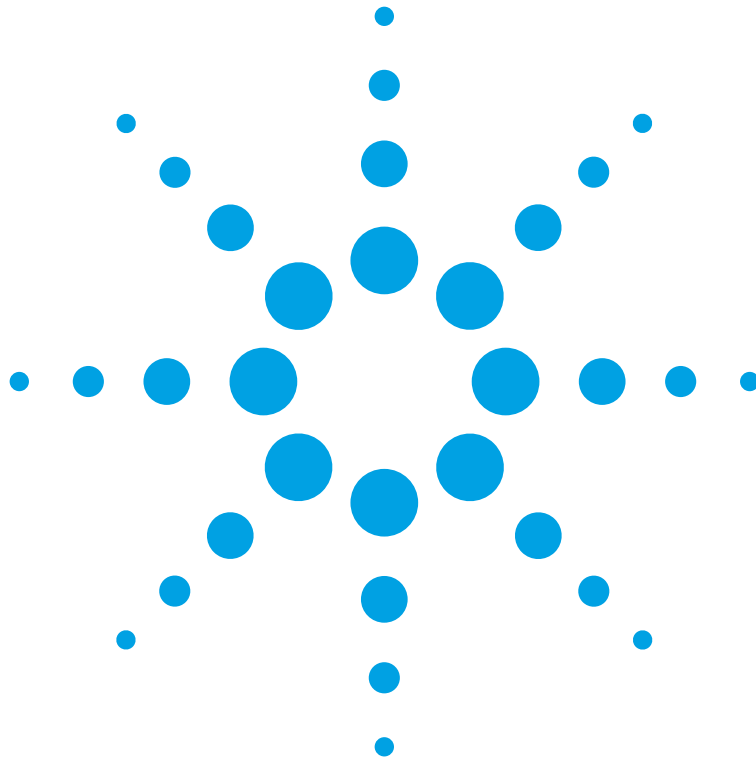


Agilent 86038A  
Optical Dispersion Analyzer  
User's Guide



# Notices

© Agilent Technologies, Inc. 2002-2004  
This document contains proprietary information that is protected by copyright. All rights are reserved.

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

Agilent Technologies Deutschland GmbH  
Herrenberger Str. 130  
71034 Böblingen  
Germany

## Manual Part Number

86038-90006

## Edition

Third edition, December 2004

Second edition, January 2003

First edition, October 2002

## 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.

## Safety Notices

### CAUTION

**A CAUTION notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in damage to the product or loss of important data. Do not proceed beyond a CAUTION notice until the indicated conditions are fully understood and met.**

### WARNING

**A WARNING notice denotes a hazard. It calls attention to an operating procedure, practice, or the like that, if not correctly performed or adhered to, could result in personal injury or death. Do not proceed beyond a WARNING notice until the indicated conditions are fully understood and met.**

## Warnings and Notices

### **WARNING**

To avoid the possibility of injury or death, you must observe the following precautions before switching on the instrument.

Insert the power cable plug only into a socket outlet provided with a protective earth contact. Do not negate this protective action by the using an extension cord without a protective conductor.

---

### **WARNING**

Never look directly into the end of a fiber or a connector, unless you are absolutely certain that there is no signal in the fiber.

---

## 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 (1888 900 8921(FAX))
Europe:	(31 20) 547 2111 (31 20) 547 2190 (FAX)
Japan:	0120 421 345 0120 421 678 (FAX)
Mexico	(52 55) 5081 9469 (52 55) 5081 9467 (FAX)
Australia:	1 800 629 485 1 800 142 134 (FAX)
Asia-Pacific:	800 930 871 800 908 476 (FAX)
Brazil	(55 11) 4197 3600 (55 11) 4197 3800 (FAX)

# Contents

<b>Getting Started</b>	9
<hr/>	
Introduction	10
General Safety Considerations	11
Setting Up the Optical Dispersion Analyzer	16
Performance Verification	20
Navigating the Front Panel Functions	26
Quick Setup Procedure	30
<b>Measurement Concepts</b>	31
<hr/>	
Introduction	32
Understanding the Attributes Being Measured	33
Description of the 86038A	41
<b>Making Measurements</b>	59
<hr/>	
Introduction	60
Quick Setup Measurement Process	61
Flexible Measurement Process	62
Setup Parameters and What They Do	63
Choosing Setup Parameters for Your Device	65
Narrowband Filter Measurement	72
Fiber Measurement	77
<b>Function Reference</b>	87
<hr/>	
Introduction	88
File Menu	89
View Menu	90
System Menu	91

Setup Menu	97
Measure Menu	107
Display Menu	111
Navigating the Front Panel	121
Toolbar Functions	122
<b>Remote Operation</b>	<b>125</b>
Overview	126
Recommended Control Sequence	130
Front Panel Maps	147
Example 1: Simple Application for Controlling the ODA	154
Example 3: Control from LabView	185
Example 4: Complete application - RepeatMeas	186
<b>Programming Commands</b>	<b>191</b>
ODA Remote Programming	192
Server to Client Events	193
Client to Server Commands	198
Base Commands	201
<b>Troubleshooting and Upgrades</b>	<b>253</b>
Introduction	254
Troubleshooting Startup Problems	255
Troubleshooting GPIB Communications Problems	257
Troubleshooting Problems with Remote Programming	261
Troubleshooting Sweep and Normalization Problems	262
Troubleshooting Incorrect Data	263
Troubleshooting the TLS	265
Solving Measurement Problems	266
Software Upgrade Procedure	270
Hard Drive Backup and Recovery	271

---

<b>Cleaning Connections for Accurate Measurements</b>	272
<b>Returning the Instrument for Service</b>	282
<b>Specifications and Regulatory Information</b>	283
<hr/>	
<b>Introduction</b>	284
<b>Angled Connector Specifications</b>	285
<b>Specifications</b>	286
<b>Regulatory Information</b>	290
<b>Declaration of Conformity</b>	291
<b>Glossary</b>	293
<hr/>	
<b>Dispersion Terminology</b>	294
<b>Index</b>	301
<hr/>	







# 1

## Getting Started

Introduction.....	10
General Safety Considerations.....	11
Setting Up the Optical Dispersion Analyzer.....	16
Agilent 86038A Front and Rear Panels.....	18
Performance Verification.....	20
Navigating the Front Panel Functions.....	25
Defining Traces.....	27
Quick Setup Procedure.....	29

# Introduction

The 86038A optical dispersion analyzer (ODA) is designed to take you into the future of optical dispersion measurements. It is designed specifically for the development of a wide range of devices including DWDM components, dispersion compensating gratings, optical fiber, amplifiers, and systems.

## Key benefits are:

- Simultaneous PMD, CD, GD, PDL, and IL measurements with a single connection and single setup reduces measurement time and test uncertainty.
- Excellent group delay repeatability, differential group delay accuracy, gain/loss accuracy, and wavelength accuracy. Refer to [Chapter 8, “Specifications and Regulatory Information”](#).
- Extremely high speed swept measurements: CD and IL measurements of 1601 wavelength points can be obtained in less than 10 seconds.
- CD measurements are automatically corrected for PMD artifacts.
- Two port DUT testing allows simultaneous testing of two devices, two ports of the same DUT, or simultaneous transmission and reflection measurements that increase throughput.
- Wider operating wavelength range: 1260 to 1640 nm (O to L band)
- Specified operation: 1510 to 1620 nm
- Dynamic drift reduction provides excellent stability and accuracy when the environmental conditions of the room or the test device are gradually changing.
- Automatic modulation frequency selection saves measurement time by automatically choosing the best modulation frequency and avoiding phase wrap errors for fiber test.
- External CDRW drive allows you to easily store data.

## 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.

---

Proceed to “Setting Up the Optical Dispersion Analyzer” on page 16 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.
-  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**

Do not use too much liquid in cleaning the Optical Spectrum Analyzer. Water can enter the front-panel keyboard, **damaging sensitive electronic components.**

---

**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.

---

# Setting Up the Optical Dispersion Analyzer

Depending on the options ordered, the 86038A ODA will be shipped in either four or five boxes. The contents of each box is:

## **Fiberglass Case:**

- Performance Network Analyzer/Test Set
- One BNC cable
- One GPIB cable (two, if system includes a wavelength meter)
- One 2-meter insulated fiber patchcord
- One 40-cm insulated fiber patchcord
- One parallel cable
- User's Guide
- System CD-ROM

## **Accessories Box:**

- Keyboard
- Mouse
- CDRW
- USB hub
- IVF reel
- TLS front cover

## **TLS Mainframe Box**

## **TLS Module Box**

## **Wavelength Meter Box (optional)**

### **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 ODA 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. Keep the shipping materials for repair or replacement.



**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 ODA, refer to [“Cleaning Connections for Accurate Measurements”](#) on page 272.

- 2 Install the TLS module into the mainframe by sliding the module in from the rear of the mainframe and tightening the four thumb screws on the rear panel.
- 3 Install the TLS front cover.

**CAUTION**

The PNA and test set are assembled as one unit. The combined weight of the performance network analyzer and test set is 35 kg (76 lbs). The removal of these instruments requires two persons using proper lifting techniques.

- 4 Place the performance network analyzer (PNA) and test set on top of the TLS mainframe.
- 5 On the rear panel, connect the cables as shown in [Figure 2](#).
- 6 Connect the USB hub to the USB port. Connect the mouse and keyboard to the USB hub. (Alternately, you can connect either the keyboard or mouse to the USB port on the front panel.)
- 7 On the front panel, connect the 40 cm insulated patchcord between TLS Output 2 and the TLS Input as shown in [Figure 1](#).
- 8 Turn on the PNA and TLS and allow the instruments to warm up for 90 minutes.

**For the Wavelength Meter Option Only**

- 9 Place the 86122A multi-wavelength meter on top of the PNA.
- 10 Connect a GPIB cable from the 86122A to the GPIB port on either the PNA or TLS mainframe.
- 11 On the 86122A, press **Setup** > **GPIB Setup** and set the address to **10**.
- 12 Connect a patchcord from the 86122A Optical Input to TLS Output 1.
- 13 Continue with “Performance Verification” on page 20.

# Agilent 86038A Front and Rear Panels

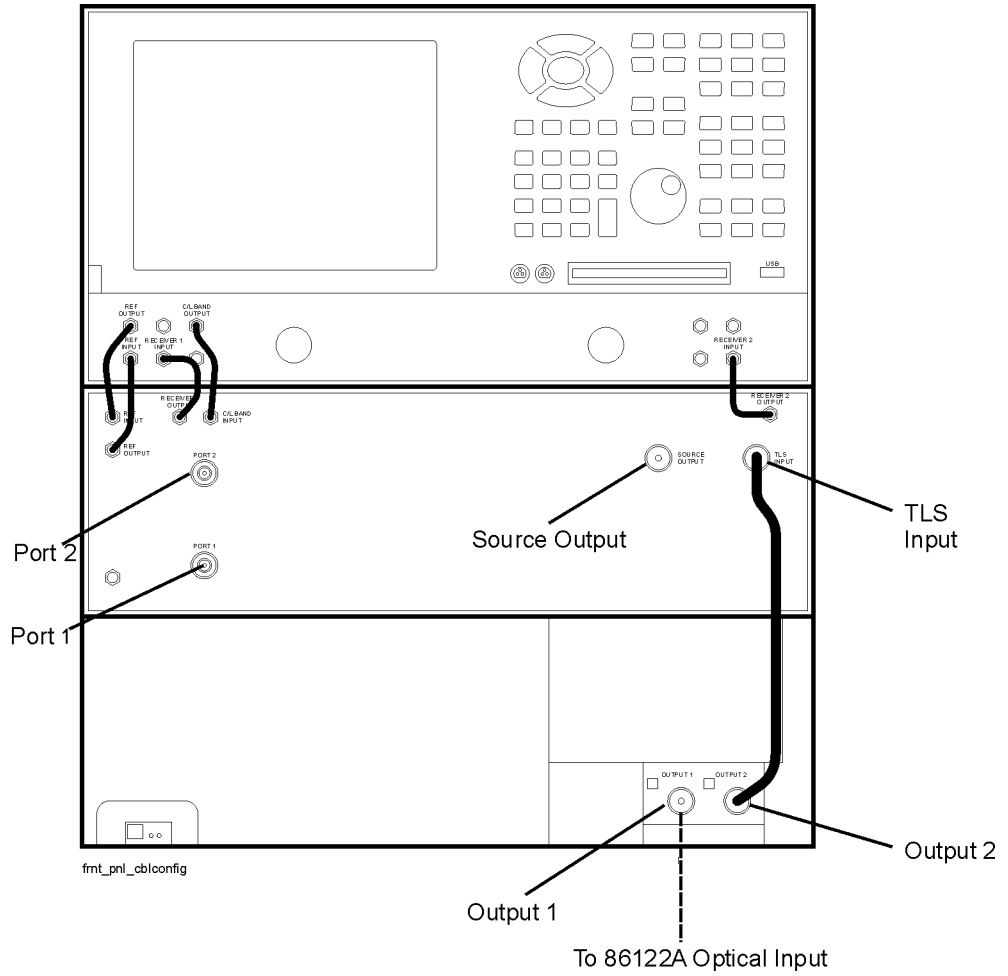


Figure 1 Front Panel

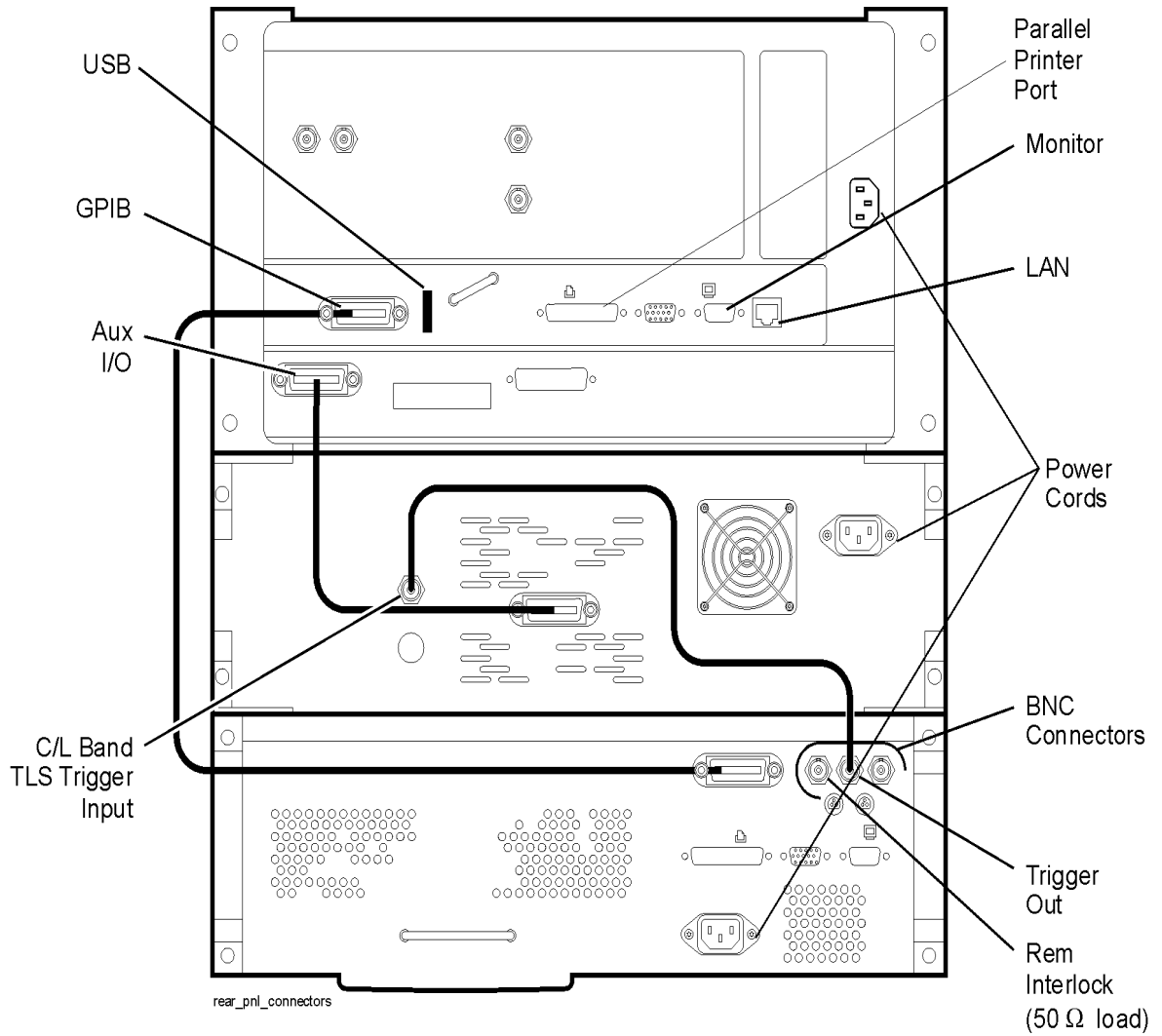


Figure 2 Rear Panel

# Performance Verification

## Purpose

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

The ODA is correctly connected

The ODA 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 ODA.

## Methodology

The verification device under test (DUT) is a 10 km spool of dispersion-shifted fiber. It has an approximately quadratic group delay curve, which when differentiated gives a linear chromatic dispersion curve. Using a quadratic curve fit for the measured group delay, the 86038A estimates the wavelength for which the chromatic dispersion curve passes through zero or, zero-dispersion wavelength. The instrument setup used is meant to duplicate settings used by NIST to measure fiber verification spools.

After completion of the on-site verification procedure, the results are compared against the data recorded at the factory. Specific parameters are specified to prove the relative group delay accuracy of the ODA.

The measurement procedure requires 81600B-160 or 81600B-150 or 81600B-200 tunable laser installed.

## Things to Consider

- Dispersion measurements are very sensitive to temperature changes. Insure that the ODA 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 Connections for Accurate Measurements” on page 272.](#)

- 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 27](#).

## Procedure

- On the front panel of the ODA, press **Preset**.

### Port 1 Stepped GD and Length Measurements

**1** Load the setup file corresponding to the Port 1 Stepped measurement:

- Press **Recall**.
- Browse to D:\ODA and click the file named IVF\_PORT1\_STEPPED.ODA.
- Click **Recall**.

This will set the instrument to the following conditions:

- PDA enabled

Table 1

Parameter	Setting	Path
Dispersion Measurement Type	GD/CD Stepped	Setup > CD/PMD > Dispersion Measurement Type > GD/CD Stepped
Length Mode	Measure	Setup > Length > Length Mode > Measure
Index of Refraction	n = 1.4711	Setup > Length > Length Mode > Index of refraction
Measurement Port	Port 1	Setup > Ports > Active Ports > Port 1
Start	1515 nm	Setup > Meas Range > Start
Stop	1580 nm	Setup > Meas Range > Stop
Normalization Settings	Use Measurement Settings	Setup > Norm Range > Use measurement settings
Increment	1 nm	Setup > Sens/Res > Resolution > Increment
Modulation Frequency	2000 MHz	Setup > Sens/Res > Resolution > RF Mod Freq
IF Bandwidth	30 Hz	Setup > Sens/Res > Sensitivity > IF Bandwidth
Samples/Point	51	Setup > Sens/Res > Sensitivity > Samples/Point

**2** Connect the ODA Source Output Port to ODA Optical Input Port 1 via the insulated 2-meter patch cord that shipped with your ODA.

- 3 Normalize (about 17 minutes):
  - a Press **Begin Norm.**
  - b Ensure that the PDA check box is checked.
  - c Ensure the active port is selected as Port 1.
  - d Click **Normalize** to start the normalization.
- 4 **Close** the Normalization panel when normalization is complete.
- 5 Connect the verification DUT:
  - a Disconnect the 2-meter patch cord from ODA Optical Input Port 1
  - b Insert the verification DUT by connecting one end to the 2-meter patch cord used during normalization using an FC/PC barrel adaptor (p/n 1005-0256).
  - c Connect the other end of the verification DUT to ODA Optical Input Port 1.
- 6 Measure the length of the verification DUT (about 7 minutes):
  - a Just above the Length Setup title bar, select **Length-Port 1** from the drop-down list.
  - b Press **Begin Length.**

The length will be shown as “Length-1: nnnnn m” where nnnnn is the verification DUT length in meters.
  - c Compare the length measurement with the length measured at the factory. The measurement should match within  $\pm 5\%$  of the factory measured value. Please note that this is not a specified measurement, but provided for your reference.
- 7 Measure the relative group delay of the verification DUT (about 3 minutes):
  - Press **Begin Meas.**
- 8 Compare the Lambda-Zero to the value provided by the factory:
  - a Press **Next Trace** until the left measurement window is selected.
  - b Press **Curve Fit** to display the curve fit setup panel.
  - c The Algorithm field should be set to **Quadratic**.
  - d The Minimum Point number in the Equation window is the stepped Lambda-Zero. Compare this number to the result measured at the factory. The result should match within  $\pm 0.375$  nm of the factory measured value.
- 9 Compare the Dispersion slope at Lambda-Zero to the value provided by the factory:
  - a Press **Next Trace** until the right measurement window is selected.
  - b Press **Curve Fit** to display the curve fit setup panel.
  - c The Algorithm field should be set to **Sellmeier-5**.
  - d The  $f''$  (min) number in the Equation window is the stepped Dispersion Slope at Lambda-Zero. Compare this number to the result measured at the factory. The result should match within  $\pm 0.075$  ps/nm<sup>2</sup> of the factory measured value.

## Port 1 Swept GD Measurement

- 1 On the front panel of the ODA, press **Preset**.
- 2 Press **Recall** and recall the state file D:\ODA\IVF\_PORT1\_SWEPT.ODA.

This will set the instrument to the following conditions:

- 3 Connect the normalization patch cord:

Table 2

Parameter	Setting	Path
Dispersion Measurement Type	GD/CD Swept	Setup > CD/PMD > Dispersion Measurement Type > GD/CD Swept
Measurement Port	Port 1	Setup > Ports > Active Ports > Port 1
Start	1515 nm	Setup > Meas Range > Start
Stop	1580 nm	Setup > Meas Range > Stop
Normalization Settings	Use Measurement Settings	Setup > Norm Range > Use measurement settings
Trace Points	1601	Setup > Sens/Res > Resolution > Trace Points
Modulation Frequency	2000 MHz	Setup > Sens/Res > Resolution > RF Mod Freq
IF Bandwidth	30 Hz	Setup > Sens/Res > Sensitivity > IF Bandwidth

- 4 Connect the ODA Source Output Port to ODA Optical Input Port 1 via the insulated 2-meter patch cord that shipped with your ODA.
- 5 Perform Normalization (about 1 minute). Ensure that the PDA Correction box is cleared.
- 6 Close the Normalization panel when normalization is complete.
- 7 Connect the verification DUT:
  - a Disconnect the 2-meter patch cord at Optical Input Port 1.
  - b Insert the verification DUT as done for the Port 1 Stepped measurement.
- 8 Press **Begin Meas** (about 1 minute).
- 9 Compare the Lambda-Zero to the value provided by the factory: The result should match within  $\pm 0.625$  nm of the factory measured value.
  - a Press **Next Trace** until the left measurement window is selected.
  - b Press **Curve Fit** to display the curve fit setup panel.
  - c The Algorithm field should be set to **Quadratic**.
  - d The Minimum Point number in the Equation window is the swept Lambda-Zero. Compare this number to the result measured at the factory.
- 10 Compare the Dispersion slope at Lambda-Zero to the value provided by the factory:
  - a Press **Next Trace** until the right measurement window is selected.

- b** Press **Curve Fit** to display the curve fit setup panel.
- c** The Algorithm field should be set to **Sellmeier-5**.
- d** The  $f''(\text{min})$  number in the Equation window is the swept Dispersion Slope at Lambda-Zero. Compare this number to the result measured at the factory. The result should match within  $\pm 0.125 \text{ ps/nm}^2$  of the factory measured value.

### Port 2 Stepped GD and Length Measurements

- 1** Press **Recall** to recall the state file D:\ODA\IVF\_PORT2\_STEPPED.ODA.
- 2** Repeat the “[Port 1 Stepped GD and Length Measurements](#)” procedure above, substituting Optical Input Port 2 where appropriate.

### Port 2 Swept GD Measurement

- 1** Press **Recall** to recall the state file D:\ODA\IVF\_PORT2\_SWEPT.ODA.
- 2** Repeat the “[Port 1 Swept GD Measurement](#)” procedure above, substituting Optical Input Port 2 where appropriate.

## Corrections

### Lambda-Zero

The zero-dispersion wavelength of dispersion-shifted single-mode fiber has a temperature coefficient of approximately  $+0.03 \text{ nm}/^\circ\text{C}$ . The fiber temperature at which the Factory Data measurements were performed is listed on the factory-provided data sheet. If the On-Site Verification is performed at a different temperature, use the following formula to adjust the post-installation results for comparison with the Factory Data. All temperatures are in degrees C.

$$\lambda_0 (\text{adjusted}) = \lambda_0 (\text{measured On-Site}) - 0.03 \text{ nm} \times (\text{Temp: On-Site} - \text{Temp: Factory})$$

### Dispersion Slope at Lambda-Zero

The dispersion slope at the zero-dispersion wavelength, expressed in  $\text{ps/nm}^2$ , is approximately constant with temperature. Thus no temperature correction is necessary.



# 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.

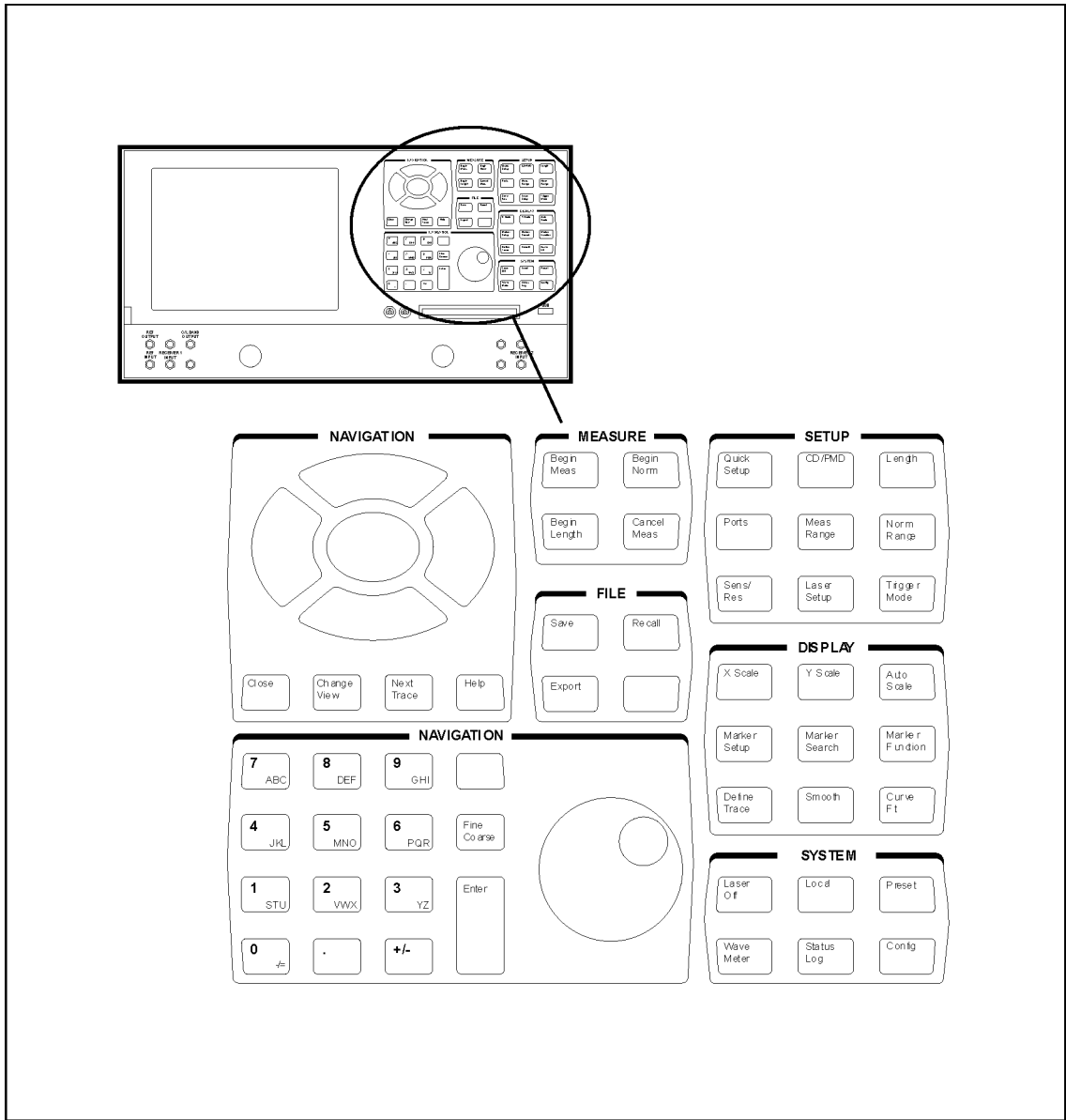


Figure 3 86038A User Interface

## 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 2.

- 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 2* 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 2* and *DGD* as the measurement type.
- 7 Press *Close* to save the trace definitions.

For each trace you can specify either Port 1 or Port 2 measurement traces and one of five measurement types.

- GD (Group Delay)
- CD (Chromatic Dispersion)
- DGD (Differential Group Delay)
- PDL (Polarization Dependant Loss)
- Gain Loss

### 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.

**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 ODA. 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 62.

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 measurement that can be made, specific devices being measured, and the setup parameters used to make the measurements.

## Setup the Normalization

- Connect the patch cord for the normalization.
- Check the connections with the Real Time Port Monitor. Refer to [“Real-Time Port Monitor”](#) on page 94.

## 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 . . . . .	32
Understanding the Attributes Being Measured . . . . .	33
Gain/Loss . . . . .	34
Group Delay/Chromatic Dispersion . . . . .	35
Polarization Mode Dispersion/Differential Group Delay . . . . .	37
Polarization Dependent Loss (PDL) . . . . .	39
Fiber length . . . . .	40
Description of the 86038A . . . . .	41
Measurement Method . . . . .	42
Choosing Setup Parameters . . . . .	43
DUT Setup Configurations . . . . .	54
Normalization and Test . . . . .	55
Swept and Stepped Measurement Modes . . . . .	57

# 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.

---



# Understanding the Attributes Being Measured

The 86038A optical dispersion analyzer allows you to measure the following attributes:

- Gain/Loss
- Group Delay (GD)
- Chromatic Dispersion (CD)
- 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 with transmission through a two-port device. It is defined as  $10 \log(\text{power in}/\text{power out})$  and is always a positive number. The mechanisms that produce insertion loss include simple absorptive effects as well as scattering and reflective effects.

#### *Tips on Measuring Insertion Loss (IL)*

- For the most accurate IL measurements, normalize with the polarization scanner running (GD/CD + DGD/PDL Stepped), using a 30 Hz IF bandwidth. This allows any DGD and PDL effects to be removed from the normalization. [Refer to “CD/PMD” on page 97](#) and to [“Sensitivity/Resolution” on page 101](#).
- 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.

The 86038A displays loss and gain within a single trace by representing loss with a positive sign and gain with a negative sign. This convention was selected because historically, the majority of the measurement applications have involved passive devices and fibers.

## Group Delay/Chromatic Dispersion

Mathematically, the group delay of a component, fiber or network is 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 86038A 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.

### *Tips on Measuring Group Delay (GD)*

- In the swept mode, over-sampling the GD shape offers the possibility of using smoothing to reduce the noise. If sampling is marginal, then smoothing sacrifices information.
- For the most accurate GD measurements, normalize with the polarization scanner running (GD/CD + DGD/PDL Stepped), using a 30 Hz IF bandwidth. This allows any DGD and PDL effects to be removed from the normalization. Refer to [“CD/PMD” on page 97](#) and to [“Sensitivity/Resolution” on page 101](#).
- When making precision GD measurements, use the Real Time Port Monitor to verify GD drift before normalizing or measuring. Fiber connectors are a frequent cause of GD drift and must be clean to a much higher standard than commonly needed for fiber optic measurements of insertion loss. Fibers require protection from normal room air currents. Fibers need to be secured and allowed time to settle after handling. Refer to [“Real-Time Port Monitor” on page 94](#).

- 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.

#### *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 118](#).

## Polarization Mode Dispersion/Differential Group Delay

The phenomenon of PMD was first observed in optical fiber. The term PMD refers to the overall process in which a lightwave decomposes into two differently polarized waves that travel with slightly different group delays. The attribute called differential group delay (DGD) is defined as the maximum group delay variation over all polarization states and is usually expressed in picoseconds (ps). It is one of two fundamental attributes used to describe first-order PMD. For most optical components, the maximum and minimum group delays at any given wavelength occur at orthogonal (opposite or non-interfering) polarization states. These are the 'modes' of polarization mode dispersion, the second fundamental attribute of PMD.

