Design and implementation of sigma–delta digital to analog converter

SONIKA^{1,*}, D D NEEMA² and R N PATEL³

¹Department of Electronics and Communication Engineering, Chhatrapati Shivaji Institute of Technology, Durg, Chhattisgarh 491001, India

²Department of Electrical and Electronics Engineering, Yugantar Institute of Technology and Management, Rajnandgaon, Chhattisgarh 491441, India

³Department of Electrical and Electronics Engineering, Shri Shankaracharya Group of Institutions, SSTC, Bhilai, Chhattisgarh 491001, India
e-mail: sonika444@gmail.com; neemadd@gmail.com; ramnpatel@gmail.com

MS received 20 March 2016; revised 9 May 2017; accepted 19 September 2017; published online 1 June 2018

Abstract. This paper presents the design and implementation of a 16-bit sigma–delta digital to analog converter (DAC) for audio applications. In order to achieve high-order noise shaping without the stability problem inherent in the design of higher order loop, cascade structure may be used. One of the drawbacks in a multi-bit cascaded sigma delta modulator is the un-cancelled noise and the nonlinearity error. The idea of the proposed architecture is to present an improved version of cascaded multi-bit sigma–delta modulator to overcome these problems. Simulation results verify the superiority of the proposed modulator. Sampling rate conversion plays a predominant role in the signal processing and has a strong influence on the overall complexity and cost of efficient sigma–delta DAC. In this paper a technique to examine filter architectures that can be effectively applied in the interpolator design is also presented; the sigma–delta modulator design is implemented on a field programmable gate array (FPGA) for verification purpose and results are presented.

Keywords. FPGA (field programmable gate array); digital to analog converter; digital signal processing; sigma–delta modulator.

1. Introduction

Digital signal processing has evolved to a sophisticated level, bringing advantages like more robustness, noise insensitivity, reliability, testability and better production yield. To bring all these merits into real-world applications, it is desirable to convert digital-domain signals into real-world analog-domain representations and real-world analog signals into digital-domain representations.

Sigma–delta modulator (SDM) has become a popular method to implement high-resolution digital to analog converter (DAC); this method uses oversampling along with noise shaping for the suppression of quantization noise in signal frequency band. The main benefits of SDM are their inherent linearity and robust analog implementation [1]. These features of the SDM make it very attractive for a number of applications.

Field programmable gate array (FPGA) devices often require a data converter to provide analog outputs from digital domain representations. With the present day progress in density of FPGA devices, systems with an increasingly higher level of integrations are achievable [2].

The proposed SDM is implemented using FPGA and results are presented.

This paper first provides an overview of DAC architecture and discusses the design of interpolator filter. The next section shows the design of cascade, dual quantizer and single-loop noise shaping architecture; after this, simulation and implementation results are discussed and finally summarized in the conclusion.

2. DAC architecture

The basic diagram of oversampled DAC along with selected word length at each stage is shown in figure 1. It consists mainly of two blocks, namely the interpolator and the sigma delta modulator [3]. The sigma delta DAC we have designed is useful for synthesizing audio signal CD format. The audio DAC accepts digital 16-bit input data at sampling rate of 44.1 kHz. Then the signal is processed by the interpolator filter. The interpolation ratio for this work is considered as 128. This signal is then entered into the noise shaping loop, which shortens the word length drastically, typically to a single bit. Next, the truncated output signal of the noise shaping is converted by the internal DAC. The

*For correspondence

Figure 1. Block diagram of sigma–delta DAC.

Figure 2. Interpolator filter.

analog output of the internal DAC contains a linear replica of the digital input signal plus a large amount of the noise due to the quantization error introduced in the noise shaping loop. Since most of the noise lies outside the signal band, it can be almost completely suppressed by the analog low pass filter following the DAC.

2.1 Interpolator filter

The role of the interpolation filter is to raise the data rate to the oversampled value. A realistic specification for synthesizing audio signal is a transition band given earlier with .005 dB pass band ripple, and 75 dB stop band damping is selected for interpolator filter.

