

DADiSP / Filters

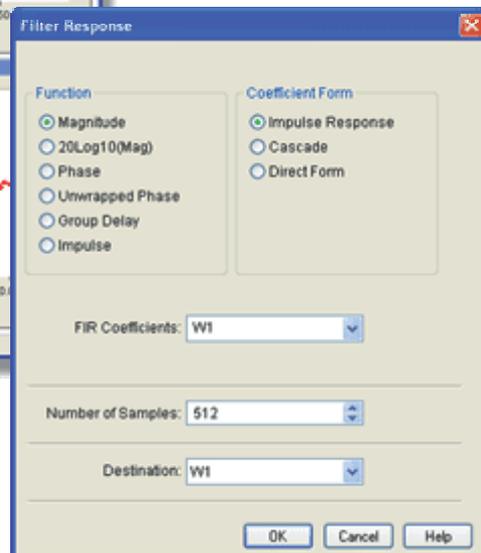
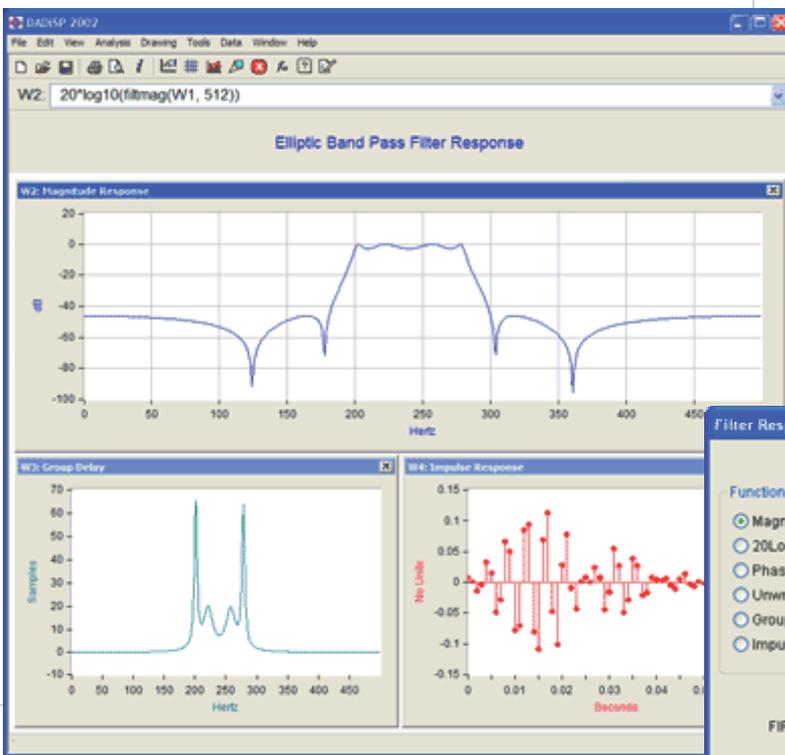
Digital Filter Design Module

◆ デジタル フィルター 設計 モジュール

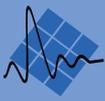
DADiSP/Filtersは、FIR (Finite Impulse Response) およびIIR (Infinite Impulse Response) デジタル フィルターの設計と解析のためのメニュー駆動型のモジュールです。DADiSPのワークシートからFIRとIIRフィルターを素早く設計、波形表示、および解析ができます。フィルターを設計した後は、生データをフィルタリングし、フィルター通過後の波形表示、解析ができます。簡単操作のダイアロ・ボックスか、あるいはシンプルな一行の関数を使って、設計したフィルターをインタラクティブに調整することができます。

