Case Study on Product Quantization Errors

Thomas J. Plummer
November 24, 2004

EEN 536
University of Miami

Contents

Abstract

Digital filters are utilized in the post processing stage of the MPEG-4 Audio3 Standard. A particular one of these filters is a high pass filter with high frequency emphasis to enhance the 'brightness' quality of speech. This filter will be designed according to the given fourth order transfer function (TF) in its Direct Form II second-order sections. Due to the finite wordlength system limitations, erroneous effects such as coefficient quantization and product quantization often occur. In particular, the later will be examined and Lp scaling distribution among second order sections will be implemented to reduce these unwanted quantization effects.

Introduction

When designing filters in the digital domain, it is relatively simple to produce a z-domain transfer function that performs exactly to the specifications that have been given. However, when the system is implemented, it must be in a finite word-length system. The finite word-length effects include the quantization of coefficients. Many times, digital IIR filters have poles and zeroes that are very close to the unit circle or poles and zeroes that are clustered together in order to achieve an idealistic frequency response. However, when quantized, a coefficient that is very close to the unit circle or clustered with others may be pushed outside the unit circle. This makes the filter unstable and effectively makes it useless. To be less sensitive to coefficient quantization, the highpass IIR filter will be implemented as a cascade of two second-order Direct Form II sections where each section is known as a bi-quad. These bi-quads realize each pole/zero pair independently of the others, therefore an error in one section will not affect the coefficients in the other. The Direct Form II realization is also a prime candidate for its low computational burden and minimum number of components. The transfer function composed of second order sections in the cascade form is represented by the below:

$$H(z) = \prod_k \frac{b_{ok} + b_{1k}Z^{-1} + b_{2k}Z^{-2}}{1 - a_{1k}Z^{-1} - a_{2k}Z^{-2}} \qquad (1)$$

One Direct Form section is realized as:

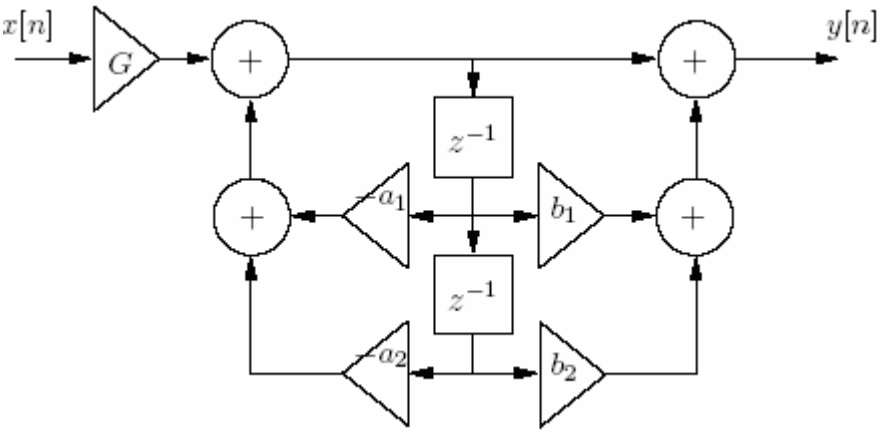

Figure 1: Second order Section Direct Form II

The multiplication function of the coefficients adds error to the filter in the form of product quantization. When two signals of bit length B=16 bits are multiplied together the result can be of length 2*B=32 bits, therefore truncation or rounding must be

implemented to maintain the uniform bit length of B=16bits. The resulting errors pass through the entire filter giving rise to output noise referred to as *round-off noise* or *round-off error*. This product round off can be modeled as the introduction at the points of error as a white noise source with magnitude dependent on the quantization resolution. This noise source will directly decrease the Signal to Noise (S/N) ratio of the filter.

This is where the Lp scaling factor is an important design factor in reducing these erroneous inner filter quantization effects. It is evident that using finite wordlengths limit the dynamic range of the filter. If this limit is exceeded, overflow occurs and severely distorts the filter output. Thus, the signal must be scaled within the filter. However, if the signal is scaled too much, then again, the S/N ration suffers as the signal level is brought closer to the noise floor.

Problem Statement

A high pass filter for use in the post-processing unit of an MPEG-4 Audio3 decoder typically takes the following form:

$$H_{HPF}(z) = K_{HPF}\frac{1+a_{11}z^{-1}+a_{12}z^{-2}}{1+b_{11}z^{-1}+b_{12}z^{-2}}\frac{1+a_{21}z^{-1}+a_{22}z^{-2}}{1+b_{21}z^{-1}+b_{22}z^{-2}}, \quad (2)$$

Where,

Coefficient	Value
K_{HPF}	+1.1
a_{11}	-1.998066423746901
a_{12}	+1.00
b_{11}	-1.962822436245804
b_{12}	+0.9684991816600951
a_{21}	-1.999633313803449
a_{22}	+0.9999999999999999
b_{21}	-1.858097918647416
b_{22}	+0.8654599838007603

Table 1: Coefficient values for equation (1)

This case study will develop a Direct Form II second order sections implementation of this digital filter. Once developed, the effects of finite wordlength on product qauntization will be examined. Also, this case study will include an optimal Lp scaling distribution between the bi-quads.

