DC Voltage Generation Using Periodic Bit-Stream Modulation

by

Sébastien Laberge, B. Eng. 1999

Department of Electrical and Computer Engineering

McGill University, Montréal



December 21st 2001

A thesis submitted to the Faculty of Graduate Studies and Research in partial fulfillment of the requirements for the degree of Master of Engineering

© Sébastien Laberge, 2001



National Library of Canada

Acquisitions and Bibliographic Services

395 Wellington Street Ottawa ON K1A 0N4 Canada Bibliothèque nationale du Canada

Acquisitions et services bibliographiques

395, rue Wellington Ottawa ON K1A 0N4 Canada

Your Me. Votre réference

Our Be Note relevance

The author has granted a nonexclusive licence allowing the National Library of Canada to reproduce, loan, distribute or sell copies of this thesis in microform, paper or electronic formats.

The author retains ownership of the copyright in this thesis. Neither the thesis nor substantial extracts from it may be printed or otherwise reproduced without the author's permission. L'auteur a accordé une licence non exclusive permettant à la Bibliothèque nationale du Canada de reproduire, prêter, distribuer ou vendre des copies de cette thèse sous la forme de microfiche/film, de reproduction sur papier ou sur format électronique.

L'auteur conserve la propriété du droit d'auteur qui protège cette thèse. Ni la thèse ni des extraits substantiels de celle-ci ne doivent être imprimés ou autrement reproduits sans son autorisation.

0-612-79079-7

Canadä

Acknowledgments

This work would not have been possible without the encouragement and support of many. First and Foremost, I would like to thank my supervisor, Professor Gordon W. Roberts, for his understanding, encouragement, and constant support over the past two years.

I would also like to highlight the technical contributions of Mohamed Hafed, Arshan Aga, and Naveen Chandra. Each of them have contributed to different areas of this thesis by sharing their work and knowledge as well as providing me valuable feedback based on the use of my work in their own applications

I would also like to underline the help and support of the many students of the MACS lab such as Antonio, Bardia, Boris, Christian, Clarence, Ian, Lige, Mona, Mourad, Naveen, and many others which I can't name all.

I am also grateful to my family and many close friends for their continuous support and encouragement throughout these many years spent earning a degree a McGill.

This work was supported by "Le Fonds pour la Formation de Chercheurs et l'Aide a la Recherche", the Canadian Microelectronics Corporation, and Micronet, a Canadian network of centres of excellence dealing with microelectronic devices, circuits, and systems.

Abstract

In recent years, the trend for continuous down-scaling of CMOS device dimension has made analog design much more challenging. This trend has been a major driving force in trying to find new approaches for designing common analog building blocks. One such block is the bandgap voltage reference. This common circuit serves the purpose of generating a fixed DC voltage reference and has uses in a wide variety of applications.

This thesis introduces a new way of generating a programmable DC voltage reference with similar performance to the traditional means. This voltage reference generator is based on periodic bit-stream modulation and relies on simple digital logic combined with a low pass filter (LPF) to demodulate the DC reference level. The advantage of the proposed DC voltage reference lies in its immunity to technology scaling as it is mostly digital. The programmability of the proposed circuit also makes it usable as a digital to analog converter (DAC).

Through simulation and experimental results obtained using a set of integrated circuits implemented in 0.35 μ m, 0.25 μ m and 0.18 μ m CMOS technologies a number of conclusions are reached. The tradeoffs between the two different bit-stream modulation scheme, pulse width modulation (PWM) and pulse density modulation (PDM), are compared yielding PDM as the best approach. The analysis and simulation of a new synthesis method will demonstrate that high-order passive RC filters yield the most attractive realization of the LPF. Experimental results will also demonstrate that

performance due to temperature variations comparable to bandgap references can be achieved. A set of experiments will also demonstrate the excellent performance of this voltage reference when used as a DAC. Lastly, the use of asynchronous logic for generating periodic bit-streams will be shown to yield promising results.

Résumé

Ces dernières années, la miniaturisation des composantes CMOS a rendu la conception de circuits analogique beaucoup plus difficile. Cette tendance a été la motivation principale derrière la recherche visant à trouver de nouvelles méthodes pour générer ces circuits analogiques. Un exemple d'un tel circuit est la référence de voltage 'bandgap'. Ce circuit est commun dans beaucoup d'application et a pour fonction de générer un voltage DC fixe.

Ce mémoire présente une nouvelle façon de générer une référence de voltage qui est programmable et donne d'aussi bons résultats que les méthodes traditionnelles. Cette référence de voltage est basé sur le concept de 'periodic bit-stream modulation'. Elle fait usage de circuits digitaux pour la majeure partie et d'un filtre analogique affin de démoduler le signal DC. Les principaux avantages de cette référence de voltage sont son immunité aux effets de miniaturisation et sa composition majoritairement digitale. De plus, cette référence de voltage peut être utiliser comme convertisseur digital/analogique à cause de sa programmabilité.

Par le biais d'expériences et de simulation effectuer sur des circuits fabrique dans une technologie CMOS, 0.35, 0.25 et 0.18, un certain nombre de conclusion seront tirée. La comparaison des deux types de modulation, 'pulse width modulation (PWM)' et 'pulse density modulation (PDM)', démontre PDM comme étant la meilleure méthode. L'analyse et la simulation d'une nouvelle méthode de synthèse pour les filtres, démontre qu'un filtre de type 'high-order passive RC' est la meilleure façon de démoduler un signal DC. Les résultats obtenus donnent des résultats comparables a une référence de voltage 'bandgap' quant à la dépendance aux variations de température. D'excellent résultats sont aussi obtenus lors de l'utilisation de cette référence de voltage comme convertisseur digital/analogique. Finalement, une architecture non synchronisée (assynchronous) pour générer le 'periodic bit-stream' de cette référence de voltage démontre des résultats très prometteurs.

Table of Contents

Chapter 1 - Introduction 1
1.1 - Motivation1
1.2 - Thesis Outline
Chapter 2 - Bitstream Theory 5
2.1 - General Theory
2.2 - Pulse Width Modulation10
2.3 - Pulse Density Modulation
2.4 - High-Order Noise Shaping
2.5 - Summary
Chapter 3 - Periodic Bit-Stream Filtering. 24
3.1 - First-Order Filtering
3.2 - High-Order Filtering
3.2.1 -Passive
3.2.2 -Active
3.3 - Semi-Digital Filtering40
3.4 - Summary
Chapter 4 - DC Voltage Generator Implementation44
4.1 - Bit-Stream Generator Implementation Overview
4.1.1 -Synchronous Scan Chain44
4.1.2 -Asynchronous Scan Chain45
4.1.3 -Memory Based Design
4.1.4 -Hardware Modulator47
4.1.5 -Automatic Test Equipment (ATE)47
4.2 - Voltage Generator Design
4.2.1 -Synchronous Voltage Generator48

4.2.2 - Asynchronous Voltage Generator	53
4.3 - Summary	64
Chapter 5 - Experimental Results	66
5.1 - Testing Methodology	
5.2 - Voltage Generator Experimental Results	68
5.2.1 -Synchronous Voltage Generator	68
5.2.2 -Asynchronous Voltage Generator	73
5.3 - Summary	78
Chapter 6 - Conclusion	80
6.1 - Summary and Discussion	80
6.2 - Future Work	81
References	83
Appendix A -Logical Effort Based Optimization	86
Appendix B - Temperature Performance	92

List of Figures

Chapter 1 -	Introduction	1
Chapter 2 -	Bitstream Theory	5
Figure 2.1	On chip voltage generator	5
Figure 2.2	Bit Stream Frequency Spectrum	6
Figure 2.3	Transient Behaviour for Different Time Constants	8
Figure 2.4	Frequency spectrum of PDM and PWM bit stream	9
Figure 2.5	Pulse Width Modulation	10
Figure 2.6	Pulse With Modulator	10
Figure 2.7	Frequency spectrum for different pulse width	11
Figure 2.8	Magnitude of the first harmonic	12
Figure 2.9	Periodic Square Wave	13
Figure 2.10	Modulation of a 0.5 level with PDM and PWM	14
Figure 2.11	Block Diagram of First Order Sigma-Delta Modulator	15
Figure 2.12	Frequency spectrum of PDM for 1/10th resolution	17
Figure 2.13	Frequency Spectrum for Different N1	18
Figure 2.14	Block Diagram for Arbitrary Order Sigma Delta Modulator.	19
Figure 2.15	High Order Modulation Frequency Response	20
Figure 2.16	Bit Selection Process	21
Chapter 3 -	Periodic Bit-Stream Filtering.	24
Figure 3.1	Filter Transient Response	
Figure 3.2	Low Pass M/2-th Order RC Filter	29
Figure 3.3	SAB Biquad stage	39
Figure 3.4	Semi-digital FIR reconstruction filter	40
Figure 3.5	Current-mode topology of an N-th order semi-digital FIR rec filter.41	construction
Figure 3.6	Novel Current-mode topology	42

Figure 3.7	Current mirroring output stage for realizing one of the curre nodes43	nt summing
Chapter 4 -	DC Voltage Generator Implementation	44
Figure 4.1	Bit Stream Generator	45
Figure 4.2	ATE Based Bit-stream Generator	47
Figure 4.3	IC Implementation Overview	49
Figure 4.4	Pulse Width Modulator	49
Figure 4.5	8-Bit Counter	50
Figure 4.6	Comparator	51
Figure 4.7	PDM Bit-stream Generator	52
Figure 4.8	Chip Micrograph of PWM/PDM Generator	53
Figure 4.9	Bi-mode scan chain architecture	54
Figure 4.10	Data Latch	55
Figure 4.11	Novel Asynchronous Cell	56
Figure 4.12	Chip Micrograph of Bi-Mode Scan-Chain	57
Figure 4.13	Signal Flow Graph of 2nd Order Multi-bit Modulator	58
Figure 4.14	3 bit DAC	59
Figure 4.15	4th Order Low-Pass SAB Filter	60
Figure 4.16	Layout of 2nd Order Multi-bit Modulator	61
Figure 4.17	Programmable Reference Implementation Overview	62
Figure 4.18	Asynchronous Cell	63
Figure 4.19	Layout of Asynchronous Reference	64
Chapter 5 -	Experimental Results	66
Figure 5.1	Output Voltage Using a Coarse External Filter for (a) PDM C PWM Generator with ~55 % duty cycle, and (c) PWM Gene ~25 % duty cycle.69	Generator, (b) erator with
Figure 5.2	Settling Behaviour of the On-Chip (a) PDM and (b) PWM C	Generators.70
Figure 5.3	Outputs of the (o) PDM and (+) PWM Generators of the Pro-	ototype IC.70
Figure 5.4	Single Ended Voltage Reference	71
Figure 5.5	Differential Voltage Reference	72
Figure 5.6	Frequency Compensated Voltage Reference	73
Figure 5.7	Measured Results with 0.085V Amplitude 1957.0315 Hz Siz	newave75
Figure 5.8	SNR/SNDR vs. Input Power	75
Figure 5.9	(a) INL and (b) DNL for Single Ended Voltage Measuremen	its78
Figure 5.10	(a) INL and (b) DNL for Differential Voltage Measurements	
Chapter 6 -	Conclusion	80
Logical Effort	Based Optimization 86	
Figure A.1	Asynchronous Control Cell Optimization	88

Figure A.2	PMOS sweep	90
Figure A.3	Data Latch Optimization	91
Temperature F	Performance 92	
Figure B.1	Area Under the Curve w/r to Propagation Delay	92
Figure B.2	Voltage versus Temperature for Different Buffer Sizing .	93

Chapter 1 - Introduction

1.1 - Motivation

Recent years have seen a tremendous increase in semiconductor packing density due to the continuous down scaling of CMOS device dimensions. This down scaling has also meant higher circuit speeds and lower power dissipation, all of which are major driving forces for the microelectronics and computer industries. However, while such scaling effects are generally desirable for digital circuits, they tend to make analog and mixed signal design more challenging [1]. For example, voltage scaling is necessary for maintaining reasonable electric fields and power consumption levels in digital circuits, but it can significantly hinder the performance of analog circuits. This being said, the ever increasing demand for high-speed communication devices means that mixed-signal and analog blocks are very likely to be present in future integrated circuits. Consequently, analog designers are going to have to cope with the challenges introduced by scaling, and this could entail getting rid of old design techniques and utilizing new ones.

A particular analog circuit which is being affected by this down scaling is the voltage reference. Voltage references are an essential analog building block and find uses in a wide variety of applications. In particular, they can be used as DC bias sources, which are needed in virtually all analog circuits, e.g., operational amplifiers, charge pumps, linear regulators, Digital to Analog Converters (DAC), Analog to Digital Converters (ADC) and delay elements.

The traditional method for generating such on-chip voltages is using a bandgap reference circuit [2][3]. An approach is to use a forward-biased diode voltage and combine it with a voltage that is proportional to absolute temperature (PTAT). This is covered in greater details in [3]. This approach will yield a reference voltage which is less temperature dependent than using just the single diode. This designs will be affected by power supply scaling because of the high voltage require to forward-bias a diode, i.e., these well proven circuits will stop working once the voltage supplies drop below a certain level [4]. Also, in order to implement the base-emitter junction a specialized device (substrate vertical pnp BJT) has to be made available in the CMOS process. Lastly, this method only yields a fixed voltage reference level, thus will require additional circuitry when multiple reference voltage levels are necessary.

This thesis presents a new technique for constructing a DC voltage reference circuit. It extends the use of periodic bit-stream modulation, which has been mostly used for the purpose of generating analog AC waveforms [5][6], to the generation of DC levels. The proposed technique makes use of mostly digital logic with the exception of a low-pass filter, which is used to demodulate the DC voltage from the periodic bit-stream. This technique is expected to benefit greatly, rather than suffer, from the down-scaling of digital integrated circuit technology. Specifically, it will not suffer from power supply scaling and, since it uses an almost all-digital implementation, its speed and area are expected to benefit greatly from scaling. However, unlike a bandgap reference this voltage reference will be dependent on the power supply from which it is derived. Lastly, this new voltage reference can be made programmable, thus enabling it to be used as a Digital-to-Analog Converter (DAC). The work presented in this thesis has been submitted for a U.S. patent [7].

