

ARTISAN®
TECHNOLOGY GROUP

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

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

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

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

Custom engineering
so your equipment works exactly as you specify.

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

Expert team | Trust guarantee | 100% satisfaction

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

Find the *Exfo FLS-2200* at our website: ***Click HERE***

User Guide

FLS-2200

Broadband Source



www.EXFO.com

Telecom Test and Measurement

EXFO

EXPERTISE REACHING OUT

Copyright © 2005–2013 EXFO Inc. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system or transmitted in any form, be it electronically, mechanically, or by any other means such as photocopying, recording or otherwise, without the prior written permission of EXFO Inc. (EXFO).

Information provided by EXFO is believed to be accurate and reliable. However, no responsibility is assumed by EXFO for its use nor for any infringements of patents or other rights of third parties that may result from its use. No license is granted by implication or otherwise under any patent rights of EXFO.

EXFO's Commerce And Government Entities (CAGE) code under the North Atlantic Treaty Organization (NATO) is 0L8C3.

The information contained in this publication is subject to change without notice.

Trademarks

EXFO's trademarks have been identified as such. However, the presence or absence of such identification does not affect the legal status of any trademark.

Units of Measurement

Units of measurement in this publication conform to SI standards and practices.

Patents

EXFO's Universal Interface is protected by US patent 6,612,750.

Version number: 2.0.0

Contents

Certification information vi

1 Introducing the FLS-2200 Broadband Source 1
Main Features 1
Available Models 2
Typical Applications 3
Conventions 4

2 Safety Information 5
Electrical Safety Information 6

3 Getting Started with Your Broadband Source 9
Installing the EXFO Universal Interface (EUI) 11
Turning On/Off the Broadband Source 12
Installing EXFO LabVIEW Drivers 14

4 Setting Up Your Broadband Source 17
Setting the Refresh Rate 17
Activating or Deactivating the Backlight 18
Setting the Contrast 18
Setting the Video Mode 19
Setting the Drive Current 20
Reverting the Broadband Source to Default Settings 21

5 Operating the Broadband Source 23
Cleaning and Connecting Optical Fibers 23
Activating or Deactivating a Source 25

6 Preparing for Remote Control 27
Linking Units with the GPIB Port 27
Linking Units with the Serial Port 28
Changing Communication Settings 29
Setting the Remote Control Mode 30
Setting GPIB Address 31
Setting Baud Rate 32
Setting Flow Control 33

7 Using Your Broadband Source in an Automated Test Environment 35
Message Management 35
Standard Status Data Structure 39
SCPI Command Structure 44
Consulting Data Types 49
Writing Remote Control Code 49
Error Message Format 51
Working with EXFO LabVIEW Drivers 52
Using the EXFO Getting Started Applications 54
Building and Using Custom VIs 59

8 Maintenance 65
Cleaning EUI Connectors 66
Replacing Fuses 69
Upgrading the Embedded Software 70
Recycling and Disposal (Applies to European Union Only) 72

9 Troubleshooting 73
Error Messages 73
Solving GPIB Common Problems 77
Contacting the Technical Support Group 78
Transportation 78

10 Warranty 79
General Information 79
Liability 80
Exclusions 81
Certification 81
Service and Repairs 82
EXFO Service Centers Worldwide 83

A Technical Specifications 85

B Data Types 89
Applicable Data Types for Input—IEEE 488.2 90
Applicable Data Types for Output—IEEE 488.2 99
Applicable Data Types for Input—SCPI 109
Special Numeric Values Received on Output 110

C IEEE 488.2 and Specific Command Reference 111
IEEE 488.2 Commands—Quick Reference 111
IEEE 488.2 Commands—Description 112
Product-Specific Commands—Quick Reference 135
Product-Specific Commands—Description 136

D SCPI-Based Errors 151
Index 167

Certification information

North America Regulatory Statement

This unit was certified by an agency approved in both Canada and the United States of America. It has been evaluated according to applicable North American approved standards for product safety for use in Canada and the United States.

Electronic test and measurement equipment is exempt from FCC part 15, subpart B compliance in the United States of America and from ICES-003 compliance in Canada. However, EXFO Inc. makes reasonable efforts to ensure compliance to the applicable standards.

The limits set by these standards are designed to provide reasonable protection against harmful interference when the equipment is operated in a commercial environment. This equipment generates, uses, and can radiate radio frequency energy and, if not installed and used in accordance with the user guide, may cause harmful interference to radio communications. Operation of this equipment in a residential area is likely to cause harmful interference in which case the user will be required to correct the interference at his own expense.

Modifications not expressly approved by the manufacturer could void the user's authority to operate the equipment.



IMPORTANT

Use of shielded remote I/O cables, with properly grounded shields and metal connectors, is recommended in order to reduce radio frequency interference that may emanate from these cables.

EXFO
Electro-Optical Engineering

DECLARATION OF CONFORMITY

Application of Council Directive(s):	73/23/EEC - The Low Voltage Directive 89/336/EEC - The EMC Directive
Manufacturer's Name:	EXFO ELECTRO-OPTICAL ENG.
Manufacturer's Address:	400 Godin Avenue Vanier, Quebec Canada G1M 2K2 (418) 683-0211
Equipment Type/Environment:	Light Industrial Scientific Equipment
Trade Name/Model No.:	FLS-2200 Broadband Light Source

Standard(s) to which Conformity is Declared:

EN 61010-1:1993/ A2: 1995	**Safety Requirements for Electrical Equipment for Measurement, Control, and Laboratory Use, Part 1: General Requirements**
EN 55022:1998/ A1:2000	**Limits and methods of measurement of radio disturbance characteristics of industrial, scientific, and medical equipment**
EN 61326:1997/ A2: 2001	**Electrical Equipment for Measurement, Control and Laboratory Use - EMC Requirements**
EN 60825-1:1994/ A2: 2001	**Safety of laser products-Part 1 :Equipment classifications, requirements and user's guide**

I, the undersigned, hereby declare that the equipment specified above conforms to the above Directive and Standards.

Manufacturer

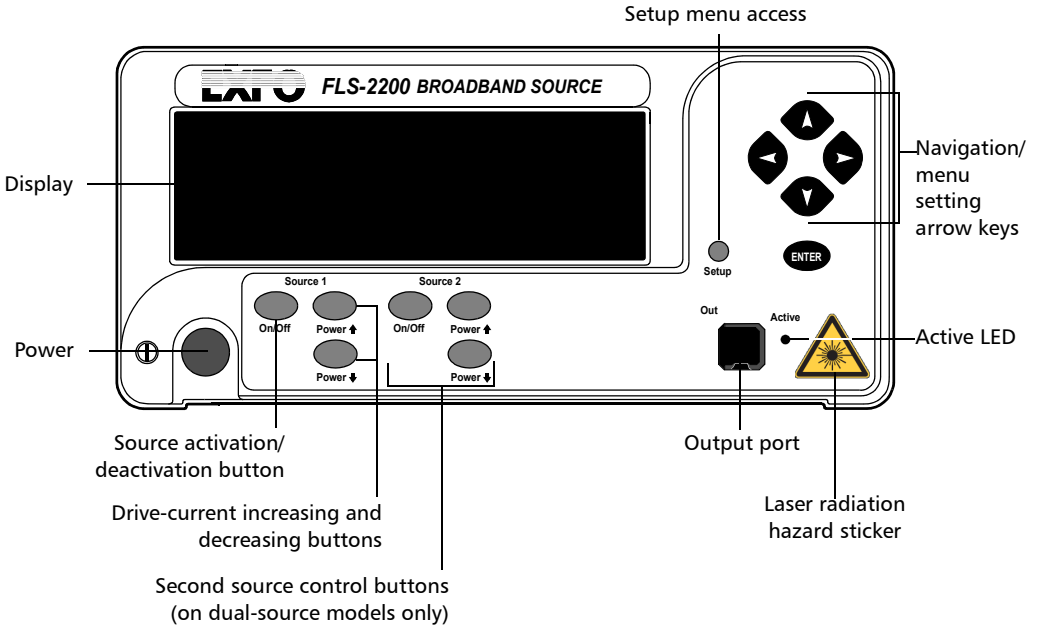
Signature: 

Full Name:	Stephen Bull, E. Eng
Position:	Vice-President Research and Development
Address:	465 Godin Avenue Vanier, Quebec, Canada
Date:	May 20 , 2003

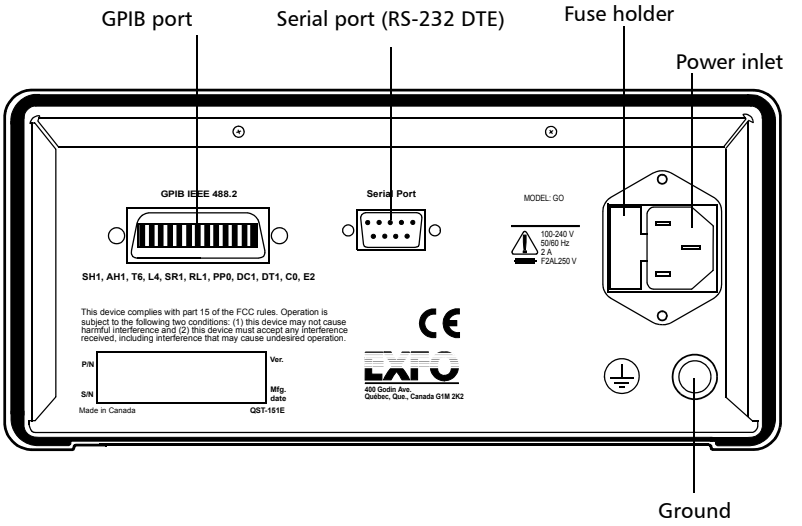
1 Introducing the FLS-2200 Broadband Source

Main Features

The FLS-2200 Broadband Source is a super-luminescent, light-emitting diode (SLED) source covering all the bands needed for telecommunications applications. It provides a broader spectral range and more power density in a singlemode fiber than a white light source.



On the back panel, you will find the ports for remote control, the power inlet and fuse holder.



The FLS-2200 Broadband Source supports local control (via its front panel) and remote control (through GPIB or RS-232 using SCPI commands or the provided LabVIEW drivers).

Available Models

The FLS-2200 Broadband Source is available in the following models:

- Models with a single super LED (SLED) covering a conventional wavelength range (5 models).
- Models with two SLEDs, covering a wider range for coarse wavelength-division multiplexing (CWDM) and dual-window applications (2 models).

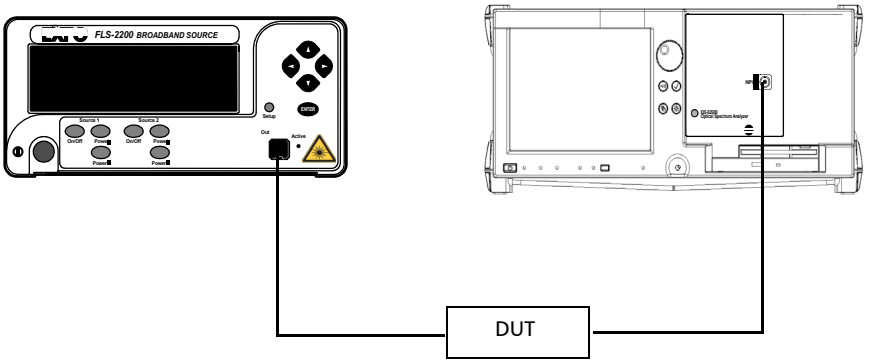
Typical Applications

You can use this light source to:

- qualify components during development or to perform Pass/Fail tests during production when you use the source with an optical spectrum analyzer (OSA).
- perform maintenance or troubleshooting tasks on a WDM network when the source is combined with an OSA.
- perform polarization mode dispersion (PMD) measurements when you use the source with the FPMD-5600 Femtosecond PMD Analyzer and the M9700 Passive Depolarizer.
- perform polarization-dependent loss (PDL) measurements when you use the source with an OSA.

The typical devices tested include passive optical network (PON) components, CWDM components, attenuators, photonic switches, broadband couplers, specialty couplers or multiplexers, Bragg gratings (the source features a bandwidth large enough to test many Bragg gratings simultaneously), etc.

A typical setup including both the FLS-2200 Broadband Source and an IQS-5250B Optical Spectrum Analyzer is shown below.



Conventions

Before using the product described in this guide, you should understand the following conventions:



WARNING

Indicates a potentially hazardous situation which, if not avoided, could result in *death or serious injury*. Do not proceed unless you understand and meet the required conditions.



CAUTION

Indicates a potentially hazardous situation which, if not avoided, may result in *minor or moderate injury*. Do not proceed unless you understand and meet the required conditions.



CAUTION

Indicates a potentially hazardous situation which, if not avoided, may result in *component damage*. Do not proceed unless you understand and meet the required conditions.



IMPORTANT

Refers to information about this product you should not overlook.

2 Safety Information

WARNING

Do not install or terminate fibers while a light source is active. Never look directly into a live fiber and ensure that your eyes are protected at all times.

WARNING

The use of controls, adjustments and procedures other than those specified herein may result in exposure to hazardous situations or impair the protection provided by this unit.

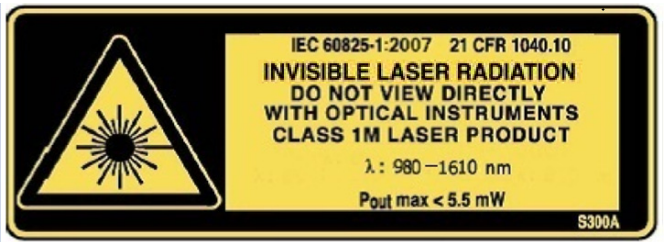
IMPORTANT

When you see the following symbol on your unit , make sure that you refer to the instructions provided in your user documentation. Ensure that you understand and meet the required conditions before using your product.

Your instrument is a Class 1M laser product in compliance with standards IEC 60825-1 2007 and 21 CFR 1040.10. Invisible laser radiation may be encountered at the output port.

The product is safe under reasonably foreseeable conditions of operation but it may be hazardous if you use optics within a diverging or collimated beam. *Do not view directly with optical instruments.*

The following label(s) indicate that the product contains a Class 1M source:



Electrical Safety Information

This unit uses an international safety standard three-wire power cable. This cable serves as a ground when connected to an appropriate AC power outlet.

Note: *If you need to ensure that the unit is completely powered off, disconnect the power cable.*



WARNING

- **Insert the power cable plug into a power outlet with a protective ground contact. Do not use an extension cord without a protective conductor.**
- **Before powering on the unit, connect all grounding terminals, extension cords and devices to a protective ground via a ground socket. Any interruption of the protective grounding is a potential shock hazard and may cause personal injury. Whenever the ground protection is impaired, do not use the unit and secure it against any accidental operation.**
- **Do not tamper with the protective ground terminal.**

The color coding used in the electric cable depends on the cable. New plugs should meet the local safety requirements and include:

- adequate load-carrying capacity
- ground connection
- cable clamp



IMPORTANT

EXFO assumes no liability if you attempt to perform internal service on this unit.



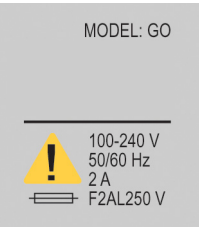
WARNING

- **Use this unit indoors only.**
- **Position the unit so that the air can circulate freely around it.**
- **Operation of any electrical instrument around flammable gases or fumes constitutes a major safety hazard.**
- **Do not remove unit covers during operation.**
- **To avoid electrical shock, do not operate the unit if any part of the outer surface (covers, panels, etc.) is damaged.**
- **Only authorized personnel should carry out adjustments, maintenance or repair of opened units under voltage. A person qualified in first aid must also be present. Do not replace any components while power cable are connected.**
- **Use only fuses with the required rated current and specified type (IEC, 5 mm x 20 mm (0.197 in x 0.787 in), fast-blow, 250 V, 2 A). Do not use repaired fuses or short-circuited fuse holders.**
- **Capacitors inside the unit may be charged even if the unit has been disconnected from its electrical supply.**

Equipment Ratings	
Temperature ➤ Operation ➤ Storage	 0 °C to 40 °C (32 °F to 104 °F) -40 °C to 70 °C (-40 °F to 158 °F)
Relative humidity[a]	0 % to 80 % non-condensing
Maximum operation altitude	2000 m (6562 ft)
Pollution degree	2
Overvoltage category	II
Power supply rating[b]	~ 100 V - 240 V; 50 Hz/60 Hz; 2 A

a. Measured in 0 °C to 31 °C (32 °F to 87.8 °F) range, decreasing linearly to 50 % at 40 °C (104 °F).
b. Not exceeding ± 10 % of the nominal voltage.

The following label is located on the back panel of the unit:

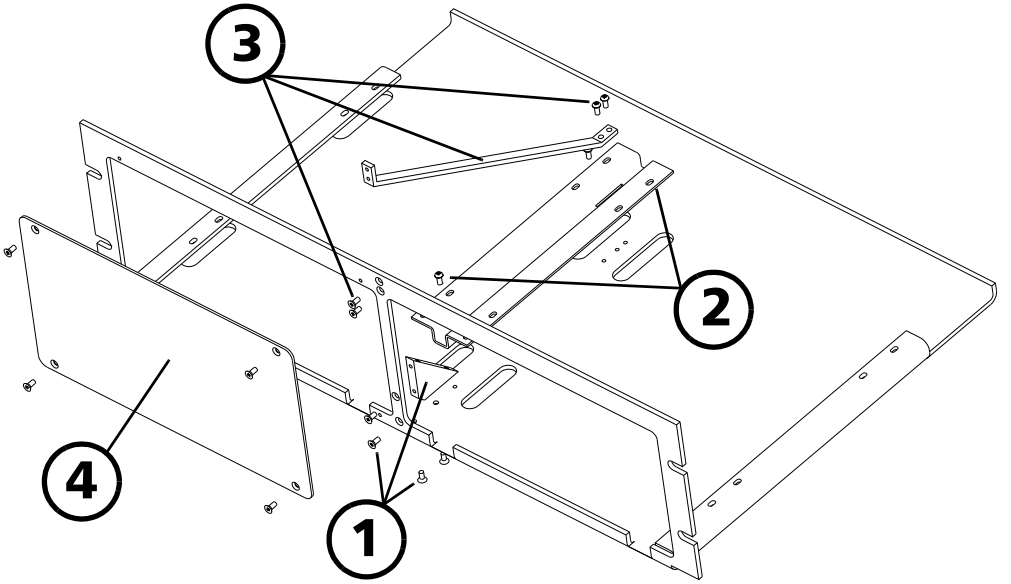


3 Getting Started with Your Broadband Source

You can place your FLS-2200 Broadband Source in a rackmount to facilitate its usage.

To install the rackmount:

1. Fix the angle iron using four flat Phillips screws.



2. Fix the rackmount bracket to the frame using two round Phillips screws.

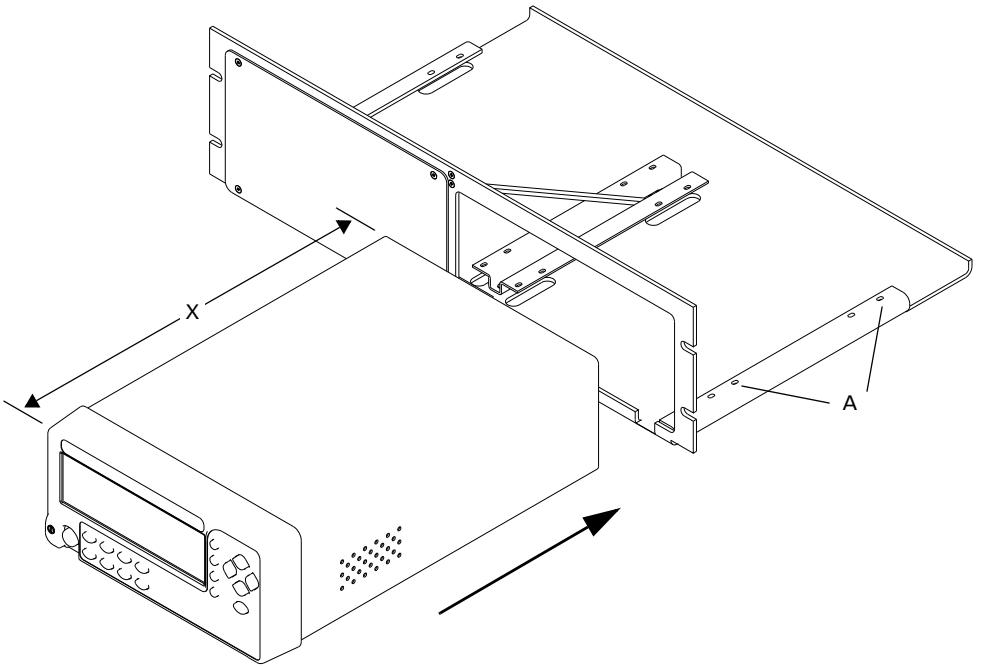
3. Fix the rackmount stiffener using two flat Phillips screws (for the front panel) and two round Phillips screws.

4. If your rackmount will contain only one unit, fix the rackmount cover plate to the empty part of the frame using four flat Phillips screws.

To install your FLS-2200 Broadband Source in a rackmount:

1. Slide the benchtop unit into the rackmount and tighten it from underneath using the four cover fixing screws.

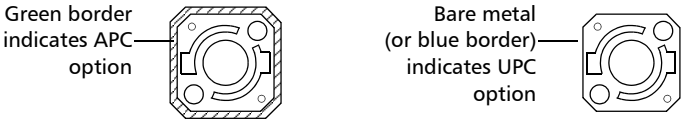
If measurement X on the illustration exceeds 11.125 in., fix the unit into the four holes identified as A. Otherwise, use the other four holes.



2. If a second benchtop is to be installed, remove the cover plate and repeat step 1.

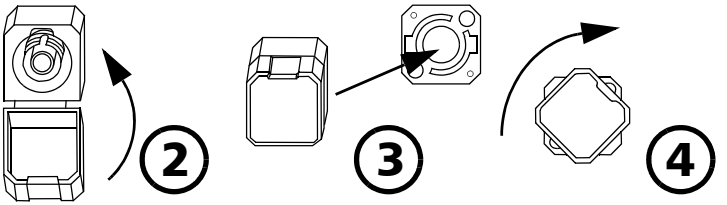
Installing the EXFO Universal Interface (EUI)

The EUI fixed baseplate is available for connectors with angled (APC) or non-angled (UPC) polishing. A green border around the baseplate indicates that it is for APC-type connectors.



To install an EUI connector adapter onto the EUI baseplate:

1. Hold the EUI connector adapter so the dust cap opens downwards.



2. Close the dust cap in order to hold the connector adapter more firmly.
3. Insert the connector adapter into the baseplate.
4. While pushing firmly, turn the connector adapter clockwise on the baseplate to lock it in place.

