IEICE TRANS. INF. & SYST., VOL.E107–D, NO.8 AUGUST 2024

PAPER Special Section on Multiple-Valued Logic and VLSI Computing

Delta-Sigma Domain Signal Processing Revisited with Related Topics in Stochastic Computing

Takao WAHO[†a)], ***Senior Member***, **Akihisa KOYAMA**[†], ***Nonmember***, ***and*** **Hitoshi HAYASHI**[†], ***Member***

SUMMARY Signal processing using delta-sigma modulated bit streams is reviewed, along with related topics in stochastic computing (SC). The basic signal processing circuits, adders and multipliers, are covered. In particular, the possibility of preserving the noise-shaping properties inherent in delta-sigma modulation during these operations is discussed. Finally, the root mean square error for addition and multiplication is evaluated, and the performance improvement of signal processing in the delta-sigma domain compared with SC is verified.

key words: *delta-sigma modulator, stochastic computing, signal processing, adder, multiplier, noise shaping, RMSE*

1. Introduction

Delta-sigma (ΔΣ) modulators are widely used in analog/digital interfaces [1], [2]. They oversample the analog input signal at a frequency much higher than the Nyquist rate to produce a bitstream consisting of 0 s and 1 s. A typical structure of the ΔΣ modulator and input/output waveforms are shown in Fig. 1. As shown in Fig. 1 (b), the frequency of appearance of 1 s in the output increases as the input sinusoidal signal increases. The ΔΣ modulator is suitable for miniaturized CMOS technology because the stringent requirements imposed on imperfect analog circuits can be alleviated by employing high-speed digital circuitry. Consequently, the application area expands beyond digital audio to include broadband communications and various sensor nodes [3], [4].

In a ΔΣ-type analog-to-digital (A/D) converter, a decimation filter is placed after the modulator to convert the oversampled bitstream into Nyquist-rate multi-bit words. A ΔΣ-type digital-to-analog (D/A) converter, in contrast, uses an interpolator to generate an oversampled multi-bit word stream. A following ΔΣ modulator generates a bitstream, which is then smoothed by a low pass filter (LPF) to obtain an analog output signal. These converters are currently used in modern signal processing systems shown in Fig. 2 (a). Suppose we can develop a processor for the bitstream from a ΔΣ modulator without the use of a decimator or interpolator. This would enable a simple signal processing system as shown in Fig. 2 (b), *i.e.*, the ΔΣ-domain signal processing system [5]–[8]. Since its proposal, various solutions have been suggested, and its importance has been re-evaluated in recent years as the environment around LSI-based signal processing systems has changed significantly [9], [10]. In this paper, we will focus on this ΔΣ-domain signal processing scheme.

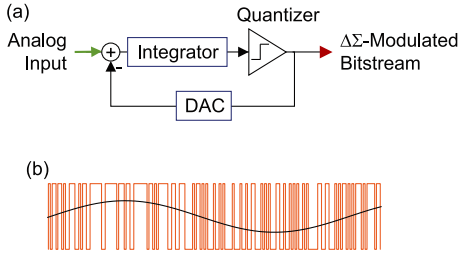

Fig. 1 (a) Block diagram of a ΔΣ modulator and (b) output bitstream example for an input sinusoidal signal.

The stochastic computing (SC) approach is similar to the signal processing in the ΔΣ domain in that it processes signals represented as bitstreams [11], [12]. The advantage of SC over conventional binary systems is that operations can be performed with simple logic gates. Its similarity to SC suggests that ΔΣ-domain processing can inherit its advantages. On the other hand, SC has its own problems, such as the need for long bitstreams to suppress errors caused by randomness and improve accuracy. This leads to serious problems of increased energy consumption and processing latency. These can be solved in the ΔΣ domain through its unique noise shaping characteristics, which will be discussed later. The SC has become a recent trend in its application to the edge AI [13]–[15], and the trend is likely to reach the ΔΣ-domain signal processing in the near future. The purpose of this paper is to review such ΔΣ-domain techniques with related topics in SC.

We have previously published a paper discussing signal processing in the ΔΣ domain along these lines [16]. In this paper, we extend that discussion, focusing in particular on the noise shaping characteristics specific to ΔΣ modulation and the associated improvements in processing accuracy. Section 2 briefly describes SC in comparison to ΔΣ modulation and reviews several circuits proposed for the SC and ΔΣ domains. In Sect. 3, adders, one of the most fundamental signal processing elements proposed previously, are introduced with some FFT analysis results. In Sect. 4, other fundamental elements, multipliers, are discussed in a similar manner. Section 5 presents the accuracy evaluation results for addition and multiplication based on the root mean square error (RMSE).

Manuscript received October 26, 2023.
Manuscript revised January 15, 2024.
Manuscript publicized April 17, 2024.

[†]Faculty of Science and Technology, Sophia University, Tokyo, 102–8552 Japan.
a) E-mail: t-waho@sophia.ac.jp

DOI: 10.1587/transinf.2023LOP0006

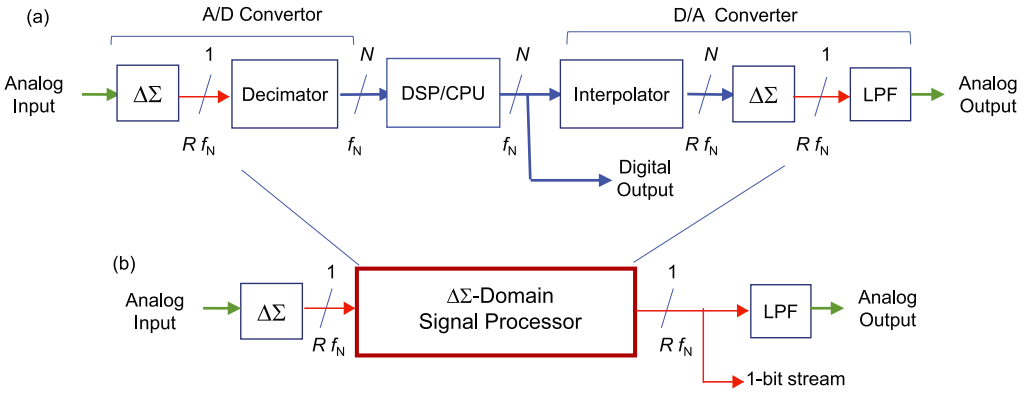

Fig. 2 (a) Conventional digital signal processing with analog interfaces and (b) ΔΣ-domain processing. R is the oversampling ratio defined by the ratio of the sampling frequency to the Nyquist rate f_N.

