UNIVERSITY OF GAZİANTEP

GRADUATE SCHOOL OF

NATURAL & APPLIED SCIENCES

A NEW AND EFFICIENT WINDOW FUNCTION FOR DIGITAL FIR FILTER DESIGN

M.Sc. THESIS

IN

ELECTRICAL AND ELECTRONICS ENGINEERING

BY

HAKAN KAPLAN

JUNE 2015

A New and Efficient Window Function For Digital FIR Filter Design

M.Sc. Thesis

in

Electrical and Electronics Engineering

University of Gaziantep

Supervisor

Prof. Dr. Arif NACAROĞLU

by

Hakan KAPLAN

June 2015

REPUBLIC OF TURKEY
UNIVERSITY OF GAZİANTEP
GRADUATE SCHOOL OF NATURAL & APPLIED SCIENCES
ELECTRICAL&ELECTRONICS ENGINEERING

Name of the thesis: A new and efficient window function for digital FIR filter design

Name of the student: Hakan KAPLAN

Exam date: 15/06/2015

Approval of the Graduate School of Natural and Applied Sciences

Prof. Dr. Metin BEDİR
Director

I certify that this thesis satisfies all the requirements as a thesis for the degree of Master of Science

E. Erçelebi

Prof. Dr. Ergun ERÇELEBİ
Head of Department

This is to certify that we have read this thesis and that in our consensus/majority opinion it is fully adequate, in scope and quality, as a thesis for the degree of Master of Science.

A. Nacaroğlu

Prof. Dr. Arif NACAROĞLU
Supervisor

Examining Committee Members:	Signature
Assoc. Prof. Dr. Müslüm ARKAN	....
Prof. Dr. Rauf MİRZABABAYEV	....
Prof. Dr. Arif NACAROĞLU	A. Nacaroğlu
Assist. Prof. Dr. Nurdal WATSUJI	....
Assist. Prof. Dr. Taner İNCE	....

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Hakan KAPLAN

ABSTRACT

A NEW AND EFFICIENT WINDOW FUNCTION FOR DIGITAL FIR FILTER DESIGN

KAPLAN, Hakan
M.Sc. in Electrical and Electronics Engineering
Supervisor: Prof. Dr. Arif Nacaroğlu
June 2015
42 pages

In this thesis, a new window function for digital FIR filter in addition to studied in literature is presented. Some of the well-known and widely used window functions given in the literatures are Kaiser Window, Cosine hyperbolic window, Exponential window, Saramaki window, Ultraspherical window, etc. Each window function has advantages and disadvantages compared to their specific application areas.

Digital filters are commonly used for application of digital signal processing. The linear phase characteristics of the finite impulse response (FIR) filters make them preferable to infinite impulse response (IIR) filters although their lengths are more. The stability property of FIR filters is the another advantage.

There are some methods of designing the FIR filters starting with the given magnitude and/or phase specifications. The one of efficient method for FIR design is to use the Fourier Series. Since the Fourier Series representation of the specified characteristics are infinite and the FIR filters have finite length, to prevent the Gibbs' Oscillations, the window function are used. Briefly, combination of modified Cosh window and Blackman window is proposed in this thesis. Especially, sharper transient and lower magnitude of attenuation are obtained. To find best optimized values, most of mathematical calculations are analyzed, and then some specific parameter values are found for proposed window function**.**

Key Words: Finite impulse response (FIR) digital filters, window methods, Blackman window, modified Cosh window.

ÖZET

SAYISAL FIR SÜZGEÇ TASARIMI İÇİN YENİ VE ETKİLİ BİR PENCERE FONKSİYONU

KAPLAN, Hakan
Yüksek Lisans Tezi, Elektrik ve Elektronik Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Arif NACAROĞLU
Haziran 2015
42 sayfa

Bu tez çalışmasında, literatürde bilinen pencere fonksiyonlarına ek olarak SDT sayısal filtreler için yeni bir pencere fonksiyonu önerilmiştir. Literatürde iyi bilinen ve sıklıkla kullanılan pencere fonksiyonlarına örnek olarak Kaiser penceresi, Kosinüs hiperbolik penceresi, Üstel penceresi, Saramaki penceresi, Ultraspherical penceresi, v.b. Herbir pencere fonksiyonun özel uygulama alanlarına göre avantaj ve dezavantajları vardır.

Sayısal sinyal işleme uygulamaları için sayısal filtreler yaygın olarak kullanılmaktadır. Sonlu darbe tepkili (SDT) olanların filtre uzunluklarının daha uzun olmasına rağmen, doğrusal faza sahip olmalarından ötürü sonsuz darbe tepkili olanlara göre tercih edilirler. SDT filtrelerin diğer bir avantajı da kararlı olmalarıdır.

SDT filtrlerin tasarımı yapılırken, başlangıçta verilen büyüklükte ve/veya faz özelliklerine göre bazı metotlar vardır. SDT filtreler için verimli metotlardan biri Fourier Serileri kullanılmasıdır. Çünkü belirtilen karakteristikde Fourier Serilerinin sonsuz olması ve SDT filtrelerinin sonsuz uzunluğu olmasından, Gibbs Salınımlarını önlemek için pencere fonksiyonları kullanılır. Kısaca, çalışmalarımda modifiye edilmiş Cosh penceresi ve Blackman pencerelerinin birleştirilmesinden yeni bir pencere fonksiyonu önerildi. Özellikle daha dik geçiş ve zayıflama büyüklüğü yönlerinden iyi sonuçlar elde edildi. Yeni önerilen pencere fonksiyonu için en iyi iyileştirilmiş değerleri bulmak için birçok matematiksel hesaplamalar analiz edilerek, bazı değerler bulundu.

Anahtar Kelimeler: Sonlu darbe tepkili (SDT) sayısal filtreler, pencereleme yöntemleri, Blackman penceresi, değiştirilmiş Cosh penceresi.

DEDICATION

To My Family

ACKNOWLEDGEMENTS

Firstly, the author wishes to express his deepest gratitude to his supervisor Prof. Dr. Arif Nacaroğlu for his guidance, advice, criticism, encouragements and insight step by step during the my master thesis.

I would like to also thank Assist. Prof. Dr Göksel Durkaya from Department of Metallurgical and Material Science Engineering in Atilim University and Assist. Prof. Dr. Hüseyin Kurtuldu from Department of Biomedical Engineering in Başkent University for their guidance, support and suggestions. And thanks to my important friend, Gökhan Selamet from Ph.D. level in Department of Bioinformatics in Gaziantep University for his help and advice.

Lastly, specials thanks to, of course, my family to grow up me and their supportance of my all education levels.

TABLE OF CONTENTS

LIST OF FIGURES

Page

LIST OF SYMBOLS AND ABBREVIATIONS

n	Discrete time index
w	Angular frequency in rad/sample
z	Index for z frequency domain
s	Index for Laplace transform
T	Sampling period in rad/sample
w_s	Sampling Frequency
N	Window or filter length
w_m	Mainlobe width
S	Sidelobe roll-off ratio in dB
R	Ripple ratio in dB
H(z)	Filter transfer function
H(s)	Filter transfer function in s domain
$H(e^{jwT})$	Frequency spectrum of filter
$\Theta(w)$	Phase response
M(w)	Amplitude response
$W(e^{jwT})$	Frequency spectrum of window
w_p	Filter passband frequency in rad/sample
w_s	Filter stopband frequency in rad/sample
w_{mp}	Proposed window function
α, β, γ	Independent parameters of proposed window function
FIR	Finitie impulse response
IIR	Infinite impulse response
DSP	Digital signal processing

CHAPTER 1

INTRODUCTION

1.1 Background

Digital filters are important and commonly used for digital signal processing (DSP). They are mainly divided into two parts, one of them is finite impulse response (FIR); other one is infinite impulse response (IIR) filters. This classification depends on their impulse responses [1].

Depending on the characteristic of the processed signal, both filters have very wide application areas. In some cases FIR filters are preferred due to their linear phase (constant delay) and stability features [2]. But when we need faster calculations, IIR filters are used owing to their shorter lengths although they may not have stable poles and constant delay [3].

Generally, in the design of IIR filters, analog approximation or numerical optimization methods are used. But since FIR filters are nonrecursive, four different design methods are given in the literature. These methods are Fourier Series with windowing, discrete Fourier transform, optimization method and numerical method. Among these methods each has some superiorities. Comparing to others, Fourier Series approach is analytical and it gives symmetric coefficients. Optimization method has more computations, so it is not commonly used in real applications [3, 8].

Fourier series with windowing technique is most appropriate way to design FIR filters. Due to the difficulties of reduction of the Gibss' oscillation effect, some insignificant terms in the series expansion are vanished (rectangular windowing). But it has been shown that the modification of the significant coefficients multiplying them with suitable weighting values makes the characteristics of the designed filter more close to the specified characteristic. [4, 30].

1.2 Problem Definition

Window functions are categorized as adjustable and fixed according to their parameters in the function. Ideally window functions must be rectangular with infinite length. Since it is impossible to realize infinitely long the FIR filters, multi parameter window functions are used to reduce the deviation from the idealities. Thus they have flexible properties for some special applications. In literature, many window functions are proposed. Problem is that, the minimization of filter order required for satisfying given prescribed specification. Higher order filters need more circuit elements, more mathematical calculations, longer processing time, more product quantization error, etc. For that reason, filter designer focuses on finding new adjustable window functions to improve filter specifications with lower order [5, 27].

1.3 Thesis Objective

Thesis objective is to find a new and efficient adjustable window function for FIR filter design to obtain higher quality nonrecursive digital filters. FIR filter coefficients are calculated using Fourier Series method with windowing technique [3].

The new proposed window function is the combination of modified Cosh window and Blackman window. Modified Cosh window has adjustable parameters. Blackman window is fixed window. As a result of combination the new window function given in this study has four independent parameters. Since the parameters are also independent to each other, it is easy to adjust for the best solution [6].

1.4 Literature Survey

Firstly, Gibbs explained that, direct truncation of Fourier Series causes oscillations and he defined the mathematics of the problem in 1899. This phenomenon is called as Gibbs' Oscillation [7].

In 1900, Fejer proposed a method to reduce Gibbs' oscillation for some applications. Later Lanczos presented a new method with some better features compared to Fejer method. In his study the function has only one discontinuity. Gibbs oscillation is characteristic of any truncated Fourier Series regardless of number of discontinuities [8, 26].

In 1946, Dolph proposed Dolph-Chebyshev window function. This window has narrow mainlobe for specified maximum sidelobe compared to other windows function [9]. In 1966, Kaiser explained a window function better than Dolph' window which has a maximum energy level for mainlobe [10].

In 2001, Ultraspherical windows were proposed by Deczky to design nonrecursive digital filters. This window success lower order filter length compared to other windows [7, 11]. Other well-known windows are Rectangular, Von Hann, Hamming, Bartlet, Blackman which are also known as fixed windows.

In the last decade, some efficient window functions are studied by many scientists. Avci has shown that the spectral structures of window functions are highly effective for window functions and narrower spectral results with better quality [7, 18, 27]

1.5 Scope of Thesis

This thesis is organized totally in five chapters. Each chapter is briefly explained below:

Chapter 1. Introduction - In this chapter, the work in the thesis is overviewed. Problem definition, research objective, prior works and thesis structure are given in this chapter.

Chapter 2. Review of Digital Filters – Mainly, digital filters are defined. Types of digital filter are expressed. Advantages and disadvantages of Finite impulse response and infinite impulse response filters are given.

Chapter 3. FIR Filter Design using Windowing Method– In this chapter, the mathematical background of the Fourier Series is given and the application of these series to design FIR filters is presented. The effect of the windowing of series coefficients is given.

Chapter 4. Proposed Window Function – A new proposed function feature is mathematically expressed and it is applied to some specific examples to show its positive and negative sides compared to Blackman window. The comparison is repeated changing the parameters of the window function.

Chapter 5. Conclusions and Future Plan – The results of the application of the window function to lowpass, highpass, bandpass filter design are given in this chapter and some problems are classified for future studies.

CHAPTER 2

REVIEW OF DIGITAL FILTERS

2.1 Introduction to Digital Filters

The digital filter transforms numerical procedure or algorithm in a given sequence of numbers to second sequence of numbers that has some desirable outputs. Desirable output means that, what kind of application is used for designing filters [12]. Digital filter has frequency sensitivity, so they are used as pass or reject signal from input frequencies.

