DIGITAL AUDIO EFFECTS APPLIED DIRECTLY ON A DSD BITSTREAM

Josh Reiss and Mark Sandler

Centre for Digital Music
Queen Mary, University of London
Mile End Road, London E14NS, U.K.
{josh.reiss|mark.sandler}@elec.qmul.ac.uk

ABSTRACT

Digital audio effects are typically implemented on 16 or 24 bit signals sampled at 44.1 kHz. Yet high quality audio is often encoded in a one-bit, highly oversampled format, such as DSD. Processing of a bitstream, and the application of audio effects on a bitstream, requires special care and modification of existing methods. However, it has strong advantages due to the high quality phase information and the elimination of multiple decimators and interpolators in the recording and playback process. We present several methods by which audio effects can be applied directly on a bitstream. We also discuss the modifications that need to be made to existing methods for them to be properly applied to DSD audio. Methods are presented through the use of block diagrams, and results are reported.

Keywords: Sigma Delta Modulation, SACD, DSD, Digital Audio Effects, Bitstream Signal Processing

1. INTRODUCTION

One-bit signals are used throughout the audio recording, editing and playback process. Most analog to digital and digital to analog converters employ a sigma delta modulator that converts a signal to a bitstream. Digital audio is often stored during production in a single bit format. In addition, the high-end audio distribution format, SuperAudio CD, employs the single bit recording format known as Direct Stream Digital, or DSD.

The benefits of the DSD format are numerous. Improvements in the traditional pulse code modulation (PCM) format from higher bit rates and higher sampling rates have experienced diminishing returns. This is partly due to the difficulties in implementing accurate high bit quantisers, but primarily due to the losses incurred from filtering. PCM systems require steep filters at the input to block any signal at or above half the sampling frequency. Ideally, a brick wall filter should be used; passing all frequencies below the Nyquist frequency, and rejecting all above. Yet an ideal brick wall filter does not exist.

In addition, requantization noise is added by the multi-stage or cascaded decimation (downsampling) digital filters used in recording and the multi-stage interpolation (oversampling) digital filters used in playback. Increasing the sample rate, as with DVD-Audio, eases the difficulty of the brick wall filter, but does not correct the problems introduced by multi-stage decimation and interpolation.

This was the inspiration for a 1 bit audio format, as first proposed by Angus [1], and independently implemented as Direct Stream Digital (see Figure 1). As in conventional PCM systems, the analog signal is first converted to digital by 64x oversampling sigma delta modulation. The result is a 1-bit digital representation of the audio signal. Where conventional systems immediately decimate the 1-bit signal into a multibit PCM code, Direct Stream Digital records the 1-bit pulses directly.

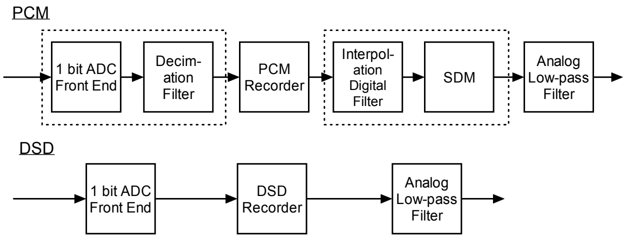

Figure 1: *The standard multibit PCM recording amd playback chain, (a), requires a decimation filter on the recording side and an oversampling filter on the playback side, whereas Direct Stream Digital, (b), enables sound to be recorded directly in the 1-bit signal format and eliminates the need for these filters.*

The resulting pulse train has some remarkable properties. The bandwidth now extends over more than 1.4 MHz. Through the use of high order sigma delta modulators (SDMs), the noise can be shifted up to inaudible frequencies. And the digital-to-analog conversion is now as simple as running the pulse train through an analog low-pass filter.

Ultra-high signal-to-noise ratios as required for DSD in the audio band are achieved through 5th-order noise shaping filters. Thus DSD can represent signals with a frequency response from DC to 100 kHz. The residual noise power is held at -120 dB through the audio band [2].

