Chapter D: The Waveform Processing Module

Datapac 2K2 User's Manual, Ver 3

Chapter D: The Waveform Processing Module

Document Release Date: 5/07/2002

D-1. Introduction

Analog waveform processing, sometimes called "signal processing" or "signal conditioning", refers to the act of modifying an analog signal by applying one or more mathematical functions to it. The processing options that Datapac 2K2's Waveform Processing Module provides are briefly described in Table D-1. Up to 10 different processing options can be applied simultaneously to each channel of a data file, and the processing options applied to each channel can be different.

One of Waveform Processing Module's most unique qualities is that it allows you to apply the selected processing operations "*on-the-fly*", meaning that you can use the processed versions of your analog signals in subsequent display, event selection, and data analysis tasks *without permanently modifying the original signals*. Consequently you can return to the original, unprocessed signals at any time, and you can elect to add, change, or eliminate individual processing operations at will. Furthermore, the Waveform Processing Module is designed to allow the processing on the fly feature to remain active at all times even when the module itself is closed. This feature makes it possible to automatically process every data file you open with no additional effort whatsoever.

Table D-1. Available Processing Options.

Copy Channel: Copies the signal in one channel to another channel.

Combine Channels: Adds, subtracts, multiplies, or divides the signals in two channels and copies the result to another channel.

Rectify: Calculates the absolute value of the difference between a selected "baseline" level and the original amplitude of each data point in a signal (full wave rectification).

Gain by Magnification: Multiplies the original amplitude of each point in a signal by a selected constant.

Gain By Peak-to-Peak: Allows you to establish new minimum and maximum peak values for a signal and adjusts the amplitude of each data point in the signal accordingly.

Offset: Adds a selected constant to the amplitude of each data point in a signal.

Invert: Inverts the amplitude of a signal around a selected threshold level.

Square: Squares the amplitude of each data point in the signal.

Square Root: Obtains the square root the amplitude of each data point in the signal.

Log: Obtains the base 10 logarithm of the amplitude of each data point in the signal.

Ln: Obtains the natural logarithm of the amplitude of each data point in the signal.

Sin: Obtains the sine of the amplitude of each data point in the signal, reported in either degrees or radians.

Cos: Obtains the cosine of the amplitude of each data point in the signal, reported in either degrees or radians.

Tan: Obtains the tangent of the amplitude of each data point in the signal, reported in either degrees or radians.

Arc Tan: Obtains the arc tangent of the amplitude of each data point in the signal, reported in either degrees or radians.

Linear Smoothing: Replaces the original amplitude value of each data point in a signal by the mean amplitude obtained for a selected interval symmetrically placed around each data point.

RMS Smoothing: Replaces the original amplitude value of each data point in a signal by the square root of the mean of the squared amplitude values obtained for the data points within a selected interval symmetrically placed around each data point.

Passive Demeaning: Adjusts the offset of a signal by setting its average amplitude to zero.

Dynamic Demeaning: Dynamically adjusts the offset of a signal by calculating the average amplitude of a moving baseline interval of selected duration and setting it to zero.

Integrate: Replaces the original amplitude of each data point in a signal by the amplitude of that point plus the sum of the amplitudes of all previous data points in the same channel. The integral may be reset to zero on elapsed time or on positive or negative slope transitions across a selected threshold level.

Differentiate: Replaces the original amplitude value of each data point in a signal with the slope of the regression line calculated for an interval of selected duration symmetrically placed around each data point.

High Pass Filter (Butterworth and FIR): Eliminates the frequency components of a signal below the selected cut-off frequency.

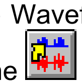
Low Pass Filter (Butterworth and FIR): Eliminates the frequency components of a signal above the selected cut-off frequency.

Band Pass Filter (Butterworth and FIR): Eliminates the frequency components of a signal above and below the selected frequency band.

Band Stop Filter (Butterworth and FIR): Eliminates the frequency components of a signal within the selected frequency band.

Notch Filter (Butterworth and FIR): Eliminates the frequency components of a signal immediately surrounding the selected center frequency.

D-2. Using the Waveform Processing Window

The **Waveform Processing Window** is the first window that appears when you enter the waveform processing module. It serves as the main control panel for initializing and adjusting signal processing operations and selecting the conditions under which they are applied. An example of the Waveform Processing Window is provided in Figure D-1. To access the Waveform Processing Window select the **Process** option in the main window's command bar or select the  icon in the main window's tool bar.

The Waveform Processing Window is divided into 3 sections. The top section provides space to enter a title, along with a series of check boxes that are used when the selected processing operations are applied "on-the-fly". These check boxes let you decide whether to use the processed or the raw versions of your signals for different tasks at the same time. See Section D-2.1 and D-2.2 for additional details.

The middle section contains a **Channels** list box along the left edge which lists all of the active analog channels in the open data file. When the highlight is placed on a channel in the list box the processing operations currently associated with it are reported to the right of the list box. Each of the processing operations are attached to one of the ten **Levels**. And since there are ten levels, you can specify up to 10 processing operations to any individual channel.

The bottom section of the Waveform Processing Window consists of two rows of buttons. The **Copy Last Channel** button lets you copy the processing parameters defined for one channel to another. The **Clear Channel** button lets you eliminate all of the processing parameters defined for the currently highlighted channel. Similarly, the Clear All button allows you to eliminate all of the processing parameters defined for all channels. Additional information is provided in the remaining sections of this chapter.

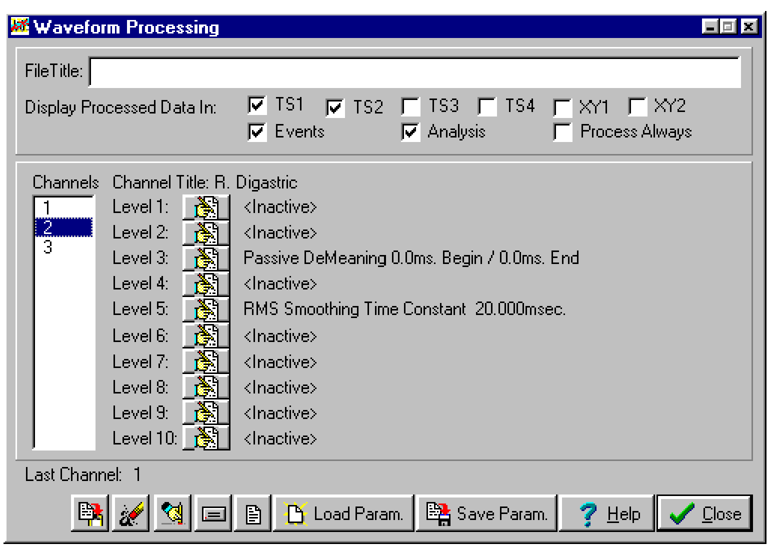


Figure D-1. The Waveform Processing Parameters Window.

Table D-2. Options of the Waveform Processing Window

Top Section:

File Title: Use this option to describe the set of processing parameters you have established. When the parameters are saved to a parameter file the title is saved as well, where it can be used as a long file name.

TS1, TS2, TS3, TS4, XY1, and **XY2 Check Boxes:** These six check boxes respectively determine whether processed data or unprocessed data appear in the four available time series displays and the two X-Y plot displays. When a given box is checked the *processed* versions of the analog signals are passed to the corresponding display window. When a box is cleared the *raw* (unprocessed) versions of the analog signals are passed to the corresponding display. For example, if the box labeled **TS1** is checked but the box labeled **TS2** is not, then the processed versions of the analog signals will be displayed whenever you open the Data Display 1 window but the unprocessed versions will be displayed whenever you open the Data Display 2 window.

Events Check Box: Determines whether the processed or raw (unprocessed) versions of your signals are used in all event selection and editing operations and displays. When checked, the program uses the processed signals to select events, to display within the Display Editing and Event Editing displays, and all other event editing operations. When clear the unprocessed versions of your signals are used.

Analyses Check Box: Determines whether the processed data are passed to all data analysis activities. For example, check this check box if you wish to use processed signals to perform signal averaging, histogram, spreadsheet, power spectrum analysis, or any other analysis activity. Clear the check box if you wish to use the original, raw signals for such activities.

Process Always Check Box: When this check box is clear the selected processing operations are deactivated when you close the Waveform Processing Window. Thus the signals are no longer processed until you reopen the window. In contrast, when the check box is checked the selected processing operations to remain in effect thereafter, even after the Processing Parameter Window is closed. In fact, the operations remain in effect even after you close and rerun Datapac 2K2 itself.

☞ The Waveform Processing Window -- like all other windows -- is considered still open even when it is minimized.

Middle Section:

Channels: This list box lists all of the active channels in the data file. When a channel is highlighted its title and the processing operations currently assigned to it are reported to the right of the list box. Thus, when you move the highlight to a different channel the information will change accordingly.

Channel Title: This line reports the title of the channel highlighted in the Channels list box.

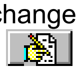
Level 1 - Level 10: These lines report the processing operation currently assigned to each of the processing levels. Since there are 10 levels, it is possible to assign up to 10 processing operations at one time to any channel. A level is labeled as **<Inactive>** when no operation is assigned to it. Processing operations can be assigned in any sequence, but they are applied to the signal according to the order of the levels that they occupy, from Level 1 to Level 10. To add, eliminate, or change a processing operation currently assigned to a particular level, select the corresponding  (**Modify**) button. For details on each option, see Sections B-9.1 through B-9.24.

Table D-2 (Continued)

Bottom Section:

Last Channel: This item indicates the last channel highlighted before the current one. More importantly, it indicates the channel whose assigned processing operations are copied to the currently highlighted channel when the (**Copy Last Channel)** button is selected.

(Copy Last Channel): Use this button to copy the list of assigned processing parameters from the last highlighted channel to the currently highlighted channel. The last highlighted channel is indicated above the button.

(Clear Channel): Use this button to clear (erase) all of the processing operations the currently assigned to the highlighted channel.

(Clear All): Use this button to clear all of the processing operations currently established for all channels.

(Update File): Use this button to permanently apply all of the processing operations to the data file.

☞ **Exercise extreme caution when updating a data file**. See Section D-7 for details.

(View Parameters): Select this button to open a window containing a table reporting all of the processing operations currently assigned to all channels. The table can be printed as well.

Load Parameters: Use this option to load (retrieve) the contents of a processing parameter file.

Save Parameters: Use this button to save all of the currently established processing operations to a file for future use.

Close: Use this button to close the Waveform Processing Window. When the **Process Always** check box is cleared, closing the window will also turn off the assigned processing operations. However, when the Process Always check box is checked, the assigned processing operations remain in effect.

D-2.1. Using Processed and Unprocessed Data for Different Tasks

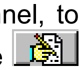
Note in the example shown Figure D-1 that there are a number of check boxes in the top section of the processing parameters window. The check boxes allow you to determine whether to employ processed or unprocessed versions of your data in various activities. Specifically, the **TS1, TS2, TS3, TS4, XY1,** and **XY2** check boxes respectively determine whether processed data or unprocessed data appear in the four available time series displays and the two X-Y plot displays. When a box is cleared the *raw* (unprocessed) versions of the analog signals are passed to the corresponding display. For example, if the box labeled **TS1** is checked but the box labeled **TS2** is not, then the processed versions of the analog signals will be displayed whenever you open the Data Display 1 time series display window but the unprocessed versions will be displayed whenever you open the Data Display 2 window. This ability to pass the results of the processing operations to some display windows but not others allows you to view both the processed and unprocessed versions of your signals on the screen at the same time.

The **Events** and **Analysis** check boxes serve an analogous, but broader, purpose: they allow you to pass the processed or raw (unprocessed) versions of the signal to all event selection and editing tasks, or all data analysis tasks, respectively. Specifically, when the **Events** check box is checked the processed versions of your signals are passed to all event selection and editing tasks. Likewise, then the **Analysis** check box is checked the processed versions of your signals are passed to all data measurement and analysis tasks in the Signal Averaging, Histogram Analysis, Power Spectrum Analysis, Force Plate Analysis, Scientific Spreadsheet, and Spike Sorting modules.

D-2.3. The Process Always Check Box

Although the **Process Always** Check Box is located in the top section of the processing parameters window along with the TS1, TS2, TS3, TS4, XY1, XY2, Events and Analysis check boxes, it serves a different purpose. The Process Always check box determines whether the processing operations remain in effect or not when you close the processing parameters window. Specifically, then the Process Always check box is cleared the selected processing operations are deactivated when you close the processing parameters window. Thus the signals are no longer processed until you reopen the window. In contrast, when the check box is checked the selected processing operations to remain in effect thereafter, even after the Processing Parameter Window is closed. In fact, the operations remain in effect even after you close and rerun Datapac 2K2 itself.

D-3. Adding, Removing or Editing an Individual Processing Option

The same general procedure is performed regardless of whether you wish to add a new processing option to a channel, to remove an option, or to edit (change) an existing option. To do any of these things, select the  **(Modify)** button associated with the corresponding Level. The only difference is the current state of the level that you select. For example, to add a processing option to a channel, select the Modify button associated with a level that is currently reported as **<Inactive>**. To remove or edit an option, select the Modify button associated with the level the option is currently assigned to.

