BridgeVIEW™ and LabVIEW™

Sound and Vibration Toolset Reference Manual

NATIONAL INSTRUMENTS™

August 1999 Edition
Part Number 322194A-01

Worldwide Technical Support and Product Information

`www.natinst.com`

National Instruments Corporate Headquarters

11500 North Mopac Expressway   Austin, Texas 78759-3504   USA   Tel: 512 794 0100

Worldwide Offices

Australia 03 9879 5166, Austria 0662 45 79 90 0, Belgium 02 757 00 20, Brazil 011 284 5011,
Canada (Ontario) 905 785 0085, Canada (Québec) 514 694 8521, China 0755 3904939, Denmark 45 76 26 00,
Finland 09 725 725 11, France 01 48 14 24 24, Germany 089 741 31 30, Hong Kong 2645 3186,
India 91805275406, Israel 03 6120092, Italy 02 413091, Japan 03 5472 2970, Korea 02 596 7456,
Mexico (D.F.) 5 280 7625, Mexico (Monterrey) 8 357 7695, Netherlands 0348 433466, Norway 32 27 73 00,
Singapore 2265886, Spain (Madrid) 91 640 0085, Spain (Barcelona) 93 582 0251, Sweden 08 587 895 00,
Switzerland 056 200 51 51, Taiwan 02 2377 1200, United Kingdom 01635 523545

For further support information, see the *Technical Support Resources* appendix. To comment on the documentation, send e-mail to `techpubs@natinst.com`.

Important Information

Warranty

The media on which you receive National Instruments software are warranted not to fail to execute programming instructions, due to defects in materials and workmanship, for a period of 90 days from date of shipment, as evidenced by receipts or other documentation. National Instruments will, at its option, repair or replace software media that do not execute programming instructions if National Instruments receives notice of such defects during the warranty period. National Instruments does not warrant that the operation of the software shall be uninterrupted or error free.

A Return Material Authorization (RMA) number must be obtained from the factory and clearly marked on the outside of the package before any equipment will be accepted for warranty work. National Instruments will pay the shipping costs of returning to the owner parts which are covered by warranty.

National Instruments believes that the information in this document is accurate. The document has been carefully reviewed for technical accuracy. In the event that technical or typographical errors exist, National Instruments reserves the right to make changes to subsequent editions of this document without prior notice to holders of this edition. The reader should consult National Instruments if errors are suspected. In no event shall National Instruments be liable for any damages arising out of or related to this document or the information contained in it.

EXCEPT AS SPECIFIED HEREIN, NATIONAL INSTRUMENTS MAKES NO WARRANTIES, EXPRESS OR IMPLIED, AND SPECIFICALLY DISCLAIMS ANY WARRANTY OF MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE. CUSTOMER'S RIGHT TO RECOVER DAMAGES CAUSED BY FAULT OR NEGLIGENCE ON THE PART OF NATIONAL INSTRUMENTS SHALL BE LIMITED TO THE AMOUNT THERETOFORE PAID BY THE CUSTOMER. NATIONAL INSTRUMENTS WILL NOT BE LIABLE FOR DAMAGES RESULTING FROM LOSS OF DATA, PROFITS, USE OF PRODUCTS, OR INCIDENTAL OR CONSEQUENTIAL DAMAGES, EVEN IF ADVISED OF THE POSSIBILITY THEREOF. This limitation of the liability of National Instruments will apply regardless of the form of action, whether in contract or tort, including negligence. Any action against National Instruments must be brought within one year after the cause of action accrues. National Instruments shall not be liable for any delay in performance due to causes beyond its reasonable control. The warranty provided herein does not cover damages, defects, malfunctions, or service failures caused by owner's failure to follow the National Instruments installation, operation, or maintenance instructions; owner's modification of the product; owner's abuse, misuse, or negligent acts; and power failure or surges, fire, flood, accident, actions of third parties, or other events outside reasonable control.

Copyright

Trademarks

BridgeVIEW™, LabVIEW™, natinst.com™, and National Instruments™ are trademarks of National Instruments Corporation.

Product and company names mentioned herein are trademarks or trade names of their respective companies.

WARNING REGARDING MEDICAL AND CLINICAL USE OF NATIONAL INSTRUMENTS PRODUCTS

National Instruments products are not designed with components and testing for a level of reliability suitable for use in or in connection with surgical implants or as critical components in any life support systems whose failure to perform can reasonably be expected to cause significant injury to a human. Applications of National Instruments products involving medical or clinical treatment can create a potential for death or bodily injury caused by product failure, or by errors on the part of the user or application designer. Because each end-user system is customized and differs from National Instruments testing platforms and because a user or application designer may use National Instruments products in combination with other products in a manner not evaluated or contemplated by National Instruments, the user or application designer is ultimately responsible for verifying and validating the suitability of National Instruments products whenever National Instruments products are incorporated in a system or application, including, without limitation, the appropriate design, process and safety level of such system or application.

Conventions

The following conventions are used in this manual:

[ ] — Square brackets enclose the units associated with a control or an indicator.

» — The **»** symbol leads you through nested menu items and dialog box options to a final action. The sequence **File»Page Setup»Options** directs you to pull down the **File** menu, select the **Page Setup** item, and select **Options** from the last dialog box.

This icon denotes a note, which alerts you to important information.

bold — Bold text denotes items that you must select or click on in the software, such as menu items, controls, buttons, and dialog box options. Bold text also denotes parameters and palette names.

italic — Italic text denotes variables, emphasis, a cross reference, or an introduction to a key concept. This font also denotes text that is a placeholder for a word or value that you must supply.

`monospace` — Text in this font denotes text or characters that you should enter from the keyboard, sections of code, programming examples, and syntax examples. This font is also used for the proper names of disk drives, paths, directories, programs, subprograms, subroutines, device names, functions, operations, variables, filenames and extensions, and code excerpts.

Contents

Chapter 5
Frequency Analysis

Chapter 6
Transient Analysis

Chapter 7
Sound Level Measurement

Chapter 8
Octave Analysis

Chapter 9
Weighting Filters

Chapter 10 Display

Appendix A Error Codes

Appendix B References

Appendix C Technical Support Resources

Glossary

Index

Figures

1

Sound and Vibration Toolset Overview

The Sound and Vibration Toolset is a collection of VIs for LabVIEW and BridgeVIEW. These VIs help you work with the typical measurements sound and vibration applications require. This includes:

- Calibration and the handling of engineering units
- Frequency analysis and transient analysis
- Sound level measurements with the typical weighting filters and fractional-octave analysis
- Additional display capabilities

Figure 1-1 illustrates how you can use these different features to perform measurements on existing data. The data can be digitized or simulated.

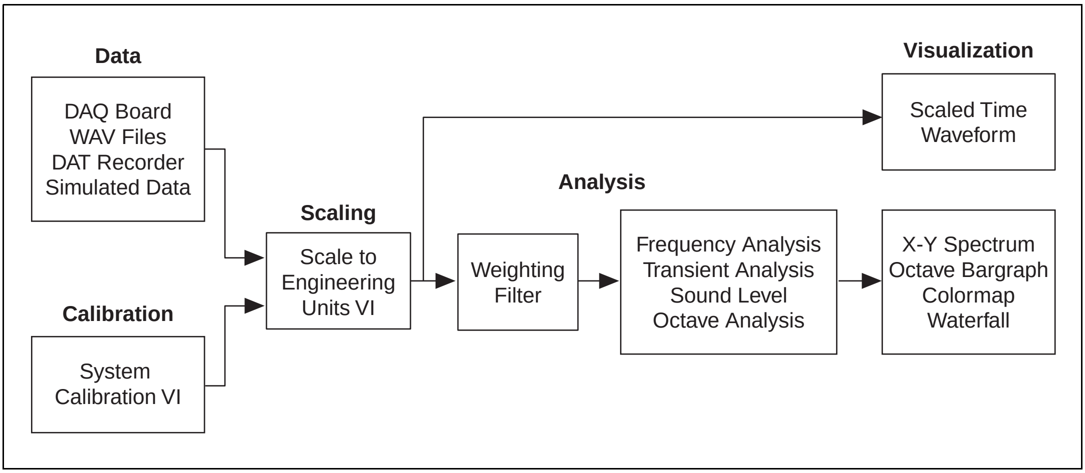


Figure 1-1. Sound and Vibration Toolset Overview Block Diagram

This chapter provides an overview of the Sound and Vibration Toolset components. For more information about each component, refer to the appropriate chapter of this manual.

Controls

The **Controls»S&V Displays** palette offers four customized graphs. You can use the bargraph, linegraph, and linegraph with cursor controls to display the results of the Octave Analysis VIs, and the colormap control to display the results of the Transient Analysis VIs. Figure 1-2 shows the **S&V Displays** palette.

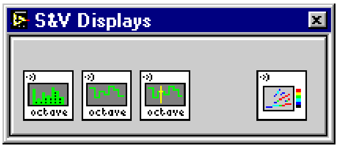


Figure 1-2. S&V Displays Palette

The Sound and Vibration Toolset also provides waterfall graphs for frequency and octave analyses. The Waterfall VIs in the **Functions»Display** palette generate and manage the external window of the waterfall graph.

Functions

The Sound and Vibration Toolset adds a palette to the **Functions** palette. That palette contains the following analysis and display capabilities:

- Units and Calibration
- Frequency Analysis
- Transient Analysis
- Sound Level Measurement
- Octave Analysis
- Weighting Filters
- Display

Figure 1-3 shows the **Functions»Sound and Vibration** palette.

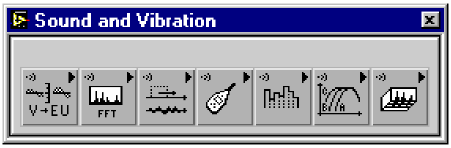


Figure 1-3. Sound and Vibration Palette

All the high-level VIs in the Sound and Vibration Toolset are designed to offer *measurement* capabilities. This means that these VIs perform the selected analysis and allow you to view the results in standard displays, such as magnitude/phase, real/imaginary part, decibels on/off, with the appropriate engineering units.

Units & Calibration

Most of the Sound and Vibration Toolset VIs expect the time waveform to be scaled to the appropriate engineering units. You can use the Scale to Engineering Units VI to accomplish this task. The Scale to Engineering Units VI relies on information the **channel info** control contains. The control contains scaling information such as the sensor sensitivity, the engineering unit to be used (including custom label), the reference value for results expressed in decibels, the eventual use of a hardware weighting filter, if any, and any eventual pregain.

The **sensor sensitivity** parameter is essential to obtain correctly scaled measurement results. You can use the value from the data sheet of the sensor, or, preferably, from a recent calibration performed with a dedicated calibrator and the System Calibration VI. Using this VI replaces any sensitivity entered for the sensor by the actual calibrated sensitivity measured with the calibrator, typically a hand-held shaker for accelerometers or a pistonphone for microphones.

Frequency Analysis

The VIs in the **Frequency Analysis** palette offer a collection of standard frequency analysis tools based on the use of the Discrete or Fast Fourier Transform (DFT or FFT). These tools provide averaged and non-averaged, frequency measurements.

When performing averaged frequency measurements, place the VI in a loop and wire a new block of time data to the VI. The VI then returns the averaged results based on all the data sent since the first call to this VI or since the averaging process was reset. Blocks of time data can be contiguous or can overlap. The blocks of time data also determine the FFT block size, and each block of time data must have a constant size. For example, a block size of 1,024 points typically leads to 400 FFT lines.

Transient Analysis

The VIs in the **Transient Analysis** palette use the Short-Time Fourier Transform (STFT).

You can call these VIs directly to obtain the frequency content of the signal versus time or you can pass the information simultaneously acquired by a tachometer to obtain the frequency content as a function of the rotational speed.

More advanced techniques for transient analysis, such as adaptive, Gabor or Wigner-Ville spectrograms, wavelets, and super resolution spectral analysis, are available in the Signal Processing Toolset available from National Instruments.

Sound Level Measurement

The VIs in the **Sound Level Measurement** palette offer typical sound level measurements. These include linear averaging (equivalent continuous level, or Leq), exponential averaging, and peak hold. The exponential averaged sound level measurements provide the standard Slow, Fast, and Impulse time constants, as well as any custom time constant.

The VIs in the **Advanced** subpalette also offer the basic tools required by sound level measurements. You can use these lower-level tools to perform the same kind of averaging on any signal. For example, you can use the same time constant to analyze a signal coming from an accelerometer instead of a microphone.

Octave Analysis

The VIs in the **Octave Analysis** palette offer an extensive set of tools to perform fractional-octave analysis, including 1/1, 1/3, 1/6, 1/12, and 1/24 octave-band analysis. In addition, these VIs can accommodate any sampling frequency and any number of fractional-octave bands.

When combined with one of the following DAQ boards from National Instruments (PCI-445*X* or NI 455*X*), these VIs offer compliance with ANSI S1.11-1986, Order 3, Type 1-D, optional range, or class 1, IEC 1260: 1995 standards.

Various averaging modes are supported, including linear averaging, exponential averaging, equal confidence, and peak hold.

Additional VIs allow you to select the frequency range to analyze, return information about filter frequency aspects, and format analysis results into a table with the ANSI/IEC standard nominal (preferred) frequencies.

Weighting Filters

The **Weighting Filters** palette offer VIs to apply A, B, or C weighting filters on the time-domain signal.

When combined with one of the following DAQ boards from National Instruments (PCI-445*X* or NI 455*X*), these VIs offer compliance with the ANSI S1.4-1983 and the ANSI S1.42-1986 standards. Based on the specified sampling frequency, the compliance with a particular Type (Type 1 or Type 0) is ensured up to a specific frequency returned by the VI.

Display

The **Display** palette provides VIs to display the results of frequency analysis and octave analysis as waterfall graphs. Some utility VIs are available to format octave results before displaying them on a standard XY graph.

The waterfall display is a separate window that includes the 3D waterfall graph and a palette to control the view, such as orientation, auto-scaling, storing/restoring a specific view, and transparency on/off. When you initialize the waterfall display, you can specify the size and position of this window.

2

Getting Started

This chapter describes the examples in the Getting Started Examples library, located in the `LabVIEW\Examples\Sound and Vibration` directory. More specific examples are available in additional libraries in the same directory.

Examples

The examples described in this chapter cover the following functions in the Sound and Vibration Toolset:

- Frequency analysis
  - Averaged FFT Analysis Example (simulated) VI
  - FFT and Octave Analysis Example (simulated) VI
- Transient analysis
  - Engine Run-up Example (simulated) VI
- Sound level measurements
  - Simple Sound Level Meter (DAQ) VI
- Octave analysis
  - ANSI Third-octave Analysis Example (simulated) VI
  - ANSI Third-octave Analyzer (DAQ) VI
  - FFT and Octave Analysis Example (simulated) VI
- Weighting filters
  - Weighting Filter Example VI
- Display
  - Waterfall Display Example VI

Some of these examples require the use of a data acquisition board. These examples are identified by (DAQ) at the end of the VI name. In comparison, any example using a simulated source is identified by (simulated) at the end of its name. The simulated source these examples use is a sine wave added to a white noise signal of approximately –90 dB (ref 1.0 V). You can select the amplitude and frequency of the simulated sine wave in these examples.

Averaged FFT Analysis Example (simulated)

This example demonstrates how you can use different averaging modes to analyze the simulated signal (sine wave plus white noise). Figure 2-1 shows the Averaged FFT Analysis Example (simulated).

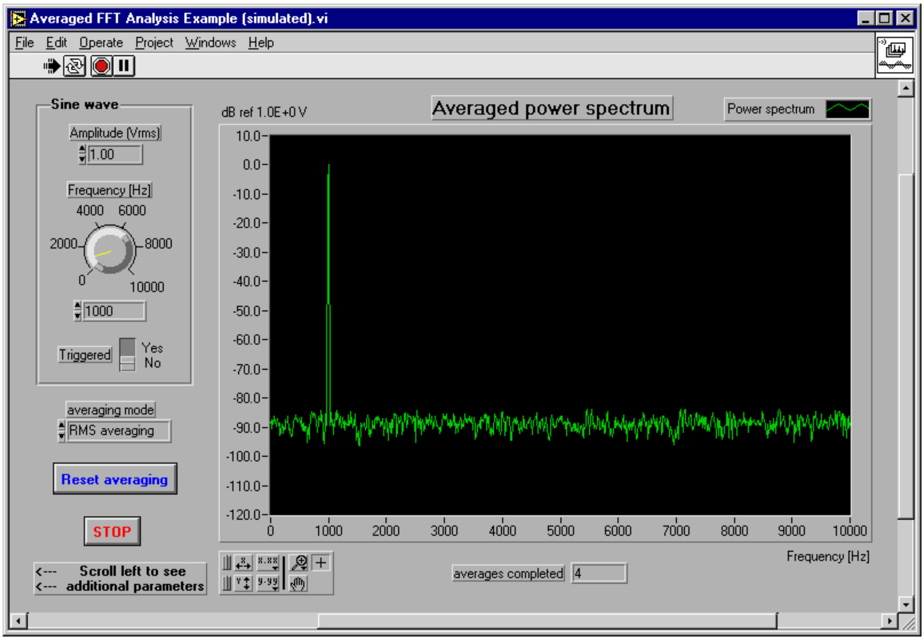


Figure 2-1. Averaged FFT Analysis Example (simulated)

The default **averaging mode** is root-mean-square (RMS). In this case, you can observe how the noise fluctuations are attenuated without any effect on the noise floor, which stays constant at approximately –90 dB (ref 1.0 V).

If you select vector averaging, you cannot obtain a valid result without triggering. Once the trigger has been turned on, the effect of the vector averaging appears clearly with a reduction of the noise floor but no effect on the fluctuations around this noise floor.

This effect is even more noticeable when you analyze frequency components with amplitudes just above the noise floor. For example, you can change the amplitude of the sine wave to 1.0E-4 Vrms and observe the differences in the averaged power spectrum between RMS and vector averaging.

Figure 2-2 shows the block diagram of the Averaged FFT Analysis Example (simulated).

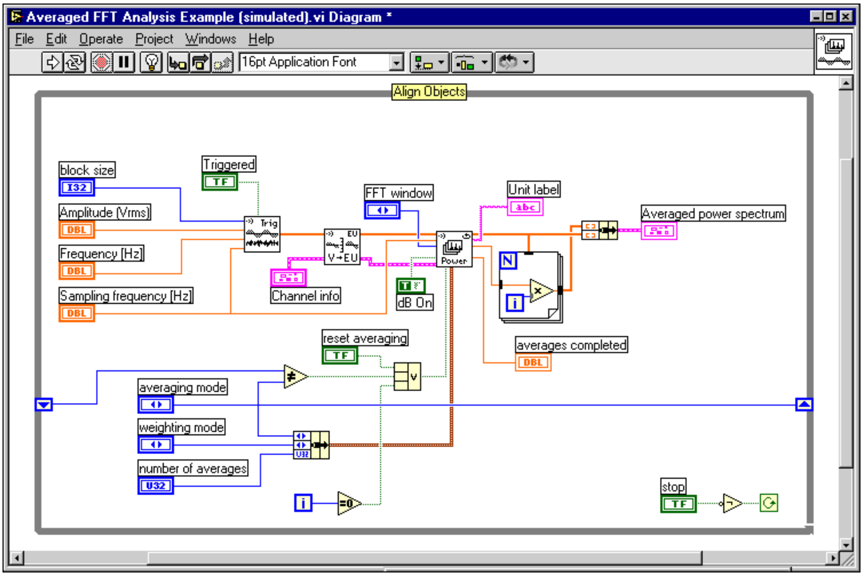


Figure 2-2. Averaged FFT Analysis Example (simulated) Block Diagram

Engine Run-up Example (simulated)

This example uses pre-recorded data to demonstrate the use of the STFT to analyze the noise an engine generates during a run-up, that is, an increase in engine speed, in this particular case, from approximately 1500 rpm to 5200 rpm. Figure 2-3 shows the Engine Run-up Example (simulated).

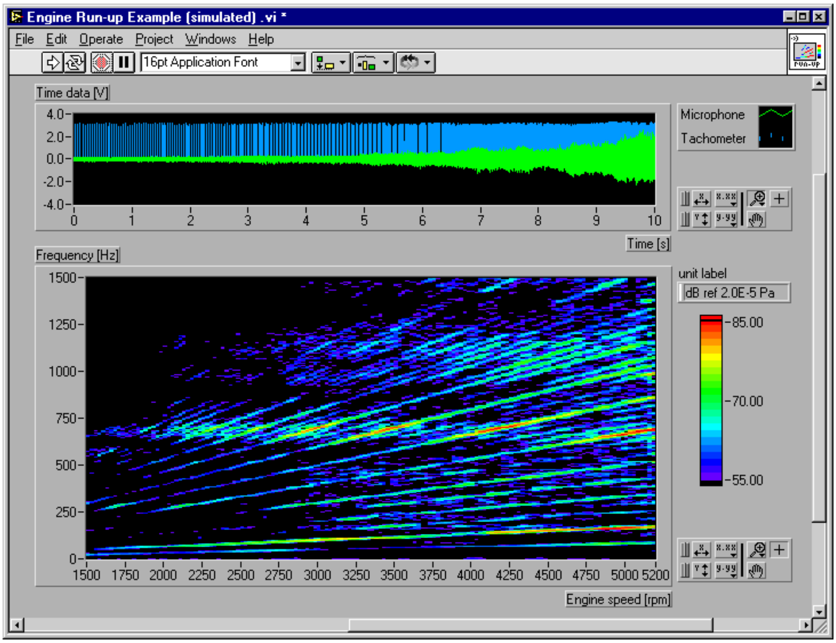


Figure 2-3. Engine Run-up Example (simulated)

Two data acquisition channels were used to acquire the signals presented in this example. The first one, appearing in green on the top graph (time waveforms), is the pressure measured by a microphone located close to the engine. The second one, appearing in blue on the top graph, is the pulse train recorded by the tachometer. This example used an optical encoder generating a pulse every full revolution of the engine.

The bottom graph (colormap) displays the result of the STFT analysis. The X axis represents the engine speed expressed in rpm, and the Y axis represents the frequency expressed in hertz. A color ramp is used to code the sound level measured by the microphone.

The various diagonal straight lines visible on this graph correspond to different orders, or harmonics of the rotational speed. This demonstrates how, in this particular example, various orders excite a structural resonance when crossing the frequency range around 675 Hz.

Additional controls and indicators are available when you scroll to the left on the front panel. The information about the microphone and the tachometer is described in the **tacho info** and the **channel info** clusters, respectively. The other parameters include the engine speed range to analyze, the **FFT block size** to use, the **window**, and any eventual **frequency weighting** to apply.

Figure 2-4 shows the block diagram of the Engine Run-up Example (simulated).

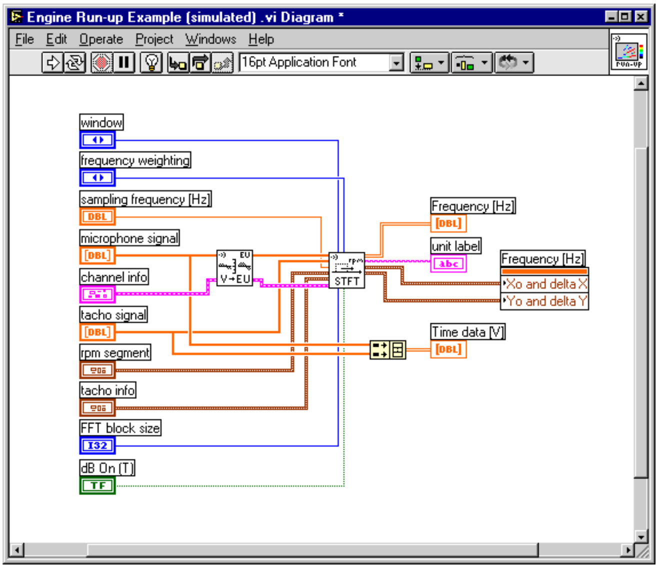


Figure 2-4. Engine Run-up Example (simulated) Block Diagram

Simple Sound Level Meter (DAQ)

This example uses a continuous buffered acquisition to acquire data from a microphone and display the time waveform and the exponential averaged sound level measured by the microphone. Figure 2-5 shows the Simple Sound Level Meter (DAQ) example.

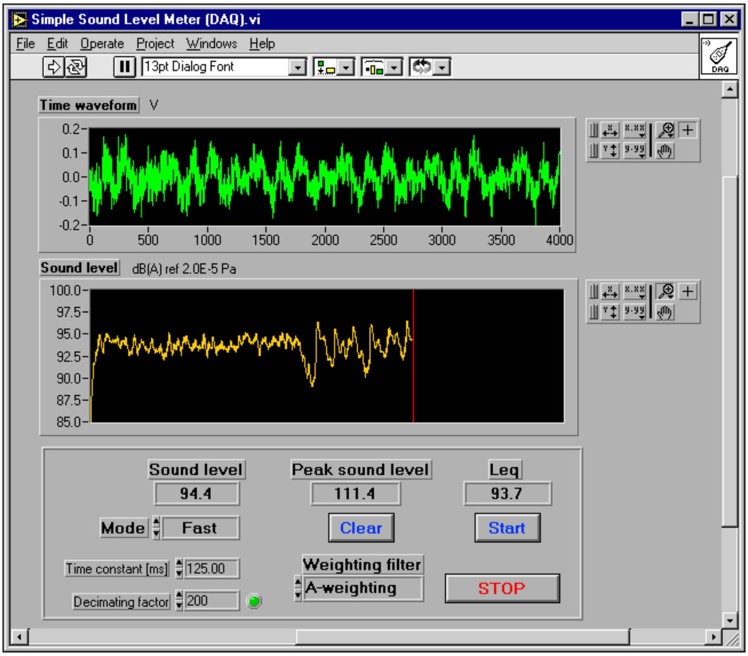


Figure 2-5. Simple Sound Level Meter (DAQ) Example

The **Mode** control allows you to select typical sound level meter time constants, such as Slow, Fast, or Impulse, or any custom time constant, as specified by the **Time constant** control. The VI decimates the exponential average sound level before displaying it on the bottom graph. This decimation process reduces the amount of data to display without losing any important information as long as the **Decimating factor** is selected properly. Refer to Exp Avg Sound Level (decimated) VI for more information on properly selecting the **Decimating factor**. If the **Decimating factor** is incorrect, the LED located to the right of the **Decimating factor** control illuminates red.

You can apply A, B, or C weighting by selecting the corresponding option in the **Weighting filter** control. This filtering operation is applied on the acquired time signal, that is, in the time-domain.

You can compute or reset the peak sound level by clicking the **Clear** button, below the **Peak sound level** indicator. You can compute the continuous equivalent sound level, or Leq, by clicking the **Start** button. To stop this VI, click on the red **Stop** button.

Figure 2-6 shows the block diagram of the Simple Sound Level Meter (DAQ) example.

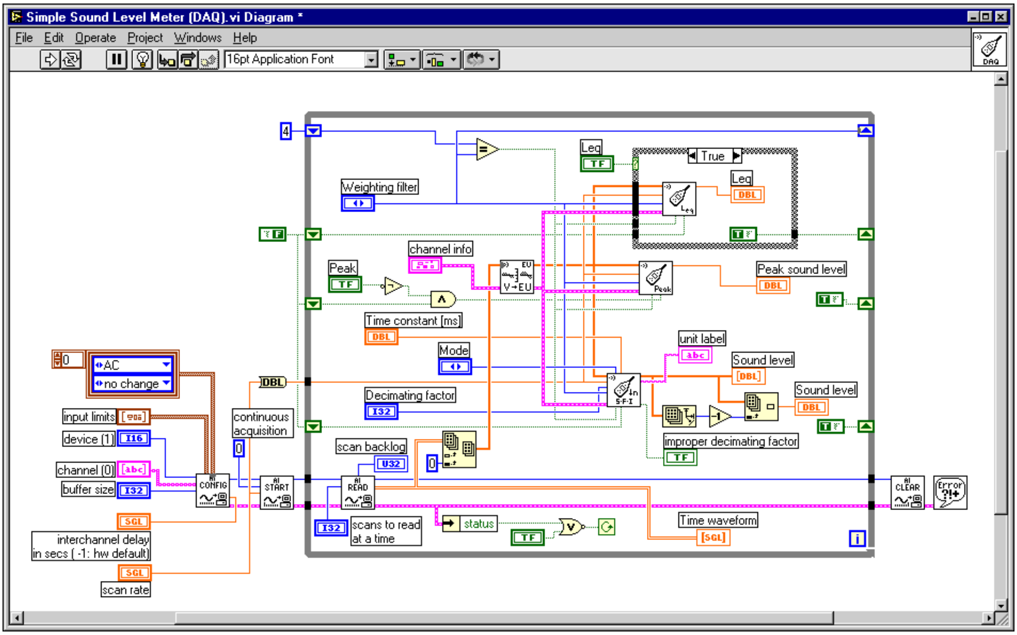


Figure 2-6. Simple Sound Level Meter (DAQ) Example Block Diagram

ANSI Third-octave Analysis Example (simulated)

This example performs third-octave analysis in accordance with the ANSI S1.11-1986 standard on a simulated sine wave. Figure 2-7 shows the ANSI Third-octave Analysis Example (simulated).