Efficient implementation of the digital interpolation filter preceding the noise shaping loop requires a multistage structure. The architecture of the interpolator filter is shown in figure 2. It consists of three stages, where the first two stages are designed using half band filter and the third stage is designed using CIC filter.

The oversampling ratio is selected as 128. In the first two stages, oversampling is done by a factor of 2 using half band filter and in the third stage, oversampling by a factor of 32 is obtained using CIC filter. The details of word length at each stage are shown in figure 2. Although CIC filter is an efficient implementation requiring no multipliers, thus reducing the cost significantly, still it is not used in the first stage because as per the specification the transition band is steep. The first stage interpolator requires minimum 75 dB of stop band attenuation for audio specifications, but the pass band to stop band attenuation in CIC is 50 dB, which is not sufficient. Therefore, half band filter is used in first two stages,

Table 1. Specification of these filters.

Filter	Normalized cut-off frequency	Stop band attenuation (dB)	Pass band ripple (dB)
HBF-1	0.22675	80	0.0001
HBF-2	0.11378	80	0.0001
CIC	0.003543	59.7	–

which has a good image rejection but the drawback is that the number of multipliers required is larger. The last stage is designed using CIC filter, which requires no multipliers. The CIC structure [4] consists of comb and integrator sections, with the comb section working at low sampling rate and integrator section performing at high sampling rate. The detailed specification of these filters is given in table 1.

2.2 First half band filter

The structure of the first half band filter is shown in figure 3. A hardware-efficient FIR filter in a tapped cascaded interconnection of identical sub-filters was used for implementing the half band filter [5].

All the coefficients are quantized and multiplied using CSD format. The advantage of multiplying in CSD format is that even multiplication is done as a shift and add operation. The coefficient of the filter was generated using a DSTool Box [6], a tool kit of delta sigma DACs. We adjusted all the word lengths such that the lowest possible amplitude of input signal is taken care of. The selected word length at each stage is shown in figure 4.

Figure 3. Interpolator structure.

Figure 4. The f2 filter.

Figure 5. CIC filter.

2.3 CIC filter

The third stage is designed using CIC filters; the main advantage is that they require no multipliers. Its structure consists of comb and integrator sections as shown in figure 5. The function for the entire CIC filter by combining the two stages of the filter at high sampling rate is given as

$$H(z) = \frac{(1 - z^{-M})^2}{M^2(1 - z^{-1})^2}.$$

Here a CIC filter of two stages is sufficient because the high frequency images are critical to the performance of the subsequent SDM. In this CIC allowed a bit growth of 4 bits in the integrator section (I/P bit 24-bit, O/P bit 28-bit) because of continuous accumulation of signals.

Preference is given to CIC filter as interpolator due to its simplicity and it requires no multiplier coefficient storage but only addition/subtraction. It is found that because of the absence of multipliers, it shows reliable improvements in terms of area and power saving.

3. Proposed 3rd-order cascade SDM for DAC

The digital to analog conversion is mainly achieved by a fully digital SDM loop at an oversampled clock rate. A data stream with 16-bit word length may be changed into a single-bit digital signal such that the baseband spectrum is preserved.

In this work the proposed design approach uses the error feedback structure as shown in figure 6. Here the

Figure 6. Error-feedback implementation of SDM.

quantization error E is obtained in analog form by subtracting the internal ADC's input from the DAC output; E is then fed back to the input through a $H_f(z)$ filter [7]. In this structure we can find that instead of the MSB output, the negative of the truncation error (consisting of the neglected LSBs) is fed back to the input through a filter with a transfer function $H_f(z)$. The error feedback structure is easier to design since the choice of the noise transfer function does not affect the signal transfer function. The signal transfer function is 1 and is not affected by the loop filter $H_f(z)$.

By appropriate choice of the loop filter $H_f(z)$, any noise transfer function can be obtained using the error feedback structure without affecting the signal. In the z domain the output of the loop is given by

$$V(z) = U(z) + [1 - H_f(z)]E(z) \quad (1)$$

where V , U and E are the z transform of the output v , the input u , and the truncation error E of the loop, respectively. To obtain a second-order noise shaping we can select the feedback filter transfer function as $H_f(z) = z^{-1}(2 - z^{-1})$; then NTF becomes $NTF = (1 - z^{-1})^2$.

