Manual de Usuario

Paleta de Componentes

LazFilters

Contenido

Visión General de los Componentes 1
Units 2
Tipos de datos 3
Estructura de Clases 4
Clase TFilter 4
Diseño 4
Propiedades 4
Métodos 4
Clase TFilterIIR 6
Diseño 6
Propiedades 6
Clase TFilterIIRLP 7
Diseño 7
Propiedades 7
Métodos 7
Clase TFilterIIRHP 8
Diseño 8
Propiedades 8
Métodos 8
Clase TFilterIIRBP 9
Diseño 9
Propiedades 9
Métodos 9
Clase TFilterIIRBS 10
Diseño 10
Propiedades 10
Métodos 10
Clase TFilterIIRLPButterworth 11
Diseño 11
Método 11
Clase TFilterIIRLPChebyshev 12
Diseño 12
Propiedades 12
Método 12
Clase TFilterIIRLPBessel 13

Diseño 13
Método 13
Clase TFilterIIRHPButterworth 14
Diseño 14
Método 14
Clase TFilterIIRHPChebyshev 15
Diseño 15
Propiedades 15
Método 15
Clase TFilterIIRHPBessel 16
Diseño 16
Método 16
Clase TFilterIIRBPButterworth 17
Diseño 17
Método 17
Clase TFilterIIRBPChebyshev 18
Diseño 18
Propiedades 18
Método 18
Clase TFilterIIRBPBessel 19
Diseño 19
Método 19
Clase TFilterIIRBSButterworth 20
Diseño 20
Método 20
Clase TFilterIIRBSChebyshev 21
Diseño 21
Propiedades 21
Método 21
Clase TFilterIIRBSBessel 22
Diseño 22
Método 22
Diseño Global de Clases 23

Visión General de los Componentes

La paleta contiene 12 componentes que tienen la siguiente nomenclatura :

Nomenclatura

Filter: Indica que es un filtro.

FilterIIR: Indica que es un filtro de respuesta infinita al impulso.

FilterIIRLP: Indica que es un filtro de respuesta infinita al impulso pasa bajo.

FilterIIRHP: Indica que es un filtro de respuesta infinita al impulso pasa alto.

FilterIIRBP: Indica que es un filtro de respuesta infinita al impulso pasa banda.

FilterIIRBS: Indica que es un filtro de respuesta infinita al impulso supresor de banda.

FilterIIRLPButterworth: Indica que es un filtro de respuesta infinita al impulso pasa bajo Butterworth.

FilterIIRLPChebyshev: Indica que es un filtro de respuesta infinita al impulso pasa bajo Chebyshev.

FilterIIRLPBessel: Indica que es un filtro de respuesta infinita al impulso pasa bajo Bessel.

FilterIIRHPButterworth: Indica que es un filtro de respuesta infinita al impulso pasa alto Butterworth.

FilterIIRHPChebyshev: Indica que es un filtro de respuesta infinita al impulso pasa alto Chebyshev.

FilterIIRHPBessel: Indica que es un filtro de respuesta infinita al impulso pasa alto Bessel.

FilterIIRBPButterworth: Indica que es un filtro de respuesta infinita al impulso pasa banda Butterworth.

FilterIIRBPChebyshev: Indica que es un filtro de respuesta infinita al impulso pasa banda Chebyshev.

FilterIIRBPBessel: Indica que es un filtro de respuesta infinita al impulso pasa banda Bessel.

FilterIIRBSButterworth: Indica que es un filtro de respuesta infinita al impulso supresor de banda Butterworth.

FilterIIRBSChebyshev: Indica que es un filtro de respuesta infinita al impulso supresor de banda Chebyshev.

FilterIIRBSBessel: Indica que es un filtro de respuesta infinita al impulso supresor de banda Bessel.

Units

Filter: Contiene propiedades y métodos que posee cualquier tipo de filtro.

FilterIIR: Contiene propiedades y métodos que poseen los filtros (IIR).

FilterIIRLP: Contiene propiedades y métodos que poseen los filtros (IIR) pasa bajos.

FilterIIRHP: Contiene propiedades y métodos que poseen los filtros (IIR) pasa altos.

FilterIIRBP: Contiene propiedades y métodos que poseen los filtros (IIR) pasa bandas.

FilterIIRBS: Contiene propiedades y métodos que poseen los filtros (IIR) supresores de bandas.