Turning On/Off the Broadband Source

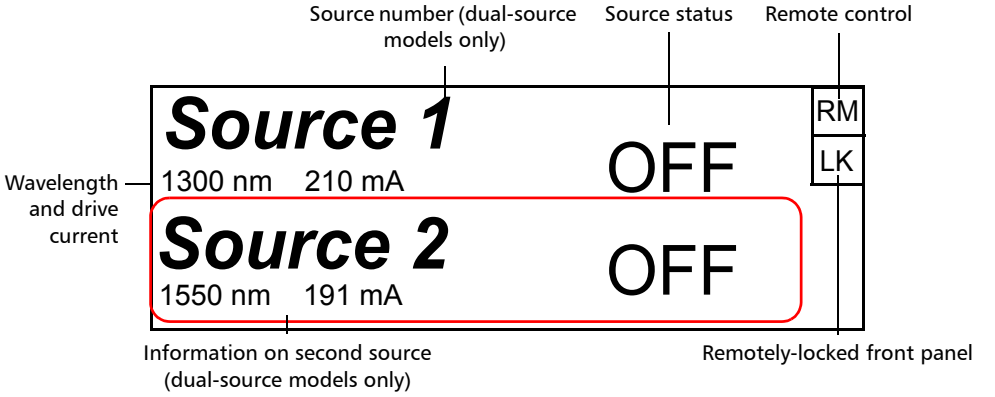
WARNING

Before turning on the source, read *Electrical Safety Information* on page 6.

To turn the Broadband Source on and off:

Press the red button located in the lower left-hand corner of the front panel.

Upon startup, the unit beeps twice, performs a self-test and then displays the main window, indicating that all sources are deactivated (OFF).



- The *source status indicator* shows whether the source is active or not (ON/OFF). In the case of an active source, a graphical element representing a light beam is also displayed.
- The *wavelength and drive current indicator* shows the source's nominal wavelength (in nanometers) and the selected drive current (in milliamperes).
- The *remote control indicator* (**RM**) appears when the unit is currently controlled by remote commands (via GPIB or RS-232 communication).

Note: *You must set the GPIB bus line to True in GPIB or send the *REM command in RS-232 for the remote control indicator to be displayed.*

- The *remotely-locked keyboard indicator* (**LK**) shows that a remote application is preventing you from using the unit's front panel buttons (keyboard)—except the red button used to turn the unit on or off.

When the unit is turned off, the current Setup menu settings remain in the unit's memory. These settings include display features and remote control.

Installing EXFO LabVIEW Drivers

Before being able to work with EXFO LabVIEW drivers, you must install the following elements on your computer:

- National Instruments LabVIEW software and the corresponding patches.
- EXFO LabVIEW drivers (including demo applications to help you get started with the drivers)

For information on these applications, see *Working with EXFO LabVIEW Drivers* on page 52.

Note: *Only administrator-level users can install software under Windows XP.*

To install the LabVIEW software:

1. Insert the *LabVIEW* CD in the CD-ROM drive.
2. The installation process should start automatically. If not, start it manually as follows:

   2a. On the Windows taskbar, click **Start** and select **Run**.

   2b. In the **Open** box locate the *autorun.exe* file.

   2c. Click **OK** to start the installation procedure and follow the on-screen instructions.

   You should keep the default names and paths suggested by the installation program.

3. Once the software installation is complete, install the patches available for your LabVIEW version.

   If the patches are not included on your LabVIEW CD, you may download them from National Instruments' Web site at *www.ni.com*.

   3a. On the Windows taskbar, click **Start** and select **Run**.

   3b. In the **Open** box, locate *Updates|setup.exe*.

3c. Click **OK** to start the installation procedure and follow the on-screen instructions.

To install the EXFO LabVIEW drivers:

1. Insert the installation CD in the CD-ROM drive.

2. Start the installation process as follows:

2a. On the Windows taskbar, click **Start** and select **Run**.

2b. In the **Open** box, locate *Labview Drivers|setup.exe* on the storage device where the drivers are located.

2c. Click **OK** to start the InstallShield Wizard and follow the on-screen instructions.

For easier use, the drivers will be installed in LabVIEW's default instrument library folder:
C:|Program Files|National Instruments|LabVIEW 6|instr.lib|EXFO.

4 Setting Up Your Broadband Source

The blue button on the right of the display provides access to the single-level *Setup* menu. You can access this menu even while the source is active to set up the various parameters.

Setting the Refresh Rate

To set the refresh rate:

1. Press the Setup button.
2. Use the up/down or left/right arrow keys to select **Refresh Rate** (the item will be displayed in reverse video).

Refresh Rate	8 Hz	RS232 / GPIB	GPIB
Backlight	ON	GPIB Addr.	12
Contrast	▼ ◐ ▲	Baud Rate	N.A.
Video Mode	STD	Flow Ctrl	N.A.
Exit			

3. Press ENTER to access the **Refresh Rate** edit box.
4. Use the up/down arrow keys to set the refresh rate between 1/2 Hz, 1 Hz, 2 Hz, 4 Hz, 8 Hz and 16 Hz.
5. Press ENTER to confirm the new refresh rate.

Activating or Deactivating the Backlight

To deactivate the backlight:

1. Press the Setup button.
2. Use the up/down or left/right arrow keys to select **Backlight** (the item will be displayed in reverse video).
3. Press ENTER to access the **Backlight** edit box.
4. Use the up/down arrow keys until the backlight value changes to *OFF*.
5. Press ENTER to confirm the new backlight setting.

To reactivate the backlight:

➤ Stand very close to the unit to see the information displayed. Repeat steps 1 to 4 above—except that you must set the backlight value to *ON*.

OR

➤ Reset the unit to the default parameters (see *Reverting the Broadband Source to Default Settings* on page 21).

Setting the Contrast

To set the contrast:

1. Press the Setup button.
2. Use the up/down or left/right arrow keys to select **Contrast** (item will appear in reverse video).
3. Press ENTER to access the **Contrast** edit box.
4. Use the up/down arrow keys to adjust the contrast as required.
5. Press ENTER to confirm the contrast adjustment.

Setting the Video Mode

To set the video mode:

1. Press the Setup button.
2. Use the up/down or left/right arrow keys to select **Video Mode** (item will appear in reverse video).
3. Press ENTER to access the **Video Mode** edit box.
4. Use the up/down arrow keys to set the required video mode (standard or reverse).

Refresh Rate	8 Hz	RS232 / GPIB	GPIB
Backlight	ON	GPIB Addr.	12
Contrast	▼ ◐ ▲	Baud Rate	N.A.
Video Mode	STD	Flow Ctrl	N.A.
Exit			

Refresh Rate	8 Hz	RS232 / GPIB	GPIB
Backlight	ON	GPIB Addr.	12
Contrast	▼ ◐ ▲	Baud Rate	N.A.
Video Mode	INV	Flow Ctrl	N.A.
Exit			

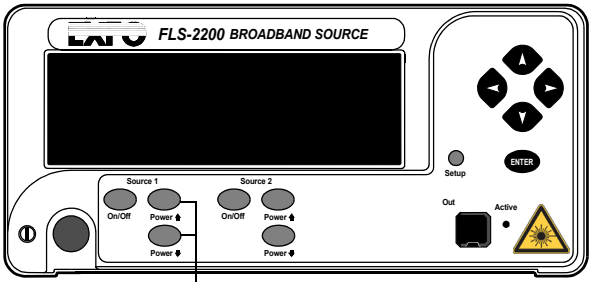
5. Press ENTER to confirm the video mode.

Setting the Drive Current

You can adjust the source's drive current to better suit your needs. The drive current can be increased or decreased by steps of 1 mA or 10 mA.

On a dual-source, each one has its own set of control buttons and has an independent drive current value.

Note: *You can set the drive current even while the source is not emitting.*



Increasing and decreasing buttons

To set the drive current:

1. Press the **Power↑** button to *increase* the value by steps of 1 mA.

   OR

   Press the **Power↓** button to *decrease* the value by steps of 1 mA.

2. To increase or decrease the value by steps of 10 mA, hold the corresponding button down.

   The displayed current value on the screen changes accordingly.

3. If you have a dual-source, you can repeat the above steps to set the drive current for the second source.

Reverting the Broadband Source to Default Settings

You may want to revert the Broadband Source to the factory settings.

The following table presents the parameters and their default values.

Parameters	Default Values
Source	OFF
Drive current	Minimum value
Backlight	ON
Videomode	STD (standard)
Refresh rate	4 Hz
RS232/GPIB (Remote control)[a]	GPIB
GPIB address[a]	12
Baud rate[a]	N.A.
Flow ctrl[a]	N.A.

a. Parameter cannot be reset by a remote control command.

To revert the source to default settings:

1. Turn off the unit.
2. Turn on the unit and press ENTER at the same time until the unit beeps three times.

5 Operating the Broadband Source

Cleaning and Connecting Optical Fibers



IMPORTANT

To ensure maximum power and to avoid erroneous readings:

- **Always inspect fiber ends and make sure that they are clean as explained below before inserting them into the port. EXFO is not responsible for damage or errors caused by bad fiber cleaning or handling.**
- **Ensure that your patchcord has appropriate connectors. Joining mismatched connectors will damage the ferrules.**

To connect the fiber-optic cable to the port:

1. Inspect the fiber using a fiber inspection microscope. If the fiber is clean, proceed to connecting it to the port. If the fiber is dirty, clean it as explained below.
2. Clean the fiber ends as follows:
   - **2a.** Gently wipe the fiber end with a lint-free swab dipped in isopropyl alcohol.
   - **2b.** Use compressed air to dry completely.
   - **2c.** Visually inspect the fiber end to ensure its cleanliness.

3. Carefully align the connector and port to prevent the fiber end from touching the outside of the port or rubbing against other surfaces.

   If your connector features a key, ensure that it is fully fitted into the port's corresponding notch.

4. Push the connector in so that the fiber-optic cable is firmly in place, thus ensuring adequate contact.

   If your connector features a screwsleeve, tighten the connector enough to firmly maintain the fiber in place. Do not overtighten, as this will damage the fiber and the port.

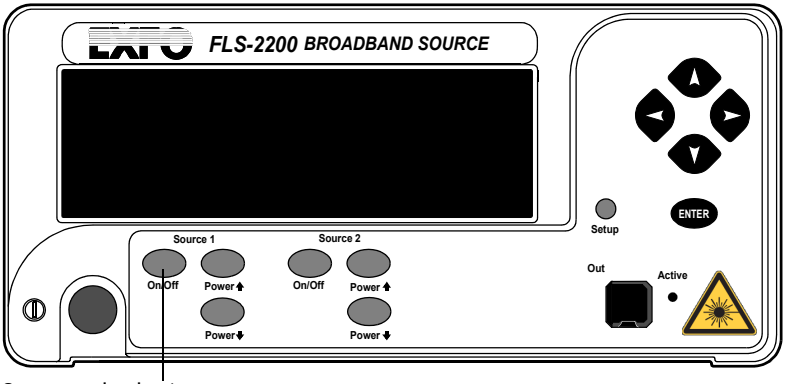
Note: *If your fiber-optic cable is not properly aligned and/or connected, you will notice heavy loss and reflection.*

EXFO uses good quality connectors in compliance with EIA-455-21A standards.

To keep connectors clean and in good condition, EXFO strongly recommends inspecting them with a fiber inspection probe before connecting them. Failure to do so will result in permanent damage to the connectors and degradation in measurements.

Activating or Deactivating a Source

On a dual-source, each one has its own set of control buttons and can be activated or deactivated separately (both ON, both OFF, one ON and one OFF).



Source activation/ deactivation button

When lit, the active LED indicates that an optical signal is being emitted from the source port.

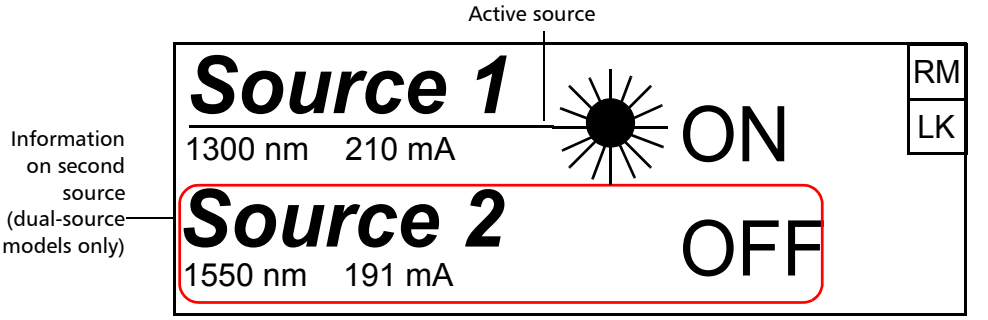
For your safety, the Broadband Source provides a three-second safety delay between the source activation and actual light emission. You can also stop light emission at any time by pressing on the activation/deactivation button.

To activate a source:

1. Press the activation/deactivation button (labeled On/Off).

Note: *If you have a dual-source, make sure you use the appropriate set of buttons (Source 1 or Source 2).*

The active LED on the unit front panel will light up, and the display will read "*Source x ON*", also showing a light beam icon. The word "*ON*" will flash during the three-second safety delay.



2. If you have a dual-source, you can repeat step 1 to activate the second one.

IMPORTANT

To obtain optimum stability, let the source warm up for 30 minutes.

To deactivate a source:

1. Press the activation/deactivation button (labeled On/Off).

The active LED on the module front will turn off and the display will read "*Source x OFF*".

2. If you have a dual-source, you can repeat the above step to deactivate the second one source.

6 Preparing for Remote Control

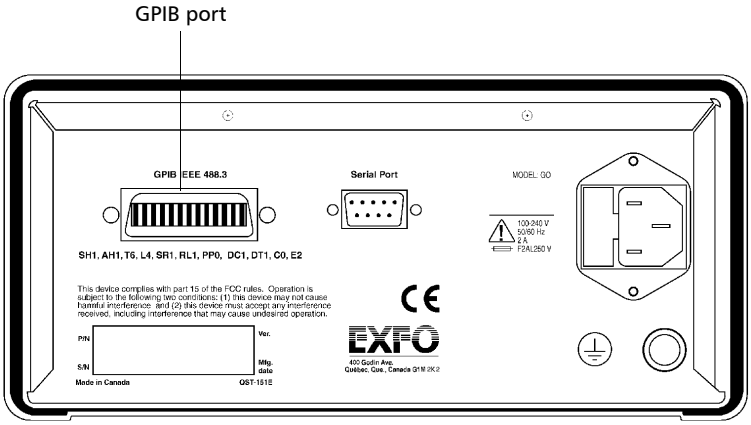
The Broadband Source can be controlled remotely either by GPIB or RS-232.

Note: *When the Broadband Source is being controlled remotely,* **RM** *appears in the upper right corner of the display.*

EXFO supplies commands that follow the guidelines determined by the SCPI consortium. The same commands are used in both GPIB and RS-232 communication. You can find detailed information about these commands in the *IEEE 488.2 and Specific Command Reference* appendix.

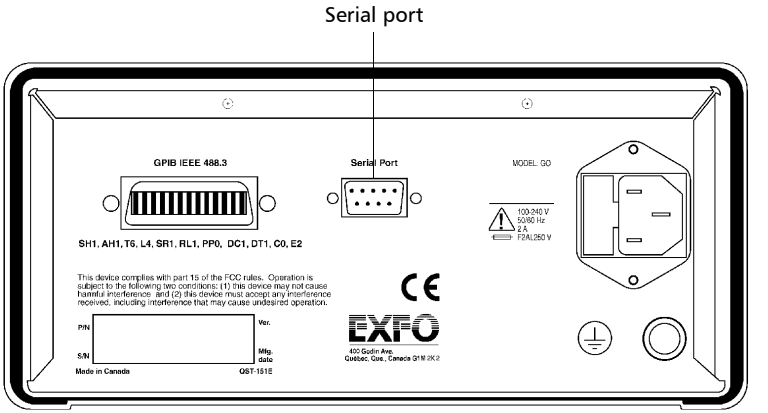
Linking Units with the GPIB Port

Your FLS-2200 Broadband Source is equipped with a GPIB port. You can simply use a GPIB cable to link it to the other unit with which you want to perform remote control. The GPIB port is located at the back of Broadband Source, as shown below.

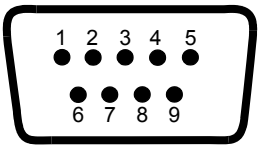


Linking Units with the Serial Port

Your FLS-2200 Broadband Source is equipped with a serial (RS-232) port to send and receive data. You can simply use a null-modem (serial) cable to link it to the other unit with which you want to perform remote control. The RS-232 port is located at the back of Broadband Source, as shown below.



The RS-232 connector uses a DTE pinout configuration.



Pin	Description	Direction
2	Receive (Rx)	Input
3	Transmit (Tx)	Output
5	Ground (Gnd)	—

Changing Communication Settings

To remotely control your Broadband Source, you must set a GPIB address or activate the RS-232 port.

Note: *Communication settings cannot be modified without turning on your unit.*

The tables present the different parameters for GPIB and RS-232 communication and their corresponding values.

Note: *EOS means "End of String." EOI means "End of Identify."*

Parameters for GPIB Communication	
Terminate Read on EOS	Yes
Set EOI with EOS on Writes	Yes
Type of compare on EOS	8 bits
EOS byte	0Ah
Send EOI at end of Writes	Yes
GPIB primary address	Value between 1 and 30 (default value: 12)
GPIB secondary address	None

Parameters for RS-232 Communication	
EOS bytes	0Ah
Baud rate	1200/2400/4800/9600/19200 bps
Parity	None
Data bits	8 bits
Stop bits	1 bit
Flow control	Software (Xon/Xoff) or None

Setting the Remote Control Mode

To remotely control the Broadband Source, you must either

- ➤ select the GPIB mode by setting a GPIB address, or
- ➤ activate the RS-232 port and set its parameters (see *Setting Baud Rate* on page 32 and *Setting Flow Control* on page 33).

To set a remote control mode:

1. Press the Setup button.
2. Use the up/down or left/right arrow buttons to select **RS232/GPIB**.

Refresh Rate	8 Hz	RS232 / GPIB	GPIB
Backlight	ON	GPIB Addr.	12
Contrast	▼ ◐ ▲	Baud Rate	N.A.
Video Mode	STD	Flow Ctrl	N.A.
Exit			

Note: *If* GPIB *is currently selected and you want to specify a GPIB address, see* *Setting GPIB Address* on page 31.

3. Press ENTER to access the **RS232 / GPIB** edit box.
4. Use the up/down arrow keys to toggle between *GPIB* and *RS232*.
5. Press ENTER to confirm.

If you selected *RS232*, the **GPIB Addr.** menu option is deactivated ("*N.A.*" is displayed).

If you selected *GPIB*, the **Baud Rate** and **Flow Ctrl** menu options are disabled ("*N.A.*" is displayed). If the currently selected GPIB address doesn't suit your needs, you can change it.

Setting GPIB Address

If GPIB is selected as the remote command mode, you can select the GPIB address you want to use from 1 to 30 (default value is 12).

To set a GPIB address:

1. Press the Setup button.
2. Use the up/down or left/right arrow keys to select **GPIB Addr.** The current GPIB address is displayed.

Note: *If you are in RS-232 mode, the GPIB address cell will display "*N.A.*" You must change the communication mode to* GPIB *before setting an address.*

Refresh Rate	8 Hz	RS232 / GPIB	GPIB
Backlight	ON	GPIB Addr.	12
Contrast	▼ ◐ ▲	Baud Rate	N.A.
Video Mode	STD	Flow Ctrl	N.A.
Exit			

3. Press ENTER, then use the up/down arrow keys to select a GPIB address between 1 and 30.
4. Press ENTER to confirm your choice.

Setting Baud Rate

The baud rate is a parameter related to RS-232 communication. It determines the speed at which data is sent between the unit and a computer, in bits per second (bps).

To change the baud rate for your remote communications:

1. Press the Setup button.
2. Use the up/down or left/right arrow keys to select **Baud Rate**. The current setting is displayed.

Note: *If you are in GPIB mode, the* **Baud Rate** *cell will display "*N.A.*" You must change the communication mode to* RS-232 *before setting the baud rate.*

Refresh Rate	8 Hz	RS232 / GPIB	RS232
Backlight	ON	GPIB Addr.	N.A.
Contrast	▼ ◐ ▲	Baud Rate	19200
Video Mode	STD	Flow Ctrl	Soft
Exit			

3. Press ENTER, then use the up/down arrow keys to select the baud rate. You can select 1200, 2400, 4800, 9600 or 19200 bps.
4. Press ENTER to confirm.

Setting Flow Control

The flow control parameter applies only to RS-232 communication. This parameter allows you to select the type of serial communication used.

You can choose the *Soft* option if you want the rate of data transmission to match the rate at which it can be processed by the device. This enables the computer and the Broadband Source to stop each other from transmitting by sending a control character (Xoff). They will also be able to restart the transmission by sending another control character (Xon). This is known as a "software handshake".

To set a flow control:

1. Press the Setup button.
2. Use the up/down or left/right arrow keys to select **Flow Ctrl**. The current setting is displayed.

Note: *If you are in GPIB mode, the* **Flow Ctrl** *cell will display "*N.A.*" You must change the communication mode to* RS-232 *before setting the flow control.*

Refresh Rate	8 Hz	RS232 / GPIB	RS232
Backlight	ON	GPIB Addr.	N.A.
Contrast	▼ ◐ ▲	Baud Rate	19200
Video Mode	STD	Flow Ctrl	Soft
Exit			

3. Press ENTER, then use the up/down arrow keys to select the type of flow you want. "*None*" means no flow control. "*Soft*" allows the unit or computer controlling it, to turn the data transmission on or off.
4. Press ENTER to confirm.

7 Using Your Broadband Source in an Automated Test Environment

EXFO supplies commands that follow the guidelines determined by the SCPI consortium and LabVIEW drivers for your FLS-2200 Broadband Source. Your application can be developed using LabVIEW.

The present chapter gives you information to help you use the provided commands and drivers to remotely control your Broadband Source.

Message Management

Each device that is physically connected to the GPIB link has its own input buffer, output queue and error/event queue. These data structures allow storage of incoming messages (single or compound commands that are sent to an instrument), responses from queries, errors and events that may occur.

Data structure	Characteristics	Clearing
Input buffer	➤ Consists of a First-In, First-Out (FIFO) data structure. ➤ Stores Data Bytes (DABs) and END messages. ➤ Delivers messages to the parser in the order that they were received from the I/O control. ➤ Maximum message length: unlimited in DABs (the input buffer size is only limited to the total size of the device memory).	The buffer will be cleared by: ➤ Turning off the power. ➤ Sending a Device Clear (DCL) message to the instrument. ➤ Sending a Selected Device Clear (SDC) message to the instrument.

Data structure	Characteristics	Clearing
	➤ An incoming byte empties the output queue. An error will be raised if the output queue contained data. Consequently, it clears the Message AVailable bit (bit number 4 –MAV from the Status Byte register). ➤ Except for the string and binary block contents, the following transformations are made on the incoming data: character conversion from lower case to upper case. conversion of "<wsp>" characters to spaces. conversion of multiple blanks to a single blank. ➤ Parser begins to process messages when the <PROGRAM MESSAGE TERMINATOR> is received or if the input buffer is full.	