According to their operations, filters are classified as four types in below;

- Lowpass filters: only low frequency passes below cutoff frequency.
- Highpass filters: only high frequency passes above cutoff frequency.
- Bandpass filters: passes frequency inside certain band.
- Bandstop filters: passes frequency outside certain band.

According to working type and physical shape, filters are divided into two groups as analog filters and digital filters. Mainly, analog filters consist of passive and active electronic components. For this filter, operational amplifiers, capacitors, resistors, inductors are typical components. Analog filter circuits can select any frequency range either from the current or voltage signals [13].

On the other hand, digital filter uses digital processor to get filter effect, for instance, personal computer and digital signal processing chips. Hence, the analog input signal must be sampled and digitized using analog to digital converter for numerical calculations of the signal. Obtained numerical values refer to binary numbers of input signal transferred to digital processor [14].

Advantages of digital filters over analog filters as listed below;

- Digital filters can be easily implemented and tested by digital processor.
- Digital filters are more stable than analog filters.
- Digital filters are programmable thus; it can be changed immediately when it is necessary using software on the processor. This operation is so difficult or takes more time to change when using analog filters.
- Digital filters work well for low frequency level signals compared analog filters [3].

According to their impulse response, digital filters are implemented into two methods which are finite impulse response (FIR) filters and infinite impulse response (IIR) filters [15].

2.2 Finite Impulse Response (FIR) Filter

These types of filters are also called as nonrecursive digital filters. Mainly, finite impulse response digital filter transfer function can be expressed as,

$$H(z) = \sum_{n=0}^{N-1} h(nT) z^{-n} = \sum_{k=0}^{N-1} a_k z^{-k} \tag{2.1}$$

where N is length and $h(nT)$ is impulse response of the filter [3].

The filter frequency response is expressed,

$$H\left(e^{jwT}\right) = M(w) e^{j\Theta(w)} = \sum_{n=0}^{N-1} h(nT) e^{-jwnT} \tag{2.2a}$$

$M(w)$ and $\Theta(w)$ are amplitude and phase responses of the filter as

$$M(w) = \left|H(e^{jwT})\right| \tag{2.2b}$$

$$\Theta(w) = arg\ H\left(e^{jwT}\right) \tag{2.2c}$$

The direct and famous methods used to design finite impulse response filter are listed below:

- Fourier series method with window technique
- Discrete Fourier transform method
- Numerical methods

- Optimization method [3, 28].

Fourier series method with window technique is closed form and it is easy, required minimum mathematical computations. But the designs are suboptimal with respect to filter complexity whereby the filter is assumed to be optimal when the filter order is lowest for designing required filter properties [3, 27].

Discrete Fourier transform method is different than other method that is useful for any given magnitude. This method is used for designing non-prototype filters where desired magnitude response can take any irregular shape. There are some disadvantages of this method.

Numerical methods, that use numerical formulas to design finite impulse response filter, can perform integration, differentiations and interpolation. Generally interpolation formulas are used and most commonly used formulas are Bessel, Gauss, Gregory-Newton, Stirling and Everret [3].

Lastly, optimization method is optimal solution for designing nonrecursive digital filters. This method is used to find filter coefficients when error is minimized but a large amount of computations are necessary [16].

2.3 Infinite Impulse Response (IIR) Filter

This type of filters is also called as recursive digital filters. Mainly, infinite impulse response digital filter transfer function is expressed as,

$$H(z) = \frac{\sum_{k=0}^{M} a_k z^{-k}}{\sum_{k=1}^{L} b_k z^{-k}} \tag{2.3}$$

To achieve causality property, degree of the numerator L must be smaller than degree of the denominator [17].

There are two types of methods to design infinite impulse response filters which are indirect and direct. In indirect method, some transformation functions are used to convert analog filter into digital filters. It means that, discrete-time transfer function H(z) is obtained replacing s with equivalent f(z) in continuous-time transfer function H(s).

In this method, obtained continuous-time transfer function that satisfies certain specifications using one of the analog filter types such as Chebyshev and Butterworth is converted into digital form using one of the methods given below:

- Bilinear transformation
- Invariant impulse-response method
- Matched-z transformation
- Modified invariant impulse-response method [3. 7].

2.4 Difference between FIR and IIR Digital Filters

The advantages of FIR over IIR filter are,

- The phase of FIR filter is strictly linear. Digital audio and image processing, biomedicine and data transmission uses this type of filters. But the IIR filter is not.
- To design FIR filter, Fast Fourier Transform is used.
- The FIR filter is non-recursive structure so finite precision error is quite small, but IIR filter is recursive type so some parasitic oscillation can occur.
- FIR filters guarantees stability, but IIR is not.
- Effect of using limited number of bits to implemented filter such as round off noise and coefficient quantization errors are much less severe in FIR filter compared to IIR filter.
- FIR filter is easy and convenient to implement [14. 19].

Disadvantages of FIR filter over IIR filter are,

- IIR filter can use formulas, data and tables of the analog filter, small calculation is necessary, but FIR filter uses computer to obtain calculations, filter length should be larger for efficient values.
- FIR filters require more filter coefficients for sharp cutoff compared to IIR. It means that, more processing time and storage is needed to perform FIR filters [14. 19].

CHAPTER 3

FIR FILTER DESIGN USING WINDOWING METHOD

3.1 Design with Fourier Series

Finite impulse response digital filters frequency response is periodic function of ω with period ω_s. It can be defined as a Fourier Series and can be written as [20]:

$$H\left(e^{-j\omega t}\right) = \sum_{n=-\infty}^{\infty} h(nT)e^{-j\omega nt} \tag{3.1a}$$

where

$$h(nT) = \frac{1}{w_s}\int_{-w_s/2}^{w_s/2} H\left(e^{j\omega t}\right)e^{j\omega nt}\, dw \tag{3.1b}$$

Replacing $e^{j\omega t} = z$, Eq. (3.1a) becomes,

$$H(z) = \sum_{n=-\infty}^{\infty} h(nT)z^{-n} \tag{3.2}$$

In a given frequency spectrum, $H(e^{j\omega t})$, using Eq. (3.1b) and Eq. (3.2) the transfer function can be found. But this transfer function is noncasual and it has infinite order. To get finite order transfer function, the series in Eq. (3.2) truncated as;

$$H(z) = \sum_{n=-(N-1)/2}^{(N-1)/2} h(nT)z^{-n} \tag{3.3}$$

where $h(nT) = 0$ for $|n| > (N-1)/2$.

Using series property, Eq. (3.3) can be converted into;

$$H(z) = h(0) + \sum_{n=1}^{(N-1)/2}[h(-nT)z^{n} + h(nT)z^{-n}] \tag{3.4}$$

In order to get causal transfer function, $H(z)$ is multplied by $z^{-(N-1)/2}$. This process causes group delay by amount of $(N-1)T/2$ [20].

3.2 FIR Filter Design with Window Technique

Main idea behind windowing technique is to choose a proper ideal frequency-selective filter (which always has noncasual and infinite-duration impulse response) then truncate (or window) its impulse response to obtain linear phase and casual finite impulse response filter. Therefore, phenomenon of this method is on selective and appropriate window function and appropriate ideal filter [21].

Window method involves a function called window function or apodization function which states that if some interval is chosen, it returns with finite nonzero value inside that interval however zero values outside the interval. When the window with chosen length is applied on IIR system, it will obviously return with a finite non-zero value inside the interval producing a FIR system and all other value that are outside the interval will be zero. We can say that, finite impulse response inside some predefined interval is viewed [22].

There are four steps to design FIR filters using window:

- Assuming idealized frequency response, then using Eq. (3.1) an idealized infinite-duration noncasual filter is obtained.
- According to property, appropriate window is selected.
- The window function constructed and applied
- Obtained finite-duration noncasual filter is converted into casual filter [7].

3.3 Window Method

In signal processing, window function is a mathematical function that is zero-valued the outside some chosen interval [22]. In Figure 3.1, passband, transition band and stopband region of low pass filter is shown. In the figure, vertical axis shows magnitude (or gain) and horizontal axis shows frequency values. The frequency ω_p demonstrates edge of passband, frequency ω_s demonstrates edge of stopband. And difference between stopband frequency ω_s and passband frequency ω_p defines width of the transition band. It means that, $\omega_t = \omega_s - \omega_p$.

Passband ripple of the filter is denoted as δ_p and magnitude of filter varies between 1- δ_p and 1+ δ_p, in addition, stopband ripple is denoted as δ_s [19].

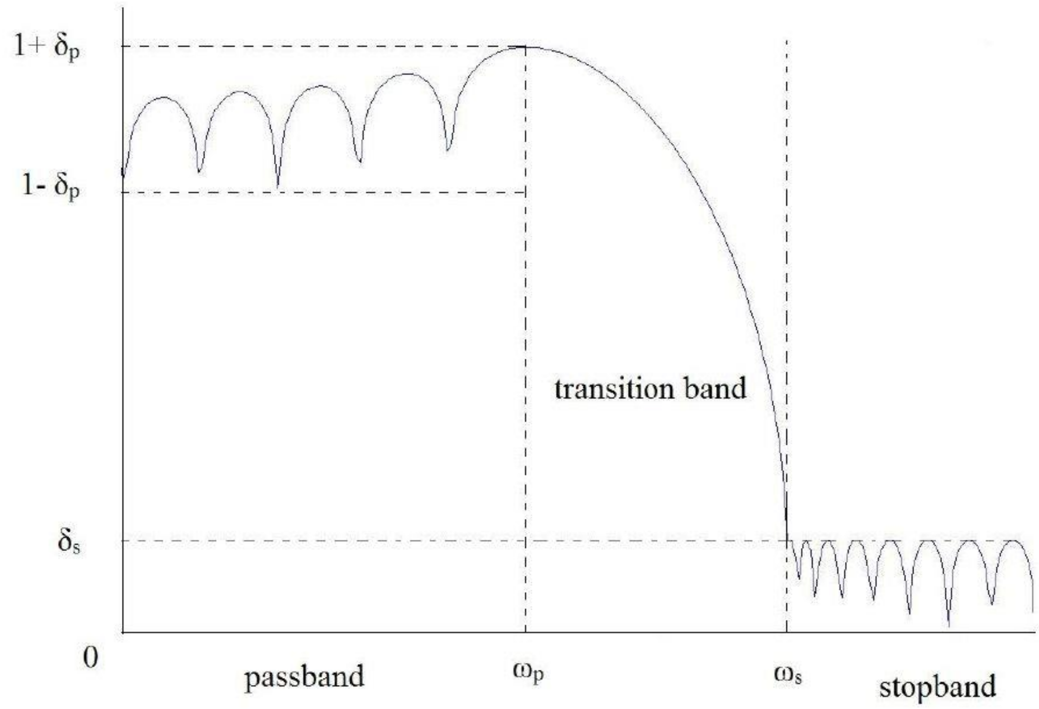

Figure 3.1 Magnitude and frequency response of lowpass filter [19]

The stopband and passband oscillations observed because slow convergence or direct truncation in the Fourier Series, these oscillations are known as Gibbs' oscillations. To get better filter characteristics, Gibbs' oscillations should be reduced. An alternative and easy way to reduce these oscillations is to use the windowing [3.19].

Window function is applied as; a window w(nT) with length of N is non-zero for $|n| \leq (N-1)/2$ and zero for otherwise. Using window, impulse response of noncasual filter is calculated in Eq. (3.5)

$$h_w(nT) = w(nT)h(nT) \tag{3.5}$$

In frequency domain representation can be written as Eq. (3.6a),

$$H_w\left(e^{j\omega T}\right) = \mathrm{W}(\mathrm{e}^{\mathrm{j}\omega\mathrm{T}})\mathrm{H}(\mathrm{e}^{\mathrm{j}\omega\mathrm{T}}) \tag{3.6a}$$

Frequency representation of window with length of N and range with $|n| \leq (N-1)/2$ can be written as,

$$W(e^{j\omega T}) = \sum_{n=-(N-1)/2}^{(N-1)/2} w(nT)e^{-jwnT} \quad (3.6b)$$

3.4 Window Spectral Characteristics

In figure 3.2, a window has a normalized amplitude spectrum in dB range is shown. Known window spectral parameters are mainlobe width (w_m), the ripple ratio (R) and sidelobe roll-off ratio (S). And these parameters define:

- Mainlobe width, w_m=2 ωr rad/sec
- Ripple ratio, R=Maximum sidelobe amplitude-mainlobe amplitude=S1 dB
- Sidelobe roll-off ratio,

S=Maximum sidelobe amplitude-minimum sidelobe amplitude= S1-SL dB [18, 23].

