RC IMPLEMENTATION OF AN AUDIO FREQUENCY BAND
BUTTERWORTH MASH
DELTA-SIGMA ANALOG TO DIGITAL DATA CONVERTER

A Thesis by
Sudheer Vijjapu
Bachelor of Engineering, Jawaharlal Technological University, 2002

Submitted to the College of Engineering
and the faculty of the Graduate School of
Wichita State University in partial fulfillment of
the requirements for the degree of
Master of Science

August 2006
RC IMPLEMENTATION OF AN AUDIO FREQUENCY BAND
BUTTERWORTH MASH
DELTA-SIGMA ANALOG TO DIGITAL DATA CONVERTER

I have examined the final copy of this thesis for form and content and recommend that it
be accepted in partial fulfillment of the requirements for the degree of Master of Science
with a major in Electrical Engineering.

_________________________________
Larry D. Paarmann, Ph.D., Committee Chair.
DEDICATION

To my parents
Mr. V. Sasi Bhushanam and Mrs. V. Santha Kumari
for their love and support
ACKNOWLEDGEMENTS

I would like to express my deep gratitude to the advisor of my thesis, Professor Dr Larry D. Paarmann for his patience and guidance throughout the thesis work. I would like to thank my parents for their constant support and love. I would also like to thank my sister Sirisha and friends Srinivas vaidya, Kalyan Ravipati, Kiran Irukulla, Srinivas Marepalli for their encouragement and support. I also thank all the members of the Delta-Sigma research group for giving me valuable suggestions on my thesis work. I thank my committee members Dr and for spending their valuable time in attending my thesis defense. Lastly, I would like to thank God for his kind blessings.
ABSTRACT

Most present day implementations of delta-sigma modulators are discrete-time ones using switched-capacitor circuits. A resistor-capacitor (RC) implementation of a delta-sigma analog to digital converter (ADC) does not use switched capacitor (SC) technology. While SC implementation has the advantages of being discrete-time, no resistors used and improved stability control, RC implementation has the advantage of no switches being used (other than quantizer) and therefore a simpler circuit implementation. Continuous-time implementations can achieve lower thermal noise levels than switched capacitor modulators. Butterworth Multi-stage Noise Shaping (MASH) architecture is one of the promising architectures to implement in continuous-time domain. For a convenient design and quantization noise spectrum shaping of a delta sigma data converter, it’s highly desirable for the Noise Transfer Function (NTF) to take the form of a high-pass filter. The MASH architecture was introduced to overcome stability problems commonly faced beyond a second order structure.

Delta-sigma data converters are new converter designs that are preferred for integrated circuits and for high-resolution applications. It is highly desirable for the NTF of delta-sigma data converters to take the form of conventional highpass filters for convenient design purposes and shaping of the quantization noise spectrum. However, conventional delta-sigma architectures allow for only low orders and very low cutoff frequencies for such highpass filters, otherwise the converter becomes unstable. In previous projects it was found that a MASH implementation (each stage being second order) of a delta-sigma data converter where the NTF of each stage is a Butterworth highpass filter holds much promise. This current project is to accomplish RC
implementation of fourth-order Butterworth MASH delta-sigma data converter. The circuit design procedure will be shown, starting with the desired NTF characteristics, and yielding the required parameters for the RC integrators with gains that are determined from the desired transfer function. The circuit simulation, yielding the bit stream frequency spectrum and the signal to noise ratio, will be based on Mentor Graphics Eldo SPICE simulations. The performance and characteristics of the circuit is fully analyzed and documented for a wide variety of variations and test conditions.
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Chapter</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>I  INTRODUCTION</td>
<td></td>
</tr>
<tr>
<td>1.1 Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2 Description of the Thesis</td>
<td>2</td>
</tr>
<tr>
<td>1.3 Thesis Organization</td>
<td>2</td>
</tr>
<tr>
<td>II  BACKGROUND</td>
<td></td>
</tr>
<tr>
<td>2.1 Overview of Analog to Digital Converters</td>
<td>4</td>
</tr>
<tr>
<td>2.1.1 Oversampling</td>
<td>5</td>
</tr>
<tr>
<td>2.1.2 Quantization</td>
<td>6</td>
</tr>
<tr>
<td>2.1.3 Noise Shaping</td>
<td>7</td>
</tr>
<tr>
<td>2.1.4 Digital Filtering and Decimation</td>
<td>8</td>
</tr>
<tr>
<td>2.2 Different types of ADC’s</td>
<td>9</td>
</tr>
<tr>
<td>2.2.1 Integrating ADC</td>
<td>9</td>
</tr>
<tr>
<td>2.2.2 Successive Approximation ADC</td>
<td>10</td>
</tr>
<tr>
<td>2.2.3 Flash ADC</td>
<td>10</td>
</tr>
<tr>
<td>2.3 Continuous time vs. Discrete time Sigma-Delta Data Converters</td>
<td>11</td>
</tr>
<tr>
<td>2.4 Delta Modulation</td>
<td>11</td>
</tr>
<tr>
<td>2.5 Various Delta-Sigma Architectures</td>
<td>12</td>
</tr>
<tr>
<td>2.5.1 First Order Delta-Sigma Converter</td>
<td>13</td>
</tr>
<tr>
<td>2.5.2 Conventional Second Order Delta-Sigma ADC</td>
<td>14</td>
</tr>
<tr>
<td>2.5.3 Second Order Butterworth ADC</td>
<td>15</td>
</tr>
<tr>
<td>III  MASH ARCHITECTURE IN S DOMAIN</td>
<td></td>
</tr>
<tr>
<td>3.1 Introduction to MASH Architecture</td>
<td>17</td>
</tr>
<tr>
<td>3.2 Analysis of Sigma-Delta Modulation in S domain</td>
<td>19</td>
</tr>
<tr>
<td>3.3 Implementation of Fourth Order Butterworth MASH Architecture in S domain</td>
<td>20</td>
</tr>
<tr>
<td>3.3.1 Obtaining H(s) from Noise Transfer Function</td>
<td>22</td>
</tr>
<tr>
<td>3.3.2 Design Example</td>
<td>23</td>
</tr>
<tr>
<td>3.3.3 Design for Implementing H₁(s) and H₂(s)</td>
<td>24</td>
</tr>
<tr>
<td>3.3.4 Component Values for H₁(s)</td>
<td>25</td>
</tr>
<tr>
<td>3.3.5 Design Example and Component Values for H₂(s)</td>
<td>25</td>
</tr>
<tr>
<td>3.3.6 Design Example and Component Values for H₆(s)</td>
<td>26</td>
</tr>
<tr>
<td>IV.  RESULTS AND SIMULATIONS</td>
<td></td>
</tr>
<tr>
<td>4.1 SPICE Simulation</td>
<td>28</td>
</tr>
<tr>
<td>4.2 Simulation Results of Fourth Order Butterworth MASH Architecture</td>
<td>29</td>
</tr>
<tr>
<td>4.3 Simulation Results of 2-Stage Fourth Order Conventional Mash Architecture</td>
<td>36</td>
</tr>
</tbody>
</table>
## LIST OF FIGURES