Although single bit, oversampled formats have been found to be excellent for archiving, A/D and D/A conversion, and recording [3], they suffer from a serious drawback in the editing and mastering phase. Few tools have been developed which allow effective processing of audio bitstreams. To apply audio effects directly on the bitstream, it is vital that requantisation, decimation and interpolation be kept to a minimum.

However, processing and audio effect creation in the 1 bit domain is appealing for many reasons. The oversampled signal has very high quality phase information, making phase vocoder-based effects easier and more accurate. Effects using variable delays, such as chorus and flange, also benefit from oversampling since interpolation of the delay is far more precise. Furthermore, 1-bit audio effects can be applied on the DSD signal directly before or after

encoding, thus maintaining the simplified production chain as in Figure 1.
The goal of this paper is to describe how to develop standard audio effects on the DSD bitstream, while minimizing intermediate conversions to multibit format (thus destroying all benefits of DSD). Previous work [4-12] has already established that suitable IIR and FIR filters can be created, as well as some mixing tools. However, common audio effects, such as compandors, expandors, reverb, modulation, and so on, have not yet been developed. In the following Sections we will demonstrate how these effects can be applied directly on a bitstream without introducing unwanted artifacts, or significant degradation of audio quality.

2. PROPERTIES OF THE DSD BITSTREAM

There are several features of DSD which distinguish it from PCM. At its heart, DSD is specified as being a 1-bit format, with a sampling rate of 64*44.1 kHz, or 2.8224 MHz [13]. Little else is specified regarding the format, although constraints are imposed for the archiving of DSD on SuperAudioCDs and the playback of those CDs (notably, restrictions on noise levels, frequency response, peak levels and DC offsets). However, the specifications of DSD also note the following properties

1. The 1-bit format is such that the 1 represents a positive output (+1) and the 0 a negative output (-1).

2. The 0 dB reference level has been set to 50% of the maximum theoretically possible modulation depth. At least 4 out of any 28 consecutive bits must be set to 1 (and similarly for 0). This maximum setting corresponds to 3.10 dB.

3. Silence patterns are defined as repeating bytes where each byte contains an equal number of 1s and 0s.

Unlike PCM, the DSD signal always has a power of 1 (the bits representing +1 and -1 levels). Thus any instantaneous measurement of signal level is meaningless. Furthermore, whereas PCM has a strict 0dB maximum, the 0 dB limit for DSD has been imposed as a safety measure. In practice, this means that a DSD signal, when put through a sigma delta modulator, is unlikely to result in instability or severe clipping since its peak levels have already been restricted to within safe margins.
Silence patterns do not make sense in 44.1 kHz PCM since any repeating pattern would be ≤ 22.05 kHz and hence potentially audible. A constant DC level represents silence in PCM. But for a DSD signal, constant levels (i.e., all zeroes or all ones) are not allowed. A repeating pattern of 8 bits or less, on the other hand, only has frequency components above 176 kHz, i.e., far outside the range of human hearing. Thus whenever inaudible output is required, a silence pattern should be used. This is important in the construction of many audio effects, such as noise-gating.

3. TIME-DOMAIN AUDIO EFFECTS

Most time-domain based audio effects have well-established implementations [14]. The general design of these effects, when implemented on a DSD signal, can follow the design used for PCM signals. In this Section we describe those design modifications which are necessary for DSD.

3.1. Bitstream addition

Perhaps the most fundamental signal processing is the addition of two signals. O'Leary and Maloberti [15] demonstrated an elegant bitstream adder (Figure 2). The oversampled nature of the bitstream allows one to use a simple feedback loop whereby two bitstreams are added along with the sum bit from the previous iteration. When the bandwidth of the input signals is far below the sampling frequency, as is the case with DSD, the output carry bits are an excellent representation of the average of the two signals.
This bitstream adder is remarkable because it requires no requantisation, and it has been shown to be highly effective for oversampled signals. The alternative, bitstream addition via the interleaving of bitstreams [16], suffers degradation of audio quality due to downsampling, phase shift and possible introduction of low-frequency noise.
However, although this bitstream adder does not explicitly perform requantisation, it amounts to the same effect. Thus it acts as a first order sigma delta modulator and introduces some noise and distortion into the audible band. The bitstream adder is suitable either for a limited duration, or when increased noise is acceptable. An alternative would involve summing the signals and then performing high order noise shaping.

