

# SWITCH

## 1200 SERIES GRID OPTICAL SWITCH

PXIE USER MANUAL



Copyright © 2021 Quantifi Photonics Ltd. 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 Quantifi Photonics Ltd. (Quantifi Photonics).

Information provided by Quantifi Photonics is believed to be accurate and reliable.

However, no responsibility is assumed by Quantifi Photonics 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 Quantifi Photonics.

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

### **Trademarks**

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

LabVIEW™, MATLAB® and Python® are trademarks of National Instruments, The MathWorks Inc., and the Python Software Foundation, respectively. Neither Quantifi Photonics nor any software programs or other goods or services offered by Quantifi Photonics are affiliated with, endorsed by, or sponsored by National Instruments, The MathWorks Inc., and the Python Software Foundation.

### **Units of Measurement**

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

### **EU Authorized Representative**

Certification Company

Veluwezoom 42

1327 AH ALMERE

The Netherlands

+31 (0)36 202 40 37

[info@certification-company.com](mailto:info@certification-company.com)

# Contents

<b>1</b>	<b>Conventions.....</b>	<b>5</b>
<b>2</b>	<b>Safety information .....</b>	<b>6</b>
2.1	Optical laser radiation precautions.....	6
2.2	Electrostatic discharge precautions.....	6
2.3	Electromagnetic compatibility.....	7
<b>3</b>	<b>Connecting optical fibers .....</b>	<b>8</b>
3.1	Cleaning and connecting optical fibers.....	8
<b>4</b>	<b>Introducing the Switch 1200 PXle Series.....</b>	<b>9</b>
4.1	Switch 1200 PXle Series overview & features.....	9
4.2	Status LEDs.....	10
<b>5</b>	<b>Handling the Switch 1200 PXle Series.....</b>	<b>11</b>
5.1	Switch 1200 PXle Series installation.....	11
5.2	Switch 1200 PXle Series uninstallation .....	12
<b>6</b>	<b>Software installation for Switch 1200 PXle Series.....</b>	<b>13</b>
6.1	Cohesion Installer information for PXle modules.....	13
6.1.1	Installation overview .....	13
6.1.2	Installation process.....	13
6.2	Quantifi Photonics PXle system utility applications.....	16
6.2.1	Cohesion Manager utility.....	16
6.2.2	Cohesion Firmware Updater utility .....	17
<b>7</b>	<b>CohesionUI application.....</b>	<b>19</b>
7.1	Accessing CohesionUI.....	19
7.2	Home page.....	19
7.3	Modules list.....	20
7.4	Set and actual values.....	20
7.5	Settings page.....	21
7.5.1	System controls for PXle modules.....	21
7.6	SCPI Command Console.....	23
7.7	Info panel.....	25
<b>8</b>	<b>Switch 1200 PXle Series control with CohesionUI.....</b>	<b>27</b>

<b>9</b>	<b>Programming guide.....</b>	<b>30</b>
9.1	Programming conventions .....	30
9.1.1	Index addressing of modules (slot, source) and units (channel).....	30
9.2	Status and event registers .....	31
9.2.1	Standard Event Status Register.....	31
9.2.2	Standard Event Status Enable Register (Mask) .....	31
9.2.3	Status Byte Register .....	31
9.2.4	Service Request Enable Register (Mask) .....	31
9.2.5	Status and event registers diagram.....	32
9.3	Common system command summary.....	33
9.4	Common system command descriptions.....	33
9.5	Specific command summary.....	35
9.6	Specific command descriptions.....	35
9.6.1	Slot commands .....	35
9.6.2	Configuration commands.....	36
9.7	Switch PXle Multi Chassis mode operation .....	38
9.7.1	NI-MAX application Multi Chassis mode.....	38
9.7.2	SCPI Multi Chassis commands.....	38
<b>10</b>	<b>Example: Control of the Switch 1200 PXle Series.....</b>	<b>40</b>
10.1	NI-MAX application.....	41
10.2	NI-VISA application.....	42
10.3	Python® 2.7 code example.....	43
10.4	MATLAB® code example .....	44
<b>11</b>	<b>Maintenance .....</b>	<b>45</b>
11.1	Annual calibration schedule.....	45
<b>12</b>	<b>Technical support.....</b>	<b>46</b>
12.1	Contacting the Technical Support Group.....	46
12.2	Transportation.....	46
<b>13</b>	<b>Warranty.....</b>	<b>47</b>
13.1	General information.....	47
13.2	Liability .....	47
13.3	Exclusions.....	47
13.4	Certification.....	47
13.5	Service and repairs.....	47

## 1 Conventions

---

Before using the instruments described in this document, take note of the following conventions:

### **WARNING**

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

### **CAUTION**

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

### **CAUTION**

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

### **IMPORTANT**

Refers to information about this module that you should not overlook.

### **NOTE**

Indicates some information that requires your attention or some extra information for the current topic.

## 2 Safety information

Before using the Switch 1200 PXle Series, ensure that the following safety information has been read and understood.

### 2.1 Optical laser radiation precautions

#### **WARNING**

Do not install or terminate fibers while the light source is active. Care must be taken to ensure that the module has been turned OFF before inspecting the end face(s) of the instrument, or any optical patch cords connected to this module. Never look directly into a live fiber; ensure that your eyes are protected at all times.

#### **CAUTION**

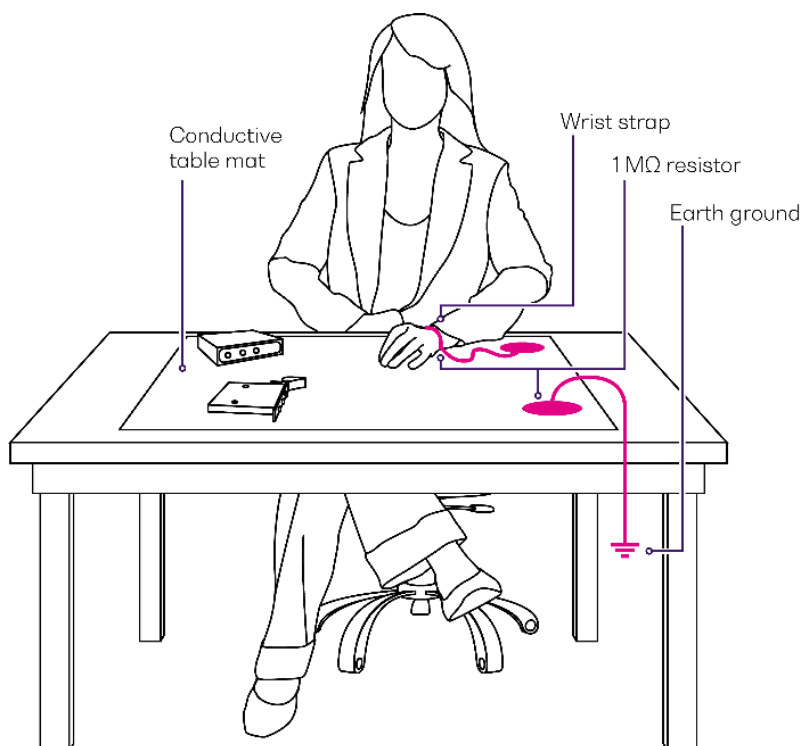
The use of controls, adjustments, and procedures other than those specified herein may result in exposure to hazardous situations involving optical radiation.

### 2.2 Electrostatic discharge precautions

#### **CAUTION**


The Switch PXle is sensitive to electrostatic discharge (ESD). Store the unused module in the original protective electrostatic packaging that the Switch PXle was shipped in.

Ensure that a wrist strap and grounding table mat is used when unpacking or handling the Switch 1200 PXle Series. Proper grounding and ESD management practices should always be followed to ensure that no ESD damage is caused to the Switch 1200 PXle Series.



## 2.3 Electromagnetic compatibility

### CAUTION

- For electromagnetic compatibility, this module is a **Class A** product. It is intended for use in an industrial environment. There may be potential difficulties in ensuring electromagnetic compatibility in other environments, due to conducted as well as radiated disturbances.
- Wherever the  symbol is printed on the unit, refer to the instructions provided in the device documentation for related safety information. Ensure that the required conditions are met and understood before using the module.

---

## 3 Connecting optical fibers

---

### CAUTION

To ensure maximum power and to avoid erroneous readings always inspect fiber end faces. Make sure they are cleaned as detailed below before inserting into any port. **Quantifi Photonics is not responsible for damage or errors caused by bad fiber cleaning or handling.**

### IMPORTANT

The type of optical connectors on the Switch 1200 PXle Series can be found printed on the front plate of the module. **Joining mismatched connectors will damage the ferrules and fibre faces.**

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

Quantifi Photonics uses high quality connectors in compliance with EIA-455-21A standards.

### 3.1 Cleaning and connecting optical fibers

**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 connect it to the desired port.
2. If the fiber is dirty, clean it as detailed below.
3. Gently wipe the fiber end with a lint-free swab dipped in isopropyl alcohol.
4. Use compressed air to dry completely.
5. Visually inspect the fiber end to ensure its cleanliness.
6. Carefully align the connector and port to prevent the fiber end from touching the outside of the port or rubbing against other surfaces. If the connector features a key, ensure that it is correctly mated into the corresponding notch of the port bulkhead.
7. Push the connector in so that the fiber-optic cable is firmly in place, thus ensuring adequate contact. If your connector features a screw sleeve, tighten the connector enough to firmly maintain the fiber in place. **Do not over tighten, as this will damage the fiber and the port bulkhead.**

### NOTE

If your fiber-optic cable is not properly aligned and/or connected, you will notice large signal loss and reflection.



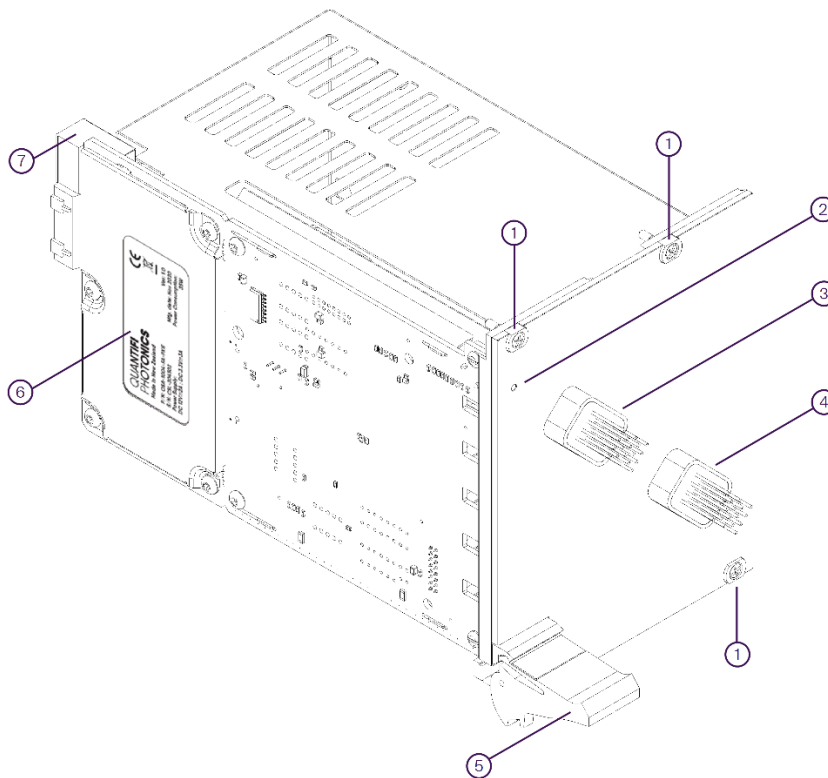
## 4 Introducing the Switch 1200 PXle Series

The Switch 1200 PXle Series module is an efficient solution for test procedure automation designed to be used on the National Instruments PXle platform. The MEMS based optical switch ensures repeatable and low-loss switching to various test set-ups utilizing the flexibility of the NxN connection setup.

### **⚠ IMPORTANT**

When plugging optical fibers into the ports or inspecting fibers, ensure that the optical source is OFF to prevent user injury.

### 4.1 Switch 1200 PXle Series overview & features



- 1 Fastening screws
- 2 Routing indicator LED
- 3 Breakout A optical ports
- 4 Breakout B optical ports
- 5 Fastening clip
- 6 Module information
- 7 PXle headers

**NOTE**

Number of breakout A and B routing ports will depend on the selected model. Attention must be paid to the port layout of the SwitchPXle, as the routing commands in the programming guide section are taken with reference to the breakout A ports.

## 4.2 Status LEDs

The status LEDs are used to denote the operation state of the Switch PXle module, and will indicate the state the switch is in.

- **Blinking yellow** - Indicates initialization on startup.
- **Solid green** - Indicates successful initialization and communication.
- **Solid red** - Indicates a switch error. Most likely and initialization error at startup.

