SpectraSorter User Manual

Aaron Ponti & Todd Duncombe *last updated September 13th, 2021*

The SpectraSorter is a C#-based software application to enable rapid UV-VIS detection using Ocean Optics Spectrophotometers, data signal processing and visualization, and dynamic communication of external triggers via an Arduino microcontroller. We designed it for high-throughput analytical chemistry applications that require real-time analysis of optical spectra to inform subsequent sample sorting. Hence its name, the “SpectraSorter.”

This user manual describes how to install and operate SpectraSorter, and details its major features.

For ease of comprehension we used the following formatting throughout:

* **A location in the software application (e.g., a tab or module) are in a bold grey text.**
* “Sub section titles or checkboxes” are in quotation marks.
* Parameters in a blue text.
* *User notes or hints are in italics.*

Hardware & Compatibility

This application was developed for and tested using an [Ocean Optics FX Spectrophotometer](https://www.oceaninsight.com/products/spectrometers/high-speed-acquisition/ocean-fx) and an Arduino MEGA 2560 microcontroller. As Ocean Optics use a common driver for all of their spectrophotometers, the [OmniDriver](https://www.oceaninsight.com/products/software/drivers/omnidriver/?qty=1), this software application should be broadly compatible.

Acknowledgements

This platform was originally based on FX Streamer, as software application provided to us by Ocean Optics: Lischtschenko, Oliver, Ocean Optics; private communication on OBP protocol, 2018.

Contents

[Basic Setup 3](#_Toc82420820)

[Acquisition Settings 5](#_Toc82420821)

[**Output** – Dropdown Selection 5](#_Toc82420822)

[**Communication** – Acquisitions Settings 5](#_Toc82420823)

[Export / Import SpectraSorter Settings 6](#_Toc82420824)

[Operating Statistics 6](#_Toc82420825)

[Output Files 6](#_Toc82420826)

[**Saving** Options 6](#_Toc82420827)

[Data File – “.csv” 7](#_Toc82420828)

[Configuration File – “.xml” 7](#_Toc82420829)

[Other Fields 7](#_Toc82420830)

[**Reference** Spectra 8](#_Toc82420831)

[“Dark Spectrum” 8](#_Toc82420832)

[“Enable Lamp” 8](#_Toc82420833)

[**Reference** Spectrum Acquisition 8](#_Toc82420834)

[**Single** Reference Spectrum 8](#_Toc82420835)

[**Accumulated** Reference Spectrum 8](#_Toc82420836)

[**Dynamic** Reference Spectrum 9](#_Toc82420837)

[Signal Processing and Wavelength Management 10](#_Toc82420838)

[“Filtering” 10](#_Toc82420839)

[Wavelength Hub 11](#_Toc82420840)

[“Select Range” 12](#_Toc82420841)

[“Triggering” 12](#_Toc82420842)

[Data Plotting 13](#_Toc82420843)

[Window Size 13](#_Toc82420844)

[Triggering Display Options 13](#_Toc82420845)

[External Triggering / Arduino Configuration 14](#_Toc82420846)

[**Parameters** 14](#_Toc82420847)

[**Performance** 14](#_Toc82420848)

[Shortcuts 15](#_Toc82420849)

# Basic Setup

**Install & Setup SpectraSorter**

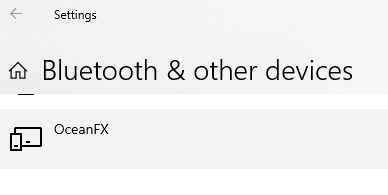
**Install latest stable release**

An installer for the latest, stable version of SpectraSorter can be downloaded from <https://github.com/SpectraSorter/SpectraSorter/releases/latest>.

