a Nyquist rate of 500 MHz and a resolution of 10 kHz are required. To obtain a resolution of 10 kHz, the acquisition time must be at least:

T = 1/Delta f = 1/10 kHz = 100 ms

For a digital oscilloscope with a memory of 100 kB, the highest frequency that can be analyzed is:

Delta f x N/2 = 10 kHz x 100 kB/2 = 500 MHz

FFT Pitfalls to Avoid

Take care to ensure that signals are correctly acquired: improper waveform positioning within the observation window produces a distorted spectrum. The most common distortions can be traced to insufficient sampling, edge discontinuities, windowing or the "picket fence" effect.
Because the FFT acts like a bank of band-pass filters centered at multiples of the frequency resolution, components that are not exact multiples of that frequency will fall within two consecutive filters. This results in an attenuation of the true amplitude of these components.

Picket Fence and Scallop

The highest point in the spectrum can be 3.92 dB lower when the source frequency is halfway between two discrete frequencies. This variation in spectrum magnitude is the picket fence effect. The corresponding attenuation loss is referred to as scallop loss. LeCroy scopes automatically correct for the scallop effect, ensuring that the magnitude of the spectra lines correspond to their true values in the time domain.

If a signal contains a frequency component above Nyquist, the spectrum will be aliased, meaning that the frequencies will be folded back and spurious. Spotting aliased frequencies is often difficult, as the aliases may ride on top of real harmonics. A simple way of checking is to modify the sample rate and observe whether the frequency distribution changes.

Leakage

FFT assumes that the signal contained within the time grid is replicated endlessly outside the observation window. Therefore if the signal contains discontinuities at its edges, pseudo-frequencies will appear in the spectral domain, distorting the real spectrum. When the start and end phase of the signal differ, the signal frequency falls within two frequency cells, broadening the spectrum.
The broadening of the base, stretching out in many neighboring bins, is termed leakage. Some cures for this problem include ensuring an integral number of periods are contained within the display grid or that no discontinuities appear at the edges. Another is to use a window function to smooth the edges of the signal.

Choosing a Window

The choice of a spectral window is dictated by the signal's characteristics. Weighting functions control the filter response shape, and affect noise bandwidth as well as side lobe levels. Ideally, the main lobe should be as narrow and flat as possible to effectively discriminate all spectral components, while all side lobes should be infinitely attenuated. The window type defines the bandwidth and shape of the equivalent filter to be used in the FFT processing.
In the same way as one would choose a particular camera lens for taking a picture, some experimenting is generally necessary to determine which window is most suitable. However, the following general guidelines should help.
Rectangular windows provide the highest frequency resolution and are thus useful for estimating the type of harmonics present in the signal. Because the rectangular window decays as a (sinx)/x function in the spectral domain, slight attenuation will be induced. Alternative functions with less attenuation (Flat Top and Blackman-Harris) provide maximum amplitude at the expense of frequency resolution; whereas, Hamming and Von Hann are good for general purpose use with continuous waveforms.

FFT Window Filter Parameters				
Window Type	**Highest Side Lobe (dB)**	**Scallop Loss (dB)**	**ENBW (bins)**	**Coherent Gain (dB)**
Rectangular	-13	3.92	1.0	0.0
von Hann	-32	1.42	1.5	-6.02
Hamming	-43	1.78	1.37	-5.35
Flat Top	-44	0.01	2.96	-11.05
Blackman-Harris	-67	1.13	1.71	-7.53

Window Type	Applications and Limitations
Rectangular	These are normally used when the signal is transient (completely contained in the time-domain window) or known to have a fundamental frequency component that is an integer multiple of the fundamental frequency of the window. Signals other than these types will show varying amounts of spectral leakage and scallop loss, which can be corrected by selecting another type of window.
Hanning (Von Hann)	These reduce leakage and improve amplitude accuracy. However, frequency resolution is also reduced.
Hamming	These reduce leakage and improve amplitude accuracy. However, frequency resolution is also reduced.
Flat Top	This window provides excellent amplitude accuracy with moderate reduction of leakage, but with reduced frequency resolution.
Blackman–Harris	It reduces the leakage to a minimum, but with reduced frequency resolution.

Improving Dynamic Range

Enhanced resolution uses a low-pass filtering technique that can potentially provide for three additional bits (18 dB) if the signal noise is uniformly distributed (white). Low-pass filtering should be considered when high frequency components are irrelevant. A distinct advantage of this technique is that it works for both repetitive and transient signals. The SNR increase is conditioned by the cut-off frequency of the ERES low-pass filter and the noise shape (frequency distribution).
LeCroy digital oscilloscopes employ FIR digital filters so that a constant phase shift is maintained. The phase information is therefore not distorted by the filtering action.

Record Length

Because of its versatility, FFT analysis has become a popular analysis tool. However, some care must be taken with it. In most instances, incorrect positioning of the signal within the display grid will significantly alter the spectrum. Effects such as leakage and aliasing that distort the spectrum must be understood if meaningful conclusions are to be arrived at when using FFT.
An effective way to reduce these effects is to maximize the acquisition record length. Record length directly conditions the effective sampling rate of the scope and therefore determines the frequency resolution and span at which spectral analysis can be carried out.

FFT Algorithms

A summary of the algorithms used in the oscilloscope's FFT computation is given here in a few steps:

1. The data are multiplied by the selected window function.
2. FFT is computed, using a fast implementation of the DFT (Discrete Fourier Transform):

$$X_n = \frac{1}{N} \sum_{k=0}^{k=N-1} x_k \times W^{nk}$$

Where: x_k is a complex array whose real part is the modified source time domain waveform, and whose imaginary part is 0; X_n is the resulting complex frequency-domain waveform; $W = e^{-2\pi j/N}$; and N is the number of points in x_k and X_n.

The generalized FFT algorithm, as implemented here, works on N, which need *not* be a power of 2.

3. The resulting complex vector X_n is divided by the coherent gain of the window function, in order to compensate for the loss of the signal energy due to windowing. This compensation provides accurate amplitude values for isolated spectrum peaks.
4. The real part of X_n is symmetric around the Nyquist frequency, that is

$R_n = R_{N-n}$

While the imaginary part is asymmetric, that is

$I_n = -I_{N-n}$

The energy of the signal at a frequency n is distributed equally between the first and the second halves of the spectrum; the energy at frequency 0 is completely contained in the 0 term.

The first half of the spectrum (Re, Im), from 0 to the Nyquist frequency is kept for further processing and doubled in amplitude:

$R'_n = 2 \times R_n \quad 0 < n < N/2$

$I'_n = 2 \times I_n \quad 0 < n < N/2$

5. The resultant waveform is computed for the spectrum type selected.

If "Magnitude" is selected, the magnitude of the complex vector is computed as:

$$M_n = \sqrt{R'^2_n + I'^2_n}$$

Steps 1–5 (previous) lead to the following result:

An AC sine wave of amplitude 1.0 V with an integral number of periods N_p in the time window, transformed with the rectangular window, results in a fundamental peak of 1.0 V magnitude in the spectrum at frequency N_p x Delta f. However, a DC component of 1.0 V, transformed with the rectangular window, results in a peak of 2.0 V magnitude at 0 Hz.

The waveforms for the other available spectrum types are computed as follows:

Phase: angle = arctan (I_n/R_n)_$M_n > M_{min}$

angle = 0 M_n M_{min}

Where M_{min} is the minimum magnitude, fixed at about 0.001 of the full scale at any gain setting, below which the angle is not well defined.

The dBm Power Spectrum:

$$dBm\ PS = 10 \times \log_{10}\left(\frac{M_n^2}{M_{ref}^2}\right) = 20 \times \log_{10}\left(\frac{M_n}{M_{ref}}\right)$$

where M_{ref} = 0.316 V (that is, 0 dBm is defined as a sine wave of 0.316 V peak or 0.224 V rms, giving 1.0 mW into 50 Ω).

The dBm Power Spectrum is the same as dBm Magnitude, as suggested in the above formula.

dBm Power Density:

$$dBm\,PD = dBm\,PS - 10 \times \log_{10}\left(ENBW \times \Delta f\right)$$

where *ENBW* is the equivalent noise bandwidth of the filter corresponding to the selected window, and *Delta f* is the current frequency resolution (bin width).

The FFT Power Average takes the complex frequency-domain data R'_n and I'_n for each spectrum generated in Step 5, and computes the square of the magnitude:

$M_n^2 = R'^2_n + I'^2_n$,

Then sums M_n2 and counts the accumulated spectra. The total is normalized by the number of spectra and converted to the selected result type using the same formulas as are used for the Fourier Transform.

Glossary

This section defines the terms frequently used in FFT spectrum analysis and relates them to the oscilloscope.

- **Aliasing -** If the input signal to a sampling acquisition system contains components whose frequency is greater than the Nyquist frequency (half the sampling frequency), there will be less than two samples per signal period. The result is that the contribution of these components to the sampled waveform is indistinguishable from that of components below the Nyquist frequency. This is **aliasing**.
  The timebase and transform size should be selected so that the resulting Nyquist frequency is higher than the highest significant component in the time-domain record.
- **Coherent Gain -** The normalized coherent gain of a filter corresponding to each window function is 1.0 (0 dB) for a rectangular window and less than 1.0 for other windows. It defines the loss of signal energy due to the multiplication by the window function. This loss is compensated for in the oscilloscope. The following table lists the values for the implemented windows.

Window Frequency Domain Parameters				
Window Type	Highest Side Lobe (dB)	Scallop Loss (dB)	ENBW (bins)	Coherent Gain (dB)
Rectangular	-13	3.92	1.0	0.0
Hanning (Von Hann)	-32	1.42	1.5	-6.02
Hamming	-43	1.78	1.37	-5.35
Flattop	-44	0.01	2.96	-11.05
Blackman–Harris	-67	1.13	1.71	-7.53

- **ENBW -** Equivalent Noise BandWidth (ENBW) is the bandwidth of a rectangular filter (same gain at the center frequency), equivalent to a filter associated with each frequency bin, which would collect the same power from a white noise signal. In the table on the previous page, the ENBW is listed for each window function implemented, given in bins.
- **Filters -** Computing an N-point FFT is equivalent to passing the time-domain input signal through N/2 filters and plotting their outputs against the frequency. The spacing of filters is Delta f = 1/T, while the bandwidth depends on the window function used (see Frequency Bins).
- **Frequency Bins -** The FFT algorithm takes a discrete source waveform, defined over N points, and computes N complex Fourier coefficients, which are interpreted as harmonic components of the input signal.
  For a real source waveform (imaginary part equals 0), there are only N/2 independent harmonic components.
  An FFT corresponds to analyzing the input signal with a bank of N/2 filters, all having the same shape and width, and centered at N/2 discrete frequencies. Each filter collects the signal energy that falls into the immediate neighborhood of its center frequency. Thus it can be said that there are N/2 "frequency bins."
  The distance in hertz between the center frequencies of two neighboring bins is always:
  Delta f = 1/T
  Where T is the duration of the time-domain record in seconds.
  The width of the main lobe of the filter centered at each bin depends on the window function used. The rectangular window has a nominal width at 1.0 bin. Other windows have wider main lobes (see table).
- **Frequency Range -** The range of frequencies computed and displayed is 0 Hz (displayed at the left-hand edge of the screen) to the Nyquist frequency (at the rightmost edge of the trace).
- **Frequency Resolution -** In a simple sense, the frequency resolution is equal to the bin width Delta f. That is, if the input signal changes its frequency by Delta f, the corresponding spectrum peak will be displaced by Df. For smaller changes of frequency, only the shape of the peak will change.
  However, the effective frequency resolution (that is, the ability to resolve two signals whose frequencies are almost the same) is further limited by the use of window functions. The ENBW value of all windows other than the rectangular is greater than Delta f and the bin

width. The table of Window Frequency-Domain Parameters lists the ENBW values for the implemented windows.

- **Leakage -** In the power spectrum of a sine wave with an integral number of periods in the (rectangular) time window (that is, the source frequency equals one of the bin frequencies), the spectrum contains a sharp component whose value accurately reflects the source waveform's amplitude. For intermediate input frequencies this spectral component has a lower and broader peak.
  The broadening of the base of the peak, stretching out into many neighboring bins, is termed *leakage*. It is due to the relatively high side lobes of the filter associated with each frequency bin.
  The filter side lobes and the resulting leakage are reduced when one of the available window functions is applied. The best reduction is provided by the Blackman–Harris and Flattop windows. However, this reduction is offset by a broadening of the main lobe of the filter.
- **Number of Points -** The FFT is computed over the number of points (Transform Size) whose upper bounds are the source number of points, and by the maximum number of points selected in the menu. The FFT generates spectra of N/2 output points.
- **Nyquist Frequency -** The Nyquist frequency is equal to one half of the effective sampling frequency (after the decimation): Delta f x N/2.
- **Picket Fence Effect -** If a sine wave has a whole number of periods in the time domain record, the power spectrum obtained with a rectangular window will have a sharp peak, corresponding exactly to the frequency and amplitude of the sine wave. Otherwise the spectrum peak with a rectangular window will be lower and broader.
  The highest point in the power spectrum can be 3.92 dB lower (1.57 times) when the source frequency is halfway between two discrete bin frequencies. This variation of the spectrum magnitude is called the *picket fence effect* (the loss is called the scallop loss).
  All window functions compensate for this loss to some extent, but the best compensation is obtained with the Flattop window.
- **Power Spectrum -** The power spectrum (V^2) is the square of the magnitude spectrum.

  The power spectrum is displayed on the dBm scale, with 0 dBm corresponding to:
  $V_{ref}^2 = (0.316\ V_{peak})^2$,
  Where V_{ref} is the peak value of the sinusoidal voltage, which is equivalent to 1 mW into 50 Ω.
- **Power Density Spectrum -** The power density spectrum (V^2/Hz) is the power spectrum divided by the equivalent noise bandwidth of the filter, in hertz. The power density spectrum is displayed on the dBm scale, with 0 dBm corresponding to (V_{ref}^2/Hz).
- **Sampling Frequency -** The time-domain records are acquired at sampling frequencies dependent on the selected time base. Before the FFT computation, the time-domain record may be decimated. If the selected maximum number of points is lower than the source number of points, the effective sampling frequency is reduced. The effective sampling frequency equals twice the Nyquist frequency.
- **Scallop Loss -** This is loss associated with the picket fence effect.

- **Window Functions -** All available window functions belong to the sum of cosines family with one to three non-zero cosine terms:

$$W_k = \sum_{m=0}^{m=M-1} a_m \cos\left(\frac{2\pi k}{N} m\right) \qquad 0 \le k < N$$

where: $M = 3$ is the maximum number of terms, a_m are the coefficients of the terms, N is the number of points of the decimated source waveform, and k is the time index.

The table of Coefficients of Window Functions lists the coefficients a_m. The window functions seen in the time domain are symmetric around the point k = N/2.

Coefficients of Window Functions			
Window Type	a_0	a_1	a_2
Rectangular	1.0	0.0	0.0
Hanning (Von Hann)	0.5	-0.5	0.0
Hamming	0.54	-0.46	0.0
Flattop	0.281	-0.521	0.198
Blackman–Harris	0.423	-0.497	0.079

FFT Setup

Setting Up an FFT

1. In the menu bar touch **Math**, then **Math Setup...** in the drop-down menu.
2. Touch a Math function trace button: **F1** through **Fx** [The number of math traces available depends on the software options loaded on your scope. See Specifications.]; a pop-up menu appears. Select **FFT**  from the menu.
3. Touch the **Single**  or **Dual** (function of a function)  button if the FFT is to be of the result of another math operation.
4. Touch inside the **Source1** field and select a channel, memory, or math trace on which to perform the FFT.
5. Touch inside the **Operator1** field: Select **FFT** from the pop-up menu if you selected **Single** function. Select another math function if you selected **Dual** function. Then touch inside the **Operator2** field and select **FFT** from the pop-up menu.
6. In the right-hand dialog, touch the **FFT** tab.

7. Choose whether to **Truncate**[1] 

or **Zero-fill**[2] 

the trace display.
8. Touch the **Suppress DC** checkbox if you want to make the DC bin go to zero. Otherwise, leave it unchecked.
9. Touch inside the **Output** type field, and make a selection from the pop-up menu.
10. Touch inside the **Window** field, select a window type.
11. Touch inside the **Algorithm** field and select either **Least Prime**[3] or **Power of 2**[4] from the pop-up menu.

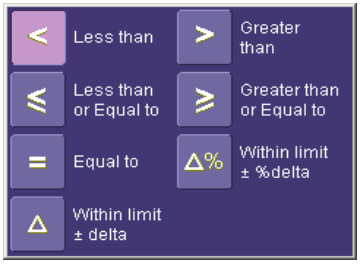
ANALYSIS

Pass/Fail Testing

Comparing Parameters

Each Pass/Fail input (**Qx**) can compare a different parameter result to a user-defined limit (or statistical range) under a different condition.

The conditions are represented by these comparison operators:



At the touch of a button, test results can also be compared to these standard statistical limits:

- current mean
- mean + 1 SD
- mean + 3 SD

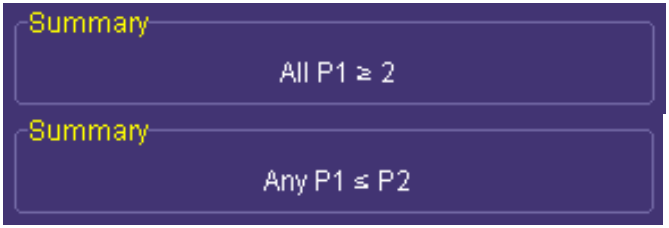
[1] When the FFT transform size does not match the record length, you can truncate the record and perform an FFT on the shorter record. This will increase the resolution bandwidth.

[2] Zero-fill is useful when the source data for the FFT comes from a math operation that shortens the record. This is commonly encountered in filtering operations like enhanced resolution. The missing data points are replaced by data values, whose amplitudes are interpolated to fit between the last data point and th first data point in the record. This guarantees that there is not a first-order discontinuity in the filled data. Since the data at the end of the record is "filled" data, it is advisable to select a weighting window other than rectangular to minimize the effect of the fill on the resulting spectrum.

[3] The default algorithm is a least primes algorithm that computes FFTs on transform sizes having lengths that can be expressed as factors of 2^N*5^K. This is very compatible with the record lengths encountered in the oscilloscope, which are often multiples of 1, 2, 4, 5, or 10.

[4] The other choice is a power of two algorithm where the record lengths are in the form of 2^N. The power of 2 algorithm generally runs faster than the least primes algorithm. The price that is paid is a record length that is not the same as the acquired signal. The power-of-two FFT uses the first 2^N points of the record. For example, if you acquire 500 points in your trace, the power-of-two FFT would only use the first 256 points.

In Dual Parameter Compare mode, your X-Stream scope gives you the option to compare to each other parameter results measured on two different waveforms. You can set your test to be true if Any waveform or All waveforms fit the criterion stipulated by the comparison condition. Your setup is conveniently shown in the Summary box of the **Qx** dialog. For example:



Mask Tests

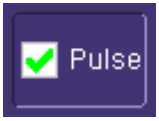
You have the choice to do mask testing by using an existing mask, or by using a mask created from your actual waveform, with vertical and horizontal tolerances that you define. Existing masks can be loaded from a floppy disk or from a network.
You can set your mask test to be True for waveforms All In, All Out, Any In, or Any Out. For example, if you select All In, the test will be False if even a single waveform falls outside the mask.
Masks that you create from your waveform can be confined to just a portion of the trace by use of a measure gate. (See Measure Gate for an explanation of how this feature works.)

Actions

By touching the **Stop Test** checkbox 

in the "Actions" dialog, you can set up the test to end after a predetermined number of sweeps that you decide.
You can also decide the actions to occur upon your waveforms' passing or failing, by selecting one or all of the following:

- stop
- audible alarm
- print image of display
- emit pulse
- save waveform

The selection **Pulse** 

causes a pulse to be output through the Aux Out connector at the front of the scope. This pulse can be used to trigger another scope. You can set the amplitude and width of the pulse as described in the Auxiliary Output Signals section of this manual.
Depending on your scope model, you can configure up to 8 pass/fail conditions.

The boolean conditions to determine if your waveform passes are as follows:

All True	All False
Any True	Any False
All Q1 to Q4 **Or** All Q5 to Q8	Any Q1 to Q4 **And** Any Q5 to Q8

Setting Up Pass/Fail Testing

Initial Setup

1. Touch **Analysis** in the menu bar, then **Pass/Fail Setup...** in the drop-down menu.
2. Touch the **Actions** tab.
3. Touch the **Enable Actions** checkbox. This will cause the actions that you will select to occur upon your waveform's passing or failing a test.
4. Touch the **Summary View** to enable a line of text that shows concisely the status of your last waveform and keeps a running count of how many sweeps have passed:



5. Touch inside the **Pass If** field, and select a boolean condition from the pop-up menu.
6. If you want to set up the test to end after a finite number of sweeps, touch the **Stop Test** checkbox. Then touch inside the **After** data entry field and enter a value, using the pop-up numeric keypad.
7. Under "If", touch either the **Pass** 

or **Fail** 

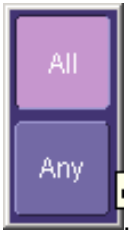
button to set the actions to occur upon your waveform's passing or failing the test.
8. Under "Then", touch the actions you want to occur: stop test, sound alarm, print result, emit pulse, or save the waveform. If you want to have the results printed and your scope is not equipped with a printer, be sure that the it is connected to a local or network printer. See Printing.
9. If you want to save your waveform automatically, touch the **Save Setup**. This will take you out of the current dialog and will open the "Save Waveform" dialog. See Saving and Recalling Waveforms.
10. Test your Pass/Fail conditions by touching the **Force Actions Once** button. Press the **Clear All** button to quickly uncheck all checkboxes if you want to change your selections.

Comparing a Single Parameter

1. Touch **Analysis** in the menu bar, then **Pass/Fail Setup...** in the drop-down menu.
2. Touch a **Qx** tab; a setup dialog for that position will open.
3. Touch inside the **Source1** field and select a source from the pop-up menu.
4. Touch inside the **Condition** field in the main dialog and select **ParamCompare** 

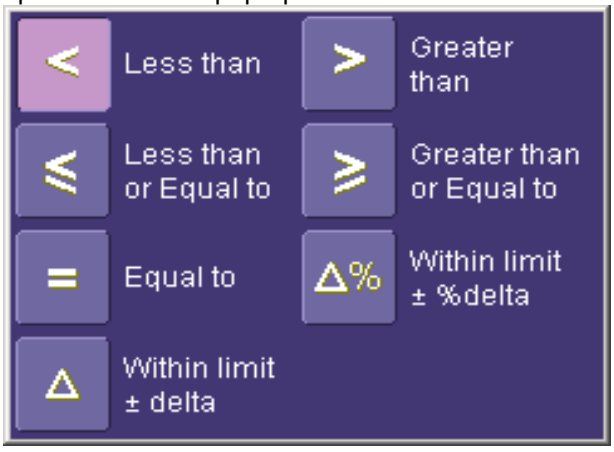
.

5. Touch inside the **Compare Values** field and select **All** or **Any** from the pop-up menu



By selecting **All**, the test is true only if every waveform falls within the set limit. By selecting **Any**, the test is be true if just one waveform falls within the limit.

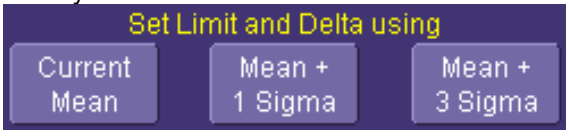
6. Touch inside the **Condition** field in the "ParamCompare" mini-dialog and select a math operator from the pop-up menu:



7. Touch inside the **Limit** field and enter a value, using the pop-up numeric keypad. This value takes the dimensions of the parameter that you are testing. For example, if you are testing a time parameter, the unit is seconds. If you chose either **WithinDeltaPct** 

or **WithinDeltaAbs** 

from the **Condition** menu, you also have the choice of setting the limit by means of the statistical buttons at the bottom of the "ParamCompare" dialog:



Comparing Dual Parameters

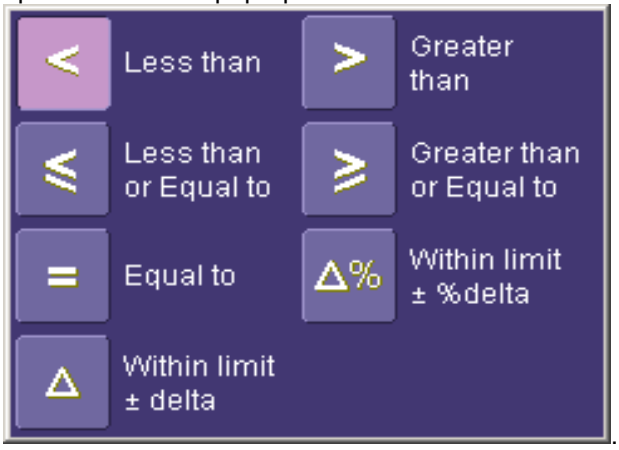
1. Touch **Analysis** in the menu bar, then **Pass/Fail Setup...** in the drop-down menu.
2. Touch a **Qx** tab; a setup dialog for that position will open.
3. Touch inside the **Condition** field in the main dialog and select **DualParamCompare** 

.
4. Touch inside the **Source1** and **Source2** fields and select a source from the pop-up menu.
5. Touch inside the "ParamCompare" mini-dialog field and select a source from the pop-up menu.
6. Touch inside the **Compare Values** field and select **All** or **Any** from the pop-up menu:



By selecting **All**, the test will be true only if every waveform falls within the limit that you will set. By selecting **Any**, the test will be true if just one waveform falls within the limit.

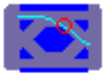
7. Touch inside the **Condition** field in the "ParamCompare" mini-dialog and select a math operator from the pop-up menu:



8. Touch inside the **Limit** field and enter a value, using the pop-up numeric keypad. This value takes the dimension of the parameter that you are testing. For example, if you are testing a time parameter, the unit is seconds.

9. If you chose either **WithinDeltaPct**  or **WithinDeltaAbs**  from the **Condition** menu, touch inside the **Delta** field and enter a value.

Mask Testing

1. Touch **Analysis** in the menu bar, then **Pass/Fail Setup...** in the drop-down menu.
2. Touch a **Qx** tab; a setup dialog for that position will open.
3. Touch inside the **Source1** field and select a source from the pop-up menu.
4. Touch inside the **Condition** field in the main dialog and select **Mask Test** .
5. From the "Test" mini-dialog, make a selection in the **Test is True when** group of buttons:



This selection means, for example, that if you select **All In** the test will be False if even a single waveform falls outside the mask.

6. From **Show Markers**, choose whether or not to have mask violations displayed.
7. If you are loading a pre-existing mask, touch the **Load Mask** tab, then the **File** button. You can then enter the file name or browse to its location.
8. If you want to make a mask from your waveform, touch the **Make Mask** tab.
9. Touch inside the **Ver Delta** and **Hor Delta** fields and enter boundary values, using the pop-up numeric keypad.
10. Touch the **Browse** button to create a file name and location for the mask if you want to save it.
11. Touch the **Gate** tab, then enter values in the **Start** and **Stop** fields to constrain the mask to a portion of the waveform. Or, you can simply touch and drag the Gate posts, which initially are placed at the extreme left and right ends of the grid.

UTILITIES

Status

The status read-only dialog displays system information including serial number, firmware version, and installed software and hardware options.

Status Dialog Access

1. In the menu bar, touch **Utilities**.
2. Touch the **Status** tab.

Remote communication

The Remote dialog is where you can select a network communication protocol, establish network connections, and configure the Remote Control Assistant log. The choice of communication protocols is limited to TCPIP and GPIB.

Please Note the Following:

- GPIB is an option and requires a GPIB card to be installed in a card slot at the rear of the scope.
- The instrument uses Dynamic Host Configuration Protocol (DHCP) as its addressing protocol. Therefore, it is not necessary to set up an IP address if your network supports DHCP. If it does not, you can assign a static address in the standard Windows 2000 network setup menu.

The Remote Control Assistant monitors communication between your PC and scope when you are operating the instrument remotely. You can log all events, or errors only. This log can be invaluable when you are creating and debugging remote control applications.

Remote Communication Setup

If you are connecting the scope to a network, first contact your Information Systems administrator. If you are connecting the scope directly to your PC, connect a GPIB or Ethernet cable between them.

1. In the menu bar touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
2. Touch the **Remote** tab.
3. Make a **Port** selection:**TCPIP** (transmission control protocol/Internet protocol) or **GPIB** (general purpose interface bus). If you do not have a GPIB card installed, the GPIB selection will not be accessible.
4. If you are using GPIB, set a GPIB address by touching inside the **GPIB Address** data entry field and enter an address.
5. Press the **Net Connections** button; the Windows **Network and Dial-up Connections** window appears.

6. Touch **Make New Connection** and use the Windows Network Connection Wizard to make a new connection; or, touch Local Area Connection to reconfigure the scope's connection if it is already connected to the network.

Configuring the Remote Control Assistant Event Log

1. In the menu bar touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
2. Touch the **Remote** tab.
3. Touch inside the **Log Mode** data entry field.
4. Select **Off**, **Errors Only**, or **Full Dialog** from the pop-up menu.
5. To export the contents of the event log to an ASCII text file, touch the **Show Remote Control Log** button: the "Event Logs" pop-up window appears. Touch inside the **DestFilename** data entry field and enter a file name, using the pop-up keyboard. Then touch the **Export to Text File** button.

Hardcopy

Printing

For print setup, refer to Printing.

Clipboard

This selection prints to the clipboard so you can paste a file into another application (like MS Word, for example).

Printing from the Clipboard

1. In the menu bar touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
2. Touch the **Hardcopy** tab.
3. Under **Colors**, touch the **Use Print Colors** checkbox if you want the traces printed on a white background. A white background saves printer toner.
4. Touch the **Grid Area Only** checkbox if you do not need to print the dialog area and you only want to show the waveforms and grids.
5. Touch the **Print Now** button.

File

Choose **File** if you want to output the screen image to storage media such as floppy drive or hard drive. When outputting to floppy disk, be sure to use a preformatted disk.

Printing to a File

1. In the menu bar touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
2. Touch the **Hardcopy** tab, then the **File** icon.
3. Touch inside the **File Format** data entry field and select a graphic file format from the pop-up menu.
4. Under **Colors**, touch the **Use Print Colors** checkbox if you want the traces printed on a white background. A white background saves printer toner.
5. Touch inside the **Directory** data entry field and type the path to the folder you want to print to, using the pop-up keyboard. Or touch the **Browse** button and navigate to the folder.

6. Touch inside the **File Name** data entry field and enter a name for the display image, using the pop-up keyboard.
7. Touch the **Grid Area Only** checkbox if you do not need to print the dialog area and you only want to show the waveforms and grids.
8. Touch the **Print Now** button.

E-Mail

The instrument also gives you the option to e-mail your screen images, using either the MAPI or SMTP protocols. Before you output to e-mail from the Utilities dialog, you first have to set up the e-mail server and recipient address in **Preference Setup**.

Sending E-mail

1. In the menu bar touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
2. Touch the **Hardcopy** tab, then the **E-mail** button.
3. Touch inside the **File Format** data entry field and select a graphic file format from the pop-up menu.
4. Under **Colors**, touch the **Use Print Colors** checkbox if you want the traces printed on a white background. A white background saves printer toner.
5. Touch the **Prompt for message to send with mail** checkbox if you want to include remarks with the image.
6. Touch inside the **Hardcopy Area** field and make a selection from the pop-up menu:
7. Touch the **Print Now** button:



Aux Output

Please refer to Auxiliary Output Signals section of this manual.

Date & Time

The instrument provides the choice of manually setting the time and date or getting it from the Internet. If you elect to get the time and date from the Internet, you need to have the scope connected to the Internet through the LAN connector on the rear panel. You can also set time zones and Daylight Savings time.

Setting the Time and Date Manually

1. In the menu bar touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
2. Touch the **Date/Time** tab.
3. Touch inside each of the **Hour**, **Minute**, **Second**, **Day**, **Month**, and **Year** data entry fields and enter a value, using the pop-up numeric keypad.
4. Touch the **Validate Changes** button.

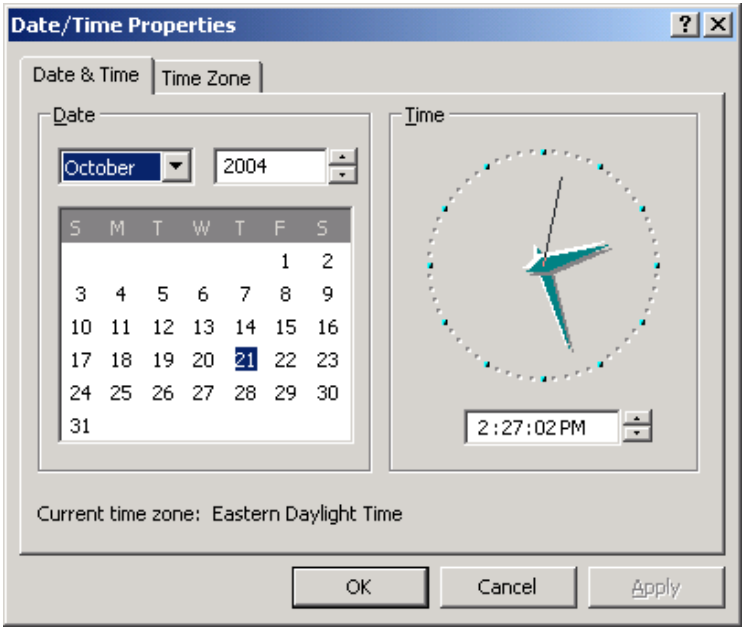
Setting the Time and Date from the Internet

The Simple Network Time Protocol (SNTP) is used.

1. Ensure that the scope is connected to the Internet through the LAN connector at the rear of the scope.
2. In the menu bar touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
3. Touch the **Date/Time** tab.
4. Touch the **Set from Internet** button.

Setting the Time and Date from Windows

1. In the menu bar touch **Utilities**, then **Utilities Setup...** in the drop-down menu.
2. Touch the **Date/Time** tab.
3. Touch the **Windows Date/Time** button  .
4. Use the **Time & Date Properties** window to configure the time, including time zone:



Options

Use this dialog to add or remove software options. For information about software options, contact your local LeCroy Sales and Service office, or visit our Web site at www.lecroy.com/options.
Options that you purchase, such as JTA2, add performance to you instrument. This added performance is seen in the new math functions or parameters that you can choose from when doing Measure or Math setups.

Preferences

Audible Feedback

You can elect to have audible confirmation each time you touch a screen or front panel control.

1. In the menu bar touch **Utilities**; then touch **Preferences** in the drop-down menu.
2. Touch the "Audible Feedback" **Enable** checkbox so that the scope emits a beep with each touch of the screen or front panel control.

Auto-calibration

You can choose to have your instrument automatically recalibrate itself whenever there is a significant change in ambient temperature. If you do not enable this option, the scope will only recalibrate at startup and whenever you make a change to certain operating conditions.

1. In the menu bar touch **Utilities**; then touch **Preferences** in the drop-down menu.
2. Touch the "Automatic Calibration" **Enable** checkbox.

Offset Control

As you change the gain, this control allows you to either keep the vertical offset level indicator stationary (when **Div** is selected) or to have it move with the actual voltage level (when **Volts** is selected). The advantage of selecting **Div** is that the waveform will remain on the grid as you increase the gain; whereas, if **Volts** is selected, the waveform could move off the grid.

Note: Regardless of whether you select **Volts** or **Div**, the "Offset" shown in the channel setup dialog always indicates volts. However, when **Div** is selected for the Offset Control, the offset in volts is scaled proportional to the change in gain, thereby keeping the division on the grid constant.

1. Touch **Utilities** ➔ **Preferences** on the drop-down menu bar.
2. Touch the **Acquisition** tab.
3. Under **Offset Setting constant in:**, touch either the **Div** or **Volts** button.

Delay Control

As you change the timebase, this control allows you to either keep the horizontal offset indicator stationary (when **Div** is selected) or to have it move with the trigger point (when **Time** is selected). The advantage of selecting **Div** is that the trigger point will remain on the grid as you increase the timebase; whereas, if **Time** is selected, the trigger point could move off the grid.

Note: Regardless of whether you select **Time** or **Div**, the "Delay" shown in the timebase setup dialog always indicates time. However, when **Div** is selected for Delay In, the delay in time is scaled proportional to the change in timebase, thereby keeping the division on the grid constant.

1. In the menu bar touch **Utilities**; then touch **Preferences** in the drop-down menu.
2. Touch the **Acquisition** tab.
3. Under **Delay Setting constant in:**, touch either the **Div** or **Volts** button.

Trigger Counter

Checking the **Reset trigger counter before starting a new acquisition** checkbox clears the trigger counter each time the scope issues an arm acquisition command. This applies when you have set a trigger Holdoff condition in the Trigger dialog in either time or events:



The default condition of this control is off (unchecked).

Performance Optimization

You can set up the scope to optimize either calculating speed or display speed. If the display update rate is of primary concern to you, optimize for Display. If acquisition and analysis are more important, optimize for analysis. Optimizing for analysis can be useful when persistence or averaging is used, giving higher priority to waveform acquisition at the expense of display update rate.
The choices are presented as a spectrum with highest values at the extremes:



1. In the menu bar touch **Utilities**; then touch **Preferences** in the drop-down menu.
2. Touch one of the optimization icons.

E-mail

Before you can send e-mail from the scope, it must first be configured.

1. In the menu bar touch **Utilities**, then **Preference Setup...** in the drop-down menu.
2. Touch the **E-mail** tab.

   Choose an e-mail server protocol: **MAPI** (Messaging Application Programming Interface) is the Microsoft interface specification that allows different messaging and workgroup applications (including e-mail, voice mail, and fax) to work through a single client, such as the Exchange client included with Windows 95 and Windows NT. MAPI uses the default Windows e-mail application (usually Outlook Express). **SMTP** (Simple Mail Transfer Protocol) is a TCP/IP protocol for sending messages from one computer to another through a network. This protocol is used on the Internet to route e-mail. In many cases no account is needed.
3. If you chose MAPI, touch inside the **Originator Address (From:)** data entry field and use the pop-up keyboard to type in the instrument's e-mail address. Then touch inside the **Default Recipient Address (To:)** data entry field and use the pop-up keyboard to enter the recipient's e-mail address.
4. If you chose SMTP, touch inside the **SMTP Server** data entry field and use the pop-up keyboard to enter the name of your server. Touch inside the **Originator Address (From:)** data entry field and use the pop-up keyboard to type in the instrument's e-mail address. Then touch inside the **Default Recipient Address (To:)** data entry field and use the pop-up keyboard to enter the recipient's e-mail address.
5. You can send a test e-mail text message by touching the **Send Test Mail** button. The test message reads "Test mail from [name of scope's email address]."

Acquisition Status

For each general category of scope operation, you can view a summary of your setups. These dialogs are not accessible through the Utilities menu, but are instead accessed from the drop-down menu bars. The categories are as follows:

- **Vertical** - select **Channels Status . . .**
- **Timebase** - select **Acquisition Status . . .**
- **Trigger** - select **Acquisition Status . . .**
- **Math** - select **Math Status . . .**

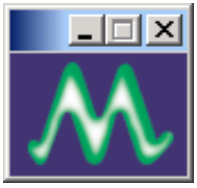
In addition to these dialogs, summaries are also provided for XY setups, memory (M1-M4) setups, and time stamps for sequence mode sampling.

Service

This button provides access to service dialogs, which are for the sole use of LeCroy service personnel. A security code is required to gain access.

Show Windows Desktop

Touching the **Show Windows Desktop** button  in the main "Utilities" dialog minimizes the instrument application to reveal the underlying desktop. To maximize the application, touch the appropriate shortcut icon:



Touch-Screen Calibration

Touching the **Touch-Screen Calibration** button  starts the calibration procedure. During the procedure, you will be prompted to touch the center of a small cross in 5 key locations on the touch-screen. Because sufficient accuracy cannot be achieved using your finger, use a stylus instead for this procedure. The calibration has a ten-second timeout in case no cross is touched. To avoid parallax errors, be sure to place your line of sight directly in front of each cross before touching it.

CUSTOMIZATION

Customizing Your Instrument

The instrument provides powerful capability to add your own parameters, functions, display algorithms, or other routines to the scope user interface without having to leave the instrument application environment. You can customize the instrument to your needs by using the power of programs such as Excel™, Mathcad™, and MATLAB™, or by scripting in VBS. Whichever method you use, the results appear on the instrument's display together with the signals that you started with. This ability offers tremendous advantages in solving unique problems for a large range of applications, with comparatively little effort from you.



Caution

Accessing the scope's automation interface from within an XDEV custom processor (VBScript, Mathcad, MATLAB, Excel, etc.) is NOT recommended.

Cases where the scope's behaviour cannot be guaranteed, or worse, cases which can cause the scope's software to crash include the following:

1. Changing "Upstream" Controls

'Upstream' controls are considered to be any control that, if changed, could provoke an infinite loop. An example would be a VBScript processor, in F1, which uses C1 as a source. If this processor changes the offset or vertical scale of C1 as a result of examining its input (C1) data, an infinite loop could occur. This could eventually cause the scope software to crash. This is not limited only to changing upstream channel controls, but includes any upstream processing also.

2. Accessing Other Results

Access to results (waveform, measurement, etc.), other than the scripting processor's own inputs, may cause incorrect measurements. The reason for this is simple: the scope contains a complex algorithm to determine in which order results are computed. This algorithm ensures that all inputs required by a processor are computed before the processor itself. If a VBScript processor decides to access, via automation, results other than those supplied to its inputs, the scope's dependency algorithm cannot be used. Therefore, the results accessed may not be coherent (they may be from a previous acquisition, or worse, could cause an infinite computation loop).

3. Reconfiguring Math or Measurements

Reconfiguring (adding or removing) Math and/or Measurements from within a custom processing function is not recommended, especially when the reconfiguration would cause the custom processor to remove itself.

Introduction

Instrument customization provides these important capabilities:

- You can export data to programs, without leaving the instrument environment.
- You can get results back from those programs, and display them on the instrument, without leaving the instrument application environment.
- Once the result is returned, you can perform additional scope operations, such as measuring with cursors, applying parameters, or performing additional functions on the waveform, in exactly the same way as for a normal waveform.
- You can program the scope yourself.

The instrument does not just provide connectivity with data downloads to other programs. It provides true customizable interaction with these other programs, and allows you to truly customize the scope to do the exact job you want it to do. The advantages to this are many:

- You can use the standard processing power of the instrument to do most of your calculations
- You only need to write the function, parameter, display algorithm, etc. that specifically applies to your need and that the instrument doesn't contain.
- You can view the final result on the instrument display, and use all of the instrument's tools to understand the result.
- You can do additional processing on the result by applying either standard instrument parameters, functions, etc. to the returned result, or even more powerfully, adding chained customized functions. For example, you can do an Excel calculation on a result with a MATLAB function applied to it.

Solutions

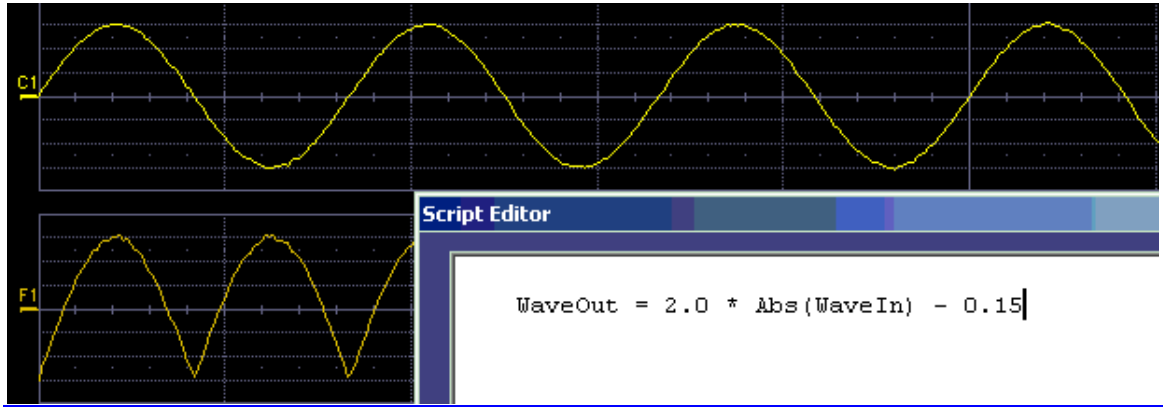
Engineers do not buy equipment; they buy solutions. But what solutions can be reached from a set of instrument waveform data? The answer is, in principle, anything that can be logically derived from those data, given the limitations of signal-to-noise ratio and processing time. Here are some examples of what can be done with a customized instrument:

- Changing the units of a grid to joules, newtons, amps, etc.
- Creating a new waveform by manipulating the data of one or two input waveforms
- Creating a new waveform without using any of the input data
- Creating a new parameter by manipulating the data of one or two input waveforms
- Changing a vertical scale or a horizontal scale from linear to non-linear

You don't have to use all the data from the input waveforms: you can select data from one or more segments, which need not be aligned in the two-input waveforms.

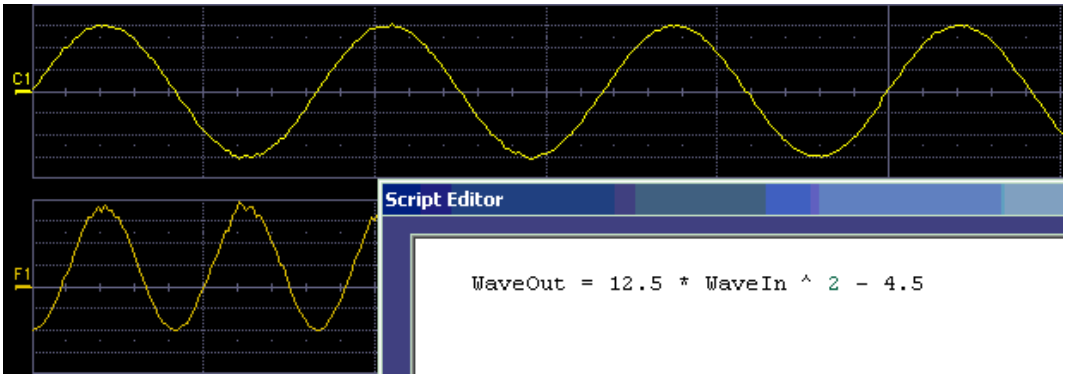
Examples

Example 1: Simple math functions using VBScript



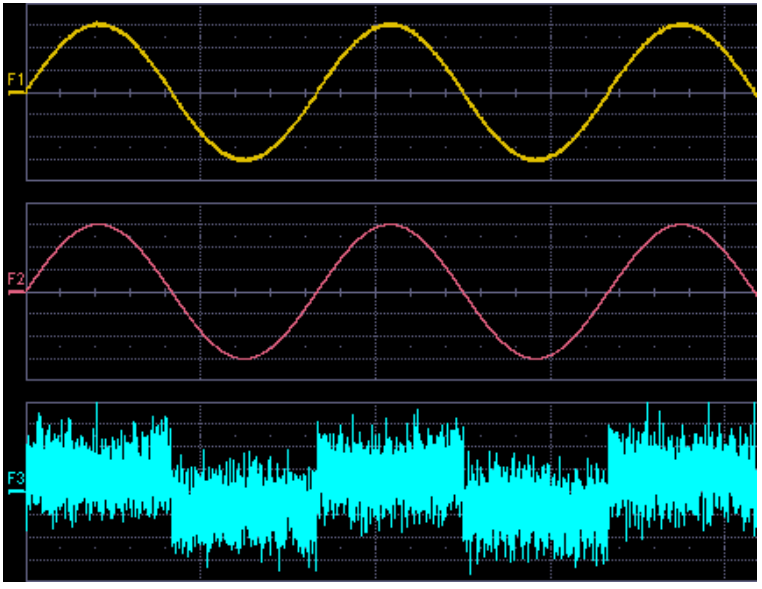
WaveOut is the waveform being returned to the instrument (F1 in this case). WaveIn is the input waveform (C1 in this case). You can see that the F1 result is displayed on the scope, and can be processed further.

Example 2: Another simple math functions using VBScript



Example 3 (as follows) doesn't use the input data at all. The middle waveform (F2) is a "golden waveform", in this case a perfect sine (subject to 16-bit resolution), that was created using a VBScript. The lower trace (F3) is a subtraction of the acquired waveform (upper trace) and the golden waveform. The subtraction (of course) contains all the noise, but it also shows the presence of a very small square wave signal.

Example 3

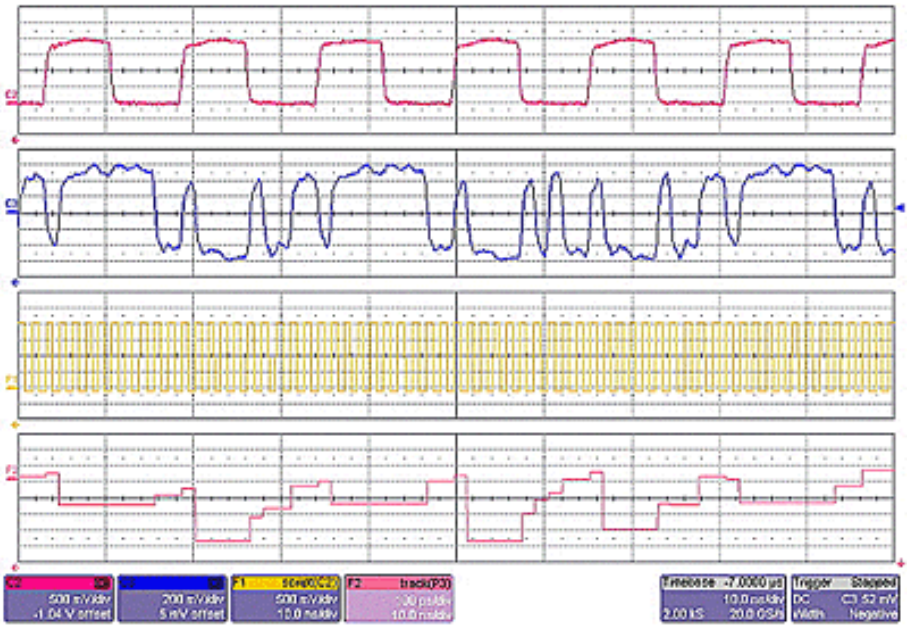


Here is the VBScript that produced the "golden sine" (F2 previous):

```
Frequency = 3000000.0   ' Frequency of real data
SampleTime = InResult.HorizontalPerStep
Omega = 2.0 * 3.1416 * Frequency * SampleTime
Amplitude = 0.15        ' Amplitude of real data
    For K = 0 To LastPoint
    newDataArray(K) = Amplitude * Sin(Omega * K)
    Next
OutResult.DataArray(True) = newDataArray ' Data in volts
```

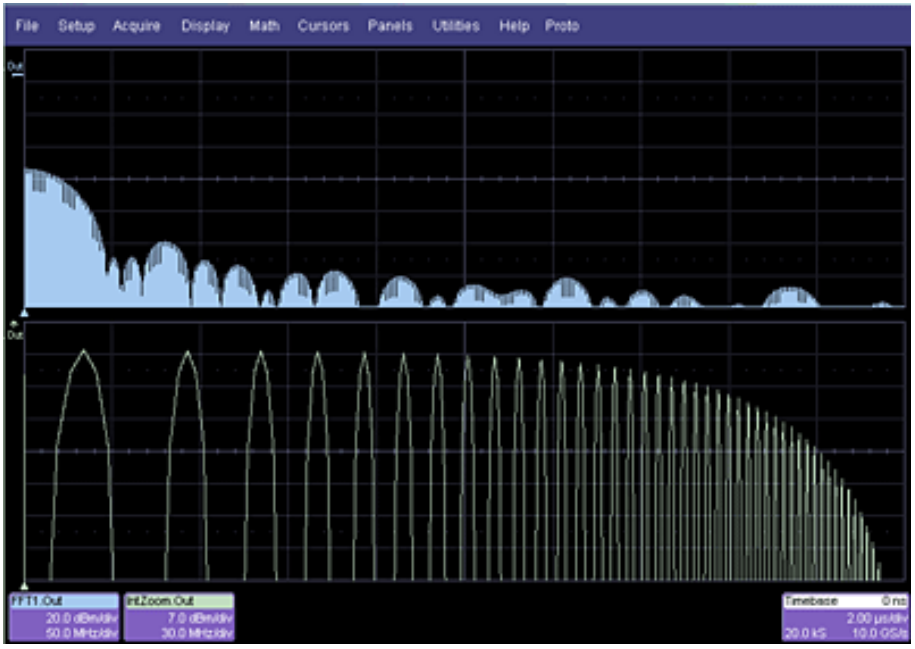
OutResult.DataArray is the waveform returned to the scope and displayed on the scope as the F2 waveform.

Example 4



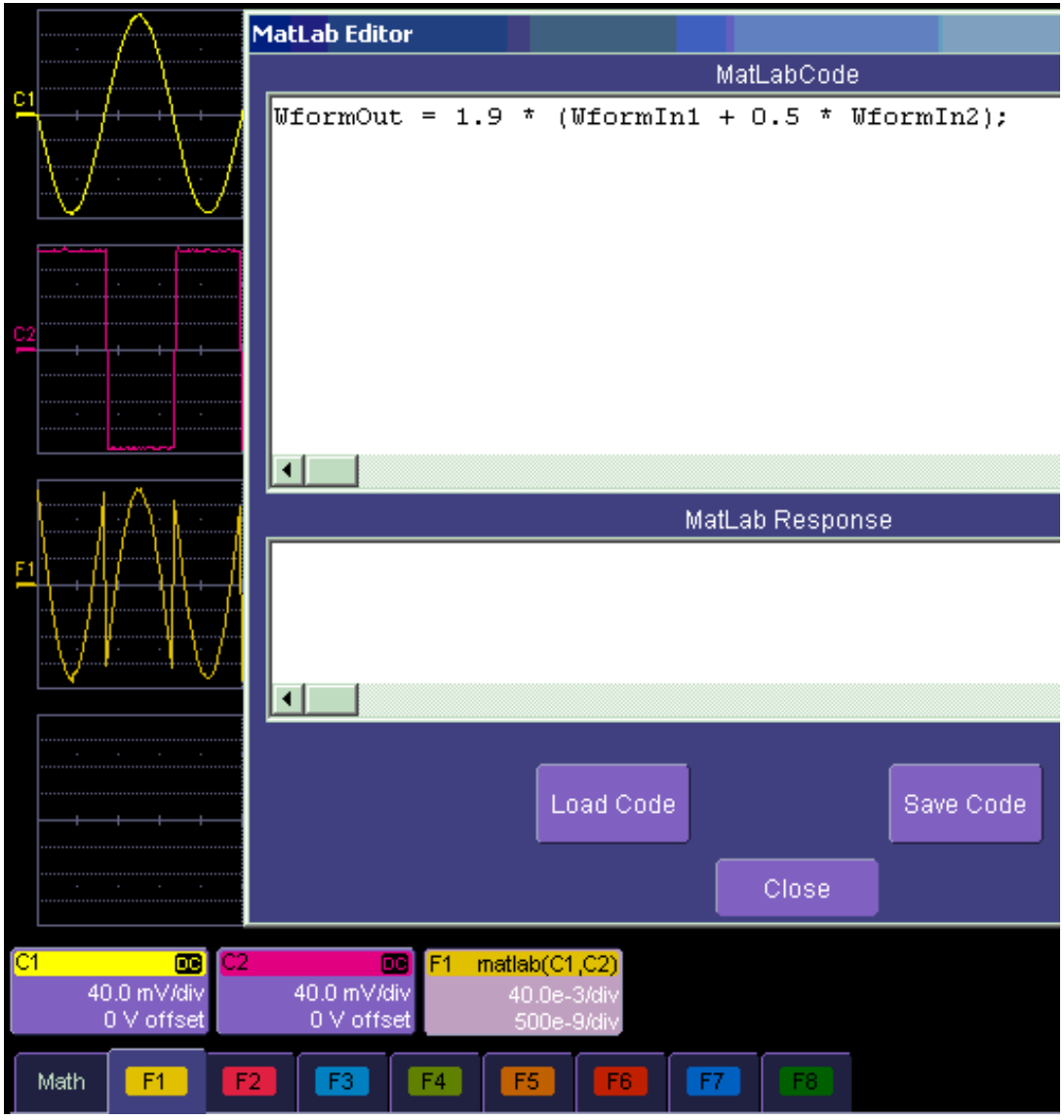
Example 4 is a measurement of DVI (Digital Video Interface) Data-Clock skew jitter measurement, using a VBScript to emulate the PLL.
In this example, a customer was not able to probe the desired clock signal. The only probing point available was the output differential clock signal (C2). However, that clock was a factor of 10 slower than the clock embedded in the data signal (C3). By using a VBScript to create a clock waveform of the appropriate frequency (waveform F1), the customer was able to display and measure data-clock skew using a LeCroy instrument function and parameter.