FilterIIRLPButterworth: Contiene propiedades y métodos que poseen los filtros (IIR) pasa bajo Butterworth.

FilterIIRLPChebyshev: Contiene propiedades y métodos que poseen los filtros (IIR) pasa bajo Chebyshev.

FilterIIRLPBessel: Contiene propiedades y métodos que poseen los filtros (IIR) pasa bajo Bessel.

FilterIIRHPButterworth: Contiene propiedades y métodos que poseen los filtros (IIR) pasa alto Butterworth.

FilterIIRHPChebyshev: Contiene propiedades y métodos que poseen los filtros (IIR) pasa alto Chebyshev.

FilterIIRHPBessel: Contiene propiedades y métodos que poseen los filtros (IIR) pasa alto Bessel.

FilterIIRBPButterworth: Contiene propiedades y métodos que poseen los filtros (IIR) pasa banda Butterworth.

FilterIIRBPChebyshev: Contiene propiedades y métodos que poseen los filtros (IIR) pasa banda Chebyshev.

FilterIIRBPBessel: Contiene propiedades y métodos que poseen los filtros (IIR) pasa banda Bessel.

FilterIIRBSButterworth: Contiene propiedades y métodos que poseen los filtros (IIR) supresor de banda Butterworth.

FilterIIRBSChebyshev: Contiene propiedades y métodos que poseen los filtros (IIR) supresor de banda Chebyshev.

FilterIIRBSBessel: Contiene propiedades y métodos que poseen los filtros (IIR) supresor de banda Bessel.

Uconst: Contiene la declaraciones de las constates.

UOperator: Contiene un conjunto de funciones para trabajar con números complejos.

UType: Contiene las declaraciones de los tipos de datos.

Tipos de datos

TfreqType: Tipo de dato para especificar frecuencias.

FilterOrder: Tipo de dato para el orden del filtro.

TfreqCutArray: Tipo de dato para arreglo de frecuencias.

PFreqCutArray: Tipo de dato para puntero a un arreglo de frecuencias.

TComplexArray: Tipo de dato para arreglo de números complejos.

PComplexArray: Tipo de dato para puntero a arreglo de números complejos.

TAS_Sample: Tipo de dato para números flotantes.

TAS_SampleArray: Tipo de dato para arreglo de números flotantes.

PTAS_SampleArray: Tipo de dato para puntero a arreglo de números flotantes.

TAS_Float: Tipo de dato para especificar la frecuencia de muestreo.

FilterProc: Tipo de dato para el procesamiento del valor de entrada de acuerdo a los parámetros del filtro. El resultado es el valor filtrado. Los datos pueden ser procesados en tiempo real.

Estructura de Clases

Clase TFilter

La clase TFilter hereda de la clase TComponent, y constituye la clase base para creación de cualquier tipo de filtro digital, ya sea IIR ó FIR.

Diseño

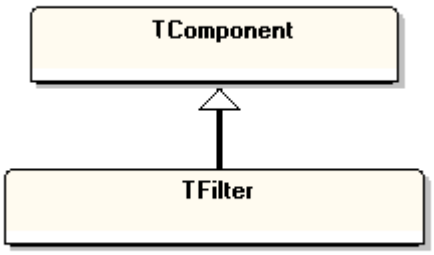


Figura 1 Clase TFilter

Propiedades

property FreqCutCount : Cantidad de frecuencias de corte que tendrá el filtro.

property FreqCut: Frecuencia de corte.

property Order: Orden del filtro.

property SampleRate: Frecuencia de muestreo.

property Gain: Ganancia de la señal de salida.

property Info: Esta propiedad contiene información del filtro (ganancia de entrada y coeficientes de salida).

Métodos

procedure Reset**; virtual**

Borra el buffer interno.

procedure SetupFilter(); **virtual**

Tiene la función de configurar el filtro, se declara virtual en esta clase y será redefinido en cada componente de acuerdo a sus funcionalidades.

procedure SetOrder(AOrder: FilterOrder)

Permite especificar el orden del filtro.

procedure SetSampleRate(ASampleRate: TAS_Float)

Permite especificar la frecuencia de muestreo.

function FilterPassAll(InputValue: TAS_Sample): TAS_Sample; **virtual**

Tipo de filtro que deja pasar hacia la salida lo mismo que esta en su entrada.

function FilterPassNothing(InputValue: TAS_Sample): TAS_Sample; **virtual**

Tipo de filtro en el cual la salida siempre es 0, independientemente de la entrada.

function FilterFilter(InputValue: TAS_Sample): TAS_Sample; **virtual**

Permite filtrar la señal en función de la configuración realizada previamente.

function FGetInfo**:** String

Devuelve la información del filtro.

function GetFrequencyResponse(FreqCut1: TFreqType): TFreqType

Devuelve la respuesta del filtro en el dominio de la frecuencia.