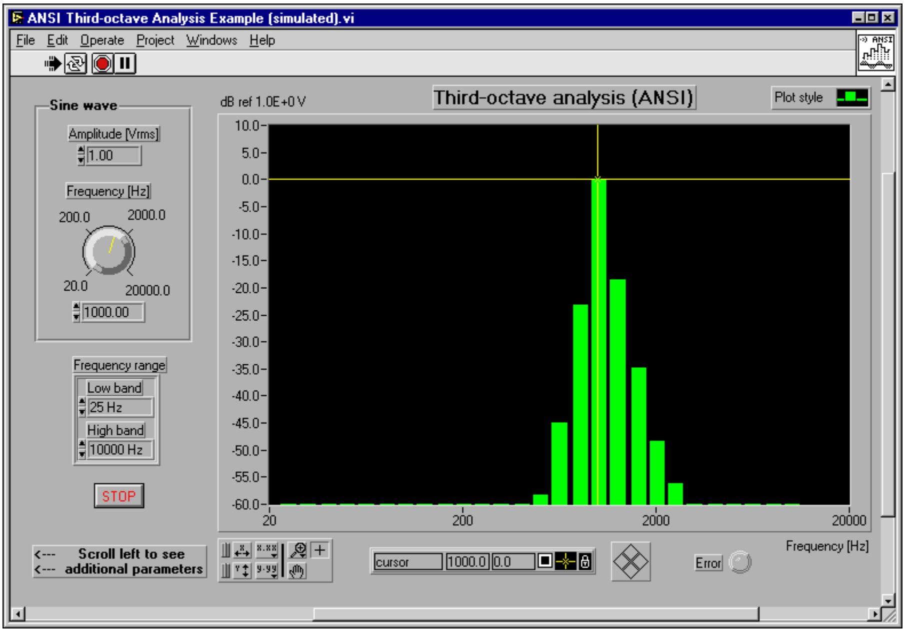


Figure 2-7. ANSI Third-octave Analysis Example (simulated)

The **Low band** and **High band** controls of the **Frequency range** control specify the frequency range of interest for the third-octave analysis.

Note The **Low band** and **High band** controls use a predefined list based on the preferred midband frequencies. The predefined list covers frequencies from 20 Hz to 20 kHz. If you want to extend the third-octave analysis to any particular frequency range, use the ANSI Fractional-octave Analysis VI and set the **bandwidth** parameter to 1/3 octave.

Additional controls are available when you scroll to the left on the front panel. You can use these controls to change the **averaging type** of the third-octave analysis. Exponential averaging is the default type.

Note Because this example uses a simulated source, there is no function to ensure the proper timing of the execution. This example runs as fast as possible, and the different time constants used for the exponential averaging are relative, not absolute. Fast is faster than

Slow but does not necessarily correspond to an actual time constant of 125 ms. The data acquisition board ensures the timing when you acquire real signals, as illustrated in the ANSI Third-octave Analyzer (DAQ) example.

The **sampling frequency** has been set arbitrarily to 51.2 kHz but can be changed to any other frequency. Changing the frequency causes the filter coefficients to be redesigned on the fly and any averaging process is restarted.

Figure 2-8 shows the block diagram of the ANSI Third-octave Analysis Example (simulated).

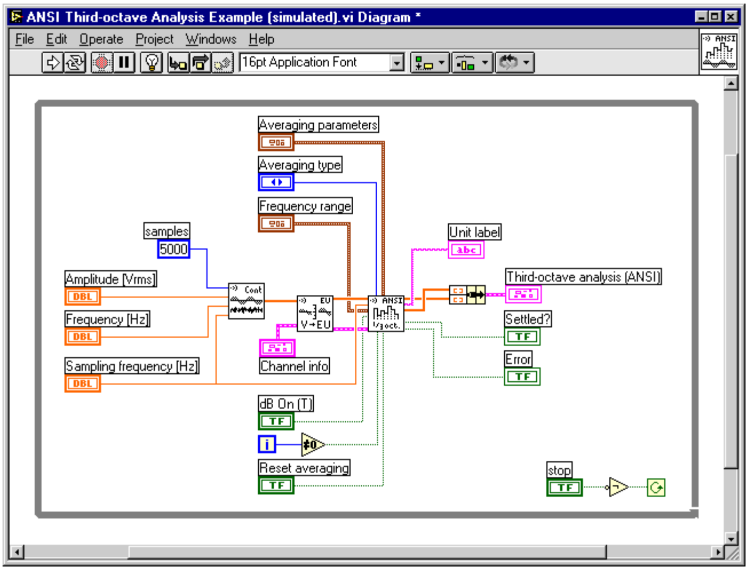


Figure 2-8. ANSI Third-octave Analysis Example (simulated) Block Diagram

ANSI Third-octave Analyzer (DAQ)

This example uses a continuous buffered acquisition to acquire data, typically from a microphone, and performs a third-octave analysis on the signal in accordance with the ANSI S1.11-1986 standard. Figure 2-9 shows the ANSI Third-octave Analyzer (DAQ) example.

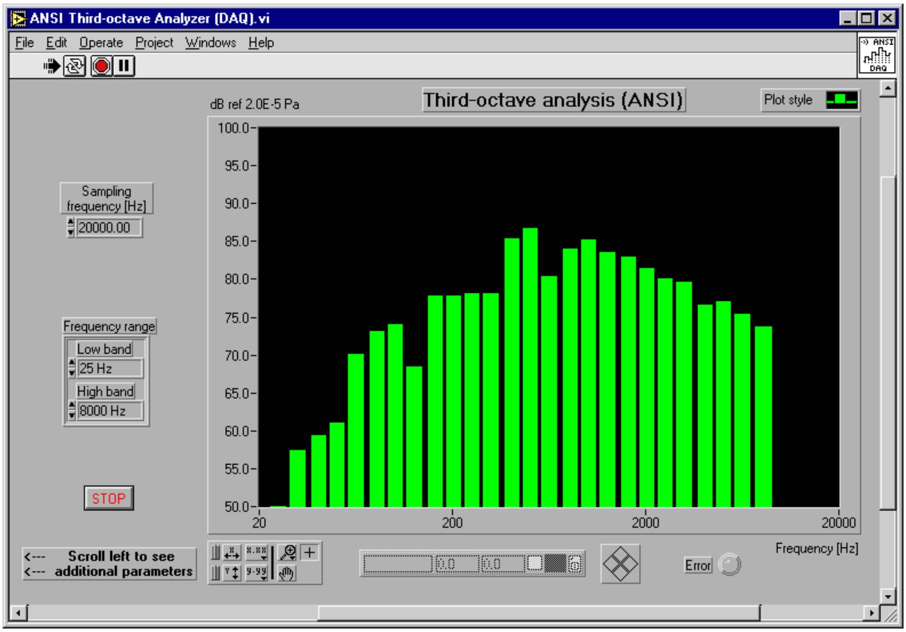


Figure 2-9. ANSI Third-octave Analyzer (DAQ) Example

Refer to the ANSI Third-octave Analysis Example (simulated) for a description of the various controls.

The only additional control is the **number of scans to read** control, visible when you scroll to the left on the front panel. This number corresponds to the amount of samples to read at a time from the circular buffer and then processed by the ANSI Third-octave Analysis VI.

Note The ANSI Third-octave Analysis VI computes the power in each third-octave band, taking into account the last block of data and the previous ones in most averaging modes. The VI returns a single result each time it is called. The refresh rate of the display is therefore determined by the ratio of the sampling rate divided by the number of scans to read. The default values, 20 kHz and 1,000 samples to read at a time, lead to 20 updates of the display per second. Modifying these parameters changes the refresh rate of the display but does not have any effect on the third-octave analysis.

Figure 2-10 shows the block diagram of the ANSI Third-octave Analyzer (DAQ) example.

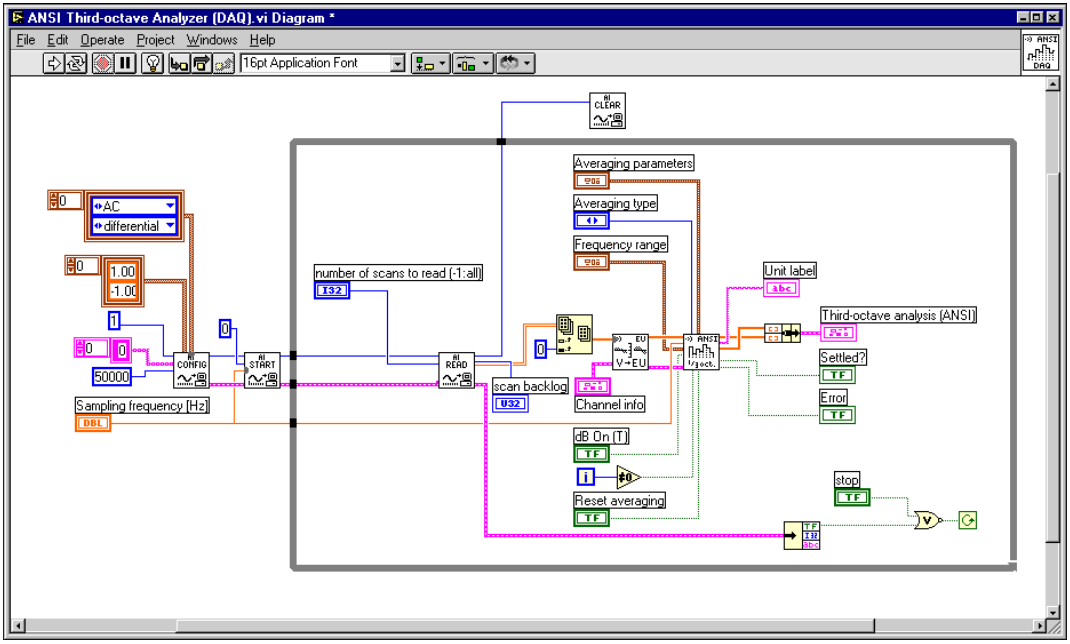


Figure 2-10. ANSI Third-octave Analyzer (DAQ) Example Block Diagram

FFT and Octave Analysis Example (simulated)

This example combines FFT analysis, octave analysis, and third-octave analysis at the same time and on the same simulated sine wave. Figure 2-11 shows the FFT and Octave Analysis Example (simulated).

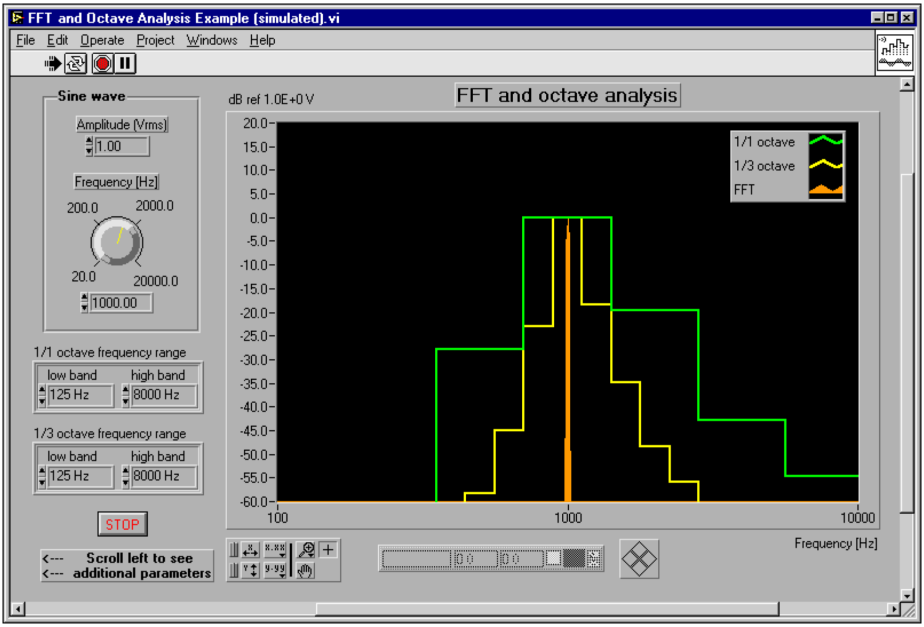


Figure 2-11. FFT and Octave Analysis Example (simulated)

The display uses a logarithmic X axis for the frequency scale. This is typical of any fractional-octave analysis and shows how the resolution of the FFT analysis decreases at low frequency based on a logarithmic scale.

This example also illustrates how the octave analysis results differ from the third-octave analysis results on the same signal.

The **low band** and **high band** controls of the **1/1 octave frequency range** and **1/3 octave frequency range** clusters specify the frequency range of interest for the octave and third-octave analyses. The **low band** and **high band** controls use a predefined list based on the preferred midband frequencies. The predefined list covers frequencies from 20 Hz to 20 kHz.

Additional controls are available when you scroll to the left on the front panel. You can use these controls to change the **averaging type** of the octave and third-octave analyses (exponential averaging is the default type), to specify the **window** for the FFT analysis, and the **block size**.

Note The **block size** parameter corresponds to the block size of the FFT analysis and to the number of data samples to use at a time by the octave and third-octave analyses.

Figure 2-12 shows the block diagram of the FFT and Octave Analysis Example (simulated).

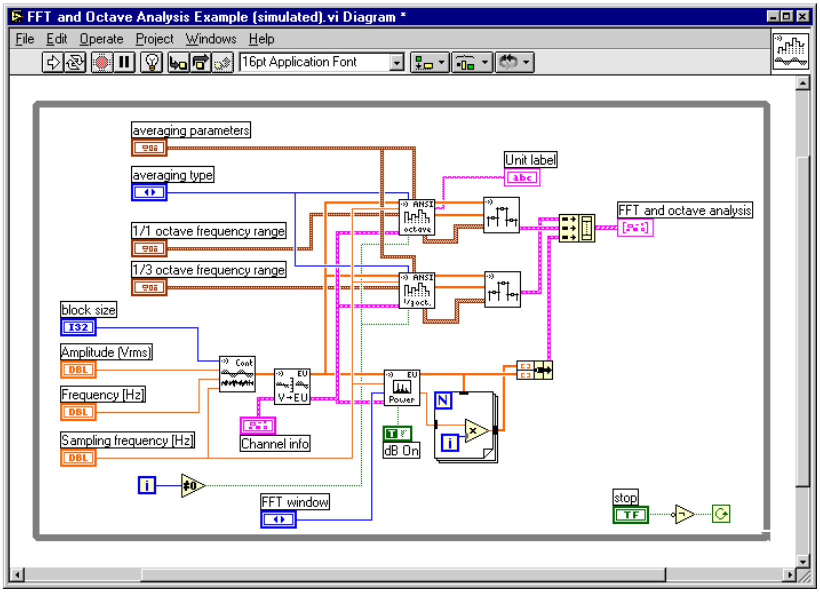


Figure 2-12. FFT and Octave Analysis Example (simulated) Block Diagram

Weighting Filter Example

This example illustrates the difference between applying A, B, or C weighting in the time domain by using filters and applying weighting in the frequency domain by subtracting the theoretical correction to the amplitude or power results from an FFT. Figure 2-13 shows the Weighting Filter example.

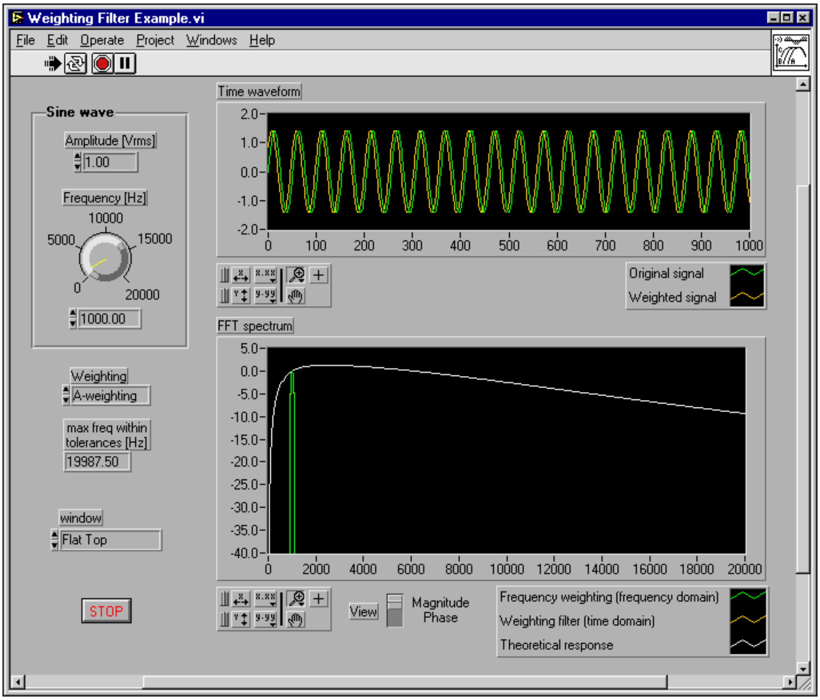


Figure 2-13. Weighting Filter Example

This example performs frequency analysis on a simulated sine wave but uses two different approaches to obtain A, B, or C weighted results. In the first case, an FFT is performed on the original signal, and the frequency weighting is applied to the results in the frequency domain. In the second case, the original signal is filtered by the selected weighting filter, and the FFT analysis is performed on the filtered signal.

The top graph displays the simulated signal and the filtered signal. You can observe on this graph how the amplitude of the signal is modified at different frequencies. Another observation is that the weighting filter not only affects the amplitude but also changes the phase. In comparison, when the weighting is applied in the frequency domain, the selected weighting has no influence on the phase.

The bottom graph illustrates how the two different approaches compare with the theoretical frequency weighting, plotted in white. The advantage of performing the frequency weighting after computing the FFT (green curve) is that if one can assume that the frequency resolution of the FFT is fine enough (sufficient FFT block size), the result perfectly matches the theoretical weighting at all frequencies. But this is relevant only if the desired result is based on the computation of an FFT.

When the weighting operation is performed with a filter (yellow plot), the frequency weighting can deviate from the theoretical curve. In accordance with the ANSI S1.4-1983 standard, the **type** selected defines the upper limit of the frequency range within tolerances. This value is returned in the **max freq within tolerances [Hz]** indicator. For frequencies over the limit, the error exceeds the tolerance of the standard. Nevertheless, use the filter to compute A, B, or C weighted quantities independently of the FFT. This is useful for sound level measurements or fractional-octave analysis.

Figure 2-14 shows the block diagram of the Weighting Filter Example.

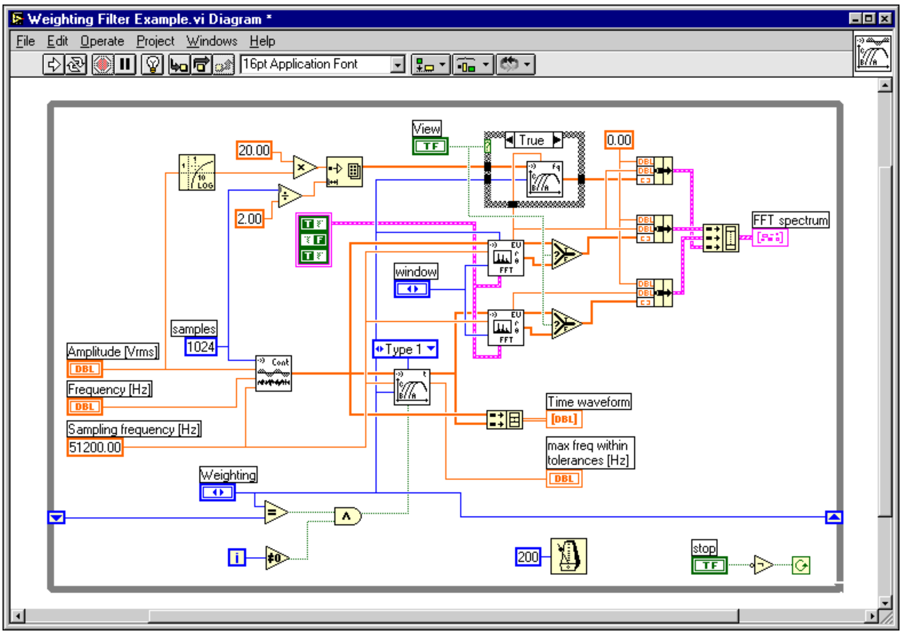


Figure 2-14. Weighting Filter Example Block Diagram

Waterfall Display Example

In this example an amplitude-modulated chirp signal is generated, analyzed by a STFT, and displayed on a waterfall display. Figure 2-15 shows the Waterfall Display example.

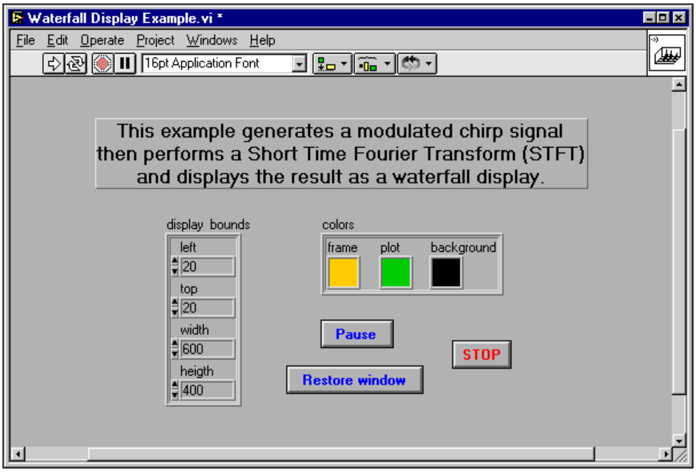


Figure 2-15. Waterfall Display Example

You can use the main window to specify parameters of the waterfall display, such as the position and size of the window and the colors used by the waterfall graph, as shown in Figure 2-15. These parameters are used only once to initialize the waterfall display. To make changes to the waterfall graph, you must stop the Waterfall Display Example VI to reset the parameters.

To pause the generation of the simulated signal, click the **Pause** button. To restore the waterfall window if it was closed unintentionally, click the **Restore window** button.

Figure 2-16 shows the block diagram of the Waterfall Display example.

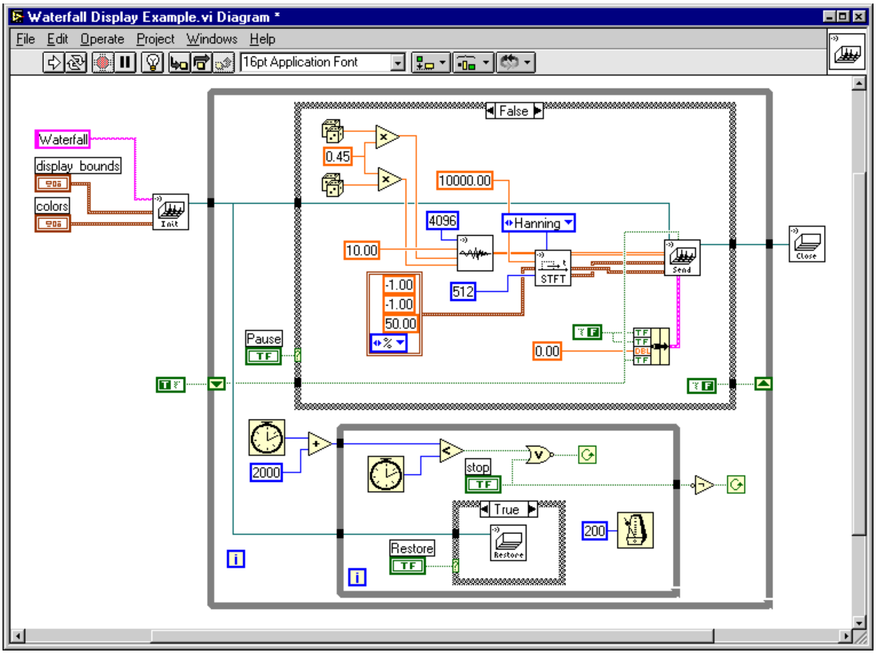


Figure 2-16. Waterfall Display Example Block Diagram

The waterfall display is a separate window that includes the 3D waterfall graph and a palette to control the view. You can use the buttons in the waterfall display window to change the perspective of the display, to store or restore any specific perspective, to scale separately or together the axes of the graph, and to turn transparency on or off, as shown in Figure 2-17.

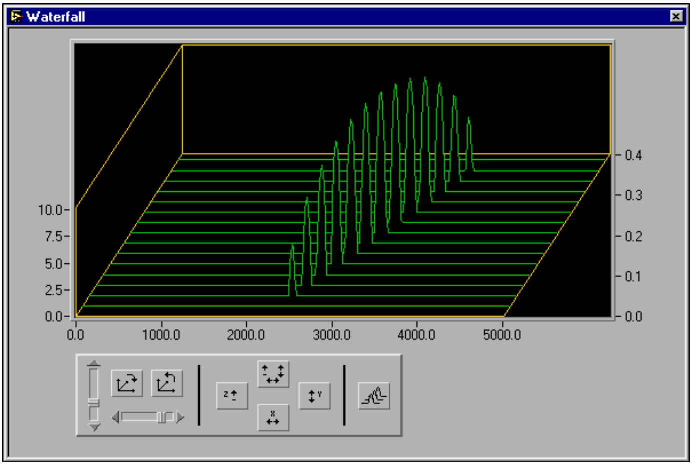


Figure 2-17. Waterfall Window

3

Controls

The **S&V Displays** palette on the **Controls** palette provides customized graphs to use with the Octave Analysis VIs and the Transient Analysis VIs. Figure 3-1 shows the **Controls»S&V Displays** palette.

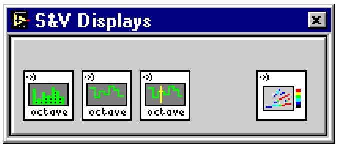


Figure 3-1. S&V Displays Palette

Octave Bargraph

The Octave Bargraph control is a standard bargraph with a logarithmic scale suited to display a single octave spectrum. An example of this display is used in the ANSI Third-octave Analysis Example (simulated), as shown in Figure 3-2.

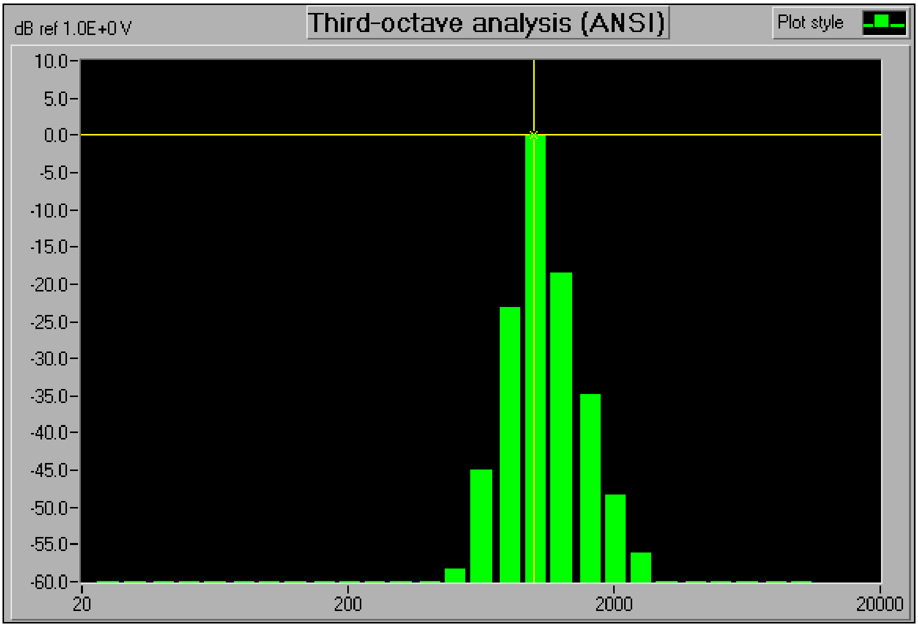


Figure 3-2. Octave Bargraph Control

Octave Linegraph

To display multiple octave spectra on the same graph, you can use the Octave Linegraph control with the Octave Display (line) VI or the Octave Linegraph with Cursor control with the Octave Display (line+cursor) VI. You can find these VIs on the **Functions»Display** palette.