Selecting one of the  (**Modify)** buttons opens a window with a list box that contains all of the available processing options. An example is presented in Figure D-2. Notice that the first option in the list is **Inactive**. This is the option to select if you wish to remove an already selected option. If that is what you want to do, highlight the Inactive option and select the **OK** button to return to the Waveform Processing Window. To add or to edit a processing option, highlight one of the other options and select the **OK** button. Each option (except Inactive) has one or more parameters associated with it. Therefore, when you select an option, a parameter window opens allowing you to enter the parameter values you wish to use. The parameters associated with each option are described in Section E-9.

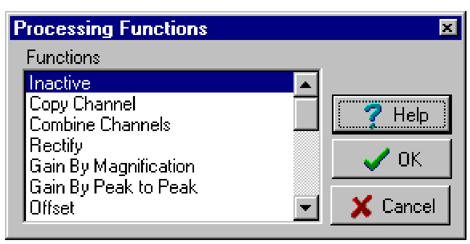


Figure D-2. The list box window used to select desired processing options. The window appears upon selecting any Modify button.

D-4. Ordering Processing Operations

You can assign any processing operation to any of the 10 levels in any desired order. Moreover, it is not necessary to occupy all lower numbered levels with a processing operation before assigning one to a higher numbered level. However, it is important to keep in mind that the **processing operations are applied in ascending order of the levels to which they are assigned**. Thus, if a processing operation is assigned to Level 1 it is performed before the processing operation assigned to the next higher numbered level. Since processing operations are not normally commutative, the order in which the operations are performed may -- and usually will -- affect the outcome.

D-5. Removing All Selected Processing Options

To remove, or to "clear" all of the processing operations currently in effect for all channels, select the  (**Clear All)** button in the Waveform Processing Window. Similarly, to clear all of the processing operations currently in effect for only the currently highlighted channel, select the  (**Clear Channel)** button.

D-6. Copying Processing Options From One Channel to Another

The  (**Copy Last Channel)** button in the Waveform Processing Window is available to copy the processing operations established for one channel to another channel. To use it, first move the highlight in the channel list box to the channel you wish to copy from, then move the highlight a second time to the channel you wish to copy to. Note that as you move the highlight the second time the **Last Channel:** item (reported above the Copy Last Channel Button) is updated to report the number of the channel you highlighted first. This is now considered the "source" channel. Likewise, the channel you highlighted second (which is also the currently highlighted channel) is now considered the "target" channel. Finally, select the **Copy Last Channel** button to copy the selected processing operations from the source to the target channel. For example, to copy the processing operations established for Channel 1 to Channel 2, first highlight Channel 1, then highlight Channel 2. Examine the **Last Channel:** item to make sure it reads "1" (indicating that Channel 1 is the source channel). Then select the **Copy Last Channel** button to copy the processing operations established for Channel 1 to Channel 2.

D-7. Applying Processing Operations Permanently Versus "On-the-Fly"

The processing operations that you specify can be permanently applied to the data file, or they can be applied *on-the-fly.* When the processing operations are permanently applied it means that the data file is permanently modified by replacing the original raw signals with their processed counterparts. In contrast, when the processing operations are applied on-the-fly it means that the processing operations are applied temporarily, while you are working with the file in Datapac 2K2, without permanently affecting the data file. But even though the processing operations are applied temporarily, the processed signals can still be used in data displays, event selection or data analysis tasks. Each method of applying processing operations has its advantages and disadvantages, as described in the following paragraphs.

Time required to perform subsequent tasks:

When processing operations are applied permanently the time required to perform subsequent tasks – such as displays, selecting events, and performing analyses – is reduced. When the processing operations are permanently applied they do not have to be completed prior to performing each subsequent task. In contrast, when processing signals on-the-fly, Datapac 2K2 must complete the processing steps before performing each task you direct it to perform. Depending upon the type of processing operations applied and the number of levels defined, the amount of time required to complete the processing operation may be inconsequential or it may be significant.

Record keeping:

When the processing operations are permanently applied it is not necessary to preserve a record of the processing operations and their sequence in order to return the file to its processed state. When the processing operations are applied on-the-fly it is usually necessary (and certainly recommended) to maintain a record of the processing operations that were applied to each channel in the form of a parameter file. Considering how easy it is to record of the processing operations you employed to a processing parameter file – and equally easy to retrieve the parameter file when needed in the future – most users are not likely to consider this a major advantage. Nonetheless, some users may still prefer to update the file and not have to worry about parameter files at all. See Section D-8 for information about saving and retrieving processing parameter files.

Reversibility:

When you process signals on-the-fly you can add, remove, or change processing operations as desired and at any time as you perform subsequent tasks. In contrast, when the processing operations are applied permanently they cannot be reversed. Therefore, it is a good idea to make a backup copy of your data file before permanently applying the processing operations.

Versatility:

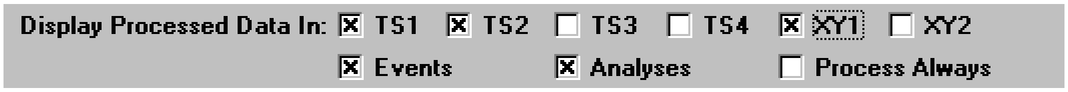
When you process signals on-the-fly you can use the TS1, TS2, TS3, TS4, XY1, XY2, Events and Analysis check boxes to decide what tasks employ processed or unprocessed versions of your signals. For example, you can display the processed signals in one display window and the raw signals in another. Or you can use the processed signals for event selection and editing tasks yet measure and analyze the raw signals. This versatility is not available when the processing operations are applied permanently. See Sections D-2.2 and D-7.1 for discussions on the use of the TS1, TS2, TS3, TS4, XY1, XY2, Events and Analysis check boxes.

Resolution:

During its operation, Datapac 2K2 performs all computations of analog signal amplitude in floating point notation, allowing amplitude values to be internally maintained in high resolution over an essentially limitless range. When performing the processing operations on-the-fly the system can take advantage of the higher resolution and extended range. In contrast, when the processing operations are permanently applied the program is required to conform to the limitations imposed by the file's original structure. Most of the time the limitations are severe, owing to the fact that most data files are maintained in binary notation, usually 12 bit, with a full scale range of rarely more than +/- 10 volts. As a result, the high resolution floating point values must be rounded off to the nearest available binary value, often resulting in a significant loss of resolution. Moreover, when the internally maintained floating point values exceed the original binary file's full scale range, the equivalent of the closest range boundary must be selected when the data are exported. In consequence the minimum and/or maximum peaks in the signal may be flattened off, or "clipped". These limitations are avoided when the original data file is maintained in floating point, rather than binary notation. In recognition of this fact, Datapac 2K2's Data File Management Module includes a Convert option that can be used to convert any binary data file to a floating point data file.

D-7.1. Applying Processing Operations On-the-Fly

To apply the established processing operations on-the-fly and make them available to subsequent display, event selection, and analysis tasks, all you need to do is to check the desired check boxes in the Waveform Processing Window. The check boxes are located across the top of the Waveform Processing Window, just below the **File Title** box. They are reproduced below -- along with the **Process Always** check box which has a different, but related function:



These check boxes allow you to decide which tasks will subsequently use processed data. Note that there are eight check boxes (not counting the **Process Always** check box, whose purpose is explained later in this section). The boxes labeled **TS1**, **TS2**, **TS3** and **TS4** correspond to the four time series data display windows that you have at your disposal within Datapac 2K2. By checking one or more of these boxes you instruct the program to display the processed version of your data in the corresponding display window. For example, if you check boxes TS1 and TS2 but leave boxes TS3 and TS4 clear (as shown in the above example), then the processed signals will be displayed when you open time series display windows 1 or 2, but the unprocessed signals will be displayed when you open time series display windows 3 or 4. Similarly, boxes **XY1** and **XY2** correspond to the two X-Y Plot display windows, so you can also individually decide whether to use one or the other (or both) of the X-Y plot display windows to display either the processed or the unprocessed versions of your data. Because you can have as many display windows open at the same time as desired it is possible to view both the processed and raw versions of your signals on the screen simultaneously for easy comparison.

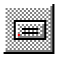
The **Events** and **Analyses** check boxes serve a similar purpose to the six display check boxes just described. However, their effects are somewhat more generally applied. Specifically, when the **Events** check box is checked the processed versions of your waveforms are used in all event selection and editing operations and displays. Similarly, when the **Analyses** check box is checked the processed versions of your waveforms are used in all data analysis operations, including results displays.

Once you have checked the desired check boxes the processed signals become available to the corresponding display, event selection, editing, and analysis tasks. And they will remain available as long as either one of two conditions are met: (1) the Waveform Processing Window remains open[1], or (2) as long as the **Process Always** check box is checked. When the Process Always check box is checked the selected processing operations remain in effect all the time – regardless of whether the Waveform Processing Window is open or not, regardless of how many data files you open, and regardless of how many times you exit and reenter Datapac 2K2. In other words, **the Process Always option provides a way to automatically apply the selected processing operations every time you use Datapac 2K2**. Consequently, if you intend to use the same processing operations on all of the data files that you collect (or open) you will probably find it easiest to check the Process Always check box. Alternatively, if you want to keep the established processing operations active only while working within Datapac 2K2, then turn them off when you exit, clear the Process Always check box but keep the Waveform Processing Window open.

---

[1] As with all windows in DATAPAC 2K2 – or in Windows itself, for that matter – the Processing Window is considered open even when it is minimized. Therefore you can minimize it to get it out of the way and still keep it open.

D-7.2. Applying Processing Operations Permanently

Select the  (**Update File)** button in the Waveform Processing Window to permanently apply the established processing operations to your data file. The time required to update the data file varies according to the size of the file and the number of processing operations in effect. Upon completion the **Process Always** check box is cleared and Waveform Processing Window is automatically closed. These steps are taken in recognition of the fact that when a file is updated, the processed signals replace the original signals and effectively become the new "raw" signals. Thus, if you were to return to the Waveform Processing Module, you must keep in mind that whatever processing operations remain in effect are applied to signals that were already processed.

☞ **Be careful when updating binary data files**: Datapac 2K2 internally maintains signal amplitude values in high resolution floating point notation. When updating a binary file, however, amplitude values must be converted to the native binary format, which is usually significantly lower in resolution and restricted in range. In consequence, the quality of the processed signal may be degraded when it is copied back to the data file -- sometimes substantially so.

D-8. Saving and Retrieving Processing Parameter Files

The Waveform Processing Module makes use of the parameter file named DP2K2.RWP to load and save its default parameters. All of the currently established options are saved to the DP2K2.RWP file when you exit the module, and restored from the file when the module is reopened. Additionally, users can create their own processing parameter files using the **Save Parameters** button. Then, using the **Load Parameters** button, the parameters can later be retrieved to replace the default parameters. Processing parameter files therefore make it easy to copy the processing operations established for one data file to another data file. Additional information regarding loading and saving parameter files can be found in Chapter 1, Sections 1-10 and 1-11.

☞ The **File Title** that is indicated in the Waveform Processing Window is saved to the parameter files along with all of the other processing parameters. When retrieving the file the title can then be used as a long file name.

D-9. The Parameters of Individual Processing Options

Each time a processing option is selected you are presented with a window containing a set of parameters that must be established for it. The parameters associated with each option are described in the following sections.

D-9.1. Copy Channel

This option allows you to copy the analog signal contained in another channel to the currently selected channel. In addition to copying the signal you can elect to add, subtract, multiply, or divide it by a constant value as well as shift the signal in time by applying a delay interval. The parameter window that opens when you select the Copy Channel option is shown in Figure D-3.

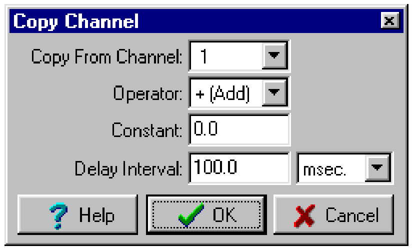


Figure D-3. The parameter window associated with the Copy Channel processing option.

Select the channel you wish to copy from with the **Copy From Channel** parameter. If you have applied processing functions to that channel, those processing functions are applied before the signal is copied to the currently selected channel.

If you wish to add, subtract, multiply, or divide the signal by a constant value, use the **Operator** parameter to select the desired operation, then enter the value of the constant you wish to use in the **Constant** parameter box.

☞ Note that adding or subtracting a constant to or from the signal has the same effect as using the **Offset** processing option. Likewise, multiplying or dividing the signal by a constant has the same effect as using the **Gain By Magnification** processing option.

To shift the signal in time, enter the desired value in the **Delay Interval** box. Use a positive value to shift the signal forward in time (i.e., the copying process is delayed by the indicated value). Likewise, use a negative value to shift the signal backward in time. For example, to construct the illustration shown in Figure D-4 the signal contained in Channel 1 (top trace) was copied to Channel 2 (bottom trace) and a delay interval of 100 milliseconds was applied. Consequently, the signal in Channel 2 is shifted to the left by a distance equal to 100 msec.

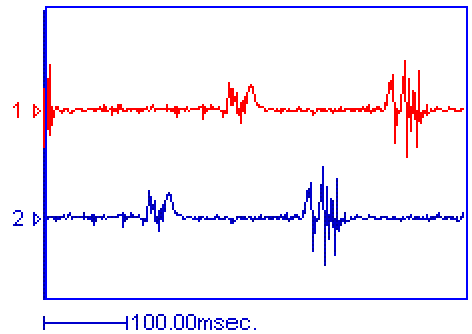
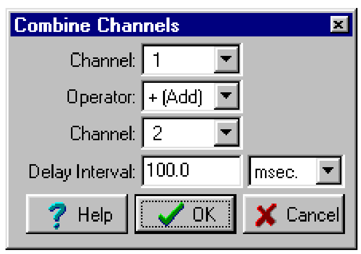


Figure D-4. An example illustrating the operation of the Copy Channel processing option. Channel 2 (lower trace) was copied from Channel 1 (upper trace) with a 100 msec. delay interval applied.

