Part 1: Optimization of Digital Filters for Delta-Sigma ADCs

Part 2: Design of a 10-bit current steering DAC in 130nm CMOS

A Project Report

submitted by

VAZE SAGAR RAJIV

in partial fulfilment of the requirements
for the award of the degree of

BACHELOR OF TECHNOLOGY

DEPARTMENT OF ELECTRICAL ENGINEERING
INDIAN INSTITUTE OF TECHNOLOGY MADRAS.

May 2013

THESIS CERTIFICATE

This is to certify that the thesis titled **Part 1: Optimization of Digital Filters for Delta-Sigma ADCs; Part 2: Design of a 10-bit current steering DAC in 130nm CMOS**, submitted by **Vaze Sagar Rajiv**, to the Indian Institute of Technology, Madras, for the award of the degree of **Bachelor of Technology**, is a bona fide record of the research work done by him under our supervision. The contents of this thesis, in full or in parts, have not been submitted to any other Institute or University for the award of any degree or diploma.

Dr. Nagendra Krishnapura
Project Advisor
Associate Professor
Dept. of Electrical Engineering
IIT-Madras, 600 036

Place: Chennai
Date: 14th May 2013

ACKNOWLEDGEMENTS

I would like to start by thanking Lord Ganesha, the bestower of wisdom and remover of obstacles, to whom I owe my very existence. I would like to thank my parents, whose tireless efforts and unwavering support enabled me to become what I am.

I would like to express my gratitude to my project advisor, Dr. Nagendra Krishnapura for his continued guidance and support, not just during my project, but during the entire duration of my studies at IIT Madras. It was Dr. Krishnapura's excellent teaching of the course *Analog Circuits* which got me interested in this area in the first place. Learning the course *Analog IC Design* from Dr. Sankaran Aniruddhan was an amazing experience, and this provided me with a solid understanding of the principles of IC design. I would also like to thank Dr. Nitin Chandrachoodan, Dr. Shreepad Karmalkar and Dr. Shanthi Pavan, from whom I have learnt various skills which have been invaluable to me in the completion of this project.

I would like to thank Mrs. Janaki for her timely help in resolving any issues I faced in the lab. I would like to thank Debasish for the discussions we had, and for his help during my entire project.

ABSTRACT

KEYWORDS: Data Converters; Delta-Sigma ADCs; Digital Filters; Current Steering DACs

This project is divided into two parts. The first part deals with various techniques for optimizing the digital filter used in a dual-channel delta-sigma ADC. The performances of different alternatives to the raised cosine filter, are computed using MATLAB simulations. From these, a possible substitute for the raised cosine filter, which leads to an improvement in performance, is identified. Finally, it is attempted to eliminate the sample and hold circuit and the equalizer, by appropriately modifying the digital filter so as to handle continuous sampling of the analog input signal. To this end, time-domain optimization of the filter is carried out using linear programming.

The second part consists of the design of a current steering DAC in a 130 nm UMC CMOS process. The DAC is a 10-bit DAC with a 6+4 segmentation (6-bit thermometric and 4-bit binary sections). The static power consumption of the analog portion is 12 mW and the area of the analog portion is 0.065 mm^2. The DAC is tested and is found to work for sampling frequencies upto 3.3 GSPS. The SFDR is 58 dB for an input frequency of 20 MHz and 40 dB for an input frequency of 200 MHz.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABBREVIATIONS

ADC	Analog to Digital Converter
CCCS	Current Controlled Current Source
CMOS	Complementary Metal Oxide Semiconductor
DAC	Digital to Analog Converter
DNL	Differential Non Linearity
DSM	Delta-Sigma Modulator
FO4	Fan-Out-4
INL	Integral Non Linearity
ISI	Inter-Symbol Interference
LP	Linear Program
LSB	Least Significant Bit
MOSFET	Metal-Oxide-Semiconductor Field Effect Transistor
MSB	Most Significant Bit
NMOS	N-type Metal-Oxide-Semiconductor
NTF	Noise Transfer Function
PMOS	P-type Metal-Oxide-Semiconductor
PPD	Peak-to-Peak Differential
PVT	Process-Voltage-Temperature
SFDR	Spurious Free Dynamic Range
SNDR	Signal to Noise and Distortion Ratio
SNR	Signal to Noise Ratio
STF	Signal Transfer Function
UMC	United Microelectronics Corporation

Part 1

Optimization of Digital Filters for Delta-Sigma ADCs

CHAPTER 1

INTRODUCTION

Over the past several years, digital processing of signals has become exceedingly important. With the advent of CMOS technology, which is predominantly oriented towards the implementation of digital systems, digital circuits are finding their place in areas which have traditionally utilized only analog circuits. ADCs are one such area, and delta-sigma ADCs are a class of ADC which often utilize more of digital circuitry than analog [2]. Delta-sigma ADCs are now being used with signals of wide frequency ranges – from DC to several hundred megahertz.

1.1 Delta-Sigma Analog to Digital Converters

A delta-sigma ADC consists of an oversampling modulator and a digital and decimation filter. Figure 1.1 shows the basic structure of a typical delta-sigma ADC.

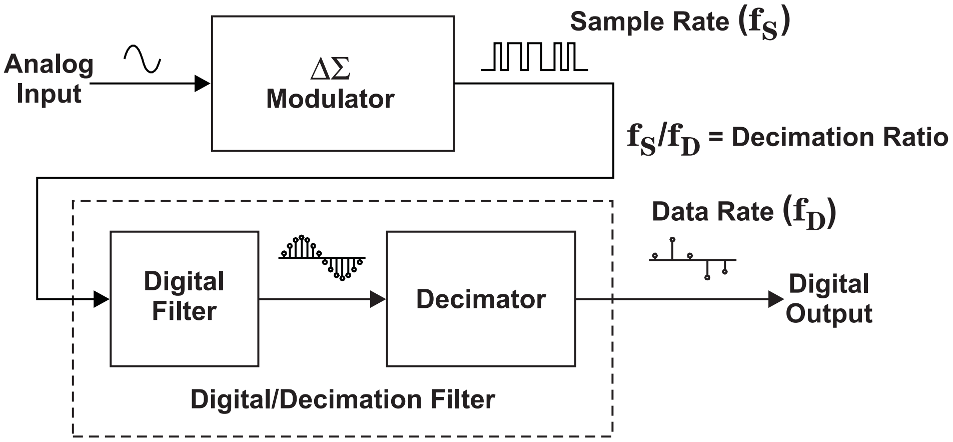


Figure 1.1: The basic architecture of a typical delta-sigma ADC. Usually, the digital and decimator filter blocks are combined into a single digital unit) [2].

A DSM is simply a 1-bit sampler. The sampling frequency of the DSM needs to be sufficiently larger than the frequency of the input analog signal – a technique known as oversampling. The digital/decimator filter operates at a frequency lower than that of the sampler. The digital filter averages the low-resolution, high-frequency output of the modulator into a signal of much higher resolution at a lower rate. Thus unlike most other data converters which operate at a single rate, a delta-sigma ADC is characterized by two rates, one is the sampling rate at the input and the other is the rate of the digital signal at the output. The ratio between the two is the decimation ratio, which strongly determines the effective resolution of the converter [2].

Figure 1.2 shows the structure of a first order DSM. An important characteristic of delta-sigma modulators is their noise shaping property. Delta-sigma modulators redistribute the noise at lower frequencies to higher frequencies. The NTF of a delta-sigma ADC is not flat. As a consequence of the averaging process, the noise at low frequencies (lower than f_S, the sampling frequency of the modulator), is largely removed from the output. On the other hand, the sampling process introduces quantization noise at high frequencies. Thus the NTF increases from a low value at DC and flattens out near f_S.

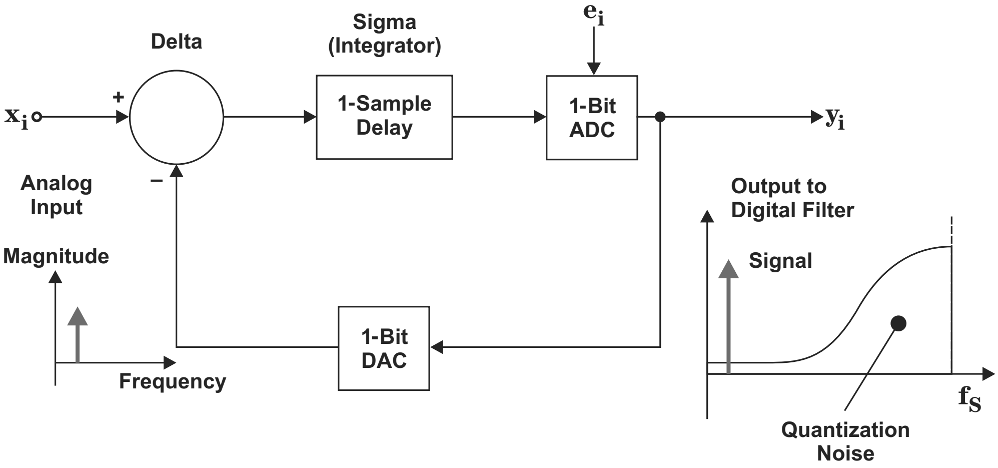


Figure 1.2: The structure of a first order DSM [2].

Besides the DSM, the other important component of a delta-sigma ADC is the digital/decimation filter, which is described in the following section. The digital filter is responsible for removing the out-of-band quantization noise introduced by the DSM, and thus the performance of the digital filter is strongly related to the SNR of the ADC [3].

1.2 The Digital/Decimation Filter

1.2.1 The Digital Filter

The digital filter is essentially a low-pass filter, which takes a weighted average of the output of the DSM. Delta-sigma ADCs generally incorporate a class of averaging filters called sinc filters (named due to their frequency response). Some delta-sigma ADCs for specialized applications (such as audio) use another filter in conjunction with the sinc filter, a technique known as two stage decimation.

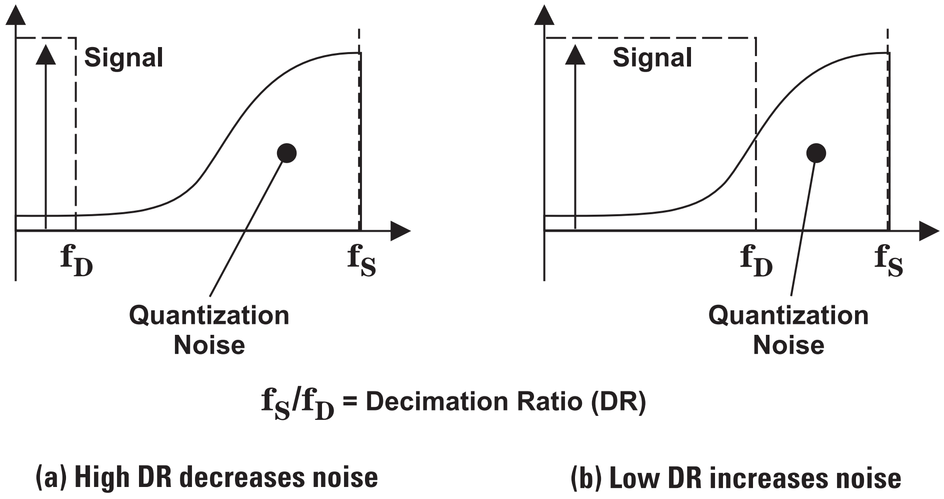
The output of the digital filter is at the same rate as the output of the sampler. In the time domain, the output of the digital filter is simply a digitized version of the original signal at a high sampling rate (equal to f_S). However in the frequency domain, the digital filter, being a low-pass filter, removes the quantization noise introduced by the sampler (essentially the digital filter averages the 1-bit sampled signal at f_S to clean up the high frequency components [3]).