Clase TFilterIIR

La clase TFilterIIR hereda de la clase TFilter, y encapsula todo lo concerniente a los filtros digitales de respuesta infinita al impulso.

Diseño

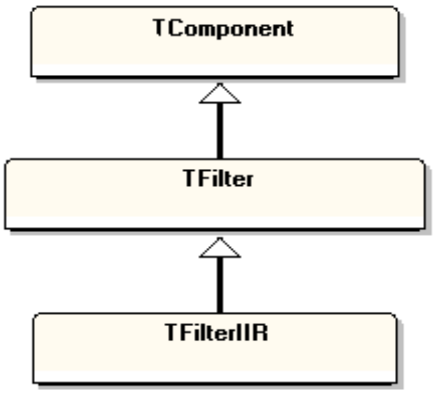


Figura 2 Clase TFilterIIR

Propiedades

property ManualReset:

Esta propiedad se debe Establecer en true si desea cambiar los parámetros del filtro (frecuencias y el tipo de filtro) durante el procesamiento de datos. Por lo general, (ManualReset = false) ya que cuando los parámetros del filtro se cambian los búferes internos se borran, así se puede conseguir algunas distorsiones en el procesamiento de la señal.

property LastError: Descripción del error.

Clase TFilterIIRLP

La clase TFilterIIRLP hereda de la clase TFilterIIR, y encapsula todo lo concerniente a los filtros digitales de respuesta infinita al impulso pasa bajo.

Diseño

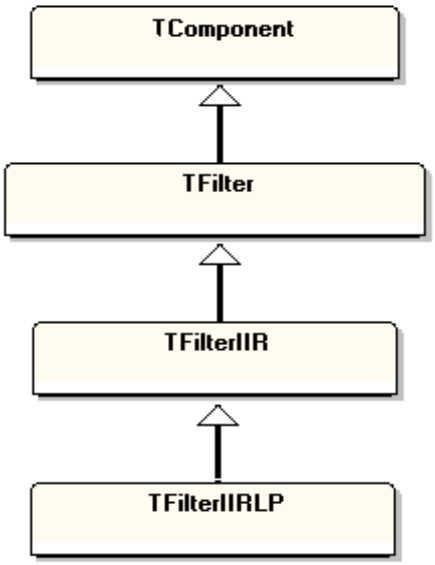


Figura 3 Clase TFilterIIRLP

Propiedades

property FreqCut1: Frecuencia de corte 1.

Métodos

procedure SetFreqCut1(const InputValue: TFreqType); **virtual**

Permite insertar el valor de la frecuencia de corte uno.

procedure ResponseLowPass

Implementa la respuesta de un filtro pasa bajo.

function GetFreqCut1() : TfreqType

Devuelve la frecuencia de corte uno.

Clase TFilterIIRHP

La clase TFilterIIRHP hereda de la clase TFilterIIR, y encapsula todo lo concerniente a los filtros digitales de respuesta infinita al impulso pasa alto.

Diseño

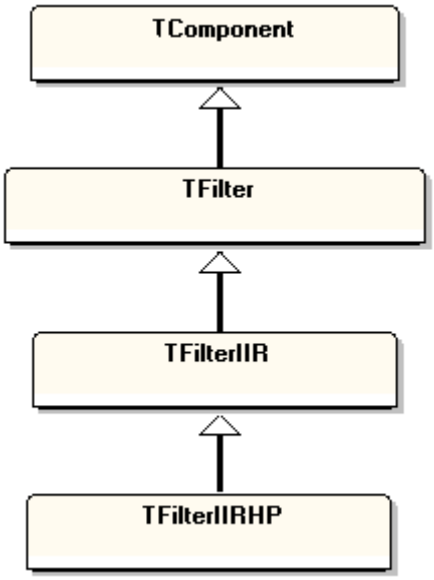


Figura 4 Clase TFilterIIRHP

Propiedades

property FreqCut1: Frecuencia de corte 1.

