



User's Guide | ODiSI-B

Version 5.2.1

**CLASS 1 LASER PRODUCT
IEC 60825-1, 2007
AND 21CFR1040.10**



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*.

Optical Distributed Sensor Interrogator Model ODiSI-B: User's Guide
ODiSI-B Software 5.2.0

© 2017

Luna

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 Glossary.....	5
2 System Overview	8
3 Quick Start Guide.....	10
3.1 ODiSI-B Control Software Setup.....	10
3.2 Touch to Locate.....	14
3.3 Measurement vs. Time	17
3.4 Plot to Picture	19
3.5 Data Logging	22
3.6 ODiSI-B Post Processor Software	30
4 Software Guide	35
4.1 Starting the Software	35
4.2 Menu Bar.....	37
4.3 Sensor.....	43
4.4 Instrument Control.....	45
4.5 Averaging	46
4.6 Data Logging	47
4.7 Measurement Display	53
4.8 Raw Data Post Processor.....	65
4.9 Measurement Streaming	70
5 Software Development Kit.....	80
5.1 Overview	80
5.2 Getting Started	80
5.3 LabVIEW Support.....	81
5.4 C/C++ Support.....	90
6 ODiSI Measurement Streaming Protocol.....	98
6.1 OMSP Structure	98
6.2 Metadata Messages	99
6.3 Measurement Message	101
6.4 OMSP Source Code Examples.....	104
7 Best Practices	105
7.1 System Operation Guidelines	105
7.2 Vibration Tolerance – Test Setup	105

7.3 Vibration Tolerance – Software Setup	105
7.4 Temperature Test Setup.....	106
7.5 Strain Test Setup.....	107
7.6 Strain Gradient Tolerance.....	107
8 Troubleshooting	108
8.1 Dropouts.....	108
8.2 Strain Relief.....	109
8.3 General Troubleshooting	110
8.4 Technical Support.....	112
Appendix A : Raw Data File and Trigger Logic.....	113
Appendix B : Expected Software Performance.....	115
Appendix C : DAC Setup.....	116
Appendix D : Measurement Theory.....	118

Caution 

Use of controls, adjustments, performance, or procedures other than those specified herein may result in hazardous laser radiation exposure and one or more safety protections may be impaired or rendered ineffective.

Warning 

The protection provided by the equipment may be impaired if the equipment is used in a manner not specified by the manufacturer, resulting in serious injury or death.

Warning 

The power cord is the main disconnect for this equipment. If it is necessary to ensure no power to the unit, remove the power cord.

1 Glossary

Acquisition Rate	This is the rate at which the ODiSI-B can acquire optical data from the connected fiber sensor.
Circular Raw Data File	Raw data logging file that has a fixed duration (# of scans) and a user set decimation rate. Once full, the oldest data collected is overwritten with new data. Logging data to this file can be controlled using the Event Trigger.
Configuration (Mode)	A set of Luna defined laser and software parameters that change the measurement capabilities of the ODiSI-B. Measurement parameters that change between configurations include: acquisition rate, maximum sensor length, gage length, and gage pitch.
Event Trigger	A software signal that can be initiated by pressing the Event button in the software or by sending a 5V rising edge electrical signal to the BNC port labeled “Event” on the back of the ODiSI-B. This triggers the start of a linear raw data file or triggers the countdown for a circular raw data file.
FPS	Acronym for Frames Per Second. This indicates the update rate of the ODiSI-B strain/temperature processing.
Gage Length	Length of the fiber used to calculate a single strain or temperature measurement.
Gage Pitch	Length of fiber between the centers of a single gage and its nearest neighbor. For the ODiSI-B this value is 50% of the gage length.
High-Definition Sensor	A fiber sensor that provides a Rayleigh scatter signature that the ODiSI-B can use to make distributed strain or temperature measurements.
High-Speed CFG Sensor	High-Speed Continuous Fiber Grating Sensor. These sensors use Bragg gratings written over their entire length to make strain measurements that are significantly more robust to environmental noise and vibration.
Linear Raw Data File	Raw data logging file that is written to once an Event Trigger is received and continues to log data until a Sample Trigger is received.

Map File	File used to store sensor specific information that correlates gage locations to points or segments of interest on the test article. This file is populated through the Touch to Locate process.
Plot to Picture	Software feature that allows plotting strain or temperature data as a heat map onto an image of the test article.
Processed Data	Data that has been converted into strain or temperature. This type of data is saved to tab delimited text files.
Profile	User defined region of the fiber sensor over which strain or temperature is computed. This region is a subset of the keyed section of the fiber sensor.
Raw Data	Optical data collected by the ODiSI-B. This data is saved to .odb files in a linear or circular manner.
Reference	Data containing the unique Rayleigh scatter pattern of a single fiber sensor. This is compared to measurement scans to calculate spectral shift.
RLC	Ruggedized LC connector. These connectors are used for the connection between the standoff cable and ODiSI-B as well as the remote module.
Sample Trigger	A software signal that can be initiated by pressing the Sample button in the software or by sending a 5V rising edge electrical signal to the BNC port labeled "Sample" on the back of the ODiSI-B. This signal triggers the end of a linear raw data file.
Screen Update Rate	The rate (in Hz) at which the ODiSI-B software can calculate strain or temperature and display it to the screen.
Sensor Key	File containing a measurement of a High-Definition Sensor's reference. All spectral shift measurements are made relative to the fiber in this state.
Sensor Length	Length of the sensing region of the fiber optic sensor.
Sensor Repository	Folder on the PC that contains sensor specific information such as sensor key files as well as map files. Default location is: C:\ProgramData\Luna Technologies\ODiSI-B\Sensors
Spectral Shift	The base processed data unit of the ODiSI-B. This is scaled to strain or temperature using the strain or temperature coefficients.

Standoff	Length of optical fiber placed between the ODiSI-B and the fiber sensor that does not count toward the system's sensor length. The ODiSI-B uses a 50m standoff with RLC connectors on each end.
Strain Coefficients	Polynomial coefficients used to scale the computed spectral shift value to units of microstrain.
Tare	File containing a set of spectral shift values to subtract from the processed data prior to converting the measurement into strain or temperature.
Temperature Coefficients	Polynomial coefficients used to scale the computed spectral shift value to units of Celsius (relative to sensor key).
Time Stamp	Recording of the time of a raw (optical) data acquisition by the ODiSI-B. Time stamps are displayed in UTC.
Touch to Locate	Process by which users can determine the fiber sensor gages that correspond to points or regions of interest on their test article.
Quality Factor	Metric that describes the strength of the cross correlation used to determine spectral shift for a single gage. This has a range of 0 – 1 where 1 is a perfect correlation, and 0 is no correlation.
Quality Threshold	A lower limit on the quality factor of the processed data. All measurement points with associated quality factor below this value are replaced with NaN.
Vibration Correction	Algorithm that removes the negative effects of a moving or vibrating standoff from the optical data prior to computing strain or temperature.
Vibration Correction Region	Region of sensing fiber that is used to compute the vibration present in the standoff. The first 20cm of PTFE jacketed sensing fiber is intended for this use. This is only utilized in High-Definition sensors.

2 System Overview

Luna Innovations' Optical Distributed Sensor Interrogator (ODiSI-B) uses swept-wavelength coherent interferometry to measure temperature and strain using optical fiber as the sensor. The ODiSI-B offers the ability to test a structure at numerous specific points of interest over an extended area.

Optical fibers undergo well defined physical changes as a result of changes in temperature and strain. The ODiSI-B is therefore, at any given time, able to measure the strain or temperature throughout the length of the fiber, at intervals as low as 0.64mm. By comparing measurements made at two different times the ODiSI-B calculates and displays the change in strain or temperature.

The ODiSI-B utilizes fiber optic sensors ranging from 1m to 20m in length to make distributed strain and temperature measurements of a test article. Due to the small diameter of the fiber (<0.2mm), the fiber can be routed into locations not usually possible with foil gages and thermocouples. The ODiSI-B has the ability to differentiate the sensing fiber from other fiber optic devices (patch cables, switches, etc.) and is therefore able to be reconfigured without a change in measurement data.

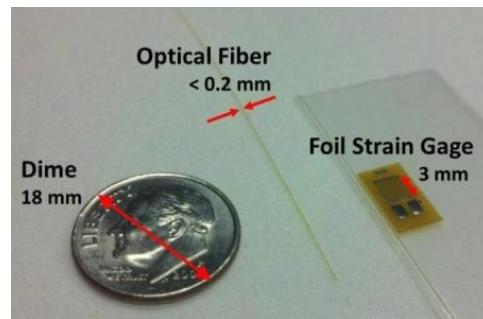


Figure 2-1 ODiSI-B sensor fiber is much thinner and longer than traditional foil strain gages



Figure 2-2 ODiSI-B measurement system. The ODiSI-B measures strain and temperature along the length of the fiber. Data is presented graphically, providing the user with unprecedented data density and resolution.

High-Speed CFG Sensors

High-Speed CFG (Continuous Fiber Grating) sensors allow the ODiSI-B to make measurements in the presence of significant environmental noise and/or vibration. These sensors are available in 2m and 5m lengths and can be interrogated using the 100Hz and 23.8Hz scan modes.

High-Definition Sensors

High-Definition sensors provide a low cost alternative to the High-Speed CFG sensors. These sensors are available in lengths of up to 20m and are ideal for making measurements in static and pseudo-static test environments.

3 Quick Start Guide

This section provides new users with a brief walkthrough that describes how to collect, save, and post process strain or temperature measurements using the ODiSI-B. For detailed descriptions of software features, proceed to Section 4: Software Guide.

Refer to the ODiSI-B Setup Guide for instructions regarding how to properly setup the ODiSI-B and PC hardware.

3.1 ODiSI-B Control Software Setup

3.1.1 Starting the Software

1. Launch the software by double clicking on the “ODiSI-B v5.2.0” desktop shortcut.
2. When the **Select Configuration** popup opens, choose a mode of operation from the dropdown menu and click **Select**.

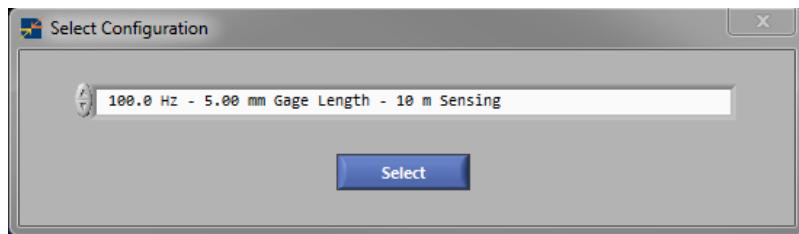


Figure 3-1 Select Configuration window

3. Once the splash screen has closed, the ODiSI-B control software will be accessible.



Figure 3-2 ODiSI-B Control Software

3.1.2 Sensor Installation

Sensors bought from Luna will include a flash drive with a key file for use by the ODiSI-B software. This sensor key needs to be installed in the **Sensor Repository** before the fiber sensor can be used.

1. Navigate to the sensor installation window: **Menu > Sensor > Install Sensor...**

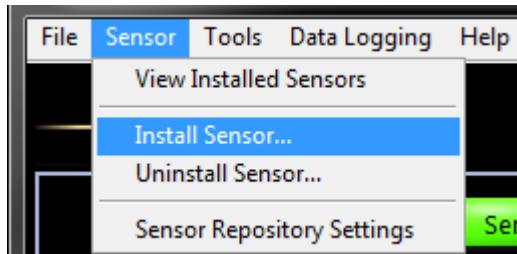


Figure 3-3 Install Sensor menu location

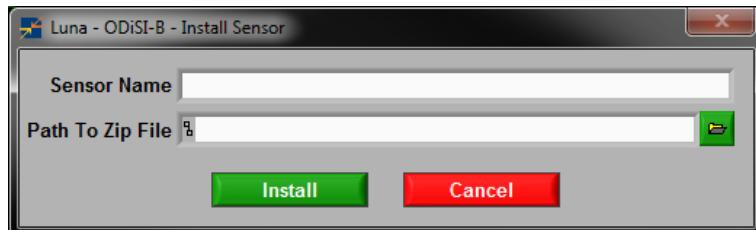


Figure 3-4 Install Sensor window

2. Enter a sensor name and then navigate to the zipped sensor key located on the flash drive included with the fiber sensor.
3. Click the **Install** button.
4. A popup will appear indicating that the sensor has been installed and the user can choose to install additional sensors. Install additional sensors at this time if desired by clicking **Yes** and repeating steps 2 and 3.

3.1.3 Measurement Setup

With sensor files installed, the ODiSI-B software is ready to be used for making strain or temperature measurements on the fiber sensor.

1. Align the optics by navigating to **Menu > Tools > Check Alignment**

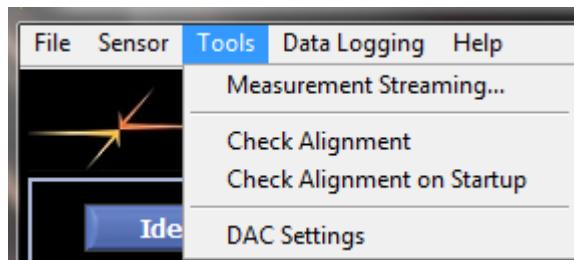


Figure 3-5 Check Alignment menu location

2. Follow the instructions in the popup windows while the system aligns itself.
- Note:** The alignment routine should be re-run if the 50m standoff cable or remote module has been re-routed or moved. The remote module can be affixed to the test structure to avoid the need for routine re-alignment.
3. Clean the fiber connector and then connect the desired fiber sensor to the remote module at the end 50m standoff cable. Ensure that the remote module selected matches the sensor type.
4. Click the **Identify Sensor** button and wait while the fiber sensor is identified.

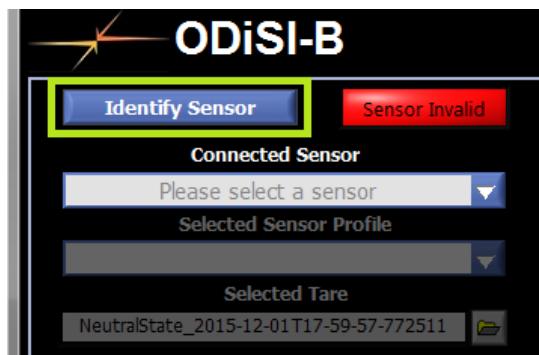


Figure 3-6 Identify Sensor button location

Note: If using a High-Speed CFG sensor, use the **Connected Sensor** dropdown menu to select it from the list of installed sensors. (The High-Speed CFG sensors can only be selected in 23.8Hz and 100Hz operating modes)

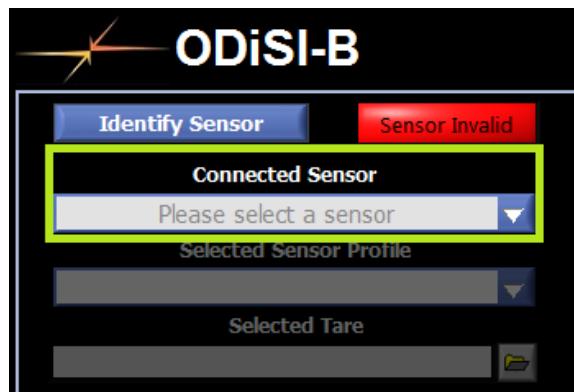


Figure 3-7 Connected Sensor dropdown menu

5. Click on the **Start Operation** button to begin taking measurements. At this Point, data will appear on the **Measurement vs. Length** graph.

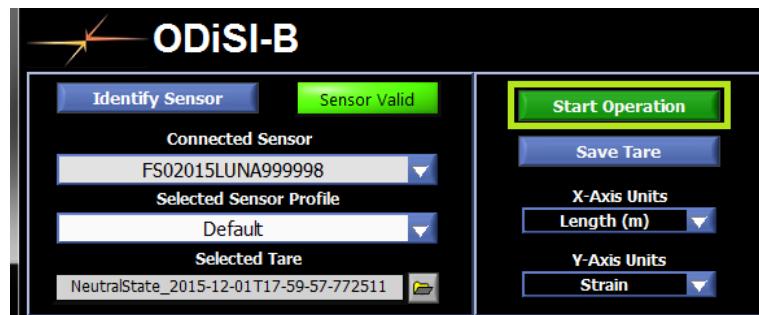


Figure 3-8 Start Operation Button location

6. If using a High-Definition sensor, ensure that the vibration correction section (orange data on plot) is not under strain.



Figure 3-9 Vibration Correction trace (orange highlight in graph) and Vibration Correction location control

7. Make small adjustments to the vibration correction location until any vibration correction errors disappear. This control is in the lower right corner of the screen. The control will be outlined in red if there are errors.
8. Take a new tare by clicking the **Save Tare** button. Enter a name for the tare into the popup window if desired and click **OK**.

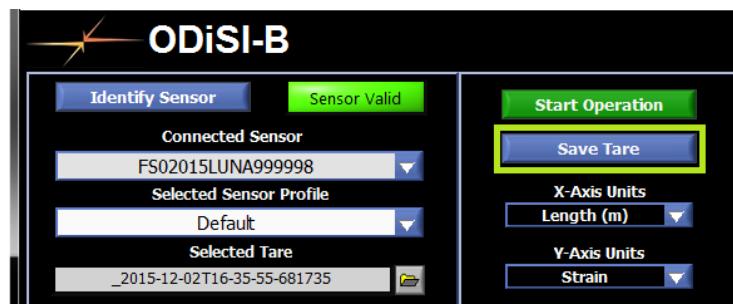


Figure 3-10 Save Tare button location

3.2 Touch to Locate

Once a new sensor has been installed and is working with the ODiSI-B software, the next step in interpreting the data is to go through the touch to locate process. This process requires the fiber sensor to be bonded or otherwise installed on the test article.

3.2.1 Identify Points

1. Navigate to the **Touch to Locate** tab in the ODiSI-B software interface

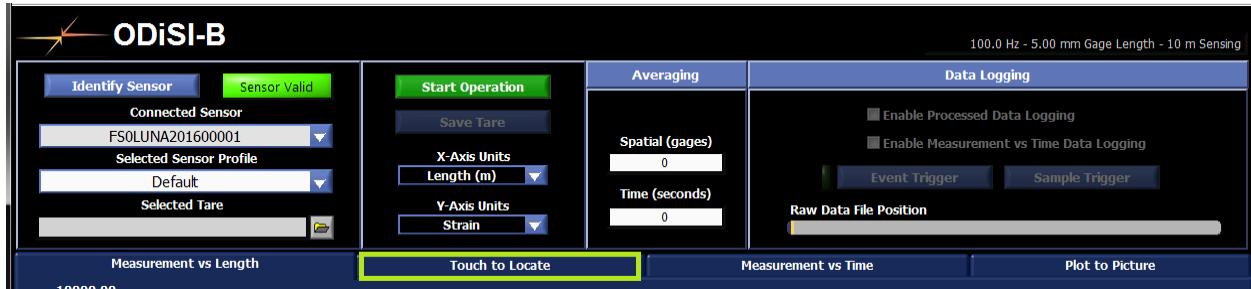


Figure 3-11 Touch To Locate tab location

1. Set a **Threshold** that is slightly above the noise floor of the data.
2. Select the appropriate **Touch Type** (Hot or Cold).

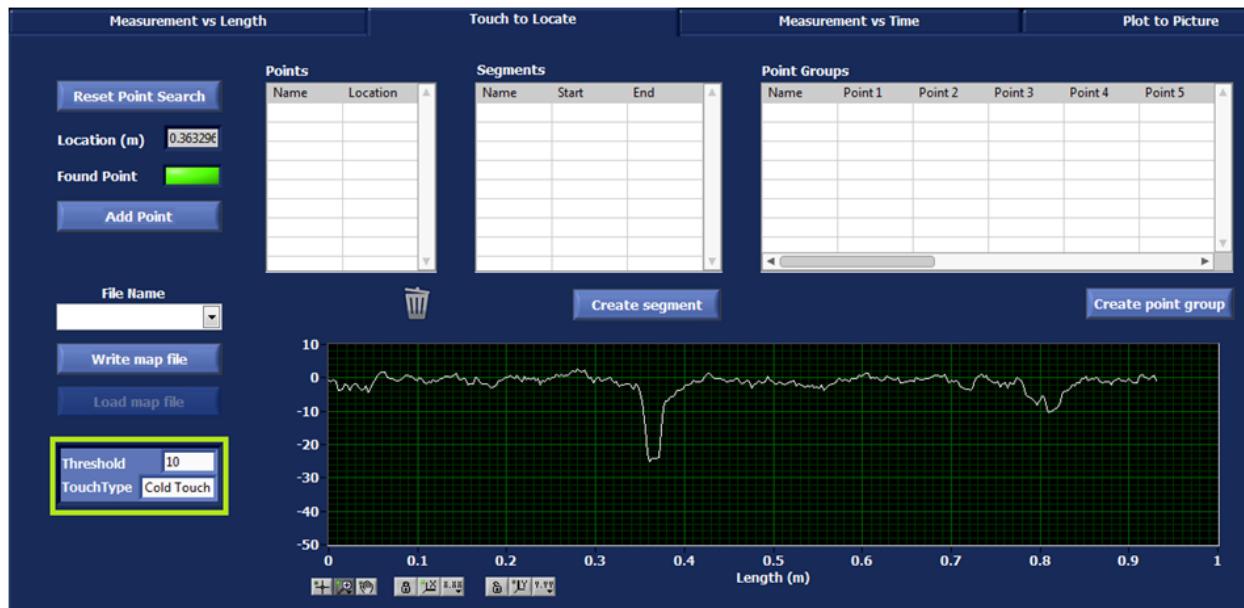


Figure 3-12 Threshold and Touch Type control locations

3. Click **Reset Point Search**
4. Apply a localized hot or cold touch to the fiber at the point of interest.
5. Click **Add Point** once the **Found Point** light turns green and the location matches that of the location where the hot or cold touch has been applied.
6. Rename the added point by triple clicking on the point name if desired.

7. If desired, delete a Point by dragging it into the trash bin below the Points array.

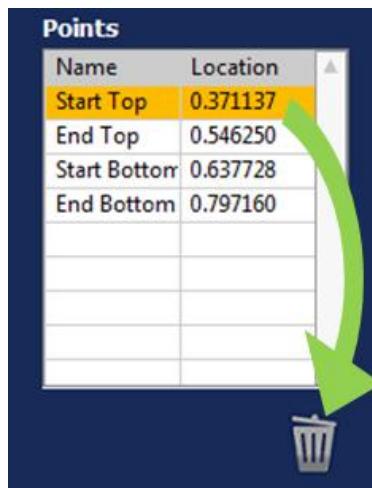


Figure 3-13 Drag Points (or Segments, or Point Groups) into the trash can to delete them.

8. Repeat steps 3 – 6 for all points of interest.

3.2.2 Create Segments

Once points of interest have been determined the user can group them into segments.

1. Click the **Create Segment** button

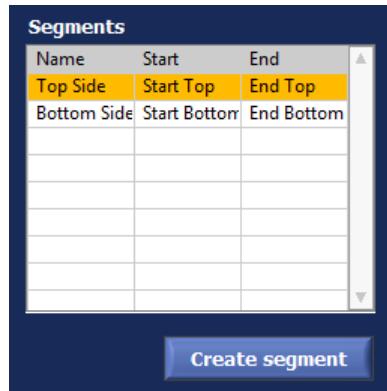


Figure 3-14 Segment array and **Create Segment** button

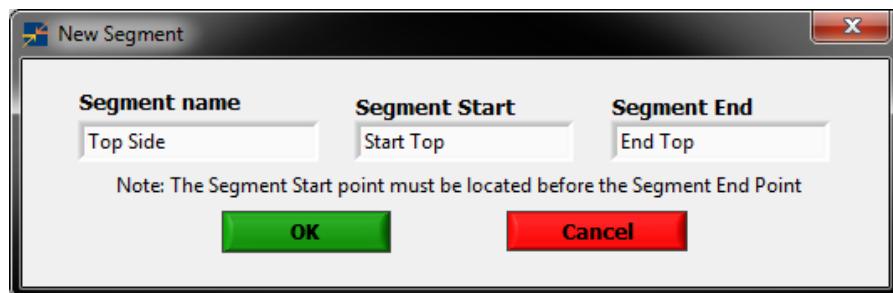


Figure 3-15 New Segment popup window

2. Name the segment
 3. Choose the start and end points from the list of points created earlier
 4. Click **OK** to add the segment to the segments list
 5. Repeat steps 1 – 4 for each desired segment

Note: The start point in the segment must occur before the end point (starting from the sensor connector).

3.2.3 Create Point Groups

Point groups can also be generated if the measurement data between the created points is not of interest.

Figure 3-16 Point Group array and **Create Point Group** button

1. Click the **Create Point Group** button
 2. Rename the segment if desired by triple clicking on the name cell
 3. Drag points from the **Points** field to the cells in the **Point Groups** array to add them to the point group
 4. Repeat steps 1 – 4 for each desired point group

3.2.4 Save the Data to a Map File

Once all points, segments and point groups have been created, they can be saved to a map file for use by other sections of the ODiSI-B software.

The screenshot shows the PointSearch software interface. On the left, there's a search bar with 'Reset Point Search' and 'Location (m) 0'. Below it are 'Found Point' and 'Add Point' buttons. A green box highlights the 'File Name' dropdown set to 'CantileverBeam' and the 'Write map file' button. In the center, there are three tables: 'Points' (with rows for Start Top, End Top, Start Bottom, and End Bottom), 'Segments' (with rows for Top Side, Start Top, End Top, and Bottom Side, Start Bottom, End Bottom), and 'Point Groups' (with rows for Cantilever E, Start Top, End Bottom, and several empty rows). At the bottom, there's a plot area with a grid, a trash can icon, and 'Create segment' and 'Create point group' buttons.

Figure 3-17 Map file saving controls

1. Enter a map file name into the **File Name** field (the user should not append a .map to the name as this will be done automatically)
2. Click the **Write Map File** button to save the file

3.3 Measurement vs. Time

The Measurement vs Time tab contains the controls necessary to view measurements from the sensor as a function of time rather than length. Points, Segments, and/or Point Groups from the map file can be viewed here, as well as custom points and segments.

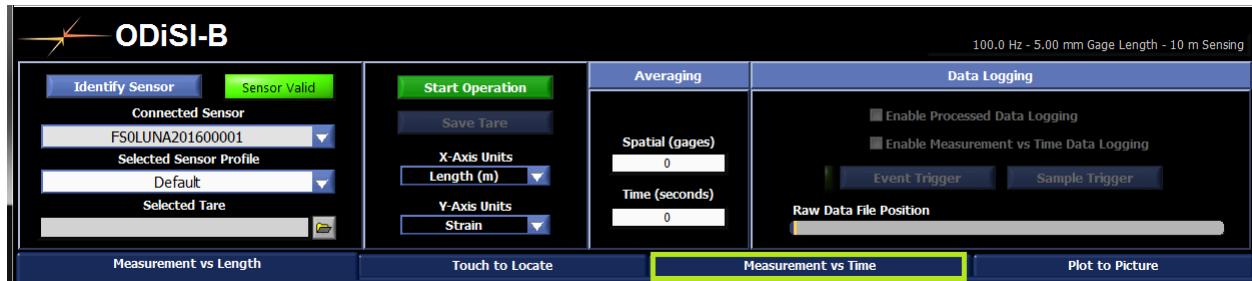


Figure 3-18 Measurement vs Time tab location

3.3.1 Adding Custom Points and Segments to the Plot

Within the **Custom Point** tab in the **Measurement vs. Time** interface



Figure 3-19 Custom Point tab within the Measurement vs Time interface

1. Enter a name into the **Name** Field
2. Enter the location of the point to be displayed into the **Start Location** field (the length units match the X-Axis units chosen in the **Instrument Control** section)
3. If adding a segment, check the **Display Segment** box and enter the location of the end point of the segment in the **End** field
4. Click the **Add to Plot** button

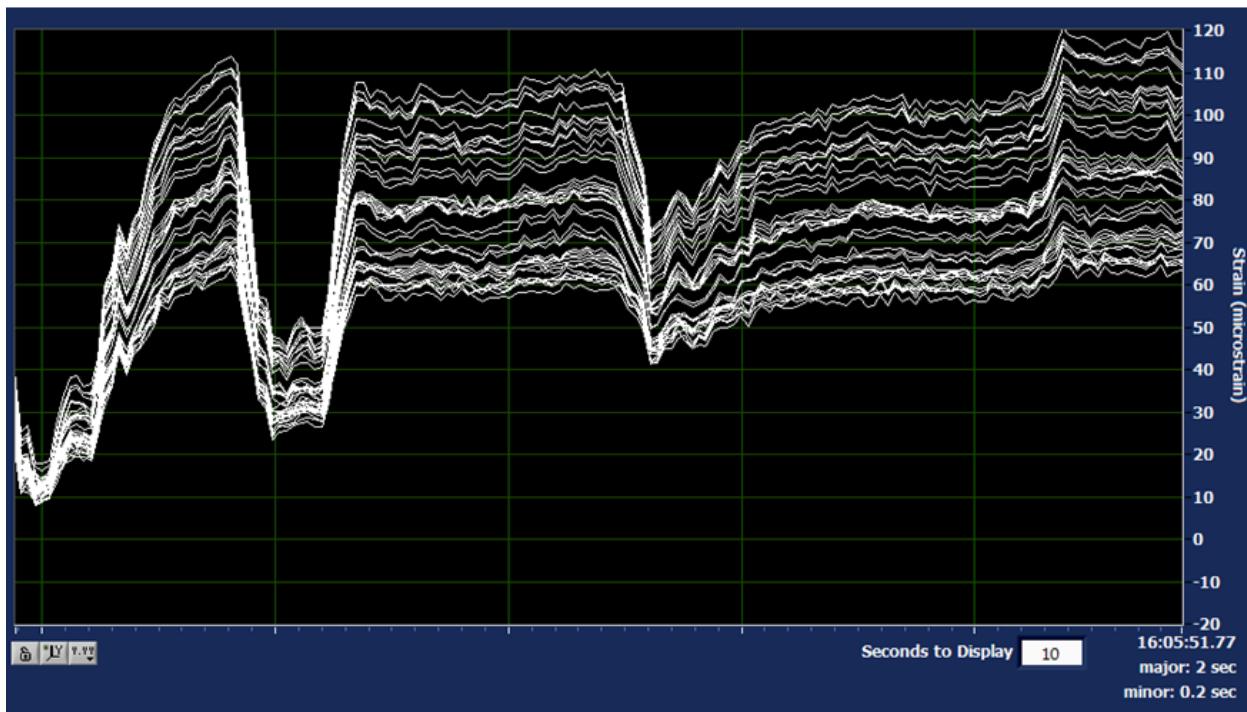


Figure 3-20 Example of data shown when a segment is plotted

3.3.2 Adding Points, Segments, and Point Groups from Map File

Traces can also be added to the plot based on the points, segments, and point groups saved within a map file.

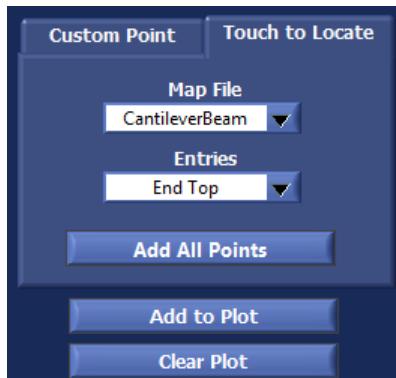


Figure 3-21 Touch to Locate tab within the Measurement vs Time interface

Within the **Touch to Locate** tab in the **Measurement vs. Time** interface

1. Select the map file from the **Map File** dropdown menu
2. Select the point, segment, or point group from the **Entries** dropdown menu
3. Click the **Add to Plot** button
4. Repeat steps 2 – 4 as desired

It is possible to add all points (not segments or point groups) to the plot at the same time by clicking the **Add all Points** button.

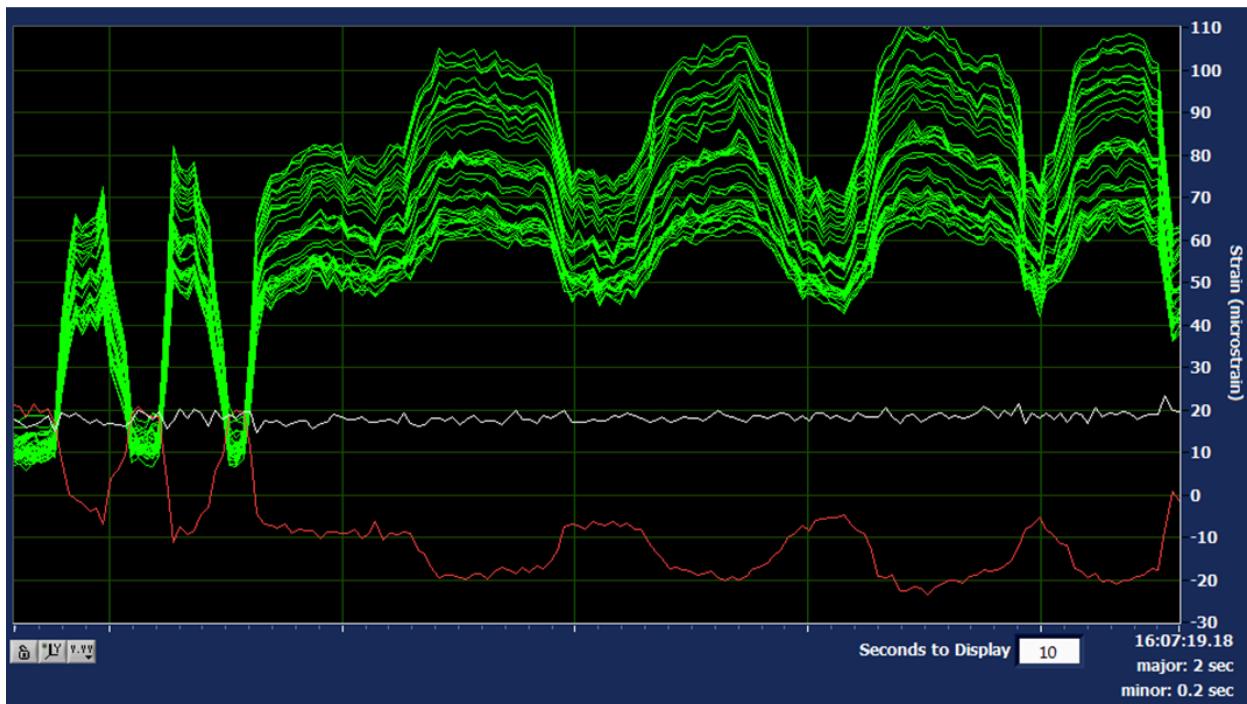


Figure 3-22 Example of data shown when plotting two points and a segment

3.4 Plot to Picture

The **Plot to Picture** feature allows the user to view measurement data as a color map along segments overlaid onto a picture of the test article.

