A Fully Digital Technique for the Estimation and Correction of DAC Error in Multi-bit Delta Sigma ADC

A DISSERTATION

SUBMITTED IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE AWARD OF THE DEGREE
OF

MASTERS OF TECHNOLOGY
IN
VLSI AND EMBEDDED SYSTEMS

Submitted by:

POOJA JAIN

2K17/VLS/16

Under the supervision of

PROF. D. R. BHASKAR

ELECTRONICS AND COMMUNICATION
DELHI TECHNOLOGICAL UNIVERSITY
(Formerly Delhi College of Engineering)
Bawana Road, Delhi-110042

JULY, 2019

ECE DEPARTMENT

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I, **POOJA JAIN**, **2K17/VLS/16** student of M.Tech (VLSI), hereby declare that the dissertation titled "**A Fully Digital Technique for the Estimation and Correction of DAC Error in Multi-bit Delta Sigma ADCs**" which is submitted by me to the Department of Electronics and Communication, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **POOJA JAIN**

Date:

ECE DEPARTMENT

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the dissertation titled "**A Fully Digital Technique for the Estimation and Correction of DAC Error in Multi-bit Delta Sigma ADCs**" which is submitted by **POOJA JAIN**, **2K17/VLS/16** [ECE Department], Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology/Bachelor of Technology, is a record of the project work carried out by the students under my supervision. To the best of my knowledge this work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi

Date:

D.R. BHASKAR

SUPERVISOR

PROFESSOR

Department of Electronics and Communication

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

ACKNOWLEDGEMENTS

A successful project can never be prepared by the efforts of the person to whom the project is assigned, but it also demands the help and guardianship of people who helped in completion of the project.

I would like to thank all those people who have helped me in this research and inspired me during my study.

With profound sense of gratitude, I thank Prof. D. R. Bhaskar, my Research Supervisor, for his encouragement, support, patience and his guidance in this research work.

Furthermore, I would like to thank my project manager Mr. Ankur Bal at ST Microelectronics for a fruitful internship program. I have been very honored and privileged to have worked under his supervision. I have benefited greatly from his deep intuition for digital circuits and systems, from his emphasis on clarity in presentation, and from his insistence on high technical and professional standards.

Thanks to my co-colleagues at ST Microelectronics for their help on numerous technical problems in my project. I would like to thank them for making critical suggestions and comments throughout the course of my project.

I take immense delight in extending my acknowledgement to my friend Abhishek and my family who have helped me throughout this research work.

POOJA JAIN

ABSTRACT

A new advanced strategy has been proposed and actualized to detect and remove DAC noise used in the feedback loop of Delta Sigma ADC comprising multi-bit quantizer. The technique has been intended for fast and high-resolution systems where the mismatch error shaping becomes incapable. The system has following highlights: First, it does not require any additional analog hardware. Second, the technique keeps running in the background and Third, the technique works under exceptionally low Oversampling Ratios (OSRs). The procedure can possibly enhance the ADC output by improving the Signal to Noise Distortion Ratio excessively near that of the perfect case. The strategy demonstrates its adequacy in evacuating the DAC noise and enhancing the resolving power of the test ADC under small OSRs. The presentation of the proposed method has been confirmed with MATLAB reproductions.

CONTENTS

CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

LIST OF TABLES

LIST OF SYMBOLS, ABBREVIATIONS

ADC	Analog to Digital Converter
DAC	Digital to analog Converter
DSM	Delta Sigma Modulator
OSR	Oversampling Rate
CIFFB	Cascade of Integrators with feed forward and feedback
DSP	Digital Signal Processing
RMS	Root Mean Square
SNR	Signal to Noise Ratio
NTF	Noise Transfer Function
STF	Signal Transfer Function
BTCC	Binary to thermometer code converter
ETF	Error Transfer Function
HPF	High Pass Filter
SCR	Scrambler
DWA	Data Weighted Averaging
COR	Correlation
Δ	Delta
*	Correlation symbol

CHAPTER 1

INTRODUCTION

In modern applications, Delta-Sigma ADCs comprising multi-bit quantizers are intended to satisfy the prerequisites of fast, high-resolution data conversions [1] – [4]. However, noise of the DAC utilized in the design can restrain the ADC output since ideal DAC does not exists. The structure of the DAC Error is demonstrated utilizing thermometer code based unit component DACs. Effect of the DAC noise on the overall performance of Sigma Delta ADC is examined. Different strategies and systems of managing DAC noise are being portrayed and based on the nature of error, a completely advanced procedure to measure the noise from the output of ADC and correct them is proposed. The adequacy of the system is portrayed with the help of simulation results. The technique is the basis for designing high speed, high resolving power delta sigma ADCs with less prerequisites on the analog circuits.

To justify the working of the above stated strategy, a continuous time third-order Delta Sigma Modulator was constructed. The suggested procedure was applied on the above system. The MATLAB simulations verify that the DAC noise measurement and removal technique is highly effective in removing the DAC Error from the DSM output and increase the resolution of the system.

1.1 Motivation

Generally, DSMs are mainly utilized for the transformation of thin-band signals, for example, voice (0-3kHz) and sound (0-20kHz) signals since DSM help in achieving the conversion at a minimum cost and give high resolution at the output [5]-[6]. In those cases, one-bit quantizer is usually utilized since the quantizer and the corresponding DAC used have perfect linearity. But the total quantization noise is large

in single-bit ADC which can be dropped under the desired level using methods like a high over-sampling ratio and noise shaping.

Nowadays, necessities of signal bandwidth are expanding (several MHz) with high resolution in applications like video capture and xDSL. Delta Sigma ADCs are best suited for these relevances because it has various features like high resolution, low cost and insensitivity to circuit non-idealities. In any case, to accomplish these, enormous OSR are unreasonable since the circuit should be quick and with low OSR, the noise shaping will be weak. Thus, we need to find different approaches to suppress the in-band noise power to accomplish good resolution.

Along these lines, Delta-Sigma ADCs are being used in which the quantizer utilized is multi-bit in nature. These quantizers help in improving the resolution in two ways [7].

1) They reduce the noise power which lies inside the band of interest (up to signal frequencies band) and total power.
2) They enhance the steadiness of the delta sigma systems.

So, multi-bit quantizers, though non-linear in nature, are used in many best in class sigma delta ADCs [8] [9] [10], which includes some best quality applications.

1.2 DAC Noise in Multiple bit Delta Sigma ADC

The utilization of quantizers having multiple bits cause a significant problem i.e., the DAC used in the loop additionally should be multi-level in nature which results in non-linearity of the DAC. This non-linearity can be viewed as an extra noise added to the perfect DAC output. The noise introduced by the DAC ventures up and down the way and appears in the ADC output after getting mixed with the input signal which directly restricts the linearity of the ADC and overall achievable resolution. For example, if the performance from the ADC is expected to be 14 bits, the in-band noise power should be 84dB underneath the full-scale signal power. The DAC noise makes it unattainable and requires some different strategies to be utilized [8].

A number of procedures have been given earlier to manage the DAC noise [11]-[19]. Dynamic element matching (DEM) is most widely utilized system [11]. It is fundamentally proposed for unit component based digital to analog converter structures [12]. By using DEM, the output bits of the quantizer which are thermometer coded are revised using certain standards before contributing to the DAC. The reworking process does not change the information value but it changes the selection criteria for the unit components used in DAC. The main impacts of these procedures are:

1) DAC noise and DAC input become uncorrelated amongst each other subsequent in removal of signal dependent tones in the ADC output.
2) The noise power gets shifted from low frequencies to high frequencies which is also known as noise shaping or mismatch shaping.

Mismatch shaping help in decreasing the impact of DAC noise just to a certain degree since shaping of the noise is dependent on the value of OSR. In very high-speed conversions, the shaping achieved from DEM becomes very inadequate for high resolution prerequisites where OSR values may have to be decreased to as low as 4.

Among different methods, Self-adjusting DACs are also being used where the unit components in the DAC are ceaselessly adjusted with the help of reference component [20]. However, in this strategy, a fine reference component is required and additional circuit is expected to alter the unit components. The foreground adjustment scheme runs the adjusting measurements during special clock cycles and stores the DAC noise data. During the normal activity of the ADC, the noise data is used to address the ADC output.

For the measurement of errors, the operation of ADC should be halted for a brief span. Various architectural changes have been acquainted to convert the foreground method into foundation one but that requires a double port DAC so that measurements of the error and normal activity of the ADC can be carried on a similar time. A correction technique in [21] incorporates an additional DAC unit component and a few changes in the delta sigma loop but results in complex plan.

1.3 Objective of the Work

The motivation behind this exploration was to locate a novel advanced system which can recognize and rectify the noise in the feedback DAC present in Delta Sigma ADC. The proposed strategy has key highlights: First, no additional analog circuits are required which will be an advantage since analog circuits benefit less from the scaling down in area and power when contrasted to digital circuits with the application of an ever-increasing cutting-edge innovations. Although analog circuits will dependably have their field of interest, nowadays digital circuits are allowed to take control over many more areas. Second, the procedure runs in the background. Third, this procedure can work for exceptionally low values of OSRs. Fourth, the procedure is versatile in nature as it can measure any percentage of noise introduced in the system. This feature separates the desired strategy apart from the current DEM techniques and meets the necessity of broadband conversions in many applications.

1.4 Organization of the Dissertation

Chapter 2 gives a concise introduction of quantization and sampling. The concepts of signal to noise ratio are being revisited. The impact of increased sampling frequency on the quantization error is studied and basics of sigma delta modulator are being covered which helps in reducing the quantization error. The need of quantizers having multiple bits in Delta Sigma ADC for high resolution systems is being pointed out. Finally, the disadvantages of utilizing DACs with multiple bits are described.

Chapter 3 describes the noise in multi-bit DAC using a straightforward approach. The effects of the DAC noise on the execution of DSM are shown and analyzed through the model. The different techniques of managing the DAC noise are talked about and their limitations are being highlighted.

Chapter 4 introduces a proposed procedure for locating and correcting the DAC noise. The center activity of the procedure is a correlation technique. Scrambling is performed for decorrelation of the input signal and the DAC noise, trailed by some post-

processing. Filtering is also performed to smother the input signal which accelerates the process of correlation.

Chapter 5 demonstrates the working of the proposed technique by actualizing a continuous time third order Delta Sigma Modulator in CIFFB fashion with low distortion. Impact of DAC error on the SNR of ADC output is depicted with the help of various simulations. The filters in the design are implemented in Polyphase fashion and filtering and other correlation operations are performed at low rate in contrast to the modulator which runs at high speed. Reproduction results using Simulink are also being shown to verify the system.

CHAPTER 2

LITERATURE SURVEY

There is an enormous demand for larger bandwidths and fast operation in modern wireless communication system design. Additionally, as of late, the use of wireless remote products has expanded in our everyday activities. Analog to Digital Converters (ADCs) assumes a noteworthy job in rapid wireless communication to accomplish these specifications. So, there is an immense enthusiasm for using data conversion procedures from the previous decade. ADCs are commonly expected to connect the analog and digital domains. There are numerous applications where ADCs are very important, for example, Radio Telecommunications, Diagnostic Imaging, Sound and Video Processing Systems, Instrumentation and Digital Radio Systems [23][24].

In general, numerous ADC models are accessible to meet these applications, which incorporate Parallel type, Sigma-Delta, Pipelined, Sub-ranging, Successive Approximation Register type and Folding ADCs and dependent on the application, and its details, proper selection of ADC can be made. However, not each ADC given above is suitable for every one of the applications [22]. In modern years, sampling rate (greater than 500 MHz) and resolution necessities are expanding to oblige for enormous data transfer capacity in numerous remote applications.

Delta Sigma modulation is a method in which high-resolution analog to digital data conversion is implemented in mixed signal designs with generally coarse circuit parts [1]-[5]. As a matter of fact, DSM data converters overwhelm mainly in designs which require high resolving power and perfect linearity in the system at less cost and speed. Despite the fact that the underlying thought of delta sigma tweak was raised decades prior, it had turned out to be mainstream in incorporated circuit structure simply after the implementation of enormous-scale digital circuits started on board since it

requires a lot of DSP [23] [24]. But ongoing applications are demanding higher speed transformations with no loss in resolution in delta sigma converters.

Data Converters (ADC and DACs) are divided into two sub classes that are Nyquist-rate data converters and oversampled data converters. Oversampled ADCs use progressively DSP functions to perform analog to digital conversion when contrasted with Nyquist-rate ADCs [22]. Also, the transition band of anti-aliasing filters is relaxed in oversampling ∑Δ ADCs. A basic model of ADC is being presented in the block diagram below in Fig. 2-1. In this part, the essentials related to delta sigma modulation are briefed, the various benefits of switching to quantization with multiple bits has been talked about, and the problems related with multi-bit quantization are surveyed.

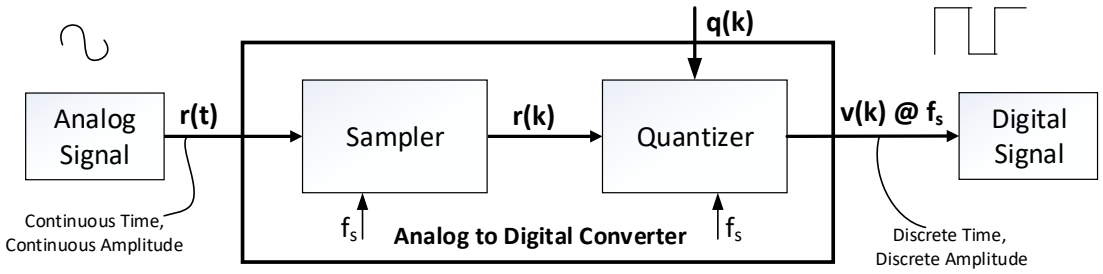

Fig.2-1 Basic Model of ADC

2.1 Sampling and Quantization

Often, we see that an original signal $r(t)$ is continuous in nature. That is, the time coordinate takes any arbitrary real values (maybe over some interim) and the value $r(t)$ of the signal also takes some random real values (again within some interval). Any signal with the above stated characteristics is called analog signals, which are continuous in both time and amplitude. In many situations, it is progressively advantageous dealing with digital signals. The signals which have a limited set of space and range. The way towards discretizing the area is called **sampling** and the way towards discretizing the range is called **quantization** [25].

Digital signals have numerous advantages over analog signals. They are particularly strong to noise, and very effective and adaptive methods for handling digital signals have been created in late time. Then again, analog signals are sometimes more

proper in specific situations. For example, most underlying practical procedures are analog (or are most conveniently demonstrated as analog) and even some sorts of information processing and transfer is also most helpfully carried out using analog signals. Along these lines, the transformation of analog signals to digital signals (and vice versa) is a significant part of many data handling frameworks.

2.1.1 Sampling a Signal

To begin, consider a sine wave signal $r(t) = \sin(\omega t)$ whose frequency ω is not too large i.e., the input is limited to low frequencies. If we sample input signal using a sampler circuit at fast rate that is if $\omega_s \gg \omega$ then we are in a position to represent our input signal using discrete number of input samples and can exactly recover $r(t)$. If the sampling frequency was chosen to be less than that of the sinusoid then reconstruction of the original signal will not be possible and this phenomenon is known as **aliasing** [25]**.** It turns out that if input signal frequency is f_B, then for reconstruction of the signal the sampling frequency is required to be at least double the input frequency i.e. $f_s \geq 2.f_B$, that is, if we sample at a faster rate which is twice or more than that in comparison to the input frequency of the sinusoid. This is also known as **Sampling Theorem** [26].

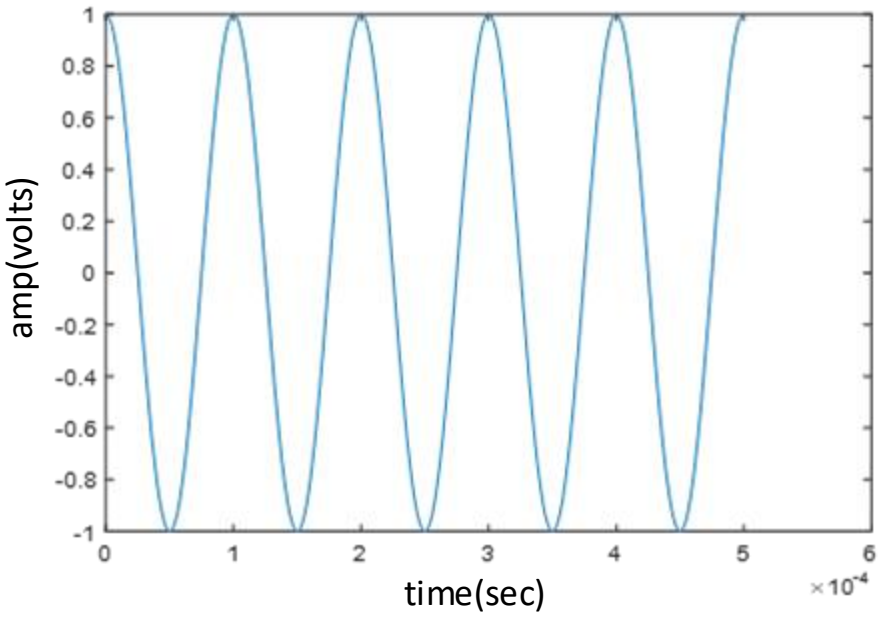

Fig. 2-2 Band Limited Input signal- r (t)

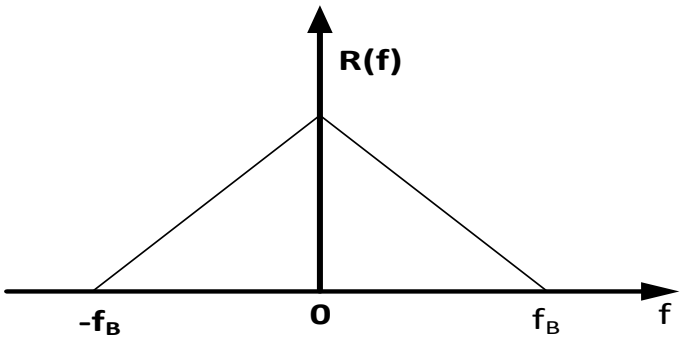

Fig. 2-3 Fourier Transform of Input Signal- r (t)

Fig.2-2 shows a band limited input sinusoid $r(t)$ having maximum frequency f_B. The Fourier Transform of input signal is presented in Fig.2-3, which shows the frequency content of the input signal. According to Sampling Theorem, for full reproduction of the input signal from its sampled version, we require the sampling frequency f_s to be equivalent to $f_s \geq 2.f_B$. The time domain and frequency domain transformation after sampling the input signal at $2.f_B$ are appeared in Fig. 2-4 and Fig. 2-5 individually. The Fourier transformation of a sampled signal consists of moved copies of R (f) at integer multiples of f_s. The frequency $2.f_B$ is also known as Nyquist rate or Nyquist frequency. Thus, we can conclude that to properly recover an input signal, we need to sample the signal at a rate equivalent to or more than Nyquist rate.

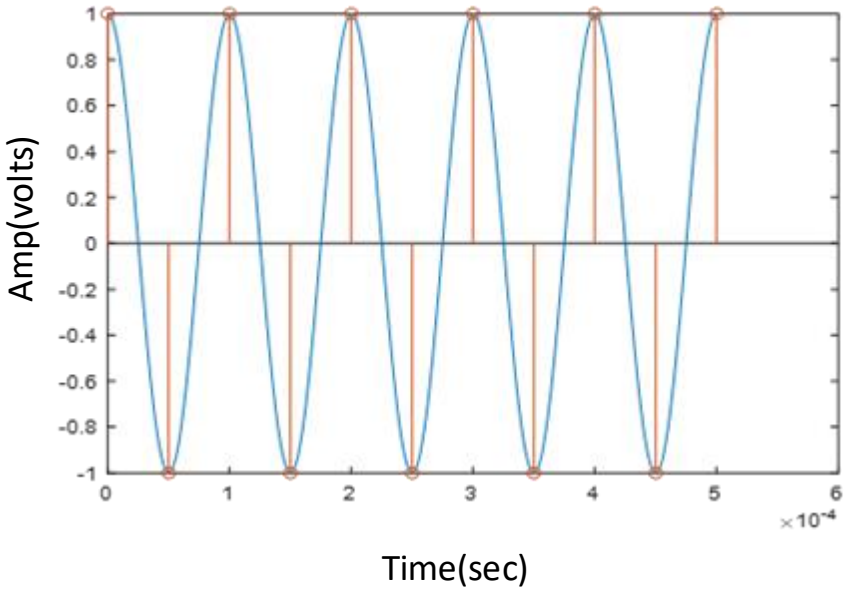

Fig. 2-4 Nyquist-rate sampled Input signal- r (t)

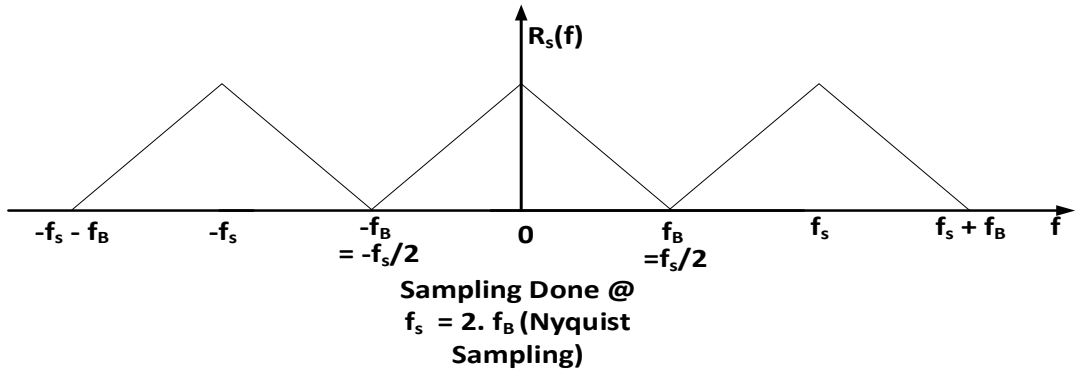

Fig. 2-5 Fourier Transform of Nyquist Sampling

2.1.2 Quantization

Quantization is a non-linear memory less phenomenon which is used to make the range (amplitude) of a signal discrete, so that the quantized signal takes only a limited set of discrete quantities. Quantization is an irreversible process and results in loss of data unlike sampling (in which precise reproduction of the signal is conceivable under reasonable conditions). Thus, the phenomenon of quantization injects deformity and error into the quantized signal that cannot be dispensed with [25].

One of the basic decisions in quantization is the quantity of discrete quantization levels to be utilized. As the quantity of steps in the quantizer increases, the amount of data used to represent each sample increases and thus the quality of the resulting signal also improves. The transfer curve of a quantizer is a staircase, which is generally uniform in nature i.e. the distinction between the levels is always fixed and is denoted by delta(Δ) [25].

The output of a quantizer $v(k)$ can be represented as the input signal $r(k)$ plus some distortion or noise introduced by the quantizer $q(k)$ (also termed as Quantization Noise) as depicted in Equation (2.1) i.e.

$$v(k) = r(k) + q(k) \tag{2.1}$$

$q(k)$ is the Quantization error injected by the quantizer which is equal to the distinction between the input and yield of the quantizer.

The transfer curve of a quantizer is presented underneath and the connection between the input signal and the quantization noise is also presented below in Fig.2-6. It tends to be seen that quantization noise remain systematically distributed between $-\frac{\Delta}{2}$ and $+\frac{\Delta}{2}$ as long as the sampled input signal $r(k)$ remain equitably distributed in the interval from $-V_{REF}$ and $+V_{REF}$ [27] where Δ is the quantization step also known as step size. The formula to calculate quantization step is given by Equation (2.2) below:

$$\Delta = \frac{\text{peak to peak value of input signal}}{2^{\mathrm{N}}}$$

$$\Delta = \frac{2.\mathrm{V_{REF}}}{2^{\mathrm{N}}} \tag{2.2}$$

In Equation (2.2) above, N is the quantity of bits used to represent levels of the quantizer and V_{REF} is the reference voltage of the input signal [27].

