PARTIAL BINARY TREE NETWORK (PBTN): A NEW DYNAMIC ELEMENT MATCHING (DEM) APPROACH FOR DIGITAL ANALOG CONVERTER (DAC)

WILLIAM YEAP KEAT SEONG

UNIVERSITI SAINS MALAYSIA

2017

PARTIAL BINARY TREE NETWORK (PBTN): A NEW DYNAMIC ELEMENT MATCHING (DEM) APPROACH FOR DIGITAL ANALOG CONVERTER (DAC)

by

WILLIAM YEAP KEAT SEONG

Thesis submitted in partial fulfillment of the requirement for the degree of
Bachelor of Engineering (Electronic Engineering)

June 2017

ACKNOWLEDGEMENT

It is my pleasure to direct my sincerest gratitude towards Dr Mohd Tafir Mustaffa for the support and guidance provided throughout this research. Dr Tafir’s timely support has been instrumental in providing much needed direction and knowledge in order to successfully complete this research.

Very special thanks to Mr. Lim Yong Cheng, who was the predecessor of this research, for sharing his valuable knowledge and experience in dealing with the research previously. Mr. Lim’s support proved to be crucial in ensuring the adequate knowledge is transferred.

Last but definitely not least, my heartfelt appreciation to friends, family, and IEEE USM Student Branch for always staying by my side throughout my journey in university.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

ADC	Analog Digital Converter
BTN	Binary Tree Network
DAC	Digital Analog Converter
DEM	Dynamic Element Matching
DNL	Differential Non-Linearity
FRDEM	Full Random Dynamic Element Matching
GCN	Generalized Cube Network
INL	Integral Non-Linearity
LSB	Least Significant Bit
MSB	Most Significant Bit
PBTN	Partial Binary Tree Network
SFDR	Spurious Free Dynamic Range

RANKAIAN POKOK SEBAHAGIAN PERDUAAN (PBTN): KAEDAH PEMADANAN ELEMEN DINAMIK (DEM) BARU UNTUK PENUKAR DIGITAL ANALOG (DAC)

ABSTRAK

Penukar Digital Analog (DAC) merupakan segmen penting yang sentiasa digunakan dalam kebanyakan sistem digital yang meyakinkan penukaran data yang tepat. Disebabkan faktor-faktor seperti penggunaan voltan yang semakin rendah, kesuntukan masa untuk kajian, serta keperluan lebar jalur yang meningkat, ini telah menyumbang kepada keperluan model DAC untuk bergantung terhadap komponen yang ditetapkan bagi melaksanakan penukaran data. Penghasilan komponen yang sempurna amatlah susah, kesilapan yang tidak sepadan masih senantiasa berlaku dan menyumbang kepada perbezaan di antara nilai komponen yang diingini dan sebenar. Salah satu cara yang popular untuk mengurangkan ralat komponen ialah Pemadanan Elemen Dinamik (DEM). Teknik perawakan ini memilih nilai input digital secara rawak sebelum memasuki blok DAC. Ini dapat menyamakan purata masa untuk setiap komponen dan kesan perbezaan komponen dalam litar elektronik dapat dikurangkan. Kelemahan bagi penyelidikan yang sedia ada ialah DAC akan memerlukan perkakasan yang berlebihan dan pengekod yang rumit. Dengan ini, banyak get penghantaran akan diperlukan menyebabkan berlakunya gelinciran. Kajian ini melaporkan keputusan simulasi 6-bit 1-MSB PBTN dengan DNL 0.00001895 LSB, INL 0.0001457 LSB, pengunaan kuasa 104.7mW; 6-bit 1-MSB PBTN dengan DNL -0.00928 LSB, INL 0.008669 LSB, pengunaan kuasa 103.8mW; 8-bit 1-MSB dengan DNL -0.00028287 LSB, INL 0.000252 LSB, pengunaan kuasa 115.4mW, and 8-bit 2-MSB dengan DNL 0.001324 LSB, INL 0.0007896 LSB, pengunaan kuasa 114.5mW.

PARTIAL BINARY TREE NETWORK (PBTN): A NEW DYNAMIC ELEMENT MATCHING (DEM) APPROACH FOR DIGITAL ANALOG CONVERTER (DAC)

ABSTRACT

Digital-to-analog converters (DACs) are essential operations in numerous digital systems which demands high performance data converters. With shrinking supply voltage, budget constraints of test times, and rising bandwidth requirement causing DAC architectures highly relying on matched components to perform data conversions. However, matched components are nearly impossible to manufacture, there will always be mismatch errors which causes discrepancies between the desired value and designed value. A popular method to minimize component mismatch error is Dynamic Element Matching (DEM). This technique is a randomization technique to select one of the appropriate codes for each of the digital input value before entering the DAC block. Using this technique, the time averages of the equivalent components at each of the component positions are equal or nearly equal to reduce the effects of component mismatches in electronic circuits. The drawback of existing designs is that the design would suffer from excessive hardware complexity. A complicated encoding is usually necessary for conventional DEM encoders which will lead to a lot of switch transitions at the same time and it will cause glitches to the output signal. This paper reports the simulation results of 6-bit 1-MSB PBTN with DNL of 0.00001895 LSB, INL of 0.0001457 LSB, power consumption of 104.7mW; 6-bit 1-MSB PBTN with DNL of -0.00928 LSB, INL of 0.008669 LSB, power consumption of 103.8mW; 8-bit 1-MSB with DNL of -0.00028287 LSB, INL of 0.000252 LSB, power consumption of 115.4mW; and 8-bit 2-MSB with DNL of 0.001324 LSB, INL of 0.0007896 LSB, power consumption of 114.5mW.

CHAPTER 1

INTRODUCTION

1.1 Background

Digital signals have the advantage of over analog signals is largely due the fact that digital signals are more immune to noise and imperfections, problems which greatly hinders the precision of analog signals. The role of DACs are illustrated in Figure 1-1.

