LOW POWER CONTINUOUS TIME SIGMA DELTA MODULATOR AND DECIMATION FILTER DESIGN

by

Mehmet İnce

B.S., Electrical and Electronics Engineering, Bogazici University, 2012

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Electrical and Electronics Engineering
Boğaziçi University
2014

ACKNOWLEDGEMENTS

I would like to express my deep gratitude to my supervisor Professor Günhan Dündar for giving me support and encouragement throughout my bachelor's degree and master education. I sincerely thank to him for his guidance, patience and understanding in preparation of my thesis.

I would like to thank to residents of the Beta Electronics Laboratory for their helps.

I am very grateful to TÜBİTAK-BİDEB for their financial support during my master education.

I would like to express my gratefulness and gratitude to my family for their love and encouragement; they are always by my side to support me throughout my life.

Finally, I would like to thank to my fiancé for her love, support and patience.

ABSTRACT

LOW POWER CONTINUOUS TIME SIGMA DELTA MODULATOR AND DECIMATION FILTER DESIGN

Sigma Delta analog to digital converters (ADC) are widely used in areas where high resolution is needed. Audio applications are one of these areas. Along with high resolution, reducing the power consumption of an ADC is a very significant design problem. A Sigma Delta ADC consists of two main blocks: analog modulator and digital decimation filter. Modulator utilizes oversampling and noise shaping processes in order to move noise to higher frequencies whereas the decimation filter down-samples the sampling frequency and eliminates redundant data which are result of the oversampling process.

Concerning low power ADC design, most of the effort is being spent on analog part of the converter. The digital decimation filter can also consume as much power as the analog part. In this thesis, a low power Continuous Time Sigma Delta modulator for audio frequencies and a low power Decimation filter for discrete time Sigma Delta modulator intended for audio application are implemented.

A 2^{nd} order continuous time Sigma Delta modulator with an oversampling ratio of 128 is first designed in MATLAB, then the proper architecture is realized in Mentor Graphics schematically using 0.18 μm CMOS technology. Simple and easy to design structure is chosen for the modulator. Most of the power is consumed in operational amplifiers of integrators; therefore, operational amplifiers with low power consumption are designed. gm-C type integrators, which employ telescopic cascode amplifiers as gm unit, are utilized in the design. Power consumption of the modulator is found to be 9.82 μW.

A decimation filter is designed for a low power discrete time Sigma Delta modulator intended for audio applications. The modulator has an oversampling ratio of 32, a signal bandwidth of 25 kHz and a sampling frequency of 1.6 MHz. The main goal of the design is to reduce the power consumption of the digital filter below that of the analog modulator.

Multistage filter structure is employed in order to save area and power. The first stage of the 3-stage structure is Cascaded Integrator Comb Filter (CIC), the second and third stages are Half-Band (HB) and Finite Impulse Response (FIR) Filters respectively. Unlike conventional form, CIC filter is also implemented in three stages and non-recursive format such that it has FIR transfer function. HB filter coefficients are generated with MATLAB FDAtool which uses Parks McCellan Algorithm. FIR filter coefficients are generated with GAM algorithm which provides coefficients with minimum number of signed power of two (SPT) terms while keeping quantization word length as small as possible.

Several low power digital design techniques are applied to filters in order to obtain a low power decimation filter. Filters are designed with MATLAB Simulink tool, realized with Verilog hardware design language (HDL) and synthesized with Synopsys Design Compiler (DC) tool using 0.18 μm CMOS technology. Measured power consumption of the Decimation filter is 7.24 μW under 1.8 V supply voltage, only half of the DT Sigma Delta analog modulator.

ÖZET

DÜŞÜK GÜÇ TÜKETİMLİ SÜREKLİ ZAMAN SİGMA DELTA MODÜLATOR VE ÖRNEK SEYRELTME SÜZGECİ TASARIMI

Sigma Delta analog sayısal çevirici, yüksek çözünürlüğün ihtiyaç duyulduğu alanlarda kullanılmaktadır. Ses uygulamaları da bu alanların başında gelir. Yüksek çözünürlük ile birlikte, analog sayısal çeviricilerin güç tüketimini azaltmakta çok önemli bir tasarım önceliğidir. Sigma Delta analog sayısal çevirici iki ana bloktan oluşur: analog modülatör ve sayısal seyreltme süzgeci. Modülatör aşırı örnekleme ve gürültü şekillendirme işlemlerini kullanarak, gürültüyü yüksek frekanslara atarken, alçak geçiren seyreltme filtresi de örnekleme miktarını düşürür ve aşırı örnekleme sonucu oluşan gereksiz bilginin atılmasını sağlar.

Düşük güç tüketimli analog sayısal çevirici tasarımı temel alındığında, çalışamalar genellikle analog kısım üzerine yoğunlaşmaktadır. Sayısal seyreltme süzgeci kısmı da analog kısım kadar güç harcayabilmektedir. Bu tezde, hem düşük güç tüketimli ses frekanslarında çalışan sürekli zaman Sigma Delta modülatör hemde ayrık zamanlı ses frekanslarında çalışan Sigma Delta modulator için seyreltme süzgeci tasarımları yapılmıştır.

2. derece aşırı örnekleme miktarı 128 olan ayrık zamanlı Sigma Delta modülatörünün öncelikle MATLAB programı üzerinde tasarımı yapılmış daha sonar ise Mentor Graphics programında 0.18 μm CMOS teknolojisi kullanılarak şematik olarak gerçeklenmiştir. Basit ve kolay tasarlanabilecek bir yapı tercih edilmiştir. Gücün büyük bir kısmı toplayıcıyı oluşturan işlemsel kuvvetlendiricilerde harcandığı için, düşük tüketimli işlemsel kuvvetlendiriciler tasarlanmıştır. İçerisinde teleskopik kaskod işlemsel kuvvetlendirici barındıran gm-C tipi toplayıcılar kullanılmıştır. Modülatörün güç tüketimi 9.82 μW olarak bulunmuştur.

Seyreltme süzgecinin, ayrık zamanlı ses frekanslarında çalışan Sigma Delta modülatör için tasarlanmıştır. Modülatörün aşırı örnekleme miktarı 32, sinyal bant genişliği 25 kHz ve örnekleme miktarı 1.6 MHz'dir. Tasarımdaki öncelikli amaç sayısal seyreltme süzgecinin güç tüketimini analog mdulatörünkinden daha düşük seviyelere düşürmektir.

Alandan ve güçten tasarruf sağlamak için çok katlı süzgeç yapısı tercih edilmiştir. 3 katlı yapının birinci katında CIC süzgeç, ikinci ve üçüncü katında ise sırasıyla Yarı-Bant ve Sonlu Dürtü Yanıtlı (FIR) süzgeç bulunmaktadır. Geleneksel yapısının aksine CIC süzgeç tasarımında da 3 katlı yapı kullannılmıştır ve tekrarlı olmayan yapıya dönüştürülmüştür; bu sayede FIR tipinde süzgeç elde edilmiştir. Yarı-Bant Süzgeç katsayıları Parks McCellan algoritması kullanan MATLAB FDAtool aracılığı ile üretilmiştir. FIR Süzgeç katsayıları da GAM algoritması ile elde edilmiştir. GAM algoritması katsayılardaki işaretli ikinin kuvvetleri terimlerini kelime uzunluklarını olabildiğince kısa tutarak azaltır.

Yukarıdakilere ek olarak, bir kısım düşük güç tüketimli sayısal tasarım teknikleri uygulanarak, düşük güç tüketimine sahip Örnek Seyreltme Süzgeci elde edilmiştir. Süzgeçler MATLAB Simulink programı ile tasarlanmış, Verilog HDL ile gerçeklenmiş ve Synopsys DC programında 0.18 μm CMOS teknolojisi kullanılarak sentezlenmiştir. 1.8 V besleme geriliminde Örnek Seyreltme Süzgecinin güç tüketimi 7.24 μW olarak ölçülmüştür. Bu değer ayrık zamanlı Sigma Delta analog modülatörün güç tüketiminin yalnızca yalnızca yarısı kadardır.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS/ABBREVIATIONS

$y(n)$	Output of the filter
$x(n)$	Input of the filter
h_k	Filter coefficients
x_{n-k}	Input shifted by k
f_s	Sampling frequency
f_N	Nyquist frequency
f_0	Input signal frequency
f_c	Maximum frequency of interest
Δ	Error in quantization
L	Modulator order
F_{pass}	Passband frequency
F_{stop}	Stopband frequency
A_{pass}	Passband ripple
A_{stop}	Stopband ripple
R	Decimation ratio
M	Differential delay
B	Number of bits
$\lceil x \rceil$	The smallest integer not less than x
$\gg n$	Right shift operation
$\ll n$	Right shift operation
N	Filter order
gm	Transconductance
a	Integrator coefficient
BW	Band width
OSR	Oversampling ratio
ADC	Analog digital converter
CM	Common mode
CMFB	Common mode feedback

VLSI	Very large scale integration
CIC	Cascaded integrator-comb
HB	Half-band
FIR	Finite impulse response
IIR	Infinite impulse response
CSD	Canonical signed digit
DAC	Digital analog converter
SPT	Signed power of two
RTL	Register transfer level
HDL	hardware description language
FoM	Figure of merit
CT	Continuous time
DT	Discrete time
S/H	Sample and hold
NTF	Noise transfer function
STF	Signal transfer function
IBN	In band noise
RZ	Return to zero
NRZ	Non-return to zero
ENOB	Effective number of bits
OTA	Operational transconductance amplifier
FFT	Fast Fourier transform
DFT	Discrete time Fourier transform
PSD	Power spectral density

1 INTRODUCTION

Mobile solutions for everyday use are becoming more and more popular. Systems containing different functions are integrated on a single device. Since they rely on battery, the demand for electronic circuits which consume less power has increased. Along with the consumer devices, medical applications with longer battery life are also in demand. The demand for low power combined with the profitability and incentives of government institutions make low power design focus of interest.

In digital circuits, processing speeds are much higher than analog counterparts. Therefore, there is a tendency to move processing into the digital domain. However, all real life signals are analog. Analog to Digital Converters are needed to convert real life analog signals to digital signals for fast processing.

In the literature, there are various types of analog to digital converters. Sigma Delta analog to digital converters are one type and are widely used because of their high speed and high resolution. Sigma Delta converters utilize oversampling which relaxes constraints on analog components and enables achieving high resolution with relatively simple and high tolerance analog components. Signals are sampled at a much higher rate than the Nyquist rate in order to shape quantization noise and place it out of the band in the spectrum. Because of this feature, a decimation filter is required at the end of the converter to decimate the signal again to the Nyquist rate.

Sigma Delta Converters have less power consumption compared to other converters. Although they operate at higher frequencies proportional to the oversampling ratio, they utilize less active elements compared to other analog to digital converters.

If the sigma delta converter is low pass, then the decimation filter is also a low pass filter. Decimation is lowering the sampling frequency of the signal. For digital data it can also be considered as eliminating redundant data in the signal. In order to prevent aliasing, the signal is processed through a low pass filter first and then its data rate is reduced. According to the sampling theorem, the required sample rate is two times the input

bandwidth in order to reliably reconstruct the input signal without distortion. However, sigma delta converter signal is oversampled in order to reduce the quantization noise in the baseband. Oversampled signal contains lots of redundant data with quantization noise mostly residing at higher frequencies. Digital filters attenuate noise and interference and eliminate the redundant data [4].

In the literature most of the effort for the low power Sigma Delta converter design is focused on the analog part, namely the modulator. However, the digital part which is the decimation filter also consumes as much power as the modulator [4]. In order to achieve low power consuming converter, both blocks should be implemented efficiently in terms of power.

There are mainly two types of modulators in Sigma Delta Converters: Discrete Time (DT) and Continuous Time (CT). In a DT ΣΔ ADC, the input data is sampled and conversion is done for this sampled data. Input is taken at discrete time intervals and converted to digital. On the other hand, in the CT counterpart, as its name suggests, noise shaping is done in the continuous time domain; input is not sampled. Operating in CT has advantages for active components. There are no instantaneous changes at the output of amplifiers; therefore, constraints on slew rate are more relaxed. Additionally, since there is no settling time constraint for CT ΣΔ ADC as opposed to DT ΣΔ ADC, oversampling ratio can be increased further, which in turn results in higher SNR values for the same power consumption.

Structure of the digital decimation filter at the end of the converter is independent of the type of the modulator. The decimation filter can be implanted either in single stage or multiple stages. Single stage implementation is costly in terms of both area and power. For the multiple-stage implementation, different types of filters can be used in each stage. Cascaded integrator comb (CIC) filters are commonly used as the first stage of the filter. Other stages can be finite impulse response filters (FIR) or infinite impulse response filters (IIR). FIR filters have a linear phase response; therefore they can achieve flat group delay [4]. Flat group delay is very important for audio signals, because it preserves the envelope of the signal. FIR filters are inherently stable and decimation can be simply incorporated.

In addition, FIR filters are easier to design. FIR filters are used commonly at the back end of the Sigma Delta Converters because of the reason mentioned above [5].

For low pass ΣΔ ADC, decimation filter is also low pass. One approach for low pass operation is CIC filter. It has very simple structure and can effectively reduce sampling frequency. The other approach is the sinc based approach. In this approach, input samples are convolved directly with filter coefficients of the sinc based digital filter. Coefficients of the filter are samples of the sinc function which has a low pass response in frequency domain. In time domain, coefficients are represented as sums of power of two terms. On the condition where input bandwidth is constant, filter coefficients can also be constant. Both CIC and sinc based filters can be implemented in multiplierless format [6, 7].

In this thesis, the aim is to design a low power decimation filter for a Sigma Delta converter intended for audio application and a low power CT Sigma Delta converter again for audio applications. The main aim of the decimation filter is to achieve power consumption less than the DT Sigma Delta modulator without compromising performance. Power consumption of the DT Sigma Delta modulator is around 15 μW and it can achieve over 90 dB SNR ratio which corresponds to over 15 bits resolution.

The designed CT Sigma Delta and decimation filter are not meant to work together. The decimation filter is designed for a specific DT Sigma Delta Converter which has different oversampling ratio from the CT Sigma Delta. Decimation factor is the only difference in the perspective of decimation filter design. Therefore, with some modifications, the same decimation filter can also be used for the CT Sigma Delta Converter.

In section II, Sigma Delta converters, in general, are discussed. In section III, design of CT Sigma Delta modulator is explained. The decimation filter is briefly reviewed and the design procedure is introduced in section IV. Finally conclusion is presented in Chapter V.

2 SIGMA DELTA CONVERTERS

2.1 Basics of Analog to Digital Conversion

In the literature, there are various types of A/D converters. Although they are different in design approach and structure, they share some properties which are also the mathematical background for conversion. In the heart of the conversion, two main operations reside: Sampling and quantization. Sampling is done uniformly in time and quantization is done in the amplitude of the signal.

2.1.1 Sampling

In time domain, the signal is represented by its amplitude versus time function. In this representation there are two variables: time and amplitude. These two properties of the function can be either discrete or continuous. Digital circuits operate with signals represented in discrete time discrete amplitude. However, in real life, all signals are continuous in time and amplitude.

A sampler transforms the continuous-time signal to its sampled data equivalent [8]. Sampling yields equally spaced delta functions whose amplitudes are the same as the signal at that point in time. In order words, the signal is multiplied by equally spaced delta functions whose amplitudes are one. The sampling operation can be represented by Equation (2.1) and is depicted in Figure 2.1

$$x(nT) = \sum x(t)\delta(t - nT)\ . \qquad (2.1)$$

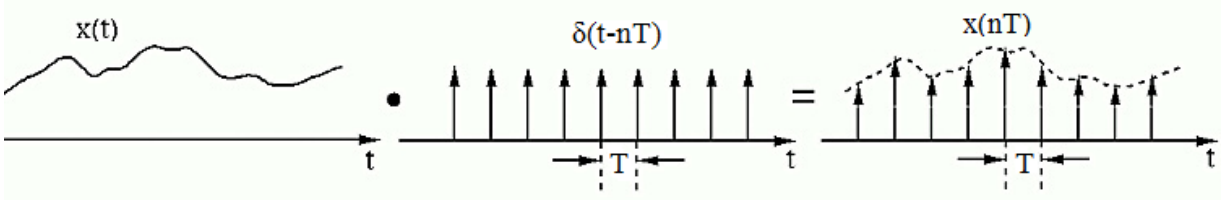


Figure 2.1. Sampling of a signal [1].

In practical circuits, delta function cannot be generated. However, it is crucial that sampled data have exactly the signal values at the sampling times. Therefore, instead of the delta function, pulses with finite duration can be used.

Sampling operation in frequency domain can be investigated with Laplace Transform. Laplace Transform of the sequence of delta functions is given in Equation (2.2).

$$L\left[\sum_{-\infty}^{\infty}\delta(t-nT)\right]=\sum_{-\infty}^{\infty}e^{-nsT} \tag{2.2}$$

Using Equations (2.1) and (2.2), Laplace transform results in,

$$L[x(nT)]=\sum_{-\infty}^{\infty}X(s-jnw_s) \tag{2.3}$$

where X(s) is the Laplace Transform of x(t). As can be deduced from Equation (2.3), quantization in time domain corresponds to periodicity in the frequency domain [9]. This is depicted in Figure 2.2.

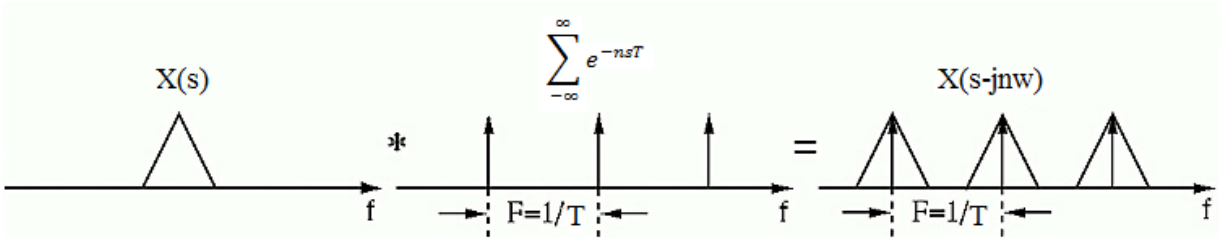


Figure 2.2. Sampling in frequency domain [1].

Sampling is an invertible process. The real signal can be reconstructed from its sampled form with an low pass filter. In order for a signal to be constructed without distortion, the sampling frequency which is f=1/T should be at least two times the bandwidth (BW) of the signal. This minimum sampling frequency is called the Nyquist rate (f_N). If the sampling frequency is lower than the Nyquist rate, the signal will alias with itself and it will be impossible to reconstruct the original signal. To assure signal BW to be

limited to $f_S/2$ an analog filter is placed at the input of the ADC, which is called the antialiasing filter (AAF). Converters working with f_N are called Nyquist Rate converters. In real implementations, the transition band of the AAF is not enough to cut off unwanted signals. It is desirable to have AAF with wider transition band because of its ease of design and lower cost in terms of area and power. Therefore, it is more practical to have sampling frequency higher than f_N. Ratio of the sampling frequency to f_N is called oversampling ratio and is given by Equation (2.4):

$$OSR = \frac{f_S}{2f_N} \tag{2.4}$$

2.1.2 Quantization

The term "quantization" is used for quantization in amplitude, which means converting continuous range of analog values into set of discrete levels. Quantization is clearly a noninvertible process since an infinite number of analog signal levels is mapped to a finite number of levels. The difference between the real value of the signal and the mapped quantization level is regarded as error. The main aim of the ADC design is to reduce this error. The number of discrete levels in the quantizer determines the number of bits. The amplitude of each quantization interval Δ is,

$$\Delta = \frac{FS}{2^B} \tag{2.5}$$

where FS is the difference between maximum and minimum of the input and B is the number of bits of the quantizer. 2^B gives number of quantization levels. As depicted in Figure 2.3, the quantization error is bounded to Δ.

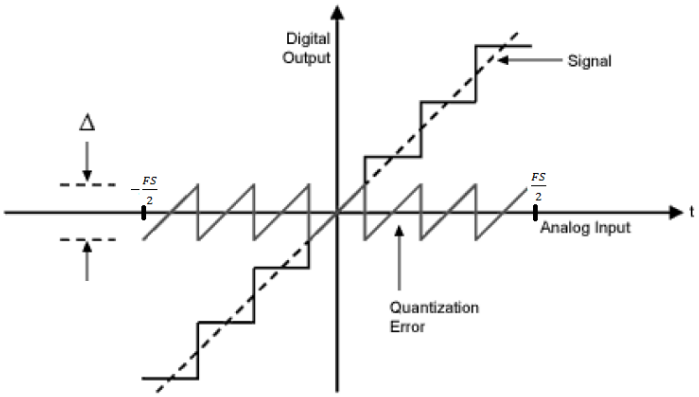


Figure 2.3. Quantization error.

Practically, quantization error can be considered to be uncorrelated with the input. Then, quantization error can be represented as white noise with probability density function shown in Figure 2.4.(a). The total quantization noise power become,

$$\sigma^2 = \int_{\frac{-\Delta}{2}}^{\frac{\Delta}{2}} e^2\, pdf\, de = \frac{\Delta^2}{12} \tag{2.6}$$

and the power spectral density of the quantization noise is,

$$S(f) = \frac{\Delta^2}{12fs} \tag{2.7}$$

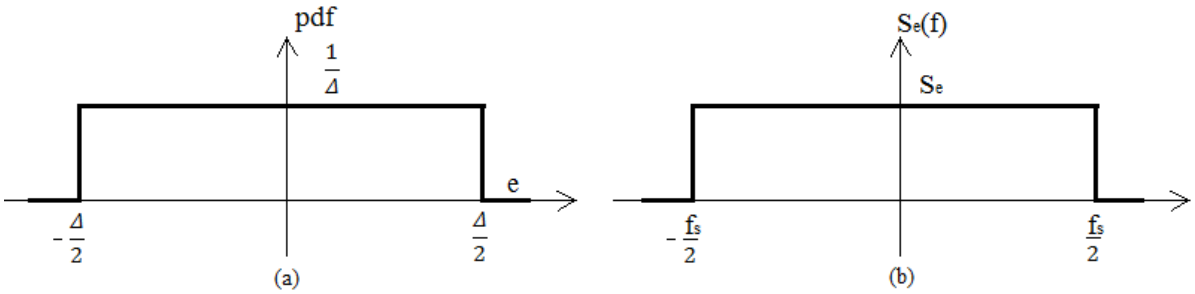


Figure 2.4. Probability density and power spectral density functions of quantization noise.

From Equation (2.7) one can deduce that as the sampling frequency increases power spectral density will drop. This relationship will be very beneficial for oversampling sigma delta ADC.

With light of the information given above, the structure of the basic ADC simplifies to Figure 2.5.

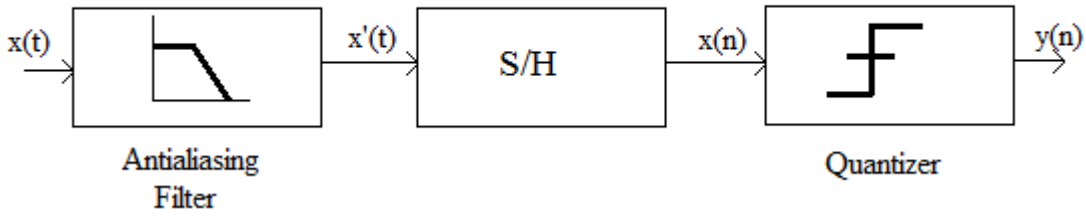


Figure 2.5. Simplified ADC structure.

2.2 Sigma Delta Converter

The ADC structure in Figure 2.5 is an open loop structure. Introducing a feedback path in the system allows closed control loop. Thus, with an additional controller in the forward or feedback path, different transfer functions can be obtained for unwanted noise on the one hand and the desired signal on the other hand, defining a signal transfer function (STF) and noise transfer function (NTF) [9]. The block diagram of the sigma delta converter with the mentioned feedback path is depicted in Figure 2.6.

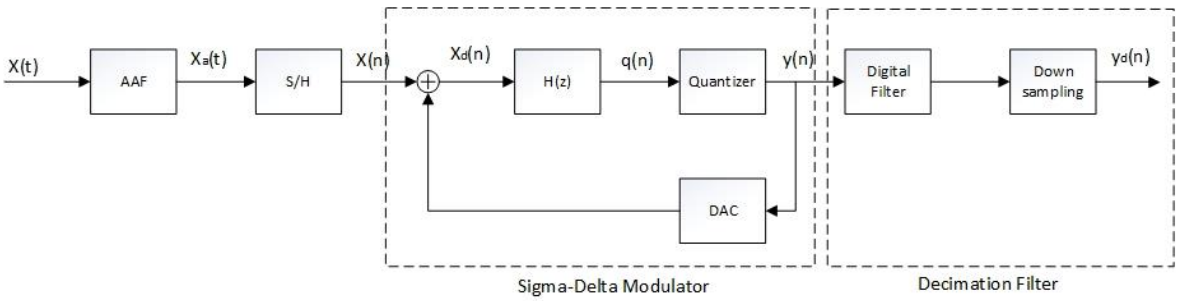


Figure 2.6 Block Diagram of Sigma Delta Converter.