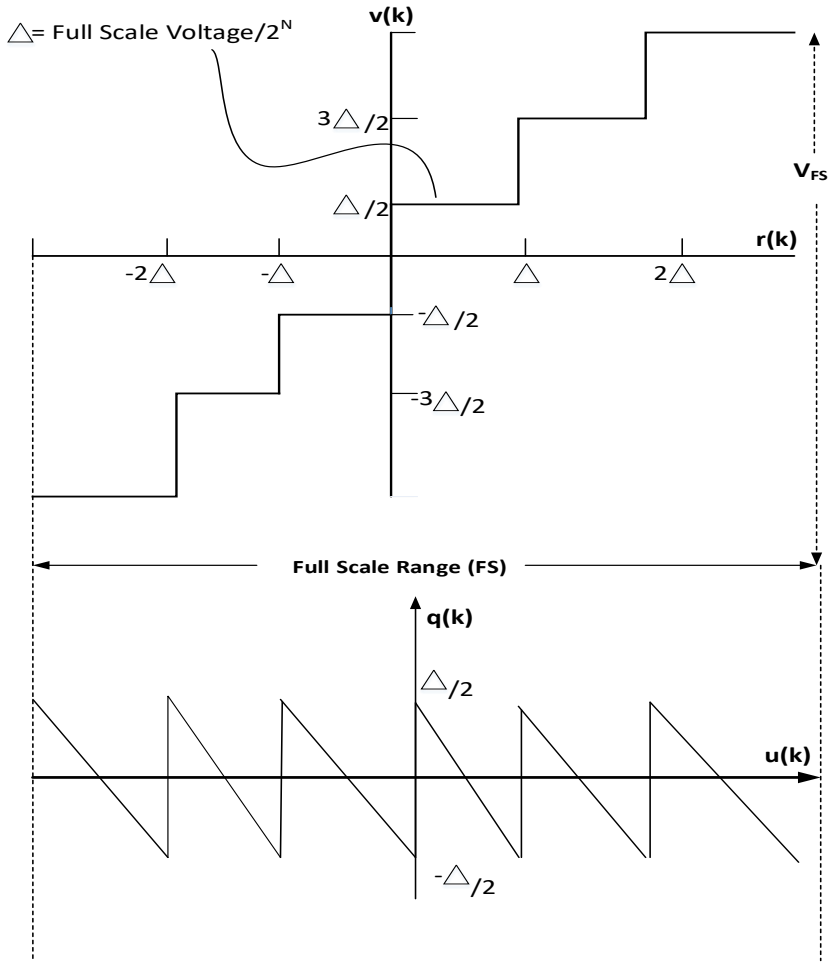

Fig. 2-6 Transfer curve of Quantizer

If the quantizer's range is N bit then the full-scale value of the input signal which will not overload the quantizer will range from $-2^{N-1}.\Delta$ to $+2^{N-1}.\Delta$

As already discussed, the quantization error lies between $-\frac{\Delta}{2}$ and $+\frac{\Delta}{2}$ for a known value of Δ presented in Fig. 2-6 below. The RMS value of Quantization Noise can be calculated by integrating the error over the error range as depicted in Equations (2.3) below:

$$v_N = \sqrt{\frac{1}{\Delta}\int_{-\frac{\Delta}{2}}^{+\frac{\Delta}{2}} e^2.de}$$

$$v_N = \sqrt{\frac{1}{\Delta}\left[\frac{e^3}{3}\right]_{-\frac{\Delta}{2}}^{+\frac{\Delta}{2}}}$$

$$v_N = \sqrt{\frac{\Delta^2}{12}} = \frac{\Delta}{2\sqrt{3}} \qquad (2.3)$$

The equation describes the noise power in terms of quantization step size or LSB width. Similarly, root mean square signal power can also be calculated as below. As the signal voltage varies from $-V_{REF}$ to $+V_{REF}$, thus the signal power will be equal to:

$$v_S = \sqrt{\frac{(Full\ Scale\ Range\ of\ input)^2}{8}}$$

$$v_S = \sqrt{\frac{4.{V_{REF}}^2}{8}}$$

Now replacing the V_{REF} by the full-scale value calculated using Equation (2.2), we get Equation (2.4)

$$v_S = \sqrt{\frac{4.(-2^{N-1}.\Delta)^2}{8}}$$

$$v_S = \sqrt{\frac{2^{2.N-2}.\Delta^2}{2}} = \sqrt{2^{2.N-3}.\Delta^2} \qquad (2.4)$$

The quantization noise will lie in a similar band as the signal if we consider the sampling frequency equivalent to the minimum rate of sampling. Neglecting the noise from other sources, the maximum Signal to Noise Ratio achievable is given by Equation (2.5) and also shown in Fig. 2-7

$$SNR_{MAX} = \left(\frac{v_S}{v_N}\right)^2$$

$$SNR_{MAX} = \frac{2^{2.\mathrm{N}-3}.\Delta^2}{\frac{\Delta^2}{12}}$$

$$SNR_{MAX} = \frac{3}{2}.2^{2N}\ which\ in\ dBs\ is$$

$$SNR_{MAX} = 6N + 1.76\ (dB) \qquad (2.5)$$

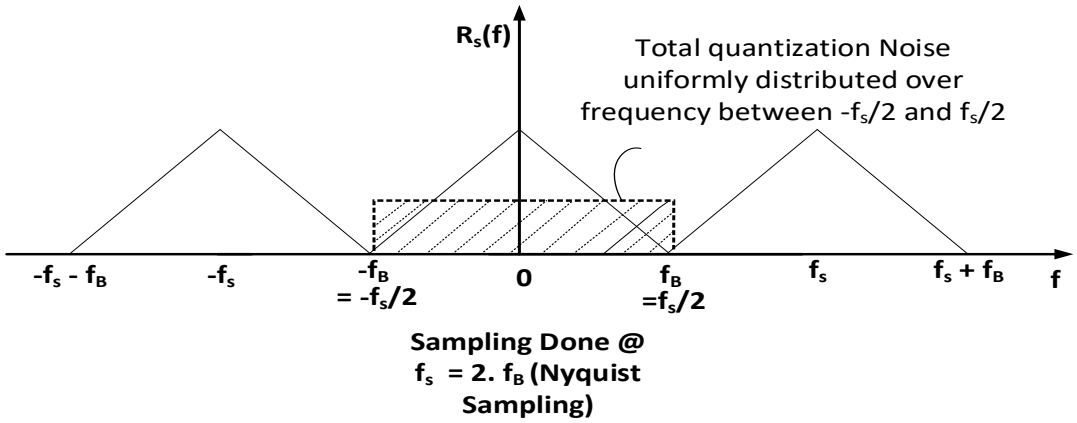

Fig. 2-7 Quantization Noise in case of Nyquist Sampling

2.2 Oversampling and Noise Shaping

If the frequency of sampling the signal f_s is expanded beyond the Nyquist rate then it is known as Oversampling [24]. The ratio of sampling frequency (f_s) to the Nyquist frequency (two times of maximum input signal frequency) is known as Oversampling Ratio (OSR). The transformations after oversampling in time and frequency domain are appeared in Fig. 2-8 and Fig. 2-9 individually.

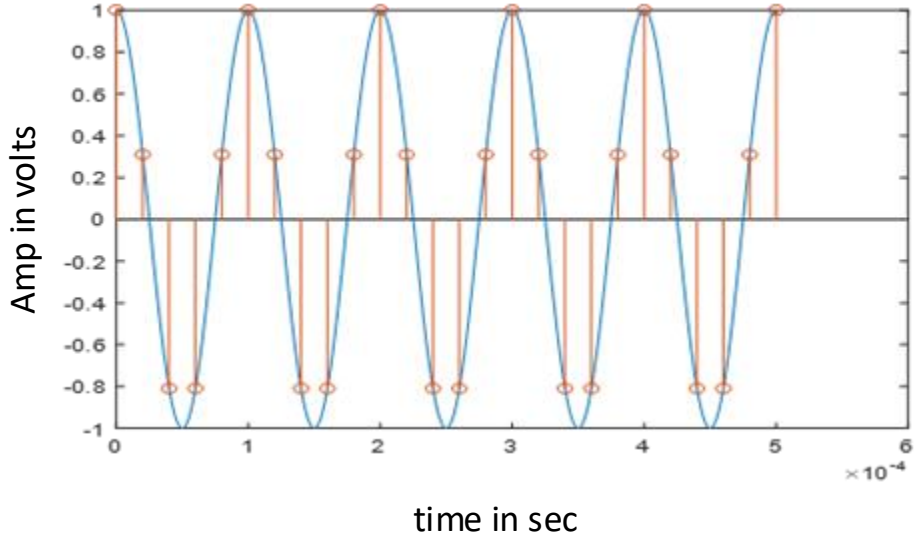

Fig. 2-8 Oversampled Input signal- r (t)

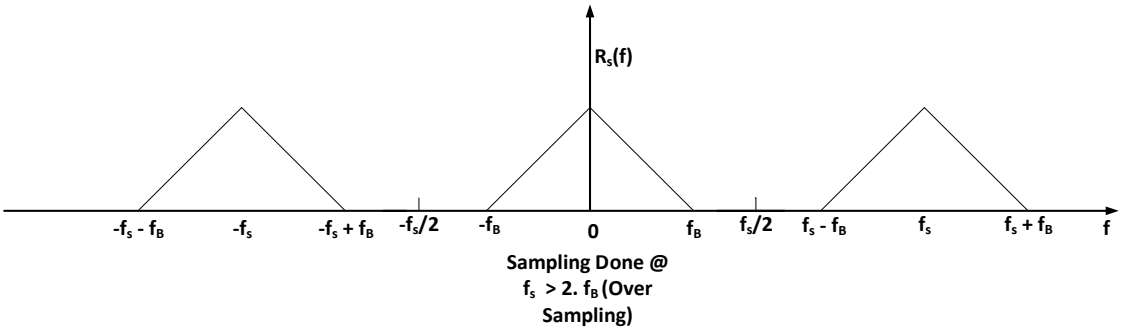

Fig. 2-9 Fourier Transform of Oversampled Signal

Increasing the OSR value has many favorable circumstances listed below:

1. It helps in relaxing the transition band requirements for anti-aliasing filters used in the design.
2. The Noise power lying in the base band will be significantly diminished while the absolute noise stays same.
3. Also, low power and low cost digital filtering can be applied in the systems having large OSR.
4. For 2X increment in OSR, the Quantization Noise power reduces by 3dB and effectively increase of 0.5 bits resolution is achieved.

For a quantizer with quantization step Δ and sampling frequency f_s, Quantization Noise gets uniformly distributed across Nyquist Bandwidth $f_s/2$. For oversampled signals, the value of $f_s/2$ will be much prominent than the signal frequency leading to a lot of noise power lying away from the signal band of interest as appeared in Fig. 2-10. While calculating SNR, the noise power which lies in the signal band is considered for calculations. Thus, the maximum signal to noise ratio is expanded under oversampling condition compared to SNR achievable in Nyquist rate sampling.

Under Nyquist rate sampling, the quantization noise possesses a similar band as signal band. Thus, the Noise power as calculated above in Equation (2.3) gives $\overline{e^2} = \frac{\Delta^2}{12}$. The Spectral Density of Noise Power is given by:

$$N_e(f) = \frac{\overline{e^2}}{f_s} = \left(\frac{\Delta^2}{12}\right).\frac{1}{f_s}$$

The Noise power in signal band is given by S_B as demonstrated below:

$$S_B = \int_{-f_B}^{+f_B} N_e(f)df = \int_{-f_B}^{+f_B} \left(\frac{\Delta^2}{12}\right).\frac{1}{f_s}df = \left(\frac{\Delta^2}{12}\right).\frac{2.f_B}{f_s}$$

For Nyquist rate sampling, $f_B = \frac{f_s}{2}$ which results in $S_{B0} = \frac{\Delta^2}{12}$. Now, considering the instant of Oversampled ADCs, the Noise power in the band of interest will be fewer than in the case of Nyquist rate sampling by Oversampling ratio:

$$S_B = S_{B0}.\frac{2.f_B}{f_s} = S_{B0}.\frac{1}{OSR}$$

Where OSR = $\frac{f_s}{2.f_B}$, Thus we can say that increase in OSR ratio decreases the amount of noise power in the band of interest.

Thus, improvement in Signal to Noise ratio is seen in case of Oversampling ADCs and is given by Equation (2.6):

$$SNR_{MAX} = 6N + 1.76 + 20log_{10}OSR\ (dB) \tag{2.6}$$

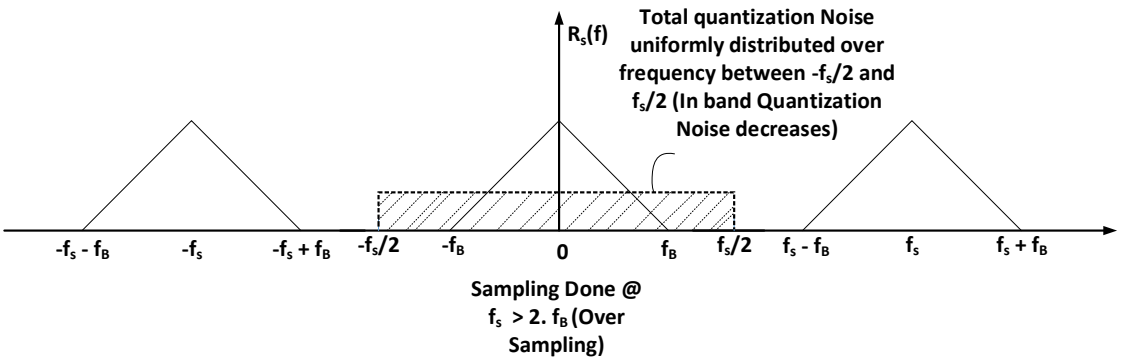

Fig. 2-10 Quantization Noise in Oversampled Signal

This implies that the SNR is boosted by 3dB/octave with the doubling in OSR value and an effective increment of 0.5 bit resolution.

Noise Shaping is a procedure by which the noise in any system is high passed or shaped out of the band of interest [27]. To get insight details about noise shaping, consider a negative feedback system with an amplifier as shown in Fig. 2-11. Let e be the error introduced by the amplifier in the system. The output of the system will be given by Equation (2.7):

$$v = \left(\frac{A}{1+A}\right).r + \left(\frac{1}{1+A}\right).q \tag{2.7}$$

As the value of A increases, $\frac{1}{1+A}$ starts decreasing. Hence, the v starts approaching r and the component of q in v decreases. If A→∞, then v = r and q does not affect the output.

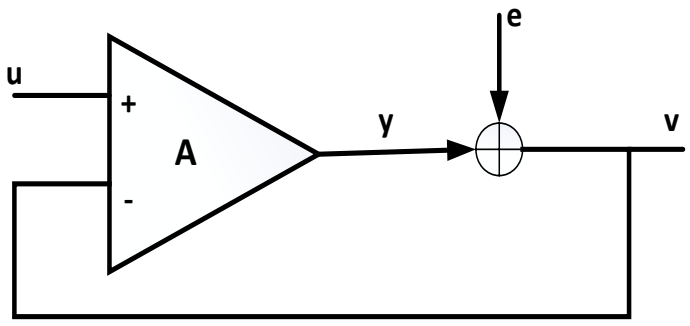

Fig. 2-11 Negative Feedback Loop System

Now if we consider the amplifier to be noiseless and embed a quantizer in the loop (since quantizer introduces quantization error), we can dispose of quantization noise but to achieve this, we need to make the value of A very large shown in Fig. 2-12 below.

One more point to note is that every device takes a finite time to operate and produce outputs. Thus, the output of Quantizer will be available to the amplifier A only in the next sample [27]. Therefore, a delay of one sample must be inserted into the feedback loop as shown in the Fig. 2-12.

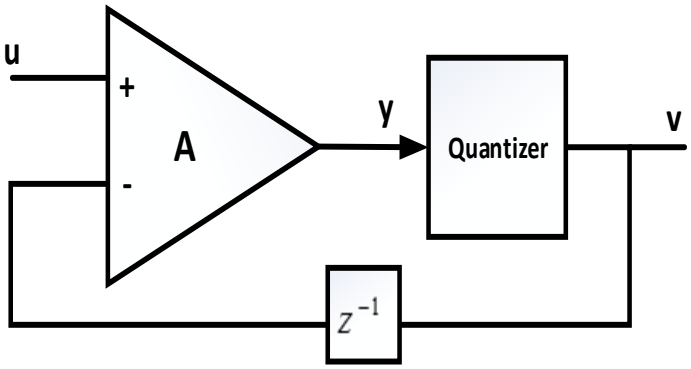

Fig. 2-12 Negative feedback system with non-zero delay

Consider Fig.2-12 shown above in which a Quantizer is embedded in the loop. The output of the feedback system is given by Equation (2.8).

$$V(z) = \big(R(z) - z^{-1}.V(z)\big)A + Q(z) \tag{2.8}$$

Which can be simplified as Equation (2.9)

$$V(z) = \left(\frac{A}{1 + Az^{-1}}\right).R(z) + \left(\frac{1}{1 + Az^{-1}}\right).Q(z) \tag{2.9}$$

In the Equation (2.9) above, we call the elementary function from r to v as Signal Transfer Function (STF) and the elementary function from q to v as Noise Transfer Function (NTF). As the quantity A→∞, STF approached unity and the NTF becomes equal to zero. Now as we can see that from Equation (2.9) above that the denominator of

both STF and NTF are same and the location of poles can be discovered out by determining the characteristic roots of the equation and is given by $z = -A$.
All the poles of the denominator polynomial should be enclosed inside the unit circle for a stable discrete-time system. For the steady framework, we need to guarantee that $|A| \leq 1$.

Thus, we get ourselves in a situation where we cannot achieve maximum Noise Suppression and get stabilized system at the same time. If our aim is to achieve maximum noise suppression then we need to maximize the amplifier gain i.e. $A >> 1$ but by doing so, the position of poles of the system changes to outside the unit circle leading to an unstable system. Thus, we need to settle in the middle to achieve both of the advantages at minimum cost to the system.

A possible solution will be to provide large value of gain A ($A = \infty$) only for low frequencies rather than for all frequencies (corresponds to the functioning of an ideal integrator). The reason being the input signal confined to only low frequencies of the spectrum due to oversampling of the signal and we need to reduce the quantization noise for low frequencies only (only in signal band) by increasing the gain at low frequencies only.

This results in two effects:

1) The quantization error will be greatly reduced with the help of feedback loop for low frequency parts of the signal.
2) At the same time, the stability of the system is not undermined since the gain is not increased beyond the point of stability.

Consider Fig. 2-13 below which shows a basic model where the frequency independent block A shown above is replaced by a frequency dependent block which provides infinite gain for low frequencies only resulting in very low magnitude of NTF at low frequencies (corresponding to signal band region). Also, the quantization noise is being demonstrated as a straight additive noise in the framework for simplicity of calculations.

The value of amplifier gain (A) is being replaced by $A = \frac{1}{1-z^{-1}}$ which corresponds to a lowest order Integrator. Also, the pole of the integrator lies at z = 1 which is on the unit circle thus the system is not unstable also.

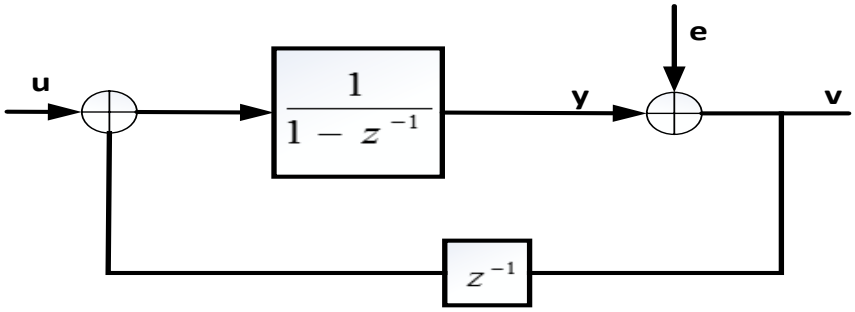

Fig. 2-13 First-order Noise Shaped Converter

Putting the value of A in Equation (2.9) above, we get Equation 2.10

$$V(z) = (1).R(z) + (1 - z^{-1}).Q(z) \qquad (2.10)$$

The STF according to the equation above is unity and NTF which is equivalent to $(1 - z^{-1})$, is Ist-order high pass response with a null at dc (z = 1 or w = 0). Thus we can conclude that if a quantizer is embedded in the loop and low frequency input signal is applied to the input, then the effect of quantization error on the input is significantly reduced as all the noise is shaped out of signal band or high pass filtered (as compared to the situation where no feedback loop exists). This is known as **Noise Shaping or First Order Noise Shaped Converter** also known as **First-order $\Delta\sum$ converter** [27]. The frequency domain shaping is shown in the Fig. 2-14 below.

The only condition for Noise shaping to be applied is that the signal must be oversampled i.e. Oversampled signals can only be Noise Shaped.

The in-band noise (when oversampling and feedback loop are used) can be generated as given in Equation (2.11) below:

$$\text{In Band Noise } = \frac{\Delta^2}{24\pi}\int_{-\frac{\pi}{OSR}}^{\frac{\pi}{OSR}} \left|(1 - e^{-j\omega})\right|^2 d\omega$$

$$= \frac{\Delta^2}{12\pi}\int_0^{\frac{\pi}{OSR}} 4sin^2\left(\frac{\omega}{2}\right)d\omega$$

$$\approx \frac{\Delta^2}{12\pi}\int_0^{\frac{\pi}{OSR}} \omega^2 d\omega = \frac{\Delta^2}{36\pi}\frac{\pi^3}{OSR^3} \qquad (2.11)$$

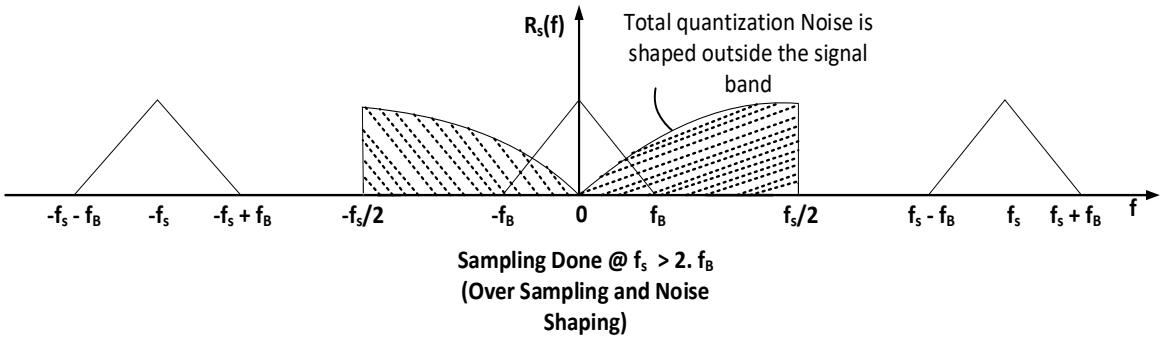

Fig. 2-14 Quantization Noise Shaped out of signal band

We can see that the in-band noise contributed by the quantizer is inversely proportional to $\propto OSR^3$ when Noise Shaping is applied along with oversampling. Thus, doubling the OSR will result in the reduction of Quantization Noise by 8 which will be equivalent to decrease in in-band noise power by 9 dB (since log (8) = 3log (2) = 3*3 dB).

As we have earlier seen that an increase of 6 dB is achieved in SNR by increasing 1-bit resolution of quantizer. Thus, by doubling the OSR, an effective increase in resolution of 1.5 bits is increased (since 9 dB is equivalent to 1.5 bits resolution).

Thus, we can conclude that:

1) High Resolution can be achieved (Quantization Noise can be decreased) by using a very high quantity of OSR but achieving such value is not practical. (Doubling of OSR will result in increase of 0.5-bit resolution).

2) On the other hand, same resolution can be produced if oversampling is combined with noise shaping and the value of OSR needed will be very less as compared to the above one. (Doubling of OSR will increase the resolution by 1.5 bit).

2.3 Basics of Delta-Sigma Modulators

Delta-sigma modulation is a standout amongst the best forms of analog to digital or digital to analog conversion in data conversion world. Its applications incorporate communication frameworks, professional sound and accuracy measurements. The objective of delta-sigma modulation is to produce higher transformation proficiency by passing only the distinction (delta) in present and previous values of samples rather than the genuine samples themselves. ADCs and digital-to-analog converters (DACs) both can utilize delta-sigma modulation [28].

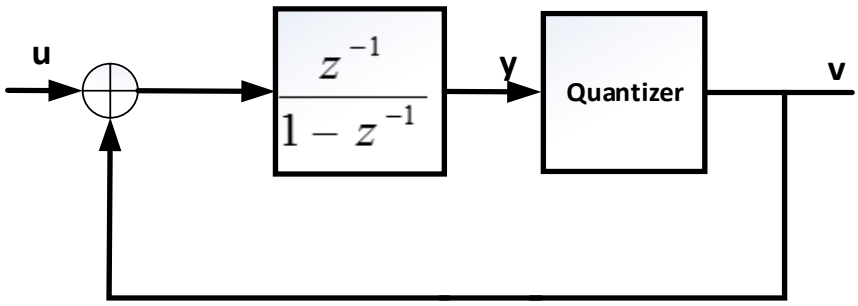

Fig.2-15 First Order Delta Sigma ADC

Delta-Sigma Modulator or Delta-sigma ADC is an oversampling type modulator which is highly immune to analog circuit imperfections making it one of the best choices for the designing of ADC interfaces in modern day SoCs. The Delta-Sigma ADCs are used for shaping out the quantization noise (Quantization Noise Shaping) of signal band in very low power and high resolving nature applications. By doing so, the effective Signal to Noise Ratio is increased inside the signal band i.e. High Resolution of the system. The basic First-order Delta Sigma Modulator is being discussed in the area above [27]. It can also be modelled as shown in Fig.2-15 above.