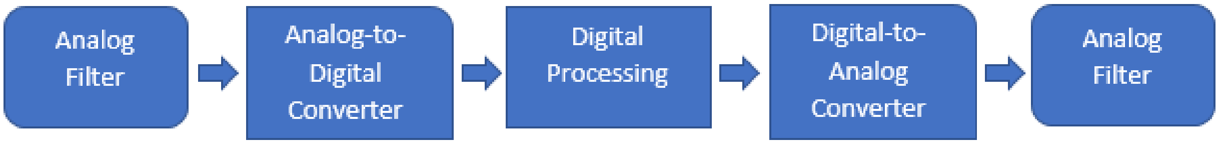

Figure 1-1 Role of DAC is digital signal processing

DACs are essential operations in many digital systems which required high performance data converters. With shrinking of supply voltage, budget constraints of test times, and rising bandwidth requirement causing DAC architectures highly relying on matched components to perform data converters. However, components matched are nearly impossible to fabricate, there are always mismatch errors which caused the difference between the designed and actual component value. Shown in Figure 1-2. Dynamic Element Matching (DEM) is one of the techniques that are commonly used to reduce component mismatch error. This technique is a randomization technique to select one of the appropriate codes for each of the digital input value before entering DAC block. With this technique, the time averages of the equivalent components at each of the component positions are equal or nearly equal to reduce the effects of component differences in electronic circuits. The drawback of existing works is DAC would suffer from excessive digital hardware complexity. A complicated encoding is usually necessary for conventional DEM encoders which will lead to a lot of switch transitions at the same time and it will bring glitches to the output signal. In this research, a new DEM algorithm is proposed on Current-Steering DACs with Partial Binary Tree Network (PBTN) algorithm to overcome glitches transitions with low complexity.

Uncertainties in photolithography edge definition, contact resistance, etching error, and process variation results in deviations and mismatch of resistor ratio from the ideal value. (Kuboki et al, 1982). In addition, other factors such as temperature gradient, alignment error, component aging, and noise contributes to mismatch errors.

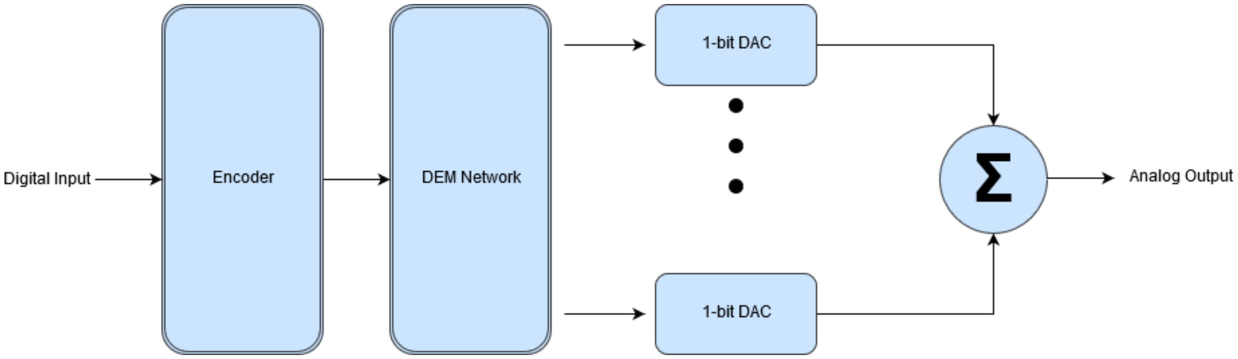

Figure 1-2 Structure of DEM DAC

1.2 Problem Statements

To achieve the desired output, most DAC designs depend on matched components, which is almost impossible to fabricate within a reasonable budget or timeframe. This causes mismatch error that will always exist, as well as discrepancies between designed and actual component specifications.

Other proposed DAC designs have utilized special design processes or laser trimming to minimize component mismatches, which unfortunately cannot be thoroughly eliminated.

Dynamic element matching (DEM) is a design technique that is capable of reducing the effects of component mismatches in electronic circuits. This technique dynamically rearranges the interconnections between components so that the effects of component mismatches can be minimized. Since the virtual positions of the components are randomized, harmonic distortions caused by mismatched components can be converted into white noise (Stubberud, 1998).

DEM also improves the dynamic performance such as spurious-free dynamic range (SFDR), as well as its static performance which is defined by differential nonlinearity (DNL) and integral nonlinearity (INL). (Guang Liang et al., 2012).

Two DEM techniques that have been used are the Full Randomization dynamic element matching (FRDEM) network and the binary tree network (BTN), with BTN having lower hardware complexity from two to six bits, but FRDEM having the lowest hardware complexity when implementing in seven bit or more. (Stubberud, 1998).

However, the drawback of implementing DEM is the resulting high hardware complexity, where complicated encoding is needed for conventional DEM encoders which will result in numerous switch transition and produces glitch to the output signal. (Teh, 2014).

Partial Binary Tree Network (PBTN) is a new DEM algorithm which possesses relatively lower hardware complexity. The design of such technique requires much fewer transmission gates compared to BTN, hence will have lower hardware complexity and reduced glitches.

While BTN implements random switching on all input bits of the DAC, 1-MSB PBTN only performs random switching on the most significant bit (MSB), while 2-MSB PBTN performs random switching on two of the MSBs. (Lim, 2016).

Previous researches had designed and validated up until 6-bit 1-MSB PBTN DEM DAC and 2-MSB PBTN DEM DAC. Both 1-MSB and 2-MSB achieved DNL error lower than 0.3632 LSB and INL error lower than 0.074 LSB. This leaves PBTN of higher bits to be designed and validated.

1.3 Objectives

- To implement current steering method to reduce the complexity of DACs.
- To improve the performance of 6-bit PBTN DEM DACs in terms of INL, DNL, power consumption, and reduce glitches.
- To design and simulate 8-bit PBTN DEM DAC in terms of INL, DNL, and power consumption.

1.4 Scope of Research

The aim of this project is to solve a problem commonly faced during the design of DACs, which is the mismatch errors caused by the difference between the designed and the fabricated component value. Hence this research implements DEM (Dynamic Element Matching) to address and solve the problem of component mismatch.