Data structure	Characteristics	Clearing
Output queue	➤ Consists of a First-In, First-Out (FIFO) data structure. ➤ When the instrument acts as a talker, it sends response messages (from the output queue) to the controller. Response messages all end with a *<RESPONSE MESSAGE TERMINATOR>*, see the appendix on data types. ➤ Total storage capacity: only limited to the device's memory. ➤ As soon as there is data in the output queue, the Message AVailable bit (bit number 4 –MAV from the Status Byte register) is set to 1. ➤ Remains empty if no query is received or if the query contains an error.	The Output queue will be cleared by: ➤ Reading all the items it contains. ➤ Turning off the power. ➤ Sending a Device Clear (DCL) message to the instrument. ➤ Sending a Selected Device Clear (SDC) message to the instrument. ➤ Attempting to send a command before reading the responses to previous queries (an error will also be raised).

Data structure	Characteristics	Clearing
Error/Event queue	➤ Consists of a First-In, First-Out (FIFO) data structure. ➤ Total storage capacity: 50 errors or events. ➤ Errors or events can be retrieved, one at a time, with *:SYSTem:ERRor[:NEXT]?*. ➤ When an error or event occurs and the Error/Event queue is full, the last item in the queue (the most recent) is removed and the Queue overflow error (error –350) is added. No new items will be stored into the queue until there will be room available. ➤ As soon as there is data in the output queue, the Error AVailable bit (bit number 2 –EAV from the Status Byte register) is set to 1.	The Error/Event queue will be cleared when: ➤ Reading all the items it contains. ➤ Turning off the power. ➤ Sending a Device Clear (DCL) message to the instrument. ➤ Sending a Selected Device Clear (SDC) message to the instrument. ➤ Using the *CLS command.

Standard Status Data Structure

Each device that is physically connected to the GPIB bus has four status registers with a structure complying with the IEEE 488.2 standard. These registers allow the controller to monitor events and get useful information on the status of the devices it controls.

➤ Standard Event Status Register (ESR)

Bits	Mnemonics	Bit Value
7	Power On	128
6	Not used	0
5	Command Error	32
4	Execution Error	16
3	Device Dependent Error	8
2	Query Error	4
1	Not used	0
0	Operation Complete	1

➤ Standard Event Status Enable Register (ESE)

Bits	Mnemonics	Bit Value
7	Power on	128
6	Not used	0
5	Command error	32
4	Execution error	16
3	Device dependent error	8
2	Query error	4
1	Not used	0
0	Operation complete	1

The following table presents a summary of the possible operations on ESR and ESE registers.

Register	Read	Write	Clear
ESR	Use *ESR?.	Impossible to write.	➤ Use *CLS. ➤ Read the register.
ESE	Use *ESE?.	Use *ESE.	Use *ESE with a value equal to 0.

➤ Status Byte Register (STB)

Bits	Mnemonics	Bit Value
7	Not used	0
6	Request service / Master summary status	64
5	Event summary bit	32
4	Message available	16
3	Not used	0
2	Error / Event queue	4
1	Not used	0
0	Not used	0

➤ Service Request Enable Register (SRE)

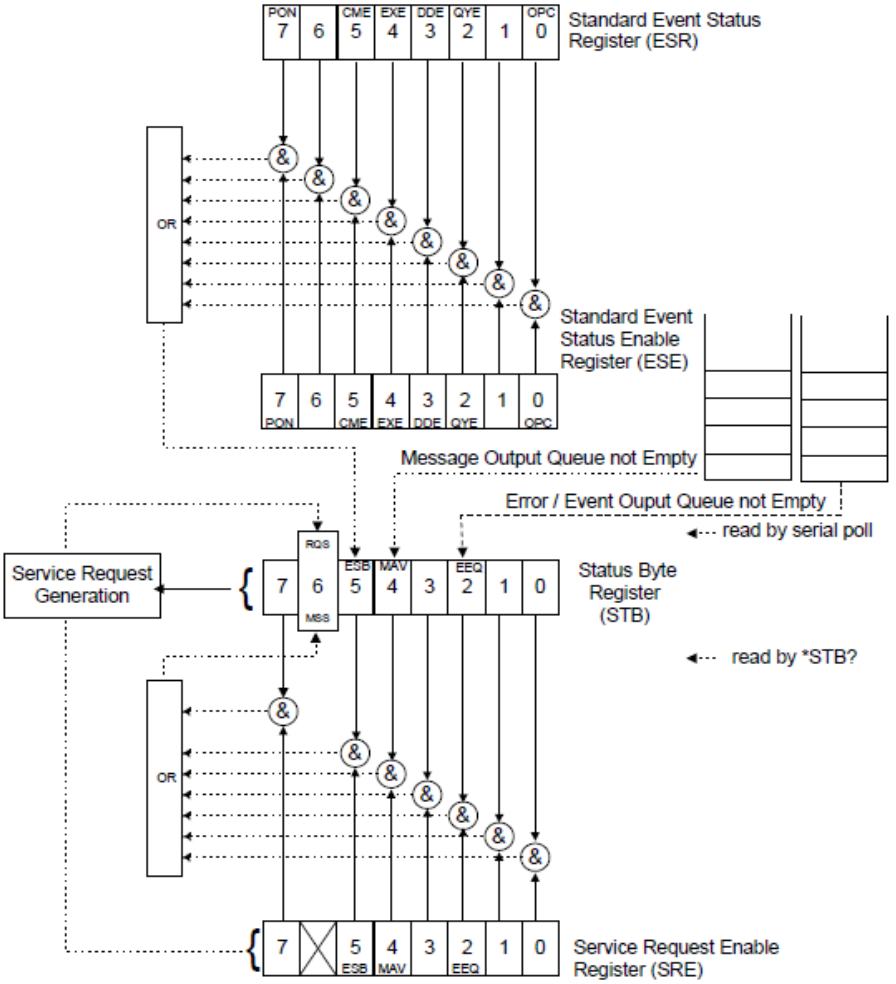
Bits	Mnemonics	Bit Value
7	Not used	0
6	Reserved	0
5	Event status byte	32
4	Message available	16
3	Not used	0
2	Error / Event queue	4
1	Not used	0
0	Not used	0

The following table presents a summary of the possible operations on STB and SRE registers.

Register	Read	Write	Clear
STB	➤ Use *STB?. ➤ Use serial poll (GPIB bus sequence that allows retrieval of the value without interrupting the current process).	Impossible to write; the register's contents is only modified when the Event registers or Queues are modified.	Use *CLS before sending a query (to clear the Event registers and Queues and by the same token clear the STB register).
SRE	Use *SRE?	Use *SRE with a value equal to 0 to disable the register or with a value equal to 1 to enable it.	Use *SRE with a value equal to 0.

The diagram displayed on the next page is a useful aid in understanding the general commands and how a service request (SRQ) is generated.

Using a service request, a device notifies the controller that an event requiring special attention occurred. The controller will then find which device generated a SRQ (its RQS bit is set) and the causes of it.


PON 7
6
CME 5
EXE 4
DDE 3
QYE 2
1
OPC 0
Standard Event Status Register (ESR)
OR
Standard Event Status Enable Register (ESE)
7 PON
6
5 CME
4 EXE
3 DDE
2 QYE
1
0 OPC
Message Output Queue not Empty
Error / Event Ouput Queue not Empty
read by serial poll
Service Request Generation
RQS
6 MSS
ESB 5
MAV 4
EEQ 2
Status Byte Register (STB)
read by *STB?
OR
7
5 ESB
4 MAV
3
2 EEQ
1
0
Service Request Enable Register (SRE)

SCPI Command Structure

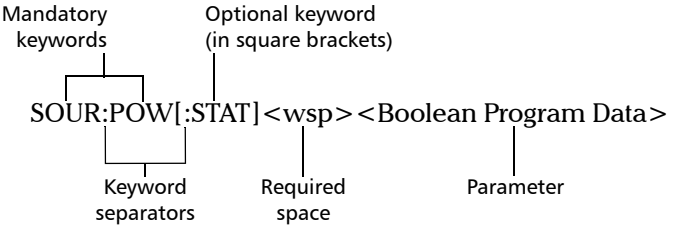
The information presented in this section provides an overview of GPIB programming. If you need detailed information, refer to:

- The International Institute of Electrical and Electronics Engineers. *IEEE Standard 488.1-1987, IEEE Standard Digital Interface for Programmable Instrumentation*. New York, 1987.
- The International Institute of Electrical and Electronics Engineers. *IEEE Standard 488.2-1992, IEEE Standard Codes, Formats, Protocols and Common Commands For Use with ANSI/IEEE Std. 488.1-1987*. New York, 1992.
- *Standard Commands for Programmable Instruments (SCPI). Volume 1: Syntax and Style*. Vers. 1999.0 May, U.S.A, 1999.

The provided commands follow the guidelines determined by the Standard Commands for Programmable Instruments (SCPI) consortium. A *program message* consists of one or more commands (and/or queries) with their appropriate parameters.

Note: *The command provided below is for guidance only; your Broadband Source may not support it.*

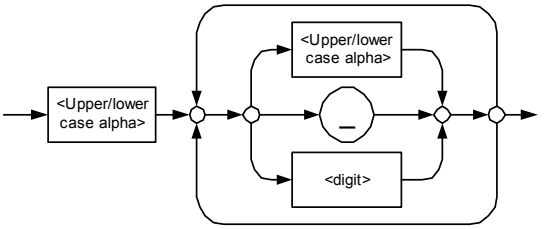
For example, a program message could contain a command used to activate or deactivate a source. The corresponding command syntax would be:



When sending a message containing the previous command, you would actually type: SOUR:POW ON.

The following table shows elements that are commonly used in the commands or queries syntax.

Item	Meaning
[ ]	Enclose optional keywords or parameters. *Do not include square brackets in your program message.*
[1..n]	Indicates that the instrument provides multiple capabilities and that you have to specify which one you want to use. If you omit the value, the command will take effect on the first capability. Multiple capabilities can be found at any branch of the command tree (root, intermediate node or terminal node). Example: If the command is :SENSe*[1..n]*:CORRection:COLLect:ZERO and you want it to take effect on the second SENSe (sensor) capability of the instrument, you may send this: :SENSe*2*:CORRection:COLLect:ZERO. *Do not include square brackets in your program message; simply enter the number.*
<wsp>	Indicates that a space is required ("wsp" stands for "white space"). Corresponds to ASCII character codes (0 to 9 and 11 to 32, in decimal). *Do not include* "<wsp>" *in your program message; simply type a space.*
<digit>	Element used in the construction of various numeric data types. Can take any value between 0 and 9 inclusively (corresponds to ASCII character codes 48 to 57, in decimal).

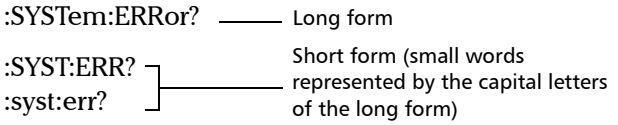
Item	Meaning
<mnemonic>	Element used in the construction of certain data types and program messages.  In the diagram above, ➤ "<Upper/lower case alpha>" corresponds to ASCII character codes (65 to 90 and 97 to 122, in decimal). ➤ "_" corresponds to an underscore character (code 95, in decimal).
< >	Text appearing between angled brackets specifies the command parameter to be sent or the response you will receive from an instrument. *Do not include angled brackets in your program message.*
|	Indicates that one, and only one, value must be selected from the available choices. Example: If the list is 0|1, you can only select 0 *or* 1. *Do not include the pipe character in your program message.*
{ }	Indicate that the enclosed parameters can appear 0 to n times when the command is used. *Do not include braces in your program message.*
:	Mandatory to separate keywords. Can be omitted at the beginning of a program message. For example, you can use either :SYST:ERR or SYST:ERR.

Item	Meaning
;	➤ Mandatory to separate the different commands of a program message when more than one command is sent at a time. In this case, it is called *<PROGRAM MESSAGE UNIT SEPARATOR>*. ➤ Also used to separate responses when multiple queries were sent in a single program message. In this case, it is called *<RESPONSE MESSAGE UNIT SEPARATOR>*.
,	➤ Mandatory to separate parameters in a command or a query. In this case, it is called *<PROGRAM DATA SEPARATOR>*. ➤ Also used to separate the various responses from a query. In this case, it is called *<RESPONSE DATA SEPARATOR>*.

There are also several conventions regarding command syntax:

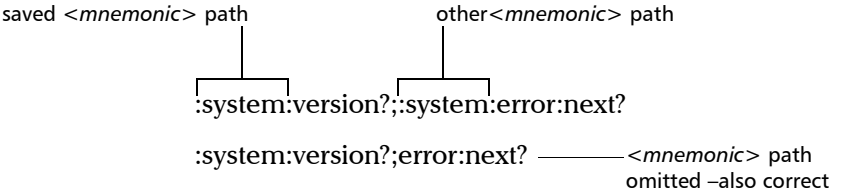
- Spelling errors will cancel the command or query.
- Commands and queries are not case-sensitive. You can type your program messages using either lower-case or upper-case letters.
- The command or query can be written using only the three- or four-letter shortcuts, only full words, or a combination of both.

  The example below shows the long and the short forms of a same query.

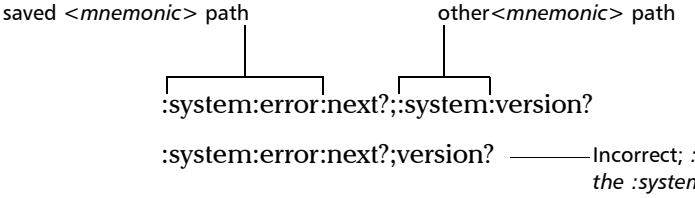


- For readability reasons, you can use extra spaces in your program messages but they won't be taken into account. For more information, see *Message Management* on page 35.

➤ You can build program messages allowing you to send more than one command at a time. Sometimes, you can omit the leading <*mnemonic*> path to simplify the program messages and speed up the search time (the parser saves the last position in the command tree).



Paths cannot be omitted in all cases. The example below would cause an error.



IEEE 488.2 required commands or queries (beginning with a *) that are part of the program message have no effect on the paths.

Example:

*:system:version?;*idn?;:system:error:next?* is equivalent to

*:system:version?;*idn?;error:next?*

Note: *Omitting the leading* <mnemonic> *path is only possible when you have more than one command or query in the program message that you send.*

➤ A program message must be ended with a <PROGRAM MESSAGE TERMINATOR>. For more information, see the appendix on data types.

Consulting Data Types

If you need information about data types used in EXFO's documentation, see the appendix on data types.

Writing Remote Control Code

Complex measurement programs may be written using any programming environment that supports GPIB communication. GPIB development kits are available for most of the popular commercial programming languages.

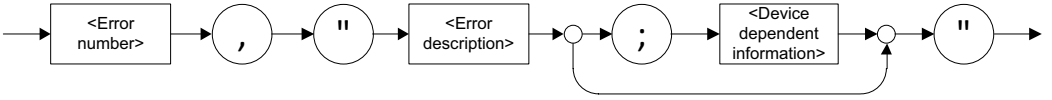
You can find all the commands and queries supported by the Broadband Source in the *IEEE 488.2 and Specific Commands* appendix.

When you write code, you must follow these rules on message reception and transmission:

- The controller must have sent a complete message to the instrument (including the message terminator) before retrieving a response.
- The controller must retrieve all the responses from previous queries (including the response terminator) before sending a new message to an instrument.
- The controller must not try to retrieve a response from an instrument if the corresponding query has not been previously sent to the instrument.
- You must pay special attention to queries that return an indefinite ASCII response. To avoid any confusion, the IEEE 488.2 standard requires that this data type be immediately followed by a response termination character. For this reason, when working with compound queries, you must ensure that a query sending an indefinite ASCII response is the last query of the series.
- Be careful when sending program messages containing multiple queries that return large amounts of data. Since the controller can only retrieve data when the instrument has finished processing the queries, it could result in problems ranging from a saturation of the output queue to the complete blocking of the whole system.

Error Message Format

System and device-specific errors are managed by the FLS-2200 Broadband Source. The generic format for error messages is illustrated in the following figure.



As shown in the above figure, the message contains three parts:

- error number
- error description
- device-dependent information

Error messages ending in a negative number are SCPI-based errors.

For more information on errors, see *Message Management* on page 35. For a complete list of possible errors, seethe appendix on SCPI-based errors.

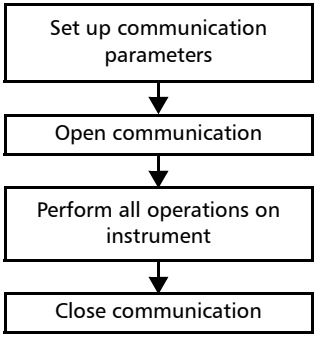
Working with EXFO LabVIEW Drivers

EXFO provides you with custom drivers that you can use to program commands for your inspection instruments.

IMPORTANT

You need to be familiar with the LabVIEW environment and programming methods to work with EXFO drivers.

Regardless of whether you work with the provided Getting Started applications or your own VIs (using EXFO drivers), the steps remain the same.



Before configuring the communication parameters via LabVIEW (provided applications or new VI), you must configure the FLS-2200 Broadband Source for remote control. For more information, see the section on preparing your unit for automation or remote control in this user documentation.

The following table presents the possible settings for communication parameters. These parameters must be set from LabVIEW for each instrument.

Parameter	RS-232	GPIB
Communication type	RS232	GPIB
VISA resource name	Select the serial port from the list	Select the GPIB device from the list

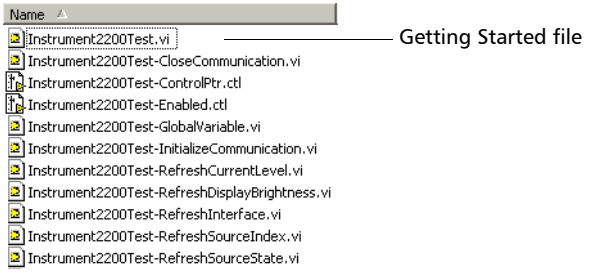
Using the EXFO Getting Started Applications

Once the LabVIEW drivers are installed, the Getting Started demo applications are available to demonstrate the following:

- How to open and close the communication link between the remote computer and the device.
- Some of the available functions (by loading the necessary *.vi* files).

All the *.vi* files related to an instrument are presented in the same folder. By default, they can be found under:

C:\Program Files\EXFO\LabVIEW Getting Started\Getting Started xxxx (where xxxx corresponds to the product code).

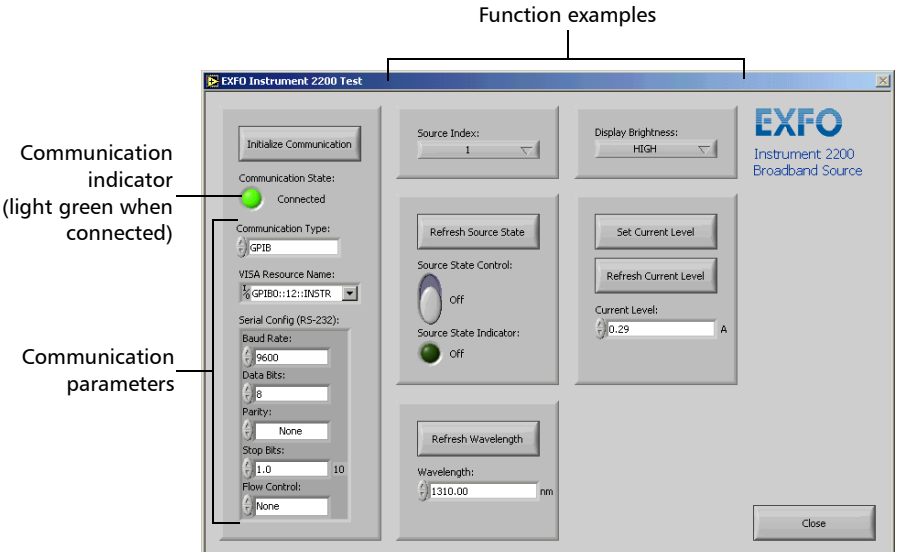


You can also directly start a demo application this way:

From the Windows task bar, click the **Start** button, then point to **All Programs** > **EXFO** > **LabVIEW Getting Started Applications**, and click **Getting Started** xxxx (where xxxx corresponds to the product code).

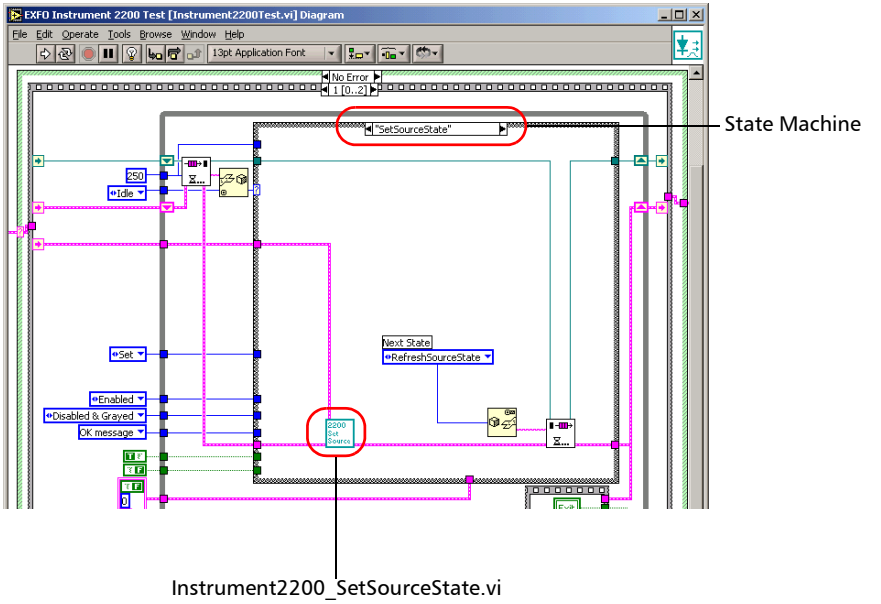
Each Getting Started application offers a user interface (called Front Panel and a design view (called Block Diagram).

On the Front Panel, you can set communication parameters between the computer and the current instrument. It also offers various controls and buttons to use the instrument easily. In fact, the application performs the necessary calls to the instrument's drivers so it is transparent to the user.

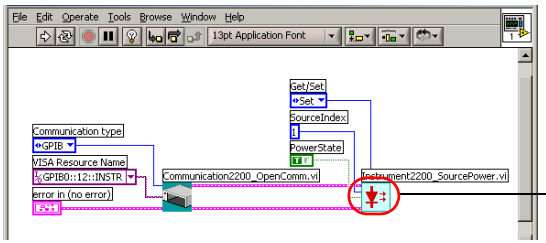


The application state (called State Machine) changes whenever an action is performed on the instrument. If you toggle to Block Diagram view, you can see the list of possible states. The application is always in one of the predefined states.

The following figure illustrates the State Machine after the user has clicked on the button allowing you to set the source state (from the Front Panel). When the State Machine changes to “SetSourceState”, the application calls “Instrument2200_PowerSource.vi”, which, in turn, calls the “SourcePower.vi”sub VI that will perform the appropriate action on the instrument.