D-9.2. Combine Channels

This option allows you to combine the signals in two different channels and copy them to the currently selected channel. Additionally you can apply a delay interval to the second channel before combining it with the first. The parameter window that opens when you select the Combine Channels option is shown in Figure D-5.



D-5. The parameter window associated with the Combine Channels processing option.

Select the channels containing the signals you wish to combine with the two **Channel** parameters. The two channels that you select can be the same channel. However, **neither channel can be the currently selected channel.** For example, if the currently selected channel is Channel 3, you can copy Channel 1 plus Channel 2 to it, or Channel 1 plus Channel 1, but you cannot copy Channel 1 plus Channel 3, because Channel 3 is the currently selected channel.

To shift the second channel in time, enter the desired value in the **Delay Interval** box. Use a positive value to shift the channel forward in time (i.e., the copying process is delayed by the indicated value). Likewise, use a negative value to shift the channel backward in time. For example, to construct the illustration shown in Figure D-6 the signal in Channel 1 (top trace) was added to the signal in Channel 2 (middle trace). A delay interval of 100 milliseconds was applied to Channel 2. The result was copied to Channel 3 (bottom trace).

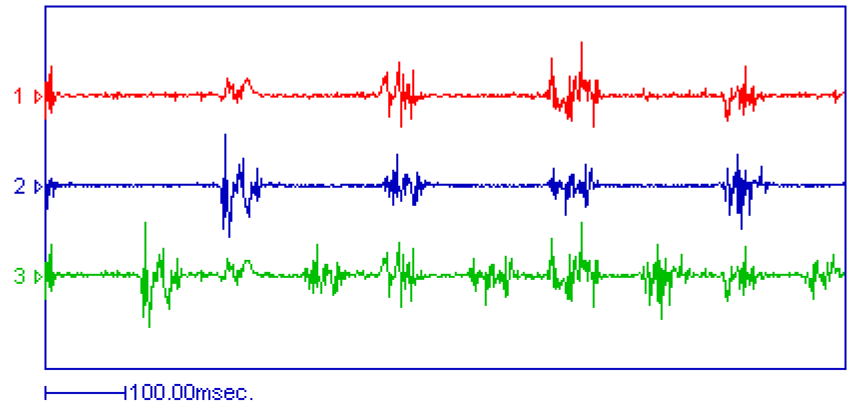


Figure D-6. An example illustrating the operation of the Combine Channels processing option. Channel 1 (top trace) was added to Channel 2 (middle trace) to produce Channel 3 (bottom trace). Note that a 100 msec delay was applied to Channel 2 before the two channels were combined.

D-9.3. Rectify

This option performs a full wave rectification operation on the signal in the selected channel. In this operation amplitude values of the signal below a selected threshold level are converted to values above the threshold, relative to their original difference. An example of the window that opens when you select the Rectify option is shown in Figure D-7. The threshold value that you enter is interpreted in the calibration units associated with the selected channel.

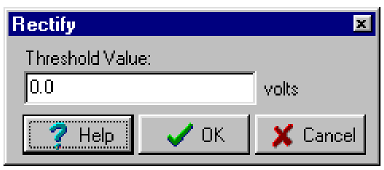


Figure D-7. The parameter window associated with the Rectify processing option.

D-9.4. Gain by Magnification

This option multiplies the signal in the currently selected channel by the entered value. An example of the parameter window that opens when you select the Gain By Magnification option is shown in Figure D-8. The entered value is interpreted in the calibration units associated with the selected channel. The units are indicated to the right of the parameter box.

Figure D-8. The parameter window associated with the Gain By Magnification processing option.

D-9.5. Gain by Peak-to-Peak

This option first scans the currently selected channel to determine the minimum and maximum amplitude values of the signal within the "reference interval", as defined by the **Start** and **End** parameters, then adjusts the amplitude of the entire signal so that the new minimum and maximum values within the reference interval equal the values entered as the **Minimum Peak** and **Maximum Peak** parameters. An example of the parameter window that opens when you select the Gain By Peak-to-Peak option is shown in Figure D-9.

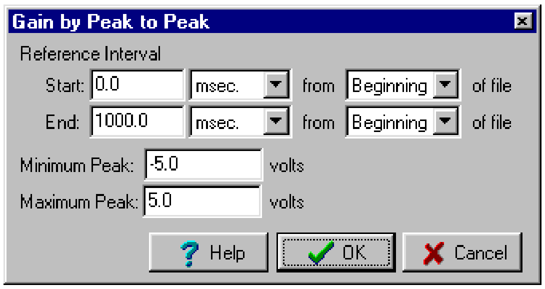


Figure D-9. The parameter window associated with the Gain By Peak-To-Peak processing option.

The **Start** and **End** parameters define the "reference interval" or the interval of the signal that the program scans for the initial minimum and maximum values. Both parameters are composed of three values which collectively define a time boundary. Reading from left to right, the first two values define the time boundary relative to either the beginning or the end of the data file. The first value specifies an operand value and the second value specifies the time units used to interpret it. Finally, the third value determines whether the boundary is measured from the beginning or the end of the data file. In the example presented above, you can see that the reference interval is defined as the first 100 milliseconds of the data file. That is, the reference interval begins 0.00 milliseconds from the beginning of the data file and ends 100 milliseconds from the beginning of the data file.

The **Minimum Peak** and **Maximum Peak** parameters respectively define the values to which the original minimum and maximum amplitude values are converted. The program then adjusts the amplitude of the entire signal accordingly.

The Gain By Peak-to-Peak option provides an easy way to calibrate the amplitude of your signal provided that you know what its minimum and maximum values are supposed to be within a known interval. For example, say you collected an interval of rectified EMG data during maximal contraction and wished to measure EMG amplitude obtained during other tasks relative to the maximum amplitude obtained during the maximal contraction. With the Gain By Peak to Peak option you can define your reference interval as the recorded segment of maximal contraction, then define the Minimum Peak value as 0.00 and the Maximum Peak amplitude as 100.

D-9.6. Offset

This option simply adds the entered value to the signal in the selected channel, thereby adjusting its baseline amplitude level. An example of the parameter window that opens when you select this option is shown in Figure D-10. Use a positive value to increase the baseline amplitude of the signal, or use a negative value to reduce it. Either way, the peak-to-peak amplitude of the signal does not change.

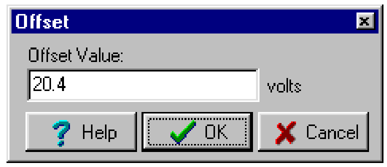


Figure D-10. The parameter window associated with the Offset processing option.

D-9.7. Invert

This option inverts the signal in the currently selected channel around the indicated level. Thus, segments of the signal with amplitudes above the indicated level are now below it and segments of the signal with amplitudes below the indicated level are now above it. An example of the parameter window that opens when you select this option is shown in Figure D-11. The entered value is interpreted in the calibration units associated with the currently selected channel, and they are indicated to the right of the entry box.

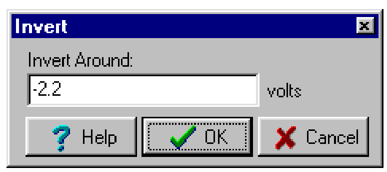


Figure D-11. The parameter window associated with the Invert processing option.

D-9.8. Square, Square Root, Log, Ln, Sin, Cos, Tan and Arc Tan

The Square, Square Root, Log, Ln, Sin, Cos, Tan and Arc Tan processing options have no parameters associated with them. Their functions are described below:

Square: Calculates the square of the original amplitude value of each point.

Square Root: Calculates the square root of the original amplitude value of each point. Negative values are set to 0.00 amplitude.

Log: Calculates the base 10 logarithm of the original amplitude value of each point. Irrational values are set to 0.00 amplitude.

Ln: Calculates the natural logarithm of the original amplitude value of each point. Irrational values are set to 0.00 amplitude.

Sin (Degrees): Calculates the sin of the original amplitude value of each point in units of degrees.

Cos (Degrees): Calculates the cosine of the original amplitude value of each point in units of degrees.

Tan (Degrees): Calculates the tangent of the original amplitude value of each point in units of degrees.

Arc Tan (Degrees): Calculates the arctangent of the original amplitude value of each point in units of degrees.

Sin (Radians): Calculates the sin of the original amplitude value of each point in units of radians.

Cos (Radians): Calculates the cosine of the original amplitude value of each point in units of radians.

Tan (Radians): Calculates the tangent of the original amplitude value of each point in units of radians.

Arc Tan (Radians): Calculates the arctangent of the original amplitude value of each point in units of radians.

D-9.9. Linear Smoothing

This option performs a moving average calculation over the duration of the signal contained in the currently selected channel. More specifically, the program replaces the amplitude value of each data point with the mean amplitude of a user-selected interval centered around that data point. The duration of the interval is called the "time constant". The parameter window that opens when you select the Linear Smoothing option is shown in Figure D-12. The value you enter for the time constant is always interpreted in milliseconds. The time constant's maximum allowed value is constrained by the amount of available memory, but the equivalent of a few thousand sample periods is usually safe. Exceeding the amount of available memory usually results in a General Protection Fault.

Figure D-12. The parameter window associated with the Linear Smoothing processing option.

D-9.10. RMS Smoothing

This option performs a moving *root mean square* (RMS) calculation over the duration of the signal contained in the currently selected channel. More specifically, the program replaces the amplitude value of each data point with the root mean square calculated for a user-selected interval centered around that data point.

The duration of the interval is called the *"time constant"*. The parameter window that opens when you select the RMS Smoothing option is shown in Figure D-13. The value you enter for the time constant is always interpreted in milliseconds. The time constant's maximum allowed value is constrained by the amount of available memory, but the equivalent of a few thousand sample periods is usually safe. Exceeding the amount of available memory usually results in a General Protection Fault.

Figure D-13. The parameter window associated with the RMS Smoothing processing option.

D-9.11. Passive Demeaning

This option obtains the mean amplitude for a user-defined reference interval, taken from the signal contained in the currently selected channel, and then subtracts that value from each data point in the entire signal. The intended result is to set the mean amplitude of the signal to zero. The option works best when the mean amplitude of the defined reference interval represents the signal's baseline amplitude. An example of the parameter window that opens when you select the Passive Demeaning option is shown in Figure D-14.

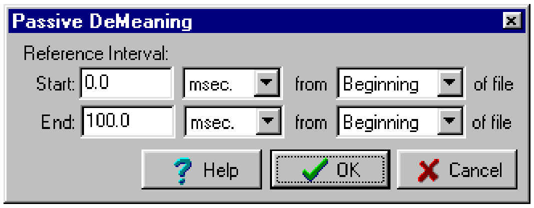


Figure D-14. The parameter window associated with the Passive Demeaning processing option.

The **Start** and **End** parameters define the "reference interval" or the interval of the signal for which the program computes the mean amplitude. Both parameters are composed of three values which collectively define a time boundary. Reading from left to right, the first two values in each case define the time boundary relative to either the beginning or the end of the data file. The first value specifies an operand value and the second value specifies the time units used to interpret it. Finally, the third value determines whether the boundary is measured from the beginning or the end of the data file. In the example presented above, you can see that the reference interval is defined as the first 100 milliseconds of the data file. That is, the reference interval begins 0.00 milliseconds from the beginning of the data file and ends 100 milliseconds from the beginning of the data file. Thus, in this example, the program calculates the mean amplitude of the first 100 milliseconds of the signal and then subtracts that value from the amplitude of each data point in the entire signal.

D-9.12. Dynamic Demeaning

The purpose of this option is to straighten out a wandering baseline and set its average amplitude to zero. It works by dynamically calculating the average amplitude of a moving baseline interval and subtracting the obtained average from every data point in the current interval. The parameter window that opens when you select the Dynamic Demeaning option is shown in Figure D-15.

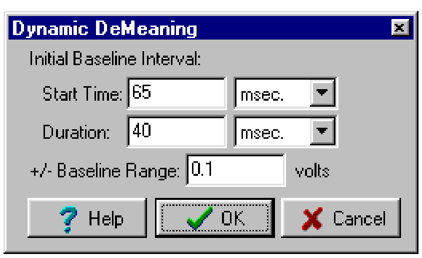


Figure D-15. The parameter window associated with the Dynamic Demeaning processing option.

As you can see from the above example, you are required to enter three sets of parameters: the **Start Time** and **Duration** of the interval of data that you wish to define as the initial baseline interval, and an amplitude range called the **Baseline Range**. The Dynamic Demeaning option begins its operation by calculating the average amplitude of the signal for the interval selected as the initial baseline interval. It then starts at the beginning of the data file and proceeds to subtract that value from every data point in the signal. As it proceeds it also scans the signal for the occurrence of an interval, with the same duration as the initial baseline interval, and where the amplitude of the signal remains within the specified baseline range. When it encounters such an interval it calculates its mean amplitude and uses the newly calculated value to subtract from the subsequent data point. It continues to move through the signal in this way, moving the interval over one data point, calculating a new average value and subtracting it from the next data point, until it encounters a data point whose amplitude value is outside the baseline range (the baseline range is determined relative to the currently calculated average value). At that point the option continues to subtract the average value from each subsequent data point, but it does not recalculate the average value until it encounters another interval where the amplitude of the signal remains within the baseline range.