There are three main issues which will be addressed in this work. First, the type of periodic bit-stream modulation scheme (PWM versus PDM) that is better suited for generating the DC signal, together with the desirable characteristics of these modulated bit-streams are discussed. Next, we shall consider practical methods in which to extract the DC signal from these bit-streams, i.e., low pass filtering techniques. In particular, we

shall consider what time and frequency characteristics these low pass filters should posses. In addition, we shall also consider whether a passive or active realization is better suited for this application. Finally, the bit-stream generator implementation issues will be addressed, such as the tradeoffs between synchronous and asynchronous designs. The next section will outline the context of each chapter and the major issues that will be addressed.

1.2 - Thesis Outline

In Chapter 2 the general operating principles for the bit-stream DC voltage generator of this thesis will be presented. The basic theory behind periodic bit-stream modulation will be explained. A detailed description of Pulse Width Modulation (PWM), Pulse Density Modulation (PDM), as well as higher order modulation schemes will be given. The tradeoffs of each and the impact they will have on the filtering portion of the voltage generator will also be discussed.

Chapter 3 will introduce filtering methods for extracting the DC level from a bitstream. Different types of filter implementations such as passive and active realizations will be considered. The synthesis methods required to implement these filters as well as the tradeoffs between the different types will be discussed.

Chapter 4 will give an overview of different ways to implement a bit-stream generator in a standard digital CMOS process. A detailed description of the different designs which were implemented over the course of this thesis will be provided.

The experiments performed for this thesis will be explained and the corresponding results provided in Chapter 5. The goal of these experiments will be to conclude about some of the main issues which motivates this thesis. Those issues being the bit-stream modulation scheme and the implementation type. The experiments will be performed on four different integrated circuits which were implemented over the course of this thesis: two in 0.35 μ m CMOS, one in 0.25 μ m CMOS and one in 0.18 μ m CMOS technology.

Finally, a summary of this thesis as well as some discussions of the results will be presented in Chapter 6. More specifically, the main design issues highlighted in this

introduction will be discussed based on the experimental results obtained. These discussions will yield a set of conclusions and also open up some possibilities for future work.

Chapter 2 - Bitstream Theory

In this chapter, the basic principles behind the bit-stream DC voltage generator of this thesis will be presented. First, the general theory of bitstream modulation will be presented. Then more specific modulation scheme along with their filter requirements will be outlined.

2.1 - General Theory

In the following discussions, binary voltage levels will be referred as ones and zeros, and analog DC voltages as values between zero and one. Hence all the voltage values are always normalized.

The basic principles behind this voltage generator are the use of digital pulse modulation to encode a DC level in the average value of a periodic digital sequence and the use of a low pass filter to extract the DC average. In Figure 2.1 the two main blocks of this voltage generator are shown. They are a bit-stream generator and a low pass filter (LPF).



Figure 2.1 On chip voltage generator

Figure 2.2 shows the frequency spectrum of a typical periodic bit-stream before low pass filtering. It consists of DC and AC components. The magnitude of the DC component is determined by the average value of the bit-stream, N_1/N_b where N_1 is the number of ones in the bit-stream and N_b the period in number of bits. The DC value will remain the same regardless of the ordering of the bits in the sequence. The AC components are harmonically related to the fundamental frequency F_f of the bit-stream sampling frequency F_s , and the length of the scan chain N_b , according to

$$F_f = \frac{F_s}{N_b}.$$
(2.1)

The magnitude of these harmonics will change depending on the order in which the bits are arranged in the periodic bit-stream.

The first design variable that must be chosen is the length of the bitstream, N_b. One can show that the normalized DC resolution ΔDC , the minimum step size, is given by

$$\Delta DC = \frac{1}{N_b}.$$
(2.2)

For illustrative purposes, if 256 levels are desired between zero and one, then the bitstream period N_b will have to be 256 bits long. Next, one must consider the effects of the AC components on these DC levels. Clearly, it is these AC components that give rise to the fast transitions in the bit-stream. It is the objective of the lowpass filter to reduce these



Figure 2.2 Bit Stream Frequency Spectrum

variations to insignificant levels. It is common practice to refer to the superposition of all AC components in the filtered signal as an AC ripple. Typically the maximum amplitude of the AC ripple is set to be less than 1/2 times the normalized DC resolution, i.e.

$$ACripple \le \frac{1}{2} \times \Delta DC = \frac{1}{2} \times \frac{1}{N_b}.$$
(2.3)

For the example when N_b is 256, the maximum allowable AC ripple would be 1/512.

The next decision that must be made is the order in which the ones and zeros are arranged in the periodic bit-stream. As was stated earlier, the desired DC value will determine the number of ones (N_1) and zeros (N_b-N_1) in the bit-stream. The order in which they are arranged will affect the distribution of power in the AC components. The optimal manner in which to select the bit pattern is to distribute the ones in such a way that it creates the maximum number of repetitions in the bit stream. In other words, minimize the period of the fundamental. By doing so, the frequency of the fundamental tone will be placed higher in frequency, as well as its harmonics. This in turn will reduce the requirements on the low pass filter's bandwidth.

The reason why this is a desirable property is explained by the following. In order to obtain a DC level with the desired amplitude resolution, a filter must be designed in order to sufficiently attenuate the AC components. The attenuation created by a first-order low-pass RC filter is described by the following equation.

$$|H(f)| = \frac{1}{\sqrt{1 + (2\pi f R C)^2}}$$
(2.4)

This filter attenuation may be characterized in terms of its cutoff frequency (3 dB bandwidth), $1/2\pi$ RC, or alternatively its RC time constant. This design choice will affect two main parameters in the design: the AC ripple and the settling time of the filter.

By increasing the RC time constant, the bandwidth is decreased. This has the effect of attenuating more of the harmonics, thus reducing the AC ripple. However, a longer settling time is required to reach the desired DC level due to the charging process

associated with the larger capacitor. For a step input of 1 V, the output voltage of the RC filter as a function of time is clearly dependent on the value of the RC time constant given by

$$V(t) \approx 1 - e^{-t/(RC)}$$
 (2.5)

The larger the RC time constant, the longer it takes the filter to reach a particular output level. This is illustrated in Figure 2.3 for two different RC time constants. The sizes of the input step were intentionally made different so that the settling behaviour is made clearly visible. Another impact of the RC time constant, will be the implementation area of the filter. The larger the value of RC, the larger the resistor and capacitor. Hence the motivation for having a bitstream modulation scheme where most of the harmonic power is higher in frequency.

As an example, if four ones are spread evenly throughout a 256 bit periodic bitstream, the fundamental period will be 256/4, or 64. Thus, the harmonic frequencies will appear at multiples of $F_s/64$. If they are placed in sequence, four ones in a row



Figure 2.3 Transient Behaviour for Different Time Constants

followed by 254 zeros, then the fundamental period is 256, and the harmonic frequencies will appear at multiples of $F_s/256$. The latter case would be considered the worst-case whereas the first case is the best case as the frequency of the fundamental is placed at the highest possible frequency. Figure 2.4 illustrates the frequency spectrum for these two cases.

In the previous example the two cases are effectively pulse width modulation (PWM) and pulse density modulation (PDM). In the case of PWM, the DC value is encoded in the width of the pulse which is set by N_1 consecutive ones. For PDM, the ones are distributed evenly throughout the period N_b , and the DC value is encoded in the density of the ones and zeros. In this example, the RC time constant for the PDM case can be made 64 times smaller than that called for in the PWM case for the same amount of AC ripple. So the PDM bit-stream will have a faster settling time than the PWM bit-stream. The following two sections will discuss further PWM and PDM along with their associated filter design requirements.



Figure 2.4 Frequency spectrum of PDM and PWM bit stream

2.2 - Pulse Width Modulation

As was just stated, for PWM, the DC value is set by the ratio of the time the bitstream is high to the total time of one period. For this kind of modulation this ratio is varied by changing the width, or duty cycle, of a pulse of period N_b . For example, to encode a voltage of 2/5, the pulse will be high for two bits and low for three bits. This example assumes N_b is 5. Figure 2.5 illustrates an example where N_b is 5 bits and T_s represents the sampling period of the bitstream. Next to each pulse width is the normalized voltage that would be obtained from filtering a periodic repetition of the pulses. One possible way to obtain such a bitstream is depicted in Figure 2.6 for a 256 bit period. For



Figure 2.5 Pulse Width Modulation



that example the output of the comparator, which is high when the count is smaller than the reference, yields the appropriate bitstream.

Figure 2.7 shows the frequency spectra of the bitstreams illustrated in Figure 2.5. The x-axis is the normalized bitstream frequency F/F_s and the y-axis is the normalized voltage level. It is seen that the DC tones have the appropriate magnitude. Note that the first harmonic is always dominant albeit, the 2nd may be of the same magnitude. Using a low pass filter, the frequency harmonics can be attenuated sufficiently to obtain the desired ac ripple on the output DC voltage.

When designing the filter all the DC levels must be looked at in order to determine the pulse width which yields the first harmonic with the highest magnitude. In Figure 2.8, the power of the first harmonic is plotted for 256 different DC levels. The code which yields the highest magnitude of the 1st harmonic is 128 bits, which corresponds to a DC voltage of 128/256 or 1/2. This will always occur for this type of modulation as will be demonstrated shortly. Thus it is always safe to design the filter for this worst case scenario.



Figure 2.7 Frequency spectrum for different pulse width



Figure 2.8 Magnitude of the first harmonic

A periodic sample set consisting of N samples can be represented by a combination of N complex exponentials having frequencies $2\pi k/N$ given by the following

$$x[n] = \sum_{k = \langle N \rangle} a_k e^{jk((2\pi)/N)n} \text{ for } k = 0, 1, 2, ..., N-1$$
(2.6)

where the spectral coefficients ak are obtained from

$$a_{k} = \frac{1}{N} \sum_{n = \langle N \rangle} x[n] e^{-jk((2\pi)/N)n} \text{ for } k = 0, 1, 2, ..., N-1.$$
(2.7)

The term corresponding to k = 0 represents a constant function, thus the DC component. Also, the term k = 1 is the first harmonic of x[n], the lowest AC frequency component. This corresponds to the fundamental period of the bit stream. The spectral coefficients for the sampled square wave illustrated in Figure 2.9 can be found to be the following:



Figure 2.9 Periodic Square Wave

$$|a_{k}| = \begin{cases} \frac{1}{N} \left| \frac{\sin[(2\pi k(N_{i}+1))/(2N)]}{\sin[(2\pi k)/(2N)]} \right| & \text{For } k \neq 0, N, 2N, ... \\ \frac{N_{i}+1}{N} & \text{For } k = 0, N, 2N, ... \end{cases}$$
(2.8)

where $N_i = N_1 - 1$ is used to compact the notation.

For PWM with a periodic bitstream of period N_b , the harmonics will be positioned at the following frequency locations:

$$H_k = \frac{2\pi k F_s}{N_b}$$
 for k = 1 to N_b-1, (2.9)

as $N = N_b$. This will be true for every pulse width thus all DC levels. It is seen from Eqn. (2.8) that most of the power is in the first harmonic, so the filter has to be designed for the appropriate attenuation at F_s/N_b . Moreover looking at Eqn. (2.8) one sees that the maximum for a_1 occur when $N_1 = N_b/2$, which is equivalent to a voltage of 1/2 as stated earlier.

2.3 - Pulse Density Modulation

With this type of modulation, the DC value is still expressed as the ratio of the time the bitstream is high to the bitstream period. But instead of being a variable width pulse with a period $N_b^*T_s$, this pulse is broken up into many smaller pulses. This is so the

density of the high pulses is uniform throughout one period of the bitstream. This modulation method effectively decreases the fundamental period increasing the frequency of the fundamental harmonics. Figure 2.10 shows an example of two bitstreams and their corresponding frequency spectrum which will generate the same DC value when filtered. In this example, it can be seen that the fundamental frequency for a PWM bitstream is $F_s/14$, and for a PDM bitstream it is $F_s/2$. This fact significantly reduce the performance requirements of the filter as the cutoff frequency can be much higher.

A PDM modulated bit-stream is simple to obtain when the number of ones divide evenly into the scan chain length N_b as was the case for the example stated in Section 2.1, but becomes more tedious when it does not. In its simplest form, the PDM encoding process is simplified if the bit-streams are obtained from a sigma delta modulator implemented in software. Figure 2.11 shows the functional blocks required to implement a 1^{st} order sigma delta modulator [3]. This modulator effectively encodes the desired DC level into a pulse density modulated bit-stream in a manner that minimizes the period of the fundamentals. By applying a DC signal of value N_1/N_b at the input of the modulator, where N_1 and N_b are relatively prime (no common factors), the period of the bitststream that is generated at the output will be N_b . This sequence can then be repeated periodically.



Figure 2.10 Modulation of a 0.5 level with PDM and PWM



Figure 2.11 Block Diagram of First Order Sigma-Delta Modulator

Note that in the last section of this chapter the use of higher order sigma-delta modulator to obtain PDM bitstreams will also be investigated.

As was the case for PWM, the spectral characteristic of PDM can be better understood and characterized using the Fourier series representations that was described in Eqn. (2.6) and (2.7). This understanding will then allow us to set the appropriate design requirements for the filter.

First, one must remember that a pulse density modulator breaks down a bit stream to the smallest period possible. Thus reducing the fundamental period and increasing the first harmonic in frequency. The fundamental frequency will vary according to the DC value that is desired. In other words, N of the Fourier's expansion will not always equal N_b of the bitstream. This is contrary to PWM in which the fundamental frequency (F_s/N) always remains the same.