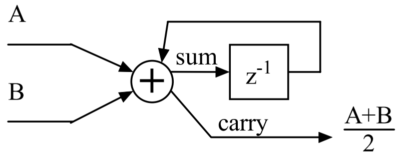

Figure 2: *A bitstream adder.*

3.2. Delay based effects

By using the bitstream adder in conjunction with multiple delays, it is possible to create a flanger or chorus effect entirely through simple logic operations on the bitstream. This is indicated in Figure 3, where BSA represents the bitstream adder from Figure 2. This implementation is very elegant and appealing because it requires no filtering, decimation, interpolation or requantisation. It deals solely with bit operations and delays. Furthermore, the delays can be set to any length, and due to the high sampling rate of DSD, there are far more options over the number of voices and their placement. To weight the delayed signals, a given delay time may be repeated in the inputs to the bitstream adders.

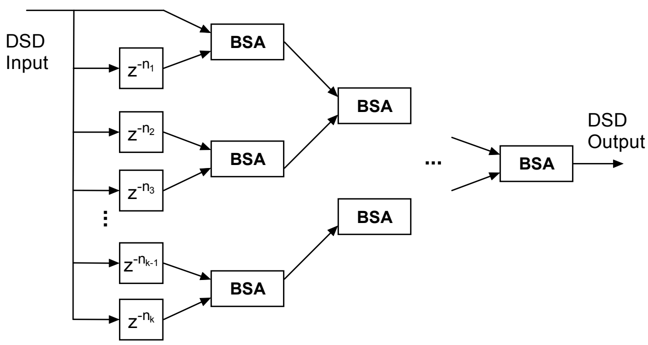

Figure 3: *Implementation of a basic flanger or chorus using the bitstream adder (BSA) of Figure 2.*

However, it suffers serious limitations in that it allows for no mixing of signals other than additively. Furthermore, the number of signals mixed in this way must be a power of 2. Successive use of the bitstream adder in parallel and series may mimic the effect of a multiplier, but significant noise might then accumulate in the audio band, and it still does not allow for easy implementation of a gain control. A bit stream multiplier is essential for volume adjustment, or for versatile mixing of signals. Therefore, most effects will be implemented using conversion to a multibit domain, and then a sigma delta modulator in the final stage is used for requantisation to DSD. As shown in Section 4, this SDM can sometimes be incorporated into the effect processing stage.

3.3. Level detector

In order to implement many effects, such as noise gating, expansion, limiting and compression, a level detector is required. In PCM, this is trivial, since the instantaneous level is given by the quantised signal at any given time. For a bitstream, however, the instantaneous value is either 0 or 1, corresponding to a 1 or -1, for input over the range [-Max, Max] where the maximum absolute value of the input is some value Max < 1.
Nevertheless, PCM level detection usually employs a time average d power of the signal and bitstream level detection can do the same. It is important however, that the time average be over roughly the same amount of time but not over the same amount of samples. The high oversampling rate demands this.
Time average level detection becomes even simpler for DSD signals. RMS estimation of power is unnecessary. One can simply count the bits. Over a window of size N, where M is the number of ones in the window, $P=|N-2M|/N$ gives an estimate of the power. A value between 0 and 1 for P can set the threshold. For most dynamic processing, standard techniques can then be applied. A variable gain can multiply the signal, with the additional requirement that the output is processed through a sigma delta modulator (and optionally, a low pass filter), to return the signal to DSD format.
For an accurate envelope detector, a simple moving average filter should not be used. A decimation filter is preferred since it more accurately represents the multibit level of the signal at any instance. It is important to note that under such a situation, decimation need only be used for level detection, and no additional decimation/interpolation is applied to the bitstream.

