Laser Receiver Control Panel (LRCP) Software

User Manual

077-1127-01

Tektronix®

Laser Receiver Control Panel (LRCP) Software

User Manual

www.tek.com

077-1127-01

Contacting Tektronix

Tektronix, Inc.
14150 SW Karl Braun Drive
P.O. Box 500
Beaverton, OR 97077
USA

For product information, sales, service, and technical support:

- In North America, call 1-800-833-9200.
- Worldwide, visit www.tek.com to find contacts in your area.

Warranty

Tektronix warrants that this product will be free from defects in materials and workmanship for a period of one (1) year from the date of shipment. If any such product proves defective during this warranty period, Tektronix, at its option, either will repair the defective product without charge for parts and labor, or will provide a replacement in exchange for the defective product. Parts, modules and replacement products used by Tektronix for warranty work may be new or reconditioned to like new performance. All replaced parts, modules and products become the property of Tektronix.

In order to obtain service under this warranty, Customer must notify Tektronix of the defect before the expiration of the warranty period and make suitable arrangements for the performance of service. Customer shall be responsible for packaging and shipping the defective product to the service center designated by Tektronix, with shipping charges prepaid. Tektronix shall pay for the return of the product to Customer if the shipment is to a location within the country in which the Tektronix service center is located. Customer shall be responsible for paying all shipping charges, duties, taxes, and any other charges for products returned to any other locations.

This warranty shall not apply to any defect, failure or damage caused by improper use or improper or inadequate maintenance and care. Tektronix shall not be obligated to furnish service under this warranty a) to repair damage resulting from attempts by personnel other than Tektronix representatives to install, repair or service the product; b) to repair damage resulting from improper use or connection to incompatible equipment; c) to repair any damage or malfunction caused by the use of non-Tektronix supplies; or d) to service a product that has been modified or integrated with other products when the effect of such modification or integration increases the time or difficulty of servicing the product.

THIS WARRANTY IS GIVEN BY TEKTRONIX WITH RESPECT TO THE PRODUCT IN LIEU OF ANY OTHER WARRANTIES, EXPRESS OR IMPLIED. TEKTRONIX AND ITS VENDORS DISCLAIM ANY IMPLIED WARRANTIES OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. TEKTRONIX' RESPONSIBILITY TO REPAIR OR REPLACE DEFECTIVE PRODUCTS IS THE SOLE AND EXCLUSIVE REMEDY PROVIDED TO THE CUSTOMER FOR BREACH OF THIS WARRANTY. TEKTRONIX AND ITS VENDORS WILL NOT BE LIABLE FOR ANY INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES IRRESPECTIVE OF WHETHER TEKTRONIX OR THE VENDOR HAS ADVANCE NOTICE OF THE POSSIBILITY OF SUCH DAMAGES.

[W2 – 15AUG04]

Table of Contents

Install software

Configure OM instrument network settings

Operating basics

Appendices

Index

List of Figures

List of Tables

Preface

This document describes how to install, configure, and operate the Laser Receiver Control Panel (LRCP), used to locate and control Tektronix OM5110, OM4245, OM4225, OM4106D, OM4006D, OM2210, and OM2012 instruments that are connected to a local network. The LRCP interface automates locating and configuring instruments and simplifies laser, modulator, and driver amplifier control, eliminating the need to use low-level ITLA commands.

The LRCP software application's main use is to control OM5110 or OM2012 instruments that are used in a stand-alone configuration as individual instruments or used together (OM5110 plus OM2012). LRCP can also control and configure other OM instruments (OM4200 series, OM4000 series, OM2210), though these instruments are normally accessed from the OM1106 Modulation Analysis Software. See the *OM1106 Modulation Analysis Software User Manual* (Tektronix part number 077-1093-xx), available to download from the Tektronix Web site (www.tek.com).

Install software

Install software

PC hardware requirements

The following are the PC requirements to install and run the standalone LRCP software. The term PC applies to a supported oscilloscope, PC, or laptop on which the software is installed, and that is connected to OM instruments over a network.

Item	Description
Operating system	U.S.A. Microsoft Windows 7 (32- or 64-bit)
Processor	Intel i7, i5 or equivalent; min clock speed 2 GHz
RAM	Minimum: 4 GB
Hard Drive Space	Minimum: 20 GB
Networking	Gigabit Ethernet (1 Gb/s) or Fast Ethernet (100 Mb/s)
Other Hardware	2 USB 2.0 ports
Adobe Reader	Adobe reader used for viewing PDF format files

What software to install

The following table shows which software to install based on the OM instrument(s) you plan to operate.

Table 1-1: Software to install based on instrument configuration

		When used with		
Instrument	When used stand-alone	OM5110	OM2012	OM4000 series or OM2210 [1]
OM5110	LRCP [2] AWG file library (optional)	LRCP [2] AWG file library (optional)	LRCP [2] AWG file library (optional)	OM1106 [3] MATLAB Scope Utility AWG file library (optional)
OM2012	LRCP [2]	LRCP [2]	LRCP [2]	OM1106 [3] MATLAB Scope Utility

1 OM4000 series = OM4245, OM4225, OM4106D, OM4006D.

2 LRCP: Laser Receiver Control Panel.

3 OM1106 software included with these instruments or is available as a purchased option.

Install LRCP software

Install the LRCP software if you are using the OM5110 or OM2012 alone, or in combination with each other only (in other words, not with OM4200, OM4000, or OM2210 instruments). (See Table 1-1 on page 1-1.)

You do ***not*** need to install the LRCP software for the OM5110 or OM2012 if:

- You already have the LRCP software installed (version 2.1 or greater).
- You have already installed the OM1106 software (version 2.1 or later) as part of the OM4200, OM4000, or OM2210 product software install.
- You are going to use the OM5110 or OM2012 with OM4200, OM4000, or OM2210 instruments. In this case, use or install the OM1106 software that ships with those instruments. The OM1106 software incorporates the LRCP controls. See the OM1106 software user manual for more information.

NOTE. *Read the installation notes and/or instructions that are in each application's installation folder before installing that software. Only install the software that is appropriate for your OM instrument, PC, and oscilloscope configuration.*

To install the LRCP standalone software:

1. Insert the software USB flashdrive in the PC.
2. Navigate to the LRCP folder on the install media.
3. Richt-click **SetupLRCP_2.1.2.XXXX.exe** and select **Install**. Follow on-screen instructions.

   The LRCP software is installed at:
   C:\Program Files(x86)\TekApplication\LRCP

AWG file library for OM5110 (optional)

The product software USB flashdrive also contains an AWG file library (at OM5110\AWG Files) for use with Tektronix AWG70001A and AWG70002A Arbitrary Waveform Generators, for generating standard and custom optical modulation signals. These files are precompensated to work with a typical AWG70001A and OM5110 combination.

See the file *OM5110 app note.pdf* in the AWG Files folder of the software media for information on using the AWG library files with the OM5110.

Configure OM instrument network settings

Verify or set instrument IP address

Before using OM software to operate the OM instruments, you must make sure that IP addresses of the instruments are set correctly for your network, to communicate with the OM software. The following sections describe how to connect OM instruments to DHCP and nonDHCP networks.

All OM instruments must be set to the same network subnet (DHCP-enabled networks do this automatically) to communicate with each other.

Verify instrument connectivity for DHCP-enabled network

OM instruments are set by default to use DHCP to automatically assign IP addresses. Therefore you do not need to specifically set the instrument IP address when connected to a DHCP network, as the DHCP server automatically assigns an IP address during instrument power-on (when the rear-panel power switch is set to On or the unit is rebooted by holding the front-panel button for 10 seconds).

The following procedure describes how to verify that OM1106 or LRCP software can detect and connect to OM instruments on a DHCP-enabled network that are set to use DHCP-generated IP addresses:

Prerequisites:

- LRCP software installed on the controller PC.
- OM instrument(s) and the controller PC are connected to the same DHCP-enabled network on the same subnet.
- OM instruments are set to use DHCP (default).

Verify instrument network connections on DHCP network

1. Connect the OM instrument(s) to the DHCP-enabled network.
2. Power on the OM instrument. The instrument queries the DHCP server to obtain an IP address. Wait until the front panel Enable/Standby button light turns off, indicating it has obtained an address.
3. Push the front panel Enable/Standby button to enable the network connection (button light turns On).
4. Double-click the **LRCP** desktop icon.
5. Click **Device Setup** to open the Device Setup dialog box.
6. Click **Auto Configure** to search the network and list all detected OM instruments. If all connected instruments are listed, then correct IP addresses were automatically assigned.

If the Device Setup dialog does not list all connected instruments:

- Verify that instruments are connected to the correct DHCP-enabled network and that the DHCP server is set to provide IP addresses on the same subnet
- Verify that instruments are powered on after connecting to the network
- Reboot the missing OM instruments by holding down their front-panel power button for 10 seconds, then repeat the Auto Configure.
- Work with your IT resource to resolve the connection problem.

7. Click **OK** to close the Device Setup dialog box and return to the LRCP window. The detected instruments display as tabs in the main LRCP window. Click a tab to show that instrument's settings.

You are now ready to use LRCP to configure and control instruments. (See page 3-1, *LRCP operating basics*.)