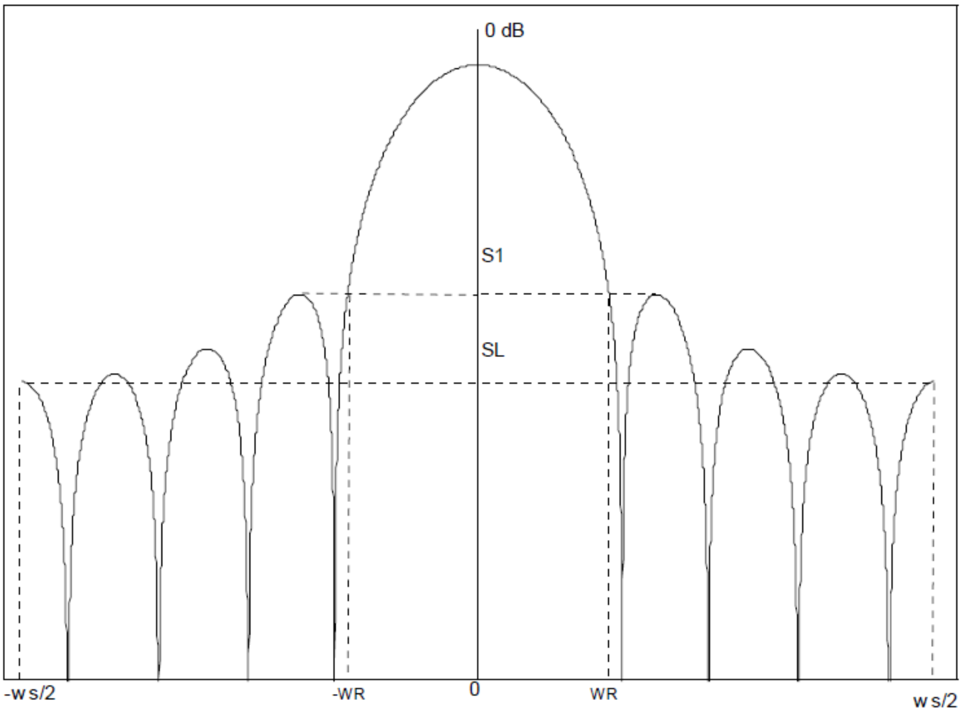

Figure 3.2 Normalized amplitude spectrum of window [7]

The following properties explain the features of the best window function.

- The mainlobe width should be narrow,
- Ripple ratio should be small,
- Sidelobe roll-off ratio should be large [18].

3.5 Well-known Window Functions

In literature, window functions are mainly divided into two groups which are fixed window and adjustable window. Fixed window has only one independent parameter which is called as window length. This parameter changes the mainlobe width. However adjustable window has two or more independent parameters, one of them is window length as fixed window but other parameters change some spectral characteristics of window [2].

3.5.1 Fixed Windows

Well-known fixed windows are Rectangular, Von Hann, Hamming and Blackman windows.

3.5.1.1 Rectangular Window

Rectangular window is simplest window function but provides the worst performance from the viewpoint of stopband attenuation [24]. It can be defined as;

$$w_r(n) = \begin{cases} 1, & |n| \leq \frac{N-1}{2} \\ 0, & otherwise \end{cases} \tag{3.7}$$

Figure 3.3 shows that rectangular window magnitude response for low pass filter [3].

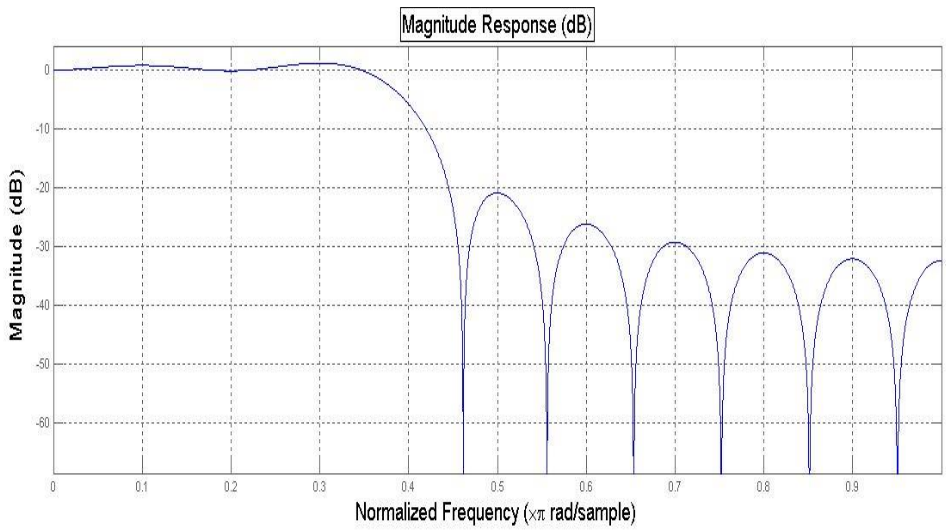

Figure 3.3 Rectangular window magnitude response, N=20

Rectangular window has narrowest mainlobe width and largest ripple ratio compared to other windows.

3.5.1.2 Von Hann Window

The Von Hann window is also known as raised cosine window and it can be defined as;

$$w_{vh}(n) = \begin{cases} 0.5 + 0.5\cos\frac{2\pi n}{N-1}, & |n| \le \frac{N-1}{2} \\ 0, & otherwise \end{cases} \tag{3.8}$$

This window has better ripple ratio than previous window but mainlobe width is wider. In figure 3.4 shows that, Von Hann window magnitude response for low pass filter [7].

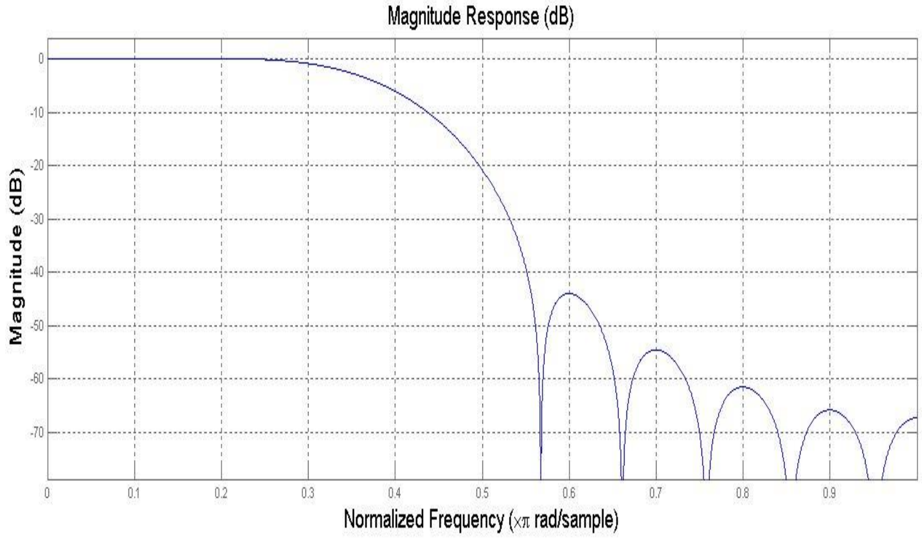

Figure 3.4 Von Hann window magnitude response, N=20

3.5.1.3 Hamming Window

The Hamming window is similar to Von Hann but to get better ripple ratio, equation is modified and it can be defined as;

$$w_h(n) = \begin{cases} 0.54 + 0.46\cos\frac{2\pi n}{N-1}, & |n| \le \frac{N-1}{2} \\ 0, & otherwise \end{cases} \tag{3.9}$$

Figure 3.5 shows Hamming window magnitude response for low pass filter. This window has better ripple ratio than Von Hann Window and has almost same mainlobe width [7].

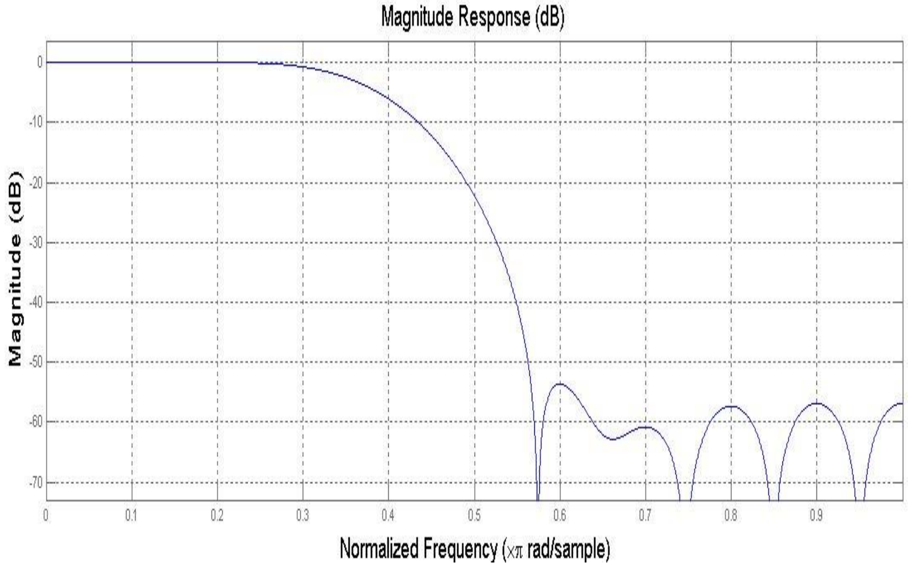

Figure 3.5 Hamming window magnitude response, N=20

3.5.1.4 Blackman Window

The Blackman Window has some cosine term compared to other window to get better ripple ratio characteristics. It can be defined as;

$$w_b(n) = \begin{cases} 0.42 + 0.5\cos\frac{2\pi n}{N-1} + 0.08\cos\frac{4\pi n}{N-1}, & |n| \leq \frac{N-1}{2} \\ 0, & otherwise \end{cases} \tag{3.10}$$

This window has better ripple ratio compared to other fixed windows however it has worst (larger) mainlobe width. Figure 3.6 shows Blackman window magnitude response.

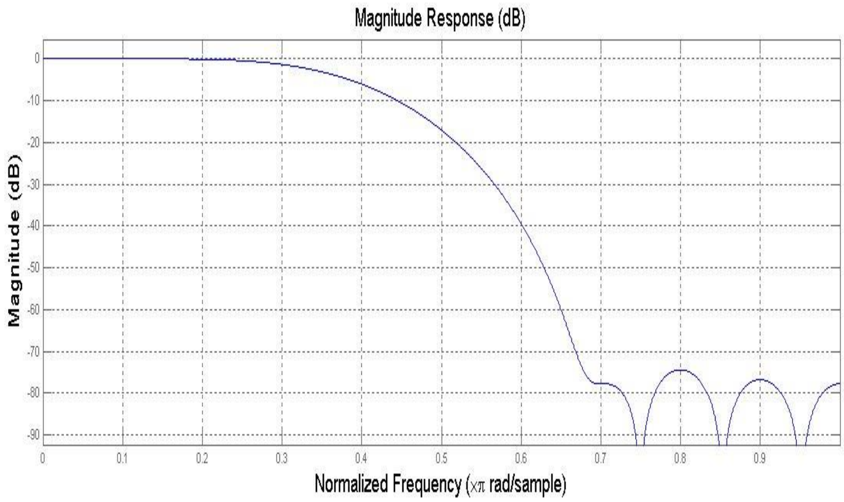

Figure 3.6 Blackman window magnitude response, N=20

3.5.2 Adjustable Windows

The well-known and widely used windows are Cosh Window, Kaiser Window and Saramaki Window.