主な機能

- ・ シンプル ユーザー インターフェース
- ・ ローパス、ハイパス、バンドパス、バンドストップ、マルチバンド フィルター
- ・ FIR フィルター設計
- ・ FIRヒルバート変換、識別器
- ・ FIR REMEZ法、カイザー・ウィンドウ法アルゴリズム
- ・ IIR フィルター設計
- ・ IIR ベッセル、バターワース、チェビシェフ、チェビシェフII、Elliptic フィルター
- ・ IIR Bilinear型変換、およびMatched Z設計アルゴリズム
- ・ 振幅、位相、グループ遅延、インパルスレスポンス
- ・ 出力係数フォーム変換



- ・ 出力係数量子化
- ・ 時間、周波数領域フィルタリング機能
- ・ ポール・ゼロ プロット



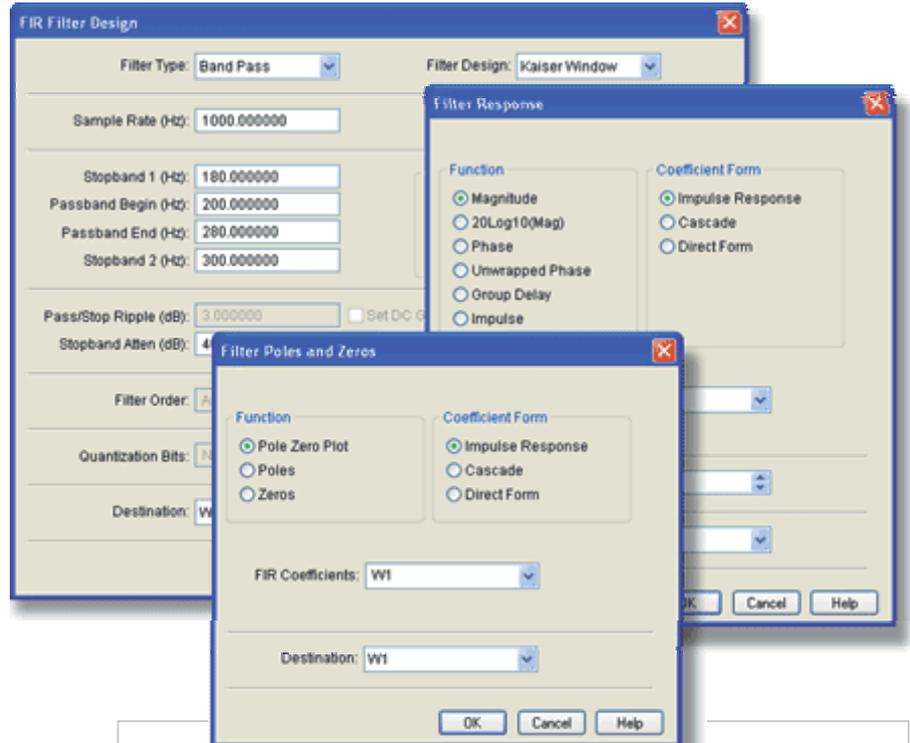
DADiSP/Filters Ver 5.0の新しい機能

DADiSP/Filters Ver5.0では、デジタルフィルターの設計及びその利用をより合理化するためにユーザ・インターフェースを大幅に改善しました。自動的なオプションの確認機能を備えた使い易いダイアログ・ボックスがフィルターの設計と分析の両方を単純化しました。様々なフィルタの構造は、フィルター係数によって容易に変換されます。また、DSPチップセットのエミュレート支援のために、クオンタイズ・ルーチンが追加されました。IIRベッセル・フィルター(Bessel Filters)及びMatched Z変換設計法が追加されました。線形位相FIRカイザー・フィルター(Kaiser Filters)が拡張され、機能増強されました。時間領域及び周波数領域のフィルタールーチンが、より効率的なフィルター処理を提供するために最適化されました。

デジタル フィルター 設計 モジュール

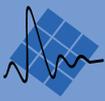
DADiSP/Filtersモジュールは、DADiSPにFIR及びIIRフィルター機能を追加するメニュー駆動型のデジタル・フィルター・モジュールです。