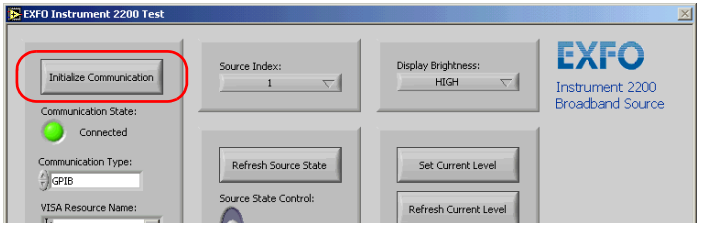


The detail of this sub VI gives precious information on how to call an instrument driver VI.



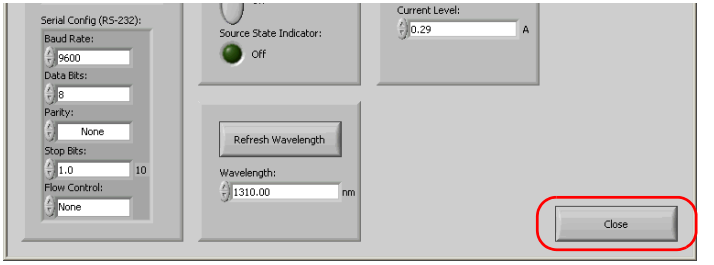
To use a Getting Started application:

1. Turn on the computer and ensure that all the remote-control parameters are set correctly.
2. Open the desired Getting Started application and run it from LabVIEW.
3. From the application's Front Panel, set the communication parameters. For information on communication parameters, see *Working with EXFO LabVIEW Drivers* on page 52.
4. Once the parameters are configured, click **Initialize Communication**.



5. Using the provided buttons and controls, perform the desired actions.

6. When you are finished, select **Close** to end the communication.



7. Close LabVIEW.

IMPORTANT

To avoid losing the original version of the Getting Started applications, do not save changes when prompted by LabVIEW.

Building and Using Custom VIs

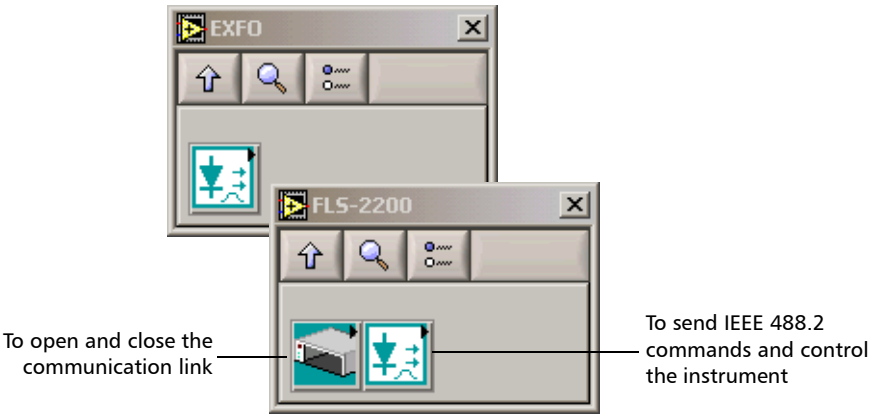
EXFO LabVIEW drivers have been designed to let you control the various instruments according to your needs, by building your own VIs in LabVIEW.

You can access EXFO drivers

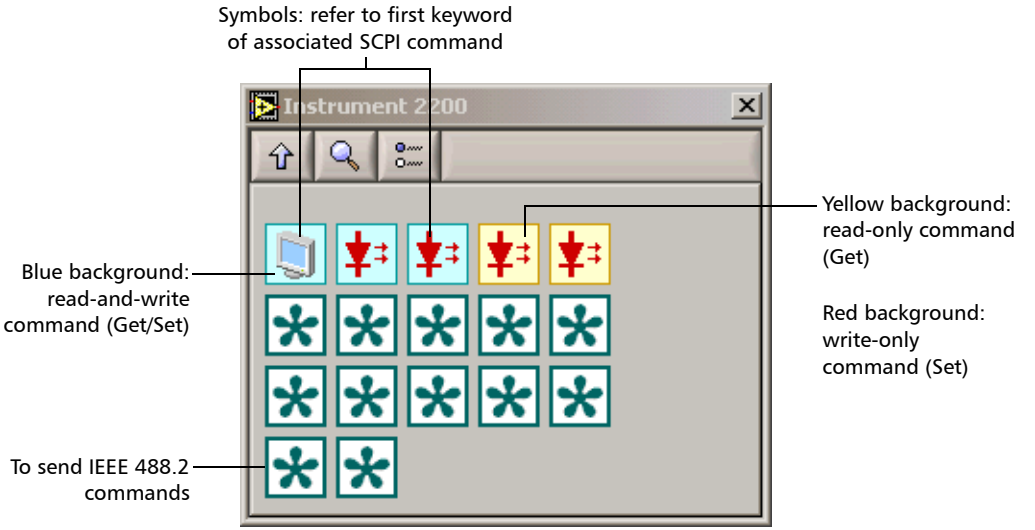
- directly from
  C:\Program Files\National Instruments\LabVIEW 6\instr.lib\EXFO
- from the LabVIEW function palettes

The EXFO palette gives you access to a sub-palette in which each icon corresponds to a set of drivers that allow you to either

- communicate with the FLS-2200 Broadband Source
- open and close communication links with the FLS-2200 Broadband Source
- send IEEE 488.2 (common) commands



When you click an icon in the palette, the corresponding sub-palette opens, giving you access to the different functions.



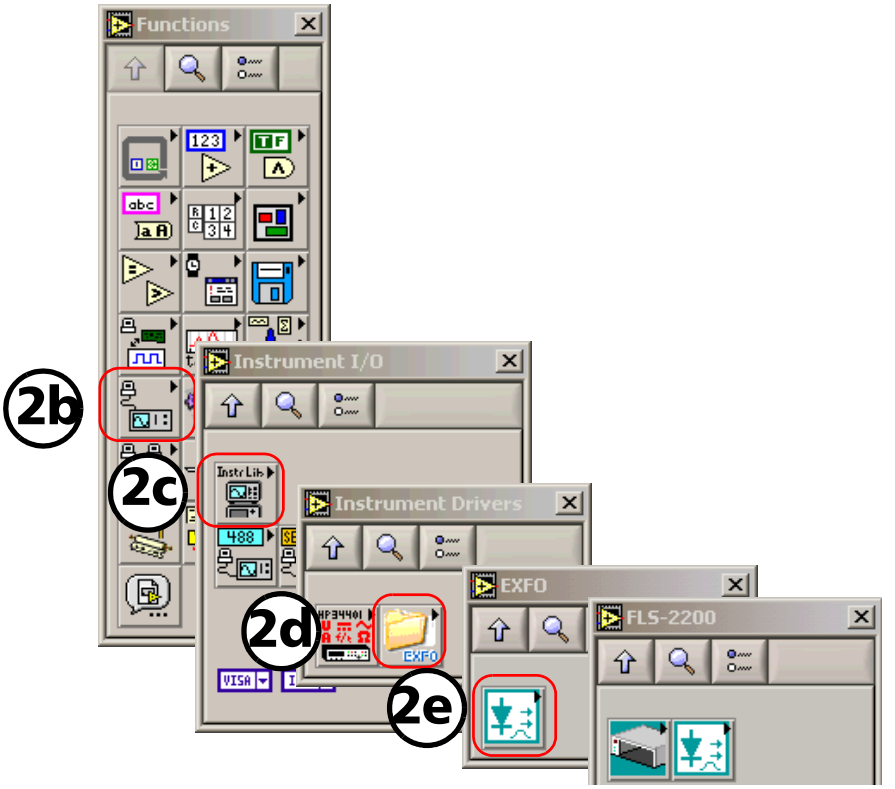
To build a custom VI:

1. Start LabVIEW and create a new VI.

2. Open the **EXFO** palette.

2a. From LabVIEW, open the Diagram Block view.

2b. Display the **Functions** palette and select **Instrument I/O**.

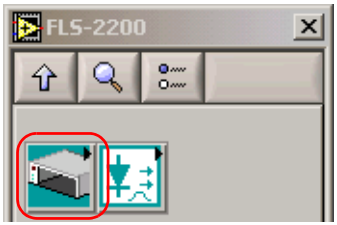


2c. From the **Instrument I/O** palette, select **Instrument Drivers**.

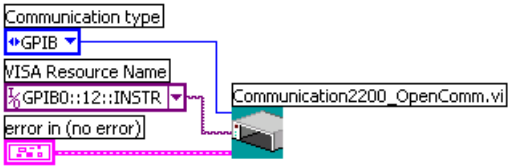
2d. From the **Instrument Drivers** palette, select **EXFO**.

2e. From the **EXFO** palette, select the icon corresponding to the FLS-2200 Broadband Source.

3. Select **EXFO Communication 2200**.

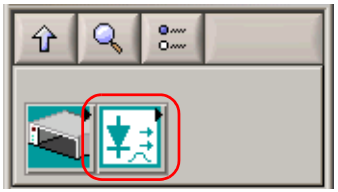


4. From the **EXFO Communication 2200** palette, select *Communication2200_OpenComm.vi* and add it to your new VI.



5. Set the communication parameters. For information on communication parameters, see *Working with EXFO LabVIEW Drivers* on page 52.

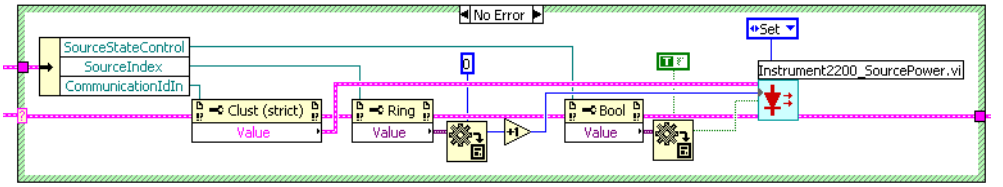
6. From the EXFO palette, select the icon corresponding to the functions of the FLS-2200 Broadband Source.



7. From the displayed palette, select the function you need and add the corresponding driver to your VI.

8. Set the required parameters and connect the instrument *Communication ID in* parameter to the *Communication ID out* parameter from CommunicationXXXX_OpenComm.vi.

The example below shows how to configure the SourcePowerState_GSet.vi to turn on the source.In this example, *Set* was chosen and the *PowerState* parameter was set to *True*.

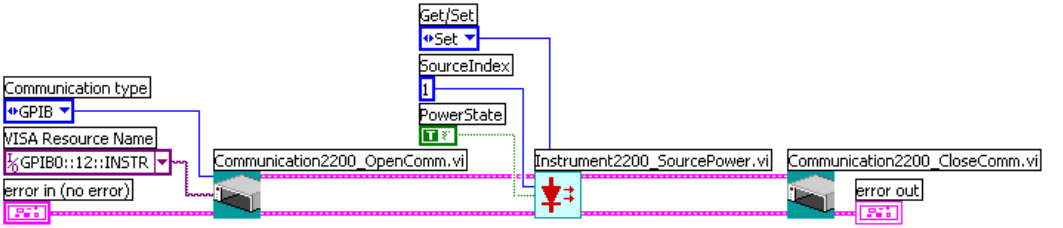


9. Repeat steps 7 and 8 for each of the functions you want to use.

However, you have to link *Communication ID in* of the new driver to *Communication ID out* of the preceding driver.

Note: *If you want to use IEEE 488.2 commands, add the desired driver to your VI and configure its parameters exactly as you would do with any instrument function.*

10. When you are finished, add *CommunicationXXXX_CloseComm.vi* to your VI.



Connect the *Communication ID out* parameter of the last function to the *Communication ID in* parameter of CommunicationXXXX_CloseComm.vi.

Note: *You only have to open communication once at the beginning, and close it when all of the desired functions will have been added.*

11. Save your work.

To use your new VI:

1. Turn on the computer and ensure that all the remote-control parameters are set correctly.

2. From LabVIEW, run the VI.

8 Maintenance

To help ensure long, trouble-free operation:

- Always inspect fiber-optic connectors before using them and clean them if necessary.
- Keep the unit free of dust.
- Clean the unit casing and front panel with a cloth slightly dampened with water.
- Store unit at room temperature in a clean and dry area. Keep the unit out of direct sunlight.
- Avoid high humidity or significant temperature fluctuations.
- Avoid unnecessary shocks and vibrations.
- If any liquids are spilled on or into the unit, turn off the power immediately, disconnect from any external power source, remove the batteries and let the unit dry completely.



WARNING

The use of controls, adjustments and procedures other than those specified herein may result in exposure to hazardous situations or impair the protection provided by this unit.

Cleaning EUI Connectors

Regular cleaning of EUI connectors will help maintain optimum performance. There is no need to disassemble the unit.



IMPORTANT

If any damage occurs to internal connectors, the module casing will have to be opened and a new calibration will be required.

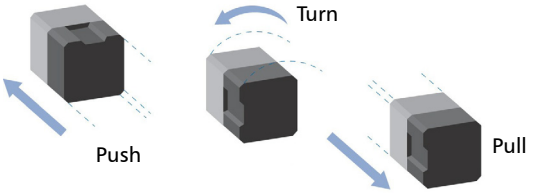


WARNING

Looking into the optical connector while the light source is active WILL result in permanent eye damage. EXFO strongly recommends to TURN OFF the unit before proceeding with the cleaning procedure.

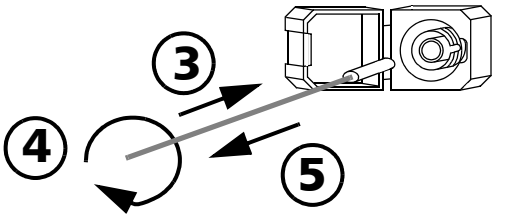
To clean EUI connectors:

1. Remove the EUI from the instrument to expose the connector baseplate and ferrule.



2. Moisten a 2.5 mm cleaning tip with *one drop* of isopropyl alcohol (alcohol may leave traces if used abundantly).

3. Slowly insert the cleaning tip into the EUI adapter until it comes out on the other side (a slow clockwise rotating movement may help).



4. Gently turn the cleaning tip one full turn, then continue to turn as you withdraw it.

5. Repeat steps ? to ? with a dry cleaning tip.

Note: *Make sure you don't touch the soft end of the cleaning tip.*

6. Clean the ferrule in the connector port as follows:

6a. Deposit *one drop* of isopropyl alcohol on a lint-free wiping cloth.

IMPORTANT

Isopropyl alcohol may leave residues if used abundantly or left to evaporate (about 10 seconds).

Avoid contact between the tip of the bottle and the wiping cloth, and dry the surface quickly.

6b. Gently wipe the connector and ferrule.

6c. With a dry lint-free wiping cloth, gently wipe the same surfaces to ensure that the connector and ferrule are perfectly dry.

6d. Verify connector surface with a portable fiber-optic microscope (for example, EXFO's FOMS) or fiber inspection probe (for example, EXFO's FIP).

7. Put the EUI back onto the instrument (push and turn clockwise).

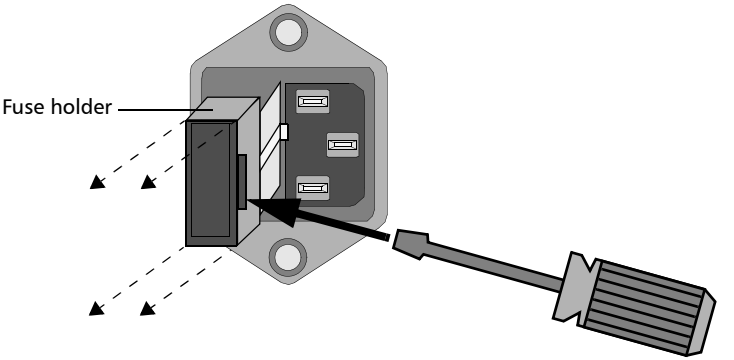
8. Throw out cleaning tips and wiping cloths after one use.

Replacing Fuses

The FLS-2200 Broadband Source contains two F2.0L250V-type fuses (5 mm x 20 mm (0.197 in x 0.787 in), fast-acting, low breaking capacity, 250 V). The fuse holder is located at the back of the Broadband Source, just beside the power inlet.

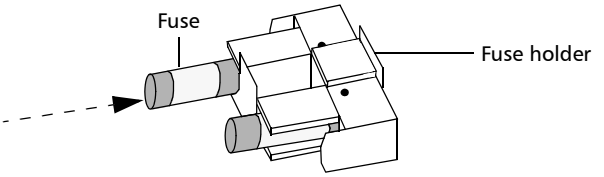
To replace a fuse:

1. Turn off the unit and disconnect the power cord.
2. Using a flat-head screwdriver as a lever, pull out the fuse holder.



3. Verify and replace the fuses as necessary.

4. Insert the new fuses into the fuse holder.



5. Ensure the fuses are placed firmly in the holder before reinstalling it in the unit.

6. Firmly push the fuse holder back into place.

Upgrading the Embedded Software

To upgrade the Broadband Source embedded software, you will need to obtain the upgrade files from EXFO's Technical Support Group. You will also need a null-modem cable.

IMPORTANT

You may upgrade software under DOS, Windows 3.1, Windows 9x, or Windows 2000. With some notebook computers, you may need to be under a DOS environment. If problems occur, contact EXFO.

To upgrade the embedded software:

1. Turn off the Broadband Source.

2. Connect a null-modem cable to the Broadband Source RS-232 port and to an unused serial communication port on your computer.

3. On your computer's hard disk, create a folder named "Test" (C:\Test).

4. Unzip or copy the upgrade files into the newly created folder.

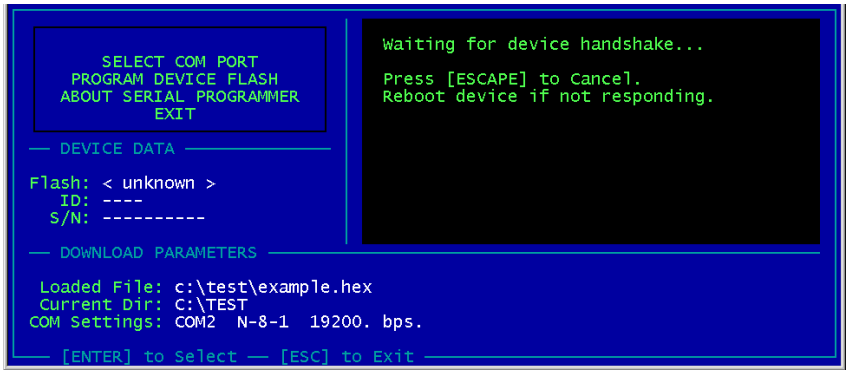
5. If the software upgrade is performed under Windows 98, you must restart your computer in DOS mode before starting the upgrade program. In other cases, simply exit to DOS.

6. Go to the "C:\Test" folder and start the upgrade program by typing the following line (spaces are required between parameters):

   Lo0006.exe /C:2 /F:c:\test\filename.hex /S:19200

   Parameters can be decoded as follows:

   - /C: serial port number (COM2 in the above example)
   - /F: file to copy on your unit (replace "filename" with the actual name of the .hex file on your hard disk)
   - /S: computer-to-unit transfer speed (if "19200" does not work, try "56700")



7. When a message about waiting for a device handshake appears, turn on the Broadband Source.

   The unit display remains off, the unit beeps once and the upgrade program starts automatically. A progress bar on the computer screen indicates the upgrade status. Once the software upgrade is complete, the message about restarting the unit appears.

8. If the software upgrade was performed under Windows 2000, an error message to the effect that the *LO0006 NTVDM has encountered a system error and to select close to terminate the application* is displayed. Click **Close** to hide the dialog box.

9. Turn the Broadband Source off, and then on again.

Some units will display the new version number at startup, otherwise press the up and right arrow keys together while the unit is turned on.

Recycling and Disposal (Applies to European Union Only)

For complete recycling/disposal information as per European Directive WEEE 2012/19/UE, visit the EXFO Web site at www.exfo.com/recycle.

9 Troubleshooting

Error Messages

Number	Description	Recommended Action
-11	Module reset error: The nulling was not performed correctly.	Restart your unit to solve the problem.
-12	Wrong module ID: The module returns the wrong ID.	Contact EXFO for assistance.
-20	Module communication error: Communication error with the module.	Restart your unit to solve the problem.
-25	Checksum error: Checksum error while reading the module's FIFO.	Restart your unit to solve the problem.
-30	Command not accepted: The command that caused the warning will be lost. The unit may continue with the program even if the command was not performed.	If problem persists, contact EXFO for assistance.
-31	Module setting error: One of the settings sent to the unit is wrong.	Review your command before sending it again.
-32	Action currently in progress: You cannot send a command while the unit is already active.	Wait until the unit is done before sending the command.
-34	Command overflow: Too many commands were sent to the unit at a time.	Wait until the unit is done before sending more commands.

Number	Description	Recommended Action
-40	FIFO not ready for reading: The unit's FIFO is not ready for reading. Commands sent will be ignored.	If problem persists, contact EXFO for assistance.
-60	SLED over-current: A problem occurred with the current going to the SLED.	Contact EXFO for assistance.
-64	EEPROM error: The EEPROM was not detected.	Contact EXFO for assistance.
-65	EEPROM checksum error: The unit detected a checksum error	Contact EXFO for assistance.
-68	Temperature error: The operating temperature of the unit is from 10 °C to 40 °C (50 °F to 104 °F).	Make sure the ambient temperature is comprised in the specified temperature range.
-69	FPGA problem: A FPGA problem was detected.	Contact EXFO for assistance.
-73	Supply voltage error: Internal voltage values are out of the operation range.	Contact EXFO for assistance.
-75	SLED temperature problem: SLED temperature is out of the operation range.	Make sure the ambient temperature is comprised in the specified temperature range.
-76	Thermo-electric cooler over-current: A problem occurred with the current going to the thermo-electric cooler.	Contact EXFO for assistance.

Number	Description	Recommended Action
-79	EEPROM access error: A problem occurred when accessing the EEPROM memory (read or write).	Contact EXFO for assistance.
100	No more room in the command pipe: A command could not be added to the command pipe.	If problem persists, contact EXFO for assistance.
101	Timeout error: The command request was not performed in the set time amount.	Contact EXFO for assistance.
102	Runtime error: The command was not performed by the unit due to a runtime error.	Contact EXFO for assistance.
103	Invalid response: A command has triggered an invalid response from the unit.	Contact EXFO for assistance.
200	Decompression error: An error occurred while decompressing the software.	Contact EXFO for assistance.
32244	Floating exception: There is an overflow while running a command.	Restart your unit to solve the problem.
32245	Stack overflow: Not enough RAM to run the command.	Restart your unit to solve the problem.
32246	Abort called in firmware: The unit used a command which it is not intended to.	Restart your unit to solve the problem.

Number	Description	Recommended Action
32300	Heap overflow: Not enough space in the heap.	Restart your unit to solve the problem.
32301 32302	Malloc overflow: Not enough RAM to run the command.	Restart your unit to solve the problem.
32303	Divide by zero: The unit attempted to perform a division by zero, which gives an infinite answer.	Restart your unit to solve the problem.
32304	Array boundary error: A table index is outside the boundaries set by the array.	Restart your unit to solve the problem.
32305	Invalid Opcode: The unit did not recognized the binary code.	Restart your unit to solve the problem.