3.5.2.1 Cosh Window

Similar to Kaiser Window, Cosh window is proposed based on cosine hyperbolic function. Cosine hyperbolic function is defined as;

$$\cosh(\mathrm{x}) = \frac{\mathrm{e}^{\mathrm{x}} + \mathrm{e}^{-\mathrm{x}}}{2} \tag{3.11}$$

Cosh window works faster than Kaiser because it has no power series expansions [7, 27]. The function of the window can be expressed as;

$$w_c(n) = \begin{cases} \dfrac{cosh\left(a_c\sqrt{1-\left(\frac{2n}{N-1}\right)^2}\right)}{cosh(a_c)}, & |n| \leq \frac{N-1}{2} \\ 0, & otherwise \end{cases} \tag{3.12}$$

In Eq. (3.12), Cosh window consists of two independent parameters, one of them is called as window length and other is a_c adjustable parameter. Changing a_c parameter, mainlobe width and ripple ratio changes. Comparing with Ultraspherical Window, it has narrower mainlobe width and smaller side-lobe roll of ratio but Cosh window has better result for wider mainlobe and larger side-lobe roll of ratio. Figure 3.7 shows Cosh window magnitude spectrum for different a_c adjustable values [7, 24].

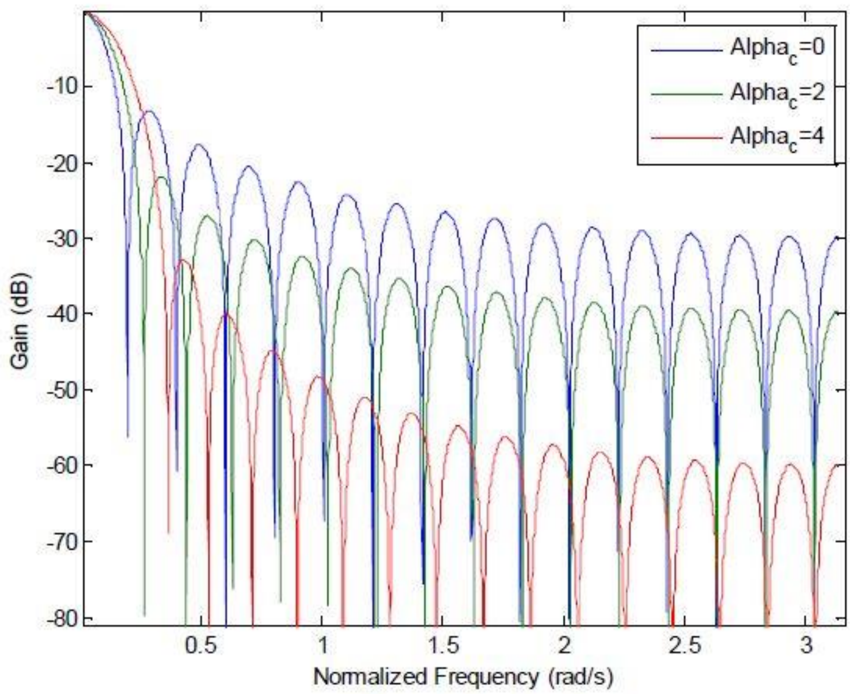

Figure 3.7 Cosh window magnitude spectrum $a_c = 0{,}2{,}4$; N=31 [7]

3.5.2.2 Kaiser Window

The Kaiser window is widely used in practice which is proposed by J. F. Kaiser and it is given by;

$$w_k(n) = \begin{cases} \dfrac{I_0\left(\beta\sqrt{1-\left(\frac{2n}{N-1}\right)^2}\right)}{I_0(\beta)}, & |n| \leq \frac{N-1}{2} \\ 0 & , \; otherwise \end{cases} \tag{3.13}$$

where I_0 is modified zero Bessel function and it is positive so power series expansion is necessary. The adjustable parameter β denotes that minimum stopband attenuation and it is chosen for transition width [21]. Figure 3.8 and 3.9 show Kaiser window spectrum for $\beta = 0.5$; and N=20 and N=100, respectively.

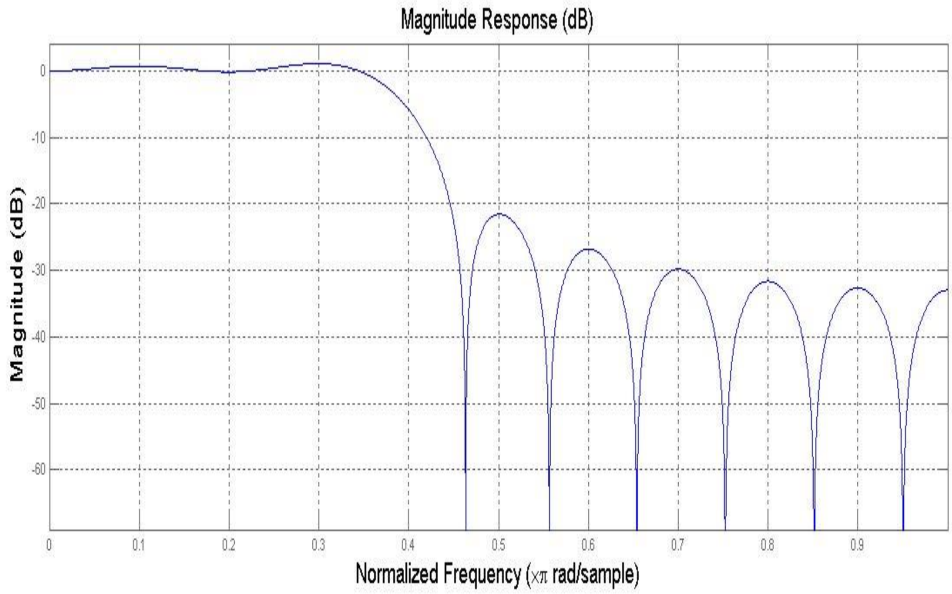

Figure 3.8 Kaiser window magnitude spectrum $\beta = 0.5$; N=20

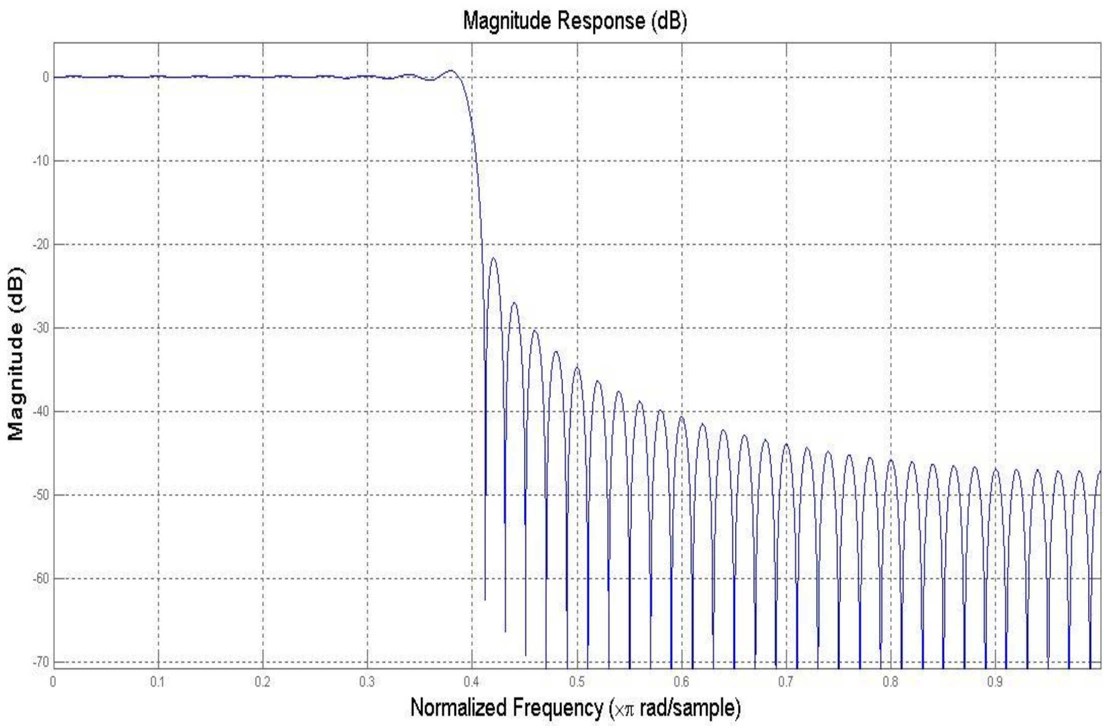

Figure 3.9 Kaiser Window magnitude spectrum $\beta = 0.5$; N=100

Comparing sidelobe roll-off ratio, Kaiser Window provides better result compared to other adjustable windows. Changing β value, mainlobe width and ripple ratio of window magnitude response changes [18].

3.5.2.3 Saramaki Window

The Saramaki window is proposed by Saramaki and it is approximation to discrete prolate functions which is about minimizing the sidelobe energy. It can be defined as in time domain [7, 25];

$$w_s(n) = \begin{cases} \hat{w}(n)/\hat{w}(0), & |n| \leq \frac{N-1}{2} \\ 0 & , \ otherwise \end{cases} \tag{3.14a}$$

where

$$\widehat{w}(n) = v_0(n) + 2\sum_{k=1}^{(N-1)/2} v_k\,(n) \tag{3.14b}$$

$$v_0(n) = \begin{cases} 1, & n = 0 \\ 0, & otherwise \end{cases} \tag{3.14c}$$

This window has better sidelobe energy than Kaiser Window and mathematically power series expansion is not necessary. Main disadvantage is that, in time domain representation, it has recursive equation [25].

CHAPTER 4

PROPOSED WINDOW FUNCTION

4.1 Introduction

Digital filters can be implemented as finite impulse response and infinite impulse response. Finite impulse response filter is studied and there are some methods to design this kind of filter. Fourier series with windowing technique is selected for designing finite impulse response digital filter in this thesis.

The proposed new window function mainly consists of two known window functions. One of them is Cosine Hyperbolic (shortly Cosh) window function which is known as adjustable window and other is Blackman window which is known as fixed window. This new window function has four independent parameters; these are α, β, γ and window length (N).

4.2 Proposed New Window Function

New window function is the combination of Cosh window function and Blackman window. Cosh window is known as adjustable window which is explained in section 3.4.2.1 and Blackman window is known as fixed window which is explained in section 3.4.1.4.

The new proposed window function has good result due to some window spectral characteristics. Cosh window is given in Eq. (3.12) and Blackman window is given in Eq. (3.10). They are successful for special filter design applications. Eq. (4.1) indicates proposed window function.

$$w_{mp}(n) = \begin{cases} \left(\dfrac{cosh\left(\gamma\left(1-\left(\frac{2n}{N-1}\right)^2\right)^{\beta}\right)}{cosh(\gamma)} \right)^{\alpha} & , \ |n| \leq \frac{N-1}{2} \\ 0 & , \ otherwise \end{cases} \tag{4.1}$$

Independent parameters, α, β, γ and window length (N) are changed in order to find optimal characteristics. If α, β and γ are all equal to zero, the window function becomes rectangular window. If $\alpha = 1$ and $\beta = 1/2$, it becomes Cosh window given in Section 3.5.2.1

4.3. Comparisons of Spectrum between Fixed Windows

Fixed windows are used for filter design but they have disadvantage. Because they have only window length parameter, it changes only mainlobe width so there is a limitation for designing filter. Some fixed windows such as Rectangular, Von Hann, Hamming and Blackman which are specified in section 3.4.1. The figure 4.1 denotes the spectral characteristics of these fixed windows. They are applied to design of low pass filter with $0.4 * \omega_s/2$ cut-off frequency and the gain of the filter for each case are shown in Fig. 4.1.