DADiSP/Filtersモジュールは、使い易いダイアログ・ボックスかシンプルな1行の関数によって、FIR(Finite Impulse Response)及びIIR(Infinite Impulse Response)デジタルフィルターの迅速な設計、分析、実行を可能にします。オンライン・ヘルプ機能と例題も用意されています。



Filters Ver 5.0の新しい機能の要約

- ・ 合理化されたインターフェース
- ・ IIRベッセル・フィルター(Bessel Filters)
- ・ IIR Matched Z設計アルゴリズム
- ・ 改善されたFIR カイザー・ウィンドウ・フィルタ(Kaiser Window Filters)
- ・ 係数変換及びクオンタイゼーション(Quantization)
- ・ 最適化されたフィルター処理機能

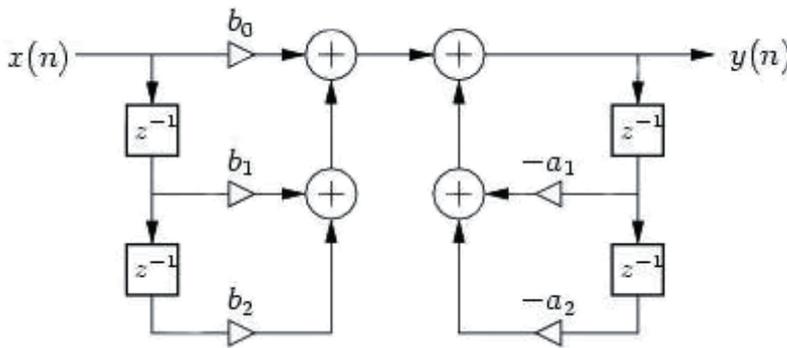


設計、分析、実行

DADiSP/Filtersモジュールは、デジタルフィルタを容易に構築可能とするもので、テストと検証のためにハードウェアの設計をエミュレートし、あるいは従来のアナログ手法で実現できないフィルタ処理を実現します。DADiSP/Filtersモジュールは、データ収集プロセスで生じたノイズを除去することを可能にします。DADiSP/Filtersモジュールは、DADiSP/GPIBLabモジュール(IEEE-488準拠の測定器からのデータをインポート)、及びDADiSP/AdvDSPモジュール(高度な信号処理技術を提供するモジュール)の機能を補強します。

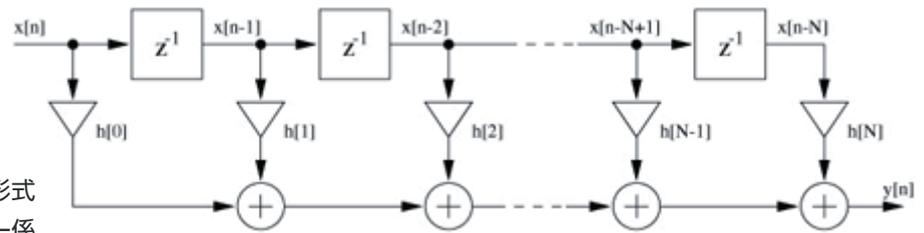
フィルタ係数変換

フィルタ係数は、カスケード、直接、及びFIR形式に変換したり、されたりします。また、フィルタ係数は、DSPチップセットをエミュレートするために量子化することができます。



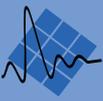
FIRフィルター

FIRモジュールは、Parks-McClellan/Remez Exchange最適化アルゴリズムを使用して、線形位相ローパス、ハイパス、バンドパス、バンドストップ、マルチバンドのフィルタ、ヒルバート変換器、及び差分器を作成します。フィルタ次数は、特定のフィルタ仕様から指定されるか自動的に推定されます。カイザー・ウィンドウ法も提供され、非常に高い次数のFIRフィルタと非常に厳しい、狭帯域フィルタを作成可能とします。