Métodos

procedure SetFreqCut1(const InputValue: TFreqType)

Permite insertar el valor de la frecuencia de corte 1.

procedure ResponseHighPass

Implementa la respuesta de un filtro pasa alto.

function GetFreqCut1() : TFreqType

Devuelve la frecuencia de corte 1.

Clase TFilterIIRBP

La clase TFilterIIRBP hereda de la clase TFilterIIR, y encapsula todo lo concerniente a los filtros digitales de respuesta infinita al impulso pasa banda.

Diseño

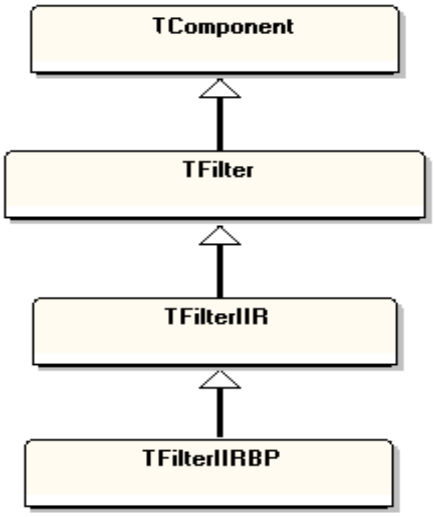


Figura 5 Clase TFilterIIRBP

Propiedades

property FreqCut1 : Frecuencia de corte 1.

property FreqCut2 : Frecuencia de corte 2.

Métodos

procedure ResponseBandPass

Implementa la respuesta de un filtro pasa banda.

procedure SetFreqCut1(const InputValue: TFreqType); **virtual**

Permite especificar el valor de la frecuencia de corte 1.

procedure SetFreqCut2(const InputValue: TFreqType); **virtual**

Permite especificar el valor de la frecuencia de corte 2.

function GetFreqCut1() : TFreqType

Devuelve la frecuencia de corte uno.

function GetFreqCut2() : TFreqType

Devuelve la frecuencia de corte 2.

Clase TFilterIIRBS

La clase TFilterIIRBS hereda de la clase TFilterIIR, y encapsula todo lo concerniente a los filtros digitales de respuesta infinita al impulso, supresores de banda.

Diseño

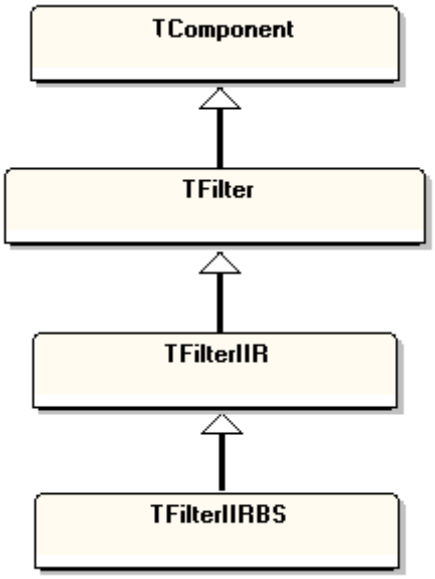


Figura 6 Clase TFilterIIRBS

Propiedades

property FreqCut1 : Frecuencia de corte 1.

property FreqCut2 : Frecuencia de corte 2.

Métodos

procedure ResponseBandStop

Implementa la respuesta de un filtro supresor de banda.

procedure SetFreqCut1(const InputValue: TFreqType); **virtual**

Permite especificar el valor de la frecuencia de corte 1.

procedure SetFreqCut2(const InputValue: TFreqType); **virtual**

Permite especificar el valor de la frecuencia de corte 2.

function GetFreqCut1() : TFreqType

Devuelve la frecuencia de corte 1.

function GetFreqCut2() : TFreqType

Devuelve la frecuencia de corte 2.

