

Agilent 86038B
Photonic Dispersion and Loss Analyzer User's
Guide



Agilent Technologies

Notices

© Agilent Technologies, Inc. 2001 - 2006

No part of this manual may be reproduced in any form or by any means (including electronic storage and retrieval or translation into a foreign language) without prior agreement and written consent from Agilent Technologies, Inc. as governed by United States and international copyright laws.

Agilent Technologies, Deutschland GmbH
Herrenberger Straße 130
71034 Böblingen, Germany.

Manual Part Number

86038-90B03

Edition

Second edition, July 2006

First edition, September 2005

Subject Matter

The material in this document is subject to change without notice.

Agilent Technologies *makes no warranty of any kind with regard to this printed material, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose.*

Agilent Technologies shall not be liable for errors contained herein or for incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Printing History

New editions are complete revisions of the guide reflecting alterations in the functionality of the instrument. Updates are occasionally

made to the guide between editions. The date on the title page changes when an updated guide is published. To find out the current revision of the guide, or to purchase an updated guide, contact your Agilent Technologies representative.

Warranty

This Agilent Technologies instrument product is warranted against defects in material and workmanship for a period of one year from date of shipment. During the warranty period, Agilent will, at its option, either repair or replace products that prove to be defective.

For warranty service or repair, this product must be returned to a service facility designated by Agilent. Buyer shall prepay shipping charges to Agilent and Agilent shall pay shipping charges to return the product to Buyer. However, Buyer shall pay all shipping charges, duties, and taxes for products returned to Agilent from another country.

Agilent warrants that its software and firmware designated by Agilent for use with an instrument will execute its programming instructions when properly installed on that instrument. Agilent does not warrant that the operation of the instrument, software, or firmware will be uninterrupted or error free.

Limitation of Warranty

The foregoing warranty shall not apply to defects resulting from improper or inadequate maintenance by Buyer, Buyer-supplied software or interfacing, unauthorized modification or misuse, operation outside of the environmental specifications for the product, or improper site preparation or maintenance.

No other warranty is expressed or implied.

Agilent Technologies specifically disclaims the implied warranties of Merchantability and Fitness for a Particular Purpose.

Exclusive Remedies

The remedies provided herein are Buyer's sole and exclusive remedies. Agilent Technologies shall not be liable for any direct, indirect, special, incidental, or consequential damages whether based on contract, tort, or any other legal theory.

Assistance

Product maintenance agreements and other customer assistance agreements are available for Agilent Technologies products. For any assistance contact your nearest Agilent Technologies Sales and Service Office.

Certification

Agilent Technologies Inc. certifies that this product met its published specifications at the time of shipment from the factory.

Agilent Technologies further certifies that its calibration measurements are traceable to the United States National Institute of Standards and Technology, NIST (formerly the United States National Bureau of Standards, NBS) to the extent allowed by the Institutes' calibration facility, and to the calibration facilities of other International Standards Organization members.

ISO 9001 Certification

Produced to ISO 9001 international quality system standard as part of our objective of continually increasing customer satisfaction through improved process control.

Agilent Technologies Sales and Service Offices

For more information about Agilent Technologies test and measurement products, applications, services, and for a current sales office listing, visit our web site:

<http://www.agilent.com/comms/lightwave>

You can also contact one of the following centers and ask for a test and measurement sales representative.

United States:	1 800 829 4444 1 800 829 4433 (FAX)
Canada:	1 877 894 4414 1 905 282 6495 (FAX)
Europe:	+31 20 547 2111 +31 20 547 2190 (FAX)
Japan:	120 421 345 120 421 678 (FAX)
Mexico	(52 55) 5081 9469 (52 55) 5081 9467 (FAX)
Australia:	800 629 485 800 142 134 (FAX)
Asia-Pacific:	+852 800 930 871 +852 2 506 9233(FAX)
Latin America	+55 11 4197 3600 +55 11 4197 3800 (FAX)

Contents

Getting Started	21
Introduction.....	22
Key benefits are:	22
General Safety Considerations.....	24
Safety Symbols.....	24
Instrument Markings.....	24
Laser Safety Considerations	26
Environmental Information.....	27
Setting Up the Photonic Dispersion and Loss Analyzer..	28
Agilent 86038B Front and Rear Panels.....	32
Performance Verification	35
Purpose.....	35
Things to Consider.....	35
Procedure	35
Navigating the Front Panel Functions.....	44
Defining Traces	45
Quick Setup Procedure	47
Prepare the Normalization	47
Quick Setup Process	47
Measurement Process	48
Measurement Concepts	49
Introduction.....	50
Understanding the Attributes Being Measured	51
Gain/Loss.....	51

Group Delay/Chromatic Dispersion/Optical Phase.....	52
Polarization Mode Dispersion/Differential Group Delay	54
Polarization Dependent Loss (PDL)	56
Fiber length.....	56
Description of the 86038B	57
Measurement Method (MPS)	59
Choosing Setup Parameters.....	61
Background	61
Relationship of setup parameters to measurement noise.....	62
Relationship of setup parameters to phase wrapping	63
Maximum RF Modulation Frequency (MHZ) ..	64
Relationship of setup parameters to aliasing ..	65
Relationship of setup parameters to resolution ..	65
Relationship of setup parameters to imaging ..	66
Relationship of setup parameters to DGD and PDL accuracy	67
Relationship of setup parameters to ripple amplitude accuracy	68
Control of resolution and noise	70
Selecting setup conditions for a new type of narrowband device	71
Suggested reading:	72
Multiple Port Configurations	73
Normalization and Measurement	74
Optical Modulator Bias Adjustment.....	74
Input Polarizer Adjustment	75
Phase Dynamic Accuracy.....	75

Length Normalization	76
Wavelength Accuracy Correction	76
Swept and Stepped Measurement Modes	77
Making Measurements	79
Introduction	80
Quick Setup Measurement Process	81
Set up the Normalization	81
Quick Setup Process	81
Measurement Process	81
Flexible Measurement Process	82
Prepare for Normalization Measurement	82
Setup and Normalization Process	82
Measurement Process	82
Setup Parameters and What They Do	83
Wavelength range	83
Wavelength increment	83
Modulation frequency	83
Samples/Point	83
IF Bandwidth	84
TLS Power	84
Choosing Setup Parameters for Your Device	85
Transmission Fiber (All Types)	85
Fiber Dispersion Compensating Module (DCM) .	86
Fiber Bragg Grating Dispersion Compensators	
(or other channel compensators)	87
Erbium Doped Fiber (Unpumped)	88
Erbium Doped Fiber (Pumped)	89
Narrowband Filter or DWDM Wavelength	

Routing Device	90
Broadband Component (Isolator, Circulator, Coupler, Attenuator and so on)	91
Function Reference	93
Introduction	94
File Menu	95
Recall	95
Save	95
Export	95
Print	96
Exit	96
View Menu	97
Minimize Application	97
System Menu	98
Laser Off	98
Laser On	98
Local	98
Preset	98
Preset Display Properties	99
Select TLS	99
Wavelength Meter	99
Status Log	99
Configuration	100
Real-Time Port Monitor	101
Optimize Optical Modulator Bias	102
Optimize Input Polarizer	103
Toggle mainframe local-lockout status	103
Zero external power meters	103
Setup Menu	104

Quick Setup	104
CD/PMD	104
Length	106
Ports	106
Measurement Range.....	106
Normalization Range.....	107
Sensitivity/Resolution	108
Laser Setup.....	112
Measure Menu.....	115
Begin Measurement	115
Begin Normalization	115
Clear.....	117
Cancel	117
Normalize	117
Close	117
Begin Length	117
Cancel Measurement	118
Display Menu	119
X Scale.....	119
Y Scale.....	120
Auto Scale	120
Marker Setup	121
Marker Search	122
Marker Functions.....	123
Define Trace	124
Smooth	125
Curve Fit	127
Navigating the Front Panel.....	129
Change View.....	129
Next Trace.....	129
Toolbar Functions.....	130

Zoom Functions	130
Measure Control Functions	132
Remote Operation	133
Overview	134
How to Use This Chapter	136
Recommended Control Sequence	138
How to Configure the PDLA for Networking .	138
How to Connect the PDLA to Your Network .	138
How to Change Network Settings	139
Install the Remote Client.	141
Program Structure: events vs. sequential	147
Front Panel Functions with No Remote Equivalent	149
Enumerations	150
Event Table	154
Troubleshooting	157
pdlaClient Front Panel Maps	158
Example 1: Simple Application for Controlling the PDLA . .	167
Starting a VB 6.0 PDLA Application	168
Install Software	168
Application Shell	168
Event Handling	170
Control Center Wavelength	174
Complete source code	175
Starting a VB.NET PDLA Application	178
Install Software	178
Application Shell	178

Event Handling	180
Take Sweep and Transfer Data	181
Control Center Wavelength	183
Complete Source Code	184
Starting a C# PDLA Application	187
Install Software	187
Application Shell	187
Event Handling	189
Take Sweep and Transfer Data	192
Control Center Wavelength	195
Complete Source Code	195
Example 2: Control from Excel	200
Complete Source Code	200
Example 3: Control from LabView	203
Example 4: Complete application - RepeatMeas.	204
PDLA Repeat Measurements	204
Triggering Measurements and Delay Between Measurements	208
Key variables:	208
Processing Measured Data	209
Updating PDLA Status.	210
Programming Commands	211
PDLA Remote Programming	212
PDLA Remote Control DLLs	212
.NET Interface Hierarchy	212
Server to Client Events	213
Connected (ByVal connected As Boolean)	213

EstimatedTestTime (ByVal sTestName As String, ByVal tsEstimate As Double)	213
NewDataUpdate (ByVal eLaserMode As enumLaserMode, ByVal eAquisitionMode As enumAcquisitionMode) ..	214
NewRealTimePortMonitorData (ByVal DUTPortInsertionLoss As Double, ByVal MonitorPortInsertionLoss As Double, ByVal DUTPortGroupDelay As Double, ByVal MonitorPortGroupDelay As Double).....	214
NewStatus (ByVal Message As String, ByVal MessageType As eEventLogType)	215
PresetProgress (ByVal progress As enumStatus)	215
Remoting (ByVal Remoting As Boolean)	215
SweepProgress (ByVal progress As enumStatus, ByVal eTestType As enumTestType, ByVal eTriggerMode As enumAcquisitionMode) ..	216
TLSSettlingProbability (ByVal dTimeRemaining As Double, ByVal eProbability As enumTLSSettlingProbability)	216
TLSSettingProgress (ByVal progress As enumStatus)	217
TriggerProgress (ByVal progress As enumStatus, ByVal eTriggerMode As enumAcquisitionMode) ..	217
Client to Server Commands	218
Base Commands	222
DispersionMode Property	222

NumPolStatesPMD Property	222
ExternalPowermeterFound Property	223
UseExternalPowermeter Property	223
DUTMinLossMaxGain: Property	224
GenerateMuellerAndPMDDData Property	224
Measure Method	225
Ports Property	225
MaxNumPorts Property	225
TriggerModeParams Property	226
AutoCancelAfterBadSweeps Property	226
Actions Commands	228
Abort Method	228
Measure Method	228
MeasureLength Method	228
Normalize Method	229
PrintScreen Method	229
Analysis - CurveFit Commands	231
Calculate Method	231
Catalog Property	232
Description Method	232
ShortName Method	233
Connectivity Commands	235
Connect Method	235
Disconnect Method	235
IsConnected Method	236
TerminateSession Method	236
Display Commands	237
AutoscaleDuringMeasurement Method	237
DefineTrace Method	237
GDOffsetMode Property	238
ToggleMode Method	238

ToggleRemoteIndicator Method	239
File Commands	240
ExportCSV Method	240
ExportMuellerPMD Method	240
ExportStatusLog Method	241
Recall Method	241
Save Method	242
Length Commands	243
IndexOfRefraction Property	243
Mode Property	243
SpecifiedLength Property	244
ValidIndexOfRefraction Method	244
ValidSpecifiedLength Method	245
MeasurementRange Commands	247
ValidStartStop Method	247
XStart Property	247
XStop Property	248
NormalizationRange Commands	249
ClearGDAndInsertionLossOffsets Method	249
ClearModBias Method	249
ClearPDA Method	250
ClearInputPolarizerAngle Method	250
PDAEnabled Property	250
UseCustomNormalization Property	251
ValidStartStop Method	251
XStart Property	252
XStop Property	252
Increment Property	253
ValidIncrement Method	253
Printing Commands	255
PrintScreen Method	255

Resolution Commands	256
RFModulationFrequency Property	256
ValidRFModulationFrequency Method.....	256
Increment Property	257
ValidIncrement Method	258
ValidTracePoints Method	258
Results Commands	260
FiberLength Method	260
Increment Property	260
XStart Method	261
XStop Method	261
XValueFromPointIndex Method	262
YData Method	262
Sensitivity Commands.....	264
DriftCorrection Property	264
IFBandwidth Property	264
SamplesPerPoint Property	265
ValidSamplesPerPoint Method	266
Status Commands	267
ErrorFlag Property	267
EstimatedTimeToTLSSettling Property.....	267
InstrumentState Property.....	268
LastError Property	268
PresetComplete Property	269
StatusLog Method.....	269
TLSSettling Property	270
TriggerComplete Property.....	270
ValidNormalization Property	271
System Commands	272
LaserState Method	272
PowerMeterZero Method.....	272

Preset Method	272
WavelengthRealtime Property.	273
.	273
SetPort Method	273
RealtimePortMonitor Method	274
System - Identity Commands	275
CompanyName Property.	275
ProductId Property.	275
ProductName Property	275
ProductVersion Property	276
RemotingVersion Property	276
TLS Commands.	278
ForceSettling Method	278
PowerSelect Property	278
MultiTlsSupport Method	279
SelectTls Method	279
TlsWvlRange Method	280
TlsWvlRangeString Method.	281
SelectedTLSEIdx Method	281
TlsIdn Method	282
TlsPn Method.	282
TlsSn Method.	283
HighPowerLimit Property	284
Troubleshooting and Upgrades	285
Introduction.	286
Troubleshooting Startup Problems	287
If the PDLA does not boot up to MS Windows XP 287	
If the PDLA Application does not automatically launch	287

If the PDLA cannot communicate with the TLS or MWM	288
Troubleshooting GPIB Communications Problems	289
1. Check the PDLA Status Log	289
2. Ensure that GPIB cables are configured correctly	289
3. Check the system GPIB Addresses	289
4. Check the tunable laser source and multi-wavelength meter firmware revision	290
5. Ensure that the GPIB setting in the PNA software is set to talker / listener	291
6. Ensure that NI GPIB driver is enabled, Agilent GPIB driver is disabled	291
7. Reboot the TLS	292
8. Reboot the PNA	292
9. Swap GPIB Cables	292
Troubleshooting Problems with Remote Programming .	293
Troubleshooting Sweep and Normalization Problems . .	294
If the Modulator Bias does not complete . . .	294
If the Wavelength Correction does not complete	295
If Mode Hops are reported in the Status Bar	295
Troubleshooting Incorrect Data	297
If measured loss is very high or parts of the curve are missing	297
If wavelengths are incorrect, or measurements are incorrect at certain wavelengths	297
If the GD and Gain/Loss data looks incorrect. . .	297
If the Group Delay measurement appears to be noisy	298
If the Group Delay measurement shows a	

sudden large offset	298
If GD ripple is being observed	298
Troubleshooting the TLS	299
Restart the TLS and realign the TLS module .	299
Test the High Power output on the TLS	299
Software Upgrade Procedure	301
Hard Drive Backup and Recovery	302
Hard Drive Replacement and Spare	303
Cleaning Information	304
Safety Precautions	304
Why is it important to clean optical devices?	305
What materials do I need for proper cleaning? . .	305
Preserving Connectors	310
Cleaning Instrument Housings	311
General Cleaning Procedure	311
How to clean connectors	311
How to clean optical head adapters	314
How to clean connector interfaces	314
How to clean bare fiber adapters	315
How to clean lenses and instruments with an optical glass plate	316
How to clean instruments with a fixed connector interface	317
How to clean instruments with a physical contact interface	317
How to clean instruments with a recessed lens interface	319
How to clean optical devices which are sensitive to mechanical stress and pressure	320
How to clean metal filters or attenuator gratings	

Additional Cleaning Information	322
How to clean bare fiber ends	322
How to clean large area lenses and mirrors	322
Other Cleaning Hints	325
Returning the Instrument for Service	326
Specifications and Regulatory Information	327
Introduction	328
Definition of Terms	329
Angled Connector Specifications	338
Specifications	339
General Characteristics	341
Conditions	342
Regulatory Information	343
Declaration of Conformity	344
Index	345



1

Getting Started

Introduction.....	22
General Safety Considerations.....	24
Setting Up the Photonic Dispersion and Loss Analyzer..	28
Agilent 86038B Front and Rear Panels	32
Performance Verification	35
Navigating the Front Panel Functions.....	44
Defining Traces	45
Quick Setup Procedure	47

Introduction

The 86038B Photonic Dispersion and Loss Analyzer (PDLA) is designed for complete optical characterization of optical fiber and components with optimized speed and accuracy. The dispersion measurements are made with the industry-standard modulation phase shift method and polarization-dependent parameters are determined by Mueller Matrix analysis.

Key benefits are:

- Simultaneous measurement of spectral CD, GD, PMD, DGD, PDL, and IL with a single connection and single setup to reduce measurement time and test uncertainty
- Fast swept-wavelength measurement of all parameters
- Selectable 6-polarization-state measurements for optimal Mueller Matrix analysis
- Wide available wavelength range with options covering 1260-1640 nm and up to 200nm sweep range with a single tunable laser
- Specified accuracy and repeatability over wide wavelength range
- Analysis for 2nd-order PMD, GD ripple, optical phase, zero-dispersion wavelength and CD slope
- Extremely accurate length measurement for fiber characterization
- Option available for enhanced IL and PDL accuracy
- Optional multiport measurement with integrated switching
- Drift correction provides excellent stability and accuracy when the environmental conditions of the room or the test device are gradually changing

NOTE

To get the most performance out of your PDLA, it is recommended that you read [Chapter 2, “Measurement Concepts”](#), [Chapter 3, “Making Measurements”](#), and [Chapter 4, “Function Reference”](#). These three chapters cover the linkages between the different measurements that can be made, specific devices being measured, and the setup parameters used to make the measurements.



Proceed to “Setting Up the Photonic Dispersion and Loss Analyzer” on page 28 for complete installation instructions.

General Safety Considerations

This product has been designed and tested in accordance with the standards listed on the Manufacturer's Declaration of Conformity, and has been supplied in a safe condition. The documentation contains information and warnings that must be followed by the user to ensure safe operation and to maintain the product in a safe condition.

Safety Symbols

CAUTION

The *caution* sign denotes a hazard. It calls attention to a procedure which, if not correctly performed or adhered to, could result in damage to or destruction of the product. Do not proceed beyond a caution sign until the indicated conditions are fully understood and met.

WARNING

The *warning* sign denotes a hazard. It calls attention to a procedure which, if not correctly performed or adhered to, could result in injury or loss of life. Do not proceed beyond a warning sign until the indicated conditions are fully understood and met.

Instrument Markings



The instruction manual symbol. The product is marked with this warning symbol when it is necessary for the user to refer to the instructions in the manual.



The laser radiation symbol. This warning symbol is marked on products which have a laser output.



The AC symbol is used to indicate the required nature of the line module input power.



The ON symbols are used to mark the positions of the instrument power line switch.




The OFF symbols are used to mark the positions of the instrument power line switch.



The CE mark is a registered trademark of the European Community.

 The CSA mark is a registered trademark of the Canadian Standards Association.

 The C-Tick mark is a registered trademark of the Australian Spectrum Management Agency.

ISM1-A This text denotes the instrument is an Industrial Scientific and Medical Group 1 Class A product.

WARNING

If this product is not used as specified, the protection provided by the equipment could be impaired. This product must be used in a normal condition (in which all means for protection are intact) only.

WARNING

No operator serviceable parts inside. Refer servicing to qualified service personnel. To prevent electrical shock do not remove covers.

WARNING

This is a Safety Class 1 Product (provided with protective earth). The mains plug shall only be inserted in a socket outlet provided with a protective earth contact. Any interruption of the protective conductor inside or outside of the instrument is likely to make the instrument dangerous. Intentional interruption is prohibited.

WARNING

To prevent electrical shock, disconnect the instrument from mains before cleaning. Use a dry cloth or one slightly dampened with water to clean the external case parts. Do not attempt to clean internally.

CAUTION

This product complies with overvoltage Category II and Pollution Degree 2.

CAUTION

VENTILATION REQUIREMENTS: When installing the product in a cabinet, the convection into and out of the product must not be restricted. The ambient temperature (outside the cabinet) must be less than the maximum operating temperature of the product by 4 °C for every 100 watts dissipated in the cabinet. If the total power dissipated in the cabinet is greater than 800 watts, then forced convection must be used.

CAUTION

Install the instrument so that the detachable power cord is readily identifiable and is easily reached by the operator. The detachable power cord is the instrument disconnecting device. It disconnects the mains circuit from the mains supply before other parts of the instrument. The front panel switch is only a standby switch and is not a LINE switch. Alternatively, an externally installed switch or circuit breaker (which is readily identifiable and is easily reached by the operator) may be used as a disconnecting device.

CAUTION

Always use the three-prong AC power cord supplied with this instrument. Failure to ensure adequate earth grounding by not using this cord may cause instrument damage.

CAUTION

This instrument has autoranging line voltage input. Be sure the supply voltage is within the specified range.

CAUTION

Use of controls or adjustment or performance of procedures other than those specified herein may result in hazardous radiation exposure.

Laser Safety Considerations

NOTE

Refer to the *Tunable Laser Modules User's Guide* for complete laser safety information.

WARNING

Do NOT, under any circumstances, look into the optical output or any fiber/device attached to the TLS output while the laser is in operation.

This system should be serviced only by authorized personnel. Do not enable the laser unless fiber or an equivalent device is attached to the optical output connector.

CAUTION

Use of controls or adjustments or performance of procedures other than those specified herein can result in hazardous radiation exposure.

Environmental Information



This product complies with the WEEE Directive (2002/96/EC) marking requirements. The affixed label indicates that you must not discard this electrical/ electronic product in domestic household waste.

Product Category: With reference to the equipment types in the WEEE Directive Annex I, this product is classed as a "Monitoring and Control instrumentation" product.

Do not dispose in domestic household waste.

To return unwanted products, contact your local Agilent office, or see

www.agilent.com/environment/product/ for more information.

Setting Up the Photonic Dispersion and Loss Analyzer

- 1 Unpack your shipment.
 - Inspect the shipping containers for damage.
 - Inspect the instruments.
 - Verify that you received the options and accessories that you ordered.

Keep the shipping containers and cushioning material until you have inspected the contents of the shipment for completeness and have checked the PDLA mechanically and electrically.

If anything is missing or defective, contact your nearest Agilent Technologies sales office. If the shipment was damaged, contact the carrier, then contact the nearest Agilent Technologies sales office.

Included with the 86038B are two patchcords of about 2 m length. The MXF Reference Fiber is a special mixed-fiber device and should be reserved for the verification procedure.

The other reference patchcord is standard single-mode fiber and should be used for normalization measurements.

CAUTION

Fiber-Optic connectors are easily damaged when connected to dirty or damaged cables and accessories. When you use improper cleaning and handling techniques, you risk expensive instrument repairs, damaged cables, and compromised measurements. Before you connect any fiber-optic cable to the PDLA, refer to [“Cleaning Information” on page 304](#).

CAUTION

If the instrument has been shipped with Option #400, for enhanced loss accuracy, then special protected fibers have been included with labels for connections to the power meters (PM), and monitor ports. These fibers can be easily damaged if connected incorrectly. Please read Step 10 before attaching these fibers.

- 2 Install the Tunable Laser Source module into the 8164B mainframe by sliding the module in from the rear of the mainframe and tightening the four thumb screws on the rear panel.
 - If more than one 8164B is included, a TLS should be installed in each one.

- If the instrument includes more than one TLS, but will use only one 8164B, choose the TLS to be installed in the 8164B for the first operation.

CAUTION

The network analyzer and optical test set are assembled as one analyzer unit. The combined weight is 38 kg (84 lbs). The lifting of this instrument requires two persons using proper lifting techniques.

- 3 Place the analyzer unit on top of the TLS mainframe.
 - If more than one 8164B is included, identify the "master" mainframe for this position. This 8164B will be used for installing switch and power meter options. The optional additional mainframes can be installed under or beside the master mainframe.
- 4 On the rear panel, connect the cables as shown in Figure 2a.
 - If more than one 8164B, the additional trigger cables should be attached in daisy-chain configuration as shown in Figure 2b. Each mainframe should also be connected to the GPIB, as shown. Note that with this trigger configuration, all 8164B units should be switched on for correct trigger behavior.
- 5 Connect the mouse and keyboard via USB ports at the back and front of the analyzer unit.
- 6 On the front panel, connect the 40 cm black protected patchcord (labeled "TLS Fiber") between TLS Output 2 and the TLS Input of the analyzer as shown in Figure 1. This patchcord is polarization maintaining fiber (PMF) and should not be interchanged with standard fibers. The connectors make angled contact.
 - Standard operation is provided by using the high-power Output 2 of the TLS, which includes variable attenuator functionality. However for special applications, the Output 1 can be used at any time to obtain a signal with very low spontaneous emission (broadband background light).
 - If a special PMF-switch for multi-TLS automation is included, this module should be inserted in Slot 4 of the master 8164B. Then the fiber from the TLS in this mainframe should be connected to Port 1 of the PMF-switch and the additional "TLS Fiber" included with the switch should be connected between the Port A (common) of the switch and the TLS Input of the analyzer. Additional tunable lasers can be similarly connected with PMF patchcords to the other ports of the PMF-switch.

For the Multiport Switch Options Only

- 7 Insert the 81591B or 81595B Optical Switch Module into the 8164B Mainframe in Slot 1.
- 8 Using the black protected patchcord provided with the switch, labeled "Switch Fiber", attach the common Connector A of the switch to the Optical Receiver port of the PDLA (or to Source Output port for switching at the DUT input).

For the Enhanced PDL and Loss Accuracy Option Only

- 9 Insert the 2 81634B Power Sensor modules into the 8164B Mainframe in Slots 2 and 3.
- 10 This step for connecting the power meters must be done correctly to avoid damage to the special patchcords used for this option.
 - The first patchcord is labeled "Monitor 1 Connector" at one end. This end should be attached to the analyzer at the Monitor 1 Port. The other end of this fiber is labeled "PM Slot2 Connector". This end should be attached to the 81634B module in Slot 2. This end has a special coating and should not make a physical connection to the fiber. This end should also not be cleaned with contact. Only air should be used if necessary for normal cleaning.
 - The second patchcord, labeled "Monitor 2 Connector" at one end, should be similarly connected between the Monitor 2 Port of the analyzer and the 81634B in Slot 3.
- 11 Turn on the PNA and TLS and allow the instruments to warm up for 90 minutes.
 - The 8164B needs to have the GPIB address 20. This will usually be set correctly at the factory. But if more than one 8164B are included, the GPIB addresses should be configured at installation.
 - **If the display screen of the 8164B is disabled by the 86038B software, then manual control can be returned by first using the 86038B menu item System>Toggle mainframe local-lockout status, and then pressing the button for Local on the right side of the 8164B display.**
 - Then press the button Config under the 8164B display and select GPIB address.
 - The "master" 8164B, located directly under the analyzer must have the address 20.

- A second 8164B should have the address 19. A third 8164B should have the address 18, and so on.
- If a special PMF-switch for multi-TLS automation is included, the unit attached to Port 2 of the switch should have the address 19, the unit attached to Port 3 should have the address 18, and so on.

For the Wavelength Meter Option Only

- 12 Place the 86122A multi-wavelength meter on top of the PNA.
- 13 Connect a GPIB cable from the 86122A to the GPIB port on either the PNA or TLS mainframe.
- 14 On the 86122A, press **Setup > GPIB Setup** and set the address to **10**.
- 15 Connect a patchcord from the 86122A Optical Input to TLS Output 1.
- 16 Continue with “[Performance Verification](#)” on page 35.

Agilent 86038B Front and Rear Panels

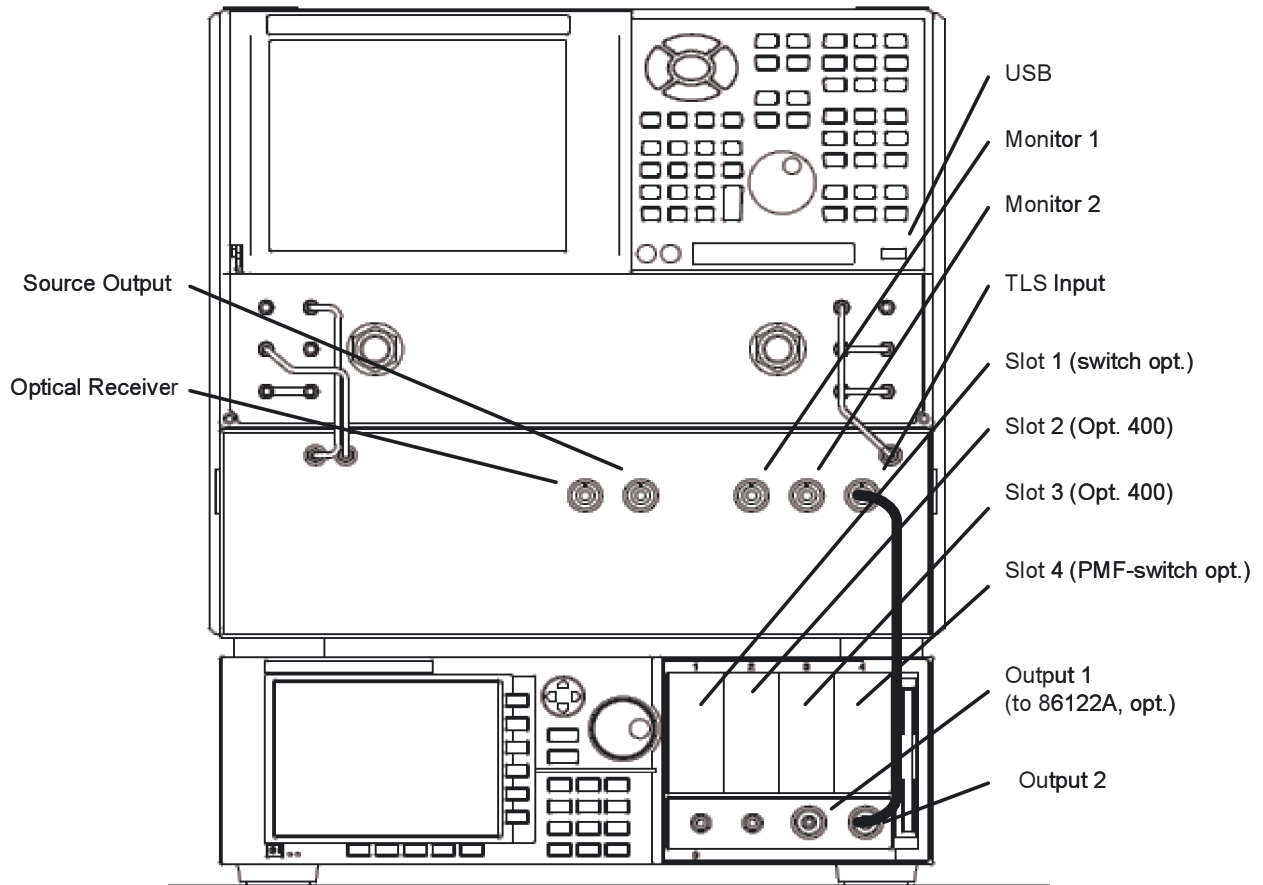


Figure 1 Front Panel

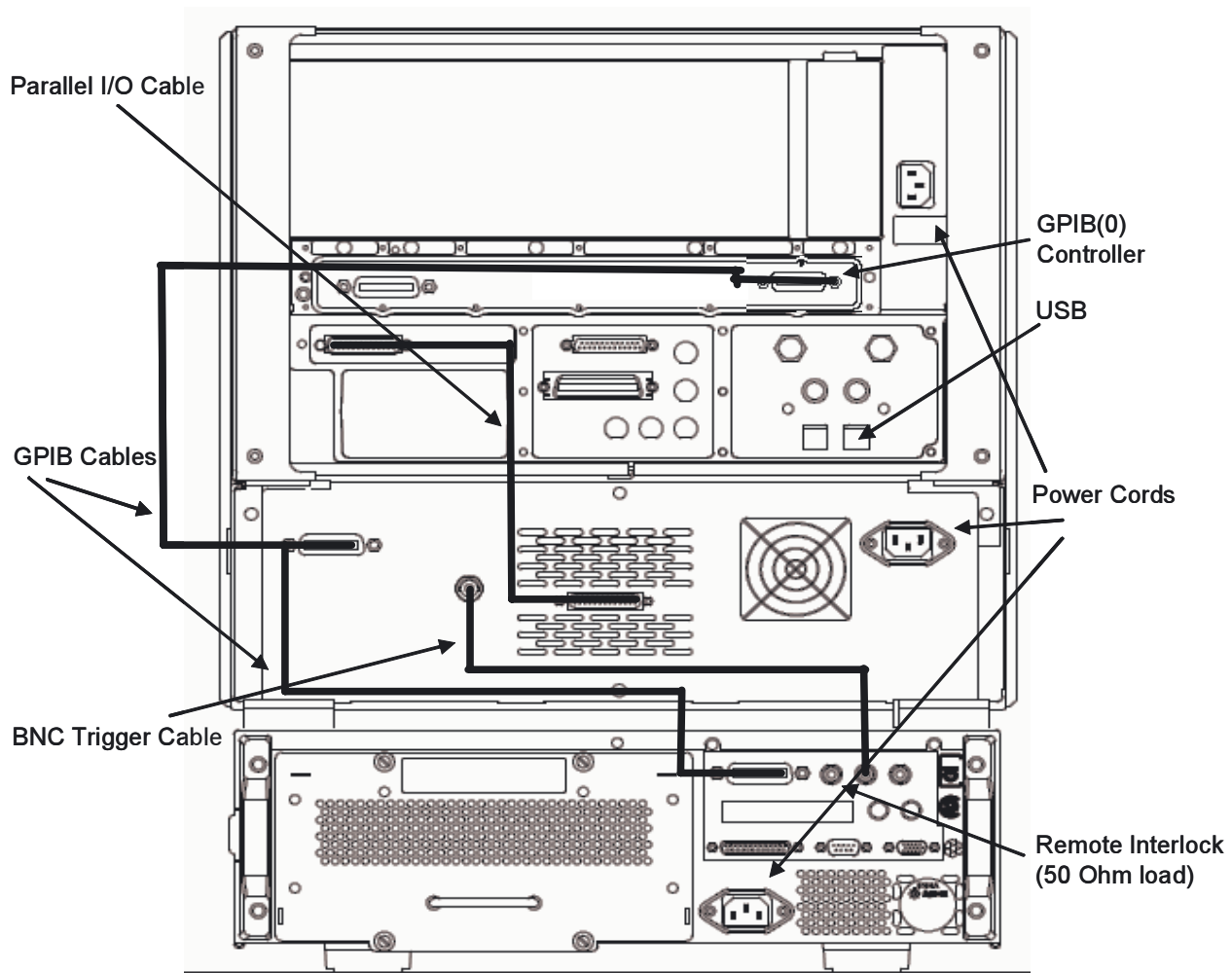


Figure 2.a Rear Panel

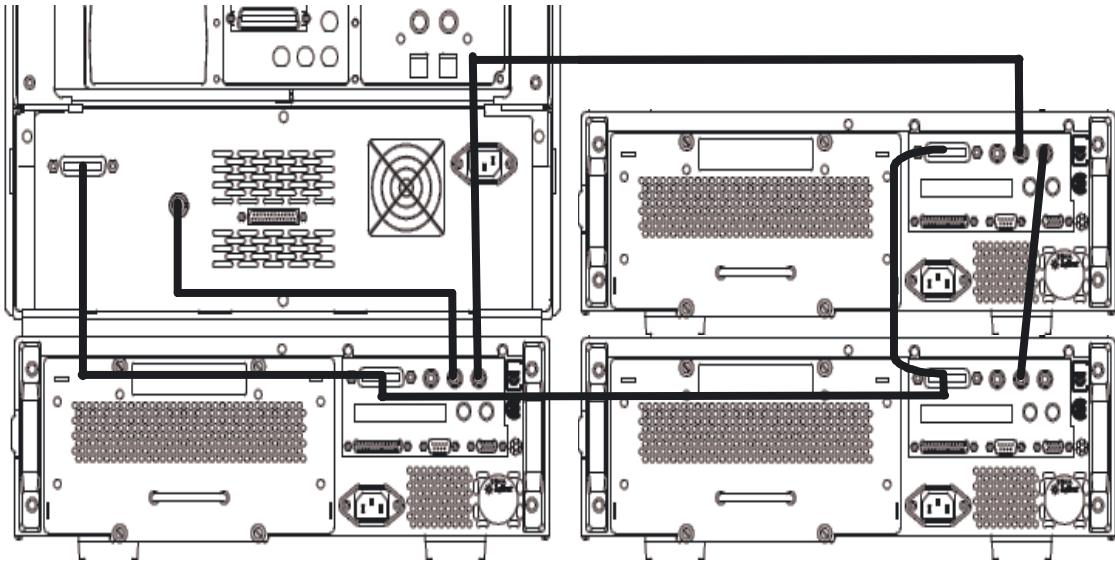


Figure 2.b Rear Panel

Performance Verification

Purpose

By performing the verification procedure upon delivery of the 86038B, you may have confidence that:

The PDLA is correctly connected

The PDLA was received in working order without damage due to shipment

Additionally, you may perform this procedure periodically to verify that the system is still in good operating condition. By recording the results of this procedure, and tracking these results over time, you will be able to detect any changes that may affect the measurement results of the PDLA.

Things to Consider

- Dispersion measurements are very sensitive to temperature changes. Insure that the PDLA has warmed up for at least 90 minutes. The verification DUT must also be allowed to reach ambient temperature for at least 60 minutes prior to measurement.
- Dirty connections will cause poor measurement quality. Be sure to clean all fiber ends, connectors, and test ports before making connections. Refer to [“Cleaning Information” on page 304](#).
- This procedure will take approximately 3 hours to complete (including the required 90 minute warm-up time). Allow enough time to fully complete the procedure.
- If you need assistance on navigating the front panel functions, refer to [“Navigating a Setup Panel” on page 46](#).

Procedure

- 1 On the front panel of the PDLA, press **Preset**.
- 2 Verify optical path. This will assure that the optical signal can be generated and detected and that good quality optical connections have been made.
 - a Connect the 2-meter reference patch-cord (not the MXF) between the Source Output and Optical Receiver

connectors of the 86038B. Arrange or fix the fiber so that it is stable.

- b From the Setup menu, select Laser Setup and set the power limit to 0 dBm.
- c From the System menu, select Optimize Input Polarizer.
- d From the System menu, select Optimize Optical Modulator Bias.
- e From the System menu, select the Real-Time Port Monitor.
- f The Traces in the “Gain/Loss” windows indicate the optical power at the detectors.
This should be between approximately -10 dBm and -15 dBm. If the power at the Monitor Port is too low, check the optical connections from the laser to the TLS Input. If the monitor power is acceptable, but the power at Port 1 is too low, then check the optical connections between the Source Output and Optical Receiver.

3 Set up the measurements with the parameters summarized in the table below, using the items in the Setup menu.

Parameter	Test Settings				
Options	#200	#116 / 316	# 115 / 315	#114 / 314	#113 / 313
Measurement	GD / CD + DGD / PDL Swept, Use 6 polarization states (if #400, use external power meter)				
Measurement Range: Start (nm)	1475	1510	1480	1390	1270
Measurement Range Stop (nm)	1625	1620	1580	1490	1375
Normalization Range	Use Measurement Settings				
Increment	1				
RF Modulation Frequency	2 000 MHz				
IF Bandwidth	70 Hz				
Length	Measured				
Index of Refraction	1.47				
Trigger Mode	Single Sweep				

4 Leaving the 2-meter reference patch-cord attached, as in Step 2, press the button Begin Normalization, select PDA Correction and then select Normalize. When the normalization measurement is completed, Close the normalization window.

- 5 Now leaving the reference patch-cord attached, reposition the loops of the patch-cord without making any sharp bends (to change the polarization), and again make sure the fiber position is stable. The stability can be checked by observing the Real-Time Port Monitor signal. Allow the fiber to settle for 1 minute. Then press the button **Begin Measurement**. This will produce a “zero-loss” and “zero-dispersion” measurement for evaluation.
 - a First check the measured Gain/Loss curve. (For guidance in using the graphical user interface, please refer to “[Defining Traces](#)” on page 45.) Since the normalization and measurement are made with the same connection, the ideal displayed loss should be 0 dB and a measurement curve should look similar to Fig. 3. The variation from 0 dB is a measure of the noise and repeatability of the loss measurement and should be within a tolerance of about ± 0.1 dB (with option 400, ± 0.02 dB).

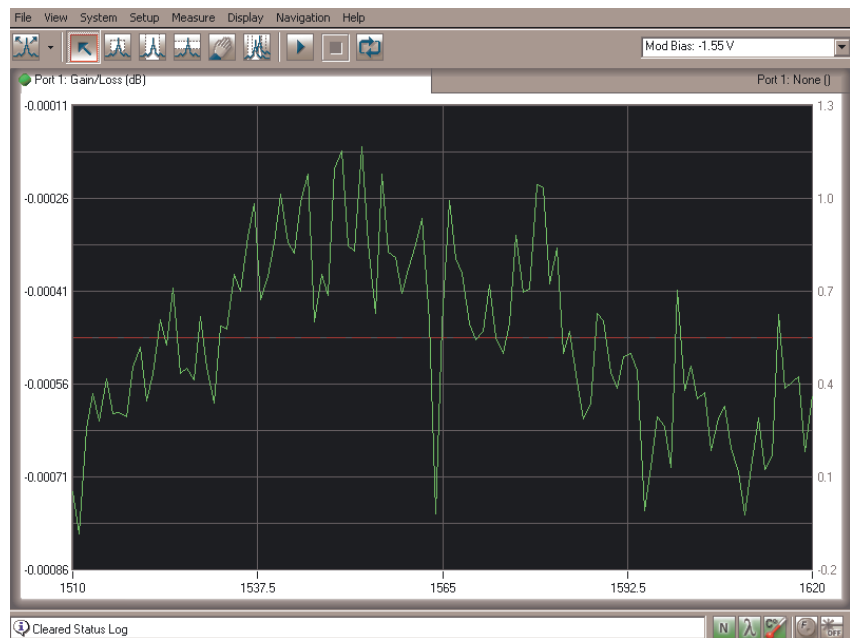


Figure 3 Sample Gain / Loss trace

- b Check the PDL curve. Since the normalization and measurement are made with the same connection and the patch-cord itself has little PDL, the ideal should be near 0 dB and a measurement curve should look similar to Fig. 4. The variation from 0 dB is a measure of the noise and intrinsic PDL of the measurement and should be within a tolerance of about ± 0.15 dB (with option 400, ± 0.05 dB).

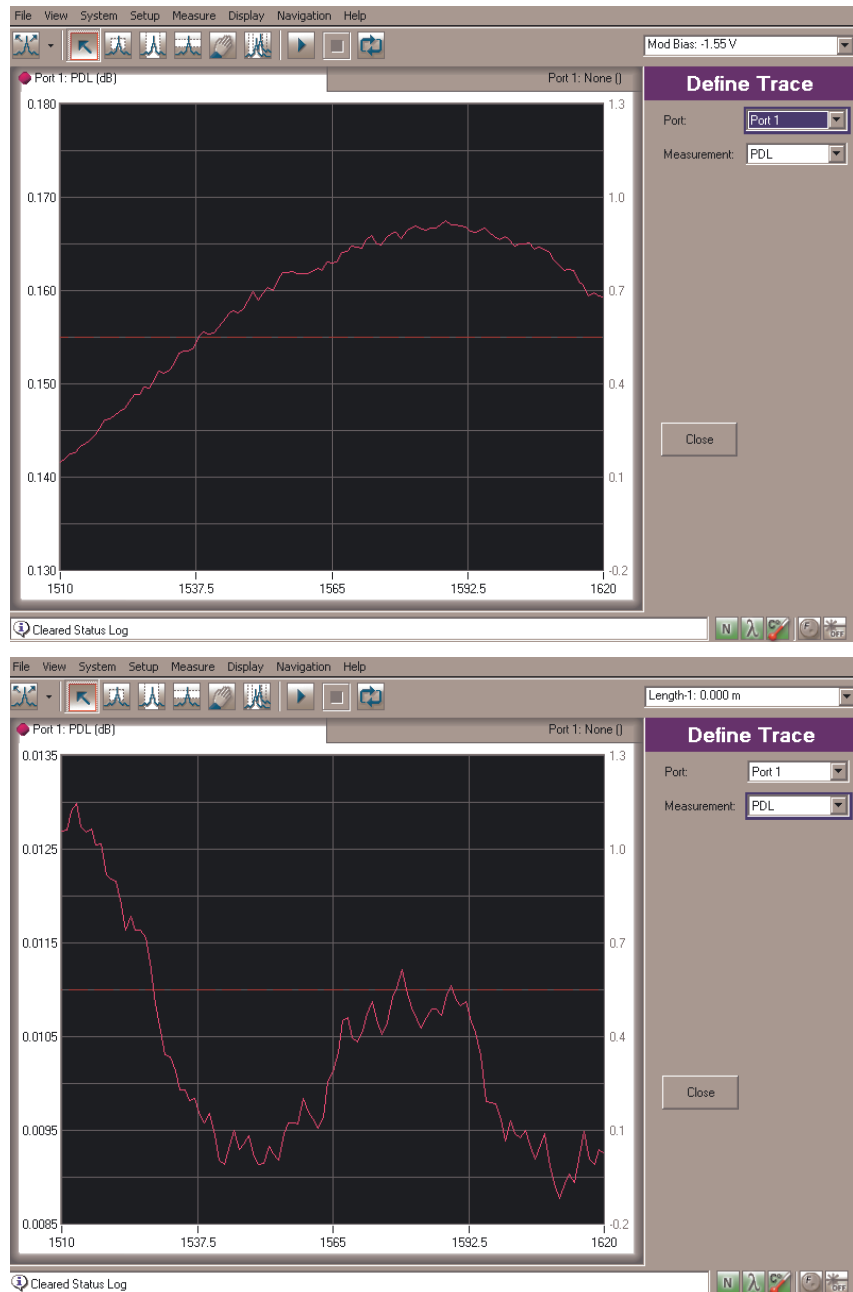


Figure 4 Sample PDL trace Top: for standard correction, Bottom: for Option 400

- c Check the GD curve. Since the normalization and measurement are made with the same connection, the relative GD should ideally be 0 ps and a measurement curve should look similar to Fig. 5. The variation from a linear fit to the data is a measure of the noise and

repeatability and should be within a tolerance of ± 0.03 ps. The offset of the data values from 0 ps. is an indication of the accuracy and should be within a tolerance of ± 0.05 ps. (Note however that temperature changes to the patch-cord can cause offsets larger than this.)

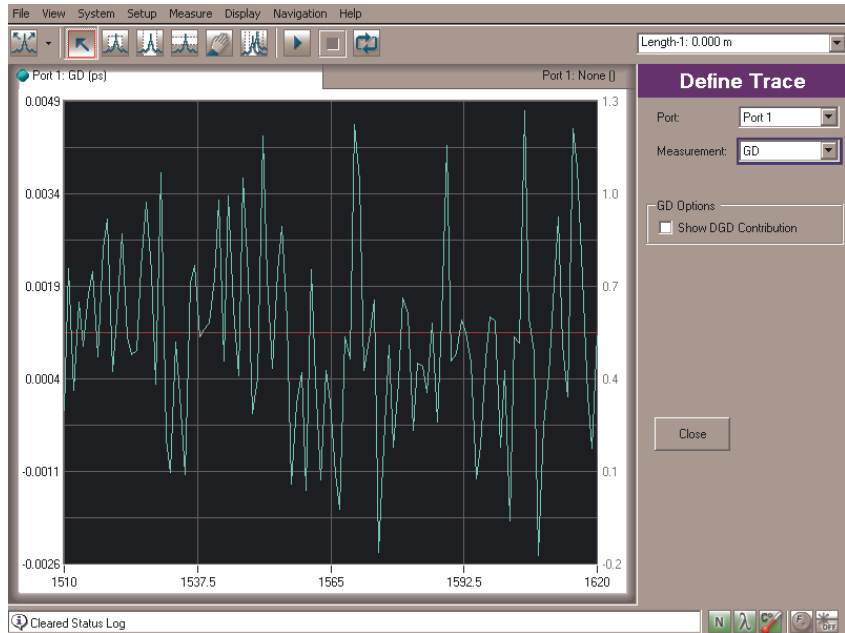


Figure 5 Sample GD trace

- d Check the DGD curve. Since the normalization and measurement are made with the same connection and the patch-cord has little DGD, the displayed DGD should be close to 0 ps, as in Fig. 6. The variation from zero is a measure of the noise and any intrinsic offset and should be within a tolerance of 0.1ps. Note also that the average DGD, PMD, can be displayed in the Define Trace tab and in the Information Field in the upper right corner. The value here should be less than 0.07 ps.

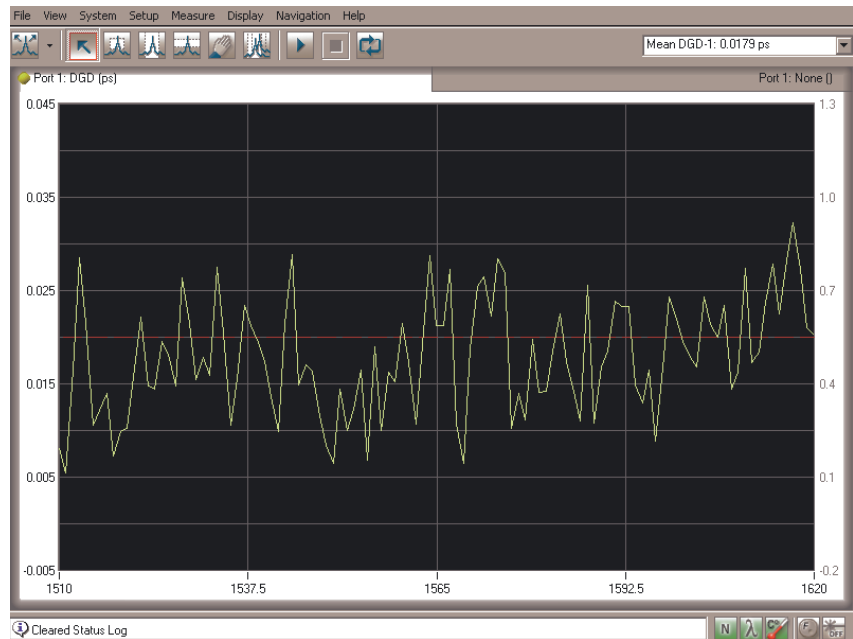


Figure 6 Sample DGD trace

- 6 Leaving the reference patch cord untouched, press Begin Length. Since the normalization and measurement are made with the same connection, the measured length value should be within ± 1 0.00002m.
- 7 For future reference and especially if there is a problem, save the measurement by selecting Save from the File menu and choosing a file name and directory.
- 8 Now disconnect the reference patch-cord from the 86038B Optical Receiver input connector. Using the FC/FC adapter, connect the free end of the reference patch-cord to the 2-m mixed-fiber patch-cord, MXF. Then connect the free end of the MXF patch-cord to the 86038B Optical Receiver input connector. Make sure that the fiber position is stable by using the Real-Time Monitor.
- 9 Then press the button Begin Measurement. This allows a measurement with additional CD and DGD.
 - a Check the DGD curve. The MXF patch-cord includes a length of polarization maintaining fiber to provide a DGD offset of about 0.15 ps. (This value is not calibrated for the device under test.) Check to see that the average DGD value lies between about 0.1 and 0.2 ps. Then use the Curve Fit function with a linear fit to display the Deviation values from the fit, as shown in Fig. 7. The

deviation trace represents the noise level of the measurement and should be within ± 0.05 ps.

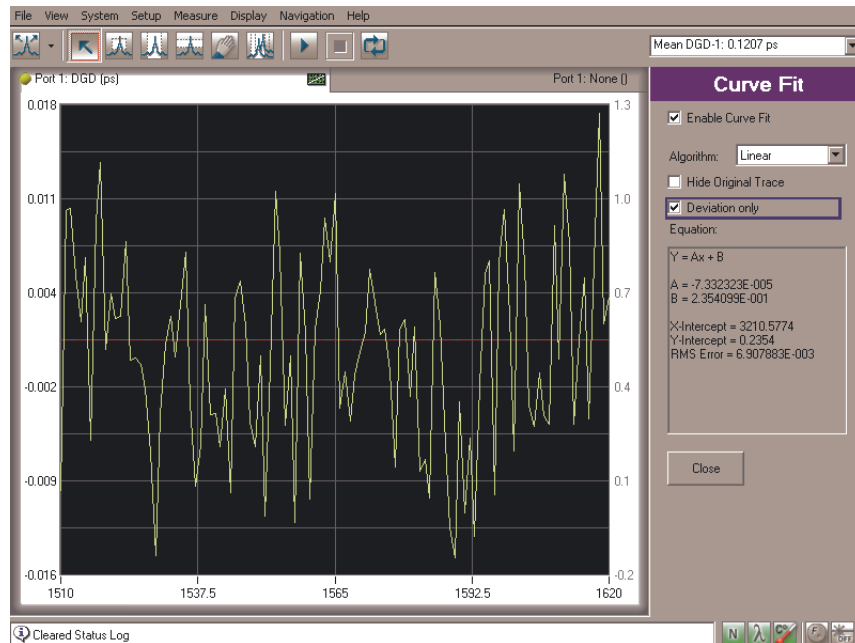


Figure 7 Sample DGD trace for the MXF fiber

- b Check the CD curve. The noise level can again be displayed using Curve Fit and Deviation only settings, as in Fig. 8. A Quadratic curve should make a good fit, but if it does not in the measured wavelength range, then a Sellmeier fit can be used. The deviation represents the noise level and should be within ± 0.07 ps/nm.

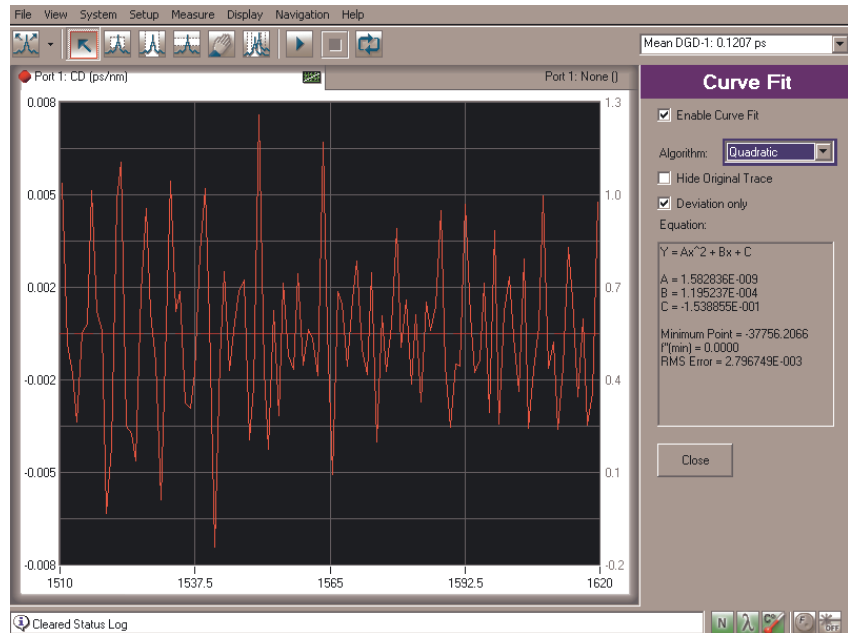


Figure 8 Sample CD trace for the MXF fiber

- c By pressing Begin Length, the length can be measured. This should be about 2 m. (This value is not calibrated for the device under test). Then the CD measurement can be displayed as CD Coefficient in ps/nm-km, as in Fig. 9. The value shown should be similar to the values for standard single-mode fiber. In particular it should show a value near 17 ps/nm-km in the range from 1550-1560 nm.