We can see that first- and second-order noise shaping loops do not require multiplication of multibit words, but only delays, additions and shifts of the binary point are needed. Such circuits reduce the circuit complexity.

This section shows the proposed cascaded digital noise shaping loop using error feedback structure. This consists of a second-order modulator cascaded with a first-order modulator as shown in figure 7. The second-order modulator is designed in such a way that both the quantization error and the single-bit output data are fed back and negative of quantization error E_1 due to the single-bit noise shaping loop is fed into the next stage, where it is truncated to M bits. The truncated error is then converted into the analog form, high pass filtered in the analog filter $H_2(z)$ and added to the analog output of 1-bit DAC to cancel the noise E_1 in the output signal V . The selection of word length at each level is shown in figure 7. The word length is adjusted in such a way that it takes care of the lowest possible amplitude of input sample.

The z transform of the signal V_1 at the output of the 1-bit DAC is given by

$$V_1(z) = U(z) + E_1(z)(1 - z^{-1})^2. \quad (2)$$

Under ideal condition the z transform of the output signal V_2 from the correction path is given as

$$V_2(z) = -E_1(z) + E_m(z)(1 - z^{-1}). \quad (3)$$

Here E_1 and E_m are the z transform of the single-bit and multibit quantization error, respectively. The order of the overall noise shaping is now the sum of the order of two noise shaping loops and is then given by

$$V(z) = V_1(z) + V_2(z) \quad (4)$$

and the final output is given as

$$V(z) = U(z) + E_m(z)(1 - z^{-1})^3. \quad (5)$$

Under ideal condition the corrected output V of the noise shaping loop no longer contains the noise-shaped E_1 but the m -bit truncation error E_m shaped by the third-order noise transfer function, resulting in an improvement of nearly m bits in the SNR.

Behavioural simulations have been performed for the proposed architecture and figure 8 plots the power spectral

Figure 7. Proposed 3rd-order cascaded SDM design.

Figure 8. PSD of 3rd-order cascaded SDM design.

density for the proposed modulator and simulation shows that the peak signal to quantization noise ratio 124 dB is achieved when the input signal amplitude is half of the 1-bit quantizer output level. The quantizer of the architecture is equal to 4 bits, and the sampling frequency and bandwidth are equal to 5 MHz and 20 kHz, respectively.

4. Proposed 3rd-order dual quantization MASH SDM

Higher order noise shaping suffers from the stability problems and one of the methods to overcome this problem is using a cascade structure where lower order single-loop modulator is cascaded to design higher order modulator. Each single-loop modulator of the cascade structure converts the quantization error from the preceding modulator; the errors of all but the last single-loop modulator are then digitally cancelled. This section proposes a third-order cascade modulator realized using dual quantization DAC architecture along with error feedback topology.

Figure 9 shows a dual quantization DAC architecture. In this system the negative of the large quantization error e_1 due to the single-bit noise shaping loop is fed into a correction path, where it is truncated to M bits ($M > 1$). The

truncated error is then converted into analog form, high pass filtered in the analog filter $H_2(z)$ and added to the analog output of the 1-bit DAC to cancel (in practice to reduce) the noise e_1 in the output signal y [8].

Under ideal conditions, the corrected output y of the noise shaping loop no longer contains the noise-shaped e_1 ; rather, the M -bit truncation error e_M is shaped the same way as e_1 . This results in an improvement of nearly M bits in the SNR. In practice, a leakage of filtered e_1 noise into y will always occur. However, due to the zeros at dc in the noise filter function, this effect might not be significant if the system is carefully designed. Similarly, the nonlinearity error d_M of the M -bit DAC is high pass filtered and thus further reduced by the analog filter $H_2(z)$ [9].

