Volume 3, Issue 10, October 2013 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Design and Implementation of Maximally Flat Fir Fi Lter Using Multrirate Technique on Recording Signal for Suppresed of Power Line Interfernce

Amandeep
M. Tech, (ECE)
P.I.E.T, Smalkha (K.u.k) India

Dharmbir Singh
ECE & Applied Science
P.I.E.T, Smalkha (K.u.k) India

Abstract: In this paper we introduce the design of Maximally Flat filter using a Multirate technique which helps to reduce the filter order of MF FIR filter. To eliminate or compress the power line interfacing of recording signal in Biomedical, control, Instrumentation and Communication Engineering .MF FIR filter are design with MATLAB programming.

Key words: Digital filters, FIR filters, Multirate Filter Design, Matlab

I. Introduction

Filter is design to attenuate the power line noise. Filtering of power line interference is very meaningful in the measurement of biomedical event, particular in the case of recording signal as well as the ECG. We use digital filter to attenuate the noise signal from the input signal. Digital filter are two types IIR filter and FIR filter .we discussed about FIR filter which have a stability and linear phase response characteristics. [1] FIR notch filter are design with some technique such as Fourier series method, Frequency sampling method, Window technique, Kaiser Window, Optimal linear phase FIR filter, Multirate Filter and Maximally Flat FIR filter. Kaiser window of FIR filter are design to eliminate the noise of ECG signal. [2] Blackman window function is used to design an FIR filter for efficient value of α, this window function provide higher side lobe attenuation comparisons to hamming and hanning window and main lobe width of this window function is slightly greater than hamming window. [3] Optimization technique are used reduce the error which cause by frequency sampling technique at the non sampled frequency point. [4] Prefiltering are used in FIR filter to decrease the number of multiplier. Mirror image quadratic polynomial and Cyclotomic polynomial is used to give sufficient desire attenuation in stop band. [5] In FIR filter make transfer function of HTs, DDs and FDs by expanding some suitable function in power series .fir HTs based on expanding the signum function into power series MFs FIR DDs are designed by expanding some inverse triangular function into power series. FIR FDs are designed by expanding the ideal transfer function into Binomial series. [6] Maximally Flat low pass FIR filter design using Baher's filter are particular useful because they yield tradeoff between the linearity of phase response and the width of transmission band for frequency selective system. [7] FIR filter can be realized either recursively or non recursively .Recursively realization of frequency sampling design are simple to program and can be very efficient .Non recursive realization include direct convolution and fast convolution. [8] Algorithm independent lower bound on the achievable approximation error and then present an Approximation method involve the solution of a fixed number of all pass extension problem so is called the Nehari shuffle [9] If we use differentiate of MF (Maximally flat) and ER (euiripple filter) do not require any FFT algorithm or any iterative technique. [10] Nonsymmetrical half band Maximally Flat Filter and used the Bernstein polynomial to derive a closed form formula for the transfer function of Group Delay Maximally Flat FIR filter [11] Decasteljau algorithm was used to obtain multiplier less Maximally Flat FIR filter. The structure is based on a recursive evaluation technique for Bernstein polynomial. [12] We use the simplex algorithm for linear programming to find the linear phase filter of minimum length which limit on the frequency response and maximize the distance from the constraints. [13] Design of Analog and Digital filter using Matlab special importance is the pole placer function that is used to synthesis filter with ER and MF pass band and arbitrary piece wire constant stop band [14] Fir filter are also design either by software VLSI technique or minimum hardware solution to improve the linear phase response [15] This paper is organized as follows. In Section 2, we propose a design of the FIR based on maximally FIR filter. Maximally-flat filters are filters in which both the passband and stopband are as flat as possible given a specific filter order. As we should come to expect, the flatness in the bands comes at the expense of a large transition band in section 2.1 FIR filters is that tend to require a large filter order and therefore a high computational cost to achieve the specifications desired. To decrease the filter order we design the Multirate Filter Design .in section 3 we purpose Multirate Filter Design which reduce filter order with Reducing the sampling rate of a signal In Section 4, design examples and coefficient quantization effects are provided. a final conclusion discussed in Section5.