During the process of literature review, it was found that Binary Tree Network DEM DACs had decent performance, but with high hardware complexity. However, with the proposed PBTN algorithm, the DACs are expected to have a significant advantage in terms of low hardware complexity and lower power consumption.

To further improve on previous researches on this algorithm, this research will also look to increase the resolution from 6-bit to 8-bit. The current mirror as current source will also be modified so that the current sources can behave more ideally. Further improvements can be done to reduce the glitches on the output of the DACs.

In practice, power consumption is a critical design criteria for almost every electronic device and electrical systems. Lowering the power consumption decreases operating expenses, increases reliability, and allows compact design that permits more functionality to be packed into the same space, with less need for fans and other cooling equipment.

1.5 Thesis Outline

This thesis begins with the introduction in Chapter 1, where the background, problem statement, objective, and scope are presented.

In Chapter 2, literature review is presented, where various types of DAC architecture are explored and documented. 6-bit and 8-bit DACs are analyzed and compared in this chapter as well. DEM algorithms are researched and compared to determine the advantages and disadvantages of each design.

Methodology is discussed in Chapter 3, where the designs of 6-bit 1-MSB PBTN DEM DAC and 2-MSB PBTN DEM DAC are presented.

Results and discussion are discussed in Chapter 4. The simulation results of 1-MSB/2-MSB PBTN DEM DAC are illustrated, together with performance metrics DNL, INL, glitch impulse area, and power consumption.

Chapter 5 houses the concluding remarks of this research, and future work to further improve on the proposed method is discussed.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Digital-to-analog converters (DAC) act as interfaces in between analog and digital domains. However, DAC designs are prone to component mismatches, which gives rise to discrepancies between desired values and actual values. These errors are non-linear distortions in pulse shape, timing, and amplitude, all of which are common in high resolution DACs. (Teh, 2014).

This research will explore the implementations of DEM, with a newly proposed algorithm known as Partial Binary Tree Network (PBTN).

2.2 DAC Architectures

A DAC produces discrete and quantized analog signal when given a digital input. And based on the digital input received, this enables switches that combine the number of fractions accordingly to produce the output. If a digital input has N bits, hence there will be 2^N possible levels of output voltage or current, shown in Figure 2-1 and Table 2-1.

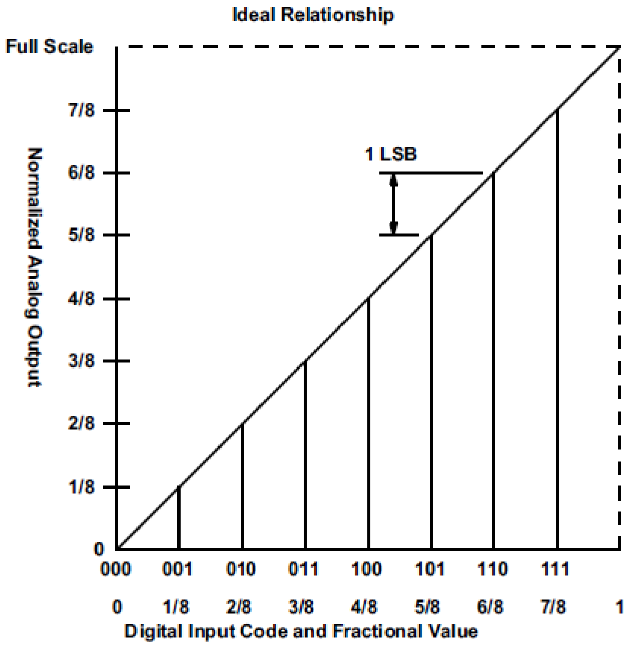

Figure 2-1 The ideal relationship (transfer function) between 3-bit digital input code and analog fractional value output

Table 2-1: A comparison of different types of digital input codes

Decimal	*Binary*	*Thermometer*	*Gray*	*2'Complement*
0	000	0000000	000	000
1	001	0000001	001	111
2	010	0000011	011	110
3	011	0000111	010	101
4	100	0001111	110	100
5	101	0011111	111	011
6	110	0111111	101	010
7	111	1111111	100	001

2.2.1 String DAC

Also known as the Kelvin divider, string DAC is the simplest DAC structure available, shown in Figure 2-2. It possesses 2^N resistors of equal resistance arranged in a series, where N is the number of input bits. Multiple switches are connected along a series of resistors. The digital input bits controls the opening and closing of the switchtes. The current from each all the closed switches are summed up to attain the output current. (Kester, 2009).

Advantages:

- Simple architecture, inherently monotonic.
- Higher glitch tolerance.
- Switching glitch not code-dependent, ideal for low distortion applications.

Disadvantages:

- Large number of resistors and switches will be required for higher resolutions.

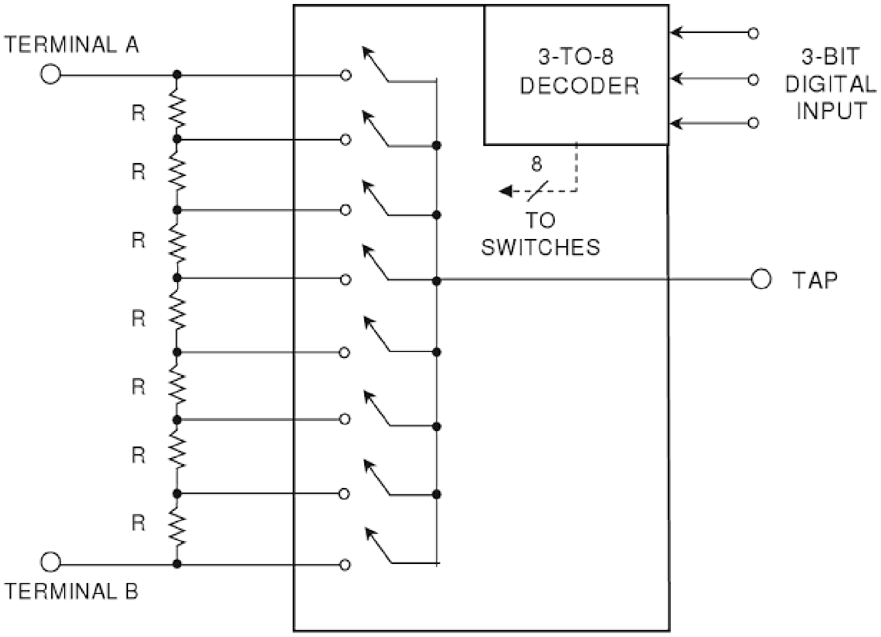

Figure 2-2 String DAC architecture

2.2.2 Multiplying DAC

Shown in Figure 2-3, Multiplying DACs or MDAC utilizes an external reference voltage as opposed to an multiplying the digital code with an internal voltage, which can only accept a narrow range of values. Conversely, external voltage can have more variation over a bigger range.