Since research in this area has been going on for more than 30 years and covers a wide variety of technical topics, there are several re-proposals of the same content or independent proposals. This paper presents them in as organized a form as possible.

2. Brief Overview

2.1 SC Compared with ΔΣ-Domain Signal Processing

Figure 3 (a) and (b) shows a stochastic number generator (SNG) and its symbol used in the SC. A comparator generates the output bitstream by comparing the input magnitude with a random number from a pseudorandom number generator as shown in Fig. 3 (c). This output bitstream looks similar to the ΔΣ-modulated one shown in in Fig. 1; the frequencies of occurrence of 1 s and 0 s are high near the peak and valley of the sine wave, respectively.

Although the shape of the two output bitstreams is similar in the time domain, there is a clear difference when viewed in the frequency domain. Figure 4 shows the FFT analysis results for output bitstreams obtained from the ΔΣ modulator and the SNG. By convention, rounding errors that occur when analog values are represented as binary values of 1 or 0 are referred to as quantization noise. The SC quantization noise spectrum is white or frequency independent, whereas the ΔΣ modulator output has an upward slope, which is called "noise shaping." In this example, since a first-order ΔΣ modulator is used, the slope is around 20 dB/dec, and the slope becomes steeper if a higher-order modulator is used.

If the input signal is bandwidth limited, the noise-shaping characteristics mean that the signal-to-quantization-noise ratio (SQNR), or simply the signal-to-noise ratio (SNR), can be improved by using a digital lowpass filter that attenuates the high-frequency quantization noise components. The SNR can be also improved by increasing the over sampling ratio (OSR), which is defined by dividing the sampling rate by the Nyquist rate. In the case of constant inputs, although the OSR cannot be defined, a similar improvement in SNR can be achieved. Thus, using a ΔΣ-modulated bitstream with noise-shaping characteristics can achieve signal processing with higher accuracy than the SC.

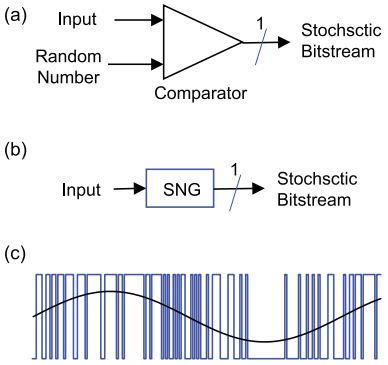

Fig. 3 (a) Stochastic number generator, (b) its circuit symbol, and (c) output bitstream example for an input sinusoidal signal.

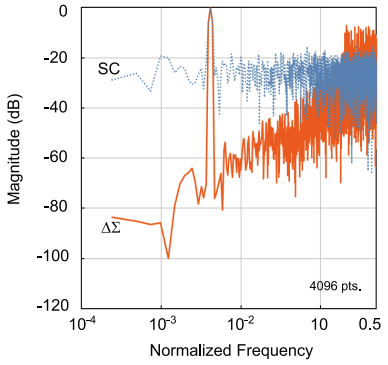

Fig. 4 Comparison of SC and ΔΣ output spectra.

2.2 ΔΣ and SC Circuits

Many signal processing circuits have been proposed for ΔΣ-modulated and SC bitstreams. Table 1 summarizes some of these examples grouped by function. For adders, those using a 1-bit full adder for ΔΣ and using a multiplexer (MUX) for the SC are well known. Also, adders using a toggle flip-flop (T-F/F) for the SC have been proposed and are noted to be

Table 1 ΔΣ and SC signal processing circuits.

Functions		ΔΣ	SC
Adders	Full Adder	[17] [18] [19] [20] [21] [22]	
	MUX	[23]	[24] [25] [26] [27]
	T-F/F		[28] [29]
	D-F/F	[30]	
	Counter	[31] [32]	[33] [34] [35]
Multipliers	Chopping	[36] [37] [6] [7] [8] [38] [39] [40]	
	AND/XNOR	[41] [42] [18] [19] [42] [20] [22] [43] [44] [10]	[24] [28] [45] [25] [26] [27] [46] [47] [48] [49]
	LPF+Binary	[6]	
	Built-in ΔΣ	[50]	
Filters	FIR	[51] [52] [53] [54] [55] [56] [57] [58] [59] [60] [61] [62]	[26] [63] [64] [65]
	IIR	[66] [67] [68] [69] [57] [59] [70]	[26]
Others	Absolute	[71] [72]	[73]
	Max/Min	[74]	[75] [76] [77]
	Logarithm/Hyperbolic	[78]	[25] [79] [80] [73]

more accurate than those using a MUX. Counter-type adders generate a binary output representing the number of 1 s in a bitstream, which requires additional circuitry to convert it back to a bit sequence.

As for multipliers, the chopping type for ΔΣ and the AND/XNOR gate for SC are well known. The latter is sometimes used in ΔΣ, but it is desirable to avoid it because of the loss of ΔΣ-specific noise shaping characteristics, as mentioned below. For ΔΣ, an LPF with a binary multiplier has also been proposed, and the chopping type can be considered as a simplified version of this multiplier. An interesting proposal is one with a built-in multiplying function in the feedback path of the ΔΣ modulator. It is unique in that the errors introduced in the multiplication are also subject to shaping.

Filters were mainly discussed for ΔΣ. Absolute, max/min, and logarithmic/hyperbolic functions were discussed in the SC. The following sections describe the details of the adder and multiplier. For filters and other circuits, refer to the references.