Additional attributes apply to the PMD of optical fiber. The PMD delay is the average value of the differential group delay over a wide wavelength range.

In highly mode coupled fiber, the PMD coefficient is defined as the PMD delay divided by the square root of the fiber length, expressed in  $\text{ps}/\sqrt{\text{km}}$ .

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)*

- With respect to DGD testing, the group delay noise versus received power is measured during the normalization and later 'backed out' statistically during computation of DUT DGD. The noise measurement is conducted at the center of the normalization range, so strictly speaking it is appropriate only to that set of conditions (TLS output, test set loss, receiver sensitivity, and so on). In other wavelength regions the noise correction may be less exact.

- When measuring DGD, any GD drift or disturbance of the DUT GD during the scan time of 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.
- When making precision DGD measurements, use the Real Time Port Monitor to verify GD drift before normalizing or measuring. Fiber connectors are a frequent cause of GD drift and must be clean to a much higher standard than commonly needed for fiber optic measurements of insertion loss. Fibers require protection from normal room air currents. Fibers need to be secured and allowed time to settle after handling. [Refer to “Real-Time Port Monitor” on page 94.](#)
- When making PDL or DGD measurements, the polarization scanner runs at a fixed speed and produces PDL or DGD changes with the majority of spectral content around 8 Hz. This limits the minimum IF bandwidth when scanning to 70 Hz, which avoids attenuating the 8 Hz components. Using a wider IF bandwidth allows a faster measurement, but raises the noise in the measurement and may slightly bias DGD and PDL toward larger values.
- During PDA normalization, 70 Hz IF bandwidth is used to characterize noise components with the scanner off, for both GD and Magnitude. These noise measurements provide the best correction of DGD and PDL when measuring in a 70 Hz IF bandwidth. [Refer to “Phase Dynamic Accuracy” on page 108.](#)
- PDL and DGD may indicate negative values, which are clearly not possible. This results from noise in the measurement. Negative values are useful when smoothing across wavelength sample points to obtain an unbiased estimate of the average DGD or average 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. If the optical input level to the test device is constant over the measurement time, the PDL is also equal to the peak-to-peak variation of output power over all polarizations.

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, the polarization states corresponding to maximum and minimum loss are diametrically opposite one another on the Poincare sphere.

Higher-order device attributes such as center wavelength and bandwidth of narrowband filters are also polarization dependent. However, it is not always possible to extract the polarization dependence of these attributes from measurements of the PDL at a series of wavelengths; the polarization state that produces the minimum loss at one wavelength may not produce the minimum at all other wavelengths. For this reason, measurement of the polarization dependence of such higher order attributes may require multiple measurements of the device response across its wavelength range, with each complete measurement performed using a different fixed input state of polarization. This produces a set of, for example, 64 device responses at 64 randomly selected states of polarization. The center wavelength and bandwidth are computed for each case, and the peak-to-peak variation in each of these attributes expresses their polarization dependence. The 86038A does not provide a function for measuring these higher order polarization dependencies.

### *Tips on Measuring Polarization Dependent Loss (PDL)*

- When making PDL or DGD measurements, the polarization scanner runs at a fixed speed and produces PDL or DGD changes with the majority of spectral content around 8 Hz. This limits the minimum IF bandwidth when scanning to 70 Hz, which avoids attenuating the 8 Hz components. Using a wider IF bandwidth allows a faster measurement, but raises the noise in the measurement and may slightly bias DGD and PDL toward larger values.
- During PDA normalization, 70 Hz IF bandwidth is used to characterize noise components with the scanner off, for both GD and Magnitude. These noise measurements provide the best correction of DGD and PDL when measuring in a 70 Hz IF bandwidth. [Refer to “Phase Dynamic Accuracy” on page 108.](#)
- PDL and DGD may indicate negative values, which are clearly not possible. This results from noise in the measurement. Negative values are useful when smoothing across wavelength sample points to obtain an unbiased estimate of the average DGD or average PDL.

## Fiber length

This measurement applies to any test device but is usually applied to optical fiber. The fiber length is derived from measurement of the variation of RF modulation phase with RF frequency, at a user defined wavelength. The optical length measurement uncertainty is below  $5 \times 10^{-5}$  for fibre lengths  $> 15$  km. The calculation requires the user to enter the value of the index of refraction; any error in the index of refraction directly impacts the length measurement.



## Description of the 86038A

The Agilent 86038A employs the modulation phase shift (MPS) method for all of its measurements. A block diagram of the system is shown in Figure 4. 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) measures the amplitude and phase of the envelope. For measurements of polarization dependence, the polarization is scanned by a four-stage, fiber coil (mouse ear) polarization controller that directly follows the modulator. In summary, two stimuli are provided by the system: wavelength and state of polarization (SOP). From the resulting information, the various measurement attributes are evaluated.

Dual receivers allow simultaneous transmission/reflection testing, or testing of two test device output ports simultaneously, or comparison of a test device with a reference device. Some of these setups require an external coupler or circulator.

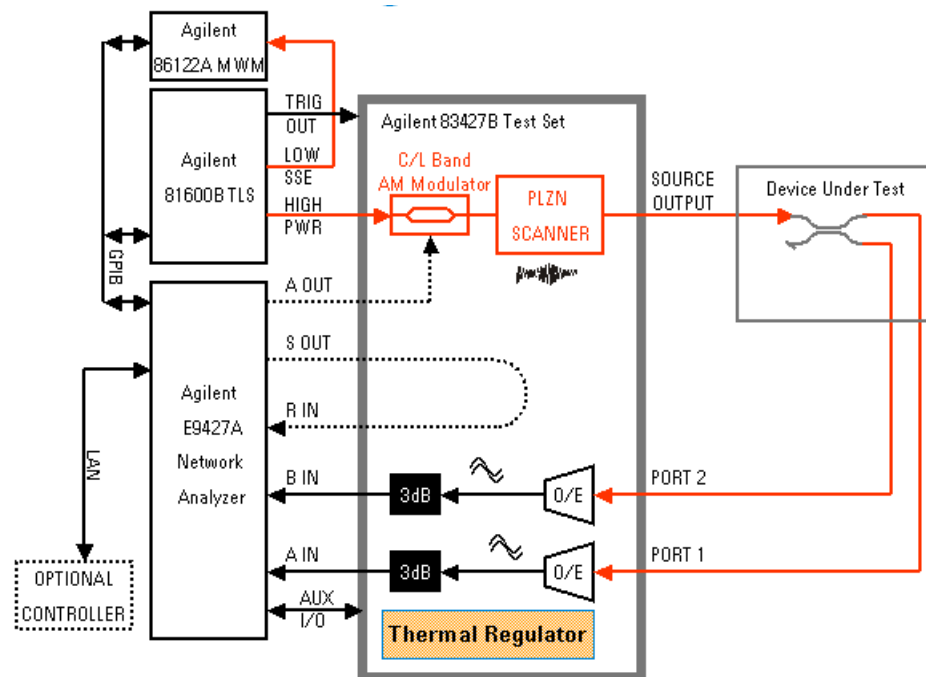


Figure 4 86038A Block Diagram

## Measurement Method

In the group delay/chromatic dispersion measurement, the wavelength is stepped or swept and the change in recovered 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 performed in the step mode with the polarization scanner running. At each wavelength the polarization is scanned for a few seconds. 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 stimulus is polarization rather than wavelength.

The PDL information is obtained from the minimum to maximum variation in amplitude of the recovered modulation envelope, where the stimulus is polarization.

The fiber length measurement is based on the variation of modulation phase as the RF modulation frequency is changed. Intuitively, the length of the test device can be expressed in terms of an equivalent number of modulation periods. At sufficiently low modulation frequencies, even the longest test device is less than a single RF cycle in length. As the RF modulation frequency is increased, the test device becomes 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 86038A. The wavelength is fixed and the stimulus is the RF modulation frequency.

## 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 phases wrapping errors, avoid aliasing errors, and improve the repeatability (reduce the noise) of chromatic dispersion and relative group delay measurements.

### Background

The Agilent 86038A 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:

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

**Equation 1**

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:

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

**Equation 2**

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:

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

**Equation 3**

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:

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

**Equation 4**

[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](#):

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

**Equation 5**

[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 86038A have a range of  $\pm 180$ -degrees. Exceeding the range of the phase detector causes a phase wrapping error. The wrap-free range is expressed by:

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

$$\Delta\phi < \pm 180^\circ$$

#### Equation 6

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  $\pm 180$ -degree range available for the next step. If the next wavelength step produces a phase change larger than  $\pm 180$  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.

The user can avoid phase wrapping errors in the step mode in either of two ways. For fiber measurements, use the Automatic Modulation Frequency Selection feature, which adjusts the modulation frequency as the measurement proceeds to avoid phase wrapping (the wavelength step is not changed from the user selected value). Second, use [Equation 6](#) to check your selections of wavelength increment step and modulation frequency, to assure that the  $\pm 180$  degree limit is not exceeded.

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. Agilent provides 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 86038A.

### 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 (←) indicate that you must decrease the wavelength increment to achieve the desired  $\Delta\tau$  RGD.

Table 3 Maximum RF Modulation Frequency (MHz) for  $<90^\circ$  phase steps  $f_{90} = 1/(4 \times \Delta\tau \times \Delta\lambda)$

$\Delta\tau$ 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	←
100	2500	500	250	50	25	5	←	←
500	500	100	50	10	5	←	←	←
1000	250	50	25	5	←	←	←	←
5000	50	10	5	←	←	←	←	←
10000	25	5	←	←	←	←	←	←
50000	5	←	←	←	←	←	←	←

### Relationship of setup parameters to aliasing

Since all measurements are sampled at discrete wavelengths, there is an implication of the performance between the sampled points. The wavelength increment and the RF modulation sideband separation must be small compared to the wavelength range over which the device under test has variations of interest.

When a device has periodic variations across wavelength, such as a Fabry-Perot filter, an interleaver, amplitude ripple, 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 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 appears as 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 5](#) 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.

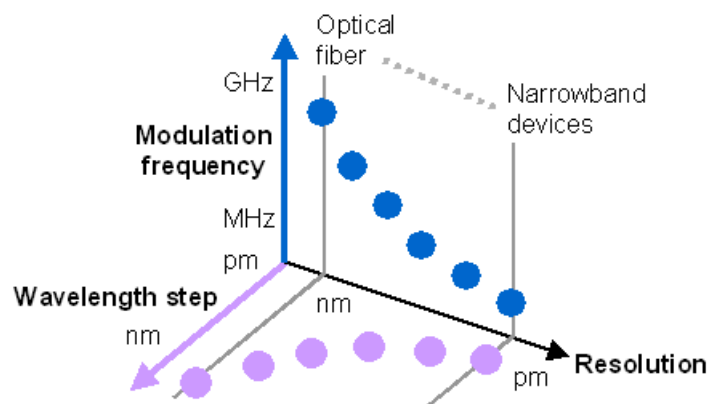


Figure 5 Adjusting wavelength step and mod. freq. to achieve desired resolution

### Relationship of setup parameters to imaging

Since we are using AM modulation, we are sampling at the two AM sidebands simultaneously and reporting the average result. Again there is an implication of the performance between the sampled points. Even if the wavelength increment is arbitrarily small, the RF modulation sideband separation must be small compared to the wavelength range 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 6.

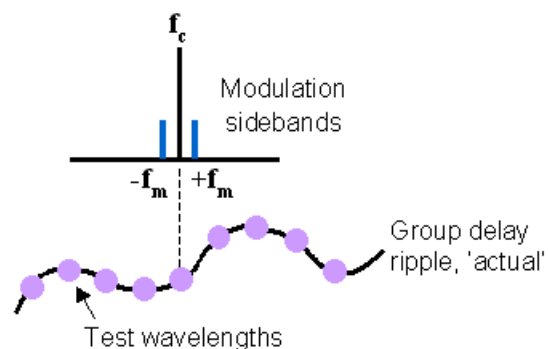


Figure 6 Selecting the modulation frequency for measurement of GD ripple

### 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 7](#) 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)

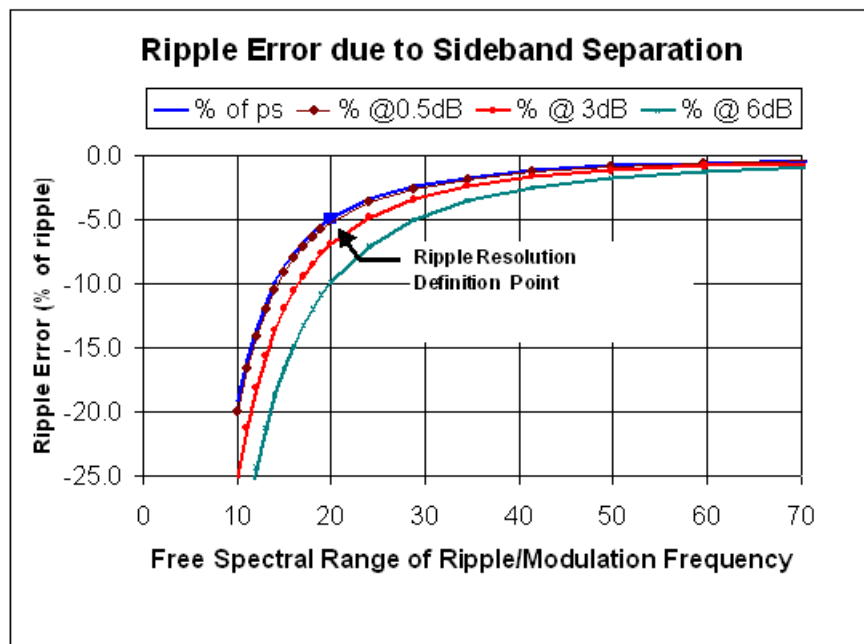


Figure 7 Ripple Error Due to Sideband Separation

The graph shown in [Figure 8](#) 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:

- 8** Enter the x-axis at the estimated period of group delay ripple to be measured.
- 9** Draw a vertical line to the curve representing the desired number of measurement samples per period of the group delay ripple.
- 10** 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 8](#) 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 86038A provides built-in smoothing functions for this purpose.

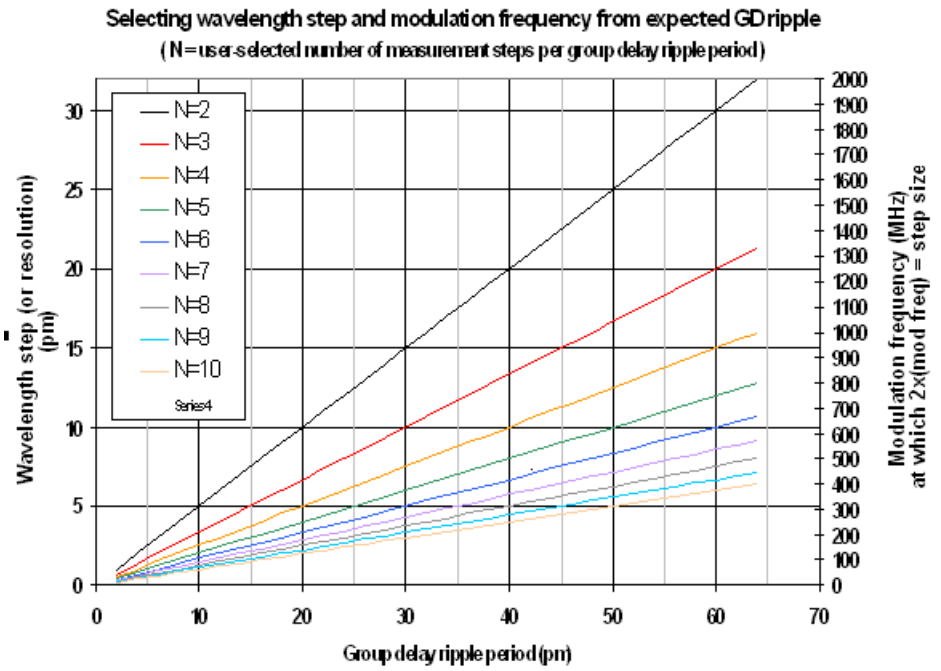


Figure 8 Selecting Wavelength Step and Modulation Frequency for GD Ripple

### Control of resolution and noise

When measuring low-dispersion, narrowband devices at high resolution, we face the noise limit implied by Equation 5. Figure 9 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.

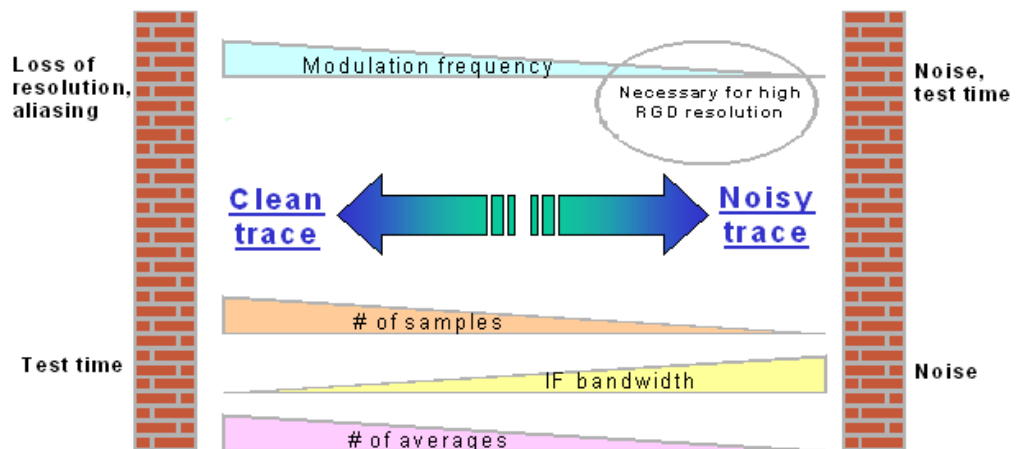


Figure 9 Managing noise in high-resolution measurements

The bottom half of Figure 9 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), 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 86038A's Quick Setup feature, which provides two slider-type controls that in turn adjust a variety of setup parameters. Assuming stepped mode operation, 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 8](#).

**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.

## DUT Setup Configurations

The 86038A's dual receivers allow extra measurement flexibility. The number of inputs and outputs of the DUT are expressed in the form NxM, where N is the number of inputs and M the number of outputs.

- 1** DUT with single input and single output, measured in transmission.
- 2** This setup configuration includes 1x1 devices or a single input/output pair of an NxM device. Either receiver port may be used.
- 3** DUT with single input and single output, measured in transmission and reflection.
- 4** Both receiver ports are used in this configuration. The optical output of the test set is connected to a directional coupler or circulator. The 'reflected' port of the coupler or circulator is connected to one of the receiver ports, and the output of the DUT is connected to the other receiver port.
- 5** DUT with single input and dual outputs, measured in transmission.
- 6** This setup takes advantage of the dual receiver ports to simultaneously measure two outputs of a NxM device.
- 7** Two 1x1 DUT's with inputs measured in transmission.
- 8** This setup is useful for comparing a pair of DUT's, or for situations that require the operator to adjust one device to match the performance of a second device.

## Normalization and Test

### Loss and Group Delay Normalization

The measurement results of the 86038A 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. The normalization measurement establishes a loss and group delay baseline for later measurements of the DUT itself.

### Optical Modulator Bias Adjust

In actual operation, the normalization measurement is accompanied by one or more additional automatic performance optimizations, all under the broad heading of 'normalization'. The 86038A contains an automated modulator bias adjustment function that is implemented when PDA is enabled during a normalization, or by itself through the menu item Measure > Optimize Optical Modulator.

Amplitude modulation in the 86038A 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 lightwaves 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 110](#).

### Phase Dynamic Accuracy

A second optimization corrects for phase dynamic accuracy errors. This correction is optional but is recommended in cases where the DUT is expected to exhibit a significant variation of insertion loss or gain over the measurement wavelength range.

Phase dynamic accuracy (PDA) is a performance attribute of the performance network analyzer (PNA) that 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 86038A 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. At each step, the RF phase is measured and associated with the RF input level. This data is used to correct future measurements. It is recommended that the PDA correction be performed at the same modulation frequency as the measurements it will be applied to.

### **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 98](#) to setup a length measurement.

### **Wavelength Accuracy Correction**

Using the optional 86122A multi-wavelength meter, wavelength correction is performed for stepped normalizations. That is, when Setup > CD/PMD > is set to one of the two stepped measurement modes. Wavelength correction is 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 100](#) and to ["Begin Normalization" on page 107](#).



## Swept and Stepped Measurement Modes

The measurements performed by the 86038A 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 mode. The stepped mode offers the best wavelength accuracy resolution and sensitivity, while the swept mode offers the highest measurement speed. For more information on measurement modes, [refer to “CD/PMD” on page 97](#).

### Stepped mode

In the stepped mode, the user enters values of start and stop wavelengths and wavelength increment (step). The system calculates a series of target wavelengths to which the laser is then tuned. The wavelength resolution of the resulting measurement is strongly influenced by the choice of wavelength increment but is also a function of the modulation frequency, discussed elsewhere.

The stepped mode provides higher wavelength accuracy than the swept mode, because the dynamics of sweeping the TLS and keeping the amplitude/phase detection in unison are not required.

The user may elect to use a wavelength meter to optimize wavelength accuracy. When enabled, this function checks the wavelength accuracy of the TLS at a series of wavelengths distributed across the user-selected measurement wavelength range. The observed wavelength errors are averaged to obtain a single wavelength correction offset value that is used to correct the wavelength scale of following measurements.

The wavelength correction factor is determined during a normalization (if the 86122A wavelength meter was detected at startup) and this factor is automatically applied to future DUT measurements. This arrangement saves measurement time compared with measuring the wavelength before each measurement or measuring the wavelength on a point-by-point basis during the measurement itself. This economy is made possible by the extremely high wavelength stability, repeatability and linearity of the tunable laser.

### **Swept mode**

In the swept mode, the user enters the start and stop wavelengths and the trace points. A sweep time is computed based on the operational limits of the TLS and the RF network analyzer. In operation, the TLS sweeps wavelength versus time while the RF network analyzer periodically samples the recovered RF amplitude and phase. Amplitude and phase samples are associated with wavelength values through the intermediate variable, time. Implicit in this process is a wavelength resolution associated with the wavelength increment between adjacent samples. As in the case of the stepped measurement, wavelength resolution is also affected by the modulation frequency, discussed elsewhere.



# 3

## Making Measurements

Quick Setup Measurement Process .....	61
Flexible Measurement Process .....	62
Setup Parameters and What They Do.....	63
Choosing Setup Parameters for Your Device.....	65
Narrowband Filter Measurement .....	72
Fiber Measurement .....	77

# 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

## Setup the Normalization

- Connect the patch cord for the normalization.
- Check the connections with the Real Time Port Monitor. [Refer to “Real-Time Port Monitor” on page 94.](#)

## 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.

# 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.

## Setup the Normalization

- Connect the patch cords needed for the normalization.
- Check the connections with the Real Time Port Monitor. Refer to “Real-Time Port Monitor” on page 94.

## Normalization Process

- Select DUT configuration and input ports.
- Choose which attributes to measure. Select stepped or swept mode.
- Enable the Length measurement if needed
- Set the measurement wavelength range.
- Set the normalization wavelength range (if different from the measurement range).
- Set the Sensitivity, Resolution, and Dynamic Drift Reduction.
- Begin the Normalization

## Measurement Process

- Connect the DUT.
- Set Trigger to Single or Repeat Sweep.
- 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.

# Setup Parameters and What They Do

A family of setup parameters allow the user to optimize 86038A 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 87.

## 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. In the stepped mode, this is the wavelength range divided by the number of steps minus one. In the swept mode, it is the wavelength range divided by the number of trace points less one.

## 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.

During polarization dependent measurements (DGD and PDL), the number of samples are taken while the polarization state is being scanned. Approximately 201 samples should be taken to ensure that a statistically sufficient sampling of the polarization states has been obtained. Fewer samples will provide a faster measurement at the expense of larger variability (noise). If the operating environment is not controlled for temperature and vibration, too many samples/point can result in erroneously high values of DGD and PDL.

### **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. The default value of  $-5$  dBm is available from the 81600B tunable laser sources over the full operating wavelength range. For the 81600B-160 tunable laser source a power level of  $+5$  dBm is available 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.



# 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 31](#). For more information on the instrument functions, refer to [“Function Reference” on page 87](#).

## Transmission Fiber (All Types)

### *Characteristics*

- Large range of dispersion values across the wavelength range
- Broadband device
- Smoothly changing group delay with relatively little fine structure
- 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

The relatively smooth and analytic behavior of the group delay of this device makes it possible to use larger wavelength increments and lower modulation frequency without loss of wavelength resolution. The following discussion provides setup parameter information specific to each device.

## Fiber Dispersion Compensating Module (DCM)

### *Characteristics*

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

### *Measurement Tips*

- For higher loss fibers, improve signal-to-noise by using a narrower IF bandwidth and a larger number of samples.
- When measuring in step mode, use the automatic modulation frequency selection feature or manually select a modulation frequency and wavelength step appropriate to the expected level of dispersion. Both strategies will eliminate the possibility of RF phase wrap errors.
- Manual selection of the modulation frequency allows the use of the Phase Dynamic Accuracy (PDA) correction process in the step mode. The PDA correction feature is available only at the single modulation frequency at which the normalization was performed. PDA correction is available in the swept mode, as the swept mode always operates at a single, user-selected modulation frequency.
- 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

### *Characteristics*

- High dispersion (steep change in group delay versus wavelength)
- Group delay ripple (fine structure in the group delay curve)

### *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.
- 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 the group delay ripple is not required, manually select a modulation frequency and wavelength step appropriate to the maximum expected level of dispersion. Automatic modulation frequency selection may not be reliable in cases of high group delay ripple.
- Manual selection of the modulation frequency allows the use of the Phase Dynamic Accuracy (PDA) correction process in the step mode. The PDA correction feature is available only at the single modulation frequency at which the normalization was performed. PDA correction is available in the swept mode, as the swept mode always operates at a single, user-selected modulation frequency.
- Of the built-in curve fits, only the linear fit is likely to be useful in this application. Use it when the entire trace is to be fitted and the group delay ripple is required. Export the data to a spreadsheet and subtract the fitted straight line from the measurement data to obtain the group delay ripple.

## Erbium Doped Fiber (Unpumped)

### *Characteristics*

- High loss (up to 10's of dB's)
- Low dispersion (very slight change in group delay over wavelength)
- Broadband device
- 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)

### *Measurement Tips*

- Insert attenuation following the EDFA to avoid overdriving the input of the 86038A. Refer to ["Specifications and Regulatory Information" on page 283](#) 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 band
- Low dispersion, DGD, PDL with possible exception of band edges
- Fine in-band structure to some attributes, possible including group delay ripple.
- Steep filter slopes
- High rejection outside the passband

### *Measurement Tips*

The combination of narrow passband, fine group delay structure, and low dispersion calls for a special strategy

- Swept mode rapidly gathers a large number of points to smooth, taking advantage of the smoothing function. The step mode can take and average a large number of samples at each wavelength, reducing the noise but increasing measurement time over the swept case.
- Start by measuring group delay and loss in the swept mode. This measurement is fast and will usually reveal the fine loss and group delay structure quite well.
- Use a low enough modulation frequency to reveal the fine structure.
- In swept mode group delay measurements, reduce trace noise by means of the built-in smoothing functions. Use a smoothing interval wide enough to reduce noise but not so wide as to smooth away the group delay features of the device.
- In stepped mode, manually select the modulation frequency. The automatic modulation frequency selection feature is intended only for transmission fiber.
- Always select the Phase Dynamic Accuracy (PDA) correction when performing a swept or stepped normalization for filter testing. This correction reduces the degree to which the large variations in device loss with wavelength are translated into small errors in group delay through RF amplitude-to-phase effects in the performance network analyzer (PNA).
- To measure DGD, select the stepped mode. Set the IF bandwidth to 70 Hz to filter out noise while not filtering out the group delay variations caused by polarization scanning. Select 201 samples to allow a sufficiently long polarization scanning time. Set the modulation frequency to 2000 MHz to achieve best signal to noise at the expense of wavelength resolution. Experiment with lower modulation frequency values and correspondingly smaller wavelength increments to reveal fine structure in the DGD response.

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

### *Characteristics*

- Very wide wavelength measurement range
- Low dispersion and DGD
- Loss 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.

# Narrowband Filter Measurement

This measurement example uses a narrow bandpass filter with a center wavelength of 1558 nm and a 3 dB bandwidth of 3 nm.

This measurement procedure will include both a swept and a stepped measurement. Stepped measurements take more time to run, but provide you with the highest measurement accuracy. Some measurement settings apply to both swept and stepped measurements, but some are set specific for a stepped or a swept measurement. In the following procedure, we will start with a swept measurement, then continue with the stepped measurement. The stepped measurement process will only call out the settings that are different for the two measurement types.

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.

## CAUTION

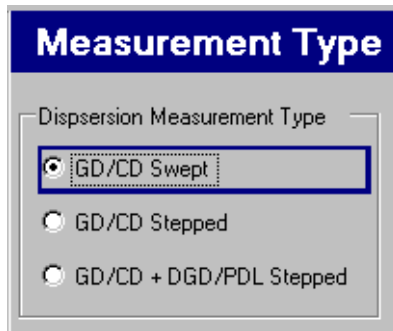
Before connecting any patchcords, you must clean the optical connections on the cable, device under test (DUT), and the ODA. Use cotton swabs and alcohol to clean the connectors, and then blow the connector surface dry with compressed air. For more information, refer to [“Cleaning Connections for Accurate Measurements”](#) on page 272.

---



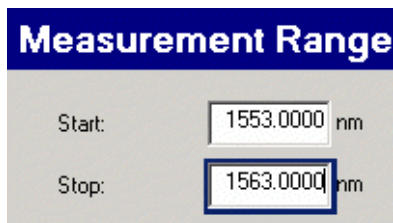
## Swept Measurement

- 9** On the analyzer press **Preset** to set the instrument to a known state.
- 10** From the Setup menu, press **CD/PMD** and select *GD/CD Swept*.



- 11** Tip: Settings in a panel are saved when another menu item or front panel key is selected. It is not necessary to press close on each setup panel.
- 12** From the Setup menu, press **Ports** and select *Port 1*.
- 13** From the Setup menu, press **Meas Range** and set the *Start* wavelength to 1553 nm and the *Stop* wavelength to 1563 nm.

This will set a 10 nm span around the center wavelength of the filter.



- 14** From the Setup menu, press **Sens/Res**. In the Sens/Res setup panel,
- a** Set Trace Points to *1601*
  - b** Set RF Mod Freq to *500 MHz*
  - c** Select an IF Bandwidth of *300 Hz*

Lower values of IF bandwidth reduce the noise but increase the measurement time.

**Sens / Res**

Resolution

Trace Points: 1601

Increment: 0.0063 nm

RF Mod Freq: 500.0 MHz

Auto Modulation Frequency

Sensitivity

IF Bandwidth: 300 Hz

Samples/Point: 201

Dynamic Drift Reduction

**15** Connect the normalization fiber from the test set Optical Output to Port 1.

**16** From the Measurement menu, press **Begin Norm**. From the Normalization Viewer,

- a select *Port 1* as the Active Port
- b select *Phase Dynamic Accuracy*

Phase Dynamic Accuracy corrects for amplitude-to-phase conversion effects in the network analyzer, in which the loss or gain changes with wavelength.

- c select *Normalize*

Normalization Options

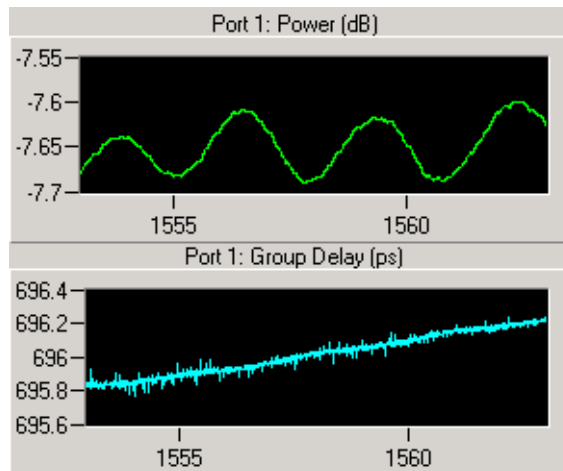
Phase Dynamic Accuracy (PDA) Correction

Port(s): Port 1

Clear... Abort Normalize Close

The normalization will take approximately 8 minutes. The majority of this time is used for the phase dynamic accuracy correction.