3.4. Modulators

Modulation involves the multiplication of an audio signal by some carrier signal, typically a sinusoid. To do this using entirely DSD signals would involve the multiplication of two bitstreams. Unfortunately, this is not as simple as the addition of bitstreams as in Figure 2. The product of two single-bit signals can be obtained with just one logical gate, an XNOR (or an AND if the signals were restricted to [0;Max]). However, this approach affects the noise-shaping characteristics. Multiplication in time domain corresponds to convolution in z-domain. Therefore, the resulting bitstream has four components: one from the convolution of the two signals, two from convolutions between one signal and the shaped noise of the other bitstream, and the last from the convolution of the two shaped noises. Since the last term has a flat frequency spectrum, the result of a multiplication of two noise-shaped bitstreams is a non noise-shaped waveform, whose in-band noise limits the accuracy of processing.

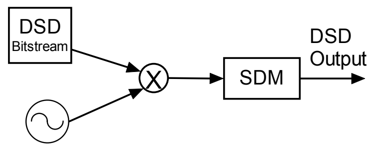

Figure 4: *Modulation of a DSD bitstream.*

Currently, the only alternative is to perform multiplication of DSD signals via decimation to a multibit domain, and then reconverting to DSD via upsampling and requantisation. This suffers severe drawbacks because of the introduction of low frequency noise.
However, since, the carrier signal is intended to be an internally generated waveform, it need not be in DSD format. This allows for mixed domain processing. The carrier signal can be generated multibit, at the DSD sampling rate. The DSD bitstream can then be multiplied by this multibit signal, and converted back to single bit output. Filtering of the output should be kept minimal since the purpose of most modulators, such as ring modulation, is to introduce new frequencies. This system is depicted in Figure 4.

3.5. Noise gating

An extreme noise gate operates simply as a threshold below which there should be no signal. A noise gate operating on a DSD signal has several important distinguishing characteristics which require modifications of the standard PCM noise gate in order to function. First, the level detector or envelope follower requires modification, as mentioned in Section 3.3.
Noise gating however, requires further modification. When the signal has been faded to zero, the output must correspond to DSD silence. It is conceivably possible that traditional techniques will produce a signal that, although representing the output of an SDM acting on zero input, will not be silent [17, 18]. This could occur due to small DC offsets or initial conditions of the SDM. This problem is especially serious because, rather than this signal being a very high frequency pattern, as DSD silence is defined, it may be a very low frequency pattern and hence audible.
For these reasons, when silence is required at the output, as may be the case in a noise gate, the output bitstream is replaced with a DSD silence pattern. If smooth transitioning between silence and low-level signal is required, then one of the switching techniques described in Section 3.6 can be applied during the fade-in and fade-out stages.

3.6. Smooth mixing and switching of bitstreams

It is well-known that switching of PCM signals can result in audible artefacts due to discontinuities in the output signal. This is avoided by strictly requiring that the PCM samples from the initial and replacement streams be identical at the point at which the switch is made. Samples around the switch should also be roughly identical to prevent abrupt changes in signal slope (and instantaneous frequency) as well.
But the DSD signal contains historical information. That is, the current signal is determined by a sequence of bits, and the next bit is a function of prior states as well as current input. Thus, sample matching is not sufficient. Smooth switching requires that the switch happen when the two bitstreams are synchronised.

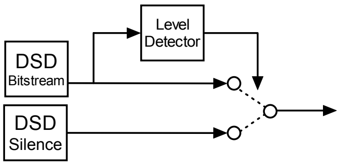

Figure 5: *A hard noise gate implemented on a DSD bitstream. A DSD silence signal must be used since constant DC levels are not possible.*

In [19], Reefman and Nuitjen described an approach to synchronisation of bitstreams which allows for seamless switching. This approach involves the use of a sigma delta modulator acting on the mix of the two input bitstreams. However, this SDM must be synchronised such that it produces the bitstream A when acting just on A, and the bitstream B when acting just on B.

In order to produce synchronisation, the integrator states, or initial conditions of the SDM, must match those integrator states. This synchroniser can be implemented by using a least squares approach to find integrator states which minimise the difference between a DSD input signal and the resulting DSD output signal. Thus editing is done as depicted in Figure 6. When synchronisation is ready, the switch is changed to the central position, and G is set to 1. G is slowly decreased to 0, then the output stream is resynchronised to input stream B, and the switch is set to the downwards position.