Solving GPIB Common Problems

Problem	Probable Cause	Solution
Unable to communicate with Broadband Source (no response from *IDN? command).	Incorrect communication type selected.	Select the correct communication type: RS-232 or GPIB.
	Incorrect communication parameters.	Check the communication parameters: bus address, baud rate, flow control, etc., as required.
	Incorrect termination characters.	Synchronize termination characters between the GPIB controller and the SCPI Manager.
	Poor bus connection.	Ensure the functioning of the controller card and make sure that the bus cable is properly connected.
	Improper configuration.	Verify that the GPIB interface is properly configured.
Receive “Undefined header” error.	Incorrect command syntax.	Verify and correct syntax.
Unstable communication.	Incorrect termination character.	Synchronize termination characters between the GPIB controller and the SCPI Manager.

Contacting the Technical Support Group

To obtain after-sales service or technical support for this product, contact EXFO at one of the following numbers. The Technical Support Group is available to take your calls from Monday to Friday, 8:00 a.m. to 7:00 p.m. (Eastern Time in North America).

Technical Support Group
400 Godin Avenue
Quebec (Quebec) G1M 2K2
CANADA

1 866 683-0155 (USA and Canada)
Tel.: 1 418 683-5498
Fax: 1 418 683-9224
support@exfo.com

For detailed information about technical support, and for a list of other worldwide locations, visit the EXFO Web site at www.exfo.com.

To accelerate the process, please have information such as the name and the serial number (see the product identification label), as well as a description of your problem, close at hand.

Transportation

Maintain a temperature range within specifications when transporting the unit. Transportation damage can occur from improper handling. The following steps are recommended to minimize the possibility of damage:

- Pack the unit in its original packing material when shipping.
- Avoid high humidity or large temperature fluctuations.
- Keep the unit out of direct sunlight.
- Avoid unnecessary shocks and vibrations.

10 Warranty

General Information

EXFO Inc. (EXFO) warrants this equipment against defects in material and workmanship for a period ofXX Number of Years XX from the date of original shipment. EXFO also warrants that this equipment will meet applicable specifications under normal use.

During the warranty period, EXFO will, at its discretion, repair, replace, or issue credit for any defective product, as well as verify and adjust the product free of charge should the equipment need to be repaired or if the original calibration is erroneous. If the equipment is sent back for verification of calibration during the warranty period and found to meet all published specifications, EXFO will charge standard calibration fees.



IMPORTANT

The warranty can become null and void if:

- **unit has been tampered with, repaired, or worked upon by unauthorized individuals or non-EXFO personnel.**
- **warranty sticker has been removed.**
- **case screws, other than those specified in this guide, have been removed.**
- **case has been opened, other than as explained in this guide.**
- **unit serial number has been altered, erased, or removed.**
- **unit has been misused, neglected, or damaged by accident.**

THIS WARRANTY IS IN LIEU OF ALL OTHER WARRANTIES EXPRESSED, IMPLIED, OR STATUTORY, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. IN NO EVENT SHALL EXFO BE LIABLE FOR SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

Liability

EXFO shall not be liable for damages resulting from the use of the product, nor shall be responsible for any failure in the performance of other items to which the product is connected or the operation of any system of which the product may be a part.

EXFO shall not be liable for damages resulting from improper usage or unauthorized modification of the product, its accompanying accessories and software.

Exclusions

EXFO reserves the right to make changes in the design or construction of any of its products at any time without incurring obligation to make any changes whatsoever on units purchased. Accessories, including but not limited to fuses, pilot lamps, batteries and universal interfaces (EUI) used with EXFO products are not covered by this warranty.

This warranty excludes failure resulting from: improper use or installation, normal wear and tear, accident, abuse, neglect, fire, water, lightning or other acts of nature, causes external to the product or other factors beyond the control of EXFO.



IMPORTANT

In the case of products equipped with optical connectors, EXFO will charge a fee for replacing connectors that were damaged due to misuse or bad cleaning.

Certification

EXFO certifies that this equipment met its published specifications at the time of shipment from the factory.

Service and Repairs

EXFO commits to providing product service and repair for five years following the date of purchase.

To send any equipment for service or repair:

1. Call one of EXFO's authorized service centers (see *EXFO Service Centers Worldwide* on page 83). Support personnel will determine if the equipment requires service, repair, or calibration.
2. If equipment must be returned to EXFO or an authorized service center, support personnel will issue a Return Merchandise Authorization (RMA) number and provide an address for return.
3. If possible, back up your data before sending the unit for repair.
4. Pack the equipment in its original shipping material. Be sure to include a statement or report fully detailing the defect and the conditions under which it was observed.
5. Return the equipment, prepaid, to the address given to you by support personnel. Be sure to write the RMA number on the shipping slip. *EXFO will refuse and return any package that does not bear an RMA number.*

Note: *A test setup fee will apply to any returned unit that, after test, is found to meet the applicable specifications.*

After repair, the equipment will be returned with a repair report. If the equipment is not under warranty, you will be invoiced for the cost appearing on this report. EXFO will pay return-to-customer shipping costs for equipment under warranty. Shipping insurance is at your expense.

Routine recalibration is not included in any of the warranty plans. Since calibrations/verifications are not covered by the basic or extended warranties, you may elect to purchase FlexCare Calibration/Verification Packages for a definite period of time. Contact an authorized service center (see *EXFO Service Centers Worldwide* on page 83).

EXFO Service Centers Worldwide

If your product requires servicing, contact your nearest authorized service center.

EXFO Headquarters Service Center
400 Godin Avenue
Quebec (Quebec) G1M 2K2
CANADA

1 866 683-0155 (USA and Canada)
Tel.: 1 418 683-5498
Fax: 1 418 683-9224
support@exfo.com

EXFO Europe Service Center
Winchester House, School Lane
Chandlers Ford, Hampshire S053 4DG
ENGLAND

Tel.: +44 2380 246800
Fax: +44 2380 246801
support.europe@exfo.com

EXFO Telecom Equipment (Shenzhen) Ltd.
3rd Floor, Building 10,
Yu Sheng Industrial Park (Gu Shu Crossing), No. 467,
National Highway 107,
Xixiang, Bao An District,
Shenzhen, China, 518126

Tel: +86 (755) 2955 3100
Fax: +86 (755) 2955 3101
support.asia@exfo.com

A Technical Specifications

IMPORTANT

The following technical specifications can change without notice. The information presented in this section is provided as a reference only. To obtain this product's most recent technical specifications, visit the EXFO Web site at *www.exfo.com*.

SPECIFICATIONS [a]

Single SLED

Parameter	FLS-2200-06 [c]	FLS-2200-02	FLS-2200-05	FLS-2200-03	FLS-2200-04
Center wavelength (nm)	980 ± 10	1300 ± 20	1485 ± 15	1550 ± 20	1610 ± 15
3 dB spectral width (nm)	≥ 20 (25 typ.)	≥ 40 (45 typ.)	≥ 50 (60 typ.)	≥ 50 (56 typ.)	≥ 50 (55 typ.)
Output power (dBm)	0	4	–3.5	5	5
Minimum spectral density (dBm/nm) [b]	–18 (970-990 nm)	–25 (1260-1360 nm)	–27 (1450-1510 nm)	–27 (1510-1590 nm)	–20 (1565-1640 nm)
Peak spectral density (dBm/nm) [b]	–13	–12	–21	–9	–10
Total power stability (dB) [d]					
15 min	± 0.01	± 0.01	± 0.01	± 0.01	± 0.01
8 hours	± 0.01	± 0.01	± 0.01	± 0.01	± 0.01
Spectral density stability (dB) [b, d, e]					
15 min	± 0.01	± 0.01	± 0.01	± 0.01	± 0.01
8 hours	± 0.015	± 0.015	± 0.015	± 0.015	± 0.015
Ripple (dB) [e, g]	0.3	0.3	0.3	0.3	0.3
Fiber type (µm)	5/125	9/125	9/125	9/125	9/125

Dual SLED

Parameter	FLS-2200-23	FLS-2200-SCL
Center wavelength (nm)	1300 ± 20/1550 ± 20	1485 ± 15/1570 ± 10
Output power (dBm) [f]	≥ 8	≥ –3.5
Minimum spectral density (dBm/nm) [b]	–28 (1260-1360, 1510-1590 nm)	–29 (1460-1625 nm)
Peak spectral density (dBm/nm) [b]	–10	–23
Total power stability (dB) [d]		
15 min	± 0.01	± 0.01
8 hours	± 0.015	± 0.015
Spectral density stability (dB) [b, d, e]		
15 min	± 0.01	± 0.01
8 hours	± 0.01	± 0.01
Ripple (dB) [e, g]	0.3	0.3
Fiber type (µm)	9/125	9/125

Notes

a. Specifications are valid at 23 °C ± 2 °C, at maximum power after warmup time (30 minutes), with isolator, for return loss of ≥ 30 dB.

b. Typical value.

c. Specifications for the 980 nm source are set without an isolator.

d. Stability is expressed as ± half the difference between the maximum and minimum values measured in the period.

e. Measured in a 0.1 nm resolution bandwidth.

f. Output power of dual SLED source is the sum of the power output of each individual SLED.

g. The ripple is specified over a spectral range limited to 3 dB below the peak power density.

GENERAL SPECIFICATIONS

Size (H x W x D)	117 mm x 222 mm x 333 mm (4 5/8 in x 8 3/4 in x 13 1/8 in)
Weight [b]	2.7 kg (5.9 lb)
Temperature	
operating	0 °C to 40 °C (32 °F to 104 °F)
storage	–40 °C to 70 °C (–40 °F to 158 °F)
Relative humidity	0 % to 80 % non-condensing

SAFETY

IEC 60825-1:A2: 2001

Class 1M LED Product

This section gives you details about the way the specifications of your Broadband Source are determined.

All measurements are made at 23 ± 1 °C with a relative humidity of 50 ± 10 %. Wavelength ranges are all according to specifications of the option under test. All measurements are made at maximum source power output, except where explicitly stated otherwise. Warmup times for all testing equipment and source under test are respected.

Element	Description	Measured with...
Minimum spectral density (dBm/nm)	Minimum optical power available per any 1-nm slice in a specified wavelength range. The spectral density is a parameter of choice for the calculation of the dynamic range of a test setup when the Broadband Source is combined with an OSA.	an OSA with a 1-nm-resolution bandwidth
Peak spectral density (dBm/nm)	Highest power density value measured over the whole power spectrum.	an OSA with a 1-nm-resolution bandwidth
3-dB spectral width (nm)	Measurement of the extent of a power spectrum corresponding to the wavelength range where the output power exceeds one half (3 dB) of the peak spectral density.	an OSA with a 1-nm-resolution bandwidth
Central wavelength (nm)	Average of the two wavelengths limiting this 3-dB bandwidth.The location of central wavelength and the 3-dB bandwidth value give you a rough image of the shape and location of the source's power spectrum. After identifying the peak power of the power spectrum, the 3-dB bandwidth is determined.	an OSA with a 1-nm-resolution bandwidth

Element	Description	Measured with...
Ripple (dB)	Defines the maximum amplitude of local variations in the spectral density of the source. These ripples are characteristic to the SLED and do not move significantly over time.	an OSA with a 0.1-nm-resolution bandwidth
Total output power (dBm)	Amount of energy measured with a power meter. It is the absolute power value measured at the end of a 3-meter patchcord connected to the source output.	a power meter set at the source's central wavelength
Degree of polarization (%)	Ratio of the polarized power over the total power. Some sources, such as DFBs, lasers and most of the SLEDs have a very high degree of polarization (> 90 %). Some others, such as white light, ASE, LEDs and some SLEDs, have a very low degree of polarization (< 10 %). Most detectors (power meters and OSAs) are sensible to polarization changes (polarization-dependent response —PDR). Selection of the emitter and detector must take these parameters into account, especially if the device under test presents a high polarization dependency.	a degree-of-polarization (DOP) meter

Element	Description	Measured with...
15-minute and 8-hour power stability (dB)	➤ The 15-minute power stability expresses the short-term power stability of the source. It corresponds to the highest power variation (Delta) measured over 15 minutes. ➤ The 8-hour power stability expresses the long-term power stability of the source. It corresponds to the highest power variation (Delta) measured over 8 hours. It gives an indication of the referencing frequency needed. Power stability is generally expressed as ± half the maximum power variation observed (Delta/2). Using a cooled Ge power meter set at the central wavelength of the source, it is possible to measure the difference between the maximum and minimum power in any consecutive 15-minute or 8-hour period within a 12-hour moving observation window. Each value, kept as a measured power, is the instantaneous value sampled at a frequency of 0.1 Hz. The source total output power is adjusted at 0 dBm when the maximum total output power of the source is greater than 0 dBm. This ensures that the saturation effects of the power meter do not affect power measurements.	a Ge power meter set at the source's central wavelength

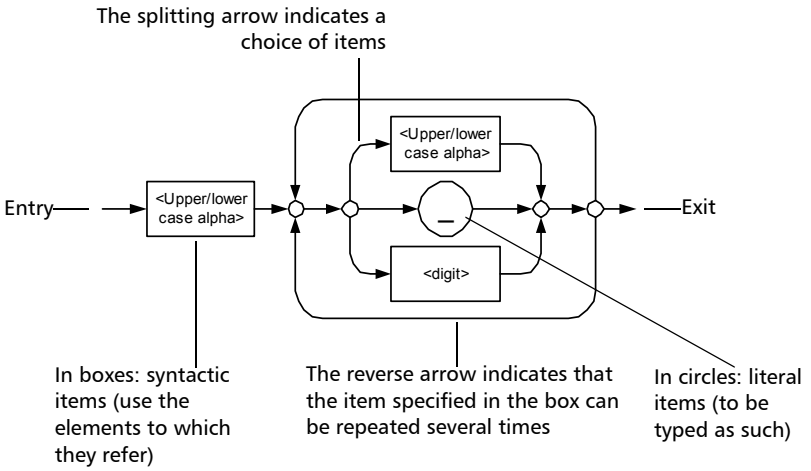
B Data Types

The following section provides an overview of the most common data types that may appear in EXFO's documentation on commands and queries. The information is supplied for guidance only.

For more detailed information, please refer to IEEE 488.2 and SCPI standards. Additional reference sources are listed in *SCPI Command Structure* on page 44.

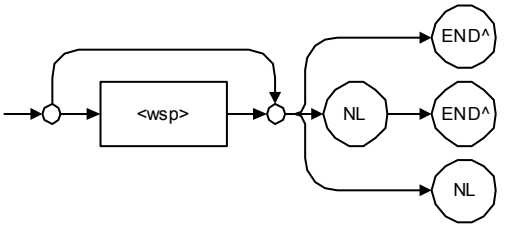
Data types are divided into two groups: <PROGRAM DATA> for the types that are used when you want to send messages to a device and <RESPONSE DATA> for the types that are used when a device sends responses to the controller.

The data types are presented in graphics often referred to as "railroad diagrams". The following example illustrates how to interpret such diagrams.



Applicable Data Types for Input—IEEE 488.2

➤ <PROGRAM MESSAGE TERMINATOR>



In the diagram above,

➤ "NL" corresponds to ASCII character code 10, in decimal (0A in binary)

➤ "END ^ " corresponds to the last data byte of the message sent with EOI = True and ATN = False

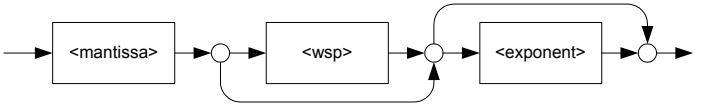
➤ <CHARACTER PROGRAM DATA>

This data type will be used to send short mnemonics when a *<DECIMAL NUMERIC PROGRAM DATA>* cannot be used.

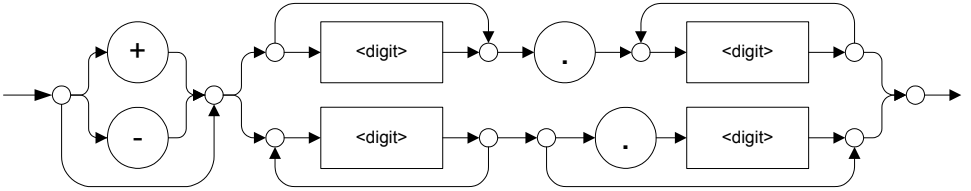
Examples: TRIANGLEWAVE, NCONTINUOUS

➤ <DECIMAL NUMERIC PROGRAM DATA> (or <NRf>)

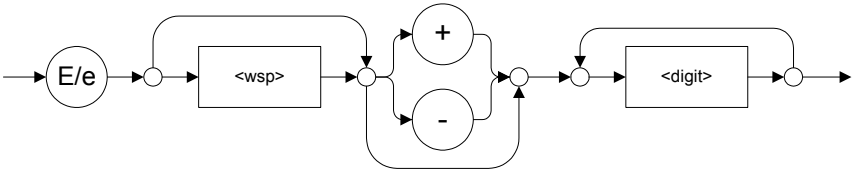
This data type includes <NR1>, <NR2> and <NR3> data types. It will be used for decimal fractions with or without an exponent. Instruments will adapt the values they receive to fit their degree of precision. For example, if an instrument has a precision of two digits after the decimal point and the incoming value is 12.048, this value will be rounded off to 12.05.



The second diagram below illustrates the <mantissa> syntax.



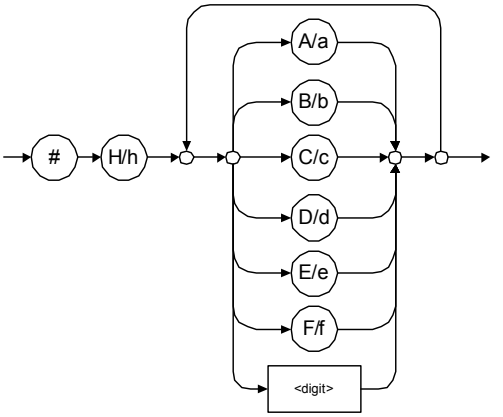
The third diagram illustrates the <exponent> syntax.



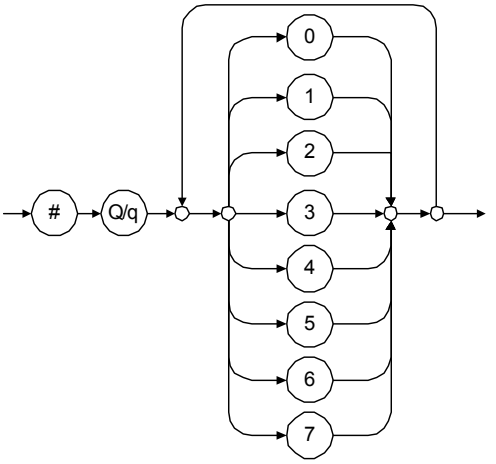
Examples: +2.0 e5, –.56E+4, 6.5e–10

➤ <NON-DECIMAL NUMERIC PROGRAM DATA>

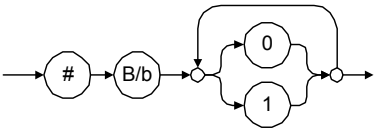
This data type will be used for integer representation in hexadecimal (base 16), octal (base 8) or binary (base 2). The numeric representations will begin with “#H” for hexadecimal, “#Q” for octal and “#B” for binary.



Examples: #Hf3bc015d, #h01a4, #hfe



Examples: #Q1234567, #q1275, #q07

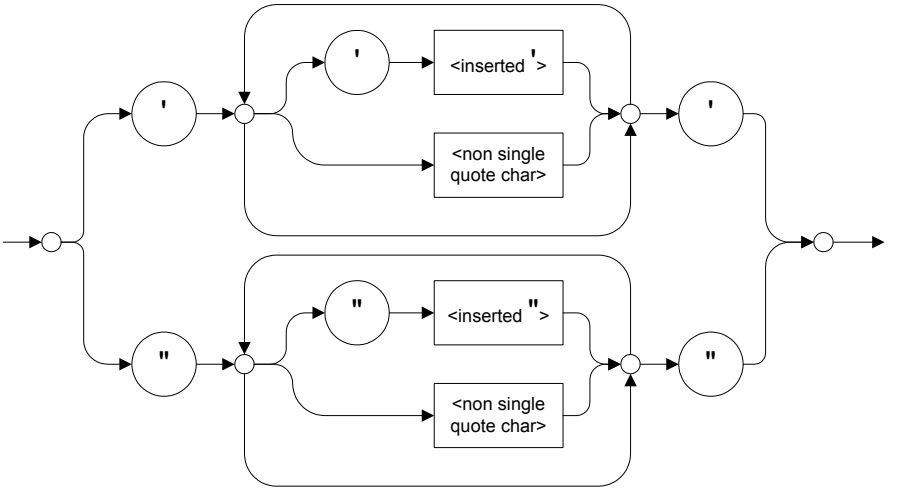


Examples: #B10010111, #b10110, #b1100

➤ <STRING PROGRAM DATA>

This data type will be used for strings containing 7-bit ASCII characters that have to be enclosed in either single- or double-quotes delimiters.

If a string needs to contain a character that is exactly the same as the delimiter, make sure to double the character to avoid syntax errors.



Examples: "SCPI Commands", 'SCPI Commands', "SCPI 'Commands'", 'SCPI "Commands"', "SCPI ""Commands""", 'SCPI ''Commands'''

➤ <ARBITRARY BLOCK PROGRAM DATA>

This data type is used to send blocks of arbitrary 8-bit information when you need to work with large amounts of data.

The actual length of the data that you send has the following structure:

➤ The first byte contains the # character.

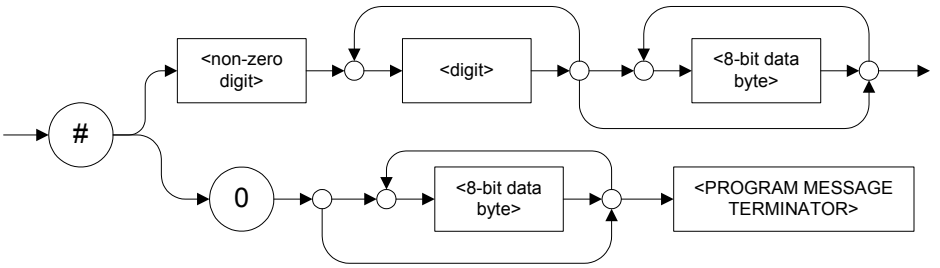
➤ The byte that immediately follows contains the number of subsequent bytes that you have to check to obtain the total length.