Antialiasing filter and sampling has already been explained. Modulator is the part which conducts actual analog to digital conversion via sampling and quantization. The quantizer in the modulator has low resolution, up to 6 bits reported in the literature [10]. In the modulator, H(z) in the forward path allows to have a different functions for noise and signal. DAC in the modulator has also low resolution, so it does not introduce extra error.

The decimation filter at the end of the converter filters the out of band noise and down sampler reduces the sampling frequency to the Nyquist rate.

If the quantizer is replaced with a linear model consisting of a gain and added error STF and NTF become:

$$STF(z) = \frac{H(z)k_q}{H(z)k_q + 1}, \qquad NTFz(z) = \frac{1}{1 + H(z)k_q} \tag{2.8}$$

where k_q is the gain in the linearized model.

2.2.1 First Order Sigma Delta modulator

For low frequency signals, simplest choice for the loop filter H(z) is integrator. The integrator transfer function is:

$$H(z) = \frac{z^{-1}}{1 - z^{-1}} \tag{2.9}$$

For low frequencies STF and NTF becomes:

$$STF(z) = z^{-1}, \qquad NTFz(z) = \frac{1 - z^{-1}}{k_q} \tag{2.10}$$

Based on Equation (2.10), the STF behaves as delay element and NTF has high pass characteristics. Therefore, noise is suppressed in the signal band close to zero and amplified at high frequencies. This phenomenon is called noise shaping. As mentioned before, by oversampling, quantization error/noise is distributed over a wider spectrum, decreasing the noise in the band of interest. Additionally, by noise shaping, this in-band noise is further decreased and SNR is increased. Because of these two properties, Sigma Delta Converters can achieve very high resolutions. The effect of oversampling and noise shaping properties of Sigma Delta Converters is illustrated in Figure 2.7.

When noise shaping introduced, the in-band quantization noise (IBN) becomes:

$$IBN \approx \frac{\Delta^2 \pi^2}{36{k_q}^2 OSR^3} \tag{2.11}$$

and if power of the signal is denoted by P_{sig}, signal to noise ratio is,

$$SNR = 10\ log_{10}\left(\frac{P_{sig}}{IBN}\right) \sim 10\ log_{10} OSR^3\ [dB] \tag{2.12}$$

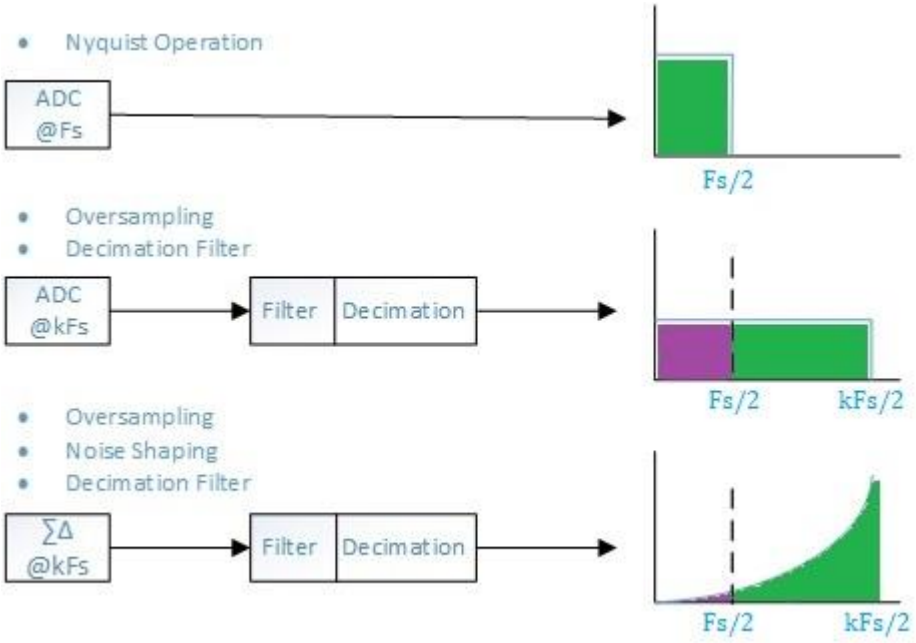


Figure 2.7. Effect of oversampling and noise shaping on power spectrum of noise.

Also,

$$SNR = 6{,}02\ x\ N + 1.76\ dB \tag{2.13}$$

where N is the number of bits.

Considering Equations (2.12) and (2.13) it can be calculated that for every doubling of OSR, 1.5 bit increase in resolution is achieved.

2.2.2 Continuous Time Sigma Delta modulator

The situation where the sampling is performed at the input is the discrete time implementation of the Sigma Delta Converter. Also, the integrator mentioned as the loop filter consists of a DT, z domain filter. In this situation, conversion is done on sampled data. Sampled data remain constant for a certain amount of time while passing through the loop filter and quantization forming the digital output.

Discrete time Sigma Delta has attracted more attention, because of the high degree of linearity and allegory between mathematics of Sigma Delta and DT implementation. In the literature, most of theoretical research and inventions are focused on DT Sigma Delta [9], e.g. [11-19].

Sigma Delta conversion can also be conducted in CT, using continuous time versions of the circuits. In CT Sigma Delta, sample and hold operation is done just before quantization and all the operations are performed in CT. In CT, errors in S/H circuitry are subject to noise shaping since it resides in the loop unlike DT case. CT structure is depicted in Figure 2.8.

CT and DT have both advantages and disadvantages with respect to each other. In DT Sigma Delta the output of the integrator changes very fast between sampling times which requires active elements with high slew rates. However in CT, output changes continuously over time; thus, slew rate requirements are relaxed and there is no settling time requirement for integration operation. Therefore, the same active elements can be operated at higher frequencies in CT Sigma Delta which result in higher performance without any extra power budget.

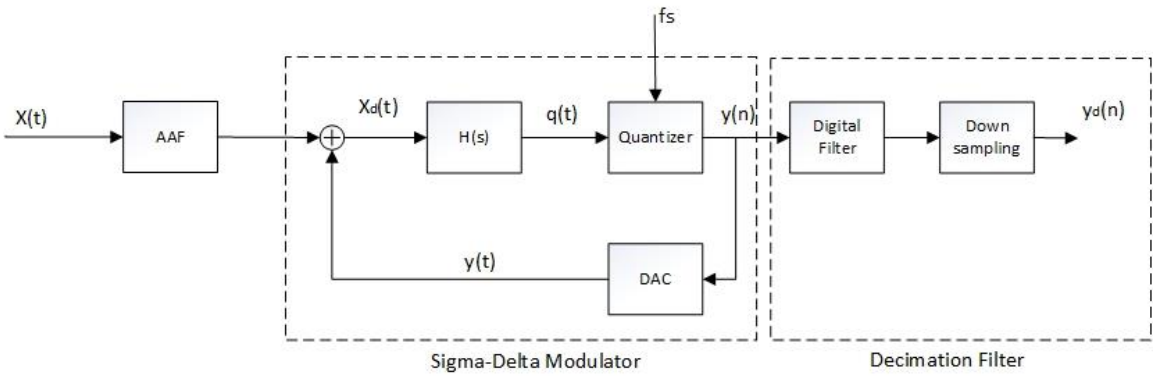
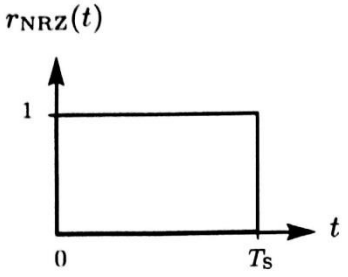


Figure 2.8. Block Diagram of CT Sigma Delta Converter.

In CT, the signal is fed back continuously; therefore, delay at the quantization and DAC circuitry accounted as error and may degrade the performance. In DT, comparators have half a clock cycle to determine the output. On the other hand, in CT, comparators of the quantizers ideally should have zero decision time. In circuits working with high frequencies decision time of the comparator poses a threat for the performance of the Sigma Delta; however, for relatively low frequency, for example audio signals, this is not a major problem.

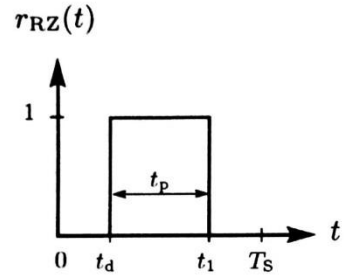
DT feedback is performed by charging a capacitor to a reference voltage and discharging it into the integrating capacitance. On the other hand, feedback in CT is applied continuously; therefore, every deviation from the ideal waveform can cause degradation in the performance. There are various types of feedback. Some forms with their time domain and s domain functions are illustrated in Figure 2.9. Among them, return to zero (RZ) and non-return to zero (NRZ) feedback types are common because of their ease of utilization. They can be generated using existing system clock and its edges [20]. On the other hand, other implementations, although they are more difficult to realize, offer some advantages.



(a)

$$r_{NRZ}(t) = \begin{cases} 1\ , & 0 \le t < T_S \\ 0\ , & \text{otherwise} \end{cases}$$

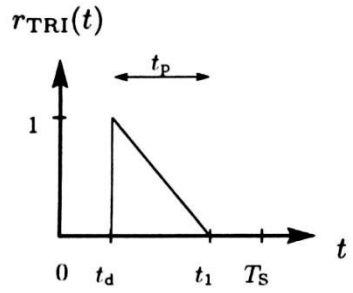
$$R_{NRZ}(s) = \frac{1 - e^{-sT_S}}{s}$$



(b)

$$r_{RZ}(t) = \begin{cases} 1\ , & t_d \le t < t_1 \\ 0\ , & \text{otherwise} \end{cases}$$

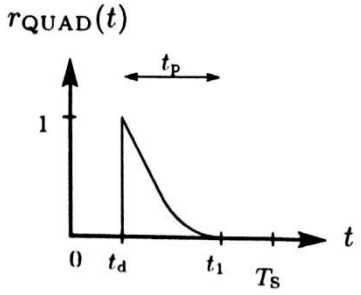
$$R_{RZ}(s) = \frac{e^{-st_d}(1 - e^{-st_p})}{s}$$



(c)

$$r_{TRI}(t) = \begin{cases} 1 - \frac{(t - t_d)}{t_p}\ , & t_d \le t < t_1 \\ 0\ , & \text{otherwise} \end{cases}$$

$$R_{TRI}(s) = \frac{e^{-st_d}}{s}\left(1 - \frac{(1 - e^{-st_p})}{st_p}\right)$$



(d)

$$r_{QUAD}(t) = \begin{cases} \left(1 - \frac{(t - t_d)}{t_p}\right)^2\ , & t_d \le t < t_1 \\ 0\ , & \text{otherwise} \end{cases}$$

$$R_{QUAD}(s) = \frac{e^{-st_d}}{s}\left(1 - \frac{2}{st_p} + \frac{2(1 - e^{-st_p})}{s^2 t_p^2}\right)$$

Figure 2.9 Common feedback pulses for CT Sigma Delta Converter [9] (a) NRZ-DAC; (b) RZ-DAC; (c) Linear decaying DAC; (d) quadratic DAC.

2.2.3 Decimation filter

In Sigma Delta operation, input data is oversampled for the sake of decreasing in-band quantization noise. At the output of the modulator, the digital signal has more than the minimum required samples. If the signal is processed in this format, it will cause unnecessary area and power consumption. It is known from the Nyquist theorem is that sampling only two times the BW of the signal is adequate for reliable reconstruction. Therefore, it is desired to reduce sampling rate after sigma delta operation. However, in order to prevent aliasing of the shaped noise, data should be filtered first and then down-sampled. Filtering in frequency domain is depicted in Figure 2.10 and down-sampling in time domain in Figure 2.11. After filter operation, thw sampling frequency is reduced to around the Nyquist rate. In Figure 2.10, signal frequency response has replicas at the multiples of the sampling frequency, which is f_s before decimation. After down-sampling, replicas will be at multiples of the reduced sampling frequency.

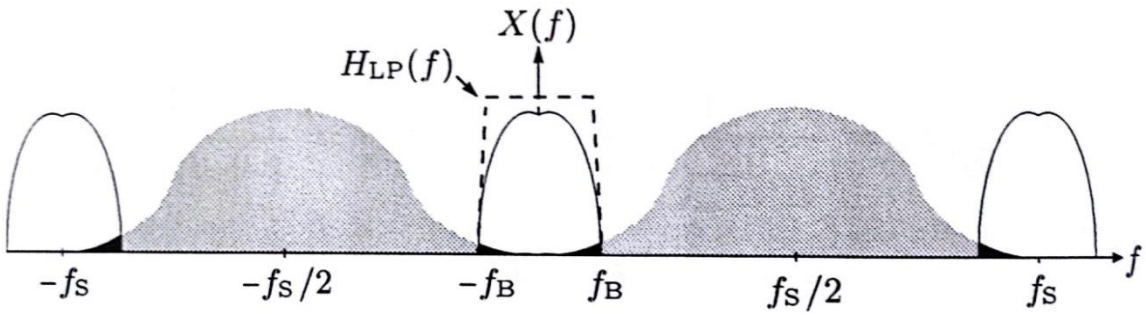


Figure 2.10. Low pass filtering of the modulator output [9].

Compared to the modulation operation, decimation is much simpler. Operating in digital domain also contributes to this fact. However, as it is mentioned before, it may consume as much power as the analog modulator; therefore, it is a concern for low power design. There are mainly two types of filters: IIR and FIR. Commonly used CIC filter is the combination of the two. These filters will be further investigated in Section 4.

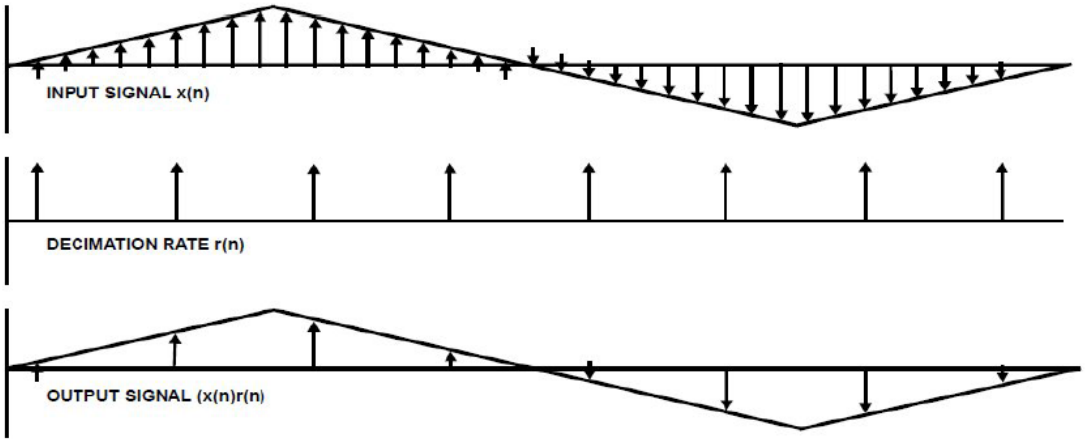


Figure 2.11.Decimation in time domain [42].

3 DESIGN OF CONTINUOUS TIME SIGMA DELTA MODULATOR

3.1 Introduction

Basics of the CT Sigma Delta modulator were explained in Section 2.2.1. In Figure 3.1 a first order CT Modulator system schematic is depicted. The order of the modulator is determined by the number of integrators and it describes the slope of the shaped noise. For an order of one, the out of band noise will have 20 dB/dec slope. As the order increases, there is less noise in the band of interest, which is the BW of the signal. Schematic is for a single bit; therefore, there is only one comparator. One bit is preferable because one bit operation is inherently linear. On the other hand, in multi-bit structures, linearity is a concern. Especially for the DAC of the CT Sigma Delta, linearity is very important and nonlinearities can cause considerable performance degradation. Also, one bit modulator design is much simple in terms of feedback implementation. This structure assumes that antialiasing filtering is done and there is no signal out of the band.

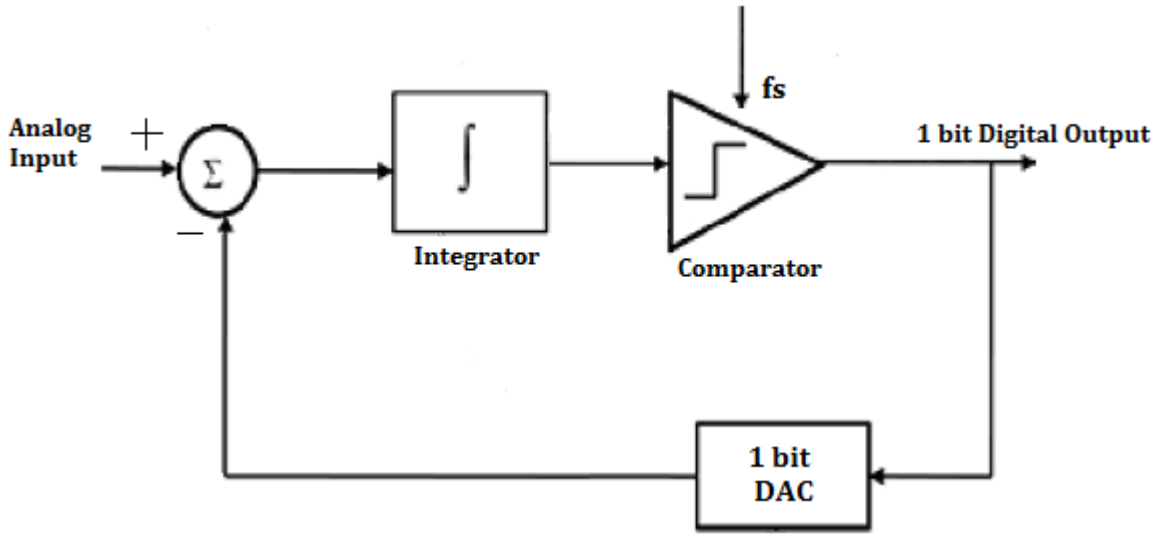


Figure 3.1. First order CT Sigma Delta modulator [42].

Increasing the order of the modulator has advantages and disadvantages. If the resolution of the modulator is taken as a reference for comparison, high order means more power consumption because of the active elements. As the order increases, stability will be

problem and much effort is needed to design a stable modulator. For higher order, the required OSR will be less and very high resolutions can be achieved.

First order is not plausible for high resolution applications because very high OSR values and multi-bit implementation should be used. As OSR increases, power consumption of the modulator is not affected but power consumption of the decimation filter increases proportionally. When converter is taken as a whole with the decimation filter its power consumption is greater for higher orders. Therefore it is practical to operate with higher order for high resolution, for instance higher than 10 bits or so. A 2nd order CT Sigma Delta modulator is illustrated in Figure 3.2.

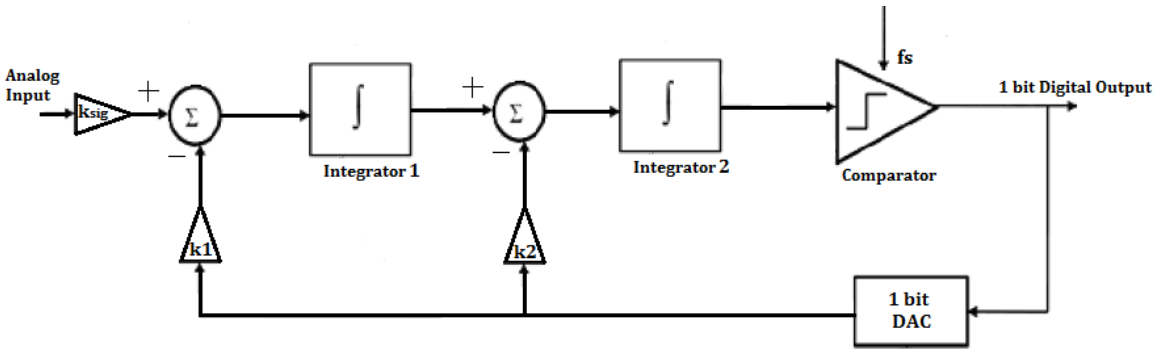


Figure 3.2. 2nd CT Sigma Delta modulator.

CT modulator consists of some parts that operate in CT and some in DT. Therefore, the signal is converted to a discrete signal and back to a continuous signal. In DT modulator, the analog input is sampled at the beginning and the whole operation, intermediate signals and transfer functions can be represented in z domain. However, because of the properties mentioned above, signal representation contains both z and s domain variables in CT modulator. Thus it is much more difficult to design CT modulator with pure calculation. There are some approaches for directly designing CT modulator without designing DT modulator [21], but because of its ease of implementation, DT to CT conversion is the prevalent choice. It is highly recommended to start with a DT modulator design having the required performance and to continue with DT to CT conversion in order to obtain the equivalent CT modulator [22].

DT to CT conversion can be obtained by impulse invariant transformation [13] or modified z transform [23-26]. In this thesis, impulse invariant transformation is used for DT- CT conversion.

If the coefficient of integrators of the 2nd order distributed feedback DT Sigma Delta modulator (Figure 3.3a) are a_1 and a_2, the loop filter transfer function becomes:

$$LF = -a_2 I(z) - a_1 a_2 I(z) = -\frac{a_2}{z-1} - \frac{a_1 a_2}{(z-1)^2} \tag{3.1}$$

where I(z) is the transfer function of the integrator. Equation (3.1) can be converted to its s domain equivalent using Table 3.1

Utilizing Table 3.1 and the DAC pulse representation of Figure 3.4, the loop filter of the CT modulator is,

$$LF(s) = -a_2 \frac{\frac{f_s}{\beta-\alpha}}{s} - a_1 a_2 \frac{\frac{1}{2}\frac{f_s(\alpha+\beta-2)}{\beta-\alpha}s + \frac{{f_s}^2}{\beta-\alpha}}{s^2} \tag{3.2}$$

If the feedback configuration is NRZ, where $\alpha = 0, \beta = 1, LF(s)$ becomes:

$$LF(s) = -\left(a_2 - \frac{a_1 a_2}{2}\right)\frac{f_s}{s} - a_1 a_2 \frac{{f_s}^2}{s^2} \tag{3.3}$$

Table 3.1. CT equivalents of the DT loop filter transfer functions [27].

$\mathcal{Z}$-domain	$\mathcal{S}$-domain equivalents with f_S (Hz) $= 1/T_S$
$\frac{1}{(z-1)}$	$\frac{w_0}{s}, \quad w_0 = \frac{f_S}{\beta - \alpha}$
$\frac{1}{(z-1)^2}$	$\frac{w_1 s + w_0}{s^2}, \quad w_0 = \frac{f_S^2}{\beta - \alpha}, \quad w_1 = \frac{1}{2}\frac{f_S(\alpha + \beta - 2)}{\beta - \alpha}$
$\frac{1}{(z-1)^3}$	$\frac{w_2 s^2 + w_1 s + w_0}{s^3}, \quad w_0 = \frac{f_S^3}{\beta - \alpha}, \quad w_1 = \frac{1}{2}\frac{f_S^2(\alpha + \beta - 3)}{\beta - \alpha},$ $w_2 = \frac{1}{12}\frac{f_S[\beta(\beta - 9) + \alpha(\alpha - 9) + 4\alpha\beta + 12]}{\beta - \alpha}$
$\frac{1}{(z-1)^4}$	$\frac{w_3 s^3 + w_2 s^2 + w_1 s + w_0}{s^4}, \quad w_0 = \frac{f_S^4}{\beta - \alpha}, \quad w_1 = \frac{f_S^3}{2}\frac{\beta + \alpha - 4}{\beta - \alpha},$ $w_2 = \frac{f_S^2}{12}\frac{(\beta - \alpha)^2 + 2\beta\alpha - 12(\beta + \alpha) + 22}{\beta - \alpha},$ $w_3 = \frac{f_S}{12}\frac{\beta^2(\alpha - 2) + \alpha^2(\beta - 2) - 8\alpha\beta + 11(\beta + \alpha) - 12}{\beta - \alpha}$

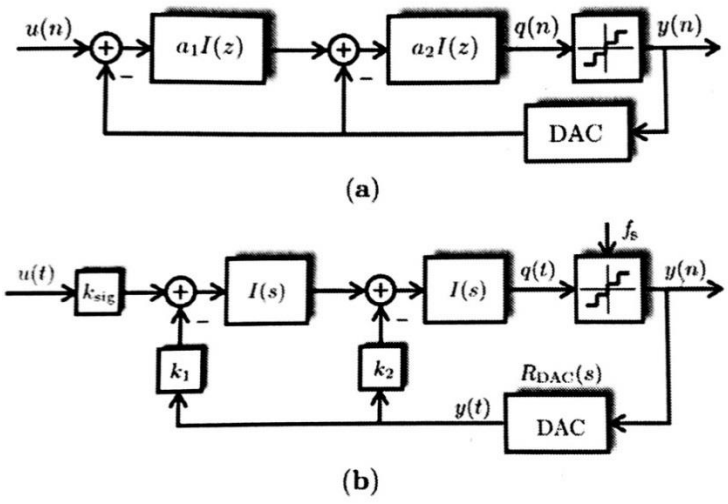


Figure 3.3. Block diagram of (a) 2nd DT Sigma Delta ; (b) a 2nd CT Sigma Delta modulator with distributed feedback [9].