**Build from source**

1. Download / Install Visual Studio 2019 with .NET Desktop development workload ([link](https://visualstudio.microsoft.com/vs/features/net-development/)).
2. Download / Install Microsoft .NET Framework 4.8 offline installer for Windows ([link](https://support.microsoft.com/en-us/topic/microsoft-net-framework-4-8-offline-installer-for-windows-9d23f658-3b97-68ab-d013-aa3c3e7495e0)).
3. Install the extension “Microsoft Visual Studio Installer Projects” from the menu “Extensions” > “Manage Extensions”.
4. Open “SpectraSorter.sln” from the “src” folder in Visual Studio.
5. In the “Solution Explorer”, right-click on the SpectraSorter solution and hit “Restore NuGet Packages”.
6. Switch to “Release” in the Solution Configurations pull-down in the Toolbar and run Build > Build Solution.
7. In the “Solution Explorer”, right-click on the SpectraSorter project and hit “Set as StartUp Project”.
8. In the Toolbar hit “Start” to initiate the SpectraSorter program.

*Note: The SpectraSorterSetup project can be used to create a redistributable Windows Installer. Right-click on SpectraSorterSetup and select Build. Once the build is complete, right-click on the project again and choose Open Folder in File Explorer. The setup files can be used to install SpectraSorter on any machine. Also note that the project is configured to package the Release version of the SpectraSorter (see point 3 above).*

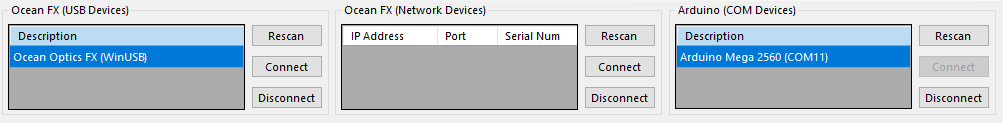
**Install Drivers for Ocean Optics Spectrophotometer**

The SpectraSorter is compatible with the [OmniDriver](https://www.oceaninsight.com/products/software/drivers/omnidriver/?qty=1) from Ocean Optics. Which they use to power all of their spectrophotometers. Install the driver for your spectrophotometer and make sure your system can detect it when it is plugged in.

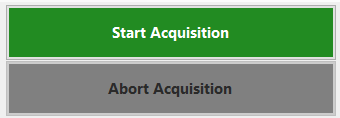
**Setup Arduino Microcontroller**

From the repository, find the Arduino control software at “…SpectraSorter/SpectraSorter\_Trigger/SpectraSorter\_Trigger.ino.” Upload it to your Arduino microcontroller using an [Arduino IDE](https://www.arduino.cc/en/software).

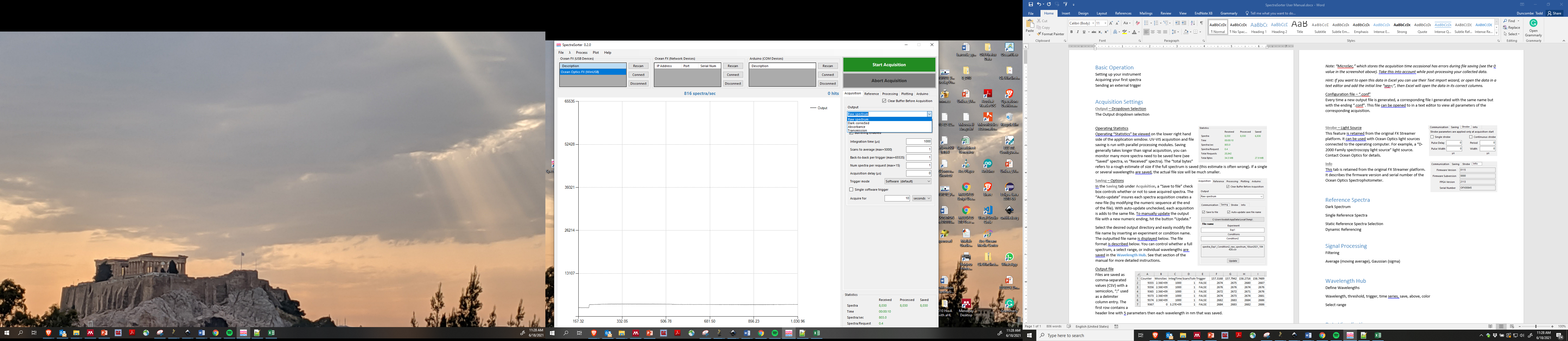
**Open SpectraSorter and Connect Hardware**

Connect your Ocean Optics Spectro-photometer via USB or Ethernet cable. Scan for the device, select the device of interest then hit connect.