Looking at the case for $N_b = 10$, where the resolution is 1/10, the bitstreams of Table 2.1 are obtained from a first order modulator. Once again these bitstream can be looked at in terms of their Fourier series expansion. The DC value is still represented as the number of ones (N₁) over N_b. Using Eqn. (2.6) to express these bitstreams it will be observed that the distribution of the power amongst the harmonic will be somewhat different than for PWM. This fact can be easily observed for the cases where N₁ divides equally into N_b. For the bitstreams of Table 2.1 this is the case for N₁ equals to 1, 2, and 5. In these cases N from Eqn. (2.8) will be respectively 10, 5, and 2. For all these cases N₁

equals one. In the case for N = 10 the equation yields tones at every $F_s/10$ just as for PWM. For the other two cases the tones are respectively situated at $F_s/5$ and $F_s/2$. In other words whenever N_1 divides equally into N_b , N will equal N_b/N_1 , and the tones for these bitstreams will be situated at every F_s/N . Using the previous fact and Eqn. (2.6), it is seen that as the magnitude of $|a_1|$ increases, so does the frequency at which it is located. It is not as simple to see this for the cases where N_1 does not divide equally into N_b . The $N_1 = 3$ case is an example of this. In this case N = 10, but the frequency can no longer be expressed by Eqn. (2.8) since the ones are spread out over N. It is however possible to obtain a good approximation by setting N equal to N_b/N_1 , and then approximating that the harmonics are at every F_s/N , or N_1F_s/N_b . This is just an approximation however, and applying Eqn. (2.6) will reveal that some power is actually present at every F_s/N_b , but the majority of it is indeed at N_1F_s/N_b .

Input	Output
0/10	00000000000
1/10	10000000010
2/10	100001000010
3/10	100100100010
4/10	100101001010
5/10	101010101010
6/10	011010110101
7/10	011011011101
8/10	011110111101
9/10	01111111101
10/10	111111111111

Table 2.1 - First order PDM bitstream for N_b = 10

The FFT of the first 6 bitstream patterns of Table 2.1 are shown in Figure 2.12. For the N₁ equals 0, 1, 2, and 5 cases where it divides equally into N_b, it is seen that the harmonics are situated at exactly N₁F_s/N_b. For the remaining cases there is power for every F_s/N_b but most of it is at N₁F_s/N_b. The remaining 4 patterns are the complements of



the first 4. Thus their FFT will remain the same except for the magnitude of the DC component.

The important characteristic that was brought forward in the previous discussion, is that as the magnitude of the 1st harmonic increases, the fundamental frequency also increases. Moreover through simulation, it is determined that the magnitude of the 1st harmonic increases at a rate of 20 dB/dec. Figure 2.13 illustrates the frequency spectrum of three bit-streams corresponding to N₁/N_b ratios of 3/16, 5/16 and 7/16. It is observed that the fundamental frequency of F_s*N_1/N_b is always largest in magnitude. If the largest harmonic of each bit-stream is plotted on the same frequency scale, it can be observed that their magnitude increase at a rate less than 20 dB/decade.



Figure 2.13 Frequency Spectrum for Different N1

Finally, the above observation of the 1st harmonic suggests that it is sufficient to design the first-order low-pass filter for the simplest bit-stream case consisting of a single one (N_1 =1) and N_b -1 zeros. Whatever is claimed for this case will apply to all other bit-stream patterns consisting of the same number of bits. This is because the first-order filter's attenuation increases at a rate of 20 dB/decade which is greater than the increase in magnitude of the fundamental component of the bit-stream and therefore offsets the increase in magnitude of the fundamental component.

2.4 - High-Order Noise Shaping

Up until now, the method used to obtain a PDM bitstream is to use a first order Delta Sigma ($\Delta\Sigma$) modulator. It is also possible to encode a DC value using a high-order modulator. The effect of high-order modulation will be to break up the periodic nature of the bitstream even more, thus further pushing the fundamental higher in frequency, thereby reducing the constraints on the filter bandwidth. However, note that for short bitstream length, i.e. small N_b , the outcome of high-order modulation is the same as for first order. Thus, the use of higher order modulation is justified when N_b is large.

The goal is to use a modulator that will leave the DC value undisturbed, and push the harmonics as far away from DC as possible in order to reduce the area and settling time of the filter. Figure 2.14 shows a block diagram of a modulator which can be implemented in software. The transfer function, H(z), can be designed to achieve any order of noise shaping (effect of pushing the harmonics higher in frequency)[3][8].

Figure 2.15 shows a plot of the frequency response of a modulated DC value for modulator orders of 1, 2, and 4. It is observed that the harmonics are shaped away from the DC tone, of which the power increases at a rate of 20, 40, and 80 dB/dec as we move away from DC for 1st, 2nd, and 4th order respectively. As the order of the modulator is increased further, the noise is pushed further away from DC and increases at an even higher rate. These characteristics will set the requirement on the filter bandwidth and order. In other words, as the order of the modulator is increased, the requirements on the cutoff frequency can be relaxed, but the roll-off of the filter will have to be higher, i.e. a high-order filter.

In order to compare the performance for different orders of modulation, an nthorder low-pass butterworth filter with transfer function



Figure 2.14 Block Diagram for Arbitrary Order Sigma Delta Modulator



Figure 2.15 High Order Modulation Frequency Response

is used. For modulator orders of 1, 2, 4, and 5 the cutoff frequency, f_c , that yields 10 bits resolution was found and is listed in Table 2.2. From this table the following observations are made. For a particular modulator order, the cutoff frequency can be relaxed as the order of the filter increases. Also, for a given filter order, the best case scenario is achieved when the order of the modulator is matched to the order of the filter. Thus, for there to be a benefit in using higher order modulation, a filter of the same order or higher must be used.

Filter\Modulator Order	1st Order ΣΔ	2nd Order ΣΔ	4th Order $\Sigma\Delta$	5th Order $\Sigma\Delta$
1	100 kHz	63 kHz	63 kHz	63 kHz
2	2 MHz	2.5 MHz	2.5 MHz	2 MHz
4	5 MHz	8 MHz	8 MHz	6.3 MHz
5	6.3 MHz	8 MHz	8 MHz	8 MHz

Table 2.2 - Cutoff Frequency required to achieve 10 bit resolution using Low-PassButterworth Filters

Two different approaches may be used to obtain the desired bit-stream. The first, is to build a sigma-delta modulator on-chip using digital circuits and connect its output to an analog filter. The other method is to use a software modulator as described previously. A sample set of N_b point is then chosen from the output and loaded into a scan chain on the chip to generate the appropriate bit-stream.

With an on chip modulator, the bit-stream essentially has an infinite length, as it is generated in real time by the modulator. Thus, the resolution of the DC generator will be limited only by the data-path width of the hardware modulator. For the scan-chain approach, the resolution is set by its length, N_b as stated in Eqn. (2.2).

The quality of the signal will be affected by the length, N_b , and the particular set of N_b bits used in the scan chain as depicted in Figure 2.16. Note that contrarily to the first order modulation case, the output of the modulator is no longer periodic. Thus, by repeating the same sequence in a periodic fashion there will be a degradation in the signal-to-noise ratio (SNR) from the infinite bitstream case, i.e., more ac ripple. In Dufort's [5] work on AC signal generation using 1-bit bitstream, it was shown that constraining the



Figure 2.16 Bit Selection Process

length of the infinite bitstream to N_b has an impact on the SNR, but as it is made larger this impact becomes negligible. In that same work it was also shown how a search optimization technique can improve the SNR of the bitstream. *In the present work, it was also found by simulation that choosing a sufficiently long bitstream and running the appropriate optimization will yield a signal of the same, or better, quality than that yielded by an infinite bitstream.* For the various orders of noise shaping experimented with (1st, 2nd, 4th, and 5th), measuring the SNR for different N_b showed that when

$$N_b \ge 2 \cdot N_{DC}, \tag{2.11}$$

where

$$N_{DC} = \frac{1}{\Delta DC},\tag{2.12}$$

the SNR will be as good as the infinite bitstream case. This can be further improved by running the optimization suggested in [5] with SNR and the proper average value as criterions.

Finally, in order to design the appropriate filter, the same approach as for the first order modulation case may be used, with a few new considerations. First, the order of the filter that is designed must be greater or equal to the order of the modulation technique used. Secondly, as higher order modulation is used the dynamic range will be reduced. This means that it will be no longer possible to generate the full range of DC voltage with higher order modulation. Consequently it will not always be possible to design the filter for the $1/N_b$ case. The suggested approach is then to design the filter that meets the resolution requirement for the first-order modulation case, and then, by simulating with a high-order bitstream, reduce the filter constraints until just before it stops satisfying the desired resolution.

2.5 - Summary

In this chapter, the general theory behind DC bitstream modulation was presented. The general filtering concepts such as bandwidth, and attenuation were presented as well. In light of these, the different bitstream modulation scheme were described and their trade-offs discussed. The following conclusion is reached. The DC bit-stream modulation scheme which has the simplest bandwidth requirements on the filter will be the most desirable. The reason being that the filter will have a lesser attenuation, thus a shorter settling time, for the same bit-rate.

Chapter 3 - Periodic Bit-Stream Filtering.

In the previous chapter, the basic theory behind the bit-stream DC voltage generator was presented. Different bit-stream modulation schemes were presented along with their respective filter requirements. In this chapter, the Low Pass Filter (LPF) component of the bit-stream voltage generator will be introduced. It will begin with a discussion of first-order filtering and then continue with high-order filtering along with the tradeoffs between passive and active implementations. Finally a semi-digital filter implementation will be introduced as a possibility for further work.

3.1 - First-Order Filtering

A first-order filter is the simplest method with which the DC average of a bitstream may be extracted. For this purpose it is best to have a LPF with a transfer function which has a zero at infinity and one pole on the negative real axis. One way to obtain such a transfer function is to use a low-pass RC filter circuit, which has the following transfer function

$$H(s) = \frac{1}{1 + sRC}.$$
 (3.1)

The unit step response of this filter will be
$$V(t) \approx 1 - e^{-t/(RC)},$$
 (3.2)

which dictates the convergence time, that is, the time it takes for the filter output to reach a particular output level. Such an implementation is good for applications where the convergence time is not critical and the requirement on the attenuation is not too strict. As more attenuation is desired, the convergence time will increase dramatically and so will the component values of the RC filter. In the case of filtering a 1st order PDM bit-stream, the following will describe the design requirements and illustrate them with a numerical example.

In the previous chapter, it was shown that a first order filter can be designed for the simplest PDM case, i.e., when $N_1 = 1$. For this simple bit-stream arrangement, the harmonics are of decreasing magnitude beginning with the DC tone of magnitude of $1/N_b$. Further, the harmonics are located at frequencies which are multiples of F_s/N_b . After passing the bit-stream through the RC filter, the RMS magnitude of the kth harmonic can be described as follows

$$\left|H_{k}\right| = \frac{1}{N_{b}} \frac{1}{\sqrt{1 + \left(2\pi R C k \frac{F_{s}}{N_{b}}\right)^{2}}}.$$
(3.3)

To satisfy the AC ripple requirements, the sum of the power in all the AC components must be made less than the RMS value of the AC ripple requirement, i.e.,

$$\sum_{k=1}^{\infty} H_k^2 \le \left(\frac{1}{2} \times \frac{1}{N_b}\right)^2.$$
(3.4)

Consequently, substituting the appropriate expression for the harmonics leads to an equation involving RC, i.e.,

$$\sum_{k=1}^{\infty} \frac{1}{1 + \left(2\pi RC \cdot k \cdot \frac{F_s}{N_b}\right)^2} \le \frac{1}{4}.$$
(3.5)

Limiting the summation to approximately 100 terms (as the higher order terms add little to the summation) enables one to numerically search for the value of RC that best satisfies the ripple requirement.

For instance, a bit-stream operating at 1 GHz is to produce a DC signal with a normalized DC resolution of 1/256. This suggests that the number of bits in the bit-stream should be at least 256. To simplify the amount of digital hardware, $N_b=256$ is selected. Next, the RC time constant of the first-order filter is determined from the following expression using the iterative numerical procedure just derived, i.e.,

$$\sum_{k=1}^{100} \frac{1}{1 + \left(2\pi RC \cdot k \cdot \frac{10^9}{256}\right)^2} \le \frac{1}{4}.$$
(3.6)

The result is RC = 99.1 ns for an RMS AC ripple of 0.14 mV or a 0.198 mV amplitude AC ripple. To select *R* and *C*, consider the spectral density of the thermal noise generated by the resistor, i.e.,

$$S_{\nu}(f) = 4kTR. \tag{3.7}$$

Here *T* is the temperature in Kelvin and *k* is the Boltzmann's constant (1.38 x 10^{-23} JK⁻¹). Note that the noise spectrum is frequency independent. When this thermal noise passes through an RC filter, the output noise signal will have an RMS value of

$$V_{N-RMS} = \sqrt{\frac{kT}{C}}.$$
(3.8)

Interestingly, it is independent of R. The latter equation essentially sets a lower bound on the capacitor value that may be used. As the RMS noise should be less than the RMS AC ripple, that is

$$V_{N-RMS} < \sum_{k=1}^{\infty} H_k^2$$
(3.9)

or

$$V_{N-RMS} < \frac{1}{\sqrt{2}} \times \frac{1}{512}$$
 (3.10)

Setting this value equal to V_{N-RMS} in Eqn. (3.8), the smallest *C* is 2.17 fF. Consequently, once *C* is set to a value greater or equal to 2.17 fF, the value of *R* may be solved using the fact that RC = 99.1 nsec. In a 0.35 µm CMOS process, choosing C = 305 fF will yield R = 330 K Ω and an equal area for the capacitor and resistor.

Referring back to Eqn. (3.2), the convergence time may be computed from RC as follows

$$t_{conv} = -RC \cdot \ln(1 - V(t_{conv})), \qquad (3.11)$$

where t_{conv} is the time at which V(t) is within half a least significant bit (LSB) of the final voltage (normalized to 1 in this case), i.e.,

$$V(t_{conv}) = 1 - \frac{1}{2} \cdot \frac{1}{N_b}.$$
 (3.12)

In the previous example a convergence time of 618.2 nsec is obtained, which is quite significant. The only way to achieve a faster convergence time for the same resolution will be the use of a high-order filter since in this case it is fixed by the choice of RC.

3.2 - High-Order Filtering

As a general rule, the transient response of a high-order $(n \ge 2)$ filter will fall into 3 classes of behaviour. These being: underdamped, overdamped, and critically damped transient behaviour. From a pole placement perspective, the underdamped filter has its poles in complex conjugate pairs in the left-half portion of the s-plane, the overdamped filter has its poles distributed on the negative real axis, and the critically damped filter has all of its poles at one location on the negative real axis [9]. For a second order biquad structure, this corresponds to a Q greater than, less than, and equal to 1/2, respectively.