Clase TFilterIIRLPButterworth

Diseño

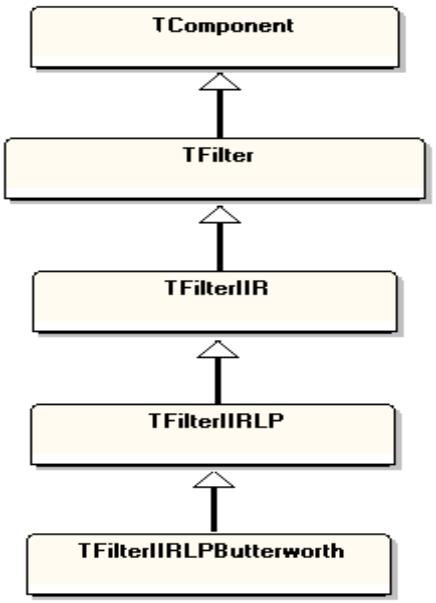


Figura 7 Clase TFilterIIRLPButterworth

Método

Se redefine el método Setup()

Clase TFilterIIRLPChebyshev

Diseño

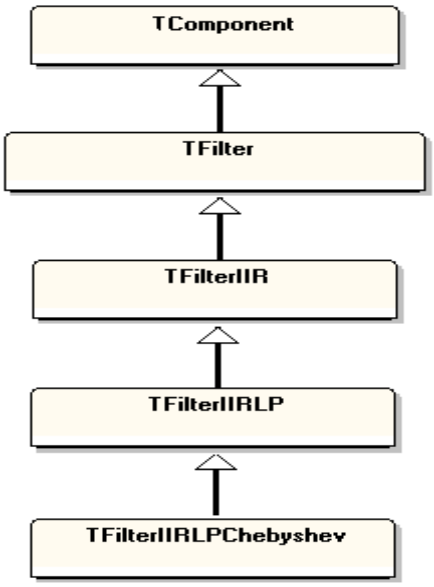


Figura 8 Clase TFilterIIRLPChebyshev

Propiedades

property Ripple: La ondulación en la banda de paso en dB.

Método

procedure SetRipple(ARipple: TAS_Sample)

Permiter especificar el ripple.

procedure SetupForChebyshev()

Para el funcionamiento de los filtros Chebyshev.

(*) Se redefine el método Setup()

Clase TFilterIIRLPBessel

Diseño

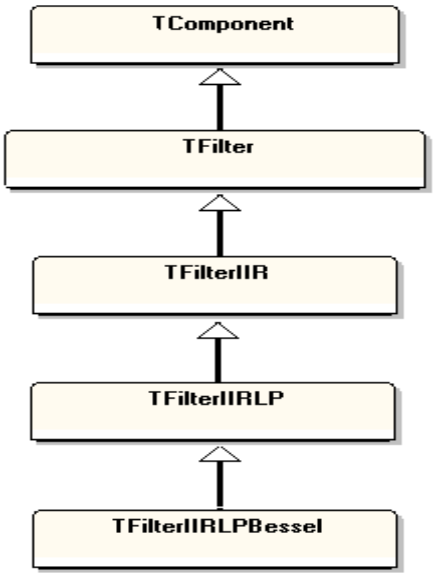


Figura 9 Clase TFilterIIRLPBessel

Método

Se redefine el método Setup()

Clase TFilterIIRHPButterworth

Diseño

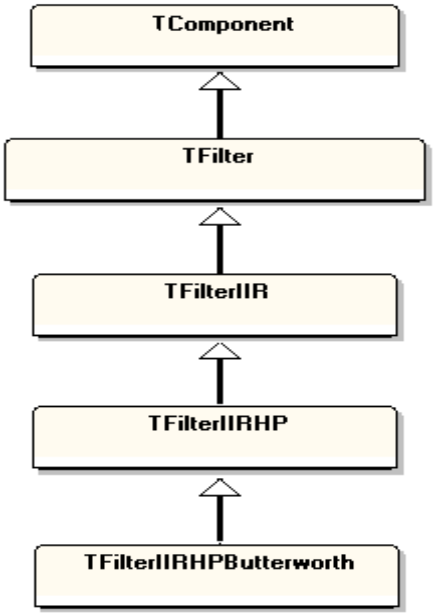


Figura 10 Clase TFilterIIRHPButterworth

Método

Se redefine el método Setup()