The preceding description hopefully makes it clear that the selection of an appropriate baseline interval, as well as an appropriate baseline range, are critical to the success of the dynamic demeaning function. In general, it is recommended that you select an interval as close to the beginning of the data file as possible, make it a reasonably short interval, and establish a relatively narrow baseline range. An example of the successful application of the dynamic demeaning option is shown in Figure D-16. The top trace shows a raw signal, before the demeaning option was applied. The bottom trace shows the same signal after application of the dynamic demeaning option. Note that the option served to stabilize the signal's baseline without appreciably affecting the events within it.

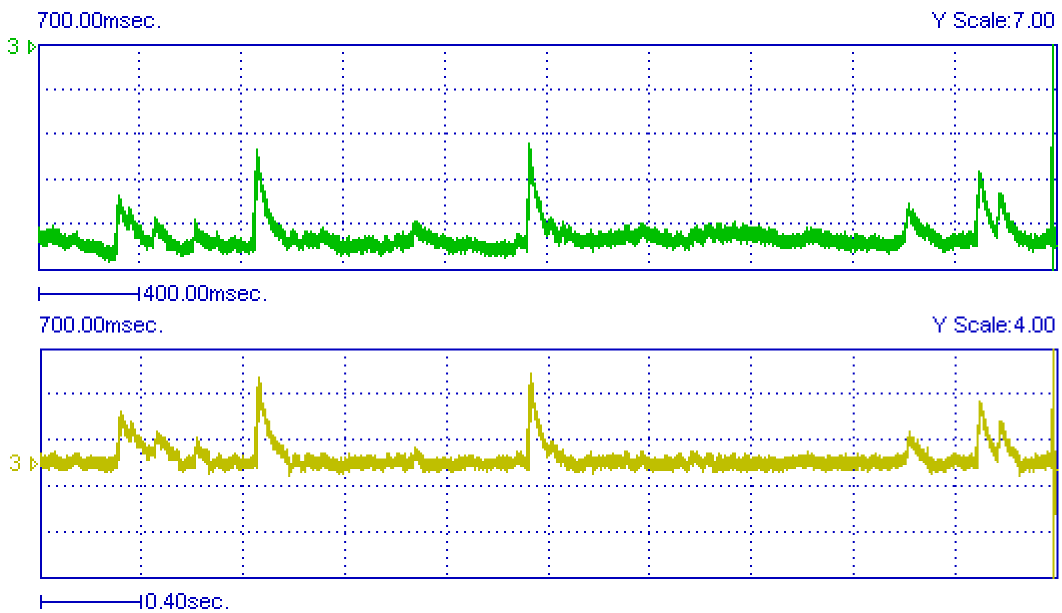


Figure D-16. An example illustrating the operation of the Dynamic Demeaning processing option. The top trace shows the original (raw) signal. The bottom trace shows the same signal after the dynamic demeaning operation was applied.

D-9.13. Integrate

Conceptually speaking, integration is the process of obtaining a measure of the area under a curve. In Datapac 2K2, this concept more specifically refers to the process of summing the amplitude values in a signal over time, until instructed to reset the value of the accumulated amplitude to zero. Integration can be effectively used in several different contexts. For example, integration can be used to obtain velocity measurements from acceleration records, to obtain volume measurements from velocity records, or to obtain a measure of activity over time.

The parameter window that opens when you select the Integrate option is shown in Figure D-17. The four buttons correspond to the four different options available for periodically resetting the integral to zero: **No Reset**, **Reset on Elapsed Time**, **Reset on Overflow**, and **Reset on Threshold**. Only one option can be used at a time.

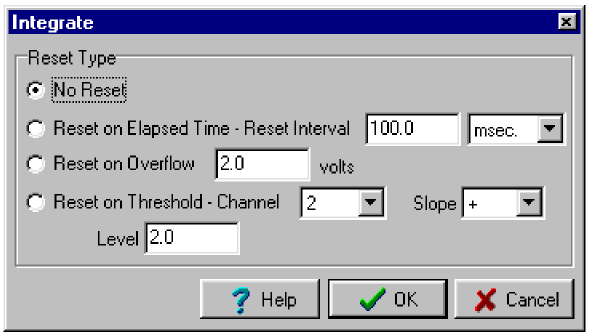


Figure D-17. The parameter window associated with the Integrate processing option.

Figure D-18 illustrates the operation of the **Reset on Threshold** and the **No Reset** options. The top trace in Figure D-18 shows the original, raw signal. The top trace was also used as the *threshold channel* to illustrate the effect of the **Reset on Threshold** option. The threshold channel is the one use to determine when the integration value is reset to zero. The threshold channel can be any channel in the data file *except* the channel that is being integrated[2]. The horizontal line running through the top trace represents the established threshold level. The middle trace in Figure D-18 shows the resulting integrated signal when the Reset on Threshold option is in effect and the **Slope** criterion was set to positive. The arrows above the trace indicate the points where the integrated value was reset to zero in response to positive slope threshold transitions detected in the threshold channel. In contrast, the bottom trace shows the resulting integrated signal when the **No Reset** option is in effect. As the name implies, no reset criterion is used when the No Reset option is in effect. Thus the signal is cumulatively integrated from the beginning to the end of the data file with no interruptions[3].

---

[2] If you wish to use the channel being integrated as the threshold channel you must copy the channel to another channel, then use the original channel as the threshold channel and integrate the copy. For example, to integrate Channel 1 while also using Channel 1 as the threshold channel, use the Copy Channel option to copy Channel 1 to Channel 2 (or any other available channel), then integrate Channel 2 using Channel 1 as the threshold channel.

[3] Generally speaking, the No Reset option is appropriate only when the average amplitude of the signal, measured over the entire duration of the data file, approximates zero. Otherwise the integral values will continue to grow in magnitude.

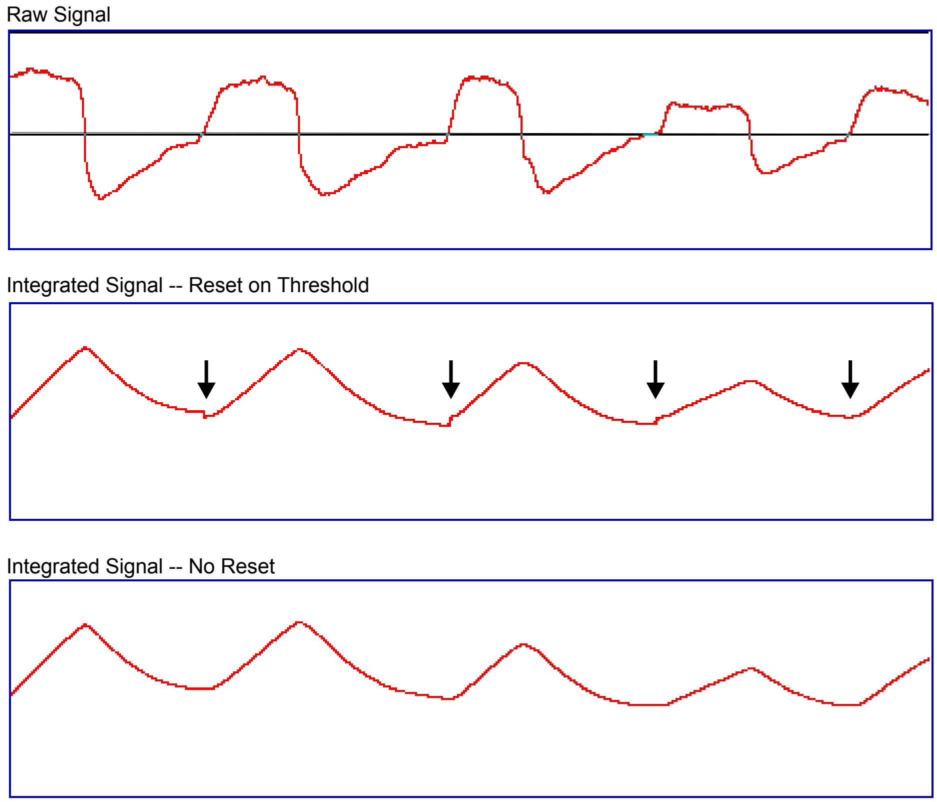


Figure D-18. Example illustrating the effects of the Reset on Threshold and the No Reset options of the Integrate processing function.

Figure D-19 illustrates the operation of the **Reset on Elapsed Time** and the **Reset on Overflow** options. The top trace in Figure D-19 shows the original, raw signal. The middle trace shows the signal after it was integrated using the **Reset on Elapsed Time** option. For the purposes of the example the **Reset Interval** parameter was set to 6 seconds, and the points where the integrated value was reset to zero are indicated with arrows above the middle trace.

The bottom trace in Figure D-19 shows the signal after it was integrated using the **Reset on Overflow** option. The **Overflow** parameter is illustrated as the horizontal line above the trace. Notice that when the established overflow value is reached the integrated value is reset to zero.

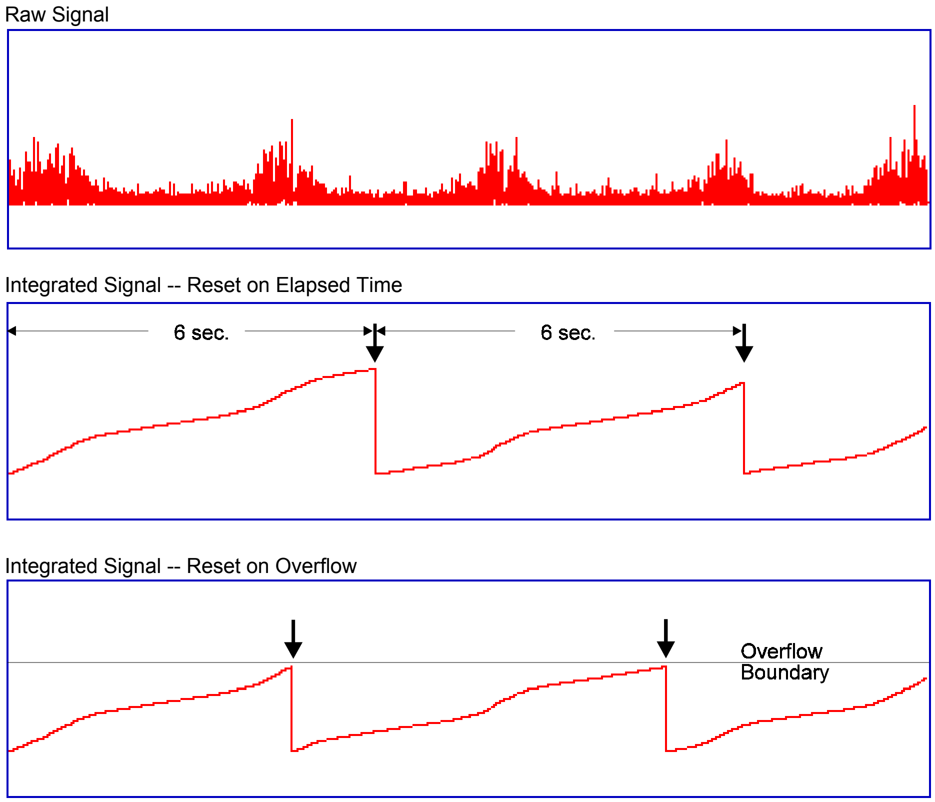


Figure D-19. Example illustrating the effects of the Reset on Elapsed Time and the Reset On Overflow options of the Integrate processing function.

D-9.14. Differentiate

This option obtains the first derivative of the signal in the currently selected channel. The derivative reflects the slope of the regression line calculated for a user-selected interval symmetrically placed around each data point. The duration of the interval is referred to as the "time constant". An example of the parameter window that opens when you select the Differentiate option is shown in Figure D-20.

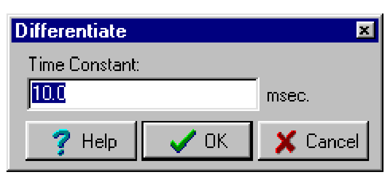


Figure D-20. The parameter window associated with the Differentiate processing option.

An example illustrating the result of the Differentiate operation is shown in Figure D-21. The top trace shows the original raw signal. The bottom trace shows the signal after differentiating with a time constant of 10 milliseconds. As you can see, the differential is useful for illustrating the velocity of change in amplitude within the original signal. Differentiated values are expressed in "units"/millisecond, where "units" equals the calibration units associated with the currently selected channel.

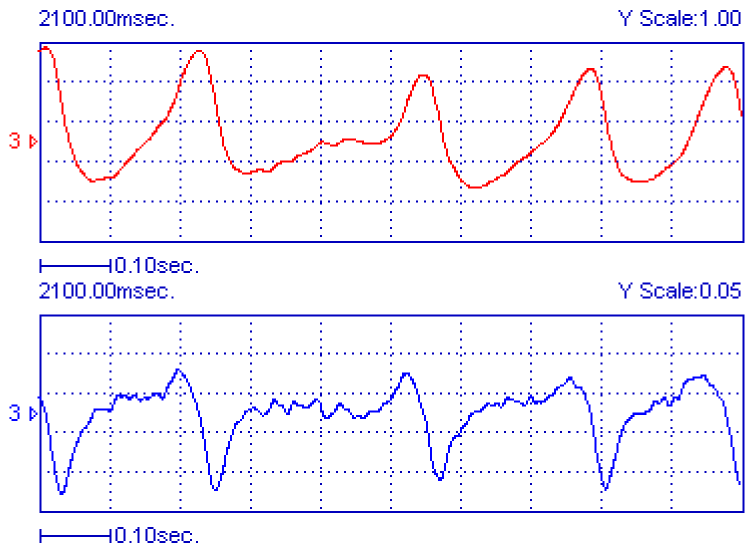


Figure D-21. An example illustrating the operation of the Differentiate processing option. The top trace shows the original (raw) signal. The bottom trace shows the same signal after the differentiation operation was applied.