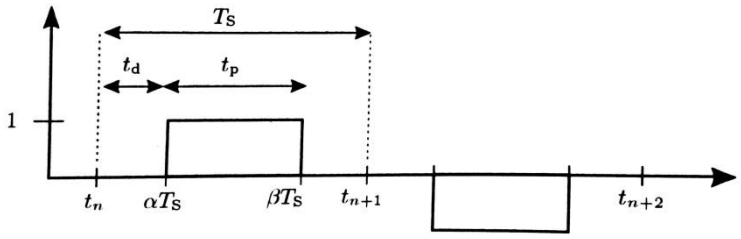


Figure 3.4. General model of rectangular DAC pulse [9].

The same loop filter transfer function can also be obtained directly from Fig 3.3. For NRZ feedback, $k_{sig}$and k_1 are equal to each other. In this condition, loop filter transfer function of the CT modulator in Figure 3.3b results in,

$$LF(s) = -k_2 I(s) - k_1 I^2(s) \tag{3.4}$$

If $I(s)$ is set to f_s/s $LF(s)$ is found to be,

$$LF(s) = -k_2 \frac{f_s}{s} - k_1 \frac{{f_s}^2}{s^2} \tag{3.5}$$

Using Equations (3.3) and (3.5), coefficients of the feedback path of the CT Sigma Delta modulator is found to be,

$$k_1 = a_1 a_2 \,, \qquad k_2 = a_2 - \frac{a_1 a_2}{2} \tag{3.6}$$

3.2 Modulator Design

A CT Sigma Delta modulator will be designed in this work. It is intended for audio applications and the main aim is to achieve low power consumption. In order to obtain low power design, 2nd order configuration with high OSR is chosen. In this way, fewer active elements are used.

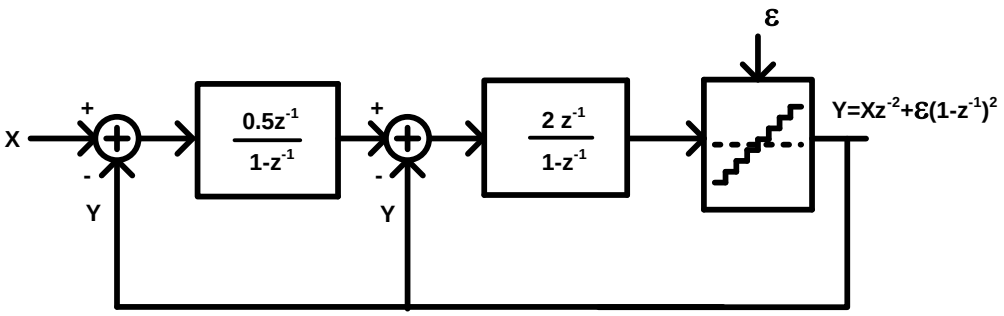


Figure 3.5. 2nd DT Sigma Delta modulator.

For the design of the 2nd order CT Sigma Delta modulator, the 2nd order DT modulator in Figure 3.5 is taken as reference. Applying impulse invariant transformation, the corresponding feedback coefficients of the CT modulator are obtained. Using (3.6), one obtains:

$$k_1 = 1\,, \qquad k_2 = \frac{3}{2} \tag{3.6}$$

The resulting CT modulator is given in Figure 3.6. DAC has NRZ type feedback pulses; therefore, k_{sig} equals k_1. CT does not have settling time limitation, unlike DT counterpart. There is no need to charge and discharge capacitances in a short time like in the case of DT modulator. Utilizing higher bit configuration lowers the swing at the output of the integrators. This can also be realized via modifying the transfer function of the system. As a result, the CT modulator is implemented as 1 bit, which means as a quantizer single comparator is used. Increasing OSR only increases the power consumption in the comparators, thus higher OSR values can be chosen for the 1 bit configuration. This configuration is set in MATLAB Simulink tool and simulated before transistor based design. Simulink system structure is depicted in Figure 3.7 and the Fast Fourier Transform (FFT) result in Figure 3.8.

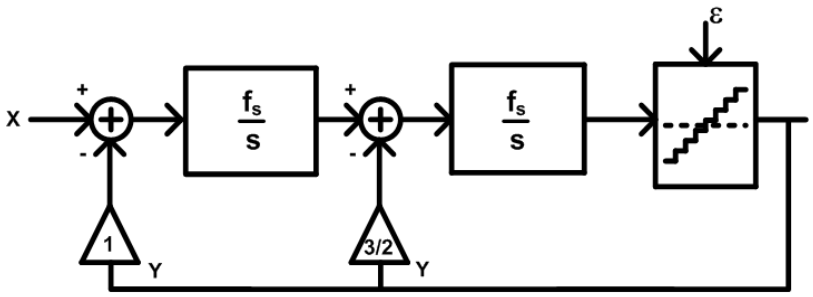


Figure 3.6. 2nd CT Sigma Delta modulator.

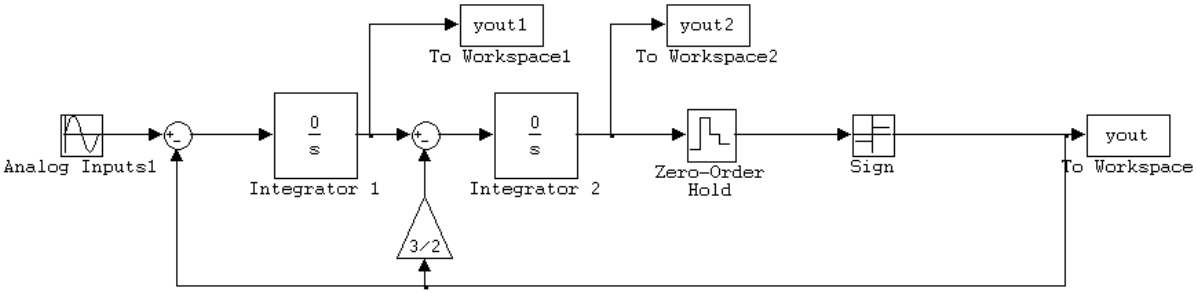


Figure 3.7. Simulink CT Modulator structure.

The obtained SNR values with Simulink simulations are about 85 dB. Effective number of bits (ENOB) for this configuration is found by,

$$ENOB = \frac{SNR - 1.78}{6.02} \tag{3.7}$$

to be 13.8. For audio applications, it can be considered as high resolution. The aim is to obtain high figure of merit with low power consumption. The structure in Fig 3.7 is chosen as a good candidate for the start.

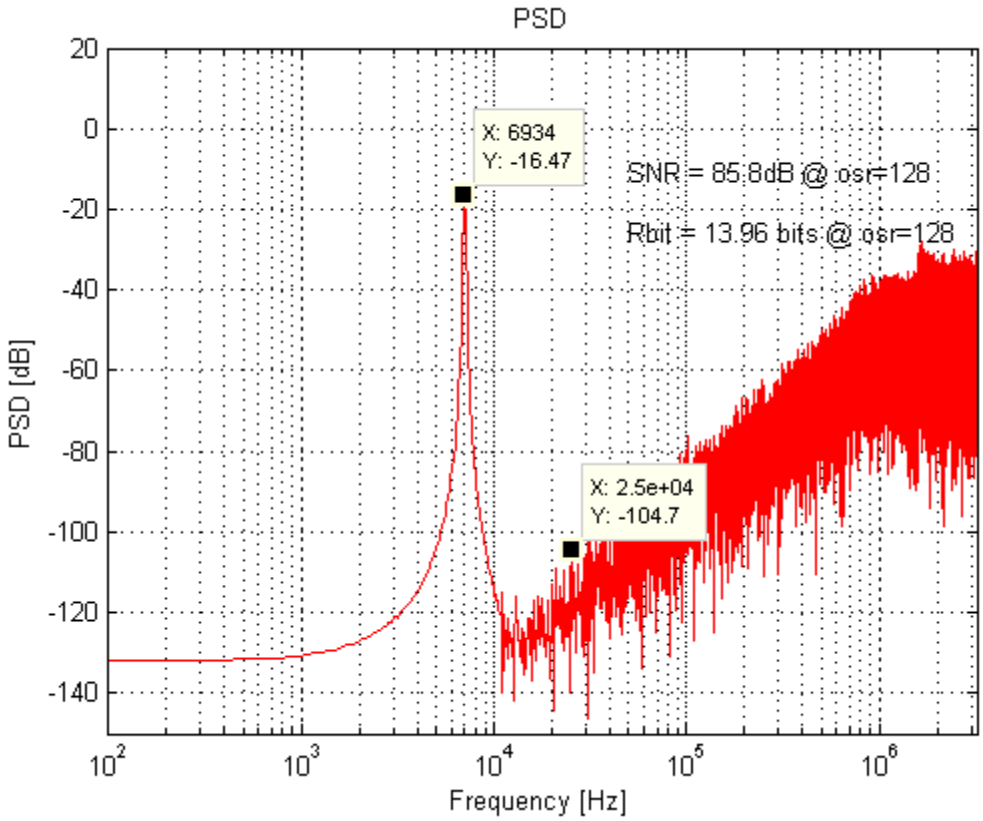


Figure 3.8. FFT of the CT Sigma Delta output.

The integrators can be realized via several methods. Primary choices for the integrators are gm-C and active RC implementations (Figure 3.9). Additionally, current mode integrators can also be used. In active RC configuration, the time constant can vary up to 30 % because of process variations. Special care is needed to reduce possible variations. Active RC configuration has better linearity than gm-C integrators; however, it is difficult to set resistor and capacitance values. For a certain range of linearity, gm-C integrators are favorable as filters, if low power consumption is a major interest [9]. Therefore, gm-C filter (Figure 3.9b) were chosen in order to implement 2nd order CT Sigma Delta modulator. Transfer function of the gm-C integrator is,

$$I(s) = \frac{gm}{sC} \tag{3.8}$$

Using (3.8) with (3.4) and (3.5), one obtains,

$$f_s = \frac{gm}{C} \tag{3.9}$$

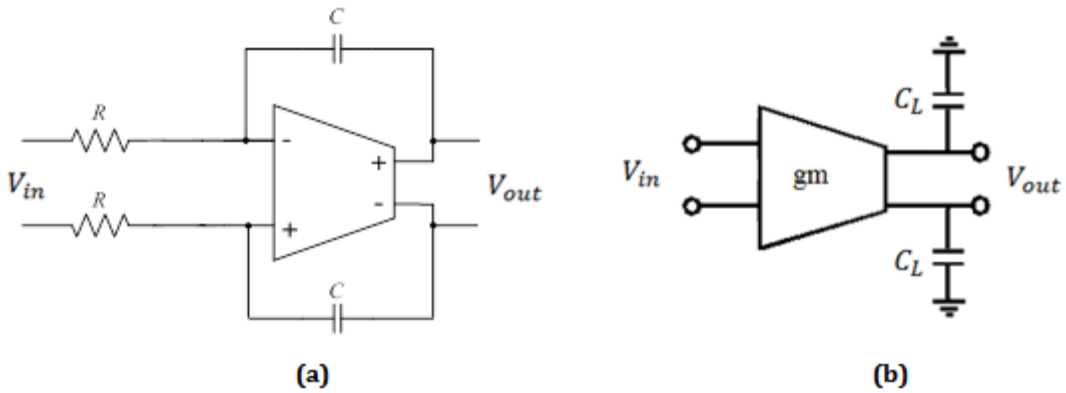


Figure 3.9. Integrator structure of (a) active RC and (b) gm-C configurations [9].

For the condition where two gm-C filters in the modulator have different coefficients, Equation (3.9) becomes,

$$f_s = \frac{gm}{aC} \tag{3.10}$$

where a is the integrator coefficient.

3.2.1 gm-C Integrator Design

The active element in Figure 3.9b is called operational transconductance amplifier (OTA). Its output current is found by,

$$I_{out} = V_{in} gm \tag{3.11}$$

and its transfer function by,

$$V_{out} = \frac{V_{in} gm}{sC} \tag{3.12}$$

The design of the gm-C integrator is dependent on the sampling frequency. As mentioned before, the Sigma Delta ADC design of interest in this work is intended for audio applications; therefore, BW of the input signal is 25 kHz. For an OSR of 128,

sampling frequency becomes 6.4 MHz. Because of area considerations, values of the capacitances were chosen to be between 100fF and 500fF. Then, Equation (3.9) becomes,

$$0.64\ \mu S \leq gm \leq 3.2\ \mu S \tag{3.13}$$

For an average value of the gm, say 1µS, the required quiescent current of the transistors in the (OTA) is,

$$I_Q = 2V_{ov}gm \approx 100nA \tag{3.14}$$

In Equation (3.14) V_{ov} is the overdrive voltage and it is in a certain range for a technology. For 0.18µm technology it is between 0.2-0.3V and in (3.14) it is taken as 0.2V. The obtained I_Q is very low and it is difficult to design an OTA having a good gain with it. Another problem is that the gm in a single OTA is not very linear. Such non-linearity in a modulator causes performance degradation; therefore, gm of the OTA should somehow be linearized.

In [28-31] some linearization techniques and linearized OTA structures are given. Most of them are for linear current conversion and have very low gain. However, in our application, the gain of the OTA should be high in order to prevent noise leakage.

Power consumption of the OTA is also very important because most of the power is consumed in the integrators in the modulator. Therefore, in order to achieve low power OTA, telescopic cascode configuration is chosen. It has the fewest branch as for an acceptable gain for 0.18 µm technology. In order to linearize the OTA, source degeneration technique is utilized. In Figure 3.10, the schematic of a telescopic cascode OTA is depicted. Figure 3.10 shows the applied source degeneration. Resistances are connected to the sources of the input transistors such that in DC condition they have no effect but in AC analysis gm of the OTA is linearized in the expense of decreased gain.

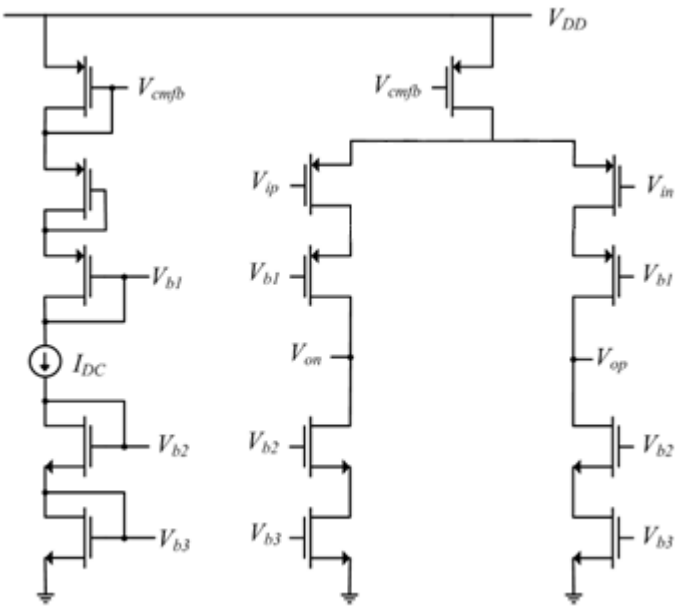


Figure 3.10 Telescopic Cascode OTA.

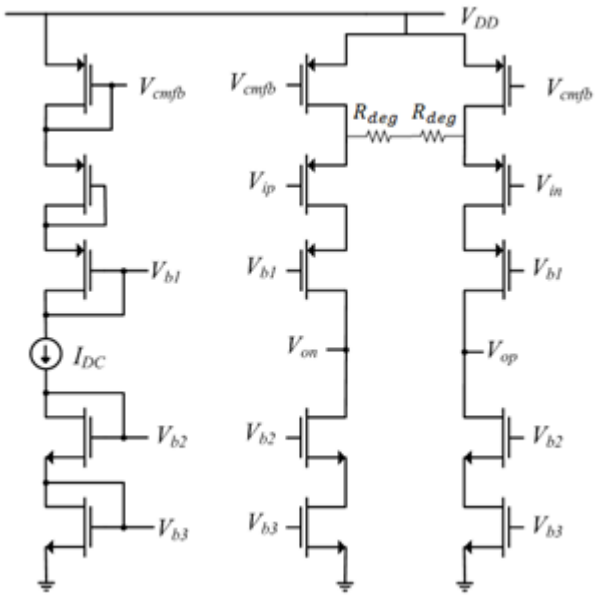


Figure 3.11 Telescopic OTA with source degeneration resistances.

If the new gm of the circuit in Figure 3.11 is called gm_{eff}, then:

$$\frac{1}{gm_{eff}} = \frac{1}{gm} + R_{deg} \tag{3.15}$$

If $gmR_{deg} \gg 1$ then the Equation (3.15) can be approximated to,

$$gm_{eff} = \frac{gm}{gmR_{deg} + 1} \approx \frac{1}{R_{deg}} \tag{3.16}$$

There is a tradeoff between linearization and the gain of the OTA. When R_{deg} is increased the OTA gets more linear, but gain decreases. According to simulations, the gain should be above 40 dB in order to prevent noise leakage. Gain is chosen to be around 45 dB in our design. The OTA in Figure 3.10 has a gain around 70 dB with a gm of 6.2 µS. In this situation, a resistor value of 800 kΩ allows achieving 45 dB gain and also gmR_{deg} product is around 5 which is not very good value for the assumption of $gmR_{deg} \gg 1$ for (3.16). However any resistance value higher than that is not only difficult to implement area wise, but also decreases gain to a value that is intolerable. Simulation results show that 800 kΩ is sufficient for the desired performance of the Sigma Delta modulator.

The designed OTA operates with a V_{dd} of 1.8 V and its common mode (CM) voltage is 0.75 V. In order to stabilize the output around CM, the OTA requires common mode feedback (CMFB) circuitry since it is fully differential. The difference between the DC level of the output of the OTA and the CM voltage is fed back to current sources at the upper part of the OTA where represented as V_{cmfb} in Figure 3.10 and Figure 3.11. A simple low gain one stage differential amplifier is suitable for this application and the circuit in Figure 3.12 is used for this purpose.

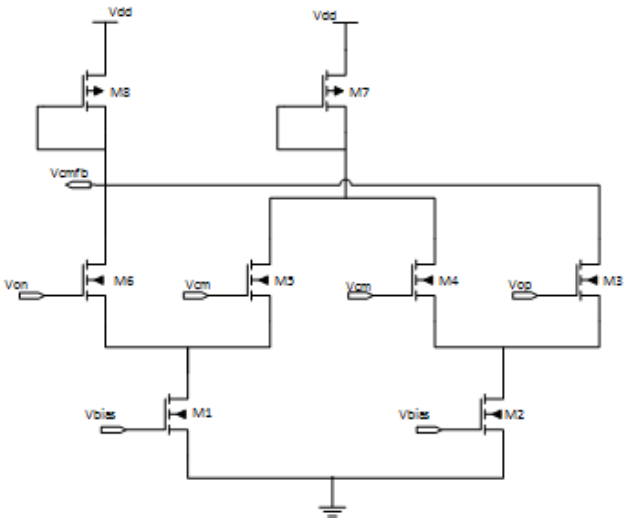


Figure 3.12. Common mode feedback circuit.

AC analysis result of the OTA with a 500fF load capacitance is given in Figure 3.13 and two tone test results for linearity analysis are depicted in Figure 3.14. In this test, two inputs with different frequencies (23 kHz and 27 kHz in Figure 3.14) form an output waveform which has some harmonics. Since the structure is fully differential, only odd harmonics are important. Among the odd harmonics, third harmonics are the main concern and they can be investigated in Figure 3.14. There is 120 dB difference between signals and their third harmonics. Additionally, some important features of the OTA are provided in Table 3.2.

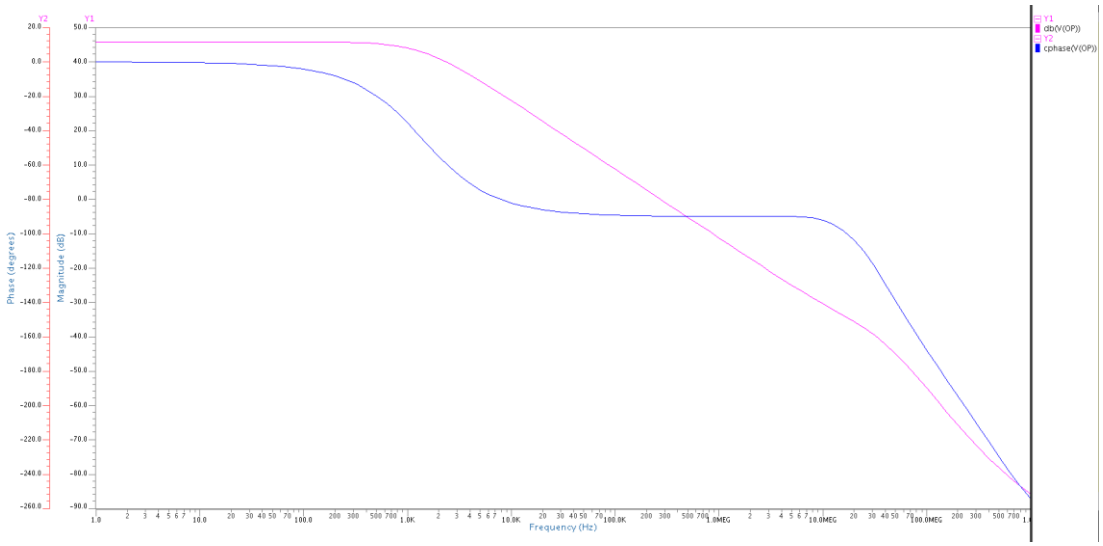


Figure 3.13. Magnitude and phase response of the OTA.

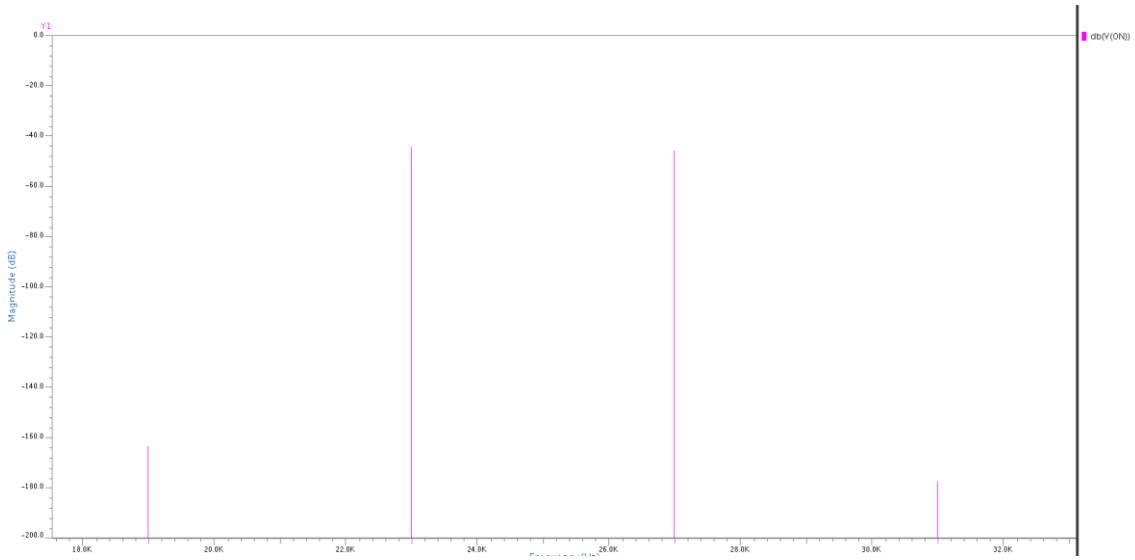


Figure 3.14. Two tone output signals and their third harmonics.

Table 3.2. OTA parameters.

Parameter	**Value**
Gain	45 dB
Power	2.89µW
Gain Band Width (GBW)	278 kHz
Phase Margin	90^{o}
Load Capacitance	500f
Power Supply	1.8V

3.2.2 Comparator

The CT Sigma Delta modulator design is 1 bit; therefore, the quantizer consists of a single comparator. Sampling of the signal is also performed at this stage; thus it should be a latched comparator. For this purpose, comparator in [32] is chosen. Its circuit schematic is illustrated in Fig 3.15.