Advantages:

- Able to continue functioning even when reference voltages is reduced to zero.
- Has the potential to work with reference voltages which are significantly greater than their supply voltage. (Kester, 2009).

The output equation of MDAC is given as:

Analog output = VREF × Digital Input × Constant

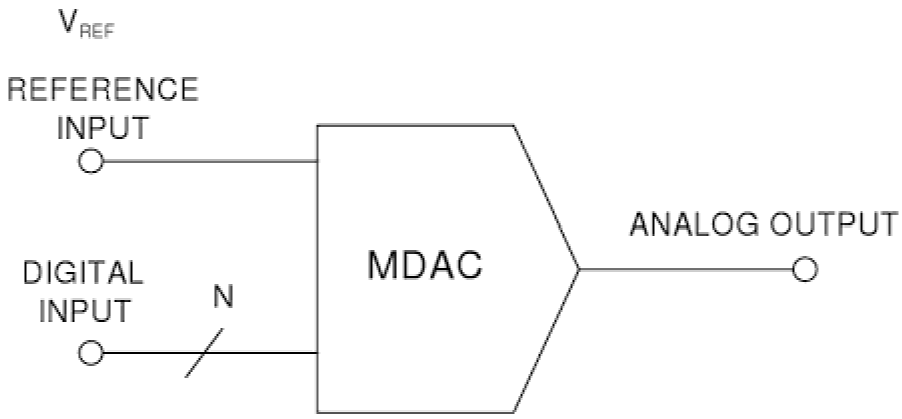

Figure 2-3 Multiplying DAC (MDAC)

2.2.3 Counting DAC

In this design shown in Figure 2-4, a sampling clock starts the counter, which is loaded with the digital code. This also sets an R/S flip-flop and the counter counts upwards at a fast rate. The R/S flip-flop will be reset when the counter reaches all 1’s.

The output produced by the R/S flip-flop is hence proportional to the complement of the digital input.

Advantage:

- Inherently monotonic.

Disadvantage:

- Resolution is sacrificed for update rate, since counter must cycle through all possible outputs during the sampling interval. (Kester, 2009).

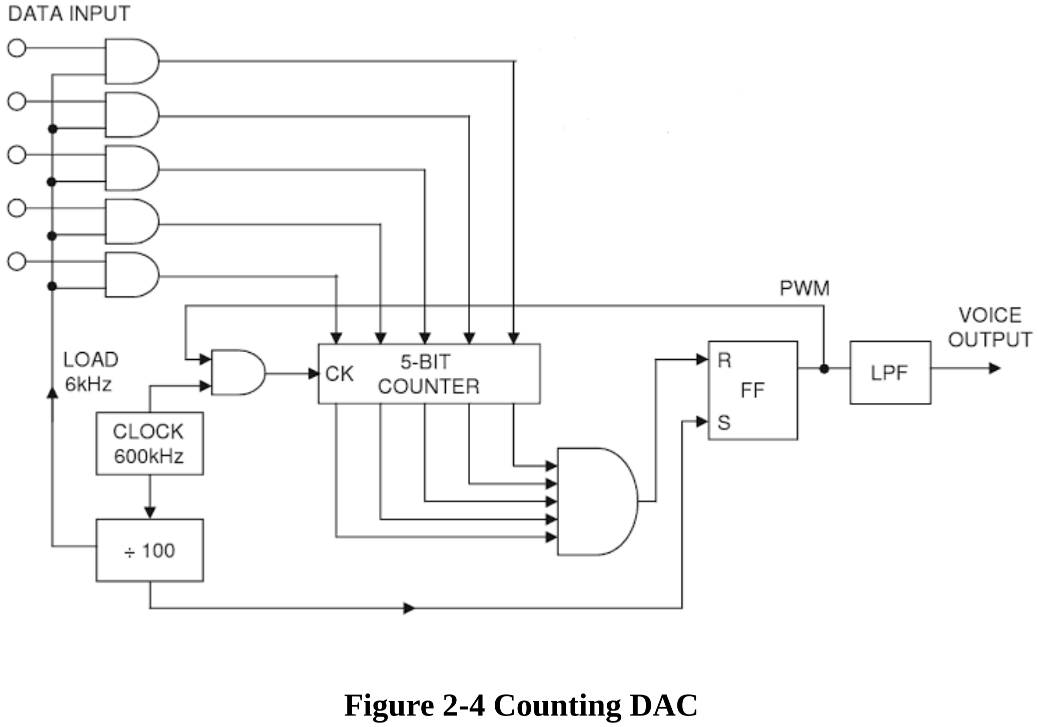

Figure 2-4 Counting DAC

2.2.4 Oversampling DAC

Also known as Delta-Sigma DAC, it consists of a digital interpolation filter, shown in Figure 2-5. The filter encodes a high resolution digital input signal into lower resolution.

Higher sampling frequency is mapped to voltages by inserting extra data points. The output is then smoothed with an analog filter. (Kester, 2009)

Advantages:

- Relatively simple circuitry.
- Higher efficiency.