Clase TFilterIIRHPChebyshev

Diseño

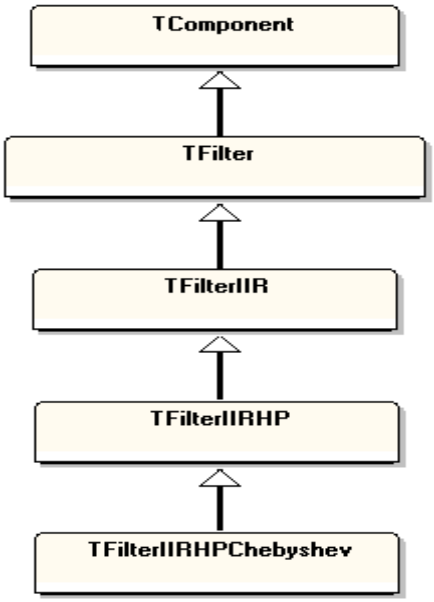


Figura 11 Clase TFilterIIRHPChebyshev

Propiedades

property Ripple: La ondulación en la banda de paso en dB.

Método

procedure SetRipple(ARipple: TAS_Sample)

Permiter especificar el ripple.

procedure SetupForChebyshev()

Para el funcionamiento de los filtros Chebyshev.

(*) Se redefine el método Setup()

Clase TFilterIIRHPBessel

Diseño

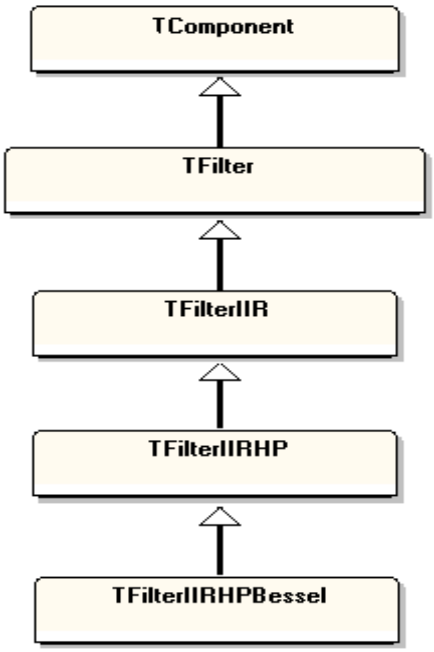


Figura 12 Clase TFilterIIRHPBessel

Método

Se redefine el método Setup()

Clase TFilterIIRBPButterworth

Diseño

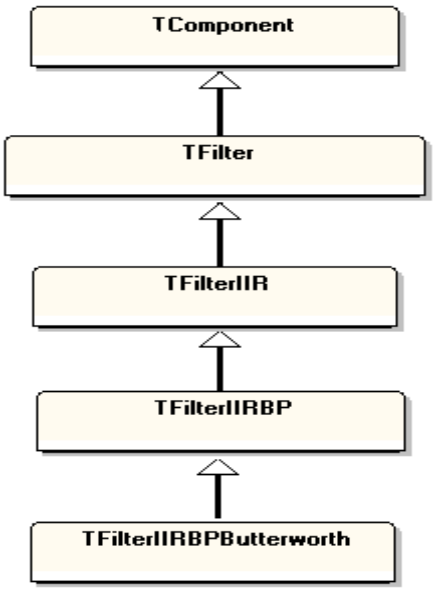


Figura 13 Clase TFilterIIRBPButterworth

Método

Se redefine el método Setup()

Clase TFilterIIRBPChebyshev

Diseño

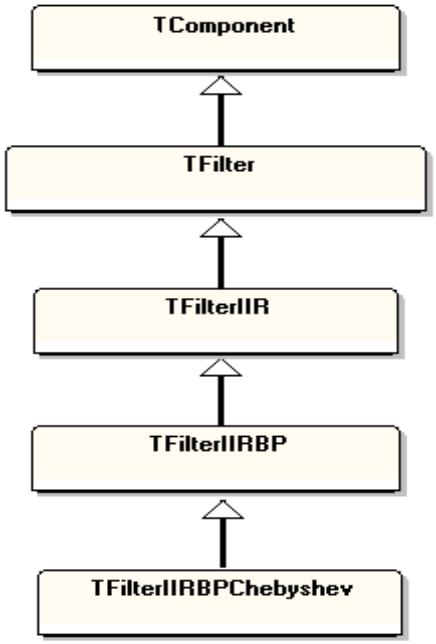


Figura 14 Clase TFilterIIRBPChebyshev

Propiedades

property Ripple: La ondulación en la banda de paso en dB.