While normalization is running, data is plotted to the Normalization Viewer graphs. The completed normalization is shown below.



**17** When the normalization is complete, select **Close** on the Normalization Viewer.

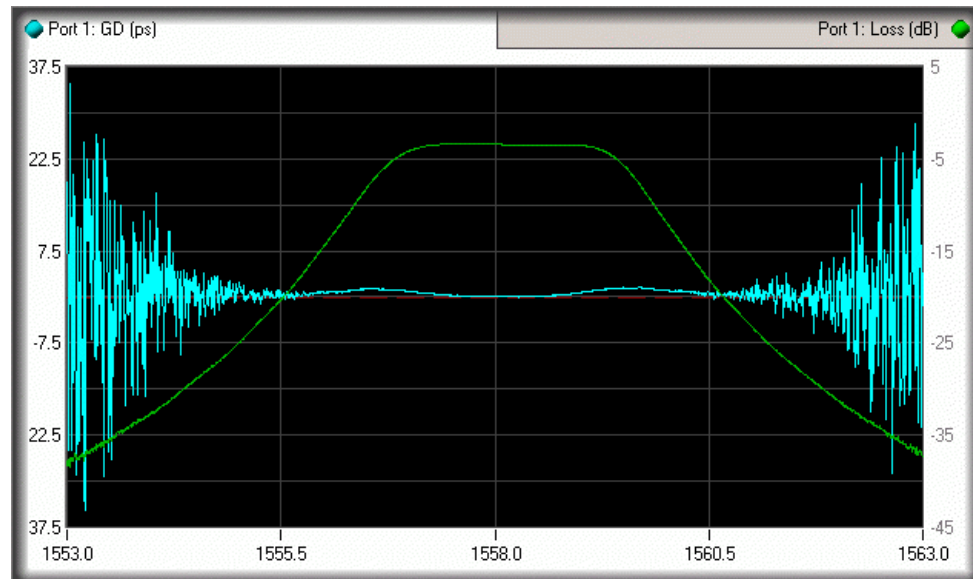
**Tip:** First-time users may want to verify the normalization process by performing a measurement of the normalization path. To do this, press **Begin Meas**. The Loss trace should be very close to 0 dB and the group delay variation should be small.

**18** Insert the filter into the normalized test path.

It is best practice to include the normalized fiber in the path with the test device, as its small dispersion contribution was captured in the normalization data.

**19** From the Measurement menu, press **Begin Meas** to start the measurement process.

When the measurement is complete, a trace is displayed for Port 1 GD (group delay), Port 1 Gain/Loss, and Port 1 CD (chromatic dispersion).



# Fiber Measurement

This measurement example uses a spool of fiber with a zero dispersion wavelength of approximately 1575 nm.

This measurement procedure will include both a swept and a stepped measurement. Stepped measurements take more time to run, but provide you with the highest measurement accuracy. Some measurement settings apply to both swept and stepped measurements, but some are set specific for a stepped or a swept measurement. In the following procedure, we will start with a swept measurement, then continue with the stepped measurement. The stepped measurement process will only call out the settings that are different for the two measurement types.

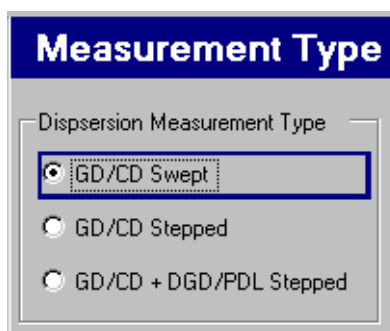
There are two different ways to access the instrument settings: from the front panel keys or from the menu bar. The front panel keys are separated into groupings by a group name which corresponds to the display menu bar names. For example, the Setup group of front panel keys has the exact same functions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

## CAUTION

Before connecting any patchcords, you must clean the optical connections on the cable, device under test (DUT), and the ODA. Use cotton swabs and alcohol to clean the connectors, and then blow the connector surface dry with compressed air. For more information, refer to “Cleaning Connections for Accurate Measurements” on page 272.

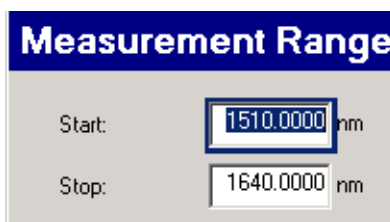
## Swept Measurement

- 1 On the analyzer press **Preset** to set the instrument to a known state.
- 2 From the Setup menu, press **CD/PMD** and select *GD/CD Swept*.



**Tip:** Settings in a panel are saved when another menu item or front panel key is selected. It is not necessary to press close on each setup panel.

- 3 From the Setup menu, press **Ports** and select *Port 1*.
- 4 From the Setup menu, press **Meas Range** and set the *Start* wavelength to 1510 nm and the *Stop* wavelength to 1640 nm.

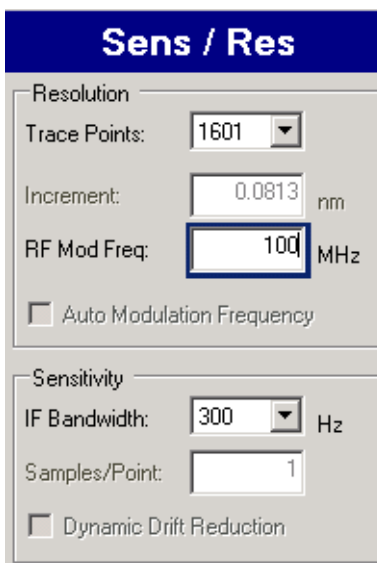


- 5 From the Setup menu, press **Sens/Res**. In the Sens/Res setup panel,
  - a Set Trace Points to *1601*
  - b Set RF Mod Freq to *100 MHz*

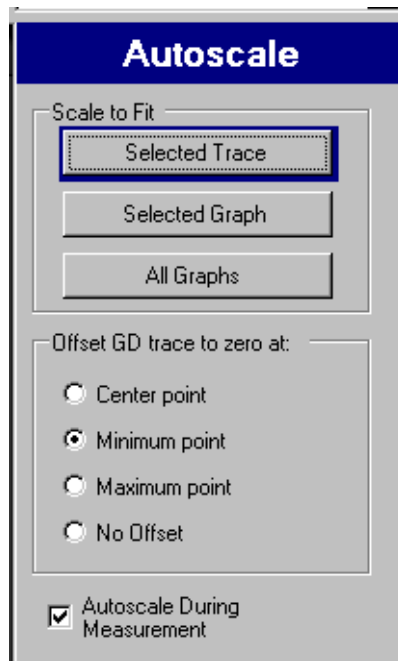
The Auto Mod Freq selection is not available for Swept measurements. Therefore, it is best to start with a low modulation frequency to avoid RF phase shift ambiguity in the case of a high dispersion fiber.

- c Select an IF Bandwidth of *300 Hz*

Lower values of IF bandwidth reduce the noise but increase the measurement time.



- 6 From the **Display** menu, press Auto Scale and set the Offset GD trace to zero at: option to *Minimum Point*.



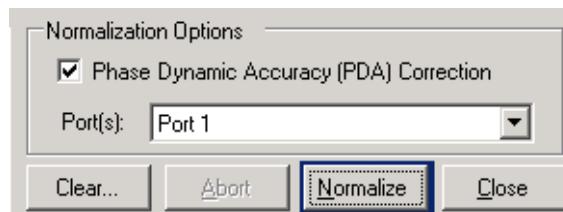
7 Connect the normalization fiber from the test set Optical Output to Port 1.

8 From the Measurement menu, press **Begin Norm**. From the Normalization Viewer,

- a select *Port 1* as the Active Port
- b select *Phase Dynamic Accuracy*

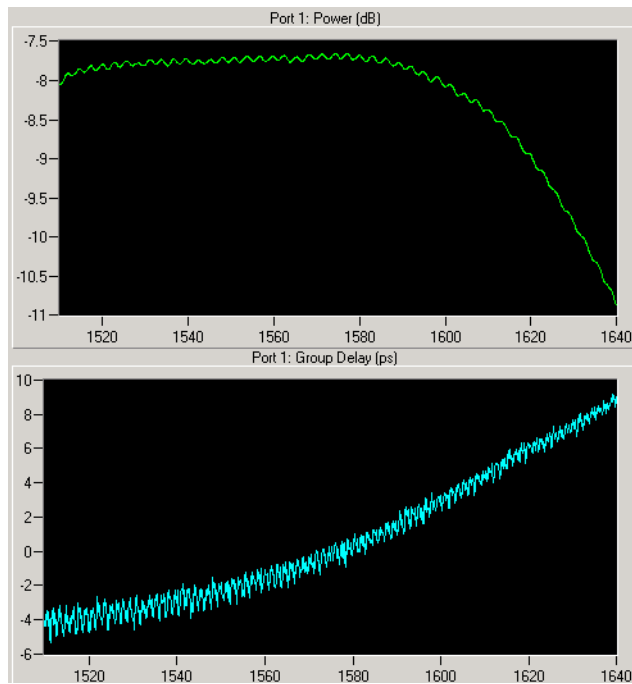
Phase Dynamic Accuracy corrects for amplitude-to-phase conversion effects in the network analyzer, in which the loss or gain changes with wavelength. Although fiber does not exhibit extreme loss changes over frequency range, a best practice is to correct for phase dynamic accuracy.

- c select *Normalize*



The normalization will take approximately 8 minutes. The majority of this time is used for the phase dynamic accuracy correction.

While normalization is running, data is plotted to the Normalization Viewer graphs. The completed normalization is shown below.



**9** When the normalization is complete, select *Close* on the Normalization Viewer.

**Tip:** First-time users may want to verify the normalization process by performing a measurement of the normalization path. To do this, press *Begin Meas*. The Loss trace should be very close to 0 dB and the group delay variation should be small.

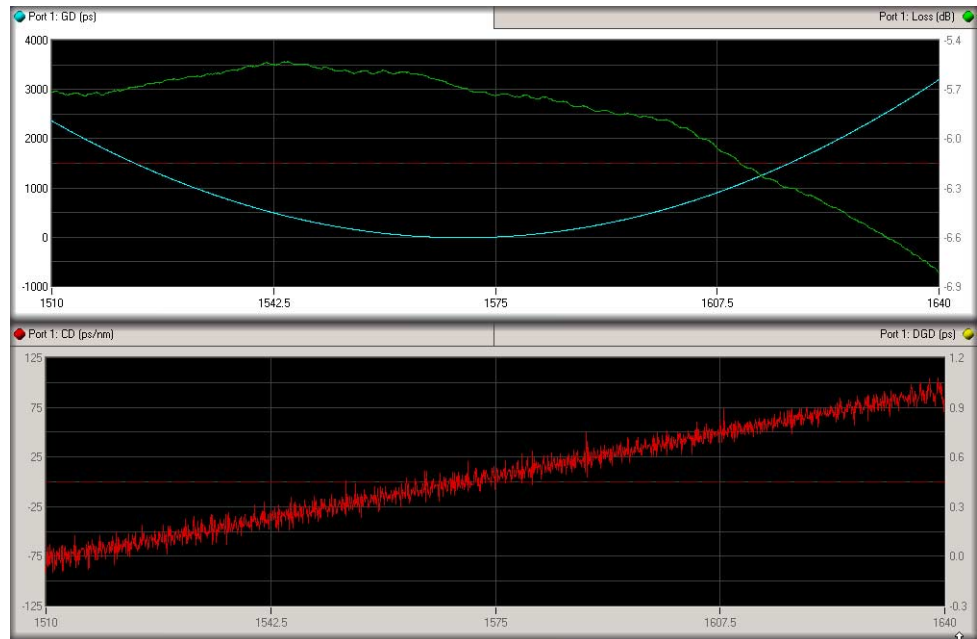
**10** Insert the filter into the normalized test path.

It is best practice to include the normalized fiber in the path with the test device, as its small dispersion contribution was captured in the normalization data.



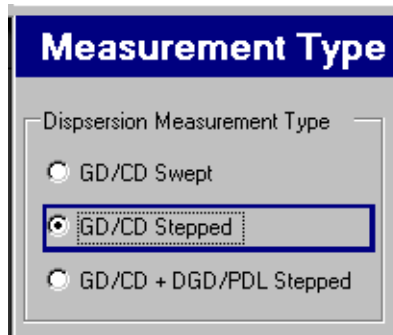
**11** From the Measurement menu, press **Begin Meas** to start the measurement process.

When the measurement is complete, a trace is displayed for Port 1 GD (group delay), Port 1 Loss, and Port 1 CD (chromatic dispersion).

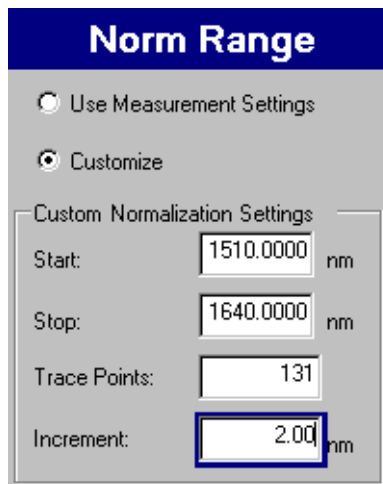


## Stepped Measurement

- 1 From the Setup menu, press **CD/PMD** and select *GD/CD Stepped*.

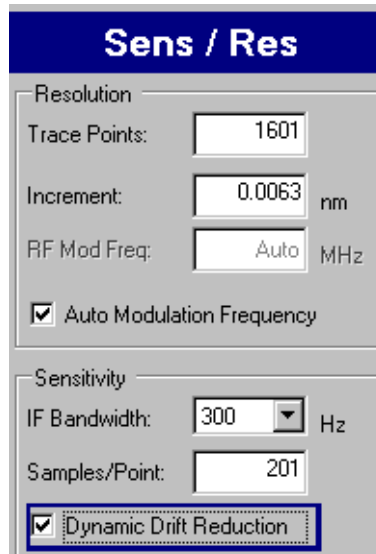


- 2 From the Setup menu, press **Norm Range**. In the Normalization setup panel,
  - a select *Customize*
  - b set the *Increment* to 2 nm

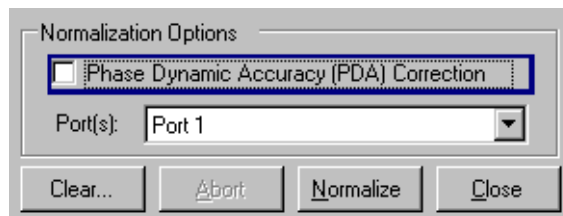


- 3 From the Setup menu, press **Sens/Res**. In the Sens/Res setup panel,
  - a Select *Auto Modulation Frequency*
  - b Select *Dynamic Drift Reduction*

Dynamic drift reduction should be used when the temperature is gradually changing. A best practice is to use dynamic drift reduction for all fiber measurements.



- 4 Connect the normalization fiber from the test set Optical Output to Port 1.
- 5 From the Measurement menu, press **Begin Norm**. From the Normalization Viewer,
  - a select *Port 1* as the Active Port
  - b select *Normalize*



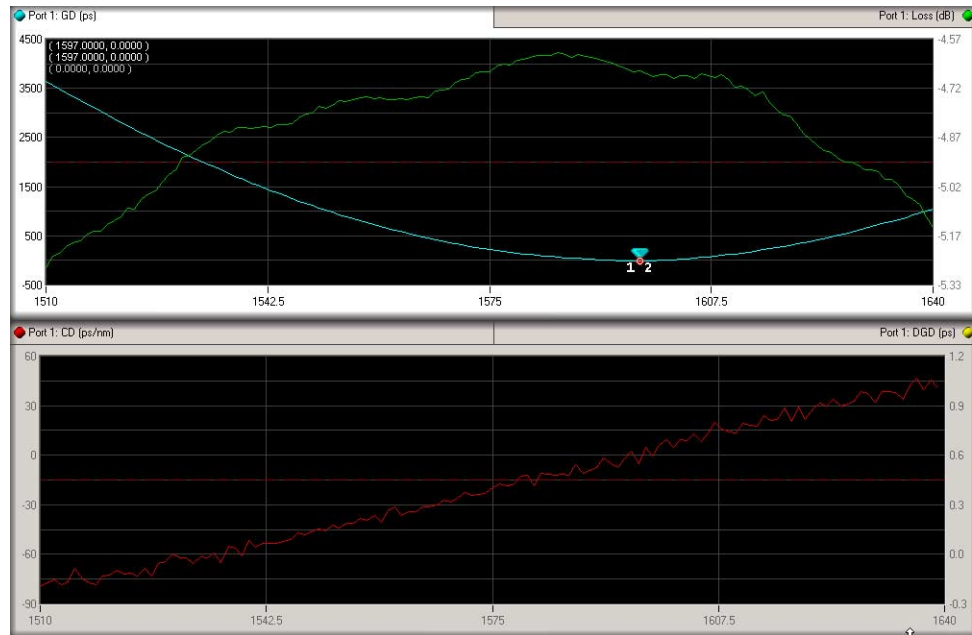
The normalization will take approximately 8 minutes. The majority of this time is used for the phase dynamic accuracy correction.

While normalization is running, data is plotted to the Normalization Viewer graphs. The completed normalization is shown below.

- 6 When the normalization is complete, select *Close* on the Normalization Viewer.
- 7 Insert the filter into the normalized test path.

**8** From the Measurement menu, press **Begin Meas** to start the measurement process.

When the measurement is complete, a trace is displayed for Port 1 GD (group delay), Port 1 Loss, and Port 1 CD (chromatic dispersion).

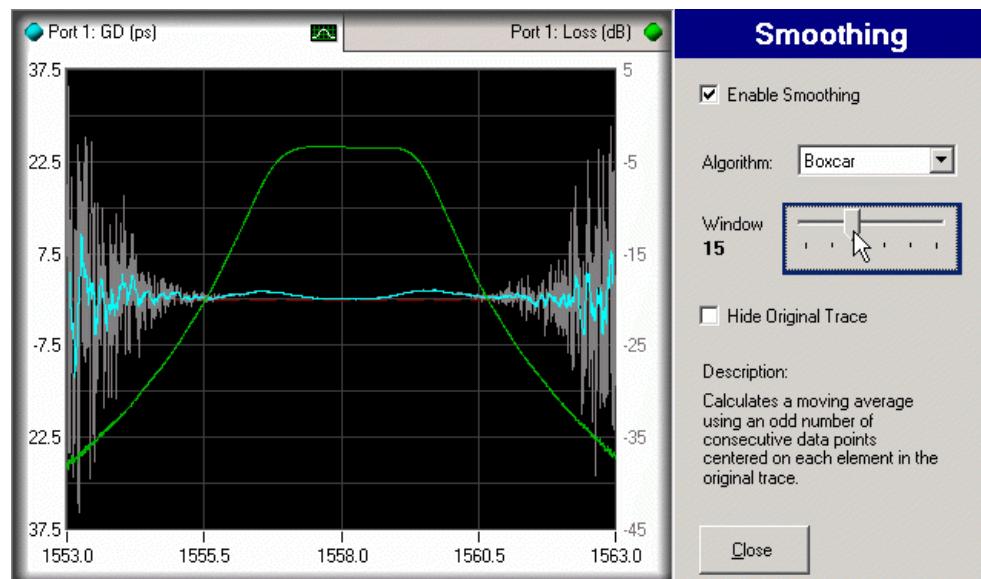


## Apply Smoothing to Remove Noise

Refer to “Smooth” on page 118 for more information.

- 1 Make the group delay trace the active trace.
- 2 From the Display menu, press Smooth. From the Smoothing setup panel,
  - a Select Enable Smoothing
  - b Select Boxcar for the algorithm
  - c Adjust the Window slide bar to optimize smoothing of the GD trace

You will want to add enough smoothing to remove the noise from the trace but not too much as to distort the response shape of the filter.







# 4

## Function Reference

Introduction .....	88
File Menu .....	89
View Menu .....	90
System Menu .....	91
Setup Menu .....	97
Measure Menu .....	107
Display Menu .....	111
Navigating the Front Panel .....	121
Toolbar Functions .....	122

# 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 59](#) 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<sup>®</sup>-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<sup>®</sup> 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.

---

## Export

Saves a file for use with external applications. There are four 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.

---

**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 ODA application. To log on to the ODA, you will need to enter the following user name and password.

User Name: ODA\_User

Password: agilentoda

## View Menu

The View menu is used to access the desktop screen.

---

**Minimize Application**

Minimizes the ODA 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 ODA from responding to front-panel key presses. Pressing Local will reestablish front-panel control.

## Preset

Cancels any current operations and sets the instrument to defined setup conditions (state).

Table 4 Default values

Function	Preset Value	Function	Preset Value
Dispersion Measurement Type	GD/CD Swept	X-Axis Scale, Minimum <sup>a</sup>	1493 nm
Active Ports	Port1	X-Axis Scale, Maximum <sup>a</sup>	1640 nm
Measurement Start Wavelength <sup>a</sup>	1493 nm	X-Axis Scale, Center <sup>a</sup>	1575 nm
Measurement Stop Wavelength <sup>a</sup>	1640 nm	X-Axis Scale, Span <sup>a</sup>	130 nm
Normalization Range	Use Measurement Settings	Y-Axis Scale, Reference Level <sup>b</sup>	0 ps
Resolution, Trace Points	1601	Y-Axis Scale, Units/Division <sup>b</sup>	1000 ps
Resolution, Increment <sup>a</sup>	0.08125 nm	Y-Axis Scale, Ref. Level Position	5
RF Modulation Frequency	2000 MHz	Auto Scale During Measurement	On
Auto Modulation Frequency	Off	Smoothing	Off
Sensitivity, IF Bandwidth	300 Hz	Curve Fit	Off
Sensitivity, Samples/Point	1	TLS Power	-5 dBm
Dynamic Drift Reduction	Off	Markers	Off

<sup>a</sup> for 81600B-160 TLS: depends on TLS type

<sup>b</sup> 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.

## 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.

Clear removes the messages from the status log.

Close closes the Status Log window.

---

## 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 ODA 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  $\pm 6^{\circ}\text{C}$ , 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 is too low. This is helpful for fiber manufactures but is not ideal for filter manufactures.

---

## Real-Time Port Monitor

Displays on a horizontally scrolling graph the real-time gain/loss and group delay measurements enabled for Port 1 and Port 2. The monitor operates in repeat sweep mode at the center wavelength of the measurement range and uses a 2 GHz Mod Frequency and 30 Hz IF Bandwidth.

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 group delay values approaching or below 50 fs. When a frequency selective test device is in the measurement path, the center wavelength should be set using Setup > Meas Range start and stop settings to a center wavelength that provides a low insertion loss point.

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.

For less demanding applications, such as  $\pm 250$  fs of RGD accuracy, usual good practice in making optical connections provides adequate results without using the Real-Time Port Monitor.

Clear Plots resets the measurements and clears the graph data.

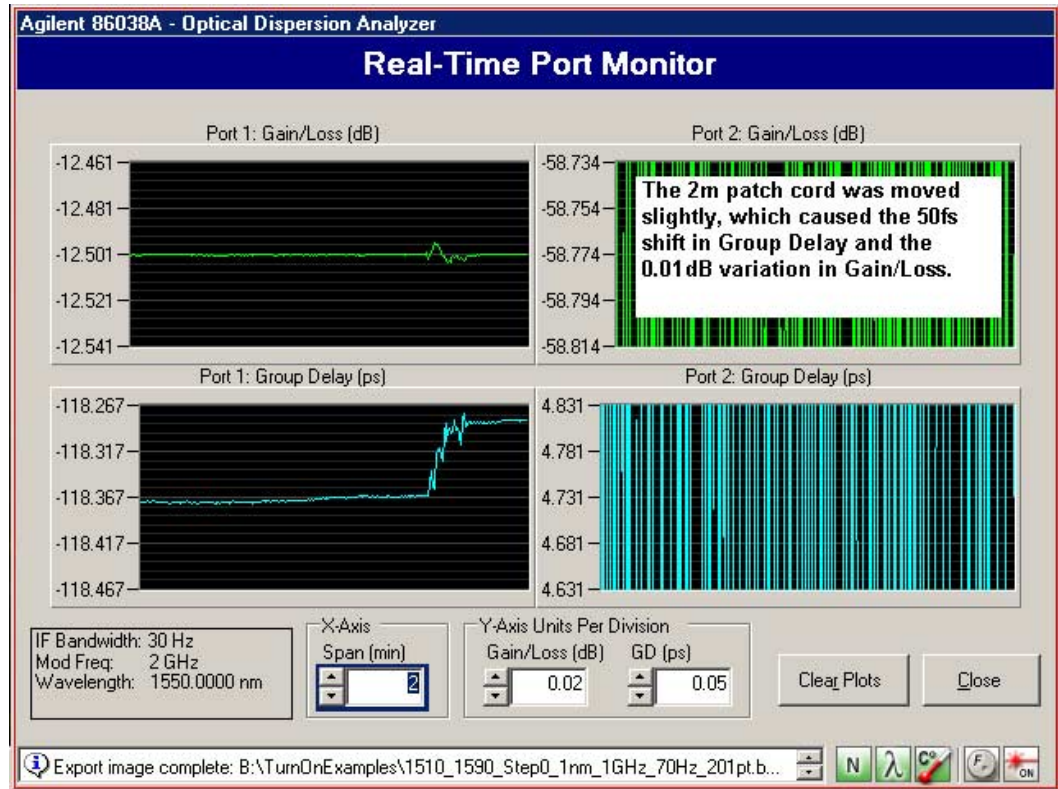


Figure 10 Example of Fiber Movement

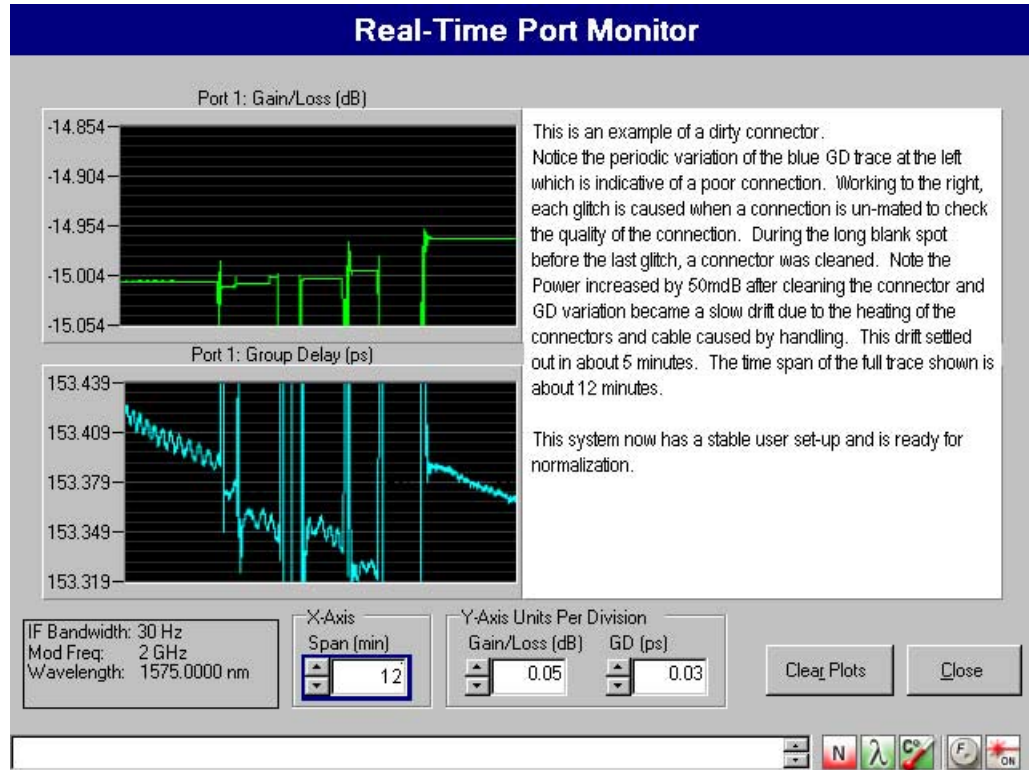


Figure 11 Example of a Dirty Connector



# 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 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. See [“Measurement Range”](#) and [“Sensitivity/Resolution”](#) on page 101.

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 increments. The density of samples is set by the combination of wavelength span (stop wavelength minus start wavelength) and the number of samples. Swept mode can dramatically reduce measurement time when the ultimate accuracy is not required. Swept mode cannot provide any polarization dependent measurements (that is, DGD or PDL).

### NOTE

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

### *Tips on Measurement Modes:*

- For a GD/CD +DGD/PDL stepped measurement the most accurate results are obtained with 70 Hz IF Bandwidth and 201 samples per point. A smaller IF Bandwidth will give PDL and DGD values that are too low. A higher IF Bandwidth will give more noise. Fewer samples will also give more noise. Refer to [“Sensitivity/Resolution”](#) on page 101 for more information on these settings.

- Step mode offers higher measurement accuracy and repeatability. Swept mode is faster
- In the swept mode, over-sampling the GD shape offers the possibility of using smoothing to reduce the noise. If sampling is marginal, then smoothing sacrifices information.
- Step mode can greatly improve signal to noise by allowing the averaging of multiple samples at each measurement wavelength. Swept mode is limited to a single sample at each wavelength.
- For more information on measurement types, [refer to “Swept and Stepped Measurement Modes” on page 57.](#)

---

## Length

### Length Mode

#### **Measure**

Calculates the length of the test device based on specialized group delay measurements and a user-entered refractive index value. Index values of 1 to 10 can be chosen. This feature is intended for fiber and cable test applications.

#### **Specify**

Specifies the length of the fiber-optic cable in kilometers. You can specify Port 1 and Port 2 Length which will also be used in computing CD and PMD coefficients.

---

## Ports

Specifies the input receiver ports used for measurement: Port 1, Port 2 or Both Ports.

---

## Measurement Range

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

**Tip:** For best curve fitting accuracy, 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, for example, in the 1300 nm window.

**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 107](#).

---

## Sensitivity/Resolution

### Resolution Area

#### Trace Points

##### *For Stepped measurement modes*

Sets the number of measurement wavelengths including the start and stop wavelengths. The greater the number of trace points the greater the measurement accuracy. However, the measurement time also increases.

##### *For Swept measurement mode*

Sets the number of trace points to either 801 or 1601.

#### Increment

Specifies the separation between adjacent measurement wavelengths. In stepped measurements, it specifies the TLS wavelength step. In swept measurements, it represents (wavelength range)/ (number of trace points minus one).

**Tip:** Select a wavelength increment just small enough to reveal the shape of the group delay curve.

#### RF Mod Freq

Specifies the RF modulation frequency used to modulate the optical signal in the ODA test set.

In general, a high modulation frequency minimizes instrumentation noise and a low modulation frequency maximizes wavelength resolution.

##### *Tips on the Modulation Frequency Selection*

- If selecting modulation frequency manually, use the selection charts 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 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.
- 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.

- 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 108](#).
- A good rule of thumb is to use a modulation frequency that is less than 1/20 of the free spectral range of the ripple effect. For example, a 500 MHz modulation frequency could be used to explore ripple with 10 GHz FSR (or about an 80 pm period) with an attenuation of about 5% of the ripple peak-to-peak value.

### 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 (←) indicate that you must decrease the wavelength increment to achieve the desired  $\Delta\tau_{RGD}$ .

Table 5 Maximum RF Modulation Frequency (MHz) for  $<90^\circ$  phase steps  $f_{90} = 1/(4 \times \Delta\tau \times \Delta\lambda)$

$\Delta\tau_{RGD}$ (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	←
100	2500	500	250	50	25	5	←	←
500	500	100	50	10	5	←	←	←
1000	250	50	25	5	←	←	←	←
5000	50	10	5	←	←	←	←	←
10000	25	5	←	←	←	←	←	←
50000	5	←	←	←	←	←	←	←

### Auto Modulation Frequency

#### *For Stepped mode measurements only!*

Automatically selects the modulation frequency to avoid RF phase wrapping. The amount of RF phase change that is generated in response to a wavelength increment is determined by the product of the wavelength increment, the modulation frequency, and the dispersion of the test device. Automatic adjustment of the modulation frequency ensures that the per-wavelength-step phase change is always less than  $\pm 180$  degrees on port 1 only.