Now we need to model quantizer for the Delta sigma modulator. As we know that Quantizer takes analog input and divides its range into levels. But at the same time,

we need to generate the analog output for those levels so that the output can be combined with the incoming input in the next sample. One way of modelling the practical Quantizer in Delta Sigma ADC is by cascading an ADC and DAC as shown in Fig. 2-16 and the Delta Sigma Modulator is appeared in Fig. 2-17. ADC and DAC added will result in some nonlinearities since the threshold levels of the ADC and DAC will be slightly shifted from their ideal values in practice [27]. These nonlinearities can be demonstrated as straightforward noise added to the system for simple mathematical calculations.

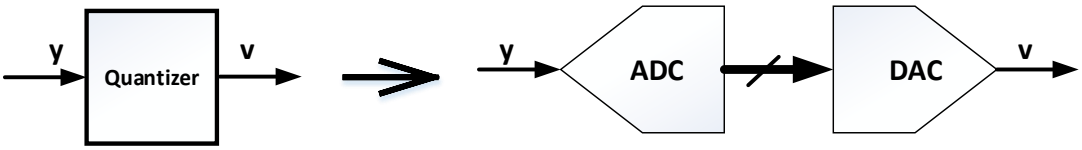

Fig. 2-16 Modelling of Quantizer as a cascade of ADC-DAC

The model of First-Order Delta-Sigma Modulator is presented along with the noise that are e_{adc} introduced by the ADC (accounting for the non-uniform spacing between the levels of ADC), e_{dac} introduced by the DAC (accounting for the unequal gaps between the levels of DAC) and e_{quant} which is the noise resulting from the quantization of analog input signal to digital levels is shown in the Fig. 2-18.

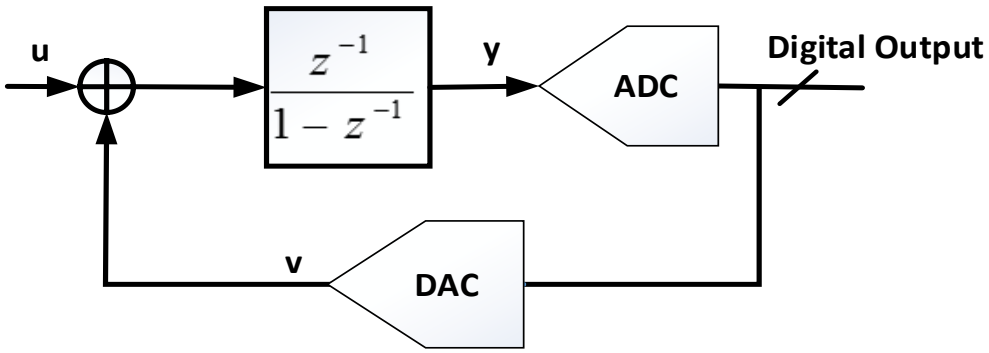

Fig. 2-17 Quantizer as ADC-DAC Cascade

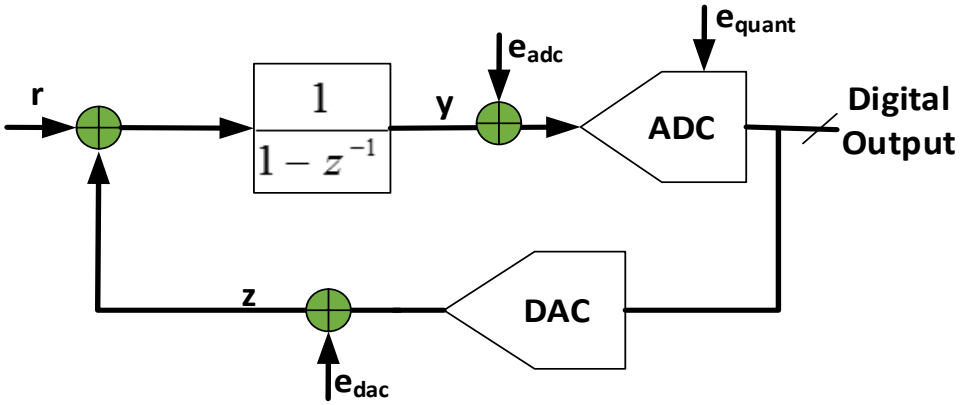

Fig. 2-18 Delta Sigma ADC with e_{dac} and e_{adc}

As already discussed in the past section of Noise Shaping, the error is shaped by the negative feedback loop out of signal band. In Fig. 2-18 above, we can see that e_{adc} just adds to the quantization noise e_{quant} and thus will not be a problem to the system (since it will also be shaped by the negative feedback loop along with quantization error). On the other hand, e_{dac} is problem for the system as it gets accumulated in the input signal and changes the output of the DSM (since it cannot be separated from the input signal) leading to degradation of the Signal to Noise Ratio at the output of DSM [27].

The effect of DAC error can be seen with the help of frequency domain analysis as shown in the Fig. 2-19. It is assumed that the DAC noise introduced in the system is a white noise spread over from 0 to $f_s/2$. The DAC noise does not get shaped out of signal band as shown in Fig. 2-20. Thus, we need to get rid of DAC Error component by some means in order to prevent the SNR from degradation.

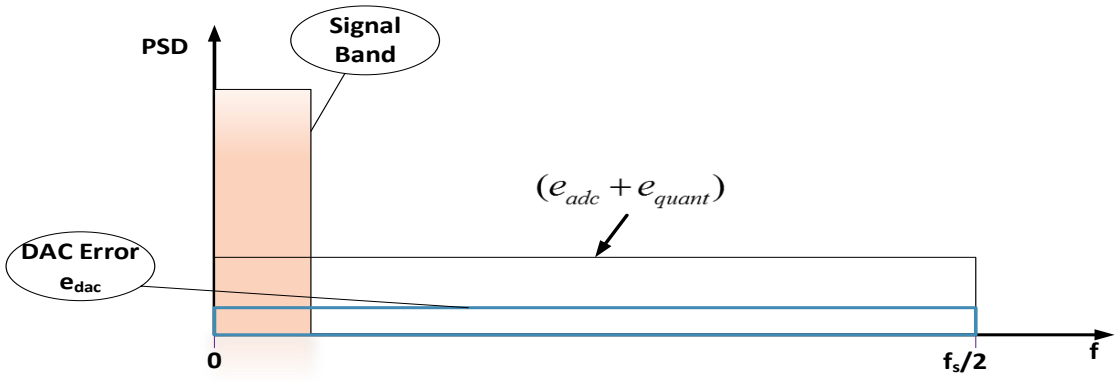

Fig. 2-19 PSD without shaping e_{dac} and e_{adc}

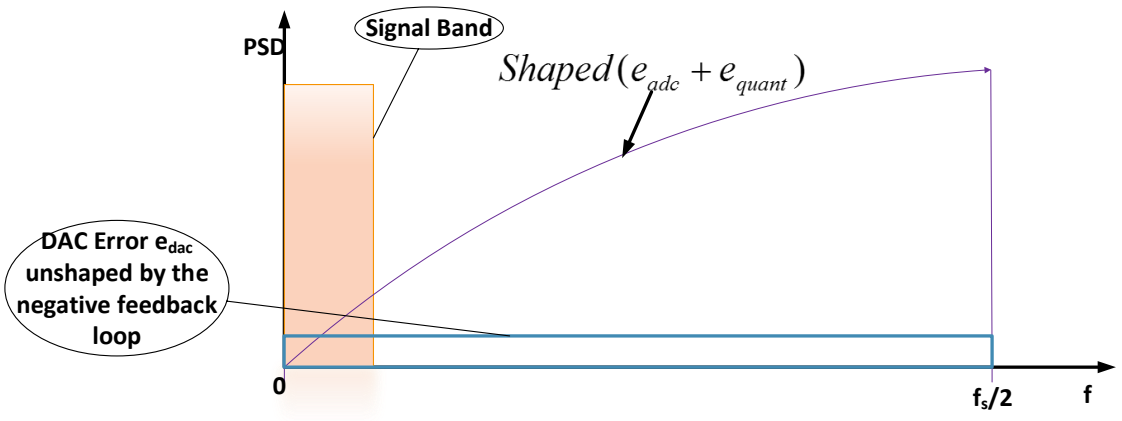

Fig. 2-20 PSD with noise shaping of e_{adc} and e_{quant}

The Delta Sigma Modulator discussed so far is first order DSM. The Noise Shaping for the ADC and quantization noise components can be made more aggressive by increasing the order of DSM used. For clarity, IInd order DSM is being appeared in Fig. 2-21 and the corresponding noise shaping achieved as compared to first order DSM is shown in Fig. 2-22. Similarly, higher order DSM can be realized to achieve more noise shaping for the noise components [27].

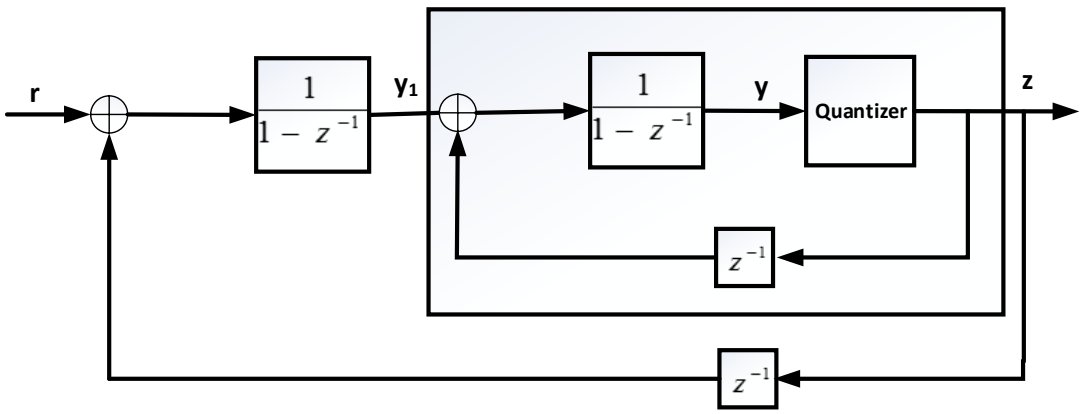

Fig. 2-21 IInd-Order Delta Sigma ADC

Consider the above demonstrated IInd-order DSM. The yield of the modulator will be given by Equation (2.12) below:

$$V(z)_{IInd\_order} = z^{-1}.R(z) + (1 - z^{-1})^2.Q(z) \qquad (2.12)$$

On studying the above equation, we can infer that as the order of modulator is expanded, increasingly forceful noise shaping is obtained as the NTF is modified.

$$STF(z)_{II} = z^{-1} \; and$$

$$NTF(z)_{II} = (1 - z^{-1})^2$$

Mathematically, the in band noise component for second order DSM will be:

$$\text{In Band Noise} = \frac{\Delta^2}{24\pi} \int_{-\frac{\pi}{OSR}}^{\frac{\pi}{OSR}} |NTF|^2 d\omega$$

$$= \frac{\Delta^2}{24\pi} \int_{-\frac{\pi}{OSR}}^{\frac{\pi}{OSR}} \left| \left(1 - e^{-j\omega}\right)^2 \right|^2 d\omega$$

For very low frequencies, the magnitude of NTF will be equal to ω, thus the equation will be modified to (2.13) given below:

$$= \frac{\Delta^2}{24\pi} \int_{-\frac{\pi}{OSR}}^{\frac{\pi}{OSR}} \omega^4 d\omega$$

$$= \frac{\Delta^2}{24.\pi} \left[\frac{\omega^5}{5} \right]_{-\frac{\pi}{OSR}}^{\frac{\pi}{OSR}} = \frac{\Delta^2}{60.\pi} \left(\frac{\pi}{OSR} \right)^5 \qquad (2.13)$$

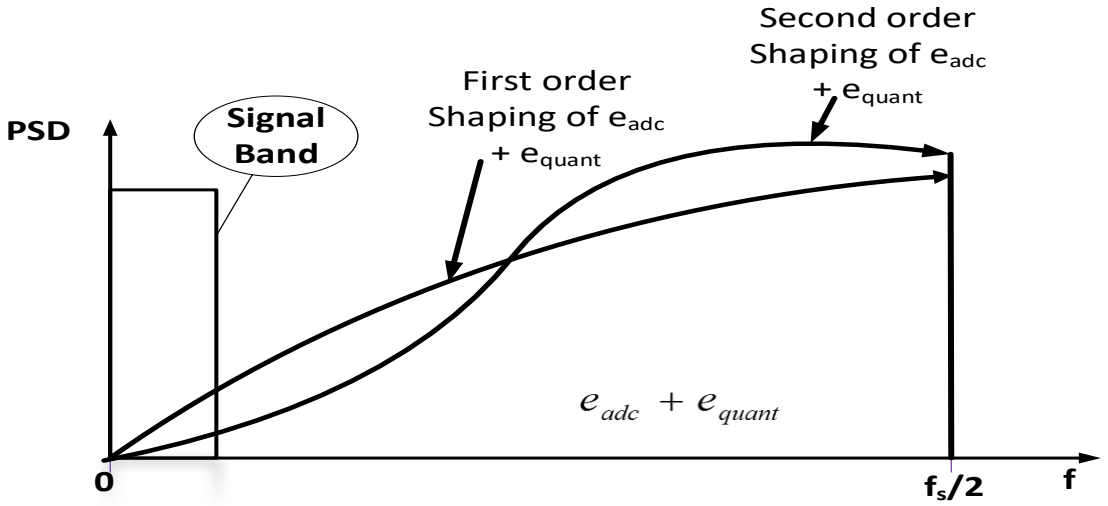

Fig. 2-22 Noise Shaping using First-order and Second-order DSM

Thus, in band noise will be proportional to $\propto OSR^5$. For a second order DSM, Doubling the OSR will reduce the in band quantization noise by 15 dB which effectively increases the resolving power by 2.5 bits (while in first order DSM-only 1.5 bits resolution was achieved and with only oversampling 0.5 bits resolution) depicted in the Fig. 2-22 below with the help of frequency domain diagram [27].

Thus, generalizing the in band quantization noise for Lth order DSM, Doubling the OSR of such modulator will result in decrease of quantization noise by 3.(2.L+1) dB and increase in resolution by (L + 0.5) bits (since the in band quantization noise is proportional to $\propto OSR^{2L+1}$)

Thus, we can see that by using oversampling and Noise shaping techniques, the quantization noise is moved from in band to out of band (or from low frequencies to high frequencies). The out of band noise is sifted by the Decimation filter used in the process before down sampling so that the noise will not fold back in signal band during down sampling process. Thus, the maximum SNR achieved using Oversampling and Noise Shaping is given by Equation (2.14):

$$SNR_{max} = 6.02N + 1.76 + (20\mathrm{L} + 10)\log_{10}OSR - 10\log_{10}\frac{\pi^{2L}}{2L+1} \qquad (2.14)$$

Where L = DSM order, OSR stands for Over Sampling Ratio and N = no of bits of the quantizer [27].

2.4 Single Bit vs Multi-bit Quantization

So far, all the basic concepts of Delta Sigma Modulator have been covered. Using Delta Sigma Modulator, we can achieve high resolution conversions rates. There are three ways of achieving high resolution in Sigma Delta Modulators which includes a high value of OSR, a higher order DSM and a quantizer comprising multiple bits resolution. Now we will try to check the feasibility of all the three methods to achieve high resolution in broadband and narrow band conversions.

In narrowband conversions, we can use high values of OSR to achieve high resolution conversions. But the problem arises in broadband conversions. Today all applications including video capture are requiring broadband conversions with high resolution. In broadband conversions, high value of OSRs are not practical since the sampling rate becomes too fast to be implementable. Also, if somehow by means of advanced technologies we could achieve high value of OSR implemented using switched capacitor circuits, it puts a pressure on the power consumption.

A high order loop has no significance without high OSRs. The only option left is using multi-bit quantizers. By using multi-bit quantizers, the resolution of quantizers is increased which lowers the total noise power. This results in two benefits: The in-band power of noise is reduced. Also, it constricts the in-band noise and intensifies the out-of-band noise to a large extent, which improves the SNR further i.e., more aggressive designing of NTF before the ADC becomes instable.

Nevertheless, multi-bit quantizers do have their own problems. If a quantizer which has multiple bits is used in the design then a non-linear DAC (having multiple bits) will be required in the feedback loop. As we have already seen in the previous sections that the error introduced due to nonlinear behavior of the feedback DAC is not shaped by the feedback loop. The error injected by the DAC limits the overall resolution of the ADC and linear behavior of ADC. If a single bit DAC is used, the error of a single-bit DAC can be separated into gain and offset noise which does not influence the presentation of the ADC. Thus, we can say that a single bit DAC has impeccable linear nature. But a multi-bit DAC error cannot be simply decomposed into gain and an offset and is not so simple. We have already seen that the DAC error can possibly debase the ADC execution and is covered in next chapter.

2.5 Poly-phase Filters

Poly phase Filtering is an efficient way of designing Filter structures for applying sampling-rate conversion and filtering to a signal. It helps in reducing the cost and complexity of the filter by changing the order of decimation/interpolation and filtering which reduces the number of computations required. Thus, we can say that Poly phase filters help in efficient implementation of systems like A/D Converter and D/A

Converter [29]. Fig. 2-23 shows the logical order of operations to be performed for filtering and resampling (say down sampling) in signal processing.

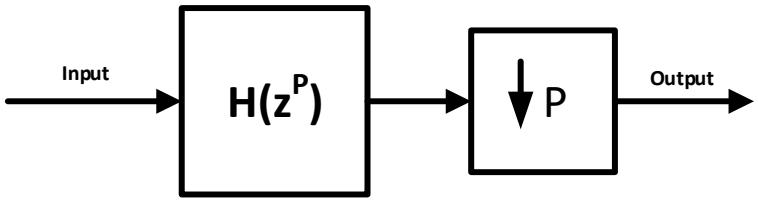

Fig. 2-23 Filtering followed by Down sampling

Performing the operations in this order means convolution between the incoming signal and the filter coefficients and then throwing away some of the computed results. This is an inefficient way of performing the operations since it leads to extra computations performed which are discarded after resampling. But if the order of operations is reversed then it will lead to aliasing and hence bad results.

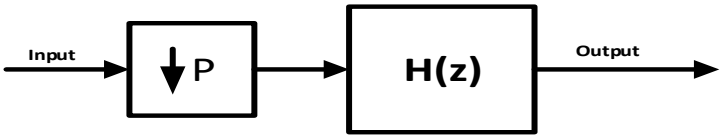

Fig. 2-24 Down sampling followed by Filtering

We can perform the operations in the order of down sampling and then filtering after applying some reordering techniques to the filter coefficients such that no aliasing takes place. This is known as Polyphase Filtering technique [29] as shown in Fig. 2-24.

Thus, to perform down sampling prior to filtering, we need to reorder our filter. Consider a FIR filter of 6 taps given by h = [h0 h1 h2 h3 h4 h5] and a signal x of length 9 is applied to the filter given by x = [x0 x1 x2 x3 ….x8]. The output of the filter will be decimated by P = 3. Then, the polyphase decomposition of filter will comprise of 3 filters with two coefficients each.

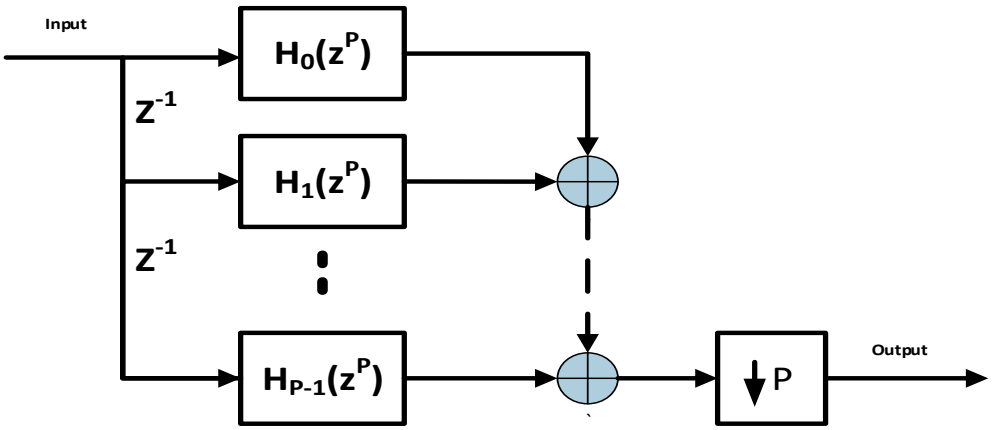

Fig. 2-25 Polyphase Implementation of Filter

The polyphase implementation of filter structure will appear like as shown in Fig. 2-25 consisting of P filters having filter coefficients reordered. Applying the noble entity for decimation, the down sampler can be moved before the poly-phase filter resulting in efficient implementation of filtering and re-sampling process shown in Fig. 2-26. The same procedure can be applied to interpolation where up sampler is moved after the poly-phase filters for efficient implementation.

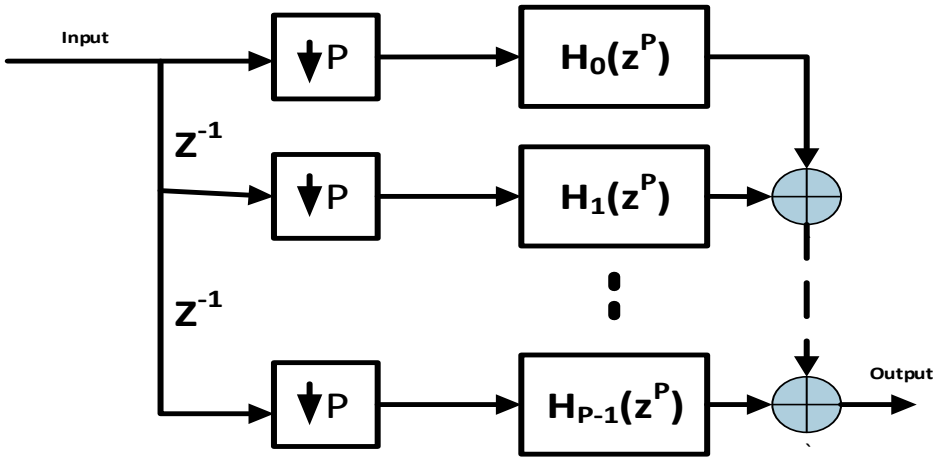

Fig. 2-26 Polyphase Implementation using noble entity

CHAPTER 3

DAC MISMATCH NOISE

`

In this chapter, the noise in multi-bit DAC structures is being studied. We will be dealing with thermometer code driven unit component-based DACs for modelling the DAC error. A model consisting of error of the DAC is being presented and based on that model, the impact of the error on the general execution of multi-bit Delta Sigma ADC is investigated.

A number of procedures which have already been used in literature for dealing with the DAC error are discussed [11]-[21]. They include 1.) Shaping of DAC noise; 2.) Self-adjusting DAC; 3.) Measurement of DAC noise and digital correction. The techniques are studied in detail and their downside are being pointed out.

3.1 Multi-bit unit component based DAC Model

Although there are numerous ways of building a multi-bit DAC structure, we are constructing the DAC from unit components and is frequently utilized. We will be dealing with unit elements based DAC structure for all the discussions from here on.

We are building a thermometer unit element based DAC structure which consists of M current sources commonly known as unit components. A DAC with (M+1) different levels will comprise of M unit components which can be current sources or switched-capacitors. The unit components will generate same nominal value Δ when switched on. The quantity of input bits to the DAC will be equivalent to the quantity of unit components used. Each unit component of DAC is triggered by one of the bits, $t_i(k)$ (i=1, 2, .M). The $t_i(k)$= 1 or -1 will select (turn on) the i th unit element at time k with a different current path.

Ideal DAC output is the aggregate of the outputs of all unit elements (shown in Fig. 3-1) and given by Equation (3.1):

$$d_{ideal}(k) = \Delta . \sum_{i=1}^{M} t_i(k) \tag{3.1}$$

$t_i(k)$ (i=1, 2, .M) are termed as the **selecting signals**.

Δ equals the quantization step.

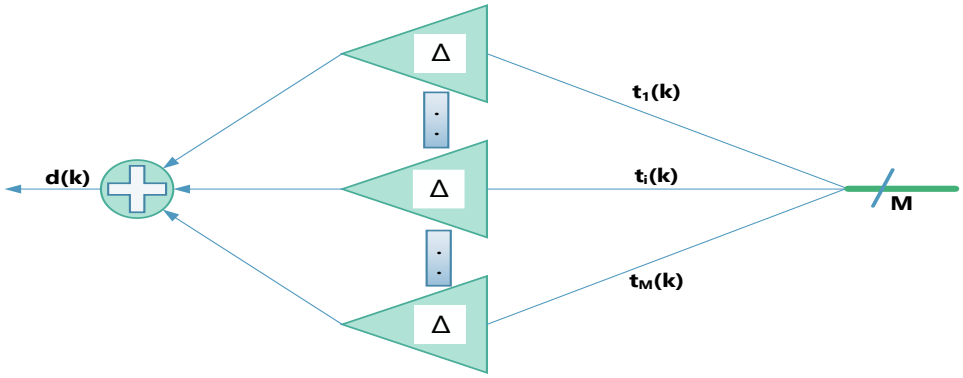

Fig. 3-1 Ideal unit component based DAC structure

Unit components can be actualized utilizing capacitors in switched capacitor DAC or using current sources in a current-steering DAC at circuit designing level.