The output of the digital filter is still at f_S. As mentioned previously, this rate is much higher than the frequencies of interest. Hence it is necessary to reduce the output rate of the digital filter. This is the done using the second part of the digital/decimator block – the decimation filter.

1.2.2 The Decimation Filter

The decimation filter is responsible for reducing the output rate of the delta-sigma ADC to reasonable levels. Most of the data present in the output of the digital filter is redundant, and in accordance with the Nyquist sampling theorem, it is possible to discard this redundant data, while preserving all the information in the original signal. This is exactly the function of the decimator.

The output rate of the decimator, $\mathrm{f_D}$, is the output rate of the delta-sigma ADC. The ratio $\frac{f_S}{\mathrm{f_D}}$ is the decimation ratio of the ADC, which strongly determines the SNR of the ADC, as shown in figure 1.3



(a) High DR decreases noise

(b) Low DR increases noise

Figure 1.3: The relation between noise and decimation ratio [2].

1.3 The Raised Cosine Filter

The raised cosine filter is a low-pass Nyquist filter. A Nyquist filter is a filter with vestigial symmetry i.e. the filter exhibits odd symmetry about $\frac{f_S}{2}$, where f_S is the frequency of the digital system. This is called the Nyquist ISI criterion, which can be stated in the time-domain as follows.

$$h(nT_S) = \begin{cases} 1; & n = 0 \\ 0; & n \neq 0 \end{cases}, \tag{1.1}$$

for all integers n, where $h(t)$ is the impulse response of the filter and T_S is the symbol period of the system. It can be shown that this time domain condition is analogous to the condition in the frequency domain [8]

The raised cosine filter is one of the simplest filters which satisfy the Nyquist criterion. This filter is widely used in communication systems due to its ability to minimize ISI. The name of this filter stems from the fact that for a particular choice of the parameter β (roll-off factor, defined later), the frequency response of this filter is identical to one period of a cosine shifted to lie above the f (horizontal) axis.

The raised cosine filter (and other ISI-free filters) are being considered in this work for use in a delta-sigma ADC, as they can filter the quantization noise without introducing any interference between samples. This enables their usage for multiplexed inputs, such as the ones being used in the dual-channel ADC under consideration.

The frequency domain description of the raised cosine filter is a piecewise function, given by.

$$H(f) = \begin{cases} T, & |f| \leq \frac{1-\beta}{2T} \\ \frac{T}{2}\left[1+\cos\left(\frac{\pi T}{\beta}\left[|f| - \frac{1-\beta}{2T}\right]\right)\right], & \frac{1-\beta}{2T} < |f| \leq \frac{1+\beta}{2T} \\ 0, & \text{otherwise .} \end{cases}$$

The filter is characterized by the parameter β, which is called the roll-off factor.

1.3.1 Roll-off factor

The roll-off factor is a measure of the excess bandwidth of the filter i.e. the bandwidth occupied beyond the Nyquist bandwidth of $\frac{1}{2T}$. $\beta = 0$ corresponds to an ideal (brick-wall) low-pass filter with its cutoff frequency equal to the Nyquist bandwidth. As the value of β is increased from 0 to 1, the roll-off of the filter becomes increasingly slower, and for $\beta = 1$, it approaches the *raised-cosine* waveform. Figure 1.4 shows the frequency response and impulse response of the raised cosine filter for various values of the parameter β.

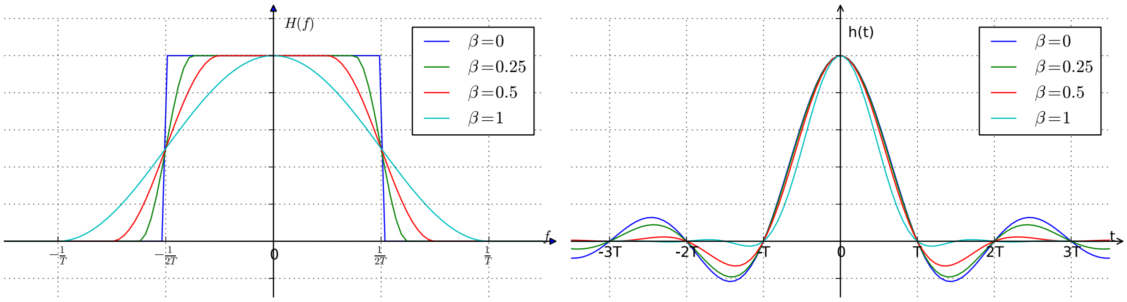


Figure 1.4: The frequency and impulse response of the raised cosine filter for various values of roll-off factor.

Several alternatives have been proposed to the raised cosine filter, such as [5], [7], [1], [11]. The performances of these alternatives vis-a-vis the raised cosine filter in a delta-sigma ADC are analyzed in the subsequent chapter.

CHAPTER 2

Performance of the Raised Cosine Filter and some Alternatives

In this chapter, the performances of the raised cosine filter and some other alternatives are analyzed. For the purpose of conciseness, the filters are henceforth denoted as follows.

Table 2.1: Filter pulse shapes under consideration.

Filter Pulse Shape	**Name Used in this Document**
Raised Cosine Filter	Rcosfir pulse
A "better than" Nyquist Pulse, [5]	Beaulieu pulse
A family of ISI-free polynomial pulses, [7]	Chandan pulse
Improved Nyquist Pulses, [1]	Fsech, Farcsech pulse
A Family of Pulse-Shaping Filters with ISI-Free Matched and Unmatched Filter Properties, [11]	Xia pulse

For the purpose of performance estimation, a dual-channel delta-sigma ADC is simulated in MATLAB using the *Delta Sigma Toolbox* [9]. Each of these filters is implemented for various lengths, and is used as the digital filter in the simulated delta-sigma ADC. The SNR is calculated for each of the filters for both IIR and FIR responses.

2.1 Performance Measures

2.1.1 SNR for Ideal (IIR) Implementations

The SNR for the IIR implementations of these filters is calculated by multiplying the NTF of the DSM with the frequency response of each of these filters. It must be noted that this is a highly ideal value as the effects of quantization and truncation are not included.

2.1.2 SNR and SNDR for Actual (FIR) Implementations

After the filters are implemented in the time domain for various lengths, they are convolved with the time domain sequence resulting from a simulation run of the DSM. Here a single sinusoid is applied as a test input. The SNR and SNDR values for various types of filters and for various filter lengths are then calculated.

2.1.3 Intersymbol Interference (ISI)

Theoretically these filters are ISI-free and thus the ISI should be zero. However due to finite resolution of quantization, there is some ISI in these implementations. For zero ISI, the Nyquist criteria says that the impulse response of the filter must go to zero at all multiples of the sampling time except for $T = 0$. Thus we want

$$h(nT_S) = \begin{cases} 1; & n = 0 \\ 0; & n \neq 0 \end{cases},$$

for all integers n, where $h(t)$ is the impulse response of the filter and T_S is the sampling time interval.

Since the practical realization of the filter may not satisfy the above relation, we can estimate the interference by calculating the sum of the impulse response values of the filter at integer multiples of the sampling time (except for T = 0). This would give the ratio of the amplitude of the desired symbol and the interference from the adjacent symbols, when a matched filter is used. This is the worst case value of ISI, when the interference from all the samples adds up.

$$\therefore \text{ISI} = \sum_{1}^{N} |h(nTs)| \,. \tag{2.1}$$

The ISI values are thus calculated for different filters of various orders using the above formula.

It must be noted that these values also depend on the machine precision of the system used for simulation. If the number of quantization bits of the filter are known, then a better measure would be calculating the ISI for a fixed number of quantization bits.

2.2 The Raised Cosine Pulse and some Alternatives

2.2.1 Rcosfir Pulse

The frequency domain description of the Rcosfir (raised cosine) pulse is the following.

$$H(f) = \begin{cases} T, & |f| \leq \frac{1-\alpha}{2T} \\ \frac{T}{2}\left[1 + \cos\left(\frac{\pi T}{\alpha}\left[|f| - \frac{1-\alpha}{2T}\right]\right)\right], & \frac{1-\alpha}{2T} < |f| \leq \frac{1+\alpha}{2T} \\ 0, & \text{otherwise} . \end{cases}$$

The value of the roll-off factor, α is chosen to be 0.5.

2.2.2 Beaulieu Pulse

The frequency domain description of the Beaulieu pulse is as follows.

$$H(f) = \begin{cases} 1, & 0 \leq f \leq B(1-\alpha) \\ \exp\left(\frac{\ln 2}{\alpha B}[f - B(1-\alpha)]\right), & B(1-\alpha) < f \leq B \\ 1 - \exp\left(\frac{\ln 2}{\alpha B}[B(1-\alpha) - f]\right), & B < f \leq B(1+\alpha) \\ 0, & \text{otherwise} , \end{cases}$$

where $B = \frac{f_S}{2}$ i.e. the Nyquist bandwidth. α is chosen to be 0.5.

The impulse response of the Beaulieu pulse is shown in figure 2.1 along with the impulse response of the Rcosfir pulse for comparison.

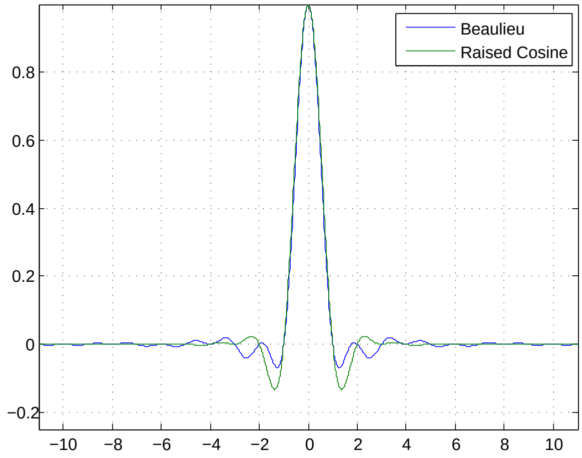


Figure 2.1: The impulse response of the Beaulieu pulse.

2.2.3 Chandan Pulse

The time domain description of the Chandan (polynomial) pulse is as follows.

$$h(t) = 3\text{sinc}(2Bt)\frac{\text{sinc}^2(\alpha Bt) - \text{sinc}(2\alpha Bt)}{(\pi\alpha Bt)^2} \; ,$$

where $B = \frac{f_S}{2}$ i.e. the Nyquist bandwidth. α is chosen to be 0.5. The impulse response of the Chandan pulse is shown in figure 2.2 along with the impulse response of the Rcosfir pulse for comparison.

