DADiSP / Filters

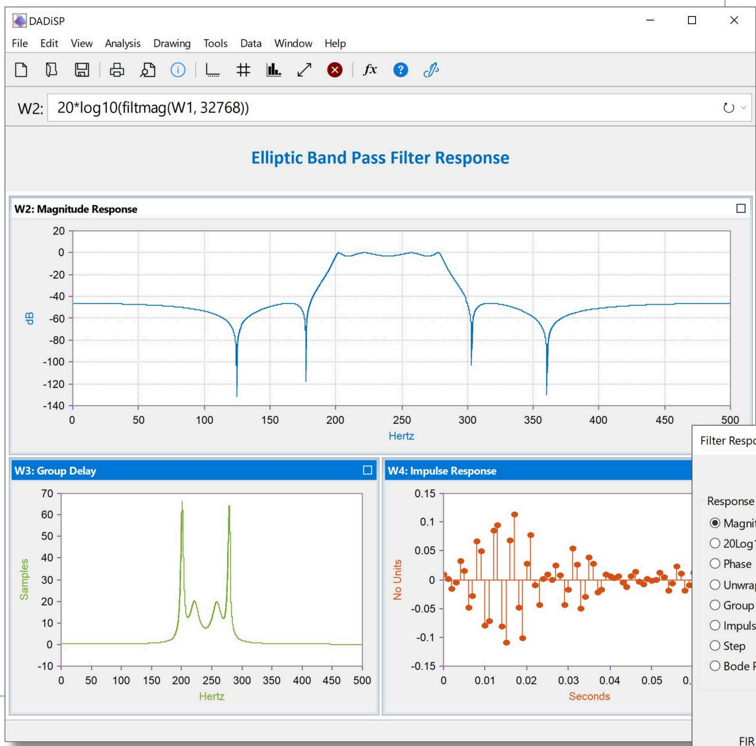
Digital Filter Design Module

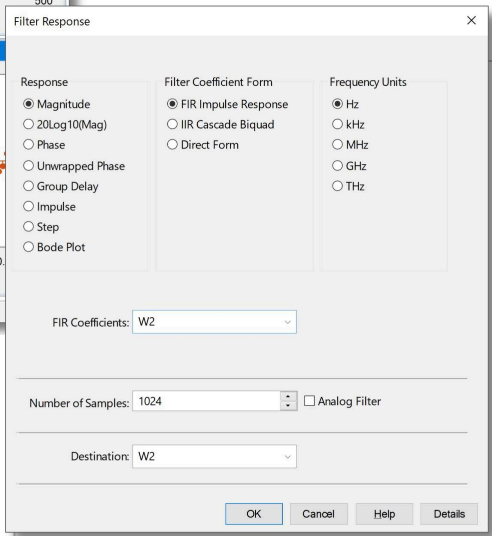
DADiSP/Filters is a comprehensive module for digital filter design and analysis. Quickly design, view and analyze both FIR (Finite Impulse Response) and IIR (Infinite Impulse Response) filters from any DADiSP Worksheet, then apply the filter to any data series and analyze the filtered results.

Easy-to-use dialog boxes or simple one line functions make digital filtering a snap for the novice while extensive functionality provides the capabilities required by filter design professionals.

KEY FEATURES

- Simple User Interface
- Lowpass, Highpass, Bandpass, Bandstop and Multiband Filters
- Finite Impulse Response (FIR) Filter Design
- FIR Hilbert Transforms and Differentiators
- FIR Remez Exchange and Kaiser Window Design Algorithms
- Infinite Impulse Response (IIR) Filter Design
- IIR Bessel, Butterworth, Chebychev I, Chebychev II and Elliptic Filters
- IIR Bilinear Transform and Matched Z Design Algorithms
- Magnitude, Phase, Group Delay and Impulse Response
- Output Coefficient Form Conversion
- Output Coefficient Quantization



