Design and VLSI Implementation of a Decimation filter for Hearing Aid Applications

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology

In

VLSI DESIGN and EMBEDDED SYSTEM

By

SAGARA PANDU

Roll No: 20507011

Department of Electronics & Communication Engineering

National Institute of Technology

Rourkela

2007

Design and VLSI Implementation of a Decimation filter for Hearing Aid Applications

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Master of Technology

In

VLSI DESIGN and EMBEDDED SYSTEM

By

SAGARA PANDU

Roll No: 20507011

Under the Guidance of

Prof.K.K.MAHAPATRA

Department of Electronics & Communication Engineering

National Institute of Technology

Rourkela

2007

NATIONAL INSTITUTE OF TECHNOLOGY
ROURKELA

CERTIFICATE

This is to certify that the Thesis Report entitled "**Design and VLSI Implementation of a Decimation filter for Hearing Aid Applications**" submitted by **Mr. Sagara Pandu (20507011)** in partial fulfillment of the requirements for the award of Master of Technology degree in **Electronics and Communication Engineering** with specialization "**VLSI Design and Embedded System**" during session 2006-2007 at National Institute of Technology, Rourkela (Deemed University) and is an authentic work by him under my supervision and guidance.

To the best of my knowledge, the matter embodied in the thesis has not been submitted to any other university/institute for the award of any Degree of Diploma.

Date:

Prof. K.K. Mahapatra
Dept. of Electronics & Communication Engg.
National Institute of Technology
Rourkela-769008

ACKNOWLEDGEMENTS

First of all, I would like to express my deep sense of respect and gratitude towards my advisor and guide **Prof. K. K. Mahapatra.** who has been the guiding force behind this work. I am greatly indebted to him for his constant encouragement, invaluable advice and for propelling me further in every aspect of my academic life. His presence and optimism have provided an invaluable influence on my career and outlook for the future. I consider it my good fortune to have got an opportunity to work with such a wonderful person.

Next, I want to express my respects to **Prof. G. S. Rath, Prof. G. Panda, Prof. S.K. Patra** and **Dr. S. Meher** for teaching me and also helping me how to learn. They have been great sources of inspiration to me and I thank them from the bottom of my heart.

I would like to thank all faculty members and staff of the Department of Electronics and Communication Engineering, N.I.T. Rourkela for their generous help in various ways for the completion of this thesis.

I would also like to mention the names of **Jitendra Kumar Das, Naresh Reddy,** and **Kamal Kumar** for helping me a lot during the thesis period.

I would like to thank all my friends and especially my classmates for all the thoughtful and mind stimulating discussions we had, which prompted us to think beyond the obvious. I've enjoyed their companionship so much during my stay at NIT, Rourkela.

I am especially indebted to my parents for their love, sacrifice, and support. They are my first teachers after I came to this world and have set great examples for me about how to live, study, and work.

Sagara Pandu

Roll No: 20507011

Dept of ECE, NIT, Rourkela

CONTENTS

ABSTRACT

Approximately 10% of the world's population suffers from some type of hearing loss, yet only small percentage of this statistic use the hearing aid. The stigma associated with wearing a hearing aid, customer dissatisfaction with hearing aid performance, the cost and the battery life. Through the use of digital signal processing the digital hearing aid now offers what the analog hearing aid cannot offer. Currently lot of attention is being given to low power VLSI design.

More and more people around the world suffer from hearing losses. The increasing average age and the growing population are the main reasons for this. The decimation filter used for hearing aid applications is designed and implemented both in MATLAB and VHDL. The decimation filter is designed using the distributed arithmetic multiplier in VHDL. Each digital filter structure is simulated using Matlab and its complete architecture is captured using Simulink. The resulting architecture is hardware efficient and consumes less power compared to conventional decimation filters. Compared to the comb-FIR-FIR architecture, the designed decimation filter architecture using Comb-half band FIR-FIR contributes to a hardware saving and reduces the power dissipation.

List of figures

LIST OF TABLES

Chapter 1

INTRODUCTION

1.1 MOTIVATION

Current research on radio frequency (RF) communication transceivers emphasizes both higher integration, to meet consumer demand for low-cost, low-power, small-form factor personal communication devices, and the ability to adapt to multiple communication standards. Higher integration can be obtained by using receiver architectures and circuit techniques that eliminate the need for external components. Utilizing a receiver architecture that performs channel select filtering on chip at base band allows for the programmability necessary to adapt to multiple communication standards.

In wireless and audio applications, oversampling analog-to-digital conversion has become popular because it achieves increased performance and flexibility by shifting signal processing complexity from analog to digital circuitry. Oversampling uses a sampling rate that is much greater than the bandwidth of the signal of interest. Digital signal processing (DSP) is then used to perform further filtering, decimation and even down conversion. The range of human hearing is generally considered to be 20 Hz to 20 kHz, but the ear is far more sensitive to sounds between 1 kHz and 4 kHz. Hence it is more useful to design a hearing aid application operating within the specified frequency range. Hearing aids are one of many modern, portable, digital systems requiring power efficient design in order to prolong battery life. Hearing aids perform signal processing functions on audio signals. With the advent of many new signal processing techniques, their requirement for higher computational ability has put additional pressure on power consumption.

Wireless telecommunication standards currently used throughout the world have channel bandwidths ranging from 6.25 kHz to 1.728 MHz. A multi-standard receiver that performs base band channel select filtering in the digital domain must have Sigma-delta analog-to-digital converter (ADC) with a wide dynamic range that can accommodate undesired channels as well as the desired. It must also be able to adapt to the various dynamic range requirements and sampling rates of the standards that are implemented. A wide dynamic range sigma-delta modulator can be used to meet these requirements for multiple standards. Sigma-delta ADC is a low-cost, low-bandwidth, low-power, high-resolution ADC and has varied applications in data acquisition, communications, signal processing and instrumentation.

A hearing aid application makes use of the oversampling concept and consists of a sigma-delta analog-to-digital converter (ADC) followed by a decimation filter as shown in Figure 1.1. The hearing aid interface circuit requires high speed and high-resolution analog-to-digital converters (ADC). Sigma-delta modulators are used to design the analog section, which requires hardware operating at high sampling rates. A sigma-delta ADC comprises of a modulator, which is analog in implementation and a digital decimator stage. The decimator which is a crucial part of a sigma-delta ADC relaxes the requirement for high precision analog circuits required for the modulator stage and also increases the final output resolution of the ADC. This work focuses mainly on the design and implementation of the decimator.

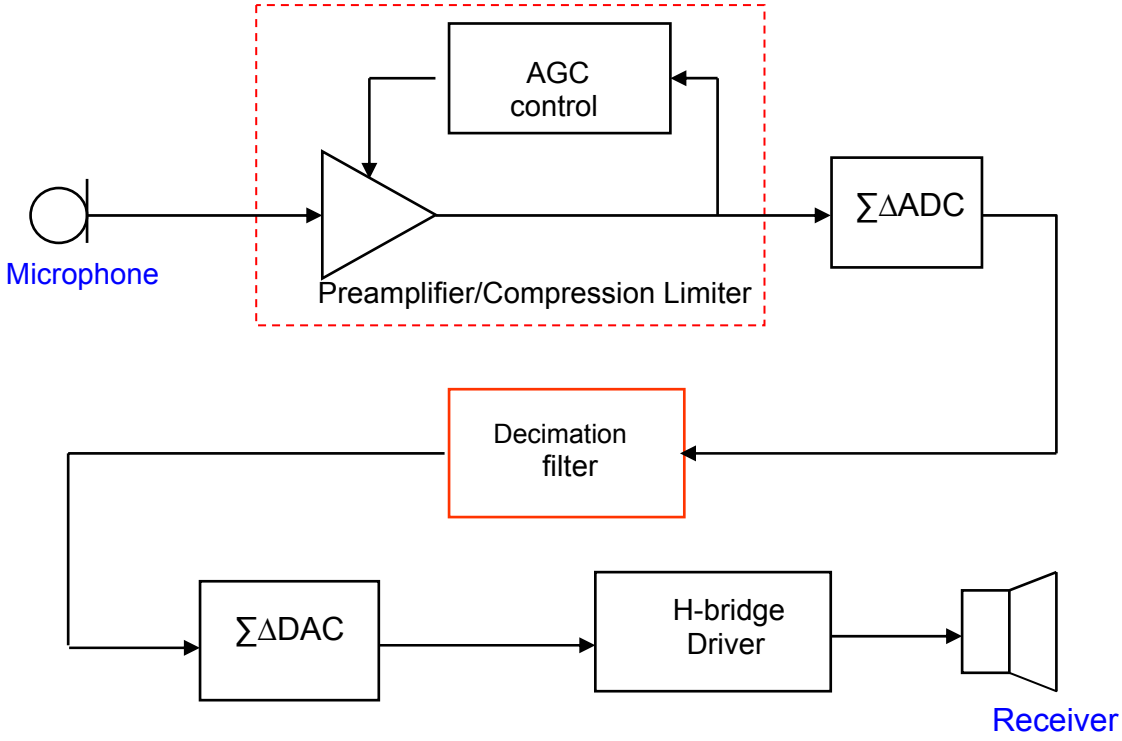


Fig1.1. the block diagram of the hearing aid application.

1.2 DIGITAL HEARING AIDS

Approximately 10% of the population suffers from some hearing loss, however only a small percentage of this category actually uses a hearing aid. There are several factors affecting market penetration. First, there is the stigma associated with wearing a hearing Aid. Second is customer dissatisfaction with the devices not meeting their expectations, third is the cost associated with the new digital versions of hearing aids

The recent development of commercial hearing aids with digital signal processing capabilities has allowed the development advanced signal processing techniques to aid the hearing impaired. The result for the wearer of the hearing aid is more accurate sound reproduction with minimum distortion and noise. Almost all of the largest hearing aid manufacturers have digital hearing aid products on the market, and of the 6 million hearing aids sold in 2001, approximately 20% were digital devices. In order to meet the small size and ultra-low power requirements of hearing aids, the existing solutions resort to custom ASIC devices for each hearing aid design. This increases the final cost of the hearing aid to spread of these digital instruments. By changing to a commercially available programmable DSP approach, the hearing aid companies could significantly reduce their costs, thereby a larger portion of the population with a lower price, better sound quality digital instrument.

There has been explosion in the number of digital hearing aids on the market in the last five years. Two very popular styles of hearing aids are available, behind the ear (BTE) & completely in the canal (CIC). Despite their higher cost, they were well received by clinicians & consumers.

Digital hearing aids today perform a variety of advanced digital signal processing algorithms, such as noise reduction and feedback canceling. For this reason hearing aids pose challenges for power and area requirements (for fabrication chip) compared to other portable devices.

It is important to determine whether digital hearing aids are better for patients; it is also significant to focus on the superior processing and features of these instruments. Digital hearing aids can't be described, as they are a separate entity from analog hearing aids. Digital simply indicates that the analog waveform is converted in to a string of number for processing, aid unfortunately; there is nothing inherently magical about this process. There for, digital is not superior just because its digital, but because digital signal processing (DSP) allows manufactures to hearing aids with enhance processing & features.

1.3 **THESIS GOALS**

The focus of this project is to design and implement a decimation filter and comb-half band FIR-FIR structure for hearing aid systems. The proposed architecture aims at reducing the hardware requirement and the power consumption. For the sake of comparison, we have designed and implemented a conventional comb-FIR-FIR decimation filter for the same specifications. This architecture is implemented using Distributed Arithmetic representation, which results in less hardware and less power consumption compared to other decimation filters.

1.4 THESIS ORGANIZATION

This thesis is organized as follows.
In chapter 2, we introduce sigma delta ADC and discuss about conventional digital to analog converter, decimator filter.
In chapter 3, we present the decimator theory and describe about comb filter, half band FIR filter, and Corrector FIR filter.
In chapter 4, we present the design of the CIC decimator filter structure and design of comb filter.
In chapter 5, we present the design of half band fir and fir filter.
In chapter 6, we present the description of Distributed Arithmetic
In chapter 7, design methodology and simulation results are given. The results include simulations done in both MATLAB and VHDL.
Finally we conclude in chapter 8.

Chapter 2

SIGMA-DELTA ADC OVERVIEW

2.1 INTRODUCTION

The performance of digital signal processing and communication systems is generally limited by the precision of the digital input signal which is achieved at the interface between analog and digital information. Sigma-Delta modulation based analog-to digital (A/D) conversion technology is a cost effective alternative for high resolution converters which can be ultimately integrated on digital signal processor ICs. The sigma-delta modulator was first introduced in 1962; but it gained the importance in recent times after the development in digital VLSI technologies. The increasing use of digital techniques in communication, biomedical and audio application has also contributed to the recent interest in cost effective high precision A/D converters. A requirement of analog-to-digital (A/D) interfaces is compatibility with VLSI technology, in order to provide for monolithic integration of both the analog and digital sections on a single die. The sigma deltą A/D converters are based on digital filtering techniques, almost 90% of the die is implemented in digital circuitry which enhances the prospect of compatibility.

Conventional converters are often difficult to implement in fine line very large scale integration (VLSI) technology. These difficulties arise because conventional methods need precise analog components in their filters and conversion circuits and because their circuits can be very vulnerable to noise and interference. The virtue of the conventional methods is their use of a low sampling frequency, which is usually the nyquist rate of the signal. By keeping these things in mind the people are going for over sampling converters, these converters make extensive use of digital signal processing. The main advantages of the sigma deltą A/D converters are mentioned below.

1. Higher reliability.
2. Increased functionality.
3. Reduced chip cost.

Those characteristics are commonly required in the digital signal processing environment of today. Consequently, the development of digital signal processing technology in general has been an important force in the development of high precision A/D converters which can be integrated on the same die as the digital signal processor itself. Conventional high-resolution A/D converters, such as successive approximation and flash type inverters, operating at the Nyquist rate (sampling frequency approximately equal to twice the maximum frequency in

the input signal); often do not make use of exceptionally high speeds achieved with a scaled VLSI technology. These Nyquist samplers require a complicated analog low pass filter (often called an anti-aliasing filter) to limit the maximum frequency input to the A/D, and sample-and hold circuitry. The high resolution can be achieved by the decimation process. Moreover, since precise component matching or laser trimming is not needed for the high-resolution sigma delta A/D converters, they are very attractive for the implementation of complex monolithic Systems that must incorporate both digital and analog functions. These features are somewhat opposite from the requirements of conventional converter architectures, which Generally require a number of high precision devices. The following chapter describes the concepts of noise shaping, over sampling, and decimation in some detail

2.2 SIGMA-DELTA ADC

Analog-to-digital converters can be categorized into two types depending upon the sampling rate. The first kind samples the analog input at the nyquist frequency fn such that $f_s = f_n = 2XB$, where fs is the sampling frequency and B is the bandwidth of the input signal. The second type of ADCs samples the analog input at much higher frequencies than the nyquist frequency and are called over sampling ADCs, sigma-delta ADCs come under this category. In sigma-delta ADC, the input signal is sampled at an over sampling frequency $f_s = KXfn$ where K is defined as the over sampling ratio and is given by the following equation.

$$K = \frac{f_s}{2B} \qquad (2.1)$$

The general block diagram and brief description of a sigma delta analog to digital Converter is shown fig 2.1. The modulator samples the analog input signal at much higher frequencies set by the over sampling ratio and converts the analog input signal into a pulse density modulated digital signal containing both the original input signal and the unwanted out-of-band noise. A decimation filter following the modulator filters out the out-of-band noise. Both the modulator and the decimator are operated with the same over sampling clock. In Fig. 2.1, the modulator shown is of first order with a 1-bit quantize and generates a 1-bit output. The output of the decimator is shown as N-bit digital data, where N is the output resolution of the ADC and is dependent on the over sampling ratio. The order of the decimator depends on the order of the modulator. Usually the order of the decimator is more than the order of the modulator.

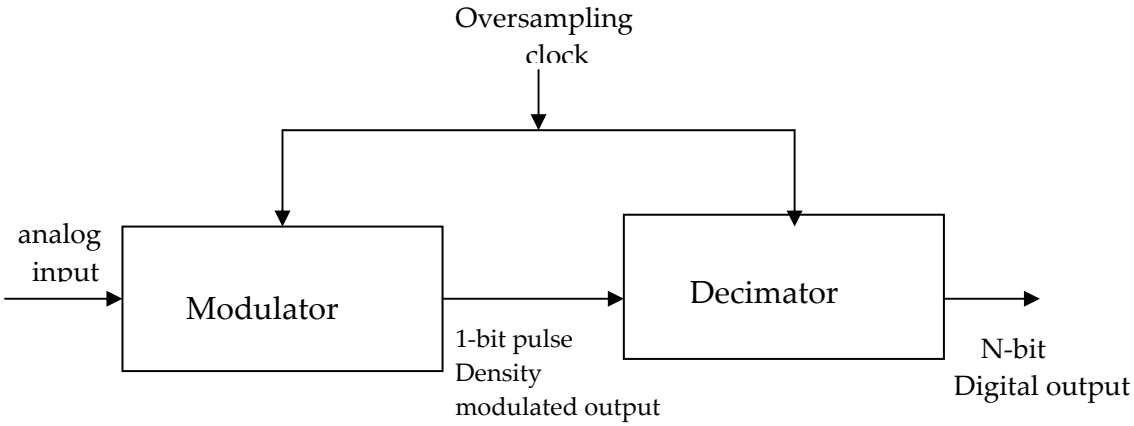


Fig 2.1 Block diagram of a sigma-delta analog to digital converter.

2.3 MODULATOR

The modulator is the analog part of a sigma-delta ADC. The final output resolution of the ADC is dependent on the order of the modulator and also the oversampling ratio set at the modulator stage. Since the modulator uses the principle of oversampling the need for antialiasing filter is eliminated and the analog input signal can directly be sampled using the oversampling clock.

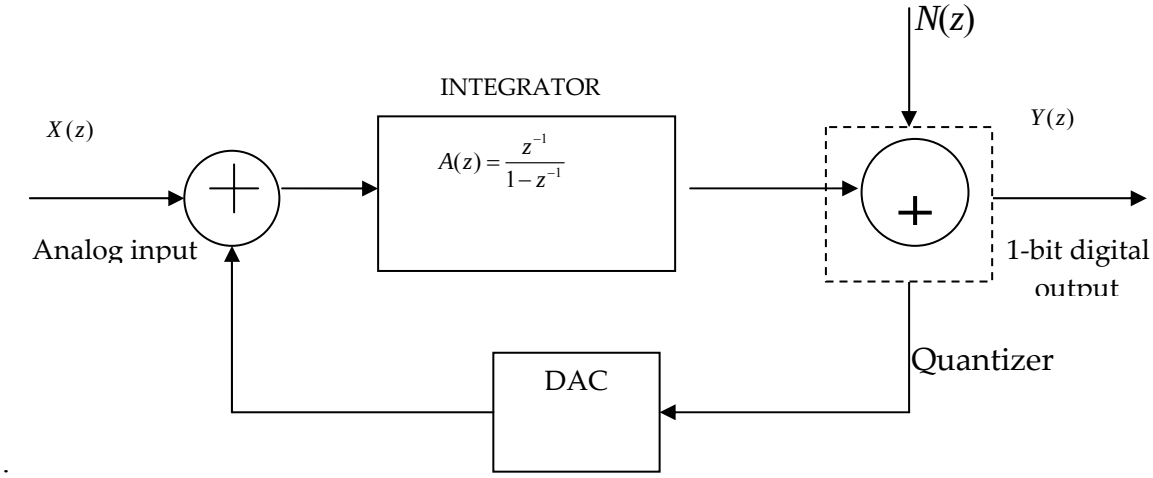


Fig 2.2 Block diagram of a first-order modulator .X(z) is the analog input signal, A(z) is the transfer function of the integrator and N(z) is the noise transfer function, Y(z) represents the output signal. DAC in the feedback loop is a 1-bit digital-to-analog converter.

Due to over sampling of the analog signal, the accuracy of the analog circuitry can be compromised with the speed. The modulator pushes the quantization noise to higher frequencies, which can be filtered out using a digital low pass filter at the decimation stage. The modulator outputs a 1-bit over sampled digital data, which is applied as an input to the

decimator. The basic block diagram of the modulator design is shown in Fig. 2.2 The difference between the analog input and the output of a digital-to-analog converter (DAC) is applied to an integrator, which is quantized to generate a pulse density modulated 1-bit digital output**.**

2.4 DECIMATOR

The process of digitally converting the sampling rate of a signal from a given higher rate f_s to a lower rate f_n is called decimation. Decimation in strict sense means reduction by 10 percent but in signal processing decimation means a reduction in sampling rate by any factor. Basically a decimator is a digital low pass filter, which also performs the operation of sample rate reduction. The sigma-delta modulator does operation of noise shaping and hence the noise is pushed to higher frequencies so that the decimation stage following the modulator can filter out this noise above the cutoff frequency, f_n. The band limited signal can then be resampled by discarding K – 1 samples out of every K samples, where K being the oversampling ratio. By averaging K samples out of the quantized sigma-delta output, the decimation filter achieves a high output resolution and also the frequency of the output data is at twice the input signal bandwidth which is the nyquist rate.

2.5. CONVENTIONAL DIGITAL TO ANALOG CONVERTORS

In general, signals are divided into two groups; an analog signal x(t),which can be defined in a continuous time domain and a digital signal, x(n),which can be represented as a sequence of numbers in a discrete-time domain .The time index n of a discrete-time signal x(n) is an integer number defined by sampling interval T. Thus, a discrete time signal, x*(t), can be represented by a sampled continuous-time signal x (t) as:

$$x^{*}x(t) = \sum_{n=-\infty}^{\infty} x(t)\,\delta(t - nT) \qquad (2.2)$$

Where

$$\delta(t) = \begin{cases} 1 & for \quad t = 0. \\ 0 & for \quad t \neq 0. \end{cases} \qquad (2.3)$$

A practical A/D converter transforms x (t) into a discrete-time digital signal, x*(t), where each sample is expressed with finite precision. Each sample is approximated by a digital code, i.e., x (t) is transformed into a sequence of finite precision or quantized samples x (n) .The generalized Analog to Digital Conversion process is given in the below figure2.3

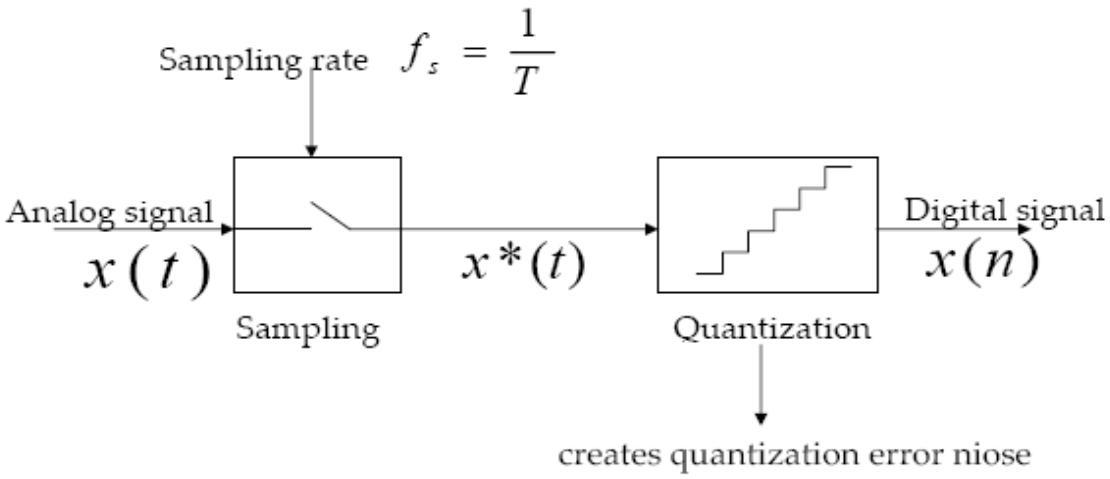


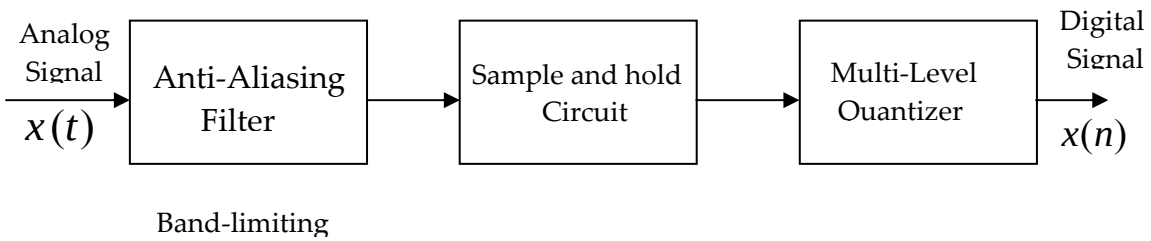
Fig 2.3 the generalized Analog to Digital Conversion Process

A/D converters can be classified into two groups according to the sampling rate criteria.