Set instrument IP address for use on non-DHCP network

To connect an OM series instrument to a non-DCHP network, you must set the IP address and related settings of the OM instrument to match those of your non-DHCP network. All devices on non-DHCP network (OM instruments, PCs running OM software, and other remotely accessed instruments such as oscilloscopes) need the same subnet values (typically the first three number groups of the IP address) to communicate, and a unique instrument identifier (the fourth number group of the IP address) to identify each instrument.

Work with your network administrator to obtain a unique IP address for each device. If your network administrator needs the MAC address of the OM instrument, the MAC address is located on the instrument rear panel label.

***NOTE.** Make sure to record the IP addresses used for each OM instrument, or attach a label with the new IP address to the instrument.*

If you are setting up a new isolated network just for controlling OM and associated instruments, Tektronix recommends using the OM instrument default IP subnet address of **172.17.200.XXX**, where XXX is any number between 0 and 255.

***NOTE.** Use the system configuration tools on the oscilloscope and computer to set their IP addresses.*

***NOTE.** If you need to change the default IP address of more than one OM instrument, you must connect each instrument separately to change the IP address.*

There are two ways to change the IP address of an OM instrument:

- Connect the OM instrument(s) to a DHCP-enabled network and use LRCP to change the IP address (easiest)
- Connect the OM instrument directly to a PC that is already set to the same IP address subnet as the OM instrument, and use LRCP to change the IP address.

Change OM instrument IP address using DHCP network

To use a DHCP network to change the IP address of an OM instrument:

1. Connect the OM instrument(s) to the DHCP-enabled network.
2. Power on the OM instrument. The instrument queries the DHCP server to obtain an IP address. Wait until the front panel Enable/Standby button light turns off, indicating it has obtained an address. Push the front panel Enable/Standby button to enable the network connection (button light turns On).
3. Double-click the **LRCP** desktop icon.
4. Click **Device Setup** to open the Device Setup dialog box.
5. Click **Auto Configure** to search the network and list all detected OM instruments.

   If the Device Setup dialog does not list all connected instruments:

   - Verify that instruments are connected to the correct DHCP-enabled network
   - Verify that instruments are powered on
   - Reboot the missing OM instruments by holding down their front-panel power button for 10 seconds, and then repeat the Auto Configure
   - Work with your IT resource to resolve the connection problem
6. Double-click in the **IP Address** field of the instrument to change and enter the new IP address for that OM instrument.
7. Click the corresponding **Set IP** button. A warning dialog box appears indicating that the IP address will be changed and that you must record the new IP address. Losing the IP address will require connecting the instrument to a DHCP router to determine its IP address.
8. Click **Yes** to set the new IP address.
9. Edit the Gateway and Net Mask (obtain this information from your network support).
10. Click **OK**.
11. Repeat steps 6 through 10 to change any other OM instrument IP addresses.
12. Exit the OM program.

13. Power off the OM instrument(s) and connect it to the non-DHCP network.
14. Run LRCP on the non-DHCP network and use the **Auto Config** button in the Device Setup dialog box to verify that the instrument is listed with the new IP address.

Change OM instrument IP address using direct PC connection

To use a direct PC connection to change the default IP address of an OM instrument, you need to:

- Install LRCP on the PC
- Use the Windows Network tools to set the IP address of the PC to match that of the OM series instrument whose IP address you need to change
- Connect the OM instrument directly to the PC, or through a hub or switch (not over a network)
- Use LRCP to change the OM instrument IP address

Do the following steps to use a direct PC connection to change the IP address of an OM series instrument:

NOTE. *If you need to change the default IP address of more than one OM instrument using this procedure, you must connect each instrument separately to change the IP address.*

Set PC IP address to match OM instrument.

1. On the PC with LRCP installed, click **Start > Control Panel**.
2. Open the **Network and Sharing Center** link.
3. Click the **Manage Network Connections** link to list connections for your PC
4. Right-click the **Local Area Connection** entry for the Ethernet connection and select **Properties** to open the Properties dialog box.
5. Select **Internet Protocol Version 4** and click **Properties**.
6. Enter a new IP address for your PC, using the same first three numbers as used by the OM instrument. For example, **172.17.200.200**. This sets your PC to the same subnet (first three number groups) as the default IP address setting for the OM series instruments.
7. Click **OK** to set the new IP address.
8. Click **OK** to exit the Local Area Connection dialog box.
9. Exit the **Control Panel** window.

Run LRCP on direct-connected PC to change OM instrument IP address.

1. Connect the OM instrument to the PC (directly, or through a hub or switch connected to the PC). Do not connect over a network.
2. Power on the OM instrument. Wait until the front panel Enable/Standby button light turns Off.
3. Push the **Enable/Standby** button again to enable the network connection (button light turns On).
4. Access the OM software connection setup controls:
5. Double-click the **LRCP** desktop icon.
6. Click **Device Setup** to open the Device Setup dialog box.
7. Click **Auto Configure** to search the network and list all detected OM instruments.

   If the Device Setup dialog does not list all connected instruments:

   - Verify that instruments are connected to the correct DHCP-enabled network
   - Verify that instruments are powered on
   - Reboot the missing OM instruments by holding down their front-panel power button for 10 seconds, and then repeat the Auto Configure
   - Work with your IT resource to resolve the connection problem
8. Double-click in the **IP Address** field of the instrument to change and enter the new IP address for that OM instrument.
9. Click the corresponding **Set IP** button. A warning dialog box appears indicating that the IP address will be changed and that you must record the new IP address. Losing the IP address will require connecting the instrument to a DHCP router.
10. Click **Yes** to set the new IP address.
11. Edit the Gateway and Net Mask (obtain this information from your network support).
12. Click **OK**.
13. Exit the OM program.
14. Power off the OM instrument.
15. Disconnect the OM instrument from the PC.
16. Repeat steps 8 through 15 to change any other OM instrument IP addresses.

17. Connect the OM instrument to the target network.

18. Run the LRCP software on the PC connected to the same network as the OM instrument to verify that the OM software detects the OM instrument.

Operating basics

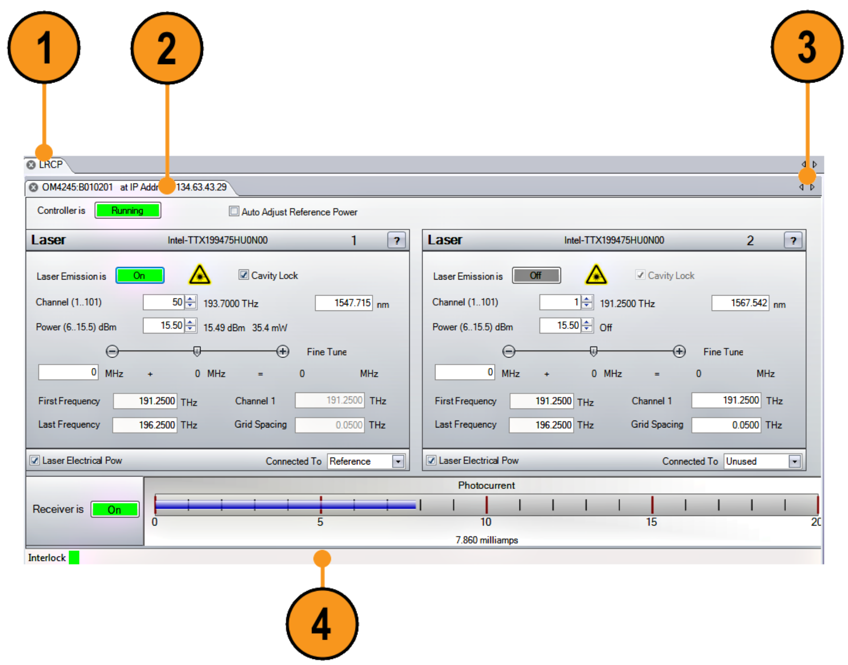
LRCP operating basics

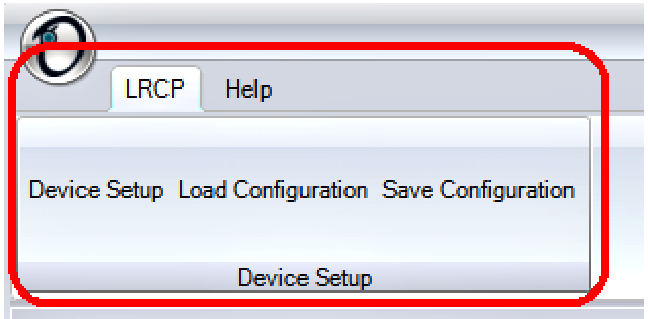
The Laser Receiver Control Panel (LRCP) stand-alone software automates locating and configuring all OM devices connected to a network. It also provides a Windows Communication Foundation (WCF) service interface to allow a local or remote process to operate the connected devices for ATE (automated test equipment) applications. The LRCP program can control any number of OM-series devices (OM5110, OM4200, OM4000 and OM2000 series).

***NOTE.** The LRCP software does not acquire or analyze data from connected instruments. Use the OM1106 Optical Modulation Analyzer software to acquire, demodulate, analyze, and visualize complex modulated signals.*

Start LRCP

Double-click the **LRCP** icon on the PC desktop to open the application user interface.