Example 5



This next example shows a logarithmic vertical scale, for which the script can be found here. (Most scripts would be far simpler than this one.)
Frequency response curves are frequently drawn on a logarithmic scale. The upper trace is a frequency spectrum of a square wave after enhanced resolution has been applied. It was created using instrument functions. The lower trace is the first lobe of the FFT display. But with a logarithmic frequency scale. Click here for the VBScript.
In addition to VBScripting, MATLAB, Mathcad, or Excel can also be used to generate a result. The F1 trace (shown below in Example 6) was calculated in MATLAB (F1=WformOut) from C1 (WformIn1) and C2 (WformIn2). The same calculation could also be done in Excel by using a simple formula in a spreadsheet cell.

Example 6


MatLab Editor
MatLabCode
WformOut = 1.9 * (WformIn1 + 0.5 * WformIn2);
MatLab Response
Load Code
Save Code
Close
C1
C2
F1
C1 DC
40.0 mV/div
0 V offset
C2 DC
40.0 mV/div
0 V offset
F1 matlab(C1,C2)
40.0e-3/div
500e-9/div
Math
F1
F2
F3
F4
F5
F6
F7
F8

Summary

The examples above illustrate only the capability to use VBScript and MATLAB. The instrument with the LeCroy XMAP software option allows you to use Excel, Mathcad, MATLAB, and VBScript in this manner. Of course, you will need to load Excel, Mathcad, or MATLAB in the scope (VBScript does not require any additional software) to take advantage of the capability. You can think of these functions as "subroutines" of the instrument's main software, which take in waveform data and other variables like vertical scale and offset and horizontal scale and offset. These functions then return a waveform or a parameter as required. In addition, you can view the calculated data directly in Excel, MATLAB, or Mathcad, if you desire.

What is Excel?

Excel is a program within Microsoft Office. With it you can place data in the cells of a spreadsheet, calculate other values from them, prepare charts of many kinds, use mathematical and statistical functions, and communicate with other programs in Office. From the instrument you can send data to Excel (where processing can take place) and return the results to the instrument.

What is Mathcad?

Mathcad is a software package from MathSoft. It provides an integrated environment for performing numerical calculations and solving equations, and communicating with other programs. Results can be presented in tabular or graphical form.

What is MATLAB?

MATLAB is a software package from MathWorks that provides an environment for work in computation and mathematics. An interactive language and graphics are provided.

What is VBS?

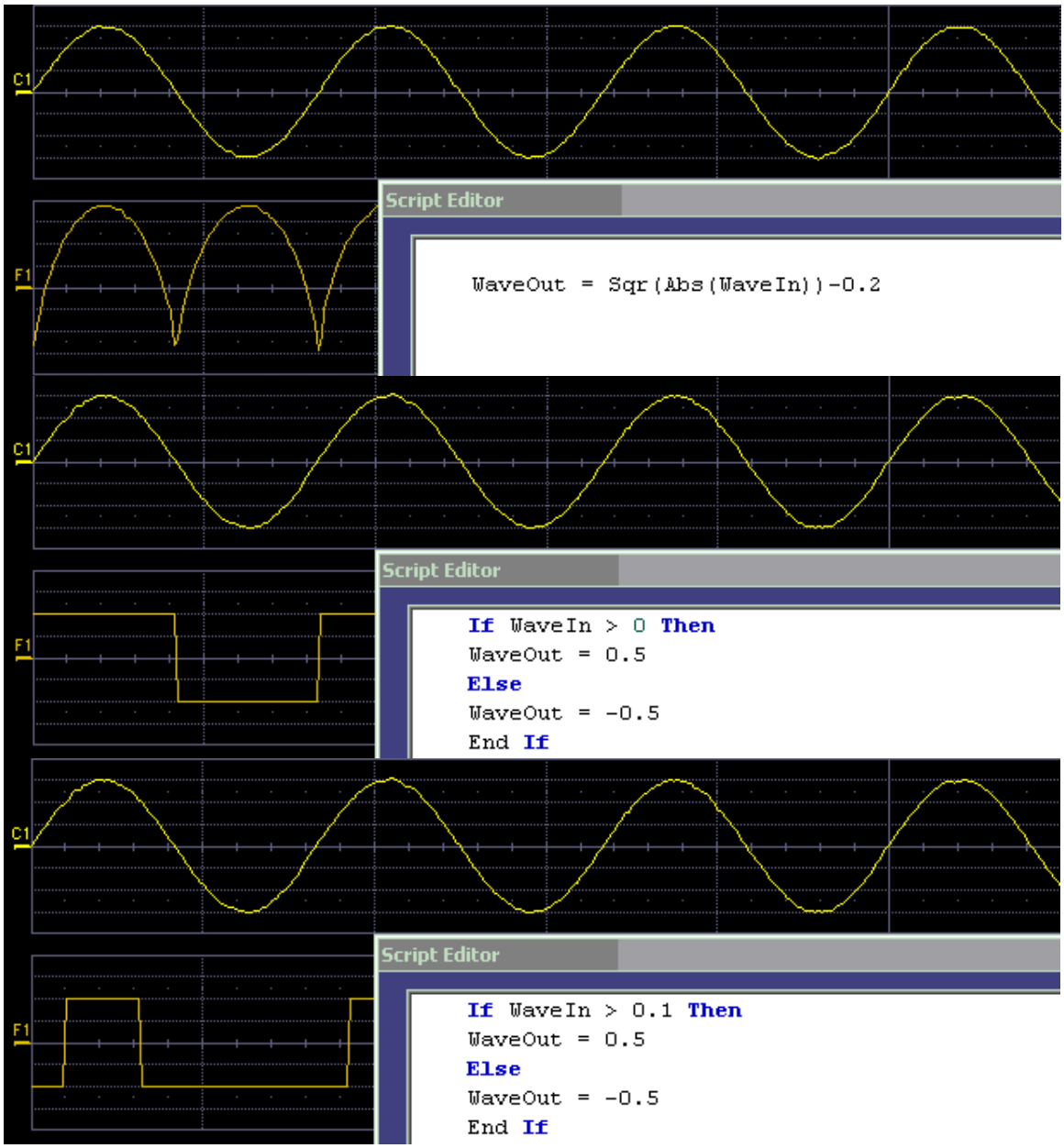
VBS is a programming language, but you don’t write it in a special environment such as C++ or Visual Basic; you write it within your own application. In the instrument, a few clicks or button pushes will get you into an editing panel where you can write what you want. You cannot crash the scope, or in any other way interfere with its workings, because the system is completely protected.

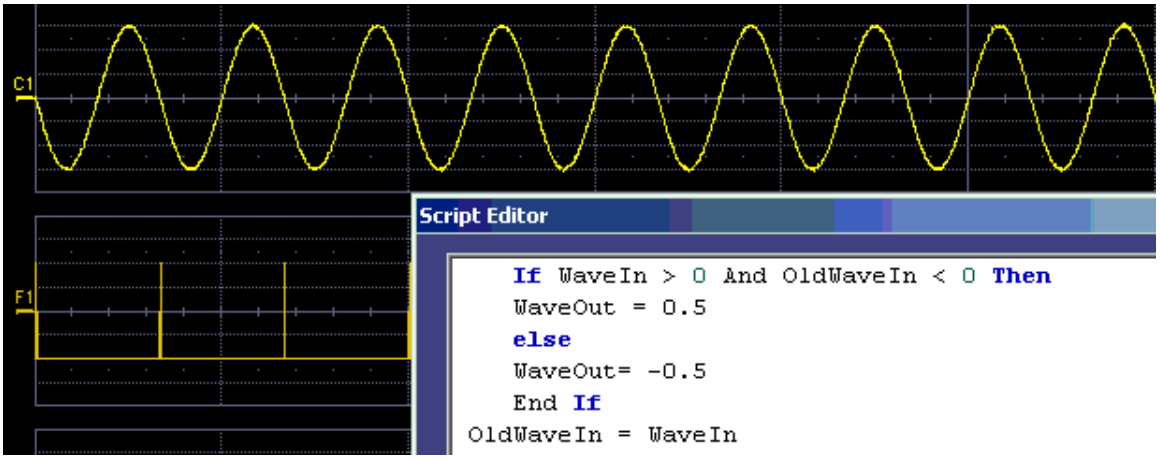
A product of Microsoft and a subset of Visual Basic, VBS can be learned very quickly if you have some experience in any programming language. The VBS processing function can collect a number of useful variables from the scope, including waveform data and useful variables such as volts per division and time per division. The output from a script can be a waveform or a parameter, and you can choose your own values for variables such as volts per division.

The idea of a VBS function is that you start with an input waveform, operate on some or all of the values with a script, and show the result on a scope grid, like any other waveform.

VBScript customization is built into the instrument, so no additional programs need to be loaded to take advantage of this capability.

The following diagrams were made by changing a small part, in some cases just one line, of a standard VBScript. VBS is a well-known standard language, with excellent support documentation, and it is easy to use in several different environments.


C1
F1
Script Editor
WaveOut = Sqr(Abs(WaveIn))-0.2
C1
F1
Script Editor
If WaveIn > 0 Then
WaveOut = 0.5
Else
WaveOut = -0.5
End If
C1
F1
Script Editor
If WaveIn > 0.1 Then
WaveOut = 0.5
Else
WaveOut = -0.5
End If



These examples are purely illustrative, but you can easily imagine that with a VBScript you can add value to the scope in a very short time. This gives you an instrument that does exactly what you want, time after time, by using your stored setups and scripts.

What can you do with a customized instrument?

If you require a result that can be derived logically from the input waveform, you can do it. Many calculations can be done with remarkably small scripts, but if you have no time for scripting, you can use one of the proprietary packages, such as Excel, MATLAB, or Mathcad, which offer immense processing power.

Scaling and Display

Scripting and programming allow a large variety of opportunities. You may, for example, be using transducers. If so, you can change the units of your waveforms, and write N (newtons), J (joules) and so on, and you can introduce scaling factors. If the transducers are non-linear, you can correct for that, too. You can also transform horizontal scales and vertical scales by manipulating the data. Logarithmic scales in amplitude and frequency are often required. Squaring and taking square roots are needed in certain applications. Here is a picture showing some graphs related to white noise, showing ways of detecting small deviations from the true distribution. The lower two graphs were generated and placed in one trace using a VBScript.
In the next example, four graphs are placed in one trace.

Golden Waveforms

This is a rich field for VBS. An example was given earlier. The only limits to the shapes that can be generated are the vertical resolution and the number of samples.

A practical example – DVI Data-Clock skew

The next example is a measurement of DVI Data-Clock skew jitter measurement, using a VBScript to emulate the PLL. A solution to a practical measurement problem was shown earlier. These are just a few of the many solutions that can be created.

Number of Samples

The various math packages can process samples as follows:

Excel	65,535 samples
Mathcad	5 MS

The number of samples that MATLAB can process is determined by memory option, as follows:

Memory Length	System DRAM	Buffer Length
STD, S, M	256 MB	40 MB
L, VL	512 MB	200 MB
XL	1 GB	400 MB
XXL	2 GB	400 MB

Calling Excel from Your Instrument

Excel can be directly called from the instrument in two ways:

Using a function	F1 through Fx [The number of math traces available depends on the software options loaded on your scope. See Specifications.]	Excel returns a waveform
Using a parameter	P1 through Px [The number of parameters available depends on the software options loaded on your scope. See Specifications.]	Excel returns a parameter

In both cases, one call to Excel can use two separate waveforms as input.

Please Note the Following:

- Excel has a calculation algorithm of 64,000 points (32,000 if you have created a chart in Excel). Therefore, make sure that your acquisition has less than this number of points if you are going to use an Excel calculation.
- To use this capability, you must have the LeCroy XMAP software option and Excel loaded in your instrument. Select **Minimize** from the instrument's **File** menu to access the Excel program directly.

How to Select a Math Function Call

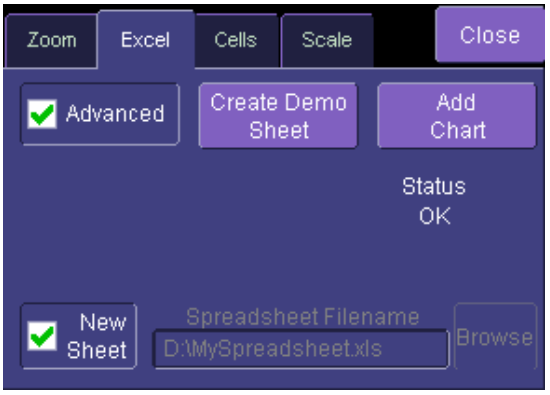
The Excel math function is selected from the **Math Operator** menu, where it appears in the **Custom** group.

How to Select a Parameter Function Call

The Excel Parameter function is selected from the **Select Measurement** menu, where it appears in the **Custom** group.

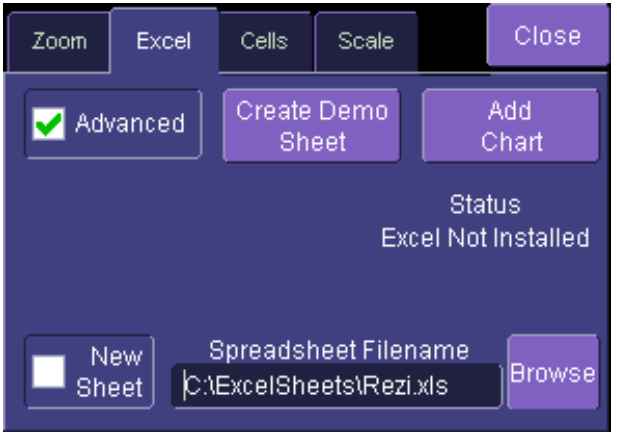
The Excel Control Dialog

Once you have invoked an Excel call, you will see a dialog at the right of the screen, allowing you to control the zoom, Excel properties, linking cells, and scale of the output trace from Excel:



Entering a File Name

If you uncheck the **New Sheet** checkbox, you can enter the file name of an existing file.

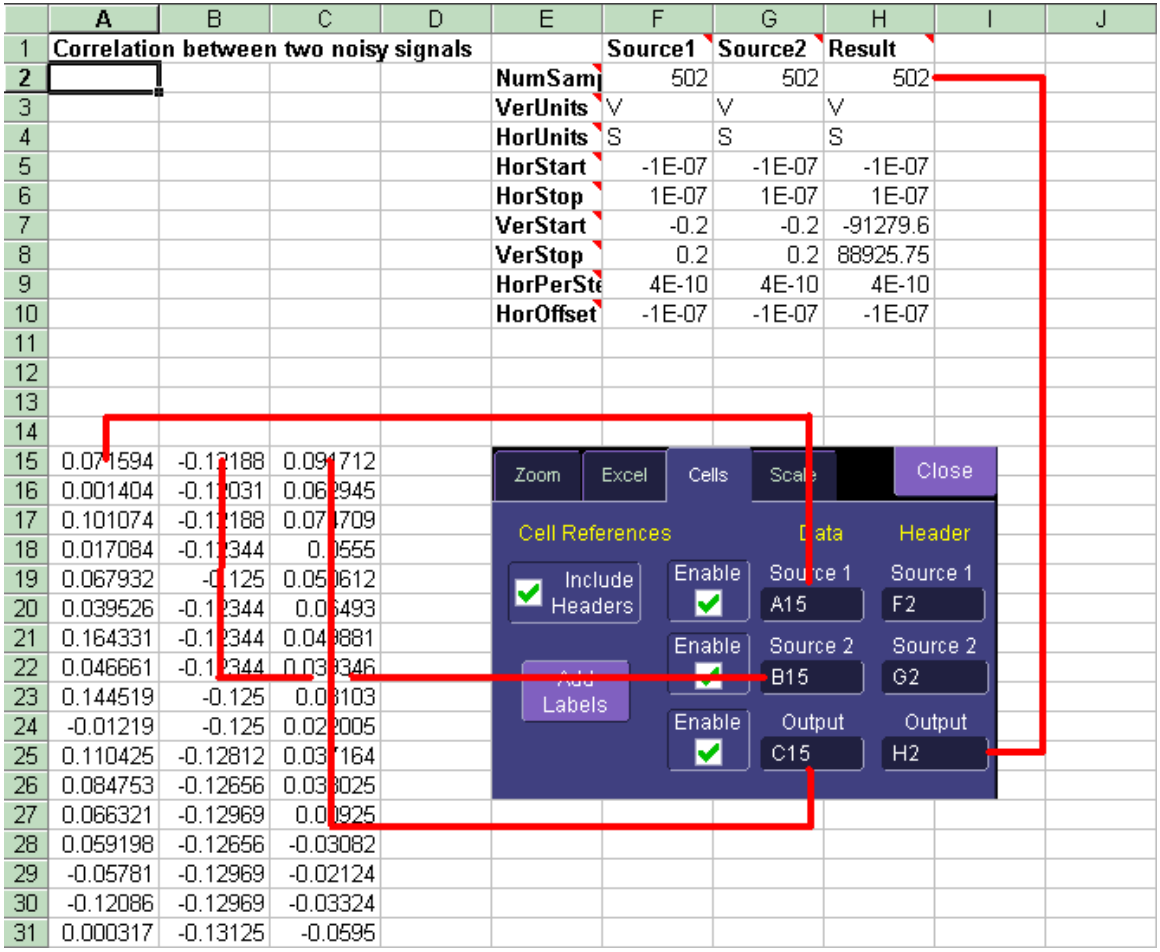


- **Create Demo Sheet –** Calls up a default Excel spreadsheet.
- **Add Chart –** Adds charts of your waveforms to Excel. You can go into Excel and create as many charts as you want.

Organizing Excel Spreadsheets

The **Cells** tab allows you to organize your Excel chart. When placing the components in the sheet, be careful to avoid over-writing needed information, especially when you are using multiple input waveforms.

As depicted here, the instrument panel has been pasted over the Excel Spreadsheet:

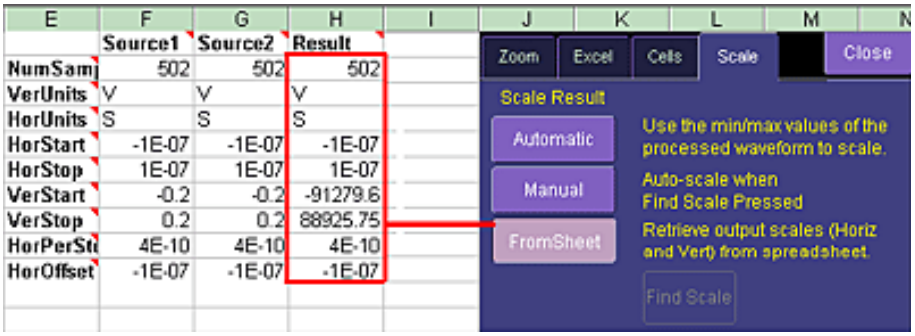


There are three arrays of data for the three waveforms: up to two inputs and one output. There are corresponding small arrays of information about each trace.

Setting the Vertical Scale

The vertical scale of the output waveform from Excel may be set in three ways:

Automatic	For each acquisition, the instrument fits the waveform into the grid.
Manual	For one acquisition, click **Find Scale**; the instrument fits the current waveform into the grid. All subsequent acquisitions will use this scale until you make a change.
From Sheet	The scale is taken from the specified cells in the Excel sheet, H2 through H10 in the example above, where cell H2 was specified as the top of the data set, as depicted below.



Trace Descriptors

The next figure explains the meanings of the descriptors for each trace.

C	D	E	F	G	H
ı two noisy signals			**Source1**	**Source2**	**Result**
Samples in trace		**NumSamj**	502	502	502
Vertical units		**VerUnits**	V	V	V
Horizontal units		**HorUnits**	S	S	S
Left edge of grid		**HorStart**	-5E-08	-5E-08	-5E-08
Right edge of grid		**HorStop**	1.5E-07	1.5E-07	1.5E-07
Bottom of grid		**VerStart**	-0.2	-0.2	-0.32585
Top of grid		**VerStop**	0.2	0.2	0.266995
Time per sample		**HorPerSt**	4E-10	4E-10	4E-10
Trigger position		**HorOffset**	-5E-08	-5E-08	-5E-08

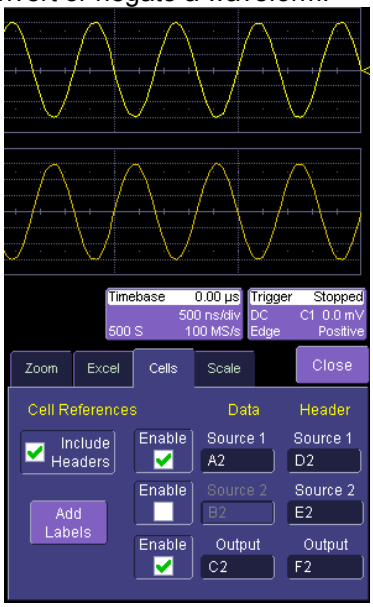
Multiple Inputs and Outputs

If you invoke two or more instrument parameter functions or waveform functions that call Excel, you will find that they all refer to the same spreadsheet by default. Thus, your spreadsheet can use the data from several waveforms, and you can derive many different combinations of output parameters and waveforms, including some of each, from your spreadsheet. You only have to be careful about the positioning of your cell ranges within the sheet so that no conflicts occur.

Because filling cells in the spreadsheet is a relatively slow process, all unwanted sources (inputs) should be left disabled (unchecked). For example, if you want one waveform and two parameters derived from the data of three waveforms, you can have one function with both sources enabled, one with one source enabled, and one with no sources enabled. The alternative is to use one input in each function.

Simple Excel Example 1

In this example we use Excel to invert or negate a waveform:



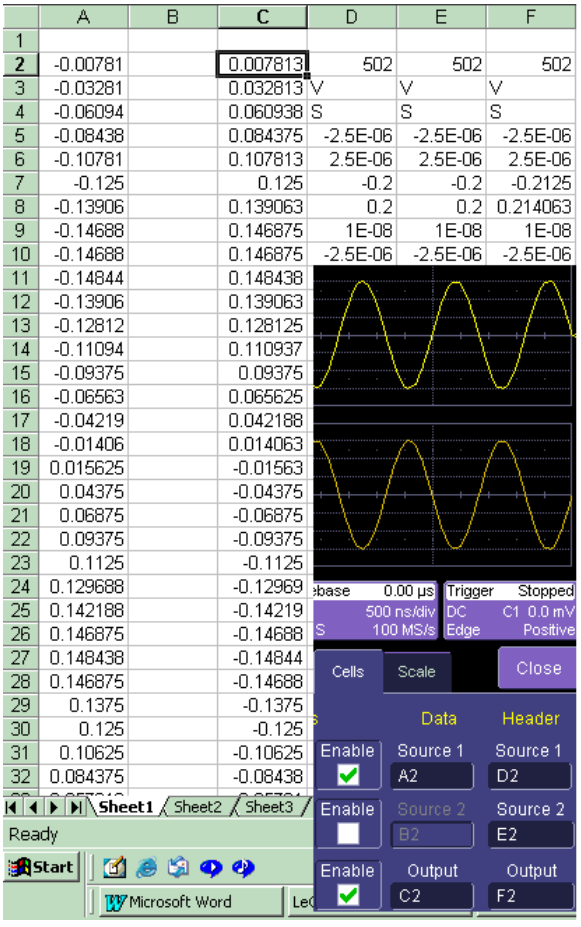
The first figure shows a part of the screen. The upper trace is the original signal. The lower is the result from Excel.

The dialog controls the location of the data in the Excel worksheet.

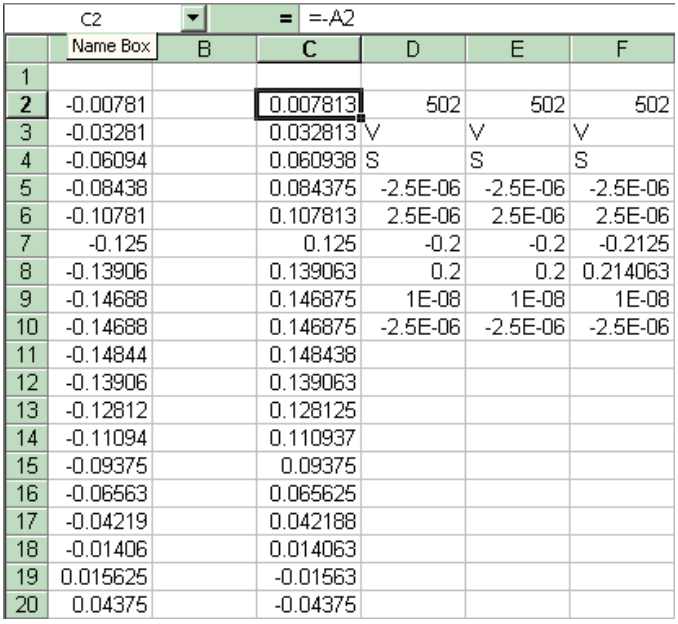
The input data are in columns A and B (though, only the first is used) and the output is in column C. All have been set to start at row 2, allowing space for a title in row 1.

Columns D, E and F contain the headers for the three waveforms. These are the set of numbers that provide the description of the scope settings, such as vertical scale and offset, and number of samples.

In this figure, the panel has been pasted onto the Excel sheet for comparison:



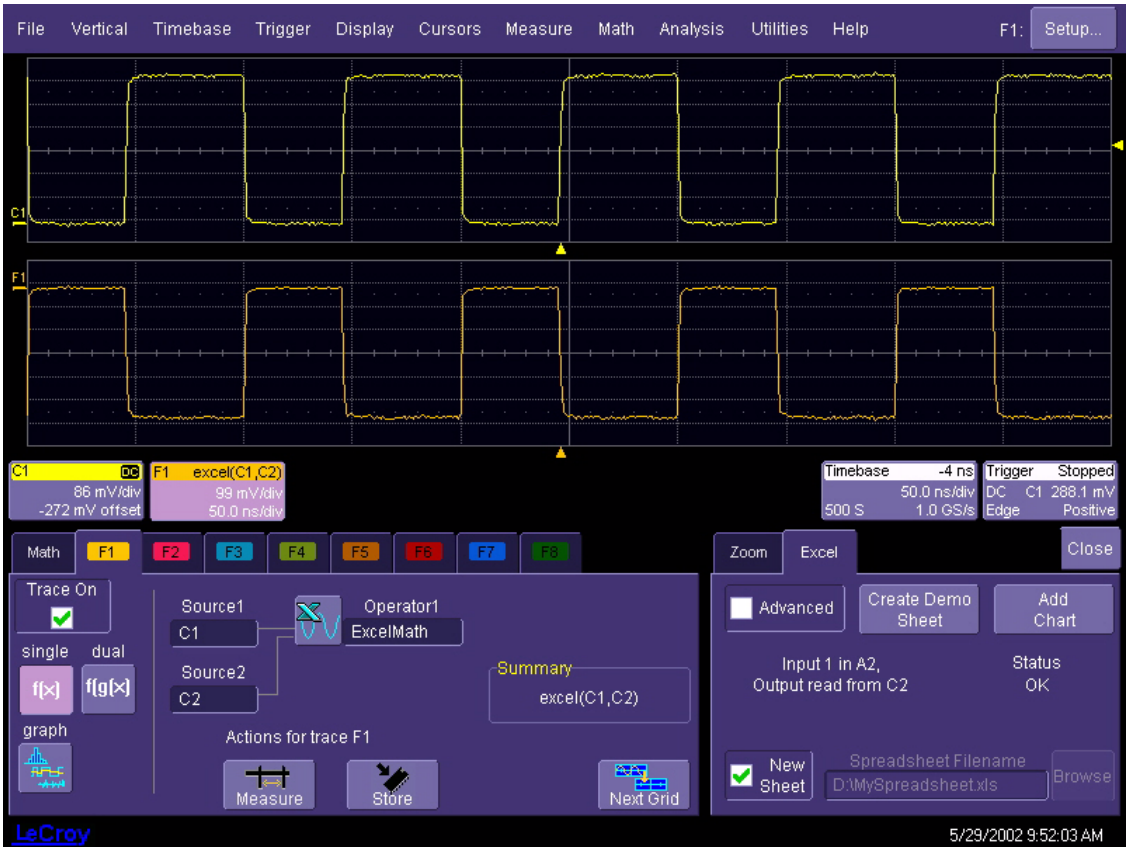
To get the output values in column C, we set C2 = - A2 and copy this formula down the column. This is the only action needed in Excel, and can be seen in the next figure:



	A	B	C	D	E	F
1						
2	-0.00781		0.007813	502	502	502
3	-0.03281		0.032813	V	V	V
4	-0.06094		0.060938	S	S	S
5	-0.08438		0.084375	-2.5E-06	-2.5E-06	-2.5E-06
6	-0.10781		0.107813	2.5E-06	2.5E-06	2.5E-06
7	-0.125		0.125	-0.2	-0.2	-0.2125
8	-0.13906		0.139063	0.2	0.2	0.214063
9	-0.14688		0.146875	1E-08	1E-08	1E-08
10	-0.14688		0.146875	-2.5E-06	-2.5E-06	-2.5E-06
11	-0.14844		0.148438			
12	-0.13906		0.139063			
13	-0.12812		0.128125			
14	-0.11094		0.110937			
15	-0.09375		0.09375			
16	-0.06563		0.065625			
17	-0.04219		0.042188			
18	-0.01406		0.014063			
19	0.015625		-0.01563			
20	0.04375		-0.04375			

Simple Excel Example 2

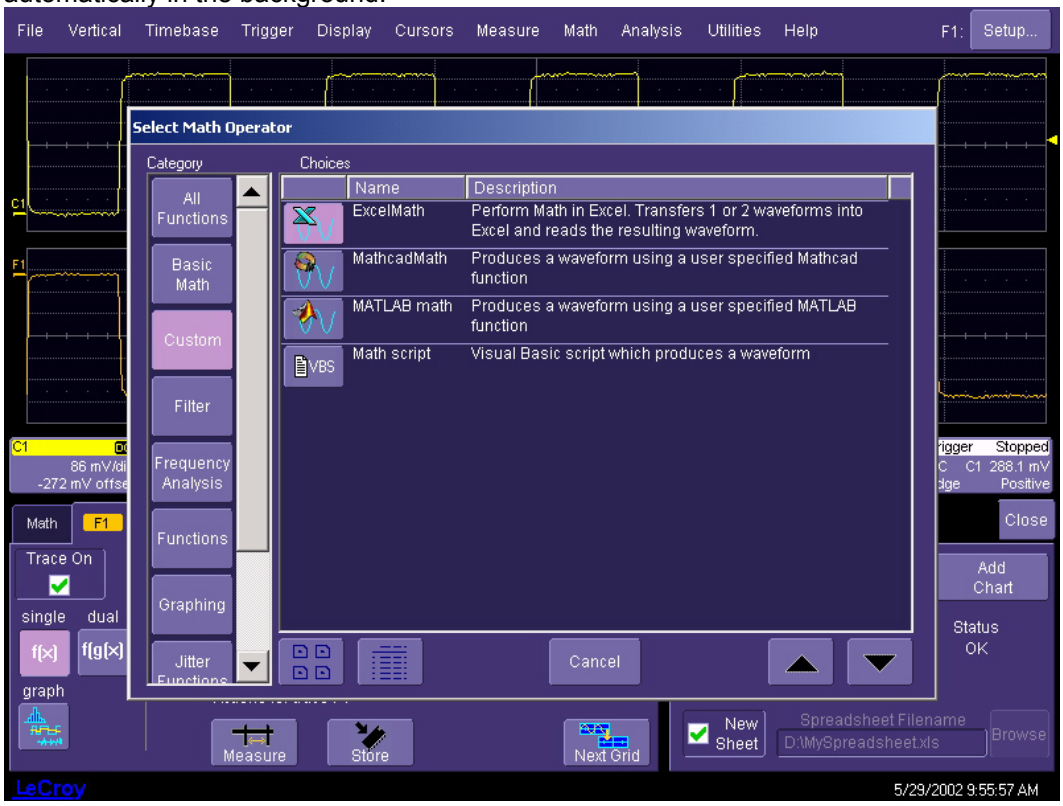
In this example we use Excel to invert or negate a waveform:



The first figure shows a part of the instrument screen. The upper trace (C1) is the original signal. The lower trace (F1) is the result calculated in Excel and displayed on the screen.
The input data is in columns A and B (though by default, only a single input/column is used), and the output is in column C. All have been set to start at row 2 (which allows for a header in row 1).
To create this waveform, you would simply do the following:

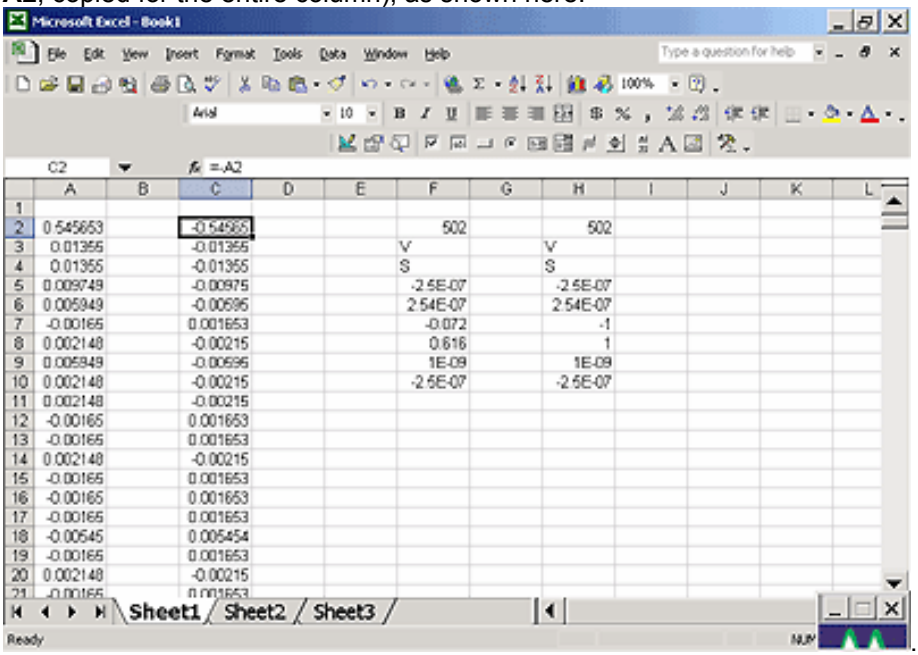
1. Ensure that your acquisition has no more than 64 kpts (the Excel calculation limit)

2. Choose a function, and select ExcelMath as **Operator1** for the function. Excel will open automatically in the background.



3. Choose File, Minimize from the menu bar to minimize the instrument display and open the Excel program.

4. Create your formula for each data point in column A (in this case, our formula for cell C2 is **–A2**, copied for the entire column), as shown here:

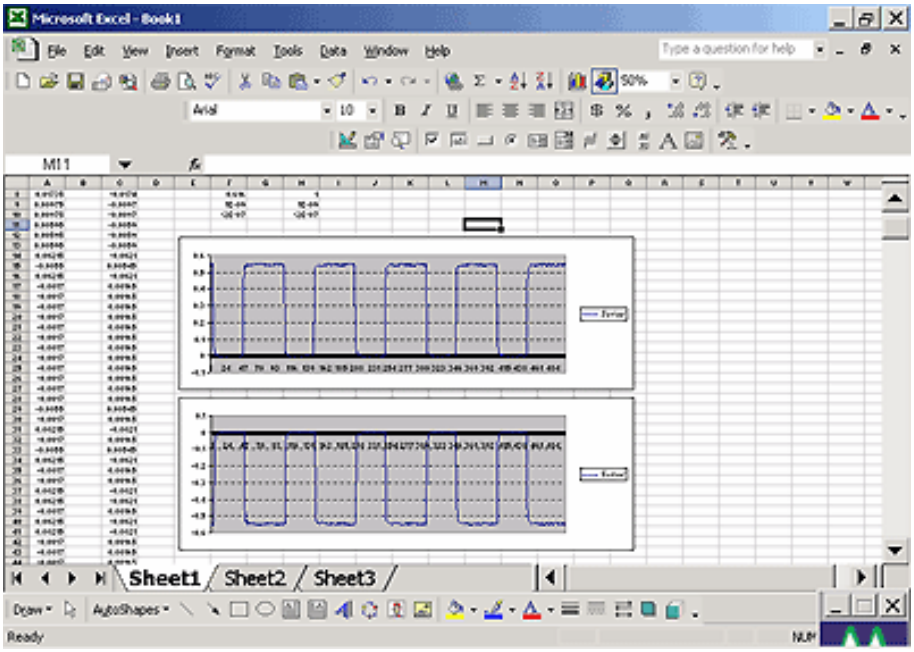


	A	B	C	D	E	F	G	H	I	J	K	L
1												
2	0.545653		-0.54565			502		502				
3	0.01355		-0.01355			V		V				
4	0.01355		-0.01355			S		S				
5	0.009749		-0.00975			-2.5E-07		-2.5E-07				
6	0.005949		-0.00595			2.54E-07		2.54E-07				
7	-0.00165		0.001653			-0.072		-1				
8	0.002148		-0.00215			0.616		1				
9	0.005949		-0.00595			1E-09		1E-09				
10	0.002148		-0.00215			-2.5E-07		-2.5E-07				
11	0.002148		-0.00215									
12	-0.00165		0.001653									
13	-0.00165		0.001653									
14	0.002148		-0.00215									
15	-0.00165		0.001653									
16	-0.00165		0.001653									
17	-0.00165		0.001653									
18	-0.00545		0.005454									
19	-0.00165		0.001653									
20	0.002148		-0.00215									
21	-0.00165		0.001653									

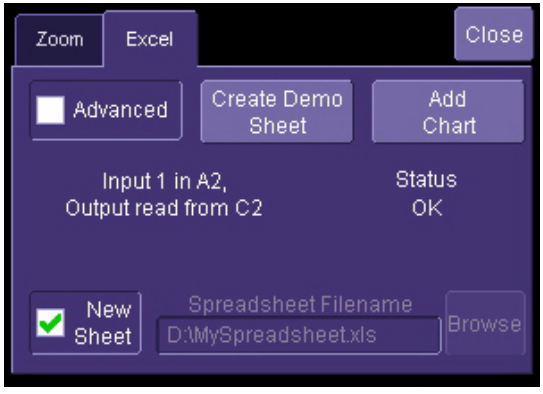
5. Retrigger the scope (if it is not currently triggering)
6. Return to the program

   Note that the only action that was needed in Excel was to create the formula in column C for each data point in column A. The instrument automatically opens Excel, puts the waveform data in the correct columns, and returns the calculated data back to the display as the chosen F trace. This Excel-calculated trace can have further measurements or math calculations performed on it, if desired.

You can also create a chart of the data in Excel automatically and view the data there:

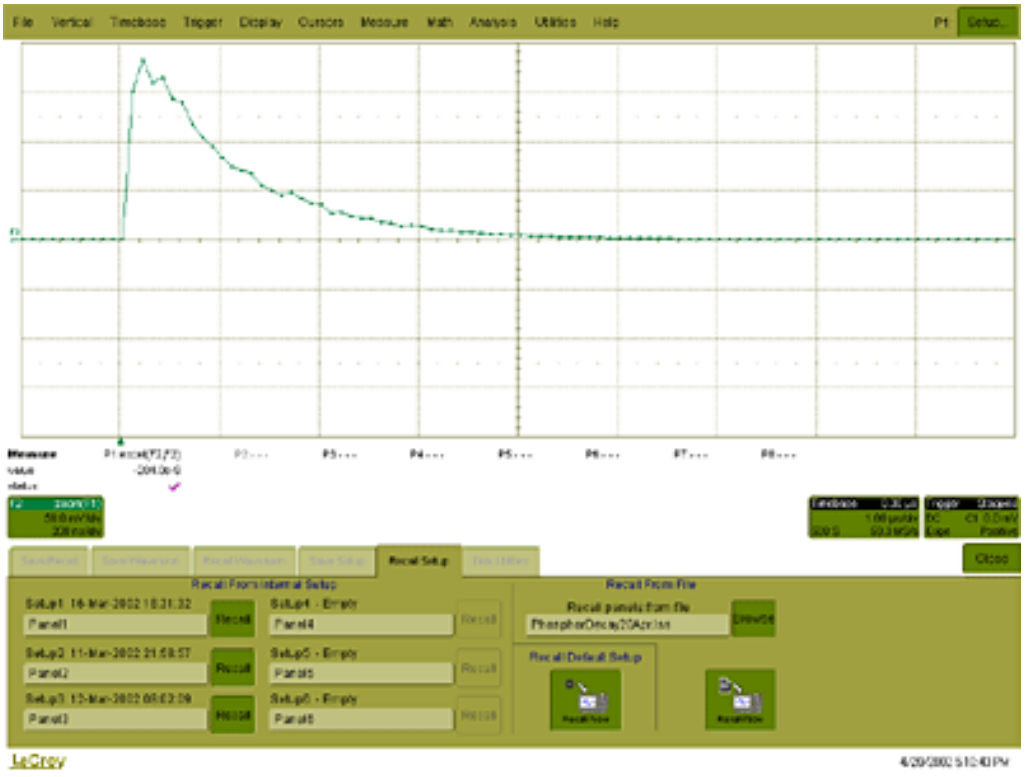


Simply press the **Add Chart** button in the instrument's Excel dialog and a chart of the input (top chart) and Excel calculated output (bottom chart) will be automatically created in the spreadsheet. The chart will be updated automatically as the scope is triggered.



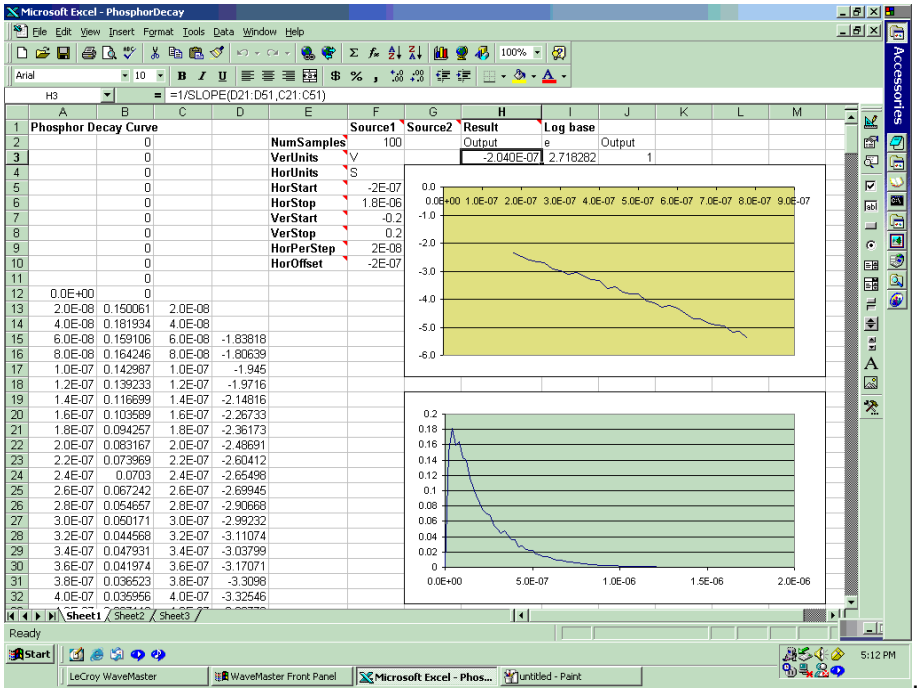
Excel Example 1: Exponential Decay Time Constant Excel Parameter

This example calculates the time constant of an exponentially falling pulse, such as the light output of a phosphor.
The first figure shows a typical pulse, including pseudo-random noise, generated by a VBScript:



The pulse was generated by a formula of the form $e^{(1 - t/TC1)} * e^{-t/TC2}$, where TC1 and TC2 are time constants, The requirement is to measure the time constant TC2, using the portion of the trace where TC1 has negligible effect. This was done using Function F1, which is not a part of the measurement process.

For the actual measurement, Parameter P1 was set up as an Excel call. In Excel, the selected portion of the trace was converted to logarithms, and the Excel function SLOPE was used, as shown here:

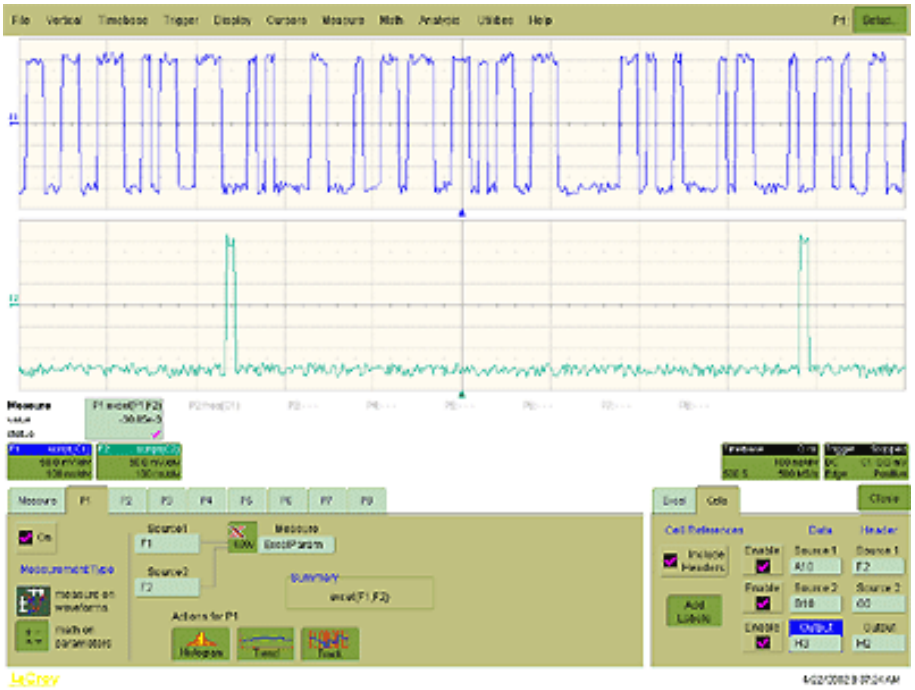


Here we see the input data in column B (with a time scale in A) created using the contents of cell F9, Horizontal Per Step. The logarithmic data are in column D, with the time scale repeated in C. The output appears in cell H3, using the formula **=1/SLOPE(D21:D51,C21:C51)**.

Excel Example 2: Gated Parameter Using Excel

This example calculates a parameter of a waveform, in a region of interest defined by the leading edges of two pulses in a separate waveform.

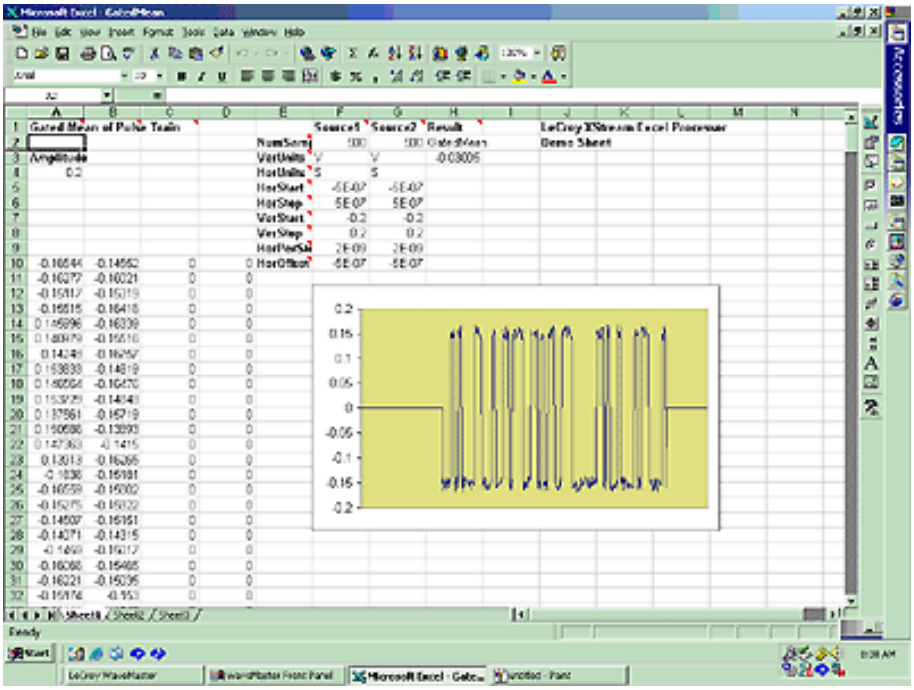
This figure shows the instrument screen:



These traces were made using VBS scripts in functions F1 and F2, based on pseudo-random numbers to provide noise and varying pulse widths. Randomize Timer was used in both scripts ensuring successive acquisitions to produce different data. Script F1 generates pulses having widths that are multiples of a set clock period. F2 generates one pulse in the first half of the time window, and one pulse in the second half. Both pulses are constrained to coincide with the clock pulses of F1. F1 and F2 are used here only as simulations and are not part of the measurement process, which only uses P1.

The call to Excel is made through Parameter P1.

The next figure shows a part of the Excel workbook:



Here we see the gated waveform that has been created in Excel. The Mean parameter during the region of interest (ROI) is placed in cell H3.

How Does this Work?

The amplitude of the signal is about 0.3 volts, and the screen height is 0.4 volts, as derived from cells F7 and Fx. A threshold value for amplitude was calculated by placing 0.5 * (Fy – Fx) in cell A4.

Remember that in the instrument the sources were defined to be A10 and B10. This means that the first point on the waveform will be read into A10, and, since the waveform has 500 points, the last point will be read into A510. The same holds true for F2 and column B, since F2 is assigned as Source2, and data is defined to write into column B starting with cell B10.

To create the gating function in column C, the cell C10 was given the following formula:

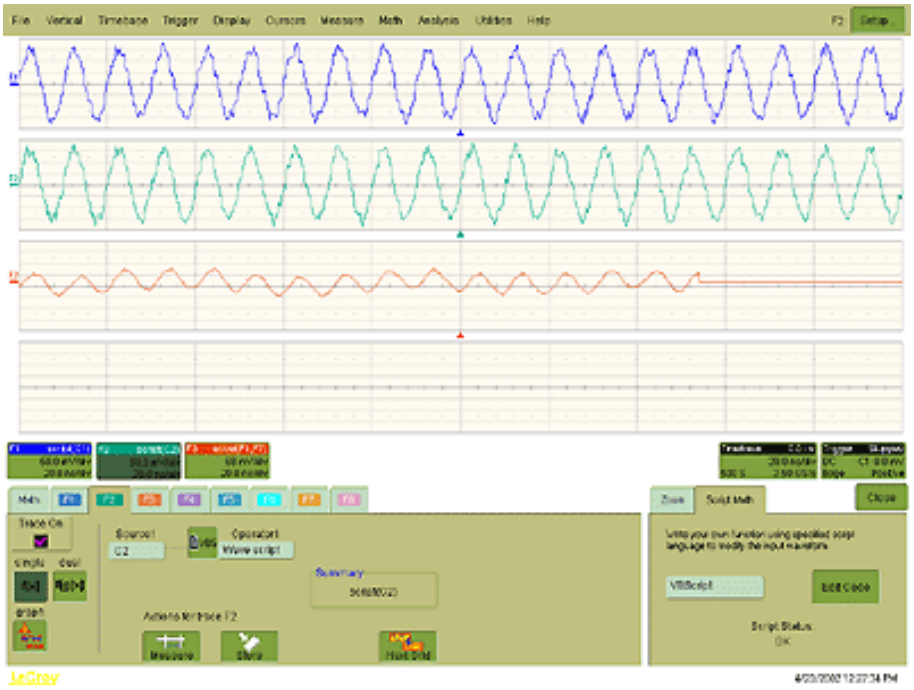
IF ((B10 – B9) > A4, 1 – C9, C9). This was copied down the column. Column D, the output column, is simply A * C.

The output was defined as cell H3.

The required mean in cell H3 is given by SUM (D10 : D509) / SUM (C10 : C509), for a 500 point waveform.

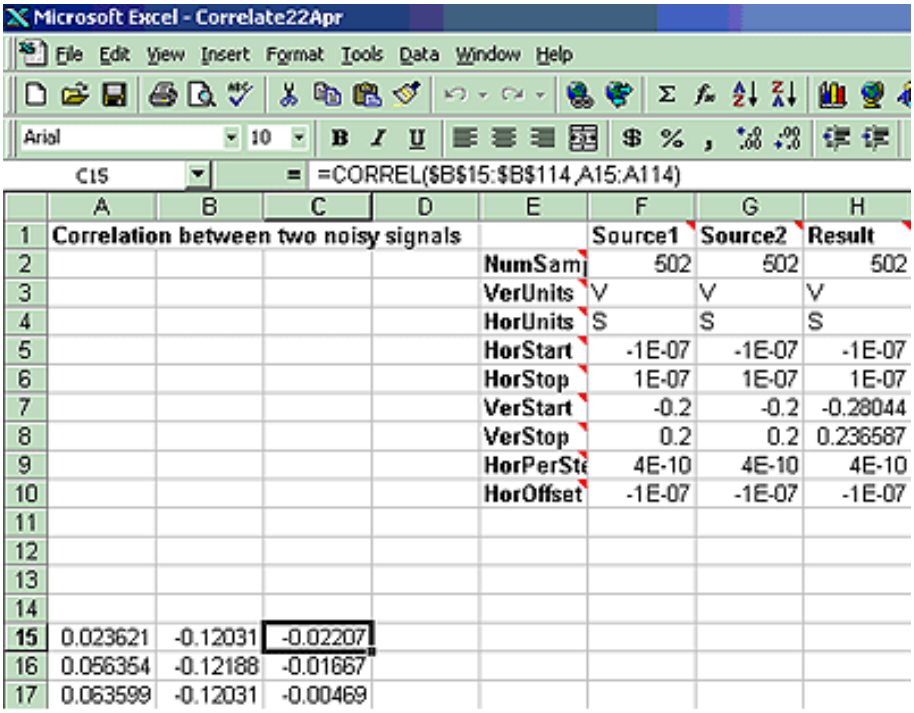
Excel Example 3: Correlation Excel Waveform Function

This example uses an Excel waveform function to examine the cross-correlation between two signals, which are both noisy sinusoidal segments. The correlation trace is, of necessity, shorter than the input traces:



The noise was generated using pseudo-random numbers. Randomize Timer was included in the VBScript to ensure that the two traces differed, and that subsequent acquisitions differed. Functions F1 and F2 are included only to simulate signals, and are not part of the measurement process, which is performed by F3.

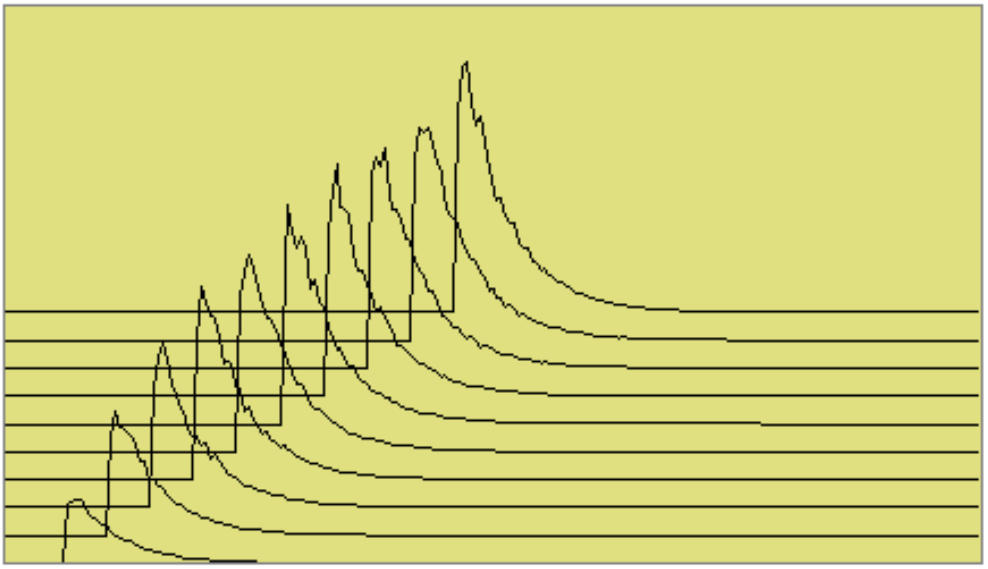
This example used the CORREL (Array1, Array2) function of Excel, as depicted below:



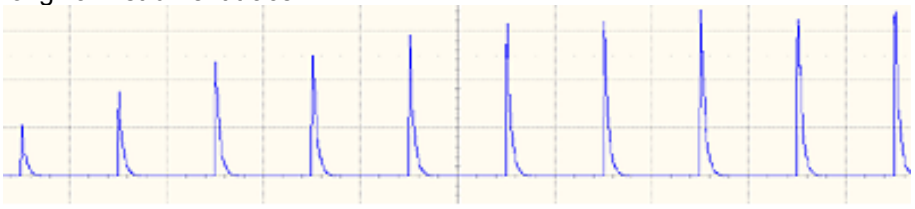
	A	B	C	D	E	F	G	H
1	Correlation between two noisy signals					Source1	Source2	Result
2					NumSam	502	502	502
3					VerUnits	V	V	V
4					HorUnits	S	S	S
5					HorStart	-1E-07	-1E-07	-1E-07
6					HorStop	1E-07	1E-07	1E-07
7					VerStart	-0.2	-0.2	-0.28044
8					VerStop	0.2	0.2	0.236587
9					HorPerSt	4E-10	4E-10	4E-10
10					HorOffset	-1E-07	-1E-07	-1E-07
11								
12								
13								
14								
15	0.023621	-0.12031	-0.02207					
16	0.056354	-0.12188	-0.01667					
17	0.063599	-0.12031	-0.00469					

Excel Example 4: Multiple Traces on One Grid

This example shows how you can place multiple traces in one picture, with only two operations in an Excel sheet. Depicted below is an example from an Excel spreadsheet.