3. Adders

3.1 SC and ΔΣ Adders

A multiplexer (MUX)-based adder commonly used in the SC [24] is shown in Fig. 5 (a). Assume that the two inputs are stochastic bitstreams, $x_S(n)$ and $y_S(n)$, representing X and Y. Also, assume that the selection signal $p(n)$ is a random variable such that the probabilities of occurrence of 0 and 1 are equal (= 1/2). Then, the resulting bitstream $z(n)$ is the scaled sum of $(X+Y)/2$. Note that the output is halved. For k inputs, the addition results in scaling by $1/k$, which degrades the processing accuracy.

In a MUX-based adder, noise is added to the output due to randomness not only in the input signals, but also in the selection signal. Also, one bit in each input bitstream is discarded as it passes through the MUX. To address these concerns, a T-F/F based adder has been proposed, as shown in Fig. 5 (b), where the selection signal is generated from the input bitstream and the bit loss is prevented [29]. In fact, it is reported that the signal processing accuracy can be

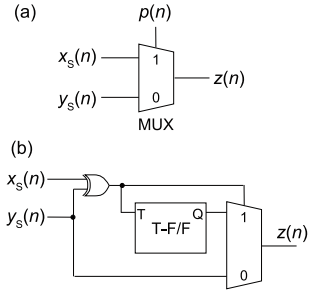

Fig. 5 SC adders using (a) a multiplexer (MUX) [24] and (b) a toggle flip-flop (T-F/F) [29].

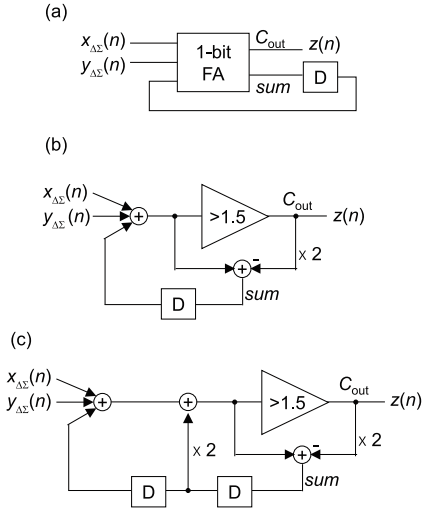

Fig. 6 (a) ΔΣ adder using a full adder (FA) [17] and (b) first-order and (c) second-order error-feedback ΔΣ modulators [1]. D is a delay element.

improved.

For ΔΣ-modulated bitstreams, an adder using a 1-bit full adder (FA) was proposed [17], which is shown in Fig. 6 (a). This circuit was proposed with reference to an earlier proposal for Δ-modulated bitstreams [81]. Note that this configuration is equivalent to a first-order error-feedback ΔΣ modulator shown in Fig. 6 (b) [21]. Because this modulator is a first-order configuration, the output has first-order

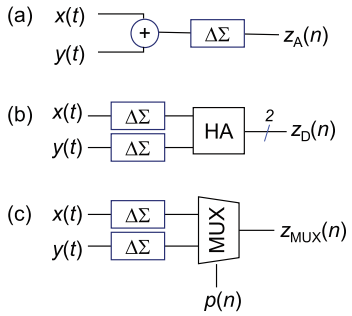

Fig. 7 Adders used to evaluate noise shaping characteristics with (a) an analog adder (b) a half adder (HA), and (c) a multiplexer (MUX).

noise shaping characteristics even if the input ΔΣ-modulated bitstreams are second-order noise shaped. If an output with second-order characteristics is required, a second-order error feedback configuration is required as shown in Fig. 6 (c). Another adder has been reported that implements the addition logic directly [18]. It consists of different circuit elements from those shown in Fig. 6 (a). However, analysis of the circuit operation shows that they are essentially the same in terms of circuit operation.

3.2 Noise Shaping Characteristics in Adders

As mentioned in Sect. 2, the advantage of ΔΣ signal processing over SC was the possibility of improved processing accuracy due to its unique noise shaping characteristics. To investigate whether these characteristics could be preserved during addition, three types of adders were considered for two sinusoidal input signals with slightly different frequencies, $x(t)$ and $y(t)$, as shown in Fig. 7: (a) analog addition before ΔΣ modulation, (b) digital addition using the half adder (HA) after ΔΣ modulation, and (c) MUX addition after ΔΣ modulation. The resulting noise shaping characteristics are shown in the Fig. 7.

Since addition is a linear operation, the frequency response after addition is the sum of the individual frequency responses. Therefore, it is not surprising that the noise shaping characteristic can be preserved in both analog addition $z_A(n)$ and digital addition $z_D(n)$. On the other hand, if MUX addition is performed in the same way as SC, the noise shaping characteristic disappears and the quantization noise becomes white, as shown in the same figure as $z_{\mathrm{MUX}}(n)$. This is due to the loss of bit information contained in one of the ΔΣ-modulated bitstreams after passing through the MUX. As a similar example, it was observed that the noise shaping characteristics disappear for a bit sequence generated by removing one bit every other bit from a ΔΣ-modulated bitstream.

Figure 9 shows the results of using the adder for the ΔΣ-modulated bitstream shown in Fig. 6 (a). It can be seen that virtually the same noise shaping characteristics can be obtained as in the case of the analog/digital addition shown in Fig. 8. Interestingly, the adder proposed for the SC shown in Fig. 5 (b) also resulted in almost the same noise shaping

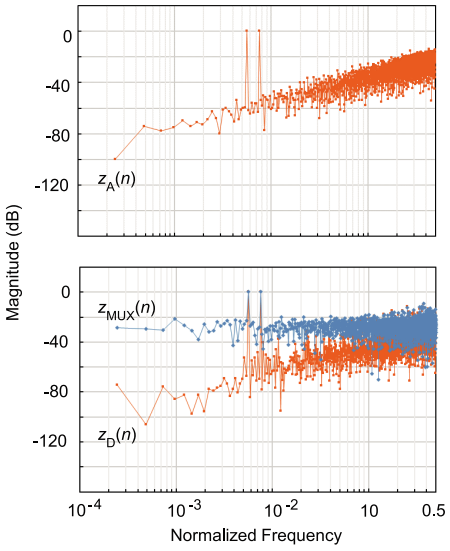

Fig. 8 Noise shaping characteristics obtained for the circuits shown in Fig. 7.