The case which will yield the best tradeoff between convergence time and resolution is the critically damped filter. In other words, a filter with all of its poles lying at

one location on the negative real axis on the s-plane. This is evident from Figure 3.1 where the transient behaviour of three different filters are plotted. Each filter response was designed to achieve a 12-bit resolution, i.e.,

AC ripple
$$< \frac{1}{2} \times \frac{1}{2^{12}}$$
. (3.13)

In terms of the filter realization, two main types of implementation can be used; these being passive or active. Each has its advantages and disadvantages. Passive structures here are assumed to be realized using resistors and capacitors in a ladder configuration. Such passive filters can only realize filter functions having an overdamped response, as the poles for such a structure can only be placed on the negative real axis and at distinct locations. As for active filters, they can be used to realize complex conjugate poles, i.e., a filter with an underdamped response, and they can also be used to realize a filter with a critically damped response.

Following is an in-depth analysis of a passive and an active filter implementation used for DC generation, along with the advantages, disadvantages and limitations of each.



Figure 3.1 Filter Transient Response

3.2.1 - **Passive**

An M/2-th order low-pass RC filter can be realized by cascading M/2 first-order low pass RC filter sections¹ as depicted in Figure 3.2. With this realization there is M/2 zeros at infinity. Poles can be anywhere on the negative real axis as long as they are at distinct locations, i.e.,

$$H(s) = \frac{1}{d_{M/2}s^{M/2} + \dots + d_2s^2 + d_1s + d_0}.$$
 (3.14)

In order to approach the critically-damped case, the poles must be placed as close to one another as is practically possible.

To synthesize such a filter one can make use of the driving-point synthesis method [10]. The goal of the driving-point synthesis method is to express the natural modes of the filter function H(s) in terms of a driving point impedance Z(s) or admittance Y(s), in such a way that it can be reduced and recognized as a known network function. The realizability condition set on the driving-point impedance or admittance function is that it must be a positive real function. This condition will constrain the location of the poles and zeros as well as the general form of the driving-point impedance or admittance function. For Z(s) or Y(s) to be positive real they must be a ratio of an even (odd) and an odd (even) polynomial. Moreover, the order of the numerator and denominator polynomials must not differ by more than one. In other words, the impedance or admittance function has the following form,



Figure 3.2 Low Pass M/2-th Order RC Filter

^{1.} It was chosen not to use inductive elements as they perform poorly at low frequency and are hard to integrate on an IC.

$$Y(s) \lor Z(s) = \frac{a_{M/2}s^{M/2} + \dots + a_2s^2 + a_1s^1 + a_0}{b_{M/2-1}s^{M/2-1} + \dots + b_1s^1 + b_0}.$$
(3.15)

Finally, the zeros and poles of this impedance or admittance function must be interleaved on the negative real axis. If these requirements are not satisfied, the driving-point impedance or admittance function will not be positive real, thus non-realizable using passive components.

The driving-point admittance function for an RC filter structure possesses the following two properties. The critical frequency nearest the origin is a zero and the critical frequency nearest infinity (or at infinity) is a pole. Looking into the output port of the M/2-th order RC filter shown in Figure 3.2 with the input port short-circuited, one can view the resulting network as a one-port network having an RC Cauer I canonical form. The two critical frequencies just mentioned will then be a zero closest to the origin and a pole at infinity.

In general, the driving point admittance of the RC network shown in Figure 3.2 looking back into the output port is as follows

$$Y_{M}(s) = \frac{1}{Z_{M}(s)} = sC_{M} + \frac{1}{R_{M-1} + \frac{1}{sC_{M-2} + \dots \frac{1}{R_{3} + \frac{1}{sC_{2} + \frac{1}{R_{1}}}}}.$$
(3.16)

However, $Y_M(s)$ is usually expressed as a rational function of two polynomials,

$$Y_M(s) = \frac{a_{M/2}s^{M/2} + \dots + a_2s^2 + a_1s^1 + a_0}{b_{M/2 - 1}s^{M/2 - 1} + \dots + b_1s^1 + b_0}.$$
(3.17)

It is therefore the objective of the present synthesis method to determine the set of R's and C's from a given set of polynomial coefficients $\{a_{M/2}, ..., a_3, a_2, a_1, a_0\}$ and $\{b_{M/2-1}, ..., b_2, b_1, b_0\}$. To achieve this, consider equating (3.16) with (3.17), to obtain

$$Y_{M}(s) = \frac{a_{M/2}s^{M/2} + \dots + a_{2}s^{2} + a_{1}s^{1} + a_{0}}{b_{M/2 - 1}s^{M/2 - 1} + \dots + a_{1}s^{1} + b_{0}}$$

$$= sC_{M} + \frac{1}{R_{M - 1} + \frac{1}{sC_{M - 2} + \dots + \frac{1}{R_{3} + \frac{1}{sC_{2} + \frac{1}{R_{1}}}}} . (3.18)$$

From this equation a set of nonlinear equations could be written from which the unknown coefficients can be solved using numerical techniques, or one could use the following iterative procedure which is derived from the LC ladder synthesis method presented in [11].

The first step is to solve for C_M by dividing both sides of the equation by s and taking the limit as $s \to \infty$ to obtain

$$\lim_{s \to \infty} \left[\frac{Y_M(s)}{s} \right] = C_M + \lim_{s \to \infty} \left[\frac{1}{s} \cdot \frac{1}{R_M + \frac{1}{sC_{M-1} + \dots + \frac{1}{R_2 + \frac{1}{sC_1 + \frac{1}{R_1}}}} \right].$$
 (3.19)

As the right most term tends to zero, one obtains

$$\lim_{s \to \infty} \left[\frac{Y_M(s)}{s} \right] = C_M.$$
(3.20)

The second step is to deflate the impedance function with the known coefficient C_M . This is done by subtracting sC_M from $Y_M(s)$ as follows

$$Y_{M}(s) - sC_{M} = \frac{1}{R_{M-1} + \frac{1}{sC_{M-2} + \dots + \frac{1}{R_{3} + \frac{1}{sC_{2} + \frac{1}{R_{1}}}}},$$
(3.21)

which is inverted to obtain

$$Z_{M-1}(s) = \frac{1}{Y_M(s) - sC_M} = R_{M-1} + \frac{1}{sC_{M-2} + \dots + \frac{1}{R_3 + \frac{1}{sC_2 + \frac{1}{R_1}}}.$$
 (3.22)

Next, it is recognized that R_{M-1} can be obtained from Eqn. (3.22) by taking the limit $s \rightarrow \infty$ on both sides

$$\lim_{s \to \infty} [Z_{M-1}(s)] = R_{M-1} + \lim_{s \to \infty} \left[\frac{1}{sC_{M-2} + \dots + \frac{1}{R_3 + \frac{1}{sC_2 + \frac{1}{R_1}}}} \right]$$
(3.23)

to obtain R_{M-I} as the limit on the right hand side goes to zero. The fourth step is to deflate the impedance function with the known coefficient R_{M-I} . This is done as follows

$$Z_{M-1}(s) - R_{M-1} = \frac{1}{sC_{M-2} + \dots + \frac{1}{R_3 + \frac{1}{sC_2 + \frac{1}{R_1}}},$$
(3.24)

which is inverted to obtain

$$Y_{M-2}(s) = \frac{1}{Z_{M-1}(s) - R_{M-1}} = sC_{M-2} + \dots \frac{1}{R_3 + \frac{1}{sC_2 + \frac{1}{R_1}}}.$$
 (3.25)

The fifth step is to repeat the first four steps until there is only R_1 left, that is

$$Z_1(s) = \frac{1}{Y_2(s) - sC_2} = R_1.$$
(3.26)

A driving point synthesis example of a third order RC-filter, which makes use of the previous method, will be given. Consider that the following transfer function with normalized poles is to be realized,

$$H(s) = \frac{1}{(s+1)(s+2)(s+4)}.$$
 (3.27)

First, the natural modes of the desired transfer function must be expressed as a driving point admittance function. This is done using the following two observations. The natural modes of the transfer function are the same as the zeros of the driving-point admittance function. The poles of the driving-point admittance function may be chosen arbitrarily as long as the resulting function is positive real. Thus, the following is a valid driving-point admittance function (one pole is placed at infinity as was mentioned earlier),

$$Y_6(s) = \frac{(s+1)(s+2)(s+4)}{(s+1.5)(s+3)}$$
(3.28)

It has been the author's experience that the best choice for the admittance pole locations is found to be midway between the zeros. This is so that the component spread is kept to a minimum.

Following the procedure described above, the first step is to calculate the residue as follows

$$\lim_{s \to \infty} \left[\frac{Y_6(s)}{s} \right] = \lim_{s \to \infty} \left[\frac{(s+1)(s+2)(s+4)}{s(s+1.5)(s+3)} \right] = 1,$$
(3.29)

where $C_6 = 1$ F is obtained. The second step is to deflate the admittance function as follows

$$Y_6(s) - s = \frac{16 + 19s + 5s^2}{(3 + 2s)(3 + s)}$$
(3.30)

and obtain

$$Z_5(s) = \frac{1}{Y_6(s) - s} = \frac{(3 + 2s)(3 + s)}{16 + 19s + 5s^2}.$$
(3.31)

Performing the third step, the following residue is obtained

$$\lim_{s \to \infty} [Z_5(s)] = \lim_{s \to \infty} \left[\frac{(3+2s)(3+s)}{16+19s+5s^2} \right] = \frac{2}{5},$$
(3.32)

thus $R_5 = 2/5 \Omega$. Executing the fourth step, one can write

$$Z_5(s) - 2/5 = \frac{1}{5} \cdot \frac{13 + 7s}{16 + 19s + 5s^2}$$
(3.33)

and

$$Y_4(s) = \frac{1}{Z_5(s) - 2/5} = 5 \cdot \frac{16 + 19s + 5s^2}{13 + 7s}.$$
 (3.34)

Repeating step one,

$$\lim_{s \to \infty} \left[\frac{Y_4(s)}{s} \right] = \lim_{s \to \infty} \left[\frac{5}{s} \cdot \frac{16 + 19s + 5s^2}{13 + 7s} \right] = \frac{25}{7} , \qquad (3.35)$$

 $C_2 = 25/7$ F is obtained. Applying step two again, one can write

$$Z_3(s) = \frac{1}{Y_4(s) - s\frac{25}{7}} = \frac{7}{20} \cdot \frac{13 + 7s}{28 + 17s}.$$
(3.36)

Then apply step three,

$$\lim_{s \to \infty} [Z_3(s)] = \lim_{s \to \infty} \left[\frac{7}{20} \cdot \frac{13 + 7s}{28 + 17s} \right] = \frac{49}{340}$$
(3.37)

to obtain $R_3 = 49/340 \Omega$. Deflating $Z_3(s)$ according to step four,

$$Y_2(s) = \frac{1}{Z_3(s) - 49/340} = \frac{68}{35} \cdot (28 + 17s), \qquad (3.38)$$

from which the residue (step one) is obtained as follows

$$\lim_{s \to \infty} \left[\frac{Y_2(s)}{s} \right] = \lim_{s \to \infty} \left[\frac{68}{35} \cdot (28 + 17s) \right] = \frac{1156}{35} .$$
(3.39)

Hence, $C_2 = 1156/35$ F. Deflate $Y_2(s)$ (step two) to get

$$Z_1(s) = \frac{1}{Y_2(s) - s\frac{1156}{35}} = \frac{5}{272},$$
(3.40)

thus completing the synthesis as $R_I = 5/272 \ \Omega$.

It is observed through experimentation with this method that the spread between the admittance zeros will dictate the component spread of the circuit realization. The closer the zeros are to one another, the greater the spread in component values. In the previous example, the spread of the capacitors was approximately 30, which is quite significant if this filter is to be implemented on a single IC.

At this point it is important to remember that the goal of the passive RC filter design of this section is to approximate the critically damped case, whereby the poles of the transfer function are placed as close as possible without overlapping. With the use of the synthesis method just presented, it is possible to realize an RC filter with its poles placed anywhere on the negative real axis as long as they don't overlap. Of course, the poles should not be placed too close to one another otherwise the spread in component values will be too large. Through a trial and error procedure¹, Table 3.1 lists the frequency-normalized pole positions for approximating a critically damped transient

response using a 2nd-8th order RC filter having a component spread ranging from 9-10.9. Also shown in this table are the corresponding resistor and capacitor values.

Filter Order	Normalized Poles	Normalized Capaci- tance (F)	Normalized Resistance $(m\Omega)$	Capacitor Spread
2	1, 2	C ₂ =9, C ₄ =1	R ₁ =83.3, R ₃ =666.7	9
3	1, 3, 6	C ₂ =9.8, C ₄ =2.6, C ₆ =1	$R_1 = 45.8, R_3 = 168.5, R_5 = 285.7$	9.8
4	1, 4, 8, 16	$C_2=10.9, C_4=5, C_6=2.25, C_8=1$	$R_1=30.8, R_3=57.7, R_5=81.6, R_7=111.1$	10.9
5	1, 5, 10, 20, 40	$C_2=9.47, C_4=4.68, C_6=2.77, C_8=1.77, C_{10}=1$	$R_1=2.4, R_3=45.8,$ $R_5=43.9, R_7=49.3,$ $R_9=47.6$	9.47
6	1, 5.5, 11, 22, 44, 88	$C_2=10.9, C_4=5.23, C_6=3.36, C_8=2.11, C_{10}=1.62, C_{12}=1$	$R_1=1.79, R_3=34.5,$ $R_5=29.3, R_7=27.6,$ $R_9=26.7, R_{11}=22.2$	10.9
7	1, 7, 14, 28, 56, 112, 224	$C_2=9.12, C_4=4.80,$ $C_6=3.57, C_8=2.44,$ $C_{10}=1.77, C_{12}=1.55,$ $C_{14}=1$	$R_1 = 1.64, R_3 = 30.0, R_5 = 21.4, R_7 = 16.5, R_9 = 13.9, R_{11} = 11.6, R_{13} = 8.8$	9.12
8	1, 8, 16, 32, 64, 128, 256, 512	$C_2=9.55, C_4=5.28,$ $C_6=4.19, C_8=2.96,$ $C_{10}=2.11, C_{12}=1.66,$ $C_{14}=1.52, C_{16}=1$	$R_{1}=1.36, R_{3}=24, R_{5}=15.7, R_{7}=11.1, R_{9}=8.6, R_{11}=6.77, R_{13}=5.30, R_{15}=3.89$	9.55

 Table 3.1 - RC Ladder Approximating a Critically-Damped Response

Returning to the problem at hand in this chapter, once a particular RC filter network is selected for the role of extracting the DC level from a bit-stream, it must be frequency scaled in such a way that the AC ripple is minimized. Unlike the approach used

The trial and error procedure begins by distributing a set of N poles along the negative real axis
of the s-plane. One pole is place at -1 rad/s and the other N-1 poles are placed somewhere below
-1 rad/s. Next, a passive RC ladder is synthesized using the procedure described in this section
and the component spread is computed. Next, the poles, except the pole at -1 rad/s, are moved in
such a way that it decreases the RC ladder component spread. The procedure is repeated until
the desired component spread is obtained.

in Section 3.1 where closed formulas were available, here we must select the frequency scale factor, say α , by iterating through several numerical simulations involving the PDM bit-stream and a particular RC ladder network. If, after an exhaustive search, the AC ripple remains outside the desired limit, then a higher order filter will be required.