An alternative switching method is proposed in Figure 7. We note first that both input and output streams are low-pass filtered, and the application of a slowly changing gain and a first order SDM should not significantly change the bandwidth of the signal. Importantly, a first order SDM will have no effect on a DSD bitstream. The difference between quantization of a bit and the original bit is zero. Thus, when G is set to 1 in Figure 7, the output bitstream is A. As G is decreased, a cumulative error based on the difference between the 2 input signals is added to the quantiser input. As G approaches 0, the difference between the output and input bitstream B also approaches 0. Eventually, the feedback term approaches a constant (typically non-zero) and the output bitstream is identical to B. The only significant introduction of noise is the non-shaped noise due to the first order SDM acting on the sum of two bitstreams when the gain is in the region $0<<G<<1$. However, this occurs over a relatively short period and is minimized since both inputs are already low-pass filtered.

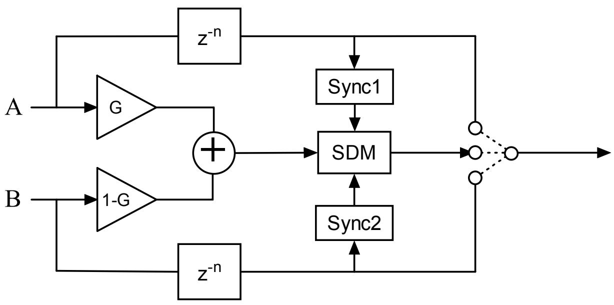

Figure 6: *Smooth switching between bitstreams using synchronisation.*

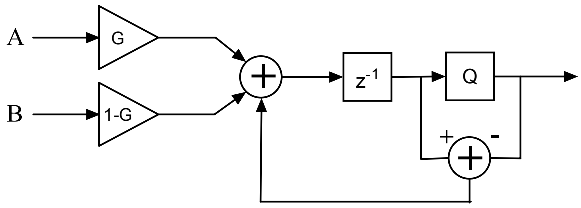

Figure 7: *Smooth switching between DSD bitstreams using a slowly changing gain and a first order SDM.*

The result of this switching scheme on input signals of frequency 1 and 2 kHz, is depicted in Figure 8. A switch is desired at 2 milliseconds. The example is particularly pernicious (and somewhat unrealistic) since the waveforms are very different; out-of-phase and with peak amplitudes of 0.2 and 0.9. The gain is changed linearly from 1 to 0 over 1,600 samples, or just over half a millisecond. Depicted are the analog input signals before conversion to bitstreams, and the output signal after decimation to multibit, 44.1 kHz using a sinc^2 filter. The resulting transition at 2 msecs is smooth without abrupt changes in amplitude or slope. There is a slight and temporary increase in frequency, but this effect can be minimised through the use of a slower gain change or eliminated completely by using a detection scheme to find a more appropriate time to perform the edit.

Improvements to this method could also be achieved by using a more effective noise shaper (higher order SDM) instead of the first order SDM in Figure 7. However, with gain equal to 1, the output bitstream would not be identical to the input bitstream. To phase out the effects of requantisation, and resynchronize the output bitstream with the input stream A, we can slowly reduce the feedback coefficients of the modulator. As feedback coefficients approach zero, the modulator becomes lower order until it approaches a first order SDM, and as before, has no effect on the bitstream.

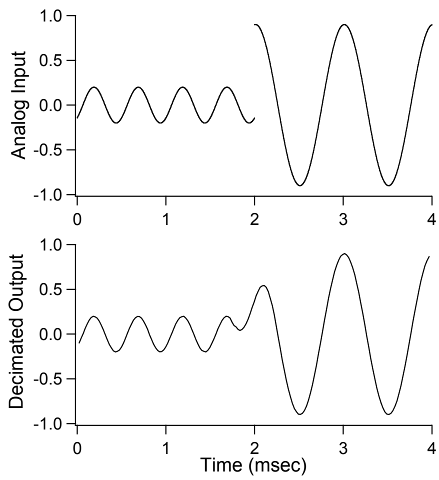

Figure 8: *Smooth switching between DSD bitstreams using the circuit from Figure 7. This is the worst case scenario, where the input bitstreams have differing amplitudes and opposing phases.*

4. FREQUENCY-DOMAIN EFFECTS

Virtually all frequency-domain based audio effects, such as equalisers, wah-wah, and phasers, require the construction of FIR or IIR filters. A significant body of research exists on 1-bit filters. A full discussion of 1-bit filter designs is beyond the scope of this work. Here, we note the main research and how 1-bit designs differ from their PCM-based equivalents.