3.1.1 DAC Noise

In practice, the unit elements used in the DAC are not ideal in nature resulting in deviations from their ideal value. The mean value of the unit elements deviates from Δ to $\beta.\Delta$. Assume e_i (i=1, 2 …M) are the estimations of standardized variations of the individual unit elements outputs from their average worth (referred to as unit component errors). Thus, the output of each unit element can be represented as $\beta.\Delta.(1 + e_i)$. A model is shown in the Fig. 3-2 below to depict these two effects.

In this research, only static variations have been considered and slow changes as compared to the rate of clock are considered. So, the unit element errors and β are assumed to be steady.

Also, the unit element errors e_i are thought to be WHITE GAUSSIAN NOISE with zero mean value and are thought to be relatively constant for a significant duration of time. Hence, it is obvious to note that:

$$\sum_{i=1}^{M} e_i = 0$$

The Non-Ideal DAC output thus becomes Equation (3.2):

$$d_{actual}(k) = \beta.\Delta.\sum_{i=1}^{M} t_i(k) + \beta.\Delta.\sum_{i=1}^{M} t_i(k).e_i \qquad (3.2)$$

$t_i(k)$ (i=1, 2, .M) are alluded to as the **selecting signals**.

Δ equals the quantization step.

β is the extra gain added to the DAC.

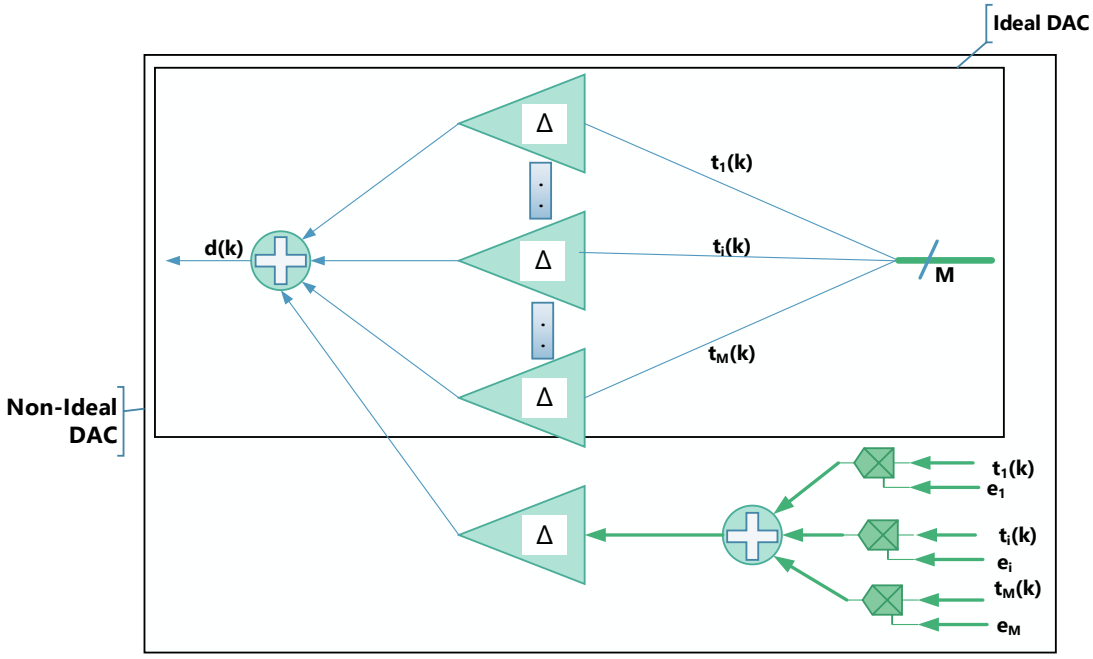

Fig. 3-2 Non-Ideal unit element based DAC structure

On comparing the Equation (3.2) with the ideal DAC output Equation (3.1), the second term is the error introduced in the DAC and is referred as **DAC Noise**.

DAC Error is basically selecting signals i.e. $t_i(k)$ multiplied by corresponding e_i's over M signals also given by Equation (3.3) below:

$$\textbf{DAC Error} = \boldsymbol{\beta}.\boldsymbol{\Delta}.\sum_{i=1}^{M} \boldsymbol{t_i}(\boldsymbol{k}).\boldsymbol{e_i} \qquad (\mathbf{3.3})$$

3.1.2 Impact of DAC Error on DSM Output

The output of multi-bit Delta Sigma ADC is a coded word, consisting of M+1 level, ranging from 0 to M (in our case M=15 for a 4-bit Quantizer). The output can be converted to binary format i.e. bits having different weights but we are implementing a thermometer code based unit element DAC in which all the unit elements have same weight. In order to get same weights, we need to convert the output to thermometer code. Therefore, a binary to thermometer decoder is embedded between the quantizer and DAC in Delta Sigma ADC appeared in Fig. 3-3.

A BTCC accepts input in coded form consisting of M+1 levels and outputs a vector of length M containing as many ones as the value of input code word and remaining -1. For example, suppose an input code word = 7, the output of BTCC will be 1111111-1-1-1-1-1-1-1-1 which is equal to 7 ones and remaining -1.

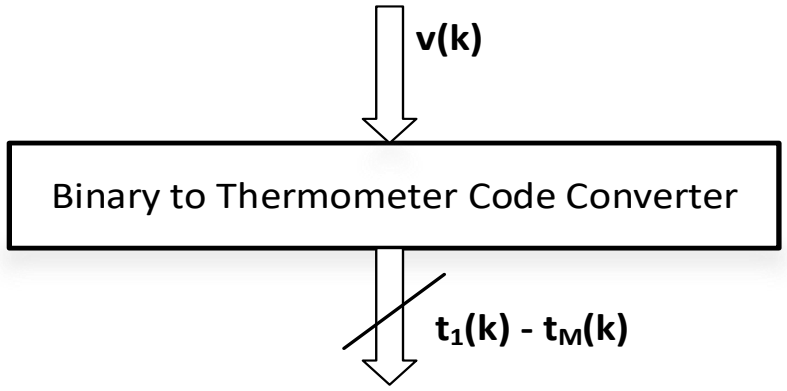

Fig. 3-3 Binary to Thermometer code converter

$t_i(k)$ (i =1, 2, .M) are the output of Binary to thermometer code converter and are alluded to as the selection signals.

The thermometer coded signal (which is a surge of 1's and -1's) is used to drive the unit element DAC, consisting of M unit components, and each unit components of the DAC is constrained by one bit of selecting signal.

Note: The main downside of thermometer coded output is that there is a fixed need on the choice of unit elements. The selecting signals i.e. $\boldsymbol{t_i(k)}$ are strongly correlated to the DAC input and as we know $\boldsymbol{t_i(k)}$ contain a scaled version of ADC output signal ($v(k)$) and $v(k)$ contains a component of input signal thereby resulting in input signal-dependent tones in ADC output resulting in degradation of ADC linearity.

K / $t_i(k)$	1	2	3	4	5
Input value	6	3	5	1	2
$t_1(k)$	1	1	1	1	1
$t_2(k)$	1	1	1	-1	1
$t_3(k)$	1	1	1	-1	-1
$t_4(k)$	1	-1	1	-1	-1
$t_5(k)$	1	-1	1	-1	-1
$t_6(k)$	1	-1	-1	-1	-1
$t_7(k)$	-1	-1	-1	-1	-1

Table-I 7-bit Thermometer code for 3-bit Quantizer output

To solve this, we need to decorrelate the selecting signals, $t_i(k)$ with the DAC input. To achieve this, we will perform some operations which will be covered in next topic:

The DAC Error shows up in the final output equation of the Delta Sigma ADC after getting filtered through the STF (z) of the ADC with a phase shift of 180^o. Thus, we can define Error Transfer Function (ETF (z)) as given by Equation (3.4) below:

$$\text{ETF (z)} = - \text{STF (z)} \tag{3.4}$$

Where $etf(k)$= impulse response of Error Transfer Function ETF (z) of the Sigma Delta ADC. Equation (3.5) shows the DAC Error at the output of Modulator having non-ideal unit components.

$$\boldsymbol{DAC\ Error\ in\ ADC\ output} \;=\; \boldsymbol{etf(k).\beta.\sum_{i=1}^{M} e_i.t_i(k)} \qquad (\mathbf{3.5})$$

Note that the $\boldsymbol{\Delta}$ in Equation (3.3) is cancelled by the $\boldsymbol{1/\Delta}$ of quantizer before appearing at the ADC output. Thus, the total output of the ADC $v(k)$ becomes Equation (3.6):

$$v(k) \;=\; stf(k).r(k) + ntf(k).q(k) + etf(k).\sum_{i=1}^{M} e_i.t_i(k) \qquad (3.6)$$

Where $stf(k)$ and $ntf(k)$ are the impulse responses of Signal Transfer Function and Noise Transfer Function of the Sigma Delta ADC respectively.

Since DAC noise is not shaped out by the delta sigma modulator as already demonstrated in Fig. 2-20, it directly affects the modulator output. Upon studying the DAC error equation, we can say that DAC noise depends on two terms:

1) The amplitude of the unit component noise.
2) The spectral properties of the selection signals.

As we can see in the equation of DAC Error above, $\boldsymbol{\beta}$ is the extra gain added to the DAC. Due to the extra gain added to the DAC, the STF and NTF of the system also changes which changes the location of poles in the transfer function of STF and NTF. This changes the STF and NTF of the system but since $\boldsymbol{\beta \ll 1,}$ it hardly impacts the performance of the system.

3.2 DAC Noise Shaping

As we have already discussed in the previous sections that DAC noise is not removed by the Delta-Sigma ADC. Also, we have seen the thermometer code unit component based DAC model in the last section. Table I above shows a 7-bit thermometer code input to the DAC (DAC assumed to have 7-unit elements) where k denotes the input sample time and $\boldsymbol{t_i(k)}$ where i =1, 2... 7 are the selection signals. The block diagram corresponding to the thermometer is additionally appeared in Fig. 3-3.

The main drawback of thermometer coded output is that there is a fixed need on the choice of unit elements. The selection signals i.e. $\boldsymbol{t_i(k)}$ are strongly correlated to the DAC input and as we know $\boldsymbol{t_i(k)}$ contain a scaled version of ADC output signal ($v(k)$) and $v(k)$ contains a component of input signal (as shown by Equation (1.2)) thereby resulting in input signal-dependent tones in ADC output resulting in degradation of ADC linearity. Also, the unit element controlled by $\boldsymbol{t_1(k)}$ has the highest priority and the one constrained by $\boldsymbol{t_8(k)}$ will have lowest priority leading to large power at low frequencies thereby degrading the resolution of ADC.

To solve this, we need to de-associate the selecting signals, $t_i(k)$ with the DAC input and change the priority of selection of unit elements.

To achieve this, we have different techniques by which we can reduce the DAC error commonly known as Dynamic Element Matching:

1) Zero-order Randomization (Scrambling)

   Scrambling is a digital rearranging process which revamps the thermometer code bits without affecting its absolute incentive as shown in Table II. By doing so, selection of unit elements in the DAC becomes dynamic and this helps in dispensing the harmonic tones in ADC output. Also, as we can see there is no noise shaping, the resolving power of ADC will still be very less in the signal band.

2) Data Weighted Averaging (DWA)

   This is a very basic and effective technique for handling the DAC Error. This is also a rearranging algorithm which rearranges the bits of the thermometer code in circular fashion i.e. it makes the unit element selection priority rotated which has an effect of first-order shaping of the selecting signals. The DAC error is moved out of band of interest to high frequencies. But this technique remains ineffectual in expelling the harmonics in the ADC output.

There are many different renditions of Dynamic Element Matching (DEM) available. Some of them help in achieving higher-order noise shaping while others help in randomization plus noise shaping (which reduces the noise shaping). Randomization help in removing signal dependent tones from the output giving good ADC linearity. Noise shaping help in achieving good resolution but that is conditional since noise shaping depends on the unit element error amplitude, OSR value and the order of Noise shaping involved.

k / $t_i(k)$	1	2	3	4	5
Input value	6	3	5	1	2
$t_1(k)$	-1	-1	-1	-1	-1
$t_2(k)$	1	-1	1	-1	-1
$t_3(k)$	1	1	1	-1	-1
$t_4(k)$	1	-1	1	1	1
$t_5(k)$	1	-1	1	-1	-1
$t_6(k)$	1	1	-1	-1	-1
$t_7(k)$	1	1	1	-1	1

Table II Scrambled 7-bit thermometer code

The problem arises in setting of broadband applications. As we have already discussed that in broadband applications high value of OSRs are not practical and Noise shaping has no big significance with low estimation of OSRs. Thus, for high resolution necessities we cannot on Noise shaping techniques for the removal of DAC error with low value of OSR. Also, higher order shaping is hard to structure and execute.

3.3 Earlier Correction Techniques

We have seen the various methods of DAC noise shaping in the previous topic to lessen the effect of noise at the output. But the restrictions of the techniques discussed guide us to search for some other way to reduce the errors by other means. In this section, we will discuss the methods which will reduce the abundancy of DAC noise

with some circuit level plans. The methods used for error correction include self-adjusting DAC and measurement of DAC deviations through different techniques discussed in this section.

The DAC proposed in [30] shows a self-adjusting DAC in which the yield of each unit component's output is contrasted with that of a perfect component taken as a reference, and dependent on the correlation results the unit components are changed in accordance with make up for the deviation. One additional unit component is used for reinforcement for the adjusted component, as appeared in Fig. 3-4. This alignment procedure keeps running out of sight for the duration of when the framework is on. Reproductions demonstrate that it is very successful on improving the exactness of the unit components. Be that as it may, additional simple hardware must be included every unit component to make it customizable and two additional unit components are required.

A system that legitimately measures the alteration of each DAC dimension's output and performs computerized revision at the output as indicated by the estimated results was proposed, and an effective usage was exhibited in [31]. Utilizing this system, the delta sigma ADC is redesigned into one-piece at the time of alignment procedure and the yield deviations of the DAC are estimated by the single-piece delta sigma ADC and put away in a Smash, as appeared in Fig. 3-5. The principle disadvantage of the strategy is that we need to interfere or stop the typical activity of the system in order to carry the adjustment algorithm since it gets executed in the frontal area.

In [32], the advanced amendment system of [31] is changed. We utilize a new delta sigma ADC for the measurement of noise which works for the estimation of less used unit-components at each clock by utilizing the additional information and yield ports, as appeared in Fig. 3-6. The noise values are additionally put away in a memory for the advanced amendment. Since the noise estimation is in the foundation, the activity of the system is going on consistently and is not interrupted. This strategy requires a double ended DAC to be confounded in plan and an additional delta sigma ADC.

In [21], another foundation computerized adjustment method is accounted for. Here, rather than direct estimation, the unit component mistakes are distinguished from the ADC yield. It requires changing of the noise transfer function of the ADC and

creation of a zero at fs/2. This leads to a gap in the frequency range since due to the introduction of zero, there will be no information signal and leads to the molding of error.

In this method, every unit component is constrained by a known value of data pattern one by one which produces an adjustment tone at fs/2. This tone contains the noise data of the component being controlled. The estimation of this particular blunder is then precisely recuperated from the alignment tone after the information signal and the quantization clamor are sifted through. After the mistakes of all unit components are gotten, the remedy is continued, as appeared in Fig. 3-7. The disadvantage of this method is self-evident: the NTF should be changed by moving one of zeros to fs/2, bringing about brought up in band quantization commotion control. In addition, an additional unit-component is required for reinforcement in the DAC.

CHAPTER 4

PROPOSED TECHNIQUE

In this research, we will find a new method for the removal of DAC Error from the Delta Sigma ADC output. For this, we will first try to separate the error term from the main output and then find a way to remove it from the overall output. Thus, this technique involves two main steps which are:

1) Error Measurement
2) Error Removal using Digital DAC.

The main advantages of this technique over the other methods discussed before are:

1) This method can take benefit of scaling of ICs and help in achieving lower power dissipation and lower area.
2) It can be easily applied to different technology nodes.
3) This method uses digital technique to increase the circuit performance.
4) This technique does not stop the typical activity of the system and keeps executing in the background area.

As already discussed, the output of the Delta- Sigma ADC consists of three terms:

1) Filtered input signal, $r'(k)$ which is equal to $r(k) * stf(k)$ where $stf(k)$ = impulse response of STF (z).

2) Filtered Quantization noise, $q'(k)$ which is equal to $q(k) * ntf(k)$ where $ntf(k)$ = impulse response of NTF (z).

3) Filtered DAC error, $\sum_{i=1}^{M} e_i * t_i'(k)$ where $t_i'(k)$is equal to $t_i(k) * etf(k)$ where $etf(k)$ = impulse response of ETF (z).

The output of the ADC , $v(k)$ can be demonstrated using Equation (4.1) as:

$$v(k) = r'(k) + q'(k) + \sum_{i=1}^{M} e_i * t_i'(k) \tag{4.1}$$

If by some means we can get a measure of $\mathbf{e_i}$ i.e. $\widehat{\mathbf{e}_\mathbf{i}}$ (Measured error), then we can use these calculated errors for correcting the ADC output as $v_c(k)$ which will be equivalently depicted with Equation (4.2):

$$v_c(k) = \sum_{i=1}^{M} t_i(k) + \sum_{i=1}^{M} t_i(k) \,.\, \widehat{e_i} \tag{4.2}$$

4.1 Error Measurement

In this section, we will try to measure the error coefficients of the unit elements used in the DAC. The block diagram used for the measurement of errors is depicted using Fig. 4-1:

Fig. 4-1 presents a Delta Sigma Modulator consisting of a 4-bit Quantizer, Binary to Thermometer Converter, Scrambler and a 4-bit DAC embedded in the negative feedback loop of the system. Apart from the modulator, FIR Filters, Down-samplers are being used in the design.

To measure the errors correctly, we will employ a **CORRELATION TECHNIQUE** according to which'' ***If an input signal modulated by a constant DC value is correlated with the same input signal then the correlation will result in the estimation of those constant DC values.''***

We will employ this technique between $\boldsymbol{v(k)}$ i.e. the output of the modulator and $\boldsymbol{t_i'(k)}$ i.e. the filtered thermometer code corresponding to the ADC output**.**

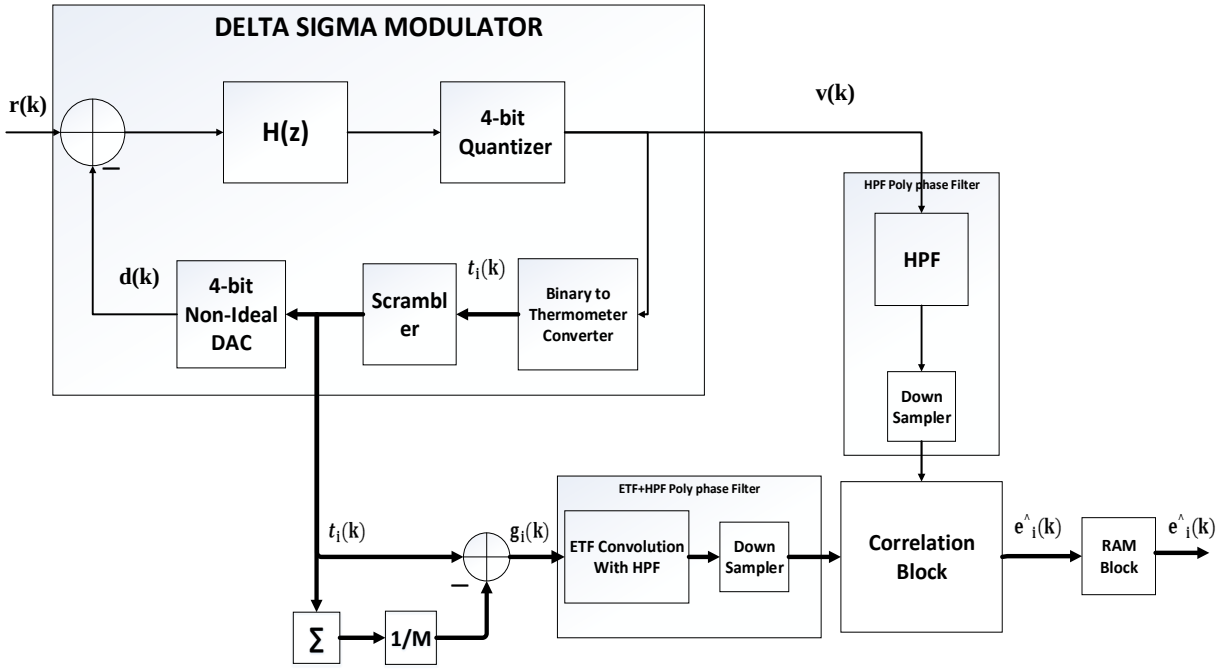

Fig. 4-1 Block Diagram of Error Measurement Technique

We have already seen that according to Equation (4.1), $v(k)$ contains a term in which $t_i^{'}(k)$ is multiplied by e_i i.e. $\sum_{i=1}^{M} t_i^{'}(k).e_i$ .When this term is correlated with $t_i^{'}(k)$, it will result in the estimation of error terms denoted as $\hat{\mathbf{e}}_\mathbf{i}$**.** Throughout the process of error measurement, we have assumed that the error introduced in the DAC are static errors and they remain relatively constant for a significant duration of time. Thus, in this case our input signal is $t_i^{'}(k)$ and e_i terms are constant DC values.

Mathematically, we can represent the correlation process as shown in Equation (4.3) below:

$$COR\left[v(k), t_i^{'}(k)\right] = COR\left[\left[r'(k) + q'(k) + \sum_{j=1}^{M} t_j^{'}(k).e_j\right], t_i^{'}(k)\right] \tag{4.3}$$

$$= COR\left[r'(k), t_i^{'}(k)\right] + COR\left[q'(k), t_i^{'}(k)\right] + COR\left\{\left[\sum_{j=1}^{M} t_j^{'}(k).e_j\right], t_i^{'}(k)\right\}$$

The first and second terms in the correlation output corresponds to correlation of input signal and quantization noise with selecting signals which must be removed to estimate the errors correctly.

Note: The correlation terms $\boldsymbol{COR}\left[\boldsymbol{r'(k), t_i'(k)}\right]$and $\boldsymbol{COR}\left[\ \boldsymbol{q'(k), t_i'(k)}\right]$ are not zero.

Reason: As we know that the ADC output $v(k)$ is represented in the form of selecting signal $t_i(k)$.

Therefore, we can infer that the selecting signals, $t_i(k)$ inherently contains a scaled version of $v(k)$, which includes the input signal and the error terms. As a result, the selecting signals are correlated with the input signal and noise signal. So, when the $v(k)$ and $t_i(k)$ are correlated, the correlation output contains extra terms which cause interference in the DAC error estimation.

We will try to remove these extra terms as far as possible. For this to happen, we will pass the ADC output through a FIR High Pass Filter which will remove most part of the input signal. The detailed analysis on this will be covered in later sections. For mathematics, assume that the correlation between information signal and $\boldsymbol{t_i'(k)}$ is null. The correlation between quantization noise and $\boldsymbol{t_i'(k)}$ will result in a very small value since most of the quantization noise is shaped out of signal band. Hence, these terms are neglected i.e.

$$COR\left[r'(k), t_i'(k)\right] \textit{ is assumed to be zero}$$

$$COR\left[\ q'(k), t_i'(k)\right] = \textit{small value}$$

Then the correlation output becomes Equation (4.4) and after expanding the Equation (4.4), we get Equation (4.5)

$$\boldsymbol{COR}\left[\ \boldsymbol{v'(k), t_i'(k)}\right] = \boldsymbol{COR}\left\{\left[\sum_{\boldsymbol{j=1}}^{\boldsymbol{M}} \boldsymbol{e_j}.\boldsymbol{t_j'(k)}\right], \boldsymbol{t_i'(k)}\right\} \qquad (\mathbf{4.4})$$

$$\boldsymbol{COR}\left[\ \boldsymbol{v'(k), t_i'(k)}\right] = \boldsymbol{e_i} + \sum_{\boldsymbol{j=1, j\neq i}}^{\boldsymbol{M}} \boldsymbol{e_j(k)}.\left[\boldsymbol{t_j'(k)} * \boldsymbol{t_i'(k)}\right] \qquad (\mathbf{4.5})$$

Where * denote the correlation operator and $v'(k)$ is the ADC output from which input signal is suppressed.

Thus, we can conclude that the correlation method stated above will results in the measurement of errors $\hat{\mathbf{e}}_\mathbf{i}$ **provided we know the correlation between the bits of selecting signal i.e.** $\boldsymbol{t_i'(k)}$. To calculate the correlation value between the various bits of the selecting signal, we need to make sure whether the bits are correlated with each other or not.