1. Nyquist rate converters
2. over sampling converters

Nyquist rate converters, such as successive approximation register (SAR), double integration and over sampling converters, sample analog signals which have maximum Frequencies slightly less than the Nyquist Frequency, $f_N = \frac{f_s}{2}$, where f_s is the sampling frequency. Meanwhile, over sampling converters perform the sampling process at a much higher rate, $f_N \ll f_s$, where f_s denotes the input sampling rate.

The following figure 2.4 represents the conventional A/D conversion process that transforms an analog input signal x(t) into a sequence of digital codes x(n) at a sampling rate of $f_s = \frac{1}{T}$,where T denotes the sampling Interval.



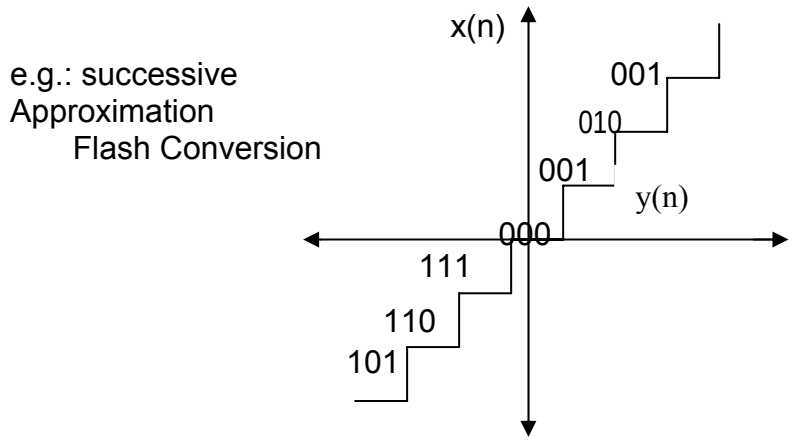


Fig 2.4 Conventional Analog to digital conversion process

Since $\delta(t-nT)$ in esq. 2-1 is a periodic function with period T, it can be represented in fourier series .

$$\sum_{n=-\infty}^{\infty} x(t)\,\delta(t-nT) = \frac{1}{T} \quad \sum_{n-\infty}^{\infty} x(t)\,e^{\frac{(j2\prod nt)}{T}}\,\frac{1}{2} \tag{2.4}$$

Combining (2.1) and (2.2) we get

$$x^*(t) = \frac{1}{T}\sum x(t)\,e^{\frac{(j2\pi nt)}{T}} = \frac{1}{T\sum_{n=-\infty}^{\infty} x(t)^{\,ej2\pi f_s nt}} \tag{2.5}$$

From Eqn. 2.4, we can know that the act of sampling is equivalent to modulating the input signal by carrier signals having frequencies at 0, fs, 2fs,….we can say in other way, the sampled signal can be expressed in the frequency domain as the summation of the original signal component and signals frequency modulated by integer multiplies of the sampling frequency as shown in fig.2-4. So from the above definitions, it is clear that input signals above the nyquist frequency can not be properly converted and they also create new signals in the base-band, which were not present in the original signal. This phenomenon is called as aliasing. This aliasing can be reduced by low pass filtering the input signal up to the nyquist frequency. This low pass filter can be called as anti aliasing filter. This low pass filter must have flat response over the frequency band of interest and should attenuate the frequencies

above the nyquist frequency. Because of the non linear phase distortion caused by this anti aliasing filter it may create harmonic distortion and audible degradation.

In addition to anti aliasing filter, a sample and hold circuit is required. Although the analog signal is continuously changing, the output of the sample and hold circuit must be constant between samples so that the signal can be quantized properly. This allows the converter enough time to compare the sampled analog signal to a set of reference levels that are usually generated internally. If the output of the sample –and –hold circuit varies during T, it can limit the performance of the A/D converter subsystem.

Each of these reference levels is assigned a digital code. Based on the results of the comparison, a digital encoder generates the code of the level the input signal is closest to. The resolution of such a converter is determined by the number and spacing of the reference levels that are predefined. For high-resolution Nyquist samplers, establishing the reference voltages is a serious challenge.

2. 6 ANALYZING FREQUENCY RESPONSE

Let us analyze the frequency domain characteristics of ADC with the technique of over-sampling. Where a DC conversion has a quantization error of up to ½ LSB, a sampled data system has quantization noise. As given in the following fig: below a perfect classical N-bit sampling ADC has an RMS quantization noise of $\frac{q}{\sqrt{2}}$ uniformly distributed within the Nyquist band of DC to $\frac{f_s}{2}$ (where q is the value of an LSB and fs is the sampling rate) . Therefore, its SNR with a full-scale sine wave input will be (6.02N+ 1.76) dB. If the ADC is less than perfect, and its noise is greater than its theoretical minimum quantization noise, then its effective resolution will be less than N-bits. Its actual resolution (often known as its Effective Number of Bits or ENOB) will be defined by

$$ENOB = \frac{(SNR - 1.76dB)}{6.02dB} \qquad (2.6)$$

In other case if choose the higher sampling rates, Kf_s, then the quantization noise will be distributed over a wider bandwidth form dc to $\frac{Kf_s}{2}$ as shown in the figure so we can apply a Low Pass Filter (LPF) to the output of the ADC to remove much of the quantization noise without effecting the wanted signal. Since the bandwidth is reduced by this digital output

filter, the data rate of the output may be lower than the original sampling rate but it still satisfies the nyquist criterion. This may be achieved by passing every M^{th} result to the output and discarding the remainder.

The process is known as "decimation" by a factor of M. M can have any integer value provided that the output data rate is more than twice the signal bandwidth.

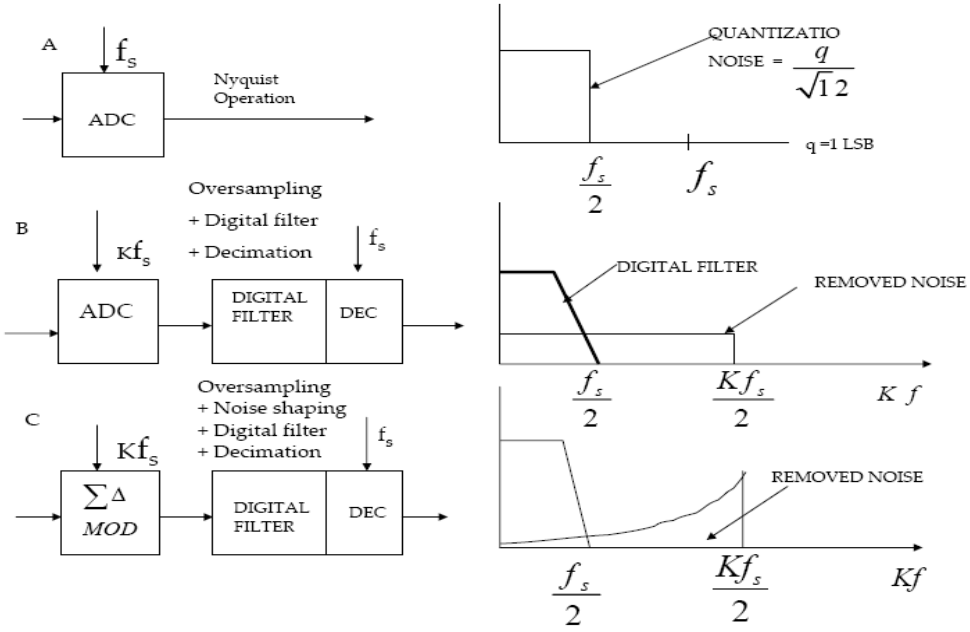


Fig.2.5 Basic operations involved in the Analog to Digital Converters

If we simply use over-sampling to improve resolution, we must over-sample by a factor of 22N to obtain an N-bit increase in resolution. The sigma delta converter does not need such a high oversampling ratio because it not only limits the signal pass band, but also shapes the quantization noise so that most of it falls outside this pass band as shown in Figure:

The following Figure shows a simple block diagram of a first order sigma delta Analog to-Digital Converter (ADC). The following fig: contains an ADC (it is generally known as a comparator), it is driven by the integrator whose input is an input signal summed with the output of the DAC which is again fed from the ADC output. If we add a digital low pass filter (LPF) and decimator at the output, then we have a sigma delta ADC. This sigma delta modulator shapes the quantization noise so that it lies above the pass band of the digital output filter, and then ENOB is much larger than the expected.

The general working of the sigma delta ADC is as follows, if we give a dc input at VIN. The integrator will constantly ramp up or down at node A. The output of the comparator id fed back through a 1 bit DAC to the summing input at the node B. There is a negative feedback loop from the comparator output through the 1 bit DAC back to the summing point. This will force the average DC voltage at node at node B to be equal to VIN. This implies that the average DAC output voltage must equal to the input voltage VIN. The average DAC output voltage is controlled by the ones-density in the 1-bit data stream from the comparator output. As the input signal increases towards +VREF, the number of "ones" in the serial bit stream increases, and the number of "zeros" decreases. Similarly, as the signal goes negative towards –VREF, the number of "ones" in the serial bit stream decreases, and the number of "zeros" increases. From a very simplistic standpoint, this analysis shows that the average value of the input voltage is contained in the serial bit stream out of the comparator. The digital filter and decimator process the serial bit stream and produce the final output data.

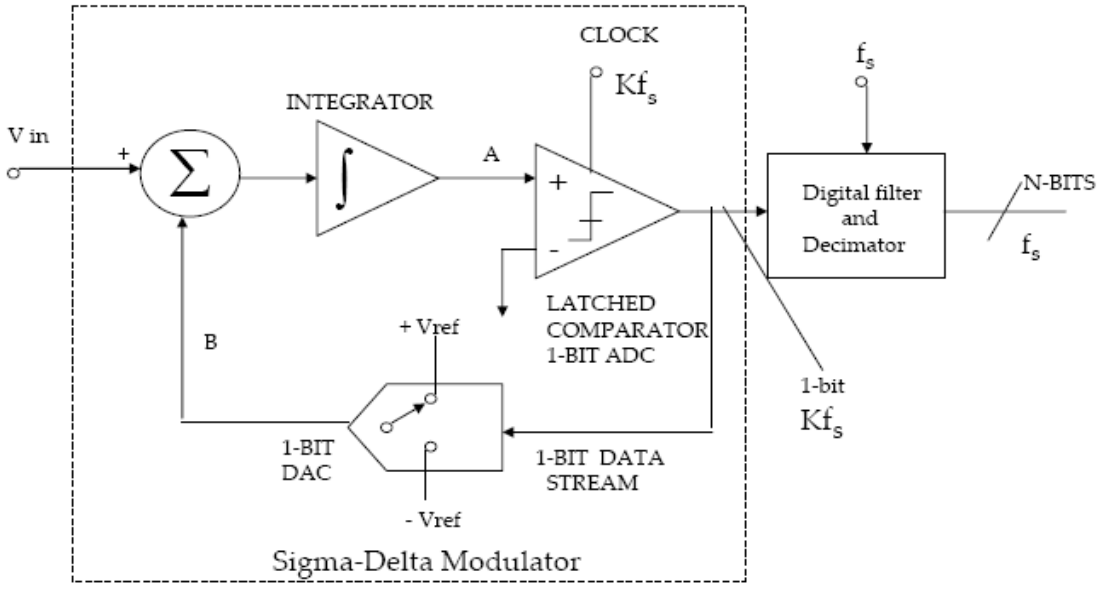


Fig 2.6 First order sigma delta ADC

The concept of noise shaping is explained in the frequency domain by the following figure. The integrator in the modulator is represented as an analog filter. Its transfer function is equal to $H(f) = \frac{1}{f}$. The amplitude response of this transfer function is inversely proportional to the input frequency. The quantization noise generated by the 1-bit quantizer is injected into the output summing block. If assume the input be X, and the Output Y, then the output of the

summer is X-Y, which is multiplied by filter transfer function, 1/f, and the result goes to one input to the output summer. So the output voltage is given by the following eq.

$$Y = \frac{1}{F(X - Y)} + Q \qquad (2.7)$$

After rearranging the above eq 2.8 and solved for Y in terms of X, f, and Q:

$$Y = \left(\frac{X}{f + 1}\right) + \left(\frac{Q.f}{f + 1}\right) \qquad (2.8)$$

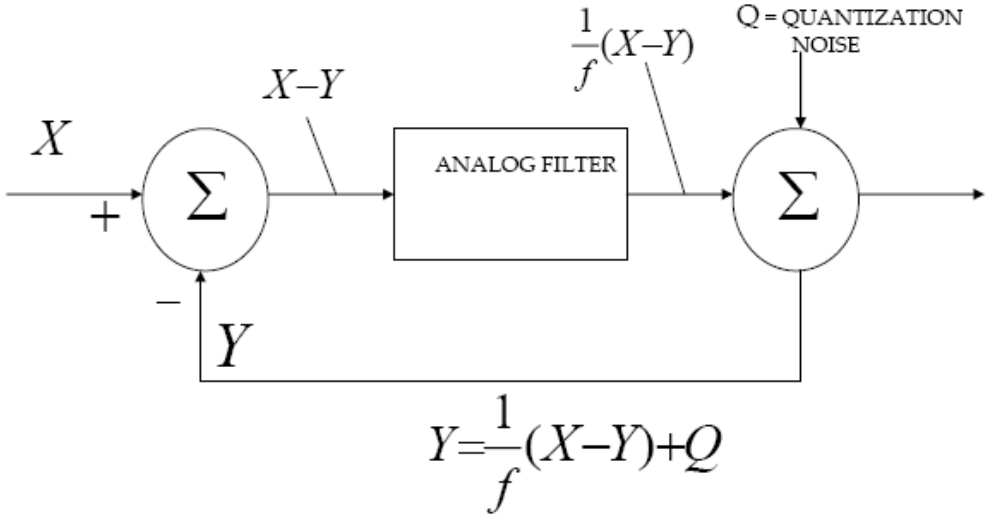


Fig 2.7 Noise shaping in the frequency domain

From the above equations it is clear that as frequency approaches zero, the output voltage Y approaches X with no noise component. At higher frequencies, the amplitude of the signal component decreases, and the noise component increases. Normally at higher frequencies the output consists primarily of quantization noise. The analog filter present in the circuit will have low pass effect on the signal, and a high pass effect on the quantization noise, by this detailed study we know that the analog filter performs the noise shaping function in the sigma delta ADC.

Chapter 3

DECIMATOR THEORY AND DESCRIPTION

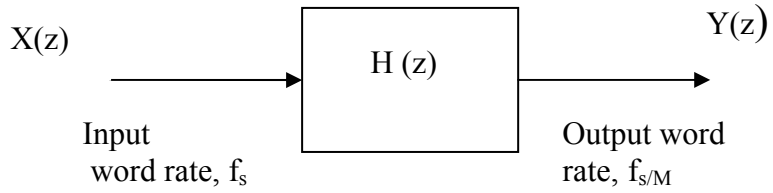
3.1 DECIMATOR THEORY

Decimation is the processes of lowering the word rate of a digitally encoded signal, which is sampled at high frequencies much above the nyquist rate. It is usually carried out to increase the resolution of an oversampled signal and to remove the out-of-band noise. In a sigma-delta ADC, oversampling the analog input signal by the modulator alone does not lower the quantization noise; the ADC should employ an averaging filter, which works as a decimator to remove the noise and to achieve higher resolutions. A basic block diagrammatic representation of the decimator is shown in Fig. 3.1.The decimator is a combination of a low pass filter and a down sampler. In Fig. 3.1 the transfer function, H(z) is representative of performing both the operations. The output word rate of the decimator is down sampled by the factor M, where M is the oversampling ratio. The function of low pass filtering and down sampling can be carried out using an averaging circuit. The transfer function of the averaging circuit is given by equation (3.1). It establishes a relation between the input and output functions (3.1).

$$H(z) = \frac{X(z)}{Y(z)} = \frac{1}{M}\sum_{x=0}^{M-1} z^{-x} \tag{3.1}$$

$$H(z) = \frac{1}{M}\frac{1-z^{-M}}{1-z^{-1}}$$

The averaging circuit defined by the equation (3.1) averages every M samples. By converting the z-domain transfer function into the frequency domain the characteristics of the circuit can be plotted. The frequency response of the decimator is given by equation (3.2) and the plot of the frequency response of the filter is shown in Fig. 3.2.



3.1: Basic block diagram of a decimator.

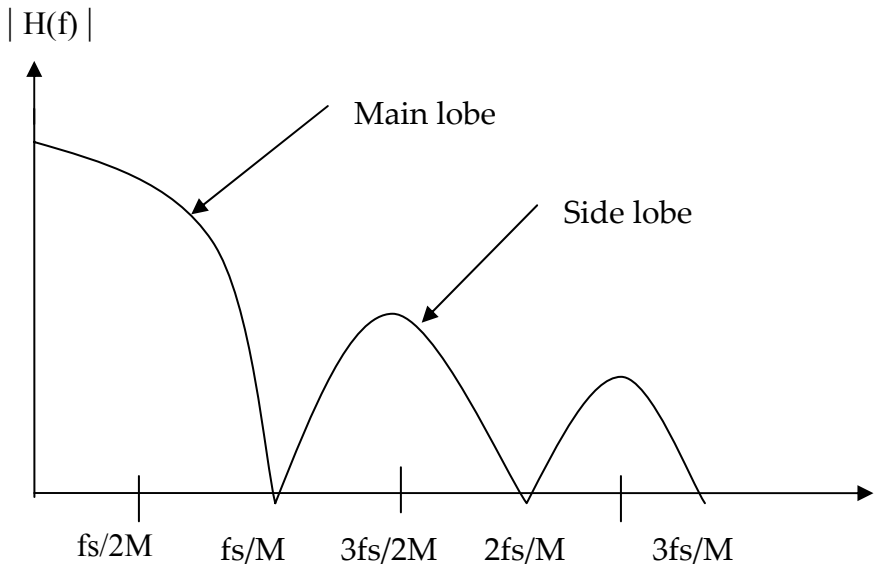


Fig. 3.2: Frequency response of a sync averaging filter.

The frequency response of the averaging circuit, which is used as a decimator, is similar to that of a sync filter. A sync filter is a digital low pass filter which can be used to filter out the high frequency noise from the modulated input signal.

$$H(z) = \frac{\sin c(M \pi \frac{f}{fs})}{\sin c(\pi \frac{f}{fs})} \qquad (3.2)$$

The signal band of interest is the range of frequencies from 0 to the signal bandwidth and $\frac{f_s}{2M}$ the sync filter attenuates any signal above the nyquist rate $\frac{f_s}{M}$, removing the out-of-band noise. In order for the decimator to satisfy the digital low pass filter characteristics the attenuation in the stop band should be high. The ratio of the main lobe to the side lobe forms a critical factor in designing a decimator. The filter characteristics can be improved to have sharp transition between the pass band and stop band and also good attenuation in the stop band by cascading the decimation stages. In Fig. 3.2, the gain of the filter is given by the value M. Increasing the value M simply means increasing the final output resolution and has no direct significance on the frequency response. The decimator averages every M samples and has an output at every M_{th} sample. As defined earlier, M is the oversampling ratio and since the output occurs at every M_{th} sample, the output rate of the decimator is $\frac{f_s}{M}$. The input

to a decimator is the sequence of bits of 1's and 0's and since the averaging operation involves the addition of these bits, the output resolution increases due to the addition of every M number of bits. As the M value increases the output resolution also increases. The relation between the numbers of bits increased to the oversampling ratio K for a sigma-delta ADC is given by equation (3.3)

$$N_{inc} = \frac{30 \log M - 5.17}{6.02} \tag{3.3}$$

The final output resolution of the decimator not only depends on the oversampling ratio but also on the input resolution. The output of the modulator is applied as an input to the decimator but the output resolution of the modulator depends on the order of the modulator designed. In this work, a first order modulator is considered; hence the output of the modulator is a 1-bit digital data. The final resolution of the decimator or the sigma-delta analog-to-digital converter is given by equation (3.4).

$$N_{final} = N_{\frac{i}{p}} + N_{inc} \tag{3.4}$$

Where N_{final} is the final output resolution, $N_{i/p}$ is the input resolution of the decimator and N_{inc} is the increase in resolution achieved by the decimator. In the present work, the decimator is designed to form a sigma-delta ADC by cascading it with an already designed modulator. The designed modulator is of first order and hence based on the following equation (3.5), the order of the decimator has to be two.

$$K = 1 + M \tag{3.5}$$

In equation (3.5), K is the order of the decimator and M is the order of the modulator. The complete transfer function of the decimator of order L is given in equation (3.6).

$$H(z) = \left(\frac{1}{M} \frac{1 - z^{-M}}{1 - z^{-1}} \right)^{K} \tag{3.6}$$

Higher order decimators can be designed by cascading single stage decimation stages. By designing a second order decimator improvement in filter characteristics is achieved.