User interface elements

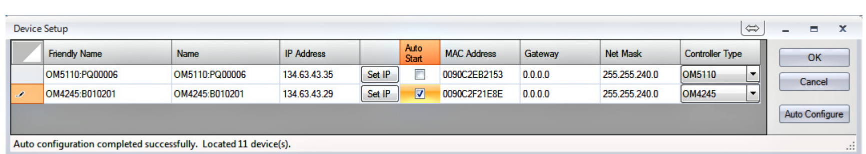
Item	Description
1	**Menu bar** lists available application actions. Click the LRCP menu item to access the Device Setup function for searching and listing instruments connected to the local network.
2	**Controller tabs** that represent one physical Laser Control device (for example, an OM2210, OM4245, and so on) that was detected on the local network. Each tab shows the laser, modulator, and/or driver amp controls associated with that instrument. The image shows the controls for an OM4225 instrument. (See page 3-6, *Controller tab fields*.)
3	**Tab scroll arrows** scroll tabs horizontally when there are more tabs to view than can be displayed on the screen. A dark arrow indicates that there are more tabs to view in that direction.
4	**Status bar** shows overall state of the communications with the instrument controllers. Each controller has a unique status bar. Erorr messages are shown in the right end of the status bar.

Detect OM instruments on a network

When you first start up LRCP you need to run the **Device Setup** task to detect all OM instruments that are on the local network. You also need to run Device Setup any time you add or remove instruments from the network.

To have LRCP detect network-connected instruments:

1. Start LRCP (double-click the **LRCP** desktop icon.
2. Click **LRCP > Device Setup** to open the Device Setup dialog box.



3. Click **Auto Configure** (right edge of dialog box) to search the network and list all detected OM instruments. This search can take a few minutes.

If the Device Setup dialog does not list all known connected instruments:

- Verify that OM instruments are connected to the correct network
- Verify that instruments are powered on and their network connection is enabled (the On/ Standby button on the front panel is on)
- Reboot the missing OM instruments by holding down their front-panel power button for 10 seconds, and then repeat the Auto Configure
- If the above items do not help, work with your IT resource to resolve the connection problem

4. (Optional) Use the Friendly Name field to attach custom labels to OM instruments that help you identify the type and/or location of the instruments. Friendly Names are retained in the LRCP software and are tied to the corresponding instrument MAC address.
5. (Optional) Use the **Set IP** button to manually set the instrument IP address. This is only necessary in a network environment that is not using DHCP to automatically assign IP Addresses. (See page 2-1, *Verify or set instrument IP address*.) The Set IP button only changes the IP address and does not save other modified fields like Friendly Name.
6. (Optional) Select Auto Start to enable auto connection and configuration of this hardware when the LRCP is launched. The Auto Start hardware is configured at OUI/LRCP launch to match the state when the LRCP was last closed. The hardware must be present at the last known IP address for the automatic connection to work.
7. Click **OK** to exit the dialog and save any changes (such as Friendly Name). LRCP lists the detected OM devices as tabs on the main screen, using the friendly name and IP address to allow for easy identification.

NOTE. *If you do not click OK, the listed instruments are not connected to LRCP or saved in the software.*

NOTE. *LRCP does not automatically update the connected devices list on startup. Disconnected or powered-off instruments will still be shown in the list and be shown as offline. You should run the Auto Configure task after starting LRCP to update the connected device list or when the software has been running for a long time.*

Connect to an OM instrument

Once detected using the Device Setup dialog box, LRCP lists the OM devices as tabs on the main screen, using the friendly name and IP address to allow for easy identification.

To connect to an instrument:

1. Click an instrument tab.
2. Click the **Offline** button. The button changes colors to show the connection status:
   a. The button turns yellow and reads "Connecting…" to show that a physical network connection is being established over a socket.
   b. The button turns teal and reads "Connected…" to show that a session is established between the device and Control Panel. Commands are sent to initialize the communications with the laser and identify their capabilities.
   c. The button turns bright green when the controller and lasers are ready to operate from the software.

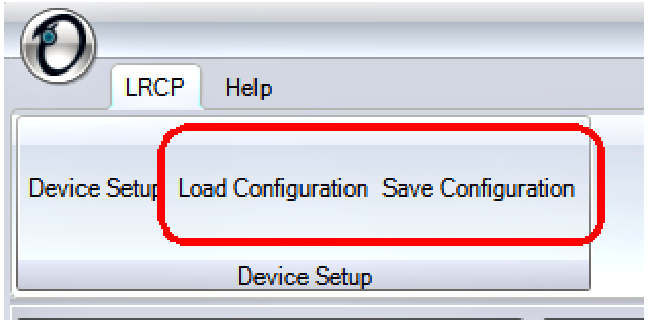
   NOTE. *The button color scheme (bright green = running or active; gray = off line or inactive; red = warning or error state) is consistent throughout the application.*

3. Once the instrument is connected, the tab populates with controls and fields relevant to the connected OM device (instrument name, laser manufacturer and model, available parameters, and so on). You can now change settings and turn the laser(s) on or off.

NOTE. *The very first time the LRCP connects to an OM5110, there is a delay while the LRCP calculates the initial modulator parameters so that they may be stored away in the LRCP Program Files directory. The modulator parameters, including optimum bias voltages and Vpi voltages for the various modulator sections, are needed to obtain proper optical bias for the modulator. The LRCP saves the current state of each OM5110 on first connection so that you can restore the parameters if needed. More information on setting the modulator parameters using the "Set Params" button is later in this section.*

Saving and loading configurations

Use the Save Configuration and Load Configuration buttons on the LRCP ribbon to save the settings of all connected OM hardware for later reloading.



Save Configuration

Click the Save Configuration button to open up a "Save As" dialog box. Browse to the desired folder and enter a name for the configuration file. Click save to store all the settings of the OM hardware showing a green "Running" status on its LRCP tab to this file.

Load Configuration

NOTE. *If there is any possibility that any OM hardware IP addresses have changed since the LRCP was last used, it is a good idea to run a* ***LRCP > Device Setup > AutoConfigure*** *before loading a configuration, to ensure that the IP addresses are known for the OM hardware to be configured.*

Click the Load Configuration to open the "Open" dialog box. Browse to the desired folder and find the file. Click Open to restore all the settings to the OM hardware listed in the specified configuration file. If the OM hardware is not in the green Running state, the LRCP attempts to connect to the hardware at the last known IP address so that the settings of that hardware may be restored. Laser emission status will also be restored. This means that if the laser was on when the configuration was saved, that laser emission is turned on when the configuration is loaded.

Loading a configuration may take some time. The status of each piece of OM hardware is displayed on its LRCP tab in the connect button area. The status of "configuring" indicates that settings are being restored. Wait for the status to change to Running before attempting to use the hardware.

Controller tab fields

The Laser controls

The Laser control area of the LRCP software displays available laser control functions for connected instruments with laser output capability.

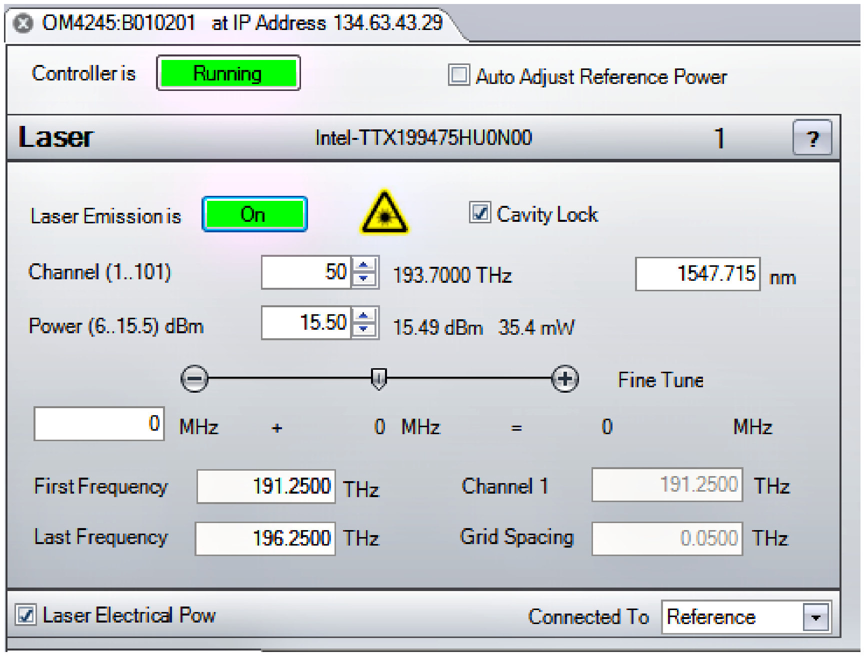


Table 3-1: Laser controls

Control	Description
Auto Adjust Reference Power	Enables the automatic control of the power setting of the laser identified as the Reference laser. The automatic control loop will set the laser power to near maximum unless the Signal input power is so large that the total photocurrent is above the recommended range. If the total photocurrent is too high, the Reference laser power setting is reduced to bring the photocurrent into the recommended range.
Laser Emission is	Enables or disables laser emission output from the front panel connectors. The emission status is indicated both by the green color of the button and by the green LED on the instrument front panel.

Table 3-1: Laser controls (cont.)