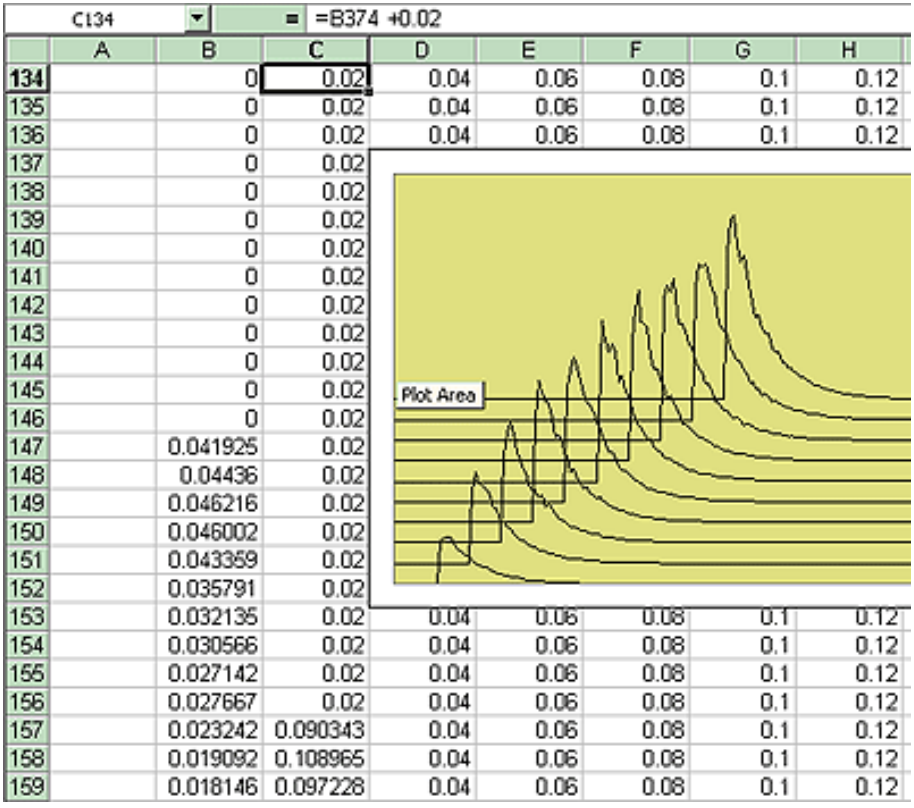


Here is an original instrument trace:



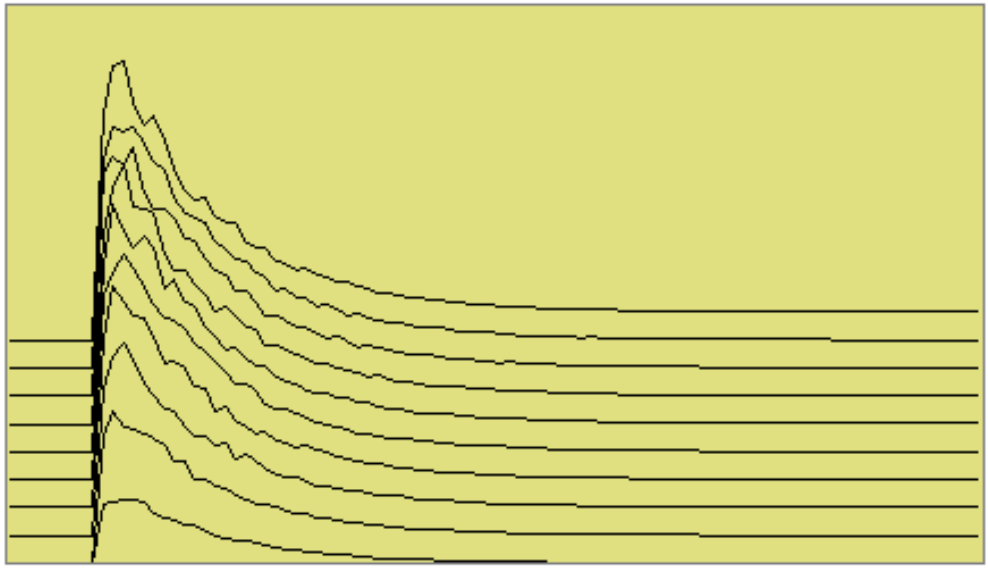
The method is very simple. First, the waveform is transferred to an Excel spreadsheet by means of an instrument Excel call. Second, two operations are needed in Excel: placing a simple formula in one cell, and copying that formula into a range of cells.

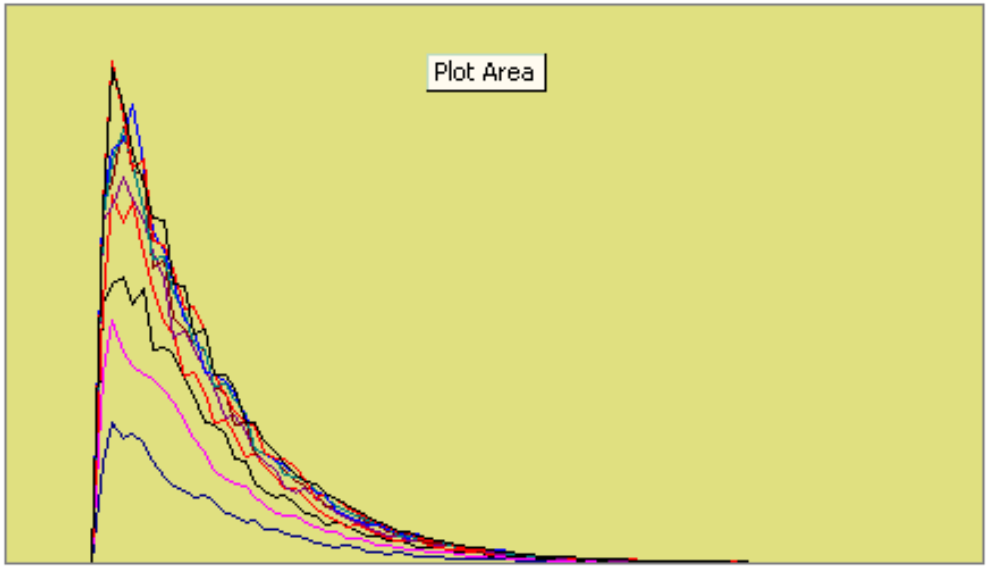
The required Excel formula required is shown as follows:



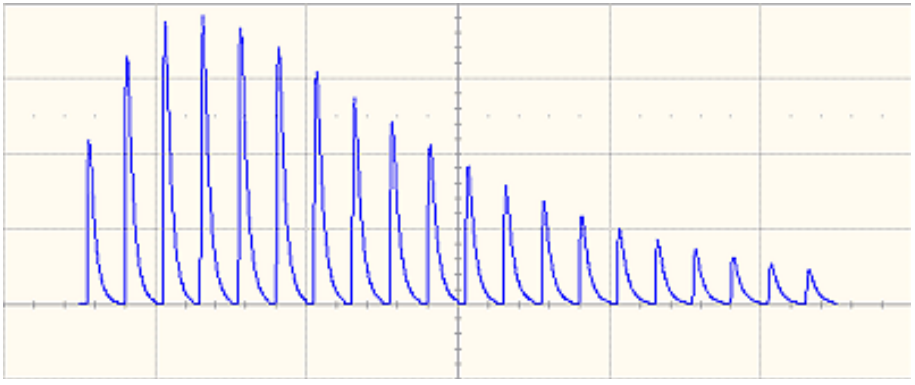
	A	B	C	D	E	F	G	H
134		0	0.02	0.04	0.06	0.08	0.1	0.12
135		0	0.02	0.04	0.06	0.08	0.1	0.12
136		0	0.02	0.04	0.06	0.08	0.1	0.12
137		0	0.02					
138		0	0.02					
139		0	0.02					
140		0	0.02					
141		0	0.02					
142		0	0.02					
143		0	0.02					
144		0	0.02					
145		0	0.02					
146		0	0.02					
147		0.041925	0.02					
148		0.04436	0.02					
149		0.046216	0.02					
150		0.046002	0.02					
151		0.043359	0.02					
152		0.035791	0.02					
153		0.032135	0.02	0.04	0.06	0.08	0.1	0.12
154		0.030566	0.02	0.04	0.06	0.08	0.1	0.12
155		0.027142	0.02	0.04	0.06	0.08	0.1	0.12
156		0.027667	0.02	0.04	0.06	0.08	0.1	0.12
157		0.023242	0.090343	0.04	0.06	0.08	0.1	0.12
158		0.019092	0.108965	0.04	0.06	0.08	0.1	0.12
159		0.018146	0.097228	0.04	0.06	0.08	0.1	0.12

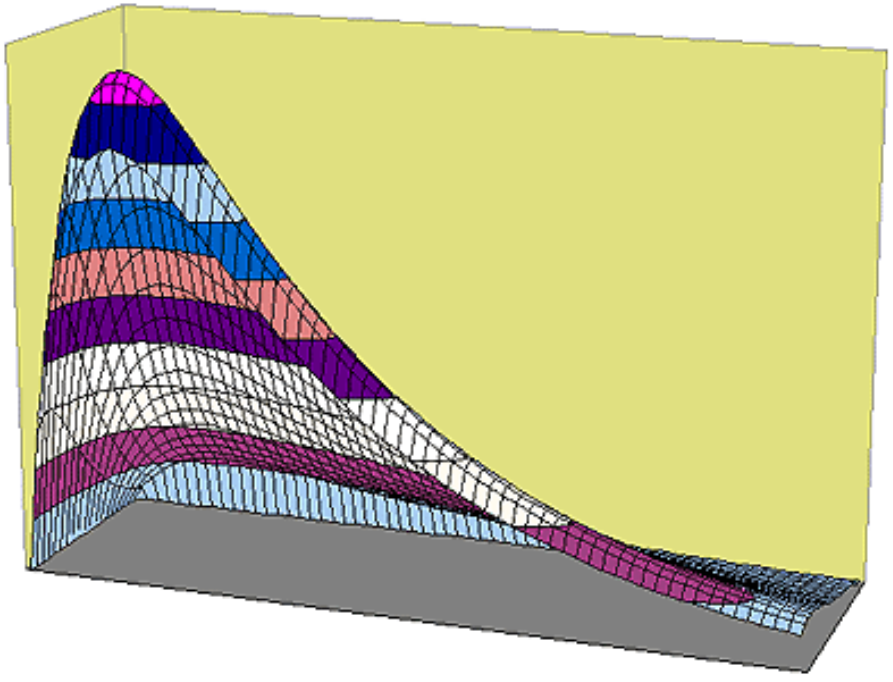
In fact, the simple expression B374 + 0.02 comprise several components. The original instrument trace is in column B, and the plot is required to start at cell B134. The traces repeat at intervals of 250 cells. Let us call this interval R. If we require a horizontal displacement D, then in cell CN we write B(N + R – D). In this example D is 10. Finally we may want a vertical displacement V, and we write B(N + R – D) + V. In this example, V is 0.02. D and V can be zero if required, as depicted below. All that remains is to copy the formula to the required range of cells.




Plot Area

Excel Example 5: Using a Surface Plot





Writing VBScripts

VBScripting is one of the custom features of your instrument. Others include the ability to work with programs such as Excel, Mathcad and MATLAB.

Types of Scripts in VBS

The instrument's VBS provides two types of script.

1. The Waveform Function script allows you to take the data from one or two traces and make a new trace whose values may depend on the values of the input trace.
2. The Parameter Function script also takes in the data from one or two traces, but it only has one output. This output is the zeroth element in the output array. It appears as a parameter value on the instrument's screen. The remainder of the array is currently not used, and is not accessible.

Excel can be called from either script type.

Loading and Saving VBScripts

From the editing panel you can save your script and you can load a previous one. Should you forget to save a script, please note that when you save your setup, it has your current scripts embedded in it. Therefore it is a good idea to save your setup frequently. It is worth saving the script separately as well, because it is saved in a suitable format for printing or off-line editing with Notepad. Note that in both these examples the input data are referred to as InResult.DataArray. You can also write InResult1.DataArray and InResult2.DataArray, which refer to the two input traces. InResult.DataArray always refers to input trace 1. These remarks hold for any script that you write.

Example Waveform Function Script: Square of a waveform

```
' Example script to produce a waveform
This example calculates the square of
the input waveform.
OutResult.Samples = InResult.Samples ' Visible trace length + 1
' Note that a trace of nominal length 1000 comprises data numbered from
' 0 to 1001. The 1001st point is not visible, so you
' normally use points 0 to 1000,
' giving 1001 points and 1000 intervals between points.
startData = 0
endData = OutResult.Samples
LastPoint = endData - 1 ' because the last point is invisible.
ReDim newArray(OutResult.Samples) ' to store the results
unscaledData = InResult.DataArray(False)
' InResult.DataArray(False) provides
' integer data from -32768 to 32767.
' InResult.DataArray(True) provides real data
' in the same physical unit as the vertical scale of the input trace.
ScaleFactor = 1.0 / 32768 ' to make the trace fill the screen.
       For i = 0 To LastPoint
```

```
       newArray(i) = ScaleFactor * (unscaledData(i)) ^ 2
       Next
OutResult.DataArray(False) = newArray ' signed long integer data output
```

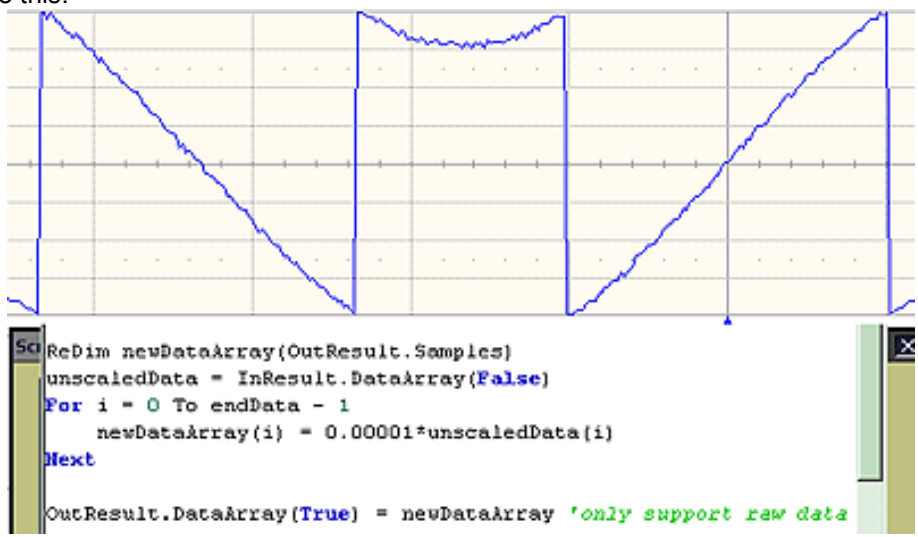
Example Parameter Function Script: RMS of a waveform

```
' Example script to produce a parameter.
' This script calculates the root mean square
' of the input waveform.
' Note that a trace of nominal length 1000 has data from
' 0 to 1001. The 1001st point is not visible, so you
' normally use points 0 to 1000,
' giving 1001 points and 1000 intervals between points.
startData = 0
endData = InResult.Samples
LastPoint = endData - 1 ' because the last point is invisible.
ReDim newArray(InResult.Samples) ' to store the results
unscaledData = InResult.DataArray(True)
' InResult.DataArray(False) provides
' integer data from -32768 to 32767.
' InResult.DataArray(True) provides real data
' in the same unit as the vertical scale of the trace.
Total = 0
       For i = 0 To LastPoint
       Total = Total + (unscaledData(i)) ^ 2
       Next
NewArray(0) = Sqr (Total / (LastPoint + 1)) Place the result in the
zeroth element.
OutResult.ValueArray(True) = newArray ' integer data output
```

The default waveform function script: explanatory notes

InResult.Samples is the number of points in the incoming waveform.
InResult.DataArray(Boolean) (or InResult1.DataArray or InResult2.DataArray) is the array of input data. If the Boolean is True you get scaled real data in the units of the trace. If the Boolean is false you get unscaled integer data in the range -32768 to + 32767.
The value of InResult.Samples is the total number of data in a trace. It is two more than the nominal value given on the screen. The first point DataArray(0), coincides with the left edge of the screen, apart from the wobble caused by the trigger-to-sample clock difference. If the trace length is nominally 500, the right edge of the screen coincides with DataArray(500), which is the 501st point. The last point, number 502, is just off the right of the screen, and is never seen. That is why the loop in the script runs only to endData - 1.
OutResult.Samples is the number of data in the output trace, and is set to be the same as the number of data in the input trace. If you set the output length less than the input length, you get a shorter trace, the remainder being made of zeroes. If you try to set the output values to something illegal, you may find that a part of the trace retains the values from a previous acquisition.

If you try to set something outside the bounds of an array, or you make some other error, or something overflows, or you ask for something impossible, such as log(-13), the instrument tells you the line number, and the nature of the problem. Other types of error may not be given the correct line number, for example, if "Next" or "End If" is omitted, because VBS does not know where it should have been.
UnscaledData is simply a copy of the input data set.
ReDim newDataArray(OutResult.Samples) defines an array of data for use as a scratch pad. Dim is short for Dimension, which is used in Visual Basic to declare a variable (even if it only has one element, in which case you omit the size of the array).
InResult.DataArray(False) means that the data are signed integers in the range -32768 to 32767. False is a Boolean value applying to the property Scaled. Scaled data are specified in the units of the vertical scale, such as volts. You get these by putting "True" instead of "False". If you want to make a section of the output trace invisible, you simply set the data values to full scale or bigger, top or bottom.
You can start with the unscaled data (False) as input, and then set the output data to scaled data (True), and you can go from scaled to unscaled. Using scaled data, an overflow will make a picture like this:



You can also start with True and convert to False, but in this case overflows will cause an error message.
Anything after a single quotation mark on a line will not be used by the instrument. This feature is intended for comments, for example

```
' This is a comment.
A = Amp * Sin(Omega * T) Calculate the output.
```

InResult.DataArray and OutResult.DataArray are only to be used as shown in the default scripts and in the example scripts: you cannot refer directly to individual elements of these arrays. You have to use your own arrays, in this example, unscaledData and newDataArray. You are not allowed to write statements like the following:

```
Y = InResult.DataArray (17)
OutResult.DataArray (257) = Z
```

Some parts of the default script must not be changed because they are a part of the interface. These are **highlighted** in the following script .

```
' TODO add your custom code here accessing OutResult and InResult
objects
' Here's a small example that just inverts the waveform.
OutResult.Samples = InResult.Samples
startData = 0
endData = OutResult.Samples
newNumPoints = endData - startData
ReDim newDataArray (OutResult.Samples)
unscaledData = InResult.DataArray (False)
        For i = 0 To endData - 1
        newDataArray (i) = - unscaledData (i)
     Next
OutResult.DataArray (False) = newDataArray _' only support raw data
```

The four highlighted quantities are parts of the interface. The names must be retained. Furthermore, InResult.Samples and InResult.DataArray are inputs, and their values cannot be changed. OutResult.Samples and OutResult.DataArray are outputs, and can be changed, but not directly through their individual elements.

The default parameter function script: explanatory notes

The default parameter script is similar to the default waveform script, but there are subtle differences.

First, the size of the data array is the same as the nominal value: you cannot use or see the extra two points. So "500 points" means just that: 500 points.

Second, the output looks like an array, but only element zero is currently used. You must copy your parameter result into newValueArray(0). As with the arrays of the Waveform Script, you cannot refer directly to elements of the input and output arrays. You may not write something like

```
OutResult.ValueArray (0) = P.
```

Note that the unit of the parameter is displayed as the same as the vertical unit of the trace, even if you have squared the data, for example, unless you change the unit yourself.

To find out how to edit a parameter script, click here.

The default parameter script is shown below.

```
' TODO add your custom code here accessing OutResult and InResult
objects
' Here's a small example that just inverts the waveform
numParam = InResult.Samples
ReDim newValueArray(numParam)
scaledData = InResult.DataArray
     For i = 0 To numParam-1
```

```
     newValueArray(i) = -scaledData(i)_' Change this to do something
useful.
     Next
OutResult.ValueArray = newValueArray 'only support raw data
```

Your parameter script should include something like this:
A. Do calculation to obtain your parameter value from the input data array.
B. newValueDataArray (0) = ParameterValue
C. OutResult.ValueArray = newValueArray
You can test this script using setup MeanDemoScriptApr2.lss.
You can edit scripts using Notepad, but you will not get any notification of errors.
You are **not** allowed to write OutResult.ValueArray(0) = MeanParameter.
InResult.DataArray and OutResult.DataArray are only to be used as shown in the default scripts and in the example scripts. You cannot refer to, or modify, any individual element in these arrays.

Scripting with VBScript

Separators

The two separators in VBS are the colon **:** and the single quotation mark .
Using the colon, you can place two or more statements on a line, for example:

```
   XMin = 0.0 : XMax = 800.0 : YMin = 0.0 : YMax = 600.0
```

There is also an implied separator whenever a new line is begun.
Using the quotation mark you can signify that the remainder of the line is a comment: non-executable material that is usually used to clarify the workings of the script. For example:

```
      RMSMax = 32767 / Sqr (2)    ' RMS of the largest sinusoid that
        can be
                                  ' fitted into the screen in unscaled mode.
```

To continue a comment on to another line, another quotation mark is required on the new line.

Variable Types

VBS supports the following variable types:

Integer	signed 16 bit value in the range -32768 to 32767
Long	signed 32 bit value in the range -2^{31} to $+2^{31}$ - 1
Single	real number or floating point number
Double	real number or floating point number
Boolean	Boolean or logical value
String	string of characters

They are supported when making comparisons using real numbers, beware of testing for equality, because of rounding errors. It may be better to apply a tolerance band. For Boolean, integers and strings, equality is valid.
You can use variables in VBS without declaring the type. The context may force an implicit type assignment. For example, if the result of a calculation is of a different type from the defined type, the type may be changed. Always set out calculations in such a way that type changes will not affect the final result in an undesirable or unpredictable way. If you want to change the type of a variable or a result, use a conversion function that will show others what you intend to happen. The conversion functions are CDbl, CInt, CLng, CSng, CStr.

Variable Names

Upper and lower case have no significance in VBS, either in variable names or in keywords (the names reserved by the system), but it is a good idea to be consistent about the spelling of a variable name to avoid confusion. At least 36 characters may be used in a variable name. These can include any combination of alphabetic and numeric characters, and the underscore character. No other punctuation character may be used in a variable name.

Do not use any of the following characters in a variable name:

```
! @ & $ # ? , * . { } ( ) [ ] = + - ^ % / ~ < > : ;
```

Just use alphanumerics and underscore, for example: Example_Name

If you have to introduce constants, give them sensible names, just like variables. For example, do not write:

```
_If RMS < 23169 Then OutputY = Y
```

Its meaning may not be obvious to someone else.

It is better to write something like this:

```
     FullScale = 32767
     RootTwo = Sqr (2.0)
     MaxRMS = FullScale / RootTwo
      . . . . .
          If RMS < MaxRMS Then . . . . .
```

But to keep your scripts fast, leave definitions like this outside your loops.

General usage

Note that white space has no effect, so you can introduce spaces for clarity, except of course within variable names, function names and other keywords. Indenting control statements can be a great help in understanding a program. For example:

```
     For K = Kstart To Kstop
     X = K * Sqr (3)
          For N = NStart To Nstop
          Y = N * N
               If Y < FullScale Then
              . . . . . .
              . . . . . .
               End If   ' End of main calculation
          Next   ' End of N loop
     Next   ' End of K loop
```

If a section becomes very long, you could provide the end with a comment, to show where it comes from.

Arithmetic Operators

As with most other languages, the arithmetic operators are used as follows:

^	Exponentiation	A ^ B = A^B = A raised to the power B
/	Division	A / B = A divided by B
\	Integer division	A \ B = A divided by B, truncated to next integer below
*	Multiplication	A * B = A multiplied by B
+	Addition	A + B = B added to A
-	Subtraction	A B = B subtracted from A

Please Note the Following:

- If there is any possibility that you will be taking the exponent of a negative number, make sure to trap any possible errors arising from such operations as trying to take the square root of a negative number. Logs of negative numbers are forbidden also.
- If there is any possibility that you will be dividing by zero, make sure to trap this.
- There are two ways of dealing with these types of problem. One is to prevent it happening by making suitable tests before the calculation is performed. The other is to let it happen, and use an error handling routine. This will be discussed later.
- Normally in VBScript you will know the range of the data, since all the incoming data are, by definition, integer (unscaled data) or real (scaled data), and they must fit into the screen of the instrument.

Results of Calculations

Sometimes you may see a statement like this:

```
A = A * A * (Cos (A) + Sin (A) )
```

The program takes the quantity represented by A and performs all of the following operations, using that original value:

1. Multiply A by itself.
2. Calculate the cosine of A.
3. Calculate the sine of A.
4. Add the cosine and the sine together.
5. Multiply that result by the square of A.

   At this point, the quantity represented by A has not been changed. Only at the end of the calculation is the final value placed in the memory location labeled A.

   Note that you can write more than one statement on a line, separated by colons, like this:

```
A = B * Cos (34 * Theta) * Sin (55 * Theta) : B = A * A + Z * Z
```

Order of Calculations

Operations are performed in the following order:

1. Contents of brackets
2. Exponentiation
3. Division and multiplication
4. Addition and subtraction

If there is any doubt as to how the calculation will be done, use brackets. These will also make the order of the calculations clear to any reader of the program, which is desirable if you are to give it to a customer, who will want to know what was intended.
Here are some examples of the uses of brackets:
Brackets are worked out before any other operations are performed.
Use brackets to force the result you want, and also to clarify a calculation.

A	1 1 1 1 1 1 1 1	255	0 1 0 1 1 0 1 0	90	(B OR C) AND (D OR E)
B	1 1 1 1 0 0 0 0	240	0 0 0 0 0 0 0 0	0	B OR (C AND D) OR E
C	1 0 1 0 1 0 1 0	130	1 1 1 1 1 0 1 0	250	B OR (C AND (D OR E))
D	0 1 0 1 0 1 0 1	85	0 1 0 1 1 1 1 1	95	((B OR C) AND D) OR E)
E	0 0 0 0 1 1 1 1	15			
F	0 0 0 0 0 0 0 0	0			

A	7	315	A * B * (C / D) * E * F
B	6	8.75	A * B * C / (D * E * F)
C	5	35	A * B * (C / (D * E)) * F
D	4		
E	3		
F	2		

Check these results to see whether any errors, deliberate or otherwise, have been introduced.

VBS Controls

```
Do                          Do                          Do
. . . .                     . . . .                     . . . .
Loop                        Loop Until . . . .          Loop While
Do Until                    Do While                    Exit Do
. . . .                     . . . .
Loop                        Loop
. . . .                     . . . .

For . . . Next                        Exit For
GoTo__This is not allowed in instrument VBS.
If . . . . Then . . . . _' On one line__
If . . . Then
ElseIf . . . Then
End If
```

```
If . . . Then . . . End If__
If . . . Then . . . Else . . . End If
Select Case
. . . .
End Select
While
. . . .
Wend
```

Choose the construction that best satisfies the requirements of speed and clarity.
The construction GoTo LabelledStatement is available in many languages, including VBA, but not in VBS. ***GOTO*** is not allowed in VBS.

IF . . . Then . . . Else . . . End If

A very simple example:

```
     If A >= 0 Then B = Sqr (A) 'Take the square root of A if A is not
negative.
     If A + B < C + D Then E = F : G = H_ 'No End Is needed if all on
one line.
```

If you need to perform a longer procedure, make this construction:

```
     If A >= 0 Then
     B = Sqr (A)
     C = 32766 * Sin ( TwoPi * B / PeriodOfSinusoid)
     End If     ' End If is needed to terminate the construction.
```

The If statement is very often used with the following Boolean expressions:

A > B	A is greater than B
A >= B	A is greater than B or equal to B
A = B	A is equal to B
A < B	A is less than B
A <= B	A is less than B or equal to B
A <> B	A is not equal to B

These statements are not like the usual program statements, such as A = B. These statements are Boolean (logic) statements, which can take the values True or False. You may even see things like "If A Then B", which means that if A is True, B gets done.
In the first example, if A is negative, we might want to write something like this:

```
     If A >= 0 Then
     B = Sqr (A)
     Else
     B = 0
     End If
```

and in fact you can make some very complex constructions using If, as in the examples below:

```
     If A < 0 Then
         If A < - 1 Then
```

```
        Z = 17
        Else_
        Z = 31
        End If
    Else_
        If A > 3 Then
        Z = 63
        Else
        Z = 127
        End If
    End If
    If A > 0 Then
        If B > 0 Then
        Z = Y
        End If
    End If
```

This is equivalent to:

```
    If ( (A > 0) And (B > 0) ) Then
    Z = Y
    End If
```

Summary of If . . . . Then . . . . Else

```
    If Boolean Then AnyVBScriptingOnOneLine
    If Boolean Then
    AnyVBScriping
    End If
    If Boolean Then
    AnyVBScripting
    Else
    AnyOtherVBScripting
    End If
```

If you write a list like this, all the Booleans will be evaluated, whether you want that or not:

```
    If A > 9 Then VBScripting1
    If A > 7 Then VBScripting2
    If A > 6 Then VBScripting3
    If A > 4 Then VBScripting4
    If A > 3 Then VBScripting5
    If A > 1 Then VBScripting6
```

Be very careful when testing for equality. There will be no trouble with Integers, Long Integers, and Strings, but Real numbers are different. Because they have so many significant digits, values that should be equal, may differ minutely after a computation. It is safer with Real numbers to test using a tolerance band.

If you find that you are building up a rather complicated set of Ifs, you might want to consider the Select Case construction.

Select Case

This is a very powerful construction, which is also easy to understand when written out. It is best for Integers and Strings, where exact values are always obtained. Here is a simple example:

```
Select Case K
     Case 7 : Y = 6 : Z = 3
     Case 7 : Y = Sqr (Sin (A) ) : Z = Sqr (Cos (A) )
     Case N : Z = Y + X
     Case Else :
End Select
```

Case N assumes that the value of N has already been set. Case Else is included to cover other cases, whether foreseen or not. It should always be included.
You can also provide lists of values.

```
Select Case K
     Case 1, 2, 3, 5, 8, 13 : Y = 55 : Z = 89
     Case 4, 9, 16, 25, 36 : Y = Sqr (Sin (A) ) : Z = Sqr (Cos (A)
)
     Case 7, 15, 31, 63, 127 : Z = Y + X
     Case Else : Z = 3
End Select
```

Case N assumes that the value of N has already been set. Case Else is included to cover other cases, whether foreseen or not. It should always be included.
Select Case is much neater than a string of Ifs and Elses, but remember: You cannot use Select Case unless you are sure of exact equality, which allows you to compare integers and strings only. You cannot put Case > 5, for example.

Summary of Select Case . . . . End Select

```
SelectCase VariableName
     Case Alist : VBScriptingA
     Case Blist : VBScriptingB
     . . . .
     Case Else : VBScriptingElse_ VBScriptingElse can be empty.
End Select
```

Do . . . Loop

This construction is useful when you do not know at programming time how many times the loop will be executed. Here are some examples:

```
Do
AnyVBSCalculation
Loop Until D > Pi
Do Until Z < Y
AnyVBSCalculation
Loop
Do
AnyVBSCalculation
```

```
Loop While D <= Pi
Do While Y >=Z
AnyVBSCalculation
Loop
```

These constructions enable you to make the test before or after the calculation. If the test is made before, the calculation might not even be done one time (if the condition for terminating were already true). With the condition at the end, the calculation is done at least one time.
Sometimes you might want to exit the loop from somewhere inside: for example, if some kind of problem is looming, such as the logarithm of a negative number.

For this case, you can use **If . . . . Then Exit Do**.
To make a pause of 10 seconds you can write:

```
NewTime = Timer + 10.0
Do Loop Until Timer >= NewTime
```

Where Timer is a clock function in the PC, which has a resolution of one second.

While . . . Wend

This is similar to Do While . . . Loop. You can write things like:

```
While ( (A > 2) And (C < 92677663) )
AnyVBCalculation
Wend
```

For . . . Next

Sometimes you know, or you think you know, the number of times that you want to do a job. For this case a For loop is ideal, especially when you have an array of numbers to work with.
Examples:

```
For K = 0 To Total
HistogramBin (K) = 0
Next
Omega = TwoPi / Period
For N = 0 To Period
Y (N) = A * Sin (Omega * N)
Next
```

Be careful about changing the counting variable in any loop. You can do this to terminate the loop early (but **Exit For** is better), but you could also prevent it from terminating at all.
For emergency exit, you can use **Exit For**. For example:

```
For K = 0 To Total
     If HistogramBin(K) = 0 Then Exit For
AnyVBScripting
Next
```

It is possible to make a For loop with steps greater than 1, as in the following example in which K takes the values 3, 7, 11, 15, ... 83.

```
For K = 3 To 82 Step 4
AnyVBScripting
Next K
```

You may place loops inside one another (nested loops), but they must all use different control variables. Example:

```
For K = 0 To N
VBScriptingK
     For L = - 7 To 17
     VBScriptingL
          For M = S To T
          VBScriptingM
          Next
     Next
Next
```

VBS keywords and functions

Note: The following Items in *italics* do not apply to this instrument.

+	Add two values or concatenate two strings.
-	Subtract two values.
*****	Multiply two values.
/	Divide two values.
****	Divide two values to obtain an integer result
Abs	Make absolute value.
Asc	Make ASCII value of a character.
Atn	Make $\tan^{-1}$ of a value. Result in range from $-\pi/2$ to $+\pi/2$ radians.
Cdbl	Convert a value to double precision floating point.
Chr	Create a character from an integer in range 0 to 255.
Cint	Convert a value to nearest integer in the range -32768 to +32767
Clng	Convert a value to nearest long integer in the range -2^{31} to $+2^{31} - 1$.
Close	*Close a file.*
Cos	Make the cosine of an angle expressed in radians.
Csng	Convert a number to single precision floating point.
Cstr	Convert a variable to a string.
Exp	Raise e to the power of the input.
Get	*Get a value from a file.*
Input	*Get some ASCII data from a file.*
Instr	Find the position of a string in a longer string.
Int	Convert to nearest integer below the input value.
Left	Take some characters at the left end of a string.
Log	Take the natural logarithm of a positive value.
Ltrim	Remove spaces at the left end of a string.
Mid	Take or insert some characters in a string.

Mod	Take the modulus of a value in terms of another value.
On Error	Take some action if an error occurs.
Open	*Open a file.*
Print	*Send some ASCII data to a file.*
Put	*Send some data to a file.*
Randomize	Randomize Timer re-seeds the pseudo-random number generator.
Read	*Read from a file.*
Right	Take some characters at the right end of a string.
Rnd	Make a random real number in the range from 0.0 to 1.0
Rtrim	Remove spaces from right hand end of a string.
Sin	Make the sine of an angle expressed in radians.
Sqr	Make the square root of a positive number.
Str	Make a string from a numerical value.
Timer	Time since midnight in seconds, with a resolution of one second.
Trim	Remove leading and trailing spaces from a string.
Val	*Get the ASCII value of a string beginning with numerical characters.*

Other VBS Words

Const	Define a constant value.
Dim	Dimension a variable.
Redim	Dimension a variable again.
Boolean	Boolean variable
Double	Double precision real variable.
Integer	Integer in the range -32768 to + 32767
Long	Long integer in the range -2^{31} to + 2^{31} - 1
Single	Single precision real variable
String	String variable
And	Logical AND
Or	Logical OR

To make a bit-by-bit comparison, logical constructions can be used with variables, as in A and B, or with tests such as If A > B Then . . .

Functions

These are mainly of the form C = F (A), where A is the argument, or input to the function.

Abs	Abs (A) calculates the absolute value of an integer or a real number, so the result is always positive or zero. A can be any number in the range of the VB system.
Atn	Atn (A) calculates the angle of which A is the tangent. Because infinitely many angles can have the same tangent, the output of Atn always lies in the range minus π / 2 to plus π / 2. The input can be any positive or negative value in the range of the VB system.
CDbl	CDbl (A) calculates a double precision real variable, equal to A.
CInt	Cint (A) calculates the integer value nearest to A, which can be any acceptable VBS number. Cint (-7.4) = -7. Integers are signed 16-bit values in the range -32767 to + 32767.
CLng	CLng (A) calculates the nearest long integer to the value A. Long integers are signed 32-bit values in the approximate range -21.5 M to + 21.5 M.
Cos	Cos (A) calculates the cosine of any integer or real number, giving an output that is never greater than plus one or less than minus one.
CSng	CSng (A) calculates a single precision real variable equal to A.
Exp	Exp (A) calculates the value of e^A.
Int	Cint (A) calculates the integer value next below A, which can be any acceptable VBS number. Int (-7.4) = -8.
Log	Log (A) calculates the natural logarithm (to base e), of any acceptable VBS number greater than zero. A negative number or zero will create an error. To calculate $Log_{10}(A)$, use $Log_{10}(A) = Log(A) / Log(10)$
Mod	A Mod (B) calculates the modulus of A, which is the remainder after A has been divided by B. 34 Mod 8 = 2. 34 Mod 55 = 0. -34 Mod 13 = -8. 21 Mod -8 = 5.
Randomize	Calculates a new seed for the pseudo-random number generator. Randomize Timer uses the real-time clock for this purpose.
Sin	Sin (A) calculates the sine of any integer or real number, giving an output that is never greater than +1 or less than -1.
Sqr	Sqr (A) calculates the square root of any integer or a real number that is not negative. If A is negative, an error will occur.
Timer	Time since the previous midnight in whole seconds.

Hints and Tips for VBScripting

Set the trigger to Single or Stopped if you need to do a lot of editing: it is faster.
Before starting a script, remove any existing scripts that you do not need. This is because errors in an existing script will give you error messages, even if your current script is perfect. And an existing good script may develop a fault if you change the setup. For example, you might change the vertical scale or the memory length and get an overflow if you did not guard against it in the script.
When starting a script, make sure that you have chosen the right kind: function or parameter. You can get some very frustrating problems if you are in the wrong mode. You can cut and paste the VBS statements if you discover this error.
If your calculation requires a long memory, development might be quicker if you test the principles on a shorter trace at first.
Note that the pseudo-random number generator is reset at the start of a script. If you want a different set of pseudo-randoms every time, put Randomize Timer in the program, to be run once, before any pseudo-randoms are generated. You can use this instruction to re-seed the generator at any time during execution.
Do not put the final statement in a loop, hoping that you can see a progressive result as some parameter changes. No output will be seen on the screen of the instrument until the script has been completely run and quitted, so only the final result will appear. If the loop runs many times, you will think that the scope has hung up.
If you want a For loop, end it with "Next" and not "Next X."
If you make a script that takes a long time to run, go back to the default setup before quitting or powering down, or you will have a long wait next time you power up.
Always use a recursive calculation when this will speed things up.
Keep everything outside a loop that does not have to be inside, to speed things up.
Make your scripts clear, not only by indenting and commenting, but by structuring neatly as well.
Sometimes it might be easier to develop your script in Excel VBA (remembering that VBA is not identical to VBS), so that you can display intermediate results. If you do this, note that you can read from a cell or write to it using statements like these:

```
A = Worksheets("Sheet1").Cells(Row, Column).Value
Worksheets("Sheet1").Cells(Row, Column).Value = B
```

Note that in VBS, after you have corrected an error and clicked on "Apply," the error message may go on flashing for a few seconds, or a few acquisitions, before being erased. Look for the "Script OK" message. Be patient before assuming that you still have a bug.
If your calculation requires data to be used at some other horizontal positions than their original ones, make sure that your algorithm does not try to send data to non-existent array positions, that is, beyond the edges of the screen. You may have to truncate your output trace, as happens with the instrument's Enhanced Resolution and Boxcar functions.
No output will emerge from a script until you press Apply.
No output will emerge from a script until it has received an input. This includes the case where the input data are not used in calculating the output data. So you must have had at least one acquisition before you see anything.
Because you can introduce undeclared variables at any point in a calculation, VBS does not check your spelling.
You can make a portion of a trace disappear if you set the values to 32767 or -32768.

You can highlight a section of a trace by making the points alternately too high and too low by a suitable amount. Providing the memory length is not too short, the compaction algorithm will give the effect of a thicker trace.
The lengths of the output trace and the input trace need not be the same. You can even make the output trace longer than the input trace, but you will need to unzoom it to see it all. This feature can be used to avoid compaction problems with non-linear horizontal scales. It can also be used to show several versions of a function at the same time, without having to set up a separate script for each one.
If your program structure is complicated, consider typing all the IFs, ELSEIFs, ENDIFs, FORs, NEXTs, etc and then clicking Apply. You wont get any output, but the system will tell you if the structure is acceptable. Then you can insert the actual program statements.
Always try to make the script as independent as possible of variables such as V/Div, T/Div, and memory length, unless that would make it harder to understand. If so, give some values as examples, and explain how the script would have to change if the variables changed.

Errors

The instrument VBS tries hard to help you when errors occur.
Errors may be of two main types:

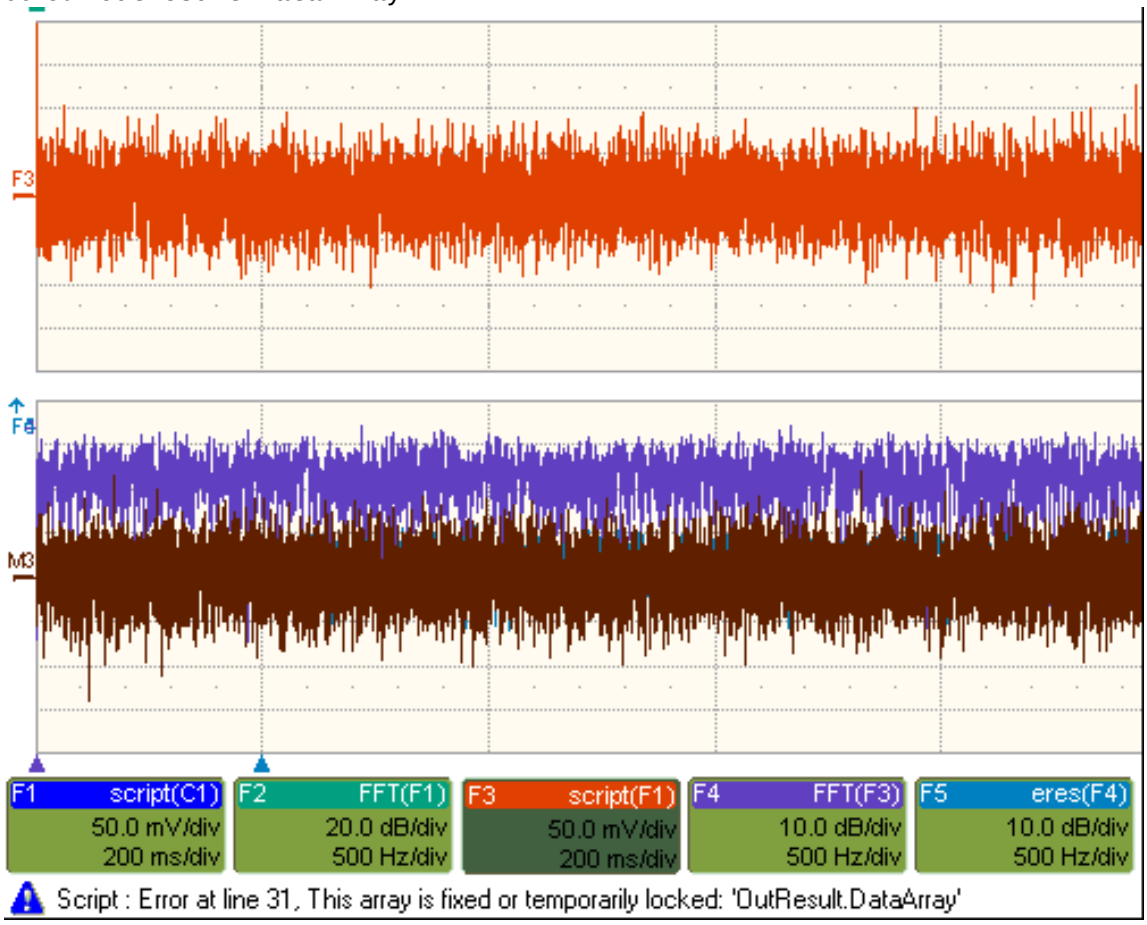
- The script may not be usable because the interpreter cannot construct a logical structure from it.
- The script may be usable, but may fail while running because an incomputable function has been requested.

Sometimes the line number given for an error is wrong. This can happen when the error is of this general type:

```
Missing "Next" Missing "End If"
Extra "Next" Missing "Until" etc.
```

This happens because VBS cannot know where you should have put the statement.
If at some point during the calculation of an output array, a value goes outside the allowed range, the calculation will stop, and you will see the new values up to the point of the stoppage. To the right of that point, the trace will display the previous values. In fact, if you deliberately recalculate only a part of a trace, you can have a mixture of new and old values.
The following figure shows an error message type seen if a calculation has tried to set a value outside the range -32768 to +32767. It takes extra time to guard against this. However, unless you are sure that it will not happen, some kind of check is needed.

In the following example, the red trace has gone outside the allowed range at the beginning, resulting in the message at the bottom of the instrument screen: This array is fixed or temporarily locked: `OutResult.DataArray`.



Error Handling

Note that the construction OnError GoTo Label: is not allowed in VBS. In fact no GoTos or labels are allowed. Therefore there is no way for you to provide handlers to deal with errors and exceptions. You must be aware of all possibilities at all points in your program, and you must either be certain that errors will not occur, or you must take action to ensure that they do not.

Examples:

Sqr	You cannot take the square root of a negative number.
Log	You cannot take the log of zero or of a negative number.
A / B	You cannot divide by zero.
Array	You cannot use an index outside the bounds of an array.
Size	Unscaled data cannot go outside the range -32768 to 32767.

If there is any possibility that any of these might occur, take steps to deal with this before it can happen.

For example, you may write some kind of generator of pseudo-random statistical values. If these belong to a distribution that in principle has an infinite range, or a finite range which is wider than the signed 16-bits allowed, check each value. If a value falls outside the range, you could set it to the maximum or generate another example.
You can, however, use one of the following:

```
On Error Resume Next
```

Followed by some code that may make some attempt to deal with the problem, or at least to allow execution to continue.

```
On Error GoTo 0
```

This cancels On Error Resume Next_

Speed of Execution

To maximize the speed of execution of a script, the most important thing you can do is to minimize the number of operations that are performed inside loops. Anything done once only is unlikely to be an important source of delay. Please note that VBS is much slower than the internal computations of the instrument, so do everything you can to save time, unless time is irrelevant to the application.
Using an array element takes longer than using a single variable. Here is an example:

```
     For K = 1 to Total
          If X (K) > X (K - 1) Then
          Y = Cos (X (K) ) * Sin (X (K) ) * Sqr (X (K) )
          End If
     Next
```

To do the same thing we could also write this, using the index only once:

```
OldXK = X (0)
     For K = 1 To Total
     XK = X (K)
          If XK > OldXK Then
```

```
          Y = Cos (XK) * Sin (XK) * Sqr (XK)
          OldXK = XK
          End If
     Next
```

VBS runs slower than the "internal" calculations, because the scripts are interpreted. This could be serious for calculations where many operations are needed on each sample, such as convolution, correlation, and long digital filters.

Scripting Ideas

What can we do in a VBS script that we cannot do with the normal instrument functions? Here are some possibilities.

- Create a new function that acts on waveform values.
- Create a new parameter.
- Create a new form of non-linear vertical scale.
- Create a new form of non-linear horizontal scale.
- Move some or all data horizontally, including reflections.
- Combine data to form digital filters.
- Show several function results side by side.
- Show several function results interleaved.

You can even create output data that are not related to the input. The output data need not even be in the same domain as the input data, because the system treats them as pure numbers. So you can create your own transforms into the frequency domain, for example.

Debugging Scripts

Until we have integrated a more comprehensive debugger for VBScript there is a workaround.

1. Download the Windows Scripting Debugger for Windows 2000 from here: http://download.microsoft.com/download/winscript56/Install/1.0a/NT45XP/EN-US/scd10en.exe
2. Enable JIT (Just In Time) debugging by setting the following registry key HKCU\Software\Microsoft\Windows Script\Settings\JITDebug = to 1 (DWORD value)
3. Place a **Stop** statement in your script.

Now, when the Stop statement is executed the debugger will open and allow single-stepping, variable examination, etc.

Using VBA or Visual Basic to debug VBScripts is not recommended since the language syntax for these three variants of basic is slightly different.

Horizontal Control Variables

InResult.HorizontalOffset	Double precision	Time shift of input waveform on grid in units of horizontal scale
OutResult.HorizontalOffset	Double precision	Time shift of output waveform on grid in units of horizontal scale
InResult.HorizontalPerStep	Double precision	Time between successive samples in the input waveform
OutResult.HorizontalPerStep	Double precision	Time between successive samples in the output waveform
InResult.HorizontalUnits	String	Horizontal units of input waveform
OutResult.HorizontalUnits	String	Horizontal units of output waveform
InResult.Samples	Integer	Number of samples in input waveform

Vertical Control Variables

InResult.VerticalOffset	Double precision	Vertical shift of input waveform on grid
OutResult.VerticalOffset	Double precision	Vertical shift of output waveform on grid
InResult.VerticalPerStep	Double precision	Difference between successive possible levels in the input waveform memory
OutResultVerticalPerStep	Double precision	Difference between successive possible levels in the output waveform memory 1 / 65536 of vertical full scale
InResult.VerticalResolution	Double precision	Difference between successive possible physical levels in the input waveform
OutResultVerticalResolution	Double precision	Difference between successive possible physical levels in the output waveform 1 / 256 of vertical full scale for channel waveforms 1 / 65536 of vertical full scale for math waveforms
InResult.VerticalUnits	String	Vertical units of input waveform
OutResult.VerticalUnits	String	Vertical units of output waveform

List of Variables Available to Scripts

FirstEventTime([out, retval] VARIANT * pVal); FirstEventTime([in] VARIANT newVal);
LastEventTime([out, retval] VARIANT * pVal); LastEventTime([in] VARIANT newVal);
UpdateTime([out, retval] VARIANT * pVal); UpdateTime([in] VARIANT newVal);
Details([in] BSTR strDetailsIID, [out, retval] VARIANT * pVal);
Status([out, retval] VARIANT * pVal); Status([in] VARIANT newVal);
ExtendedStatus([out, retval] VARIANT * pVal); ExtendedStatus([in] VARIANT newVal);
StatusDescription([out, retval] BSTR * pVal); StatusDescription([in] BSTR newVal);
DataArray([in, defaultvalue(TRUE)] BOOL arrayValuesScaled,
    [in, defaultvalue(LEC_ALL_DATA)] int numSamples,
    [in, defaultvalue(0)] int startIndex,
    [in, defaultvalue(1)] int sparsingFactor,

```
    [out, retval] VARIANT *pArray);
DataArray([in, defaultvalue(TRUE)] BOOL arrayValuesScaled,
    [in, defaultvalue(LEC_ALL_DATA)] int numSamples,
    [in, defaultvalue(0)] int startIndex,
    [in, defaultvalue(1)] int sparsingFactor,
    [in] VARIANT array);
HorizontalUnits([out, retval] BSTR *pVal); HorizontalUnits([in] BSTR newVal);
Samples([out, retval] int *pVal); Samples([in] int newVal);
HorizontalResolution([out, retval] double *pVal); HorizontalResolution([in] double newVal);
HorizontalPerStep([out, retval] double *pVal); HorizontalPerStep([in] double newVal);
HorizontalOffset([out, retval] double *pVal); HorizontalOffset([in] double newVal);
Sweeps([out, retval] int *pVal); Sweeps([in] int newVal);
HorizontalVariances([out, retval] int *pVal); HorizontalVariances([in] int newVal);
HorizontalVarianceArray([out, retval] VARIANT * pArray);
HorizontalVarianceArray([in] VARIANT array);
HorizontalFrameStart([out, retval] double *pVal); HorizontalFrameStart([in] double newVal);
HorizontalFrameStop([out, retval] double *pVal); HorizontalFrameStop([in] double newVal);
VerticalFrameStart([out, retval] double *pVal); VerticalFrameStart([in] double newVal);
VerticalFrameStop([out, retval] double *pVal); VerticalFrameStop([in] double newVal);
VerticalResolution([out, retval] double *pVal);  VerticalResolution([in] double newVal);
VerticalPerStep([out, retval] double *pVal); VerticalPerStep([in] double newVal);
VerticalOffset([out, retval] double *pVal); VerticalOffset([in] double newVal);
VerticalMinPossible([out, retval] double *pVal); VerticalMinPossible([in] double newVal);
VerticalMaxPossible([out, retval] double *pVal); VerticalMaxPossible([in] double newVal);
VerticalUnits([out, retval] BSTR *pVal); VerticalUnits([in] BSTR newVal);
```

Communicating with Other Programs from a VBScript

The ability of The instrument to communicate with other programs opens up immense possibilities, both for calculation and for graphics, making the assembly of reports relatively simple.

Communicating with Excel from a VBScript

Although there are direct instrument calls to Excel and other programs, you may wish to do this from a VBScript. Here is an example:

```
OutResult.Samples = InResult.Samples
startData = 0
endData = OutResult.Samples
ReDim newData(OutResult.Samples)
USD = InResult.DataArray(False)
LastPoint = endData - 1
Set ExcelApp = GetObject(,"Excel.Application")
ExcelApp.Visible = True
ExcelColumnA = 2      'Column where the data will appear in Excel
ExcelRow = 10         'Row where the data will start
ExcelColumnB = 3      ' Column where the output data will appear in
Excel
     For K = 0 To LastPoint
```

```
     ExcelApp.ActiveSheet.Cells("ExcelRow + K, ExcelColumnA ") = -
USD(K)
     Next
Once the data are in Excel, any Excel functions can be applied to the
data. The results can be returned to the VB script.
     For K = 0 To LastPoint
     NDA(K) = ExcelApp.ActiveSheet.Cells("ExcelRow + K, ExcelColumnB")
     Next
```

Transferring data cell by cell is very slow, so it is better to do a block transfer.

Calling MATLAB from the Instrument

Note: Load MATLAB version 6.5 just as you would on a PC. Once loaded, open MATLAB from the desktop, then close it again before attempting to open it from the instrument application. Restarting the application is necessary in order to update the system registry.

MATLAB can be directly called from the instrument in two ways:

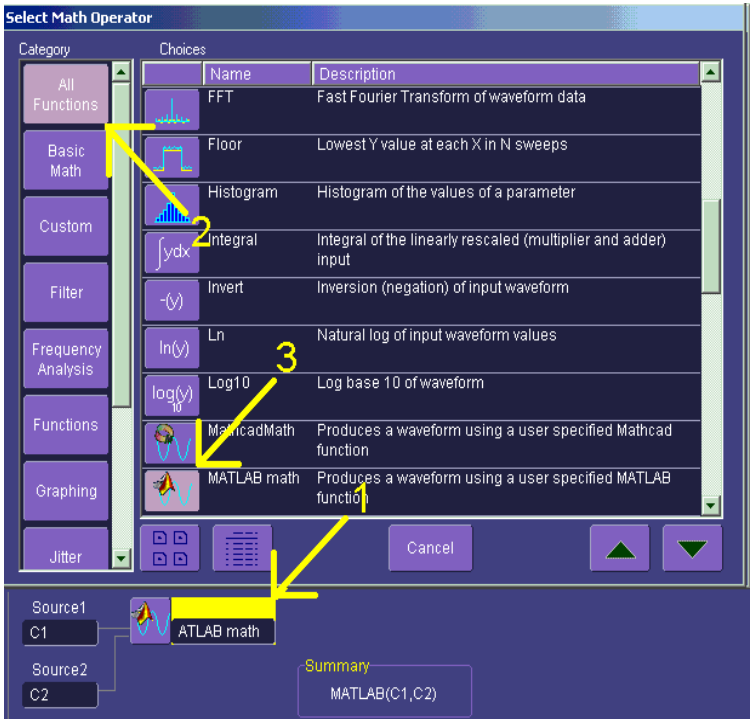
Using a function	F1 through Fx [The number of math traces available depends on the software options loaded on your scope. See Specifications.]	MATLAB returns a waveform
Using a parameter	P1 through Px	MATLAB returns a parameter

In both cases, one call to MATLAB can use two separate waveforms as input, providing much greater computing power than is available by calling MATLAB from a VBScript.