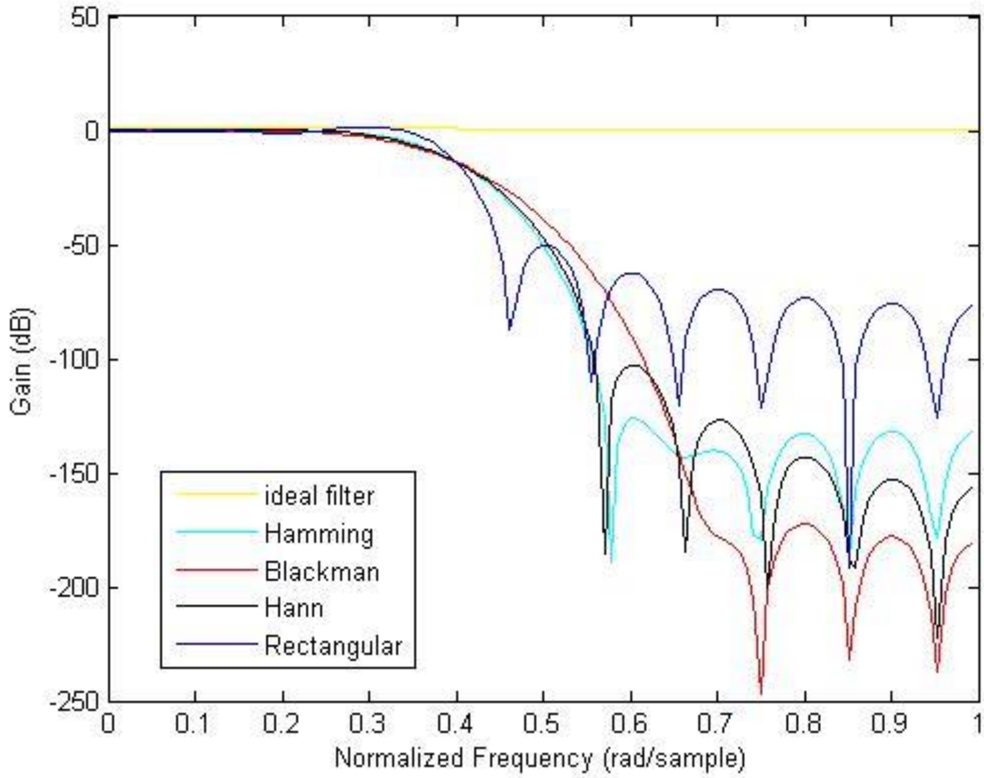

Figure 4.1 Fixed windows magnitude response comparisons, N=20

In Fig. 4.1 the horizontal axis defines normalized frequency (1 corresponds to the half of the sampling frequency ω_s) and vertical axis defines the gain of the low pass filter in dB scale. Yellow line denotes ideal low pass filter, blue line denotes rectangular window, cyan line denotes hamming window, black line denotes Von Hann window and red line denotes Blackman window applications.

As it is obvious in Fig. 4.1, the application of the rectangular window has sharper transient between pass band and stopband region however having longer and higher ripples in stopband region. On the other hand Blackman window has the minimum gain (maximum attenuation) in the stopband region although it results with wider (slower) transient. Hamming and Von Hann windows give almost similar transient and ripples in both regions. Hence if sharper passband is required, rectangular window is preferred but if stopband attenuation is more important, Blackman window is preferable

In Eq. (3.10), Blackman window has more term than Hamming window given in Eq. (3.9). So Blackman has more accurate results and cascading the cosine term into it reduces sidelobes. It means that, more efficient in terms of less power in sidelobe [22].

4.4 Comparisons of Proposed Window and Blackman Window for LPF

The gains of lowpass filter designed by Blackman window and both Blackman and proposed window are shown in Fig. 4.2 in black and red colors respectively. Proposed window function given in Eq. 3.10 is first applied to the design of the lowpass filter. The specification of the lowpass filter is taken as the ideal characteristics with $0.4 * \omega_s/2$ cut- off frequency where $\omega_s = {}^{2\pi}/_{T}$ the sampling frequency is and T is the sampling period. The given characteristic is first designed by using Fourier Series method and Blackman window is applied as default window in Matlab. The proposed window is subsequently is applied to modify the coefficients of the transfer function for different values of α, β, and γ. The gain of the resultant transfer function is compared with the ideal characteristics and the best of the gain is chosen. In the algorithm, α, β, and γ values are increased as 0.1 at each step of calculation between 0.1 and 3. The part of the program is:

```
for alpha = 0.1:0.1:3
    for beta = 0.1:0.1:3
        for gamma = 0.1:0.1:3
            . . .
end
      end
            end
```

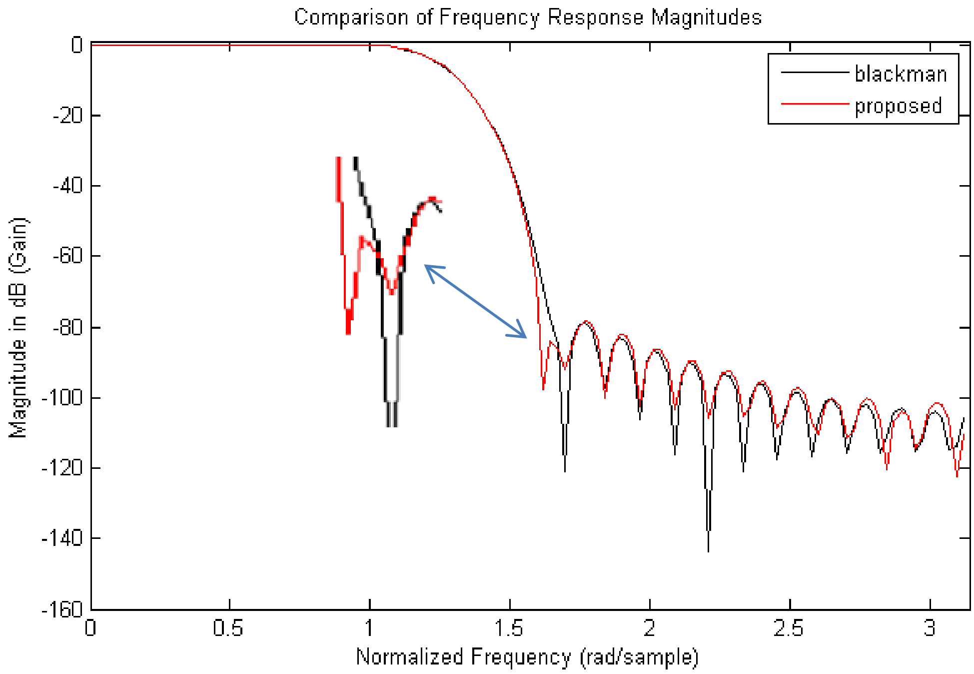

Figure 4.2 Spectrum Comparison of Proposed window and Blackman window, N=50

As a result of the program, the sharper transient for lowpass filter is found as $\alpha = 0.2$, $\beta = 1.3$, $\gamma = 0.5$ and the gain of the filter is shown in Fig. 4.2, comparing the result with Blackman window application. It is obvious that the first minimum of the gain for proposed window application happens at the frequency $(\omega_s/128)$ much lower than the Blackman window application. The order of the filter N is taken as 50.

The similar windows are applied for 100^{th} order filter length and sharper attenuation observed at lower frequencies are shown in Fig. 4.3. In this case stopband attenuations are also better for the application of the proposed window. In this figure, the first minimum in the gain happens at the frequency of $w_s/256\ rad/sec$ much smaller than the Blackman application. In this filter design, independent parameters are found as $\alpha = 0.6, \beta = 1.4$ and $\gamma = 0.2$.

The last applications of the low pass filter are designed for 150^{th} order length. The proposed window gives better result not only the cut-off sharpness but also the stopband attenuations. This is shown in Figure 4.4. The first minimum gain of the proposed window is smaller than Blackman window in terms of frequency $w_s/256\ rad/sec$. In here, proposed window parameters are: $\alpha = 0.3, \beta = 1.4$ and $\gamma = 0.3$.

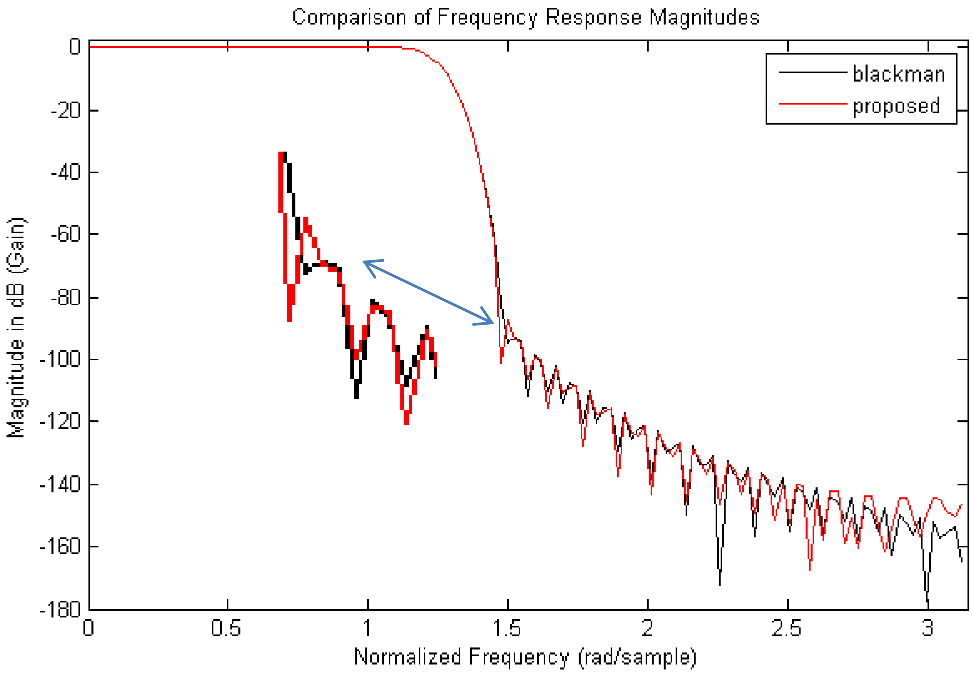

Figure 4.3 Spectrum Comparison of Proposed window and Blackman window, N=100

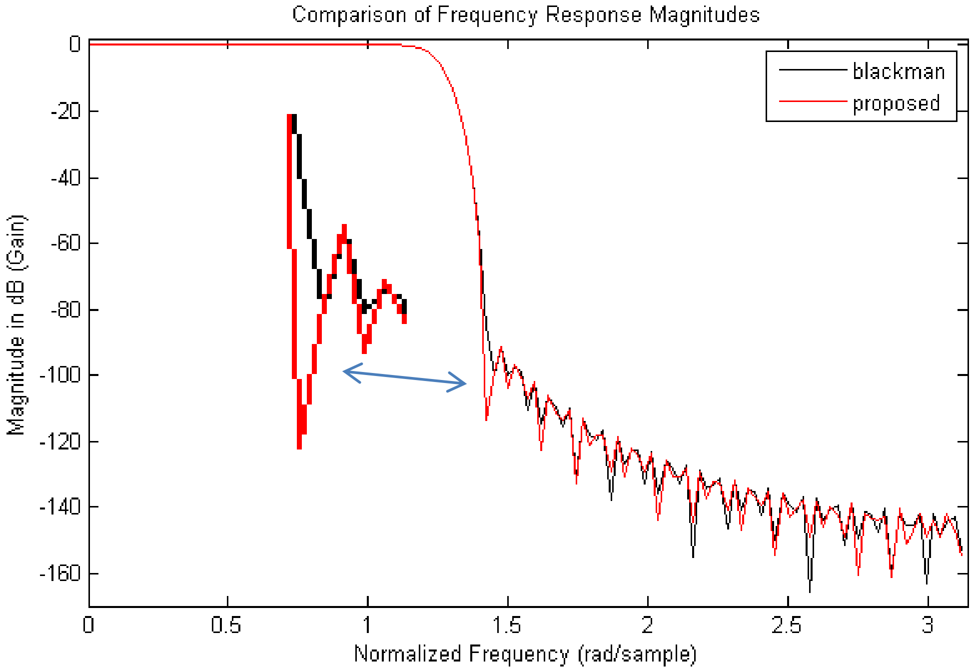

Figure 4.4 Spectrum Comparison of Proposed window and Blackman window, N=150

As seen in Fig. 4.2, Fig. 4.3 and Fig. 4.4, changing window length (N) doesn't affect the proposed window efficiency for designing lowpass filter. The results are investigated in terms of first minimum gain and sharpness of transient.

4.5 Application of Proposed Window for Different Filter Types

In section 4.4, proposed window is applied for low pass filter design. In this part, high pass filter and bandpass filter are designed using proposed window function. Fig. 4.5, Fig. 4.6 and Fig. 4.7 show the comparison of the high pass filter gains obtained applying Blackman window and proposed window for different filter length such as $N = 50,\ N = 100\ and\ N = 150$ respectively.

The specification of high pass filter is applied as ideal characteristics with $0.6 * \omega_s/2$ cut-off frequency. The first application of proposed window for the high pass filter is designed for 50^{th} order filter length. Proposed window has sharper transient than the Blackman window as in the Figure 4.5. The first minimum gain is obtained for the proposed window at the frequency $w_s/256$ higher than the Blackman window. In this design, independent parameters of proposed window are calculated as $\alpha = 0.4,\ \beta = 2.3$ and $\gamma = 0.1$.