Control	Description
Cavity Lock	Enables or disables the ITLA laser cavity lock. Certain laser models have a cavity lock feature that increases their frequency accuracy at the expense of dithering the frequency; this feature is toggled with the Cavity Lock button. Cavity Lock is necessary to tune the laser, but can be unchecked to suppress the dither. Ordinarily, Cavity Lock should be enabled (selected) so that the laser is able to tune, change power level, and lock on to its frequency reference. However, once tuning is complete and the laser has stabilized, you can disable Cavity Lock to turn off the frequency dither needed for locking the laser to its reference. The laser can hold its frequency for days without the benefit of the frequency dither. This feature is helpful where the lowest phase noise is required.
Channel	Sets the laser channel. Type a number or use the up/down arrows to choose a channel. The range of channels available depends on the type of laser, the First Frequency, and the Grid. The finer the Grid, the more channels are available for a given laser. The channel range is indicated next to the word Channel. The laser channel can also be set by entering a wavelength in the text box to the right of the channel entry. The laser will tune to the nearest grid frequency.
Power	Sets the laser power level. Type or use the up/down arrows to select the laser power level. The allowed power range is shown next to the control.
Fine Tune	Enables tuning the laser off grid up to 12 GHz. Change this value by typing a number in the text box or by dragging the slider. The sum of the text box and slider values is sent to the laser. Once the laser accepts the new value, that value is displayed after the '=' sign.
First Frequency	Shows the lowest frequency to which you can tune the laser. Readout only.
Last Frequency	Shows the highest frequency to which you can tune the laser. Readout only.
Channel 1	Settable when emission is off. This is the definition of Channel 1.
Grid Spacing	Sets the laser grid spacing. Settable (with 100 MHz resolution) when emission is off. 0.1, 0.05 or 0.01THz are typical choices. Use 0.01 THz if tuning to arbitrary (non-ITU-grid) frequencies. Using this grid plus Fine Tune, any frequency in the laser band is accessible.
Laser Electrical Power	Turns on or off electrical power to the laser module. This should normally be selected (checked). Unchecking this box turns off electrical power to the laser module. Only turn off electrical power to reset the laser to its power-on state, or to preserve laser lifetime if a particular laser is never used.
Connected To	Sets where this laser is connected. The control software must know if this laser is being used as the Reference for a coherent receiver. Select Reference if this laser is connected to the Reference (LO) input of a coherent receiver.

Channel setting within the ITLA grid gives the corresponding frequency (in THz) and wavelength (in nm). Power is set within the range allowed by the laser.

Use the Fine Tune slider bar to fine tune the laser, which typically works over a range of ±10 GHz from the center frequency of the channel selected.

The Modulator controls

When connected to an OM5510, the Modulator section of an instrument tab sets the optical modulator bias parameters. Which controls and fields are shown depends on whether Auto-Set is enabled or disabled (check box at the bottom of the tab).

The Auto-Set check box enables or disables the modulator automatic optical bias function settings. When the check box is selected, settings are controlled automatically based on the specified signal level and type. When Auto-Set is cleared, you can manually enter modulator settings.

Auto settings view (Auto-Set check box selected).

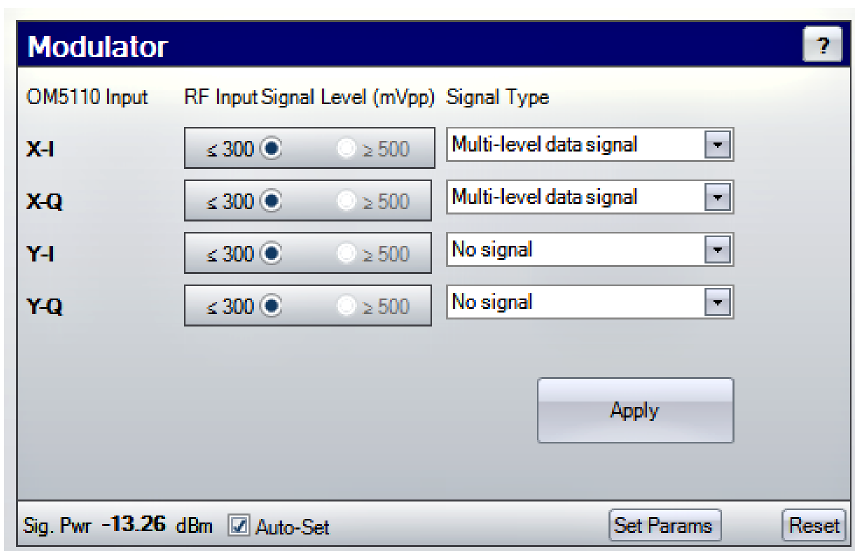


Table 3-2: Modulator controls (Auto-Set mode)

Control	Description
RF Input Signal Level (mVpp)	Set whether the input signal to each instrument input (X-I, X-Q, Y-I, and Y-Q) is less than or greater than the listed value. ***NOTE.*** *Signal level should be less than 300 mV_{pp} or greater than 500 mV_{pp}. Values between 300 mV_{pp} and 500 mV_{pp} require reducing the electrical amplifier gain or use of external attenuators to obtain a signal level between 100 mV_{pp} and 300 mV_{pp}.*
Signal Type	Sets the input signal type. Valid types are No Signal, Binary data signal, and Multi-level data signal.

Table 3-2: Modulator controls (Auto-Set mode) (cont.)

Control	Description
Apply	Send the settings to the instrument. When the wait circle disappears, your settings have been applied. The OM5110 retains these settings until they are changed. No settings are sent or retained by the OM5110 until you click the Apply button.
Sig. Pwr	(Readout only) The Modulated Output Signal power (abbreviated Sig. Pwr.) readout at the bottom of the Modulator control area. If the output is too high or too low, it may temporarily affect the controller circuits of the OM5110. In this case the power readout changes color and mouse-over text is available to indicate that optical bias and power readout may not be precise. There is no harm operating like this if the input optical power is within the specified range.
Set Params	Opens the Set Modulator Parameters dialog to set the Optimum Bias Voltage and Vpi Voltage parameters. ***NOTE.*** *It is particularly important to have a good estimate for the XP and YP quadrature phase settings. See the calibration section for details.*
Reset	Sets the optical bias control voltages to the default values. This is helpful whenever a major change is made to the system such as turning on the laser or input signals. Clicking Reset generally helps the system reach steady-state operation the fastest.

Manual settings view (Auto-Set check box cleared). The Manual Settings View provides the greatest degree of control flexibility, but is more complex than Automatic Settings View. Since each setting may take five seconds to be stored in an instrument, and possibly several minutes to reach steady state, it is best to use the Automatic Settings View where all the settings are established at once. The Manual Settings View is helpful when it is necessary to make fine adjustments to optimize a signal, or when it is desirable to impair the signal.

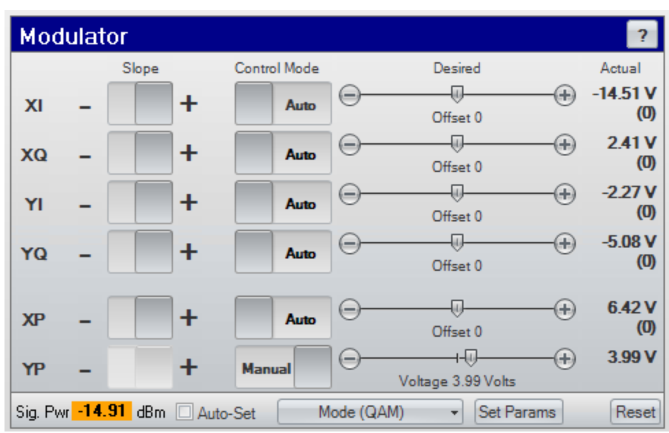


Table 3-3: Modulator controls (manual mode)

Control	Description
Slope	Usually **-** for > 500 mV_{pp} inputs and **+** for < 300 mV_{pp} inputs. The **-** causes lock at minimum attenuation and the **+** at maximum attenuation.
Control Mode	Auto to use automatic optical bias control based on feedback from the output optical signal. Manual to set the optical modulator bias voltage to a particular value.
Voltage/Offset	The slider control is used to set the desired voltage when in Manual mode or to set the Offset when in Auto mode. Offset is the amount to offset the bias from where it would normally be in Auto mode. The units are arbitrary and vary based on Optical Input power. The Offset must be tuned while observing the Modulated Optical Output signal on an appropriate optical signal analyzer to obtain the desired signal behavior.
Actual	This column shows the voltages at the optical modulator bias inputs. The value in parentheses is the actual Offset value.
Signal Mode	The optical bias controller behaves differently depending on the type of electrical signal input. Large-signal binary signals require 2-pol QPSK mode. QAM signals generally require QAM mode. Again it is best to use the Automatic Settings View which chooses the most appropriate Signal Mode automatically.
Set Modulator Parameters	The 6 modulator sections of the OM5110 modulator (X-I, X-Q, Y-I, YQ, XP, and YP) each have particular null voltages, where that section outputs minimum optical power, and Vpi voltages, which is the voltage difference between null and peak transmission. This type of information is needed by the OM5110 optical bias controller to properly control the modulator sections. The OM5110 is preprogrammed at the factory with the optimum bias and Vpi voltages. Optimum bias voltages are stored rather than null voltages to make them easier to set. The optimum bias voltages do change with time, and are different for different RF drive levels. It is not important for these values to be very precise. You should update your modulator parameters only if the OM5110 fails to obtain proper optical bias within a few minutes. Providing a better set of optimum bias voltages speeds the time to proper optical bias. The Vpi voltages do not change appreciably with time or temperature and may be left at their factory-set values.