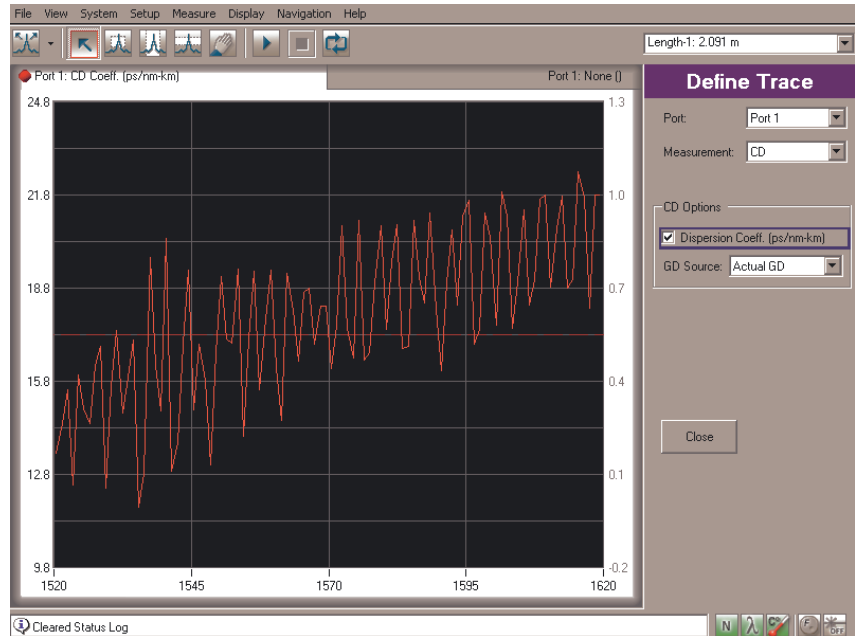


Figure 9 Sample CD Coefficient trace for the MXF fiber

This completes the verification procedure and together with up-to-date instrument calibration assures the specified performance.

Additional verification of Zero Dispersion Wavelength and Zero Dispersion Slope measurements is available by using the 86038B-510 Verification Fiber option. For this verification, please follow the instructions included with the 10 km verification fiber spool.

Navigating the Front Panel Functions

There are two different ways to access the instrument settings: From the front panel keys or from the menu bar. The front panel keys are separated into groupings by a group name which corresponds to the display menu bar names. For example, the Setup group of front panel keys has the exact same functions as the Setup menu on the menu bar. Either access method can be used alone or in combination for any measurement.

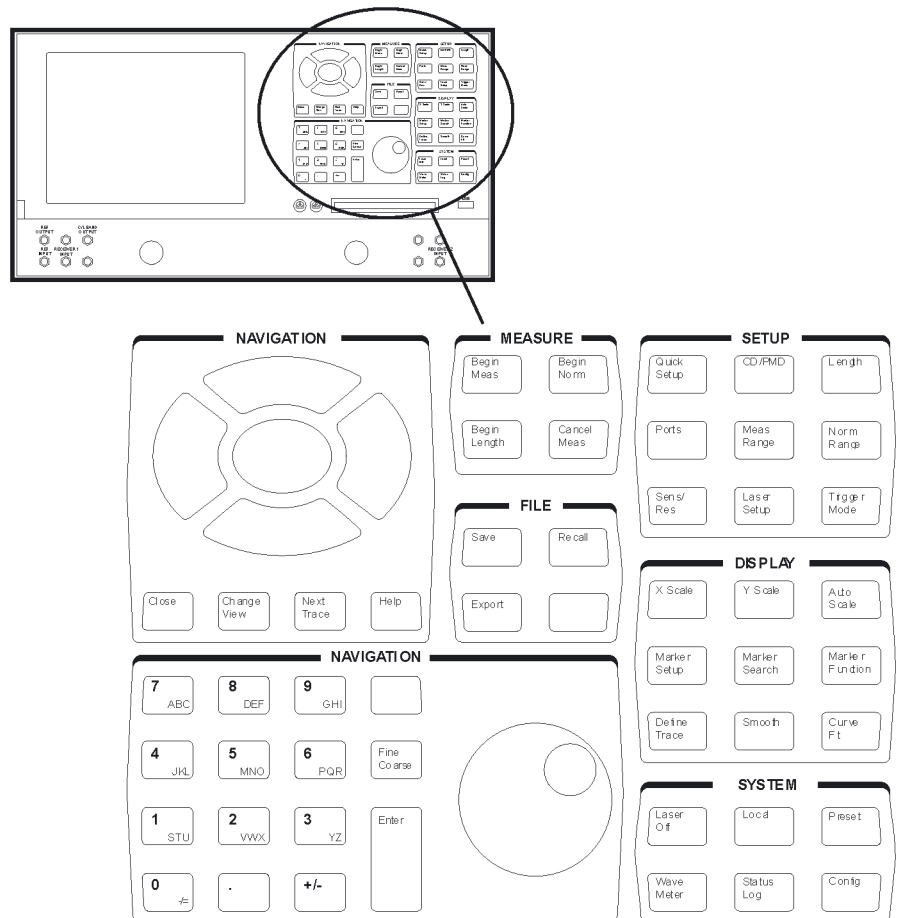


Figure 10 86038B User Interface

Defining Traces

The Define Trace function assigns a measurement on a specified port to a specific trace on the graph. Up to two measurement traces can be assigned to each of the four graphs.

The following example will show you how to assign the CD trace and the DGD trace for Port 1.

- 1 From the Navigation area, press **Change View** until all four graphs are shown on the display.
- 2 From the Navigation area, press **Next Trace** until the desired graph and trace tab are highlighted (that is, shown in white).
- 3 From the Display area, press **Define Trace**.
- 4 In the Define Trace setup panel, select *Port 1* and *CD* as the Measurement type.
- 5 From the Navigation area, press **Next Trace** to highlight the second trace tab on the graph.
- 6 In the Define Trace setup panel, select *Port 1* and *DGD* as the measurement type.
- 7 Press *Close* to save the trace definitions.

For each trace you can specify any active port's measurement traces and one of nine measurement types.

- GD (Group Delay)
- CD (Chromatic Dispersion)
- DGD (Differential Group Delay)
- PDL (Polarization Dependant Loss)
- Gain/Loss
- PCD (Polarization Dependent CD)
- Depol. Rate (Depolarization)
- 2nd Order PMD
- Optical Phase

Select Active Trace

From the Navigation area, press **Next Trace** until the desired graph and trace tab are highlighted (that is, shown in white).

Toggling Between Graph Views

From the Navigation area, press **Change View** until all four graphs are shown on the display.

Navigating a Setup Panel

Most front panel keys access a setup panel where additional parameters can be defined.

- Use the navigation keys (arrow keys) to move from field to field.
- When a text box is highlighted, you can enter a new value using the RPG knob or the numeric key pad. When a check box is highlighted, use the Select navigation key to toggle between states.
- When a drop-down list box is highlighted, press Select to open the list. Then, use the navigation keys to select the value. Press Select to choose the function.
- When a check box is highlighted, use the Select navigation key to toggle between states.
- When a drop-down list box is highlighted, press Select to open the list. Then, use the navigation keys to select the value. Press Select to choose the function.

Tip: The Select navigation key works the same way as a mouse click.

Quick Setup Procedure

For your convenience, a quick setup feature has been included on the PDLA. The Quick Setup key accesses a series of control panels that help you specify the basic settings needed for making a measurement. A measurement time estimation is provided so that you can trade off measurement time versus the measurement settings.

The following procedure outlines the steps to make a precisely defined measurement.

NOTE

For experienced users, a detailed process can be used which provides more flexibility than the Quick Setup procedure. Refer to [“Flexible Measurement Process” on page 82](#).

To get the most performance out of your PDLA, it is recommended that you read [Chapter 2, “Measurement Concepts”](#), [Chapter 3, “Making Measurements”](#), and [Chapter 4, “Function Reference”](#). These three chapters cover the linkages between the different measurement that can be made, specific devices being measured, and the setup parameters used to make the measurements.

Prepare the Normalization

- Connect the reference patch cord for the normalization between the Source Output and Optical Receiver connectors.
- Check the connections with the Real Time Port Monitor. Refer to [“Real-Time Port Monitor” on page 101](#).

Quick Setup Process

- On the front panel, press Quick Setup.
- Select DUT configuration and input ports.
- Choose which attributes to measure. Select stepped or swept mode.
- Set the measurement wavelength range.
- Set the resolution slider (lowest to highest resolution).
- Begin the Normalization



Measurement Process

- Connect the DUT.
- Begin the Measurement.
- View the data and manipulate the display and markers.
- Smooth or curve fit the data as desired.
- Save the measurement and/or export the data.

2

Measurement Concepts

Introduction.....	50
Understanding the Attributes Being Measured	51
Gain/Loss	51
Group Delay/Chromatic Dispersion/Optical Phase .	52
Polarization Mode Dispersion/Differential Group Delay	
54	
Polarization Dependent Loss (PDL).....	56
Fiber length	56
Description of the 86038B	57
Measurement Method (MPS).....	59
Choosing Setup Parameters	61
Multiple Port Configurations.....	73
Normalization and Measurement	74
Swept and Stepped Measurement Modes.....	77

Introduction

NOTE

To get the most performance out of your PDLA, it is recommended that you read [Chapter 2, “Measurement Concepts”](#), [Chapter 3, “Making Measurements”](#), and [Chapter 4, “Function Reference”](#). These three chapters cover the linkages between the different measurements that can be made, specific devices being measured, and the setup parameters used to make the measurements.

Understanding the Attributes Being Measured

The 86038B allows you to measure the following attributes:

- Gain/Loss
- Group Delay (GD)
- Chromatic Dispersion (CD)
- Optical Phase
- Polarization Mode Dispersion (PMD) including 2nd - order PMD
- Differential Group Delay (DGD)
- Polarization Dependant Loss (PDL)
- Length

This section will define each of these measurement attributes and provide some information of the mechanism that produce the effects.

Gain/Loss

Insertion Loss

Insertion loss is a measure of the attenuation of the optical signal level between the input and output of a device. It is defined as $10 \log (\text{power in}/\text{power out})$ and is always a positive number for devices without gain. The mechanisms that produce insertion loss include simple absorptive effects as well as scattering and reflective effects.

In long fibers, the insertion loss scales with the fiber length. Then the parameter Attenuation Coefficient is useful, which is the insertion loss divided by the fiber length, usually expressed in units of dB/km.

Tips on Measuring Insertion Loss (IL) For the most accurate IL measurements, the PDL should be measured at the same time to average the polarization dependence out of the IL result. A 30 Hz IF bandwidth can be used to minimize noise. Refer to “CD/PMD” on page 104 and to “Sensitivity/Resolution” on page 108.

- When measuring ripple effects in either GD or IL, the step size and modulation frequency are each important. If the step size is too coarse, there is the risk of aliasing the ripple rate to appear at a longer period. Even when the step size is sufficiently small to avoid aliasing, The modulation frequency could be large enough to reduce the apparent peak-to-peak amplitude of the ripple.

Gain

Gain is a measure of the increase of optical signal level with transmission through the test device. It is defined as $10 \log(\text{power out}/\text{power in})$ and is always a positive number. In the field of lightwave telecommunications, gain is produced by technologies including doped fiber, semiconductor and Raman amplifiers.

The 86038B displays loss and gain within a single trace, representing gain with a positive sign and loss with a negative sign.

Group Delay/Chromatic Dispersion/Optical Phase

Mathematically, the group delay of a component, fiber or network is proportional to the derivative of optical phase of the transmitted signal with respect to optical frequency. More intuitively, group delay is the time required for information in a signal (that is, any specific point on the modulation waveform) to travel the length of the optical path. By contrast, the phase delay is the time required for a point on the unmodulated optical carrier wave to travel the length of the optical path. The 86038B measures the group delay, not the phase delay.

The group delay of most optical fiber and devices varies with wavelength, a phenomenon referred to as chromatic dispersion. Mathematically, the chromatic dispersion is the second derivative of the optical phase with respect to optical frequency. In conventional terms, the chromatic dispersion at a given wavelength is simply the local slope of the curve of group delay versus wavelength, expressed in picoseconds per nanometer (ps/nm). In a non-dispersive medium the group delay curve is independent of wavelength and the group and phase delays are equal.

The chromatic dispersion coefficient at a given wavelength is defined as the chromatic dispersion at that wavelength divided by the length of the fiber, expressed in ps/(nm x km).

The zero-dispersion wavelength (usually called lambda zero) is the wavelength at which the slope of the group delay versus wavelength curve equals zero.

The dispersion slope at lambda zero is in turn the slope of the dispersion curve at that wavelength. The slope at lambda zero is used by system designers to estimate the dispersion at wavelengths away from lambda zero.

Optical components such as wavelength filters also have dispersion, due to the mathematical relationship between the dependence on wavelength of transmission amplitude and phase. Often the GD dependence on wavelength over a pass-band is not conveniently described by a single slope value, CD, and is better described by the GD variation range within the pass-band. In the case of a dispersion compensating device, the deviation of GD from the GD spectrum of the desired CD value, is a useful parameter often called GD ripple.

Similarly, the deviation of the Optical Phase from design is valuable to evaluate a device. The Optical Phase, as the frequency integral of GD (multiplied by -1), is ideally linear in optical frequency for a device with no dispersion. The negative slope of optical phase is the absolute group delay and is proportional to the length of the optical path. Usually only relative phase is of interest and optical phase is reported with an arbitrary constant offset.

Tips on Measuring Group Delay (GD)

- In the swept mode, over-sampling the GD spectrum by using a small wavelength step than required for the desired wavelength resolution can be used together with smoothing at that resolution to reduce the noise. But if the wavelength step is too large, smoothing will remove wavelength-dependent details from the spectrum.
- For the most accurate GD measurements, the measurement should be made with polarization dependence (GD/CD + DGD/PDL). This allows any DGD and PDL effects to be averaged out. The lowest noise can be achieved with high modulation frequency and low IF bandwidth frequency. Refer to “CD/PMD” on page 104 and to “Sensitivity/Resolution” on page 108.

- When making precision GD measurements, use the Real Time Port Monitor to verify DUT stability before normalizing or measuring. The length and thus GD of long fibers changes significantly with temperature, so rapid temperature changes should be avoided. Fiber connectors can also be a source of instability and should be cleaned and secured carefully. Fibers require protection from normal room air currents and need to be secured and allowed time to settle after handling. Refer to “[Real-Time Port Monitor](#)” on page 101. Especially for stepped-wavelength measurements, which take more time, the drift correction feature of the 86038B dramatically reduces sensitivity of the GD measurements to DUT instability by referencing each GD measurement point to a reference wavelength measurement.
- When measuring ripple in either GD or IL spectra, the step size and modulation frequency are each important. If the step size is too large, there is the risk of aliasing, giving the wrong ripple period. Even when the step size is sufficiently small to avoid aliasing, a too high modulation frequency could increase the signal linewidth and thus resolution so as to falsely reduce the apparent peak to peak amplitude of the ripple. The ripple can be displayed as “Deviation” from the Curve Fit function.

Tips on Measuring Chromatic Dispersion (CD)

- The combination of moderate smoothing of both the GD trace and the CD trace may lower the CD trace noise and preserve the CD wavelength resolution better than smoothing the CD trace alone. Refer to “[Smooth](#)” on page 125.

Polarization Mode Dispersion/Differential Group Delay

The group delay of an optical element can depend on the polarization state of the light. The differential group delay, DGD, is the difference between the GD values at the two principal states of polarization, which are orthogonal to each other. In a simple element like a waveplate, the PSP correspond to the fast and slow axes. The DGD at a particular wavelength thus gives the magnitude of the GD dependence on polarization and is usually expressed in ps. The DGD at the wavelength of the signal determines the amount of signal degradation.

Optical fiber can have nonzero DGD due to imperfect cross-section or due to external stress. Each segment of a long fiber can have a different set of PSP, randomly oriented from segment to segment. When the fast PSPs are closely aligned then the DGD values of segments will be added, while alignment of fast

with slow PSP allows the total DGD to be reduced. This degree of alignment along a fiber typically depends on wavelength and also varies with time, due to temperature and stress changes. Therefore the DGD at a given time for a certain wavelength may not be the same as obtained in an earlier measurement. But an average over many measurement points can be used to determine how the DGD will behave statistically over time. In particular, the probability that a certain DGD limit is exceeded can be determined. This average DGD value is known as polarization mode dispersion, PMD, which is thus a useful parameter for optical fiber and other complex optical paths.

The statistical way that DGD accumulates in a fiber span means that it increases proportional to the square-root of the fiber length. Thus, the PMD coefficient, is defined as the PMD divided by the square root of the fiber length, expressed in ps/ $\sqrt{\text{km}}$.

The causes of PMD are several. Even the simplest optical device usually exhibits a slight dependence of group delay on input polarization. The behavior may be produced by anisotropy of the index of refraction in directions perpendicular (transverse) to the direction of propagation. Differential group delay is also observed in devices in which the light is split into two paths according to polarization and later recombined. In this situation, any difference in the parallel path lengths produces a differential group delay. In optical integrated circuits a common cause of PMD is the asymmetry in the cross sectional profile of the waveguides. In optical fiber, the main cause of differential group delay is non-circularity of the optical core. The problem was extremely severe in some of the older installed fibers.

Tips on Measuring Differential Group Delay (DGD)

- Good DGD measurements with the 86038B depend on making low-noise GD measurements and on stability of the DUT during the measurement. This means that high modulation frequency and low IF bandwidth are beneficial. Especially for measurements of long fibers, the drift correction feature of the 86038B dramatically reduces the sensitivity of the swept-wavelength DGD measurements to DUT instability, especially temperature drift, by remeasuring the GD spectrum at the first polarization state at the end of the measurement. When making precision DGD measurements, use the Real Time Port Monitor to verify GD drift before normalizing or measuring. Refer to “[Real-Time Port Monitor](#)” on page 101.

- When measuring DGD, any GD drift or disturbance of the DUT GD during each measurement can bias the measurement point to higher DGD. This effect might be seen as a single measurement point spike if the DUT or fibers were bumped during the measurement.
Note that DGD is far more sensitive to this than PDL.

Polarization Dependent Loss (PDL)

Polarization dependent loss (PDL) is the peak-to-peak variation in the loss of a test device over all polarization states.

The mechanisms that produce PDL include material absorption, oblique optical reflections, fiber bends and various aspects of optical circuit design. In cases of passive components exhibiting pure PDL (without significant PMD), the polarization states corresponding to maximum and minimum loss are diametrically opposite one another on the Poincare sphere.

Tips on Measuring Polarization Dependent Loss (PDL)

- Similar considerations as for DGD apply to making accurate PDL measurements. In particular, noise and drift should be minimized. Care should also be taken to avoid other sources of PDL in the measurement setup, like switches with appreciable PDL or noncontacted angled connectors.

Fiber length

This measurement may be applied to any test device, but is usually applied to optical fiber. The fiber length is derived from measurement of the derivative of RF modulation phase with RF frequency, at a user-defined wavelength. The optical length measurement uncertainty is below 10^{-6} of the total length, for fibre lengths ≤ 50 km. The 50 km length limit applies to the specification, but very accurate measurements can be made to any practical length. The measurement determines optical length, nL , so calculation of physical length requires the user to enter the value of the index of refraction; any error in the index of refraction directly impacts the length measurement.

Note also that the length also changes with temperature significantly with respect to this accuracy level. A rough figure for the temperature dependence of the optical length on temperature for standard SMF is about $10^{-5}/^{\circ}\text{C}$.

Description of the 86038B

The Agilent 86038B employs the modulation phase shift (MPS) method for all of its measurements. A block diagram of the system is shown in [Figure 11](#). Light from the tunable laser is amplitude modulated by passing the light through a balanced lithium niobate Mach-Zehnder modulator that is driven by an RF source. The signal passes through the test device to an optical receiver for recovery of the modulation envelope. The performance network analyzer (PNA-L) measures the amplitude and phase of the envelope. For measurements of polarization dependence, the polarization is set by a polarization controller that directly follows the modulator. Thus the optical stimulus to the DUT can be varied in both wavelength and state of polarization (SOP). From the resulting information, the various measurement attributes are evaluated.

The available wavelength range can be chosen, based on the wavelength option of the 81600B tunable laser. The modular laser can easily be exchanged to make measurements in a different wavelength region. But even without changing the laser module, the 81600B can provide up to a 200 nm wide wavelength range.

An additional internal optical receiver also allows the amplitude and phase of the signal into the DUT to be monitored, thus providing optimized separation of the DUT properties from any other changes to the optical signal.

When equipped with Option #400, the 83427C includes additional couplers to tap the optical signal to and from the DUT. The tapped signal is routed to two high performance power sensor modules mounted in the 8164B. The low polarization sensitivity of these sensors enhances the accuracy of the loss and PDL measurements.

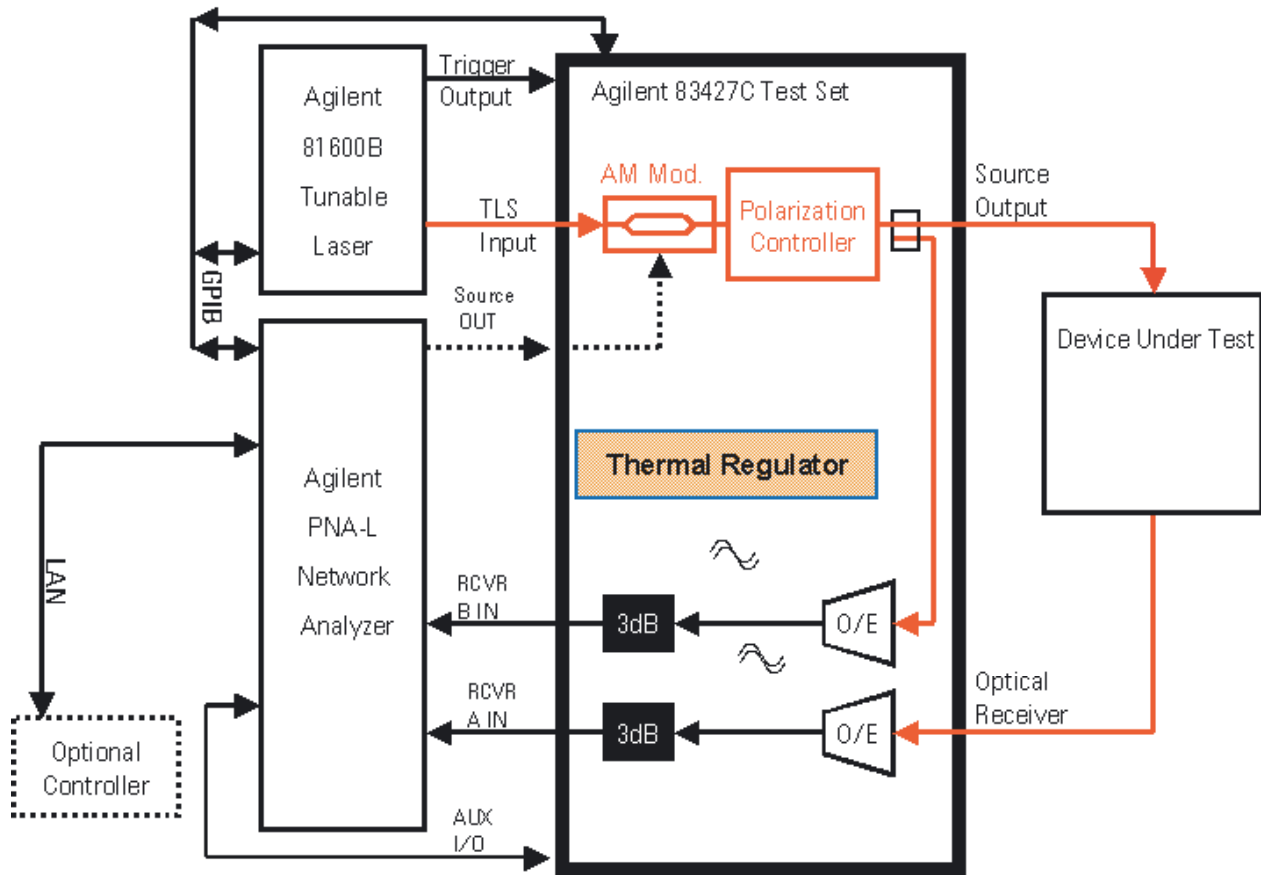


Figure 11 86038B Block Diagram

Measurement Method (MPS)


For the group delay/chromatic dispersion measurement, the wavelength is stepped or swept and the change in RF modulation phase is recorded. The change of phase is converted to an equivalent change in the group delay according to the formula shown in [Equation 1](#). The derivative of group delay with respect to wavelength gives chromatic dispersion. See [Equation 2](#).

The loss or gain measurement proceeds in the same way, but the measurement is based on the amplitude of the recovered modulation envelope.

Both the group delay and the loss/gain measurement require an initial normalization measurement of a through path. After normalization, the test device is inserted in the through path and the group delay and loss/gain is measured in comparison with the normalization data.

PDL and PMD measurements are made by performing the above measurement four (or optionally six) times at different states of input polarization. Using Mueller Matrix analysis, these measurements allow calculation of the phase shift and amplitude for any other polarization state, and in particular for those that give maximum and minimum values. The DGD at each wavelength is calculated using the minimum to maximum phase change in the same formula used to calculate group delay, but in this case the polarization is varied rather than the wavelength. The PDL information is similarly obtained from the minimum to maximum variation in amplitude of the recovered modulation envelope.

The fiber length measurement is based on the variation of modulation phase as the RF modulation frequency is changed. For this measurement, the wavelength is kept constant. Intuitively, the length of the test device can be expressed in terms of an equivalent number of modulation periods. As the RF modulation frequency is increased, the test device is more and more RF cycles in length. This increase is more rapid when the test device is long. This is the basis for the length measurement performed by the 86038B. The very high frequency accuracy and tuning range of the instrument and the algorithm used



assure extremely high accuracy over a very wide length range, from less than a millimeter to 50km and beyond. The phase samples for the length measurement are made at two orthogonal states of optical polarization so that the result is polarization averaged.

The Optical Phase trace is calculated as the optical-frequency integral of the GD. Since the GD trace displays relative GD values, the measured or specified length value is used to provide an absolute scale for the GD. Thus the slope of Optical Phase does correspond to absolute GD when the length is correctly determined. The Optical Phase trace is relative and begins at zero.

Choosing Setup Parameters

This section presents the mathematical relationships on which the group delay and dispersion measurements are based. You will see how wavelength resolution and measurement sensitivity depend on the instrument setup parameters. This knowledge will help you to effectively resolve group delay ripple, avoid phase wrapping errors, avoid aliasing errors, and improve the repeatability (reduce the noise) of chromatic dispersion and relative group delay measurements.

Background

The Agilent 86038B employs the modulation phase shift method. Light is intensity modulated with an RF tone and applied to the device under test. The transmitted (or reflected) signal is detected to recover the modulation envelope, and the envelope phase is measured relative to the RF source. Any change in the group delay $\Delta\tau$ of the test device produces a corresponding change in the modulation phase. In practice, the wavelength is stepped or swept and the change in the group delay $\Delta\tau$ for each wavelength increment is calculated from the measured change in phase according to:

Equation 1

$$\Delta\tau_{(\Delta\lambda)} = \frac{\Delta\phi}{360^\circ} \cdot \frac{1}{f_m}$$

where $\Delta\phi$ is the phase change in degrees produced by a small wavelength step, f_m is the modulation frequency in Hz, and the subscript $\Delta\lambda$ indicates that the change in group delay being measured was produced in response to an incremental change in wavelength. The first term on the right side of [Equation 1](#) is the fraction of cycles of modulation phase shift produced by the wavelength change. The second term is the time period of a single RF cycle ($1/\text{freq} = \text{period}$). The product of the two terms has units of time.

The attribute called dispersion is defined by:

Equation 2

$$D = \frac{\Delta\tau}{\Delta\lambda}$$

where $\Delta\tau$ is the change in group delay in seconds corresponding to a change in wavelength $\Delta\lambda$ in meters. In practice, the dispersion is expressed in units of picoseconds per nanometer (ps/nm), where $1\text{ps}=10^{-12}$ seconds.

The dispersion coefficient expresses the relationship of dispersion to fiber length and is used to specify optical fiber. It is obtained by dividing the dispersion value by the length of the fiber:

Equation 3

$$D_{\text{coeff}} = \frac{1}{L} \cdot \frac{\Delta\tau}{\Delta\lambda}$$

where L is the length of the fiber. The dispersion coefficient is usually expressed in ps/nm·km.

Combining [Equation 1](#) and [Equation 2](#), we obtain:

Equation 4

$$\Delta\phi = 360^\circ \cdot D \cdot f_m \cdot \Delta\lambda$$

[Equation 4](#) shows that the amount of phase change measured in response to a wavelength step is the product of device dispersion, modulation frequency and wavelength step. This equation provides several key insights into the capabilities of the modulation phase shift measurement method.

Relationship of setup parameters to measurement noise

Here we examine how the terms on the right hand side of [Equation 4](#) influence measurement noise. This relationship is expressed by adding a phase noise term to [Equation 4](#):

Equation 5

$$\Delta\phi_{\text{total}} = (360^\circ \cdot D \cdot f_m \cdot \Delta\lambda) + \Delta\phi_{\text{noise}}$$

[Equation 5](#) shows that as the product of dispersion, modulation frequency and wavelength step grows smaller, the impact of phase noise increases.

Three measurement applications sometimes require that wavelength increment and/or modulation frequency be reduced, possibly at the expense of increased noise. These are:

- Resolution of spectrally narrow group delay features

- Avoidance of phase wrapping when measuring highly dispersive devices.
- Avoidance of aliasing when measuring ripple.

Equation 5 warns us not to reduce modulation frequency more than necessary.

Relationship of setup parameters to phase wrapping

Equation 4 implies that the selection of modulation frequency and wavelength increment play a part in the potential problem of phase wrapping. The phase detectors used in the Agilent 86038B have a range of ± 180 -degrees. Exceeding the range of the phase detector causes a phase wrapping error. The wrap-free range is expressed by:

Equation 6

$$\Delta\phi = 360^\circ \cdot D \cdot f_m \cdot \Delta\lambda$$
$$\Delta\phi < \pm 180^\circ$$

Consider this limitation in the context of the stepped mode of operation. After each wavelength step, the phase detector is automatically zeroed to make the entire ± 180 -degree range available for the next step. If the next wavelength step produces a phase change larger than ± 180 degrees, the phase detector loses the history of the actual phase change and gives us only an apparent phase change. For example, if a wavelength step produces an actual phase change of positive 185 degrees, it will be interpreted by the phase detector as a change of negative 175 degrees. This is the phase wrapping error. The phase detector is zeroed mathematically, not physically, so there is no noise contribution, regardless of the wavelength step size. Use Equation 6 to check your selections of wavelength increment step and modulation frequency, to assure that the ± 180 degree limit is not exceeded. For convenience, the Sens/Res Setup Panel displays a calculation of the maximum measurable CD for the chosen parameters.

Phase wrapping errors can also occur in the swept mode, though they are less likely to occur when using small wavelength increments (large numbers of amplitude and phase samples across the measurement wavelength range).

Diagnosing a phase wrapping error is not always easy, because the extent to which phase wrapping affects the shape of the measured relative group delay curve depends upon the details of both the test device and the measurement setup parameters. Be conservative in the use of Equation 6, as it is possible for phase wrapping errors to go entirely unnoticed. In addition to the display of maximum CD in the Sens/Res setup panel, Agilent provides a similar calculation in an Excel spreadsheet calculator for estimating the maximum allowable step size. It is called the Wavelength Step Calculator for Modulation Phase Shift CD Measurements and can be found in the Library section of the public Agilent product web page for the 86038B.

Maximum RF Modulation Frequency (MHZ)

The following table lists recommended modulation frequencies. Any modulation smaller than that listed will give correct results (that is, no phase wrapping). Smaller modulation frequencies will have more noise and better resolution. Table cells that have an arrow symbol (\mathcal{R}) indicate that you must decrease the wavelength increment to achieve the desired $\Delta\tau$ RGD.

Table 2-1. Maximum RF Modulation Frequency (MHz) for $<90^\circ$ phase steps $f_{90} = f_{90} = 1/(4 \times \Delta\tau \times \Delta\lambda)$

$\Delta\tau$ RGD (ns/nm)	$\Delta\lambda$ Wavelength Increment (nm)							
	0.001	0.005	0.01	0.05	0.1	0.5	1	5
0.1	any	any	any	any	any	any	2500	500
0.5	any	any	any	any	any	1000	500	100
1	any	any	any	any	2500	500	250	50
5	any	any	any	1000	500	100	50	10
10	any	any	2500	500	250	50	25	5
50	any	1000	500	100	50	10	5	\mathcal{R}
100	2500	500	250	50	25	5	\mathcal{R}	\mathcal{R}
500	500	100	50	10	5	\mathcal{R}	\mathcal{R}	\mathcal{R}
1000	250	50	25	5	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}
5000	50	10	5	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}
10000	25	5	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}
50000	5	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}	\mathcal{R}

Relationship of setup parameters to aliasing

Since stepped measurements are sampled at discrete wavelengths, there is no measurement of the performance between the sampled points. Thus wavelength increment and the RF modulation sideband separation must be chosen to be small compared to the wavelength spacing over which the device under test has variations of interest.

When a device such as Fabry-Perot filter or interleaver has periodic variations across wavelength, known as amplitude or group delay ripple, the wavelength increment must be small enough to collect two or more samples per period. If the wavelength increment is so large that less than two samples per ripple period are measured, the result will be aliased to an incorrect representation. When the result is aliased, it appears at a false period, longer than the actual period. An aliased measurement often has an unusual shape, but can appear with the expected shape at a multiple of the expected period.

To eliminate aliasing, use a smaller wavelength increment. If the RF modulation sideband separation is less than the wavelength increment, and the wavelength increment can be reduced without changing the period or shape of the measurement, then aliasing is not present.

Relationship of setup parameters to resolution

Two measurement parameters affect resolution: the wavelength increment and the RF modulation frequency. In most cases both are reduced to improve wavelength resolution or amplitude resolution. The RF modulation frequency may be constrained by phase wrapping, noise, imaging, or ripple amplitude accuracy. The wavelength increment may be constrained by aliasing, phase wrapping, ripple amplitude accuracy, or measurement speed. Since these have overlapping constraints, it may be necessary to compromise on acceptable settings.

The three-dimensional graph in [Figure 12](#) shows the selection of wavelength step and modulation frequency to achieve high wavelength resolution. Reduction of only modulation frequency may fail to achieve the desired resolution and simply raise the noise floor.

Since the modulation frequency limits the wavelength resolution, if a wavelength increment is chosen significantly smaller than this resolution in swept-wavelength measurements, the 86038B performs a Gaussian smoothing on the data at the relevant resolution. This reduces noise while maintaining the true resolution.

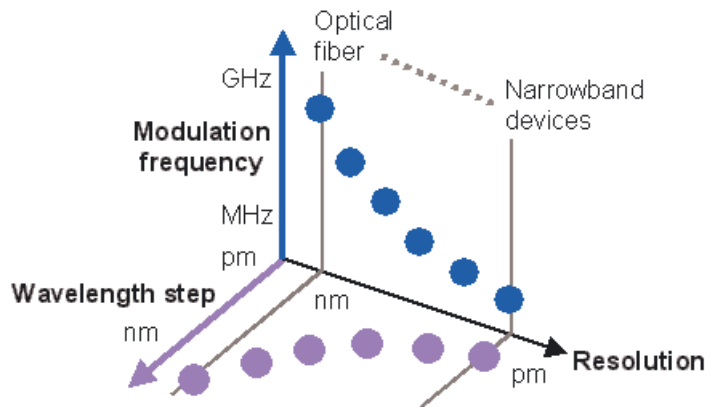


Figure 12 Adjusting wavelength step and mod. freq. to achieve desired resolution

Relationship of setup parameters to imaging

Since the 86038B uses AM modulation, it is actually sampling at the two AM sidebands simultaneously with the carrier wavelength, and reporting the average result. This causes the dependence of resolution on frequency. Again there is an implication for detecting features between the sampled points. Even if the wavelength increment is arbitrarily small, the RF modulation sideband separation must be small compared to the wavelength spacing over which the device under test has variations of interest. If the RF modulation sideband separation is more than the wavelength increment, these two different sample points can cause imaging.

Consider an example of a narrow band-pass filter being measured with a very small wavelength increment and an RF modulation frequency so large that each sideband is stepped through the filter pass band while the other sideband is well rejected. For this example the pass band would appear at two wavelengths, each incorrect in wavelength by half the sideband separation, and each attenuated in amplitude.

For an RF modulation frequency of 1000 MHz, the sideband separation is 2000 MHz, which gives an approximate sideband separation of 16 pm.

The effects of imaging cannot be completely eliminated, but they can be reduced to acceptable levels by choosing a sufficiently small RF modulation frequency.

To achieve a high-resolution measurement of group delay ripple, the modulation frequency should be low enough that the modulation sidebands track up and down the group delay ripple as a pair, as illustrated in Figure 13.

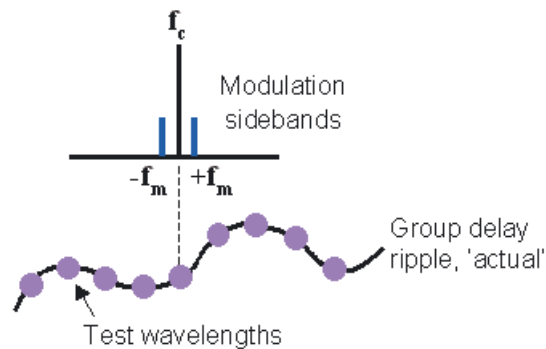


Figure 13 Selecting the modulation frequency for measurement of GD ripple

Relationship of setup parameters to DGD and PDL accuracy

As with GD measurements, increasing the modulation frequency reduces the influence of phase noise on DGD. Because typical DGD values below 10 ps are much smaller than the modulation period, even at the highest modulation frequency, phase wraps are usually not a problem. However at high DGD, the lower and upper sidebands of the modulated signal experience different Jones matrices from each other and from the carrier wavelength and are converted into differing output polarization states. This distorts the signal. Therefore at quite high DGD levels, the modulation should be reduced to keep the sidebands closer together. For example, 2 GHz modulation should only be used up to about 25 ps DGD. Also as with GD, if the DGD must be measured with high wavelength resolution, the frequency should be reduced so that the signal width including the sidebands does not exceed this resolution.

Relationship of setup parameters to ripple amplitude accuracy

Assuming a sufficiently small wavelength increment has been chosen to reveal ripple across wavelength without aliasing, the peak-to-peak amplitude of the ripple for insertion loss or group delay will appear slightly less than the true amplitude.

One attenuation effect is caused by not sampling the phenomena at its extremes. To reduce this effect, use a smaller wavelength increment.

Another attenuation effect is caused when the two AM sidebands produced by the RF modulation frequency stimulate the DUT at different wavelengths, which occur at different points with respect to the ripple phenomena.

This error in ripple amplitude depends on the period of the ripple and the RF modulation frequency. Use smaller RF modulation frequency to reduce this error.

Let us express the period of the ripple by its free spectral range (FSR), which is the optical frequency change across one period of the ripple. For example, ripple with a period of 80pm has a FSR of about 10 GHz.

The case analyzed by [Figure 14](#) is when the sidebands lie equally spaced on either side of a peak of a sinusoidal ripple shape. The ‘% of ps’ curve shows the percent error for GD ripple versus the FSR of the ripple divided by the RF modulation frequency. For small values of amplitude ripple, a similar curve applies. For amplitude ripple of 6 dB peak-to-peak, use the ‘% @ 6dB’ curve to determine the theoretical attenuation of the ripple amplitude.

For example group delay ripple with 10 GHz FSR (80pm period), when measured with a 500 MHz RF modulation frequency, will appear to have a peak-to-peak amplitude with -5% error, or 95% of the true amplitude.

If the RF modulation frequency is larger than one-fourth the FSR, strange effects can appear. For example when $(2 \times \text{ModFreq})/(\text{RippleFSR}) = 0.5, 1.5, 2.5,$ and so on, the ripple may appear to vanish. However, it is interesting to note the ripple FSR will not be aliased to a false ripple rate in this condition (provided the wavelength increment is less than half the ripple period)

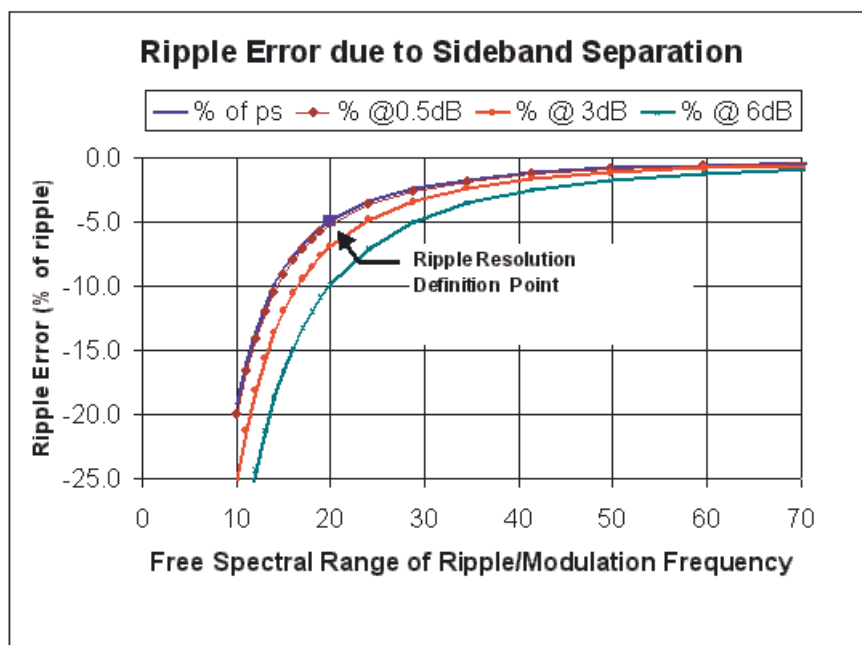


Figure 14 Ripple Error Due to Sideband Separation

The graph shown in Figure 15 is an aid to selecting the wavelength step and modulation frequency for measurement of fine group delay ripple. To use the chart, follow these steps:

- 1 Enter the x-axis at the estimated period of group delay ripple to be measured.
- 2 Draw a vertical line to the curve representing the desired number of measurement samples per period of the group delay ripple.
- 3 Draw a horizontal line to the left and right hand axes. Read the wavelength step and the modulation frequency from the left and right vertical axes, respectively.

To balance the limiting effects of both parameters, the chart in Figure 15 assumes that the double-sided spectral width of the RF modulation spectrum is identical to the wavelength increment. Of course, in some measurement situations it may be appropriate to take smaller wavelength steps than the chart would suggest. For example, to over-sample, then post-smooth the data to reduce trace noise. The 86038B provides built-in smoothing functions for this purpose.

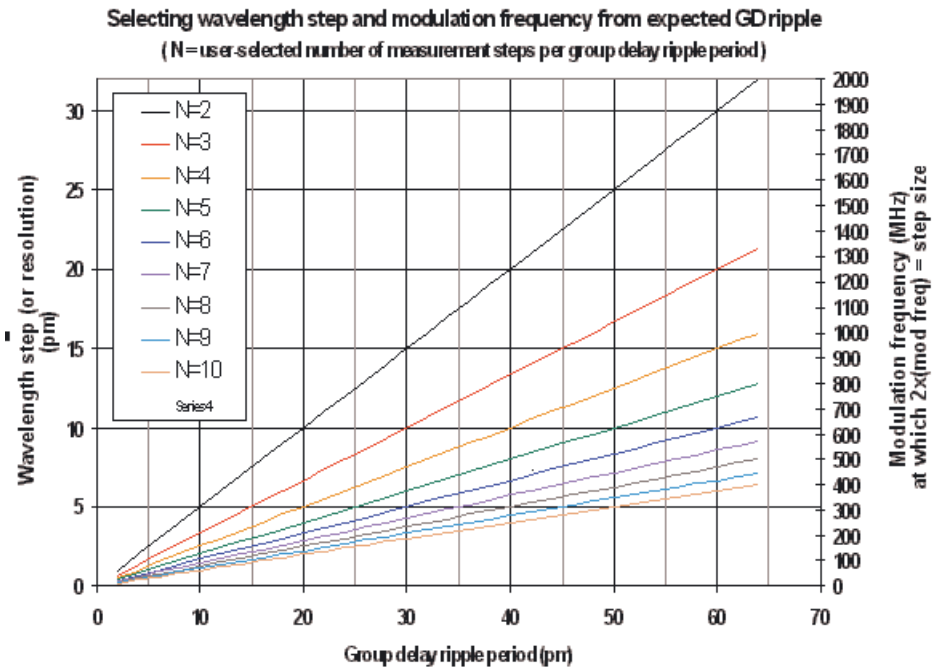


Figure 15 Selecting Wavelength Step and Modulation Frequency for GD Ripple

Control of resolution and noise

When measuring low-dispersion, narrowband devices at high resolution, we face the noise limit implied by Equation 5. Figure 16 provides an intuitive guide to the management of this noise. The brick walls represent limits. The left-right axis of the figure represents the trace noise of the group delay measurement. The upper half of the figure shows the modulation frequency. We reduce into achieve better resolution (movement to the right in the figure), but eventually the trace becomes noisier.

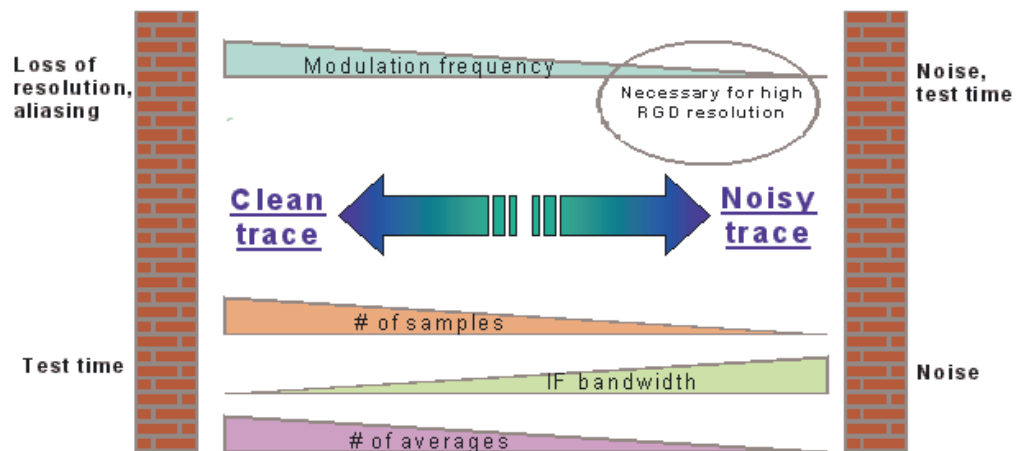



Figure 16 Managing noise in high-resolution measurements

The bottom half of Figure 16 shows three setup parameters that can be adjusted to push our measurement back toward the low noise, high repeatability side of the figure. We can increase the number of samples taken at each wavelength in the stepped mode or the number of points by reducing the wavelength increment in swept mode, achieving a phase noise reduction through the averaging of all of these phase samples. We can also decrease the IF bandwidth. Finally, in extreme cases, we may elect to repeat the measurements and average the traces to reduce noise. These three noise control approaches all lengthen the measurement time.

Selecting setup conditions for a new type of narrowband device

Identifying the best setup conditions is a process that should be performed once before testing a new class of device. Once the appropriate setup conditions are determined, they can be saved and used for all future measurements of devices that have similar characteristics.

The simplest approach is to use the 86038B's Quick Setup feature, which provides two slider-type controls that in turn adjust the setup parameters in a consistent way. You may wish to start by choosing a relatively small wavelength increment and setting the Quick Setup slider controls for less speed but more resolution and sensitivity.



After viewing the measurement results, you may wish to speed up the measurement by taking larger wavelength steps or by moving the slider controls. If phase wrapping occurs, you can identify it by the change in the shape of the group delay curve compared to the earlier conditions.

A second approach to choosing appropriate setup parameters is to check your values using the equations and graphs presented earlier in this section. For example, if the device is expected to exhibit fine group delay ripple of a particular wavelength period, select a wavelength step and modulation frequency from [Figure 15](#).

Suggested reading:

Fiber Optic Test and Measurement (Derickson Ed.)

T.Niemi et al., "Limitations of the phase shift method in measuring dense group delay ripple in fiber Bragg gratings", IEEE Photonics Technology Letters, vol.13, no.12, p.1334, December 2001.

Multiple Port Configurations

The 86038B includes the 8164B Lightwave Measurement System platform. Modular instruments can be installed in the spare slots of this platform. In particular, optional switches can be used to provide multipoint measurement support. The 86038B software supports the use of a 1x2 switch or 1x4 switch to connect the PDLA to multiple input or output ports of a device. For example, the 1X4 switch can be used to allow measurement of 3 or 4 channels of an AWG multiplexer, and to store all of these results in one data file. A 1x2 switch can be added to measure both output ports of an add-drop filter.

Normalization and Measurement

The measurement results of the 86038B are actually the results of two measurements, taken with and without the DUT inserted in the path. A 'normalization' measurement, taken without the DUT in the path, captures the loss and group delay characteristics of the measurement system and test path under the chosen set of measurement parameters. The normalization measurement establishes a loss and group delay baseline for later measurements of the DUT itself.

By saving the results after a normalization, the results can be restored at a later time if they are later replaced by making measurements with different parameters.

Optical Modulator Bias Adjustment

The normalization measurement is accompanied by one or more additional automatic performance optimizations, all under the broad heading of 'normalization'. The 86038B contains an automated modulator bias adjustment function that is performed during a normalization, or by itself through the menu item System > Optimize Optical Modulator Bias. Amplitude modulation in the 86038B is accomplished using a lithium niobate Mach-Zehnder modulator. In this type of modulator, the light is split into two paths which experience different phase shifts in response to a co-propagating RF wave. The two light paths rejoin at the output, where interference between the differentially phase-shifted paths determines the instantaneous optical output level.

The modulator is biased by an adjustable DC voltage in order to select the optimum operating point. The automatic bias adjustment routine fixes the wavelength and the modulation frequency, then measures the variation in modulated optical output while changing the modulator bias voltage. From the resulting curve of output level versus bias voltage, the optimum bias voltage is chosen. For more information refer to [“Optimize Optical Modulator Bias” on page 102](#).

Input Polarizer Adjustment

The 86038B also has an automated adjustment of the input polarizer in the polarization controller. This polarizer should be adjusted for optimal alignment with the polarization of the light from the tunable laser. This alignment can vary somewhat depending on the wavelength range to be measured, and due to environmental changes. The optimization is performed at the start of normalization, or by itself through the menu item System > Optimize Input Polarizer. If the function is activated manually, the position will only be reset if the optimum position has changed by more than 3°. In that case, the current normalization will be invalidated and a new normalization should be made. This function can thus be used to test whether the normalization should be repeated due to environmental changes to the polarization.

Phase Dynamic Accuracy

An optional procedure corrects for phase dynamic accuracy errors. This correction is recommended in most cases where the DUT is expected to exhibit a significant variation of insertion loss or gain over the measurement wavelength range or when high accuracy DGD is desired. This procedure adds about one minute to the normalization routine.

Phase dynamic accuracy (PDA) describes the extent to which a change in the input RF power level causes an apparent (erroneous) change in RF phase. The PDA error is highest at very high and very low RF power levels. The 86038B reduces the impact of the PDA error by means of a user selectable calibration routine that runs as part of the normalization process. The routine sets the modulation frequency and the wavelength, then steps the TLS optical output level (using attenuation) in increments from low to full output power. At each step, the RF phase is measured and associated with the RF input level. This data is used to correct future measurements.

The PDA data are associated with a particular modulator frequency and will not be used if the frequency is changed.

Since changing the wavelength range can change the maximum power level, this may cause PDA data to be cleared. PDA correction should not be used to measure amplifying devices if the output power will exceed the maximum available from the laser instrument.

Length Normalization

If a Length measurement is specified with the Measure mode, the Normalization path's length is measured so that it can be subtracted out.

$$\text{DUT Length} = \text{Total Length} - \text{Normalized Path Length.}$$

Refer to [“Length” on page 106](#) to setup a length measurement.

Wavelength Accuracy Correction

Using the optional 86122A Multi-Wavelength Meter, wavelength correction is performed for stepped normalizations, when a stepped measurement is chosen.

Wavelength correction is then always performed if an 86122A is connected. Five wavelength points are measured and the average offset is applied. For more information, refer to [“Normalization Range” on page 107](#) and to [“Begin Normalization” on page 115](#).

Swept and Stepped Measurement Modes

The measurements performed by the 86038B involve tuning the laser source while observing changes in the recovered RF amplitude and phase. The wavelength can be changed in either a stepped or a swept (continuous) mode. The 86038B includes advances to optimize the performance with continuously swept wavelength measurements, which greatly reduce measurement time and facilitate the use of small wavelength increments for high wavelength resolution applications. The time reduction also reduces the sensitivity of GD/CD measurements to GD drift in the device, such as a long fiber with changing temperature.

However the stepped mode may be preferred for measurements at a limited set of wavelengths, or to allow averaging many measurement samples to minimize noise. This mode also minimizes the elapsed time involved in individual DGD measurement points and thus minimizes the influence of GD drift on the result, which can be important for long fibers with low PMD. The stepped measurements may also be desirable in some cases because each measurement point is made at a fixed wavelength, rather than accumulated over the increment.



3

Making Measurements

Introduction.....	80
Quick Setup Measurement Process	81
Flexible Measurement Process	82
Setup Parameters and What They Do.....	83
Choosing Setup Parameters for Your Device.....	85

Introduction

NOTE

To get the most performance out of your ODA, it is recommended that you read [Chapter 2, “Measurement Concepts”](#), [Chapter 3, “Making Measurements”](#), and [Chapter 4, “Function Reference”](#). These three chapters cover the linkages between the different measurements that can be made, specific devices being measured, and the setup parameters used to make the measurements.

Quick Setup Measurement Process

Set up the Normalization

- 1 Connect the patch cord for the normalization between the optical ports of the PDLA (which may include switch connections).
- 2 Check the connections with the Real Time Port Monitor. Refer to [“Real-Time Port Monitor” on page 101](#).

Quick Setup Process

- 1 On the front panel, press Quick Setup.
- 2 Select port configuration, if switch is installed.
- 3 Select laser for desired wavelengths if more than one TLS is attached. The TLS Output 2 is standard and usually recommended but the low-SSE Output 1 can be configured if needed.
- 4 Choose which attributes to measure. Select stepped or swept mode.
- 5 If desired, select advanced features: Six Polarization States, generation of matrix data, or use of external power meters if Option #400 is installed.
- 6 Set the measurement wavelength range.
- 7 Set the resolution slider (lowest to highest resolution).
- 8 Set the Sensitivity Slider (low to high sensitivity) considering that higher sensitivity requires more time.
- 9 Check the box for Drift Correction when measuring the DGD/PDL of long fibers.
- 10 Begin the Normalization

Measurement Process

- 1 Connect the DUT.
- 2 Begin the Measurement.
- 3 View the data and manipulate the display and markers.
- 4 Smooth or curve fit the data as desired.
- 5 Save the measurement and/or export the data.

Flexible Measurement Process

The following procedure outlines the steps to make a precisely defined measurement. For experienced users, this process provides more flexibility than the Quick Setup procedure.

Prepare for Normalization Measurement

- 1 Connect the patch cords needed for the normalization between the optical ports of the PDLA (which may include switch connections).
- 2 Check the connections with the Real Time Port Monitor. Refer to [“Real-Time Port Monitor” on page 101](#).

Setup and Normalization Process

- 1 Select port configuration if a switch is installed
- 2 Select Laser if more than one are installed, based on wavelength range.
- 3 Choose which attributes to measure. Select stepped or swept mode.
- 4 Enable the Length measurement if needed.
- 5 Set the measurement wavelength range.
- 6 Set the normalization wavelength range (if different from the measurement range).
- 7 Set the wavelength increment, modulation frequency, IF bandwidth, and Drift Correction.
- 8 Begin the Normalization, after selecting the port or ports for normalization and whether to use PDA.

Measurement Process

- 1 Connect the DUT.
- 2 Set Trigger to Single or Repeat Sweep.
- 3 Begin the Measurement.
- 4 View the data and manipulate the display and markers.
- 5 Smooth or curve fit the data as desired.
- 6 Save the measurement and/or export the data.

Setup Parameters and What They Do

A family of setup parameters allow the user to optimize 86038B performance for a wide variety of measurement applications. Following is a brief description of the setup parameters. For a complete description of each parameter, refer to “[Function Reference](#)” on page 93.

Wavelength range

The range of wavelengths over which the measurements are taken, usually entered by means of start and stop wavelengths.

Wavelength increment

The wavelength separation of adjacent measurement samples. Together with the wavelength range, this determines the number of points in the spectrum. The maximum number is 16001.

Modulation frequency

The frequency of the AM modulation impressed on the optical signal by the modulator inside the optical test set. The modulation frequency helps determine wavelength resolution and sensitivity.

Samples/Point

The number of times the recovered RF amplitude and phase are measured at each measurement wavelength (trace point). In the swept mode only a single sample is taken. In the stepped mode, the number of samples may be selected over a wide range. The samples are averaged to reduce the effect of instrumentation noise.

Fewer samples will provide a faster measurement at the expense of larger variability (noise)..