3.2 CASCADED INTEGRATOR COMB (CIC) FILTER THEORY

The CIC filter is a combination of digital integrator and digital differentiator stages which perform the operation of digital low pass filtering and decimation. The CIC filters do not require any multiplier circuits and hence are very economical for implementation in hardware and the problems with cascading faced by the accumulate and dump circuit are also overcome with the CIC design. Equation (3.7) gives the transfer function of the CIC filter in z-domain which is similar to equation (3.6) except the numerator term and the denominator terms are separated. The numerator represents the transfer function of a differentiator and the denominator indicates that of an integrator.

$$H(z) = H^{N}_{combn}(z) * H^{N}_{integrator}(z) = (1 - z^{-MD})^{N} * \left(\frac{1}{1 - z^{-1}}\right)^{N} = \left(\sum_{K=0}^{MD} z^{-K}\right)^{N} \quad (3.7)$$

$$H(z) = \frac{1}{M^{K}}\left(\frac{1 - z^{-M}}{1 - z^{-1}}\right)^{K} \quad (3.8)$$

The transfer function of both the CIC filter and the accumulate and dump circuit are Similar but the way both circuits are implemented are different. The CIC filter first performs the averaging operation then follows it with the decimation not like in accumulate and dump circuit where the averaging and decimation operations occur at the same time. A simple block diagram of a first order CIC filter is shown in Fig. 3.4. The hardware needed to implement the CIC filter shown in Fig. 2.4 is very significant because of the delay elements that are used in the differentiator stage. It can be seen that the differentiator circuit needs M delay elements. Usually the delays are implemented using registers. As will be discussed in the later sections that as the oversampling ratio increases, the number of delay elements also increases and as well the number of register bits that are used to store the data. The above design also requires another decimation circuit for decreasing the data rate, which requires additional hardware. Hence it becomes very cumbersome to design the differentiator with many delay elements. The problem with the area can be overcome by implementing a clock divider circuit in between the integrator and differentiator stages as shown in Fig. 3.4. The clock divider circuit divides the oversampling clock signal by the oversampling ratio M. By dividing the clock frequency by M the number of delay elements used in the differentiator can be reduced to just one. In Fig.3.4, the integrator operates at the sampling clock frequency, f_s while the differentiator operates at down sampled clock frequency of $\frac{f_s}{M}$ by operating the

differentiator at lower frequencies; a reduction in the power consumption is achieved. In this research, the CIC filter based on this model has been designed.

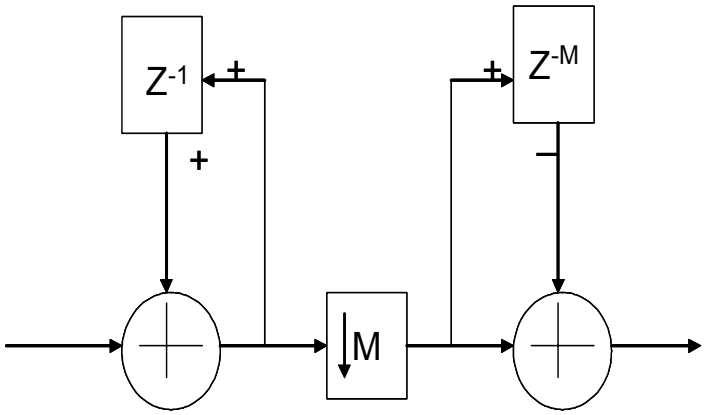


Fig. 3.3. First order CIC filter with a decimation factor

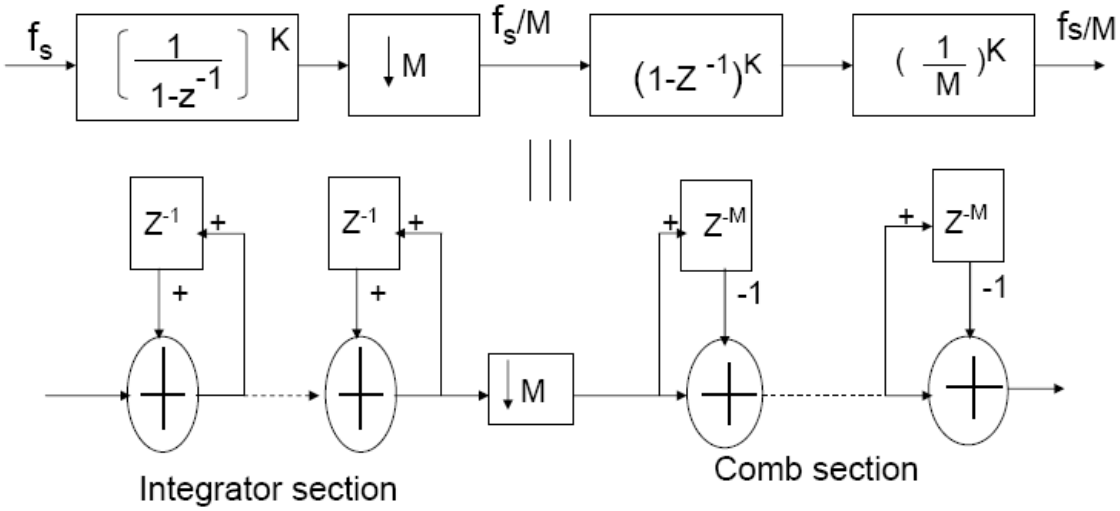


Fig 3.4. Block diagram of basic structure of CIC Decimation filter

3.2.1 The Integrator

The integrator circuit is similar to an accumulator which is used to accumulate or store the sum of the input data. It is a single-pole Infinite Impulse Response (IIR) filter with a filter coefficient factor of one. The transfer function of the integrator is shown in equation 3.8. The integrator section of CIC filters consists of N digital integrator stages operating at high sample rate f_s .

$$y(n) = x(n) + y(n-1) \qquad (3.8)$$

This system is also known as an accumulator. The transfer function for an integrator on the z-plane is

$$H(z) = \frac{1}{1 - Z^{-1}} \qquad (3.9)$$

$$\left|H(f)\right| = \frac{1}{\sqrt{2\left(1 - \cos(2\pi \frac{f}{f_s}\right)}} \qquad (3.10)$$

$$\left|H_1(e^{jw})\right|^2 = \frac{1}{2(1 - \cos w)}$$

The output of the integrator is the sum of the present input and the past output as can be Observed from the time domain representation equation (3.8). Based on equation (3.8), a stick diagrammatic representation of the digital integrator can be modeled and is shown in Fig. 3.5. The delay element is used to delay the output signal by one clock period and can be implemented using a memory element. A simple register can be used to achieve the delay. The transfer function in z-domain representation can be converted into a frequency domain by substituting Z with $e^{j2\pi\ (f/fs)}$. The magnitude response of the integrator is given by equation (3.10) and the plot showing the magnitude response of the integrator is shown in Fig 3.6

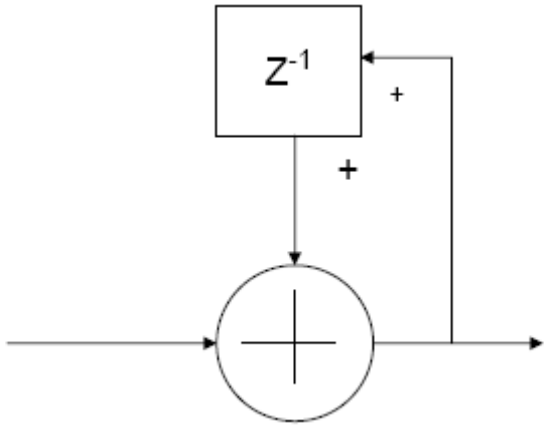


Fig 3.6.Bacic Integrator

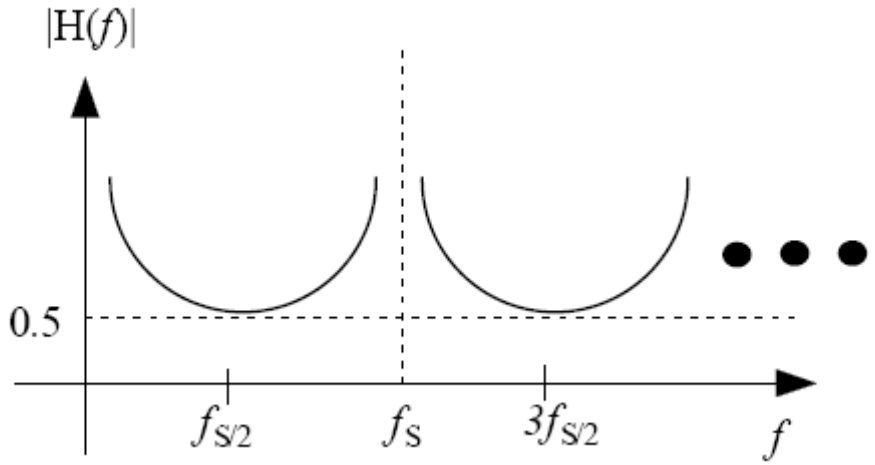


Figure 3.6: Magnitude response of a digital integrator.

The magnitude plot of the integrator shows that the integrator has an infinite gain at DC and at multiples of the sampling frequency f_s. The integrator has a minimum value of 0.5 but the frequency of operation that is of interest is at f_s. Since the gain is infinite at DC and it f_s. might cause the integrator to become unstable and there is every chance that the register used in the delay element could overflow causing data loss. In order to avoid problems with register overflow, two's complement coding scheme is used.

3.2 .2 The Differentiator (comb filter)

A differentiator circuit also called as a comb filter is a Finite Impulse Response (FIR) filter. A comb filter is a digital low pass filter. The time domain and the transfer function of the differentiator are given in equation (3.11). From the time domain representation it can be explained that the output of the differentiator is the difference between the present input and the past input.

$$y(n) = x(n) - y(n - DM) \qquad (3.11)$$

$$H(z) = \frac{Y(z)}{X(z)} = (1 - z^{-1})$$

In this equation, D is a design parameter used to control the filter's frequency response. D can be any positive integer, but it is usually limited to 1 or 2. The corresponding transfer at f_s.

$$H_{comb}(z) = 1 - z^{-MD} \qquad (3.12)$$

When M = 1 and D = 1, the power response is a high-pass function with 20 dB per decade (6 dB per octave) gain (after all, it is the inverse of an integrator). When MD$\neq$ 1, then the power response takes on the familiar raised cosine form with MD cycles from 0 to 2π.Based on equation (3.10), the stick diagram of the differentiator can be modeled and is shown in Fig. 3.8. The transfer function is converted into the frequency response and the expression for the magnitude response is given by equation (3.13). The plot of magnitude response of the differentiator is shown in Fig. 3.8. The two's complement output of the integrator is applied as the input to the differentiator, and so the differentiator also uses the two's complement scheme of coding.

$$|H(f)| = \sqrt{2\left(1-\cos(2\pi\frac{f}{f_s}\right)} \qquad (3.13)$$

The final output of the decimator which is the output of the differentiator circuit has to be in binary form for further signaling processing. So the two's complement output is converted back to the binary form. It should be noted that the differentiator operates at a different clock frequency compared to the clock frequency of the integrator as explained above. Because of this, both the circuits act as individual blocks and can be used for cascading in order to form a cascaded integrator comb filter.

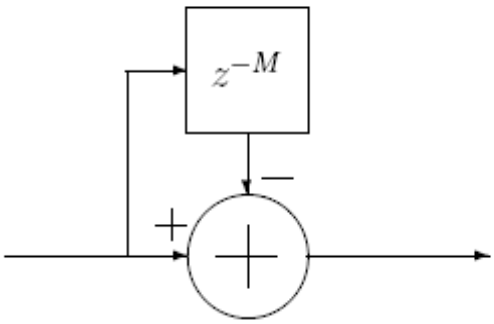


Fig 3.7 the basic comb filter

To summarize, a CIC decimator would have N cascaded integrator stages clocked at f_s followed by a rate change by a factor M, followed by N cascaded comb stages running at $\frac{f_s}{M}$ (3.9). As each integrator has unity feedback coefficient this result in register overflow in all integrator stages. This can be diluted if the two following points are met: firstly, the filter is

implemented with two's complement arithmetic or other number system which allows wrap around between the most positive and most negative numbers; and secondly the range of number system is equal to or exceeds the maximum magnitude expected at the output of the composite filter.

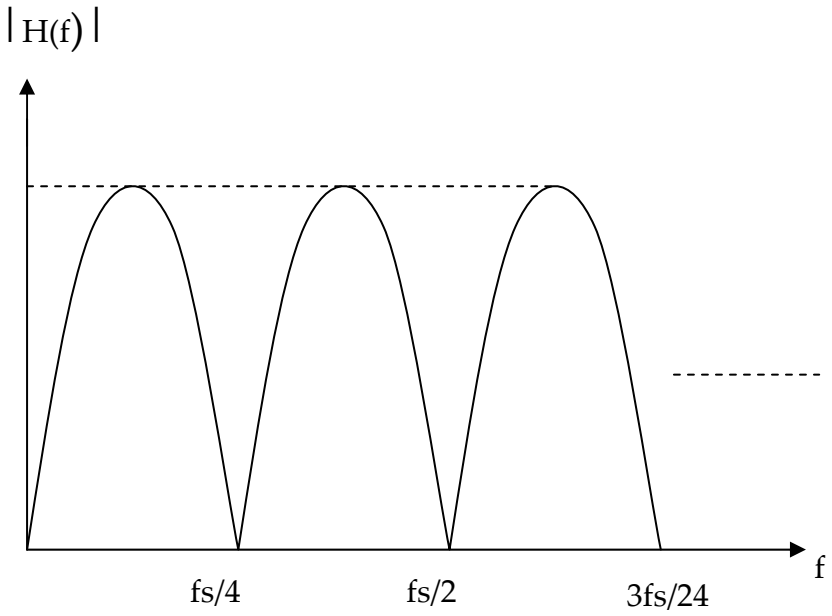


Figure 3.8: Magnitude response of a digital differentiator.

3.2.3 Multi-Order CIC Filter

As explained above a CIC filter is formed by cascading the digital integrator and the digital differentiator. In order to have better filter characteristics higher order CIC filter is designed based on equation (3.5). In the present work, a second order CIC filter is designed, its block diagram of it is shown in Fig. 3.11. The filter has a second order integrator and a second order differentiator circuits separated by a clock divider circuit. The integrator stage and the differentiator stage operate at different frequency. The complete block diagram of the N order CIC decimation filter implementation is shown in Fig. 3.11.

The integrator section of CIC filters consists of N ideal digital integrator stages operating at the high sampling rate fs. Each stage is implemented as a one-pole filter with a unity feedback coefficient. The system function for a single integrator is shown in eqn (3.9). The comb section operates at the low sampling rate $\frac{f_s}{M}$. Where *M* is the integer rate change factor. This section consists of *N* comb stages with a differential delay of *D* samples per stage. The differential delay is a filter design parameter used to control the filter's frequency

response. In practice, the differential delay is usually held to D= 1 or 2. The system function for a single comb stage referenced to the high sampling rate is (3.12). There is a rate change switch between the two filter sections

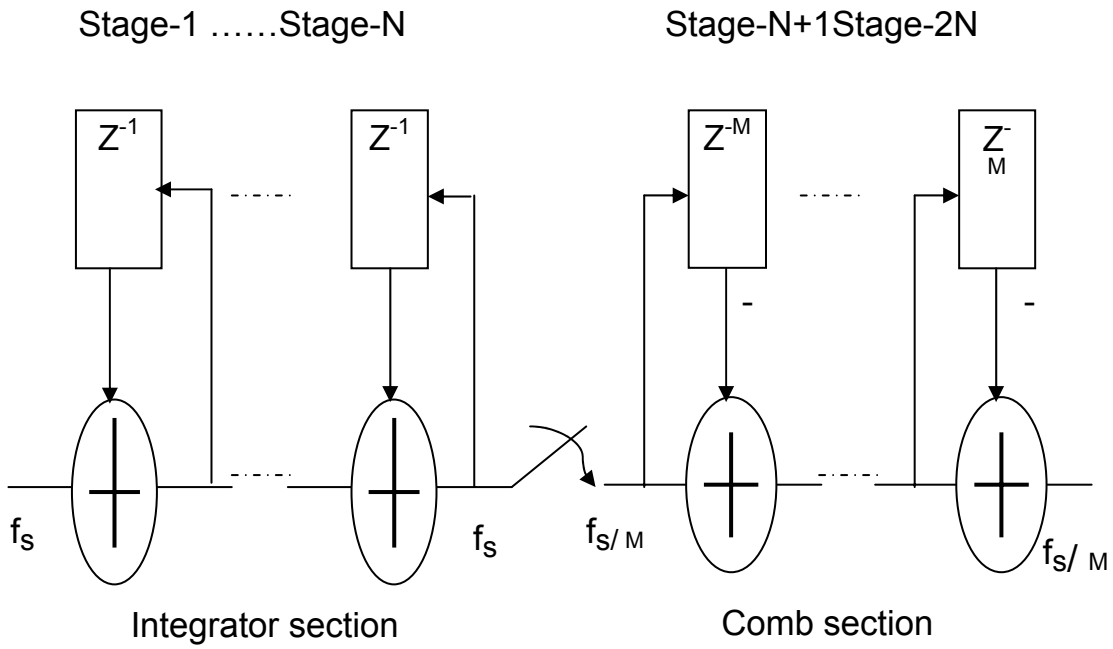


Fig 3.9 the N^{th} order Block diagram of the CIC Decimation filter

For decimation, the switch sub samples the output of the last integrator stage, reducing the sampling rate from fs to *fs/M.* and for interpolation, the switch causes a rate increase by a factor of M by inserting *M - 1* zero valued samples between consecutive samples of the comb section output. It follows from (3.9) and *(3.12)* that the system function for the composite CIC filter referenced to the high sampling rate fs is shown in (3.8).It is implicit from the last form of the system function that the CIC filter is functionally equivalent to a cascade of N uniform FIR filter stages. A conventional implementation consists of a cascade of ***N*** stages each requiring *MD* storage registers and one accumulator. Taking advantage of the rate change factor, one of the ***N*** stages can be simplified to use only *D* storage registers.

The economics of CIC filters derive from the following sources:

- No multipliers are required
- No storage is required for filter coefficients
- Intermediate storage is reduced by integrating at high sampling rate and comb Filtering at low sampling rate
- Little external control or complicated local timing is required
- This design can also be used for different ranges of rate change factors, R, with the addition of a scaling circuit and minimal changes to the filter timing.

Applications of CIC Filters

The application for CIC filters seems to be in areas where higher sampling rates make multipliers an uneconomical choice and areas where large rate change factors would require large amounts of coefficient storage or fast impulse response generation

3.2.4 Frequency Characteristics

The transfer function for a CIC filter at f_s is

$$H(z) = H^{N}_{combn}(z) * H^{N}_{integrator}(z) = (1 - z^{-MD})^{N} * \left(\frac{1}{1 - z^{-1}}\right)^{N} = \left(\sum_{K=0}^{MD} z^{-K}\right)^{N} \qquad (3.9)$$

This equation shows that even though a CIC has integrators in it, which by themselves have an infinite impulse response, a CIC filter is equivalent to N FIR filters requiring MD storage registers per stage, each having a rectangular impulse response. Since all of the coefficients of these FIR filters are unity, and therefore symmetric, a CIC filter also has a linear phase response and constant group delay In general, CIC filters have a low pass frequency characteristic. The frequency response is given by

$$z = e^{\frac{j2\pi f}{M}} \qquad \text{for} \qquad 0 \le f \le \frac{M}{2} \qquad (3.10)$$

Where f is the frequency relative to low sampling rate, $\frac{f_s}{M}$

$$n_0^2 = \int_0^{f\,0} e^2(f)df = e_{rms}^2(2 f_0 T) = \frac{e_{rms}^2}{OSR} \quad f_0 < f < \frac{f_s}{2}$$

as part of filter design process, R, M and N are chosen to provide acceptable passband characteristics over the frequency range from 0 to a predetermined cut-off frequency f_c expressed relative to low. Sampling rate. The power response is

$$P(f) = \left|\frac{\sin \pi\pi Df}{\sin \frac{\pi f}{M}}\right|^{2N} \qquad (3.11)$$

For larger values of R, the power response over a limited frequency range by

$$F(z) = \sum_{k=0}^{L-1} f[k] z^{-k}$$

$$p(f) = \left[MD \frac{\sin \pi Df}{\pi Df} \right]^{2N} \qquad \text{For } 0 \le f \le \frac{1}{D} \qquad (3.12)$$

This approximation can be used for many practical design problems. For instance, the error between P and P^ is less than 1dB for MD≥10, 1≤N≤7 and 0 ≤ f ≤ 255 The magnitude response at the output of the filter can be shown to be

$$H(f) = \left| \frac{\sin \pi Df}{\sin \frac{\pi f}{M}} \right|^{N} \qquad (3.13)$$

By using the relation sinx ≈ x for small x and some algebra. We can approximate this Function for large M as

$$H(f) = \left| MD \frac{\sin \pi Df}{\pi Df} \right|^{N} \qquad \text{for} \quad 0 \le f < \frac{1}{D} \quad (3.14)$$

We can notice a few things about the response. One is that the output spectrum has nulls at multiples of $1 < \frac{1}{D}$. D is also used to control the placement of nulls. The region around every M^{th} null is folded into the passband causing aliasing errors. In addition, the region around the null is where aliasing/imaging occurs. If we define fc to be the cut-off of the usable passband, then the aliasing/imaging regions are at

$$(i - f_c) \le f \le (i + f_c) \qquad (3.15)$$

for $f \le \frac{1}{2}$ and i = 1, 2, _ _ _ ,[$\frac{M}{2}$]. if $f_c \le \frac{D}{2}$, then the maximum of these will occur at the lower edge of the first band, $1 - f_C$. The system designer must take this into consideration, and adjust M, D, and N as needed.

Chapter 4

DECIMATION FILTER STRUCTURE AND DESIGN

4.1 FILTER STRUCTURE AND DESIGN

In this work the design of a decimation filter is presented for integrating with an existing designed modulator to form a complete sigma-delta ADC. we use multi-stage decimation filter which means the single decimation filter is replaced by cascaded filters. In this chapter, we are going to talk about the filter architecture used in this work, including their structures, strengths and drawbacks. the first step in designing a decimation filter is to decide which types of filters will be used and where decimation will occur. This chapter explores the issues involved in choosing filter architecture for a hearing aid application. The relative power of several architectures is compared, resulting in the three-stage architecture that is chosen to implement this filter.

Decimation filters must be computationally efficient since the filtering is usually performed at a high rate. A sigma-delta ADC generates a bit-stream at a rate of 1.28 MHz. This bit stream passes through a decimation filter and is down-sampled to a rate of 20 kHz. we use a decimation factor of 16 to decimate from 1.28 MHz to 80 kHz. We design a narrow band transition decimation filter which has a transition band of 10 kHz with the input sampling frequency of 1.28 MHz, a pass band of 4 kHz and cutoff frequency of 15 kHz. we have chosen a pass band of 4 kHz because the ear is sensitive to all sounds within 4 kHz. The filter has a pass band ripple of 0.001, which corresponds to the flat response of the filter. Since the transition band is a small percentage of the sampling rate, the filter will have many taps, and the zeros will be closer to each other causing the filter to be more sensitive to noise. The conventional decimation filter is implemented by this method, and the proposed system eliminates such problems by using a multistage multirate approach. The power consumption is directly proportional to the number of taps and the operating frequency. We have taken the specification of designed decimation filter is shown in Table (1).

Sampling frequency (Fs)	1.28 MHz
Decimation factor (M)	16
Pass band frequency ripple	4 kHz
Pass band ripple	0.001
Cutoff frequency	15 kHz
Decimation filter stages (N)	5

Table1. Filter specifications.

We have implemented the decimation filter using a half-band FIR-FIR structure. The block diagram of the complete decimation filter is presented in Figure 4.1.