To determine the optimum bias voltage values.

1. Connect the OM5110 to an analyzer, such as the OM4245, to analyze the signal quality of the OM5110. Connect the necessary signal inputs and turn on the laser source.
2. Use the **Modulator Auto-Set** view to set up the OM5110 for the required signal types and drive levels. Click **Apply**. Wait for this step to complete.

3. Deselect the **Auto-Set** box to see the Manual Settings view. Wait for the analyzer to report that the optical bias is correct.

4. If the optical bias does not meet your requirements, use the Manual Control Mode or the Offset function to correct the optical bias. This is easiest if the OM5110 is connected for single polarization IQ operation. That is, there should be proper drive signals connected to either XI and XQ or to YI and YQ. The X parameters are determined with XI and XQ driven, the Y parameters are determined with YI and YQ driven. It is important to drive both I and Q or the phase (XP or YP) will not be known.

   After connecting the XI and XQ signals, use Auto Control Mode for XI, XQ, YI, YQ, and manual control for XP and YP. Try several values for XP leaving YP alone, waiting each time for XI, XQ, YI, and YQ to auto bias. Once proper X constellation bias is achieved, record these values and then move the drive signals to YI and YQ and repeat the process.

   If the autobias does not work for several different XP voltage settings, verify that the signal levels are < 300 mVpp or > 500 mVpp and that the Auto Set panel was correspondingly configured and Applied.

5. Record the voltages shown on the Manual Settings view once the optical bias value meets your requirements.

6. Click **Set Params**. Enter the voltages shown in the Manual Settings view (step 5) as the Null Voltages in the Set Parameters dialog box.

NOTE. *If using Set Params results in worse values, click **Restore Initial Values** to reload the settings originally detected by the LRCP at first connection to the OM5110.*

7. Click **OK**.

8. To verify the Null Voltage values, change every segment to **Manual Control Mode** and click **Reset**. The voltages shown should match those found in step 5) to within 0.01 V.

9. Return to the Auto-Set view and click **Apply** to return to automatic control.

The Driver Amp controls

The Driver Amp control area of the LRCP software controls the behavior of the optical modulator RF Input electrical amplifier. This two-stage amplifier can work in both linear and nonlinear modes to enable both linear electrical-to-optical conversion and binary optical signal generation which is insensitive to the electrical input signal level.

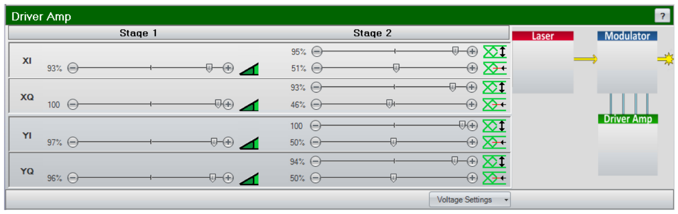


Table 3-4: Driver Amp controls

Control	Description
Stage 1	Sets the first stage of electrical amplification. You can adjust the gain of each Stage 1 amplifier. This should not be needed for most applications, but is helpful to balance the amplitude of I-Q signals when operating in the linear range (< 300 mVpp electrical input).
Stage 2	Sets the second stage of electrical amplification. When operating with >500 mVpp electrical input, you can adjust the crossing point and amplitude of the signal driving the optical modulator. These controls are not effective in the linear range (< 300 mVpp) and can be left at their default values.
Voltage Settings	**Save current voltages as power-on defaults**, which stores all of the current Driver Amp settings in the OM5110 as the new defaults. **Restore to factory defaults**, which loads the factory default values for the Driver Amp, overriding the current values. When the OM5110 is turned on and off by the rear-panel Primary power switch, or when it loses mains power, only the “power-on default settings,” and “factory defaults” are retained.

Each of the adjustments for linear gain, nonlinear crossing point, and nonlinear amplitude are indicated by a value in percent. This value is provided to help documentation of the amplifier settings. The control is not strictly proportional to this value, so these settings must be determined experimentally using the appropriate optical signal analyzer.

The block diagram image to the right of the Driver Amp is for reference to show the signal flow relationship of the Laser, Modulator, and Driver Amp controls. The diagram does not change or show any values while changing control parameters.

Appendices

Appendix A: OM5110 theory of operation

The OM5110 contains a dual-polarization IQ optical modulator capable of producing optical modulation at up to 46 Gbaud for binary modulation and 34 Gbaud for small-signal modulation. The optical modulator translates the RF input signals to the frequency of the Optical Input using dual nested Mach-Zehnder modulators to provide RF IQ modulation on two orthogonal polarizations at the Optical Output.

The linear two-stage amplifiers increase the RF input signals by 20 dB before going to the modulator. The amplifier gain reduces the input-referred Vpi voltages of the modulators to approximately 350 mV. Full-amplitude binary-phase-shift-keyed modulation (BPSK) is achieved with signals greater than 700 mVpp but no more than 1 Vpp is required for complete saturation.

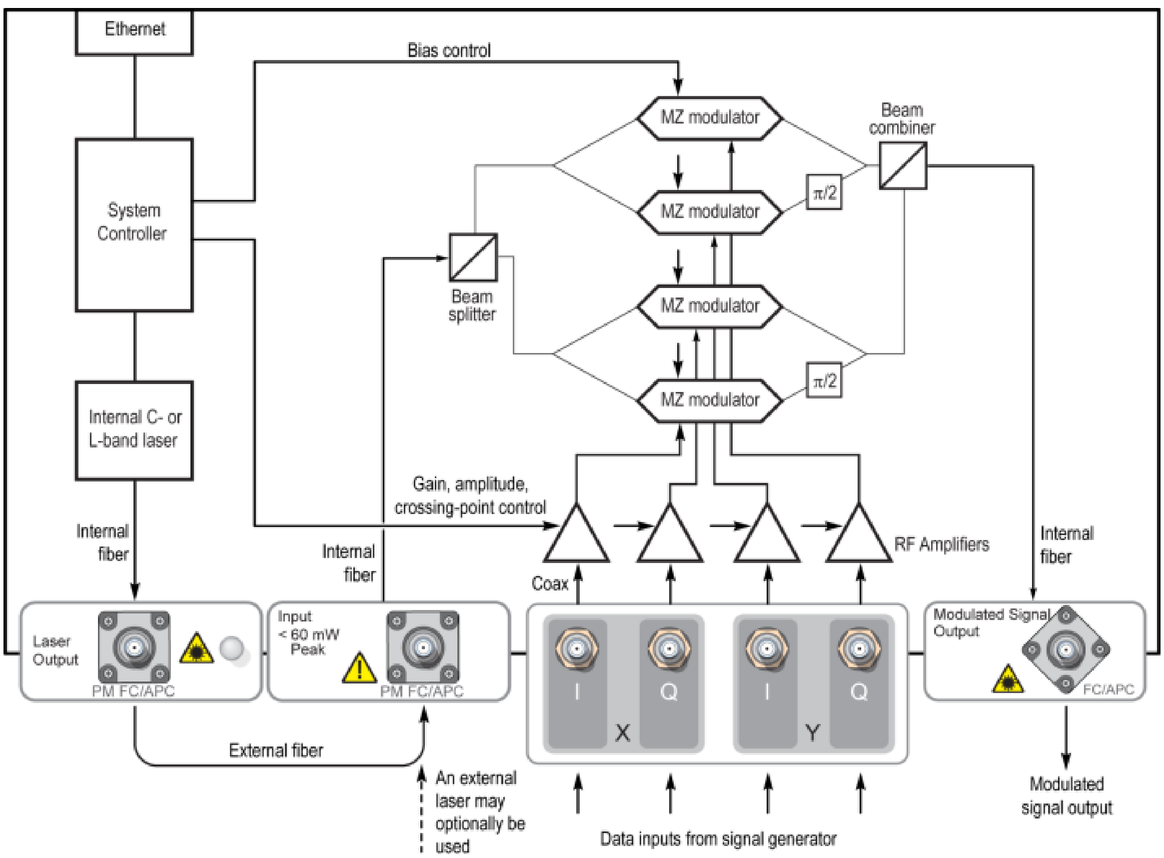


Figure A-1: OM5110 block diagram

For 2-polarization IQ modulation, 4 RF input signals are required. If the OM5110 is used for other modulation types, only the necessary number of inputs need be connected, for example, a single RF data signal is required for 1-pol BPSK. The LRCP software is used to configure the optical modulator bias controller for the modulation type. The optical bias controller makes sure that each of the data modulators and IQ-phase control modulators are biased at the proper points for IQ modulation. Each IQ RF-input pair should have minimum correlation to obtain the best result with automatic optical bias control. For example, putting exactly the same signal on I and Q will not necessarily result in the expected IQ modulation unless manual optical bias control is used.