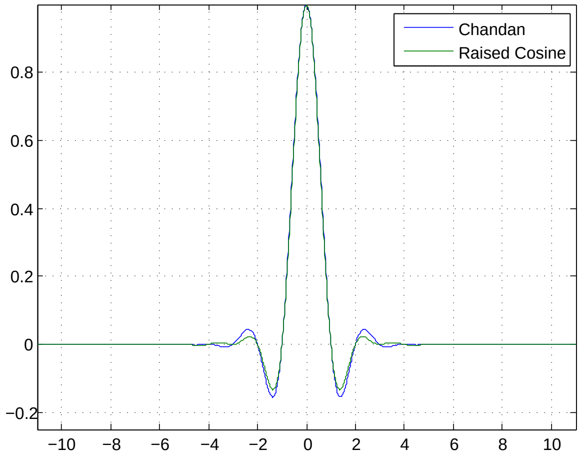


Figure 2.2: The impulse response of the Chandan pulse.

2.2.4 Fsech pulse

The frequency domain description of the Fsech pulse is as follows.

$$H(f) = \begin{cases} 1, & |f| \leq B(1-\alpha) \\ \text{sech}(\gamma(|f| - B(1-\alpha))), & B(1-\alpha) < |f| \leq B \\ 1 - \text{sech}(\gamma(B(1+\alpha) - |f|)), & B < |f| \leq B(1+\alpha) \\ 0, & \text{otherwise} \; , \end{cases}$$

where $B = \frac{f_S}{2}$ i.e. the Nyquist bandwidth. α is chosen to be 0.5. The impulse response of the Fsech pulse is shown in figure 2.3 along with the impulse response of the Rcosfir pulse for comparison.

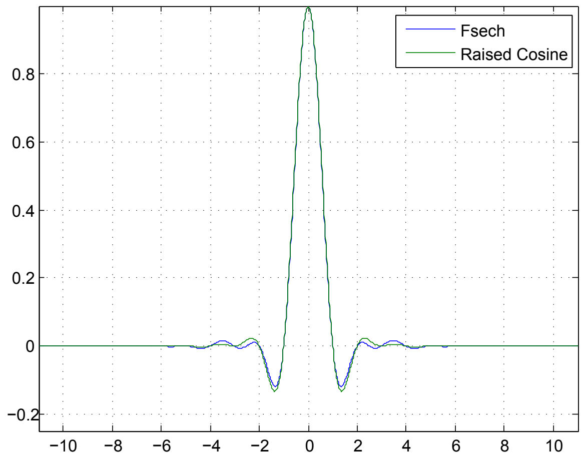


Figure 2.3: The impulse response of the Fsech pulse.

2.2.5 Farcsech pulse

The frequency domain description of the Farcsech pulse is as follows.

$$H(f) = \begin{cases} 1, & |f| \leq B(1-\alpha) \\ 1 - \frac{1}{2\alpha B\gamma}\text{arcsech}\left(\frac{1}{2\alpha B}(B(1+\alpha) - |f|)\right), & B(1-\alpha) < |f| \leq B \\ \frac{1}{2\alpha B\gamma}\text{arcsech}\left(\frac{1}{2\alpha B}(|f| - B(1+\alpha))\right), & B < |f| \leq B(1+\alpha) \\ 0, & \text{otherwise} , \end{cases}$$

where $B = \frac{f_S}{2}$ i.e. the Nyquist bandwidth. α is chosen to be 0.5. The impulse response of the Farcsech pulse is shown in figure 2.4 along with the impulse response of the Rcosfir pulse for comparison.

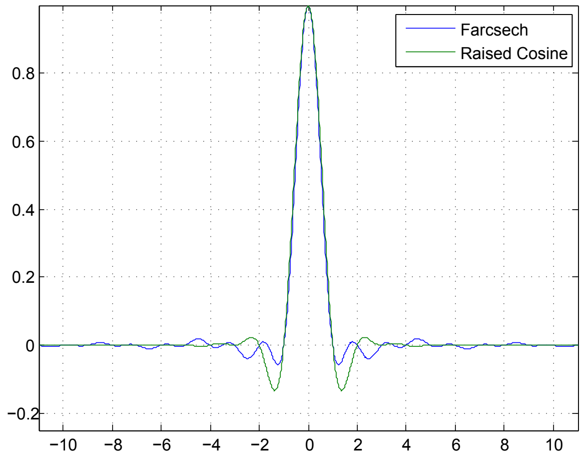


Figure 2.4: The impulse response of the Farcsech pulse.

2.2.6 Xia pulse

The frequency domain description of the Xia pulse is as follows.

$$H(f) = \begin{cases} 1, & |f| \le \frac{2}{3}B \\ \frac{1}{2}(1+\cos(\pi V(\frac{3Bf}{2}-1))), & \frac{2}{3}B < f < \frac{4}{3}B \\ 1-\frac{1}{2}(1+\cos(\pi V(\frac{3Bf}{2}-1))), & -\frac{4}{3}B < f < -\frac{2}{3}B \\ 0, & \text{otherwise ,} \end{cases}$$

where V is a polynomial function which controls the smoothness of the response. The constraints on V are as follows.

$$V(x) = \begin{cases} 0, & x \le 0 \\ 1, & x \ge 1\ . \end{cases}$$

and

$$V(x) + V(1-x) = 1, x \in \mathcal{R}\ .$$

The particular $V(x)$ chosen here is

$$V(x) = \begin{cases} 0, & x \le 0 \\ x^4(35 - 84x + 70x^2 - 20x^3), & 0 \le x \le 1 \\ 1, & x \ge 1\ . \end{cases}$$

The impulse response of the Xia pulse is shown in figure 2.5 along with the impulse response of the Rcosfir pulse for comparison.

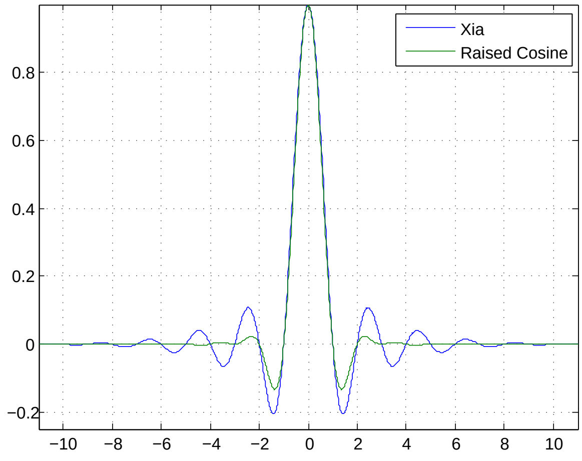


Figure 2.5: The impulse response of the Xia pulse.

2.3 Simulation Results

2.3.1 SNR for Ideal (IIR) Implementations

The SNR for the IIR implementations are calculated for each of the filters and are tabulated below.

Table 2.2: SNR for IIR Implementations.

Filter	SNR
Raised Cosine	108.50 dB
Beaulieu	108.28 dB
Fsech	108.46 dB
Farcsech	108.22 dB
Chandan	108.38 dB
Xia	108.69 dB

From table 2.2, it can be seen that without truncation and quantization, there is hardly any difference in the SNRs for the various filters.

2.3.2 SNR and SNDR for FIR Implementations

The SNR and SNDR values for the FIR implementations of each of the filters are calculated for various truncation lengths and are tabulated in tables 2.3 and 2.4. The truncation length of the filter is equal to $2L \times OSR + 1$, where L is the number of input symbols and OSR is the oversampling ratio (64). It can be seen that the Chandan pulse shape gives the best performance for moderate truncation lengths. For large truncation lengths, the performances of the Xia and Beaulieu pulse shapes are comparable.

For the pulse shapes which are characterized by a roll-off factor, figure 2.6 gives the variation of the SNR with the roll-off factor.

2.3.3 Inter-Symbol Interference (ISI)

The values of ISI are calculated using (2.1) and are tabulated in table 2.5. It can be seen that the Chandan pulse shape gives the best performance in terms of ISI.

Table 2.3: SNR for the simulated DSM.

Filter	SNR			
	$L = 1$	$L = 3$	$L = 5$	$L = 7$
Raised Cosine	105.54 dB	110.27 dB	108.84 dB	110.42 dB
Beaulieu	105.46 dB	108.43 dB	108.14 dB	108.30 dB
Chandan	103.72 dB	110.43 dB	111.28 dB	112.11 dB
Fsech	105.28 dB	109.33 dB	109.93 dB	110.17 dB
Farcsech	106.07 dB	107.57 dB	108.24 dB	107.26 dB
Xia	103.03 dB	110.67 dB	109.76 dB	111.05 dB
Best Performance	Farcsech	Xia	Chandan	Chandan
Filter	$L = 9$	$L = 11$	$L = 13$	$L = 15$
Raised Cosine	110.72 dB	110.32 dB	109.85 dB	110.33 dB
Beaulieu	108.21 dB	109.54 dB	109.05 dB	109.37 dB
Chandan	111.82 dB	110.04 dB	110.55 dB	111.98 dB
Fsech	110.44 dB	110.24 dB	110.59 dB	110.70 dB
Farcsech	107.25 dB	106.68 dB	108.73 dB	109.16 dB
Xia	110.72 dB	111.17 dB	111.49 dB	108.13 dB
Best Performance	Chandan	Xia	Xia	Chandan

Table 2.4: SNDR for the simulated DSM.