II. Maximally-flat FIR filters:

Maximally-flat filters are filters in which both the passband and stopband are as flat as possible given a specific filter order. As we should come to expect, the flatness in the bands comes at the expense of a large transition band (which will also be maximum). There is one less degree of freedom with these filters than with those we have look at so far. The only way to decrease the transition band is to increase the filter order

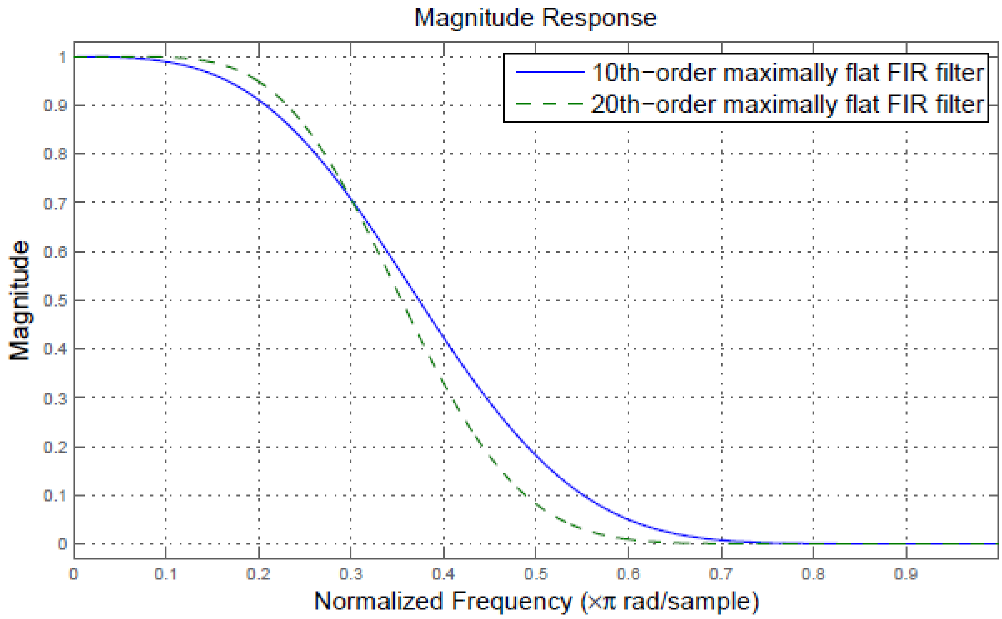


Fig: (2.a) shows two maximally-flat FIR filters. Both filters have a cutoff frequency of 0.3π. The filter with smaller transition width has twice the filter order as the other.

The Maximally-flat stopband of the filter means that its stopband attenuation is very large. However, this comes at the price of a very large transition width. These filters are seldom used in practice (in particular with fixed-point implementations) because when the filter's coefficients are quantized to a finite number of bits, the large stopband cannot be achieved (and often is not required anyway) but the large transition band is still a liability. Compare the stopband attenuation of a maximally-flat FIR filter implemented using double-precision floating-point arithmetic with that of the same filter implemented with 16-bit fixed-point coefficients. The comparison of the two implementations is shown in Figure (2.b). The fixed-point implementation starts to differ significantly from the floating-point implementation at about 75-80 dB. Nevertheless, both filters have the same large transition width