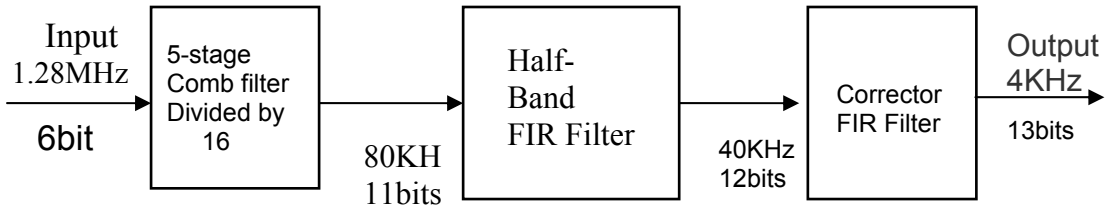


Figure 4.1 The block diagram of the proposed system

4.2. IMPLEMENTATION OF THE COMB FILTER

The input of the decimation filter is generated from a sigma-delta modulator running at 1.28 MHz. At the sigma-delta modulator output, the out-band signals surrounding multiples of sampling frequency alias into the desired band after decimation occurs. Comb filter can be designed to present a notch at each of the frequencies that will alias to the base-band. Moreover, comb filter needs no multiplier and consists of simple operations suitable at high frequencies. The response of the comb filter is a low pass filter with a sharper cutoff. Due to these reasons, comb filters are used for the first stage of decimation. Naviner suggests connecting a cascade of comb filters to reduce the sampling frequency, which will be used as the input to the successive stages. The comb filter decimates by a factor *M* as shown in (4.1).

$$H(z) = \frac{1}{M^K}\left(\frac{1-z^{-M}}{1-z^{-1}}\right)^K \qquad (4.1)$$

The integrator section of CIC filters consists of N ideal digital integrator stages operating at the high sampling rate 1.28 MHz. Each stage is implemented as a one-pole filter with a unity feedback coefficient. the system function for a single integrator is (3.9).

The comb section operates at the low sampling rate $\frac{f_s}{M}$ where *M* is the integer rate change factor. This section consists of ***N*** comb stages with a differential delay of *M* samples per stage. The differential delay is a filter design parameter used to control the filter's frequency response. In practice, the differential delay is usually held to $M = 1 \text{or } 2$. The system function for a single comb stage referenced to the high sampling rate is (3.12).

There is a rate change switch between the two filter sections. For decimation, the switch sub samples the output of the last integrator stage, reducing the sampling rate from $fs,$ to $\frac{f_s}{M}$. and for interpolation, the switch causes a rate increase by a factor of *M* by inserting *M - 1* zero valued samples between consecutive samples of the comb section output (3.12).

The cascade of comb filters decimates the input signal by half the over sampling ratio to simplify the FIR filter computation. Cascading several multirate stages increases the design efficiency, which reduces the total number of coefficients and in turn results in hardware savings. The cascade of comb filters is connected to a divide-by-*N* down-sampler, which corresponds to the decimation factor. Because we consider a sigma–delta modulator with order L = 4, a cascade of K=5 comb filters is necessary (K =L+1). As a result, the Comb filter is implemented as a cascade of five integrators followed by a divide by M =16 down-sampler and five differentiators as shown in fig (4.2)

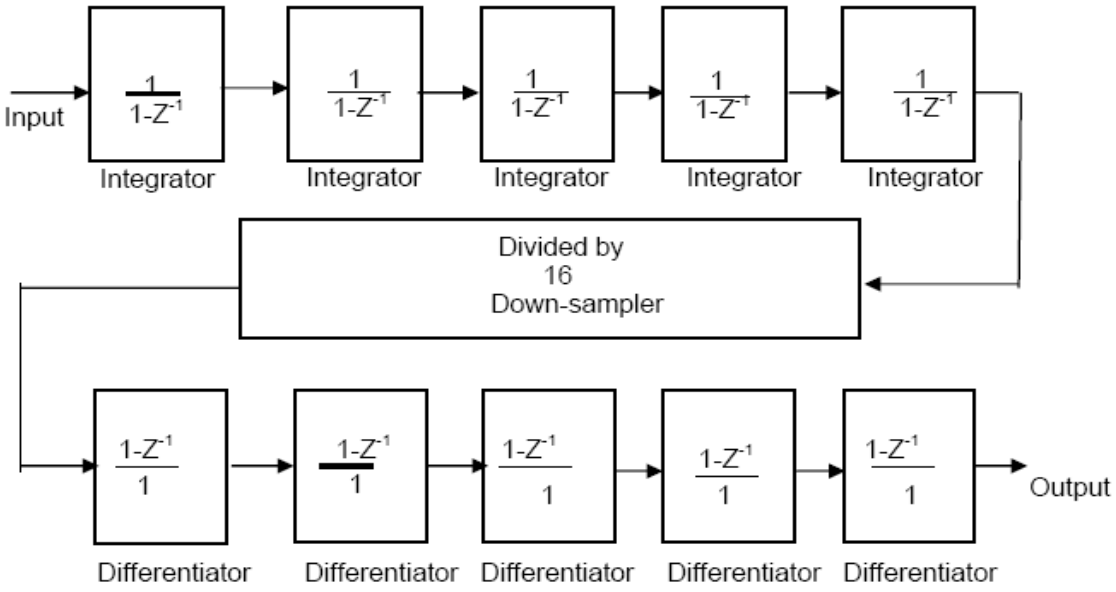


Fig 4.2: The 5-Stage divide bye 16 comb

The comb filter provides attenuation of about –65 dB as shown in Figure (4.3)**.** The comb filter stage has a $\sin c^6(f)$ frequency response with a decimation factor 16.

The comb filter decimates from 1.28 MHz to 80 kHz due to a decimation factor of *M* = 16. Unfortunately, comb filter design presents two drawbacks .an insufficient attenuation in stop

band and distortion in pass band. Insufficient attenuation can be overcome by cascading several filters. A corrector filter can compensate distortion in pass band.

CIC structure is very popular in the first stage design of the multi-stage decimation filter design since it has lot of appreciated characteristics, especially in the high sample frequency field. It is apparent to find it economical because CIC filter has not multipliers and storage for filter coefficient. As the price, it results in undesired passband gain leading to too large passband ripple which has bad effect on the signal demodulation. Therefore, we need a compensation filter to resolve this loss.

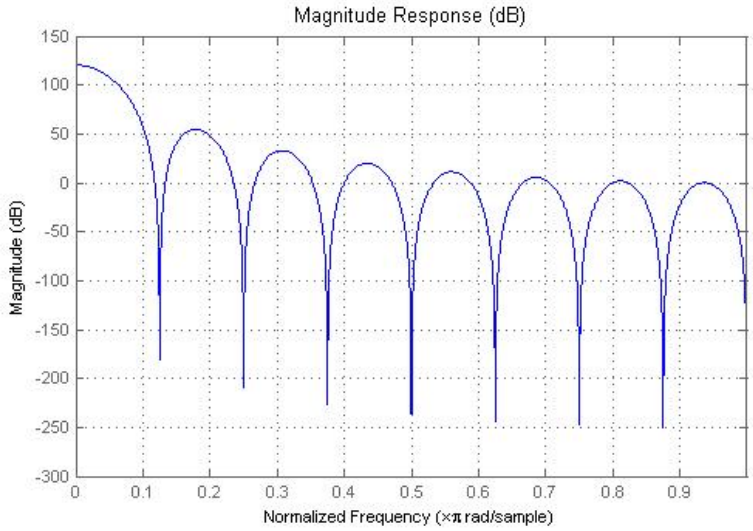


Figure 4.3 Attenuation for 5-stage comb filter

The specifications of our decimation filter are given below:

Input sampling frequency (Fs) : 1.28 MHz
Output sampling frequency : 80 kHz
Rate change factor (M) : 16
Differential Delay (D) : 1
Number of Stages (K) : 5
Pass band Ripple : 0.001dB

Chapter 5

FIR AND HALF-BAND FILTER DESIGN

5.1 FIR FILTER DESIGN

A finite impulse response (FIR) filter is a filter structure that can be used to implement almost any sort of frequency response digitally. An FIR filter is usually implemented by using a series of delays, multipliers, and adders to create the filter's output. Fig 5.1.shows the basic block diagram for an FIR filter of length N. The delays result in operating on prior input samples. The h(m) values are the coefficients used for multiplication, so that the output at time n is the summation of all the delayed samples multiplied by the appropriate coefficients.

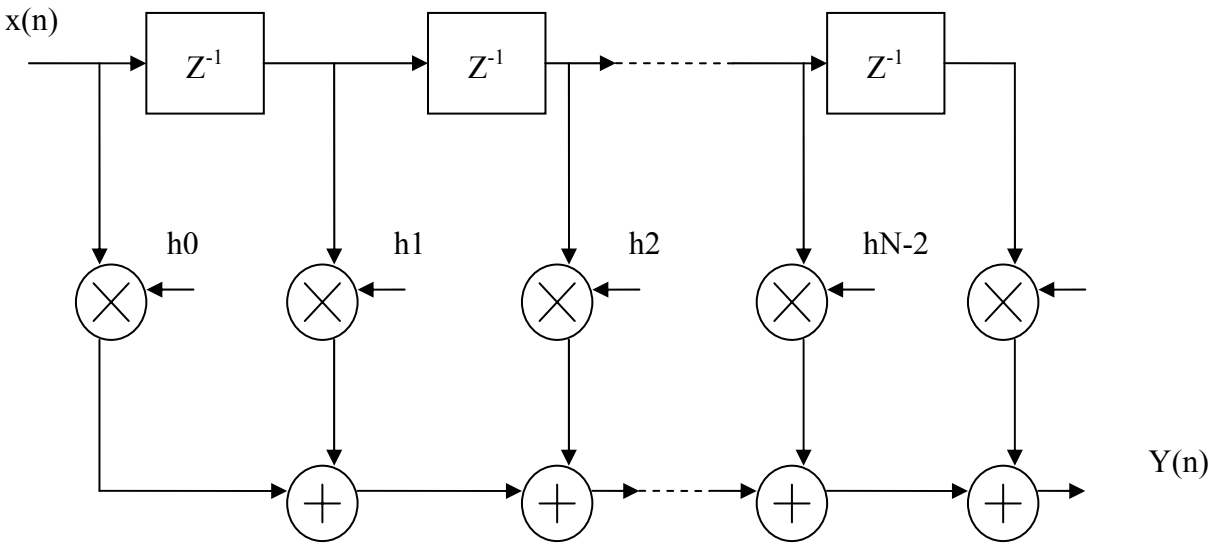


Fig 5.1 The logical structure of an FIR filter

An FIR filter works by multiplying an array of the most recent n data samples by an array of constants (called the tap coefficients), and summing the elements of the resulting array. (This operation is commonly called a dot product.) The filter then inputs another sample of data (which causes the oldest piece of data to be thrown away) and repeats the process. A finite impulse response (FIR) filter is a type of a digital filter. It is 'finite' because its response to an impulse ultimately settles to zero. This is in contrast to infinite impulse response filters which have internal feedback and may continue to respond indefinitely.

The basic terms involved in FIR filter can be explained as:

Filter Coefficients: The set of constants, also called tap weights, used to multiply against delayed sample values. For an FIR filter, the filter coefficients are, by definition, the impulse response of the filter.

• **Impulse Response:** A filter's time domain output sequence when the input is an impulse. An impulse is a single unity-valued sample followed and preceded by zero-valued samples. For an FIR filter the impulse response of a FIR filter is the set of filter coefficients.

• **Tap**: The number of FIR taps, typically N, tells us a couple things about the filter. Most importantly it tells us the amount of memory needed, the number of calculations required, and the amount of "filtering" that it can do. Basically, the more taps in a filter results in better stop band attenuation (less of the part we want filtered out), less rippling (less variations in the pass band), and steeper roll off (a shorter transition between the pass band and the stop band).

• **Multiply-Accumulate (MAC):** In the context of FIR Filters, a "MAC" is the operation of multiplying a coefficient by the corresponding delayed data sample and accumulating the result. There is usually one MAC per tap.

A Finite Impulse Response (FIR) digital filter is one whose impulse response is of finite duration. This can be stated mathematically as

$$h(m) = \begin{cases} 0, & m \le \tau 1 \quad -\infty < \tau 1 < \tau 2 < +\infty \\ 0 & m \ge \tau 2 \end{cases} \tag{5.2}$$

Where $h(m)$ denotes the impulse response of the digital filter, m is the discrete time index, and $\tau 1$ and $\tau 2$ are constants. A difference equation is the discrete time equivalent of a continuous time differential equation. The general difference equation for a FIR digital filter is,

$$y(n) = \sum_{m=0}^{N-1} h(m)\, x(n-m) \tag{5.3}$$

Where $y(n)$ the filter output at is discrete time instance m, $h(m)$ is the m^{th} feed forward tap, or filter coefficient, and $x(n-m)$ is the filter input delayed by m samples. The Σ denotes

summation from $m = 0$ to $m = N - 1$ where N is the number of feed forward taps in the FIR filter. Note that the FIR filter output depends only on the previous N inputs. here specialized to finite-length impulse responses. Since the time extension of the impulse response is N + 1 sample, we say that the FIR filter has length N + 1.This feature is why the impulse response for a FIR filter is finite. The transfer function is obtained as the Z transform of the impulse response and it is a polynomial in the powers of z^{-1}.The roots of this polynomial constitute the zeros of the filter.

$$\sum_{m=0}^{N-1} h(m)\, z^{-m} = h(0) + h(1)z^{-1} + ... + h(N)z^{-N} \qquad (5.4)$$

An FIR filter has linear phases if its units sample response satisfies the conditions

$$h(m) = \pm\, h(N - 1 - m) \quad m = 0, 1, ... N - 1 \qquad (5.5)$$

Since such polynomial has order N, we also say that the FIR filter has order N. when the input to a FIR filter is the Kronecker delta function δ(n), the impulse ripples through the tapped delay line of the filter, and the output at time m (for m = 0 to N-1) is the value of the m^{th} tap. (The function is defined as $\partial(n) = 1$ for n=0, and $\partial(n) = 0$ for $n \neq 0$).once the impulse passes through the tapped delay line, the output of the filter is zero. This is because the tapped delay is (and remains) filled with zeros.

5.1.1 Advantages of FIR filters

FIR filters are simple to design and they are guaranteed to be bounded input-bounded output (BIBO) stable. By designing the filter taps to be symmetrical about the center tap position, a FIR filter can be guaranteed to have linear phase. This is a desirable property for many applications such as music and video processing. FIR filters also have a low sensitivity to filter coefficient quantization errors.

5.1.2 Ideal lowpass filter

An ideal low pass filter passes all frequency components of a signal below a designated cutoff frequency, *wc* and rejects all frequency components of a signal above *wc* according to.

$$H_{LP}(e^{jw}) = \begin{cases} 1, & 0 \le w \le wc \\ 0, & wc < w \le \pi \end{cases} \tag{5.6}$$

The impulse response of an ideal low pass filter in Eq. 5.6 can be found from

$$h_{LP}[n] = \frac{\sin(w_c n)}{\pi n} \qquad -\infty < n < \infty \tag{5.7}$$

Because the impulse response required to implement the ideal lowpass filter is infinitely long, it is not possible to design an ideal FIR lowpass filter. Finite length approximations to the ideal impulse response lead to the presence of ripples in both the passband ($w < w_c$) and the stopband ($w > w_c$) of the filter, as well as to a nonzero transition width between the passband and stopband of the filter (see Figure 5.2).

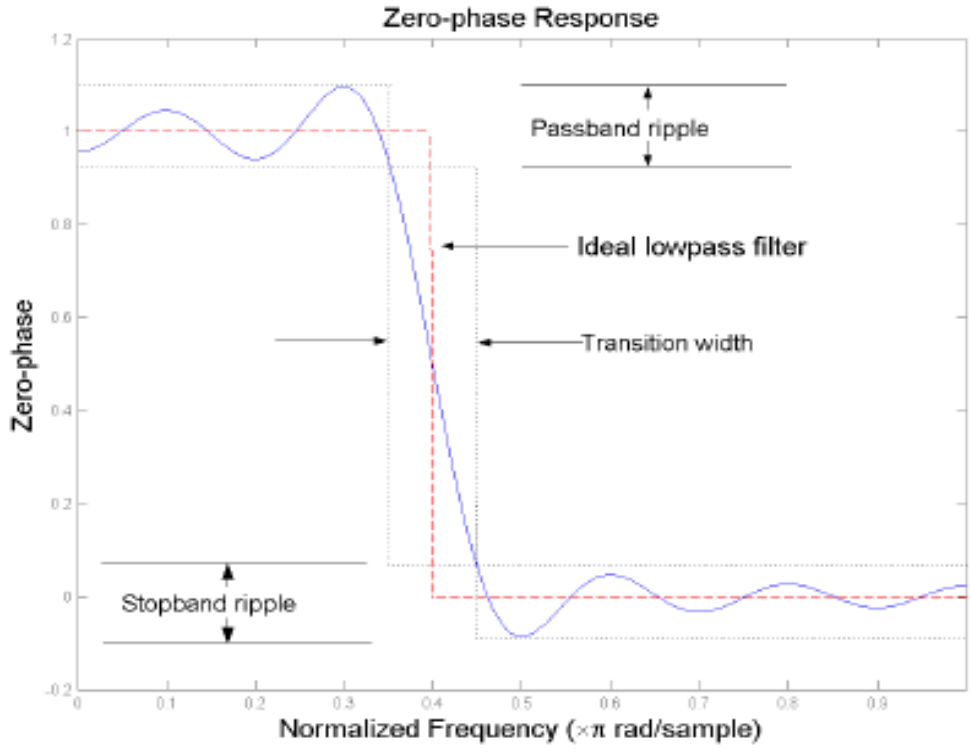


Fig 5.2: Illustration of the typical deviations from the ideal
Lowpass filter when approximating with an FIR filter, w*c* =0.4p.

5.1.3 FIR filter design specifications

Both the passband/stopband ripples and the transition width are undesirable but unavoidable deviations from the response of an ideal lowpass filter when approximating with a finite impulse response. Practical FIR designs typically consist of filters that meet certain design specifications, i.e., that have a transition width and maximum passband/stopband ripples that do not exceed allowable values.

In addition, one must select the filter order, or equivalently, the length of the truncated impulse response. A useful metaphor for the design specifications in FIR design is to think of each specification as one of the angles in a triangle as in Figure 5.3

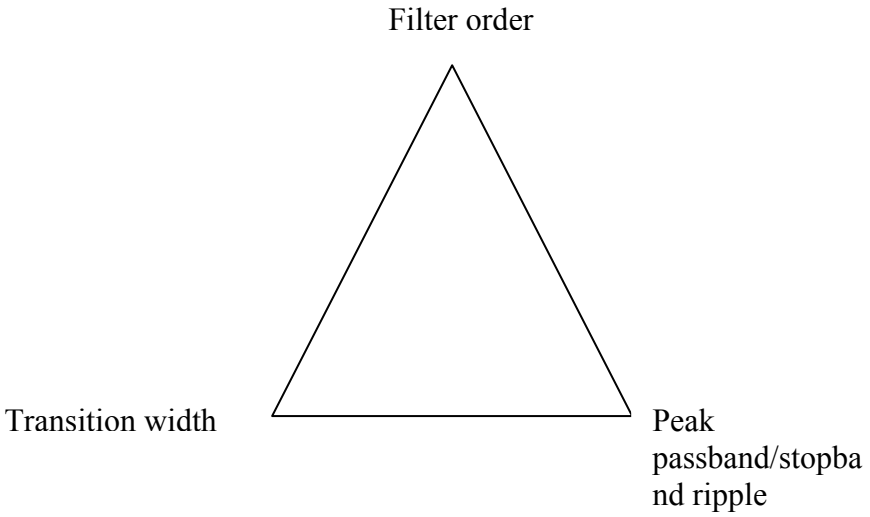


Figure 5.3: FIR design specifications represented as a triangle.

The metaphor is used to understand the degrees of freedom available when designating design specifications. Because the sum of the angles is fixed, one can at most select the values of two of the specifications. The third specification will be determined by the design algorithm utilized. Moreover, as with the angles in a triangle, if we make one of the specifications larger/smaller, it will impact one or both of the other specifications.

As an example, consider the design of an FIR filter that meets the following specifications:

1. Cutoff frequency: 0.4p rad/sample
2. Transition width: 0.06p rad/sample
3. Maximum passband/stopband ripple: 0.05

The filter can easily be designed with the truncated-and windowed impulse response algorithm implemented in fir1 (or using fdatool) if we use a Kaiser window. The zero-phase response of the filter is shown in Figure 5.4. Since we have fixed the allowable width and peak ripples, the order is determined for us. Close examination at the passband-edge frequency, $w_p = 0.37\pi$ and at the stopband-edge frequency $w_s = 0.43\pi$ shows that the peak passband/stopband ripples are indeed within the allowable specifications. Usually the specifications are exceeded because the order is rounded to the next integer greater than the actual value required. The pass band-edge frequency is the boundary between the passband and the transition band. If the transition width is TW, the passband edge frequency w*p* is given in terms of the cutoff frequency wc by $wp = wc - \frac{TW}{2}$ .Similarly, the stopband-edge frequency is given by $ws = wc + \frac{TW}{2}$

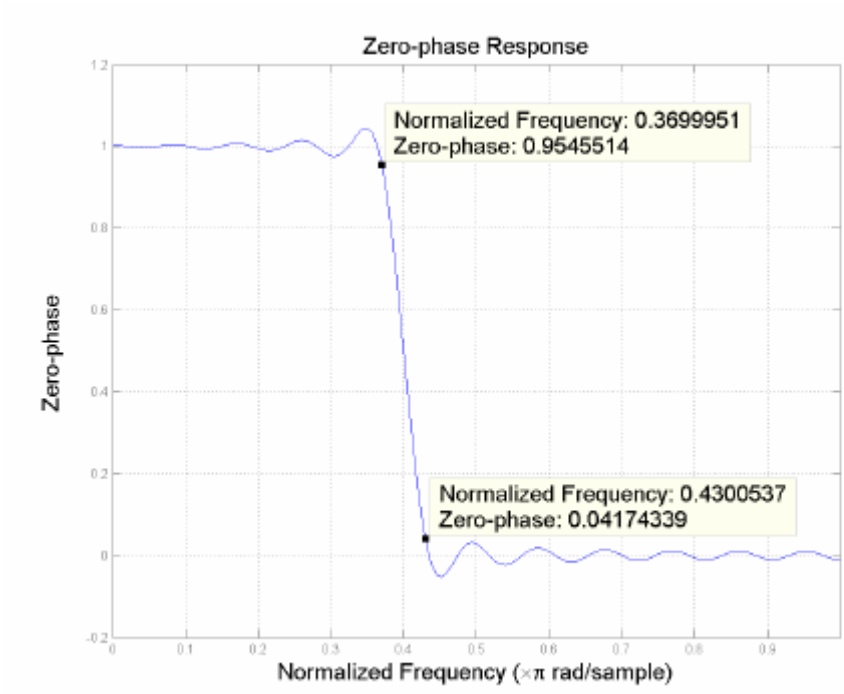


Figure 5.4: Kaiser Window design meeting predescribed specifications.