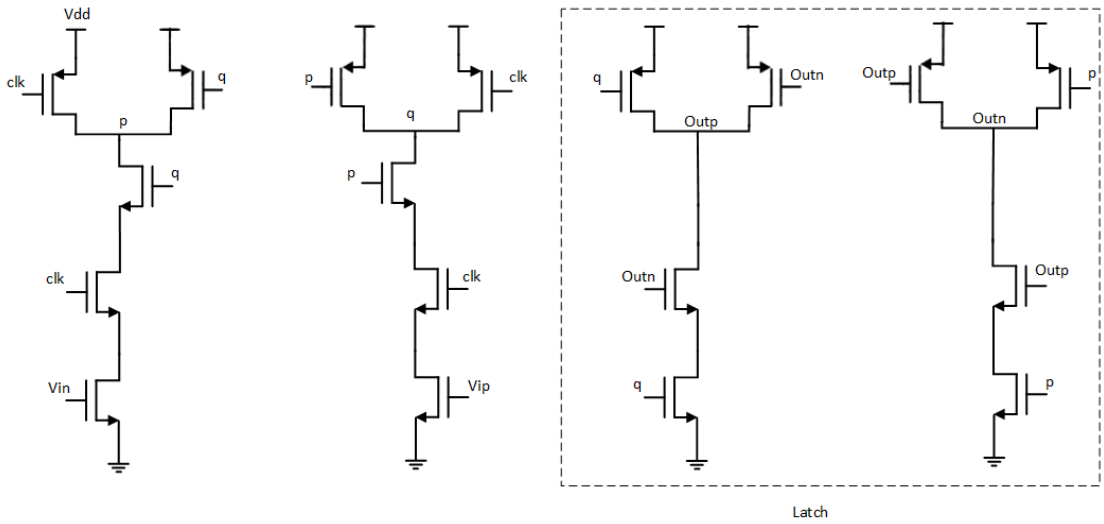


Figure 3.15. Schematic of the comparator.

“Vin” and “Vip” are the negative and positive inputs respectively. “Outn” and “Outp” are their corresponding outputs. The comparator can operate with a Vdd down to 1.2V; however, it is operated at 1.8 V in the modulator in order to speed up the comparison. Transient time simulation results of the comparator are provided in Figure 3.16 as an

example. Two differential sinusoidal inputs are given and p, q, outn, and outp are illustrated.

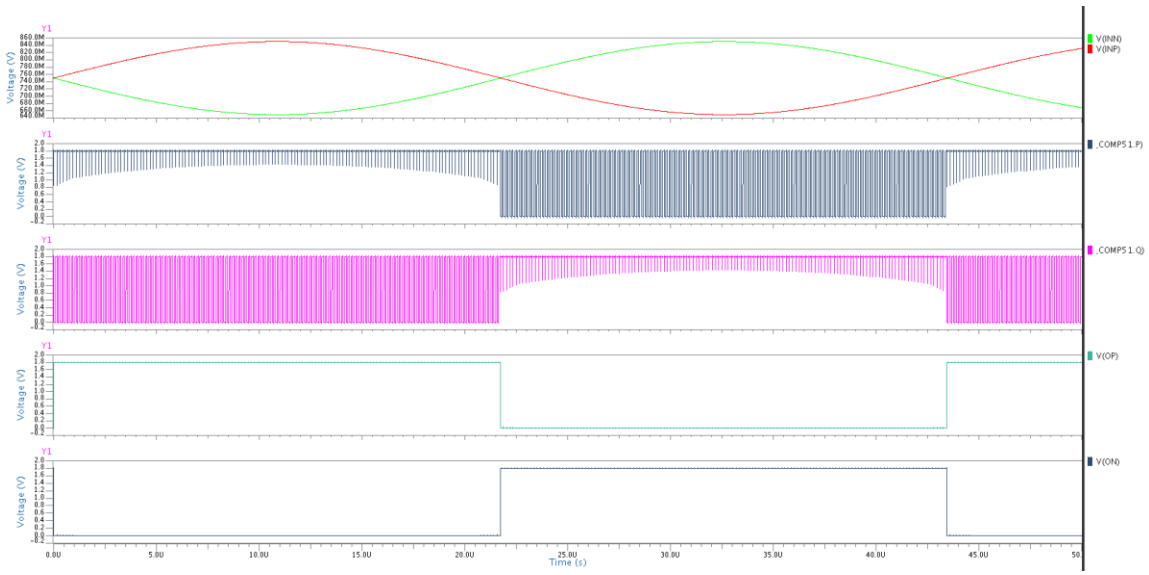


Figure 3.16. Transient simulation of the comparator.

3.2.3 DAC Design

For the DAC, NRZ pulse feedback is chosen because of its ease of implementation. The pulse is of the form shown in Figure 3.4 with $\alpha=0$ and $\beta=1$. Circuit implementation of the NRZ DAC feedback is very simple. DAC feedback implementation is depicted in Figure 3.17. In the actual circuit, switches are pass transistors controlled by outputs of the comparator and current sources in Figure 3.17 are realized by using transistors with simple current mirrors.

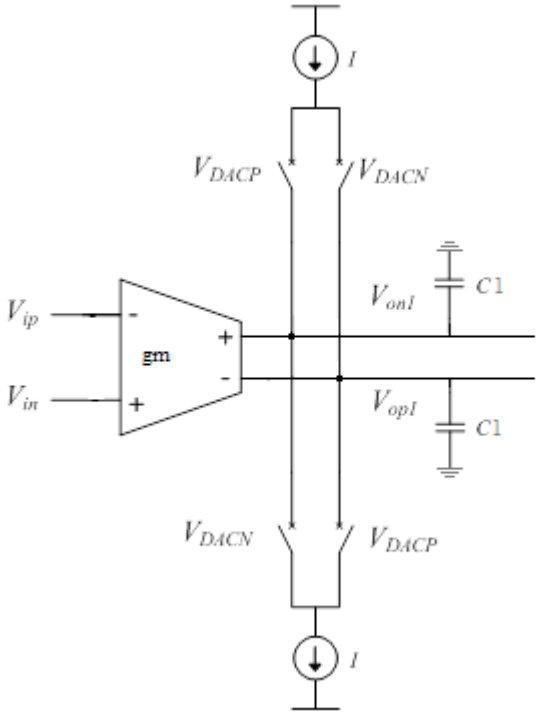


Figure 3.17 Implementation of DAC with feedback.

3.2.4 CT Sigma Delta modulator Realization

After designing each block according to the models in Figures 3.6 and 3.7, the full schematic of the modulator was designed as in Figure 3.18. In MATLAB Simulink simulations (Figure 3.7), coefficients of the integrators were found as 0.33 and 1 for the first and second integrators, respectively. According to Equation (3.10) C_1 and C_2 capacitances in Figure 3.18 are 395f F and 120f F. Signal bandwidth is 25 kHz and the sampling frequency becomes 6.4 MHz with an OSR of 128. Introduction of the integrator coefficients change the transfer function and the feedback currents of the first and second stage. Formerly, they were *I* and *1.5I* respectively, but they become *I* and *0.5I*. Current values are determined by FFT simulations and the best SNR is obtained by applying 120nA to the first stage and 60nA to the second stage. FFT of the output in this configuration is depicted in Figure 3.19. SNR of the output signal is 82 dB and ENOB is 13.3.

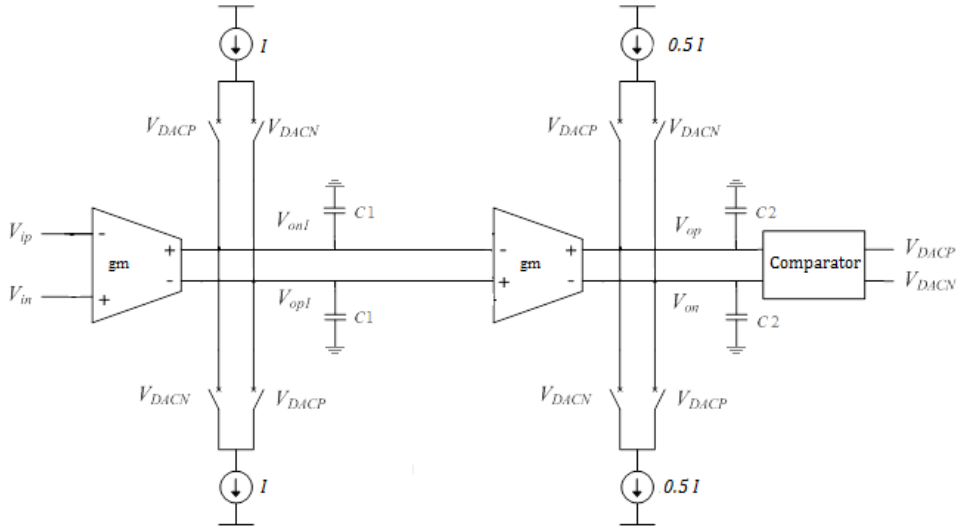


Figure 3.18. Schematic of the modulator.

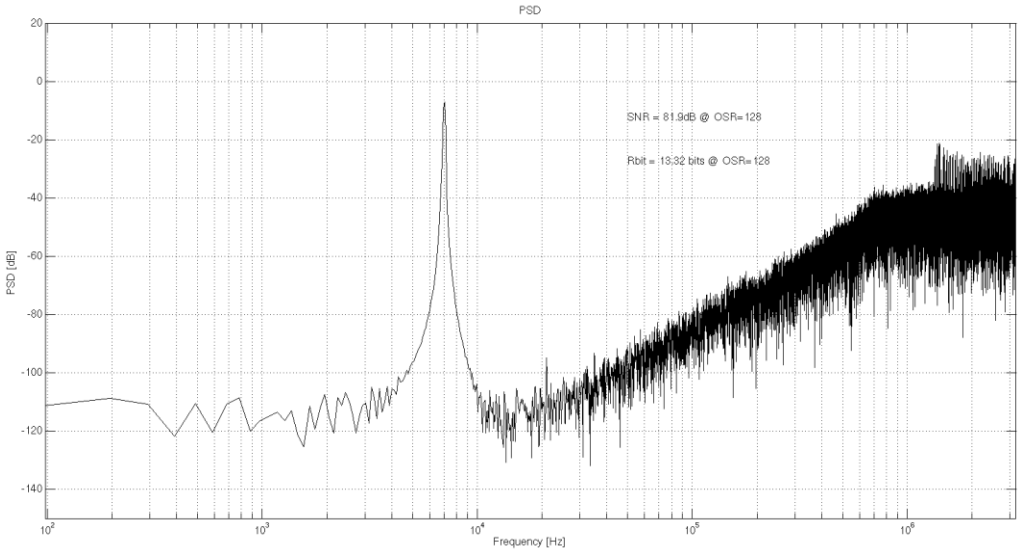


Figure 3.19. FFT of the modulator output for distributed feedback configuration.

In addition to distributed feedback, feedforward configuration was also tested. Feedforward allows decreasing the voltage swings at the outputs of the integrators, which in turn is a desired condition when linearity range of the OTA is considered. The schematic of the feedforward topology is illustrated in Figure 3.20.

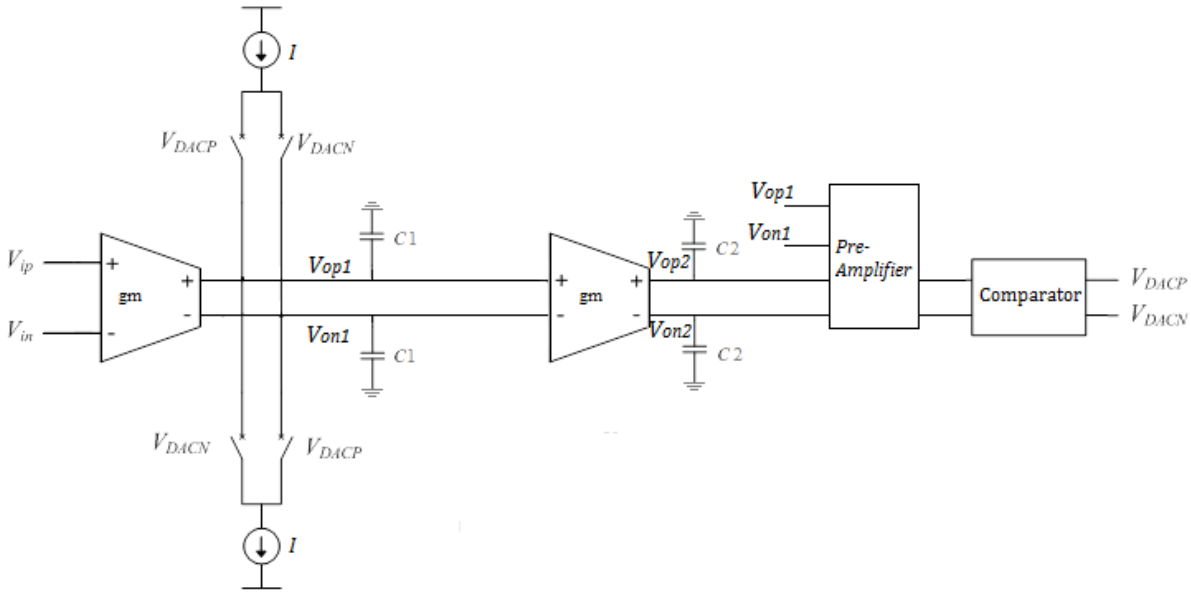


Figure 3.20 Schematic of feedforward configuration of the modulator.

The pre-amplifier in the schematic has exactly the same circuit structure as the CMFB amplifier in Figure 3.12. The pre-amplifier prevents kick-back noise at the comparator and amplifies the signal such that comparison delay and error are decreased. It also performs subtraction operation for feeding signal from the output of the first stage to the output of the second stage while isolating outputs from each other. In feedforward configuration, feedback current is 100nA. Simulations are conducted in MATLAB Simulink tool. Integrator coefficients are 0.3 and 0.5 for the first and the second stage respectively. Corresponding capacitances were found to be 450f F and 260f F. However in Eldo spice tool of Mentor Graphics the best results are obtained from capacitance values of 475f F and 270f F. Output FFT of the modulator is shown in Figure 3.21. The result in Figure 3.21 is for the ideal DAC pulse. Feedforward configuration has 5 dB better SNR value which corresponds to 0.8 bits increase in ENOB. Then, ideal DAC was replaced with a real one and simulated. The obtained circuit simulation result is depicted in Figure 3.22. There is about 2.5 dB SNR loss when the ideal DAC is replaced. The reason is that when switches and current sources are replaced with transistors, the currents for upper and lower parts did not become equal because of varying V_{DS} of the current sources. In addition, DAC pulses create current on parasitic capacitances of the switch transistors which also affects the performance.

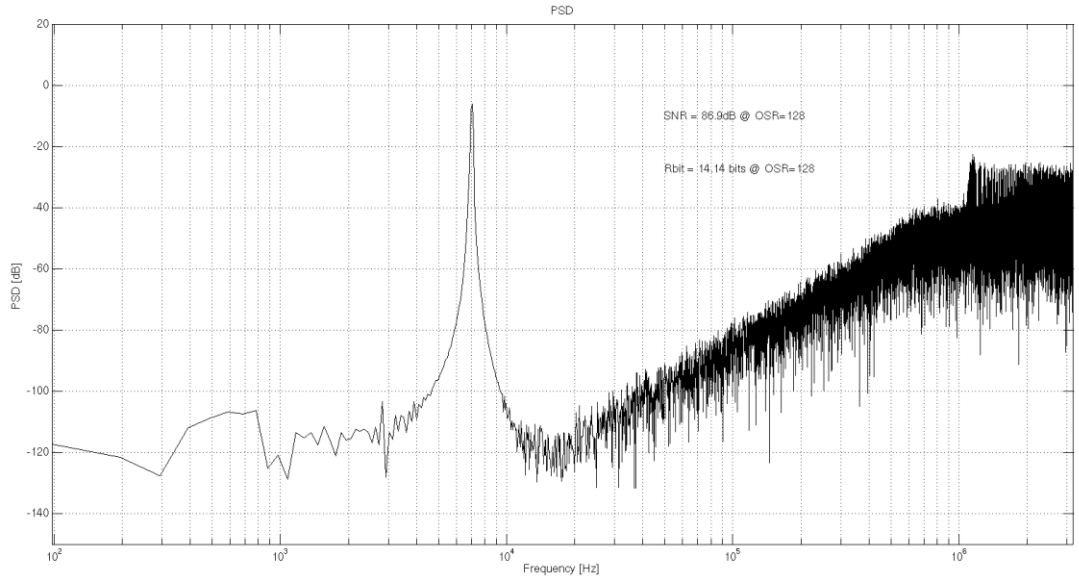


Figure 3.21 FFT of the modulator output for feedforward configuration.

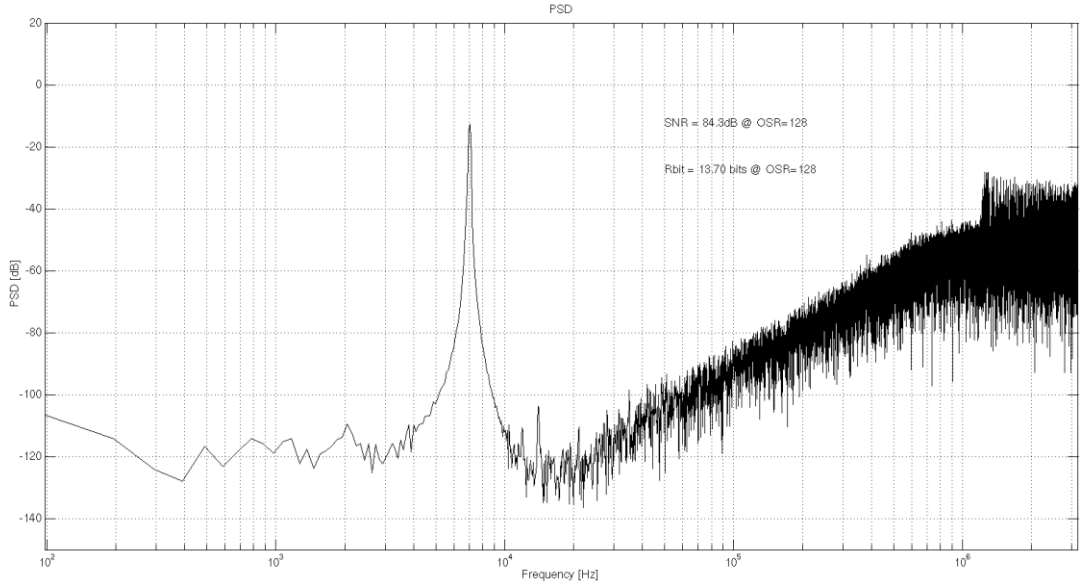


Figure 3.22 FFT result of feedforward configuration with real feedback implementation.

Figure of merit (FoM) for the modulator is calculated according to:

$$\text{FoM} = \frac{\text{Power}}{2BW2^{\text{ENOB}}} \tag{3.17}$$

Power consumption of the modulator is 9.82 µW, ENOB is 13.7 and BW is 25 kHz. Using Equation (3.17), the corresponding FoM becomes 14.7 fJ/con. In the literature, very good

designs have FoM between 5 and 10 fJ/con. According to value we have found, it can be asserted that, the designed modulator has fairly good FoM.

4 DECIMATION FILTER

4.1 Introduction

ΣΔ ADCs are widely used, because they are suitable for applications which require high speed and high resolution. Reducing the power consumption of an ADC is a very significant design problem. A ΣΔ ADC consists of two main blocks: analog modulator and digital decimation filter. Concerning low power ADC design, most of the effort is being spent on the analog part of the converter. As stated in [4], the digital decimation filter can also consume as much power as the analog part.

In this thesis, several low power digital filter design techniques are combined to design a decimation filter for a certain ΣΔ ADC intended for audio applications. From the results, it can be seen that combining all of these techniques can yield a fully synthesizable structure whose power consumption is less than a typical low power modulator.

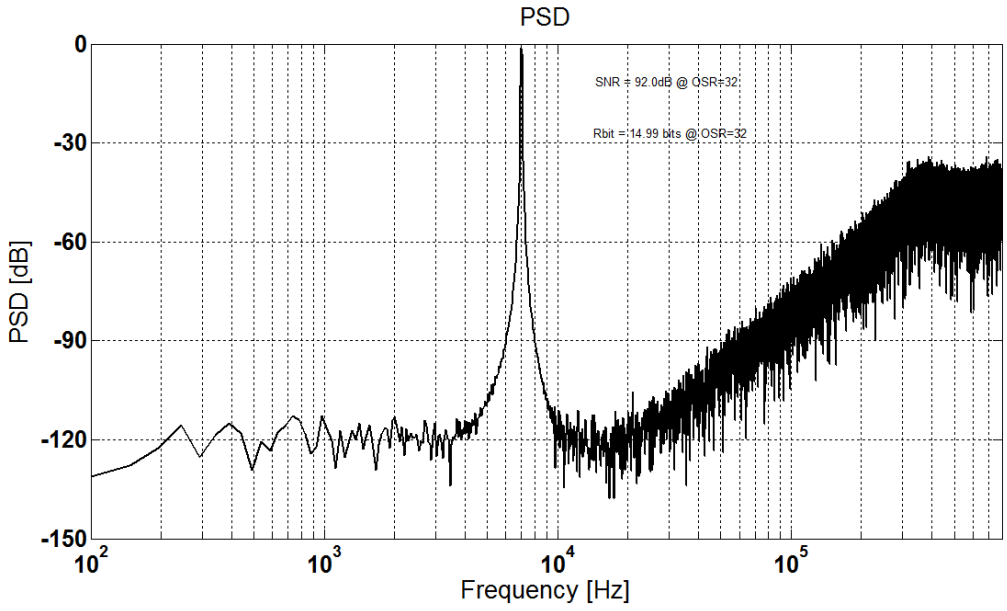


Figure 4.1. PSD of the 3-stage 3-bit ΣΔ modulator output.

The analog modulator of the DT ΣΔ ADC is a 3-stage 3-bit structure with a sampling frequency of 1.6 MHz (32 oversampling ratio) for 25 kHz input bandwidth. It can achieve

ENOB around 15 bits according to circuit simulation. Power spectral density (PSD) of the modulator is shown in Figure 4.1. Specifications of the decimation filter are determined according to the PSD in Figure 4.1. The main goal is to eliminate the out of band noise, shaped by the modulator, without any significant SNR degradation. In addition, power consumption of the Decimation filter should be less than that of the analog modulator which has power consumption around 15 μW.

Decimation consists of two main steps. One is to reduce the high frequency signal components with digital low pass filter, and the second is to reduce the sampling rate of the signal. For instance, if down-sampling ratio is N, only the Nth sample of the signal is kept while the others are discarded. Down-sampling alone causes high frequency signal components to be misinterpreted by subsequent users of the data, which is a form of distortion called aliasing. In the first step, out of band signals (noise) are suppressed so that they do not degrade SNR through aliasing. In the second step, every Mth sample is kept and rest are discarded for decimation by M. In order words, if the signal has F sample/sec data rate, it goes through a register which operates at F/M Hz frequency. Therefore frequency of the signal is decimated by M. Decimation procedure can be incorporated into the filter operation.

One possibility for the filter choice is a CIC filter. It is first mentioned in Hogenauer's article [33]. Since then, it is widely used in Decimation filters. It has a simple structure and can efficiently achieve high decimation ratios. However its noise suppression is insufficient and it creates a droop in the passband. Therefore, it cannot be used alone for high resolution applications. On the other hand, it is a very good candidate as the input stage of multistage structures. In this thesis, the decimation filter is multistage and CIC is used as the first stage; therefore, it will be investigated thoroughly later.

The decimation filter is simple a low pass filter. Hence, it can be implemented in a single stage with IIR or FIR filters. The FIR filter has a linear phase response; therefore, it can achieve flat group delay [4]. Flat group delay is very important for audio signals, because it preserves envelope of the signal. FIR filters are inherently stable and decimation can be simply incorporated. Therefore, in this thesis, FIR filter implementation is preferred.

4.2 Decimation filter Architecture

The aim of decimation filter is reduce the sampling frequency to a Nyquist rate of 50 kHz by preserving the SNR as much as possible. On the other hand, power consumption of the decimation filter should be comparable to or less than that of the DT Sigma Delta modulator which is around 15µW. ΣΔ Modulator specifications are summarized in Table 4.1. According to these specifications and PSD of the output, characteristics of the decimation filter are provided in Table 4.2.

Table 4.1. ΣΔ Modulator Specifications.

Parameters	**Values**
Bandwidth (BW)	25kHz
Sampling Frequency (F_s)	1.6 MHz
Oversampling Ratio (OSR)	32
Modulator Order (L)	3
Number of Bits	3
Signal to Noise Ratio	92dB
Effective Number of Bits	15

In order to meet the decimation filter specifications in one stage, a standard 581 tap FIR filter is required. It is impractical to design such a filter since it takes a large area and it will consume much more power than the analog modulator. Therefore, in order to minimize size and power consumption of the Decimation filter, the filter is implemented in three main stages. Architecture of the filter with frequencies at the nodes is shown in Figure 4.2.

Table 4.2. Decimation filter Specifications.