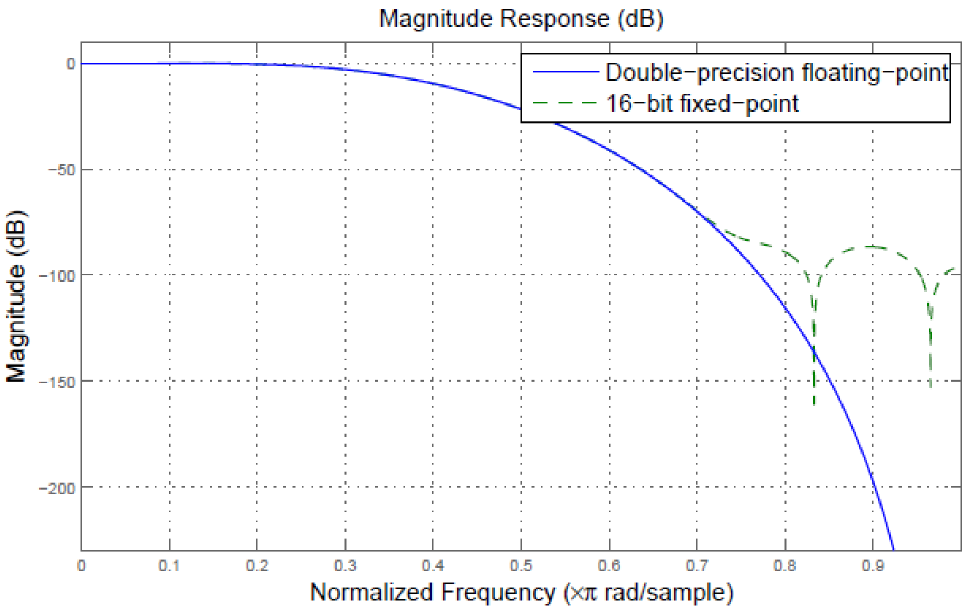


Fig: (2.b) a maximally-flat FIR filter implemented with double-precision floating point arithmetic and the same filter implemented with 16-bit fixed-point coefficients.

The maximally-flat passband may be desirable because it causes minimal distortion of the signal to be filtered in the frequency band that we wish to conserve after filtering. So it may seem that a maximally-flat passband and a euiripple or perhaps sloped stopband could be a thought after combination. However, if a small amount of ripple is allowed in the passband it is always possible to get a smaller transition band and most applications can sustain a small amount of passband ripple.

Example 2.1 we can approach a maximally-flat passband by making the passband ripple of an euiripple design progressively smaller. However, for the same filter order and stopband attenuation, the transition width increases as a result.

Consider the following two filter designs:

Matlab programmed

```
Hf = fdesign.lowpass ('N, Fc, Ap, Ast', 70, .3, 1e-3, 80);
Heq = design (Hf,'equiripple');
Hf2 = fdesign.lowpass ('N, Fc, Ap, Ast', 70, .3, 1e-8, 80);
Heq2 = design (Hf2,'equiripple');
```

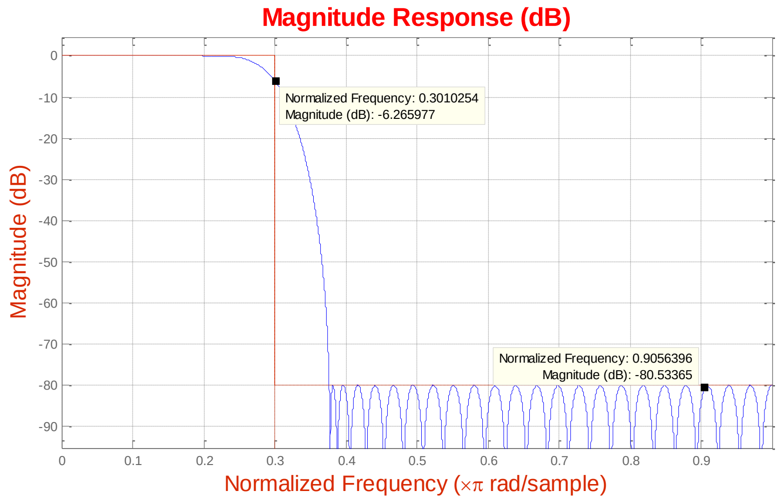


Fig: (2.c) Equiripple filters with passband approximating maximal flatness. The better the approximation , the larger the transition band.