Analysis

Developing the Appropriate Filter Structure

Since the transfer function is given as two-second order sections it would make sense to design the filter as a set of two-second order sections. It was earlier stated that the direct form II structure of second order sections provides improved performance in the coefficient sensitivity and keeps the number of delays to a minimum. Therefore, it will be used. Using the standard direct form II structure, each section of the filter can be developed by inspection and the complete filter will simply be the two sections connected in cascaded form. The gain of the filter will be placed at the beginning. The realization of the filter can be seen here:

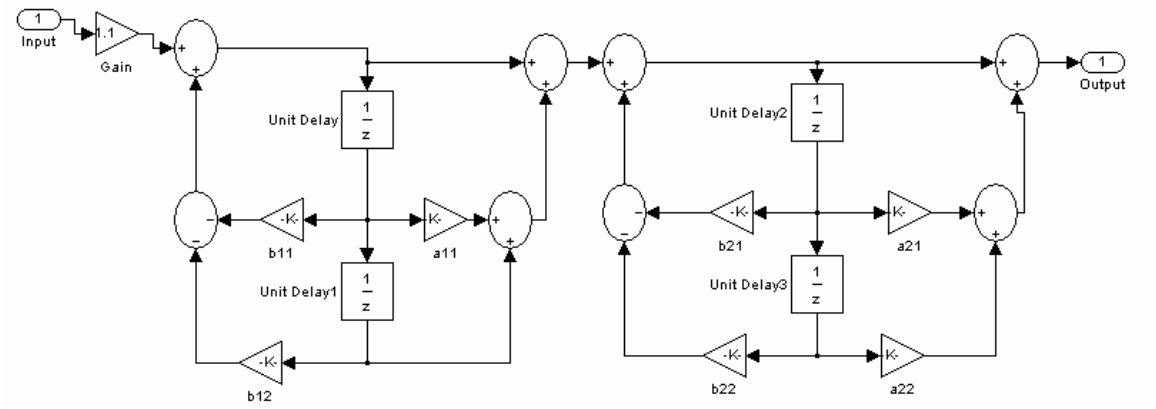

Figure 2: Realization of given transfer function as cascaded SOS direct form II filter

Note that the value of a_{12} does not need a multiplier to be represented because the value of this coefficient is exactly one. This means that there will be one less multiplier to contend, thus the calculation overhead and number of components is reduced. Also make note of the poles on the left for negative feedback and their corresponding closest zeroes on the right of each second order section.

General Second Order Section Direct Form II Filter Analysis

Once the filter structure has been established, an analysis of the filter without any quantization must be completed. This will ensure that the filter behaves as expected and will set the benchmark to which further analysis will be compared. In MATLAB, the transfer function can be derived from the Simulinx realization using the DLINMOD command. The following frequency/phase response, pole/zero plots, and impulse response are representative of how this filter will behave with no quantization effects.