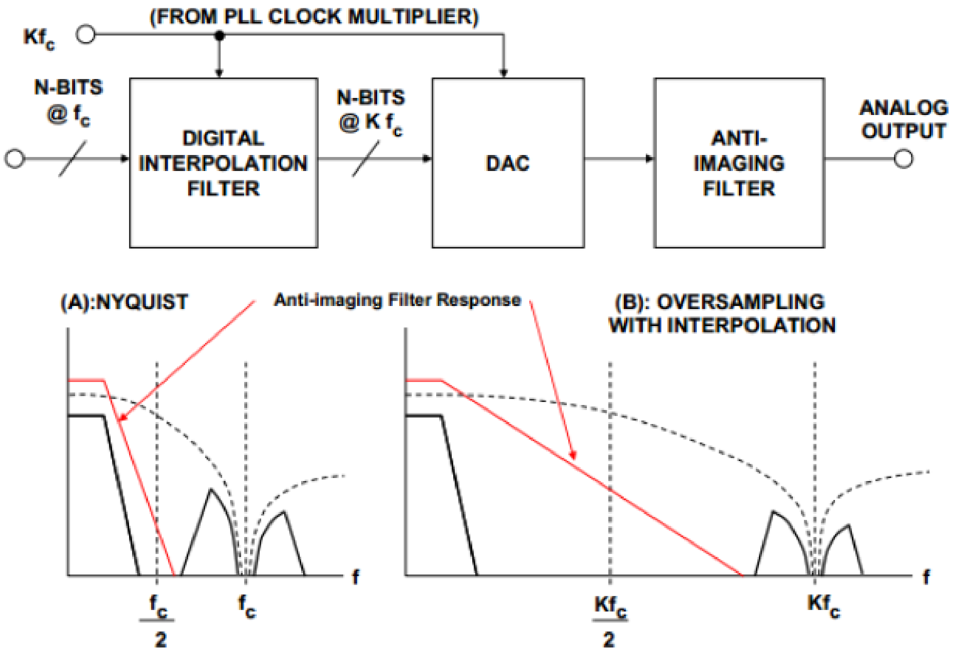

Figure 2-5 Oversampling DAC

2.2.5 Current Steering DAC

Shown in Figure 2-6, this DAC architecture is made up of multiple current sources each connected to a switch controlled by the digital code inputs. The output is obtained by summing the current from all then current sources.

Advantages:

- High performance in terms of INL and DNL
- Suitable for generating high frequency signals.

Disadvantage:

- High hardware complexity in higher resolutions. (Myderrizi and Zeki, 2010).

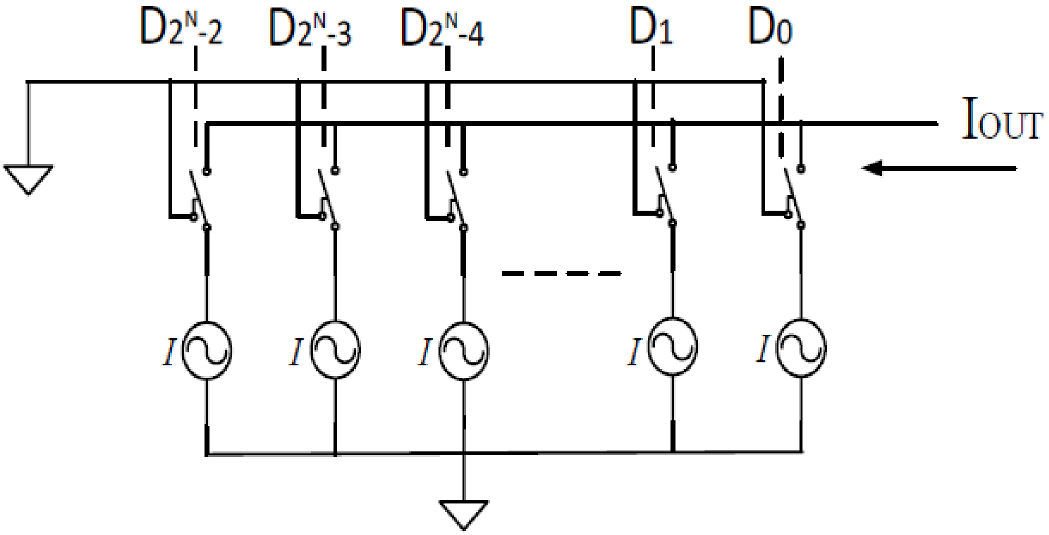

Figure 2-6 Current Steering DAC

2.3 Dynamic Element Matching (DEM) DAC Architecture

Shown in Figure 2-7, DEM DACs have enabled major performance improvements over the last decade in oversampling delta-sigma data converters, pipelined ADCs, and high resolution

Nyquist-rate DACs. In many cases, they effectively eliminate component mismatches as a performance limiting source of error.

In DACs without DEM, mismatches among nominally identical circuit elements inevitably introduced during circuit fabrication cause nonlinear distortion. By scrambling the usage pattern of the elements, DEM causes the error resulting from the mismatches to be pseudorandom noise that is uncorrelated with the input sequence instead of nonlinear distortion. In mismatch-scrambling DEM DACs, the noise is white, and in mismatch-shaping DEM DACs, the noise is spectrally shaped. The former is used in Nyquist-rate applications such as pipelined ADCs that require highly linear DACs. The latter are used in oversampling applications, i.e., applications such as delta-sigma (ΔΣ) data converters in which the DAC's input signal occupies a bandwidth of much less than half the sample rate. The idea is to shape the power spectral density (PSD) of the noise to lie mostly outside of the input signal band. (Galton, 2010).