Filter	SNR			
	$L = 1$	$L = 3$	$L = 5$	$L = 7$
Raised Cosine	24.96 dB	76.38 dB	96.13 dB	103.51 dB
Beaulieu	28.56 dB	55.58 dB	67.28 dB	80.85 dB
Chandan	24.97 dB	61.71 dB	98.54 dB	102.32 dB
Fsech	26.16 dB	62.64 dB	80.27 dB	86.43 dB
Farcsech	29.49 dB	59.06 dB	64.67 dB	80.03 dB
Xia	23.70 dB	47.80 dB	56.40dB	66.04 dB
Best Performance	Farcsech	Raised Cosine	Chandan	Raised Cosine
Filter	$L = 9$	$L = 11$	$L = 13$	$L = 15$
Raised Cosine	106.93 dB	106.65 dB	106.32 dB	106.67 dB
Beaulieu	86.54 dB	92.46 dB	108.10 dB	107.98 dB
Chandan	107.44 dB	106.54 dB	106.43 dB	105.66 dB
Fsech	83.85 dB	83.98 dB	86.77 dB	88.53 dB
Farcsech	74.61 dB	76.47 dB	78.23 dB	89.38 dB
Xia	68.97 dB	87.75 dB	93.31 dB	97.56 dB
Best Performance	Chandan	Raised Cosine	Beaulieu	Beaulieu

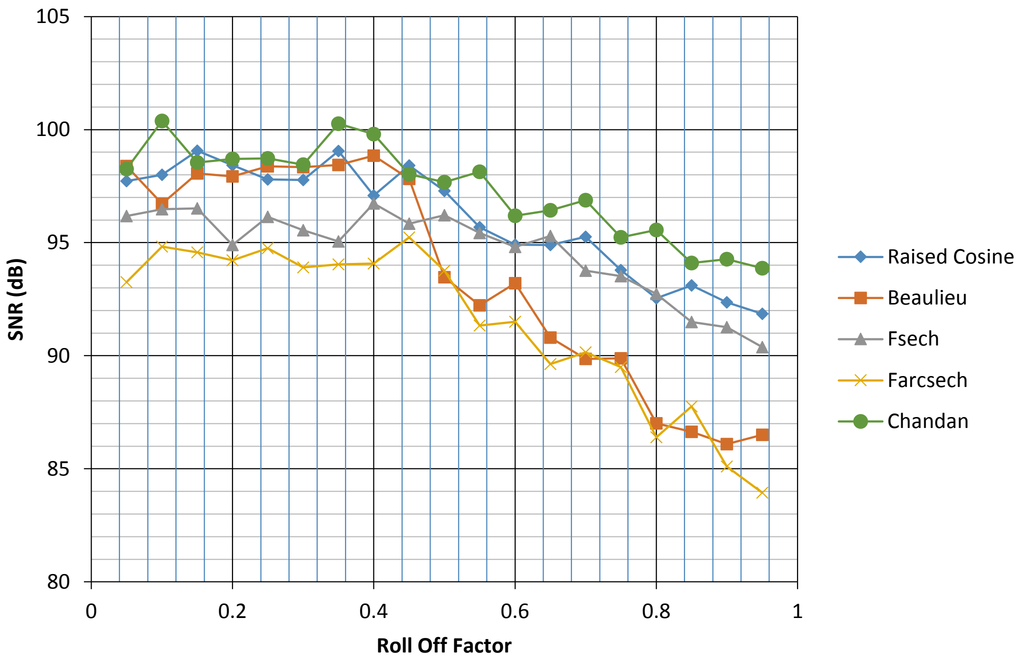


Figure 2.6: The variation of SNR with the roll-off factor for various pulse shapes.

Table 2.5: Practically Realizable ISI values.

Filter	ISI				
	L=3	L=5	L=7	L=9	L=11
Raised Cosine	5.9085e-010	1.1900e-009	1.9863e-009	2.1867e-009	2.3719e-009
Beaulieu	2.7103e-010	4.2105e-010	1.0927e-009	1.2122e-009	1.4772e-009
Fsech	6.4887e-010	4.7082e-009	6.3760e-009	7.4696e-009	7.8703e-009
Farcsech	1.4156e-009	2.8787e-009	4.2040e-009	5.6472e-009	7.0239e-009
Chandan	5.8554e-017	5.9274e-017	5.9927e-017	6.0069e-017	6.0468e-017
Xia	2.0861e-010	4.6564e-010	4.9046e-010	5.3698e-010	5.5954e-010
Best Performance	Chandan	Chandan	Chandan	Chandan	Chandan

2.4 Conclusion

From the various plots and tables in the previous section, it can be seen that most of the proposed replacements for the raised cosine pulse shape perform equally or worse than it. However the ISI-free polynomial pulse described in [7] (Chandan pulse) performs better than the raised cosine pulse in most simulation cases. Hence it could be a viable replacement for the raised cosine digital filter for use in the ADC.

CHAPTER 3

Digital Filter Optimization using Linear Programming

For the dual-channel ADC under consideration, it was observed that the sample and hold circuit was consuming excessive amounts of power [6]. The presence of the sample and hold circuit made it necessary to use an equalizer (in order to compensate for the distortion due to the sample and hold block). The presence of the sample and hold and equalizer distorts the response of the digital filter, so that the Nyquist zero-ISI criterion (1.1) is no longer satisfied.

Theoretically, it is possible to eliminate with the sample and hold block altogether by suitably modifying the digital filter to handle the sampling of a continuous analog signal. This requires imposition of additional constraints in the time-domain, which are outlined below.

3.1 Digital Filter in the Absence of Sample and Hold

If a sample and hold circuit is not used, and the analog input signal is continuously sampled into the ADC, a modification is required in the Nyquist zero-ISI criterion (1.1). Instead of having a single zero at intervals of OSR (where OSR is the oversampling rate of the delta-sigma ADC), we now require that the filter have M consecutive zeros (where M is the number of input samples to be averaged by the filter – analogous to the hold period of the sampler) after every $OSR - M$ non-zero samples. The central M samples of the filter must be all ones, as we want the gain of the currently sampled symbol to be maximum in comparison with the other symbols (assuming the filter impulse response is normalized).

The new constraints on the digital filter in the time domain are

$$h[n] = \begin{cases} 1; & 0 \leq n < M \\ 0; & kOSR \leq n < kOSR + M,\ k \in \mathcal{Z}/\{0\} \end{cases}, \tag{3.1}$$

In the frequency domain, the digital filter should be a low-pass filter with a cutoff frequency of $\frac{1}{OSR}$. The constraints in equation (3.1) are represented graphically in figure 3.1.

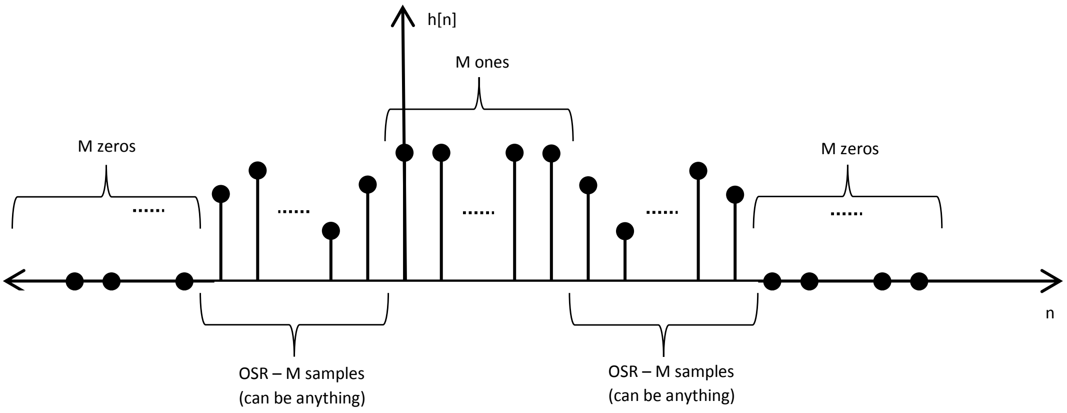


Figure 3.1: The time domain constraints (3.1) on the digital filter in the absence of an equalizer.

One of the ways to solve such a filter design problem, with constraints in both the time domain and the frequency domain, is to use linear programming as described in the next section.

3.2 Digital Filter Design using Linear Programming

A linear program is a problem in optimization theory having the following form.

$$\begin{aligned} \text{Minimize} \quad & c'x \\ \text{such that} \quad & Ax = b \\ & x \geq 0\,, \end{aligned}$$

where c and x are vectors of size n, b is a vector of size m and A is a matrix of size $m \times n$. The function to be minimized $c'x$ is called the cost function of the program. n is the number of independent variables and m is the number of independent constraints.

3.2.1 Filter Design as a Linear Program

The problem of minimizing the weighted error in an FIR filter,

$$E(\omega) = W(\omega)(A(\omega) - D(\omega)) \ ,$$

where, for a type I FIR filter,

$$A(\omega) = \sum_{n=0}^{M} a(n)cos(n\omega) \ ,$$

can be formulated as a linear program in the following manner [10].

$$\begin{array}{rll}
\text{Minimize} & \delta & \\
\text{over} & a(n), \delta & \\
\text{such that} & A(\omega) - \dfrac{\delta}{W(\omega)} \leq D(\omega) & \text{for } \omega \in B \\
\text{and} & -A(\omega) - \dfrac{\delta}{W(\omega)} \leq -D(\omega) & \text{for } \omega \in B \\
where & B = \{w \subset [0, \pi] | W(\omega) > 0\} &
\end{array}$$

The variables in this formulation are $a(0) \ldots a(M),$ and δ. As the cost function and constraints are linear functions of the variables, this is a linear program. However in order to solve this problem on a computer, it is necessary to discretize the frequency variable (ω), which has been taken to be a continuum in this formulation. This is done by dividing the ω–axis into a suitably dense grid of points.

The solver for this linear program is implemented in MATLAB, and the impulse responses for various lengths are obtained. For the purpose of simulation, the ratio $\frac{M}{OSR}$ is taken to be $\frac{1}{2}$. Figures 3.2, 3.3, 3.4 show the frequency responses of the filters for $M = 5,\ M = 7,$ and $M = 25$ respectively. Figures 3.5, 3.6, 3.7 show the corresponding impulse responses.

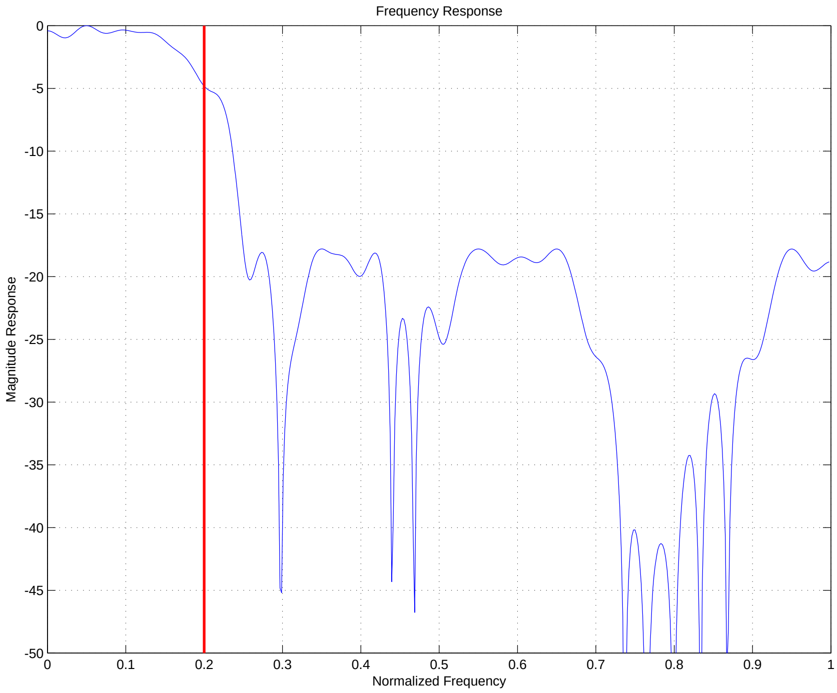


Figure 3.2: Frequency response for the filter designed using linear programming for $M = 5$.