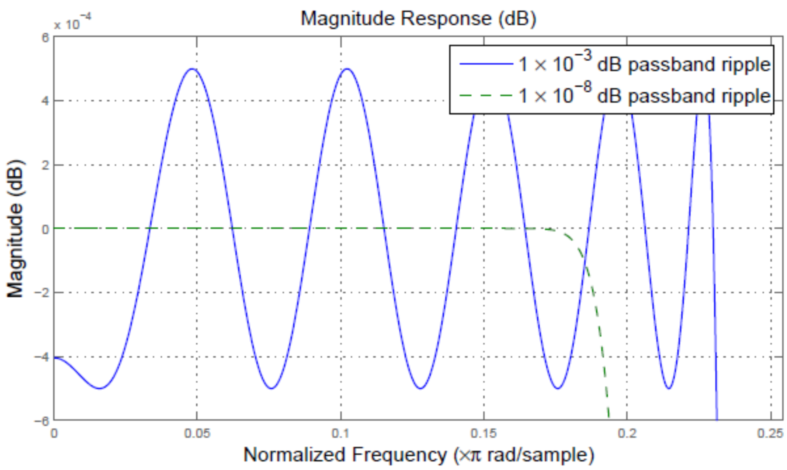


Figure (2.d): The two filters are shown in Figure (c). It is generally best to allow some passband ripple as long as the application at hand supports it given that a smaller transition band results. The passband details are shown in Figure (d)

2.1Disadvantages Maximally FIR filter:

Understanding FIR filter design is a matter of understanding the tradeoffs involved and the degrees of freedom available. A drawback of maximally FIR filters is that tend to require a large filter order and therefore a high computational cost to achieve the specifications desired. There are many ways of addressing this. One is to use IIR filters. Another is to use multistage and/or Multirate techniques that use various FIR filters connected in cascade (in series) in such a way that each filter shares part of the filtering duties while having reduced complexity when compared to a single-stage design. The idea is that for certain a specification to combined complexity of the filters used in multistage design is lower than the complexity of a comparable single-stage design. We will be looking at all these approaches in the following chapters. We will then look into implementation of filters and discuss issues that arise when implementing a filter using fixed-point arithmetic.

III. Multirate Filter Design

Multirate signal processing is a key tool to use in many signal processing implementations. The main reason to use Multirate signal processing is efficiency. Digital filters are a key part of Multirate signal processing as they are an enabling component for changing the sampling rate of a signal.. In this paper we will talk about designing filters for Multirate systems. Whether we are increasing or decreasing the sampling rate of a signal, a filter is usually required. The most common filter type used for Multirate applications is a low pass filter we start by presenting filters in the context of reducing the sampling rate of a signal (decimation). We want to emphasize the following: if you reduce the bandwidth of a signal through filtering, you should reduce its sampling rate accordingly. We will see that the above statement applies whether the bandwidth is decreased by an integer or a fractional factor. Understanding fractional Sampling-rate reduction requires understanding interpolation. We present first interpolation as simply finding samples lying between existing samples (not necessarily increasing the sampling rate). We then use this to show how to increase the sampling-rate of a signal by an integer factor. Finally we extend this paradigm in order to perform fractional decimation/ interpolation.

3.1. Reducing the sampling rate of a signal

In the previous paper we have discussed the basics of FIR and IIR filter design. We have concentrated mostly on lowpass filters. A lowpass filter reduces the bandwidth of the signal being filtered by some factor. When the bandwidth of a signal is reduced, we usually want to reduce the sampling rate in a commensurate manner. Otherwise, we are left with redundant data, given that we are over-satisfying the Nyquist sampling criterion. Any processing, storage, or transmission of such redundant data will result in unnecessary use of resources. Specifically, if the input signals x[n] has a bandwidth Bx and the filtered signal y[n] has a bandwidth By related to the input bandwidth by

$$\frac{B_x}{B_y} = M$$

We should reduce the sampling frequency of y[n] by a factor of M.

3.2. Decimating by an integer factor

If M is an integer the procedure is straightforward. We can downsample y[n] by keeping one of every M samples

y down[m] = y [nM].

