SCPI Python Example | LUNA 6415

Many regions prohibit the disposal of WEEE (Waste Electrical and Electronic Equipment) in the normal waste stream, to comply with the Restriction of Hazardous Substances (RoHS) released into the environment. Please contact your local waste authority for instructions on proper recycling of the electronic product(s) described in this *User Guide*.

SCPI Python Example *LUNA 6415*

Luna Innovations Inc.
3155 State Street
Blacksburg, VA 24060
Phone: (866) 586-2682
Fax: (540) 961-5191
E-mail: solutions@lunainc.com
Web: www.lunainc.com

Table of Contents

1 Introduction

This document provides the user information about using SCPI commands to operate the Luna 6415 remotely using Python.

A Python script is provided that provides an example of how to connect, query and set several properties, take measurements as well as plot graphs.

The purpose of this sample is to provide a source code example showing how remote Luna 6145 application can be achieved by using SCPI commands with one of the popular scripting languages, Python. The sample is not intended to be a fully featured program for use in a production environment. Users should write their own Python test application code based on this example.

This document assumes that the user already has some familiarity with the following areas: Luna 6415 operation, SCPI commands and Python.

1.1 Luna 6415 SCPI support

Luna 6415 Users Guide has a full coverage on the proper methods to set up and use the Luna 6415 application software, as well as more detailed information on Luna's SCPI support. For the user's convenience, several relevant paragraphs from the Users Guide are presented in the below in this section to give the users a general picture on how the SCPI is implemented in the Luna 6415.

> *SCPI commands may be sent to the controller over a network interface using the TCP/IP protocol. The IP address of the controller is displayed in the "About" settings page for your reference.*
>
> *Port 5025 is used to communicate with the controller over a raw TCP/IP socket connection. Use this port when sending SCPI commands from most all programs since it is more compliant with the SCPI standard than port 5024.*
>
> *Port 5024 is used to communicate with the controller using a telnet client. Use this port when you are sending SCPI commands manually as it has several advantages over port 5025 for manual use (Refer to the User Guide for more details).*
>
> *The following are a few example of command sequences to help demonstrate the use of SCPI of the Luna 6415.*
>
> *Keep in mind that it might be a good idea to follow each SCPI command or query with :SYST:ERR? to determine if an error was generated during the previous program message unit. For the sake of brevity, these command sequences examples don't demonstrate the use of the :SYST:ERR? query.*

1.1.1 *Example 1: Take Sequential Measurements*

```
DEL REFL          # Use REFLection mode, not TRANsmission mode
LENG 20           # Use 20 meter mode, not the 50 meter or 100 meter modes
GIND 1.4682       # Set the group index to 1.4682
MEAS:RL? 3,0.05 # Take a measurement and get the return loss at a distance of 3m with a width of 0.05m
```

```
MEAS:RL? 3ft      # Take a measurement and but get the return loss at 3ft (still with a width of 0.05m)
MEAS:OFDR?        # Take a measurement and return the amplitudes from the OFDR plot
MMEM:STOR ofdr,“test1”     # Store the measurement data to the file “test1.ofdr”
MMEM:STOR tsv_ose,”test1”# Store all the measurement data to the file “test1.tsv”
```

1.1.2 *Example 2: Get Multiple Results from a Single Measurement*

```
DEL REFL          # Use REFLection mode, not TRANsmission mode
LENG 20           # Use 20 meter mode, not 50 meter or 100 meter modes
GIND 1.4682       # Set the group index to 1.4682
CONF:RL DEF,0.05          # Set the return loss width to 0.05 meters
CONF:IL DEF,0.2   # Set the insertion loss width to 0.2 meters
CONF:SPEC DEF,0.5         # Set the spectral cursor width to 0.5 meters
INIT              # Start a measurement
FETC:OFDR?        # Wait for the measurement to complete and get amplitudes in dB from the OFDR plot
FETC:DIST?        # Get distances in meters from the OFDR plot
FETC:SPEC? 1      # Get the return loss values in dB from the spectral plot with the spectral cursor at 1m
FETC:WAV?         # Get the wavelengths in nm from the spectral plot
FETC:EVEN?        # Get the event table
FETC:RL? 1.2      # Get the return loss at a distance of 1.2 meters from the unit
FETC:RL? 3ft      # Get the return loss at a distance of 3 feet from the unit
FETC:IL? 3.4m     # Get the insertion loss at a distance of 3.4 meters from the unit
```

1.1.3 *Example 3: Get a Subset of the Measurement Data*

```
DEL REFL          # Use REFLection mode, not TRANsmission mode
LENG 20           # Use 20 meter mode, not 50 meter or 100 meter modes
INIT              # Start a measurement
CONF:OFDR 0, 1.5, 2.5  # Configure FETC:OFDR? and DIST? queries to return data from 1.5m to 2.5m
FETC:OFDR?        # Wait for the measurement to complete and get amplitudes in dB from the OFDR plot
FETC:DIST?        # Get distances in meters from the OFDR plot
```

Please refer to the User Guide for a complete list of Luna 6415 supported SCPI commands.