Also the window function is designed for 100^{th} order filter length for application of high pass filter. Unlike the 50^{th}, the proposed window have better results not only for cut-off sharper transient but also stopband attenuation value in terms of first minimum of the gain. Figure 4.6 show the proposed window has higher cut-off frequency of $w_s/256$ than the Blackman window application. After cycle process for independent parameters of proposed window are found as $\alpha = 0.6,\ \beta = 2.3$ and $\gamma = 0.1$.

Another application of high pass filter is designed for 150^{th} order length as in the Figure 4.7. Similarly to other order filter lengths, red line gives better result for sharper transient between passband and stopband regions. The first minimum in the gain curve of red line is appeared at higher frequency $w_s/256$ than the black line. For this application, the parameters are simulated as $\alpha = 0.8,\ \beta = 2.3$ and $\gamma = 0.1$.

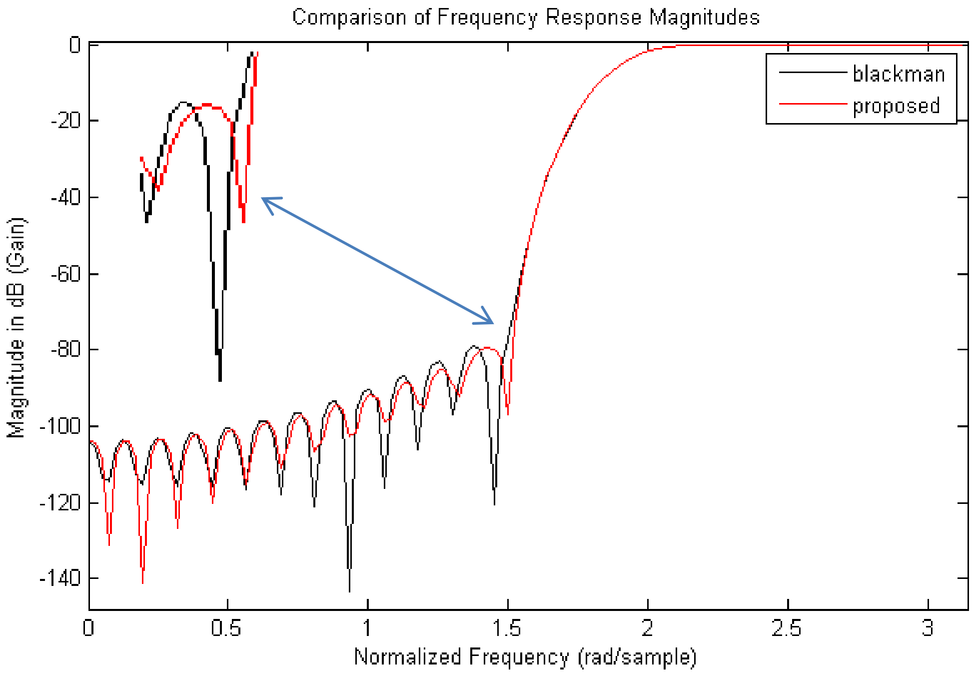

Figure 4.5 Spectrum Comparison of Proposed window and Blackman window for high pass filter, N=50

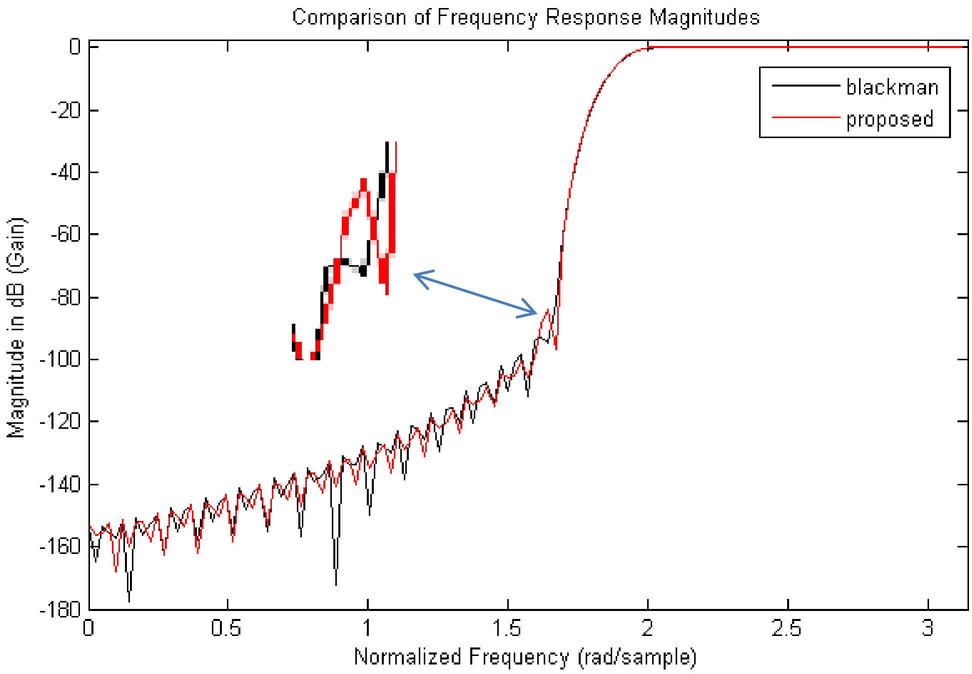

Figure 4.6 Spectrum Comparison of Proposed window and Blackman window for high pass filter, N=100

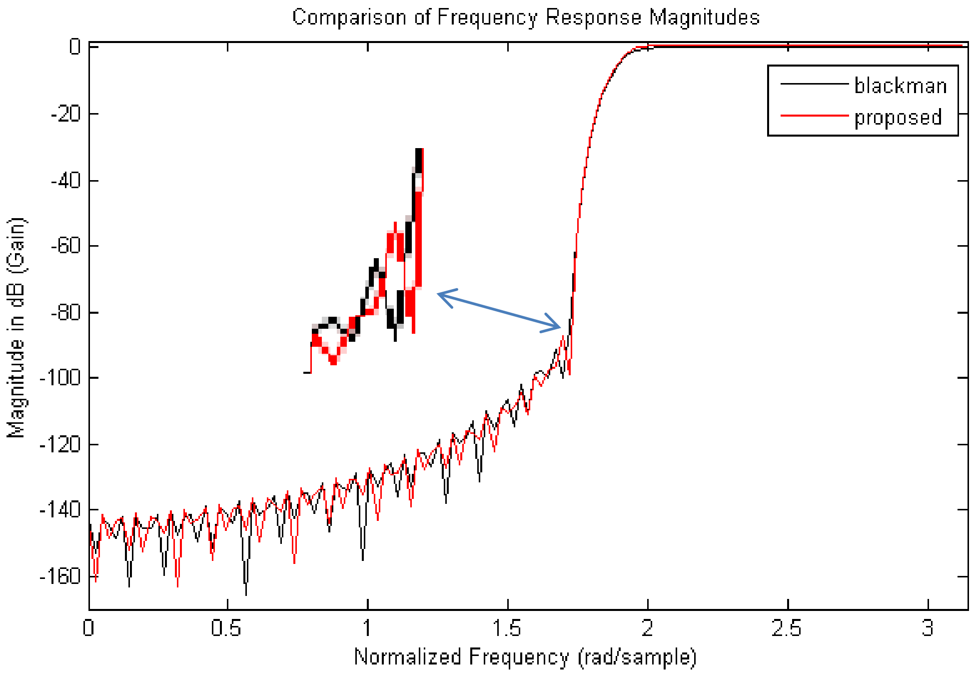

Figure 4.7 Spectrum Comparison of Proposed window and Blackman window for high pass filter, N=150

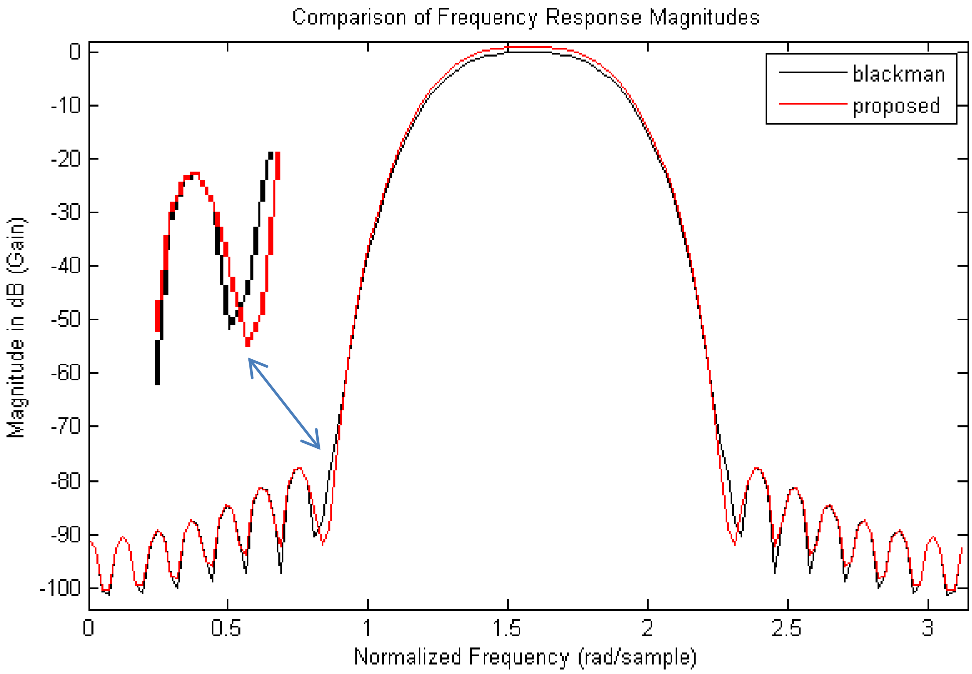

Figure 4.8 Spectrum Comparison of Proposed window and Blackman window for bandpass filter, N=50

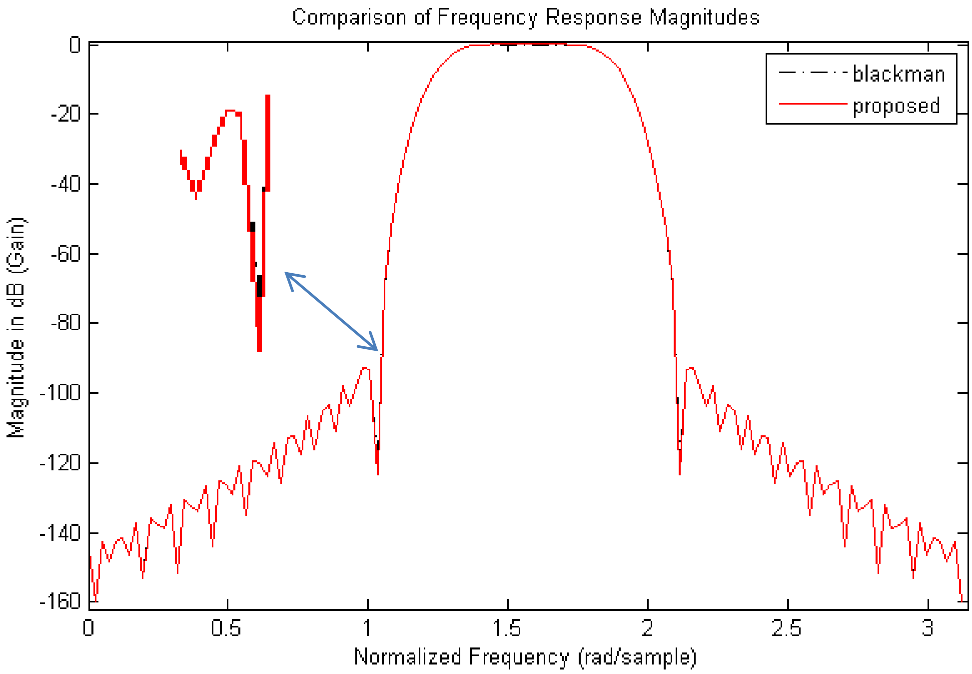

Figure 4.9 Spectrum Comparison of Proposed window and Blackman window for band pass filter, N=100