**Tip:** Use auto mode if the approximate group delay/dispersion characteristics are not known in advance.

**Tip:** When using both ports and auto modulation frequency, connect the device with the largest dispersion to port 1.

## Sensitivity Area

### **IF Bandwidth**

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. 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.
- When making PDL or DGD measurements, the polarization scanner runs at a fixed speed and produces PDL or DGD changes with the majority of spectral content around 8 Hz. This limits the minimum IF bandwidth when scanning to 70 Hz, which avoids attenuating the 8 Hz components. Using a wider IF bandwidth allows a faster measurement, but raises the noise in the measurement and may slightly bias DGD and PLDL toward larger values.
- During PDA normalization, 70 Hz IF bandwidth is used to characterize noise components with the scanner off, for both GD and Magnitude. These noise measurements provide the best correction to DGD and PDL when measuring 70 Hz IF bandwidth. For more information on PDA, refer to [“Phase Dynamic Accuracy”](#) on page 108.

### **Samples/Point**

Determines the number of amplitude and phase measurements (samples) that are taken at each measurement wavelength (trace point).

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. Large number of points improves the polarization mode coverage to provide more accurate DGD and PDL. 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. Ex-



- pect to trade-off measurement speed.
- 201 samples per point is a good compromise of polarization scan coverage, collecting enough data to average out noise, and keeping the measurement time short enough to avoid drift effects. In many applications, 401 samples may not improve the measurement accuracy due to drift effects.

## Dynamic Drift Reduction

### *For stepped mode measurements only*

This system function reduces measurement error contributed by gradual, continuous temperature changes in the test path 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.

---

## Laser Setup

### TLS Power

This control sets the TLS output power. The default value of  $-5$  dBm is available from the 81600B TLS over the full wavelength range. For the 81600B-160 TLS a power level of  $+5$  dBm is available from 1520 nm to 1610 nm. 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 phase dynamic accuracy (PDA) effects are more pronounced. After changing the TLS power, normalization (without PDA) is required.

The TLS power level can be lowered to eliminate artifacts of Gain Stage Switching (possibly 200 fs, typically 50 fs, discontinuities in group delay). These may occur when the power level at the receiver is comparable to a 0 dB insertion loss DUT with about  $-9$  dBm TLS power, and again at a power level 9 dB higher. Gain Stage switching only occurs at these two power levels. Port 1 and Port 2 may switch at slightly different levels. Alternatively, the TLS power can be increased to move the artifacts away from the region of interest. [Refer to "If you see errors in Group Delay measurements but no error is visible in Gain/Loss" on page 268](#) for an example.

If the requested power is larger than available, no indication is given at the ODA user interface. While sweeping, the excess power condition is announced at the upper left of the TLS display with the flag: 'Exp'. This can be viewed by removing the TLS cover and pressing the "local" softkey on the TLS.

There is slightly more TLS power available in step mode than in swept mode. Refer to the TLS specifications for maximum power level in various spans.

### Settle TLS Now

Initiates the TLS settle (Lambda Zero) process on the tunable laser source. The ODA 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 realignment. A status message, Estimated Time Until Next TLS Settling, indicates the time before a TLS Settling is required. Under normal temperature conditions, the status message may show 120 minutes as the estimate. This status message is updated every minute.

A TLS settle can be performed in two ways:

- When the estimated time turns from a green to yellow background, then initiating a Settle TLS Now will start the TLS settle process. Note that a TLS Settling can be initiated by the user at anytime.
- When the estimated time turns from a yellow to red background, a TLS settle will automatically be performed in 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 settle 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 function initiates one sweep of the measurement range.

### Repeat Sweep

Continuously sweeps until the measurement is canceled or the instrument is set to Single Sweep.

**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.

---

### Begin Normalization

Accesses the Normalization setup panel. Normalization is used to measure the loss and group delay of the test path. Later this data is subtracted from the DUT measurement to generate measurement results. [Refer to "Normalization Range" on page 100](#) to specify custom normalization settings.

A normalization can optionally run the following tests:

- Length

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 98](#) to setup a length measurement.

- Wavelength Correction

Using the optional 86122A multi-wavelength meter, wavelength correction is performed for stepped normalizations. That is, when Setup > CD/PMD > is set to one of the two stepped measurement modes. Wavelength correction is always performed if an 86122A is connected. Five wavelength points are measured and the average offset is applied.

- Optimize Optical Modulator Bias routine

If Phase Dynamic Accuracy is selected, normalization also runs an Optimize Optical Modulator Bias routine.

Once the normalization has begun, a “Measurement in Progress” message appears at the bottom of the display and the status bar will show the estimated time of each test process. For more information on the normalization process, refer to [“Normalization and Test” on page 55](#).

### Phase Dynamic Accuracy

Corrects for amplitude-to-phase conversion effects in the network analyzer. If your test device exhibits no loss variation across the wavelength range, then clear this selection to speed up the normalization time.

#### *Tips on Phase Dynamic Accuracy Correction*

- PDA calibration is an optional normalization step that reduces the amplitude to phase conversion in the network analyzer. 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.
- 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 some advantage in attenuating the power level seen at the receiver. The region of 10 to 25 dB of DUT insertion loss has a smaller PDA effect, so for experiments changing the modulation frequency without recalling the proper normalization, this region might simulate 86037C behavior. There may be cases where the accuracy needed for the measurement does not require PDA corrections, in which case they would be optional. But, turning off the PDA corrections would seldom improve the accuracy of the measurement. For example, one of the higher slope regions of PDA at 2 GHz modulation frequency can cause GD errors of 380 fs/dB when PDA corrections are off. Similarly, PDA at 10 MHz modulation frequency can cause GD errors of 18 ps/dB when PDA corrections are off.

### Ports

**Port 1** measures Port 1 only.

**Port 2** measures Port 2 only.

**All Ports Simultaneously** measures Port 1 and Port 2 at the same time.

**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** aborts the normalization process.

**Normalize** starts the normalization process.

**Close** exits the Normalization Viewer.

Table 6

Operation	Optimize Optical Modulator	Gain/GD Normalization	PDA Normalization	Length Normalization	WL Accuracy Normalization
Install new SW version	−3.0 V*	blank*	blank*	blank*	blank*
Cycle power/load ODA application	last used +	last used*	last user*	last used +	last used +
Preset	last used	last used	last used	last used	last used
Recall State	recalled	recalled	recalled	recalled	recalled
Normalize with PDA, length, 86122A	measured	measured	measured	measured	measured
Normalize without PDA, with length and 86122A	last used	measured	last used +	measured	measured
Normalize without length, with PDA and 86122A	measured	measured	measured	last used	measured
Normalize without 86122A, with PDA and length	measured	measured	measured	measured	last used
Change number of ports	last used	last used	last used	last used	last used
Change swept, stepped, scanned	last used	last used +	last used	last used	last used
Change wavelength range	last used	last + extended ends*	last used +	last used +	last used +
Change RF Mod. Freq.	last used +	last used *	last used *	last used	last used
Change IF bandwidth	last used	last used +	last used	last used	last used
TLS Settling	last used	last used +	last used	last used	last used +

## Operations Affecting Normalization

Below is a table of operations affecting the ODA normalization state. The indications of “none”, “+”, a minor source of measurement error, or “\*”, a significant source of measurement error, for the operation being changed are only a rough guide. In particular, when using preset, the assumption is that other items on the list are not being changed, or will be set back to the state used for normalization. Similarly, when changing number of ports, it is assumed the change is to a configuration with a valid normalization.

---

### 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.

---

### Optimize Optical Modulator Bias

Performs an auto bias on the modulator in the test set. A mod bias is also performed each time Phase Dynamic Accuracy is run during a Normalization. For more information on the modulator bias routine, refer to [“Begin Normalization” on page 107](#) and to [“Optical Modulator Bias Adjust” on page 55](#).

---

### Cancel Operation

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 Meas 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 Meas 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 Meas 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 Meas Range setup panel.

### View Full Width

Sets the graph's start and stop wavelength range to the same values specified for the Meas Range setup panel.

### 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 units 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.

### Units/Division

Specifies the units per division of the vertical scale. 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.

### Ref. Level Position

Determines the position of the reference level on the display graticule.

### 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 units are important.

---

## Auto Scale

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

### 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 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 as specified in the Sens/Res setup panel.

---

## 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 120](#) 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

Allows you to move 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 on screen responses are accepted as discrete spectral responses. To be accepted, each trace peak/pit must rise, and then fall by at least the excursion value about a given spectral component. Setting the value too high will result in failure to include the smaller responses. Setting the value too low will cause all spectral components to be accepted.

### **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 currently active marker to the center of the viewable range.

### To Reference Position

Moves the currently active marker to the reference level position on the active trace.

### Zero GD Trace (*Applies only to GD Trace*)

Group delay is a relative measurement. Use this function to offset the GD trace at the selected marker to a y-value of zero. Also see "Offset GD trace to zero at:" on page 113.

### 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 will show you how to assign the CD trace and the DGD trace for Port 2.

- 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 2* 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 2* and *DGD* as the measurement type.
- 7 Press *Close* to save the trace definitions.

For each trace you can specify either Port 1 or Port 2 measurement traces and one of five measurement types.

- GD (Group Delay)
- CD (Chromatic Dispersion)
- DGD (Differential Group Delay)
- PDL (Polarization Dependant Loss)
- Gain Loss

## GD Options Area

### Show DGD Contribution

Displays error bars on the group delay trace to show how much the GD varied do to polarization scanning. This option only applies when DGD data exists. See “CD/PMD”.

### 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.

## CD Options Area

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.

---

## Smooth

Smoothing controls (averages) the noise without losing too much detail. The smoothed trace takes over the current trace color and the unsmoothed (raw) trace is preserved 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 is applied forward and backward to eliminate a bias. The traces obtained from each pass are averaged to obtain the result.

**Gaussian** takes a value for sigma (standard deviation) and applies a point centered gaussian smooth.

**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 the 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.
- Refer to “Apply Smoothing to Remove Noise” on page 85 for more information.

**Window**

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

The precision of 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  $\lambda_0$ ,  $S_0$  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.

### Enable Curve Fit

When selected, applies the specified curve fitting algorithm to the trace.

### Algorithm

**Linear** calculates single parameter least squares fit. The equation is the form:  $y = mx + b$ . It is commonly used for dispersion-shifted fiber, in which waveguide dispersion is dominant.

**Quadratic** curve is a least squares fit to the equation  $y = Ax^2 + Bx + C$ . This is a good approximation to any parabolic shape.

**3rd Order Sellmeier** curve fit is commonly used for dispersion-unshifted fiber, in which material dispersion is dominant.

**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.

### Equation Area

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

### 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 allow you to view both port 1 and port 2 measurement traces at the same time. Refer to “Define Trace” on page 117 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 with select all traces 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 as specified in the Sens/Res setup panel.

### 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



Reduces the wavelength span on the active graph. Therefore, the zoomed-in 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



Reduces 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, mod bias, 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.

# 5

## Remote Operation

Overview .....	126
Recommended Control Sequence .....	130
How to Configure the ODA for Networking .....	130
How to Connect the ODA to Your Network .....	130
How to Change Network Settings .....	131
Install the Remote Client .....	133
Program Structure: events vs. sequential .....	138
Front Panel Functions with No Remote Equivalent .....	140
Enumerations .....	141
Event Table .....	144
Troubleshooting .....	146
Front Panel Maps .....	147
Example 1: Simple Application for Controlling the ODA .....	154
Example 2: Control from Excel .....	183
Example 3: Control from LabView .....	185
Example 4: Complete application - RepeatMeas .....	186

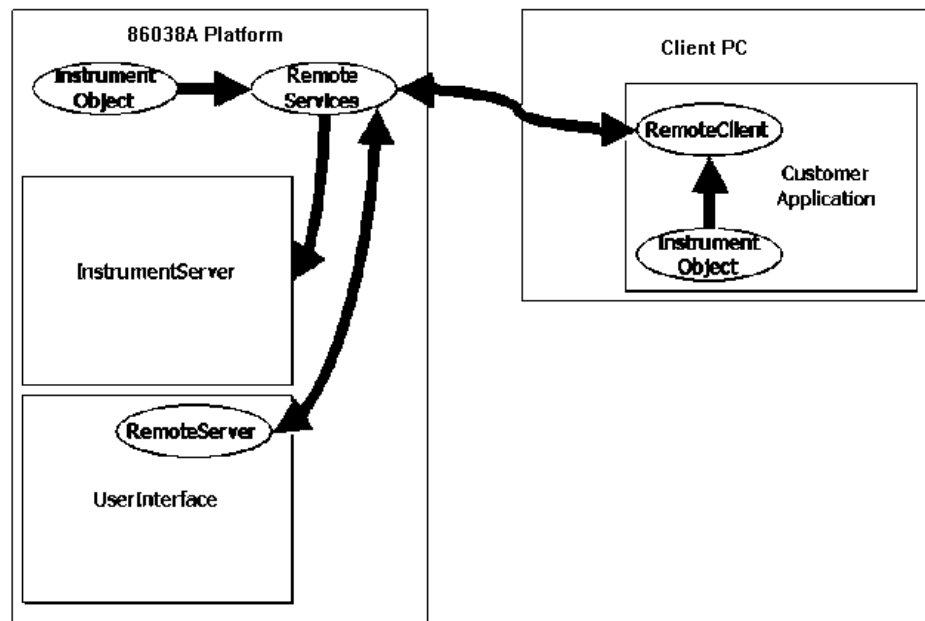
# Overview

This chapter will help you control an ODA from your own computer. The chapter covers how to write your own applications, and how to control the ODA from existing applications such as Excel and LabView.

The ODA 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 ODA from many established applications such as Visual Basic 6.0 and VBA, as well as from .NET enabled applications such as C#.

The ODA 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 ODA, 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 ODA user interface as a Serve 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 ODA and retrieve measurement data.



This section assumes that you are familiar with the front panel operation of the ODA. 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 ODA. These examples start from a blank project allowing you to see all the steps required.

Agilent Technologies does not recommend controlling the ODA 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 ODA 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 179, then review the “Recommended Control Sequence” on page 130. You will find the “Front Panel Maps” on page 147 very useful for identifying specific commands.

Note that “[Example 1: Simple Application for Controlling the ODA](#)” 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 ODA.

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 ODA from LabView.

Example 4 is a complete application for taking ODA 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 ODA 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 ODA. 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 ODA” on page 154 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 ODA system.

## How to Configure the ODA for Networking

Remote programming of the ODA is only possible if the ODA is connected to a local area network (LAN) via the built-in LAN connector. When the ODA 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 ODA to Your Network

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

**NOTE**

Do not connect the ODA to a network that is configured to automatically install software on network devices. Installing or overwriting files on the ODA 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 ODA LAN connector supports 10 Base-T and 100 Base-T Ethernet networks using TCP/IP and other Microsoft supported networking protocols. The ODA uses Microsoft® Windows 2000.

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 ODA 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 2000 (Start > Help).

### NOTE

By default, as the instrument starts up, you are logged on as an administrator with the logon name ODA\_User. Agilent only recommends using the ODA application while you are logged on as an administrator.

You can change network settings by using the standard Microsoft® Windows 2000 functions. To access these functions, you must exit the ODA application. You can do this by clicking Exit on the File menu. Enter the password “agilentoda,” and then click Yes.

### *To view or change the computer machine name*

### NOTE

If you only wish to *view* the machine name, you can do this while the ODA 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. Enter the password “agilentoda,” 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 Network Identification 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 ODA to your network.

---

- 1** On the File menu, click Exit. Enter the password "agilentoda," and then click Yes.
- 2** Click Start, point to Settings, and then click Network and Dial-up Connections.
- 3** Right-click the network connection that you want to configure, and then click Properties.
- 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

- 1 Install the .NET framework from Microsoft. Go to [www.microsoft.com](http://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 86038A CD shipped with the ODA contains the Remote Client Installation to install the ODA specific DLL's and the programming examples. Insert the CD into the CD drive, use Windows Explorer to find ODA Remote Client Installer Folder.

### 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 odaClient

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

To provide better compatibility, the ODA 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:

- 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.

- Declare the object.

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

```
Private WithEvents odaClient As RemoteClient.Communicator
```

- Instantiate the object.

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

```
Set odaClient = New RemoteClient.Communicator
```

#### 4 Establish Connection to the ODA

The first thing you have to do with your `odaClient` is to establish a connection to the particular ODA you want to talk to. You need to know the network name of the ODA. Once the ODA 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 odaClient.Connectivity.Connect(<insert oda name>)
```

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

```
odaClient.Connectivity.IsConnected.
```

For every call to `Connect()` there has to be a call to `Disconnect()` because the ODA 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.

```
(odaClient.Connectivity.TerminateSession)
```

#### 5 Set Measurement Type, Range and Port(s)

Next, the ODA 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:

```
odaClient.MeasurementRange.XStart = 1535  
odaClient.MeasurementRange.XStop = 1545  
odaClient.DispersionMode = eDispersionMode_CD_Swept  
odaClient.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 number of trace points:

```
odaClient.Resolution.Swept.TracePoints = eTracePointsSweptMode_Step801
```

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

Table 7 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, and Polarization Dependant Loss with the tunable laser stepping across the specified wavelengths.

If a stepped mode has been chosen, you may want to specify the wavelength step size. This indirectly sets the number of points in the trace:

```
odaClient.Resolution.Stepped.Increment = 0.02
```

The above example sets a step of 0.02 nm. This would set 501 trace points for the requested 10 nm sweep (1535 to 1545 nm).

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

```
ePortsEnabled_Port_1_Enabled
ePortsEnabled_Port_2_Enabled
ePortsEnabled_All_Ports_Enabled
```

## 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 138](#) for a more detail discussion about how to synchronize your program with the ODA.

```
'Use measurement range and port for normalization
odaClient.NormalizationRange.UseCustomNormalization = False
'Have user make connection
MsgBox ("Connect ODA for normalization")
'Start the normalization
odaClient.Actions.Normalize (False)
' 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 odaClient.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 ODA 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. If your test device exhibits no loss variation across the wavelength range, then do not run PDA, as it adds approximately 7 minutes to the normalization time. The command to enable it:

```
odaClient.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 ODA for measurement")
' Start the sweep
odaClient.Actions.Measure
' Let sweep get started before polling
Sleep 1000
Dim elapsedTime As Double
elapsedTime = 0
' Poll the ODA 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 odaClient.Status.TriggerComplete = True
```

## 8 Transfer Data

Now that there is valid data in the ODA, 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 ODA 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 is the only measurement with offset wavelengths.

This example queries the gain:

```
Dim gainData() As Double
' Get the data and figure out the wavelength axis
gainData = odaClient.Results.YData(eMeasurementType_Gain,
eODAPort_One)
npoin = UBound(gainData) - LBound(gainData)+1
xStart = odaClient.Results.XStart(eMeasurementType_Gain)
xStop = odaClient.Results.XStop(eMeasurementType_Gain)
xStep = (xStop - xStart) / (npoin - 1)
```

See Table 8, “ODA Enumerations,” on page 141 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 ODA

The ODA keeps track of who is connected to it. If one of the connections quits responding, the ODA 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:

```
odaClient.Connectivity.Disconnect
```

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

```
odaClient.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 ODA 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 multi task.

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

```
writeOda(oda, "STAR 1520 nm")
writeOda(oda, "STOP 1535 nm")
writeOda(oda, "INIT:IMM; *OPC?")
value = readOda(oda) // 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 ODA, you send the commands and then wait for the ODA to indicate that it has completed the sweep:

```
odaClient.MeasurementRange.XStart = 1520
odaClient.MeasurementRange.XStop = 1535
odaClient.Actions.Measure
sleep(1000)
loop
  sleep(100)
  elapsedTime += 100
while elapsedTime < MAX_TIME and odaClient.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 ODA. 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 ODA (which is a good idea) or you can have a similar wait loop. To check for connectivity:

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

To open the session and wait for a valid connection:

```
odaClient.Connectivity.Connect()
sleep(5000) \\ wait 5 seconds
loop
  sleep(100)
  elapsedTime += 100
while elapsedTime < MAX_TIME and odaClient.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 ODA will involve enumerated types - where there is a finite number of possible values, each with a unique spelling. Because the ODA 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 8 ODA Enumerations

Enumeration	Values in VB 6	Values in .NET
eDispersionMode	eDispersionMode_CD_PMD_Stepped	eDispersionMode.CD_PMD_Stepped
odaClient.DispersionMode	eDispersionMode_CD_Stepped	eDispersionMode.CD_Stepped
	eDispersionMode_CD_Swept	eDispersionMode.CD_Swept
eEventLogType	eEventLogType_EventError	eEventLogType.EventError
Event: NewStatus	eEventLogType_Information	eEventLogType.Information
	eEventLogType_Warning	eEventLogType.Warning
		ODASensitivity.eGDOffsetMode: <sup>1</sup>
eGDOffsetMode	eGDOffsetMode_CenterPoint	.CenterPoint
odaClient.Display.GDOffsetMode	eGDOffsetMode_MaximumPoint	.MaximumPoint
	eGDOffsetMode_MinimumPoint	.MinimumPoint
	eGDOffsetMode_No_Offset	.No_Offset

Table 8 ODA Enumerations

Enumeration	Values in VB 6	Values in .NET
eIFBandwidth odaClient.Sensitivity.IFBandwidth	eIFBandwidth_IFBW_30Hz	ODASensitivity.eIFBandwidth: <sup>1</sup> .IFBW_30Hz
	eIFBandwidth_IFBW_70Hz	.IFBW_70Hz
	eIFBandwidth_IFBW_100Hz	.IFBW_100Hz
	eIFBandwidth_IFBW_200Hz	.IFBW_200Hz
	eIFBandwidth_IFBW_300Hz	.IFBW_300Hz
	eIFBandwidth_IFBW_1000Hz	.IFBW_1000Hz
	eIFBandwidth_IFBW_2000Hz	.IFBW_2000Hz
	eIFBandwidth_IFBW_40000Hz	.IFBW_40000Hz
eInstrumentStatus odaClient.Status.InstrumentState	eInstrumentStatus_Aborting	eInstrumentStatus.Aborting
	eInstrumentStatus_Idle	eInstrumentStatus.Idle
	eInstrumentStatus_Initializing	eInstrumentStatus.Initializing
	eInstrumentStatus_TLSSettling	eInstrumentStatus.TLSSettling
	eInstrumentStatus_Measuring	eInstrumentStatus.Measuring
eLayoutMode odaClient.Display.ToggleMode	eLayoutMode_QUAD_GRAPH	eLayoutMode.QUAD_GRAPH
	eLayoutMode_DUAL_GRAPH	eLayoutMode.DUAL_GRAPH
	eLayoutMode_SINGLE_GRAPH	eLayoutMode.SINGLE_GRAPH
eLengthMode odaClient.Length.Mode	eLengthMode_Measure	ODALength.eLengthMode: <sup>1</sup> .Measure
	eLengthMode_Specify	.Specify
eMeasNormDataType odaClient.Results.FiberLength	eMeasNormDataType_Measurement	eMeasNormDataType.Measurement
	eMeasNormDataType_Normalization	eMeasNormDataType.Normalization
eMeasurementType Used in defining traces and reading data back.	eMeasurementType_CD	eMeasurementType.CD
	eMeasurementType_Gain	eMeasurementType.Gain
	eMeasurementType_GD	eMeasurementType.GD
	eMeasurementType_Length	eMeasurementType.Length
	eMeasurementType_None	eMeasurementType.None
	eMeasurementType_PDL	eMeasurementType.PDL
	eMeasurementType_PMD	eMeasurementType.PMD
enumAcquisitionMode event: NewDataUpdate	enumAcquisitionMode_eDriftMonitor	enumAcquisitionMode.eDriftMonitor
	enumAcquisitionMode_eLength	enumAcquisitionMode.eLength
	enumAcquisitionMode_eMeasurement	enumAcquisitionMode.eMeasurement
	enumAcquisitionMode_eModBias	enumAcquisitionMode.eModBias
	enumAcquisitionMode_eNormalization	enumAcquisitionMode.eNormalization

Table 8 ODA Enumerations

Enumeration	Values in VB 6	Values in .NET
enumLaserMode event: NewDataUpdate	enumLaserMode_eStepped enumLaserMode_eSwept	enumLaserMode.eStepped enumLaserMode.eSwept
enumOnOff Used many places.	enumOnOff_eOff enumOnOff_eOn	enumOnOff.eOff enumOnOff.eOn
enumStatus Used to indicate progress, such as trigger or sweep progress.	enumStatus_ABORTED enumStatus_COMPLETE enumStatus_START	enumStatus.ABORTED enumStatus.COMPLETE enumStatus.START
enumTestType	enumTestType_DriftMonitor enumTestType_GD_CD_PMD enumTestType_GD_CD_Stepped enumTestType_GD_CD_SWEPT enumTestType_Length enumTestType_ModBias enumTestType_PDA enumTestType_WavelengthCorrection	enumTestType.DriftMonitor enumTestType.GD_CD_PMD enumTestType.GD_CD_Stepped enumTestType.GD_CD_SWEPT enumTestType.Length enumTestType.ModBias enumTestType.PDA enumTestType.WavelengthCorrection
enumTLSSettlingProbability Event: TLSSettlingProbability	enumTLSSettlingProbability_eHigh enumTLSSettlingProbability_eLow enumTLSSettlingProbability_eMedium	enumTLSSettlingProbability.eHigh enumTLSSettlingProbability.eLow enumTLSSettlingProbability.eMedium
eODAPort Used several places to refer to ports.	eODAPort_One eODAPort_Two	eODAPort.One eODAPort.Two
ePortsEnabled odaClient.Ports	ePortsEnabled_All_Ports_Enabled ePortsEnabled_Port_1_Enabled ePortsEnabled_Port_2_Enabled	ePortsEnabled.All_Ports_Enabled ePortsEnabled.Port_1_Enabled ePortsEnabled.Port_2_Enabled
eTraceLetter odaClient.Display.DefineTrace	eTraceLetter_TraceA eTraceLetter_TraceB	eTraceLetter.TraceA eTraceLetter.TraceB
eTracePointsSweptMode For normalization and measurement trace point setting.	eTracePointsSweptMode_Step1601 eTracePointsSweptMode_Step801	ODAResolution.eTracePointsSweptMode: <sup>1</sup> .Step1601 .Step801
eTriggerType odaClient.TriggerModeParams	eTriggerType_TriggerContinuous eTriggerType_TriggerSingle	eTriggerType.TriggerContinuous eTriggerType.TriggerSingle

<sup>1</sup> This command comes off the root:  
Agilent.LWD.Ag86038x.InstrumentObjects.

## Event Table

As mentioned in other sections, the ODA 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 `odaClient`, 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 9 ODA Events

Event	Description
	Equivalent Status Property
	C# Delegate
Connected	<p>This event occurs whenever a remote client connects to or disconnects from the ODA. A boolean pass parameter is set true on connection.</p> <p><code>odaClient.Connectivity.IsConnected()</code></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>odaClient.Status.TriggerComplete</code> to detect end of sweep.</p> <p>NewDataUpdateDelegate</p>
NewStatus	<p>This event occurs whenever a new status message is presented to the user on the front panel of the ODA. The pass parameters include the message itself and the severity of the message.</p> <p><code>odaClient.Status.ErrorFlag</code>, <code>odaClient.Status.LastError</code></p> <p>NewStatusDelegate</p>
PresetProgress	<p>This event occurs when the ODA begins and completes an instrument preset. The pass parameter indicates whether it is the beginning or, end.</p> <p><code>odaClient.Status.PresetComplete</code></p> <p>ProgressDelegate</p>



Table 9 ODA Events

Event	Description
	Equivalent Status Property
	C# Delegate
Remoting	<p>This event occurs when the user presses the local key on the ODA front panel and begins controlling the ODA manually even though a remote process is connected.</p> <p>force with <code>odaClient.Display.ToggleRemotIndicator</code></p> <p><b>RemotingDelegate</b></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>odaClient.Status.TriggerComplete</code> to detect end of sweep.</p> <p><b>SweepProgressDelegate</b></p>
TLSSettlingProbability	<p>This event occurs when the ODA queries the latest temperature reading from the tunable laser. The ODA calculates the estimated number of minutes until a Laser Settling is required, and reports it as a pass parameter.</p> <p><code>odaClient.Status.EstimatedTimeToTLSSettling</code></p> <p><b>TLSSettlingProbabilityDelegate</b></p>
TLSSettlingProgress	<p>This event occurs when the ODA starts, or completes the process of settling the tunable laser after temperature has drifted. The ODA will not make measurements while the settling is in progress.</p> <p><b>Status.TLSSettling</b></p> <p><b>ProgressDelegate</b></p>
TriggerProgress	<p>This event occurs at the start and finish of most ODA 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><b>Status.TriggerComplete</b></p> <p><b>TriggerProgressDelegate</b></p>

## Troubleshooting

As you develop your application, sooner or later you may lose synchronization with the ODA. 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 ODA, and usually the problem goes away at that point. If the problem is severe, the ODA 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 ODA synchronized.

If Automation errors persist, go into `Project > References` and clear the ODA objects, and then close the dialog. Reopen the dialog and reselect the references. Refer to [“Example 1: Simple Application for Controlling the ODA” on page 154](#) for a step-by-step procedure of adding references.

# Front Panel Maps

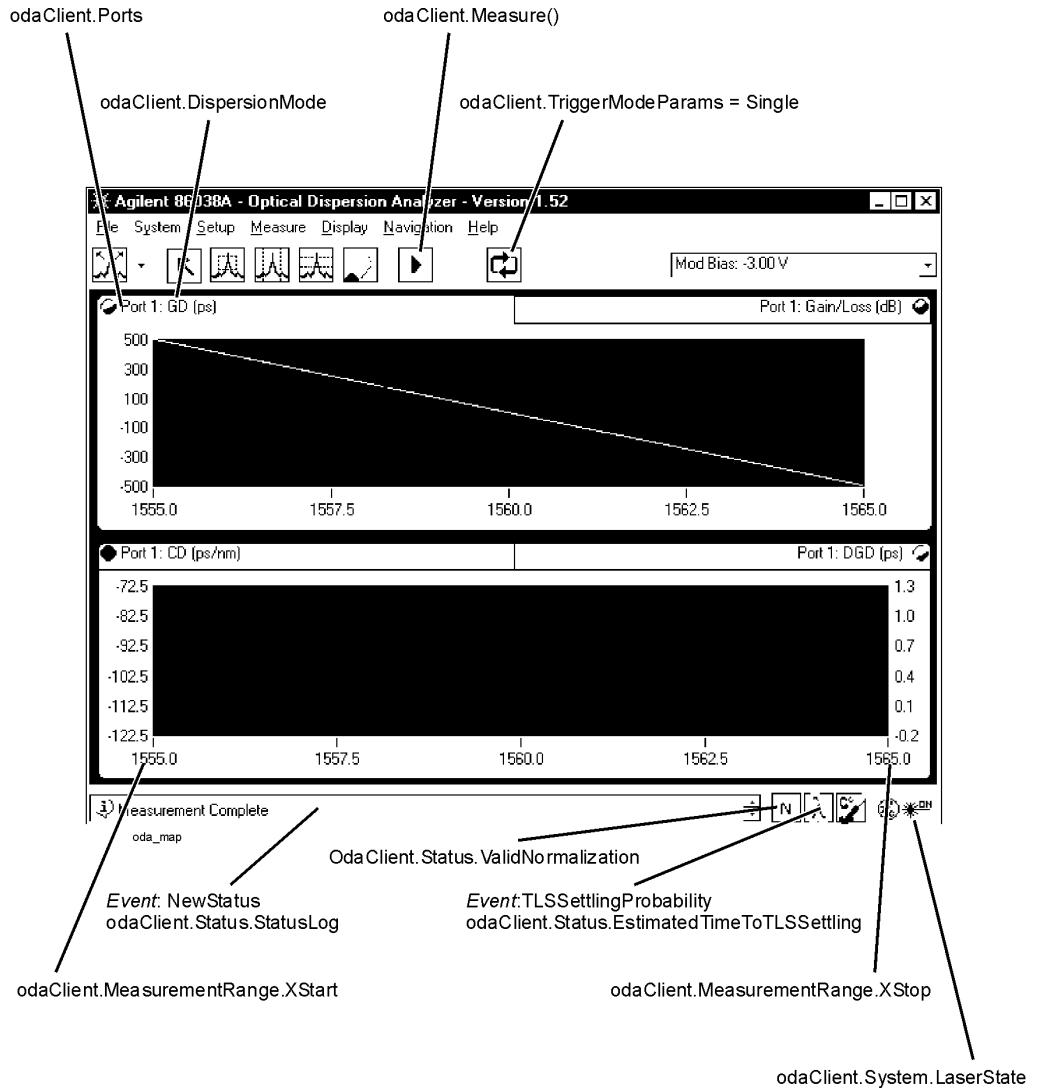


Figure 12 Front Panel Related Commands

**Agilent 86038A - Optical Dispersion Analyzer - Version 1.52**

File System Setup Measure Display Navigation Help

<u>Q</u> uick Setup...	Ctrl+Shift+Q
<u>C</u> D / PMD...	Ctrl+Shift+C
<u>L</u> ength...	Ctrl+Shift+L
<u>P</u> orts...	Ctrl+Shift+P
<u>M</u> easurement Range...	Ctrl+Shift+M
<u>N</u> ormalization Range...	Ctrl+Shift+N
<u>S</u> ensitivity / Resolution...	Ctrl+Shift+R
<u>L</u> aser Setup...	Ctrl+Shift+Z
<u>T</u> rigger Mode...	Ctrl+Shift+T

**Measurement Type**

Dispersion Measurement Type

- GD/CD Swept ①
- GD/CD Stepped ②
- GD/CD + DGD/PDL Stepped ③

Notes:

- Gain/Loss is always measured
- DGD/PDL measurements use polarization scanning

quick Setup

Ports

sens

CD/PMD

Meas Range

Laser Setup

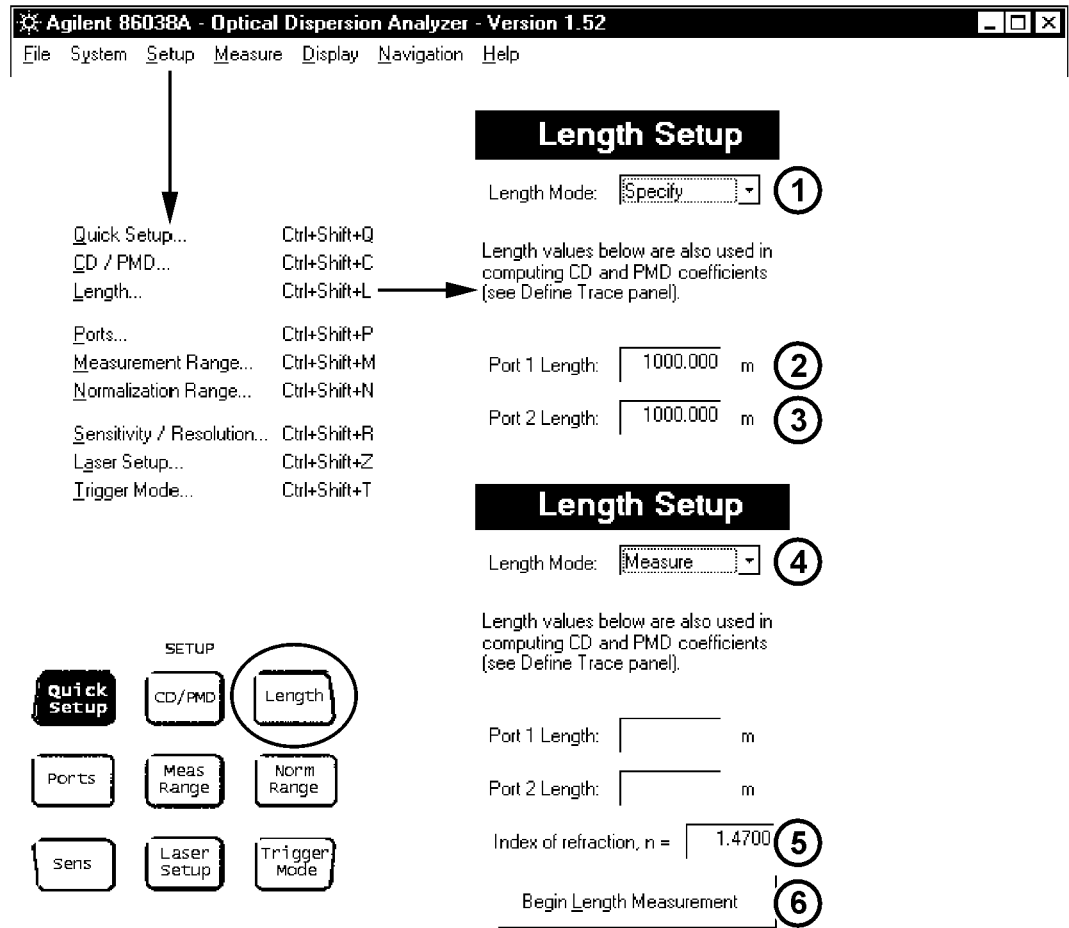
Length

Norm Range

Trigger Mode

- 1 OdaClient.DispersionMode = eDispersionMode\_CD\_Swept
- 2 OdaClient.DispersionMode = eDispersionMode\_CD\_Stepped
- 3 OdaClient.DispersionMode = eDispersionMode\_CD\_PMD\_Stepped

Figure 13 Measurement Type Related Commands



- 1 odaClient.Length.Mode = eLengthMode\_Specify
- 2 odaClient.Length.SpecifiedLengthPort1 = <value>
- 3 odaClient.Length.SpecifiedLengthPort2 = <value>
- 4 odaClient.Length.Mode = eLengthMode\_Measure
- 5 odaClient.Length.IndexOfRefraction= <value>
- 6 odaClient.Actions.MeasureLength()

Figure 14 Length Measurement Related Commands

The screenshot shows the 'Agilent 86038A - Optical Dispersion Analyzer - Version 1.52' software window. The 'Setup' menu is open, listing various configuration options with their respective keyboard shortcuts. An arrow points from the 'Measurement Range...' menu item to a 'Measurement Range' dialog box. The dialog box contains two input fields: 'Start' with the value '1555.0000 nm' (circled with a '1') and 'Stop' with the value '1585.0000 nm' (circled with a '2'). Below these fields is a 'Copy From Selected Graph' button. At the bottom left, a 'SETUP' front panel map shows a grid of buttons, with the 'Meas Range' button circled in red.

Menu Item	Keyboard Shortcut
Quick Setup...	Ctrl+Shift+Q
CD / PMD...	Ctrl+Shift+C
Length...	Ctrl+Shift+L
Ports...	Ctrl+Shift+P
Measurement Range...	Ctrl+Shift+M
Normalization Range...	Ctrl+Shift+N
Sensitivity / Resolution...	Ctrl+Shift+R
Laser Setup...	Ctrl+Shift+Z
Trigger Mode...	Ctrl+Shift+T

**Measurement Range**

Start:  nm ①

Stop:  nm ②

Copy From Selected Graph

SETUP

Buttons: Quick Setup, CD/PMD, Length, Ports, Meas Range, Norm Range, Sens, Laser Setup, Trigger Mode

- 1 OdaClient.MeasurementRange.XStart = <value>
- 2 OdaClient.MeasurementRange.XStop = <value>

Figure 15 Measurement Range Related Commands

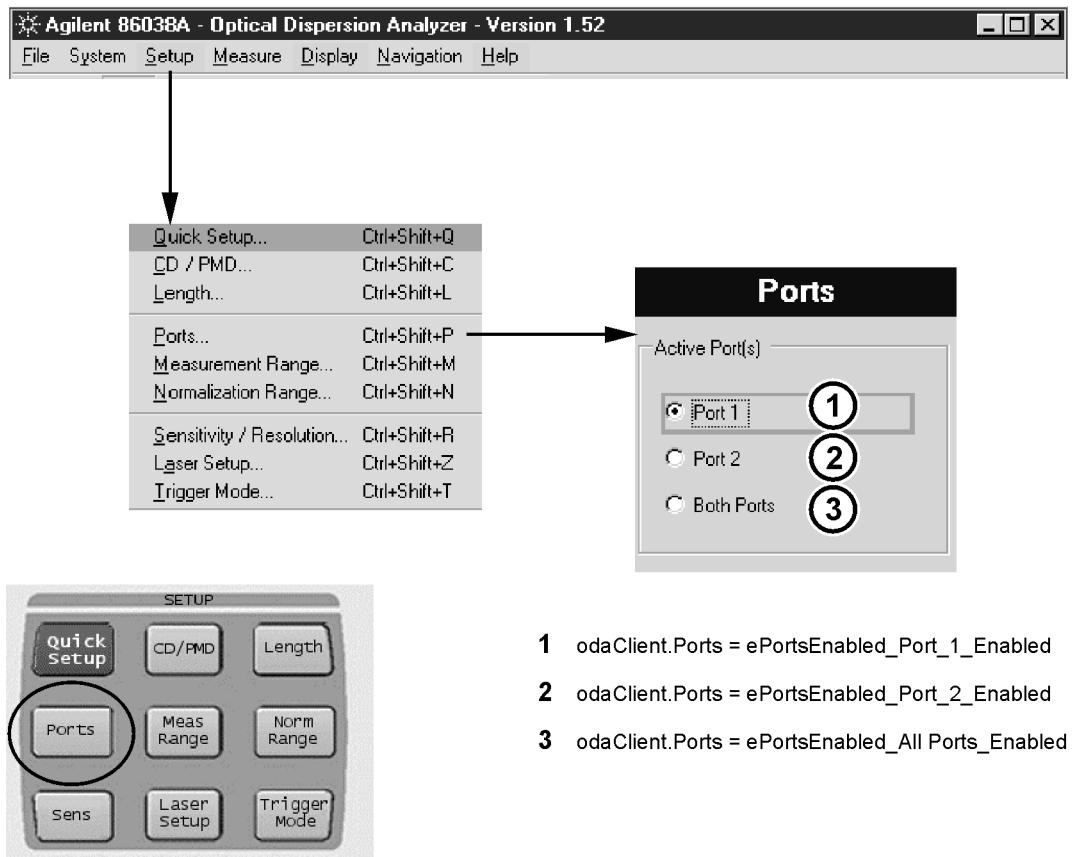


Figure 16 Port Selection Related Commands

**Agilent 86038A - Optical Dispersion Analyzer - Version 1.52**

File System Setup Measure Display Navigation Help

- Quick Setup... Ctrl+Shift+Q
- CD / PMD... Ctrl+Shift+C
- Length... Ctrl+Shift+L
- Ports... Ctrl+Shift+P
- Measurement Range... Ctrl+Shift+M
- Normalization Range... Ctrl+Shift+N
- Sensitivity / Resolution... Ctrl+Shift+R
- Laser Setup... Ctrl+Shift+Z
- Trigger Mode... Ctrl+Shift+T

**Norm Range**

Use Measurement Settings (1)

Customize

**Norm Range**

Use Measurement Settings

Customize (2)

Custom Normalization Settings

Start: 1457.0000 nm (3)

Stop: 1583.0000 nm (4)

Trace Points: 1601 (5)

Increment: nm (6)

SETUP

Quick Setup CD/PMD Length

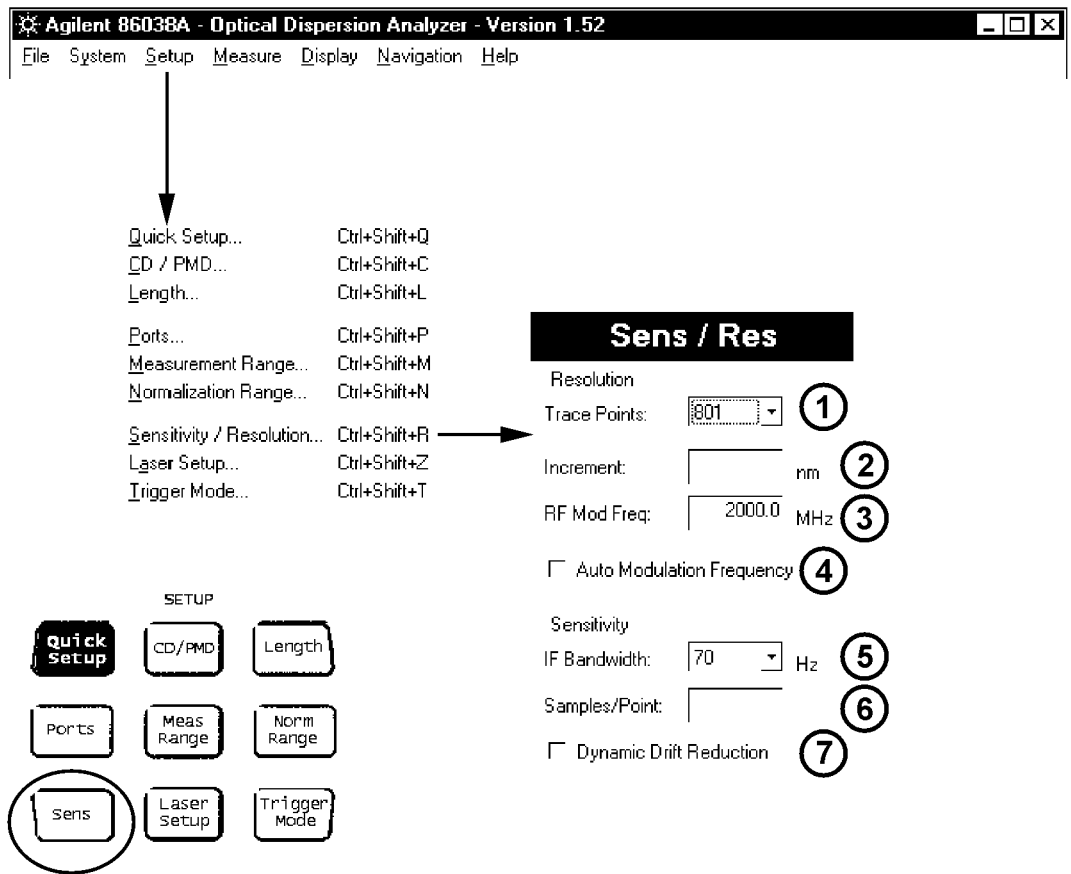
Ports Meas Range Norm Range

Sens Laser Setup Trigger Mode

- odaClient.NormalizationRange.UseCustomNormalization = False
- OdaClient.NormalizationRange.UseCustomNormalization = True
- OdaClient.NormalizationRange.XStart = <value>
- OdaClient.NormalizationRange.XStop = <value>
- odaClient.NormalizationRange.Swept.TracePoints = eTracePointsSweptMode\_Step1601
- odaClient.NormalizationRange.Stepped.Increment = <value>

Figure 17 Normalization Range Related Commands





- 1 `odaClient.Resolution.Swept.TracePoints = eTracePointsSweptMode_Step801`
- 2 `odaClient.Resolution.Stepped.Increment = <value>`
- 3 `OdaClient.Resolution.RFModulationFrequency= <value>`
- 4 `OdaClient.Resolution.Stepped.AutoModulationFrequency = False`
- 5 `odaClient.Sensitivity.IFBandwidth = eIFBandwidth_IFBW_70Hz`
- 6 `odaClient.Sensitivity.SamplesPerPoint = <value>`
- 7 `odaClient.Sensitivity.DynamicDriftReduction = False`

Figure 18 Sensitivity and Resolution Related Commands

## Example 1: Simple Application for Controlling the ODA

This example demonstrates a simple visual application to control the ODA. Refer to [“Recommended Control Sequence” on page 130](#) 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 ODA Remote Control\Examples

## Starting a VB 6.0 ODA Application

### Install Software

- Refer to “Recommended Control Sequence” on page 130.

### Application Shell

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

- 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 ODA 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 ODA as the default text. (Refer to the ODA Help > About to find your ODA 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 ODA.
- 6 Click **Browse**, then navigate to  
C:\Program Files\Agilent\Agilent ODA 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 Optical Dispersion 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 odaClient object as a member right after that line:

```
Private WithEvents odaClient As RemoteClient.Communicator
```

The odaClient will host all the interaction with the ODA. 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:

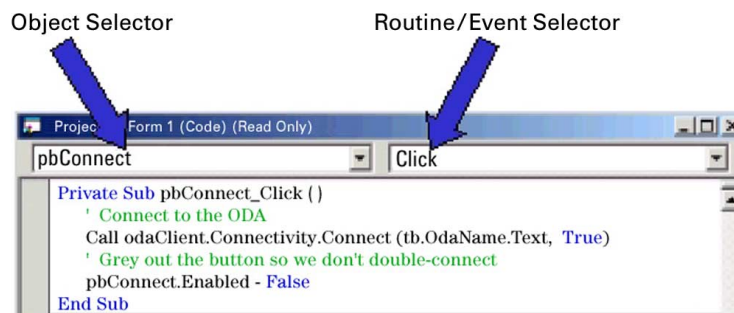
```
Set odaClient = New RemoteClient.Communicator
```

- 10 Before you can talk to the ODA, 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 ODA is established here. Place the following lines above the final End Sub:

```
Call odaClient.Connectivity.Connect(tbOdaName.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 ODA 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:

```
odaClient.Connectivity.Disconnect
odaClient.Connectivity.TerminateSession
```

The first line disconnects from the ODA itself, the second line cleans up the `odaClient`. 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 ODA front panel. A green banner will appear on the ODA, announcing Remote User Connected. If this does not appear, it is most likely because the ODA Name is incorrect, or the ODA itself is not properly configured for networking (check your LAN connection). Go to the ODA front panel and select `Help > About`. The name is on the About panel.

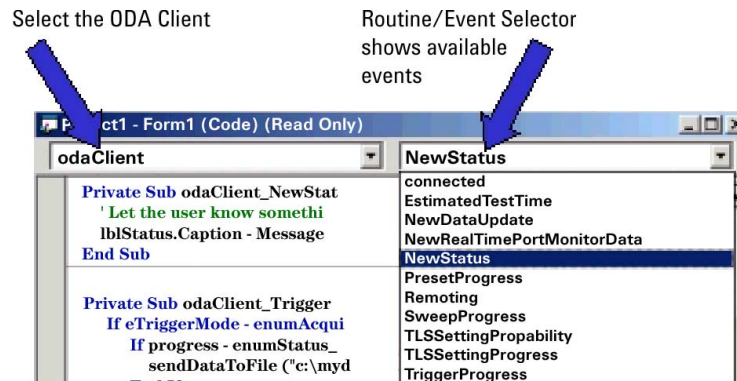
When you exit the program, it will disconnect from the ODA, leaving it free for front panel or other remote use.

## Event Handling

Event handling is added to monitor the ODA events. A very useful event is a new status message. The status message is shown on the bottom of the ODA 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 ODA display.

When an event like a new status message occurs, the ODA 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 ODA 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 `odaClient` 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 `odaClient`. 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 "ODA 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 **odaClient**. 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 `odaClient_NewStatus`, that will be called when a new status message is displayed on the ODA. 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 `odaClient_NewStatus`:
 

```
lblStatus.Caption = Message
```
- 3 Click the **Run** button, and connect to the ODA. The first message you will see is the Client connected message. Press **Local** on the ODA 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 ODA.

All of the ODA 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 ODA
If odaClient.Connectivity.IsConnected = False Then
    MsgBox ("Can't take a measurement yet - not connected.")
    Exit Sub
End If
odaClient.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 ODA 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 **odaClient** 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 `odaClient_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 =
```

<input type="checkbox"/>	enumAquisitionMode_eDriftMonitor
<input type="checkbox"/>	enumAquisitionMode_eLength
<input type="checkbox"/>	enumAquisitionMode_eMeasurement
<input type="checkbox"/>	enumAquisitionMode_eModBias
<input type="checkbox"/>	enumAquisitionMode_eNarmalization

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 odaClient\_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 = odaClient.Results.YData _
    (eMeasurementType_Gain,eODAPort_One)

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

  xStart = odaClient.Results.xStart(eMeasurementType_Gain)
  xStop = odaClient.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 ODA 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 ODA 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 ODA. 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 ODA as part of the sweep.

Changing a parameter consists of determining which property on which sub-object of `odaClient` 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 ODA. Find the function `pbSweep_click`, and add the following lines to check the value and send it to the ODA. The ODA 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 `odaClient.Measure`:

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

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

```
centerWavel = CDbI(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
```

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

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

**Complete source code**

```

Option Explicit

' Declare the odaClient.
Private WithEvents odaClient As RemoteClient.Communicator

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

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

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

Private Sub odaClient_TriggerProgress(ByVal progress As InstrumentOb-
jects.enumStatus, ByVal eTriggerMode As InstrumentObjects.enumAcqui-
sitionMode)
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 = odaClient.Results.YData _
(eMeasurementType_Gain,eODAPort_One)

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

xStart = odaClient.Results.xStart(eMeasurementType_Gain)
xStop = odaClient.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 ODA 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 ODA
Call odaClient.Connectivity.Connect(tbOdaName.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 ODA
If odaClient.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 = odaClient.MeasurementRange.xStart
stopWavel = odaClient.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

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

' Trigger the measurement
odaClient.Measure

End Sub
```

## Starting a VB.NET ODA Application

### Install Software

- Refer to “Recommended Control Sequence” on page 130.

### Application Shell

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

- 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 “ODA Name:” and a text box named tbOdaName. In the Properties dialog, click on **Text** and type the network name of your ODA as the default text. (Refer to ODA **Help > About** to find your ODA 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 ODA.
- 5 Select **Browse**, then navigate to C:\Program Files\Agilent\Agilent ODA 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 oda client object as a member:

```
Private WithEvents odaClient As RemoteClient.Communicator
```

The odaClient will host all the interaction with the ODA. 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:

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

- 8 Before you can interface with the ODA, 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 ODA, place the following lines between the Private Sub ... End Sub block:

```
odaClient.Connectivity.Connect(tbOdaName.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 ODA 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:

```
odaClient.Connectivity.Disconnect()
odaClient.Connectivity.TerminateSession()
```

The first line disconnects from the ODA itself, the second line cleans up the odaClient. 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 ODA front panel. A green banner will appear on the ODA, announcing Remote User Connected. If this does not appear, it is most likely because the ODA Name is incorrect, or the ODA itself is not properly configured for networking (check the LAN connection). Go to the ODA front panel and select Help > About to verify the ODA Name.

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

## Event Handling

Event handling is added to monitor the ODA events. A very useful event is a new status message. The status message is shown on the bottom of the ODA 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 ODA display.

When an event like a new status message occurs, the ODA 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 `odaClient` 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 `odaClient`. 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 `ODA 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 **`odaClient`**. 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 `odaClient_NewStatus`, which will be called when a new status message is displayed on the ODA. 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 odaClient_NewStatus( ...  
    ' Let the user know something happened  
    lblStatus.Text = Message  
End Sub
```

- 3 Click the **Run** button, and connect to the ODA. The first message you will see is the `Connected` message. Press **Local** on the ODA 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 ODA.

All of the ODA 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 ODA
If odaClient.Connectivity.IsConnected = False Then
    MessageBox.Show("Can't take a measurement yet - not con-
nected.")
    Return
End If
odaClient.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 ODA 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 odaClient 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 odaClient\_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.ODACom-
mon.enumAcquisitionMode.eMeasurement Then
' Only do something if the measurement is complete
  If progress = Agilent.LWD.Ag86038x.InstrumentObjects.ODACom-
mon.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 `odaClient_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 = odaClient.Results.YData _
  (Agilent.LWD.Ag86038x.InstrumentObjects.ODACommon.eMeasurement-
  Type.Gain, _
  Agilent.LWD.Ag86038x.InstrumentObjects.ODACommon.eODAPort.One)
  npoin = UBound(gainData) - LBound(gainData)
  xStart = odaClient.Results.XStart _
  (Agilent.LWD.Ag86038x.InstrumentObjects.ODACommon.eMeasurement-
  Type.Gain)
  xStop = odaClient.Results.XStop _
  (Agilent.LWD.Ag86038x.InstrumentObjects.ODACommon.eMeasurement-
  Type.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 ODA 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 ODA 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 ODA. First, a text box is added for the user to enter the desired value, and then the entered value will be sent to the ODA 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 ODA. Find the function pbSweep\_click, and add the following lines to check the value and send it to the ODA. The ODA 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 odaClient.measure:

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

```
startWavel = odaClient.MeasurementRange.XStart
stopWavel = odaClient.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
```

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

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

## Complete Source Code

```
Public Class Form1
    Inherits System.Windows.Forms.Form
    Private WithEvents odaClient 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
        odaClient = New RemoteClient.Communicator()
    End Sub

    Private Sub pbConnect_Click(ByVal sender As System.Object, ByVal e
As System.EventArgs) Handles pbConnect.Click
        odaClient.Connectivity.Connect(tbOdaName.Text)
        pbConnect.Enabled = False
    End Sub
End Class
```

```

End Sub

Private Sub Form1_Closing(ByVal sender As Object, ByVal e As System.ComponentModel.CancelEventArgs) Handles MyBase.Closing
    odaClient.Connectivity.Disconnect()
    odaClient.Connectivity.TerminateSession()
End Sub

Private Sub odaClient_NewStatus(ByVal Message As String, ByVal MessageType As Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eEventLogType) Handles odaClient.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 ODA
    If odaClient.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, oldCenter As Double

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

    centerWavel = CDBl(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

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

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

Private Sub odaClient_TriggerProgress(ByVal progress As Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.enumStatus, ByVal eTriggerMode As Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.enumAcquisitionMode) Handles odaClient.TriggerProgress
    ' Only do something if this is a measurement
    If eTriggerMode = Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.enumAcquisitionMode.eMeasurement 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 = odaClient.Results.YData(Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eMeasurementType.Gain, _
    Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eODAPort.One)
npoin = UBound(gainData) - LBound(gainData)

xStart = odaClient.Results.XStart(Agilent.LWD.Ag86038x.InstrumentObjects.ODACCommon.eMeasurementType.Gain)
xStop = odaClient.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 ODA 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# ODA Application

### Install Software

- Refer to “Recommended Control Sequence” on page 130.

### Application Shell

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

- 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** “ODA Name:” and a text box named tbOdaName.
- 5 In the Properties dialog, click on **Text** and type the network name of your ODA as the default text. (Refer to ODA Help > About to find the ODA 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 ODA. Select **Browse**, then navigate to C:\Program Files\Agilent\Agilent ODA 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 ODA involves using lots of constants such as eMeasurementType. The ‘using’ directive at the top of your C-sharp file lets you use the ODA 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 oda client object as a member. Insert the following line after the Text Box and Label declarations:

```
// Declare a local object to communicate with the ODA
private RemoteClient.Communicator odaClient;
```

The odaClient will host all the interaction with the ODA. 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:

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

- 9 Before you can talk to the ODA, 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 ODA is established. Place the following lines inside the curly braces:

```
odaClient.Connectivity.Connect(tbOdaName.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 ODA 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:

```
odaClient.Connectivity.Disconnect();
```

```
odaClient.Connectivity.TerminateSession();
```

The first line disconnects from the ODA itself, the second line cleans up the odaClient.

- 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 ODA front panel. A green banner will appear on the ODA, announcing Remote User Connected. If this does not appear, it is most likely because the ODA Name is incorrect, or the ODA itself is not properly configured for networking (check the LAN connection). Go to the ODA front panel and select Help > About. The ODA name is on the About panel.

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

## Event Handling

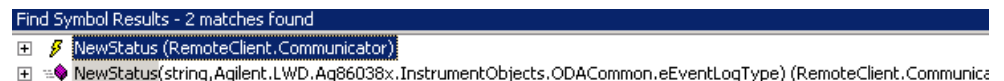
Event handling is added to monitor the ODA events. A very useful event is a new status message. The status message is shown on the bottom of the ODA display, which indicates when a sweep has begun, ended, and so on. By reflecting those messages on the remote application, you provide invaluable feedback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

When an event, like a new status message occurs, the ODA informs you by calling a method on your object. In fact, the ODA 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 ODA) and the event consumer (your code.) So, instead of passing a reference to your function to the `odaClient`, you pass a reference to your function to a delegate, and then pass the delegate to the `odaClient`.

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 `odaClient` has an event called `NewStatus`. The ODA 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 ODA could not do what you asked), Information (simple status update, like Sweep Started) or Warning (ODA 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 ODA 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 odaClient 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 ODA 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 "ODA Status."
- 2 Go to the code window and find where you declared the odaClient. Add the declaration of a local copy of the delegate here:

```
// Declare event handlers for ODA all!
private ODARemoting.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 ODA 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
odaClient.NewStatus += this.NewStatusHandler;
```

- 5** Since you registered the handler, you have to be careful to unregister it when you are done talking to the ODA. Find the `Form1_Closing` function, and insert the following lines before terminating the session:

```
// Unregister our status handler
odaClient.NewStatus -= this.NewStatusHandler;
```

- 6** Click the **Run** button, and connect to the ODA. The first message you will see is the Connected message. Press **Local** on the ODA 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 ODA.

All of the ODA 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 ODA
if (odaClient.Connectivity.IsConnected() == false)
{
    MessageBox.Show("Can't take a measurement yet - not con-
    nected.");
    return;
}
odaClient.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 ODA 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 `odaClient`.



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:

```
odaClient.TriggerProgress += this.TriggerHandler;
```

To unregister the handler, find the line where you `-= this.NewStatusHandler` and add the following line:

```
odaClient.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 ODA 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 = odaClient.Results.YData(ODACCommon.eMeasurement-
Type.Gain, ODACCommon.eODAPort.One);
npoin=gainData.Length;

// Get the start and stop wavelengths
xStart = odaClient.Results.XStart(ODACCommon.eMeasurement-
Type.Gain);
xStop = odaClient.Results.XStop(ODACCommon.eMeasurement-
Type.Gain);
xStep = (xStop - xStart)/ (npoin - 1);

// Open the file
StreamWriter file = new StreamWriter(fileName, false);
file.WriteLine("Sample ODA 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 ODA 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 ODA. First, a text box is added for the user to enter the desired value, and then the entered value will be sent to the ODA 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 ODA. Find the function pbSweep\_click, and add the following lines to check the value and send it to the ODA. The ODA 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=odaClient.MeasurementRange.XStart;
        stopWavel=odaClient.MeasurementRange.XStop;
        oldSpan=stopWavel - startWavel;

        // Write new start and stop
        odaClient.MeasurementRange.XStart=centerWavel - oldSpan/2;
        odaClient.MeasurementRange.XStop=centerWavel + oldSpan/2;

```

- 3** It works better if you first set the ODA 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 tbOdaName;
        private System.Windows.Forms.Label label1;
        private System.Windows.Forms.Label lblStatus;

        // Declare a local object to communicate with the ODA
        private RemoteClient.Communicator odaClient;

        // Declare event handlers for ODA - 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.eEvent-
LogType e)
    {
        // Let the user know something happened
        lblStatus.Text=msg;
    }

    private void TriggerProgressEvent(ODACCommon.enumStatus status,
ODACCommon.enumAcquisitionMode acqMode)
    {
        if (acqMode == ODACCommon.enumAcquisitionMode.eMeasure-
ment)
        {
            // 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)
    {

```

```

file // Transfer the gain data over from the ODA and save it to a
double[] gainData;
double xStart;
double xStop;
double xStep;
int npoin;
string buf;
int cnt;
double x;

// Get the gain data from port one
gainData = odaClient.Results.YData(ODACCommon.eMeasurement-
Type.Gain, ODACCommon.eODAPort.One);
npoin=gainData.Length;

// Get the start and stop wavelengths
xStart = odaClient.Results.XStart(ODACCommon.eMeasurement-
Type.Gain);
xStop = odaClient.Results.XStop(ODACCommon.eMeasurement-
Type.Gain);
xStep = (xStop - xStart)/ (npoin - 1);

// Open the file
StreamWriter file = new StreamWriter(fileName, false);
file.WriteLine("Sample ODA 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.Component-
Model.CancelEventArgs e)
{
    // Release the ODA
    odaClient.Connectivity.Disconnect();

    // Unregister our status handler
    odaClient.NewStatus -= this.NewStatusHandler;
    odaClient.TriggerProgress -= this.TriggerHandler;

    // Shut down the object
    odaClient.Connectivity.TerminateSession();
}

private void pbConnect_Click(object sender, System.EventArgs e)
{
    // Instantiate the odaClient
    odaClient = new RemoteClient.Communicator();

    // Connect to the ODA
    odaClient.Connectivity.Connect(tbOdaName.Text);

    // Create our event handlers
    this.NewStatusHandler = new ODARemoting.NewStatusDele-
gate(this.NewStatusEvent);
    this.TriggerHandler = new ODARemoting.TriggerProgressDele-
gate(this.TriggerProgressEvent);

    // Register event handlers

```

```

odaClient.NewStatus += this.NewStatusHandler;
odaClient.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 ODA
if (odaClient.Connectivity.IsConnected() == false)
{
    MessageBox.Show("Can't take a measurement yet - not con-
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=odaClient.MeasurementRange.XStart;
stopWavel=odaClient.MeasurementRange.XStop;
oldSpan=stopWavel - startWavel;
// Write new start and stop
odaClient.MeasurementRange.XStart=centerWavel - oldSpan/2;
odaClient.MeasurementRange.XStop=centerWavel + oldSpan/2;
// Take the measurement
odaClient.Measure();
}
}
}