Angus [4] provided a means of implementing FIR and IIR filters on the DSD bitstream. This was based partly on prior work on FIR filters by Wong and Gray [5, 6] and Kershaw, et. al. [7] and IIR filters by Johns and Lewis [8, 9], and on his own work concerning the processing of one bit digital audio signals [10].

Equalisation is usually implemented by shelving filter design using first order filters. In [4], Angus demonstrated a bass cut/boost control filter which acts directly on the DSD bitstream. He reported roughly equivalent performance to PCM equalizers.

4.1. FIR Filters

Filters for DSD input and output signals have several design considerations which distinguish them from their PCM equivalents. The main alterations are not the same for IIR filters and FIR filters [11]. For a one-bit FIR filter acting on a 64 times oversampled DSD signal, the delay line consists of z^{-64} delays rather than z^{-1} delays. In effect, the taps are subsampled. This has the effect of zero-interleaving the impulse response by a factor of 64. The frequency response is replicated throughout the entire frequency range. This would thus demand a high order filter, except for the fact that this replicated response is outside the audible range. In general, the out-of-band frequency response is irrelevant. Whether the signal needs additional filtering is then dependent on the use of the filter and on the requirements for the high frequency content of the signal. Alternatively, one could redesign the filter using single delays and take into account the high sample rate and single bit input. This approach involves a combination of cascaded integrators and a sparse tap filter [4]. It is efficient, removes the high frequency noise and can achieve the desired frequency response.

4.2. IIR Filters

IIR filtering of a DSD signal, on the other hand, does not change the delays but changes the coefficients. The coefficients of the filter can be calculated in the same way as for PCM, but the oversampling implies that their values will be very different.

As has been mentioned, requantisations should be kept to a minimum. Thus, if the filtering consists of IIR/FIR filters, a noise shaping filter and a low pass filter, then these stages should be combined in such a way that there is only one requantisation in the final stage. Figure 9 depicts an IIR filter which incorporates an SDM-based requantiser. Although such a design is efficient and eliminates the multi-bit stage, it does not differ greatly from a cascade of one bit filters followed by a remodulator.

Minimising decimation, interpolation, and requantisation is not a drawback. These filters add to system complexity and degrade performance. In addition, filtering in the oversampled domain is advantageous because it relaxes specifications on anti-alias and reconstruction filters at the analog interfaces, thus improving phase linearity [12].

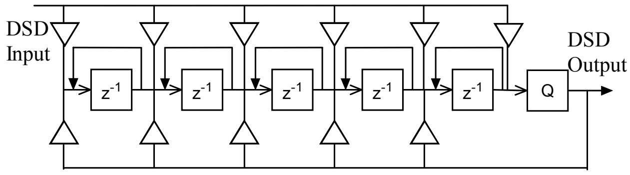

Figure 9: *Configuration of a combined IIR filter and remodulator.*

5. CONCLUSIONS

This work concerned how to apply audio effects directly on a DSD bitstream. The general architecture of many effects is approximately the same. However, major modifications need to be made to level detection, noise gating, and switching methods. Conversions to the multibit domain, quantisations and filtering should be minimized. Thus, wherever possible, processing stages should be combined and a single requantisation step should be placed at the end.

One subject which has not yet been adequately investigated is an empirical comparison of audio effects implemented on PCM and DSD signals. All the effects methods discussed within were analysed via the use of simple SDMs for requantisation and a decimation filter allowing comparison with PCM effects. However, this introduces further noise and thus direct comparison is not easy. Development of sophisticated decimation filters and implementation of high order SDMs would allow for a more rigorous analysis. Also, proper analysis of audio effects on DSD signals requires listening tests comparing the signal before and after the effect is applied. However, DSD signals are hard to come by. A new audio format, DSDIFF, has been proposed for the exchange and storage of DSD-encoded audio [20]. As the format gains acceptance, DSD sample files will become available and direct comparison of audio effects on DSD and PCM signals will become possible.