Parameters	**Values**
Sampling Frequency (input)	1.6 MHz
Decimation Ratio (M)	32
Passband Frequency (F_{pass})	25 kHz
Stopband Frequency (F_{stop})	32 kHz
Passband Ripple (A_{pass})	0.01
Stopband Attenuation (A_{stop})	60dB

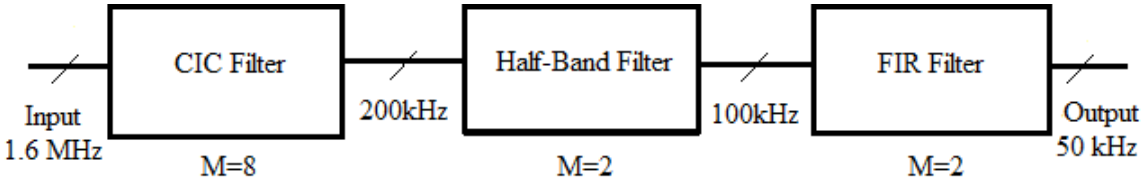


Figure 4.2. Architecture of the 3-stage Decimation filter.

CIC filter is used as the first stage because it has a simple structure that can largely reduce the sampling frequency; however, minimum attenuation in the stopband is insufficient [34]. In addition it creates a large droop in the pass band when decimation is chosen close to OSR. Therefore a decimation factor of 8 is chosen for the CIC filter. Any value higher than that causes considerable droop in the pass-band. Because of the limitations of the CIC, some other filter structures should be used as the later stages. At this stage, Half-Band (HB) filter is a plausible candidate. It is a special type of FIR filter capable of decimation by 2. However, it does not have adequate noise suppression in order to prevent aliasing when the signal is decimated to its Nyquist frequency. Therefore in terms of area, power and performance, it is more effective to use two stages following the CIC filter.

The design of FIR filter and HB filter will be investigated in Section 4.3. Then, the CIC filter will be presented. This way, low power techniques can be explained in a more orderly manner and it will be easier to follow through the design of the filters.

4.3 FIR filters

4.3.1 Introduction

The general form of a lowpass filter is shown in Figure 4.3. The important parameters are marked on the figure. A_{pass} is passband ripple, f_{pass} is passband frequency, f_{stop} is stopband frequency and $f_c = (f_{pass} + f_{stop})/2$ is the cut-off frequency. A_{stop} is the stopband attenuation, f_s is the sampling frequency and $\Delta f = (f_{pass} - f_{stop})$ is the transition band.

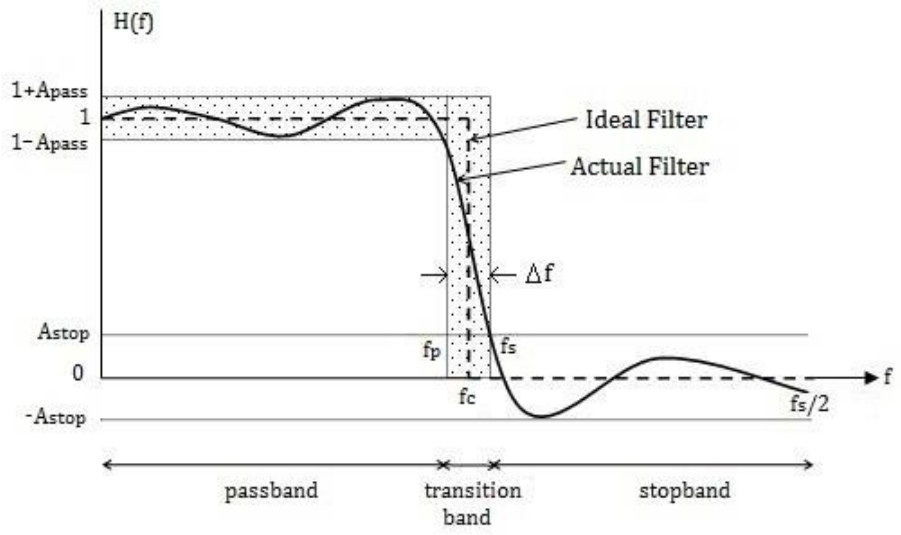


Figure 4.3. Frequency response of a digital filter [59].

The time domain impulse response of the ideal filter is found by,

$$h(t) = 2f_c sinc(2f_c t) \tag{4.1}$$

where

$$sinc(x) = \frac{\sin(x)}{x} \tag{4.2}$$

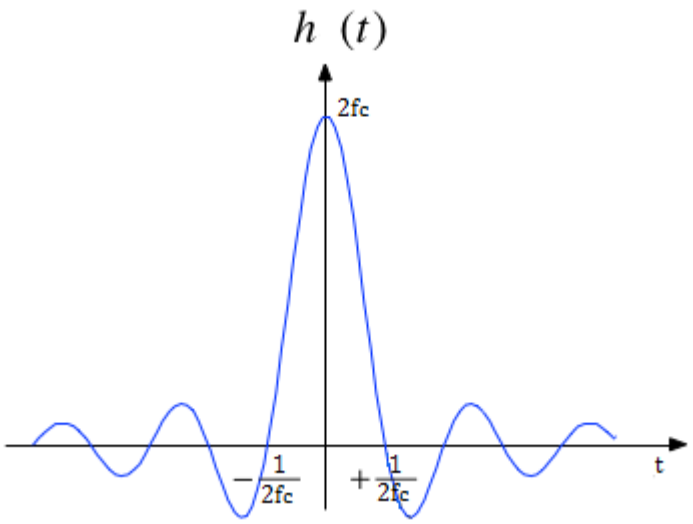


Figure 4.4. Impulse response of lowpass filter.

Impulse response of sinc function is depicted in Figure 4.4. The impulse response of the filter goes to infinity; therefore, it is impossible to implement an ideal filter in real life. The solution is to truncate the impulse response. The price for truncation is obtaining the non-ideal filter in Figure 4.3. As the impulse response gets longer and longer, magnitude response of the filter gets closer to the ideal one.

In discrete time, $h(t)$ becomes,

$$h(nT) = 2f_c sinc(2f_c nT) \tag{4.3}$$

where T is the sampling instances in time domain. Sinc function in DT is represented by sampling the response in Figure 4.4, and resembles Figure 4.5.

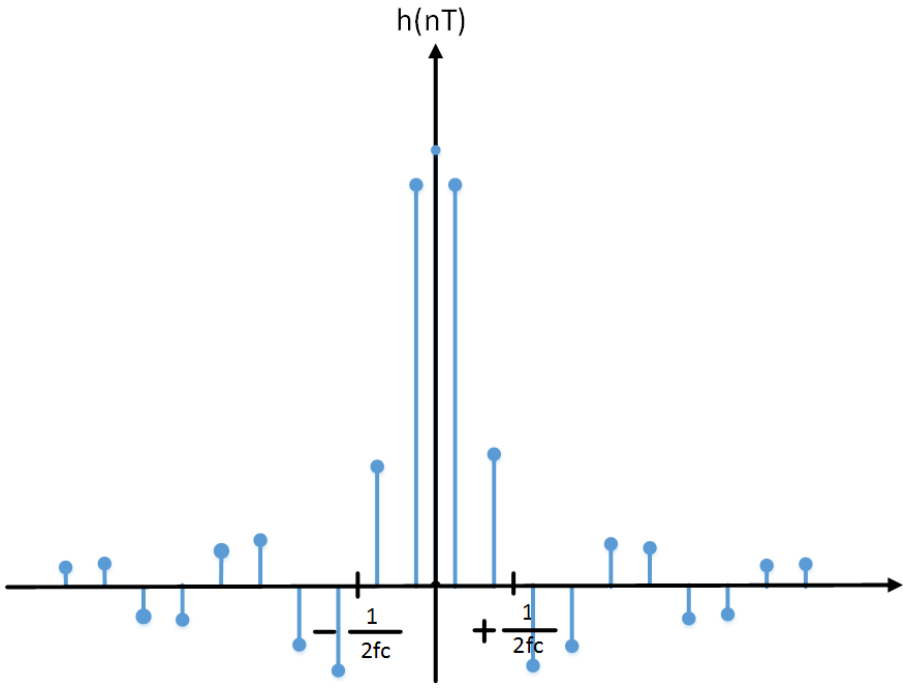


Figure 4.5. DT sinc function waveform.

Sampled data points of the sinc function are the coefficients of the FIR filter. In frequency domain, the signal waveform is multiplied by the magnitude response of the filter shown in Figure 4.3. Multiplication in frequency domain corresponds to convolution in time domain or vice versa. Therefore, the filter operation can be represented as:

$$y(n) = \sum_{k=0}^{N-1} h_k x_{n-k} \tag{4.4}$$

In Equation (4.4), h_k is the k[th] coefficient of the filter, x_n is the input and $y(n)$ is the output of the filter. In z domain, one can obtain:

$$H(z) = \sum_{k=0}^{N-1} h_k z^{-k} \tag{4.5}$$

According to Equations (4.4) and (4.5), the block diagram of the FIR filter is presented in Figure 4.6. A coefficient-delay pair is called a tap.

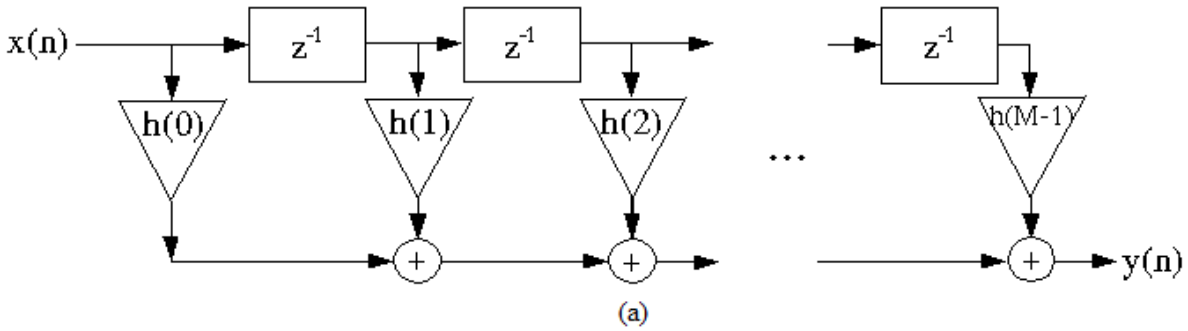


Figure 4.6. Block diagram of FIR filter.

The frequency response of the FIR filter using (4.5) can be written as:

$$H(w) = \sum_{k=0}^{N-1} h_k z^{-jwk} \tag{4.6}$$

Utilizing Euler's formula, Equation (4.6) can also be expressed as:

$$H(w) = \sum_{k=0}^{N-1} h_k coswk - j \sum_{k=0}^{N-1} h_k sinwk \tag{4.7}$$

It is clearer in (4.7) that the FIR filter is periodic and conjugate symmetric. In addition linear phase is observable in (4.6). It can also be observed Figure 4.5 that FIR filters have symmetric coefficients. For N order FIR filter,

$$h_k = h_{N-k-1} \tag{4.8}$$

Filter operation does not depend on the input; therefore, the coefficients are constant and multiplications are constant coefficient multiplication. In this situation, multiplication can be realized by shift and add operation rather than using a multiplier, which consumes large power. Coefficients coded in binary representation consist of ones and the zeros and number of non-zero elements determines the number of additions. Shift operation can be done via hard-wiring which does not account for any power consumption. Therefore, the number of non-zero elements determines the power consumption of the filter with delay elements which are registers.

In this thesis, in order to reduce the power consumption of FIR filters, low power arithmetic and structural solutions are investigated and applied. In the literature, there are many low power design methodologies. For instance, structurally; reducing calculations [35-37], minimizing switching activity [38], polyphase decomposition [39] and arithmetically; Canonical Signed Digit (CSD) representation [40], reducing power consumption of adders [41], reduction of Signed Power of Two (SPT) terms [3] are proposed. In this thesis, several low power techniques are combined to obtain low power FIR filter and HB filter. Utilized techniques are explained in Section 4.3.2.

4.3.2 Low Power Techniques

The Design methods presented below are utilized in both the FIR and HB filters, some of them are also used in the CIC filter.

4.3.2.1 Canonical Signed Digit Representation

Canonical Signed Digit (CSD) is a signed digit number system similar to Booth encoding and it aims to reduce the number of non-zero digits. Other than representing numbers with ones and zeros CSD also utilizes a “-1”. In CSD notation, consecutive

elements cannot be non-zero. Therefore, for an n-bit number, number of non-zero elements is at most n/2, whereas in 2's complement representation, all bits can be non-zero. When constant coefficient multiplication is considered, this feature of CSD is very helpful for decreasing power consumption. As mentioned before, constant coefficient multiplication can be done with shift and add operation. CSD introduces subtraction operation with the extra "-1" element, but greatly reduces non-zero elements. Number of additions and subtractions are one fewer than the number of non-zero elements in a coefficient. Power consumptions of addition and subtraction are practically the same. On average, CSD numbers contain about 33% fewer non-zero digits than 2's complement numbers [42]. One can deduce that this corresponds to around 30% less power consumption.

Table 4.3. Conversion of 2's complement numbers to CSD numbers.

carry-in	x_{i+1}	x_i	**carry-out**	c_i
0	0	0	0	0
0	0	1	0	1
0	1	0	0	0
0	1	1	1	-1
1	0	0	0	1
1	0	1	1	0
1	1	0	1	-1
1	1	1	1	0

Conversion of 2's complement numbers to CSD numbers is summarized in Table 4.3 [43]. 2's complement numbers are represented by $X = x_n^* x_{n-1} x_{n-2} x_{n-3} \dots \dots x_2 x_1$ and CSD numbers are represented by $C = c_{n-1} c_{n-2} c_{n-3} \dots \dots c_2 c_1$. x_n^* is 0 for positive numbers and 1 for negative numbers, it can be obtained by replicating most significant bit (MSB).

In order to comprehend the multiplication operation with CSD numbers, the example below can be studied.

CSD representation of 15 and 28 can be found as,

2's Complement CSD

$$28 = \quad (\,011100\,)_2 \qquad\qquad (\,+00-00\,)_2 \tag{4.9}$$

$$15 = \quad (\,01111\,)_2 \qquad\qquad (\,+000-\,)_2 \tag{4.10}$$

Here "+" represents 1 and "−" represents 0. Multiplication with "28" requires a minimum of 2 additions and similarly "15" requires 3 additions. However, if the numbers 28 and 15 are encoded in CSD, they both require 1 subtraction. (4.11) and (4.12) show the shift and add operations for multiplication with 15 and 28. "<< n" is shift to the left by n times.

$$A * 15 \;=\; (A)_2 \ll 4 - (A)_2 \tag{4.11}$$

similarly,

$$B * 28 \;=\; (B)_2 \ll 5 - (B)_2 \ll 2 \tag{4.12}$$

4.3.2.2 Sub-expression Sharing

Sub-expression sharing [44] is utilized for reducing the number of addition and subtraction operations by using common sequences in coefficients. It can be applied both within a coefficient and between coefficients.

Every coefficient consists of -1, 1, and 0 values when encoded in CSD and even when the coefficients are different from each other, they most probably contain some common sequences. If a constant value is going to be multiplied (by shift and add operation) with these coefficients, there is no need to calculate common sequences more than once.

For example, if some value is to be multiplied by A = $[10\bar{1}00101\bar{1}]$, the sequence $[10\bar{1}]$ should be calculated only once. Calculations are as follows,

$$a_1 = x * [10\bar{1}] \;=\; x \ll 2 - x \tag{4.13}$$

$$a_2 = a_1 \ll 5 \tag{4.14}$$

$$x * A = \; a_2 + a_1 \tag{4.15}$$

As observed through calculations, number of addition operations is reduced. As number of coefficients increase, the probability of common sequences increases. Number of addition and subtraction operations is directly related to power consumption; therefore, sub-expression sharing method helps to reduce the power consumption of a filter considerably.

In order to further illustrate the method, a 6 tap filter example is shown in Table 4.4. According to the table, Equations through (4.16) to (4.21) summarize the multiplication using sub-expression sharing method. Coefficients of the FIR filters are symmetric as mentioned before and numbers 1 to 8 represent the place of digits to the right of the decimal point. Magnitudes of all coefficients are less than 1. x is the input and a, represents intermediate multiplication results with sub-expressions.

Table 4.4. Coefficients of 6 tap FIR filter.

	1	**2**	**3**	**4**	**5**	**6**	**7**	**8**
h_0= h_5	1	0	1	0	0	$\bar{1}$	0	0
h_1= h_4	0	1	0	1	0	1	0	$\bar{1}$
h_2= h_3	0	1	0	1	0	0	$\bar{1}$	0

$$a_1 = x * [101] = x + x \gg 2 \tag{4.16}$$

$$a_2 = x * [10\bar{1}] = x - x \gg 2 \tag{4.17}$$

$$a_3 = x * [10100\bar{1}] = a_1 - x \gg 5 \tag{4.18}$$

$$x * h_0 = a_3 \gg 1 \tag{4.19}$$

$$x * h_1 = a_1 \gg 2 + a_2 \gg 6 \tag{4.20}$$

$$x * h_0 = a_3 \gg 2 \tag{4.21}$$

4.3.2.3 Filter Structure

There are two main structures for the FIR filter; one is the direct form which is shown in Figure 4.7a, the other one is the transposed direct form in Figure 4.7b. According to Equation (4.4), two different structures give exactly the same output, the only difference is the order of delay and multiplication with the coefficients.

They both have some advantages and disadvantages. In direct form, register word lengths are much less than the transposed direct form since registers are placed before the multiplications. However, its critical path consists of one multiplication and adders which amount to a number equal to the order of the filter. Especially for a filter with high order and high frequency, this is very problematic. Also, the gain of having short word length at the registers is lost by glitches along this long combinational path.

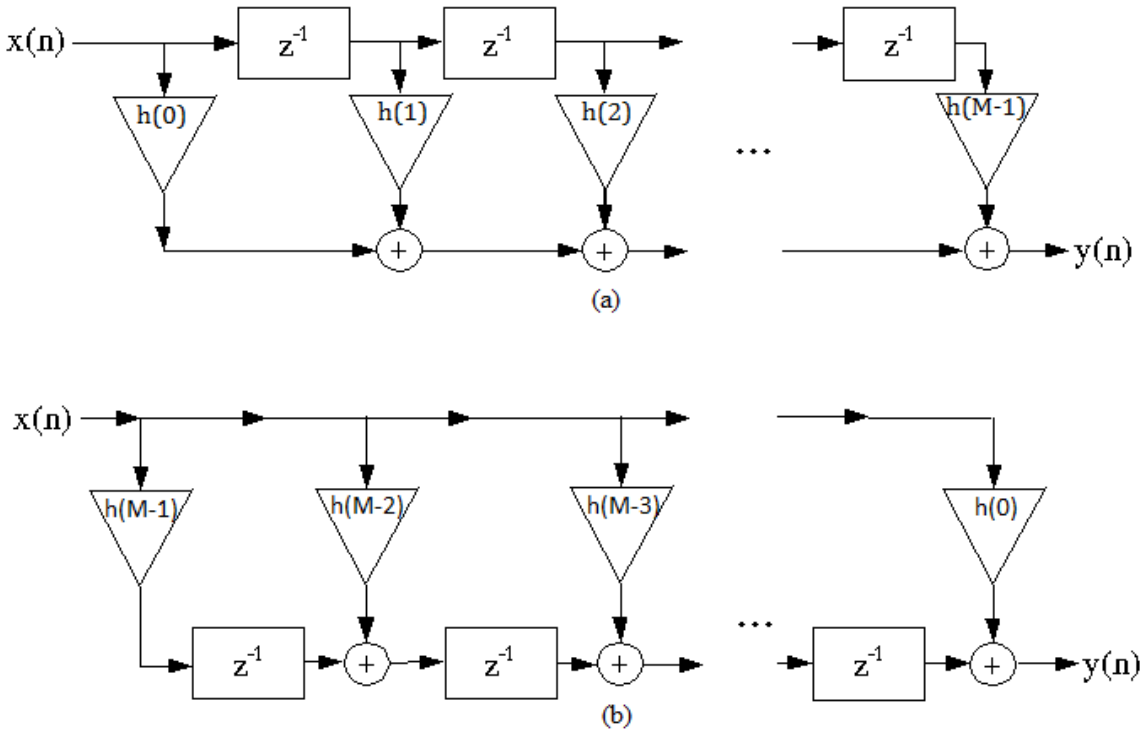


Figure 4.7. Two structures of the FIR filter: (a) direct form, (b) transposed direct form.

On the other hand, the transposed direct form has a shorter critical path, which consists of one multiplication, one delay, and one addition. Since registers are placed

between the adders, it has less glitching. In terms of dynamic power, glitches have considerable impact on power consumption. However, the delay registers are placed after the multiplication; therefore they have increased word length. In direct form word length of the registers is equal to that of the input, however in transposed direct form word length of the registers is sum of the word lengths of input and coefficient.

In [45] both filter structures are investigated and the conclusion is that the transposed direct form is the least power consuming architecture. In addition, transposed direct form allows utilizing sub-expression sharing method and reuse of symmetric coefficients. The reason behind this is that, in direct form delayed input data is different from each other and multiplication is applied to different numbers. However, in transposed direct form, every coefficient is multiplied with same input data; therefore, intermediate multiplication results and symmetric coefficients can be reused. This makes transposed direct form an even better alternative when power consumption is the main issue.

4.3.2.4 Polyphase Decomposition

Normal operation of the filter looks like the diagram in Figure 4.8 where decimation follows the filter. Incorporation of decimation into the filter is mentioned as the advantage of the FIR filter. This can be done following the equations below, starting from the definition of the FIR filter.

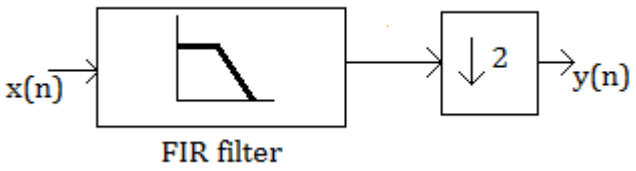


Figure 4.8. Basic block diagram of FIR filter with decimation.

FIR filter transfer function is given by,

$$H(z) = \sum_{k=0}^{N-1} h_k z^{-k} \tag{4.22}$$

Output of the filter is obtained by multiplication of the filter transfer function and the input in z domain:

$$Y(z) = X(z) * \sum_{k=0}^{N-1} h_k z^{-k} \tag{4.23}$$

Filter can be separated into terms with even and odd number of delays (z^{-1}):

$$Y(z) = \sum_{k=0,2,4\ldots}^{N-1} X(z) h_k z^{-k} + \sum_{k=1,3,5\ldots}^{N-2} X(z) h_k z^{-k} \tag{4.24}$$

Here, filter order N is considered to be odd. Equation (4.24) can also be written as:

$$Y(z) = \sum_{k=0,2,4\ldots}^{N-1} X(z) h_k z^{-k} + z^{-1} \sum_{k=0,2,4\ldots}^{N-3} X(z) h_{k+1} z^{-k} \tag{4.25}$$

In more compact format,

$$Y(z) = X(z) E_0(z^2) + z^{-1} X(z) E_1(z^2) \tag{4.26}$$

where,

$$E_0(z) = \sum_{k=0,1,2\ldots}^{(N-1)/2} X(z) h_{2k} z^{-k} \quad and \quad E_1(z) = \sum_{k=0,1,2\ldots}^{(N-3)/2} X(z) h_{2k+1} z^{-k} \tag{4.27}$$

E_0 contains coefficients with even indices and E_1 with odd indices. The new block diagram of the filter is depicted in Figure 4.9.

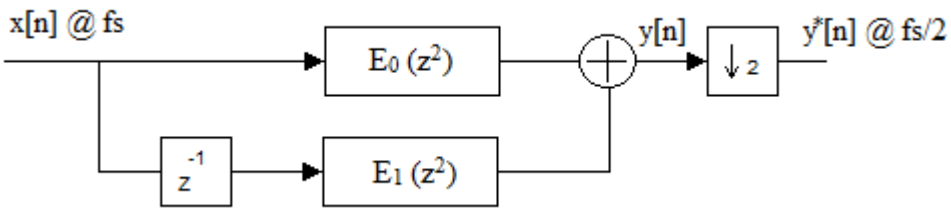


Figure 4.9. Block diagram of FIR filter with coefficients with odd and even indices separated.

"z^{-1}" terms represent one sample delay of data. In digital domain, one sample is determined by one clock cycle of the sampling frequency. For the sake of simplicity, z^{-1} is called one delay, z^{-2} is called two delays and so on. Practically thinking, one can simply comprehend that two delays at frequency f_s is equal to one delay at frequency $f_s/2$. Following this fact, decimation by two can be moved to the front end of the filter like in Figure 4.10.