2.1 Generalized Analog to Digital Converter .............................................................. 4
2.2 Characteristics of a Quantizer ............................................................................... 7
2.3 Result of Noise Shaping ....................................................................................... 8
2.4 Digital Filter and Decimation ............................................................................... 9
2.5 Block Diagram of a Delta Modulator .................................................................... 12
2.6 First-Order Sigma-Delta Data Converter ............................................................ 13
2.7 Input and Output of First Order Sigma-Delta Converter ...................................... 14
2.8 Conventional Two Stage Second Order Sigma-Delta Data Converter ................. 14
2.9 Single stage Delta-Sigma data converter which uses a second order Butterworth Highpass NTF ................................................................................................... 16
3.1 Linearized Model ............................................................................................... 19
3.2 Butterworth MASH Architecture ....................................................................... 21
3.3 Architecture of a Single Second Order Stage in Continuous Time ..................... 24
4.1 Magnitude Frequency Response of Overall NTF ................................................ 30
4.2 Signal Band Detail of Overall NTF ..................................................................... 31
4.3 Magnitude Frequency response of NTF of each stage of Butterworth MASH Architecture ........................................................................................................... 31
4.4 Magnitude Frequency response of Overall STF ............................................... 32
4.5 Signal Band Detail of Overall STF ..................................................................... 32
4.6 Butterworth Fourth order MASH Delta-Sigma Data Converter .......................... 33
4.7 Magnitude Spectrum Analysis of first stage of Butterworth MASH Architecture ................................................................................................................. 34
4.8 Magnitude Spectrum Analysis of Butterworth MASH Architecture .................35
4.9 Conventional Second-Order Delta-Sigma Architecture.................................37
4.10 Fourth Order Conventional MASH Architecture ........................................38
4.11 Magnitude Spectrum Analysis of 2-Stage Fourth Order Conventional MASH Architecture ........................................................................................................39
CHAPTER I
INTRODUCTION

1.1 Motivation

The modern day high resolution applications and integrated circuits use Delta-Sigma data converters for their converter design. Delta-Sigma data converters have two very important features. First, they are preferred for integrated circuit (IC) technology. Delta-Sigma ADCs have desirable implementation characteristics for VLSI design. And what is most important, they do not require precision components like resistors which are used as external components in an IC design which are expensive and need more space. Therefore, Delta-Sigma ADCs are attractive for many commercially available ADC ICs. They are also suited for embedded system ICs, such as those found in many cell phones, and other highly integrated systems. Secondly, where high precision analog-to-digital conversion is required, Delta-Sigma ADCs have advantages. Delta-Sigma data converters perform noise shaping of quantization noise so that most of the noise is pushed beyond the signal band, which can be further reduced by low pass filtering. These converters achieve very high resolution by oversampling.

Sigma-delta data converters use time-redundant data to exchange operation speed for resolution and thus increase robustness against unavoidable parasitic and components tolerances. Consequently, they are very well suited for on-chip design of high-resolution analog interfaces in mixed signal application specific integrated circuits (ASICs) fabricated in general-purpose CMOS technologies. Although initial activities on sigma-delta converters ICs focused mainly on audio applications, their current application scope is considerably wider from instrumentation to telecom. The exploration of these
application fields are driven by industry in the direction of increased operation speed and resolution, decreased power consumption, and the convenience to migrate to modern low-voltage submicron technologies.

1.2 Description of the thesis

Nowadays various types of advanced Delta-Sigma architectures have been developed for better performance. Some of the architectures are pure feed-forward in design, some are pure feedbacks, and some use a combination of these two. In this thesis project a totally new architecture will be discussed using the simulations. Simulations are carried out using MATLAB 6.5 and DESIGN ARCHITECT of Mentor Graphics for SPICE simulations. Butterworth Fourth Order MASH architecture design will be proposed, and its performance is fully analyzed and compared to the existing architectures. The primary aim of this thesis is as follows: To design the Butterworth Fourth Order MASH architecture which has better performance and a wider stop-band with more attenuation than the conventional fourth order MASH. Data is presented illustrating the enhanced performance of the MASH architecture.

1.3 Thesis Organization

Following this chapter, Chapter 2 provides an overview of analog to digital conversions and reviews the fundamental advantages of over sampling technique as a method to perform high resolution data conversion.

Chapter 3 provides the design algorithm for implementing a Butterworth MASH filter in time domain using all ideal continuous time components with integrators is explained in detail.
In Chapter 4, the simulation results for the proposed architectures are documented.

Chapter 5 presents the conclusions.
CHAPTER 2

BACKGROUND

2.1 Overview of Analog to Digital Data Converters

Analog to digital conversion is defined as a process that converts a continuous time analog signal into multi level digital signal without any significant change in its essential content. Analog to digital converter is a device that converts an analog signal to a digital signal with a given accuracy and resolution. The principle of delta-sigma architecture is to make rough evaluations of the signal, to measure the error, integrate it and then compensate for that error.

Figure 2.1 illustrates a generalized analog to digital converter. Anti aliasing filter, sample and hold circuit and the quantizer or analog to digital conversion are the three basic components of the block diagram shown in figure 2.1. To prevent aliasing, the input must be bandlimited to frequencies below one-half the desired sampling rate. This can be performed by sending the analog signal through a low pass filter by using an anti-aliasing filter. The sampling rate is atleast equal to twice the signal frequency which is nothing but nyquist rate. The anti-aliasing filter is a low pass filter with a cut-off frequency of $\Omega_c = \frac{\Omega_s}{2}$, where $\Omega_s$ is the sampling frequency. $x_c(t)$ is the input to the anti-aliasing filter $x_a(t)$ is the output of the anti-aliasing filter.