## 5 Handling the Switch 1200 PXle Series

### CAUTION

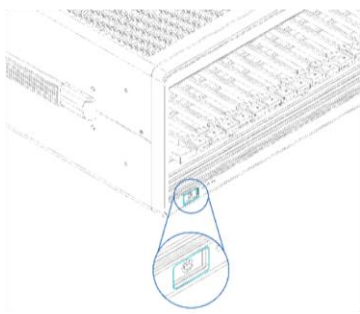
- Do not remove the Switch 1200 PXle Series module from the antistatic packaging until instructed during the following installation procedure.
- The Switch PXle is sensitive to ESD. Ensure you are wearing a grounded wrist strap at all times when handling the Switch PXle to prevent damage from electrostatic discharge.
- Take care not to handle the optical connectors on the module, as once they are exposed to skin contact this may leave corrosive residue and damage the connector.

### 5.1 Switch 1200 PXle Series installation

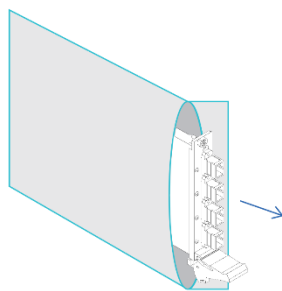
#### WARNING

DO NOT attempt to remove or adjust any component of the PXle chassis while the power is on. Ensure the chassis is powered OFF, and that the correct handling procedure detailed herein is followed when removing or installing any modules.

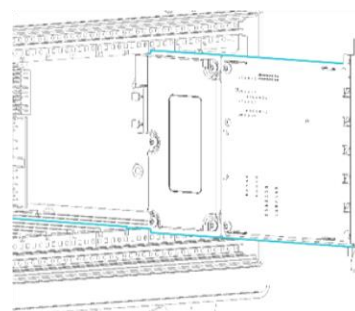
**STEP 1:** Power OFF the Chassis



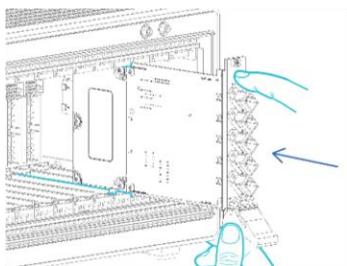
**STEP 2:** Remove the module from the antistatic bag. Retain bag



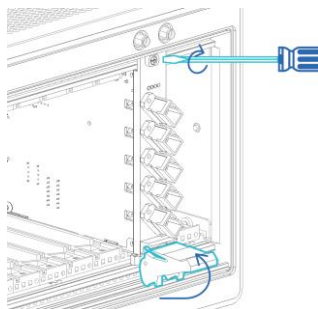
**STEP 3:** Align module with slot guide rails



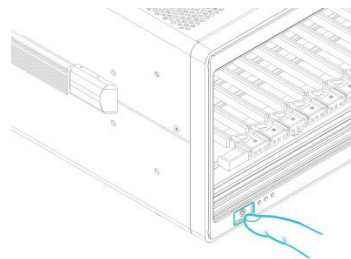
**STEP 4:** Push module into slot until resistance is felt from the backplane connection



**STEP 5:** Engage the fastening clip. Secure all fastening screws



**STEP 6:** Power ON the chassis

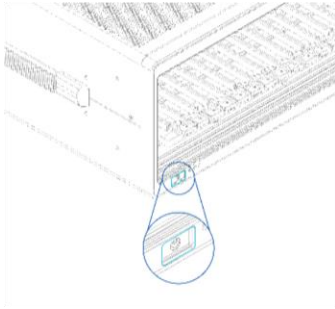


### IMPORTANT

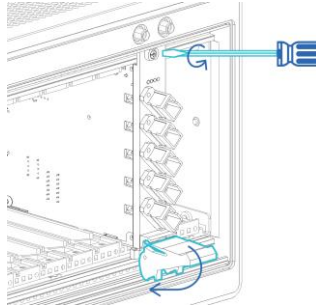
After powering on the PXle chassis, please wait at least **2 minutes** before attempting to communicate with the module. This will allow the chassis enough time to finish boot procedures and initialize the communication server.

## 5.2 Switch 1200 PXle Series uninstallation

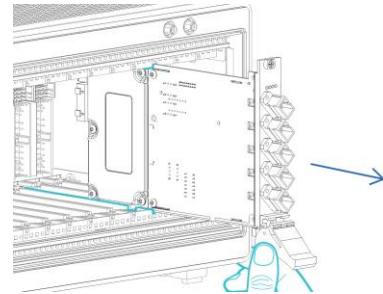
**STEP 1:** Power OFF the chassis



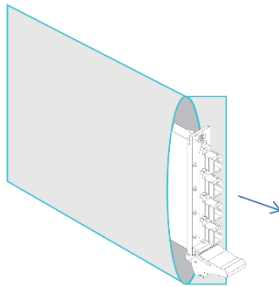
**STEP 2:** Unsecure the fastening screws and fastening clip



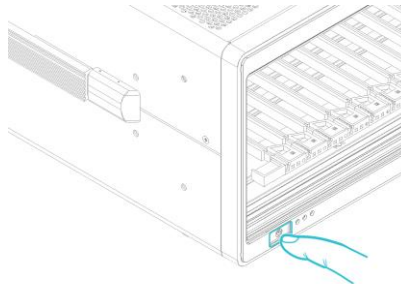
**STEP 3:** Pull out the module.  
USE THE FASTENING CLIP TO PULL. DO NOT PULL ON THE CONNECTORS



**STEP 4:** Store module in antistatic bag



**STEP 5:** Power ON the chassis



## 6 Software installation for Switch 1200 PXle Series

### 6.1 Cohesion Installer information for PXle modules

#### **⚠ IMPORTANT**

The software must be installed on the PXle controller for the PXle Chassis in which the Quantifi Photonics modules will be installed, or the controller PC in the case of a MXI setup.

**Minimum System Requirements:** 64-bit OS, Windows 7 or above.

**Recommended System Requirements:** 64-bit Windows 10.

The **Cohesion Installer** is a **single installation package** that contains all the required drivers and software, to support and control Quantifi Photonics modules on the PXle Platform.

#### 6.1.1 Installation overview

For the PXle controller to communicate with the Switch PXle module installed in the chassis, software and driver installations are necessary. This software is contained in the **Cohesion Installer** single installer package.

- **CohesionDriver:** Drivers for Quantifi Photonics PXle Modules
- **CohesionSCPI:** VXI11 compliant server for remote SCPI communication
- **CohesionUI:** Web-based Graphical User Interface

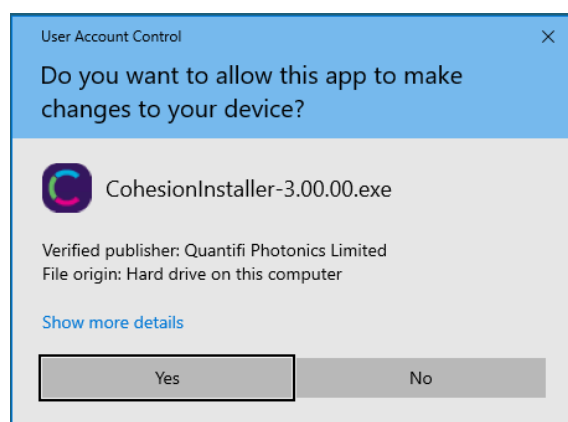
#### **⚠ IMPORTANT**

It is recommended that you **save all work and close any open programs** before attempting to install the required software packages above.

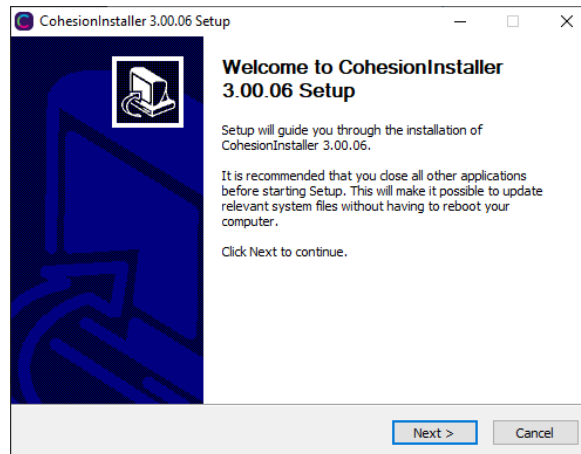
#### 6.1.2 Installation process

**Windows 10 64-bit:**

1. Locate and run the installer **CohesionInstaller<version\_number>.exe** from the provided USB media device (or download from the Quantifi Photonics [website](#)).

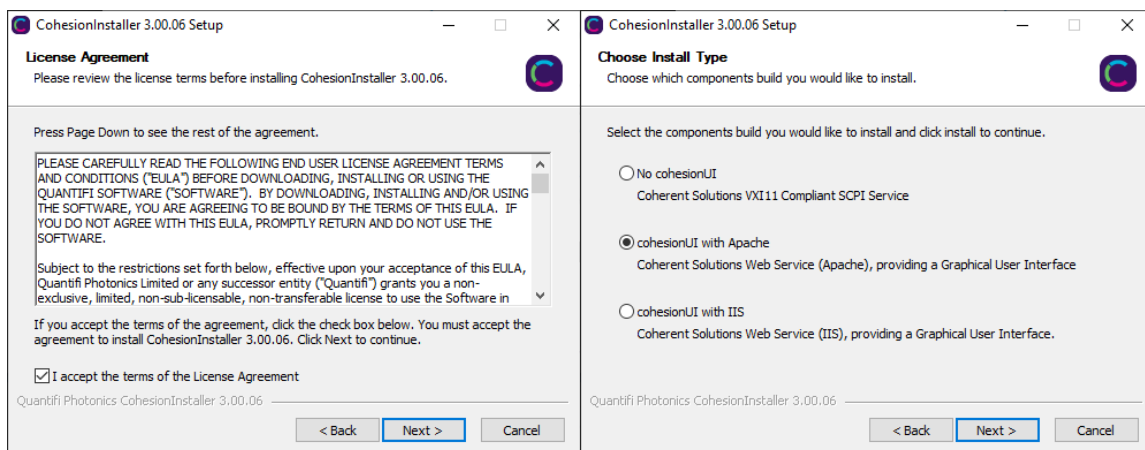


2. Follow the on-screen installation prompts.

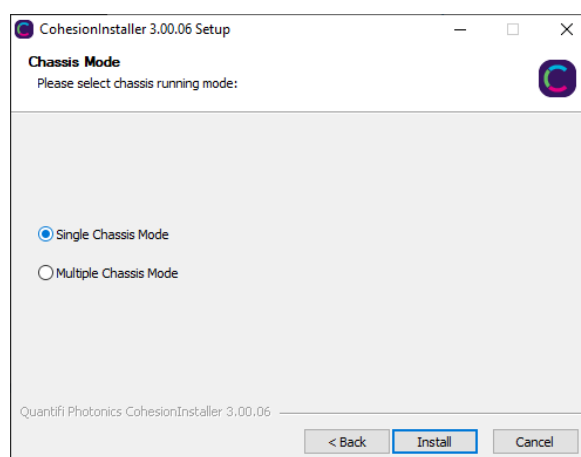


3. Continue with the installation by following the on-screen installation prompts.

Choose the default installation type – **CohesionUI with Apache**.



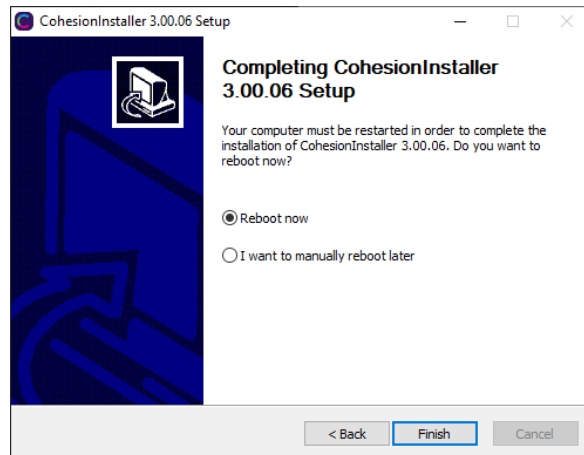
4. Installation will continue with the **Chassis Mode** selection. The default setting is **Single Chassis Mode**. If unsure, proceed with the default mode setting.



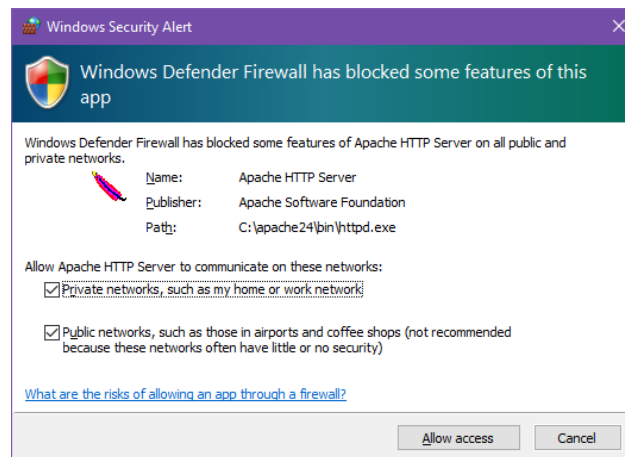
### **⚠ IMPORTANT**

To operate in Multiple Chassis Mode, additional hardware modules are required. The Chassis Mode can be changed at any time, so it is recommended to select **Single Mode** until all other configuration requirements have been met.