IF Bandwidth

This control sets the bandwidth of noise-reducing filter circuitry and algorithms in the RF network analyzer. A lower value of IF Bandwidth provides better noise rejection but also increases measurement time. For the same impact on measurement time, reducing IF bandwidth is more effective in suppressing noise than increasing the samples/point.

TLS Power

This control sets the TLS output power limit. The default value of +5 dBm is available from the 81600B-160 tunable laser source, for example, from 1520 nm to 1610 nm. For other tunable laser sources refer to the individual TLS specification. At higher TLS power levels, Amplitude Dynamic Range and Group Delay Noise is improved, but phase dynamic accuracy (PDA) effects are more pronounced. After changing the TLS power, normalization (without PDA) is required.

The 86038B will automatically choose the highest TLS power available over the chosen wavelength range, but the power limit can be used to reduce the chosen power, for example to avoid power changes when the wavelength range is changed. It can also be used when measuring amplifying devices to avoid exceeding the safe power to the optical receiver.

Choosing Setup Parameters for Your Device

A list of characteristics, suggestions and relationships are provided for a variety of device types. For more information on the measurements that you can make, refer to [“Measurement Concepts” on page 49](#). For more information on the instrument functions, refer to [“Function Reference” on page 93](#).

Transmission Fiber (All Types)

Characteristics

- Large range of dispersion values across the wavelength range
- Wide wavelength range of interest
- Smoothly changing group delay with relatively little fine structure with respect to wavelength
- Group delay is easy to curve fit using conventional model curves
- Differential group delay (DGD) may exhibit multiple peaks and valleys
- Approximately 0.25 dB/km attenuation

Measurement Tips

The relatively smooth and analytic behavior of the group delay of this device makes it possible to use larger wavelength increments and higher modulation frequency without concern for loss of wavelength resolution.

A good choice is 2 GHz, as used for the specifications.

Wavelength increment may be large, like 0.5 nm or more, except if a high CD level requires closer steps to avoid phase wrapping.

If the PMD is higher than about 10ps, then steps smaller than 0.5 nm should also be chosen for the DGD measurement, because the rate at which the DGD of a fiber changes with wavelength is proportional to the PMD.

It is also useful for the PMD result to measure over a wide wavelength range, even if the fiber is only used in a more limited range, because the wider range will give a better average value.

Since the number of points does not need to be too high, the time taken by using a lower IF bandwidth like the 70Hz in the specifications should be acceptable. But a higher rate can be used if the noise is still acceptable, especially when the fiber attenuation is not too high.

Alternately, or when a high CD necessitates it, a higher density of points can be used, together with a higher IF bandwidth to maintain measurement speed, and the GD results can be smoothed to the desired resolution to reduce the noise, especially for the CD calculation.

Since the absolute GD of long fibers can change substantially with small temperature changes, the measurement time for long fibers (tens of km) should not be set too high, and the Drift Correction functionality should be used to reduce drift sensitivity.

Since the CD uncertainty increases for small values of wavelength resolution (high resolution), and high resolution is usually not needed for fiber, the resolution should be kept down by either using wide wavelength increment or smoothing the data to a wide resolution. It is recommended to smooth the GD spectrum and then determine the CD from this, but smoothing the CD curve itself or a combination of the two gives similar results.

Fitting the data results in even lower noise, effectively basing each CD value on all of the points in the fit wavelength range, with the accuracy dependent on the validity of the chosen fit type.

Fiber Dispersion Compensating Module (DCM)

Characteristics

- High dispersion (steep change in group delay versus wavelength).
- Wide wavelength range of interest
- Significant loss
- The zero dispersion wavelength falls far outside the 1500 nm window.

Measurement Tips

The tips for transmission fiber also apply here. But the high dispersion of many of the devices will require either smaller wavelength increment or lower modulation frequency.

Considering the higher insertion loss often associated with these devices, it may be best to maintain the higher modulation frequency like 2 GHz to minimize noise. In this case a smaller increment is needed, but since the wavelength resolution is not required, the data can be smoothed to an effectively wider resolution. To keep the measurement time low, with the high number of points, swept measurement mode is usually recommended. The IF bandwidth should be chosen to achieve both acceptable noise and measurement time.

Often these devices are designed for use in a particular wavelength range, like the C-band.

However for PMD measurement of devices with low PMD, it is advised to measure over a wider range to improve the average value.

If stepped measurements are used, the following tips also apply:

- For higher loss fibers, improve signal-to-noise by using a narrower IF bandwidth and a larger number of samples.
- Especially when measuring in step mode select a modulation frequency and wavelength step appropriate to the expected level of dispersion to eliminate the possibility of RF phase wrap errors.
- When measuring high dispersion fibers in step mode with manual modulation frequency selection, the use of 2000 MHz modulation frequency may require the selection of a low wavelength increment to avoid aliasing. This increases test time. To reduce test time, lower the modulation frequency and increase the wavelength increment.

Fiber Bragg Grating Dispersion Compensators (or other channel compensators)

Characteristics

- High dispersion (steep change in group delay versus wavelength)
- Often a narrow wavelength range or set of narrow channels to be measured
- Group delay ripple (fine structure in the group delay curve)

- PDL and DGD may be present due to polarization dependence of the filter wavelength or if a circulator is included

Measurement Tips

- If the group delay ripple is to be measured, it is necessary to select a small enough modulation frequency and wavelength interval to resolve the ripple. Since too wide smoothing will also hide the ripple, it is better to use a low IF bandwidth if necessary to reduce the noise.
- The swept mode offers very small wavelength intervals and high measurement speed. The step mode may take an unacceptable amount of time if the wavelength step is very small and the wavelength range is large.
- When measuring in step mode in cases where resolving group delay ripple is not required, select a modulation frequency and wavelength step appropriate to the maximum expected level of dispersion.
- The use of curve fits is likely to be useful in this application. The markers can be used to limit the fit to the relevant wavelength range. The deviation spectrum, the difference between the measured data and the fit, can be displayed to show the ripple.
- The component may also be characterized in terms of optical phase ripple. Since the phase increases linearly with optical frequency when there is no dispersion, the deviation from a quadratic curve fit with respect to frequency, over the channel width, is a good basis for determining ripple when a fixed channel dispersion is desired.

Erbium Doped Fiber (Unpumped)

Characteristics

- High loss (up to 10's of dB's)
- Low dispersion (very slight change in group delay over wavelength)
- Wide wavelength range of interest
- Little if any fine group delay structure

Measurement Tips

- The high loss of this device degrades the signal to noise, making it necessary to combat the noise with various setup conditions. For best signal to noise, select 2000 MHz modulation frequency.
- Improve signal to noise by using a narrower IF bandwidth (70 Hz or even 30 Hz) and a larger number of samples.
- Reduce trace noise by means of the built-in smoothing functions.
- Swept mode gathers a large number of points to smooth, taking advantage of the smoothing function. However, the step mode allows taking and averaging a large number of samples at each wavelength, also reducing the noise. Try both modes to find the best trade-off of accuracy and measurement speed.

Erbium Doped Fiber (Pumped)

Characteristics

- High gain (20 to 30 dB)
- High power output (>20 dBm)
- Amplified spontaneous emission (ASE) ‘noise’
- Often a broadband device
- Low dispersion (very slight change in group delay over wavelength)
- PMD is usually specified and may come from isolator

Measurement Tips

- Insert attenuation following the EDFA to avoid overdriving the input of the 86038B. Refer to “[Conditions](#)” on page 342 for input power limitations.
- Improve signal to noise by using a narrower IF bandwidth (70 Hz or even 30 Hz). To get even a lower noise, the number of samples can also be increased.
- Reduce trace noise by means of the built-in smoothing functions.

- Swept mode gathers a large number of points to smooth, taking advantage of the smoothing function. However, the step mode allows taking and averaging a large number of samples at each wavelength, also reducing the noise. Try both modes to find the best trade-off of accuracy and measurement speed.

Narrowband Filter or DWDM Wavelength Routing Device

Characteristics

- Very narrow wavelength range or ranges
- Rapid changes of measurement parameters with wavelength
- Dispersion and PDL often associated with filter edges
- Fine in-band structure of some attributes, possibly including group delay ripple.
- Steep filter slopes
- High rejection outside the passband

Measurement Tips

The combination of narrow passband and fine group delay structure requires high wavelength resolution measurements

- Swept mode should be used to measure many points with small increment, like 10 pm, in a reasonable time, including the polarization dependence. The high wavelength resolution also usually require using a small modulation frequency. Try 500 MHz. To reduce noise, the low IF bandwidths, like 30 Hz may be required. Further noise reduction could be obtained by using a smaller increment, like 1 pm and then Gaussian smoothing to a 10 pm resolution.
- Select the Phase Dynamic Accuracy (PDA) correction when performing a swept or stepped normalization for filter testing, if GD at high loss is also of interest. This correction reduces the degree to which the large variations in device loss with wavelength are translated into small errors in group delay.
- Amplitude ripple and GD ripple can be analyzed by using an appropriate Curve Fit to the data and then displaying the “deviation”.

Broadband Component (Isolator, Circulator, Coupler, Attenuator and so on)

Characteristics

- Wide wavelength measurement range
- Usually low CD and PMD, but especially PMD may need to be verified
- Loss may be high, but relatively independent of wavelength

Measurement Tips

The broadband nature of these devices allows you to trade off wavelength resolution for faster measurement speed.

- Set the modulation frequency in manual mode to 2000 MHz.
- In either swept or stepped mode, optimize measurement speed by setting the wavelength increment just small enough to resolve the wavelength dependencies of the attributes being measured.
- If you use the smoothing function to reduce trace noise, you may need to increase the number of trace points. (for example, reduce the wavelength increment) to avoid loss of resolution.



4

Function Reference

Introduction.....	94
File Menu.....	95
View Menu	97
System Menu	98
Setup Menu.....	104
Measure Menu.....	115
Display Menu	119
Navigating the Front Panel.....	129
Toolbar Functions.....	130

Introduction

The following section provides descriptions of each software function. The functions are listed in the order that they appear in the Instrument's menu bar. The same functions are accessible from the front panel. Refer to [“Making Measurements” on page 79](#) for complete information on how to make timely, accurate measurements on your device under test (DUT).

This section assumes that you have a working knowledge of an MS Windows®-based computer and its operating conventions. It also assumes you know how to open, save, and close files. For help with any of these techniques, please see your MS Windows® documentation.

NOTE

To get the most performance out of your ODA, it is recommended that you read [Chapter 2, “Measurement Concepts”](#), [Chapter 3, “Making Measurements”](#), and [Chapter 4, “Function Reference”](#). These three chapters cover the linkages between the different measurements that can be made, specific devices being measured, and the setup parameters used to make the measurements.

File Menu

The File menu is used to save, recall, and print data.

Recall

Retrieves all previously saved instrument state, trace data, and normalization data.

Save

Saves the current instrument settings, trace data points, and normalization data. The file is saved with a .oda extension. When this file is recalled using File >Recall, all the setup parameters will be set to the saved conditions.

Selecting **Additional Data export** or **Additional Mueller/PMD export** will also cause the corresponding .csv or .pme file to be exported with the same file name. See Export below.

Export

Saves a file for use with external applications. There are six supported export file formats:

.csv	ASCII text file of comma separated values. This type of files includes measurement results and settings in a format that can be viewed easily with a spreadsheet or analysis packages such as Excel or MatLab.
.jpg	Image file format that contains a screen capture of the ODA application. The benefit of this format is that the file size is small, however the image quality is reduced.
.bmp	Image file format that contains a screen capture of the ODA application. The benefit of this format is that the image quality is high, however the file size is very large.
.tif	Image file format that contains a screen capture of the ODA application. The benefit of this format is that the file size is small and the image quality is high. This format is not supported by some applications.
.log	ASCII text file used to save the PDLA Status Log. This can be helpful to document any problems that occur and to facilitate service.
.pme	ASCII text file of comma separated values. This exports Mueller Matrix and PMD Vector data, if generation of this data was selected in the Measurement Type Setup panel at the time of measurement. The data consists of 9 columns with the form: wavelength (nm), m11, m12, m13, m14, GDavg, PMD1, PMD2, PMD3, where m11-m14 is the top row of the Mueller Matrix at the given wavelength and (PMD1, PMD2, PMD3) is the PMD vector.

Print

Prints a copy of the graphical display and pertinent instrument settings. Use the parallel port on the rear panel of the PNA to connect to your printer.

Exit

Exits the PDLA application. This is usually only necessary for performing administration of the computer. Since exiting the program and restarting it results in restarting the 90 minute warm-up cycle, it is usually preferable to minimize the application.



View Menu

The View menu is used to access the desktop screen.

Minimize Application

Minimizes the PDLA application window.

System Menu

Laser Off

Turns the laser off at the TLS output. The laser will turn back on when a measurement is started (Begin Meas, Begin Norm, Begin Mod Bias, or Begin Length).

Laser On

Turns the laser on at the TLS output.

Local

Returns local front-panel control. Remote programming can prevent the PDLA from responding to front-panel key presses. Pressing Local will reestablish front-panel control.

Preset

Cancels any current operations and sets the instrument to pre-defined setup condition (state). Preset also causes the GPIB bus to be searched for new instruments.

Table 4-2. Default values

Function	Preset Value	Function	Preset Value
Dispersion Measurement Type	GD/CD Swept	X-Axis Scale, Minimum ^a	1493 nm
Length	Specify 1000m	X-Axis Scale, Maximum ^a	1640 nm
Active Ports	Port1	X-Axis Scale, Center ^a	1566,5 nm
Measurement Start Wavelength ^a	1493 nm	X-Axis Scale, Span ^a	147 nm
Measurement Stop Wavelength ^a	1640 nm	Y-Axis Scale, Reference Level ^b	0 ps
Normalization Range	Use Measurement Settings	Y-Axis Scale, Ref. Level Position	5
Resolution, Increment ^a	0.5 nm	Auto Scale During Measurement	On
RF Modulation Frequency	2000 MHz	Smoothing	Off
Sensitivity, IF Bandwidth	300 Hz	Curve Fit	Off
Sensitivity, Samples/Point	1	TLS Power	+ 5 dBm
Drift Correction	Off	Markers	Off

^a for 81600B-160 TLS: depends on TLS type

^b Varies by trace type.

Preset Display Properties

Resets all the settings contained in the Display menu to their default values. It differs from a normal preset in that measurement settings and trace data are preserved.

Select TLS

Selects the TLS for measurement, if more than one are attached to the GPIB. Selecting "Wait for first TLS only" allows the initialization procedure to complete after one TLS is recognized, to avoid waiting for the other TLS modules to boot.

Wavelength Meter

Uses the 86122A multi-wavelength meter to determine a wavelength offset correction for the TLS during the Normalization process. To use this function, the Low SSE output of the TLS must be connected to the 86122A during normalization. The wavelength meter is used only during normalization and not before or during actual measurements.

Status Log

Opens the Status Log window and displays a history of all instrument errors and messages with the time the event occurred. The Status Log window often shows additional explanation details for messages and warnings that appear in the User Interface window.

Clear removes the messages from the status log.

Close closes the Status Log window.

Export Status Log opens the Export window to save the status log.

Configuration

The status of the internal thermal regulator is displayed in the lower right of the display with a thermometer icon. State Red indicates the regulator is out of regulation because the environment is too hot or too cold. State Yellow indicates internal temperature is waiting for a timed delay to reach GD stability, or temperature change is large in the past 2.5 minutes. State Green indicates stable temperature, ready for normalization or measurement.

When the PDLA power is cycled, the internal thermal regulator will automatically come on, choose an initial temperature set point, and provide the state Yellow timed delay for 80 minutes to reach state Green. (When calibrated measurements are possible.)

If the environment has a significant temperature change over the past 2.5 minutes, state Yellow will be displayed. When the environment becomes stable again, state Green will be displayed without delay.

New Heater Set Point

If the environment changes more than ± 6 C, from the turn-on state, the icon may turn Red indicating thermal regulation is no longer possible. To reset the thermal regulator to the present environment, right click on the thermometer icon and left click on the prompt box. After resetting the thermal regulator, the icon will show state Yellow for 40 to 80 minutes as the system comes back to thermal equilibrium.

Tip: Hover the mouse cursor over the icon for a thermal regulator status report.

Show Low Power Warnings

When enabled, the instrument will place warning messages in the status log when the detected power from the DUT path is too low. This is helpful for detecting problems, but can be distracting for devices like filters that have very high attenuation at some wavelengths. If low power warnings are disabled, the system will continue to display warnings if the

internal monitor power is too low. These warnings either indicate a need for optimization of the modulator bias or input polarization, a problem in the optical connection to the tunable laser, or an internal optical problem. Please refer to the Troubleshooting chapter for further information.

Automatic Canceling

If TLS wavelength instability is detected in a sweep, and a repetition of the sweep is again unstable, the measurement is canceled and a message appears in the log. In this case, it may be necessary to reduce the sweep range for the measurement. Deactivating this function allows the measurement to complete, but care should be taken in using the results.

Real-Time Port Monitor

Displays on a horizontally scrolling graph the real-time gain/loss and group delay measurements at the DUT Port receiver and monitor receiver. The Port Monitor operates in repeat sweep trigger mode at fixed wavelength and uses a 2 GHz modulation frequency and 30 Hz IF bandwidth. The wavelength can be set in the wavelength text box, generally to the center of the measurement wavelength range, or to the pass-band in the case of a wavelength-filtering DUT.

The real-time monitor can be used to check the stability of the test system, test device, and interconnecting cables before performing an actual measurement. This is especially useful when measuring small relative group delay values approaching or below 50 fs and for long-fiber measurements, for which temperature instability causes substantial GD variation. After using the Real-Time Port Monitor to verify that the DUT connections are good and test environment is stable, the setup is ready for a measurement. The stability of the setup could appear different for the DUT than for the normalization due to actual temperature changes, longer fiber lengths, insertion loss, mechanical movements, air currents, and so on.

Clear Plots resets the measurements and clears the graph data.

Port buttons allow selecting the Port configuration if an integrated switch is installed.

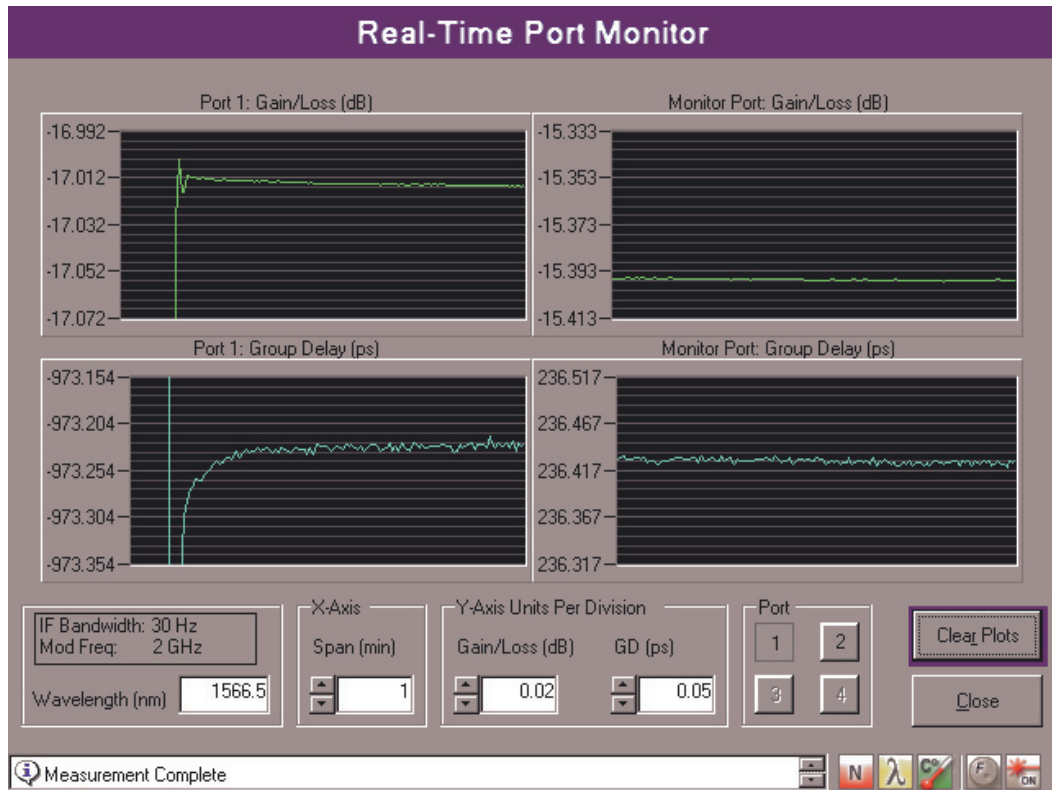


Figure 17 Real-Time Port Monitor

Optimize Optical Modulator Bias

Allows manual activation of the procedure to find the optimum modulator bias, performed at the center of the current measurement wavelength range. During normal measurement procedures, this is activated automatically as needed during normalization measurements. Manual activation can be used for diagnostic purposes, especially if the setup seems to deliver insufficient optical power. In that case, this optimization can be applied before using the Real-Time Port Monitor. Manual use will cause the current normalization measurement to become invalid. (A normalization measurement can be saved to reload after testing the change of parameters.)

Optimize Input Polarizer

Allows manual activation of the procedure to find the optimum input polarizer position, performed at the center of the current measurement wavelength range. This procedure aligns the polarizer with the input polarization from the tunable laser. During normal measurement procedures, this is activated automatically as needed during normalization measurements. Manual activation can be used for diagnostic purposes, especially if the setup seems to deliver insufficient optical power. In that case, this optimization can be applied before using the Real-Time Port Monitor. Manual use will cause the current normalization measurement to become invalid, if a new position is selected. If the position found is within 3° of the previous setting, the old setting and normalization data will be maintained. (A normalization measurement can be saved to reload after testing the change of parameters.)

Toggle mainframe local-lockout status

This command can be used to allow or lock out manual "local" access to the 8164B display and user interface. When local access is locked out, the 86038B software expects exclusive control of the instruments and does not redundantly resend the settings, which allows faster measurements. Disabling the display also helps prevent accidental interruption of system operations from front panel buttons. When local access is allowed, pressing "Local" causes the instrument display to be shown and front panel buttons to be activated.

Zero external power meters

If power meter modules are installed in the 8164B mainframe for Option 400, then this command activates the Zero correction function of the modules to remove electrical offsets. This should be used after the instruments are warmed up and before starting making measurements requiring high dynamic range. The tunable laser will be deactivated during this measurement. Make sure that no other light sources are entering the power meters. Details to this function are found in the Lightwave Measurement System User's Guide.

Setup Menu

Quick Setup

Accesses a series of control panels that help you specify the basic settings needed for making a measurement. A measurement time estimation is provided so that you can trade off measurement time versus the measurement settings.

CD/PMD

Allows you to specify the measurements and measurement mode (swept or stepped).

- GD/CD Swept
- GD/CD + DGD/PDL Swept
- GD/CD Stepped
- GD/CD + DGD/PDL Stepped

In Stepped mode, a number of phase and amplitude measurements are performed at each wavelength. The wavelength resolution is set by the wavelength step size and the modulation frequency. Refer to [“Sensitivity/Resolution” on page 108](#). Stepped measurements including DGD/PDL are performed by repeating the number of samples for each wavelength at 4 or 6 states of polarization.

In Swept mode, the phase and amplitude are sampled periodically as the wavelength is continuously swept. This yields a single sample of both phase and amplitude at a series of wavelength values. The density of samples is set by the wavelength increment. Swept mode can dramatically reduce measurement time. The sweep speed is chosen automatically, depending on the wavelength increment and the IF bandwidth. Swept measurements including DGD/PDL are made by repeating the wavelength sweep at 4 or 6 states of polarization.

Use 6 Pol.-states

Select this to determine DGD and PDL from matrix analysis based on measurements at 6 polarization states, instead of 4 states. Using 6 polarization states reduces the sensitivity of the matrix analysis to the uncertainty of the polarization state settings. This is useful to reduce the effect of wavelength dependence of the polarization states when measuring over a

wide wavelength range or at short wavelengths like 1300nm, as well as to reduce the influence of PDL on the DGD measurements. The extra sweeps increase the measurement time, but the extra sweeps also reduce noise so the IF bandwidth can be reduced to make faster sweeps.

Generate Mueller Matrix and PMD Vector data

Selecting this option causes the Mueller Matrix and PMD Vector data to be kept in memory, together with the parameters like PDL and DGD that are derived from them. The matrix data are then saved together with other data and settings by the "Save" function and can be exported to a comma-separated variable text file by the "Export" function. Selecting this option increases the .oda file size.

External Power Meter

When Option #400 is installed, this field is enabled. Selecting "Use External Power meter" causes Gain/Loss, PDL and Mueller Matrix data to be based on measurements with the Power Sensor modules instead of the standard Optical Receiver. This provides enhanced IL and PDL accuracy specifications. Indicating the correct expected "DUT Min Loss/Max Gain" allows the instrument to determine the maximum expected power to the power meter and to choose the optimum power range for the measurements.

NOTE

In swept mode testing of a very long fiber (that is, thousands of km), there is a possibility of a slight wavelength offset caused by the delay through the fiber, depending on the sweep speed used..

Tips on Measurement Modes:

- Swept mode is faster and the 86038B has been designed for optimized swept performance. But stepped mode can be used to achieve minimum noise by averaging many samples. Stepped mode also allows measurement of the lowest DGD or PMD values in long fibers sensitive to temperature drift, by minimizing the time between polarization state measurements.

- In the swept mode, over-sampling the GD shape with small wavelength increment offers the possibility of using smoothing to reduce the noise. The 86038B automatically provides Gaussian smoothing of the data to the resolution limit set by the modulation frequency.
- Stepped mode can greatly improve signal to noise by allowing the averaging of multiple samples at each measurement wavelength.
- For more information on measurement types, refer to [“Swept and Stepped Measurement Modes”](#) on page 77.

Length

Length Mode

Measure Determines the length of the test device based on network-analyzer group delay measurements and a user-entered refractive index value. Index values of 1 to 10 can be chosen. This feature is intended especially for fiber and cable test applications, but sub-millimeter accuracy can be achieved on short devices. The procedure measures and stores the absolute group delay of the test device at the chosen wavelength. The displayed length in meters is based on the user-entered refractive index, which can also be changed after the measurement.

Specify Permits user specification of the length of the test device in meters.

Ports

Permits specification of the ports to be used for measurement. Multiple ports are available when an optional switch is integrated. The port numbers are then associated with the corresponding switch position. When no switch is included, only the default Port 1 is available.

Measurement Range

The start and stop wavelength settings define the x-axis limits for both stepped and swept measurement.

Tip: For best PMD accuracy of fiber measurements, measure over as wide a wavelength range as possible. The zero dispersion wavelength is most accurately measured when it falls near the middle of the measurement wavelength range. Accuracy is slightly degraded if it falls near the edge of the wavelength range, and is significantly degraded if it falls far outside.

Wavelengths can be set within the range reachable by the currently chosen tunable laser. However some combinations of wide wavelength increment and high IF bandwidth may result in a small shortening of the laser wavelength range available for measurements at the highest sweep speed. In this case the corresponding points of the traces will not be measured.

Swept measurements over long wavelength ranges or at wavelengths where the laser has low power may include the effects of a mode hop, where the laser can make a small jump in wavelength. The wavelength is accurately measured at the point where data samples are triggered, but some error can be introduced because the swept samples are integrated over part of the wavelength interval between the triggers. This can sometimes be seen as isolated spikes, especially in the DGD traces. See the tunable laser specifications for the conditions for mode-hop free continuous sweeps.

Start (nm)

Sets the start wavelength. This setting is independent of the settings specified for the X Scale Display which affect the viewable range only.

Stop (nm)

Sets the stop wavelength. This setting is independent of the settings specified for the X Scale Display which affect the viewable range only.

Copy From Selected Graph

Sets the start and stop wavelength to the displayed endpoints of the selected graph.

Normalization Range

Specifies the wavelength range and the wavelength increment for the normalization process.

Tip: Choose a normalized wavelength range that is equal to or greater than the measurement wavelength range.

Use Measurement Settings

Uses the measurement settings specified for start and stop wavelength, in the Meas Range setup panel, and Trace Points and Increment settings specified in the Sensitivity/Resolution setup panel.

Customize

Allows you to specify Start and Stop Wavelength, Trace Points, and Increment independent from the settings specified for the measurement.

Begin Normalization shortcut accesses the Begin Normalization panel.

For more information on performing a normalization, refer to [“Begin Normalization” on page 115](#).

Sensitivity/Resolution

Resolution Area

Increment Specifies the separation between adjacent measurement wavelengths. The minimum increment is 0.0001 nm.

Trace Points Shows the number of measurement wavelengths including the start and stop wavelengths. The value is determined from the chosen measurement range and increment. The 86038B can measure spectra with up to 16001 points.

RF Mod Freq Specifies the RF modulation frequency used to modulate the optical signal in the PDLA test set.

In general, a higher modulation frequency reduces instrumentation noise and a lower modulation frequency improves wavelength resolution.

Max CD range Based on the chosen wavelength interval and modulation frequency, the maximum measurable dispersion (positive or negative) is displayed. This can be compared to the expected CD when choosing these parameters. Measurement of a fiber or component with CD higher than this maximum may have errors due to phase wrapping.

Tips on the Modulation Frequency Selection

- Use the **Max CD range** display or the selection chart in Chapter 2 to choose a combination of wavelength increment and modulation frequency that will avoid phase wrapping errors and aliasing errors. For example, when measuring high dispersion fibers, use a small wavelength increment to avoid aliasing. This increases test time. To reduce test time, lower the modulation frequency and increase the wavelength increment.
- Changing the modulation frequency between normalization and device measurements causes a loss/gain offset due to the frequency response details of the optical receiver and the network analyzer RF circuitry. This point is true even when PDA correction is not selected. If the frequency is changed after the normalization measurement, the normalization will become invalid. In this case, DGD/PDL measurements will not be performed before the normalization is redone. Simple GD/CD measurements will be performed, to allow quick assessment of different parameter sets, but the results will not be normalized.
- The shape of the PDA error changes significantly for different modulation frequencies, so, if the modulation frequency is changed, a normalization with PDA should be performed. For more information on PDA, refer to “[Phase Dynamic Accuracy](#)” on page 115.
- A good rule of thumb is to use a modulation frequency that is less than 1/20 of the free spectral range of any ripple to be measured. For example, a 500 MHz modulation frequency could be used to explore ripple with 10 GHz FSR (or about an 80 pm period at 1550nm) with an attenuation of about 5% of the ripple peak-to-peak value.

Sensitivity Area

IF Bandwidth Sets the bandwidth of the signal filtering, trading off noise reduction (sensitivity) against measurement speed. A lower IF bandwidth value means lower noise (high sensitivity) and a longer measurement time. For swept-mode measurements, the 86038B chooses a laser sweep speed based on the IF bandwidth and the wavelength interval, so that the samples are integrated over most of the interval. Settings of long intervals and high IF bandwidth may correspond to higher sweep speeds than achievable. In this case the samples will only be integrated over a short part of the interval, which may result in undersampling. Reducing the IF bandwidth can then reduce noise without resulting in slower sweep rates. Reduce the IF bandwidth as needed to control trace noise. This is particularly important when measuring a high insertion loss device. See [“Smooth”](#).

Tips on IF Bandwidth Selection For higher loss (typically longer) fibers, improve signal to noise by using a narrower IF bandwidth. Expect to trade-off measurement speed.

Duty Factor

For swept mode measurements only! As mentioned above, the laser sweep speed depends on the wavelength interval and IF Bandwidth parameter settings. Since the speed can only be set to particular values, some combinations of parameter settings are able to produce a better match between the integration interval and the wavelength interval. The estimated ratio of these two is displayed as the Duty Factor. If this value is low, such as below 50%, then reducing the IFBW can improve the duty factor and noise level without increasing the measurement time. Use this display to optimize the parameters.

Samples/Point

For Stepped mode measurements only! Determines the number of amplitude and phase measurements (samples) that are taken at each measurement wavelength (trace point). If polarization resolved measurements for DGD/PDL are chosen, then 4 (or 6) measurements are made for each sample.

In the Swept measurement mode the samples per point is always 1. In Stepped measurement modes, you select how many samples to take at each measurement wavelength. The multiple samples are averaged to generate a single trace point. A larger number of points means more averaging, suitable for reducing trace noise in cases of high insertion loss test devices. Note that a large number of samples increases susceptibility to drift.

Tips on Number of Samples

- In stepped mode, use a larger number of samples to improve signal to noise. Expect to trade-off measurement speed.
- 101 samples per point is often a good compromise, collecting enough data to average out noise, and keeping the measurement time short enough to avoid drift effects.

Drift Correction This system function reduces measurement error contributed by gradual, continuous temperature changes in the test device, especially long fibers. It is appropriate for use when the temperature of the room or device is gradually changing. This function may not help against rapid or stepped changes, as due to movement of air over the test path.

In swept measurements, the elapsed time between wavelength points is very short, but drift may occur between the sweeps for each polarization state. Selecting Drift Correction in GD/CD+DGD/PDL Swept mode causes the sweep at the first polarization state to be repeated after the four (or six) sweeps are finished. Any drift between the first sweep and this repetition is then used to correct the data, assuming a continuous rate of drift. This function is neither needed nor available for swept measurements of GD/CD only.

In stepped measurements, the elapsed time between polarization states is usually short but temperature drift in very long links may still influence the DGD results. This is corrected in the same way as for swept measurements, with a repeated measurement at the first polarization state.

In stepped measurements, the elapsed time between wavelength points is longer and drift may also distort the GD and CD spectra. To correct this, in Stepped mode the Drift Correction feature also causes the spectrum to be measured twice, once stepping with increasing wavelength and once with decreasing wavelength and the result is formed by averaging the two measurements. This doubles both the number of samples used for averaging and the time required for the measurement.

Laser Setup

TLS Selection

If more than one TLS is available, this setting allows choice of the desired TLS and corresponding wavelength range. This setting should be made before choosing the Start and Stop Wavelengths in the Measurement Range Setup.

TLS Power Limit

This control sets the upper limit for TLS output power. The 86038B will set the TLS output power to the maximum level that can be delivered over the complete normalization or measurement range, but will not set a value higher than the TLS Power Limit. The operator can use this setting to limit or set TLS output power. A limit may for example be useful when measuring an amplifying device, to avoid exceeding the input power limit of the receiver (0 dBm). The maximum deliverable output power depends primarily on the wavelength range, but also somewhat on the sweep speed and wavelength increment. An appropriate limit setting at normalization can be used when trying measurements at different increment and IFBW settings, in order to assure that the measurement can always be made at the same power level. When the measurement cannot be made at the same power level, the normalization data will be invalidated.

The 81600B #200 laser, for example, provides -1 dBm over its full wavelength range and $+3$ dBm over the range of specified dynamic performance. For other tunable laser sources please refer to the individual TLS specification. At higher TLS power levels, amplitude dynamic range and group delay noise is improved, but amplitude-to-phase conversion errors increase, so use of phase dynamic accuracy (PDA) correction is more important.

If the power limit is set higher than the TLS can deliver over the measurement range, then the limit does not change the power used by the 86038B.

TLS Output

This setting can be used to change the TLS Output used from the standard "high power" Output 2 to the Low-SSE Output 1. The latter can be used to measure higher out-of-band rejection (isolation) for some wavelength-filter devices. Note however that Output 1 provides less laser power and may thus reduce the available power dynamic range of the measurement. The PDA correction, which can be chosen during Normalization, is not available when using Output 1.

Settle TLS Now

Initiates the TLS Settling, "Lambda Zero", process on the tunable laser source. The laser monitors temperature change, and automatically performs a recalibration of the wavelength if required to maintain specified operation. The PDLA status message, Estimated Time Until Next TLS Settling, indicates the estimated time before such a process is required. Under normal temperature conditions, the status message shows up to 120 minutes as the estimate. This status message is updated every minute.

A TLS Settling can be initiated in two ways by the PDLA:

- TLS Settling can be initiated by the user at anytime, with the Settle TLS Now control. This may be convenient before a long measurement process to avoid interruption.
- When the estimated time indicator turns from a yellow to red background, a TLS settling will automatically be started within approximately 4 minutes. In Swept mode, the current measurement will finish before the TLS Settling is initiated. In Stepped mode, the measurement will pause after the current step is measured and restart with the next step once the TLS Settling is complete. We recommend that once the icon turns from green to yellow, that you start a TLS Settling immediately, before beginning a new measurement.

Tip: A TLS Setting Status indicator is also displayed as a Lambda symbol at the bottom right of the display. The color of the Lambda symbol changes in the same way as described above. A TLS Settling can be initiated by right-clicking the icon and selecting Settle TLS Now.

Trigger Mode

Single Sweep

In swept measurement mode, this mode causes the Begin Measurement control to initiate one measurement of the wavelength range.

Repeat Sweep

In this mode, the spectral measurements are continuously repeated until the measurement is canceled or the instrument is set to Single Sweep. This mode results in an optical spectrum analyzer-like real-time display and is valuable while making adjustments.

Tip: Fast access to the Trigger Mode setting is from the icon on the toolbar.

Measure Menu

Begin Measurement

Starts the measurement process as specified in the Setup menu. A measurement progress indicator is shown at the bottom left of the display. The indicator only appears during a measurement. The estimates of measurement time given with the progress bar are approximations and do not include all possible activities of the instrument.

Begin Normalization

Accesses the Normalization Viewer panel. Normalization is used to measure the loss and group delay of the instrument path. Later this data is subtracted from the DUT measurement to generate measurement results. Valid normalization data, as indicated by a green icon at the bottom of the screen, is always required before measurements including DGD/PDL. Measurements of only GD/CD can be performed without valid normalization, to allow quick evaluation of a device or the setup parameters, but the results will not be fully calibrated. Existing normalization data will be invalid for measurements, when some setup parameters are changed, and then a new normalization is required. Changes for which normalization data become invalid include: increase of measurement range beyond the normalization range, increment decrease by more than a factor of two, TLS output power change, and modulation frequency change. (Note that changing IFBW or increment sometimes also changes the deliverable TLS output power, which may invalidate the normalization data. This can be avoided by slightly limiting the output power.) Changing the mode of measurement between swept and stepped or adding DGD/PDL will also invalidate the current normalization.

Phase Dynamic Accuracy

Enables normalization measurements to correct for amplitude-to-phase conversion effects in the PDLA. Selecting this correction causes an extra optical power-dependent calibration to be made, which is used to correct the measurement results. The extra calibration requires about 1 minute. If PDA data

already exists for the chosen wavelength range and modulation frequency, the measurement will not be repeated but the existing data will be used in the measurement. To prevent its use or force a new PDA calibration, the PDA Correction data can be cleared.

Tips on Phase Dynamic Accuracy Correction

- If the PDA is performed, both swept and stepped measurements will automatically apply it so long as the modulation frequency is unchanged. The use of PDA correction is recommended for all measurements but may be omitted in cases where the loss variation of the test device over wavelength is small and when the dispersion being measured is very high. Some fiber measurements fit this description.
- PDA should also be selected for measuring low values of DGD, because small amount of PDL in the DUT and instrument can contribute to the DGD result via the amplitude-to-phase conversion.
- The PDA effect varies with modulation frequency, and for this reason the phase dynamic accuracy correction is valid only at the modulation frequency at which the PDA calibration was performed.

There is often some advantage in attenuating the power level seen at the receiver to reduce the need for PDA correction. The amplitude-to-phase conversion effect is strongest at high input power to the receiver, so if a test device has little attenuation, limiting the TLS output power can reduce this effect without significantly increasing noise. Turning off the PDA corrections speed up normalization, but would seldom improve the accuracy of the measurement. For example, one of the higher slope regions of PDA correction at 2 GHz modulation frequency can cause GD errors of 380 fs/dB when PDA corrections are off. Similarly, at 10 MHz modulation frequency, GD errors of 18 ps/dB can result when PDA corrections are off. In order to prevent measured PDA correction data from being used in a measurement, the results should be cleared with the control described below.

Normalize All Ports

If a switch is installed to provide the multiple ports option, this box can be selected if a separate set of normalization data for each port is desired. If the box is not selected, normalization will only be measured at one port and used for the measurements of each port.

The drop-down Port selector allows the port for the normalization measurement to be selected. If all ports will be normalized, the operator selects the box and then sequentially selects each port for measurement and makes the appropriate connection with the normalization patchcord.

The normalization measurement results can be viewed in the graphs. The controls at the top right of the display can be used to select the port and the polarization state to be viewed. If only GD/CD measurements are made, there will only be results for Pol.-State 1.

Clear

Clear accesses a panel to clear the normalization data. You can choose to clear the modulator bias value, phase dynamic accuracy data, or GD and Gain/Loss offsets.

Cancel

Cancel aborts the normalization process. When Cancel is selected, the current laser sweep will be finished before the measurement can be stopped.

Normalize

Normalize starts the normalization process.

Close

Close exits the Normalization Viewer.

Begin Length

Accesses the Begin Length setup panel and starts the specialized group delay measurement of a fiber-optic cable. See [“Length”](#) for information on setting up a length measurement.



Cancel Measurement

Stops the measurement, normalization, or length measurement whichever one is currently in progress.

Display Menu

X Scale

Minimum

Specifies the start wavelength to be viewed on the graphs. This setting is independent of the measurement wavelength range set in the Measurement Range setup panel.

Maximum

Specifies the stop wavelength to be viewed on the graphs. This setting is independent of the measurement wavelength range set in the Measurement Range Setup panel.

Center

Specifies the center wavelength to be viewed on the graphs. This setting is independent of the measurement wavelength range set in the Measurement Range setup panel.

Span

Specifies the wavelength span to be viewed on the graphs. This setting is independent of the measurement wavelength range set in the Measurement Range setup panel.

View Full Width

Sets the graph's start and stop wavelengths range to the same values as the existing measurement curves.

Synchronize All Graphs

Sets all four graphs to the same X scale settings of the currently active graph.

Snap Enabled

When selected, the mouse pan and zoom functions are snapped to a rounded value of wavelength units. This is useful when you want to keep the units in an easy to calculate form. Clear Snap Enable when you need precision tuning of a portion of the graph where the exact settings are important.

Y Scale

Ref Level

Specifies the reference level value, the maximum expected value to be measure, at the reference level position. The Y-scale units depend on the type of measurement.

- DGD and GD measurements are specified in ps.
- Gain/Loss and PDL measurements are specified in dB.
- CD measurements are specified in ps/nm or ps/nm-km.
- PCD is specified in ps/nm.
- Depolarization Rate is specified in ps.
- 2nd order PMD is specified in ps².
- Optical phase is specified in radians

Units/Division

Specifies the units per division of the vertical scale.

Ref. Level Position

Sets the position of the reference level to the chosen scale line on the graph, where 0 is the bottom and 10 is the top line.

View Full Range

Ensures that the highest and lowest Y-value is visible on the trace.

Snap Enabled

When selected, the mouse pan and zoom functions are snapped to a rounded value of amplitude units. This is useful when you want to keep the units in an easy to calculate form. Clear Snap Enable when you need precision tuning of a portion of the graph where the exact settings are important.

Auto Scale

Centers the graph around the response. This function is useful after using the zoom functions.

Scale to Fit

Selected Trace Performs an auto scale for the selected trace on the selected graph only.

Selected Graph Performs an auto scale for the selected graph only.

All Graphs Performs an auto scale on all graphs.

Offset GD trace to zero at:

Group delay traces are relative measurements. Depending on the device, you can set the zero point to one of the following options.

- Center Point
- Minimum Point
- Maximum Point
- No Offset

Autoscale During Measurement For Swept measurements, an autoscale is performed on all graphs at the end of a measurement.

For Stepped measurements, an autoscale on all graphs is performed at the end of each wavelength increment.

Marker Setup

Show Markers

Turns the marker functions on or off.

Tip: Toggle the markers off, then on to reposition the markers near the center of the viewable range.

Marker Position

Allows you to enter in a specific wavelength for both marker 1 and marker 2.

Tip: If markers are turned on, the curve fit function will only apply to the region of the trace that is between the markers. Refer to [“Curve Fit” on page 127](#) for more information.

Marker Search

Shortcut accesses the Marker Search panel. This is useful when using a mouse to navigate the functions.

Marker Function

Shortcut accesses the Marker Function panel.

Marker Search

Moves the active marker to either the peaks or pits of the trace.

Far Left places the marker on the last peak or pit on the left side of the trace.

Next Left places the marker on the next peak or pit left of the current marker position.

Next Right places the marker on the next peak or pit right of the current marker position.

Far Right places the marker on the last peak or pit on the right side of the trace.

Lowest places the marker on the lowest peak or pit amplitude on the trace.

Next Down places the marker on the next lowest peak or pit amplitude of the current marker position

Next Up places the marker on the next highest peak or pit amplitude of the current marker position.

Highest places the marker on the highest peak or pit amplitude on the trace.

Search Criteria

Search Type Specifies the active marker as either a peak or pit marker.

Peak places the active marker on a peak of the active trace which meets the excursion criteria. Use the marker search toolbar to navigate to the desired peak of the trace.

Pit places the active marker on a pit of the active trace which meets the excursion criteria. Use the marker search toolbar to navigate to the desired pit of the trace.

Active Marker Specifies which marker is currently active, Marker 1 or Marker 2.

Excursion Sets the peak or pit excursion value for the marker search functions. It is used to determine which spectral features are classified as peaks or pits. To be classified, each trace peak or pit must rise, and then fall by at least the excursion value. Setting the value too high will result in failure to include the smaller responses. Setting the value too low will cause all spectral features to be chosen.

Marker Function Shortcut accesses the Marker Function panel.

Marker Search Shortcut accesses the Marker Search panel.

Marker Functions

Active Trace

Displays the currently active trace. Use the Next Trace key in the Navigation area to change the active trace.

Active Marker

Displays and allows you to specify the active marker. Before using Marker Search and Marker Functions, the Active Marker must first be identified.

To Center

Moves the scale to put the currently active marker at the center of the viewable vertical range.

To Reference Position

Moves the scale to put the currently active marker at the reference level position on the display.

Zero GD Trace (Applies only to GD Trace)

Group delay is a relative measurement. Use this function to offset the GD trace to set the marker-selected point to a GD of zero. Also refer to [“Offset GD trace to zero at:” on page 121](#).

Marker Setup

Shortcut accesses the Marker Setup panel.

Marker Function

Shortcut accesses the Marker Function panel.

Define Trace

Assigns a measurement on a specified port to a specific trace on the graph. Up to two measurement traces can be assigned to each of the four graphs.

The following example shows how to assign the CD trace and the DGD trace for Port 1.

- 1 From the Navigation area, press **Change View** until all four traces are shown on the display.
- 2 From the Navigation area, press **Next Trace** until the desired graph and trace tab are highlighted (that is, shown in white).
- 3 From the Display area, press **Define Trace**.
- 4 In the Define Trace setup panel, select *Port 1* and *CD* as the measurement type.
- 5 From the Navigation area, press **Next Trace** to highlight the second trace tab on the graph.
- 6 In the Define Trace setup panel, select *Port 1* and *DGD* as the measurement type.
- 7 Press *Close* to save the trace definitions.

For each trace you can specify the Port and one of nine measurement types.

- GD (Group Delay)
- CD (Chromatic Dispersion)
- DGD (Differential Group Delay)
- PDL (Polarization Dependent Loss)
- Gain/Loss
- PCD
- Depol. Rate
- 2nd Order PMD
- Optical Phase

GD Options Area

Show DGD Contribution Displays error bars on the group delay trace to show how much the GD varies due to polarization dependence. This option only applies when DGD data exists. Refer to “CD/PMD” on page 104.

CD Options Area

Dispersion coefficient (ps/nm-km) Dispersion coefficient can be displayed if the device length has been measured or specified (See “Length”). If this option is enabled, each value in the CD trace is divided by the device length (km) to yield the CD coefficient.

GD Source By default, the CD trace is calculated using the slope of the actual GD trace. If a GD trace has been smoothed or curve fitted, selecting that trace as the GD Source may provide a lower noise CD trace.

Gain Options Area

Show PDL Contribution Adds display of maximum and minimum Gain/Loss traces, with respect to polarization state, if PDL data exists.

Attenuation Coeff. (dB/km) The attenuation coefficient can be displayed if the device length has been measured or specified (See "Length"). If this option is enabled, each value in the Gain/Loss trace is divided by the device length (km) to yield the attenuation coefficient.

Smooth



Smoothing performs the selected type of averaging of neighboring wavelength points to reduce the noise without losing too much detail. The smoothed trace takes over the current trace color and the unsmoothed (raw) trace is shown in gray.

Enable Smoothing

When selected, smoothing is turned on.

Algorithm

Boxcar calculates a moving average using an odd number of consecutive data points centered on each element in the original trace.

Exponential smoothing calculates each smoothed point as a weighted average of the measured trace points, where the contribution of each point, offset from the wavelength of the smoothed point, is reduced by the Weight factor with respect to the next closer point. The smoothing is applied both forward and backward to eliminate a bias.

Gaussian takes a value for sigma (standard deviation) and applies a point centered gaussian smooth. The slider sets the value of sigma in units of the wavelength interval.

Hanning performs a weighted average smooth where the weights are 25%, 50%, 25% for a 3-point window.

Tips on the Smooth Function

- Be cautious when using smoothing functions, as severe smoothing can affect the accuracy of a succeeding curve fitting process.
- The fact that direct differentiation of the group delay data emphasizes the noise suggests applying smoothing to the group delay curve before differentiation.
- The combination of moderate smoothing of both the GD trace and the CD trace may lower the CD trace noise and preserve the CD wavelength resolution better than smoothing the CD trace alone.

Window

Use the slide bar to select the number or weight of consecutive trace points that are processed to yield one smoothed point (at the center of the Window). The smoothing process is repeatedly centered on each trace point.

Hide Original Trace

Removes the original trace from the graph so that only the smoothed trace is visible.

Curve Fit



Extracting individual values, like zero-dispersion wavelength, from chromatic dispersion measurements is greatly improved by fitting an appropriate model equation to the measured relative group delay data. Because direct differentiation of the raw group delay data tends to amplify the effect of noise, the values λ_0 , S_0 and $D(\lambda)$, the value of dispersion at any particular wavelength, are calculated from the fitted curve. The model equation should be chosen according to the type of device being measured.

Curve Fitting also allows for comparing measurement to the desired result and determining the deviation or ripple.

Enable Curve Fit

When selected, the specified curve fitting algorithm is applied to the trace. If markers are activated, only the wavelength range of the trace between the markers is used for the fit.

Algorithm

The fits described below can be made dependent on wavelength or optical frequency; that is the independent variable x in the equations can be either wavelength or frequency. The labels for the frequency dependent fits include the text "(frequ.)".

Linear fits the data to a straight line. The equation is of the form: $y = mx + b$. It is commonly used for dispersion-shifted fiber in the 1550 nm wavelength region, in which waveguide dispersion is dominant. This is also a useful fit for dispersion compensating devices and to evaluate the GD ripple of filter components. Similarly, the Linear (frequ.) curve fit is appropriate for fitting the optical phase of a component with no dispersion, and the deviation gives the phase ripple.

Cubic makes a least squares fit to the 3rd-order polynomial equation, $C_3x^3 + C_2x^2 + C_1x + C_0$,

Quadratic makes a least squares fit to the equation $y = Ax^2 + Bx + C$. This is a good approximation to any parabolic shape, for example to locate the zero-dispersion wavelength from GD data. The Quadratic (frequ.) fit is appropriate for the optical phase of a measurement with constant dispersion (at least over a short wavelength range).

3rd Order Sellmeier fit, ax^2+b+cx^{-2} , is commonly used for dispersion-unshifted fiber, in which material dispersion is dominant.

5th Order Sellmeier The 5th-order Sellmeier fit, $ax^4+bx^2+c+dx^{-2}+ex^{-4}$, although more affected by noise and instabilities in the measurement path, provides more general purpose curve fitting. The five-term Sellmeier fit can yield multiple zero-dispersion wavelengths. All the values and their associated slopes are shown in the graphs. The system searches for dispersion zeros in a wavelength range equaling approximately five times the measurement span (2.5 times each side of the center wavelength). This allows identification of zero-dispersion wavelengths which fall outside of the measurement range. In some cases, due to the peculiarity of the five-term Sellmeier fit, zero-dispersion wavelengths found outside of the measurement range may not correspond to actual zero-dispersion wavelengths of the device under test. This fit often provides the lowest RMS Error of the various fits, so it's deviation is useful for assessing noise and short-period ripple.

Equation Area

Displays the general form of the selected equation and additional information like parameter values specific to the selected curve fit.

Deviation Only

The trace displays the difference between the measured data and the curve fit. This is useful to evaluate a fit and for evaluating trace ripple.

Hide Original Trace

Removes the original trace from the graph so that only the curve fit trace is visible.

Navigating the Front Panel

Change View

Displays one, two, or four graphs sequentially as Change View is pressed. Multiple graph views show a combination of ports and measurement traces at the same time. Refer to “[Define Trace](#)” on page 124 to define each graph with specific measurement traces.

To make a particular graph full screen, activate the desired trace using the Next Trace key and then press Change View until just that one graph is displayed.

Next Trace

Activates each graph trace in a sequential manner each time the Next Trace key is pressed. If one graph is displayed, repeated key presses selects one trace and then the other. If two or more graphs are displayed, repeated key presses select each trace in a sequential order.

Use Next Trace to make any changes to the Display area functions. That is, X Scale, Y Scale, Auto Scale, Marker Setup, Marker Search, Define Trace, Smoothing, or Curve Fit. These functions are defined specifically for each trace.

Toolbar Functions

The Zoom functions in the toolbar are only available with a mouse. The Measurement Control functions are also available from the Measure menu and the Measure area on the front panel.

Zoom Functions



Auto Scale

Returns the x-axis scale that is shared by both traces to the measurement range of the current data. Sets the y-axis scale for the selected trace to values that ensure that the entire trace is visible.

Auto Scale Selected Trace

Performs an auto scale for the selected trace on the selected graph only.

Auto Scale Selected Graph

Performs an auto scale for the selected graph only.

Auto Scale All Graphs

Performs an auto scale on all graphs.

Auto Scale During Measurement

For Swept measurements, an auto scale is performed on all graphs at the end of a measurement.

For Stepped measurements, an autoscale on all graphs is performed at the end of each wavelength increment.



Object Select

Turns off the currently active zoom or pan function and allows you to select and move markers, if markers are turned on.

Full Zoom



Reduces both the x axis and the y axis on the active graph. Therefore, the zoomed-in area expands to fill the entire graph.

To zoom in on a portion of the trace, click the toolbar's Full Zoom icon. Next, click and drag to draw a rectangle on the desired portion of the trace. If you make an error while zooming in, click the Auto Scale icon to restore full view.

Horizontal Zoom



Zooms the wavelength span on the active graph. Therefore, the chosen wavelength area expands to fill the entire display.

To zoom in on a portion of the trace, click the toolbar's Horizontal Zoom icon. Next, click and drag to draw a rectangle on the desired wavelength portion of the trace. If you make an error while zooming in, click the Auto Scale icon to restore full view.

Once you are viewing the minimum possible span, attempting to zoom further will attempt to center the viewable range on the center of the area dragged on with the mouse, but the span will remain the same.

Vertical Zoom



Zooms the y axis span on the active graph. Therefore, the zoomed-in area expands to fill the entire graph.

To zoom in on a portion of the trace, click the toolbar's Vertical Zoom icon. Next, click and drag to draw a rectangle on the desired vertical portion of the trace. If you make an error while zooming in, click the Auto Scale icon to restore full view.

Pan



Moves the entire trace on the active graph with the motion of the mouse pointer. Click the Pan icon and then click on the graph. This function is useful when you zoom in on a portion of the trace, and need to quickly view another portion.

Measure Control Functions



Begin Measurement

Starts the measurement process as specified in the Setup menu. A measurement progress indicator is shown at the bottom left of the display. The indicator only appears during a measurement.



Cancel Measurement

Stops the measurement, normalization, modulator bias optimization, or length measurement whichever one is currently in progress.



Trigger Mode (Toggle Continuous Sweep)

Switches between continuous sweep mode and single sweep mode each time this icon is pressed.



Information Field

The drop-down menu field in the upper right corner can be used to display a parameter of particular interest. Setup parameters like modulator bias or input polarizer position can be shown as well as device results like length and PMD (average DGD) value. Device results can also be chosen among the available ports.

5

Remote Operation

Overview	134
Recommended Control Sequence.	138
How to Configure the PDLA for Networking	138
How to Connect the PDLA to Your Network.	138
How to Change Network Settings.	139
Install the Remote Client	141
Program Structure: events vs. sequential	147
Front Panel Functions with No Remote Equivalent	149
Enumerations.	150
Event Table.	154
Troubleshooting.	157
pdlaClient Front Panel Maps	158
Example 1: Simple Application for Controlling the PDLA.	167
Example 2: Control from Excel.	200
Example 3: Control from LabView.	203
Example 4: Complete application - RepeatMeas.	204

Overview

This chapter will help you control a PDLA from your own computer. The chapter covers how to write your own applications, and how to control the PDLA from existing applications such as Excel and LabView. Note that applications for remote control can also be run on the 86038B itself, which is useful for automated measurement procedures.