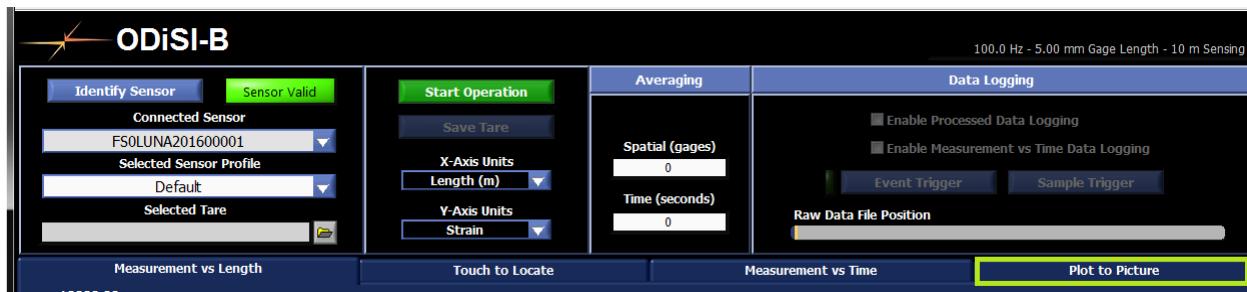


Figure 3-23 Plot to Picture tab location

1. Click Stop Operation if the software is running
2. Navigate to a picture of the test article using the **Picture Location** file path control

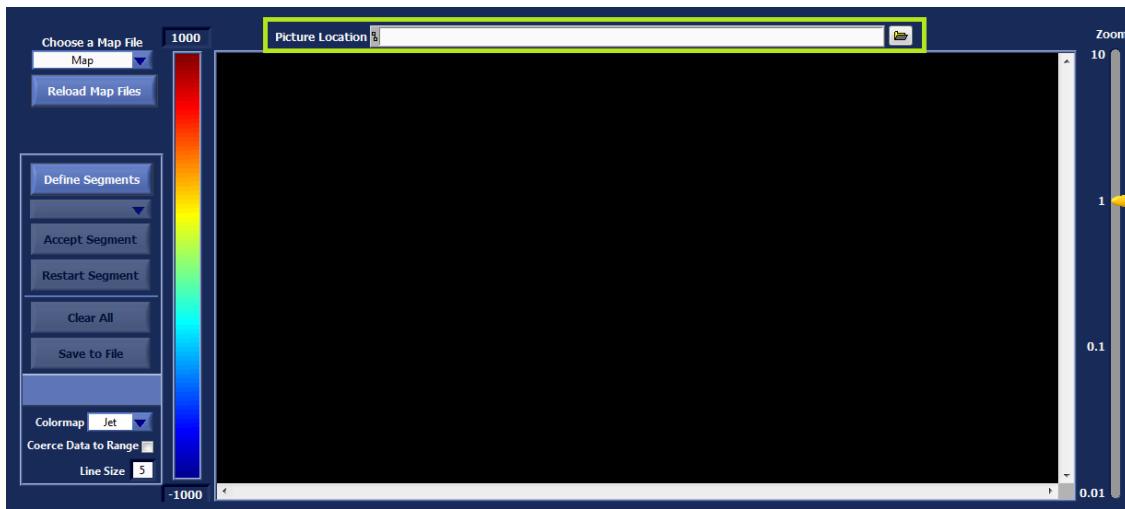


Figure 3-24 Picture Location control located at the top of the Plot to Picture interface

3. Select the recently-created map file

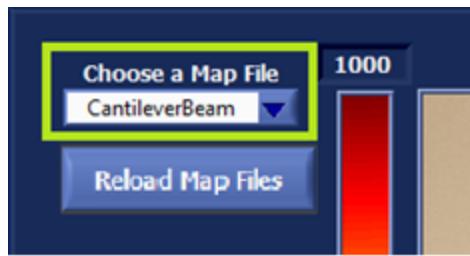


Figure 3-25 Map File selection control location

4. Click on the **Define Segments** button



Figure 3-26 Segment creation mode enabled after clicking Define Segments

5. Select the segment that is to be plotted onto the picture from the dropdown menu
6. Define the path of the sensor across the test article by clicking along its location on the picture. A line will be created that connects all of the chosen points.



Figure 3-27 Creating a line on the picture that matches the path of the fiber sensor

7. When satisfied with the line segment, click the **Accept Segment** button. If not, click the **Restart Segment** button to start over.
8. Repeat steps 5 – 7 until lines have been created for all of the desired segments
9. Click the **Save to File** button to save the created lines to the map file
10. Click the **Defining** button to exit segment definition mode
11. Adjust the color scale by changing the limits above and below the color bar



Figure 3-28 Color scale range control locations

12. Click **Start Operation** to start displaying data on the picture

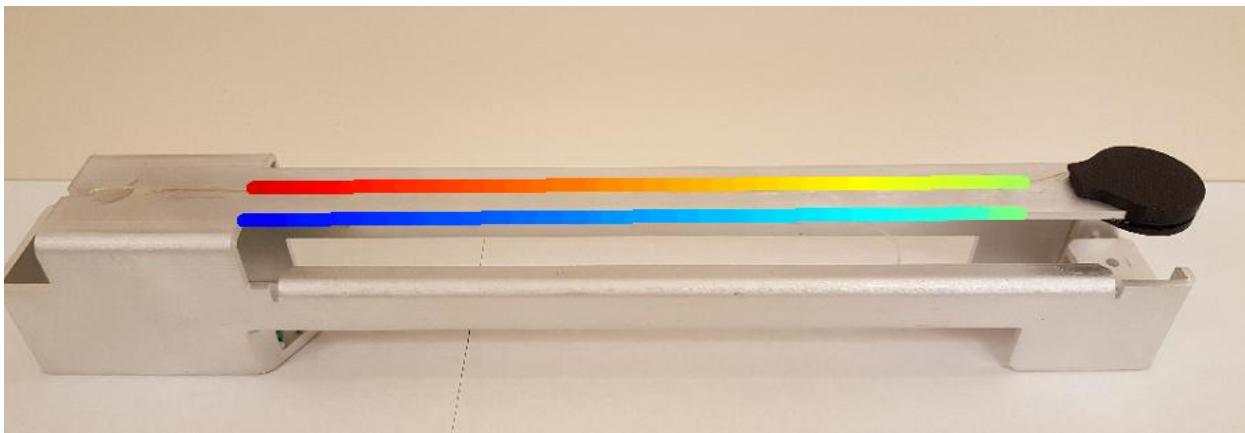


Figure 3-29 Strain being displayed as a heat map onto an image of a cantilever beam. The top side of the beam is under tension and shows a linear transition from red to green. The bottom side is under compression and shows a linear transition from blue to green.

3.5 Data Logging

ODiSI-B users can save measurement data in the form of raw (optical) or text (processed strain or temperature) files.

3.5.1 Processed Data Files

Processed data files can be created to store strain or temperature measurement data for gages of interest (Measurement vs. Time) or along the entire length of the fiber sensor (Measurement vs. Length).

Measurement vs. Length

1. Click on **Stop Operation** if the software is currently processing data.
2. Navigate to **Data Logging > Processed Data Logging Settings...**
3. Navigate to the **Processed Data File** tab.

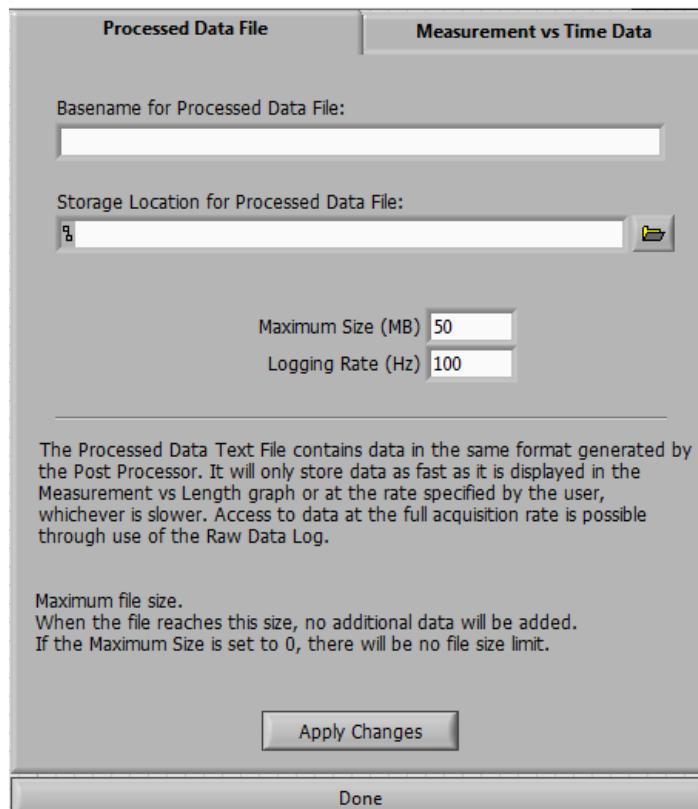


Figure 3-30 Processed Data File settings

4. Enter a base file name. This will be combined with a timestamp to create a unique file name.
5. Navigate to a folder to store the logged data using the **Storage Location for Processed Data File** browser.
6. Enter a maximum file size or leave at 0 to write to the file indefinitely.
7. Choose the rate at which to write measurement data to the log file.
8. Click **Apply Changes** to save the logging settings.
9. Click the **Done** button to leave the Data Logging Settings Window.
10. While the system is acquiring data (Start Operation) the logging controls will unlock.

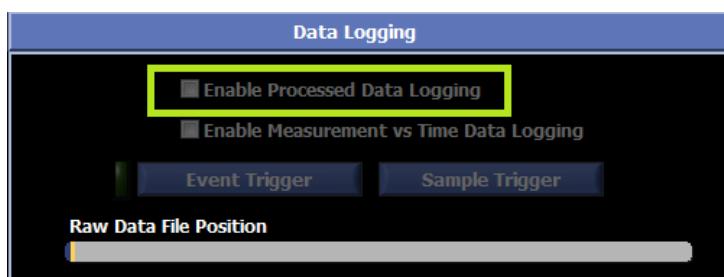


Figure 3-31 Logging Controls for the Processed Data File

11. Clicking on the **Enable Processed Data Logging** will allow the software to start writing all data displayed in the Measurement vs. Length plot to the folder selected. The scan counter to the right of the checkbox will increment each time data is written to the file.

Measurement vs. Time

1. Click on **Stop Operation** if the software is currently processing data.
2. Navigate to **Data Logging > Processed Data Logging Settings...**
3. Navigate to the **Measurement vs Time Data** tab.

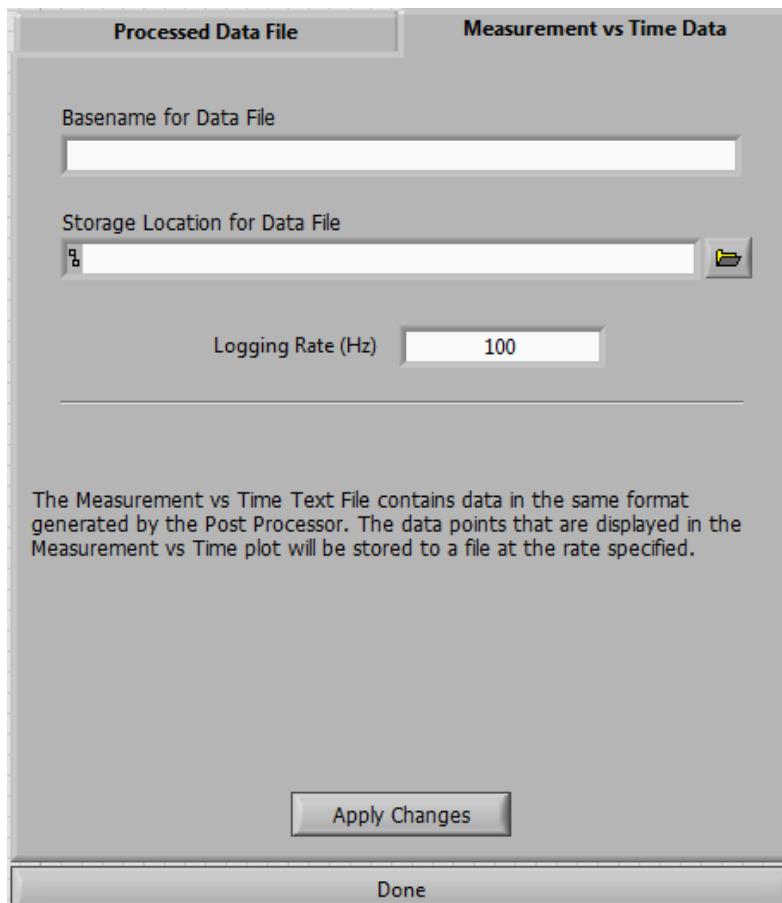


Figure 3-32 Measurement vs Time Data Tab

4. Enter a base file name. This will be combined with a timestamp to create a unique file name.
5. Navigate to a folder to store the logged data using the **Storage Location for Data File** browser.
6. Enter a logging rate. This will control how frequently the software writes the processed data to the Measurement vs Time text file. Data cannot be written to this file at rates higher than the processing rate of the software. The FPS indicator in the lower right corner of the software displays the processing rate.
7. Click **Apply Changes** to save the logging settings.

8. Click the **Done** button to leave the Data Logging Settings Window.
9. Navigate to the **Measurement vs Time** tab of the control software.
10. While the system is processing data (**Start Operation**) the **Enable Data Logging** checkbox at the bottom of the screen will be enabled.

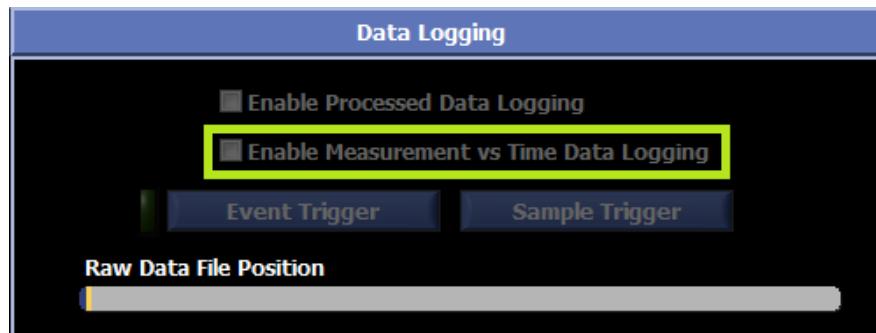


Figure 3-33 Location of the Enable Data Logging checkbox

11. Checking the **Enable Measurement vs Time Data Logging** checkbox will trigger the software to write all data that is presented in the Measurement vs Time chart to a tab delimited text file.

3.5.2 Raw Data Files

Raw data files contain optical data that can be read by the ODiSI-B post processor and processed into strain or temperature in the same way as the control software. These files can be written to at rates of up to the full acquisition rate of the ODiSI-B.

Circular Raw Data Logging

Raw data can be logged to a circular file and controlled using the Event Trigger. Once created, this file is written to whenever the ODiSI-B software is processing data. When the file has been filled, new data will overwrite the oldest data in the file.

1. Click on **Stop Operation** if the software is currently processing data.
2. Navigate to **Data Logging > Raw Data Logging Settings...**
3. A raw data configuration wizard will open, allowing the user to configure how data will be stored.
4. File Location and Name
 - a. Specify a Save Folder that is located on the solid state drive of the PC (E:\)
 - b. Specify a File Base Name for the raw data file. This will have a timestamp appended to it.

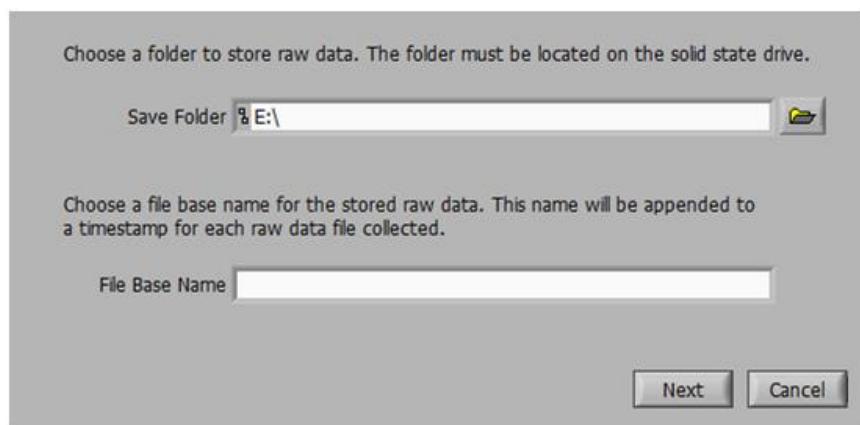


Figure 3-34 Raw data file location controls

5. File Writing Method

- Select the Circular option.

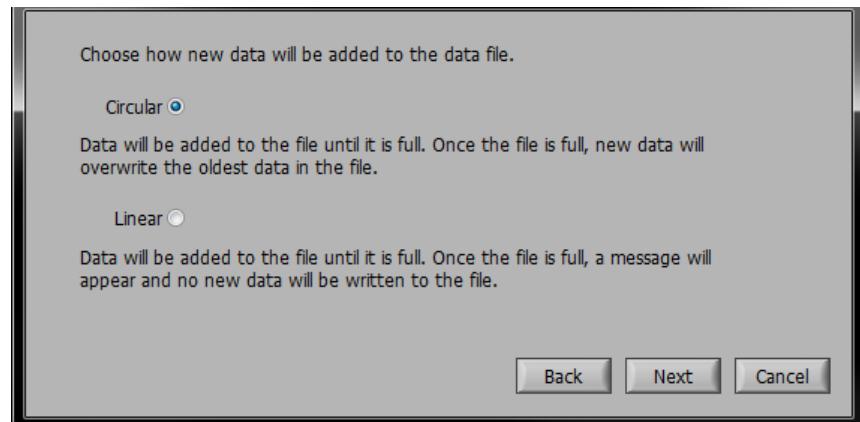


Figure 3-35 File logging method controls

6. File Options

- Specify the Number of Acquisitions to store to the log file. This will be translated to a file size and duration.
Note: The file size is limited to half of the available drive space.
- Choose a Down Sampling Factor. This allows the user to save every nth acquisition. Changing this will automatically adjust the File Duration, Sample Rate, and Sample Period shown.
- Select which method to use to stop logging to the circular raw data file.
 - Ignore Trigger – The Event Trigger does not stop the file logging
 - Start on Trigger – When an Event Trigger is received, the file is stopped once all data is from after the trigger.
 - Stop on Trigger – When an Event Trigger is received, the file is immediately closed and no new data is logged.

- iv. Center on Trigger – When an Event Trigger is received, the software continues to write to the file until half of the data is from scans after the trigger.

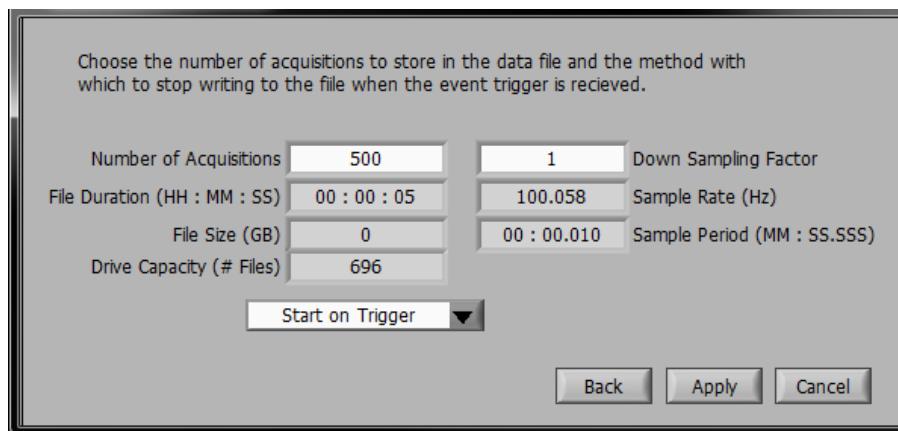


Figure 3-36 Circular data file options

7. Click **Apply** to save the raw data logging settings and exit the wizard.
8. While the ODiSI-B is taking measurements (**Start Operation**) the software will write acquisitions to the circular log file at the rate specified by the down sampling factor. This can be observed by watching the **Raw Data File Position** progress bar in the Data Logging section of the interface.

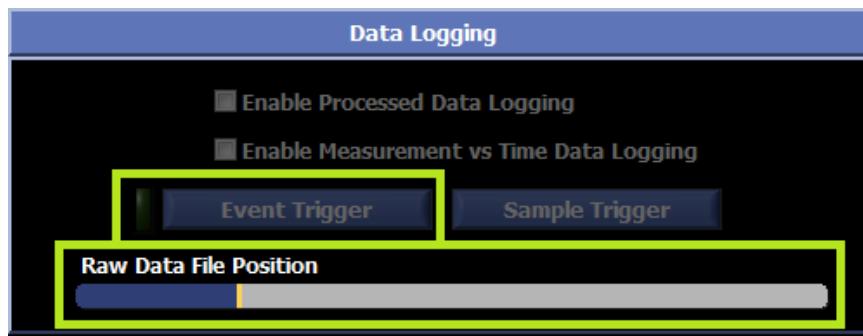


Figure 3-37 Event Trigger and Raw Data File Position

9. Clicking on the **Event Trigger** button or sending a 5v electrical pulse into the **Event** BNC port on the rear of the ODiSI-B will pause writing to the circular log file in the manner chosen in step 6. Electrical specifications for the external **Event** trigger can be found in Appendix A.
10. Once the software has paused writing to the log file, it will append the timestamp corresponding to the Event Trigger and create a new raw data log file with the same properties as the one closed.

Linear Raw Data Logging

Raw data can be logged in a linear fashion, where the oldest data in the file will never be overwritten. Linear raw data logging uses the Event Trigger to begin the logging process and the Sample Trigger to stop the logging process. This is useful if the user wishes to gather data for a period of time with an unknown length.

1. Click on **Stop Operation** if the software is currently processing data.
2. Navigate to **Data Logging > Raw Data Logging Settings...**
3. A raw data configuration wizard will open, allowing the user to configure how data will be stored.
4. File Location and Name
 - a. Specify a Save Folder that is located on the solid state drive of the PC (E:\)
 - b. Specify a File Base Name for the raw data file. This will have a timestamp appended to it.

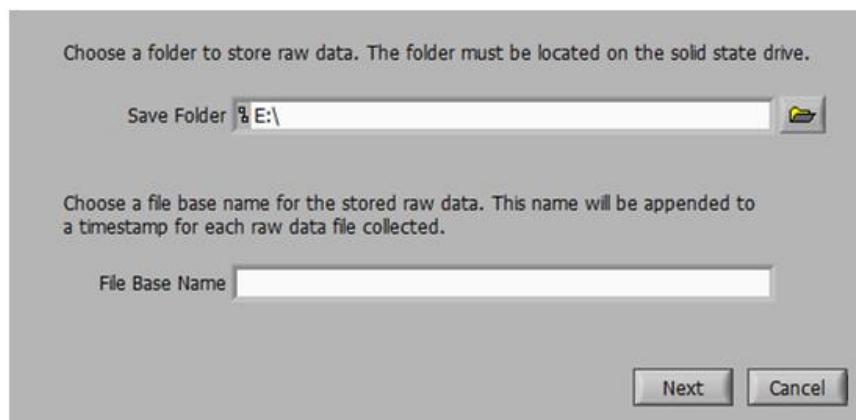


Figure 3-38 Raw data file location controls

5. File Writing Method

- a. Select the Linear option.

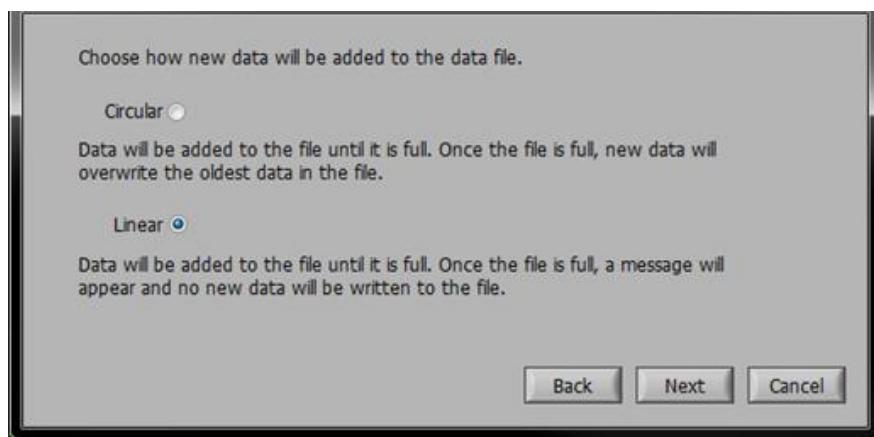


Figure 3-39 File logging method controls

6. File Options

- a. Specify the maximum number of acquisitions to store in the file. The time duration of the file as well as the file size will be displayed.

Note: The file is limited to half of the available disk space.

- b. Choose a Down Sampling Factor. This allows the user to save every nth acquisition. Changing this will automatically adjust the File Duration, Sample Rate, and Sample Period shown.

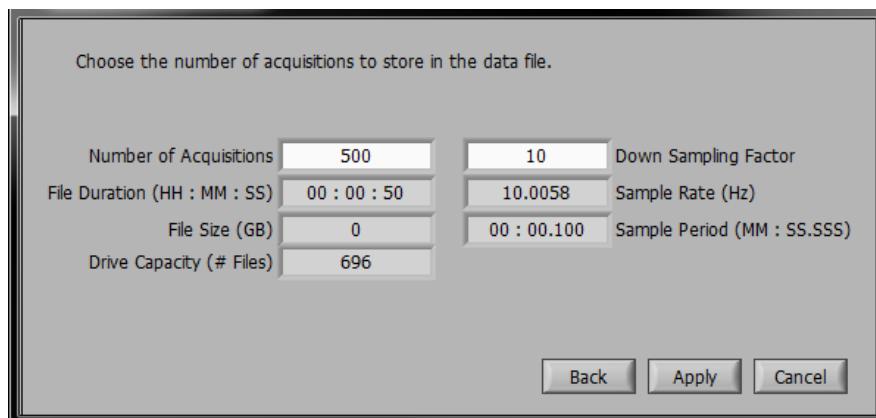


Figure 3-40 Linear data file options

7. Click **Apply** to save the raw data logging settings and exit the wizard.
8. With the ODiSI-B scanning (**Start Operation**) the user can control logging to the raw data log file from the Data Logging section of the interface.

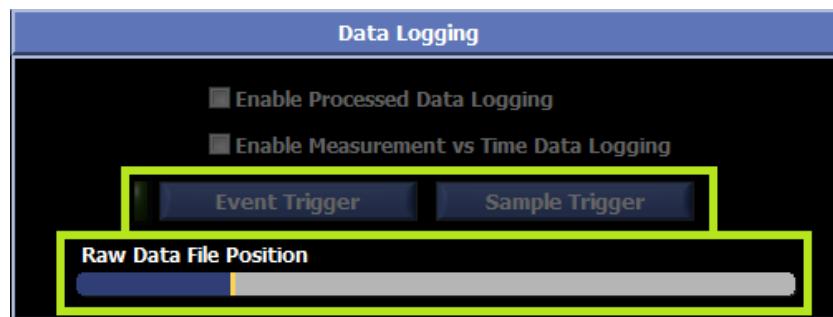


Figure 3-41 Linear Raw Data Logging Controls

9. Clicking on the Event Trigger button or sending the ODiSI-B a 5V electrical trigger to the Event BNC port will result in the software starting to log data to the linear log file.
10. Clicking on the Sample Trigger button or sending the ODiSI-B a 5V electrical trigger to the Sample BNC port will result in the software stopping data logging.
11. If the maximum number of acquisitions is reached, the Raw Data File Position bar will turn red and the software will stop logging raw data to the file.

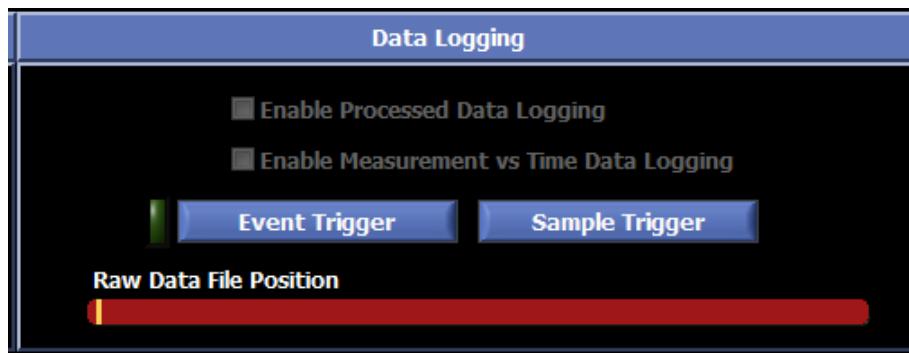


Figure 3-42 Red colored Raw Data File Position indicator showing that the file is full and logging has stopped.

12. A new raw data log file will be created each time an Event Trigger is received or until the solid state drive of the PC is full.

3.6 ODiSI-B Post Processor Software

The ODiSI-B Post Processor allows users to load and process saved raw data files into strain or temperature measurements. These measurements can be viewed and saved to text files in the same way as the ODiSI-B control software.

3.6.1 Starting the Software

1. Launch the software by double clicking on the "ODiSI-B Post Processor v5.2.0" desktop shortcut.
2. When the file browser window opens, navigate to and select the raw data (.odb) file to be processed.
3. Once the file has been loaded, the post processing software will activate and process the first measurement in the log file.

3.6.2 File Playback

Playback controls are located in the upper part of the screen and are used to play through or view individual measurements contained within the log file being processed.

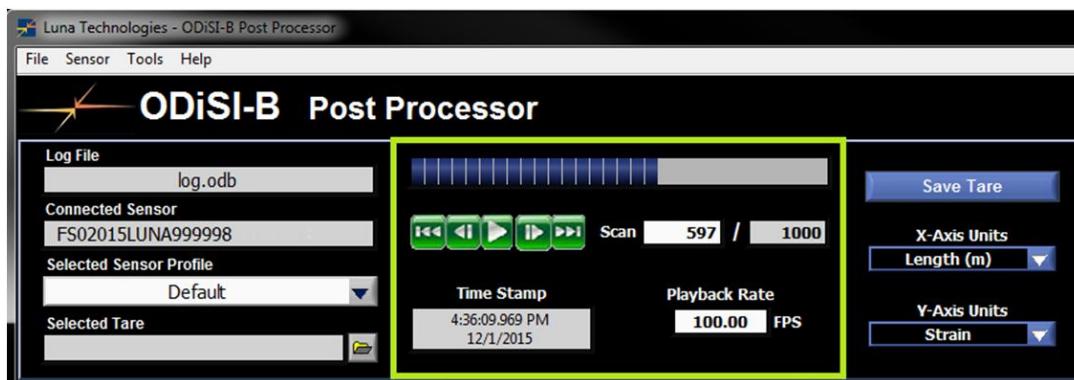


Figure 3-43 Playback Controls

1. Select appropriate X and Y Axis units for the measurement file.

2. Click the play button  to process the data file one scan at a time at the rate shown in the **Playback Rate** control.
3. Click on the  or  buttons to step a single scan forward or backward respectively.
4. Alternatively, the user can move through the log file by clicking and dragging on the progress bar at the top of the playback controls section or by manually entering a scan number into the **Scan** control.

3.6.3 Saving Processed Data

Like with the ODiSI-B control software, processed data can be saved from the post processor in two forms.

Measurement vs. Length

1. Click on **Stop Operation** if the software is currently processing data.
2. Click on the white gear button in the upper right corner of the Data Logging section.

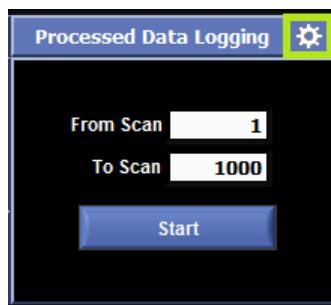


Figure 3-44 Location of the Data Logging Settings Gear

3. Navigate to the **Processed Data File** tab.

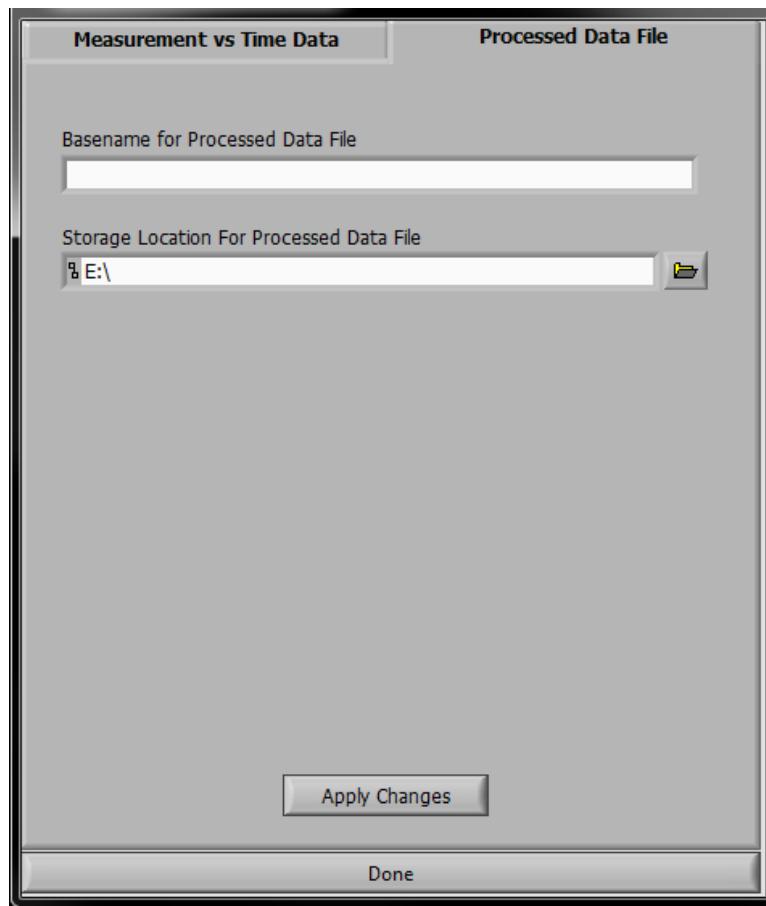


Figure 3-45 Processed Data File settings

4. Enter a file name into the **Basename for Processed Data File** field. A UTC time stamp will be added to this name to prevent accidental overwriting of the data file.
5. Navigate to a folder using the **Storage Location for Processed Data File** browser.
6. Click **Apply Changes** to save the logging settings.
7. Click the **Done** button to leave the Data Logging Settings Window.
8. Enter a start and end scan for logging into the **From Scan** and **To Scan** fields respectively.

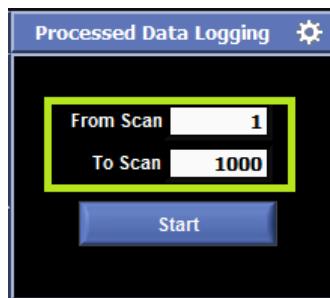


Figure 3-46 Logging Controls for the Processed Data File

9. Once the user clicks **Start**, the software will process each scan, display it to the screen, and log it to the file selected. The rate at which this happens can be controlled by the **Playback Rate** control described in the playback controls section.