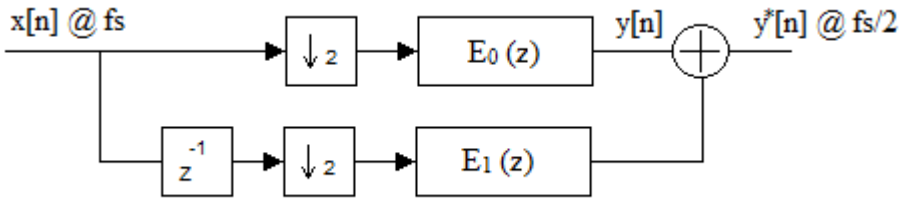


Figure 4.10. Block diagram of FIR filter with decimation moved to the front of the filter.

At the lower side of the block diagram in Figure 4.10, where E_1 function input is delayed once at higher frequency then down-sampled. In order to understand the procedure it is helpful to name adjacent samples of the input data as odd and even terms. In this analogy, even samples go to the upper part where they are multiplied by the coefficients with even indices and odd samples go to the lower part, where similarly they are multiplied by the coefficient with odd indices. A more illustrative example can be seen in Figure 4.11.

This process is called polyphase decomposition. Here, it is explained for decimation by 2 however it can be applied to FIR filters with any value of decimation factor. In digital circuits dynamic power accounts for most of the power consumption; static power on the

other hand makes up only a small portion of the overall power consumption. Average dynamic power consumption is found by,

$$P_{dyn} = \frac{1}{2} f C V_{dd}^2 \tag{4.28}$$

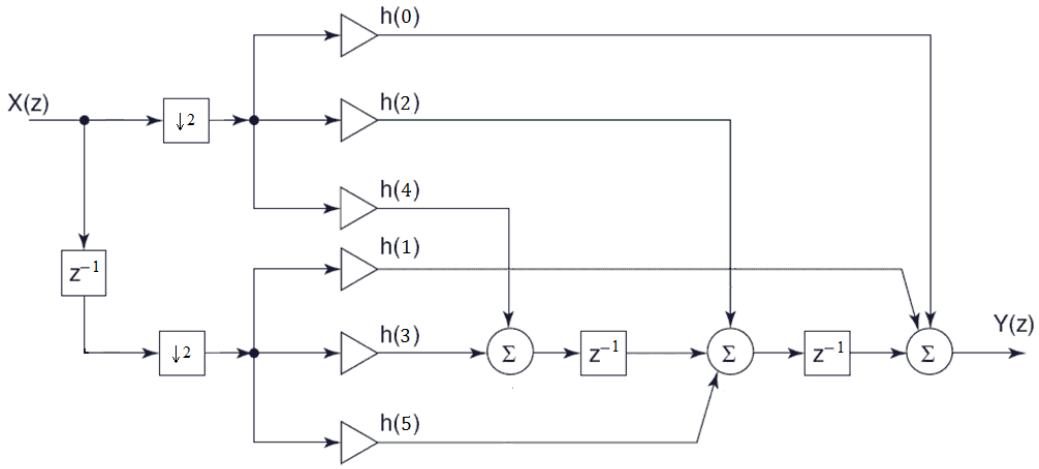


Figure 4.11. Polyphase decomposition of 6 tap FIR filter that has transposed direct form.

As (4.28) suggests, power consumption is proportional to the frequency of operation. After polyphase decomposition, unit delay (registers) at the front of the filter is the only elements operating at high frequency, f_s Figure 4.10. Operating frequency of almost the entire filter is halved; therefore, power consumption of the filter is halved. Without compromising anything, half of the power is saved.

Utilizing polyphase decomposition enables the designer to use half the frequency at all the components except the unit delay at the input side before decimation. This unit delay operates at un-decimated frequency. If the bit length of the input signal is high, this situation affects power consumption of the filter. In addition, specifically for this thesis, the intended DT Sigma-Delta Modulator is realized by using double sampling and an OSR of 32 resulting in an 800 kHz signal. At the input stage polyphase structure is used. Therefore, the only place where 1.6 MHz clock signal is used is the unit delay. However, for the unit delay, instead of using positive edge of the higher frequency clock signal, negative edge of clock operating at half the frequency can be used [46]. For instance, instead of using positive edge of 1.6 MHz signal for registers used as delay blocks, negative edge of the 800 kHz clock can be used. Unit delay should be defined in terms of the higher frequency clock

signal and it is clear that the interval between the positive edge and the negative edge of the halved frequency clock signal is equal to the unit delay. By this way, power is saved both by eliminating need for the highest clock frequency and operating the delay blocks at the input side with halved frequencies. The latter has a more pronounced effect when the bit length of the input is high.

4.3.2.5 Word length truncation

In the filters, the input signal is multiplied by the coefficients and the results are summed. After multiplication, word length of the results is the sum of word lengths of the multiplicands. Addition operation increases the word length by one bit. With all the coefficient multiplications and summations data word lengths may increase to a value that is much more than the required resolution. In order to get rid of the excessive hardware sum least significant bits (LSB) should be truncated. Up until some point, the effect of the truncation on the filter response is minimal but after a certain point it degrades performance considerably. There is a tradeoff between performance and power consumption while truncating word lengths at intermediate nodes in filters. In addition to truncation, resolution of the filter coefficients can be limited to certain digits. For the filter coefficients operation is not truncation but quantization; they are rounded. The same tradeoff also applies for quantization of coefficients. In this thesis, optimum word lengths of the data at different nodes of the filter and coefficient resolutions are determined by Simulink simulations such that while saving power, no considerable SNR degradation occurs at the output of the filter.

4.3.3 Half-Band Filter Design

Half-Band (HB) Filters are specialized filters for decimation by 2 and they have a cut-off frequency of 0.5π when the sampling frequency is shown by 2π in discrete time Fourier Transform (DFT) representation. Because of the place of the cut-off frequency odd coefficients of the HB filter other than the middle coefficient are zero. Practically half of the coefficients are zero; therefore, arithmetic operations are halved and the power consumption as well. However, HB filter is not suitable for decimation to Nyquist rate, since the required passband frequency is 0.5π. HB filter clearly cannot have 0.5π passband frequency.

In this thesis, FIR filters are mentioned before the CIC filter. CIC is at the first stage of the filter, so CIC specifications and the output characteristics of the CIC should be known in order to determine the features of the HB filter. Here, in order to ease understanding of the utilization of the low power techniques which are common for CIC and FIR filters, FIR filters were explained before the CIC filter. The CIC filter is designed first and HB filter specifications are determined according to architecture in Figure 4.2, decimation filter specifications in Table 4.2, and CIC output signal characteristics. Specifications of the HB filter are illustrated in Table 4.5.

Table 4.5. HB filter Specifications.

Parameters	**Values**
Sampling Frequency (input)	200KHz
Decimation Ratio (M)	2
Passband Frequency (F_{pass})	25 kHz
Stopband Frequency (F_{stop})	75 kHz
Passband Ripple (A_{pass})	0.001
Stopband Attenuation (A_{stop})	56dB

Table 4.6. Coefficients of the HB filter.

	Quantized Coefficients	**CSD**	**Power of 2**
h_0	0.012939453125	0.00000+0−0+0+	$2^{-6} - 2^{-8} + 2^{-10} + 2^{-12}$
h_1	0	0	0
h_2	−0.06396484375	0.000−0000−0+0	$-2^{-4} - 2^{-9} + 2^{-11}$
h_3	0	0	0
h_4	0.0301513671875	0.0+0+0−0+0+0−	$2^{-2} + 2^{-4} - 2^{-6} + 2^{-8} + 2^{-10} - 2^{-11}$
h_5	0.5	0.+	2^{-1}
h_6	0.0301513671875	0.0+0+0−0+0+0−	$2^{-2} + 2^{-4} - 2^{-6} + 2^{-8} + 2^{-10} - 2^{-11}$
h_7	0	0	0
h_8	−0.06396484375	0.000−0000−0+0	$-2^{-4} - 2^{-9} + 2^{-11}$
h_9	0	0	0
h_{10}	0.012939453125	0.00000+0−0+0+	$2^{-6} - 2^{-8} + 2^{-10} + 2^{-12}$

HB filter is designed using FDAtool in MATLAB. FDAtool utilizes Parks-McClellan algorithm [2]. The filter is equiripple, which suggests that ripple at the stopband equals to the ripple in the passband. This can also be seen in Table 4.5, where A_{pass} and A_{stop} are more or less equal. Discrepancy between A_{pass} and A_{stop} is caused by quantization of the filter coefficients. Coefficients are shown by their CSD equivalences in Table 4.6. For CSD

in Table 4.6, "+" stands for 1 and "−" stands for −1. In 2's Complement representation coefficients are quantized to 13 bits with signed bit, when they are converted to CSD 12 bits resolution is enough to represent coefficients.

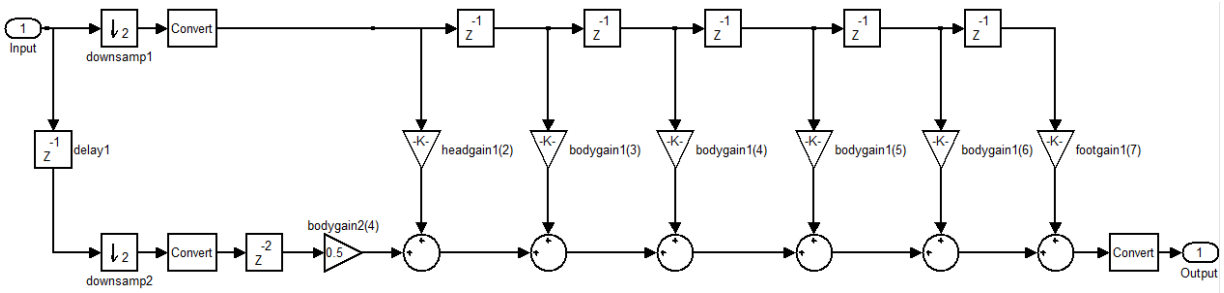


Figure 4.12. Block Diagram of the designed filter in Simulink.

Figure 4.12 illustrates the block diagram of the HB filter in MATLAB Simulink tool. It is in direct form; however, the actual filter is realized in transposed direct form. Simulink does not show filters in transposed direct form. Hence, for simulation purposes, the block diagram in Figure 4.12 is used. Magnitude response of the HB filter is shown in Figure 4.13. On the graph, the frequency is normalized and π represents $f_s/2$ in the figure. Cut-off frequency is 0.5 π and magnitude of the filter at this level is -6 dB, which is characteristic of HB filters.

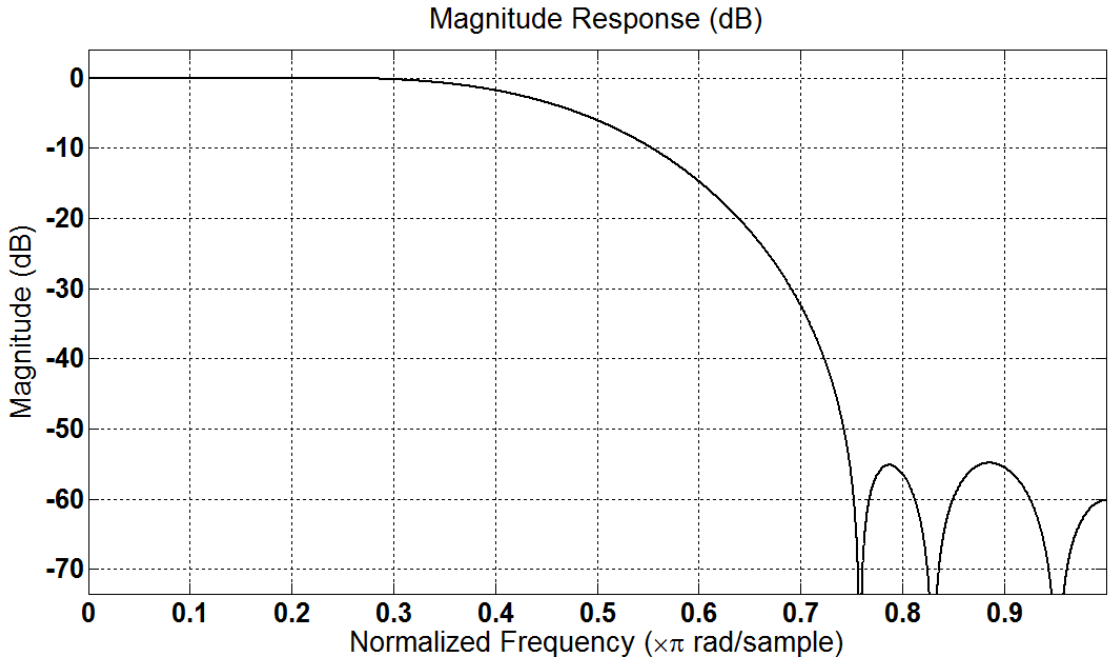


Figure 4.13. Magnitude response of the HB filter.

The HB filter is 11 tap and 7 seven coefficients out of 11 are non-zero. Only 4 of them are different since the filter coefficients are symmetric. The only non-zero odd index coefficient is the middle one which has value of 0.5 and simply realized by one shift of the data to the right. All other non-zero coefficients are even index coefficients and they are at the same branch when polyphase decomposition is applied. This allows applying sub-expression sharing among non-zero coefficients and reusage of the hardware for the same coefficients. As can be observed from Table 4.6,

$$h_0 = h_{10}\,, \quad h_2 = h_8 \quad \text{and} \quad h_4 = h_6 \tag{4.29}$$

Only the coefficients $h_0, h_2\ and\ h_4$ are calculated. Sub-expression sharing is applied using the CSD representations of the coefficients in Table 4.6. Grouping of the expressions are depicted in Figure 4.14. Every color and color group represents expressions to be shared.

Table 4.7. Sub-expression sharing applied to the HB filter coefficients.

	1	2	3	4	5	6	7	8	9	10	11	12
h_0	0	0	0	0	0	1	0	N	0	1	0	1
h_2	0	0	0	N	0	0	0	0	N	0	1	0
h_4	0	1	0	1	0	N	0	1	0	1	0	N

Equations used to calculate the multiplication of the input data with the coefficients are given below. x_0 denotes even samples of the input and x_1 denotes the odd samples of the input.

$$a_1 = [+0\ +] = x_0\ + x_0 \gg 2 \tag{4.30}$$

$$a_2 = [+0\ -] = \ x_0 - \ x_0 \gg 2 \tag{4.31}$$

$$a_3 = [+0 + 0\ -] = \ x_0\ + a_2 \gg 2 \tag{4.32}$$

$$[-0\ +] = \ - a_2 \tag{4.33}$$

Common expressions are found by only 3 addition/subtraction operation. Calculation of multiplication continues with Equations (4.34) - (4.37).

$$h_0 * x_0 = a_2 \gg 6 + a_1 \gg 10 \quad (4.34)$$

$$h_2 * x_0 = -x_0 \gg 4 - a_2 \gg 9 \quad (4.35)$$

$$h_4 * x_0 = a_3 \gg 2 + a_3 \gg 8 \quad (4.36)$$

The only odd sample and coefficient multiplication is found with a single shift to the right,

$$h_5 * x_1 = x_1 \gg 1 \quad (4.37)$$

Calculations of the coefficient multiplication for the HB filter designed in polyphase form, is given in Figure 4.14. The figure schematically explains the operations provided above. Numbers on the lines represents the amount of shift operation to the right.

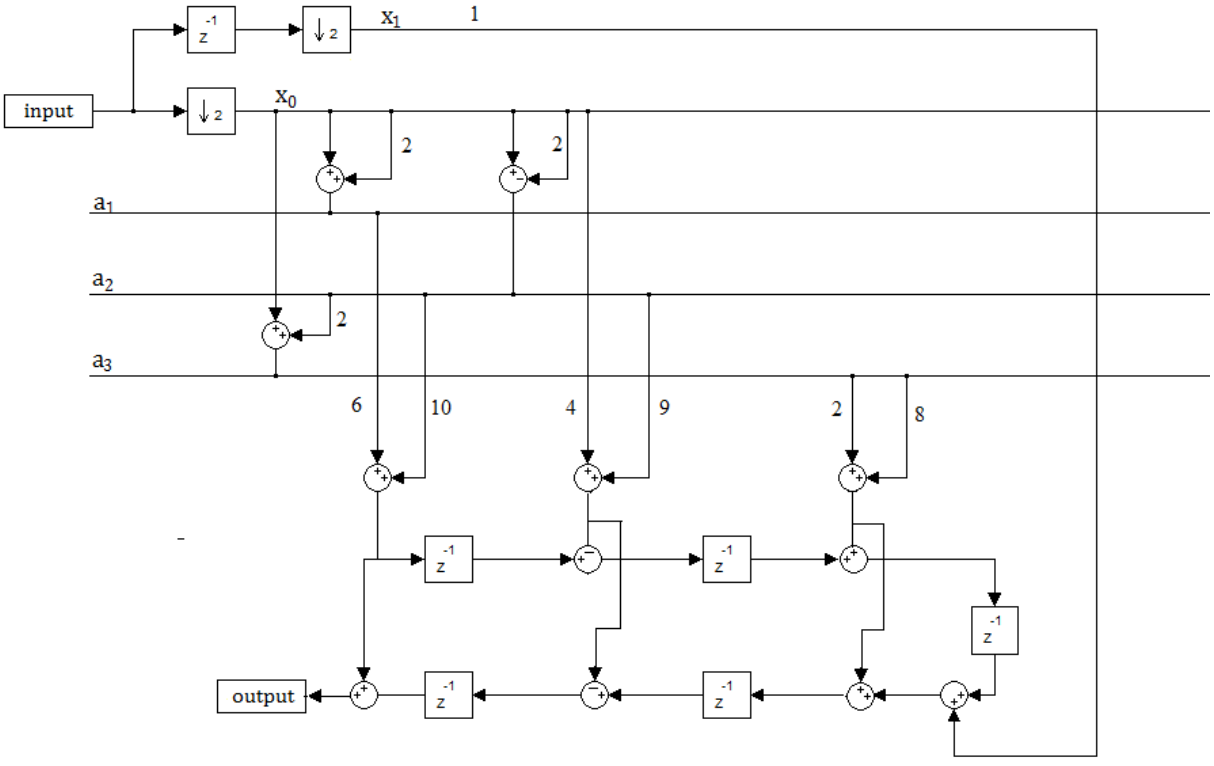


Figure 4.14. HB filter block diagram with calculations of coefficient multiplication.

Data at the output of the CIC has 15 bit word length. Word length increases to 27 bits after the multiplication with the coefficients inside HB filter. Just after multiplication, but before the delay line, 6 bits in the LSB are truncated. The amount of the bits that are

truncated are determined by Simulink simulation such that there is not any significant SNR degradation at the output. This suggests that truncation is done after the design of decimation filter. Addition/subtraction operations also increase bit length. At the output of the filter, word length of the data is reduced to 18 bits and data is given to FIR filter at the last stage as input.

Overall, in HB filter 12 addition/subtraction operators and 8 delay blocks are used. The HB filter is realized with Verilog Hardware Description Language (HDL). The design method is register transfer level (RTL). Adders/subtractions are not optimized; therefore, they are synthesized using ripple carry adders by the Verilog tool. The designed filter is compared with the desired one in Simulink and they are perfectly matching, giving exactly same outputs. Hence transfer functions are exactly equal.

4.3.4 FIR filter Design

Similar to HB filter, FIR filter specifications are determined by investigating the output characteristics of the previous stage, the architecture in Figure 4.2 and the decimation filter specifications in Table 4.2. Some important parameters of the FIR filter are provided in Table 4.8. Its passband frequency is chosen to be 22 kHz and it has 3 dB attenuation at 25 kHz. 3 dB attenuation is tolerable considering the fact that audio signals have frequency range a little less than 25 kHz. Choosing the passband frequency at 22 kHz also helps to limit the aliasing noise in to baseband.

Table 4.8. Specifications of the FIR filter.

Parameters	**Values**
Sampling Frequency (input)	100KHz
Decimation Ratio (M)	2
Passband Frequency (F_{pass})	22 kHz
Stopband Frequency (F_{stop})	32 kHz
Passband Ripple (A_{pass})	0.01
Stopband Attenuation (A_{stop})	60dB

FIR filter coefficients are obtained by using GAM algorithm [3]. GAM algorithm was proposed by Mustafa Aktan for designing low power efficient linear phase FIR filters.

The algorithm reduces the number of signed power of two (SPT) terms in the coefficients while keeping quantization word length as small as possible [47]. Filter coefficients are found according to given specifications of the filter. GAM algorithm is not used for HB filter because algorithm does not recognize specialty of the HB filter and gives all the coefficients non-zero therefore it becomes inefficient to use this algorithm for HB filter.

Coefficients of the FIR filter are given in Table 4.9 with their 2's Complement and CSD representations. In CSD form, word length of the coefficients is 11 bits.

Table 4.9. Coefficients of the FIR filter.

	Quantized Coefficients	**CSD**	**Power of 2**
$h_0 = h_{26}$	0.002929687500000	0.0000000+0–00	$2^{-8} – 2^{-10}$
$h_1 = h_{25}$	0.003906250000000	0.0000000+0000	2^{-8}
$h_2 = h_{24}$	-0.004882812500000	0.0000000–0–00	$–2^{-8} – 2^{-10}$
$h_3 = h_{23}$	-0.007812500000000	0.000000–00000	$–2^{-7}$
$h_4 = h_{22}$	0.010742187500000	0.00000+0–0–00	$2^{-6} – 2^{-8} – 2^{-10}$
$h_5 = h_{21}$	0.013183593750000	0.00000+00–0–0	$2^{-6} – 2^{-9} – 2^{-11}$
$h_6 = h_{20}$	-0.021484375000000	0.0000–0+0+000	$–2^{-5} + 2^{-7} + 2^{-9}$
$h_7 = h_{19}$	-0.018066406250000	0.00000–00–0–0	$–2^{-6} – 2^{-9} – 2^{-11}$
$h_8 = h_{18}$	0.044921875000000	0.000+0–00–000	$2^{-4} – 2^{-6} – 2^{-9}$
$h_9 = h_{17}$	0.024414062500000	0.0000+0–00+00	$2^{-5} – 2^{-7} + 2^{-10}$
$h_{10} = h_{16}$	-0.093750000000000	0.00–0+0000000	$–2^{-3} + 2^{-5}$
$h_{11} = h_{15}$	-0.028808593750000	0.0000–000+0+0	$–2^{-5} + 2^{-9} + 2^{-11}$
$h_{12} = h_{14}$	0.313476562500000	0.0+0+00000+00	$2^{-2} + 2^{-4} + 2^{-10}$
h_{13}	0.529296875000000	0.+000+000–000	$2^{-1} + 2^{-5} – 2^{-9}$

The desired specifications can also be achievable with a 26 tap filter; however selecting an even number would prevent the reusage of hardware when polyphase decomposition method is applied. The reason is that, in polyphase decomposition coefficients are separated to 2 branches by their indices; even and odd indexed coefficients are gathered at the same branch as in Figure 4.11. In order to reuse the symmetric coefficients, they should have same kind of indices like even or odd. Therefore, instead of a 26 tap filter, a 27 tap FIR filter is chosen.

Filter coefficients are used in a filter block in Simulink tool for simulation purposes. Magnitude response of the filter is obtained in Simulink as well. Magnitude response of the

filter is depicted in Figure 4.15. π in the figure denotes half the sampling frequency which is 50 kHz.

Sub-expression sharing method is also applied to the FIR filter. Table 4.10 depicts the grouping of the common sequences in filter coefficients. It should be noted that sub-expression sharing method is applied to odd indexed and even indexed coefficients separately because of the polyphase filter architecture. x_0 denotes even samples of the input and x_1 denotes odd samples of the input.

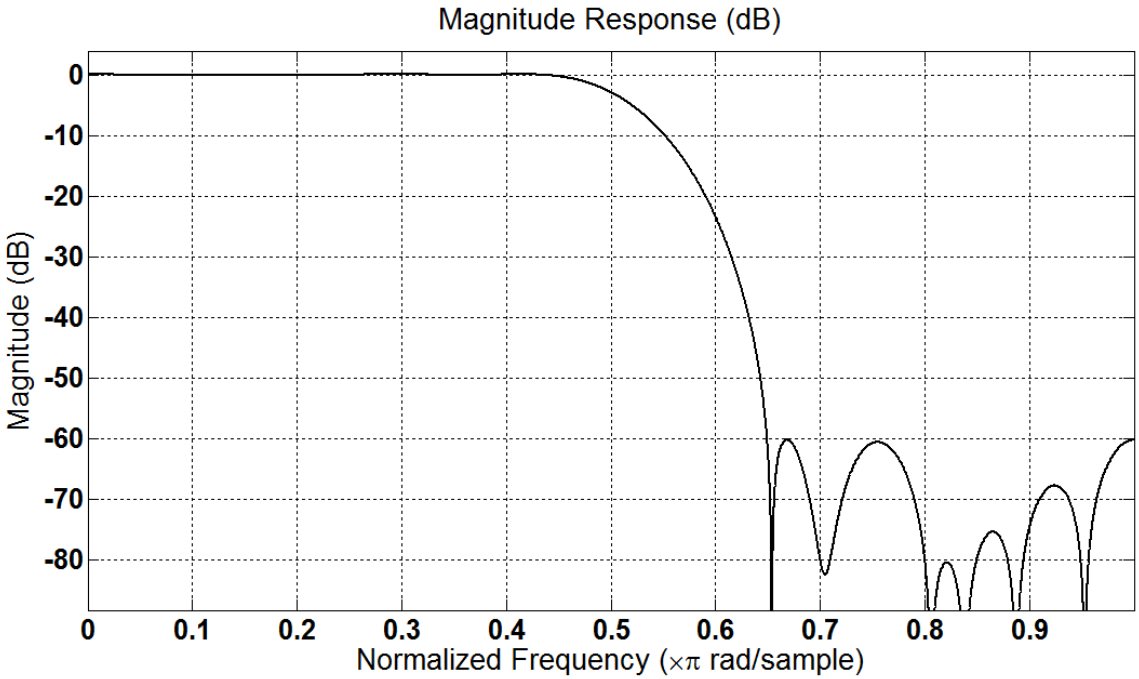


Figure 4.15. Magnitude Response of the FIR filter.