Note: If you do not place a semicolon ";" at the end of a line, MATLAB will show the calculated value in the result window, significantly slowing down the processing rate. This feature is best kept for diagnostics.

How to Select a Waveform Function Call

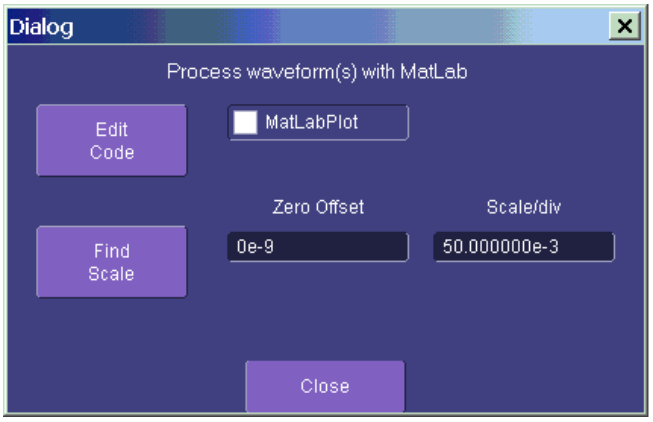
The MATLAB Waveform functions are selected from the **Select Math Operator** menu. Please note that once you have clicked on "MATLAB Wave" there will be a slight pause before MATLAB starts.



Source 1 and Source 2 are the waveforms that MATLAB will use.

The MATLAB Waveform Control Panel

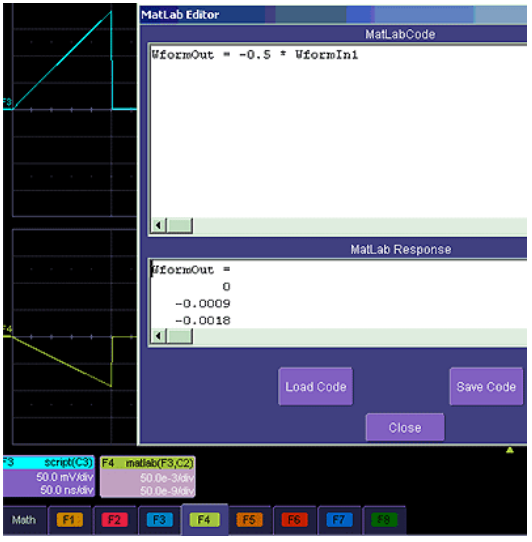
Once you have invoked a MATLAB waveform call, you will see the zoom dialog at the right of the screen. Touch the MATLAB tab to see a panel like this:



Touch **Find Scale** to make your output fit the grid, or use the text boxes to choose a scale.

MATLAB Waveform Function Editor - Example

By touching **Edit Code**, you can reach the MATLAB Editor where you will see the default waveform function. If you are familiar with MATLAB, you might prefer to launch MATLAB and create a MATLAB function that performs your task. Your program in the instrument could then be a one-line call of your MATLAB function.

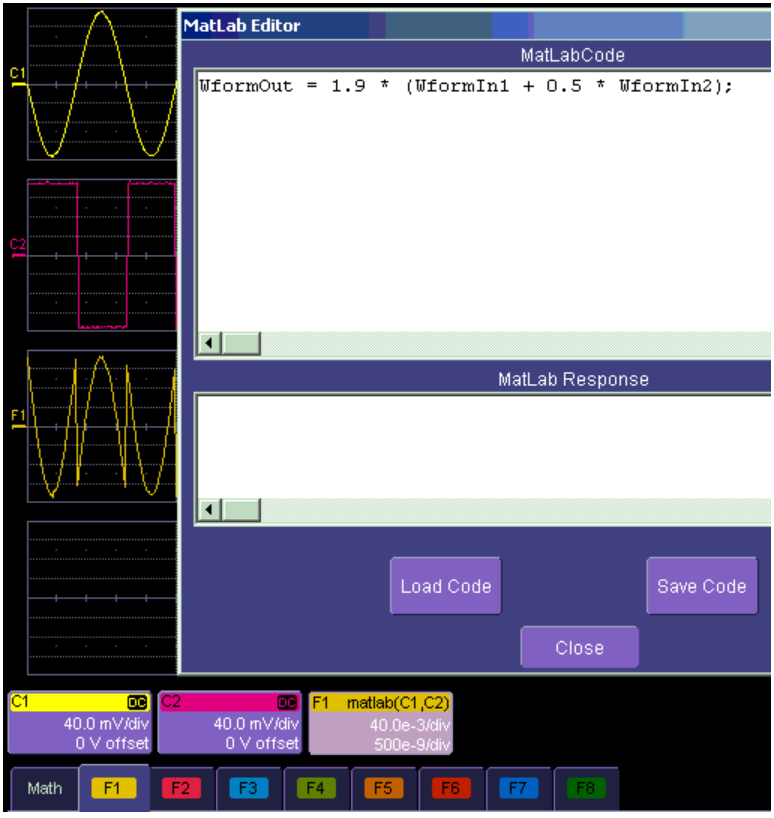


This is the default waveform function, with one important change – the semi-colon (;) has been removed from the end of the line. If the semicolon is present, your function will run much faster, because the output values will not be shown in MATLAB Response. With a long waveform, the time needed to display it could be quite long. The response values can be useful during development and debugging. Any line without a semicolon will produce a visible MATLAB Response.
From this panel you can save your code, load a previous code, and edit your function. A powerful feature of MATLAB is that you can refer to an entire waveforn as a vector. The two input waveforms are WformIn1 and WformIn2, while the output is WformOut. You can also refer to individual samples, such as WformIn1(34), and sequences of samples, such as WformIn(55:89)

You can write statements such as these:

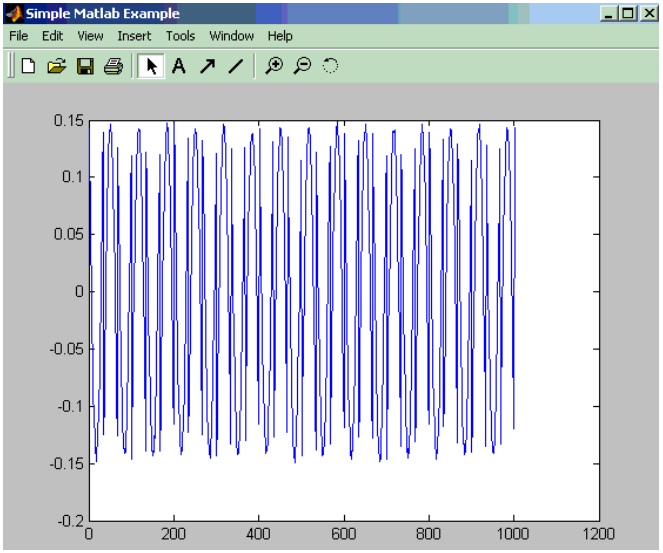
```
WformOut(5) = WformIn(5)
WformOut(89) = WformIn(144)
WformOut(34:55) = WformIn(34:55)
WformOut(233:377) = WformIn(100:244)
```

This very simple example adds a rescaled copy of Channel 2 to a copy of Channel 1, and then rescales the result:

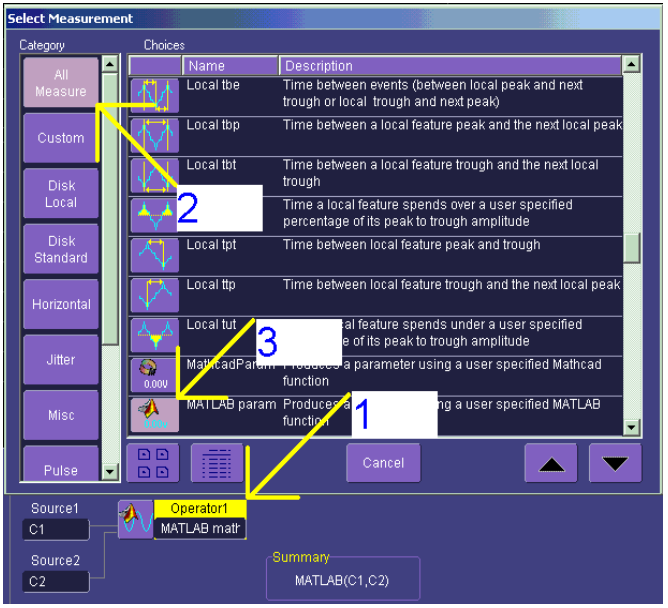


MATLAB Example Waveform Plot

If you touch the **MATLAB Plot** checkbox you will see a MATLAB plot like this one:



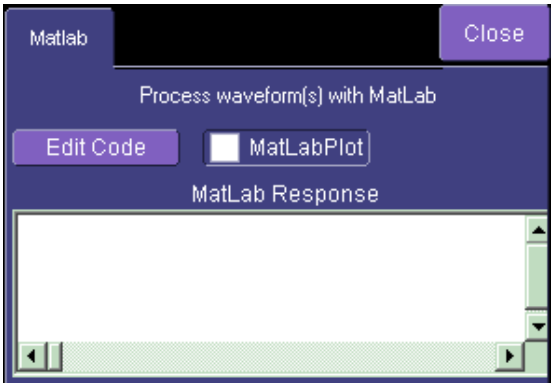
How to Select a MATLAB Parameter Call



Menu position for MATLAB parameter call in **Select Measurement** menu.

The MATLAB Parameter Control Panel

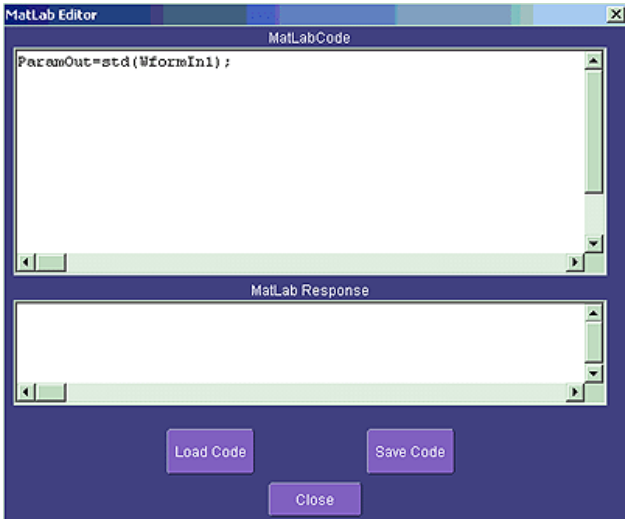
Once you have invoked a MATLAB parameter call, a mini-dialog to the right of the main dialog will appear:



You can touch the **MATLAB Plot** checkbox if you want to see a plot in MATLAB as well as getting a result in the instrument.

The MATLAB Parameter Editor

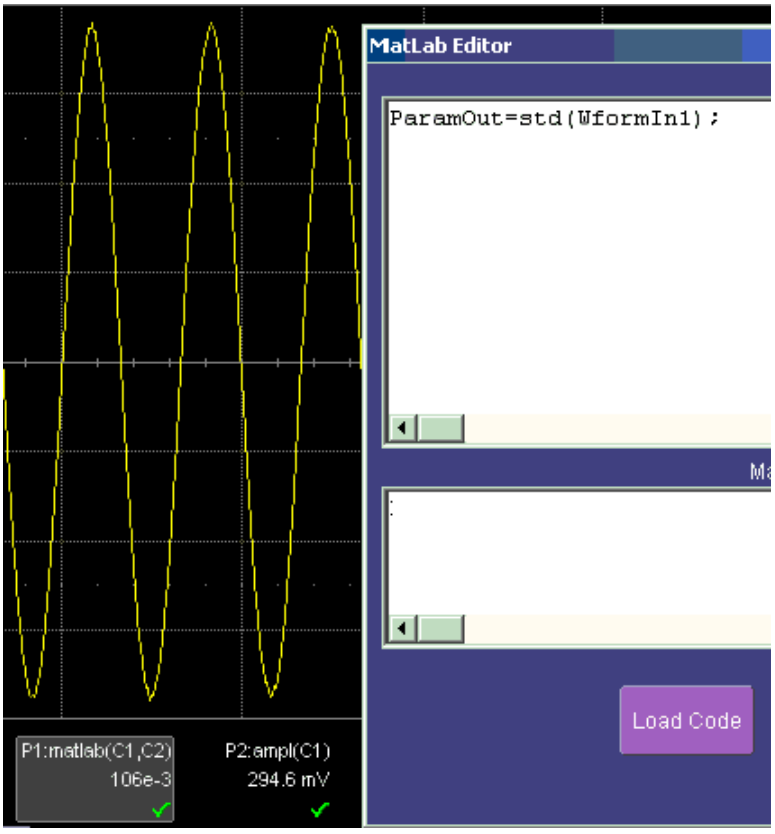
By touching **Edit Code**, you can reach the MATLAB Editor:



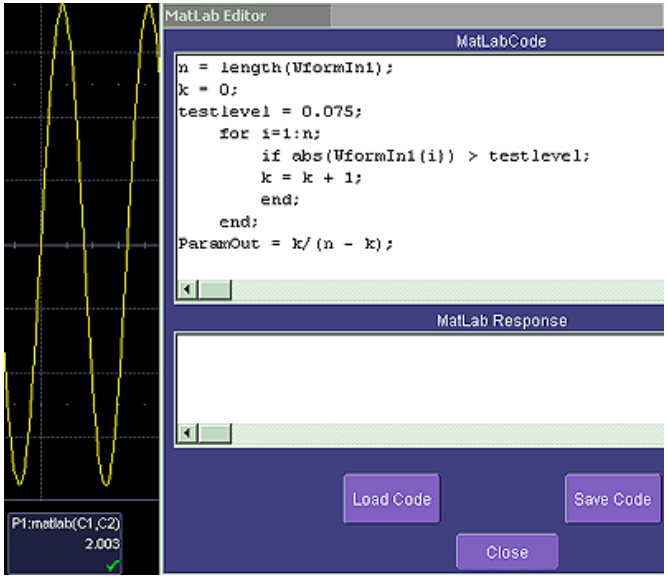
This simple example shows the MATLAB function Standard Deviation acting on input channel 1, and the result would be shown in the **MATLAB Response** pane for an amplitude of 0.15 volt. You can load an existing MATLAB program, using the **Load Code** button, and you can save the current program, using the **Save Code** button.

If you are familiar with MATLAB you might prefer to launch MATLAB and create a MATLAB function that performs your task. Your program in the instrument could then be a one-line call of your MATLAB function.

MATLAB Example Parameter Panel

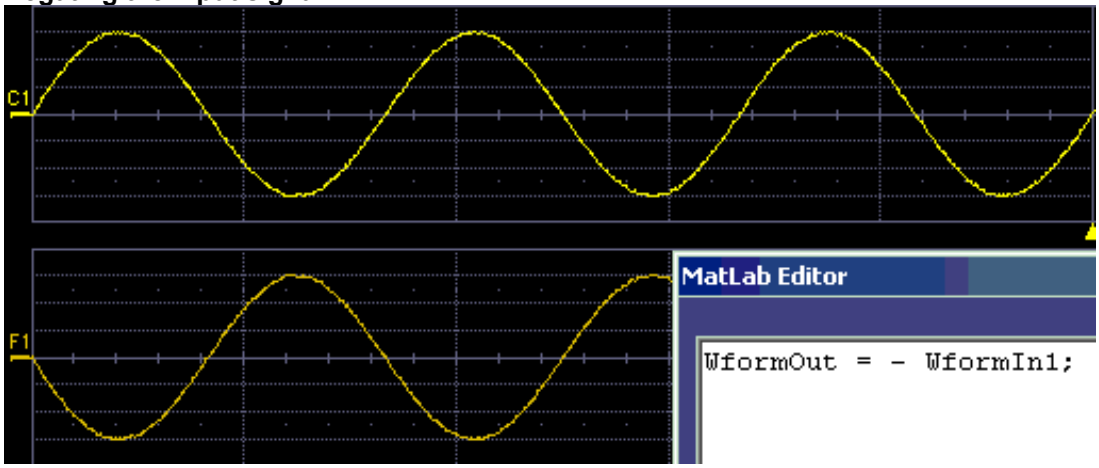


The next example calculates the ratio of the number of data points that are above a given level to the number of points below the level, in this case one half of the amplitude.

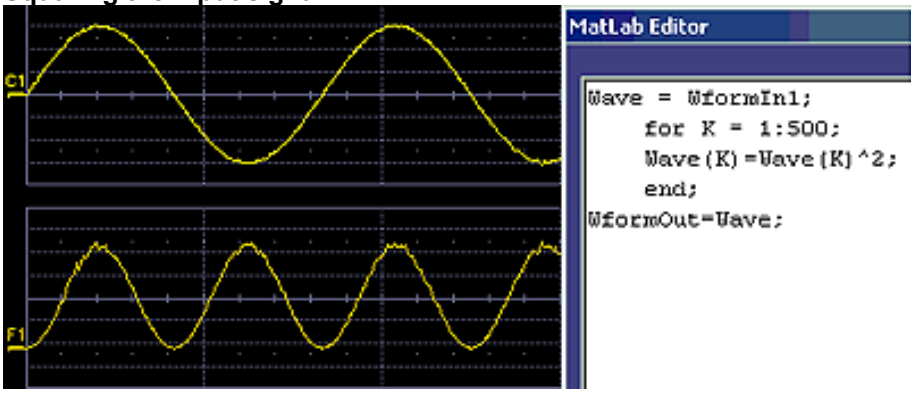


More Examples of MATLAB Waveform Functions

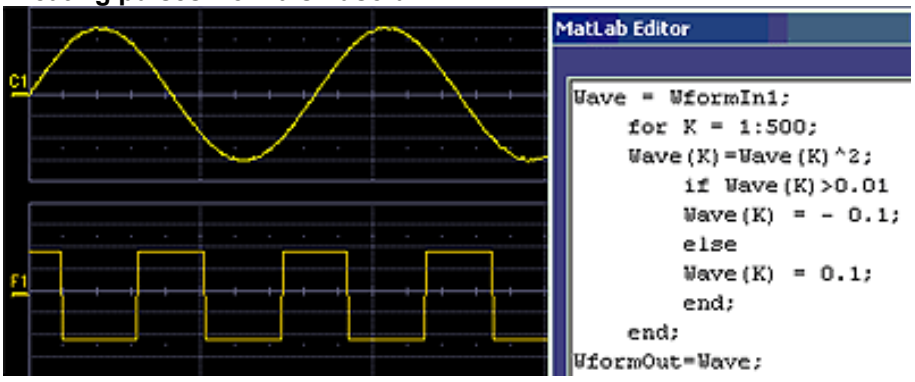
Negating the input signal.



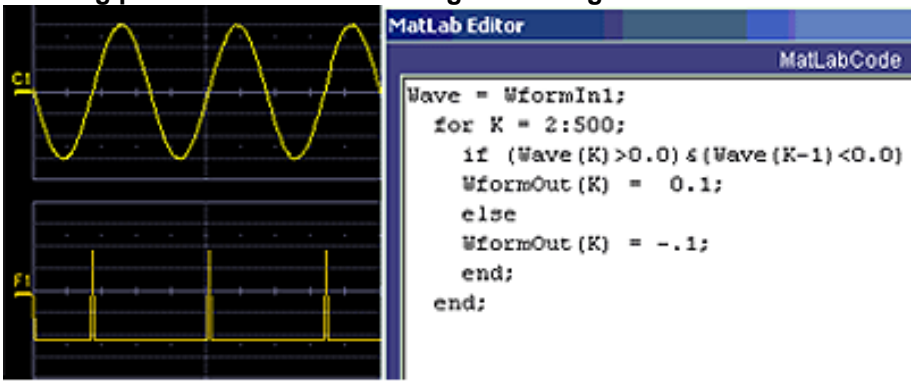
Squaring the input signal.



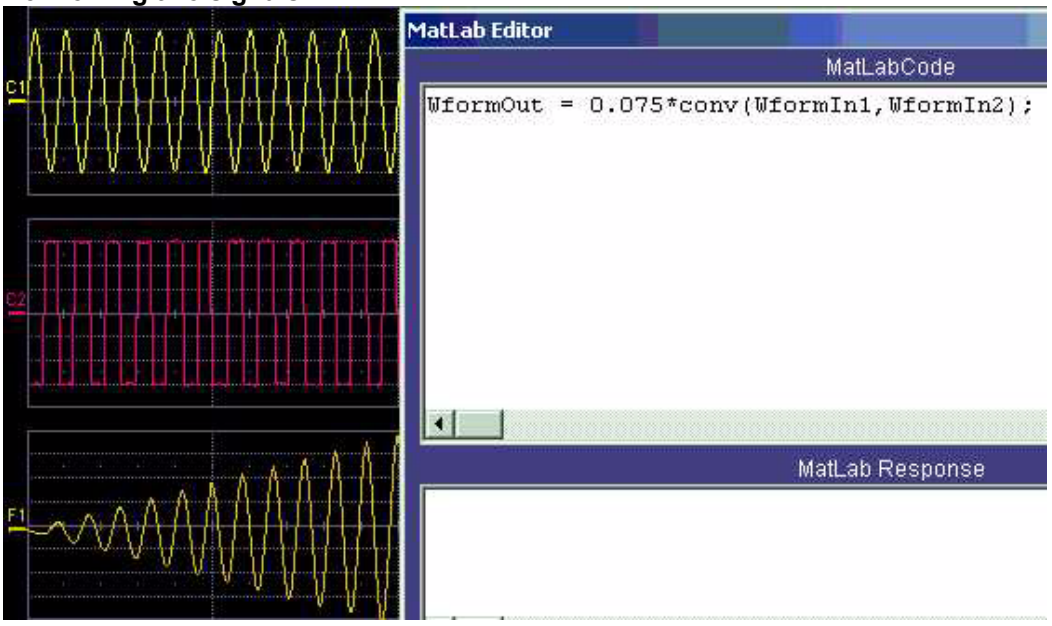
Creating pulses from a sinusoid.



Creating pulses at the zero crossings of the signal.



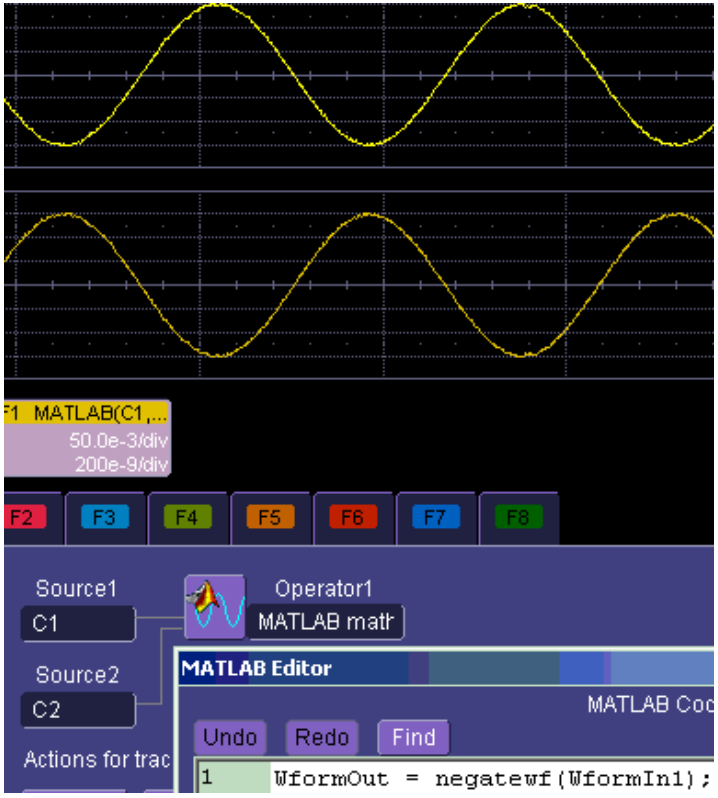
Convolving two signals.



Creating Your Own MATLAB Function

The procedure is simple. Create a MATLAB function using any text editor, and save it as a MATLAB m-file by giving it a name of the form Filename.m. Call the function using the MATLAB math editor or the MATLAB parameter editor as appropriate. A simple example is shown below.

```
function out = negatewf(wf1)
      % NEGATEWF changes the sign of all the data.
out = -wf1;
```



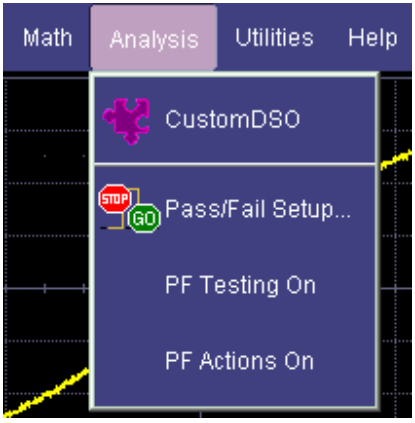
CUSTOMDSO

What is CustomDSO?

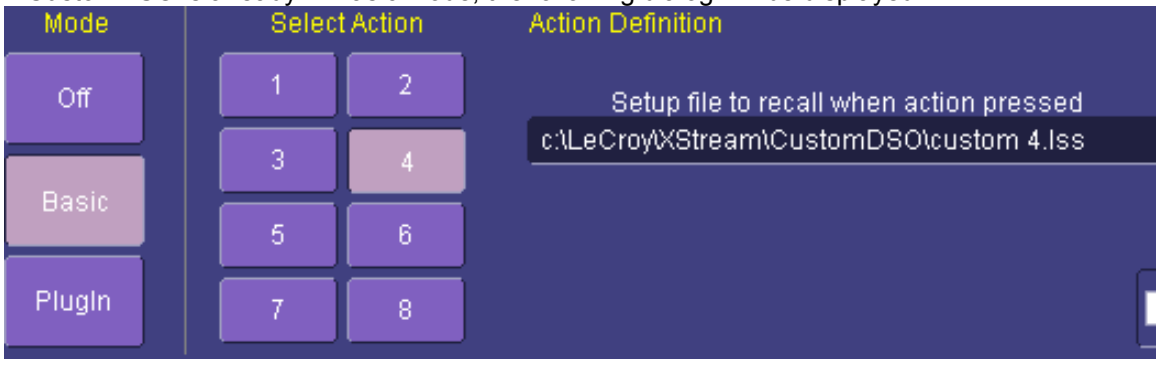
CustomDSO, in its Basic mode, allows you to create DSO setups that can be called by the touch of a single button. The recalled setups can themselves include calls to other setups. A very simple example would be a toggle between two setups. Rings of three or more setups are possible, as are trees, or any other topology that you need. Basic mode also allows you to recall VBScripts that can set up all or part of the scope and do many other things.
Another more powerful feature is the PlugIn, which allows you to add your own ActiveX controls to a setup. These controls are powered by routines written in Visual Basic. With ActiveX controls you can create your own user interfaces to suit your own preferences. A large number of interactive devices are available: button, checkbox, radio button, list box, picture box, and common dialogue box.

Invoking CustomDSO

CustomDSO can be invoked from the **Analysis** drop-down menu:

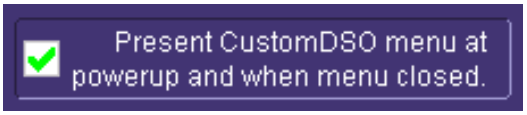


If CustomDSO is already in Basic mode, the following dialog will be displayed:



CustomDSO Basic Mode

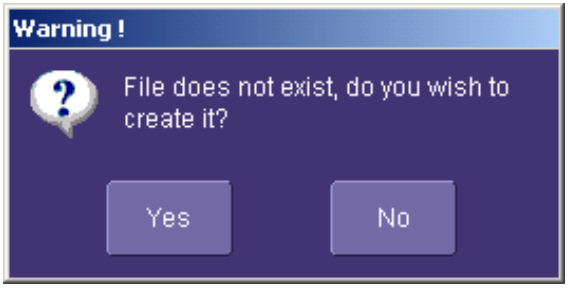
The **Basic** CustomDSO mode offers eight Action buttons, each of which can call a different setup when touched. The "Action Definition" dialog is used to enter a CustomDSO setup file name by means of the pop-up keyboard.
By clicking the checkbox, the eight CustomDSO buttons will continue to be available at the bottom of the screen after you close the CustomDSO dialog. Furthermore, they will appear automatically each time the DSO is powered up.



Editing a CustomDSO Setup File

If the file does not exist, touch the **Edit** button 

; a dialog will appear for you to create the file.



If the file does already exist, the **Edit** button enables you to modify it. The **Edit** button allows you to edit the file that is named in the **Setup file to recall** field, and not the file of the setup that the instrument is currently in, unless these happen to be the same.
In the example used here, three setup files were made, called CustomA.lss, CustomB.lss and CustomC.lss. Fragments from all three are shown below.

```
1160 Set CustomDSO = XStreamDSO.CustomDSO
1161 ' CustomDSO Setup A.lss
1162 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\A.lss"
1163 CustomDSO.ActionEnable1 = False
1164 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\B.lss"
1165 CustomDSO.ActionEnable1 = True
1166 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\C.lss"
1167 CustomDSO.ActionEnable1 = True
1168 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\A.lss"
1169 CustomDSO.ActionEnable1 = False
 1160 Set CustomDSO = XStreamDSO.CustomDSO
1161 ' CustomDSO Setup B.lss
1162 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\A.lss"
```

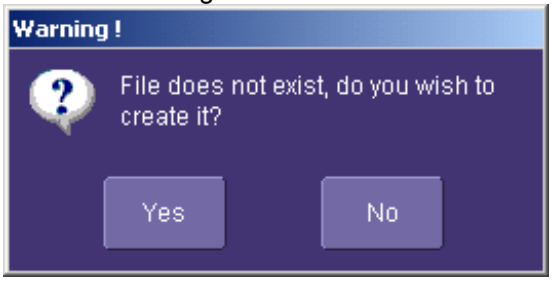
```
1163 CustomDSO.ActionEnable1 = True
1164 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\B.lss"
1165 CustomDSO.ActionEnable1 = False
1166 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\C.lss"
1167 CustomDSO.ActionEnable1 = True
1168 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\A.lss"
1169 CustomDSO.ActionEnable1 = False
 1160 Set CustomDSO = XStreamDSO.CustomDSO
1161 ' CustomDSO Setup C.lss
1162 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\A.lss"
1163 CustomDSO.ActionEnable1 = True
1164 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\B.lss"
1165 CustomDSO.ActionEnable1 = True
1166 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\C.lss"
1167 CustomDSO.ActionEnable1 = False
1168 CustomDSO.ActionScript1 = "c:\LeCroy\XStream\CustomDSO\A.lss"
1169 CustomDSO.ActionEnable1 = False
```

The text in green following a single quotation mark is a VBS comment and causes no action.
The text in red contains the path and name of the setup file associated with the numbered button. This setup will be called when the button is pressed.
The boolean (in blue) decides whether the action button will invoke the setup or remain inactive. For example, in setup B, **A.lss** and **C.lss** can be invoked, but not B, which is already in place.
As you see from the line numbers in the program fragments, the setup files are rather long because they include all the information needed to set the DSO to the required state. But if you want to make a very short file that changes only a few variables (for example, the action button settings) you can make a file that includes only the relevant instructions. This usage assumes that the remainder of the DSO is already in the required state. This is an example of the complete compatibility of the instrument's software. The same commands can be used in setups, in scripts, or in remote control commands in external programs, whether resident in the instrument or in an external computer.

Creating a CustomDSO Setup File

If you touch the **Edit** button 

when the **Setup file to recall** field contains the name of a non-existent file, you will see a message like this:



If you then touch **Yes**, the DSO will display a file like this:

```
' XStreamDSO ConfigurationVBScript ...
```

```
' Created by CustomDSO ...
On Error Resume Next
set dso = CreateObject("LeCroy.XStreamDSO.1")
' dso.Display.GridMode = "Dual"
' dso.Acquisition.C1.VerScale = 0.1
' dso.Acquisition.Horizontal.HorScale = 1e-6
' dso.Acquisition.TriggerMode = "Auto"
```

You can add to this fragment any commands you need.

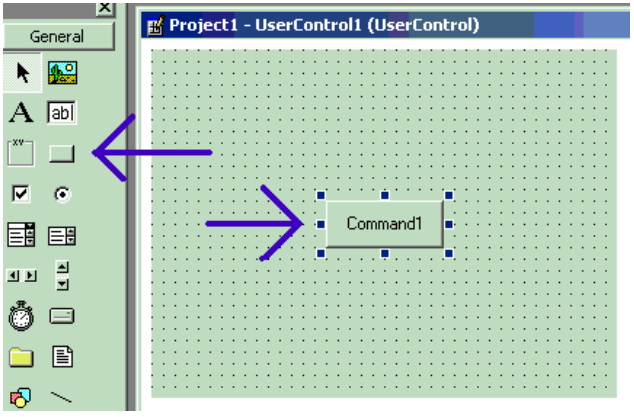
CustomDSO PlugIn Mode

This is the mode in which CustomDSO really shows its power. You can insert any ActiveX control or graph.

Creating a CustomDSO PlugIn

Follow these steps to create an example Visual Basic PlugIn:

1. Start a new VB project. Select **ActiveX Control** from the **New** tab.
2. Resize the control. A. In the Properties window set Width 11940. B. In the Properties window set Height 2475.
3. Place two buttons on the control. A. Double click on the command button at left of screen (left arrow below). B. Move and resize the resulting button as required, using the handles (right arrow below). C. Repeat for the second button. D. In the Properties window set the Name properties to **SingleButton** and **AutoButton**, respectively. E. Set the button Caption properties to **Single** and **Auto**, respectively.



4. Create code for the buttons. A. Double click on the **Single** button. B. In the resulting code window, insert code to make the following subroutine:

```
Private Sub SingleButton_Click()
Dim app as Object
Set app = CreateObject("LeCroy.XStreamApplication")
app.Acquistion.TriggerMode = "Stopped"
End Sub
```

Double click on the **Auto** button.

In the resulting code window, insert code to make the following subroutine:

```
Private Sub AutoButton_Click()
Dim app as Object
Set app = CreateObject("LeCroy.XStreamApplication")
app.Acquistion.TriggerMode = "Auto"
End Sub
```

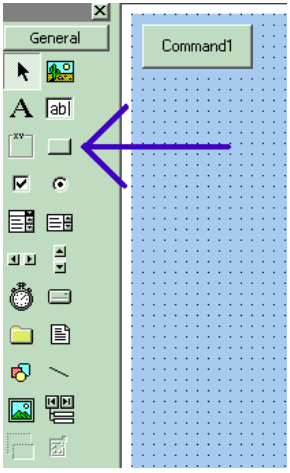
5. Test the Component in Internet Explorer. (This is an optional, but very useful step, because you can test your work without installing anything in the instrument.) A. Start the instrument. B. Click the **Run** button In Visual Basic. C. Click the **Stop** button in Visual Basic when you have finished.
6. Make the Project in Visual Basic. A. Click the **Stop** button in Visual Basic. B. Select Make Project1.ocx from the File menu.
7. Install the PlugIn in the instrument. A. Start the instrument. B. Select **ActiveDSO** in the Analysis Menu. C. Select **PlugIns** mode. D. Type "Project1.UserControl1" in the "COM ProgID of Plug-In" text box. E. Click the Install button under the text box.
8. Now Click the new **Auto** and **Single** buttons to see their effects.

Properties of the Control and its Objects

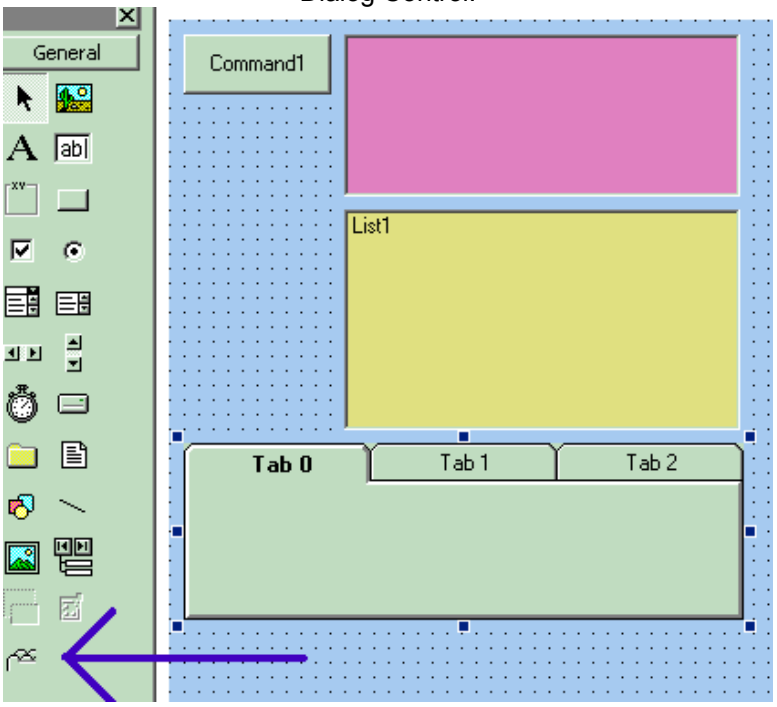
Using the **View Properties** button in Visual Basic, you can customize your PlugIn to your exact requirements. Among the most useful properties are the following: Height, Width, BackColor, Name, and Caption.

You can gain access to the properties of your objects by Clicking View – Properties. Positions and sizes of objects can be changed from View – Object, by dragging the object or one of its handles.

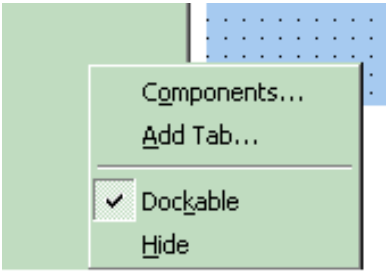
You can insert any available control into your plug-in. The basic control set is shown in a toolbar at the left of the screen in the picture below. Double click on any control to insert it into the plug-in. In the following example, a command button has just been inserted.



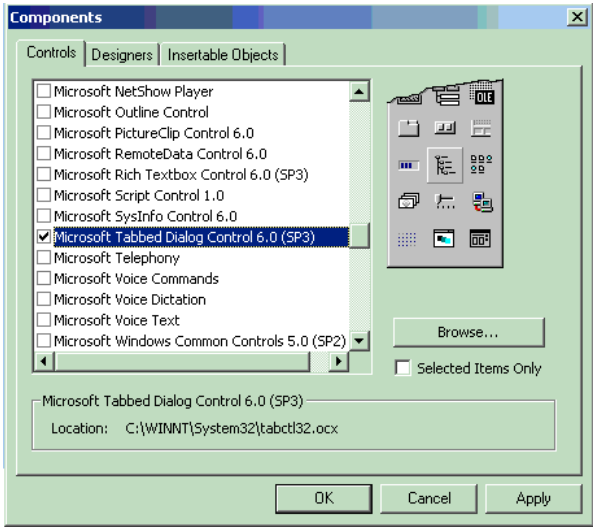
In the next example you can see a command button, a picture box, a list box and a Tabbed Dialog Control.



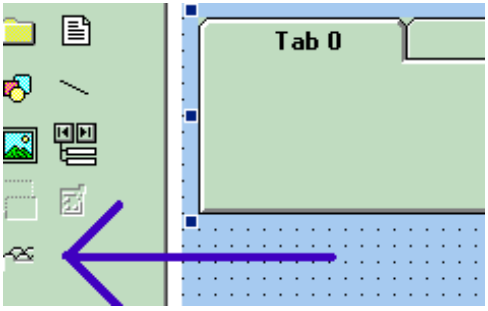
The Tabbed Control (arrow) is not in the basic tool box. To gain access to it, right click in the tool box at left (but not on an icon.) You will see this menu:



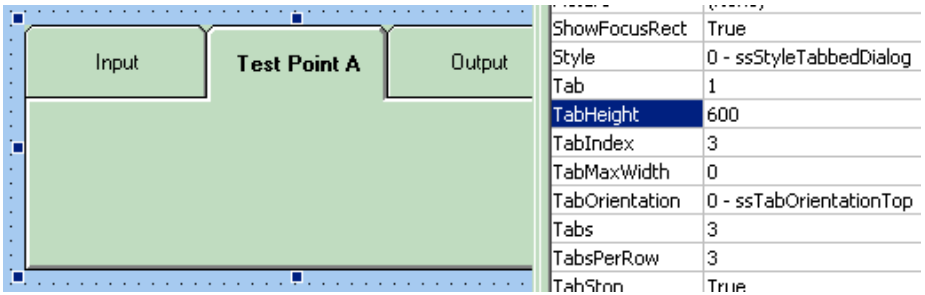
Now select the Microsoft Tabbed Control as shown below, and click on Apply. The control will be added into the toolbox at the left of the screen, where you can double click on it as usual.



The new control is shown below (arrow).

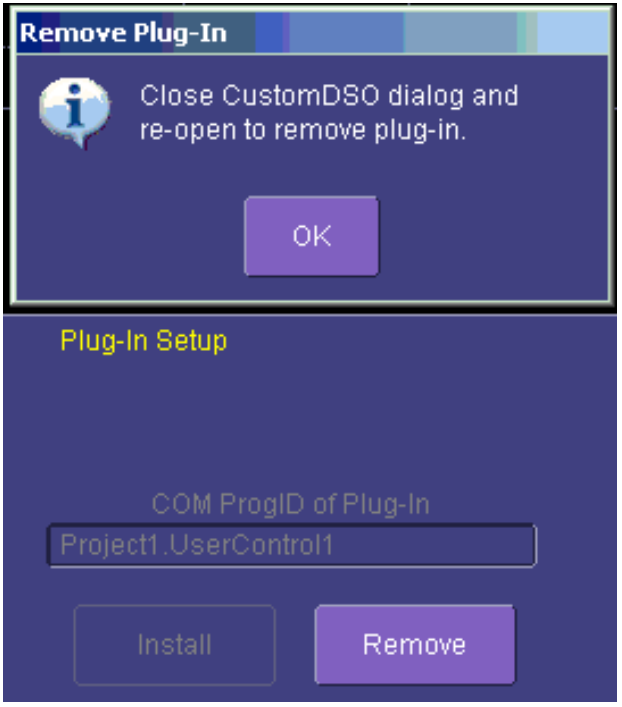


The system is very versatile, and you can place controls on the tabs of the Tabbed Control. Look in the properties window to see how you can customize your tabs, illustrated as follows.



Removing a PlugIn

To remove a plug-in, click on Remove in the PlugIn dialog as follows:



Close the CustomDSO dialog and then reopen it. At this point, the plug-in should be removed.

PlugIn Example 1: Exchanging Two Traces on the Grids

The example assumes that the instrument is in dual-grid mode, and that there are at least two visible traces. The routine looks for the visible traces, in the order C1 . . . C4, F1 . . . . Fx The number of math traces available depends on the software options loaded on your scope. See Specifications., and it exchanges the first two it finds whenever the button is pressed. Note that arrays of objects can be constructed, allowing numerous objects to be accessed in simple loops.

```
Private Sub Command1_Click()


Dim wm As Object
Set wm = CreateObject("LeCroy.XStreamApplication")


Set acq = wm.Acquisition ' To save typing
Set mat = wm.Math        ' To save typing
Dim t(16) As Object
' Create an array of objects to allow looping.
Set t(1) = acq.C1 : Set t(2) = acq.C2
Set t(3) = acq.C3 : Set t(4) = acq.C4
Set t(5) = mat.F1 : Set t(6) = mat.F2
Set t(7) = mat.F3 : Set t(8) = mat.F4
Set t(9) = mat.F5 : Set t(10) = mat.F6
Set t(11) = mat.F7 : Set t(12) = mat.F8

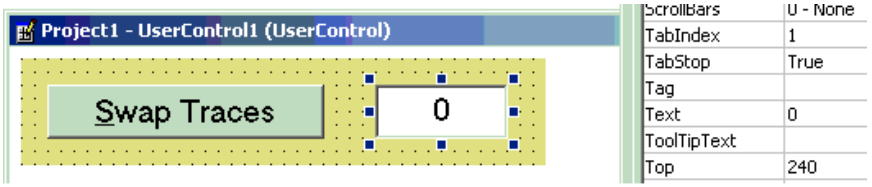

Dim trace As Integer
trace = 0: views = 0
' Exchange the traces on the grids.
    Do
    trace = trace + 1
        If t(trace).View = "True" Then
        views = views + 1
            If t(trace).UseGrid = "YT1" Then
            t(trace).UseGrid = "YT2"
            Else
            t(trace).UseGrid = "YT1"
            End If
        End If
    Loop Until ((trace = 12) Or (views = 2))
' Show the parity of the last swap.
    If Command1.Caption = "Swap A" Then
    Command1.Caption = "Swap B"
    Else
    Command1.Caption = "Swap A"
    End If


Dim TextString As String
TextString = Text1.Text
Dim TextValue As Integer
TextValue = Val(TextString) + 1
TextString = Str(TextValue)
TextString = Trim(TextString)
```

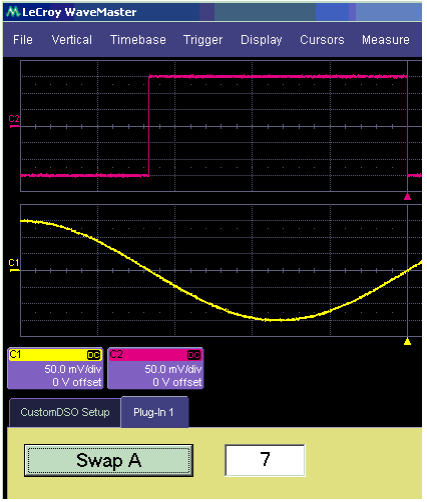
```
Text1.Text = TextString

End Sub
```

This routine exchanges the first two traces that it finds. You can make it exchange all the traces on a dual grid by changing the penultimate line to this - Loop Until trace = 12
The next figure shows the Visual Basic Screen just after the Text Box text has been set to “0” in the Properties Window, thus defining the initial value.

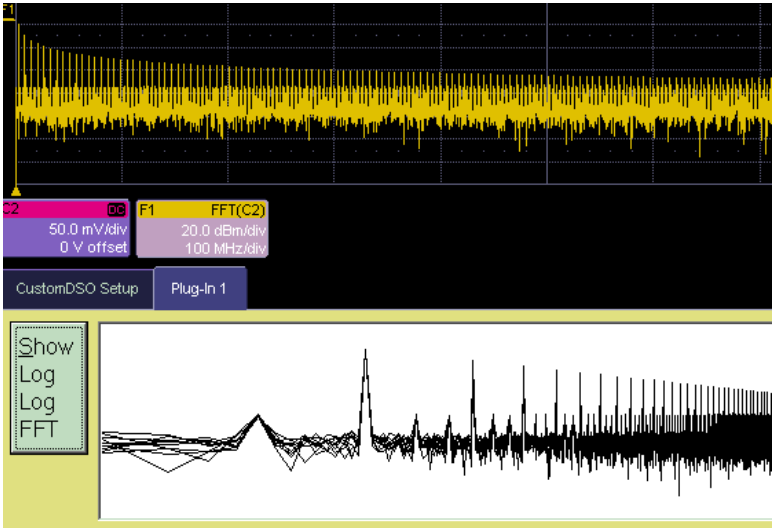


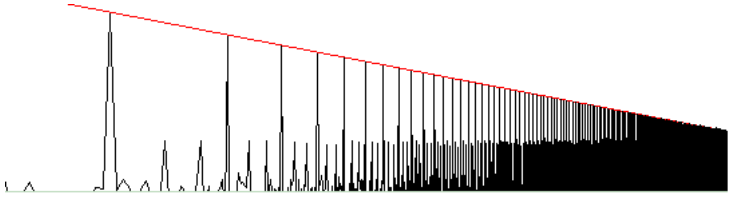
Here is the result after seven swaps. The counting method could be useful in any routine where numerous operations, such as triggers, have to be performed. In fact, the caption of the button could have been used to show the number of operations.
ActiveX offers a large range of standard controls, including list boxes for selection from a list, and picture boxes for drawing graphs and charts.



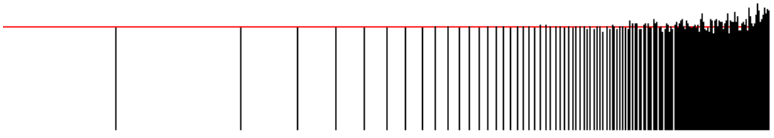
Second Example PlugIn: Log-Log FFT Plot

A frequent requirement is to plot a frequency spectrum on two logarithmic scales. The instrument provides a vertical scale, so CustomDSO has only to change the horizontal one. Here is an example. The first figure has been truncated on the right side.





These examples were made with two different instrument setups: in the second, the FFT was zoomed vertically. The graph has a red line to represent the theoretical envelope for the peaks. This has great potential for testing the transmission characteristics of amplifiers and filters, since the output can be compared with a theoretical curve. Furthermore, if the output is divided by the curve, the result for a perfect DUT would be a horizontal line, which is easy to inspect. The example below has been magnified vertically by a factor of ten. The rise at the right side occurs because the signal is descending into the noise.



```
Private Sub Command1_Click()
'   Draw a DSO trace on a logarithmic horizontal scale.
Dim WM As Object
Set WM = CreateObject("LeCroy.XStreamApplication")


Dim Samples As Long
Samples = WM.Math.F1.Out.Result.Samples
Samples = Samples - 1 ' Make it a round number.
'   Calculate the horizontal scale.
LogSamples = Log(Samples)
XScale = Samples / LogSamples


'   Set the scale using DSO variables
Dim Top, Bot As Single
Top = WM.Math.F1.Out.Result.VerticalFrameStop
Bot = WM.Math.F1.Out.Result.VerticalFrameStart
Picture1.Scale (0, Top)-(Samples, Bot)


Dim Wave
Wave = WM.Math.F1.Out.Result.DataArray

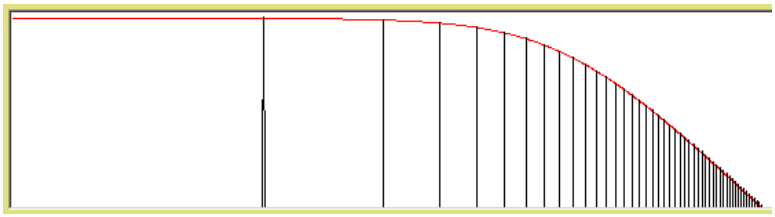

Dim Black, White, Blue, Red As Long
Black = 0: White = &HFFFFFF
Blue = &HFF4444: Red = &HFF


'   Draw a theoretical curve for the peaks.
StartPoint = Top + 20#: EndPoint = -54.5
Picture1.Line (0, StartPoint)-(Samples, EndPoint), Red


'  Draw the plot with linear interpolation between points.
    For X = 1 To Samples
    LogX = XScale * Log(X): Y = Wave(X)
        If X > 1 Then
        Picture1.Line (LogX, Y)-(OldLogX, OldWave), Black
        End If
    OldLogX = LogX: OldWave = Y
    Next X


End Sub
```

Here is an example showing a simple one-pole roll-off compared to a curve.



Control Variables in CustomDSO

The simplest way to select variables for use in CustomDSO is to use LeCroy's X-Stream Browser.

LABNOTEBOOK

Introduction to LabNotebook

LeCroy's LabNotebook feature extends the documentation capabilities of your scope. It allows you to create an annotated notebook entry containing all displayed waveforms, the setup of the DSO, and user-supplied annotation. The notebook entry can then be converted to hardcopy format - pdf, rtf, or html - and printed or e-mailed. You can also use the default report layout or configure your own, and even substitute your own company logo in the header.
Notebook entries are stored in an internal database and are available for recall at any time. Besides storing the waveform data, LabNotebook also stores your panel setups and parameter measurements. You have the capability to back up the database to external media.
The Flashback feature allows you to recall the state of the DSO at a later date, including the saved waveforms and the DSO setup, so that you can make additional measurements. A keyword filter makes it easy to find the correct notebook entry to recall.
You can choose which notebook to use for your entries, and label the notebook by project or user. If the scope is shared among several users, for example, or used for different projects, the data can be kept separately. Similarly, hardcopy reports can be stored in different folders.

Preferences

You should set your preferences before creating notebook entries.

Miscellaneous Settings



You can elect to name notebook entries with the default date and time by leaving the top box unchecked. Check the box if you want the opportunity to rename the notebook entry as soon as it is created.
Check the middle box if you want to be able to annotate a notebook entry as soon as it is created.

Check the last box if you want to generate a notebook entry by simply touching the Hardcopy (Print) front panel button . By checking this box, you override any other configuration for this button; for example, send e-mail or output to printer.

Hardcopy Setup



Check the **Use Print Colors** checkbox to place your waveforms on a white background in the notebook entry. This will save printer ink later when you print the hardcopy report.
Touch inside **Hardcopy Area** to determine how much of the screen image to include in the report: grid area only, grid area plus dialog, whole screen.

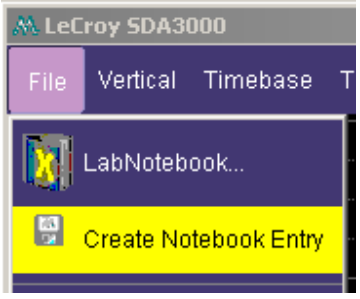
E-mail Setup



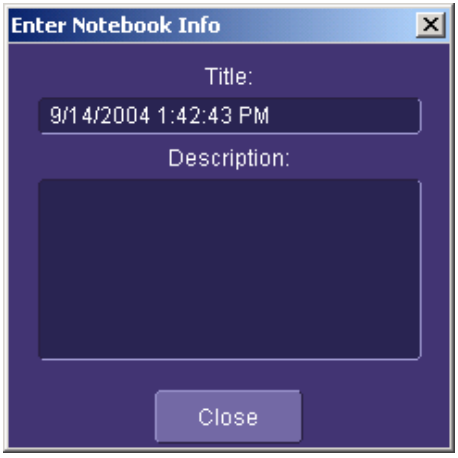
You can e-mail just the pdf or html report; or, you can include additional files: trace data (.trc) for each waveform in the report, a screen dump, a scope setup file, and an xml report record. Touch the checkbox to enable the extra report segments.
Touch the **Configure E-Mail** button to set the recipient address and server information.

Creating a Notebook Entry

1. Touch **File** in the menu bar, then **Create Notebook Entry** in the drop-down menu:



A dialog box is shown for providing a title and comments for the entry. By default, the entry is titled with the current date and time.



2. Touch inside the **Title** field and enter a title, using the pop-up keyboard. Then touch inside the **Description** field and enter a description, if desired, and touch **Close**.
3. The notebook entry will display your waveforms in "print colors," that is, on a white background to save printer ink, if you selected that option in notebook Preferences. Otherwise, the waveforms will appear on a black background. A drawing toolbar appears at top:


The pen tool enables you to write or draw in freehand. You can use a mouse, or a stylus to do this using the touch-screen. Once you click off, you can drag your note anywhere on your waveform.


The circle tool enables you to create a circle around a waveform feature that you want to point out. Once you click off, the circle is drawn and you can drag it anywhere on the screen.


The arrow tool enables you to draw lines with arrowheads for callouts. You can rotate these lines through 360 degrees and drag them to any location on the screen.


The text tool enables you to enter text callouts on your report. When you touch this tool, a dialog box opens in which to enter text by means of a pop-up keyboard.



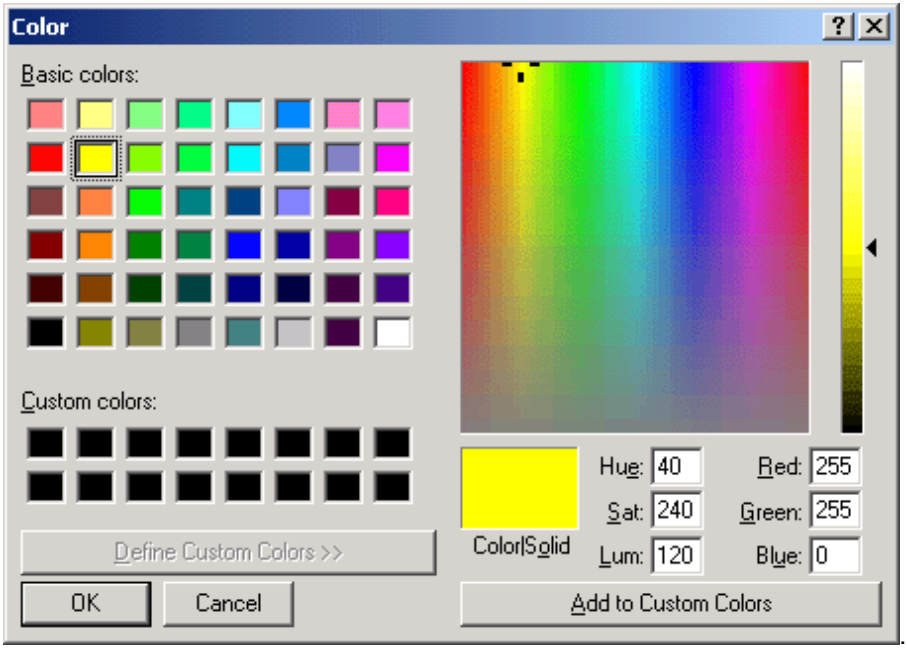
After you touch **Close**, your text will appear on the display as a draggable object.