To frequency scale an RC ladder network, one simply divides all the capacitors by the scale factor according to the following,

$$C_{freq-scaled} = \frac{C}{\alpha}.$$
 (3.41)

Finally, one can impedance scale the RC network so that the R's and C's are realizable on an IC by simply multiplying the impedance of the R's and C's by the same scale factor ξ , i.e.,

$$R_{impedance-scaled} = \xi \cdot R \text{ and } C_{impedance-scaled} = \frac{C}{\xi}.$$
 (3.42)

The major advantage of an all passive low-pass filter is robustness. The reconstructed DC value will be insensitive to component matching and variation. This is due to the nature of its transfer function, which has unity gain at DC irrespective of the value of RC. The only effect component variation will have on the DC generator is its AC ripple. Thus as long as the filter has enough margin, it can be readily assumed that the correct DC value and AC characteristics will be obtained.

The main disadvantage is that it is hard to approximate the critically damped case while keeping a reasonable component spread. Also, as the filter order is increased, the poles become more distant from one another, thus reducing the advantages of going to a higher order filter.

3.2.2 - Active

There exist a good number of active structures that can realize a filter with a critically damped response. One such structure is the single amplifier biquadratic (SAB) active filter.

SAB filters make use of one op amp and an RC network in the feedback loop to implement a biquadratic response. The main advantage of this type of active filter is that it only requires one op amp to realize a second order biquadratic stage, where other active filter implementations may require as many as four. Figure 3.3 illustrates an example of an SAB filter. This filter is obtained by applying the complementary transform to an SAB filter with a bridged-T network in its feedback path [9]. The transfer function of this filter yields the following low pass biquadratic filter function

$$\frac{V_{out}}{V_{in}} = \frac{\frac{1}{R_1 R_2 C_1 C_2}}{s^2 + s \left(\frac{1}{R_2 C_2} + \frac{1}{R_1 C_2}\right) + \frac{1}{R_1 R_2 C_1 C_2}}.$$
(3.43)

Typically R_1 and R_2 are made equal to R. Also C_2 is set to C and $C_1 = C/m$, where m is a scaling factor. With these change of variables one can see that

$$m = 4Q^2 \tag{3.44}$$

and

$$CR = 2(Q/\omega_o). \tag{3.45}$$

For the current synthesis problem, Q = 1/2 in order to obtain a critically damped response. This sets the scaling constant m = 1 according to Eqn. (3.44). The synthesis problem will then become a matter of selecting R and C such that the cutoff frequency requirement is satisfied, i.e,

$$\omega_o = \frac{1}{RC}.$$
(3.46)

An important design consideration of an active filter for DC generation is its ability to handle rapid input signal transition related to PDM bit-stream. For the filter topology shown in Figure 3.3, the fast edges of the PDM signal are slowed down by the front-end RC structure before the signal appears at the input of the op-amp, hence, avoiding op-amp slew-rate limiting effects. For other SAB filter topologies this requirement might not always be met as the input may not always go through an RC structure before reaching the op amp. Lastly, in order to obtain a filter of order greater than two, a number of these SAB biquad section can be cascaded.

There are two major advantages to an active implementation, one is that it is possible to realize the ideal biquad (Q =1/2) that is the best compromise between convergence time and resolution and, two, it is also possible with certain SAB filter topologies to introduce a DC gain, thus adding an extra degree of freedom to the system. However the use of operational amplifiers introduce noise and non-linearity. Moreover, the DC gain present in active implementation is dependent on component matching, which is a source of gain-error.

Table 3.2 shows a comparison of a first order passive filter with a 4th order passive and a 4th order active filter. The first order passive design is a simple low-pass RC filter while the 4th order passive design is the high order RC filter of Figure 3.2 with M = 8, and the 4th order active design is a cascade of two of the SAB stage depicted in Figure 3.3. One can see from this table that a high-order filter will yield a considerably faster convergence time. As to the differences in performance between high-order active and



Figure 3.3 SAB Biquad stage

passive structures, one can observe a faster convergence time for the active realization due to Q = 1/2, but at a greater cost in area and complexity, and of course, process sensitivity.

Filter	1 st Order Passive RC	4 th Order Passive RC	4 th Order SAB
Poles (rad/sec)	198.69 K	12.54 M, 50.15 M, 1.00 G, 2.01 G	39.64 M, 39.64 M, 39.64 M, 39.64 M
R (KOhms)	503	$R_1=4.1, R_3=9.2,$ $R_5=13.1, R_7=17.8$	$R_{11} = R_{12} = R_{21} = R_{22} = 5$
C (pF)	10	$C_2=5.4, C_4=2.5, C_6=1.1, C_8=0.5$	$C_{11} = C_{12} = C_{21} = C_{22} = 5$
Settling Time	40 µsec	644 nsec	381 nsec

Table 3.2 - Comparison of the different filters

3.3 - Semi-Digital Filtering

Another method with which the DC average could be extracted from a bit-stream is to use a semi-digital FIR reconstruction filter. Although this thesis does not experimentally explore this filter type, it is included here for its future potential. Figure 3.4 shows the general structure of an N-th order FIR filter. From this figure, the input x[n] is the modulated bit-stream introduced in the previous chapter, and y[n] is a discrete analog



Figure 3.4 Semi-digital FIR reconstruction filter

output. The coefficients a_1 through a_N are the analog weighing coefficients. The transfer function of such an FIR filter is given by

$$H(z) = a_1 z^{-1} + a_2 z^{-2} + \dots + a_{N-1} z^{-N+1} + a_N z^{-N}.$$
 (3.47)

The frequency response of such an FIR filter is dictated by the choice of coefficients a_1 to a_N . It is possible to realize arbitrary frequency responses, such as low-pass, high-pass, and band pass, by changing the value of these coefficients. For the present purpose, a low-pass frequency response is desired in order to extract the DC component of the bit-stream. There exist a number of algorithms and methods to determine the value of coefficients that will yield the desired frequency response. One common one is the Parks-McClellan algorithm which is available within the MATLAB software. One other algorithm based on least-square method is described in [12].

One possible way to implement the filter of Figure 3.4 is to use a differential current mode topology as is depicted in Figure 3.5. In this topology, the weight of each coefficients a_1 to a_N , is determined by the value of the current source I_1 to I_N , i.e. $a_i = I_i$. The delays are implemented using D flip-flops, not shown, of which the Q and \overline{Q} output are used to drive each summing node. In [13], a novel current steering topology illustrated



Figure 3.5 Current-mode topology of an N-th order semi-digital FIR reconstruction filter.

in Figure 3.6 is introduced to replace the one of Figure 3.5. In the aforementioned work, it is shown that this novel topology has numerous advantages, with the most significant being the need for a single current source instead of N. In this topology, the weight of a coefficient is determined by the ratio of differential pairs in the ith branch and the total number of differential pair in the filter, i.e.,

$$a_i = \frac{T_i}{\sum\limits_{i=1}^{N} T_i}$$
(3.48)

The output of such a filter is a discrete time analog current. So an output stage that performs a current to voltage conversion will be required at the output nodes of the filter. Additionally to doing the current to voltage conversion, the output stage must introduce a pole in order to attenuate the repeating images introduced by the discrete nature of the filter. Such an output stage is depicted in Figure 3.7.



Figure 3.6 Novel Current-mode topology.





What makes such an implementation attractive is that it is mostly digital. The only analog components are a current source, and the output stages. The D flip-flops and switch banks, which account for the greater proportion of this filter, are digital and will shrink in size as the technology scale. Further work has to be done in order to make a reasonable comparison with the other method presented so far. In [13] some promising results were presented for when this filter implementation is used to reconstruct an AC waveform from a PDM bit-stream.

3.4 - Summary

In this chapter, the filter component of the voltage generator of this thesis was presented. It was shown that a high-order filter is more desirable when low convergence time and high resolution are of importance. The requirements to achieve the best resolution versus convergence time tradeoff for a high-order filter were also introduced. The design and synthesis of a passive and an active filter which best meet these requirements was also presented.

Chapter 4 - DC Voltage Generator Implementation

Thus far the theory behind bit-stream generation and filtering has been presented. In this chapter, a brief overview of different implementations of the bit-stream generator will be introduced. Also, the circuits which were designed and implemented for the purpose of this thesis will be presented in detail.

4.1 - Bit-Stream Generator Implementation Overview

The goal of this section is to give the reader a broad overview of some possible ways a bit-stream generator can be implemented. The descriptions are very high level and used to illustrate the trade-offs such as the achievable bit-rate, the amplitude resolution, the area, and the power consumption of each design at a qualitative level. More details will be given for three of these implementation methods in the last section as they were used in the design of the prototype IC's of this thesis.

4.1.1 - Synchronous Scan Chain

Figure 4.1 illustrates how a series of N_b D flip-flops can be combined to form a synchronous scan-chain. The basic building block, a flip-flop, is readily available in



Figure 4.1 Bit Stream Generator

standard cell libraries, and can also be synthesized using CAD tools. The data can be loaded into the scan chain either serially from off chip or in parallel if the data already resides on the chip.

The maximum bit-rate depends on the propagation delay of one flip-flop. The rate of the clock going to all the flip-flops will set the bit-rate of the bit-stream. Proper layout techniques must be used to minimize clock skew, and to provide sufficient clock buffering for larger design. With this implementation the amplitude resolution is set by $1/N_b$. In other words it depends on the total number of flip-flops. This will have a direct impact on the total area of the scan-chain as it is directly proportional to the bit-stream length. Finally, the total power dissipated will also be proportional to the bit-stream length.

4.1.2 - Asynchronous Scan Chain

It is possible to make use of an asynchronous First-In-First-Out (FIFO) buffer to implement an asynchronous scan-chain which does not require a clock. Essentially a FIFO eliminates the need for a clock by introducing a communication protocol between each stage of the scan-chain. As with the clocked scan-chain, the bit-patterns can be loaded serially from off chip or in parallel from an on chip memory.

The speed of an asynchronous implementation is technology and process dependent. The bit-rate will be determined by the time it takes a stage in the FIFO to propagate a data bit. The benefit of this implementation over the previous one is that one does not have to worry about design issues such as clock buffering, skew, and generation.

The amplitude resolution is still set by $1/N_b$, thus dependent on the total number of stages in the FIFO. Area wise a FIFO stage is equivalent to a flip-flop, so the area impact is the same as for the synchronous scan-chain. Due to the clock-less nature of this implementation, no power will be dissipated by a stage when the state of a data bit remains the same. In other words, power will only be dissipated when a data bit toggles state. So the worst case scenario will be when each stage has an opposing state, thus causing them to toggle at every cycle. The power will be proportional to the number of transition in the bit-stream and has the advantage over the last implementation that it can be controlled by the choice of bit-pattern loaded into the scan-chain.

4.1.3 - Memory Based Design

Memory of different forms such as dynamic or static Random Access Memory (RAM) and Read Only Memory (ROM) can be used to generate bit-streams. Memory can also be used in conjunction with one of the previous implementations to load data in parallel into a scan chain or it can be used to source the bit stream directly.

The memory must be loaded from off chip using an appropriate memory controller. The bit-rate achievable will be constrained by how fast data can be accessed into memory. The amplitude resolution (minimum step size) is determined by the size of the memory, which will set N_b . In terms of area and power, memories tend to be more efficient, especially dynamic RAM. Moreover, memory is already part of a lot of systems. Thus it can be made available for the bit-stream generator at little or no extra cost in area.

4.1.4 - Hardware Modulator

Another implementation that can be considered is to implement the algorithm that generates the bit-stream on chip using digital hardware such as adders, counters, etc ... Such implementations can generate PWM, PCM, and also PDM bit-streams [14][15].

One advantage of such implementations is that no bit-stream loading is necessary. A 2^n long bit-stream may be generated by setting n control bits. However, the bit-rate of such implementation is less than using the scan-chain approach due to the use of arithmetic blocks like adders and comparators. There is also a loss in flexibility in such a design as the bit-stream encoding scheme is set by the hardware implementation.

The resolution of this implementation will be dictated by the width of the data paths in the arithmetic blocks used. For higher resolution applications this type of implementation will be more energy and power efficient than for the scan-chain approach as less logic gates will be needed.

4.1.5 - Automatic Test Equipment (ATE)

A significantly different approach to generate a DC signal is to use an external source as the bit-stream generator. Figure 4.2 shows an example of how a production tester can be used to generate bit patterns to feed a chip. Although very application specific, i.e., mainly for testing applications, this method has numerous advantages. The resolution is no longer dependent on the implementation's architecture as the bit-stream length N_b can



Figure 4.2 ATE Based Bit-stream Generator

be arbitrarily set in sofware. Moreover, there is no area or power penalty at the chip level since the bit-stream generator is all off chip. However the achievable bit-rate will be lower than for previous implementations as the bit-stream comes from off-chip and is limited in speed by the package parasitics and the ATE capabilities.

4.2 - Voltage Generator Design

In this section, the different prototype IC's that were designed and fabricated for the purpose of this thesis will be presented. The designs fall into two categories which are synchronous and asynchronous voltage generator. A synchronous scan-chain and a PWM hardware modulator are designed in the first subsection, and a set of asynchronous scanchain designs are described in the last subsection. The goals of the different experiments, which will be performed in the next chapter, will also be stated for each of these designs.