A typical example is the FFT and Octave Analysis Example (simulated), where the results of octave and third-octave analyses are displayed on the same graph, as shown in Figure 3-3.

Using an Octave Linegraph with Cursor control with the Octave Display (line + cursor) VI allows you to lock the cursor to an invisible second plot that contains the results of the octave analysis. The first plot displays the octave bands.

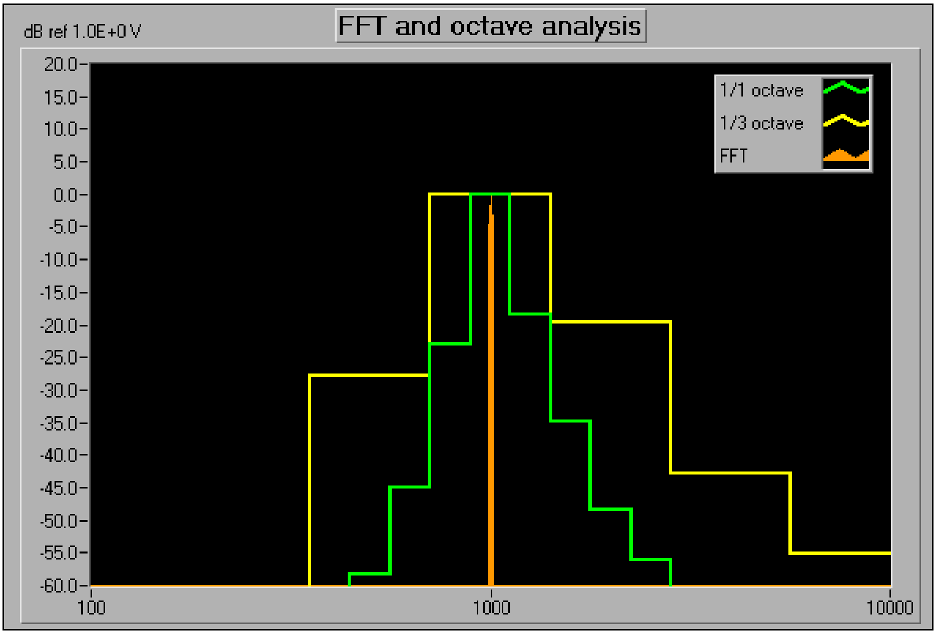


Figure 3-3. Octave Linegraph Control

Colormap

The Colormap control provides a customized intensity graph with a rainbow color scale to display results from transient analysis. You can display these 3D results using colors to code for the third dimension. The Engine Run-up Example (simulated) illustrates the use of this graph to display results as a colormap, as shown in Figure 3-4.

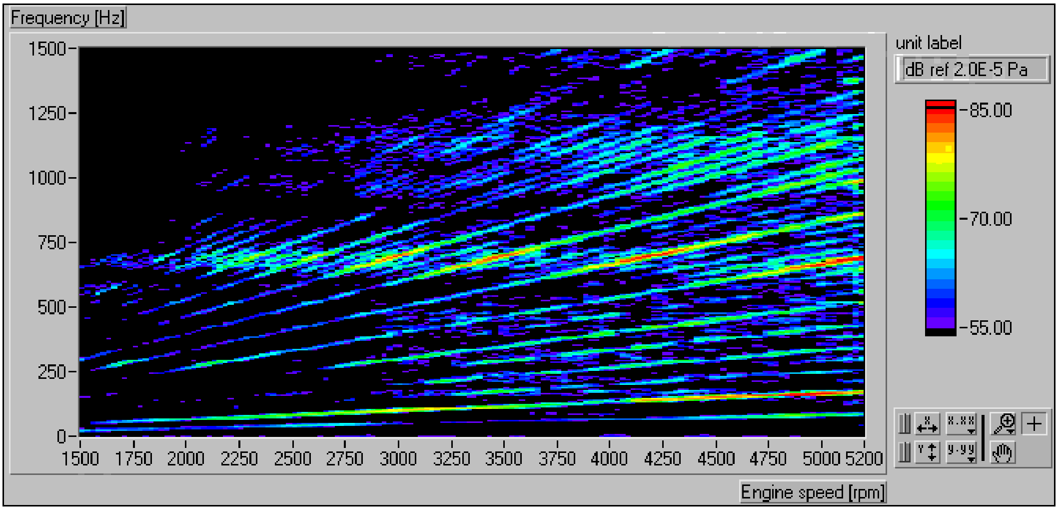


Figure 3-4. Colormap Control

4

Units & Calibration

This chapter describes the Units & Calibration VIs. Figure 4-1 shows the **Functions»Units & Calibration** palette.

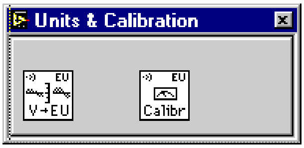


Figure 4-1. Units & Calibration Palette

Introduction

An important feature of the Sound and Vibration Toolset is that all high-level VIs return results with the appropriate units. Time-domain signals can be scaled to the correct engineering units, frequency spectra can be expressed in decibels, phase information can be returned in degrees or radians, and so on.

To handle units properly, the high-level VIs expect the signal to be scaled to the appropriate engineering units. You can use the Scale to Engineering Units VI to accomplish this task. The Scale to Engineering Units VI uses the information from the **channel info** control, as shown below in Figure 4-2.

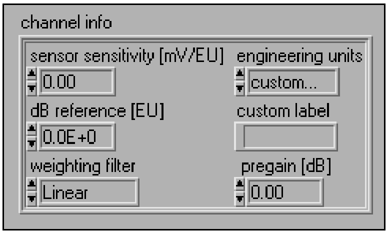


Figure 4-2. Channel Info Control

The **channel info** control contains the following parameters related to a specific input channel:

- Sensor sensitivity [mV/EU] is the sensitivity of the sensor used with this channel. This sensitivity can be a value you enter from a datasheet or can be calibrated using the System Calibration VI.
- Engineering units contains a list of typical engineering units used in the sound and vibration field. If this channel is connected to a microphone, select Pa. If you are using an accelerometer, choose g, m/s^2, or in/s^2. If the units you need to use are not listed, use the custom option and specify the unit to use in the custom label control.
- Custom label is used to specify any custom label not found in the engineering units list. If the engineering units control is not set to custom, the custom label control is ignored.
- dB reference [EU] specifies the reference value to use when computing results in decibels. For sound pressure level measurements, if engineering units are set to Pa, this decibels reference is typically 20E-6 Pa. Results are then returned in decibels ref 20E-6 Pa.
- Weighting filter allows you to specify if any weighting filter (A, B, or C) has been applied to the signal by an external device, such as a filter or an amplifier, before the signal reaches the DAQ board. If a weighting filter has been applied, setting this parameter prevents the VI from applying any additional weighting filter to the weighted signal.

Note Selecting any weighting filter does not perform any filtering, rather it assigns the correct units to the value the VI returns. If you want to apply weighting, refer to Chapter 9, *Weighting Filters*.

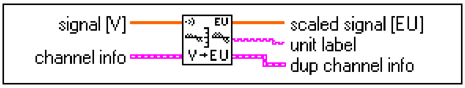
- Pregain [dB] specifies if any pregain, expressed in decibels, has been applied by an external amplifier to the signal before the signal reaches the DAQ board.

The System Calibration VI allows you to perform an end-to-end calibration on a specific channel. This is typically done with a dedicated calibrator, usually a pistonphone for microphones or a hand-held shaker for accelerometers.

The System Calibration VI uses the characteristics of the calibrator (reference calibration value and frequency) to perform the calibration. If no error is encountered during this process, the calibrated sensor sensitivity is returned and stored in the calibrated **channel info** cluster.

Scale to Engineering Units

Scales the original **signal**, expressed in volts, to the selected engineering units (EU). The scaling factor is based on the information **channel info** contains. This scaling factor can be accurately calibrated with the System Calibration VI.



signal [V] contains the signal to be scaled, generally expressed in volts if acquired by a DAQ board.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

- **sensor sensitivity [mV/EU]** contains the sensitivity of the sensor, expressed in mV/EU.

- **engineering units** is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

- **dB reference [EU]** is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

- **custom label** contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

- **weighting filter** indicates if any weighting filter (A, B, or C) has been applied to the signal.

- **pregain [dB]** corresponds to the pregain, expressed in decibels, applied by an external amplifier.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

unit label returns a string that contains the selected engineering units.

dup channel info contains the information relative to the measurement system used before the signal reaches the DAQ board. This information is identical to **channel info**.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

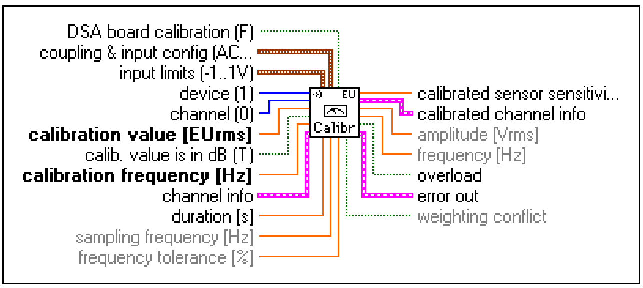
custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

System Calibration

Performs an end-to-end calibration on the selected **channel**. The VI returns **calibrated sensor sensitivity** and stores it in **calibrated channel info**.



DSA board calibration (F) specifies if a board level calibration has to be performed before the system calibration. This option is available only for DSA boards (PCI-445*X* or NI 455*X*). Default is FALSE.

coupling & input config (AC:2; no change: 0) is a cluster that specifies the coupling and input configuration for the selected channel. The default input is AC coupling and no change for **input config**.

coupling

0: Do not change the **coupling** setting

1: DC

2: AC

3: Ground

4: Internal reference

input config

0: Do not change the **input config** setting

1: Differential

2: Referenced single-ended

3: Non-referenced single-ended

input limits (–1..1V) is a cluster that assigns the limits for the selected channel. The default for **input limits** is 1.0 V for the high limit and –1.0 V for the low limit.

high limit (1.0V) specifies the maximum scaled data the device measures for the selected channel.

low limit (–1.0V) specifies the minimum scaled data the device measures for the selected channel.

device (1) is the device number you assigned to the plug-in DAQ board during configuration. The default device number is 1.

channel (0) specifies the analog input channel to be calibrated. The default input is channel 0.

calibration value [EU rms] specifies the reference signal level, expressed in the engineering units selected in **channel info**, generated by the calibrator, such as a pistonphone, or a shaker.

calib. value is in dB (T) specifies if the calibration value is expressed in decibels. If this is the case, the decibels reference value has to be specified in **channel info**. Default is TRUE.

calibration frequency [Hz] specifies the frequency of the reference signal, expressed in hertz.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibel reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

duration [s] specifies how many seconds of data are acquired to calibrate the selected channel. The actual duration of the calibration process could be slightly longer than this value, especially if you select **DSA board calibration**, which causes the VI to perform board-level calibration first.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

frequency tolerance [%] specifies the difference, expressed in percent, this VI tolerates between the actual measured frequency and the specified **calibration frequency**. If the difference exceeds this value, the VI returns an error.

calibrated sensor sensitivity [mV/EU] returns the measured sensor sensitivity, expressed in mV/EU. The VI also returns this value as the **sensor sensitivity** in **calibrated channel info**.

calibrated channel info contains the calibrated sensor sensitivity. All other information is identical to **channel info**.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibel reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

amplitude [Vrms] returns the measured amplitude of the calibration signal, expressed in volts, root-mean-square (Vrms). This is the estimated amplitude of the signal at the calibration frequency only.

frequency [Hz] returns the estimated frequency of the calibration signal, expressed in hertz. If this value is not within the specified **frequency tolerance**, this VI returns an error.

overload specifies if the selected channel has been overloaded during the calibration process. If the channel has been overloaded, increase **input limits** or use a lower calibration value if the calibrator permits.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

status is TRUE if an error occurred.

code is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. Refer to Appendix A, *Error Codes*, for a code description.

source shows where an error occurred. The source string is usually the name of the VI that produced the error.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal.

5 Frequency Analysis

This chapter describes the Frequency Analysis VIs. Figure 5-1 shows the **Functions»Sound and Vibration Toolset»Frequency Analysis** palette.

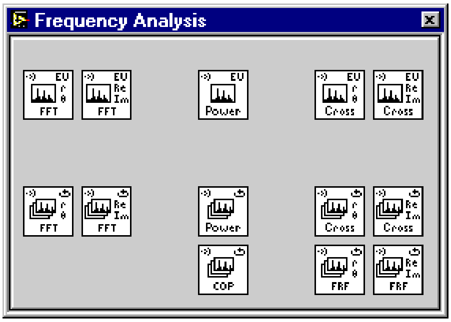


Figure 5-1. Frequency Analysis Palette

Introduction

The Frequency Analysis VIs offer various frequency measurements based on the FFT, if the length of the time record is a power of two, or on the DFT if the length of the time record is not a power of two. You can use these VIs to do averaged, and non-averaged (single block) frequency analysis.

You can view most results as magnitude and phase or as real and imaginary parts. In the case of VIs returning magnitude and phase, the **view** cluster allows you to specify if the magnitude has to be expressed in decibels and if the phase has to be unwrapped and returned in degrees or radians.

Averaging

Non-averaged measurements include the following:

- FFT spectrum
- Power spectrum
- Cross power spectrum

Averaged measurements include the following:

- FFT spectrum
- Power spectrum
- Cross power spectrum
- Frequency response function (H1, H2, or H3) and coherence
- Coherent output power spectrum and coherence

Available averaging modes include the following:

- RMS averaging
- Vector averaging
- Peak hold

RMS averaging reduces signal fluctuations but not the noise floor. The noise floor is not reduced because RMS averaging averages the energy, or power, of the signal. This also causes averaged RMS quantities of single-channel measurements to have zero phase. RMS averaging for dual-channel measurements preserves important phase information.

Vector averaging eliminates noise from synchronous signals. Vector averaging computes the average of complex quantities directly. The real and imaginary parts are averaged separately, reducing noise but usually requiring a trigger.

Peak hold averaging retains the rms peak levels of the averaged quantities. Peak hold is performed at each frequency line separately, retaining rms peak levels from one FFT record to the next.

Note Cross spectrum and frequency response do not support all these averaging modes.

Averaged measurements are computed according to the following equations.

RMS Averaging

FFT spectrum $\sqrt{\langle X^* \bullet X \rangle}$

power spectrum $\langle X^* \bullet X \rangle$

cross spectrum $\langle X^* \bullet Y \rangle$

frequency response $\dfrac{\langle X^* \bullet Y \rangle}{\langle X^* \bullet X \rangle}$ (H1)

$\dfrac{\langle Y^* \bullet Y \rangle}{\langle Y^* \bullet X \rangle}$ (H2)

$$\mathrm{H3} = \frac{(\mathrm{H1} + \mathrm{H2})}{2}$$

where, X is the complex FFT of signal x (stimulus),
Y is the complex FFT of signal y (response),
X^* is the complex conjugate of X,
Y^* is the complex conjugate of Y, and
$\langle X \rangle$ is the average of X, real and imaginary parts being averaged separately.

Vector Averaging

FFT spectrum $\langle X \rangle$

power spectrum $\langle X^* \rangle \bullet \langle X \rangle$

cross spectrum $\langle X^* \rangle \bullet \langle Y \rangle$

frequency response $\dfrac{\langle Y \rangle}{\langle X \rangle}$ (H1 = H2 = H3)

where, X is the complex FFT of signal x (stimulus),
Y is the complex FFT of signal y (response),
X^* is the complex conjugate of X,
$\langle X \rangle$ is the average of X, real and imaginary parts being averaged separately, and
$\langle Y \rangle$ is the average of Y, real and imaginary parts being averaged separately.

Peak Hold

FFT spectrum $\mathrm{MAX}\sqrt{\langle X^* \bullet X\rangle}$

power spectrum $\mathrm{MAX}\langle X^* \bullet X\rangle$

where, X is the complex FFT of signal x (stimulus),
X^* is the complex conjugate of X, and
$\langle X\rangle$ is the average of X, real and imaginary parts being averaged separately.

Coherence and Coherent Output Power

The coherence and coherent output power are computed according to the following equations.

Coherence $$\gamma^2 = \frac{\left|\langle X^* \bullet Y\rangle\right|^2}{\langle X^* \bullet X\rangle\langle Y^* \bullet Y\rangle}$$

Coherent output power $$\mathrm{COP} = \gamma^2\langle Y^* \bullet Y\rangle$$

where, X is the complex FFT of signal x (stimulus),
Y is the complex FFT of signal y (response),
X^* is the complex conjugate of X,
Y^* is the complex conjugate of Y, and
$\langle X\rangle$ is the average of X, real and imaginary parts being averaged separately.

Linear and Exponential Weighting

RMS and vector averaging support the following two different weightings.

- Linear weighting
- Exponential weighting

Weighting is applied according to the following equation:

$$Y_i = \frac{N-1}{N}Y_{i-1} + \frac{1}{N}X_i$$

where, X_i is the result of the analysis performed on the ith block, and
Y_i is the result of the averaging process from X_1 to X_i.
$N = i$ for linear weighting, and
N is a constant for exponential weighting $(N = 1$ for $i = 1)$.

Windowing

Single-channel measurements accept the following windows:

0: Uniform

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

Dual-channel measurements accept the same windows plus:

9: Force-Exponential

Using the Force-Exponential window requires additional parameters specified by the **force/exp settings** cluster.

Frequency Weighting

In addition to windowing, most of these VIs also offer frequency weighting (A, B, or C).

Note If used, frequency weighting is applied to the result of the frequency analysis in the frequency-domain. The weighting is applied to the magnitude only and does not effect the phase. This is different from applying a real time-domain filter offered by the Weighting Filters VIs. For more information, refer to Chapter 9, *Weighting Filters*.

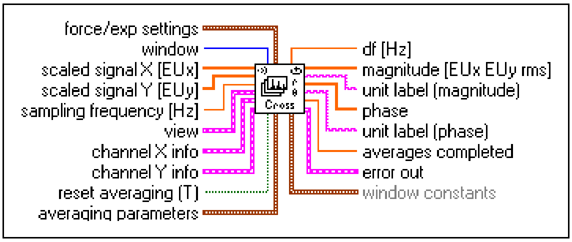
Averaged Cross Spectrum (Mag-Phase)

Computes the averaged cross power spectrum of the input signals. Typically, X signal is the stimulus, and Y signal is the response of the system. Results are returned as magnitude and phase.

Each block of data corresponds to a single FFT block and has to be passed individually to this VI. The first time this VI is called, **reset averaging** must be TRUE (default value) to start a new averaging process. For subsequent calls, **reset averaging** must be FALSE to continue the averaging process. **Averages completed** returns the number of averages completed so far.

Averaging parameters specifies how the averaging is performed. This includes the averaging mode (no averaging, vector averaging, or RMS averaging), the weighting mode (linear or exponential) and the number of averages (used only if the weighting mode is exponential).

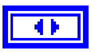
The **view** cluster allows you to display magnitude results in decibels, and the phase results unwrapped or not, in radians or degrees.



force/exp settings is a cluster that contains the parameters for the force and exponential windows.

force window [%] specifies the length, expressed in percent of the total length of the signal, of the force window that is used if **window** is set to Force-Exponential.

exponential window [%] specifies the decay rate of the exponential window that is used if **window** is set to Force-Exponential. This value is the remaining level, expressed in percent, of the window that is applied at the end of the signal.



window is the time-domain window to be used and can be one of the following values:

0: None (Uniform)

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

9: Force-Exponential

scaled signal X [EUx] contains the scaled signal, expressed in the selected engineering units. This is generally the stimulus or excitation signal in dual-channel measurements.

scaled signal Y [EUy] contains the scaled signal, expressed in the selected engineering units. This is generally the response signal in dual-channel measurements.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

view is a cluster that defines how the different results from this VI are returned. Parameters include the ability to use decibels, to unwrap the phase, and to convert the phase to degrees.

dB On (F) specifies if results are expressed in decibels. Default is FALSE.

unwrap phase (F) specifies if the phase has to be unwrapped (TRUE) or not (FALSE, default). Unwrapping eliminates discontinuities with an absolute value greater than π.

convert to degree (F) specifies if the phase results have to be converted from radians to degrees. Default is FALSE, which means that results are expressed in radians.

channel X info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel X.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

channel Y info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel Y.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

reset averaging (T) specifies if the selected averaging process has to be reset. Default is TRUE.

averaging parameters is a cluster that defines how the averaging is computed.

averaging mode specifies the averaging mode. Available modes are No averaging, RMS averaging (default), and Vector averaging.

weighting mode specifies the weighting mode. Available modes are Linear and Exponential.

number of averages specifies the number of averages that is used if the **weighting mode** is Exponential. If **weighting mode** is Linear, this parameter is ignored.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

magnitude [EUx EUy rms] returns the magnitude of the averaged cross power spectrum.

unit label (magnitude) returns a string that corresponds to the selected magnitude units.

phase returns the phase of the averaged cross power spectrum.

unit label (phase) returns a string that corresponds to the selected phase units (radian or degree).

averages completed returns the number of averages completed so far.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

status is TRUE if an error occurred.

code is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. Refer to Appendix A, *Error Codes*, for a code description.

source shows where an error occurred. The source string is usually the name of the VI that produced the error.

window constants contains the following important constants for the selected window:

eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

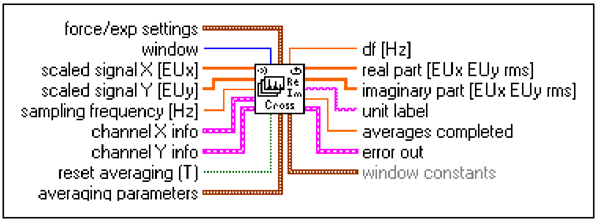
coherent gain is the inverse of the scaling factor applied due to the window.

Averaged Cross Spectrum (Real-Im)

Computes the averaged cross power spectrum of the input signals. Typically, X signal is the stimulus, and Y signal is the response of the system. Results are returned as real and imaginary parts.

Each block of data corresponds to a single FFT block and has to be passed individually to this VI. The first time this VI is called, **reset averaging** must be TRUE (default value), to start a new averaging process. For subsequent calls, **reset averaging** must be FALSE to continue the averaging process. **Averages completed** returns the number of averages completed so far.

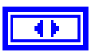
Averaging parameters specifies how the averaging is performed. This includes the averaging mode (no averaging, vector averaging or RMS averaging), the weighting mode (linear or exponential) and the number of averages (used only if the weighting mode is exponential).



force/exp settings is a cluster that contains the parameters for the force and exponential windows.

force window [%] specifies the length, expressed in percent of the total length of the signal, of the force window that is used if **window** is set to Force-Exponential.

exponential window [%] specifies the decay rate of the exponential window that is used if **window** is set to Force-Exponential. This value is the remaining level, expressed in percent, of the window that is applied at the end of the signal.



window is the time-domain window to be used, and can be one of the following values:

0: None (Uniform)

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

9: Force-Exponential

scaled signal X [EUx] contains the scaled signal, expressed in the selected engineering units. This is generally the stimulus or excitation signal in dual-channel measurements.

scaled signal Y [EUy] contains the scaled signal, expressed in the selected engineering units. This is generally the response signal in dual-channel measurements.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

channel X info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel X.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

channel Y info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel Y.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

reset averaging (T) specifies if the selected averaging process has to be reset. Default is TRUE.

averaging parameters is a cluster that defines how the averaging is computed.

averaging mode specifies the averaging mode. Available modes are No averaging, RMS averaging (default), and Vector averaging.

weighting mode specifies the weighting mode. Available modes are Linear and Exponential.

number of averages specifies the number of averages that is used if the **weighting mode** is Exponential. If **weighting mode** is Linear, this parameter is ignored.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

real part [EUx EUy rms] returns the real part of the averaged cross power spectrum.

imaginary part [EUx EUy rms] returns the imaginary part of the averaged cross power spectrum.

unit label returns a string that contains the selected engineering units.

averages completed returns the number of averages completed so far.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

status is TRUE if an error occurred.

code is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. Refer to Appendix A, *Error Codes*, for a code description.

source shows where an error occurred. The source string is usually the name of the VI that produced the error.

window constants contains the following important constants for the selected window:

eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

coherent gain is the inverse of the scaling factor applied due to the window.

Averaged FFT Spectrum (Mag-Phase)

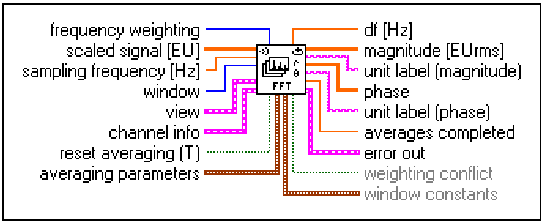
Computes the averaged FFT spectrum of the input signal. FFT results are returned as magnitude and phase.

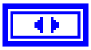
Each block of data corresponds to a single FFT block and has to be passed individually to this VI. The first time this VI is called, **reset averaging** must be TRUE (default value), to start a new averaging process. For subsequent calls, **reset averaging** must be FALSE to continue the averaging process. **Averages completed** returns the number of averages completed so far.

Averaging parameters specifies how the averaging is performed. This includes the averaging mode (no averaging, vector averaging, RMS averaging, or peak hold), the weighting mode (linear or exponential), and the number of averages (used only if the weighting mode is exponential).

The **view** cluster allows you to display magnitude results in decibels, and the phase results unwrapped or not, in radians or degrees.

The selected **frequency weighting** (A, B, or C) is applied to the results in the frequency domain. There is no time-domain filtering operation.

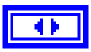




frequency weighting specifies the weighting (A, B, or C) to apply to the signal. This frequency weighting is applied to the spectrum in the frequency domain. Use Linear if no weighting is needed.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.



window is the time-domain window to be used, and can be one of the following values:

0: Uniform

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

view is a cluster that defines how the different results from this VI are returned. Parameters include the ability to use decibels, to unwrap the phase, and to convert the phase to degrees.

dB On (F) specifies if results are expressed in decibels. Default is FALSE.

unwrap phase (F) specifies if the phase has to be unwrapped (TRUE) or not (FALSE, default). Unwrapping eliminates discontinuities with an absolute value greater than π.

convert to degree (F) specifies if the phase results have to be converted from radians to degrees. Default is FALSE, which means that results are expressed in radians.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

reset averaging (T) specifies if the selected averaging process has to be reset. Default is TRUE.

averaging parameters is a cluster that defines how the averaging is computed.

averaging mode specifies the averaging mode. Available modes are No averaging, RMS averaging (default), Vector averaging, and Peak hold.

weighting mode specifies the weighting mode. Available modes are Linear and Exponential.

number of averages specifies the number of averages that is used if the **weighting mode** is Exponential. If **weighting mode** is Linear, this parameter is ignored.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

magnitude [EUrms] returns the magnitude of the averaged FFT spectrum.

unit label (magnitude) returns a string that corresponds to the selected magnitude units.

phase returns the phase of the averaged FFT spectrum.

unit label (phase) returns a string that corresponds to the selected phase units (radian or degree).

averages completed returns the number of averages completed so far.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

status is TRUE if an error occurred.

code is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. Refer to Appendix A, *Error Codes*, for a code description.

source shows where an error occurred. The source string is usually the name of the VI that produced the error.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and **frequency weighting** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

window constants contains the following important constants for the selected window:

eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

coherent gain is the inverse of the scaling factor applied due to the window.

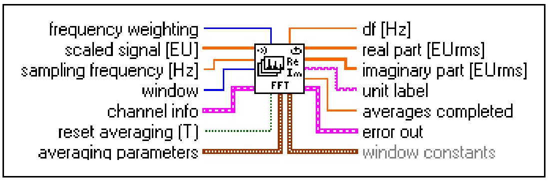
Averaged FFT Spectrum (Real-Im)

Computes the averaged FFT spectrum of the input signal. FFT results are returned as real and imaginary parts.