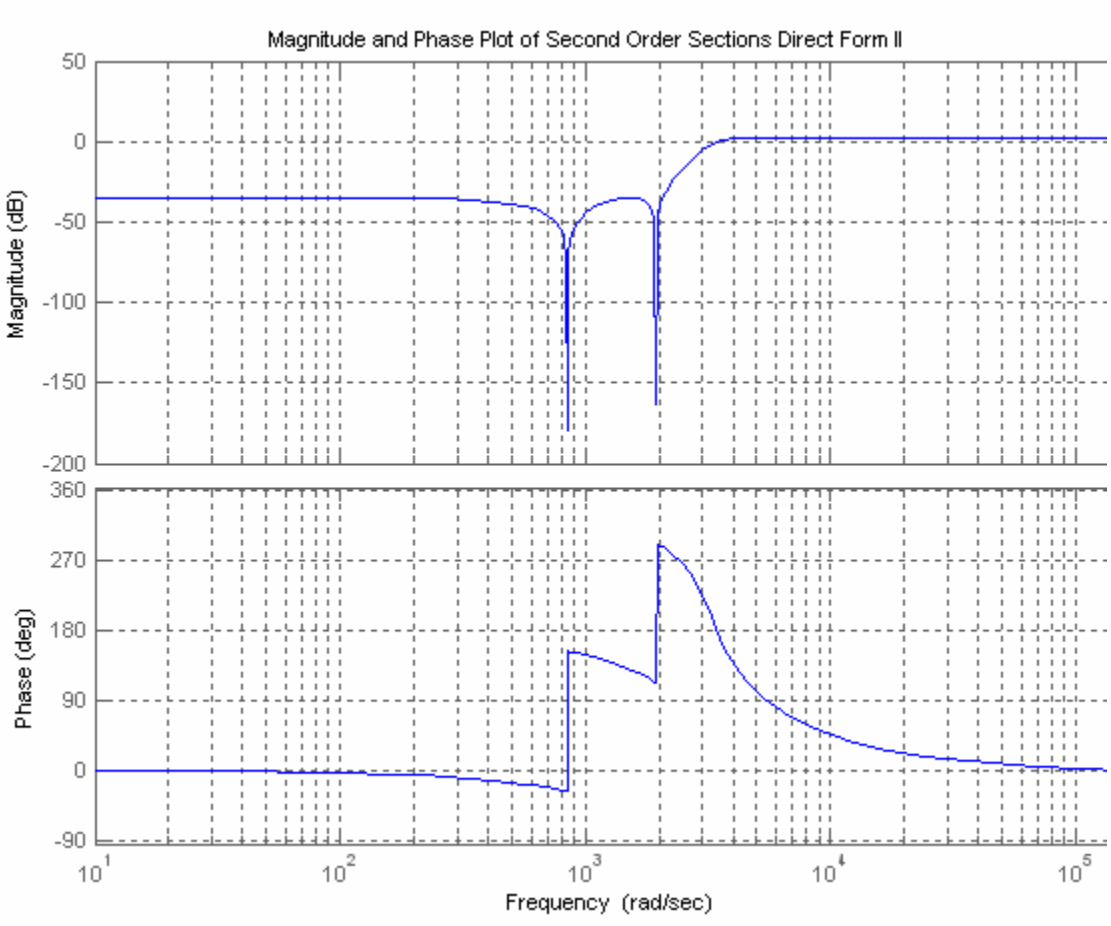

Figure 3: Magnitude and phase response of filter with no quantization effects

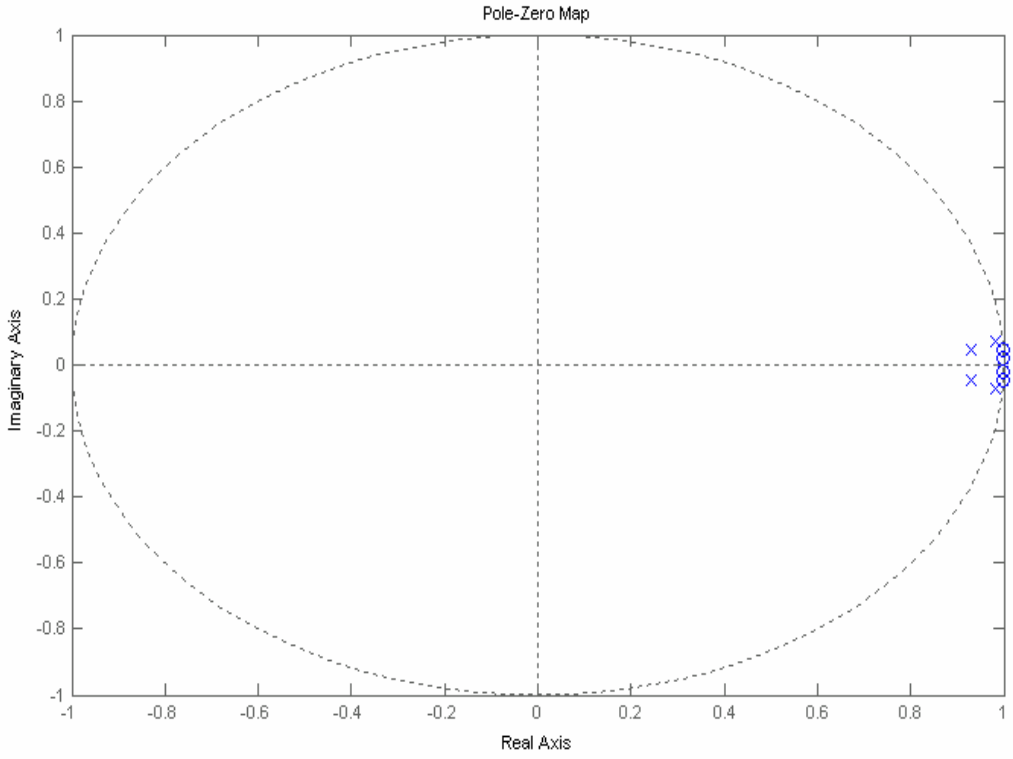

Figure 4: Pole/zero locations of the filter with no coefficient quantization

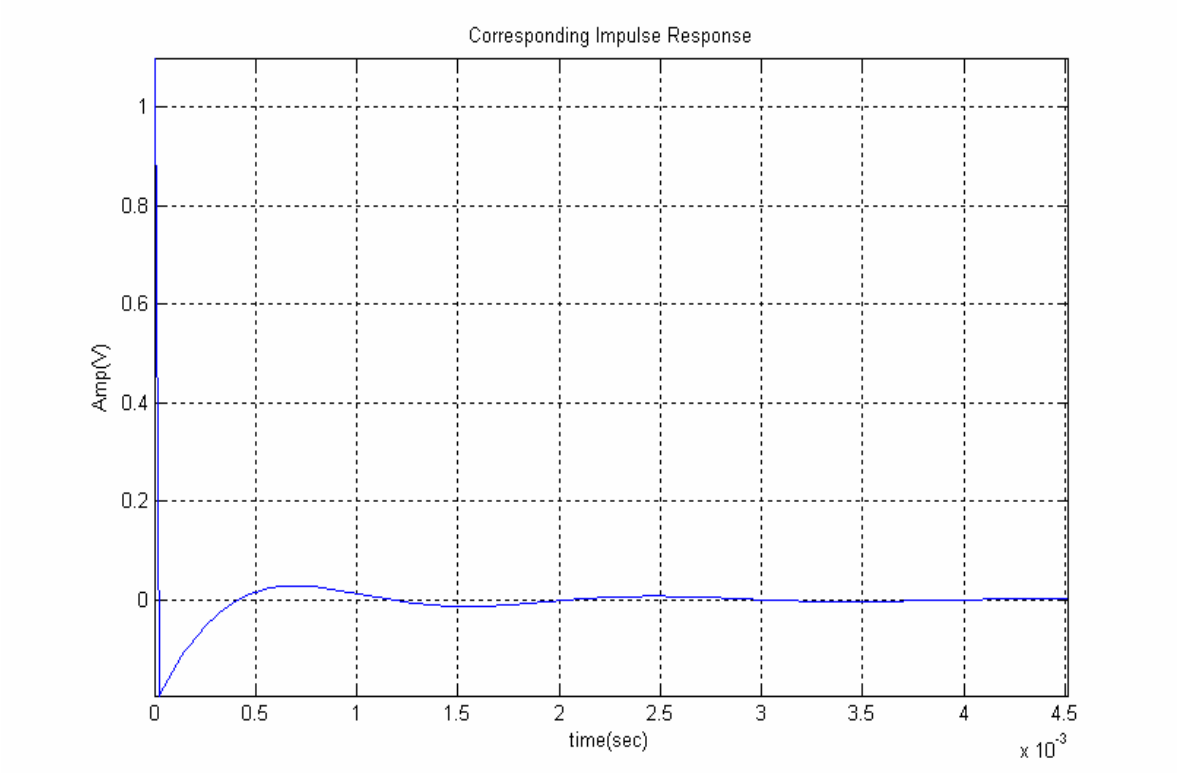

Figure 5: Corresponding impulse response of implemented high pass filter

Product Quantization Analysis

When the coefficients of a filter are used to multiply the incoming signals, there usually are more bits in the result than the finite wordlength register can store, therefore the result needs to be rounded. This introduces error that can be represented in the filter as white noise sources. Each multiplier produces its own noise source. If the following conditions are true:

- Quantization level q is small~ dynamic range and wordlength dependent
- I/P sequence $x(n)$ is wideband and fluctuates rapidly and over several quantization levels between sample

Then it is reasonable to assume each noise source:

- Is a realization of a wide-sense stationary process
- Is uniformly distributed over one quantization level q
- Is uncorrelated at two different sample instances
- Is uncorrelated with the other noise sources, its own i/p and the total system i/p