Figure 10 illustrates a DAC system similar to the dual quantizer MASH structure. In this system, instead of simply truncating e_1 to M bits, a second noise-shaping loop is incorporated into the correction path that shapes the multi-bit truncation error e_M . As a result, the error due to e_M is even more reduced; the order of the overall noise shaping is now the sum of the order of the two noise-shaping loops.

Based upon dual-truncation MASH structure the proposed design of third-order dual quantization noise shaping system is shown in figure 11.

The z transform of the signal y_1 at the output of the 1-bit DAC is given by

$$y_1 = x_1 + \frac{(1 - z^{-1})^2}{1 - 0.75z^{-1}} E_1. \quad (6)$$

Under ideal conditions, the z transform of the output signal y_2 from the correction path is given as

$$y_2 = \frac{(1 - z^{-1})^3}{(1 - 0.75z^{-1})} E_m - \frac{(1 - z^{-1})^2}{1 - 0.75z^{-1}} E_1. \quad (7)$$

Here e_1 (e_m) is the z transform of the single-bit (multi-bit) quantization error E_1 (E_m). The sum of y_1 and y_2 , which is

Figure 9. Dual-truncation sigma–delta noise shaping loop.

Figure 10. A dual-truncation MASH structure.

Figure 11. A 3rd-order dual quantization noise-shaping system.

the z transform of the output signal from the dual quantization noise shaping system is then given by

$$y = x_1 + \frac{(1 - z^{-1})^3}{1 - 0.75z^{-1}} E_m. \quad (8)$$

From Eq. (8) we can find that under ideal conditions the corrected output y of the noise-shaping loop no longer contains the noise-shaping e_1 but the M -bit truncation error

E_M shaped by third-order noise transfer function. This results in an improvement of nearly M bits in the SNR.

The word lengths of the various adders and shift registers were optimized using simulation with various input signals. Maximum signal amplitude at each node was then used to determine the corresponding word length of the adders and shift register in the system. We have adjusted all the bit lengths such that it will take care of the lowest possible

Figure 12. PSD of 3rd-order dual quantization MASH SDM.

amplitude of input sample. The maximum input signal amplitude for the input 16-bit digital signal is $\pm 2^{15}$ to utilize the dynamic range of the noise shaping loop fully; the 1-bit quantizer output was set to be $\pm 2^{16}$, which is twice of the maximum input signal amplitude. Under various input conditions, limited simulations indicate that the internal signal amplitudes (e_1 , x_4) are always less than $\pm 2^{17}$. The word lengths of the adder and shift registers in the noise-shaping loop were therefore chosen to be 18 bits.

Simulation indicated that the word length of the adders associated with the quantizer can be much less than 18 bits. There are totally 5 adders used in the basic noise-shaping loop and 2 adders are used in correction path; hence this results in a total of five 18-bit adders needed for use of whole noise-shaping loop system.

As indicated in Eq. (8), the noise transfer function from the basic noise-shaping loop has a pole at $z = 0.75$ in addition to the two desired zeros. The pole helps increase the loop stability and reduces the internal signal amplitude for a given range of the input signal amplitude. This reduction in turn reduces the bit width requirement for the internal digital adders and shift registers.

A MATLAB system level simulation is performed; figure 12 plots the power spectral density for the proposed modulator and the simulation shows that the peak signal to quantization noise ratio of 111 dB is achieved. The quantizer of the architecture is equal to 4 bits and the sampling frequency and bandwidth are equal to 5 MHz and 20 kHz, respectively.

5. Proposed 3rd-order single-loop SDM for DAC

In order to achieve high resolution in an oversampled DAC, the single-loop SDM is also one of the choices. On that basis a 3rd-order single-loop modulator for DAC is proposed. It consists of two accumulators and error signal due to multi-bit quantizer is fed back to only the third accumulator as shown in figure 13.

The output signal of the modulator can be given as $V(z) = U(z)z^{-1} + e_m(z)(1 - z^{-1})^3$.

We can see that the final output of the proposed modulator contains input signal and the quantization noise receiving third-order noise shaping. We can find that the quantization noise originates from a multi-bit quantizer, so the modulator dynamic range is improved by an increase in the quantizer resolution N .