The Arduino device is only required when using external triggers. It is not required when performing spectroscopy. Connect it via USB, scan for the device in the “Arduino (COM Devices)” Window and connect.

**Perform Acquisition**A big button at the top right of SpectraSorter allows the user to start or abort an acquisition. This can be done via the shortcut CTRL+R (start) and ended with CTRL+X (abort). Go to **Help > Shortcuts** for more.

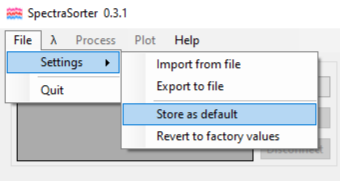
# Acquisition Settings

******Output** – Dropdown Selection  
The **Output** dropdown selection controls the type of data that is saved during acquisitions. The default is “Raw Spectrum,” which saves the absolute intensity values recorded by the spectrophotometer. The other options, “Dark Corrected,” “Absorbance,” and “Transmission,” save processed data that can be generated after a dark spectrum and a reference spectrum is acquired. See the **Reference Spectra** section of this manual for more details. Also notice that for both “Absorbance” and “Transmission” if the dark spectrum is not present, it will be ignored, and if the reference spectrum is not present, the absorbance/transmission will be set to 0.

### **Communication** – Acquisitions Settings

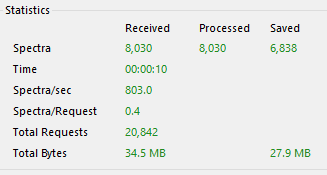
These are critical parameters that dictate how the spectrophotometer itself operates and return information to the software. They are described below in their relative order of importance.

* **Integration time (µs):** Defines the time over which the sensors collect information for a single spectrum. The minimum integration time for our FX Spectrophotometer was 10 µs, but we typically operated the system from between 250 µs to 5000 µs, depending on the application.
* **Acquire for (seconds, minutes, hours, spectra):** Defines the time of the run in terms of seconds, minutes, hours, or spectra. Be wary that this can produce many GB of data if **Saving** is enabled for a long time, make sure your settings are correct and that you have sufficient space.
* **Scans to average:** Scans to average determines how many scans should the spectrophotometer average prior to returning a spectrum to the user. This features is used to reduce noise, we typically do not average scans.
* **Num spectra per request (max=15):** The number of spectra returned for each request to the spectrophotometer. This parameter presents tradeoffs to users. As each request step to the spectrophotometer has an associated delay, requesting 15 at a time can significantly speed up the Spectra per second acquisition. But, it will introduce a variable delay between each spectra’s individual acquisition and when the software platform will process it (and potentially trigger an event). We typically use 1 spectra per request.
* **Back-to-Back per trigger:** For each trigger event received, whether hardware or software trigger, the spectrometer will retrieve this many spectra back-to-back at its maximum acquisition rate of about 1 spectrum every 222 microseconds. This feature requires that buffering is enabled.
* **Acquisition delay:** This delay setting allows for a user-specified delay in spectral acquisition after a trigger is received by the spectrometer.
* **Trigger mode:** Trigger mode should be left on “Software” for correct analysis behavior.

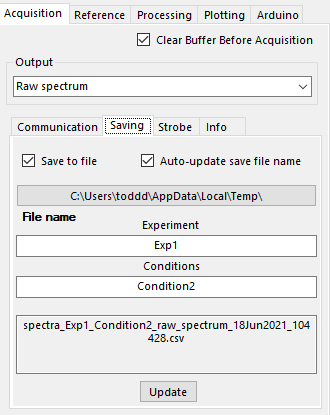
Export / Import SpectraSorter Settings  
All settings for SpectraSorter can be saved externally with “Export to file” as an .xml file. Later it can be reloaded with “Import from file.” Further, any settings can be stored as default whenever SpectraSorter is loaded with “Store as default.” To revert back to the initial blank default settings, select “Revert to default.”

After every acquisition is run, an identical .xml settings file is created in the saving directly (whether saving is turned on or not). These settings can also be imported using this interface.