5.2 HALF-BAND FIR FILTER DESIGN

5. 2.1 INTRODUCTION

By considering the application requirements, FIR and IIR structures can be used to meet the design specifications. FIR filters offer great control over filter shaping and linear phase performance with waveform retention over the pass band. Due to its all-zero structure, the FIR filter has a linear phase response necessary for audio application, but at the expense of the high filter order. IIR filters, however, can be designed with much smaller orders than FIR filters at the expense of the nonlinear phase. Since it is very difficult to design a linear phase IIR filter, we have designed a half-band FIR filter for the same application. In half-band filters, the number of taps is reduced considerably since the odd coefficients are zeros, which reduces the hardware and the power consumption. As a result, we use half-band filters instead of the direct conventional FIR Filters.

Definition of half-band filter

The definition of half-band filters was introduced for FIR filters and is based on the symmetry in the pass-band and the stop-band. half band filter is shown in Figure 5.5 If we denote by Fa and Fp the normalized boundary frequencies of the pass-band and the stop-band, for half-band filters it must be,

$$F_a = \frac{1}{2} - F_p \tag{5.8}$$

The normalization being made with respect to the sampling frequency. The approximation error for the magnitude response | H (e^{jw}) | in the pass-band and the stop-band is made equal.

$$\delta p = \delta a \tag{5.9}$$

It should be noticed that the ripple in the pass-band amounts to 2δ , i.e. according to the definition for FIR filters it is two times greater than that in the stop-band. In the middle of the band, the magnitude response reduces to a half:

$$\left| H_{FIR}(e^{\frac{j\pi}{2}}) \right| = \frac{1}{2} \qquad (5.10)$$

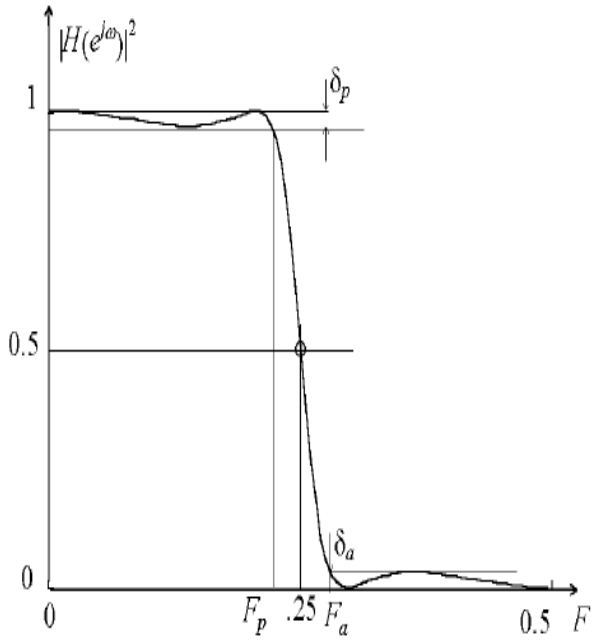


Fig 5.5Magnitude response of a Half Band Filter

It is known that the FIR filter could be realized in such a manner to have a linear phase characteristic. It is also known that the analogue elliptic filter of minimal Q-factors has the frequency response equal to the ripple in the pass-band and the stop-band. On the basis of these statements it is easy to conclude that by a bilinear transformation of elliptic minimal Q-factors analogue prototype, with the corresponding adjustment of frequency scales a half band FIR filter could be obtained, which will correspond fully to already established definition; the only difference would be that the ripple of the magnitude response in the pass-band and the stop-band will be equal.

5.2.2 Implementation of Half band filter

Half band filters are one type of the symmetric FIR filters. All of their odd coefficients are equal to zero except for the center one, which is equal to $\frac{1}{2}$.This architecture results in fewer taps, less hardware, and lower power. Half band filters are constrained to be equiripple filters

with the property that their frequency response has a value of 0.5 at the frequency fs/4, where fs is the sampling rate. Because of this restriction, they are not suitable for decimation of more orders. An idealized half band filter magnitude response is shown in Figure 5.7, in half-band filters, the number of taps is reduced considerably since the odd coefficients are zeros, which reduces the hardware and the power consumption. As a result, we use half-band filters instead of the direct conventional FIR filters. For the design of half-band filter with impulse response *h(n)*, we used the Remez algorithm to determine filter coefficients, as in (5.11).

$$h(n) = \begin{cases} 0 \;\; if \quad n = \dfrac{N-1}{2} \\ 1 \qquad otherwise \end{cases} \tag{5.11}$$

In the hope that it will be easier to follow, we shall describe the design in terms of a filter of length 11, involving an equation $n = 5 = \frac{(11-1)}{2}$ of degree, writing the equations out in full. The extension to the case for general, where n may be any positive odd integer, Should be obvious.

The number of taps in an FIR filter is directly proportional to the stop band rejection, the ratio of the sampling frequency and the transition band. We design a half-band FIR filter, which has a transition band of about 15 KHz with the input sampling frequency of 80 KHz, a pass band of 20 KHz, a cutoff frequency of 35 KHz, and a stop band attenuation of -65 dB. We selected a pass band ripple of 0.001 to obtain a maximally flat response. The designed FIR filter has 11coefficients and is implemented using the transpose direct form (TDF) as shown in Figure 5.6, to reduce the hardware due to the symmetric characteristics of its coefficients.

Half band filters have two important characteristics

1. The pass band and stop band ripples must be the same.
2. The pass band edge and stop band edge frequencies are equidistant from the half-band frequency pi/2.

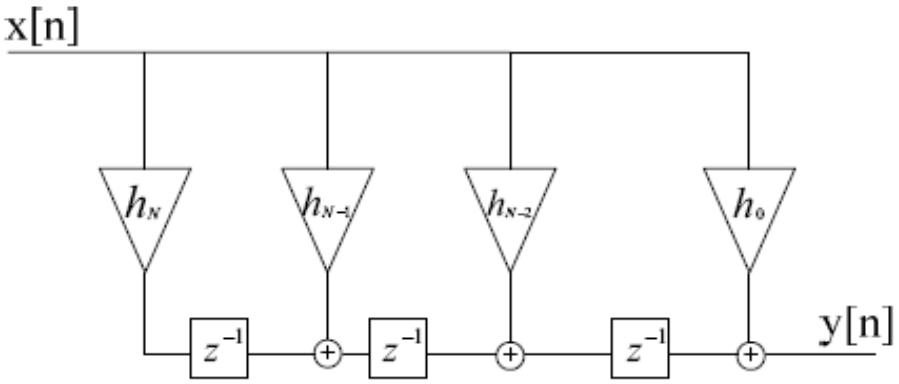


Fig 5.6: The FIR Transpose direct form

As I said above that the half-band filters contains very less number of taps, almost reduced to 50% since the odd coefficients are zeros. This considerably reduces the hardware and the power consumption. As a result, by the use half-band FIR filters instead of direct conventional FIR filters we can save some power and area. The designed FIR filter has 11 coefficients and is implemented using the transpose direct form (TDF) as shown in Fig 5.6. Fig 5.6 reduces the hardware due to the symmetric characteristics of its coefficients. computational complexity since only half the number of coefficients is used to implement the decimation filter This transpose direct form (TDF) implementation reduces the power consumption by approximately one-half because the filter operates at the decimated rate instead of the input rate.

5.3 CORRECTOR FILTER DESIGN

The last stage of our decimation filter chain needs a compensation for the passband droop caused by the CIC decimation that has a sinc-like response. an additional FIR filter is designed to push out of band undesired signals. The FIR filter is used in the last stage instead of a shaping filter for less power consumption because a shaping filter has more taps than an FIR filter. The multi-rate multi-stage implementation reduces the sampling frequency, so the direct form is used. The direct form implementation is based on the use of symmetric coefficients of the FIR filters. as a result, only half the number of coefficients is used to implement the low power decimation filter. The designed FIR filter has a pass band of 4 KHz and a stop band of 15 KHz with a sampling frequency of 40 KHz. The human ear is very sensitive to all sounds within 4 KHz, due to which the designed filter has a pass band of 4

KHz. The output of the corrector filter has a sampling frequency of 20 KHz, which is ideal for the operation of the following signal processing stages.

An FIR with constant coefficient is a linear time invariant (LTI) filter digital filter. The output of a FIR of the order or length L, to an input time-series x [n], is given by a finite version of convolution sum given by:

$$y = x[n] * f[n] = \sum_{k=0}^{L-1} f[k]\, x[n-k] \qquad (5.12)$$

Where $f(0) \neq 0$ through $f(L-1) \neq 0$ are filter's L coefficients.

In z-domain the above equation is expressed as

$$Y(z) = F(z)\, X(z) \qquad (5.13)$$

Where F (z) is the FIR's transfer function defined in the z-domain by

$$F(z) = \sum_{k=0}^{L-1} f[k]\, z^{-k} \qquad (5.14)$$

The Lth order FIR filter consists of a group of tapped delay lines, adders and multipliers. One of the operands of the multipliers is called the "tap weight" or the filter coefficients. The roots of the polynomial F (z) define the zeros of the filter. Due to the presence of only *zeros,* the FIR filters are also referred to as all zero filters.

A variation of the direct FIR filter structure is the transposed structur*e*. A direct form FIR filter can be converted into transposed form by simply:

- Exchanging the input and the output.
- Inverting the direction of signal flow
- Substituting an adder by a fork, and vice versa.

The advantage of the transposed structure is that we don't need an extra shift register and there is no need for an extra pipeline stage for the adder (tree) of the products to achieve high throughput.

Chapter 6

DISTRIBUTED ARITHMETIC

6.1 DESCRIPTION OF DISTRIBUTED ARITHMETIC

Basically, distributed arithmetic uses precalculated partial products that are stored in a look-up table (LUT) to efficiently compute inner-products. Let *x* be a data vector of length *N* that is multiplied by a constant coefficient vector *a* of length *N,* and *y* be the result.

$$y = \sum_{i=1}^{N} x_i a_i \tag{6.1}$$

When two's-complement representations are used for representing the signals x_i , the inner-product can be written as shown in Esq. (6.2). Here W_d is the word length of x_i .

$$y = \sum_{i=1}^{N} a_i \left(\sum_{j=1}^{Wd-1} x_{ij} 2^{-j} - x_{i0} \right) \tag{6.2}$$

By interchanging the summation order, eqn.(6.3) is obtained. The summation is first made over the terms between bit *j* of x_i and a_i then over the bits *j* and finally, a subtraction of the partial bit-products involving the sign bits x_{i0} is performed.

$$y = \sum_{j=1}^{Wd-1} 2^{-j} \sum_{i=1}^{N} a_i x_{ij} - \sum_{i=1}^{N} a_i x_{i0} \tag{6.3}$$

Defining F_j as a function of bit *j* and of each x_i

$$F_j = a_1 x_{1j} + a_2 x_{2j} + ... + a_N x_{Nj} \tag{6.4}$$

We obtain

$$y = \sum_{j=1}^{Wd-1} F_j 2^{-j} - F_0 \tag{6.5}$$

The function F_i can only take 2^N different values that can be precalculated and stored in a look-up table. Bit j of each data A_1 is used to address this lookup table. By rewriting Eq. (6.4) in Horner's form, we obtain

$$y = [...(0 + F_{wd-1})2^{-1} + F_{wd-2}1)2^{-1} + ... + F_1]2^{-1} + (-F_0) \tag{6.6}$$

Eq. (6.6) shows that only three different operations are required for calculating the inner-product. First, a table look-up to obtain the value of F_j , then addition or subtraction, and finally a division by two that can be realized by a right-shift when two's complement representation is used. an addition

6.1.1 Architecture for Distributed Arithmetic

A block diagram of a distributed arithmetic processor (DAP) is shown in Fig. 6.1. The DAP consists of a look-up table (LUT) with decoder, and a shift accumulator. Using bit-serial input data with the least significant bit (lbs.) first of word length W_d , one bit of each the *N* inputs are used to address to the lookup table. This looked-up value F_j of width w_c is added to the accumulated value, divided by two, i.e., right-shifted one position, and stored. After each shift one output bit y_{lsb} is generated in serial form. This is repeated until the sign-bit of the x_i : s arrives, and then a subtraction is performed. The most significant part y_{msb} is generated in parallel form.

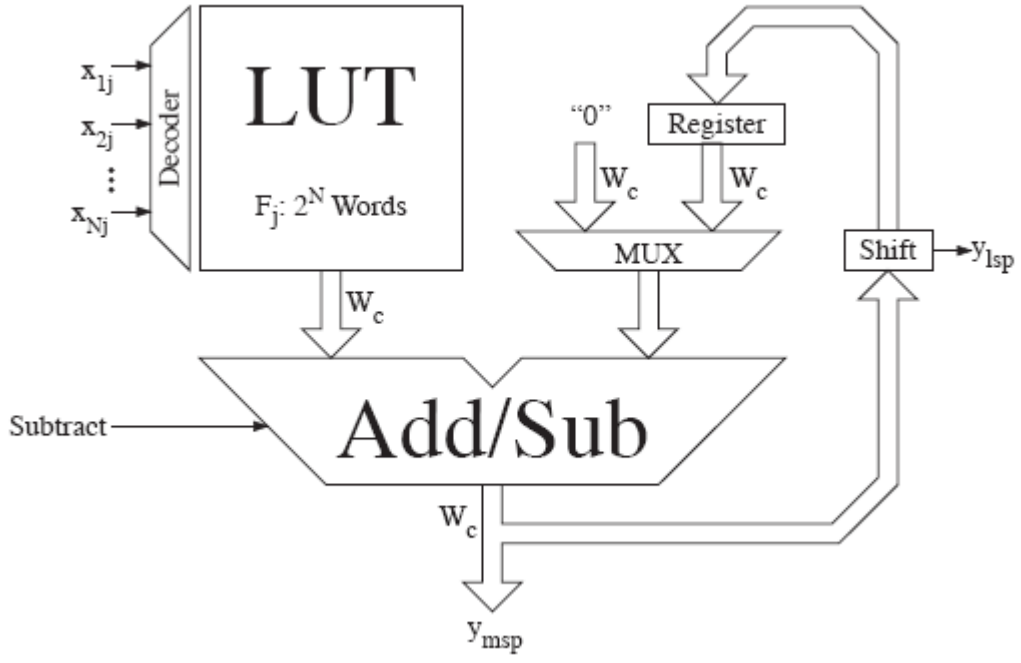


Figure 6.1 Distributed arithmetic processor.

6.1.2 Reduction of the look-up-Table

There exist a number of variations on distributed arithmetic that aims at reducing the size of the look-up table. The method found in is of particular interest. This method is based on a modified two's-complement representation of the coefficients. This recoding of the coefficients is denoted by Smith as Offset Binary Coding (OBC), which reduces the look-up table by a factor of two. The offset binary coding can be extended further, reducing the look-up table in steps by factor of two from 2^N down to N. However, this requires additional hardware in terms of adders and multiplexers. Thus, the latency is increased. The inner-product in Eq. (6.1) can be split in L parts with $\frac{N}{L}$ terms in each partial inner-product. For the final summation of the L parts, an adder tree is required and the latency is thus increased also for this method

6.1.3 Offset Binary Coding

Using an equivalent expression for the number x_i

$$x_i = \frac{1}{2}\left[x_i - (-x_i) \right] \tag{6.7}$$

The two's complement representation of x_i with word length W_d can be expressed as

$$x_i = \hat{x}_{i0} 2^{-1} + \sum_{j=1}^{w_d - 1} \hat{x}_{ij} 2^{-(j+1)} - 2^{-W_d} \tag{6.8}$$

$$\hat{x}_{ij} = x_{ij} - \bar{x}_{ij}$$

$$\hat{x}_{ij} = x_{ij} - \bar{x}_{ij} \tag{6.9}$$

Where $\bar{x}_{ij}$ is the complement of the bit x_{ij}, i.e., a bit inversion? Here the x_{ij} can only take the values and -1 instead of 0 and 1 as for normal two's-complement coding. Now Eq. (6.3) can be rewritten

$$y = \sum_{j=1}^{Wd-1} 2^{-(j+1)} \sum_{i=1}^{N} a_i \hat{x}_{ij} - \sum_{I=1}^{N} a_i \hat{x}_{i0} 2^{-1} + \sum_{i=0}^{N} a_i 2^{-Wd} \tag{6.10}$$

Defining $F\hat{}_J$ for the offset binary coding in a similar way as F_J was defined; Eq. (6.10) now can be rewritten as

$$y = 2^{-1}\left(\sum_{i=0}^{W_d-1} 2^{-j} F\hat{}_J - F\hat{}_0 + A_0 2^{-(W_d-1)}\right) \tag{6.11}$$

Where A_0 is equal to

$$A_0 = \sum_{i=1}^{N} a_i \tag{6.12}$$

FIJ still can take only 2^N values but only 2^{N-1} different magnitude values. Consequently the look-up table can be reduced to contain only half the number of words.

Compared to Eq. (6.4), a factor 2^{-1} and an offset A_0 are introduced; hereof the name offset binary coding. The look-up table is anti-symmetric around the middle of the table, i.e., the entries are pair wise equal in magnitude but with different signs. The factor 2^{-1} in the Eq. (6.11) can either be merged with the F_J :s or computed with a shift operation. If the factor 2^{-1} is computed with a shift operation and not included in the $F\hat{}_J$:s, and A_0 the relationship between F_J and $F\hat{}_J$ is given by Eq. (6.13) below.

$$\hat{}F\hat{}_j = 2(F_j - \frac{1}{2}A_0) = 2F_j - A_0 \tag{6.13}$$

Example.

In Table 6.1 an example of a look-up table for an inner-product N=3 with is shown. It is obvious that the look-up table is anti-symmetric around the middle of the table. Hence, the look-up table can be reduced to the half by letting one term, in this case x_{3j} control the sign of the precalculated look-up entry.

x_{3j}	x_{2j}	x_{1j}	F_j	$\hat{F}_j$
0	0	0	0	$-(a_1+a_2+a_3)$
0	0	1	a_1	$-(-a_1+a_2+a_3)$
0	1	0	a_2	$-(a_1-a_2+a_3)$
0	1	1	a_1+a_2	$-(-a_1-a_2+a_3)$
1	0	0	a_3	$(-a_1-a_2+a_3)$
1	0	1	a_1+a_3	$(a_1-a_2+a_3)$
1	1	0	a_2+a_3	$(-a_1+a_2+a_3)$
1	1	1	$a_1+a_2+a_3$	$(a_1+a_2+a_3)$

Table 6.1 Look-up table for N=3.

6.1.4 Architecture for Distributed Arithmetic using offset binary coding

The architecture in Fig. 6.1 can be modified for the use of offset binary coding as shown in Fig. 6.2. Here, x_{3j} the control the sign of the look-up table and the inverting of the other x_{ij} : s. Instead of clearing the shift-accumulator at the beginning of each computation, the offset is A0 loaded. This is possible due to the data independence A_0 of. The overhead cost for the offset binary coding is small compared to the gain obtained from the reduction in the size of the look-up table.

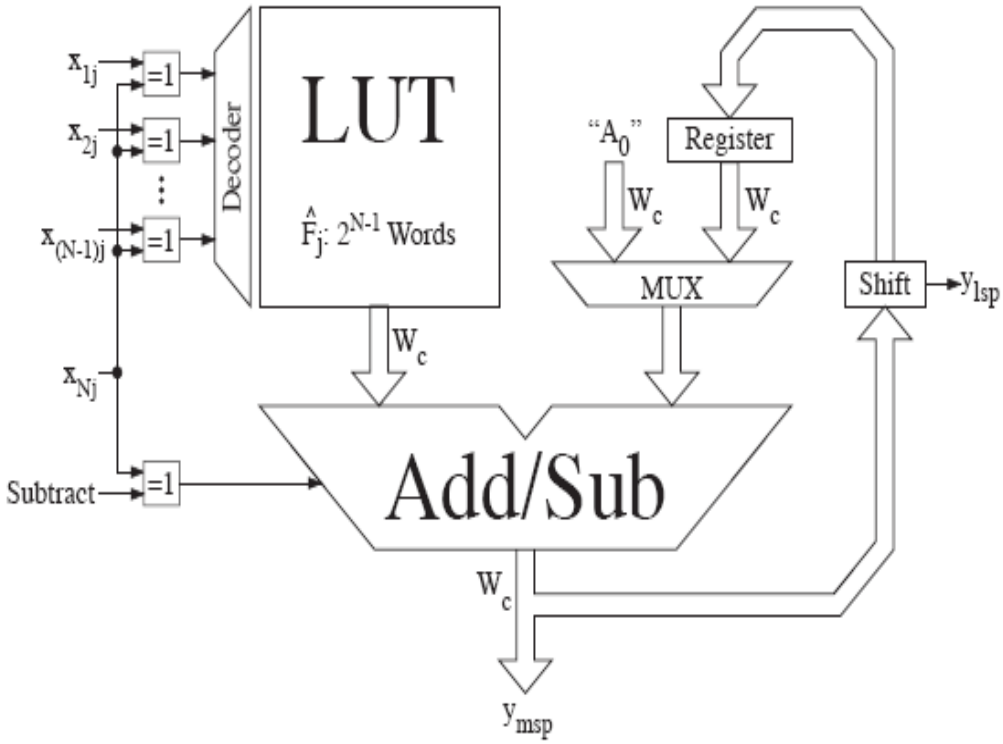


Figure 6.2 Distributed arithmetic processor for offset binary coding.