D-9.15. High Pass FIR Filter

This option passes the signal through a filter that eliminates the low frequency components of the signal. The parameter window that opens when you select the High Pass FIR Filter option is shown in Figure D-22.

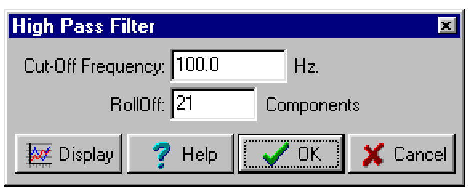


Figure D-22. The parameter window associated with the High Pass FIR Filter processing option.

The filter has two user-adjustable parts: the **Cut-Off Frequency** and the **Rolloff**. The **Cutoff Frequency** refers to the frequency on which the filter is centered. For example, a high pass filter with a cut-off frequency of 100 Hz passes frequencies above 100 Hz and eliminates frequencies below 100 Hz. Shown in Figure D-23 are the response profiles of two high pass filters. A similar display can be obtained by selecting the **Display** button in the parameter window.

The curved line in each display represents the filter's response profile, or the gain with which it passes different frequencies of the spectrum. As you can see, low frequencies are passed with a gain of zero (or nearly so) whereas high frequencies are passed with a gain of 1 (or nearly so). But note that the two filters differ in terms of their cut-off frequency. The one on the left has a cutoff frequency of 150 Hz while the one on the right has a cutoff frequency of 350 Hz.

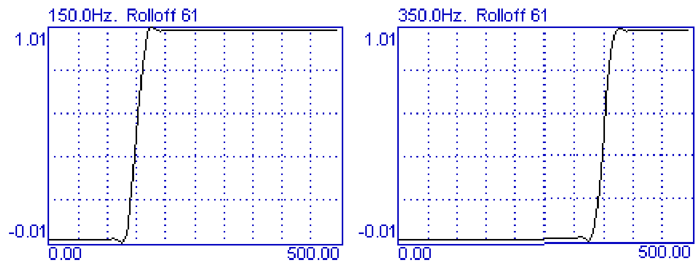


Figure D-23. Response profiles of two high pass FIR filters. The two filters differ only in terms of their cut-off frequencies.

Also notice that neither filter is infinitely efficient. If they were infinitely efficient the response profile curve would appear as a perfectly square edge at the cutoff frequency. In real life, frequency filters are never infinitely efficient. There is always a band of frequencies around the defined cutoff frequency which are not completely eliminated or completely passed. That's why the response profile appears as a curve. Moreover, the response profile curve is always centered on the cutoff frequency. That means that the frequency components of the signal that are exactly on the cutoff frequency are passed with a gain of 0.5. Those slightly below the cutoff frequency are passed with a gain of less than 0.5 but greater than 0.0 while those slightly above the cutoff frequency are passed with a gain of greater then 0.5 but less than 1.0.

The steepness of the response profile curve represents the filter's **Rolloff** characteristics. In Datapac 2K2 a FIR filter's rolloff is determined by the number of components in the frequency algorithm. The greater the number of components the better the rolloff. Figure D-24 compares the response profiles of two additional pass filters. Both of them have a cutoff frequency of 250 Hz, but the one on the left has a rolloff of 21 components while the one on the right has a rolloff of 81 components. Notice that the filter whose response profile is shown on the right is more efficient than -- i.e., has a higher rolloff than -- the filter whose response profile is shown on the left.

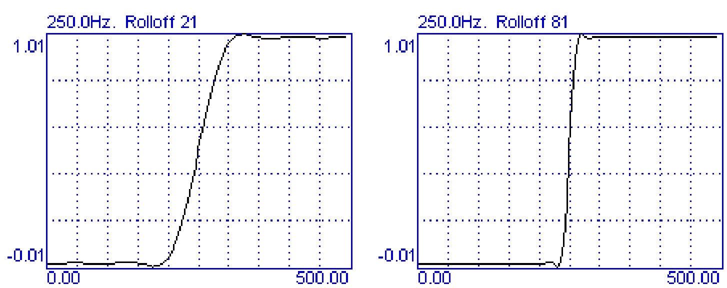


Figure D-24. Response profiles of two high pass frequency filters with the same cut-off frequency but different rolloff efficiencies.

☞ See Section D-10 for a comparison of the FIR and Butterworth filter algorithms.

D-9.16. High Pass Butterworth Filter

This option passes the signal through a filter that eliminates the low frequency components of the signal. The parameter window that opens when you select the High Pass Butterworth Filter option is shown in Figure D-25.

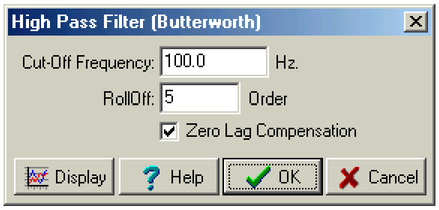


Figure D-25. The parameter window associated with the High Pass Butterworth Filter processing option.

The filter has three user-adjustable parameters: the **Cut-Off Frequency**, the **Rolloff**, and the **Zero Lag Compensation**. The **Cutoff Frequency** refers to the frequency at which the filter passes the signal with an attenuation of -3dB (when zero lag compensation is disabled). For example, an ideal high pass filter with a cut-off frequency of 100 Hz passes frequencies above 100 Hz and eliminates frequencies below 100 Hz. The response profiles of two high pass Butterworth filters are shown in Figure D-26. Notice the curved line in each panel represents the filter's response profile, or the gain with which it passes different frequencies of the spectrum. As you can see, low frequencies are passed with a gain of zero (or nearly so) whereas high frequencies are passed with a gain of 1 (or nearly so). But note that the two filters differ in terms of their cut-off frequency. The one on the left has a cutoff frequency of 150 Hz while the one on the right has a cutoff frequency of 350 Hz.

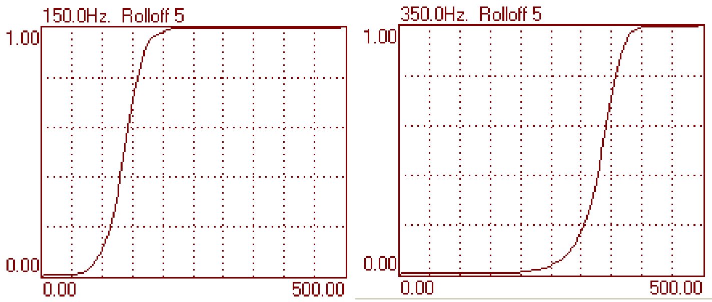


Figure D-26. Response profiles of two high pass Butterworth filters. The two filters differ only in terms of their cut-off frequencies.

Also notice that neither filter is "ideal", or infinitely efficient. If they were ideal the response profile curve would appear as a perfectly square edge at the cutoff frequency. In real life, frequency filters are never infinitely efficient -- there is always a band of frequencies around the defined cutoff frequency which are not completely eliminated or completely passed. That's why the response profile appears as a curve. Notice that in Butterworth filters (with the zero lag compensation feature disabled) the cutoff frequency marks the point where the signal is passed with an attenuation of -3dB. Those slightly below the cutoff frequency are passed with greater attenuation, although not completely eliminated, while those slightly above the cutoff frequency are passed with less attenuation, although still some. The steepness of the curve represents the filter's **Rolloff** characteristics. In the Butterworth filter options a filter's rolloff is determined by the number of orders specified (orders are sometimes called "poles"). The greater the number of orders the better the rolloff. The response profiles of two high pass Butterworth filters are shown in Figure D-27. Both of them have a cutoff frequency of 250 Hz, but the one on the left has a two order rolloff while the one on the right has a 10 order rolloff.

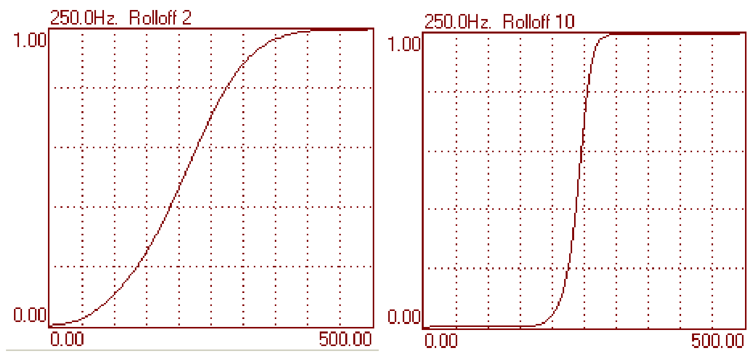


Figure D-27. Response profiles of two high pass Butterworth filters with the same cut-off frequency but different rolloff efficiencies.

Butterworth filters are examples of a class of frequency filters known as infinite impulse response (IIR) filters. One characteristic of IIR filters is that they introduce a phase lag in the resulting signal. In other words, amplitude changes that occur at some frequencies will be delayed in time relative to the original signal. The latency of the lag is also highly non-linear, meaning that different frequencies will be delayed by different latencies. As a result, the signal becomes distorted in time. Fortunately the effect of the filter is symmetric, meaning that exactly the opposite distortion occurs when the filter is applied in the negative time direction when compared to the positive time direction. Thus, when the filter is applied in the postive time direction, then again in the negative time direction, the phase distortions cancel out. We call this process **zero lag compensation**. To enable the zero lag compensation feature, check the **Zero Lag Compensation** check box. We highly recommend that you use this feature when you intend to make latency measurements within or between filtered signals.

Zero lag compensation also has the effect of sharpening the response profile of the filter in addition to shifting its cutoff frequency slightly. See Section D-12 for details.

☞ ***Always check your Butterworth filter's response profile!*** Never use a filter whose profile is not a smooth curve with the endpoints at gains of 0.00 and 1.00. See Section D-11 for details. Also see Section D-10 for a comparison of the FIR and Butterworth filter algorithms.

D-9.17. Low Pass FIR Filter

This option passes the signal through a filter that eliminates the high frequency components of the signal. The parameter window that opens when you select the Low Pass FIR Filter option is shown in Figure D-27.

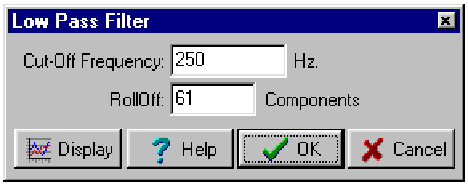


Figure D-27. The parameter window associated with the Low Pass Filter processing option.

Like a high pass FIR filter, a low pass filter has two user-adjustable parts: the **Cut-Off Frequency** and the **Rolloff**. The **Cutoff Frequency** refers to the frequency on which the filter is centered. For example, a low pass filter with a cut-off frequency of 100 Hz passes frequencies below 100 Hz and eliminates frequencies above 100 Hz. Shown in Figure D-28 are the response profiles of two low pass filters. Notice the curved line in each figure represents the filter's response profile, or the gain with which it passes different frequencies of the spectrum. Both filters pass low frequencies with a gain of one (or nearly so) and pass high frequencies with a gain of zero (or nearly so). However, the two filters differ in terms of their cut-off frequency. The one on the left has a cutoff frequency of 150 Hz while the one on the right has a cutoff frequency of 350 Hz.

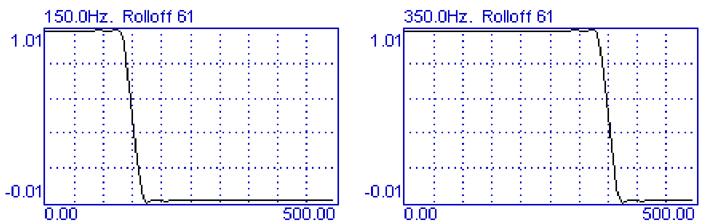


Figure D-28. Response profiles of two low pass FIR filters. The two filters differ only in terms of their cut-off frequencies.

Also notice that neither filter is infinitely efficient. If they were the response profile curve would have a perfectly square edge at the cutoff frequency. In real life, frequency filters are never infinitely efficient -- there is always a band of frequencies around the defined cutoff which are not completely eliminated or completely passed. That's why the response profile appears as a curve. The curve is always centered on the cutoff frequency, reflecting the fact that the frequency components of the signal that are exactly on the cutoff frequency are passed with a gain of 0.5. Those slightly below the cutoff frequency are passed with a gain of less than 1.0 but greater than 0.5 while those slightly above the cutoff frequency are passed with a gain of less than 0.5 but greater than 0.0.

The steepness of the curve represents the filter's **Rolloff** characteristics. In Datapac 2K2 a FIR filter's rolloff is determined by the number of components in the frequency algorithm The greater the number of components the better the rolloff. The response profiles of two low pass filters are shown in Figure D-29. Both of them have a cutoff frequency of 250 Hz, but the one on the left has a rolloff of 21 components while the one on the right has a rolloff of 81 components.