![Figure 2.1 Generalized Analog to Digital Converter](image-url)
In ideal case, the frequency response of the anti-aliasing filter is given by,

\[
H_{aa}(j\Omega) = \begin{cases} 
1, & |\Omega| < \Omega_\epsilon < \pi / T \\
0, & |\Omega| > \Omega_\epsilon 
\end{cases}
\]

The anti-aliasing filter is used to make the input signal bandlimited. But in practice, the frequency response of the filter cannot be bandlimited ideally. So \( H_{aa}(j\Omega) \) can be made very small for \( |\Omega| > \pi / T \) for aliasing to be minimized. With the help of external clock, ADC can be caused to start and complete an analog to digital conversion for every \( T \) seconds. ADC’s typically use a sample and hold circuit as the conversion is not instantaneous. \( X_o(t) \) is the output of the sample and hold circuit and is given by,

\[
X_o(t) = \sum_{n=-\infty}^{\infty} x[n]h_0(t - nT)
\]

where \( x[n] = x_a(nT) \) are the ideal samples of \( X_a(T) \) and \( h_0(t) \) is the impulse response of the zero order hold system which can be shown as

\[
h_0(t) = \begin{cases} 
1, & 0 < t < T \\
0, & otherwise 
\end{cases}
\]

### 2.1.1 Oversampling

The first operation in analog to digital conversion is sampling. The input signal is sampled repeatedly over a period of time so as to determine the characteristic that contains the analog property such as the signal amplitude. Consider a sine wave input to a multi-bit ADC. The input is sampled at a frequency \( F_s \). \( F_s \) must be at least twice the bandwidth of the input signal according to the nyquist theory. If the signal is sampled at a frequency higher than Nyquist frequency (\( F_{NR} \)) then it is called oversampling and it is given by \( O_k = \frac{F_s}{F_{NR}} \).
When observing the results of FFT analysis on the digital output a single tone and large amount of random noise extending from the DC to the Fs/2 is observed, this is known as quantization noise. The input to ADC is continuous, but the output is digital and a discrete function, whose number of different states is determined by the converter’s resolution. Therefore, some information is lost and some distortion is introduced into the signal during the process of conversion from analog to digital.

The signal-to-noise ratio, SNR is the ratio of fundamental amplitude to the root mean square sum of all the frequencies representing noise. Usually for an N-bit ADC, \( \text{SNR} = 6.02N + 1.76 \text{db} \) (Park, 1999). Number of bits must be increased to improve the SNR in a conventional ADC. The SNR can be easily improved by a technique called over sampling. The SNR for a one-bit ADC is 7.78 dB (6.02 +1.76). Each factor of 4 over sampling increases the SNR by 6 dB, and each 6 dB increase is equivalent to gaining one-bit. Sigma delta enables a gain of more then 6 db for each factor of 4 times over sampling.

2.1.2 Quantization

Quantization in amplitude and sampling in time are the two main functions of all digital modulators. After sampling the signal samples must also be quantized in amplitude to a finite set of output values.

Typical characteristics of the quantizer for the input signal plotted against the x-axis and output signal plotted against the y-axis is shown in figure 2.2.

The quantized signal \( y \) is represented by a linear function \( Gx \) with an error \( e \).

\[
y = Gx + e
\]
Figure 2.2 Characteristics of a quantizer (taken from (Haykin, 1997)).

In this example, the level spacing delta = 2. The slope of the straight line is the gain G, which passes through the center of quantization characteristics so that when the quantizer does not saturate, the error is bounded by ± delta/2.

2.1.3 Noise Shaping

Noise shaping can be better understood with the figure 2.3 shown below. This figure shows a difference amplifier, an integrator, and a one-bit comparator with a feedback loop with a one-bit ADC. The purpose of the feedback DAC is to maintain the average output of the integrator near the comparator’s reference level.

This figure shows a difference amplifier, an integrator, and a one-bit comparator with a feedback loop with a one-bit ADC. The purpose of the feedback DAC is to maintain the average output of the integrator near the comparator’s reference level. It is evident from the figure that the noise is kept low in signal band and pushed to higher frequencies.
The density of “ones” at the modulator output is proportional to the input signal. For an increasing input, the comparator generates a greater number of “ones.” By summing the error voltage the integrator acts as a low pass filter to the input signal and a high-pass filter to the quantization noise. Thus, the noise is pushed into higher frequencies. Figure 2.3 shows the noise shaped spectrum.

Therefore in a lower frequency band, the desired input signals are passed and changed very little while the noise is suppressed to a great extent.

With higher-order modulators, the noise is even more suppressed. This is called noise shaping. Due to noise shaping, higher-order modulators produce less output noise.

2.1.4 Digital Filtering and Decimation

Figure 2.4 below shows the digital filter and decimation. The output of the sigma delta modulator is a one-bit data stream at the sampling rate, which can be in the mega
hertz range. The main function of the digital and decimation filter is to extract information from this data stream and reduce the data rate to a more useful value.

Decimation takes place in delta-sigma converters at the output of the low-pass filter. Since the bit stream is clocked and multiplied with the oversampling rate, the output of the low-pass filter also is clocked with the oversampling rate. However, the sample rate clock (twice the input bandwidth) is required at the digital output only. For an over sampling rate of 64, for example, every 64th sample is taken and all others are discarded. This is possible because the signal bandwidth is limited by the low pass filter.

The digital filter improves the ADC resolution by averaging the one-bit data stream and removes quantization noise that is outside the band of interest. It determines the signal bandwidth, settling time, and stop-band rejection.

2.2 Different types of ADC

There are various types of analog to digital converters that exist and they employ different methods to arrive at a digital representation of the voltage signal. Some of them are described in this section.

2.2.1 Integrating ADC

Integrating ADC is best suited for measurements of signals that fluctuate slowly. These converters reduce the unwanted signal (noise) by averaging the signal. They provide high resolution and rejection of noise. This architecture provides an approach to
converting bandwidth analog signal to its digital representation. Integrating ADC’s are used in portable instruments including digital panel meters and digital multi-meters.

2.2.2 Successive Approximation ADC

Successive approximation ADC’s uses digital logic to converge on the value closest to the input voltage. These converters are used for faster and high resolution applications typically with sample rates fewer than 5 mega samples per second. Successive approximation ADC consists of a sample and hold circuit, a register consisting of clocked flip-flops and gates, a internal reference DAC and a voltage comparator. When the successive approximation ADC is clocked, it generates a series of digital codes, which are fed into the reference DAC one at a time. These digital codes are produced in binary search fashion. Successive approximation ADC’s most common range of resolution is from 8 to 16 bits and they have low power consumption as well as a small form factor.