Each block of data corresponds to a single FFT block and has to be passed individually to this VI. The first time this VI is called, **reset averaging** must be TRUE (default value), to start a new averaging process. For subsequent calls, **reset averaging** must be FALSE to continue the averaging process. **Averages completed** returns the number of averages completed so far.

Averaging parameters specifies how the averaging is performed. This includes the averaging mode (no averaging, vector averaging, RMS averaging, or peak hold), the weighting mode (linear or exponential), and the number of averages (used only if the weighting mode is exponential).

The selected **frequency weighting** (A, B, or C) is applied to the results in the frequency domain. There is no time-domain filtering operation.



frequency weighting specifies the weighting (A, B, or C) to apply to the signal. This frequency weighting is applied to the spectrum in the frequency domain. Use Linear if no weighting is needed.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

window is the time-domain window to be used, and can be one of the following values:

0: Uniform

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

reset averaging (T) specifies if the selected averaging process has to be reset. Default is TRUE.

averaging parameters is a cluster that defines how the averaging is computed.

averaging mode specifies the averaging mode. Available modes are No averaging, RMS averaging (default), Vector averaging, and Peak hold.

weighting mode specifies the weighting mode. Available modes are Linear and Exponential.

U32 **number of averages** specifies the number of averages that is used if the **weighting mode** is Exponential. If **weighting mode** is Linear, this parameter is ignored.

DBL **df [Hz]** is the frequency resolution of the spectrum, expressed in hertz.

[DBL] **real part [EUrms]** returns the real part of the averaged FFT spectrum.

[DBL] **imaginary part [EUrms]** returns the imaginary part of the averaged FFT spectrum.

abc **unit label** returns a string that contains the selected engineering units.

DBL **averages completed** returns the number of averages completed so far.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

TF **status** is TRUE if an error occurred.

I32 **code** is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. Refer to Appendix A, *Error Codes*, for a code description.

abc **source** shows where an error occurred. The source string is usually the name of the VI that produced the error.

TF **weighting conflict** returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and **frequency weighting** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

window constants contains the following important constants for the selected window:

DBL **eq noise BW** is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

DBL **coherent gain** is the inverse of the scaling factor applied due to the window.

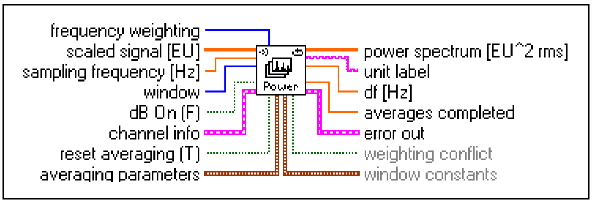
Averaged Power Spectrum

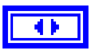
Computes the averaged power spectrum of the input signal.

Each block of data corresponds to a single FFT block and has to be passed individually to this VI. The first time this VI is called, **reset averaging** must be TRUE (default value), to start a new averaging process. For subsequent calls, **reset averaging** must be FALSE to continue the averaging process. **Averages completed** returns the number of averages completed so far.

Averaging parameters specifies how the averaging is performed. This includes the averaging mode (no averaging, vector averaging, RMS averaging, or peak hold), the weighting mode (linear or exponential), and the number of averages (used only if the weighting mode is exponential).

The selected **frequency weighting** (A, B, or C) is applied to the results in the frequency domain (no time-domain filtering operation).

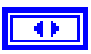




frequency weighting specifies the weighting (A, B, or C) to apply to the signal. This frequency weighting is applied to the spectrum in the frequency domain. Use Linear if no weighting is needed.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.



window is the time-domain window to be used, and can be one of the following values:

0: Uniform

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

dB On (F) specifies if results are expressed in decibels. Default is FALSE.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

reset averaging (T) specifies if the selected averaging process has to be reset. Default is TRUE.

averaging parameters is a cluster that defines how the averaging is computed.

averaging mode specifies the averaging mode. Available modes are No averaging, RMS averaging (default), Vector averaging, and Peak hold.

weighting mode specifies the weighting mode. Available modes are Linear and Exponential.

number of averages specifies the number of averages that is used if the **weighting mode** is Exponential. If the **weighting mode** is Linear, this parameter is ignored.

power spectrum [EU^2 rms] returns the averaged power spectrum.

unit label returns a string that contains the selected engineering units.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

averages completed returns the number of averages completed so far.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

status is TRUE if an error occurred.

code is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. Refer to Appendix A, *Error Codes*, for a code description.

source shows where an error occurred. The source string is usually the name of the VI that produced the error.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and **frequency weighting** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

window constants contains the following important constants for the selected window:

eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

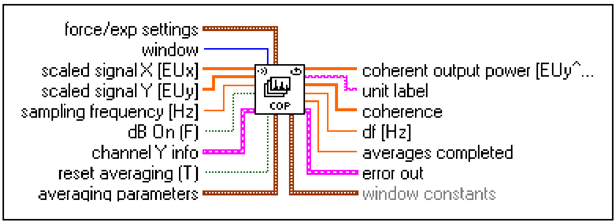
coherent gain is the inverse of the scaling factor applied due to the window.

Coherent Output Power

Computes the coherent output power and the coherence based on the input signals. Typically, X signal is the stimulus and Y signal is the response of the system.

Coherent output power is typically an averaged measurement. Each block of data corresponds to a single FFT block and must be passed individually to this VI. The first time this VI is called, **reset averaging** must be TRUE (default value), to start a new averaging process. For subsequent calls, **reset averaging** must be FALSE to continue the averaging process. **Averages completed** returns the number of averages completed so far.

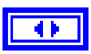
Averaging parameters specifies how the averaging is performed. The only available averaging mode is RMS averaging. Other parameters include the weighting mode (linear or exponential) and the number of averages (used only if the weighting mode is exponential).



force/exp settings is a cluster that contains the parameters for the force and exponential windows.

force window [%] specifies the length, expressed in percent of the total length of the signal, of the force window that is used if **window** is set to Force-Exponential.

exponential window specifies the decay rate of the exponential window that is used if **window** is set to Force-Exponential. This value is the remaining level, expressed in percent, of the window that is applied at the end of the signal.



window is the time-domain window to be used, and can be one of the following values:

0: None (Uniform)

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

9: Force-Exponential

scaled signal X [EUx] contains the scaled signal, expressed in the selected engineering units. This is generally the stimulus or excitation signal in dual-channel measurements.

scaled signal Y [EUy] contains the scaled signal, expressed in the selected engineering units. This is generally the response signal in dual-channel measurements.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

dB On (F) specifies if results are expressed in decibels. Default is FALSE.

channel Y info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel Y.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

reset averaging (T) specifies if the selected averaging process has to be reset. Default is TRUE.

averaging parameters is a cluster that defines how the averaging is computed.

averaging mode specifies the averaging mode. The only available averaging mode is RMS averaging (default).

weighting mode specifies the weighting mode. Available modes are Linear and Exponential.

number of averages specifies the number of averages that is used if the **weighting mode** is Exponential. If the **weighting mode** is Linear, this parameter is ignored.

coherent output power [EUy^2 rms] returns the coherent output power.

unit label returns a string that contains the selected engineering units.

coherence returns the coherence.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

averages completed returns the number of averages completed so far.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

status is TRUE if an error occurred.

code is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. Refer to Appendix A, *Error Codes*, for a code description.

source shows where an error occurred. The source string is usually the name of the VI that produced the error.

window constants contains the following important constants for the selected window:

eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

coherent gain is the inverse of the scaling factor applied due to the window.

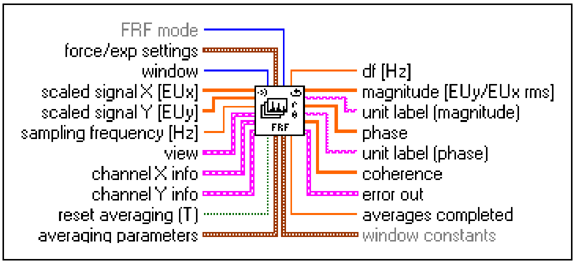
Frequency Response (Mag-Phase)

Computes the frequency response and the coherence based on the input signals. Typically, X signal is the stimulus, and Y signal is the response of the system. Results are returned as magnitude and phase.

Frequency response is typically an averaged measurement. Each block of data corresponds to a single FFT block and must be passed individually to this VI. The first time this VI is called, **reset averaging** must be TRUE (default value), to start a new averaging process. For subsequent calls, **reset averaging** must be FALSE to continue the averaging process. **Averages completed** returns the number of averages completed so far.

Averaging parameters specifies how the averaging is performed. This includes the averaging mode (no averaging, vector averaging, or RMS averaging), the weighting mode (linear or exponential), and the number of averages (used only if the weighting mode is exponential).

The **view** cluster allows you to display magnitude results in decibels or not, and the phase results unwrapped or not, in radians or degrees.



FRF mode specifies how to compute the frequency response function (FRF). Available modes are H1, H2, and H3. Default mode is H1.

force/exp settings is a cluster that contains the parameters for the force and exponential windows.

force window [%] specifies the length, expressed in percent of the total length of the signal, of the force window that is used if **window** is set to Force-Exponential.

DBL

exponential window [%] specifies the decay rate of the exponential window that is used if **window** is set to Force-Exponential. This value is the remaining level, expressed in percent, of the window that is applied at the end of the signal.

window is the time-domain window to be used, and can be one of the following values:

0: None (Uniform)

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

9: Force-Exponential

[DBL]

scaled signal X [EUx] contains the scaled signal, expressed in the selected engineering units. This is generally the stimulus or excitation signal in dual-channel measurements.

[DBL]

scaled signal Y [EUy] contains the scaled signal, expressed in the selected engineering units. This is generally the response signal in dual-channel measurements.

DBL

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

view is a cluster that defines how the different results from this VI are returned. Parameters include the ability to use decibels, to unwrap the phase, and to convert the phase to degrees.

TF

dB On (F) specifies if results are expressed in decibels. Default is FALSE.

TF

unwrap phase (F) specifies if the phase has to be unwrapped. Default is TRUE. Unwrapping eliminates discontinuities with an absolute value greater than π.

convert to degree (F) specifies if the phase results have to be converted from radians to degrees. Default is FALSE, which means that results are expressed in radians.

channel X info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel X.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

channel Y info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel Y.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

reset averaging (T) specifies if the selected averaging process has to be reset. Default is TRUE.

averaging parameters is a cluster that defines how the averaging is computed.

averaging mode specifies the averaging mode. Available modes are No averaging, RMS averaging (default), and Vector averaging.

weighting mode specifies the weighting mode. Available modes are Linear and Exponential.

number of averages specifies the number of averages that is used if the **weighting mode** is Exponential. If the **weighting mode** is Linear, this parameter is ignored.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

magnitude [EUy/EUx rms] returns the magnitude of the frequency response.

unit label (magnitude) returns a string that corresponds to the selected magnitude units.

phase returns the phase of the frequency response.

unit label (phase) returns a string that corresponds to the selected phase units (radian or degree).

coherence returns the coherence.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

status is TRUE if an error occurred.

code is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. Refer to Appendix A, *Error Codes*, for a code description.

source shows where an error occurred. The source string is usually the name of the VI that produced the error.

averages completed returns the number of averages completed so far.

window constants contains the following important constants for the selected window:

eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

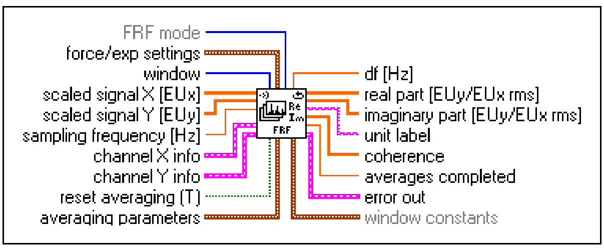
coherent gain is the inverse of the scaling factor applied due to the window.

Frequency Response (Real-Im)

Computes the frequency response and the coherence based on the input signals. Typically, X signal is the stimulus, and Y signal is the response of the system. Results are returned as real and imaginary parts.

Frequency response is typically an averaged measurement. Each block of data corresponds to a single FFT block and must be passed individually to this VI. The first time this VI is called, **reset averaging** must be TRUE (default value), to start a new averaging process. For subsequent calls, **reset averaging** must be FALSE to continue the averaging process. **Averages completed** returns the number of averages completed so far.

Averaging parameters specifies how the averaging is performed. This includes the averaging mode (no averaging, vector averaging, or RMS averaging), the weighting mode (linear or exponential), and the number of averages (used only if the weighting mode is exponential).



FRF mode specifies how to compute the frequency response function (FRF). Available modes are H1, H2, and H3. Default mode is H1.

force/exp settings is a cluster that contains the parameters for the force and exponential windows.

force window [%] specifies the length, expressed in percent of the total length of the signal, of the force window that is used if **window** is set to Force-Exponential.

exponential window [%] specifies the decay rate of the exponential window that is used if **window** is set to Force-Exponential. This value is the remaining level, expressed in percent, of the window that is applied at the end of the signal.

window is the time-domain window to be used, and can be one of the following values:

0: None (Uniform)

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

9: Force-Exponential

scaled signal X [EUx] contains the scaled signal, expressed in the selected engineering units. This is generally the stimulus or excitation signal in dual-channel measurements.

scaled signal Y [EUy] contains the scaled signal, expressed in the selected engineering units. This is generally the response signal in dual-channel measurements.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

channel X info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel X.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

channel Y info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel Y.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

reset averaging (T) specifies if the selected averaging process has to be reset. Default is TRUE.

averaging parameters is a cluster that defines how the averaging is computed.

averaging mode specifies the averaging mode. Available modes are No averaging, RMS averaging (default), and Vector averaging.

weighting mode specifies the weighting mode. Available modes are Linear and Exponential.

number of averages specifies the number of averages that is used if the **weighting mode** is Exponential. If **weighting mode** is Linear, this parameter is ignored.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

real part [EUy/EUx rms] returns the real part of the frequency response.

imaginary part [EUy/EUx rms] returns the imaginary part of the frequency response.

unit label returns a string that contains the selected engineering units.

coherence returns the coherence.

averages completed returns the number of averages completed so far.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

status is TRUE if an error occurred.

code is the error code number identifying an error. A value of 0 means no error, a negative value means an error, and a positive value is a warning. Refer to Appendix A, *Error Codes*, for a code description.

source shows where an error occurred. The source string is usually the name of the VI that produced the error.

window constants contains the following important constants for the selected window:

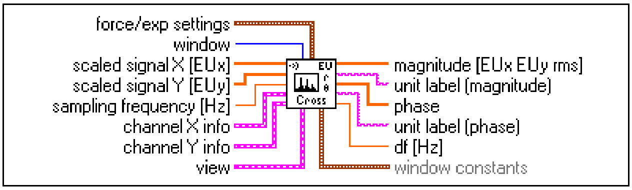
eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

coherent gain is the inverse of the scaling factor applied due to the window.

SV Cross Spectrum (Mag-Phase)

Computes the cross power spectrum of the input signals. Typically, X signal is the stimulus, and Y signal is the response of the system. Results are returned as magnitude and phase.

The **view** cluster allows you to display magnitude results in decibels, and the phase results unwrapped or not, in radians or degrees.



force/exp settings is a cluster that contains the parameters for the force and exponential windows.

force window [%] specifies the length, expressed in percent of the total length of the signal, of the force window that is used if **window** is set to Force-Exponential.

exponential window [%] specifies the decay rate of the exponential window that is used if **window** is set to Force-Exponential. This value is the remaining level, expressed in percent, of the window that is applied at the end of the signal.

window is the time-domain window to be used, and can be one of the following values:

0: None (Uniform)

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

9: Force-Exponential

scaled signal X [EUx] contains the scaled signal, expressed in the selected engineering units. This is generally the stimulus or excitation signal in dual-channel measurements.

scaled signal Y [EUy] contains the scaled signal, expressed in the selected engineering units. This is generally the response signal in dual-channel measurements.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

channel X info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel X.

> **sensor sensitivity [mV/EU]** contains the sensitivity of the sensor, expressed in mV/EU.
>
> **engineering units** is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.
>
> **dB reference [EU]** is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.
>
> **custom label** contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.
>
> **weighting filter** indicates if any weighting filter (A, B, or C) has been applied to the signal.
>
> **pregain [dB]** corresponds to the pregain, expressed in decibels, applied by an external amplifier.

channel Y info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel Y.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

view is a cluster that defines how the different results from this VI are returned. Parameters include the ability to use decibels, to unwrap the phase, and to convert the phase to degrees.

dB On (F) specifies if results are expressed in decibels. Default is FALSE.

unwrap phase (F) specifies if the phase has to be unwrapped. Default is TRUE. Unwrapping eliminates discontinuities with an absolute value greater than π.

convert to degree (F) specifies if the phase results have to be converted from radians to degrees. Default is FALSE, which means that results are expressed in radians.

magnitude [EUx EUy rms] returns the magnitude of the cross power spectrum.

unit label (magnitude) returns a string that corresponds to the selected magnitude units.

phase returns the phase of the cross power spectrum.

unit label (phase) returns a string that corresponds to the selected phase units (radian or degree).

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

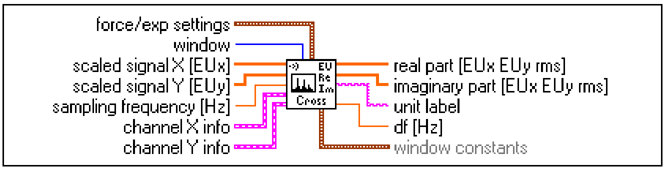
window constants contains the following important constants for the selected window:

eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

coherent gain is the inverse of the scaling factor applied due to the window.

SV Cross Spectrum (Real-Im)

Computes the cross power spectrum of the input signals. Typically, X signal is the stimulus, and Y signal is the response of the system. Results are returned as real and imaginary parts.



force/exp settings is a cluster that contains the parameters for the force and exponential windows.

force window [%] specifies the length, expressed in percent of the total length of the signal, of the force window that is used if **window** is set to Force-Exponential.

exponential window [%] specifies the decay rate of the exponential window that is used if **window** is set to Force-Exponential. This value is the remaining level, expressed in percent, of the window that is applied at the end of the signal.

window is the time-domain window to be used, and can be one of the following values:

0: None (Uniform)

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

9: Force-Exponential

scaled signal X [EUx] contains the scaled signal, expressed in the selected engineering units. This is generally the stimulus or excitation signal in dual-channel measurements.

scaled signal Y [EUy] contains the scaled signal, expressed in the selected engineering units. This is generally the response signal in dual-channel measurements.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

channel X info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel X.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

channel Y info contains the information relative to the measurement system used before the signal reaches the DAQ board for channel Y.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

real part [EUx EUy rms] returns the real part of the cross power spectrum.

imaginary part [EUx EUy rms] returns the imaginary part of the cross power spectrum.

unit label returns a string containing the selected engineering units.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

window constants contains the following important constants for the selected window:

eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

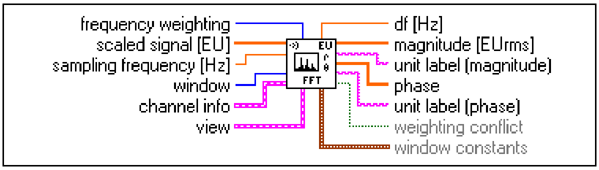
coherent gain is the inverse of the scaling factor applied due to the window.

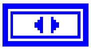
SV FFT Spectrum (Mag-Phase)

Computes the FFT spectrum of the input signal. FFT results are returned as magnitude and phase.

The **view** cluster allows you to display magnitude results in decibels, and the phase results unwrapped or not, in radians or degrees.

The selected **frequency weighting** (A, B, or C) is applied to the results in the frequency domain. There is no time-domain filtering operation.

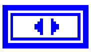




frequency weighting specifies the weighting (A, B, or C) to apply to the signal. This frequency weighting is applied to the spectrum, that is, in the frequency-domain. Use Linear if no weighting is needed.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units. This is generally the stimulus or excitation signal in dual-channel measurements.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.



window is the time-domain window to be used, and can be one of the following values:

0: Uniform

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

view is a cluster that defines how the different results from this VI are returned. Parameters include the ability to use decibels, to unwrap the phase, and to convert the phase to degrees.

dB On (F) specifies if results are expressed in decibels. Default is FALSE.

unwrap phase (F) specifies if the phase has to be unwrapped (TRUE) or not (FALSE, default). Unwrapping eliminates discontinuities with an absolute value greater than π.

convert to degree (F) specifies if the phase results have to be converted from radians to degrees. Default is FALSE, which means that results are expressed in radians.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

magnitude [EUrms] returns the magnitude of the FFT spectrum.

unit label (magnitude) returns a string that corresponds to the selected magnitude units.

phase returns the phase of the FFT spectrum.

unit label (phase) returns a string that corresponds to the selected phase units (radian or degree).

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and **frequency weighting** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

window constants contains the following important constants for the selected window:

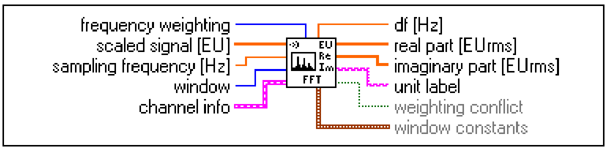
eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

coherent gain is the inverse of the scaling factor applied due to the window.

SV FFT Spectrum (Real-Im)

Computes the FFT spectrum of the input signal. FFT results are returned as real and imaginary parts.

The selected **frequency weighting** (A, B, or C) is applied to the results in the frequency domain. There is no time-domain filtering operation.



frequency weighting specifies the weighting (A, B, or C) to apply to the signal. This frequency weighting is applied to the spectrum, that is, in the frequency-domain. Use Linear if no weighting is needed.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

window is the time-domain window to be used, and can be one of the following values:

0: Uniform

1: Hanning

2: Hamming

3: Blackman-Harris

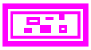
4: Exact Blackman

5: Blackman

6: Flat Top

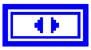
7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris



channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

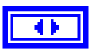
sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.



engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.



weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

real part [EUrms] returns the real part of the FFT spectrum.

imaginary part [EUrms] returns the imaginary part of the FFT spectrum.

unit label returns a string containing the selected engineering units.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and **frequency weighting** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

window constants contains the following important constants for the selected window:

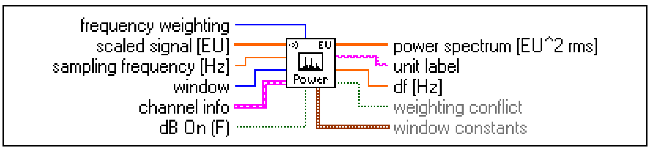
eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

coherent gain is the inverse of the scaling factor applied due to the window.

SV Power Spectrum

Computes the power spectrum of the input signal.

The selected **frequency weighting** (A, B, or C) is applied to the results in the frequency domain. There is no time-domain filtering operation.



frequency weighting specifies the weighting (A, B, or C) to apply to the signal. This frequency weighting is applied to the spectrum, that is, in the frequency-domain. Use Linear if no weighting is needed.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

window is the time-domain window to be used, and can be one of the following values:

0: Uniform

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

dB On (F) specifies if results are expressed in decibels. Default is FALSE.

power spectrum [EU^2 rms] returns the power spectrum.

unit label returns a string containing the selected engineering units.

df [Hz] is the frequency resolution of the spectrum, expressed in hertz.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and **frequency weighting** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

window constants contains the following important constants for the selected window:

eq noise BW is the equivalent noise bandwidth of the selected window. To compute the power in a given frequency span, divide a sum of individual FFT lines by this value.

coherent gain is the inverse of the scaling factor applied due to the window.

6 Transient Analysis

This chapter describes the Transient Analysis VIs. Figure 6-1 shows the **Functions»Sound and Vibration Toolset»Transient Analysis** palette.

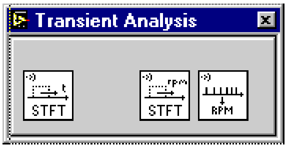


Figure 6-1. Transient Analysis Palette

Introduction

The Transient Analysis VIs are dedicated to the analysis of non-stationary signals.

These VIs use the STFT and can be applied directly to extract frequency information as a function of time. In the case of rotating machines, assuming that a tachometer signal is simultaneously acquired, these VIs can extract frequency information as a function of the rotational speed.

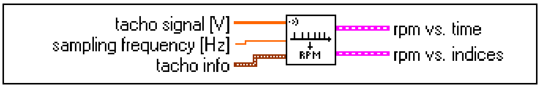
The results the STFT generates are typically displayed on a colormap or on a waterfall display. The Transient Analysis VIs return the scale information needed to properly scale the axes of these displays by using attribute nodes (colormap) or by directly passing that information to the selected display VI (waterfall).

Convert to RPM (analog)

Converts the signal acquired by a tachometer (pulse train) into a rotational speed, expressed in revolutions per minute (rpm), versus time curve.

The **tacho info** cluster specifies the characteristics of the pulses generated by the tachometer (threshold and pulse width), the number of pulses per revolution, and the number of pulses used to compute the rotational speed.

This VI returns the rotational speed as a function of indices, which allows you to perform additional analysis at any specific rotational speed.



DBL **tacho signal [V]** contains the analog tachometer signal, expressed in volts.

DBL **sampling frequency [Hz]** specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

tacho info is a cluster that contains the information relative to the tachometer.

- DBL **threshold [V]** is the value, expressed in volts, used to define the transition between low and high states in the pulse train the tachometer generates.
- I32 **pulse width (2)** specifies the minimum number of samples required to consider the high state as a valid pulse from the tachometer. Default value is 2.
- I32 **avg on n pulses (2)** specifies the number of pulses used to compute the speed of the rotating machine. Default value is 2.
- I32 **pulse/revolution (1)** specifies the number of pulses per revolution the tachometer generates. Default value is 1.

rpm vs. time returns a cluster that contains the rotational speed versus time.

- DBL **time [s]** is the time, expressed in seconds.
- DBL **speed [rpm]** is the rotational speed, expressed in revolutions per minute.

rpm vs. indices returns a cluster that contains the rotational speed versus the indices.

- DBL **indices** is an array that contains the indices corresponding to the rotational speed.
- DBL **speed [rpm]** is the rotational speed, expressed in revolutions per minute.

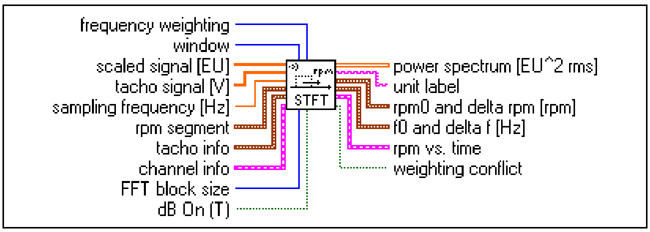
STFT vs RPM (analog)

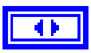
Performs an STFT on the input signal as a function of the rotational speed recorded by **tacho signal**.

The **rpm segment** cluster specifies the revolutions per minute range to analyze (low and high revolutions per minute) and the revolutions per minute increment.

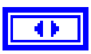
The **tacho info** cluster specifies the characteristics of the pulses generated by the tachometer (threshold and pulse width), the number of pulses per revolution, and the number of pulses used to compute the rotational speed.

This VI returns a 2D array that contains the **power spectrum** at the various selected rotational speeds. The VI returns evolution of the rotational speed (revolutions per minute) as a function of time and the information (**rpm0 and delta rpm**, **f0 and delta f**) needed to display the results on a colormap or a waterfall display.





frequency weighting specifies the weighting (A, B, or C) to apply to the signal. This frequency weighting is applied to the spectrum in the frequency domain. Use Linear if no weighting is needed.



window is the time-domain window to be used, and can be one of the following values:

0: Uniform

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

tacho signal [V] contains the analog tachometer signal, expressed in volts.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

rpm segment specifies which segment of the rotational speed to analyze.

- **from [rpm]** specifies the initial value for the rotational speed segment, expressed in rpm.
- **to [rpm]** specifies the final value for the rotational speed segment, expressed in rpm.
- **rpm increment [rpm]** specifies the increment of rotational speed, expressed in rpm.

tacho info is a cluster that contains the information relative to the tachometer.