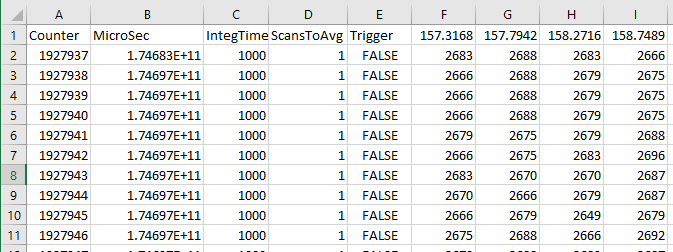
This is very useful to record the spectrophotometer settings for a given experiment. Further, if the program happens to crash, you can quickly retain your desired setup.

Operating Statistics  
Operating “Statistics” can be viewed on the lower right hand side of the application window. UV-VIS acquisition and file saving is run with parallel processing modules. Saving generally takes longer than signal acquisition. You can monitor how many more spectra need to be saved here (see “Saved” spectra, vs “Received” spectra). The “total bytes” refers to the size of spectral data saved (generally it is ~40% of the final CSV file size).

## Output Files

******Saving** Options  
In the **Saving** tab under **Acquisition**, a “Save to file” check box controls whether or not to save acquired spectra. The “Auto-update” ensures each new acquisition creates a new file (by modifying the numeric sequence at the end of the file). With auto-update unchecked, each acquisition is added to the same file. To manually update the output file with a new numeric ending, hit the button “Update.”

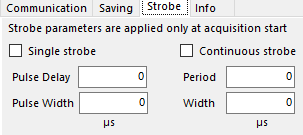
Select the desired output directory and modify the file name by inserting an experiment or condition name. The outputted file name is displayed. The file format is described below. You can control whether a full spectrum, a select range, or individual wavelengths are saved in the **Wavelength Hub**. See that section of the manual for more detailed instructions.

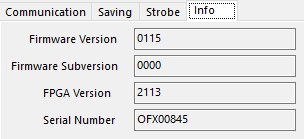
Data File – “.csv”  
Files are saved as comma-separated values (CSV) files with a semicolon, “;” used as a delimiter column entry. The first row contains a header line with five parameters then each wavelength in nm that was saved.

*Hint: If you want to open this data in Excel you can use their Text import wizard, or open the data in a text editor and add the initial line “sep=;”, then Excel will open the data in its correct columns.*

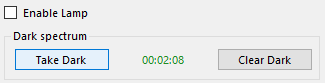
Configuration File – “.xml”  
Every time a new output file is generated, a corresponding file is generated with the same name but with the ending “.xml”. This file can be opened to in a text editor to view all parameters of the corresponding acquisition. These settings can also be imported into SpectraSorter using the “File > Settings > Import from file” from the top toolbar.

## Other Fields

**Strobe** – Light Source  
This feature is retained from the original FX Streamer platform. It can be used with Ocean Optics light sources connected to the operating computer. For example, a “D-2000 Family spectroscopy light source” light source. Contact Ocean Optics for details.

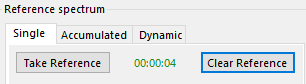
**Info**  
This tab is retained from the original FX Streamer platform. It describes the firmware version and serial number of the Ocean Optics Spectrophotometer.

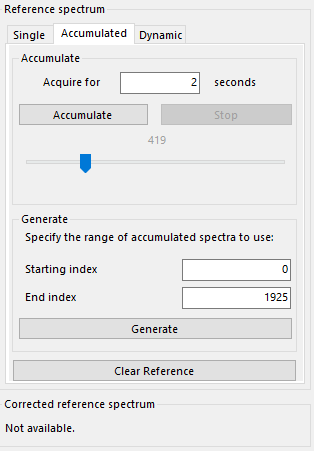
# **Reference** Spectra

“Dark Spectrum”  
The “Dark Spectrum” establishes the relative noise at a given acquisition setting and environment. This should be performed every time the acquisition settings are changed or the environment appreciably changes. Turn off your light source then hit button “Take Dark.” The corresponding dark spectrum will appear in the plot region. Make sure this looks correct, occasionally there will be residual spectra still in the cache and it will not properly load the new dark acquisition. If this is the case, click “Take Dark” several more times until the correct dark spectra is acquired. The timer indicates how long ago the spectrum was acquired.