Measurement vs. Time

1. Click on the white gear button in the upper right corner of the Data Logging section.

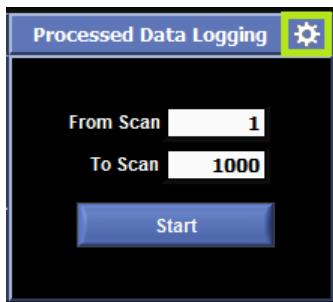


Figure 3-47 Location of the Data Logging Settings Gear

2. Navigate to the **Measurement vs Time Data** tab.

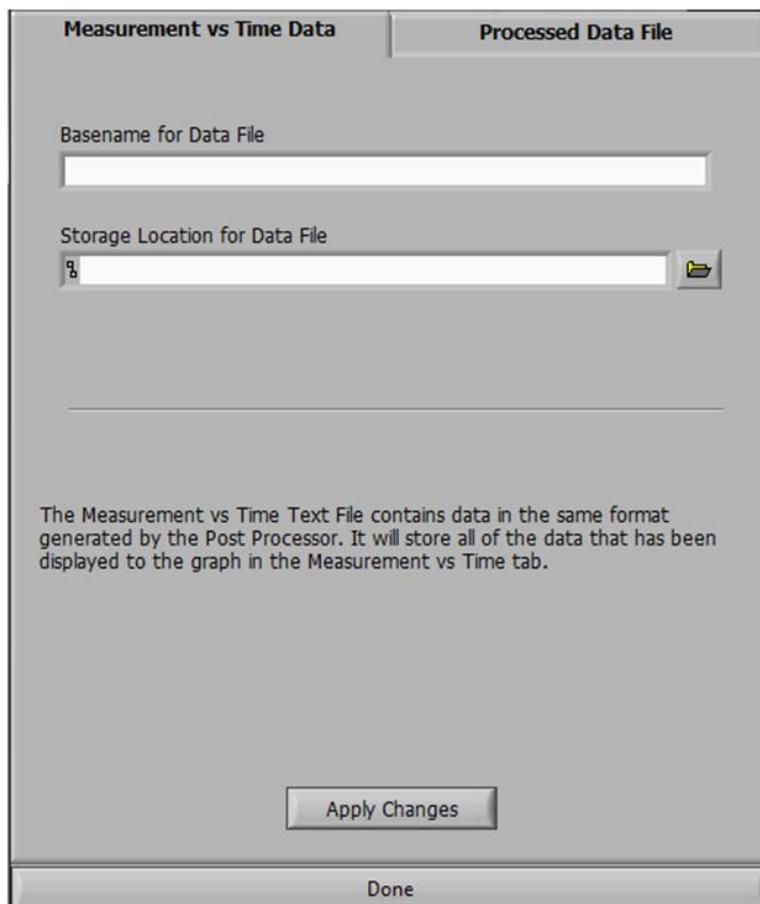


Figure 3-48 Measurement vs Time Data Tab

3. Enter a base file name. This will be combined with a timestamp to create a unique file name.

4. Navigate to a folder to store the logged data using the **Storage Location for Data File** browser.
5. Click **Apply Changes** to save the logging settings.
6. Click the **Done** button to leave the Data Logging Settings Window.
7. Navigate to the **Measurement vs Time** tab of the control software.
8. Once the user clicks **Start**, the software will process each scan, display it to the screen, and log the Measurement vs Time data.

4 Software Guide

The control software is the primary interface between the user and the instrument. It provides all of the necessary tools to perform temperature and strain measurements. This chapter provides an overview of the location and function of all controls and indicators included in the control software.

4.1 Starting the Software

Double click on the ODiSI-B icon on the desktop or select Luna Technologies > ODiSI-B from the Start Menu. The ODiSI-B control software will open and a splash screen will show while the system configures.



Figure 4-1 ODiSI-B software splash screen

While loading the software will open a window that allows the user to select the desired mode of operation.

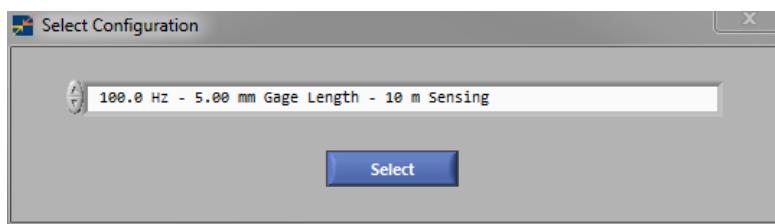


Figure 4-2 Select Configuration window allows the user to choose from all available scan modes

The ODiSI-B system is capable of operating in a number of modes. These are listed in the table below. Each mode balances tradeoffs in acquisition rate, sensing length, and gage pitch. The currently selected mode of operation is displayed in the upper right corner of the main **ODiSI-B Control** window. This display includes the mode-specific acquisition rate, gage length and sensing length. A more thorough set of specifications can be found in Appendix A.

Mode of Operation	Maximum Sensing Length (m)	Acquisition Rate (Hz)	Gage Pitch (mm)	Gage Length (mm)	Supports High-Speed CFG
Standard	10	100	2.61	5.22	Yes
High Speed	2	250	2.61	5.22	No
High Resolution	10	23.8	0.65	1.30	Yes
Extended Length	20	50	2.61	5.22	No

Table 1 ODiSI-B modes of operation

The main ODiSI-B control software is comprised of seven functional areas:

1. The **Menu Bar** contains tools for selecting log file locations and names, installing and uninstalling sensors, choosing a sensor repository, aligning the system, and configuring the optional DAC.
2. The **Sensor** area contains buttons and indicators to identify sensors, select profiles, and manage tares.
3. The **Instrument Control** section provides control over the starting and stopping of acquisitions and the units selection for measurement data display.
4. The **Averaging** area contains the tools necessary to apply spatial and temporal averaging to the processed data.
5. The **Data Logging** area contains the buttons and indicators necessary to control the storage of raw (optical) and processed (strain or temperature) data to the PC.
6. The **Measurement Display** area offers four tabs that can be used to view and interpret the processed data in a number of ways.
7. The **Error Reporting** area indicates the presence and details of various system errors.

4.2 Menu Bar

The **Menu Bar** contains tools used to control and view key elements of the software. It is disabled while the software is processing data.

4.2.1 File Menu

The **File** menu contains the exit button which is the suggested method for closing the ODiSI-B Control software.

4.2.2 Sensor Menu

1. View Installed Sensors

This opens a window that can be used to view all sensors installed on the current PC. Double clicking any sensor will open the **Update Sensor Information** window.

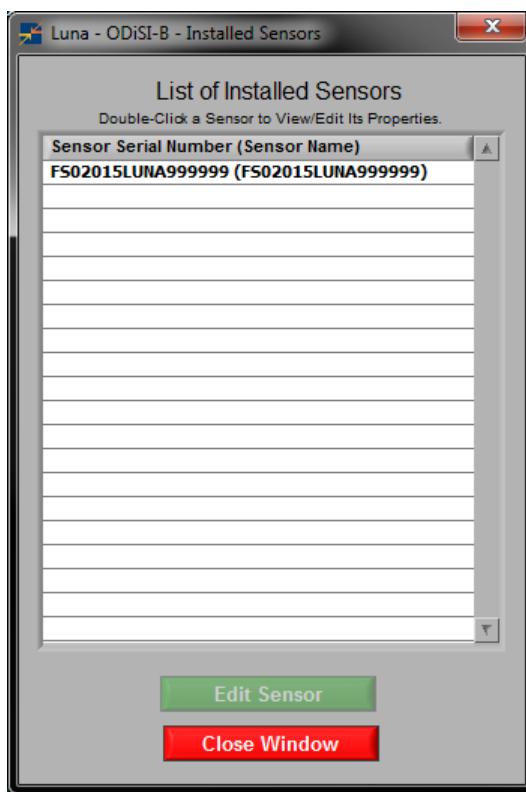


Figure 4-3 Installed Sensors window

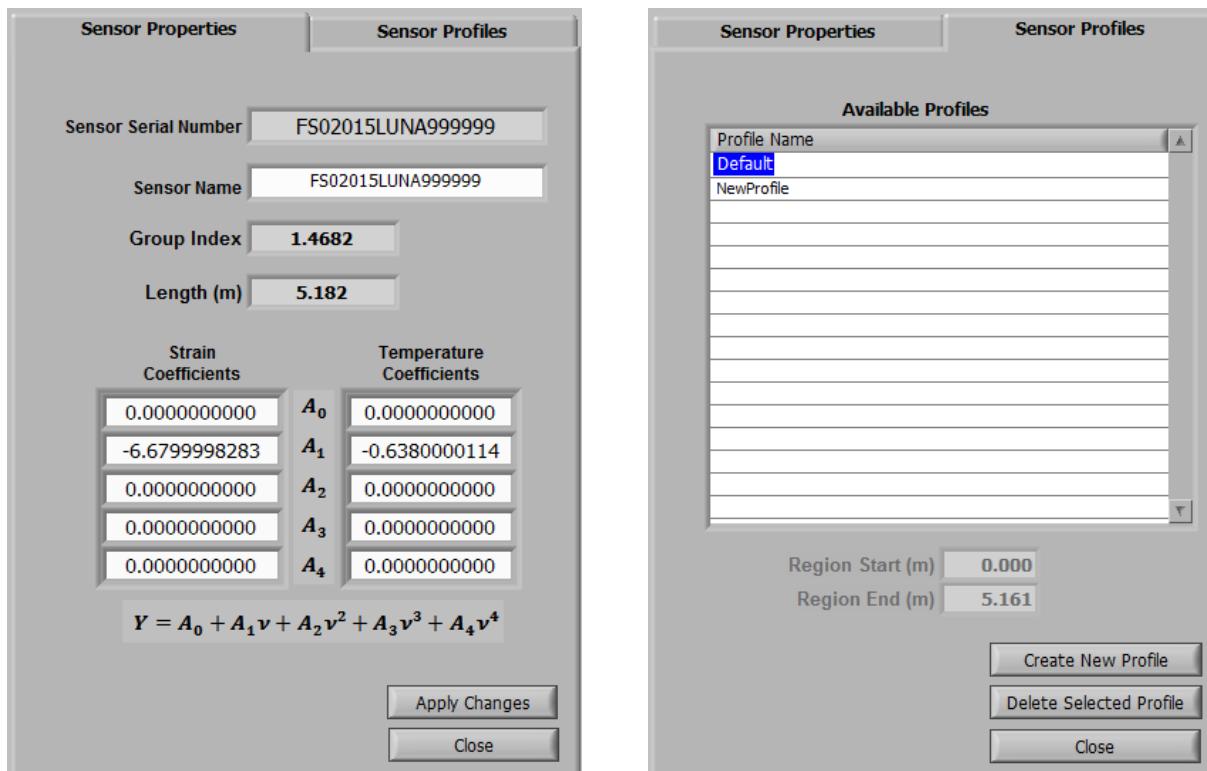


Figure 4-4 Update Sensor Information window allows the user to change sensor specific data including strain and temperature coefficients (gage factors) as well as create new sensor profiles.

The **Update Sensor Information** window allows the user to change various sensor specific values including:

- Sensor Name
- Strain Coefficients
- Temperature Coefficients (High-Definition sensors only)
- Create/Delete Profiles (High-Definition sensors only)

Profiles provide a way of only processing data over a subset of a High-Definition fiber sensor. Shortening the processing region will result in a moderate increase in the screen update rate of the software. In order to provide valid vibration correction, it is important that the first 20cm of this region is left unstrained even when it is not included in the profile.

Tip: If a secondary temperature measurement sensor (i.e. thermocouple) is being used, its reading can be entered in the A_0 temperature coefficient field. Enter this external reading when a tare is taken and the subsequent temperature measurements will be absolute ($^{\circ}\text{C}$) rather than relative.

2. Install Sensor...

The **Install Sensor** window allows the user to add sensor key files to the sensor repository for use by the ODiSI-B.

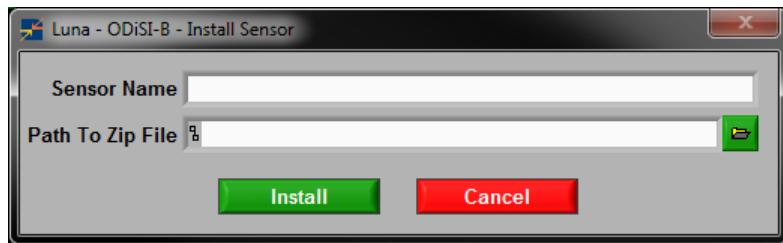


Figure 4-5 Install Sensor window

When installing a sensor, the user can choose to associate their own **Sensor Name** with the particular sensor. This **Sensor Name** will appear in all of the ODiSI-B software rather than the serial number associated with the sensor key.

3. Uninstall Sensor...

The **Uninstall Sensor** window allows the user to remove a sensor file from the sensor repository when it is no longer in use. Uninstalling a sensor will permanently remove all data associated with the selected sensor (tares, profiles and, map files).

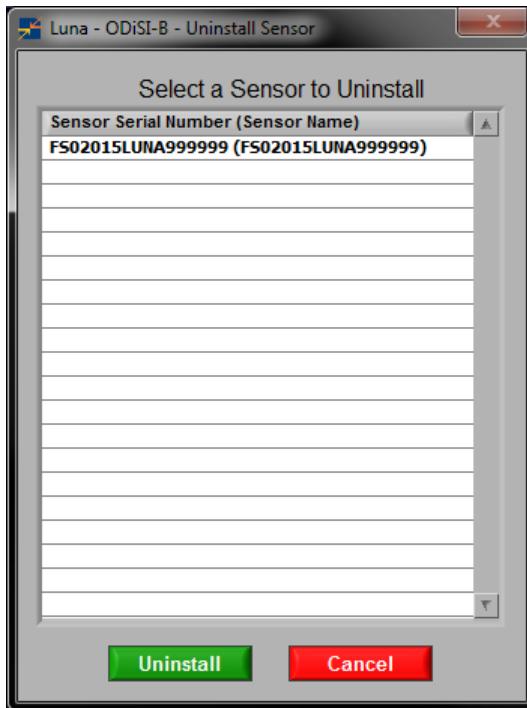


Figure 4-6 Uninstall Sensor window

4. Sensor Repository Settings

This allows the user to change the location of the sensor repository on their PC. The default location is *C:\ProgramData\Luna Technologies\ODiSI-B\Sensors*

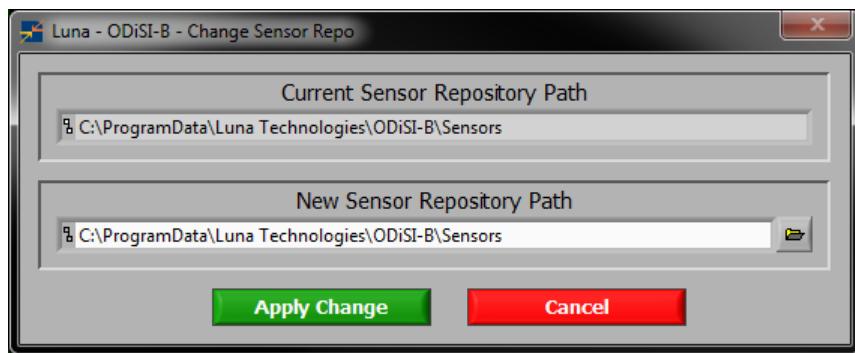


Figure 4-7 Sensor Repository Settings window

4.2.3 Tools Menu

1. Measurement Streaming

The **Measurement Streaming** option opens the Measurement Streaming Settings window. This feature allows users to stream real-time data using the ODiSI Measurement Streaming Protocol or MTS AeroPro DDI protocols. More information can be found in section 4.9 Measurement Streaming.

2. Check Alignment

Clicking on **Check Alignment** will initiate a check of the current alignment of the internal optical network. A dialog box will open prompting the user to remove any sensors or jumpers from the remote module prior to checking alignment.

If the system is not well aligned another dialog box will open asking the user whether or not to undergo an automated alignment.

Note: The alignment routine should be re-run any time the 50m standoff cable or remote module has been re-routed or moved. The remote module can be affixed to the test structure to avoid the need for routine re-alignment.

3. Check Alignment on Startup

Checking this option will prompt the software to initiate an alignment check each time the control software is launched. This is left unchecked by default.

4. DAC Settings

Note: Only available with the purchase of the Analog Output option

The **DAC Settings** option opens up the **DAC Settings** window which allows the user to configure the data sent to the 16 channel DAC.

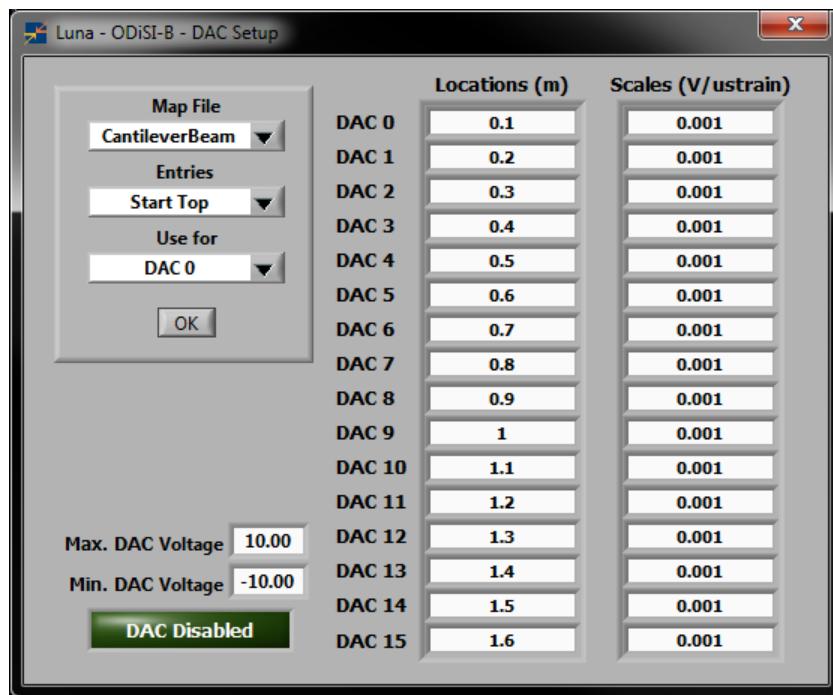


Figure 4-8 DAC Settings window

The DAC settings window allows the user to configure each of the 16 analog out channels individually. Each channel (row) can be assigned to a different measurement gage along the fiber and can be given a unique voltage / strain (or temperature) output scaling. It is possible to assign the locations based on points gathered during the Touch to Locate process and stored in the sensor's map file using the dropdown menus located to the right of the screen. A detailed description of the Touch to Locate process and creation of map files is located in the *Touch to Locate* description.

4.2.4 Data Logging Menu

1. Processed Data Logging Settings

The **Processed Data Logging Settings** option opens the **Data Logging Settings** window which contains the controls necessary to configure the storage of raw (optical) and processed (strain or temperature) data. More information can be found in section [4.6 Data Logging](#).

2. Raw Data Logging Settings

The **Raw Data Logging Settings** option opens a wizard that allows the user to configure the storage options for the raw data collected by the ODiSI-B. More information can be found in section [4.6 Data Logging](#).

4.2.5 Help Menu

Users Guide

Selecting this option opens the ODiSI-B User's Guide using Adobe Acrobat Reader.

Fiber Sensor Installation Guide

Selecting this option opens the ODiSI Fiber Optic Sensor Installation Guide using Adobe Acrobat Reader. This document provides instructions for the correct installation of Luna's fiber optic sensing fiber.

What's New

Selecting this option opens a popup window that displays information regarding the changes that have been implemented in the current version of the ODiSI-B software.

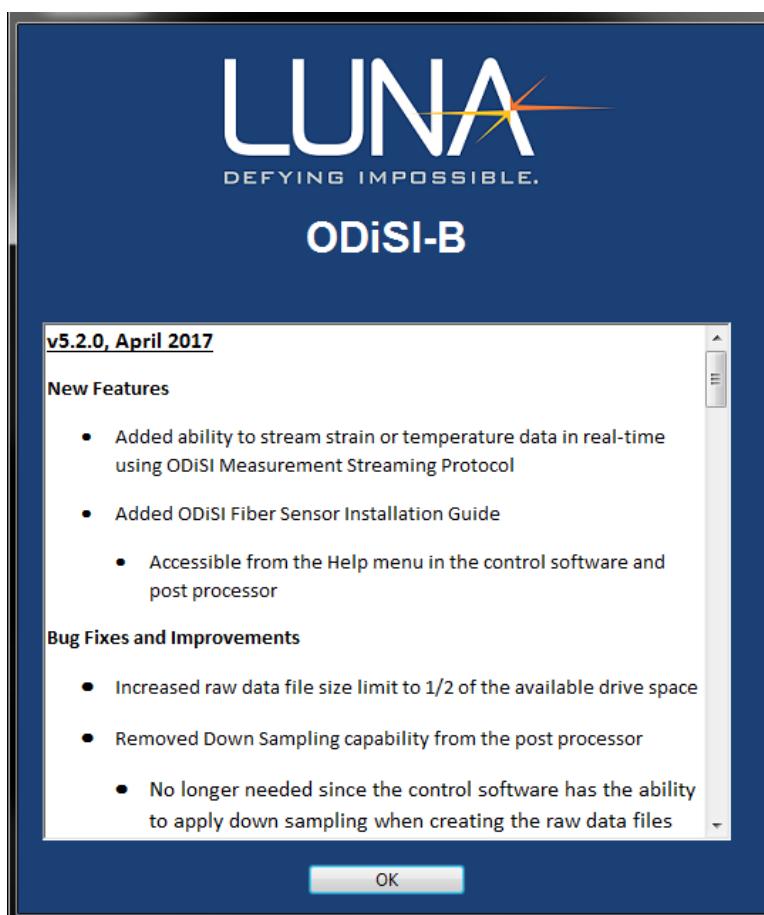


Figure 4-9 What's New Screen

About

Selecting this option opens the About window which contains the software and firmware version information in addition to contact information for Luna's Customer Support.



Figure 4-10 About window

4.3 Sensor

The **Sensor** section of the interface is located in the upper left corner of the screen and contains the controls and indicators related to the particular sensor connected to the system.

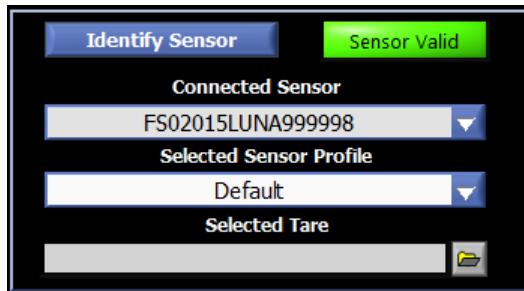


Figure 4-11 Sensor section of the interface

Clicking **Identify Sensor** automatically identifies which sensor is connected to the instrument. If a sensor in the sensor repository matches the connected sensor, the **Sensor Valid** light will turn green (red **Sensor Invalid** if the system fails to identify the sensor) and the sensor name will be shown in the **Connected Sensor** field.

If the name of the connected sensor is known, it is possible to select that sensor from the sensor repository using the **Connected Sensor** dropdown menu. When a new sensor is selected, the software will attempt to match the reference data to the measurement data of the connected sensor. If they match, the **Sensor Valid** light will turn green.

Inability to identify or select a sensor (red **Sensor Invalid** light) may be due to any of the following:

- Sensor is not yet installed and its key is not yet in the sensor repository
- The sensor has been damaged
- The connection between the sensor and the 50m standoff cable is dirty
- There is significant strain present in the first 20cm of the sensor (in the Teflon tube)
- The sensor or standoff is experiencing vibration or motion

The **Selected Sensor Profile** field displays the active sensor profile. Profile information can be viewed by double clicking on a sensor while within the **Installed Sensors** window (accessible through the file menu).

Note: Sensor profiles are only available for High-Definition sensors due to their length dependent processing speed.

The **Selected Tare** field shows the user which of the available tares are currently being applied to the data being displayed to the screen. Clicking on the browse button  opens the **Select a Tare** window which allows the user to select from and delete tares that have already been created.

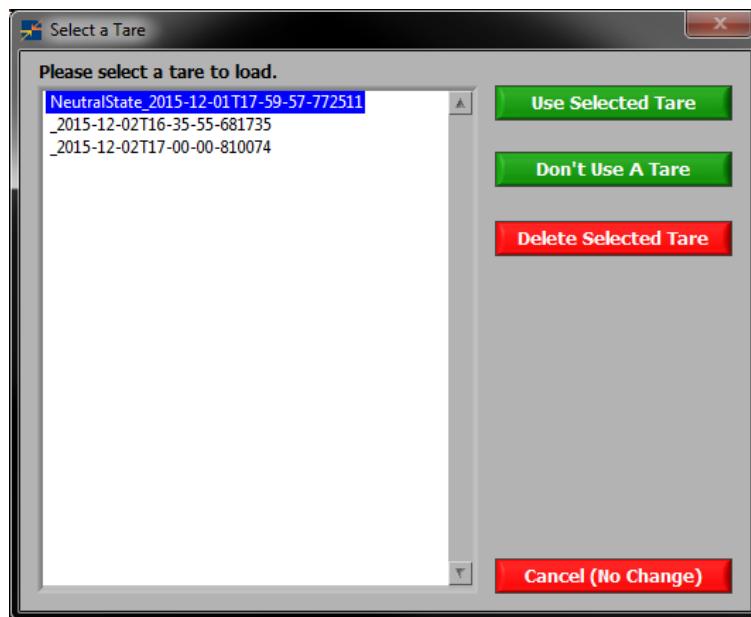


Figure 4-12 Select a Tare window

When a tare is selected, the measurements displayed on the various graphs of the user interface will be calculated with the tare applied.

4.4 Instrument Control

The **Instrument Control** section, located to the right of the **Sensor** section, allows the user to select the measurement type to be displayed in the ODiSI-B software, whether strain or temperature. The length units (meters, feet, etc.) used can also be changed here.



Figure 4-13 Instrument Control section of the ODiSI-B interface

Clicking the **Start Operation** button will initiate continuous measurement of the attached sensor and display of data in the interface. While measuring, the **Start Operation** button will turn red and will be renamed **Stop Operation**. Clicking on the **Stop Operation** button will stop the continuous processing of data.

Note: Data logging is only possible while the software is performing continuous measurements.

Clicking the **Save Tare** button triggers the collection of a new tare. A popup window will appear once the data has been collected asking the user to give the tare a unique name. A date and UTC time stamp will be appended to any name entered.

Once a new tare has been created, it will automatically be applied to the data and will be shown in the **Selected Tare** field in the **Sensor** section of the interface.

The **X-Axis Units** dropdown allows the user to configure the length scale used when referring to the fiber sensor. Available options include: Length (m), Length (mm), Length (ft), Length (in), Time (ns), and Segment Index. The segment index option is the base length unit and labels each measurement gage according to their proximity to the start of the sensor key.

The **Y-Axis Units** dropdown allows the user to change the units of the displayed processed data.

Note: Strain is the only available option when using High-Speed CFG sensors.

- Strain – Displays the calculated strain relative to the selected tare
- Temperature – Displays the calculated change in temperature between the current measurement and the selected tare (in units of °C)
- Frequency Shift – Displays the frequency shift of the light reflected from the fiber between the tare and the measured data. This is the base unit of the ODiSI-B and is scaled to strain or temperature using the coefficients located within the attached sensor's profile.
- Delay - Displays the temporal delay (relative to the reference) of light traveling down the fiber, as it is strained, heated, or cooled.

- Quality Factor - Displays a calculated merit function to assess the quality of the measurement, which scales from 0 to 1. Values above 0.38 indicate that the strain and temperature measurements at the corresponding locations are accurate. See the appendices for more information.

Note: Changing the X and Y axis units impacts the data stored in processed data files but has no impact on data stored in raw data files. Raw data files can be processed into any combination of X and Y units using the post processing software.

4.5 Averaging

The **Averaging** section is located to the right of the **Instrument Control** section at the top of the user interface. The **Averaging** section contains the controls necessary to apply spatial and temporal averaging to the processed data displayed in all graphs of the interface.

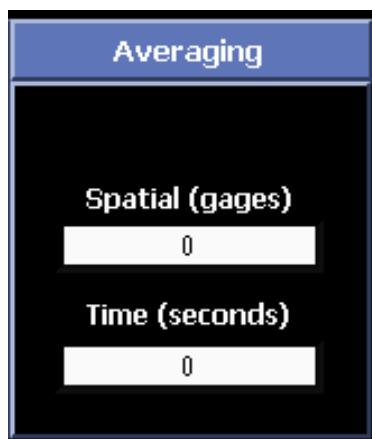


Figure 4-14 Averaging section of the ODiSI-B interface

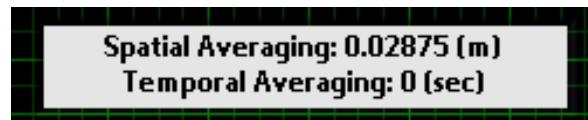


Figure 4-15 Averaging information is overlaid onto the Measurement vs. Length plot

4.5.1 Spatial Averaging

The spatial averaging function applies a rolling average to the data along the length of the fiber; averaging each gage with the gages surrounding it. The width of the rolling average is defined in terms of number of gages. When configuring this type of averaging it is important to remember that the ODiSI-B gages are overlapped by 50%.

4.5.2 Temporal Averaging

The temporal averaging function applies a uniformly weighted rolling average to the data; averaging each gage with its previous measurements over a fixed time window. The unit of the temporal averaging field is seconds. Since this average is only applied to data from prior scans, it will induce latency in the measurement equal to half of the average duration.

Notes:

- Setting the size to 0 for spatial or temporal will turn off the respective averaging.
- If NaNs are present in the data as a result of the quality factor filter, the output will be an average of all of the real data points within the bin. If all data points in the bin are NaNs, the resulting averaged data will also report a NaN.
- Averaging impacts the data displayed in the interface and the data logged using the Processed Text File or Measurement vs Time File functions. Data stored in raw data files (Raw Data Log Files) are not impacted by averaging.

4.6 Data Logging

The **Data Logging** section, located in the upper right corner of the interface, contains the controls and indicators necessary to monitor and configure all data storage available through the ODiSI-B software.

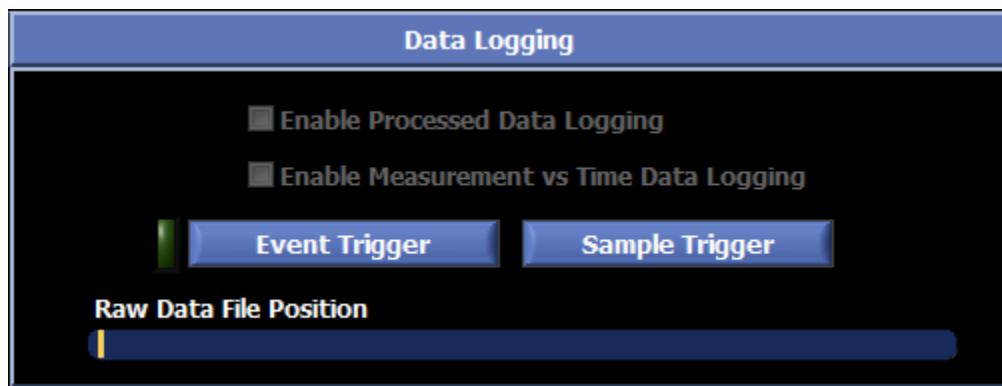


Figure 4-16 Data Logging section of the ODiSI-B interface

Navigate to **Data Logging > Raw Data Logging Settings** or **Processed Data Logging Settings** to configure the ways in which the data is saved.

Data can be stored using any combination of the following:

- Raw Data File – raw data stored at rates up to the full acquisition rate of the ODiSI-B (.odb). These files can have data written to them in a linear or circular fashion.
- Processed Data File – processed (strain or temperature) data at the screen update rate or a lower, user defined, rate (.txt)
- Measurement vs Time File – processed (strain or temperature) data for selected gages at the screen update rate or a lower, user defined, rate (.txt)

4.6.1 Raw Data Files

The ODiSI-B software can store raw data files at rates as high as the full acquisition rate of the ODiSI-B. These files can be loaded into the post processor and converted into strain or temperature measurements. Logic diagrams for the raw data logging and triggering can be found in : Raw Data File and Trigger Logic.

Circular Raw Data File

The **Circular Raw Data File** is a rotating raw data buffer that is written to whenever the **Start Operation** button is clicked. This file is written to at rates up to the full acquisition rate of the ODiSI-B and can be converted to strain/temperature using the post processor.

The location, size, and method of interruption of the Circular Raw Data File can be changed using the Raw Data Storage wizard accessed from **Data Logging > Raw Data Storage Settings**.

The wizard allows for the user to control a range of file properties including:

- File storage location – the user can specify a folder to store data files. This needs to remain on the PC's solid state drive or an external USB3 solid state drive ¹.
- File Base Name – This name will have a timestamp corresponding to the time of the Event trigger appended to it and is used to differentiate between various log files.
- Down Sampling Factor – this factor controls the rate at which raw data will be stored to the file. Data can be stored to the file at rates corresponding to integer fractions of the full acquisition rate of the ODiSI-B.
- File size – this allows users to control how many raw data acquisitions to store in the file before beginning to overwrite old data.
- Trigger Action – this controls how the software will end writing to the log file when an Event Trigger is received.
 - Ignore Trigger – Keep writing to the file even after a trigger is received.
 - Start on Trigger – Once the trigger is received, overwrite the full data file with new data.
 - Stop on Trigger – Once the trigger is received, stop writing to the data file.
 - Center on Trigger – Once the trigger is received, overwrite the oldest half of the data file.

Linear Raw Data File

The **Linear Raw Data File** is comprised of raw data that can be processed into strain or temperature using the ODiSI-B post processor. It is a separate file that can be written to at a rate lower than the acquisition rate of the ODiSI-B.

As with the Circular Raw Data File, the location and size of the Linear Raw Data File can be changed using the Raw Data Storage wizard accessed from **Data Logging > Raw Data Storage Settings**.

The wizard allows for the user to control a range of file properties including:

- File storage location – the user can specify a folder to store data files. This needs to remain on the PC's solid state drive or an external USB3 solid state drive ¹.

- File Base Name – This name will have a timestamp corresponding to the time of the Event Trigger appended to it and is used to differentiate between various log files.¹
- Down Sampling Factor – this factor controls the rate at which raw data will be stored to the file. Data can be stored to the file at rates corresponding to integer fractions of the full acquisition rate of the ODiSI-B.
- File size – this allows users to control the maximum number of acquisitions to store in a single raw data file.

Once created, the Linear Raw Data File is written to once an Event Trigger is received. The file will be written to at the rate specified until the Sample Trigger is received or the file is full. If the file is filled prior to receiving a Sample Trigger, the Raw Data File Position indicator will turn red.

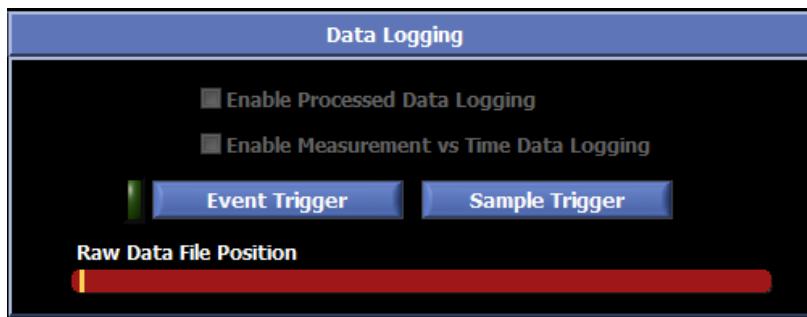


Figure 4-17 Example of the Raw Data File Location indicator turning red once the linear raw data file is full

Note: If the file is filled prior to receiving a Sample Trigger, it is still necessary to issue a Sample Trigger in order to close the existing file and prepare the software to create a new data file.