Note: The $\boldsymbol{t_i(k)}$ **are not correlated with each other.**

Reason: As we know that $t_i(k)$ consists of a stream of 1's and -1's only. It is significant to say that

$$\sum_{i=1}^{M} t_i(k) \neq 0$$

Thus $t_i(k)$ are decorrelated with each other. Therefore**, the correlation between the** $\boldsymbol{t_i(k)}$ **will be zero.** Hence, the errors could only be estimated if by some means the selecting signals are correlated with each other since only then we would be able to calculate the correlation between various bits of selecting signal.

4.1.1 Generation of a new selecting signal - $\boldsymbol{g_i(k)}$

In order to correlate the selecting signals with each other, we will generate a new term $g_i(k)$ which will be obtained by removing a part of $\sum_{i=1}^{M} t_i(k)$ from each bit of the selecting signal $t_i(k)$ shown with the help of Equation (4.6)

$$g_i(k) = t_i(k) - \frac{1}{M}\sum_{i=1}^{M} t_i(k) \qquad (4.6)$$

Note: The $\boldsymbol{g_i(k)}$ **are correlated to each other**

Reason: We will try to find $\sum_{i=1}^{M} g_i(k)$as shown below:

$$\sum_{i=1}^{M} g_i(k) = \sum_{i=1}^{M} t_i(k) - \sum_{i=1}^{M} \frac{B}{M}$$

Let $\sum_{i=1}^{M} t_i(k) = B$ *For calculations.*

$$\sum_{i=1}^{M} g_i(k) = \sum_{i=1}^{M} t_i(k) - M.\frac{B}{M}$$

$$\sum_{i=1}^{M} g_i(k) = B - B = 0$$

By doing so, the sum of $n_i(k)$ will result to zero i.e.

$$\sum_{\boldsymbol{i=1}}^{\boldsymbol{M}} \boldsymbol{g_i(k)} = \mathbf{0}$$

Which means that $g(k)$ are correlated with each other. Also, $g_i(k)$ are passed through the ETF (z) to generate $g_i^{'}(k)$.

$$g_i^{'}(k) = g_i(k).etf(k)$$

Applying summation on both sides, we get:

$$\sum_{i=1}^{M} g_i^{'}(k) = \sum_{i=1}^{M} g_i(k).etf(k)$$

Hence,

$$\sum_{i=1}^{M} g_i^{'}(k) = 0, \qquad \text{since } \sum_{i=1}^{M} g_i(k) = 0$$

4.1.2 Impact of Generated Signal on Modulator output

Now we will see whether the generated term $g_i(k)$ has any impact on the Actual DAC output and corresponding ADC output. On substituting the value of $t_i(k)$ from the Equation (4.6) in the Non- Ideal DAC output Equation (3.2), we get Equation (4.7):

$$d_{actual}(k) = \Delta.\sum_{i=1}^{M} t_i(k) + \Delta.\sum_{i=1}^{M} t_i(k).e_i$$

$$d_{modified}(k) = \Delta.\sum_{i=1}^{M} t_i(k) + \Delta\sum_{i=1}^{M}\left(g_i(k) + \frac{1}{M}\sum_{i=1}^{M} t_i(k)\right)e_i$$

$$d_{modified}(k) = \Delta \sum_{i=1}^{M} t_i(k) + \Delta . \sum_{i=1}^{M} g_i(k) . e_i + \Delta . \frac{v(k)}{M} \sum_{i=1}^{M} e_i$$

Since $\sum_{i=1}^{M} e_i = 0$, (because the errors are **assumed** to be white in nature with zero mean value. Therefore,

$$d_{modified}(k) = \Delta . \sum_{i=1}^{M} t_i(k) + \Delta . \sum_{i=1}^{M} g_i(k)\, e_i \tag{4.7}$$

Therefore, on comparing Equations (2.2) and (4.7), we can see that DAC Error can be represented as Equation (4.8) in ADC output:

$$\text{DAC Error } = \Delta . \sum_{i=1}^{M} g_i(k) \, . e_i \tag{4.8}$$

With the help of modified DAC output $d_{modified}(k)$ in Equation (4.7), the ADC output turns out to be Equation (4.9):

$$v(k) = r'(k) + q'(k) + \sum_{i=1}^{M} g_i^{'}(k) . e_i \tag{4.9}$$

4.1.3 Correlation Calculations

The Correlation between two signals a (k) and b (k) utilized in the process is defined as in Equation (4.10) below:

$$COR[a(k), b(k)] = \frac{\sum_{k=1}^{k} a(k) . b(k)}{\sum_{k=1}^{k} abs(a(k))} \tag{4.10}$$

Where a (k) and b (k) are the signals engaged in the correlation process.

As we have already assumed that the selecting signals are de-correlated with the input signal and noise signal, the correlation between $v(k)$ and $g_i'(k)$ can be computed to get an estimate measure of errors as shown below:

$$\boldsymbol{COR}[\boldsymbol{v}(\boldsymbol{k}), \boldsymbol{g_i'}(\boldsymbol{k})] = \boldsymbol{COR}[\boldsymbol{r'}(\boldsymbol{k}), \boldsymbol{g_i'}(\boldsymbol{k})] + \boldsymbol{COR}[\boldsymbol{q'}(\boldsymbol{k}), \boldsymbol{g_i'}(\boldsymbol{k})] + \boldsymbol{COR}\left\{\left[\sum_{j=1}^{M} \boldsymbol{e_j}.\boldsymbol{g_j'}(\boldsymbol{k})\right], \boldsymbol{g_i'}(\boldsymbol{k})\right\}$$

The first two terms of the correlation equation evaluate to zero as per the assumptions. Therefore, the correlation output will be equal to Equation (4.11):

$$\boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_i'}(\boldsymbol{k})] = \boldsymbol{e_i} + \left\{\sum_{j=1, j\neq i}^{M} \boldsymbol{e_j}(\boldsymbol{k}).\boldsymbol{COR}[\boldsymbol{g_j'}(\boldsymbol{k}), \boldsymbol{g_i'}(\boldsymbol{k})]\right\} \qquad (\mathbf{4.11})$$

Where $v'(k)$ is the ADC output that does not contain input signal i.e. High Pass Filtered ADC output. The term $g_i'(k)$ is a matrix of size M × 1 and the modulator output is of size 1 × 1. Thus, correlation between them results in a matrix of size M × 1 as shown below:

$$\begin{bmatrix} \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_1'}(\boldsymbol{k})] \\ \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_2'}(\boldsymbol{k})] \\ \vdots \\ \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_M'}(\boldsymbol{k})] \end{bmatrix} = \begin{bmatrix} 1 & \boldsymbol{COR}[\boldsymbol{g_2'}(\boldsymbol{k}), \boldsymbol{g_1'}(\boldsymbol{k})] & \cdots & \boldsymbol{COR}[\boldsymbol{g_M'}(\boldsymbol{k}), \boldsymbol{g_1'}(\boldsymbol{k})] \\ \boldsymbol{COR}[\boldsymbol{g_1'}(\boldsymbol{k}), \boldsymbol{g_2'}(\boldsymbol{k})] & 1 & \cdots & \boldsymbol{COR}[\boldsymbol{g_M'}(\boldsymbol{k}), \boldsymbol{g_2'}(\boldsymbol{k})] \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{COR}[\boldsymbol{g_1'}(\boldsymbol{k}), \boldsymbol{g_M'}(\boldsymbol{k})] & \boldsymbol{COR}[\boldsymbol{g_2'}(\boldsymbol{k}), \boldsymbol{g_M'}(\boldsymbol{k})] & \cdots & 1 \end{bmatrix} . \begin{bmatrix} \boldsymbol{e_1} \\ \boldsymbol{e_2} \\ \vdots \\ \boldsymbol{e_M} \end{bmatrix}$$

$$\begin{bmatrix} \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_1'}(\boldsymbol{k})] \\ \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_2'}(\boldsymbol{k})] \\ \vdots \\ \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_M'}(\boldsymbol{k})] \end{bmatrix} = \boldsymbol{T} . \begin{bmatrix} \boldsymbol{e_1} \\ \boldsymbol{e_2} \\ \vdots \\ \boldsymbol{e_M} \end{bmatrix}$$

Where

$$\mathrm{T} = \begin{bmatrix} 1 & COR[g_2'(k), g_1'(k)] & \cdots & COR[g_M'(k), g_1'(k)] \\ COR[g_1'(k), g_2'(k)] & 1 & \cdots & COR[g_M'(k), g_2'(k)] \\ \vdots & \vdots & \ddots & \vdots \\ COR[g_1'(k), g_M'(k)] & COR[g_2'(k), g_M'(k)] & \cdots & 1 \end{bmatrix}$$

is a M × M Matrix.

Thus, using matrix properties, we can find that error coefficients will be equal to Equation (4.12):

$$\begin{bmatrix} \widehat{e_1} \\ \widehat{e_2} \\ \vdots \\ \widehat{e_M} \end{bmatrix} = T^{-1} \cdot \begin{bmatrix} COR[v'(k), g_1'(k)] \\ COR[v'(k), g_2'(k)] \\ \vdots \\ COR[v'(k), g_M'(k)] \end{bmatrix} \quad (4.12)$$

To get an estimate measure of error coefficients, we need to find the following terms:

1) The value of matrix T that had been introduced above. To be able to get the values of the matrix, we need to find the inverse of matrix T.
2) The correlation between the bits of the new selecting signal.

Inverse of T Matrix:

Now, as each correlation terms of the matrix T can be easily computed. We will check whether the inverse of the matrix exists or not. By rules of mathematics, if the summation of row or a column turns out to be zero, then inverse of the matrix does not exists. Let us compute the summation of a row/column of matrix **T** which will be equal to:

$$= COR[g_1'(k), g_1'(k)] + COR[g_2'(k), g_1'(k)] + COR[g_3'(k), g_1'(k)] + \ldots + COR[g_M'(k), g_1'(k)]$$

$$= COR\left[\sum_{i=1}^{M} g_i'(k), g_1'(k)\right]$$

As we know that $g_i'(k)$ are correlated to each other i.e. the summation of $g_i'(k)$ is zero.

$$i.e. \sum_{i=1}^{M} g_i'(k) = 0$$

Therefore, summation of each row and column of **T** matrix will also be equal to zero i.e.

$$CORR\left[\sum_{i=1}^{M} g_i'(k), g_1'(k)\right] = 0$$

Thus, we can say that T matrix is a **singular matrix** and by the rules of mathematics, for a singular matrix determinant is not defined and therefore, **T inverse will not exist**. To

compute $\boldsymbol{T^{-1}}$**,** we remove one row and one column from the **T** matrix resulting into (M-1)×(M-1) Matrix.

The modified **T** matrix is notated as $\boldsymbol{T'}$ given in Equation (4.13)

$$\boldsymbol{T'} = \begin{bmatrix} 1 & COR\left[g_2'(k), g_1'(k)\right] & \cdots & COR\left[g_{M-1}'(k), g_1'(k)\right] \\ COR\left[g_1'(k), g_2'(k)\right] & 1 & \cdots & COR\left[g_{M-1}'(k), g_2'(k)\right] \\ \vdots & \vdots & \ddots & \vdots \\ COR\left[g_1'(k), g_{M-1}'(k)\right] & COR\left[g_2'(k), g_{M-1}'(k)\right] & \cdots & 1 \end{bmatrix} \quad (4.13)$$

Now error coefficients can be recovered as in Equation (4.14):

$$\begin{bmatrix} \widehat{\boldsymbol{e_1}} \\ \widehat{\boldsymbol{e_2}} \\ \vdots \\ \widehat{\boldsymbol{e_{M-1}}} \end{bmatrix} = (\boldsymbol{T'})^{-1} \cdot \begin{bmatrix} \boldsymbol{COR}\left[\boldsymbol{v'(k)}, \boldsymbol{g_1'(k)}\right] \\ \boldsymbol{COR}\left[\boldsymbol{v'(k)}, \boldsymbol{g_2'(k)}\right] \\ \vdots \\ \boldsymbol{COR}\left[\boldsymbol{v'(k)}, \boldsymbol{g_{M-1}'(k)}\right] \end{bmatrix} \quad (4.14)$$

Further, to compute the value of $\boldsymbol{T'}$matrix, we need to determine the correlation between the bits of selecting signal, $g_i'(k)$. Since **T** matrix consists of correlation of $\boldsymbol{g_j'(k)}$and $\boldsymbol{g_i'(k)}$ which are correlated to each other hence they can be computed easily.

Calculation of Correlation term: $\boldsymbol{COR\left[g_j'(k), g_i'(k)\right]}$

As we know,

$\sum_{i=1}^{M} \boldsymbol{g_i'(k)} = \boldsymbol{0}$ derived earlier.

$\boldsymbol{g_i'(k)} = -\sum_{j=1, j\neq i}^{M} \boldsymbol{g_j'(k)}$ holds true for all i, j.

Since the $\boldsymbol{g_i'(k)}$are correlated with each other, then

$\boldsymbol{g_j'(k)} = \boldsymbol{\beta}.\boldsymbol{g_i'(k)} + \boldsymbol{u_j(k)}$ i.e. $g_i(k)$ is correlated with $g_j(k)$and the correlation factor is β between them. $\boldsymbol{u_j(k)}$ = **uncorrelated component of** $\boldsymbol{g_j'(k)}$ **with** $\boldsymbol{g_i'(k)}$**.**

Let us find the value of $COR\left[g_j^{'}(k), g_i^{'}(k)\right]$:

Consider M = 4 (only for derivation)

$\sum_{i=1}^{M} g_i^{'}(k) = 0$,

Then, $g_1^{'}(k) + g_2^{'}(k) + g_3^{'}(k) + g_4^{'}(k) = 0$

⇨ $g_1^{'}(k)$ = - $(g_2^{'}(k) + g_3^{'}(k) + g_4^{'}(k))$

⇨ $g_1^{'}(k)$ = - $(g_2^{'}(k) + (g_2^{'}(k) + u_3(k)) + g_2^{'}(k) + u_4(k))$ where $g_3^{'}(k) = g_2^{'}(k) + u_3(k)$as per equation and so on.

⇨ $g_1^{'}(k)$ = - ($3.g_2^{'}(k) + u_3(k) + u_4(k)$) where $(u_3(k) + u_4(k))$ is the uncorrelated part.

⇨ $g_2^{'}(k) = \frac{-1}{3}.g_1^{'}(k) + (u_3(k) + u_4(k))$

Generalizing for M, we get

⇨ $g_j^{'}(k) = \frac{-1}{M-1} * g_i^{'}(k) + uncorrelated\ part$

⇨ Therefore, we can conclude for $\boldsymbol{\beta} = \frac{-1}{M-1}$i.e. the correlation between $g_j^{'}(k)$and $g_i^{'}(k)$ is$\frac{-1}{M-1}$.

⇨ Hence,

$$COR\left[g_j^{'}(k), g_i^{'}(k)\right] = \frac{-1}{M-1} \qquad (4.15)$$

Therefore, we can now represent the $\boldsymbol{T'}$matrix as follows:

$$\boldsymbol{T'} = \begin{bmatrix} 1 & COR[g_2^{'}(k), g_1^{'}(k)] & \cdots & COR[g_{M-1}^{'}(k), g_1^{'}(k)] \\ COR[g_1^{'}(k), g_2^{'}(k)] & 1 & \cdots & COR[g_{M-1}^{'}(k), g_2^{'}(k)] \\ \vdots & \vdots & \ddots & \vdots \\ COR[g_1^{'}(k), g_{M-1}^{'}(k)] & COR[g_2^{'}(k), g_{M-1}^{'}(k)] & \cdots & 1 \end{bmatrix}$$

Replacing all the correlation terms by $\frac{-1}{M-1}, we\ get$ Equation (4.16)

$$\boldsymbol{T'} = \begin{bmatrix} 1 & \frac{-1}{M-1} & \cdots & \frac{-1}{M-1} \\ \frac{-1}{M-1} & 1 & \cdots & \frac{-1}{M-1} \\ \vdots & \vdots & \ddots & \vdots \\ \frac{-1}{M-1} & \frac{-1}{M-1} & \cdots & 1 \end{bmatrix} \tag{4.16}$$

Where $\boldsymbol{T'}$ is a (M-1) × (M-1) matrix

Inverse of $\boldsymbol{T'} = (\boldsymbol{T'})^{-1}$ is given by Equation (4.17) which is calculated by matrix manipulations:

$$(\boldsymbol{T'})^{-1} = \begin{bmatrix} \frac{M-1}{M} & 0 & \cdots & 0 \\ 0 & \frac{M-1}{M} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{M-1}{M} \end{bmatrix} + \frac{M-1}{M} \cdot \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix} \tag{4.17}$$

The second term in the $(\boldsymbol{T'})^{-1}$ can be neglected as a DC offset. Therefore, $(\boldsymbol{T'})^{-1}$ can be described as Equation (4.18):

$$(\boldsymbol{T'})^{-1} = \begin{bmatrix} \frac{M-1}{M} & 0 & \cdots & 0 \\ 0 & \frac{M-1}{M} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{M-1}{M} \end{bmatrix} \tag{4.18}$$

The estimated errors $\widehat{\boldsymbol{e_i}}$ can be calculated as below and given by Equation (4.19):

$$\begin{bmatrix} \widehat{\boldsymbol{e_1}} \\ \widehat{\boldsymbol{e_2}} \\ \vdots \\ \widehat{\boldsymbol{e_{M-1}}} \end{bmatrix} = (\boldsymbol{T'})^{-1} \cdot \begin{bmatrix} \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{n_1'}(\boldsymbol{k})] \\ \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{n_2'}(\boldsymbol{k})] \\ \vdots \\ \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{n_{M-1}'}(\boldsymbol{k})] \end{bmatrix}$$

$$\begin{bmatrix} \widehat{\boldsymbol{e_1}} \\ \widehat{\boldsymbol{e_2}} \\ \vdots \\ \widehat{\boldsymbol{e_{M-1}}} \end{bmatrix} = \begin{bmatrix} \frac{M-1}{M} & 0 & \cdots & 0 \\ 0 & \frac{M-1}{M} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{M-1}{M} \end{bmatrix} \cdot \begin{bmatrix} \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_1'}(\boldsymbol{k})] \\ \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_2'}(\boldsymbol{k})] \\ \vdots \\ \boldsymbol{COR}[\boldsymbol{v'}(\boldsymbol{k}), \boldsymbol{g_{M-1}'}(\boldsymbol{k})] \end{bmatrix}$$

$$\widehat{\boldsymbol{e_i}} = \frac{M-1}{M}.\boldsymbol{COR}\left[\boldsymbol{v'(k)}, \boldsymbol{g_i'(k)}\right] \qquad (\mathbf{4.19})$$

Note:

$\widehat{\boldsymbol{e_i}}$ are calculated for (i = 1, 2, …, M-1). Thus, we need to compute $\widehat{\boldsymbol{e_M}}$ also. As we know from the previous discussions that sum of error terms is assumed to be zero.

$$\sum_{i=1}^{M} e_i = 0$$

$$Thus, \quad e_1 + e_2 + \cdots + e_{M-1} + e_M = 0$$

$$e_M = -\sum_{i=1}^{M-1} e_i$$

$$\boldsymbol{Hence}, \qquad \widehat{\boldsymbol{e_M}} = -\sum_{i=1}^{M-1} \widehat{\boldsymbol{e_i}}$$

4.1.4 Filtering to remove the input signal from DSM output

The output of the Delta-Sigma ADC discussed above given by Equation (4.1) also presented below:

$$v(k) = r'(k) + q'(k) + \sum_{i=1}^{M} e_i.g_i'(k)$$

Now while performing the correlation calculations between $v(k)$ and $g_i'(k)$ (to measure errors), we assumed that input signal - $r'(k)$ to be decorrelated with $g_i'(k)$ i.e.

$$\boldsymbol{COR}\left[\boldsymbol{r'(k)}, \boldsymbol{g_i'(k)}\right] = \mathbf{0},$$

which results in the calculations of errors. But in reality, the input signal do cause intervention in the correlation process. In order to circumvent this, we need to remove the input signal from the output.

As we know that our information signal is a low frequency signal i.e. Input signal is confined to low frequencies of the spectrum. Therefore, the Modulator output can be made devoid of the input signal by passing the output through a High Pass Filter [33].

$$v'(k) = \mathrm{h(k)} * v(k)$$

Where $h\,(k)$ is the impulse response of High Pass Filter as presented in Fig. 4-2.

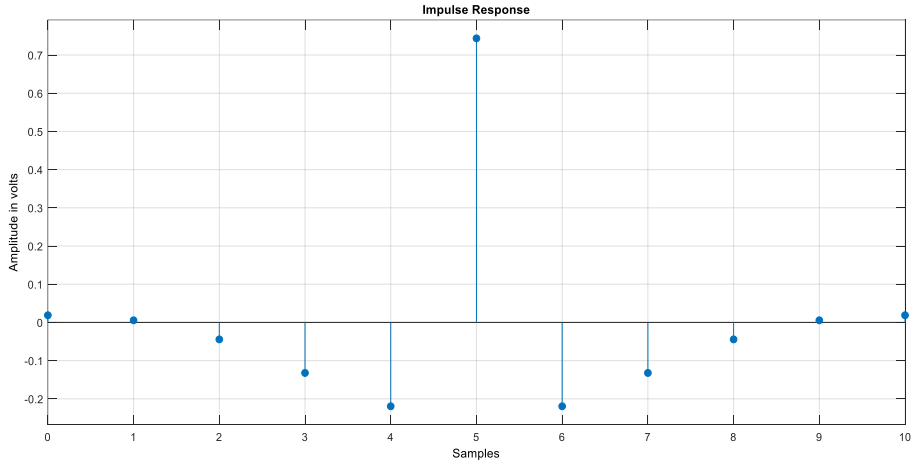

Fig. 4-2 Impulse Response of HPF

The correlation of noise with selecting signal i.e. $COR\left[\, q'(k), g_i^{'}(k)\right]$ will be a very small value and hence will be considered zero. After passing the ADC output through a High Pass Filter, input signal will be removed but it also leads to removal of some DAC error, which lies at low frequencies.

From the basics of filtering, we can say that $v'(k)$, that is the filtered $v(k)$, will comprise of many forms of $v(k)$. Each form will be suspended by distinct clock periods and multiplied by corresponding Filter coefficients. For correct measurement of DAC error, we need to use some other shifted version of $v'(k)$ (containing the DAC Error) for the correlation.

In practice, we take the form of $v(k)$ which contains maximum power i.e. the form with maximum value of HPF coefficient. Now we need to coincide $g_i^{'}(k)$ with $v'(k)$ before

correlating them. We can pass $g_i^{'}(k)$ also through a High Pass filter to match the delay as depicted in the Fig. 4-1.

4.1.5 Down Sampling

So far, we have covered all the calculations needed in the correlation block. We can deduct from the calculations that for implementation of such blocks, we need to have multipliers working at a fast speed comparable to the fast speed of the modulator for the system to operate properly. Nevertheless, implementation of multipliers and dividers at such speeds (GHz) is practically unachievable and if by some means one is able to implement such system, it would be power hungry. Thus, we need to slow down the rate before performing such operations.

For this to achieve, we will reduce our sampling rate with the help of down samplers in the system. Down sampling is a process by which we can reduce the rate of sampling i.e. the frequency of operation is lowered. By doing so, the blocks working at high speed can be easily integrated with the blocks working at lower speed.

4.2 Error Removal using Digital DAC Replica

To understand the method proposed for the removal of error coefficients from the system, we need to get insight about the idea applied in the method. Consider a General Diagram of Sigma Delta Modulator as presented in Fig. 4-3 below:

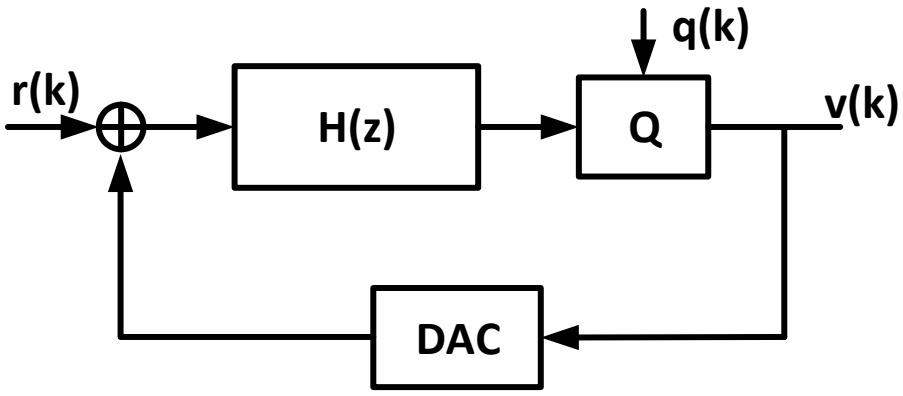

Fig. 4-3 Block Diagram of DSM

In this the Quantizer (Q), DAC and Loop transfer function H (z) are embedded in the loop shown above, the quantization error appears in the Modulator output after passing through a noise transfer function (NTF) of the Modulator and the input signal appears in the modulator output after passing through the signal transfer function (STF) of the Modulator.

$$v(k) = u(k).stf(k) + q(k).ntf(k)$$

$$\text{STF (z)} = \frac{H(z)}{1 + H(z)}$$

$$\text{NTF (z)} = \frac{1}{1 + H(z)}$$

The loop transfer function H (z) is basically a low pass function or we can say it is just the transfer function of an integrator given by:

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

Putting this value of H (z) in the equation of NTF (z) we get,

$$\text{NTF (z)} = \frac{1}{1 + \frac{z^{-1}}{1 - z^{-1}}} = 1 - z^{-1}$$

Which is nothing but the transfer function of a High Pass Filter since the NTF (z) has a zero at DC.