“Enable Lamp”  
This feature is retained from the original FX Streamer platform. It can be used with Ocean Optics light sources connected to the operating computer. For example, a “D-2000 Family spectroscopy light source” light source. Contact Ocean Optics for details.

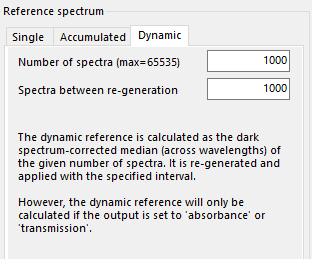
## **Reference** Spectrum Acquisition

**Single** Reference Spectrum  
The “Reference Spectrum” establishes the reference spectra from which “Absorbance” or “Transmission” optical measurements are calculated. Ensure your light source is on and the correct reference sample is in place, then hit “Take Reference.” The corresponding Reference Spectrum will appear in the plot region. Make sure this looks correct, occasionally there will be residual spectra still in the cache and it will not be the most recent acquisition. If this is the case, click “Take Reference” several more times until the correct spectra is acquired.

**Accumulated** Reference Spectrum  
In many cases it may not be possible to place the desired reference sample in front of the sensor. For example, if samples are moving quickly past the sensor. We developed this **Accumulated** module to account for this.

First, setup your acquisition as desired, build your desired **Wavelength Hub**, and take a “Dark Spectrum.” Then set your accumulation window in the “Acquire for \_ seconds,” box. Your reference spectra/spectras should appear over this time period. Then hit “Accumulate.” The spectrophotometer will run until it accumulate all spectra over that time window and store it in cache. We recommend picking a time window that accumulates less than 10k spectra to prevent the risk of the software locking up.

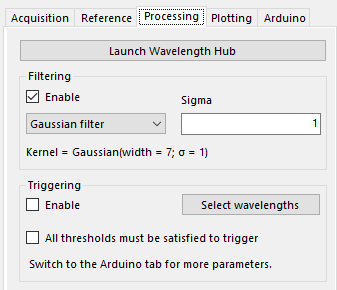
Once you accumulated, you can view the spectra individually or as a time series for the wavelengths you previously defined in your **Wavelength Hub**. From the top of your window go to **Plot > Accumulated spectra** or **Plot > Accumulated time series**. You can search through the accumulated spectra by dragging the blue maker on the scroll bar seen below. Once you have identified your desired reference spectra/spectra’s insert their starting and ending index in the “Generate” subsection, then hit Generate. If you selected a range, the average spectra over that range will be used. Then, the reference spectrum will be displayed.

**Dynamic** Reference Spectrum  
In cases where the reference spectrum may change over time, it may be of interest to perform dynamic referencing. This feature calculates a reference spectrum as the median spectrum over a defined number of spectra and over a certain interval. This feature will only be of use if the desired reference spectra is present in front of the sensor over 50% of the time.

If you start a run using dynamic referencing you will still need to define an initial reference using the single or accumulated methods described previously.

*Note: The recalculation of a reference spectrum WILL INTRODUCE A SLIGHT DELAY during that recalculation period. So if you are at risk of under-sampling, you may not want to perform dynamic referencing.*

# Signal Processing and Wavelength Management

The **Processing** tab includes a link to the **Wavelength Hub** and settings related to data filtering and a triggering.

### “Filtering”

Data filtering can be enabled for each spectrum by clicking the “Enable” check box. Filtering is run by an “Average filter,” (i.e., a standard moving average) or a “Gaussian filter” (i.e., a Gaussian weighted moving average). The size of the moving area can be set in the window **prior** to starting a run. Indeed, please notice that enabling and disabling filtering during a run is allowed, but changing the filter kernel size can only be done in between acquisitions.

## Wavelength Hub

The **Wavelength Hub** is the most important module in operating the SpectraSorter. Here, particular wavelengths are defined for time series plotting, saving, and triggering. Prepare the hub prior to your run, as most features cannot be modified during a run. Once established consider saving this setting with **File > Settings > Export to File**.

**Wavelength (nm)**: Define the wavelength of interest.

**Threshold (output units)**: If you are performing triggering, define the threshold value at which triggering should occur. This is in units of the output (e.g., raw spectrum, dark corrected, absorbance, or transmission).