There are two primary modes of operation for the OM5110: large-signal and small-signal. Large-signal modulation occurs from around 500 mVpp to where the amplifier is completely saturated at 1 Vpp. In this range both the amplifier and modulator have a nonlinear characteristic that provide higher quality modulation for binary signals. As the amplifier becomes saturated, use the LRCP software to adjust the amplitude and duty-cycle distortion (crossing-point) provided by the amplifier's second stage of gain. Factory settings provide 2Vpi modulation amplitude and 50% crossing point, but you can adjust these to accommodate different signal sources.

The small signal mode of operation is most often used for multi-level inputs used to create QAM signals. The amplifier remains linear up to 300 mVpp so the amplitude and duty cycle adjustments will have little effect. In this case, you adjust the first amplifier gain stage to equalize the small signal gains of the different channels. The factory settings typically provide 2% amplitude matching between input channels. Use this mode of operation to convey baseband multi-level IQ signals such as 16-QAM to optical frequencies.

The OM5110 is ideally suited for use with the AWG70001A and RFxpress application software for creating arbitrary signals that are precompensated for the response of the AWG and the OM5110 to provide the desired modulated optical waveform. See the Application Note provided with the OM5110 AWG file library for further information on using the OM5110 with the AWG70001A and RFxpress.

Appendix B: The LRCP automated test equipment (ATE) interface

LRCP has two types of Windows Communication Foundation (WCF) interface to allow control from a user application. Both types of interface provide full functionality and compatibility with simple interfaces such as MATLAB and client application programs.

The LRCP ATE interface

The Automated Test Equipment (ATE) interface exposes the LRCP functionality through a Windows Communication Foundation (WCF) service. As the LRCP is used with all OM4000 series or OM5110 instruments, its interface exposes more commands than those used by the instrument software user interface.

LRCP basic service interface

- The basic service for LRCP is available on port 9000.
- The basic service uses wsBasicHTTPBinding to be compatible with applications like MATLAB or Labview that only support the simpler binding.
- Exposes a subset of the advanced service commands.
- The basic service is referenced at the following URL:

  http://localhost:9000/LaserReceiverControlPanel/Laser_ReceiverServiceBasic/

LRCP advanced service interface

- The LRCP advanced service is available on port 9300.
- The LRCP advanced service uses a netTcpBinding (which is not available in MATLAB) and uses events to provide a time-efficient interface.
- The advanced service is referenced at the following URL:

  net.tcp://localhost:9300/LaserReceiverControlPanel/Laser_ReceiverService/
- Wrapper client DLLs (LRCPATEClient.DLL) have been installed into your documents folder under \TekApplications\ATE Support Files. Copy the

appropriate DLL (32-bit or 64-bit) to your ATE application's project folder and add a reference to the DLL to your project.

NOTE. *In previous releases it was required that the user edit their App.Config file for the client applications to supply URL information for accessing the advanced service. This is no longer necessary if user adds the reference to LRCPATEClient.DLL to their client ATE application. The detail of the WCF Service is wrapped in the DLL.*

- A new constructor was added to the LRCPATEClient class to allow the user to change the service's URL and/or port. This is useful for accessing the service remotely or for dealing with port conflicts.

  Syntax: LRCPATEClient(string url, int port)

  Coding example: (C#):

  LRCPATEClient myLRCPService = new LRCPATEClient("155.90.55.23", 9300);

NOTE. *The advanced service binding in earlier versions of OMA software was wsHTTPBinding, referenced at URL http://localhost:9000/LaserReceiverControlPanel/Laser_ReceiverService/. These are not available in the new OMA software. Make sure to use the new binding and URL when you update your ATE code to run with the new OMA software.*

NOTE. *MATLAB returns strings, not numeric values. To convert returned string values to numeric values, use the following: VarName = CmndName(obj); x = str2num(VarName). For example: LoFreq = GetLOFreq(obj); x = str2num(LoFreq).*

WARNING. *WCF services can turn lasers on and off. Verify that no one is physically working with lasers or fibre while running ATE applications. Power off the OM instruments to disable lasers.*

LRCP ATE service function list

The following are the available instrument commands (OM4000 series, OM2000 series, OM5110) for both the basic and advanced service interfaces, and show their functionality using the MATLAB syntax. Each command lists the instruments with which that command operates.

- **int AvailableLasers(classname);**
  Description: Returns the count of available lasers on the active controller.
  Controller Types: All
  Example: `AvailableLasers(obj);`
  Returns: ans = 2

- **bool Calibrate(classname);** (OM5110 only)
  Description: Performs an automatic modulator calibration to determine the optimal modulator parameters. These are the same parameters that are manually set using the user interface Set Params button, or the SetManualCalibration() function.

  Calibrate() is an automatic calibration that requires the modulator control mode (see GetActualModulatorMode()) be set to 2-pol QPSK and that >500 mV_{pp} binary signals are applied to all four inputs. Longer patterns are better (2^{31} PRBS is optimal). Each of the four input patterns should be different in some way: a different pattern, the same pattern delayed, or a different seed.

  The modulator must receive adequate input power levels to produce a signal power that is high enough to avoid a power level warning. If these conditions are not met the resulting calibration can result in unstable optical bias. See the section on Set Paramaters to restore these values to factory values.
  Controller Types: OM5110
  Example: `Calibrate(obj);`
  Returns: ans = true/false

- **bool Connect(classname);**
  Description: Connects to the active controller, starts controller running.
  Controller Types: All
  Example: `Connect(obj);`
  Returns: ans = true

- **bool Disconnect(classname);**
  Description: Disconnects from the active controller, takes offline.
  Controller Types: All
  Example: `Disconnect(obj);`
  Returns: ans = true

- **bool GetActualCavityLock(classname);**
  Description: Returns the actual cavity lock state for the active controller/laser. Locked = True.
  Controller Types: All
  Example: `GetActualCavityLock(obj);`
  Returns: ans = true

- **double GetActualChannel(classname);**

Description: Returns the actual channel number for the active controller/laser.
Controller Types: All
Example: `GetActualChannel(obj);`
Returns: ans = 1

- **double GetActualChannel1(classname);**
  Description: Returns the actual channel 1 frequency (in THz) for the active laser.
  Controller Types: All
  Example: `GetActualChannel1(obj);`
  Returns: ans = 191.5

- **controlmode GetActualControlMode(classname, enum_modulator);** (OM5110 only)
  Description: Returns the control mode for the modulator that is passed in the ‘modulator’ parameter.
  Controller Types: OM5110
  Example: `GetActualControlMode(obj, modulator.YQ);`
  Returns: ans = one of the following:
  `controlmode.automatic`
  `controlmode.manual`
  `controlmode.notset`

- **float GetActualCurrent(classname, string_voltageName);** (OM5110 only)
  Description: Returns current (mA) associated with the specified input voltage.
  Controller Types: OM5110
  Example: `GetActualCurrent(obj, 'YQ G1');`
  Returns: ans = 30

  The following are valid voltageName string values:

  'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **bool GetActualEmitting(classname);**
  Description: Returns the emission status of the active laser. Emitting = True.
  Controller Types: All
  Example: `GetActualEmitting(obj);`
  Returns: ans = true

- **byte GetActualFactoryDefault(classname, string_voltageName);** (OM5110 only)
  Description: Returns the factory default value for the specified voltage, in the range of 0 to 255.
  Controller Types: OM5110
  Example: `GetActualFactoryDefault(obj, 'XQ D2');`
  Returns: ans = 0

  The following are valid voltageName string values:

'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **short GetActualFineTuneFrequency(classname);**
  Description: Returns the actual fine tune frequency (in MHz) of the active laser.
  Controller Types: All
  Example: `GetActualFineTuneFrequency(obj);`
  Returns: ans = 0

- **double GetActualGridSpacing(classname);**
  Description: Returns the actual grid spacing (in THz) of the active laser.
  Controller Types: All
  Example: `GetActualGridSpacing(obj);`
  Returns: ans = 0.05

- **byte GetActualMaxPotSetting(classname, string_voltageName);** (OM5110 only)
  Description: Returns the maximum allowed step setting value for the specified voltage, in the range of 0 to 255
  Controller Types: OM5110
  Example: `GetActualMaxPotSetting(obj, 'XI G1');`
  Returns: ans = 100

  The following are valid voltageName string values:

  'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **float GetActualMaxVoltage(classname, string_voltageName);** (OM5110 only)
  Description: Returns the maximum voltage value for the specified voltage, in Volts.
  Controller Types: OM5110
  Example: `GetActualMaxVoltage(obj, 'YI D2');`
  Returns: ans = 10.2

  The following are valid voltageName string values:

  'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **byte GetActualMinPot(classname, string_voltageName);** (OM5110 only)
  Description: Returns the minimum allowed step setting value for the specified voltage, in the range of 0 to 255
  Controller Types: OM5110
  Example: `GetActualMinPot(obj, 'XI G2');`
  Returns: ans = 100

  The following are valid voltageName string values:

'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **float GetActualMinVoltage(classname, string_voltageName);** (OM5110 only)
  Description: Returns the minimum voltage value for the specified voltage, in Volts.
  Controller Types: OM5110
  Example: `GetActualMinVoltage(obj, 'YQ G1');`
  Returns: ans = 5.5