The PDLA is a remoting enabled, Microsoft .NET instrument that can be controlled across any LAN that can serve an http web page. The provided remote control client has an Active X interface and a .NET interface, so you can control the PDLA from many established applications such as Visual Basic 6.0 and VBA, as well as from .NET enabled applications such as C#.

The PDLA uses .NET remoting as the foundation for its external communications. Remoting is the process of programs or components interacting across different processes or machines. This technology provides the foundation for distributed applications and it replaces DCOM.

In .NET remoting, the server program publishes an object on a network channel and the client program subscribes to that channel when loading or connecting to that object. In the case of the PDLA, a RemoteServices object is published to an http channel and the subscribing client program is the RemoteClient. A Remoting server (RemoteServer) is embedded in the PDLA user interface as a Server Activated Object. (That is, the user interface starts the Remoting Server when the application starts and is running when a client connects.) The RemoteClient is a layer of abstraction, which provides customers with an easy to use interface that provides events and methods to control the PDLA and retrieve measurement data.

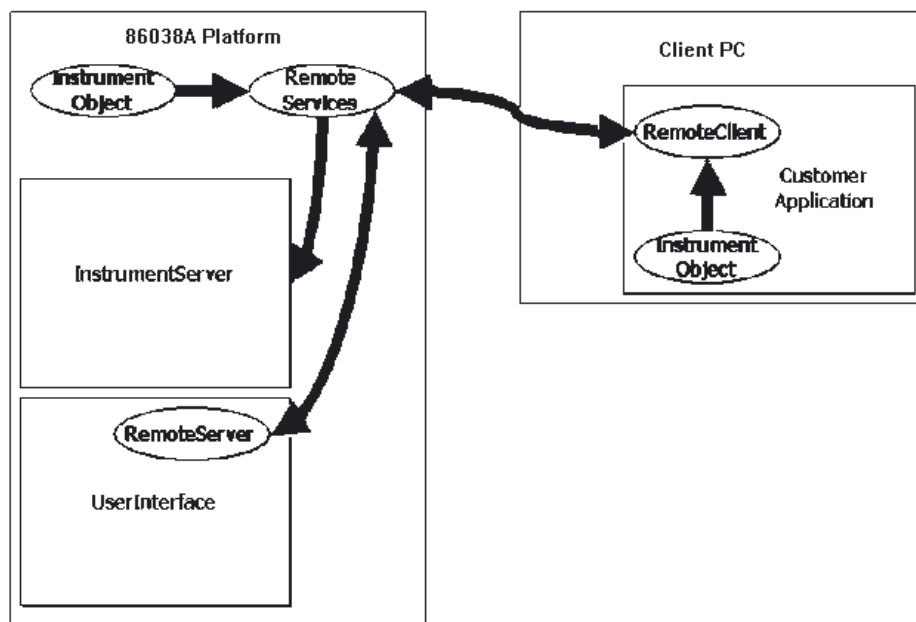


Figure 18

This section assumes that you are familiar with the front panel operation of the PDLA. If you are unfamiliar with the instrument, refer to [Chapter 1, “Getting Started”](#) for an introduction to the front panel, and [Chapter 3, “Making Measurements”](#) for good measurement techniques.

While this chapter assumes you are familiar with your programming environment, it does not assume familiarity with controlling remote objects from within that environment. Detailed examples are provided for VB 6.0, VB.NET and C#, the most common environments for controlling the PDLA. These examples start from a blank project allowing you to see all the steps required.

Agilent Technologies does not recommend controlling the PDLA from C or C++ because of the difficulty of interfacing with COM objects that return safe arrays.

This chapter also includes examples that pull data directly from the PDLA into both Excel and LabView. These are very useful if you are setting up measurements manually, but want to analyze the results on your own computer.

How to Use This Chapter

Depending on your familiarity with programming client/server models, you will want to use this chapter differently. If the subject is new to you, read the background material in this chapter and then go through Example 1 step-by-step for a complete introduction to creating a client application. If the client/server model is familiar, you may wish to start with “Complete Source Code” on page 195, then review the “Recommended Control Sequence” on page 138. You will find the “pdlaClient Front Panel Maps” on page 158 very useful for identifying specific commands.

Note that “Example 1: Simple Application for Controlling the PDLA 167” consists of the same simple program implemented three times: once in VB 6.0, VB.NET and C# respectively. Choose the version for the language you are most likely to use.

Example 1 is presented in a series of steps, meant to be manually typed in (or pasted from the electronic version of this manual). The example shows you how to create a project and all the steps it takes to interface with the PDLA.

Example 2 demonstrates how to transfer data directly into Microsoft Excel, using Excel 2000 or later. Example 2 uses a VBA macro which is very similar to VB 6.0.

Example 3 provides a set of LabView vi's allowing you to easily control the PDLA from LabView.

Example 4 is a complete application for taking PDLA measurements at specific intervals over a period of time. It demonstrates proper error handling and includes considerably more safety checking than the other examples do.

NOTE

A VC++ remote programming example has not been provided because:

- .NET passes its arrays to COM as Safe Arrays. This is easily handled by VB 6.0 but VC++ takes a lot of extra work. The Class Wizard cannot handle them all. Programmatically it takes a bit of code to do this on the VC++ side (but according to the documentation it can be done). This primarily affects the retrieval of the raw data array.
- VC++ has difficulty dealing with two DLLs that have interdependencies. The RemoteClient DLL uses data types in the InstrumentObjects DLL. One example used the Class Wizard and converted all the data types to basic data types, for example, int, double, and long. The InstrumentObjects data types are much easier to use.
- Events may not be threadsafe in VC++. Internally in the RemoteClient, code has been added to pass the events to the clients GUI thread. This may not work in VC++ and the client may have to do this.

Tip: The electronic version of the User's Guide is available from the PDLA Help menu and on the CD that was supplied with the instrument.

Because this manual is also distributed electronically, the formatting of code segments has been left to line-break where it needs to. This allows you to use the text selector tool to grab the code and paste it into your program. This means that the word-processor may occasionally insert a dash on the text as it appears on the screen that is not there when you copy the text. If you are manually typing the lines, you have to watch for and remove those dashes.

Recommended Control Sequence

This section provides a detailed discussion of the steps required to control the PDLA. This section does not discuss language-specific considerations, even though the example code is in VB 6.0. Refer to [“Example 1: Simple Application for Controlling the PDLA” on page 167](#) to experiment with these concepts. Example 1 has versions specific to VB 6.0, VB.NET and C# that show you how to start with a blank project through working with the PDLA system.

How to Configure the PDLA for Networking

Remote programming of the PDLA is only possible if the PDLA is connected to a local area network (LAN) via the built-in LAN connector. When the PDLA is connected to a network, it is also possible to connect it to network printers and remote servers, with access to shared folders and files.

How to Connect the PDLA to Your Network

The PDLA comes configured for DHCP networking, and has a default machine name. In many cases, connecting the PDLA to your LAN is simply a case of registering the machine name with your IT department.

NOTE

Do not connect the PDLA to a network that is configured to automatically install software on network devices. Installing or overwriting files on the PDLA computer system may impact the operation of the instrument. Please contact your network administrator or IT department to find out if you have this type of network.

NOTE

The PDLA LAN connector supports 10 Base-T and 100 Base-T Ethernet networks using TCP/IP and other Microsoft supported networking protocols. The PDLA uses Microsoft® Windows XP.

To connect to your LAN, power-down your instrument, connect an active LAN cable, then power-up your instrument.

How to Change Network Settings

You can change the PDLA network settings as needed so that it connects properly to your specific network.

NOTE

Because your network settings are unique to your IT infrastructure, Agilent Technologies will not be able to assist you with connecting your instrument to your network. Please contact your network administrator or IT department for assistance. For more information, refer to the MS Windows resource kit (available from Microsoft) that is appropriate for your computer system. You can also refer to the online Help for Windows XP (Start > Help).

NOTE

By default, as the instrument starts up, you are logged on as an administrator with the logon name PNA_Admin. The login password, which is usually not needed, is “agilent”. Agilent only recommends using the PDLA application while you are logged on as an administrator.

You can change network settings by using the standard Microsoft® Windows functions. To access these functions, you must exit the PDLA application. You can do this by clicking Exit on the File menu.

To view or change the computer machine name

NOTE

If you only wish to *view* the machine name, you can do this while the PDLA application is running. On the Help menu, click About. The machine name is displayed near the middle of the dialog box.

- 1 On the File menu, click Exit and then click Yes.
- 2 Click Start, point to Settings, and then click Control Panel.
- 3 Double-click the System icon and click on the Computer Name tab. From here you can view or change the machine name.
- 4 When you have finished making changes, restart the instrument.

To configure TCP/IP to use DNS or WINS

NOTE

If using a protocol other than TCP/IP, please contact your IT department for assistance.

NOTE

Editing your instrument's protocols and file access permissions can result in unwanted behaviors that are difficult to reverse. Ensure that your changes are valid!

NOTE

Please consult with your network administrator concerning advanced TCP/IP and multi-protocol configuration settings to support your network.

NOTE

Please contact your network administrator or IT department if you have any problems connecting the PDLA to your network.

- 1 On the File menu, click Exit and then click Yes.
- 2 Click Start, point to Settings, and then click Network and Dial-up Connections.
- 3 Then click Local Area ConnectionProperties.
- 4 On the General tab (for a local area connection) or the Networking tab (all other connections), click Internet Protocol (TCP/IP), and then click Properties. From here, you can make all desired changes.
- 5 When you have finished making changes, restart the instrument.

NOTE

For more information, click Start > Help > Index, and search for "DNS" or "WINS" or "static" or "dynamic."

To configure TCP/IP for static or dynamic addressing

- To get started, follow the same steps listed above.

Install the Remote Client

NOTE

This installation is not for the PDLA itself. (Applications using the remote programming commands can be run on the 86038B itself without installing the remote client package.)

- 1 Install the .NET framework from Microsoft. Go to www.microsoft.com and search for 'How to get the Microsoft .NET framework'. Be sure to get the framework and all the service packs. Make sure that you get the framework, not the SDK (software development kit.)
- 2 The 86038B CD shipped with the PDLA contains the Remote Client Installation to install the PDLA specific DLL's and the programming examples. Insert the CD into the CD drive, use Windows Explorer to find PDLA Remote Client Installer Folder. If a more recent version of the 86038B User Interface software is installed on the 86038B, then the corresponding version of the Remote Client installation must also be used. The most recent version is available from the Agilent web site.

NOTE

If you scan the GPIB, or change the hardware selection, you will need to start a new measurement (including the setting of the measurement parameters, and the making of a reference measurement) before you can make further measurements.

3 Instantiate pdlaClient

Before you can interface with the PDLA, you have to declare the client object. The client object is an object written by Agilent and provided to you as part of the PDLA Remote Client install. This object goes out over the LAN to interface with the PDLA, and provides a defined interface for you to use.

To provide better compatibility, the PDLA actually has two remote clients - an Active X component for use with Visual Basic 6.0 and applications such as Excel, Matlab and LabView, and a .NET interface for use with languages such as VB.NET and C#.

There are three things you typically have to do to instantiate a client:

a Add the reference

For the Active X interface, you need to reference InstrumentObjects.tlb and RemoteClient.tlb. For .NET, you need to reference the DLL's directly: InstrumentObjects.dll and RemoteClient.dll. These files are by default installed in C:\Program Files\Agilent\Agilent ODA Remote Control. In Visual Studio, you add references from the Project > Add Reference menu.

b Declare the object.

In Visual Studio, you typically declare the object separately from instantiating it. For Visual Basic 6.0:

```
Private WithEvents pdlaClient As RemoteClient.Communicator
```

c Instantiate the object.

Use the keyword "new" to instantiate the object. For Visual Basic 6.0:

```
Set pdlaClient = New RemoteClient.Communicator
```

4 Establish Connection to the PDLA

The first thing you have to do with your pdlaClient is to establish a connection to the particular PDLA you want to talk to. You need to know the network name of the PDLA. Once the PDLA is on the LAN, you can get the network name from the front panel by select Help > About. The network name is displayed near the middle of the panel.

```
Call pdlaClient.Connectivity.Connect(<insert pdla name>)
```

Once you have made the connection request, wait for the connection to complete (approximately 5 seconds). The PDLA will send you an event when the connection is valid, or you can query

```
pdlaClient.Connectivity.IsConnected.
```

For every call to Connect() there has to be a call to Disconnect() because the PDLA tracks how many remote users are connected. You can connect and disconnect as many times as you like if you want to turn control over to another program or to the front panel. At the very end of your program execution you should also terminate your session to clean up the connection.

```
(pdlaClient.Connectivity.TerminateSession)
```

5 Set Measurement Type, Range and Port(s)

Next, the PDLA needs to be configured for the measurement that you want to make. Set up the measurement before doing the normalization so that the normalization will cover the measurement conditions. If you have several measurements to make, you may wish to manually configure the normalization wavelength range to cover all of the measurements. In this case, you would defer this step until after normalization.

A sample configuration setting up for a swept CD and Gain measurement from 1535 to 1545 nm on Port 1 might look like:

```
pdlaClient.MeasurementRange.XStart = 1535
pdlaClient.MeasurementRange.XStop = 1545
pdlaClient.DispersionMode = eDispersionMode_CD_Swept
pdlaClient.Ports = ePortsEnabled_Port_1_Enabled
```

The start and stop wavelength is set in nm. Since a swept mode was chosen, you may want to specify the increment:

```
pdlaClient.Resolution.Increment=0.01
```

There are four different measurement modes, the same choices you get on the CD/PMD front panel setup menu:

Table 5-3. Measurement Modes

eDispersionMode_CD_Swept	Measure Gain, Relative Group Delay, and Chromatic Dispersion with the tunable laser sweeping across the specified wavelengths.
eDispersionMode_CD_Stepped	Measure Gain, Relative Group Delay, and Chromatic Dispersion with the tunable laser stepping across the specified wavelengths.
eDispersionMode_CD_PMD_Stepped	Measure Gain, Relative Group Delay, Differential Group Delay, Chromatic Dispersion, Polarization Dependant Loss and second Order PMD properties with the tunable laser stepping across the specified wavelengths.
eDispersionMode_CD_PMD_Swept	Measure Gain, Relative Group Delay, Differential Group Delay, Chromatic Dispersion, Polarization Dependant Loss and second Order PMD properties with the tunable laser sweeping across the specified wavelengths.

The above example sets a step of 0.01 nm. This would set 1001 trace points for the requested 10 nm sweep (1535 to 1545 nm).

You can activate port 1, port 2, or both:

If your setup supports multiport measurements, you can activate single ports:

eODAPort_One Or eODAPort_Two Or ...

You can also activate any combination of ports. If you want to measure Port 1 and Port 2:

```
eODAPort_One + eODAPort_Two
```

If your setup does not support multiport measurements, always activate

```
eODAPort_One
```

6 Normalize

With the system configured for the measurement, the next step is to normalize. This consists of prompting the user to make the connections, starting the normalization and waiting for it to finish. In this example the trigger status is polled until the normalization has been completed. Refer to [“Program Structure: events vs. sequential” on page 147](#) for a more detail discussion about how to synchronize your program with the PDLA.

```
'Use measurement range and port for normalization
pdlaClient.NormalizationRange.UseCustomNormalization = False
'Have user make connection
MsgBox ("Connect PDLA for normalization")
'Start the normalization
pdlaClient.Actions.Normalize ()
' Give it time to get started before checking flags
Sleep 1000

' Wait for TriggerComplete, but no longer than 30 minutes
Dim elapsedTime As Double
elapsedTime = 0
Do
DoEvents
Sleep 100
elapsedTime = elapsedTime + 100
If elapsedTime / 1000 / 60 > 30 Then Exit Do
Loop Until pdlaClient.Status.TriggerComplete = True
```

Tip: The normalization will be performed on the enabled port(s). If both ports are enabled, both ports will be normalized in one sweep, which requires an external coupler. If you do not want the coupler to be included in the normalization, you can temporarily enable port 1, normalize, then enable port 2, normalize, and then enable both ports for measurement.

The sleep statement is used above to slow the loop down - you do not want to query the PDLA hundreds of times a second. To use the sleep command, you need to declare it at the top of your module:

```
Private Declare Sub Sleep Lib "Kernel32" (ByVal dwMilliseconds As Long)
```


Phase dynamic accuracy (PDA) is used to correct for amplitude-to-phase conversion effects in the measurement receiver. The command to enable it:

```
pdlaClient.NormalizationRange.PDAEnabled = True
```

7 Measure

Trigger the measurement, then wait for the sweep to complete. This looks very much like normalization:

```
' Make sure not connected for normalization
MsgBox ("Connect PDLA for measurement")
' Start the sweep
pdlaClient.Actions.Measure
' Let sweep get started before polling
Sleep 1000
Dim elapsedTime As Double
elapsedTime = 0
' Poll the PDLA waiting for the sweep to complete
' NOTE: 30 minutes may not be long enough!
Do
DoEvents
Sleep (100)
elapsedTime = elapsedTime + 100
If elapsedTime / 1000 / 60 > 30 Then Exit Do
Loop Until pdlaClient.Status.TriggerComplete = True
```

8 Transfer Data

Now that there is valid data in the PDLA, you will want to transfer it out. To determine the wavelength axis, read the start and stop wavelengths and the number of points (You can determine the number of points in the returned arrays.) then calculate the wavelength for each point as part of your application.

NOTE

When the PDLA takes the derivative of group delay to calculate chromatic dispersion, the effective wavelength axis shifts one-half of a point spacing upwards (and there is one less point.) So, the wavelength queries force you to specify the measurement type, so you can query the correct wavelengths. Gain, Group Delay, DGD, and PLDL share the same wavelengths, CD, 2nd order PMD, PCD and Depol. Rate all use the offset wavelengths.

This example queries the gain:

```
Dim gainData() As Double
' Get the data and figure out the wavelength axis
gainData = pdlaClient.Results.YData(eMeasurementType_Gain,
eODAPort_One)
npoin = UBound(gainData) - LBound(gainData)+1
```

```
xStart = pdlaClient.Results.XStart(eMeasurementType_Gain)
xStop = pdlaClient.Results.XStop(eMeasurementType_Gain)
xStep = (xStop - xStart) / (npoin - 1)
```

See Table 5-4, “PDLA Enumerations,” on page 150 for the other possible values of the eMeasurementType enumeration.

When this code snippet finishes, the gain data will be in gainData, and you can calculate the wavelength of any point from the xStart and xStep variables.

9 Disconnect from the PDLA

The PDLA keeps track of who is connected to it. If one of the connections quits responding, the PDLA will go through a time-consuming clean-up process to remove references to the dead connection. To avoid this, and to avoid Visual Studio getting confused, it is very important to issue a disconnect statement for every connect statement, even when debugging. If you are debugging and decide to halt program execution, you should either manually run the following statement, or run a function (such as Form_Close) that executes it:

```
pdlaClient.Connectivity.Disconnect
```

In addition, it is necessary to terminate the session when your program is closing down to allow clean up on both sides:

```
pdlaClient.Connectivity.TerminateSession
```

Program Structure: events vs. sequential

A traditional remote control application consists of a list of actions that you send to the instrument, expecting it to execute them in that order and to tell you when it is done. This makes programming easy - you can do your whole measurement in a single function or sub-routine. Because everything always happens serially in the same order, this style of programming is often called sequential or functional programming.

A more advanced approach is to take advantage of multitasking to overlap different instruments and pre and post calculations done in the computer. This is more complex to program, but can have significant execution time advantages. In this scenario, the base of your application becomes a state machine that launches the appropriate activities, then wakes up when the activities complete and determines what to do next. The state machine wakes up when it receives events, such as a thread completing a calculation or an instrument completing a sweep. Because of this use of events, this style of programming is often called event based.

The PDLA is configured as an event based remote client because you can make an event based client behave sequentially, but it is difficult to make a sequential client efficiently multitask.

For a specific example, consider a typical sequence programming GPIB using SCPI (which you cannot do with the PDLA.) Taking a sweep may look like:

```
writePdlA(pdla, "STAR 1520 nm")
writePdlA(pdla, "STOP 1535 nm")
writePdlA(pdla, "INIT:IMM; *OPC?")
value = readPdlA(pdla) // block on sweep complete
```

This example tells the instrument the sweep parameters, triggers the sweep and then blocks until the OPC? returns a 1, indicating the sweep is complete.

To do the same thing in the PDLA, you send the commands and then wait for the PDLA to indicate that it has completed the sweep:

```
pdlaClient.MeasurementRange.XStart = 1520
pdlaClient.MeasurementRange.XStop = 1535
pdlaClient.Actions.Measure
sleep(1000)
loop
  sleep(100)
  elapsedTime += 100
while elapsedTime < MAX_TIME and pdlaClient.Status.TriggerComplete != TRUE
```

Notice that the loop has a sleep statement, which stops the program for 100 ms, letting other programs run, and a timeout if nothing happens for too long. You cannot query a TriggerComplete flag too soon after requesting a sweep because the flag is not clear until the sweep actually starts.

This same loop will work for a sweep or a normalization.

You will also have to wait for the initial connect to the PDLA. It typically takes about five seconds for the connection to be fully established, you can either check for connectivity each time you start a block of activity with the PDLA (which is a good idea) or you can have a similar wait loop. To check for connectivity:

```
if pdlaClient.Connectivity.Connected == FALSE Then
    messageToUser("Lost connection to PDLA, unable to proceed")
    return
end if
```

To open the session and wait for a valid connection:

```
pdlaClient.Connectivity.Connect()
sleep(5000) \\ wait 5 seconds
loop
    sleep(100)
    elapsedTime += 100
while elapsedTime < MAX_TIME and pdlaClient.Connectivity.Connected != TRUE
```

Front Panel Functions with No Remote Equivalent

This section lists front panel functions that do not have an equivalent remote command. In general, functions that are used only rarely and functions that are better implemented on your computer (for example, markers) have no remote control equivalent.

- X Scale
- Y Scale
- Auto Scale
- Marker Setup
- Marker Search
- Marker Function
- Smoothing
- Wavemeter
- Config
- Optical Test Set Heater Set Point
- Navigation

Enumerations

Many communications with the PDLA will involve enumerated types - where there is a finite number of possible values, each with a unique spelling. Because the PDLA exports both an Active X (for example, VB 6.0) and a .NET interface, there are two slightly different versions of all the enumerations. The .NET version takes the form object.name, whereas the Active X interface uses object_name. When you use VB 6.0, the enumerations are automatically qualified for you by the function that is passing or returning them, so you only need to know the spelling below. In .NET, you must enter the complete object path to the enumeration. Fortunately, in most cases IntelliSense will offer you a list of options so you will not have to type the complete object path.

In .NET, the enumerations all come off the following root, unless otherwise specified:

Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon

In VB 6.0, the enumerations all come off of the InstrumentObjects interface, although you will not need to know this.

Table 5-4. PDLA Enumerations

Enumeration	Values in VB 6	Values in .NET
eDispersionMode pdlaClient.DispersionMode	eDispersionMode_CD_PMD_Stepped eDispersionMode_CD_Stepped eDispersionMode_CD_Swept eDispersionMode_CD_PMD_Swept	eDispersionMode.CD_PMD_Stepped eDispersionMode.CD_Stepped eDispersionMode.CD_Swept eDispersionMode.CD_PMD_Swept
eDUTLossGain pdlaClient.DUTMinLossMaxGain	eDUTLossGain_eGain_10dB eDUTLossGain_eGain_0dB eDUTLossGain_eGain_M10dB eDUTLossGain_eGain_M20dB eDUTLossGain_eGain_M30dB eDUTLossGain_eGain_M40dB eDUTLossGain_eGain_M50dB	eDUTLossGain.eGain_10dB eDUTLossGain.eGain_0dB eDUTLossGain.eGain_M10dB eDUTLossGain.eGain_M20dB eDUTLossGain.eGain_M30dB eDUTLossGain.eGain_M40dB eDUTLossGain.eGain_M50dB
eEventLogType Event: NewStatus	eEventLogType_EventError eEventLogType_Information eEventLogType_Warning eEventLogType_SuccessAudit eEventLogType_FailureAudit	eEventLogType.EventError eEventLogType.Information eEventLogType.Warning eEventLogType.SuccessAudit eEventLogType.FailureAudit

Table 5-4. PDLA Enumerations

Enumeration	Values in VB 6	Values in .NET
eGDOffsetMode pdlaClient.Display.GDOffsetMode	eGDOffsetMode_CenterPoint eGDOffsetMode_MaximumPoint eGDOffsetMode_MinimumPoint eGDOffsetMode_No_Offset	ODASensitivity.eGDOffsetMode: ¹ .CenterPoint .MaximumPoint .MinimumPoint .No_Offset
eIFBandwidth pdlaClient.Sensitivity.IFBandwidth	eIFBandwidth_IFBW_30Hz eIFBandwidth_IFBW_70Hz eIFBandwidth_IFBW_100Hz eIFBandwidth_IFBW_200Hz eIFBandwidth_IFBW_300Hz eIFBandwidth_IFBW_1000Hz eIFBandwidth_IFBW_2000Hz eIFBandwidth_IFBW_30000Hz	ODASensitivity.eIFBandwidth: ¹ .IFBW_30Hz .IFBW_70Hz .IFBW_100Hz .IFBW_200Hz .IFBW_300Hz .IFBW_1000Hz .IFBW_2000Hz .IFBW_30000Hz
eInstrumentStatus pdlaClient.Status.InstrumentState	eInstrumentStatus_Aborting eInstrumentStatus_Idle eInstrumentStatus_Initializing eInstrumentStatus_TLSSettling eInstrumentStatus_Measuring	eInstrumentStatus.Aborting eInstrumentStatus.Idle eInstrumentStatus.Initializing eInstrumentStatus.TLSSettling eInstrumentStatus.Measuring
eLayoutMode pdlaClient.Display.ToggleMode	eLayoutMode_QUAD_GRAPH eLayoutMode_DUAL_GRAPH eLayoutMode_SINGLE_GRAPH	eLayoutMode.QUAD_GRAPH eLayoutMode.DUAL_GRAPH eLayoutMode.SINGLE_GRAPH
eLengthMode pdlaClient.Length.Mode	eLengthMode_Measure eLengthMode_Specify	ODALength.eLengthMode: ¹ .Measure .Specify
eMeasNormDataType pdlaClient.Results.FiberLength	eMeasNormDataType_Measurement eMeasNormDataType_Normalization	eMeasNormDataType.Measurement eMeasNormDataType.Normalization
eMeasurementType Used in defining traces and reading data back.	eMeasurementType_CD eMeasurementType_Gain eMeasurementType_GD eMeasurementType_Length eMeasurementType_None eMeasurementType_PDL eMeasurementType_PMD eMeasurementType_PCD eMeasurementType_DepolRate eMeasurementType_SecOrdPMD eMeasurementType_OptPhase	eMeasurementType.CD eMeasurementType.Gain eMeasurementType.GD eMeasurementType.Length eMeasurementType.None eMeasurementType.PDL eMeasurementType.PMD eMeasurementType.PCD eMeasurementType.DepolRate eMeasurementType.SecOrdPMD eMeasurementType.OptPhase
eNormalizationMode pdlaClient.NormalizationMode	eNormalizationMode_SinglePort eNormalizationMode_AllPorts	eNormalizationMode.SinglePort eNormalizationMode.AllPorts

Table 5-4. PDLA Enumerations

Enumeration	Values in VB 6	Values in .NET
enumAcquisitionMode event: NewDataUpdate	enumAcquisitionMode_eDriftMonitor enumAcquisitionMode_eLength enumAcquisitionMode_eMeasurement enumAcquisitionMode_eModBias enumAcquisitionMode_eNormalization enumAcquisitionMode_eOptPolarizer enumAcquisitionMode_eZeroExtPWM	enumAcquisitionMode.eDriftMonitor enumAcquisitionMode.eLength enumAcquisitionMode.eMeasurement enumAcquisitionMode.eModBias enumAcquisitionMode.eNormalization enumAcquisitionMode.eOptPolarizer enumAcquisitionMode.eZeroExtPWM
enumLaserMode event: NewDataUpdate	enumLaserMode_eStepped enumLaserMode_eSwept	enumLaserMode.eStepped enumLaserMode.eSwept
enumOnOff Used many places.	enumOnOff_eOff enumOnOff_eOn	enumOnOff.eOff enumOnOff.eOn
enumPolStatesPMD pdlaClient.NumPolStatesPMD	enumPolStatesPMD_eFour enumPolStatesPMD_eSix	enumPolStates.eFour and enumPolStates.eSix
enumStatus Used to indicate progress, such as trigger or sweep progress.	enumStatus_ABORTED enumStatus_COMPLETE enumStatus_Complete_Warn enumStatus_START	enumStatus.ABORTED enumStatus.COMPLETE enumStatus.Complete_Warn enumStatus.START
enumTestType	enumTestType_DriftMonitor enumTestType_GD_CD_PMD_Stepped enumTestType_GD_CD_Stepped enumTestType_GD_CD_SWEPT enumTestType_GD_CD_PMD_Swept enumTestType_Length enumTestType_ModBias enumTestType_PDA enumTestType_WavelengthCorrection enumTestType_OptimizePolarizer enumTestType_ZeroExternalPWM	enumTestType.DriftMonitor enumTestType.GD_CD_PMD_Stepped enumTestType.GD_CD_Stepped enumTestType.GD_CD_SWEPT enumTestType.GD_CD_PMD_Swept enumTestType.Length enumTestType.ModBias enumTestType.PDA enumTestType.WavelengthCorrection enumTestType.OptimizePolarizer enumTestType.ZeroExternalPWM
enumTLSSettlingProbability Event: TLSSettlingProbability	enumTLSSettlingProbability_eHigh enumTLSSettlingProbability_eLow enumTLSSettlingProbability_eMedium enumTLSSettlingProbability_eNever	enumTLSSettlingProbability.eHigh enumTLSSettlingProbability.eLow enumTLSSettlingProbability.eMedium enumTLSSettlingProbability.eNever
eODAPort Used several places to refer to ports.	eODAPort_One eODAPort_Two eODAPort_Three eODAPort_Four	eODAPort.One eODAPort.Two eODAPort.Three eODAPort.Four
eTLSPowerSelect pdlaClient.TLS.PowerSelect	eTLSPowerSelect_High eTLSPowerSelect_LowSSE	eTLSPowerSelect.High eTLSPowerSelect.LowSSE

Table 5-4. PDLA Enumerations

Enumeration	Values in VB 6	Values in .NET
eTraceLetter pdlaClient.Display.DefineTrace	eTraceLetter_TraceA eTraceLetter_TraceB	eTraceLetter.TraceA eTraceLetter.TraceB
eTriggerType pdlaClient.TriggerModeParams	eTriggerType_TriggerContinuous eTriggerType_TriggerSingle	eTriggerType.TriggerContinuous eTriggerType.TriggerSingle

¹ This command comes off the root: Agilent.LWD.Ag86038x.InstrumentObjects.

Event Table

As mentioned in other sections, the PDLA lends itself very well to event based programming. Events are very easy to use in Visual Basic. In C#, you have to know the name of the event and the delegate. The table below lists the available events on the `pdlaClient`, along with the associated status property if you prefer to query rather than wait for events. The C# delegates all come off of `ODARemoting`, as in `ODARemoting.ConnectedDelegate`.

Table 5-5. PDLA Events

Event	Description
	Equivalent Status Property
	C# Delegate
Connected	<p>This event occurs whenever a remote client connects to or disconnects from the PDLA. A boolean pass parameter is set true on connection.</p> <p>pdlaClient.Connectivity.IsConnected()</p> <p>ConnectedDelegate</p>
EstimatedTestTime	<p>An estimate of how long the current test is going to take is sent at the beginning of each test. The first parameter is the name as a string, the second is the test time estimate in seconds.</p> <p>(Not available by query)</p> <p>EstimatedTestTimeDelegate</p>
NewDataUpdate	<p>Indicates when new measurement data is available. Can indicate a partial trace update in stepped mode; swept mode is always a complete trace.</p> <p>Use <code>pdlaClient.Status.TriggerComplete</code> to detect end of sweep.</p> <p>NewDataUpdateDelegate</p>
NewRealTimePortMonitorData	<p>Indicates that a new set of data from the real-time measurement is available.</p> <p>NewRealTimePortMonitorDataDelegate</p>

Table 5-5. PDLA Events

Event	Description
	Equivalent Status Property
	C# Delegate
NewStatus	<p>This event occurs whenever a new status message is presented to the user on the front panel of the PDLA. The pass parameters include the message itself and the severity of the message.</p> <p>pdlaClient.Status.ErrorFlag, pdlaClient.Status.LastError</p> <p>NewStatusDelegate</p>
PresetProgress	<p>This event occurs when the PDLA begins and completes an instrument preset. The pass parameter indicates whether it is the beginning or, end.</p> <p>pdlaClient.Status.PresetComplete</p> <p>ProgressDelegate</p>
Remoting	<p>This event occurs when the user presses the local key on the PDLA front panel and begins controlling the PDLA manually even though a remote process is connected.</p> <p>force with <code>pdlaClient.Display.ToggleRemoteIndicator</code></p> <p>RemotingDelegate</p>
SweepProgress	<p>This event occurs when a sweep is begun, when it is complete, and if it is aborted. The pass parameter indicates which of the three states occurred.</p> <p>Use <code>pdlaClient.Status.TriggerComplete</code> to detect end of sweep.</p> <p>SweepProgressDelegate</p>
TLSSettlingProbability	<p>This event occurs when the PDLA queries the latest temperature reading from the tunable laser. The PDLA calculates the estimated number of minutes until a Laser Settling is required, and reports it as a pass parameter.</p> <p>pdlaClient.Status.EstimatedTimeToTLSSettling</p> <p>TLSSettlingProbabilityDelegate</p>

Table 5-5. PDLA Events

Event	Description
	Equivalent Status Property
	C# Delegate
TLSSettlingProgress	<p>This event occurs when the PDLA starts, or completes the process of settling the tunable laser after temperature has drifted. The PDLA will not make measurements while the settling is in progress.</p> <p>Status.TLSSettlingProgressDelegate</p>
TriggerProgress	<p>This event occurs at the start and finish of most PDLA actions such as sweeping or normalization. The pass parameters indicate what the current activity is and whether it has started, completed, or aborted.</p> <p>Status.TriggerComplete</p> <p>TriggerProgressDelegate</p>

Troubleshooting

As you develop your application, sooner or later you may lose synchronization with the PDLA. This is indicated by errors that talk about the “http connection” or “Automation error.” If you get this type of error, the best thing to do is save and exit Visual Studio, then reopen the project. Visual Studio usually puts your projects in the Documents list on the Start menu, making it easy to open the project. When you rerun the project, your computer will resynchronize with the PDLA, and usually the problem goes away at that point. If the problem is severe, the PDLA will post a message and undergo a clean-up for a lost connection, this will take approximately one minute.

If you are getting this problem frequently, make sure that you always execute a `disconnect()` for every `connect()`, even when debugging. Executing a `TerminateSession()` just before your application exits will help keep the PDLA synchronized.

If Automation errors persist, go into `Project > References` and clear the PDLA objects, and then close the dialog. Reopen the dialog and reselect the references.

pdlaClient Front Panel Maps

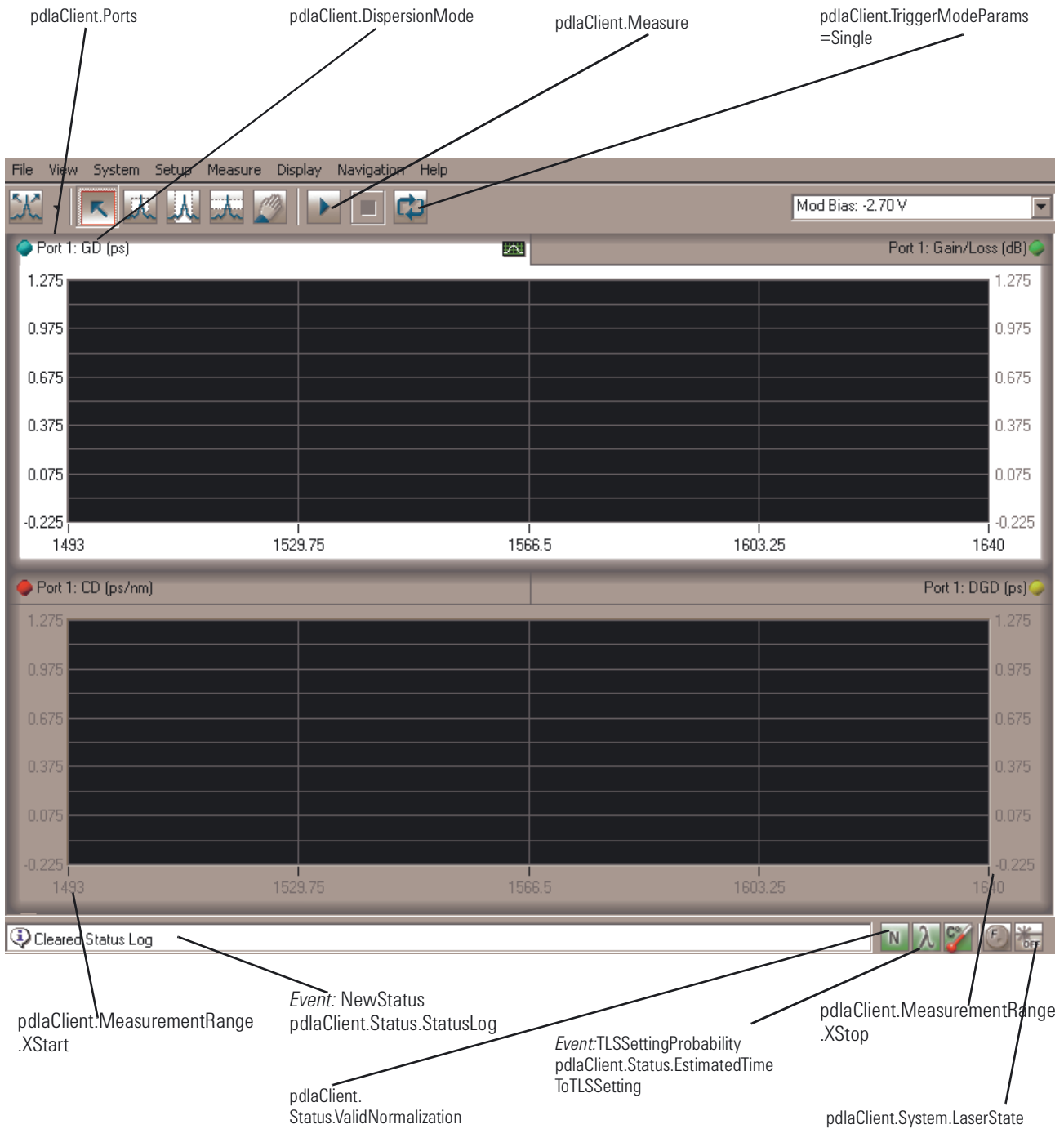
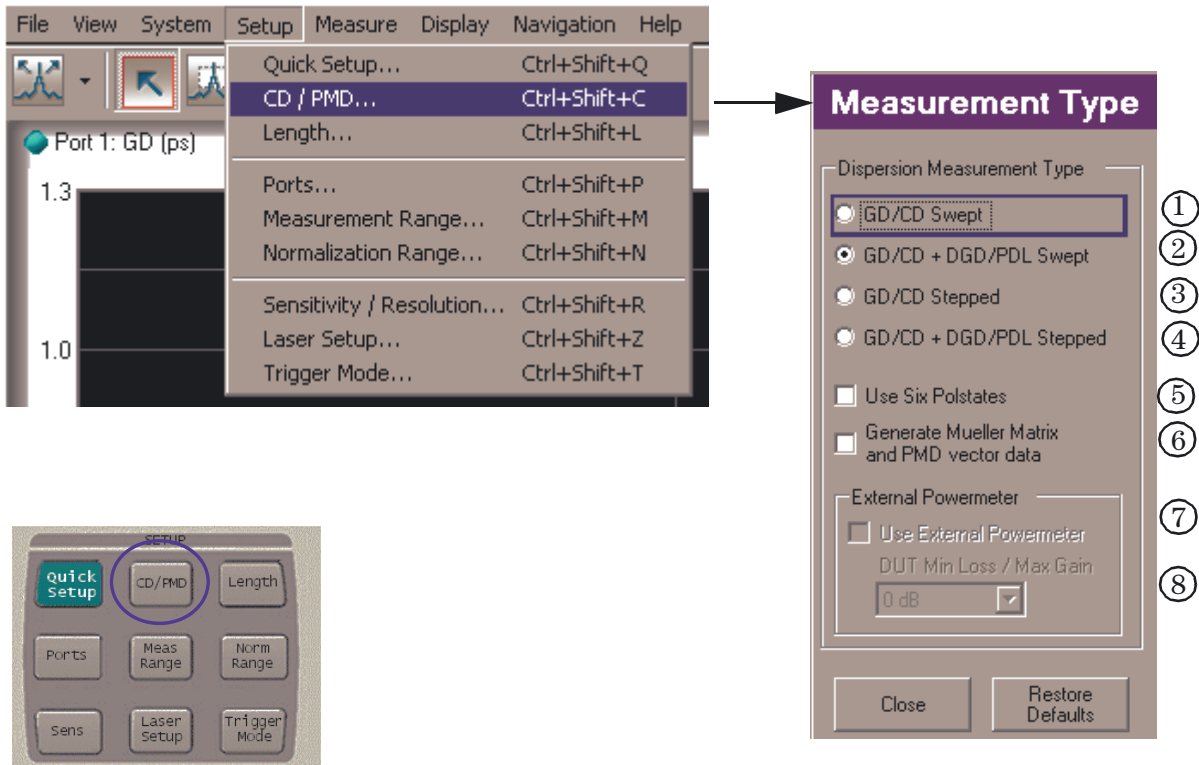
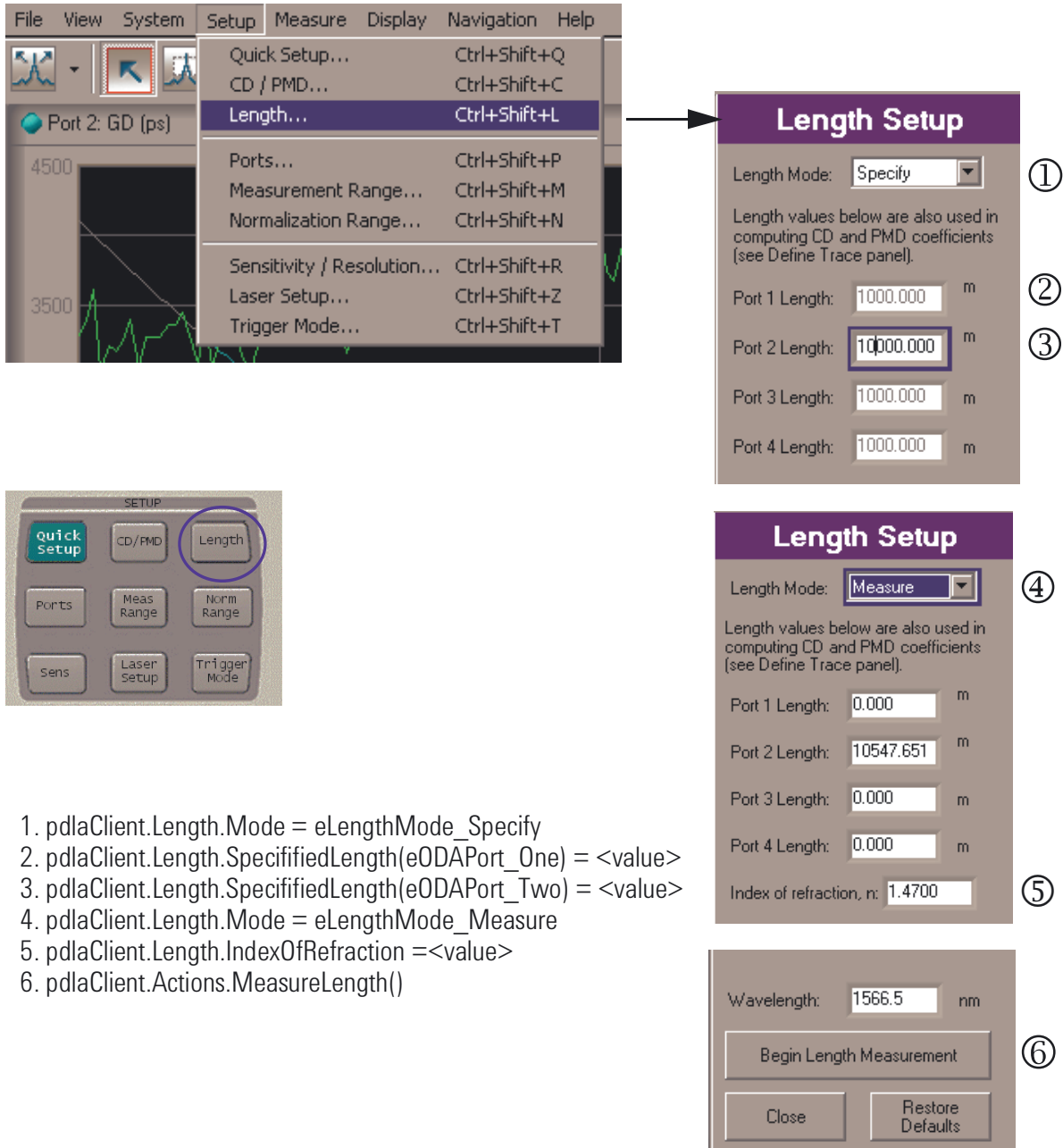


Figure 19 Front Panel Related Commands



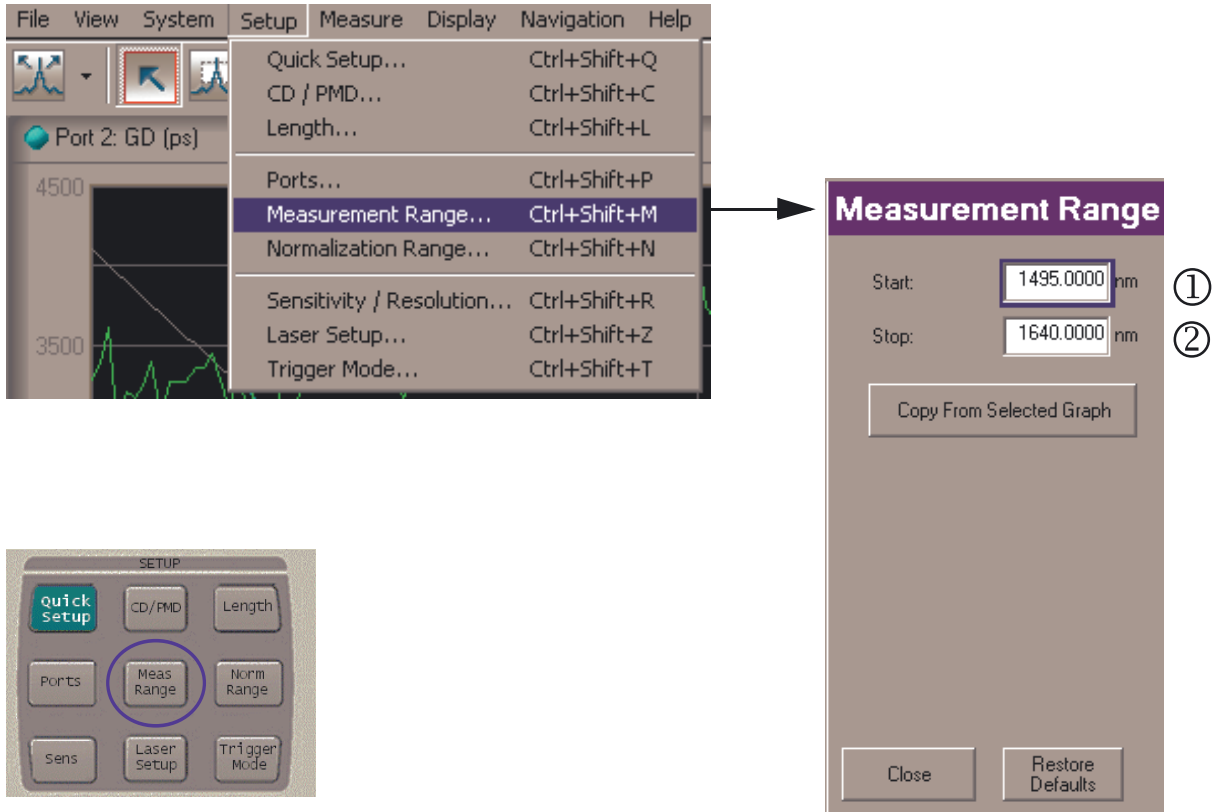
1. pdlaClient.DispersionMode=eDispersionMode_CD_Swept
2. pdlaClient.DispersionMode=eDispersionMode_CD_PMD_Swept
3. pdlaClient.DispersionMode=eDispersionMode_CD_Stepped
4. pdlaClient.DispersionMode=eDispersionMode_CD_PMD_Stepped
5. pdlaClient.NumPolStatesPMD=eNumPolStatesPMD_eSix
6. pdlaClient.GenerateMuellerAndPMDData=False
7. pdlaClient.UseExternalPowermeter=True
8. pdlaClient.DUTMinLossMaxGain=eDUTLossGain_eGain_0dB

Figure 20 Measurement Type Related Commands



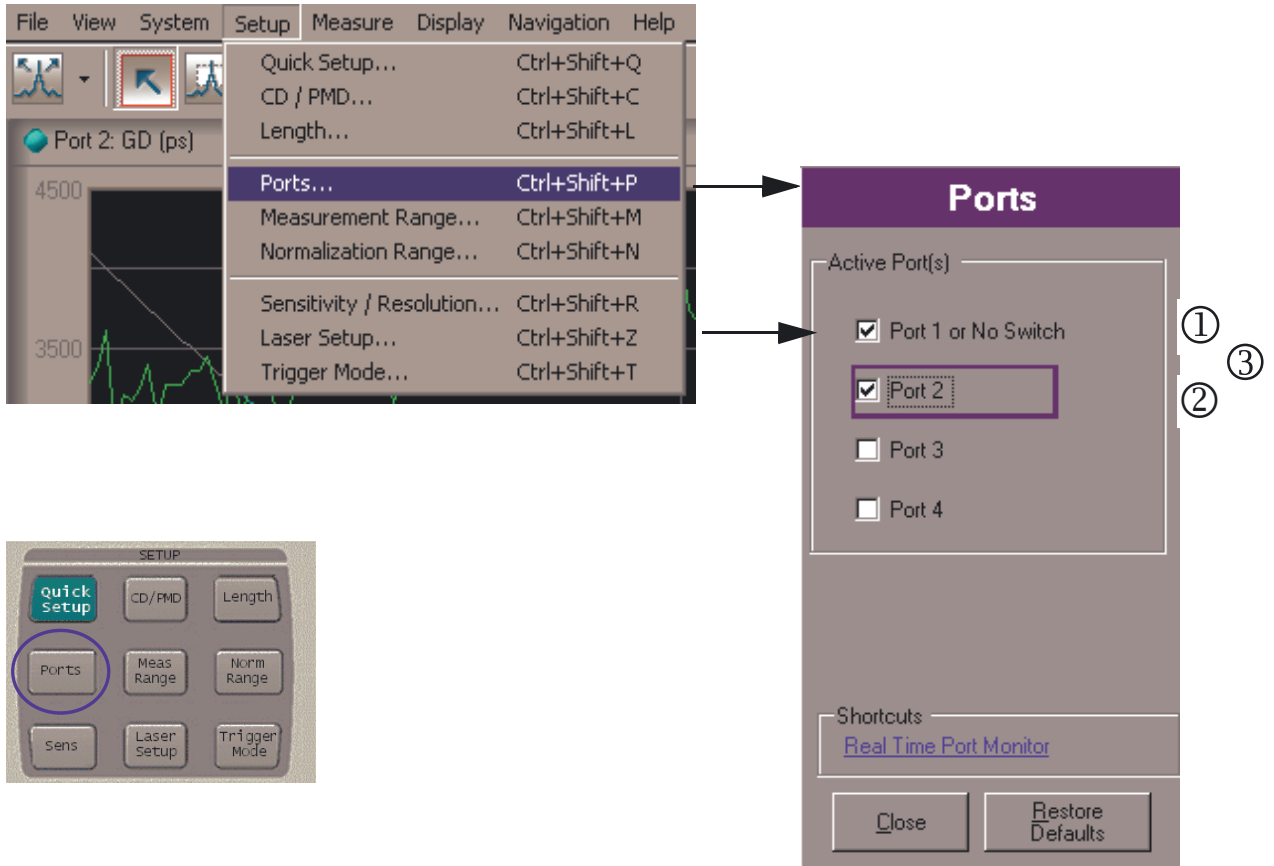
1. `pdlaClient.Length.Mode = eLengthMode_Specify`
2. `pdlaClient.Length.SpecifiedLength(eODAPort_One) = <value>`
3. `pdlaClient.Length.SpecifiedLength(eODAPort_Two) = <value>`
4. `pdlaClient.Length.Mode = eLengthMode_Measure`
5. `pdlaClient.Length.IndexOfRefraction = <value>`
6. `pdlaClient.Actions.MeasureLength()`

Figure 21 Length Measurement Related Commands



1. `pdlaClient.MeasurementRange.XStart = <value>`
2. `pdlaClient.MeasurementRange.XStop = <value>`

Figure 22 Measurement Range Related Commands



1. `pdlaClient.Ports = eODAPort_One`
2. `pdlaClient.Ports = eODAPort_Two`
3. `pdlaClient.Ports = eODAPort_One + eODAPort_Two`