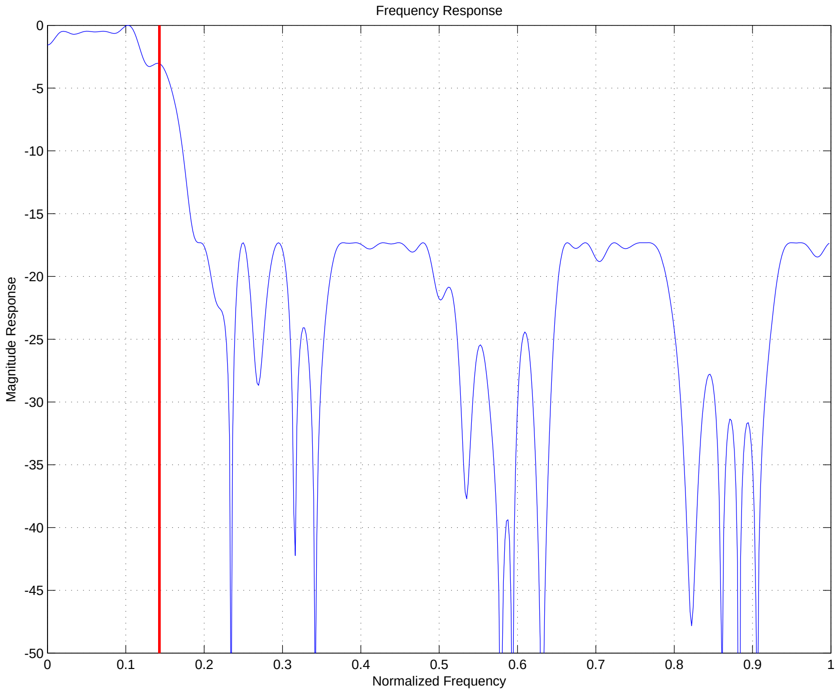


Figure 3.3: Frequency response for the filter designed using linear programming for $M = 7$.

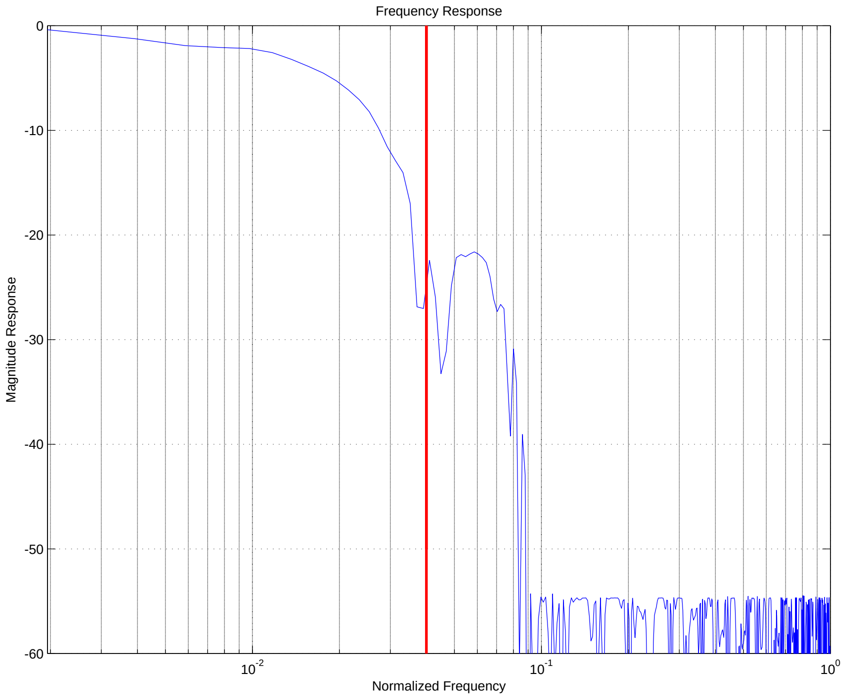


Figure 3.4: Frequency response for the filter designed using linear programming for $M = 25$.

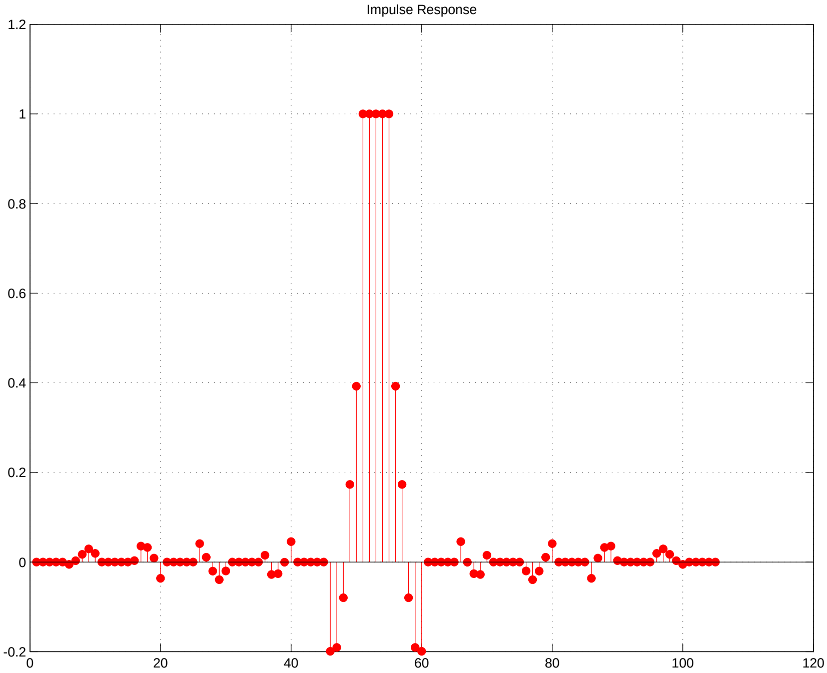


Figure 3.5: Impulse response for the filter designed using linear programming for $M = 5$.

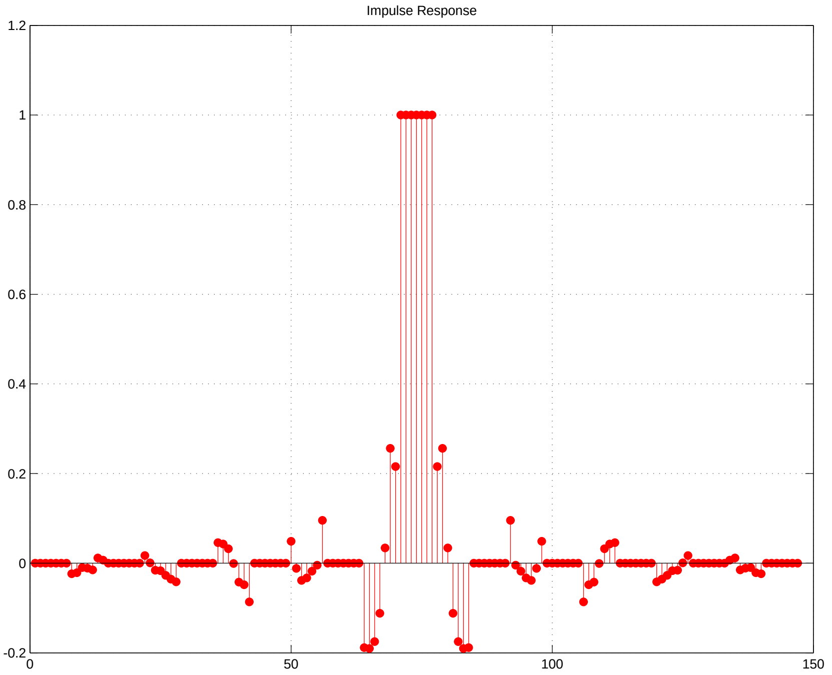


Figure 3.6: Impulse response for the filter designed using linear programming for $M = 7$.

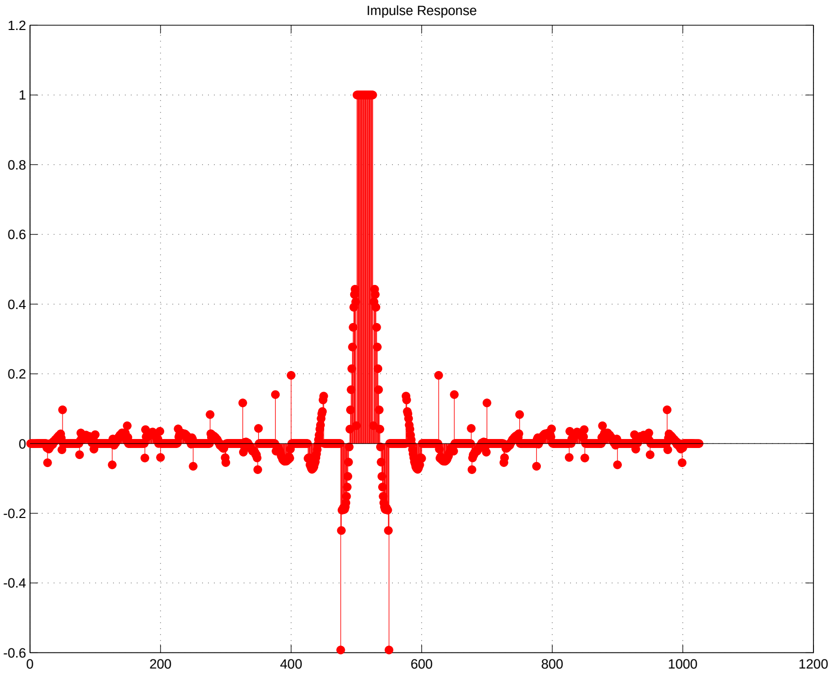


Figure 3.7: Impulse response for the filter designed using linear programming for $M = 25$.

3.3 Performance Analysis of the Designed Filter

From the frequency and impulse response plots in the previous section, it can be seen that linear programming fails to give satisfactory results for the filter to be designed. This is not a deficiency of the linear programming technique of filter optimization but is due to the constraints given in (3.1).

A discrete time sequence can be considered as a sampled version of a continuous time sequence. Let $h[n]$ be the discrete time sequence and $x(t)$ be the corresponding continuous time version. For $h[n]$ to be identically zero for M samples (from $kOSR$ to $kOSR + M$), $x(t)$ must be zero for a length of MT_S (from $(kOSR)T_S$ to $(kOSR + M)T_S$)[1], where T_S is the sampling period.

If $x(t)$ is an analytic function, it can be shown that the zeros of $x(t)$ are isolated or $x(t)$ is zero everywhere. Here $x(t)$ cannot be an analytic function, and the roll-off of the Fourier coefficients of $x(t)$ is slower than $\frac{1}{k^{m+1}}$, where $x(t) \in C^m$, and the Fourier coefficients are $X[k]$. The Fourier coefficients of the discrete sequence are the Fourier coefficients of the continuous time repeated periodically. Hence the Fourier coefficients of $h[n]$ will not decay any faster than $\frac{1}{k^{m+1}}$ and the performance of a filter satisfying (3.1) will be poor.

3.4 Conclusion

As shown in the previous section, the performance of a digital filter satisfying (3.1) is poor. Hence it is not possible to eliminate the sample and hold circuitry by simply designing an M-zero filter (3.1). Other techniques for filter optimization need to be investigated and possible architectural changes in the ADC are needed before the sample and hold circuitry can be removed.