  The following are valid voltageName string values:

  'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **modulatormode GetActualModulatorMode(classname);** (OM5110 only)
  Description: Returns the modulator mode (notset, dpqpsk, qam, or arbirary)
  Controller Types: OM5110
  Example: `GetActualModulatorMode(obj);`
  Returns: ans = one of the following:
  modulatormode.notset
  modulatormode.dpqpsk
  modulatormode.qam
  modulatormode.arbitrary

- **double GetActualOffset(classname, enum_modulator);** (OM5110 only)
  Description: Returns the offset voltage adjustment value for the modulator that is in Automatic mode.
  Controller Types: OM5110
  Example: `GetActualOffset(obj);`
  Returns: ans = offset value

- **double GetActualPower(classname);**
  Description: Returns the actual power (in dBm) of the active laser.
  Controller Types: All
  Example: `GetActualPower(obj);`
  Returns: ans = 14.5

- **byte GetActualPowerOnDefault(classname, string_voltageName);** (OM5110 only)
  Description: Returns the power-on default setting for the specified voltage, in the range of 0 to 255.
  Controller Types: OM5110
  Example: `GetActualPowerOnDefault(obj, 'YQ G1');`
  Returns: ans = 150

  The following are valid voltageName string values:

  'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **slope GetActualSlope(classname, enum_modulator);** (OM5110 only)
  Description: Returns the slope setting (positive, negative, or notset) of the specified modulator (XI, XQ, YI, YQ).
  Controller Types: OM5110
  Example: `GetActualSlope(obj, modulator.XQ);`
  Returns: ans = one of the following:
  slope.notset
  slope.positive
  slope.negative

- **byte GetActualStep(classname, string_voltageName);** (OM5110 only)
  Description: Returns the step setting for the specified voltage, in the range of 0 to 255.
  Controller Types: OM5110
  Example: `GetActualStep(obj, 'YQ G1');`
  Returns: ans = 10

  The following are valid voltageName string values:

  'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **double GetActualVoltage(classname, enum_modulator);** (OM5110 only)
  Description: Returns the voltage setting of the specified modulator (XI, XQ, YI, YQ), in Volts.
  Controller Types: OM5110
  Example: `GetActualVoltage(obj, modulator.YQ);`
  Returns: ans = setting in Volts

- **double GetCalculatedFrequency(classname, enum_laserUsageType);**
  Description: Searches all of the connected controllers for the first laser of the specified laser usage type and returns the calculated frequency (in THz).
  Valid laserUsageType values are: unused, signalx, signaly, signalxy, reference.
  Controller Types: All
  Example: `GetCalculatedFrequency(obj, reference);`
  Returns: ans = 191.5

- **string[] GetControllers(classname);**
  Description: Returns a list (array) of controller devices (strings) that are being controlled by the serving application.
  Controller Types: All
  Example: `Controllers = GetControllers(obj);`
  Returns:
  'OM5110:Prototype1'
  'OM2210:8180123'
  'OM4006:6300121'

- **double GetFirstFrequency(classname);**
  Description: Returns the first frequency (in THz) of the active laser.
  Controller Types: All
  Example: `GetFirstFrequency(obj);`

Returns: ans = 191.5

- **bool GetInterlock(classname);**
  Description: Returns the current interlock state of the active controller. The normal, working state is TRUE. If the interlock is disconnected from the back of the instrument or if the instrument is powered off, this function returns FALSE.
  Controller Types: All
  Example: `GetInterlock(obj);`
  Returns: ans = true

- **string GetIP(classname);**
  Description: Returns the IP Address (as a string) for the active controller.
  Controller Types: All
  Example: `Address = GetIP(obj);`
  Returns: Address = '172.17.200.114'

- **double GetLastFrequency(classname);**
  Description: Returns the last frequency (in THz) of the active laser.
  Controller Types: All
  Example: `GetLastFrequency(obj);`
  Returns: ans = 196.25

- **double GetOpticalPower(classname);** (OM5110 only)
  Description: Returns the optical power setting, in dBm.
  Controller Types: OM5110
  Example: `GetOpticalPower(obj);`
  Returns: ans = 12.4

- **double GetOpticalPowerAdjustment(classname);** (OM5110 only)
  Description: Returns the optical power adjustment setting, in dBm.
  Controller Types: OM5110
  Example: `GetOpticalPowerAdjustment(obj);`
  Returns: ans = 10.4

- **float GetPhotoCurrent(classname);**
  Description: Returns the photocurrent (in mA) of the receiver in the active controller.
  Controller Types: OM4245, OM4225, OM4106D, OM4006D
  Example: `GetPhotoCurrent(obj);`
  Returns: ans = 11.034

- **Dictionary<string, float> GetPhotoDiodeCurrents(classname);**
  Description: Returns the photo diode currents (mA) of the individual channels of the active controller.
  Controller Types: OM4245, OM4225
  Example: `GetPhotoDiodeCurrents(obj);`
  Returns: The format depends on the environment. In a .NET application the dictionary contains the keys XQP, XQN, XIP, XIN, YQP, YQN, YIP, and YIN. Access the current values as follows: < return_variable_name> ["RXIP"]

- **enum_polarization GetPolarization(classname);** (OM2210 only)
  Description: Returns the polarization state.
  Valid Polarization states: filter1, filter2, unknown, hardwarefailed.
  Controller Types: OM2210
  Example: `GetPolarization(obj);`
  Returns: ans = filter2

- **double GetPowerByLaserUsageType(classname, enum_laserusagetype);** (OM2210 only)
  Description: Returns the current power reading for the specified laser.
  Valid laser usage types: unused, signalx, signaly, signalxy, reference.
  Controller Types: All
  Example: `GetPowerByLaserUsageType(obj, laserusagetype.signalx)`
  Returns: ans = 13.5

- **bool InitializePolarization(classname);** (OM2210 only)
  Description: Sets the initial polarization state. Returning True = Successful.
  Controller Types: OM2210
  Example: `InitializePolarization(obj);`
  Returns: ans = true

- **bool IsActiveControllerChosen5110Y(classname);** (OM5110 only)
  Description: Returns true if the active controller is an OM5110; returns false if it is not an OM5110.
  Controller Types: OM5110
  Example: `IsActiveControllerChosen5110Y(obj);`
  Returns: ans = true/false

  The ATE methods for the modulator use the following enumerations:

  enum_modulator (YQ, YI, YP, XQ, XI, XP)
  modulatormode (noset, dpqpsk, qam, arbitrary)
  slope (notset, negative, positive)
  controlmode (notset, automatic, manual)

- **bool Reset(classname);** (OM5110 only)
  Description: Resets the modulator.
  Controller Types: OM5110
  Example: `Reset(obj);`
  Returns: ans = true/false

- **bool SetActiveControllerByIPAddress(classname, string_ipAddress);**
  Description: Sets the active controller by IP Address. True = Successful.
  Controller Types: All
  Example: `SetActiveControllerByIPAddress(obj, '172.17.200.112');`
  Returns: ans = true
  Returns (after GetIP(obj)): ans = '172.17.200.112'

- **bool SetActiveControllerByName(classname, string_activeController);**

Description: Sets the active controller by name. True = Successful.
Controller Types: All
Example: `SetActiveControllerByName(obj, 'OM4106:6300121');`
Returns: ans = true

- **bool SetActiveLaser(classname, byte_activeLaser);**
  Description: Sets the active laser. Returning True = Successful.
  Controller Types: All
  Example: `SetActiveLaser(obj, 2);`
  Returns: ans = true

- **bool SetControllerAndLaserByUsageType(classname, enum_laserUsageType);**
  Description: Searches the "running" controllers for a laser matching the requested usage type (usually reference) and selects that controller and laser.
  Controller Types: OM4245, OM4225, OM4006D, OM4106D, OM2210, OM2012
  Valid laserUsageType values are: unused, signalx, signaly, signalxy, reference.
  Returns: ans = True - First laser for the specified usage type was selected
  False - No lasers found on running controllers for the specified usage type

- **bool SetDesiredCavityLock(classname, bool_desiredCavityLock);**
  Description: Sets the desired cavity lock state for the active laser. 1 (True) = Locked. Returning True = Successful.
  Controller Types: All
  Example: `SetDesiredCavityLock(obj, 1);`
  Returns: ans = true

- **bool SetDesiredChannel(classname, int_desiredChannel, bool_waitUntilComplete);**
  Description: Sets the channel number for the active laser. Returning True = Successful.
  Controller Types: All
  Example: `SetDesiredChannel(obj, 45, true);`
  Returns: ans = true

- **bool SetDesiredChannel1(classname, double_desiredChannel1);**
  Description: Sets the channel 1 frequency (in THz) for the active laser. Can only be set if the active laser is NOT emitting. Returning True = Successful.
  Controller Types: All
  Example: `SetDesiredChannel1(obj, 192.5);`
  Returns: ans = true

- **bool SetDesiredControlMode(classname, enum_modulator, enum_controlmode);** (OM5110 only)
  Description: Sets the control mode of the specified modulator. Returns true if control mode was set, or false if mode was not set.
  Controller Types: OM5110
  Example: `SetDesiredControlMode(obj, modulator.XI, controlmode.automatic);`