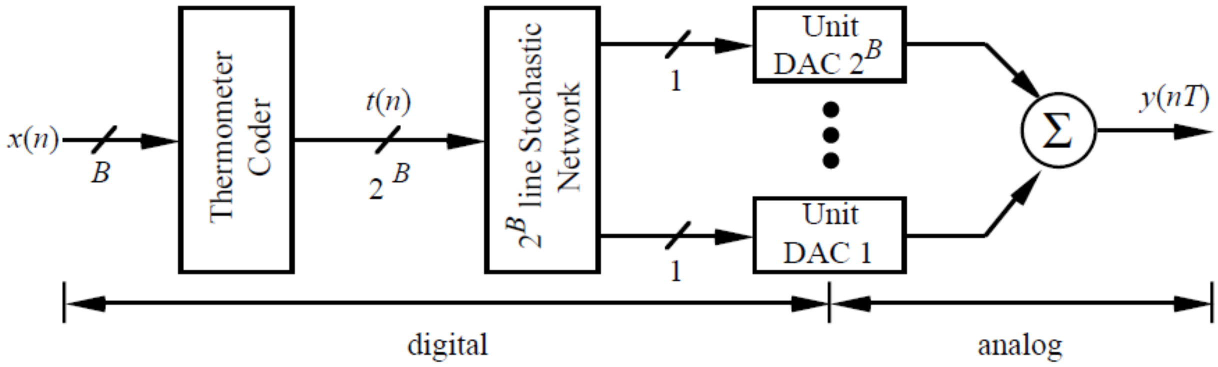

Figure 2-7 General topology of a B-bit stochastic DEM DAC

2.3.1 Full Randomization Dynamic Element Matching Network (FRDEM)

FRDEM networks are constructed using switching blocks as shown in Figure 2-8. An input signal of *k+1* bit is introduced into *b(n)*. The virtual position of all bits is determined by the control signal $C_{k-1}(n)$.

When the control signal is LOW, the network sends *k* LSBs to the lower outputs, whereas *k* copies of the MSB are sent to the upper output.

When the control signal is HIGH, the network sends *k* LSBs to the higher outputs, whereas *k* copies of the MSB are sent to the upper output. In other words, the case is reversed. (Stubberud, 1998).

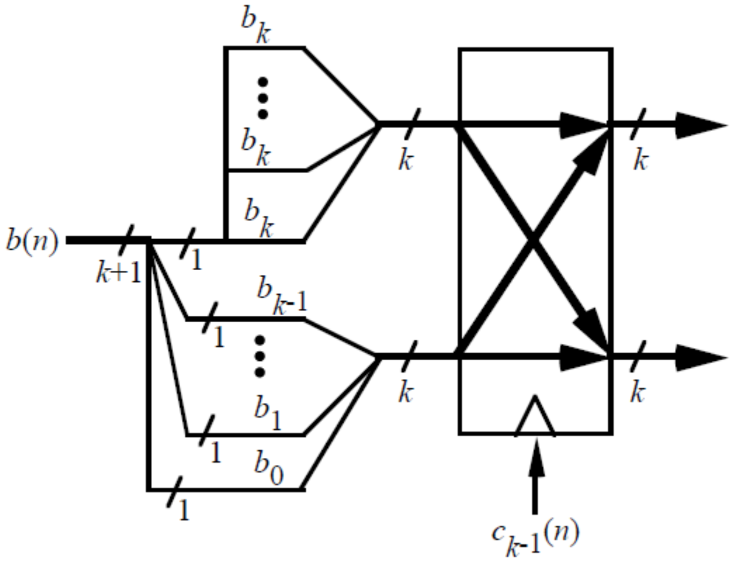

Figure 2-8 FRDEM switching block

2.3.2 Binary Tree Network

Binary tree network (BTN) is one of the implementation of Generalized Cube Network (GCN) on the operations of both the thermometer code and the 2^B line stochastic network.

A 1-bit BTN is shown in Figure 2-9. A k+1 bit BTN can be constructed using the topology as shown in Figure 2-10. (Stubberud, 1998).

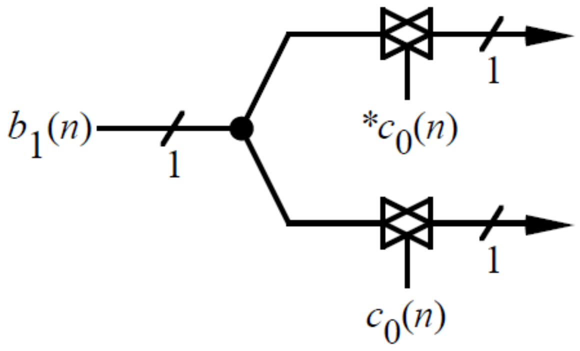

Figure 2-9 1-bit BTN

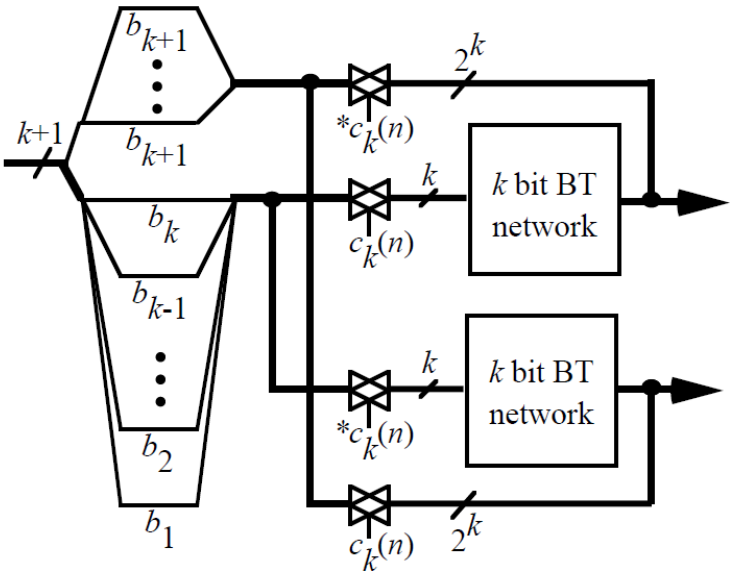

Figure 2-10 k+1 bit BTN

When the control signal *$C_k(n)$* is LOW, the network sends 2^k copies of the input MSB to the upper output, and *k* LSBs to the lower output.

When *$C_k(n)$* is HIGH, the network sends 2^k copies of the input MSB to the lower output, and *k* LSBs to the upper output.