The idea is depicted in Figure 4.1 and is referred to as decimation. The filter H (z) is a lowpass filter that roughly reduces the bandwidth by a factor of M. Once the bandwidth is reduced, the sampling-rate should be reduced accordingly. The downsample that follows H (z) reduces the sampling rate by keeping one out of every M of its input samples. However, the procedure as described so far is inefficient in that after the filter processes a set of M samples, only one sample kept. All the computation involved in obtaining the other M− 1 samples is wasted.

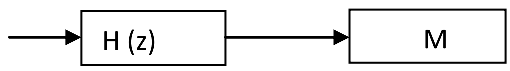


Figure 3.a: Reducing the sampling rate after the bandwidth of the signal is reduce

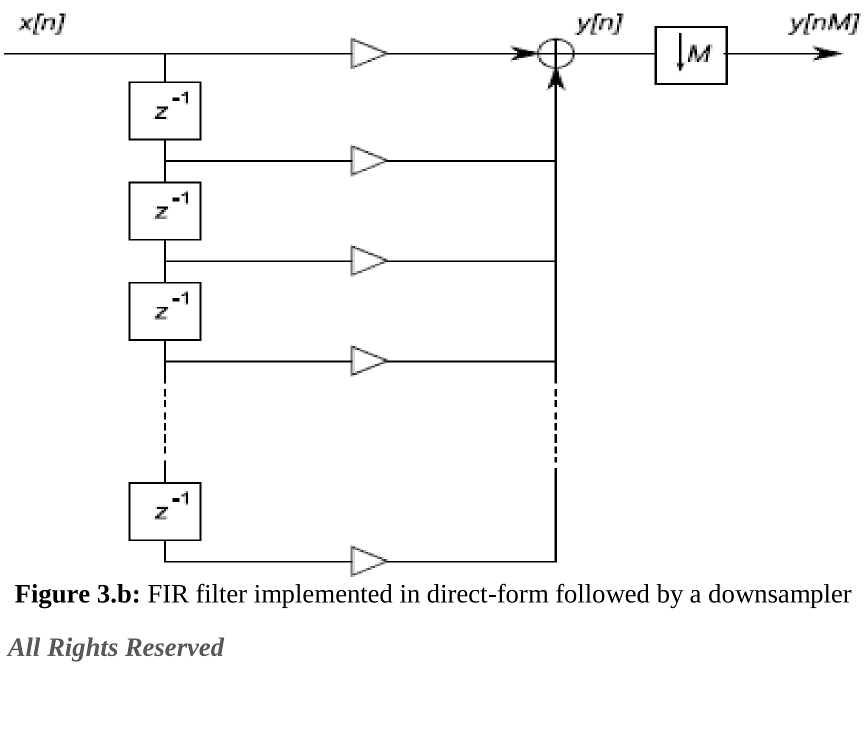


Figure 3.b: FIR filter implemented in direct-form followed by a downsampler

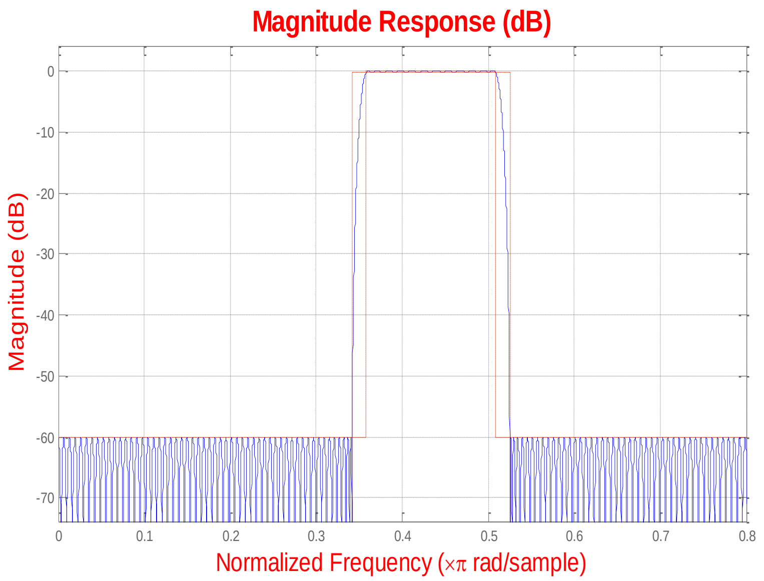
IV. Simulation and Results:-

Example 1 A 3th-band Nyquist filter reduces the bandwidth by a factor of 3. We can simultaneously reduce the sampling rate by a factor of 3 by designing a decimator Nyquist filter,

```
f = fdesign.decimator(3,'Nyquist',3,'TW,Ast',0.1, 50);
h = design(f,'kaiserwin');M = 16; % Decimation factor
Fc = 0.0625;
TW = 0.008;
Fp = Fc-TW/2;
Fst = Fc+TW/2;Ap = 1;
Ast = 80;
Hf = fdesign.decimator (M,'lowpass','Fp,Fst,Ap,Ast',...