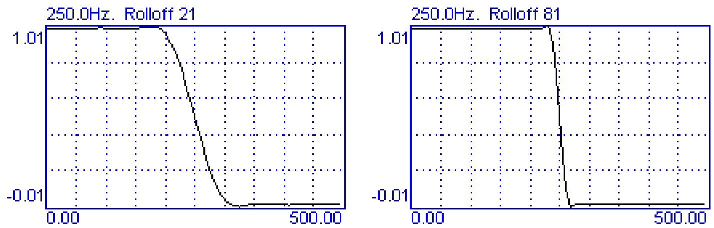


Figure D-29. Response profiles of two low pass frequency filters with the same cut-off frequency but different rolloff efficiencies.

See Section D-10 for a comparison of the FIR and Butterworth filter algorithms.

D-9.18. Low Pass Butterworth Filter

This option passes the signal through a filter that eliminates the high frequency components of the signal. The parameter window that opens when you select the Low Pass Butterworth Filter option is shown in Figure D-30.

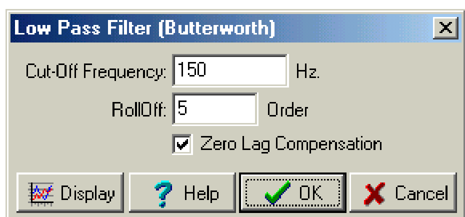


Figure D-30. The parameter window associated with the Low Pass Butterworth Filter processing option.

The filter has three user-adjustable parameters: the **Cut-Off Frequency**, the **Rolloff**, and the **Zero Lag Compensation**. The **Cutoff Frequency** refers to the frequency at which the filter passes the signal with an attenuation of -3dB (when zero lag compensation is disabled). For example, an ideal low pass filter with a cut-off frequency of 100 Hz passes frequencies below 100 Hz and eliminates frequencies above 100 Hz. Shown in Figure D-31 are the response profiles of two low pass filters. Notice the curved line in each figure represents the filter's response profile, or the gain with which it passes different frequencies of the spectrum. As you can see, high frequencies are passed with a gain of zero (or nearly so) whereas low frequencies are passed with a gain of 1 (or nearly so). But note that the two filters differ in terms of their cut-off frequency. The one on the left has a cutoff frequency of 150 Hz while the one on the right has a cutoff frequency of 350 Hz.

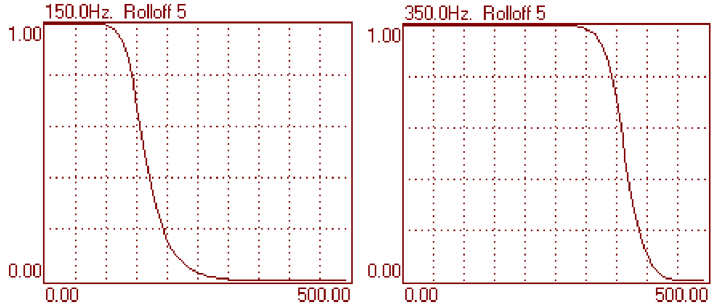


Figure D-31. Response profiles of two low pass Butterworth filters. The two filters differ only in terms of their cut-off frequencies.

Also notice that neither filter is "ideal", or infinitely efficient. If they were ideal the response profile curve would appear as a perfectly square edge at the cutoff frequency. In real life, frequency filters are never infinitely efficient -- there is always a band of frequencies around the defined cutoff frequency which are not completely eliminated or completely passed. That's why the response profile appears as a curve. Notice that in Butterworth filters (with the zero lag compensation feature disabled) the cutoff frequency marks the point where the signal is passed with an attenuation of -3dB. Those slightly above the cutoff frequency are passed with greater attenuation, although not completely eliminated, while those slightly below the cutoff frequency are passed with less attenuation, although still some. The steepness of the curve represents the filter's **Rolloff** characteristics. In the Butterworth filter options a filter's rolloff is determined by the number of orders specified (orders are sometimes called "poles"). The greater the number of orders the better the rolloff. The response profiles of two low pass Butterworth filters are shown in Figure D-32. Both of them have a cutoff frequency of 250 Hz, but the one on the left has a two order rolloff while the one on the right has a 10 order rolloff.

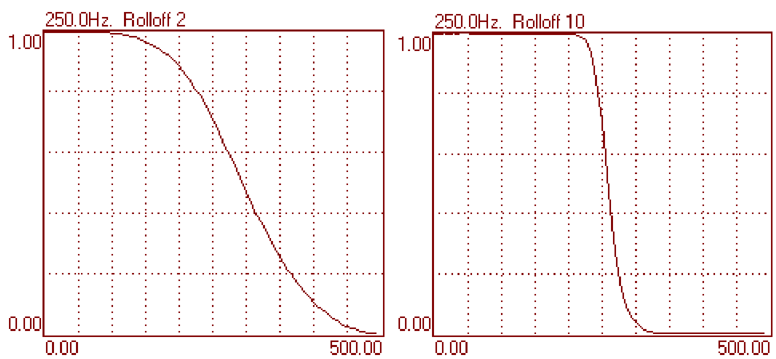


Figure D-32. Response profiles of two low pass Butterworth filters with the same cut-off frequency but different rolloff efficiencies.

Butterworth filters are examples of a class of frequency filters known as infinite impulse response (IIR) filters. One characteristic of IIR filters is that they introduce a phase lag in the resulting signal. In other words, amplitude changes that occur at some frequencies will be delayed in time relative to the original signal. The latency of the lag is also highly non-linear, meaning that different frequencies will be delayed by different latencies. As a result, the signal becomes distorted in time. Fortunately the effect of the filter is symmetric, meaning that exactly the opposite distortion occurs when the filter is applied in the negative time direction when compared to the positive time direction. Thus, when the filter is applied in the postive time direction, then again in the negative time direction, the phase distortions cancel out. We call this process **zero lag compensation**. To enable the zero lag compensation feature, check the **Zero Lag Compensation** check box. We highly recommend that you use this feature when you intend to make latency measurements within or between filtered signals.

Zero lag compensation also has the effect of sharpening the response profile of the filter in addition to shifting its cutoff frequency slightly. See Section D-12 for details.

☞ ***Always check your Butterworth filter's response profile!*** Never use a filter whose profile is not a smooth curve with the endpoints at gains of 0.00 and 1.00. See Section D-11 for details. Also see Section D-10 for a comparison of the FIR and Butterworth filter algorithms.

D-9.19. Band Pass FIR Filter

A band pass filter passes the frequency components in a signal within the specified frequency band and eliminates the components above and below the band. The parameter window that opens when you select the Band Pass FIR Filter option is shown in Figure D-33.

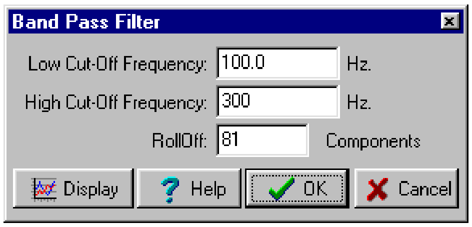


Figure D-33. The parameter window associated with the Band Pass FIR Filter processing option.

In Datapac 2K2 a band-pass FIR filter has three user-adjustable parts: the **Low Cut-Off Frequency**, the **High Cut-Off Frequency**, and the **Rolloff**. The **Low** and **High Cutoff Frequencies** define the boundaries of the frequency band passed by the filter. For example to pass a frequency band of 100 to 300 Hz, the low cut-off frequency must be 100 Hz and the high cut-off frequency must be 300 Hz. The curved line in the example shown in Figure D-34 represents the response profile of a band pass filter with low and high cutoff frequencies set at 100 and 300 Hz, respectively. The height of the curve at any point represents the gain at which the filter passes the corresponding frequency. As you can see, low and high frequencies are passed with a gain of zero (or nearly so), whereas middle frequencies are passed with a gain of one (or nearly so).

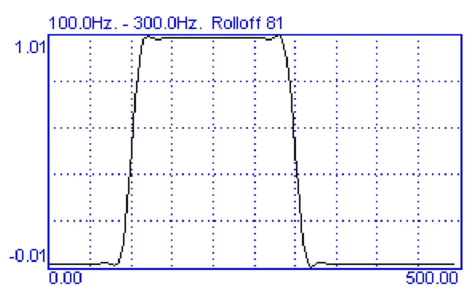


Figure D-34. The response profile of a band pass FIR filter with low and high cutoff frequencies set at 100 Hz and 300 Hz, respectively.

The filter shown in Figure D-34 is not infinitely efficient. If it was the response profile curve would have perfectly square edges at the cutoff frequencies. In real life, frequency filters are never infinitely efficient -- there is always a band of frequencies around the defined cutoff frequencies which are not completely eliminated or completely passed. That's why the response profile appears as a curve. The rising and falling edges of the curve are always centered on the cutoff frequencies, reflecting the fact that the frequency components of the signal that are exactly on the cutoff frequencies are passed with a gain of

0.5. Those slightly above or below the cutoff frequencies are passed with a gain of less than 1.0 but greater than 0.0. The steepness of the curve around the cutoff frequencies represents the filter's **Rolloff** characteristics. In Datapac 2K2 a FIR filter's rolloff is determined by the number of components in the frequency algorithm The greater the number of components the better the rolloff. The response profiles of two band pass filters are shown in Figure D-35. Both of them have the same cutoff frequencies (100 Hz and 300 Hz), but the one on the left has a rolloff of 21 components while the one on the right has a rolloff of 81 components.

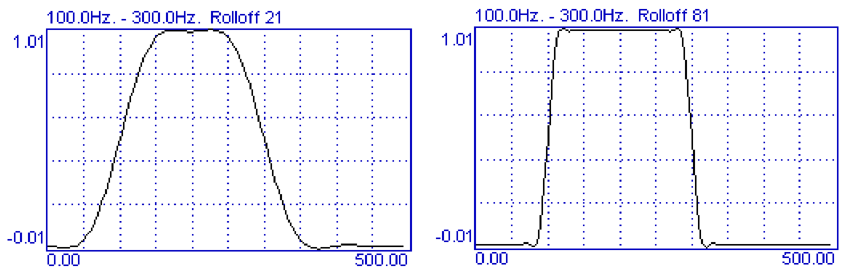


Figure D-35. Response profiles of two band pass FIR filters, both with the same cut-off frequencies but with different rolloff efficiencies.

See Section D-10 for a comparison of the FIR and Butterworth filter algorithms.

D-9.20. Band Pass Butterworth Filter

A band pass filter passes the frequency components in a signal within the specified frequency band and eliminates the components above and below the band. The parameter window that opens when you select the Band Pass Butterworth Filter option is shown in Figure D-36.

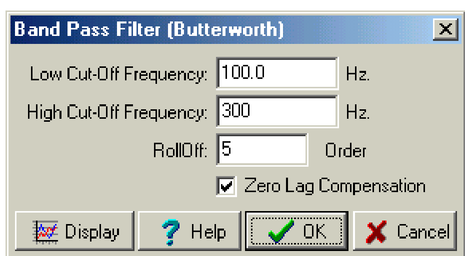
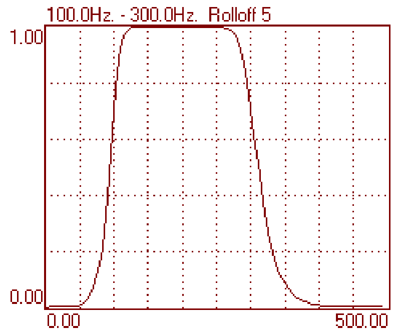


Figure D-36. The parameter window associated with the Band Pass Butterworth Filter processing option.

In Datapac 2K2 a band-pass Butterworth filter has four user-adjustable parts: the **Low Cut-Off Frequency**, the **High Cut-Off Frequency**, the **Rolloff**, and the **Zero Lag Compensation**. The **Low** and **High Cutoff Frequencies** define the boundaries of the frequency band passed by the filter. For example to pass a frequency band of 100 to 300 Hz, the low cut-off frequency must be 100 Hz and the high cut-off frequency must be 300 Hz. The curved line in the example shown in Figure D-37 represents the response profile of such a filter. The height of the curve at any point represents the gain at which the filter passes the corresponding frequency. As you can see, low and high frequencies are passed with a gain of zero (or nearly so), whereas middle frequencies are passed with a gain of one (or nearly so).



D-37. The response profile of a band pass Butterworth filter with low and high cutoff frequencies set at 100 Hz and 300 Hz, respectively.

The filter shown in Figure D-37 is not infinitely efficient. If it was the response profile curve would have perfectly square edges at the cutoff frequencies. In real life, frequency filters are never infinitely efficient -- there is always a band of frequencies around the defined cutoff frequencies which are not completely eliminated or completely passed. That's why the response profile appears as a curve. The points on the rising and falling edges of the curve where the frequencies of the signal are passed with an attenuation of -3dB are the cutoff frequencies (when the zero lag compensation feature is disabled). Frequencies near the cutoff frequencies are passed with greater or lesser attenuation, but not completely passed or eliminated. The steepness of the curve around the cutoff frequencies represents the filter's **Rolloff** characteristics. In the Butterworth filter options a filter's rolloff is determined by the number of orders specified. In the case of band pass and band stop filters, 1 pole is equal to two orders. The greater the number of orders the better the rolloff. The response profiles of two high pass Butterworth filters are shown in Figure D-38. Both of them have cutoff frequencies of 100 and 300Hz, but the one on the left has a two order rolloff while the one on the right has a 10 order rolloff.

