

A MECHANISM FOR THE DETECTION AND REMOVAL OF LIMIT CYCLES IN THE OPERATION OF SIGMA DELTA MODULATORS

J.D. Reiss*, M.B. Sandler*

*Dept. of Electronic Engineering, Queen Mary, University of London, U.K.
Mile End Road, London, U.K. E14NS
josh.reiss,mark.sandler@elec.qmul.ac.uk
Fax +44 207-882-7997

Keywords: Sigma delta modulation, limit cycles.

Abstract

Sigma delta modulators (SDMs) may suffer from limit cycles, where the output bits may enter a repeating pattern. Current methods of preventing this phenomenon introduce unwanted noise, do not always succeed, and are implemented when not needed. We present a more effective method for detecting and removing unwanted limit cycles. This destroys the periodicity of sigma-delta ADC modulator's output sequence and, thereby, removes the limit cycles. Compared with conventional methods, the SDM has less SNR penalty and the mechanism is simple to implement. Moreover, the SDM has a higher allowed input dynamic range than conventional modulator dithering schemes. Analog and digital implementations of the limit cycle detection and removal schemes are discussed for both feedforward and feedback designs. Results are reported which demonstrate the success of these methods.

1 Introduction

Sigma delta modulation is a popular method of converting signals from analogue to digital and vice-versa. It typically involves converting a signal into a low-bit, highly oversampled representation. It benefits greatly from the oversampling in that a feedback path may be used to shape the quantization noise into high frequencies where it is not noticeable. Due to its low circuit complexity and robustness against circuit imperfections, 1-bit sigma delta-based analog-to-digital and digital-to-analog converters are widely used in audio applications, such as cellular phone technology and high-end stereo systems.

Sigma delta modulation, originally conceived by de Jager[1], is a well-established technique. However, theoretical understanding of the concept is limited[2]. The most important progress in the description of sigma delta modulators is reported in the work of Risbo[3] and Hein[4], while a useful linearization technique is described in [5] and further elaborated on by Magrath[6]. Yet in all these developments, there is no unified description of SDMs.

Instead, several models are provided, each of which describes some aspects of an SDM to a certain accuracy.

Fundamental work on limit cycles in SDMs has usually been constrained to low order SDMs[7-9], and hence is of little practical value to engineers who use high order noise shaping techniques. Recent work by the authors and their collaborators has significantly advanced the theory of limit cycles in sigma delta modulators[10-14]. Most notably, in [12], results were derived concerning the character of limit cycles for a general feedforward SDM (also called interpolative SDM), and on their stability in particular. In [11], similar results were obtained for feedback sigma delta modulators.

Limit cycle prevention is typically achieved by adding a signal, with a uniform or triangular probability distribution, just prior to quantisation[15]. When this dithering sequence is added, an output bit may be flipped (output bit changed from +1 to -1, or from -1 to +1), and the periodic output pattern might be destroyed. However, the dither decreases the signal-to-noise ratio, the stability, and the dynamic range of the sigma delta modulator. Furthermore, it is added when it is not needed, and in many situations may not be sufficient to destroy a limit cycle.

The work of the authors has lead to a greater understanding of the cause and the behavior of limit cycles. We have exploited this understanding to devise a more effective method of detecting and removing limit cycles.

The paper is organized as follows. In Sec. 2, the mathematical framework, based on a state space description of the SDM, is presented. All the following sections are based on this formulation. In Sec. 3, a method is presented to detect limit cycles in feedforward SDMs. An alternative method, using only shift registers, is described in Sec. 4. Properties of the limit cycle removal mechanism are discussed in Section 5. In Sec. 6, we discuss the differences that must be taken into account for limit cycle detection and removal in feedback SDMs. Throughout, these methods are analysed to give quantitative results concerning the probability of false limit cycle detection, the time required for limit cycle removal, and the choice of parameters.

2 State space description

A convenient way to describe the time domain behaviour of an SDM is the state space description. This represents the state of the SDM at any time as a matrix operation applied to the state at the previous clock cycle. The power of the state space description is that it allows us to create a very compact description of the state of the SDM from time $t=0$ to time $t=n$. For an N^{th} order feedforward (or iterative) SDM,

$$\mathbf{s}(n+1) = \mathbf{A} \mathbf{s}(n) + (u(n) - y(n)) \mathbf{d} \quad (1)$$

where

$$y(n) = \text{sgn}\left(\sum_{i=1}^N c_i s_i(n)\right) \quad (2)$$

This description gives the state of the SDM in terms of a transition matrix \mathbf{A} applied to the previous state vector, and a vector \mathbf{d} applied to the scalar quantisation error, $u(n)-y(n)$. Figure 1 gives an example of a typical fifth order sigma delta modulator. The coefficients \mathbf{c} determine the noise shaping characteristics, each T represents a unitary delay, and the loop around each delay T represents an integrator.