Note: *If you use a zero as the first digit (#0), it has to be followed by a <PROGRAM MESSAGE TERMINATOR> to allow the device to detect the end of the <ARBITRARY BLOCK PROGRAM DATA>. This will also force immediate termination of the message.*

For example, if you send the following data (here, values are expressed in decimal instead of binary for easier readability):

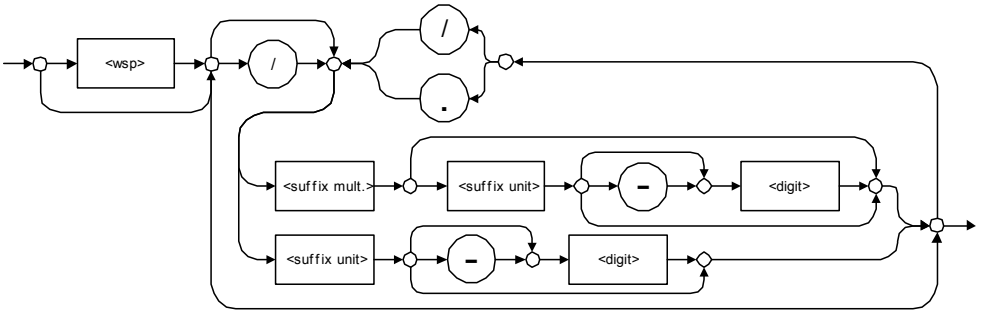
2 1 3 7 6 8 9 2 ...

The byte that immediately follows the # contains 2, which means that you would have to read the two following bytes to know the length (in bytes) of the retrieved data. The bytes indicate 1 and 3. The length will then be 13 bytes. The actual response will begin at byte number 5, in this case.



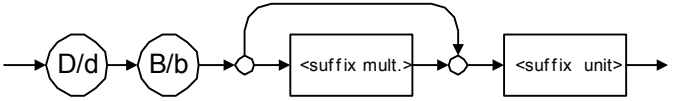
➤ <SUFFIX PROGRAM DATA>

This data type is used when units and multipliers have to be sent.



Examples: nm, kHz, km/s2, uW

A relative unit (dB) can be referenced to an absolute level, as shown on the following diagram.



Examples: db, dbm, dBW

The following table illustrates the possible forms for <suffix mult.>:

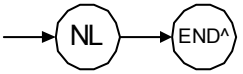
Name	Value	Mnemonic
Exa	1E18	EX
Peta	1E15	PE
Tera	1E12	T
Giga	1E9	G
Mega	1E6	MA
Kilo	1E3	K
Milli	1E-3	M
Micro	1E-6	U
Nano	1E-9	N
Pico	1E-12	P
Femto	1E-15	F
Atto	1E-18	A

The table below gives the possible forms for <suffix unit>:

Reference Unit	Suffix Unit
Degrees	DEG
Radians	RAD
Amperes	A
Volts	V
Hertz	HZ
Meters	M
Watts	W
DBs ref to 1mW	DBM
Decibels	DB
Degrees Celsius	CEL
Degrees Fahrenheit	FAR
Kelvins	K
Seconds	S
Hours	HR
Minutes	MIN

Applicable Data Types for Output—IEEE 488.2

- <RESPONSE MESSAGE TERMINATOR>



In the diagram above,

- "NL" corresponds to ASCII character code 10, in decimal (0A in binary)
- "END ^ " corresponds to the last data byte of the message sent with EOI = True and ATN = False

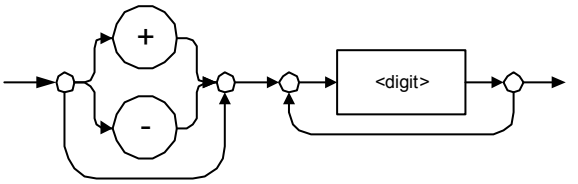
- <CHARACTER RESPONSE DATA>

This data type will be used by a device to return short mnemonics when a *<DECIMAL NUMERIC PROGRAM DATA>* cannot be used. The returned information is sent in the long form and in upper case.

Examples: TRIANGLEWAVE, NCONTINUOUS

➤ <NR1 NUMERIC RESPONSE DATA> (or <NR1>)

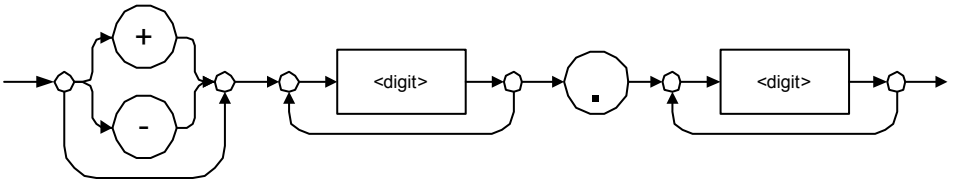
This data type will be used by a device to return positive or negative integers.



Examples: 4, –23, 90

➤ <NR2 NUMERIC RESPONSE DATA> (or <NR2>)

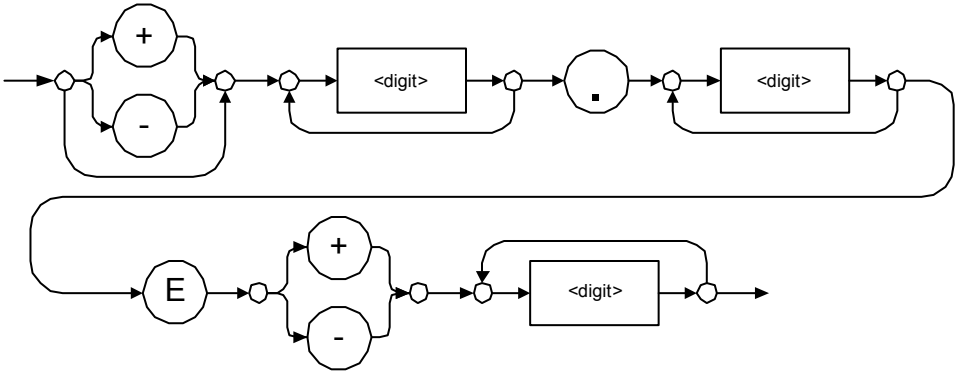
This data type will be used by a device to return positive or negative real numbers (fixed-point numbers).



Examples: 23.45, 1.22, –4.55

➤ <NR3 NUMERIC RESPONSE DATA> (or <NR3>)

This data type will be used by a device to return positive or negative exponential numbers (floating-point numbers).



Examples: 4.3E–3, –8.9456E8, 123E–5

➤ Special Numeric Values Received on Output

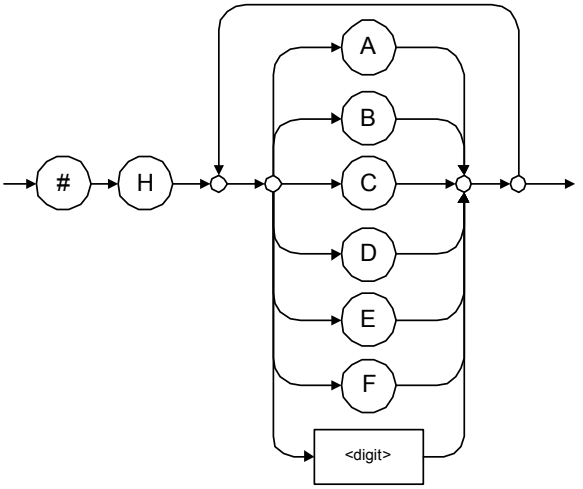
In some cases, an instrument may send values indicating that an unusual event has occurred. The following tables present the possible values.

Value is	ASCII 4 bytes	PACKED 4 bytes
Under range	2143289345.000000	7FC00001
Over range	2143289346.000000	7FC00002
Invalid	2143289347.000000	7FC00003
Inactive	2143289348.000000	7FC00004

Value is	ASCII 8 bytes	PACKED 8 bytes
Under range	9221120237577961472	7FF8000020000000
Over range	9221120238114832384	7FF8000040000000
Invalid	9221120238651703296	7FF8000060000000
Inactive	9221120239188574208	7FF8000080000000

➤ <HEXADECIMAL NUMERIC RESPONSE DATA>

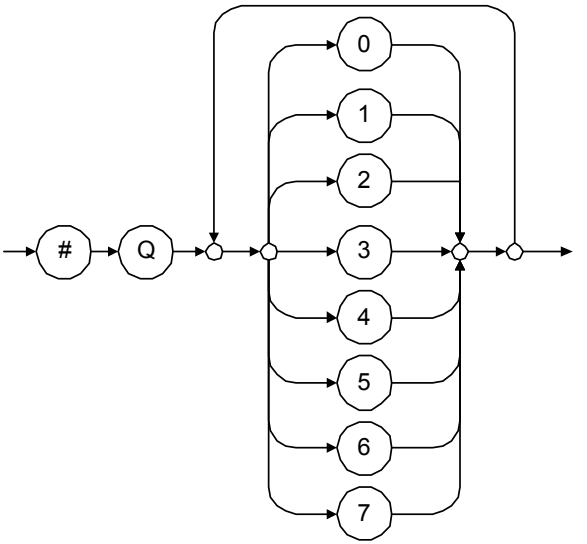
This data type will be used by a device to return integer representations in hexadecimal (base 16).



Examples: #HA3C5, #H0123C, #H010F

➤ <OCTAL NUMERIC RESPONSE DATA>

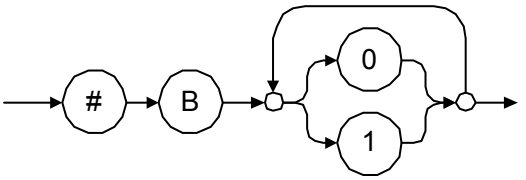
This data type will be used by a device to return integer representations in octal (base 8).



Examples: #Q753214, #Q0124, #Q0725

➤ <BINARY NUMERIC RESPONSE DATA>

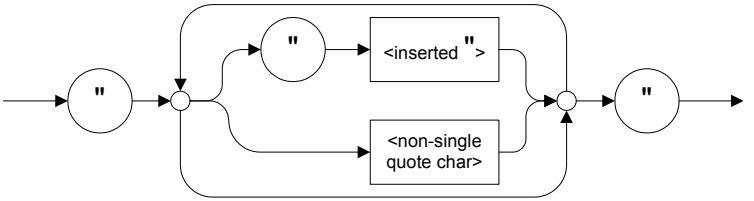
This data type will be used by a device to return integer representations in binary (base 2).



Examples: #B11011110101, #B110100, #B0100

➤ <STRING RESPONSE DATA>

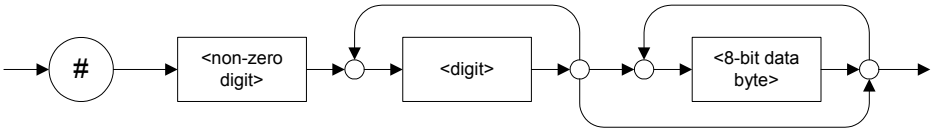
This data type will be used by a device to return strings containing 7-bit ASCII characters and especially when text has to be displayed since even the non-printable characters are also returned.



Examples: "SCPI Commands", "SCPI ""Commands"""

➤ <DEFINITE LENGTH ARBITRARY BLOCK RESPONSE DATA>

This data type is used by a device to return blocks of 8-bit binary information with a fixed and predetermined length.



The actual length of the retrieved data has the following structure:

➤ The first byte contains the # character.

➤ The byte that immediately follows contains the number of subsequent bytes that you have to check to know the total length.

For example, if you receive this response (here, values are expressed in decimal instead of binary for easier readability):

2 1 3 7 6 8 9 2 ...

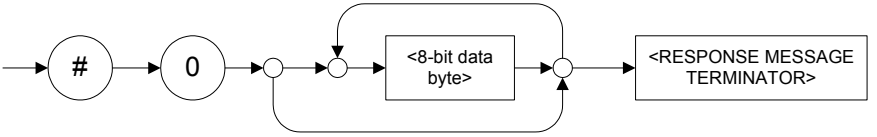
The byte that immediately follows the # contains 2, which means that you have to read the two following bytes to know the length (in bytes) of the retrieved data. The bytes indicate 1 and 3. The length will then be 13 bytes. The actual response will begin at byte number 5, in this case.

Examples: #14<DAB><DAB><DAB><DAB>,
#3004<DAB><DAB><DAB><DAB>

where "<DAB>" stands for data byte

➤ <INDEFINITE LENGTH ARBITRARY BLOCK RESPONSE DATA>

This data type is used by a device to return blocks of 8-bit binary information when the block length was not predefined or when data has to be computed later.

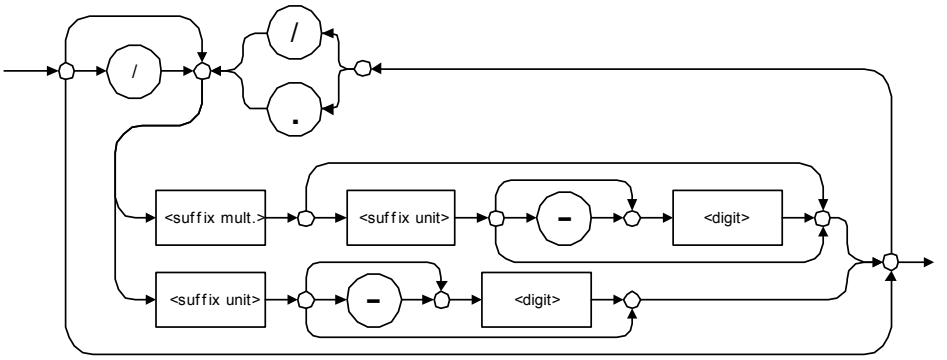


Note: *If you receive a zero as the first digit (#0), it is necessarily followed by a <RESPONSE PROGRAM MESSAGE TERMINATOR> to allow you to detect the end of the <INDEFINITE LENGTH ARBITRARY BLOCK RESPONSE DATA>.*

Example: #0<DAB><DAB><DAB><DAB><terminator> where "<DAB>" stands for data byte.

➤ <SUFFIX RESPONSE DATA>

This data type is used by a device to return units and multipliers.

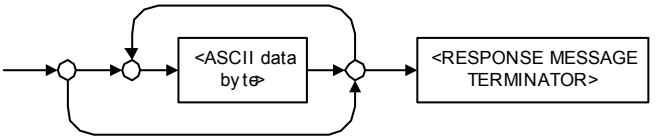


Examples: DBW, W, KHZ

➤ <ARBITRARY ASCII RESPONSE DATA>

This data type is used by a device to return information when it is impossible to use any other data type.

Example: To the *IDN? query, the device will return this response in an arbitrary ASCII bytes format:
EXFO E.O. Engineering,IQS-510P,125-2A55,1.0.1.97



Applicable Data Types for Input—SCPI

SCPI data types include the IEEE 488.2 data types (see *Applicable Data Types for Input—IEEE 488.2* on page 90) with certain additional restrictions.

- <numeric_value>: abbreviated form of the decimal numeric element. It differs from the <DECIMAL NUMERIC PROGRAM DATA> "<NRf>" described in IEEE 488.2.

  Several forms of <CHARACTER PROGRAM DATA> are defined as special forms of numbers. These are: MINimum, MAXimum, DEFault, UP, DOWN, Not A Number (NAN), INFinity and Negative INFinity (NINF). The following special forms are likely to be used by EXFO's instruments in certain commands or queries:

  - DEFault: This special <numeric_value> parameter forces the instrument to select a value, which is deemed to be convenient to you.
  - MINimum|MAXimum: These special <numeric_value> parameters refer to the instrument's limit values. MINimum corresponds to the value closest to negative infinity that the function can accept. MAXimum corresponds to the largest value that the function can accept.
  - <Boolean Program Data>: This form is often used as a shorthand of the <DECIMAL NUMERIC PROGRAM DATA>ON|OFF form.

    <Boolean Program Data> parameters have a value of 0 or 1 and are not followed by any unit.

    On input, an <NRf> is rounded to an integer.

    A non-zero result is interpreted as 1.

    ON and OFF are accepted on input for readability purposes. They correspond respectively to 1 and 0. However, on output, they appear as 1 or 0, never ON or OFF.

Special Numeric Values Received on Output

It is possible that an instrument returns unusual values in certain cases. For information on these values, see Applicable Data Types for Output—IEEE 488.2 *on page 99*.

C IEEE 488.2 and Specific Command Reference

This chapter presents detailed information about the commands and queries supplied with your FLS-2200 Broadband Source.

IEEE 488.2 Commands—Quick Reference

The Broadband Source recognizes the required commands identified in IEEE 488.2. The table below summarizes these commands.

Command	Function
*CLS	Clear status command
*ESE	Standard event status enable command
*ESE?	Standard event status enable query
*ESR?	Standard event status register query
*IDN?	Identification query
*LOK[a]	Set Remote Lockout programming state
*LOK?[a]	Remote Lockout programming state query
*OPC	Operation complete command
*OPC?	Operation complete query
*REM[a]	Set Remote programming state
*RST	Reset command
*SRE	Service request enable command
*SRE?	Service request enable query
*STB?	Read status byte query
*TST?	Self-test query
*WAI	Wait for pending operations to be completed

a. This command can only be used with RS-232 communication.

IEEE 488.2 Commands—Description

*CLS

Description The *CLS command clears the Standard Event Status Register and the Error/Event Queue.

Syntax *CLS

Parameter(s) None

*ESE

Description The *ESE command sets the Standard Event Status Enable Register bits, as defined in the table below. This register contains a mask value for the bits to be enabled in the Standard Event Status Register.

MSB Standard Event Status Enable Register LSB

PON	N.U	CME	EXE	DDE	QYE	N.U.	OPC

Syntax *ESE<wsp><RegisterValue>

Parameter(s) *RegisterValue:*

The program data syntax for <RegisterValue> is defined as a <DECIMAL NUMERIC PROGRAM DATA> element.

The <RegisterValue>, expressed in base 2, represents the bit values of the Standard Event Status Enable Register.

*ESE

The table below shows the contents of this register.

Bit	Weight	Meaning
PON	128	Power ON Enable
N.U.	64	Not used
CMD	32	CoMmanD Error Enable
EXE	16	Execution Error Enable
DDE	8	Device Dependent Error Enable
QRY	4	QueRry Error Enable
N.U.	2	Not used
OPC	1	Operation Complete Enable

A value of 1 in the Enable Register enables the corresponding bit in the Status Register, a value of 0 disables the bit. The value of the <RegisterValue> shall be in the range of 0 through 255.

Example(s)

*ESE 25
where 25 = (bit EXE, bit DDE and bit OPC)

*ESE 0
clears the content of the Standard Event Status Enable register

See Also

*ESE?
*ESR?

*ESE?

Description

The *ESE? query allows the programmer to determine the current contents of the Standard Event Status Enable Register. See the contents of this register below.

MSB Standard Event Status Enable Register LSB

PON	N.U	CME	EXE	DDE	QYE	N.U.	OPC

Syntax

*ESE?

Parameter(s)

None

Response Syntax

<RegisterValue>

*ESE?

Response(s)

RegisterValue:

The response data syntax for <RegisterValue> is defined as a <NR1 NUMERIC RESPONSE DATA> element.

The <RegisterValue> ranges from 0 through 255.

The <RegisterValue> value expressed in base 2 (binary) represents the bit values of the Standard Event Status Enable register. See below.

Bit	Weight	Meaning
PON	128	Power ON Enable
N.U.	64	Not used
CMD	32	CoMmanD Error Enable
EXE	16	Execution Error Enable
DDE	8	Device Dependent Error Enable
QRY	4	QueRry Error Enable
N.U.	2	Not used
OPC	1	Operation Complete Enable

Example(s)

*ESE? returns 133
where 133 = (bit PON, bit QYE and bit OPC)

See Also

*ESE
*ESR?

*ESR?

Description

The *ESR? query allows the programmer to determine the current contents of the Standard Event Status Register. Reading the Standard Event Status Register clears it. See the contents of this register below.

MSB Standard Event Status Enable Register LSB

PON	N.U	CME	EXE	DDE	QYE	N.U.	OPC

Syntax

*ESR?

Parameter(s)

None

Response Syntax

<RegisterValue>

*ESR?

Response(s)

RegisterValue:

The response data syntax for <RegisterValue> is defined as a <NR1 NUMERIC RESPONSE DATA> element.

The <RegisterValue> ranges from 0 through 255.

The <RegisterValue> value expressed in base 2 (binary) represents the bit values of the Standard Event Status register. See below.

Bit	Weight	Meaning
PON	128	Power ON Enable
N.U.	64	Not used
CMD	32	CoMmanD Error Enable
EXE	16	Execution Error Enable
DDE	8	Device Dependent Error Enable
QRY	4	QueRry Error Enable
N.U.	2	Not used
OPC	1	Operation Complete Enable

Example(s)

*ESR? returns 33
where 33 = (bit CME and bit OPC)

See Also

*ESE
*ESE?

*IDN?

Description	The intent of the *IDN? query is for the unique identification of devices over the system interface.
Syntax	*IDN?
Parameter(s)	None
Response Syntax	<Identification>
Response(s)	*Identification:* The response data syntax for <Identification> is defined as an <ARBITRARY ASCII RESPONSE DATA> element. The response syntax for the *IDN? query, <Identification> is defined as an <ARBITRARY ASCII RESPONSE DATA> element. This implies that the *IDN? query should be the last <QUERY MESSAGE UNIT> in a <TERMINATED PROGRAM MESSAGE>. The response is organized into four fields separated by commas. The field definitions are as follows: Field 1 (Manufacturer): EXFO E. O. Engineering Field 2 (Model): Instrument Model

*IDN?

	Field 3 (Serial number): ASCII character (0 if not available) Field 4 (Firmware level): ASCII character (0 if not available) ASCII character 0 represents a single ASCII-encoded byte with a value of 30 (48 decimal). The presence of data in all fields is mandatory. If either field 3 or 4 is not available, the ASCII character 0 shall be returned for that field. A field may contain any 7-bit ASCII-encoded bytes in the range of 20 through 7E (32 through 126 decimal) except commas (2C, 44 decimal) and semicolons (3B, 59 decimal).
Example(s)	*IDN? returns EXFO E.O. Engineering,FLS-2200,243478,1.0.1.0
Notes	The overall length of the *IDN? response is less than or equal to 72 characters.

	*LOK
Description	This command is used to set the Remote Lockout programming state.
Syntax	*LOK<wsp><LockoutState>
Parameter(s)	*LockoutState:* The program data syntax for <LockoutState> is defined as a <Boolean Program Data> element. The <LockoutState>special forms ON and OFF are accepted on input for increased readibility. ON corresponds to 1 and OFF to 0. The <LockoutState> parameter is the new lockout state of the Broadband Source: “0”- Removing the Lockout state of the source. “1”- Enabling the Lockout state of the source.
Example(s)	*LOK 1
Notes	This command can only be used when working with RS-232 communication.
See Also	*LOK?

*LOK?

Description	This query returns the Remote Lockout programming state indicating if the Broadband Source has been locked out by a remote application.
Syntax	*LOK?
Parameter(s)	None
Response Syntax	<LockoutState>
Response	*LockoutState:* The response data syntax for <LockoutState> is defined as an <NR1 NUMERIC RESPONSE DATA> element. The <LockoutState> response corresponds to the remote lockout state of the Broadband Source: "0"- The source is unlocked. "1"- The source is locked.
Example(s)	*LOK? returns 1
Notes	This command can only be used when working with RS-232 communication.
See Also	*LOK

	*OPC
Description	The *OPC command allows synchronization between the instrument and an external controller. The *OPC command causes the instrument to set bit 0 (Operation Complete) in the Standard Event Status Register to the TRUE (logic 1) state when the instrument completes all pending operations. Detection of the Operation Complete message can be accomplished by continuous polling of the Standard Event Status Register using the *ESR? common query command. However, using a service request eliminates the need to poll the Standard Event Status Register thereby freeing the controller to do other useful work.
Syntax	*OPC
Parameter(s)	None
See Also	*OPC? *WAI

*OPC?