Then from probability theory for wide sense stationary systems, each noise source can be modeled as a white noise source and the sources can be summed to a common node. Also, for Sign Magnitude Rounding (SMR) quantization scheme that is being used, the following is true, where *e* represents the error:

$$
\begin{aligned}
& q = 2^{-\#Bits} \\
& \text{Range} = [-q/2, +q/2] \\
& \mu_e = 0 \\
& \sigma_e^2(m) = \frac{q^2}{12}\delta(m) \\
& \phi_{ee}(m) = \frac{q^2}{12}\delta(m)
\end{aligned}
\qquad (3)
$$

Since all of the noise sources are uncorrelated, the effects of them can be summed and added to each section at one node, so the summed noise source have a mean and variance of:

$$
\begin{aligned}
& \mu_e = 0 \\
& \sigma_{ee}^2 = \sum_{i=0}^{M} \sigma_{b,i}^2 + \sum_{j=0}^{N} \sigma_{a,j}^2
\end{aligned}
\qquad (4)
$$

Where M is the number of pole coefficient multipliers and N is the number of zero coefficients multipliers. Now the noise sources can be modeled in Simulinx as shown below.

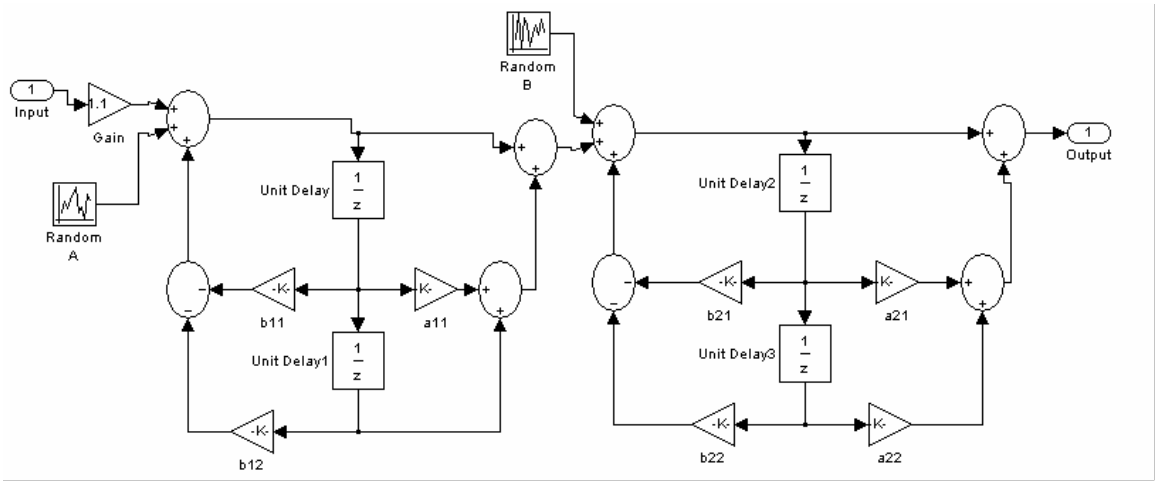

Figure 6: Filter structure with noise sources added to simulate quantization noise

Notice this filter set up has two white noise sources added to simulate the product quantization. The noise sources will always have a mean of zero, since the SMR scheme is being used, but the variance will change according to the quantization level- *q* and the number of multiplier in each bi-quad. For 16-bit FXP SMR, the first noise source will have a covariance of:

$$\sigma_{ee}^2 = (M+N+1)\frac{q^2}{12} = (2+1+1)\frac{(2^{-16})^2}{12} = \mathbf{7.7610e-011} \qquad (5)$$

And the second bi-quad will have a covariance of:

$$\sigma_{ee}^2 = (M+N+1)\frac{q^2}{12} = (2+2+1)\frac{(2^{-16})^2}{12} = \mathbf{9.7013e-011} \qquad (6)$$

The corresponding results are presented in the next three figures.