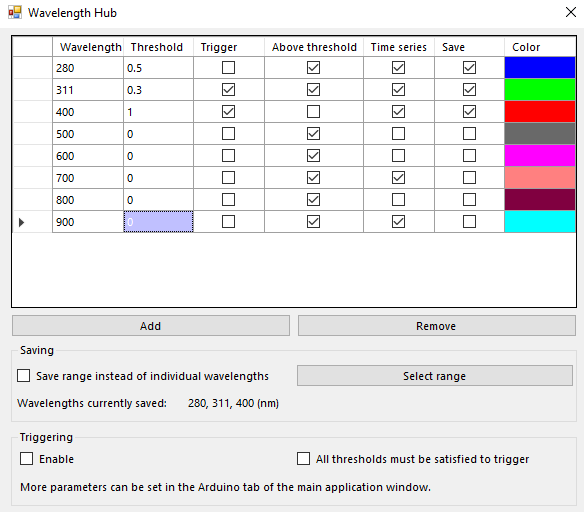
**Trigger (checkbox)**: If checked, this wavelength will be involved in triggering events, if triggering is globally activated.

**Above Threshold (checkbox)**: If checked while the trigger box is checked, triggering is performed above the threshold value. If unchecked while the trigger box is checked, triggering is performed when the wavelength is below this threshold. If the trigger box is unchecked this has no function.

**Time series (checkbox)**: If checked, this wavelength will appear in a time series plot.

**Save (checkbox)**: If checked and wavelength saving is enabled, this wavelength will be saved along with other wavelengths in the output file. For wavelength saving to be enabled, global saving must be enabled AND the box “save range instead of individual wavelengths,” must be unchecked.

**Color (button)**: Click the color icon and define the desire color for plotting this given wavelength.



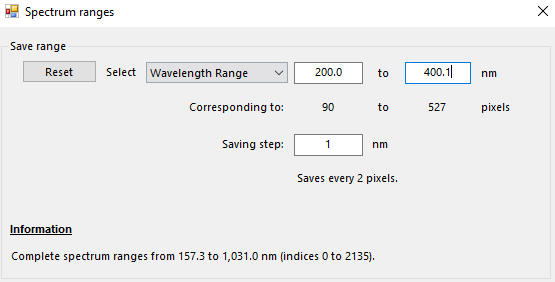
In the example above, the program would perform as follows:

Saving: 280, 311 and 400 nm (if saving is globally activated)

Triggering: 311 nm above 0.3, 400 nm below 1 (if triggering is globally activated)

Timeseries Plotting: 280 nm, 311 nm, 400 nm, 700 nm, and 900 nm.

### “Select Range”

Instead of saving wavelengths you can save the complete spectrum or a subset range of it. By clicking “Select Range,” in the **Wavelength Hub**. It opens a spectrum range selection interface. Here you can select a pixel or wavelength range as well as the step size over that range. To get back to full spectrum select “Reset.” To make sure the range is saved instead of individual wavelengths, make sure to select “Save range instead of individual wavelengths,” back in the **Wavelength Hub**.

### “Triggering”

Triggering refers to sending an external trigger via the Arduino microcontroller. This setting can only be enabled if an Arduino microcontroller is connected and detected by SpectraSorter, see the Basic Section of the manual for more details on that.

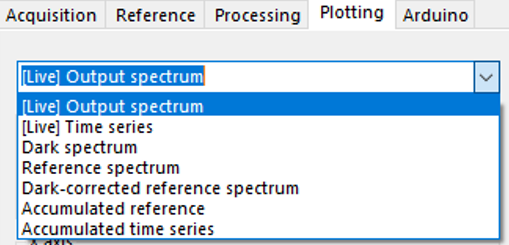
Triggering is be enabled by selecting “Enable”. The triggering thresholds as well as which wavelengths they apply to are defined in the **Wavelength Hub**.

#### Triggering Operator: AND or OR

The “All thresholds must be satisfied to trigger” checkbox makes triggering function as an AND operator. Meaning, a trigger will only be initiated if ALL wavelength thresholds defined in the **Wavelength Hub** are met. If it is unchecked, a trigger operates like an OR operator. A trigger will be initiated if ANY of the wavelength thresholds are met.