For even indexed coefficients, common expressions are found by Equations (4.38) to (4.42).

$$a_1 = x_0 - x_0 \gg 2 \tag{4.38}$$

$$a_2 = x_0 + x_0 \gg 2 \tag{4.39}$$

$$a_3 = x_0 - a_2 \gg 2 \tag{4.40}$$

$$a_4 = a_1 - \ x_0 \gg 5 \tag{4.41}$$

$$a_5 = a_2 + \ x_0 \gg 8 \tag{4.42}$$

Table 4.10. Common sub-expressions of the FIR filter coefficients.

	1	2	3	4	5	6	7	8	9	10	11
h_0								1	0	N	
h_1								1			
h_2								N	0	N	
h_3							N				
h_4						1	0	N	0	N	
h_5						1	0	0	N	0	N
h_6					N	0	1	0	1		
h_7						N	0	0	N	0	N
h_8				1	0	N	0	0	N		
h_9					1	0	N	0	0	1	
h_{10}			N	0	1						
h_{11}					N	0	0	0	1	0	1
h_{12}		1	0	1	0	0	0	0	0	1	
h_{13}	1	0	0	0	1	0	0	0	N		
h_{14}		1	0	1	0	0	0	0	0	1	
h_{15}					N	0	0	0	1	0	1
h_{16}			N	0	1						
h_{17}					1	0	N	0	0	1	
h_{18}				1	0	N	0	0	N		
h_{19}						N	0	0	N	0	N
h_{20}					N	0	1	0	1		
h_{21}						1	0	0	N	0	N
h_{22}						1	0	N	0	N	
h_{23}							N				
h_{24}								N	0	N	
h_{25}								1			
h_{26}								1	0	N	

According to these common expressions, multiplications are,

$$P[0] = h_0 * x_0 \ = \ a_1 \gg 8 \tag{4.43}$$

$$P[2] = h_2 * \ x_0 \ = - \ a_2 \gg 8 \tag{4.44}$$

$$P[4] = h_4 * x_0 \ = \ a_3 \gg 6 \tag{4.45}$$

$$P[6] = h_6 * \ x_0 \ = - a_3 \gg 5 \tag{4.46}$$

$$P[8] = h_8 * \ x_0 \ = \ a_4 \gg 4 \tag{4.47}$$

$$P[10] = h_{10} * x_0 = -a_1 \gg 3 \quad (4.48)$$

$$P[12] = h_{12} * x_0 = a_5 \gg 2 \quad (4.49)$$

Coefficients with indices 26, 24 … 14 are equal to coefficients with indices 0, 2 … 12, respectively.

Similarly for odd coefficients, common sub-expressions are as follows:

$$b_1 = x_1 + x_1 \gg 2 \quad (4.50)$$

$$b_2 = x_1 - b_1 \gg 3 \quad (4.51)$$

$$b_3 = x_1 + b_1 \gg 3 \quad (4.52)$$

$$b_4 = x_1 - x_1 \gg 2 \quad (4.53)$$

$$b_5 = b_4 + x_1 \gg 5 \quad (4.54)$$

$$b_6 = x_1 - x_1 \gg 4 \quad (4.55)$$

$$b_7 = b_6 - x_1 \gg 6 \quad (4.56)$$

$$b_8 = x_1 + b_6 \gg 4 \quad (4.57)$$

Utilizing the sub-expressions, multiplications are calculated as:

$$P[1] = h_1 * x_1 = x_1 \gg 8 \quad (4.58)$$

$$P[3] = h_3 * x_1 = -x_1 \gg 7 \quad (4.59)$$

$$P[5] = h_5 * x_1 = b_2 \gg 6 \quad (4.60)$$

$$P[7] = h_7 * x_1 \quad = -b_3 \gg 6 \tag{4.61}$$

$$P[9] = h_9 * x_1 \quad = \quad b_5 \gg 5 \tag{4.62}$$

$$P[11] = h_{11} * x_1 \quad = -\ b_7 \gg 5 \tag{4.63}$$

$$P[13] = h_{13} * x_1 \quad = \quad b_8 \gg 1 \tag{4.64}$$

P denotes the multiplication results of the data and the coefficients. In order to form a structure, odd and even elements of P are placed in a table (Table 4.11) with respect to delay amounts. If equal coefficients are represented with same index Table 4.12 is obtained. The black line in Table 4.12 is the axis of symmetry for delay line in Figure 4.16.

Table 4.11. Multiplications and their corresponding delays.

Delay Amount	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
Elements of P	0	2	4	6	8	10	12	14	16	18	20	22	24	26
Elements of P	1	3	5	7	9	11	13	15	17	19	21	23	25	

Table 4.12. Multiplications and their corresponding delays revised.

Delay Amount	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**	**13**
Elements of P	0	2	4	6	8	10	12	12	10	8	6	4	2	0
Elements of P	1	3	5	7	9	11	13	11	9	7	5	3	1	

According to the Table 4.12 structure in Figure 4.16 is obtained.

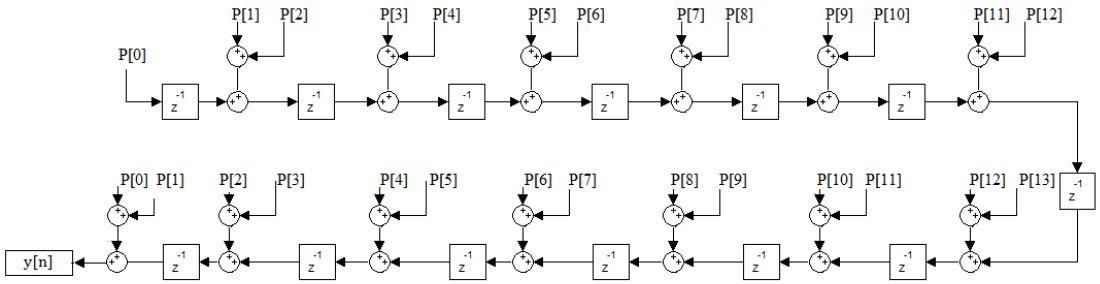


Figure 4.16. Delay line of the FIR filter.

Data at the output of the HB filter has 18 bit word length. Word length increases to 30 bits after multiplication with the coefficients inside the FIR filter. Just after multiplication before the delay line, 6 bits in the LSB are truncated. Amount of the bits that are truncated are determined by Simulink simulation such that there is no significant SNR degradation at the output. This suggests that truncation is done once the decimation filter is complete. Addition/subtraction operations also increase bit length. At the output of the filter word length of the data is reduced to 17 bits, which is also output bit length.

Overall, in the FIR filter 36 addition/subtraction operators and 16 delay blocks are used. The FIR filter is realized with Verilog HDL. Design method is register transfer level (RTL). Adders/subtractions are not optimized. The designed filter is compared with the desired one in Simulink and they perfectly matched, giving exactly same outputs. Consequently their transfer functions are exactly equal.

4.4 CIC filter

4.4.1 Introduction

CIC filter was first presented in [33] in 1981. Since then, it has been commonly used in decimation filters. The structure of Hogenauer's CIC filter is shown in Figure 4.17. It consists of IIR integrator and FIR Comb sections. Number of integrator and comb (differentiator) pairs determines the order of the CIC. In Figure 4.17; R is the decimation factor, f_s is the sampling frequency and M is the differential delay. M is usually restricted to 1 or 2. The switch in the middle decimates the sampling rate.

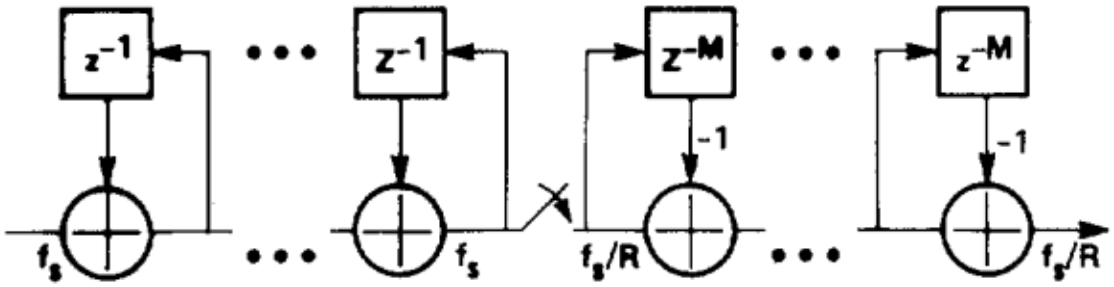


Figure 4.17. Conventional CIC filter.

Transfer function of the CIC is,

$$H(z) = \frac{(1 - z^{-RM})^L}{(1 - z^{-1})^L} \tag{4.65}$$

Frequency response of the CIC filter is given in the Figure 4.18. The figure depicts the frequency response of a 4th order CIC filter with a decimation factor of 8. CIC filter is very efficient in terms of power and area, since it can largely decimate the sampling ratio. Its efficiency can be understood by its frequency response. CIC has nulls at the multiples of the decimated frequency (f_s/R); therefore, the aliasing signal is attenuated. If the signal band is f_c, then the aliasing regions on the CIC are described by,

$$f_s\left(\frac{i}{R}\right) - f_c \leq f \leq f_s\left(\frac{i}{R}\right) + f_c \tag{4.66}$$

where $i = 1,2 \ldots (R - 1)$.

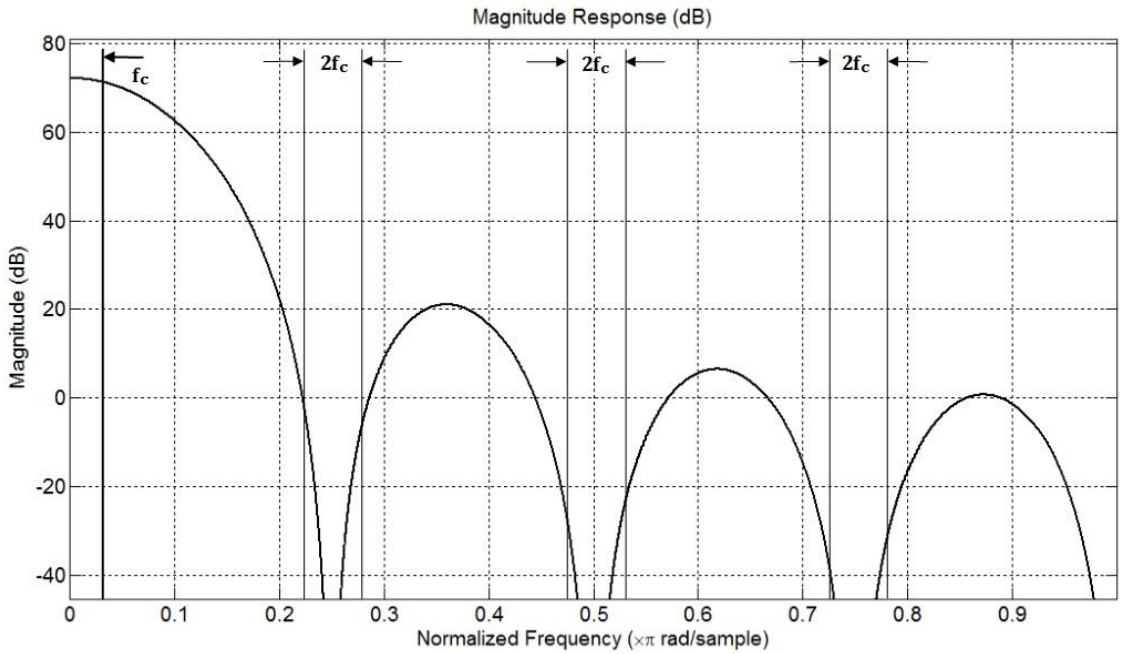


Figure 4.18. Frequency response of a CIC filter.

IIR integrator section of the CIC inherently overflows, but this can be compensated by the comb section and undisturbed data can be obtained at the output of the filter; however, for this compensation, the CIC should have a certain resolution (bit length). Minimum bit length B_{min} is,

$$B_{out} = \lceil L\ (log_2 RM) + B_{in} \rceil \tag{4.67}$$

L is the order of the CIC and B_{in} is the input signal word length. "$\lceil x \rceil$" operator means smallest integer not less than x.

4.4.2 CIC filter Design

In Section 4.2 features of the CIC filter were explained. Decimation factor, R, is 8. As a rule of thumb, the order of the CIC filter should be one greater than the noise shaping order of the delta sigma modulator [48]. Since order of the modulator is 3, order of the CIC, L, is chosen as 4. Differential delay, M, is chosen as 1. Conventional CIC having the specifications mentioned above is shown in Figure 4.19. The structure is designed with MATLAB Simulink tool. Figure 4.20 depicts its magnitude response for frequency range normalized to sampling frequency.

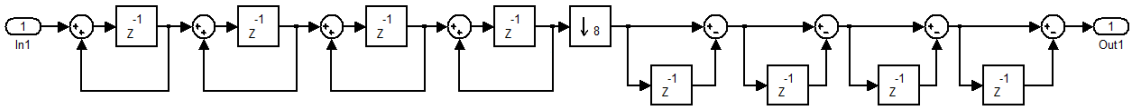


Figure 4.19. Structure of the conventional 4th order CIC filter with decimation factor of 8.

Input word length of the CIC is 3, and according to Equation (4.67), minimum word length of the registers and adders inside the CIC are found to be 15.

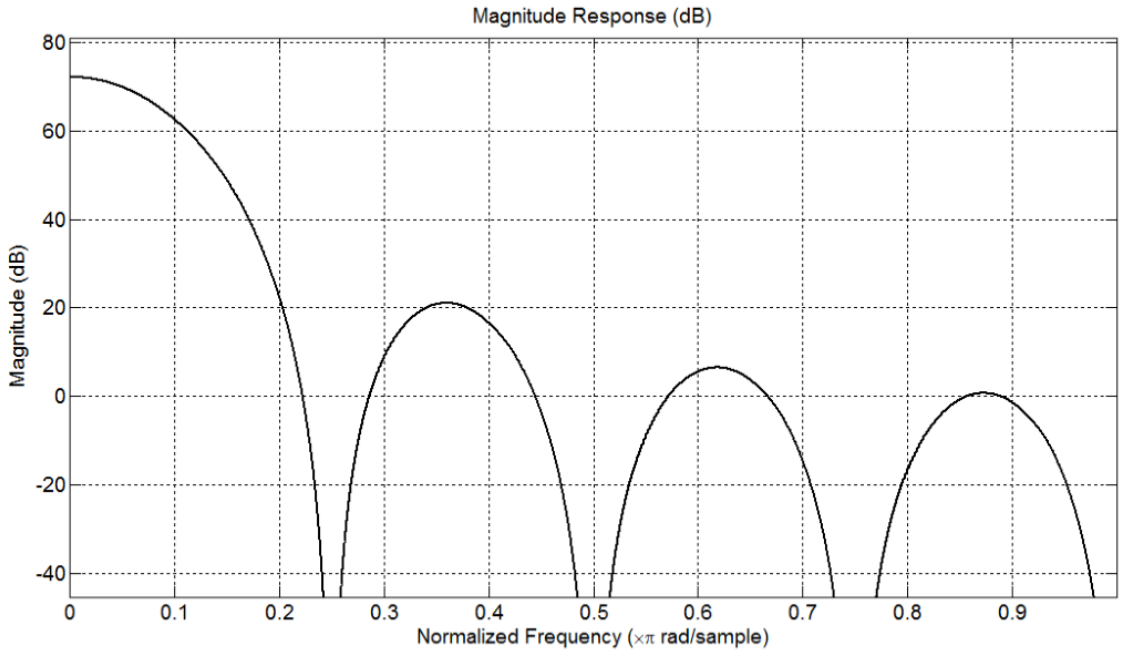


Figure 4.20. Magnitude response of the conventional 4th order CIC with decimation factor of 8.

Integrator section of the CIC filter is IIR; therefore, all the blocks of the CIC should be implemented in full precision which is 15 bits. CIC filter implementation uses less hardware compared to the following stages; however, it operates at much higher frequency, especially its IIR part. Therefore, it dissipates a large amount of power. In order to decrease the power consumption of the filter it would be wise to get rid of the IIR part and turn it to FIR format. In addition to conversion to FIR format, applying polyphase decomposition would be beneficial in terms of power consumption.

Transfer function of the intended CIC is,

$$H(z) = \frac{(1 - z^{-8})^4}{(1 - z^{-1})^4} = (1 + z^{-1} \ldots + z^{-7})^4 \tag{4.68}$$

In order to convert it into FIR format, denominator of the function should be eliminated. However, if numerator is divided arithmetically it would be very complicated and difficult to implement such a filter since there will be 8 phases for polyphase decomposition.

At this stage rather than implementing decimation factor of 8 in a single stage, using [49] a 3 stage CIC filter structure is proposed; each stage has a decimation factor of 2 (Figure 4.21). According to Equation (4.67), minimum word lengths of output of stages are 7, 11 and 15 in order. Therefore, first two stages will have reduced word lengths and the last two stages will operate at reduced frequency. This implementation will have reduced power dissipation and simpler structure [49]. Up to %50 power saving can be observed.

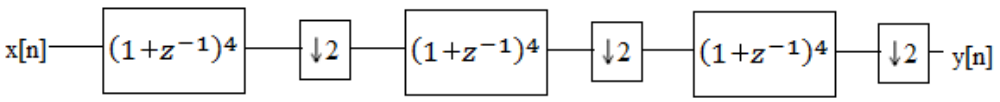


Figure 4.21. Three stage CIC implementation.

Structure of each stage is similar to Figure 4.19 but decimation factor is 2. Their transfer function is given by following Equation (4.65),

$$H(z) = \frac{(1 - z^{-2})^4}{(1 - z^{-1})^4} \tag{4.68}$$

It can also be written in FIR form as,

$$H(z) = (1 + z^{-1})^4 \tag{4.69}$$

and if the terms with odd and even number of delays are grouped, one obtains,

$$H(z) = (1 + 6z^{-2} + z^{-4}) + z^{-1}(4 + 4z^{-2}) \tag{4.70}$$

$$H(z) = \mathrm{E}_0(z^2) + z^{-1}\,E_1(z^2) \tag{4.71}$$

$$\mathrm{E}_0(\mathrm{z}) = (1 + 6z^{-1} + z^{-2}) \tag{4.72}$$

$$\mathrm{E}_1(\mathrm{z}) = (4 + 4z^{-1}) \tag{4.73}$$

Equation (4.70) is in similar format as Equation (4.26) and it can be implemented in polyphase form using Equations (4.72) and (4.73). CIC filter is converted to FIR form and all low power techniques also apply to CIC. The filter defined by Equation (4.70) can be implemented as in Figure 4.22. The filter structure is transposed direct form, hardware is reused for the same coefficients, and unit delay before decimation by 2 is operated with negative edge of the clock having half the sampling frequency of the input. Therefore, as explained in section 4.3.2.4 the need for 1.6 MHz clock signal is eliminated. In addition, word lengths of the registers are optimized since the structure is non-recursive. All 3 stages in CIC filter have the same block diagram. Only the word lengths of the registers and adders/subtractors differ.

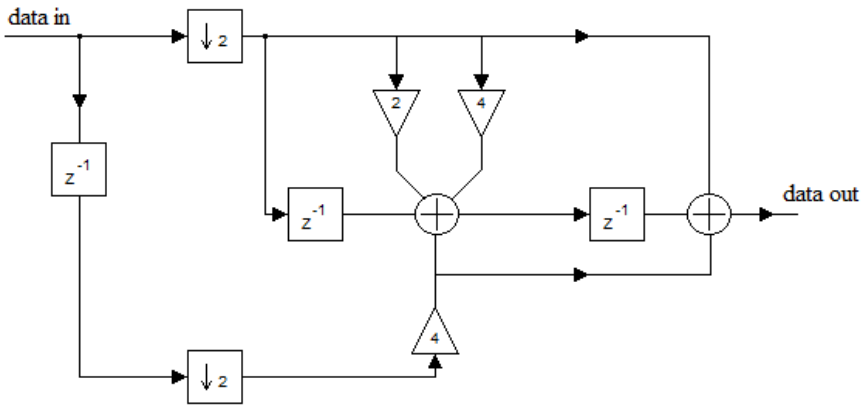


Figure 4.22. Structure of one stage of the CIC filter.

Magnitude responses are the same for each stage of the CIC and are given in Figure 4.23.

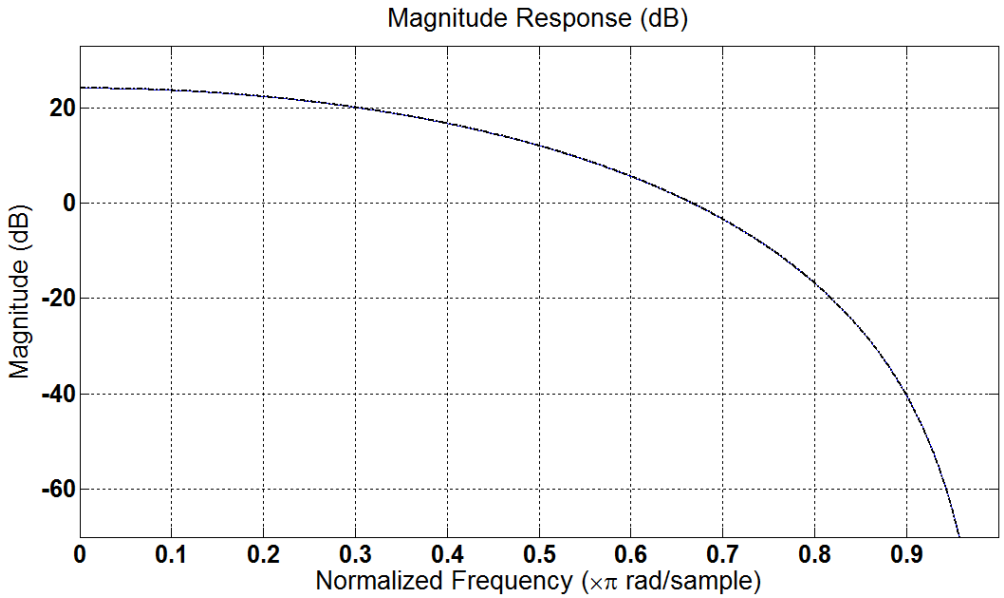


Figure 4.23. Magnitude response of one stage of CIC versus normalized frequency.

In Figure 4.23, π corresponds to $f_s/2$. Since it is in digital domain, magnitude response has replicas at $2k\pi$. Interval between π - 2π is the mirror image of the interval 0 to π. Some important parameters of the single stage and 3-stage CIC filter structure is provided in Table 4.13. In signal baseband droops in the CIC filter magnitude responses are 0.89 dB and 0.87 dB for single stage and 3-stage CIC filter implementations, respectively. They can be regarded as the same.

Table 4.13. Specifications of the single stage and 3-stage CIC filter.

	Droop (dB)	**Worst Case Aliasing Level (dB)**	**Noise level at worst case aliasing (dB)**
First Stage	0.041	-104.8	-40
Second Stage	0.162	-80.7	-40
Third Stage	0.662	-60	-58
Overall Response	0.87	-60	-58
Single stage CIC	0.89	-69	-58

Another important parameter is the worst case aliasing attenuation level of the CIC filter magnitude response. For the 3-stage CIC filter implementation, the decisive element is the last stage. Attenuation level of the last stage at worst case aliasing condition is -60 dB. This is the value at the edge of the aliasing band; filter magnitude response goes down rapidly for aliasing condition. For instance, at 25 kHz aliasing level of the filter is -60 dB and at 15 kHz it goes down to -80 dB. Noise level of the input (Figure 4.1) is around -58

dB; therefore aliasing levels of the CIC filter do not pose any thread to the performance of the filter.