[1] It is possible that $x(t)$ is only zero at the sampling instances $\{(kOSR)T_S,\ (kOSR + 1)T_S,\ \ldots\ (kOSR + M)T_S\}$ and is non-zero at intermediate values, however this would mean that the sampled spectrum of $x(t)$ has suffered aliasing – since a non-zero function (from $(kOSR)T_S$ to $(kOSR + M)T_S$) is mapped to a set of zeros (from $kOSR$ to $kOSR + M$) after sampling, which is not the case

Part 2

Design of a 10-bit current steering DAC in 130nm CMOS

CHAPTER 4

Introduction

The current steering architecture is commonly used for realizing high-speed DACs, often used in communication systems. At a fundamental level, a current steering DAC simply consists of a number of unit current cells connected in parallel. Each current cell consists of a current source in series with a switch, as shown in figure 4.1. By appropriately switching the individual current cells, the desired variation in the output current is obtained. This output current is commonly driven through a resistive load to obtain an output voltage.

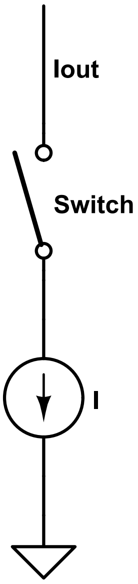


Figure 4.1: A unit current cell used in a current-steering DAC.

4.1 Binary versus Thermometric Architecture

There are two commonly used choices for the values of current supplied by the unit cells. Let I be the LSB current supplied by the DAC. Then each the current cells may supply I, $2I$ and so on upto $(2^n - 1)I$ for an n–bit DAC as shown in figure 4.2.

This is called binary encoding. Binary encoding in an n bit DAC requires n current cells. The number of current cells being linear in the DAC bits, binary encoding is very efficient in terms of decoding complexity.

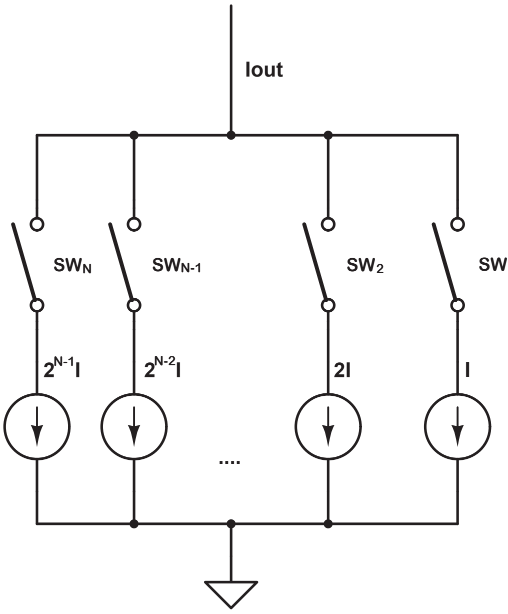


Figure 4.2: Binary encoded current-steering DAC.

Instead of having binary weighted current cells, each of the current cells can be equally sized to supply the current I as shown in 4.3. In this case, $2^n - 1$ current cells are needed. This is called thermometric encoding.

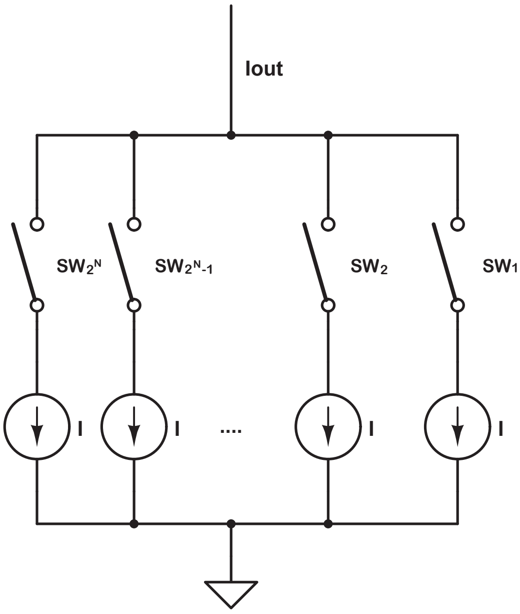


Figure 4.3: Thermometer encoded current-steering DAC.

Unlike binary encoding, the number of unit cells in a DAC grow exponentially with the number of DAC bits. The decoding circuitry for the thermometric encoding is thus more complex than for binary encoding.

The binary coding however does suffer from a major drawback in comparison with thermometric coding. In binary coding, whenever there is a transition, owing to mismatch in the cells, the cells may not all switch simultaneously. Thus there may be a momentary glitch in the output.

For a binary coded sequence, the maximum glitch occurs for a mid code transition i.e. for a $\{10\ldots0\} \rightarrow \{01\ldots1\}$ transition. It may so happen that the MSB cell switches of only after all other cells have turned on or vice-versa. Thus the maximum value of this glitch is equal to $\frac{I_{FS}}{2}$, where I_{FS} is the full-scale current of the DAC. On the other hand, the thermometric code is monotonic (for any transition, a number of cells may switch on or off but not both simultaneously). Hence thermometric encoding does not suffer from such glitches unlike binary encoding.

In practice most DACs are neither fully thermometric nor fully binary but a combination as both (known as segmented DACs). This is done to strike a balance between performance (from a thermometric section) and low complexity (from a binary section). Thus the MSB bits of the DAC, which are responsible for driving large currents are implemented using thermometer coding and the LSB bits are implemented using binary coding. The maximum glitch in this case is much smaller than a fully binary DAC. Generally the segmentation is chosen to have roughly the same number of bits in the thermometric and binary sections. In the design described in this document, a 6+4 segmentation is chosen (6-bit thermometric section and 4-bit binary section). For such a DAC, the maximum glitch is equal to the value of a single thermometric unit cell i.e. $\frac{I_{FS}}{2^6}$ for a 6+4 segmented DAC.

CHAPTER 5

Current Cell Design

Figure 5.1 shows a the current cell used in the DAC. Transistors M1 and M2 together form a cascoded current source. Besides offering a higher output impedance than a single MOSFET, the use of a cascode also helps reduce the capacitance at the common node. M1 needs to be sized according to matching constraints described later in this chapter. M2 is a current controlled current source (usually referred to as a cascode device), and can be minimum sized, reducing the capacitance seen at the tail node.

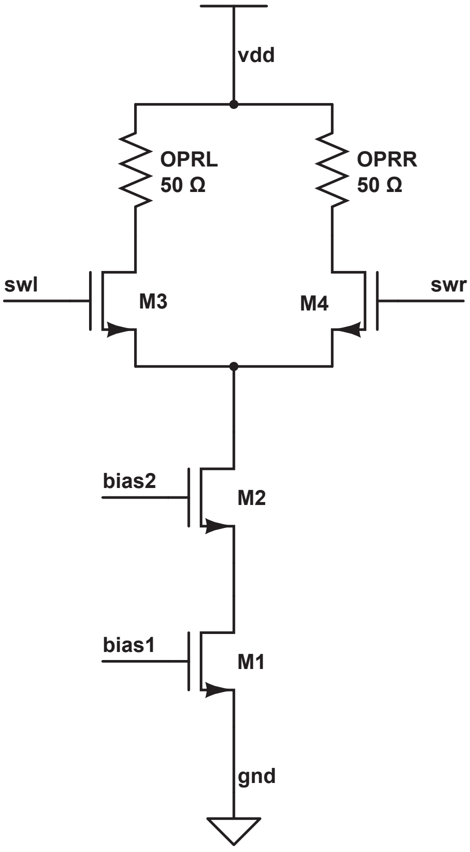


Figure 5.1: The basic current cell used in the DAC.

The use of a differential pair is necessitated to avoid turning off transistor M1 (which is large sized due to matching constraints, and thus turning it on/off would require the charging/discharging of a large capacitance), when turning off the cell. Hence a differential output is used and the current from M1 is driven into either OPRL or OPRR by controlling M3 and M4.

The load resistances OPRL and OPRR are shared by all the current cells. For this design, approximately 0.5 V PPD swing is accommodated for load resistances of 50 Ω. This gives a full scale current of approximately 10 mA, Based on this guideline, I_{FS} is fixed at 10.24 mA, which gives I_{LSB} as 10 μA

5.1 Sizing of Current Source based on Mismatch Constraints

The primary source of glitches in the DAC is due to mismatch in the current source transistors (M1). Let the LSB current in a unit cell be described by a normal distribution,

$$I_0 \sim N(I, \sigma_I)\ .$$

Then the subsequent currents can be written as

$$\begin{aligned}
I_1 &\sim N(2I, \sqrt{2}\sigma_I) \\
I_2 &\sim N(4I, 2\sigma_I) \\
I_3 &\sim N(8I, 2\sqrt{2}\sigma_I) \\
I_4 &\sim N(16I, 4\sigma_I) \\
&\vdots \\
I_{67} &\sim N(16I, 4\sigma_I)
\end{aligned}$$

$$.$$

The maximum glitch occurs when all the binary cells switch off and a thermometric cell switches on or vice-versa. The distribution of the current due to this glitch is given by

$$\begin{aligned}
DNL_{MAX} &= I_4 - (I_3 + I_2 + I_1 + I_0) \\
\therefore DNL_{MAX} &\sim N(I, sqrt(31)\sigma_I) \qquad (5.1)
\end{aligned}$$

We need to ensure that the size of the current cell transistor is chosen so that this DNL falls within acceptable limits.

Here we assume the following constraint.

$$\begin{aligned} 3\sigma_{DNL} &< \frac{LSB}{2} \\ \therefore 3\sqrt{31}\sigma_I &< \frac{I}{2} \\ \therefore \frac{\sigma_I}{I} &< \frac{1}{33.40} \end{aligned}$$

For a sufficient margin, we chose the following constraint.

$$\frac{\sigma_I}{I} < \frac{1}{50} \tag{5.2}$$

From [4] we have,

$$(WL)_{min} = \frac{1}{2}\left[A_\beta^2 + \frac{4A_{VT}^2}{(V_{GS}-V_T)^2}\right] \Big/ \left(\frac{\sigma_I}{I}\right)^2 . \tag{5.3}$$

The RHS of equation (5.3) is minimum, when both the terms are equal, which on substituting the values of A_β and A_{VT} from the process parameters, gives

$$(V_{GS} - V_T) = 0.2\,V \tag{5.4}$$

Substituting (5.2) and (5.4) in (5.3) gives

$$(WL) \geq 6\,\mu^2 . \tag{5.5}$$

Using $I_{LSB} = 10\,\mu A$ gives a constraint on the ratio $\frac{W}{L}$, which together with (5.5) gives

$$\begin{aligned} W &= 3.2\,\mu \\ L &= 2.4\,\mu \end{aligned} \tag{5.6}$$

Equation (5.6) gives the size of M1 for a unit current cell. All other transistors are minimum sized. For a current cell of any other size, all the transistors are scaled up accordingly.

Table 5.1 gives the sizes of transistors M1, M2, M3, M4 as a function of current cell size.

Table 5.1: Transistor sizes in a current cell of size $n \times I_{LSB}$.