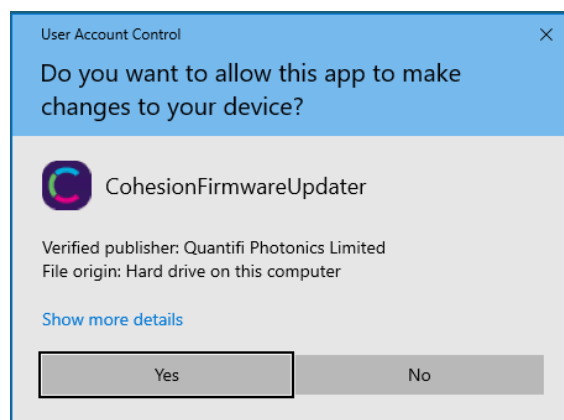
5. At the end of the installation, it is recommended to select the **Reboot now** option, and click **Finish** to complete the installation process.



6. Windows Security Alert may prompt the user for network access. It is recommended that both options be ticked, to allow any network configuration.



7. After rebooting the system, on startup a User Account Control prompt will be displayed to run the Cohesion Firmware Updater Utility. Click **Yes** and proceed with the application.



## 6.2 Quantifi Photonics PXle system utility applications

Contained within the **CohesionInstaller** are two utility applications:

- **Cohesion Manager**
- **Cohesion Firmware Updater**

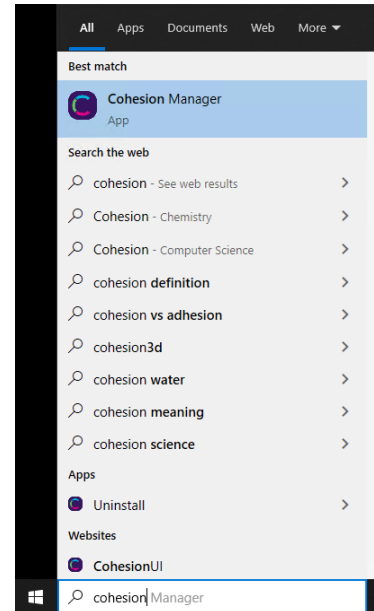
### 6.2.1 Cohesion Manager utility

The Cohesion Manager utility serves as a single window application to give the user an overview of the status of all the Cohesion Software Services running on the system.

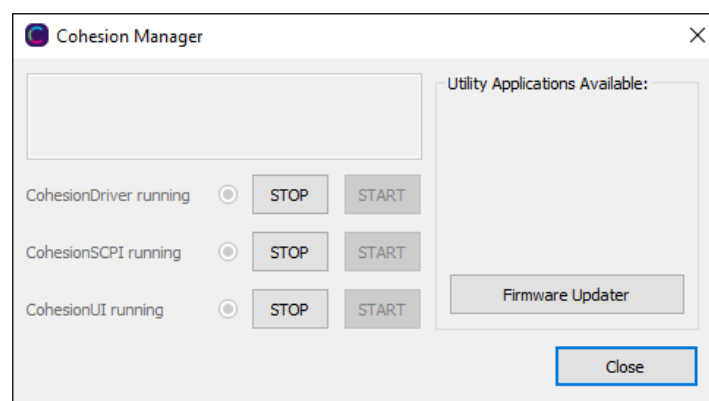
The Cohesion Manager can be accessed via the Windows Start Menu.

Cohesion Manager is designed as a single reference source to check the status of all the running services in one contained window. The utility also allows the user to start or stop the **CohesionDriver service**, **CohesionSCPI service**, or **CohesionUI service** independently.

By default, all these Cohesion Software Services will start automatically on startup of the Windows OS and need to be running to facilitate proper communication with the Quantifi Photonics PXle modules. If there is an issue in detecting or communicating with the modules, run the Cohesion Manager to check the status of the software services.



- **(Required) CohesionDriver** – The CohesionDriver service which directly manages the installed Quantifi Photonics modules.
- **(Required) CohesionSCPI** – The CohesionSCPI service which is the VXI11 compliant SCPI interface for TCP communication with the installed Quantifi Photonics modules.
- **(Optional) CohesionUI** – An optional web service providing a graphical interface for simplified operation of the installed Quantifi Photonics modules.



### ⚠ IMPORTANT

The **Driver and SCPI services** need to be running to facilitate communication with any installed Quantifi Photonics module, **therefore they are listed as REQUIRED**.

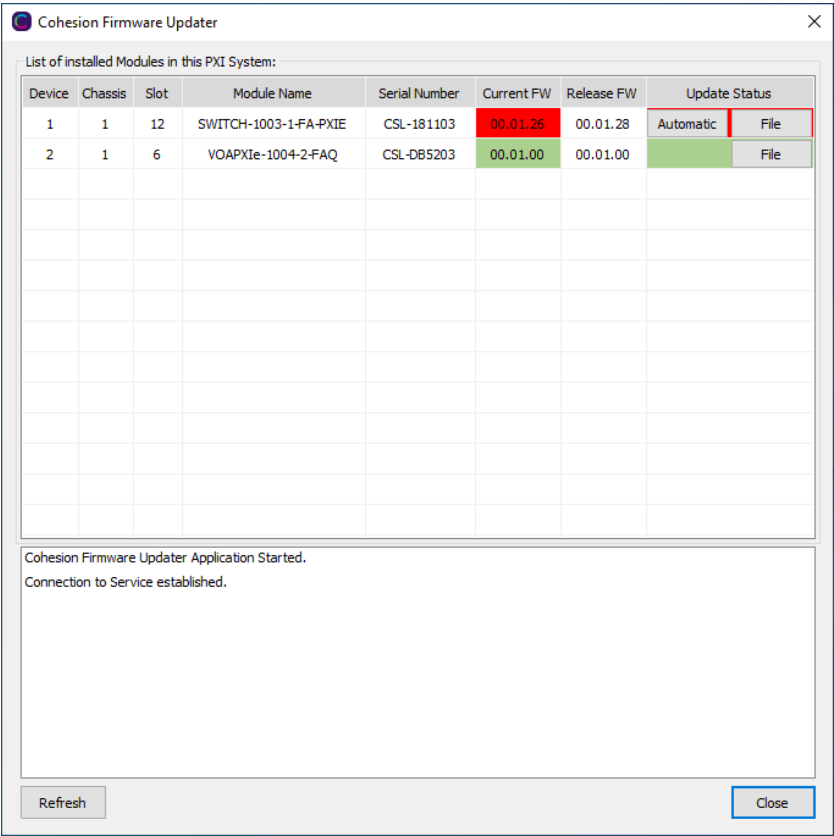
On the right side of the Cohesion Manager window a list of all the installed Quantifi Photonics system utilities is displayed.



6.2.2 Cohesion Firmware Updater utility

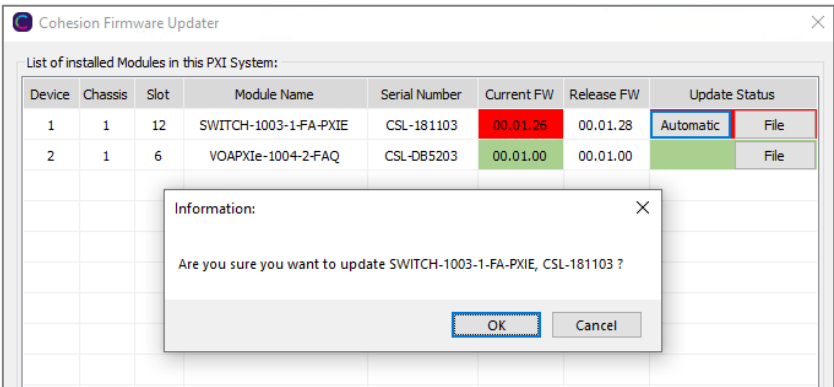
The Cohesion Firmware Updater utility serves as a single window summary application to display the current firmware status of all the Quantifi Photonics PXIe modules installed in the chassis.

The Cohesion Firmware Updater utility can be accessed via the Cohesion Manager application (see section 6.2.1 for more information).

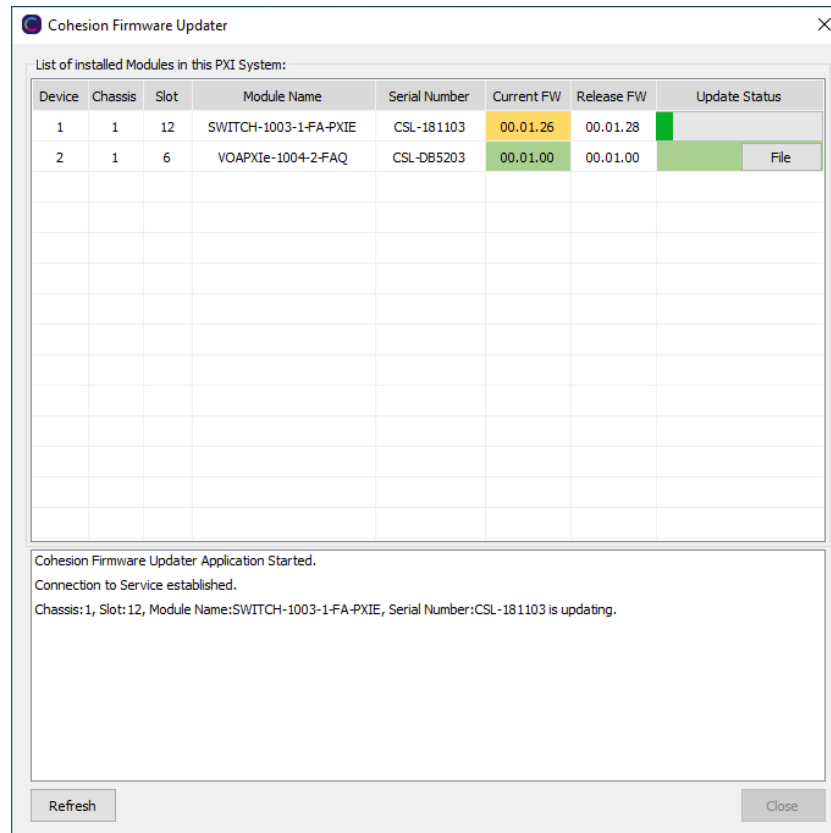


Whenever a new version of the CohesionInstaller is installed on the system, the Cohesion Firmware Updater utility will automatically launch after the system is rebooted. It will show the user the firmware status of all installed Quantifi Photonics modules and allow the user to update the firmware to a new version if applicable.

If a module's firmware is out of date, it is highly recommended to update the firmware to the new available version. Clicking the **Automatic** button will update the module(s) to the latest supported firmware for the installed packages on the system.



After clicking OK, the firmware update progress will be shown in the **Update Status** column. The **File** button is reserved for updating a module to a specific firmware version if required.



## 7 CohesionUI application

CohesionUI is a web-based application that you can use to control any Switch 1200 PXle Series module from Quantifi Photonics. Its cutting-edge design offers a sleek modern interface, cross-device compatibility, multi-instrument control, customizable views, and remote access.

### 7.1 Accessing CohesionUI

To use CohesionUI, you need the IP address of the host chassis, with which you can access the chassis either locally or remotely. For local access, use the embedded PXle controller operating system, and for remote access, use any compatible device that is connected to the PXle chassis via an ethernet connection.

To obtain the IP address, open the **Command Prompt** window on the chassis controller and then run the `ipconfig` command. Note down the IPv4 address that is displayed. For the local IP address, you can use 127.0.0.1 instead.