4.2.1 - Synchronous Voltage Generator

In this subsection, the designs are based on the synchronous scan-chain, and hardware modulator implementation methods briefly introduced in the previous section. The goal of these designs is to implement a DC reference voltage with an 8-bit resolution, i.e., 256 DC levels. They are to be used to experimentally verify the concepts of PDM and PWM introduced in Chapter 2.

Figure 4.3 illustrates the different components which make up the prototype IC. The different components are a PWM generator, an RC filter, and a PDM generator. This IC is shared with another design which is labelled "Other Circuitry". This other design is not relevant to the present goal and is not discussed. However, the phase locked loop (PLL), which is part of this other design will be used as a clock signal.



Figure 4.3 IC Implementation Overview

Figure 4.4 shows the different parts of the pulse width modulator component. They are a counter and a comparator. The counter is illustrated in Figure 4.5. It is clocked by the PLL and continuously counts in a circular fashion. It is 8-bit wide, thus can count up to 256. All the logic blocks used to implement this counter are standard cells available in the CMOSP35 technology library made available through the Canadian Microelectronics Corporation (CMC). The critical path, which is three gate delay, sets the maximum clock rate at 650 MHz.





The comparator illustrated in Figure 4.6 compares the counter output to an 8-bit reference input. This reference input sets the width of the pulse. When the count is zero it causes the output to be go high and when the count is equal to the reference input it causes the output to go low. Once again this component is implemented using standard cells. Note that for the D flip-flop generating the output, the cell with the maximum driving capability was chosen since it must drive the LPF.

A low pass RC filter is integrated on chip and connected to the output of the PWM generator. The LPF is a second-order low pass RC filter. For this filter $R_1 = R_3 = 130 \text{ K}\Omega$ and C2 = C4 = 12.5 pF. It is designed to provide 30 % more attenuation than needed for an 8-bit resolution at an F_s of 500 MHz. It can therefore allow for a reduction in clock speed and process variation. The design of this filter was not done using the methodology explained in the previous chapter as it came at an earlier stage of this research work.



Figure 4.6 Comparator

In order to generate a PDM bit-stream, a 256 bit long scan-chain, as illustrated in Figure 4.7, is also integrated on the chip. It is clocked by the same PLL as the PWM generator. The bit pattern is loaded from off chip through a multiplexer (MUX). This scanchain does not have a filter on chip, thus the output is routed to a digital pin and connected to an off-chip filter. The digital pin, which is available as a standard cell, provides the buffering necessary to drive the off chip LPF. The flip-flops used to implement this scanchain are taken from the standard cell library and have a maximum operating frequency in excess of 800 MHz.

Finally the PLL which is integrated on chip as part of another design is used in order to obtain an F_s as large as possible. The specifications for this PLL give a maximum operating frequency of 500 MHz, thus $F_s = 500$ MHz. With the type of packaging used, a PGA 84, input/output pins can have a maximum switching rate of about 20 MHz, which is why a PLL is needed.

This design was implemented in a 3.3 V 0.35 μ m CMOS technology. Figure 4.8 shows a micrograph of this design with the important areas highlighted. Note that special consideration was taken in order to isolate the counter of the PWM generator from the RC filter. This was done using guard rings around the filter, which serve the purposes of suppressing some of the switching noise introduced by the counter and the PLL.



Figure 4.7 PDM Bit-stream Generator



Figure 4.8 Chip Micrograph of PWM/PDM Generator

4.2.2 - Asynchronous Voltage Generator

In this subsection, the asynchronous scan-chain implementation method of Section 4.2.2 will be used. Three asynchronous scan-chain designs will be presented, each one building on the previous one.

Design A

With this first design, it is desired to investigate the use of asynchronous logic for bit-stream generation.

Figure 4.9 shows the high level view of the present prototype IC. The main components are a micro pipeline and two first order RC LPF. The micro pipeline circuit used was proposed by Charles E. Molnar in [16]. Each stage of the micro pipeline is composed of two sub-circuits, a data latch register (D0, D1, ...) and a control cell (A0, A1, ...). The data latch register is made up of n data latches in parallel. Each of these registers is controlled by a control cell. The bit-lines between the 1st and 2nd data stage are shifted up or down in order to obtain a 64 bit long circular single-bit bit-stream. Note that a



Figure 4.9 Bi-mode scan chain architecture

sufficient amount of buffering is added between the control logic and the registers to drive the large load. The output of the bit-stream is taken at one input of the multiplexer (MUX) and the other input is used to load the initial bit-stream into the pipeline.

The control cell is a latch which toggles between two states. It supplies two signals Enable (En) and Enable (En), which are the complement of one another. These signals are used to control the data latch register of which one of the n latches is shown in Figure 4.10. When En is high it causes the data latch to be transparent or empty. In other words it contains no data. When En is high it causes the data latch to be in the opaque or full state, thus it contains a data item. The logic function performed by the control cell is to toggle its state when the current stage is full and the next stage is empty. As the next stage toggles from empty to full it will effectively capture the data of the previous stage.

Figure 4.11 shows a novel implementation of the control cell which is used for this design. This novel design was introduced by Laberge and Negulescu in [17] and replaces the original design proposed in [16]. In order to achieve the fastest bit-rate possible, the micro-pipeline must be initialized properly. This is the reason for the C1, C2, Reset and Set inputs in Figure 4.11. The fastest bit rate achievable is when full and empty stages are



Figure 4.10 Data Latch



Figure 4.11 Novel Asynchronous Cell

interleaved. This is because a full stage needs to be followed by an empty stage in order to be able to make its data go forward. Thus, if every full stage is always followed by an empty one, all the data items will move together in the pipeline. Therefore, to maximize throughput, the control logic is initialized with alternate states, i.e., every other stage has set asserted while the others have reset asserted for initializing. The states of the control cells can be toggled one step at the time, using the C1 and C2 control lines, in order to load the bit-stream in the data path. Because of the presence of these two control lines, it is possible to run the micro pipeline in one of two modes, synchronous or asynchronous, i.e., using a non-overlapping clock for C1 and C2 will run the scan-chain synchronously and asserting both will let the scan-chain run asynchronously.

As can be seen in Figure 4.9, the output of the micropipeline is sent through an inverter resulting in an inverted and non-inverted bit-stream. These two bit-stream are then connected to two different LPF resulting in a differential output voltage. The filters are first order RC LPF. They both have an R = 520 K Ω and C = 5.46 pF. These values are chosen to provide enough attenuation to obtain 12-bit resolution at an F_s of 1 GHz.

The sizing and optimization procedure of the circuits in Figures 4.10 and 4.11 was the focus of the work presented in [17]. This work is repeated in Appendix A for the readers convenience.

Figure 4.12 shows a micrograph of this implementation in 0.35 μ m CMOS technology with the significant components labelled. When doing the layout special care was taken in minimizing the interconnect lengths. For this reason a dummy row of asynchronous cells was added in order to avoid one long feedback connection. This is illustrated in Figure 4.9.

Design B



This next design is used as part of a larger design, in particular a 2nd order multibit sigma delta modulator. The purpose of this DC voltage generator design is to realize a

Figure 4.12 Chip Micrograph of Bi-Mode Scan-Chain

3-bit digital to analog converter (DAC). Figure 4.13 illustrates how this 3-bit DAC fits in with the rest of the modulator.

The requirements for this DAC are that it generates 8 different voltage levels, that it be able to switch between DC levels at an effective rate of 3.072 MHz, and the output voltage be precise to a 12-bit accuracy. The use of a 7 bit long scan chain allows the following 8 normalized DC differential voltages to be generated: 0/7, 1/7, 2/7, 3/7, 4/7, 5/7, 6/7, and 7/7. The effective rate sets the maximum time for the DAC to produce a stable output. In the present case this must be within 325 nsec of the inputs of the DAC changing. Finally, the 12-bit accuracy requirement sets how much attenuation the filter must provide.

The implementation of choice for this application is the use of an asynchronous topology as mentioned in Section 4.1.2. The reason for using an asynchronous implementation is the clockless nature of such an implementation, and the high bit-rate which can be in the gigahertz range with the right technology. Contrarily to the design in Section 4.2.2, the state of the control cell (full or empty) is used to store the bit pattern instead of data latches. Figure 4.14 illustrates the different components making up the 3-bit DAC. This DAC works as follows. On the first phase of the "Modulator Clock", the "Digital Control Block" decodes its 3-bit input into one of 8 possible 7-bit PDM pattern



Figure 4.13 Signal Flow Graph of 2nd Order Multi-bit Modulator



Figure 4.14 3 bit DAC

using combinational logic. The pattern is then parallel loaded into the "Asynchronous Scan-Chain" which is let to free run. By the end of the first phase of the clock, the cycling bit-stream produces a stable DC reference voltage at the output of the "Analog Filter". During the second phase nothing happens as this is when the output is sampled by the rest of the system. Because of the high resolution and the fast convergence time requirement of this design, an active filter implementation is chosen. The LPF implemented in this IC is a 4^{th} order SAB low-pass filter, which is illustrated in Figure 4.15.

This design was implemented in a 0.25 μ m CMOP25 technology. Figure 4.16 shows a micrograph of the layout of the implemented 2nd order multi-bit modulator. The parts making up the 3-Bit DAC are highlighted as the "DC Generator" and "SAB Filter".



Figure 4.15 4th Order Low-Pass SAB Filter


Figure 4.16 Layout of 2nd Order Multi-bit Modulator

Design C

For the last implementation, the asynchronous scan-chain just introduced is expanded to generate more DC output levels. The goal of this design is to verify the performance and usability of a large asynchronous scan-chain on chip. The scan-chain must be sufficiently long so that it is possible to generate more than 2^{12} DC levels.

Figure 4.17 shows an overview of the main components which make up this prototype IC. The scan-chain is made up of 2^{12} asynchronous cells and is laid out in the pattern reflected by that figure. The asynchronous cell used in this design is depicted in Figure 4.18. It is essentially the same one that was used in the previous design with the addition of a D flip-flop. The reason is for it to be able to load the chain in series. The D flip-flops are used to load the initial state of each cell serially. When Load is asserted, the whole asynchronous scan-chain is loaded in parallel. The control line, C, controls whether the chain is free running or stopped. Thus for the chain to run asynchronously Load must be deasserted and C asserted.



Figure 4.17 Programmable Reference Implementation Overview



Figure 4.18 Asynchronous Cell

For this design, a divider is connected at the output of one stage in order to be able to determine the speed at which the scan-chain is cycling. This divider is made up of 8 toggle flip-flop in series and will divide the frequency of the bit-stream by 2^8 .

The LPF for this design is split in two. A sufficiently large resistor is integrated on chip, and the capacitor part of the RC filter is then comprised of the package pin itself and an external capacitor. The two resistor are 20 K Ω each, and the capacitor may be any off the shelve value as convergence time is not a concern for this design.

This design was implemented in a 0.18 μ m CMOS technology. Figure 4.19 shows a micrograph of the layout of the implemented asynchronous reference with the relevant parts highlighted.

4.3 - Summary

In this chapter a number of possible ways to implement the DC voltage generator of this thesis were presented. From the discussion of each, it is seen that the best



Figure 4.19 Layout of Asynchronous Reference

implementation will be highly application specific. In the second and last part, the designs which were designed and implemented for this thesis were presented in greater details. Under the first category, the designs are synchronous and require an on chip PLL to achieve a high bit-rate. As for the later category, the designs are asynchronous and do not need a clock. The experimental results for these designs will be presented in the following chapter.

Chapter 5 - Experimental Results

The designs implemented over the course of this thesis were introduced in chapter 4. In this chapter, the methodology and performance metrics used to test the designs will be explained in the first part, and the experimental results for these designs will be presented in the second part. The experiment performed will reflect the goals that were set forth for in the last chapter.

5.1 - Testing Methodology

In this section, the methodology and the performance metrics used to evaluate the different designs of this thesis are introduced. Unless otherwise stated all the testing in this chapter is carried out using a Teradyne A567 mixed signal tester. Also, all the circuits were mounted on a custom built two layer PCB to interface with the testing equipment. The different metrics used throughout this chapter are Differential Nonlinearity (DNL), Integral Nonlinearity (INL), AC ripple, Temperature Coefficient (TC), and power. What follows is a brief description of each metric and how they are obtained.

DNL provides a measure of the uniformity of the DAC step size between DAC codes in terms of one LSB [18]. First, a sequence of analog voltages, S(i), corresponding to each DAC codes is obtained experimentally using a multimeter. The following equation is then used to obtaining the DNL curve

$$DNL(i) = \frac{S(i+1) - S(i) - V_{LSB}}{V_{LSB}} \text{ LSBs},$$
(5.1)

where V_{LSB} is the gain of the best fit line for S(i). The INL curve gives a comparison of the actual DAC curve S(i) and the corresponding best fit curve normalized to one LSB as follows

$$INL(i) = \frac{S(i) - S_{REF}(i)}{V_{LSB}}.$$
(5.2)

These two measurement are useful in showing if a DAC performs to its claimed resolution, i.e., if it operates within a DNL and INL of +/-1/2 LSB.

As was claimed repeatedly over the course of this thesis, the AC ripple of the voltage generator should be less than +/- 1/2 LSB. This can be done by measuring the amplitude of the ripple in the steady state output of the voltages reference. One other method is to look at the spectral characteristic of the steady state DC voltage and measure the Signal to Noise plus Distortion Ratio (SNDR). This is done as follows

$$SNDR|_{dB} = 10\log\left(\frac{P_S}{P_{total}}\right) dB,$$
 (5.3)

where P_S is the power of the DC tone, and P_{total} is the total noise power. From this figure of merit an equivalent resolution may be derived in terms of number of bits as follow

$$n|_{equiv} = \frac{SNDR|_{dB} - 1.761 \text{ db}}{6.02}.$$
 (5.4)

The temperature coefficient gives a measure of how insensitive a voltage reference is to temperature variations. It is calculated using the following relationship in units of parts per million per degree Celsius ($ppm^{0}C$)

$$TC = \frac{(\Delta V)/(\Delta T)}{V_{ref}},$$
(5.5)

where V_{ref} is the programmed reference voltage and ΔV is the total voltage variation over the temperature range ΔT . The temperature characterization is performed using an apparatus which blows temperature controlled air through a glass bell, which is set atop the circuit to be tested. The temperature is regulated and measured using a thermocouple glued to the package of the circuit.