# Data Plotting

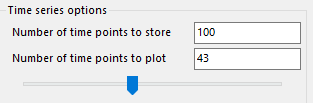
In the **Plotting** tab the user can control what data is displayed, the window size, and whether triggering thresholds and events are displayed.

In the dropdown menu the user can select: “[Live] Output Spectrum,” “[Live] Time Series,” “Dark Spectrum,” “Reference Spectrum,” “Dark-corrected Reference Spectrum,” or the “Accumulated reference” and “Accumulated time series” if the Static reference spectra method was used.

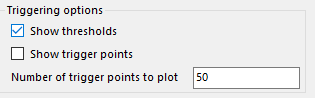
The Output Spectrum and Time Series data type is defined by the one selected in the **Acquisition** tab. The wavelengths defined in the Time Series are selected in the **Wavelength Hub**.

### Window Size

The “Y axis” can be auto scaled or manually adjusted for all plot types. For all spectrum plot types (e.g, [Live] Output Spectrum), the “X axis” is controlled by manually setting the desired wavelength range.

For all time series plot types (e.g., [Live] Time series), the “X axis” is defined by the number of spectra displayed. Before a run starts define the “Number of time points to store.” This sets a buffer size for storing the most recent spectral data and cannot be changed during a run. In practice it sets the maximal range of the “X axis” during a time series plot. The user can zoom in and view a smaller subset of the data by using the scroll bar to change the “Number of time points to plot.”

### Triggering Display Options

In “Triggering options,” the user can select whether or not to plot thresholds, triggering events, and how many trigger points to plot within the display window.

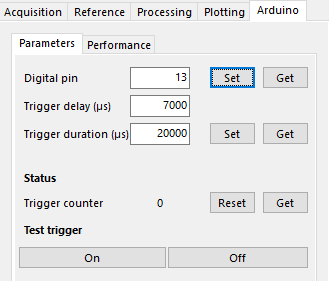
After selecting “Show thresholds,” thresholds will be plotted as a dotted line in the same color as the wavelength that they are thresholding for. In addition, they will be added to the display legend with an arrow up or down. The arrow indicates if the threshold is above or below.

“Show trigger points” plots vertical black lines along the time series when a triggering event occurs. The “number of trigger points to plot” sets the maximum number of trigger points to be displayed at a given time.

# External Triggering / Arduino Configuration

The **Arduino** tab allows users to define the parameters of an external trigger.

### **Parameters**

**Digital pin:** Sets the pin on the Arduino microcontroller which will output a 5V signal when a trigger occurs. Users should use the voltage difference between this pin on the microcontroller and its ground as the external trigger. For example, connecting it to a function generator.

**Trigger delay (µs):**  Sets the microsecond delay between when the trigger event is detected, to when it is sent to the microcontroller.

**Trigger duration (µs):**  Sets the microsecond duration for how long the microcontroller will send the 5V trigger.

**Trigger Counter:** Allows to the user to check how many trigger signals have been sent to the microcontroller since the counter was last reset.

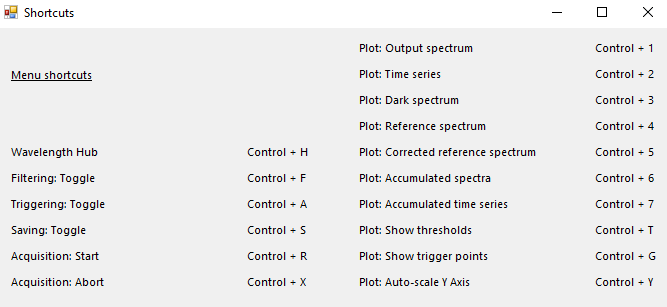
**Test Trigger:** The user can test the trigger by turning it “On” or Off.”

### C:\Users\toddd\AppData\Local\Microsoft\Windows\INetCache\Content.Outlook\963291Y1\arduino_speedtest.png**Performance**

The performance tab allows users to run an Arduino communication speed test. Take the “Average trigger time” into account when setting your trigger delay time.

# Shortcuts

Keyboard shortcuts can be found in **Help > Shortcuts**. The shortcuts as of June 18, 2021 are shown below.



A number of settings can be turned on or off from the base of the application window.