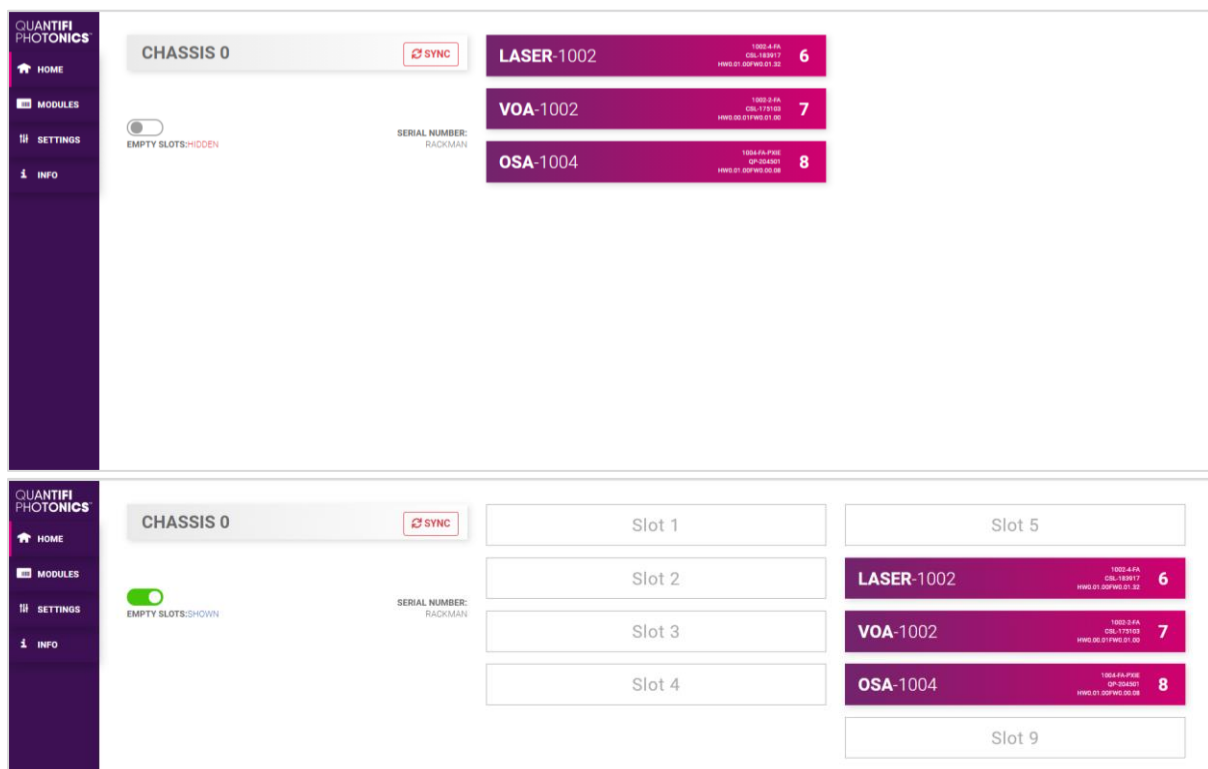
To access CohesionUI locally or remotely, open a compatible browser (Google Chrome or Microsoft Edge) and use the IP address 127.0.0.1 as the URL. Alternatively, you can double-click the desktop icon for CohesionUI or select it from the **Start** menu.



### 7.2 Home page

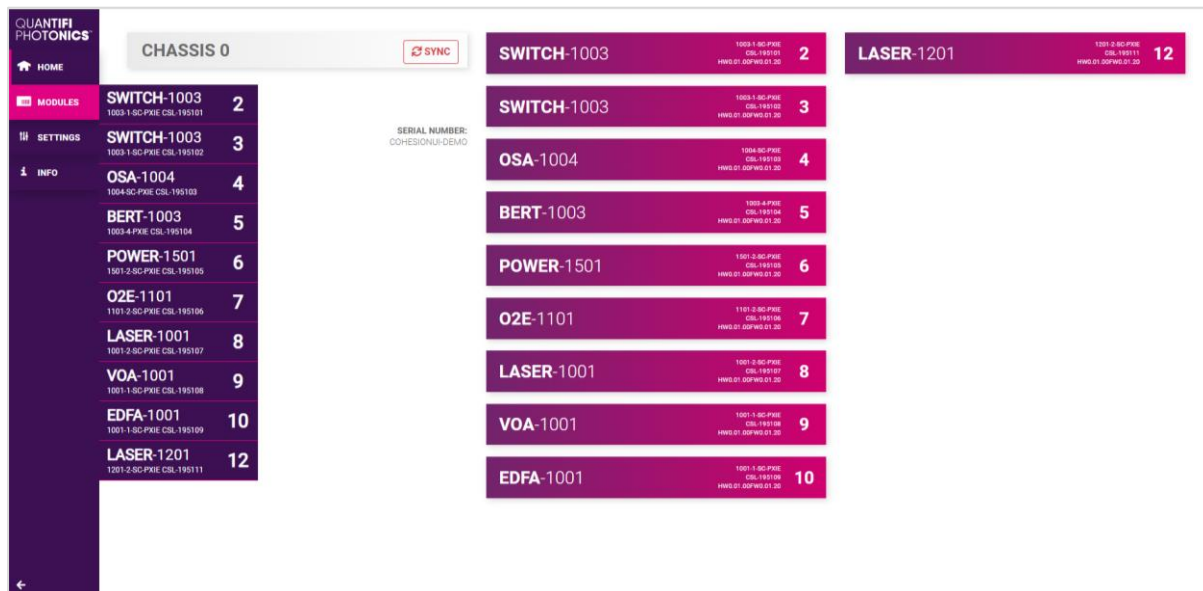
The main landing page in CohesionUI is called the **HOME** page. It displays a graphical representation of the module arrangement in the PXle chassis.

White numbers are displayed beside each module corresponding the slot in which they are installed. The **EMPTY SLOTS** button will toggle the page view to hide (HIDDEN), or to show (SHOWN) the empty slots in the PXle chassis. The default setting is HIDDEN.



### 7.3 Modules list

All the installed modules are displayed on the HOME page. To access the controls for a module, click the corresponding module name, or hover over the **MODULES** button and select a module from the displayed list. Controls for a specific module, or a channel in a module are then displayed.



### 7.4 Set and actual values

Some Quantifi Photonics products will allow the user to set a given parameter's value and then read that parameter (eg. Laser, VOA, O2E, etc). In order to help the user to distinguish between a set value and an actual read value, CohesionUI will format these values differently according to the legend in the top right corner of the window.

- **ACTUAL**: The actual value of the parameter, defined by querying the module.
- **SET**: The intended value of a given parameter, defined by user input.

In the following example, the POWER is SET to 10.00 dBm, but the **ACTUAL** value is **-16.02 dB**. Thus, the user can see both the current and user defined value of a given parameter.

The SET and **ACTUAL** values are only displayed for appropriate parameters which require user input. For parameters that report a value and do not depend on user input, only an **ACTUAL** value is displayed.



## 7.5 Settings page

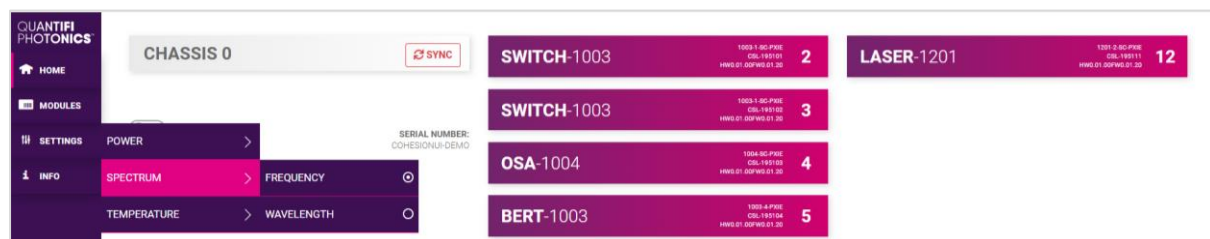
The **SETTINGS** page is used to configure the CohesionUI settings and unit preferences or to synchronize / reinitialize the system. These controls can be accessed by clicking the **SETTINGS** button.

Step size refers to the amount by which the attenuation, frequency, or power is increased or decreased when the + or - button is clicked.



### NOTE

- The unit preferences and settings can be set by hovering over the **SETTINGS** button in the left side menu. This will bring up a dropdown menu that lists all settings for a quick access.
- Whenever the chassis is power cycled, CohesionUI reverts to default settings.



### 7.5.1 System controls for PXle modules

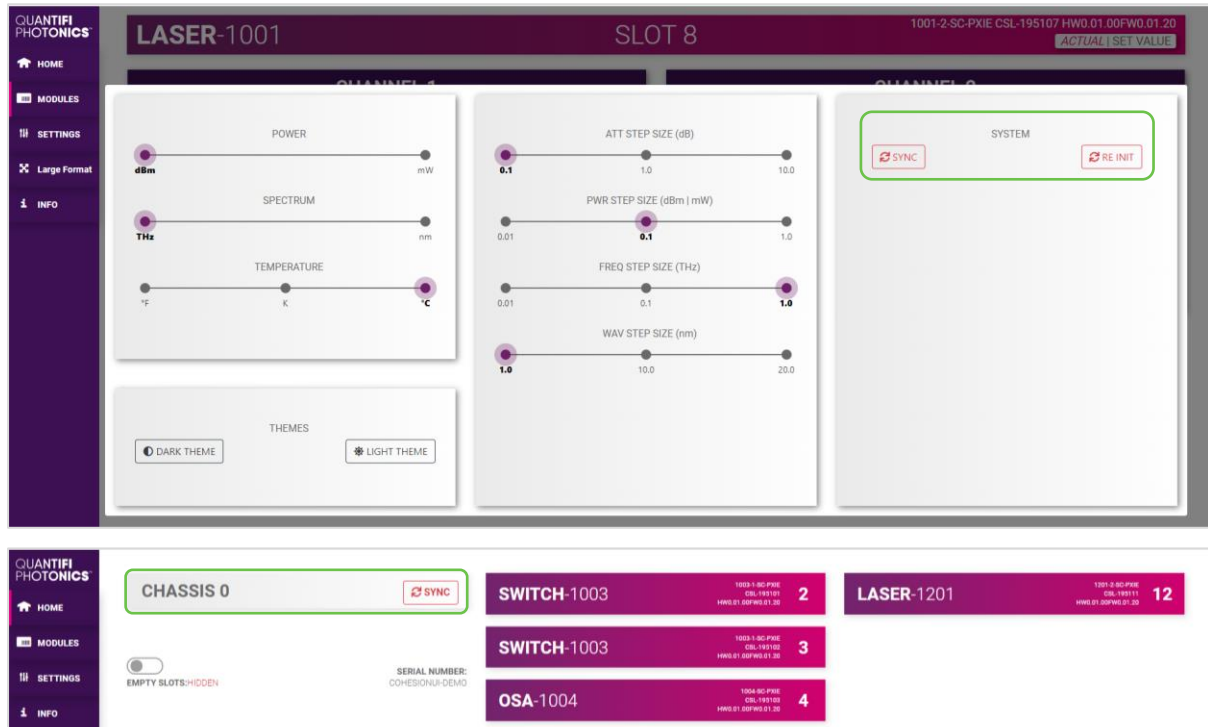
On the **SETTINGS** window there is a **SYSTEM** controls section. These controls are to facilitate re-discovery of any Quantifi Photonics PXle modules there may have been installed after initial startup, or if no modules are displayed in the CohesionUI window. This is useful for users who are operating in a multi-chassis MXI setup, instead of the standard PXle embedded controller setup.

There are two actions in the **SYSTEM** controls section:

- **SYNC**: Synchronize CohesionUI with the latest information from the CohesionSCPI service.
- **RE-INIT**: Re-initialize CohesionUI by synchronizing the CohesionSCPI service with the CohesionDriver service.

The **SYNC** button is also displayed on the **HOME** page beside every chassis in the setup. This allows any chassis to be synchronized independently. After clicking the **SYNC** button,

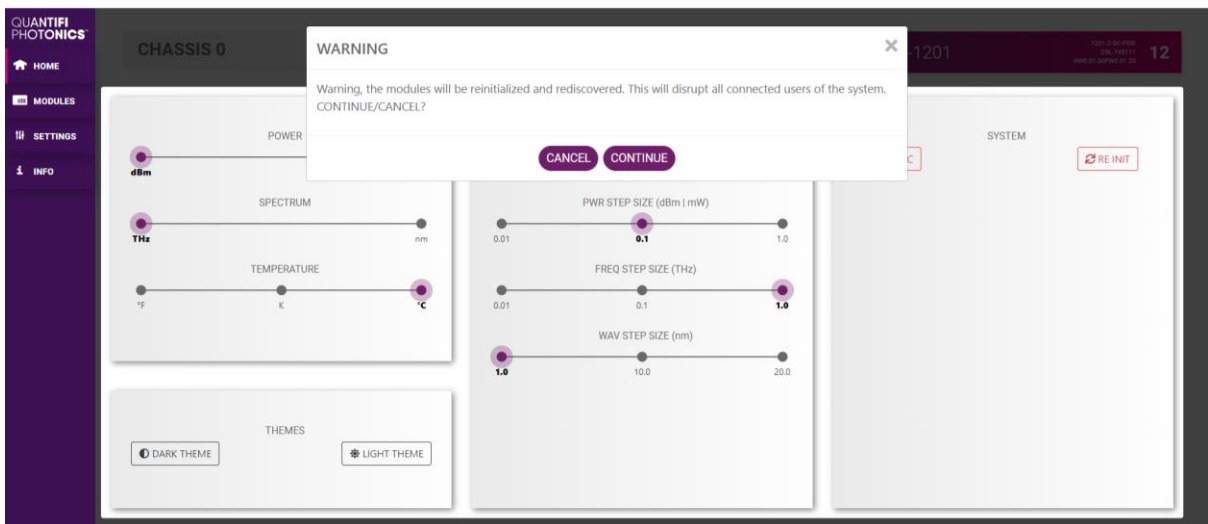
CohesionUI will disable the page while it is synchronizing with the CohesionSCPI service. Once it is complete, the page will be functional again.