2.2.3 Flash ADC

The Flash converter is the converter that operates in one clock cycle. The converter consists of a resistance ladder, an array of comparators and an encoder. The endpoints of the resistance ladder define the input range of the ADC, while the intermediate points define the code transition levels. The input voltage $V_{\text{in}}$ is simultaneously compared with all code transition levels in the comparators array. If $V_{\text{in}}$ exceeds the k-th transition level, the k-th comparator output will be 1 else it will be 0. Flash converters consume lot of power, have relatively low resolution and can be quite expensive.
2.3 Continuous time vs. Discrete time Sigma-Delta Data Converters

Discrete-time delta sigma modulators are implemented using switched capacitor (SC) circuit techniques. In SC circuits, amplifiers with high gain-bandwidth product (GBW) satisfy the settling requirements. Typically, the GBW is seven times higher than the sampling frequency. By nature, Continuous time delta sigma modulators are not sensitive to settling behavior. As a result, Continuous delta-sigma modulators can potentially operate at higher clock frequency and/or with less power consumption. Note that in a Continuous time modulator, the loop filter provides additional anti-aliasing filtering, which is beneficial when having to handle large interferers. In SC circuits, the in-band noise is bounded by the capacitor size. Contrary to a Continuous time modulator, in a Discrete time modulator, large glitches appear on the op-amp virtual ground node of op-amps-RC integrators due to switching transient. Therefore, a Continuous time modulator achieves better linearity performance. Today, Continuous time modulators are preferred to discrete time modulators, whatever be the application.

2.4 Delta Modulation

Analog-to-digital signal conversion in which (a) the analog signal is approximated with a series of segments, (b) each segment of the approximated signal is compared to the original analog wave to determine the increase or decrease in relative amplitude, (c) the decision process for establishing the state of successive bits is determined by this comparison, and (d) only the change of information is sent, i.e., only an increase or decrease of the signal amplitude from the previous sample is sent whereas a no-change condition causes the modulated signal to remain at the same 0 or 1 state of the previous sample.
Figure 2.5 shows the block diagram of a delta modulator. Delta modulation is based on quantizing the change in the signal from sample to sample rather than the absolute value of the signal at each sample. Since the output of the integrator in the feedback loop of Figure tries to predict the input $x(t)$, the integrator works as a predictor. The prediction error term, in the current prediction is quantized and used to make the next prediction. The quantized prediction error (delta modulation output) is integrated in the receiver just as it is in the feedback loop. That is, the receiver predicts the input signals. The predicted signal is smoothed with a lowpass filter. Delta modulators, furthermore, exhibit slope overload for rapidly rising input signals, and their performance is thus dependent on the frequency of the input signal. In theory, the spectrum of quantization noise of the prediction error is flat and the noise level is set by the 1-bit comparator.

### 2.5 Various Delta-Sigma Architectures

There are different types of delta-sigma architectures. First order and second order architectures are the basic types. Delta sigma architectures are not extended beyond second order because of instability in the internal feedback circuit. But various advanced techniques are now developed to increase the order beyond second order architectures. It is usually assumed that the quantization noise is a signal independent uniform white random process. There can be various feedback loops inside delta-sigma modulator...
architecture. These modulators can be cascaded to form multistage delta-sigma modulators.

2.5.1 First-Order Sigma-Delta Data Converter

A first order sigma delta converter is shown in the above figure 2.6. It consists of an integrator and a comparator with a one-bit DAC in a feedback loop. The modulator output has only one bit of information either +1 or -1. The modulator output $y(n)$ is converted to $y(t)$ by a one-bit DAC. The input to the integrator of the modulator is the difference between the input signal $x(t)$ and the quantized output value of $y(n)$ converted back to the predicted analog signal $y(t)$. Considering the DAC is perfect then the difference between the input signal $x(t)$ and the predicted analog signal $y(t)$ is the quantization error. This error is summed up by the integrator and once again quantized by a one-bit ADC. Even though the quantization error at every sampling instance is large due to the nature of the two-level quantizer, the action of the sigma-delta modulator loop is to generate a ±1, which can be averaged over several input sample periods to produce a very precise result. The averaging is usually performed by a decimation filter at the output of the modulator. The input and output waveforms of the first-order modulator are shown in figure 2.7 below. In each clock cycle, the value of the output of the modulator is either ±1. These modulators can achieve a very high SNR.

![Figure 2.6 First-Order Sigma-Delta Data Converter (taken from (Park, 1999)).](image)
2.5.2 Conventional Second Order Delta-Sigma ADC

Figure 2.8 represents a Conventional second order delta-sigma ADC. It consists of two integrators, a sample and hold circuit and a comparator with a feedback loop feeding to the input differential gain amplifier as shown in figure. The modulator output has only one bit of information either +1 or -1. The input to the integrator of the modulator is the difference between the input signal $x(t)$ and the quantized output value.
of \( y(n) \) converted back to the predicted analog signal \( y(t) \). The main reason is that the modulators with second-order transfer function involve the cancellation of the past two samples and thus exhibit stronger attenuation at low frequencies.

### 2.5.3 Second Order Butterworth ADC

Figure 2.9 represents a second order Butterworth ADC. It consists of one stage whose NTF is a second Butterworth highpass filter and is given by,

\[
NTF = \frac{s^2}{s^2 + \sqrt{2}\Omega_c s + \Omega_c^2}
\]

\[
H(s) = \frac{1}{NTF} = \frac{1}{\frac{s^2}{s^2 + \sqrt{2}\Omega_c s + \Omega_c^2}} = \frac{\sqrt{2}\Omega_c s + \Omega_c^2}{s^2}
\]

\( H(s) \) can be implemented with the help of three integrators (\( Y_1, Y_2, \) and \( Y_3 \)) as shown in above figure. This is followed by a sample and hold circuit (\( Y_4 \)) and a comparator (comp1). The output of the comparator is fed back to the input and it is subtracted from the input signal as shown in the figure. Therefore the input to the integrators of the modulator is the difference between the input signal and the quantized output value. Second order Butterworth ADC achieves a higher SNR as compared to conventional first order and second order ADC’s. The main reason is that with a correct choice of cutoff frequency \( \Omega_c \), it is possible to meet the 3dB gain rule and the causality rule. Also, the Butterworth alignment tends to be less susceptible to oscillations caused by input signals as compared to the conventional first and second order ADC’s.
Figure 2.9 Single Stage Delta-Sigma Data Converter which uses a Second Order Butterworth Highpass NTF.
CHAPTER 3

MASH ARCHITECTURE IN S DOMAIN

3.1 Introduction to MASH Architecture

Delta sigma data converters have various advantages over the classical Nyquist rate converters. These converters make use of oversampling and noise shaping for accuracy and thus the converter is quite insensitive to mismatch and other circuit imperfections. These converters require the use of digital filtering to remove the out-of-band quantization noise. Delta sigma data converters are very suitable to implement high speed analog to digital converters as they have the combination of relaxed analog circuit requirements and the use of digital signal processing.