For the 5th order modulator described in Figure 1, $\mathbf{d}=(1,0,0,0,0)^T$ and the transition matrix is

$$\mathbf{A} = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 \end{pmatrix} \quad (3)$$

The compact representation gives the means to directly view the consequences of a limit cycle. If the limit cycle has period P we have, by definition,

$$y(n+P) = y(n) \quad (4)$$

An important assumption in earlier work is that a periodic bit output pattern implies a periodic orbit in state space variables. In [12], it was proven that, in general, a limit cycle in the output bitstream exists if and only if there is a limit cycle in the state space variables. That is, (4) is equivalent to

$$\mathbf{s}(n+P) = \mathbf{s}(n) \quad (5)$$

Although in its pure definition, a limit cycle is a periodic pattern of infinite duration, in practical situations finite duration periodic sequences can be equally annoying. Thus a limit cycle detection and removal algorithm should be successful even if Eq. (5) is only true for a finite number of values, i.e., we must be able to detect and remove limit cycles when Eq. (4) is approximately true for a finite time.

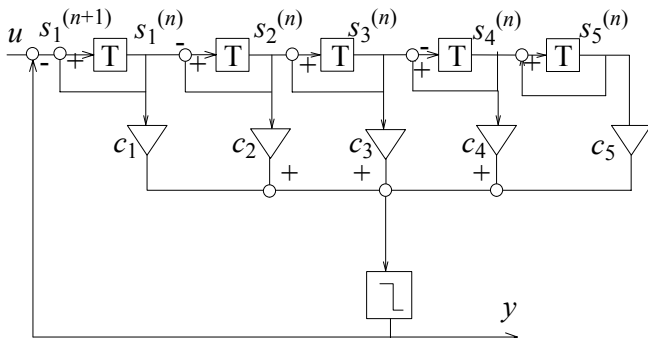


Figure 1. States in a 5th order SDM.

3 State space-based limit cycle detection

Eq. (5) provides a simple method of determining if a limit cycle exists. At a given iterate which we set to 0, $\mathbf{s}(0)$ may be stored in a buffer. For each successive iterate, $1, 2, \dots, i, \dots$, up to some value P_{max} , $\mathbf{s}(i)$ is computed. If constant input is applied and, for some i , $\mathbf{s}(i) = \mathbf{s}(0)$, then the theorem described in the previous section guarantees that a limit cycle of period P exists.

This method, while exact, has several drawbacks. It requires that a vector of size N be stored. At each time iterate, up to N comparisons must be made. This is unnecessarily complicated. More importantly, it does not allow for a simple method of making approximate comparisons. When the state space variables are very close to a limit cycle condition, periodic output may be sustained long enough to be problematic. An appropriate measure of the required proximity of the state space variables needed for temporary limit cycle behavior is not obvious and may not be simple to compute.

We have devised a preferable alternative method which relies on calculation of a single scalar quantity[16]. This quantity is easily found from the operation of the SDM. Its key features are outlined below.

- It allows us to find all limit cycles that occur up to a given period.
- It allows us to find short term limit cycles which repeat for only a small number of periods.
- It is robust to the choice of parameter settings.
- It may detect limit cycles which occur at any time during the operation of the SDM
- It is independent of the order of the SDM and its noise shaping characteristics
- It is independent of the input to the SDM and its initial conditions
- The mechanism of limit cycle detection operates at the speed of the sigma delta modulator
- It may be used in tandem with any limit cycle removal method.

As an example, consider the fifth order sigma delta modulator given by[12],

$$c_1=0.5761069262, c_2=0.1624753515, c_3=0.0276093301, \\ c_4=0.0028053934, c_5=0.0001360361$$

With an input of 0.7, and initial conditions $\mathbf{s}=0$, it exhibits limit cycle behavior. Figure 2 depicts a time series of the input to the quantiser. The circled points represent those where limit cycle behaviour has been identified. Clearly, limit cycle behaviour has been correctly identified. The inability to recognise a limit cycle during its initial periods is due to the choice of parameters in the detection method and due to the fact that a limit cycle may be defined by its repetitive nature, which is not observed until after several repetitions.