Figure 23 Port Selection Related Commands

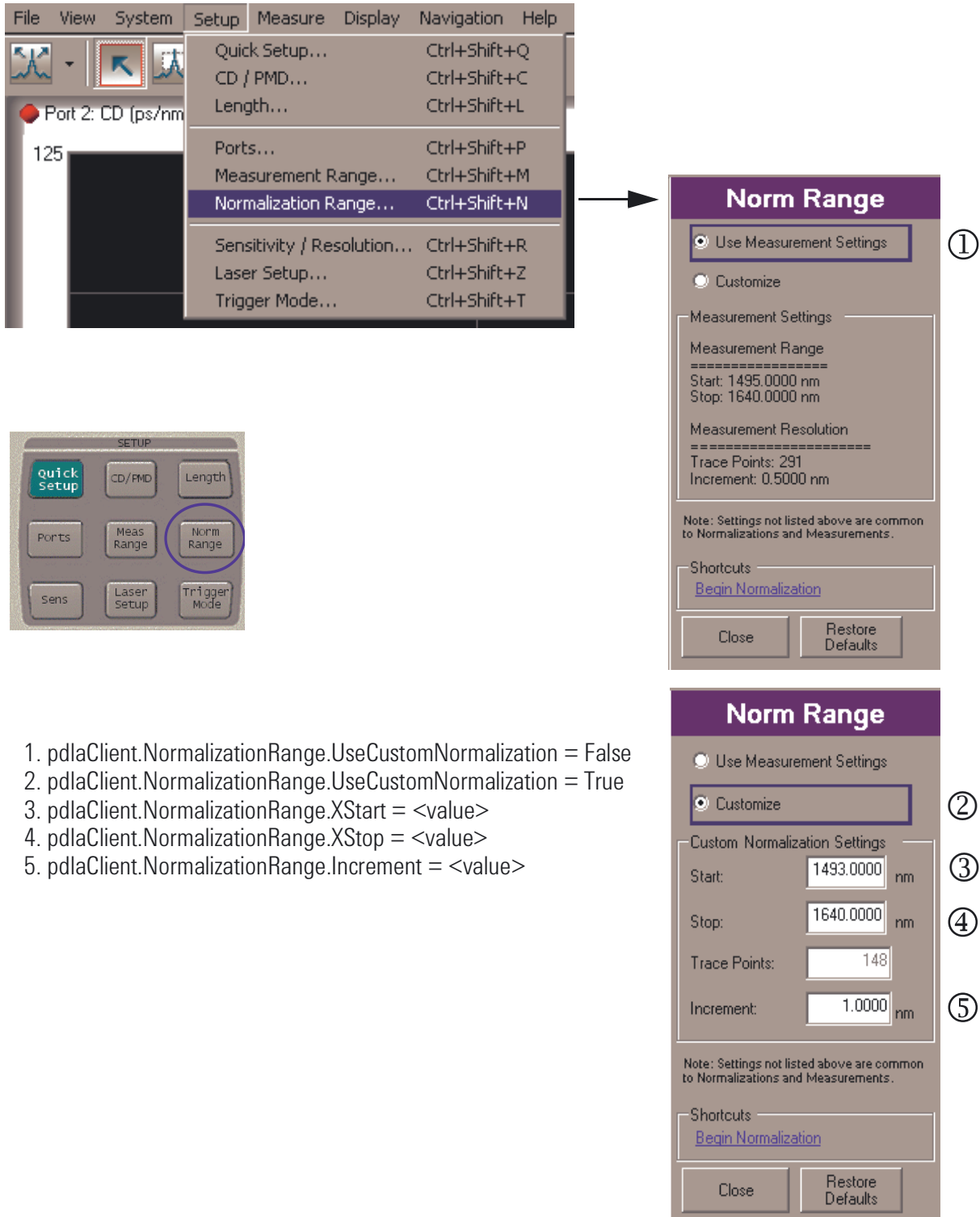
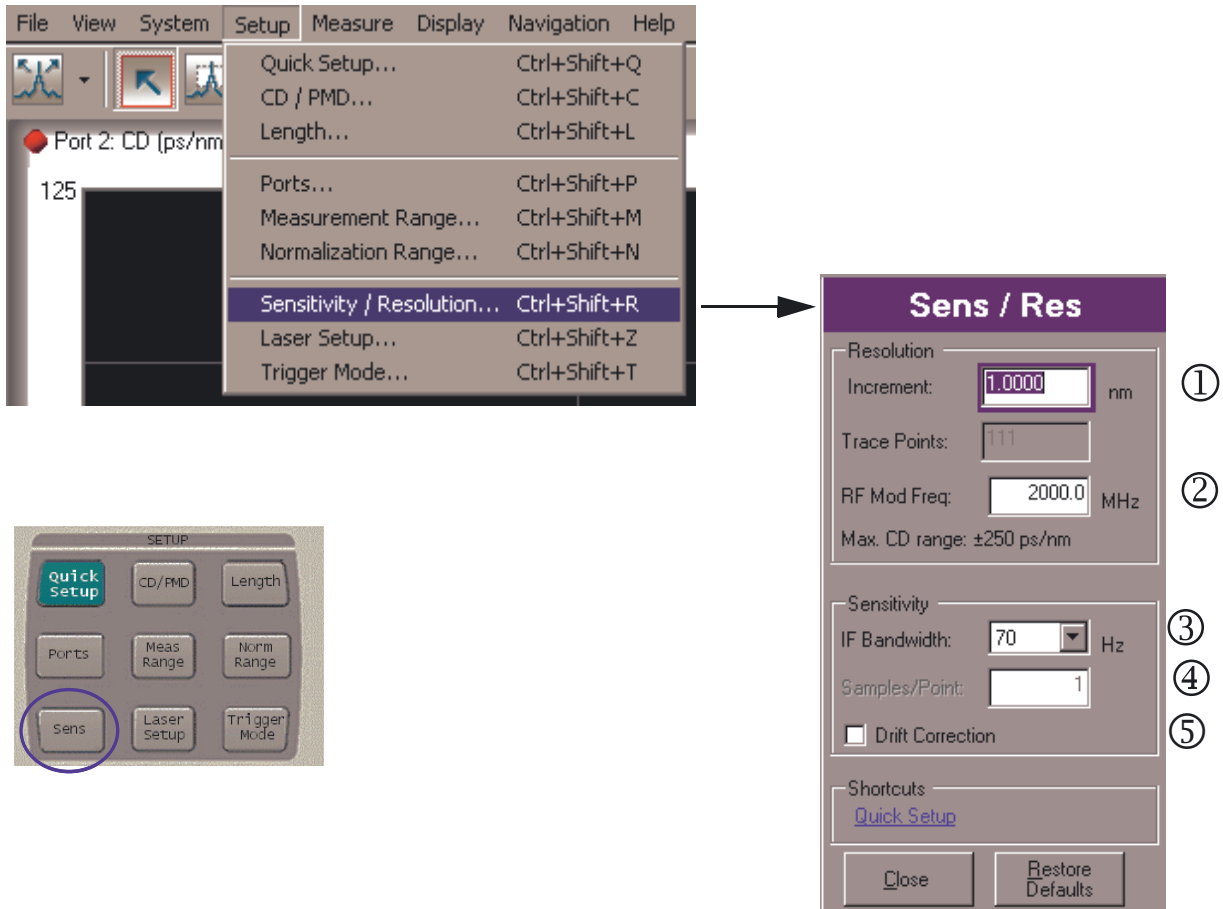
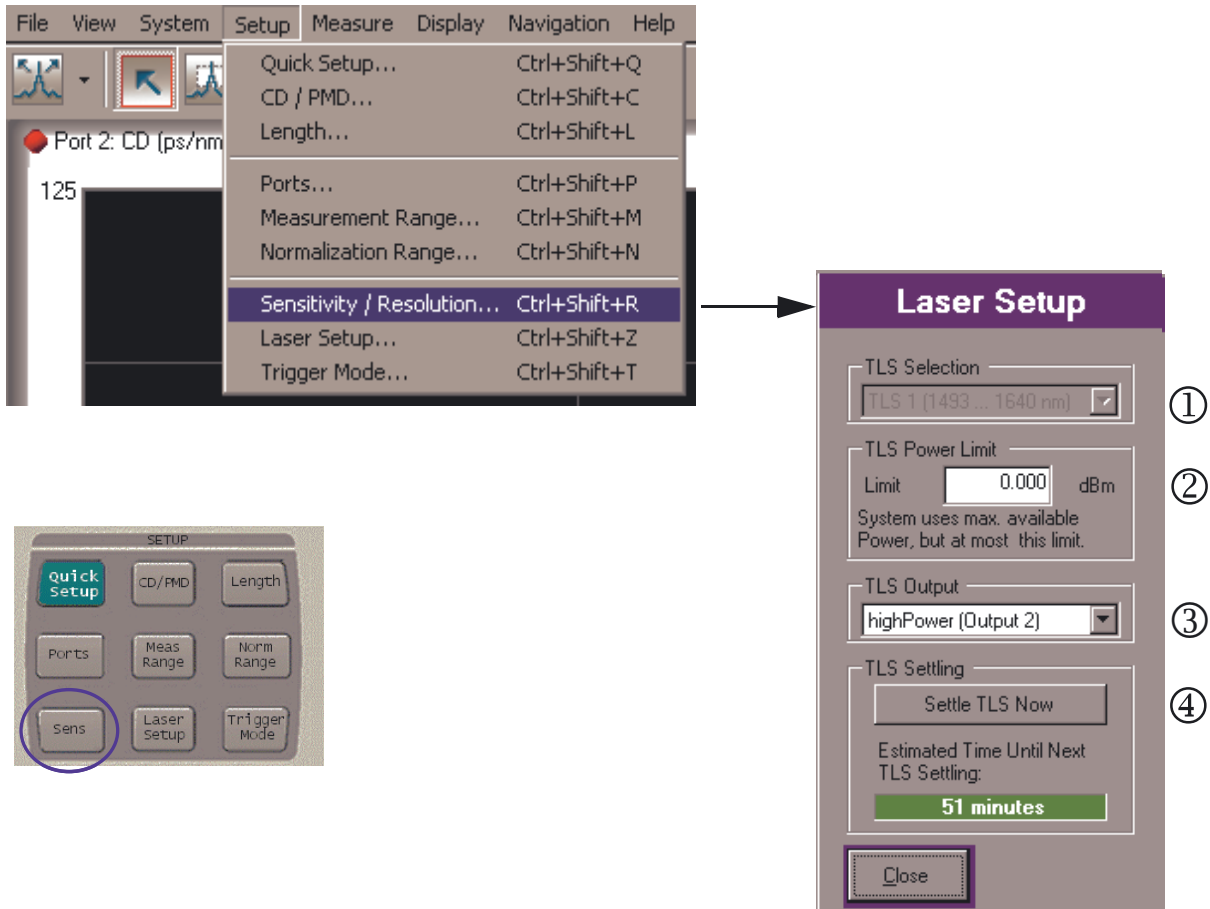


Figure 24 Normalization Range Related Commands



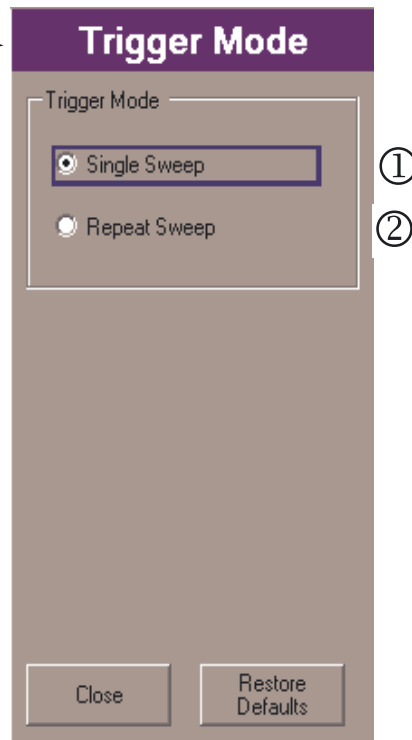
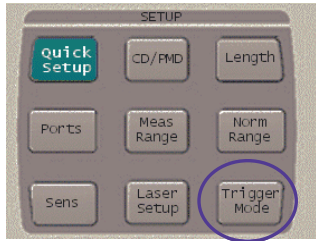
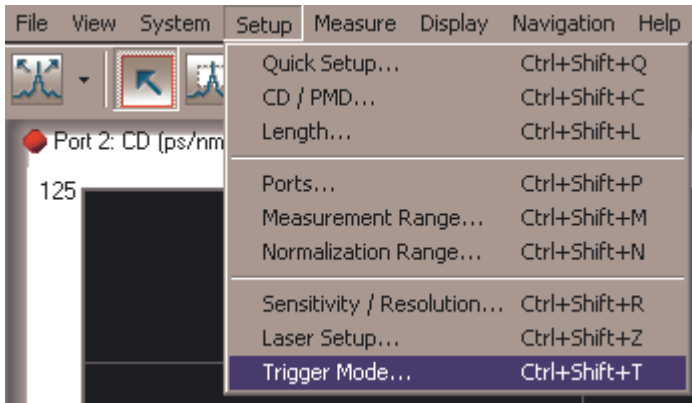
1. `pdlaClient.Resolution.Increment = <value>`
2. `pdlaClient.Resolution.RFModulationFrequency= <value>`
3. `pdlaClient.Sensitivity.IFBandwidth = eIFBandwidth_IFBW_70Hz`
4. `pdlaClient.Sensitivity.SamplesPerPoint = <value>`
5. `pdlaClient.Sensitivity.DriftCorrection = False`

Figure 25 Sensitivity and Resolution Related Commands



1. retVal = pdlaClient.TLS.SelectTLS(0)
2. pdlaClient.TLS.HighPowerLimit = <value>
3. pdlaClient.TLS.PowerSelect = eTLSPowerSelect_High
4. pdlaClient.TLS.ForceSetting

Figure 26 Laser Setup Related Commands



1. `pdlaClient.TriggerModeParams = eTriggerType_TriggerSingle`
2. `pdlaClient.TriggerModeParams = eTriggerType_TriggerContinuous`

Figure 27 Trigger Mode Related Commands

Example 1: Simple Application for Controlling the PDLA

This example demonstrates a simple visual application to control the PDLA. Refer to [“Recommended Control Sequence” on page 138](#) for the full recommended sequence. For simplicity, this application omits normalization, but covers the rest of the steps.

This example is presented in three different languages: VB6.0, VB.NET and C#. Go through the version for the language you are most likely to use.

The source code is included with the Remote Control Installation. Navigate down to C:\Program Files\Agilent\Agilent PDLA Remote Control\Examples

Starting a VB 6.0 PDLA Application

Install Software

Refer to “Recommended Control Sequence” on page 138.

Application Shell

The following steps show you how to build a simple shell to connect to the PDLA.

- 1 Run Visual Basic 6.0.
- 2 Double click on Standard Exe to open a new project. Use the default name, Project1.

This creates a project with a blank frame. To the left of the window is a tool bar with all the different controls you can put on the frame. Hovering the mouse over the icon will give a short description.

- 3 Click the Label icon (icon with capital “A”), then click and drag to create a box in the upper left of Form 1. On the right side of the window, there should be a dialog titled Properties. (If not, click View > Properties Window.) In the Properties dialog, scroll down to choose Caption and then type ODA Name. Press Enter and notice that PDLA Name appears on Form 1.
- 4 From the tool bar, click the Text Box icon, then click and drag to create a box in the upper right of Form 1. In the Properties dialog, click Name and type tbOdaName. Click on Text and type the network name of your PDLA as the default text. (Refer to the PDLA Help > About to find your PDLA network name.)
- 5 Click Project > References. This brings up a dialog that shows the selected DLL's that you can use in this project. Two DLL's are used to enable communication with the PDLA.
- 6 Click Browse, then navigate to
C:\Program Files\Agilent\Agilent PDLA Remote Control. Click on InstrumentObjects.tlb to add InstrumentObjects to the list, then click Open.
- 7 Click Browse again and navigate to RemoteClient.tlb in the same directory.
- 8 Select Open, which takes you back to the References dialog.

Note that the name of the Remote Client might appear as a long string "This is the RemoteClient for the Photonic Dispersion and Loss Analyzer." Click OK to close the References dialog.

- 9 Right click on the Form and select View Code. This brings up a simple Form.vb Code window, which is probably empty except for Option Explicit. Add the pdlaClient object as a member right after that line:

```
Private WithEvents pdlaClient As RemoteClient.Communicator
```

The pdlaClient will host all the interaction with the PDLA. Before you can use it, though, you need to 'new' the object. Go back to the Form view, and double click on an unused area of the form. This takes you to the Form1_Load routine, where you can insert the following line directly above End Sub:

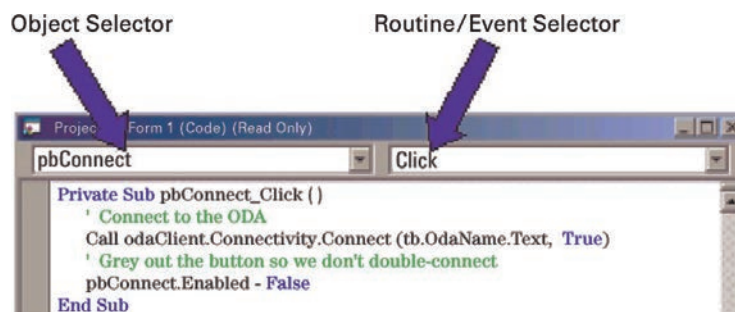
```
Set pdlaClient = New RemoteClient.Communicator
```

- 10 Before you can talk to the PDLA, you have to establish a link to it. From the tool bar, place a Button on the form. From the Properties dialog, Name the button pbConnect with Caption Connect. Double click the button to edit the source code. This will place you at pbConnect_Click. The connection to the PDLA is established here. Place the following lines above the final End Sub:

```
Call pdlaClient.Connectivity.Connect(tbPdlaname.Text)
pbConnect.Enabled = False
```

You need to implement a disconnect method. This avoids a time-consuming clean up when a connection goes stale and has to be cleaned out of the PDLA interface.

The second line grays out the Connect button so you do not forget and connect twice.



- 11 Go to the code window, and click the drop down list box at the upper left corner of the Code window. It probably says pbConnect, since you were just editing the connect method. Change it to Form. Now the list box at the upper right corner

lists the various events you can write code for. It probably says Load. Click on the selections, and choose Unload. This will take you to Form_Unload, where you can insert the following lines:

```
pdlaClient.Connectivity.Disconnect  
pdlaClient.Connectivity.TerminateSession
```

The first line disconnects from the PDLA itself, the second line cleans up the pdlaClient. Make sure that TerminateSession is the last thing you do before exiting this routine.

- 12 Click the Start button from the tool bar located at the top of the Visual Basic application (or Run > Start) to compile the application. If there are errors, fix them before continuing. Once the compile is successful, the application window will appear. Click the Connect button, and watch the PDLA front panel. A green banner will appear on the PDLA, announcing Remote User Connected. If this does not appear, it is most likely because the PDLA Name is incorrect, or the PDLA itself is not properly configured for networking (check your LAN connection). Go to the PDLA front panel and select Help >About. The name is on the About panel.

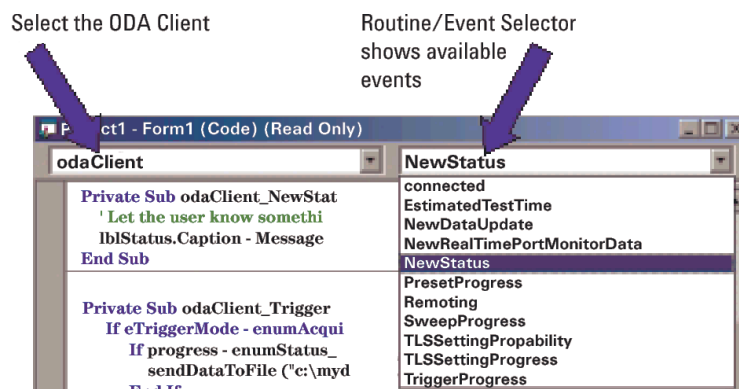
When you exit the program, it will disconnect from the PDLA, leaving it free for front panel or other remote use.

Event Handling

Event handling is added to monitor the PDLA events. A very useful event is a new status message. The status message is shown on the bottom of the PDLA display. It announces when a sweep has begun, finished, and so on. By reflecting those messages on the remote application, you provide invaluable feedback to the user, particularly if they cannot see the PDLA display.

When an event like a new status message occurs, the PDLA informs you by calling a method on your object. The Visual Basic editor knows what events you can receive, and handles all the interaction with the PDLA itself. Visual Basic presents a list of available events in the function list. When you select the event, it creates the function to hold your event handling code.

In **Step 9** above, the `pdlaClient` declared used the keyword “`WithEvents`”. This instructs Visual Basic to find the events and make them available to you. Go into the code window and click on the selector in the upper left corner (it probably says `Form`). It presents a list of objects you have available. Select the `pdlaClient`. Now click on the selector in the upper right corner (it probably says `SweepProgress`.) You get a list of events you can choose to receive and write code for. If the event is in light gray, this indicates that you have not implemented it yet. If it is in bold type face, there is code for it. You do not need to implement all the events, just the ones you find useful.



The steps below walk through adding an event handler for `NewStatus`.

- 1 Add a display line to show user status messages. Add a text label as wide as the form, and call it `lblStatus`. Default the caption to "PDLA Status."
- 2 Create the event function. Go to the code window. Click the object selector in the upper left corner of the window, and select `pdlaClient`. Now click the event method selector in the upper right corner of the display, and select `NewStatus`. This will take you to a function called `pdlaClient_NewStatus`, that will be called when a new status message is displayed on the PDLA. The first pass parameter is the message itself, the second pass parameter is an enumerated type telling you if the message was an error, informational, or a warning. To display the status, insert the following line inside `pdlaClient_NewStatus`:


```
lblStatus.Caption = Message
```
- 3 Click the Run button, and connect to the PDLA. The first message you will see is the Client connected message. Press Local on the PDLA and take a sweep. You will see the progress messages appearing on the application as the

measurement progresses. When you exit the application, it will release the PDLA.

All of the PDLA events follow a similar pattern. The most useful events are `TriggerProgress`, which tells you when a measurement or normalization is complete, and `TLSSettlingProbability`, which warns you how long until the laser needs to perform a temperature settling.

Take Sweep and Transfer Data

In this exercise, you will trigger a sweep from your application, and when the sweep completes, transfer the data across and save it in a file. This consists of adding a button to trigger the sweep, and adding an event handler for the `TriggerProgress` event, to detect when the sweep is finished and save the data in `C:\mydata.txt`.

- 1 Add a button to the UI and Name it `pbSweep`, Caption `Take Sweep`.
- 2 Double click the button to go to the `pbSweep_Click` function, and insert the following lines to make sure you are connected before actually taking a measurement:

```
' Don't do anything if not connected to PDLA
If pdlaClient.Connectivity.IsConnected = False Then
  MsgBox ("Can't take a measurement yet - not connected.")
  Exit Sub
End If
pdlaClient.Measure
```

- 3 Run the software, click `Connect` and wait for the connection to complete. Then click `Take Sweep`, and watch the sweep progress on both the PDLA and on the application message bar.
- 4 Add an event routine for `TriggerProgress`.

The `TriggerProgress` event has two pass parameters - `enumStatus` and `enumAcquisitionMode`. This indicates the status and progress of the current activity.

This function checks the `enumAcquisitionMode` to determine the type of trigger event (that is, measurement, normalization, TLS settling and so on) and then checks the `enumStatus` to see if the event is finished, indicating data is available. If so, call `sendDataToFile()`.

In the Source Code window, select `pdlaClient` in the Object Selector in the upper left hand corner of the window, then select `TriggerProgress` in the Method Selector in the upper right hand corner. This takes you to the sub

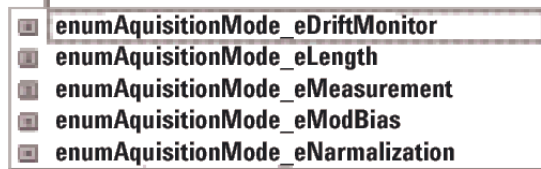
pdlaClient_TriggerProgress, with two pass parameters already declared. Progress is an enumerated type which tells you whether you are at the begin or end, and eTriggerMode, which tells you whether the current activity is a measurement or a normalization.

Click on the next line after the sub declaration, and type:

```
If eTriggerMode =
```

When you type the '=' (equals sign), the Visual Basic editor will Intellisense all the possible values of the enumeration, and present you a list. In this case you want eMeasurement. Intellisense makes dealing with events and enumerated types much easier in Visual Basic.

```
Private Sub odaClient_TriggerProgress (ByVal progress  
If eTriggerMode =
```



Fill out the function so it looks like:

```
If eTriggerMode = enumAcquisitionMode_eMeasurement Then  
  If progress = enumStatus_COMPLETE Then  
    sendDataToFile ("c:\mydata.txt")  
  End If  
End If
```

- 5 Add the sendDataToFile function. This is a very simple implementation using text streams. Insert this function immediately after pdlaClient_TriggerProgress:

```
Private Sub sendDataToFile(ByVal fileName As String)  
  Dim gainData() As Double  
  Dim xStart As Double  
  Dim xStop As Double  
  Dim xStep As Double  
  Dim npoin As Integer  
  Dim cnt As Integer  
  Dim idx As Integer  
  Dim buf As String  
  
  ' Get the data and figure out the wavelength axis  
  gainData = pdlaClient.Results.YData_  
  (eMeasurementType_Gain,eODAPort_One)  
  
  npoin = UBound(gainData) - LBound(gainData)+1  
  
  xStart = pdlaClient.Results.xStart(eMeasurementType_Gain)  
  xStop = pdlaClient.Results.xStop(eMeasurementType_Gain)
```

```

xStep = (xStop - xStart) / (npoint - 1)

' Open the file and write the data
Dim outputFileNum As Integer
outputFileNum = 123

    Open fileName For Output As outputFileNum
    Write #outputFileNum, "Sample PDLA File written from VB 6.0"
Write #outputFileNum, Now
Write #outputFileNum, "Wavelength (nm)", "Gain (dB)"

idx = 0
For cnt = LBound(gainData) To UBound(gainData)
    buf = CStr(xStart + xStep * idx) + "," + CStr(gainData(cnt))
    Write #outputFileNum, xStart + xStep * idx, gainData(cnt)
    idx = idx + 1
Next
Close #outputFileNum
End Sub

```

- 6 Run the software, connect and take a sweep. Then look for c:\mydata.txt. You can look at it with a text editor or import it into a spreadsheet.

Tip: This program will grab data even if a sweep is triggered from the front panel. Run the software, and then connect. Go to the PDLA front panel and click Local, then take a sweep. You will see the update messages appear on the application, and the application will detect end-of-sweep and transfer the data.

- 7 Exit the application when completed.

Control Center Wavelength

This section shows how to control a simple parameter on the PDLA. This example will use the center wavelength. A text box is added for the user to enter the desired value, and then send that value to the PDLA as part of the sweep.

Changing a parameter consists of determining which property on which sub-object of `pdlClient` to modify, then modifying it.

- 1 Add a text field, name it `txtCenter`, set default text to 1550. Then add a label that says "Center Wavelength, nm: ".
- 2 Modify the take sweep function to send the value to the PDLA. Find the function `pbSweep_click`, and add the following lines to check the value and send it to the PDLA. The PDLA thinks in terms of start/stop wavelength, so the center wavelength must be converted to a start and stop

wavelength. Find the pbSweep_click method, and insert the following code immediately before pdlaClient.Measure:

```
Dim centerWavel, startWavel, stopWavel, span, oldCenter As Double
```

```
startWavel = pdlaClient.MeasurementRange.XStart  
stopWavel = pdlaClient.MeasurementRange.XStop  
span = stopWavel - startWavel
```

```
centerWavel = CDb(txtCenter.text)  
If centerWavel < 1400 Or centerWavel > 1700 Then  
    MessageBox.Show("Wavelength out of range, defaulted to 1550")  
    centerWavel = 1550  
    txtCenter.Text = "1550"  
End If
```

```
pdlaClient.MeasurementRange.XStart = centerWavel - span / 2  
pdlaClient.MeasurementRange.XStop = centerWavel + span / 2
```

It works better if you first set the PDLA to a relatively narrow sweep, such as 1530 to 1540 nm.

Complete source code

```
Option Explicit
```

```
' Declare the pdlaClient.  
Private WithEvents pdlaClient As RemoteClient.Communicator
```

```
Private Sub Form_Load()  
    ' Instantiate the pdlaClient  
    Set pdlaClient = New RemoteClient.Communicator  
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)  
    ' Disconnect from PDLA and clean up on our side  
    pdlaClient.Connectivity.Disconnect  
    pdlaClient.Connectivity.TerminateSession  
End Sub
```

```
Private Sub pdlaClient_NewStatus(ByVal Message As String, ByVal MessageType As  
InstrumentObjects.eEventLogType)  
    ' Let the user know something happened  
    lblStatus.Caption = Message  
End Sub
```

```
Private Sub pdlaClient_TriggerProgress(ByVal progress As  
InstrumentObjects.enumStatus, ByVal eTriggerMode As  
InstrumentObjects.enumAcquisitionMode)  
    If eTriggerMode = enumAcquisitionMode_eMeasurement Then  
        If progress = enumStatus_COMPLETE Then  
            sendDataToFile ("c:\mydata.txt")  
        End If  
    End If  
End Sub  
Private Sub sendDataToFile(ByVal fileName As String)
```

```

Dim gainData() As Double
Dim xStart As Double
Dim xStop As Double
Dim xStep As Double
Dim npoin As Integer
Dim cnt As Integer
Dim idx As Integer
Dim buf As String

' Get the data and figure out the wavelength axis
gainData = pdlaClient.Results.YData_
(eMeasurementType_Gain,eODAPort_One)

npoin = UBound(gainData) - LBound(gainData)+1

xStart = pdlaClient.Results.xStart(eMeasurementType_Gain)
xStop = pdlaClient.Results.xStop(eMeasurementType_Gain)

xStep = (xStop - xStart) / (npoin - 1)

' Open the file and write the data
Dim outputFileNum As Integer
outputFileNum = 123

    Open fileName For Output As outputFileNum
    Write #outputFileNum, "Sample PDLA File written from VB 6.0"
Write #outputFileNum, Now
Write #outputFileNum, "Wavelength (nm)", "Gain (dB)"

idx = 0
For cnt = LBound(gainData) To UBound(gainData)
    buf = CStr(xStart + xStep * idx) + "," + CStr(gainData(cnt))
    Write #outputFileNum, xStart + xStep * idx, gainData(cnt)
    idx = idx + 1
Next
Close #outputFileNum
End Sub

Private Sub pbConnect_Click()
' Connect to the PDLA
Call pdlaClient.Connectivity.Connect(tbPdlName.Text)
' Grey out the button so we don't double-connect
pbConnect.Enabled = False
End Sub

Private Sub pbSweep_Click()
' Don't do anything if not connected to PDLA
If pdlaClient.Connectivity.IsConnected = False Then
    MsgBox ("Can't take a measurement yet - not connected.")
Exit Sub
End If

' Set the start wavelength
Dim centerWavel, startWavel, stopWavel, span, oldCenter As Double

startWavel = pdlaClient.MeasurementRange.xStart

```



```
stopWavel = pdlaClient.MeasurementRange.xStop
span = stopWavel - startWavel

centerWavel = CDb1(txtCenter.Text)
If centerWavel < 1400 Or centerWavel > 1700 Then
    MsgBox ("Wavelength out of range, defaulted to 1550")
    centerWavel = 1550
    txtCenter.Text = "1550"
End If

pdlaClient.MeasurementRange.xStart = centerWavel - span / 2
pdlaClient.MeasurementRange.xStop = centerWavel + span / 2

' Trigger the measurement
pdlaClient.Measure

End Sub
```

Starting a VB.NET PDLA Application

Install Software

Refer to “Recommended Control Sequence” on page 138.

Application Shell

The following steps show you how to build a simple shell to connect to the PDLA.

- 1 Run Visual Studio .NET.
- 2 Open a new project and save it to a local directory. Use the default file name of WindowsApplication1.

This creates a project with a blank frame. To add controls to the frame, click View > Toolbox (or click the icon on the toolbar.)

- 3 From the toolbox, add a label with a Caption “PDLA Name:” and a text box named tbPdlName. In the Properties dialog, click on Text and type the network name of your PDLA as the default text.(Refer to PDLA Help > About to find your PDLA network name.
- 4 From the main menu, click Project > Add Reference. This brings up a dialog that defaults to the .NET tab. You will be adding the .NET incarnation of the PDLA.
- 5 Select Browse, then navigate to C:\Program Files\Agilent\Agilent PDLA Remote Control.
- 6 Click InstrumentObjects.dll, then ctrl-click on RemoteClient.dll (you can add both at once.) Select Open, which brings you back to the References dialog. You will see the two dll’s in the Selected Components pane. Select OK, and InstrumentObjects and RemoteClient will appear in the Solution Explorer on the right side of the Visual Studio window.
- 7 Right click on the form and select View Code. This brings up a simple From1.vb code window. Find the line that reads: Inherits System.Windows.Forms.Form. and add right after that line the pdla client object as a member:

```
Private WithEvents pdlaClient As RemoteClient.Communicator
```

The pdlaClient will host all the interaction with the PDLA. Before you can use it, though, you have to 'new' the object. Go back to the form view, and double click on an unused area of the form. This takes you to the Form1_Load routine, where you can insert the following line:

```
pdlaClient = New RemoteClient.Communicator()
```

- 8 Before you can interface with the PDLA, you need to establish a link to it. Place a Button on the form, name it pbConnect with text Connect. Double click the button to edit the source code. This will place you at pbConnect_Click. To establish the connection to the PDLA, place the following lines between the Private Sub ... End Sub block:

```
pdlaClient.Connectivity.Connect(tbPdlaNam.Text)
```

```
pbConnect.Enabled = False
```

You need to implement a disconnect method, since whenever you connect, you need to disconnect when you are done. This avoids a time-consuming clean up when a connection goes stale and has to be cleaned out of the PDLA interface.

The second line grays out the Connect button so you do not forget and connect twice.

- 9 Go to the code window, and click the selector at the upper left corner of the code window. It probably says Form1. Change it to say (Base Class Events). Now the selector at the upper left corner lists the various events we can write code for. It probably says (Declarations). Click on the selections, and choose Closing. This will take you to Form1_Closing, where you can insert the following lines:

```
pdlaClient.Connectivity.Disconnect()
```

```
pdlaClient.Connectivity.TerminateSession()
```

The first line disconnects from the PDLA itself, the second line cleans up the pdlaClient. Make sure that TerminateSession is the last thing you do before exiting this routine.

- 10 Click the Run button to compile the application. If there are errors, fix them before continuing. Once the compile is successful, the application window will appear. Click the Connect button, and watch the PDLA front panel. A green banner will appear on the PDLA, announcing Remote User Connected. If this does not appear, it is most likely because the PDLA Name is incorrect, or the PDLA itself is not properly configured for networking (check the LAN connection). Go to the PDLA front panel and select Help > About to verify the PDLA Name.

When the program is exited, the remote session will be disconnected from the PDLA leaving it free for front panel or other remote use.

Event Handling

Event handling is added to monitor the PDLA events. A very useful event is a new status message. The status message is shown on the bottom of the PDLA display, which indicates when a sweep has begun, sweep ended, and so on. By reflecting those messages on the remote application, you provide invaluable feedback to a user, particularly if they cannot see the PDLA display.

When an event like a new status message occurs, the PDLA informs you by calling a method on your object. Visual Basic presents a list of available events in the function list. When you select the event, it creates the function to hold your event handling code.

In [Step 7](#) above, the `pdlClient` was declared using the keyword “`WithEvents`”. This instructs Visual Basic to find the events and make them available to you. Go into the code window and click on the selector in the upper left corner (it probably says `Form1` now.) It presents a list of objects you have available. Select the `pdlClient`. Now click on the selector in the upper right corner (it probably says `Declarations`). It presents a list of events you can choose to receive and write code for. If the event is in light gray, you have not implemented it yet. If it is in bold type face, there is code for it. You do not have to implement all the events, just the ones you find useful.

The steps below walk through adding an event handler for `NewStatus`.

- 1 Add a display line to show user status messages. Add a text label as wide as the form, and call it `lblStatus`. Default the text to `PDLA Status`.
- 2 Create the event function. Go to the code window. Click the object selector in the upper left corner of the window, and select `pdlClient`. Now click the event method selector in the upper right corner of the display, and click `NewStatus`. This will take you to a function called `pdlClient_NewStatus`, which will be called when a new status message is displayed on the PDLA. The first pass parameter is the message itself, the second pass parameter is an enumerated type telling you

if the message was an error, informational or a warning. To display the status, insert the following line inside:

```
Private Sub pdlaClient_NewStatus( ...  
    ' Let the user know something happened  
    lblStatus.Text = Message  
End Sub
```

- 3 Click the Run button, and connect to the PDLA. The first message you will see is the Connected message. Press Local on the PDLA and take a sweep. You will see the progress messages appearing on the application as the measurement progresses. When you exit the application, it will release the PDLA.

All of the PDLA events follow a similar pattern. The most useful events are TriggerProgress, which tells you when a measurement or normalization is complete, and TLSSettlingProbability, which warns you how long until the laser needs to perform a temperature settling.

Take Sweep and Transfer Data

In this exercise, we will trigger a sweep from our application, and when the sweep completes, we will transfer the data across and save it in a file. This consists of adding a button to trigger the sweep, and adding an event handler for the TriggerProgress event, so we can detect when the sweep is finished and save the data in C:\mydata.txt.

- 1 Add a button to the UI and name it pbSweep, with a caption of Take Sweep.
- 2 Double click the button to go to the pbSweep_Click function, and insert the following lines to make sure you are connected before actually taking a measurement:

```
' Don't do anything if not connected to PDLA  
If pdlaClient.Connectivity.IsConnected = False Then  
    MessageBox.Show("Can't take a measurement yet - not connected.")  
    Return  
End If  
pdlaClient.Measure()
```

- 3 Run the software, click Connect and wait for the connection to complete. Click Take Sweep and watch the sweep progress on both the PDLA and on the application message bar.
- 4 Add an event routine for TriggerProgress.

The TriggerProgress event has two pass parameters – enumStatus and enumAcquisitionMode. This indicates the status and progress of the current activity.

This function checks the enumAcquisitionMode to determine the type of trigger event (that is, measurement, normalization, and so on) and then checks the enumStatus to see if the measurement is finished, indicating data is available. If so, call sendDataToFile().

In the source code window, select pdlaClient in the Object Selector in the upper left corner of the window, then select TriggerProgress in the Method Selector in the upper right corner. This takes you to the sub pdlaClient_TriggerProgress, with two pass parameters already declared. Progress is an enumerated type which tells you whether you are at the begin or end, and eTriggerMode, which tells you whether the current activity is a measurement or a normalization.

- 5 Click on the next line after the sub declaration, and type:

```
If eTriggerMode =
```

When you type the '=', the Visual Basic editor will Intellisense all the possible values of the enumeration, and present you a list. In this case we want eMeasurement. Intellisense makes dealing with events and enumerated types much easier in visual basic.

Fill out the function so it looks like:

```
' Only do something if this is a measurement
If eTriggerMode =
Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.enumAcquisitionMode.eMea
surement Then
' Only do something if the measurement is complete
If progress =
Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.enumStatus.COMPLETE Then
' Write the data to disk
sendDataToFile("c:\mydata.txt")
End If
End If
```

- 6 Add the sendDataToFile function. This is a very simple implementation using text streams. Insert this function immediately after pdlaClient_TriggerProgress:

```
Private Sub sendDataToFile(ByVal fileName As String)
Dim gainData() As Double
Dim xStart As Double
Dim xStop As Double
Dim xStep As Double
Dim npoin As Integer
Dim cnt As Integer
Dim idx As Integer
Dim buf As String

' Get the data and figure out the wavelength axis
gainData = pdlaClient.Results.YData _
(Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eMeasurementType.Gain, _
```

```

Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eODAPort.One)
npoin = UBound(gainData) - LBound(gainData)

xStart = pdlaClient.Results.XStart_
(Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eMeasurementType.Gain)
xStop = pdlaClient.Results.XStop_
(Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eMeasurementType.Gain)

xStep = (xStop - xStart) / (npoin - 1)

' Open the file and write the data
Dim file As System.IO.StreamWriter
file = New System.IO.StreamWriter(fileName)
file.WriteLine _
("Sample PDLA File, written from Visual Basic.net")
file.WriteLine("Wavelength (nm), Gain (dB)")

idx = 0
For cnt = LBound(gainData) To UBound(gainData)
    buf = CStr(xStart + xStep * idx) + "," + CStr(gainData(cnt))
    file.WriteLine(buf)
    idx = idx + 1
Next
file.Close()
End Sub

```

- 7 Run the software, connect and take a sweep. Then look for c:\mydata.txt. You can look at it with a text editor or import it into a spreadsheet.

This program will grab data even if a sweep is triggered from the front panel. Run the software, and then connect. Go to the PDLA front panel and click Local, then take a sweep. You will see the update messages appear on our application, and the application will detect end-of-sweep and transfer the data.

- 8 Exit the application when you are done.

Control Center Wavelength

This section shows how to control the center wavelength on the PDLA. First, a text box is added for the user to enter the desired value, and then the entered value will be sent to the PDLA as part of the sweep.

- 1 Add a text field, name it txtCenter, set the default text to 1550. Add a label with the text, Center Wavelength, nm.
- 2 Modify the take sweep function to send the value to the PDLA. Find the function pbSweep_click, and add the following lines to check the value and send it to the PDLA. The PDLA thinks in terms of start/stop, so the center

wavelength must be converted to a start and stop wavelength. Find the pbSweep_click method, and insert the following code immediately before pdlaClient.measure:

```
Dim centerWavel, startWavel, stopWavel, span, oldCenter As Double
```

```
startWavel = pdlaClient.MeasurementRange.XStart  
stopWavel = pdlaClient.MeasurementRange.XStop  
span = stopWavel - startWavel
```

```
centerWavel = CDb1(txtCenter.text)  
If centerWavel < 1400 Or centerWavel > 1700 Then  
    MessageBox.Show("Wavelength out of range, defaulted to  
1550")  
    centerWavel = 1550  
    txtCenter.Text = "1550"  
End If
```

```
pdlaClient.MeasurementRange.XStart = centerWavel - span / 2  
pdlaClient.MeasurementRange.XStop = centerWavel + span / 2
```

- 3 It works better if you first set the PDLA to a relatively narrow sweep, such as 1530 to 1540 nm.

Complete Source Code

```
Public Class Form1  
    Inherits System.Windows.Forms.Form  
    Private WithEvents pdlaClient As RemoteClient.Communicator  
    ' Windows form generated code omitted  
  
    Private Sub Form1_Load(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles MyBase.Load  
        pdlaClient = New RemoteClient.Communicator()  
    End Sub  
  
    Private Sub pbConnect_Click(ByVal sender As System.Object, ByVal e As  
System.EventArgs) Handles pbConnect.Click  
        pdlaClient.Connectivity.Connect(tbPdlaname.Text)  
        pbConnect.Enabled = False  
    End Sub  
  
    Private Sub Form1_Closing(ByVal sender As Object, ByVal e As  
System.ComponentModel.CancelEventArgs) Handles MyBase.Closing  
        pdlaClient.Connectivity.Disconnect()  
        pdlaClient.Connectivity.TerminateSession()  
    End Sub  
  
    Private Sub pdlaClient_NewStatus(ByVal Message As String, ByVal MessageType As  
Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eEventLogType) Handles  
pdlaClient.NewStatus  
        ' Let the user know something happened  
        lblStatus.Text = Message  
    End Sub
```



```

Private Sub pbSweep_Click(ByVal sender As System.Object, ByVal e As
System.EventArgs) Handles pbSweep.Click
    ' Don't do anything if not connected to PDLA
    If pdlaClient.Connectivity.IsConnected = False Then
        MessageBox.Show("Can't take a measurement yet - not connected.")
        Return
    End If

    ' Change the center wavelength. Read existing start and stop wavelength to get
    ' span, use new center to recalculate start and stop
    Dim centerWavel, startWavel, stopWavel, span, oldCenter As Double

    startWavel = pdlaClient.MeasurementRange.XStart
    stopWavel = pdlaClient.MeasurementRange.XStop
    span = stopWavel - startWavel

    centerWavel = CDb(txtCenter.text)
    If centerWavel < 1400 Or centerWavel > 1700 Then
        MessageBox.Show("Wavelength out of range, defaulted to 1550")
        centerWavel = 1550
        txtCenter.Text = "1550"
    End If

    pdlaClient.MeasurementRange.XStart = centerWavel - span / 2
    pdlaClient.MeasurementRange.XStop = centerWavel + span / 2

    ' Now trigger the measurement
    pdlaClient.Measure()
End Sub

Private Sub pdlaClient_TriggerProgress(ByVal progress As
Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.enumStatus, ByVal
eTriggerMode As
Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.enumAcquisitionMode)
Handles pdlaClient.TriggerProgress
    ' Only do something if this is a measurement
    If eTriggerMode =
Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.enumAcquisitionMode.eMea
surement Then
        ' Only do something if the measurement is complete
        If progress =
Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.enumStatus.COMPLETE Then
            ' Write the data to disk
            sendDataToFile("c:\mydata.txt")
        End If
    End If
End Sub

Private Sub sendDataToFile(ByVal fileName As String)
    Dim gainData() As Double
    Dim xStart As Double
    Dim xStop As Double
    Dim xStep As Double
    Dim npoin As Integer
    Dim cnt As Integer
    Dim idx As Integer

```

```

Dim buf As String

' Get the data and figure out the wavelength axis
gainData =
pdlaClient.Results.YData(Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eMeasurementType.Gain, _
    Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eODAPort.One)
npoin = UBound(gainData) - LBound(gainData)

xStart =
pdlaClient.Results.XStart(Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eMeasurementType.Gain)
xStop =
pdlaClient.Results.XStop(Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eMeasurementType.Gain)

xStep = (xStop - xStart) / (npoin - 1)

' Open the file and write the data
Dim file As System.IO.StreamWriter
file = New System.IO.StreamWriter(fileName)
file.WriteLine("Sample PDLA File, written from Visual Basic.net")
file.WriteLine("Wavelength (nm), Gain (dB)")

idx = 0
For cnt = LBound(gainData) To UBound(gainData)
    buf = CStr(xStart + xStep * idx) + "," + CStr(gainData(cnt))
    file.WriteLine(buf)
    idx = idx + 1
Next
file.Close()
End Sub
End Class

```

Starting a C# PDLA Application

Install Software

Refer to “Recommended Control Sequence” on page 138.

Application Shell

The following steps show you how to build a simple shell to connect to the PDLA.

- 1 Run Visual Studio .NET.
- 2 Open a new C# project, a windows application. This creates a project with a blank frame.
- 3 To add controls to the frame, click View > Toolbox.
- 4 From the toolbox, add a label with a Caption “PDLA Name:” and a text box named tbPdlName.
- 5 In the Properties dialog, click on Text and type the network name of your PDLA as the default text. (Refer to PDLA Help > About to find the PDLA network name.)
- 6 From the main menu, click Project > Add Reference. This brings up a dialog that defaults to the .NET tab. You will be adding the .NET incarnation of the PDLA. Select Browse, then navigate to C:\Program Files\Agilent\Agilent PDLA Remote Control. Click on InstrumentObjects.dll, then ctrl-click on RemoteClient.dll (you can add both at once.) Click Open, which brings you back to the References dialog. You will see the two dll’s in the Selected Components pane. Click OK, and InstrumentObjects and RemoteClient will appear in the Solution Explorer on the right hand side of the Visual Studio window.
- 7 Controlling the PDLA involves using lots of constants such as eMeasurementType. The ‘using’ directive at the top of your C-sharp file lets you use the PDLA specific constants without fully qualifying them. At the head of your C-sharp source file (right click on Form and select View Code), locate the block of using statements. Add these at the end of the block:

```
using Agilent.LWD.Ag86038x.InstrumentObjects;  
using Agilent.LWD.Ag86038x;
```

- 8 In the same source file as above, locate the summary description for

Form1: public class Form1 ...

Find the block of private declarations of the form components. There should be a Text Box and a Label. You are going to add the pdla client object as a member. Insert the following line after the Text Box and Label declarations:

```
// Declare a local object to communicate with the PDLA
private RemoteClient.Communicator pdlaClient;
```

The pdlaClient will host all the interaction with the PDLA. Before you can use it, though, you have to 'new' the object. Go back to the Form view, and double click on an unused area of the form. This takes you to the Form1_Load routine, where you can insert the following line:

```
pdlaClient = new RemoteClient.Communicator();
```

- 9 Before you can talk to the PDLA, you have to establish a link to it. Place a Button on the form and name it pbConnect with text Connect. Double click the button to edit the source code. This will place you at pbConnect_Click. This is where the connection to the PDLA is established. Place the following lines inside the curly braces:

```
pdlaClient.Connectivity.Connect(tbPdlaName.Text);
pbConnect.Enabled = false;
```

You need to implement a disconnect method, since whenever you connect, you need to disconnect when you are done. This avoids a time-consuming clean up when a connection goes stale and has to be clean out of the PDLA interface.

The second line grays out the Connect button so we don't forget and connect twice.

- 10 Go to the Form display window, and click on a blank area of the form to select the form. Go to the Properties window, and select events (small yellow lightning bolt at top of form.) Find and double click on the closing event. This will take you to the Form1_Closing routine, where you can insert the following lines:

```
pdlaClient.Connectivity.Disconnect();
pdlaClient.Connectivity.TerminateSession();
```

The first line disconnects from the PDLA itself, the second line cleans up the pdlaClient.

- 11 Click the Run button to compile the application. If there are errors, fix them before continuing. Once the compile is successful, the application window will appear. Click the Connect button, and watch the PDLA front panel. A green banner will appear on the PDLA, announcing Remote User

Connected. If this does not appear, it is most likely because the PDLA Name is incorrect, or the PDLA itself is not properly configured for networking (check the LAN connection). Go to the PDLA front panel and select Help > About. The PDLA name is on the About panel.

When the program is exited, it will disconnect from the PDLA, leaving it free for front panel or other remote use.

Event Handling

Event handling is added to monitor the PDLA events. A very useful event is a new status message. The status message is shown on the bottom of the PDLA display, which indicates when a sweep has begun, ended, and so on. By reflecting those messages on the remote application, you provide invaluable feedback to a user, particularly if they cannot see the PDLA display.

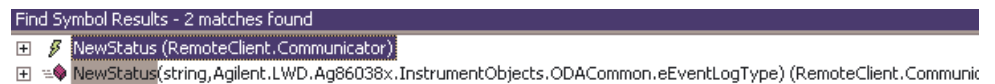
When an event, like a new status message occurs, the PDLA informs you by calling a method on your object. In fact, the PDLA will call as many methods as you care to register – it just keeps a list of functions to call, and runs through that list when the event occurs.

Implementing events is a little more complicated because Microsoft chose to use the Delegate mechanism as an abstraction layer between the event source (the PDLA) and the event consumer (your code.) So, instead of passing a reference to your function to the `pdlaClient`, you pass a reference to your function to a delegate, and then pass the delegate to the `pdlaClient`.

To reiterate, you define in your object a function that you wish to be called when an event occurs. That function has to take the pass parameters defined by that event – you don't get to choose. Call that function the event function. Once you have your event function, you instantiate a delegate for that function. The delegate you use depends on the pass parameters of the event function. If the event function takes the generic event pass parameters (an object reference and an `EventArgs`), you can use a generic delegate. In general, the event will have customized pass parameters and you must use the delegate class supplied

with the event. When a class implements a delegate, Microsoft recommends appending Handler to the name. Some people find this confusing, since in the past what we have called here the event function (the function you write that gets called when the event occurs) has also been called the event handler.

Now consider the specific example of a new status message. The pdlaClient has an event called NewStatus. The PDLA promises to fire it when a new status message appears. To take advantage of this, I write a routine called NewStatusEvent. Go into Visual Studio, and select View, Other Windows, Object Browser. Select the Find Symbol tool on the object browser (look for the binoculars.) Type in NewStatus and click Find. You get the result below:



The first hit (with the lightning bolt) represents the event itself. The second hit is the function prototype. NewStatus is expecting a function that accepts a string and an eEventLogType. Note that the fully qualified eEventLogType is Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eEventLogType. To see the values defined in the enumeration, find the InstrumentObjects object in the left hand pane of the object browser. Expand that entry, and find Agilent.LWD.Ag86038x.InstrumentObjects, and expand that. Scan down the resultant list to find ODACCommon.eEventLogType, and click on that. It shows the five different possible values. It turns out the FailureAudit and SuccessAudit are unused default log values. The only values we will ever see are EventError (something bad happened and PDLA could not do what you asked), Information (simple status update, like Sweep Started) or Warning (PDLA was still able to make a measurement, but wasn't able to do precisely as asked.) The first pass parameter is the text of the message displayed to the user on the PDLA front panel. So, to be informed of a new status message, implement a function like this:

```
private void NewStatusEvent( string msg, ODACCommon.eEventLogType
e )
{
    // Let the user know something happened
    lblStatus.Text=msg;
}
```

Note that because of the ‘using’ statement at the head of the file, you did not have to type the whole root to eEventLogType – a considerable typing savings.

Now you need to tell the pdlaClient about the new function. First you have to find the delegate that knows how to handle NewStatus events. The delegates are also found underneath the ODACCommon object where you found the eEventLogType. Return to the Object Browser window and scroll down to find the ODARemoting.NewStatusDelegate object. That is the delegate you need to use for this event, and you will find delegates for all the PDLA events in the same place, with the same name as the event with the word Delegate appended. Note that several similarly spelled versions appear, but you only want the one spelled exactly the same as the event with the word Delegate appended.

The steps below walk you through adding the appropriate declarations to create the delegate and register our event handler.

- 1 Add a display line to show a user status messages. Add a text label as wide as the form, and call it lblStatus. Default the text to “PDLA Status.”
- 2 Go to the code window and find where you declared the pdlaClient. Add the declaration of a local copy of the delegate here:

```
// Declare event handlers for PDLA all!  
  
private pdlaRemoting.NewStatusDelegate NewStatusHandler;
```

- 3 Find the declaration of Form1_Closing, and insert this routine after it:

```
private void NewStatusEvent( string msg, ODACCommon.eEventLogType e )  
{  
    // Let the user know something happened  
    lblStatus.Text=msg;  
}
```

- 4 Now, register the handler when the connection to the PDLA is opened. Find the pbConnect_Click method, and add the following lines

```
// Create our event handlers  
this.NewStatusHandler = new ODARemoting.NewStatusDelegate(  
this.NewStatusEvent );
```

```
// Register event handlers  
pdlaClient.NewStatus += this.NewStatusHandler;
```

- 5 Since you registered the handler, you have to be careful to unregister it when you are done talking to the PDLA. Find

the `Form1_Closing` function, and insert the following lines before terminating the session:

```
// Unregister our status handler
pdlaClient.NewStatus -= this.NewStatusHandler;
```

- 6 Click the Run button, and connect to the PDLA. The first message you will see is the Connected message. Press Local on the PDLA and take a sweep. You can see the progress messages appearing on the application as the measurement progresses. When you exit the application, it will release the PDLA.

11 of the PDLA events follow a similar pattern. The most useful events will be `TriggerProgress`, which tells you when a measurement or normalization is complete, and `TLSSettlingProbability`, which warns you how long until the laser needs to perform a temperature settling.

Take Sweep and Transfer Data

In this exercise, you will trigger a sweep from the application, and when the sweep completes, transfer the data across and save it in a file. This consists of adding a button to trigger the sweep, adding an event handler for the `TriggerProgress` event to detect when the sweep is finished, and saving the data in `C:\mydata.txt`.

- 1 Add a button to the UI and name it `pbSweep`, caption Take Sweep.
- 2 Double click the button to go to the `pbSweep_Click` function, and insert the following lines to make sure you are connected before actually starting a measurement:

```
// Don't do anything if not connected to PDLA
if (pdlaClient.Connectivity.IsConnected() == false)
{
    MessageBox.Show("Can't take a measurement yet - not connected.");
    return;
}
pdlaClient.Measure();
```

- 3 Run the software, click Connect and wait for the connection to complete. Then click Take Sweep, and watch the sweep progress on both the PDLA and on the application message bar.

Add an event handler to `TriggerProgress`. This involves declaring the delegate, instantiating the delegate, then adding and removing the handler from `pdlaClient`.

The TriggerProgress event has two pass parameters – enumStatus and enumAcquisitionMode to determine the type of trigger event (that is measurement, normalization, and so on) and then checks the enumStatus to see if the measurement is finished. To declare the delegate, go to the source code window and find the line where we declare the NewStatusHandler, and add this line after it:

```
private ODARemoting.TriggerProgressDelegate TriggerHandler;
```

- 4 Find the line where you new ODARemoting.NewStatusDelegate, and add this line after it:

```
this.TriggerHandler = new ODARemoting.TriggerProgressDelegate(  
this.TriggerProgressEvent );
```

To register the handler, find the line where you += this.NewStatusHandler; and add the following line:

```
pdlaClient.TriggerProgress += this.TriggerHandler;
```

To unregister the handler, find the line where you -= this.NewStatusHandler and add the following line:

```
pdlaClient.TriggerProgress -= this.TriggerHandler;
```

- 5 Create the event function. In this function you have to check the enumAcquisitionMode to see if the trigger has to do with a measurement (could be a normalization) and then check the enumStatus to see if the measurement is finished, indicating data is available. If so, call sendDataToFile().

In the source code file, find where NewStatusEvent is declared, and add this function after it:

```
private void TriggerProgressEvent( ODACCommon.enumStatus status,  
ODACCommon.enumAcquisitionMode acqMode )  
{  
    if (acqMode == ODACCommon.enumAcquisitionMode.eMeasurement)  
    {  
        // Instrument is making a measurement  
        if (status == ODACCommon.enumStatus.COMPLETE)  
        {  
            // The measurement is done, transfer data!  
            sendDataToFile("c:\mydata.txt");  
        }  
    }  
}
```

- 6 Add the sendDataToFile function. This is a very simple implementation using text streams:

```
private void sendDataToFile(string fileName)  
{  
    // Transfer the gain data over from the PDLA and save it to a file  
    double[] gainData;  
    double xStart;  
    double xStop;
```

```

double xStep;
int npoin;
string buf;
int cnt;
double x;

// Get the gain data from port one
gainData = pdlaClient.Results.YData(ODACCommon.eMeasurementType.Gain,
ODACCommon.eODAPort.One);
npoin=gainData.Length;

// Get the start and stop wavelengths
xStart = pdlaClient.Results.XStart(ODACCommon.eMeasurementType.Gain);
xStop = pdlaClient.Results.XStop(ODACCommon.eMeasurementType.Gain);
xStep = (xStop - xStart)/ (npoin -1);

// Open the file
StreamWriter file = new StreamWriter(fileName, false);
file.WriteLine("Sample PDLA File");
file.WriteLine("Wavelength (nm), Gain (dB)");

// Write the data
for (cnt=0;cnt<npoin;++cnt)
{
    x = xStart + cnt * xStep;
    buf = x.ToString() + ", " + gainData[cnt].ToString();
    file.WriteLine(buf);
}

// Close the file
file.Close();
}

```

- 7 Run the software, connect and take a sweep. Look for c:\mydata.txt. You can look at it with a text editor or import it into a spreadsheet.

Tip: This program will grab data even if sweep is triggered from front panel. Run the software, and then connect. Go to the PDLA front panel and click Local, then take a sweep. You will see the update messages appear on our application, and the application will detect end-of-sweep and transfer the data.

- 8 Exit the application when done.

Control Center Wavelength

This section shows how to control the center wavelength on the PDLA. First, a text box is added for the user to enter the desired value, and then the entered value will be sent to the PDLA as part of the sweep.