Thus, we can broadly say that the Quantization Noise appears at the ADC output after getting High-passed through the transfer function of NTF (z).

We can infer from above result that "if any error e(k) is introduced in the feedback loop, this error gets high passed just after the point of its introduction in the loop". Applying the same theory for the DAC error, we can conclude that the **DAC error also gets high passed just after the point of its introduction in the loop.** But in spite of being high passed this error appears at the output along with the input signal (since signal is not affected by the NTF (z) of the system).

Another question arises whether or not we can just tap the point after DAC in the design to get better performance with the performance close to ideal and negligible amount of DAC error interference in the signal band. The answer is **NO** since the point just after DAC has analog values. We are implementing a data conversion from analog to digital domain. So, we need to have our data in digital domain.

If by some means we could make a replica of the DAC in digital domain similar to the one in the feedback loop and inject the estimated noise terms in the Digital DAC, we could achieve same performance and results. Revisiting the DAC designing section, we can recollect from Equation (1.4) that the DAC output is given as:

$$d_{actual}(k) = \Delta.\sum_{i=1}^{M} t_i(k) + \Delta.\sum_{i=1}^{M} t_i(k)\, e_i$$

In addition, we have seen that Δ of the DAC is cancelled by Δ (quantization step) of the Quantizer. Thus Δ seems to have no role in the output equation of the ADC. Hence, to make a Digital DAC, we need to implement the DAC Equation (1.26) in a digital fashion as follows:

$$v_{Replica\_DAC\_correct}(k) = \sum_{i=1}^{M} t_i(k) + \sum_{i=1}^{M} t_i(k).\widehat{e_i} \qquad (1.26)$$

The model diagram of the proposed Digital DAC Correction is presented in Fig. 4-4 below. The Replica Digital DAC will takes two inputs, which includes measured errors $\widehat{e_i}$ (calculated in previous section of error measurement) and $t_i(k)$ (thermometric coded output of the modulator) to generate the corrected output $v_{Replica\_DAC\_correct}(k)$. To implement the DAC in a Digital Fashion, we have performed the following procedure:

1) The inputs $t_i(k)$ and $\widehat{e_i}$ are scaled to have 16-bit precision by multiplying them with 2^16 and rounding the results.
2) Digital Multipliers and adders are used to implement the Equation (1.26).

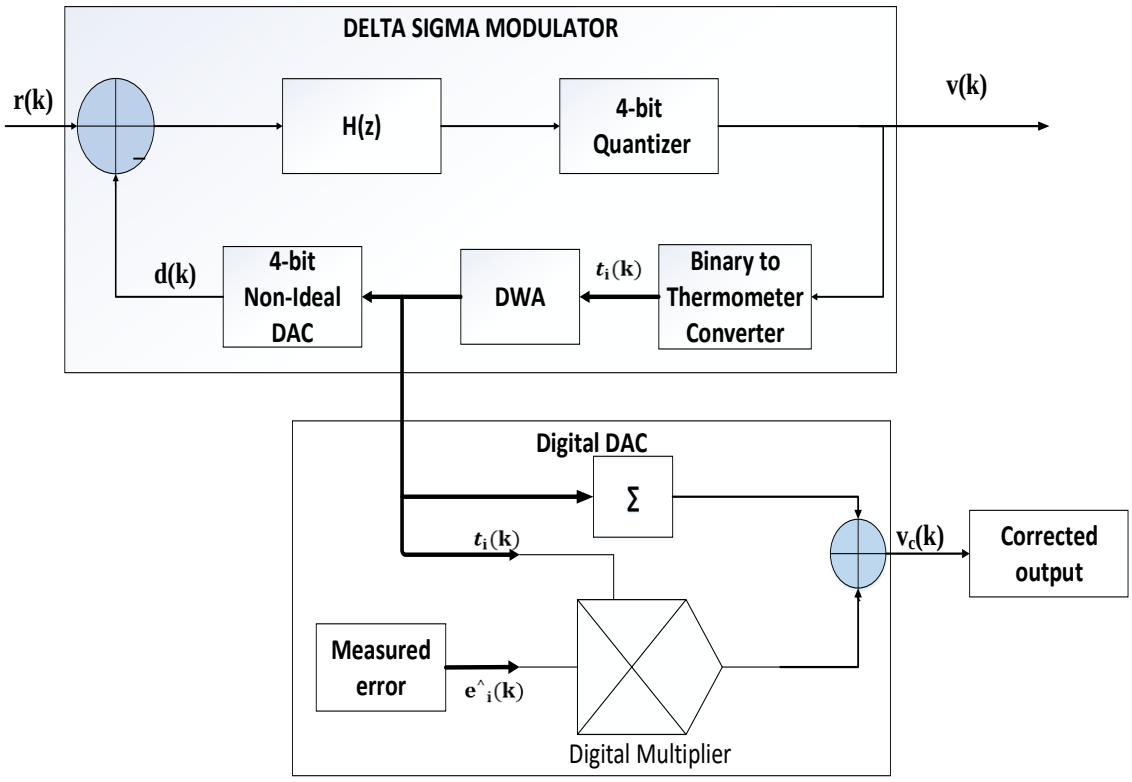

Fig. 4-4 Proposed Correction technique

CHAPTER 5

IMPLEMENTATION AND SIMULATION RESULTS

To demonstrate the functionality of the proposed method, a Continuous Time Third-order Sigma Delta Modulator has been built and the Measurement technique and Error Removal technique has been applied on the actualized ADC. The Fig. 5-1 below shows a third order Sigma Delta ADC realized using MATLAB software. All the simulations have been performed on Simulink and using Schreier Toolbox for Sigma-Delta Modulators [34].

5.1 SDM Modulator

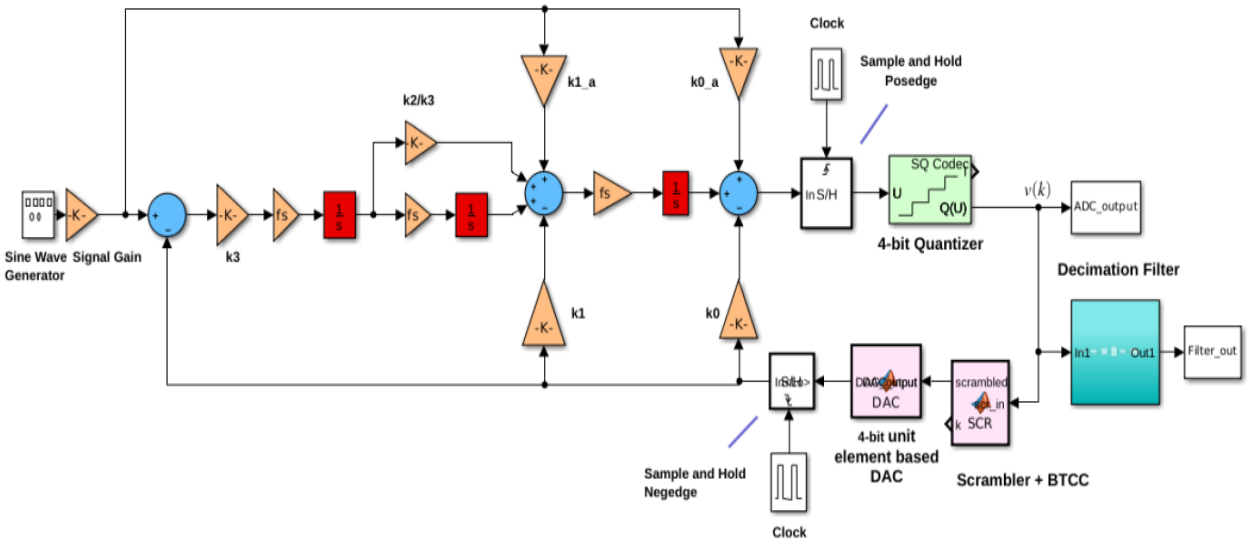

Fig. 5-1 Continuous Time 3rd order 4-bit Delta Sigma ADC

In Fig. 5-1 shown above, a third order CT Delta Sigma ADC is built in CIFF-B (Cascade of Integrators with feed-forward and feedback) fashion [27]. The input is a sine wave having frequency f_{in} = 9.30 MHz with amplitude ranging from – 1 to + 1 appeared in Fig. 5-2. Since the input is a continuous time signal, we need to sample the input at a finite

sample time. The sampling frequency chosen is f_s = 2.4 GHz and the value of OSR chosen is 30. Thus, a sample and hold circuit is being placed just after the integrators. The sampled signal is also depicted in Fig. 5-3.

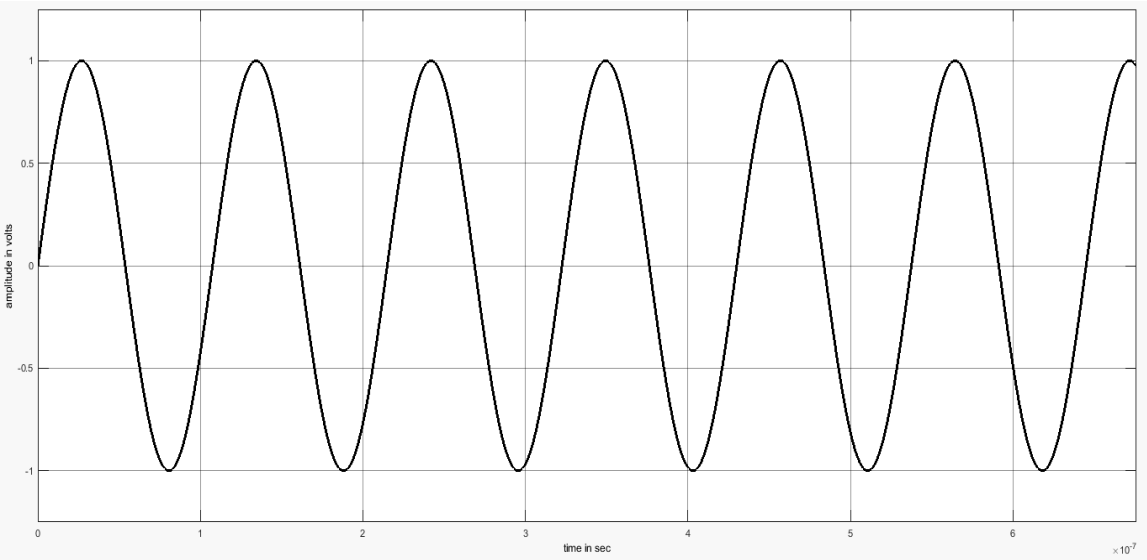

Fig. 5-2 Input signal having f_{in} = 9.30 MHz

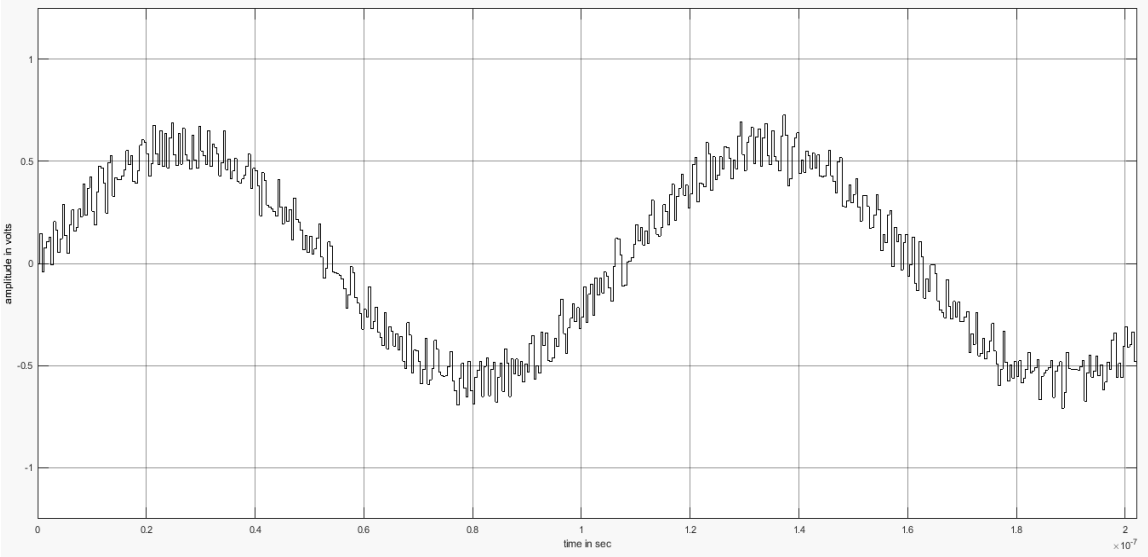

Fig. 5-3 Sampled input signal using a S/H circuit at f_s = 2.4 GHz

A 4-bit Quantizer has been inserted which converts the input signal into a set of defined values also known as **code word**. The output of ADC passes through a Scrambler + BTCC block that converts the code word into thermometer code and randomizes the bit values. The output of the Scrambler + BTCC block goes to the 4-bit unit element based DAC. Since we have used a 4-bit Quantizer in the design, we also need to convert back the

digital values into analog values before comparing them with the input signal. Therefore, a 4-bit DAC has been used which converts the digital value into analog value.

As we can see in the design, two Sample and Hold circuits have been placed, one before the Quantizer and one after the DAC. This has been done to incorporate delay in the loop. The first S/H circuit samples the inputs at positive edge of the clock and the second S/H circuit samples the data at negative edge of the clock thus leading to a delay of half clock period. As we have already studied in the theory part, delay free loop cannot exist in practical world. The delay time is the time taken by the Quantizer and the DAC to operate properly.

The output waveform of the ADC is presented in Fig. 5-4. Mathematically, if we calculate Signal to Noise Ratio using formula given in Equation (2.14). we get SNR equal to 110 dB. The SNR calculated for the above system with zero DAC noise is shown below in Fig. 5-5. Also, as we can see we have oversampled the signal in the sigma delta ADC. Thus, before taking the output from the modulator we also need to reduce the sample rate i.e. we need to decimate the signal. A Decimator filter has been put after the modulator output to reduce the sample rate of the design and filter the output.

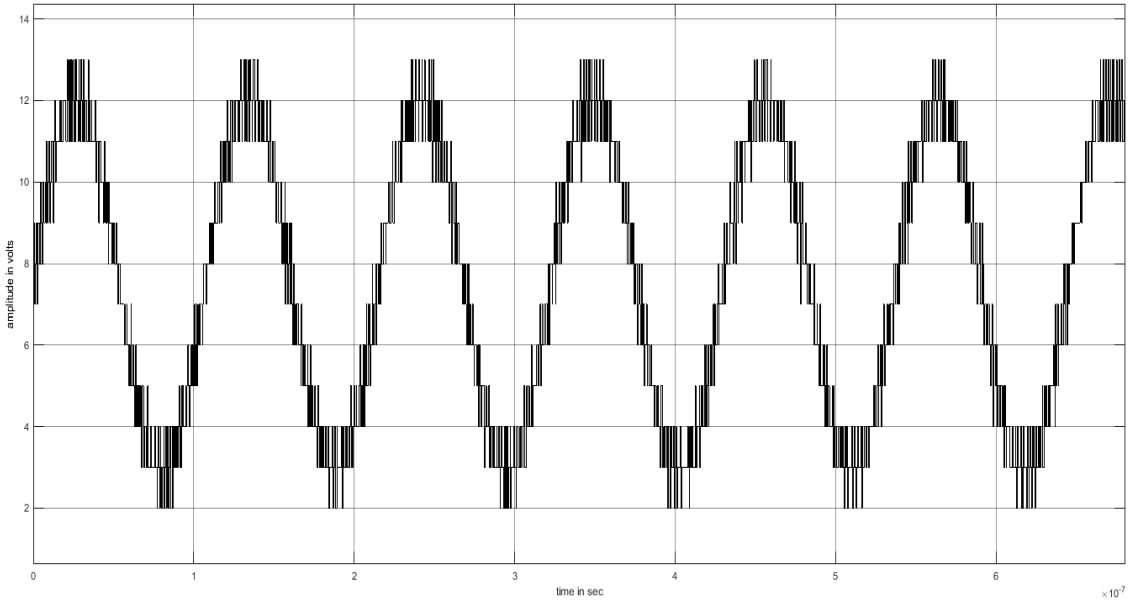

Fig. 5-4 Delta Sigma Modulator output

The Ideal Modulator output introduced in the system is presented in Fig. 5-5:

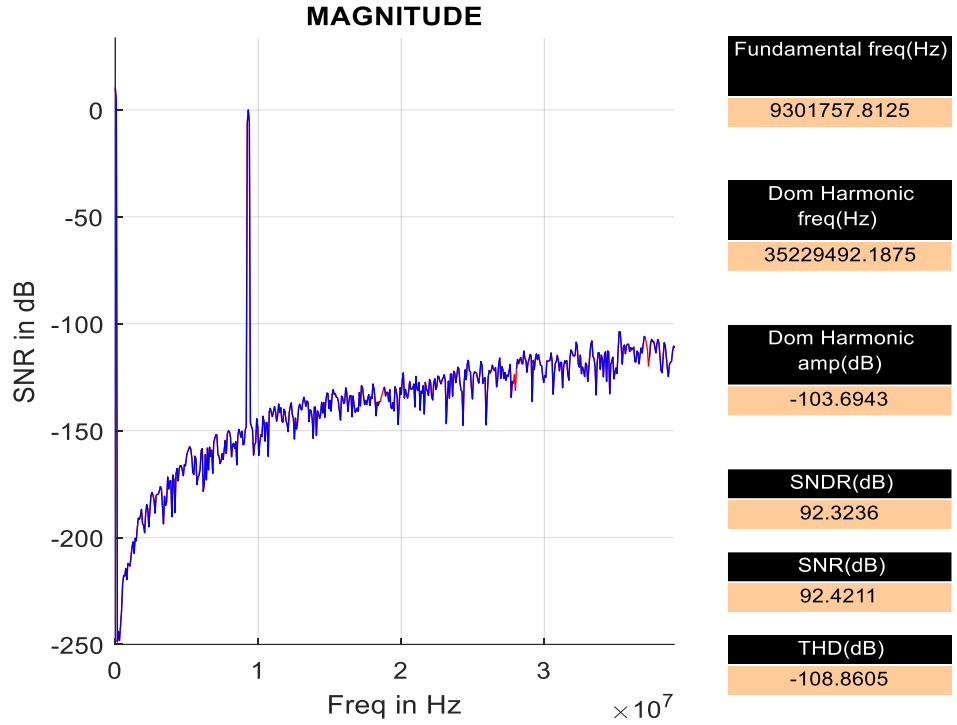

Fig. 5-5 SNR of Ideal Modulator (zero DAC noise)

The Modulator output with DAC noise -1% (No Scrambling) introduced in the system is presented in Fig. 5-6:

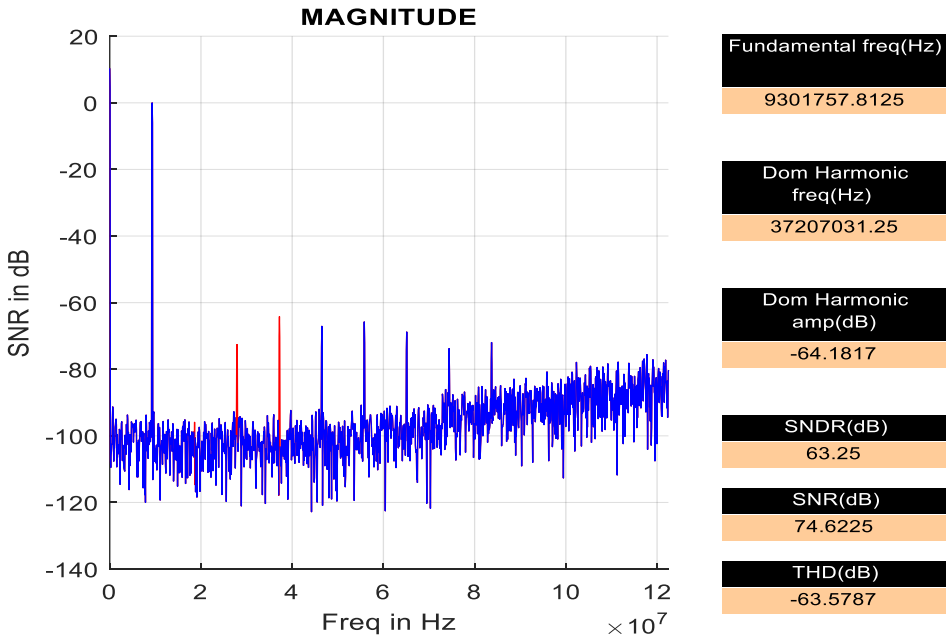

Fig. 5-6 SNR of DSM Modulator with 1% DAC Noise

The Modulator output with DAC noise -1% (Scrambling) introduced in the system is presented in Fig. 5-7:

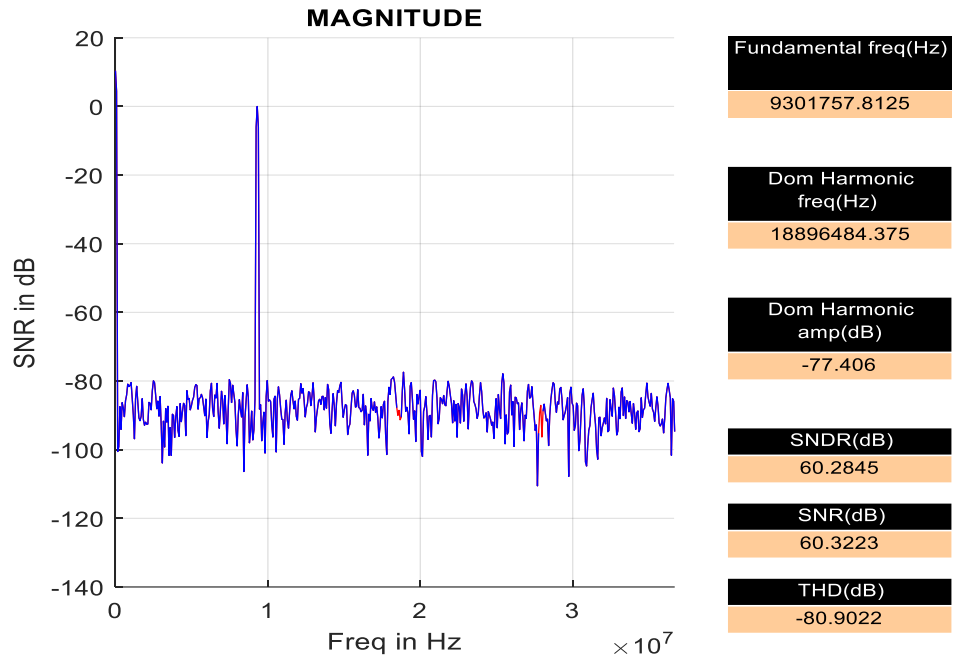

Fig. 5-7 SNR of DSM Modulator with 1% DAC Noise (Scrambled)

The Modulator output with DAC noise -2% (No Scrambling) introduced in the system is depicted below in Fig. 5-8:

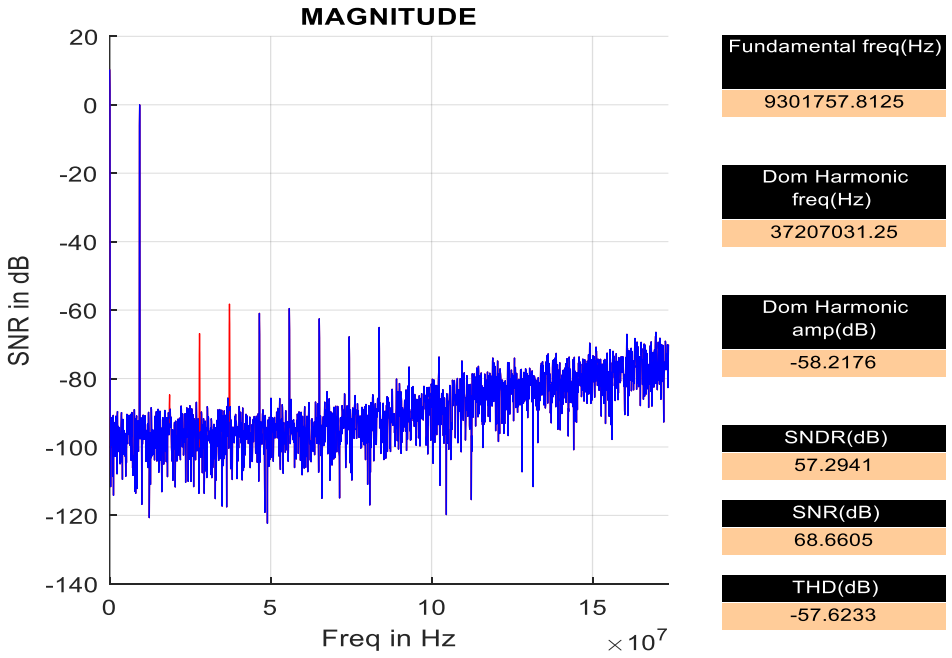

Fig. 5-8 SNR of DSM Modulator with 2% DAC Noise

The Modulator output with DAC noise -2% (Scrambling) introduced in the system is presented below in Fig. 5-9:

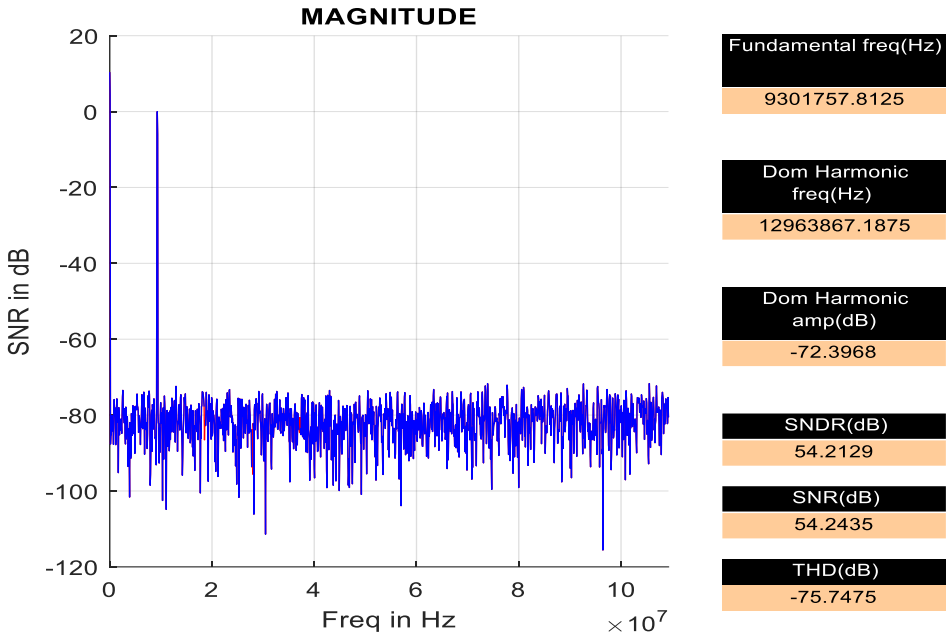

Fig. 5-9 SNR of DSM Modulator with 2% DAC Noise (Scrambled)

The Modulator output with DAC noise -3% (No Scrambling) introduced in the system is shown below in Fig. 5-10:

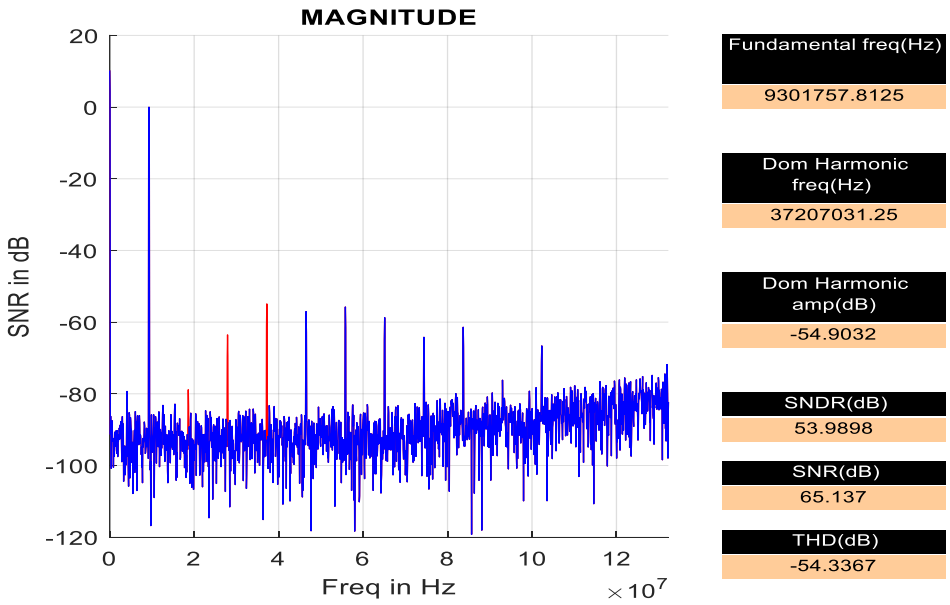

Fig. 5-10 SNR of DSM Modulator with 3% DAC Noise

The Modulator output with DAC noise -3% (Scrambling) introduced in the system is shown below in Fig. 5-11:

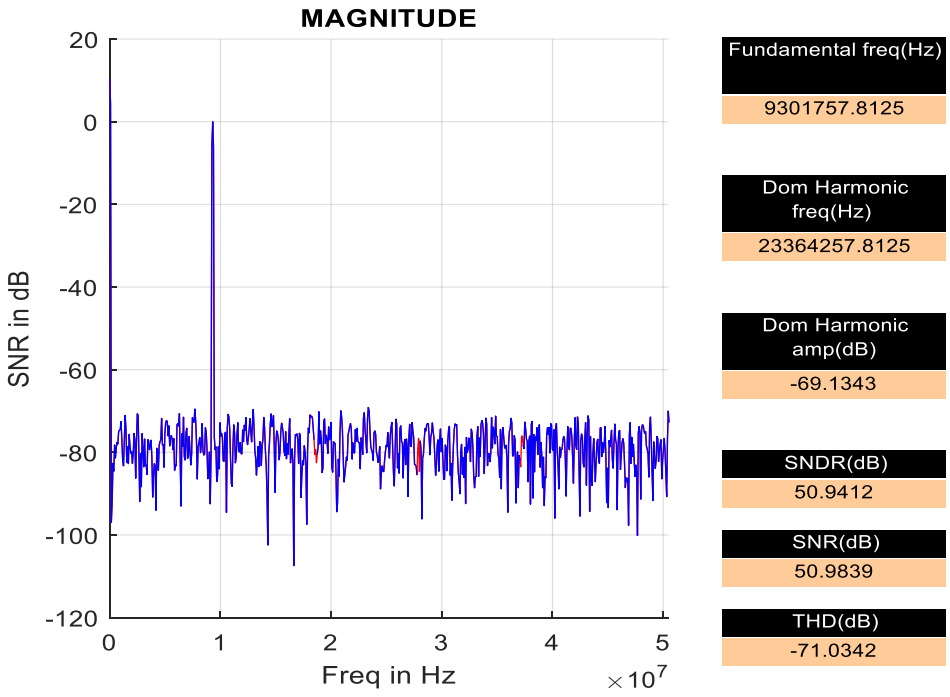

Fig. 5-11 SNR of DSM Modulator with 3% DAC Noise (Scrambled)

The Modulator output with DAC noise -10% (No Scrambling) introduced in the system is shown below in Fig. 5-12:

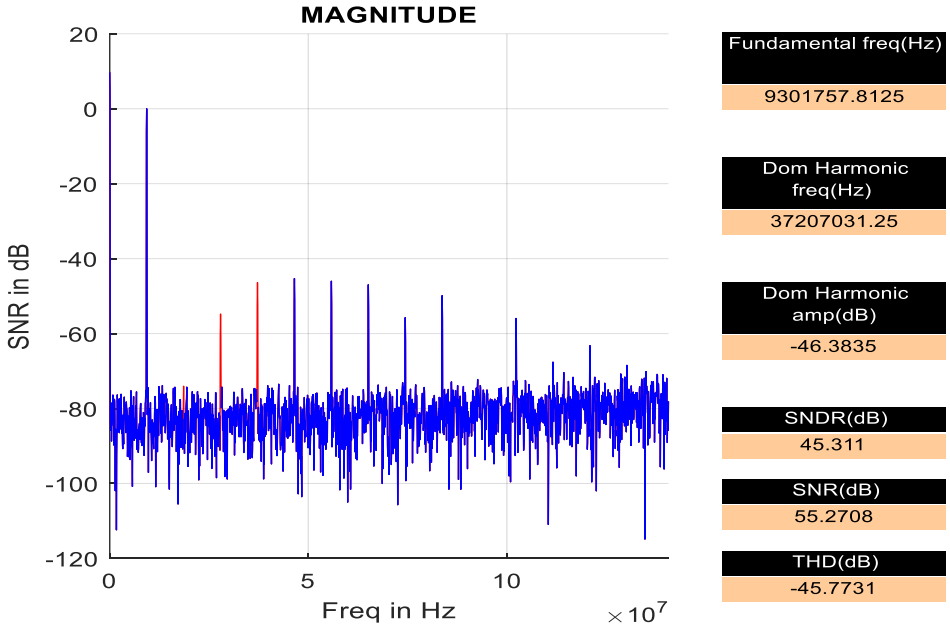

Fig. 5-12 SNR of DSM Modulator with 10% DAC Noise

The Modulator output with DAC noise -10% (Scrambling) introduced in the system is shown below in Fig. 5-13:

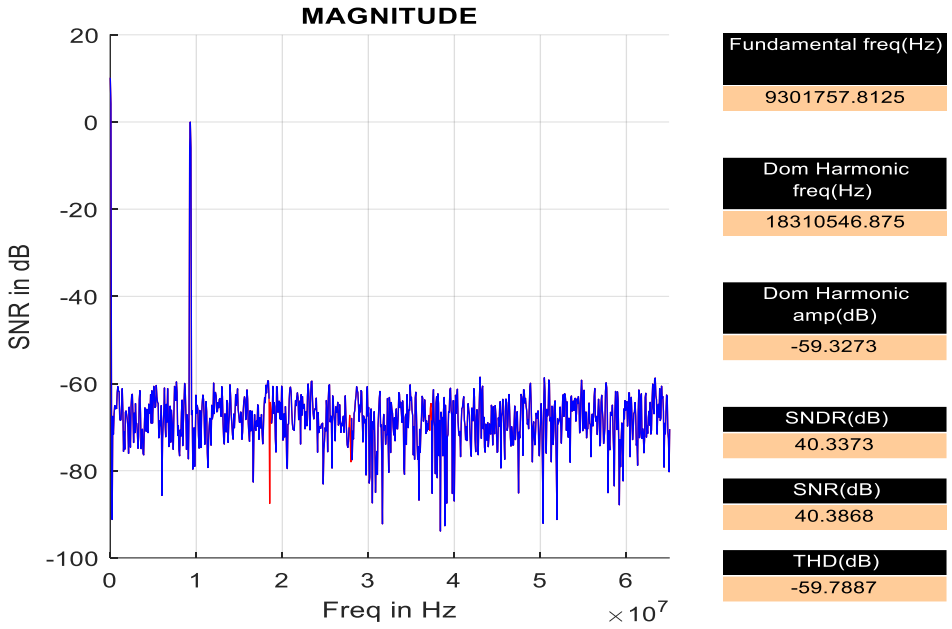

Fig. 5-13 SNR of DSM Modulator with 10% DAC Noise (Scrambled)

We can see a significant drop in the SNR value by introduction of different DAC noise % in the system that confirms the impact of DAC noise on the system performance. Hence, we need to deal with the DAC noise.

5.2 Error Measurement

Fig. 5-14 and Fig. 5-15 shows the implemented DAC noise measurement model used in the design. Here we have used FIR Filters followed by down samplers as depicted in purple color. To improve the design, we have implemented the filters in poly-phase fashion as shown in Fig. 5-15 The advantages of using the poly phase filters have already been discussed in chapter 2.

The below models have been simulated by injecting different percentage of DAC noise in the feedback DAC present in the model. The noise injected are random in nature and are assumed to have zero mean.

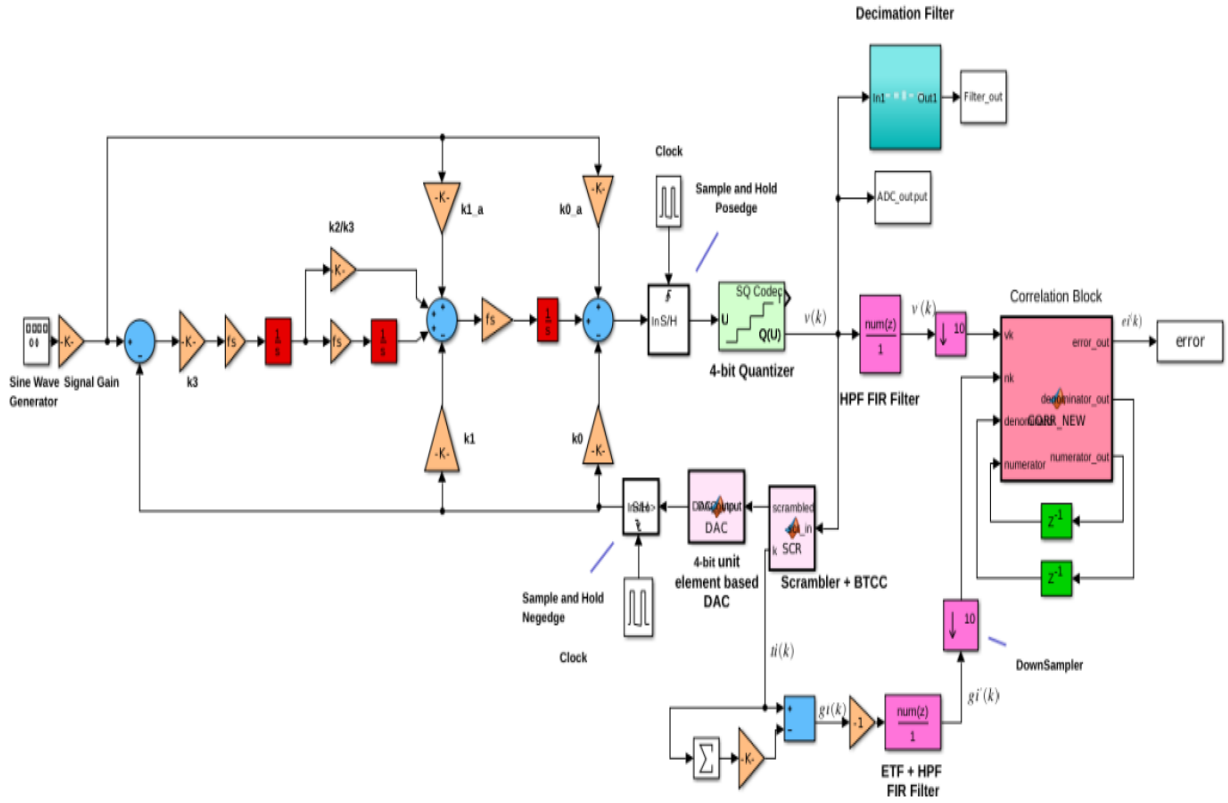

Fig. 5-14 Error Measurement Model

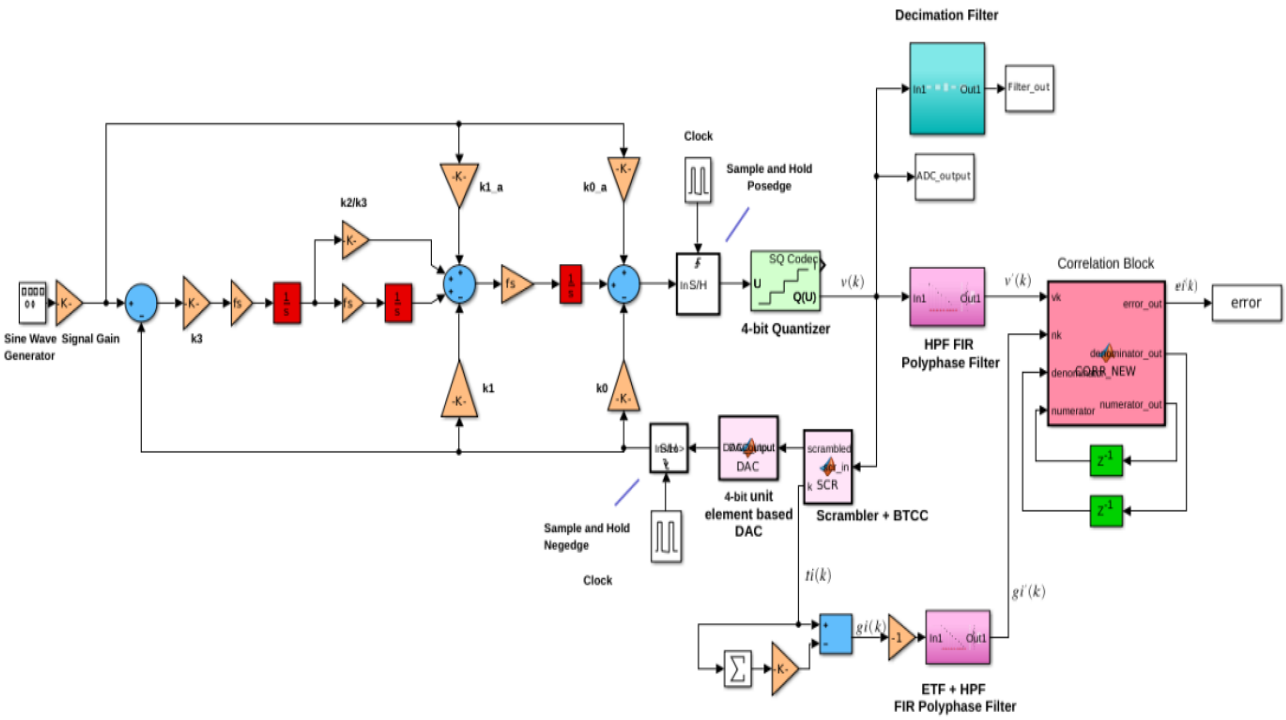

Fig. 5-15 Error Measurement Model using Poly-phase Filter Implementation

Fig. 5-16 – Fig. 5-19 below shows different random noise values introduced in the DAC and the measured values of noise using the measurement technique.

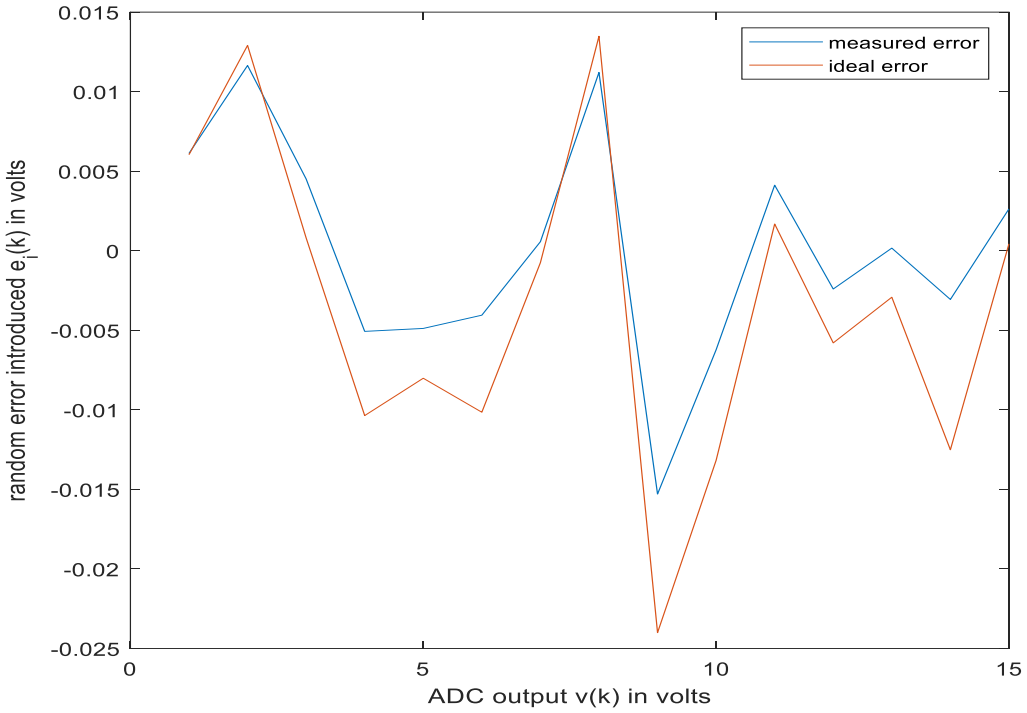

Fig. 5-16 Comparison of 1% Introduced and Measured Noise

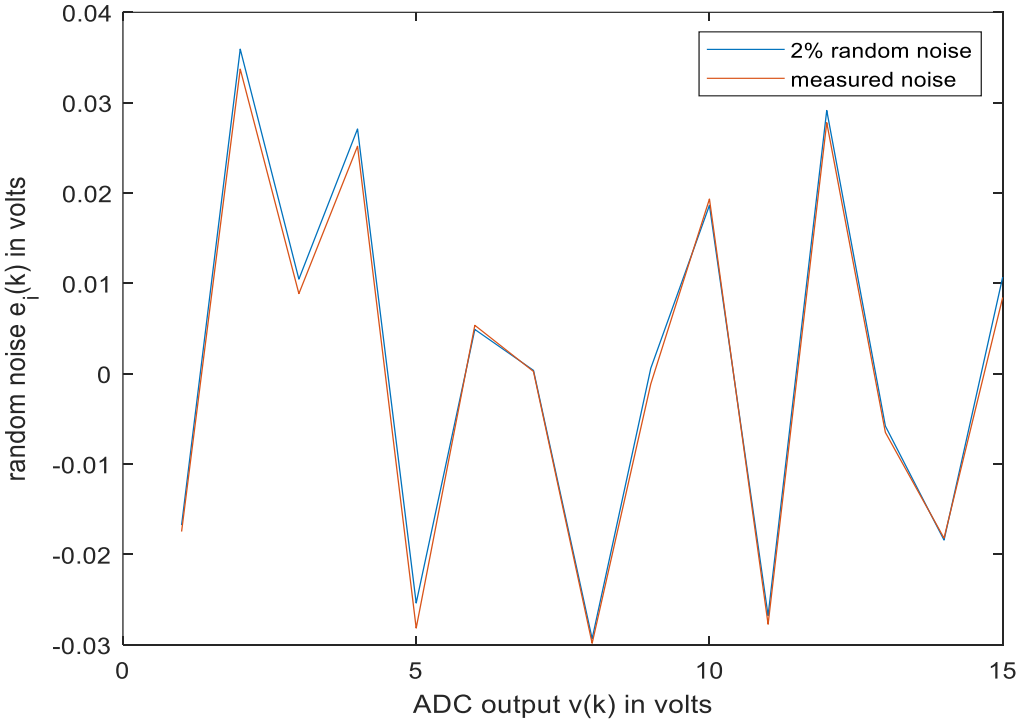

Fig. 5-17 Comparison of 2% Introduced and Measured Noise

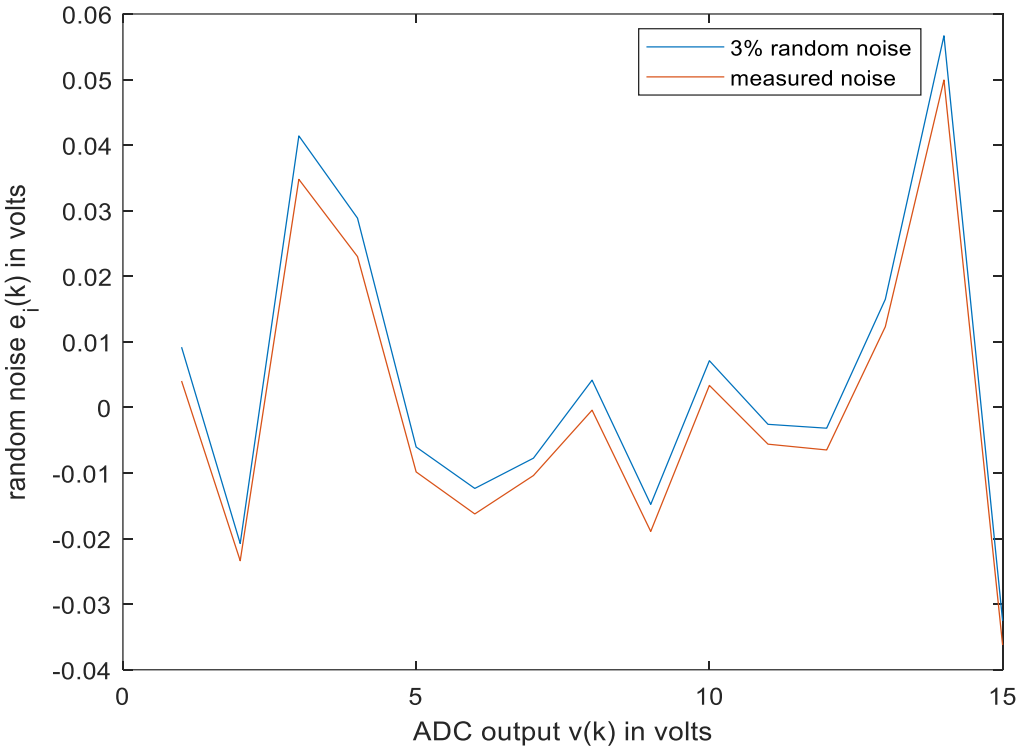

Fig. 5-18 Comparison of 3% Introduced and Measured Noise

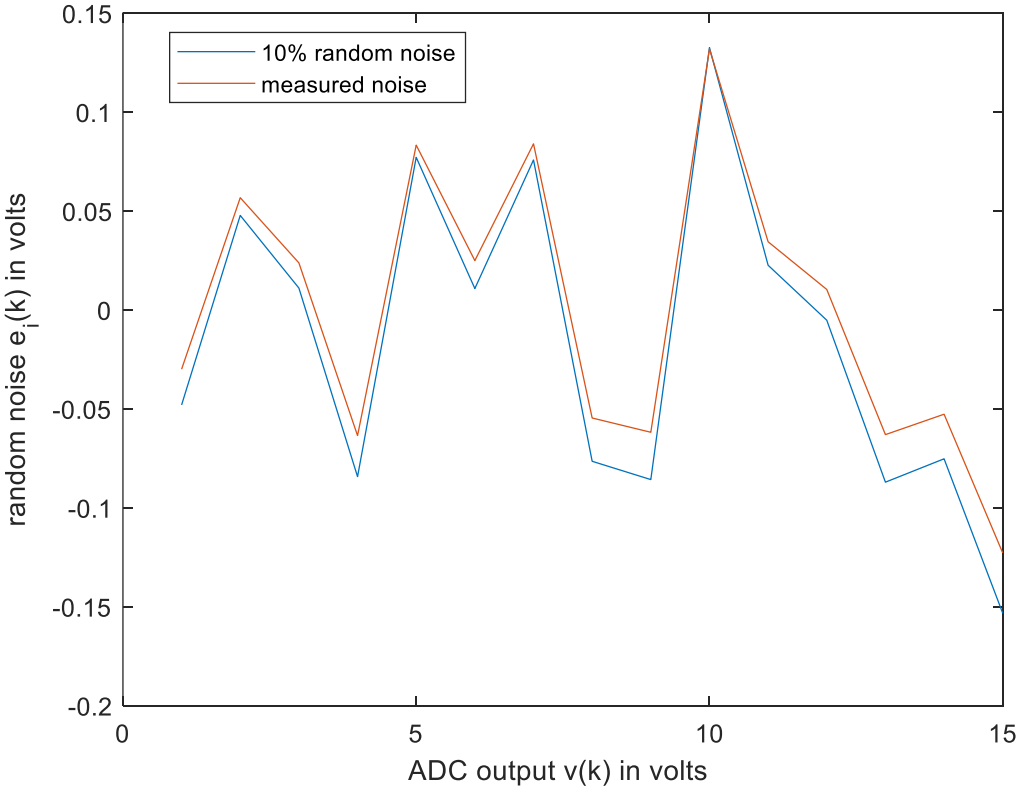

Fig. 5-19 Comparison of 10% Introduced and Measured Noise

5.3 Error Removal using Digital DAC Replica

Fig. 5-20 below shows the block diagram of implemented Correction block in MATLAB.