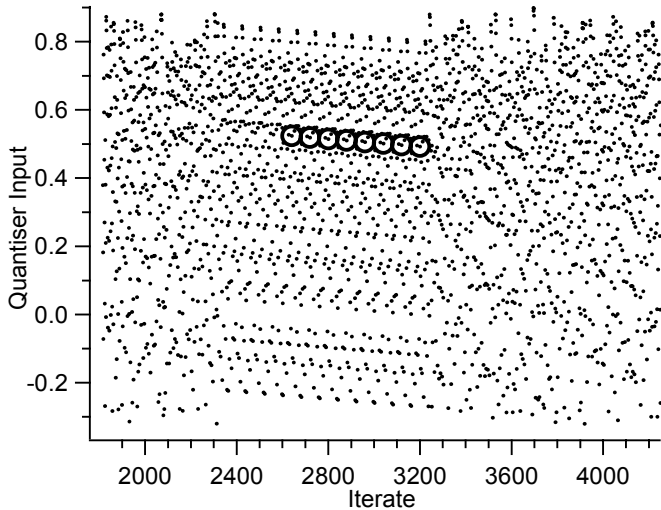


Figure 2. Plot of the input to the quantiser as a function of the iterate. The circled points represent where limit cycle behaviour has been identified.

4 Limit cycle detection at the SDM output

Shift registers and bit comparisons are often much easier to implement than any circuitry (analog or digital) for comparison of real numbers or voltages. Furthermore, one can conceive of situations where it is easier to access the output bitstream than the state space variables. Thus, we must consider methods of detecting limit cycles using bit comparisons alone.

A naïve approach would be to implement many shift registers, each one representing periodic output for a different limit cycle. The current output could then be compared with each shift register to see if it appears that limit cycle behavior is occurring. However, this would require on the order of 2^p shift registers, and 2^p comparisons.

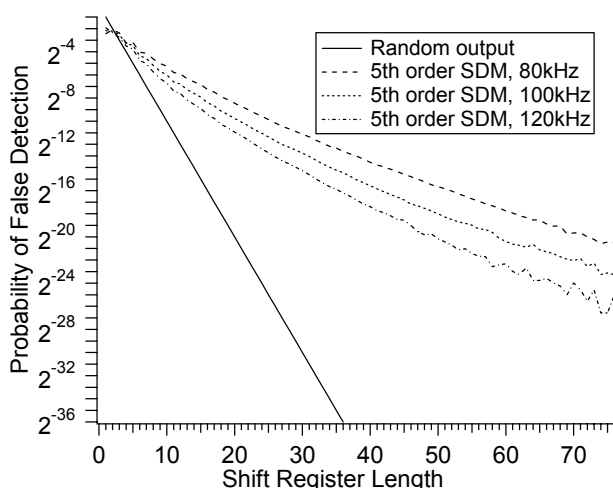


Figure 3. The probability of falsely detecting a limit cycle as a function of shift register length.

We have devised a method, using only two shift registers, which will find all possible limit cycles up to a given period (which may be larger than the shift register length). It has all the advantages of the state space method which were described in Section 3. Furthermore, it is robust against any additional parameters (such as shift register length) and easy to implement.

False detections are very rare. If the output is truly random, then false detections occur with a probability 2^{-Q} , where Q is the length of the shift register. However, the output is far from random. This is partly due to the fact that the input is not random (bandlimited, with amplitude safely within stability limits), but also because the sigma delta modulation prevents certain sequences from occurring, regardless of input[10].

Figure 3 depicts false detections of limit cycle behaviour as a function of the shift register length Q . Each data point was generated using 100, 1 million point long sequences (after initial startup transients were removed), where each sequence has as input a sinusoid with a randomly generated frequency between 80kHz and 130kHz, and set to random initial conditions. It can be clearly seen that, though the probability of a false detection is far greater than would be the case for a truly random sequence, it is still low enough to be insignificant.

5 Limit cycle removal

In [12], it was shown that, the application of dither just before the quantiser, as in Figure 4, is a sub-optimal form of limit cycle removal. This is because it has no effect on the state space variables unless it results in a change to the output bitstream. We have devised an alternative method with the following advantages[16]:

- It may be applied continuously, or only when a limit cycle has been detected.
- If used in tandem with a limit cycle detection mechanism, it is independent of the choice of detection method.
- It is guaranteed to remove any limit cycle.
- It is independent of all SDM characteristics.
- Its effect on the SDM (other than removing limit cycles) is minimal.
- It is robust to the choice of parameter settings.

Since this modification is both minimal and guaranteed to work, it is preferable to the commonly used alternative of adding dither or noise to the input to the quantiser. However, dithering may still be used if so desired (for instance, to minimise noise modulation).