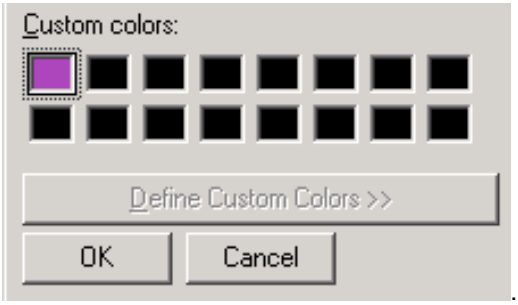
These are the three default colors that you can select for shapes, lines, and text. To use additional colors, touch **More**.



When you touch **More**, a Custom box opens with the default color yellow displayed. Touch the yellow button to open the full color palette



When you have chosen a custom color, touch **Add to Custom Colors**; the color will appear in the **Custom Colors** palette



Then touch the color to enable it, and touch **OK**. The next object that you create will be in that color.



If you want to erase a drawing object, touch it to select it, then touch **Erase Selected**.



Touch **Erase All** to erase all drawn objects and text.



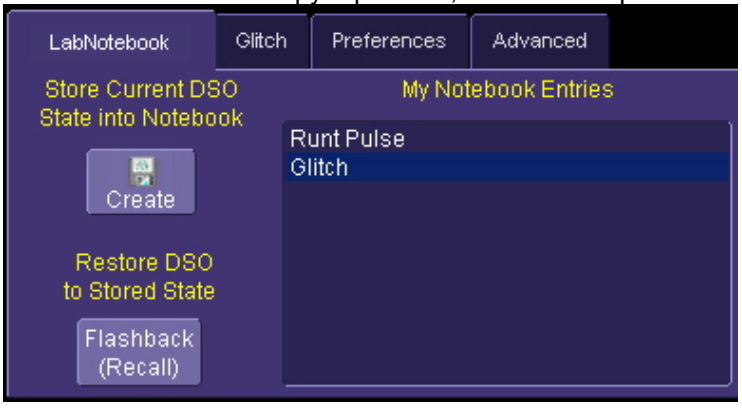
Touch **Undo** to discard the last object drawn.



The **Move Toolbar** button enables you to place the toolbar anywhere on the screen. Touch the button a second time to return it to its original fixed location.

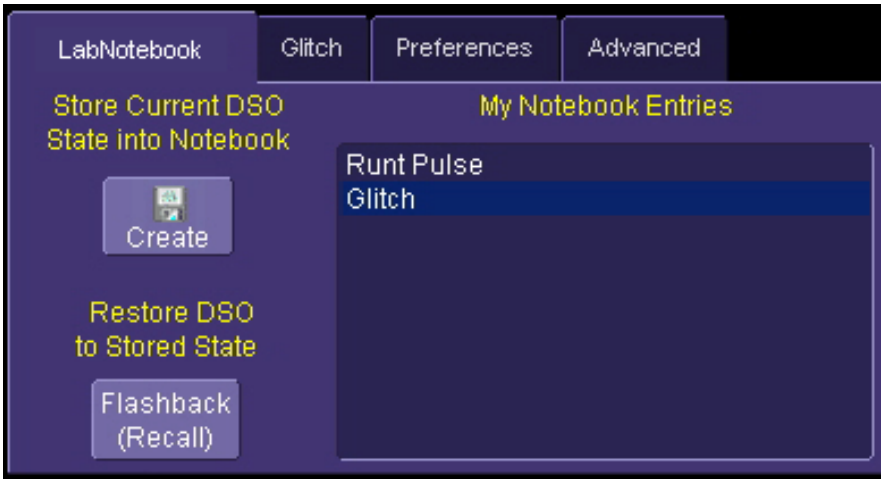


Touch **Done** when you are finished annotating the notebook entry. The name of the entry will appear in the list box in the "LabNotebook" dialog. You can now create a hardcopy report of it, and email or print it out.



Recalling Notebook Entries

After a notebook entry is made, you can recall it at any time. The recall includes waveforms and scope settings.



1. Select the notebook entry from the list box.
2. Touch **Flashback**.
3. To exit Flashback, touch the **Undo Flashback** button 

in the top-right corner of the screen, or press the **Auto** trigger button.

Note: The flashback feature currently recalls the DSO Setup, and all displayed waveforms. Some forms of 'result data' are not recalled, including:

- Persistence data. This will be saved in the hardcopy, and will be printed in the report, but will not be recalled during Flashback.
- Histogram data. Histograms internally have a 32-bit resolution, but when stored into a trace file and recalled during flashback they are clipped to 16-bits.
- Floating point waveforms. Certain math operations result in the creation of floating point waveforms with much higher resolution than can be stored in a 16-bit waveform file. This extra resolution will not be preserved when traces are recalled using flashback.
- Cumulative Measurements. Any measurements that are on when the Lab Notebook entry is created are not saved individually in the database, other than being embedded in the hardcopy image. This means that when flashback is used, the measurements will be recomputed using the waveform data that was recalled. Normally this will not pose a problem, but if cumulative measurements were on, which accumulated data from multiple acquired waveforms, they will loose their history and show instead only the results from the stored waveforms.

Creating a Report

Once the notebook entry is created, you can easily generate a hardcopy report for e-mailing or printing.

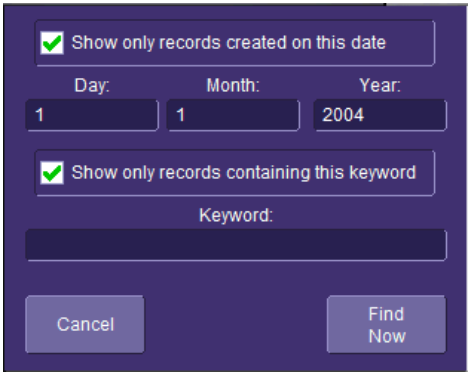
Previewing a Report

Before creating a report, you can preview it by simply touching the **View** button 

. To exit the preview, touch the **Close** button at the right of the dialog.

Locating a Notebook Entry

A search filter is provided to help you locate the notebook entry you want to make a report of. You can search by date or keyword.



Touch the **Filter** button 

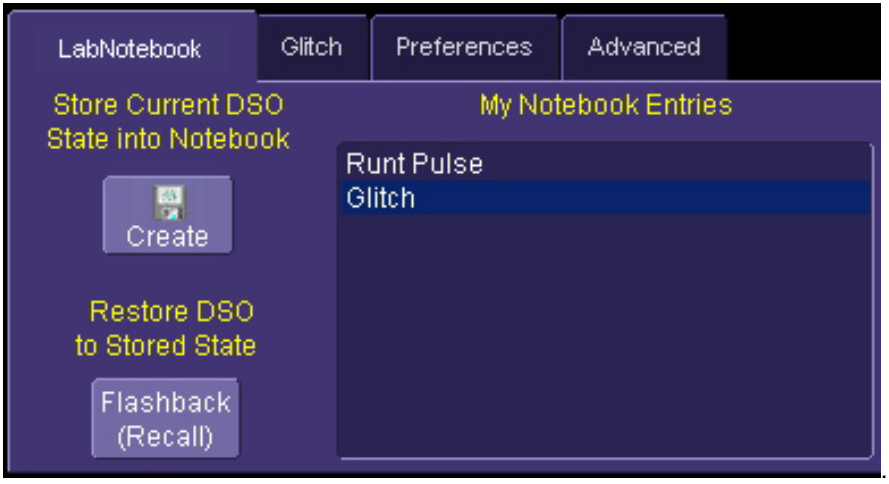
. A search dialog box opens.

Touch inside the **Day**, **Month**, and **Year** fields and enter a date. Or touch inside the **Keyword** field and enter a keyword or phrase.

Touch **Find Now**. Only the entries fitting the date or keyword criteria will now appear in the list box.

Creating the Report

1. Select a notebook entry in the list box



2. Touch inside the **Format** field and select a report format from the pop-up menu



3. Touch the **Create Report** button.
4. A dialog box opens in which to name the report and select a folder to contain the report. Touch inside the **File name** field and enter a name using the pop-up keyboard.
5. If you want to e-mail or print the data to a network printer, touch **More Actions**, then the **Print** or **E-Mail** button. If you select **Print**, a Windows dialog box will open for you to select a printer and set options. If you select **E-Mail**, the report will be sent immediately to the e-mail address configured in Utilities Preferences.

Formatting the Report

LeCroy provides a default report format (template); however, you can use your own format, including company logo.



1. Touch the **Advanced** tab.
2. Touch inside the **Directory** field and navigate to a folder to contain the reports.
3. Touch the **Browse** button next to Template to navigate to an existing report format that you want to use. Or touch inside the **Template** field and enter the name and path to the template, using the pop-up keyboard. Otherwise, touch the **Use Default** checkbox to use LeCroy's format.
4. To use a logo other that the one provided, which indicates the scope that produced the report, browse to the bit map file or touch inside the **Logo** field and enter the name and path to the file, using the pop-up keyboard. Otherwise, touch the **Use Default** checkbox to use LeCroy's logo



Note: If you elect to use your own logo bit map, do not use a bit map larger than 180 pixels (height) x 100 pixels (width).

Managing Notebook Entry Data

Adding Annotations

You can add annotations to your notebook entry at any time.

1. Touch the "LabNotebook" tab.
2. Touch the notebook entry you want to annotate in the scroll list box. A new tab will appear bearing the name of the selected notebook entry.
3. Touch the new tab, then the **Scribble** button 

. The notebook entry will appear again with the drawing toolbar, described in Creating a Notebook Entry.

Deleting Notebook Entries

1. Touch the "LabNotebook" tab.
2. Touch the **Delete All** button 

to clear the database, or select a notebook entry in the list box, then touch the **Delete** button 

to discard just that one entry.

Saving Notebook Entries to a Folder

You can save notebook entries to a folder other than the default.

1. Touch the tab bearing the name of the notebook entry.
2. Touch the **Save Data to** button 

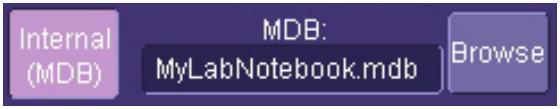
. A navigation window opens, which provides the opportunity also to open Windows Explorer to navigate to the folder.
3. Touch the **Zip** checkbox 

if you want to compress the data before archiving.

Managing the Database

You can begin a new database for your notebook entries at any time, back up the current one, or compress the data.

Selecting a Database for Backup or Compression



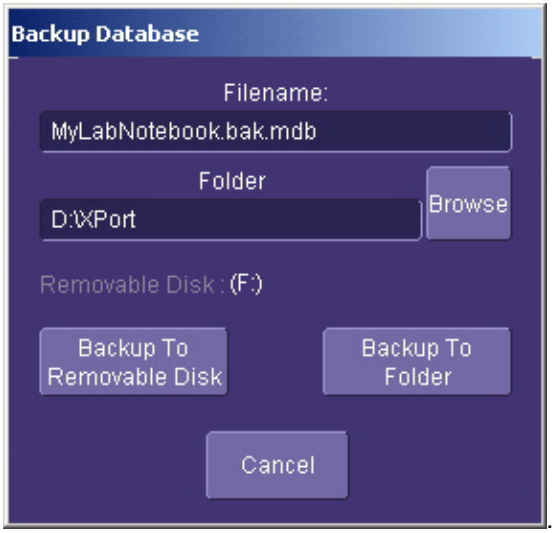
1. Touch the **Advanced** tab.
2. Touch the **Browse** button. A navigation window opens. Navigate to the database you want to work on



Touch **Compact** to reduce the size of a database. This function "defragments" the notebook after a large amount of entries have been deleted.

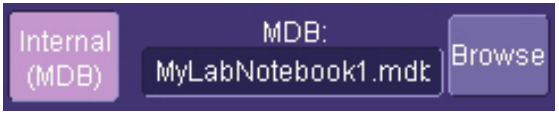


Insert a memory stick into a USB port, then touch **Backup** to send the database to the external media:



Starting a New Database

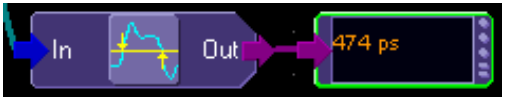
Touch the **Start New** button. The name of the notebook database will be incremented by 1:



PROCESSING WEB OPTION

This feature is available with the XWEB software option.
The Processing Web provides a graphical way to quickly and easily set up math functions and parameter measurements. Using the Processing Web, you can chain together many more math-on-math functions than you can using the Math Setup dialog, where you are limited to two functions. In addition, you can insert a parameter measurement for any math output waveform anywhere in the web.
The "web" analogy derives from the nodes and connecting lines used to construct the web. Nodes are math functions selected from the **Add Math Processor** menu, parameters selected from the **Add Measure Processor** menu, or parameter math functions from the **Add Parameter Math Processor** menu.

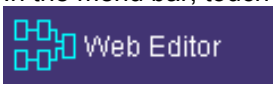
Another key feature of the Processing Web is that you can preview your waveform at any math or parameter node in the web. Math previews are thumbnail images of the waveform. For parameters, the statistic displayed is the value of the last acquisition.



Once you have created a Processing Web setup, you can save and recall it for future use, the same as for any panel setup.

Using the Web Editor

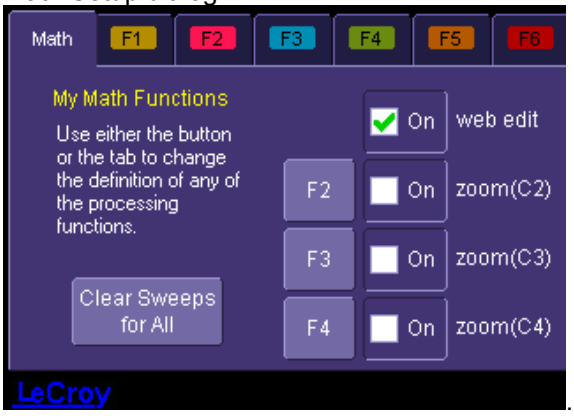
1. In the menu bar, touch **Display**, then **Web Editor** in the drop-down menu.



2. Touch the **Math** tab and select a math location (F1 to Fx [The number of math traces available depends on the software options loaded on your scope. See Specifications.]) for the new math function that you are about to create by touching the **Web Edit** button:



. Once you select a math location for web editing, it cannot be used for another math function, and will appear as unavailable in the Math Setup dialog:



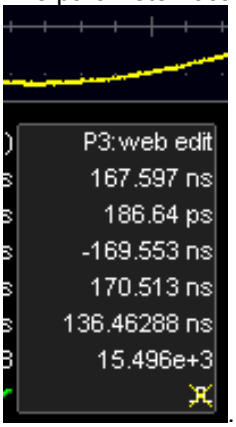
.

However, you can cancel web processing within the "Math Setup" dialog by touching the single function, double function, or graph button. Touch the **Measure** tab, then touch the **Web Edit** button, if you want to dedicate a parameter location (P1 to Px) for web processing:

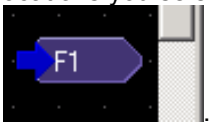


.

The parameter location you choose will display "Web Edit" under the waveform display grid:



3. Touch the **Web Editor** tab to return to the web setup dialog. The math and parameter locations you selected appear as outputs at the far right:



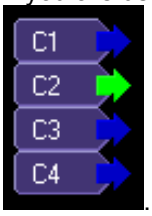
You may have to scroll up or down to see it.

4. Touch the **Add Math** button and select a math function from the **Add Math Processor** menu.

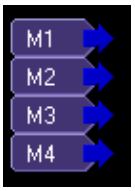
The math function icon will appear on the web setup field: 

. Touch and drag the icon to the desired location.

5. If you are using channel inputs, touch the arrow of a channel input icon:



Then drag a line from the channel to the input of the math function icon. If your math function is a dual input function (such as ratio), select a second input and drag another line to the second math input. If you are using a memory location:



(M1 to M4) as an input, drag a line to the math function in the same way as for channel inputs.

Note: You can use a combination of channel input and memory input to your math function.

6. Touch the output arrow of the math function icon and drag a line to the Fx output on the right-hand side of the setup field. Your math function is complete.

Adding Parameters

Add parameter measurements in the same way as for math functions. Parameters can be connected to any math function in the web.

Adding Previews

1. Touch the **Add Preview** button: 

. A scope-like icon will appear:
.

2. Touch the output arrow on the math function or parameter icon and drag a line to the input arrow of the preview icon. A thumbnail view of your signal will appear if the preview icon is connected to a channel output or math function output. If it is connected to a parameter output, a numeric value of the last acquisition will be displayed:


.

Exiting the Web Editor

To exit, touch the **Close** tab; or, in the menu bar, touch **Display** then **Scope Display** in the drop-down menu:



The scope display will return to the normal waveform display grid.

Viewing the Output

1. Touch **Math** in the menu bar, then **Math Setup...** in the drop-down menu.
2. Touch the **On** checkbox for the function you want to view:



SERIAL DATA ANALYZER

Serial Data Analyzer Standard and Optional Capabilities

The Serial Data Analyzer is an instrument designed to provide comprehensive measurement capabilities for evaluating serial digital signals. In addition to the WaveShape Analysis features in the standard WaveMaster scope, the SDA provides eye pattern testing and comprehensive jitter analysis, including random and deterministic jitter separation, and direct measurement of periodic jitter, DDj, and DCD. The SDA also provides the capability to directly measure failed bits and to indicate their locations in the bit stream.

Please Note the Following:

- **SDA** - name of the instrument: Serial Data Analyzer
- **ASDA-J** - Advanced Serial Data Analysis package, available only on the SDA
- **SDM** - Serial Data Mask testing package, available on WaveMaster, WaveRunner 6000A Series, and WavePro 7000A Series oscilloscopes. Not available on the SDA.

SDA Capabilities

In addition to all the standard WaveMaster scope measurement functions, the SDA provides two other types of measurements: jitter and eye pattern. The ASDA-J option also adds bit error rate analysis to the SDA. These measurements are available together in the summary screen, as well as in individual modes.

Measurements on the SDA are performed on long, continuous acquisitions of the signals under test. All jitter measurements and all displays are based on times of successive edges of the signal only; nothing is relative to the trigger. As a result, they are not affected by trigger jitter.

Acquisitions should be long enough to include at least several thousand UI of the signal under test: 30,000 UI or more is optimal. To see low frequency jitter, it may be desirable to acquire longer records. Acquisitions can be up to the full available memory depth of the instrument (up to 100M samples with option XXL), which may take considerable time to process.

Serial Data Analysis, which includes mask testing and jitter parameters (Rj, Dj, Tj, DDJ, Pj, DCD), is standard in the SDA. It is also available with option ASDA-J, which adds a major upgrade in capability over the standard SDA instrument. The different measurements available with each configuration are shown in **Table 1**.

SDM Capabilities

The capabilities of option SDM are standard in the SDA, so it is not available for purchase for the SDA. This option is only available for the WaveMaster, WaveRunner 6000A Series, and WavePro 7000A Series of oscilloscopes. SDM adds eye pattern testing to these oscilloscopes.
The option also adds other key components to the basic scope, including JTA2 with its TIE@lvl parameter. TIE@lvl is a JTA2 measurement that measures the time interval error of the crossing points of the signal under test and, with option SDM, also includes a golden PLL clock recovery module that is used for forming the eye pattern without an external trigger. Standard masks are included with option SDM as indicated in Table 2. Note that not all data rates can be tested with all oscilloscopes. The analog bandwidth limits the upper data rate that can be tested.

ASDA-J Capabilities

ASDA-J adds several key capabilities to the SDA. In its standard form, the SDA includes eye pattern testing with mask hit indication; Jitter testing, including jitter bathtub computation and separation of jitter into its random and deterministic components; as well as the breakdown of deterministic jitter into periodic, data dependent, and duty cycle distortion.
Option ASDA-J adds the following analysis features:

- Mask violation location This is the ability to list and view the individual bits that violate the selected mask. (Mask violation location takes much more time than jitter testing; acquisition size should be just a few thousand UI.)
- Filtered jitter Processes the time interval error trend vs. time with a user-selectable band-pass filter. This feature provides peak-to-peak and rms measurements of the jitter on the filtered waveform.
- ISI plot Generates an eye diagram including only those affects from data dependent sources. The user can select from 3 to 10 bit patterns for this test and can view the contribution from any individual pattern. The ISI plot is an alternate method for measuring data dependent jitter when the signal under test does not contain a repeating bit pattern.
- Bit error test with error map Measures the number of bit errors and error rate on the acquired waveform by converting the wave shape to a bit stream and comparing the result to a user-definable reference pattern. The data can be further divided into frames that can be arranged in a 3-dimensional map with frame number on the Y-axis, bit number on the X-axis, and failed bits shown in a light color.

AVAILABLE MEASUREMENTS			
Single-Signal Measurements	**SDA (std.)**	**SDA w/ ASDA-J**	**SDM option**
Data Stream			
Mask testing w/ software PLL clock recovery	X	X	X
Mask violation locator		X	
Jitter Rj, Dj, Tj, ISI, DCD (DDj), Pj	X	X	
Filtered jitter		X	
ISI plot		X	
Edge-to-edge jitter		X	
Effective and MJSQ jitter breakdown		X	
Bit error testing with error map		X	
N-cycle jitter parameter (data)	X	X	
Eye Pattern Measurements			
Average power	X	X	X
Bit rate	X	X	X
Extinction ratio	X	X	X
Eye amplitude	X	X	X
Eye bit error rate	X	X	X
Eye crossing	X	X	X
Eye height	X	X	X
Eye width	X	X	X
One level	X	X	X
Zero level	X	X	X
Q factor	X	X	X

Table 1. Measurements Available in SDA, ASDA-J, and SDM.

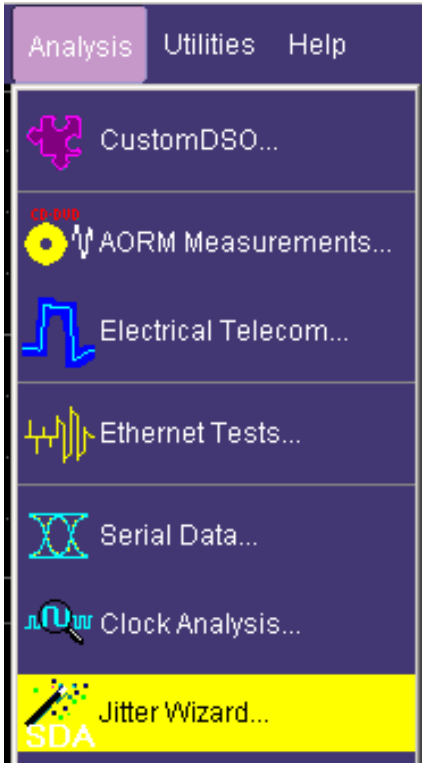
STANDARD	MODE
Gigabit Ethernet 1000Base-CX	TX normalized/absolute, RX
Gigabit Ethernet (optical long-haul) 1000Base-LX	TX
Gigabit Ethernet (optical short-haul) 1000Base-SX	TX
10GBase-LX4	TX normalized
DVI	Transmitter, receiver low, receiver high, cable test low, cable test high
Fibrechannel (optical) FC2125, FC1063	TX normalized
Fibrechannel (electrical) FC531, FC266, FC133	TX normalized, TX absolute, receiver
IEEE1394b	400 beta TP2 absolute, 400 beta TP2 normalized, 400 beta receive
Infiniband 2.5 Gb/s	Transmitter
RapidIO serial	3.125Gb/s TX (long haul/short haul), Rx 2.5Gb/s TX (long haul/short haul), Rx 1.25Gb/s TX (long haul/short haul), Rx
SONET	OC-1, OC-3, OC-12, OC-48, STS-1 eye, STS-3 transmit, STS-3 interface
SDH	STM-1, STM-4, STM-16
PCI Express	TX add-in transition, TX add-in de-emphasized, TX system transition, TX system de-emphasized, RX
Serial ATA 1.5 Gb/s	TX connector, RX connector
USB2.0	TX (near end/far end), RX (near end/far end), TX driver output, RX receiver input
XAUI	Driver far, Driver near

Table 2. Standard Masks Included with Option SDM

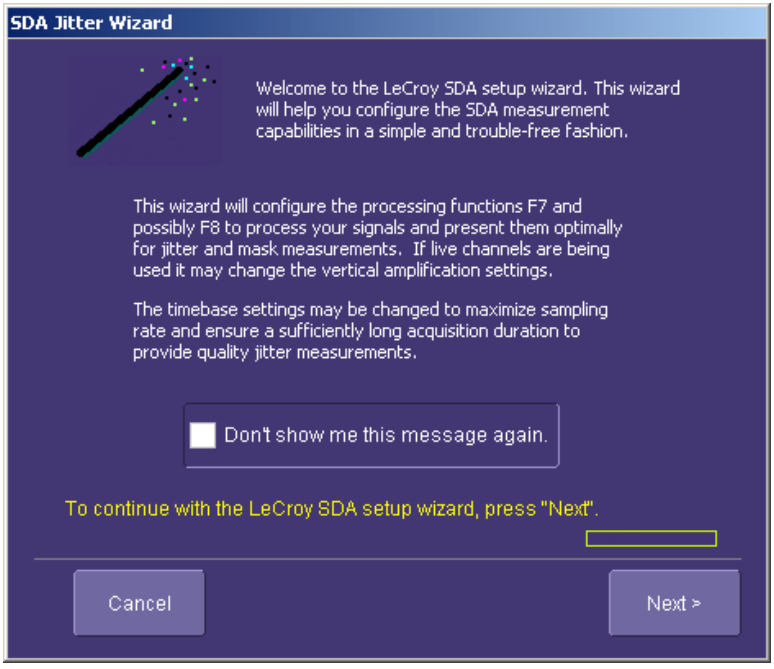
Jitter Wizard

The Jitter Wizard is a unique feature available with the ASDA-J option that automatically configures the SDA for optimal jitter measurement. As you provide information regarding the signal under test, the instrument selects the most appropriate settings. The wizard adjusts the sampling rate, memory depth, and vertical scale; and adds additional interpolation to the signal where appropriate.

The jitter wizard is accessed from the Analysis drop-down menu:



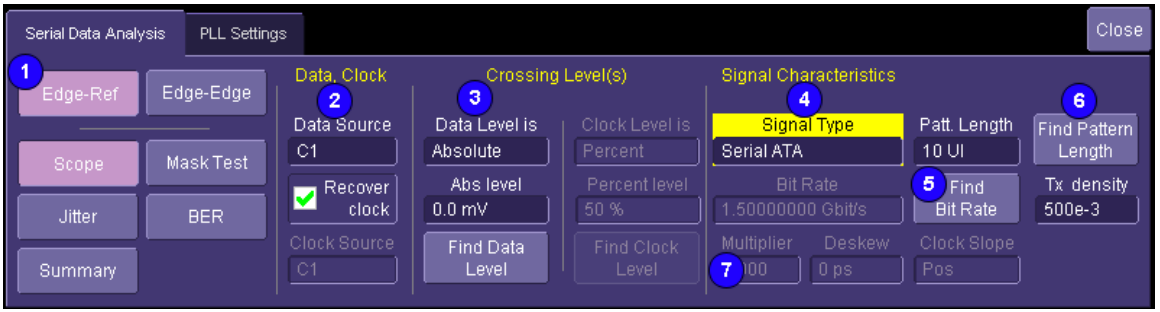
The initial dialog for the wizard gives you an overview of wizard operation. This dialog can be disabled after the first use of the wizard by checking the "Don't show me this message again" checkbox. In this case, the dialog will not be displayed until the instrument is restarted.



The remainder of the jitter setup wizard guides you through a series of dialogs that request input concerning the signal type and measurement method that is desired. Explanations of the various choices that can be made are also provided.
Each dialog has a default selection that represents the most common configuration. You can accept the defaults by simply touching the Next button in each dialog.

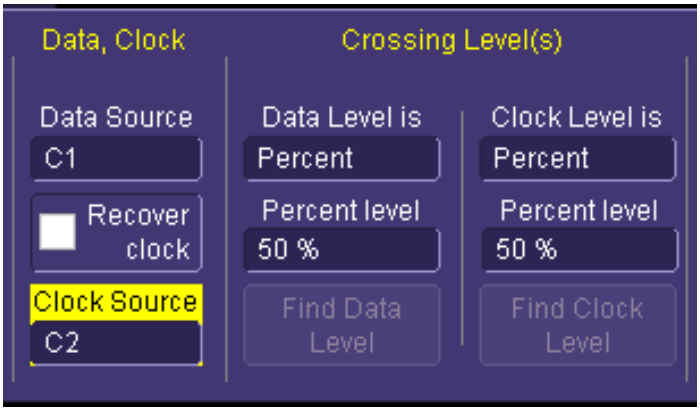
SDA Basic Setup

Note: Many of the measurements in the SDA require both a high sampling rate and long memory to compute accurately. Verify that you are in two channel mode (20 GS/s) and that at least 400k samples are being captured before performing any SDA measurements. Lower sampling rates can result in less accurate jitter measurements, and short record lengths can give incomplete eye patterns or jitter displays that diverge.



1. To access the SDA dialogs, press the Serial Data front panel button; or, touch **Analysis** in the menu bar, then **Serial Data** in the drop-down menu. The display defaults to the Scope dialog, which enables you to set up SDA measurements while you continue to view currently displayed waveforms. Touching the **Scope** button turns off SDA features. In the image above are shown two measurement modes: **Edge-Ref** and **Edge-Edge**. The Edge-Edge mode is optional and only available with the ASDA-J option. If this option is not present, the two mode buttons are not displayed. The Edge-Edge mode is explained separately in this manual; the information presented here applies mainly to the standard (Edge-Reference) mode.
2. Touch inside the **Data Source** field and select a data source, then touch inside the **Clock Source** field and select a clock input. If you want to recover the clock from the signal, touch the **Recover clock** checkbox; the **Clock Source** field will become inactive. The "Crossing level(s)" section in this dialog allows you to set the voltage level at which the signal timing is measured. The crossing level is set separately for the data and clock (if an external clock is selected) and can be either absolute or relative. The **Absolute** crossing level in volts (or watts for an optical signal) can be set directly, or can be found automatically by touching the **Find Data Level** button. The level is found by locating the midpoint between the highest and lowest signal levels in the current acquisition. The **Relative** level is automatically set to the selected percentage on each acquisition.

3. There are two crossing-level controls in the SDA main dialog: one for the data signal under test and another for the reference clock (if the **Recover clock** checkbox is unchecked). The crossing time used by both the jitter and eye pattern measurements is determined as the absolute time at which the signal level crosses the selected threshold. Touch inside the **Data Level Is** or **Clock Level Is** field and select either **Absolute** or **Percent** from the pop-up menu.



4. If you selected **Absolute**, touch inside the **Abs level** field and enter a voltage value. Alternatively, you can touch the **Find Data Level** or **Find Clock Level** button. If you selected Percent, touch inside the **Percent level** field and enter a percentage value, using the pop-up keypad. The level is automatically set to the selected percentage on each acquisition.

Note: The absolute level setting should be used for jitter measurements over several acquisitions because the relative setting can potentially remove jitter by tracking slowly, thereby varying level shifts that could occur between acquisitions.

5. Touch inside the **Signal Type** field and select a standard signal type from the pop-up menu. The signal type defines the compliance masks and bit rates for the selected standard. If you selected **Custom** from the **Signal Type** menu, touch inside the **Bit Rate** field and enter a bit rate, using the pop-up keypad. By touching the **Find Bit Rate** button, you can have the bit rate determined automatically. This button is also available when a standard data type is selected.

6. The **Pattern Length** (number of bits in the pattern) control sets the repetition length of the data pattern in the signal under test. Serial data signals generally transmit a repeating data pattern of some sort. For example a PRBS7 pattern repeats every 127 bits while a K28.5 repeats every 20 bits. Enter the pattern length in the **Pattern Length** control. Alternatively, the SDA can automatically find the pattern length by means of the **Find Pattern Length** button. The **Find Pattern Length** button determines the pattern repetition length (if any) in the current acquisition and sets the **Pattern length** field to this value. The Pj measurement uses the pattern length value to determine which peaks in the jitter spectrum are caused by the pattern repetition. An incorrect value in the Pattern Length control will result in overestimating the Pj and, as a result, underestimating the random jitter.
7. The **Multiplier**, **Deskew**, and **Clock Slope** controls adjust the positioning of the clock relative to the data edges when an external clock source is selected. These controls are active only if the **Recover clock** checkbox is left unchecked. The **Multiplier** control is used when a sub-rate clock is applied. For example, if a clock signal at 1/10 the bit rate is used, the Multiplier control should be set to 10 in order to get a single eye pattern on the display.
   The **Deskew** control enables positioning of the clock edges relative to the data signal. It shifts the clock signal relative to the data signal.
   A clock signal goes through one complete cycle during each bit interval. The edge timing can be measured relative to the rising or falling slope of the clock by means of the **Clock Slope** setting.
   The **Tx Density** control indicates the ratio of transitions in the data record to the total number of bits in the signal. Normally, an NRZ data stream does not have transitions during every bit interval. For example, there are cases where adjacent 1 or 0 values exist. On average, approximately 50% of the bit intervals have transitions so the Tx density is 500e-3. The actual acquired data set may have a transition density different from 50%, however. The total jitter measurement uses this transition density to normalize the measured histogram into a pdf (probability density function). Many instruments simply assume 50% when performing this normalization which can lead to errors. These errors are eliminated in the SDA through the use of the measured transition density in the normalization process. The Tx density is automatically measured when the **Find Pattern Length** button is pressed.

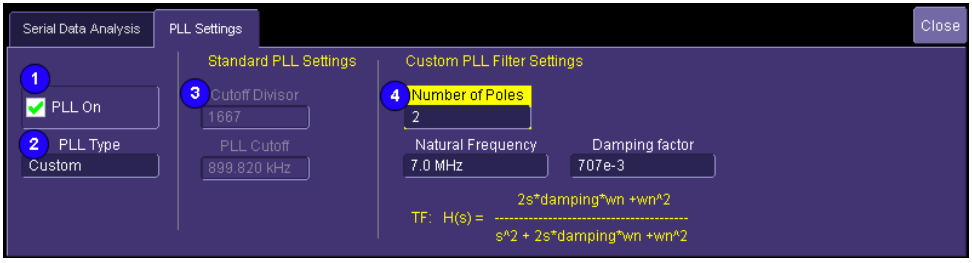
Please Note the Following:

- If the pattern length exceeds the number of bits in the acquired waveform or there is no repeating pattern present, the control indicates a value equal to the total number of bits in the current acquisition. In this case, the pattern length control should be set to 2 so the Pj is not underestimated.
- Overestimated Pj values result in incorrect pattern length. Such overestimations cause Dj to read high and Rj to read low.

PLL Setup

The "PLL Settings" dialog contains the controls to set the type and bandwidth of the digital PLL used in the jitter, eye pattern, and bit error rate tests. The PLL bandwidth limits the response of the recovered clock to high rate variations in the data rate. For example, a PLL bandwidth of 750 kHz will allow the recovered clock to track frequency variations below this rate, thereby removing their effect from jitter and eye pattern measurements. The software PLL implemented in the SDA allows you to choose among four types of PLL.

The selected PLL is applied to either the data stream under test or the selected clock source when the PLL On control is checked. The PLL recovers a reference clock from the selected source, which is used by all subsequent SDA measurements (jitter, eye pattern, and, with option ASDA-J, bit error rate).



1. Place a check in the **PLL On** checkbox to enable it.
2. Touch inside the **PLL Type** field to select the type of PLL to be used in the clock recovery function. The four choices are **FC Golder**, **PCI Express**, **DVI**, and **Custom**.
   - **FC Golden** is the default selection and implements the "golden" PLL as defined in the Fibrechannel specification. By default, the golden PLL is set to a cutoff frequency of 1/1667 times the bit rate of the signal under test. This ratio can be adjusted from 1/10 to 1/1e6.
   - The **PCI Express** PLL uses a filter that approximates the PCI-SIG compliance requirement. The PCI-SIG compliance procedure describes a processing algorithm that measures the average bit rate over 3500 consecutive unit intervals (UI or bit intervals). The 250 UI in the center of this 3500 UI window are then processed using the average bit rate as a reference clock. The 3500 UI window is then shifted by some number of UI and the process is repeated. Measurement continues until the end of the acquired data record is reached. The PCI Express PLL selection in the SDA models the sliding 3500 UI clock recovery and 250 UI processing windows using a digital low-pass filter whose cutoff frequency is approximately 1.5 MHz.
   - The **DVI** selection follows the requirements of the DVI (Digital Video Interactive) and HDMI (High Definition Multimedia Interface) specifications. These specifications call out a clock recovery function that has a single-pole PLL loop response with a cutoff of 4 MHz.
   - The **Custom** selection allows you to select either a first or second order loop response. The first order response allows you to select a pole frequency that sets the PLL cutoff, and a zero frequency that must be higher than the pole frequency that limits the stop-band attenuation.

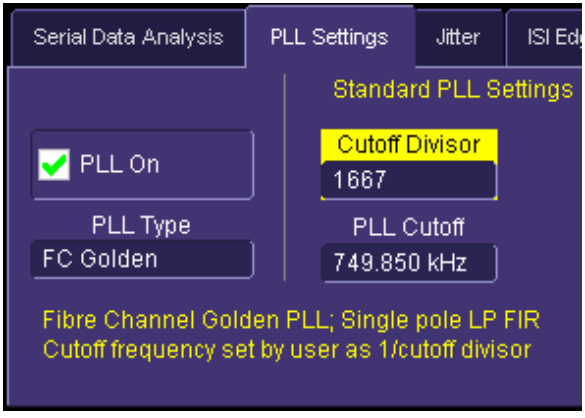
The second order PLL allows the selection of the natural frequency and damping factor. The damping factor determines the transient behavior of the phase locked loop and is variable from 2 to 0.5. A damping factor above 0.707 results in an under-damped response in which the PLL over-corrects to a sudden change in frequency, but reacts quicker to the change. A damping factor below 0.707 gives an under-damped response that reacts slower to sudden changes in frequency, but does not over-correct. The default value of 0.707 represents a critically damped response that provides the fastest reaction time without over-correcting. The second order PLL with a damping factor of 0.707 is specified in the serial ATA generation II document. This type of PLL is also very useful for measuring signals with spread-spectrum clocking because it accurately tracks and removes the low-frequency clock spreading while allowing the signal jitter to be measured.

The natural frequency is somewhat lower than the actual 3 dB cutoff frequency given by the following equation:

$$B_{3dB} = \frac{\omega_n}{2\pi}\left(2\zeta^2 + 1 + \sqrt{(2\zeta^2+1)^2 + 1}\right)^{1/2}$$

The quantity ζ is the damping factor, and ω_n is the natural frequency. For a damping factor of 0.707, this relationship is $f_c = 2.06\ f_n$.

3. **Settings for the FC Golden PLL -** Touch inside the **Cutoff Divisor**[5] field and provide a value using the pop-up keypad.



[5] The PLL cutoff divisor is the value by which the bit rate is divided to compute the cutoff frequency for the loop bandwidth of the clock recovery operation for sequential eye pattern, jitter, and bit error rate functions. This control is variable from 10 to 1,000,000. A low PLL cutoff divisor means that the PLL will track and, therefore, attenuate jitter at higher frequencies. The default value of 1667 causes the clock recovery to operate as a "golden PLL," as defined in the Fibrechannel specification.

The default value of 1667 is the industry standard for a "Golden PLL" and equals the ratio of the Bit Rate to the PLL Cutoff frequency. The **PLL Cutoff** frequency control reads the frequency corresponding to the Cutoff Divisor. Alternatively, the **PLL Cutoff** frequency may be entered and the nearest cutoff divisor will be computed from this entry.

4. Settings for the **Custom** PLL: Touch inside the **Number of Poles** control to select the order of the PLL. The number of poles can be 1 or 2. Touch inside the **Natural Frequency** control to set the natural frequency of the loop filter. The **Damping factor** control allows a damping factor setting between 2 and 0.5.

Summary

This is a quad-grid view of your signal: eye diagram, bathtub [The bathtub curve can be thought of as the "tails" of the TIE histogram, or the integral of the histogram. It is a logarithmic representation of bit error rate (BER). The greater the degree of separation between the sides of the bathtub curve, the better the BER. Deterministic jitter dominates the shape of the bathtub curve near the edges of the graph where the curves are at their highest amplitude, while random jitter determines the shape of the curve near the middle.], TIE trend [This is a trend of the time interval error measurements between the data transitions and the reference clock. This is the time record of the information contained in the TIE histogram.], and TIE histogram [A histogram of the time interval error for all bits in the waveform. Note that while the eye diagram and mask are displayed, mask hits are not detected in this view.].

Note: The Summary screen does not allow mask testing.

To access the "Summary" dialog, do the following:

1. Press the SDA front panel button; or, touch **Analysis** in the menu bar, then **Serial Data** in the drop-down menu. The display will default to the "Scope" dialog, which enables you to set up SDA measurements while you continue to view currently displayed waveforms.
2. Touch the **Summary** button.

Mask Test

Press the SDA front panel button; or, touch **Analysis** in the menu bar, then **Serial Data** in the drop-down menu. The display will default to the "Scope" dialog, which enables you to set up SDA measurements while you continue to view currently displayed waveforms. See the Basic Setup and PLL Setup sections for details on setting up the clock recovery.

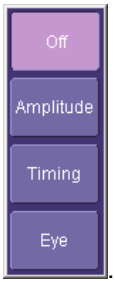
Eye Setup

The PLL must be set up or an appropriate external clock must be supplied in order for the mask test function to operate.

1. Touch the **Mask Test** button. The "Eye Diagram" dialog opens.
2. Touch inside the **Mode** field and select either **Sequential**[6] or **Traditional**[7] from the pop-up menu:



3. Touch inside the **Persistence** field and select persistence mode: **Monochrome** or **Color graded**. The Monochrome selection will display frequency of occurrence in levels of intensity of the same color, while the Color graded selection will use a color spectrum to indicate frequency of occurrence.
4. If you would like to display another time-synchronized signal in its own grid, touch inside the **User Signal** field and select another signal source from the pop-up menu. This will enable you to see the signal correlated to the failed bits in the eye mask when using the eye mask failure locator.
5. The **Mask Type**[8] field allows you to select different modes for the selected standard. Individual modes for a given standard define specific masks; for example, transmit or receive.
6. Under "Measure" touch inside the **Type** field, then select a set of parameter measurements from the pop-up menu:



[6] The Sequential eye pattern is created using a long acquisition of waveform data that is then processed to generate a recovered bit clock, and divided up into bit-sized segments that are then overlaid in a persistence map. The clock signal control is not used in this mode.

[7] The Traditional eye pattern is created by capturing, in a persistence display, many short records on the order of one bit interval (equal to the inverse of the bit rate). The SDA is triggered by an external trigger signal, which is designated in the **Clock** field.

[8] Each standard has a set of required tests. Some of the standards specify several types of masks. So a single standard can have a normalized mask and/or absolute mask for the transmitter. The standard requires you to make sure that the signal passes both the normalized and the absolute masks.

The measurements made for each parameter set are as follows:

Amplitude	ehght(Eye)	Size of the vertical opening of the eye diagram. This parameter is defined as: (mean ones level - 3) - (mean zero level + 3)
	one(Eye)	Logic one level of an eye diagram. This is the mean value of the one (high) state.
	zero(Eye)	Logic zero level of an eye diagram. This is the mean value of the zero (low) state.
	eampl(Eye)	Difference between the mean one and zero levels of an eye diagram
Timing	rise	Transition time from 20% to 80% for a rising edge
	fall	Transition time from 80% to 20% of a falling edge
Eye	ewdth(Eye)	Size of the horizontal opening of an eye diagram, minus 3 sigma on each side, where sigma is the standard deviation of the jitter.
	ER(Eye)	Ratio of the optical power levels in an eye diagram (extinction ratio), defined as log(p1/p0), where p1 is the power level for a "1" and p0 is the level of a "0" level.
	eyeBER(Eye)	Bit Error Rate estimated from an eye diagram
	ecross(Eye)	Level of the zero crossings in an eye diagram as a percentage of the eye amplitude
	avgpwr(Eye)	Average power level of a waveform. This is computed as the average of the sum of the ones and zeroes levels.

7. If you selected **Amplitude** or **Eye**, touch inside the **Slice Width** [The slice width is a percentage of the duration of a single bit, i.e., the part of the pattern over which the extinction ratio is measured. By setting a percentage value, you indicate how much of the central portion of the bit width to use. Slice width is marked by the dashed vertical lines on the display.] field and enter a value from 0 to 100%.

Mask Margin

1. Touch inside the **X** field and enter a value from 0 to 100%. As you enlarge the mask's X margin, you lengthen the horizontal dimension, bringing the mask closer to your waveform. Consequently, you will have more failures.
2. Touch inside the **Y** field and enter a value from 0 to 100%. As you enlarge the mask's Y margin, you widen the vertical dimension, bringing the mask closer to your waveform. Consequently, you will have more failures.
3. Touch the **Vertical Auto fit** checkbox if you want to scale the eye pattern.

Note: The vertical autofit function sets the scaling of the eye pattern so that the one level is at the second vertical division above center, and the zero level is at the second division below center. The **Vertical Auto fit** checkbox is automatically checked or unchecked depending on the Signal Type that you selected. For example, when an absolute mask like XAUI is selected, the **Vertical Auto fit box** is unchecked; but, it is checked for a normalized mask like FC1063.

Scaling for absolute mask signals is accomplished by setting the vertical scale of the input signal.

Testing

1. Touch the Mask Violation Locator tab to display the mask testing dialog
2. Touch the **On** checkbox to begin testing the signal bits against the selected mask.
3. ASDA-J option only: Failed bits are displayed in a table and as a waveform in the lower grid. The scale of the lower grid in which the failed bit waveform is set by the **Bits in Locator** field. The failed bit is centered around the middle two divisions and the vertical scale is the same as the eye pattern.
4. Touch the **Stop On Error** checkbox if you want to halt testing when an error occurs. But testing will continue until the end of the acquired waveform is reached.
5. ASDA-J option only: Touch the **Show Location** checkbox, then touch inside the **Show** field and select **All** or **Near X-Y** from the pop-up menu. **All** means that all failures will be shown, regardless of where they occur in the eye pattern. **Near X-Y** means that if you select a failure marker, either by touching the marker on the waveform or by touching the bit number in the Failures list, only the subsequent failures near the selected marker will be shown.

Note: The X and Y coordinates indicate the position of the selected failure on a grid normalized to one, read from left to right and top to bottom.

6. ASDA-J option only: Touch inside the **N Failures** field and, using the pop-up keypad, enter the number of failures after which data accumulation will end, up to 10,000. Multiple failures can occur on each bit, so the value of N failures will always be equal to or greater than the number of failed bits.
7. ASDA-J option only: Touch inside the **Bits in Locater** field and, using the pop-up keypad, enter the number of bits to be displayed in the bad bits trace. This trace shows the part of the waveform that violates the mask.

Bit Error Rate - ASDA-J Option Only

1. Press the SDA front panel button; or, touch **Analysis** in the menu bar, then **Serial Data** in the drop-down menu. The display defaults to the "Scope" dialog, which enables you to set up SDA measurements while you continue to view currently displayed waveforms.
2. Touch the **BER** button. This brings you to the BER setup dialog.
3. Under "Pattern," touch inside the **Method** field and select **Pattern** (expected bit pattern in the data stream), **PRBS** (a pseudo-random bit sequence), or **File** (a saved user ASCII hex file).
4. If you chose **Pattern**, use the pop-up keyboard to enter a hexadecimal number as a bit pattern. If you chose **PRBS**, touch inside the **PRBS Sequence** field and select a bit sequence from the pop-up menu. The digit at the end of PRBSx represents the number of shift registers to be used in the generation of the pseudo-random bit sequence used for BER testing. The sequences generated for each n value are as described in ITU-O.150. If you chose **File**, enter the path to the file, using the pop-up keyboard; or, touch the **Browse** button and navigate to the file.

5. Touch the **Show Error Map** checkbox to enable a display of the error map. The error map displays bit errors in a three-dimensional display, with errors shown in white and correct bits in dark gray. The display is a 2-color surface map with frame number in the Y direction and bit number in the X direction. A frame is indicated by the frame sync bit sequence and is composed of the bits from the end of one frame sync to the start of the next one. If frame sync bits are not specified, the bits are laid out in fixed-length rows, starting at the top left corner of the screen and proceeding from left to right and top to bottom in a raster pattern.
6. Touch the **Show Params** checkbox to display parameters BER, Nbits, False0, and False1.
7. You can use the fields under **Frame**, to align the same bits one on top of the other so that bit failures will appear as vertical lines.



Touch inside the **Mode** field and make a selection from the pop-up menu.

8. If you selected **Header** or **Size and Header**, touch inside the **Frame header** field and enter a hexadecimal number representing the prefix before the actual data. This prefix will be ignored and only the data will be examined. If you selected **Size** or **Size and Header**, touch inside the **Frame size** field and enter the number of bytes in the frame, using the pop-up keypad. **Frame size** divides your waveform into equal pieces of the size that you enter.

Serial Trigger

Warning

To prevent damage to the serial trigger module, never apply external voltages to the clock and data output connectors.

The SDA "A" models include a serial pattern trigger, and a Clock and Data Recovery module. This CDR module is built into the instrument and is accessible through the channel 4 input. The signal on channel 4 is always present at the input to the serial pattern trigger module. The serial trigger also includes two outputs that are located on the front panel just to the left of the channel 4 input. These two SMA jacks allow access to the recovered clock and data signals from the serial trigger module. The signals at these connectors are nominally zero mean with a peak-to-peak amplitude of 330 mV.

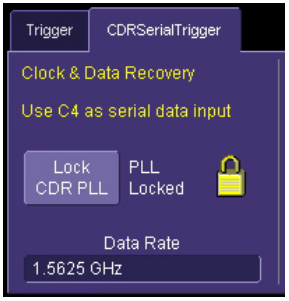
Serial Trigger Setup

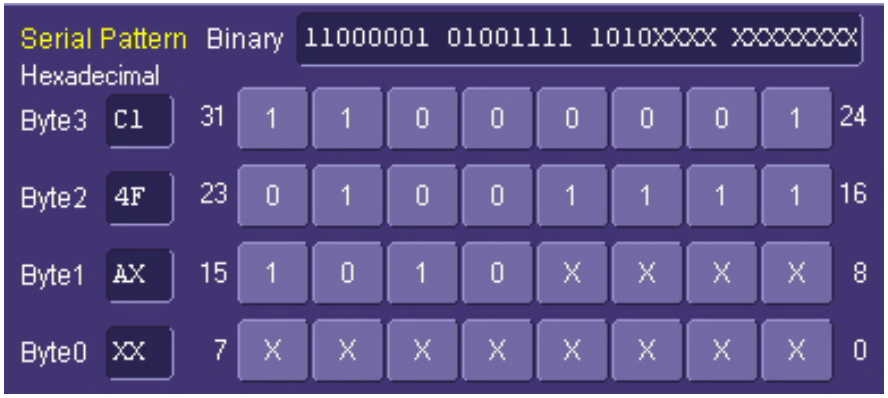
The serial trigger dialog is accessible from two different menus.

- By touching the **Trigger** descriptor label 

, then the **Serial Trigger** button 

, you can access the trigger phase locked loop and set the data pattern on which to trigger:





- It is also accessible from the channel 4 "Vertical Adjust" dialog:



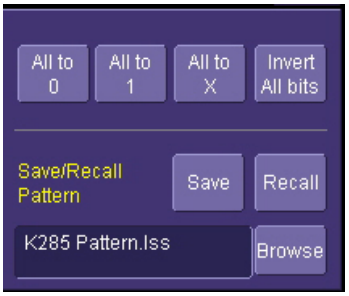
PLL Locking

The PLL in the serial trigger module must be locked for the trigger to operate. Locking is indicated by the closed padlock icon . The icon becomes an open padlock and flashes when the PLL is not locked. Lock the PLL by pressing the **Lock CDR PLL** button 

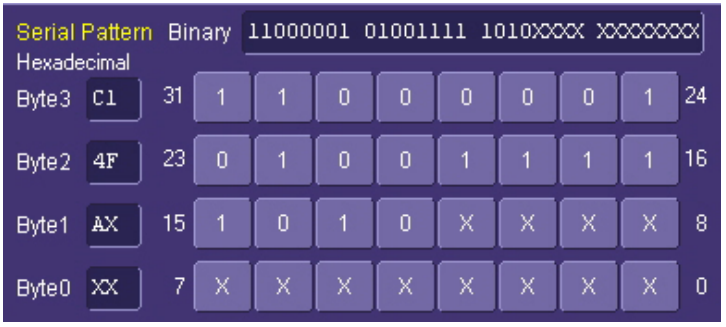
. Once locked, a "PLL Locked" message appears and the data rate is displayed. The PLL will remain locked until the signal is removed from channel 4 or its data rate is changed, regardless of whether the triggering mode is set to serial or not.

Setting the Data Pattern

The 32 bits in the pattern are set using the controls on the right half of the "CDRSerialTrigger" dialog:

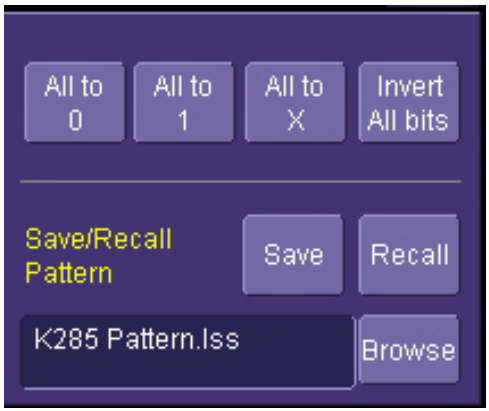


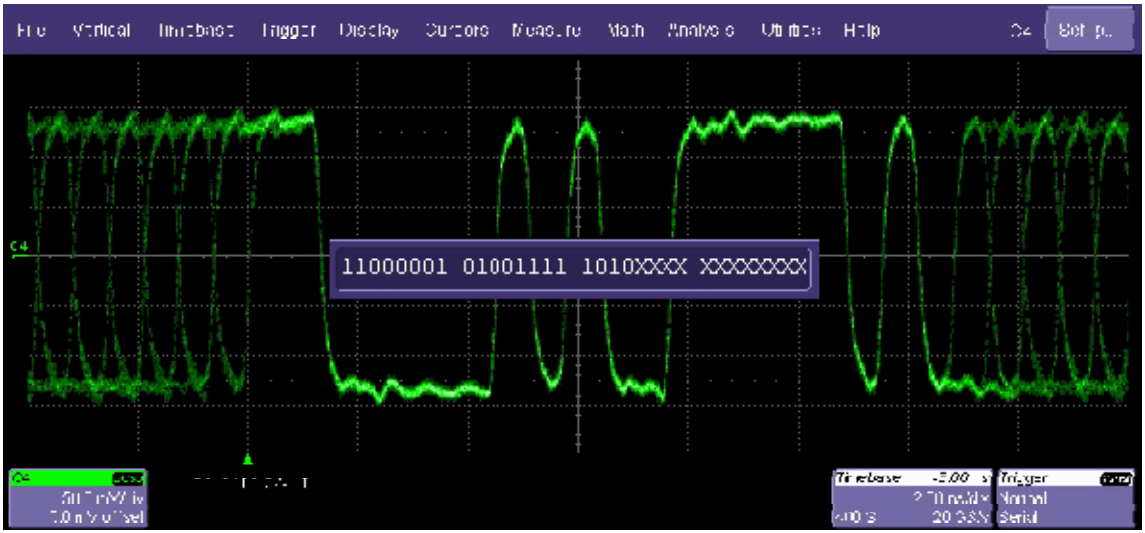
Using the matrix of buttons, you can set the value of each bit in the pattern to 1, 0 or X (dont care). The values are selected by touching each button and cycling through to the desired value:



The buttons are arranged in four rows, each row corresponding to an 8-bit word, with corresponding bit values displayed alongside each row of 8 buttons. Hexadecimal digits corresponding to each group of 4 buttons are displayed to the left. Hex digits that contain one or more don't care values are displayed as an **X**. The 32-bit binary pattern is displayed in the box above the matrix of buttons.

A group of buttons is provided to quickly set the pattern to all 1s, all 0s, or all Xs. And a fourth button is provided to invert the entire data pattern:





Example Bit Pattern and Resulting Waveform

Storing and Recalling Serial Data Patterns into the Trigger

It is often convenient to save frequently used trigger patterns in a file so that you do not have to repeatedly reenter them. The serial trigger pattern is also stored in the setup file whenever you save the panel file through the File ➔ Save Setup dialog. So the trigger pattern can be recalled by simply recalling the corresponding panel file. However, the panel setup file saves the entire state of the instrument, which may not always be desirable. For this reason, the SDA provides a way to save the serial trigger pattern separately.
The **Save/Recall Pattern** control can be used to save the pattern alone.