Método

procedure SetRipple(ARipple: TAS_Sample)

Permiter especificar el ripple.

procedure SetupForChebyshev()

Para el funcionamiento de los filtros Chebyshev.

(*) Se redefine el método Setup()

Clase TFilterIIRBPBessel

Diseño

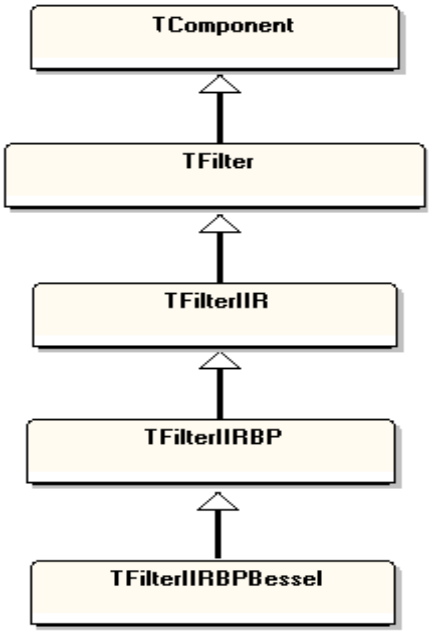


Figura 15 Clase TFilterIIRBPBessel

Método

Se redefine el método Setup()

Clase TFilterIIRBSButterworth

Diseño

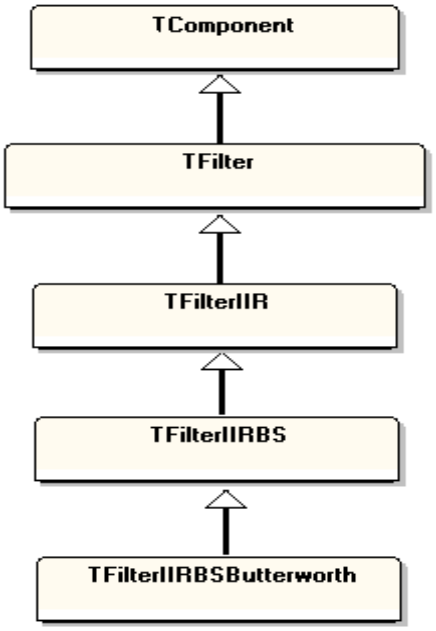


Figura 16 Clase TFilterIIRBSButterworth

Método

Se redefine el método Setup()

Clase TFilterIIRBSChebyshev

Diseño

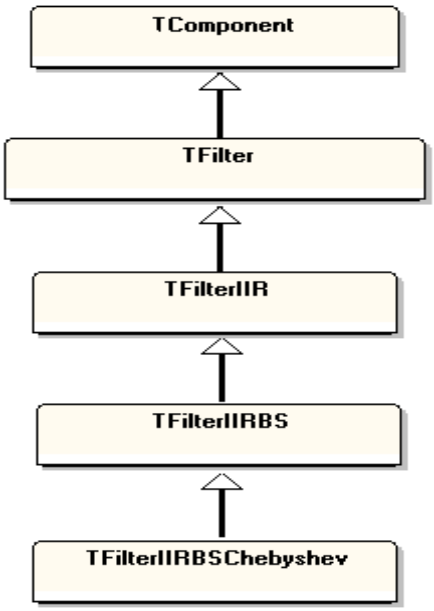


Figura 17 Clase TFilterIIRBSChebyshev

Propiedades

property Ripple: La ondulación en la banda de paso en dB.

Método

procedure SetRipple(ARipple: TAS_Sample)

Permiter especificar el ripple.

procedure SetupForChebyshev()

Para el funcionamiento de los filtros Chebyshev.

(*) Se redefine el método Setup()

