rgb lasersystems

LEADING PHOTONICS

Software Development Kit

Quick Start Guide

Quick Start Guide
RGB Lasersysteme GmbH
Software Development Kit
Version: 1.1.1
Date: July 29, 2012

Product specifications and descriptions in this manual are subject to change without notice. RGB Lasersysteme GmbH will not be responsible for errors and omissions in this manual or for direct or incidental damages in connection with the use of this device or information.

© 2012 RGB Lasersysteme GmbH
Donaupark 13
93309 Kelheim
Germany

Telephone: +49 9441 1750 33 – 0
Website: http://www.rgb-laser.com
E-Mail: sales@rgb-laser.de

Contents

1 Software Development Kit

The RGB Lasersystems Software Development Kit (SDK) can be used to control the spectrometer and take spectra from your own custom software. It includes a Windows Dynamic Link Library (DLL) for the .NET framework, documentation and sample code. It also includes preliminary support for controlling laser modules.

1.1 Prerequisites

The system requirements are the same as for the application software Waves:

- Windows 7, Vista or XP (32 or 64 bit)
- .NET Framework Version 3.5 (available on the CD and as a free download)
- USB 2.0

You can use the Qwave SDK with any programming language that can connect to .NET DLLs. This includes C#, Visual Basic .NET, C++/CLI, Delphi, LabView, Matlab and Mathematica. If you are using a programming language that can connect to COM, but not to .NET libraries, you may need to use Microsoft's Assembly Registration Tool (`Regasm.exe`).

You should have a basic understanding of the concepts of object-oriented programming in order to understand the structure and functionality of the SDK library.

1.2 Getting started

In order to get familiar with the SDK quickly, we suggest to:

1. Read the next chapter 1.3 to understand the structure of the SDK library.
2. Read the basic steps required in order to take a spectrum (chapter 1.5).
3. Study the sample projects included in the SDK.
4. Add the SDK library to your own project (chapter 1.4).
5. Implement the basic functionality of taking a spectrum in your project.
6. Implement advanced features as required.

1.3 Classes Overview

The SDK library contains classes for controlling a spectrometer or laser as well as abstract base classes for various types of measurement and control devices. To control a Qwave or Qmini spectrometer, please use the `RgbSpectrometer` class. For a Qstick, please use the `Qstick` class.

So the main class that you need in order to communicate with the spectrometer is:

```
    RgbSpectrometer or Qstick
```

which is a descendant of the `ExtendedSpectrometer` class,
which is a descendant of the `Spectrometer` class,
which is a descendant of the `Device` class.

The `Device` class includes basic features used by all kinds of measurement and control devices. The `Spectrometer` class is a base class for a simple spectrometer. `ExtendedSpectrometer` is a base class for a spectrometer with more features. `RgbSpectrometer` implements all the features specified in the abstract base classes in order to control a Qwave spectrometer.

The library also includes the `SimulatedSpectrometer` class, that may be used instead to test your software in case you don't have a real device available, just like choosing "Simulated Spectrometer" in Waves. However, this class is currently only a descendant of the `Spectrometer` class and does not support the additional features in `ExtendedSpectrometer`.

You can find the full documentation for all relevant classes in the SDK documentation contained in the `RgbDriverKit documentation.chm` help file.

1.4 Adding the SDK library to your project

The main library to connect to your spectrometer is in the `RgbDriverKit.dll` file. Please copy it to your main project folder. The Qwave device driver for Windows, which is installed during the installation of Waves, must also be present.

Visual Studio

If you are using one of Microsoft's languages, please add the library in Visual Studio to your project references (in the Solution Explorer window). Then import the `RgbDriverKit` library to your source code files (for example in C# or C++: "`using RgbDriverKit;`"). For enhanced IntelliSense support in Visual Studio, you may also want to copy the `RgbDriverKit.xml` file into the same folder as the library.

LabVIEW

In order to use a .NET DLL in LabVIEW, you need to:

- Use the .NET palette (Connectivity -> .NET) to find all of the functions available.
- Use a constructor node in order to instantiate a class within the .NET assembly.
- Use property nodes and invoke nodes to access properties and methods of the class by wiring in the class reference from the constructor node to the property or invoke node.

Specific LabVIEW VIs for controlling the Qwave spectrometer are planned for a later release of the SDK.

Other languages

For more information on how to connect to the library from other programming languages, please try a Google search for something like "use .NET DLL in [your language]".

For example, users of Matlab may find useful information at:
`http://www.mathworks.de/help/techdoc/matlab_external/brpb5k6.html`

1.5 Taking a spectrum

First you need to instantiate an `RgbSpectrometer` or `Qstick` object to control your spectrometer. You could use the class constructor, of course, but in most cases it'll be easier to search for spectrometers by calling the static `SearchDevices()` method and then choose from one of the returned objects, which are actual instances of the `RgbSpectrometer` or `Qstick` class.

Then, the steps required for taking a simple spectrum are:

1. Open the connection to the device using the `Open()` method.
2. If you need the wavelengths, get an array containing the wavelengths of each pixel with `GetWavelengths()`.
3. Set the exposure time using the `ExposureTime` property.
4. Start the exposure with the `StartExposure()` method.
5. Wait (or do something else) while the `Status` property is `SpectrometerStatus.TakingSpectrum`.
6. Read the spectrum with the `GetData()` method.

The full documentation for these methods and properties can be found in the `RgbDriverKit documentation.chm` help file.

All methods and some properties may throw an exception as described in the SDK documentation help file. Always make sure that these exceptions are properly handled by placing these methods inside a `try{} catch{}` block.

1.6 Deploying your project

If you would like to deploy your project to other users, please:

- Include the library file `RgbDriverKit.dll`
- Make sure that the .NET framework 3.5 is installed
- Make sure that the device driver for the spectrometer is installed

The device driver is normally installed during the installation of Waves. If you would like to use the spectrometer without installing Waves, you can find the files for the device driver in the "Qwave driver" or "Qstick driver" folder in the program folder of a Waves installation.

2 Technical Support

2.1 Getting Help

If you have any questions, comments or requests concerning your Qwave spectrometer or the Waves software, please go to

`http://www.rgb-laser.com`

and click on “Support” on the menu bar.

There you’ll also find details on how to contact our Technical Support Team via e-mail or telephone.

2.2 Software Updates

Software Updates are expected to be released about twice a year. To check for updates, you can choose “Check for Updates” from the Help menu or go to the “Support” section on our website www.rgb-laser.com.