```

## Example 2: Control from Excel

### Complete Source Code

```

Option Explicit
Private WithEvents m_ODARemoteControl As RemoteClient.Communicator
Public Sub initOdaLink()
    Set m_ODARemoteControl = New RemoteClient.Communicator
End Sub
Public Sub closeOdaLink()
    m_ODARemoteControl.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_ODARemoteControl.Connectivity.Connect (ODA_ID_Box.Value)

    ' Wait for ODA connection to come alive
    DoEvents
    Application.Wait (Now + TimeValue("0:00:01"))
    DoEvents
    While m_ODARemoteControl.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_ODARemoteControl.Results.XStart(eMeasurementType_Gain)
    xinc = m_ODARemoteControl.Results.Increment

    'Get CD Data - Port One
    Dim GDData() As Double
    GDData =
m_ODARemoteControl.Results.YData(eMeasurementType_GD,
eODAPort_One)

    'Get Gain Data - Port One
    Dim GainData() As Double
    GainData =
m_ODARemoteControl.Results.YData(eMeasurementType_Gain,
eODAPort_One)

    ' close session
    m_ODARemoteControl.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 = "Wave-
length, 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

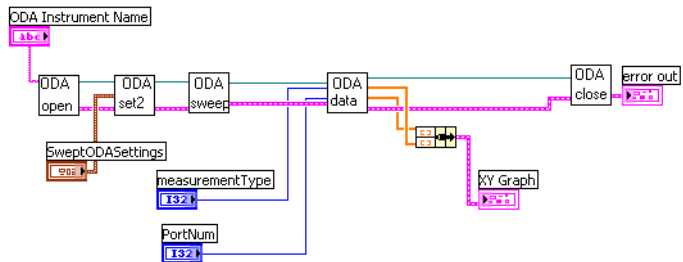
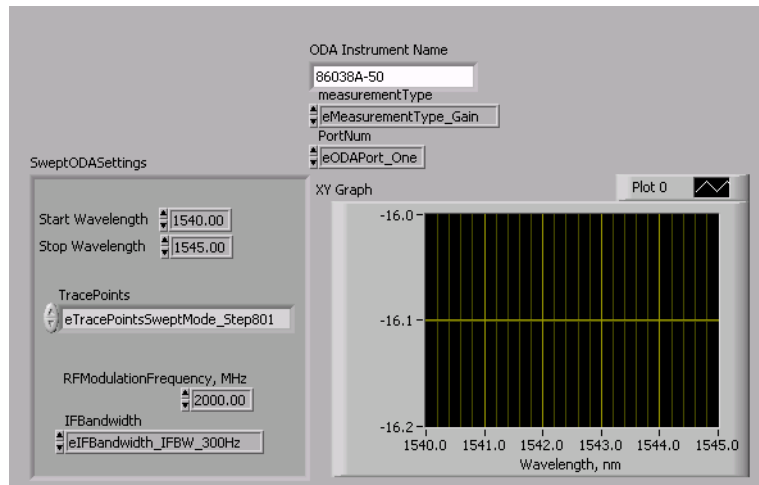


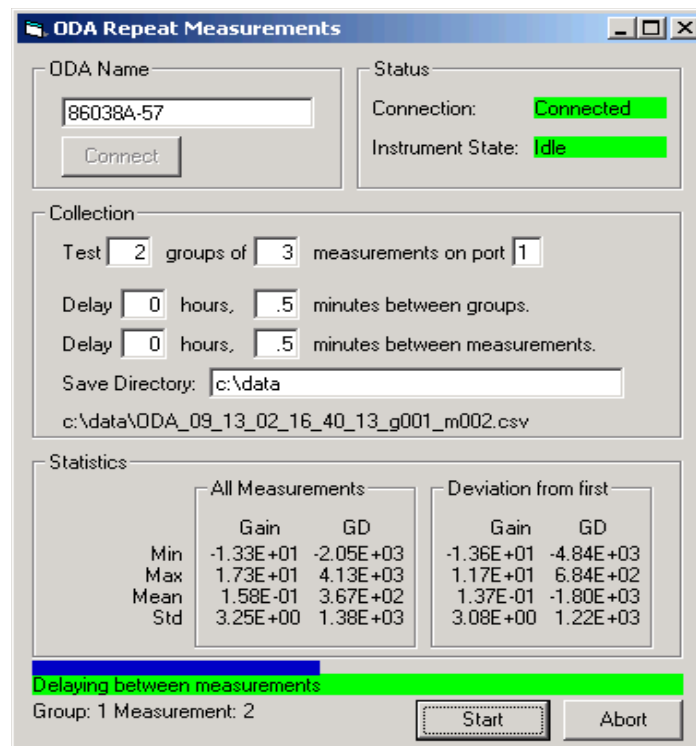
Table 10 Provided vi's

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 ODA.
ODAdata.vi	This vi transfers data from the ODA 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

### ODA Repeat Measurements

ODA Repeat Measurements is an example program distributed with the ODA in both binary and source code form. To use it, you set up a measurement you like on the ODA, 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 ODA, performing the required normalization.
- 2 Go to the ODA Help >About menu to obtain the ODA computer name. Type that name into the ODA 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.

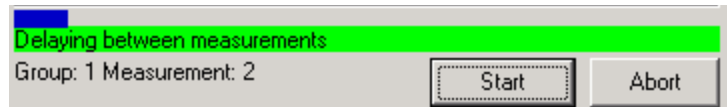
- 6 Click **Start** and observe the measurement results as they are taken.

		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

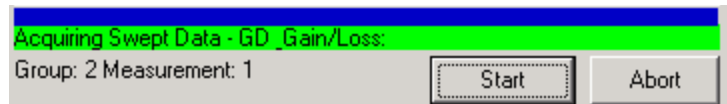
The statistics are calculated for Gain and Group Delay. The All Measurements block is calculated treating all wavelengths of all measurements equally. This is useful for measuring very flat and stable

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 area is the Status area. The blue bar is used to show the timer between ODA measurements:



During the measurement itself, the blue bar is full and the Status area reflects what the ODA is doing:



**7** Install the .NET Framework from Microsoft. Refer to "Install the Remote Client" on page 133.

**8** Install the remote programming application from the ODA documentation CD located at:

C:\Program Files\Agilent\Agilent ODA 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 ODA 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 ODA 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 ODA Status

Updating the ODA status is completely asynchronously from any measurement operation. Every 500 ms, StatusTimer wakes up and queries the ODA to find out what it is doing. This is handled in StatusTimer\_Timer(). In addition, whenever a new status message appears on the ODA front panel, the NewStatus event handler, odaClient\_NewStatus(), reads the message and displays it for the remote user as well.

# 6

## Programming Commands

ODA Remote Programming .....	192
Server to Client Events .....	193
Client to Server Commands .....	198
Base Commands .....	201
Actions Commands .....	203
Analysis - CurveFit Commands.....	205
Connectivity Commands .....	208
Display Commands.....	210
File Commands .....	213
Length Commands .....	215
MeasurementRange Commands .....	219
NormalizationRange Commands .....	221
NormalizationRange - Stepped Commands .....	225
Normalization Range - Swept Commands .....	227
Printing Commands .....	228
Resolution Commands.....	229
Resolution - Stepped Commands.....	231
Resolution - Swept Commands .....	234
Results Commands.....	235
Sensitivity Commands .....	239
Status Commands.....	242
System Commands.....	247
System - Identity Commands .....	248
TLS Commands .....	251

# ODA Remote Programming

The ODA 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 ODA measurement engine in some other development environment.

## ODA Remote Control DLLs

The ODARemoteControl DLLs provide a communication link with the ODA application. The DLLs are comprised of a set of Properties, Methods, and Events that together provide a basic set of remote ODA 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 ODA Remote Control\.

## .NET Interface Hierarchy

The ODA 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 86038A ODA to update the client with current data or instrument settings. In this context, the ODA 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 ODA. 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 ODA. 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 ODA

### **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.)

### **Parameters**

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

## **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 ODA 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 ODA 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 ODA 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 ODA 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 ODA 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 86038A ODA 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 odaClient As RemoteClient.Communicator`  
`odaClient = New RemoteClient.Communicator()`
- Examples shown with each command are presented in VB.NET and VB 6.0.

<b>Base Commands</b>	<b>201</b>
DispersionMode Property .....	201
Measure Method .....	201
Ports Property .....	202
TriggerModeParams Property .....	202
<b>Actions Commands</b>	<b>203</b>
Abort Method .....	203
Measure Method .....	203
MeasureLength Method .....	203
Normalize Method .....	204
PrintScreen Method .....	204
<b>Analysis - CurveFit Commands</b>	<b>205</b>
Calculate Method .....	205
Catalog Property .....	206
Description Method .....	206
ShortName Method .....	207
<b>Connectivity Commands</b>	<b>208</b>
Connect Method .....	208
Disconnect Method .....	208
IsConnected Method .....	209
TerminateSession Method .....	209
<b>Display Commands</b>	<b>210</b>
AutoscaleDuringMeasurement Method .....	210
DefineTrace Method .....	210
GDOffsetMode Property .....	211
ToggleMode Method .....	212
ToggleRemoteIndicator Method .....	212
<b>File Commands</b>	<b>213</b>
ExportCSV Method .....	213
Recall Method .....	213
Save Method .....	214
<b>Length Commands</b>	<b>215</b>
IndexOfRefraction Property .....	215
Mode Property .....	215
SpecifiedLengthPort1 Property .....	216
SpecifiedLengthPort2 Property .....	216
ValidIndexOfRefraction Method .....	217
ValidSpecifiedLength Method .....	217
<b>MeasurementRange Commands</b>	<b>219</b>
ValidStartStop Method .....	219
XStart Property .....	219
XStop Property .....	220
<b>NormalizationRange Commands</b>	<b>221</b>
ClearGDAndInsertionLossOffsets Method .....	221
ClearModBias Method .....	221
ClearPDA Method .....	222
PDAEnabled Property .....	222
UseCustomNormalization Property .....	222
ValidStartStop Method .....	223
XStart Property .....	223
XStop Property .....	224

<b>NormalizationRange - Stepped Commands</b>	<b>225</b>
Increment Property.....	225
ValidIncrement Method.....	225
<b>Normalization Range - Swept Commands</b>	<b>227</b>
TracePoints Property.....	227
<b>Printing Commands</b>	<b>228</b>
PrintScreen Method.....	228
<b>Resolution Commands</b>	<b>229</b>
RFModulationFrequency Property.....	229
ValidRFModulationFrequency Method.....	229
<b>Resolution - Stepped Commands</b>	<b>231</b>
AutoModulationFrequency Property.....	231
Increment Property.....	231
ValidIncrement Method.....	232
ValidTracePoints Method.....	232
<b>Resolution - Swept Commands</b>	<b>234</b>
TracePoints Property.....	234
<b>Results Commands</b>	<b>235</b>
FiberLength Method.....	235
Increment Property.....	235
XStart Method.....	236
XStop Method.....	236
XValueFromPointIndex Method.....	237
YData Method.....	237
<b>Sensitivity Commands</b>	<b>239</b>
DynamicDriftReduction Property.....	239
IFBandwidth Property.....	239
SamplesPerPoint Property.....	240
ValidSamplesPerPoint Method.....	240
<b>Status Commands</b>	<b>242</b>
ErrorFlag Property.....	242
EstimatedTimeToTLSSettling Property.....	242
InstrumentState Property.....	243
LastError Property.....	243
PresetComplete Property.....	244
StatusLog Method.....	244
TLSSettling Property.....	245
TriggerComplete Property.....	245
ValidNormalization Property.....	246
<b>System Commands</b>	<b>247</b>
LaserState Method.....	247
Preset Method.....	247
<b>System - Identity Commands</b>	<b>248</b>
CompanyName Property.....	248
ProductId Property.....	248
ProductName Property.....	248
ProductVersion Property.....	249
RemotingVersion Property.....	249
<b>TLS Commands</b>	<b>251</b>
ForceSettling Method.....	251
HighPower Read Only Property.....	251



# Base Commands

## DispersionMode Property

Selects CD Stepped, CD Swept, or CD\_PMD Stepped. 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

```
odaClient.DispersionMode=  
_InstrumentObjects.ODACCommon.eDispersionMode.CD_Swept
```

### VB 6.0 Syntax

```
odaClient.DispersionMode=eDispersionMode_CD_Swept
```

## 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

```
odaClient.Measure()
```

### VB 6.0 Syntax

```
odaClient.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.ePortsEnabled

### VB.NET Syntax

```
odaClient.Ports =  
_InstrumentObjects.ODACCommon.ePortsEnabled.Port_1_Enabled
```

### VB 6.0 Syntax

```
odaClient.Ports=ePortsEnabled_Port_1_Enabled
```

## 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

```
odaClient.TriggerModeParams =  
_InstrumentObjects.ODACCommon.eTriggerType.TriggerSingle
```

### VB 6.0 Syntax

```
odaClient.TriggerModeParams = eTriggerType_TriggerSingle
```

## 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

```
odaClient.Actions.Abort()
```

#### VB 6.0 Syntax

```
odaClient.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

```
odaClient.Actions.Measure()
```

#### VB 6.0 Syntax

```
odaClient.Actions.Measure
```

### MeasureLength Method

Starts a length measurement.

#### VB.NET Syntax

```
odaClient.Actions.MeasureLength()
```

#### VB 6.0 Syntax

```
odaClient.Actions.MeasureLength
```

## Normalize Method

Starts a normalization. Set the pass parameter to False to perform a normalization, or to True to only perform a modulator bias adjustment.

### Parameters

Optional ByVal ModBiasOnly As Boolean = False

If ModBiasOnly = True, then only an optical modulator optimization is done during normalization.

### VB.NET Syntax

```
odaClient.Actions.Normalize(False)
```

### VB 6.0 Syntax

```
odaClient.Actions.Normalize
```

## PrintScreen Method

This command is equivalent to the front panel Print button.

### VB.NET Syntax

```
odaClient.Actions.PrintScreen()
```

### VB.6.0 Syntax

```
odaClient.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 ODA.

### 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 = odaClient.Analysis.CurveFit.Calculate _  
("Quadratic", XStart, XIncrement, InTraceData, GeneralForm, Equation)
```

#### VB 6.0 Syntax

```
Dim OutTraceData() As Double
```

```
OutTraceData = odaClient.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 = odaClient.Analysis.CurveFit.Catalog
```

### VB 6.0 Syntax

```
Dim MyCat() As String
```

```
MyCat = odaClient.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 = odaClient.Analysis.CurveFit.Description("Quadratic")
```

**VB 6.0 Syntax**

```
Dim desc As String
```

```
desc = odaClient.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 = odaClient.Analysis.CurveFit.ShortName("Quadratic")
```

**VB 6.0 Syntax**

```
Dim eqName As String
```

```
eqName = odaClient.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 ODA.

### Connect Method

Connects a remoting session to the remote server. Pass it the network name of the ODA. The network name can be found by looking at the Help > About dialog from the ODA front panel. You must wait until the connected event returns true before sending commands.

You can also use the IP address of the ODA 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 ODA.

#### VB.Net Syntax

```
odaClient.Connectivity.Connect("MYODA")
```

#### VB 6.0 Syntax

```
odaClient.Connectivity.Connect("MYODA")
```

### Disconnect Method

Disconnects the remoting session from the ODA. Events will not be sent to the remote client once it disconnects.

#### VB.NET Syntax

```
odaClient.Connectivity.Disconnect()
```

#### VB 6.0 Syntax

```
odaClient.Connectivity.Disconnect
```



## IsConnected Method

Checks to see if a remoting connection is established to the ODA. 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 (odaClient.Connectivity.IsConnected() = True) Then
    MsgBox("Connected")
End If
```

### VB 6.0 Syntax

```
If (odaClient.Connectivity.IsConnected = True) Then
    MsgBox("Connected")
End If
```

## TerminateSession Method

This method unregisters the communicator channel to the ODA. TerminateSession should be called when remoting to the ODA has been concluded.

### VB.NET Syntax

```
odaClient.Connectivity.TerminateSession
```

### VB 6.0 Syntax

```
odaClient.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.Stepped.Increment property.

#### Parameters

ByVal enabled As Boolean - True turns on auto scaling.

#### Related Command

Resolution.Stepped.Increment property

#### VB.NET Syntax

```
odaClient.Display.AutoscaleDuringMeasurement(True)
```

#### VB 6.0 Syntax

```
odaClient.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

```
odaClient.Display.DefineTrace(0, _  
InstrumentObjects.ODACCommon.eTraceLetter.TraceA, _  
InstrumentObjects.ODACCommon.eODAPort.One, _  
InstrumentObjects.ODACCommon.eMeasurementType.GD)
```

### VB 6.0 Syntax

```
Call odaClient.Display.DefineTrace(0, _  
eTraceLetter_TraceA, _  
eODAPort_One, _  
eMeasurementType_GD)
```

## GDOffsetMode Property

The ODA 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

```
odaClient.Display.GDOffsetMode = _  
InstrumentObjects.ODASensitivity.eGDOffsetMode.CenterPoint
```

### VB 6.0 Syntax

```
odaClient.Display.GDOffsetMode = eGDOffsetMode_CenterPoint
```

## ToggleMode Method

Toggles the ODA 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 86038A user interface.

### VB.NET Syntax

```
odaClient.Display.ToggleMode _  
(InstrumentObjects.ODACCommon.eLayoutMode.QUAD_Graph)
```

### VB 6.0 Syntax

```
odaClient.Display.ToggleMode(eLayoutMode_QUAD_Graph)
```

## ToggleRemoteIndicator Method

Use this method to return the ODA to local control. This is equivalent to pressing the front panel Local key. To put the ODA in local mode, call this method with a value of False.

### Parameters

ByVal remote As Boolean - True sets the ODA UI to remote mode. False sets the ODA UI to local mode.

### VB.NET Syntax

```
odaClient.Display.ToggleRemoteIndicator(True)
```

### VB 6.0 Syntax

```
odaClient.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<sup>®</sup> or Excel<sup>®</sup>.

An error will be raised if the specified path does not exist on the ODA. 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 ODA system itself.

#### VB.NET Syntax

```
odaClient.File.ExportCSV("D:\Data\MyExport.csv")
```

#### VB 6.0 Syntax

```
odaClient.File.ExportCSV("D:\Data\MyExport.csv")
```

### 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 ODA.

#### Parameters

ByVal filename As String - Full filename and path of the ODA file. The file must exist on the ODA system.

#### VB.NET Syntax

```
odaClient.File.Recall("D:\Data\MyState.oda")
```

#### VB 6.0 Syntax

```
odaClient.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 .oda 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 ODA. Also, any existing file by the same name will be overwritten.

### Parameters

ByVal filename As String - Full filename and path of the ODA file. The file will be saved on the ODA system itself.

### VB.NET Syntax

```
odaClient.File.Save("D:\Data\MyState.oda")
```

### VB 6.0 Syntax

```
odaClient.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

```
odaClient.Length.IndexOfRefraction = 1.50
```

#### VB 6.0 Syntax

```
odaClient.Length.IndexOfRefraction = 1.50
```

### Mode Property

Determines if the ODA should use the user-specified value for length or the measured value.

#### Property Value

InstrumentObjects.ODALength.eLengthMode

eLengthMode.Measure - The ODA will measure length based on the index of refraction.

eLengthMode.Specify - The ODA will use the Specified Length property to set length.

#### VB.NET Syntax

```
odaClient.Length.Mode = _  
InstrumentObjects.ODALength.eLengthMode.Measure
```

**VB 6.0 Syntax**

```
odaClient.Length.Mode=eLengthMode_Measure
```

**SpecifiedLengthPort1 Property**

Specifies the length of the fiber-optic cable (in meters) for port 1. The length will also be used in computing CD and PMD coefficients.

**Property Value**

Double

**Related Command**

Length.ValidSpecifiedLength method

**VB.NET Syntax**

```
odaClient.Length.SpecifiedLengthPort1=3.8
```

**VB 6.0 Syntax**

```
odaClient.Length.SpecifiedLengthPort1=3.8
```

**SpecifiedLengthPort2 Property**

Specifies the length of the fiber-optic cable (in meters) for port 2. The length will also be used in computing CD and PMD coefficients.

**Property Value**

Double

**Related Command**

Length.ValidSpecifiedLength method

**VB.NET Syntax**

```
odaClient.Length.SpecifiedLengthPort2=3.8
```

**VB 6.0 Syntax**

```
odaClient.Length.SpecifiedLengthPort2=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
```

```
odaClient.Length.ValidIndexOfRefraction (minN, maxN, defaultN)
```

### VB 6.0 Syntax

```
Dim minN As Double
```

```
Dim maxN As Double
```

```
Dim defaultN As Double
```

```
Call odaClient.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

odaClient.Length.ValidSpecifiedLength(minLEN, maxLEN, defaultLEN)

**VB 6.0 Syntax**

Dim minLEN As Double

Dim maxLEN As Double

Dim defaultLEN As Double

Call odaClient.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
```

```
odaClient.MeasurementRange.ValidStartStop (minWL, maxWL)
```

#### VB 6.0 Syntax

```
Dim minWL As Double
```

```
Dim maxWL As Double
```

```
Call odaClient.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 = odaClient.MeasurementRange.XStart odaClient.Measurement_  
Range.XStart = myStart -1.0
```

**VB 6.0 Syntax**

```
Dim myStart As Double
```

```
myStart = odaClient.MeasurementRange.XStart odaClient.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 = odaClient.MeasurementRange.XStop odaClient.Measurement_  
Range.XStop = myStop +1.0
```

**VB 6.0 Syntax**

```
Dim myStop As Double
```

```
myStop = odaClient.MeasurementRange.XStop odaClient.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

```
odaClient.NormalizationRange.ClearGDAndInsertionLossOffsets _  
(InstrumentObjects.ODACCommon.eODAPort.One)
```

#### VB 6.0 Syntax

```
odaClient.NormalizationRange.ClearGDAndInsertionLossOffsets _  
(eODAPort_One)
```

### ClearModBias Method

Clears the current optical modulator bias value.

#### VB.NET Syntax

```
odaClient.NormalizationRange.ClearModBias()
```

#### VB 6.0 Syntax

```
odaClient.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

```
odaClient.NormalizationRange.ClearPDA()
```

### VB 6.0 Syntax

```
odaClient.NormalizationRange.ClearPDA
```

## 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

```
odaClient.NormalizationRange.PDAEnabled = True
```

### VB 6.0 Syntax

```
odaClient.NormalizationRange.PDAEnabled = True
```

## UseCustomNormalization Property

Instructs the ODA 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 ODA to use the NormalizationRange settings, False tells the ODA to use Measurement settings.

### VB.NET Syntax

```
odaClient.NormalizationRange.UseCustomNormalization = True
```

**VB 6.0 Syntax**

```
odaClient.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
```

```
odaClient.NormalizationRange.ValidStartStop(minWL, maxWL)
```

**VB 6.0 Syntax**

```
Dim minWL As Double
```

```
Dim maxWL As Double
```

```
Call odaClient.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**

```
odaClient.NormalizationRange.UseCustomNormalization = True
```

```
odaClient.NormalizationRange.XStart = 1550.0
```

**VB 6.0 Syntax**

```
odaClient.NormalizationRange.UseCustomNormalization = True
```

```
odaClient.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**

```
odaClient.NormalizationRange.UseCustomNormalization = True
```

```
odaClient.NormalizationRange.XStop = 1560.0
```

**VB 6.0 Syntax**

```
odaClient.NormalizationRange.UseCustomNormalization = True
```

```
odaClient.NormalizationRange.XStop = 1560.0
```



## NormalizationRange - Stepped Commands

The NormalizationRange.Stepped property of the remote client object returns an object that handles settings unique to stepped normalization.

### Increment Property

Specifies the separation between adjacent measurement wavelengths.

#### Property Value

Double - The wavelength spacing between points, in nanometers.

#### Related Command

NormalizationRange.Stepped.ValidIncrement method

#### VB.NET Syntax

```
odaClient.NormalizationRange.Stepped.Increment = 0.5
```

#### VB 6.0 Syntax

```
odaClient.NormalizationRange.Stepped.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.Stepped.Increment property

#### VB.NET Syntax

```
Dim minXinc As Double
```

```
Dim maxXinc As double
```

```
Dim defXinc As Double
```

```
odaClient.NormalizationRange.Stepped.ValidIncrement _  
(minXinc, maxXinc, defXinc)
```

### **VB 6.0 Syntax**

```
Dim minXinc As Double
```

```
Dim maxXinc As double
```

```
Dim defXinc As Double
```

```
Call odaClient.NormalizationRange.Stepped.ValidIncrement _  
(minXinc, maxXinc, defXinc)
```

## Normalization Range - Swept Commands

The `NormalizationRange.Swept` property of the remote client object returns an object that handles settings unique to swept normalization.

### TracePoints Property

Sets the number of trace points to either 801 or 1601. The greater the number of points the greater the measurement accuracy. However, measurement time also increases.

### Property Value

`InstrumentObjects.ODAResolution.eTracePointsSweptMode` - The number of trace points for a swept mode normalization.

### VB.NET Syntax

```
odaClient.NormalizationRange.Swept.TracePoints = _  
ODAResolution.eTracePointsSweptMode.Step1601
```

### VB 6.0 Syntax

```
odaClient.NormalizationRange.Swept.TracePoints = _  
eTracePointsSweptMode_Step1601
```

## 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

```
odaClient.Printing.PrintScreen()
```

#### VB 6.0 Syntax

```
odaClient.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 ODA 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

```
odaClient.Resolution.RFModulationFrequency = 2000
```

#### VB 6.0 Syntax

```
odaClient.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

odaClient.Resolution.ValidRFModulationFrequency (Min, Max, Default)

**VB 6.0 Syntax**

Dim Min As Double

Dim Max As Double

Dim Default As Double

Call odaClient.Resolution.ValidRFModulationFrequency (Min, Max, Default)

## Resolution - Stepped Commands

The Stepped property of the Resolution object returns an object, which handles resolution settings unique to stepped mode.

### AutoModulationFrequency Property

Automatically selects the modulation frequency to avoid RF phase wrapping. The amount of RF phase change that is generated in response to a wavelength increment is determined by the product of the wavelength increment, the modulation frequency, and the dispersion of the test device. Automatic adjustment of the modulation frequency insures that the per-wavelength-step phase change is always less than  $\pm 180$  degrees.

#### Property Value

Boolean - If True, the RF modulation frequency will be set automatically by the ODA.

#### VB.NET Syntax

```
odaClient.Resolution.Stepped.AutoModulationFrequency = True
```

#### VB 6.0 Syntax

```
odaClient.Resolution.Stepped.AutoModulationFrequency = True
```

### Increment Property

Specifies the separation between adjacent measurement wavelengths.

#### Property Value

Double - The wavelength spacing between points, in nanometers.

#### Related Command

Resolution.Stepped.ValidIncrement method,  
Resolution.Stepped.ValidTracePoints method

#### VB.NET Syntax

```
odaClient.Resolution.Stepped.Increment = .63
```

#### VB 6.0 Syntax

```
odaClient.Resolution.Stepped.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.Stepped.Increment property,  
Resolution.Stepped.ValidTracePoints method

### VB.NET Syntax

```
Dim minXinc As Double
```

```
Dim maxXinc As double
```

```
Dim defXinc As Double
```

```
odaClient.Resolution.Stepped.ValidIncrement (minXinc, maxXinc, defXinc)
```

### VB 6.0 Syntax

```
Dim minXinc As Double
```

```
Dim maxXinc As double
```

```
Dim defXinc As Double
```

```
Call odaClient.Resolution.Stepped.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.Stepped.Increment property,  
Resolution.Stepped.ValidIncrement method

**VB.NET Syntax**

Dim MinXtrc As Double

Dim MaxXtrc As Double

odaClient.Resolution.Stepped.ValidTracePoints (MinXtrc, MaxXtrc)

**VB 6.0 Syntax**

Dim MinXtrc As Double

Dim MaxXtrc As Double

Call odaClient.Resolution.Stepped.ValidTracePoints (MinXtrc, MaxXtrc)

## Resolution - Swept Commands

The Swept property of the Resolution object returns an object that handles resolution settings unique to swept mode.

### TracePoints Property

Sets the number of points used in a swept mode measurement. Choices are 801 or 1601 points.

#### Property Value

InstrumentObjects.ODAResolution.eTracePointsSweptMode - The number of trace points for a swept mode measurement.

#### VB.NET Syntax

```
odaClient.Resolution.Swept.TracePoints=_ODAResolution.eTracePoints  
SweptMode.Step801
```

#### VB 6.0 Syntax

```
odaClient.Resolution.Swept.TracePoints=_eTracePoints  
SweptMode_Step801
```

## 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 = odaClient.Results.FiberLength _
(InstrumentObjects.ODACCommon.eMeasNormDataType.Measurement, _
InstrumentObjects.ODACCommon.eODAPort.One)
```

#### VB 6.0 Syntax

Dim Length As Double

```
Length = odaClient.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

$$\lambda = XStart + (i - 1)XIncrement, i = 1 \text{ to number of points}$$

**Property Value**

Double - The x-axis increment

**VB.NET Syntax**

```
Dim XInc As Double
```

```
XInc = odaClient.Results.Increment
```

**VB 6.0 Syntax**

```
Dim XInc As Double
```

```
XInc = odaClient.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 = odaClient.Results.XStart_  
(InstrumentObjects.ODACCommon.eMeasurementType.Gain)
```

**VB 6.0 Syntax**

```
Dim sVal As Double
```

```
sVal = odaClient.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 = odaClient.Results.XStop_  
(InstrumentObjects.ODACCommon.eMeasurementType.Gain)
```

**VB 6.0 Syntax**

```
Dim xStop As Double
```

```
xStop = odaClient.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 = odaClient.Results.XValueFromPointIndex_  
(InstrumentObjects.ODACCommon.eMeasurementType.Gain,2)
```

**VB 6.0 Syntax**

```
Dim xVal as Double
```

```
xVal = odaClient.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 = odaClient.Results.YData _  
(InstrumentObjects.ODACCommon.eMeasurementType.Gain, _  
InstrumentObjects.ODACCommon.eODAPort.One)
```

**VB 6.0 Syntax**

Dim rawData() As Double

```
rawData = odaClient.Results.YData(eMeasurementType_Gain, _  
eODAPort_One)
```

## Sensitivity Commands

### DynamicDriftReduction Property

*For Stepped measurement modes only*

Sets/Gets the DynamicDriftReduction 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.

#### Property Value

Boolean = True if enabled

#### VB.NET Syntax

```
If(odaClient.Sensitivity.DynamicDriftReduction = False) Then  
    odaClient.Sensitivity.DynamicDriftReduction = True  
End If
```

#### VB 6.0 Syntax

```
If(odaClient.Sensitivity.DynamicDriftReduction = False) Then  
    odaClient.Sensitivity.DynamicDriftReduction = 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**

```
odaClient.Sensitivity.IFBandwidth = _  
InstrumentObjects.ODASensitivity.eIFBandwidth.IFBW_100Hz
```

**VB 6.0 Syntax**

```
odaClient.Sensitivity.IFBandwidth=eIFBandwidth_IFBW_100Hz
```

**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**

```
odaClient.Sensitivity.SamplesPerPoint = 201
```

**VB 6.0 Syntax**

```
odaClient.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 ODA.

### **Related Command**

Sensitivity.SamplesPerPoint property

### **VB.NET Syntax**

```
Dim minSamples As Integer
```

```
Dim maxSamples As Integer
```

```
Dim defSamples As Integer
```

```
odaClient.Sensitivity.ValidSamplesPerPoint _  
(minSamples, maxSamples, defSamples)
```

### **VB 6.0 Syntax**

```
Dim minSamples As Integer
```

```
Dim maxSamples As Integer
```

```
Dim defSamples As Integer
```

```
Call odaClient.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(odaClient.Status.ErrorFlag = True) Then
```

```
    MsgBox("An error occurred.")
```

```
End If
```

#### VB 6.0 Syntax

```
If(odaClient.Status.ErrorFlag = True) Then
```

```
    MsgBox("An error occurred.")
```

```
End If
```

### EstimatedTimeToTLSSettling Property

Indicates the estimated time until the next TLSSettling begins.

The ODA 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

```
odaClient.Status.EstimatedTimeToTLSSettling
```

**VB 6.0 Syntax**

```
odaClient.Status.EstimatedTimeToTLSSettling
```

**InstrumentState Property**

Indicates the current state of the ODA, that is, idle, initializing, measuring, TLS Settling, or aborting. Refer to [“ODA Enumerations” on page 141](#) for possible values.

**Property Value**

InstrumentObjects.ODACCommon.eInstrumentStatus - Denotes the current instrument state.

**VB.NET Syntax**

```
Dim curStatus As InstrumentObjects.ODACCommon.eInstrumentStatus
curStatus = odaClient.Status.InstrumenState
```

**VB 6.0 Syntax**

```
Dim curStatus as InstrumentObjects.eInstrumentStatus
curStatus = odaClient.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(odaClient.Status.ErrorFlag = True) Then
    MsgBox("Error:" + odaClient.Status.LastError)
End If
```

**VB 6.0 Syntax**

```
If(odaClient.Status.ErrorFlag = True) Then
    MsgBox("Error:" + odaClient.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 = odaClient.Status.PresetComplete
```

### VB 6.0 Syntax

```
Dim PreComplete As Boolean
```

```
PreComplete = odaClient.Status.PresetComplete
```

## StatusLog Method

Returns an array of status log entries that appear as messages within the status log on the ODA 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 = odaClient.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 = odaClient.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 = odaClient.Status.TLSSettling
```

**VB 6.0 Syntax**

```
Dim TLSIsSettling As Boolean
TLSIsSettling = odaClient.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 = odaClient.Status.TriggerComplete
```

**VB 6.0 Syntax**

```
Dim TrigComplete As Boolean
```

```
TrigComplete = odaClient.Status.TriggerComplete
```

**ValidNormalization Property**

Sets the value to True when the ODA settings fall within the current normalization boundaries.