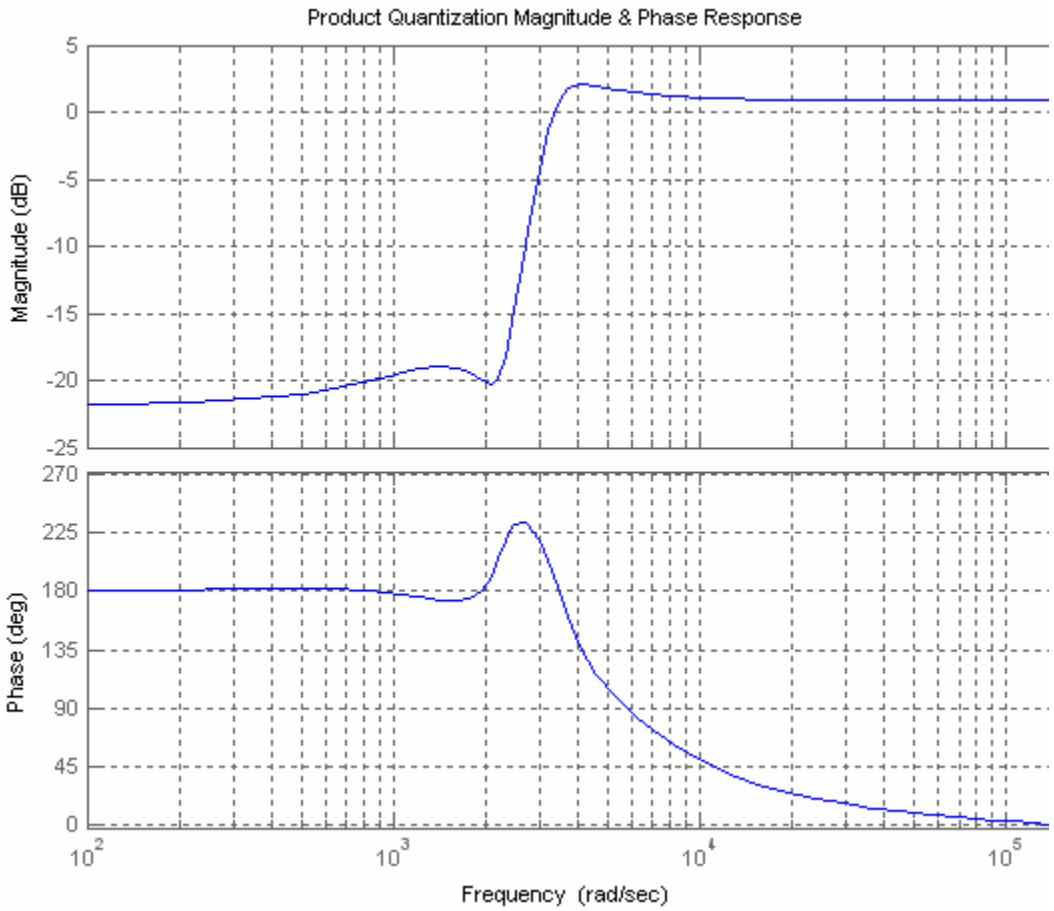

Figure 7: Magnitude and phase response with 16-bit FXP SMR product quantization

The frequency/phase response are slightly different from the original however there is no serious distortion at SMR of 16-bit wordlength due to product quantization.

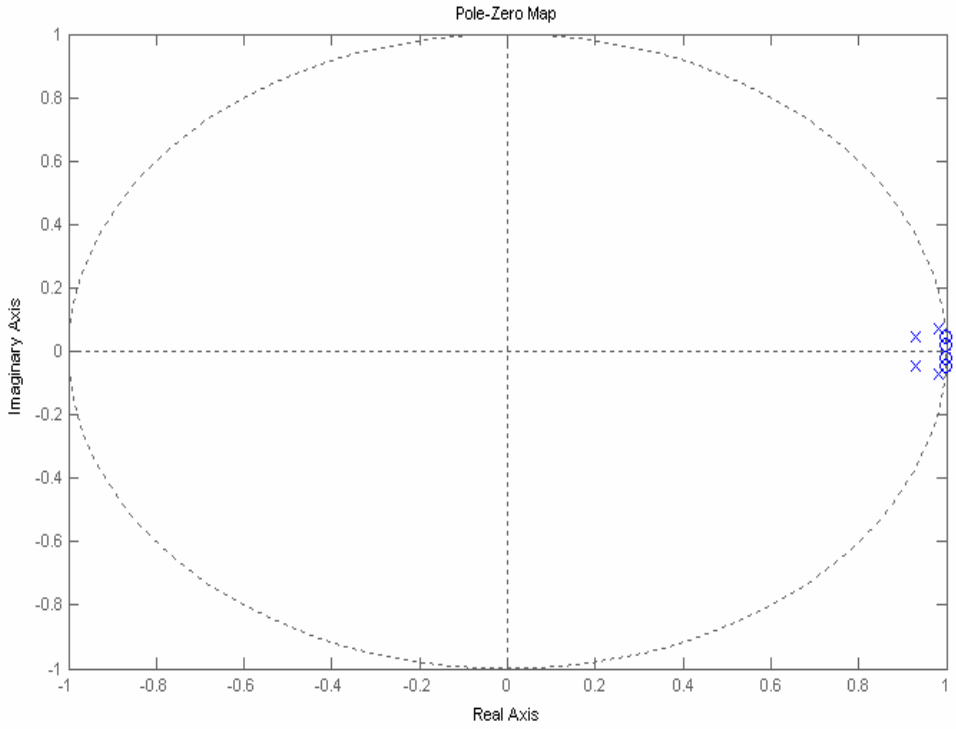

Figure 8: Pole /zero map of filter with 16-bit FXP SMR product quantization

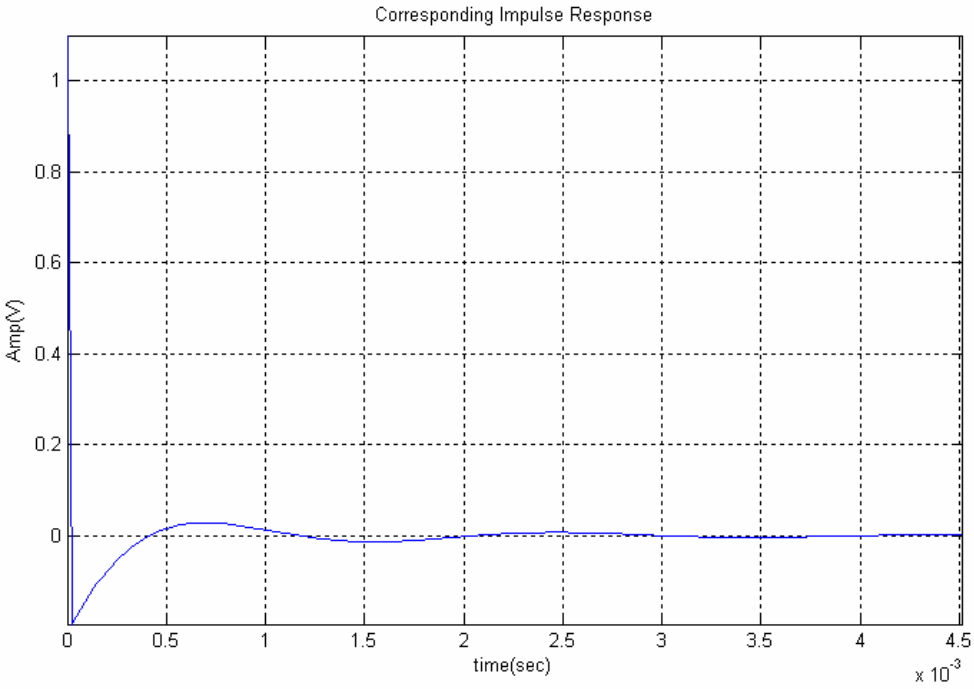

Figure 9: Impulse response of filter 16-bit FXP SMR product quantization.