Behavioural simulations have been performed under the same condition mentioned before and figure 14 plots the power spectral density for the proposed modulator and simulation shows that the peak signal to quantization noise ratio of 119 dB is achieved.

6. Simulation results

There are no critical performance advantage and disadvantage of one digital noise-shaper topology over another. However, the cost of the hardware implementation may vary greatly. The loop operation is fully described by the

Figure 14. PSD of 3rd-order single-loop SDM.

Figure 13. Single-loop sigma-delta modulator.

Figure 15. Dynamic range comparisons.

difference equation relating the output to the input and the error. If two's-complement notation is used, the data overflow of the intermediate operations inside the loop does not affect the end result either, provided that the number of bits of the end result is within the designed bit resolution [10]. Thus, the most important objective in choosing digital noise-shaper topologies for a given noise-transfer function is efficient implementation. For this reason, the noise-shaper structures discussed will be error feedback structures due to their simplicity both analytically and in implementation.

In case of error feedback structure, the circuit implementing the subtracter and the loop filter $H(z)$ in the feedback path is a digital circuit. If no overflow occurs, the operation is perfect and no errors are introduced in these blocks. This makes the error feedback structure well suited for use in implementing digital noise-shaping loops. It often leads to a more efficient circuit implementation than the sigma-delta noise-shaping structures.

All the proposed designs in this paper use multi-bit internal DAC instead of a single-bit one to reduce the oversampling ratio and increase the conversion accuracy. However, the multi-bit DAC does introduce nonlinearity distortion, and the approach used in this paper for reducing the multi-bit DAC nonlinearity error is to use dual quantizer techniques. Here, the system uses two internal DACs; one is single-bit and the other is multi-bit. The single-bit DAC is used in the signal path while the multi-bit DAC is used in an added path called the correction path. Since the

multi-bit DAC is not directly placed in the signal path, its non-linearity error can be noise-shaped by an analog differentiator such that the in-band noise introduced from this error is very small at the system output. In addition, there is an added pole in the signal path noise transfer function, which helps in stabilizing the signal path noise-shaping loop and reduces the internal signal amplitude in the noise-shaping loop. This pole helps in reducing the noise gain at high frequencies at the cost of some increased low-frequency noise gain. A dual quantizer noise shaping system, with a pole at $z = 0.75$, is included in the noise-transfer function of the basic noise-shaping loop as shown in figure 11. After this, the second cascade design approach is discussed, which does not include a pole in the basic noise-shaping loop. A comparison of both the designs shows that the NTF with a pole at $z = 0.75$ results in increase of noise gain at low frequency by a factor of two, as compared with NTF with a pole at $z = 0$ whereas simulation results show that the pole helps in increasing the loop stability and reduces the internal signal amplitude for a given range of input signal amplitude. This reduction in turn reduces the bit width requirement for the internal digital adders and shift registers.

The MATLAB simulink tool is used for system level simulation of SDM. The audio DAC accepts digital 16-bit input data at sampling rate of 44.1 kHz. Then the signal is processed by the interpolator filter. The interpolation ratio is considered as 128. It is then interpolated to a 5.6448 MHz sampling rate using the three-stage interpolator as described in section 2. We assume a 4-bit quantizer adopted in all the designs. Figure 15 shows the simulated signal to noise distortion ratio (SNDR) as a function of the input signal strength for all the three designs and the summary of audio band SNDR for all proposed simulation is given in table 2.

7. Implementation details

The HDL workflow advisor of MATLAB was used to model and implement the design on to the FPGA. The HDL Workflow Advisor is a tool that generates HDL

Table 2. SNDR values.