Fp,Fst,Ap,Ast);
Hd = design(Hf,'equiripple');
cost(Hd)
result:
Number of Multipliers          : 642
Number of Adders               : 641
Number of States               : 640
MultPerInputSample             : 40.125
AddPerInputSample              : 40.0625
```

Example 2 Suppose we are interested in retaining the band between 0.25p and 0.4565p. The bandwidth is reduced by a factor M = 6. However, the band-edges are not between kp/M and (k + 1)p/M for M = 4,5,6. The band-edges do lie between p/3 and 2p/3, so if we design a bandpass filter we can at least decimate by 3

```
TW = 0.1/6; % Transition width
Fc1 = 0.25;
Fc2 = 0.4565;
M = 3; % Decimation factor
Hf=fdesign.decimator(M,'bandpass','Fst1,Fp1,Fp2,Fst2,Ast1,Ap,Ast2',...
Fc1-TW/2,Fc1+TW/2,Fc2-TW/2,Fc2+TW/2,60,.2,60);
Hd = design(Hf,'equiripple');
cost(Hd)
Result =
Number of Multipliers          : 312
Number of Adders               : 311
Number of States               : 309
MultPerInputSample             : 104
AddPerInputSample              : 103.6667
```



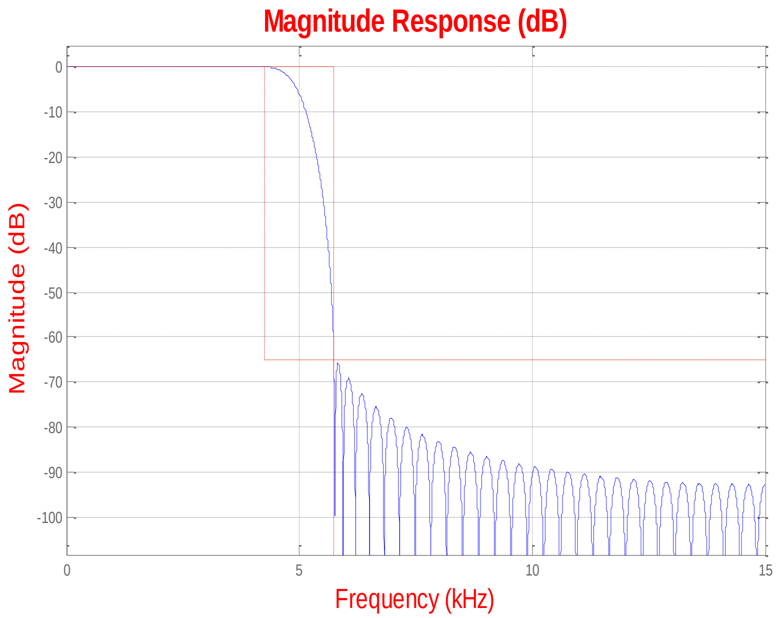
Example 3 Suppose we have a signal sampled at 30 kHz and we reduce its bandwidth by a factor of 3. The band of interest extends from 0 Hz to 4250 Hz. In order to remove redundancy, we reduce the sampling-rate by a factor of 3 as