Oversampling converters are the converters whose input signal is sampled much faster than the Nyquist rate. The signal bandwidth of the input signal is denoted by $f_b$ and the nyquist rate is given by,

$$f_N = 2 \times f_b$$

A generalized single stage delta sigma modulator with an input sampled by the modulator at an oversampled rate between 16 and 256 times the Nyquist rate is clearly an oversampling converter. The converter has a filter $H(z)$ and a 1-bit quantizer. The output of the quantizer is fed back as input through another filter $F(z)$ and subtracted from the input. It results in the average value of the quantized output to follow the input. An error $e(n)$ is hence the difference between the quantizer output and quantizer input. The Signal transfer function (STF) is given by,

$$\frac{H(z)}{1 + H(z)F(z)}$$
The Noise transfer function (NTF) is given by

\[
\frac{1}{1 + H(z)F(z)}
\]

The digital low pass filter removes the out-of-band quantization noise from the analog modulator. The modulator is followed by a decimator, which reduces the sampling rate down to Nyquist rate.

A high resolution oversampling analog to digital converter can utilize MASH (Multi Stage Noise Shaping modulator) architecture and can be implemented with VLSI MOS technology. For a second order delta sigma modulator, there are significant improvements in terms of dynamic range and reduction of idle pattern tones. To realize higher order modulators, lower order modulators can be cascaded each with its own quantizer. Each single loop modulator in cascade will cancel the quantization error from previous modulator. Quantization errors from all modulators are digitally cancelled except for the last modulator.

The second order modulator cancels the first stage quantization error. The output of second modulator is combined with the output of first modulator, thus canceling the error. The whole system would still be stable (provided the modulators are stable) because the cascaded modulator structure contains only feed forward paths and no feedback between the individual modulators. This is a distinct advantage for cascaded structure. In this way it can be inherently stable for any number of modulators over all regions of operation, but at some point the noise, which may not be removed, from the previous modulators would limit cascading for higher orders, as no gain would be realized by adding more stages.
3.2 Analysis of Sigma-Delta Modulator in S Domain

A linearized model of a first-order sigma-delta modulator in S domain is shown in figure 3.1. It can be observed that the feedback loop does not have a delay element.

The signal-transfer function STF is given by

\[ Y(s) = \left[ X(s) - Y(s) \right] \frac{1}{s} \]

\[ STF = \frac{Y(s)}{X(s)} = \frac{1}{s} \left( \frac{1}{1 + \frac{1}{s}} \right) = \frac{1}{s+1} \]

When the quantization noise

\[ N(s) = 0 \]

The system transfer function is given by

\[ H(s) = \frac{1}{s} = \frac{Y(s)}{X(s) - Y(s)} \]

The noise-transfer function NTF when the input signal X(s) is zero is given by

\[ Y(s) = -Y(s) \left[ \frac{1}{s} \right] + N(s) \]

Figure 3.1 Linearized model. (taken from (Park, 1999)).
\[ NTF = \frac{Y(s)}{N(s)} = \frac{s}{s+1} \]

Therefore, the STF and NTF can be expressed in terms of \( H(s) \) as,

\[ STF = \frac{Y(s)}{X(S)} = \frac{H(s)}{1+H(s)} \quad (3.1) \]

\[ NTF = \frac{Y(s)}{N(s)} = \frac{1}{1+H(s)} \quad (3.2) \]

where \( x(t) \) is the input signal, \( y(t) \) is the output signal, and \( n(t) \) is the quantization noise.

It follows from equations (3.1) and (3.2) that

\[ STF = H(s) NTF = 1 - NTF \]

Since one of the properties of the sigma-delta modulator is to push the quantization noise to higher frequencies beyond the signal band, the signal-transfer function will be constant across the signal band, and the noise transfer function will be a high-pass transfer function.

MASH architecture is considered since the conventional delta-sigma data converter will be unstable in higher orders. In the case of MASH architectures, the use of integrators to represent \( H(s) \) will lead the system to be stable.

3.3 Implementation of Fourth Order Butterworth MASH architecture in S Domain

The figure 3.2 represents two stage fourth order Butterworth MASH architecture in cascade arrangement. \( x(t) \) is the analog signal input, \( y[n] \) is final output of the two stages. \( H_1(s) \) and \( H_2(s) \) are the transfer functions of the two stages. Let \( STF_n \) and \( NTF_n \) be the signal and noise transfer functions of \( n \)th stages respectively. \( Q_1(s) \) and \( Q_2(s) \) are the quantization errors of the two stages respectively. \( Y_1(s) \) is the output of first stage and \( Y_2(s) \) is the output of second stage respectively. \( Y_{out} \) is the final output obtained by
summing the output of two stages. Let \( Y_{2c}(s) \) be the output of the compensation filter \( H_c(s) \).

From the figure it is evident that the output of first stage is

\[
Y_1(s) = X(s) \text{STF}_1 + Q_1(s) \text{NTF}_1
\]  

(3.3)

Equation (3.3) can also be written as

\[
Y_1(s) = X(s) \text{STF}_1 + Q_{10}(s)
\]  

(3.4)

Where \( Q_{10}(s) \) is shaped quantization of the first stage.

Also the output of second stage is given by

\[
Y_2(s) = -Q_1(s) \text{STF}_1 + Q_2(s) \text{NTF}_2
\]  

(3.5)

From (3.4), the above equation (3.5) can be written as

\[
Y_2(s) = -Q_{10}(s) \text{STF}_1 + Q_2(s) \text{NTF}_2
\]

Then the output of the compensation filter would be

\[
Y_{2c}(s) = -H_c(s) \text{STF}_1 Q_{10}(s) + H_c(s) Q_2(s) \text{NTF}_2
\]

Figure 3.2 Butterworth MASH architecture.
In order to negate the quantization noise from the first stage it would be desirable to have

\[ H_c(s) \ STF_1 = 1 \]  

(3.6)

Each stage of the MASH architecture represents a Butterworth second order NTF with cut off frequency as \( \Omega_c \).