Design method	Pole position	NTF (noise transfer function)	SNDR (dB)	Max stable amplitude (%)
3rd-order cascaded SDM architecture (figure 7)	Second-order error feedback noise-shaping loop (with all poles at $z = 0$)	$(1 - z^{-1})^3$	124	80
3rd-order dual quantization noise-shaping system (figure 11)	Second-order error feedback noise shaping loop (with one pole at $z = 0.75$)	$\frac{(1-z^{-1})^3}{1-0.75z^{-1}} E_m$	111	85
Single-loop 3rd-order design (figure 13)	Single-loop design	$(1 - z^{-1})^3$	119	82

code from fixed-point MATLAB code, and simulates and verifies the HDL against the fixed-point algorithm. HDL Coder then synthesizes the HDL code and optionally runs a place and route process that takes the circuit description produced by the previous mapping process, and emits a circuit description suitable for programming an FPGA.

For the implementation and optimization of proposed design the following steps are performed: first step is the generation of HDL code and a test bench for the proposed simulink subsystem. As we know system level simulation use floating point numbers, a fixpoint implementation had to be derived for the FPGA realization. For this the proposed simulink design is realized using a single-bit DAC in place of multi bit DAC to make the circuit compatible for generation of VHDL code and test bench; then, in the next step, verification of the generated HDL test bench using the ModelSim simulator is performed and report is generated to test that the HDL simulation matches the numerics and latency with respect to the fixed point.

MATLAB simulation after this synthesis and timing analysis is performed through integration with Xilinx® ISE synthesis tools. The Xilinx ISE controls all aspects of the development flow and provides synthesis report, RTL

schematic diagram and finally produce the bit stream for FPGA implementation.

The HDL coder generates the resource utilization report and high-level resource report where the resource utilization report summarizes multipliers, adders/subtractors and registers consumed by the device. The details of the resource utilization reports for all the three designs are shown in table 3 and the high-level resource report analyses the effects of optimizations; the proposed design uses optimization for reducing number of bits representing each data path, that is, use minimal bits while still meeting the SNR requirement of the application. The bit length is adjusted in such a way that it takes care of the lowest possible amplitude of input samples while meeting the SNR requirement. The details of word length are discussed in proposed design sections.

For the implementation purpose the complete design has been prototyped using different 6 FPGA kits. The synthesis report explores hardware resource, power and delay report. The synthesis analysis shows that due to the portability among different FPGA kits (Virtex-6, Spartan-6, Virtex-7 low voltage and Kirtex-7) synthesis report shows exactly the same hardware utilization report. All the three SDM designs use the same number of slice register, flip flop, etc. as shown in detail in figure 16.

Similarly, due to compatibility between Virtex-4 and Spartan-3A, FPGA kits provide the same hardware utilization report for all three SDM designs. The slice utilization report by Virtex-4 and Spartan-3A for all three designs is shown in detail in figure 17.

Similarly, results of implementation in terms of power, frequency and delay report have been observed for all designs using different FPGA kits. Synthesis results vary in different FPGA kits. Synthesis report using different FPGA kits is summarized in tables 4, 5 and 6 for single loop,

Table 3. Resource utilization report.

Detail	Dual quantizer design	Cascade design	Single-loop design
Multiplier	1	0	0
Adder/subtractor	18	10	6
Registers	5	5	3
RAM	0	0	0
Multiplexer	9	2	1

Figure 16. Details of slice utilization using Virtex-6, Spartan-6, Virtex-7 low voltage and Kirtex-7 for all three designs.

Figure 17. Details of slice utilization using Virtex-4 and Spartan-3A for all three SDM designs.

Table 4. Implementation details of third-order single-loop Δ - Σ modulator.

Implementation details	Virtex-4	Spartan-3A	Virtex-6	Spartan-6	Virtex-7 low voltage	Kirtex-7
Total on chip power (W)	0.340	0.027	4.262	0.064	0.529	0.107
Maximum frequency (MHz)	154.057	94.46	25.614	140.679	183.412	274.808
Maximum combinational path delay (ns)	6.069	7.427	2.23	5.591	2.509	0.396

Table 5. Implementation details of third-order cascade Δ - Σ modulator.