Clicking the **RE-INIT** button will bring up a prompt to continue, since this action will temporarily disconnect all modules while the re-initialization with the CohesionDriver service is in progress. Once it is complete, the page will be functional again.

#### NOTE

The **RE-INIT** action will disconnect any connected users to the PXle system while the action is being completed. All modules will be disabled during this time.



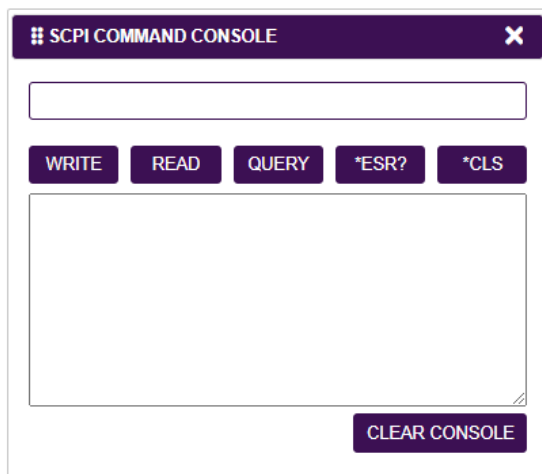
## 7.6 SCPI Command Console

The CohesionUI SCPI Command Console enables you to communicate with Quantifi Photonics PXI modules via SCPI commands. It enables you to test commands and verify their syntax.

For details on available SCPI commands, refer to the programming guide in this manual.

To open the SCPI Command Console:

- On the CohesionUI menu, click **CONSOLE**.  
The console will appear in the bottom right corner.

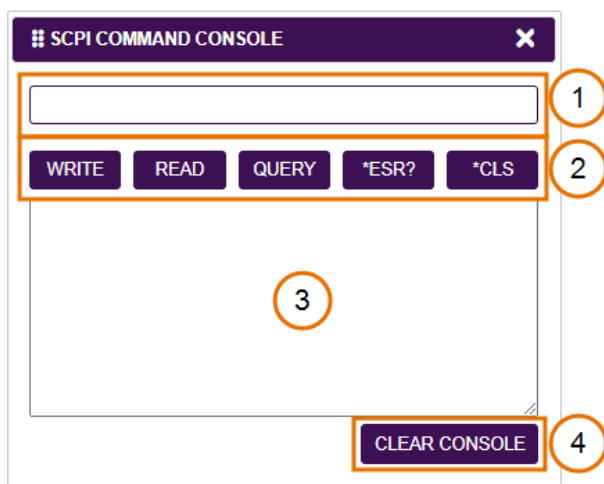


You can move the console by clicking on the title bar and dragging it to any position on the screen. On closing and re-opening, the console will re-appear at its last position.

The console remains open when navigating between different modules. It floats on top of the UI so that you can observe the effect of SCPI commands on a module in real-time.

To communicate with a module via the SCPI Command Console:

1. Enter a command.
2. Select action(s).
3. Review the action response in the output area.
4. (optional) Clear the output area.

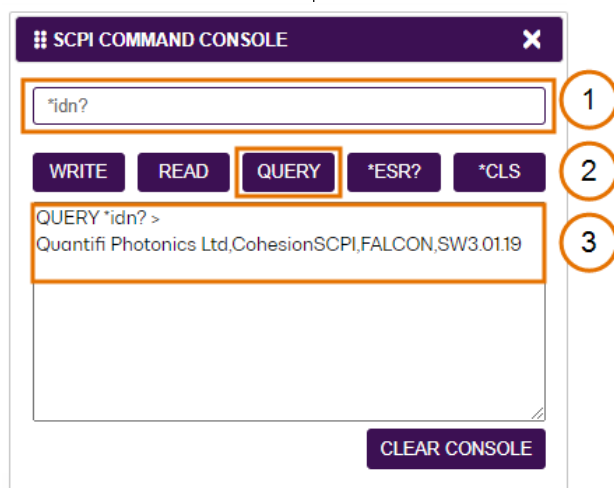


You can choose from these SCPI command actions:

Action	Meaning	FAILED response
WRITE	Send the command to the instrument	The command is invalid. Please check the command and syntax.
READ	(after WRITE) Request the response from the instrument	Response buffer is empty.
QUERY	WRITE and READ	
*ESR?	Query the status event status register (ESR) – this will give you more details and specific information about command failures.  For details on error codes, please refer to the programming guide in this manual.	
*CLS	Clear the response buffer and start fresh – useful when getting out of sync with WRITE and READ actions	

#### Example 1: Send instrument identification query **\*idn?**

1. Enter the command: **\*idn?**
2. Click **QUERY**.
3. The module returns the requested information.

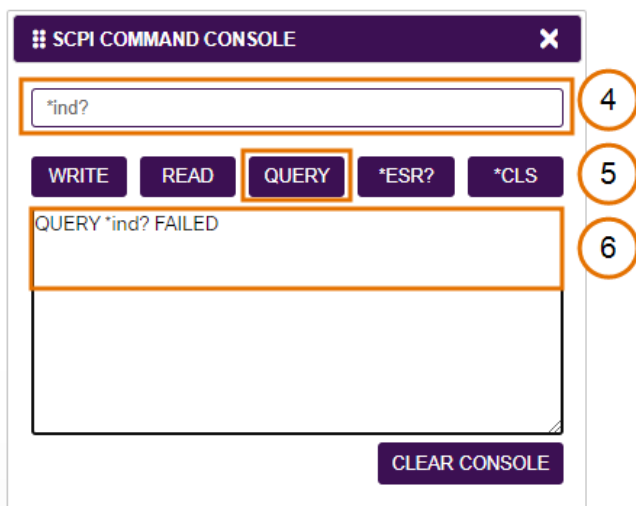


#### Example 2: What happens when I send an incorrect command?

4. Enter an incorrect command, for example: **\*ind?**
5. Click **QUERY**.



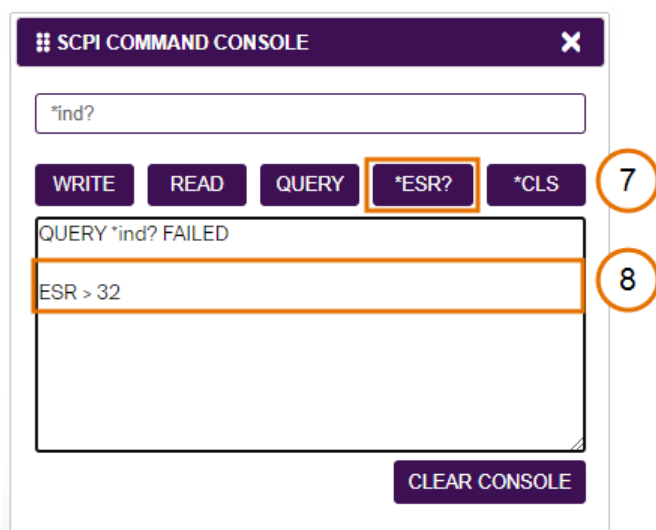
6. The module returns FAILED.



### Example 3: Investigate a command failure

7. Click `*ESR?` to query the event status register and request information about the command failure.
8. The instrument returns the error code, for example `32`.

For details on error codes, please refer to the `*ESR?` command in the programming guide.



## 7.7 Info panel

Clicking the **INFO** button will display an information panel on the right side of the page. Information such as the chassis operation mode, manufacturer, model, and serial number of the chassis, CohesionUI version number, and the version of CohesionSCPI service running on the chassis is displayed in this panel.

QUANTIFI  
PHOTONICS

HOME

MODULES

SETTINGS

INFO

CHASSIS 0

SYNC

EMPTY SLOTS: HIDDEN

SERIAL NUMBER:  
COHESIONUI-DEMO

SWITCH-1003

1003-1-80-PXle  
CBL-1901101  
HWREV: 01\_000 HWID: 01\_30

2

SWITCH-1003

1003-1-80-PXle  
CBL-1901101  
HWREV: 01\_000 HWID: 01\_30

3

OSA-1004

1004-4-PXle  
CBL-1901101  
HWREV: 01\_000 HWID: 01\_30

4

BERT-1003

1003-4-PXle  
CBL-1901101  
HWREV: 01\_000 HWID: 01\_30

5

POWER-1501

1501-2-80-PXle  
CBL-1901101  
HWREV: 01\_000 HWID: 01\_30

6

O2E-1101

1101-2-80-PXle  
CBL-1901101  
HWREV: 01\_000 HWID: 01\_30

7

LASER-1001

1001-2-80-PXle  
CBL-1901101  
HWREV: 01\_000 HWID: 01\_30

8

VOA-1001

1001-1-80-PXle  
CBL-1901101  
HWREV: 01\_000 HWID: 01\_30

9

EDFA-1001

1001-1-80-PXle  
CBL-1901101  
HWREV: 01\_000 HWID: 01\_30

10

LASER-1201

COHESIONUI™

COMPANY  
QUANTIFI PHOTONICS LTD

MODEL  
COHESIONSCRIPT

SERIAL  
COHESIONUI-DEMO

UI VERSION  
3.00.02

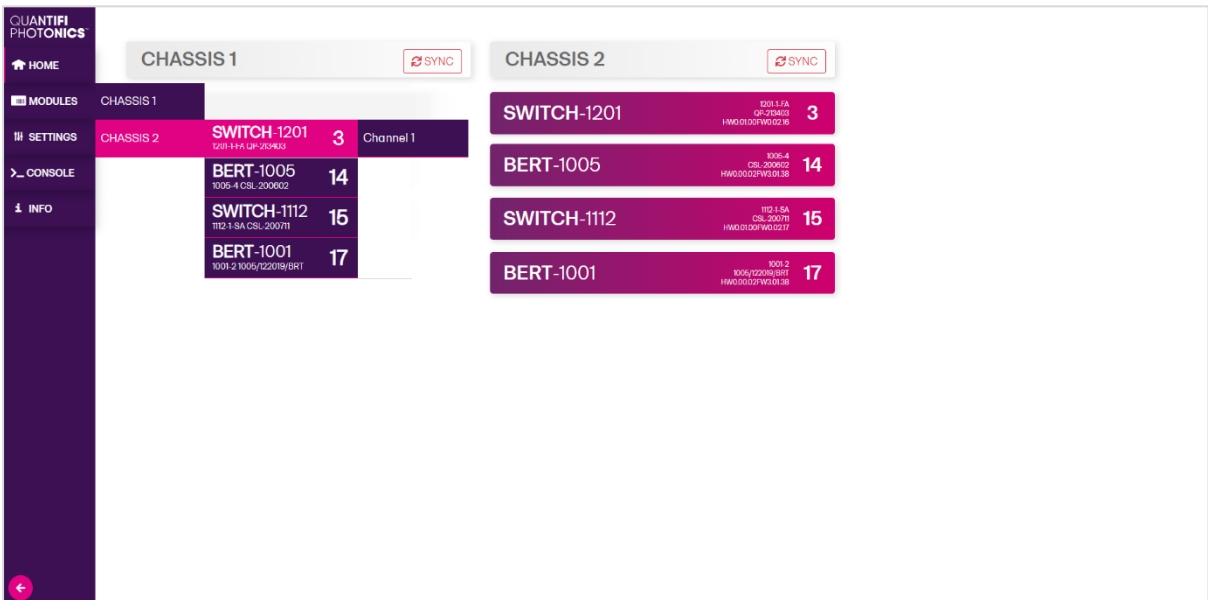
SERVER VERSION  
3.00.05

DRIVER VERSION  
3.01.04

CHASSIS MODE  
SINGLE

## 8 Switch 1200 PXle Series control with CohesionUI

To control the Switch PXle module through CohesionUI, click the desired module installed in the chassis. Alternatively, hovering over the **MODULES** menu button on the left will bring up a dropdown menu that the Switch PXle module can also be selected from.

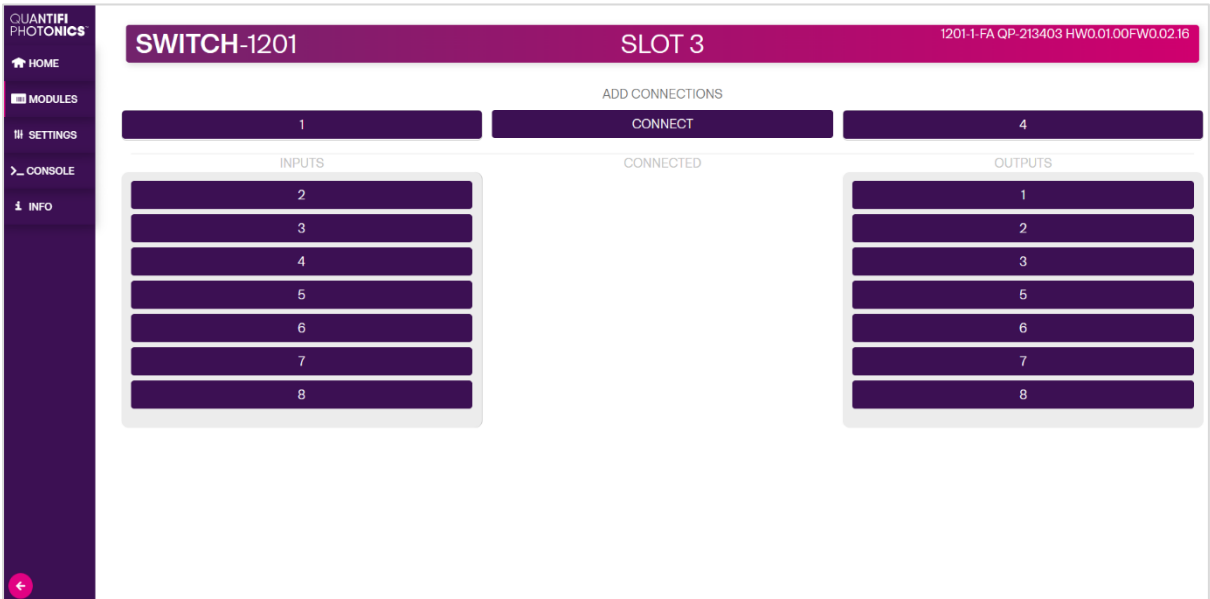


After clicking the desired Switch PXle module, its control page is displayed. All information relating to the module such as model number, serial number and firmware versions are displayed in the top right corner of the window.