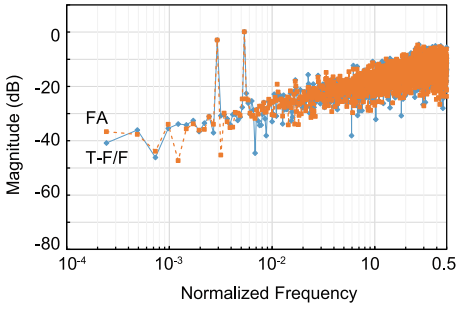

Fig. 9 Noise shaping characteristics obtained for the circuits shown in Figs. 5 (b) and 6(a).

characteristics. To obtain the noise shaping characteristic, it is necessary to take the difference of the quantization error generated for two consecutive samples, which requires one delay element included in the circuit. The reason why the noise shaping characteristics were obtained with the T-F/F adder is probably because the T-F/F behaves as a delay element. This suggests that the noise-shaping characteristics can be obtained from ΔΣ-modulated bitstream inputs even with a circuit proposed for SC operations.

4. Multipliers

4.1 SC and ΔΣ Multipliers

Simple logic gates, AND and XNOR, shown in Fig. 10 are used to multiply unipolar and bipolar encoded SC bitstreams, respectively [24]. These logic circuits were also used for ΔΣ bitstreams as shown in Table 1. However, since multiplication is a nonlinear operation, there is no guarantee that the noise shaping characteristics are preserved as they are in the addition operation. In fact, the product of two functions becomes a convolution of the original functions after the

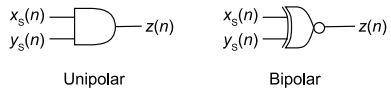

Fig. 10 SC multipliers for unipolar and bipolar encodings [24].

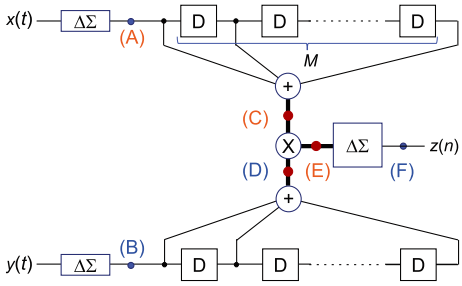

Fig. 11 Multiplier for ΔΣ-modulated bitstreams [6].

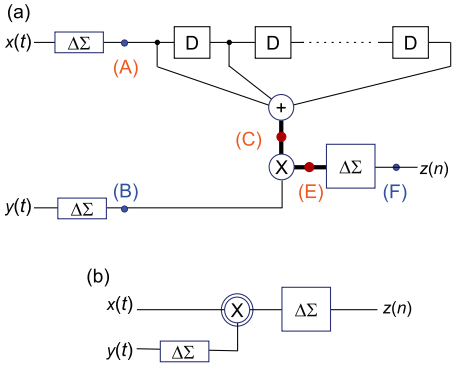

Fig. 12 Variations of the multiplier shown in Fig. 11: (a) a configuration with a simplified lower path and (b) a multiplier replacing the upper path with the original analog path, called a chopping multiplier [36]. Double circled multiplier denotes an analog switch.

Fourier transform. Therefore, high-frequency quantization noise components are mixed into the low-frequency signal band, and the noise shaping characteristics are corrupted.

To prevent this, an attempt has been made to suppress the high frequency components with a lowpass filter (LPF) before the product operation in order to preserve the noise shaping characteristics. Such a circuit is shown in Fig. 11, where two FIR filters are used as LPFs and the product is calculated on the resulting multi-bit binary signals which appear at the nodes (C) and (D). A ΔΣ modulator is placed after the binary multiplier to pass the signal at (E) in a bitstream to the next stage as $z(n)$. In other words, the ΔΣ modulator placed between (E) and (F) converts a multi-bit word into a single bit $z(n)$. It should be noted that our ΔΣ-domain signal processing uses single-bit sequences rather than multi-bit words as signals passing through the system.

Consider the circuit shown in Fig. 12 (a), where the FIR filter in the lower half path shown in Fig. 11 is removed. Since the analog input signal $x(t)$ is digitized at node (A) and reconstructed at node (C), it can be rewritten as shown in Fig. 12 (b) using an analog switch or chopper instead of a binary multiplier. This is a chopping-type multiplier that

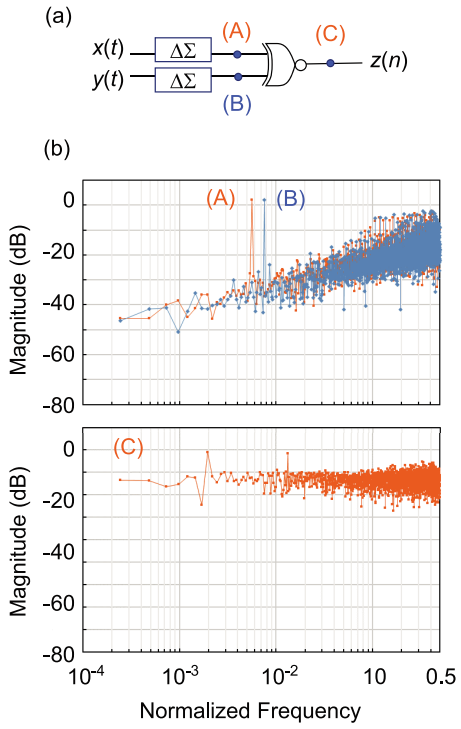

Fig. 13 (a) SC multiplier used with two ΔΣ-modulated input bitstreams and (b) noise shaping characteristics obtained at the three node in (a).

was proposed independently of the one shown in Fig. 11.

4.2 Noise Shaping Characteristics in Multipliers

First, consider the output spectrum of the XNOR-type SC multiplier shown in Fig. 13 (a). Here, two ΔΣ-modulated bitstreams for two sine waves, $x(t)$ and $y(t)$, with slightly different frequencies are used as the inputs. Figure 13 (b) shows that the spectra at nodes (a) and (b) are certainly noise shaped. At the output node (C), however, the noise shaping characteristics disappeared, although there are two peaks at positions corresponding to the sum and difference of the input frequencies, which means that multiplication is indeed taking place. This proves the prediction mentioned above.