- 1 Add a text field, name it txtCenter, default text to 1550. Then add a label that says Center Wavelength, nm:.
- 2 Modify the take sweep function to send the value to the PDLA. Find the function pbSweep_click, and add the following lines to check the value and send it to the PDLA. The PDLA thinks in terms of start/stop, so you have to convert the center wavelength to a start and stop wavelength:

```
// Get the center wavelength
double centerWavel, startWavel, stopWavel, oldSpan;
centerWavel = double.Parse(txtCenter.Text);
if (centerWavel < 0 || centerWavel > 1700)
    {
    MessageBox.Show("Center Wavelength out of range, defaulted to 1550 nm");
    centerWavel=1550;
    txtCenter.Text = centerWavel.ToString();
    }

// Convert center to start, stop
startWavel=pdlaClient.MeasurementRange.XStart;
stopWavel=pdlaClient.MeasurementRange.XStop;
oldSpan=stopWavel - startWavel;

// Write new start and stop
pdlaClient.MeasurementRange.XStart=centerWavel - oldSpan/2;
pdlaClient.MeasurementRange.XStop=centerWavel + oldSpan/2;
```

- 3 It works better if you first set the PDLA to a relatively narrow sweep, such as 1530 to 1540 nm.

Complete Source Code

```
using System;
using System.Drawing;
using System.Collections;
using System.ComponentModel;
using System.Windows.Forms;
using System.Data;
using System.IO;
using Agilent.LWD.Ag86038x.InstrumentObjects;
using Agilent.LWD.Ag86038x;

namespace WindowsApplication1
{
    /// <summary>
    /// Summary description for Form1.
    /// </summary>
    public class Form1 : System.Windows.Forms.Form
```

```

{
private System.Windows.Forms.TextBox tbPdlName;
private System.Windows.Forms.Label label1;
private System.Windows.Forms.Label lblStatus;

// Declare a local object to communicate with the PDLA
private RemoteClient.Communicator pdlaClient;

// Declare event handlers for PDLA - you have to do them all!
private ODARemoting.NewStatusDelegate NewStatusHandler;
private ODARemoting.TriggerProgressDelegate TriggerHandler;
private System.Windows.Forms.Button pbSweep;
private System.Windows.Forms.Button pbConnect;
private System.Windows.Forms.TextBox txtCenter;
private System.Windows.Forms.Label label2;

/// <summary>
/// Required designer variable.
/// </summary>
private System.ComponentModel.Container components = null;

public Form1()
{
//
// Required for Windows Form Designer support
//
InitializeComponent();

//
// TODO: Add any constructor code after InitializeComponent call
//
}

/// <summary>
/// Clean up any resources being used.
/// </summary>
protected override void Dispose(bool disposing)
{
if(disposing)
{
if (components != null)
{
components.Dispose();
}
}
base.Dispose(disposing);
}

#region Windows Form Designer generated code
/// <summary>
/// Required method for Designer support - do not modify
/// the contents of this method with the code editor.
/// </summary>

/// <summary>

```

```

/// The main entry point for the application.
/// </summary>
[STAThread]
static void Main()
{
    Application.Run(new Form1());
}

private void Form1_Load(object sender, System.EventArgs e)
{

}

private void NewStatusEvent(string msg, ODACCommon.eEventLogType e)
{
    // Let the user know something happened
    lblStatus.Text=msg;
}

private void TriggerProgressEvent(ODACCommon.enumStatus status,
ODACCommon.enumAcquisitionMode acqMode)
{
    if (acqMode == ODACCommon.enumAcquisitionMode.eMeasurement)
    {
        // Instrument is making a measurement
        if (status == ODACCommon.enumStatus.COMPLETE)
        {
            // The measurement is done, transfer data!
            sendDataToFile(@"c:\mydata.txt");
        }
    }
}

private void sendDataToFile(string fileName)
{
    // Transfer the gain data over from the PDLA and save it to a file
    double[] gainData;
    double xStart;
    double xStop;
    double xStep;
    int npoin;
    string buf;
    int cnt;
    double x;

    // Get the gain data from port one
    gainData = pdlaClient.Results.YData(ODACCommon.eMeasurementType.Gain,
ODACCommon.eODAPort.One);
    npoin=gainData.Length;

    // Get the start and stop wavelengths
    xStart = pdlaClient.Results.XStart(ODACCommon.eMeasurementType.Gain);
    xStop = pdlaClient.Results.XStop(ODACCommon.eMeasurementType.Gain);
    xStep = (xStop - xStart)/ (npoin -1);
}

```

```

// Open the file
StreamWriter file = new StreamWriter(fileName, false);
file.WriteLine("Sample PDLA File");
file.WriteLine("Wavelength (nm), Gain (dB)");

// Write the data
for (cnt=0;cnt<npoin;++cnt)
{
    x = xStart + cnt * xStep;
    buf = x.ToString() + ", " + gainData[cnt].ToString();
    file.WriteLine(buf);
}

// Close the file
file.Close();
}

private void Form1_Closing(object sender,
System.ComponentModel.CancelEventArgs e)
{
    // Release the PDLA
    pdlaClient.Connectivity.Disconnect();

    // Unregister our status handler
    pdlaClient.NewStatus -= this.NewStatusHandler;
    pdlaClient.TriggerProgress -= this.TriggerHandler;

    // Shut down the object
    pdlaClient.Connectivity.TerminateSession();
}

private void pbConnect_Click(object sender, System.EventArgs e)
{
    // Instantiate the pdlaClient
    pdlaClient = new RemoteClient.Communicator();

    // Connect to the PDLA
    pdlaClient.Connectivity.Connect(tbPdlName.Text);

    // Create our event handlers
    this.NewStatusHandler = new
ODARemoting.NewStatusDelegate(this.NewStatusEvent);
    this.TriggerHandler = new
ODARemoting.TriggerProgressDelegate(this.TriggerProgressEvent);

    // Register event handlers
    pdlaClient.NewStatus += this.NewStatusHandler;
    pdlaClient.TriggerProgress += this.TriggerHandler;

    // Disable the button to avoid double-connect
    pbConnect.Enabled = false;
}

private void pbSweep_Click(object sender, System.EventArgs e)
{
    // Don't do anything if not connected to PDLA

```

```

if (pdlaClient.Connectivity.IsConnected() == false)
{
    MessageBox.Show("Can't take a measurement yet - not connected.");
    return;
}
// Get the center wavelength
double centerWavel, startWavel, stopWavel, oldSpan;
centerWavel = double.Parse(txtCenter.Text);
if (centerWavel < 0 || centerWavel > 1700)
{
    MessageBox.Show("Center Wavelength out of range, defaulted to 1550 nm");
    centerWavel=1550;
    txtCenter.Text = centerWavel.ToString();
}
// Convert center to start, stop
startWavel=pdlaClient.MeasurementRange.XStart;
stopWavel=pdlaClient.MeasurementRange.XStop;
oldSpan=stopWavel - startWavel;
// Write new start and stop
pdlaClient.MeasurementRange.XStart=centerWavel - oldSpan/2;
pdlaClient.MeasurementRange.XStop=centerWavel + oldSpan/2;
    // Take the measurement
pdlaClient.Measure();

}

}

}

```

Example 2: Control from Excel

Complete Source Code

```
Option Explicit
Private WithEvents m_PDLARemoteControl As RemoteClient.Communicator
Public Sub initPdlaLink()
    Set m_PDLARemoteControl = New RemoteClient.Communicator
End Sub
Public Sub closePdlaLink()
    m_PDLARemoteControl.Connectivity.Disconnect
End Sub
Private Sub GoButton_Click()
    ' Clear the previous data
    Sheet1.Cells(9, 1).Select
    Range(Selection, Selection.End(xlToRight)).Select
    Range(Selection, Selection.End(xlDown)).Clear

    If Sheet1.ChartObjects.Count > 0 Then
        Sheet1.ChartObjects(1).Delete ' Delete last chart
    End If

    ' Open session
    m_PDLARemoteControl.Connectivity.Connect (PDLA_ID_Box.Value)

    ' Wait for PDLA connection to come alive
    DoEvents
    Application.Wait (Now + TimeValue("0:00:01"))
    DoEvents
    While m_PDLARemoteControl.Connectivity.IsConnected = False
        Application.Wait (Now + TimeValue("0:00:01"))
        DoEvents
    Wend

    ' Read x axis info
    Dim XStart As Double
    Dim xinc As Double
    Dim XData() As Double
    XStart = m_PDLARemoteControl.Results.XStart(eMeasurementType_Gain)
    xinc = m_PDLARemoteControl.Results.Increment

    'Get CD Data - Port One
    Dim GDData() As Double
    GDData = m_PDLARemoteControl.Results.YData(eMeasurementType_GD,
    eODAPort_One)

    'Get Gain Data - Port One
    Dim GainData() As Double
    GainData = m_PDLARemoteControl.Results.YData(eMeasurementType_Gain,
    eODAPort_One)

    ' close session
    m_PDLARemoteControl.Connectivity.Disconnect
```



```

' write data to spreadsheet
Dim npoin As Integer
Dim cnt As Integer
npoin = UBound(GainData) - LBound(GainData) + 1
ReDim XData(npoin)
For cnt = 0 To npoin - 1
    XData(cnt) = XStart + cnt * xinc
Next cnt

Sheet1.Cells(9, 1).Value = "Wavelength"
Sheet1.Cells(10, 1).Resize(npoin).Value = _
    Application.WorksheetFunction.Transpose(XData)
Sheet1.Cells(9, 2).Value = "Gain, dB"
Sheet1.Cells(10, 2).Resize(npoin).Value = _
    Application.WorksheetFunction.Transpose(GainData)
Sheet1.Cells(9, 3).Value = "Group Delay, ps"
Sheet1.Cells(10, 3).Resize(npoin).Value = _
    Application.WorksheetFunction.Transpose(GDDData)

' Do a quick plot

Sheet1.Cells(9, 1).Select
Range(Selection, Selection.End(xlToRight)).Select
Range(Selection, Selection.End(xlDown)).Select

Charts.Add
ActiveChart.ChartType = xlXYScatterLinesNoMarkers
ActiveChart.Location Where:=xlLocationAsObject, Name:="Sheet1"
With ActiveChart
    ' Get rid of time data as a plotted series
    .SeriesCollection(1).Delete
    ' Specify the X axis for the other two series
    .SeriesCollection(1).XValues = Range(Sheet1.Cells(10, 1), Sheet1.Cells(9 + npoin,
1))
    .SeriesCollection(2).XValues = Range(Sheet1.Cells(10, 1), Sheet1.Cells(9 + npoin,
1))
    ' set title
    .HasTitle = True
    .ChartTitle.Characters.Text = "Gain and Group Delay"
    ' Add x axis with title
    .Axes(xlCategory, xlPrimary).HasTitle = True
    .Axes(xlCategory, xlPrimary).AxisTitle.Characters.Text = "Wavelength, nm"
    ' Add left hand y axis with title
    .Axes(xlValue, xlPrimary).HasTitle = True
    .Axes(xlValue, xlPrimary).AxisTitle.Characters.Text = "Gain, dB"
    .Axes(xlValue, xlPrimary).CrossesAt = -200
    ' Add right hand y axis with title
    .SeriesCollection(2).Select
    .SeriesCollection(2).AxisGroup = 2
    .Axes(xlValue, xlSecondary).HasTitle = True
    .Axes(xlValue, xlSecondary).AxisTitle.Characters.Text = "delay, ps"
    ' Move the legend to the bottom of the plot
    .Legend.Position = xlLegendPositionBottom
End With

```



End Sub

Example 3: Control from LabView

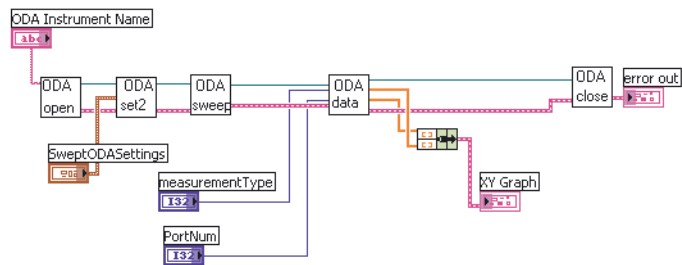
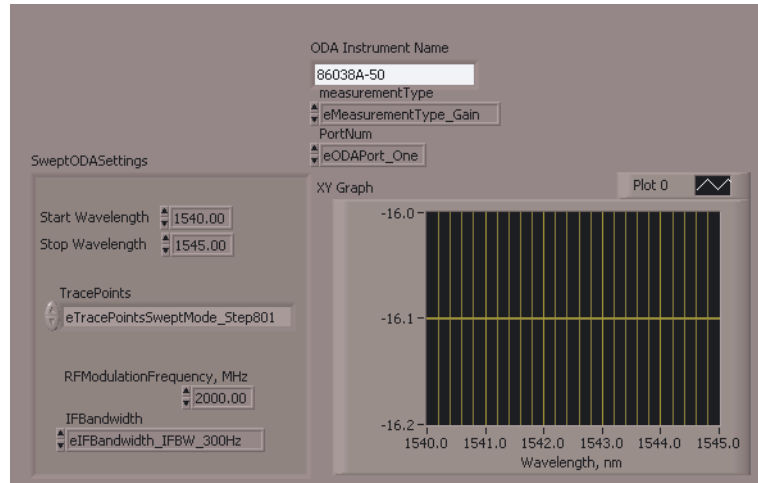


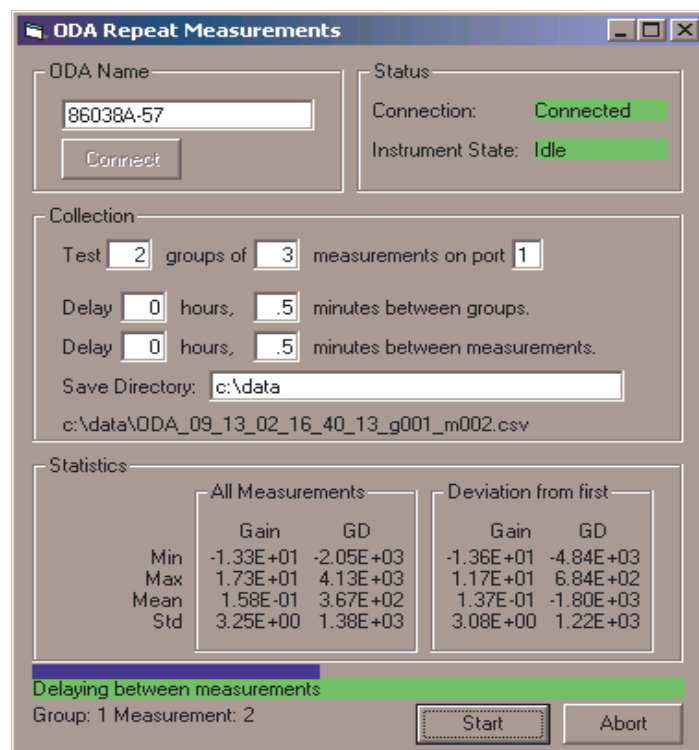
Table 5-6. Provided vi's

example.vi	A simple example that makes a swept (vs. stepped) measurement and presents a graph.
ODAclose.vi	This should be the last vi used, to close the session with the PDLA.
ODAdata.vi	This vi transfers data from the PDLA into LabView
ODAsset1_stepped.vi	This vi contains all the settings for making a stepped gain and CD measurement.
ODAsset2_swept.vi	This vi contains all the settings for making a swept gain and CD measurement.
ODAsweep.vi	This vi triggers a sweep and waits for it to complete.

Example 4: Complete application - RepeatMeas

PDLA Repeat Measurements

PDLA Repeat Measurements is an example program distributed with the PDLA in both binary and source code form. To use it, you set up a measurement you like on the PDLA, then run the executable on your own computer. You key in how many times you would like the measurement made, and at what interval, and the program triggers the measurements and records the data on your computer.



- 1 Set up the desired measurement on the PDLA, performing the required normalization.
- 2 Go to the PDLA Help >About menu to obtain the PDLA computer name. Type that name into the PDLA Name field, and click Connect. After a pause while contact is established, the Connection status should turn green:

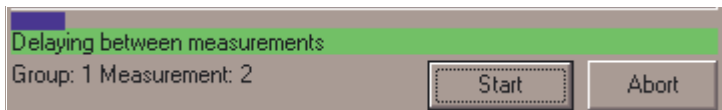


- 3 Enter the number of sweeps you would like to occur. If you want a lot of sweeps as fast as the instrument can acquire, enter the number of measurements you want in the measurements block and leave everything else at the default. If you want the program to pause between measurements, enter the length of time to pause. You can enter fractional minutes. Sometimes for statistical purposes, you may want a burst of measurements repeated at some interval. For example, you may want 16 measurements taken every hour for 12 hours.
- 4 Make sure to tell the program which port to pull data from. The measurements will be run as you configure from the front panel, and the program will pull the data from the port you specify (whether or not new data has been gathered for that port.)
- 5 Enter the directory you would like the data to be stored to on the local computer. You will get one file for every measurement with Wavelength, Gain, Group Delay and Chromatic Dispersion. The file name is noted at the bottom of the Collection frame. You will also get a single ODA_stats file with the statistics for the Gain and Group Delay. The program will use as short a file name as it can. For data files, it uses ODA_<date>_g001_m001.csv, where g is followed by the group number, m is followed by the measurement number. If the name is already in use, the program will insert the date to make the name unique, and avoid overwriting data. The <date> used is the time the first sweep is completed. The same <date> string is used for the entire run, so all the files have the same signature. The format of putting the date before the group and measurement indexes forces a standard ASCII directory sorting to group all the data from a single run together in the correct order. Make sure the directory you specify exists before you start the run.
Tip: the files are easier to deal with if you put each run into a separate directory.
- 6 Click Start and observe the measurement results as they are taken.

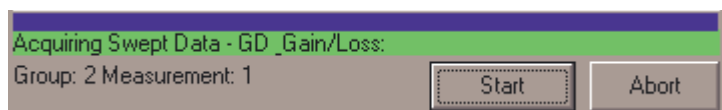
Statistics					
	All Measurements		Deviation from first		
	Gain	GD	Gain	GD	
Min	-3.98E+01	-1.03E+01	-7.62E-01	-4.64E+00	
Max	-2.92E+01	3.88E+00	6.80E-01	4.31E+00	
Mean	-3.00E+01	-3.15E-01	-6.52E-04	-1.77E-01	
Std	1.63E+00	1.43E+00	7.09E-02	3.39E-01	

The statistics are calculated for Gain and Group Delay. The All Measurements block is calculated treating all wavelengths of all measurements equally. This is useful for measuring very flat and stable devices like a length of fiber. For devices that have more features to the response, the statistics for the deviation from the first measurement are also calculated. This shows you at a glance how stable the device was after the first measurement. This information is also stored at the end of the run in the ODA_stats file.

Just below the Statistics are is the Status area. The blue bar is used to show the timer between PDLA measurements:



During the measurement itself, the blue bar is full and the Status area reflects what the PDLA is doing:




- 7 Install the .NET Framework from Microsoft. Refer to [“Install the Remote Client” on page 141](#).
- 8 Install the remote programming application from the PDLA documentation CD located at:

C:\Program Files\Agilent\Agilent PDLA Remote
Control\Examples\Example4_RepeatMeas\Application\

And run OdaRepeatMeas.exe

There are three main functional blocks in the code:

- Triggering measurements and setting up the delay between measurements

- 
- Processing the measured data into statistics and writing data to the file
 - Updating the PDLA status

Triggering Measurements and Delay Between Measurements

Once the connection is established to the instrument and the measurements started, a timer called MeasTimer on the form is used to delay between measurements. Because the MeasTimer can only delay about one minute, and since we need to update the progress bar, the MeasTimer only delays about 150 ms at a time. It wakes up, updates the progress bar, then goes back to sleep.

The main routines involved are:

- MeasTimer_Timer() – wakes up every 150 ms, updates progress bar, decides if it is time to take data.
- startMeasurement() – triggers the PDLA to make the measurement. The TriggerProgress event communicates when the sweep is finished and provides a chance to process the measured data.

Key variables:

- groupIdx – Current group, starting with 1
- measIdx – Current measurement, starting with 1
- numGroups – Number of user requested groups
- numMeas – Number of user requested measurements per group
- delayPerGroup – User requested delay per group
- delayPerMeas – User requested delay between each measurement
- nextDelay – Remaining time left before next measurement
- requestedDelay – Either the delayPerMeas (if between measurements) or the delayPerGroup (if between groups)
- TimerMax – How long to wait between progress bar updates.

Processing Measured Data

The event handler for TriggerProgress detects when a sweep is complete and calls GetAndSaveRawData to process the data. This routine transfers the data, forms the appropriate file name, saves the file, and updates the statistics for the gain and the group delay. Key to the name of the file is the groupID, which holds the date string required to make the name unique. If there are no files in the save directory, groupID will be blank. The groupID is made progressively more specific – adding date, then time if needed.

The statistics are accumulated in statistics structures called a statsBlock. There are two forms of the statistics provided – the absolute values, and the differential from the first sweep. Variables that begin with allTrace accumulate the absolute values, and variables that begin with deltaTrace accumulate the delta values.

When all the groups have been acquired, the statistics are written to a summary file.

Updating PDLA Status

Updating the PDLA status is completely asynchronously from any measurement operation. Every 500 ms, StatusTimer wakes up and queries the PDLA to find out what it is doing. This is handled in StatusTimer_Timer(). In addition, whenever a new status message appears on the PDLA front panel, the NewStatus event handler, pdlaClient_NewStatus(), reads the message and displays it for the remote user as well.

6

Programming Commands

PDLA Remote Programming.....	212
Server to Client Events.....	213
Client to Server Commands.....	218
Base Commands.....	222
Actions Commands.....	228
Analysis - CurveFit Commands.....	231
Connectivity Commands.....	235
Display Commands.....	237
File Commands.....	240
Length Commands.....	243
MeasurementRange Commands.....	247
NormalizationRange Commands.....	249
Increment Property.....	253
Printing Commands.....	255
Printing Commands.....	255
Resolution Commands.....	256
Results Commands.....	260
Sensitivity Commands.....	264
Status Commands.....	267
System Commands.....	272
System - Identity Commands.....	275
TLS Commands.....	278

PDLA Remote Programming

The PDLA remote programming interface uses Microsoft.NET Remoting technology. It is controlled by manipulating the properties and methods exposed by the server object. The list of events, properties and methods in this section describe the interface that is available to a programmer wanting to use the PDLA measurement engine in some other development environment.

PDLA Remote Control DLLs

The PDLARemoteControl DLLs provide a communication link with the PDLA application. The DLLs are comprised of a set of Properties, Methods, and Events that together provide a basic set of remote PDLA capabilities. The two DLLs of interest are:remoteClient.dll and InstrumentObjects.dll. By default these two DLLs are installed to: C:\Program Files\Agilent\Agilent PDLA Remote Control\.

.NET Interface Hierarchy

The PDLA Remote Control organizes its APIs into a Function Tree hierarchy. This associates like functions into logical groupings. Therefore, the remoting commands are grouped and listed by function, then command type in the following sections.

Server to Client Events

The server to client events allow the 86038B PDLA to update the client with current data or instrument settings. In this context, the PDLA is the server and your program is the client. These events are asynchronous. That is, you cannot determine when they are going to occur. You need to set up event handlers for the events you want to catch.

Connected (ByVal connected As Boolean)

Indicates that a remote session has connected to/disconnected from the PDLA. Once you have requested a connection using the Connect method on the Connectivity object, you must wait for this event to return a true before sending commands to the PDLA. If you receive this event after your connection has been established, and the value is false, you need to re-establish your connection.

Parameters

connected - True = Connected to the PDLA

Related Commands

- Connectivity.Connect method
- Connectivity.Disconnect method
- Connectivity.TerminateSession method

EstimatedTestTime (ByVal sTestName As String, ByVal tsEstimate As Double)

Sends estimated time of a particular test. This event is sent once at the beginning of a test. An action such as a Normalization may be comprised of several tests. In this case, the EstimatedTestTime event will be sent at the beginning of each test.

Parameters

sTestName - The name of the test.

tsEstimate - The estimated time of the test in seconds.

NewDataUpdate **(ByVal eLaserMode As enumLaserMode, ByVal eAcquisitionMode As enumAcquisitionMode)**

Indicates when new measurement data is available. For swept mode measurements, this event is fired when the sweep is complete. For stepped mode measurements, a NewDataUpdate event is fired for each step. This event only gets sent during measurements, normalization and length measurements. (Not during Optimize Optical Modulator Bias or Real Time Port Monitor.)

Parameter

eLaserMode - Applies to a stepped or swept mode measurement.

eAcquisitionMode - Indicates data is for a measurement, normalization, or length measurement.

Related Commands

- Actions.Measure method
- Actions.Normalize method
- Actions.MeasureLength method

NewRealTimePortMonitorData **(ByVal DUTPortInsertionLoss As Double, ByVal MonitorPortInsertionLoss As Double, ByVal DUTPortGroupDelay As Double, ByVal MonitorPortGroupDelay As Double)**

Indicates that a new set of data from the real-time port measurement is available.

Parameters

DUTPortInsertionLoss - The insertion loss result for the DUT channel in dB.

MonitorPortInsertionLoss - The insertion loss result for the Monitor channel in dB.

DUTPortGroupDelay - The group delay result for the DUT channel in ps.

MonitorPortGroupDelay - The group delay result for the Monitor channel in ps.

Related Commands

System.RealTimePortMonitor

NewStatus

(ByVal Message As String, ByVal MessageType As eEventLogType)

Indicates that a new status message is available, which can be displayed as part of the system status.

Parameters

Message - Message content.

MessageType - Indicates the severity of the message (informational or error)

Related Command

- Status.StatusLog method

PresetProgress

(ByVal progress As enumStatus)

Indicates that the PDLA has started/completed a preset (or initialization).

Parameters

Progress - Indicates that the preset/initialization has started or completed.

Related Command

- System.Preset method

Remoting

(ByVal Remoting As Boolean)

Indicates that the PDLA is under local or remote control. If you press the Local key after you establish a connection, you will receive this event with the value False.

Parameters

Remoting - True = Remoting has control of the PDLA user interface.

Related Command

- Display.ToggleRemoteIndicator method

SweepProgress

(ByVal progress As enumStatus, ByVal eTestType As enumTestType, ByVal eTriggerMode As enumAcquisitionMode)

Indicates a sweep has begun, has been aborted, or has been completed. Note that some actions may involve several sweeps. Wait for a TriggerProgress event to determine when the action is complete.

Parameters

progress - Indicates a measurement has begun, completed, or aborted.

eTestType - Indicates the type of test

eTriggerMode - Indicates the subset the event resides in. (Normalization, Measurement, ModBias, DriftMonitor, Length)

TLSSettlingProbability

(ByVal dTimeRemaining As Double, ByVal eProbability As enumTLSSettlingProbability)

Indicates the estimated time until the next laser settling begins. The PDLA relies on the tunable laser to provide high resolution and dynamic range. The laser has a mechanically tuned cavity that is sensitive to temperature change, requiring the laser to track the internal temperature and occasionally perform a correction. Under normal temperature conditions, the status message may show 120 minutes as the estimate. This status message is updated every minute.

Parameters

dTimeRemaining - Indicates the time remaining until the next TLS Settling. Time is in minutes.

eProbability - The probability of a TLS Settling being performed.

Related Command

- TLS.ForceSettling method

TLSSettingProgress (ByVal progress As enumStatus)

Indicates a laser settling has started or completed. Once a laser settling has begun, the PDLA will not be able to make measurements until it is complete. To avoid settling in the middle of your test, track the estimated time until laser settling. When it falls below 10 minutes, you should force a TLS settling with the TLS.ForceSettling method.

Parameters

progress - Indicates TLS Settling has begun or completed.

Related Command

- TLS.ForceSettling method

TriggerProgress (ByVal progress As enumStatus, ByVal eTriggerMode As enumAcquisitionMode)

Indicates a trigger has begun, has been aborted, or has been completed. Some actions such as normalization may include several sweeps. This event is provided to tell you when the entire action is complete. In general, this event is more useful than SweepProgress because you only get one “begin” and one “completed” per action.

Parameters

progress - Indicates the measurement action has begun, completed, or has been aborted.

eTriggerMode - Indicates the subset the event resides in. (Normalization, Measurement, ModBias, DriftMonitor, Length)

Client to Server Commands

The client to server commands allow you to communicate with the 86038B PDLA using either a .NET or COM interface.

- The commands listed in the following section assume an instance of the remote client has been created using the following VB.NET commands:
- `Public WithEvents pdlaClient As RemoteClient.Communicator`
`pdlaClient = New RemoteClient.Communicator()`
- Examples shown with each command are presented in VB.NET and VB 6.0.

Base Commands	222
DispersionMode Property	222
NumPolStatesPMD Property	222
ExternalPowermeterFound Property	223
UseExternalPowermeter Property	223
DUTMinLossMaxGain: Property	224
GenerateMuellerAndPMDData Property	224
Measure Method	225
Ports Property	225
MaxNumPorts Property	225
TriggerModeParams Property	226
AutoCancelAfterBadSweeps Property	226
Actions Commands	228
Abort Method	228
Measure Method	228
MeasureLength Method	228
Normalize Method	229
PrintScreen Method	229
Analysis - CurveFit Commands	231
Calculate Method	231
Catalog Property	232
Description Method	232
ShortName Method	233
Connectivity Commands	235
Connect Method	235
Disconnect Method	235
IsConnected Method	236
TerminateSession Method	236
Display Commands	237
AutoscaleDuringMeasurement Method	237
DefineTrace Method	237
GDOffsetMode Property	238
ToggleMode Method	238
ToggleRemoteIndicator Method	239
File Commands	240
ExportCSV Method	240
ExportMuellerPMD Method	240
ExportStatusLog Method	241
Recall Method	241
Save Method	242
Length Commands	243
IndexOfRefraction Property	243
Mode Property	243
SpecifiedLength Property	244

ValidIndexOfRefraction Method	244
ValidSpecifiedLength Method	245
MeasurementRange Commands	247
ValidStartStop Method.	247
XStart Property	247
XStop Property.	248
NormalizationRange Commands.	249
ClearGDAndInsertionLossOffsets Method.	249
ClearModBias Method	249
ClearPDA Method.	250
ClearInputPolarizerAngle Method	250
PDAEnabled Property	250
UseCustomNormalization Property	251
ValidStartStop Method.	251
XStart Property	252
XStop Property.	252
Increment Property	253
ValidIncrement Method	253
Printing Commands	255
Printing Commands	255
PrintScreen Method	255
Resolution Commands	256
RFModulationFrequency Property.	256
ValidRFModulationFrequency Method.	256
Increment Property	257
ValidIncrement Method	258
ValidTracePoints Method.	258
Results Commands	260
FiberLength Method.	260
Increment Property	260
XStart Method.	261
XStop Method	261
XValueFromPointIndex Method	262
YData Method	262
Sensitivity Commands	264
DriftCorrection Property.	264
IFBandwidth Property.	264
SamplesPerPoint Property	265
ValidSamplesPerPoint Method	266
Status Commands.	267
ErrorFlag Property.	267
EstimatedTimeToTLSSettling Property.	267
InstrumentState Property	268
LastError Property.	268
PresetComplete Property	269

StatusLog Method	269
TLSSettling Property	270
TriggerComplete Property	270
ValidNormalization Property	271
System Commands	272
LaserState Method	272
PowerMeterZero Method	272
Preset Method	272
WavelengthRealtime Property	273
SetPort Method	273
RealtimePortMonitor Method	274
System - Identity Commands	275
CompanyName Property	275
ProductId Property	275
ProductName Property	275
ProductVersion Property	276
RemotingVersion Property	276
TLS Commands	278
ForceSettling Method	278
PowerSelect Property	278
MultiTlsSupport Method	279
SelectTls Method	279
TlsWvlRange Method	280
TlsWvlRangeString Method	281
SelectedTLSEidx Method	281
TlsIdn Method	282
TlsPn Method	282
TlsSn Method	283
HighPowerLimit Property	284

Base Commands

DispersionMode Property

Selects CD Stepped, CD Swept, CD_PMD Stepped or CD_PMD Swept. Note that group delay and gain/loss are always measured. This is equivalent to selecting CD/PMD on the front panel.

Property Value

InstrumentObjects.ODACCommon.eDispersionMode

VB.NET Syntax

```
pdlaClient.DispersionMode=  
_InstrumentObjects.ODACCommon.eDispersionMode.CD_Swept
```

VB 6.0 Syntax

```
pdlaClient.DispersionMode=eDispersionMode_CD_Swept
```

NumPolStatesPMD Property

Read the value of this property to find out what number of polarization states is used for PMD and PDL measurements, or to set the value of this property for the number of pol.-states to be used for future PMD and PDL measurements.

Property Value

InstrumentObjects.ODACCommon.eNumPolStatesPMD

VB.NET Syntax

```
pdlaClient.NumPolStatesPMD =  
InstrumentObjects.ODACCommon.eNumPolStatesPMD.eFour
```

VB 6.0 Syntax

```
pdlaClient.NumPolStatesPMD = eNumPolStatesPMD_eFour
```

ExternalPowermeterFound Property

The value is set to true, if external power meters have been found in the current setup. These power meters could be used for IL and PDL measurements. Also see the "UseExternalPowermeter" property.

Read-Only Property

Boolean - Tells if external power meters are found in the current setup.

VB.NET Syntax

Dim PWMFound As Boolean

PWMFound = pdlaClient.ExternalPowermeterFound

VB 6.0 Syntax

Dim PWMFound As Boolean

PWMFound = pdlaClient.ExternalPowermeterFound

UseExternalPowermeter Property

This property defines if external power meters should be used for IL and PDL data. If no external power meters have been found in the current setup, the value of this property is ignored. Also see the "ExternalPowermeterFound" property.

Property Value

Boolean - True if external power meters are to be used

VB.NET Syntax

pdlaClient.UseExternalPowermeter = True

VB 6.0 Syntax

pdlaClient.UseExternalPowermeter = True

DUTMinLossMaxGain: Property

The value of this property specifies the expected minimum loss or maximum gain of the device under test. This value is used by the PDLA system to determine the maximum power to be detected by the external power meters. If no external power meters have been found in the current setup, the value of this property is ignored.

Property Value

InstrumentObjects.ODACCommon.eDUTLossGain

VB.NET Syntax

```
pdlaClient.DUTMinLossMaxGain =  
InstrumentObjects.ODACCommon.eDUTLossGain.eGain_M20dB
```

VB 6.0 Syntax

```
pdlaClient.DUTMinLossMaxGain =  
eDUTLossGain_eGain_M20dB
```

GenerateMuellerAndPMDData Property

Use this property to control the generation of Mueller matrix and PMD vector data for export. Generation of this data results in larger .oda files, when storing measurement results. See also: ExportMuellerPMD Method.

Property Value

Boolean - True if Mueller matrix and PMD vector data for ASCII export is generated

VB.NET Syntax

```
pdlaClient.GenerateMuellerAndPMDData = True
```

VB 6.0 Syntax

```
pdlaClient.GenerateMuellerAndPMDData = True
```


Measure Method

Starts a new measurement. This method corresponds to the Begin Meas key on the front panel. Use this method to start a measurement after you have set the measurement parameters, and then watch the TriggerProgress event for completion.

VB.NET Syntax

```
pdlaClient.Measure()
```

VB 6.0 Syntax

```
pdlaClient.Measure
```

Ports Property

Read the value of this property to find out which port or ports are enabled, or set the value of this property to select the active port or ports. This is equivalent to pressing Ports on the front panel.

Property Value

```
InstrumentObjects.ODACCommon.eODAPort
```

VB.NET Syntax

```
pdlaClient.Ports =  
InstrumentObjects.ODACCommon.eODAPort.One _  
+ InstrumentObjects.ODACCommon.eODAPort.Two
```

VB 6.0 Syntax

```
pdlaClient.Ports=eODAPort_One + eODAPort_Two
```

MaxNumPorts Property

Tells the maximum number of ports which could be measured with the current setup

Read-Only Property

```
Integer
```

VB.NET Syntax

```
Dim MaxPorts As Integer  
MaxPorts = pdlaClient.MaxNumPorts
```

VB 6.0 Syntax

```
Dim MaxPorts As Integer  
MaxPorts = pdlaClient.MaxNumPorts
```

TriggerModeParams Property

Used to read or set the instrument trigger mode.

Single sweep initiates one sweep on the measurement range. Repeat sweep continuously sweeps until the measurement is canceled or the instrument is set to single sweep. Most remote applications use single sweep.

Property Value

```
InstrumentObjects.ODACCommon.eTriggerTypeVB.NET Syntax  
pdlaClient.TriggerModeParams =  
_InstrumentObjects.ODACCommon.eTriggerType.TriggerSingle
```

VB 6.0 Syntax

```
pdlaClient.TriggerModeParams = eTriggerType_TriggerSingle
```

AutoCancelAfterBadSweeps Property

This property controls the behavior of the PDLA system in case of problems during a measurement sweep. If the value of this property is True, the system stops the current measurement if it detects problems during the last sweep. If the value of this property is False, the measurement is continued and the user has to decide if the results are useful or not. In every case, the system reports a warning about the problem.

Property Value

Boolean - Set this value to True, to activate automatic canceling.

VB.NET Syntax

```
pdlaClient.AutoCancelAfterBadSweeps = True
```

VB 6.0 Syntax

```
pdlaClient.AutoCancelAfterBadSweeps = True
```

Actions Commands

The Actions property returns an actions object, which has methods for starting and stopping measurements and normalizations.

Abort Method

Stops the current measurement, normalization, real-time port monitor, or length measurement.

VB.NET Syntax

```
pdlaClient.Actions.Abort()
```

VB 6.0 Syntax

```
pdlaClient.Actions.Abort
```

Measure Method

Starts a new measurement. This method corresponds to the Begin Meas key on the front panel. Use this method to start a measurement after you have set the measurement parameters, and then watch the TriggerProgress event for completion.

VB.NET Syntax

```
pdlaClient.Actions.Measure()
```

VB 6.0 Syntax

```
pdlaClient.Actions.Measure
```

MeasureLength Method

Starts a length measurement.

VB.NET Syntax

```
pdlaClient.Actions.MeasureLength()
```

VB 6.0 Syntax

```
pdlaClient.Actions.MeasureLength
```

Normalize Method

Starts a normalization.

Parameters

Optional ByVal NormMode As

InstrumentObjects.ODACCommon.enumAcquisitionMode =
eNormalization

Allowed NormMode values are:

InstrumentObjects.ODACCommon.enumAcquisitionMode.eModBi
as

InstrumentObjects.ODACCommon.enumAcquisitionMode.eOptPo
larizer

InstrumentObjects.ODACCommon.enumAcquisitionMode.eNorm
alization

If NormMode = eModBias, then only an optical modulator
optimization is done during normalization.

If NormMode = eOptPolarizer, then only the input polarizer is
optimized during normalization.

If NormMode = eNormalization, then a complete Normalization
measurement is done.

VB.NET Syntax

```
pdlaClient.Actions.Normalize(InstrumentObjects.ODACCommon.  
enumAcquisitionMode.eNormalization)
```

VB 6.0 Syntax

```
pdlaClient.Actions.Normalize(InstrumentObjects.ODACCommon.  
enumAcquisitionMode.eNormalization)
```

PrintScreen Method

This command is equivalent to the front panel Print button.

VB.NET Syntax

```
pdlaClient.Actions.PrintScreen()
```



VB.6.0 Syntax

`pdlaClient.Actions.PrintScreen`

Analysis - CurveFit Commands

The Analysis property returns an object, which in turn, contains the CurveFit property. The CurveFit property returns an object with methods and properties for controlling the application of curve fits to the data returned from the PDLA.

Calculate Method

Applies the specified curve fit to the passed-in data and returns the resultant trace.

Parameters

ByVal CurveFitName As String - The name of the curve fit. A list of the curve fit names can be retrieved via the Catalog command.

ByVal XStartPoint As Double - The start wavelength of the input data.

ByVal XIncrement As Double - The wavelength spacing between points of the input data.

ByRef InputActualData() As Double - The data to apply the curve fit to.

ByRef GeneralForm As String - The general form of the equation for the specified curve fit.

ByRef Equation As String - The actual equation that best fits the input data using the specified curve fit.

Returns

Double() - An array of the curve fitted data is returned.

VB.NET Syntax

```
Dim OutTraceData() As Double
```

```
OutTraceData = pdlaClient.Analysis.CurveFit.Calculate _  
("Quadratic", XStart, XIncrement, InTraceData, GeneralForm, Equation)
```

VB 6.0 Syntax

```
Dim OutTraceData() As Double
```

```
OutTraceData = pdlaClient.Analysis.CurveFit.Calculate _  
("Quadratic", XStart, XIncrement, InTraceData, GeneralForm,  
Equation)
```

Catalog Property

Returns a list of available curve fits as an array of strings.

Property Value

String() - Contains an array of strings for all the curve fits supported by the system.

VB.NET Syntax

```
Dim MyCat() As String  
MyCat = pdlaClient.Analysis.CurveFit.Catalog
```

VB 6.0 Syntax

```
Dim MyCat() As String  
MyCat = pdlaClient.Analysis.CurveFit.Catalog
```

Description Method

Returns one of the following descriptions, corresponding to the passed-in short curve fit name.

Linear calculates a single parameter least squares fit. The equation is in the form: $y = Ax + B$. It is commonly used for dispersion-shifted fiber, in which waveguide dispersion is dominant.

Quadratic calculates a second order polynomial fit. The equation is in the form:

$$y = Ax^2 + Bx + C .$$

3rd Order Sellmeier curve fit is commonly used for dispersion-unshifted fiber, in which material dispersion is dominant. The equation is in the form:

$$y = A^2 + B + Cx^{-2}$$

5th Order Sellmeier curve fit, although more affected by noise and instabilities in the measurement path, provides more general purpose curve fitting. The five-term Sellmeier fit can yield multiple zero-dispersion wavelengths. All the values and their associated slopes are shown in the graphs. The system searches for dispersion zeros in a wavelength range equaling approximately five times the measurement span (2.5 times each side of the center wavelength). This allows identification of zero-dispersion wavelengths which fall outside of the measurement range. In some cases, due to the peculiarity of the five-term Sellmeier fit, zero-dispersion wavelengths found outside of the measurement range may not correspond to actual zero-dispersion wavelengths of the device under test. The equation is in the form: $Ax^4 + Bx^2 + C + Dx^{-2} + Ex^{-4}$.

Parameters

ByVal CurveFitName As String - The short name of the curve fit. A list of the curve fit names can be retrieved via the Catalog command.

Returns

String - Contains a description of the curve fit.

VB.NET Syntax

```
Dim desc As String  
desc = pdlaClient.Analysis.CurveFit.Description("Quadratic")
```

VB 6.0 Syntax

```
Dim desc As String  
desc = pdlaClient.Analysis.CurveFit.Description("Quadratic")
```

ShortName Method

Returns the short name of the curve fit. That is, linear, quadratic, Sellmeier-3, or Sellmeier-5.

Parameters

ByVal CurveFitName As String - The name of the curve fit. A list of the curve fit names can be retrieved via the Catalog command.

Returns

String - Contains the short name of the curve fit, for example quadratic.

VB.NET Syntax

```
Dim eqName As String
```

```
eqName = pdlaClient.Analysis.CurveFit.ShortName("Quadratic")
```

VB 6.0 Syntax

```
Dim eqName As String
```

```
eqName = pdlaClient.Analysis.CurveFit.ShortName("Quadratic")
```

Connectivity Commands

The Connectivity property of the remote client returns a connectivity object, with methods and properties for controlling the LAN connection to the PDLA.

Connect Method

Connects a remoting session to the remote server. Pass it the network name of the PDLA. The network name can be found by looking at the Help > About dialog from the PDLA front panel. You must wait until the connected event returns true before sending commands.

You can also use the IP address of the PDLA in string form. For example "xxx.xxx.xxx.xxx". An error will be raised if the Connect method is called on a remote client that is already connected.

Parameters

ByVal serverMachineName As String - The network name of the PDLA.

VB.Net Syntax

```
pdlaClient.Connectivity.Connect("MYPDLA")
```

VB 6.0 Syntax

```
pdlaClient.Connectivity.Connect("MYPDLA")
```

Disconnect Method

Disconnects the remoting session from the PDLA. Events will not be sent to the remote client once it disconnects.

VB.NET Syntax

```
pdlaClient.Connectivity.Disconnect()
```

VB 6.0 Syntax

```
pdlaClient.Connectivity.Disconnect
```

IsConnected Method

Checks to see if a remoting connection is established to the PDLA. It is usually more efficient to use the Connected event, but this is a query you can make at any time.

Returns

Boolean - True is connected.

VB.NET Syntax

```
If (pdlaClient.Connectivity.IsConnected() = True) Then
    MsgBox("Connected")
End If
```

VB 6.0 Syntax

```
If (pdlaClient.Connectivity.IsConnected = True) Then
    MsgBox("Connected")
End If
```

TerminateSession Method

This method unregisters the communicator channel to the PDLA. TerminateSession should be called when remoting to the PDLA has been concluded.

VB.NET Syntax

```
pdlaClient.Connectivity.TerminateSession
```

VB 6.0 Syntax

```
pdlaClient.Connectivity.TerminateSession
```

Display Commands

The Display property of the remote client object returns a display object with methods and properties for controlling the displayed traces.

AutoscaleDuringMeasurement Method

For swept measurements, an auto scale is performed on all graphs at the end of a measurement.

For stepped measurements, an auto scale is performed on all graphs at the end of each wavelength step as specified in the Resolution.Increment property.

Parameters

ByVal enabled As Boolean - True turns on auto scaling.

Related Command

Resolution.Increment property

VB.NET Syntax

```
pdlaClient.Display.AutoscaleDuringMeasurement(True)
```

VB 6.0 Syntax

```
pdlaClient.Display.AutoscaleDuringMeasurement(True)
```

DefineTrace Method

Assigns a measurement on a specified port to a specific trace on the graph. Up to two measurements can be assigned to each of the four graphs.

Parameters

ByVal graphNumber As Integer - The graph number assigned to the measurement: 0 = Top Left, 1 = Bottom Left, 2 = Top Right, 3 = Bottom Right.

ByVal trace As InstrumentObjects.ODACCommon.eTraceLetter - Selects the left- or right-hand trace on the specified graph.

ByVal port As InstrumentObjects.ODACCommon.eODAPort - Selects the port number for the measurement.

ByVal measurementType As
InstrumentObjects.ODACCommon.eMeasurementType - The type
of measurement to display: CD, Gain/Loss, GD, PDL, or DGD.

VB.NET Syntax

```
pdlaClient.Display.DefineTrace(0, _  
InstrumentObjects.ODACCommon.eTraceLetter.TraceA, _  
InstrumentObjects.ODACCommon.eODAPort.One, _  
InstrumentObjects.ODACCommon.eMeasurementType.GD)
```

VB 6.0 Syntax

```
Call pdlaClient.Display.DefineTrace(0, _  
eTraceLetter_TraceA, _  
eODAPort_One, _  
eMeasurementType_GD)
```

GDOffsetMode Property

The PDLA normally sets the group delay at the center trace point to zero. Writing a value to this property can change the point that is zeroed to the minimum or maximum of the trace. Reading the value of the property returns the current setting.

Property Value

InstrumentObjects.ODASensitivity.eGDOffsetMode

VB.NET Syntax

```
pdlaClient.Display.GDOffsetMode = _  
InstrumentObjects.ODASensitivity.eGDOffsetMode.CenterPoint
```

VB 6.0 Syntax

```
pdlaClient.Display.GDOffsetMode =  
eGDOffsetMode_CenterPoint
```

ToggleMode Method

Toggles the PDLA user interface graph display between one, two, and four graphs. This method corresponds to the Change View front panel key.

Parameters

ByVal mode As InstrumentObjects.ODACCommon.eLayoutMode - Defines the number of graphs that will be displayed on the 86038B user interface.

VB.NET Syntax

```
pdlaClient.Display.ToggleMode _  
(InstrumentObjects.ODACCommon.eLayoutMode.QUAD_Graph)
```

VB 6.0 Syntax

```
pdlaClient.Display.ToggleMode(eLayoutMode_QUAD_Graph)
```

ToggleRemoteIndicator Method

Use this method to return the PDLA to local control. This is equivalent to pressing the front panel Local key. To put the PDLA in local mode, call this method with a value of False.

Parameters

ByVal remote As Boolean - True sets the PDLA UI to remote mode. False sets the PDLA UI to local mode.

VB.NET Syntax

```
pdlaClient.Display.ToggleRemoteIndicator(True)
```

VB 6.0 Syntax

```
pdlaClient.Display.ToggleRemoteIndicator(True)
```

File Commands

The File property returns a file object that has methods to control the save and recall of data and instrument settings.

ExportCSV Method

Saves the measurement settings and raw data. This file is saved as an ASCII text file with a .csv extension. The data is comma separated and can be read directly into spreadsheets and analysis packages such as Matlab[®] or Excel[®].

An error will be raised if the specified path does not exist on the PDLA. Also, any existing file by the same name will be overwritten.

Parameters

ByVal filename As String - Full filename and path of the export file. The file will be saved on the PDLA system itself.

VB.NET Syntax

```
pdlaClient.File.ExportCSV("D:\Data\MyExport.csv")
```

VB 6.0 Syntax

```
pdlaClient.File.ExportCSV("D:\Data\MyExport.csv")
```

ExportMuellerPMD Method

Saves the PMD vector data and the first row of the Mueller matrix of the measured device. These data is only available if the appropriate measurement type is selected. The data is saved as an ASCII text file with a .pme extension. The data is comma separated. An error will be raised if the specified path does not exist on the PDLA. Also, an existing file with the same name will be overwritten. See also: GenerateMuellerAndPMDData Property

Parameters

ByVal filename As String - Full filename and path of the export file.

The file will be saved on the PDLA system itself. If the name does not contain the .pme extension it is automatically appended.

VB.NET Syntax

```
pdlaClient.File.ExportMuellerPMD("D:\Data\MyExport.pme")
```

VB 6.0 Syntax

```
pdlaClient.File.ExportMuellerPMD("D:\Data\MyExport.pme")
```

ExportStatusLog Method

Saves the Status Log of the PDLA into a file. The data is saved as an ASCII text file with a .log extension. An error will be raised if the specified path does not exist on the PDLA. Also, an existing file with the same name will be overwritten.

Parameters

ByVal filename As String - Full filename and path of the export file.

The file will be saved on the PDLA system itself. If the name does not contain the .log extension it is automatically appended.

VB.NET Syntax

```
pdlaClient.File.ExportStatusLog("D:\Data\MyExport.log")
```

VB 6.0 Syntax

```
pdlaClient.File.ExportStatusLog("D:\Data\MyExport.log")
```

Recall Method

Recalls a previously saved instrument state, including settings, trace data, and normalization data. An error will be raised if the specified file name does not exist on the PDLA.

Parameters

ByVal filename As String - Full filename and path of the PDLA file. The file must exist on the PDLA system.

VB.NET Syntax

```
pdlaClient.File.Recall("D:\Data\MyState.oda")
```

VB 6.0 Syntax

```
pdlaClient.File.Recall("D:\Data\MyState.oda")
```

Save Method

Saves the current instrument settings, trace data points, and normalization data. The file is saved with a .pdla extension. When this file is recalled using the Recall method, all the setup parameters will be set to the saved conditions.

An error will be raised if the specified path does not exist on the PDLA. Also, any existing file by the same name will be overwritten.

Parameters

ByVal filename As String - Full filename and path of the PDLA file. The file will be saved on the PDLA system itself.

VB.NET Syntax

```
pdlaClient.File.Save("D:\Data\MyState.oda")
```

VB 6.0 Syntax

```
pdlaClient.File.Save("D:\Data\MyState.oda")
```

Length Commands

The Length property of the remote client object returns a length object with methods and properties to control the measurement of fiber length.

IndexOfRefraction Property

Sets and reads the IndexOfRefraction for use with length measurements when Mode = Measure. The length of the test device is based on specialized group delay measurements and a user-defined refractive index value. This feature is intended for fiber and cable test applications.

Property Value

Double

Related Command

Length.ValidIndexOfRefraction method

VB.NET Syntax

```
pdlaClient.Length.IndexOfRefraction = 1.50
```

VB 6.0 Syntax

```
pdlaClient.Length.IndexOfRefraction = 1.50
```

Mode Property

Determines if the PDLA should use the user-specified value for length or the measured value.

Property Value

InstrumentObjects.ODALength.eLengthMode

eLengthMode.Measure - The PDLA will measure length based on the index of refraction.

eLengthMode.Specify - The PDLA will use the Specified Length property to set length.

VB.NET Syntax

```
pdlaClient.Length.Mode = _  
InstrumentObjects.ODALength.eLengthMode.Measure
```

VB 6.0 Syntax

```
pdlaClient.Length.Mode=eLengthMode_Measure
```

SpecifiedLength Property

Specifies the length of the fiber-optic cable (in meters) for the specified port. The length will also be used in computing CD and PMD coefficients.

Parameters

ByVal port As InstrumentObject.ODACCommon.eODAPort

Property Value

Double

Related Command

Length.ValidSpecifiedLength method

VB.NET Syntax

```
pdlaClient.Length.SpecifiedLength(InstrumentObject.ODACom  
mon.eODAPort.One)=3.8
```

VB 6.0 Syntax

```
pdlaClient.Length.SpecifiedLength(InstrumentObject.ODACom  
mon.eODAPort_One)=3.8
```

ValidIndexOfRefraction Method

Returns valid minimum, maximum, and default values of index of refraction for length measurements.

Parameters

ByRef Minimum As Double- holds minimum index of refraction.

ByRef Maximum As Double - holds maximum index of refraction.

ByRef DefaultValue As Double - holds default index of refraction.

Related Command

Length.IndexOfRefraction property

VB.NET Syntax

Dim minN As Double

Dim maxN As Double

Dim defaultN As Double

pdlaClient.Length.ValidIndexOfRefraction (minN, maxN, defaultN)

VB 6.0 Syntax

Dim minN As Double

Dim maxN As Double

Dim defaultN As Double

Call pdlaClient.Length.ValidIndexOfRefraction (minN, maxN, defaultN)

ValidSpecifiedLength Method

Returns valid minimum, maximum, and default values for the specified length of a fiber-optic cable.

Parameters

ByRef Minimum As Double - holds minimum specified length (in meters).

ByRef Maximum As Double - holds maximum specified length (in meters).

ByRef DefaultValue As Double - the default specified length (in meters).

Related Command

Length.SpecifiedLengthPort1 property and
Length.SpecifiedLengthPort2 property

VB.NET Syntax

Dim minLEN As Double

Dim maxLEN As Double

Dim defaultLEN As Double

pdlaClient.Length.ValidSpecifiedLength(minLEN, maxLEN,
defaultLEN)

VB 6.0 Syntax

Dim minLEN As Double

Dim maxLEN As Double

Dim defaultLEN As Double

Call pdlaClient.Length.ValidSpecifiedLength(minLEN, maxLEN,
defaultLEN)

MeasurementRange Commands

The MeasurementRange property of the remote client object returns a measurement range object that contains the functionality of the front panel Measurement Range panel.

ValidStartStop Method

Returns valid minimum and maximum wavelength values for the measurement.

Parameters

ByRef Minimum As Double - minimum wavelength accepted, in nm.

ByRef Maximum As Double - maximum wavelength accepted, in nm.

Related Command

MeasurementRange.XStart property, MeasurementRange.XStop property

VB.NET Syntax

```
Dim minWL As Double
```

```
Dim maxWL As Double
```

```
pdlaClient.MeasurementRange.ValidStartStop (minWL, maxWL)
```

VB 6.0 Syntax

```
Dim minWL As Double
```

```
Dim maxWL As Double
```

```
Call pdlaClient.MeasurementRange.ValidStartStop (minWL, maxWL)
```

XStart Property

Sets and reads the start wavelength for the next measurement.

Property Value

Double - The start wavelength in nanometers.

Related Command

MeasurementRange.ValidStartStop method

VB.NET Syntax

```
Dim myStart As Double  
myStart = pdlaClient.MeasurementRange.XStart  
pdlaClient.Measurement _  
Range.XStart = myStart -1.0
```

VB 6.0 Syntax

```
Dim myStart As Double  
myStart = pdlaClient.MeasurementRange.XStart  
pdlaClient.Measurement _  
Range.XStart = myStart -1.0
```

XStop Property

Sets and reads the stop wavelength for the next measurement.

Property Value

Double - The stop wavelength in nanometers.

Related Command

MeasurementRange.ValidStartStop method

VB.NET Syntax

```
Dim myStop As Double  
myStop = pdlaClient.MeasurementRange.XStop  
pdlaClient.Measurement _  
Range.XStop = myStop +1.0
```

VB 6.0 Syntax

```
Dim myStop As Double  
myStop = pdlaClient.MeasurementRange.XStop  
pdlaClient.Measurement _  
Range.XStop = myStop +1.0
```


NormalizationRange Commands

The NormalizationRange property of the remote client object returns a normalization range object that controls most of the settings unique to normalization. Those settings unique to a stepped normalization are in Normalization.Stepped, those unique to swept normalization are in Normalization.Swept.

ClearGDAndInsertionLossOffsets Method

Clears the GD and Insertion loss offsets for the specified port.

Parameters

By Val PortNum As InstrumentObjects.ODACCommon.eODAPort - The port on which to clear GD and Gain/Loss normalization data.