- **threshold [V]** is the value, expressed in volts, used to define the transition between low and high states in the pulse train the tachometer generates.
- **pulse width (2)** specifies the minimum number of samples required to consider the high state as a valid pulse from the tachometer. Default value is 2.
- **avg on n pulses (2)** specifies the number of pulses used to compute the speed of the rotating machine. Default value is 2.
- **pulse/revolution (1)** specifies the number of pulses per revolution the tachometer generates. Default value is 1.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

- **sensor sensitivity [mV/EU]** contains the sensitivity of the sensor, expressed in mV/EU.
- **engineering units** is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in dB. The typical dB reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in dB, applied by an external amplifier, if any.

FFT block size specifies the number of samples used to compute one FFT.

dB On (T) specifies if results are expressed in dB. Default is TRUE.

power spectrum [EU^2 rms] returns the power spectrum.

unit label returns a string that contains the selected engineering units.

rpm0 and delta rpm [rpm] is a cluster that specifies the minimum value of the rotational speed (rpm0) and the increment (delta revolutions per minute).

rpm0 specifies the minimum value of the rotational speed.

delta rpm specifies the rotational speed increment, expressed in rpm.

f0 and delta f [Hz] is a cluster that specifies the minimum value of the frequency (f0) and the increment (delta f).

f0 specifies the minimum value of the frequency.

delta f specifies the frequency resolution, expressed in hertz.

rpm vs. time returns a cluster that contains the rotational speed versus time.

time [s] is the time, expressed in seconds.

speed [rpm] is the rotational speed, expressed in rpm.

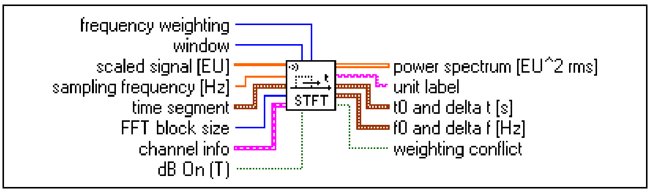
weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and **frequency weighting** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

STFT vs Time

Performs an STFT on the input signal.

The **time segment** cluster specifies the time segment to analyze and the time increment and its units (%, s, or ms).

This VI returns a 2D array that contains the **power spectrum** at various selected times. The VI also returns the information (**t0 and delta t**, **f0 and delta f**) needed to display the results on a colormap or waterfall display.



frequency weighting specifies the weighting (A, B, or C) to apply to the signal. This frequency weighting is applied to the spectrum in the frequency domain. Use Linear if no weighting is needed.

window is the time-domain window to be used, and can be one of the following values:

0: Uniform

1: Hanning

2: Hamming

3: Blackman-Harris

4: Exact Blackman

5: Blackman

6: Flat Top

7: Four Term Blackman-Harris

8: Seven Term Blackman-Harris

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

time segment specifies which segment of the time-domain signal to analyze.

from [s] specifies the initial value for the time segment, expressed in seconds.

to [s] specifies the final value for the time segment, expressed in seconds.

time increment specifies the time increment between two consecutive FFT blocks.

time increment units (%) sets the units used to specify the **time increment**. Default value is percentage, with 100 percent corresponding to no overlapping.

FFT block size specifies the number of samples used to compute one FFT.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in dB. The typical dB reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in dB, applied by an external amplifier.

dB On (T) specifies if results are expressed in dB. Default is TRUE.

power spectrum [EU^2 rms] returns the power spectrum.

unit label returns a string that contains the selected engineering units.

t0 and delta t [s] is a cluster that specifies the minimum value of the time segment (t0) and the increment (delta t).

t0 specifies the minimum value of the time segment, expressed in seconds.

delta t specifies the time increment, expressed in seconds.

f0 and delta f [Hz] is a cluster that specifies the minimum value of the frequency (f0) and the increment (delta f).

f0 specifies the minimum value of the frequency, expressed in hertz.

delta f specifies the frequency resolution, expressed in hertz.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and **frequency weighting** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

7

Sound Level Measurement

This chapter describes the Sound Level Measurement VIs. Figure 7-1 shows the **Functions»Sound and Vibration Toolset»Sound Level Measurement** palette.

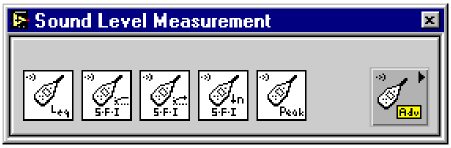


Figure 7-1. Sound Level Measurement Palette

Introduction

These VIs offer typical sound level measurements, including the following time averaging modes:

- Linear (equivalent continuous sound level, or Leq)
- Exponential
- Peak hold

Linear averaging is computed by integrating the square of the signal over a fixed time interval and dividing by the time interval.

The exponential averaging mode supports any custom time constant and the following three standard time constants:

- Slow (1,000 ms)
- Fast (125 ms)
- Impulse (35 ms if the signal is rising and 1,500 ms if the signal is falling)

Peak hold returns the maximum value of the instantaneous sound level. Peak mode literally is not an actual *averaging* mode.

In the case of linear averaging and peak hold, the associated VI returns a single value that corresponds to the sound level at the end of the time record. To obtain intermediate results, you must split a long time record into several smaller records.

Three VIs are available for exponential time averaging. The first one, Exp Avg Sound Level VI, returns the exponential average sound level computed after each individual sample of the time record. The VI returns the sound level as an array that has the same size as the original time record.

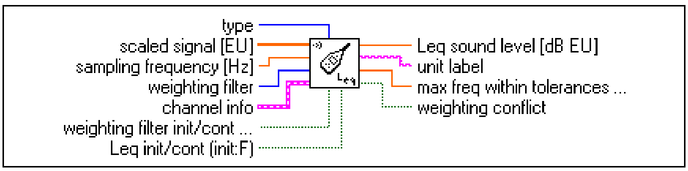
The Exp Avg Sound Level (final value) VI is similar to the linear averaging or peak hold VIs and returns a single value that corresponds to the sound level at the end of the time record.

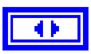
The Exp Avg Sound Level (decimated) VI is a compromise between these two techniques. It allows you to decimate the result of the exponential average sound level. Choose a decimating factor to generate a new sampling frequency in accordance with the Shannon Sampling Theorem to avoid aliasing problems. This means that the new time period after decimation must not exceed half the time constant used by the exponential averaging filter.

In addition to these averaging modes, these VIs also offer A, B, or C weighting filters to obtain directly sound level measurements in dB(A), dB(B), or dB(C).

Equivalent Continuous Sound Level

Computes the equivalent continuous sound level of the input signal.





type specifies the weighting network filter type according to the ANSI S1.4-1983 standard to use to determine the frequency range within the tolerance limits.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

weighting filter specifies the weighting filter (A, B, or C) to apply to the signal. Use Linear if no weighting is needed.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical dB reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

weighting filter init/cont (init:F) controls the initialization of the weighting filter. When **weighting filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **weighting filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

Leq init/cont (init:F) controls the initialization of the equivalent continuous measurement. To analyze a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous processing of all remaining blocks.

Leq sound level [dB EU] returns the equivalent continuous sound level (Leq), expressed in decibels.

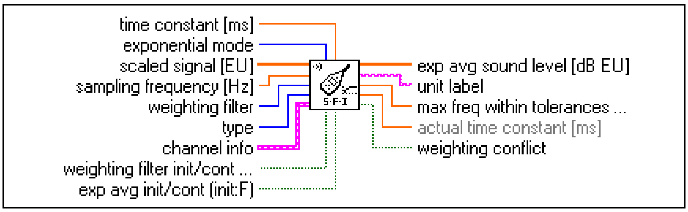
unit label returns a string that contains the selected engineering units.

max freq within tolerances [Hz] returns the maximum frequency, expressed in hertz, that is within the tolerance limits based on the selected **type**. Tolerance limits are based on the ANSI S1.4-1983 standard.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and the selected **type** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

Exp Avg Sound Level

Computes the exponential averaged sound level of the input signal X, based on the selected exponential averaging mode, and returns this sound level for the entire signal, expressed in decibels.



time constant [ms] specifies the time constant used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

weighting filter specifies the weighting filter (A, B, or C) to apply to the signal. Use Linear if no weighting is needed.

type specifies the weighting network filter type according to the ANSI S1.4-1983 standard to use to determine the frequency range within the tolerance limits.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical dB reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

weighting filter init/cont (init:F) controls the initialization of the weighting filter. When **weighting filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **weighting filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

exp avg init/cont (init:F) controls the initialization of the exponential averaging filter. When **exp avg init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **exp avg init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

exp avg sound level [dB EU] returns the exponential averaged sound level, expressed in decibels.

unit label returns a string that contains the selected engineering units.

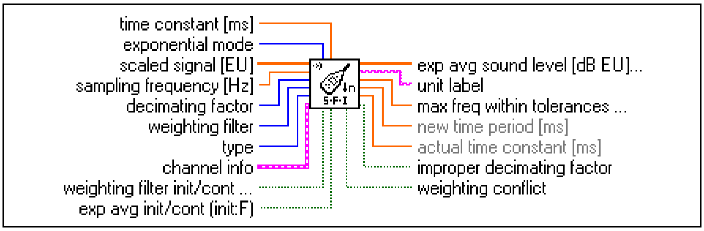
max freq within tolerances [Hz] returns the maximum frequency, expressed in hertz, that is within the tolerance limits based on the selected **type**. Tolerance limits are based on the ANSI S1.4-1983 standard.

actual time constant [ms] returns the time constant actually used for the selected exponential averaging mode. Slow mode returns 1,000 ms. Fast mode returns 125 ms. Impulse mode returns 35 ms.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and the selected **type** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

Exp Avg Sound Level (decimated)

Computes the exponential averaged sound level of the input signal X, based on the selected exponential averaging mode, and decimates this sound level by the selected decimating factor. Choose a decimating factor to generate a new sampling frequency in accordance with the Shannon Sampling Theorem to avoid aliasing problems. This means that the **new time period** must not exceed half the actual time constant.



time constant [ms] specifies the time constant used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

decimating factor specifies the decimating factor used for the decimation of the exponential averaged sound level.

weighting filter specifies the weighting filter (A, B, or C) to apply to the signal. Use Linear if no weighting is needed.

type specifies the weighting network filter type according to the ANSI S1.4-1983 standard to use to determine the frequency range within the tolerance limits.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

- **sensor sensitivity [mV/EU]** contains the sensitivity of the sensor, expressed in mV/EU.

- **engineering units** is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

- **dB reference [EU]** is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical dB reference for sound pressure level is 20E-6 Pa.

- **custom label** contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

- **weighting filter** indicates if any weighting filter (A, B, or C) has been applied to the signal.

- **pregain [dB]** corresponds to the pregain, expressed in decibels, applied by an external amplifier.

weighting filter init/cont (init:F) controls the initialization of the weighting filter. When **weighting filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **weighting filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

exp avg init/cont (init:F) controls the initialization of the exponential averaging filter. When **exp avg init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **exp avg init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

exp avg sound level [dB EU] (decimated) returns the decimated exponential averaged sound level, expressed in decibels.

unit label returns a string that contains the selected engineering units.

max freq within tolerances [Hz] returns the maximum frequency, expressed in hertz, that is within the tolerance limits based on the selected **type**. Tolerance limits are based on the ANSI S1.4-1983 standard.

new time period [ms] returns the new time period that results from the decimation process. Choose a **decimating factor** to generate a new sampling frequency in accordance with the Shannon Sampling Theorem to avoid aliasing problems. This means that the new time period must not exceed half the **actual time constant**.

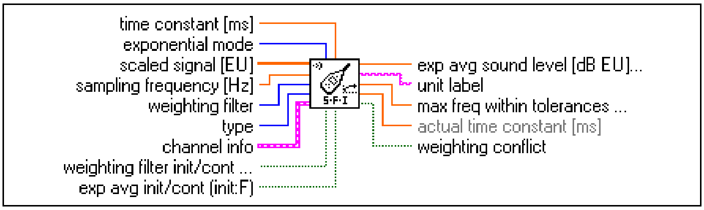
actual time constant [ms] returns the time constant actually used for the selected exponential averaging mode. Slow mode returns 1,000 ms. Fast mode returns 125 ms. Impulse mode returns 35 ms.

improper decimating factor returns TRUE if the selected **decimating factor** generates a new sampling frequency that is not in accordance with the Shannon Sampling Theorem and can potentially cause aliasing problems.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and the selected **type** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

Exp Avg Sound Level (final value)

Computes the exponential averaged sound level of the input signal X, based on the selected exponential averaging mode, and returns the last value of this sound level, expressed in decibels.



DBL **time constant [ms]** specifies the time constant used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

[DBL] **scaled signal [EU]** contains the scaled signal, expressed in the selected engineering units.

DBL **sampling frequency [Hz]** specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

weighting filter specifies the weighting filter (A, B, or C) to apply to the signal. Use Linear if no weighting is needed.

type specifies the weighting network filter type according to the ANSI S1.4-1983 standard to use to determine the frequency range within the tolerance limits.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

DBL **sensor sensitivity [mV/EU]** contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical dB reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

weighting filter init/cont (init:F) controls the initialization of the weighting filter. When **weighting filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **weighting filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

exp avg init/cont (init:F) controls the initialization of the exponential averaging filter. When **exp avg init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **exp avg init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

exp avg sound level [dB EU] (final value) returns the last value of the exponential averaged sound level, expressed in decibels.

unit label returns a string that contains the selected engineering units.

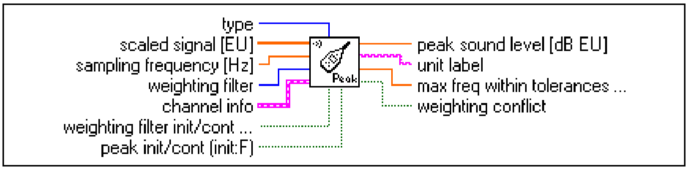
max freq within tolerances [Hz] returns the maximum frequency, expressed in hertz, that is within the tolerance limits based on the selected **type**. Tolerance limits are based on the ANSI S1.4-1983 standard.

actual time constant [ms] returns the time constant actually used for the selected exponential averaging mode. Slow mode returns 1,000 ms. Fast mode returns 125 ms. Impulse mode returns 35 ms.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and the selected **type** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

Peak Sound Level

Computes the peak sound level of the input signal.



type specifies the weighting network filter type according to the ANSI S1.4-1983 standard to use to determine the frequency range within the tolerance limits.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

weighting filter specifies the weighting filter (A, B, or C) to apply to the signal. Use Linear if no weighting is needed.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical dB reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

weighting filter init/cont (init:F) controls the initialization of the weighting filter. When **weighting filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **weighting filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

peak init/cont (init:F) controls the initialization of the peak detection. To analyze a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous peak detection on all remaining blocks.

peak sound level [dB EU] returns the peak sound level, expressed in decibels.

unit label returns a string that contains the selected engineering units.

max freq within tolerances [Hz] returns the maximum frequency, expressed in hertz, that is within the tolerance limits based on the selected **type**. Tolerance limits are based on the ANSI S1.4-1983 standard.

weighting conflict returns TRUE if **channel info** shows that a weighting filter has already been applied to the signal and the selected **type** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

Sound Level Measurement—Advanced Subpalette

This section describes the Advanced Sound Level Measurement VIs located in the **Functions»Sound and Vibration Toolset»Sound Level Measurement»Advanced** subpalette, as shown in Figure 7-2. These are low-level VIs designed for the advanced user.

The **Advanced** subpalette provides the basic tools required for the various averaging modes previously described. You can use these VIs to perform the same kind of averaging on any signal, not necessarily those acquired from a microphone. A typical example is to compare the results of an exponentially averaged sound level measurement with the level an accelerometer measures, using the same time constant for the exponential averaging on the signal.

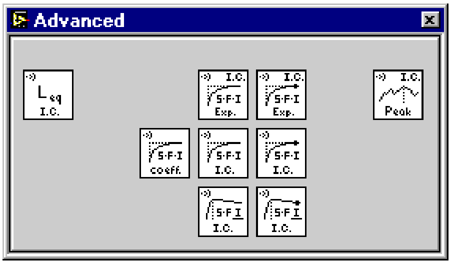
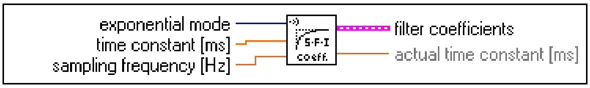
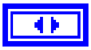


Figure 7-2. Sound Level Measurement Advanced Subpalette

Coefficients for Exponential Averaging

Generates the filter coefficients that correspond to the selected **exponential mode** or custom **time constant**.





exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

time constant [ms] specifies the time constant used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

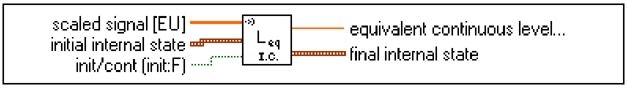
filter coefficients returns a cluster that contains the filter coefficients designed to perform the selected exponential averaging mode.

actual time constant [ms] returns the time constant actually used for the selected exponential averaging mode. Slow mode returns 1,000 ms. Fast mode returns 125 ms. Impulse mode returns 35 ms.

Equivalent Continuous Level with IC

Computes the equivalent continuous level of the input signal.

The **initial internal state** and **final internal state** keep track of the internal state of the averaging process to ensure continuous operation if you use continuous mode.



scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

initial internal state specifies the initial internal state.

squared sum keeps track of the current squared sum.

size keeps track of the total number of samples averaged.

init/cont (init:F) controls the initialization of the internal states. When **init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **init/cont (init:F)** is TRUE, the internal states are initialized to the final states from the previous call to this instance of this VI. To average a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous averaging of all remaining blocks.

equivalent continuous level [EU^2 rms] returns the equivalent continuous level of the input signal.

final internal state returns the final internal state.

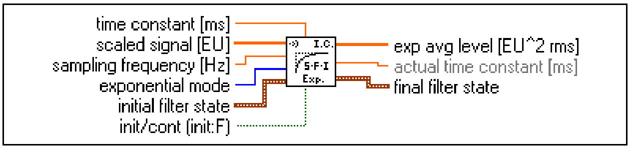
squared sum keeps track of the current squared sum.

size keeps track of the total number of samples averaged.

Exp Avg Level with IC

Computes the exponential averaged level of the input signal X based on the selected exponential averaging mode and returns this level for the entire signal.

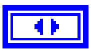
The **initial filter state** and the **final filter state** parameters keep track of the internal state of the filter to ensure continuous operation of the filter if continuous mode is used.



time constant [ms] specifies the time constant used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.



exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), Impulse, and Custom. The Custom mode allows you to specify any **time constant**.

initial filter state specifies the initial internal state of the filter.

exponential averaging keeps track of the internal state of the exponential averaging filter.

peak detector keeps track of the internal state of the peak detection operation.

init/cont (init:F) controls the initialization of the internal filter states. When **init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block, and to TRUE for continuous filtering of all remaining blocks.

exp avg level [EU^2 rms] returns the exponential averaged level.

actual time constant [ms] returns the time constant actually used for the selected exponential averaging mode. Slow mode returns 1,000 ms. Fast mode returns 125 ms. Impulse mode returns 35 ms.

final filter state contains the final internal filter state. To filter samples continuously, you have to pass **final filter state** as the **initial filter state** when the next call to this VI is performed.

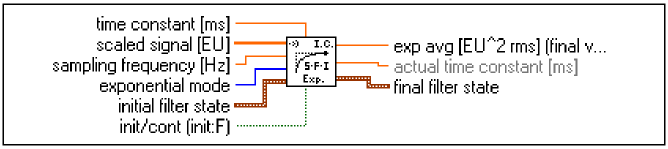
exponential averaging keeps track of the internal state of the exponential averaging filter.

peak detector keeps track of the internal state of the peak detection operation.

Exp Avg Level with IC (final value)

Computes the exponential averaged level of the input signal X based on the selected exponential averaging mode and returns only the last value of this level.

The **initial filter state** and the **final filter state** parameters keep track of the internal state of the filter to ensure continuous operation of the filter if continuous mode is used.



time constant [ms] specifies the time constant used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), Impulse, and Custom. The Custom mode allows you to specify any **time constant**.

initial filter state specifies the initial internal state of the filter.

- **exponential averaging** keeps track of the internal state of the exponential averaging filter.
- **peak detector** keeps track of the internal state of the peak detection operation.

init/cont (init:F) controls the initialization of the internal filter states. When **init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block, and to TRUE for continuous filtering of all remaining blocks.

exp avg [EU^2 rms] (final value) returns the last value of the exponential averaged level.

actual time constant [ms] returns the time constant actually used for the selected exponential averaging mode. Slow mode returns 1,000 ms. Fast mode returns 125 ms. Impulse mode returns 35 ms.

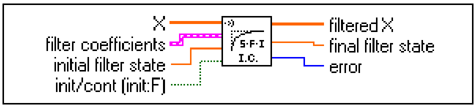
final filter state contains the final internal filter state. To filter samples continuously, you have to pass **final filter state** as the **initial filter state** when the next call to this VI is performed.

- **exponential averaging** keeps track of the internal state of the exponential averaging filter.
- **peak detector** keeps track of the internal state of the peak detection operation.

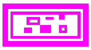
Exponential Filter with IC

Applies a single time constant exponential averaging on the input signal X based on the **filter coefficients** and returns the entire filtered signal.

The **initial filter state** and the **final filter state** parameters keep track of the internal state of the filter to ensure continuous operation of the filter if continuous mode is used.



X is the input array of samples to filter.



filter coefficients is a cluster that contains the filter coefficients designed to perform the selected exponential averaging mode.

initial filter state specifies the initial internal state of the filter.

init/cont (init:F) controls the initialization of the internal filter states. When **init/cont (init:F)** is FALSE (default), initialize the internal states to zero. When **init/cont (init:F)** is TRUE, initialize the internal filter states to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

filtered X returns the output array of filtered samples.

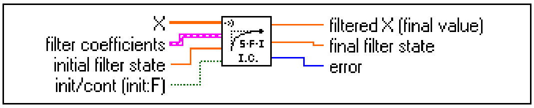
final filter state contains the final internal filter state. To filter samples continuously, you have to pass **final filter state** as the **initial filter state** when the next call to this VI is performed.

error allows you to wire this output to the Find First Error VI to produce an error cluster. You can then wire this cluster to the Simple Error Handler VI or the General Error Handler VI for an immediate report on any errors. Select **Help»Online Reference** to find the error codes LabVIEW returns.

Exponential Filter with IC (final value)

Applies a single time constant exponential averaging on the input signal X based on the **filter coefficients**, and returns only the last value of the filtered signal.

The **initial filter state** and the **final filter state** parameters keep track of the internal state of the filter to ensure continuous operation of the filter if continuous mode is used.



DBL **X** is the input array of samples to filter.

filter coefficients is a cluster that contains the filter coefficients designed to perform the selected exponential averaging mode.

DBL **initial filter state** specifies the initial internal state of the filter.

TF **init/cont (init:F)** controls the initialization of the internal filter states. When **init/cont (init:F)** is FALSE (default), initialize the internal states to zero. When **init/cont (init:F)** is TRUE, initialize the internal filter states to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

DBL **filtered X (final value)** returns the last filtered sample.

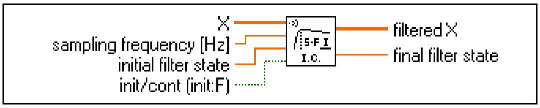
DBL **final filter state** contains the final internal filter state. To filter samples continuously, you have to pass **final filter state** as the **initial filter state** when the next call to this VI is performed.

I32 **error** allows you to wire this output to the Find First Error VI to produce an error cluster. You can then wire this cluster to the Simple Error Handler VI or the General Error Handler VI for an immediate report on any errors. Select **Help»Online Reference** to find the error codes LabVIEW returns.

Impulse Peak Detector with IC

Detects peaks with a decay time constant of 1,500 ms on the input signal X and returns the entire filtered signal.

The **initial filter state** and the **final filter state** parameters keep track of the internal state of the filter to ensure continuous operation of the filter if continuous mode is used.



X is the input array of samples to filter.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

initial filter state specifies the initial internal state of the filter.

init/cont (init:F) controls the initialization of the internal filter states. When **init/cont (init:F)** is FALSE (default), initialize the internal states to zero. When **init/cont (init:F)** is TRUE, initialize the internal filter states to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

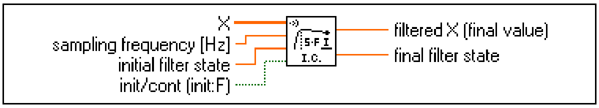
filtered X returns the output array of filtered samples.

final filter state contains the final internal filter state. To filter samples continuously, you have to pass **final filter state** as the **initial filter state** when the next call to this VI is performed.

Impulse Peak Detector with IC (final value)

Detects peaks with a decay time constant of 1,500 ms on the input signal X and returns only the last value of the filtered signal.

The **initial filter state** and the **final filter state** parameters keep track of the internal state of the filter to ensure continuous operation of the filter if continuous mode is used.



X is the input array of samples to filter.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

initial filter state specifies the initial internal state of the filter.

init/cont (init:F) controls the initialization of the internal filter states. When **init/cont (init:F)** is FALSE (default), initialize the internal states to zero. When **init/cont (init:F)** is TRUE, initialize the internal filter states to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

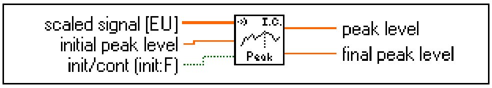
filtered X (final value) returns the last filtered sample.

final filter state contains the final internal filter state. To filter samples continuously, you have to pass **final filter state** as the **initial filter state** when the next call to this VI is performed.

Peak Level with IC

Detects the peak level based on the input signal and the **initial peak level**.

The **initial peak level** and the **final peak level** parameters keep track of the internal state of the peak detection process to ensure continuous operation if continuous mode is used.



scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

initial peak level specifies the initial peak level.

init/cont (init:F) controls the initialization of the peak detection process. When **init/cont (init:F)** is FALSE (default), initialize the internal peak level to zero. When **init/cont (init:F)** is TRUE, initialize the internal filter states to the final peak level from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous peak detection on all remaining blocks.

peak level returns the maximum peak level.

final peak level returns the final peak level.

8

Octave Analysis

This chapter describes the Octave Analysis VIs. Figure 8-1 shows the **Functions»Sound and Vibration Toolset»Octave Analysis** palette.

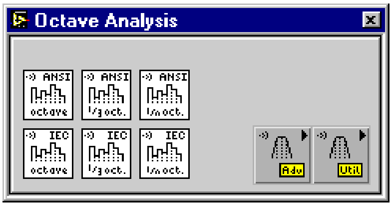


Figure 8-1. Octave Analysis Palette

Introduction

The **Octave Analysis** palette provides specific VIs to perform the following analyses:

- Octave analysis
- Third-octave analysis
- Fractional-octave analysis (1/1, 1/3, 1/6, 1/12, and 1/24 octave)

When combined with one of the following DAQ boards from National Instruments (PCI-445*X* or NI 455*X*), these VIs offer the following compliance:

- ANSI S1.11-1986, Order 3, Type 1-D, optional range
- Class 1, IEC 1260:1995

In the case of the ANSI standard, the default order of the bandpass filters is 3, leading to Type 1-D filter. However, when you use the Design Octave Filter Set VI in the **Advanced** subpalette, you can select the order of these filters to achieve a better Type and Sub-Type.

In addition, these VIs can accommodate any sampling frequency and any number of fractional-octave bands.

For octave filters, the sampling frequency should be at least 3 times the center frequency of the highest frequency band. For all other fractional-octave filters, the sampling frequency should be at least 2.5 times the center frequency of the highest frequency band.

According to the ANSI S1.11-1986 standard, the midband frequency, or the center frequency of the bandpass filter, is defined as follows:

$$f_i = 1000 \bullet 2^{ib}$$

where, f_i is the center frequency of the i^{th} bandpass filter, expressed in hertz
i is an integer, referred to in these VIs as the band selector (when $i = 0, f_0 = 1$ kHz, which is the reference frequency for the audio range)
b is the bandwidth designator ($b = 1$ for octave, b = 1/3 for 1/3 octave, $b = 1/6$ for 1/6 octave, b = 1/12 for 1/12 octave, and b = 1/24 for 1/24 octave).