Compared to conventional dithering techniques that add random noise to the input of the quantizer, this novel technique has a higher allowed input dynamic range and higher signal-to-noise-plus-distortion-ratio (SNDR). It may also be implemented successfully without the use of a limit cycle detector.

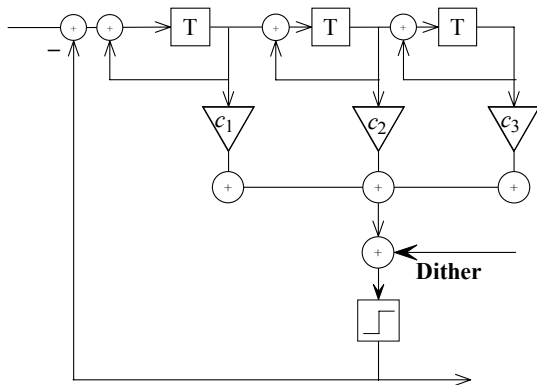


Figure 4. A block diagram of a third order SDM with the standard placement of dither.

One implementation of the limit cycle removal mechanism involves the addition of a small amount of noise, placed such that it is guaranteed to remove any limit cycle. Figure 5 demonstrates how the amount of added noise affects the time it takes to destroy a limit cycle. This is a worst case scenario. Not only have the initial conditions of the modulator been chosen to guarantee that the dynamics fall exactly on a limit cycle, but they have also been chosen so that the most stable possible period 12 limit cycle is produced, and that the initial conditions are as far as possible from those that would produce a bit flip and thus destroy the limit cycle. Nevertheless, even noise on the order of 10^{-6} (~ 120 dB) is sufficient to eliminate the limit cycle long before it becomes problematic. Again, this is a worst case scenario. Typically, noise on the order of -140 dB can be applied, which is suitable for even high end audio applications.

6 Limit cycle detection in feedback SDMs

A popular alternative design to the feedforward, or interpolative, SDM, is the feedback SDM. This is often used when a superior anti-aliasing effect of the signal transfer function is required[2, 11]. In Figure 6, the block diagram of a 3rd order feedback SDM is depicted. This represents a typical SDM design, which is often used in practical designs[17]. We can easily see that, for the modulator presented here,

$$\begin{aligned} \mathbf{s}^{(n+1)} &= \mathbf{A} \mathbf{s}^{(n)} + u^{(n)} \mathbf{d} - y^{(n)} \mathbf{c} \\ y^{(n)} &= \text{sgn}(s_N^{(n)}) \end{aligned} \quad (6)$$

where $y(n)$ is the output bit at clock cycle n , and $s_i(n)$ are the integrator outputs, called state variables. The last integrator output, $s_N(n)$, is also the quantizer input signal.

The propagation of the states \mathbf{s} can be written in matrix notation as:

$$\mathbf{s}^{(n)} = \mathbf{A}^n \mathbf{s}^{(0)} + \left[\sum_{i=0}^{n-1} \mathbf{A}^{(n-i-1)} (u^{(i)} \mathbf{d} - y^{(i)} \mathbf{c}) \right] \quad (7)$$

where \mathbf{c} is a vector of feedback coefficients, \mathbf{A} is an $N \times N$ transition matrix for an SDM of order N , and $\mathbf{c} = (c_1, \dots, c_N)^T$ and \mathbf{d} describe how the input and feedback, respectively, are distributed.

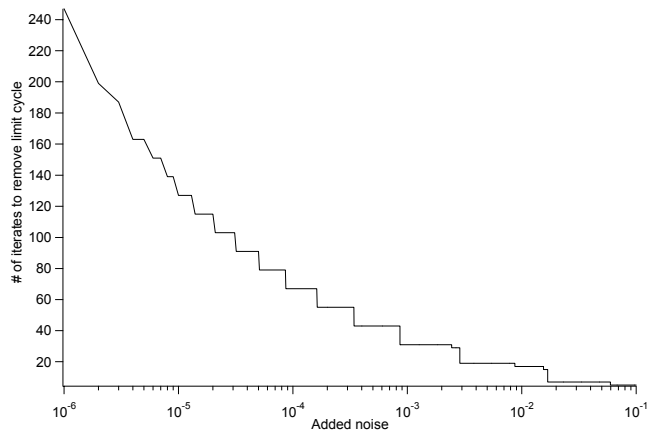


Figure 5. Number of iterates before a limit cycle is removed as a function of the added noise. Initial conditions were set to the most stable period 12 limit cycle for a 5th order SDM (corner frequency of 100kHz, OSR=64x44.1kHz).