Next, consider the spectrum at each node of the multiplier shown in Fig. 11, again assuming two bitstreams for sine waves with slightly different frequencies for the inputs as described above. Figure 14 shows the simulation results. The two sinusoidal signals after the ΔΣ modulation have normal shaping spectra as shown by (A) and (B). After passing through the FIR filter, the high-frequency components were suppressed and the quantization noise component has been flattened as shown by (C) and (D). The notches in the high-frequency region are due to the transfer characteristics of the FIR filter, which in this case is a first-order sinc filter. At node (E) after the product operation, there are two peaks corresponding to the sum and difference of the input frequency, and a white quantization noise component. Although the shape of node (E) is similar to (C) in Fig. 13, the noise level is much lower, confirming the effect of the LPFs. The signal

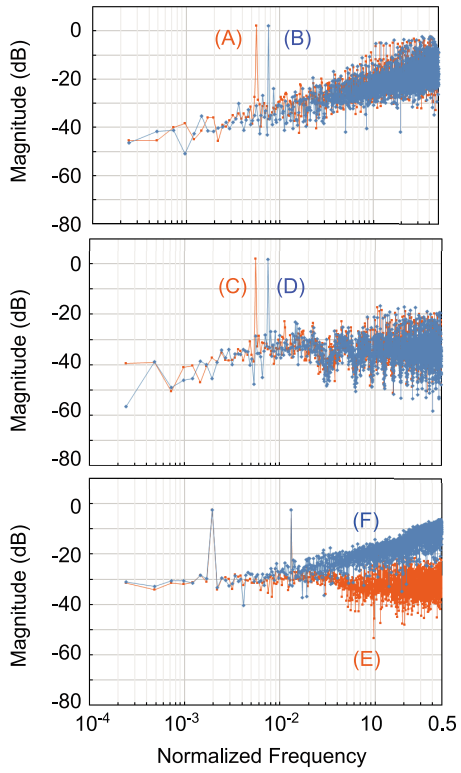

Fig. 14 Noise shaping characteristics obtained at each node in Fig. 11.

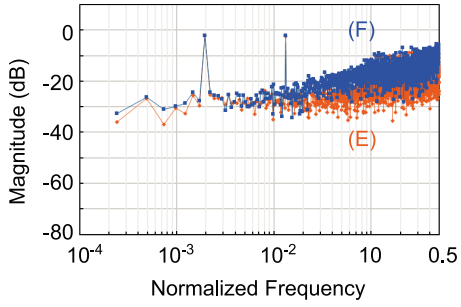

Fig. 15 Noise shaping characteristics obtained at the two node in Fig. 12 (a).

at the ΔΣ-modulated node (F) shows the noise-shaped quantization noise. Note that the low-frequency noise components of the signal at (E) are still present after ΔΣ modulation, which results in no low-frequency shaping as seen in (A) and (B).

Finally, the noise shaping characteristics of the chopping-type multiplier in Fig. 12 are shown in Fig. 15. The characteristics for nodes (A), (B), and (C) are the same as those shown in Fig. 14. The characteristics for nodes (E) and (F) are almost identical to those shown in Fig. 14. Hence, it can be concluded that the noise shaping characteristics can be maintained even in the chopping type.

The reason why noise shaping characteristics can be preserved with chopping-type multipliers is as follows [7], [8]. Since the Fourier transform of a sine wave is a delta function, if one of the functions in the convolution is a sine wave, the result is simply a shift of the other function along the frequency axis, so that the shaping characteristics are preserved. In the configuration shown in Fig. 12, one analog input is a sine wave, so the noise shaping characteristics of the other input, the ΔΣ-modulated bitstream that controls the chopping, remain unchanged in the output. This holds true even when the analog inputs are general functions including a constant, if their signal bandwidth is sufficiently narrow.

5. RMSE Evaluation

Reducing power in the frequency domain means reducing variation in the time domain. In other words, maintaining noise shaping characteristics means reducing noise power in the low-frequency signal band, which in turn improves the accuracy of signal processing. To confirm this, the accuracy of the sum and product operations for two constant inputs was evaluated. When constant inputs are represented by bitstreams, quantization errors occur that depend on the input values and the corresponding bitstreams. To avoid these dependencies, the RMSE was obtained for 1000 randomly generated pairs of inputs, and used as an accuracy evaluation index. The addition and multiplication were performed for 1000 pairs of randomly generated constant inputs, a_i and b_i, using the circuit shown in Fig. 16 (a) through (d). The root mean square error was evaluated. For addition it is defined as

$$RMSE = \sqrt{\frac{1}{1000}\sum_{i=1}^{1000}(c_i^{SS/\Delta\Sigma S} - (a_i + b_i))^2}. \quad (1)$$

The RMSE was also obtained for multiplication with a similar formula.

For the SC bitstreams, Figs. 16 (a) and (b) were used as the adder and multiplier. First-order filters, denoted by sinc1, *i.e.*, a simple moving average, were also used to estimate the results. Figures 16 (c) and (d) show those for the ΔΣ-modulated bitstream, respectively. A k-th order sinc filter, sinc$k$, was used to calculate the estimated value, where $k = 1, 2, 3$. ΔΣj represents a j-th order ΔΣ modulator, where k was assumed to be 1 or 2 in this study. The attenuation slope of the transfer function of the k-order sinc filter is balanced with the positive slope of the noise shaping characteristics of the $j(= k)$-order ΔΣ modulator. Therefore, for the combination of ΔΣj and sinc$k$, the quantization noise can be effectively blocked if $j < k$. Therefore, the combinations $(j, k) = (1, 2)$ and $(j, k) = (2, 3)$ were chosen and the results were compared with those of the SC and $(j, k) = (1, 1)$ cases.

In Fig. 16 (e), the RMSE values thus obtained are plotted as a function of N, the length of the bitstream to which the sinc filter is applied. The RMSE decreases as N increases. More importantly, the slope of the RMSE becomes steeper as the order of the ΔΣ modulator and sinc filter increases. The slope of the SC RMSE is $1/\sqrt{N}$, which is consistent with remarks in the previous literature [82]–[84].