Implementation details	Virtex-4	Spartan-3A	Virtex-6	Spartan-6	Virtex-7 Low voltage	Kirtex-7
Total on chip power (W)	0.447	0.027	4.262	0.064	0.529	0.107
Maximum frequency (MHz)	85.91	55.008	179.072	12.568	144.98	216.86
Maximum combinational path delay (ns)	6.03	7.423	0.483	5.525	0.478	0.394

Table 6. Implementation details of third-order dual quantization noise shaping system.

Implementation details	Virtex-4	Spartan-3A	Virtex-6	Spartan-6	Virtex-7 Low voltage	Kirtex-7
Total on chip power (W)	0.452	0.034	1.293	0.014	0.529	0.107
Maximum frequency (MHz)	38.95	38.20	6.017	37.450	54.625	82.306
Maximum combinational path delay (ns)	27.06	33.29	11.872	21.762	15.057	9.721

cascade design and dual quantization, respectively. It can be clearly seen that all the implementations meet the required 6 MHz operating frequency. We can observe that the third-order dual quantization noise-shaping system requires the largest number of slices and also the operating frequency is the minimum at 38 MHz; the third-order single-loop SDM requires less number of slices and the operating frequency is 274 MHz.

8. Conclusion

In this paper, we have presented the design and implementation of a sigma-delta audio DAC on FPGA platforms. The sigma-delta converter has been studied using simulink and both the system level simulation and the RTL level simulation results have been presented. For the design of SDM, we had used first- and second-order digital noise

shaper loops with error feedback in a digital implementation. This topology has some definite advantages. The difference between quantizer input and output can be obtained by simply taking the LSBs of a digital number; no hardware is required. Also, the feedback FIR filter is very easily accomplished, especially when the coefficients are simple powers of 2. The advantage of the error-feedback structure is the lack of multipliers. Here, we use the simplest possible modulator design that is of first and second order. The interpolator design is implemented using half band and CIC filters. Preference is given to the CIC filter due to its simplicity and it requires no multiplier coefficient storage but only addition or subtraction. The architecture for changing the sampling rate decides the overall complexity and cost of D/A sigma–delta converter. This results in effective area and power saving. By selecting different design parameters, we are able to achieve different SNR ratios for various audio applications. The maximum achievable SNR from our designs is around 124 dB.

References

- [1] Norsworthy S R, Schreier R and Temes G C 2005 *Delta–sigma data converters: theory, design, and simulation*. New York: IEEE Press, pp. 309–332
- [2] Memon T D, Beckett P and Sadik A Z 2013 Power-area performance characteristics of FPGA-based sigma delta FIR filters. *J. Signal Process. Syst.* 70(3): 275–288
- [3] Schreier R 1993 An empirical study of high-order single-bit delta-sigma modulators. *IEEE Trans. Circuits Syst. Analog Digit. Signal Process.* 40(8): 461–466
- [4] Saramaki T 1987 Design of FIR filters as a tapped cascaded interconnection of identical sub-filters. *IEEE Trans. Circuits Syst.* 34(9): 1011–1029
- [5] Eegene B H 1981 An economical class of digital filters for decimation and interpolation. *IEEE Trans. Acoust. Speech Signal Process.* 29(2): 200–210
- [6] Schreier R The delta–sigma tool box. <http://www.mathworks.it/matlabcentral>, File Exchange >Control and System Modeling >Control Design > delsig
- [7] Naus P J, Dijkmans E C, Stikvoort E F, Holland D J and Brandinal B 1987 A CMOS stereo 16-bit D/A converter for digital audio. *IEEE J. Solid State Circuits* 22(3): 390–395
- [8] Xu X, Temes G C and Schreier R 1992 The implementation of dual-truncation sigma delta D/A converters. In: *Proceedings of the IEEE International Symposium on Circuits and Systems*, vol. 2, pp. 597–600
- [9] Leslie T C and Singh B 1990 An improved sigma delta modulator architecture. In: *Proceedings of the IEEE International Symposium on Circuits and Systems*, pp. 372–375
- [10] Sukumaran A and Pavan S 2014 Low power design techniques for single-bit audio continuous-time delta sigma ADCs using FIR feedback. *IEEE J. Solid State Circuits* 49(11): 2515–2525