4.6.2 Processed Data File

The user has the option to save processed strain or temperature data directly to a tab delimited text file as fast as the screen update rate of the ODiSI-B software. Control over the Processed Data File is possible through the **Data Logging Settings** window.

¹ Logging to an external USB 3 solid state drive tested using a Samsung Portable SSD T3

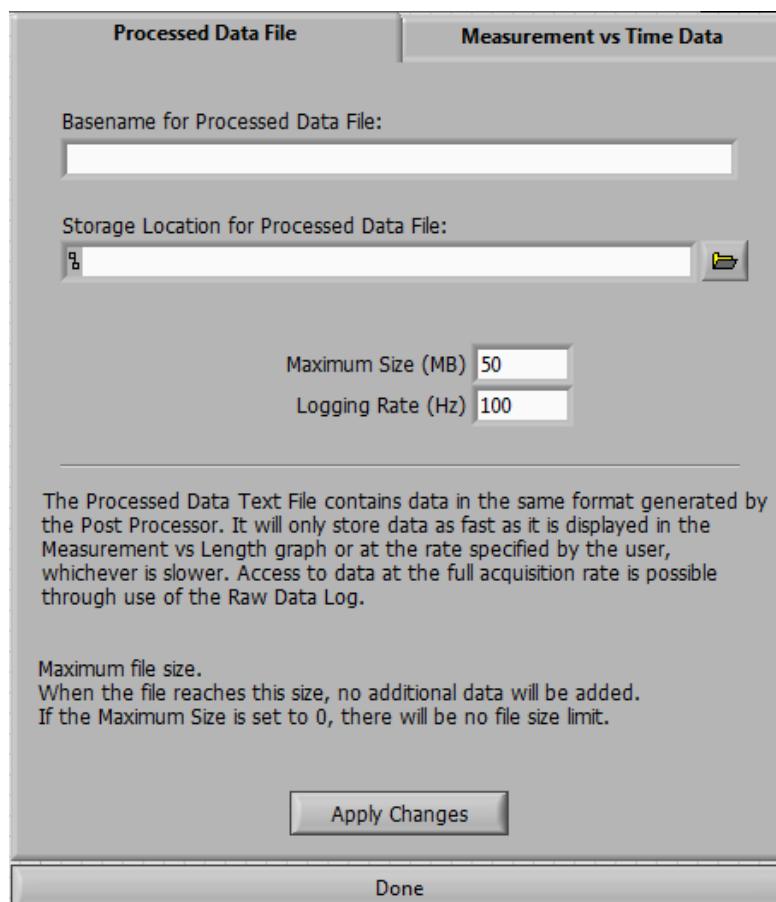


Figure 4-18 Processed Data File tab of the Data Logging Settings window

Within the **Processed Data File** tab of the **Data Logging Settings** window, there are controls that enable the user to change the **Base Name**, **Storage Location**, **Max File Size**, and **Maximum Logging Rate** of the Processed Data File. A date/time will be appended to the base name entered to create a unique file name. The maximum file size control allows the user to place a limit on the size of the text file. The software stops writing to it once this size has been reached. An entered maximum file size of 0 MB will remove any file size limitation and the software will write to the file as long as space is available on the drive. Finally, the **Logging Rate** control allows the user to place a limit on how frequently the ODiSI-B software writes scans to the Processed Data File. If this rate is higher than the screen update rate of the ODiSI-B software, the log file will be written to as fast as possible.

The **Processed Data File** is formatted as a tab delimited text file with a header and a 2D array of strain or temperature data where each row is a different scan and each column is a different gage (location along the fiber). The first column contains the UTC time stamp while the first row contains gage location.

Filename	\\\endor\A-Team\temporary\datafile.txt					
First Column is Timestamp						
First Row is Location (m)						
Matrix is Strain (microstrain)						
-----	0	0.002614	0.005227	0.007841	0.010455	0.013068
2014-08-12T17:13:03.755794	2.08765	4.15394	3.54389	3.00304	1.25921	0.204553
2014-08-12T17:13:03.765789	1.46045	2.69927	1.44265	1.79435	1.2601	0.163225
2014-08-12T17:13:03.775784	4.24355	5.72183	3.80541	3.5637	2.6161	1.32038
2014-08-12T17:13:03.785777	3.03499	4.85323	4.18715	3.19642	2.12016	0.838889
2014-08-12T17:13:03.795772	0.820364	2.36942	1.53977	2.40057	0.965366	-0.13795
2014-08-12T17:13:03.805767	0.844993	4.52286	3.18605	2.9776	2.37473	1.92774
2014-08-12T17:13:03.815759	4.59216	5.46751	3.07043	3.33242	1.27408	0.917864

Figure 4-19 The formatting of the Processed Data File

Note: The data stored to this file will match the data displayed in the Measurement vs. Length graph. This means that any averaging or quality factor filtering applied to the data displayed will be present in the stored data.

Note: A new processed data file will be created each time the ODiSI-B software begins/resumes processing (**Start Operation**).

4.6.3 Measurement vs. Time File

The **Measurement vs. Time Data File** allows the user to save tab delimited text files containing the same data points displayed in the **Measurement vs Time** graph. As with the other methods of data storage, the properties of the Measurement vs. Time Data File can be edited from within the **Data Logging Settings** window.

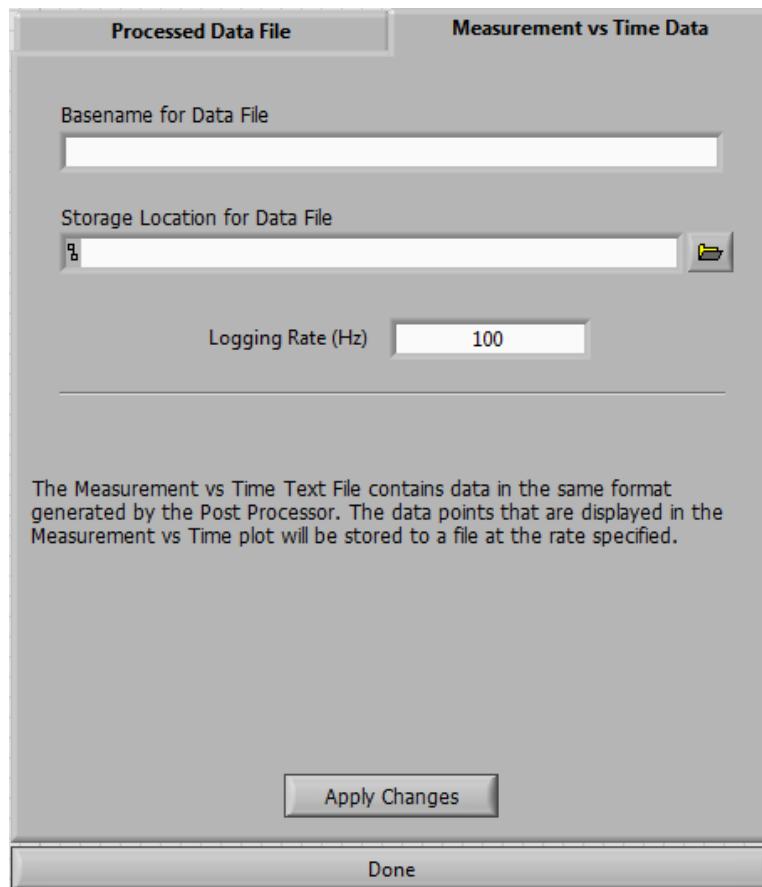


Figure 4-20 Measurement vs Time Data tab of the Data Logging Settings window

Through the Data Logging Settings window it is possible to edit both the **Base Name** and **Storage Location** of the Measurement vs. Time Data File. As with the other tabs within this window, the user must click on **Apply Changes** in order to apply and save any changes to the Measurement vs Time Data File.

Data stored in this file is similar in format to the Processed Data File. However it has an additional row above the location row that contains the title of the point shown in the **Measurement vs Time** display.

Point(s) vs Time				
Matrix is Strain				
First Row is Segment Name				
Second Row is Location (m)				
-----	Small Hole	Medium Hole	Large Hole	Point 6
-----	4.7307	4.7699	4.8195	5.1306
2014-08-13T16:56:23.091609	100.305	111.467	125.92	83.072
2014-08-13T16:57:20.874514	92.011	124.587	129.37	83.97
2014-08-13T16:57:20.984452	91.629	115.87	127.721	83.968

Figure 4-21 Measurement vs Time Data File organization

4.7 Measurement Display

The Measurement Display section encompasses the central section of the interface and contains the controls and indicators necessary to view processed (strain or temperature) data in a number of ways, each contained in a separate tab.

4.7.1 Measurement vs. Length

The **Measurement vs. Length** tab is the default view of the processed data and allows the user to view the strain or temperature data as a function of length down the fiber sensor. It is important to note that the Touch to Locate, Measurement vs. Time, and Plot to Picture tabs all utilize the data displayed within this plot.



Figure 4-22 Measurement vs. Length tab of the ODiSI-B interface

The **Measurement vs Length** graph can be manipulated using the graph property buttons located just below the graph on the left hand side of the screen.

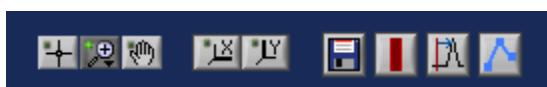


Figure 4-23 Graph property buttons

Scaling

The graph can be scaled either manually or automatically. Automatic scaling is carried out by clicking on the buttons to rescale the X or Y axis independently. The scaling can be changed manually by double clicking on the first or last axis scale marking and entering a new value.

Zooming

The user can zoom into a specific region of the graph by manually adjusting the scale or using the built in zoom controls. Clicking on the button will open up a menu containing the zoom controls.

Zoom Window

This tool allows the user to zoom in on a rectangular region of the **Measurement vs. Length** graph. To use this tool, first select it from the zoom menu button. Then left click and drag to create a box within the **Measurement vs. Length** graph. Once the left mouse button is released the X-axis and Y-axis scales will change to show only the selected region.

X-Axis Zoom

This tool allows the user to zoom in on a region of the X-axis only, without adjusting the Y-axis scale. To use this tool, first select it from the zoom menu button. Then left click and drag to create a horizontal window within the **Measurement vs. Length** graph. Once the left mouse button is released the X-axis scale will change to show only the selected region.

Y-Axis Zoom

This tool allows the user to zoom in on a region of the Y-axis only, without adjusting the X-axis scale. To use this tool, first select it from the zoom menu button. Then left click and drag to create a vertical window within the **Measurement vs. Length** graph. Once the left mouse button is released the Y-axis scale will change to show only the selected region.

Zoom to Fit

Selecting this button from the zoom menu returns the **Measurement vs. Length** graph to the full scale of all data. This is the same as clicking both the and buttons.

Zoom In

This tool allows the user to zoom in (magnify) the **Measurement vs. Length** graph from the point where the cursor is clicked. To use this tool, first select it from the zoom menu button. Then click once to magnify once; or click, hold and release to magnify several times at the cursor location.

Zoom Out

This tool allows the user to zoom out from the point where the cursor is clicked. To use this tool, first select it from the zoom menu button. Then click once to zoom out once; or click, hold and release to zoom out several times at the cursor location.

Panning

The Pan tool allows the user to move the data across the X-axis and Y-axis in one motion. To use the pan tool, select it from the graph property button group and then click and drag on a point within the graph. This will move the center of the graph without making any changes to the axis scales.

Save Tool

The user can save the current graph as a .bmp, .jpg, or .png image file by clicking on .

Vertical Cursor

The vertical cursor allows the user to view the strain or temperature reading at a specific x axis location. To enable it click on the red cursor button and a red cursor will appear on the graph. To move the cursor to a new position, click on the button and then click and

drag the red cursor to a new position on the graph. The X and Y positions of the data where the cursor intersects the measurement data will be displayed in the upper right corner of the **Measurement vs Length** plot. Clicking the  button will move the vertical cursor to the highest nearby measurement point.

Line Style

The line style button  toggles the **Measurement vs. Length** plot's data display between showing a series of individual points or a line that is created through linear interpolation of the measurement points.

Quality Threshold

The **Quality Factor**, described in the Instrument Control section (4.4), is a value that describes the validity of the spectral shift measurement. This threshold replaces any data on the plot that does not have a quality above the specified value (0.38 by default) with a NaN.



Figure 4-24 Quality Threshold control

Note: Quality factor is not available for High-Speed CFG sensors.

Vibration Correction

The controls for the Quality Threshold and Vibration Correction are located below and to the right of the Measurement vs. Length plot.



Figure 4-25 Vibration Correction controls

The ODiSI-B software has the capability of removing the effects of vibration in the standoff. This correction relies on a 10cm to 20cm of unstrained fiber located shortly after the fiber sensor connector. This is the reason sensors supplied by Luna have 20cm of Teflon tubing after their connector.

The vibration correction button  toggles the visibility of the vibration correction region on the data in the **Measurement vs. Length** graph. The section of fiber used for vibration correction will show up as an additional orange trace. The region used for this correction can be adjusted by changing the field to the right of the vibration correction toggle.

Note: The vibration correction region should be placed as close to the start of the sensor as is possible. If large amounts of vibration correction errors are present, move this region down the sensor in small increments until the frequency of the errors is reduced.

Note: Vibration correction is not available for High-Speed CFG sensors.

4.7.2 Touch to Locate

Touch to Locate is the process by which a user can map locations of interest in their test article to the length scale of the fiber sensor. This can be done by identifying locations of interest using a hot or cold point touch to the fiber, resulting in a temperature spike along the

sensor. As examples, a hot touch can be applied with the tip of a soldering iron set to low heat, and a cold touch can be applied with a Q-tip sprayed with canned air. The **Touch to Locate** tab contains tools that enable the user to walk through this process in a more automated way. When points and regions of interest have been identified, they can be saved to a .map file and used throughout other regions of the ODiSI-B software to focus on the data relevant to the users test.

Prior to going through the Touch to Locate process, it is recommended that the user draw a diagram of the fiber routing over the test article with markings that denote points or regions of interest.

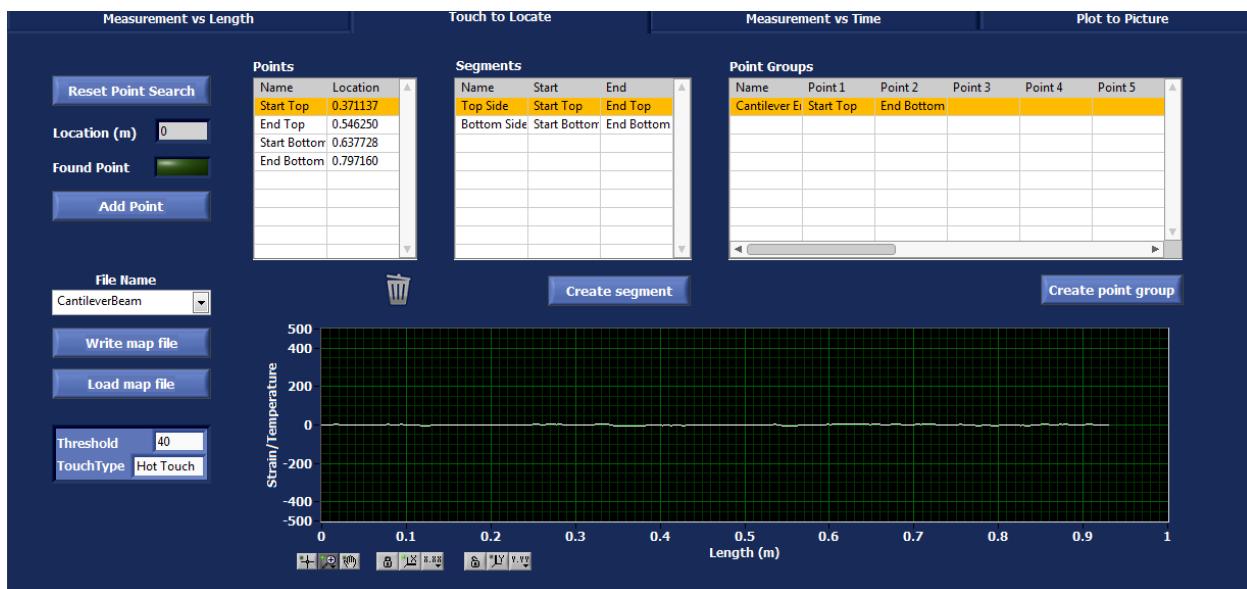


Figure 4-26 Touch to Locate tab of the ODiSI-B interface

The **Touch to Locate** tab can be broken into six functional areas:

- Point Identification Settings** – These settings need to correspond to the method used to induce a localized strain or temperature change in the fiber. The **Threshold** sets the absolute level that a persistent strain or temperature peak must pass in order to be considered a point of interest. The user should look at the Touch to Locate graph and set this threshold based on the spread about zero of measurement values. The **Touch Type** control changes whether the algorithm is looking for peaks or valleys in the data.



Figure 4-27 Point Identification Settings section of the Touch to Locate tab

- Point Detection Controls** –These buttons are used to identify and add points to the map file. Clicking on the **Reset Point Search** button will reset the point finding algorithm and apply a tare to the plot. Localized strain or temperature events will

be evaluated from this moment onward. When the Touch to Locate algorithm finds a persistent point, it will populate the **Location** indicator and turn on the green **Found Point** light. If the reported location matches the location at which the strain or temperature event occurred, the user can click **Add Point** to save the point.



Figure 4-28 Point Detection Controls

The names and locations of the points can be edited by triple clicking on the name or location in the **Points** field and typing a new value in. It is useful to assign detailed point names if the map file is intended for use in other areas of the ODiSI-B software.

3. **Measurement Display** – This section contains a small Measurement vs. Length plot that can be used to check the validity of the Touch to Locate point finding routine.
4. **Map File** - The map file section contains the recorded **Points**, **Segments**, and **Point Groups** gathered for a particular sensor. Once a series of points have been collected using the Point Detection controls, they can be used to construct segments and point groups. At any time, points, segments, or point groups can be removed by clicking and dragging the desired cell into the trash can located below the **Points** field.

Points	
Name	Location
Start Top	0.371137
End Top	0.546250
Start Bottom	0.637728
End Bottom	0.797160

Segments		
Name	Start	End
Top Side	Start Top	End Top
Bottom Side	Start Bottom	End Bottom

Point Groups					
Name	Point 1	Point 2	Point 3	Point 4	Point 5
Cantilever E	Start Top	End Bottom			

Figure 4-29 Map File section of the Touch to Locate tab

Clicking on the **Create Segment** button will open up a window where the user can enter a segment name and choose a start and end point from the list of points already gathered (by clicking on the Segment Start and Segment End fields).

Note: The start point must occur before the end point (when starting from the sensor connector).

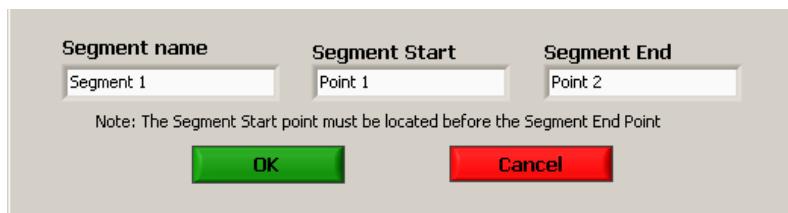


Figure 4-30 New Segment window

Clicking on the **Create Point Group** button will add a new group to the Point Group array. Users can then drag and drop points from the Points section into the row of the group they belong to.

5. **Map File Saving and Loading** – This section contains the tools needed to write a new map file or load a previously created map file. If creating a new file, enter the desired map file name into the blank **File Name** dropdown and click **Write Map File**. A popup will appear telling the user that the map file was saved successfully.

Map files will be saved to the folder associated with the identified sensor within the sensor repository.

If loading a map file, use the **File Name** dropdown to select the map file from a list of files associated with the identified sensor and click **Load map file**.

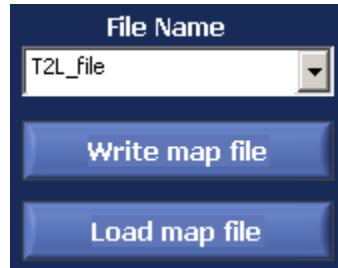


Figure 4-31 Map file saving and loading controls

4.7.3 Measurement vs. Time

The **Measurement vs. Time** tab allows users to display select measurement points, segments, or point groups as a function of time. This is particularly useful when comparing ODiSI-B measurements to those gathered using traditional point sensors such as foil strain gages or thermocouples.

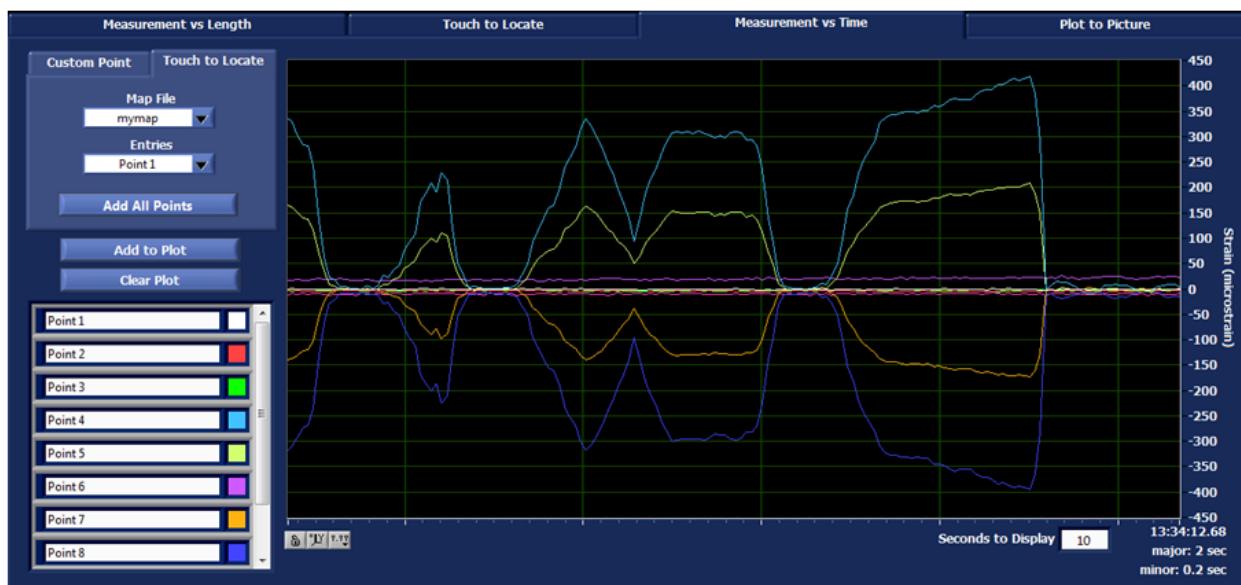


Figure 4-32 Measurement vs. Time tab

The **Measurement vs. Time** display can be broken into two distinct sections, **Point Addition** and the **Measurement vs. Time** graph.

1. **Point Addition** – This section contains the controls necessary to add traces to the graph manually or from a map file associated with the identified sensor.

If creating a trace manually, first select the **Custom Point** tab. Enter a name followed by a trace start point. In order to display the data from a segment of fiber, click the **Display Segment** checkbox and enter an end point as well. Once the desired name, start, and end points are chosen, click **Add to Plot** to add the trace to the **Measurement vs. Time** graph.

If creating a trace from data stored in a map file, first select the **Touch to Locate** tab. Select the relevant map file and then pick a point, segment, or point group from the list. When the desired trace is selected, click **Add to Plot** to add the trace to the **Measurement vs. Time** graph.



Figure 4-33 Point Addition Controls

It is possible to change the colors of the traces by left clicking on the color box to the right of the point or segment name in the plot legend. This will open up a color browser that contains recently used colors as well as a color mixer.

2. Measurement vs. Time Graph – contains the time domain plot.

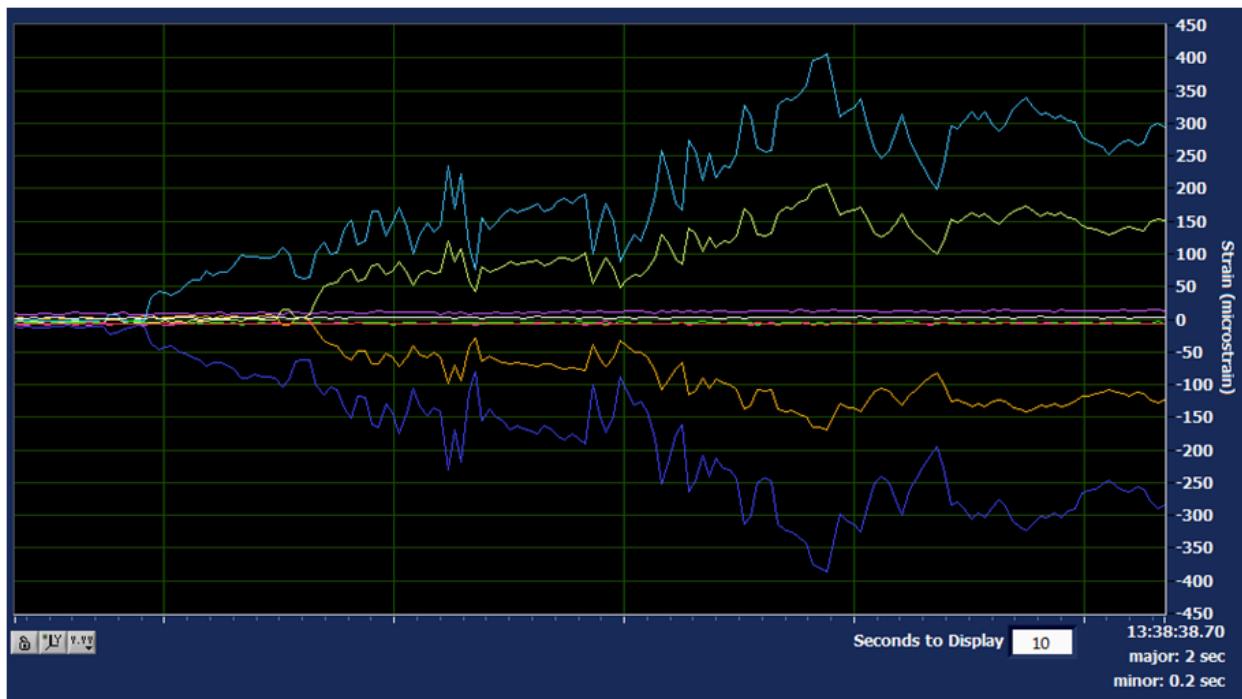


Figure 4-34 Measurement vs. Time graph and graph controls

New data is added to the right of the graph. Older data scrolls to the left. The lower right corner of the graph shows the time of the most recent data and indicates the spacing of the major and minor tick marks on the graph.

Seconds to Display controls the amount of data displayed in the graph. The x-axis legend and the update rate of the graph will be automatically updated in order to hold the requested amount of data.

4.7.4 Plot to Picture

The **Plot to Picture** feature allows users to display strain or temperature from regions of the fiber sensor as a color map overlaid onto a picture of their test article. This provides the user with the ability to quickly assess the strain or temperature distribution over their part and helps them identify the location of strain concentrations or hot spots.

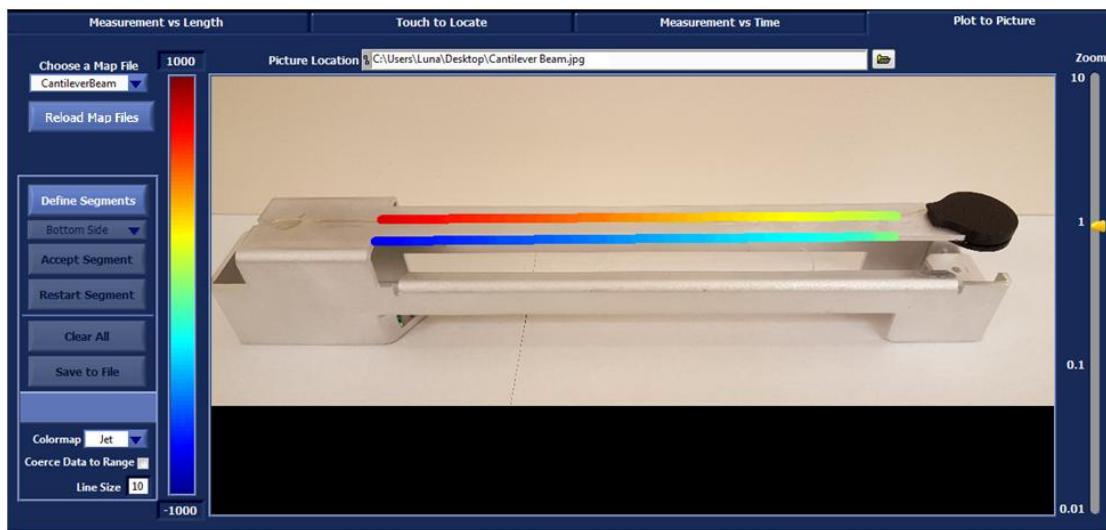


Figure 4-35 Plot to Picture tab in the ODiSI-B interface

This section of the software can be broken into three functional areas; the Picture Display, the Color Controls, and Segment Management.

1. Picture Display

The **Picture Display** section is comprised of the picture region, the zoom tool, and the picture loading control. Using the **Picture Location** file path control, browse to and select a picture to load into the display. The display can accept PNG, JPG, and BMP image file types. When a file has been selected, it will be loaded into the display and a zoom setting will be chosen to fit the entire image on the screen.

Note: Although the **Plot to Picture** display can accommodate images with high resolution it is good practice to scale the pictures down to a size closer to that of the display (1000x500 px) in order to reduce the memory used by the software.

The user can zoom into regions of the image by adjusting the **Zoom** level on the right side of the screen and then using the vertical and horizontal scroll bars on the edges of the image display.

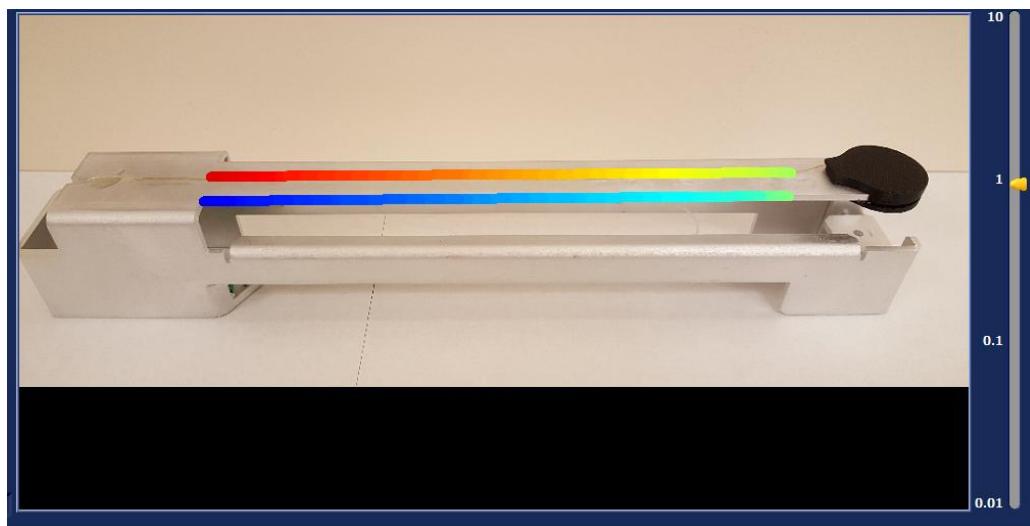


Figure 4-36 Picture Display section

2. Segment Management

A key part of the Plot to Picture process is defining the location of the sensor segments on the test article. This is done through use of the **Segment Management** controls located to the left side of the screen.



Figure 4-37 Segment Management controls

The first step in the plot to picture process is to choose the map file that contains the segments that the user wishes to plot. Map files can be selected using the **Choose a Map File** dropdown menu. If a fiber path has already been drawn and saved it will be loaded automatically when a map file is selected. If the desired map file was created

Note: If no map files have been created for the connected fiber sensor, use the Touch to Locate feature to create one. Click the **Reload Map Files** button to use it in the **Plot to Picture** tab.

To enable the **Segment Definition** controls, first ensure that the ODiSI-B software is not processing by clicking **Stop Operation**. This will enable the **Define Segments** button. Click on **Define Segments**, this will change the plot to picture software into a configuration mode that enables the **Segment Dropdown**, **Accept Segment**, **Restart Segment**, **Clear All**, and **Save to File** buttons.

Follow the steps below in the order shown to plot individual segments of the sensor data to the screen.

1. Select the desired segment from the map file using the **Segment Dropdown** list.
2. Click the points on the picture that define the path of the fiber. As new points are added, the screen will update with a green line connecting the points. There must be at least a start and end point, but a segment can be made to curve by drawing it in many short sections between the start and end point.

Click **Accept Segment** if the displayed line is satisfactory

Note: It is possible to overwrite a segment that has already been created by selecting it and redrawing the segment.

3. If the line is not satisfactory, click on the **Restart Segment** button

When all segments have been mapped to the image, save these paths to the map file by selecting **Save to File**. Clicking **Clear All** will remove all segment traces from the screen and allow the user to start over.

At this point, the user can click on the **Defining** button to exit the configuration mode and allow the ODiSI-B software to resume processing data. Once the ODiSI-B resumes processing, the measurement data will be displayed on the lines created.

3. Color Controls

The **Color Controls** section allows the user to adjust the scale of the strain or temperature information displayed. To the right of the picture, a color bar displays the active color spectrum. The numbers above and below the color bar are controls that allow the user to change the strain or temperature values that correspond to the extremes of the color range (click on these fields to edit the range).

The box to the lower left of the color bar provides access to three controls over the sensor segments being displayed. First is the **Colormap** dropdown. This allows the user to toggle between three different color schemes that can be used: Jet, Hot, and Cold.

By default, strain or temperature measurements that fall outside of the range defined by the color bar will be represented by a color that contrasts the selected color map (pink

for Jet and green for Hot and Cold). The user can instead choose to display these measurements as the colormap limits by checking the **Coerce Data to Range** checkbox.

Finally users can also change the width of the displayed segments by changing the **Line Size** value (units of pixels).

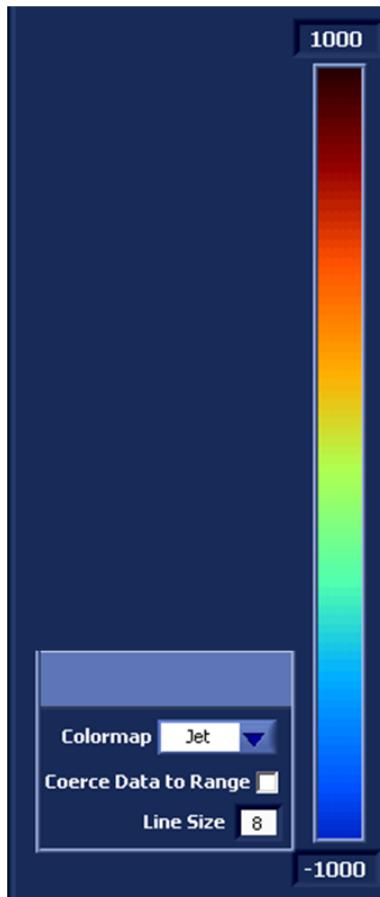


Figure 4-38 Plot to Picture color controls

4.8 Raw Data Post Processor

The ODiSI-B **Post Processor** software allows users to reprocess .odb files gathered using the sample or event triggered logging. These files can be processed into strain or temperature and the results can be saved to a tab delimited text file for viewing outside of the ODiSI-B software.