2.4 Static Performance

2.4.1 Offset Error

Shown in Figure 2-11, offset error refer to the magnitude in which the output waveform shifts upwards or shifts downwards offset error describes how much the entire DAC transfer function is shifted up or down. (Duke, 2013).

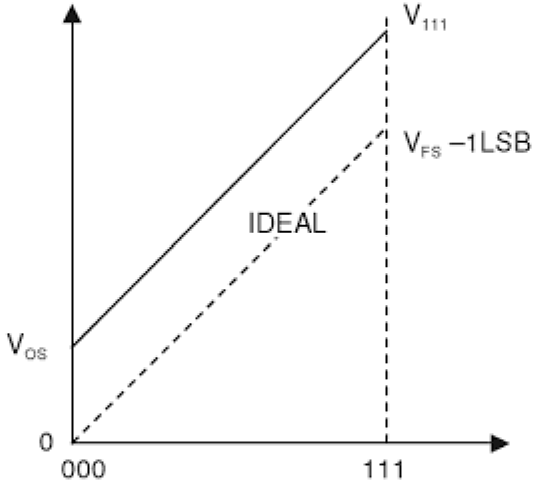

Figure 2-11 DAC offset error

2.4.2 Gain Error

Shown in Figure 2-12, gain error measures how much the actual output waveform diverges from the ideal waveform gain error compares how the real DAC transfer function’s slope relates to the ideal slope. (Duke, 2013).

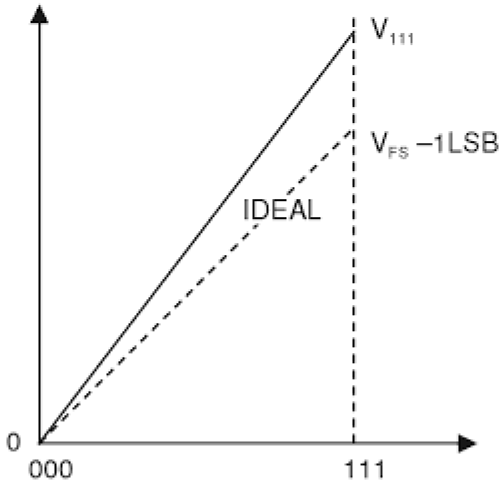

Figure 2-12 DAC gain error

2.4.3 DNL (Differential Non-Linearity) and INL (Integral Non-Linearity)

Shown in Figure 2-13, DNL measures the difference in step height of each step, with respect to the ideal step size. Shown in Figure 2-13, INL measures the deviation of each step from the ideal straight line. Both DNL and INL can be expressed in positive or negative values. Positive values show that the actual step is higher than the ideal value, whereas negative value shows otherwise. (Duke, 2013)

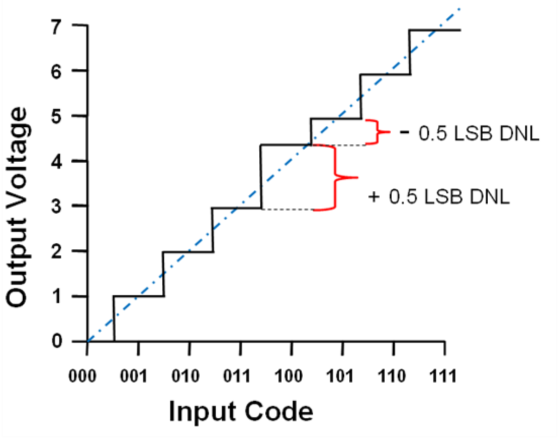

Figure 2-13 Differential nonlinearity (DNL)

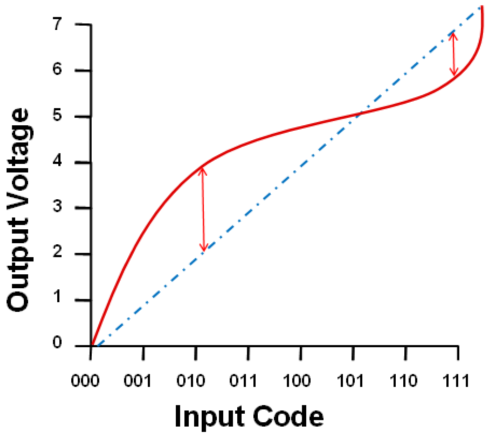

Figure 2-14 Integral nonlinearity (INL)

2.5 Dynamic Performance

2.5.1 Settling Time

Shown in Figure 2-15, whenever a new input code is given to the DAC, an initial spike can be observed at the output before output settles in a steady state, this is also known as a transient state. One valid method to define the setting time is the time taken for the output to exit the error band. (Kester, 2009). Ideally, the settling time should be minimized to lower the distortions in the output waveform.

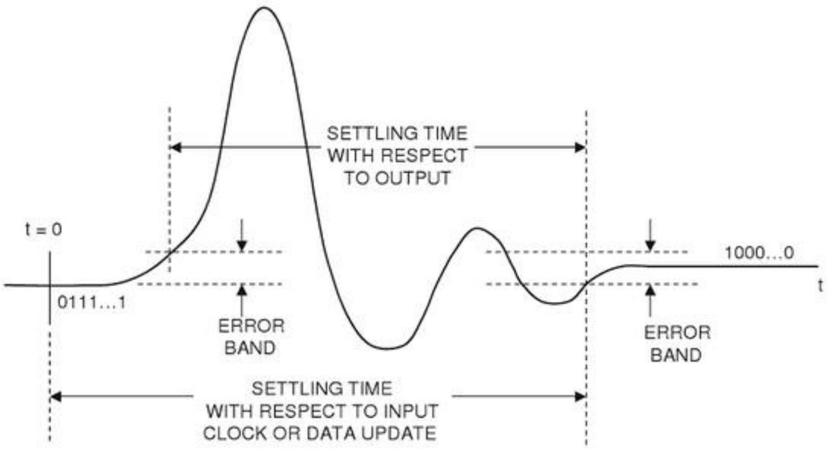

Figure 2-15 Setting time

2.5.2 Glitch

Shown in Figure 2-16, a glitch is when the output of a DAC overshoots, undershoots, or both, relative to the desired value and usually occurs when the input changes from one code to another. Glitch is cause by signal-dependent error injected from the digital inputs to the analog output. This problem is caused by the difference in switching time for different bits, and most noticeable when several bits change at once. (Myderrizi & Zeki, 2010).