The Switch PXle module is always in a **PARK** state by default. When in **PARK**, the Switch PXle will act like an optical terminator.

To route the Switch PXle module to the desired port, first click the intended A port (INPUTS) and then select the intended B port (OUTPUTS). Clicking **CONNECT** will set the routing state.



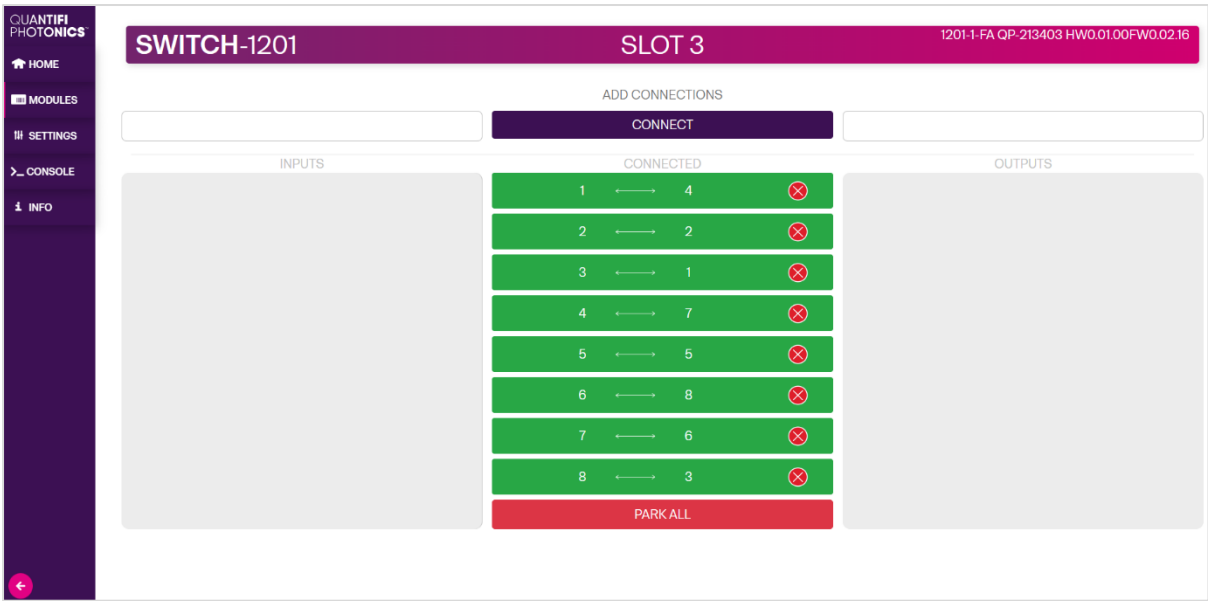
The Switch PXle module can route the light very quickly, however there is a slight delay between re-routing the switch and CohesionUI updating the screen.

Each routed port pair will be displayed in a green box in the **CONNECTED** list.



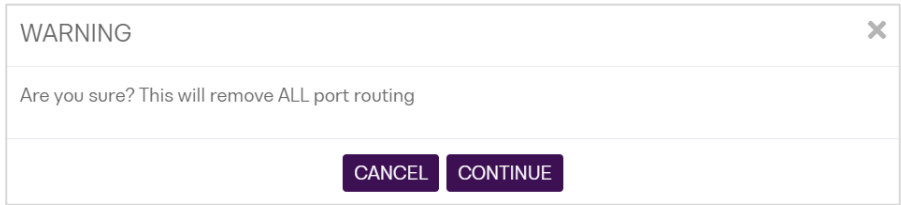
To disconnect a routed pair, click the **red cross** for the pair.

To disconnect all routed pairs and set the routing for all A ports (INPUTS) to PARK (0), you can click the **PARK ALL** button.



**NOTE**

When the **PARK ALL** button is clicked, a system popup will appear to confirm the action. By clicking this, all the connected port pairs will be disconnected and set to the default PARK state.



## 9 Programming guide

### Introduction

Remote communication with the CohesionSCPI service is achieved through the Standard Commands for Programmable Instruments (SCPI). Support for VISA I/O API over TCP/IP is provided by the VXI-11 compliant CohesionSCPI service. With VISA communication drivers installed on the client, the implementation of VISA programming within environments such as MATLAB becomes available.

This guide provides general information on the commands available to communicate with the CohesionSCPI service remotely using the VISA I/O.

### **⚠ IMPORTANT**

In NI-MAX a RIO interface will show up, however there are no communication methods available or implemented on this interface. Quantifi Photonics products are **ONLY** accessible through the **VISA TCPIP INSTR** interface provided by the CohesionSCPI service installed on the system.

### 9.1 Programming conventions

This section details the programming and measurement conventions to follow while executing the commands for the CohesionSCPI service.

Parameter	Default Unit	Alternative Units
Power	DBM	DBM
Frequency	HZ	THZ, GHZ, MHZ, KHZ
Frequency Fine	HZ	THZ, GHZ, MHZ, KHZ
Wavelength	NM	NM, PM

Argument	Data Format
<wsp>	Specifies whitespace character (01 <sub>16</sub> – 09 <sub>16</sub> , 0B <sub>16</sub> – 20 <sub>16</sub> ).
<value>	Is numerical data, an integer, a decimal, exponential (10e-9 or 5.8e6) or string.
[VALUE1   VALUE2]	A parameter choice. The '   ' separates the unique parameters available, only one of the choices can be used. In the example, either the input parameter [VALUE1] or [VALUE2] can be used, but not both. Some commands may have more than two choices available. This parameter can be omitted where the command has a default defined in the command description.

#### 9.1.1 Index addressing of modules (slot, source) and units (channel)

When executing commands, it is almost always necessary to provide the index of a specific Switch PXle module or an index of a specific installed unit.

**For the commands that require index values:**

- <c>: is the chassis index in which the specific blade module is installed; this is an integer, inclusive of 0.
- <n>: is the slot (or source) index of the specific blade module, this is an integer, <1 to 18>
- <m>: is the channel index of a specific unit in the module, this is an integer, <1 to 6>.

### Message queues

Information is exchanged in the form of messages. These messages are held in input and output queues. The output queue stores responses to query commands. The CohesionSCPI service transmits any data in the output queue when a read request is received. Unless specified, all output response data is transmitted in ASCII format.

## 9.2 Status and event registers

### 9.2.1 Standard Event Status Register

The Standard Event Status Register (SESR) is modified by the Switch PXle with the results of the command operations.

Bit	Description
7 (MSB), 6	Not used
5	Is set when a <b>Command Error</b> event has been detected
4	Is set when a command <b>Execution Error</b> has been detected
3	Is set when a <b>Device Dependent Error</b> event has been detected
2	Is set when there a <b>Query Error</b> event has been detected
1	Not used
0 (LSB)	Is set when an <b>Operation Complete</b> event has been generated

### 9.2.2 Standard Event Status Enable Register (Mask)

The Standard Event Status Enable Register (SESR Mask) is used to build the Event Status Bit (ESB) within the Status Byte Register (STB). To ignore any of the events detected and set in the SESR, set the corresponding bit within the SESR Mask to 0. The STB can then be queried and the value of the ESB can be used to determine service request requirements based on the SESR Mask applied.

#### NOTE

The default bit values within the SESR Mask are all 0.

### 9.2.3 Status Byte Register

The Status Byte Register (STB) is built from all other status registers and masks. This register can be used in queries to determine if an event has been detected and where that event has been detected.

Bit	Description
7 (MSB)	Not used
6	The Master Summary Status (MSS) bit is set from the <b>STB</b> and <b>SRE Mask</b>
5	The Event Status Bit (ESB) is set from the <b>SESR</b> and the <b>SESR Mask</b>
4	Message Available (MAV) is set when there is <b>data in the output queue</b>
3, 2, 1, 0 (LSB)	Not used

### 9.2.4 Service Request Enable Register (Mask)

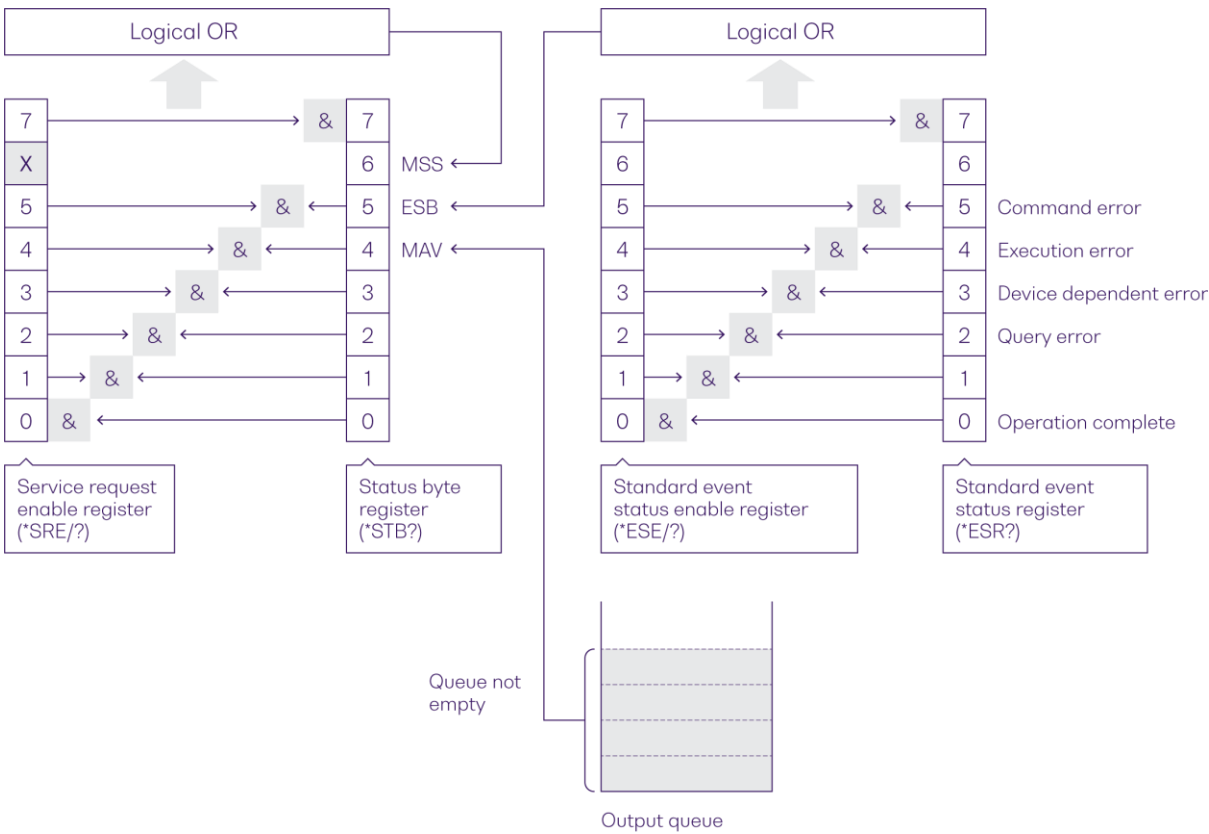
The Standard Request Enable Register (SRE Mask) is used to build the Master Summary Status Bit (MSS) within the Status Byte Register (STB). To ignore any of the events detected and set in the STB register itself, set the corresponding bit within the SRE Mask to 0. The STB can then be queried and the value of the MSS can be used to determine the type of service request required based on the SRE Mask applied.

#### NOTE

The default bit values within the SESR Mask are all 0.