4.8.1 Software Startup

As with the ODiSI-B control software, the **ODiSI-B Post Processor** is accessible through a desktop shortcut or through the start menu within the Luna Technologies folder. Upon launching, the software will prompt the user to select a raw data file for processing. Once a file is selected, the post processor will load information from the sensor repository and enable the interface.

The **Post Processor** is very similar in layout to the control software. The majority of the software functions in the same way as the control software already covered. The following

sections will describe in detail the features that are different when compared to the control software.

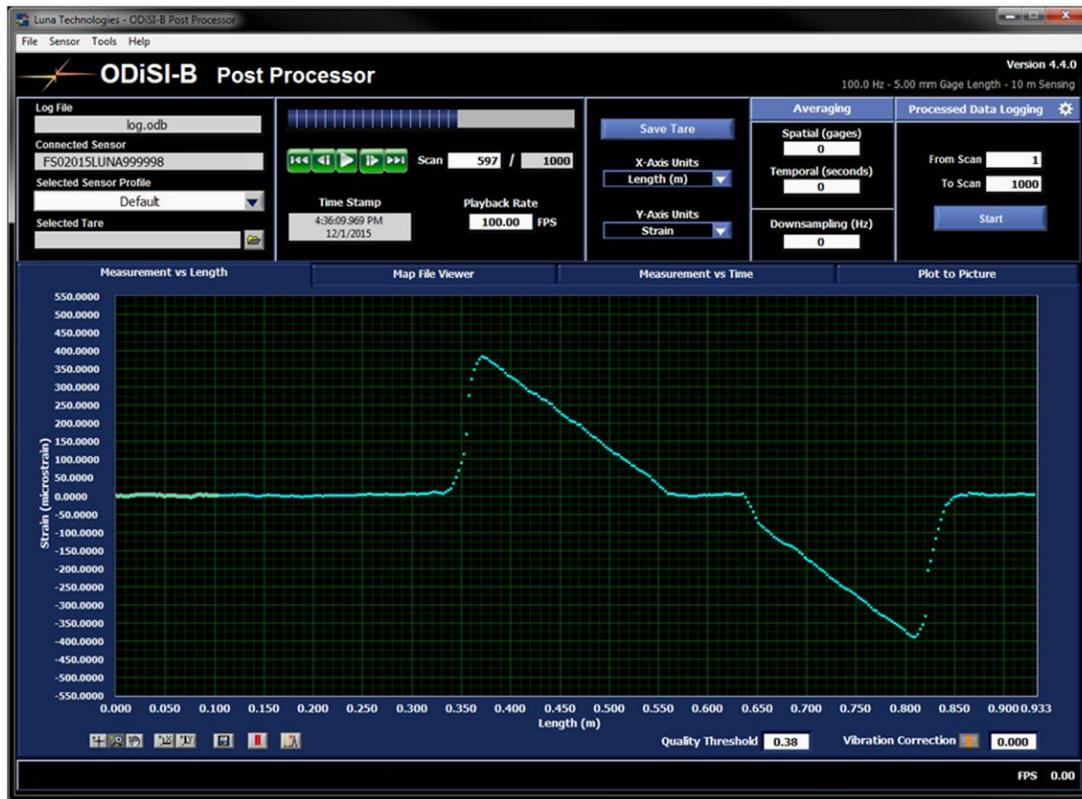


Figure 4-39 ODiSI-B Post Processing Software

4.8.2 File and Sensor Information

This section of the post processor contains file and sensor specific information. Details such as the log file being processed, corresponding sensor, profile, and selected tare are displayed here. As with the control software, users can select sensor profiles or apply tares for data processing.

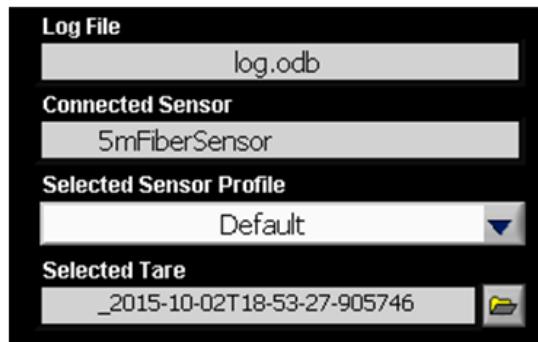


Figure 4-40 File and Sensor Information for High Definition Sensors



Figure 4-41 File and Sensor Information for High-Speed CFG Sensors

When post processing High-Speed CFG Sensor data, the first scan in the file is used as the tare if a tare file is not selected using the **Selected Tare** control. This means that the first scan will show strain of all zeros, and all other scans will show strain relative to the first scan. If a tare is loaded, then all scans (including the first) will show strain relative to the selected tare.

4.8.3 Playback Controls

The **Playback Controls** section contains the buttons necessary to manage playback parameters. The progress bar at the top indicates the location of the current scan within the entire raw data file. It can be dragged back and forth to move to a different scan in the raw data file. The **Scan** field can also be edited to move to a new location in the file.

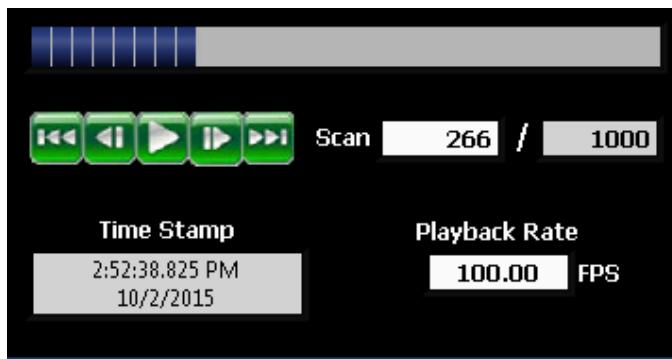


Figure 4-42 Playback Controls Section

The green playback buttons below the progress bar allow users to play through the data file. The button takes the processing software back to the first scan in the log file. The button takes the software back a single scan from where it currently is processing within the log file. The continuous playback button instructs the software to play through the data file one scan at a time until it reaches the end of the file. The button instructs the software to move to the next scan in the log file. The button instructs the software to move to the final scan in the file.

The **Time Stamp** field displays the time stamp of the current scan. The time value displayed is the time that that particular scan was acquired by the ODiSI-B while in the control software (shown as UTC).

The **Playback Rate** field allows the user to change the rate at which raw data is processed into strain or temperature when using the continuous playback button . Values can range from 1Hz to 100Hz. If the value entered is faster than the **Post Processor** can process data, the software will process the raw data into strain or temperature as fast as possible and display the actual rate in the **FPS** indicator located in the lower right corner.

4.8.4 Averaging

The **Averaging** feature is the same as the control software. As with the control software, spatial and temporal averaging are controlled in units of gages and seconds respectively.

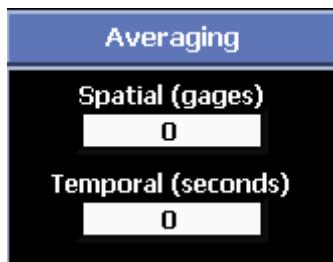


Figure 4-43 Post Processor Averaging

4.8.5 Processed Data Logging

The **Processed Data Logging** section contains the controls necessary to configure and save processed data to a tab delimited text file. As with the control software, the data logging settings can be accessed by clicking on the white gear in the upper right corner. This opens up a window in which the user can configure the names and locations of the two types of processed data files.

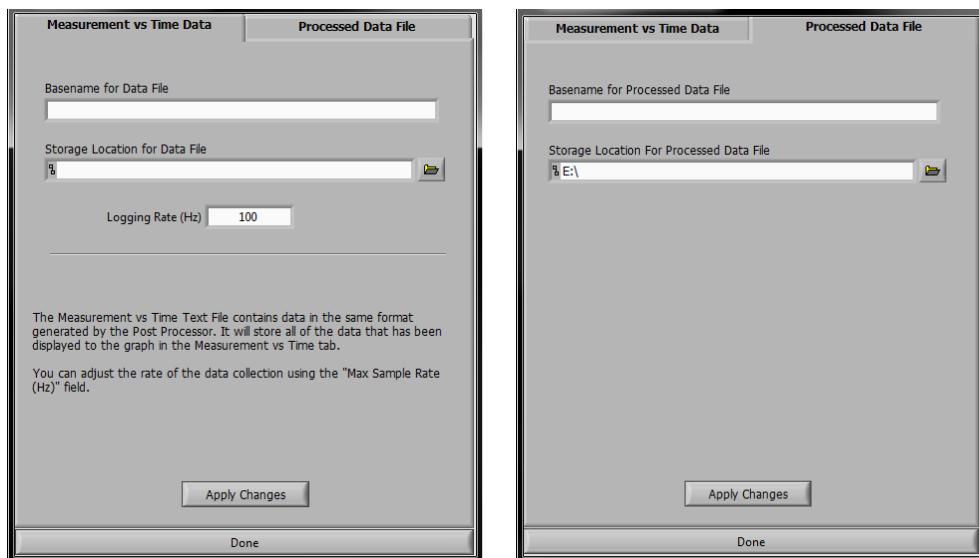


Figure 4-44 Processed Data Logging Settings window

As with the control software, the **Processed Data File** contains the strain or temperature information of the entire fiber sensor for every scan processed. It is formatted as a tab delimited text file with a header and a 2D array of strain or temperature data where each row

is a different scan and each column is a different gage (location along the fiber). The first column contains the timestamp while the first row contains the gage location.

Filename	\\\endor\A-Team\temporary\datafile.txt					
First Column is Timestamp						
First Row is Location (m)						
Matrix is Strain (microstrain)						
-----	0	0.002614	0.005227	0.007841	0.010455	0.013068
2014-08-12T17:13:03.755794	2.08765	4.15394	3.54389	3.00304	1.25921	0.204553
2014-08-12T17:13:03.765789	1.46045	2.69927	1.44265	1.79435	1.2601	0.163225
2014-08-12T17:13:03.775784	4.24355	5.72183	3.80541	3.5637	2.6161	1.32038
2014-08-12T17:13:03.785777	3.03499	4.85323	4.18715	3.19642	2.12016	0.838889
2014-08-12T17:13:03.795772	0.820364	2.36942	1.53977	2.40057	0.965366	-0.13795
2014-08-12T17:13:03.805767	0.844993	4.52286	3.18605	2.9776	2.37473	1.92774
2014-08-12T17:13:03.815759	4.59216	5.46751	3.07043	3.33242	1.27408	0.917864

Figure 4-45 Sample processed data file

The **Measurement vs. Time Data** file is created in the same way as the matching file from the control software. Refer to the **Measurement vs. Time File** section for more information.

4.9 Measurement Streaming

The ODiSI-B can stream measurement data using two different protocols:

- ODiSI Measurement Streaming Protocol
- MTS AeroPro DDI

ODiSI Measurement Streaming Protocol (OMSP) is available on all ODiSI-B units. OMSP is a simple protocol that allows developers to receive ODiSI data in their own programs. Information the protocol and how to develop client code is available in [Section 6: ODiSI Measurement Streaming Protocol](#).

The ODiSI-B can export data to MTS AeroPro using the MTS DDI interface. The AeroPro export feature must be enabled in the ODiSI-B instrument in order to enable this feature. AeroPro must be installed separately. Users must have an AeroPro license and an AeroPro DDI license key.

4.9.1 What data is streamed?

The ODiSI-B streams data at the screen update rate while the ODiSI-B is processing data. The Post Processor has no streaming capability; data can only be streamed while it is being acquired, and only at the rate that the system is capable of processing it.

Points and segments must be defined in a map file before they can be streamed. The user selects a map file to use when enabling streaming. All points and segments defined in the selected map file will be streamed.

Measurement data is streamed as a collection of named scalar values. AeroPro receives these values as “signals”, while OMSP refers to them as “gages”.

Points are streamed using the names they were given in the map file. Segments are expanded into a list of individual values (one for each possible point along the segment) and are named with the name of the segment followed by an underscore and a sequential number. For example, a segment named SegA will be streamed as a series of points named SegA_01, SegA_01, SegA_02, ... etc. These are the names that will be received by AeroPro or by the OMSP client.

4.9.2 Enabling Measurement Streaming

After a sensor has been selected, select *Tools, Measurement Streaming...* from the ODiSI-B program menu.

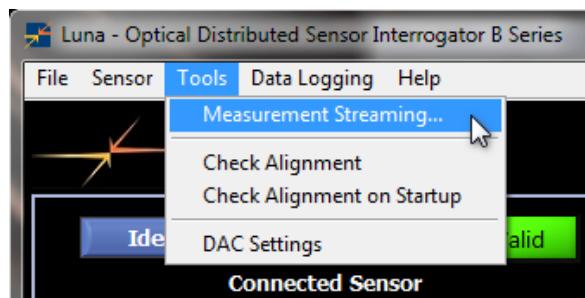


Figure 4-46 Measurement Streaming menu option

In the dialog window, select *ODiSI Measurement Streaming Protocol* or *AeroPro DDI* to enable one of the streaming protocols. Select *None* to disable streaming.



Figure 4-47 Measurement Streaming Settings dialog

OMSP and DDI offer slightly different streaming options. If enabling OMSP, select the map file that contains the points and segments that should be streamed to the client. Specify a test name, and a TCP port number on which to communicate. The OMSP client will need to know the port number that the ODiSI-B is streaming on. Select OK to save these settings.

When using OMSP, the Data Streaming Options dialog can be reopened to change the test name or the selected map file without losing the connection to the OMSP client. If the TCP port number is changed, then any active connection will be closed.

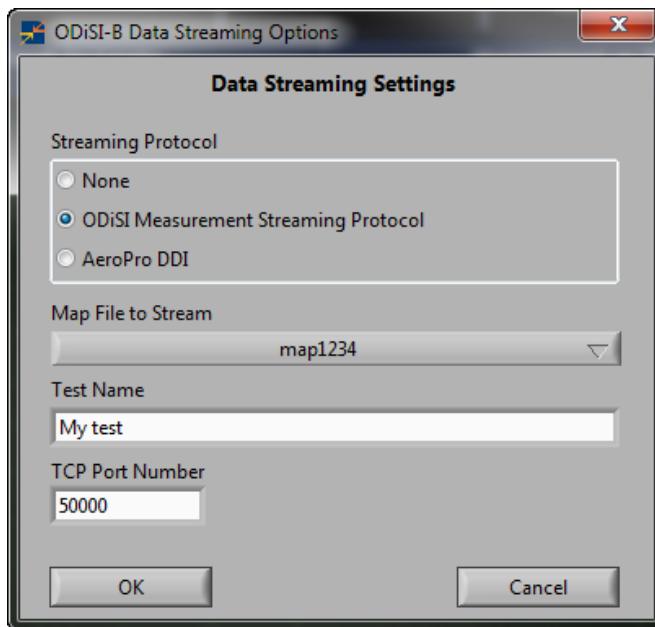


Figure 4-48 OMSP streaming options

If enabling AeroPro DDI, select the select the map file that contains the points and segments that should be exported. Select *OK* to save these settings.

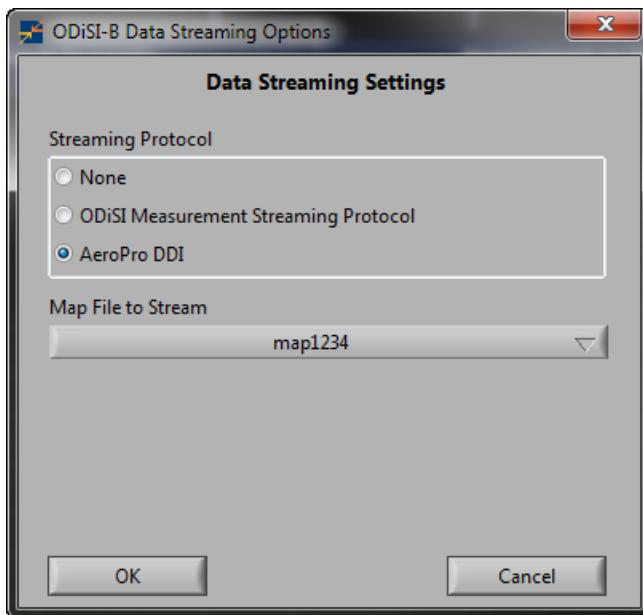


Figure 4-49 AeroPro DDI streaming options

4.9.3 Streaming Indicator

An indicator in the bottom left of the ODiSI-B program window shows that measurement streaming been enabled. In addition to showing which type of streaming has been enabled, the indicator also shows the selected map file. When OMSP is enabled, an additional indicator shows whether an OMSP client is connected.

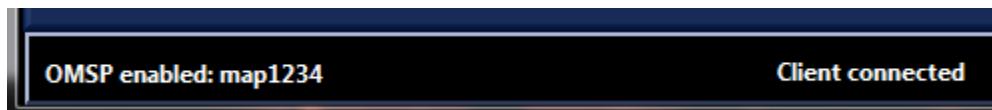


Figure 4-50 OMSP streaming indicators (with client connected)

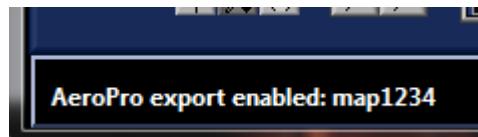


Figure 4-51 AeroPro streaming indicator

4.9.4 Example OMSP Client

An example OMSP client is installed with the ODiSI-B software. Source code for this client is available in the SDK. (See [Section 6.4 OMSP Source Code Examples](#) for details.)

The client is intended as a demonstration of the OMSP streaming capabilities. The client has no capability to store files, export data, or do anything other than display live data as it is streamed. Users can copy and expand the client source code to suit the needs of their test environment.

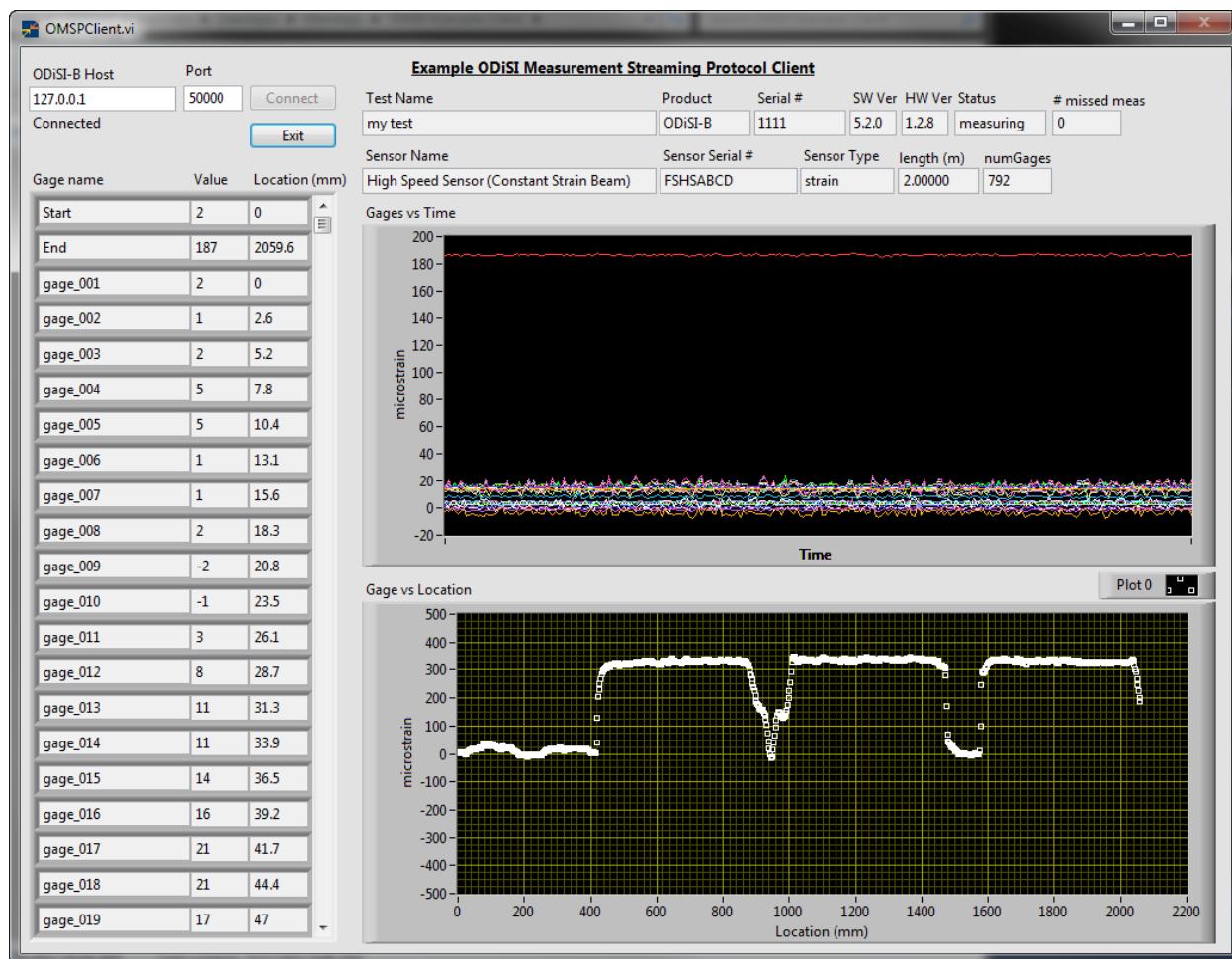


Figure 4-4-52 Example OMSP Client

To use the example client, first enable OMSP streaming in the ODiSI-B software. Then start the OMSP Example Client. Specify the ODiSI-B host name or IP address (127.0.0.1 can be used if the client is running on the same PC as the ODiSI-B software). Specify the TCP port that was selected in the ODiSI-B OMSP options, and press Connect.

The test information and list of gage names and locations should be populated soon after connecting, when the client receives the first Metadata message. When the ODiSI-B starts processing, the gage values will be displayed and the data will be displayed in the two graphs.

The top graph plots the gage values as they are received (similar to the Measurement vs Time display in the ODiSI-B software). Only the first 25 gage values are displayed in this graph. This number can be changed in the included source code.

The bottom graph plots the gage values against the gage locations (similar to the Measurement vs Length display).

4.9.5 Luna MTS Data Server

The ODiSI-B uses a program called *Luna MTS Data Server* in order to facilitate the DDI communication to AeroPro. Once the user has enabled AeroPro export and selected a map

file, the *Luna MTS Data Server* icon is displayed in the Windows system tray in the lower right corner of the Windows desktop.

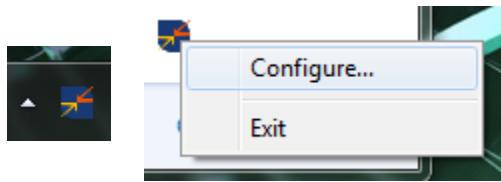


Figure 4-53 Luna MTS Data Server icon is located in the system tray

The user can right-click the icon and select *Configure* to bring up a number of options.

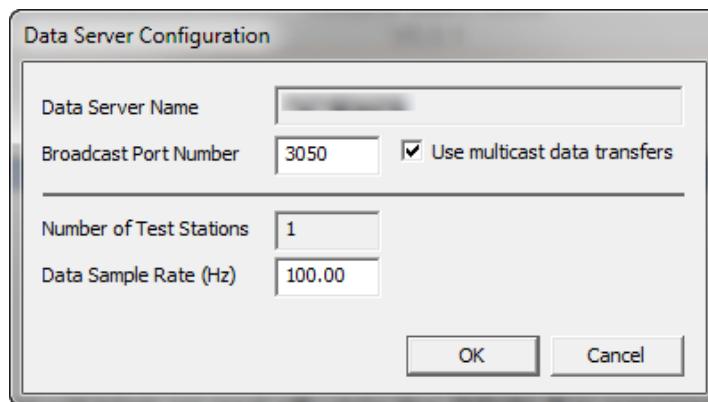


Figure 4-54 Data Server Configuration window

Data Server Name shows the server name which AeroPro must connect to in order to receive the exported ODiSI-B data. Normally, this will be the same as the PC name. This value cannot be changed from this Configuration dialog. It can be set by running the AeroPro setup.

Broadcast Port Number shows the port that AeroPro must listen to in order to receive the exported ODiSI-B data. It may be necessary to change this value if other data providers are already using the default port number.

Number of Test Stations is also an informative value and cannot be changed. For the ODiSI-B, this value should always be 1.

Data Sample Rate (Hz) is the maximum rate at which data will be sent from the ODiSI-B to AeroPro.

4.9.6 Importing ODiSI-B Data in AeroPro

The use of AeroPro is outside the scope of this document. However, this section will describe the basics of importing ODiSI-B data into an AeroPro test.

1. Start Processing in the ODiSI-B before importing data into AeroPro. AeroPro cannot see the names of the exported signal values unless they are actively being streamed.
2. In AeroPro, select the Configuration tab. Right-click the test name and select New, External Connection. Note that the ODiSI-B must already be actively exporting data to

AeroPro (with the green indicator light blinking) before AeroPro will be able to see and import any external signals.

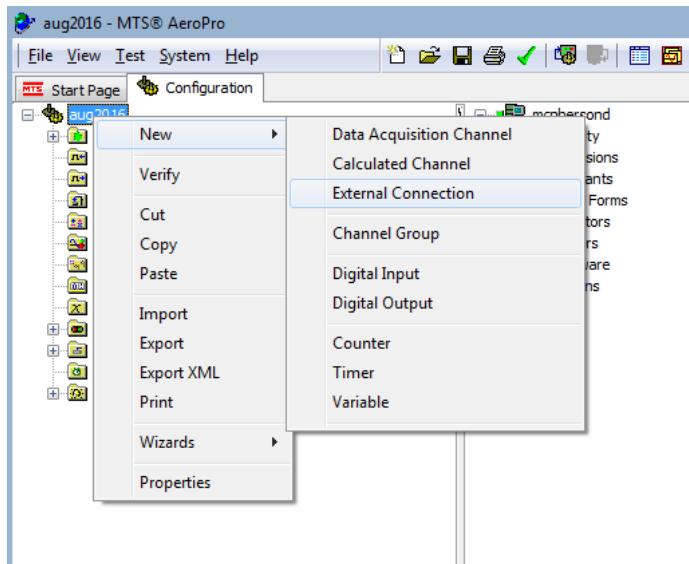


Figure 4-55 Establish an external connection

3. Select DDI as the type of external connection.

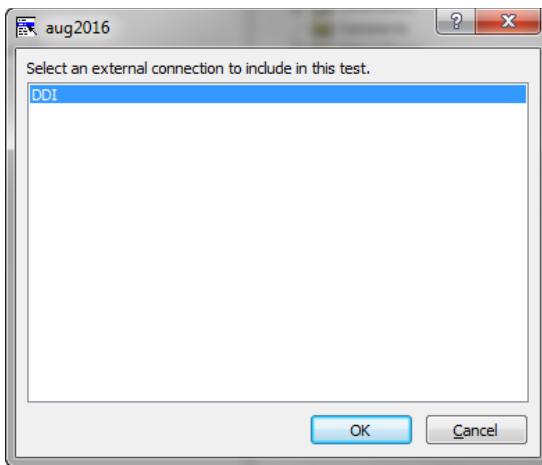


Figure 4-56 Select the DDI connection type

4. Follow the Setup External Data Connection Wizard.

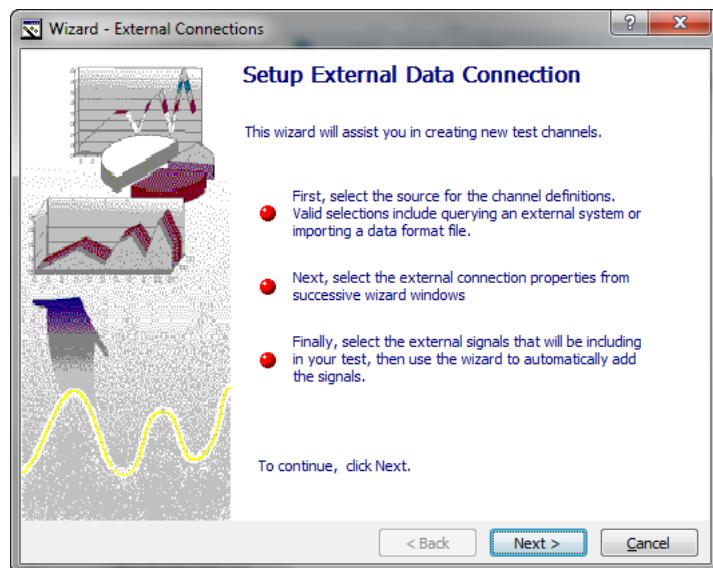


Figure 4-57 External Data Connection wizard

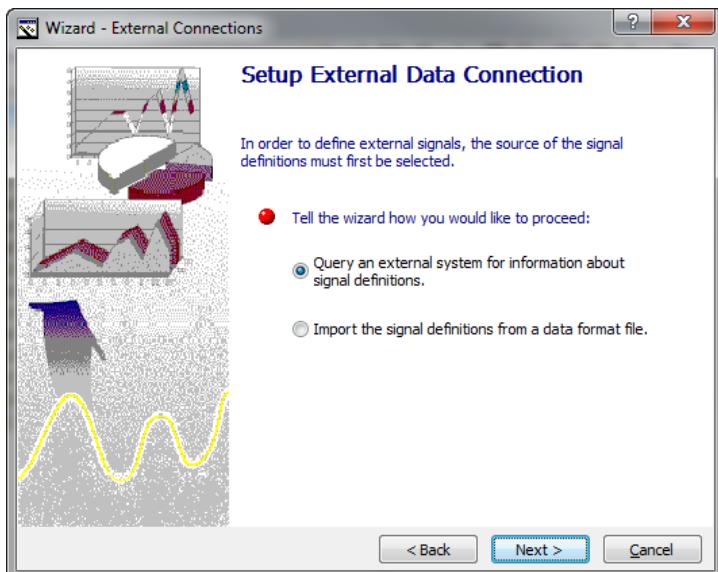


Figure 4-58 Query an external system for signal definition

5. Select DDI and “Standard TCP/IP Connection”.

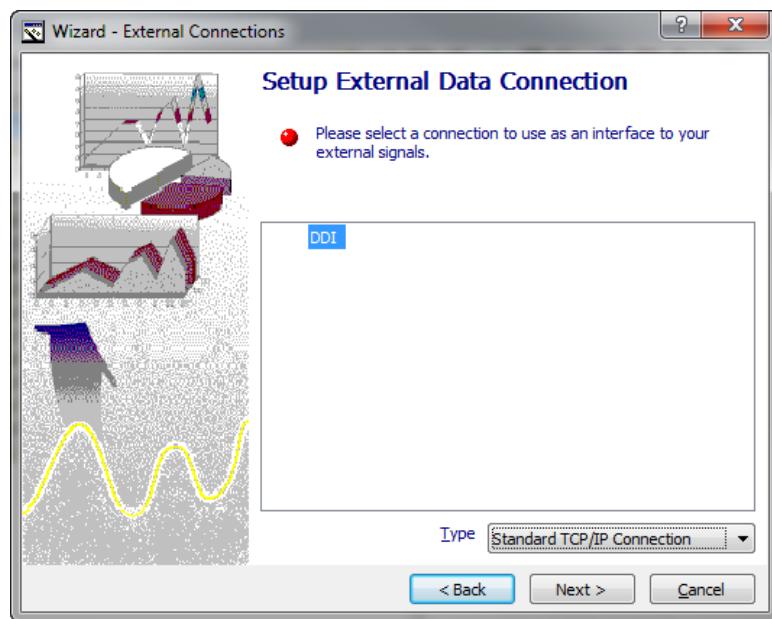


Figure 4-59 Configure the DDI as a Standard TCP/IP Connection

6. Specify the name or IP address of the machine providing DDI data. These should be the values shown in the *Luna MTS Data Server* configuration screen.

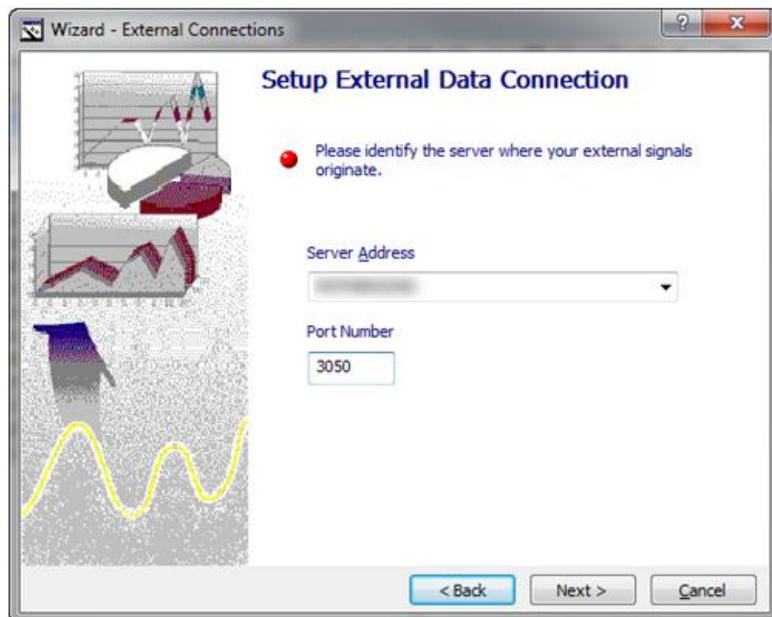


Figure 4-60 Specify the server's IP address and port number

7. Select the test station to receive data from. *Luna MTS Data Server* should be the only option.

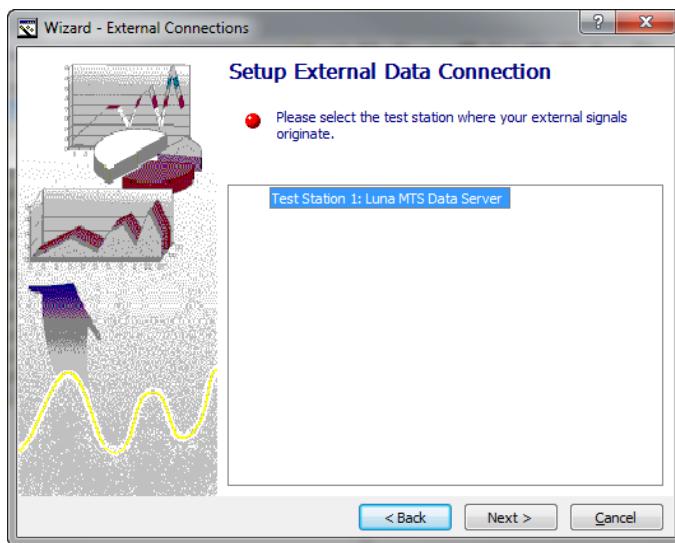


Figure 4-61 Select the Luna MTS Data Server from the list of connected test stations

8. Select the data stream. There should be only one.

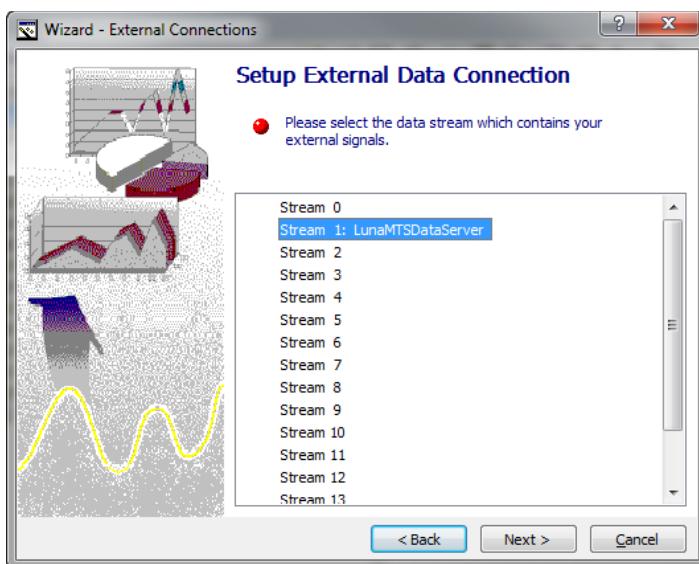


Figure 4-62 Select which data stream to monitor

9. Select the signals to include in the test. This list should contain all of the signals being exported from the ODiSI-B software.