The results are almost exactly the same as the pole/zero map and impulse response with no product quantization noise. Therefore, it can be noted that 16-bits provides enough S/N ratio that the effects of product quantization will not be readily noticed. For an 8-bit FXP SMR scheme, the covariance can be calculated similarly.

For the first noise source:

$$\sigma_{ee}^2 = (M+N+1)\frac{q^2}{12} = (2+1+1)\frac{(2^{-8})^2}{12} = \mathbf{5.0863e-006} \qquad (7)$$

And for the second:

$$\sigma_{ee}^2 = (M+N+1)\frac{q^2}{12} = (2+2+1)\frac{(2^{-8})^2}{12} = \mathbf{6.3578e-006} \qquad (8)$$

The corresponding results for product quantization using 8-bit FXP SMR is presented in the next three figures.

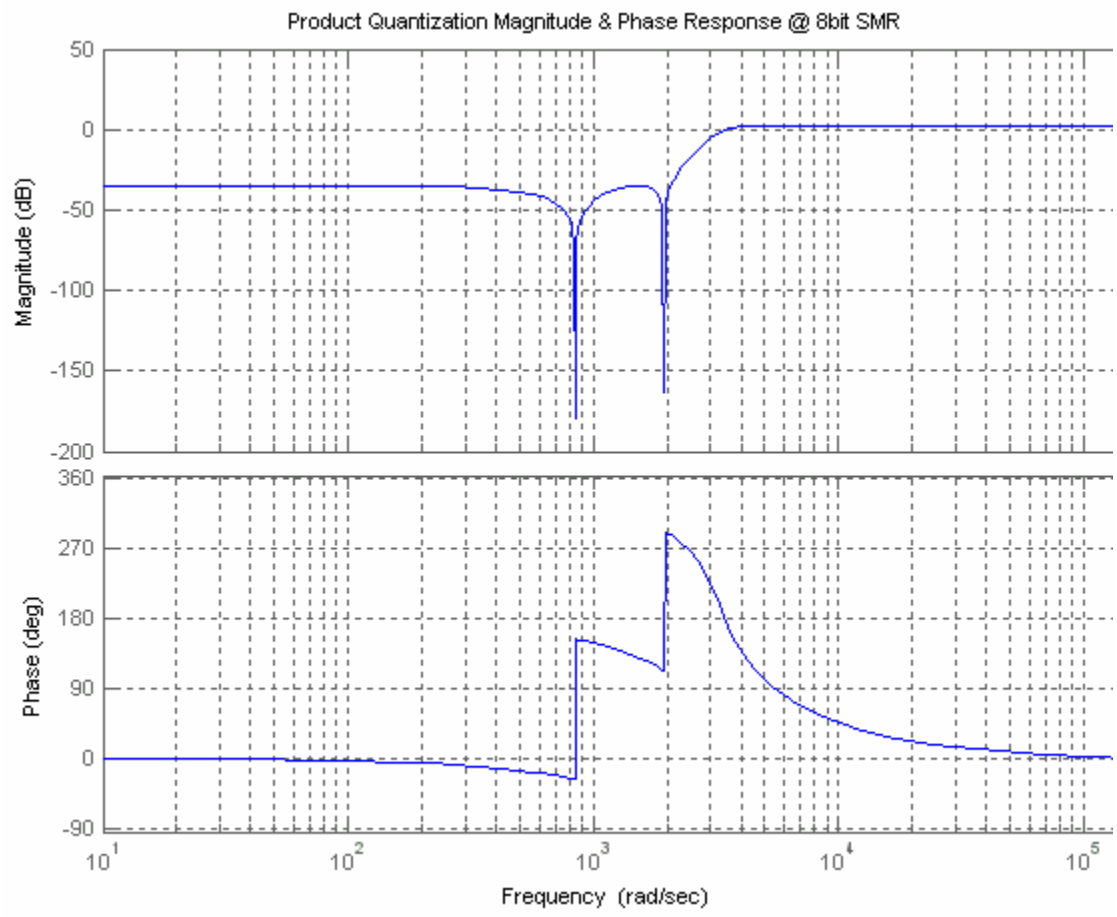

Figure 10: Magnitude and phase response with 8-bit FXP SMR product quantization

For 8-bit FXP SMR the frequency/phase response and the pole/zero map are surprisingly similar to the original.