Similar to HB and FIR filters, the CIC filter is realized using Verilog HDL. Both single stage CIC and 3-stage CIC are implemented with Verilog. The designed filters are have transfer functions same as the ideal ones in Simulink. Their transfer functions are also the same. Their power comparison is mentioned in section 6. For a 3-stage CIC filter, a total 15 registers and addition/subtraction blocks are utilized; average word length of the blocks is 7 to 8.

4.5 Results

Decimation filter blocks are first realized and simulated in MATLAB Simulink tool. CIC, HB and FIR filters are implemented as explained throughout this chapter. Filter blocks are also realized with Verilog HDL. Two different decimation filters are obtained: ideal one in MATLAB Simulink tool and the real one in Verilog HDL. Ideal and the real one compared in terms of performance.

At first, the decimation filter was synthesized with Leonardo Spectrum of Mentor Graphics. Leonardo Spectrum synthesizes all levels of abstraction, and minimizes the amount of logic required, resulting in a final netlist description in chosen technology [50]. The netlist uses the standard library of 0.18 μm CMOS technology which is then imported to Mentor Graphics. Digitally, the decimation filter operated as expected and exactly matched with the output waveform of the ideal one in MATLAB Simulink tool.

Standard library of the 0.18 μm technology consists of Verilog descriptions and layouts; therefore, in order to obtain the power consumption, layout of the filter should be drawn. However, because of the problems in auto place and route function in Mentor Graphics with standard cell library of 0.18 μm technology, layout could not be drawn. Leonardo Spectrum cannot evaluate power synthesis even though technology contains power parameters in Verilog implementation of the standard library gates. In order to obtain power consumption Synopsys Design Compiler (DC) is utilized. It is the equivalent tool of Synopsys, works in same manner with Leonardo Spectrum and has power analysis

feature. Decimation filter is also synthesized with Synopsys DC tool, power consumption of the blocks and overall filter is obtained.

The decimation filter is designed for 3-bit DT Sigma-Delta ADC intended for audio applications. The DT Sigma-Delta Modulator has 1.6 MHz sampling frequency, corresponds to OSR of 32. Transient output series of the Modulator obtained by circuit simulation is applied to the decimation filter as input. PSD of the output of the DT Sigma-Delta Modulator is given in Figure 4.26 and corresponding decimation filter output waveform is depicted in Figure 4.27. Decimation filter output is investigated for the modulator output with different SNR values and the obtained results are summarized in Table 4.14. As can be observed from the table, the decimation filter causes very low SNR degradation; its input and output resolutions are practically the same. SNR degradation is 1.6 dB for the worst case and 0.5 dB for the best. The pre-determined performance target has been achieved.

Table 4.14. Decimation filter output SNR values for different inputs.

Input SNR (dB)	**Output SNR (dB)**
94.5	92.9
93.1	92.3
92	91.5
85.3	84.5

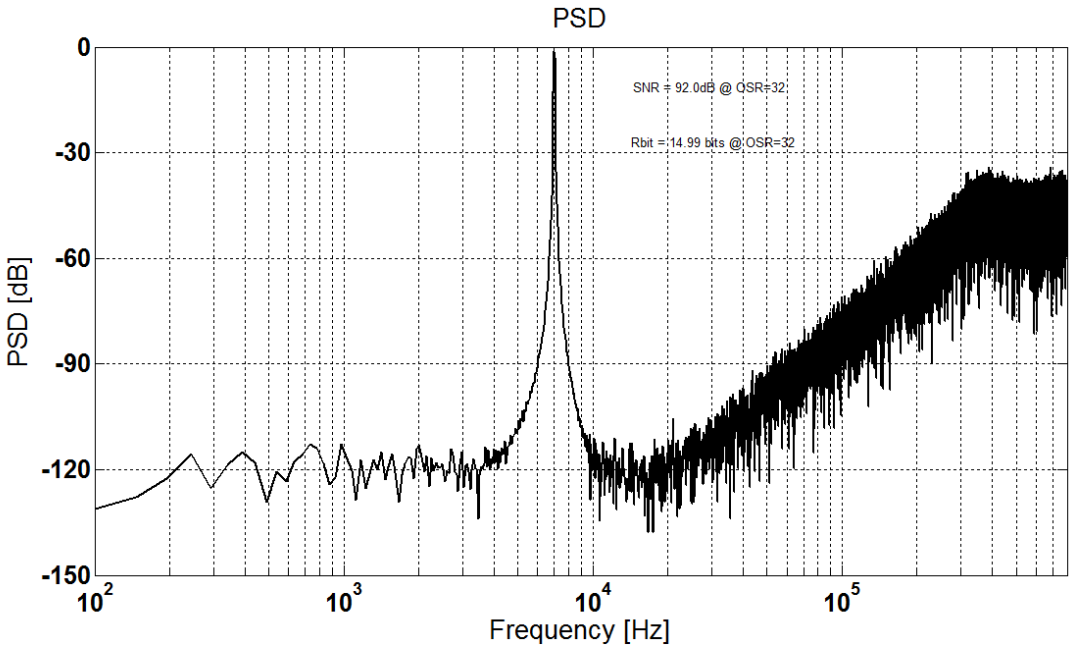


Figure 4.24. PSD of the output of the DT Sigma-Delta ADC.

The DT Sigma-Delta Modulator has a power consumption around 15 µW. Power target of the decimation filter is below this level. Power consumption of the decimation filter and its sub-blocks, operating under a supply voltage of 1.8 V are given in Table 4.13.

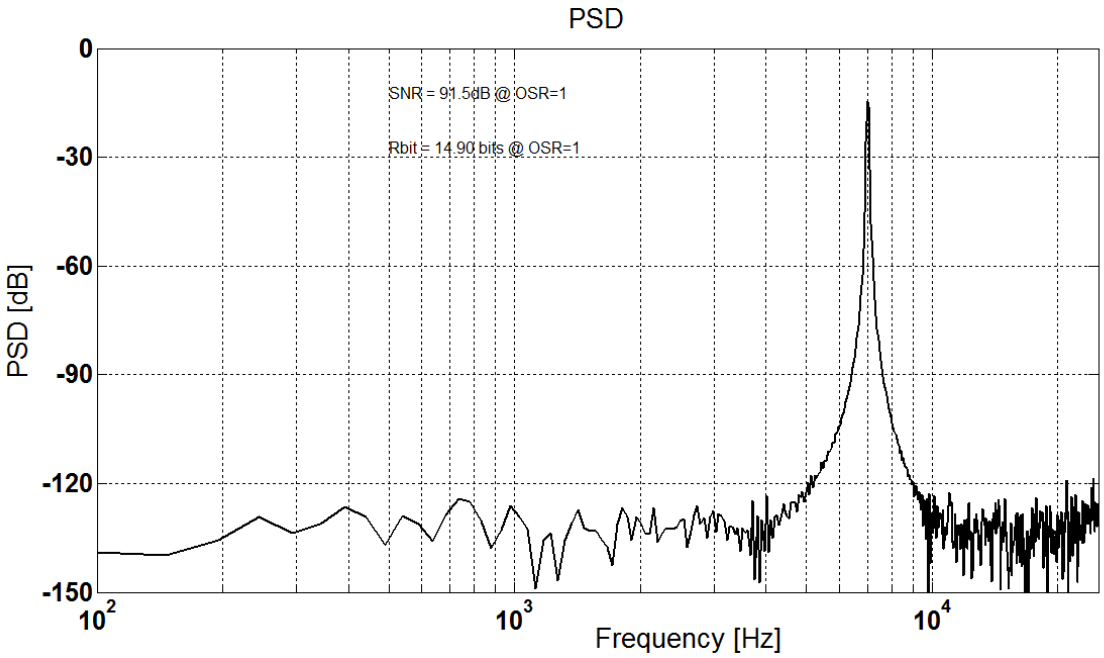


Figure 4.25. PSD of the output of Decimation filter.

Table 4.15. Power consumption of Decimation filter and its sub-blocks.

Block	**CIC**	**Half Band**	**FIR**	**Total**
Power (µW)	3.43	1.26	1.80	7.24

Total power consumption is 7.24 µW; which is only half of that of the analog modulator. Sub-blocks are compiled separately and therefore their power consumptions are somewhat less than the total power consumption. The reason is that in the overall system simulation they will load each other and increase the capacitance, hence power consumption. CIC filter consumes about half of the power. According to power analysis, power consumption of the single stage CIC filter is twofold that of the 3-stage CIC filter.

Power consumption of the synthesized filter is compared to decimation filters in the literature. In order to properly compare them, FoM measure in Equation (4.74) is adopted [42].

$$\text{FoM} = \frac{Power}{Output\ Wordlength\ x\ Sampling\ Frequency} \tag{4.74}$$

In Table 4.16 technologies, supply voltages, output word lengths, sampling frequencies, and power consumptions of different studies in the literature are presented along with the decimation filter designed in this thesis. Proposed structure in thesis has far better results than previous decimation filters.

Table 4.16. Comparison of different Decimation filters in the literature.

	Technology	Supply Voltage (V)	Resolution (bits)	Sampling Frequency (kHz)	Power (µW)	FoM (pJ/bits)
[6]	0.35 µm CMOS	3.3	8	8	0.310	4.85
[51]	-	1	1.2	19.2	20	86.8
[52]	0.18 µm CMOS	1.8	14	5644.8	9000	138
[53]	1 µm CMOS	3	16	11300	6500	35.9
[54]	0.18 µm CMOS	0.9	16	3072	23.6	0.48
[55]	0.7 µm CMOS	3.3	24	6144	20000	135.63
[56]	0.6 µm CMOS	3.3	16	25000	155000	387.5
[42]	0.35 µm CMOS	1.2	10	2500	159	6.3
[57]	0.13 µm CMOS	1.2	12	500000	4800	0.8
This Thesis [58]	0.18 µm CMOS	1.8	17	1600	7.24	0.27

5 CONCLUSION & FUTURE WORK

In this thesis, a low power CT Sigma Delta modulator and a low power Decimation filter were designed and implemented in 0.18 μm CMOS technology.

For the CT Sigma Delta modulator, a simple easy to design 2^{nd} order structure is chosen with feed-forward configuration and utilizing NRZ feedback type. CT Sigma Delta modulator is implemented at schematic level and simulation results are obtained for evaluation of the performance. It operates as expected according to these simulation results; however, layout of the modulator can be drawn and post-layout analysis can be conducted in order to further investigate the performance of the modulator. Post-layout simulations obviously will provide better information about the CT Sigma Delta modulator; on the other hand, the desired target is achieved in the framework of this thesis.

Decimation filter is intended for low power DT Sigma Delta modulator intended for audio applications. The modulator has very high resolution and consumes very low power. The aim is to have a decimation filter consuming less power and not causing considerable performance degradation. Multistage filter architecture is preferred for the purpose, CIC filter is at the input stage with HB and FIR filters following it. Multistage structures save both from power and area. Decimation filter characteristics with the corresponding sub-blocks are obtained by using MATLAB. Coefficients of the HB filter also generated using the MATLAB FDAtool. FIR filter is obtained with GAM algorithm. GAM algorithm promises using the least number of non-zero terms in coefficients of FIR filters.

Instead of utilizing conventional CIC filter, a 3-stage CIC filter is implemented. This structure also allows implementing the filter purely in FIR format; therefore, it is implemented in the same way as HB and FIR filters. In order to reduce the power of the filters, multiplierless filter architecture is adopted. In order to further reduce power consumption of the filters and obtain very low power decimation filter several digital low power design techniques are utilized. Filter is realized with Verilog HDL and synthesized in 0.18 μm CMOS technology with Synopsys DC tool. Designed filter works exactly same as the ideal filter set in MATLAB Simulink. Power consumption of the Decimation filter is

only half of power consumption of DT Sigma Delta modulator. Results are also compared with filters in the literature. The designed decimation filter achieves very low FoM value and surpasses the previous designs.

As a future work, layout of the Decimation filter can be drawn and combined with the modulator for investigating performance of ADC as a whole. Overall ADC performance can be measured and if performance specifications are satisfied they can be fabricated on a single chip.

REFERENCES

1. Ruye W., *Signal Sampling*, http://fourier.eng.hmc.edu/e102/lectures/samplingThm/node1.html, 2012.

2. Parks, T., McClellan, J., "Chebyshev Approximation for Nonrecursive Digital Filters with Linear Phase," *IEEE Transactions on Circuit Theory, v*ol.19, no.2, pp.189-194, March 1972.

3. Aktan, M., Yurdakul, A., Dundar, G., "An Algorithm for the Design of Low-Power Hardware-Efficient FIR Filters," *IEEE Transactions on Circuits and Systems I: Regular Papers,* , vol.55, no.6, pp.1536-1545, July 2008.

4. Norsworthy S.R., R. Schreier, G.C. Temes, *Delta-Sigma Data Converters Theory, Design, and Simulation*, IEEE Press, New York, 1998.

5. Jarman D., *A Brief Introduction to Sigma-Delta Conversion*, http://www.intersil.com/data/an/an9504.pdf, 1995.

6. Gerosa A., Neviani, A., "A Low Power Decimation Filter For A Sigma-Delta Converter Based On A Power-Optimized Sinc Filter," *Proceedings of International Symposium on Circuits and Systems*, vol.2, pp.245-248, May 2004.

7. Gürsoy Ö., O. Sağlamdemir, M. Aktan, S. Talay, G. Dündar, "Low Power Decimation Filter Architectures for Sigma-Delta ADC's," *Proceedings of ELECO'05*, pp. 72-75, 7-11 December 2005.

8. Maloberti F., *Data Converters*, Springer, Dordrecht, 2008.

9. Ortmanns M., Gerfers F., *Continuous-Time Sigma-Delta A/D Conversion*, Springer, Dordrecht, 2006.

10. Brooks, T.L., Robertson, D.H., Kelly, D.F., Del Muro, A., Harston, S.W., "A cascaded sigma-delta pipeline A/D converter with 1.25 MHz signal bandwidth and 89 dB SNR," *IEEE Journal of Solid-State Circuits,* vol.32, no.12, pp.1896-1906, December 1997.

11. Candy, J.C., "A Use of Double Integration in Sigma Delta Modulation," *IEEE Transactions on Communications,* vol.33, no.3, pp.249-258, March 1985.

12. Mederio F., Perez-Verdu B., Rodriguez-Vazquez A., *Top-Down Design of High Performance Sigma-Delta Modulators*, Kluwer Academic Publisher, Dordrecht, 1999.

13. Marques, A., Peluso, V., Steyaert, M.S., Sansen, W.M., "Optimal parameters for ΔΣ modulator topologies," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing,* vol.45, no.9, pp.1232-1241, September 1998.

14. Geerts Y., Steyaert M., Sansen W., *Design of Multi-Bit Delta-Sigma A/D Converters*, Kluwer Academic Publisher, Dordrecht, 2002.

15. Hayashi, T., Inabe, Y., Uchimura, K., Kimura, T., "A multistage delta-sigma modulator without double integration loop," *IEEE International Solid-State Circuits Conference,* vol.29, pp.182-183, 19-21 February 1986.

16. Fischer, G., Davis, A.J., "Alternative topologies for sigma-delta modulators-a comparative study," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing,* vol.44, no.10, pp.789-797, October 1997.

17. Matsuya, Y., Uchimura, K., Iwata, Atsushi, Kobayashi, T., Ishikawa, M., Yoshitome, T., "A 16-bit oversampling A-to-D conversion technology using triple-integration noise shaping," *IEEE Journal of Solid-State Circuits,* vol.22, no.6, pp.921-929, December 1987.

18. Longo, L., Copeland, M., "A 13 bit ISDN-band oversampled ADC using two-stage third order noise shaping," *Proceedings of Custom Integrated Circuits Conference,* vol.21, no.2, pp.1-4, 1988.

19. Boser, B.E., Wooley, B.A., "The design of sigma-delta modulation analog-to-digital converters," *IEEE Journal of Solid-State Circuits*, vol.23, no.6, pp.1298-1308, December 1988.

20. Gerfers, F., Ortmanns, M., Manoli, Y., "A 1 V, 12-bit wideband continuous-time ΣΔ modulator for UMTS applications," *Proceedings of the 2003 International Symposium on Circuits and Systems (ISCAS)*, vol.1, pp.921-924, 25-28 May 2003.

21. Bruckner, T., Kiebler, M., Lorenz, M., Zorn, C., Mathis, W., Ortmanns, M., "Calculating transfer functions of CT sigma-delta modulators with arbitrary DAC waveforms," *IEEE International Symposium on Circuits and Systems (ISCAS)*, pp.1264-1267, 19-23 May 2013.

22. Oliaei, O., "Design of continuous-time sigma-delta modulators with arbitrary feedback waveform," *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing*, vol.50, no.8, pp.437-444, August 2003.

23. Gerfers, F., Manoli, Y., "A design strategy for low-voltage low-power continuous-time ΣΔ A/D converters," *Proceedings of Design, Automation and Test in Europe*, pp.361-368, 2001.

24. Shoaei O., *Continuous-Time Delta-Sigma A/D Converters for High Speed Applications*, Ph.D. Thesis, Carleton University, 1995.

25. Jury E.I., *Theory and Applications of the Z-transform Method*, John Wiley & Sons, New York, 1964.

26. Gao, W., Shoaei, O., Snelgrove, W.M., "Excess loop delay effects in continuous-time delta-sigma modulators and the compensation solution," *IEEE International Symposium on Circuits and Systems (ISCAS)*, vol.1, pp.65-68, 9-12 Jun 1997.

27. Cherry J.A., Snelgrove W.M., *Continuous-Time Delta-Sigma Modulators for High speed A/D Conversion*, Kluwer Academic Publisher, Dordrecht, 1999.

28. Masoom, A., Hadidi, K., “A 1.5-V, Constant-Gm, Rail-to-Rail Input Stage Operational Amplifier,” *IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pp.632-635, 10-13 December 2006.

29. Trung-Kien Nguyen, Sang-Gug Lee, “Low-voltage, low-power CMOS operation transconductance amplifier with rail-to-rail differential input range,” *Proceedings of IEEE International Symposium on Circuits and Systems (ISCAS)*, pp.4-8, 21-24 May 2006.

30. Quoc-Hoang Duong, Trung-Kien Nguyen, Sang-Gug Lee, “Low-voltage low-power high dB-Linear CMOS exponential function generator using highly-linear V-I converter,” *Proceedings of the International Symposium on Low Power Electronics and Design (ISLPED)*, pp.349-352, 25-27 August 2003.

31. Ferri, G., Sansen, Willy, “A rail-to-rail constant-g_m low-voltage CMOS operational transconductance amplifier,” *IEEE Journal of Solid-State Circuits*, vol.32, no.10, pp.1563-1567, October 1997.

32. Yao L., Steyaert M., Sansen W., *Low-Power Low-Voltage Sigma-Delta Modulators In Nanometer CMOS*, Springer, Dordrecht, 2006.

33. Hogenauer, E., “An economical class of digital filters for decimation and interpolation,” *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol.29, no.2, pp.155-162, April 1981.

34. Roh, H., Byun, S., Choi, Y., and Roh, J, "Fully synthesised decimation filter for delta-sigma A/D converters," *International Journal of Electronics*, vol.97, no.6, pp.663-676, June 2010.

35. Chua C.C., B. Gwee, J. S. Chang, “A Low Voltage Micro Power Asynchronous Multiplier for a Multiplierless FIR Filter,” *Proceedings of ISCAS*, pp.381-384, 2003.

36. Bhattacharya M., J. Astola, “Multiplierless Implementation of Recursive Digital Filters Based on Coefficient Translation Methods in Low Sensitivity Structures,” *IEEE International Symposium on Circuits and Systems (ISCAS),* pp.697-700, 2001.

37. Park J., W. Jeong, H. Choo, “High Performance and Low Power FIR Filter Design Based on Sharing Multiplication,” *International Symposium on Low Power Electronics and Design (ISLPED),* pp.295-300, 2002.

38. Aktan M., G. Dündar, “Design of Digital Filters for Low Power Applications Using Integer Quadratic Programming,” *Proceedings of PATMOS’05*, pp.137-145, 2005.

39. Bellanger, M.G., Bonnerot, G., Coudreuse, M., “Digital filtering by polyphase network:Application to sample-rate alteration and filter banks,” *IEEE Transactions on Acoustics, Speech and Signal Processing,* vol.24, no.2, pp.109-114, April 1976.

40. Park I.C., H.J. Kang, “Digital Filter Synthesis Based on Minimal Signed Digit Representation,” *The Design Automation Conference (DAC),* pp.468-473, 2001.

41. Kang H.J., H.Kim, I.C. Park, “FIR Filter Synthesis Algorithms for Minimizing the Delay and the Number of Adders,” *International Conference on Computer Aided Design (ICCAD),* pp.51-54, 2000.

42. Kayaduman F., *Design of Low Power Decimation Filter for Sigma-Delta Analog Digital Converters,* M. S. Thesis, Boğaziçi University, 2011.

43. Hewlitt R.M., E. S. Swartzlander, “Canonical Signed Digit Representation for FIR Digital Filters,” *IEEE Workshop on Signal Processing Systems*, pp.416-426, 2000.

44. Hartley, R.I., “Subexpression sharing in filters using canonic signed digit multipliers,” *IEEE Transactions on Circuits and Systems II: Analog and Digital Signal Processing,* vol.43, no.10, pp.677-688, October 1996.

45. Gustafsson, O., Johansson, K., Johansson, H., Wanhammar, L., "Implementation of Polyphase Decomposed FIR Filters for Interpolation and Decimation Using Multiple Constant Multiplication Techniques," *IEEE Asia Pacific Conference on Circuits and Systems (APCCAS)*, pp.924-927, 2006.

46. Younis, N., Ashour, M., Nassar, A., "Power-Efficient Clock/Data Distribution Technique for Polyphase Comb Filter in Digital Receivers," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol.56, no.8, pp.639-643, August 2009.

47. Aktan M., *High Level Power Efficient Synthesis of FIR Based Digital Systems*,Ph.D. Thesis, Boğaziçi University, 2008.

48. Candy, J.C., "Decimation for Sigma Delta Modulation," *Transactions on Communications, IEEE*, vol.34, no.1, pp.72-76, January 1986.

49. Gao, Y., Jia, L., Isoaho, J., Tenhunen, H., "A Comparison Design of Comb Decimators for Sigma-Delta Analog-to-Digital Converters," *Journal of Analog Integrated Circuits and Signal Processing*, vol.22, no.1, pp.51-60, January 2000.

50. *Leonardo Spectrum for Altera HDL Synthesis Manual*, http://www.eng.auburn.edu /~agrawvd/COURSE/E6200_Fall08/PROJECT/VHDLSynthesisGuide.pdf, 2001.

51. Chong K.F., P. K. Gopalakrishnan, T. H. Teo, "Low Power Approach for Decimation Filter Hardware Realization," *Proceedings of World Academy of Science, Engineering and Technology*, vol. 32, pp. 2070-3740, 2008.

52. Hao L., H. Yan, R.C.C. Cheung, H. Xiaoxia, M. Shaoyu, Y. Peng, Z. Dazhong, "A High Performance Low Power ∑Δ ADC for Digital Audio Applications," *Journal of Semiconductors*, vol.31, no.5, pp.9-13, 2010.

53. Brandt B.P., B.A. Wooley, "A Low Power, Area Efficient, Digital Filter for Decimation and Interpolation," *IEEE Journal of Solid State Circuits*, vol. 29, no.6, pp.679-687, 1994.

54. Parameswaran S., N. Krishnapura, “A 100 μW Decimator for a 16 bit 24 KHz Bandwidth Audio ∑Δ Modulator,” *Proceedings of International Symposium on Circuits and Systems*, pp.2410-2413, 2010.

55. Fujimori I., K. Koyama, D. Trager, F. Tam, L. Longo, “A 5V Single Chip Delta Sigma Audio A/D Converter with 111 dB Dynamic Range,” *IEEE Journal of Solid State Circuits*, vol. 32, no.3, pp.329-336, March 1997.

56. Maulik P.C., M.S. Chadha, W.L. Lee, P.J. Crawley, “A 16-Bit 250 KHz Delta-Sigma Modulator and Decimation Filter,” *IEEE Journal of Solid-State Circuits*, vol.35, no.4, pp.458-467, 2010.

57. Jing Li, Ran Li, Ting Yi, Zhiliang Hong, Bill Yang Liu, “VLSI implementation of high-speed low power decimation filter for LTE sigma-delta A/D converter application,” *International Conference on ASIC (ASICON), 2011 IEEE*, pp.453-456, 2011.

58. Ince, M., Akcakaya, F.M., Dundar, G., "A 7.3 μW decimation filter for 15 bit 25 kHz audio ΣΔ modulator,” *Faible Tension Faible Consommation (FTFC), 2014 IEEE*, pp.1-4, 4-6 May 2014.

59. Nguyen H. P., *FIR Filter Design: The Window Design Method*, http://cnx.org/content/m28304/latest/, 2009.