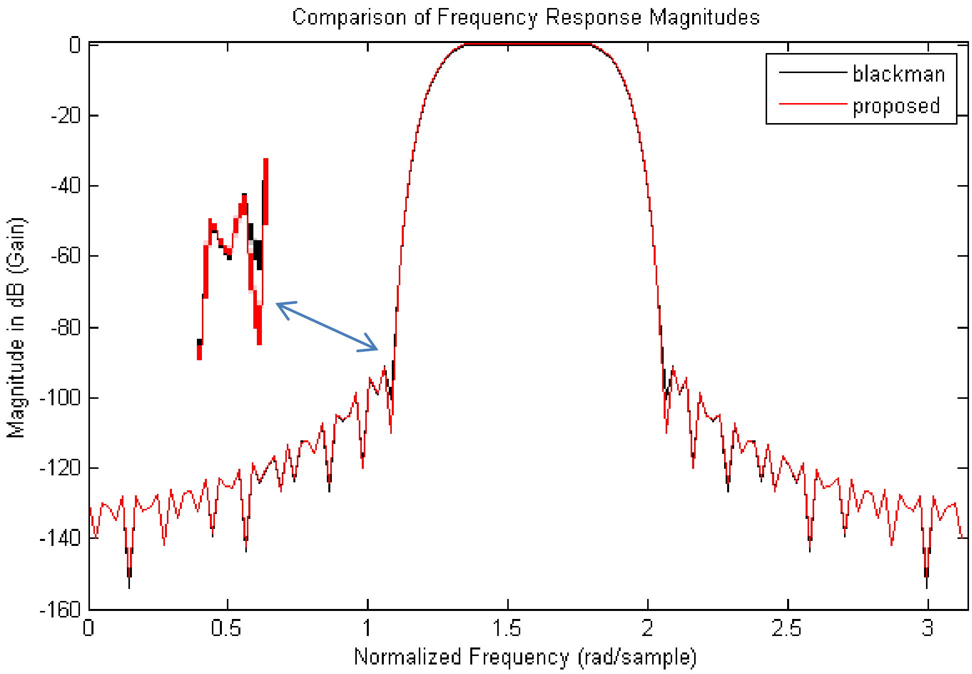

Figure 4.10 Spectrum Comparison of Proposed window and Blackman window for band pass filter, N=150

Last application of the proposed window is designed for band pass filter. The specification of the band pass filter is given as the ideal characteristics with cut- off frequencies $0.4 * \omega_s/2$ and $0.6 * \omega_s/2$ for left and right side of the filter respectively.

As in Figure 4.8, the proposed window has better result in terms of sharper transient and higher the first minimum of the gain in stopband attenuation than the Blackman window. Clearly, first minimum of the gain of the proposed window appears at the frequency $w_s/256$ much lower for right side and higher for left side than the Blackman window for 50^{th} filter order. The independent parameters of window are found as $\alpha = 1.3,\ \beta = 1.5$ and $\gamma = 0.1$.

Unlike the 50^{th} filter order, although cut-off frequencies are almost same for both Blackman and proposed window, the magnitude of the attenuations at the cut-off frequencies are better for later case as in Figure 4.9 and Figure 4.10. Proposed window parameters are calculated like $\alpha = 0.9,\ \beta = 0.5$ and $\gamma = 0.1$ for 100^{th} filter order and $\alpha = 0.6,\ \beta = 1.5$ and $\gamma = 0.1$ for the filter length $N = 150$.

Mainly, changing the type of filter (lowpass filter, highpass filter and bandpass filter) and changing the window length (N=50, N=100 and N=150) does not affect success of the proposed window spectral characteristic (sharper transient and stopband attenuation).

4.6 Main Program and Flow Chart

```
clc, clear
%% Loading different types of filters
LPF = Space_Low_Pass(); % Low Pass Filter Space file
HPF = Space_High_Pass(); % High Pass Filter Space file
BPF = Space_Band_Pass(); % Band Pass Filter Space file

space = BPF; %% Choose one of the filter that has been loaded

%%% Matlab filter data result is calculating for comparing proposed window
[matlab_transfer] = fir2(space.N, space.frequence , space.magnitude, blackman(space.N+1));
[matlab_function,w] = freqz(matlab_transfer, 1, space.bin_size);
matlab_abs = abs(matlab_function); % Matlab built in function filter result
matlab_dbscale = db_scale (matlab_abs);
%%%
error_index = 1; % For storing all applied alpha, beta, gamma values
for alpha = 0.1:0.1:3
  for beta = 0.1:0.1:3
     for gamma = 0.1:0.1:3
     my_out = myFilter (alpha, beta, gamma, matlab_transfer, matlab_function, space);
     temp_db= db_scale (my_out);
```

```
                  % Storing alpha beta gamma and error value in an array
        error (error_index, 1) = alpha;
        error (error_index, 2) = beta;
        error (error_index, 3) = gamma;
        error (error_index, 4) = space.Error_Func(temp_db, space.bin_size);

        if (error(error_index,4) < 0 )
           export_graph_png %% Only negative value error files plotted
        end

        error_index = error_index + 1;
        %disp (error_index) % Printing step number for showing working progress
        end
    end
end

[minerr, indx] = min ( error(:,4) ); % selecting minimum error configuration
disp ( error (indx, :) );
plot_db % Plots db function for our minimum error result and Matlab built in functions
```

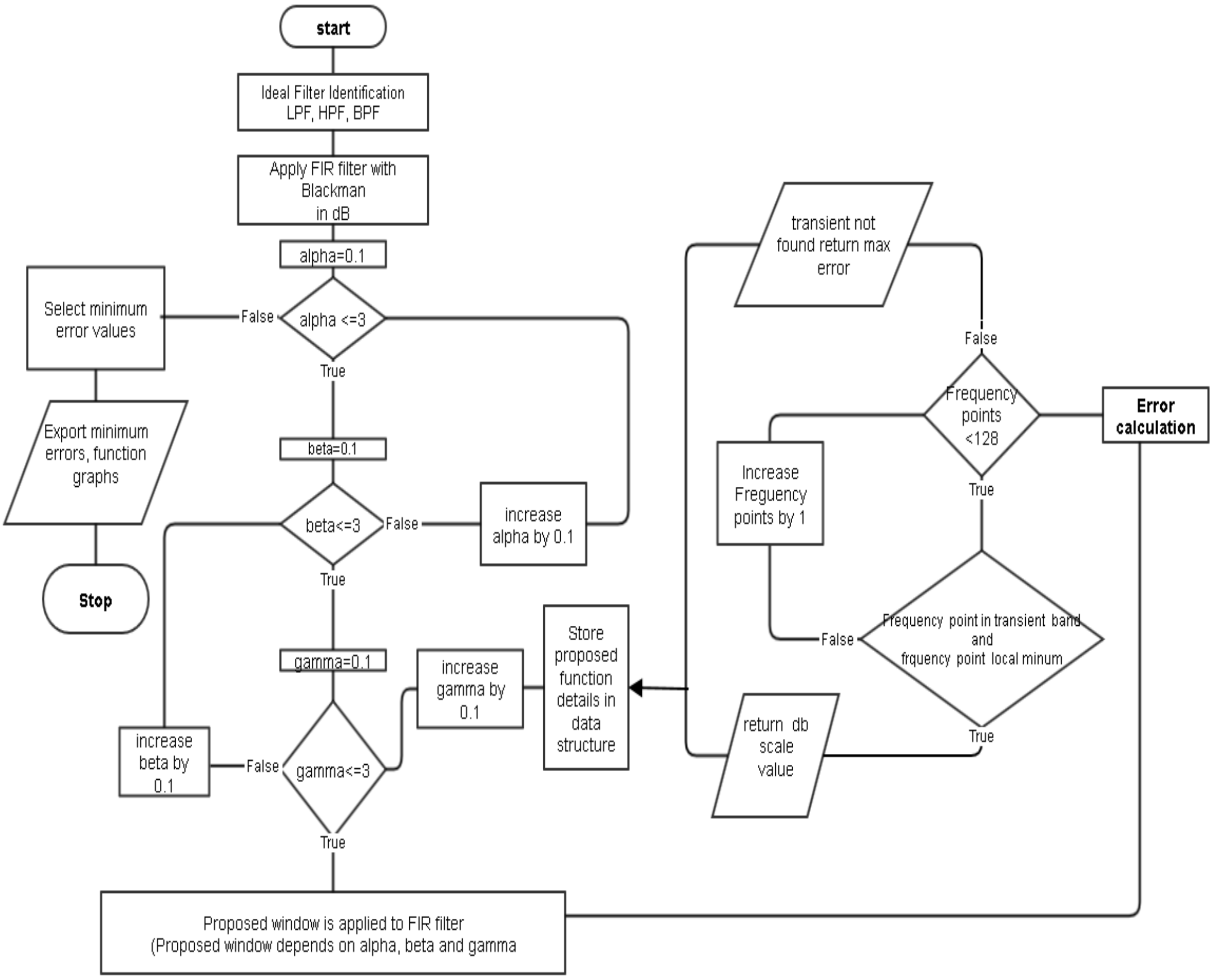

Figure 4.11 Flow chart of the program

CHAPTER 5

CONCLUSIONS AND FUTURE PLAN

A new and efficient window is proposed and applied for design of nonrecursive digital filters. Direct truncation of Fourier Series causes Gibbs' Oscillations. In order to reduce these oscillation Fourier Series with the application of new window function is studied in this thesis.

Window function has spectral characteristics which are ripple ratio, sidelobe roll-off ratio and mainlobe width. To get better spectral characteristics, window function should smaller ripple ratio, large sidelobe roll-off ratio and narrow mainlobe width. Smaller ripple ratio causes smaller ripples in the passband and stopband region. Narrower mainlobe width causes smaller transition width between passband and stopband regions in the filter.

Proposed window is derived version of cosine hyperbolic window function (Cosh window) and Blackman window. It has four independent parameters which are shown as α, β, γ and window length. Changing these parameters, window function characteristics change. Proposed window is compared with Blackman window in terms of transient sharpness and the first minimum gain for stopband attenuations. Blackman window has the minimum gain (maximum attenuation) in the stopband region also it results wider transient. Compared to other fixed windows, it has higher sidelobe attenuation. Blackman window is used for Fast Fourier Transform, in some signal processing and measurement. Because for this type of applications, higher sidelobe attenuation is required.

Proposed window function is analyzed for different filter types such as low pass, high pass and band pass filters and different window length like $\mathrm{N} = 50$, $\mathrm{N} = 100$ and $\mathrm{N} = 150$. Proposed window function is successful for different window length in terms of sharper transient and higher the first gain value. Different window lengths are applied and it has been observed that the results are also good in terms of sharper transient at the cut-off frequency. The important disadvantage of the window is that it has same transient for bandpass filter for 100^{th} and 150^{th} filter order but it gives good result in the stopband attenuations.

In the future, changing some parameter or adding new parameter, a new window function can be found for some applications. Saramaki window maybe applied window to change more sharper transient between passband and stopband region. When this feature is obtained, it will be used for new nonrecursive filter designs.

REFERENCES

[1] Kaya, Turgay and İnce, M. C. (2010). Pencere fonksiyon aileleri ve uygulama alanları. *Erciyes University Journal of the Institute and Science and Technology*. **26** (3): 291-306, September.

[2] Nouri, M., Ghaemmaghami, S. S. and Falahati, A. (2011). An improved window based on cosine hyperbolic function. *Multidisciplinary Journals in Science and Technology, Journal of Selected Areas in Telecommunications (JSAT)*. July Edition

[3] Antoniou, A. (2006). *Digital signal processing: Signals, systems and filters*. McGraw- Hill, New York, USA.

[4] Taşdemir, D. Ç and Nacaroğlu, A. (2012). A New Window Function for FIR Filter Design. *7. Proc. International Coneference on Electrical and Electronics Engineering (ELECO 2012)*. Bursa, Turkey. 267- 270.

[5] Saramaki, T. (1991). Adjustable windows for the design of FIR filters-a tutorial. *IEEE Electrotechnical Conference, 1991. Proceedings., 6th Mediterranean*. **1**, May, 28-33.

[6] Kaplan, H. and Nacaroğlu A. (2015). FIR filtre tasarımı için yeni bir pencere fonksiyonu. *23. Sinyal İşleme ve İletişim Uygulamaları Kurultayı (SIU 2015),* Malatya, Turkey, 2353-2357, 16-19 May 2015.

[7] Avci, K. and Nacaroğlu, A. (2008). Ph.D. Thesis, Design of High-Quality Low-Order Nonrecursive Digital Filters Using the Window Functions. *Department of Electrical and Electronics Engineering, University of Gaziantep*. Gaziantep. Turkey.