VB.NET Syntax

```
pdlaClient.NormalizationRange.ClearGDAndInsertionLossOffsets _  
(InstrumentObjects.ODACCommon.eODAPort.One)
```

VB 6.0 Syntax

```
pdlaClient.NormalizationRange.ClearGDAndInsertionLossOffsets _  
(eODAPort_One)
```

ClearModBias Method

Clears the current optical modulator bias value.

VB.NET Syntax

```
pdlaClient.NormalizationRange.ClearModBias()
```

VB 6.0 Syntax

```
pdlaClient.NormalizationRange.ClearModBias
```

ClearPDA Method

Clears the current phase dynamic accuracy data. This is automatically done when a normalization is performed with PDAEnabled set to True.

VB.NET Syntax

```
pdlaClient.NormalizationRange.ClearPDA()
```

VB 6.0 Syntax

```
pdlaClient.NormalizationRange.ClearPDA
```

ClearInputPolarizerAngle Method

Clears the current setting of the input polarizer. Use this to force a new input polarizer optimization with the next normalization measurement. After clearing the input polarizer angle, no measurements can be made until an input polarizer optimization is performed.

VB.NET Syntax

```
pdlaClient.NormalizationRange.ClearInputPolarizerAngle()
```

VB 6.0 Syntax

```
pdlaClient.NormalizationRange.ClearInputPolarizerAngle
```

PDAEnabled Property

Phase dynamic accuracy corrects for amplitude-to-phase conversion effects in the instrument receiver. If your test device exhibits no loss variation across the wavelength range, then disable this function to speed up the normalization time. You can read or set the value.

Property Value

Boolean - True if PDA is enabled.

VB.NET Syntax

```
pdlaClient.NormalizationRange.PDAEnabled = True
```

VB 6.0 Syntax

```
pdlaClient.NormalizationRange.PDAEnabled = True
```

UseCustomNormalization Property

Instructs the PDLA to use the XStart, XStop, Increment, and TracePoints from the NormalizationRange. These settings are therefore independent from the settings specified for the measurement. PDAEnabled is not affected by this property.

Property Value

Boolean - True tells the PDLA to use the NormalizationRange settings, False tells the PDLA to use Measurement settings.

VB.NET Syntax

```
pdlaClient.NormalizationRange.UseCustomNormalization =  
True
```

VB 6.0 Syntax

```
pdlaClient.NormalizationRange.UseCustomNormalization =  
True
```

ValidStartStop Method

Returns valid minimum and maximum wavelength values for normalization, in nm.

Parameters

ByRef Minimum As Double - The smallest valid wavelength, in nm.

ByRef Maximum As Double - The largest valid wavelength, in nm.

Related Command

NormalizationRange.XStart property,
NormalizationRange.XStop property

VB.NET Syntax

```
Dim minWL As Double
```

```
Dim maxWL As Double
```

```
pdlaClient.NormalizationRange.ValidStartStop(minWL,  
maxWL)
```

VB 6.0 Syntax

```
Dim minWL As Double
```

```
Dim maxWL As Double
```

```
Call pdlaClient.NormalizationRange.ValidStartStop(minWL,  
maxWL)
```

XStart Property

Sets and reads the start wavelength for the next normalization.

Property Value

Double - The start wavelength in nanometers.

Related Command

NormalizationRange.XStop property

VB.NET Syntax

```
pdlaClient.NormalizationRange.UseCustomNormalization =  
True
```

```
pdlaClient.NormalizationRange.XStart = 1550.0
```

VB 6.0 Syntax

```
pdlaClient.NormalizationRange.UseCustomNormalization =  
True
```

```
pdlaClient.NormalizationRange.XStart = 1550.0
```

XStop Property

Sets and reads the stop wavelength for the next normalization.

Property Value

Double - The stop wavelength in nanometers.

Related Command

NormalizationRange.XStart property

VB.NET Syntax

```
pdlaClient.NormalizationRange.UseCustomNormalization =  
True
```

```
pdlaClient. NormalizationRange.XStop = 1560.0
```

VB 6.0 Syntax

```
pdlaClient.NormalizationRange.UseCustomNormalization =  
True
```

```
pdlaClient. NormalizationRange.XStop = 1560.0
```

Increment Property

Specifies the separation between adjacent measurement wavelengths.

Property Value

Double - The wavelength spacing between points, in nanometers.

Related Command

NormalizationRange.ValidIncrement method

VB.NET Syntax

```
pdlaClient.NormalizationRange.Increment =0.5
```

VB 6.0 Syntax

```
pdlaClient.NormalizationRange.Increment =0.5
```

ValidIncrement Method

Returns valid minimum, maximum, and default values for the increment, in nanometers.

Parameters

ByRef Minimum As Double - Minimum X-axis increment is returned.

ByRef Maximum As Double - Maximum X-axis increment is returned.

ByRef DefaultValue As Double - Default X-axis increment is returned.

Related Command

NormalizationRange.Increment property

VB.NET Syntax

```
Dim minXinc As Double
```

```
Dim maxXinc As double
```

```
Dim defXinc As Double
```

```
pdlaClient.NormalizationRange.ValidIncrement _  
(minXinc, maxXinc, defXinc)
```

VB 6.0 Syntax

```
Dim minXinc As Double
```

```
Dim maxXinc As double
```

```
Dim defXinc As Double
```

```
Call pdlaClient.NormalizationRange.ValidIncrement _  
(minXinc, maxXinc, defXinc)
```

Printing Commands

The Printing property of the remote client object returns an object which allows the graphical display to be printed.

PrintScreen Method

Prints a copy of the graphical display and pertinent instrument settings to the default printer. This method is equivalent to pressing the front panel Print key.

VB.NET Syntax

```
pdlaClient.Printing.PrintScreen()
```

VB 6.0 Syntax

```
pdlaClient.Printing.PrintScreen
```

Resolution Commands

The Resolution property of the remote client object returns an object, which handles the resolution of the measurement.

RFModulationFrequency Property

Specifies the RF modulation frequency used to modulate the optical signal in the PDLA test set.

In general, a high modulation frequency minimizes instrumentation noise and a low modulation frequency maximizes wavelength resolution.

Property Value

Double - RF Modulation Frequency in MHz

Related Command

Resolution.ValidRFModulationFrequency method

VB.NET Syntax

```
pdlaClient.Resolution.RFModulationFrequency = 2000
```

VB 6.0 Syntax

```
pdlaClient.Resolution.RFModulationFrequency = 2000
```

ValidRFModulationFrequency Method

Returns valid minimum, maximum, and default RF modulation frequency values for the measurement.

Parameters

ByRef Minimum As Double - Minimum RF Modulation Frequency allowed, in MHz

ByRef Maximum As Double - Maximum RF Modulation Frequency allowed, in MHz

ByRef Default Value As Double - Default RF Modulation Frequency used, in MHz.

Related Command

Resolution.RFModulationFrequency property

VB.NET Syntax

Dim Min As Double

Dim Max As Double

Dim Default As Double

pdlaClient.Resolution.ValidRFModulationFrequency (Min, Max, Default)

VB 6.0 Syntax

Dim Min As Double

Dim Max As Double

Dim Default As Double

Call pdlaClient.Resolution.ValidRFModulationFrequency (Min, Max, Default)

Increment Property

Specifies the separation between adjacent measurement wavelengths.

Property Value

Double - The wavelength spacing between points, in nanometers.

Related Command

Resolution.ValidIncrement method, Resolution.ValidTracePoints method

VB.NET Syntax

pdlaClient.Resolution.Increment = .63

VB 6.0 Syntax

pdlaClient.Resolution.Increment = .63

ValidIncrement Method

Returns valid minimum, maximum, and default values for the wavelength step size, in nanometers.

Parameters

ByRef Minimum As Double - Minimum X-axis increment is returned.

ByRef Maximum As Double - Maximum X-axis increment is returned.

ByRef DefaultValue As Double - Default X-axis increment is returned.

Related Command

Resolution.Increment property, Resolution.ValidTracePoints method

VB.NET Syntax

```
Dim minXinc As Double
```

```
Dim maxXinc As double
```

```
Dim defXinc As Double
```

```
pdlaClient.Resolution.ValidIncrement (minXinc, maxXinc,  
defXinc)
```

VB 6.0 Syntax

```
Dim minXinc As Double
```

```
Dim maxXinc As double
```

```
Dim defXinc As Double
```

```
Call pdlaClient.Resolution.ValidIncrement _  
(minXinc, maxXinc, defXinc)
```

ValidTracePoints Method

Returns valid minimum and maximum trace point values.

Parameters

ByRef Minimum As Double - Minimum number of trace points is returned



ByRef Maximum As Double - Maximum number of trace points is returned

Related Command

Resolution.Increment property, Resolution.ValidIncrement method

VB.NET Syntax

```
Dim MinXtrc As Double
```

```
Dim MaxXtrc As Double
```

```
pdlaClient.Resolution.ValidTracePoints (MinXtrc, MaxXtrc)
```

VB 6.0 Syntax

```
Dim MinXtrc As Double
```

```
Dim MaxXtrc As Double
```

```
Call pdlaClient.Resolution.ValidTracePoints (MinXtrc, MaxXtrc)
```

Results Commands

The Results property of the remote client object returns an object that provides you access to the trace data and the x-axis values.

FiberLength Method

Retrieves the measured fiber length after a length measurement is completed.

Parameters

ByVal MeasNormType As InstrumentObjects.ODACCommon.eMeasNormData Type - The measurement type.

ByVal PortNum As InstrumentObjects.ODACCommon.eODAPort. - The port number.

Returns

Double - Returns the fiber length.

VB.NET Syntax

```
Dim Length As Double
Length = pdlaClient.Results.FiberLength _
(InstrumentObjects.ODACCommon.eMeasNormDataType.Measur
ement, _
InstrumentObjects.ODACCommon.eODAPort.One)
```

VB 6.0 Syntax

```
Dim Length As Double
Length = pdlaClient.Results.FiberLength _
(eMeasNormDataType_Measurement, eODAPort_One)
```

Increment Property

Returns the wavelength step size (or increment), in nm, for the last measurement. You can build the wavelength array with

$\lambda = XStart + (i - 1) \times Increment$, $i = 1$ to number of points

Property Value

Double - The x-axis increment

VB.NET Syntax

```
Dim XInc As Double
```

```
XInc = pdlaClient.Results.Increment
```

VB 6.0 Syntax

```
Dim XInc As Double
```

```
XInc = pdlaClient.Results.Increment
```

XStart Method

Returns the Start wavelength for the last measurement of the specified type, in nm. Note that because of the inherent derivative to calculate dispersion, the dispersion XStart differs from the GD and Gain XStart by one half of an increment.

Returns

Double - The start wavelength for the last measurement, in nanometers

VB.NET Syntax

```
Dim sVal As Double
```

```
sVal = pdlaClient.Results.XStart _
```

```
(InstrumentObjects.ODACCommon.eMeasurementType.Gain)
```

VB 6.0 Syntax

```
Dim sVal As Double
```

```
sVal = pdlaClient.Results.XStart (eMeasurementType_Gain)
```

XStop Method

Returns the stop wavelength for the last measurement of the specified type, in nm.

Returns

Double - The stop wavelength for the last measurement in nanometers.

VB.NET Syntax

```
Dim xStop As Double  
xStop = pdlaClient.Results.XStop_  
(InstrumentObjects.ODACCommon.eMeasurementType.Gain)
```

VB 6.0 Syntax

```
Dim xStop As Double  
xStop = pdlaClient.Results.XStop(eMeasurementType_Gain)
```

XValueFromPointIndex Method

Calculates the wavelength for the specified trace point. This can also be calculated from XStart and Increment.

Parameters

```
ByVal MeasurementType As  
InstrumentObjects.ODACCommon.eMeasurementType  
ByVal PointIndex As Integer
```

Returns

Double - Returns the wavelength of the point index passed to the method.

VB.NET Syntax

```
Dim xVal As Double  
xVal = pdlaClient.Results.XValueFromPointIndex_  
(InstrumentObjects.ODACCommon.eMeasurementType.Gain,2)
```

VB 6.0 Syntax

```
Dim xVal as Double  
xVal = pdlaClient.Results.XValueFromPointIndex_  
(eMeasurementType_Gain,2)
```

YData Method

Returns the data array for the last measurement for the specified measurement type and port. Use UBound for VB 6.0 or Length for VB.NET to find the number of trace points, and XStart and Increment to find the wavelengths

Parameters

ByVal MeasurementType As InstrumentObjects.ODACCommon.eMeasurementType - The measurement data type to retrieve.

ByVal PortNum As InstrumentObjects.ODACCommon.eODAPort - Port number to retrieve the data.

Returns

Double() - The raw data array.

VB.NET Syntax

```
Dim rawData() As Double  
rawData = pdlaClient.Results.YData _  
(InstrumentObjects.ODACCommon.eMeasurementType.Gain, _  
InstrumentObjects.ODACCommon.eODAPort.One)
```

VB 6.0 Syntax

```
Dim rawData() As Double  
rawData = pdlaClient.Results.YData(eMeasurementType_Gain, _  
_  
eODAPort_One)
```

Sensitivity Commands

DriftCorrection Property

For Stepped measurement modes and polarization resolved Swept measurement modes only

Sets/Gets the DriftCorrection value. This system function reduces measurement error contributed by gradual, ambient temperature changes that change the response of the DUT or the measurement system itself. It is appropriate for use when the temperature of the room is gradually rising or falling, or when the test device gradually transitions from one temperature to another. This function is not suited to rapid or stepped temperature changes. The movement of air over the test path can generate rapid, small group delay changes. Reduce this type of error by covering the test path to block air currents. Turning this function on will approximately double your test time for stepped measurements.

Property Value

Boolean = True if enabled

VB.NET Syntax

```
If(pdlaClient.Sensitivity.DriftCorrection = False) Then  
    pdlaClient.Sensitivity.DriftCorrection = True  
End If
```

VB 6.0 Syntax

```
If(pdlaClient.Sensitivity.DriftCorrection = False) Then  
    pdlaClient.Sensitivity.DriftCorrection = True  
End If
```

IFBandwidth Property

Sets the bandwidth of the filtering process that trades off measurement noise floor (sensitivity) against measurement speed. A lower IF bandwidth value means lower noise (high sensitivity) and a longer measurement time. Reduce the IF bandwidth as needed to control trace noise. This is particularly important when measuring a high insertion loss device.

Property Value

InstrumentObjects.ODASensitivity.eIFBandwidth

VB.NET Syntax

```
pdlaClient.Sensitivity.IFBandwidth = _  
InstrumentObjects.ODASensitivity.eIFBandwidth.IFBW_100Hz
```

VB 6.0 Syntax

```
pdlaClient.Sensitivity.IFBandwidth=eIFBandwidth_IFBW_100H  
z
```

SamplesPerPoint Property

Sets the number of amplitude and phase measurements (samples) that are taken at each measurement wavelength (trace point).

In Swept measurement mode, the recovered modulation amplitude and phase are sampled periodically as the wavelength is swept. As a result, the wavelength resolution of the measurement is limited by wavelength range/(number of points per sweep – 1). Note that wavelength resolution is also affected by the modulation frequency.

In Stepped measurement mode, you select how many samples to take at each measurement wavelength. The multiple samples are averaged to generate a single trace point. A larger number of points means more averaging, suitable for reducing trace noise in cases of high insertion loss test devices.

Property Value

Integer - The number of samples to take at each point.

Related Command

Sensitivity.ValidSamplesPerPoint method

VB.NET Syntax

```
pdlaClient.Sensitivity.SamplesPerPoint = 201
```

VB 6.0 Syntax

```
pdlaClient.Sensitivity.SamplesPerPoint = 201
```

ValidSamplesPerPoint Method

Returns valid minimum, maximum, and default values for the SamplesPerPoint.

Parameters

ByRef Minimum As Integer - Minimum Samples per point is returned.

ByRef Maximum As Integer - Maximum Samples per point is returned.

ByRef DefaultValue As Integer - Default Samples per point for the PDLA.

Related Command

Sensitivity.SamplesPerPoint property

VB.NET Syntax

```
Dim minSamples As Integer
Dim maxSamples As Integer
Dim defSamples As Integer
pdlaClient.Sensitivity.ValidSamplesPerPoint _
(minSamples, maxSamples, defSamples)
```

VB 6.0 Syntax

```
Dim minSamples As Integer
Dim maxSamples As Integer
Dim defSamples As Integer
Call pdlaClient.Sensitivity.ValidSamplesPerPoint _
(minSamples, maxSamples, defSamples)
```

Status Commands

ErrorFlag Property

Sets the value to True when an error has occurred. Reading the ErrorFlag automatically sets the flag back to False.

Property Value

Boolean - True when an error has occurred.

VB.NET Syntax

```
If(pdlaClient.Status.ErrorFlag = True) Then  
    MsgBox("An error occurred.")  
End If
```

VB 6.0 Syntax

```
If(pdlaClient.Status.ErrorFlag = True) Then  
    MsgBox("An error occurred.")  
End If
```

EstimatedTimeToTLSSettling Property

Indicates the estimated time until the next TLSSettling begins.

The PDLA relies on the tunable laser to provide high resolution and dynamic range. The laser has a mechanically tuned cavity that is sensitive to temperature change, requiring the laser to track the internal temperature and occasionally perform an adjustment. Under normal temperature conditions, this value may show 120 minutes as the estimate. This value is updated every minute.

Property Value

Double - The estimated time in minutes to the next TLS Settling.

Related Command

TLS.ForceSettling method, Status.TLSSettling property

VB.NET Syntax

```
pdlaClient.Status.EstimatedTimeToTLSSettling
```

VB 6.0 Syntax

```
pdlaClient.Status.EstimatedTimeToTLSSettling
```

InstrumentState Property

Indicates the current state of the PDLA, that is, idle, initializing, measuring, TLS Settling, or aborting. Refer to [“PDLA Enumerations” on page 150](#) for possible values.

Property Value

InstrumentObjects.ODACCommon.eInstrumentStatus - Denotes the current instrument state.

VB.NET Syntax

```
Dim curStatus As  
InstrumentObjects.ODACCommon.eInstrumentStatus  
curStatus = pdlaClient.Status.InstrumenState
```

VB 6.0 Syntax

```
Dim curStatus as InstrumentObjects.eInstrumentStatus  
curStatus = pdlaClient.Status.InstrumentState
```

LastError Property

Returns a string describing the error. Reading the LastError automatically sets the property back to an empty string.

Property Value

String - A description of the error.

VB.NET Syntax

```
If(pdlaClient.Status.ErrorFlag = True) Then  
    MsgBox("Error:" + pdlaClient.Status.LastError)  
End If
```

VB 6.0 Syntax

```
If(pdlaClient.Status.ErrorFlag = True) Then  
    MsgBox("Error:" + pdlaClient.Status.LastError)
```

End If

PresetComplete Property

Sets the value to True when the preset is complete and False when it is not.

Property Value

Boolean - True when the Preset/Initialization is complete.

VB.NET Syntax

```
Dim PreComplete As Boolean
PreComplete = pdlaClient.Status.PresetComplete
```

VB 6.0 Syntax

```
Dim PreComplete As Boolean
PreComplete = pdlaClient.Status.PresetComplete
```

StatusLog Method

Returns an array of status log entries that appear as messages within the status log on the PDLA user interface.

Returns

InstrumentObjects.RemoteStatusEntry - An array of entries containing the status log information.

VB.NET Syntax

```
Dim ctr As Integer
Dim entries() As InstrumentObjects.RemoteStatusEntry
entries = pdlaClient.Status.StatusLog()
For ctr = 0 To entries.GetUpperBound(0)
    Console.WriteLine("Time:" + entries(ctr).TimeStamp +
"Summary:" + entries(ctr).Summary + "Type:" +
entries(ctr).MessageType.ToString + "Detail:" +
entries(ctr).Detail)
Next
```

VB 6.0 Syntax

```
Dim ctr As Integer
Dim entries() As RemoteStatusEntry
entries = pdlaClient.Status.StatusLog
For ctr = 0 to UBound (entries)
    Debug.Print(entries(ctr).Summary)
Next
```

TLSSettling Property

Checks if a TLS Settle is in progress.

Property Value

Boolean - True if the TLSSettling is in progress.

Related Command

TLS.ForceSettling method

VB.NET Syntax

```
Dim TLSIsSettling As Boolean
TLSIsSettling = pdlaClient.Status.TLSSettling
```

VB 6.0 Syntax

```
Dim TLSIsSettling As Boolean
TLSIsSettling = pdlaClient.Status.TLSSettling
```

TriggerComplete Property

Sets the value to True when the trigger is complete.

Property Value

Boolean - True when the last Measurement or Normalization is complete.

VB.NET Syntax

```
Dim TrigComplete As Boolean
TrigComplete = pdlaClient.Status.TriggerComplete
```

VB 6.0 Syntax

```
Dim TrigComplete As Boolean
```

```
TrigComplete = pdlaClient.Status.TriggerComplete
```

ValidNormalization Property

Sets the value to True when the PDLA settings fall within the current normalization boundaries.

Read-Only Property

Boolean - True when current PDLA settings fall within the current normalization boundaries.

VB.NET Syntax

```
Dim ValNorm As Boolean
```

```
ValNorm = pdlaClient.Status.ValidNormalization
```

VB 6.0 Syntax

```
Dim ValNorm As Boolean
```

```
ValNorm = pdlaClient.Status.ValidNormalization
```

System Commands

LaserState Method

Turns the laser on/off at the TLS output. The laser will turn back on when a measurement is started.

Parameters

OnOffState - Specifies the desired state of the laser.

VB.NET Syntax

```
pdlaClient.System.LaserState _  
(Instrument.Objects.ODACCommon.enumOnOff.eOn)
```

VB 6.0 Syntax

```
pdlaClient.System.LaserState (enumOnOff_eOn)
```

PowerMeterZero Method

Causes the external power meters, if installed, to perform the zero function.

VB.NET Syntax

```
pdlaClient.System.PowerMeterZero()
```

VB 6.0 Syntax

```
pdlaClient.System.PowerMeterZero
```

Preset Method

Sets the instrument to defined setup conditions. Preset can only be called when the PDLA is idle.

VB.NET Syntax

```
pdlaClient.System.Preset()
```

VB 6.0 Syntax

```
pdlaClient.System.Preset
```


WavelengthRealtime Property

Gets or sets the wavelength for the RealtimePortMonitor measurement. Set this wavelength prior to calling "RealtimePortMonitor"

Property Value

Double - The wavelength in nm

VB.Net Syntax

```
Dim realtimeWvl as Double  
realtimeWvl = pdlaClient.System.WavelengthRealtime
```

VB 6.0 syntax

```
Dim realtimeWvl as Double  
realtimeWvl = pdlaClient.System.WavelengthRealtime
```

SetPort Method

Activates the port for the RealtimePortMonitor measurement. Set the port prior to calling "RealtimePortMonitor". If the setup does not support multiport measurements, select port "One"

Parameters

eODAPort - The port to measure, only single ports are allowed

VB.Net Syntax

```
pdlaClient.System.SetPort(InstrumentObjects.ODACCommon.eODAPort.One)
```

VB 6.0 syntax

```
pdlaClient.System.SetPort(InstrumentObjects.ODACCommon.eODAPort_One)
```

RealtimePortMonitor Method

Starts a realtime monitor measurement at the specified wavelength and at the specified port. The data are transferred by the "NewRealTimePortMonitorData" Event

VB.Net Syntax

```
pdlaClient.System.RealtimePortMonitor()
```

VB 6.0 syntax

```
pdlaClient.System.RealtimePortMonitor()
```

System - Identity Commands

CompanyName Property

Returns Agilent Technologies as the company name of the 86038B PDLA.

Property Value

String - Company name.

VB.NET Syntax

```
Dim CompName As String
```

```
CompName = pdlaClient.System.Identity.CompanyName
```

VB 6.0 Syntax

```
Dim CompName As String
```

```
CompName = pdlaClient.System.Identity.CompanyName
```

ProductId Property

Returns the product number, 86038B.

Property Value

String - Product number

VB.NET Syntax

```
Dim ProdID As String
```

```
ProdID = pdlaClient.System.Identity.ProductID
```

VB 6.0 Syntax

```
Dim ProdID As String
```

```
ProdID = pdlaClient.System.Identity.ProductID
```

ProductName Property

Returns the product name.

Property Value

String - Product name.

VB.NET Syntax

```
Dim ProdName As String
```

```
ProdName = pdlaClient.System.Identity.ProductName
```

VB 6.0 Syntax

```
Dim ProdName As String
```

```
ProdName = pdlaClient.System.Identity.ProductName
```

ProductVersion Property

Returns the version of the PDLA system. The major and minor version numbers of the remote client and the PDLA system should match.

Property Value

String - Software version of PDLA system software.

VB.NET Syntax

```
Dim ProdVersion As String
```

```
ProdVersion = pdlaClient.System.Identity.ProductVersion
```

VB 6.0 Syntax

```
Dim ProdVersion As String
```

```
ProdVersion = pdlaClient.System.Identity.ProductVersion
```

RemotingVersion Property

Returns the software version number of the remoting software. The major and minor version numbers of the remote client and the PDLA system should match. A mismatch in versions may cause unexpected results.

Property Value

String - Version of remote client software.

VB.NET Syntax

```
Dim RemVersion As String
```

```
RemVersion = pdlaClient.System.Identity.RemotingVersion
```

VB.6.0 Syntax

```
Dim RemVersion As String
```

```
RemVersion = pdlaClient.System.Identity.RemotingVersion
```

TLS Commands

ForceSettling Method

Performs a TLS settling process. Use this method to force a TLSSettling when your application has free time rather than having a TLSSettling automatically start at an unexpected time.

Related Command

Status.EstimatedTimeToTLSSettling property,
Status.TLSSettling property

VB.NET Syntax

```
pdlaClient.TLS.ForceSettling()
```

VB 6.0 Syntax

```
pdlaClient.TLS.ForceSettling
```

PowerSelect Property

The setting of this property tells the PDLA which optical output of the laser source is to be used for measurements. Make sure that the specified output is the one which is also optically connected to the PDLA system.

Property Value

```
InstrumentObjects.ODATLS.eTLSPowerSelect
```

VB.NET Syntax

```
pdlaClient.TLS.PowerSelect =  
InstrumentObjects.ODATLS.eTLSPowerSelect.High
```

VB 6.0 Syntax

```
pdlaClient.TLS.PowerSelect = eTLSPowerSelect_High
```

MultiTlsSupport Method

Returns the number of valid tunable lasers found in the current setup.

Returns

Integer - the number of tunable lasers found in the current setup

Parameters

ByVal switchable As Boolean

If switchable = True, the minimum of the number of tunable lasers and the number of ports in the source switch is returned. If switchable = False, the number of tunable lasers is returned.

VB.NET Syntax

```
Dim numTLS As Integer  
numTLS = pdlaClient.TLS.MultiTlsSupport(True)
```

VB 6.0 Syntax

```
Dim numTLS As Integer  
numTLS = pdlaClient.TLS.MultiTlsSupport(True)
```

SelectTls Method

Use this method to select the laser source for the next measurement.

Returns

Boolean - True if everything worked well, False if idx is invalid

Parameters

ByVal idx As Integer

The zero-based index of the tunable laser which is to be used for the next measurement

VB.NET Syntax

```
Dim retVal As Boolean  
retVal = pdlaClient.TLS.SelectTls(0)
```

VB 6.0 Syntax

```
Dim retVal As Boolean  
retVal = pdlaClient.TLS.SelectTls(0)
```

TlsWvlRange Method

Use this method to query minimum and maximum wavelengths for the tunable laser sources.

Returns

Boolean - True if a tunable laser source with the specified index exists

Parameters

ByVal idx As Integer

The zero-based index of the tunable laser for which the wavelengths are queried

ByRef minWvl As Double

Holds the minimum wavelength of the tunable laser in 'nm'

ByRef maxWvl As Double

Holds the maximum wavelength of the tunable laser in 'nm'

VB.NET Syntax

```
Dim retVal As Boolean  
Dim minWvl As Double  
Dim maxWvl As Double  
retVal = pdlaClient.TLS.TlsWvlRange(0, minWvl, maxWvl)
```

VB 6.0 Syntax

```
Dim retVal As Boolean  
Dim minWvl As Double  
Dim maxWvl As Double  
retVal = pdlaClient.TLS.TlsWvlRange(0, minWvl, maxWvl)
```


TlsWvlRangeString Method

Use this method to query minimum and maximum wavelengths for the tunable laser sources.

Returns

String - The minimum and maximum wavelengths as a string in the format "min ... max nm"

Parameters

ByVal idx As Integer

The zero-based index of the tunable laser for which the wavelengths are queried

ByRef exists As Boolean

True if a tunable laser source with the specified index exists

VB.NET Syntax

Dim exists As Boolean

Dim wvlString As String

wvlString = pdlaClient.TLS.TlsWvlRangeString(0, exists)

VB 6.0 Syntax

Dim exists As Boolean

Dim wvlString As String

wvlString = pdlaClient.TLS.TlsWvlRangeString(0, exists)

SelectedTLSIdx Method

This method returns the index of the currently selected tunable laser source.

Returns

Integer - The index of the currently selected tunable laser source

VB.NET Syntax

Dim idx As Integer

idx = pdlaClient.TLS.SelectedTLSIdx()

VB 6.0 Syntax

```
Dim idx As Integer  
idx= pdlaClient.TLS.SelectedTLSEIdx
```

TlsIdn Method

This method returns the identifier of the tunable laser source as a string. The identifier string contains the manufacturer name, product number and serial number.

Returns

String - the identifier of the tunable laser source

Parameters

ByVal idx As Integer

The zero-based index of the tunable laser for which the identifier is queried

ByRef exists As Boolean

True if a tunable laser source with the specified index exists

VB.NET Syntax

```
Dim exists As Boolean  
Dim idn As String  
idn = pdlaClient.TLS.TlsIdn(0, exists)
```

VB 6.0 Syntax

```
Dim exists As Boolean  
Dim idn As String  
idn = pdlaClient.TLS.TlsIdn(0, exists)
```

TlsPn Method

This method returns the product number of the tunable laser source as a string.

Returns

String - the product number of the tunable laser source

Parameters

ByVal idx As Integer

The zero-based index of the tunable laser for which the product number is queried

ByRef exists As Boolean

True if a tunable laser source with the specified index exists

VB.NET Syntax

Dim exists As Boolean

Dim prodnumber As String

prodnumber = pdlaClient.TLS.TlsPn(0, exists)

VB 6.0 Syntax

Dim exists As Boolean

Dim prodnumber As String

prodnumber = pdlaClient.TLS.TlsPn(0, exists)

TlsSn Method

This method returns the serial number of the tunable laser source as a string.

Returns

String - the serial number of the tunable laser source

Parameters

ByVal idx As Integer

The zero-based index of the tunable laser for which the serial number is queried

ByRef exists As Boolean

True if a tunable laser source with the specified index exists

VB.NET Syntax

Dim exists As Boolean

Dim serialnumber As String

serialnumber = pdlaClient.TLS.TlsSn(0, exists)

VB 6.0 Syntax

Dim exists As Boolean

Dim serialnumber As String

serialnumber = pdlaClient.TLS.TlsSn(0, exists)

HighPowerLimit Property

Gets or sets the upper limit of the TLS output power used for any measurement.

Property Value

Double - the TLS output power in dBm

VB.NET Syntax

Dim PowerLimit as Double

PowerLimit = pdlaClient.TLS.HighPowerLimit

VB 6.0 Syntax

Dim PowerLimit as Double

PowerLimit = pdlaClient.TLS.HighPowerLimit

7

Troubleshooting and Upgrades

Introduction	286
Troubleshooting Startup Problems	287
Troubleshooting GPIB Communications Problems 289	
Troubleshooting Problems with Remote Programming	293
Troubleshooting Sweep and Normalization Problems	294
Troubleshooting Incorrect Data	297
Troubleshooting the TLS	299
Software Upgrade Procedure	301
Hard Drive Backup and Recovery	302
Cleaning Information	304
Returning the Instrument for Service	326

Introduction

This chapter begins by explaining how to troubleshoot basic problems with the PDLA computer system and hardware. It then explains how to solve some common measurement problems. The rest of the chapter covers other important customer support information.

Please note that the following acronyms are used:

- PDLA: Optical Dispersion Analyzer (Agilent 86038B)
- PNA: Performance Network Analyzer
- TLS: Tunable Laser Source
- MWM: Multi-wavelength Meter

Troubleshooting Startup Problems

If the PDLA does not boot up to MS Windows XP

1 Check Connections

Ensure that the PNA is plugged in. If an external monitor is being used, ensure it is plugged in and connected to the PNA. If this does not solve the problem, continue with the next step.

2 Reload the C: Drive Image

The Agilent PDLA has the ability to backup and restore a hard drive image, in case of operating system error. If the PDLA is unable to boot into the Windows XP option screen, please refer to [“Hard Drive Backup and Recovery” on page 302](#). If this does not solve the problem, continue with the next step.

3 Replace the PNA Disc Drive

If the system does not restore from the hard disk recovery image, the PNA Disc Drive may need to be replaced.

If the PDLA Application does not automatically launch

The PDLA Application should launch automatically after Windows XP boots. If not, the application can be launched from the PDLA-UI shortcut on the desktop, or from Start > Programs > Photonic Dispersion and Loss Analyzer > Photonic Dispersion and Loss Analyzer.

If you are able to successfully launch the PDLA application in this way, it is possible that AutoStart for the application has been disabled. To re-enable AutoStart, go to C:\Program Files\Agilent\Photonic Dispersion and Loss Analyzer\Config, and double click on the "toggle PDLA autostart.vbs" link. A message will appear stating the status of AutoStart (enabled or disabled) with the option to change. Ensure that AutoStart is enabled.

If the PDLA cannot communicate with the TLS or MWM

When the application launches, it attempts to contact the tunable laser source, the test set, and the multi-wavelength meter (when applicable). If there is any difficulty in contacting these instruments, error messages will be displayed and the application may not launch fully. There is a 30-minute time-out for contacting the TLS as the TLS takes at least 12 minutes to fully boot.

If there is any difficulty in contacting these instruments, ensure that there is power to the test set, and refer to [“Troubleshooting GPIB Communications Problems”](#) on page 289.

Troubleshooting GPIB Communications Problems

1. Check the PDLA Status Log

If the PDLA application is running, check the PDLA Status Log for any warnings or errors. Press Status Log to view a list of status messages.

Highlighting a message in the log may cause additional details to be shown.

2. Ensure that GPIB cables are configured correctly.

Check the GPIB cables; they should match [Figure 2.b on page 1-34](#).

If the MWM is used, it must also be attached with another GPIB cable.

3. Check the system GPIB Addresses.

The system is configured to work with predetermined GPIB addresses for the tunable laser source and multi-wavelength meter (when applicable). If the GPIB addresses for these instruments have been changed, the PDLA application will not be able to communicate with them.

The factory default address for an 8164B and an 86122A is the same – 20. If both instruments are on the bus with the same GPIB address, you may get confusing error messages, likely a message that says the PDLA Application cannot contact the tunable laser source.

Follow the details below to check and reset the GPIB addresses on the tunable laser source and multi-wavelength meter. The GPIB address cannot be changed from the PDLA software, but must be altered on the individual instruments.

Instrument	Model #	GPIB Address	Path to set GPIB Address
Lightwave Measurement System with Tunable Laser Source Module	8164B / 81600B	20	Config > GPIB Address
Multi-Wavelength Meter	86122A	10	Setup > GPIB Setup

4. Check the tunable laser source and multi-wavelength meter firmware revision

The PDLA Application has been designed and tested to work with the 8164B Lightwave Measurement System and 81600B or 81640B Tunable Laser Source. An 86122A Multi-Wavelength Meter may also be added to improve the wavelength accuracy of the measurements. There is a minimum firmware revision required to work with the application. If the instrument is not the correct model number, or does not meet the minimum firmware revision requirements, the PDLA will not work with the instrument. Check the instrument's firmware revisions by following the steps below.

Instrument	Model #	Min F/W Rev.	To check firmware revision
Lightwave Measurement System	8164A or 8164B	V4.50	Local > Config > About Mainframe. The mainframe firmware revision information is on the bottom line.
Tunable Laser Source Module	81640B or 81600B	V4.10	Local > Config > About Modules > Slot 0 > Enter.
Multi-Wavelength Meter	86122A	1.0	Help > About 86122A. View under System Information > Software Revision

To upgrade the Instrument

The necessary files for upgrading the instruments are shipped on the OCT Support CD and are stored on the PDLA in the "Additional Drivers" folder. The most recent versions can also be found on the Agilent web site, under www.agilent.com/comms/octfirmware.

To upgrade the instruments, please follow the instructions in the User's Guide of each instrument. The instructions are also available from the Agilent web site. The upgrade process requires the use of a computer. For this purpose, an external computer or the PDLA itself can be used. To use the PDLA, first exit the PDLA application.

5. Ensure that the GPIB setting in the PNA software is set to talker / listener

The PDLA is based on Agilent's performance network analyzer platform. The two instruments use GPIB in different ways. In the PNA software, the GPIB on the instrument can be set as a talker / listener, so that the instrument can be controlled remotely over GPIB. The PNA software can also be set to control other instruments remotely over GPIB.

The PDLA uses the GPIB as a controller, to communicate with the tunable laser source and multi-wavelength meter. If the GPIB settings have been changed in the PNA software, the PDLA will no longer be able to communicate with the other instruments.

To check the GPIB setting in the PNA software, first exit out of the PDLA Application. Then, launch the PNA software from C:\Program Files\Agilent\Network Analyzer\835x.exe. When the PNA software loads, go to System > Configure > SICL/GPIB ... From the SICL/GPIB setup panel, ensure that under GPIB, the talker / listener radio button is selected, and the address is set to 16. Close the panel, and exit the PNA software. Open the PDLA application from the PDLA shortcut on the desktop, or from Start Menu > Programs > Photonic Dispersion and Loss Analyzer > Photonic Dispersion and Loss Analyzer.

6. Ensure that NI GPIB driver is enabled, Agilent GPIB driver is disabled.

The PDLA is based on Agilent's performance network analyzer platform. The two instruments use GPIB in different ways. Both Agilent GPIB TNT Interface and National Instruments GPIB Interface are present on the PDLA. In order for the PDLA to communicate with the tunable laser source and multi-wavelength meter, both of these device drivers must be present. In the device manager, (Start > Settings > Control Panel > System > Hardware > Device Manager), you will see a listing for GPIB Devices, under which the Agilent GPIB TNT Interface is listed. There will also be a listing for National Instruments GPIB Interfaces. When the GPIB Devices listing is expanded, a red X will appear in front of the Agilent GPIB TNT Interface. This is the correct configuration for the PDLA software to work properly. DO NOT ENABLE the Agilent GPIB interface in the Device Manager. Under National Instruments GPIB Interfaces,

the AT-GPIB/TNT interface will be enabled. This again is the correct configuration for the PDLA software to work properly. If the device settings have been changed, please restore them to the condition described above for the system GPIB to function properly.

7. Reboot the TLS

- 1 Exit the PDLA application (File > Exit >Yes).
- 2 Reboot the TLS.
- 3 Wait for the TLS to start up (it will take at least 12 minutes to fully boot). Check to see if there are any errors or requests for realignment.
- 4 Start the PDLA software.

8. Reboot the PNA

- 1 Exit the PDLA application (File > Exit > Yes).
- 2 Ensure that all of the instruments are at the proper GPIB addresses.
- 3 Reboot the PDLA.

9. Swap GPIB Cables

If the PDLA still cannot communicate with the TLS or MWM, try using different GPIB cables.



Troubleshooting Problems with Remote Programming

For troubleshooting problems with remote programming, refer to [“Troubleshooting” on page 157](#).

Troubleshooting Sweep and Normalization Problems

If the instrument initialization is successful, but the instrument is unable to make wavelength sweeps for the normalization and measurement, there may be a problem with the triggering from the tunable laser source. Confirm that the BNC cable is properly connected and functioning between the 8164B trigger output and the Optical Test Set trigger input.

The PDLA will not sweep if the Modulator Bias or Wavelength Correction does not complete. Refer to the steps below for whichever condition you suspect.

NOTE

If the application hangs during a measurement, restart the PDLA.

If the Modulator Bias does not complete

The Modulator Bias will not complete if the input power detected is too low. This will usually be indicated by a “No signal Detected” message in the status log.

- 1 Ensure that the test set power cord is plugged in.
- 2 Ensure that the TLS is properly connected. Use the Real Time Port Monitor to check the power at the Monitor port. Usually the Monitor Port will show about: -10dBm to -15dBm. Less power may indicate a problem with the fiber connection between TLS and Analyzer.
- 3 Perform System>Optimize Optical Modulator Bias and Optimize Input Polarizer. Check if the power level is improved with Real-Time Port Monitor.
- 4 Isolate the problem to the TLS or Analyzer. Refer to [“Test the High Power output on the TLS” on page 299](#).

If the Wavelength Correction does not complete

NOTE

Only applies if 86122A is installed

The wavelength correction will not complete if there is a low or missing signal.


- 1 Ensure that the test set power cord is plugged in.
- 2 Ensure that the MWM is plugged in and turned on. The MWMs GPIB cable should be attached to the rear of the PNA and have a GPIB address set to 10. If any of the above conditions are not met, correct the setup and then exit the PDLA software (File > Exit > Yes) and restart. The PDLA software will detect the MWM on power up, or by selecting Preset.
- 3 Make sure that there is a short patch fiber running from the Low SSE output of the TLS to the input on the MWM.
- 4 Look at the status area of the MWM to ensure that there are no errors displayed. If there are errors, refer to the MWM online Help.

If Mode Hops are reported in the Status Bar

Mode hops (small sudden shifts of the laser wavelength) can occur swept-mode measurements for wide wavelength ranges, some wavelength ranges near the end of the tunable laser range and for low laser power settings. Even when this occurs, the tunable laser wavelength is accurately determined at the trigger for each measurement sample. However since the swept measurement samples are integrated over the time between samples, the mode hop can influence some results. A typical result is a spike in the DGD data.

The 86038B records and reports the occurrence of any such mode hops. If one occurs during a sweep, the sweep is automatically repeated. If it occurs again, then it is assumed to be unavoidable and the measurement will be completed with a report of the mode hops. This report can be used to clarify that bad measurement points at these wavelengths are not due to device under test.

Please refer to the tunable laser specifications for the conditions under which mode-hop free continuous sweeps are specified.



If sweeps occur with more serious laser instabilities, the measurement will be cancelled. This may indicate conditions under which the laser cannot correctly operate or that it needs realignment or repair. If it is desired to allow such measurements to complete despite the instability, this can be chosen from the System>Config menu.

Troubleshooting Incorrect Data

If measured loss is very high or parts of the curve are missing

The optical power from the DUT may be too low. This will usually be indicated by a “Low Signal on DUT path” message in the status log.

- 1 Check the power level at the correct port with the rest-time port monitor. Assure that the DUT does not have excessive loss at the chosen wavelength.
- 2 If the power level is too low compared to the Monitor Port, considering the expected DUT loss, check and clean the optical connections.
- 3 If the TLS high power output indicator light will not turn on, the TLS may need realignment. Please refer to [“Restart the TLS and realign the TLS module” on page 299](#).

NOTE

The “Low Signal on DUT path” message will also appear if the DUT has high loss in part of the spectrum, like a filter. If desired, the message can be turned off from the: System>Configuration tab. Warnings will still appear for low Monitor Port power.

If wavelengths are incorrect, or measurements are incorrect at certain wavelengths

- 1 Reboot the PNA and TLS.
- 2 If the wavelength errors or wavelength dependent errors still exist, troubleshoot the TLS. Refer to [“Restart the TLS and realign the TLS module” on page 299](#).

If the GD and Gain/Loss data looks incorrect

- Refer to [“Choosing Setup Parameters” on page 61](#).

NOTE

If the Group Delay curve is steep, there is a possibility that the change in phase between two points is too large. This condition will result in a phase wrap. Refer to [“If the Group Delay measurement shows a sudden large offset” on page 298](#).

If the Group Delay measurement appears to be noisy

Make sure that the environment in which the system is placed is stable. Also keep in mind that the instrument requires a 90 minute warm-up. During that time, measurements will show drift. The instrument stability can be observed by attaching a patch fiber and running the Real-Time Port Monitor to check the power and stability. To run the Real-Time Port Monitor, attach a mouse and click on System->Real Time Port Monitor (F12).

If the Group Delay measurement shows a sudden large offset

Sudden offsets in group delay are typically the results of a phase wrap; refer to [“Relationship of setup parameters to phase wrapping” on page 63](#). Phase wraps occur when the phase change from one step to the next exceeds 360 degrees. To avoid phase wrapping, try decreasing the Modulation Frequency or Step Size. Remember that if the Modulation Frequency is changed, a new normalization needs to be performed.

The Auto Modulation frequency option will aid in avoiding phase wraps when measuring fiber spools in stepped mode. To enable Auto Modulation frequency, press the Sens/Res front panel key and put a check in the Auto Modulation Frequency box.

If GD ripple is being observed

If unexpected group delay ripple is observed, first consider the amplitude of the ripple. The scale might be zoomed-in to reveal features that have not been visible with other instruments due to trace noise.

Next, note the period of the ripple. If the wavelength increment can be reduced and the period of the ripple remains the same, then the ripple is not aliased. If the ripple period does change, then keep reducing the wavelength increment to reveal the true period. Once the true period is found, the cause of the ripple may be evident, such as an etalon cavity that could be present in the device under test.

A common problem that leads to GD ripple is dirty fiber connections..

Refer to [“Relationship of setup parameters to ripple amplitude accuracy” on page 68](#) for more information.

Troubleshooting the TLS

- If you suspect faulty TLS wavelength performance, refer to [“Restart the TLS and realign the TLS module” on page 299](#).
- If you suspect faulty TLS power performance, refer to [“Test the High Power output on the TLS” on page 299](#).
- Please refer also to the Tunable Laser Source User’s Guide and Lightwave Measurement System User’s Guide for further information.

Restart the TLS and realign the TLS module

- 1 Exit out of the PDLA application (File > Exit > Yes). Turn the TLS power off, then back on.
- 2 If the TLS needs realignment, a message box will be displayed. Follow the instructions to run the realignment. This process can take up to 45 minutes.
- 3 After the realignment is complete, restart the PDLA application by double clicking on the PDLA desktop icon, or navigating to Start > Programs > Photonic Dispersion and Loss Analyzer > Photonic Dispersion and Loss Analyzer.
- 4 If wavelength results are still incorrect, check the TLS output wavelength with a multi-wavelength meter. The TLS module may need to be replaced.

Test the High Power output on the TLS

- 1 Ensure the laser is off, and disconnect the patchcord from TLS output at the Source Input on the test set.
- 2 Connect TLS Output 2 to a power meter.
- 3 Set the power meter wavelength to the center of TLS wavelength range (e.g. 1550 nm), and the output power to 0 dBm and then -5 dBm.
 - If the TLS output power as read by the power meter matches the output power set on the TLS within 0.5 dB, the TLS is operating properly. Ensure that there is power to the test set. Carefully clean the test set connections, optimize the Modulator Bias and Input Polarizer while a patchcord is attached, and retest the system with the Real-Time Port Monitor. If the TLS power output is accurate and the test is reading low, the Test Set may need to be repaired.

- If the TLS output power is off by more than 0.5 dB, or there is no output signal, check the optical connections and finally replace the TLS module.

NOTE

Low SSE output on the TLS: If using a MWM for wavelength correction, you should test the Low SSE output on the TLS. Follow the steps above, using TLS Output 1. If there is a power discrepancy between the TLS and MWM, use the power meter to isolate where the failure lies.

Software Upgrade Procedure

The 86038B is built on the N5230A performance network analyzer (PNA-L).

Information about the current PNA-L firmware version and the PNA upgrade process can be found on this web page:
<http://na.tm.agilent.com/pna/firmware/firmware.htm>

To Upgrade the firmware on the PDLA:

- 1 Check the PDLA website for firmware upgrades:
www.agilent.com/comms/oct.
- 2 Download the latest software revision. For some upgrades both the PNA and PDLA firmware may also need to be upgraded.
- 3 Run the PDLA User Interface setup package.
 - The installer can be launched by double-clicking on the executable.
 - Select Next or Continue for all prompts.
- 4 (Optional) Download the PDLA Remote Client Installer. Install this on computers that will be used to remotely control the PDLA. The Remote Client should always use the same version Client software as the PDLA User Interface version. Do not install the Remote Client on the PDLA instrument itself. Refer to Chapter 5, “Remote Operation.”

Hard Drive Backup and Recovery

The following procedure is used to back up or restore the Windows operating system of Agilent 86038B.

NOTE

MS Windows based instruments must have a keyboard attached prior to starting this procedure.

- 1 Restart the instrument to access the boot option screen. A keyboard is required. If the system begins to restore from Hibernation, press F8 and then choose “Delete restoration data and proceed to system boot menu”. The Agilent PDLA offers two boot options:
 - a Agilent Technologies PNA Series Network Analyzer
 - b Network Analyzer Backup and Recovery Program
- 2 *Quickly* select the second option using the arrow key. (The first option is the default, and will be automatically selected within a few seconds.)
- 3 When the Agilent Technologies System Utilities Screen appears, select from the following options.
 - a 1. Restore the Factory Image

This process restores your system to the original, factory-installed operating system and system software. A recovery should be done if MS Windows is corrupted or if you have been instructed to do so by an Agilent service engineer.
 - b 4. Exit the recovery program.
- 4 Enter the number associated with the desired action. For example, enter 2 if you want to exit the utility with no changes.
- 5 The procedure will begin. Follow the onscreen instructions to complete the process. The procedure will take approximately 10 to 20 minutes to complete.
- 6 When the process is finished, restart the instrument and reinstall the 86038B software.

Hard Drive Replacement and Spare

The Agilent 86038B can be provided with an easily replaceable Hard Drive via the Option #520. This option takes advantage of the modular back-panel hard drive configuration of the PNA. This option is useful for controlled working environments, where storage devices should not leave the premises, by permitting the instrument to have an “inside” and an “outside” disk drive. The option can also be used to provide a backup, to minimize downtime in case of a drive failure.

The hard drive module will be shipped with a pre-installed Windows XP operating system and pre-installed PNA software.

The unit can be installed into the PNA using the instructions enclosed with the drive.

When the unit is then started, the PNA software will be launched. Close this software interface to install the PDLA User Interface software.

The setup package for the PDLA software is included on the CD enclosed with the spare drive, and can also be obtained via the Agilent web page: www.agilent.com/comms/octsolutions

This software package can be transferred to the PDLA using the LAN connection, an external USB CD drive or via another USB storage device.

The setup package should then be started on the PDLA by double-clicking on the setup file and accepting the recommended settings.

After this setup, the instrument will launch with the PDLA user interface at the next boot.

After successfully making this installation, and any other user configurations, it is recommended to create a user backup image according to the procedure in the previous section.

Cleaning Information

The following Cleaning Information contains some general safety precautions, which must be observed during all phases of cleaning. Consult your specific optical device manuals or guides for full information on safety matters.

Please try, whenever possible, to use physically contacting connectors, and dry connections. Clean the connectors, interfaces, and bushings carefully after use.

If you are unsure of the correct cleaning procedure for your optical device, we recommend that you first try cleaning a dummy or test device.

Agilent Technologies assume no liability for the customer's failure to comply with these requirements.

Safety Precautions

Please follow the following safety rules:

- Do not remove instrument covers when operating.
- Ensure that the instrument is switched off throughout the cleaning procedures.
- Use of controls or adjustments or performance of procedures other than those specified may result in hazardous radiation exposure.
- Make sure that you disable all sources when you are cleaning any optical interfaces.
- Under no circumstances look into the end of an optical device attached to optical outputs when the device is operational. The laser radiation is not visible to the human eye, but it can seriously damage your eyesight.
- To prevent electrical shock, disconnect the instrument from the mains before cleaning. Use a dry cloth, or one slightly dampened with water, to clean the external case parts. Do not attempt to clean internally.
- Do not install parts or perform any unauthorized modification to optical devices.
- Refer servicing only to qualified and authorized personnel.

Why is it important to clean optical devices?

CAUTION

Optical power levels above 100 mW applied to single mode connectors can easily damage the connector if it is not perfectly clean. Also, scratched or poorly cleaned connectors can destroy optical connectors mechanically. Always make sure that your optical connectors are properly cleaned and unscratched before connection. However, Agilent Technologies Deutschland GmbH assumes no responsibility in case of an operation that is not compliace with the safety instructions as stated above.

In transmission links optical fiber cores are about 9 mm (0.00035") in diameter. Dust and other particles, however, can range from tenths to hundredths of microns in diameter. Their comparative size means that they can cover a part of the end of a fiber core, and thus degrade the transmission quality. This will reduce the performance of your system.

Furthermore, the power density may burn dust into the fiber and cause additional damage (for example, 0 dBm optical power in a single mode fiber causes a power density of approximately 16 million W/m²). If this happens, measurements become inaccurate and non-repeatable.

Cleaning is, therefore, an essential yet difficult task. Unfortunately, when comparing most published cleaning recommendations, you will discover that they contain several inconsistencies. In this chapter, we want to suggest ways to help you clean your various optical devices, and thus significantly improve the accuracy and repeatability of your lightwave measurements.

What materials do I need for proper cleaning?

Some Standard Cleaning Equipment is necessary for cleaning your instrument. For certain cleaning procedures, you may also require certain Additional Cleaning Equipment.

Standard Cleaning Equipment

Before you can start your cleaning procedure you need the following standard equipment:

- Dust and shutter caps
- Isopropyl alcohol
- Cotton swabs

- Soft tissues
- Pipe cleaner
- Compressed air

Dust and shutter caps All of Agilent Technologies' lightwave instruments are delivered with either laser shutter caps or dust caps on the lightwave adapter. Any cables come with covers to protect the cable ends from damage or contamination.

We suggest these protective coverings should be kept on the equipment at all times, except when your optical device is in use. Be careful when replacing dust caps after use. Do not press the bottom of the cap onto the fiber too hard, as any dust in the cap can scratch or pollute your fiber surface.

If you need further dust caps, please contact your nearest Agilent Technologies sales office.

Isopropyl alcohol This solvent is usually available from any local pharmaceutical supplier or chemist's shop. Results will vary depending on the purity of the alcohol.

If you use isopropyl alcohol to clean your optical device, do not immediately dry the surface with compressed air (except when you are cleaning very sensitive optical devices). This is because the dust and the dirt is dissolved in the alcohol and will leave behind filmy deposits after the alcohol has evaporated. You should therefore first remove the alcohol and the dust with a soft tissue, and then use compressed air to blow away any remaining filaments.

If possible avoid using denatured alcohol containing additives. Instead, apply alcohol used for medical purposes.

Never drink this alcohol, as it may seriously damage to your health.

Do not use any other solvents, as some may damage plastic materials and claddings. Acetone, for example, will dissolve the epoxy used with fiber optic connectors. To avoid damage, only use isopropyl alcohol.

Cotton swabs We recommend that you use swabs such as Q-tips or other cotton swabs normally available from local distributors of medical and hygiene products (for example, a supermarket or a chemist's shop). You may be able to obtain various sizes of swab. If this is the case, select the smallest size for your smallest devices.

Ensure that you use natural cotton swabs. Foam swabs will often leave behind filmy deposits after cleaning.

Use care when cleaning, and avoid pressing too hard onto your optical device with the swab. Too much pressure may scratch the surface, and could cause your device to become misaligned. It is advisable to rub gently over the surface using only a small circular movement.

Swabs should be used straight out of the packet, and never used twice. This is because dust and dirt in the atmosphere, or from a first cleaning, may collect on your swab and scratch the surface of your optical device.

Soft tissues These are available from most stores and distributors of medical and hygiene products such as supermarkets or chemists' shops.

We recommend that you do not use normal cotton tissues, but multi-layered soft tissues made from non-recycled cellulose. Cellulose tissues are very absorbent and softer. Consequently, they will not scratch the surface of your device over time.

Use care when cleaning, and avoid pressing on your optical device with the tissue. Pressing too hard may lead to scratches on the surface or misalignment of your device. Just rub gently over the surface using a small circular movement.

Use only clean, fresh soft tissues and never apply them twice. Any dust and dirt from the air which collects on your tissue, or which has gathered after initial cleaning, may scratch and pollute your optical device.

Pipe cleaner Pipe cleaners can be purchased from tobacconists, and come in various shapes and sizes. The most suitable one to select for cleaning purposes has soft bristles, which will not produce scratches.

The best way to use a pipe cleaner is to push it in and out of the device opening (for example, when cleaning an interface). While you are cleaning, you should slowly rotate the pipe cleaner.

Only use pipe cleaners on connector interfaces or on feed through adapters. Do not use them on optical head adapters, as the center of a pipe cleaner is hard metal and can damage the bottom of the adapter.

Your pipe cleaner should be new when you use it. If it has collected any dust or dirt, this can scratch or contaminate your device.

The tip and center of the pipe cleaner are made of metal. Avoid accidentally pressing these metal parts against the inside of the device, as this can cause scratches.

Compressed air Compressed air can be purchased from any laboratory supplier.