Transistor	W	L	Fingers
M1	$n \times 3.2\,\mu$	$2.4\,\mu$	$2 \times n$
M2	$n \times 320\,n$	$120\,n$	$2 \times n$
M3	$n \times 160\,n$	$120\,n$	n
M4	$n \times 160\,n$	$120\,n$	n

5.2 Latch Design

If both swl and swr signals are turned off momentarily, then the current source transistor (M1) gets switched off. In normal operation however one of swl or swr should be on, and there will be a large glitch in the output as the large capacitance of M1 needs to be charged while turning it on. Hence it is necessary to ensure that swl and swr are never off simultaneously. This is done by using a latch to control the signals, which is driven from a single select input. Thus all switching signals will be synchronized to a common clock, and the number of glitches will be reduced. The latch used is a static CMOS D-latch as shown in figure 5.2. The sizing of the latch transistors is done so as to ensure that whenever there is a transition in swl and swr, the signal switching on does so before the other signal switches off. To ensure this, the PMOS transistors ($\frac{W}{L} = \frac{960n}{120n}$) are made larger than the NMOS transistors ($\frac{W}{L} = \frac{160n}{120n}$). This ensures that the rise time of the signal turning on is much lower than the fall time of the signal turning off.

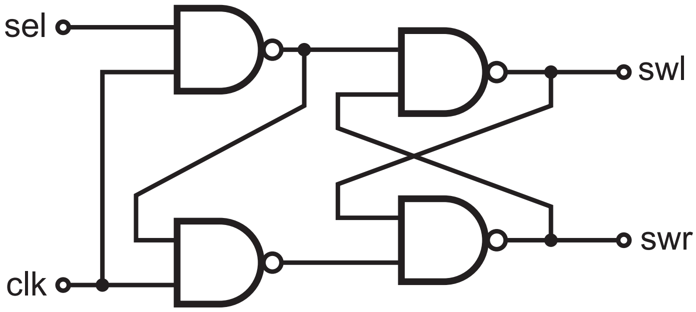


Figure 5.2: The D-latch used to synchronize the switching signals.

5.3 Binary to Thermometric Decoder

The presence of a thermometric coded section in the DAC necessitates the use of a binary to thermometric decoder. For purposes of testing the DAC, this is implemented in Verilog-A, the code for which is given below.

```
// VerilogA for SVBTPDAC, decoder, veriloga

`include "constants.vams"
`include "disciplines.vams"

module decoder (din, dout);
   input [5:0] din;
   voltage [5:0] din;

   output [62:0] dout;
   voltage [62:0] dout;

   parameter real vmax = 1.2;
   parameter real vmin = 0;
   parameter real vtrans = 0.6;
   parameter real trise = 100p from (0:inf);
   parameter real tfall = 100p from (0:inf);
   parameter real tdel = 50p from (0:inf);

   integer out[62:0];
   integer i, sum;

   analog begin
      sum = 0;
      generate i (0,5)
         sum=sum+((V(din[i])>vtrans)?1:0)*pow(2,i);

      for (i=0; i<63; i=i+1) begin
         if(i<sum)
            out[i]=1;
         else
            out[i]=0;
      end

      generate i (0,62)
         V(dout[i])<+transition(1.2*out[i],tdel,trise,tfall);
   end
endmodule
```

The decoder generates the appropriate 63 thermometric control lines from the 6 binary input lines.

CHAPTER 6

Simulation Results

Using a FO4 chain of inverters, the T_{rise}, T_{fall}, T_{del} for the clock signal are estimated to be $100\,ps$, $100\,ps$, and $50\,ps$ respectively. Based on these values, the design is found to operate for a clock frequency of $3.3\,GHz$, and the same is used for all simulations.

6.1 Transient Simulation

The transient simulation plots for test sinusoids of frequencies $20\ MHz$ and $200\ MHz$ are shown in figures 6.1 and 6.2 respectively.

6.2 Noise Performance

Figures 6.3 and 6.4 show the SFDR plots for test sinusoids of frequencies $20\ MHz$ and $200\ MHz$ respectively. The SFDR is $57\ dB$ for the $20\ MHz$ sinusoid and $40\ dB$ for the $200\ MHz$ sinusoid. Figure 6.5 shows the variation of SFDR with the input frequency.

6.3 Linearity

Figures 6.6 and 6.7 show the INL and DNL of the DAC as a function of the input code.

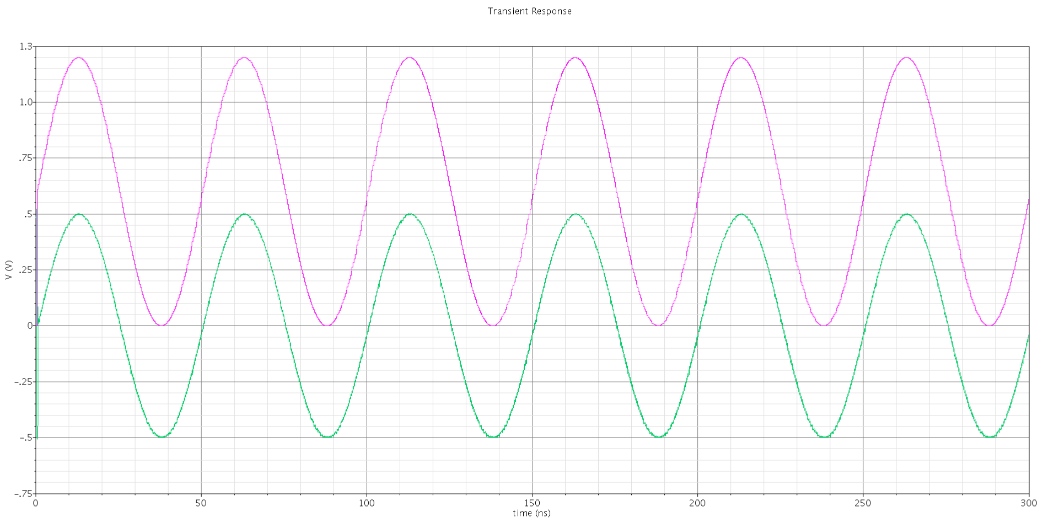


Figure 6.1: Transient simulation for a $20\ MHz$ test sinusoid, with the output of an ideal DAC for comparison.

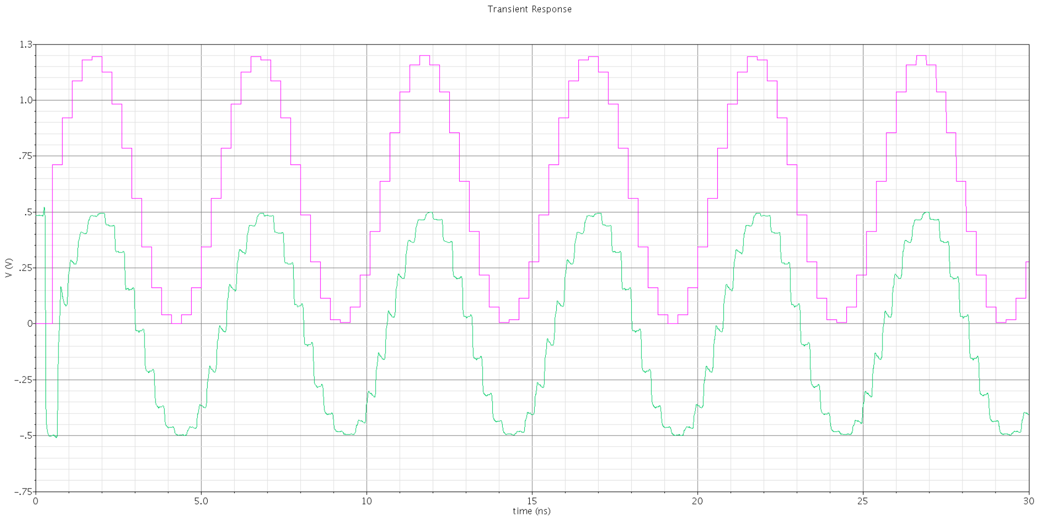


Figure 6.2: Transient simulation for a 200 MHz test sinusoid, with the output of an ideal DAC for comparison.

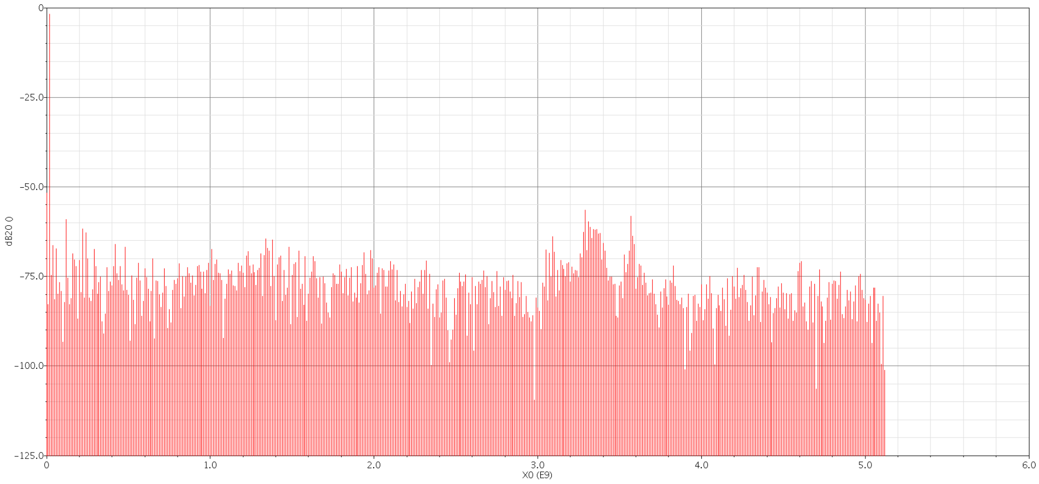


Figure 6.3: SFDR for a 20 MHz test sinusoid.

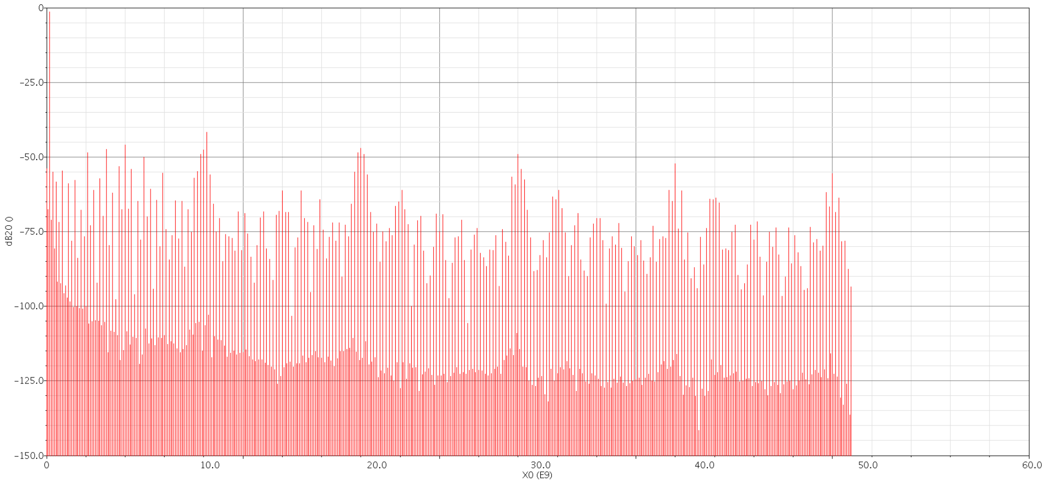


Figure 6.4: SFDR for a 200 MHz test sinusoid.