IIRフィルター

IIRモジュールは、ベッセル、バターワース、チェビシェフI、チェビシェフII、及び楕円設計法をサポートし、ローパス、ハイパス、バンドパス、及びバンドストップの再帰フィルタを作成します。Bilinear変換法が採用され、得られたIIR係数は、最適多段カスケードフォーマットが従来の直接的なフォームで決定されます。さらに、モジュールは、IIRベッセル・フィルタ用の線形位相特性を近似するMatched-Z法をサポートします。

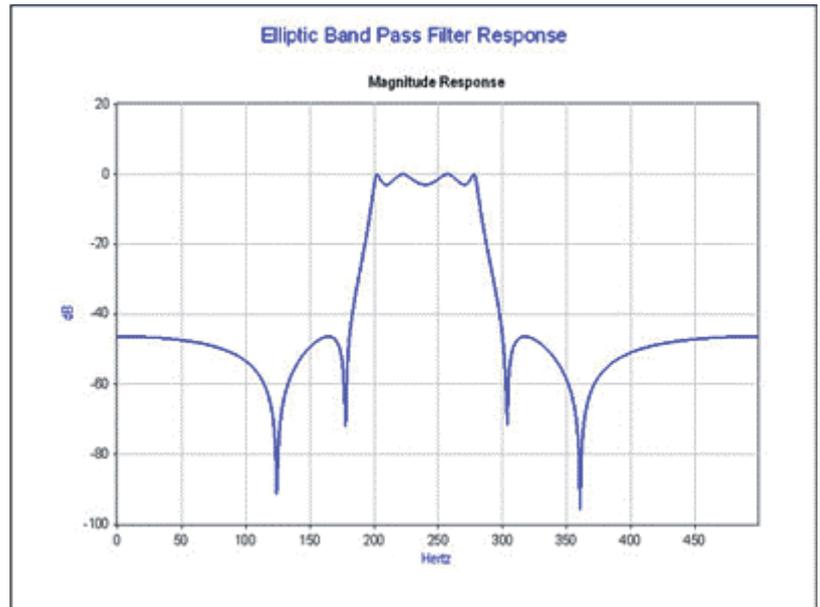


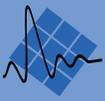
フィルター・レスポンス

任意のフィルターに対して、インパルス、振幅、位相及び群遅延特性を計算することができます。ポール・ゼロ・プロットもサポートされます。有効な時間及び周波数領域のフィルター・アルゴリズムが、フィルターに任意のシリーズを適用するために用意されています。

DADiSPに完全に統合されました

DADiSP/Filtersモジュールは、DADiSPに完全に統合され、デジタルフィルターの設計、分析、表示及び処理の環境を提供します。DADiSP/Filtersモジュールのユーザ・インターフェースは、対話型で、変数リストや公式を記憶する必要を無くし、さらに以前行なわれたフィルター設計データを素早く取り出すことができるようになっています。フィルターの係数は、DADiSPウィンドウに自動的に表示され、さらにDADiSPまたは他のプログラムによって再び利用できるように保存することもできます。





フィルター関数

DADiSP/Filtersモジュールを使う場合、大部分のユーザは、対話ベースのインターフェースを使いますが、関数も使えます。DADiSP/Filters は、50以上の関数を持っています。以下にその要約を示します。

* Ver.5.0で新たに追加された関数、または機能向上された関数を示します。

FIR フィルター関数

| | |
|------------|-------------------------------------|
| bandpass | FIR線形位相バンドパス・フィルターの設計 |
| bandstop | FIR線形位相バンドストップ・フィルターの設計 |
| diff | FIR 微分器の設計 |
| fastfilter | FFTベースのFIRフィルタリング |
| highpass | FIR線形位相ハイパス・フィルターの設計 |
| hilbert | FIRヒルバート変換器の設計 |
| kwbpass* | カイザー・ウィンドウFIRバンドパス・フィルターの設計 |
| kwbstop* | カイザー・ウィンドウFIR バンドストップ・フィルターの設計 |
| kwhpass* | カイザー・ウィンドウFIRハイパス・フィルターの設計 |
| kwlpas* | カイザー・ウィンドウFIRローパス・フィルターの設計 |
| lowpass | FIR線形位相ローパス・フィルターの設計 |
| remez | FIR線形位相マルチバンド・フィルターの作成 FIR線形位相バン |