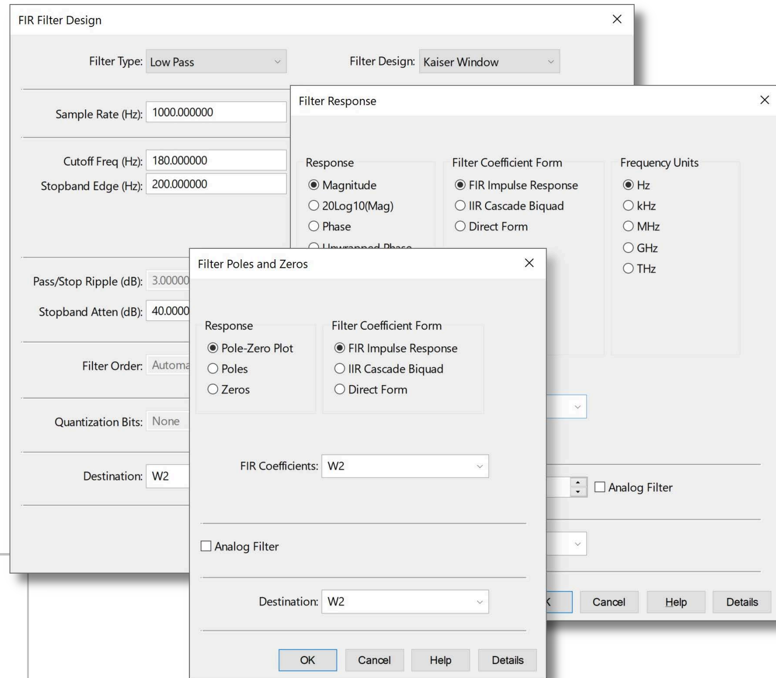


- Time and Frequency Domain Filtering Functions
- Pole-Zero Plots
- Bode Plots
- Phaseless IIR Filtering

New Features

DADiSP/Filters includes a number of functional and performance related improvements to simplify and optimize the design and analysis of filters.

Optimized time and frequency domain filtering routines for FIR and IIR filters provide more efficient filter processing. Optional end point padding reduces ramp up and ramp down transients for both FIR and IIR filtering operations. Phaseless IIR filtering with automatic initial conditions correction is included. A single filter can be applied to multi-column input series for multi-channel processing.



FILTERS NEW FEATURES SUMMARY

- Optimized Time and Frequency Domain Processing Functions
- Transposed Direct Form II Filter Implementation
- Optional End Point Padding for reduced Transients
- Constrained Fixed Order IIR Filter Designs
- Multi-channel Filtering
- Phaseless IIR Filtering
- IIR Analog Filter Design
- Bessel Filter Normalization Options
- Improved Stability and Accuracy
- Bode Plots for Frequency Domain Response
- Expanded and Detailed Help

Filters can be designed with user specified pass band ripple, transition band width and stop band attenuation or user specified fixed order. Fixed order designs automatically constrain secondary filter characteristics.

IIR filters support analog filter design. Multiple Bessel filter normalization methods are provided. IIR filter coefficients are automatically sequenced to provide the highest stability and accuracy for data filtering. Filtering uses the Transposed Direct Form II structure for faster processing with greater insensitivity to round off errors.

Expanded, extensive and detailed on-line help assists both the novice user and expert professional.

Digital Filter Design Module

DADiSP/Filters is a menu-driven, digital filtering module that adds complete FIR and IIR filtering capabilities to DADiSP. DADiSP/Filters allows you to quickly design, analyze and process both FIR (Finite Impulse Response) and IIR (Infinite Impulse Response) digital filters from easy-to-use dialog boxes or simple one line functions. On-line help and examples are also provided.

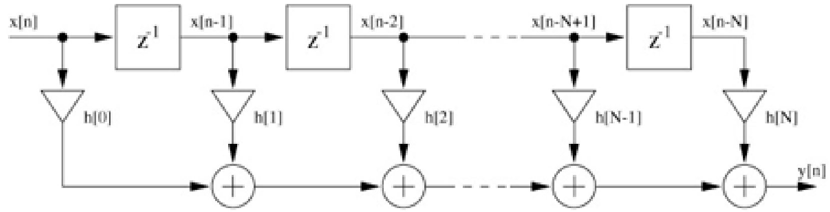
Design, Analyze and Apply

DADiSP/Filters gives you the power to easily build digital filters that emulate hardware based designs for testing and verification or perform filtering operations not possible with traditional analog methods. DADiSP/Filters allows you to remove noise generated during the data collection process.