2 SCPI Python Sample Script (LwaScpi.py)

The provided Python sample, LwaScpi.py is used to demonstrate how to perform a variety of tasks via the SCPI interface. The list of tasks are:

- Connect to Lightwave Analyzer
- Get device identification string
- Query group index
- Set Group Index
- Make sure group index has changed
- Initiate a new measurement

- Get return loss at location 1.2 meters
- Get y-axis OFDR data for measurement
- Get x-axis OFDR data for measurement
- Plot OFDR data
- Plot OFDR data from 1.5m to 2.5m
- Show current spectral cursor position and width
- Set spectral cursor to location = 1.0 m, width = 0.5m
- Get spectral data and wavelength

2.1 Setup

In order to use the plotting features in the sample script the matplotlib library needs to be installed. It is assumed that Python and pip are already installed on the system that will be running the script. The system connecting to the controller can be either a Linux system or Windows system. To install the matplotlib library simply enter the following command from the terminal window:

sudo pip install matplotlib

Using the 'sudo' prefix would only be for a Linux system. Windows does not need that keyword. To verify the install was successful execute the following commands from the Python prompt:

```
import matplotlib
matplotlib.__version__
```

The version at the time this document was created was 3.5.1. Any version after this should work as well.

Before running this sample, the user needs to input the correct controller computer IP address that is displayed in main application's "About" settings page (Figure). The Port # can be either 5025 or 5024.

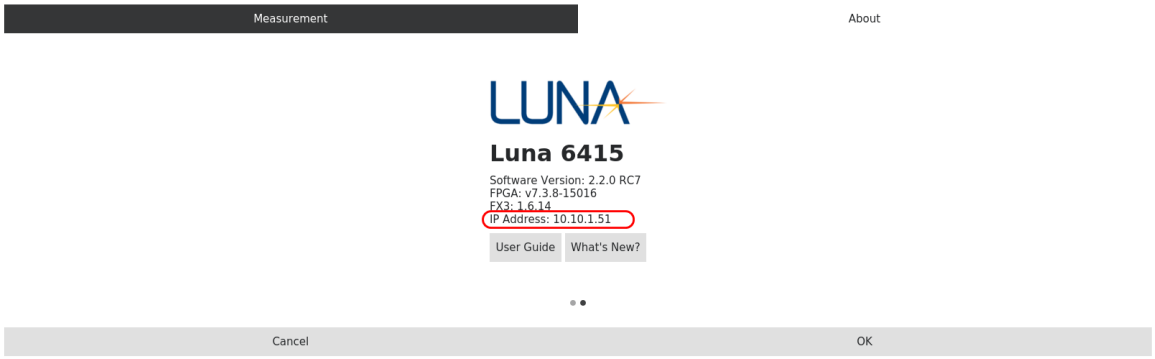

Figure 1. IP Address displayed in "About" page

The user can set the IP address and Port # within the Python script. They can be set on the following lines:

 # IP address of laptop controlling the Luna Lightwave Analyzer
 # 127.0.0.1 is the local machine
ip_addr = '127.0.0.1'
 # TCP/IP port for SCPI connection. Use 5025 for automated
 # connections, or 5024 for a more verbose telnet connection
ip_port = 5025

Use the '127.0.0.1' address is you are going to run the LwaScpi.py script on the computer that is acting as the controller for the 6415 unit. Otherwise use the IP address displayed in the "About" settings page.

2.2 Running LwaScpi.py script

The easiest way to run the LwaScpi.py sample script is from the command line. In Linux just run the python command (this can be just 'python' or 'python3' or 'python3.8', however it is setup on the computer) with the script name as the only command line parameter.

Ex: python LwaScpi.py

For Windows, run the script from the command prompt (cmd.exe) or the PowerShell by just typing the name of the script, LwaScpi.py.

The script can also be run from any Integrated Development Environment (IDE) that supports Python development, i.e. Visual Studio Code, Eclipse with PyDev, IDLE, etc.

3 Technical Support

If you should have any problems with or questions about the information contained in this document, please don't hesitate to contact our technical support staff via one of the following methods:

Luna Innovations Inc.

3155 State Street

Blacksburg, VA 24060

Toll-Free Support: (866) 586-2682

Fax: (540) 961-5191

E-mail: solutions@lunainc.com

Web: www.lunainc.com