It is interesting to compare this with the equivalent feedforward design, where the state space equations are given by (1) and (2). For both designs, the placement of the transition matrix \mathbf{A} in the state space equations is identical. However, for feedforward designs, the coefficient vector has no direct effect on the state space variables \mathbf{s} , and only acts as a weighting term on the quantisation. Whereas for the feedback design, the coefficient vector acts as a constant that is added or subtracted from the state space variables every iteration. This implies that the dynamics of feedback and feedforward designs are similar, although short term limit cycles are much more rare in feedback SDMs.

Modifications must be made in limit cycle detection and removal mechanisms, but both have been successfully applied to this design.

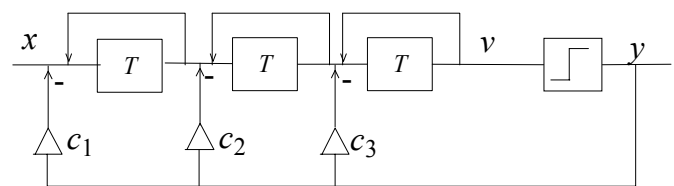


Figure 6. Block diagram of a third order feedback SDM.

7 Conclusion

In this work, we have described some of the properties of limit cycle detection and removal schemes which have been implemented for feedforward and feedback SDMs. These methods successfully detect any limit cycle up to a given period and may remove any limit cycle. They have a very low probability of false detection, are easy to implement, and outperform current techniques such as dithering. Implementation details will be fully described in later work. It is hoped that these methods may soon be realised in commercial SDMs.

References

- [1] F. de Jager, "Delta modulation - a method of {PCM} transmission using the one unit code," *Philips Research Report*, vol. 7, pp. 442-466, 1952.
- [2] S. Norsworthy, R. Schreier, and G. Temes, *Delta-Sigma Data Converters*: IEEE Press, 1997.
- [3] L. Risbo, "Sigma-Delta Modulators - Stability Analysis and Optimization," PhD Thesis, *Electronics Institute*. Lyngby: Technical University of Denmark, 1994, pp. 179. <http://eivind.imm.dtu.dk/publications/phdthesis.html>
- [4] S. Hein and A. Zakhor, *Sigma Delta Modulators: nonlinear decoding algorithms and stability analysis*. New York: Kluwer Academic Publishers, 1993.
- [5] S. H. Ardalan and J. J. Paulos, "An Analysis of Nonlinear Behavior in Delta-Sigma Modulators," *IEEE Transactions on Circuits and Systems I: Fundamental Theory and Applications*, vol. 34, pp. 593-603, 1987.
- [6] A. J. Magrath, "Algorithms and Architectures for High Resolution Sigma-Delta Converters," PhD Thesis, *Electronic Engineering*. London: King's College, University of London, 1996.
- [7] S. I. Mann and D. P. Taylor, "Limit cycle behavior in the double-loop bandpass Sigma Delta A/D converter," *IEEE Transactions on Circuits and Systems-II*, vol. 46, pp. 1086-1089, 1999.
- [8] N. Bridgett and C. P. Lewis, "Effect of initial conditions on limit cycle performance of second order sampled data sigma delta modulator," *Electronics Letters*, vol. 26, pp. 817 - 819, 1990.
- [9] V. Friedman, "The Structure of limit cycles in sigma delta modulation," *IEEE Trans. Communication*, vol. 36, pp. 972-979, 1988.
- [10] J. D. Reiss and M. Sandler, "The harmonic content of a limit cycle in a DSD bitstream," Proceedings of the Audio Engineering Society 116th Convention, Berlin, Germany, 2004.
- [11] D. Reefman, J. D. Reiss, E. Janssen, and M. Sandler, "Description of Limit Cycles in Feedback Sigma Delta Modulators," Proceedings of the Audio Engineering Society 117th Convention, San Francisco, USA, 2004.
- [12] D. Reefman, J. D. Reiss, E. Janssen, and M. Sandler, "Description of limit cycles in Sigma Delta Modulators," *accepted for IEEE Transactions on Circuits and Systems I*, pp. 30, 2004.
- [13] 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.
- [14] D. Reefman, J. D. Reiss, E. Janssen, and M. B. Sandler, "Stability Analysis of Limit Cycles in High Order Sigma Delta Modulators," Proceedings of the Audio Engineering Society 115th Convention, New York, New York, 2003.
- [15] S. P. Lipshitz and J. Vanderkooy, "Towards a Better Understanding of 1-Bit Sigma-Delta Modulators," Proceedings of the Audio Engineering Society 110th Convention, Amsterdam, Holland, 2001.
- [16] J. D. Reiss and M. Sandler, "On methods for detecting and eliminating limit cycles in SDMs (*in preparation*)," 2005.
- [17] 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