DADiSP/Filters is the perfect complement to GPIBLab, DADiSP's acquisition module for collecting data from IEEE-488 based instruments and DADiSP/AdvDSP, a sophisticated module for advanced signal processing techniques. In addition, the ProPac Module increases the speed of many DADiSP/Filters functions by significantly accelerating the computation of the core FFT and convolution functions.

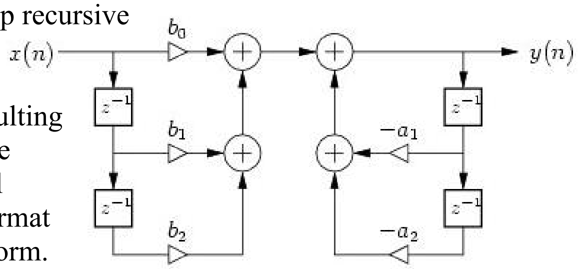
FIR Filters

The FIR module creates linear phase lowpass, highpass, bandpass, bandstop, multiband, Hilbert transformers and differentiators using the Parks-McClellan/Remez Exchange optimal design algorithm. The filter order can be specified or automatically estimated from the particular filter specifications. The Kaiser Window method is also provided, capable of creating both very high order FIR filters and extremely tight, narrowband filters.



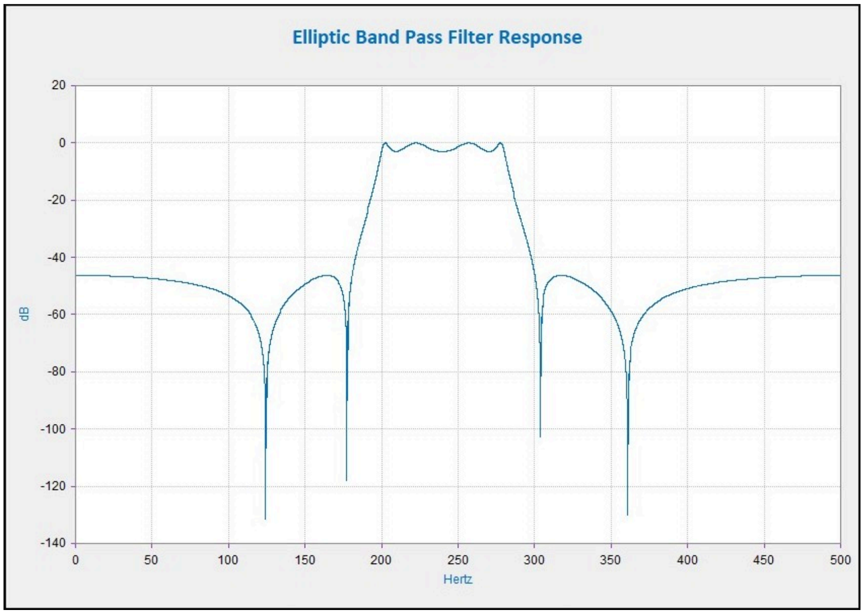
IIR Filters

The IIR module supports Bessel, Butterworth, Chebychev I, Chebychev II and Elliptic designs for lowpass, highpass, bandpass and bandstop recursive filters. The Bilinear transform method is employed and the resulting IIR coefficients can be determined in optimal multistage cascade format or traditional Direct form. The module also supports the Matched-Z transform method to approximate linear phase characteristics for IIR Bessel filters.



Coefficient Conversion

Filter coefficients can be converted to and from Cascade, Direct and FIR form and the coefficients can be quantized to emulate DSP chipsets.



Filter Response

The impulse, magnitude, phase and group delay characteristics can be calculated for any filter. Bode and Pole-Zero plots are also supported. Efficient time and frequency domain filtering algorithms are provided to apply the filter to any series.