According to the IEC 1260:1995 standard, the midband frequency, or the center frequency of the bandpass filter, is defined as follows:

$$f_i = 1000 \bullet 2^{ib} \text{ for } 1/n \text{ octave filters when } n \text{ is odd}$$

$$f_i = 1000 \bullet 2^{\frac{(i+1)b}{2}} \text{ for } 1/n \text{ octave filters when } n \text{ is even}$$

where, f_i is the center frequency of the i^{th} bandpass filter, expressed in hertz
i is an integer, referred to in these VIs as the band selector (when $i = 0, f_0 = 1$ kHz, which is the reference frequency for the audio range)
b is the bandwidth designator (b = 1 for octave, b = 1/3 for 1/3 octave, b = 1/6 for 1/6 octave, b = 1/12 for 1/12 octave and b = 1/24 for 1/24 octave)

Note Both standards use the base 2 system for these midband frequencies instead of the base 10 system. This means that the ratio of two midband frequencies is a fractional power of 2, not 10.

These exact midband frequencies are used to design the filters for octave analysis, but all VIs typically return the nominal midband frequencies, also called the preferred frequencies. In the case of octave and 1/3 octave analysis, these nominal frequencies are tabulated in the standards. In the case of 1/6, 1/12, and 1/24 octave analysis, these nominal frequencies are calculated in accordance with the *Annex A (informative)* of the IEC 1260:1995 standard.

All octave analysis VIs support the following averaging types:

- Linear averaging
- Exponential averaging
- Equal confidence averaging
- Peak hold

Each of these four different averaging types uses additional information specified in the **averaging parameters** cluster.

Linear averaging is computed by integrating the square of the signal over a fixed time interval and dividing by the time interval.

Exponential averaging supports any custom time constant and the following three standard time constants:

- Slow (1,000 ms)
- Fast (125 ms)
- Impulse (35 ms if the signal is rising and 1,500 ms if the signal is falling)

Equal confidence averaging is basically exponential averaging with a specific time constant set for each band separately. Each time constant is set so that the standard deviation of the band power measurement equals the confidence value, in decibels, specified in the **averaging parameters** cluster.

Peak hold returns the maximum of the instantaneous band power. Peak hold literally is not an actual *averaging* mode.

When starting or resetting the filtering operation of the octave filters, a certain time is required before the measurements are valid. This time is called the settling time and is defined as 5 divided by the bandwidth of any particular filter. The lowest frequency band has the smallest bandwidth and defines the settling time required before the complete octave measurement can be considered valid. The **settled?** Boolean indicator returns a TRUE value as soon as all the filters are settled.

When you perform linear averaging or peak hold, the items in the **averaging parameters** cluster allow you to specify how the averaging process or the peak detection handles the settling time.

For linear averaging, the following choices are available:

- Start averaging once settled—averaging process starts once all the octave filters are settled. This is the default selection.
- Continuous averaging—averaging process starts immediately, without waiting for the filters to be settled.
- Single averaging—averaging is performed only on the last block of time data (**scaled signal**) sent to this VI. This is equivalent to resetting the averaging process each time this VI is called.

For peak hold, the following choices are available:

- Start peak detection once settled—peak detection starts once all the octave filters are settled. This is the default selection.
- Continuous peak detection—peak detection starts immediately, without waiting for the filters to be settled.
- Single peak detection—peak detection is performed only on the last block of time data (**scaled signal**) sent to this VI. This is equivalent to resetting the averaging process each time this VI is called.

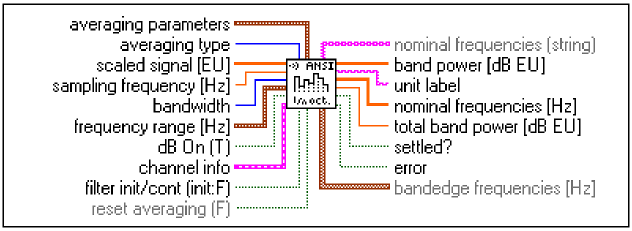
ANSI - Fractional-octave Analysis

Performs a fractional-octave analysis on the input signal. The fractional-octave filters are designed in accordance with ANSI S1.11-1986, Order 3, Type 1-D, optional range. To meet this standard, the sampling frequency must be at least 3 times the highest midband frequency for octave analysis and at least 2.5 times the highest midband frequency for 1/3, 1/6, 1/12, or 1/24 octave analysis.

Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Based on the selected **averaging type**, this VI uses the additional settings in **averaging parameters**.

This VI returns the power in each fractional-octave band and the **total band power**, both expressed in decibels. The total band power is defined as the sum of the individual band powers.

To ensure a continuous filtering operation call this VI the first time with **filter init/cont** set to FALSE and then call it with **filter init/cont** set to TRUE.



averaging parameters is a cluster that defines how the selected **averaging type** is computed.

linear mode specifies how the linear averaging is performed if the **averaging type** is set to Linear. The following linear averaging modes are available:

- Start averaging once settled (default)
- Continuous averaging
- Single averaging

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

time constant [ms] specifies the time constant used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

confidence [dB] specifies the confidence value, expressed in decibels, used if the **averaging type** is set to Equal confidence.

peak mode specifies how the peak detection is performed if the **averaging type** is set to Peak. The following peak detection modes are available:

- Start peak detection once settled (default)
- Continuous peak detection
- Single peak detection

averaging type specifies the type of averaging that is applied to the result of the 1/*n* octave filtering. Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Each type of averaging uses specific parameters defined in the **averaging parameters** cluster.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

bandwidth specifies the bandwidth of the 1/*n* octave filters. Available bandwidths are 1, 1/3, 1/6, 1/12, and 1/24 octaves.

frequency range [Hz] is a cluster that specifies the low and high frequencies, expressed in hertz, of the 1/*n* octave analysis

low freq. specifies the low frequency of the frequency range.

high freq. specifies the high frequency of the frequency range.

dB On (T) specifies if results are expressed in decibels. Default is TRUE.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

TF **reset averaging (F)** specifies if the selected averaging process has to be reset. Default is FALSE.

[abc] **nominal frequencies (string)** is an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band. For 1 octave and 1/3 octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards. For 1/6 octave, 1/12 octave, and 1/24 octave, strings are formatted according to Annex A of the IEC 1260:1995 standard.

[DBL] **band power [dB EU]** returns the power, expressed in decibels, associated with each 1/*n* octave band.

abc **unit label** returns a string containing the selected engineering units.

[DBL] **nominal frequencies [Hz]** is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band.

DBL **total band power [dB EU]** returns the sum of the power contained in all the 1/*n* octave bands.

TF **settled?** returns TRUE when all 1/*n* octave filters are settled.

TF **error** is TRUE if an error is encountered during the design process of the filter.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each 1/*n* octave band. This includes the low and high frequencies and the exact midband frequency.

DBL **f low** contains the low frequency of the 1/*n* octave band.

DBL **f high** contains the high frequency of the 1/*n* octave band.

DBL **midband** contains the exact midband, or center, frequency of the 1/*n* octave band.

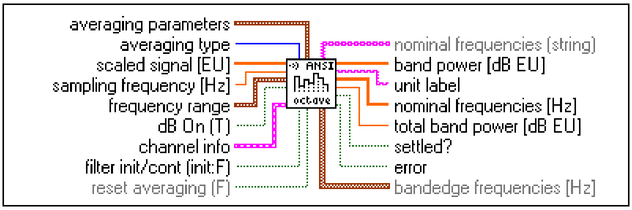
ANSI - Octave Analysis

Performs an octave analysis on the input signal. The octave filters are designed in accordance with ANSI S1.11-1986, Order 3, Type 1-D, optional range. To meet this standard, the sampling frequency must be at least three times the highest midband frequency.

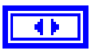
Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Based on the selected **averaging type**, this VI uses the additional settings in **averaging parameters**.

This VI returns the power in each octave band and the **total band power**, both expressed in decibels. The total band power is defined as the sum of the individual band powers.

To ensure a continuous filtering operation call this VI the first time with **filter init/cont** set to FALSE and then call it with **filter init/cont** set to TRUE.

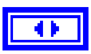


averaging parameters is a cluster that defines how the selected **averaging type** is computed.



linear mode specifies how the linear averaging is performed if the **averaging type** is set to Linear. The following linear averaging modes are available:

- Start averaging once settled (default)
- Continuous averaging
- Single averaging



exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

time constant [ms] specifies the time constant, expressed in ms, used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

confidence [dB] specifies the confidence value, expressed in decibels, used if the **averaging type** is set to Equal confidence.

peak mode specifies how the peak detection is performed if the **averaging type** is set to Peak. The following peak detection modes are available:

- Start peak detection once settled (default)
- Continuous peak detection
- Single peak detection

averaging type specifies the type of averaging that is applied to the result of the octave filtering. Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Each type of averaging uses specific parameters defined in the **averaging parameters** cluster.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

frequency range is a cluster that specifies the low and high frequencies of the octave analysis

low band specifies the center frequency of the lowest octave band.

high band specifies the center frequency of the highest octave band.

dB On (T) specifies if results are expressed in decibels. Default is TRUE.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

filter init/cont (init:F) controls the initialization of the octave filters. When **filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

reset averaging (F) specifies if the selected averaging process has to be reset. Default is FALSE.

nominal frequencies (string) is an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each octave band. For octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards.

band power [dB EU] returns the power, expressed in decibels, associated with each octave band.

unit label returns a string containing the selected engineering units.

nominal frequencies [Hz] is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each octave band.

total band power [dB EU] returns the sum of the power contained in all the octave bands.

settled? returns TRUE when all octave filters are settled.

error is TRUE if an error is encountered during the design process of the filter.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each octave band. This includes the low and high frequencies and the exact midband frequency.

f low contains the low frequency of the octave band.

f high contains the high frequency of the octave band.

midband contains the exact midband or center frequency of the octave band.

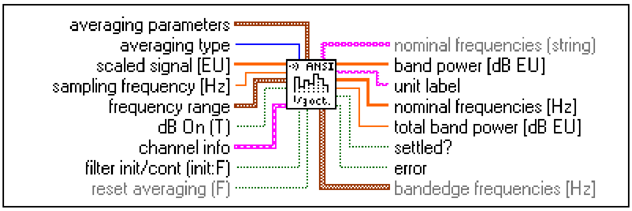
ANSI - Third-octave Analysis

Performs a third-octave analysis on the input signal. The third-octave filters are designed in accordance with ANSI S1.11-1986, Order 3, Type 1-D, optional range. To meet this standard, the sampling frequency must be at least 2.5 times the highest midband frequency.

Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Based on the selected **averaging type**, this VI uses the additional settings in **averaging parameters**.

This VI returns the power in each third-octave band and the **total band power**, both expressed in decibels. The total band power is defined as the sum of the individual band powers.

To ensure a continuous filtering operation call this VI the first time with **filter init/cont** set to FALSE and then call it with **filter init/cont** set to TRUE.



averaging parameters is a cluster that defines how the selected **averaging type** is computed.

linear mode specifies how the linear averaging is performed if the **averaging type** is set to Linear. The following linear averaging modes are available:

- Start averaging once settled (default)
- Continuous averaging
- Single averaging

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

time constant [ms] specifies the time constant, expressed in ms, used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

confidence [dB] specifies the confidence value, expressed in decibels, used if the **averaging type** is set to Equal confidence.

peak mode specifies how the peak detection is performed if the **averaging type** is set to Peak. The following peak detection modes are available:

- Start peak detection once settled (default)
- Continuous peak detection
- Single peak detection

averaging type specifies the type of averaging that is applied to the result of the third-octave filtering. Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Each type of averaging uses specific parameters defined in the **averaging parameters** cluster.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

frequency range is a cluster that specifies the low and high frequencies of the third-octave analysis

low band specifies the center frequency of the lowest third-octave band.

high band specifies the center frequency of the highest third-octave band.

dB On (T) specifies if results are expressed in decibels. Default is TRUE.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

filter init/cont (init:F) controls the initialization of the third-octave filters. When **filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

reset averaging (F) specifies if the selected averaging process has to be reset. Default is FALSE.

nominal frequencies (string) is an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each third-octave band. For third-octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards.

band power [dB EU] returns the power, expressed in decibels, associated with each third-octave band.

unit label returns a string containing the selected engineering units.

nominal frequencies [Hz] is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each third-octave band.

total band power [dB EU] returns the sum of the power contained in all the third-octave bands.

settled? returns TRUE when all third-octave filters are settled.

error is TRUE if an error is encountered during the design process of the filter.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each third-octave band. This includes the low and high frequencies and the exact midband frequency.

- **f low** contains the low frequency of the third-octave band.
- **f high** contains the high frequency of the third-octave band.
- **midband** contains the exact midband or center frequency of the third-octave band.

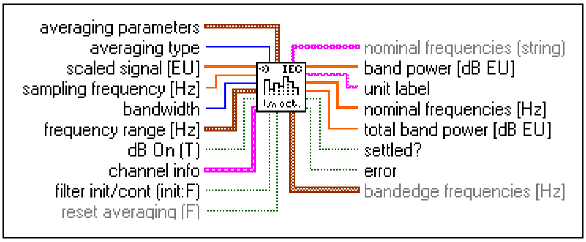
IEC - Fractional-octave Analysis

Performs a fractional-octave analysis on the input signal. The fractional-octave filters are designed as fractional-octave-band filters, class 1, IEC 1260:1995. To meet this standard, the sampling frequency must be at least 3 times the highest midband frequency for octave analysis and at least 2.5 times the highest midband frequency for 1/3, 1/6, 1/12 or 1/24 octave analysis.

Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Based on the selected **averaging type**, this VI uses the additional settings in **averaging parameters**.

This VI returns the power in each fractional-octave band and the **total band power**, both expressed in decibels. The total band power is defined as the sum of the individual band powers.

To ensure a continuous filtering operation call this VI the first time with **filter init/cont** set to FALSE and then call it with **filter init/cont** set to TRUE.



averaging parameters is a cluster that defines how the selected **averaging type** is computed.

linear mode specifies how the linear averaging is performed if the **averaging type** is set to Linear. The following linear averaging modes are available:

- Start averaging once settled (default)
- Continuous averaging
- Single averaging

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

time constant [ms] specifies the time constant, expressed in ms, used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

confidence [dB] specifies the confidence value, expressed in decibels, used if the **averaging type** is set to Equal confidence.

peak mode specifies how the peak detection is performed if the **averaging type** is set to Peak. The following peak detection modes are available:

- Start peak detection once settled (default)
- Continuous peak detection
- Single peak detection

averaging type specifies the type of averaging that is applied to the result of the 1/*n* octave filtering. Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Each type of averaging uses specific parameters defined in the **averaging parameters** cluster.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

bandwidth specifies the bandwidth of the 1/*n* octave filters. Available bandwidths are 1, 1/3, 1/6, 1/12, and 1/24 octaves.

frequency range [Hz] is a cluster that specifies the low and high frequencies, expressed in hertz, of the 1/*n* octave analysis

- **low freq.** specifies the low frequency of the frequency range.
- **high freq.** specifies the high frequency of the frequency range.

dB On (T) specifies if results are expressed in decibels. Default is TRUE.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

- **sensor sensitivity [mV/EU]** contains the sensitivity of the sensor, expressed in mV/EU.
- **engineering units** is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.
- **dB reference [EU]** is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.
- **custom label** contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.
- **weighting filter** indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

reset averaging (F) specifies if the selected averaging process has to be reset. Default is FALSE.

nominal frequencies (string) is an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band. For 1 octave and 1/3 octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards. For 1/6 octave, 1/12 octave, and 1/24 octave, strings are formatted according to Annex A of the IEC 1260:1995 standard.

band power [dB EU] returns the power, expressed in decibels, associated with each 1/*n* octave band.

unit label returns a string containing the selected engineering units.

nominal frequencies [Hz] is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band.

total band power [dB EU] total band power returns the sum of the power contained in all the 1/*n* octave bands.

settled? returns TRUE when all 1/*n* octave filters are settled.

error is TRUE if an error is encountered during the design process of the filter.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each 1/*n* octave band. This includes the low and high frequencies and the exact midband frequency.

f low contains the low frequency of the 1/*n* octave band.

f high contains the high frequency of the 1/*n* octave band.

midband contains the exact midband or center frequency of the 1/*n* octave band.

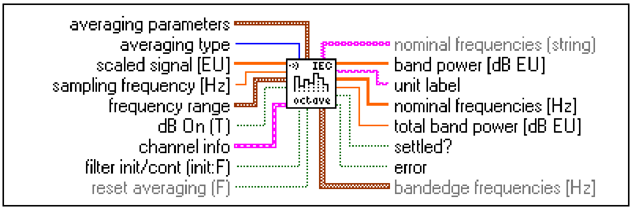
IEC - Octave Analysis

Performs an octave analysis on the input signal. The octave filters are designed as octave-band filters, class 1, IEC 1260:1995. To meet this standard, the sampling frequency must be at least three times the highest midband frequency.

Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Based on the selected **averaging type**, this VI uses the additional settings in **averaging parameters**.

This VI returns the power in each octave band and the **total band power**, both expressed in decibels. The total band power is defined as the sum of the individual band powers.

To ensure a continuous filtering operation call this VI the first time with **filter init/cont** set to FALSE and then call it with **filter init/cont** set to TRUE.



averaging parameters is a cluster that defines how the selected **averaging type** is computed.

linear mode specifies how the linear averaging is performed if the **averaging type** is set to Linear. The following linear averaging modes are available:

- Start averaging once settled (default)
- Continuous averaging
- Single averaging

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

time constant [ms] specifies the time constant, expressed in ms, used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

confidence [dB] specifies the confidence value, expressed in decibels, used if the **averaging type** is set to Equal confidence.

peak mode specifies how the peak detection is performed if the **averaging type** is set to Peak. The following peak detection modes are available:

- Start peak detection once settled (default)
- Continuous peak detection
- Single peak detection

averaging type specifies the type of averaging that is applied to the result of the octave filtering. Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Each type of averaging uses specific parameters defined in the **averaging parameters** cluster.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

frequency range is a cluster that specifies the low and high frequencies of the octave analysis

low band specifies the center frequency of the lowest octave band.

high band specifies the center frequency of the highest octave band.

dB On (T) specifies if results are expressed in decibels. Default is TRUE.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

filter init/cont (init:F) controls the initialization of the octave filters. When **filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

reset averaging (F) specifies if the selected averaging process has to be reset. Default is FALSE.

nominal frequencies (string) is an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each octave band. For octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards.

band power [dB EU] returns the power, expressed in decibels, associated with each octave band.

unit label returns a string containing the selected engineering units.

nominal frequencies [Hz] is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each octave band.

total band power [dB EU] returns the sum of the power contained in all the octave bands.

settled? returns TRUE when all octave filters are settled.

error is TRUE if an error is encountered during the design process of the filter.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each octave band. This includes the low and high frequencies and the exact midband frequency.

f low contains the low frequency of the octave band.

f high contains the high frequency of the octave band.

midband contains the exact midband or center frequency of the octave band.

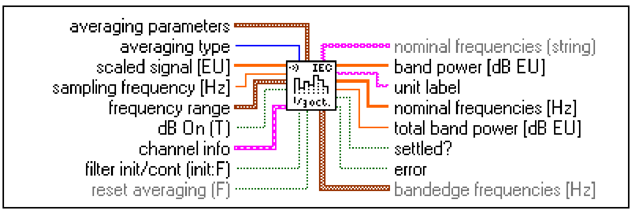
IEC - Third-octave Analysis

Performs a third-octave analysis on the input signal. The third-octave filters are designed as third-octave-band filters, class 1, IEC 1260:1995. To meet this standard, the sampling frequency must be at least 2.5 times the highest midband frequency.

Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Based on the selected **averaging type**, this VI uses the additional settings in **averaging parameters**.

This VI returns the power in each third-octave band and the **total band power**, both expressed in decibels. The total band power is defined as the sum of the individual band powers.

To ensure a continuous filtering operation call this VI the first time with **filter init/cont** set to FALSE and then call it with **filter init/cont** set to TRUE.



averaging parameters is a cluster that defines how the selected **averaging type** is computed.

linear mode specifies how the linear averaging is performed if the **averaging type** is set to Linear. The following linear averaging modes are available:

- Start averaging once settled (default)
- Continuous averaging
- Single averaging

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

time constant [ms] specifies the time constant, expressed in ms, used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

confidence [dB] specifies the confidence value, expressed in decibels, used if the **averaging type** is set to Equal confidence.

peak mode specifies how the peak detection is performed if the **averaging type** is set to Peak. The following peak detection modes are available:

- Start peak detection once settled (default)
- Continuous peak detection
- Single peak detection

averaging type specifies the type of averaging that is applied to the result of the third-octave filtering. Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Each type of averaging uses specific parameters defined in the **averaging parameters** cluster.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

frequency range is a cluster that specifies the low and high frequencies of the third-octave analysis

low band specifies the center frequency of the lowest third-octave band.

high band specifies the center frequency of the highest third-octave band.

dB On (T) specifies if results are expressed in decibels. Default is TRUE.

channel info contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a control that contains a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

filter init/cont (init:F) controls the initialization of the third-octave filters. When **filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

reset averaging (F) specifies if the selected averaging process has to be reset. Default is FALSE.

nominal frequencies (string) is an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each third-octave band. For third-octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards.

band power [dB EU] returns the power, expressed in decibels, associated with each third-octave band.

unit label returns a string containing the selected engineering units.

nominal frequencies [Hz] is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each third-octave band.

total band power [dB EU] returns the sum of the power contained in all the third-octave bands.

settled? returns TRUE when all third-octave filters are settled.

error is TRUE if an error is encountered during the design process of the filter.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each third-octave band. This includes the low and high frequencies and the exact midband frequency.

- **f low** contains the low frequency of the third-octave band.
- **f high** contains the high frequency of the third-octave band.
- **midband** contains the exact midband or center frequency of the third-octave band.

Advanced Subpalette

The **Advanced** subpalette provides a VI to perform fractional-octave analysis and low-level VIs to design the octave filter coefficients and perform the actual filtering. These VIs are designed for the advanced user.

This subpalette also provides the basic tools required for the various averaging modes described earlier in this chapter. Figure 8-2 shows the **Functions»Sound and Vibration Toolset»Octave Analysis»Advanced** subpalette.

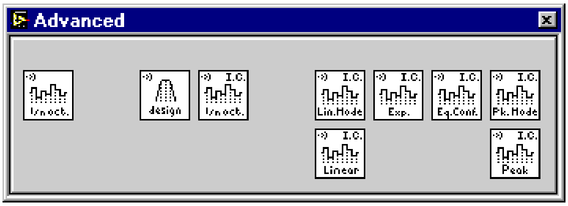


Figure 8-2. Octave Analysis Advanced Subpalette

Design Octave Filter Set

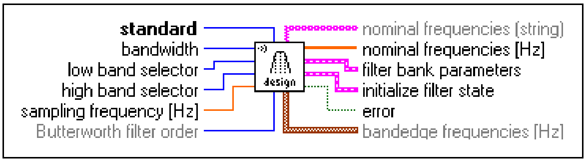
Designs the fractional-octave filters. This design process generates the **filter bank parameters** and the **initialize filter state**.

You can use the following standards to design the fractional-octave filters:

- ANSI S1.11-1986, Order x, Type 1-D, optional range. In this case, the optional **Butterworth filter order** parameter specifies the order used to design the filters. The default value is Order 3 (x = 3);
- Fractional-octave-band filters, class 1, IEC 1260:1995.

For both standards, the sampling frequency must be at least 3 times the highest midband frequency for octave analysis and at least 2.5 times the highest midband frequency for 1/3, 1/6, 1/12, or 1/24 octave analysis. Available bandwidths are 1/1, 1/3, 1/6, 1/12, and 1/24 octave.

The frequency range is based on the low and high band selectors. A value of 0 corresponds to the reference frequency of the audio-frequency range, 1kHz. Each successive value corresponds to the next fractional-octave band. Positive values cover the range above 1 kHz, and negative values cover the range below 1 kHz.



standard specifies which standard to use for the design of the 1/*n* octave filters. The following two standards are available:

- ANSI S1.11-1986
- IEC 1260:1995

Note In both cases, exact midband frequencies are computed according to the base 2 system.

bandwidth specifies the bandwidth of the 1/*n* octave filters. Available bandwidths are 1, 1/3, 1/6, 1/12, and 1/24 octaves.

low band selector specifies the band number of the lowest frequency 1/*n* octave filter.

high band selector specifies the band number of the highest frequency 1/*n* octave filter.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

Butterworth filter order specifies the order of the Butterworth filters used for 1/*n* octave analysis. In the case of the ANSI standard, the default value is Order 3, but you can select any order between 3 and 20. If the IEC standard is used, this parameter is ignored.

nominal frequencies (string) is an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band. For 1 octave and 1/3 octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards. For 1/6 octave, 1/12 octave, and 1/24 octave, strings are formatted according to Annex A of the IEC 1260:1995 standard.

nominal frequencies [Hz] is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band.

filter bank parameters returns a cluster that contains the parameters needed to perform the 1/*n* octave analysis.

initialize filter state returns a cluster that contains the initial internal state.

error is TRUE if an error is encountered during the design process of the filter.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each 1/*n* octave band. This includes the low and high frequencies and the exact midband frequency.

f low contains the low frequency of the 1/*n* octave band.

f high contains the high frequency of the 1/*n* octave band.

midband contains the exact midband or center frequency of the 1/*n* octave band.

Fractional-octave Analysis

Performs a fractional-octave analysis on the input signal.

The fractional-octave filters are designed in accordance with one of the following standards:

- ANSI S1.11-1986, Order 3, Type 1-D, optional range
- Fractional-octave-band filters, class 1, IEC 1260:1995

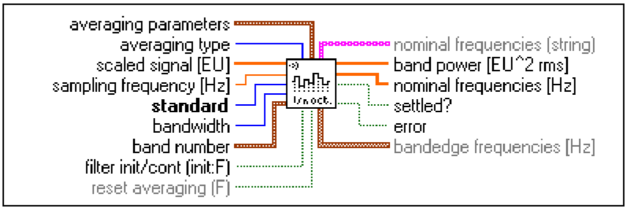
For both standards, the sampling frequency must be at least 3 times the highest midband frequency for octave analysis and at least 2.5 times the highest midband frequency for 1/3, 1/6, 1/12, or 1/24 octave analysis. Available bandwidths are 1/1, 1/3, 1/6, 1/12, and 1/24 octave.

The frequency range is based on the low and high bands specified in the **band number** cluster. A value of 0 corresponds to the reference frequency of the audio-frequency range 1 kHz. Each successive value corresponds to the next fractional-octave band. Positive values cover the range above 1 kHz, and negative values cover the range below 1 kHz.

Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Based on the selected **averaging type**, this VI uses the additional settings in a**veraging parameters**.

This VI returns the power in each fractional-octave band.

To ensure a continuous filtering operation call this VI the first time with **filter init/cont** set to FALSE and then call it with **filter init/cont** set to TRUE.



averaging parameters is a cluster that defines how the selected **averaging type** is computed.

linear mode specifies how the linear averaging is performed if the **averaging type** is set to Linear. The following linear averaging modes are available:

- Start averaging once settled (default)
- Continuous averaging
- Single averaging

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

time constant [ms] specifies the time constant, expressed in ms, used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

confidence [dB] specifies the confidence value, expressed in decibels, used if the **averaging type** is set to Equal confidence.

peak mode specifies how the peak detection is performed if the **averaging type** is set to Peak. The following peak detection modes are available:

- Start peak detection once settled (default)
- Continuous peak detection
- Single peak detection

averaging type specifies the type of averaging that is applied to the result of the 1/*n* octave filtering. Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Each type of averaging uses specific parameters defined in the **averaging parameters** cluster.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

standard specifies which standard to use for the design of the 1/*n* octave filters. The following two standards are available:

- ANSI S1.11-1986
- IEC 1260:1995

Note In both cases, exact midband frequencies are computed according to the base 2 system.

bandwidth specifies the bandwidth of the 1/*n* octave filters. Available bandwidths are 1, 1/3, 1/6, 1/12, and 1/24 octaves.

band number is a cluster that contains the low and high band numbers used for the 1/*n* octave analysis.

low band specifies the band number of the lowest 1/*n* octave band.

high band specifies the band number of the highest 1/*n* octave band.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

reset averaging (F) specifies if the selected averaging process has to be reset. Default is FALSE.

nominal frequencies (string) is an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band. For 1 octave and 1/3 octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards. For 1/6 octave, 1/12 octave, and 1/24 octave, strings are formatted according to Annex A of the IEC 1260:1995 standard.

band power [EU^2 rms] returns the power associated with each 1/*n* octave band.

nominal frequencies [Hz] is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band.

settled? returns TRUE when all 1/*n* octave filters are settled.

error is TRUE if an error is encountered during the design process of the filter.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each 1/*n* octave band. This includes the low and high frequencies and the exact midband frequency.