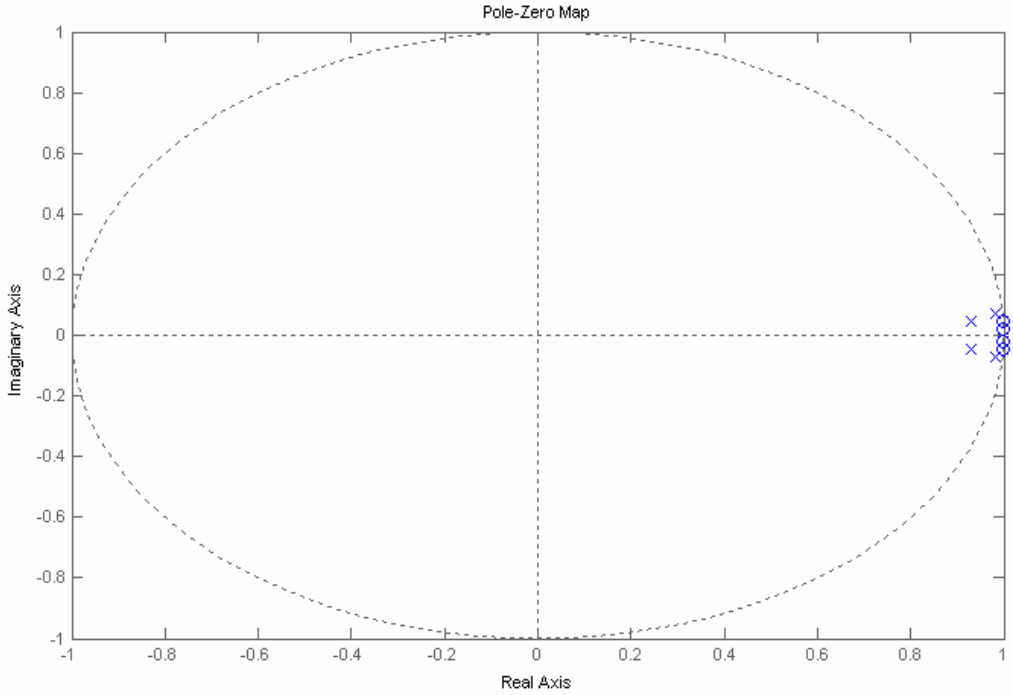

Figure 11: Pole /zero map of filter with 16-bit product quantization

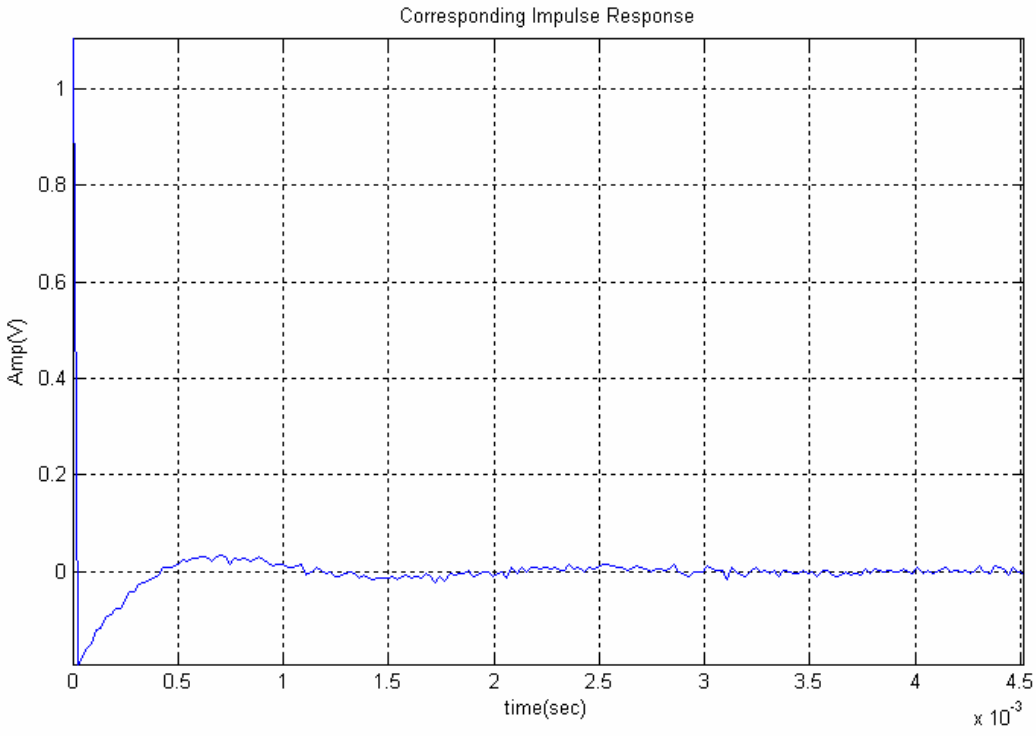

Figure 12: Impulse response of filter with 8-bit FXP SMR product quantization

Here a small amount of noise has been added to the impulse response. This is comparable to the amount of noise that would occur from *round-off errors* in an 8-bit FXP SMR scheme due to product quantization.

Lp Scaling

With a finite wordlength, the quantization of a signal implies there is an inherent dynamic range available. This range is directly proportional to the number of bits used. Overflow occurs if this range is exceeded at any time, which severely distorts the output of the filter. Therefore, scaling must be implemented to ensure the signal stays within the dynamic range of the registers at all points in the filter. However, with too much scaling noise from *roundoff errors* will begin to overcome the signal and reduce the S/N ratio at the filter output. Consider the following:

Given that we are dealing with a bounded input bounded output filter, there is an arbitrary node k with output $w_k(n)$ within a TF $H_k(z)$ that has an initial scaling of λ and input sequence x . Then a sufficient condition for $|w_k(n)| \le M_{overflow}$ is:

$$\lambda x_{\max} \sum_{m=-\infty}^{\infty} |h_k(m)| \le M_{overflow} \Leftrightarrow \lambda x_{\max} \le \frac{M_{overflow}}{\sum_{m=-\infty}^{\infty} |h_k(m)|} \tag{9}$$

To ensure all internal signal values remain bounded by $M_{overflow}$ ~ normally one, we need (9) to be true for all TFs $H_k(z)$, thus:

$$\lambda x_{\max} \le \frac{M_{overflow}}{\max_k \sum_{m=-\infty}^{\infty} |h_k(m)|} \tag{10}$$

Besides considering only the maximum value of the absolute system response, filter design engineers are accustom to a more efficient method of considering the size of the frequency response through the notion of a norm. In particular the Lp-norm, where the Lp-norm of $H_k(\omega)$ is:

$$\|H\|_p = \left[\frac{1}{2\pi} \int_0^{2\pi} |H(\omega)|^p \, d\omega \right]^{1/p}, \; p \ge 1 \tag{11}$$

With simple calculus it can be realized that L_1 will provide the average size of the TF and L_∞ will provide the maximum size of the TF.

Now to ensure that all signals remain bounded by $M_{overflow}$, using Lp-norm we therefore need:

$$\lambda x_{\max} \le \frac{M_{overflow}}{\mathbf{max}_k \left\| H_k \right\|_p} \qquad (12)$$

This is referred to as Lp-*scaling*.