Bit	Description
7 (MSB)	Not used
6	The Master Summary Status (MSS) bit is set from the <b>STB</b> and <b>SRE Mask</b>
5	The Event Status Bit (ESB) is set from the <b>SESR</b> and the <b>SESR Mask</b>
4	Message Available (MAV) is set when there is data in the output queue
3, 2, 1, 0 (LSB)	Not used

9.2.5 Status and event registers diagram





### 9.3 Common system command summary

Common commands	Description
*CLS	-Clear Status command
*IDN?	-Query the chassis identification
*OPC?	-Query the Operation Complete Status
*OPT?	-Query the modules managed by the CohesionSCPI service
*ESR?	-Query the Standard Event Status Register

### 9.4 Common system command descriptions

Command	*CLS
Syntax	*CLS
Description	Clear Status command
Parameters	No parameters
Response	No response
Example	*CLS

Command	*ESR?		
Syntax	*ESR?		
Description	Query the Standard Event Status Register		
Parameters	No parameters		
Response	Unsigned integer 8 bit value for the register <0 to 255>, as a string.		
	Bit	Description	Decimal Value
	7 (MSB)	Not used	0
	6	Not used	0
	5	Command Error	32
	4	Execution Error	16
	3	Device dependent Error	8
	2	Not used	0
	1	Not used	0
0 (LSB)	Not used	0	
Example	*ESR? -> 8		
	*ESR? -> 32		

#### ⚠ IMPORTANT

It is recommended to use the \*ESR? command query after every command that is sent to the device.

The \*ESR? query will be able to catch:

- **Device dependent Error** – the device is reporting an error in operation.
- **Execution Error** – SCPI was unable to execute the given command.
- **Command Error** – SCPI was unable to parse the given command, likely due to an incorrect command.

Command	*IDN?
Syntax	*IDN?
Description	Query the chassis identification
Parameters	No parameters
Response	Comma separated string with the <manufacturer>,<server name>,<chassis controller name>,<server version>
Example	*IDN? -> Quantifi Photonics, CohesionSCPI service, PXIE-8133, FW2.0.15

Command	*OPC?
Syntax	*OPC?
Description	Query the Operation Complete Status
Parameters	No parameters
Response	1 is returned if all the modules installed in the chassis are ready to execute commands 0 is returned if any module installed in the chassis still has a command to execute in the input queue
Example	*OPC? -> 1

Command	*OPT?
Syntax	*OPT?
Description	Query the modules managed by the CohesionSCPI service
Parameters	No parameters
Response	Response will be a comma separated string of the installed modules in the chassis
Example	*OPT? -> ,Switch-1002-2-FA-PXIe,Switch-1003-1-FC-PXIe,,VOA-1001-2-FA-PXIe,,,,,O2E-1001-1-FC-PXIe,,,,,,,,

## 9.5 Specific command summary

Slot commands	Description
:SLOT<n> :OPC? :OPTions? :IDN?	- Query the status of the Operation Complete bit - Query the modules installed on the slot - Query the Identifier for the slot; returns the manufacturer, part number, serial number, hardware and firmware versions
Configuration commands	Description
:ROUTe<n> :ERROR? :CHANnel<m> :ERROR? :STATE/?	- Query the error status of the Switch - Query the error status of the specified channel of the Switch - Set or query the routing state of the Switch

## 9.6 Specific command descriptions

### 9.6.1 Slot commands

Command	:SLOT<n>:OPC?
Syntax	:SLOT<n>:OPC?
Description	Query the status of the Operation Complete bit
Parameters	No parameters
Response	1 is returned if the module is ready to execute a new operation 0 is returned if the module is busy
Example	:SLOT1:OPC? -> 1

Command	:SLOT<n>:OPTions?
Syntax	:SLOT<n>:OPTions?
Description	Query the modules installed on the slot
Parameters	No parameters
Response	The response will be a comma separated string of channels installed in the Switch. If a module is not installed in a channel, it will not return any identification string
Example	:SLOT3:OPT? -> 1,,,

Command	:SLOT<n>:IDN?
Syntax	:SLOT<n>:IDN?
Description	Query the Identifier for the slot; returns the manufacturer, part number, serial number, hardware and firmware versions
Parameters	No parameters
Response	Comma separated string containing the <manufacturer>, <part number>, <serial number>, <hardware version><firmware version>
Example	:SLOT1:IDN? -> Quantifi Photonics, SWITCH-1201-1-SA-PXIE, QP-180101, HW1.0FW1.02 Hardware and firmware versions are not separated by a comma

## 9.6.2 Configuration commands

Command	:ROUTe<n>:ERROR?
Syntax	:ROUTe<n>:ERROR?
Description	Query the error status of the Switch
Parameters	No parameters
Response	The response of an integer value representing a bitmask of switch errors. 8        Switch D(4) reports an error 4        Switch C(3) reports an error 2        Switch B(2) reports an error 1        Switch A (1) reports an error 0        There are no reported errors
Example	:ROUT1:ERROR? -> 1

Command	:ROUTe<n>:CHANnel<m>:ERROR?
Syntax	:ROUTe<n>:CHANnel<m>:ERROR?
Description	Query the error status of the specified channel of the Switch PXle module
Parameters	No parameters
Response	1        Switch reporting an error 0        No error on switch
Example	:ROUT1:CHAN1:ERROR? -> 0

Command	:ROUTe<n>:CHANnel<m>:PORTS?
Syntax	:ROUTe<n>:CHANnel<m>:PORTS?
Description	Query the number of ports on the Switch PXle module
Parameters	None
Response	A comma separated string denoting the port layout of the Switch PXle module as <value1>,<value2>, where the port layout of the switch follows as value1 x value2. <value1> corresponds to the number of A ports, and <value2> corresponds to the number of B ports.
Example	:ROUT3:CHAN1:PORTS? -> 8,8

Command	:ROUTe<n>:CHANnel<m>:STATE
Syntax	:ROUTe<n>:CHANnel<m>:STATE<wsp><value1>,<value2>...<valueN>
Description	Set the routing state of the N A ports of the Switch PXle
Parameters	The Switch 1200 PXle Series are NxM grid switches, meaning that the intended route port for each of the N A ports must all be set together. Each of the N numbered A ports must be assigned a unique B port, numbered from 0 to M, where M is the number of B ports.  <value1>:        A valid port B number representing the routing port for Port A <sub>1</sub> . <value2>:        A valid port B number representing the routing port for Port A <sub>2</sub> . <valueN>:        A valid port B number representing the routing port for Port A <sub>N</sub> .  A value of 0 denotes a PARK state.
Response	No response
Example	:ROUT1:CHAN1:STATE 2,4,6,8,1,3,0,7 <b>Note:</b> In this example ports A <sub>1</sub> ,A <sub>2</sub> ,A <sub>3</sub> ,A <sub>4</sub> ,A <sub>5</sub> ,A <sub>6</sub> ,A <sub>7</sub> ,A <sub>8</sub> have been routed to ports B <sub>2</sub> ,B <sub>4</sub> ,B <sub>6</sub> ,B <sub>8</sub> ,B <sub>1</sub> ,B <sub>3</sub> ,PARK,B <sub>7</sub> .

Command	:ROUTe<n>:CHANnel<m>:STATE?
Syntax	:ROUTe<n>:CHANnel<m>:STATE?<wsp>[ALL]
Description	Query the routing states for all the A ports of the Switch PXle module
Parameters	ALL: Returns the SET and ACT (actual) route state for each of the N A ports.
Response	<p>If no parameter is provided, the response is a comma separated string of the SET route port for each A port as &lt;SET<sub>A1</sub>&gt;,&lt;SET<sub>A2</sub>&gt;,...&lt;SET<sub>AN</sub>&gt;.</p> <p>If the ALL parameter is provided, the response is a colon separated string representing the SET and ACT route values for each of the N A ports as &lt;SET<sub>A1</sub>&gt;,&lt;ACT<sub>A1</sub>&gt;:&lt;SET<sub>A2</sub>&gt;,&lt;ACT<sub>A2</sub>&gt;:...&lt;SET<sub>AN</sub>&gt;,&lt;ACT<sub>AN</sub>&gt;. Each A port's SET and ACT values are separated by a comma (,), and each port is separated by a colon (:).</p>
Example	<p>:ROUT3:CHAN1:STATE? -&gt; 2,4,6,8,1,3,0,7</p> <p>:ROUTE3:CHAN1:STATE? ALL -&gt; 2,2:4,4:6,6:8,8:1,1:3,3:0,0:7,7</p> <p><b>Note:</b> In this example the return indicates that port A<sub>1</sub> has been set to B<sub>2</sub> and is actually at B<sub>2</sub>, port A<sub>2</sub> has been set to B<sub>4</sub> and is actually at B<sub>4</sub>,...port A<sub>8</sub> has been set to B<sub>7</sub> and is actually at B<sub>7</sub>. Port A<sub>7</sub> has been set to PARK and is actually in PARK.</p>

## 9.7 Switch PXle Multi Chassis mode operation

Multiple chassis can be connected to operate in **Multi Chassis Mode**. To operate in Multi Chassis Mode, CohesionSCPI service must be version 1.02.06 or later.

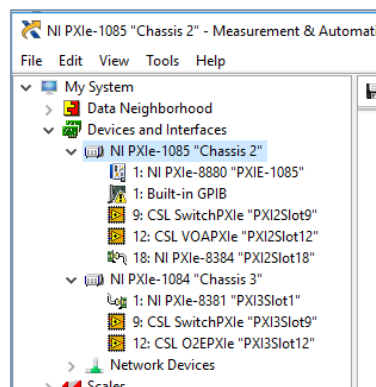
### 9.7.1 NI-MAX application Multi Chassis mode

#### **⚠ IMPORTANT**

The CohesionSCPI service does not manage the chassis numbers. These are controlled by the NI Platform Services (and through NI-MAX).

Even if the CohesionSCPI service is in Multi Chassis mode, if a chassis is connected but has no installed modules, it will not show up when \*OPT? is run.

In the example shown below, there are two chassis connected via the PXle-8384 to PXle-8381 connection. Chassis #2 has the controller running CohesionSCPI service, and Chassis #3 is the 'extended' chassis.



### 9.7.2 SCPI Multi Chassis commands

Command	:SYSTEM:CHASSIS?
Syntax	:SYSTEM:CHASSIS?<wsp>[LIST MODE]
Description	Query the Chassis Mode configuration
Parameters	No parameters
Response	<p>LIST: Returns a comma separated list of valid chassis index numbers discovered by the CohesionSCPI service. These are chassis that have modules installed.</p> <p>MODE: Returns the current Chassis Mode the CohesionSCPI service is operating in (SINGLE or MULTI).</p> <p>None: Returns the number of chassis managed by the CohesionSCPI service. If operating in SINGLE mode, this will always return 1.</p>
Example	<p>In Single chassis mode:</p> <pre>:SYSTEM:CHASSIS? -&gt; 1 :SYSTEM:CHASSIS? LIST -&gt; 0 :SYSTEM:CHASSIS? MODE -&gt; SINGLE</pre> <p>In Multi chassis mode:</p> <pre>:SYSTEM:CHASSIS? -&gt; 2 :SYSTEM:CHASSIS? LIST -&gt; 2,3 :SYSTEM:CHASSIS? MODE -&gt; MULTI</pre>

**⚠ IMPORTANT**

Changing the CohesionSCPI service Chassis Mode will rediscover all Chassis and installed modules.

Command	<code>:SYSTEM:CHASSIS</code>
Syntax	<code>:SYSTEM:CHASSIS&lt;wsp&gt; [SINGLE   MULTI]</code>
Description	Set the Chassis Mode configuration
Parameters	<b>SINGLE:</b> Set CohesionSCPI service to operate in SINGLE Chassis Mode <b>MULTI:</b> Set CohesionSCPI service to operate in MULTI Chassis Mode
Response	No response
Example	<code>:SYSTEM:CHASSIS SINGLE</code>

In Multi chassis mode, all the commands given above in the Specific Command Summary will still work, but they must be prefixed with `:CHASSIS<c>`.

**Common command example:**

Single Chassis Mode	<code>:SLOT2:IDN?</code>
Multi Chassis Mode	<code>:CHASSIS1:SLOT2:IDN?</code>

**Specific command example:**

Single Chassis Mode	<code>:SOUR2:CHAN2:POW? MAX</code>
Multi Chassis Mode	<code>:CHASSIS1:SOUR2:CHAN2:POW? MAX</code>

## 10 Example: Control of the Switch 1200 PXle Series

The following is a simple example of how to control the Switch 1200 PXle Series module by using SCPI commands. See the previous section for specific details and extra parameters that the listed commands accept.

After any command, it is recommended to query the `*ESR?` command. This will allow debugging of unreceived or incorrect commands that were sent to the product.