well. We use a 3rd-band Nyquist filter, therefore the cutoff frequency is equal to Fs/(2M) = 5000 Hz. The transition band is set to 1500 Hz so that it extends from 4250 Hz 57Hz.

```
M = 2;
Band = 2;
Fs = 30e3;
Hf = fdesign.decimator (M,'Nyquist', Band,'TW, Ast', 1500, 65,
Fs);
Hd = design (Hf,'kaiserwin');

Ans =

Number of Multipliers          : 41
Number of Adders               : 40
Number of States               : 80
MultPerInputSample             : 20.5
AddPerInputSample              : 20
```



V. Conclution

We have presented a fast and robust procedure for the design of Maximally FIR filters. The proposed procedure is suitable for reduce the filter order and complexity of filter real time notch filtering and therefore a high computational cost to achieve the specifications desired. There are many ways of addressing this. multistage and/or Multirate techniques that use various FIR filters connected in cascade (in series) in such a way that each filter shares part of the filtering duties while having reduced complexity when compared to a single-stage design. The idea is that for certain specifications to combined complexity of the filters used in multistage design are lower than the complexity of a comparable single-stage design. We will be looking at all these approaches in the following paper. We will then look into implementation of filters and discuss issues that arise when implementing a filter using fixed-point arithmetic

Reference

1. Miroslav Vlˇcek and Ladislav Jireˇs, Czech Technical University, Faculty of Electrical Engineering Technique 2, 166 27 Praha 6 , Czech Republic , vlcek@fd.cvut.cz
2. Bhumika Chandrakar, International Journal of Advanced Research in Computer and Communication Engineering Vol. 2, Issue 3, March 2013 ISSN (Print) : 2319-5940ISSN (Online) : 2278-1021
3. Dr.S.S Baduria MIT Gwalior Saurab Singh rajput International journal of Electrical ,Electronic and Mechanical control Vol .1 issue 2012
4. Arojit Roychowdhury, M.Tech. credit seminar report, Electronic Systems Group, EE Dept, IIT Bombay, submitted November2002
5. Hisakazu Kikuchi
6. Soo-Chang Pei, Fellow, IEEE, and Peng-Hua Wang, IEEE Transactions on circuit and system – Part 1: Fundamental theory and application , VOL.48,No 4,April 2001
7. Saed Samadi, Member, Transactions on signal processing, VOL. 48, No. 7, July 2000
8. Lawrence R.Rabiner ,Member IEEE Transaction on Audio and electro acoustics VOL.No.3 , September 1971
9. Peter J. Kootsookos, Robert R.Bitmead, Fellow, IEEE, and Michael Green, IEEE Transactions on signal processing. VOL.40.NO.8AUGUST 1992.

10. Miroslav Vlˇcek and Pavel Zahradn´ık, IEEE Transactions on and system -1 regular paper , AND SYSTEMS-I: REGULAR PAPERS, VOL. 51, and No. 3, march 2004
11. Saed Samadi ,Akinori Nishara .Tokyo institute of technology ,Tokyo Japan
12. S. Samadi ,T. Cooklev ,A. Nishihara , Department of Physical Electronics ,Tokyo Institute of Technology,2-12-1 Ookayama ,Meguro –Ku Tokya0 152 , Japan
13. T. W. Parks Electrical Engineering Cornell University ,Ithaca ,NY 14850
14. Per Löwenborg, Oscar Gustafsson, and Lars Wanhammar, Department of Electrical Engineering, Linköping University, SE-581 83 Linköping, Sweden
15. Gopichauhann, Department of ECE, NIT, Triuchirappali, IJECC, VOL 3. Issue(1) NCRTCT ,ISSN 2249-071X