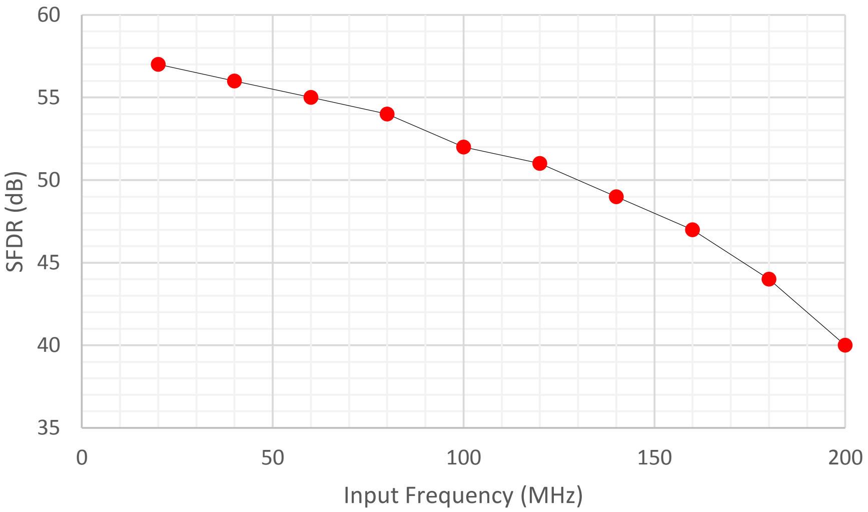


Figure 6.5: SFDR variation with input frequency.

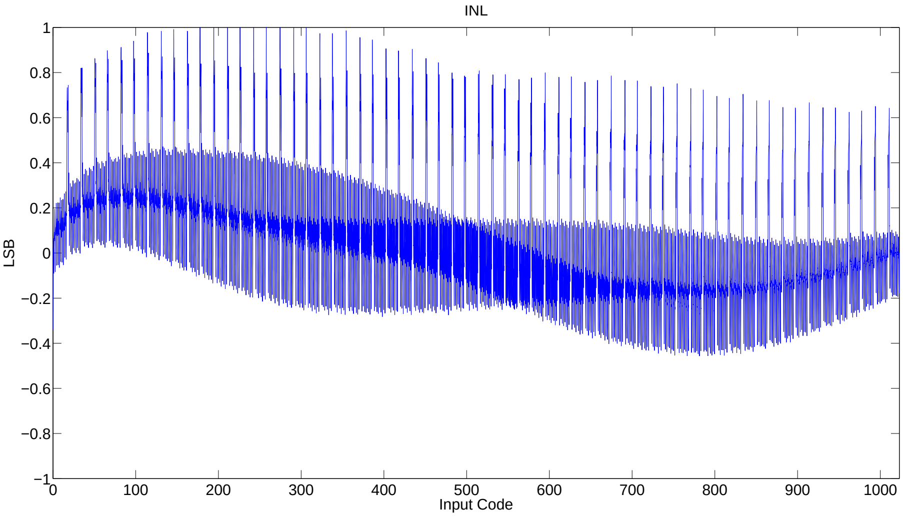


Figure 6.6: INL as a function of input code.

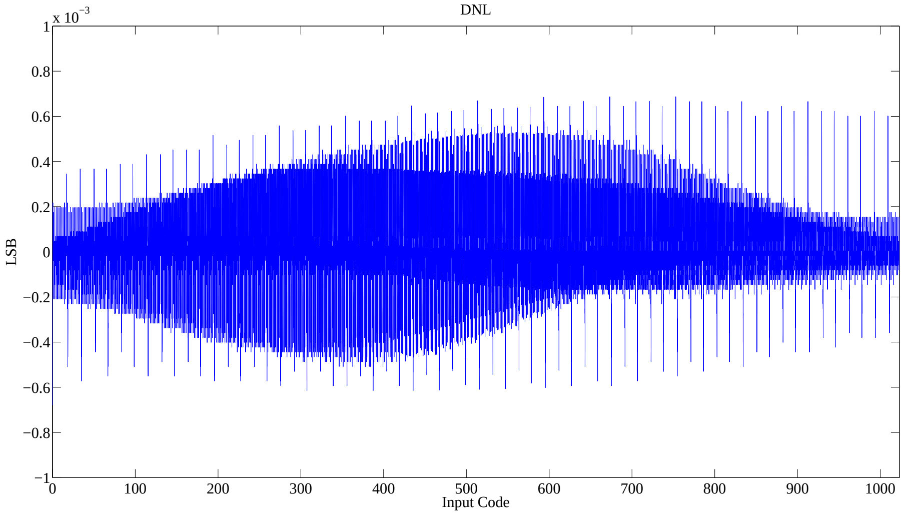


Figure 6.7: DNL as a function of input code.

CHAPTER 7

Layout

Since matching between the current cells is important, it is necessary to take into account process gradients while carrying out the layout. Each of the current cells of size 16 (which form the bulk of the DAC), are divided into four cells of size 4 each and laid out in a common centroid layout, so that the effect of gradients is nullified. Thus the 63 current cells of size 16 are divided into 4 cells each, and laid out in a grid of size 16×16, such that the centroid of each group of 4 cells lies at the center. A ring of dummy cells surrounds the grid to ensure an identical environment. The layout of the current cell of size 4 is shown below (For other current cells, the same layout is retained and only the sizes of the transistors M1, M2, M3, and M4 are appropriately scaled).

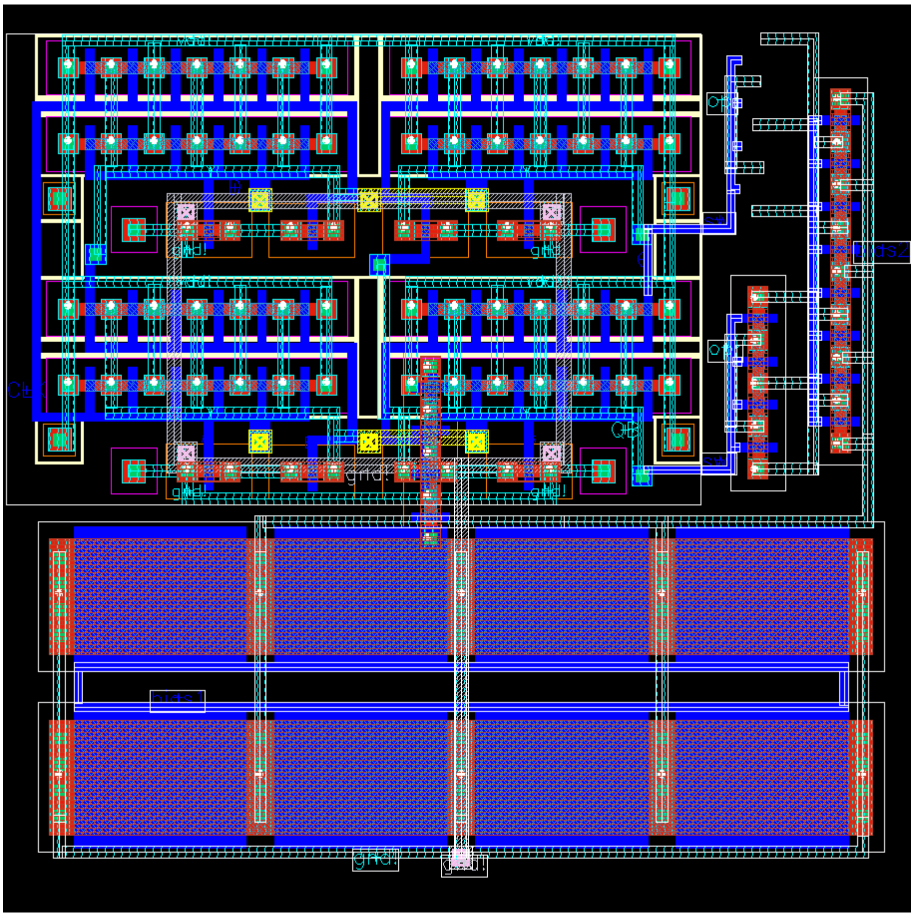


Figure 7.1: Layout of a current cell of size 4.

CHAPTER 8

Conclusion

This document describes the design of a 10-bit current steering DAC in a 130 nm UMC process. A 6+4 segmentation is used (6-bit thermometric and 4-bit binary sections). The area occupied by the analog section of the DAC is 0.065 mm^2. The static power consumption of the analog portion is 12 mW. The DAC is tested and is found to work for sampling frequencies up to 3.3 GSPS. The SFDR is 58 dB for an input frequency of 20 MHz and 40 dB for an input frequency of 200 MHz.

REFERENCES

[1] A. Assalini and A. Tonello, "Improved Nyquist Pulses", *IEEE Communications Letters*, Vol. 8, No.2, pp. 87 – 88, 2004.

[2] B. Baker, "How delta-sigma ADCs work, Part 1", *Analog Applications Journal – Texas Instruments Incorporated*, 3Q, pp. 13 – 16, 2011.

[3] B. Baker, "How delta-sigma ADCs work, Part 2", *Analog Applications Journal – Texas Instruments Incorporated*, 4Q, pp. 5 – 7, 2011.

[4] J. Bastos *et al*, "A 12-Bit Intrinsic Accuracy High-Speed CMOS DAC", *IEEE Journal of Solid State Circuits*, Vol. 33, No.12, pp. 1959 – 1969, 1998.

[5] N. Beaulieu, C. Tan, and M. Damen, "A "better than" Nyquist Pulse", *IEEE Communications Letters*, Vol. 5, No.9, pp. 367 – 368, 2001.

[6] D. Behera, "A Technique to get Memoryless Conversion from Delta-Sigma Modulators", Private Communication, 2012

[7] S. Chandan, P. Sandeep, and A. K. Chaturvedi, "A family of ISI-free polynomial pulses", *IEEE Communications Letters*, Vol. 9, No.6, pp. 496 – 498, 2005.

[8] M. Joost, "Theory of Root-Raised Cosine Filter", 2010.

[9] R. Schrier, *Delta Sigma Toolbox*, http://www.mathworks.in/matlabcentral/fileexchange/19-delta-sigma-toolbox.

[10] I. Selesnick, *Linear-Phase FIR Filter Design by Linear Programming*, http://cnx.org/content/m11115/latest.

[11] X. Xia, "A Family of Pulse-Shaping Filters with ISI-Free Matched and Unmatched Filter Properties", *IEEE Transactions on Communication*, Vol. 45, No.10, pp. 1157 – 1158, 1997.