Returns: ans = true/false

- **bool SetDesiredEmittingOff(classname);**
  Description: Sets the Active Laser to Off (not emitting). Returning True = Successful.
  Controller Types: All
  Example: `SetDesiredEmittingOff(obj);`
  Returns: ans = true

- **bool SetDesiredEmittingOn(classname);**
  Description: Sets the Active Laser to emitting. Returning True = Successful.
  Controller Types: All
  Example: `SetDesiredEmittingOn(obj);`
  Returns: ans = true

- **bool SetDesiredFineTuneFrequency(classname, short_desiredFineTuneFrequency, bool_waitUntilComplete);**
  Description: Sets the desired fine tune frequency (in MHz) of the active laser.
  Controller Types: All
  Example: `SetDesiredFineTuneFrequency(obj, 300, true);`
  Returns: ans = true

- **void SetDesiredFrequency(double_desiredFrequency);**
  Description: Sets the channel and fine tune frequency of the selected laser to the specified frequency.
  Controller Types: OM2210, OM2012

- **bool SetDesiredGridSpacing(classname, double_desiredGridSpacing);**
  Description: Sets the desired grid spacing (in THz) of the active laser. Can only be set if the active laser is NOT emitting. Returning true = Successful.
  Controller Types: All
  Example: `SetDesiredGridSpacing(obj, 0.05,0);`
  Returns: ans = true

- **bool SetDesiredModulatorMode(classname, enum_modulatorMode);** (OM5110 only)
  Description: Sets the specified modulator mode. Returns true if mode was set, or false if mode was not set.
  Controller Types: OM5110
  Example: `SetDesiredModulatorMode (obj, modulatormode.qam)`
  Returns: ans = true/false

- **bool SetDesiredOffset(classname, enum_modulator, double_offset);** (OM5110 only)
  Description: Sets the offset for the specified modulator. Returns true if offset was set, or false if offset was not set.

  Valid offset value range depends on the modulator that is being set:
  Q and I modulators range from -2000 to 2000.
  P modulators range from -6000 to 6000.
  Controller Types: OM5110

Example: `SetDesiredOffset(obj, modulator.XI, 2000);`
Returns: ans = true/false

- **bool SetDesiredPower(classname, double_desiredPower, bool_waitUntilFinished);**
  Description: Sets the desired power (in dBm) of the active laser. For WaitUntilFinished, 0 (False) = don't wait. Returning True = Successful.
  Controller Types: All
  Example: `SetDesiredPower(obj, 13, 0);`
  Returns: ans = true

- **bool SetDesiredPowerOnDefault(classname, string_voltageName, byte_step);** (OM5110 only)
  Description: Sets the default power-on value for the specified voltage. Returns true if value was set, or false if value was not set.
  Controller Types: OM5110
  Example: `SetDesiredPowerOnDefault(obj, 'YI G1', 125);`
  Returns: ans = true/false

  The following are valid voltageName string values:

  'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **bool SetDesiredSlope(classname, enum_modulator, enum_slope);** (OM5110 only)
  Description: Sets the slope for the specified modulator to positive or negative. Returns true if slope was set, or false if slope was not set.
  Controller Types: OM5110
  Example: `SetDesiredSlope(obj, modulator.XI, slope.positive);`
  Returns: ans = true/false

- **bool SetDesiredVoltage(classname, enum_modulator, double_voltage);** (OM5110 only)
  Description: Sets the voltage of the specified modulator. Returns true if voltage was set, or false if voltage was not set.
  Controller Types: OM5110
  Example: `SetDesiredVoltage(obj, modulator.XI, 10.2);`
  Returns: ans = true/false

- **bool SetDesireStep(classname, string_voltageName, byte_step);** (OM5110 only)
  Description: Sets the step for the specified voltage. Returns true if value was set, or false if value was not set.
  Controller Types: OM5110
  Example: `SetDesireStep(obj, 'YQ G1', 135);`
  Returns: ans = true/false

  The following are valid voltageName string values:

'YQ G1', 'YI G1', 'YQ G2', 'YI G2', 'YQ D2', 'YI D2', 'XI D2', 'XQ D2', 'XI G2', 'XQ G2', 'XI G1', 'XQ G1'

- **bool SetManualCalibration(classname);** (OM5110 only)
  Description: This is the same as the SetParams button in the user interface. It sends a new set of null voltages and Vpi voltages to the OM5110 for use by the optical bias controller. Returns true if the calibration was successfully started, and false if the calibration was not started.
  Controller Types: OM5110
  Example: Calibrate(obj, float, float, float, float, float, float, float, float, float, float, float, float);
  Returns: ans = true/false

- **bool SetOpticalPowerAdjustment(classname, double_power);** (OM5110 only)
  Description: Sets the optical power adjustment, in dBm. Returns true if power adjustment value was set, false if the value was not set.
  Controller Types: OM5110
  Example: `SetOpticalPowerAdjustment(obj, 10.1);`
  Returns: ans = true/false

- **bool SetPolarizationIn(classname);** (OM2210 only)
  Description: Puts both polarization filters in. Returning True = Successful.
  Controller Types: OM2210
  Example: `SetPolarizationIn(obj);`
  Returns: ans = true

- **bool SetPolarizationOut(classname);** (OM2210 only)
  Description: Puts both polarization filters out. Returning True = Successful.
  Controller Types: OM2210
  Example: `SetPolarizationOut(obj);`
  Returns: ans = true

- **bool SetReceiverOff(classname);**
  Description: Turns the receiver off in the active controller. Returning True = Successful.

CAUTION. *Make sure that laser power is off before running this command, otherwise the photoreceiver could be damaged.*

NOTE. *To turn off lasers, click the LRCP software Laser Emission button to **Off**, or push the front-panel **Power** button to power off the instrument.*

  Controller Types: OM4245, OM4225, OM4106D, OM4006D
  Example: `SetReceiverOff(obj);`
  Returns: ans = true

- **bool SetReceiverOn(classname);**

Description: Turns the receiver on in the active controller. Returning True = Successful.
Controller Types: OM4245, OM4225, OM4106D, OM4006D
Example: `SetReceiverOn(obj);`
Returns: ans = true

- **void TogglePolarization(classname);** (OM2210 only)
  Description: Toggles the polarization state by moving both filters to the opposite position.
  Controller Types: OM2210
  Example: `TogglePolarization(obj);`

LRCP control in MATLAB

The Laser/Receiver Control Panel tab communicates with other programs using port 9000 on the computer running the OMA software.

NOTE. *Make sure that the LRCP tab is open before using this interface.*

To initialize the interface, open a MATLAB application on the desktop before starting the OMA software. Then enter the following commands in the MATLAB desktop command window:

```
url = 'http://localhost:9000/LaserReceiverControlPanel/
  Laser_ReceiverServiceBasic/?wsdl';

createClassFromWsdl(url);

obj = Laser_ReceiverServiceBasic;
```

Where:

- ‘url= ...‘ specifies the URL or path to a WSDL application programming interface (API) that defines the web service methods, arguments, and transactions for the Laser/Receiver Control Panel.
- ‘createClassFrom...’ creates the new class based on the specified API and builds a series of M-Files for accessing the Laser/Receiver Control Panel service.
- ‘obj=Laser_Receiver...’ instantiates the object class name and opens a connection to the service.

These commands only need to be run whenever the service interface (available methods) changes.

To display an up-to-date listing of methods for the initialized service, enter the following in the MATLAB desktop command window:

```
methods(obj);
```

Matlab returns a list of available functions for the initialized service. The list may also show new functions that have been added (and may not be documented in this manual). To show information on a specific function in the list:

1. Click a function name in the list to position the cursor in that function.
2. Right-click and select **Help on Selection**. MATLAB displays the function information.

ATE functionality in MATLAB

MATLAB supports a limited subset of services, namely the Basic service. This section describes how to create and address the functions from MATLAB.

LRCP control

The Laser/Receiver Control Panel communicates with other programs using port 9000 on the computer running the Control Panel software. MATLAB 2009a and later versions have a built-in capability that makes control from MATLAB easy if you are running the February 2010 or later release of the Laser/Receiver Control Panel.

NOTE. *Make sure that the LRCP is running before using this interface.*

Initialize the interface in the MATLAB desktop command window with the following commands:

```
url = 'http://localhost:9000/LaserReceiverControlPanel/
Laser_ReceiverServiceBasic/?wsdl';
```

```
createClassFromWsdl(url);
```

```
obj = Laser_ReceiverServiceBasic;
```

Where:

The first specifies the URL or path to a WSDL application programming interface (API) that defines the web service methods, arguments, and transactions for the LRCP.

The second creates the new class based upon that API and builds a series of M-Files for accessing the Laser/Receiver Control Panel service.

The third instantiates the object class name and opens a connection to the service.

These commands only need to be run anytime the service interface (available methods) changes.

To get an up-to-date listing of methods for the service, type the following:

```
methods(obj)
```

MATLAB should return the same functions and any new functions that have been added. These functions are self-documented when they are generated. By enabling the MATLAB help window, you can find out the function's parameters by typing the function name followed by a "(" and waiting for the help to display. The LRCP methods which return numerical values will return the numbers as strings in MATLAB. These strings can be turned back into numbers using the str2num function in MATLAB.

Index

Index

T

V

W