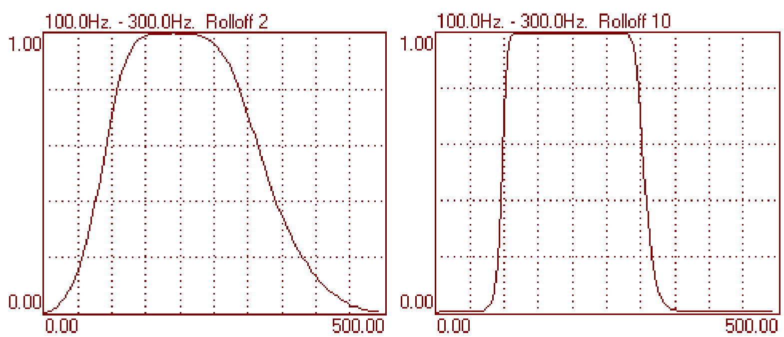


Figure D-38. Response profiles of two band pass Butterworth filters, both with the same cut-off frequencies but with different rolloff efficiencies.

Butterworth filters are examples of a class of frequency filters known as infinite impulse response (IIR) filters. One characteristic of IIR filters is that they introduce a phase lag in the resulting signal. In other words, amplitude changes that occur at some frequencies will be delayed in time relative to the original signal. The latency of the lag is also highly non-linear, meaning that different frequencies will be delayed by different latencies. As a result, the signal becomes distorted in time. Fortunately the effect of the filter is symmetric, meaning that exactly the opposite distortion occurs when the filter is applied in the negative time direction when compared to the positive time direction. Thus, when the filter is applied in the postive time direction, then again in the negative time direction, the phase distortions cancel out. We call this process **zero lag compensation**. To enable the zero lag compensation feature, check the **Zero Lag Compensation** check box. We highly recommend that you use this feature when you intend to make latency measurements within or between filtered signals.

Zero lag compensation also has the effect of sharpening the response profile of the filter in addition to shifting its cutoff frequency slightly. See Section D-12 for details.

☞ ***Always check your Butterworth filter's response profile!*** Never use a filter whose profile is not a smooth curve with the endpoints at gains of 0.00 and 1.00. See Section D-11 for details. Also see Section D-10 for a comparison of the FIR and Butterworth filter algorithms.

D-9.21. Band Stop FIR Filter

A band stop filter eliminates the frequency components in a signal within the specified frequency band and passes the components above and below the band. The parameter window that opens when you select the Band Stop FIR Filter option is shown in Figure D-39.

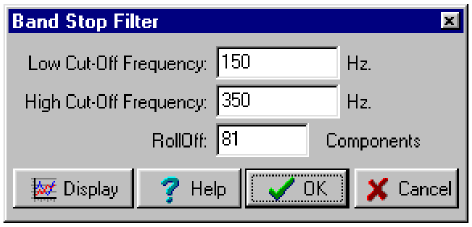


Figure D-39. The parameter window associated with the Band Stop FIR Filter processing option.

In Datapac 2K2 a band-stop FIR filter has three user-adjustable parts: the **Low Cut-Off Frequency**, the **High Cut-Off Frequency**, and the **Rolloff**. The **Low** and **High Cutoff Frequencies** define the boundaries of the frequency band eliminated by the filter. For example to eliminate a frequency band of 150 to 350 Hz, the low cut-off frequency must be 150 Hz and the high cut-off frequency must be 350 Hz. The curved line in the example shown in Figure D-40 represents the response profile of such a filter. The height of the curve at any point represents the gain at which the filter passes the corresponding frequency. As you can see, low and high frequencies are passed with a gain of one (or nearly so), whereas middle frequencies are passed with a gain of zero (or nearly so).

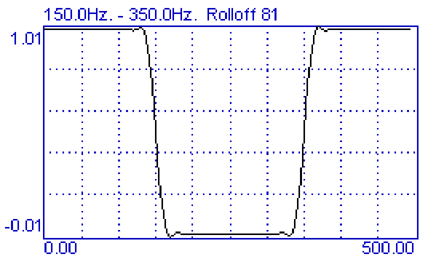


Figure D-40. The response profile of a band stop FIR filter with low and high cutoff frequencies set at 150 Hz and 350 Hz, respectively.

The filter shown in Figure D-40 is not infinitely efficient. If it was the response profile curve would have perfectly square edges at the cutoff frequencies. In real life, frequency filters are never infinitely efficient -- there is always a band of frequencies around the defined cutoff frequencies which are not completely eliminated or completely passed. That's why the response profile appears as a curve. The rising and falling edges of the curve are always centered on the cutoff frequencies, reflecting the fact that the frequency components of the signal that are exactly on the cutoff frequencies are passed with a gain of

0.5. Those slightly above or below the cutoff frequencies are passed with a gain of less than 1.0 but greater than 0.0. The steepness of the curve around the cutoff frequencies represents the filter's **Rolloff** characteristics. In Datapac 2K2 a FIR filter's rolloff is determined by the number of components in the frequency algorithm The greater the number of components the better the rolloff. The response profiles of two band stop FIR filters are shown in Figure D-41. Both of them have the same cutoff frequencies (150 Hz and 350 Hz), but the one on the left has a rolloff of 21 components while the one on the right has a rolloff of 81 components.

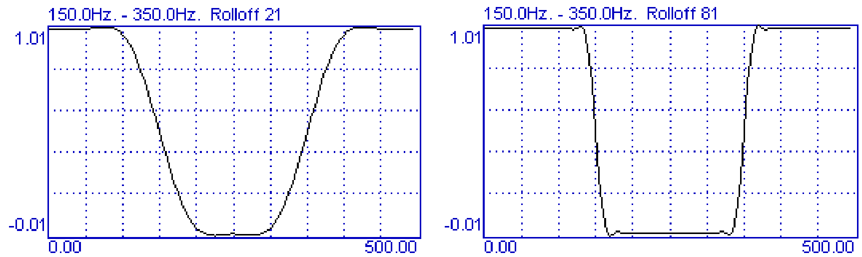


Figure D-41. Response profiles of two band stop FIR filters, both with the same cut-off frequencies but with different rolloff efficiencies.

☞ See Section D-10 for a comparison of the FIR and Butterworth filter algorithms.

D-9.22. Band Stop Butterworth Filter

A band stop filter eliminates the frequency components in a signal within the specified frequency band and passes the components above and below the band. The parameter window that opens when you select the Band Stop Butterworth Filter option is shown in Figure D-42.

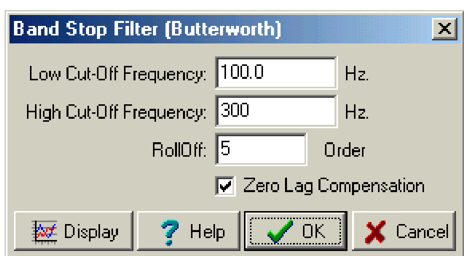


Figure D-42. The parameter window associated with the Band Stop Butterworth Filter processing option.

In Datapac 2K2 a band-stop Butterworth filter has four user-adjustable parts: the **Low Cut-Off Frequency**, the **High Cut-Off Frequency**, the **Rolloff**, and the **Zero Lag Compensation**. The **Low** and **High Cutoff Frequencies** define the boundaries of the frequency band passed by the filter. For example to eliminate a frequency band of 100 to 300 Hz, the low cut-off frequency must be 100 Hz and the high cut-off frequency must be 300 Hz. The curved line in the example shown in Figure D-43 represents the response profile of such a filter. The height of the curve at any point represents the gain at which the filter passes the corresponding frequency. As you can see, low and high frequencies are passed with a gain of one (or nearly so), whereas middle frequencies are passed with a gain of zero (or nearly so).

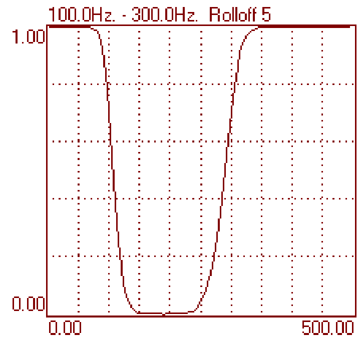


Figure D-43. The response profile of a band stop FIR filter with low and high cutoff frequencies set at 100 Hz and 300 Hz, respectively.

The filter shown in Figure D-43 is not infinitely efficient. If it was the response profile curve would have perfectly square edges at the cutoff frequencies. In real life, frequency filters are never infinitely efficient -- there is always a band of frequencies around the defined cutoff frequencies which are not completely eliminated or completely passed. That's why the response profile appears as a curve. The points on the rising and falling edges of the curve where the frequencies of the signal are passed with an attenuation of -3dB are the cutoff frequencies (when the zero lag compensation feature is disabled). Frequencies near the cutoff frequencies are passed with greater or lesser attenuation, but not completely passed or eliminated. The steepness of the curve around the cutoff frequencies represents the filter's **Rolloff** characteristics. In the Butterworth filter options a filter's rolloff is determined by the number of orders specified. In the case of band pass and band stop filters, 1 pole is equal to two orders. The greater the number of orders the better the rolloff. The response profiles of two high pass Butterworth filters are shown in Figure D-44. Both of them have a cutoff frequencies of 100 and 300Hz, but the one on the left has a two order rolloff while the one on the right has a 10 order rolloff.

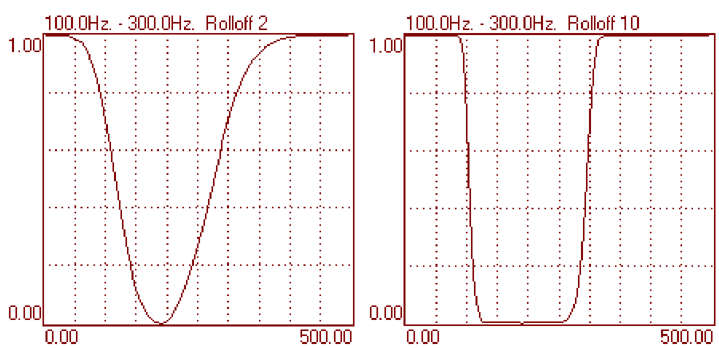


Figure D-44. Response profiles of two band stop Butterworth filters, both with the same cut-off frequencies but with different rolloff efficiencies.

Butterworth filters are examples of a class of frequency filters known as infinite impulse response (IIR) filters. One characteristic of IIR filters is that they introduce a phase lag in the resulting signal. In other words, amplitude changes that occur at some frequencies will be delayed in time relative to the original signal. The latency of the lag is also highly non-linear, meaning that different frequencies will be delayed by different latencies. As a result, the signal becomes distorted in time. Fortunately the effect of the filter is symmetric, meaning that exactly the opposite distortion occurs when the filter is applied in the negative time direction when compared to the positive time direction. Thus, when the filter is applied in the postive time direction, then again in the negative time direction, the phase distortions cancel out. We call this process **zero lag compensation**. To enable the zero lag compensation feature, check the **Zero Lag Compensation** check box. We highly recommend that you use this feature when you intend to make latency measurements within or between filtered signals.

Zero lag compensation also has the effect of sharpening the response profile of the filter in addition to shifting its cutoff frequency slightly. See Section D-12 for details.

☞ ***Always check your Butterworth filter's response profile!*** Never use a filter whose profile is not a smooth curve with the endpoints at gains of 0.00 and 1.00. See Section D-11 for details. Also see Section D-10 for a comparison of the FIR and Butterworth filter algorithms.

D-9.23. Notch FIR Filter

The notch filter option is a special type of band stop filter whose purpose is to eliminate a narrow band of frequencies around a user-selected center frequency. An example of the parameter window that opens when you select the Notch FIR Filter option is shown in Figure D-45.

Figure D-45. The parameter window associated with the Notch FIR Filter processing option.

Unlike the other frequency filtering options within Datapac 2K2, the Notch Filter option does not request a rolloff value. That is because the rolloff of the filter is predetermined at 61 components.

D-9.24. Notch Butterworth Filter

The notch filter option is a special type of band stop filter whose purpose is to eliminate a narrow band of frequencies around a user-selected center frequency. An example of the parameter window that opens when you select the Notch Butterworth Filter option is shown in Figure D-46.

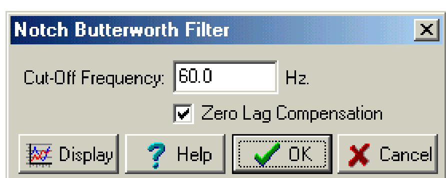


Figure D-46. The parameter window associated with the Notch Butterworth Filter processing option.

The response profile of a 60 Hz Butterworth notch filter is shown in Figure D-47. The Butterworth type of notch filter uses a special algorithm referred to as a bandstop resonator function with a fixed order of 2. The result is a very efficient filter both in terms of its rolloff characteristics and its speed of application. Therefore, in most applications where a notch filter is to be applied, the Butterworth alternative is preferable to the FIR alternative. There are some situations, however, where a wider notch is desired. In that case, use the FIR alternative.

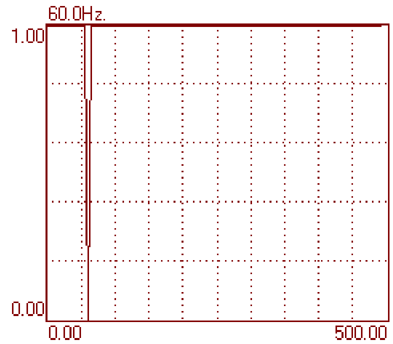


Figure D-47. The response profile of a notch Butterworth filter set at 60 Hz.