Using the Serial Pattern Trigger

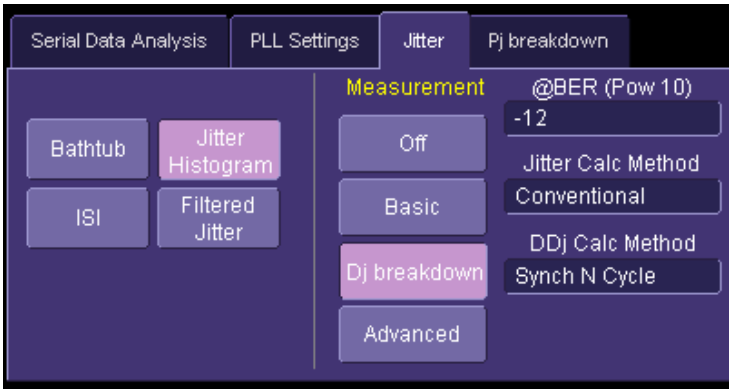
To select the serial pattern mode, press the **Serial** button 

in the Trigger Setup dialog. The trigger point appears at the start of the pattern in the data stream on the display, and the horizontal position knob can be used to position the trace in the same way as for other trigger modes.
The trigger can be set to single, normal, or auto. In auto mode, the scope will arm and trigger itself even if the selected pattern is not detected. However, serial pattern triggering is only valid in the normal or single trigger modes.

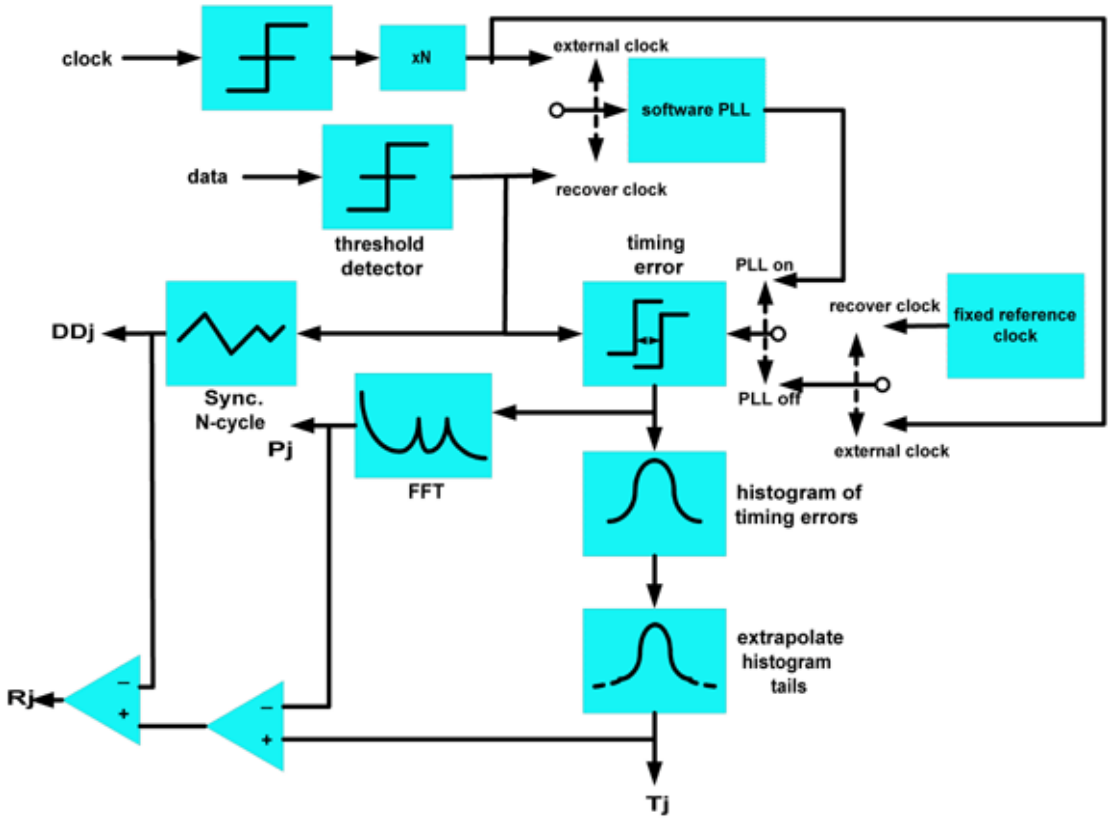
Jitter Setup

Jitter measurements are enabled by touching the Jitter button once the signal is set up. The Jitter dialog contains four display buttons and four measurement buttons. The displays include **Bathtub**, **Jitter histogram**, **ISI**, and **Filtered Jitter**. You can switch display modes without re-acquiring the signal.
The Measurement buttons control which jitter measurements are displayed. The **Basic** button displays the total jitter (Tj), Random jitter (Rj), and deterministic jitter (Dj). The **Dj Breakdown** button displays the components of Dj — periodic jitter (Pj), duty cycle distortion (DCD), and data dependent jitter (DDj) — in addition to the basic Rj, Dj, and Tj. The ASDA-J option adds the **Advanced** button which, when selected, replaces the Tj, Rj, Dj display with the peak-to-peak and rms values of the filtered jitter.

The SDA measures jitter by determining the timing error between the edge crossings of the data signal under test and a reference clock, which can be derived either from the data or an external clock. The measurement can also be performed directly relative to the edges of the external clock. A flow diagram of the SDA jitter measurement is shown as follows:



Total jitter is derived from the histogram of timing error measurements. The histogram of measurements is accumulated and the tails of the distribution are extrapolated to form an estimate of the histogram for a population of 10^{16} measurements. The extrapolated histogram is then integrated to determine the cumulative distribution function which forms the sides of the bathtub curve. The bathtub curve gives the eye opening (or 1UI - Tj(UI)) for all bit error rates down to 10^{-16}. The total jitter for a given bit error rate is found by measuring the width of the bathtub curve at the y-axis value corresponding to the desired bit error rate. The periodic jitter and data dependent jitter are measured directly on the sequence of timing error measurements, and the random jitter is found by measuring the difference between the total jitter and the components of deterministic jitter.

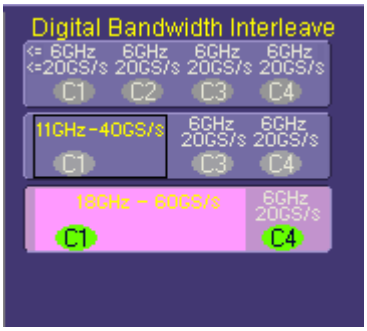

clock
xN
external clock
software PLL
data
recover clock
threshold
detector
timing
error
PLL on
recover clock
fixed reference
clock
DDj
Sync.
N-cycle
Pj
FFT
PLL off
external clock
histogram of
timing errors
extrapolate
histogram
tails
Rj
Tj


File
Vertical
Timebase
Trigger
Display
Cursors
Measure
Math
Analysis
Utilities
Help
BT...
Setup...
Measure
P1:(Tj)
P2:(ConvRj)
P3:(ConvDj)
P4:(Pj)
P5:(DCD)
P6:(DDj)
value
62.78 ps
2.36 ps
29.5 ps
5.1 ps
2 ps
24.4 ps
status
Timebase 0.00 µs
1.00 µs/div
200 kS 20 GS/s
Trigger
Stop 0.0 mV
Edge Positive
Serial Data Analysis
PLL Settings
Jitter
Close
Bathtub
Jitter Histogram
ISI
Filtered Jitter
Measurement
Off
Basic
Dj breakdown
Advanced
@BER (Pow 10)
-12
Jitter Calc Method
Conventional
DDj Calc Method
Synch N Cycle
setup
theory
LeCroy
10/29/2004 2:22:37 PM

SDA DBI Controls

The SDA 18000/13000/11000/9000 oscilloscopes achieve higher bandwidth and sample rate using an innovative LeCroy technology called Digital Bandwidth Interleaving (DBI). This technology allows resources to be borrowed from unused channels to multiply not only sample rate but also bandwidth.

SDA18000 DBI Controls



When 18 GHz is selected, the active channels are either C1 or C4 or both, providing 60 GS/s sampling on the 18 GHz channel, and 20 GS/s sampling on the 6 GHz channel.

When 11 GHz is selected, the active channels are C1, C3 and C4. The 11 GHz channel provides 40 GS/s sampling, while the 6 GHz channels provide 20 GS/s sampling.

When 6 GHz is selected, all 4 channels are active, each providing 20 GS/s.

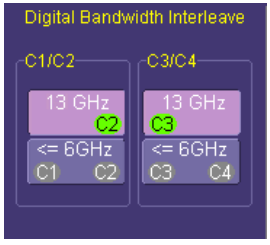
	18 GHz mode		11 GHz mode		6 GHz mode
Bandwidth	18 GHz	6 GHz	11 GHz	6 GHz	6 GHz
Number of Channels	1	1	1	2	4
Sample Rate	60 GS/s	20 GS/s	40 GS/s	20 GS/s	20 GS/s

SDA 13000/11000/9000 DBI Controls



When 13/11/9 GHz is selected, the active channels are either C2 or C3 or both, providing 40 GS/s sampling on each channel. For 6 GHz bandwidth, all four channels can be used, providing 20 GS/s sampling on each channel. Channels can be set to allow 13/11/9 GHz and 6 GHz bandwidths at the same time, as shown at left.

Bandwidth	13/11/9 GHz	6 GHz
Number of Channels	2	4
Sample Rate	40 GS/s	20 GS/s



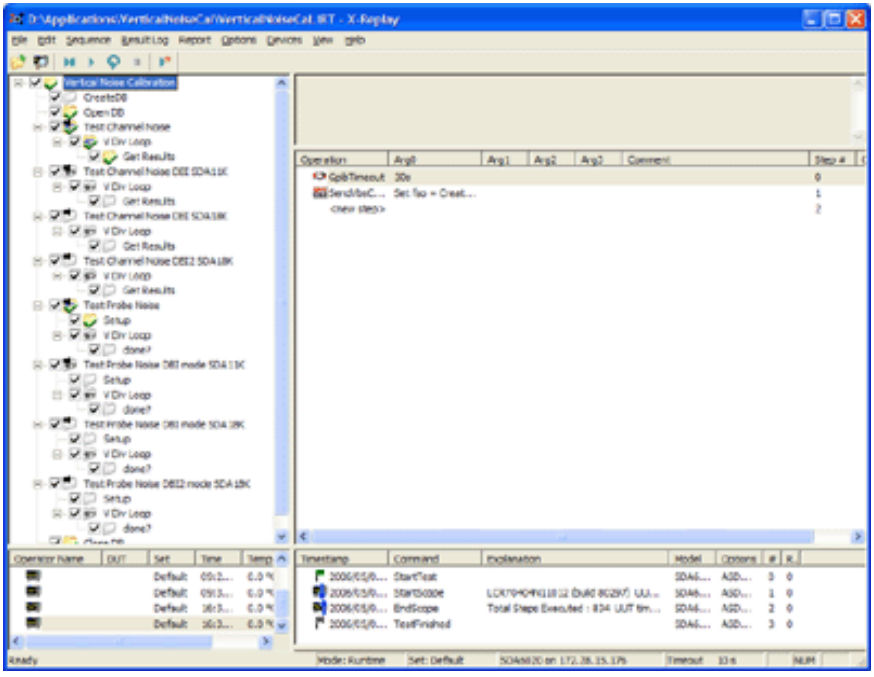
Vertical Noise Calibration

Your SDA contains a script (.irt) to calibrate vertical noise. The script runs in LeCroy's X-Replay environment, which is launched automatically when you open the script. By means of the script, the SDA accounts for vertical noise when calculating jitter. This feature applies only in Jitter mode.

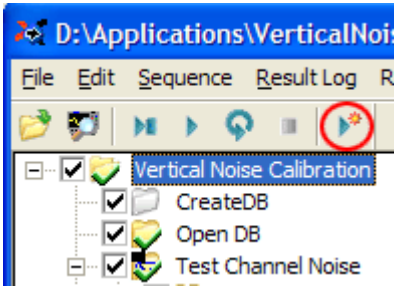
If your SDA does not have X-Replay loaded, this free utility is available for download from www.lecroy.com

Running the Script

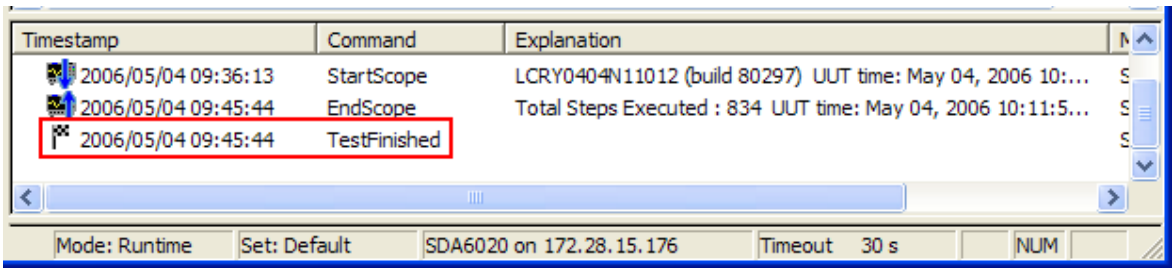
1. Minimize the scope application by selecting **File**, **Minimize**.
2. Open Windows Explorer and navigate to D:\Applications\VerticalNoiseCal\VerticalNoiseCal.irt. The X-Replay window opens:



3. In the X-Replay toolbar, touch the **Start a New Session** button

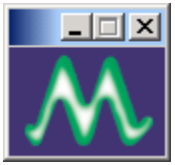


4. Enter any pertinent information in the "Session Info" dialog box if desired, and touch **OK**.
5. A "Required connections" dialog box opens, which prompts you to detach any probes connected to the scope. Touch **OK** when done, or you may **Skip this test group**.
6. If you opted to run the first part of the calibration in the previous step, each channel is tested at two offset positions for each volts/div setting. Then a second dialog box opens, which prompts you to attach any probes desired. Touch **OK** when done, or you may **Skip this test group**.
7. When testing is finished, notification is given in the lower right-hand pane of the X-Replay window:



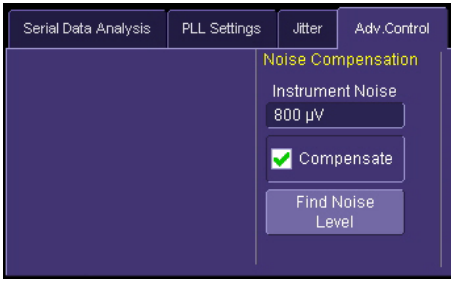
Using Noise Compensation

1. Maximize the scope application window by touching the SDA task bar button or desktop icon:

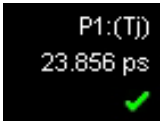


2. Press the front panel **SERIAL DATA** button to open the "Serial Data" main dialog.
3. Touch the **Jitter** button, then the **Adv. Control** tab.

4. Touch the **Compensate** checkbox to enable noise compensation, then the **Find Noise Level** button. The value displayed under "Instrument Noise" is for the active channel at the current volts/div setting. If you change channels or volts/div, press the **Find Noise Level** button again:



Jitter Measurements



Tj Total jitter at a specific BER. The value is determined by the width of the curve expressing the confidence limits of the extrapolated TIE histogram. The value represents the expected range of values observed for a number of observed measurements equal to 1/BER. The Tj parameter is accumulated over all acquisitions since the start of the measurement or the last clear sweeps operation. The total population of the TIE histogram is indicated in the waveform descriptor box for the bathtub curve and Htie (the TIE histogram).



Rj Random jitter is obtained indirectly by first determining deterministic jitter through various direct analyses of the TIE trend. Once all of Dj is found, the Rj value is deduced from the equation Rj = (Tj(BER) - Dj)/ $Tj_{(1\text{-sigma})}$(BER), where the value for BER is typically 10e-12 (i.e., Rj is not a function, but a single value). The bit error rate is selected in the **@BER (pow 10)** control in the jitter menu. Rj is expressed in terms of an rms value, whereas Tj and Dj are expressed as peak-to-peak values.



Dj Deterministic jitter is the peak-to-peak non-random part of the total jitter. This parameter is the sum of the measured peak-to-peak values of periodic jitter (Pj) and data dependent jitter (DDj). The DDj includes the effects of both inter symbol interference and duty cycle distortion.



Pj Peak-to-peak magnitude of the periodic components of the TIE trend. It is measured by analyzing peaks in the Fourier transform of the trend of the time interval error. It is necessary to specify the pattern repetition length (if there is one) so that spectral lines harmonically related to this pattern rate do not contribute to the estimate of Pj, since the spectral energy associated with data pattern-related spectral components is included in the DDj measurement. The total periodic jitter is the complex sum of the spectral components listed in the table under the Pj breakdown tab.



DCD Duty Cycle Distortion is the mean difference between the width of positive going pulses (low to high to low) and negative going pulses (high to low to high) measured over all pulses in the acquired waveform. The widths are measured at the same amplitude as specified for TIE (i.e., not necessarily at 50% of the signal amplitude). This measurement is a component of DDJ and included in the DDj value.



DDj The peak-to-peak jitter caused by systematic effects related to the sequence of data transitions.

Pj Breakdown

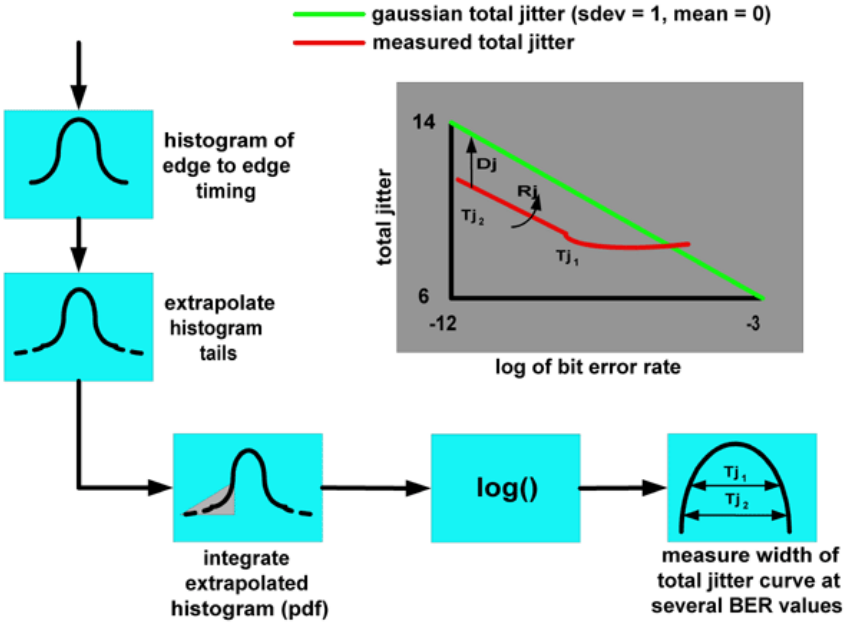
The Pj Breakdown tab reveals a table of components of periodic jitter. This table lists the peak-to-peak amplitude and rate (frequency) of each Pj component. The components are listed from largest to smallest. The Pj readout below the grid on the display is the complex sum of the components listed in this table.

Alternate Jitter Breakdown Methods (option ASDA-J only)

The ASDA-J option adds two additional jitter breakdown methods. These methods are termed **Effective** and **MJSQ** and are selected in the **Jitter Calc. Method** control when the instrument has the ASDA-J option present. The Effective and MJSQ methods provide alternate ways of determining the random and deterministic jitter but do not include the breakdown of deterministic jitter into periodic and data dependent parts. When either of these modes is selected, the jitter breakdown button the jitter menu becomes grayed (unavailable) and only the basic jitter display is shown. The DDj plot can still be viewed in this mode; however, the information from this plot is not used in the computation of Dj.

Effective Jitter

The effective jitter mode is entered when **Effective** is selected in the **Jitter Calc Method** control. Effective jitter is determined from the measured total jitter by evaluating the total jitter at several bit error rate values and solving Tj = $Tj_{(sigma=1)}$(BER)*Rje + Dje. The term $Tj_{(sigma=1)}$ is the total jitter of a Gauss Ian (normal) distribution of jitter with a standard deviation of 1 second. The two unknowns in this equation (Rje and Dje) are found by solving for several Tj values at BER levels below 10^{-10}. The jitter breakdown is the best-fit to the bathtub curve for very low BER values, but does not take into account the jitter contribution at the very top of the bathtub curve. The figure below shows the flow of the effective jitter measurement.



P2:(EffeRj) 1.466 ps

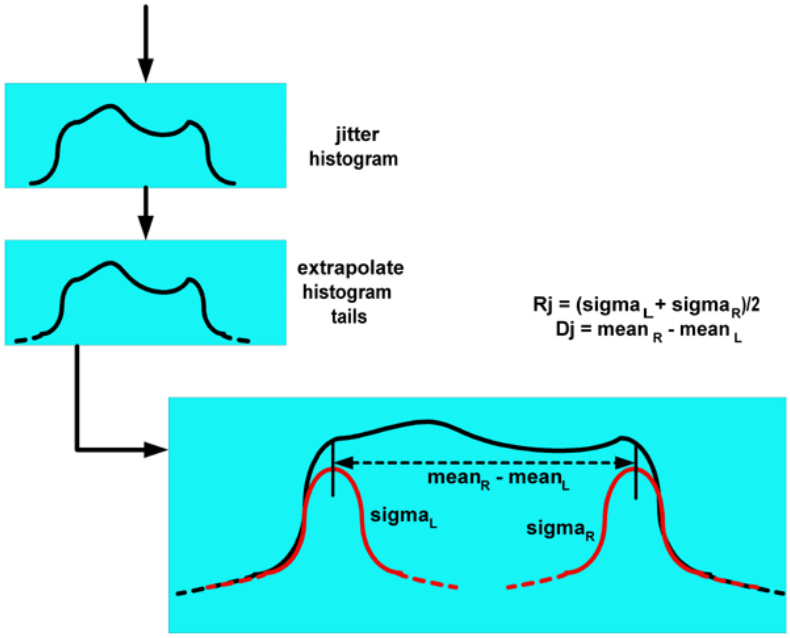
EffeRj Effective Random Jitter is one of the two constants used to estimate the function Tj(BER) for very low BER. Rje is obtained through the analysis of the growth of the extrapolated histogram of TIE as a function of BER, fitting the form: Tj(BER) = Dje + $Tj_{(sigma=1)}$(BER)*Rje

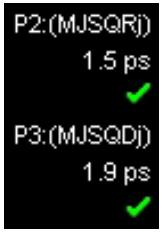
P3:(EffeDj) 3.536 ps

EffeDj Effective Deterministic Jitter is one of the two constants used to estimate the function Tj(BER) for very low BER. Rje is obtained through the analysis of the growth of the extrapolated histogram of TIE as a function of BER, fitting the form: Tj(BER) = Dje + $Tj_{(sigma=1)}$(BER)*Rje

MJSQ Jitter

The other method of determining the random and deterministic components is based on the procedure described in the Fibrechannel MJSQ (Methods for Jitter and Signal Quality) document. This method is similar to the effective jitter described above except that it operates directly on the jitter probability density function (pdf). A pair of normal distributions (one for each side) is found by adjusting their standard deviations so that they best fit the tails of the measured pdf. The difference between their mean values is the Dj and the average of their standard deviations is the Rj. This method is shown as follows.





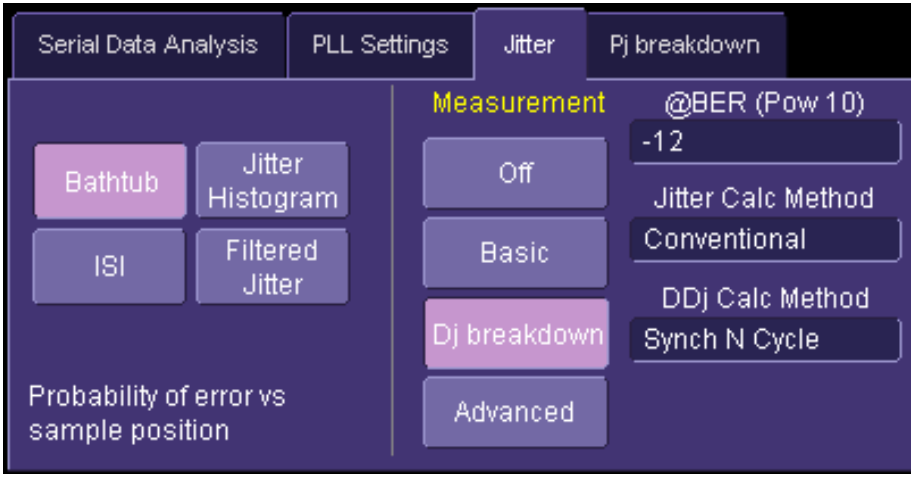
MJSQRj MJSQ Random Jitter is determined by finding the average of the two standard deviations of the best-fit Gaussian distributions to the left and right sides of the measured jitter probability density function.

MJSQDj MJSQ Deterministic Jitter is the difference between the mean values of the best-fit Gaussian distributions to the left and right sides of the measured jitter probability density function.

Bathtub Curve

The bathtub curve is the integral of the jitter probability density function (PDF – derived by normalizing the extrapolated TIE histogram - see the Theory section) for all possible sampling points within one unit interval (UI). The right side of the bathtub curve is the integral of the jitter pdf from approximately -1/2 of a UI to zero and the left side is the integral from +1/2 of a UI to zero, where the jitter PDF is centered about zero, and zero is defined as the ideal crossing time of a bit. The bathtub curve is normalized to 1 UI. It is a logarithmic representation of bit error rate (BER) vs. the sampling point. The greater the degree of separation between the sides of the bathtub curve, the better the BER.
The horizontal distance between the sides of the bathtub curve at a given Y value (bit error rate) is a measure of the eye width at that bit error rate.

1. Press the SDA front panel button; or, touch **Analysis** in the menu bar, then **Serial Data** in the drop-down menu. The display defaults to the "Scope" dialog, which enables you to set up SDA measurements while you continue to view currently displayed waveforms.
2. Touch the **Jitter** button. The Jitter Bathtub setup dialog is displayed. You can enable several parameter measurements from this dialog:



3. For basic measurements, touch the **Basic** button, then touch inside the **@BER(Pow 10)**[9] field. Enter an exponent value, using the pop-up keypad.

[9] The bit error rate (BER) is the percentage of bits that have errors relative to the total number of bits received in a transmission. Thus, a BER of 10^{-6} means that out of 1,000,000 bits transmitted, one bit was in error. The BER is an indication of how often a packet of data must be retransmitted because of error. The setting of the @BER control affects the indicated jitter values by selecting the vertical level at which the width of the total jitter (integrated TIE histogram) is measured. In general the Tj value will increase with decreasing bit error rate (more negative power of ten).

Jitter Filter

The jitter filter is a band-pass filter that is applied to the TIE-vs.-time data measured on the signal under test. The filtered waveform of the trend of TIE over time is displayed in the waveform grid when Filtered Jitter is selected. The filter is implemented as an FIR (Finite Impulse Response) digital filter with a maximum tap length of 2000. Not all high-pass filters are realizable, owing to the tap length restriction or the data record size or both. The instrument will apply a low-pass filter if the band-pass filter cannot be realized. The filter is applied only to the filtered jitter trace and does not affect the other jitter measurements: Tj, Rj, Dj etc. The peak-to-peak and rms value of the filtered jitter trace will be displayed below the trace.
The clock recovery PLL results in a jitter spectrum with a high-pass characteristic determined by the type and loop bandwidth of the PLL (see the Theory section). The combination of the low-pass jitter filter and the PLL cutoff will result in a band-pass filter for cases where the jitter filter function cannot realize the high-pass filter selected.

1. Touch the **Filtered Jitter** button.
2. Touch inside the **Lower Limit** field and enter a value, using the pop-up keypad. To realize a low pass-filter, select **Set to min.** in the pop-up keypad. In this case the lower frequency cutoff will be set by the PLL cutoff frequency.



3. Touch inside the **Upper Limit** field and enter a value, using the pop-up keypad.

TIE Histogram

A TIE histogram is often useful to display the raw measured histogram of the measured jitter values because it can give clues as to the sources of deterministic jitter and it also gives a good indication of whether the extrapolated total jitter value is likely to be accurate. The bins in the histogram at the extreme tails (the last 5 to 10 bins on each side) are used to determine the best-fit curve (see the Theory section for more details). Generally, if the histogram is smooth between the two tails, that is, there are no deep troughs in the shape, the curve fitting and extrapolation will give an accurate and stable Tj value.
There are some cases where very large amounts of ISI or periodic jitter cause deep troughs in the shape of the histogram. In these cases, a very large population is required in the histogram to ensure a sufficient population in the tail regions to achieve a reliable fit. Generally 1.5 to 2 million measurements in the histogram are sufficient to resolve the tails, but more data is always helpful. A histogram that appears smooth is a good indicator that the total jitter will be accurate.

The TIE histogram shows the measured jitter distribution that is the source for the bathtub curve and total jitter parameter.

1. Touch the **Jitter Histogram** button. The histogram will be displayed in the center of the bathtub curve.
2. Touch the **Bathtub** button to toggle the histogram display off.

DDJ (Synchronous N-Cycle Plot)

The DDj (data dependent jitter) is measured on the data stream when the **Conventional** jitter breakdown method is selected. The DDj is measured by finding the average crossing time for each edge in the data pattern and comparing this to the nominal crossing time, based on the expected bit rate of the signal under test. This measurement requires a repeating data pattern to operate. If the data pattern in the signal does not repeat at least once in the acquired data record, the DDj parameter will indicate zero even though this type of jitter is present. A graphical display of the DDj can be viewed by clicking the **ISI** button in the **Jitter** menu. This display shows the position error of the average crossing point for each edge in the pattern.

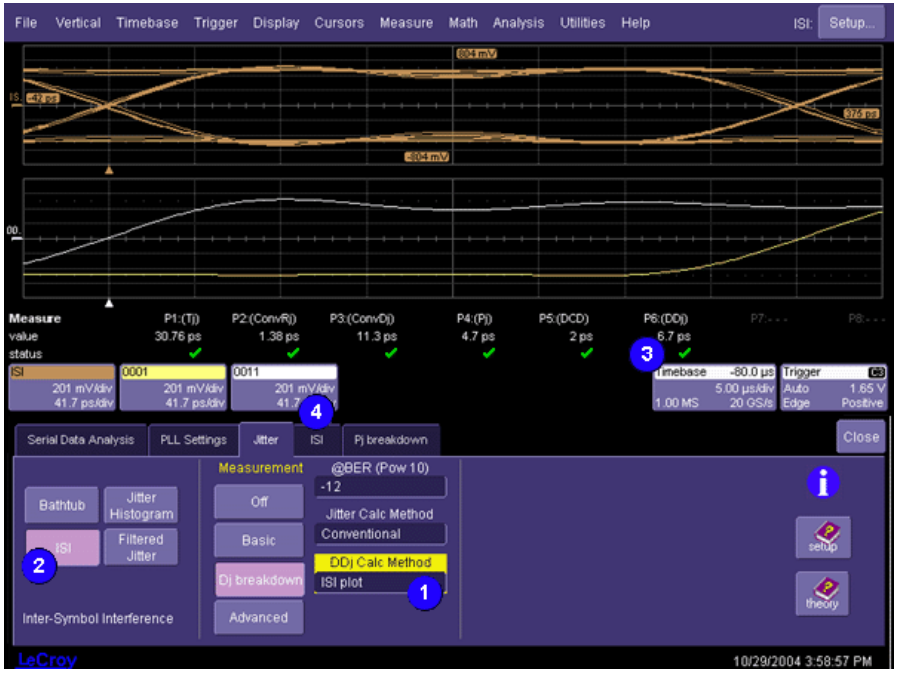


1. Touch inside the **DDj Calc Method** control and select **Synch N Cycle** for a repeating pattern (available for Edge-Ref or Edge-Edge method).
2. Touch the **ISI** button.

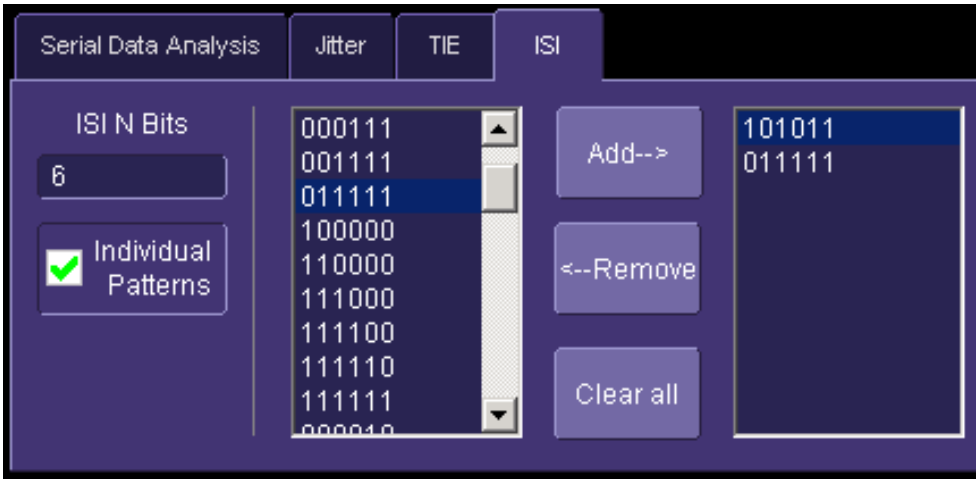
3. Note the change in the DDj measured value.
4. Touch the **ISI Edge-Edge** tab.
    A. Touch the **Detect Pattern** button to find the pattern and length. Normally this is done automatically when the signal is set up in the SDA main menu. The **Patt. Length** control in the SDA main dialog indicates the repetition length of the pattern.
    B. The pattern is indicated in Hexadecimal in the **Pattern** window. This window allows you to view and edit the pattern. The **Find Pattern Length** button in the SDA main dialog searches the pattern for specific hex values.
    C. Check the **Error Relative to First edge** checkbox to display the plot of edge position errors relative to the first transition in the data pattern.
    D. Check the **Show Graphic Bit Pattern** checkbox to display a trace showing the bit pattern in the grid with the synchronous n-cycle plot. The transitions of the pattern line up with the measurement points in the plot.

DDj (ISI plot) – ASDA-J Option Only

A repeating data pattern is not always present in cases where there is no ability to program the PHY under test. The ASDA-J option includes an alternate method of measuring data dependent jitter that does not rely on a repeating pattern. This plot shows an eye pattern consisting of the averaged waveform trace from each pattern in the data stream of a specified length, which can be adjusted from 3 to 10 bits. The value of DDj is the width of the zero crossing at the right side of the ISI eye at the threshold level set by the DDj threshold value.



1. Select “ISI Plot” in the **DDj Calc. Method** control for a random pattern (available only for Edge-Ref jitter method). This control is only available with the ASDA-J option.
2. Touch the **ISI** button; the ISI plot will be displayed.
3. Note the change in the DDj measured value.
4. Touch the **ISI** tab in the "Jitter" dialog.
    - Touch inside the **ISI N Bits** field to set the pattern length for the ISI plot.
    - Touch the **Individual Patterns** checkbox to display the bit patterns of the length specified in the **ISI N Bits** field. Only the patterns that exist in the acquired data stream is shown:



    - Touch any bit pattern in the list, then touch **Add** to display that trace. Up to seven patterns can be placed in the box at the right and displayed as traces.
    - Clear any pattern and trace from the screen by touching the pattern in the list box on the right, then touching **Remove**. Touch **Clear all** to remove all patterns.
    - The averaged traces become clearer as more occurrences of each pattern are observed. If one of the traces does not seem to get clearer, try reducing the number of bits considered **ISI N Bits** from 6 to a lower value.

Edge-to-Edge Basic Setup

The ASDA-J option adds a second measurement mode defined as edge-to-edge. In this mode, all jitter measurements are made between adjacent edges of the data pattern and no reference clock is used. Only jitter measurements are available in this mode, so the mask test, bit error rate, and summary buttons are inactive when the **Edge-Edge** mode is selected.



1. Touch inside the **Data Source** field and select a data source.
2. The "Crossing Level(s)" section in this dialog allows you to set the voltage level at which the signal timing is measured. It can be set to either absolute or percent. The absolute crossing level — in volts (or watts for an optical signal) — can be set directly or can be automatically found by touching the **Find Data Level** button. This level is determined by locating the midpoint between the highest and lowest signal levels in the current acquisition. The percent level is automatically set to the selected percentage on each acquisition.
3. If you selected absolute, touch inside the **Abs level** field and enter a voltage value. Alternatively, you can touch the **Find Data Level** or **Find Clock Level** button. If you selected Percent, touch inside the **Percent level** field and enter a percentage value, using the pop-up keypad. The level is automatically set to the selected percentage on each acquisition.

Note: The absolute level setting should be used for jitter measurements over several acquisitions because the relative setting can potentially remove jitter by tracking slowly, thereby varying level shifts that could occur between acquisitions.

4. Touch inside the **Signal Type** field and select a standard signal type from the pop-up menu. The signal type defines the bit rates for the selected standard.
5. If you selected **Custom** from the **Signal Type** menu, touch inside the **Bit Rate** field and enter a bit rate, using the pop-up keypad. By touching the **Find Bit Rate** button, you can have the bit rate determined automatically. This button is also available when a standard data type is selected.

6. The **Pattern Length** control sets the repetition length of the data pattern in the signal under test. Serial data signals generally transmit a repeating data pattern of some sort. For example a PRBS7 pattern repeats every 127 bits while a K28.5 repeats every 20 bits. Enter the pattern length in the **Pattern Length** control. Alternatively, the SDA can automatically find the pattern length by means of the **Find Pattern Length** button. The **Find Pattern Length** control determines the pattern repetition length (if any) in the current acquisition and sets the **Pattern length** control to this value. The Pj measurement uses the pattern length value to determine which peaks in the jitter spectrum are caused by the pattern repetition. An incorrect value in the Pattern Length control will result in overestimation of the Pj and, as a result, underestimation of the random jitter.
7. The **Tx Density** control indicates the ratio of transitions in the data record to the total number of bits in the signal. Normally, an NRZ data stream does not have transitions during every bit interval. For example, there are cases where adjacent 1 or 0 values exist. On average, approximately 50% of the bit intervals have transitions, so the Tx density is 500e-3. The actual acquired data set may have a transition density different from 50%, however. The total jitter measurement uses this transition density to normalize the measured histogram into a pdf [probability density function]. Many instruments simply assume 50% when performing this normalization, which can lead to errors. These errors are eliminated in the SDA through the use of the measured transition density in the normalization process. The Tx density is automatically measured when the Find Pattern Length button is pressed.

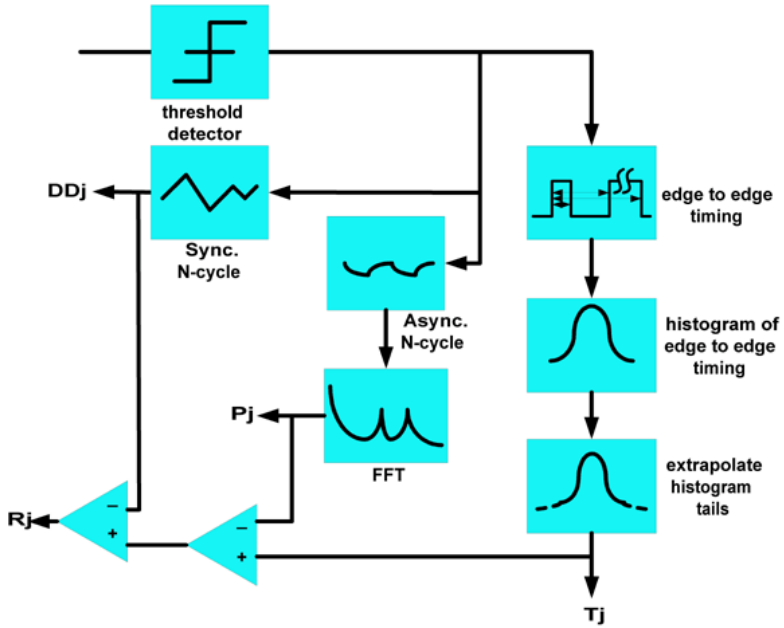
Edge-Edge Jitter Measurements

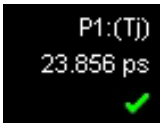
Jitter measurements are enabled by selecting **Jitter** in the SDA main menu when the mode is set to Edge-Edge. The available displays include a jitter bathtub curve, TIE histogram, ISI display, and N-cycle vs. N plot.

The Measurement buttons control which jitter measurements are displayed.

	The **Basic** button displays the total jitter (Tj), Random jitter (Rj), and the deterministic jitter (Dj).
	The **Dj Breakdown** button enables the display of the components of Dj --- periodic jitter (Pj), duty cycle distortion (DCD), and data dependent jitter (DDj) --- in addition to the basic Rj, Dj, and Tj.
	The ASDA-J option adds the **Advanced** button which, when selected, replaces the Tj, Rj, Dj display with the peak-to-peak and rms values of the filtered jitter.

The SDA measures jitter by determining the timing between the edge crossings of the data signal under test and comparing these measurements with the nominal unit interval (UI). A flow diagram of the edge to edge jitter measurement is shown below. The total jitter is derived from the histogram of timing error measurements. The histogram of measurements is accumulated and the tails of the distribution are extrapolated to form an estimate of the histogram for a population of 10^{16} measurements. The extrapolated histogram is then integrated to determine the cumulative distribution function, which forms the sides of the bathtub curve. The bathtub curve gives the eye opening (or 1UI - Tj(UI)) for all bit error rates down to 10^{-16}. The total jitter for a given bit error rate is found by measuring the width of the bathtub curve at the y-axis value corresponding to the desired bit error rate. The periodic jitter and data dependent jitter are measured directly on the sequence of timing error measurements and the random jitter is found by measuring the difference between the total jitter and the components of deterministic jitter.





Tj Total jitter at a specific BER. The value is determined by the width of the curve expressing the confidence limits of the extrapolated TIE histogram. The value represents the expected range of values observed for a number of observed measurements equal to 1/BER. The Tj parameter is accumulated over all acquisitions since the start of the measurement, or the last clear sweeps operation. The total population of the TIE histogram is indicated in the waveform descriptor box for the bathtub curve and Htie (the TIE histogram)



Rj Random jitter is obtained indirectly by first determining deterministic jitter through various direct analyses of the TIE trend. Once all of Dj is found, the Rj value is deduced from the equation Rj = (Tj(BER) - Dj)/ $Tj_{(1\text{-}sigma)}$(BER), where the value for BER is typically 10e-12 (i.e., Rj is not a function, but a single value). The bit error rate is selected in the **@BER (pow 10)** control in the jitter menu. Rj is expressed in terms of an RMS value, whereas Tj and Dj are expressed as peak-to-peak values.



Dj Deterministic jitter is the peak-to-peak non-random part of the total jitter. This parameter is the sum of the measured peak-to-peak values of periodic jitter (Pj) and data dependent jitter (DDj). The DDj includes the effects of both inter-symbol interference and duty cycle distortion.



Pj Pj is the peak-to-peak magnitude of the periodic components of the TIE trend. It is measured by analyzing peaks in the Fourier transform of the trend of the time interval error. It is necessary to specify the pattern repetition length (if there is one) so that spectral lines harmonically related to this pattern rate do not contribute to the estimate of Pj, since the spectral energy associated with data pattern-related spectral components is included in the DDj measurement. The total periodic jitter is the complex sum of the spectral components listed in the table under the **Pj breakdown** tab.



DDj The peak-to-peak jitter caused by systematic effects related to the sequence of data transitions.

Edge-Edge Jitter Measurement Controls

Adjust Rj

The TIE measurements in the edge-to-edge mode are measured by determining the timing between transitions of the data signal and comparing this to the nominal UI. The measurement is essentially the difference between two random values (the location of each transition in time). The random part of this TIE measurement (random jitter) is the sum of the random jitter values from each of the two edges and, as a result, is larger by a factor of the square root of 2 compared to the actual TIE (measured relative to a reference). The **Adjust Rj** control divides the measured edge-edge Rj by the square root of 2 to convert it to the equivalent edge-reference value. At the same time, the total jitter is recomputed when this control is selected by multiplying the adjusted Rj by the appropriate number of standard deviations determined by the selected BER (14 for 10e-12 BER). The result is that Tj and Rj more closely approximate the edge-reference measurements when the **Adjust Rj** control is checked.

Async N Cycle Plot

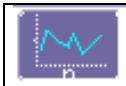
In the absence of a reference clock, periodic jitter cannot be directly measured from the data record because there is no timing reference on which to base the Fourier transform. The SDA employs a function called the Alternate N Cycle plot to measure periodic jitter on edge-to-edge TIE values. The plot displays the standard deviation of a histogram of the timing variations for transitions in the data signal that are spaced *n* UI apart. Each Y-value in this plot represents the standard deviation of the histogram for a given value of *n* as *n* varies from 1 to the maximum value set in the **Max N** field, the upper limit of which is determined by the number of UIs in the analyzed signal. The **HPF Corner Freq.** readout displays the minimum frequency jitter component that can be measured.

Edge-Edge Spacing Controls

The histogram and bathtub curve of TIE measurements are derived by determining the time interval between transitions of the data signal. The timing is measured between adjacent edges that are not always 1 UI apart, depending upon the data pattern. The **For Edge-Edge Use All Edges** checkbox, when checked, includes all adjacent transitions in its measurement. When this control is un-checked, only transitions that are the selected number of UI apart are measured. In this case, the measurement compares transitions 1 with N, 2 with N+2, 3 with N+3, etc. The **Use only UI spacing** control sets the UI spacing to be used in the jitter measurement.

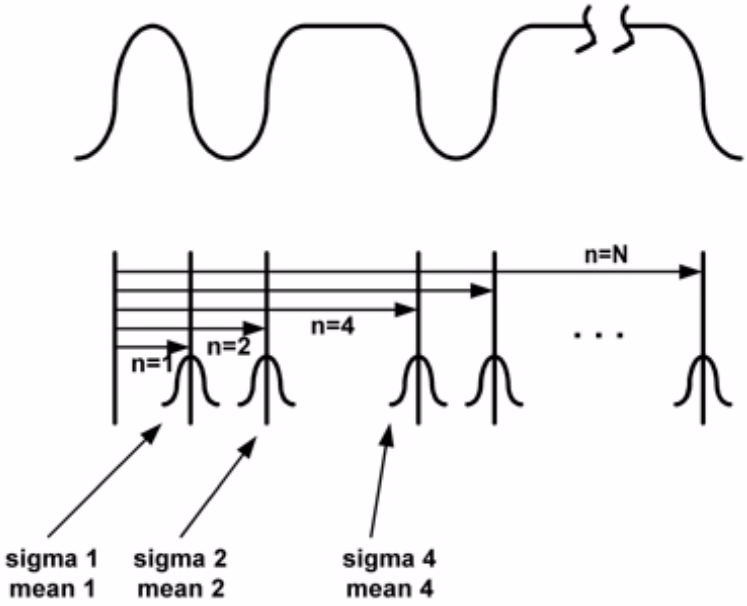
SDA FUNCTION REFERENCE

There are several specialized math functions that appear in the SDA math menu. These functions make up the building blocks of the SDA jitter and eye pattern measurements and are available from the **Select Math Operator** menu. The following table presents an overview.

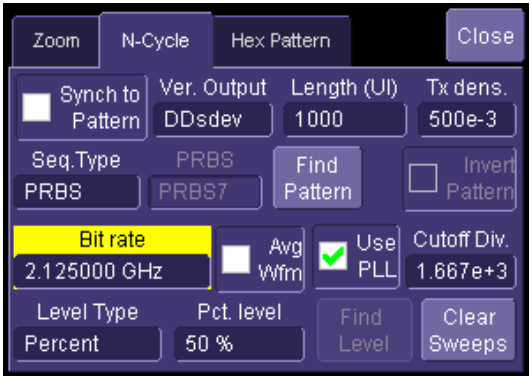
	AltNcycle	Alternate N-cycle jitter plot - this function displays a plot of the standard deviation of the jitter for all edges spaced N UI apart for 1 < N < Nmax. Where Nmax can be set up to 2M.
	Htie to BER	Extrapolates a histogram of time interval error measurements to the desired bit error rate. This function creates the bathtub and total jitter curves. Total jitter, as well as the effective and MJSQ breakdowns, use the output of this function.
	Jitter filter	Filter the trend of jitter values with the specified band-pass filter. This function expects a trend of jitter values as an input. It is further required that the measurements be time-continuous, that is, there is one measurement for each consecutive UI in the data signal.
	Slice2Persist	This function takes a waveform and divides it up into sequences that are 1 UI long based on a separate clock input. The output is a persistence map showing the eye pattern.

AltNcycle

Timing of the transitions in the data waveform is measured for each transition and plotted as a function of the number of unit intervals over which the timing is measured. The "N-Cycle" dialog is used to control how this measurement is performed, and the diagram and table below describe the function in detail. The N-cycle plot displays the mean or standard deviation of the edge placement in the waveform relative to each other (data to data) or to a reference clock (clock to data).



The N-cycle plot displays the mean or standard deviation of the distribution of timing for edges spaced n UI apart as a function of n. The value of n can vary from 1 to some maximum value, including all intermediate values, or can be synchronous with the transitions in a repeating pattern.



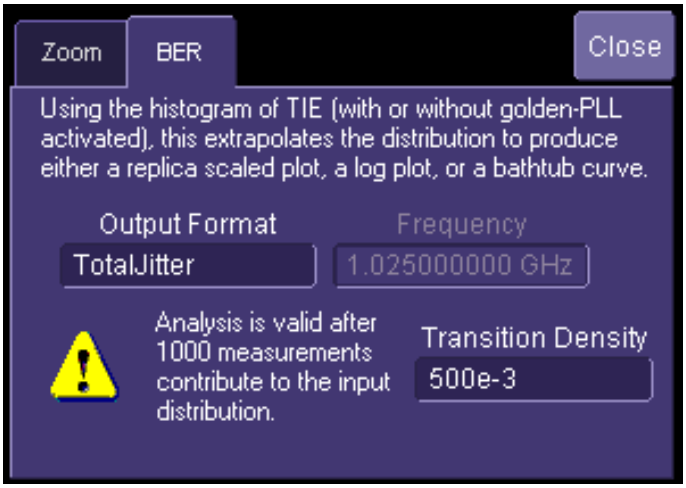
AltNcycle Control Summary

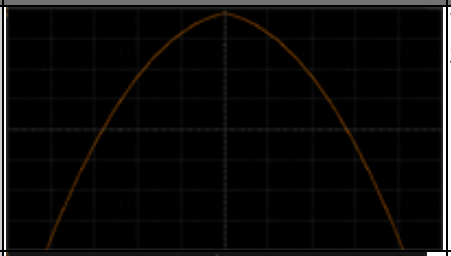
Control	Values	Description
Synch to pattern	On/off	Synchronizes the measurement to a repeating data pattern (if present) in the signal when this control is checked. The timing is measured between the first transition in the pattern and each subsequent transition. For an N-bit pattern, each transition is measured in succession and measurements are accumulated by shifting the measurement window by N bits and repeating the measurement.
Ver. Output	Mean, sdev, DDmean, DDsdev	These are the vertical units of the measurement. Mean gives the average position of edges for each value of N relative to a reference clock at the bit rate. Sdev gives the standard deviation for each edge for each value of N relative to the reference clock. DDmean gives the mean value of each edge relative to the first edge normalized to a unit interval for each N. DDsdev gives the standard deviation for each edge relative to the first edge normalized to a UI for each N.
Length (UI)	The maximum N value	This field displays the pattern length if **Sync to Pattern** is selected. Otherwise, this is the maximum N value displayed in the plot.
Tx dens.	0.0 to 1.0	Transition density of the data stream. This control reads out the measured transition density of the signal when the Find Pattern button is pressed. This field is not used by the N-cycle plot.
Seq. Type	PRBS or Pattern	Selects whether the repeating pattern is a standard PRBS or a general pattern. This control is only used when snyc to pattern is selected.
PRBS	5 to 15	When PRBS is selected, sets the length.

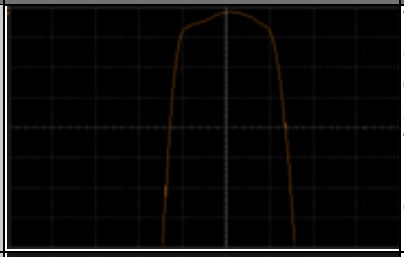
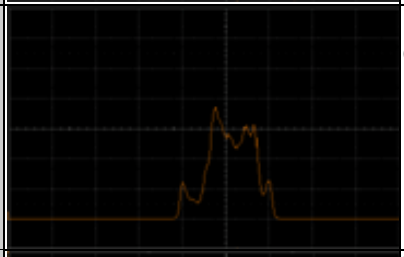
Control	Values	Description
Find Pattern		Pressing this button searches the signal for a repeating pattern and loads it into the function when it is found. The Length (UI) and Tx dens. controls are then updated.
Invert Pattern	On/off	Checking this box inverts the found pattern bit by bit.
Bit rate		The symbol rate of the signal under test. This value must be entered by the user in order to set the nominal UI.
Avg Wfm	On/off	Averages the N cycle plots measured on successive acquisitions. This is especially useful for removing random effects from standard deviation plots.
Use PLL	On/off	Use the software golden PLL to generate the reference clock for edge timing. This setting is ignored when DDmean or DDsdev is selected.
Cutoff Div.	PLL cutoff divisor	When the PLL is selected, this control sets the loop bandwidth as a ratio of the bit rate (loop BW = data rate/cutoff divisor).
Find Level		When the level type is absolute, clicking this control finds the mean signal level.
Clear Sweeps		Pressing this button clears the sweeps that have been accumulated in the averaging mode.

Htie to BER

This function takes as an input a histogram and converts it to one of 6 output waveforms. Normally, the histogram contains a set of time interval error (TIE) measurements from which a total jitter measurement is desired. The histogram should have a population of at least 100k and a minimum of 500 bins. The histogram should also be completely contained on the instrument screen (it may be necessary to set the horizontal scale to do this). The table below outlines the settings for this function.

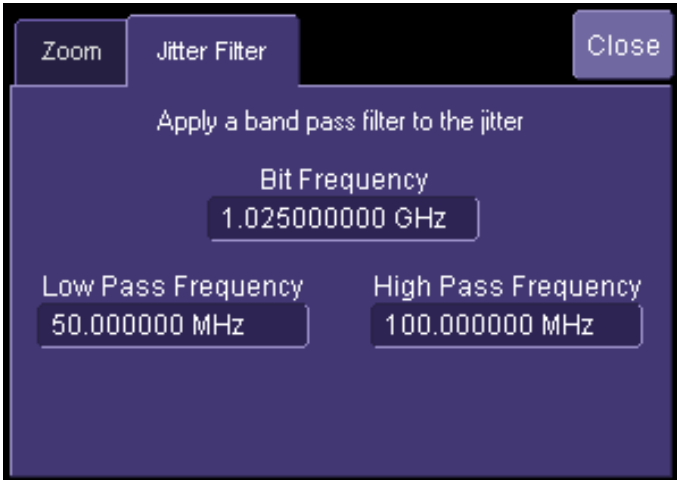


Output Format		Waveform Type
TjGaussian		The total jitter curve for a Gaussian with zero mean and unity standard deviation
LogEstTIE		The log of the estimated (extrapolated) time interval error histogram

Control	Values	Description
Total Jitter		The total jitter curve which, shows the cumulative probability of an edge being greater (right-hand side) or less (left-hand side) than its nominal location in the center. The total jitter at a particular bit error rate is measured from the width of this curve at the desired bit error rate given by the vertical location over which the width is measured.
EstTIE		Estimated tie histogram. This shows the original histogram of TIE measurements that has been extrapolated to a population of 10^{16}.
LogTIE		The log of the TIE histogram
Bathtub		The bathtub curve. This is a an alternate version of the total jitter curve scaled horizontally to 1 UI. The right-hand side of the curve is taken from the left-hand side of the total jitter curve, and the left-hand side comes from the right-hand side of the total jitter curve.
Frequency	100 kHz to 4.5 GHz	The bit rate of the signal. This control is only active when the bathtub output is active.
Transition density	0.1 to 1.0	The transition density in the data signal. Usually assumed to be 0.5. This represents the number of bit transitions compared to the total number of bits in the data stream. This value is used to normalize the extrapolated histogram.

Jitter Filter Function

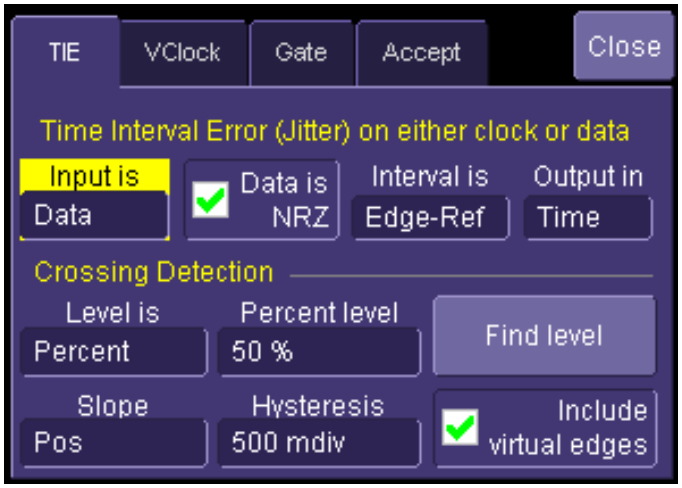
The jitter filter function generates a time sequence of jitter measurements that are filtered by the selected band-pass filter. Low-pass and high-pass frequencies are set in the function's menu. You must enter the bit frequency (data rate) to set the sampling rate for the digital filter. This function is only valid when operating on the TIETrend trace generated by the SDA. The low frequency corner is limited by the record length (longer records are required for lower cutoff frequencies). If an invalid lower cutoff frequency is chosen, the filter reverts to low-pass.