Description	Command example
Identify the Switch PXle module	
1. Query to confirm the correct PXle chassis is setup	<code>:*IDN?</code>
2. Query the available module configuration	<code>:*OPT?</code>
3. Query the identification information for a specific slot module	<code>:SLOT3:IDN?</code>
4. Query the port configuration of the SWITCH	<code>:ROUTe3:CHANne11:PORTS?</code>
Setting the state of the Switch PXle module	
1. Set the output route of Port A5 to Port B8. All other ports are left unrouted.	<code>:ROUTe3:CHANne11:STATE</code> <code>0,0,0,0,8,0,0,0</code>
2. Change the routing state of port A1 to port B6, and set port A5 to PARK.	<code>:ROUTe3:CHANne11:STATE</code> <code>6,0,0,0,0,0,0,0</code>
Querying the state of the Switch PXle module	
1. Query the routing setup of the SWITCH	<code>:ROUTe3:CHANne11:STATE?</code>

### IMPORTANT

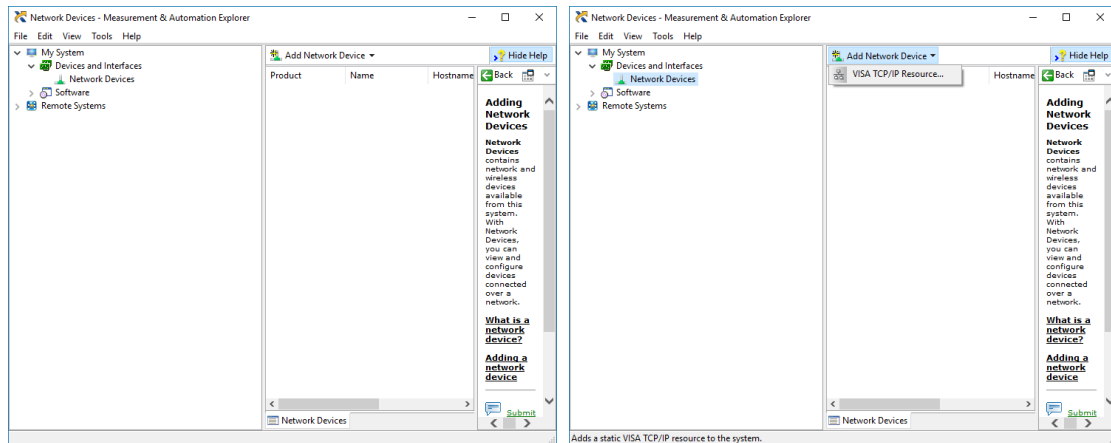
The following section details the various methods that a user may send these commands to the Switch PXle via **SCPI** commands



## 10.1 NI-MAX application

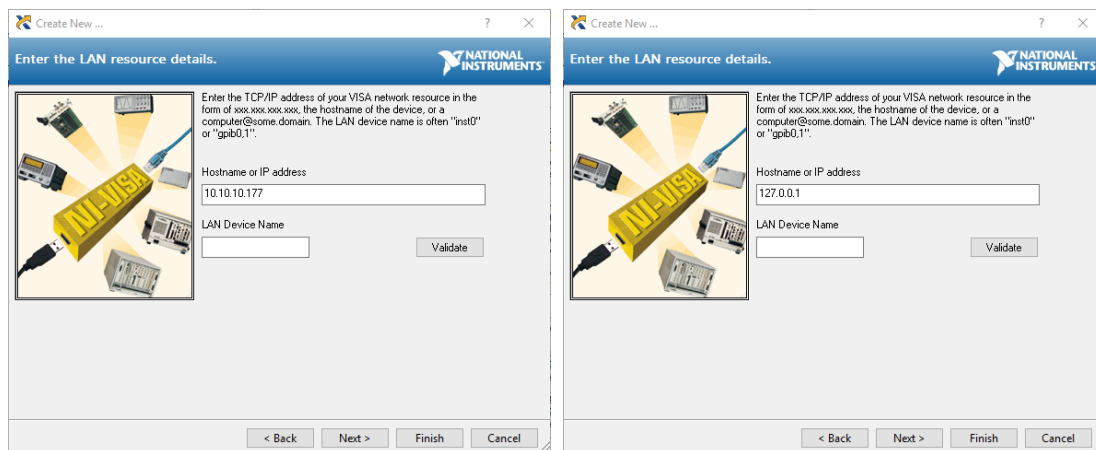
To communicate with any Switch PXIe module, the chassis / instrument must first be setup as a TCP/IP instrument.

1. After installing NI-MAX, launch the application. In the left side panel of the window, click the **Devices and Interfaces** option. A drop down of available instruments detected will show up.
2. Click on **Network Devices**, then click **Add Network Devices** and select **VISA TCP/IP Resource**.



3. Select **Manual Entry of LAN Instrument**. Enter in the Hostname or IP Address. The top image is an example of operating remotely, the bottom image is an example of operating locally.

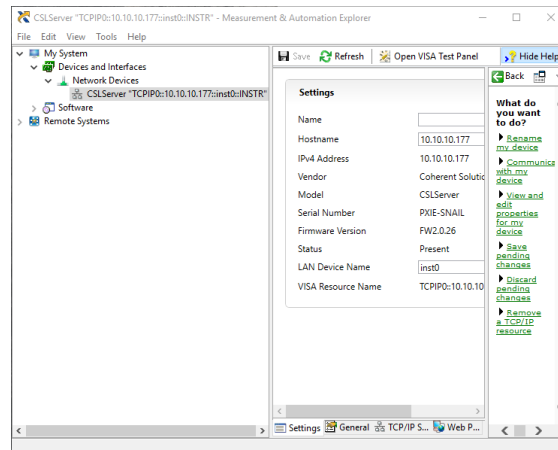
Note when operating locally, enter in the localhost IP address of **127.0.0.1**. Click **Finish** to end the setup process.



## 10.2 NI-VISA application

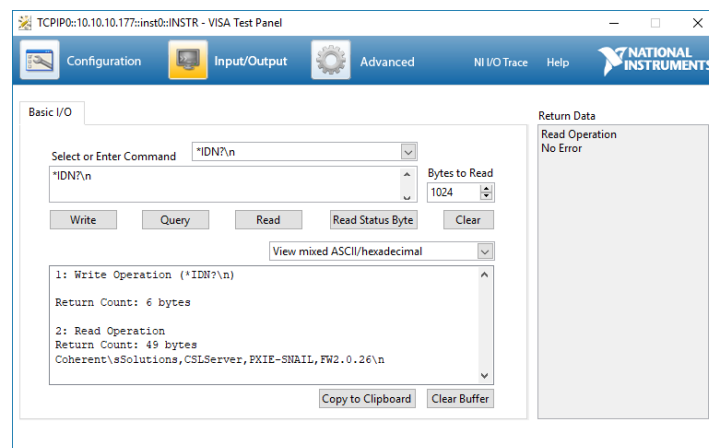
NI-VISA is used to communicate with the PXIe chassis or installed modules / instruments. The above steps must be completed before attempting to communicate using NI-VISA.

1. Launch NI-MAX. In the left-hand side menu, select an instrument from the **Network Devices** list.



2. On the right-hand side panel, select **Open VISA Test Panel**. A new window will popup. Click the **Input / Output** button from the window menu.

Valid chassis and module commands can be entered in, and their returns queried.



### 10.3 Python® 2.7 code example

The following example shows how to communicate with the Switch PXle module using Python code. For a list of supported and valid SCPI commands, refer to the **Programming Guide**.

```
# You can get Vxill from pip:
# pip install python-vxill==0.9
import vxill
from vxill.vxill import VxillException
# replace this with the IP of your device
ip = "127.0.0.1"
try:
    print("connecting to " + ip + " ... ")
    instrument = vxill.Instrument(ip)
    print("connected")
    print("checking IDN...")
    command = "*IDN?"
    data = instrument.ask(command)
    print("IDN: " + data)
    print("checking OPT...")
    command = "*OPT?"
    data = instrument.ask(command)
    print("OPT: " + data)
    # replace this with a valid command for your device (read # the
    # programming guide section for examples)
    command = ""
    print("writing a specific command")
    instrument.write(command)
    print("checking ESR")
    command = "*ESR?"
    data = instrument.ask(command)
    print("*ESR?: " + data)
except VxillException as e:
    # pass
    print("ERROR" + str(e) + ", command: " + str(command))
```

## 10.4 MATLAB® code example

To communicate with the Switch PXIe module in MATLAB® the installation of a VISA IO driver is required. These drivers enable the creation of the Interface Object for instrument communication.

If developing locally on the Switch PXIe Platform, then these will already be installed. However, if development is on a remotely connected system the VISA Libraries, e.g. National Instruments NI-VISA will have to be installed.

### **IMPORTANT**

MATLAB 2010x or later with the Instrument Control Toolbox is required to execute the code detailed in this section.

The following example shows how to communicate with the Switch PXIe using MATLAB code. For a list of supported and valid SCPI commands, refer to the **Programming Guide**.

```
% Find a VISA-TCPIP object. This is if the VISA object has already been
% created with tmtool or has been removed from the workspace without
% first being closed (cleanly disconnected).
PXIE_Chassis = instrfind('Type', 'visa-tcpip', ...
    'RsrcName', 'TCPIP0::10.10.10.89::inst0::INSTR', 'Tag', '');
% Create the 'agilent' VISA-TCPIP object if it does not exist
% otherwise use the object that was found.
if isempty(PXIE_Chassis)
    PXIE_Chassis = visa('agilent', 'TCPIP0::10.10.10.89::inst0::INSTR');
else
    fclose(PXIE_Chassis);
    PXIE_Chassis = PXIE_Chassis (1);
end
% Open the connection to the VISA object.
fopen(PXIE_Chassis);
% Query the PXIE_Chassis.
response = query(PXIE_Chassis, '*IDN?');
disp('The *IDN query response:');
disp(response);
response = query(PXIE_Chassis, '*OPT?');
disp('The *OPT query response:');
disp(response);
% Replace this with a valid command for your device (read the programming
% guide section for examples)
command = ''
% Close the connection to the object.
```

---

## 11 Maintenance

---

To help ensure long, trouble-free operation:

- Always inspect fiber-optic connectors before using them and clean them if necessary.
- Keep the module free of dust.
- Store the module 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 module, power off the chassis immediately. Remove the module and allow to 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.

### 11.1 Annual calibration schedule

To ensure that the Switch 1200 PXle Series module is performing as expected, we recommend it is re-calibrated every 12 months. As an optical product will naturally degrade over time, it is important to periodically re-test the module, to confirm that it is working to specification.

All Quantifi Photonics products are calibrated during manufacture, and each product is shipped to the customer with a Calibration Certificate. On this certificate, the calibration date, as well as the next calibration due date are mentioned.

We recommend your product is returned for re-calibration before the listed due date, to ensure continued performance of the product. For re-calibration service information, or to send in a product for re-calibration service, email [support@quantifiphotonics.com](mailto:support@quantifiphotonics.com).

If the Calibration Certificate has been misplaced, or the calibration due date is not known, email [support@quantifiphotonics.com](mailto:support@quantifiphotonics.com).

## 12 Technical support

---

### 12.1 Contacting the Technical Support Group

To obtain after-sales service or technical support for this module, contact Quantifi Photonics.

The Technical Support Group is available to take your calls Monday to Friday, 9:00 a.m. to 5:00 p.m. (New Zealand Time).

#### Technical Support Group

Tel.: +64 9 478 4849

[support@quantifiphotonics.com](mailto:support@quantifiphotonics.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.

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

## 13 Warranty

---

### 13.1 General information

Quantifi Photonics Ltd. (Quantifi Photonics) warrants from the date of the original shipment (the Warranty Period) that this module will conform to specifications and will be free from defects in material and workmanship for the applicable Warranty Period. Quantifi Photonics also warrants that the equipment will meet applicable specifications under normal use.

#### **IMPORTANT**

The warranty can become null and void if:

- The unit has been tampered with, repaired, or worked upon by unauthorized individuals or non-Quantifi Photonics personnel.
- The warranty sticker has been removed.
- The unit has been opened, other than as explained in this guide.
- The unit serial number has been altered, erased, or removed.
- The 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 QUANTIFI PHOTONICS BE LIABLE FOR SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES.

For full warranty terms and conditions, please visit [www.quantifiphotonics.com](http://www.quantifiphotonics.com).

### 13.2 Liability

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

Quantifi Photonics shall not be liable for damages resulting from improper usage, transportation or unauthorized modification of the module, its accompanying accessories and software.

### 13.3 Exclusions

Quantifi Photonics 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 Quantifi Photonics 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 Quantifi Photonics.

### 13.4 Certification

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

### 13.5 Service and repairs

To send any equipment for service, repair or calibration please contact the Technical Support Group.

# Test. Measure. Solve.

Quantifi Photonics is transforming the world of photonics test and measurement. Our portfolio of optical and electrical test instruments is rapidly expanding to meet the needs of engineers and scientists around the globe. From enabling ground-breaking experiments to driving highly efficient production testing, you'll find us working with customers to solve complex problems with optimal solutions.

**To find out more, get in touch with us today.**

**General Enquiries**  
**Technical Support**  
**Phone**  
**North America**

[sales@quantifiphotonics.com](mailto:sales@quantifiphotonics.com)  
[support@quantifiphotonics.com](mailto:support@quantifiphotonics.com)  
+64 9 478 4849  
+1-800-803-8872



[quantifiphotonics.com](https://www.quantifiphotonics.com)

**QUANTIFI**  
**PHOTONICS™**