It is essential that your compressed air is free of dust, water and oil. Only use clean, dry air. If not, this can lead to filmy deposits or scratches on the surface of your connector. This will reduce the performance of your transmission system.

When spraying compressed air, hold the can upright. If the can is held at a slant, propellant could escape and dirty your optical device. First spray into the air, as the initial stream of compressed air could contain some condensation or propellant. Such condensation leaves behind a filmy deposit.

Please be friendly to your environment and use a CFC-free aerosol.

Additional Cleaning Equipment

Some Cleaning Procedures need the following equipment, which is not required to clean each instrument:

- Microscope with a magnification range about 50X up to 300X
- Ultrasonic bath
- Warm water and liquid soap
- Premoistened cleaning wipes
- Polymer film
- Infrared Sensor Card

Microscope with a magnification range about 50X up to 300X A microscope can be found in most photography stores, or can be obtained through or specialist mail order companies. Special fiber-scopes are available from suppliers of splicing equipment.

Ideally, the light source on your microscope should be very flexible. This will allow you to examine your device closely and from different angles.

A microscope helps you to estimate the type and degree of dirt on your device. You can use a microscope to choose an appropriate cleaning method, and then to examine the results. You can also use your microscope to judge whether your optical device (such as a connector) is severely scratched and is, therefore, causing inaccurate measurements.

Ultrasonic bath Ultrasonic baths are also available from photography or laboratory suppliers or specialist mail order companies.

An ultrasonic bath will gently remove fat and other stubborn dirt from your optical devices. This helps increase the life span of the optical devices.

Only use isopropyl alcohol in your ultrasonic bath, as other solvents may cause damage.

Warm water and liquid soap Only use water if you are sure that there is no other way of cleaning your optical device without causing corrosion or damage. Do not use hot water, as this may cause mechanical stress, which can damage your optical device.

Ensure that your liquid soap has no abrasive properties or perfume in it. You should also avoid normal washing up liquid, as it can cover your device in an iridescent film after it has been air dried.

Some lenses and mirrors also have a special coating, which may be sensitive to mechanical stress, or to fat and liquids. For this reason we recommend you do not touch them.

If you are not sure how sensitive your device is to cleaning, please contact the manufacturer or your sales distributor.

Premoistened cleaning wipes Use pre-moistened cleaning wipes as described in each individual cleaning procedure. Cleaning wipes may be used in every instance where a moistened soft tissue or cotton swab is applied.

Polymer film Polymer film is available from laboratory suppliers or specialist mail order companies.

Using polymer film is a gentle method of cleaning extremely sensitive devices, such as reference reflectors and mirrors.

Infrared Sensor Card Infrared sensor cards are available from laboratory suppliers or specialist mail order companies.

With the help of this card you are able to inspect the shape of laser light emitted. The invisible laser beam is projected onto the sensor card. The light beam's infrared wavelengths are reflected at visible wavelengths, so becoming visible to the normal eye as a round spot.

Take care never to look into the end of a fiber or any other optical component, when they are in use. This is because the laser can seriously damage your eyes.

Preserving Connectors

Listed below are some hints on how best to keep your connectors in the best possible condition.

Making Connections

Before you make any connection you must ensure that all cables and connectors are clean. If they are dirty, use the appropriate cleaning procedure.

When inserting the ferrule of a patchcord into a connector or an adapter, make sure that the fiber end does not touch the outside of the mating connector or adapter. Otherwise you will rub the fiber end against an unsuitable surface, producing scratches and dirt deposits on the surface of your fiber.

Dust Caps and Shutter Caps

Be careful when replacing dust caps after use. Do not press the bottom of the cap onto the fiber as any dust in the cap can scratch or dirty your fiber surface.

When you have finished cleaning, put the dust cap back on, or close the shutter cap if the equipment is not going to be used immediately.

Always keep the caps on the equipment when it is not in use.

All of Agilent Technologies' lightwave instruments and accessories are shipped with either laser shutter caps or dust caps. If you need additional or replacement dust caps, contact your nearest Agilent Technologies Sales/Service Office.

Immersion Oil and Other Index Matching Compounds

Wherever possible, do not use immersion oil or other index matching compounds with your device. They are liable to impair and dirty the surface of the device. In addition, the characteristics of your device can be changed and your measurement results affected.

Cleaning Instrument Housings

Use a dry and very soft cotton tissue to clean the instrument housing and the keypad. Do not open the instruments as there is a danger of electric shock, or electrostatic discharge. Opening the instrument can cause damage to sensitive components, and in addition your warranty will be invalidated.

General Cleaning Procedure

Light dirt

If you just want to clean away light dirt, observe the following procedure for all devices:

- Use compressed air to blow away large particles.
- Clean the device with a dry cotton swab.
- Use compressed air to blow away any remaining filament left by the swab.

Heavy dirt

If the above procedure is not enough to clean your instrument, follow one of the procedures below.

If you are unsure of how sensitive your device is to cleaning, please contact the manufacturer or your sales distributor

How to clean connectors

Cleaning connectors is difficult as the core diameter of a single mode fiber is only about 9 μ m. This generally means you cannot see streaks or scratches on its surface. To be certain of the condition of the surface of your connector and to check it after cleaning, you need a microscope.

WARNING

In the case of scratches, or of dust that has been burnt onto the surface of the connector, you may have no option but to polish the connector. This depends on the degree of dirtiness, or the depth of the scratches. This is a difficult procedure and should only be performed by a skilled person, and as a last resort as it wears out your connector.

Never look into the end of an optical cable that is connected to an active source.

To assess the projection of the emitted light beam you can use an infrared sensor card. Hold the card approximately 5 cm from the output of the connector. The invisible emitted light is projected onto the card and becomes visible as a small circular spot.

Cleaning procedure for high-power single mode connections

Optical single mode connectors for high-power applications (optical power levels above 100 mW) require careful cleaning to prevent the power density of burning dust or dirt into the fiber causing permanent damage to the devices and/or connectors. If this happens, measurements become inaccurate and unrepeatable.

The “Preferred Procedure” on page 313 or the “Procedure for Stubborn Dirt” on page 313 must be strictly followed for each part of the optical connection (connector, connector interface, and physical connector interface).

Always make sure that the fiber end-faces are properly cleaned and unscratched before connection. The fiber end faces must be visually inspected using a microscope with a magnification of at least 400x. For recommended fiber inspection microscopes, please refer to personnel in Agilent’s Service Team.

The connection should be made immediately after cleaning and inspection to prevent the connection (connector, connector interface, and physical connector interface) from becoming dusty or dirty again.

Preferred Procedure

An Optical Connector Cleaner, which resembles a VCR cleaning tape, is a device that can be used to clean grease from the surface of a connector.

- 1 Blow away any surface dust with compressed air..
- 2 Press the button on the side of the Optical Connector Cleaner device to ensure that a fresh strip of tape is ready.
- 3 Position the connector interface on the tape.
- 4 Holding the connector interface against the tape, rotate the interface about 180 degrees, then slide it across the surface of the tape.

Alternative Procedure

Use the following procedure if an Optical Connector Cleaner is not available.

- 1 Clean the connector by rubbing a new, dry cotton swab over the surface using a small circular movement.
- 2 Blow away any remaining lint with compressed air.

Procedure for Stubborn Dirt

Use this procedure when there is greasy dirt on the connector:

- 1 Moisten a new cotton swab with isopropyl alcohol.
- 2 Clean the connector by rubbing the cotton swab over the surface using a small circular movement.
- 3 Take a new, dry soft tissue and remove the alcohol, dissolved sediment and dust, by rubbing gently over the surface using a small circular movement.
- 4 Blow away any remaining lint with compressed air.

An Alternative Procedure

A better, more gentle, but more expensive cleaning procedure is to use an ultrasonic bath with isopropyl alcohol.

- 1 Hold the tip of the connector in the bath for at least three minutes.
- 2 Take a new, dry soft tissue and remove the alcohol, dissolved sediment and dust, by rubbing gently over the surface using a small circular movement.
- 3 Blow away any remaining lint with compressed air.

How to clean optical head adapters

CAUTION

Some adapters have an anti-reflection coating on the back to reduce back reflection. This coating is extremely sensitive to solvents and mechanical abrasion. Extra care is needed when cleaning these adapters.

When using optical head adapters, periodically inspect the optical head's front window. Dust and metal particles can be propelled through the adapter's pinhole while inserting the connector ferrule into the receptacle. These dirt particles collect on the head's front window, which can lead to incorrect results if not removed.

Preferred Procedure

Use the following procedure on most occasions.

- 1 Clean the adapter by rubbing a new, dry cotton swab over the surface using a small circular movement.
- 2 Blow away any remaining lint with compressed air.

Procedure for Stubborn Dirt

Use this procedure when there is greasy dirt on the adapter:

- 1 Moisten a new cotton swab with isopropyl alcohol.
- 2 Clean the adapter by rubbing the cotton swab over the surface using a small circular movement.
- 3 Take a new, dry soft tissue and remove the alcohol, dissolved sediment and dust, by rubbing gently over the surface using a small circular movement.
- 4 Blow away any remaining lint with compressed air.

How to clean connector interfaces

CAUTION

Be careful when using pipe cleaners, as the core and the bristles of the pipe cleaner are hard and can damage the interface.

Do not use pipe cleaners on optical head adapters, as the hard core of normal pipe cleaners can damage the bottom of an adapter.

Preferred Procedure

Use the following procedure on most occasions.

- 1 Clean the interface, when no lens is connected, by pushing and pulling a new, dry pipe cleaner into the opening. Rotate the pipe cleaner slowly as you do this.
- 2 Blow away any remaining lint with compressed air.

Procedure for Stubborn Dirt

Use this procedure when there is greasy dirt on the interface:

- 1 Moisten a new pipe cleaner with isopropyl alcohol.
- 2 Clean the interface by pushing and pulling the pipe cleaner into the opening. Rotate the pipe cleaner slowly as you do this.
- 3 Using a new, dry pipe cleaner, and a new, dry cotton swab remove the alcohol, any dissolved sediment and dust.
- 4 Blow away any remaining lint with compressed air.

How to clean bare fiber adapters

Bare fiber adapters are difficult to clean. Protect from dust unless they are in use.

CAUTION

Never use any kind of solvent when cleaning a bare fiber adapter as solvents can:

- Damage the foam inside some adapters.
- Deposit dissolved dirt in the groove, which can then dirty the surface of an inserted fiber.

Preferred Procedure

Use the following procedure on most occasions.

- 1 Blow away any dust or dirt with compressed air.

Procedure for Stubborn Dirt

Use this procedure when there is greasy dirt on the adapter:

- 1 Clean the adapter by pushing and pulling a new, dry pipe cleaner into the opening. Rotate the pipe cleaner slowly as you do this.

CAUTION

Be careful when using pipe cleaners, as the core and the bristles of the pipe cleaner are hard and can damage the adapter.

- 2 Clean the adapter by rubbing a new, dry cotton swab over the surface using a small circular movement.
- 3 Blow away any remaining lint with compressed air.

How to clean lenses and instruments with an optical glass plate

Some lenses have special coatings that are sensitive to solvents, grease, liquid and mechanical abrasion. Take extra care when cleaning lenses with these coatings. Some instruments, for example, Agilent's optical heads have an optical glass plate to protect the sensor.

CAUTION

Do not attempt to access the internal parts of an Agilent N3988A video microscope for cleaning or for any other purpose.

Lens assemblies consisting of several lenses are not normally sealed. Therefore, use as little alcohol as possible, as it can get between the lenses and in doing so can change the properties of projection.

If you are cleaning an Agilent 8162*A optical head, periodically inspect the optical head's front window for dust and other particles. Dust and particles can be propelled through the optical head adapter's pinhole while inserting a connector ferrule into the receptacle. Particles on the optical head's front window can significantly impair measurement results.

NOTE Do not dry the lens by rubbing with with cloth or other material, which may scratch the lens surface.

Preferred Procedure

Use the following procedure on most occasions.

- 1 Clean the lens by rubbing a new, dry cotton swab over the surface using a small circular movement.
- 2 Blow away any remaining lint with compressed air.

Procedure for Stubborn Dirt

Use this procedure when there is greasy dirt on the lens:

- 1 Moisten a new cotton swab with isopropyl alcohol.
- 2 Clean the lens by rubbing the cotton swab over the surface using a small circular movement.
- 3 Using a new, dry cotton swab remove the alcohol, any dissolved sediment and dust.

- 4 Blow away any remaining lint with compressed air.

How to clean instruments with a fixed connector interface

You should only clean instruments with a fixed connector interface when it is absolutely necessary. This is because it is difficult to remove any used alcohol or filaments from the input of the optical block.

It is important, therefore, to keep dust caps on the equipment at all times, except when your optical device is in use.

If you do discover filaments or particles, the only way to clean a fixed connector interface and the input of the optical block is to use compressed air.

If there are fluids or fat in the connector, please refer the instrument to the skilled personnel of Agilent's service team.

NOTE Both the surface and the jacket of the attached connector should be completely dry and clean.

CAUTION

Only use clean, dry compressed air. Make sure that the air is free of dust, water, and oil. If the air that you use is not clean and dry, this can lead to filmy deposits or scratches on the surface of your connector interface. This will degrade the performance of your transmission system.

Never try to open the instrument and clean the optical block by yourself, because it is easy to scratch optical components, and cause them to become misaligned.

How to clean instruments with a physical contact interface

Remove any connector interfaces from the optical output of the instrument before you begin the cleaning procedure.

Cleaning interfaces is difficult as the core diameter of a single mode fiber is only about 9 μm. This generally means you cannot see streaks or scratches on the surface. To be certain of the degree of pollution on the surface of your interface and to check whether it has been removed after cleaning, you need a microscope.

WARNING

Never look into an optical output, because this can seriously damage your eyesight.

To assess the projection of the emitted light beam you can use an infrared sensor card. Hold the card approximately 5 cm from the interface. The invisible emitted light is projected onto the card and becomes visible as a small circular spot.

Optical single mode connections for high-power applications (optical power levels above 100mW) require careful cleaning to prevent the power density of burning dust or dirt into the fiber causing permanent damage of the devices and/or connectors. If this happens, measurements become inaccurate and non-repeatable.

Preferred Procedure

Use the following procedure on most occasions.

- 1 Clean the interface by rubbing a new, dry cotton swab over the surface using a small circular movement.
- 2 Blow away any remaining lint with compressed air.

Procedure for Stubborn Dirt

Use this procedure when there is greasy dirt on the interface:

- 1 Moisten a new cotton swab with isopropyl alcohol.
- 2 Clean the interface by rubbing the cotton swab over the surface using a small circular movement.
- 3 Take a new, dry soft tissue and remove the alcohol, dissolved sediment and dust, by rubbing gently over the surface using a small circular movement.
- 4 Blow away any remaining lint with compressed air.

NOTE This procedures must be strictly followed for each part of the optical connection (connector, connector interface and physical connector interface).

NOTE The connection should be made immediately after cleaning and inspection to prevent the connection (connector, connector interface and physical connector interface) from getting dusty or dirty again.

NOTE Always make sure that the fiber end faces are properly cleaned and unscratched before connection. The fiber end faces must be visually inspected using a microscope with a magnification of at least 400x. For recommended fiber inspection microscopes, please refer to Agilent 's service team personnel.

How to clean instruments with a recessed lens interface

For instruments with a *deeply* recessed lens interface (for example the Agilent 81633A and 81634A Power Sensors) do NOT follow this procedure. Alcohol and compressed air could damage your lens even further.

Keep your dust and shutter caps on when your instrument is not in use. This should prevent it from getting too dirty.

If you must clean such instruments, please refer the instrument to the skilled personnel of Agilent's service team.

Preferred Procedure

Use the following procedure on most occasions.

- 1 Blow away any dust or dirt with compressed air.
If this is not sufficient, then
 - a Clean the interface by rubbing a new, dry cotton swab over the surface using a small circular movement.
 - b Blow away any remaining lint with compressed air.

Procedure for Stubborn Dirt

Use this procedure when there is greasy dirt on the interface, and using the preferred procedure is not sufficient.

Using isopropyl alcohol should be your last choice for recessed lens interfaces because of the difficulty of cleaning out any dirt that is washed to the edge of the interface.

- 1 Moisten a new cotton swab with isopropyl alcohol.
- 2 Clean the interface by rubbing the cotton swab over the surface using a small circular movement.
- 3 Take a new, dry soft tissue and remove the alcohol, dissolved sediment and dust, by rubbing gently over the surface using a small circular movement.
- 4 Blow away any remaining lint with compressed air.

How to clean optical devices which are sensitive to mechanical stress and pressure

Some optical devices, such as the Agilent 81000BR Reference Reflector, which has a gold plated surface, are very sensitive to mechanical stress or pressure. Do not use cotton swabs, soft tissues or other mechanical cleaning tools, as these can scratch or destroy the surface.

Preferred Procedure

Use the following procedure on most occasions.

- 1 Blow away any dust or dirt with compressed air.

Procedure for Stubborn Dirt

To clean devices that are extremely sensitive to mechanical stress or pressure you can also use an optical clean polymer film. This procedure is time-consuming, but you avoid scratching or destroying the surface.

- 1 Put the film on the surface and wait at least 30 minutes to make sure that the film has had enough time to dry.
- 2 Remove the film and any dirt with special adhesive tapes.

Alternative Procedure


For these types of optical devices you can often use an ultrasonic bath with isopropyl alcohol. Only use the ultrasonic bath if you are sure that it won't cause any damage any part of the device.

- 1 Put the device into the bath for at least three minutes.
- 2 Blow away any remaining liquid with compressed air.

If there are any streaks or drying stains on the surface, repeat the cleaning procedure.

How to clean metal filters or attenuator gratings

This kind of device is extremely fragile. A misalignment of the grating leads to inaccurate measurements. Never touch the surface of the metal filter or attenuator grating.



Be very careful when using or cleaning these devices. Do not use cotton swabs or soft tissues, as there is the danger that you cannot remove the lint and that the device will be destroyed by becoming mechanically distorted.

Preferred Procedure

Use the following procedure on most occasions.

- 1 Use compressed air at a distance and with low pressure to remove any dust or lint.

Procedure for Stubborn Dirt

Do not use an ultrasonic bath as this can damage your device.

Use this procedure when there is greasy dirt on the device:

- 1 Put the optical device into a bath of isopropyl alcohol, and wait at least 10 minutes.
- 2 Remove the fluid using compressed air at some distance and with low pressure. If there are any streaks or drying stains on the surface, repeat the whole cleaning procedure.

Additional Cleaning Information

The following cleaning procedures may be used with other optical equipment:

- How to clean bare fiber ends
- How to clean large area lenses and mirrors

How to clean bare fiber ends

Bare fiber ends are often used for splices or, together with other optical components, to create a parallel beam.

The end of a fiber can often be scratched. You make a new cleave. To do this:

- 1 Strip off the cladding.
- 2 Take a new soft tissue and moisten it with isopropyl alcohol.
- 3 Carefully clean the bare fiber with this tissue.
- 4 Make your cleave and immediately insert the fiber into your bare fiber adapter in order to protect the surface from dirt.

Preferred Procedure

There is an easy method for removing dust from bare fiber ends.

- 1 Touch the bare fiber end with adhesive tape. Any dust will be removed.

How to clean large area lenses and mirrors

Some mirrors, as those from a monochromator, are very soft and sensitive. Therefore, never touch them and do not use cleaning tools such as compressed air or polymer film.

Some lenses have special coatings that are sensitive to solvents, grease, liquid and mechanical abrasion. Take extra care when cleaning lenses with these coatings.

Lens assemblies consisting of several lenses are not normally sealed. Therefore, use as little liquid as possible, as it can get between the lenses and in doing so can change the properties of projection.

Preferred Procedure

Use the following procedure on most occasions.

- 1 Blow away any dust or dirt with compressed air.

Procedure for Stubborn Dirt

Use this procedure when there is greasy dirt on the lens:

CAUTION

Only use water if you are sure that there is no other way of cleaning your optical device without causing corrosion or damage. Do not use hot water, as this may cause mechanical stress, which can damage your optical device.

Ensure that your liquid soap has no abrasive properties or perfume in it. You should also avoid normal washing up liquid, as it can cover your device in an iridescent film after it has been air dried.

Some lenses and mirrors also have a special coating, which may be sensitive to mechanical stress, or to fat and liquids. For this reason we recommend you do not touch them.

If you are not sure how sensitive your device is to cleaning, please contact the manufacturer or your sales distributor.

- 1 Moisten the lens or the mirror with water.
- 2 Put a little liquid soap on the surface and gently spread the liquid over the whole area.
- 3 Wash off the emulsion with water, being careful to remove it all, as any remaining streaks can impair measurement accuracy.
- 4 Take a new, dry soft tissue and remove the water, by rubbing gently over the surface using a small circular movement.
- 5 Blow away remaining lint with compressed air.

Alternative Procedure A

To clean lenses that are extremely sensitive to mechanical stress or pressure you can also use an optical clean polymer film. This procedure is time-consuming, but you avoid scratching or destroying the surface.

- 1 Put the film on the surface and wait at least 30 minutes to make sure that the film has had enough time to dry.
- 2 Remove the film and any dirt with special adhesive tapes.

Alternative Procedure B

If your lens is sensitive to water then:

- 1 Moisten the lens or the mirror with isopropyl alcohol.
- 2 Take a new, dry soft tissue and remove the alcohol, dissolved sediment and dust, by rubbing gently over the surface using a small circular movement.
- 3 Blow away remaining lint with compressed air.

Other Cleaning Hints

Selecting the correct cleaning method is an important element in maintaining your equipment and saving you time and money. This chapter highlights the main cleaning methods, but cannot address every individual circumstance.

This section contain some additional hints which we hope will help you further. For further information, please contact your local Agilent Technologies representative.

Making the connection

Before you make any connection you must ensure that all lightwave cables and connectors are clean. If not, then use the appropriate cleaning methods.

When you insert the ferrule of a patchcord into a connector or an adapter, ensure that the fiber end does not touch the outside of the mating connector or adapter. Otherwise, the fiber end will rub up against something which could scratch it and leave deposits.

Lens cleaning papers

Some special lens cleaning papers are not suitable for cleaning optical devices like connectors, interfaces, lenses, mirrors and so on. To be absolutely certain that a cleaning paper is applicable, please ask the salesperson or the manufacturer.

Immersion oil and other index matching compounds

Do not use immersion oil or other index matching compounds with optical sensors equipped with recessed lenses. They are liable to dirty the detector and impair its performance. They may also alter the property of depiction of your optical device, thus rendering your measurements inaccurate.

Cleaning the housing and the mainframe

When cleaning either the mainframe or the housing of your instrument, only use a dry and very soft cotton tissue on the surfaces and the numeric pad. Never open the instruments as they can be damaged. Opening the instruments puts you in danger of receiving an electrical shock from your device, and renders your warranty void

Returning the Instrument for Service

Agilent Technologies aims to maximize the value you receive, while minimizing your risk and problems. We strive to ensure that you get the test and measurement capabilities you paid for and obtain the support you need. Our extensive support resources and services can help you choose the right Agilent products for your applications and apply them successfully. Every instrument and system we sell has a global warranty. Support is normally available for at least five years beyond the production life of the product. Two concepts underlie Agilent's overall support policy: "Our Promise" and "Your Advantage".

Our Promise means your Agilent test and measurement equipment will meet its advertised performance and functionality. When you are choosing new equipment, we will help you with product information, including realistic performance specifications and practical recommendations from experienced test engineers. When you use Agilent equipment, we can verify that it works properly, help with product operation, and provide basic measurement assistance for the use of specified capabilities, at no extra cost upon request. Often, many self-help tools are available.

Your Advantage means that Agilent offers a wide range of additional expert test and measurement services, which you can purchase according to your unique technical business needs. Solve problems efficiently and gain a competitive edge by contracting with us for calibration, extra-cost upgrades, out-of-warranty repairs, and on-site education and training, as well as design, system integration, project management, and other professional engineering services. Experienced Agilent engineers and technicians worldwide can help you maximize your productivity, optimize the return on investment of your Agilent instruments and systems, and obtain dependable measurement accuracy for the life of those products.



8

Specifications and Regulatory Information

Introduction.....	328
Angled Connector Specifications	338
Specifications	339
Regulatory Information	343
Declaration of Conformity.....	344

Introduction

This chapter contains specifications and characteristics for the Agilent 86038B optical dispersion analyzer. For specifications specific to the 81600B tunable laser sources refer to the specifications chapter in the Tunable Laser Modules User's Guide.

CAUTION

Agilent Technologies does not recommend installing any additional software on the system controller. Some third party software, including printer drivers, may impair system operation.

Definition of Terms

Specification: describes a guaranteed product performance that is valid under the specified conditions. Specifications are based on a coverage factor 1 of 2 (unless otherwise stated), corresponding to a level of confidence of >95%.

Typical value: a characteristic describing the product performance that is usually met, but not guaranteed.

Generally, all specifications are valid after warm-up, at the stated operating conditions and measurement settings, in swept-wavelength measurement mode.

Absolute wavelength uncertainty

Specifies the maximum difference between displayed and actual wavelength. Displayed wavelength refers to the wavelength associated with a point in the measurement result.

Wavelength is defined as wavelength in vacuum.

Conditions: “Stepped mode” or “Swept mode (continuous sweep mode)” as specified. Other conditions as specified.

Chromatic dispersion (CD)

The derivative over wavelength of the “Group delay (GD)” or, equivalently, of the “Relative group delay (rGD)”, usually expressed in ps/nm.

Chromatic dispersion (CD) uncertainty

Specifies the uncertainty of the measurement results for “Chromatic dispersion (CD)”, expressed as the maximum difference between the measured value and actual value.

Conditions: as specified.

Measurement: The specification is derived from the “Relative Group delay (rGD) uncertainty” and the “Relative wavelength uncertainty”.

Depolarization rate

The depolarization rate, D , quantifies the absolute value of the differential change of the normalized Polarization mode dispersion (PMD) vector Ω_n , with rotational frequency change ω , expressed in [s].

$$D = |\Delta\Omega_n / \Delta\omega|$$

Note: The depolarization rate is a measure of how fast the PMD vector changes direction with the rotational frequency of the optical wave (or with the wavelength).

Differential group delay (DGD) repeatability

The agreement of repeated measurements of “Differential group delay (DGD)” on a stable test device and under constant environmental conditions, expressed as:

$$\pm 2 \cdot \text{StDev}_i\{\text{DGD}_i(\lambda)\}.$$

$\text{DGD}(\lambda)$ denotes the DGD at any wavelength and StDev denotes the standard deviation over repetitions i .

Conditions: Stable setup (without cable movement or re-connections).

Measurement: see “Differential group delay (DGD)” uncertainty.

Differential group delay (DGD)

The dependence of the “Group delay (GD)” at a fixed wavelength resulting from applying all possible polarization states, expressed as the difference between the maximum and the minimum group delay value.

Differential group delay (DGD) uncertainty

Specifies the maximum difference between measured and actual “Differential group delay (DGD)” .

Conditions: Measurement settings and operation mode as specified.

Measurement: Using a standard SM fiber with nominally 0 ps DGD and various PM fibers with specified DGD over the specified wavelength range.

Gain (loss)

Gain is defined as the difference in the transmitted power, expressed in dB, between the test device measurement, P_b , and the normalization measurement, P_a , given a constant power of the laser source, calculated as:

$$\text{Gain[dB]} = 10\log (P_b/P_a) = P_b[\text{dBm}] - P_a[\text{dBm}]$$

Loss is calculated with the same formula as gain and corresponds to negative values.

Note 1: The PDLA shows gain (rather than loss). That is, lossy test devices are shown with negative values.

Note 2: Gain (loss) includes connector effects at normalization and at test device measurement.

Note 3: In general, loss depends on wavelength and polarization state.

Gain (loss) uncertainty

Specifies the maximum difference between measured and actual “Gain (loss)”, expressed in dB.

Conditions: operation mode as specified. Loss range as specified..

Group delay (GD)

The optical signal delay time caused by a device or fiber. The signal delay time in the normalization measurement is defined as zero.

Note: In general, group delay depends on wavelength and polarization state. In some cases (for example to determine the chromatic dispersion) only the variation of the group delay over wavelength is of interest, not its absolute value. In these cases the “Relative group delay (rGD)” can be used where an arbitrary delay value is added to the group delay values, typically to bring its minimum value to zero.

Intermediate frequency bandwidth (IFBW)

The setting for the bandwidth of the filter employed in the network analyzer portion of the PDLA.

Note: the IFBW setting influences measurement noise and measurement speed.

Length uncertainty

Specifies the maximum difference between the measured length of a fiber, based on a given value of the effective group refractive index n at the wavelength of measurement, and its actual length. The length uncertainty specification does not include any uncertainty in the entered value of n .

Conditions: Maximum fiber length as specified, refractive index range as specified.

Measurement: Derived from the “Relative Group delay (rGD) uncertainty” and “Modulation frequency” uncertainty specification.

Note: Changes in n , e.g. due to temperature fluctuations, lead to additional uncertainties in the length measurement. The temperature dependence of n is typically on the order of $10^{-5} / ^\circ\text{C}$.

Maximum Safe Optical Input Power

The maximum optical power that can safely be applied to the 86038B Receiver Input port without risking damage to the receiver.

Attention! Applying more than the specified maximum safe power may damage the receiver unit!

Measurement time

The time from starting a test device measurement until the result is displayed on the screen.

Conditions: Measurement mode and settings as specified. Excludes normalization measurement time.

Modulation frequency

A measurement parameter of the 86038B system specifying the frequency of the network analyzer used for amplitude modulation of the optical measurement signal.

Modulation frequency range

The range of modulation frequencies available for measurement of device attributes (loss, group delay, DGD, PDL).

Operating conditions

The environmental conditions for system operation. The system must not be operated outside these conditions.

Note: For a system mounted in a rack the environmental conditions within the rack apply.

Polarization-dependent chromatic dispersion (PCD)

The PCD, P, quantifies the differential change of the magnitude of the Polarization mode dispersion (PMD) vector, with wavelength change expressed in [ps/nm],

$$P = \Delta/\Omega/\Delta\lambda$$

Polarization dependent loss (PDL)

The dependence of the “Gain (loss)” of a device on the input polarization state, calculated as the difference between maximum and minimum loss (gain), usually expressed in dB.

Polarization dependent loss (PDL) uncertainty

Specifies the uncertainty of the measurement results for “Polarization dependent loss (PDL)”, expressed as the maximum difference between the measured value and actual value.

Conditions: operation mode as specified. Loss range as specified.

Measurement: Verified at the central wavelength of the 86038B system wavelength range.

Polarization mode dispersion (PMD)

The arithmetic average of the “Differential group delay (DGD)” over a broad specified wavelength span.

Note: In highly mode-coupled devices, such as standard single mode fibers of kilometer length, the DGD depends on wavelength. Stress or temperature change may change the dependence of DGD on wavelength. It can be shown that the DGD characteristics of such devices can be described statistically by PMD.

Polarization mode dispersion (PMD) uncertainty

Specifies the maximum difference between measured and actual “Polarization mode dispersion (PMD)” .

Conditions: Measurement settings and operation mode as specified. Wavelength span and wavelength increment as specified.

Measurement: See “Differential group delay (DGD) uncertainty” .

Polarization mode dispersion (PMD) vector

The PMD vector represents the polarization dependence of the “Group delay (GD)” . For a given wavelength, the magnitude of the PMD vector is the “Differential group delay (DGD)” at that wavelength, and the vector direction corresponds to the fast principal state of polarization in Mueller notation.

Relative group delay (rGD)

The difference in the “Group delay (GD)” at two different wavelengths. An rGD curve over wavelength shows the group delay difference with respect to the GD at a fixed reference wavelength (e.g. the start wavelength).

Relative group delay (rGD) repeatability

The agreement of repeated measurements of “relative group delay (rGD)”, on a stable test device and under constant environmental conditions, expressed as:

$$\pm 2 \cdot \text{StDev}_i\{ \text{rGD}_i() \}.$$

rGD() denotes the rGD curve over wavelength , and StDevi denotes the standard deviation over repetitions i.

Conditions: Stable setup (without cable movement or re-connections).

Relative Group delay (rGD) uncertainty

Specifies the maximum difference between measured and actual “Relative group delay (rGD)” . The relative group delay (rGD) uncertainty is half the span between maximum and minimum of all differences over wavelength.

Conditions: Measurement settings and operation mode as specified.

Measurement: The measurement uncertainty of GD and relative GD results from

1. frequency uncertainty of the PDLA output modulator (as specified for the network analyzer)

2. phase shift measurement error of the modulated signal at the PDLA receiver between test device measurement and normalization measurement

Part 2 (phase shift measurement error) is determined by measuring the rGD of a stable 2 m SMF patchcord with different modulation frequencies from 1.90 GHz to 2.00 GHz with 0.02 GHz steps. The relative GD uncertainty is \pm twice the standard deviation of the measured rGD values at each wavelength. Reference wavelength is the middle of the measured wavelength range.

Relative wavelength uncertainty

When calculating the difference between measured and actual wavelength over a specified wavelength range, the relative wavelength uncertainty is half the span between maximum and minimum of all differences. Wavelength is defined as wavelength in vacuum.

Conditions: “Swept mode (continuous sweep mode)” or “Stepped mode” as specified. Other conditions as specified.

Second order PMD

The second-order PMD, S, quantifies the magnitude of the differential change of the “Polarization mode dispersion (PMD)” vector, $f\Omega$, with rotational frequency change, $\Delta\omega$, expressed in [ps²],

$$S = |\Delta\Omega / \Delta\omega|$$

Note: Second-order PMD is expressed in [ps/nm] at 1550 nm by multiplying the [ps²] value with the factor 0.784 nm/ps.

Specified temperature

The temperature range where the specifications apply (if not differently stated).

Note: For a system built into a rack, the temperature within the rack applies.

Stepped mode

In stepped mode the tunable laser source is operated statically, so that a measurement is made while the tunable laser source is set to a fixed wavelength.

Note: A tunable laser source can alternatively be operated in “Swept mode (continuous sweep mode)” .

Swept mode (continuous sweep mode)

In continuous sweep mode the tunable laser source is operated dynamically, so that a measurement is made while the wavelength of the tunable laser source changes in a defined way (given by start wavelength, end wavelength and sweep speed).

Note: A tunable laser source can alternatively be operated in “Stepped mode” .

System dynamic range

The maximum test device loss for which amplitude measurements can be performed.

Wavelength increment

The nominal wavelength increment in a measurement result.

Wavelength range

The range of wavelength that is available for the measurement. Wavelength is defined as wavelength in vacuum.

Note: The wavelength range for which the specifications of the 86038B apply may be different.

Wavelength resolution

The product of “Wavelength increment” and the selected smoothing width (if smoothing is used).

Zero dispersion slope

The slope of the chromatic dispersion curve over wavelength at the “zero-dispersion wavelength, usually expressed in ps/nm^2 .

Zero dispersion slope uncertainty

Specifies the uncertainty of the measurement results for “Zero dispersion slope”, expressed as the maximum difference between the measured value and actual value.

Conditions: valid for measurements on fibers. Zero dispersion slope derived from a suitable fit (quadratic or Sellmeier) to the chromatic dispersion data measured around the zero dispersion wavelength. Measured wavelength range for fitting as specified. Wavelength increment as specified. Other conditions: same as for “Relative group delay (rGD)” and “Relative wavelength uncertainty” .

Measurement: The specification is derived from the “Chromatic dispersion (CD) uncertainty” and the (dynamic) “Relative wavelength uncertainty” .

Zero-dispersion wavelength

Wavelength where the “Chromatic dispersion (CD)” of the fiber under test equals zero.

Zero-dispersion wavelength uncertainty

Specifies the uncertainty of the measurement results for “zero-dispersion wavelength”, expressed as the maximum difference between the measured value and actual value.

Conditions: valid for measurements on fibers. Zero dispersion wavelength derived from a suitable fit (quadratic or Sellmeier) to the chromatic dispersion data measured around the zero dispersion wavelength. Measured wavelength range for fitting as specified. Wavelength increment as specified. Other conditions: same as for “Relative Group delay (rGD) uncertainty” and “Relative wavelength uncertainty” .

Measurement: The specification is derived from the “Chromatic dispersion (CD) uncertainty” and the (dynamic) absolute wavelength uncertainty of the used tunable laser source.

Angled Connector Specifications

Specifications require an angled connector at the source output and at the optical receiver.

Angled contact connectors help you to control return loss. With angled fiber endfaces, reflected light tends to reflect into the cladding, reducing the amount of light that reflects back to the source.

The contact connector on your TLS is angled, so only use a cable with angled connectors to connect to the TLS input on the PDLA. The PDLA TLS input requires angled connectors. Do not use a cable with a flat connector on either the angled PDLA TLS input connector or on the angled TLS output connector.

The angled connector symbol is typically colored green. You should connect straight contact fiber end connectors with neutral sleeves to straight contact connectors and connect angled contact fiber end connectors with green sleeves to angled contact connectors.

Specifications

Specifications^[a] apply to swept mode (if not stated otherwise).

Group Delay and Differential Group Delay ^[b]			
loss ≤ 4 dB	loss ≤ 10 dB	loss ≤ 20 dB, typ.	loss ≤ 30 dB, typ.
Relative Group Delay (rGD) Uncertainty ^{[c],[d]}			
± 50 fs	± 100 fs	± 350 fs	± 4 ps
Relative Group Delay (rGD) Repeatability ^{[c],[d]}			
± 30 fs	± 50 fs	± 300 fs	± 3.5 ps
Differential Group Delay (DGD) Uncertainty ^{[e],[f]}			
± 100 fs ± 2.5% of DGD	± 150 fs ± 3% of DGD	± 500 fs ± 2% of DGD	± 5 ps ± 12% of DGD
Differential Group Delay (DGD) Repeatability ^{[e],[f]}			
± 50 fs ± 1% of DGD	± 100 fs ± 2% of DGD	± 350 fs ± 1.5% of DGD	± 3.5 ps ± 12% of DGD
PMD Uncertainty (typical) ^{[e],[f],[h]}			
± 70 fs ± 2% of PMD	± 100 fs ± 2% of PMD	± 300 fs ± 2% of PMD	± 3 ps ± 4% of PMD
Modulation Frequency Range		5 MHz to 2.5 GHz	

[a] Ambient temperature change max. ± 0.5°C since normalization. TLS maximum power set to 0 dBm. Sweep over specified wavelength range. Specification does not include instability in test device. Specified loss ranges include loss of test device and any additional switches or connections in the optical path. Specification valid on day of calibration.

[b] Modulation frequency $f_{\text{mod}} = 2$ GHz. IFBW = 70 Hz, wavelength increment = 1 nm, PDA correction ON.
Note: for modulation frequencies, f_{mod} , different from 2 GHz, uncertainties typically scale by the factor “2 GHz/ f_{mod} ”.

[c] GD/CD Swept Mode.

[d] Based on phase measurement uncertainties and modulation frequency uncertainty.
For rGD >> 1000 ps add ± 10⁻⁶ of rGD (contribution from modulation frequency uncertainty).

[e] DGD/PDL Swept Mode using 6 polarization states.

[f] Tested with a set of fibers (SMF and PMF) covering a range of DGD values.

[h] Averaged over 100 nm wavelength span with wavelength increment ≤ 1 nm.

Length

Length Uncertainty	± (0.02 mm + 1x10 ⁻⁶ of length) typ. ^[i]
--------------------	--

[i] Not including uncertainty of entered value for group refractive index. Specification valid for length ≤ 50 km, based on a refractive index of 1.4.

Amplitude^[i]

Polarization Dependent Loss (PDL) Uncertainty ^[k]	Without Opt. 400: $\pm (0.15 \text{ dB} + 3\% \text{ of PDL})$ typ. With Opt. 400: $\pm (0.05 \text{ dB} + 3\% \text{ of PDL})$
Gain/Loss Uncertainty	Without Opt. 400: $\pm (0.1 \text{ dB} + 5\% \text{ of loss})$ ^[l] typ. With Opt. 400: $\pm 0.02 \text{ dB}$ (loss < 10dB) $\pm 0.04 \text{ dB}$ (loss < 30dB)

[j] DGD/PDL Swept Mode using 6 polarization states, excluding connectors.
[k] Loss from 0 dB to 4 dB; Opt.400 specified over the specified range of the tunable laser option, but limited to within 1450-1625 nm.
[l] Loss from 0 dB to 40 dB.

Dynamic range

System dynamic range	> 40 dB typ. ^[m]
----------------------	-----------------------------

[m] For TLS power level 0 dBm. For higher power levels, the dynamic range increases correspondingly.

Wavelength

Wavelength Range ^[n]	
Opt. #120 or #320: With Agilent 81600B-200 Tunable Laser Source	1440 nm to 1640 nm
Opt. #116 or #316: With Agilent 81600B-160 or 81640B	1495 nm to 1640 nm
Opt. #115 or #315: With Agilent 81600B-150 or 81680B	1450 nm to 1590 nm
Opt. #114 or #314: With Agilent 81600B-140	1370 nm to 1495 nm
Opt. #113 or #313: With Agilent 81600B-130	1260 nm to 1375 nm
Minimum Wavelength Increment	0.1 pm

[n] PDLA specifications apply for sweeps within the following wavelength ranges. Excluding mode-hops. For the conditions for mode-hop free continuous sweeps, please see the tunable laser specifications in the tunable laser User's Guide.
Opt. #120 or #320: 1475 nm to 1625 nm,
Opt. #116 or #316: 1510 nm to 1620 nm,
Opt. #115 or #315: 1480 nm to 1580 nm,
Opt. #114 or #314: 1390 nm to 1490 (typical specifications). Excluding water absorption wavelengths.
Opt. #113 or #313: 1270 nm to 1375 (typical specifications). Excluding water absorption wavelengths.

Absolute Wavelength Uncertainty ^[o]		
Stepped mode	with Agilent 86122A	$\pm 2 \text{ pm typ.}$
	without Agilent 86122A	$\pm 10 \text{ pm, } \pm 3.6 \text{ pm typ.}$
Swept mode	with Agilent 86122A	$\pm 4 \text{ pm typ.}$
	without Agilent 86122A	$\leq \pm 6.1 \text{ pm typ.}$

Relative Wavelength Uncertainty^[o]	
Stepped mode	± 5 pm, ± 2 pm typ.
Swept mode ^[r]	≤ ± 4 pm typ.

[o] Valid for one month and within a ± 4.4 K temperature range after automatic wavelength zeroing. For details, refer to tunable laser's wavelength uncertainty specification.

Chromatic Dispersion^[t]

Specifications for selected wavelength resolution^[u] Δλ

loss ≤ 4 dB	loss ≤ 10 dB	loss ≤ 20 dB, typ.	loss ≤ 30 dB, typ.
Chromatic Dispersion (CD) Uncertainty			
$\pm \frac{70 \text{ fs}}{\Delta\lambda} \pm \frac{0.003 \text{ nm}}{\Delta\lambda} \times CD$	$\pm \frac{140 \text{ fs}}{\Delta\lambda} \pm \frac{0.003 \text{ nm}}{\Delta\lambda} \times CD$	$\pm \frac{0.5 \text{ ps}}{\Delta\lambda} \pm \frac{0.003 \text{ nm}}{\Delta\lambda} \times CD$	$\pm \frac{6 \text{ ps}}{\Delta\lambda} \pm \frac{0.003 \text{ nm}}{\Delta\lambda} \times CD$
Example: ± (0.07 + 0.3% CD) ps/nm for 1 nm wavelength resolution, with a fiber of CD dispersion and loss ≤ 4 dB.			
Zero Dispersion Wavelength Uncertainty^{[w],[x]}			
$\pm 0.008 \text{ nm}^{[v]} \pm \frac{0.007 \text{ ps/nm}}{ZDS}$	$\pm 0.008 \text{ nm}^{[v]} \pm \frac{0.014 \text{ ps/nm}}{ZDS}$	$\pm 0.008 \text{ nm}^{[v]} \pm \frac{0.05 \text{ ps/nm}}{ZDS}$	$\pm 0.008 \text{ nm}^{[v]} \pm \frac{0.6 \text{ ps/nm}}{ZDS}$
Example: ± 0.015 nm typ. for a fiber with zero dispersion slope ZDS = 1 ps/nm ² and loss ≤ 4 dB.			
Zero Dispersion Slope (ZDS) Uncertainty^{[w],[x]}			
$\pm 0.002 \text{ ps/nm}^2 \pm 0.1\% \times ZDS$	$\pm 0.004 \text{ ps/nm}^2 \pm 0.1\% \times ZDS$	$\pm 0.016 \text{ ps/nm}^2 \pm 0.1\% \times ZDS$	$\pm 0.16 \text{ ps/nm}^2 \pm 0.1\% \times ZDS$
Example: ± 0.003 ps/nm ² for a fiber with zero dispersion slope ZDS = 1 ps/nm ² and loss ≤ 4 dB.			
Repeatability of Chromatic Dispersion, Zero Dispersion Wavelength and Zero Dispersion Slope			
typ. < 60% of the respective uncertainty	typ. < 60% of the respective uncertainty	-	-

[t] Uncertainties derived from rGD uncertainty and from ± 2 pm relative wavelength uncertainty for Δλ between 0.1 nm and 10 nm. Parameters and conditions as for rGD.

[u] Wavelength resolution Δλ equals (wavelength increment * selected smoothing width) (see manual). Valid for Δλ between 0.1 nm and 10 nm.

[v] Typical without wavelength meter option. (± 0.006 nm with wavelength meter option.)

[w] For fibers. Determined by a curve fit over a wavelength range of 10 nm including the zero-dispersion wavelength, with wavelength increment ≤ 1 nm, assuming that the selected curve fit sufficiently represents the fiber dispersion curve (the fit does not contribute to the error).

[x] Method confirmed on NIST Standard Reference Material 2524.

General Characteristics

(Minimum configuration including one tunable laser)

Assembled dimensions: (H x W x D) 55.5 cm x 43.5 cm x 55.5 cm
(21.9 in x 17.2 in x 21.9 in)

Power Requirements 100 – 240 V~ 50 – 60 Hz

3 power cables

N5230A: max. 350 W

8164B (incl modules): max. 270 VA

83427C: max. 250 VA

Net weight Standard system: 54 kg (120 lbs)

Measurement time 30 s

(DGD/PDL swept mode with 4 polarization states, IFBW 70 Hz, 2 nm wavelength range and 10 pm wavelength increment, or 100 nm wavelength range and 1 nm wavelength increment)

Conditions

Maximum Safe Optical Input Power: +4 dBm

Warm-up time: 90 minutes

Storage temperature: -40°C to +70°C

Operating temperature: +10°C to +35°C

Specified temperature: +20°C to +30°C

Humidity: < 80% r.h., non-condensing

Recommended re-calibration period: 1 year

Regulatory Information

- **Compliance with Canadian EMC Requirements**
This ISM device complies with Canadian ICES-001.
Cet appareil ISM est conforme à la norme NMB-001 du Canada.

Table 8-7. Notice for Germany: Noise Declaration

Acoustic Noise Emission	Geräuschemission
LpA < 70 dB	LpA < 70 dB
Operator position	am Arbeitsplatz
Normal operation	normaler Betrieb
per ISO 7779	nach DIN 45635-1

Declaration of Conformity

 Agilent Technologies	DECLARATION OF CONFORMITY According to ISO/IEC Guide 22 and CEN/CENELEC EN 45014	
---	--	---

Manufacturer's Name: Agilent Technologies International sarl
Manufacturer's Address: Rue de la Gare 29
 CH-1110 Morges
 Switzerland

Declares under sole responsibility that the product as originally delivered

Product Name: Photonic Dispersion and Loss Analyzer
Product Number: 86038B containing N5230A and 83427C
Product Options: *This declaration covers all options of the above products.*

complies with the essential requirements of the following applicable European Directives, and carries the CE marking accordingly:

- The Low Voltage Directive 73/23/EEC, amended by 93/68/EEC
- The EMC Directive 89/336/EEC, amended by 93/68/EEC

and conforms with the following product standards:


EMC	Standard	Limit
	IEC 61326:1997+A1:1998+A2:2000 / EN 61326:1997+A1:1998+A2:2001+A3 :2003	
	CISPR 11:1997+A1:1999 / EN 55011:1998+A1:1999	Group 1 Class A
	IEC 61000-4-2:2001 / EN 61000-4-2:1995+A1:1998+A2:2001	4kV CD, 8kV AD
	IEC 61000-4-3:2002+A1:2002 / EN 61000-4-3:2002+A1:2002	3 V/m, 80-1000 MHz
	IEC 61000-4-4:2001 / EN 61000-4-4:1995+A1:2001+A2:2001	0.5kV signal lines, 1kV power lines
	IEC 61000-4-5:2001 / EN 61000-4-5:1995+A1:2001	0.5 kV line-line, 1 kV line-ground
	IEC 61000-4-6:1995+A1:2000 / EN 61000-4-6:1996+A1:2001	3V, 0.15-80 MHz
	IEC 61000-4-8:2001 / EN 61000-4-8:1993+A1:2001	30 A/m
	IEC 61000-4-11:1994+A1:2000 / EN 61000-4-11:1994+A1:2001	1 cycle/100%
	Canada: ICES-001:1998	
	Australia/New Zealand: AS/NZS CISPR 11:2004	
Safety	IEC 61010-1:2001 / EN 61010-1:2001	
	Canada: CAN/CSA-C22.2 No. 61010-1:2004	
	USA: UL 61010-1:2004	

Supplementary Information:

The products were tested in a typical configuration with Agilent Technologies test systems.

This DoC applies to above-listed products placed on the EU market after:

2005-August-01
 Date


 Hans-Martin Fischer
 Name
 Product Regulations Representative
 Agilent Technologies
 Title

For further information, please contact your local Agilent Technologies sales office, agent or distributor.

Index

#

- 86038B
 - block diagram 58
- 86122A
 - firmware revision 290
 - installation 31
 - problems 288, 295
 - settings 99
 - wavelength correction 76

A

- Action commands 228
- activate
 - markers 130
 - trace 129
 - zoom 130
- active marker 122
- active trace 123
- Active X interface 150
- adjusting modulator bias 74, 75
- aliasing errors 61, 65
- amplitude phase conversion effects 115
- amplitude scale 120
- amplitude zoom 131
- Analysis - Curve Fit commands 231
- angled connectors 329, 338
- assigning trace names 45
- auto scale functions 120, 121, 130

B

- Base commands 222
- Begin Measurement 132
- block diagram 58
- bmp file format 95
- Broadband Component 91
- broadband devices 85

C

- C# programming 187
- caution, definition of 24
- CE mark 24
- change view 129

- changing network settings 139
- changing trace names 45
- Cleaning
 - Additional equipment 308
 - Connector Interfaces 314
 - General Procedure 311
 - High-power single-mode connectors 312
 - How to clean connectors 311
 - Importance of 305
 - Materials required 305
 - Preserving Connectors 310
 - Safety Precautions 304
- client to server commands 218
- communication problems 289, 291
- computer machine name, viewing or changing 139
- Connectivity commands 235
- Connectors
 - How to clean 311
 - Preserving 310
- continuous sweep 114
- control functions 132
- CSA mark 25
- csv file format 95
- curve fit algorithms 127
- Curve Fit function 127
- customize normalization settings 107

D

- data errors 297
- dB per division 120
- declaration of conformity 344
- default settings 98
- default values
 - display 99
 - measurement settings 98
- device types, choosing setup parameters 85
- dispersion measurements 51
- Display commands 237
- display markers 121
- Display menu 119
- display multiple graphs 129
- DNS, using 139
- drift correction 111
- DUT, choosing setup parameters 85

E

- enumerations 150
- erbium doped fiber, measuring 88
- event table, remote programming 154
- Excel programming example 200
- export files 95

F

- FDA laser safety 26
- Fiber Bragg Grating dispersion compensators, measuring 87
- fiber dispersion compensating module, measuring 86
- fiber length measurements 56
- fiber measurement 85
- File commands 240
- file formats 95
- File menu 95, 97
- firmware upgrade procedure 301
- frequency zoom 131
- front panel
 - connections 32
 - overview 44

G

- GD ripple 69
- GD trace offset 123
- GPIB problems 289, 291
- group delay
 - noisy trace 298
 - ripple 61, 298
 - smoothing 53
 - trace offset to zero 121

H

- hard drive backup and recovery 302, 303
- horizontal zoom 131

I

- IF bandwidth 110
- imaging, causes of 66
- improving signal to noise 111

insertion loss 51
installation 28
installing remote client 141
instrument
 preset 98
 returning for service 326
 warm-up 35
internal thermal regulator 100
Introduction 80
ISM1-A 25

J

jpg file format 95

K

key functions 44
keyboard, connecting 29

L

Lab View programming example 203
lambda icon 112
LAN connection 138
laser
 classifications 26
 safety 26
 setup 112
length
 measurements 56, 106
 normalization 76
Length commands 243

M

making a measurement 47
marker
 position 121
 search 122
 search criteria 122
 setup 121
 to center 123
 to reference position 123
measure control functions 132
Measure menu 115
measurement
 length 106
 method 57
 modes 77
 noise 62
 progress indicator 115
 reducing error 111
 repeatability 61
 resolution 65
Measurement Range commands 247
measurement range, setting 106
measurements 51
 fiber 85

 length 56
 optimized 51
 polarization dependant loss 56
 selecting types of 104
 theory of 59
modulation phase shift method 57, 61
modulator bias 74, 75
modulator bias problems 294
mouse, connecting 29
mouse-only functions 130
multiple graphs 129
multi-wavelength meter
 firmware revision 290
 installation 31
 problems 288, 295
 settings 99
 wavelength correction 76

N

Narrowband Filter or DWDM Wavelength Routing Devices 90
network settings, how to change 139
networking 138
next trace selection 129
noise declaration 343
noise, reducing 110
normalization
 customizing 107
 initiating 115
 problems 294
 selecting ports 117
 setting measurement range 107
Normalization Range commands 249

O

object select 130
offset GD trace 123
offset GD trace to zero 121
operation verification 35
optical modulator bias 74, 75
optical modulator bias problems 294
optimizing measurements 51

P

pan zoom 131
parts list 28
PDLA Remote Programming 212
peak excursion 122
peak marker 122
performance verification 35
phase dynamic accuracy 74, 75, 115
phase wrapping 63
phase wrapping errors 61
pit marker 122

polarization dependant loss 56
polarization mode dispersion 54
polarization scanner 59
port selection 106
ports, selecting for normalization 117
preset display 99
preset states 98
print 96
printer port 34
Printing commands 255
printing data 96
programming example
 C# 187
 Excel 200
 Lab View 203
 VB 6.0 168
 VB.NET 178

Q

Quick Setup 47, 104

R

real time port monitor 101
 examples 102
rear panel connections 32
recall data 95
reducing measurement error 111
reference level position 120
reference level, setting 120
regulatory information 343
remote client, installation 141
repeat sweep 114
resetting the instrument 98
resolution 108
Resolution commands 256
resolution, increasing 65
Results commands 260
returning for service 326
RF modulation frequency
 maximum values 64
 setting 108
ripple 298
 amplitude accuracy 68
 GD or insertion loss 52
ripple error, sideband separation 69

S

Safety
 Cleaning 304
 safety 24
 safety considerations 26
 safety information 24
 safety symbols 24

- samples per point 110
- save data 95
- selecting a trace 129
- sensitivity 108
- Sensitivity commands 264
- sensitivity, setting 110
- server to client events 213
- service
 - returning for 326
- settle TLS 112
- setup
 - markers 121
 - parameters 83
 - parameters, theory of 61
 - wizard 104
- Setup menu 104
- setup wizard 47
- shipping contents 28
- Show DGD Contribution function 125
- show markers 121
- single sweep 114
- smoothing
 - algorithms 125
 - selection 125
- software upgrade procedure 301
- span, display setting 119
- startup problems 287, 291
- Status commands 267
- status log, error messages 99
- stepped measurements 77
- sweep problems 294
- swept measurements 77
- symbols, safety 24
- System commands 272
- system diagram 58
- system preset 98

T

- TCP/IP configuration 139
- temperature icon 100
- theory of operation 57
- thermometer icon 100
- tif file format 95
- TLS
 - commands 278
 - firmware revisions 290
 - problems 288, 299
 - settling 112
- toggle displays 129
- toolbar functions 130
- trace names, defining 45
- trace points, setting 108
- trigger mode 114, 132

- troubleshooting 287
- turning markers on 130

U

- units per division 120
- upgrade procedure 301
- user interface 44

V

- VB 6.0 programming 168
- VB.NET programming 178
- verification test 35
- vertical zoom 131

W

- warm-up time 35
- warning, definition of 24
- wavelength correction 76
- wavelength correction problems 295
- wavelength meter
 - firmware revision 290
 - installation 31
 - problems 288, 295
 - settings 99
 - wavelength correction 76
- wavelength scale 119
- wavelength zoom 131
- WINS, using 139

X

- x scale 119

Y

- y scale, amplitude display 120

Z

- zero GD trace 123
- zoom functions 130, 131

© Agilent Technologies Manufacturing GmbH &
Co.KG 2006



Printed in Germany
Second Edition, July 2006
80638-90B03