Fully Integrated

The filters module is fully integrated with DADiSP to provide a complete digital filter design, analysis, display and processing environment. The DADiSP/Filters user interface is dialog based, eliminating the need to memorize argument lists or formulas and allows quick recall of previous filter designs. The filter coefficients are automatically displayed in a DADiSP window and can be saved for further use by DADiSP or other programs.

Filter Functions

Although most users access DADiSP/Filters through the dialog based interface, DADiSP/Filters includes over standalone 50 functions. The following table is a summary of each function.

* Indicates new or improved functions.

FIR Filters Functions

bandpass	Designs a FIR linear phase bandpass filter
bandstop	Designs a FIR linear phase bandstop filter
diff	Designs a FIR differentiator
fastfilter	FFT based FIR filtering
highpass	Designs a FIR linear phase highpass filter
hilbert	Designs a FIR Hilbert transformer
kwbpass*	Designs a Kaiser window FIR bandpass filter
kwbstop*	Designs a Kaiser window FIR bandstop filter
kwhpass*	Designs a Kaiser window FIR highpass filter
kwlpass*	Designs a Kaiser window FIR lowpass filter
lowpass	Designs a FIR linear phase lowpass filter
remez	Creates multiband FIR linear phase filters

Filter Coefficient Conversion Functions

cas2dir*	Converts Cascade form to Direct form
dir2cas*	Converts Direct form to Cascade form
fir2dir*	Converts FIR impulse form to Direct form
fir2cas*	Converts FIR impulse form to Cascade form

IIR Filters Functions

bessel*	Designs an IIR Bessel filter
butterworth	Designs an IIR Butterworth filter
cascade	Filters a time domain input with an IIR filter
cheby1	Designs an IIR Chebychev I filter
cheby2	Designs an IIR Chebychev II filter
elliptic	Designs an IIR Elliptical filter

Filter Response Functions

filtgrpdelay*	Calculates group delay of any filter
filtmag*	Calculates any filter magnitude response
filtimp*	Calculates any filter impulse response
filtphase*	Calculates any filter phase response
firmag	Calculates FIR filter magnitude response
firphase	Calculates FIR filter phase response
iirimp*	Calculates IIR filter impulse response
iirmag*	Calculates IIR filter magnitude response
iirphase*	Calculates IIR filter phase response
filtgrpdelay*	Calculates group delay of any filter
filtmag*	Calculates any filter magnitude response
filtimp*	Calculates any filter impulse response

Filtering Functions

dirfilter*	Apply Direct form filter in the time domain
dirfilterF*	Apply Direct form filter in the frequency domain
filtdataF*	Apply any filter in the frequency domain
filtdata*	Apply any filter in the time domain
firfilterF*	Apply FIR filter in the frequency domain
firfilter	Apply FIR filter in the time domain
iirfilterF*	Apply IIR filter in the frequency domain
iirfilter	Apply IIR filter in the time domain
dirfilter*	Apply Direct form filter in the time domain
dirfilterF*	Apply Direct form filter in the frequency domain
filtdataF*	Apply any filter in the frequency domain
filtdata*	Apply any filter in the time domain

Misc Filtering Functions

filtzeros*	Calculates zeros of any filter
filtpoles*	Calculates poles of any filter
fir	Evaluates a FIR difference equation
firpz	Creates an FIR filter zero plot
firzeros	Calculates zeros of an FIR filter
fullfir	Converts FIR filter to full band linear phase
iir	Evaluates an IIR difference equation
iirgrpdelay*	Calculates group delay of an IIR filter
iirpoles	Calculates poles of an IIR filter
iirpz	Creates an IIR filter pole-zero plot
iirzeros	Calculates zeros of an IIR filter
polecoef	Converts IIR biquad to direct pole coeff form
quantize*	Quantize filter coefficients to N bits
zerocoef	Converts IIR biquad to direct zero coeff form
unwrap	Phase unwrapping using Schafer's algorithm