Butterworth filters are examples of a class of frequency filters known as infinite impulse response (IIR) filters. One characteristic of IIR filters is that they introduce a phase lag in the resulting signal. In other words, amplitude changes that occur at some frequencies will be delayed in time relative to the original signal. The latency of the lag is also highly non-linear, meaning that different frequencies will be delayed by different latencies. As a result, the signal becomes distorted in time. Fortunately the effect of the filter is symmetric, meaning that exactly the opposite distortion occurs when the filter is applied in the negative time direction when compared to the positive time direction. Thus, when the filter is applied in the postive time direction, then again in the negative time direction, the phase distortions cancel out. We call this process **zero lag compensation**. To enable the zero lag compensation feature, check the **Zero Lag Compensation** check box. We highly recommend that you use this feature when you intend to make latency measurements within or between filtered signals.

D-10. FIR and Butterworth Filters – A Qualitative Comparison

"FIR" is an acronym for finite impulse response, a class of digital filters. More specifically the FIR filters contained within Datapac 2K2 are all symmetric, non-recursive finite impulse response filters incorporating Lanczos' smoothing function to attenuate the rippling around the cutoff frequency that is normally found in this class of filters. Technical details can be found in the account by Hamming (Digital Filters. Prentice-Hall: Englewood Cliffs, NJ, 1977). Butterworth filters are a type of IIR (infinite impulse response) filter. More specifically, the high pass, low pass, band pass, and band stop filter algorithms contained within Datapac 2K2 employ the bilinear transform method. Butterworth notch filters, on the other hand, use a bandstop resonator function with a fixed order of 2.

Both the Butterworth and FIR filters offer adjustable cutoff frequencies and rolloff efficiencies, and both types of filters perform well in eliminating unwanted frequencies from your signals. Consequently, the rationale for selecting one type of filter over another is largely a matter of taste. But there are some differences between them which may be important in some situations. The following paragraphs describe these differences in a qualitative manner.

Filter Efficiency Over the Frequency Spectrum
The response profile of a FIR filter is linear across the spectrum. The response profile of a Butterworth filter is logarithmic. Consider the example shown in Figure D-48. The two panels in A show two high pass FIR filters whose parameters differ only in terms of their cutuff frequency. Likewise the two panels in B show two Butterworth filters whose parameters differ only in terms of their cutuff frequency. Note that the response profiles of the two FIR filters in A are identical even though they are positioned at different ends of the frequency spectrum. In contrast the response profiles of the two Butterworth filters in B differ markedly depending upon where they are positioned in the spectrum. A Butterworth filter is more efficient when its cutoff frequency is located at the low end of the spectrum.

It is important to note, of course, that in the example shown in Figure D-48 we are plotting frequency on a linear scale. If we used a logarithmic scale then the response profiles of the Butterworth filters would appear identical and the response profiles of the FIR filters would not. Specifically, when plotted on a logarithmic scale, a FIR filter is more efficient when its cutoff frequency is located at the high end of the spectrum. The point is this: it is easier to construct a more efficient filter using a FIR filter when the cutoff frequency is located at the high end of the spectrum, and using a Butterworth filter when the cutoff frequency is located at the low end of the spectrum. Of course, one can adjust the efficiency of either type of filter by changing the number of components (in the case of FIR filters) or the number of orders (in the case of Butterworth filters). But remember also that as the number of components or orders increase the filter takes more time to apply.

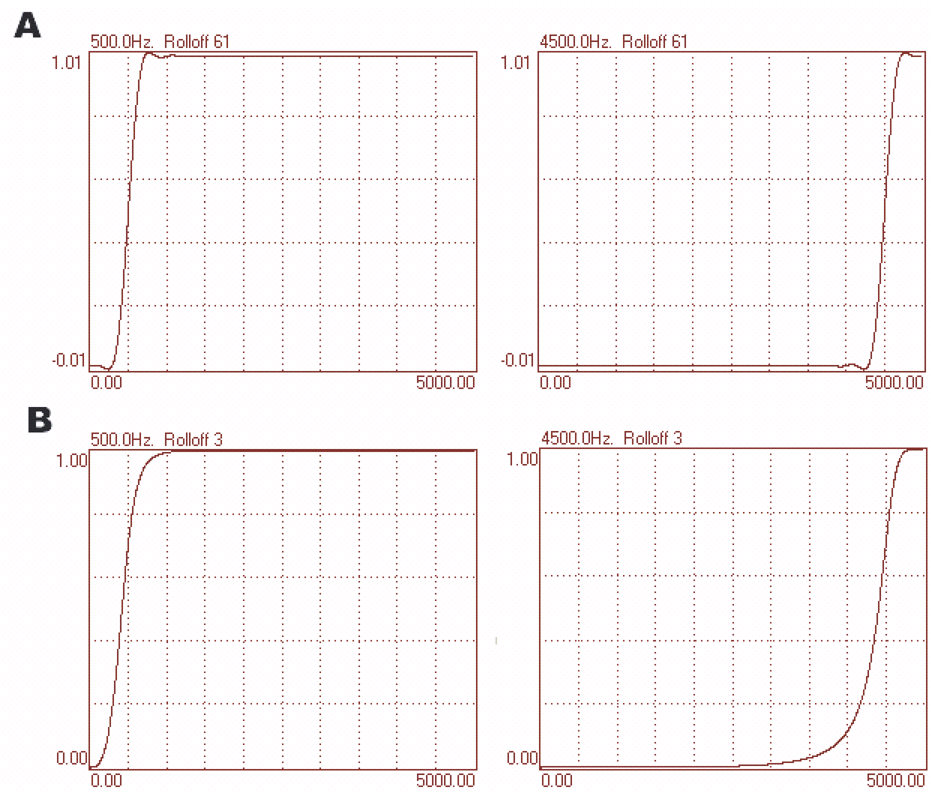


Figure D-48. A comparison of FIR and Butterworth response profiles when placed in different locations in the frequency spectrum.

Phase Lag

Another characterstic of Butterworth filters is their effect on the time-course of the signal to which they are applied. Specifically, a Butterworth filter introduces phase lags -- changes in the shape of the signal that make some component frequencies in the signal appear to occur later in time than they actually do. Consider the panel labeled as A in the example shown in Figure D-49. The red trace represents the original raw signal. The black trace shows the same signal after it has passed through a low pass Butterworth filter. Note that the filter has caused a phase lag in the dominant frequencies of the signal -- in other words, the filtered signal appears to be delayed in time relative to the raw signal. Fortunately phase lags can be eliminated in Butterworth filters by enabling the zero lag compensation feature. The panel labeled as B in Figure D-49 shows the raw signal and the filtered signal when zero lag compensation is enabled. Note that the phase lags have been eliminated.

The zero lag compensation feature eliminates phase lags by passing the signal through the filter twice -- once in the positive time direction and again in the negative time direction. Because the phase lags that are introduced with each pass are symmetric, they cancel each other out when the signal is passed through in opposite directions. But since the signal must pass through the filter twice the time required to apply the filter is doubled. Additional information about the effects of zero lag compensation is provided in Section D-12. Phase lags are not a property of FIR filters, and no compensation is therefore required.

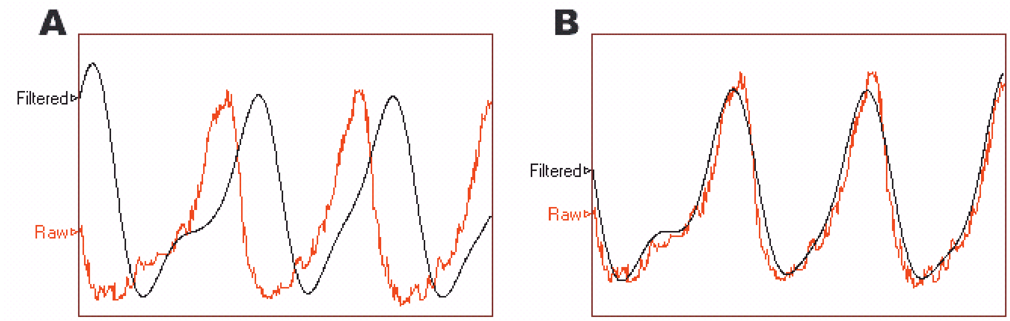


Figure D-49. The effect of applying zero lag compensation to a Butterworth filter.

D-11. Details and Limitations of the Butterworth Filter Algorithm

Datapac 2K2 uses Butterworth digital filters designed by the bilinear transform method for its low pass, high pass, band pass and band stop filters. Butterworth notch filters use a bandstop resonator function with an order of 2. It is the nature of the Butterworth algorithm that as the number of orders (poles) increases, some of the terms used in the calculation of the filter get extremely small in magnitude. A point is ultimately reached where the magnitude cannot be adequately represented in the 64 bit double precision arithmetic that Datapac employs. And at that point, errors are made in the calculation of the filter. Windows operating systems do not inform when such an overflow condition occurs, so it must be determined empirically. Unfortunately, that has turned out to be extremely difficult to achieve in software because the point of overflow changes in complex ways as a function of the number of orders in the filter, the sampling rate, and the cutoff frequency. We at RUN Technologies were therefore forced with a difficult decision: either (1) restrict the number of acceptable orders, or (2) let the user decide whether the filter they have designed is acceptable. We chose the latter alternative because we felt the former would have proven too restrictive if applied to a broad range of conditions. Besides, by looking at the response profile of a filter, it is very easy for the user to decide whether it is acceptable or not. Examples of the response profiles of an acceptable filter and an unacceptable one are shown in Figure D-50. Both are high pass filters. The acceptable filter is on the left. Note that its response profile is a smooth curve with its endpoints at gain = 0.00 and gain = 1.00. All of these conditions must be met for the filter to be considered acceptable. The response profile of an unacceptable filter is on the right. Note that the response profile is not a smooth curve, and although one endpoint is at gain = 0.00, the other is not at gain = 1.00.

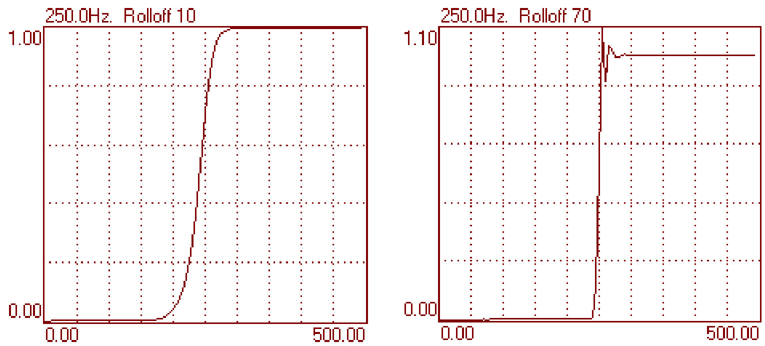


Figure D-50. Response profiles of acceptable (left) and unacceptable (right) Butterworth filters.

D-12. Effects of Zero Lag Compensation

Zero lag compensation is the process of eliminating phase lags in a signal when a Butterworth filter is applied to it. The compensation is achieved by passing the signal through the filter twice -- once in the positive time direction and once in the negative time direction. Since the signal is passed through the filter twice, you are in effect filtering a filtered signal. In addition to eliminating phase lags this process has two additional effects. First, the response profile is slightly sharpened. Second, the cutoff frequency is slightly shifted. Both effects are illustrated in the example shown in Figure D-51. Both panels show the response profile of a high pass Butterworth filter with the same cutoff frequency and rolloff. The only difference is that the left panel shows the response profile of the filter with the zero lag compensation feature disabled while it is enabled in the right panel. Note that the response profile shown in the right panel is sharper than the left panel. The cutoff frequency is indicated with a solid red line in the middle of each panel. With the zero lag compensation feature disabled (the left panel) the filter passes frequencies at the cutoff point with an attenuation of 3 dB, which equates to a gain of about 0.707. But with the zero lag compensation feature is enabled (the right panel) the attenuation is squared, resulting in a gain of about 0.5.

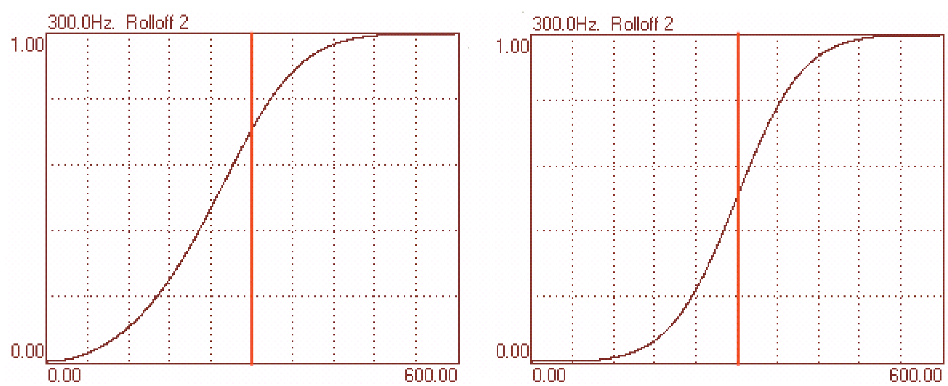


Figure D-51. The effects of zero lag compensation on the Butterworth filter response profile.