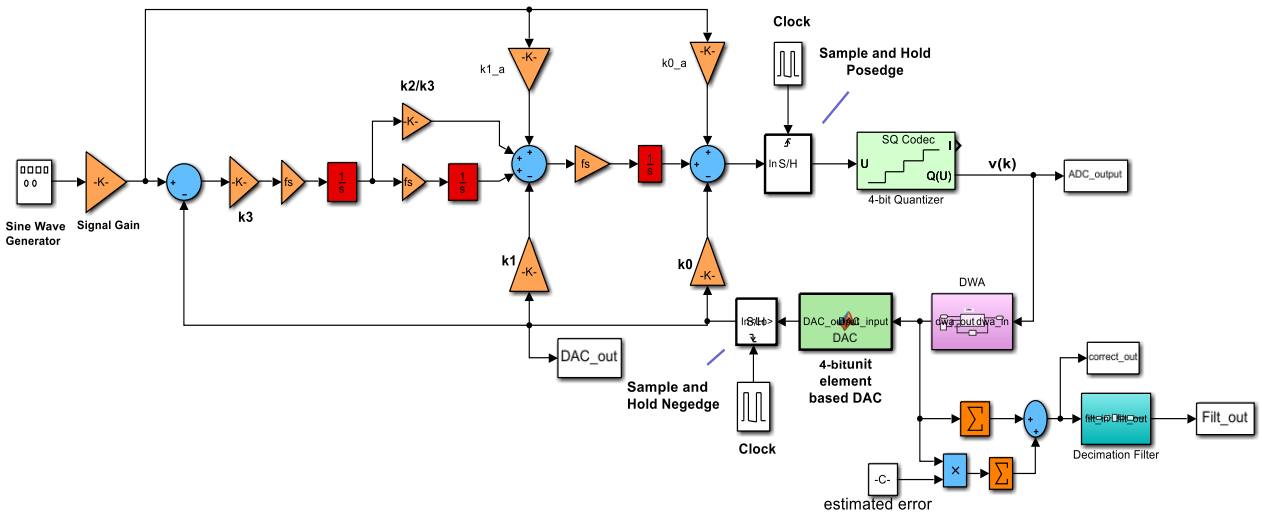

Fig. 5-20 Implemented Correction Block in MATLAB

The corrected ADC outputs with different noise profiles are being presented below in Fig. 5-21 to Fig. 5-24:

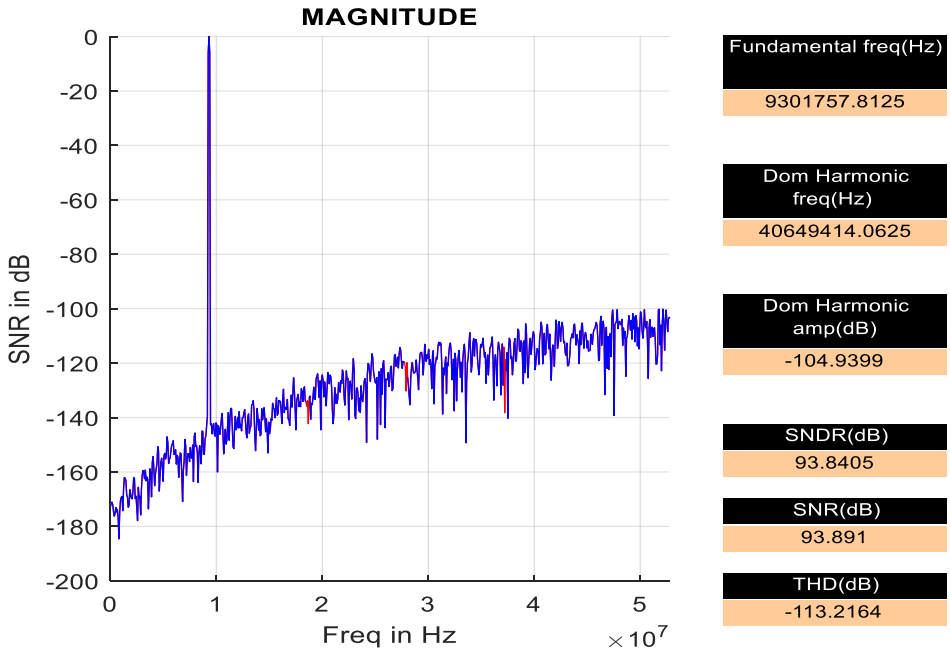

Fig. 5-21 Corrected output after introduction of 1% noise in the system

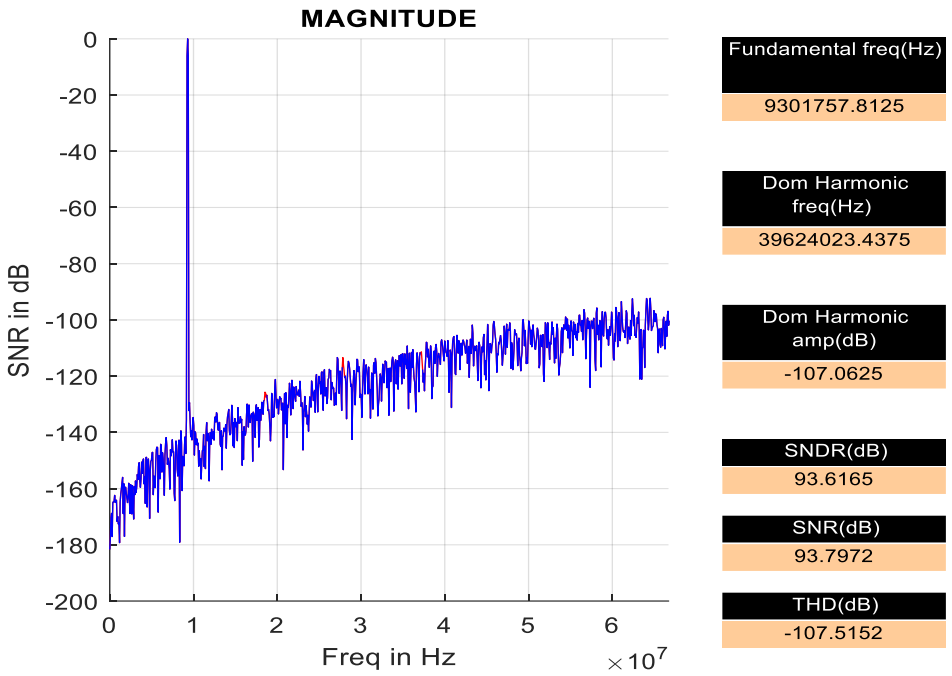

Fig. 5-22 Corrected output after introduction of 2% noise in the system

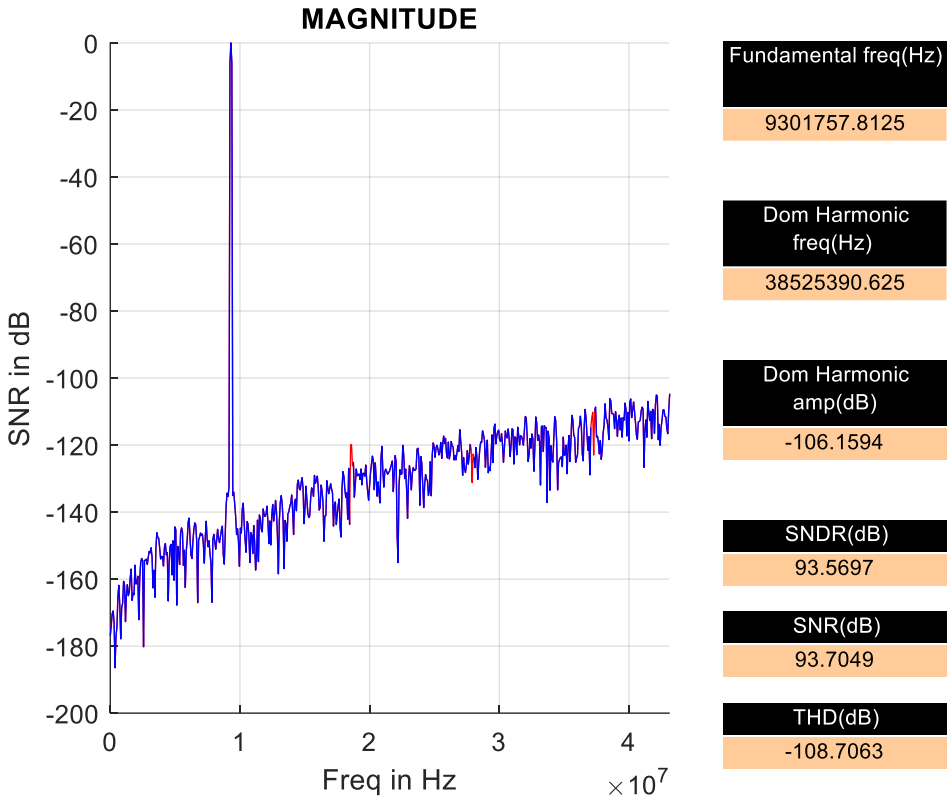

Fig. 5-23 Corrected output after introduction of 3% noise in the system

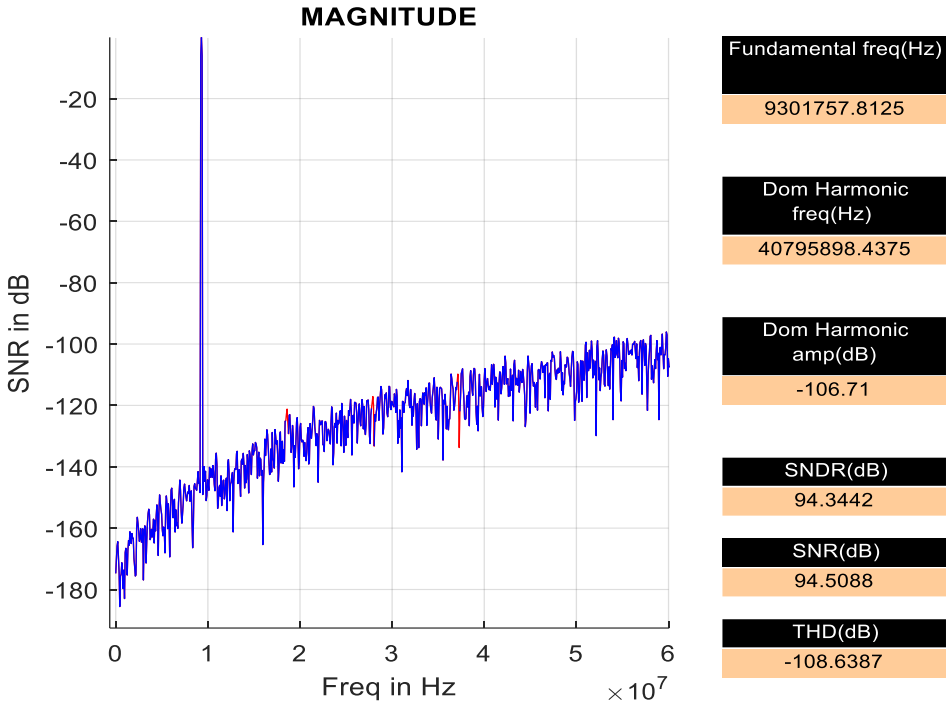

Fig. 5-24 Corrected output after introduction of 10% noise in the system

The corrected output of the system is passed through a decimation filter to recover the original signal efficiently. The filtered output is presented in Fig. 5-25 below:

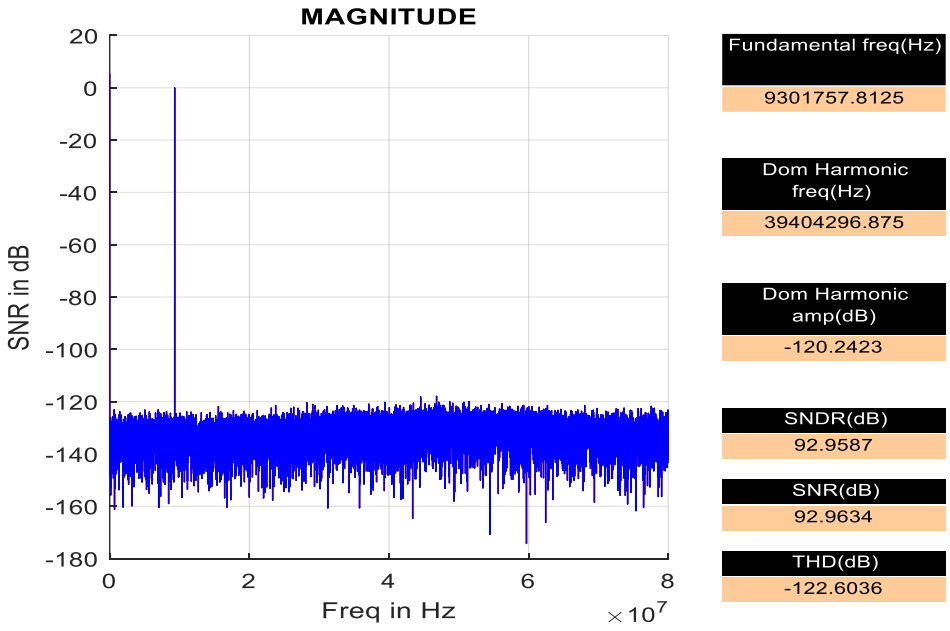

Fig. 5-25 Filtered output after passing through the Decimation Filter

CONCLUSION

In this work, we have presented a novel advanced procedure for the recognition and rectification of DAC noise from the output of DSM. Various concerned subjects have been dealt for the proper understanding of the system. They comprise:

1. The basics of delta sigma modulator and the need of quantizers with multiple bits as opposed to single bit ones.
2. Demonstration of the DAC noise using unit components and its effects on ADC output.
3. Investigation of different techniques of managing the DAC noise along with the limitations.
4. Reproduction of the proposed procedure with the help of a 3^{rd} order Delta-Sigma Modulator realized in MATLAB Simulink.

The procedure was found to be very competent in removing noise from the output of the modulator and thus increasing performance of the realized system. The procedure is versatile in nature as it is capable of rectifying any random amount of noise introduced by the DAC in the system.

REFERENCES

[1] J. Archana and V. Verma, "Sigma-Delta Modulator Design and Analysis for Audio Application," in International Journal of Engineering Trends and Technology (IJETT), vol. 28, no. 1, Oct 2015.

[2] M. Bashir, S. Rao Patri and K. S. R. Krishna Prasad, "A 159 µW, Fourth Order, Feed-forward, Multi-bit Sigma-Delta Modulator for 100 kHz Bandwidth Image Sensors in 65-nm CMOS Process," Radio Engineering, vol. 27, no. 2, June 2018.

[3] J. R. Custodia and J. Goes, "A 1.2-V 165-µW 0.29-mm^2 Multi-bit Sigma-Delta ADC for Hearing Aids Using Nonlinear DACs and With Over 91 dB Dynamic-Range," in IEEE Transactions on Biomedical Circuits and Systems.

[4] V. Kuri and Dr. A. Chattopadhyay, "Review on Sigma-Delta Modulator," in International Journal of Engineering Research and General Science, vol. 3, Issue. 5, Oct. 2015.

[5] D. A. Kerth, D. B. Kasha, T. G. Mellissionos, D. S. Piasecki, and E. J. Swanson, "A 120 dB linear switched-capacitor delta-sigma modulator," IEEE International Solid-State Circuits Conference, Digest of Technical Papers, pp. 196-197, Feb. 1994.

[6] Y. Matsuya, K. Uchimura and A. Iwata, "A 16 b oversampling A-to-D conversion technology using triple-integration noise shaping," IEEE Journal of Solid-State Circuits, vol. sc-22, pp. 921-929, Dec. 1987.

[7] J. Snehalatha, “The Design of Multi Bit Quantizer Sigma-Delta Modulator Analog to Digital Converter,” International Journal of Science and Research (IJSR), vol. 4, Issue 9, Sept. 2015.

[8] I. Fujimori et al., “A 90-dB SNR 2.5-MHz output-rate ADC using cascaded multi-bit delta-sigma modulation at 8x oversampling,” IEEE J. Solid-State Circuits, vol. 35, no. 12, pp. 1820.1828, Dec. 2000.

[9] K. Vleugels, S. Rabii and B. Wooley, “A 2.5 V Sigma-Delta Modulator for Broadband Communications Applications,” IEEE J. Solid-State Circuits, vol. 36, no. 12, pp. 1887.1899, Dec. 2001.

[10] Y. Geerts, M. S. J. Steyaert and W. Sansen, “A high-performance multi-bit delta-sigma CMOS ADC,” IEEE J. Solid-State Circuits, vol. 35, no. 12, pp. 1829-1840, Dec. 2000.

[11] A. Yasuda, H. Tanimoto and T. Iida, “A third-order Δ-Σ modulator using second-order noise-shaping dynamic element matching,” IEEE J. Solid-State Circuits, vol. 33, no. 12, pp. 1879-1886, Dec. 1998.

[12] R. Schreier and B. Zhang, “Noise-shaped Multi-bit D/A Converter employing Unit Elements,” Electronic Letters, vol. 31, no. 20, pp. 1712–1713, Sept. 28, 1995.

[13] R. T. Baird and T. S. Fiez, “Linearity enhancement of multibit Delta Sigma and D/A converters using data weighted averaging,” IEEE Trans. Circuits Syst.-II, vol. 42, no. 12, pp. 753-762, Dec. 1995.

[14] R. Adams and K. Nguyen, “A 113-dB SNR oversampling DAC with segmented noise-shaped scrambling,” IEEE J. Solid-State Circuits, vol. 33, no. 12, pp. 1871-1878, Dec. 1998.

[15] I. Galton, “Spectral shaping of circuit errors in digital-to-analog converters,” IEEE Trans. Circuits Syst.-II, vol. 44, no. 10, pp. 808–817, Oct. 1997.

[16] P. Rombouts and L. Weyten, “A study of dynamic element-matching techniques for 3-level unit elements,” IEEE Trans. Circuits Syst.-II, vol. 47, no. 11, pp. 1177-1187, Nov. 2000.

[17] S. Reekmans, J. De Maeyer, P. Rombouts, and L. Weyten, “Quadrature mismatch shaping for digital-to-analog converters,” IEEE Trans. Circuits Syst.-I, vol. 53, no. 12, pp. 2529–2538, Dec. 2006.

[18] N. Sun and P. Cao, “Low-Complexity High-Order Vector-Based Mismatch Shaping in Multibit Delta Sigma ADCs,” IEEE Trans. Circuits Syst.-II, vol. 58, no. 12, pp. 872–876, Dec. 2011.

[19] N. Sun, “High-Order Mismatch-Shaping in Multi-bit DACs,” IEEE Trans. Circuits Syst.-II, vol. 58, no. 6, pp. 346–350, June 2011.

[20] R. T. Baird and T. S. Fiez, “A low oversampling ratio 14-b 500-kHz ΔΣ ADC with a self-calibrated multi-bit DAC,” IEEE J. Solid-State Circuits, vol. 31, no. 3, pp. 312 -320, March 1996.

[21] C. Petrie and M. Miller, “A background calibration technique for multi-bit delta-sigma modulators”, Proceedings of the 2000 IEEE International Symposium on Circuits and Systems, vol. 2, pp. 29-32, May 2000.

[22] K. Lokesh Krishna, M. Jagadeesh, T. Ramashri and K. Anuradha, “Design and Implementation of a Low Power Second Order Sigma-Delta ADC,” International J. of Computer Science and Information Technologies (IJCSIT), vol. 5(5), pp. 6044-6049, 2014.

[23] S. R. Norsworthy, R. Schreier and G. C. Temes, “Delta-Sigma Data Converters,” IEEE Press, Piscataway, NJ, 1997.

[24] J. C. Candy and G. C. Temes, “Oversampling Delta-Sigma Data Converters,” IEEE Press, New York, 2014.

[25] S. R. Kulkarni, “Introduction to Electrical Signals and Systems,” Princeton University, 1994

[26] A. V. Oppenheim et al, “Signals and Systems,” Prentice Hall, 2nd edition, August 6, 1996.

[27] S. Pavan, R. Schreier, and Gabor C. Temes, “Understanding Delta-Sigma Data Converters, Second Edition,” The Institute of Electrical and Electronics Engineers, Inc. Published by John Wiley & Sons, Inc.

[28] B. Razavi, “The Delta-Sigma Modulator,” Spring 2016, IEEE Solid-State Circuits Magazine.

[29] L. R. Rabiner, “Multi-rate Digital Signal Processing”, Princeton-Hall,1983

[30] U. K. Moon, J. Silva, J. Steensgaard and G. C. Temes, “A switched-capacitor DAC with analog mismatch correction,” in Proceedings of the 2000 IEEE Internationa*l* Symposium on Circuits and Systems, vol. 4, pp. 421-424, May 2000.

[31] M. Sarhang-Nejad, “Realization of a High-Resolution Multi-bit Sigma-Delta Analog/Digital Converter,” Ph.D. thesis, University of California, Los Angeles, 1991.

[32] P. Kiss, U. Moon, J. Steensgaard, J. T. Stonick, and G. C. Temes, **“**Multi-bit delta-sigma ADC with mixed-mode DAC error correction,” in Proceedings of the IEEE International Symposium on Circuits and Systems, pp. I.280. I.283, 8 May 2001.

[33] X. Wang, U. Moon, M. Liu, and G. Temes, “Digital correlation technique for the estimation and correction of DAC errors in multi-bit MASH delta-sigma ADCs,” IEEE Int. Symp. Circuits Syst., vol. 4, pp. 691-694, May 2002.

[34] Richard Schreier, “The delta-sigma toolbox 5.1 for Matlab 5.0, Matlab code and documentation,” April 1998.