Bit Frequency	100 kHz to 4.5 GHz	The bit rate of the signal under test
Low Pass Frequency	Set by record length and sample rate (bit rate)	The lower corner frequency of the band-pass filter
High Pass Frequency	Set by record length and sample rate (bit rate)	The upper corner frequency of the band-pass filter

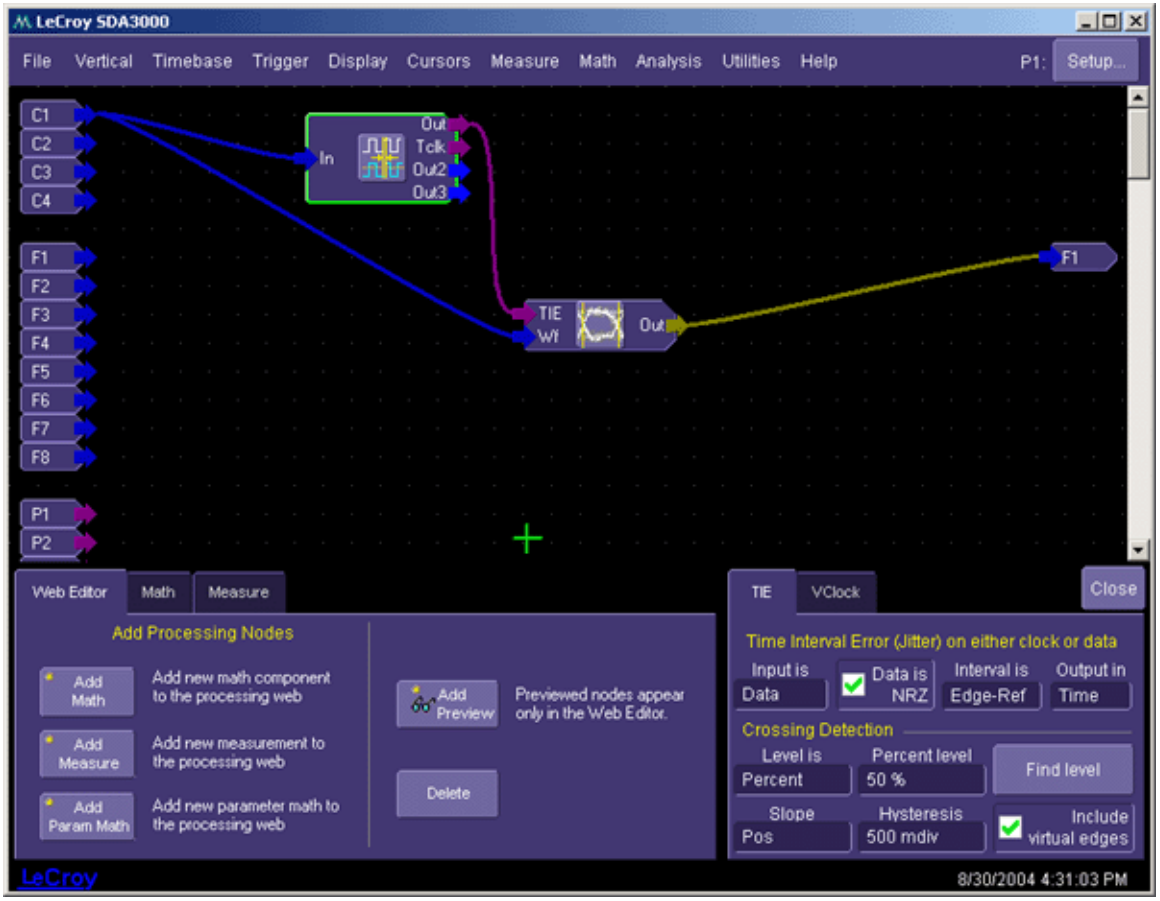
Slice2Persist

Note: This function can only be used in the Web Edit mode, available with the XWEB option.



The **Include virtual edges** checkbox must be checked in the TIE@lvl parameter dialog to include both transition and non-transition edges in the eye pattern.

The Slice2Persist function takes as its input a data waveform and a clock signal. The clock must consist of a sequence of time values that increase at the nominal unit interval for each successive sample. For example, a 2.5 Gb/s bit stream should have a clock input consisting of the numbers 0, 4e-10, 8e-10, 12e-10, …, which increase at a 400 ps per UI rate. The clock signal is derived from the output of the TIE@lvl parameter, which contains a sequence of adjustment values that are added to the nominal bit period for each UI. These correction values allow the TIE@lvl parameter to act as a phase locked loop. The web edit setup is shown as follows:



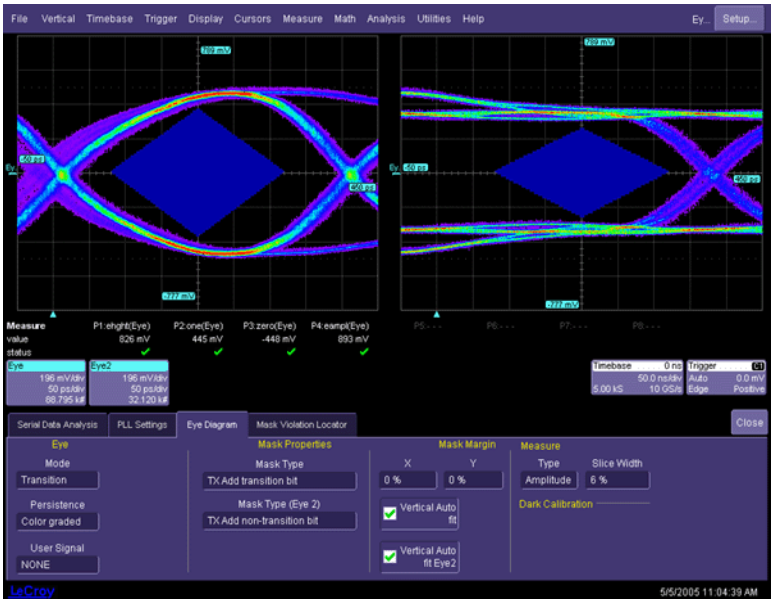
Frequency	100 kHz to 4.5 GHz	The bit rate of the signal under test. This number can also be set to a sub-multiple of the bit rate to show more than one eye. For example, 2.5 GHz will show one eye of a 2.5 Gb/s signal while setting the control to 1.25 GHz will display two eyes.
Auto Scale Vertical		Adjusts the vertical scale so that the one level is centered on 2.5 divisions above the screen center and the zero level is set to -2.5 divisions.
Auto clear sweeps		Clears the eye pattern display on each new acquisition.
Clear Sweeps		Clears the current persistence map from the display.
Scale Top to	4 V to -4 V	Scales the one level to a specific voltage on the display.
Scale Bottom to	4 V to -4 V	Scales the zero level to a specific voltage on the display.

MULTI-EYE MEASUREMENTS

Overview of Multi-Eye Measurement Tools

The Multi Eye Measurement Tool is designed to improve the analysis and processing of serial data signals in high-speed buses and interfaces commonly used in desktop, server, and storage applications. It is available with version 4.3 of X-Stream software in SDAs and WaveMaster DSOs. Its main features are:

- Three Additional Modes of Eye Diagram display:
- Front Side Bus (FSB) Application
- Transition/Non-Transition Bit Application (PCI Express)
- Gated (Qualified) Eye Diagram
- Eye Diagram Measurements available in every mode:
- Amplitude measurements (Height, Amplitude, Zero Level, One Level)
- Timing Measurements (Rise Time, Fall Time)
- Eye-Specific (Crossing, Eye BER)
- Seamless Integration with Standard masks:
- User-Selectable, Dual mask inputs
- Dual Mask Violation Testing



Transition Mode Display

Setup and Installation

Multi-Eye Setup and Installation

The Multi Eye Measurement Tool does not require any additional software installation, as it is fully integrated into the SDA software. However, depending on the specific mode of operation selected, each of the three new modes of operation requires distinct instrument and channel setups.

For example, the FSB mode requires setting up multiple channels to carry out the eye diagram measurements, as described in the example below. The Gated Eye Diagram requires at least two inputs: The Data and the Gate signal. The Transition Mode requires at least one input (the Data Stream, but depending on the specific serial data standard, a clock input may also be required to recover the clock information).

These signals (except for the strobes) are differential signals, so the use of differential probes is often required to carry out the measurements. As with any high-speed precision measurements, all probes and SMA cables should be calibrated and deskewed prior to any data collection.

Example Setups

FSB Eye Mode Configuration

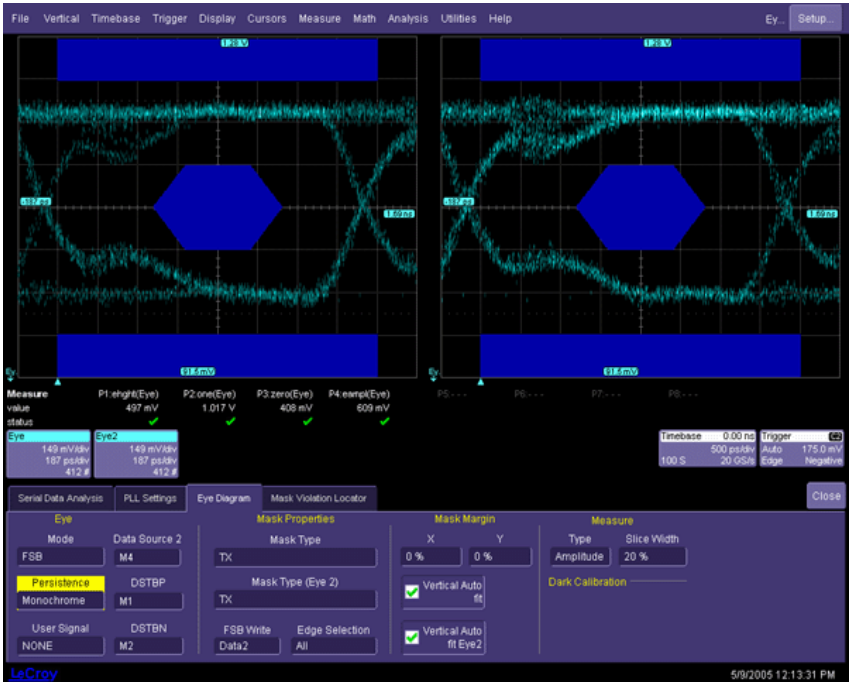
Channel Setup

- DataStrobep: Channel 1 (M1)
- DataStroben: Channel 2 (M2)
- Data1: Channel 3 (M3)
- Data2: Channel 4 (M4)

Serial Data Analysis Setup

1. Select Signal Type: FSB 533MHz, 667MHz or 800MHz
2. Uncheck Recover Clock - No clock recovery is required in FSB case
3. Select Eye Diagram Tab, then configure the screen as follows:
    - Mode: FSB
    - Data Source 2: Channel 4 (M4)
    - DSTBP: Channel 1 (M1)
    - DSTBN: Channel 2 (M2)

The screen should look like this:



Eye Diagram Setup and Measurements

1. Select Mask for each Eye Diagram - Depending on the Standard selected, there may be one or several types of serial data masks available.
2. Select a Mask Margin (as % of the nominal mask size) if desired.
3. Select a Measurement Type. There are three groups of parameters:

Amplitude

- Eye Height
- One Level (Eye)
- Zero Level (Eye)
- Eye Amplitude

Timing

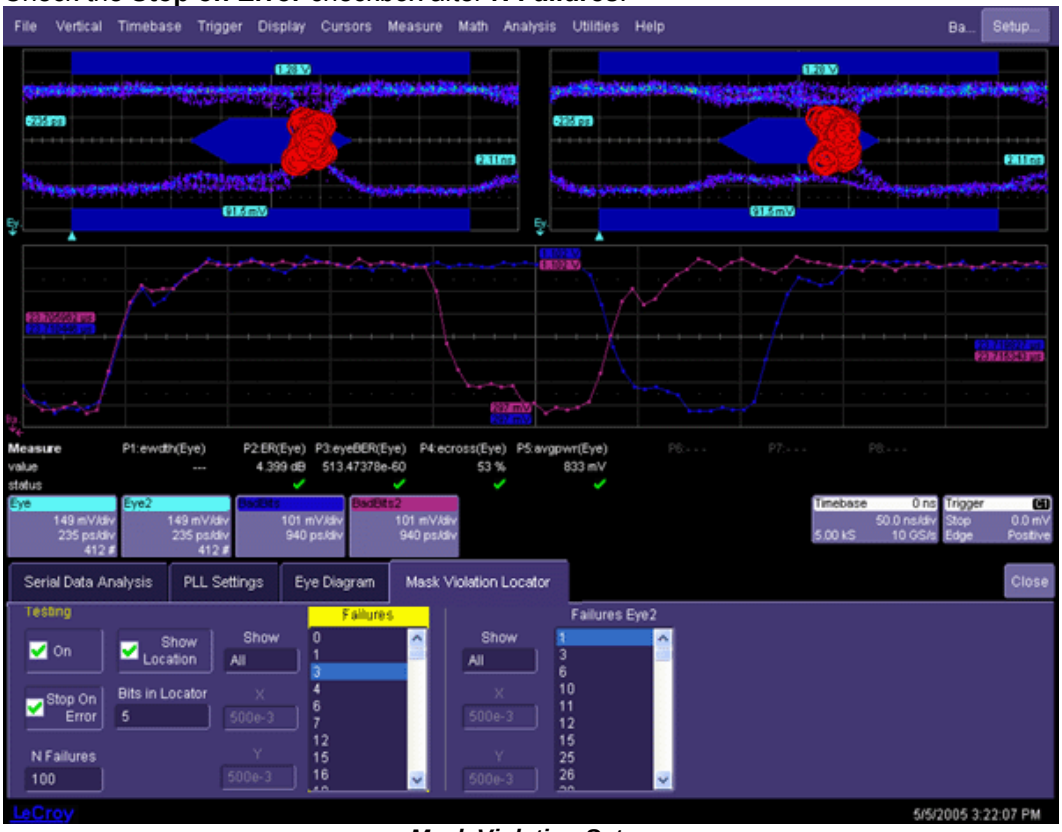
- Eye RiseTime
- Eye FallTime

Eye

- Eye Width
- Eye RMS Jit
- Eye BER
- Eye Crossing
- Avg Pwr (Eye)

Mask Test (Mask Violation Locator) Setup

1. Select the "Mask Violation Locator" tab.
2. Check the **Testing On** checkbox to detect mask violations.
3. Check the **Show Location** checkbox to select display of bits where mask violations occur.
4. Check the **Stop on Error** checkbox after **N Failures**.



Mask Violation Setup

Front Side Bus (FSB)

Introduction to FSB

Front Side Bus is also known as the processor bus, memory bus, or system bus. The FSB connects the processors to the main memory, and is used to connect to other resources within the computer. The FSB can range from speeds of 66 MHz, 133 MHz, 100 MHz, 266 MHz, 400 MHz, and higher.

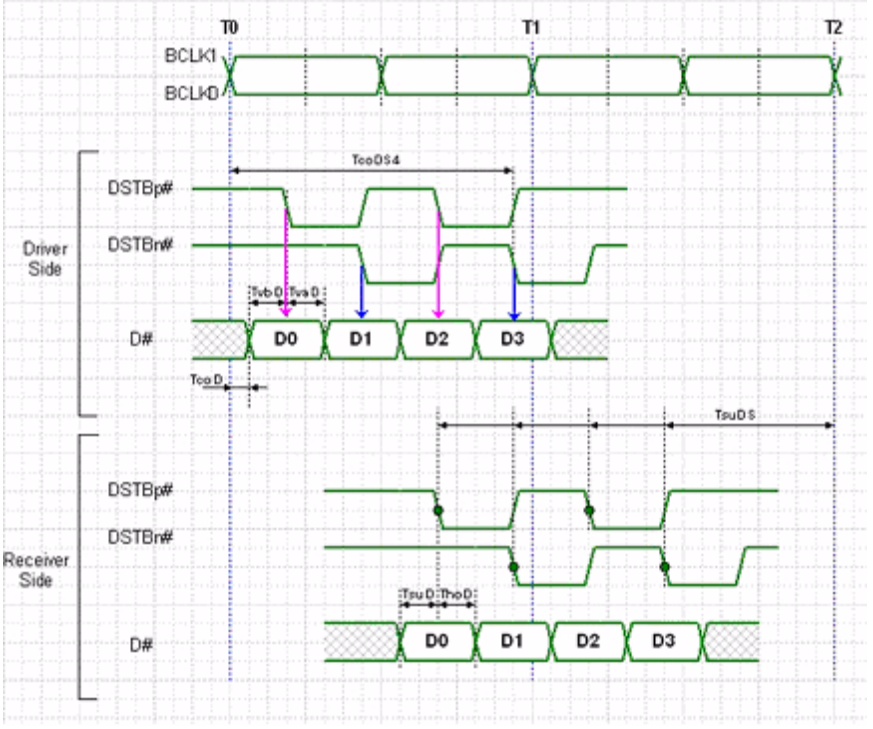
The FSB speed can generally be set either using the system BIOS or with jumpers located on the computer motherboard. FSB can accommodate one or two CPUs. This section of the manual describes single CPU operation.

FSB Theory of Operation

There are three sub-buses in the FSB: data bus, address bus, and common clock bus. All these buses are bidirectional; information can flow in either direction. The direction of the bit flow will be determined by the relative position of the high-to-low transition in the strobe pins (address, data or clock) when simultaneously probed at both ends of the applicable bus. For example, when signal A transition is delayed relative to signal B transition, this means that B is the transmitter and A is the receiver. The bus clock (BCLK) signal is formed by a differential pair, BCLK0 and BCLK1, running at 533 MHz. This is the trigger signal for the bit transactions, with the strobe signals as the qualifiers for the actual bit transfers.

Data Bus Characteristics (Refer to Figure)

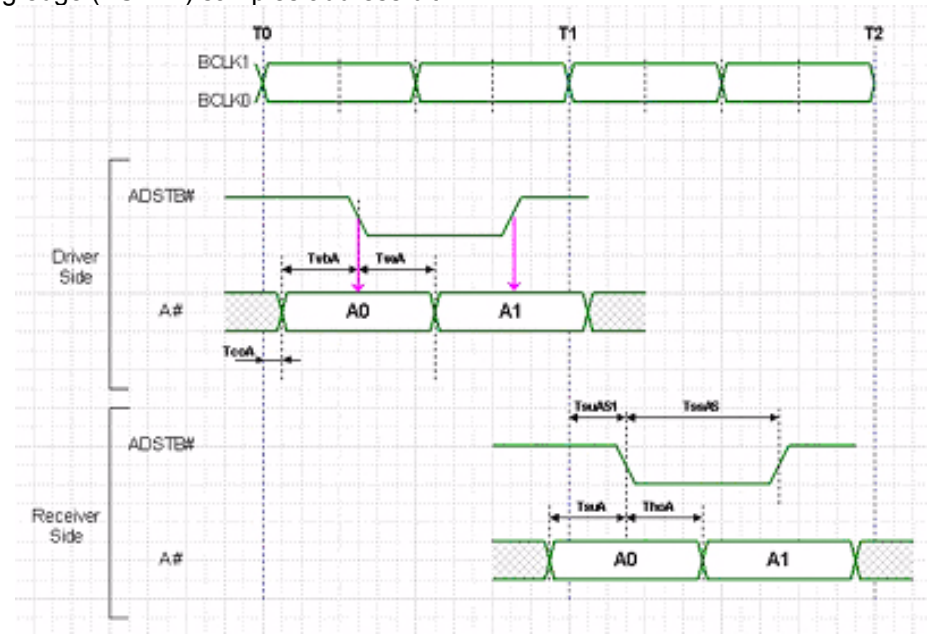
- During every period of BCLK, 4 bits are sent or there's an idle state: high voltage on both strobes and relevant data pins.
- Bit transfers occur in quad multiples of bits only (4, 8, 12…).
- Data Strobe positive (DSTBp#) samples bits 0 and 2 on consecutive falling edges
- Data Strobe negative (DSTBn#) samples bits 1 and 3 on consecutive falling edges



Data Bus Timing Diagram ©Intel Corporation

Address Bus Characteristics (Refer to Figure):

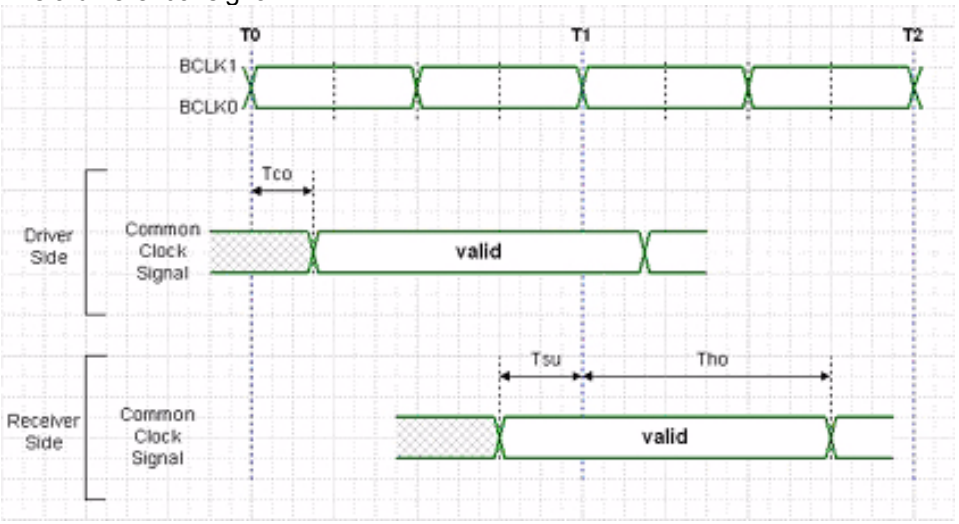
- During every period of BCLK, 2 bits are sent or there's an idle state (high voltage on both strobes and relevant data).
- Bit transfers occur in double multiples of bits only (2, 4, 6…).
- There's sampling of address bits on both falling and rising edges of ADSTB
- Falling edge (ASTB#) samples address bit 0
- Rising edge (ASTB#) samples address bit 1



Address Bus Timing Diagram ©Intel Corporation

Common Clock Bus Characteristics (Refer to Figure)

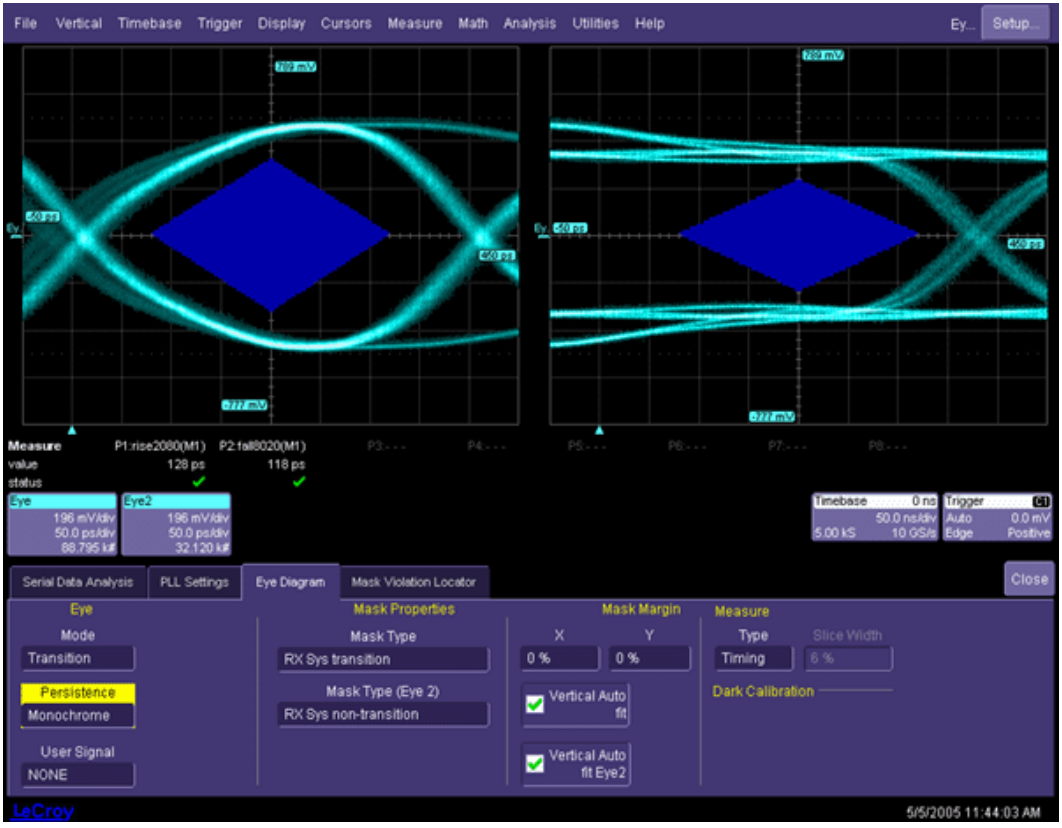
- During every period of BCLK, 1 bit is sent or there's an idle state meaning high voltage on the signal.
- BCLK is a differential signal.



Common Clock Bus Timing Diagram © Intel Corporation

Transition/Non-Transition Eye Diagram

A dual display eye diagram is formed with bits that are of non-changing polarity (non-transition) and changing polarity (transition). This display mode is useful for those serial data standards that utilize mask testing for both types of bit sequences (PCI Express and FB-DIMM Point to Point).



Transition Mode

For example, Section 4.3.3.1 of PCI Express Specification defines the requirements for Transition and Non-transition eye diagrams:

> "There are two eye diagrams that must be met for the Transmitter. Both eye diagrams must be aligned in time using the jitter median to locate the center of the eye diagram. The different eye diagrams will differ in voltage depending whether it is a transition bit or a de-emphasized bit. The exact reduced voltage level of the de-emphasized bit will always be relative to the transition bit."

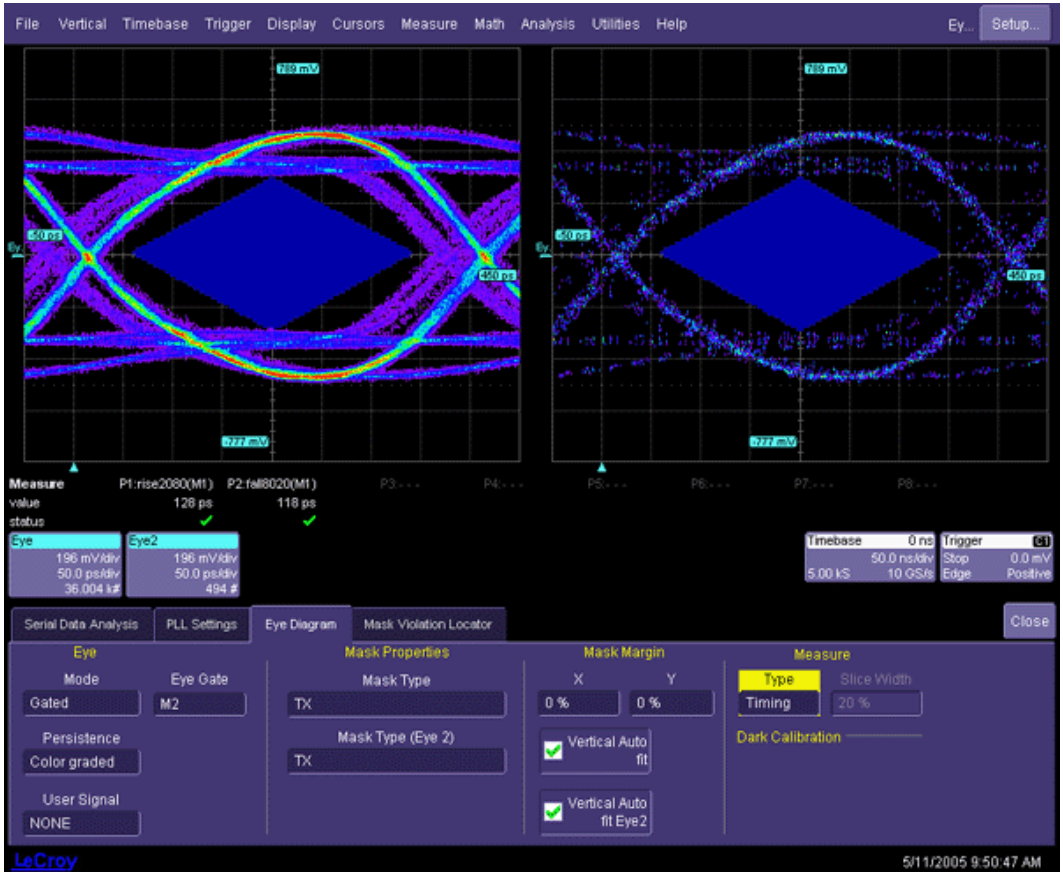
Gated (Qualified) Eye Diagram

The Gated Eye Diagram mode utilizes a separate signal (the “Gate” or qualifier) to create dual eye diagrams based on the polarity of the Gate.
Here’s an example of a gated eye diagram. Channel 1 (M1) contains the raw data and Channel 2 (M2) contains a Gate signal. While the Gate signal is high, the UIs will transfer to Eye display; When the Gate signal goes low, the UIs will transfer to Eye2 display



Data (M1) and Gate (M2) signal sources

The following is the Eye diagram resulting from applying the Gate signal. When the Gate is high, the bit data goes to one eye diagram. When the Gate is low, data goes to the other eye.



Gated Eye Diagram Example

SDA THEORY

The SDA operates by processing a long signal acquisition. The processes include clock recovery, eye pattern computation, jitter measurement, and bit error testing. All of these operations are performed on the same data record. The processes are described in detail in this section.

Clock recovery

An accurate reference clock is central to all of the measurements performed by the SDA. The recovered clock is defined by the locations of its crossing points in time. Starting with zero, the clock edges are computed at specific time intervals relative to each other. A 2.5 GHz clock, for example, will have edges separated in time by 400 ps.

The first step in creating a clock signal is the creation of a digital phase detector. This is simply a software component that measures the location in time at which the signal crosses a given threshold value. Given the maximum sampling rate available, 20 GHz, interpolation is necessary in most cases. Interpolation is automatically performed by the SDA. Interpolation is not performed on the entire waveform; rather, only the points surrounding the threshold crossing are interpolated for the measurement. A cubic interpolation is used, followed by a linear fit to the interpolated data, to find the crossing point. This is shown in Figure 1.

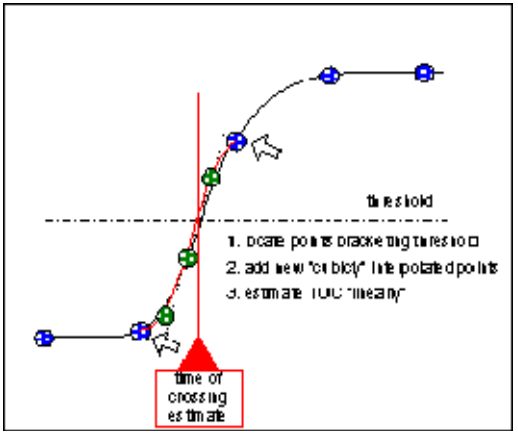


Figure 1. SDA Threshold Crossing Algorithm

Clock recovery implementation in the SDA is shown in Figure 2. This algorithm generates time values corresponding to a clock at the data rate. The computation follows variations in the data stream being tested through the use of a feedback control loop that corrects each period of the clock by adding a portion of the error between the recovered clock edge and the nearest data edge.

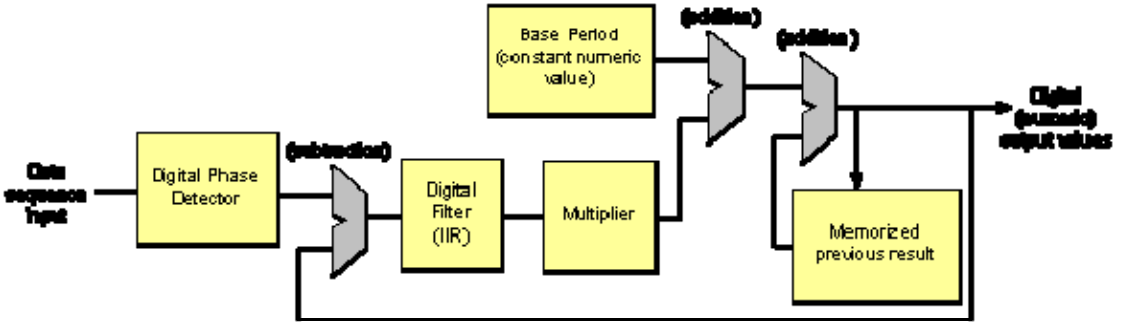


Figure 2. Clock Recovery Implementation

As shown in Figure 2, the initial output and the output of the digital phase detector are set to zero. The next time value output is equal to the nominal data rate. This value is fed back to the comparator on the far left which compares this time value to the measured time of the next data edge from the digital phase detector. The difference is the error between the data rate and the recovered clock. This difference is filtered and added to the initial base period to generate the corrected clock period. The filter controls the rate of this correction by scaling the amount of error that is fed back to the clock period computation. This filter is implemented in the SDA as a single-pole infinite impulse response (IIR) low-pass filter. The equation of this filter is:

$$y_k = \frac{1}{n} x_k + \left(1 - \frac{1}{n}\right) y_{k-1}$$

The value of y_k is the correction value for the k^{th} iteration of the computation and x_k is the error between the k^{th} data edge and the corresponding clock edge. Note that the current correction factor is equal to the weighted sum of the current error and all previous correction values. The multiplier value is set to one in the SDA, and the value of n is the PLL cutoff divisor that is set from the SDA main menu. The cutoff frequency is Fd/n where Fd is the data rate. This filter is related to its analog counterpart through a design process known as impulse invariance and is only valid for cutoff frequencies much less than the data rate. For this reason, the minimum PLL cutoff divisor setting is 20 in the SDA.

The factor *n* determines the number of previous values of the correction value *y* that is used in the computation of the current correction value. This is theoretically infinite; however, practically there is a limit to the number of past values included. One can define a "sliding window" equivalent to a number of UI (unit intervals) of the data signal for a given value of *n*. This is useful for measuring signals such as serial ATA and PCI-Express, where the specifications call for clock recovery over a finite window. The equivalent bandwidth of the sliding window is given by a sin(x)/x function. The first null of this function occurs at x = π or ½ the bit rate (the digital equivalent of the frequency of a signal at the sampling rate is 2π and the sampling rate for clock recovery is the data rate). This is scaled by the window size to be 2π / N where N is the window in UI. The 3 dB point of the sin(x)/x function is at 0.6π / N or 0.3Fd/N for a window length of *N*. This gives us a relationship between *N* and *n*:

Fd/n = 0.3Fd/N or n = N/0.3

For a sliding window size of 250, the equivalent value of *n* would be 833.

Eye Pattern

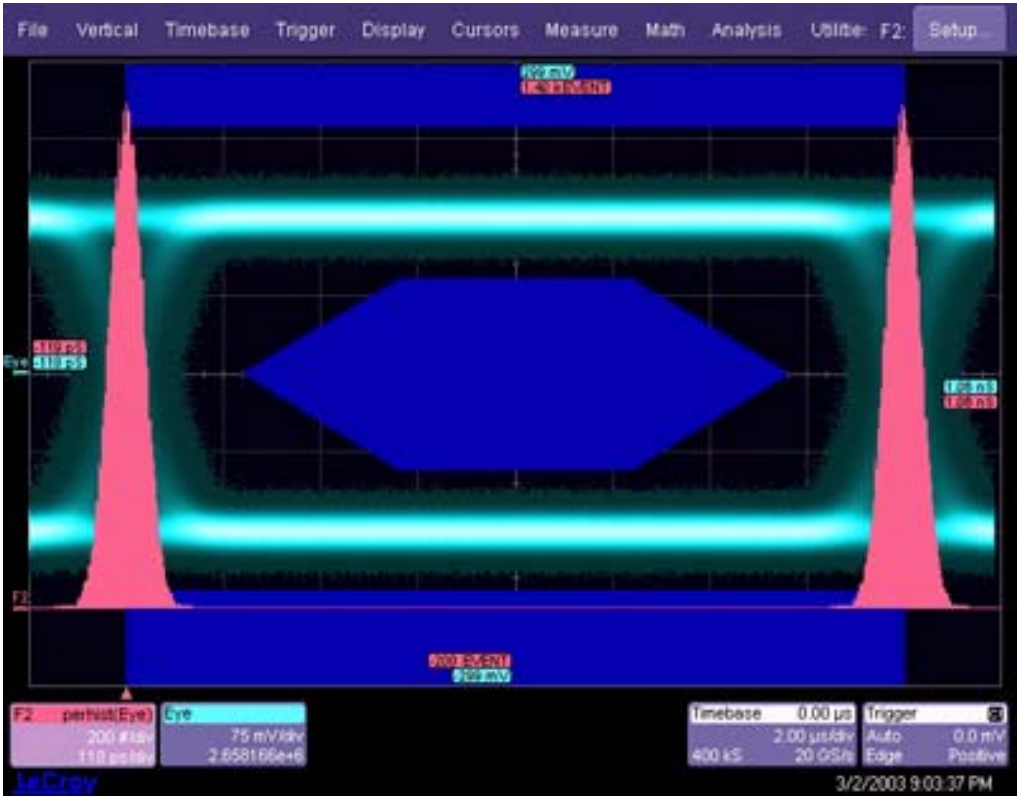


Figure 3. Histogram of Zero Crossing in Eye Pattern Showing Jitter Distribution

An eye diagram shows all values a digital signal takes on during a bit period. A bit period (or UI) is defined by the data clock, so some sort of data clock is needed to measure the eye pattern. The traditional method of generating an eye pattern involves acquiring data on an oscilloscope and using the data clock as a trigger. One or more samples are taken on each trigger. The samples are stored in a persistence map with the vertical dimension equal to the signal level, and the horizontal position equal to the sample position relative to the trigger (or data clock). As many data points are collected, the eye pattern fills in with multiple occurrences of time and amplitude values counted by incrementing counters in each x,y "bin." Timing jitter is indicated by the horizontal distribution of the points around the data crossings. The histogram of the bins around the crossing points provides the distribution of jitter amplitude.

A recovered clock is used if there is no access to a data clock. The recovered clock is normally a hardware PLL designed to operate at specific data rates and with a cutoff frequency of Fd/1667. One of the major drawbacks of a hardware clock recovery circuit is that jitter associated with the trigger circuit adds to the measured jitter by creating uncertainty in the horizontal positioning of the eye pattern samples.

The SDA measures eye patterns without using a trigger by using aforementioned the software clock recovery to divide the data record into segments along the time values of the clock. For the purpose of dividing the timeline into segments, the time resolution in the waveform record is infinite. The samples occur at fixed intervals of 50 ps/pt (for a 20 GS/s sampling rate). The samples are positioned relative to the recovered clock timing points and the segments delimited by the clock samples are overlaid by aligning the clock samples for each segment. A monochrome or color persistence display is used to show the distribution of the eye pattern data. The jitter added by the measurement system (in this case) is from the sampling clock which, for the SDA is very low: on the order of 1 ps rms.

Eye Violation Locator (ASDA Option)

The eye pattern is measured by overlaying segments of a continuous acquisition. Since the complete data record is available, the location of individual bits can be determined by comparing each bit interval in the original waveform to the selected mask. The mask is aligned horizontally along the mean bit interval, and vertically along the mean one and zero level in the case of a relative mask. Absolute masks exist for some standards and are defined in the vertical dimension by specific voltage values. Figure 4 below shows this alignment. When mask testing is turned on, the entire waveform is scanned bit-by-bit and compared to the mask. When a mask hit is detected, the bit number is stored and a table of bit values is generated. The table is numbered starting with the first bit in the waveform. This table can be used to index back to the original waveform to display the waveform of the failed bit.

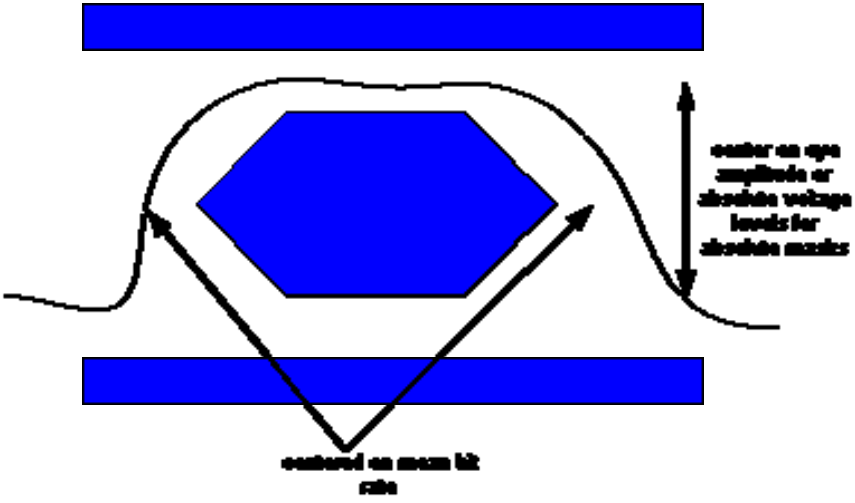


Figure 4. Eye Mask Alignment for Violation Locator

Eye Pattern Measurements

There are several important measurements that are made on eye patterns. These are specified as required tests for many standards. Eye measurements mainly deal with amplitude and timing, which are outlined as follows:

Eye Amplitude

Eye amplitude is a measure of the amplitude of the data signal. The measurement is made using the distribution of amplitude values in a region near the center of the eye (normally 20% of the distance between the zero crossing times). The simple mean of the distribution around the “0” level is subtracted from the mean of the distribution around the “1” level. This difference is expressed in units of the signal amplitude (normally voltage).

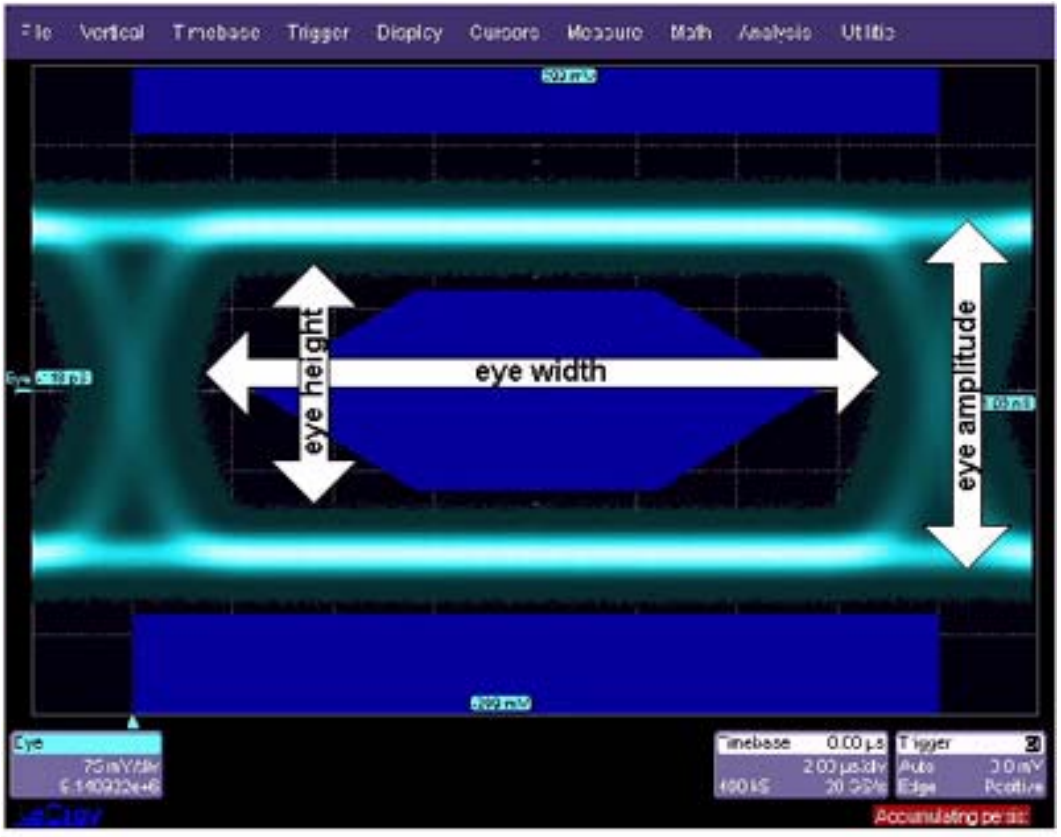


Figure 5. Eye Measurements

Eye Height

The eye height is a measure of the signal-to-noise ratio of a signal. The mean of the "0" level is subtracted from the mean of the "1" level as in the eye amplitude measurement. This number is modified by subtracting the standard deviation of both the "1" and "0" levels. The measurement basically gives an indication of the eye opening.

Eye Width

This measurement gives an indication of the total jitter in the signal. The time between the crossing points is computed by measuring the mean of the histograms at the two zero crossings in the signal. The standard deviation of each distribution is subtracted from the difference between these two means.

Extinction Ratio

This measurement, defined only for optical signals, is the ratio of the optical power when the laser is in the ON state to that of the laser in the OFF state. Laser transmitters are never fully shut off because a relatively long period of time is required to turn the laser back on, thus limiting the rate at which the laser can operate. The extinction ratio is the ratio of two power levels (one very near zero) and its accuracy is greatly affected by any offset in the input of the measurement system. Optical signals are measured using optical-to-electrical converters on the front end of the SDA. Any DC offset in the O/E must be removed prior to measurement of the extinction ratio. This procedure is known as dark calibration. The output of the O/E is measured with no signal attached (i.e., dark), and this value is subtracted from all subsequent measurements.

Eye Crossing

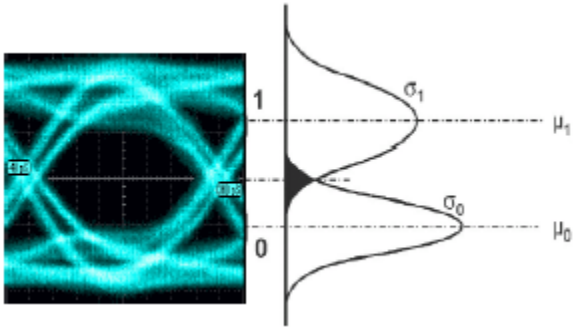
Eye crossing is the point at which the transitions from 0 to 1 and from 1 to 0 reach the same amplitude. This is the point on the eye diagram where the rising and falling edges intersect. The eye crossing is expressed as a percentage of the total eye amplitude. The eye crossing level is measured by finding the minimum histogram width of a slice taken across the eye diagram in the horizontal direction as the vertical displacement of this slice is varied.

Average Power

The average power is a measure of the mean vale of all levels that the data stream contains. It can be viewed as the mean of a histogram of a vertical slice through the waveform covering an entire bit interval. Unlike the eye amplitude measurement where we separate the ones and zeroes histograms, the average power is the mean of both histograms. Depending on the data coding that is used, the average power can be affected by the data pattern. A higher density of ones, for example, will result in a higher average power. Most coding schemes are designed to maintain an even ones density resulting in an average power that is 50% of the overall eye amplitude.

Q Factor or BER

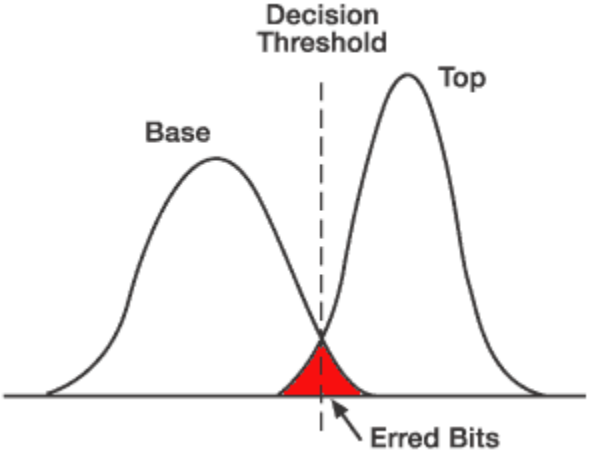
The Q factor is a measure of the overall signal-to-noise ratio of the data signal. It is computed by taking the eye amplitude (the difference between the mean values of the "1" and "0" levels) and dividing it by the sum of the noise values (standard deviations of the "1" and "0" levels). All of these measurements are taken in the center (usually 20%) of the eye.



$$Q-factor = \frac{\mu_1 - \mu_0}{\sigma_1 + \sigma_0}$$

eyeBER

EyeBER is the estimated bit error rate from an eye diagram. It is calculated using the intersection of the distribution of the one and zero levels. EyeBER differs from BER in that eyeBER is calculated from Q-factor and is, therefore, based on signal-to-noise ratio; BER, on the other hand, is based on jitter.



$$eyeBER = \frac{e^{-(Q^2/2)}}{Q\sqrt{2\pi}}$$

Jitter Measurement

Jitter is measured by the relative variation in the location in time of the transitions of the signal level across a specific level. For clock signals, the relative time between threshold crossings (rising-to-rising or falling-to-falling) is measured. Data signals, on the other hand, generally require the measurement of the relative positioning of the data signal to the sampling clock, which is related to setup and hold time. Because of its random nature, jitter is normally described in terms of its probability density function (PDF).

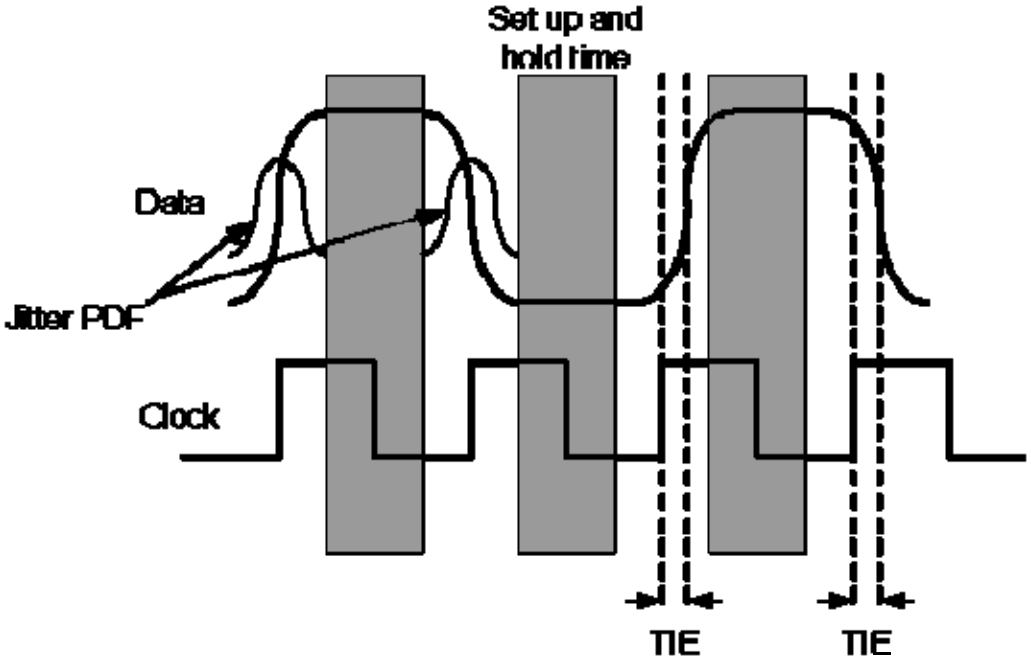


Figure 6. Setup and hold time requirement for error-free operation. Data transitions within the setup and hold time (gray area) will result in bit errors. Time interval error (TIE) is the time difference between clock and data edges. The PDF of TIE is a measure of the probability of an edge occurring during the setup and hold time.

The processes that make up jitter are complex and come from many different random and non-random (deterministic) sources. The PDF of the jitter is the convolution of all individual component PDF's. Measurements are able to estimate the jitter PDF but are not able to determine the distributions of the random and deterministic parts of the overall distribution. The lack of exact measurements for the jitter distributions of Rj and Dj has lead to the use of a simplified model for the total jitter. Equation 1 describes this model which was first presented in the Fibrechannel MJSQ document.

$$Tj(BER) \approx \psi(BER) * Rj + Dj \quad (1)$$

Equation 1 is a heuristic that describes total jitter as a function of bit error rate (BER) and is related to a distribution consisting of a Gaussian convolved with a pair of impulses as shown in Figure 7. The constants Rj and Dj represent all of the components of random and deterministic jitter. The Greek letter Psi is a function of BER and represents the total peak-to-peak jitter of a unit normal distribution (i.e., a Gaussian with zero mean and a standard deviation of 1 at the specified bit error rate. The process of determining Rj and Dj involves finding the “best fit” values that solve equation 1. There are many possible ways to fit Rj and Dj to (1) and since it is a simplification, no single set of solutions can completely describe the behavior of actual jitter completely. It is for this reason that the SDA uses two separate methods to measure Rj and Dj, effective and direct, and presents these to the user.

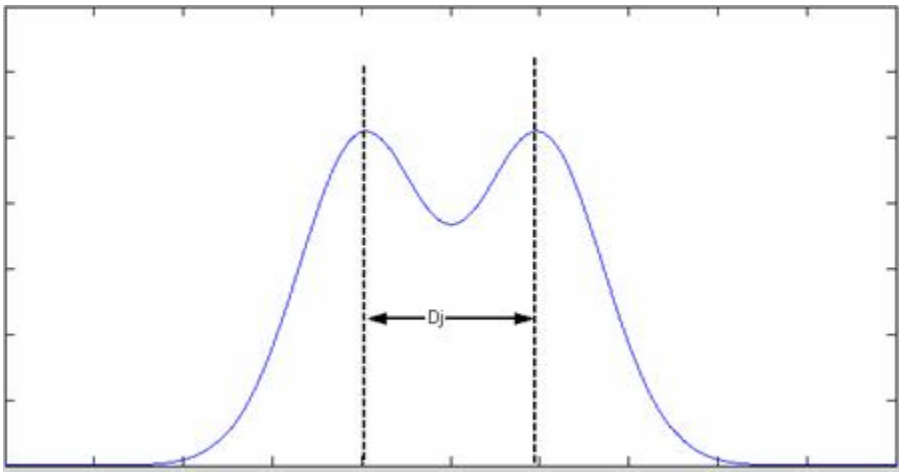


Figure 7. Jitter PDF model corresponding to the heuristic in equation 1. The random jitter is modeled by a Gaussian and the deterministic jitter by a pair of impulses separated by the value of the parameter Dj. The curve shown is the convolution of Rj and Dj.

Bit Error Rate and Jitter

Equation 1 shows that the total jitter is a function of bit error rate. This relationship is based on the effect that jitter has on the bit error rate of a system. The bit error rate is influenced by other parameters of the system, such as noise, so it is not correct to say that BER and jitter are equivalent. It is the contribution to the overall bit error rate caused by jitter that is shown in Equation 1. A bit error will occur when the data signal transitions from one state to another during the setup and hold time, as shown in Figure 6. Since jitter has a random component, the location in time of the transitions varies over a range of values. The longer the transitions are observed, the greater this range will be. Now, if we think of each transition in the data signal as the change in a bit value, then a transition at the wrong time (i.e., outside the setup and hold window) will lead to a bit error. The probability of this event is equivalent to the bit error rate contribution due to jitter. The total jitter gives a confidence interval for the jitter in that it will not exceed a certain value to a confidence of (1-BER). The term "bit error rate" is commonly used in this context to refer to the jitter confidence interval in many specifications.

EyeBER differs from BER in that eyeBER is calculated from Q-factor and is, therefore, based on signal-to-noise ratio; BER, on the other hand, is based on jitter.

Total Jitter

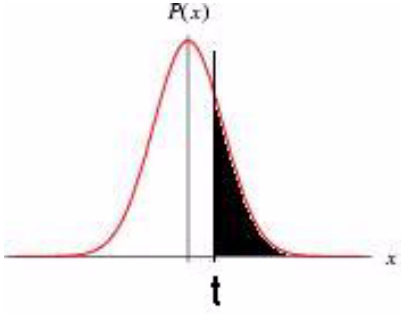


Figure 8. Probability of a data edge displacement greater than time t from the sampling clock. The mean value of the distribution is 0 which represents perfect alignment.

The total jitter is the peak-to-peak jitter in a clock or data signal within a specified confidence, equal to 1-BER. An example of a normally distributed jitter PDF is shown in Figure 8. In order to determine the total jitter from the PDF, the probability of the jitter exceeding a certain value *t* must be evaluated. This is done by integrating the PDF from a time *t* to $+\infty$ which will give the total probability of an edge occurring at or after this time. The probability can be computed for all values of *t* by integrating the PDF separately for $t > 0$ and $t < 0$. The resulting curve, shown in

Figure 9, gives the total probability of an edge being greater than *t* (or less than -*t*). The contribution to the system BER by jitter is given by the probability that an edge occurs at a time greater than *t* as we mentioned earlier. In order to guarantee a BER contribution from jitter below a certain value, the positive and negative values of *t* are chosen so that the probability of an edge at a time greater and less than these times is equal to the desired bit error rate. These jitter values can be measured by finding the intersection between a horizontal line at the bit error rate and the total jitter curve. The horizontal spacing between these two points is the total jitter.