Description	The *OPC? query allows synchronization between the instrument and an external controller by reading the Output Queue or by waiting for a service request on the Message Available (MAV) bit in the Status Byte Register. The *OPC? query causes the instrument to place an ASCII character, 1, into its Output Queue when the device completes all pending operations. A consequence of this action is that the MAV bit in the Status Byte Register is set to state 1.
Syntax	*OPC?
Parameter(s)	None
Response Syntax	<Acknowledge>
Response(s)	*Acknowledge:* The response data syntax for <Acknowledge> is defined as a <NR1 NUMERIC RESPONSE DATA> element. The <Acknowledge> response is a single ASCII-encoded byte corresponding to 1. The receipt of an <Acknowledge> response indicates that all pending selected device operations have been completed.
Example(s)	*OPC? Return 1
See Also	*OPC *WAI

	*REM
Description	This command is used to set the Remote programming state that determines if the source will be controlled locally or remotely.
Syntax	*REM<wsp><RemoteState>
Parameter(s)	*RemoteState:* The program syntax data for <RemoteState> is defined as a <Boolean Program Data> element. The <RemoteState> special forms ON and OFF are accepted on input for increased readibility. ON corresponds to 1 and OFF to 0. The <RemoteState> parameter is the new remote state of the Broadband Source: “0”- to set Local state. “1”- to set Remote state.
Example(s)	*REM 1
Notes	This command can only be used with RS-232 communication.

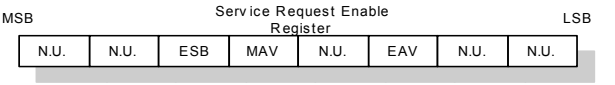
*RST

Description	The *RST command performs a device reset. This command is the third reset level in a three-level reset strategy. The Reset command shall do the following: a) Sets the device-specific functions to a known state that is independent of the past-use history of the device. b) Forces the device into OCIS state (Operation complete Command Idle State). c) Forces the device into OQIS state (Operation complete Query Idle State). The Reset command explicitly DOES NOT affect the following: a) The state of the Communication interface. b) The Output Queue. c) Any Event Enable Register setting, including the Standard Event Status Enable Register setting. d) Any Event Register setting, including the Standard Event Status Register settings. e) Calibration data that affects device specifications. f) The Service Request Enable Register setting.
Syntax	*RST
Parameter(s)	None

*SRE

Description

The *SRE command sets the Service Request Enable Register bits. See the contents of this register below. This register contains a mask value to enable the bits in the Status Byte Register.



Syntax

*SRE<wsp><RegisterValue>

Parameter(s)

RegisterValue:

The program data syntax for <RegisterValue> is defined as a <DECIMAL NUMERIC PROGRAM DATA> element.

The <RegisterValue> value ranges from 0 through 255.

The <RegisterValue>, expressed in base 2 (binary), represents the bit values of the Service Request Enable Register.

*SRE

See the contents of this register below.

Bit	Weight	Meaning
N.U.	128	Not used
N.U.	64	Not used
ESB	32	Event Summary Bit Enable
MAV	16	Message AVailable Enable
N.U.	8	Not used
EAV	4	Error / Event AVailable Enable
N.U.	2	Not used
N.U.	1	Not used

A bit value of zero shall indicate a disabled condition.

Example(s)

*SRE 52
where 52 = (bit ESB, bit MAV and bit EAV)

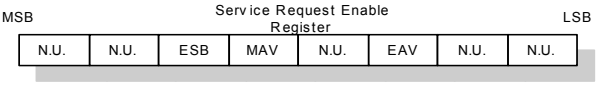
See Also

*SRE?
*STB?

*SRE?

Description

The *SRE? query allows the programmer to determine the current contents of the Service Request Enable Register. See the contents of this register below.



Bit	Weight	Meaning
N.U.	128	Not used
N.U.	64	Not used
ESB	32	Event Summary Bit Enable
MAV	16	Message AVailable Enable
N.U.	8	Not used
EAV	4	Error / Event AVailable Enable
N.U.	2	Not used
N.U.	1	Not used

Syntax *SRE?

Parameter(s) None

Response Syntax <RegisterValue>

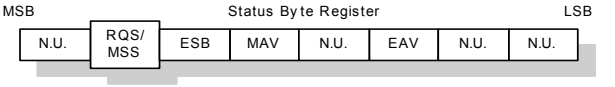
*SRE?

Response(s)	*RegisterValue:* The response data syntax for <RegisterValue> is defined as a <NR1 NUMERIC RESPONSE DATA> element. The <RegisterValue> ranges from 0 through 255. When converted to binary (base 2), the <RegisterValue> represents the current bit values of the Service Request Enable Register.
Example(s)	*SRE Return 32 (bit ESB)
See Also	*SRE *STB?

*STB?

Description

The *STB? query allows the programmer to read the status byte and Master Summary Status bit. See the content of this register below.



Syntax

*STB?

Parameter(s)

None

Response Syntax

<RegisterValue>

*STB?

Response(s)

RegisterValue:

The response data syntax for <RegisterValue> is defined as a <NR1 NUMERIC RESPONSE DATA> element.

The <RegisterValue> ranges from 0 through 255.

The <RegisterValue> value, expressed in base 2 (binary) represents the bit values of the Status Byte Register. See the contents of this register below.

Bit	Weight	Meaning
N.U.	128	Not used
RQS/ MSS	64	ReQuest Service (read by serial polling)/MaSter Summary bit (read by *STB?)
ESB	32	Event Summary Bit Enable
MAV	16	Message AVailable Enable
N.U.	8	Not used
EAV	4	Error / Event AVailable Enable
N.U.	2	Not used
N.U.	1	Not used

Example(s)

*STB? Return 68
where 68 = (bit MSS and bit EAV)

See Also

*SRE
*SRE?

	*TST?
Description	The *TST? query causes an internal self-test and places a response into the Output Queue indicating whether or not the device completed the self-test without any detected errors. Upon successful completion of *TST?, the device settings is restored to their values prior to the *TST?.
Syntax	*TST?
Parameter(s)	None
Response Syntax	<Result>
Response(s)	*Result:* The response data syntax for <Result> is defined as a <NR1 NUMERIC RESPONSE DATA> element.

*TST?

	<Result> corresponds to a decimal value indicating the sum of all corresponding errors: "0"–No errors "1"–N/A "2"–Supply voltage error: "4"–SLED over-current: "8"–N/A "16"–SLED temperature problem: "32"–Thermo-electric cooler over-current: "64"–N/A "128"–EEPROM access error: "256"–EEPROM checksum error: "512"–EEPROM error: "1024"–FPGA problem: "2048"–Temperature error: "4096"–N/A "8192"–N/A
Example(s)	*TST? Return 0 (self-test was completed with success)

	*WAI
Description	The *WAI command shall prevent the device from executing any further commands or queries until the no-operation-pending flag becomes TRUE.
Syntax	*WAI
Parameter(s)	None
Example(s)	*WAI
See Also	*OPC *OPC?

Product-Specific Commands—Quick Reference

The table below summarizes commands specific to the Broadband Source.

Command						Parameter(s)	P.
DISPlay	BRIGhtness					<Brightness>|MAXimum|MINimum|DEFault	136
	BRIGhtness?					[MAXimum|MINimum|DEFault]	138
SOURce[1..n]	CURRent	[LEVel]	[IMMediate]	[AMPLitude]		<Current[<wsp>A]>|MAXimum|MINimum	140
				[AMPLitude]?		[MAXimum|MINimum|DEFault]	142
	POWer	STATe				<PowerState>	144
		STATe?					145
	WAVelength	[CW]?					146
		COUNt?					147
SYSTem	ERRor	[NEXT]?					148
	VERSion?						150

Product-Specific Commands—Description

:DISPlay:BRIGhtness	
Description	This command allows you to control the intensity of the display. *At RST, the display brightness is at maximum.
Syntax	:DISPlay:BRIGhtness<wsp><Brightness>|MAXimum|MINimum|DEFault
Parameter(s)	*Brightness:* The program data syntax for <Brightness> is defined as a <numeric_value> element. The <Brightness> special forms MINimum, MAXimum and DEFault are accepted on input. MINimum allows to set the instrument to the smallest supported value. MAXimum allows to set the instrument to the greatest supported value. DEFault allows the instrument to select a value for the <Brightness> parameter.

:DISPlay:BRIGhtness

	The <Brightness> parameter corresponds to the intensity of the display unit. 1, corresponds to full intensity 0, corresponds to the lowest intensity Any other value will be rounded to the nearest value (0 or 1). For example, if the value is 0.2, it will be rounded to 0. If the value is 0.8, it will be rounded to 1.
Example(s)	DISP:BRIG 1 DISPLAY:BRIGHTNESS 0
See Also	DISPlay:BRIGhtness?

:DISPlay:BRIGhtness?

Description

This query returns the intensity of the display.

*RST does not affect this command.

Syntax

:DISPlay:BRIGhtness?[<wsp>MAXimum|MINimum|DEFault]

Parameter(s)

Parameter 1:

The program data syntax for the first parameter is defined as a <CHARACTER PROGRAM DATA> element. The allowed <CHARACTER PROGRAM DATA> elements for this parameter are: MAXimum|MINimum|DEFault.

MINimum is used to retrieve the instrument's smallest supported value.
MAXimum is used to retrieve the instrument's greatest supported value.
DEFault is used to retrieve the instrument's default value.

Response Syntax

<Brightness>

Response(s)

Brightness:

The response data syntax for <Brightness> is defined as a <NR1 NUMERIC RESPONSE DATA> element.

The <Brightness> response corresponds to the intensity of the display unit.

:DISPlay:BRIGhtness?

	1, corresponds to full intensity 0, corresponds to the lowest intensity
Example(s)	DISP:BRIG? Return 1
See Also	DISPlay:BRIGhtness

:SOURce[1..n]:CURRent[:LEVel][:IMMediate][:AMPLitude]

Description

This command sets the source's current level. This value can be set even if the source is off.

At *RST, the current level is device-dependent.

Syntax

:SOURce[1..n]:CURRent[:LEVel][:IMMediate][:AMPLitude]<wsp><Current[<wsp>A]>|MAXimum|MINimum

Parameter(s)

Current:

The program data syntax for <Current> is defined as a <numeric_value> element followed by an optional <SUFFIX PROGRAM DATA> element. The allowed <SUFFIX PROGRAM DATA> element is A. The <Current> special forms MINimum and MAXimum are accepted on input.

MINimum allows to set the instrument to the smallest supported value.
MAXimum allows to set the instrument to the greatest supported value.

The <Current> parameter is the new source current drive, in amperes (A).

:SOURce[1..n]:CURRent[:LEVel][:IMMediate][:AMPLitude]

Example(s)	SOUR:CURR 100 mA SOUR:POW:STAT ON
Notes	You can specify the desired SLED. In SOURce[1..n], n corresponds to the total number of SLEDs your unit contains.
See Also	SOURce[1..n]:CURRent[:LEVel][:IMMediate][:AMPLitude]?

:SOURce[1..n]:CURRent[:LEVel][:IMMediate][:AMPLitude]?

Description

This query returns the source's current level.

At *RST, the current level is device-dependent.

Syntax

:SOURce[1..n]:CURRent[:LEVel][:IMMediate][:AMPLitude]?[<wsp>MAXimum|MINimum|DEFault]

Parameter(s)

Parameter 1:

The program data syntax for the first parameter is defined as a <CHARACTER PROGRAM DATA> element. The allowed <CHARACTER PROGRAM DATA> elements for this parameter are: MAXimum|MINimum|DEFault.

MINimum is used to retrieve the instrument's smallest supported value.
MAXimum is used to retrieve the instrument's greatest supported value.
DEFault is used to retrieve the instrument's default value.

Response Syntax

<Current>

Response(s)

Current:

The response data syntax for <Current> is defined as a <NR3 NUMERIC RESPONSE DATA> element.

The <Current> response corresponds to the current source current drive.

:SOURce[1..n]:CURRent[:LEVel][:IMMediate][:AMPLitude]?

Example(s)	SOUR:CURR:LEV:IMM:AMPL? MIN Return 5.000000E-2 SOUR:CURR:LEV:IMM:AMPL? Return 1.000000E-1
Notes	You can specify the desired SLED. In SOURce[1..n], n corresponds to the total number of SLEDs your unit contains.
See Also	SOURce[1..n]:CURRent[:LEVel][:IMMediate][:AMPLitude]

:SOURce[1..n]:POWer:STATe

Description	This command turns the optical source on or off. When source is on, the red LED (Active) on the instrument's front panel lights up. *RST sets the optical source to OFF.
Syntax	:SOURce[1..n]:POWer:STATe<wsp><PowerState>
Parameter(s)	*PowerState:* The program data syntax for <PowerState> is defined as a <Boolean Program Data> element. The <PowerState> special forms ON and OFF are accepted on input for increased readability. ON corresponds to 1 and OFF corresponds to 0. The <PowerState> parameter corresponds to the sources new power state. 1 or ON, turns the optical source on. 0 or OFF, turns the optical source off.
Example(s)	SOUR:POW:STAT ON
Notes	You can specify the desired SLED. In SOURce[1..n], n corresponds to the total number of SLEDs your unit contains.
See Also	SOURce[1..n]:POWer:STATe?

:SOURce[1..n]:POWer:STATe?

Description	This query returns a value indicating the state of the optical source (on or off). *RST sets the optical source to OFF.
Syntax	:SOURce[1..n]:POWer:STATe?
Parameter(s)	None
Response Syntax	<PowerState>
Response(s)	*PowerState:* The response data syntax for <PowerState> is defined as a <NR1 NUMERIC RESPONSE DATA> element. The <PowerState> response corresponds to the state of the source power. 0- The optical source is off. 1- The optical source is on.
Example(s)	SOUR:POW:STAT OFF SOUR:POW:STAT?
Notes	You can specify the desired SLED. In SOURce[1..n], n corresponds to the total number of SLEDs your unit contains.
See Also	SOURce[1..n]:POWer:STATe

:SOURce[1..n]:WAVelength[:CW]?

Description	This query returns the source's current wavelength. *RST does not affect this command.
Syntax	:SOURce[1..n]:WAVelength[:CW]?
Parameter(s)	None
Response Syntax	<Wavelength>
Response(s)	*Wavelength:* The response data syntax for <Wavelength> is defined as a <NR3 NUMERIC RESPONSE DATA> element. The <Wavelength> response corresponds to the wavelength of the currently selected source.
Example(s)	SOUR:WAV:COUN? SOUR:WAV?
Notes	You can specify the desired SLED. In SOURce [1..n], n corresponds to the total number of SLEDs your unit contains.
See Also	SOURce[1..n]:WAVelength:COUNt?

:SOURce[1..n]:WAVelength:COUNt?

Description	This query returns the number of sources in the unit. *RST does not affect this command.
Syntax	:SOURce[1..n]:WAVelength:COUNt?
Parameter(s)	None
Response Syntax	<WavelengthCount>
Response(s)	*WavelengthCount:* The response data syntax for <WavelengthCount> is defined as a <NR3 NUMERIC RESPONSE DATA> element. The <WavelengthCount> response corresponds to the number of sources in the unit.
Example(s)	SOUR:WAV? SOUR:WAV:COUN?
Notes	You can specify the desired SLED. In SOURce [1..n], n corresponds to the total number of SLEDs your unit contains.
See Also	SOURce[1..n]:WAVelength[:CW]?

:SYSTem:ERRor[:NEXT]?

Description

The SYSTem:ERRor[:NEXT]? queries the error/event queue for the next item and removes it from the queue. The response message consists of two fields separated by commas <Code>,<Description[,Info]>.

SYSTem:ERRor[:NEXT]? is a query only and, therefore, does not have an associated *RST state.

Syntax

:SYSTem:ERRor[:NEXT]?

Parameter(s)

None

Response Syntax

<Code>,<Description[,Info]>

Response(s)

➤ *Code:*

The response data syntax for <Code> is defined as a <NR1 NUMERIC RESPONSE DATA> element.

The <Code> is a unique integer in the range [-32768, 32767]. All positive numbers are instrument-dependent. All negative numbers are reserved by the SCPI standard with certain standard error/event codes described in an appendix of this document. The zero value is also used to indicate that no error or event has occurred.

➤ *Description[,Info]:*

The response data syntax for <Description[,Info]> is defined as a <STRING RESPONSE DATA> element.

:SYSTem:ERRor[:NEXT]?

The <Description[,Info]> parameter of the full response is a quoted string containing a description followed by information text [,Info]. Each <Code> has a unique and fixed <Description> associated with it. The <Date> and <Time> are appended to the [,info] separated by a semi-colon using the following format:
<Date><wsp><Time> where
<Date> = Year/Month/Day
<Time> = Hour,Minute,Second (24 hour time)

The maximum length of <Description[,Info]> is 255 characters. For standard defined error/event <Codes>, the <Description> is sent exactly as indicated in the appendix of this document.

Example(s)

SYST:ERR:NEXT? returns -222,"Data out of range"
SYST:ERR:NEXT? returns -222,"Data out of range,instrument monomodule 2100, 2001/11/29 14:56:16.259"

:SYSTem:VERSion?

Description	The SYSTem:VERSion? query returns a value corresponding to the SCPI version number to which the device complies. The SYSTem:VERSion? is a query only and, therefore, does not have an associated *RST state.
Syntax	:SYSTem:VERSion?
Parameter(s)	None
Response Syntax	<Version>
Response(s)	*Version:* The response data syntax for <Version> is defined as a <NR2 NUMERIC RESPONSE DATA> element. The <Version> is shown in the form Year.Revision, where Year represents the year-version (i.e. 1990) and Revision represents an approved revision number for that year. If no approved revisions are claimed, then this extension is 0.
Example(s)	SYSTem:VERSion? returns 1999.0 (no approved revisions are claimed)

D SCPI-Based Errors

Error Number	Description	Probable Cause
–100	“Command error”	This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that a Command Error as defined in IEEE 488.2, 11.5.1.1.4 has occurred.
–101	“Invalid character”	A syntactic element contains a character which is invalid for that type; for example, a header containing an ampersand, SETUP&. This error might be used in place of errors –114, –121, –141, and perhaps some others.
–102	“Syntax error”	An unrecognized command or data type was encountered; for example, a string was received when the device does not accept strings.
–103	“Invalid separator”	The parser was expecting a separator and encountered an illegal character; for example, the semicolon was omitted after a program message unit, *EMC 1:CH1:VOLTS 5.
–104	“Data type error”	The parser recognized a data element different than one allowed; for example, numeric or string data was expected but block data was encountered.
–105	“GET not allowed”	A Group Execute Trigger was received within a program message (see IEEE 488.2, 7.7).
–108	“Parameter not allowed”	More parameters were received than expected for the header; for example, the *EMC common command only accepts one parameter, so receiving *EMC 0,1 is not allowed.
–109	“Missing parameter”	Fewer parameters were received than required for the header; for example, the *EMC common command requires one parameter, so receiving *EMC is not allowed.

Error Number	Description	Probable Cause
–110	“Command header error”	An error was detected in the header. This error message should be used when the device cannot detect the more specific errors described for errors –111 through –119.
–111	“Header separator error”	A character which is not a legal header separator was encountered while parsing the header; for example, no white space followed the header, thus *GMC"MACRO" is an error.
–112	“Program mnemonic too long”	The header contains more that twelve characters (see IEEE 488.2, 7.6.1.4.1).
–113	“Undefined header”	The header is syntactically correct, but it is undefined for this specific device; for example, *XYZ is not defined for any device.
–114	“Header suffix out of range”	The value of a numeric suffix attached to a program mnemonic makes the header invalid.
–115	“Unexpected number of parameters”	The number of parameters received does not correspond to the number of parameters expected. This is typically due to an inconsistency with the number of instruments in the selected group (see section on INSTrument:DEFine:GROup).
–120	“Numeric data error”	This error, as well as errors –121 through –129, are generated when parsing a data element which appears to be numeric, including the non-decimal numeric types. This particular error message should be used if the device cannot detect a more specific error.
–121	“Invalid character in number”	An invalid character for the data type being parsed was encountered; for example, an alpha in a decimal numeric or a “9” in octal data.

Error Number	Description	Probable Cause
–123	“Exponent too large”	The magnitude of the exponent was larger than 32000 (see IEEE 488.2, 7.7.2.4.1).
–124	“Too many digits”	The mantissa of a decimal numeric data element contained more than 255 digits, excluding leading zeros (see IEEE 488.2, 7.7.2.4.1).
–128	“Numeric data not allowed”	A legal numeric data element was received, but the device does not accept one in this position for the header.
–130	“Suffix error”	This error, as well as errors –131 through –139, are generated when parsing a suffix. This particular error message should be used if the device cannot detect a more specific error.
–131	“Invalid suffix”	The suffix does not follow the syntax described in IEEE 488.2, 7.7.3.2, or the suffix is inappropriate for this device.
–134	“Suffix too long”	The suffix contained more than 12 characters (see IEEE 488.2, 7.7.3.4).
–138	“Suffix not allowed”	A suffix was encountered after a numeric element which does not allow suffixes.
–140	“Character data error”	This error, as well as errors –141 through –149, are generated when parsing a character data element. This particular error message should be used if the device cannot detect a more specific error.
–141	“Invalid character data”	Either the character data element contains an invalid character, or the particular element received is not valid for the header.
–144	“Character data tool long”	The character data element contains more than twelve characters (see IEEE 488.2, 7.7.1.4).
–148	“Character data not allowed”	A legal character data element was encountered where prohibited by the device.

Error Number	Description	Probable Cause
–150	“String data error”	This error, as well as errors –151 through –159, are generated when parsing a string data element. This particular error message should be used if the device cannot detect a more specific error.
–151	“Invalid string data”	A string data element was expected, but was invalid for some reason (see IEEE 488.2, 7.7.5.2); for example, an END message was received before the terminal quote character.
–158	“String data not allowed”	A string data element was encountered but was not allowed by the device at this point in parsing.
–160	“Block data error	This error, as well as errors –161 through –169, are generated when parsing a block data element. This particular error message should be used if the device cannot detect a more specific error.
–161	“Invalid block data”	A block data element was expected, but was invalid for some reason (see IEEE 488.2, 7.7.6.2); for example, an END message was received before the length was satisfied.
–168	“Block data not allowed”	A legal block data element was encountered but was not allowed by the device at this point in parsing.
–170	“Expression error”	This error, as well as errors –171 through –179, are generated when parsing an expression data element. This particular error message should be used if the device cannot detect a more specific error.
–171	“Invalid expression”	The expression data element was invalid (see IEEE 488.2, 7.7.7.2); for example, unmatched parentheses or an illegal character.
–178	“Expression data not allowed”	A legal expression data was encountered but was not allowed by the device at this point in parsing.