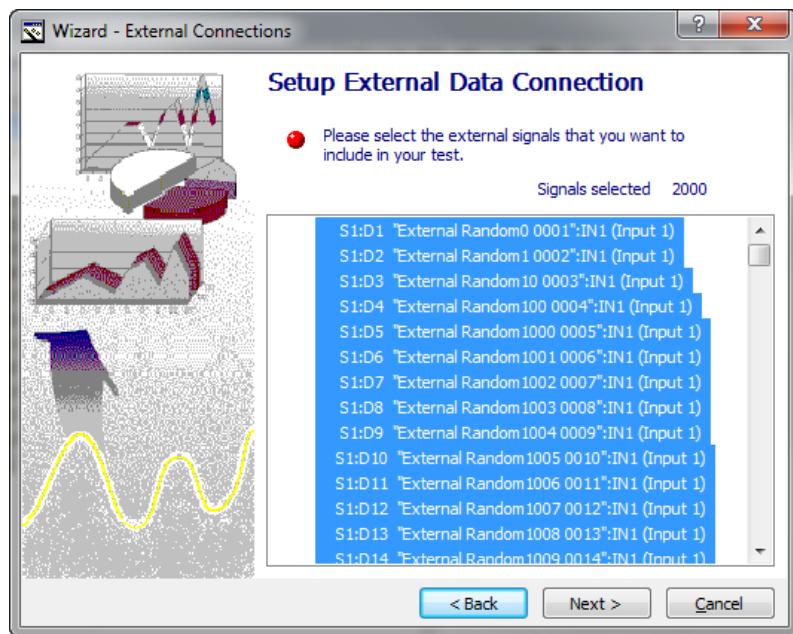


Figure 4-63 Select the signals to include

10. Click *Finish*.

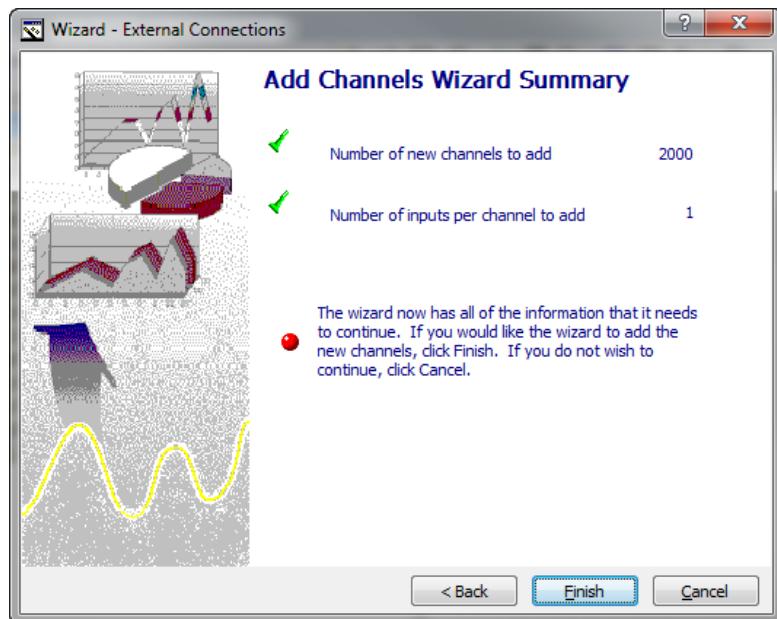


Figure 4-64 Confirm selection and finish

5 Software Development Kit

5.1 Overview

The ODiSI-B SDK is a collection of software routines that allows users to develop custom software utilities targeted for a specific measurement application. By utilizing the SDK, ODiSI-B users have the ability to write custom graphical user interfaces, implement custom data analysis schemes, or integrate the ODiSI-B into a larger data acquisition system.

The SDK includes sample programs in both C++ and LabVIEW. These example programs were written and tested using Visual Studio 2012 and LabVIEW 2016.

The example programs call functions contained in DLLs (primarily *ODB.DLL*) to control the ODiSI-B and to perform calculation. It may be possible to call these same DLL functions from other programming languages; however C++ and LabVIEW are the only languages currently supported by Luna.

5.2 Getting Started

5.2.1 Copying SDK Files

The SDK consists of a set of directories and files, which should be copied to a directory on your PC. For the sake of clarity, it is recommended that the software be copied to the folder *C:\ODiSI-B-B_SDK_v5.2.0*. An alternate path can be used if desired, but remainder of these instructions will assume that you have used the default name.

The SDK contains the following folders:

- *C++ Examples* – contains example .cpp files for a range of ODiSI-B tasks
- *include* – contains the ODBSDK.h file listing all of the available SDK functions
- *lib* – includes the ODB.lib and all necessary DLLs for linking against the ODiSI-B SDK. Also includes *ODiSI-B SDK.llb* which contains the LabVIEW SDK and examples.

5.2.2 Runtime Installer

If using the SDK on a PC that was not provided by Luna it will be necessary to install the ODiSI-B software prior to using the SDK. Once the software has been installed, ensure that the GPU and LunaPCIe drivers are installed and up to date.

- GPU drivers can be found and updated through Nvidia's website:
<http://www.nvidia.com/Download/index.aspx>
- The LunaPCIe driver is located at:
C:\Program Files (x86)\Luna Technologies\ODiSI-B vx.x.x
Refer to Appendix E for instructions on installing this driver.

The easiest way to begin using the SDK is to view and run the provided example programs. These programs demonstrate the various operations that can be performed with the SDK, and can serve as starting templates for writing your own programs.

5.3 LabVIEW Support

The LabVIEW library C:\ODiSI-B_SDK_v5.2.0\lib\ODiSI-B SDK.llb contains low level VIs that call the DLLs in the C:\ODiSI-B_SDK_v5.2.0\lib folder. These VIs serve as building blocks for writing more complex programs.

The LabVIEW LLB also contains example programs that use the low level VIs to control the ODiSI-B and compute measurements. These programs demonstrate how to load configurations, connect to the system, retrieve and set information, acquire measurements, log and load measurements, as well as process and plot data.

This guide will now explain the function of each of the example programs.

5.3.1 Basic Scan

BasicScan.vi demonstrates the most basic functions that are required to connect to the device, perform a single measurement, and display the strain data from the measurement. It also demonstrates the proper error retrieval and program termination procedures.

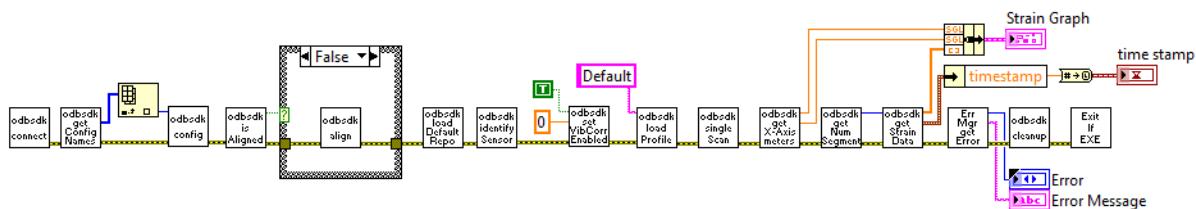


Figure 5-1 Basic Scan

5.3.2 Basic Scan 250 Hz

BasicScan250.vi expands the Basic Scan example to include the selection of a different scan mode (configuration). To select a scan mode, the *odb sdk_getConfigNames.vi* is called to retrieve the names and corresponding configuration index of each the available modes. In this example, the 1D array of names is searched against the string that matches the desired configuration mode. In this case we are selecting the 250 Hz mode. (It may be necessary to modify this example to use a different mode if 250 Hz mode is not enabled in the ODiSI-B being used.)

The 1D array search returns the index that matches the desired mode, which is then used to index the configuration index array. This yields the configuration index of the mode we want to load, which is passed as an input to *odb sdk_configure.vi*. The remainder of the example is identical to the Basic Scan example.

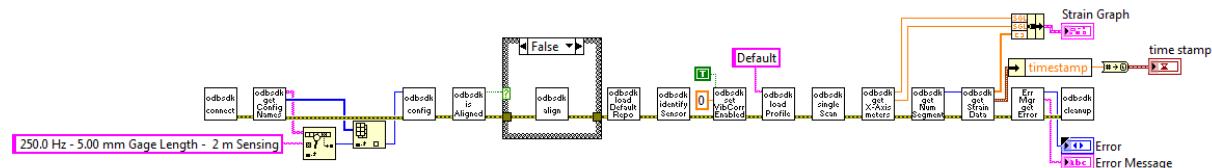


Figure 5-2 Basic Scan 250

5.3.3 Basic Scan Plus Info

BasicScanPlusInfo.vi expands upon the Basic Scan example by adding indicators to display additional information. The *odb sdk_get...* and *odb sdk_list...* functions can be called at any time without interrupting the ODiSI-B measurement process.

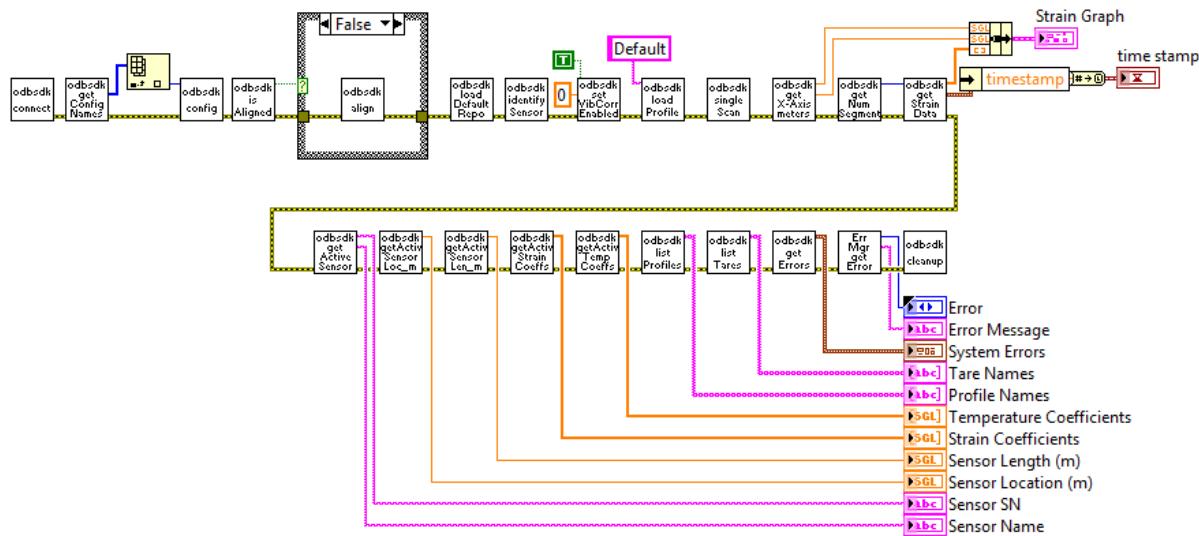


Figure 5-3 Basic Scan Plus Info

5.3.4 Continuous Scan

ContinuousScan.vi expands upon the Basic Scan example by replacing the single scan with continuous acquisition and processing.

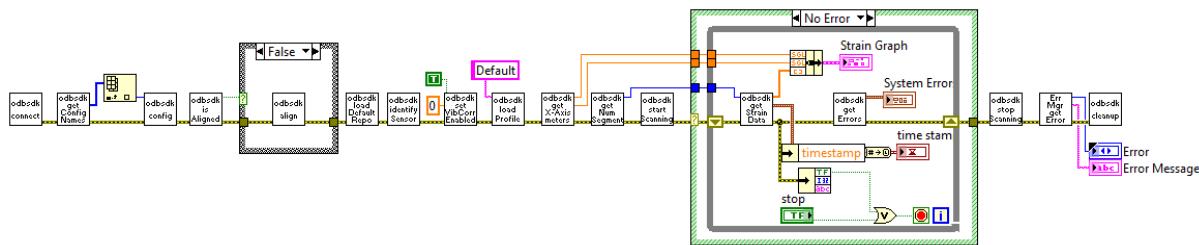


Figure 5-4 Continuous Scan

Once the ODiSI-B is scanning, it is possible to loop the data processing steps to create a continuous acquisition program. The loop configuration shown in the example will display data to the strain graph as fast as the computer can convert the raw data into engineering units. This speed is typically a fraction of the acquisition rate and increases with shorter sensor lengths. When finished, it is important to use the *odb sdk_stopScanning* function prior to the *odb sdk_cleanup* function. If the order is reversed or one of the two functions is omitted, the software can freeze and communication with the ODiSI-B will cease until the system has been restarted.

5.3.5 Check Laser Status

CheckLaserStatus.vi expands on the Continuous Scan example to check the status of the laser after every scan. This is a new feature in ODiSI-B v5.0.0. *odbsdk_getLaserStatus.vi* returns one of four values: *normal*, *okay*, *warning*, or *error*, based on the laser's operating temperature.

CheckLaserStatus.vi retrieves the laser status and displays it to the screen.

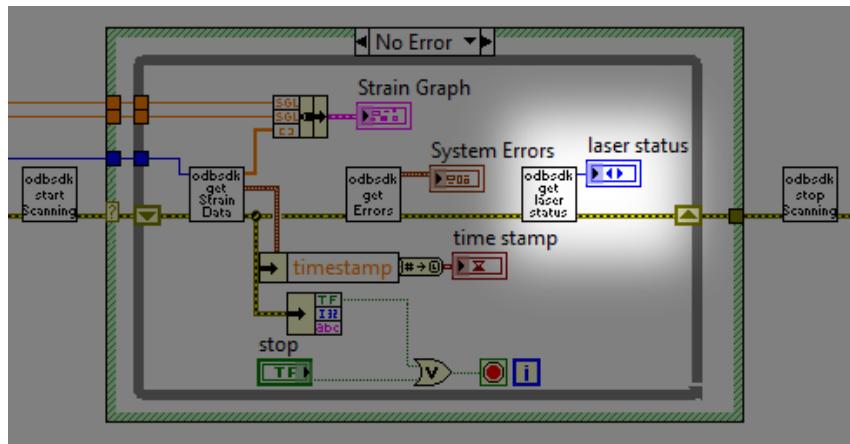


Figure 5-5 Check Laser Status

In a real application, the software should stop operating if the laser status is reported as *error*, and provide a warning message to the user if the status is *warning*.

In the ODiSI-B control software, a warning message is shown once if the laser reaches the *warning* status. The warning is not repeated if the status dithers between *warning* and *okay*, but a warning will be displayed again if the laser returns to *normal* status and then *warning* again.

5.3.6 Continuous Scan High-Speed CFG Sensor

ContinuousScanHighSpeedCGFSensor.vi demonstrates how to use the SDK with High-Speed CFG sensors. High-Speed CFG sensors are a new feature in ODiSI-B v5.0.0.

The main difference from an SDK standpoint is that High-Speed CFG sensors cannot be automatically identified by calling *odbsdk_identifySensor.vi*. Instead, a High-Speed CFG sensor must be manually selected using *odbsdk_setSensor.vi*.

High-Speed CFG sensors only operate in 23.8 Hz and 100 Hz acquisition modes. This example begins by choosing 100 Hz mode, rather than accepting the first available mode as many of the other examples do.

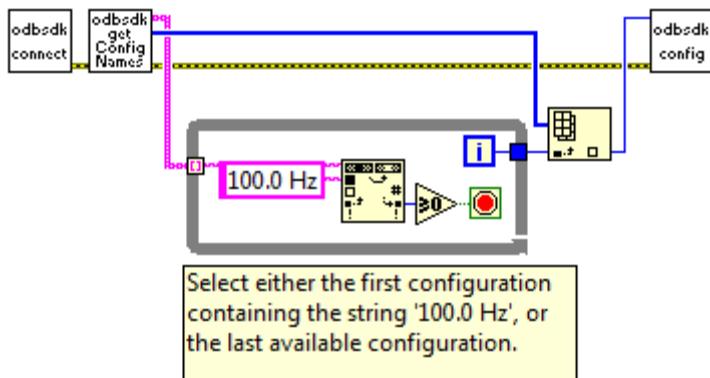


Figure 5-6 Select a scan mode

Instead of calling *odb sdk_identifySensor.vi* to identify a sensor, this example retrieves a list of all of the sensors and calls *odb sdk_getSensorType.vi* to determine whether they are High Definition or High-Speed CFG sensors. It then chooses the first High-Speed CFG sensor from the list and calls *odb sdk_setSensor.vi* to select it.

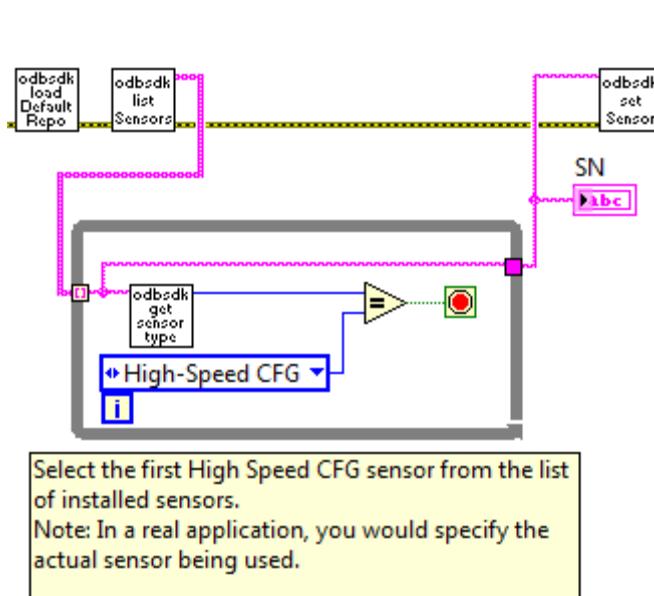


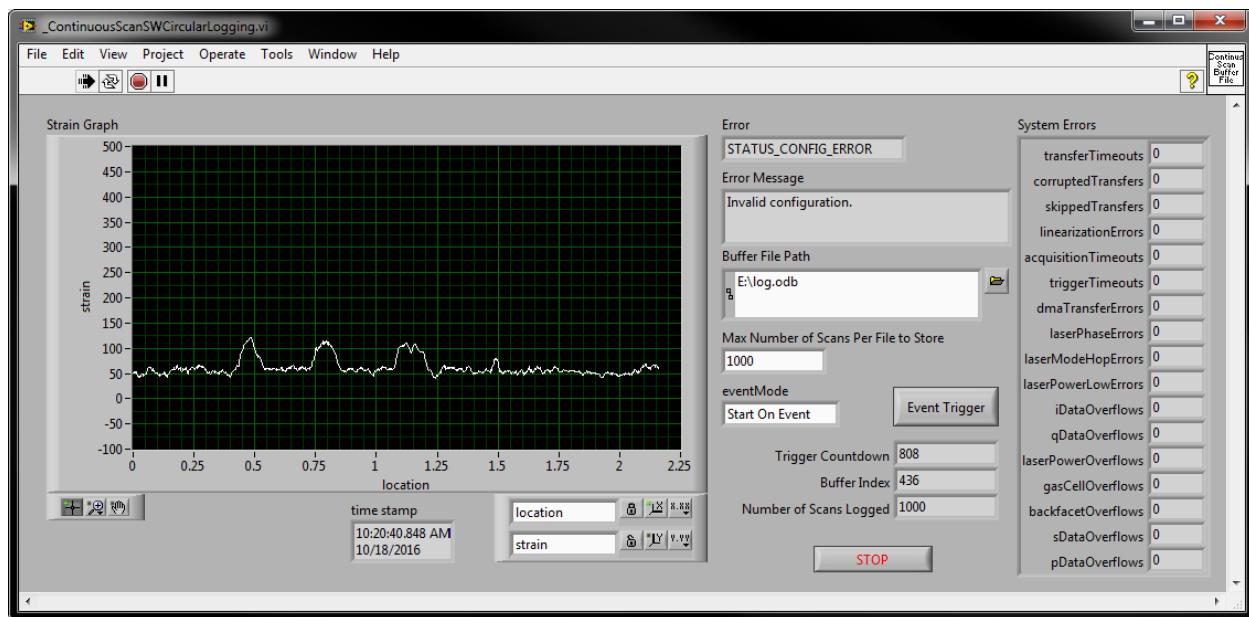
Figure 5-7 Use the Set Sensor function to choose a High-Speed CFG sensor

Note that while this works as an example, a real application would need to select the actual sensor that is connected to the instrument.

The rest of the processing is identical to the Continuous Scan example.

5.3.7 Continuous Scan SW Circular Logging

ContinuousScanSWCircularLogging.vi expands upon the Continuous Scan example by logging raw data to a circular log file.



On the front panel, the *Number of Scans to Store* control sets the size of the circular raw data file that the program will create. Circular raw data files contain a fixed number of entries, with the oldest scan data being constantly replaced by the latest scan. (This is the same way that the Circular Raw Data File operates in the main ODIsI-B control software.)

Number of Scans Logged and *Buffer Index* show the number of scans currently written to the file and the current file position that is being written. When the program starts, *Number of Scans Logged* will increase from 0 to *Number of Scans to Store* as the file is initially filled. *Buffer Index* will cycle over the number of scans, continuously replacing the oldest acquisition data with the most recent data.

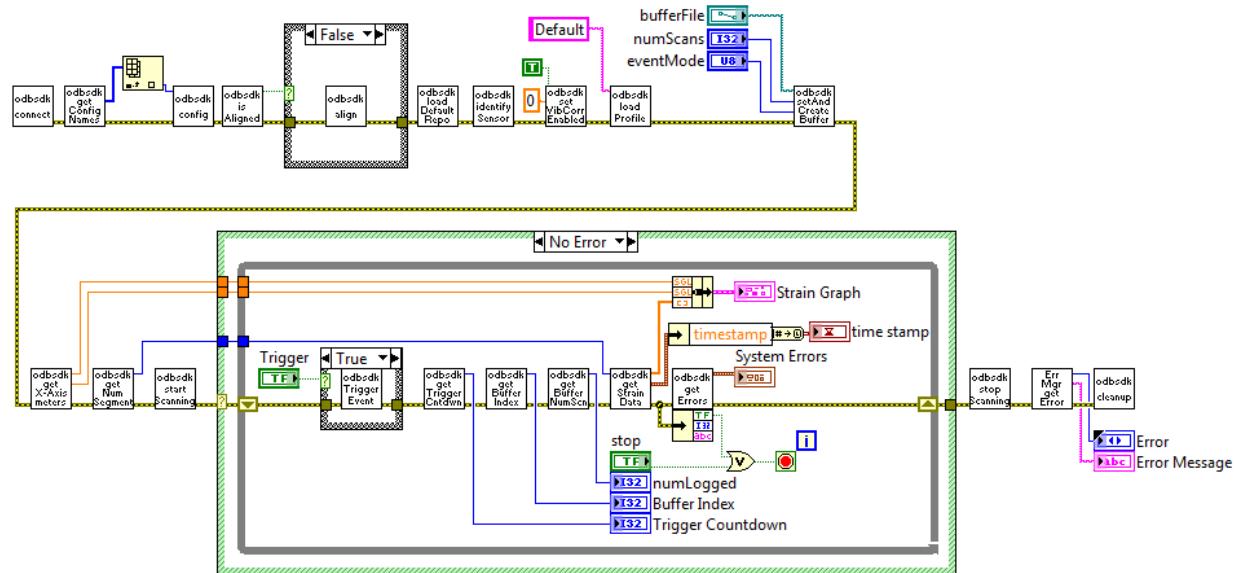


Figure 5-8 Continuous Scan SW Circular Logging

`odbdk_setAndCreateBufferFile.vi` creates a new raw data file that will be filled with data whenever the ODIsI-B device is placed in continuous acquisition mode. The *numScans*

input controls the number of scans, and thus the window of time, that the circular log file will be able to store. Note that the circular log file must always be placed on a solid state drive (SSD).

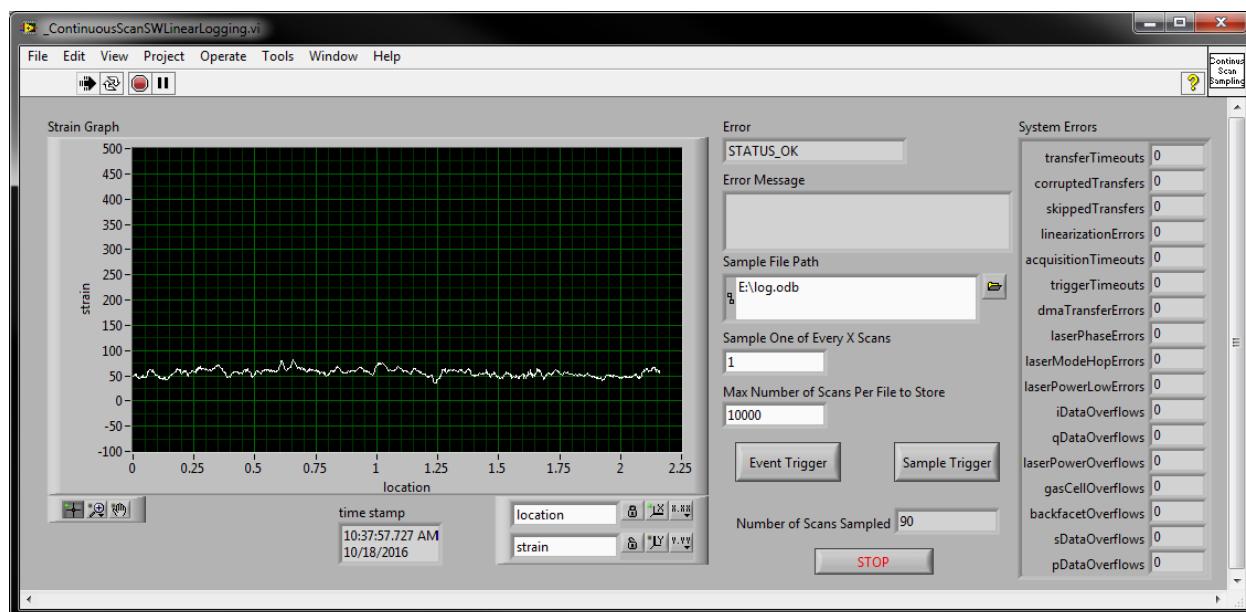
odbsdk_triggerEvent.vi sends the event trigger, causing the log file to stop adding new entries according to the method chosen (start, stop, or center).

odbsdk_getTriggerCountdown.vi returns the number of scans remaining to be written to the output file. This is a useful function, as a return value of 0 indicates that the circular raw data file has been completely written. A value of -1 indicates that the trigger has not yet been received.

odbsdk_getBufferIndex.vi and *odbsdk_getBufferNumScans.vi* return the current buffer index and number of scans currently written to the data file. These functions are unrelated to the triggering and can be used to determine the file position and size of a circular raw data file.

5.3.8 Continuous Scan SW Linear Logging

The Continuous Scan SW Triggered example creates a raw data file like the previous example, but provides a way to stop writing to the data file at a desired moment.



The *Event Trigger* button tells the software to start logging to the raw data file.

The *Sample Trigger* button tells the software to stop writing to the raw data file.

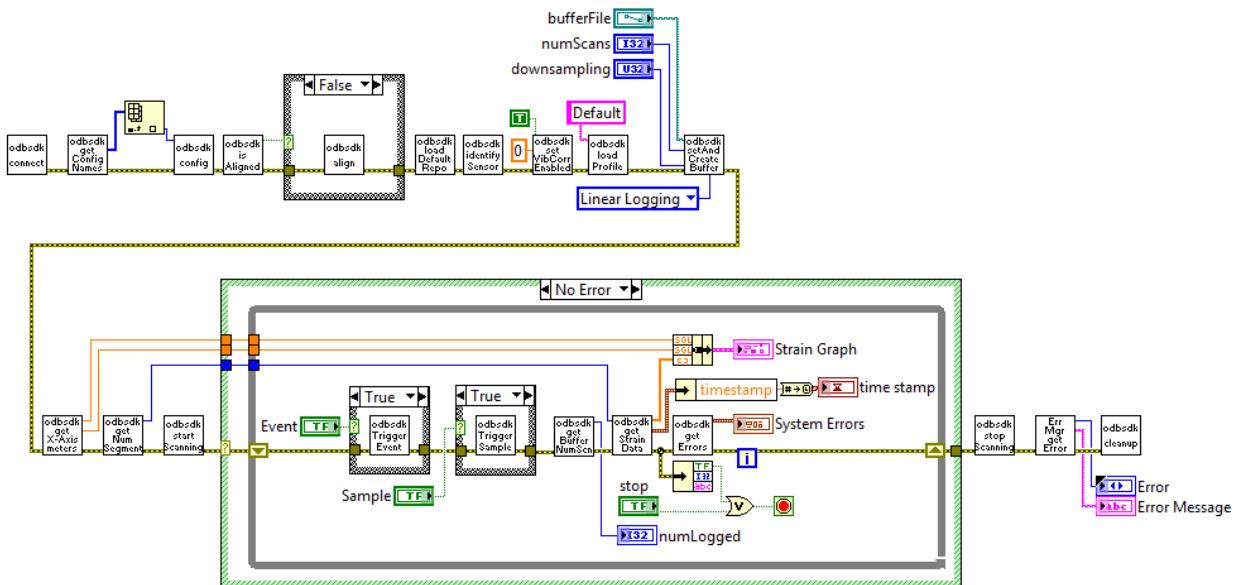


Figure 5-9 Continuous Scan SW Linear Logging

As in the previous example, the circular raw data file is created by calling `odbSdk_setAndCreateBufferFile`. This time, the file log type is set to 1 (linear logging).

`odbSdk_triggerEvent.vi` sends the event trigger, causing the log file to start adding new entries.

`odbSdk_triggerSample.vi` sends the sample trigger, causing the log file to stop adding new entries.

5.3.9 Continuous Scan Tare

Continuous Scan Tare expands upon the Continuous Scan example by allowing the user to tare the processed data.

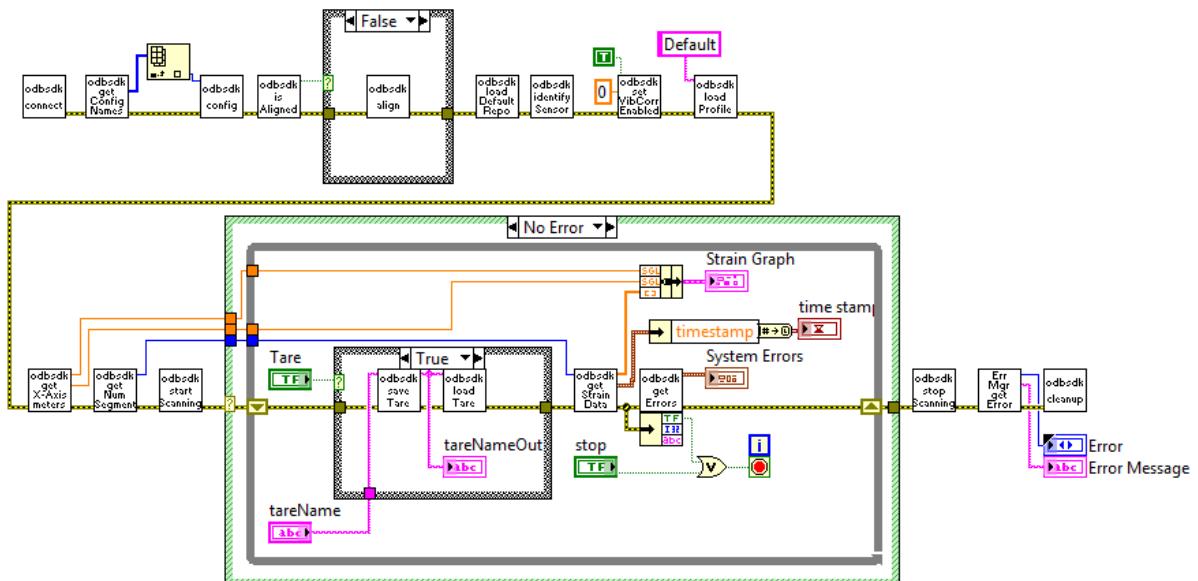


Figure 5-10 Continuous Scan Tare

odbsdk_saveTare.vi saves the current data as a tare. The tare is saved in the raw data format so that you can always load the same tares regardless of how the processing region or vibration correction region have been set. Tare files are placed in the sensor repository in the folder of the active sensor. Note that *odbsdk_saveTare.vi* provides an output for the full tare name, which appends a timestamp to the input tare name. When loading tares, you must use the full tare name output by *odbsdk_saveTare.vi* or *odbsdk_listTares.vi*. *odbsdk_loadTare.vi* will load a tare file to zero all future measurement processing.

5.3.10 Post Processing

PostProcessing.vi demonstrates the basic procedure for post processing a log file acquired by the ODiSI-B instrument.

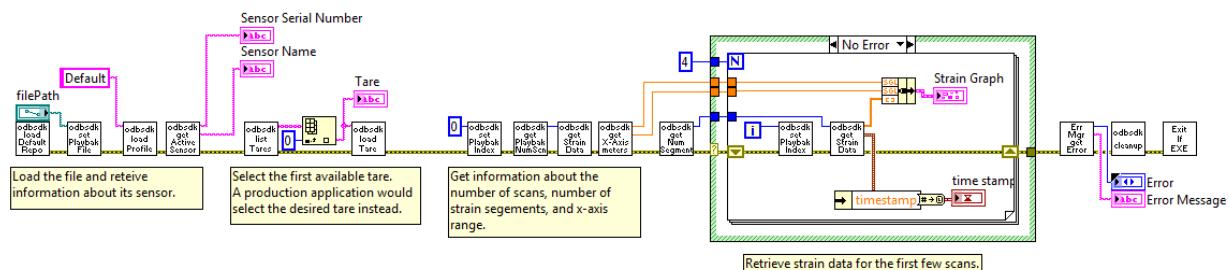


Figure 5-11 Post Processing

When post processing, there is no need to connect to the ODiSI-B instrument. You must still, however, load the sensor repository. *odb sdk_setPlaybackFile.vi* will load a log file for post processing playback. When the playback log file is loaded, the software checks the sensor SN stored in the log file against the sensors in the loaded sensor repository. If you have not installed the sensor in the current sensor repository, an error will be returned.