- **f low** contains the low frequency of the 1/*n* octave band.
- **f high** contains the high frequency of the 1/*n* octave band.
- **midband** contains the exact midband or center frequency of the 1/*n* octave band.

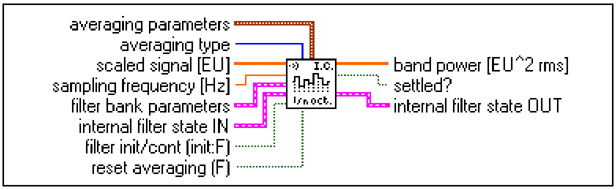
Fractional-octave Analysis with IC

Performs a fractional-octave analysis on the input signal. This VI requires the **filter bank parameters** and **initialize filter state** information the Design Octave Filter Set VI generates.

Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Based on the selected **averaging type**, this VI uses the additional settings in **averaging parameters**.

This VI returns the power in each fractional-octave band.

The **internal filter state IN** and **internal filter state OUT** keep track of the internal state of the filter to ensure continuous operation of the filter. To achieve this continuous operation, you must call this VI the first time with **filter init/cont** set to FALSE. Call it all subsequent times with **filter init/cont** set to TRUE. Wire the **internal filter state OUT** of this call to the **internal filter state IN** for the next call to this VI.



averaging parameters is a cluster that defines how the selected **averaging type** is computed.

linear mode specifies how the linear averaging is performed if the **averaging type** is set to Linear. The following linear averaging modes are available:

- Start averaging once settled (default)
- Continuous averaging
- Single averaging

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

time constant [ms] specifies the time constant, expressed in ms, used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

confidence [dB] specifies the confidence value, expressed in decibels, used if the **averaging type** is set to Equal confidence.

peak mode specifies how the peak detection is performed if the **averaging type** is set to Peak. The following peak detection modes are available:

- Start peak detection once settled (default)
- Continuous peak detection
- Single peak detection

averaging type specifies the type of averaging that is applied to the result of the 1/*n* octave filtering. Available averaging types are Linear, Exponential (default), Equal confidence, and Peak. Each type of averaging uses specific parameters defined in the **averaging parameters** cluster.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

filter bank parameters is a cluster that contains the parameters needed to perform the 1/*n* octave analysis.

internal filter state IN is a cluster that contains the initial internal states of the 1/*n* octave filters.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

reset averaging (F) specifies if the selected averaging process has to be reset. Default is FALSE.

band power [EU^2 rms] returns the power associated with each 1/*n* octave band.

settled? returns TRUE when all 1/*n* octave filters are settled.

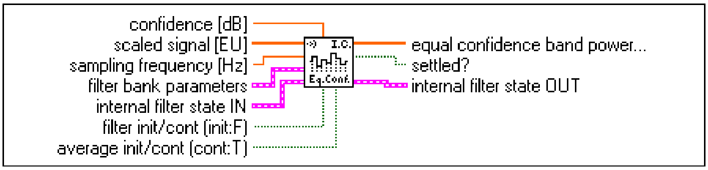
internal filter state OUT returns a cluster that contains the final internal states of the 1/*n* octave filters.

Fractional-octave Analysis with IC (equal conf)

Performs an equal confidence fractional-octave analysis on the input signal. This VI requires the **filter bank parameters** and **initialize filter state** information the Design Octave Filter Set VI generates.

This VI returns the equal confidence power in each fractional-octave band.

The **internal filter state IN** and **internal filter state OUT** keep track of the internal state of the filter to ensure continuous operation of the filter. To achieve this continuous operation, you must call this VI the first time with **filter init/cont** set to FALSE. Call it all subsequent times with **filter init/cont** set to TRUE. Wire the **internal filter state OUT** of this call to **internal filter state IN** for the next call to this VI.



confidence [dB] specifies the confidence value, expressed in decibels.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

filter bank parameters is a cluster that contains the parameters needed to perform the 1/*n* octave analysis.

internal filter state IN is a cluster that contains the initial internal states of the 1/*n* octave filters.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the internal state is initialized to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

average init/cont (cont:T) controls the initialization of the averaging process. When **average init/cont (cont:T)** is FALSE, the internal states are initialized to zero. When **average init/cont (cont:T)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To average a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous averaging of all remaining blocks.

equal confidence band power [EU^2 rms] returns the equal confidence band power associated with each 1/*n* octave band.

settled? returns TRUE when all 1/*n* octave filters are settled.

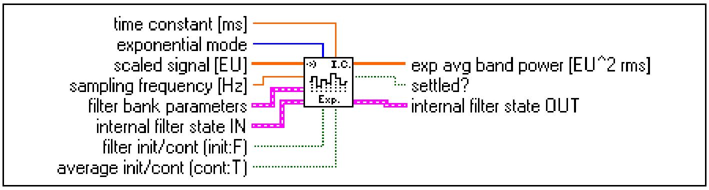
internal filter state OUT returns a cluster that contains the final internal states of the 1/*n* octave filters.

Fractional-octave Analysis with IC (exp avg)

Performs an exponential average fractional-octave analysis on the input signal. This VI requires the **filter bank parameters** and **initialize filter state** information the Design Octave Filter Set VI generates.

This VI returns the exponential average power in each fractional-octave band.

The **internal filter state IN** and **internal filter state OUT** keep track of the internal state of the filter to ensure continuous operation of the filter. To achieve this continuous operation, you must call this VI the first time with **filter init/cont** set to FALSE. Call it all subsequent times with **filter init/cont** set to TRUE. Wire the **internal filter state OUT** of this call to the **internal filter state IN** for the next call to this VI.



time constant [ms] specifies the time constant, expressed in ms, used for exponential averaging if the selected mode is Custom. In all other cases, this parameter is ignored.

exponential mode specifies the exponential averaging mode. Standard modes include Slow, Fast (default), and Impulse. The Custom mode allows you to specify any **time constant**.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

filter bank parameters is a cluster that contains the parameters needed to perform the 1/*n* octave analysis.

internal filter state IN is a cluster that contains the initial internal states of the 1/*n* octave filters.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

average init/cont (cont:T) controls the initialization of the averaging process. When **average init/cont (cont:T)** is FALSE, the internal states are initialized to zero. When **average init/cont (cont:T)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To average a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous averaging of all remaining blocks.

exp avg band power [EU^2 rms] returns the exponential averaged band power associated with each 1/*n* octave band.

settled? returns TRUE when all 1/*n* octave filters are settled.

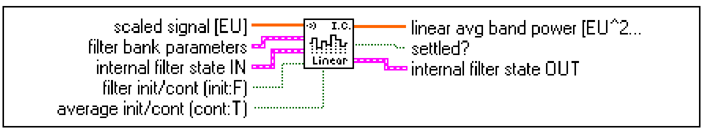
internal filter state OUT returns a cluster that contains the final internal states of the 1/*n* octave filters.

Fractional-octave Analysis with IC (linear avg)

Performs a linear average fractional-octave analysis on the input signal. This VI requires the **filter bank parameters** and **initialize filter state** information the Design Octave Filter Set VI generates.

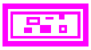
This VI returns the linear average power in each fractional-octave band.

The **internal filter state IN** and **internal filter state OUT** keep track of the internal state of the filter to ensure continuous operation of the filter. To achieve this continuous operation, you must call this VI the first time with **filter init/cont** set to FALSE. Call it all subsequent times with **filter init/cont** set to TRUE. Wire the **internal filter state OUT** of this call to the **internal filter state IN** for the next call to this VI.



scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

filter bank parameters is a cluster that contains the parameters needed to perform the 1/*n* octave analysis.



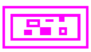
internal filter state IN is a cluster that contains the initial internal states of the 1/*n* octave filters.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

average init/cont (cont:T) controls the initialization of the averaging process. When **average init/cont (cont:T)** is FALSE, the internal states are initialized to zero. When **average init/cont (cont:T)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To average a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous averaging of all remaining blocks.

linear avg band power [EU^2 rms] returns the linear averaged band power associated with each 1/*n* octave band.

settled? returns TRUE when all 1/*n* octave filters are settled.



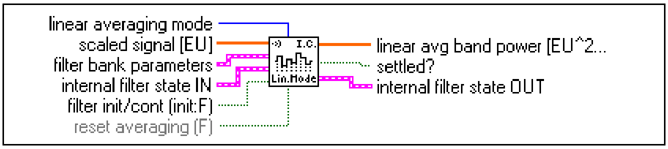
internal filter state OUT returns a cluster that contains the final internal states of the 1/*n* octave filters.

Fractional-octave Analysis with IC (linear mode)

Performs a linear average fractional-octave analysis on the input signal. This VI requires the **filter bank parameters** and **initialize filter state** information the Design Octave Filter Set VI generates.

This VI returns the linear average power in each fractional-octave band.

The **internal filter state IN** and **internal filter state OUT** keep track of the internal state of the filter to ensure continuous operation of the filter. To achieve this continuous operation, you must call this VI the first time with **filter init/cont** set to FALSE. Call it all subsequent times with **filter init/cont** set to TRUE. Wire the **internal filter state OUT** of this call to the **internal filter state IN** for the next call to this VI.



linear averaging mode specifies how the linear averaging is performed. Available linear averaging modes are:

- Start averaging once settled (default)
- Continuous averaging
- Single averaging

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

filter bank parameters is a cluster that contains the parameters needed to perform the 1/*n* octave analysis.

internal filter state IN is a cluster that contains the initial internal states of the 1/*n* octave filters.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

reset averaging (F) specifies if the selected averaging process has to be reset. Default is FALSE.

linear avg band power [EU^2 rms] returns the linear averaged band power associated with each 1/*n* octave band.

settled? returns TRUE when all 1/*n* octave filters are settled.

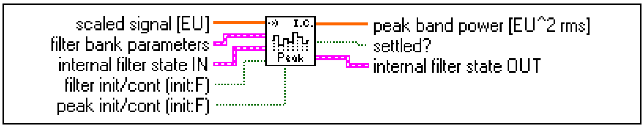
internal filter state OUT returns a cluster that contains the final internal states of the 1/*n* octave filters.

Fractional-octave Analysis with IC (peak)

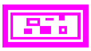
Performs a peak fractional-octave analysis on the input signal. This VI requires the **filter bank parameters** and **initialize filter state** information the Design Octave Filter Set VI generates.

This VI returns the peak power in each fractional-octave band.

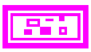
The **internal filter state IN** and **internal filter state OUT** keep track of the internal state of the filter to ensure continuous operation of the filter. To achieve this continuous operation, you must call this VI the first time with **filter init/cont** set to FALSE. Call it all subsequent times with **filter init/cont** set to TRUE. Wire the **internal filter state OUT** of this call to the **internal filter state IN** for the next call to this VI.



scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.



filter bank parameters is a cluster that contains the parameters needed to perform the 1/*n* octave analysis.



internal filter state IN is a cluster that contains the initial internal states of the 1/*n* octave filters.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

peak init/cont (init:F) controls the initialization of the peak detection. To analyze a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block, and to TRUE for continuous peak detection on all remaining blocks.

peak band power [EU^2 rms] returns the peak band power associated with each 1/*n* octave band.

settled? returns TRUE when all 1/*n* octave filters are settled.

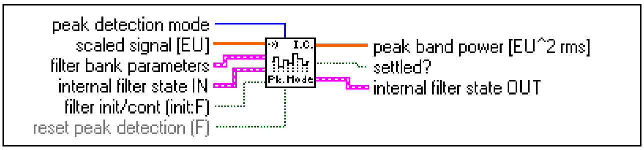
internal filter state OUT returns a cluster that contains the final internal states of the 1/*n* octave filters.

Fractional-octave Analysis with IC (peak mode)

Performs a peak fractional-octave analysis on the input signal. This VI requires the **filter bank parameters** and **initialize filter state** information the Design Octave Filter Set VI generates.

This VI returns the peak power in each fractional-octave band.

The **internal filter state IN** and **internal filter state OUT** keep track of the internal state of the filter to ensure continuous operation of the filter. To achieve this continuous operation, you must call this VI the first time with **filter init/cont** set to FALSE. Call it all subsequent times with **filter init/cont** set to TRUE. Wire the **internal filter state OUT** of this call to the **internal filter state IN** for the next call to this VI.



peak detection mode specifies how the peak detection is performed. Available peak detection modes are:

- Start peak detection once settled (default)
- Continuous peak detection
- Single peak detection

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

filter bank parameters is a cluster that contains the parameters needed to perform the 1/*n* octave analysis.

internal filter state IN is a cluster that contains the initial internal states of the 1/*n* octave filters.

filter init/cont (init:F) controls the initialization of the 1/*n* octave filters. When **filter init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

reset peak detection (F) specifies if the peak detection process has to be reset (TRUE) or not (FALSE, default).

peak band power [EU^2 rms] returns the peak band power associated with each 1/*n* octave band.

settled? returns TRUE when all 1/*n* octave filters are settled.

internal filter state OUT returns a cluster that contains the final internal states of the 1/*n* octave filters.

Utilities Subpalette

The **Utilities** subpalette offers tools to specify the frequency range of interest when performing 1/1 octave, 1/3 octave, or more general fractional-octave analysis.

Two VIs obtain information about characteristic frequencies of the octave filters, such as the nominal, or preferred, frequencies, the exact midband frequencies, and the bandedge frequencies. Figure 8-3 shows the **Functions»Sound and Vibration Toolset»Octave Analysis»Utilities** subpalette.

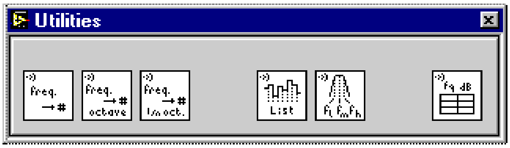
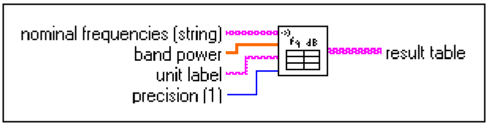


Figure 8-3. Octave Analysis Utilities Subpalette

Build Result Table

Generates a table based on the results of a fractional-octave analysis. The first column of the table contains the nominal frequencies. The second column contains the band power with the appropriate unit label.

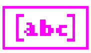


nominal frequencies (string) is an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band. For octave and 1/3 octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards. For 1/6 octave, 1/12 octave, and 1/24 octave, strings are formatted according to Annex A of the IEC 1260:1995 standard.

band power is an array that contains the results of the 1/*n* octave analysis

unit label is a string containing the selected engineering units.

precision (1) specifies the number of digits to display after the decimal point. Default value is 1.

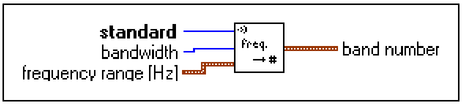


result table returns a table that contains the nominal frequencies in the first column and the 1/*n* octave band power in the second column.

Frequency Range to Band Number

Converts the selected **frequency range** to the corresponding **band number**, based on the specified **bandwidth** and **standard** (ANSI or IEC).

The **band number** is based on the low and high bands specified in the **frequency range** cluster. A value of 0 corresponds to the reference frequency of the audio-frequency range, 1 kHz. Each successive value corresponds to the next fractional-octave band. Positive values cover the range above 1 kHz, and negative values cover the range below 1 kHz.



standard specifies which standard to use. The following two standards are available:

- ANSI S1.11-1986
- IEC 1260:1995

bandwidth specifies the bandwidth of the 1/*n* octave filters. Available bandwidths are 1, 1/3, 1/6, 1/12, and 1/24 octaves.

frequency range [Hz] is a cluster that specifies the low and high frequencies, expressed in hertz, of the 1/*n* octave analysis

low freq. specifies the low frequency of the frequency range.

high freq. specifies the high frequency of the frequency range.

band number returns a cluster that contains the low and high band numbers used for the 1/*n* octave analysis.

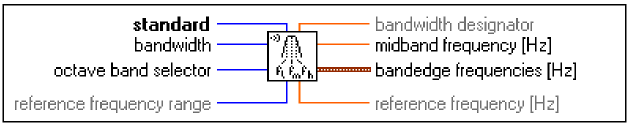
low band specifies the band number of the lowest 1/*n* octave band.

high band specifies the band number of the highest 1/*n* octave band.

Octave Band Frequencies

Returns the midband and bandedge frequencies associated with a particular fractional-octave band based on the specified **bandwidth** and **standard** (ANSI or IEC).

The **octave band selector** specifies the fractional-octave band. A value of 0 corresponds to the reference frequency of the audio-frequency range, 1 kHz. Each successive value corresponds to the next fractional-octave band. Positive values cover the range above 1 kHz, and negative values cover the range below 1 kHz.



standard specifies which standard to use. The following two standards are available:

- ANSI S1.11-1986
- IEC 1260:1995

Note In both cases, exact midband frequencies are computed according to the base 2 system.

bandwidth specifies the bandwidth of the 1/*n* octave filters. Available bandwidths are 1, 1/3, 1/6, 1/12, and 1/24 octaves.

octave band selector specifies the band number of the 1/*n* octave filter.

reference frequency range specifies the reference frequency used for the 1/*n* octave analysis. Available ranges are Infrasonic (1 Hz), Audio (1 kHz, default), and Ultrasonic (1 MHz).

bandwidth designator returns an integer associated with the selected fractional-octave bandwidth. For octave band filters, b=1; for 1/3 octave band filters, b=3; for 1/6 octave band filters, b=6; and so on.

midband frequency [Hz] contains the exact midband or center frequency, expressed in hertz, of the 1/*n* octave band.

bandedge frequencies [Hz] returns a cluster that contains the high and low frequencies associated with the selected 1/*n* octave band.

f low contains the low frequency of the 1/*n* octave band.

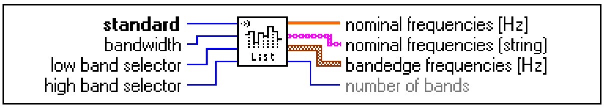
f high contains the high frequency of the 1/*n* octave band.

reference frequency [Hz] returns the reference frequency, expressed in hertz, used to compute the center frequency of any fractional-octave band. The reference frequency for the audio range is 1 kHz.

Octave Band Frequencies List

Returns nominal, or preferred, frequencies and bandedge frequencies based on the specified **standard** (ANSI or IEC), **bandwidth**, and low and high band selectors.

The frequency range is based on the low and high band selectors. A value of 0 corresponds to the reference frequency of the audio-frequency range, 1 kHz. Each successive value corresponds to the next fractional-octave band. Positive values cover the range above 1 kHz, and negative values cover the range below 1 kHz.



standard specifies which standard to use. The following two standards are available:

- ANSI S1.11-1986
- IEC 1260:1995

Note In both cases, exact midband frequencies are computed according to the base 2 system.

bandwidth specifies the bandwidth of the 1/*n* octave filters. Available bandwidths are 1, 1/3, 1/6, 1/12, and 1/24 octaves.

low band selector specifies the band number of the lowest frequency 1/*n* octave filter.

high band selector specifies the band number of the highest frequency 1/*n* octave filter.

nominal frequencies [Hz] returns an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band.

nominal frequencies (string) returns an array that contains, in text format, the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band. For 1 octave and 1/3 octave bands, strings are returned based on tabulated values specified by the ANSI and IEC standards. For 1/6 octave, 1/12 octave, and 1/24 octave, strings are formatted according to Annex A of the IEC 1260:1995 standard.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each 1/*n* octave band. This includes the low and high frequencies and the exact midband frequency.

f low contains the low frequency of the 1/*n* octave band.

f high contains the high frequency of the 1/*n* octave band.

midband contains the exact midband or center frequency of the 1/*n* octave band.

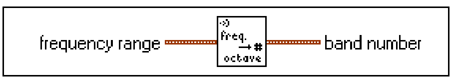
number of bands returns the number of fractional-octave bands for the selected frequency range.

Octave Range to Band Number

Converts the selected octave **frequency range** to the corresponding **band number**.

The **band number** is based on the low and high bands specified in the **frequency range** cluster. A value of 0 corresponds to the reference frequency of the audio-frequency range, 1 kHz. Each successive value corresponds to the next octave band. Positive values cover the range above 1 kHz, and negative values cover the range below 1 kHz.

The **frequency range** cluster uses a pre-defined list of frequencies from 16 Hz to 16 kHz, in octave steps. If you need to use any other specific frequency outside of this frequency range, use the Frequency Range to Band Number VI instead.



frequency range is a cluster that specifies the low and high frequencies of the octave analysis

low band specifies the center frequency of the lowest octave band.

high band specifies the center frequency of the highest octave band.

band number returns a cluster that contains the low and high band numbers used for the octave analysis.

low band specifies the band number of the lowest octave band.

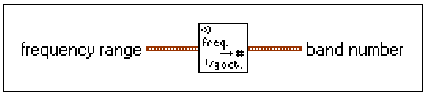
high band specifies the band number of the highest octave band.

Third-Octave Range to Band Number

Converts the selected third-octave **frequency range** to the corresponding **band number.**

The **band number** is based on the low and high bands specified in the **frequency range** cluster. A value of 0 corresponds to the reference frequency of the audio-frequency range, 1 kHz. Each successive value corresponds to the next third-octave band. Positive values cover the range above 1 kHz, and negative values cover the range below 1 kHz.

The **frequency range** cluster uses a pre-defined list of frequencies from 20 Hz to 20 kHz in third-octave steps. If you need to use any other specific frequency outside of this frequency range, you should use the Frequency Range to Band Number VI instead.



frequency range is a cluster that specifies the low and high frequencies of the third-octave analysis

low band specifies the center frequency of the lowest third-octave band.

high band specifies the center frequency of the highest third-octave band.

band number returns a cluster that contains the low and high band numbers used for the third-octave analysis.

low band specifies the band number of the lowest third-octave band.

high band specifies the band number of the highest third-octave band.

9

Weighting Filters

This chapter describes the Weighting Filters VIs. Figure 9-1 shows the **Functions»Sound and Vibration Toolset»Weighting Filters** palette.

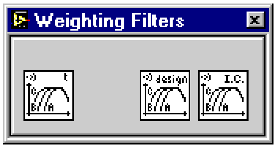


Figure 9-1. Weighting Filters Palette

Introduction

The **Weighting Filters** palette provides VIs to apply an A, B, or C weighting filter on time-domain signals.

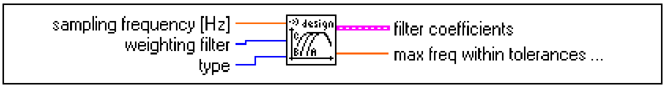
When combined with the appropriate DAQ board from National Instruments (PCI-445*X* or NI 455*X*), these VIs offer compliance with the ANSI S1.4-1983 and the ANSI S1.42-1986 standards.

These VIs accommodate any sampling frequency and design the filter coefficients to target the attenuation curves defined by these standards. Given the selected sampling frequency, the compliance with a particular type (Type 1 or Type 0) is ensured up to a specific frequency. These VIs return this frequency as the maximum frequency within tolerances.

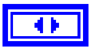
Note Selecting Type 1 or Type 0 has no influence on the design process but specifies how the actual attenuation must be tested against the ideal attenuation. Type 0 leads to a lower maximum frequency within tolerances, or a smaller valid frequency range.

Design Weighting Filter

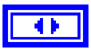
Designs the A, B, or C weighting filter coefficients. The design is based on the ANSI S1.42-1986 standard. This VI also returns the maximum frequency, expressed in hertz, within the tolerance limits based on the selected **type** (Type 1 or Type 0), according to the ANSI S1.4-1986 standard.



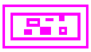
sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.



weighting filter specifies the weighting filter (A, B, or C) to apply to the signal. Use Linear if no weighting is needed.



type specifies the weighting network filter type according to the ANSI S1.4-1983 standard to use to determine the frequency range within the tolerance limits.



filter coefficients returns the filter coefficients associated with the selected weighting filter.

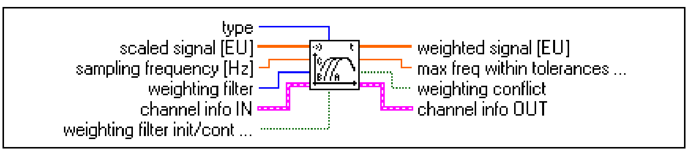
max freq within tolerances [Hz] returns the maximum frequency, expressed in hertz, that is within the tolerance limits based on the selected **type**. Tolerance limits are based on the ANSI S1.4-1983 standard.

Weighting Filter

Applies the selected weighting filter (A, B, or C) on the input signal.

Based on the selected **type** (Type 1 or Type 0), this VI returns the maximum frequency within tolerances for this particular type.

To ensure continuous filtering operation, call this VI the first time with **weighting filter init/cont** set to FALSE and then call it with **weighting filter init/cont** set to TRUE.



type specifies the weighting network filter type according to the ANSI S1.4-1983 standard to use to determine the frequency range within the tolerance limits.

scaled signal [EU] contains the scaled signal, expressed in the selected engineering units.

sampling frequency [Hz] specifies the sampling frequency, expressed in hertz. If you are using a DAQ board, confirm that the board supports this frequency.

weighting filter specifies the weighting filter (A, B, or C) to apply to the signal. Use Linear if no weighting is needed.

channel info IN contains the information relative to the measurement system used before the signal reaches the DAQ board.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a selector containing a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

weighting filter init/cont (init:F) controls the initialization of the weighting filter. When **weighting filter init/cont (init:F)** is FALSE (default), the VI designs the filter coefficients and initializes the internal states to zero. When **weighting filter init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

weighted signal [EU] returns the output array of filtered samples.

max freq within tolerances [Hz] returns the maximum frequency, expressed in hertz, that is within the tolerance limits based on the selected **type**. Tolerance limits are based on the ANSI S1.4-1983 standard.

weighting conflict returns TRUE if **channel info IN** shows that a weighting filter has already been applied to the signal and the selected **type** is not Linear. No weighting is applied if **weighting conflict** is TRUE.

channel info OUT returns the information relative to the measurement system used before the signal reaches the DAQ board. The **weighting filter** information is changed according to the selected **weighting filter**. If **channel info IN** shows that a weighting filter was applied to the signal, no weighting is applied and a warning, **weighting conflict** is TRUE, is issued if the selected **weighting filter** is different than Linear.

sensor sensitivity [mV/EU] contains the sensitivity of the sensor, expressed in mV/EU.

engineering units is a selector containing a list of engineering units, including V, Pa, g, m/s^2, m/s, m, in/s^2, in/s, in, and custom units.

dB reference [EU] is the reference value, expressed in the selected engineering units, used when results are computed in decibels. The typical decibels reference for sound pressure level is 20E-6 Pa.

custom label contains the string used for units when custom engineering units are selected. If **engineering units** is not set to custom, this parameter is ignored.

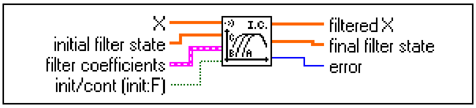
weighting filter indicates if any weighting filter (A, B, or C) has been applied to the signal.

pregain [dB] corresponds to the pregain, expressed in decibels, applied by an external amplifier.

Weighting Filter with IC

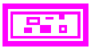
Filters the input sequence using the coefficients specified by the **filter coefficients** cluster.

The **initial filter state** and **final filter state** keep track of the internal state of the filter to ensure continuous operation of the filter if continuous mode is used. To ensure continuous filtering operation, call this VI the first time with **init/cont (init:F)** set to FALSE and then calling it with **init/cont (init:F)** set to TRUE.



X is the input array of samples to filter.

initial filter state specifies the initial internal state of the filter.