[8] Bergen, S.W.A. (2005). Design of the Ultraspherical window function and its applications. PhD Dissertation. University of Victoria. Department of Electrical and Computer Engineering. Victoria. Canada

[9] Dolph, C. L. (1946). A current distribution for broadside arrays which optimizes the relationship beetwen beamwidth and side-lobe level. *Proc. IRE*, June, **34**, 335-348.

[10] Avci, K. (2014). Kaiser-Hamming penceresi ve yinelemesiz sayısal süzgeç tasarımı için performans analizi. *Journal of the Faculty of Engineering and Architecture of Gazi University,* **29**. 823-833.

[11] Bergen, S.W.A. and Antoniou, A. (2005). Design of nonrecursive digital filters using the Ultraspherical window function. *EURASIP Journal on Applied Signal Processing,* **12**, 1910-1922.

[12] https://www-sigproc.eng.cam.ac.uk/Main/OP205

[13] El-Ali, T. S. (2004). *Discrete systems and digital signal processing with MATLAB*. Columbia, South Carolina. CRC Press.

[14] Taşdemir, Ç. D. and Nacaroğlu, A. (2013). M.sc Thesis, A new window function for FIR filter design. *Department of Electrical and Electronics Engineering, University of Gaziantep*. Gaziantep. Turkey.

[15] Shayesteh, M. G. and Kashtiban, M. M. (2009). FIR filter design using a new window function. IEEE Digital Signal Processing 16. International Conference, August, Santorini-Hellas, 1-6.

[16] Roychowdhury, A. (2002). FIR filter design techniques. *Seminar Report.Electronic Systems Group*. EE Dept. IIT Bombay. India.

[17] Proakis, J. G. and Manolakis, D. G. (1996). *Digital signal processing: Principles, algorithms and applications*. Prentice-Hall. Inc. USA.

[18] Avci, K. and Nacaroğlu, A. (2008). A new window based on exponential window. *IEEE Ph.D. Research in Microelectronics and Electronics (PRIME 2008)*. June. Istanbul, Turkey, 69-72.

[19] Peng, S. (2013). Bachelor Thesis, Design and analysis of FIR filters based on matlab. *School of Computer Science, Physics and Mathematics, Electrical Engineering,* Linnæus University. Kalmar, Sweden.

[20] Antoniou, A. (2005). *Digital filters: Analysis, design and applications*. Second Edition. McGraw-Hill, New York, NY, USA.

[21] Ingle, K. V. and Proakis (2010). *Digital Signal Processing with Matlab*. Cengage Learning, Third Edition, USA.

[22] Chakraborty, S. (2013). Advantages of Blackman window over Hamming window method for designing FIR filter. *International Journal of Computer Science and Engineering Technology,* **4**, 2229-3345.

[23] Jain, S and Verma. D. K. (2013). Comparison of various window functions used in FIR filter designing. *International Journal of Computers & Technology*. 8, 902-907.

[24] Avci, K. and Nacaroğlu, A. (2008). Modification of Cosh window family. Proc. of Third International Conference on Information and Communication Technologies (ICTTA'08). April. Damascus, Syria, 291-292.

[25] Saramaki, T. (1989). A class of window functions with nearly minimum sidelobe energy for designing FIR filters. *Proc. IEEE Int. Symp. Circuits and systems (ISCAS'89*), Portland, Ore, USA, May**, 1**, 359-362

[26] Schwarzweller, A. R. and Vintermental. M. (2002). On designing FIR filters using window based on Gegenbauer polynomials. *IEEE International Symposium on Circuits and Systems (ISCAS 2002)*. **1**. 413-416.

[27] Avci, K. and Nacaroğlu, A. (2008). Cosh window family and its application to FIR filter design. Accepted for the publication in *International Journal of Electronics and Communications-AEU*

[28] Sharma, S.N., Saxena R., Saxena, S.C. (2004). Design of FIR filter using variable window families: A comparative study. *J. Indian Inst. Sci.*, Sept.-Oct., **84**, 155-161.

[29] Rajput, S. S. and Bhadauria, S. S. (2012). Implementation of FIR filter using efficient window function and its application in filtering a speech signal. *International Journals of Electrical, Electronics and Mechanical Controls*. **1/1**, November

[30] Fuchs, J.J. (2002). New windows for tunable length FIR Filter Design. *Proc. IEEE ICASSP'02.* May, Orlando, USA, **2**, 1521-1524.

APPENDIX

The developed Matlab programs are shown for proposed window function expressed as below. Independent parameters of proposed window are calculated by Matlab software. Frequency response magnitudes for proposed window function are drawn. All Matlab programs are indicated as M-files. There are eleven M- files.

A.1 Main

```
%%% Main Program file for exacuting Filter test %%%

clc, clear
%% Loading different types of filters
LPF = Space_Low_Pass(); % Low Pass Filter Space file
HPF = Space_High_Pass(); % High Pass Filter Space file
BPF = Space_Band_Pass(); % Band Pass Filter Space file

space = BPF; %% Choose one of the filter that has been loaded

%%% Matlab filter data result is calculating for comparing proposed window
[matlab_transfer] = fir2(space.N, space.frequence , space.magnitude,
blackman(space.N+1));
[matlab_function,w] = freqz(matlab_transfer, 1, space.bin_size);
matlab_abs = abs(matlab_function); % Matlab built in function filter result
matlab_dbscale = db_scale (matlab_abs);
%%%

error_index = 1; % For storing all applied alpha, beta, gamma values
for alpha = 0.1:0.1:3
   for beta = 0.1:0.1:3
      for gamma = 0.1:0.1:3
      my_out = myFilter (alpha, beta, gamma, matlab_transfer, matlab_function, space);
      temp_db= db_scale (my_out);
                  % Storing alpha beta gamma and error value in an array
      error (error_index, 1) = alpha;
      error (error_index, 2) = beta;
      error (error_index, 3) = gamma;
      error (error_index, 4) = space.Error_Func(temp_db, space.bin_size);

      if (error(error_index,4) < 0 )
         export_graph_png %% Only negative value error files plotted
      end
```

```
        error_index = error_index + 1;
        %disp (error_index) % Printing step number for showing working progress
        end
    end
end

[minerr, indx] = min ( error(:,4) ); % selecting minimum error configuration
disp ( error (indx, :) );
plot_db % Plots db function for our minimum error result and Matlab built in functions
```

A.2 Low Pass Filter

```
% Low Pass Settings
function ret = Space_Low_Pass ()
space.frequence = [0 0.1 0.2 0.3 0.4 0.4 0.5 0.6 0.7 0.8 0.9 1.0];
space.magnitude = [1 1 1 1 1 0 0 0 0 0 0 0];
space.bin_size = 128;
space.N = 100;
space.Error_Func=@Calc_Error_LPF
for i = 1:space.bin_size
    if (i <= 52)
        space.ideal_filter (i) = 1;
    else
        space.ideal_filter (i) = 0;
    end
end
ret = space;
```

A.3 High Pass Filter

```
function ret = Space_High_Pass ()
space.frequence = [0 0.1 0.2 0.3 0.4 0.5 0.6 0.6 0.7 0.8 0.9 1.0];
space.magnitude = [0 0 0 0 0 0 0 1 1 1 1 1];
space.bin_size = 128;
space.N = 150;
space.Error_Func=@Calc_Error_HPF
for i = 1:space.bin_size
    if (i <= 76)
        space.ideal_filter (i) = 0;
    else
        space.ideal_filter (i) = 1;
    end
end
ret = space;
```

A.4 Band Pass Filter

```
function ret = Space_Band_Pass ()
space.frequence = [0 0.1 0.2 0.3 0.4 0.4 0.5 0.6 0.6 0.7 0.8 0.9 1.0];
space.magnitude = [0 0 0 0 0 1 1 1 0 0 0 0 0];
space.bin_size = 128;
space.N = 150;
space.Error_Func=@Calc_Error_BPF
for i = 1:space.bin_size
   if (i <= 51)
      space.ideal_filter (i) = 0;
   elseif (i <= 76)
      space.ideal_filter (i) = 1;
   else
      space.ideal_filter (i) = 0;
   end
end
ret = space;
```

A.5 Error Calculation for LPF

```
function error = Calc_Error_LPF (test, N)
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        y_limit = -50;
        x_limit = 70;
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %% Scanning from 1 to N by increasing 1
        prev=test(1);
   i=2;
   while ( test(i) < prev+1 ) && i < N
     prev = test(i);
     i = i+1;
   end
        i = i-1;
   if (i < x_limit) && test(i) < y_limit
      error = test(i);
            disp(i)
            disp(test(i))
   else
      error = 1;
   end
```

A.6 Error Calculation for HPF

```
function error = Calc_Error_HPF (test, N)
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        y_limit = -50;
```

```
        x_limit = 70;
        %%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
        %% Scanning from N to 1 by decreasing 1
        prev=test(N);
   i=N-1;
   if (abs(test(N)) > 1)
      error = 1;
      return
   end
   while i >0 && ( test(i) < prev+1 )
     prev = test(i);
     i = i-1;
   end
        i=i+1;
   if (i > x_limit) && test(i) < y_limit
     error = test(i);
     disp(i)
     disp(test(i))
   else
      error = 1;
   end
```

A.7 Error Calculation for BPF

```
function error = Calc_Error_BPF (test, N)

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
   mid = N/2; %% Selecting mid point which is on for BPF
   left_x_limit = 44;
   left_y_limit = -101;
   right_x_limit = 86;
   right_y_limit = -101;

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

        %% Scanning to the Left from mid point
        prev=test(mid);
   i=mid-1;
   if (abs(test(mid)) > 1) %% start point must be near to 0 with "1" unit range
      error = 1;
      return
   end
   while i > 0 && ( test(i) < prev+1 ) %% current value must be smaller than prev with
"1" unit range
     prev = test(i);
     i = i-1;
   end
        i=i+1;
   if (i > left_x_limit) && test(i) < left_y_limit
```

```
        error = test(i);
    else
        error = 1;
        return % Evaluating from function
    end

    %% Scanning to the right from mid point
    prev=test(mid);
    i=mid+1;
    while ( test(i) < prev+1 ) && i < N %% current value must be smaller than prev with
"1" unit range
       prev = test(i);
       i = i+1;
    end
          i=i-1;
    if (i < right_x_limit) && test(i) < right_y_limit
        error = error + test(i);
    else
        error = 1;
    end
```

A.8 Logarithmic dB Scale

```
function rtrn = db_scale (x)
    rtrn = 20 * log10 (x);
```

A.9 dB Scale Plot

```
%% Ploting the final result for proposed and blackman filter
hold all
my_out = myFilter ( error(indx,1), error(indx,2), error(indx,3), matlab_transfer,
matlab_function, space );
my_dbscale = db_scale (my_out);

plot ( w, matlab_dbscale, 'b' );
plot ( w, my_dbscale, 'g' );
```

A.10 Export as Picture

```
% To export Graph As PNG file
hold on
plot (w, temp_db, 'r')
plot (w, matlab_dbscale, 'k')
xlabel('Normalized Frequency ( rad/sample)')
ylabel('Magnitude in dB (Gain)')
title('Comparison of Frequency Response Magnitudes')
legend('proposed','blackman')
filename =
sprintf('out/out_%.1f_%.1f_%.1f_%.3f.png',alpha,beta,gamma,error(error_index,4));
```

```
print ('-dpng',fullfile(filename) )
close all
```

A.11 Applied Filter

```
function rvalue = myFilter (alpha,beta,gamma,m_transfer, m_function, space)
   for i = 1:space.N+1
        my_Coshwindow (i) = ( ( cosh( gamma * (1 - ( 2*i / (space.N-1) )^2 )^(beta) ) ) / 
cosh(gamma) )^(alpha);
        my_windowed   (i) = m_transfer(i) * my_Coshwindow(i);
   end

   [my_function,w] = freqz( my_windowed, 1, space.bin_size );
   my_fcorrected   = ( max(abs( m_function(1) ) ) / max (abs( my_function(1) ) )) * 
my_function;
   rvalue = abs( my_fcorrected );
```

PUBLICATION:

A version of this thesis was published Signal Processing and Communications Applications Conference (SIU), 2015 23rd. Web link of published paper:

http://ieeexplore.ieee.org/Xplore/home.jsp