**Read-Only Property**

Boolean - True when current ODA settings fall within the current normalization boundaries.

**VB.NET Syntax**

```
Dim ValNorm As Boolean
```

```
ValNorm = odaClient.Status.ValidNormalization
```

**VB 6.0 Syntax**

```
Dim ValNorm As Boolean
```

```
ValNorm = odaClient.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

```
odaClient.System.LaserState _  
(Instrument.Objects.ODACCommon.enumOnOff.eOn)
```

#### VB 6.0 Syntax

```
odaClient.System.LaserState (enumOnOff_eOn)
```

### Preset Method

Sets the instrument to defined setup conditions. Preset can only be called when the ODA is idle.

#### VB.NET Syntax

```
odaClient.System.Preset()
```

#### VB 6.0 Syntax

```
odaClient.System.Preset
```

## System - Identity Commands

### CompanyName Property

Returns Agilent Technologies as the company name of the 86038A ODA.

#### Property Value

String - Company name.

#### VB.NET Syntax

```
Dim CompName As String
```

```
CompName = odaClient.System.Identity.CompanyName
```

#### VB 6.0 Syntax

```
Dim CompName As String
```

```
CompName = odaClient.System.Identity.CompanyName
```

### ProductId Property

Returns the product number, 86038A.

#### Property Value

String - Product number

#### VB.NET Syntax

```
Dim ProdID As String
```

```
ProdID = odaClient.System.Identity.ProductID
```

#### VB 6.0 Syntax

```
Dim ProdID As String
```

```
ProdID = odaClient.System.Identity.ProductID
```

### ProductName Property

Returns the product name.

#### Property Value

String - Product name.



**VB.NET Syntax**

```
Dim ProdName As String
```

```
ProdName = odaClient.System.Identity.ProductName
```

**VB 6.0 Syntax**

```
Dim ProdName As String
```

```
ProdName = odaClient.System.Identity.ProductName
```

**ProductVersion Property**

Returns the version of the ODA system. The major and minor version numbers of the remote client and the ODA system should match.

**Property Value**

String - Software version of ODA system software.

**VB.NET Syntax**

```
Dim ProdVersion As String
```

```
ProdVersion = odaClient.System.Identity.ProductVersion
```

**VB 6.0 Syntax**

```
Dim ProdVersion As String
```

```
ProdVersion = odaClient.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 ODA 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 = odaClient.System.Identity.RemotingVersion
```

**VB.6.0 Syntax**

```
Dim RemVersion As String
```

```
RemVersion = odaClient.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

```
odaClient.TLS.ForceSettling()
```

#### VB 6.0 Syntax

```
odaClient.TLS.ForceSettling
```

### HighPower Read Only Property

Returns the power setting of TLS Output 2.

#### Property Value

Double - the TLS power setting.

#### VB.NET Syntax

```
Dim Power As Double
```

```
Power = odaClient.TLS.HighPower
```

#### VB.6.0 Syntax

```
Dim Power As Double
```

```
Power = odaClient.TLS.HighPower
```





# 7

## Troubleshooting and Upgrades

Introduction.....	254
Troubleshooting Startup Problems .....	255
Troubleshooting GPIB Communications Problems....	257
Troubleshooting Problems with Remote Programming	261
Troubleshooting Sweep and Normalization Problems .	262
Troubleshooting Incorrect Data .....	263
Troubleshooting the TLS.....	265
Solving Measurement Problems .....	266
Software Upgrade Procedure.....	270
Hard Drive Backup and Recovery .....	271
Cleaning Connections for Accurate Measurements ..	272
Returning the Instrument for Service .....	282

# Introduction

This chapter begins by explaining how to troubleshoot basic problems with the ODA 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 initialisms are used:

- ODA: Optical Dispersion Analyzer (Agilent 86038A)
- PNA: Performance Network Analyzer
- TLS: Tunable Laser Source
- MWM: Multi-wavelength Meter

# Troubleshooting Startup Problems

## If the ODA does not boot up to MS Windows 2000

### 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: Image

The Agilent ODA has the ability to backup and restore a hard drive image, in case of operating system error. If the ODA is unable to boot into the Windows 2000 option, please [refer to "Hard Drive Backup and Recovery" on page 271](#). If this does not solve the problem, continue with the next step.

### 3 Replace the PNA

If the system does not restore from the hard disk recovery image, the PNA will need to be replaced.

## If the ODA Application does not automatically launch

The ODA Application should launch automatically after Windows 2000 boots. If not, the application can be launched from the ODA shortcut on the desktop, or from Start > Programs > Agilent Optical Dispersion Analyzer > Agilent 86038A.

If you are able to successfully launch the ODA 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\Optical Dispersion Analyzer\Config, and double click on the "toggle ODA 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.

Similarly, ensure that the "toggle ODA loading autostart.vbs" link is enabled.

## If the ODA 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 257.](#)



# Troubleshooting GPIB Communications Problems

## 1. Check the ODA Status Log

If the ODA application is running, check the ODA Status Log for any warnings or errors. Press Status Log to view a list of status messages.

## 2. Ensure that GPIB cables are configured correctly.

Check the GPIB cables, they should match [Figure 2 on page 19](#).

## 3. Check the system GPIB Addresses.

The system is configured to work with set GPIB addresses for the tunable laser source and multi-wavelength meter (when applicable). If the GPIB addresses for these instruments have been changed, the ODA application will not be able to communicate with them.

The 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 ODA 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 ODA software, but must be altered on the individual instruments.

Table 11

Instrument	Model #	GPIB Address	Path to set GPIB Address
Lightwave Measurement System with Tunable Laser Source Module	8164B / 81640B or 81600B	20	Config > GPIB Address
Multi-Wavelength Meter	86122A	10	Setup > GPIB Setup

### 3. Check the tunable laser source and multi-wavelength meter firmware revision

The ODA Application has been designed and tested to work with the 8164B Lightwave Measurement System and 81640B or 81600B Tunable Laser Source. An 86122A Multi-Wavelength Meter may also be added to an ODA to improve the wavelength accuracy of the measurements. There is a minimum firmware revision required to work with the ODA application. If the instrument is not the correct model number, or does not meet the minimum firmware revision requirements, the ODA will not work with the instrument. Check the instrument's firmware revisions by following the steps below.

Table 12

Instrument	Model #	Min F/W Rev.	To check firmware revision
Lightwave Measurement System	8164A or 8164B	V4.02	Local > Config > About Mainframe. The mainframe firmware revision information is on the bottom line.
Tunable Laser Source Module	81640B	V4.01	Local > Config > About Modules > Slot 0 > Enter.
	81600B #160 81600B #200	V4.07 V4.07	
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 ODA in the "Additional Drivers" folder. The most recent versions can also be found on the Agilent web site, under [www.agilent.com/comms/octfirmware](http://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 ODA controller itself can be used. To use the ODA controller, first exit the ODA application (File>Exit>"agilentoda">Yes).

#### **4. Ensure that the GPIB setting in the PNA software is set to talker / listener**

The ODA 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 ODA 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 ODA 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 ODA Application (File > Exit > "agilentoda" > Yes). 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 ODA application from the ODA shortcut on the desktop, or from Start Menu > Programs > Agilent Optical Dispersion Analyzer > Agilent 86038A.

#### **5. Ensure that NI GPIB driver is enabled, Agilent GPIB driver is disabled.**

The ODA 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 ODA. In order for the ODA 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 Agilent GPIB TNT Interface. This is the correct configuration for the ODA 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 ODA 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.

## 6. Reboot the TLS

- 1 Exit the ODA application (File > Exit > “agilentoda” > 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 ODA software.

## 7. Reboot the PNA

- 1 Exit the ODA application (File > Exit > “agilentoda” > Yes).
- 2 Ensure that all of the instruments are at the proper GPIB addresses.
- 3 Reboot the PNA.

## 8. Swap GPIB Cables

If the ODA 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 146.

# Troubleshooting Sweep and Normalization Problems

The ODA 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 PNA.

### If the Modulator Bias does not complete

The Modulator Bias will not complete if the input power detected is too low.

- 1 Ensure that the test set power cord is plugged in.
- 2 Ensure that the proper port is selected and a clean patch fiber is connected to the selected port. Use the Real Time Port Monitor to check the power at the input port. See [“1. Check the Modulator Bias”](#).
- 3 Isolate the problem to the TLS or MWM. Refer to [“Test the High output on the TLS” on page 265](#).

### If the Wavelength Correction does not complete

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, exit the ODA software (File > Exit > “agilentoda” > Yes) and restart. The ODA software will detect the MWM on power up.
- 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.
- 5 Isolate the problem to the TLS or MWM. Refer to [“Test the High output on the TLS” on page 265](#), and test the Low SSE output on the TLS.

# Troubleshooting Incorrect Data

## If wavelengths are incorrect, or measurements 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 265](#).

## If the Stepped GD and Gain/Loss data looks incorrect

### NOTE

If the ODA-measured-power for your device is different than expected, the first step is to ensure that the fiber connections are clean and secure, refer to [“Cleaning Connections for Accurate Measurements” on page 272](#). Make sure that the test set power cable is plugged in. Also, make sure that the insulated patch cord is connected from the High Output of the TLS to the TLS Input on the test set, and that the green indicator light for the High Output of the TLS is on. If the TLS high power output indicator light is not on, realign the TLS. Please refer to [“Restart the TLS and realign the TLS module” on page 265](#).

- 1 Ensure that the proper port is selected and a clean patch fiber is connected to the selected port.
- 2 If the data still looks incorrect, isolate the problem to the TLS or MWM. Refer to [“Test the High output on the TLS” on page 265](#).
- 3 If the data still looks incorrect, refer to [“Choosing Setup Parameters” on page 43](#).

## If Stepped GD, Gain/Loss, DGD, or PDL looks incorrect

You should first suspect a problem with the polarization scanner inside the test set. Follow the steps below:

- 1 While the stepped measurement is running, listen carefully to see if the Polarization Scanner turns on.
  - If it doesn't turn on, try powering down the ODA. Restart and try the measurement again.
  - If the Polarization Scanner still fails to start, the Test Set needs to be replaced.
- 2 You can also check the results of the PDL measurement.
  - If the PDL values measured on the normalization patchcord are close to the noise floor (around 0.1dB), the polarization scanner is OK.
  - If the PDL values are extremely low (around 0.01dB), the polarization scanner

may be faulty. Some ODAs have very low PDL. Test a device with known polarization dependent loss, greater than 0.2 dB PDL, to isolate a polarization scanner failure. If the PDL still reads extremely low, the polarization scanner is faulty and the test set will need to be replaced.

- 3 If the data still looks incorrect, refer to [“If the Stepped GD and Gain/Loss data looks incorrect” on page 263](#).

## If the SWEPT data looks incorrect

### 1. Check the Modulator Bias

Connect a patch fiber and run the Modulator Bias measurement. Without the proper value, optimal power may not be reaching the PNA receiver. Connect a mouse and click on Measure->Optimize Optical Modulator Bias (F8) and then run your measurement again. After the Modulator Bias routine finishes, open the Real-Time Port Monitor by clicking on System->Real Time Port Monitor (F12). If the Gain/Loss values are still low, there may be a problem with the Test Set. With a short patch cord, Gain/Loss should be approximately -16 dB.

### 2. Check the Trigger Line

Ensure that the trigger line is connected between the TLS Trigger Out and the Test Set TLS Trigger Input.

#### 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 267](#).



# Troubleshooting the TLS

- If you suspect faulty TLS wavelength performance, refer to “Restart the TLS and realign the TLS module” on page 265.
- If you suspect faulty TLS power performance, refer to “Test the High output on the TLS” on page 265.

## Restart the TLS and realign the TLS module

- 1 Exit out of the ODA application (File > Exit > “agilentoda” > 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 ODA application by double clicking on the ODA desktop icon, or navigating to Start > Programs > Agilent Optical Dispersion Analyzer > Agilent 86038A.
- 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 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 module.
  - 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, and retest the system. If the TLS power output is accurate and the test is reading low, it may need to be replaced.
  - If the TLS output power is off by more than 0.5 dB, or there is no output signal, 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.

# Solving Measurement Problems

## If problems occur testing narrowband devices

The five-term Sellmeier fit is generally inappropriate for very narrowband devices, such as Fabry-Perot filters. The result can be roughness in the fitted curve and multiple values of lambda zero. This is not a problem when testing fibers.

## If results of different curve fit models do not agree

The three curve fit types differ in their ability to conform to the actual dispersion curve of the tested fiber. As a result, a variation on the order of a few hundredths to a few tenths of a nanometer can be observed in the value of a lambda zero. The three-term Sellmeier curve fit tends to yield a lower lambda zero value than the other two curve fitting algorithms.

Although the five-term Sellmeier curve fit has greater ability to fit the measured data, it is also more easily affected by noise. Therefore, the quadratic fit will generally give a more repeatable value of lambda zero. For most DSF, the mean values of lambda zero returned by the five-term and quadratic curve fits are in good agreement while the three-term fit yields a lower value.

## If fitted curve does not follow the actual data

The shape of the curve fitted to the relative group delay data depends upon the device under test and the type of fit selected. Conditions of high DUT insertion loss, low DUT dispersion or narrow wavelength range can also affect the quality of the fit. The system will always try to fit the selected curve to the data. Because the 5-term Sellmeier curve fit has more freedom to follow the shape of the data, it may display multiple zero dispersion wavelengths under some conditions.

## 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 an 80 minute warm-up. During that time, measurements will show a lot of drift. How the instrument is performing in its environment can be observed by attaching a patch fiber and running the Real-Time Port Monitor to check the power stability. To run the Real-Time Port Monitor, attach a mouse and click on System->Real Time Port Monitor (F12).

To achieve the most accurate GD, any DGD component may need to be removed. To measure GD with the DGD component removed, normalize and measure for DGD (GD/CD + DGD/PDL, 70Hz IFBW, 201 Samples/Point). The GD measured in this mode is more accurate because it is given at the mean of the DGD, thus the DGD component is removed. To check how significant the DGD component is, view the GD with 'show DGD component'. If the DGD component is negligible, the measurement can be made faster without measuring DGD (GD/CD, 30Hz IFBW, 21 or 51 Samples/Point).

To achieve the lowest noise GD, when measuring DGD is not a consideration, use the largest modulation frequency possible (2.5GHz, unless limited by other considerations) with 30Hz IFBW and 21 or 51 Samples/Point. Smoothing the GD trace will further lower the GD noise. When using smoothing, a larger number of wavelength steps also lowers the noise. The drift effects of the measurement setup may be more significant than the GD noise, so a faster measurement time (fewer Samples/Point, fewer wavelength steps) may give more accurate results.

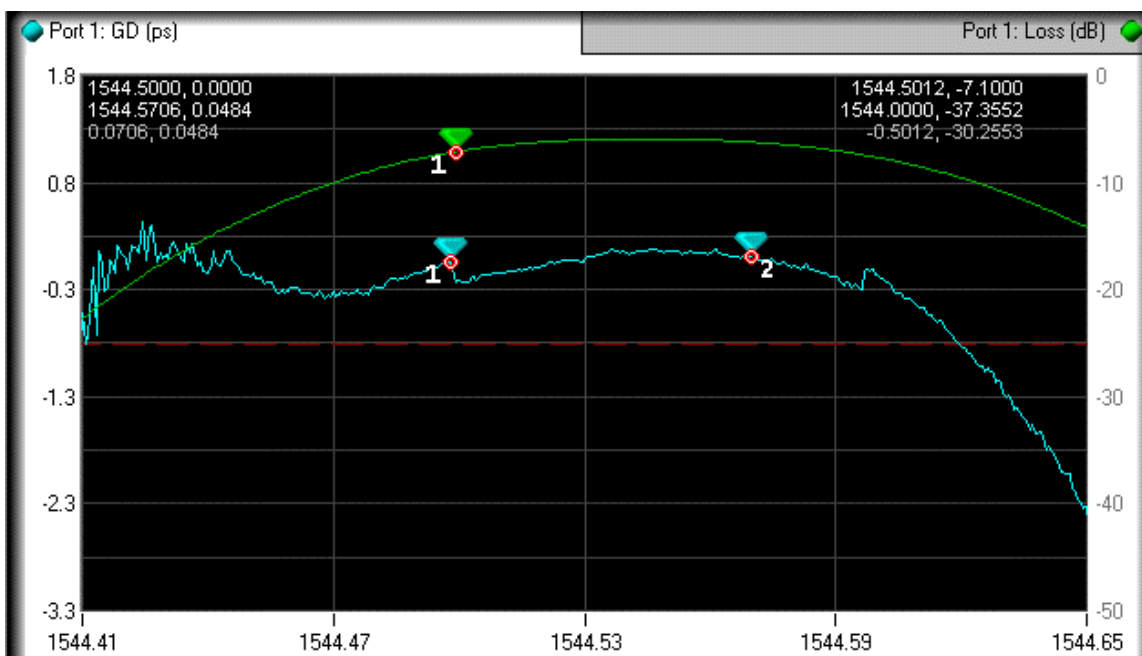
### **If the Group Delay measurement shows a sudden large offset**

Sudden offsets in group delay are typically the results of a phase wrap [refer to “Relationship of setup parameters to phase wrapping” on page 45](#). 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 you see errors in Group Delay measurements but no error is visible in Gain/Loss

This is most likely the result of Gain Stage switching. The ODA is based on a Performance Network Analyzer which uses Step Gain values to keep the input signal in the most linear region of the 16 bit ADC. The Step Gain is selected based on the power level detected by the Network Analyzer. While the Network Analyzer has corrections for the Gain/Loss measurements, no such corrections exist for phase. As the input power changes and causes a Gain Stage switch, a 200 fs jump in Group Delay may be observed (50fs is typical). The TLS power level can be lowered to eliminate this artifact. Alternatively, the TLS power level can be raised to move the artifact out of the region of interest. Refer to "Laser Setup" on page 105 for more information.



**NOTE**

For more information, refer to Chapter 8, "Specifications and Regulatory Information."

## 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 that makes sense for the device under test.

Another factor to consider is the step size used for normalization and the step size used in the measurement may need to be the same. Also, make sure that you have a valid normalization.

A common problem that leads to GD ripple is dirty fiber connections. See [“Example of a Dirty Connector”](#).

Refer to [“Relationship of setup parameters to ripple amplitude accuracy”](#) on page 48 for more information.

# Software Upgrade Procedure

The 86038A is built on the 8801A performance network analyzer (PNA). Firmware for the ODA is designed to work with a specific version of the PNA firmware.

To Upgrade the firmware on the ODA:

- 1** Check the ODA website for firmware upgrades: [www.agilent.com/comms/oct](http://www.agilent.com/comms/oct).
- 2** Download the latest software revision. For some upgrades both the PNA and ODA firmware will need to be upgraded. Download the install packages for both the ODA and PNA from the *same table row*.
- 3** Run the PNA Installer (when applicable). Versions higher than A.02.51 are not currently supported.
  - The installer can be launched by double-clicking on the executable.
  - Select **Easy Connection** instead of More Secure.
  - For all other prompts, select **Next** or **Continue**.
- 4** Run the ODA Installer.
  - The installer can be launched by double-clicking on the executable.
  - Select **Next** or **Continue** for all prompts.
- 5** (Optional) Download the ODA Remote Client Installer. Install this on computers that will be used to remotely control the ODA. *Do not install the Remote Client on the ODA instrument itself.* Refer to Chapter 5, "Remote Operation."

# Hard Drive Backup and Recovery

The following procedure is used to back up or restore the operating system on an Agilent Technologies instrument/system running on a MS Windows platform.

## NOTE

MS Windows based instruments must have a keyboard attached prior to starting this procedure.

**1** Restart the instrument/system controller. The Agilent ODA offers three boot up options:

- 1. Agilent Technologies 86038A
- 2. Operating System Recovery Console
- 3. Backup and Recovery Options.

**2** *Quickly* select option 3. (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.

1. Recover Factory's Backup 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.

2. Create User Backup Image

This process creates a backup image of your system. This process should be done if you have modified the network parameters and/or drivers.

3. Recover User Backup Image

This process restores your system to the user-created backup-image. A recovery should be done if MS Windows is corrupted or if you have been instructed to do so by an Agilent service engineer.

4. Exit

Exits the utility.

**4** Enter the number associated with the desired action. For example, enter 2 if you want to create a user backup image of your system.

**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/system controller.

# Cleaning Connections for Accurate Measurements

Today, advances in measurement capabilities make connectors and connection techniques more important than ever. Damage to the connectors on calibration and verification devices, test ports, cables, and other devices can degrade measurement accuracy and damage instruments. Replacing a damaged connector can cost thousands of dollars, not to mention lost time! This expense can be avoided by observing the simple precautions presented in this book. This book also contains a brief list of tips for caring for electrical connectors.

## Choosing the Right Connector

A critical but often overlooked factor in making a good lightwave measurement is the selection of the fiber-optic connector. The differences in connector types are mainly in the mechanical assembly that holds the ferrule in position against another identical ferrule. Connectors also vary in the polish, curve, and concentricity of the core within the cladding. Mating one style of cable to another requires an adapter. Agilent Technologies offers adapters for most instruments to allow testing with many different cables. [Figure 19 on page 273](#) shows the basic components of a typical connectors.

The system tolerance for reflection and insertion loss must be known when selecting a connector from the wide variety of currently available connectors. Some items to consider when selecting a connector are:

- How much insertion loss can be allowed?
- Will the connector need to make multiple connections? Some connectors are better than others, and some are very poor for making repeated connections.
- What is the reflection tolerance? Can the system take reflection degradation?
- Is an instrument-grade connector with a precision core alignment required?
- Is repeatability tolerance for reflection and loss important? Do your specifications take repeatability uncertainty into account?
- Will a connector degrade the return loss too much, or will a fusion splice be required? For example, many DFB lasers cannot operate with reflections from connectors. Often as much as 90 dB isolation is needed.



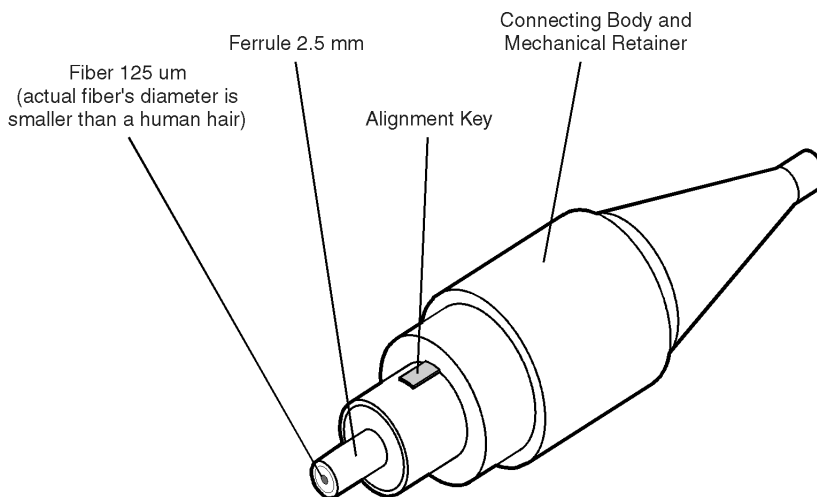


Figure 19 Basic components of a connector.

Over the last few years, the FC/PC style connector has emerged as the most popular connector for fiber-optic applications. While not the highest performing connector, it represents a good compromise between performance, reliability, and cost. If properly maintained and cleaned, this connector can withstand many repeated connections.

However, many instrument specifications require tighter tolerances than most connectors, including the FC/PC style, can deliver. These instruments cannot tolerate connectors with the large non-concentricities of the fiber common with ceramic style ferrules. When tighter alignment is required, Agilent Technologies instruments typically use a connector such as the Diamond HMS-10, which has concentric tolerances within a few tenths of a micron. Agilent Technologies then uses a special universal adapter, which allows other cable types to mate with this precision connector. See [Figure 20](#).



Figure 20 Universal adapters to Diamond HMS-10.

The HMS-10 encases the fiber within a soft nickel silver (Cu/Ni/Zn) center which is surrounded by a tough tungsten carbide casing, as shown in [Figure 21](#).

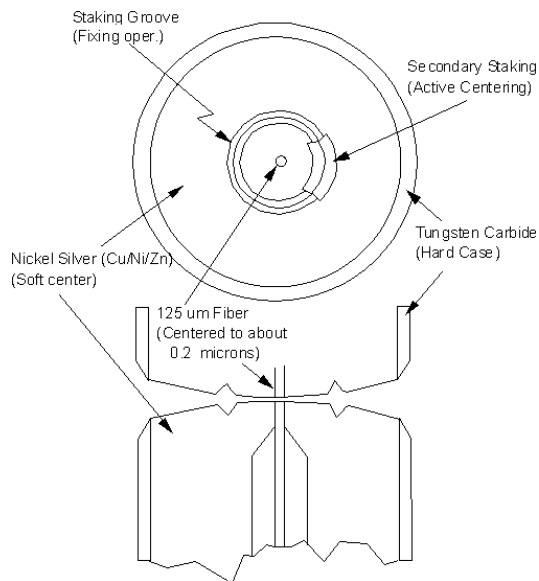


Figure 21 Cross-section of the Diamond HMS-10 connector.

The nickel silver allows an active centering process that permits the glass fiber to be moved to the desired position. This process first stakes the soft nickel silver to fix the fiber in a near-center location, then uses a post-active staking to shift the fiber into the desired position within 0.2  $\mu\text{m}$ . This process, plus the keyed axis, allows very precise core-to-core alignments. This connector is found on most Agilent Technologies lightwave instruments.

The soft core, while allowing precise centering, is also the chief liability of the connector. The soft material is easily damaged. Care must be taken to minimize excessive scratching and wear. While minor wear is not a problem if the glass face is not affected, scratches or grit can cause the glass fiber to move out of alignment. Also, if unkeyed connectors are used, the nickel silver can be pushed onto the glass surface. Scratches, fiber movement, or glass contamination will cause loss of signal and increased reflections, resulting in poor return loss.

## Inspecting Connectors

Because fiber-optic connectors are susceptible to damage that is not immediately obvious to the naked eye, poor measurements result without the user being aware. Microscopic examination and return loss measurements are the best way to ensure good measurements. Good cleaning practices can help ensure that optimum connector performance is maintained. With glass-to-glass interfaces, any degradation of a ferrule or the end of the fiber, any stray particles, or finger oil can have a significant effect on connector performance. Where many repeat connections are required, use of a connector saver or patch cable is recommended.

Figure 22 shows the end of a clean fiber-optic cable. The dark circle in the center of the micrograph is the fiber's 125  $\mu\text{m}$  core and cladding which carries the light. The surrounding area is the soft nickel-silver ferrule.

Figure 23 shows a dirty fiber end from neglect or perhaps improper cleaning. Material is smeared and ground into the end of the fiber causing light scattering and poor reflection. Not only is the precision polish lost, but this action can grind off the glass face and destroy the connector.

Figure 24 shows physical damage to the glass fiber end caused by either repeated connections made without removing loose particles or using improper cleaning tools. When severe, the damage of one connector end can be transferred to another good connector endface that comes in contact with the damaged one. Periodic checks of fiber ends, and replacing connecting cables after many connections is a wise practice.

The cure for these problems is disciplined connector care as described in the following list and in "Cleaning Connectors" on page 279.

Use the following guidelines to achieve the best possible performance when making measurements on a fiber-optic system:

- Never use metal or sharp objects to clean a connector and never scrape the connector.
- Avoid matching gel and oils.

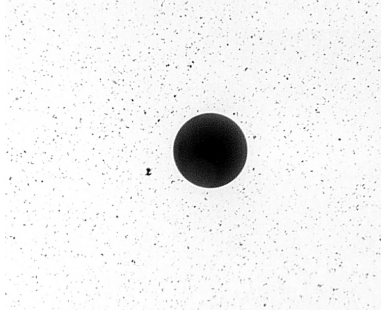


Figure 22 Clean, problem-free fiber end and ferrule.

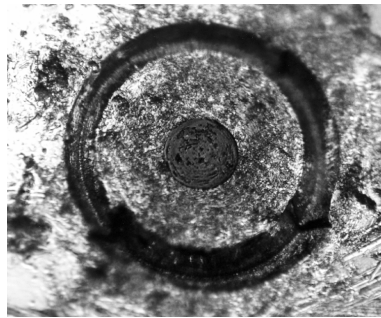


Figure 23 Dirty fiber end and ferrule from poor cleaning.

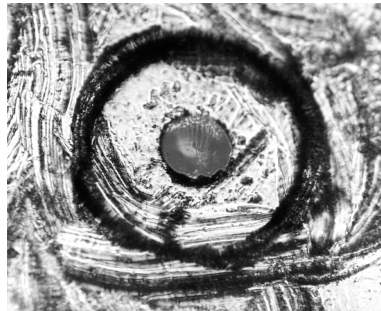


Figure 24 Damage from improper cleaning.

While these often work well on first insertion, they are great dirt magnets. The oil or gel grabs and holds grit that is then ground into the end of the fiber. Also, some early gels were designed for use with the FC, non-contacting connectors, using small glass spheres. When used with contacting connectors, these glass balls can scratch and pit the fiber. If an index matching gel or oil must be used, apply it to a freshly cleaned connector, make the measurement, and then immediately clean it off. Never use a gel for longer-term connections and never use it to improve a damaged connector. The gel can mask the extent of damage and continued use of a damaged fiber can transfer damage to the instrument.

- When inserting a fiber-optic cable into a connector, gently insert it in as straight a line as possible. Tipping and inserting at an angle can scrape material off the inside of the connector or even break the inside sleeve of connectors made with ceramic material.
- When inserting a fiber-optic connector into a connector, make sure that the fiber end does not touch the outside of the mating connector or adapter.
- Avoid over tightening connections.

Unlike common electrical connections, tighter is not better. The purpose of the connector is to bring two fiber ends together. Once they touch, tightening only causes a greater force to be applied to the delicate fibers. With connectors that have a convex fiber end, the end can be pushed off-axis resulting in misalignment and excessive return loss. Many measurements are actually improved by backing off the connector pressure. Also, if a piece of grit does happen to get by the cleaning procedure, the tighter connection is more likely to damage the glass. Tighten the connectors just until the two fibers touch.

- Keep connectors covered when not in use.
- Use fusion splices on the more permanent critical nodes. Choose the best connector possible. Replace connecting cables regularly. Frequently measure the return loss of the connector to check for degradation, and clean every connector, every time.

All connectors should be treated like the high-quality lens of a good camera. The weak link in instrument and system reliability is often the inappropriate use and care of the connector. Because current connectors are so easy to use, there tends to be reduced vigilance in connector care and cleaning. It takes only one missed cleaning for a piece of grit to permanently damage the glass and ruin the connector.

### *Measuring insertion loss and return loss*

Consistent measurements with your lightwave equipment are a good indication that you have good connections. Since return loss and insertion loss are key factors in determining optical connector performance they can be used to determine connector degradation. A smooth, polished fiber end should produce a good return-loss measurement. The quality of the polish establishes the difference between the "PC" (physical contact) and the "Super PC" connectors. Most connectors today are physical contact which make glass-to-glass connections, therefore it is critical that the area around the glass core be clean and free of scratches. Although the major area of a connector, excluding the glass, may show scratches and wear, if the glass has maintained its polished smoothness, the connector can still provide a good low level return loss connection.

If you test your cables and accessories for insertion loss and return loss upon receipt, and retain the measured data for comparison, you will be able to tell in the future if any degradation has occurred. Typical values are less than 0.5 dB of loss, and sometimes as little as 0.1 dB of loss with high performance connectors. Return loss is a measure of reflection: the less reflection the better (the larger the return loss, the smaller the reflection). The best physically contacting connectors have return losses better than 50 dB, although 30 to 40 dB is more common.