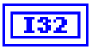


filter coefficients is a cluster that contains the filter coefficients.

init/cont (init:F) controls the initialization of the internal filter states. When **init/cont (init:F)** is FALSE (default), the internal states are initialized to zero. When **init/cont (init:F)** is TRUE, the internal filter states are initialized to the final filter states from the previous call to this instance of this VI. To filter a large data sequence that has been split into smaller blocks, set this control to FALSE for the first block and to TRUE for continuous filtering of all remaining blocks.

filtered X is the output array of filtered samples.

final filter state returns the final internal filter state. In order to filter samples continuously, final filter state has to be passed as **initial filter state** when the next call to this VI is performed.



error allows you to wire this output to the Find First Error VI to produce an error cluster. This cluster can then be wired to the Simple Error Handler VI or the General Error Handler VI for an immediate report on any errors. Select **Help»Online Reference** to find the error codes returned by LabVIEW.

10

Display

This chapter describes the Display VIs. Figure 10-1 shows the **Functions»Sound and Vibration Toolset»Display** palette**.**

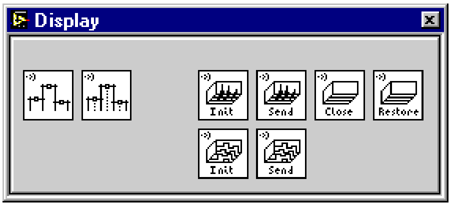


Figure 10-1. Display Palette

Introduction

The VIs in the **Display** palette display waterfall graphs and provide utilities to format octave results before displaying them in a standard XY graph. For more information, refer to Chapter 3, *Controls*.

Specific VIs display the results of frequency analysis and octave analysis in a waterfall graph.

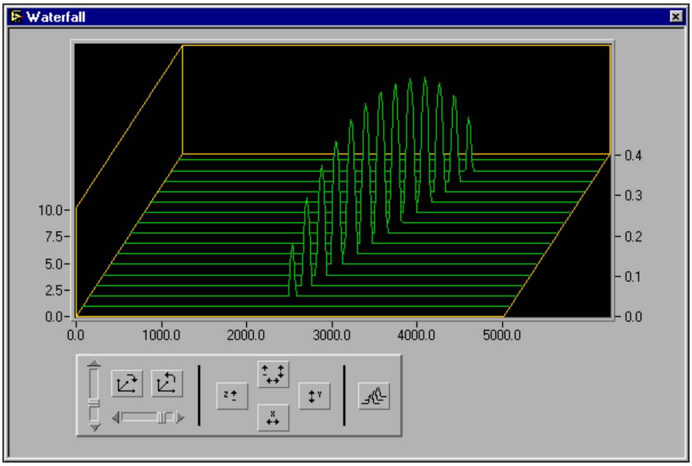


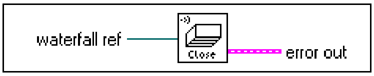
Figure 10-2. Waterfall Window

The waterfall display, refer to Figure 10-2, is an external window that you must initialize before you send any results to it, and close afterward. You can specify the size, position, title, background, and frame and plot colors when you initialize the window.

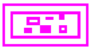
The waterfall display also allows you to control various properties of the graph. Two sliders allow you to change the perspective along the X and Z axes. You can store and restore any view at a later time. You can autoscale all three axes independently or simultaneously. Notice that autoscaling the Z axis only affects the maximum value of this axis. You must select the minimum value of the Z axis separately because this value defines the baseline of the waterfall graph. You can set this value by double-clicking on it or by specifying the value when you send data to the waterfall display. You also can turn transparency on or off.

Close Waterfall Display

Closes the waterfall display specified by the reference. This VI is used for standard waterfall and octave waterfall display graphs.

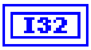


waterfall ref is the reference for the waterfall display.



error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

status is TRUE if an error occurred.

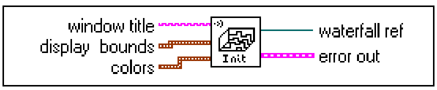


code is the number identifying an error or warning. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code. Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.

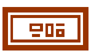
source is a string that indicates the origin of the error, if any. Usually source is the name of the VI in which the error occurred.

Initialize Waterfall Display for Octave

Initializes the waterfall display to display octave spectra and generates a waterfall reference to use to send data to this display or to close the display. This VI also specifies the title, colors and display bounds of the waterfall display. This VI does not open the waterfall display. The waterfall display opens when it receives its first data from Send Data to Waterfall for Octave VI.



window title specifies the name that appears in the window title bar.



display bounds is a cluster that specifies the bounds, in pixels, of the waterfall window.

left specifies the left position of the waterfall window.

top specifies the top position of the waterfall window.

width specifies the width of the waterfall window.

height specifies the height of the waterfall window.

colors is a cluster that specifies the different colors to use for the frame, plot, and background.

frame specifies the color to use for the frame in the waterfall display.

plot specifies the color to use for the plot in the waterfall display.

background specifies the color to use for the background in the waterfall display.

waterfall ref returns the reference for the octave waterfall display.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

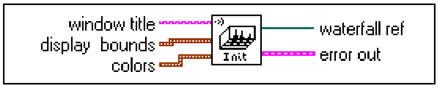
status is TRUE if an error occurred.

code is the number identifying an error or warning. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code. Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.

source is a string that indicates the origin of the error, if any. Usually source is the name of the VI in which the error occurred.

Initialize Waterfall Display

Initializes the waterfall display and generates a waterfall reference to use to send data to this display or to close the display. This VI also specifies the title, colors and display bounds of the waterfall display. This VI does not open the waterfall display. The waterfall display opens when it receives its first data from Send Data to Waterfall VI.



window title specifies the name that appears in the window title bar.

display bounds is a cluster that specifies the bounds, in pixels, of the waterfall window.

left specifies the left position of the waterfall window.

top specifies the top position of the waterfall window.

width specifies the width of the waterfall window.

height specifies the height of the waterfall window.

colors is a cluster that specifies the different colors to use for the frame, plot, and background.

frame specifies the color to use for the frame in the waterfall display.

plot specifies the color to use for the plot in the waterfall display.

background specifies the color to use for the background in the waterfall display.

waterfall ref returns the reference for the waterfall display.

error out returns a cluster that describes the error status after this VI executes. **error out** shows the error, if any, that occurred in this VI. Use the error handler VIs, located in the **Functions»Time & Dialog** palette, to look up the error code and to display the corresponding error message.

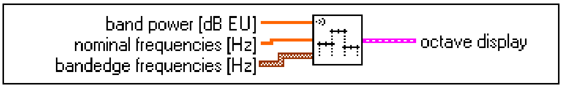
status is TRUE if an error occurred.

code is the number identifying an error or warning. If status is TRUE, code is a non-zero error code. If status is FALSE, code can be zero or a warning code. Use the error handler VIs to look up the meaning of this code and to display the corresponding error message.

source is a string that indicates the origin of the error, if any. Usually source is the name of the VI in which the error occurred.

Octave Display (line)

Converts the result of any fractional-octave analysis into an octave spectrum to display as a line graph. Use this VI to display multiple octave spectra on the same graph.



band power [dB EU] contains the power, expressed in decibels, associated with each 1/*n* octave band.

nominal frequencies [Hz] is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each 1/*n* octave band. This includes the low and high frequencies and the exact midband frequency.

f low contains the low frequency of the 1/*n* octave band.

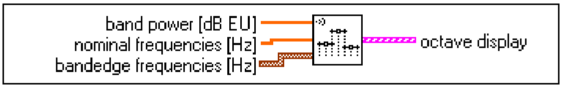
f high contains the high frequency of the 1/*n* octave band.

midband contains the exact midband or center frequency of the 1/*n* octave band.

octave display contains the octave spectrum in a format suitable to display as a line graph.

Octave Display (line+cursor)

Converts the results of any fractional-octave analysis into an octave spectrum to display as a line graph. Use this VI to display multiple octave spectra on the same graph. This VI generates two plots for each fractional-octave result. The second plot is used to lock a cursor to the center frequencies.



band power [dB EU] contains the power, expressed in decibels, associated with each 1/*n* octave band.

nominal frequencies [Hz] is an array that contains the nominal, or preferred, frequencies as opposed to the exact frequencies associated with each 1/*n* octave band.

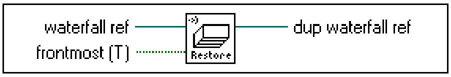
bandedge frequencies [Hz] is an array that contains the frequencies associated with each 1/*n* octave band. This includes the low and high frequencies and the exact midband frequency.

- **f low** contains the low frequency of the 1/*n* octave band.
- **f high** contains the high frequency of the 1/*n* octave band.
- **midband** contains the exact midband or center frequency of the 1/*n* octave band.

octave display contains the octave spectrum in a format suitable to display as a line graph.

Restore Waterfall Window

Restores any existing waterfall window that was closed unintentionally.



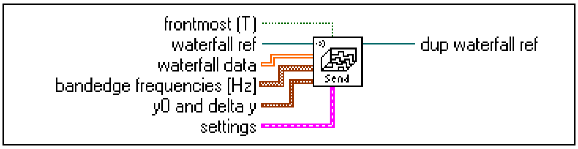
waterfall ref is the reference for the waterfall display.

frontmost (T) specifies whether to force the waterfall window as the frontmost window. Default is TRUE.

dup waterfall ref returns the waterfall reference.

Send Data to Waterfall for Octave

Sends the waterfall data (octave spectra) to the waterfall display for octave. Use the **settings** cluster to specify how to display the new data.



frontmost (T) specifies whether to force the waterfall window as the frontmost window. Default is TRUE.

waterfall ref is the reference for the octave waterfall display.

waterfall data contains the octave spectra to display in the waterfall display.

bandedge frequencies [Hz] is an array that contains the frequencies associated with each 1/*n* octave band. This includes the low and high frequencies and the exact midband frequency.

f low contains the low frequency of the 1/*n* octave band.

f high contains the high frequency of the 1/*n* octave band.

midband contains the exact midband or center frequency of the 1/*n* octave band.

y0 and delta y is a cluster that specifies the minimum value of the Y axis (y0), and the increment (delta y) for this axis.

y0 specifies the minimum value of the Y axis.

delta y specifies the increment for the Y axis.

settings is a cluster that specifies how to display the waterfall data.

clear graph before redrawing (F) specifies to clear the waterfall graph before displaying the new data. Default is FALSE.

set baseline (F) specifies to use a new value for the baseline. If not (default), the minimum value of the Z axis is used as the baseline.

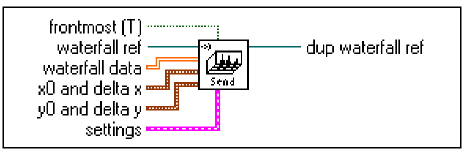
baseline specifies the value to use as the baseline. This value is used only if **set baseline** is TRUE.

autoscale (T) specifies to autoscale the waterfall graph based on the new data sent. Default is TRUE.

dup waterfall ref returns the waterfall reference.

Send Data to Waterfall

Sends the waterfall data to the waterfall display. Use the **settings** cluster to specify how to display the new data.



frontmost (T) specifies whether to force the waterfall window as the frontmost window. Default is TRUE.

waterfall ref is the reference for the waterfall display.

waterfall data contains the data to display in the waterfall display.

x0 and delta x is a cluster that specifies the minimum value of the X axis (x0), and the increment (delta x) for this axis.

x0 specifies the minimum value of the X axis.

delta x specifies the increment for the X axis.

y0 and delta y is a cluster that specifies the minimum value of the Y axis (y0), and the increment (delta y) for this axis.

y0 specifies the minimum value of the Y axis.

delta y specifies the increment for the Y axis.

settings is a cluster that specifies how to display the waterfall data.

clear graph before redrawing (F) specifies to clear the waterfall graph before displaying the new data. Default is FALSE.

set baseline (F) specifies to use a new value for the baseline. If not (default), the minimum value of the Z axis is used as the baseline.

baseline specifies the value to use as the baseline. This value is used only if **set baseline** is TRUE.

autoscale (T) specifies to autoscale the waterfall graph based on the new data sent. Default is TRUE.

dup waterfall ref returns the waterfall reference.

A

Error Codes

This appendix describes the error codes the Sound and Vibration Toolset software returns.

Code	Description
`-10007`	A channel, port, or counter is out of range for the device type or device configuration; the combination of channels is not allowed; or the scan order must be reversed (0 last).
`-10801`	An error occurred during the calibration process. Possible reasons for this error include incorrect connection of the stimulus signal, incorrect value of the stimulus signal, or malfunction of your DAQ device.
`-20008`	The input arrays do not contain the correct number of data values for this function.

B

References

This appendix lists the reference material used for the Sound and Vibration Toolset. For more information about the theory implemented in this toolset, refer to the following documents:

American National Standards Institute. ANSI S1.4-1983: *Specification for Sound Level Meters,* 1983.

American National Standards Institute. ANSI S1.11-1986: *Specification for octave-band and fractional-octave-band analog and digital filters.* New York: Acoustical Society of America, 1986.

American National Standards Institute. ANSI S1.42-1986: *Design Response of Weighting Networks for Acoustical Measurements,* 1986.

Crocker, Malcolm J. *Handbook of Acoustics*. New York: John Wiley & Sons, Inc, 1998.

International Standard IEC 266 First edition. 1975-07-15: *Preferred Frequencies for Measurements*, 1975.

International Standard IEC 651 First edition. 1979: *Sound Level Meters,* 1979.

International Standard IEC 651 Amendment 1. 1993-09: *Sound Level Meters,* 1993.

International Standard IEC 804 First edition. 1985: *Integrating-averaging Sound Level Meters,* 1985.

International Standard IEC 804 Amendment 1. 1989-09: *Integrating-averaging Sound Level Meters,* 1989.

International Standard IEC 804 Amendment 2. 1993-09: *Integrating-averaging Sound Level Meters,* 1993.

International Standard IEC 1260 First edition. 1995-07: *Octave-band and Fractional-octave-band Filters,* 1995.

Randall, R.B. *Frequency Analysis*. Nærum, Denmark: Brüel & Kjær, 1987.

J. R. Hassall and K. Zaveri. *Acoustic Noise Measurements*. Brüel & Kjær, 1988.

C

Technical Support Resources

This appendix describes the comprehensive resources available to you in the Technical Support section of the National Instruments Web site and provides technical support telephone numbers for you to use if you have trouble connecting to our Web site or if you do not have internet access.

NI Web Support

To provide you with immediate answers and solutions 24 hours a day, 365 days a year, National Instruments maintains extensive online technical support resources. They are available to you at no cost, are updated daily, and can be found in the Technical Support section of our Web site at `www.natinst.com/support`.

Online Problem-Solving and Diagnostic Resources

- **KnowledgeBase**—A searchable database containing thousands of frequently asked questions (FAQs) and their corresponding answers or solutions, including special sections devoted to our newest products. The database is updated daily in response to new customer experiences and feedback.
- **Troubleshooting Wizards**—Step-by-step guides lead you through common problems and answer questions about our entire product line. Wizards include screen shots that illustrate the steps being described and provide detailed information ranging from simple getting started instructions to advanced topics.
- **Product Manuals**—A comprehensive, searchable library of the latest editions of National Instruments hardware and software product manuals.
- **Hardware Reference Database**—A searchable database containing brief hardware descriptions, mechanical drawings, and helpful images of jumper settings and connector pinouts.
- **Application Notes**—A library with more than 100 short papers addressing specific topics such as creating and calling DLLs, developing your own instrument driver software, and porting applications between platforms and operating systems.

Software-Related Resources

- **Instrument Driver Network**—A library with hundreds of instrument drivers for control of standalone instruments via GPIB, VXI, or serial interfaces. You also can submit a request for a particular instrument driver if it does not already appear in the library.
- **Example Programs Database**—A database with numerous, non-shipping example programs for National Instruments programming environments. You can use them to complement the example programs that are already included with National Instruments products.
- **Software Library**—A library with updates and patches to application software, links to the latest versions of driver software for National Instruments hardware products, and utility routines.

Worldwide Support

National Instruments has offices located around the globe. Many branch offices maintain a Web site to provide information on local services. You can access these Web sites from `www.natinst.com/worldwide`.

If you have trouble connecting to our Web site, please contact your local National Instruments office or the source from which you purchased your National Instruments product(s) to obtain support.

For telephone support in the United States, dial 512 795 8248. For telephone support outside the United States, contact your local branch office:

Australia 03 9879 5166, Austria 0662 45 79 90 0, Belgium 02 757 00 20, Brazil 011 284 5011, Canada (Ontario) 905 785 0085, Canada (Québec) 514 694 8521, China 0755 3904939, Denmark 45 76 26 00, Finland 09 725 725 11, France 01 48 14 24 24, Germany 089 741 31 30, Hong Kong 2645 3186, India 91805275406, Israel 03 6120092, Italy 02 413091, Japan 03 5472 2970, Korea 02 596 7456, Mexico (D.F.) 5 280 7625, Mexico (Monterrey) 8 357 7695, Netherlands 0348 433466, Norway 32 27 73 00, Singapore 2265886, Spain (Madrid) 91 640 0085, Spain (Barcelona) 93 582 0251, Sweden 08 587 895 00, Switzerland 056 200 51 51, Taiwan 02 2377 1200, United Kingdom 01635 523545

Glossary

Numbers/Symbols

2D	Two-dimensional.
3D	Three-dimensional.

A

accelerometer	A sensor mounted on a structure to measure the acceleration at a particular location in one or multiple directions.
A/D	analog to digital.
aliasing	Aliasing is a phenomenon whereby an analog signal of frequency greater than half the sampling frequency (Nyquist frequency) appears, after sampling, at a frequency less than half the sampling frequency. *See also* anti-aliasing filter.
ANSI	American National Standards Institute.
anti-aliasing filter	To avoid aliasing, analog lowpass filters are used before A/D conversion to filter out the frequencies greater than half the sampling frequency. Since they are used to prevent aliasing, these analog lowpass filters are known as anti-aliasing filters. *See also* Nyquist frequency.
array	Ordered, indexed list of data elements of the same type.
autoscaling	Ability of scales to adjust to the range of plotted values. On graph scales, this feature determines maximum and minimum scale values.

B

bandedge frequency	The upper and lower cutoff frequencies of an ideal bandpass filter.
bandpass filter	A filter with a single transmission band extending from a lower bandedge frequency greater than zero to a finite upper bandedge frequency.

block diagram	Pictorial description or representation of a program or algorithm. In G, the block diagram, which consists of executable icons called nodes and wires that carry data between the nodes, is the source code for the VI. The block diagram resides in the block diagram window of the VI.
BridgeVIEW	A program development application for real-time process monitoring and control. BridgeVIEW uses the graphical development environment called G.
Butterworth filter	A special kind of filter which has no ripple in the passband or the stopband.

C

calibrator	A controlled source generating a known level of excitation used to calibrate a sensor.
cluster	A set of ordered, unindexed data elements of any data type including numeric, Boolean, string, array, or cluster. The elements must be all controls or all indicators.
coherence	Gives a measure of the degree of linear dependence between two signals, as a function of frequency.
coherent output power spectrum	The coherent output power spectrum gives a measure of what part of the (output) power spectrum is fully coherent with the input signal.
colormap	A method of displaying 3-dimensional data on a 2-dimensional plot using color.
control	Front panel object for entering data to a VI interactively or to a subVI programmatically.
Controls palette	Palette containing front panel controls and indicators.
cross power spectrum	The cross power spectrum of two signals has an amplitude that is the product of the two amplitudes, and a phase that is the difference of the two phases.

D

DAQ	*See* data acquisition.
data acquisition	DAQ. Process of acquiring data, typically from A/D or digital input plug-in boards.
dB	Decibels—a logarithmic unit for measuring ratios of levels. If the levels are specified in terms of power, then 1 dB = 10*log10 (P/Pr) where P is the measured power and Pr is the reference power. If the levels are specified in terms of amplitude, then 1 dB = 20*log10(A/Ar) where A is the measured amplitude and Ar is the reference amplitude.
default	A preset value. Many VI inputs use a default value if you do not specify one.
default input	The default value of a front panel control.
DFT	Discrete Fourier Transform—determines the amplitude and phase of frequency components present in a time domain digital signal.
DIO	Digital input/output

E

engineering unit (EU)	Term of data measurement, such as Pa, m/s^2, g, and so on.
equal confidence	Special exponential averaging mode used for fractional-octave analysis. For equal confidence the time constant for each band is set individually so that the relative confidence in the measurement is equal across all the bands. There is a 68% probability that the results will be within +/- the specified confidence level of the true mean value.
equivalent continuous level (Leq)	The energy average level of a signal over a given time interval.
error message	Indication of a software or hardware malfunction, or an unacceptable data entry attempt.

error structure	The LabVIEW error structure consists of a Boolean status indicator, a numeric code indicator, and a string source indicator.
exponential averaging	Time-averaging technique that gives recent data more importance than older data.

F

Fast	Exponential averaging using a time constant of 125 ms.
FFT	Fast Fourier Transform—an efficient and fast method for calculating the Discrete Fourier Transform. The number of samples is usually constrained to be a power of two. The Fast Fourier Transform determines the amplitude and phase of frequency components present in a time domain digital signal.
FFT block size	The number of samples used to compute an FFT. This number is usually constrained to be a power of two.
FFT lines	The number of FFT lines is related to the FFT block size. Theoretically the number of lines is half of the block size, but it is practically reduced to 80% of that value due to the anti-aliasing filter. For example, a 400 lines FFT is based on a block size of 1024 points.
filter bank	A group of filters.
filtering	A type of signal conditioning that allows you to modify the frequency content of a signal.
fractional-octave	The interval between two frequencies, one of which is a fractional power of two of the other.
frequency response function	Represents the ratio of output-to-input in the frequency domain, and fully characterizes stable linear, time invariant systems.
front panel	Interactive user interface of a VI. Modeled after the front panel of physical instruments, it is composed of switches, slides, meters, graphs, charts, gauges, LEDs, or other controls or indicators.
function	Built-in execution element, comparable to an operator, function, or statement in a conventional language.
Functions palette	Palette containing block diagram structures, constants, communication features, and VIs.

G

g	Unit for measuring acceleration. One g = 9.81 m/s^2, the acceleration due to gravity at the surface of Earth.
G	Graphical programming language used in LabVIEW and BridgeVIEW.
gain	The amplification or attenuation of a signal.
graph	A 2D display of one or more plots. A graph receives and plots data as a block.

H

Hz	Hertz. Cycles per second.

I

IC	Initial Condition(s).
IEC	International Electrotechnical Commission.
Impulse	Exponential averaging using a time constant of 35 ms, if the signal is rising and 1,500 ms if the signal is falling.
indicator	Front panel object that displays output.
input limits	The upper and lower voltage inputs for a channel. You must use a pair of numbers to express the input limits.
intensity map/plot	A method of displaying three dimensions of data on a 2D plot with the use of color.

L

LabVIEW	Laboratory Virtual Instrument Engineering Workbench. Program development application based on the programming language G used commonly for test and measurement purposes.
Leq	*See* equivalent continuous level (Leq).
library	*See* VI library.

linear averaging — Time-averaging technique that weights all data in the average equally.

LLB — LabVIEW VI Library.

M

microphone — Sensor used to convert sound pressure variations into an electrical signal, usually when the acoustic medium is air.

midband frequency — The center frequency of a bandpass filter, defined as the geometric mean of the bandedge frequencies.

N

noise — Any unwanted signal. Noise can be generated by internal sources such as semiconductors, resistors, and capacitors, or from external sources such as the AC power line, motors, generators, thunderstorms, and radio transmitters.

nominal frequency — Rounded midband frequency for the designation of a particular fractional-octave filter. This term is used by the IEC standards, but nominal frequencies are identical to the preferred frequencies defined in the ANSI standards.

nonstationary signal — Signal whose frequency content changes within a captured frame.

numeric controls and indicators — Front panel objects used to manipulate and display input and output numeric data.

Nyquist frequency — Half the sampling frequency. Any analog frequency component above the Nyquist frequency will, after sampling, be converted (aliased) to a frequency below the Nyquist frequency. *See also* aliasing and anti-aliasing filter.

O

octave — Refers to the interval between two frequencies, one of which is twice the other. For example, frequencies of 250 Hz and 500 Hz are one octave apart, as are frequencies of 1 kHz and 2 kHz.

order	Harmonic of the rotational speed of rotating machinery. As an example, in the case of the shaft rotating at 6000 rpm, the first order component occurs at a frequency of 100 Hz (6000/60), whereas the third order component would occur at a frequency of 300 Hz.

P

Pa	Pascal. International unit of pressure.
peak hold	Peak detection process retaining the maximum value of a signal.
palette	Display of icons that represent possible options.
pistonphone	Microphone calibrator generating a known sound pressure level, typically at a certain reference frequency.
preferred frequency	Rounded midband frequency for the designation of a particular fractional-octave filter. This term is used by the ANSI standards, but preferred frequencies are identical to the nominal frequencies defined in the IEC standards.
pregain	Any gain applied to a signal by an external device (amplifier, preamplifier, signal conditioning, and so on) before the data acquisition board.

R

Reference sound pressure	A reference pressure of 20E-6 Pa. This reference pressure was conventionally chosen to correspond to the quietest sound at 1,000 Hz that the human ear can detect.
RMS	Root Mean Square.
RMS averaging	RMS averaging is used to average the energy (or power) of a signal. RMS averaging reduces signal fluctuations but not the noise floor. RMS quantities of single-channel measurements have zero phase. RMS averaging for dual-channel measurements is defined in such a way to preserve important phase information.
rpm	Revolutions per minute.

S

s	Seconds.
sampling frequency	The rate at which a continuous waveform is digitized.
sensor	A device that converts a physical stimulus (such as force, sound, pressure, motion) into a corresponding electrical signal.
settling time	The amount of time required for a signal to reach its final value within specified limits.
Shannon Sampling Theorem	States that to properly sample a signal, the signal must not contain frequencies above the Nyquist frequency.
Slow	Exponential averaging using a time constant of 1,000 ms.
sound pressure level	In decibels, 20 times the logarithm to the base 10 of the ratio of the sound pressure, in a stated frequency band, to the reference sound pressure.
spectral leakage	A phenomenon whereby the measured spectral energy appears to leak from one frequency into other frequencies. It occurs when a sampled waveform does not contain an integral number of cycles over the time period during which it was sampled. The technique used to reduce spectral leakage is to multiply the time-domain waveform by a window function. *See also* windowing.
STFT	Short-Time Fourier Transform.
string controls and indicators	Front panel objects used to manipulate and display input and output text.
subVI	VI used in the block diagram of another VI; comparable to a subroutine.

T

tacho	*See* tachometer.
tachometer	Device used to measure the rotational speed of a rotating part.
third-octave	Ratio between two frequencies, equal to $2^{1/3}$.

V

V	Volts.
vector averaging	Computes the average of complex quantities directly, that is, the real and imaginary parts are averaged separately. Vector averaging actually eliminates noise from synchronous signals and usually requires a trigger.
VI	*See* virtual instrument (VI).
VI library	Special file that contains a collection of related VIs for a specific use.
virtual instrument (VI)	Program in LabVIEW or BridgeVIEW; so-called because it models the appearance and function of a physical environment.

W

waterfall	A 3-dimensional plot displaying the amplitude of spectral components as a function of both time and frequency. The frequency spectrum is displayed as a curve for each specified time instant. Several such curves (for different time instants) are displayed simultaneously.
wavelet analysis	Unlike Fourier analysis, in which the signal is compared with sine and cosine functions that last for an infinite time, the so-called wavelet analysis compares the signal with time-limited pieces of waveforms known as wavelets. Because the wavelets are concentrated in a short time period, the resulting analysis gives the frequency information, and provides the corresponding time information.
weighting filter	Filter used to reproduce the varying sensitivity of the human ear to sound at different frequencies. Originally, A-weighting was intended to represent the varying sensitivity of the ear to sound pressure levels ranging between 40 and 60 dB ref 20E-6 Pa. Subsequently, B-weighting and C-weighting were developed to represents the varying sensitivity of the ear over higher sound pressure level ranges.
white noise	Noise that has the same power spectral density at all frequencies. As an example, the average power of white noise in a 100 Hz bandwidth between 300 Hz and 400 Hz, is the same as the average power of white noise in the 100 Hz bandwidth between 10,000 Hz and 10,100 Hz.

window	*See* windowing.
windowing	Technique used to reduce spectral leakage by multiplying the time-domain waveform by a window function. The process of windowing reduces the amplitudes of discontinuities at the edges of a waveform, thereby reducing spectral leakage. If the waveform contains an integral number of cycles, there is no spectral leakage. *See also* spectral leakage.

Index

G

I

L

M

N

O

P

R

S

T

U

V

W