The RMSE for ΔΣ1+sinc1 decreased in proportion to

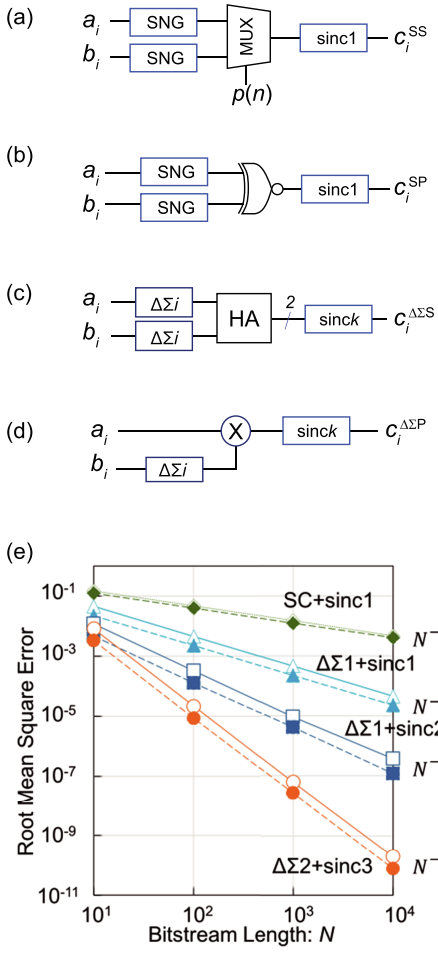

Fig. 16 Circuits used for estimating the root mean square error (RMSE) (a)-(d) and RMSE as a function of the bitstream length (N) (e). Open and closed plots in (e) represent the sum and product, respectively. "SC+sinc1" is the results from the circuits shown in (a) and (b), while "ΔΣi+sinc$k$" corresponds to the circuits (c) and (d), where $i = 1, 2$ and $k = 1, 2, 3$.

$1/N$. This can be explained as follows. Based on the linear model of the first-order ΔΣ modulator, its noise transfer function can be expressed as

$$NTF(z) = 1 - z^{-1}. \tag{2}$$

The transfer function of the first-order sinc filter can be represented as

$$H_1(z) = \frac{1}{N}\frac{1 - z^{-N}}{1 - z^{-1}}. \tag{3}$$

Therefore, the quantization noise $Q_1(z)$ after the output of the first-order ΔΣ modulator passes through the first-order sinc filter can be written as

$$\begin{aligned} Q_1(z) &= H_1(z)NTF(z)E(z) \\ &= \frac{1}{N}(1 - z^{-N})E(z). \end{aligned} \tag{4}$$

In the time domain, this corresponds to

$$q_1(n) = \frac{1}{N}[e(n) - e(n - N)], \tag{5}$$

where $e(n)$ is the quantization noise of the ΔΣ modulator. If its variance can be expressed as $\langle e(n)^2 \rangle = \sigma_e^2$, then

$$\sqrt{\sigma_{q1}^2} = \sqrt{\langle (q_1(n) - \overline{q_1(n)})^2 \rangle} = \sqrt{\frac{2\sigma_e^2}{N^2}} \propto \frac{1}{N} \tag{6}$$

holds, which indicates that the RMSE is proportional to $1/N$. In the same way, the slope values could be obtained in other cases and verified to be consistent with the values shown in Fig. 16.

The SNG in this study used random numbers generated using the Mersenne twister (MT) method. Similar results were obtained with random numbers generated by the LFSR method. In contrast, it is known that SNG using Sobol' sequences can improve the RMSE [47], [48]. However, even in this case, the RMSE decreased only in proportion to $1/N$. As shown in Fig. 16, it is clear that the use of ΔΣ-modulated bitstreams can improve the performance even further.

6. Conclusion

Signal processing with ΔΣ modulated bitstreams is reviewed, along with related topics in SC. Specifically, adders and multipliers are discussed as basic signal processing circuits. The noise-shaping characteristics inherent in ΔΣ modulation are preserved for adders that include a delay element. The noise-shaping characteristics are lost in a simple MUX-type adder, which is widely used in SC. However, they can be preserved in a special T-F/F MUX adder. Multiplication is a nonlinear operation, and to preserve the noise-shaping characteristics, the high-frequency components in the ΔΣ bitstream must be blocked before multiplication. A chopping-type multiplier effectively works well to preserve the noise-shaping characteristics. The noise-shaping characteristics reduce the RMSE and improve the signal processing accuracy compared to SC. In particular, the reduction is significant when a second-order ΔΣ modulator with an appropriate LPF is used instead of a first-order one.

References

[1] S. Pavan, R. Schreier, and G.C. Temes, Understanding Delta-Sigma Data Converters, 2nd Ed., IEEE Press Series on Microelectronic Systems, Wiley, 2017.
[2] T. Waho, Introduction to Analog-to-Digital Converters: Principles and Circuit Implementation, River Publishers Circuits and Systems, River Publishers, 2019.
[3] J.M. de la Rosa, "AI-assisted sigma-delta converters-application to cognitive radio," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol.69, no.6, pp.2557–2563, 2022.
[4] J. Wang, Z. Jia, L.A. Campos, and C. Knittle, "Delta-sigma modulation for next generation fronthaul interface," Journal of Lightwave Technology, vol.37, no.12, pp.2838–2850, 2019.
[5] M. Freedman and D.G. Zrilić, "Nonlinear arithmetic operations on the delta sigma pulse stream," Signal processing, vol.21, no.1, pp.25–35, 1990.
[6] F. Maloberti, "Non conventional signal processing by the use of sigma delta technique: a tutorial introduction," Proc. 1992 IEEE International Symposium on Circuits and Systems, pp.2645–2648 vol.6, 1992.