Therefore,

\[ NTF_1 = NTF_2 = \frac{s^2}{s^2 + \sqrt{2} \Omega_c s + \Omega_c^2} \]  

(3.7)

### 3.3.1 Obtaining \( H(s) \) from Noise-Transfer Function NTF

In the following section, a fourth order Butterworth high-pass NTF is specified, and the transfer function \( H(s) \) is determined. NTF will be of the form

\[ NTF = \frac{s^4}{s^4 + d_1 s^3 + d_2 s^2 + d_3 s + d_4} \]

\[ NTF = \frac{1}{1 + H(s)} \]

\[ H(s) = \frac{1 - NTF}{NTF} \]

(3.8)

\[ NTF = \frac{N(s)}{D(s)} \]

where

\[ N(s) = s^4 \]

and

\[ D(s) = s^4 + d_1 s^3 + d_2 s^2 + d_3 s + d_4 \]
Equation (3.10) may be expressed as

\[ H(s) = \frac{D(s) - N(s)}{N(s)} \]

\[ H(s) = \frac{d_1 s^3 + d_2 s^2 + d_3 s + d_4}{s^4} \]

3.3.2 Design Example

This section details an example resulting in the full determination of \( H_1(s) \) and \( H_2(s) \). Let the NTF be a fourth-order Butterworth high-pass filter with \( A_s = 60\, dB \) and \( f_s = 650kHz \). The NTF is given by

\[ NTF = \frac{s^4}{s^4 + d_1 s^3 + d_2 s^2 + d_3 s + d_4} \]

In this MASH architecture we consider that \( H_1(s) \) is equal to \( H_2(s) \) and they are derived from the NTF of each stage using the equation (3.8) and the NTF of each stage is a Butterworth second order high pass polynomial. Let the overall noise transfer function be \( NTF_0 \) and let \( NTF_1 \) and \( NTF_2 \) be the noise transfer functions of first and second stages respectively.

\[ NTF_0 = NTF_1 \times NTF_2 \quad (3.9) \]

\( NTF_1 \) and \( NTF_2 \) are both equal and each of them represent a second order Butterworth high pass filter. Therefore from equation (3.7),

\[ NTF_1 = NTF_2 = \frac{s^2}{s^2 + a_1 s + a_0} = \frac{s^2}{s^2 + \sqrt{2} \Omega_c s + \Omega_c^2} \quad (3.10) \]

\( H_1(s) \) and \( H_2(s) \) can be computed by equation (3.8) resulting in

\[ H_1(s) = H_2(s) = \frac{1 - NTF}{NTF} = \frac{a_1 s + a_0}{s^2} = \frac{\sqrt{2} \Omega_c}{s} + \frac{\Omega_c^2}{s^2} \quad (3.11) \]
where $a_1 = 4.084 \times 10^6$ and $a_0 = 1.66 \times 10^{13}$

The magnitude responses for STF, NTF, $H(s)$, $H_1(s)$, and $H_2(s)$ can be determined by running the MATLAB file BUTTER.m in the Appendix.

3.3.3 Design for Implementing $H_1(s)$ and $H_2(s)$

The transfer functions $H_1(s)$ and $H_2(s)$ are implemented as biquad circuits as shown in figure 3.3.

The necessary design equations are also specified. Implementing $H_1(s)$ as a biquad circuit where

$$H_1(s) = H_2(s) = \frac{a_1 s + a_0}{s^2}$$

$$\frac{v_{out}}{v_{in}} = \frac{a_1 s + a_0}{s^2} = \frac{1}{s} + \frac{a_2}{s} + \frac{a_3}{s}$$

where $a_0 = a_2 a_3$

$$a_1 = \frac{1}{R_1 C_1} \quad a_2 = \frac{1}{R_2 C_2} \quad a_3 = \frac{1}{R_3 C_3}$$

![Figure 3.3 Architecture of a Single Second Order Stage in Continuous Time](image-url)
3.3.4 Component values for $H_1(s)$:

The values obtained from the matlab file BUTTER.m are as follows:

- $a_1 = 4.084 \times 10^6$
- $a_2 = 5.774 \times 10^6$
- $a_3 = 5.774 \times 10^6$

Let $R_1 = R_2 = R_3 = 1 \, \text{k}\Omega$

Then $C_1 = 5.6 \, \text{nF}$, $C_2 = 7.9 \, \text{nF}$, $C_3 = 7.9 \, \text{nF}$

By substituting all the component values and using only ideal op-amps $H_1(s)$ is implemented as shown in figure 3.3.

3.3.5 Design Example and Component values for $H_2(s)$:

$H_2(s)$ and $H_1(s)$ are both equal in this implementation. The necessary design equations for $H_2(s)$ are same as $H_1(s)$ which is as follows:

$$H_1(s) = H_2(s) = \frac{a_1s + a_0}{s^2} = \frac{\sqrt{2}\Omega_c}{s} + \frac{\Omega_c^2}{s^2}$$

$$\frac{v_{out}}{v_{in}} = \frac{a_1s + a_0}{s^2} = \frac{a_1}{s} + \frac{a_2}{s} \frac{a_3}{s}$$ where $a_0 = a_2 \times a_3$

$$a_1 = \frac{1}{R_1C_1} \quad a_2 = \frac{1}{R_2C_2} \quad a_3 = \frac{1}{R_3C_3}$$

The component values for $H_2(s)$ are obtained from the matlab file buttermash.m are as follows:

- $a_1 = 4.084 \times 10^6$
- $a_2 = 5.774 \times 10^6$
- $a_3 = 5.774 \times 10^6$

Let $R_1 = R_2 = R_3 = 1 \, \text{k}\Omega$
Then $C_1 = 5.6 \text{nF}$  $C_2 = 7.9 \text{nF}$  $C_3 = 7.9 \text{nF}$