Clase TFilterIIRBSBessel

Diseño

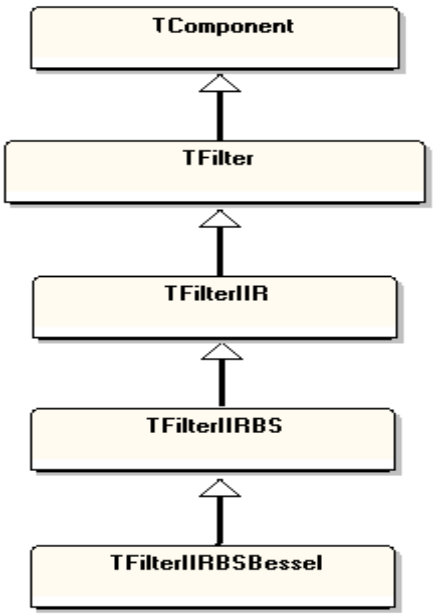
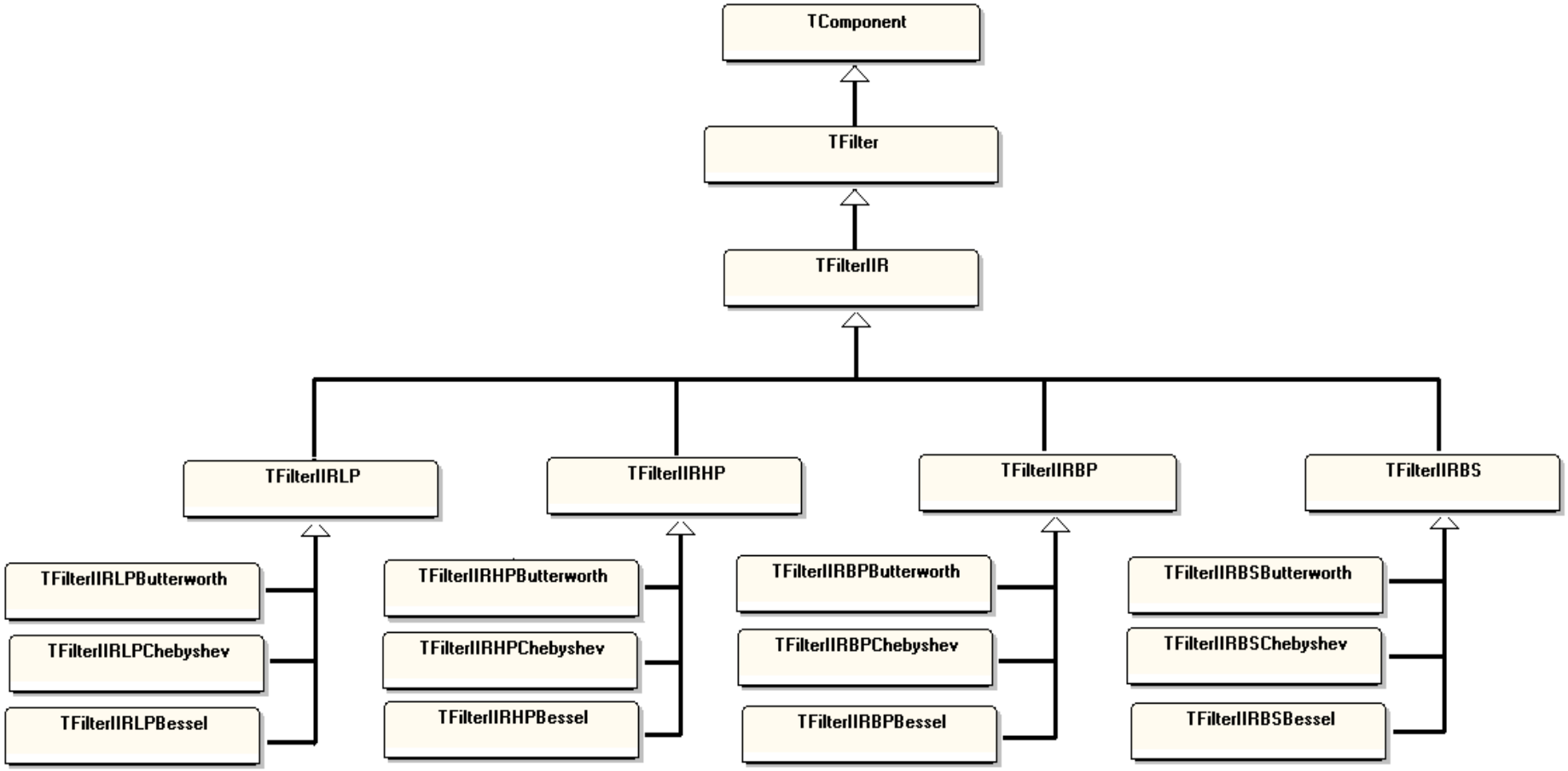


Figura 18 Clase TFilterIIRBSBessel

Método

Se redefine el método Setup()

Diseño Global de Clases