For example, when an input bit changes instantly from 011111 to 100000, the most significant bit might undergo change faster than all the other bits, this results in the output appearing to be 111111 for a very brief amount of time. Such cases are predominant when the MSB has a different value than all the other bits. (Nesboe, 2010).

Glitch is characterized by measuring the glitch impulse area, where positive and negative pulses are added and cancel each other out, and is approximated by considering the first four pulses of glitch only.

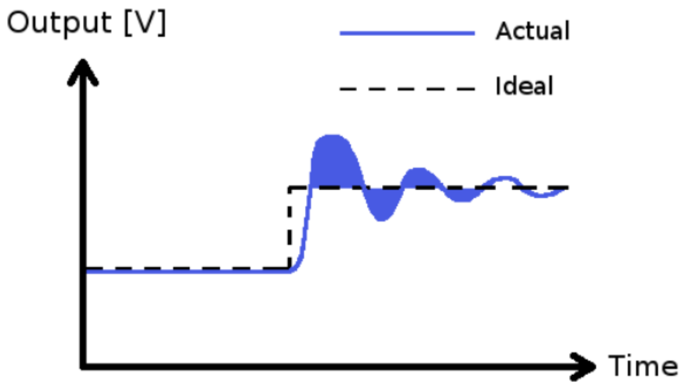

Figure 2-16 Actual output waveform showing glitch pulses, as compared with the ideal output waveform

2.5.3 Spurious Free Dynamic Range (SFDR)

Shown in Figure 2-17, SFDR defines the usable dynamic range of a DAC output before factoring in spurious noise that causes interference and/or distortion to the fundamental signal. SFDR is the measure of the difference in amplitude between the fundamental and largest harmonically or non-harmonically related spur from direct current to the full Nyquist Bandwidth. (Garcia, 1995) The equation to calculate SFDR is given by:

SFDR = RMS Fundamental Signal – Amplitude of Largest Spur

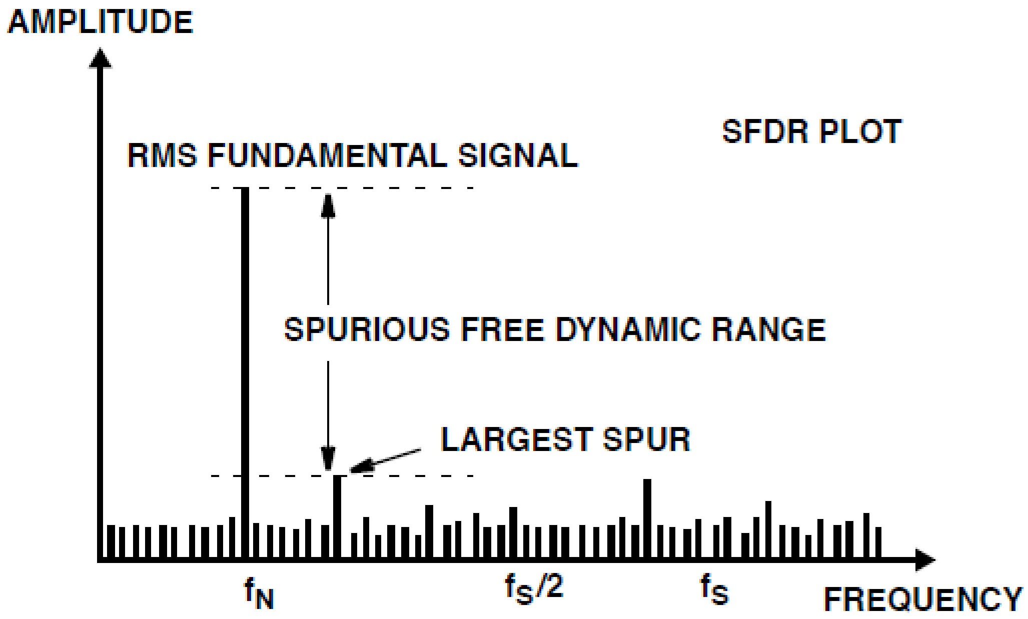

Figure 2-17 DAC Spurious Free Dynamic Range (SFDR)

2.5.4 Methods to improve SFDR

SFDR is directly related to the linearity performance of the DAC, in addition to glitch performance. However, eliminating glitch is a challenging task, which involves switching to a more simplified DAC design. Methods to eliminate SFDR include:

1) Reduce data rate
2) Reduce clock edge rate
3) Bypassing DAC
4) Implementing parallel capacitors for decoupling and noise reduction
5) Implementing surface mount components to minimize lead inductance and stray capacitance. (Garcia, 1995).

2.6 Summary

Different architectures of DACs are explored and discussed in this chapter comparing their advantages and disadvantages.

Two algorithms of dynamic element matching (DEM) where discussed as well, namely full random dynamic element matching (FRDEM) and binary tree network (BTN).

Several metrics of analyzing the performance of DACs are presented, where static performance includes integral nonlinearity (INL), differential nonlinearity (DNL), offset error and gain error. Dynamic performance included settling time, glitch, and spurious free dynamic range (SFDR).

CHAPTER 3

METHODOLOGY

3.1 Introduction

The goal of this research is to further improve on the performance of the PBTN DEM DAC and to increase the resolution from 6-bit to 8-bit.

Research on relevant and related topics is required before the designing phase of this research and is achieved by studying research papers. Current DAC topology and specifications will be researched to determine the best configuration and design to proceed with.

Next, mastering the use of the simulation software Cadence Virtuoso is done by studying tutorials manuals and videos. After which the circuit design will be performed using Cadence, where different DAC architecture will be drawn and tested. Simulation will be performed after the circuit is designed.

The process is illustrated in a flowchart in Figure 3-1.