3.3.6 **Design Example and component values for implementing $H_c(s)$:**

The output of first stage is

$$Y_1(s) = X(s)STF_1 + Q_1(s)NTF_1$$

This can also be written as

$$Y_1(s) = X(s)STF_1 + Q_{10}(s)$$

where $Q_{10}(s)$ is shaped quantization of the first stage

Also the output of second stage is given by

$$Y_2(s) = -Q_1(s)STF_1 + Q_2(s)NTF_2$$

From (3.3.2) the above equation can be written as

$$Y_2(s) = -Q_{10}(s)STF_1 + Q_2(s)NTF_2$$

Then the output of the compensation filter would be

$$Y_{2c}(s) = -H_c(s)STF_1 Q_{10}(s) + H_c(s)Q_2(s)NTF_2$$

In order to negate the quantization noise from the first stage it would be desirable to have

$$H_c(s)STF_1 = 1$$

The signal transfer function is given by,

$$STF = \frac{H(S)}{1 + H(S)} \tag{3.12}$$

Substituting the value of $H(s)$ from equation (3.11) in above equation (3.12), it can be shown that,

$$STF = \frac{\sqrt{2} \Omega_c s + \Omega_c^2}{s^2 + \sqrt{2} \Omega_c s + \Omega_c^2}$$

$$|STF(jw)|^2 = \frac{\Omega_c^2 + 2\Omega_c^2 w^2}{(\Omega_c^2 - w^2)^2 + 2\Omega_c^2 w^2}$$
\[ |STF(0)|^2 = 1 \]

Substituting the above value in equation (3.6), it can be shown that,

\[ H_c(s) = 1 \]

At \( \Omega_c \), \( |STF(j\Omega_c)|^2 = \frac{3}{2} \)
CHAPTER IV
RESULTS AND SIMULATIONS

4.1 SPICE Simulation

The challenging aspect of designing the sigma-delta modulators is the simulation of performance at the device level. The circuit simulation, yielding the bit stream frequency spectrum and the signal to noise ratio, will be based on Mentor Graphics Eldo SPICE simulations. Mentor Graphics is an efficient simulated program with integrated circuit emphasis Eldo (SPICE) tool to deal with different signal techniques. Device-level simulation is the ultimate to produce accurate estimation that is more dependable before fabrication. It is, therefore, desirable that the simulations and optimization of sigma-delta data converters be realized at the device level.

The input sinusoidal signal of the simulation was chosen to be 0.8v, 5kHz, which was very much lower than the desired bandwidth of the modulator. Because the input signal frequency is very much smaller than the sampling frequency which is in the megahertz range, the simulation time was very long.

AC steady state analysis was done on the circuit and the magnitude response of H(s) and STF was obtained. Obtained response was similar to the desired magnitude response from the Matlab BUTTER.m. Figure 4.1 shows the magnitude frequency response of overall NTF. The overall NTF is computed by multiplying the NTF’s of each stage as shown in equation (3.9). Figure 4.2 shows the signal band detail of the figure 4.1. It can be observed that the cut-off frequency of overall NTF is not 650kHz whereas the cut-off frequency for each NTF is 650kHz which is evident from the figure 4.3. Figure 4.4 shows the magnitude frequency response of the overall STF. Figure 4.5 is same as
figure 4.4 except for a narrower band of frequencies from 10k to 100k. It can be observed that the response of STF is constant across the signal band from DC to 20kHz except for a slight variation of 0.01dB as evident from the figure.

A 5kHz sine wave is used for the input of the modulator and the transient analysis was done. The output was a bit stream of +1 and -1. Simulations are run for transient analyses for 50ms and the frequency spectrum is obtained from FFT. The specifications used for obtaining FFT are as follows:

- **TSTART** = 0
- **TSTOP** = 50m
- **F_S** = 5MEG
- **NbPTS** = 250000
- **FREF** = 5000
- **FMIN** = 0
- **FMAX** = 500000

Signal to Noise Ratio is computed from the obtained frequency spectrum.

### 4.2 Simulation results of Fourth order Butterworth MASH architecture

Figures 4.6 represents two stage fourth order Butterworth delta sigma data converter. The Mentor-Graphics schematic shown in the figure 4.6 is experimented with a sine wave input of 0.8v and a frequency of 5kHz for various cutoff frequencies of the Butterworth highpass filter. It is observed that from many simulations that the best results (in terms of SNR) are obtained for a cut-off frequency 650kHz. The component values mentioned in the schematic figure 4.6 are calculated for cut-off frequency of 650kHz. The sampling frequency f_s is taken as 5MHz. Simulations are run for transient analyses for
50ms and the frequency spectrum is obtained from FFT. The obtained magnitude spectrum is shown in figure 4.8. Also the magnitude spectrum of the first stage of the MASH architecture is shown in figure 4.7. This clearly indicates that the performance of the MASH architecture is superior to the Butterworth second order delta-sigma ADC. The MATLAB implementation for given architecture is shown in the Appendix. Signal to Noise Ratio is computed from the obtained magnitude spectrum and it is noted as 65dB for the signal band from DC to 20kHz for the Butterworth MASH architecture.

Figure 4.1 Magnitude Frequency response of Overall NTF.
Figure 4.2 Signal Band detail of overall NTF.

Figure 4.3 Magnitude Frequency response of NTF of each stage of the Butterworth MASH Architecture.
Figure 4.4 Magnitude Frequency response of Overall STF.

Figure 4.5 Signal Band detail of Overall STF.
Figure 4.6 Butterworth Fourth Order MASH Delta-Sigma Data Converter.
Figure 4.7 Magnitude Spectrum Analysis of first stage of Butterworth MASH Architecture.
Figure 4.8 Magnitude Spectrum Analysis of Butterworth MASH Architecture.
4.3 Simulation results of 2-stage Fourth Order Conventional Mash Architecture

A 2-stage Fourth Order Conventional Mash Architecture was designed to compare the performance with Fourth order Butterworth MASH architecture. The same input values have been used for this architecture as was used for Butterworth MASH architecture so that a fair comparison is done. FFT is done on the digital output obtained after running the simulation for 50ms. The SNR obtained in this case is 61dB, which gives a resolution of 10 bits. SNR was computed for Audio band (0-20kHz). This clearly shows that the performance of Butterworth MASH architecture is superior to Conventional MASH architecture. The schematic and the spectrum of 2-stage Fourth Order Conventional MASH architecture are shown below in figures 4.10 and 4.11 respectively.
Figure 4.9 Conventional Second-Order Delta-Sigma Architecture.
Figure 4.10 Fourth Order Conventional MASH Architecture.
Figure 4.11 Magnitude Spectrum Analysis of 2-Stage Fourth Order Conventional MASH Architecture.
CHAPTER V