Error Number	Description	Probable Cause
–180	“Macro error”	This error, as well as errors –181 through –189, are generated when defining a macro or executing a macro. This particular error message should be used if the device cannot detect a more specific error.
–181	“Invalid outside macro definition”	Indicates that a macro parameter placeholder ($<number) was encountered outside of a macro definition.
–183	“Invalid inside macro definition”	Indicates that the program message unit sequence, sent with a *DDT or *DMC command, is syntactically invalid (see IEEE 488.2, 10.7.6.3).
–184	“Macro parameter error”	Indicates that a command inside the macro definition had the wrong number or type of parameters.
–200	“Execution error”	This is the generic syntax error for devices that cannot detect more specific errors. This code indicates only that an Execution Error as defined in IEEE 488.2, 11.5.1.1.5 has occurred.
–201	“Invalid while in local”	Indicates that a command is not executable while the device is in local due to a hard local control (see IEEE 488.2, 5.6.1.5); for example, a device with a rotary switch receives a message which would change the switches state, but the device is in local so the message can not be executed.
–202	“Settings lost due to rtl”	Indicates that a setting associated with a hard local control (see IEEE 488.2, 5.6.1.5) was lost when the device changed to LOCS from REMS or to LWLS from RWLS.
–203	“Command protected”	Indicates that a legal password-protected program command or query could not be executed because the command was disabled.

Error Number	Description	Probable Cause
–210	“Trigger error”	-----
–211	“Trigger ignored”	Indicates that a GET, *TRG, or triggering signal was received and recognized by the device but was ignored because of device timing considerations; for example, the device was not ready to respond. Note: a DT0 device always ignores GET and treats *TRG as a Command Error.
–212	“Arm ignored”	Indicates that an arming signal was received and recognized by the device but was ignored.
–213	“Init ignored”	Indicates that a request for a measurement initiation was ignored as another measurement was already in progress.
–214	“Trigger deadlock”	Indicates that the trigger source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be started until a GET is received, but the GET would cause an INTERRUPTED error.
–215	“Arm deadlock”	Indicates that the arm source for the initiation of a measurement is set to GET and subsequent measurement query is received. The measurement cannot be started until a GET is received, but the GET would cause an INTERRUPTED error.
–220	“Parameter error”	Indicates that a program data element related error occurred. This error message should be used when the device cannot detect the more specific errors described for errors –221 through –229.
–221	“Settings conflict”	Indicates that a legal program data element was parsed but could not be executed due to the current device state (see IEEE 488.2, 6.4.5.3 and 11.5.1.1.5.)

Error Number	Description	Probable Cause
–222	“Data out of range”	Indicates that a legal program data element was parsed but could not be executed because the interpreted value was outside the legal range as defined by the device (see IEEE 488.2, 11.5.1.1.5.)
–223	“Too much data”	Indicates that a legal program data element of block, expression, or string type was received and contained more data than the device could handle due to memory or related device-specific requirements.
–224	“Illegal parameter value”	Used where an exact value, from a list of possible choices, was expected.
–225	“Out of memory”	The device has insufficient memory to perform the requested operation.
–226	“Lists not same length”	Attempted to use LIST structure having individual LIST's of unequal lengths.
–230	“Data corrupt or stale”	Possibly invalid data; new reading started but not completed since last access.
–231	“Data questionable”	Indicates that measurement accuracy is suspicious.
–232	“Invalid format”	Indicates that a legal program data element was parsed but could not be executed because the data format or structure is inappropriate. For example when loading memory tables or when sending a SYSTem:SET parameter from an unknown instrument.

Error Number	Description	Probable Cause
–233	“Invalid version”	Indicates that a legal program data element was parsed but could not be executed because the version of the data is incorrect to the device. This particular error should be used when file or block data formats are recognized by the instrument but cannot be executed for reasons of version incompatibility. For example, file or instrument version that are not supported.
–240	“Hardware error”	Indicates that a legal program command or query could not be executed because of a hardware problem in the device. The definition of what constitutes a hardware problem is completely device-specific. This error message should be used when the device cannot detect the more specific errors described for errors –241 through –249.
–241	“Hardware missing”	Indicates that a legal program command or query could not be executed because of missing device hardware; for example, an option was not installed. The definition of what constitutes missing hardware is completely device-specific.
–250	“Mass storage error”	Indicates that a mass storage error occurred. This error message should be used when the device cannot detect the more specific errors described for errors –251 through –259.
–251	“Missing mass storage”	Indicates that a legal program command or query could not be executed because of missing mass storage; for example, an option that was not installed. The definition of what constitutes missing mass storage is device-specific.
–252	“Missing media”	Indicates that a legal program command or query could not be executed because of a missing media; for example, no disk. The definition of what constitutes missing media is device-specific.

Error Number	Description	Probable Cause
–253	“Corrupt media”	Indicates that a legal program command or query could not be executed because of corrupt media; for example, bad disk or wrong format. The definition of what constitutes corrupt media is device-specific.
–254	“Media full”	Indicates that a legal program command or query could not be executed because the media was full; for example, there is no room on the disk. The definition of what constitutes a full media is device-specific.
–255	“Directory full”	Indicates that a legal program command or query could not be executed because the media directory was full. The definition of what constitutes a full media directory is device-specific.
–256	“File name not found”	Indicates that a legal program command or query could not be executed because the file name on the device media was not found; for example, an attempt was made to read or copy a nonexistent file. The definition of what constitutes a file not being found is device-specific.
–257	“File name error”	Indicates that a legal program command or query could not be executed because the file name on the device media was in error; for example, an attempt was made to copy to a duplicate file name. The definition of what constitutes a file name error is device-specific.
–258	“Media protected”	Indicates that a legal program command or query could not be executed because the media was protected; for example, the write-protect tab on a disk was present. The definition of what constitutes protected media is device-specific.

Error Number	Description	Probable Cause
–260	“Expression error”	Indicates that an expression program data element related error occurred. This error message should be used when the device cannot detect the more specific errors described for errors –261 through –269.
–261	“Math error in expression”	Indicates that a syntactically legal expression program data element could not be executed due to a math error; for example, a divide-by-zero was attempted. The definition of math error is device-specific.
–270	“Macro error”	Indicates that a macro-related execution error occurred. This error message should be used when the device cannot detect the more specific errors described for errors –271 through –-279.
–271	“Macro syntax error”	Indicates that a syntactically legal macro program data sequence, according to IEEE 488.2, 10.7.2, could not be executed due to a syntax error within the macro definition (see IEEE 488.2, 10.7.6.3.)
–272	“Macro execution error”	Indicates that a syntactically legal macro program data sequence could not be executed due to some error in the macro definition (see IEEE 488.2, 10.7.6.3.)
–273	“Illegal macro label”	Indicates that the macro label defined in the *DMC command was a legal string syntax, but could not be accepted by the device (see IEEE 488.2, 10.7.3 and 10.7.6.2); for example, the label was too long, the same as a common command header, or contained invalid header syntax.
–274	“Macro parameter error”	Indicates that the macro definition improperly used a macro parameter placeholder (see IEEE 488.2, 10.7.3).

Error Number	Description	Probable Cause
–275	“Macro definition too long”	Indicates that a syntactically legal macro program data sequence could not be executed because the string or block contents was too long for the device to handle (see IEEE 488.2, 10.7.6.1).
–276	“Macro recursion error”	Indicates that a syntactically legal macro program data sequence could not be executed because the device found it to be recursive (see IEEE 488.2, 10.7.6.6).
–277	“Macro redefinition not allowed”	Indicates that a syntactically legal macro label in the *DMC command could not be executed because the macro label was already defined (see IEEE 488.2, 10.7.6.4).
–278	“Macro header not found”	Indicates that a syntactically legal macro label in the *GMC? query could not be executed because the header was not previously defined.
–280	“Program error”	Indicates that a downloaded program-related execution error occurred. This error message should be used when the device cannot detect the more specific errors described for errors –281 through –289. A downloaded program is used to add algorithmic capability to a device. The syntax used in the program and the mechanism for downloading a program is device-specific.
–281	“Cannot create program”	Indicates that an attempt to create a program was unsuccessful. A reason for the failure might include insufficient memory.
–282	“”Illegal program name	The name used to reference a program was invalid; for example, redefining an existing program, deleting a nonexistent program, or in general, referencing a nonexistent program.
–283	“Illegal variable name”	An attempt was made to reference a nonexistent variable in a program.

Error Number	Description	Probable Cause
–284	“Program currently running”	Certain operations dealing with programs may be illegal while the program is running; for example, deleting a running program might not be possible.
–285	“Program syntax error”	Indicates that a syntax error appears in a downloaded program. The syntax used when parsing the downloaded program is device-specific.
–286	“Program runtime error”	-----
–290	“Memory use error”	Indicates that a user request has directly or indirectly caused an error related to memory or <data_handle>, this is not the same as “bad” memory.
–291	“Out of memory”	-----
–292	“Referenced name does not exist”	-----
–293	“Referenced name already exist”	-----
–294	“Incompatible type”	Indicates that the type or structure of a memory item is inadequate.
–300	“Device-specific error”	This is the generic device-dependent error for devices that cannot detect more specific errors. This code indicates only that a Device-Dependent Error as defined in IEEE 488.2, 11.5.1.1.6 has occurred.
–310	“System error”	Indicates that some error, termed “system error” by the device, has occurred. This code is device-dependent.
–311	“Memory error”	Indicates some physical fault in the device's memory, such as parity error.

Error Number	Description	Probable Cause
–312	“PUD memory lost”	Indicates that the protected user data saved by the *PUD command has been lost.
–313	“Calibration memory lost”	Indicates that nonvolatile calibration data used by the *CAL? command has been lost.
–314	“Save/Recall memory lost”	Indicates that the nonvolatile data saved by the *SAV? command has been lost.
–315	“Configuration memory lost”	Indicates that nonvolatile configuration data saved by the device has been lost. The meaning of this error is device-specific.
–320	“Storage fault”	Indicates that the firmware detected a fault when using data storage. This error is not an indication of physical damage or failure of any mass storage element.
–321	“Out of memory”	An internal operation required more memory than was available.
–330	“Self-test failed”	-----
–340	“Calibration failed”	-----
–350	“Queue overflow”	A specific code entered into the queue in lieu of the code that caused the error. This code indicates that there is no room in the queue and an error occurred but was not recorded.
–360	“Communication error”	This is the generic communication error for devices that cannot detect the more specific errors described for errors –361 through –363.
–361	“Parity error in program message”	Parity bit not correct when data received, for example, on a serial port.
–362	“Framing error in program message”	A stop bit was not detected when data was received for example, on a serial port (for example, a baud rate mismatch).

Error Number	Description	Probable Cause
–363	“Input buffer overrun”	Software or hardware input buffer on serial port overflows with data caused by improper or nonexistent pacing.
–365	“Time out error”	This is a generic device-dependent error.
–400	“Query error”	This is the generic query error for devices that cannot detect more specific errors. This code indicates only that a Query Error as defined in IEEE 488.2, 11.5.1.1.7 and 6.3 has occurred.
–410	“Query INTERRUPTED”	Indicates that a condition causing an INTERRUPTED Query error occurred (see IEEE 488.2, 6.3.2.3); for example, a query followed by DAB or GET before a response was completely sent.
–420	“Query UNTERMINATED”	Indicates that a condition causing an UNTERMINATED Query error occurred (see IEEE 488.2, 6.3.2.2); for example, the device was addressed to talk and an incomplete program message was received.
–430	“Query DEADLOCKED”	Indicates that a condition causing an DEADLOCKED Query error occurred (see IEEE 488.2, 6.3.1.7); for example, both input and output buffers are full and the device cannot continue.
–440	“Query UNTERMINATED after indefinite response”	Indicates that a query was received in the same program message after an query requesting an indefinite response was executed (see IEEE 488.2, 6.5.7.5).
–500	“Power on”	The instrument has detected an off to on transition in its power supply.
–600	“User request”	The instrument has detected the activation of a user request local control

Error Number	Description	Probable Cause
–700	“Request control”	The instrument requested to become the active IEEE 488.1 controller-in-charge.
–800	“Operation complete”	The instrument has completed all selected pending operations in accordance with the IEEE 488.2, 12.5.2 synchronization protocol.

Index

A

AC requirements 8
address, GPIB 30, 31
adjusting contrast 18
after-sales service 78
angled brackets 46

B

backlight
  default 21
  setting 18
baud rate, setting 32
benchtop, installing in a rackmount 10
books, SCPI and IEEE 488.2 44
braces 46
brackets
  angled 46
  square 45

C

capacitors 7
caution
  of personal hazard 4
  of product hazard 4
certification information vi
changing
  communication settings 29
  drive current 20
  parameters 17
cleaning
  EUI connectors 66
  fiber ends 23
  front panel 65
code writing, GPIB 49
codes, error 51
colon 46
comma 47
commands
  IEEE 488.2 112
  IEEE 488.2, quick reference 111
  SCPI 44
commands, specific 135
common commands 111
communication
  changing settings 29
  speed 32
configuring display 18, 19
connectors, cleaning 66
contrast, setting 18
control character 33
conventions, programming 44, 47, 50
conventions, safety 4
covers, unit 7
current software version 72
current wavelength 13
current, electrical 8
current, modifying 20
customer service 82

D

data
  display 12
  transmission 33
  types 89
data input
  IEEE 488.2 90
  SCPI 109
data output IEEE 488.2
  special 102, 110
  standard 99
deactivating backlight 18
default values 21
defining values 17
diagram of the menus 17
digit 45

disconnecting unit 6
display
configuring 18, 19
overview 12
drive current
indicator 13
modifying 20
drivers, LabVIEW 14

E

emission, light 25
end of message 48
EOI (End or Identify) 29
EOS (End of String) 29
equipment returns 82
error messages in remote control 51
error/event queue 38
errors related to unit 73
EUI
baseplate 11
connector adapter 11
dust cap 11
EUI connectors, cleaning 66
examples
LabVIEW 54
EXFO universal interface. *see* EUI

F

fiber ends, cleaning 23
flow control
default 21
setting 33
front panel, cleaning 65
fuse
holder 2
replacement 7, 69
type 69

G

GPIB
address 30, 31
commands 111
default address 21
parameters 29
port 2, 27
troubleshooting 77
GPIB commands 135
ground 2

H

handshake, software 33

I

identification label 78
IEEE 488.2 commands 111, 112
indicator
locked keyboard 13
remote control 13, 27
source status 13
wavelength/drive current 13
indoor use 7
inlets 6
input
buffer 35
IEEE 488.2 data types 90
SCPI data types 109
input current 8
installing
rackmount 9
unit 7
instrument control settings 29
internal memory 13
inverted video mode 19

K

keyboard
  definition ... 13
  locked indicator ... 13
keywords, SCPI ... 46

L

label
  source port ... 1
label, identification ... 78
LabVIEW
  demo application ... 54
  installing drivers ... 14
  principles ... 52
laser radiation hazard sticker ... 1
light emission ... 25
linking units
  GPIB port ... 27
  serial port ... 28
list separator ... 46
locked keyboard ... 13

M

main window ... 12
maintenance
  EUI connectors ... 66
  front panel ... 65
  general information ... 65
mandatory commands ... 111, 112
marker, remote programming state (RM) ... 124
maximum input current ... 8
memory, internal ... 13
menu
  diagram ... 17
  Setup ... 17
message termination ... 48
messages, remote control ... 35
mnemonic, definition ... 46
modifying drive current ... 20
mounting EUI connector adapter ... 11
multiple capabilities ... 45

N

non-volatile memory ... 13

O

on/off
  SLED source ... 25
  unit ... 12
operating source ... 23
original parameters ... 21
output IEEE 488.2
  special ... 102, 110
  standard ... 99
output queue ... 37

P

parameters
  backlight ... 18
  contrast ... 18
  GPIB ... 29
  refresh rate ... 17
  resetting ... 21
  RS-232 ... 29
  saving ... 13
  setting ... 17
  video mode ... 19
pipe character ... 46
port
  GPIB ... 2, 27
  serial ... 2, 28
power
  cable ... 6
  indicator ... 13
  inlet ... 2
  on/off ... 12
  plug ... 6
  up/down ... 20
power source, AC ... 8

problems with GPIB 77
product
  identification label 78
  specifications 85
PROGRAM MESSAGE TERMINATOR 48
programmable instruments, standards . 27, 44
programming, commands and queries 111
programming, GPIB 49

Q

queries, IEEE 488.2 111
queries, specific 135
queue
  error/event 38
  input 35
  output 37

R

rackmount, installing 9
receiving data
  with GPIB port 27
  with serial port 28
refresh rate setting 17
register
  diagram 42, 43
  ESE 40
  ESR 39
  SRE 41
  STB 41
remote control
  default 21
  description of commands 44
  error messages 51
  GPIB 27
  indicator 13, 27
  messages 35
  parameters 29
  RS-232 28
repairing unit 7
replacing fuses 69
required commands 111
resetting the unit 21
return merchandise authorization (RMA) 82
RM marker 124
RS-232
  commands 111
  connector pinout configuration 28
  for software upgrade 70
  parameters 29
  port 28
  speed 32
RS-232 commands 135
rules
  programming 50
  syntax 47
  syntax conventions 44

S

safety
  caution 4
  conventions 4
  power cable 6
  warning 4
SCPI
  commands 44
  data types 109
  guidelines 27, 44
  parameters 29
  references 44
SCPI commands 135
screen
  adjusting 18, 19
  overview 12
self-test 12
semicolon 47
sending control character 33
sending data
  with GPIB port 27
  with serial port 28
separator 46, 47

serial
communication ... 33
port ... 2
service and repairs ... 82
service centers ... 83
service request enable register (SRE) ... 41
setting
backlight ... 18
baud rate ... 32
contrast ... 18
flow control ... 33
instrument control utility ... 29
parameters ... 17
refresh rate ... 17
video mode ... 19
settings for remote control ... 29
setup button ... 17
shipping to EXFO ... 82
SLED, turning on/off ... 25
software
handshake ... 33
upgrade ... 70
source
activating/deactivating ... 25
changing drive current ... 20
default status ... 21
error messages ... 73
label ... 1
operating ... 23
status indicator ... 13
space ... 45
specific commands and queries ... 135
specifications, product ... 85
square brackets ... 45
standard event status
enable register (ESE) ... 40
register (ESR) ... 39
standard status data structure
diagram ... 42, 43
general ... 39
status
byte register (STB) ... 41
of source ... 13
remote control ... 27
storage requirements ... 65
symbols, safety ... 4
syntax
rules ... 44, 47
SCPI ... 44
symbols ... 45

T

technical specifications ... 85
technical support ... 78
temperature for storage ... 65
transportation requirements ... 65, 78
troubleshooting, GPIB ... 77
turning on/off
source ... 25
unit ... 12
type of fuses ... 7
type, serial communication ... 33
types, data ... 89

U

unit
covers ... 7
disconnecting ... 6
installing ... 7
powering ... 12
repairing ... 7
status ... 27
ventilation ... 7
up/down, source power ... 20
upgrading the software ... 70
using source ... 23

V

value, defining 17
ventilation 7
vertical bar 46
video mode setting 19

W

warranty
  certification 81
  exclusions 81
  general 79
  liability 80
  null and void 79
wavelength indicator 13
white space 45
window, display 12
wsp 45

X

Xoff/Xon characters 33

NOTICE
通告

CHINESE REGULATION ON RESTRICTION OF HAZARDOUS SUBSTANCES
中国关于危害物质限制的规定

NAMES AND CONTENTS OF THE TOXIC OR HAZARDOUS SUBSTANCES OR ELEMENTS CONTAINED IN THIS EXFO PRODUCT
包含在本 **EXFO** 产品中的有毒有害物质或元素的名称和含量

O	Indicates that this toxic or hazardous substance contained in all of the homogeneous materials for this part is below the limit requirement in SJ/T11363-2006 表示该有毒有害物质在该部件所有均质材料中的含量均在 SJ/T11363-2006 标准规定的限量要求以下。
X	Indicates that this toxic or hazardous substance contained in at least one of the homogeneous materials used for this part is above the limit requirement in SJ/T11363-2006 表示该有毒有害物质至少在该部件的某一均质材料中的含量超出 SJ/T11363-2006 标准规定的限量要求。

Part Name 部件名称	Toxic or hazardous Substances and Elements 有毒有害物质和元素					
	Lead 铅 (Pb)	Mercury 汞 (Hg)	Cadmium 隔 (Cd)	Hexavalent Chromium 六价铬 (Cr VI)	Polybrominated biphenyls 多溴联苯 (PBB)	Polybrominated diphenyl ethers 多溴二苯醚 (PBDE)
Enclosure 外壳	O	O	O	O	O	O
Electronic and electrical sub-assembly 电子和电子组件	X	O	X	O	X	X
Optical sub-assembly[a] 光学组件 [a]	X	O	O	O	O	O
Mechanical sub-assembly[a] 机械组件 [a]	O	O	O	O	O	O

a. If applicable.
如果适用。

MARKING REQUIREMENTS
标注要求

Product 产品	Environmental protection use period (years) 环境保护使用期限 (年)	Logo 标志
This Exfo product 本 EXFO 产品	10	
Battery[a] 电池 [a]	5	

a. If applicable.
如果适用。

P/N:1064481

www.EXFO.com · info@exfo.com

CORPORATE HEADQUARTERS	400 Godin Avenue	Quebec (Quebec) G1M 2K2 CANADA Tel.: 1 418 683-0211 · Fax: 1 418 683-2170
EXFO AMERICA	3400 Waterview Parkway Suite 100	Richardson, TX 75080 USA Tel.: 1 972-761-927 · Fax: 1 972-761-9067
EXFO EUROPE	Winchester House, School Lane	Chandlers Ford, Hampshire S053 4DG ENGLAND Tel.: +44 2380 246 800 · Fax: +44 2380 246 801
EXFO ASIA-PACIFIC	100 Beach Road, #25-01/03 Shaw Tower	SINGAPORE 189702 Tel.: +65 6333 8241 · Fax: +65 6333 8242
EXFO CHINA	Beijing Global Trade Center, Tower C, Room 1207, 36 North Third Ring Road East, Dongcheng District	Beijing 100013 P. R. CHINA Tel.: +86 (10) 5825 7755 · Fax: +86 (10) 5825 7722
EXFO SERVICE ASSURANCE	270 Billerica Road	Chelmsford MA, 01824 USA Tel.: 1 978 367-5600 · Fax: 1 978 367-5700
EXFO FINLAND	Elektroniikkatie 2	FI-90590 Oulu, FINLAND Tel.: +358 (0) 403 010 300 · Fax: +358 (0) 8 564 5203
TOLL-FREE	(USA and Canada)	1 800 663-3936

© 2013 EXFO Inc. All rights reserved.
Printed in Canada (2013-03)

EXPERTISE REACHING OUT

Artisan Technology Group is an independent supplier of quality pre-owned equipment

Gold-standard solutions

Extend the life of your critical industrial, commercial, and military systems with our superior service and support.

We buy equipment

Planning to upgrade your current equipment? Have surplus equipment taking up shelf space? We'll give it a new home.

Learn more!

Visit us at **artisantg.com** for more info on price quotes, drivers, technical specifications, manuals, and documentation.

Artisan Scientific Corporation dba Artisan Technology Group is not an affiliate, representative, or authorized distributor for any manufacturer listed herein.

We're here to make your life easier. How can we help you today?

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

ARTISAN® TECHNOLOGY GROUP