Chapter 7

SIMULATION RESULTS

7.1 MATLAB RESULTS.

7.1.1 The sigma-delta analog to digital conversion

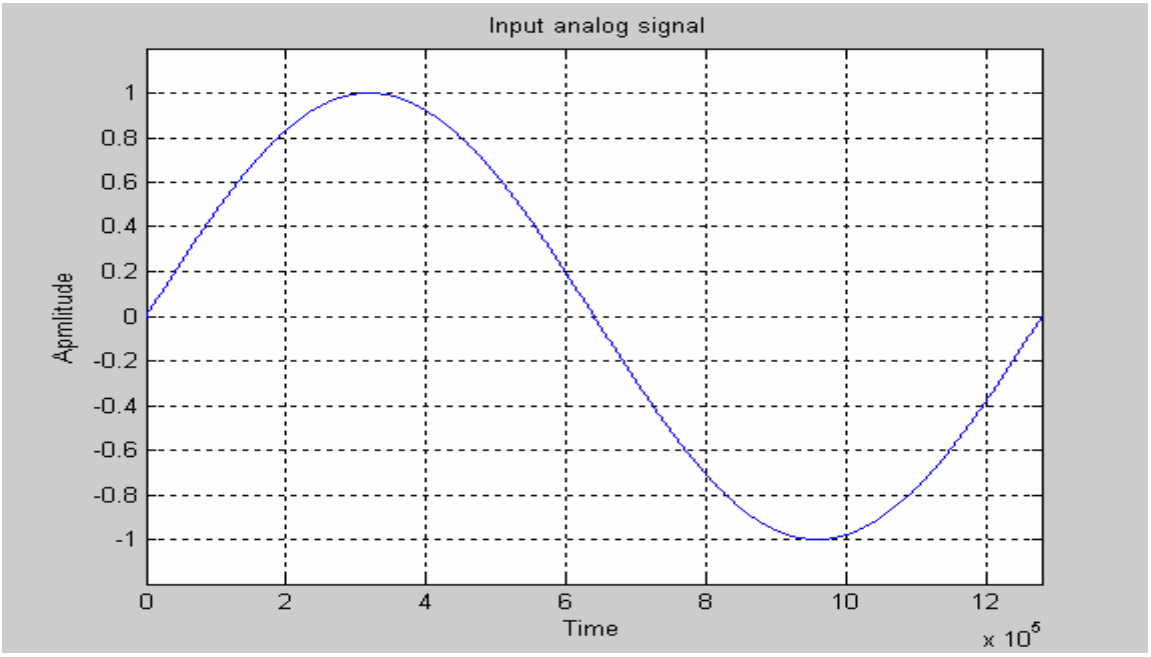


Fig 7.1(a) input waveforms of the sigma-delta quantizer for a sinusoidal input frequency is 1.28MHz

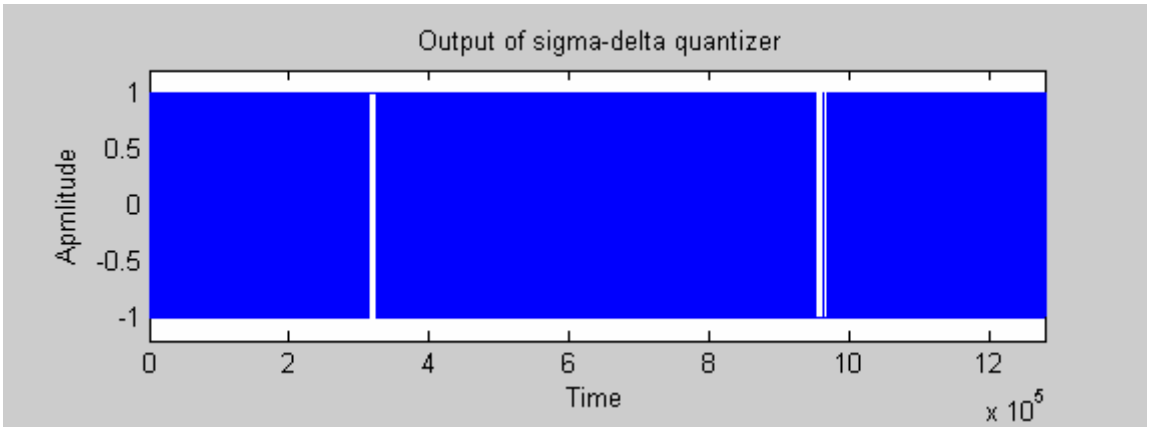


Fig 7.1 (b) The output waveforms of the sigma-delta quantizer

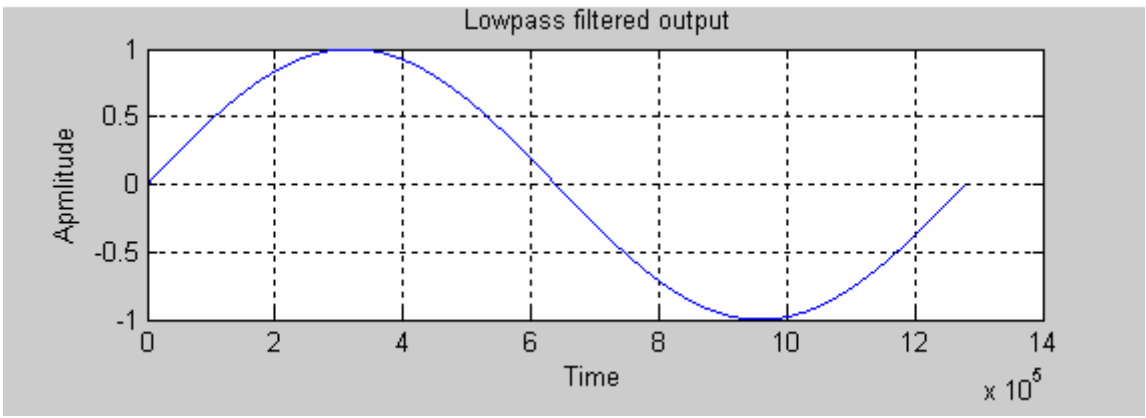


Fig 7.1 (c) The lowpass filtered version of the waveform of figure Fig 7.1 (b)

7.1.2 The sigma-delta digital to analog conversion

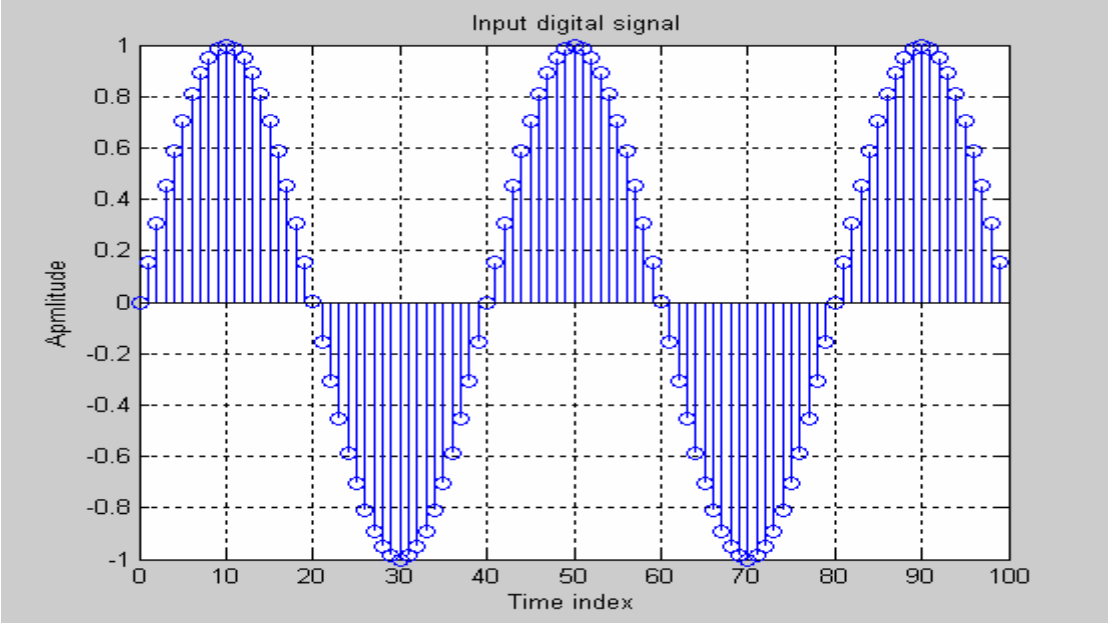


Fig 7.2 (a) input waveforms of the sigma-delta quantizer for a sinusoidal input frequency is 4 kHz

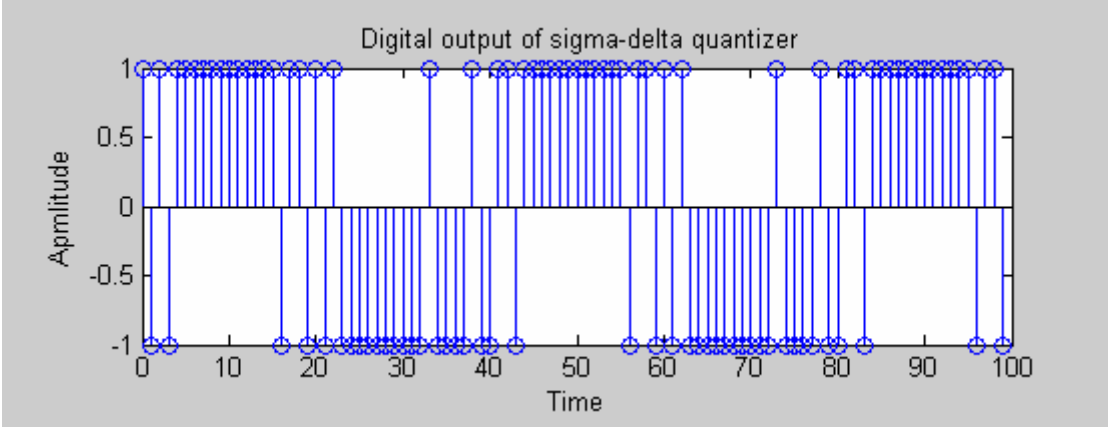


Fig 7.2 (b) The output waveforms of the sigma-delta quantizer

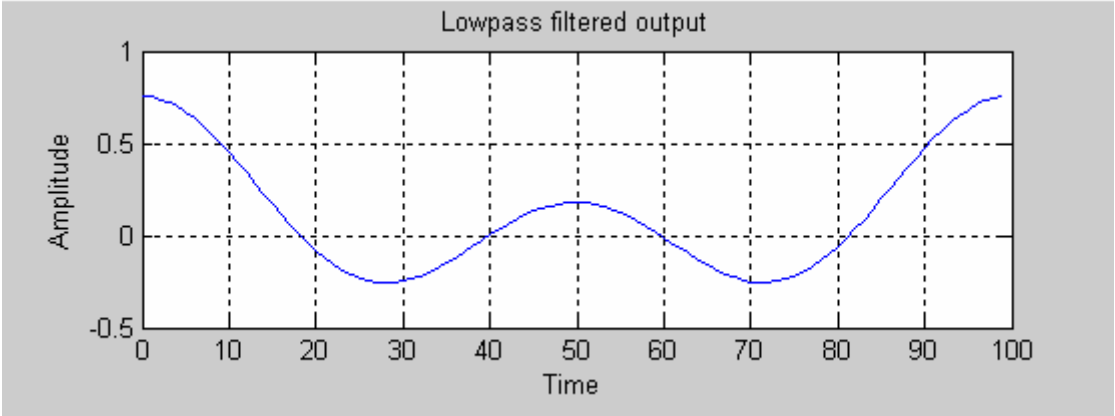


Fig 7.2 (c) The lowpass filtered version of the waveform of figure fig 7.2 (b)

7.1.3 Frequency response of the 5-stage Comb filter

The frequency response of 5-stage Cascaded Integrated comb (CIC) Filter is shown in Fig.7.1: The specifications of our design comb filter are, the input of the sampling frequency (Fs) is 1.28 MHz, and the comb filter stages have a sinc^6 (f) frequency response with decimation factor of M=16. And the comb section consists of K=5 stages, with a differential delay of *D =1*. The comb filter provides attenuation of about -65dB as in shown in Fig 7.2.

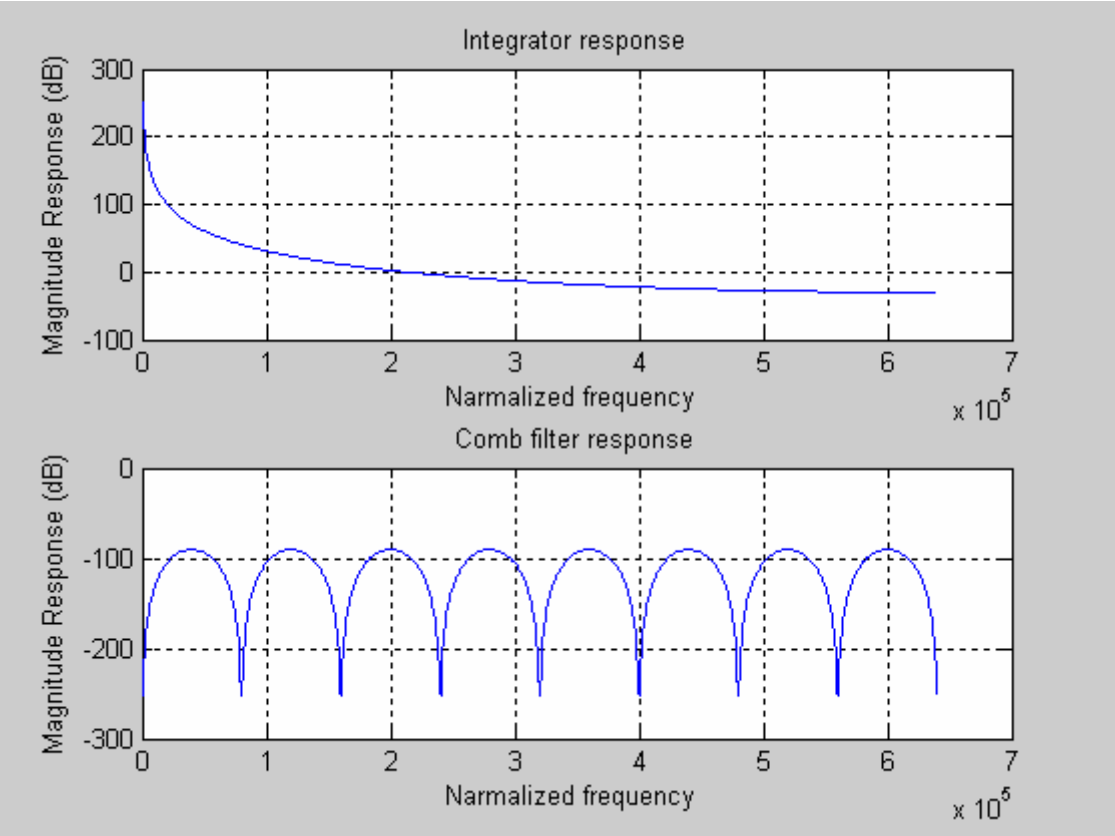


Fig.7.3 Frequency response of 5-stage Cascaded Integrated comb (CIC) Decimation: Upper fig shows Integrator section response and lower fig shows the comb section

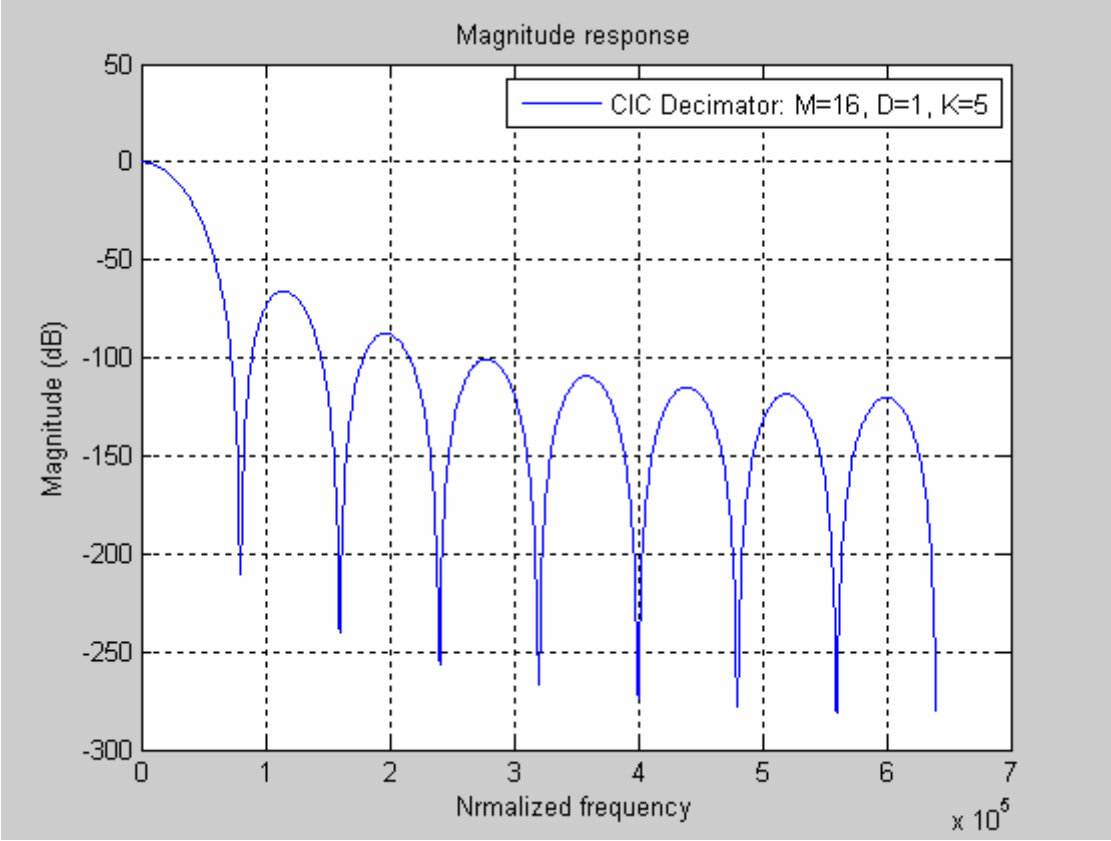


Fig 7.4 Attenuation for a 5-stage comb filter

The comb filter decimates from 1.28 MHz to 80 kHz due to a decimation factor of M=16. Comb filter design present two drawbacks: an insufficient attenuation in stop band and distortion in pass band. A corrector filter can compensate distortion in pass band

7.1.4 Frequency response of t he low pass FIR filter

The number of taps in an FIR filter is directly proportional to the stop band rejection, the ratio of the sampling frequency and the transition band, The specifications of our design low pass FIR are, The filter order is N = 22, The transition band about 15 kHz with the input sampling frequency of 80 kHz, a pass band of 20 kHz, a cutoff frequency of 35 kHz, and stop band attenuation of -65dB and pass band ripple is 0.001. The magnitude response and impulse response of low pass FIR filter is shown in Fig 7.5

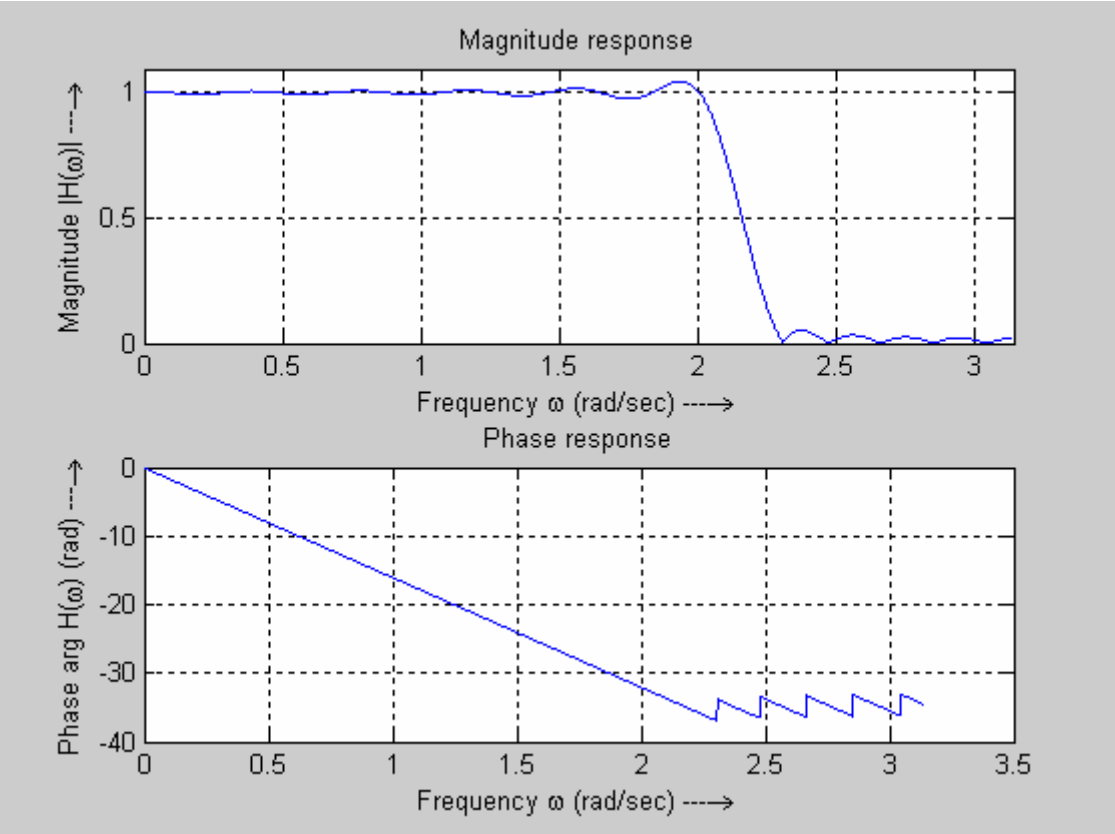


Fig.7.5 Magnitude response of a low pass FIR filter

7.1.5 Magnitude response of the Half-band FIR and the Correction FIR filters

The number of taps in an FIR filter is directly proportional to the stop band rejection, the ratio of the sampling frequency and the transition band, The specifications of our design half-band FIR filter are, The filter order is N = 11 the transition band about 15 kHz with the input sampling frequency of 80 kHz, a pass band of 20 kHz, a cutoff frequency of 35 kHz, and stop band attenuation of -65dB and pass band ripple is 0.001.The magnitude, impulse and pass band equiripple response of half-band filter is shown in Fig 7.6.