### *Visual inspection of fiber ends*

Visual inspection of fiber ends can be helpful. Contamination or imperfections on the cable end face can be detected as well as cracks or chips in the fiber itself. Use a microscope (100X to 200X magnification) to inspect the entire end face for contamination, raised metal, or dents in the metal as well as any other imperfections. Inspect the fiber for cracks and chips. Visible imperfections not touching the fiber core may not affect performance (unless the imperfections keep the fibers from contacting).

#### **WARNING**

**Always remove both ends of fiber-optic cables from any instrument, system, or device before visually inspecting the fiber ends. Disable all optical sources before disconnecting fiber-optic cables. Failure to do so may result in permanent injury to your eyes.**

---

## Cleaning Connectors

The procedures in this section provide the proper steps for cleaning fiber-optic cables and Agilent Technologies universal adapters. The initial cleaning, using the alcohol as a solvent, gently removes any grit and oil. If a caked-on layer of material is still present, (this can happen if the beryllium-copper sides of the ferrule retainer get scraped and deposited on the end of the fiber during insertion of the cable), a second cleaning should be performed. It is not uncommon for a cable or connector to require more than one cleaning.

### CAUTION

Agilent Technologies strongly recommends that index matching compounds *not* be applied to their instruments and accessories. Some compounds, such as gels, may be difficult to remove and can contain damaging particulates. If you think the use of such compounds is necessary, refer to the compound manufacturer for information on application and cleaning procedures.

Table 13 Cleaning Accessories

Item	Agilent Part Number
Cotton swabs	8520-0023
Small foam swabs	9300-1223

Table 14 Dust Caps Provided with Lightwave Instruments

Item	Agilent Part Number
Laser shutter cap	08145-64521
FC/PC dust cap	08154-44102
Biconic dust cap	08154-44105
ST dust cover	1401-0291

*To clean a non-lensed connector*

### CAUTION

Do not use any type of foam swab to clean optical fiber ends. Foam swabs can leave filmy deposits on fiber ends that can degrade performance.

- 1 Apply pure isopropyl alcohol to a clean lint-free cotton swab or lens paper.

Cotton swabs can be used as long as no cotton fibers remain on the fiber end after cleaning.

- 2 Clean the ferrules and other parts of the connector while avoiding the end of the fiber.
- 3 Apply isopropyl alcohol to a new clean lint-free cotton swab or lens paper.
- 4 Clean the fiber end with the swab or lens paper.

Do not scrub during this initial cleaning because grit can be caught in the swab and become a gouging element.

- 5 Immediately dry the fiber end with a clean, dry, lint-free cotton swab or lens paper.
- 6 Blow across the connector end face from a distance of 6 to 8 inches using filtered, dry, compressed air. Aim the compressed air at a shallow angle to the fiber end face.

Nitrogen gas or compressed dust remover can also be used.

**CAUTION**

Do not shake, tip, or invert compressed air canisters, because this releases particles in the can into the air. Refer to instructions provided on the compressed air canister.

- 7 As soon as the connector is dry, connect or cover it for later use.

If the performance, after the initial cleaning, seems poor try cleaning the connector again. Often a second cleaning will restore proper performance. The second cleaning should be more arduous with a scrubbing action.

### *To clean an adapter*

The fiber-optic input and output connectors on many Agilent Technologies instruments employ a universal adapter such as those shown in the following picture. These adapters allow you to connect the instrument to different types of fiber-optic cables.



Figure 25 Universal adapters.



**1** Apply isopropyl alcohol to a clean foam swab.

Cotton swabs can be used as long as no cotton fibers remain after cleaning. The foam swabs listed in this section's introduction are small enough to fit into adapters.

Although foam swabs can leave filmy deposits, these deposits are very thin, and the risk of other contamination buildup on the inside of adapters greatly outweighs the risk of contamination by foam swabs.

**2** Clean the adapter with the foam swab.**3** Dry the inside of the adapter with a clean, dry, foam swab.**4** Blow through the adapter using filtered, dry, compressed air.

Nitrogen gas or compressed dust remover can also be used. Do not shake, tip, or invert compressed air canisters, because this releases particles in the can into the air. Refer to instructions provided on the compressed air canister.

# 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 .....	284
Angled Connector Specifications .....	285
Specifications .....	286
Regulatory Information .....	290
Declaration of Conformity.....	291

# Introduction

This chapter contains specifications and characteristics for the Agilent 86038A optical dispersion analyzer. For specifications specific to the 81640 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.

The distinction between specifications and characteristics is described as follows:

- **Specifications** describe warranted system performance for the temperature range of  $25\text{ }^{\circ}\text{C} \pm 5\text{ }^{\circ}\text{K}$  after 90 minutes of warm-up.
- **Characteristics** provide useful, but nonwarranted information about the functions and performance of the instrument. *Characteristics are printed in italics.* Characteristics represent any one of the following:
  - The average or median value based on a significant number of units.
  - A tolerance interval or proportion of performance distribution derived from the measurement of a significant number of units. The proportion is greater than 80%.
  - A parameter with a quantity that is not subject to variations.
  - Specifications and characteristics apply over the following instrument operating conditions:
    - DUT pigtail fibers thermally protected, fixed, and settled for 3 minutes before test.
    - Measurements over the 1520 nm to 1610 nm wavelength range of the tunable laser source (TLS) which has Side Mode Suppression Ratio (SMSR) > 40 dB.
    - The instruments temperature has been stabilized after 90 minutes of continuous operation.

# Angled Connector Specifications

Specifications require an angled connector at the source output and at optical input ports 1 and 2 of 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, you can only use a cable with angled connectors to connect to the TLS input on the ODA. The ODA TLS input requires angled connectors. Do not use a cable with a flat connector on either the angled ODA 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 are valid in the 1520 - 1610 nm wavelength range.**

Table 15 Group Delay and Differential Group Delay Measurement

Stepped Relative Group Delay Repeatability <sup>a,b,c</sup>	
0 dB level	±20 fs
< 0 dB to -10 dB level (characteristic)	±50 fs
< -10 dB to -20 dB level (characteristic)	±150 fs
< -20 dB to -30 dB level (characteristic)	±500 fs
< -30 dB to -40 dB level (characteristic)	±5 ps
Stepped Differential Group Delay Accuracy (characteristic) <sup>a,d</sup>	± 90 fs ±9% of Measured DGD
Stepped Differential Group Delay Repeatability <sup>a,c,d</sup>	± 20 fs ±7% of Measured DGD
Group Delay Time Resolution	1 fs
Modulation Frequency Range	5 MHz to 2.5 GHz

<sup>a</sup> At the same temperature as the normalization temperature ±0.5°K.

<sup>b</sup> Measured using a 2.2 meter thermally isolated SMF patch cord, Modulation Frequency = 2 GHz, IFBW = 30 Hz, 51 phase samples per trace point, 10nm wavelength step size, receiver input power -12 dBm to -18 dBm across the wavelength range (TLS output approximately -5 dBm). Normalization performed with Phase Dynamic Accuracy selected. The level is relative to a 0.5m SMF patch cord normalization device.

<sup>c</sup> Repeatability is defined as the worse (plus or minus) standard deviation over the wavelength range 1520 nm to 1610 nm from 10 sweeps.

<sup>d</sup> Verified with two characterized standards that have 1.45 ps and 0.19 ps DGD in the operating wavelength range, Modulation Frequency = 2 GHz, IFBW = 70 Hz, 801 phase samples per trace point, 10 nm wavelength step size, receiver input power -12 dBm to -18 dBm across the wavelength range (TLS output approximately -5 dBm). Normalization performed with Phase Dynamic Accuracy selected.

Table 16 Swept Relative Group Delay Measurement

<i>Swept Relative Group Delay Offset (characteristic) <sup>a,b</sup></i>	<i>±250 fs</i>
<i>Swept Relative Group Delay Repeatability (characteristic) <sup>b,c</sup></i>	<i>±25 fs</i>

<sup>a</sup> Offset is defined as the difference in relative group delay between stepped and swept measurements.

<sup>b</sup> Measured using a 2.2meter thermally isolated SMF patch cord, Modulation Frequency = 2 GHz, IFBW = 30Hz, receiver input power -12 dBm to -18 dBm across the wavelength range (TLS output approximately -5 dBm). Swept measurements performed with 51 trace points, stepped measurements performed with 1.8 nm step size. Normalization performed with Phase Dynamic Accuracy selected.

<sup>c</sup> Repeatability is defined as the worse (plus or minus) standard deviation over the wavelength range 1520 nm to 1610 nm from 10 sweeps.

Table 17 Amplitude Measurement

<i>Polarization Dependent Loss Accuracy (characteristic)</i> <sup>a</sup>	$\pm 0.25$ dB
<i>System Dynamic Range (characteristic)</i> <sup>b,c</sup>	50 dB
<i>Gain/Loss accuracy (characteristic)</i> <sup>d</sup>	$\pm 0.25$ dB

<sup>a</sup> Verified with a PDL standard that has 0.46 dB PDL in the operating wavelength range, Modulation frequency = 2 GHz, IFBW = 70 Hz, with 201 samples per trace point, 10nm step size, receiver input power -12 dBm to -18 dBm across the wavelength range (TLS output approximately -5 dBm). Normalization performed with Phase Dynamic Accuracy selected.

<sup>b</sup> TLS power set at +2 dBm.

<sup>c</sup> Measured at 1550 nm, Modulation Frequency = 2 GHz, 801 phase samples per trace point, IFBW = 70 Hz, 10 averages.

<sup>d</sup> Excludes connector repeatability.

Table 18 Wavelength Measurement

Operating Wavelength Range with Agilent 81600B#200	1440 nm to 1640 nm
with Agilent 81600B#160 or 81640B TLS	1495 nm to 1640 nm
with Agilent 81600B#150	1450 nm to 1590 nm
with Agilent 81600B#140	1370 nm to 1495 nm
with Agilent 81600B#130 <sup>d</sup>	1260 nm to 1375 nm
with Agilent 81640A TLS	1510 nm to 1640 nm
Minimum Wavelength Step Size	0.1 pm
Absolute Wavelength Accuracy <sup>a,b</sup> <i>Stepped mode with Agilent 86122A (characteristic)</i>	$\pm 1$ pm
Stepped mode without Agilent 86122A and with 81600B or 81640B	$\pm 10$ pm
Stepped mode without Agilent 86122A and with 81640A	$\pm 15$ pm
Relative Wavelength Accuracy <sup>a,c</sup> Stepped mode without Agilent 86122A and with 81600B or 81640B	$\pm 5$ pm
<i>Stepped mode without Agilent 86122A and with 81600B or 81640B (characteristic)</i>	$\pm 2$ pm
Stepped mode without Agilent 86122A and with 81640A	$\pm 7$ pm
<i>Stepped mode without Agilent 86122A and with 81640A (characteristic)</i>	$\pm 3$ pm

<sup>a</sup> Valid for one month and within a  $\pm 4.4$  K temperature range after automatic wavelength zeroing. Measured with wavelength meter based on wavelength in vacuum.

<sup>b</sup> Refer to Agilent 81640A and 81640B TLS absolute wavelength accuracy specification.

<sup>c</sup> Refer to Agilent 81640A and 81640B TLS relative wavelength accuracy specification.

<sup>d</sup> With Agilent 86122A: 1270,1 nm to 1375 nm

Table 19 Optical Fiber Chromatic Dispersion Measurement

Zero Dispersion Wavelength Accuracy (characteristic) <sup>a,b,c,d</sup>	$\pm 150 \text{ pm}$
Zero Dispersion Wavelength Repeatability (characteristic) <sup>a,b</sup>	$\pm 9 \text{ pm}$
Accuracy of dispersion slope at the zero dispersion wavelength (characteristic) <sup>a,e</sup>	$\pm 25 \text{ fs/nm}^2$
Repeatability of dispersion slope at the zero dispersion wavelength (characteristic) <sup>a,e</sup>	$\pm 3 \text{ fs/nm}^2$

<sup>a</sup> Measurements performed with Modulation frequency = 2 GHz, IFBW = 30 Hz, stepped mode with 51 phase samples per trace point, 1 nm wavelength step size, 1515 nm to 1580 nm wavelength range, receiver input power = -43 dBm at 1550 nm. Normalization performed with Phase Dynamic Accuracy selected and an Agilent 86122A multi-wavelength meter attached to the system.

<sup>b</sup> Based on measurements of NIST Standard Reference Material 2524, Serial No. 0001, using a quadratic fit to each raw relative group delay trace.

<sup>c</sup> Includes the characterization uncertainty of the zero dispersion wavelength as quoted by NIST of  $\pm 60 \text{ pm}$  (2 standard deviations).

<sup>d</sup> Includes the uncertainty in the temperature measurement used to correct the zero dispersion wavelength of  $\pm 30 \text{ pm}$  (2 standard deviations).

<sup>e</sup> Based on measurements of NIST Standard Reference Material 2524, Serial No. 0001, using a 5-term Sellmeier fit to each raw group delay trace.

Table 20 Power Requirements

E9247A	
Voltage, Frequency	100 / 115 V~, 50 / 60 / 400 Hz 230 / 240 V~, 50 / 60 Hz
max. Power	350 W
83427B	
Voltage, Frequency	115 / 230 V, 50 / 60 Hz
max. Power	250 VA

Table 21 Environmental Conditions

Operating Temperature	+10° C to +35° C
Specified Temperature Range	25° C $\pm$ 5° K
Operating Humidity	<80% RH
Storage Temperature	-40° C to +70° C
Storage Humidity	5% to 95% at +40° C
Altitude	2000 m maximum (8164B limit)



Table 22 Dimensions and Weights

	Dimensions	Weight
E9427A	222 mm H x 425 mm W x 426mm D	24 kg (54 lbs)
8164B	145 mm H x 426 mm W x 545 mm D	23 kg (51 lbs)
83427B	132 mm H x 426 mm W x 480 mm D	10 kg (25 lbs)
86122A	138 mm H x 425 mm W x 520mm D	14.5kg (32 lbs)

# 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 23 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



**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:** Optical Dispersion Analyzer  
**Product Number:** 86038A containing E9427A and 83427B  
**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:

	Standard	Limit
<b>EMC</b>	IEC 61326:1997+A1:1998+A2:2000 / EN 61326:1997+A1:1998+A2:2001	
	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 / EN 61000-4-3:2002	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 2064.1		
<b>Safety</b>	IEC 61010-1:2001 / EN 61010-1:2001	
	IEC 60825-1:2001 / EN 60825-1:1994+A11:1996+A2:2001	
	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:

2004-October-01  
Date

  
Hans-Martin Fischer

Product Regulations Representative PMD, DVS  
Agilent Technologies  
Title

For further information, please contact your local Agilent Technologies sales office, agent or distributor.





# 9 Glossary

Dispersion Terminology.....	294
-----------------------------	-----

# Dispersion Terminology

## $\Delta\tau$ **Differential Group Delay (DGD)**

The difference in group delay between the fast and slow polarization modes of a fiber or device at a particular wavelength and temperature.

## $\langle\Delta\tau\rangle$ **Mean Differential Group Delay (DGD)**

In fiber measurements, the mean value of the DGD over wavelength and temperature. Applies to the long-fiber regime, and assumes sufficient range of wavelength or temperature.

## $\omega$ **Radian Frequency**

The rate in which an angle changes. Expressed in radians/second. Also known as angular velocity.

## $f$

Frequency in Hertz.

## $L_C$ **Correlation Length**

Distance over which polarization state is decorrelated.

## **ASE Amplified Spontaneous Emissions**

The unpolarized noise output of an erbium doped fiber amplifier (EDFA).

## **Basis States**

Two orthogonally polarized beams which, when superimposed, create a third polarization state. Assumes monochromatic light.

## **Beat Length**

The shortest length of uniformly birefringent fiber over which the fast and slow polarization modes experience 360-degrees of phase shift.

## **Birefringence**

The property of optical fiber and devices that causes light to decompose into two orthogonally polarized waves traveling at slightly different speeds.

**Chromatic Dispersion (CD)**

In digital lightwave technology, pulse broadening caused by the variation of group delay across the spectrum of the signal. Defined as the derivative of the group delay with respect to wavelength and expressed in ps/nm. See also "[Chromatic Dispersion Coefficient](#)".

**Chromatic Dispersion Coefficient**

In the context of transmission fiber, the chromatic dispersion value divided by the length of fibers in kilometers. Usually expressed in ps/km.

**Coherence**

A signal property related to  $1/\text{spectral width}$ . Narrow spectra mean high coherence, wide spectra mean low coherence. When low coherence signals are separated and recombined, they interfere only if the path lengths are nearly equal. High coherence signals interfere over large path differences.

**Correlation Length**

In the context of long, mode-coupled optical fiber, the fiber length over which the polarization state of the propagating signal becomes uncorrelated relative to the input polarization state. Usually expressed in meters.

**Curve Fitting**

Post-processing of the measurement trace data in which a model mathematical curve is optimally fitted to the trace. The 86038A ODA provides a default set of curve fitting functions for fiber test: quadratic, 3-term and 5-term Sellmeier. Also provide is a linear curve fit that can be applied to some components. The user can create and install new curve fitting functions.

**Distributed Feedback Laser (DFB)**

Used as a source in high-speed fiber systems. Single line, narrow linewidth.

**Differential Group Delay (DGD)**

The difference in group delay between the fast and slow polarization modes of a fiber or device at a particular wavelength and temperature.

**Degree of Polarization (DOP)** Degree of Polarization. The fraction of the optical power (intensity) that is polarized. Fiber coil polarization controllers can manipulate this part of the light.

**Erbium Doped Fiber Amplifier (EDFA)**

Made WDM systems possible.

**Ergodic**

In an Ergodic system, the statistical properties remain the same over a change of independent variables. Long-fiber different group delay (DGD) is Ergodic between wavelength and temperature, assuming wide variation of each.

**Frequency Selective Device**

An optical component that routes or rejects signals on the basis of frequency (or wavelength), but not necessarily a narrowband device. See also "[Narrowband Device](#)".

**Intermediate Frequency Bandwidth (IFBW)**

The bandwidth of the filter employed in the network analyzer portion of the 86038A ODA. When testing high insertion loss devices, a narrower IF bandwidth decreases measurement noise and increases measurement time.

**Lambda Zero**

See "[Zero Dispersion Wavelength](#)".

**Long Fiber Regime**

In the context of transmission fiber, the situation in which the fiber is sufficiently long that a large amount of mode coupling exists. A more formal term is 'random mode coupling'.

**Mean Differential Group Delay (DGD)**

In fiber measurements, the mean value of the DGD over wavelength or temperature. Applies to the long-fiber regime, and assumes sufficient range of wavelength or temperature.

**Mode Coupling**

See "[Polarization Mode Coupling](#)".

**Modal Dispersion**

In digital lightwave transmission links using multimode fiber, pulse broadening caused by the decomposition of the signal into optical modes that have different path lengths. This problem is avoided in single-mode fibers.



**Modulation Frequency**

The frequency of the sinusoidal RF signal used to modulate the intensity of the test signal in the 86038A ODA. See also [“Resolution”](#).

**Narrowband Device**

A particular type of frequency selective device in which a relatively narrow range of wavelengths are transmitted or reflected. See also [“Frequency Selective Device”](#).

**Orthogonally**

Two same frequency, monochromatic beams are orthogonally polarized when they can be superimposed without interference effects. Orthogonal states are located diametrically opposite one another on the Poincare sphere.

**Polarization Dependant Gain (PDG)**

A characteristic of EDFA's in which the gain experienced by the signal is slightly different from the gain experienced by the noise in the orthogonal polarization state.

**Polarization Dependant Loss (PDL)**

The variation in loss over all polarization states. Usually expressed in dB.

**Polarization Hole Burning (PHB)**

One of the processes underlying polarization dependent gain (PDG). A slight polarization dependence of the de-excitation of erbium ions allows slightly more noise to be produced in the polarization state orthogonal to the signal.

**Polarization Mode Dispersion (PMD)**

In digital lightwave technology, pulse broadening caused by the variation of group delay with input polarization state. See also [“PMD Delay”](#).

**PMD Coefficient**

Is the mean DGD divided by the square root of the system length in kilometers.

**PMD Compensator (PMDC)**

Usually an active system situated before the optical receiver.

**PMD Delay**

In the long fiber regime, the mean DGD over a wide range of wavelength or temperature.

**Polarization Maintaining Fiber (PMF)**

A specially constructed fiber exhibiting a birefringence value several thousand times stronger than transmission fiber. It is used to guide linearly polarized light from point-to-point, usually between components or instruments. It exhibits high-PMD, usually more than 1 ps of DGD per meter.

**Poincare Sphere**

A spherical representation of polarization state in which each possible state maps into a point on the sphere.

**Polarimeter**

An instrument that measures the polarization properties of light.

**Polarization Adjuster**

See “[Polarization Controller](#)”.

**Polarization Controller**

A device that allows you to transform an input state to a desired output state.

**Polarization Mode Coupling**

In optical fiber, the phenomenon in which energy exchanges between local fast and slow polarization modes. It can be caused by a fiber interface or by a local disturbance of the intrinsic (native) birefringence of the fiber.

**Principal State of Polarization (PSP)**

In the context of optical fiber in the randomly mode coupled regime, the input PSP's are the two orthogonal states that identify the fast and slow modes of the overall fiber link. They are associated with another pair of PSP's defined at the output. By definition, the PSP's are the states at which the polarization transformation of the link is independent of wavelength (over a suitably narrow increment of wavelength). See also “[PSP Bandwidth](#)” .

**PSP Bandwidth**

The width of the wavelength or frequency interval over which the PSP's remain fixed. See also PSP.

**Random Mode Coupling**

In the context of transmission fiber, the situation in which the fiber is sufficiently long that a large amount of mode coupling exists. Sometimes referred to as the long-fiber regime.

**Resolution**

The ability of the 86038A ODA to reveal fine details in the response of the test device (for example, a fine group delay ripple, or a steep filter slope). Resolution depends on various setup factors, especially wavelength step size and modulation frequency.

**Retarder**

Birefringent device used as a phase shifter.

**Samples**

In the context of the 86038A ODA in the Step mode of operation, the 'Samples/Point' refers to the number of amplitude and phase values acquired during a single sweep of the network analyzer at each given measurement wavelength.

**Scan Polarization**

Explore all polarization states for the purpose of measuring the polarization dependence of a test device.

**Scramble Polarization**

Change the polarization state in a fashion that makes the light appear unpolarized to a measuring instrument such as a power meter, spectrum analyzer or high resolution spectrometer.

**Sensitivity**

The ability of the 86038A ODA to reveal the response of the test device in the presence of high optical attenuation. Sensitivity depends of various setup factors, especially number of samples and IF bandwidth.

**Slope at Lambda Zero**

The slope of the curve of dispersion versus wavelength at the zero dispersion wavelength. This value is used by system designers to estimate the dispersion at other wavelengths. See also [“Zero Dispersion Wavelength”](#) .

**Smoothing**

Post-processing of the measurement trace data to remove higher-frequency components such as noise. The 86038A ODA provides a default set of smoothing functions and allows you to create and install new ones.

**State of Polarization (SOP)**

Describes the polarization state of the polarized portion of a lightwave signal. See also [“Degree of Polarization \(DOP\)”](#) , [“Poincare Sphere”](#) , and [“Stokes Parameters”](#) .

**Stokes Parameters**

A set of values describing the power (intensity) and polarization characteristics of a lightwave. Stokes parameters can be used to describe fully, partially, or unpolarized light.

**Wavelength Resolution**

See [“Resolution”](#) .

**Wavelength Step**

In the 86038A ODA operating in the Step mode, the separation between adjacent test wavelengths. See also [“Resolution”](#) .

**Waveplate**

Birefringent device used as a phase shifter.

**Zero Dispersion Wavelength**

In optical fiber testing, a wavelength at which the group delay versus wavelength curve has zero slope. See also [“Slope at Lambda Zero”](#) .

# Index

## #

---

- 86038A
  - block diagram 41
- 86122A
  - firmware revision 258
  - improving accuracy 57
  - installation 17
  - problems 255, 262
  - settings 92
  - wavelength correction 56, 107

## A

---

- accessing the ODA name 131
- Action commands 203
- activate
  - markers 122
  - trace 121
  - zoom 122
- active marker 115
- active trace 116
- Active X interface 141
- adjusting modulator bias 55
- aliasing errors 43, 46
- amplitude phase conversion effects 108
- amplitude scale 112
- amplitude zoom 123
- Analysis - Curve Fit commands 205
- angled connectors 285
- assigning trace names 28
- attenuator, measuring 71
- auto scale functions 112, 122

## B

---

- Base commands 201
- Begin Measurement 124
- block diagram 41
- bmp file format 89
- broadband devices 65, 71

## C

---

- C# programming 172
- caution, definition of 11
- CE mark 12

- change view 121
- changing network settings 131
- changing trace names 28
- characteristics, instrument 284
- chromatic dispersion specifications 287
- circulator, measuring 71
- cleaning
  - accessories for connectors 279
  - adapters 280
  - connectors 279
  - fiber-optic connections 272, 279
  - non-lensed connectors 279
- cleaning connectors 272
- client to server commands 198
- communication problems 257, 259
- computer machine name, viewing or changing 131
- Connectivity commands 208
- connector care 272
- connector cleaning 272
- connectors
  - accessories 279
  - cleaning 279
- continuous sweep 106
- control functions 124
- cotton swabs 279
- coupler, measuring 71
- CSA mark 12
- csv file format 89
- curve fit algorithms 120
- Curve Fit function 120
- customize normalization settings 100

## D

---

- data errors 263
- dB per division 112
- declaration of conformity 291
- default settings 92
- default values
  - display 92
  - measurement settings 92
- device types, choosing setup parameters 65
- differential group delay specifications 286
- dimensions 288, 289

- dispersion measurements 33
- Display commands 210
- display markers 114
- Display menu 111
- display multiple graphs 121
- DNS, using 132
- dust caps 279
- DUT, choosing setup parameters 65
- DWDM wavelength routing device, measuring 70
- dynamic drift reduction 105
- dynamic range 286

## E

---

- enumerations 141
- environmental conditions 288, 289
- erbium doped fiber, measuring 68
- event table, remote programming 144
- example measurement 72, 77
- Excel programming example 183
- export files 89

## F

---

- FDA laser safety 15
- Fiber Bragg Grating dispersion compensators, measuring 67
- fiber dispersion compensating module, measuring 66
- fiber length measurements 40
- fiber measurement 65, 77
- fiber optics
  - cleaning connections 272
- File commands 213
- file formats 89
- File menu 89, 90
- filter measurement 72
- firmware upgrade procedure 270
- foam swabs 279
- frequency zoom 123
- front panel
  - connections 18
  - overview 26

**G**

gain loss accuracy 286  
 GD ripple 50  
 GD trace offset 116  
 general specifications 288, 289  
 GPIB problems 257, 259  
 group delay  
   noisy trace 266  
   ripple 43, 269  
   smoothing 35  
   specifications 286  
   trace offset to zero 113

**H**

hard drive backup and recovery 271  
 height 288, 289  
 horizontal zoom 123

**I**

IF bandwidth 104  
 imaging, causes of 47  
 improving signal to noise 105  
 input connector 272  
 insertion loss 34  
 installation 16  
 installing remote client 133  
 instrument  
   characteristics 284  
   preset 92  
   returning for service 282  
   specifications 284  
   warm-up 20  
 internal thermal regulator 93  
 ISM1-A 12  
 isolators, measuring 71

**J**

jpg file format 89

**K**

key functions 26  
 keyboard, connecting 17

**L**

Lab View programming example 185  
 lambda icon 105  
 LAN connection 130  
 laser  
   classifications 15  
   safety 15  
   setup 105  
 length

measurements 40, 98  
 normalization 56

Length commands 215

**M**

making a measurement 30  
 marker  
   position 114  
   search 114  
   search criteria 115  
   setup 114  
   to center 116  
   to reference position 116

measure control functions 124

Measure menu 107

measurement  
   example 72, 77  
   length 98  
   method 41  
   modes 57  
   noise 44  
   problems 266  
   progress indicator 107  
   reducing error 105  
   repeatability 43  
   resolution 47

Measurement Range commands 219

measurement range, setting 98

measurements 33  
   fiber 65  
   length 40  
   optimized 33  
   polarization dependant loss 39  
   selecting types of 97  
   theory of 42

modulation frequency, effects of PDA 108

modulation phase shift method 41, 43

modulator bias 55, 108, 110

modulator bias problems 262

mouse, connecting 17

mouse-only functions 122

multiple graphs 121

multi-wavelength meter  
   firmware revision 258  
   improving accuracy 57  
   installation 17  
   problems 255, 262  
   settings 92  
   wavelength correction 56, 107

**N**

narrowband devices 70  
 narrowband filter, measuring 70  
 network settings, how to change 131  
 networking 130  
 next trace selection 121

noise declaration 290

noise, reducing 104

normalization  
   customizing 100  
   initiating 107  
   operations that affect 110  
   problems 262  
   selecting ports 108  
   setting measurement range 100  
   theory of 55

Normalization Range commands 221

**O**

object select 122  
 ODA name, viewing or changing 131  
 offset GD trace 116  
 offset GD trace to zero 113  
 operation verification 20  
 operations affecting normalization 110  
 optical modulator bias 55  
 optical modulator bias problems 262  
 optimize optical modulator bias 108, 110  
 optimizing measurements 33

**P**

pan zoom 123  
 parts list 16  
 peak excursion 115  
 peak marker 115  
 performance verification 20  
 phase dynamic accuracy 55, 108  
 phase wrapping 45  
 phase wrapping errors 43  
 pit marker 115  
 polarization dependant loss 39  
 polarization mode dispersion 37  
 polarization scanner 42  
 port selection 98  
 ports, selecting for normalization 108  
 power requirements 288, 289  
 preset display 92  
 preset states 92  
 print 90  
 printer port 19  
 Printing commands 228  
 printing data 90  
 programming example  
   C# 172  
   Excel 183  
   Lab View 185  
   VB 6.0 155  
   VB.NET 164

**Q**


---

Quick Setup 30, 97

**R**


---

real time port monitor 94  
 examples 95

rear panel connections 18

recall data 89

reducing measurement error 105

reducing noise 85

reference level position 112

reference level, setting 112

regulatory information 290

remote client, installation 133

repeat sweep 106

resetting the instrument 92

resolution 101

Resolution commands 229

resolution, increasing 47

Results commands 235

returning for service 282

RF modulation frequency  
 maximum values 46  
 setting 101

ripple 269  
 amplitude accuracy 48  
 GD or insertion loss 34

ripple error, sideband separation 50

**S**


---

safety 11

safety considerations 15

safety information 11

safety symbols 11

sample measurement 72, 77

samples per point 104

save data 89

selecting a trace 121

sensitivity 101

Sensitivity commands 239

sensitivity, setting 104

server to client events 193

service  
 returning for 282

settle TLS 105

setup  
 markers 114  
 parameters 63  
 parameters, theory of 43  
 wizard 97

Setup menu 97

setup wizard 30

shipping contents 16

Show DGD Contribution function 117

show markers 114

single sweep 106

smoothing  
 algorithms 118  
 removing noise 85  
 selection 118

software upgrade procedure 270

span, display setting 111

startup problems 255, 259

Status commands 242

status log, error messages 93

stepped measurements 57

swabs 279

sweep problems 262

swept measurements 57

swept relative group delay specifications 286

symbols, safety 11

System commands 247

system diagram 41

system preset 92

system specifications 288, 289

**T**


---

TCP/IP configuration 132

temperature icon 93

theory of operation 41

thermometer icon 93

tif file format 89

TLS  
 commands 251  
 firmware revisions 258  
 problems 255, 265  
 settling 105

toggle displays 121

toolbar functions 122

trace names, defining 28

trace points, setting 101

trigger mode 106, 124

troubleshooting 255

turning markers on 122

**U**


---

units per division 112

upgrade procedure 270

USB hub 17

user interface 26

**V**


---

VB 6.0 programming 155

VB.NET programming 164

verification test 20

vertical zoom 123

**W**


---

warm-up time 20

warning, definition of 11

wavelength accuracy specifications 287

wavelength correction 56

wavelength correction problems 262

wavelength meter  
 firmware revision 258  
 improving accuracy 57  
 installation 17  
 problems 255, 262  
 settings 92  
 wavelength correction 56, 107

wavelength scale 111

wavelength zoom 123

weight 288, 289

WINS, using 132

**X**


---

x scale 111

**Y**


---

y scale, amplitude display 112

**Z**


---

zero dispersion specifications 287

zero GD trace 116

zoom functions 122, 123







© Agilent Technologies GmbH 2004

Printed in Germany December 2004  
Third edition, December 2004



86038-90006