[7] V. da Fonte Dias, "Sigma-delta signal processing," 1994 IEEE International Symposium on Circuits and Systems (ISCAS), pp.421–424 vol.5, 1994.
[8] V. da Fonte Dias, "Signal processing in the sigma-delta domain," Microelectronics Journal, vol.26, no.6, pp.543–562, 1995.
[9] D.G. Zrilic, Functional Processing of Delta-Sigma Bit-Stream, Springer, 2020.
[10] N. Temenos and P.P. Sotiriadis, "A stochastic computing sigma-delta adder architecture for efficient neural network design," IEEE Journal on Emerging and Selected Topics in Circuits and Systems, vol.13, no.1, pp.285–294, 2023.
[11] W.J. Gross and V.C. Gaudet, Stochastic Computing: Techniques and Applications, Springer International Publishing, 2019.
[12] M. Alawad and M. Lin, "Survey of stochastic-based computation paradigms," IEEE Trans. Emerg. Topics Comput., vol.7, no.1, pp.98–114, 2019.
[13] I. Polian, J.P. Hayes, V.T. Lee, and W. Qian, "Guest editors' introduction: Stochastic computing for neuromorphic applications," IEEE Design and Test, vol.38, no.6, pp.5–15, 2021.
[14] Y. Liu, S. Liu, Y. Wang, F. Lombardi, and J. Han, "A survey of stochastic computing neural networks for machine learning applications," IEEE Trans. Neural Netw. Learn. Syst., vol.32, no.7, pp.2809–2824, 2021.
[15] B.R. Gaines, "A conceptual framework for stochastic neuromorphic computing," IEEE Design and Test, vol.38, no.6, pp.16–27, 2021.
[16] T. Waho, A. Koyama, and H. Hayashi, "Delta-sigma domain signal processing: A review with relevant topics in stochastic computing," 2023 IEEE 53rd International Symposium on Multiple-Valued Logic (ISMVL), pp.88–93, 2023.
[17] P. O'leary and F. Maloberti, "Bit stream adder for oversampling coded data," Electronics Letters, vol.26, no.20, pp.1708–1709, 1990.
[18] H. Fujisaka, N. Masuda, M. Sakamoto, and M. Morisue, "Arithmetic circuits for single-bit digital signal processing," ICECS'99. Proc. ICECS '99. 6th IEEE International Conference on Electronics, Circuits and Systems (Cat. no.99EX357), pp.1389–1392, 1999.
[19] H. Fujisaka, M. Sakamoto, and M. Morisue, "Bit-stream signal processing circuits and their application," IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences, vol.85, no.4, pp.853–860, 2002.
[20] C.-W. Ng, N. Wong, and T.-S. Ng, "Bit-stream adders and multipliers for tri-level sigma-delta modulators," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol.54, no.12, pp.1082–1086, 2007.
[21] C.W. Ng, N. Wong, and T.S. Ng, "Efficient FPGA implementation of bit-stream multipliers," Electronics Letters, vol.43, no.9, pp.496–497, 2007.
[22] C.W. Ng, N. Wong, and T.S. Ng, "Quad-level bit-stream adders and multipliers with efficient FPGA implementation," Electronics Letters, vol.44, no.12, pp.722–724, 2008.
[23] F. Maloberti and P. O'Leary, "Processing of signals in their oversampled delta-sigma domain," China., 1991 International Conference on Circuits and Systems, vol.1, pp.438–441, 1991.
[24] B.R. Gaines, "Techniques of identification with the stochastic computer," Proc. International Federation of Automatic Control Symposium on Identification, Progue, 1967.
[25] B.D. Brown and H.C. Card, "Stochastic neural computation. I. computational elements," IEEE Trans. Comput., vol.50, no.9, pp.891–905, 2001.
[26] Y.-N. Chang and K.K. Parhi, "Architectures for digital filters using stochastic computing," 2013 IEEE International Conference on Acoustics, Speech and Signal Processing, pp.2697–2701, 2013.
[27] B. Yuan, Y. Wang, and Z. Wang, "Area-efficient scaling-free DFT/FFT design using stochastic computing," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol.63, no.12, pp.1131–1135, 2016.
[28] W.J. Poppelbaum, C. Afuso, and J.W. Esch, "Stochastic computing elements and systems," Proc. Nov. 14-16, 1967, fall joint computer conference, pp.635–644, 1967.
[29] V.T. Lee, A. Alaghi, J.P. Hayes, V. Sathe, and L. Ceze, "Energy-efficient hybrid stochastic-binary neural networks for near-sensor computing," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017, pp.13–18, 2017.
[30] Y. Liang, Z.G. Wang, Q. Meng, and X.D. Guo, "Design of high speed high snr bit-stream adder based on ΣΔ modulation," Electronics letters, vol.46, no.11, pp.752–753, 2010.
[31] Y. Liu and W. Tang, "A delta sigma based finite impulse response filter for EEG signal processing," 2015 IEEE 58th International Midwest Symposium on Circuits and Systems (MWSCAS), pp.1–4, 2015.
[32] A. Klein and W. Schumacher, "Algebraic operations on delta-sigma bit-streams," Mathematical and Computational Applications, vol.23, no.3, p.49, 2018.
[33] K. Kim, J. Lee, and K. Choi, "Approximate de-randomizer for stochastic circuits," 2015 International SoC Design Conference (ISOCC), pp.123–124, 2015.
[34] N. Temenos and P.P. Sotiriadis, "Nonscaling adders and subtracters for stochastic computing using Markov chains," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol.29, no.9, pp.1612–1623, 2021.
[35] N. Temenos and P.P. Sotiriadis, "Modeling a stochastic computing nonscaling adder and its application in image sharpening," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol.69, no.5, pp.2543–2547, 2022.
[36] W.N. Cheung, "Correlation measurement by delta-sigma modulation," IEEE Trans. Ind. Electron. and Control Instrumentation, vol.IECI-26, no.2, pp.88–92, 1979.
[37] A. Cichocki and R. Unbehauen, "Application of SC MOS delta and delta-sigma modulators to nonlinear signal processing," 1988., IEEE International Symposium on Circuits and Systems, vol.3, pp.2233–2236, 1988.
[38] F. Op't Eynde, "A power metering ASIC with a sigma-delta-based multiplying ADC," Proc. IEEE International Solid-State Circuits Conference - ISSCC '94, pp.186–187, 1994.
[39] E. Dallago, G. Sasone, M. Storti, and G. Venchi, "Experimental analysis and comparison on a power factor controller including a delta-sigma processing stage," IEEE Trans. Ind. Electron., vol.45, no.4, pp.544–551, 1998.
[40] A. Daglio, P. Malcovati, and F. Maleberti, "A multiplier-free digital rms calculation unit for integrated microsystems," Proc. Third International Workshop on Design of Mixed-Mode Integrated Circuits and Applications (Cat. no.99EX303), pp.183–186, IEEE, 1999.
[41] D. Lagoyannis and K. Pekmestzi, "Multipliers of delta-sigma sequences," Radio and Electronic Engineer, vol.51, no.6, pp.281–286, 1981.
[42] T.S. Lande, T.G. Constandinou, A. Burdett, and C. Toumazou, "Running cross-correlation using bitstream processing," Electronics Letters, vol.43, no.22, pp.1181–1183, 2007.
[43] P. Gonzalez-Guerrero, X. Guo, and M. Stan, "SC-SD: Towards low power stochastic computing using sigma delta streams," 2018 IEEE International Conference on Rebooting Computing (ICRC), pp.1–8, 2018.
[44] P. Gonzalez-Guerrero, S.G. Wilson, and M.R. Stan, "Error-latency trade-off for asynchronous stochastic computing with ΔΣ streams for the IoT," 2019 32nd IEEE International System-on-Chip Conference (SOCC), pp.97–102, 2019.
[45] B.R. Gaines, "Stochastic computing systems," Advances in information systems science, pp.37–172, 1969.
[46] H. Sim and J. Lee, "A new stochastic computing multiplier with application to deep convolutional neural networks," 2017 Proc. 54th ACM/EDAC/IEEE Design Automation Conference (DAC), pp.1–6, 2017.
[47] S. Liu and J. Han, "Energy efficient stochastic computing with Sobol sequences," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2017, pp.650–653, 2017.
[48] S. Liu and J. Han, "Toward energy-efficient stochastic circuits using parallel Sobol sequences," IEEE Transactions on Very Large Scale