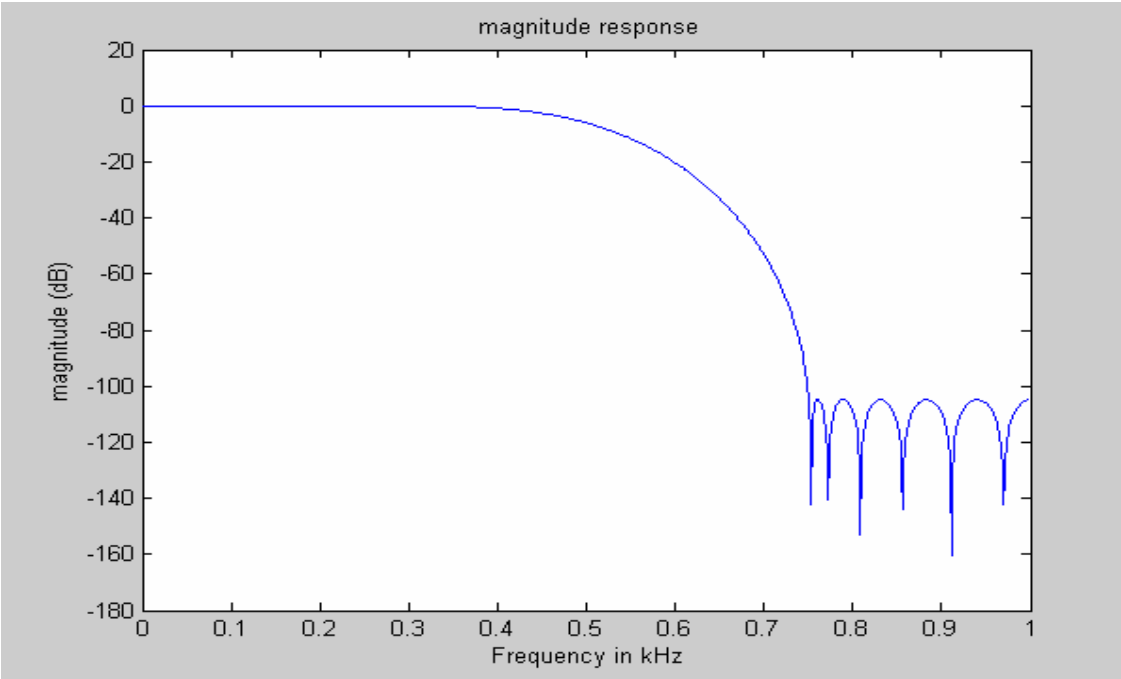
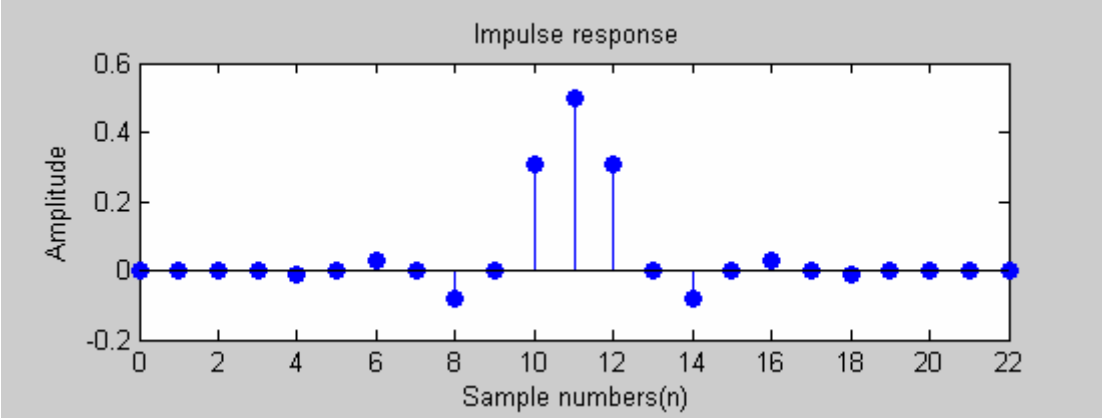
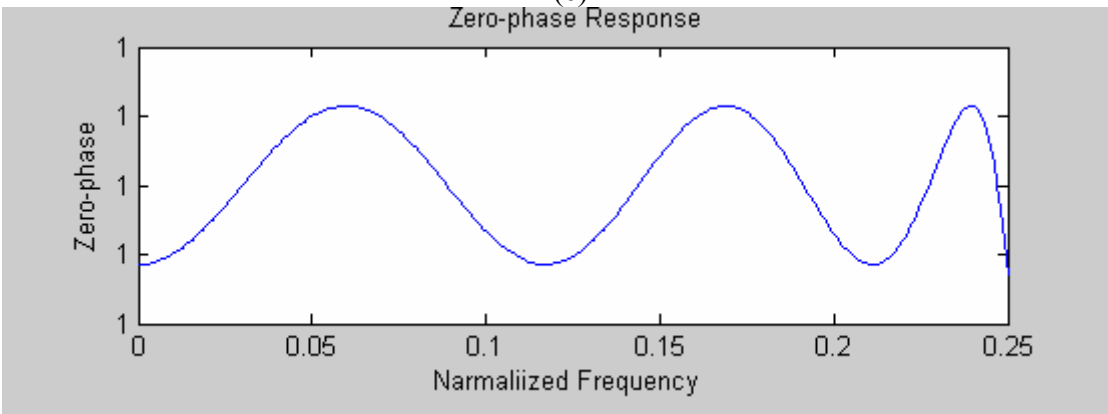


Fig 7.6 :(a) Magnitude response of half-band FIR filter



(b) Impulse response

(c)



(c) Equiripple in pass band

The third stage of the proposed decimation filter is Corrector FIR filter. In the first stage, the design of comb section presents two drawbacks: an insufficient attenuation in stop band and distortion in pass band. Corrector filter can compensate distortion in pass band.

The designed FIR filter has a pass band of 4 KHz and a stop band of 15 KHz with a sampling frequency of 40 KHz, pass band of 4 KHz. The output of the corrector filter has a sampling frequency of 20 kHz; the magnitude response of corrector filter is shown in Fig 7.7.

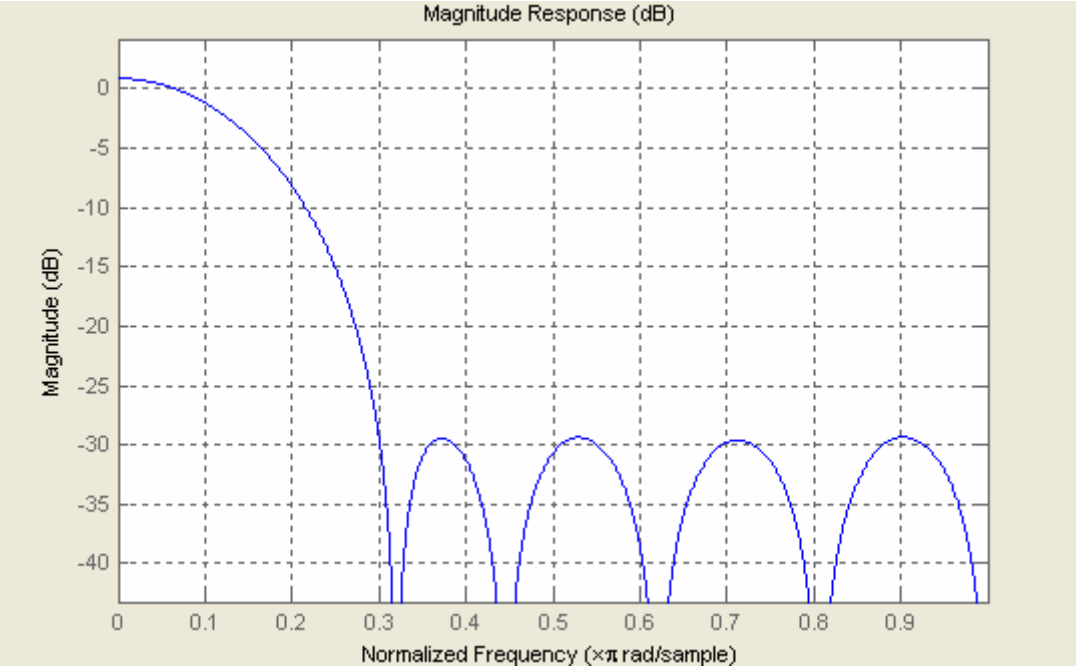


Fig 7.7 the magnitude response of the last stage corrector FIR filter

7.2 SIMULATION RESULTS IN MATLAB SIMULINC USING DSP BLOCKSET AND XILINX BLOCKSET

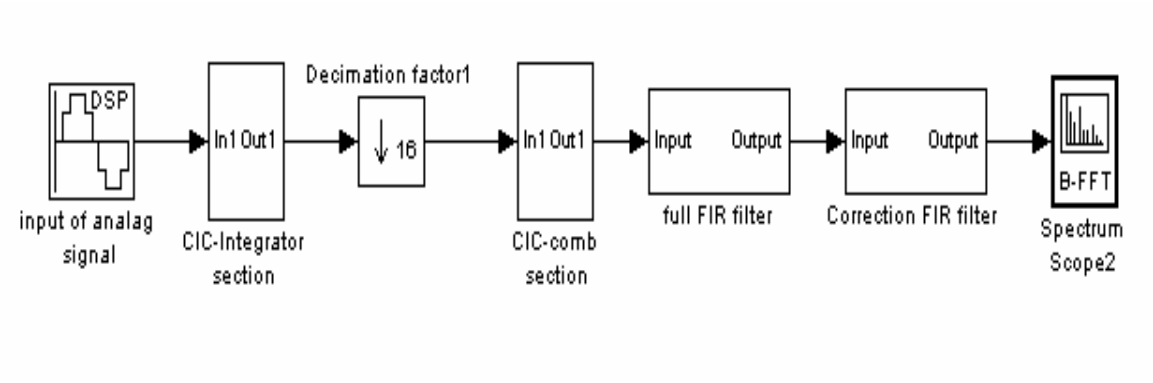


Fig 7.8: (a) Implementation the 5-stage comb-FIR-FIR Decimation filter using DSP Block set.

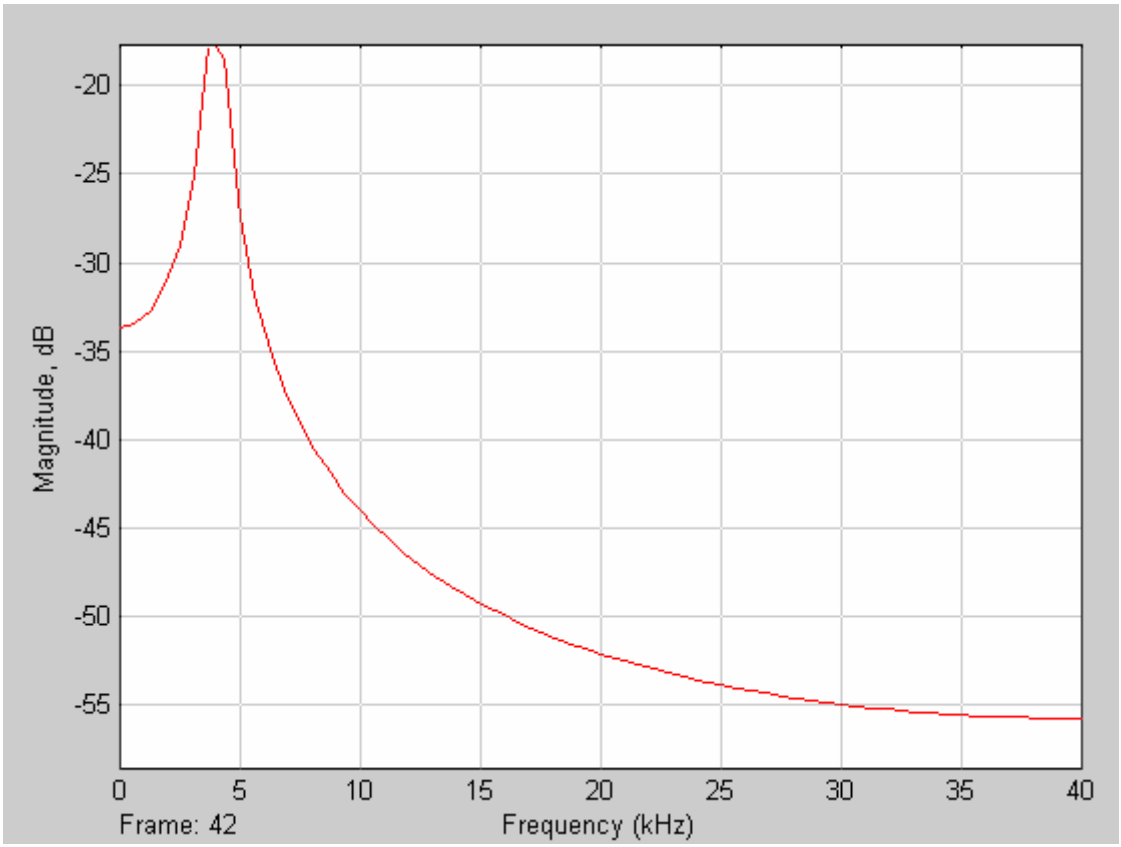


Fig 7.8: (b) The output response of the comb-FIR-FIR filter

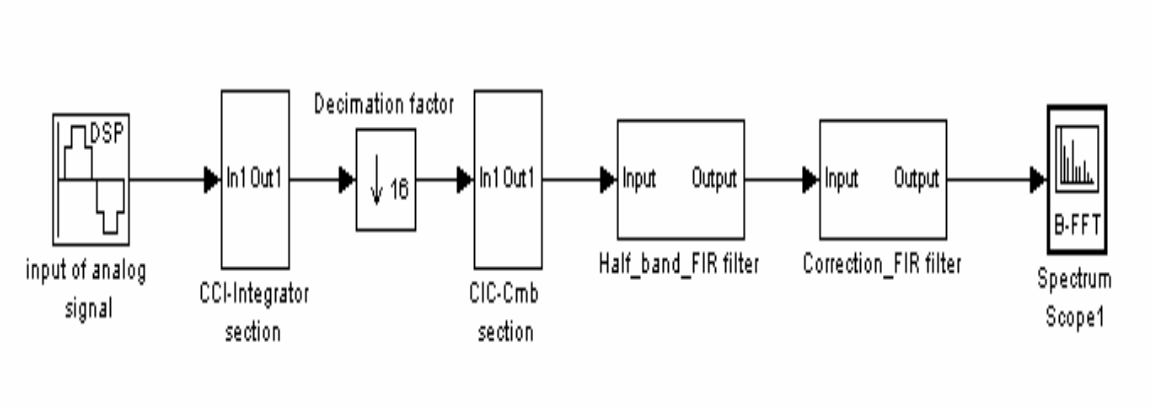


Fig 7.9: (a) Implementation the 5-stage comb-half-band-FIR-FIR Decimation filter using DSP Block set.

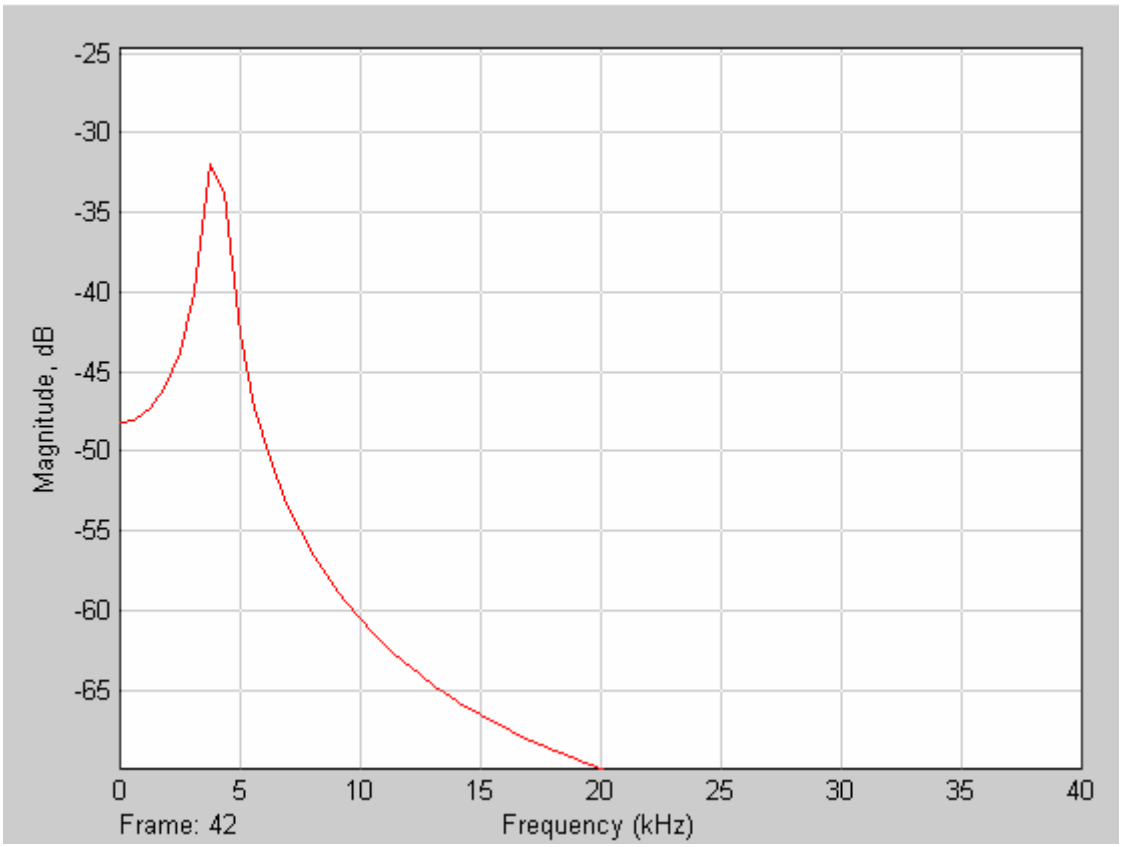
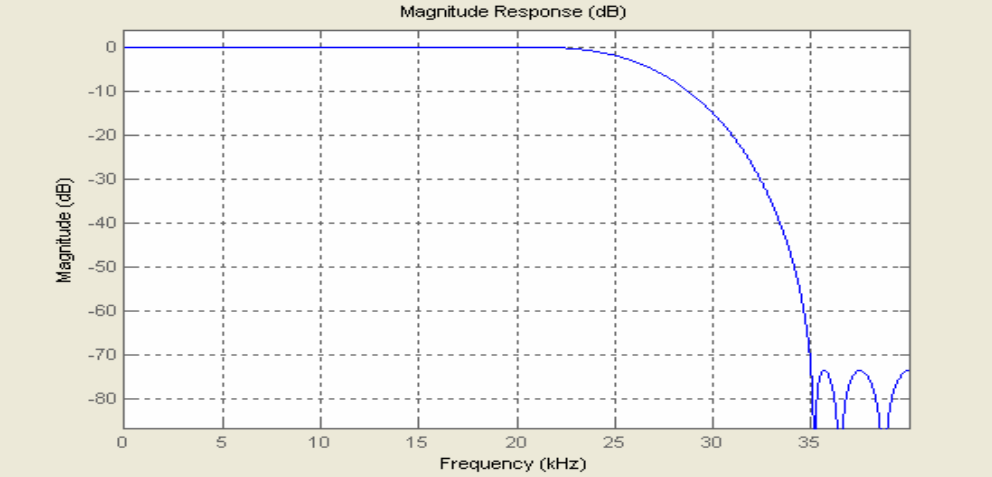
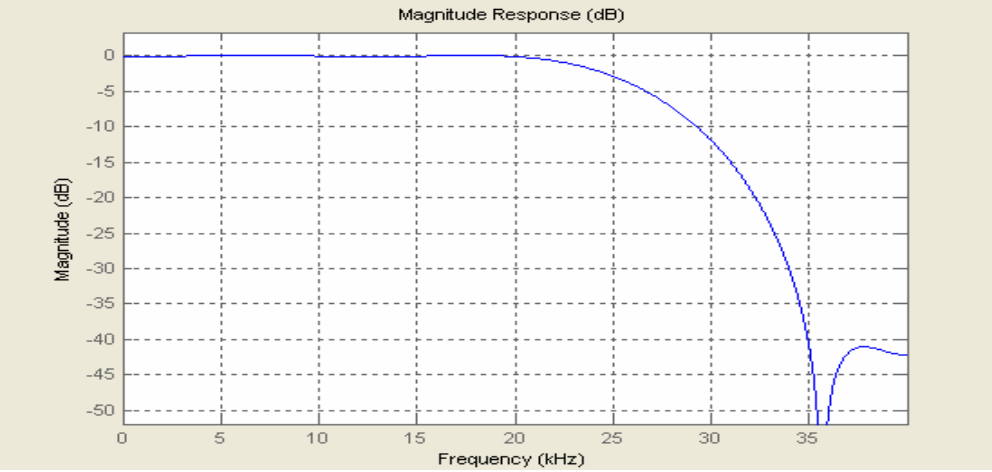


Fig 7.9: (b) The output response of the comb-half-band FIR-FIR

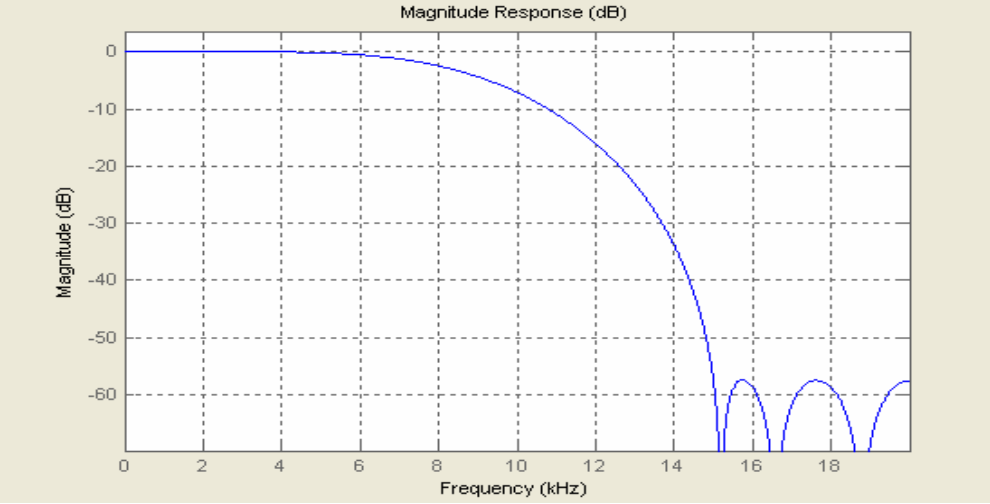
The figure shows the output of designed filter has a pass band of 4 kHz and a sampling frequency of 20 kHz.