Once a log file has been loaded for playback, you may use *odbsdk_getPlaybackNumScans.vi* to return the number of scans available in the log file. *odbsdk_setPlaybackIndex.vi* will load a measurement from the given index within the file. Calling *odbsdk_setPlaybackIndex.vi* is similar to calling *odbsdk_singleScan.vi*; the specified measurement will be loaded into memory and processed so that processing results may be returned with *odbsdk_getStrainData.vi*, etc.

5.3.11 Point vs Time

The Point v. Time example demonstrates a method of displaying the strain of a single gage as a function of time. This type of trace closely resembles what is generated by strain gages and can be useful for comparison to other strain measuring systems.

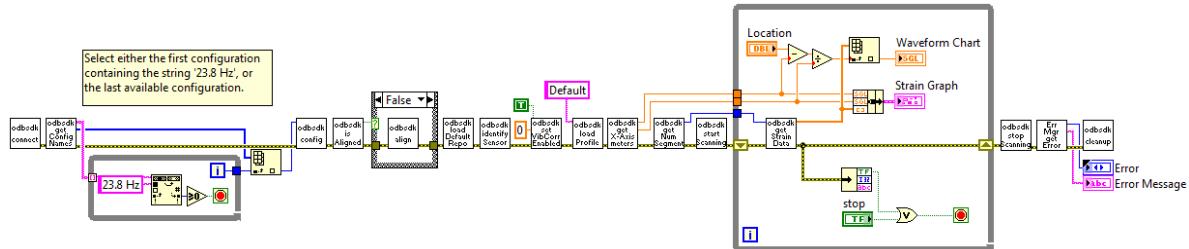


Figure 5-12 Point vs Time

5.4 C/C++ Support

There is a C/C++ header file included with the SDK. This file is all that is required to begin compiling and working with the library. Below is the name of the file along with a brief description.

ODBSDK.h –contains all of the prototypes of the functions that are specific to the ODiSI-B product.

C/C++ examples similar to the LabVIEW examples detailed above are also included with the SDK.

5.4.1 Function Descriptions

odbsdk_align

Perform an alignment of the ODiSI-B internal optics. Nothing should be attached to the standoff during this operation.

odbsdk_cleanup

Prepares the ODiSI-B DLLs for correct termination. This should be the last SDK function called.

odbsdk_configure

Choose and load the scan mode of the ODiSI-B. This may take a while to complete.

odbsdk_connect

Establish PCIe communications with the ODiSI-B. This should be the first function called when attempting to acquire new data.

odbsdk_convertUnits

Convert between various length units supported by the ODiSI-B software system. (meters, millimeters, feet, inches, nanoseconds (time of flight), and segment index)

odbsdk_editCoefficients

Sets strain and temperature coefficients for a chosen sensor. Both sets of coefficients are 0-4th order.

odbsdk_editSensorProfile

Change the start and end locations of the selected sensor profile.

odbsdk_getActiveSensor

Retrieve the SN and name of the sensor currently in use.

odbsdk_getActiveSensorLength_(x)

Return the length of the active sensor in one of the ODiSI-B length units.

odbsdk_getActiveSensorLoc_(x)

Returns the location of the active sensor in one of the standard ODiSI-B units.

odbsdk_getActiveSensorStrainCoeffs

Return the 0-4th order strain coefficients for the sensor currently in use.

odbsdk_getActiveSensorTemperatureCoeffs

Return the 0-4th order temperature coefficients for the sensor currently in use.

odbsdk_getBufferIndex

Return the index of the circular raw data buffer currently being written to.

odbsdk_getBufferNumScans

Return the number of scans being stored in the circular raw data buffer (number of indices).

odbsdk_getConfigNames

Return the names of the configurations available on the ODiSI-B.

odbsdk_getDelayData

Return the delay from the distal standoff connector to the start of the sensing region.

odbsdk_getError

Return a description of the last reported error generated within the SDK functions.

odbsdk_getErrors

Return descriptions of all errors generated within the SDK functions.

odbsdk_getNumSegments

Return the number of segments (gages) being processed for the active fiber sensor.

odbsdk_getLaserStatus

Retrieves the status of the laser. One of four enumerated values, LASER_TEMPERATURE_NORMAL, LASER_TEMPERATURE_OKAY, LASER_TEMPERATURE_WARNING, or LASER_TEMPERATURE_ERROR is returned.

odbsdk_getPlaybackNumScans

Return the number of scans in a raw data file.

odbsdk_getQualityData

Return a 1D array of Quality Factor data calculated from the last processed scan (live data) or the current scan index (raw data).

odbsdk_getStrainData

Return a 1D array of strain (microstrain) data calculated from the last processed scan (live data) or the current scan index (raw data).

odbsdk_getTemperatureData

Return a 1D array of temperature (degrees Celsius) data calculated from the last processed scan (live data) or the current scan index (raw data).

odbsdk_getTriggerCountdown

Return the number of scans left until the current event trigger action is completed.

odbsdk_getXAxis_(x)

Return the x axis length and gage length in one of the ODiSI-B standard units. (meters, millimeters, feet, inches, and segment index)

odbsdk_getXAxisSegment

Use to limit the strain or temperature data retrieved to a smaller (continuous) region of the fiber sensor.

odbsdk_identifySensor

Run the identification routine that scans the connected sensor and compares it to each of the installed sensors and stops when a match has been found or if none of the installed sensors match. This function cannot identify High-Speed CFG sensors. Instead, odbsdk_setSensor must be called to select these sensors.

odbsdk_installSensorFromZip or **odbsdk_installSensor**

Install a sensor to the ODiSI-B sensor repository from the zip file supplied with the sensor or generated by the Sensor Generation software.

odbsdk_isAligned

Check the alignment of the internal optics of the ODiSI-B. It is best to disconnect everything from the distal end of the standoff when performing this check.

odbsdk_listProfiles

Return a 1D array containing the names of the profiles associated with the active fiber sensor. Profiles are useable between all scan modes.

odbsdk_listSensors

Return a 1D array containing the sensors installed in the currently active sensor repository.

odbsdk_listTares

Return a 1D array containing the names of the tares associated with the active fiber sensor.

odbsdk_loadDefaultSensorRepo

Set the active sensor repository to the default location

C:\ProgramData\Luna Technologies\ODiSI-B\Sensors

odbsdk_loadProfile

Activate one of the profiles associated with the active fiber sensor. Call odbsdk_listProfiles to see all available or call odbsdk_editProfile to change an existing profile or odbsdk_createProfile to generate a new profile. Profiles are useable between all scan modes.

odbsdk_loadSensorRepo

Attempt to load a sensor repository from a directory other than the default location.

odbsdk_loadTare

Activate one of the tares associated with the active fiber sensor. Call odbsdk_listTares to see all available or odbsdk_saveTare to create a new tare.

odbsdk_newSensorProfile

Create a new sensor profile and store it in the currently loaded sensor repository.

odbsdk_pollTriggers

Check to see if a 5v signal has been received by the sample trigger or event trigger port on the ODiSI-B since the last time the poll triggers function ran.

odbsdk_resetTriggerPoll

Reset the poll used to check for event or sample triggers. This should be called after configuring the ODiSI-B if the software will be using odbsdk_polltriggers.

odbsdk_saveTare

Create a new tare for the currently active sensor. Note: tares can be used between profiles and are independent of the measurement or length units currently being used. Tares are stored in the current sensor repository in a folder corresponding to the scan mode currently enabled.

odbsdk_getActiveSensorStrainCoeffs

Return the 0-4th order strain coefficients currently being used to compute strain from spectral shift data.

odbsdk_getActiveSensorTemperatureCoeffs

Return the 0-4th order temperature coefficients being used to compute strain from spectral shift data.

odbsdk_setAndCreateBufferFile

Specify the parameters of the raw data file and create it.

odbsdk_setDACCoefficients

Set the Voltage response for each of the 16 DAC channels. The units for this coefficient are volts/microstrain.

odbsdk_setDACLocations

Assign 10 fiber locations to the 16 channels of the DAC.

odbsdk_setDACValues

Choose both the fiber locations and coefficients of the 16 DAC channels.

odbsdk_setDACVoltageRange

Specify the minimum and maximum voltage range the connected DAC is capable of attaining. This is used to ensure the correctness of the voltage/microstrain coefficients.

odbsdk_setPlaybackFile

Choose the file to be used when post processing raw data files.

odbsdk_setPlaybackIndex

Set the location (scan number) in the raw data file to process. You must first have specified a file using odbsdk_setPlaybackFile.

odbsdk_setProcessedTextEnabled

Enable the software to save processed data to the text file chosen using odbsdk_setProcessedTextFile.

odbsdk_setProcessedTextFile

Choose the location of the text file where processed data will be stored if odbsdk_setProcessedTextEnabled is set to true.

odbsdk_setProcessingRegion_(x)

Choose the processing region of the fiber using one of the ODiSI-B standard length units. This provides an alternative to creating profiles.

odbsdk_setQualityFactorThreshold

Choose the Quality Factor threshold (values range from 0 to 1) that will be used to filter all data that is processed. The default value of 0.38 will be applied even if this function is not called.

odbsdk_setSensor

Set the active sensor using the sensor SN. This may be called rather than using odbsdk_identifySensor if you already know which sensor is connected. This function is the only way to use a High-Speed CFG sensor. Automatic identification using odbsdk_identifySensor is not possible.

odbsdk_setVibrationCorrEnabled

Enable or disable the vibration correction from being applied to your processed data. Also has input for specifying the start location of the region used to generate the correction.

odbsdk_singleScan

Instruct the laser to take a single scan.

odbsdk_startScanning

Instruct the laser to begin scanning at the rate determined by the scan mode chosen using odbsdk_configure.

odbsdk_stopScanning

Instruct the laser to stop scanning.

odbsdk_triggerEvent

Send an event trigger to the software. If raw data logging has been configured this will either start logging (linear file) or prompt the file to stop in the manner chosen (circular).

odbsdk_triggerSample

Send a sample trigger to the software. If logging raw data using a linear log file, this will cause the software to stop logging to the file.

odbsdk_uninstallSensor

Delete all files associated with an individual sensor from the currently active sensor repository.

odbsdk_viewCoefficients

View the 0-4th order coefficients currently being used to calculate strain or temperature.

LabVIEW is a registered trademark of National Instruments Inc.

Visual C++ is a registered trademark of Microsoft Corporation

6 ODiSI Measurement Streaming Protocol

ODiSI Measurement Streaming Protocol (OMSP) is a protocol used to stream measurement data from the ODiSI-B to a client. Users can write client code to connect to the ODiSI-B and receive measurement data for use in their own programs.

OMSP was designed to be easy to understand and simple to write client code for. The protocol consists of JSON-formatted data sent over a TCP/IP connection. JSON is a lightweight data interchange format that is easy to read and parse and is supported in virtually all programming languages.

Example C++ OMSP client code is provided, which can serve as a code base or an example for developing OMSP client code. In addition, a DLL is provided that can be called to create an OMSP client without writing any TCP or JSON code.

6.1 OMSP Structure

OMSP is a simple format consisting of two message types. Metadata messages provide information about the ODiSI-B system, the active sensor, and the names and locations of the gages that will be streamed. Measurement messages contain the gage values generated from a single measurement.

A new measurement message is sent each time the ODiSI-B computes a strain or temperature measurement. Metadata messages are sent less frequently -- once for every 100 measurement messages while the ODiSI-B is processing, or once per second while the ODiSI-B is not processing.

6.1.1 Streaming connection

The ODiSI-B streams data over a TCP/IP connection. The TCP/IP port can be configured; the default port is 50000.

Upon configuration by the user, the ODiSI-B software opens a port and begins listening for a client connection. The client program will initiate a TCP connection to the OMSP server on the specified port number. As soon as a connection is established, the ODiSI will begin sending messages to the client.

If the ODiSI loses the connection with the client, it will resume listening for a new client message and will resume sending data once a new connection has been established. The ODiSI will not send measurement data that was missed during the time the connection was down.

The application data is one way. That is, the ODiSI sends data to the client but does not look for any responses. Future revisions will include a separate control channel which will be used to start and stop measurement streaming.

Each new message starts at the beginning of a new TCP packet.

6.1.2 Message Types

Two types of messages are transmitted over the streaming protocol: a Measurement Message and a Metadata Message.

The Measurement Message is a simple format, consisting mostly of an array of numeric data which represents the measured strain and temperature gage values. The metadata

needed to fully interpret the data (gage and sensor names, gage locations, units, etc) require too much space to be transmitted with every measurement and are instead transmitted in a less frequent Metadata Message.

Both the Measurement Message and the Metadata Message use JSON formatting. This data is human readable text, encoded as UTF-8. The language of the JSON fields is always English, and is not affected by the language selection made in the main software. However, all fields that hold user defined values (test name, sensor name, gage name) support the full UTF-8 character set.

The ODiSI will transmit a Measurement Message for every acquired measurement. A Metadata Message will be transferred once per second or once per 100 measurements, whichever is the longer interval.

When the ODiSI is stopped and not acquiring data, the Metadata Message will be sent once per second to keep the connection alive. Measurement Messages will be sent when ODiSI operation is started again.

6.2 Metadata Messages

6.2.1 Contents

The Metadata Message is formatted as human readable UTF-8 text, using JSON formatting. A CRC-16-ANSI (0x8005) checksum (encoded as 4 hexadecimal characters) follows the JSON text. A NULL character follows the checksum and marks the end of the Metadata Message.

The metadata message contains the following information, formatted as JSON name/value pairs.

Description	JSON name	Value
Message type	"message type"	String "metadata"
Message format version	"message version"	Integer. 1 is the initial version.
Product	"product"	String Ex: "ODiSI-B"
System serial number	"system serial number"	String
Software version	"software version"	String. Ex: "5.2.0"
Hardware version	"hardware version"	String. Ex: "1.0.0"
Test name	"test name"	String. User defined in ODiSI software.
System status	"system status"	String. "stopped" or "measuring".
An array of sensors	"sensors"	Array of JSON elements contained in [] brackets

Each sensor contains the following information, formatted as JSON name/value pairs. In a multi-channel system, any unused channels are omitted from the metadata message. The ODiSI-B will only specify a single channel.

Description	JSON name	Value
Channel	“channel”	Integer. Always 1 for ODiSI-B.
Sensor name	“sensor name”	String. User defined in ODiSI software.
Sensor serial number	“sensor serial number”	String
Sensor part number	“sensor part number”	String
Sensor length	“length (m)”	Decimal number indicating length in meters
Sensor type	“sensor type”	String. “HD strain”, “HD temperature”, or “HS CFG strain”.
Units	“units”	String. “microstrain” or “deg C”
An array of gages	“gages”	Array of JSON elements contained in [] brackets

Each gage contains the following information, formatted as JSON name/value pairs.

Description	JSON name	Value
Gage name	“gage name”	String. User defined in ODiSI software.
Gage location	“location (mm)”	Decimal number indicating the gage location in millimeters

This provides all of the information needed to parse the measurement data transmissions into specific gage values.

The “system status” field indicates whether the system is currently acquiring and producing data or is stopped. A value of “measuring” indicates that the system is producing data. A value of “stopped” indicates that the system is not producing data. When the system is stopped, the system will send Metadata Messages only. When the system is measuring, it will send both Metadata and Measurement Messages.

No changes will be made to the sensor and gage configurations when the system is processing. When the metadata messages indicate a system status of “measuring”, the client can be guaranteed that the sensor and gage information will not change from one metadata message to the next.

The client can also use the Metadata Message checksum to determine whether changes have been made to the configuration. If a Metadata Message contains the same information as the previously transmitted Metadata Message, its checksum will remain the same as well.

When the system is in a “stopped” state, the sensor and gage configuration may be changed by the ODiSI user. The metadata messages will reflect the current configuration, but the client should be aware that this information may change.

The JSON portion of the Metadata Message is followed by a 16 bit CRC-16-ANSI (0x8005) checksum, encoded as a 4-character hexadecimal number. The checksum is calculated over the entire length of the JSON text, including the starting and final braces.

The checksum is followed by a single NULL character to mark the end of the Metadata Measurement transmission.

6.2.2 Example

```
{  
    "message type": "metadata",  
    "message version": 1,  
    "product": "ODiSI-B",  
    "system serial number": "20170DB10032",  
    "sw version": "5.2.0",  
    "hw version": "1.0.0",  
    "test name": "My Test",  
    "system status": "measuring",  
    "sensors": [  
        { "channel": 1,  
            "sensor name": "left wing",  
            "sensor serial number": "FS02017LUNA001234",  
            "sensor part number": "HDS05LC220P",  
            "length (m)": 5.1,  
            "sensor type": "strain",  
            "units": "microstrain",  
            "gages": [  
                { "gage name": "Left front", "location (mm)": 1456 },  
                { "gage name": "Left mid", "location (mm)": 2494 },  
                { "gage name": "Left back", "location (mm)": 4611 }  
            ]  
        }  
    ]  
}
```

}8F0A

6.3 Measurement Message

6.3.1 Contents

The Measurement Message is formatted as human readable UTF-8 text, using JSON formatting. A CRC-16-ANSI checksum (encoded as 4 hexadecimal characters) follows the JSON text. A NULL character follows the checksum and marks the end of the Metadata Message.

The Measurement Message includes a product name and a system serial number. This allows the client program to receive data from multiple ODiSI units. Each ODiSI will transmit its own Metadata Message containing an identifying system serial number. The system serial number in the Measurement Message allows the client to determine which ODiSI unit the measurement data is from, and which Metadata Message to use to interpret the data.

The Measurement Message includes a sequence number, which is incremented by the ODiSI with each measurement transmission. This number allows the client to determine if any measurements have been missed.

The Measurement Message includes a timestamp, which is split into separate JSON fields for year, month, day, hours, minutes, seconds, and milliseconds. There is also a field to indicate the time zone of the timestamp. The ODiSI-B always uses UTC timestamps, regardless of the local time zone.

The Measurement Message includes a channel number, which is used to map the array of data values to the gages defined for a particular sensor in the Metadata Measurement. The Measurement Message does not contain gage names or units, so it is necessary to refer to the Metadata Message to fully interpret the gage values contained in the Measurement Message.

The “data” element of the Measurement Message contains an array of decimal gage values, formatted as human-readable ASCII text. The JSON value *null* will be used to indicate a NaN, or “not a number” value. (Under certain conditions, the ODiSI may not be able to calculate a strain or temperature value for a certain gage. In these events, the ODiSI reports NaN as the gage value.)

The ODiSI-B will transmit integer values only for strain measurements, or values with a single decimal place (i.e. a tenth of a degree C) for temperature measurements. However, the ODiSI Measurement Streaming Protocol itself places no limitations on the precision of the numbers in the data array.

The element “number of gages” indicates the length of the “data” array. This value should already be known and should match the size of the “gages” array for the specified channel in the Metadata Message. It is included as a sanity check.

The Measurement Message header contains the following information, formatted as JSON name/value pairs.

Description	JSON name	Value
Message type	“message type”	String “measurement”
Message format version	“message version”	Integer. 1 is the initial version.
Product	“product”	String Ex: “ODiSI-B”
System serial number	“system serial number”	String
Sequence number	“sequence number”	Integer, 0 - 4,294,967,295. (Unsigned 32-bit integer.) Sequence number will wrap back to 0 after

		4,294,967,295.
Timestamp	“year”	Integer, 4-digit year. Ex: 2017
	“month”	Integer, 1 – 12
	“day”	Integer, 1 – 31
	“hours”	Integer, 0 - 23 (24 hour notation)
	“minutes”	Integer, 0 - 59
	“seconds”	Integer, 0 - 59
	“milliseconds”	Integer, 0 - 9999
Time zone	“time zone”	String. Ex: “UTC-0”
Channel	“channel”	Integer. Should match one of the “channel” fields in the “sensors” array of the Metadata Message. Always 1 for ODiSI-B.
Number of gages	“number of gages”	Integer
Data	“data”	Array of decimal numbers. The JSON value <i>null</i> indicates NaN.

The JSON portion of the Measurement Message is followed by a 16 bit CRC-16-ANSI checksum, encoded as a 4-character hexadecimal number. The checksum is calculated over the entire length of the JSON text, including the starting and final braces.

The checksum is followed by a single NULL character to mark the end of the Measurement Message transmission.

6.3.2 Example

```
{
  "message type": "measurement",
  "message version": 1,
  "product": "ODiSI-B",
  "system serial number": "20170DB10032",
  "sequence number": 125,
  "year": 2017,
  "month": 1,
  "day": 13,
  "hours": 13,
  "minutes": 23,
  "seconds": 14,
  "milliseconds": 56,
  "time zone": "UTC-0",
  "channel": 1,
```

```
"number of gages": 4,  
  "data": [ 805, -754.5, null, 901 ]  
}AF37
```

6.4 OMSP Source Code Examples

The ODiSI-B includes two example OMSP clients, including full source code. Both of these contained in the SDK\OMSP folder.

The purpose of these clients is to allow customers to test out the ODiSI-B's OMSP streaming functionality, and to provide source code examples showing how streamed ODiSI-B measurements can be received and integrated into another program.

Neither example is intended to be a fully featured client for use in a production environment. Users should not try integrate the example OMSP client executable into their workflow. Instead, users should read OMSP data by writing their own client code based on the examples.

6.4.1 OMSP Minimal Client

The SDK\OMSP\OMSP Minimal Client folder contains a C++ project *ODiSI_Meas_Client* which demonstrates how to receive OMSP data on a simple level. It runs in a Windows Console and prints received values to the screen with no graphical display. It also does no error checking.

This simple example demonstrates the logic that should be used to write an OMSP client program, regardless of the language used. The basic steps are:

- Connect to a TCP server (the ODiSI-B software)
- Receive TCP data
- Parse the received data as json
- Use metadata messages to build a table of expected gages
- Read the gage values from measurement messages

6.4.2 OMSP Graphical Client with DLL

The folder SDK\OMSP\OMSP Client DLL contains a C++ project that creates a more fully featured OMSP client, including error checking and parsing off all OMSP fields.

This project creates a DLL which can be called from another program in order to integrate the OMSP client into other software. Alternately, the code from OMSPCClientDLL can be copied into another C++ project, or used as a model for a non-C++ program, to add an OMSP client to another program without using DLL calls.

The folder SDK\OMSP LabVIEW GUI uses the DLL created by OMSPCClientDLL to build a graphical client. Users must have National Instruments LabVIEW version 2012 or later to run the example code. An executable version of this OMSP Example Client is also installed with the ODiSI-B software. Its usage is documented in [Section 4.9.4 Example OMSP Client](#).

7 Best Practices

This chapter describes suggested best practices in handling and running the ODiSI-B hardware and software in order to obtain the most accurate measurements.

7.1 System Operation Guidelines

The ODiSI-B system is a sensitive instrument that provides precise strain or temperature change measurements at a high spatial resolution along the length of an optical fiber sensor. It is best suited for static or low loading rate tests.

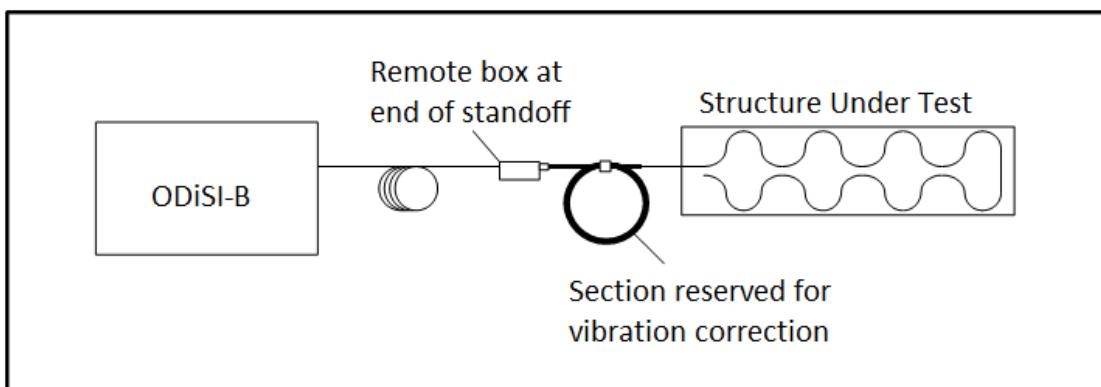
7.2 Vibration Tolerance – Test Setup

With the high sensitivity of the instrument, it is susceptible to environmental disturbances frequently associated with vibration or motion of the sensor. Since the ODiSI-B operates on signals reflected from the sensor, environmental disturbances can corrupt the data leading up to that point in the measurement. It is the user's responsibility to design tests with care to ensure that environmental disturbances do not couple into the sensing fiber.

Be wary of vibration from other machinery on the test floor, vibration coupling to the sensor from the load frame and shaking of unbounded sensor fiber. Care and effort should be taken to isolate the remote module from sources of vibration. Suggestions include using foam blocks to mount under the remote module or surround it.

7.3 Vibration Tolerance – Software Setup

The ODiSI-B software is able to compensate for environmental disturbances that have coupled into the 50 m standoff leading up to the sensing fiber. The first 20 cm of an ODiSI-B sensor is reserved for measuring vibrations that have coupled into the standoff. For optimal performance, the user should minimize strain and temperature gradients over this region. An effective method is to place the first 20 cm of the sensing fiber into a loop, and securing the loop with a piece of tape as shown in Figure 7-1. The loop will allow this portion of the sensor to come to a uniform strain state. Further, the loop also helps manage this additional length before the sensing region. To further reduce the impact of vibration, use the fastest sensing mode possible.



7-1 Diagram showing setup of vibration correction region of the sensing fiber.

7.4 Temperature Test Setup

The accuracy of temperature measurements can be improved by taking the following into consideration.

7.4.1 Thermal Contact and Thermal Gradients

Any heated or cooled surface will have a temperature gradient going away from it. As an example, for a hot plate set on a lab bench, a temperature gradient exists in the air above the hot plate. Consequently, for an accurate measurement of the plate temperature, the fiber sensor needs to be in contact with the plate. A fiber sensor that is in contact with the hot plate will register a higher measurement compared to a fiber sensor that is not in contact with the hot plate as the hot plate heats up, and a direct comparison will be invalid. Similar considerations should be given to measurement comparisons between a fiber sensor and other sensors e.g. a K-type thermocouple. A perfect correlation can be expected only when sensors are co-located within an environment in thermal equilibrium.

7.4.2 Sensor Response Time

The fiber has a very small thermal mass, especially so compared to a K-type thermocouple. The fiber will respond faster to temperature change than will the K-type thermocouple. A jacketed fiber however will report a slightly damped response (longer response time) due to the insulation effect of the jacket material as well as the air immediately around the fiber. Temperature measurement comparisons between different sensor types should consider sensor response times in addition to sensor co-location. The response time of a jacketed fiber can be increased by using thermal grease.

7.4.3 Strain Coupling

It is important that the fiber sensor is freely floating and not bound to any surfaces in order to report accurate temperature change measurements. Frictional strain can couple into the fiber temperature measurements due to a CTE mismatch between the fiber and the surface it is laying on, especially if the fiber is constrained from moving freely. As such it is advisable to lay out Teflon jacketed temperature sensors in as straight a configuration as possible as multiple tight bends can cause the fiber to bind to the Teflon jacket. Additionally, friction can be reduced by inserting graphite powder between the fiber sensor and its protective jacket.

7.4.4 Coating Effects

The polyimide coating absorbs moisture, which can change the sensor's thermal response. Consider sealing the sensor from air if humidity is likely to change during the test. In addition, you could also use desiccant within the test setup.

It has been observed that exercise cycling the sensor beyond the temperature range of interest is conducive to obtaining accurate measurements as this relaxes the polyimide coating and ensures repeatable performance over the temperature range of interest.

Contact Luna for discussions on custom sensors with the polyimide coating stripped off.

7.4.5 Temperature Coefficients

The temperature change curves are generated by converting the spectral shift curve from values in GHz to °C at known applied temperatures. This conversion is done using a linear fit. The default coefficients for this fit have been set for standard room temperature and zero strain. Temperature change measurement accuracies can be improved either by using 4th order polynomial coefficients or piece-wise linear coefficients. These methods are addressed extensively in Technical Note ENFY1403.

7.5 Strain Test Setup

Key to obtaining quality strain measurements is to ensure that the fiber sensor is properly attached to the test piece. The surface needs to be prepared (scuffed and cleaned with ethanol) and the fiber needs to be uniformly bonded down. Epoxy selection is also very important as it determines cure temperature, optimal temperature range for use, and the surface roughness of the test piece.

Each bonded region should be at least 7 gage lengths long, to ensure that the measurement region of interest is sufficiently far away from any edge effects (e.g. bonded-to-unbonded fiber transitions). For tests at elevated temperature that require thermal compensation of strain measurements, ensure that the compensation measurement device (unbonded fiber, thermocouple, or dummy gage) is in the same thermal environment as the strain sensor.

7.6 Strain Gradient Tolerance

Strain or temperature measurement outliers may result from excessive strain variation over the gage length. For High-Definition sensors the algorithm used to compute the spectral shift for each gage assumes a uniform strain across the gage. The presence of strain gradients will invalidate this assumption as strain gradients broaden the cross correlation peak, leading to a reduction in the signal-to-noise ratio. An estimate for the change allowable over a gage length is $517 \mu\epsilon$ ($\sim 50^\circ\text{C}$) for a 1.25 mm gage and $129 \mu\epsilon$ ($\sim 12^\circ\text{C}$) for a 5 mm gage.

The processing algorithms used for High-Speed CFG sensors do not make this assumption and are able to make measurements even in the presence of large strain or temperature gradients.

8 Troubleshooting

This chapter describes methods to troubleshoot ODiSI-B hardware and software errors.

If you have problems or error messages not listed here, or if the solutions provided here do not resolve a problem, please contact Luna Innovations toll free at (1-866-586-2682) or by e-mail at support@lunainc.com.

8.1 Dropouts

Environmental disturbances, such as mechanical vibration, present in the sensing region typically cause the displayed data at a location to vary randomly. Since the ODiSI-B spectral shift calculation is dependent on a correlation between a measurement and reference spectra over a gage length, if a strong correlation is not observed the second strongest peak can be any noise peak in the correlation background. This random nature of miscalculated points along the fiber causes them to be referred to as dropouts or outliers. A dropout is typically correlated with a low quality factor and can be filtered by the ODiSI-B software by setting the Quality Threshold appropriately.

If a high number of dropouts are observed along the length of the sensing fiber, the frequency of dropouts along the length of the sensor can help diagnose the problem with the test setup as detailed in the following chart:

Dropout Profile	Symptom	Resolution
Dropouts are observed at a high frequency along the entire length of the sensing fiber.	The standoff vibration correction cannot remove all vibration occurring in the standoff.	Investigate standoff length for excessive movement or contact with machinery that may be causing an oscillation of the standoff. Ensure that the vibration correction region of the sensing fiber is free of strain gradients by looping the fiber as shown in figure 9-1.
	The connector of the sensing fiber to the standoff or the connector of the standoff to the instrument has a poor optical connection.	Clean and reconnect optical connectors.
	The termination has been damaged causing a strong reflection to washout the sensing length.	Investigate termination integrity and use index matching gel to reduce reflection at the interface.

Dropouts increase in frequency along the length of the sensing fiber in a linear manner.	The sensing fiber is experiencing a common motion throughout the sensing length. The motion effect is constructively interfering along the sensing length.	Reduce test parameters of the loading structure (ie, load amplitude and frequency). Ensure test artifact is isolated from environmental disturbances.
Dropouts vary in frequency randomly along the sensing fiber.	The sensing fiber is experiencing uniform motion throughout the sensing length.	Ensure test artifact is isolated from environmental disturbances. Ensure unbonded fiber is secured and not experiencing motion.
No valid data observed after a location along the sensing fiber.	The optical fiber has broken at that location.	Crush the broken sensor to try and recover signal quality. Apply index matching gel to the crushed tip if possible. If signal quality cannot be improved, sensor must be replaced.
Dropout frequency is limited to areas of rapid strain transitions.	The strain change across a gage length is too great for the ODiSI-B calculation.	Attempt running the ODiSI in high resolution mode to decrease the gage length, effectively reducing the amount of strain variation across a given gage length.

8.2 Strain Relief

Similar to the vibration correction region, the sensor identification algorithm uses the signal of the first 20 cm of the attached sensor (the strain relief section) to find the fiber key, or reference of the fiber in a known strain state, in an archive of installed sensor files. Strain gradients within the first 20 cm can hinder the ability of the system to correctly identify an attached and installed sensor. The strain in this region should ideally be only variation in room temperature. Placing the first 20 centimeters in a loop as detailed in figure 9-1 assists in the fiber keying algorithm.

Dropout Profile	Symptom	Resolution
Vibration Correction error indicator toggles red.	The strain and temperature gradient in the region highlighted as the Vibration Correction region is too high.	View the strain or temperature along the sensor without selecting a Tare. Move the Vibration Correction region along the sensor length to an area with a flat strain profile.
Unable to Identify sensor	The strain and temperature gradient in the region highlighted as the Vibration Correction region is too high.	Relieve strain and temperature gradients within the first 20 cm of the sensor. Confirm strain relief by looking at the Measurement vs Length plot without selecting a Tare.

8.3 General Troubleshooting

8.3.1 The ODiSI-B or PC Will Not Power On

Cause(s)	Solution(s)
The ODiSI-B and/or the PC is not plugged in or turned on (and the laptop battery is dead, if applicable).	The ODiSI-B and the PC require separate power cords. Plug in the ODiSI-B and/or the PC, or replace dead batteries with charged ones if using a laptop PC. Instructions for the PC battery are located in the provided PC manual. Make sure to turn on the ODiSI-B power switch on the front of the instrument, and the PC according to its instructions.
Blown fuse.	Replace the fuse. For instructions, see "Replacing Fuses" in Chapter 7

8.3.2 Excessive Noise in Data

Cause(s)	Solution(s)
Mechanical vibration.	Move or shut down any sources of vibration. Isolate the ODiSI-B from those sources.
Loose fiber connection.	Tighten or reconnect the fiber connections.
Dirty fiber or bulkhead connections.	Clean all connections. For instructions, see "Cleaning Connectors" in Chapter 7

ODiSI-B not aligned.	Align the Optics using Tools\Align in the ODiSI-B Software. Affixing the remote module to the test fixture or test artifact will improve alignment stability.
There is a strong reflection in the optical path.	Check for any breaks in the fiber sensor.
Patch cable too long	Shorten the leads to comply with the length limit of the ODiSI-B.
Cracks in bulkhead connector alignment sleeve.	Contact Luna Technologies at 1-866-LUNAOVA or support@lunainc.com for service.

8.3.3 No Data After Scan

Cause(s)	Solution(s)
Incorrect graph range settings.	Rescale the plot axes.
Display Processed Data is disabled.	Make sure the Display Processed Data box is checked.

8.3.4 Instrument Drifts Out of Alignment

Cause(s)	Solution(s)
Temperature changes will cause the instrument to drift.	Tools\Align Affixing the remote module to the test fixture or test artifact will improve alignment stability.