CONCLUSIONS AND FUTURE WORK

This thesis discussed the design and implementation of Butterworth MASH delta-sigma ADC. This architecture as discussed was a simple circuit with no complexity as it did not make use of any switched capacitors. It’s a pure RC circuit involving three integrators and Operational Amplifiers. Two special cases are investigated: first, where each of the two stages implemented a second-order Butterworth highpass NTF, the second, where the parameters are set such that the overall response was that of conventional fourth order MASH architecture. The proposed architecture was compared to the Fourth Order Conventional MASH architecture and simulation results of both the architectures were presented. From the simulation results it was clearly noted that the performance of Butterworth MASH architecture is enhanced when compared to the conventional MASH architecture. The SNR of Butterworth MASH architecture is obtained to be $65dB$ (for the signal band from DC to $20kHz$) having an 11 bit resolution and the SNR of Fourth Order Conventional MASH is obtained to be $61dB$ (for the signal band from DC to $20kHz$) having a 10-bit resolution.

RC implementation of Butterworth MASH architecture can be implemented with real components using practical double ended operational amplifiers. This ensures the design is in a balanced environment and it might lead to a noisy environment. Using Mentor Graphics software, this project can be implemented with real components and complete IC design to tape out for fabrication through MOSIS.
LIST OF REFERENCES


MATLAB Code for Design of MASH Implementation – BUTTER.m

% Comparison between a conventional second-order delta-sigma data
% converter and a second-order Butterworth delta-sigma data converter.
%
fc = 650E3;
w = 2*pi*fc;
k1 = sqrt(2)*w;
k2 = w;
%
% Conventional second-order:
cc = 1E6;
NTCN = [1 0 0];
NTCD = [1 cc cc*cc];
STCN = cc*cc;
STCD = NTCD;
%
% Butterworth second-order:
NTBN = [1 0 0];
NTBD = [1 k1 k2*k2];
STBN = [k1 k2*k2];
STBD = NTBD;
%
% Calculate the magnitude frequency responses:
ff = 1.500E3;

ww = ff*2*pi;

NTC = freqs(NTCN,NTCD,ww);

NTC = NTC / NTC(500E3);

NTCD = 20*log10(abs(NTC));

NTB = freqs(NTBN,NTBD,ww);

NTB = NTC / NTC(500E3);

NTBD = 20*log10(abs(NTB));

STC = freqs(STCN,STCD,ww);

STC = STC / STC(500E3);

STCD = 20*log10(abs(STC));

STB = freqs(STBN,STBD,ww);

STB = STB / STB(500E3);

STBD = 20*log10(abs(STB));

% Calculate the ideal sine squared response:

STS = (sin(pi .* ff/1.0E6))^2;

STSDB = 20*log10(STS);

% Plot the NTFs:

semilogx(ff,STSDB,'k-',ff,NTCD,'b--',ff,NTBD,'r-.'

xlabel('Frequency in Hz')

ylabel('Magnitude in dB')

grid

axis([1 500E3 -200 5])
title('NTF: Sine black solid, Conventional blue dashed, Butterworth red dashdot')
pause
close
%
semilogx(ff,STSDB,'k-',ff,NTCDB,'b--',ff,NTBDB,'r-.')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([1E3 100E3 -100 -20])
title('NTF: Sine black solid, Conventional blue dashed, Butterworth red dashdot')
pause
close
%
semilogx(ff,STSDB,'k-',ff,NTCDB,'b--',ff,NTBDB,'r-.')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([10E3 500E3 -60 5])
title('NTF: Sine black solid, Conventional blue dashed, Butterworth red dashdot')
pause
close
%
semilogx(ff,STSDB,'k-',ff,NTCDB,'b--',ff,NTBDB,'r-.')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([50E3 500E3 -20 5])
title('NTF: Sine black solid, Conventional blue dashed, Butterworth red dashdot')
pause
close
%

% Plot the STFs:
semilogx(ff,STCDB,'b--',ff,STBDB,'r-.')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([1 500E3 -10 5])
title('STF: Conventional blue dashed, Butterworth red dashdot')
pause
close
%
semilogx(ff,STCDB,'b--',ff,STBDB,'r-.')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([5E3 30E3 -0.1 0.5])
title('STF: Conventional blue dashed, Butterworth red dashdot')

pause

close

fc = 650E3;

wc = 2*pi*fc;

NUM = [ 1 0 0 0 0];

DEN = [ 1 2*sqrt(2)*wc 4*wc^2 2*sqrt(2)*wc^3 wc^4];

NTF = tf(NUM,DEN);

ff = 1:500E4;

ww = ff*2*pi;

NTB = freqs(NUM,DEN,ww);

NTB = NTB ./ NTB(500E4);

NTBDB = 20*log10(abs(NTB));

% Plot the NTFs:

semilogx(ff,NTBDB,'-k')

xlabel('Frequency in Hz')

ylabel('Magnitude in dB')

grid

axis([1E3 6E6 -200 30])

title('NTF: Butterworth dash')

pause

close

%
semilogx(ff,NTBDB,'-k')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([1E5 200E4 -100 20])
title('NTF:Butterworth dash')
pause
close

% %
semilogx(ff,NTBDB,'r-.')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([10E3 500E3 -60 5])
title('NTF:Butterworth red dashdot')
pause
close

% 
semilogx(ff,NTBDB,'r-.')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([50E3 500E3 -20 5])
title('NTF: Butterworth red dashdot')
pause
close
NUM = [ 1 0 0 0 0];
DEN = [ 1 2*sqrt(2)*wc 4*wc^2 2*sqrt(2)*wc^3 wc^4];
NTF = tf(NUM,DEN);
ff = 1:500E4;
ww = ff*2*pi;
NTB = freqs(NUM,DEN,ww);
NTB = NTB ./ NTB(500E4);
NTBDB = 20*log10(abs(NTB));
% Plot the NTFs:
semilogx(ff,NTBDB,'-k')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([1E3 6E6 -200 30])
title('NTF: Butterworth dash')
pause
close
%
semilogx(ff,NTBDB,'-k')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([1E5 200E4 -100 20])
title('NTF:Butterworth dash')
pause
close

semilogx(ff,NTBDB,'r-.')
xlabel('Frequency in Hz')
ylabel('Magnitude in dB')
grid
axis([10E3 500E3 -60 5])
title('NTF:Butterworth red dashdot')
pause
close

%