In our case of two cascaded second order sections, the same Lp-scaling technique is desired but in a distributed sense among the two sections. First, we must consider our two sections individually, in order to determine which section contains pole/zero pairs closer to the unit circle which produces the most *round-off errors* in the system. Then we must distribute the scaling in order of highest scaling to least across the two sections in order of most to least sensitive bi-quad in quantization. Manually you would choose the gains (or scaling parameters) K_k such that:

$$\prod_{l=1}^{k} K_l \, x_{\max} \le \frac{M_{overflow}}{\left\| \prod_{l=1}^{k} H_l \right\|_p}, \qquad k = \overline{1, L}. \quad \text{Where } L \text{ is the number of sections} \qquad (13)$$

Digital filter designers most commonly use L_2-*scaling* and L_∞-*scaling* methods when designing digital filters. L_2-*scaling* provides improved protection against quantization noise while L_∞-*scaling* provides improved protection against overflow. The below realizations implement the distributed scaling of our filter for both.

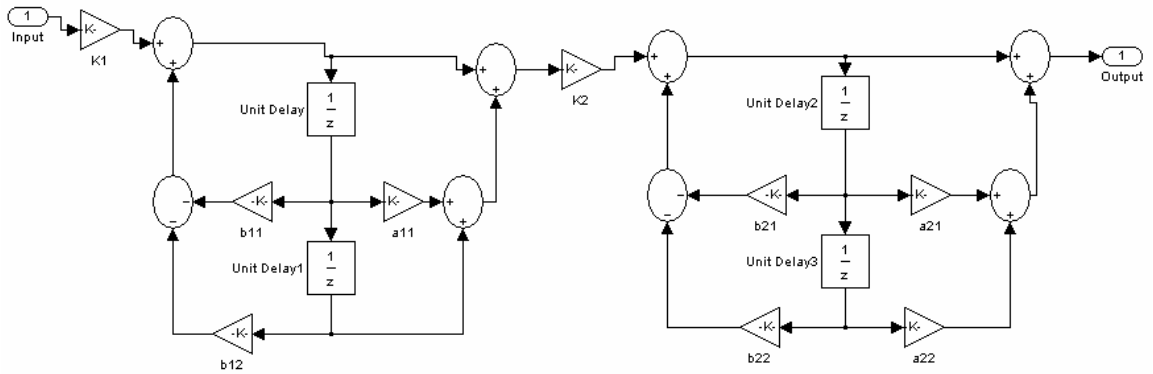

Figure 13: Distributed Lp-scaling realization

Where the gain distributions in the next table are found by analyzing MATLAB function SS2SOS with 'TWO' and 'INF' in the scaling parameter. Careful rearranging in Simulinx was performed with *round-off errors* like in the product quantization analysis using FXP SMR to find the optimum position of these gains corresponding to figure 13.

	L_2-scaling	L_∞-scaling
K1	2.7082	2.6880
K2	(0.0189)*21.4932	(0.0023)*174.6893

Table 2: Lp-scaling distribution gains

With the gains distributed like they are in Figure 13 and Table 2, the corresponding frequency/phase response, pole/zero map and impulse response for each case matched that of the original. However, the gain placement with the small number in parenthesis in Table 2, gave more *round-off noise* in both scaling cases when multiplied with the value in K1 as opposed to K2. The noise magnitude in the system impulse response in this sense was much greater with L_∞-scaling than for L_2-scaling, which was expected for L_2-scaling provides improved S/N ratio and L_∞-scaling provide better overflow protection and we are dealing with added noise.

Conclusions

When dealing with finite wordlength IIR filter design, the second order section Direct Form II makes a superb candidate. In terms of coefficient sensitivity in the process of quantizing the infinite wordlength coefficients, the second order sections realize each pole/zero independently. Therefore coefficient quantization errors only affect the independent pole/zero pair corresponding to that section. Also, the Direct Form II is easy to implement through inspection and requires the least amount of components with low computation burden. The *round-off errors* introduced by the multipliers produce unwanted noise in the output, thus decreasing S/N ratio. When the number of bits used to quantize the signal is high, this is usually not a problem as seen in the case using FXP SMR at 16-bits. However, as constraints get less flexible on the number of bits that may be used, the *round-off noise* can overtake the filter as seen starting to take effect in the case using FXP SMR at 8-bits. To overcome erroneous quantization effects proper Lp-scaling techniques are utilized. In particular, when aiming to improve S/N ratio the L_2-scaling technique can be implemented distributed across each bi-quad according to maximum signals at a shared node within each section. If overflow is the problem, than the L_∞-scaling distribution approach is effective in the same manner. The key to this case study was making use of the fine tools MATLAB and Simulinx have to offer in simulating the MPEG-4 Audio3 post processing filter and analyzing the effects of product quantization and scaling to determine and understand the optimum realization.

References

Carnegie, Mellon, “Control Tutorial for MATLAB, Digital Control Example: Designing Pitch Controller using State Space Method”, The University of Michigan, August 1997. [Online] Available: http://www.engin.umich.edu/group/ctm/examples/pitch/digPCSS.html

HELP, MATLAB, “Version 6.5.0.180913a Release 13,” The Math Works Inc. June 18, 2002.

Information Technology- Very Low Bitrate Audio-visual Coding, Part 3 Audio, MPEG Working Group, International Standards Organization International Electrotechical Commission (ISO/IEC) Std. ISO/IEC FCD 14 496-3 Subpart 1, May 1998. [Online]. Available: http://www.mp3-tech.org/programmer/docs/ISO_14496-3.pdf

K. Premaratne, “EEN 536 Digital Filter Structures Class Notes,” Department of Electrical and Computer Engineering, University of Miami. Pgs. 1-31. October 2004.

S. Battista, F. Casalino, and C. Lande, “MPEG-4: a multimedia standard for the third millennium 1,” IEEE Multimedia, vol. 6, no. 4 Oct. /Dec. 1999. [Online] Available: http://www.computer.org/multimedia/articles/MPEG4_3.htm

S. III. O. Julius, “Introduction to Digital Filters with Audio Applications,” Center for Computer Research in Music and Acoustics (CCRMA), Stanford University, May 2004 [Online] Available: http://ccrma-www.stanford.edu/~jos/filters/