8.3.5 Errors on Software Startup

Cause(s)	Solution(s)
The PCIe cable connections might be loose.	Check the connections to ensure that the cable is connected properly to both the instrument and the PC.
The instrument is not turned on.	Ensure that the instrument is plugged into the proper type of outlet and turn on the power switch.
The software is in an unknown state.	Exit and restart the software.

The hardware is in an unusable state and is not responding to the software.	Exit the software. Turn the instrument off and then back on again. Restart the PC and the control software.
The PCIe cable is faulty.	Replace the cable
Communication Error.	Turn computer off. Turn on ODiSI-B. Restart Computer.

8.3.6 The Optics Were Detected to Not Be Aligned Properly

Cause(s)	Solution(s)
This is just a warning letting the user know that the optics are not aligned properly.	Align the optics with Tools\Align. Affixing the remote module to the test fixture or test artifact will improve alignment stability.

8.4 Technical Support

If you have any problems with or questions about the information contained in this document, please contact our technical support staff by one of the following methods:

Luna Technologies, A Division of Luna Innovations Incorporated

3155 State Street

Blacksburg, VA 24060

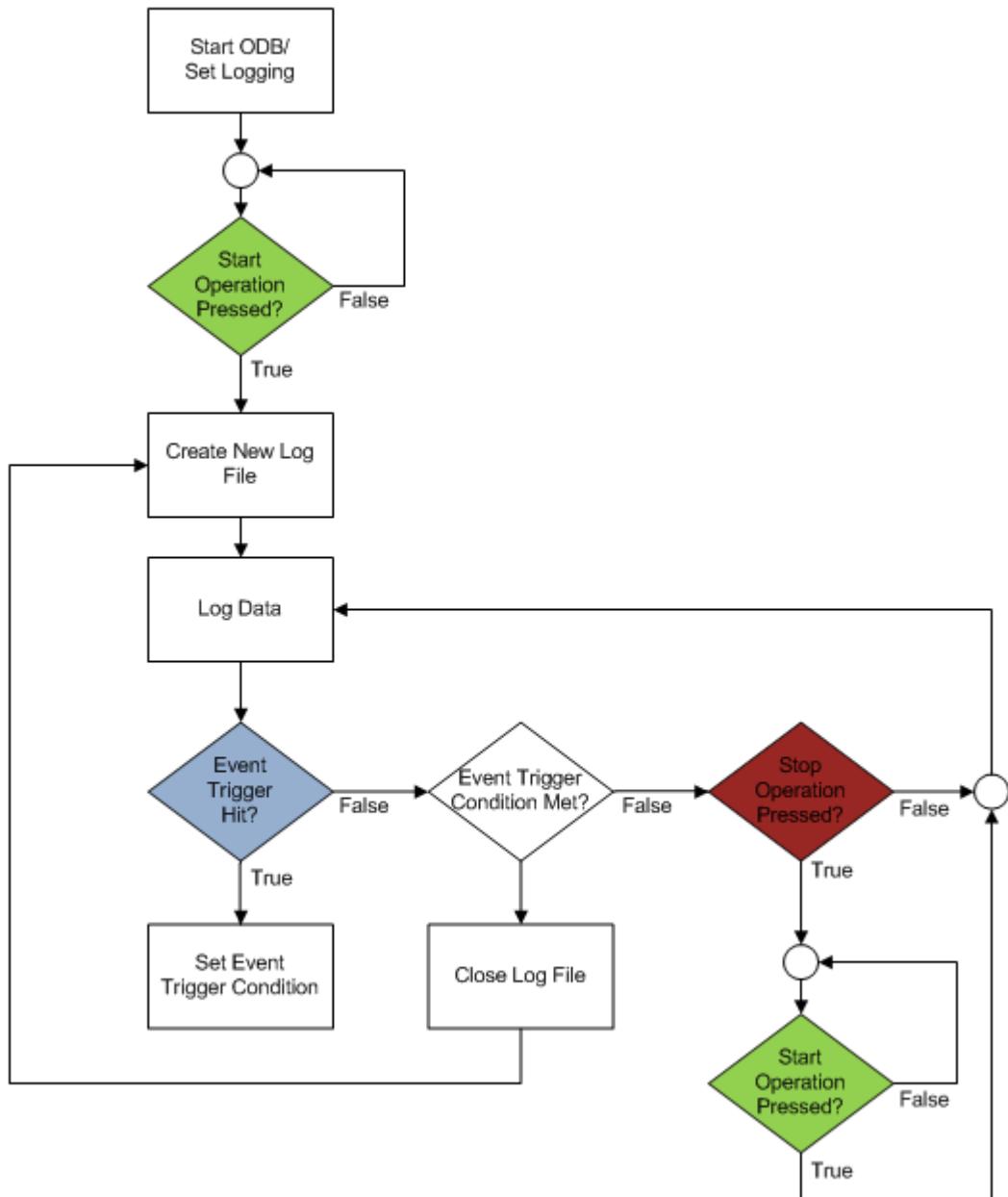
Main Phone: (540) 961-5190

Toll-Free Support: 1.866.LUNA OVA (866.586.2682)

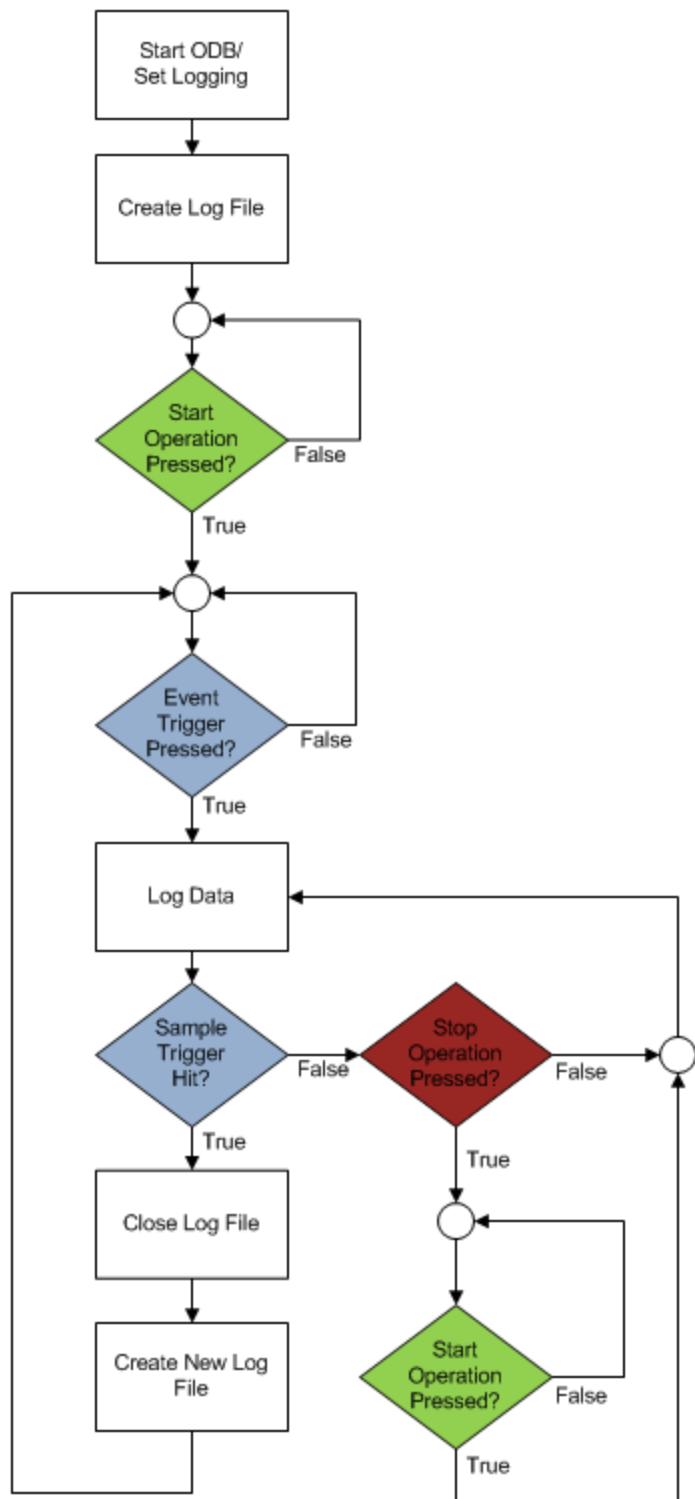
E-mail: support@lunainc.com

www.lunainc.com

Appendix A: Raw Data File and Trigger Logic



B-8-1 Logic diagram for the circular raw data file trigger behavior



B-8-2 Logic diagram for the linear raw data file trigger behavior

Appendix B: Expected Software Performance

The following information is relevant to High-Definition sensors only. High-Speed CFG sensors can be interrogated and post processed at the same rate as the data is acquired.

The measurement update rate as displayed in the Main Graph and Measurement vs Time Graph depends on the mode of operation and the sensor length. This variation depends on the processing time required for each data set. The trends are as shown in the figure below.

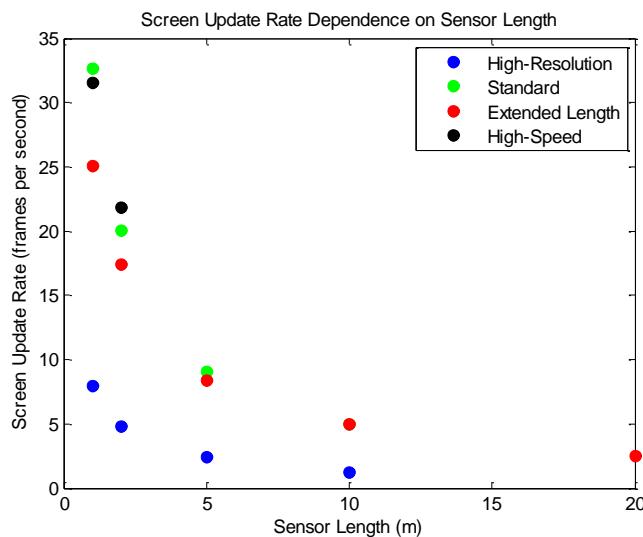


Figure B-1 Screen Update Rate Dependence on Sensor Length and Scan Mode

The amount of time it takes to post process data depends on the operational mode as well as the sensor length. The trends are as shown in the figure below. Timings scale linearly with number of scans.

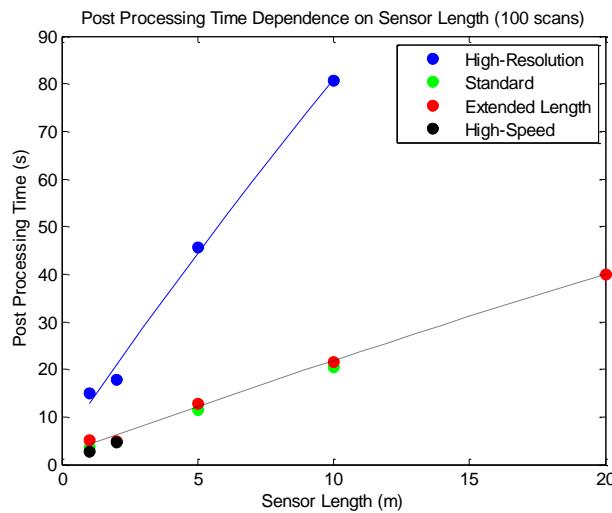


Figure B-2 Post Processing Time Dependence on Sensor Length (for 100 scans)

Appendix C: DAC Setup

Min / Max DAC Voltage

In order to determine the minimum and maximum voltage range of the DAC, launch the Instacal software from Start > Measurement Computing > Instacal. From the main window, ensure that the correct board is highlighted and click on Configure...

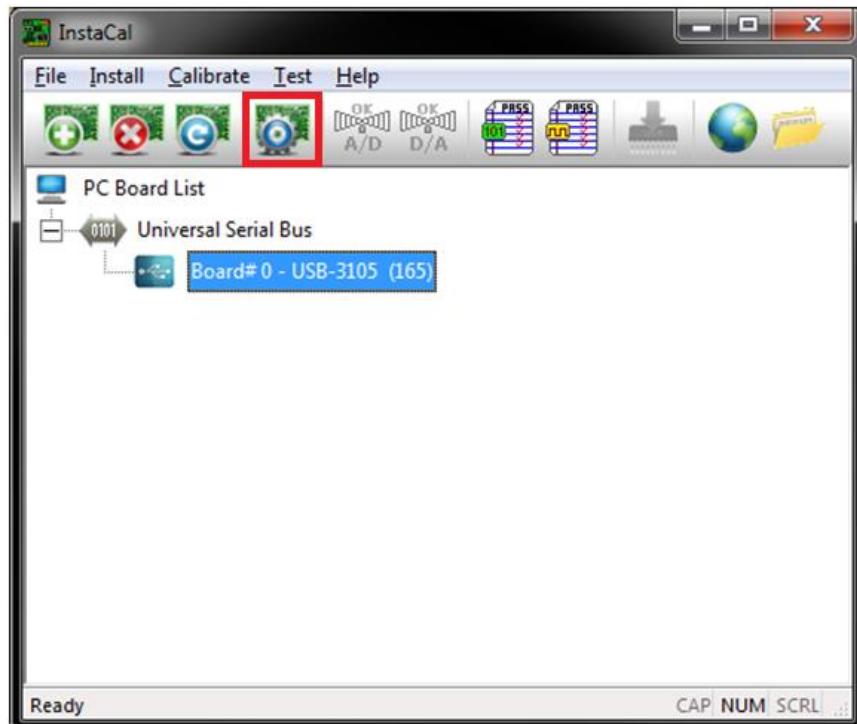


Figure C-1 Instacal Window (configure button is highlighted in red)

Using the dropdown menus in the Board Configuration window, ensure that all channels are configured for a +/- 10V D/A range. When completed, click the OK button to save the settings and close the window.

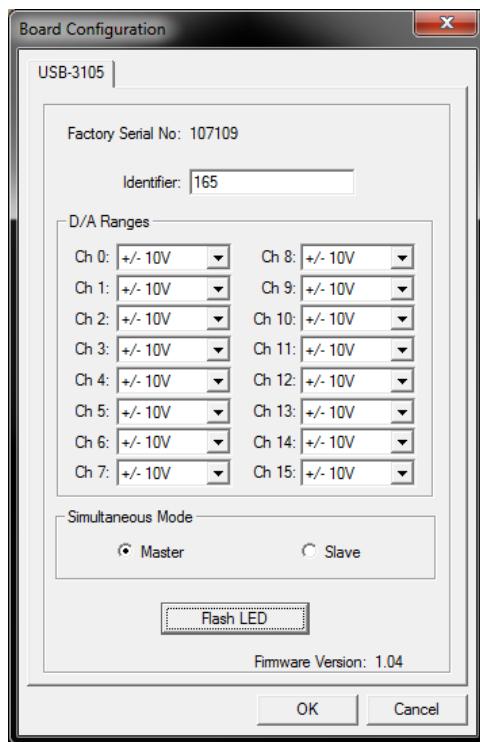


Figure C-2 Instacal Configuration Window

While within the main Instacal window, using the menu bar, navigate to Test>Analog. The Board Test window will pop up. The actual minimum and maximum voltages for each of the output channels can be determined by clicking and turning the DAC input knob or entering a value below the knob (0 to 65535). Read off the DAC Range minimum and maximum values and input these into the DAC Settings Window of the ODiSI-B control software.

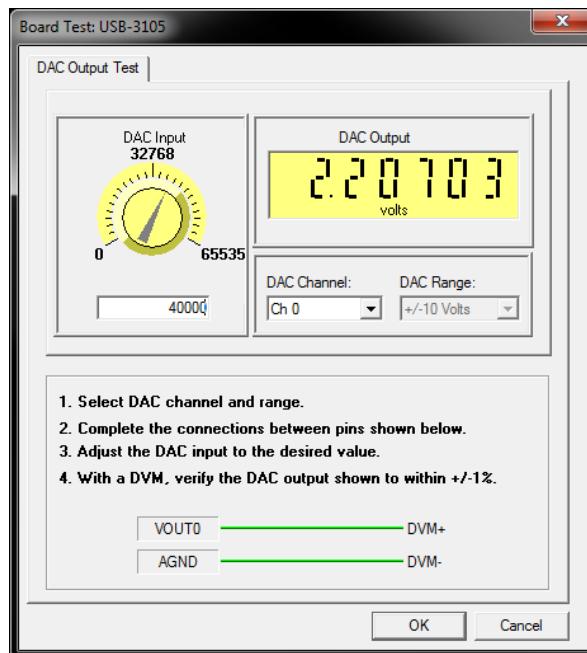


Figure C-3 Instacal DAC Output Test Window

Appendix D: Measurement Theory

The Luna ODiSI-B utilizes swept-wavelength interferometry to interrogate fiber optic sensors. Physical changes to the sensor create a measurable change to how light is scattered from locations along the optical fiber sensor. By comparing the scattered light of a sensor to a reference measurement of the fiber that was recorded at a known strain state, one can determine the physical state of the fiber at the time of measurement. The state of the fiber is coupled to the local environmental temperature and strain.

Fiber Optic Interferometry

The ODiSI-B uses swept-wavelength coherent interferometry. This technique is presented at an introductory level. For more information see the references at the end of this section.

The two basic types of fiber interferometers are shown in Figure F-1.

- A) A Mach-Zehnder design uses one 3 dB coupler to split the light, and another 3 dB coupler to recombine the light. In general, the two paths between the couplers are of different lengths. A detector is placed at the output of the second coupler.
- B) A Michelson interferometer uses a single 3 dB coupler to both split and recombine the light. Mirrors are placed at each of the outputs on one side of the coupler, and in general the two paths are of different lengths. On the other side of the coupler, one lead is used to inject the light while the other lead directs light to a detector.

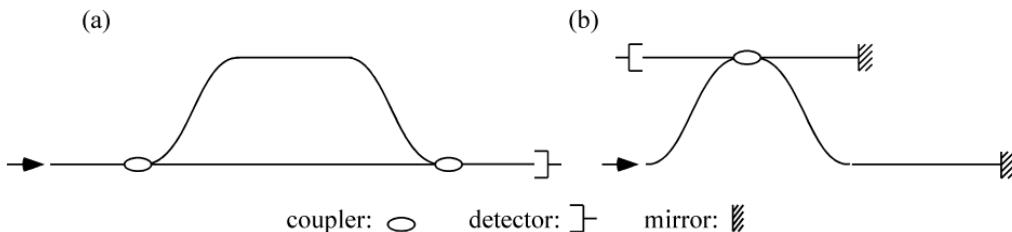


Figure F-8-3 Two basic types of fiber interferometers: (a) Mach-Zehnder and (b) Michelson.

Both of these interferometer designs can be modeled using the same set of mathematics. Consider an input field of the form

$$E_{in} = E_0(t)e^{-i\omega(t)t}$$

Where ω is angular optical frequency, and $\omega(t)$ describes the instantaneous frequency of the tunable laser source.

At the first coupler, the input field is split into two fields, E_1 and E_2 . After propagating through different lengths of fiber, the two fields are recombined, and the resulting field at the detector is

$$E_{out} = E_0(t + \tau_1)e^{-i\omega(t+\tau_1)t} + E_0(t + \tau_2)e^{-i\omega(t+\tau_2)t}$$

Where τ_1 and τ_2 are the delays through the two paths of the interferometer.

The electrical output of the detector is proportional to the optical intensity, I , which is given by the square magnitude of the electrical field:

$$I(\omega) = |E_0(t)^2 + E_0(t-\tau)|^2 + 2E_0(t)E_0(t - \tau) \cos[\omega(t)\tau]]$$

Where the time delay difference between the interferometer paths $\tau = \tau_1 - \tau_2$.

Optical Network for ODiSI-B

A schematic of Luna's ODiSI-B optical network is shown in Figure F-2. The optical system is comprised of a tunable laser source (TLS), an interferometer (which includes the sensor), and a detector.

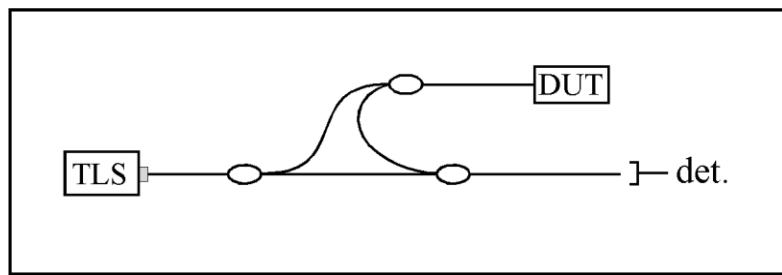


Figure F-8-4 ODiSI-B optical network

The math described in the previous section can be applied to this network by including a sensor in one arm of a Mach-Zender interferometer. We can describe the sensor by its frequency domain, linear transfer function, $H(\omega)$; the linear transfer function contains information about the amplitude, $\rho(\omega)$, and phase, $\phi(\omega)$, response of the sensor and is given by $H(\omega) = \rho(\omega)e^{i\phi(\omega)}$. Given the amplitude and phase response, the detected power is proportional to the intensity, I , given by

$$I(\omega) = |E_0(t)^2 + E_0(t-\tau)|^2\rho(\omega)^2 + 2\rho(\omega)E_0(t)E_0(t - \tau) \cos[\omega(t)\tau - \phi(\omega)]]$$

Next a Fourier transform is performed allowing the three terms in the above expression to be separated spectrally. Note that the first two terms will have only low frequency characteristics, and thus will appear at or near $t = 0$ in the time-domain. The device information resides in the third term of this expression, called the interference term. The interference term oscillates at the frequency $\omega(t)\tau$, while the other terms do not oscillate. Its location in the time-domain will be determined by the delay difference τ between the two arms of the interferometer. Provided that τ is large enough, the device response will be separable from the low frequency terms, and thus measurable. The Luna ODiSI-B is designed such that the internal delay path through the device under test is long enough such that the low frequency terms will not influence the measurement. The Fourier transform of a single, localized reflection is plotted in Figure F-3. Because the initial data was acquired as a function of optical frequency, ω , the transformed data is a function of time. For this reason, the transformed data is referred to as time domain data.

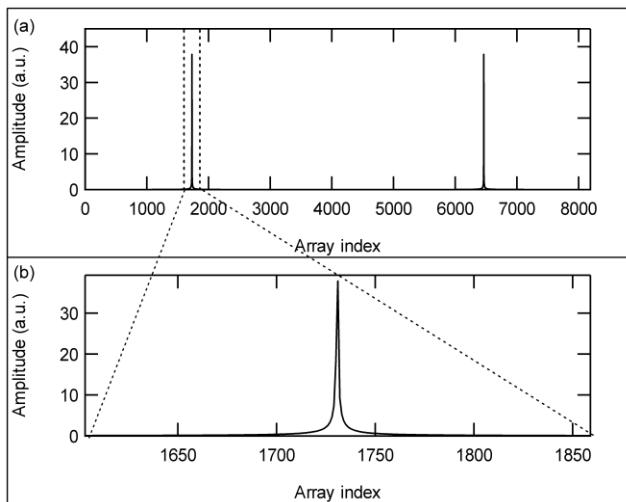


Figure F-8-5 (a) The Fourier transform of interference fringes.

(b) The segment of the time domain data that contains device information.

When a measurement is made with the ODiSI-B, the continually distributed scatter throughout the sensing fiber contributes to the interference pattern as measured by the optical detectors. The physical distance of the scatter location in the fiber is related to the time of flight of light through the sensing fiber. In order to more easily locate the change in temperature or strain in the sensor, the time-domain data can be scaled in units of length using the speed of light and the group index of the sensing fiber.

Distributed Sensing Parameters

The ODiSI-B uses swept-wavelength interferometry (SWI) to measure the Rayleigh backscatter as a function of position in the optical fiber. Rayleigh backscatter in optical fibers is caused by random fluctuations in the index profile along the fiber length. When observing the amplitude of light scattered along the fiber, a highly repeatable profile is observed that is a unique and static property of a given fiber.

The physical length and index of refraction of the fiber are intrinsically sensitive to environmental parameters: temperature and strain, and to a lesser extent, pressure, humidity (if the fiber coating is hydroscopic), electromagnetic fields, etc. In most practical cases the effects of temperature and strain will dominate the spectral response of the Rayleigh backscatter, so we will concentrate our examination on these parameters.

This SWI-based technique enables robust and practical distributed temperature and strain measurements with millimeter-range spatial resolution over tens to hundreds of meters of standard fiber and with strain and temperature resolutions as fine as $1 \mu\epsilon$ and 0.1°C .

Measurement Technique (High-Definition Sensors)

A strain or temperature sensor is formed by first measuring and storing the Rayleigh scatter signature of the fiber under test (FUT) at an ambient state; this data is stored as the reference measurement of the fiber and is referred to as a fiber key. During an ODiSI-B

measurement, the scatter profile is measured at a later time with strain or a temperature change applied at some point along the length of the fiber. The scatter profiles from the two data sets are correlated along the length of the sensor to determine the spectral shift of the scattered light as a result of the applied temperature or strain.

A change in temperature or in strain from the reference state results in a shift in the spectrum of light scattered in the fiber. This shift in the spectrum in response to strain, ϵ , or temperature, T , is analogous to a shift in the resonance wavelength, $\Delta\lambda$, or the spectral shift, $\Delta\nu$ of a Bragg grating:

$$\frac{\Delta\lambda}{\lambda} = \frac{\Delta\nu}{\nu} = K_T T + K_\epsilon \epsilon$$

Where λ and ν are the mean optical wavelength and frequency, and K_T and K_ϵ are the temperature and strain calibration constants, respectively. The default values for these constants are set at values common for most germanosilicate core fibers: $K_T = 6.45 \times 10^{-6}$ °C-1 and $K_\epsilon = 0.780$. The ODiSI-B software allows the user to set their own values for these constants, according to their specific application.

The values for K_T and K_ϵ are somewhat dependent on the doping type and concentration in the core of the fiber, but also to a lesser extent on the composition of the cladding and coating. Variations of 10% in K_T and K_ϵ between standard telecom fibers are common.

It is important to note that the Gauge Length, Δz , affects the spectral resolution and the signal-to-noise ratio of the measurement. There is, therefore, a relationship between the spatial resolution of the measurement and its accuracy in measuring the change in strain or temperature. The longer the gauge length used, the better the measurement accuracy. However, if the strain or temperature varies rapidly with position, a smaller segment size is often necessary to prevent distortion of the cross correlation spectra.

Measurement Technique (High-Speed CFG Sensors)

The processing of High-Speed CFG sensors does not require cross correlations to measure spectral shift. Due to their significantly higher reflectivity (compared to Rayleigh backscatter) it is possible to make spectral shift measurements at the native OFDR resolution (<1.0mm). For these sensors, it is possible to take advantage of the Fourier Shift Theorem.

$$f(x - a) \Leftrightarrow e^{-2\pi i a s} F(S)$$

The change in delay domain (fiber length) phase between two measurements can be scaled into the desired optical frequency shift and then scaled into strain or temperature. These measurements of strain or temperature are then filtered to generate lower noise measurements with spatial resolution on the order of a few millimeters.

Spectral Shift

In SWI the backscatter optical power $U(u)$ is collected in the spectral frequency domain: detectors collect light backscattered from the FUT (fiber under test) as the laser spectral frequency is tuned through a range of frequencies. The data is processed with a Fourier

Transform to generate the backscatter optical power $U(\tau)$ as a function of time delay τ , which can be scaled to show the backscattered amplitude as a function of FUT length.

Although the backscatter spectrum appears random for a segment of fiber, it is actually determined by fluctuations in the core index of refraction and is stable, unique, and repeatable. For a specific fiber segment j , the reflection spectrum is denoted as $U_j(v)$.

If the fiber undergoes a change in strain or temperature, the reflection spectrum at that location in the fiber will experience a shift in optical frequency Δu_j and can be denoted $U_j(v - \Delta u_j)$. The shift in the reflection spectrum Δu_j is computed by performing a cross correlation operation on $U_j(u)$ and $U_j(v - \Delta u_j)$. This spectral shift can then be related to a temperature or strain change, as shown in the equation:

$$\frac{\Delta\lambda}{\lambda} = \frac{\Delta\nu}{v} = K_T T + K_\varepsilon \varepsilon$$

Quality Factor

The Quality Factor is a measure of the strength of the correlation between a measurement and reference spectra. The data is normalized in the ODiSI-B calculation such that a strong correlation is represented by a value of (1) and a weak correlation will approach a value of (0). The calculation will find the strongest peak within the correlation result. The height of the peak will be reported as the quality of the correlation and the location of the peak in the correlation result is the shift in spectra between the measurement and reference. Ultimately this shift is proportional to the temperature or strain change at that location. The noise floor of the correlation result typically has peaks of quality factor score (0.2-0.3). If a strong correlation peak is not observed due to measurement conditions, the reported strain will be invalid. The ODiSI-B software allows the user to filter reported strain results based on the resulting correlation strength, or quality factor, at all sensing locations.

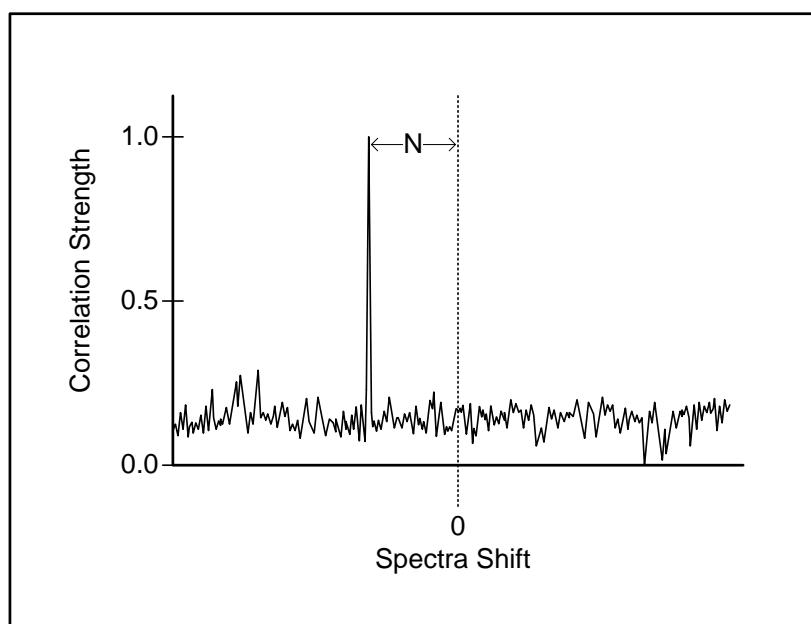


Figure F-8-6 Diagram Depicting Correlation Strength

Strain and Temperature Change

Both temperature and strain cause a change in the light scattered from the fiber. Through judicious experimental planning, the ODiSI-B can provide both temperature and strain measurements along the length of a sensor.

In the absence of strain the temperature change can be written as:

$$\text{Temperature Change, } \Delta T = -\frac{\bar{\lambda}}{cK_T} \Delta v,$$

Where $\bar{\lambda}$ is the center wavelength of the scan and c is the speed of light. Similarly, in the absence of a temperature change, the strain can be written as:

$$\text{Strain, } \varepsilon = -\frac{\bar{\lambda}}{cK_\varepsilon} \Delta v,$$

Assuming a scan center wavelength of 1550 nm, we can substitute in the constants K_T and K_ε to yield the following conversion factors: $\varepsilon = (-6.67 \mu\epsilon/\text{GHz})\Delta u$ and $\Delta T = (-0.634^\circ\text{C}/\text{GHz})\Delta u$.

Thus the distributed temperature and strain curves are merely rescaled copies of the same spectral shift signature.

Temperature Change and Strain Coefficients

The temperature change and strain curves are generated by converting the spectral shift curve from values in GHz to °C or $\mu\epsilon$. This conversion is done using a 4th order polynomial fit. The default coefficients for this fit have been set for standard room temperature and zero strain. The user can customize these coefficients for other operating regimes by recording the frequency shift of a fiber over a region of interest, while measuring temperature or strain on that fiber externally.

References

- B. Soller, D. Gifford, M. Wolfe and M. Froggatt, "High resolution optical frequency domain reflectometry for characterization of components and assemblies," Optics Express, 13:666-674, Jan. 2005.
- B. J. Soller, M. Wolfe, M. E. Froggatt, "Polarization resolved measurement of Rayleigh backscatter in fiber-optic components," paper NWD3, OFC Technical Digest, Los Angeles, March 2005.
- M. Froggatt, T. Erdogan, J. Moore, S. Shenk, "Optical frequency domain characterization (OFDC) of dispersion in optical fiber Bragg gratings," FF2, Special Meeting on Bragg Gratings, Photosensitivity, and Poling in Glass Waveguides, Sept. 1999.

M.M. Ohn, S.Y. Huang, S. Sandgren, R. Measures, T. Alavie, "Measurement of fiber grating properties using an interferometric and Fourier-transform-based technique," WJ2, Optical Fiber Communication Conference, March 1997.

M. Froggatt, J. Moore, and T. Erdogan, "Full complex transmission and reflection characterization of a Bragg grating in a single laser sweep," WB1, Optical Fiber Communication Conference, March 2000.

M. Froggatt, E. Moore, and M. Wolfe, "Interferometric measurement of dispersion in optical components," WK1, Optical Fiber Communication Conference, March 2002.

U. Glombitza and E. Brinkmeyer, "Coherent frequency-domain reflectometry for characterization of single-mode integrated-optical waveguides," J. Lightwave Tech. 11, 1377-1384, 1993.

D. Gifford, B. Soller, M. Wolfe, and M. Froggatt, "Distributed Fiber-Optic Sensing using Rayleigh Backscatter," WB 4, p. 5, European Conference on Optical Communications (ECOC) Technical Digest, Glasgow, Scotland, 2005.

M. Froggatt, B. Soller, D. Gifford, and M. Wolfe, "Correlation and keying of Rayleigh scatter for loss and temperature sensing in parallel optical networks," OFC Technical Digest, paper PDP 17, Los Angeles, March, 2004.

B. J. Soller, D. Gifford, M. Wolfe, and M. Froggatt, "High resolution optical frequency domain reflectometry for characterization of components and assemblies," Opt. Express, 13, pp.666-674, 2005.

B. J. Soller, M. Wolfe, M. E. Froggatt, "Polarization resolved measurement of Rayleigh backscatter in fiber-optic components," paper NWD3, OFC Technical Digest, Los Angeles, March, 2005.

M. Froggatt and J. Moore, "High resolution strain measurement in optical fiber with Rayleigh scatter," Appl. Opt., 37, pp. 1735-1740, 1998.

K. O. Hill, G. Meltz, "Fiber Bragg grating technology—fundamentals and overview," Journal of Lightwave Technology, 15(8), pp.1263-1274, 1997.

A. Kersey, M. Davis, H. Patrick, M. LeBlanc, and K. Koo, "Fiber grating sensors," Journal of Lightwave Technology, 15(8), pp. 1442-1462, 1997.

A. Othonos, "Fiber Bragg gratings," Rev. Sci. Instrum., 68(12), pp. 4309-4340, 1997.

Y. Roa, "In-fibre Bragg grating sensors," Meas. Sci. Technol., 8, pp. 355-375, 1997.

Class 1 Laser Product

The Luna Technologies ODiSI-B is a Class 1 Laser Product, which meets the requirements of Class 1 in standards IEC 60825-1: 2007, and complies with 21CFR1040.10.

Maximum rated output power: 9.0 mW

Internal laser module maximum rated output power: 20.0 mW Emitted wavelengths: 1510 to 1570 nm