(a)

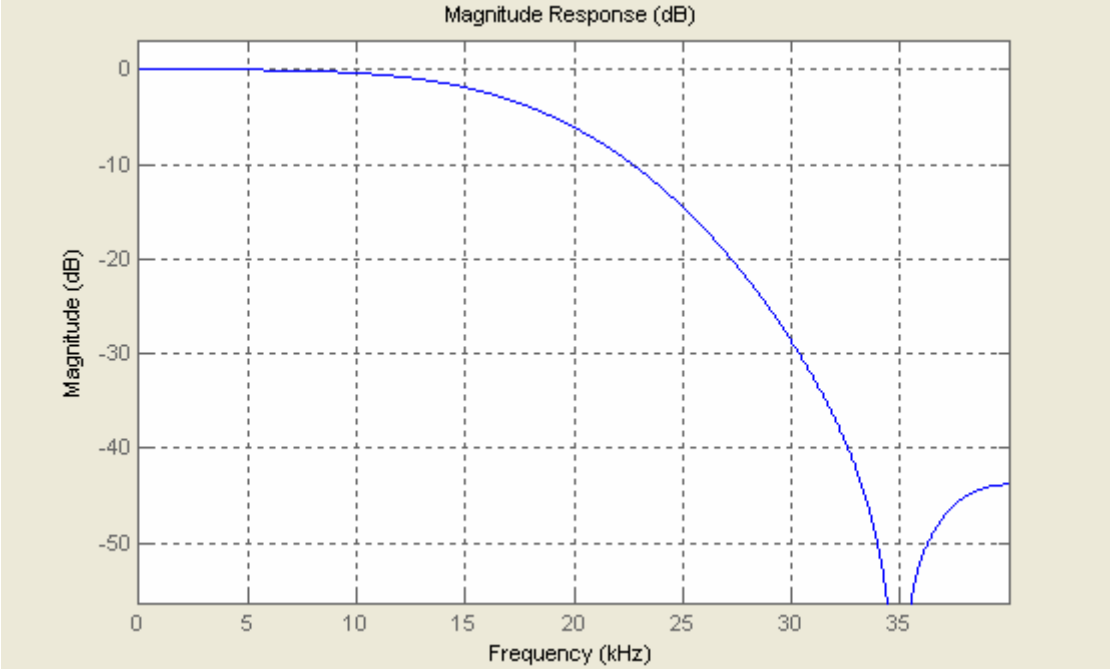


(b)

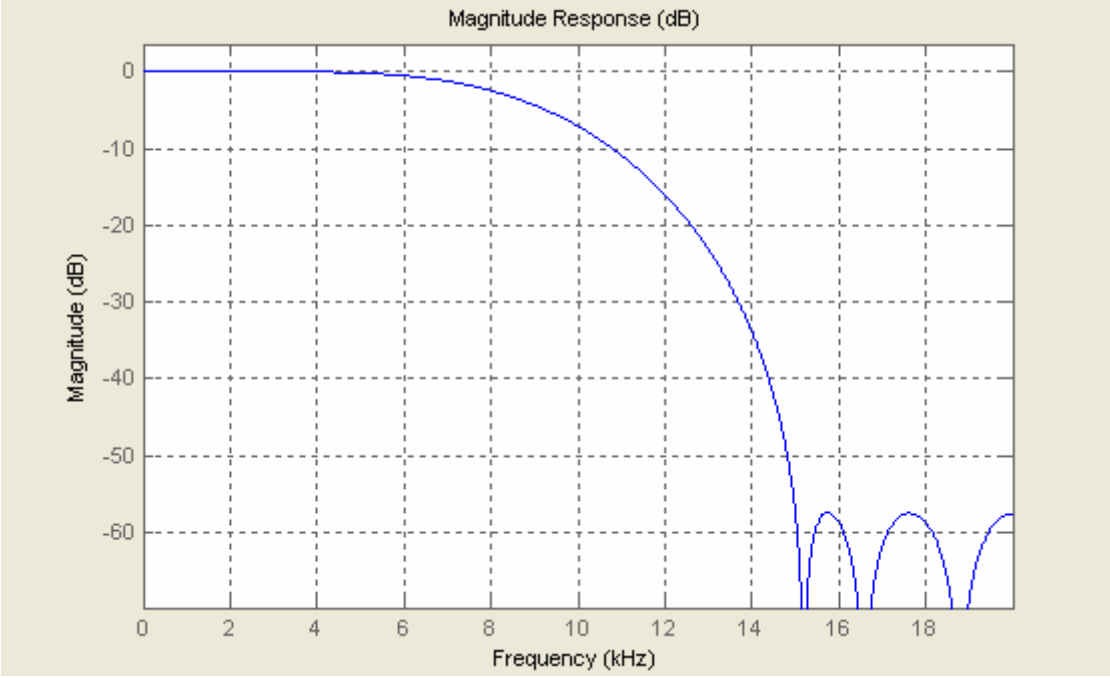


(c)

Fig 7.10 The output response of comb-FIR-FIR: (a) the filter order 22 (b) response of the filter order11 (c) response of the output of the last stage Corrector FRI filter



(a) half-band-FIR filter with order 11



(b)

Fig 7.11. **T**he output responses of comb-half-band-FIR-FIR: (b) shows the output of the last stage Corrector FRI filter

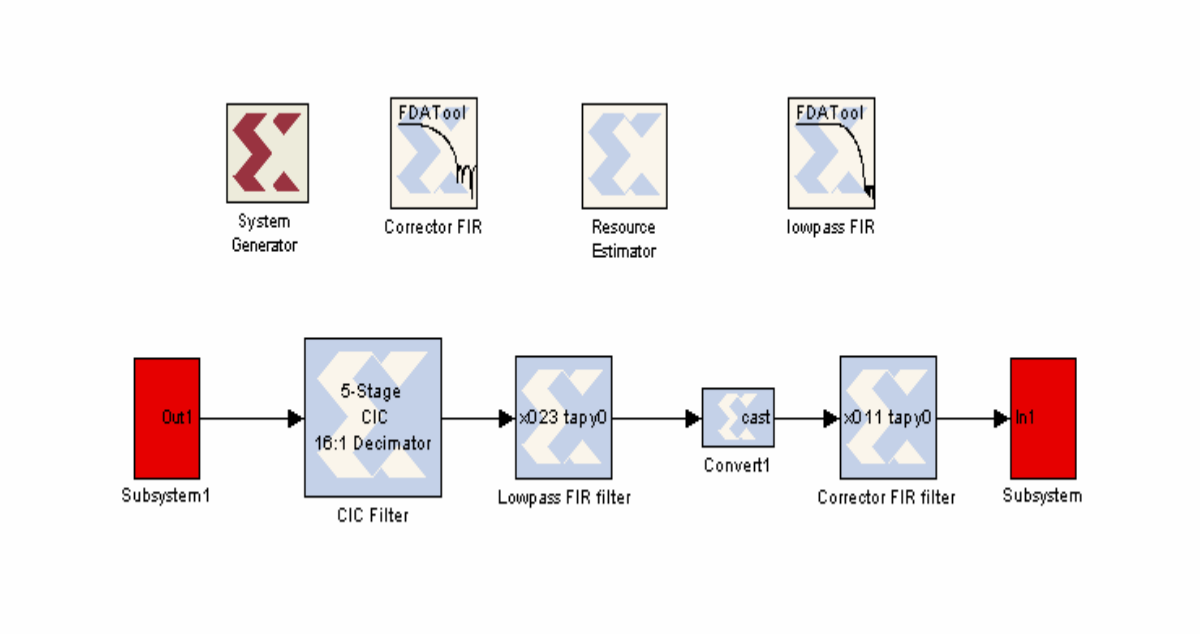


Fig 7.12 (a) Comb-FIR-FIR filter implementation using Xilinx blackest

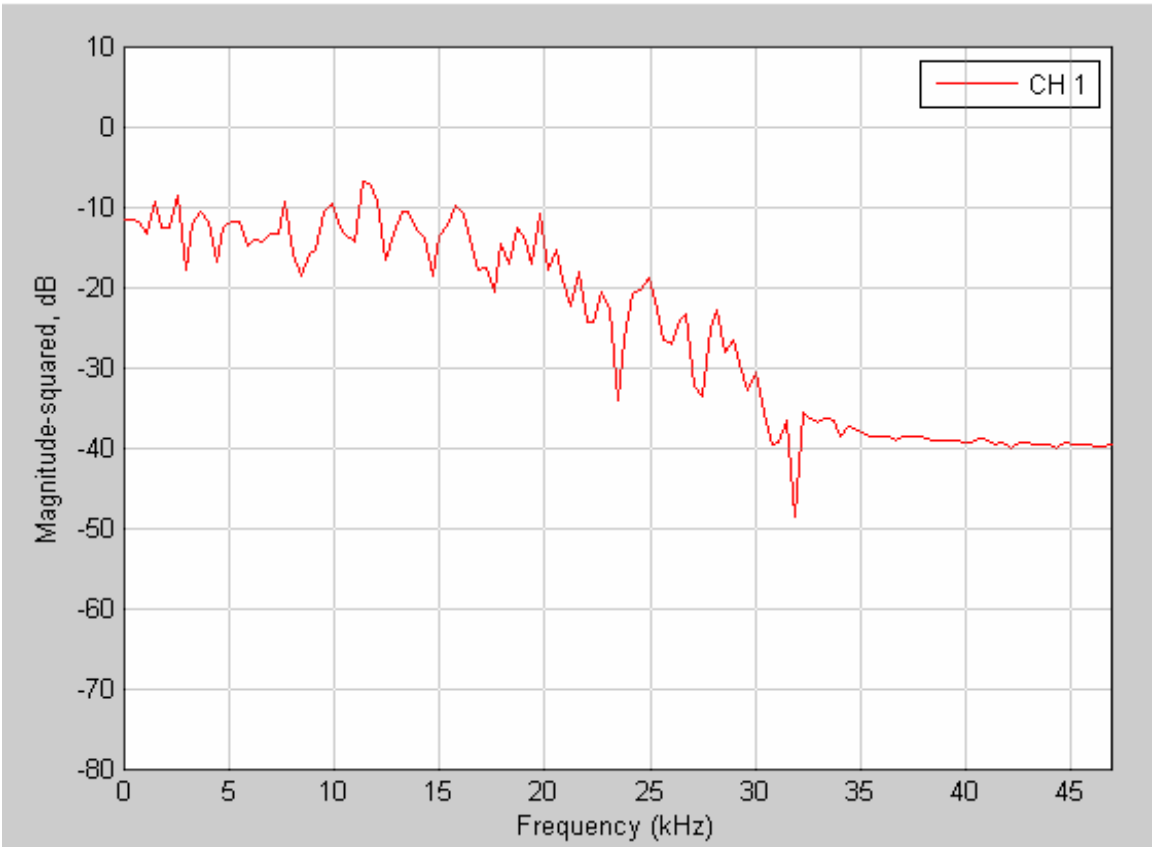


Fig 7.12(b) The output response of the comb-FIR-FIR using xilinx block set

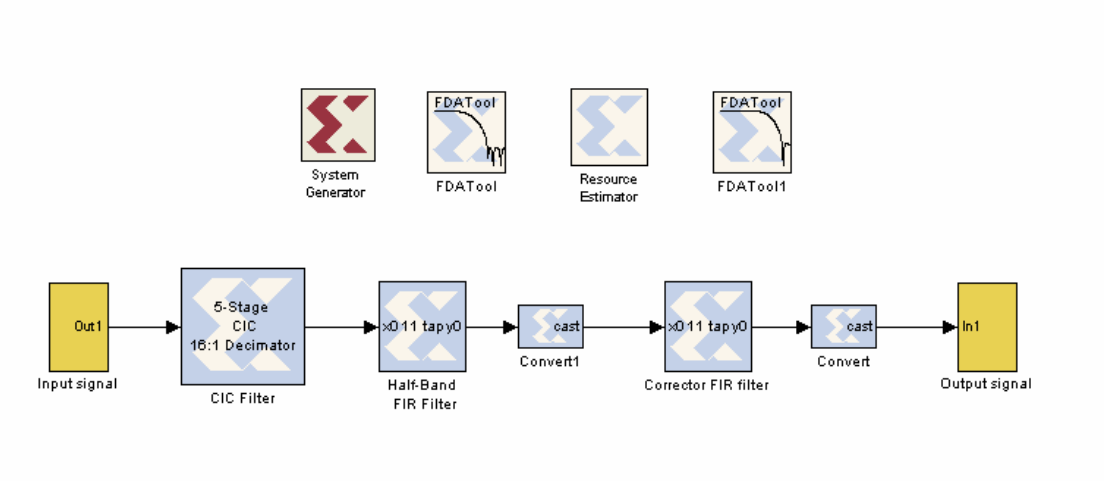


Fig 7.13 (a) Comb-half-band FIR-FIR filter implementation using Xilinx blackest

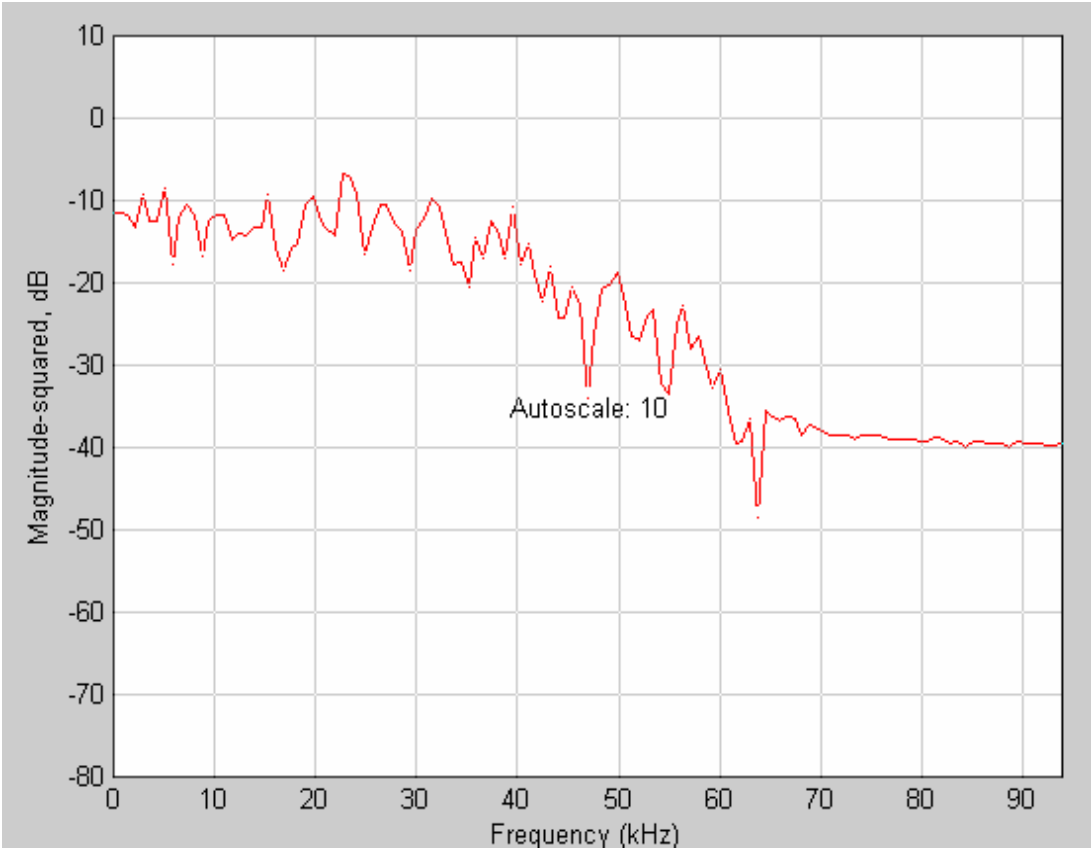


Fig 7.13 (b) the output response of the comb-half-band FIR-FIR using Xilinx block set

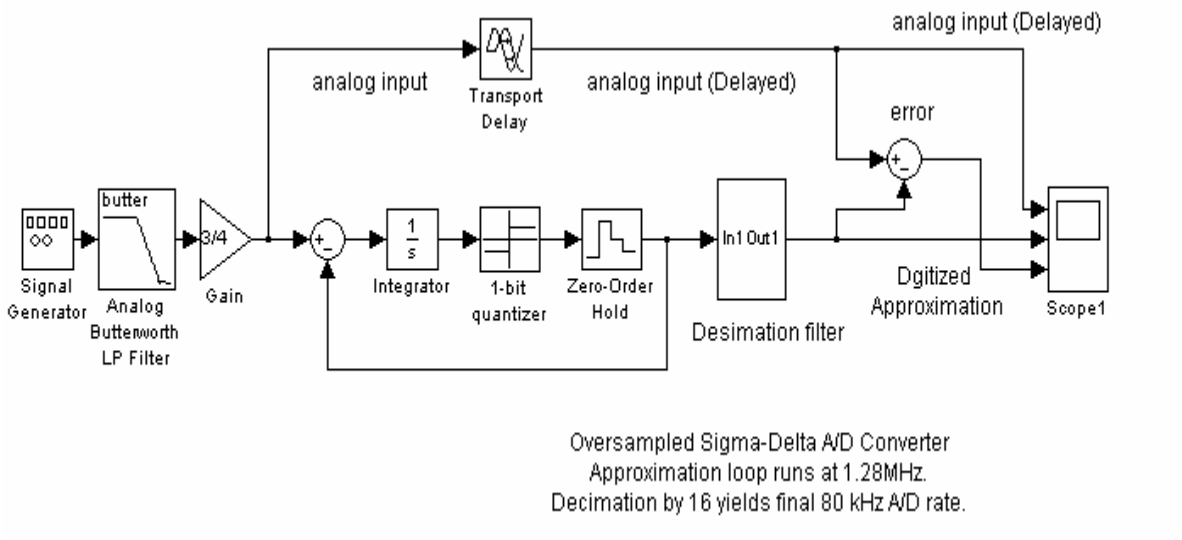


Fig 7.14 (a) Analog-to-digital conversion using a sigma-delta implementation

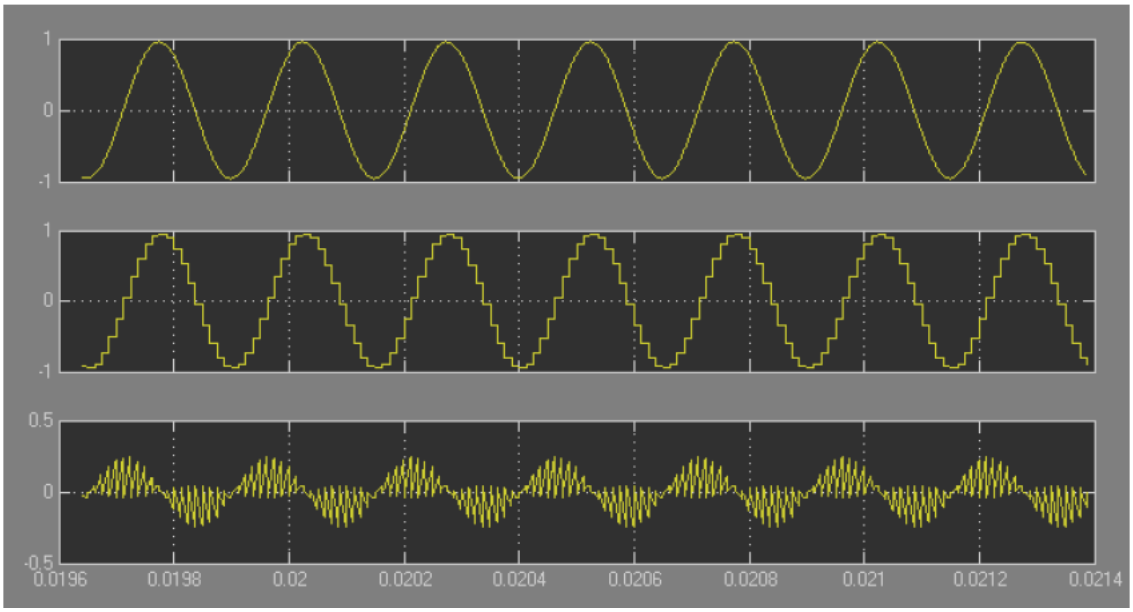


Fig7.14(b) the The output response of the Analog-to-digital conversion using a sigma-delta .First one is analog input (delayed), middle one is digitized approximation and the lost one is error signal.

7.3 VHDL SIMULATION RESULTS.

Table 7.1 shows the cell usage for the Comb-half-band FIR-FIR filter design and comb-FIR-FIR filter design. The comb-half band FIR-FIR filter design uses less hardware compared to the comb-FIR-FIR filter design. The power consumption is less using comb-half band FIR-FIR filter design compared to the comb-FIR-FIR filter.

Cells used	Slices	Slice Flip Flops	4-LUT	logic	Shift registers	IO
Comb-FIR-FIR	695	1474	773	628	145	22
Comb-half-band-FIR-FIR	639	1174	699	584	115	22

Table 7.1: Cells Usage for the design of CIC-Decimation filter for Hearing Aid Component in VIRTEXII PRO (XC2VP4-FF672)

Frequency used for calculating power consumption is 128 MHz. Table 7.2 shows that the proposed decimation filter using Comb-half-band-FIR-FIR architecture requires less hardware and contributes to hardware saving and a power saving compared to the comb-FIR-FIR architecture.

Decimation filter architectures	No of taps	Power consumption
5-stage Comb-FIR-FIR	22	61mW
5-stage Comb-half-band-FIR-FIR	11	50 mW

Table 7.2: Comparisons of different CIC decimation filter architectures

Chapter 8

CONCLUSION

The comb-half-band FIR-FIR decimation filter is designed using Matlab and checked for real-time implementation using Simulink. The decimation filter is designed for 6-bit data stream input. The final output of the filter is 13 bits, and the stop band attenuation obtained is -65 dB. In addition, the distributed arithmetic multiplier is used for implementing in VHDL. Specifically, we compare the relative power consumption of two designs; the cell usage for each design is obtained using the synthesis report. The proposed decimation filter architecture requires less hardware and contributes to a hardware saving compared to the comb-FIR-FIR architecture. Decimation filters for audio applications use FIR filters because of the ease with which linear phase can be achieved. However, linear phase over the entire band is not often required.

Future work

Future works include designing decimation filters using n-band filters and examining their performance.

9. REFERENCES

1) Venugopal V and Abed, Khalid.H. ‘’Design and implementation of a decimation filter for hearing aid applications’’. Nerurkar, S.B.8-10 April 2005 Page(s):111 – 115
2) John G. Proakis, Dimitris G.Monalikes, and D.Sharma. “Digital Signal Processing principles, Algorithms, and Applications’’, Third Edition.
3) P. M. Aziz, H. V. Sorenson and J. Van der Spiegel, “An overview of Sigma Delta. Converters,” IEEE Signal Processing Magazine, vol.13, pp. 61-84, January 1996.
4) Steven W. Smith, The Scientist and Engineer's Guide to ‘’Digital Signal Processing’’, Chapter 22, Second edition, 1999.
5) D. George, W. Sjursen, J. R. Hochschild, et al“A 1.1-V .270-µA mixed-signal hearing aid chip,” IEEE Journal of Solid-State Circuits,vol. 37, no.12, December 2002.
6) J. C. Candy, “Decimation for sigma-delta modulation,” IEEE Trans. Commun., vol. COM-34, pp. 72–76, Jan. 1986.
7) R. E. Crochiere and L. R. Rabiner, ‘’Multirate Digital Signal Processing’’ Prentice-Hall, 1983.
8) L. Naviner, J.F.Naviner, “Design and Prototyping of a ∑Δ Decimator Filter for DECT .Standard”, Proceedings of 43rd IEEE Midwest symposium on Circuits and Systems, 2000.
9) V. Oppenheim and R. W. Schafer, ‘’Discrete-Time Signal Processing. Prentice-Hall’’, 1989.
10) Tormod Njolstad and Tor A. Ramstad, ‘’ Low-power Narrow Transition-band FIR Filters’ Norwegian University of science and Technology, dept of physicalelectronics, 7034, trondheim , Norway.
11) Peter Y. K. Cheung, Eric S.K.See ‘’A comparison OF Decimation Filter Architectures for Sigma-Delta A/D Converters’’ Department of Electrical Engineering, Imperil college of science, Technology and Medicine, londan, SW7 2BT, England.
12) Alan N. Willson, Jr., Fellow, and H. J. Orchard, Life Fellow, ‘’A Design Method for Half- Band FIR Filters ‘’ IEEE .vol. 45, No. 1, January 1999.
13) Rashid Ashid Ansari ’’ Satisfying the Haar Condition in Half band FIR Filter’’IEEE. vol *36.* No I . January 1988.

14) P. P. Vaidyanathan, ‘’Multirate Systems and Filter Banks’’. Englewood.Cliffs, NJ: Prentice-Hall, 1993.

15) R. M. Gray, “Oversampled sigma-delta modulation,” IEEE Trans.ommun**.,** vol.COM-35, pp. 481-489, May 1987.

16) R. E. Crochiere and L. R. Rabiner, “Interpolation and decimation of digital- signals- a tutorial review,” Proc.of the IEEE, vol. 69, pp. 300-331, March 1981.

17) C. J. Barret, “Low-power decimation filter design for multi-standard transceiver Applications,” MS (EE) Thesis, University of California, Berkeley.

18) S. R. Norsworthy, R. Schreier & G. C. Temes, “Delta-Sigma Data Converters’’ Theory, Design, and Simulation”, IEEE Press, 1998.

19) O. Shoaei “Continuous-time Delta-Sigma A/D Converters for high-peedpplications’’ Ph.D. dissertation, Carleton Univ., Ottawa, Ont., canada, Feb.1996.

20) R. Ansari: Elliptic fillter design for a class of generalized halfband filters., IEEE Trans. on Acoustic, Speech and Signal Processing, Vol. ASSP-33, No. 4, pp. 1146-1150, Oct. 1985.