Figure 9. Total jitter curve. The vertical values of this curve represent the probability of a data transition occurring at a time represented by the horizontal axis. The horizontal center of the plot is 0 ps. The two markers are placed at the vertical level corresponding to a bit error rate of 10 e-12 and the horizontal distance between these two points is the total jitter at this bit error rate.



Figure 10. The bathtub curve is constructed by rescaling the total jitter curve in Figure 9 to one unit interval, and centering the right side of the total jitter curve at 0 UI and the left side at 1 UI (the left and right sides of the bathtub curve).

A common way to view the total jitter is by plotting the bit error rate as a function of sampling position within a bit interval. This curve, commonly referred to as the "bathtub" curve is derived from the total jitter curve by scaling it to one bit interval (UI). The right half of the bathtub curve is taken from the left half of the total jitter curve and the left half is taken from the right half of the total jitter curve. The bathtub curve corresponding to the total jitter curve in Figure 9 is shown in Figure 10.

Extrapolating the PDF

Measuring the total jitter requires that the probability density function (PDF) of the jitter be known exactly. The SDA measures the jitter PDF by collecting a histogram of TIE measurements. This histogram approximates the PDF by counting the number of edges occurring within the time period delimited by each bin in the histogram. In order to accurately measure jitter contributions at very low bit error rates such as 1012, the histogram must contain measurements with populations that are below 1 in 1016 (one TIE measurement out of 1016 at a certain value). This number of data transitions would take approximately 38 days at 3 Gb/s. Measuring this number of edges is clearly impractical.

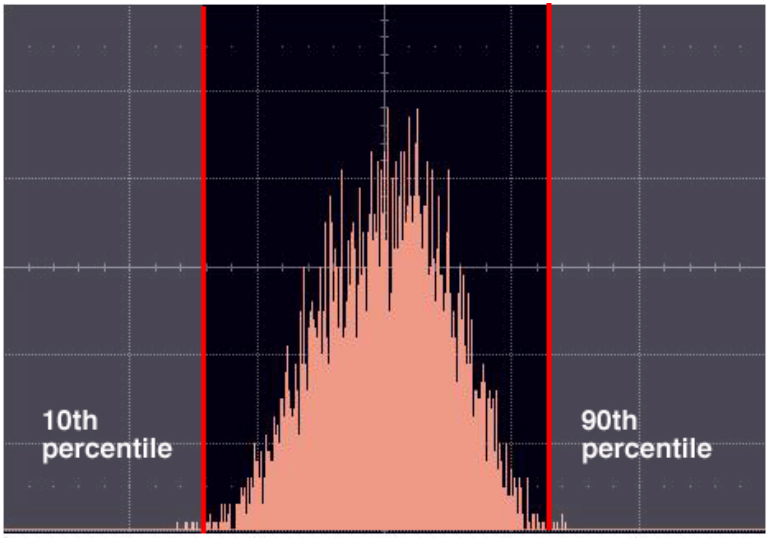


Figure 11: The measured histogram of TIE values is extrapolated by fitting curves to the bins below the 10th and above the 90th percentile. The log of the histogram is used to simplify this process to a quadratic fit.

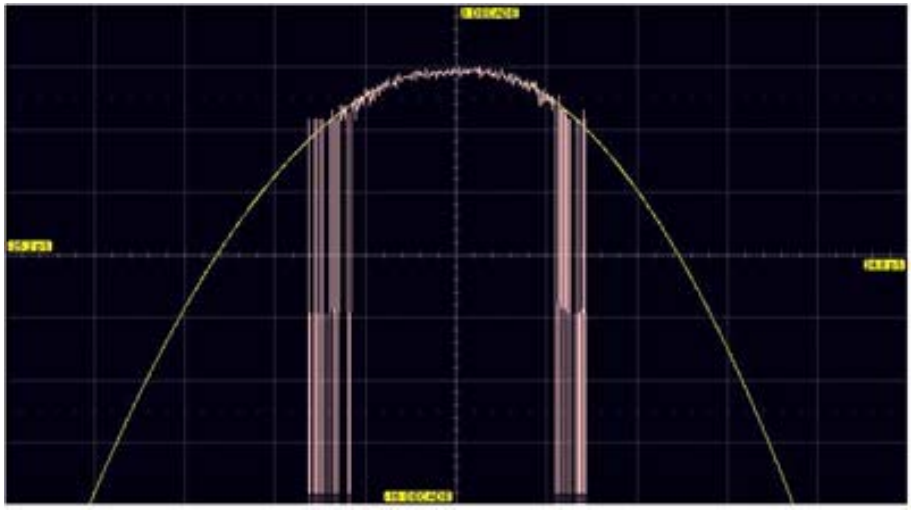


Figure 12: Logarithm of the measured TIE histogram superimposed on the extrapolated curve (in yellow). The extrapolation uses a quadratic curve fit to the histogram bins at the extremes.

A smaller data set is extrapolated in order to estimate the data for the larger sample size. The extrapolation of the measured histogram of TIE values uses the random nature of jitter at the extremes of the histogram to extrapolate the bins below the 10th percentile and above the 90th percentile. The central part of the distribution is dominated by deterministic jitter, while the extremes are entirely random. The bins in these ranges behave as a normal distribution as the jitter range is increased, that is, their populations fall as exp(-t2). Taking the logarithm of the histogram makes this relationship quadratic, so the extrapolation is simply a quadratic curve fit to the extremes of the log of the jitter histogram. The extrapolated histogram is used to compute the total jitter curve described above and is normalized so that the sum of the populations of all of the bins is one. The integrals described above are implemented by summing the extrapolated histogram bins.

Separating Rj and Dj – Two Methods

The total jitter curve is the basis for estimating the magnitude of Rj and Dj. Since the total jitter curve is derived directly from the signal under test, its value is the most accurate representation of the jitter for a given bit error rate. There are basically two ways of separating the random and deterministic jitter. The first method, which models the growth of total jitter as BER is decreased, leads to the effective jitter parameters Rje and Dje. These values are effective in the sense that they provide an equivalent total jitter model for low bit error rates. Starting with the total jitter curve, the growth in the total jitter as a function of decreasing BER is plotted. The curve described by equation 1 is fitted to the measured curve by selecting the Rj (slope) and Dj (intercept) to minimize the error in the fit.

The second method of fitting Rj and Dj to the measured data is based on direct measurements of the deterministic jitter. Random jitter is the difference between this value and the total jitter at the selected bit error rate measured from the total jitter curve. This, of course, exactly matches the measured total jitter at the selected bit error rate, but is a poor predictor of the jitter for bit error rates below this level. The motivation behind employing this method is to better represent the contribution of deterministic jitter in the overall jitter at the specified bit error rate.

Each method of measuring Rj and Dj results in different values for the standard deviation and spacing between the Gaussian curves in the distribution in Figure 7. The total jitter at the specified bit error rate, however, is the same for either distribution.

Effective Random and Deterministic Jitter

The effective jitter components Rje and Dje represent the best fit values for equation 1 to the behavior of the measured total jitter as the observation time is increased or, equivalently, the bit error rate is decreased. For a given bit error rate, the total jitter is measured from the width of the total jitter curve. The value of the total jitter as the bit error rate is decreased can be plotted as shown in Figure 13. The vertical axis of the plot on the left is the log of bit error rate. The Gaussian nature of the jitter at the extremes of the distribution results in a total jitter that grows approximately linearly with the log of BER, as shown in the upper curve in the plot on the right. The function N(BER)in equation 1 represents the width of a normal distribution with a variance of one at a given confidence level equal to 1-BER. The lower curve on the plot on the right shows the variation of N(BER) with the log of BER which is approximately linear. The values of Rje and Dje are chosen so that the lower curve lies on top of the upper one. From equation 1, it can be seen that Rje is a slope parameter while Dje adjusts the intercept point.

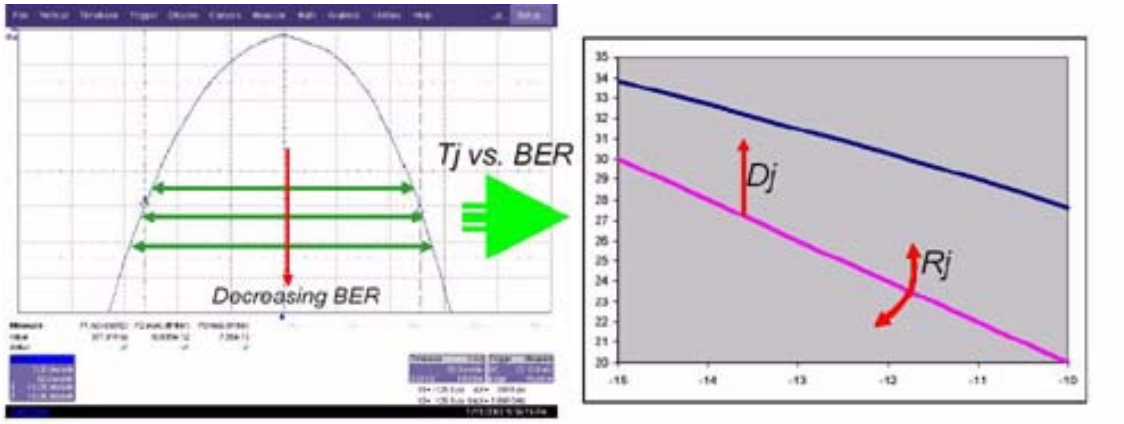


Figure 13. The change in total jitter with BER is represented by the values Tjn in the total jitter curve on the left. These values are plotted vs. bit error rate in the upper line in the chart on the right while the lower line shows the variation vs. BER for a pure Gaussian. Rje and Dje are chosen such that the curves line up.

The jitter computed using this method allows equation 1 to accurately model the jitter behavior of systems as a function of bit error rate. This model is especially useful when computing jitter margins in system applications.

Direct Measurement of Deterministic Jitter

Deterministic jitter can be completely characterized by measuring the threshold crossing times of the data signal over a finite time period. The two classes of deterministic jitter are periodic and data dependent.

Data dependent jitter is caused by system effects that are dependent on the data pattern. A common source of data dependent jitter is the frequency response of the channel through which the serial data signal is transmitted. In this case, data patterns with many transitions (such as a 101010… pattern) contain more high frequencies in their spectrum than patterns containing fewer transitions (11001100…, for example). The patterns with higher frequency content will be attenuated and phase shifted relative to the lower frequency patterns. In addition to data dependent jitter, the rise and fall times of the data bits can be different. The detection threshold in the receiver is normally set to the 50% amplitude (midway between the “1” and “0” levels) so unequal rise and fall times will generate jitter. This type of jitter is known as Duty Cycle Distortion (DCD).

When **DDj Calc. Method** is set to **ISI Plot**, the SDA uses a patent-pending method to measure both forms of data dependent jitter. The method uses the history of a number of bits in the waveform to determine their effect on the transition of a given bit. A user-selectable number of bits (from 3 to 10) is used in the measurement. The acquired waveform is processed in segments the size of the selected number of bits. For example, if 5 bits are selected, segments 5 UI long are examined. For each segment, the value of the 5 bits is determined and each group of 5 bits is averaged with segments of similar value. When the entire waveform is scanned, a set of up to 32 (for 5 bits) waveforms are created. The averaging process removes all random jitter, noise and periodic jitter from the segments. The waveform segments are overlaid by lining up the first transition of each of the segments and measuring the relative timing of the transitions to the last (5th in this example) bit.

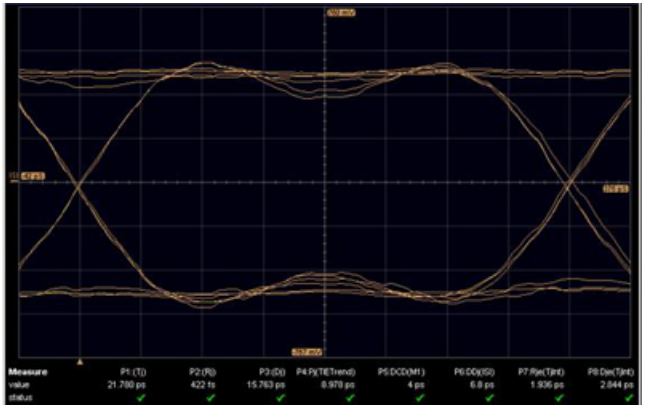


Figure 14. DDj measurement procedure. Averaged waveform segments for each pattern in the data stream are overlaid by lining up the first data transition. The curve above shows all of the transitions between the next-to-last bit and the last bit on the right side. The DDj is measured by examining the width of this crossing point at the selected threshold level.

Periodic jitter is measured by examining the spectrum of the trend of TIE values. The time interval error is measured for each edge in the data stream. Where no edges are present, as is the case for consecutive "1" or "0"' values, edges are inserted at the expected data transition times. These inserted edges do not add any additional jitter since they are placed at the ideal edge locations for the data rate. The trend of TIE is continuous and the spectrum can be computed. Periodic jitter is the complex sum of the spectral components excluding those associated with the repetition frequency of the data pattern and its harmonics.
The deterministic jitter is computed by adding up the periodic (Pj) and data dependent (DDj) components. The random jitter is computed using Equation 1 and subtracting the measured deterministic jitter from the total jitter at the selected bit error rate.

$$Rj = \frac{Tj(BER) - Dj}{\Psi(BER)} \quad (2)$$

Comparing Models

The plot in Figure 15 shows the bathtub curves for the measured values, as well as both estimates. Viewed in this way, it is clear why both the effective and direct measurements for Rj and Dj are used. Both estimates arrive at the same total jitter at the specified BER (10-12 in this case) but they give different values of Tj at other BER values. The effective jitter values give a very accurate prediction of total jitter for bit error rates below about 10-10, which is where they are fitted. The direct measurement underestimates the total jitter at error rates below the specified one and overestimates the jitter above this. Note that the effective parameters underestimate the jitter for high bit error rates. The three curves in Figure 15 show the resulting bathtub curves from the measured signal (blue line) and the two models: Hj(BER) for the direct Dj method (red line) and Hje(BER) for the effective jitter method (green line).

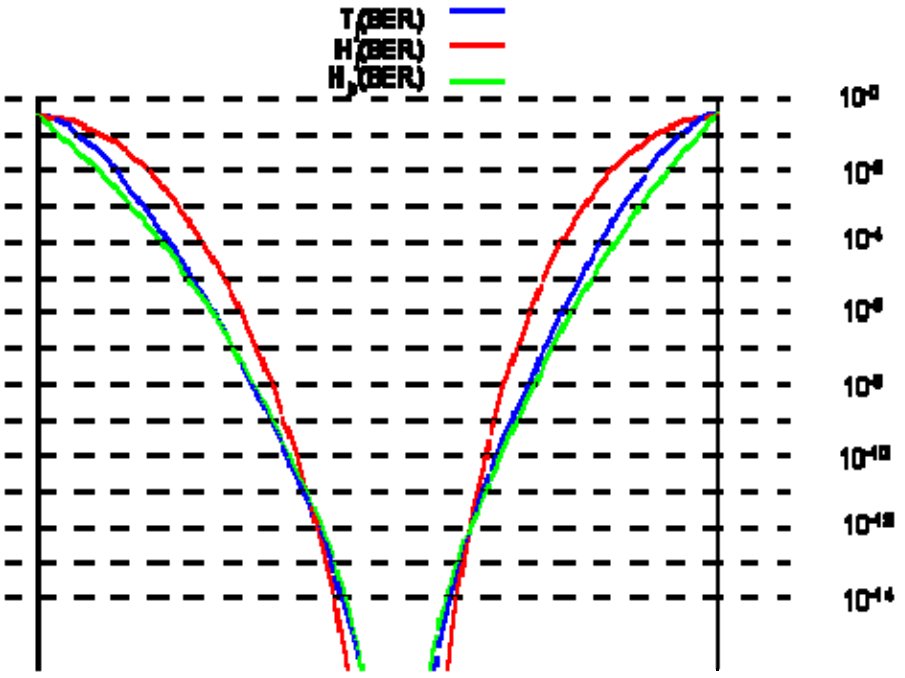


Figure 15. Measured jitter bathtub curve (blue curve) based on the extrapolation of the measured TIE histogram. The red and green curves are the estimated bathtub curves based on the direct Dj and effective measurements respectively. Note that the direct Dj method underestimates the total jitter below the BER at which it is computed.

Bit Error Rate

The SDA measures bit error rate directly on the captured bit stream by using the recovered clock to sample the waveform, and a user selectable threshold. The data are assumed to be NRZ so a high level is interpreted as a "1" and a low level is interpreted as a "0". The bit stream that is decoded in this process is compared, bit-by-bit with a user-defined known pattern. Since the instrument does not have any information as to which bit in the pattern it has received, a searching algorithm is used to shift the known pattern along the received data until a match is found. A match is determined when more than half of the bits are correct for a given shift of the known pattern. No match can be found if the bit error rate is over 50%, or if the wrong pattern is selected. In this case, the bit error rate will indicate 0.5 meaning that exactly ½ of the bits are in error which, of course, is the worst case.

Bit Error Map

A further level of debugging is available through the bit error map. This display is a view of the bit errors in the data stream relative to any framing that may be present in the signal. There are several options for framing that may be set. The general form of the data signal is shown here. The header portion is a fixed pattern that can be set to any pattern. The header must be one or more bytes if it is present. The software searches for the header and treats the bits between headers as a frame. Each frame is displayed as a line of pixels in an x-y map and each successive frame is displayed below the previous one in a raster fashion. Bit errors are computed only on the payload sections of the data stream.

Framing can also be defined by a specific number of bits without a header. An example of this is a pseudorandom bit sequence (PRBS) of a specific length, 127 bits for example. In this case, setting the frame size to 127 will display one repetition of this sequence per line of the error map. Bit errors are displayed in a lighter color, whereas non-errors are shown in dark blue. By displaying bit errors on a frame-by-frame basis, pattern dependent errors can be clearly seen as lightly colored vertical lines in the error map.

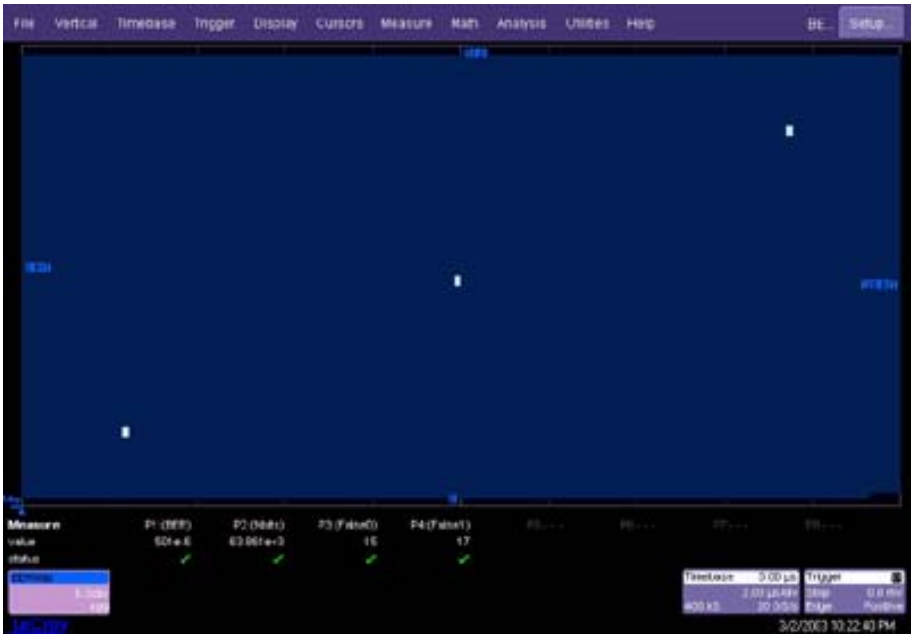


Figure 16. Bit Error Map for 127-Bit Pattern Containing Random Errors (White Squares)

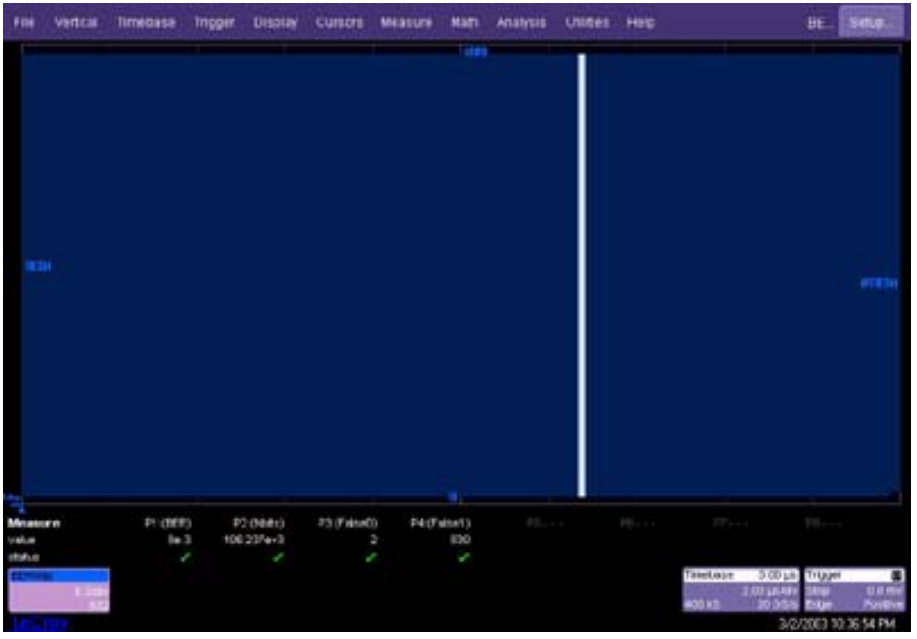


Figure 17. Bit Error Map for 127-Bit Pattern Containing Pattern Dependent Errors

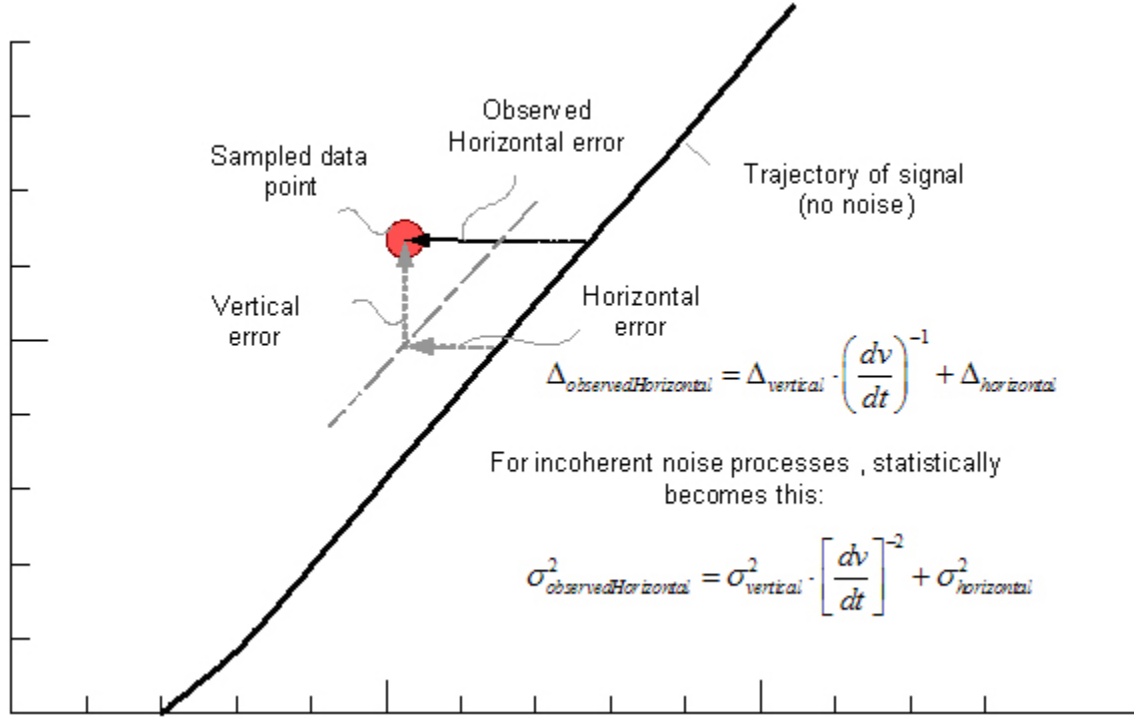
Vertical Noise Compensation

Introduction

Jitter is a measure of the timing error of data or clock signal transitions. The error is determined by measuring the time at which the signal crosses a given amplitude level and comparing it to the "expected" transition time. In a perfect world, this error would be entirely due to timing jitter but, as is often the case, things are not so simple. All signals contain amplitude noise and this noise can be interpreted as timing jitter in certain circumstances leading to higher overall jitter values. The noise on the signal under test is, and should be, included in any jitter measurement, since the data receiver would also interpret this as jitter. Any noise added by the test system should, however, not be included as this will overestimate the jitter. The SDA includes a feature that removes the system noise from the instrument, which uses the rms noise of the oscilloscope and probe, if present. This section explains the theory of operation of the noise compensation.

Signal Slew Rate and How Vertical Noise Converts to Jitter

In RF electronics, this process is known as AM to PM conversion. The basic principle is that the slew rate of the signal serves as a mechanism for vertical noise to be confused with horizontal (timing) noise, and vice versa. The following image illustrates:



It is easy to see how if the slew rate is very high (i.e., very low rise-time) that the effect of vertical noise is low and, conversely, that if the slew rate is very low, the vertical noise can dominate.

The Relative Impact of Noise on Jitter (Quadratic Addition of Noise)

The "quadrature addition" used to "add" the noise components is really only precise for Gaussian distributions of variations (even though this works well for many combinations of incoherent phenomena).
What's important in this application is that (at least for current instruments) the instrument's contribution to vertical noise is almost always Gaussian in nature. Furthermore, this can be confirmed by in-situ calibration procedures.

How the Measurement System Noise Is Subtracted from the Jitter Measurement

The timing noise can be very complex, exhibiting periodic jitter (*Pj*) and, for data streams, ISI induced data dependent jitter *DDj*. Other forms of bounded uncorrelated jitter (*BUj*) can contribute, as well, to make the jitter distribution shape quite non-Gaussian.
If, however (as is usually the case), the jitter analysis breaks down the distribution characteristic into *Rj* and *Dj*, the systematic effects are lumped into the *Dj* part. The *Rj* part of this jitter behaves as a Gaussian, and the vertical noise contribution (from the instrument) can be compensated as follows:

$$Rj_{compensated} = \sqrt{Rj^2_{observed} - \left[\frac{Rv_{instrument}}{dv/dt}\right]^2}$$

So, if the *dv/dt* can be well estimated by the instrument, and if the instrument's contribution to vertical noise is known (and incoherent with the signal, which is nearly always the case), this compensated Rj figure can be reported.
As such, the LeCroy SDA software has been equipped to perform this compensation, and applies it to the *Rj* figure and not the *Dj* figure; however, the corresponding effect on *Tj* is reported as well.
Per model noise figures are supplied as a built-in database for each SDA oscilloscope, and a procedure for updating the values for a specific oscilloscope channel and probe are provided.

Q-scale Theory

Introduction

Jitter is an important aspect of signal integrity for both optical and electrical serial data streams (and clocks). The SDA (serial data analysis) software is designed to measure the jitter and its components: random jitter (Rj), deterministic jitter (Dj), data dependent jitter (DDj) duty cycle distortion (DCD), and periodic jitter (Pj). The SDA uses a powerful method called "Normalized Q-scale Analysis" to estimate/measure the random and bounded, uncorrelated jitter components. The following section presents the technical background underlying this method.

Interpretation of TIE Histogram – the Distribution of Edge Transition Times vs. Ideal (Expected) Transition Times

For the purposes of this discussion (in connection with jitter measurement[10]) the entire subject surrounds the matter of interpreting the observed distribution of timing errors. This observed distribution is the histogram of Time Interval Error (TIE) values, obtained through analysis of either clock or NRZ data waveforms acquired by a digital recording instrument (such as a digital oscilloscope).

Relationship between Histograms and PDF

A histogram is nothing more than a form of data representation that expresses the frequency of occurrence of measurement values sorted or "binned" into adjacent, equal width contiguous intervals (or bins). When the timing errors (TIE) are collected as a histogram, the histogram serves as an approximation to the Probability Density Function (PDF) of this statistically based phenomenon (jitter). The PDF is (in theory) a smooth function determined by the underlying physics of the measured phenomenon (and of course what we actually observe includes the physics of the instrumentation as well).
The PDF is a continuous function, and reflects integrals of the probability (see http://en.wikipedia.org/wiki/Probability_density_function) over each interval of measurement value, *x*.

$$\Pr(a \le x < b) = \int_{x=a}^{x=b} PDF(x) \cdot dx$$

That is to say, the density of probability as a function of the measured quantity when integrated over a given region gives the probability that any measurement value will be within that region. The process of forming a histogram is based upon a pre-specified set of bin boundaries (meeting the above conditions of contiguity and equal width). A further constraint, which is usually unspoken, is that the histogram range must cover all possible observation values if it is to be useful.

$$h_i = h_{leftmost} + i \cdot h_{width}$$

The set of observations (of measured quantity *x*) are "binned" or counted for each range of histogram bin. The resulting histogram is a collection of populations (or counts) for each bin region.

$$H_i = Population_i = \#observations(h_i \le x < h_{i+1})$$

Now a measurement histogram (like the one we will analyze to estimate jitter) represents a single experiment, with some number of trials or individual TIE measured values. It is only an approximation of the PDF insofar as the true PDF plays its probabilistic role and so is reflected in the resulting observations.

[10] The "normalized Q-scale" method can be applied to other statistical studies and measurements; in particular, for examining the nature of vertical noise distributions.

$$\Pr(h_i \le x < h_{i+1}) = \int_{x=h_i}^{x=h_{i+1}} PDF(x) \cdot dx = \frac{\#observations(h_i \le x < h_{i+1})}{\#observations(-\infty \le x < \infty)} = \frac{H_i}{\sum H_i}$$

For this case (jitter) the observed distribution is the histogram of time interval error (TIE) values, obtained through analysis of either clock or NRZ data waveforms acquired by a digital recording instrument (like a digital oscilloscope).

Integrating the PDFs

The Cumulative Distribution Function (CDF or c.d.f.) expresses the probability that an observation will fall between $-\infty$ and the value *x*.

$$CDF(x) = \Pr(-\infty < x' < x) = \int_{x'=-\infty}^{x'=x} PDF(x') \cdot dx'$$

Of course this is a purely theoretical value. We can, however, calculate the Empirical Distribution Function (EDF or e.d.f.) by summing the histogram from the left extreme to some value *x*.

$$EDF(x = h_i) = \frac{\sum_{k=0}^{k=i-1} H_k}{\sum_{k=0}^{k=N-1} H_k} = \frac{1}{P_{total}} \sum_{k=0}^{k=i-1} H_k$$

Now, for our purposes and in keeping with tradition for discussions of jitter, we make a small change of representation. We are interested in variations in timing (jitter) before and after the mean timing value (or, to left or right of the mean). As such, and in order to view the timing errors in a symmetric fashion, we paste together two halves, the right-hand and the left-hand parts of the EDF, where the left-hand part is summed from the left (increasing bin index, starting with leftmost bin) and the right-hand part is summed from the rightmost bin with decreasing bin index, noting that these are approximations of the right-hand and left-hand parts of the “bathtub curve.”

$$EDF_{left}(x = h_i) = \frac{1}{P_{total}} \sum_{k=0}^{k=i-1} H_k$$

$$EDF_{right}(x = h_i) = \frac{1}{P_{total}} \sum_{k=N-1}^{k=i} H_k$$

In practice, these two functions are joined at the median of the histogram (the bin containing the median, or the bin for which 50% of the total population is in that bin or those with lower index, and consequently 50% of the total population also falls within that bin and those bins with higher index.

To the point of all of this, in telecommunications or data communications, the relative rate of bit errors is called BER[11].

Extrapolation of the Distribution Tails (Extremes)

Historical Note: Before this approach was taken by LeCroy, the extrapolation was applied to the histogram tails, whereas now the extrapolation is applied to the EDF; in fact, to each half of the EDF, as previously described.

While the EDF is well defined in the central region of the histogram, where events are populous, it is poorly defined at the extremes. This is of course the nature of the problem of analyzing jitter, since we are most interested in learning about the nature of those most rare (timing error) events. So we strive to obtain an estimation of how this extremal behavior is even beyond where we have real observed data. To this end we strive to fit the data at the extremes of the EDF to plausible mathematical forms (suggested by the underlying physics). One such mathematical form is the "error function" which is closely related to the CDF of a Gaussian distribution.

The Error Function erf(x), Inverse Error Function erf^{-1}(x) and Related Functions

$$CDF(x) = \frac{1 - erf(x)}{2}$$

Where the error function itself is

$$erf(x) = \frac{2}{\sqrt{\pi}} \int_0^x e^{-t^2} dt$$

And the complementary error function

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_x^\infty e^{-t^2} dt = 1 - erf(x)$$

But more importantly, the inverse error function *erf-1(x)* is the inverse function (neither reciprocal nor complementary). This function gives the displacement *x*, for which a given value is the error function of that displacement. (See http://mathworld.wolfram.com/InverseErf.html for more detail on the inverse error function.)

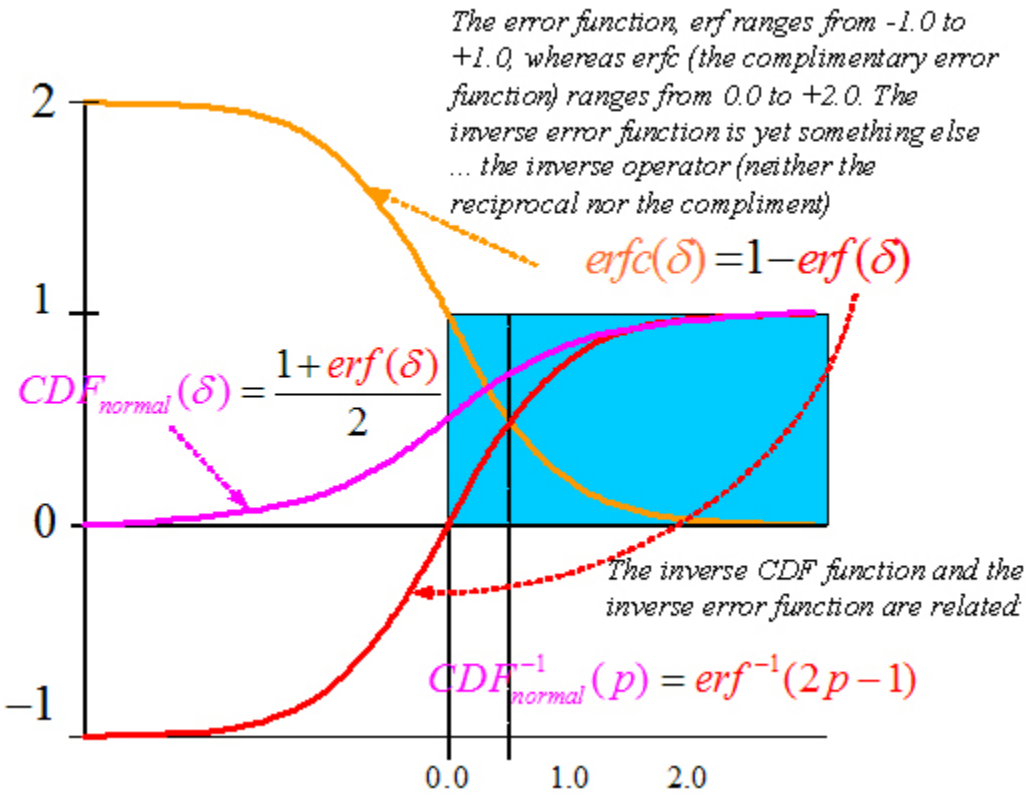
$$erf^{-1}(erf(x)) = x$$

And

$$erf(erf^{-1}(z)) = z \text{ for } -1 < z > 1$$

[11] Sometimes called bit error ratio, BER is the relative rate of bit errors compared to the bit rate, often expressed as a power of ten. For example: the BER is "ten to the minus 15th."

The following figure can help to put the general shapes of these functions in perspective. In particular, remember not to confuse the complementary error function *(erfc)* with the inverse error function (*erf*$^{-1}$).



The Relationship between the Inverse Error Function and Total Jitter

It is noteworthy that the inverse error function has no closed analytical form. It is also noteworthy that the heuristic jitter equation is linearly related to this inverse error function. Recall this equation:

$$Tj(BER) = Dj + \alpha(BER) \cdot Rj$$

Usually when this equation is presented, a qualifying remark might be "and the function *α(BER)* is the number of standard deviations for a Gaussian with sigma of 1, that corresponds to the specified bit error ratio (BER)." Now, it turns out that *α(BER)* is to within a constant factor, exactly the inverse error function, $erf^{-1}(1\text{-}BER)$. That is:

$$\alpha(BER) = 2 \cdot \left| CDF^{-1}_{Gaussian}\left(\frac{BER}{2\rho_{tx}}\right) \right|$$

$$\alpha(BER) = 2 \cdot \left| erf^{-1}\left(\frac{BER}{\rho_{tx}} - 1\right) \right|$$

Where in both cases we explicitly incorporate the transition density[12], since the purpose of the alpha factor is to calculate *Tj*, and transition density is required for this purpose (since jitter is only pertinent for bit errors that have transitions).

Application of Error Function to Measured Jitter CDF (on Q-scale)

There is a notion in science of "preferred coordinates." The notion goes a bit like this: "a physical-mathematical relationship can be most simply expressed when the coordinates of the problem are well chosen." An example is Kepler's Laws. When expressed in the usual Cartesian coordinate system, those laws are rather obscure; but, expressed in polar coordinates, they are quite simple.

The problem of analyzing the EDF in order to predict a CDF is best served by a coordinate transformation for the variable BER.

The subject of Q-Scale has been proposed and described by several sources. The desired transformation is to a new variable Q obtained from BER as follows:

$$Q(BER) = CDF^{-1}_{Gaussian}(BER/2)$$

The Cumulative Distribution Function, for a normal Gaussian is related to the error function in the following way.

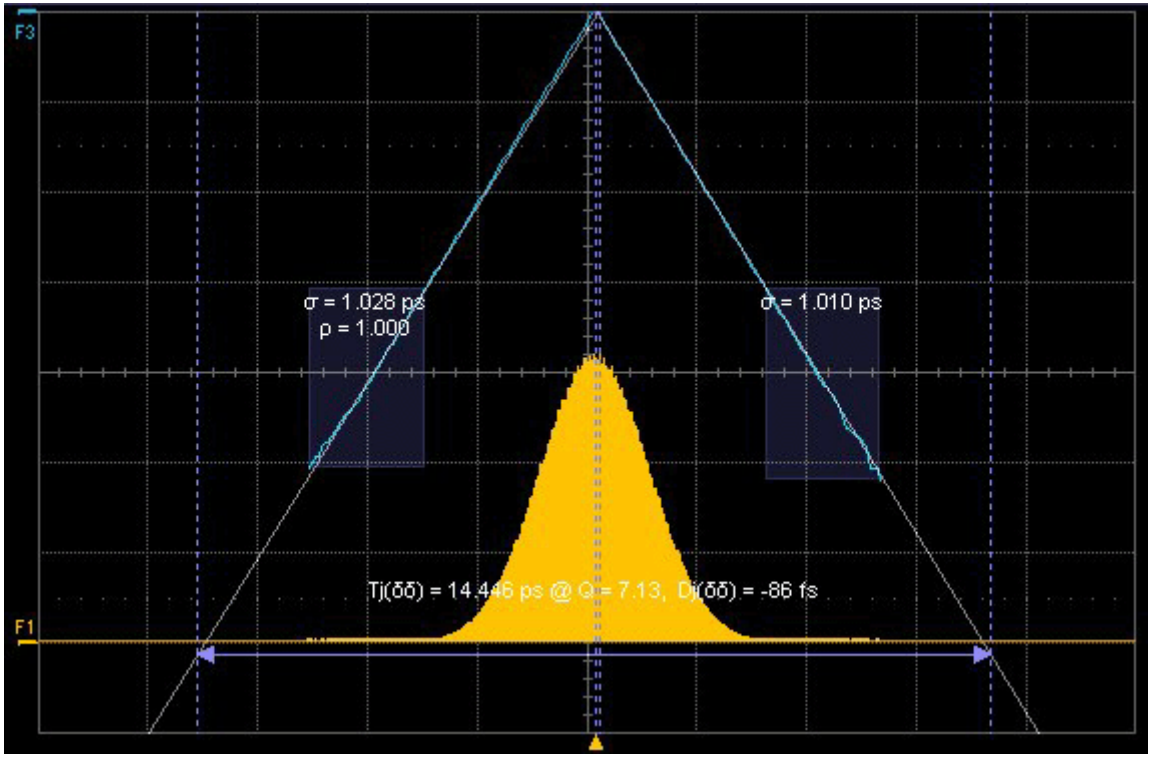
$$CDF(x) = \frac{1 - erf(x)}{2}$$

The *CDF* being the function that provides the probability for an event (for a system obeying a Gaussian probability density function) occurring to the left of the value *x*. Note also, by inspection:

$$CDF^{-1}(p) = erf^{-1}(2p - 1)$$

[12] This is the ratio of transitions between bit values to the total number of bits (so upper bounded by 1) but normally about 0.5 for standard test patterns and, in particular, for PRBS patterns. See white paper from LeCroy on this subject.

The reason this is such an interesting (and thus preferred) representation for the CDF and EDF is that on this scale the CDF of a Gaussian PDF is a straight line. When the CDF or EDF is of the modified symmetric form, then their graphs appear as the upper lines of a triangle. Below is a plot of an EDF (simulation) for a Gaussian PDF with a sigma of 1 picosecond.

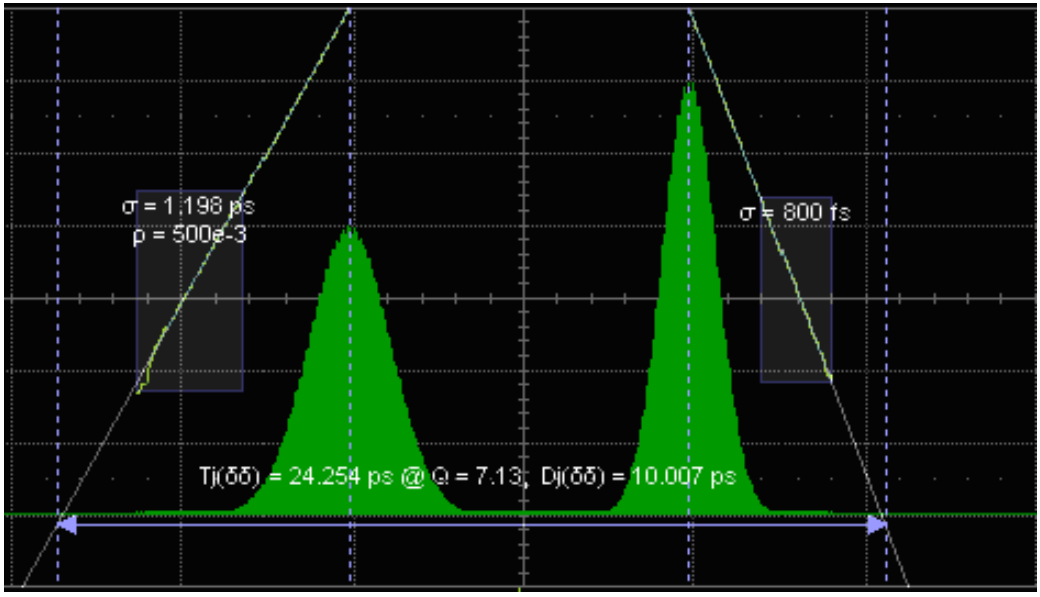


The other interesting attribute of this representation is that the slope of the lines give the sigma of the distribution. This is all common treatment so far. All is well for a single Gaussian distribution.

Now, enter the idea that this coordinate transformation should have a variable normalization factor. This normalization is such that when the area of an un-normalized Gaussian is ρ_{norm}, then that resulting CDF manifests as straight lines with slope revealing sigma, and intercept with Q=0, giving the mean of the Gaussian PDF. This is the “normalized Q-Scale” where:

$$Q_{norm}(BER) = CDF^{-1}\left(\frac{BER}{2\rho_{norm}}\right)$$

As an example, when two different Gaussian distributions are analyzed in this way, their EDFs appear as below. The linear fits on this coordinate scale yields the proper (simulated) values for their sigmas and means.



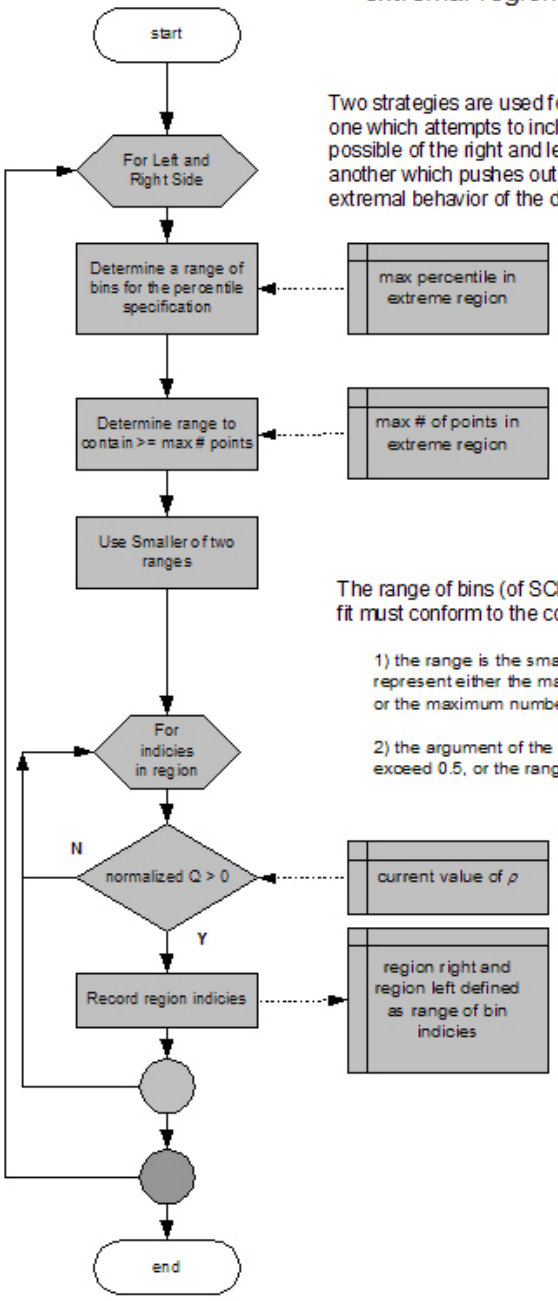
Automatic Renormalization of the Q-scale

This is the heart of the (patent pending) method for analysis of EDF, yielding best estimates of the underlying PDFs defining the measured system's behavior.
The procedure is straightforward: create the symmetric EDF for each side of the distribution.

The method can be described in steps as:

1. Forming a set of elements to be fitted, which are 3 values each, a vertical (Q-scale) value, a horizontal coordinate (for the case of time variable jitter, but the method is extensible to vertical noise analysis or, in fact, any variable), and an associated error. Statistically, the error of the input variable $\Sigma\rho_n$ is furnished as the square root of the total population of observations (the same sum) contributing to the estimate (of EDF in this case). Upper and lower values of BER are obtained for these variations, and an error inversely proportional to the error is assigned to the data point.
2. Varying the Q-Scale normalization factor over a range of plausible values, the "best" fit to linear behavior is determined.
3. Using this value to correctly scale the probability, an intercept (with $Q_{norm}(BER)=0$) is obtained for both the right-hand side and the left-hand side of the $CDF(Q_{norm}(BER))$. This value may be interpreted as the mean-value associated with the noise/jitter contributor with sigma equal to the reciprocal of the slope obtained from the best linear fit, and amplitude (area) equal to the best normalization factor itself.
4. Under the assumption of symmetric noise/jitter contributions a single ideal normalization may be obtained and used, however, ideally both extremes of the EDF must be treated independently, yielding strengths (ρ_{norm}) and sigmas (reciprocal slope) to characterize the extremal behavior of each side of the EDF distribution. In this case we show only an example of one side of the EDF.

Selection Method for points in extremal regions to be fitted

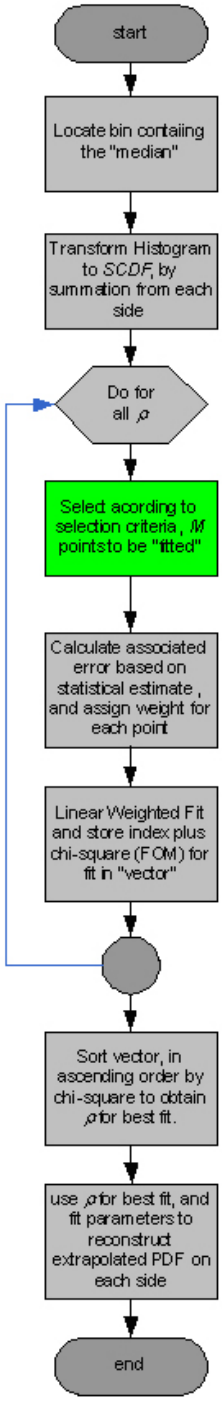


Two strategies are used for this procedure, one which attempts to include as much as possible of the right and left shoulders, and another which pushes out to find the extremal behavior of the distribution.

The range of bins (of SCDF) used for the fit must conform to the conditions:

1) the range is the smaller of the ranges required to represent either the maximum percentile of the PDF or the maximum number of points specified.

2) the argument of the inverse CDF must not exceed 0.5, or the range is reduced.



The smallest integer value k, for which $\frac{N_{total}}{2} \le \sum_{i=0}^{i=k} N_i$

$$SEDF_{m \le k}(N_i, N_{total}) = \left(\frac{1}{N_{total}} (\sum_{i=0}^{i=m} N_i) \right)$$ left *SEDF*

$$SEDF_{m > k}(N_i, N_{total}) = \left(\frac{1}{N_{total}} (\sum_{i=M-1}^{i=m} N_i) \right)$$ right *SEDF*

$$\rho_{norm} = 1.0, 0.95, 0.90, \ldots 0.05$$

$$Q_{norm}(BER) = CDF^{-1}_{Gaussian}(BER/(\rho_{norm} \cdot 2))$$

$$Q_i(N_i, N_{total}) = CDF^{-1}_{Gaussian}\left(\frac{1}{(\rho_{norm} \cdot 2N_{total})} (\sum_{i=0}^{i=k} N_i) \right)$$

$$if(N == 1)$$

$$w_i = 1/\Delta_i = 1/\left(2Q((N_i + \sqrt{N_i})/N_{total}\right)$$

$$elseif(N > 1)$$

$$w_i = 1/\Delta_i = 1/\left(\begin{array}{l} (Q((N_i + \sqrt{N_i})/N_{total}) - \\ (Q((N_i - \sqrt{N_i})/N_{total}) \end{array} \right)$$

$$x_i = x_0 + (i \cdot \Delta x)$$

Weighted Linear Fit based upon the minimization of chi-square wr t. the variable m and b (slope and intercept)

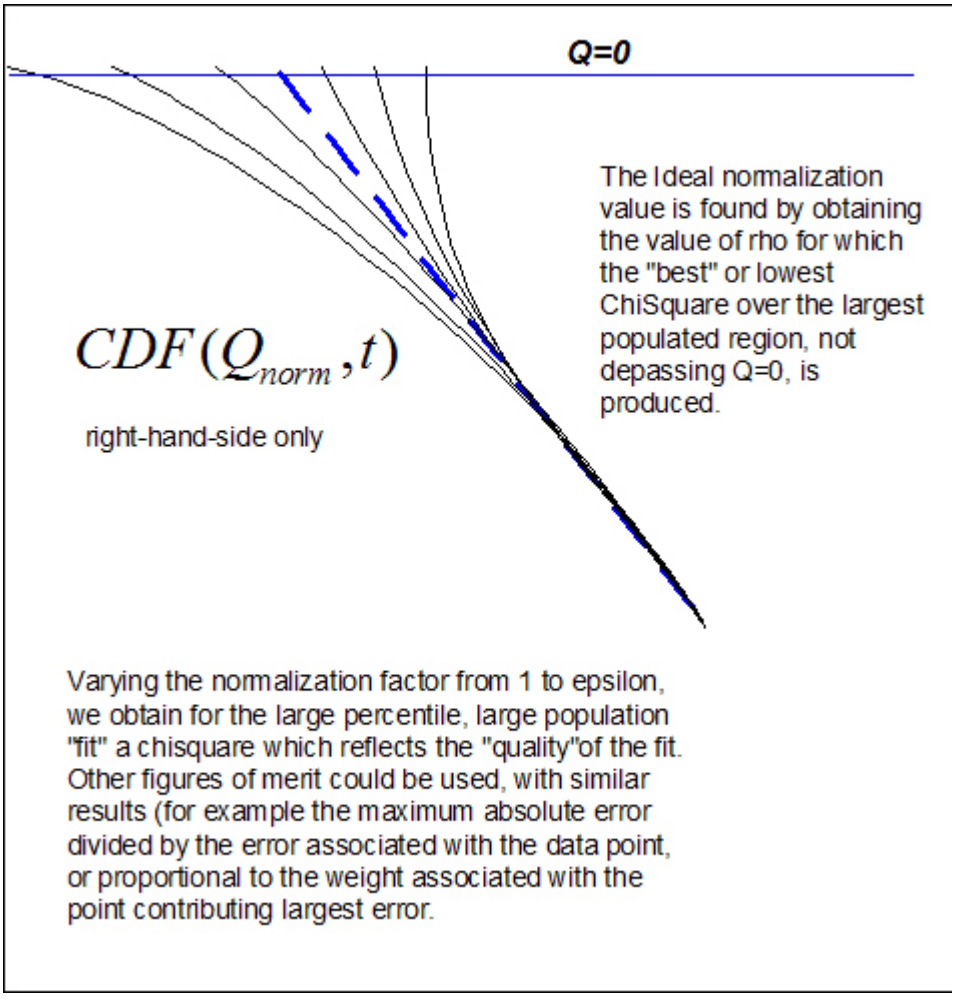
$$\chi^2 = \frac{\sum_{j=0}^{M-1} w_j^2 (Q_j - mx_j - b)^2}{\sum_{j=0}^{M-1} w_j^2}$$

Sort from lowest chi-square to highest, to determine the ideal value for ρ

$$\sigma = |1/m| = |1/slope| \qquad \mu_L = line_L(x @ y = 0) = -b/m$$

$$PDF'_i = \left(\frac{\rho_{norm}}{\sigma\sqrt{2\pi}} e^{-\frac{(x_0 - j\Delta x - \mu_L)^2}{2\sigma^2}} \right)$$

To fill in PDF bins to the left (induding region fit from left), and rather similarly fror the right hand side, using results of the righthand fit.



Obtaining Deterministic and Random (Gaussian) Components from the Normalized Q-scale Diagram

While there are detailed differences between this and other approaches for obtaining Rj and Dj, the most important figure we need to establish is total Jitter. Referring roughly back to the heuristic jitter equation, we rewrite this as:

$$Tj(BER) = Dj + 0.5\alpha(\frac{BER}{W_{left}}) \cdot Rj_{left} + 0.5\alpha(\frac{BER}{W_{right}}) \cdot Rj_{right}$$

Where Dj is the separation between the two means of the distributions.

If the distribution is governed by only two Gaussians of equal weight, this equation degenerates into

$$Tj(BER) = Dj + 0.5\alpha(2BER) \cdot Rj_{left} + 0.5\alpha(2BER) \cdot Rj_{right}$$

And if, in addition, the two Gaussians have the same sigma (or Rj), then

$$Tj(BER) = Dj + \alpha(2BER) \cdot Rj$$

This equation is disturbingly different from

$$Tj(BER) = Dj + \alpha(BER) \cdot Rj$$

This is because in the traditional Dual-Dirac discussion, we typically "forget" that the two Gaussians are only half-strength.
In general, to approximate the traditional Dj and Rj value, we are still using the separation of the "outermost means, and approximating Rj by the mean of the two sigmas.
We do, however, calculate Tj from the most precise estimation of the CDF from the renormalized Q-scale fit, using correct weights and sigmas to reconstruct a theoretical PDF, using the central region of the histogram "as is" and the extremes replaced by theoretical Gaussians of the correct strength, sigma, and mean to estimate each extreme region. This is then reconstructed into a CDF, so Tj(BER) becomes a simple calculation on that curve.

§ § §

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

(217) 352-9330 | sales@artisantg.com | artisantg.com

ARTISAN® TECHNOLOGY GROUP