フィルター係数変換関数

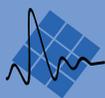
| | |
|----------|------------------------|
| cas2dir* | カスケード形式を直接形式に変換 |
| dir2cas* | 直接形式をカスケードの形式に変換 |
| fir2dir* | FIRインパルス形式を直接形式に変換 |
| fir2cas* | FIRインパルス形式をカスケードの形式に変換 |

IIRフィルター関数

| | |
|-------------|----------------------|
| bessel* | IIRベッセル・フィルターの設計 |
| butterworth | IIRバターワース・フィルターの設計 |
| cascade | IIRフィルタで時間領域入力をフィルター |
| cheby1 | IIRチェビシェフ。フィルターの設計 |
| cheby2 | IIRチェビシェフIIフィルターの設計 |
| elliptic | IIR楕円フィルターの設計 |

フィルター・レスポンス関数

| | |
|---------------|---------------------|
| filtgrpdelay* | フィルターの遅延を計算 |
| filtmag* | フィルターの振幅応答の計算 |
| filtimp* | フィルターのインパルス応答の計算 |
| filtphase* | フィルターの位相応答の計算 |
| firmag | FIRフィルターの振幅応答の計算 |
| firphase | FIRフィルターの位相応答の計算 |
| iirimp* | IIRフィルターのインパルス応答の計算 |
| iirmag* | IIRフィルターの振幅応答の計算 |
| iirphase* | IIRフィルターの位相応答の計算 |



フィルター関数

| | |
|--------------------------|--------------------|
| <code>dirfilter*</code> | 時間領域に直接形式フィルターを適用 |
| <code>dirfilterF*</code> | 周波数領域に直接形式フィルターを適用 |
| <code>filtdataF*</code> | 周波数領域に任意のフィルターを適用 |
| <code>filtdata*</code> | 時間領域に任意のフィルターを適用 |
| <code>firfilterF*</code> | 周波数領域にFIRフィルターを適用 |
| <code>firfilter</code> | 時間領域にFIRフィルターを適用 |
| <code>iirfilterF*</code> | 周波数領域にIIRフィルターを適用 |
| <code>iirfilter</code> | 時間領域にIIRフィルターを適用 |

その他のフィルター関数

| | |
|---------------------------|-----------------------------|
| <code>filtzeros*</code> | 任意のフィルターのゼロを計算 |
| <code>filtpoles*</code> | 任意のフィルターのポールを計算 |
| <code>fir</code> | FIR差分方程式の評価 |
| <code>firpz</code> | FIRフィルターのゼロ・プロットを作成 |
| <code>firzeros</code> | FIRフィルターのゼロを計算 |
| <code>fullfir</code> | FIRフィルターの全バンドを線形位相に変換 |
| <code>iir</code> | IIR差分方程式を評価 |
| <code>iirgrpdelay*</code> | IIRフィルターの遅延を計算 |
| <code>iirpoles</code> | IIRフィルターのポールを計算 |
| <code>iirpz</code> | IIRフィルターのポール・ゼロのプロットを作成 |
| <code>iirzeros</code> | IIRフィルターのゼロを計算 |
| <code>polecoef</code> | 2次IIRを直接ポール係数形式に変換 |
| <code>quantize*</code> | フィルター係数をNビットへ量子化 |
| <code>zerocoeff</code> | 2次IIRを直接ゼロ係数形式に変換 |
| <code>unwrap</code> | シェーファアのアルゴリズムにより位相を非循環形式に変換 |