Finally, the last metric which is investigated for some of the designs is the power consumption. The approach used is to measure the total current drawn through the power supplies under various operating conditions.

5.2 - Voltage Generator Experimental Results

In this section a series of experiments are performed on both the synchronous and asynchronous voltage generators. The experimental results will be presented for the synchronous designs in the first subsection and for the asynchronous designs in the last subsection.

5.2.1 - Synchronous Voltage Generator

In Section 4.2.1 a prototype IC was designed with the goal of verifying the concepts of PDM and PWM. This IC integrated a PWM generator, a PDM generator, an RC filter, and a PLL as was depicted in Figure 4.3 on page 49. The PWM generator uses a counter based architecture, Figure 4.4 on page 49, and is connected to the RC filter. The PDM generator makes use of a 256 bit scan-chain, Figure 4.7 on page 52, and must be connected to an off-chip filter. The PLL which was part of another design on the same IC did not function properly, thus the results presented in this section are obtained using an external clock. The total area taken up by the PWM generator and RC filter is 0.13 mm², which is quite small. The area taken up by the PDM generator alone is the same (but without a filter).

For the first experiment, an external clock of 10 MHz and a coarse external low pass RC filter with a 3 dB cutoff frequency of ~10 MHz is used. With this setup the steady state output of both modulator is obtained. In Figure 5.1 these steady state outputs are

plotted on the same graph. Plot (a) is obtained when the PDM generator is used and plots (b) and (c) are obtained when the PWM generator is used. The first important observation to make is that for the same RC filter the DC voltage obtained is a lot less AC ripple when the PDM generator is used than when the PWM generator is used. This is expected from the theory set forth in chapter 2. Another interesting observation when looking at plots (b) and (c) is that one can see a ramping effect in the steady state output of the PWM modulator. This corresponds to the duty cycle of the bit-stream (shown under each plot for convenience).

The next experiment is to find the two RC time constants that will yield the same AC ripple for the PWM and PDM generator. The settling time for these two different time constants is plotted in Figure 5.2. From this figure it is seen that the settling time is much smaller for the PDM generator which is a consequence of its smaller RC, 68 μ sec. compared to 4 msec. for the PWM generator.

Figure 5.3 shows the output of the PWM and PDM voltage generators when they are programmed to generate 32 DC levels within a 3.3 Volts range. With the use of the external filter, both techniques were found to yield an 8-bit resolution. More specifically a



Figure 5.1 Output Voltage Using a Coarse External Filter for (a) PDM Generator, (b) PWM Generator with ~55 % duty cycle, and (c) PWM Generator with ~25 % duty cycle.

maximum DNL (INL) of 0.15 (0.4) LSB. This demonstrates a good linearity for both techniques.

The next set of experiments is intended to investigate the temperature characteristics of the voltage generator by using a scan-chain. Recalling the first design introduced in Section 4.2.2, it was a 64-bit scan-chain designed to operate in one of two modes, synchronous or asynchronous, and provided differential DC outputs as illustrated in Figure 4.9 on page 54. For the purpose of temperature characterization this scan-chain



Figure 5.2 Settling Behaviour of the On-Chip (a) PDM and (b) PWM Generators.



Figure 5.3 Outputs of the (o) PDM and (+) PWM Generators of the Prototype IC.

is used synchronously and a single ended voltage of 1.25 V is programmed and denoted V_{ref} in Eqn. (5.5).

In Figure 5.4 the temperature is swept from 100 $^{\circ}$ C to -5 $^{\circ}$ C then back to 100 $^{\circ}$ C. The voltage is measured from the 1.25 V single ended output. The results from this sweep demonstrate that there is essentially no hysteresis with this design and that the TC of this circuit is 288 ppm/ $^{\circ}$ C when taken single ended.

Next the temperature is swept from 0 °C to 100 °C with the voltage measured differentially. The single ended voltage is programmed to 1.025 V in order to obtain a differential voltage $V_{ref} = V_{ref+} - V_{ref-}$ equal to 1.25 V. This is performed at two different bit-stream frequencies (F_s), 2 MHz and 20 MHz, and the results are shown in Figure 5.5. It is seen that the TC is almost halved when the output voltage is taken differentially. Another important result to notice is that the voltage at one particular temperature changes with the bit-stream frequency. This opens up the possibility for compensating for the temperature drift through a simple change in the clock speed. Such an example is shown in



Figure 5.4 Single Ended Voltage Reference



Figure 5.5 Differential Voltage Reference

Figure 5.6, and demonstrates that a TC of less than 11 ppm/^oC may be achieved using this method.

Table 5.1 summarizes the temperature performance obtained with this voltage reference. It is seen that a TC comparable to a low order band-gap reference circuit is obtained without any use for compensation or calibration [2]. This is expected from the theoretical work on the underlying principles which is included in Appendix B. From this table, one also sees that by using a simple frequency compensation scheme it is possible to achieve a TC on the same order as state-of-the-art curvature-corrected bandgap reference circuit [19].

Voltage Reference Measurement Type	TC (ppm/°C)	Percentage Variation
Single Ended	288	1.1 %
Differential	165	0.6 %
Differential (Frequency Compensated)	11	0.02 %

Table 5.1 - Temperature Characterisation Results



Figure 5.6 Frequency Compensated Voltage Reference

With the previous experiments the concepts of PDM and PWM were verified and found to yield the linearity it was designed for. The results also showed that PDM puts less strict requirements on the filter bandwidth yielding a faster settling time as a consequence. Lastly the voltage generator was found to have good temperature characteristics.

5.2.2 - Asynchronous Voltage Generator

In this subsection a series of experiments is performed on the three asynchronous implementations which were introduced in Section 4.2.2. The goal of these experiments is to verify the usability and performance of asynchronous designs in generating DC voltage references.

The first implementation was meant to verify the use of an asynchronous micropipeline for the purpose of bit-stream generation. As depicted in Figure 4.9 on page 54, this design is made up of a set of data registers, a chain of asynchronous control cells, and a differential LPF. The total area taken up by this design is 0.2 mm². It is found that when loading the bit-stream and shifting it synchronously using an external clock, the voltage reference functions properly, hence its use in the last section as a synchronous scan-chain. However, when the bit-stream is shifted asynchronously the data gets corrupted. The cause of this problem is found to be the insufficient speed at which the data latches propagate the bits. In other words, the control logic, when running asynchronously, switches faster than the data latches can propagate the data. So the testing on this design yielded very little except motivating the new architecture used in the next two designs.

The next design, which is a modified version of the last one, was integrated as part of a 2nd order multi-bit sigma-delta modulator to serve the purpose of a 3-bit DAC. This DAC occupies an area of 0.65 mm². Some design for testability was included, such as onchip switches connecting the outputs of the DAC to pads. However, testing the DAC independently from the modulator was rather problematic. The main reason being that the switches used introduce some harmonic distortions, which greatly decrease the linearity of the measured voltages. Consequently, since the linearity of the DAC has to be comparable to that of the overall modulator [20], the performance of the modulator will be used to evaluate the performance of the DAC.

The testing of the modulator was performed by Naveen Chandra and is summarized in [21]. Figure 5.7 (a) and (b) shows the FFT of a 1957.03125 Hz sinewave input with an amplitude of 0.085 volts. It is apparent from this figure that the modulator performs the appropriate noise shaping. Next, the SNR and SNDR is obtained for different amplitude of the sinewave. This is seen in Figure 5.8. The dynamic range is measured to be 75 dB, with a peak SNR of 72 dB, and a peak SNDR of 67 dB.



Figure 5.7 Measured Results with 0.085V Amplitude 1957.0315 Hz Sinewave

From these figures of merit and Eqn. (5.4) it is seen that the modulator can achieve 10.8 bit of accuracy, it is then safe to assume that the DAC achieves at least that level of accuracy.

For this last set experiments, the goal is to verify the use of the asynchronous scanchain of the previous design to generate more DC levels. This design allows for more testability than the previous one. Looking back at Figure 4.17 on page 62 this design is



Figure 5.8 SNR/SNDR vs. Input Power

made up of a 4108 bit asynchronous scan-chain, a frequency divider, and a differential bitstream output going through resistors only. The total area occupied by this design is 2.6 mm^2

The main problem encountered with this design is its power consumption. Due to the nature of the asynchronous scan chain, power is used by a cell when it toggles state. Consequently, the power used by the whole scan-chain will depend on the number of transitions in the bit-pattern, i.e., the ratios of ones and zeros. The power consumption will be the lowest when there are only a few ones in the bit-stream, and the highest when the bit-stream is half ones and half zeros. The power is determined experimentally to vary between 200 mW and 2 W corresponding to a bit-stream with 1 one and a bit-stream with 1054 ones respectively.

A consequence of the high power consumption for certain bit-pattern is that the high amount of current being drawn through the power supplies causes a significant voltage supply drop at the asynchronous cell level. This drop in voltage supply has a direct consequence on the speed of the cell, which is proportional to the supply voltage. This means that as the voltage coded in the bit-pattern approaches 1/2 (1054 ones) the voltages supply drop will increase and the scan-chain speed will slow down. This is verified experimentally and the bit-rates corresponding to different DC levels are listed in Table 5.2. The DC levels in that table were chosen so that they divide evenly into the bit-stream length 4108. This is so that the pattern at output of the frequency divider of Figure 4.17 on page 62 is periodic, thus its frequency easily measurable.

DC Level (normalized)	Bit-Rate (GHz)
2/4108	2.2
4/4108	2.2
13/4108	2.1
52/4108	2.1
79/4108	2.2
316/4108	~1.8
1027/4108	~1.5

Table 5.2 - Scan-Chain Speed

In Section 5.2.2 it was observed that the output voltage varies as a function of the bit-rate. This fact is also true in the present design. So in order to obtain good performance, the DC levels should be generated at a similar and constant bit-rate. Otherwise, a significant amount of distortion will be introduced by the changing bit-rate. From Table 5.2 it is seen that this rate is relatively constant for the normalized voltages from 1/4108 to 79/4108. For higher normalized voltage the bit-rate slows down in addition to becoming variable, i.e., the frequency measured varies over time.

Figure 5.9 shows a plot of the INL and DNL measured for normalized voltages ranging from 0 to 283/4108. Looking at the DNL plot, it is clearly seen how the amplitude resolution of the voltage reference drops as the normalized voltage increase. For the range of voltage where the bit-rate remains relatively constant (codes 0-100) the voltage reference yields a 12 bit amplitude resolution. By the time the voltage increases to 283/4108 the amplitude resolution decreases to 9 bit. When the output voltage is measured differentially, the plots of Figure 5.10 are obtained for INL and DNL. There is no noticeable difference from the single ended case.



Figure 5.10 (a) INL and (b) DNL for Differential Voltage Measurements

From this set of experiment, it is seen that an asynchronous implementation works and can yield a good resolution and fast bit rates. There is however certain design issues such as power consumption which must be addressed before this approach is viable for generating a large number of DC levels.

5.3 - Summary

In this chapter the experimental results for the different designs implemented over the course of this thesis were presented. Table 5.3 summarizes the main performance metrics which were obtained. The experiments performed on the synchronous designs demonstrated the concepts of PDM and PWM. These designs were tested using a low frequency clock (F_s) as it had to be generated off chip. The use of a lower clock frequency had the effect of setting the fundamental frequency F_s/N_b lower in frequency. So a filter with larger RC was needed as can be seen from the example in Section 3.1. Due to this larger RC, the filter required a greater area and had to be implemented off-chip. With the asynchronous designs, the use of an off chip clock was no longer required and consequently the filters could all be implemented on chip due to their smaller area and the faster bit-rate (F_{s}). This allowed for better performance as less interconnects were present between the bit-stream generator and the filter. The downside was that for larger asynchronous designs, power consumption became a problem causing the circuit to malfunction. Finally, the asynchronous scan-chain shows promising results when used for short scan-chains and the synchronous approach when used for longer scan-chains. However, the synchronous approach would necessitate an on-chip PLL in order achieve the high bit-rate necessary to allow the filters to be integrated on chip, thus increasing the design complexity.

Design		TC (ppm/ ^o C)	Bit-Rate	Power (Watts)	INL (LSB)	DNL (LSB)	Resolu- tion (bits)
Synchronous	Counter Based	-	10 MHz	-	0.4	0.15	8
	Scan-Chain	164 or 11 ^a	2 - 20 MHz	17.4 m ^b	0.4	0.15	8
Asynchro- nous	3-bit DAC	-	-	-	-	-	10.8
	12-bit Ref- erence	-	1 - 2.2 GHz.	200 m - 2.2	0.8 - 2 ^c	0.4 - 3 ^c	12 - 9 ^c

 Table 5.3 - Experimental Results Summary

a. When Frequency compensation is used.

b. Extrapolated from standard cell specifications at a bit-rate of 20 MHz.

c. For the usable voltage range tested (0 - 283/4108).

Chapter 6 - Conclusion

This thesis presented a new technique for constructing a DC voltage reference. This technique made use of periodic bit-stream modulation in order to encode DC values in the average of the high and low bits of a bit pattern. It then made use of a low pass filter in order to extract this average and obtain an analog voltage reference. This technique used almost all digital logic, thus enabling it to benefit from the scaling trends in CMOS technology. Also, this reference was fully programmable making it a lot more versatile than traditional voltage references. In what follows is a summary and discussion of the different issues which were addressed over the course of this thesis.

6.1 - Summary and Discussion

In chapter 2, the two main kinds of bit-stream modulation, PWM and PDM, were presented. The PDM based approach proved to be the favoured technique. The reason being that in order to obtain a steady-state output with the same AC ripple, the filter required for PWM had a much smaller bandwidth than the one required for PDM. Consequently, the settling time was a lot less for the PDM bit-stream.

In Chapter 3, the reconstruction filter of the voltage generator was introduced. High-order filtering was shown to be the best approach in order to achieve a high amplitude resolution with a small settling time. As for the use of an active or passive filtering implementation, the latter showed a lot more promise. The reason being that only simple building blocks such as resistors and capacitors were required and that the implementation was shown to be immune to component mismatches and process variations. Moreover, the traditional complexity of designing such high-order passive RC filters was overcome with the introduction of a new synthesis method, thus making this type of filters easy to design.