6. ACKNOWLEDGMENTS

The authors gratefully acknowledge the contribution of Prof. James Angus for his comments and criticism concerning this work.

7. REFERENCES

[1] J. A. S. Angus, "The One Bit Alternative for Audio Processing and Mastering," *Proceedings of the Audio Engineering Society Conference on Managing the Bit Budget*, London, pp. 34–40, 1994.

[2] D. Reefman and E. Janssen, "Signal processing for Direct Stream Digital: A tutorial for digital Sigma Delta modulation and 1-bit digital audio processing," *Philips Research*, Eindhoven, White Paper 18 December 2002

[3] D. Reefman and P. Nuijten, "Why Direct Stream Digital (DSD) is the best choice as a digital audio format," *Proceedings of the Audio Engineering Society 110th Convention,* Amsterdam, Holland, 2001.

[4] J. A. S. Angus, "Direct Digital Processing of 'Super Audio CD' Signals," *Proceedings of the Audio Engineering Society 108th Convention,* Paris, France, 2000.

[5] P. W. Wong and R. M. Gray, "FIR filters with sigma-delta modulation encoding," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 38, pp. 979–990, 1990.

[6] P. W. Wong, "Fully sigma-delta modulation encoded FIR filter," *IEEE Transactions on Signal Processing*, vol. 40, pp. 1605–1610, 1992.

[7] S. Kershaw, *et al.*, "Realization and Implementation of Sigma Delta Bitstream FIR Filter," *IEE Proc. Circuits Devices Syst.*, vol. 143, pp. 267–273, 1996.

[8] D. A. Johns and D. M. Lewis, "Design and analysis of delta-sigma based IIR filters," *IEEE Trans. on Circuits and Systems - II: Analog and Digital Signal Processing*, vol. 40, pp. 233–240, 1993.

[9] D. A. Johns and D. M. Lewis, "IIR filtering on sigma-delta modulated signals," *Electronic Letters*, vol. 27, pp. 307–308, 1991.

[10] N. M. Casey and J. A. S. Angus, "One Bit Digital Processing of Audio Signals," *Proceedings of the Audio Engineering Society 95th Convention,* New York, 1993.

[11] S. Kershaw, "Sigma delta bitstream processors - analysis and design," *Electronic Engineering*. London: Kings College, 1996, pp. 325.

[12] J. A. S. Angus and S. Draper, "An Improved Method For Directly Filtering S-D Audio Signals," *Proceedings of the Audio Engineering Society 104th Convention,* Amsterdam, 1998.

[13] Sony and Philips, "Super Audio CD Audio Signal Properties (SACD Scarlet Book)," March 2003. `www.superaudiocd.philips.com/Assets/Downloadablefile/SACD_Au_DE_130(scarlet-book)-2325.pdf`

[14] *Dafx:Digital Audio Effects*: Wiley, John & Sons, 2002.

[15] P. O'Leary and F. Maloberti, "Bit stream adder for oversampling coded data," *Electronic Letters*, pp. 1708–1709, 1990.

[16] A. K. Lu and G. W. Roberts, "An analog multi-tone signal generator for built-in self-test applications," *Proceedings of the IEEE International Test Conference,* Washington, pp. 650–659, 1994.

[17] D. Reefman, *et al.*, "Stability Analysis of Limit Cycles in High Order Sigma Delta Modulators," *Proceedings of the Audio Engineering Society 115th Convention,* New York, New York, 2003.

[18] J. D. Reiss and M. B. Sandler, "They Exist: Limit Cycles in High Order Sigma Delta Modulators," *Proceedings of the 114th Convention of the Audio Engineering Society,* Amsterdam, The Netherlands, 2003.

[19] D. Reefman and P. Nuijten, "Editing and switching in 1-bit audio streams," *Proceedings of the 110th Convention of the AES,* Amsterdam, Netherlands, 2001.

[20] Philips, "Specification DSD Interchange File Format," Version 1.4 ed., 2003. `www.superaudiocd.philips.com/InformationCenter/NO/FArticleSummary.asp?lNodeId=3404`