The main question that was set forth in Chapter 4 was the use of synchronous versus asynchronous logic for implementing the voltage reference generator. Three main types of implementations were designed which fit into these two categories: a synchronous scan-chain, a synchronous hardware PWM generator, and an asynchronous scan-chain. More specifically, a synchronous scan-chain and a PWM generator were designed on the same IC in 0.35 μ m CMOS technology and three separate asynchronous scan-chains based implementation were designed on three separate IC's in 0.35 μ m, 0.25 μ m and 0.18 μ m CMOS technologies, respectively.

In Chapter 5 the experiments performed on the aforementioned designs and the results were discussed. The experiments performed on the synchronous scan-chain and PWM generator confirmed the conclusions reached earlier about PWM versus PDM. Also, temperature characterization experiments were performed on the synchronous scan-chain design and found to yield results comparable to a traditional bandgap voltage reference. Finally, the asynchronous designs were found to yield good performance but had some issues with power consumption when made too large. In retrospect, the synchronous designs appears to be more reliable for larger designs than the asynchronous design. More work is required to be done on the asynchronous scan-chain in order to make it usable for larger designs.

6.2 - Future Work

A number of aspect of this DC voltage generator still has the potential for further research. One of them is the FIR semi-digital filtering approach which was briefly introduced in Section 3.3. Also, the design of high-order filter using the synthesis method presented in Section 3.2.1 was only verified using simulation. Implementing such a filter

to replace the SAB filter which was used in Section 4.2.2 would yield valuable insight into this technique.

Another area of interest would be to investigate the integration of a hardware PDM modulator on-chip. For higher resolution application, such an implementation could greatly cut down on the area, and maybe even on the power consumption. It is however suspected that this would be at a cost in speed (bit-rate).

Finally, the results obtained with the asynchronous scan-chain were quite promising. Further investigation for a more compact and faster asynchronous structure could yield even better results as well as solve the problems associated with the power consumption of the larger design.

References

[1]	A. Annema, "Analog Circuit Performance and Process Scaling", <i>IEEE Trans. Circuits & Systems II: Analog & Digital Signal Processing</i> , Vol. 46, No. 6, pp. 711-725, June 1999.
[2]	G. A. Rincon-Mora, Voltage References From Diodes to Precision High- Order Bandgap Circuits, New York, John Wiley and Sons, 2002.
[3]	D. A. Johns, K. Martin, <i>Analog Integrated Circuit Design</i> , New York, John Wiley and Sons, 1997.
[4]	A. Pierazzi, A. Boni, C. Morandi, "Band-gap references for near 1-V operation in standard CMOS technology", <i>IEEE Conference on Custom Integrated Circuits</i> , pp. 463-466, 2001.
[5]	B. Dufort, G. W. Roberts, "On-chip analog signal generation for mixed- signal built-in self-test", <i>IEEE Journal of Solid-State Circuits</i> , Vol. 34, Issue 3, pp. 318-330, March 1999.
[6]	E. M. Hawrysh, G. W. Roberts, "An integration of memory-based analog signal generation into current DFT architectures", <i>IEEE Transactions on Instrumentation and Measurement</i> , Vol. 47, Issue 3, pp. 748 -759, June 1998.
[7]	G. W. Roberts, M. M. Hafed, S. Laberge, "Programmable DC Generator", US Patent 09/844,277, Filed April 2001.
[8]	X. Haurie, G. W. Roberts, "A multiplier-free structure for 1-bit high-order digital delta-sigma modulators", <i>Proc. of the 38th Midwest Symposium on Circuits and Systems</i> , Vol. 2, pp. 889-892, 1996.
[9]	A. S. Sedra, K. C. Smith, <i>Microelectronic Circuits</i> , New York, Oxford University Press, 1991.

- [10] S. Karni, *Network Theory: Analysis and Synthesis*, Allyn and Bacon Inc., 1966.
- [11] G. W. Roberts, *Linear and Nonlinear Analog Signal Processing*, pp. 24-34, Jan. 2001 Course Notes.
- I. W. Selesnick, M. Lang, C. S. Burns, "Constrained least square design of FIR filter without specified transition bands", *IEEE Trans. on Signal Processing*, Vol. 44, No. 8, pp. 1879-1892, Aug. 1996.
- [13] A. Aga, G. W. Roberts, "A CMOS digitally programmable current steering semidigital FIR reconstruction filter", *Proc. IEEE Symposium on Circuits and Systems*, Vol. 1, pp. 168-171, May 2001
- [14] C. Halper, M. Heiss, G. Brasseur, "Digital-to-analog conversion by pulsecount modulation methods", *IEEE Transactions on Instrumentation and Measurement*, Vol. 45, Issue 4, pp. 805-814, Aug. 1996
- [15] M. M. Hafed, S. Laberge, G. W. Roberts, "A Robust Deep Submicron Programmable DC Voltage Generator", *Proc. IEEE Symposium on Circuits and Systems*, Vol 4, pp. 5-8, 2000.
- [16] C. E. Molnar, I. W. Jones, W. S. Coates, J. K. Lexau, "A FIFO Ring Performance Experiment", *Third International Symposium on Advanced Research in Asynchronous Circuits and Systems*, pp. 279 -289, 1997.
- [17] S. Laberge, R. Negulescu, "An Asynchronous FIFO with Fights: Case Study in Speed Optimization". *Proc. IEEE International Conference on Electronics, Circuits and Systems*, Vol. 2, pp. 755-758, Dec. 2000
- [18] M. Burns, G. W. Roberts, An Introduction to Mixed-Signal IC Test and Measurement, New York, Oxford University Press, 2001.
- [19] G. A. Rincon-Mora, and P. E. Allen, "A 1.1 V Current-mode and piecewise-linear curvature-corrected bandgap reference", *IEEE Journal of Solid-State Circuits*, Vol. 33, No. 10, October 1998.
- [20] J. C. Candy, G. C. Temes. "Oversampling methods for A/D and D/A conversion". *Oversampling Delta-Sigma Data Converters: Theory, Design, and Simulation*, IEEE Press Collection of Papers, Piscataway, New Jersey, pp.1-25, 1992.
- [21] Naveen Chandra, "Top-Down Design Techniques for Delta-Sigma Modulators", *M. Eng. Thesis, McGill University*, 2001.

[22] I. E. Sutherland, B. Sproull, D. Harris, *Logical Effort: Designing Fast CMOS Circuits*, San Francisco, Morgan Kaufmann Publishers, 1999.

Appendix A - Logical Effort Based Optimization

Logical effort is a method to determine the optimum number of logic stages and the optimum transistor sizes for minimizing delays in a logic circuit. In this appendix an overview of the logical effort concepts that are used for optimizing the circuits of Chapter 4 is presented. For more details, the reader should refer to [22].

The method is based on the following delay model for a logic gate

$$d = f + p, \tag{A.1}$$

where f is the stage effort which varies with the size of the gate and p is the parasitic delay assumed to be constant. The stage effort is further described as

$$f = gh, \tag{A.2}$$

where g is the logic effort, which depends on the topology of the gate, and h is the electric effort

$$h = \frac{C_{out}}{C_{in}},\tag{A.3}$$

which relates the capacitance that loads the output of the gate, C_{out} , and the capacitance, C_{in} , presented by a single input terminal of the gate.

Logical effort is also applicable to optimizing multi-stage logic networks. The total effort of a path through the circuit can be expressed as

$$F = GHB \tag{A.4}$$

where G is the path logical effort, H is the path electrical effort, and B is the branching effort. G and H can be obtained from Eqns (A.2) and (A.3) as follows

$$G = \prod g_i \ H = \frac{C_{out}}{C_{in}},\tag{A.5}$$

where C_{out} and C_{in} correspond to the path in question rather than a single gate. The branching effort *B* is the product, for each node of the path, of the input capacitance of the next stage of the path over the total capacitance at that node of the path. When no fanning out is present, *B* equals one and may be ignored.

A main result of logical effort is that the optimum delay through the path will be achieved when each logic gates along the path share the same stage effort, f. Thus each gates may be sized using a stage effort of $F^{I/N}$, where N is the total number of stage in the path. The following equation is then used to size each of the logic gates,

$$C_{in_i} = \frac{g_i C_{out_i}}{\hat{f}} \tag{A.6}$$

So far it has been assumed that the logic gates are symmetric, i.e., that all the input transistors of the same type have the same sizes. However, it is possible to size a logic gate in such a manner that certain inputs will be sped-up while others slowed. In that case, different inputs have different logical effort. Asymmetric gates are treated in [22] by introducing a symmetry factor *s* in the logical effort calculation. It is also possible to favour rising input transitions vs. falling input transitions, and vice versa. This is handled in [22] by introducing the skew factor, γ , which is the ratio of PMOS width to NMOS width of a gate input. This factor affects the input threshold voltage at which a logic gate switches.

The speed at which the state of the control cell, which is repeated in Figure A.1 for convenience, toggles will set the maximum rate at which the micro pipeline can shift data, thus is it important to optimize this cell for maximum speed. For the control cell to function properly, the following delay constraint must be met. A rising transition at the



Figure A.1 Asynchronous Control Cell Optimization

gate of Tx1 must propagate faster around the loop G1-G2-G3-G4 in Figure A.1, than around G2a-G3a-G4a. This is to guarantee that the input of Tx1 and Tx1a are never high at the same time. It is possible, however, to make the longer chain respond faster, by noting that the delay constraint applies only for rising transitions at the gate of Tx1. For this purpose, skewed gates are used. For example, if "En" is asserted, it will deassert itself when the input of Tx1 goes high. To satisfy the delay requirement, a zero must propagate back to its input before a one has a chance to propagate to Tx1a. By using a skew factor γ =1 for G2 and G2a, it will increase the rise-time of the inverter therefore slowing the propagation of the one to Tx1a, but leaving the propagation of zero back to Tx1 unaffected.

The basic theory of logical effort is used to optimize all the logic in the micro pipeline (data latches, buffers, MUX), but it can not be applied directly to the control cell. The reason is the presence of fights between logic gates in the cell. It is however possible to separate the circuit into sub-blocks so that logical effort can be used for most of it. Referring to Figure A.1, all the circuit components outside the dashed box can be optimized according to logical effort. So the loop G2-G3-G4 may be sized using logical effort.

In Figure A.1, C1, C2, Set and Reset are control signals required to initialize and load the micro pipeline. They only come into play when initializing and loading the FIFO. The speed of these inputs is not critical as they will not be switching when the pipeline is in normal mode. It is then possible to make use of the symmetry factor s to favour the transitions along the critical path. This will have the effect of making the logical effort g along the critical path have a value closer to the ideal case, that of an inverter.

Here is the proposed optimization routine:

• Optimize G2-G3-G4 loop using logical effort.

- Make G2 the smallest the technology will allow, and set γ to 1 to meet the delay constraint.

- Use a symmetry factor, s, of 1/4 for G3 and G4 to favour the inputs on the critical path.

- Make G5 load the critical path by 3 gate widths, 2 for PMOS and 1 for NMOS (minimum size inverter).

- To size the logic gates along the loop, the value of 3.39 suggested in [22] for optimum stage effort can be used as a starting point. This value may be varied upon reiteration of the optimization routine.

- Using the size of G4 and the previous stage effort, the width of Tx1 is set to the maximum load G4 can drive (C_{out} in equation A.6).
- The NMOS in G1 serves the only purpose of providing a trickle of current to hold the current state of the control cell, thus may be made minimal.
- The PMOS in G1 is sized the following way:

- The circuit in the dashed box is simulated as a stand alone circuit.

- Fixing the width of Tx1 to the previously determined value, the width of the PMOS is swept while the propagation delay through the cell is measured (from 50% of input transition of Tx1 to 50% output transition of G2).

- This will yield the PMOS width with the smallest propagation delay as illustrated in

Figure A.2.

Note that the values for γ and *s* are given for illustrative purposes, and can be chosen differently depending on the speed and area requirements of the optimized circuits.

Finally, to optimize for the data latch logical effort is also used. In Figure A.3, the gate made up of m1 - m4 is set to minimum size as it is only meant to provide a trickle of current for the latch to hold its state when in the opaque state. The gate made up of m5 - m8 and the inverter can then be sized according to logical effort.





Figure A.3 Data Latch Optimization

Appendix B - Temperature Performance

Referring to chapter 3, one sees that the effect of the low pass RC filter on the bit stream is to extract the normalized area under the signal. This area will vary with temperature. Thus the voltage level resulting when this bit-stream is filtered will also vary with temperature. This effect can be explained in terms of the propagation delays through the bit-stream driver, an inverter. The low-to-high (t_{pLH}) , and high-to-low (t_{pHL}) propagation delays through an inverter increase with temperature. The rate of increase of the two delays will change with temperature and alter the area under the curve of the bit-stream, i.e., if t_{pLH} increases faster than t_{pHL} with temperature, then the area will decrease, and if the oposite happens, then the area will increase. See for instance Figure B.1.

From experimentation, a relationship between the size of an inverter and the variation in the propagation delays can be established. When the PMOS and NMOS are proportionally sized according to their respective mobilities, the difference between the



t_{pLH} increases faster than t_{pHL}



 t_{pHL} increases faster than t_{pLH}

Figure B.1 Area Under the Curve w/r to Propagation Delay

change in their delays is essentially constant and, hence, the area under the curve remains constant. However, if the aspect ratio of the PMOS is dominant, the high-to-low delay increases at a faster rate than the low-to-high delay, so the area under the curve will increase with temperature. The reverse happens when the aspect ratio of the NMOS is dominant. The effect of the different buffer sizes on the reference voltage is plotted in Figure B.2.

One approach to size the buffer would be to characterize the complete circuit as a function of temperature, and then design a buffer stage to compensate for temperature change. This could even be done using a binary-weighed output buffer to allow for post-processing calibration.

Also, it is important to note that the frequency of the bit-stream will have an impact on how much the reference voltage is affected by temperature variation. If the frequency is higher, the effect of temperature will be greater. This can be explained in terms of the ratio of the propagation delay time to the period of the bit stream. The smaller



Analog ground generation using different buffer sizing

Figure B.2 Voltage versus Temperature for Different Buffer Sizing

the ratio (the slower the frequency), the smaller the voltage variation due to temperature. This provides another control variable that can be used for post-processing calibration.