Integration (VLSI) Systems, vol.26, no.7, pp.1326–1339, 2018.
[49] B. Li, M.H. Najafi, and D.J. Lilja, "Low-cost stochastic hybrid multiplier for quantized neural networks," ACM Journal on Emerging Technologies in Computing Systems (JETC), vol.15, no.2, pp.1–19, 2019.
[50] F.N. Buhler, A.E. Mendrela, Y. Lim, J.A. Fredenburg, and M.P. Flynn, "A 16-channel noise-shaping machine learning analog-digital interface," 2016 IEEE Symposium on VLSI Circuits (VLSI-Circuits), pp.1–2, 2016.
[51] C.-H. Wei and N.-C. Chen, "Sigma-delta modulation adaptive digital filter," 1988 IEEE International Symposium on Circuits and Systems (ISCAS), vol.1, pp.523–526, 1988.
[52] P.W. Wong, "FIR digital filters with sigma-delta modulated inputs and their implementation," IEEE International Symposium on Circuits and Systems, vol.2, pp.1260–1263, 1990.
[53] P.W. Wong, "Fully sigma-delta modulation encoded FIR filters," IEEE Trans. Signal Process., vol.40, no.6, pp.1605–1610, 1992.
[54] Q. Huang and G.S. Moschytz, "Analog multiplierless LMS adaptive FIR filter structures," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process., vol.40, no.12, pp.790–794, 1993.
[55] S. Kershaw and M. Sandler, "Digital signal processing on a sigma-delta bitstream," IEE Colloquium on Oversampling Techniques and Sigma-Delta Modulation, pp.9/1–9/8, 1994.
[56] S. Summerfield, S.M. Kershaw, and M.B. Sandler, "Sigma-delta bitstream filtering in VLSI," Proc. 1994 37th Midwest Symposium on Circuits and Systems, vol.2, pp.1200–1203, 1994.
[57] H. Qiuting, "A capacitor-area efficient realization of sigma-delta modulator-based FIR/IIR analog linear-phase filters," 1996 IEEE International Symposium on Circuits and Systems (ISCAS), vol.1, pp.69–72, 1996.
[58] A.C. Thompson, P. O'Shea, Z.M. Hussain, and B.R. Steele, "Efficient single-bit ternary digital filtering using sigma-delta modulator," IEEE Signal Process. Lett., vol.11, no.2, pp.164–166, 2004.
[59] T. Memon, P. Beckett, and A.Z. Sadik, "Sigma-delta modulation based digital filter design techniques in FPGA," International Scholarly Research Notices, vol.2012, pp.1–10, 2012.
[60] T.D. Memon, P. Beckett, and A.Z. Sadik, "Power-area-performance characteristics of FPGA-based sigma-delta fir filters," Journal of Signal Processing Systems, vol.70, pp.275–288, 2013.
[61] T.D. Memon and P. Beckett, "The impact of alternative encoding techniques on field programmable gate array implementation of sigma-delta modulated ternary finite impulse response filters," Australian Journal of Electrical and Electronics Engineering, vol.10, no.1, pp.107–116, 2013.
[62] A. Vlachos, N. Temenos, and P.P. Sotiriadis, "Exploring the effectiveness of sigma-delta modulators in stochastic computing-based FIR filtering," 2021 10th International Conference on Modern Circuits and Systems Technologies (MOCAST), pp.1–4, 2021.
[63] H. Ichihara, T. Sugino, S. Ishii, T. Iwagaki, and T. Inoue, "Compact and accurate digital filters based on stochastic computing," IEEE Trans. Emerg. Topics Comput., vol.7, no.1, pp.31–43, 2019.
[64] K. Papachatzopoulos, C. Andriakopoulos, and V. Paliouras, "Novel noise-shaping stochastic-computing converters for digital filtering," 2020 IEEE International Symposium on Circuits and Systems (ISCAS), pp.1–5, 2020.
[65] N. Temenos, A. Vlachos, and P.P. Sotiriadis, "Efficient stochastic computing fir filtering using sigma-delta modulated signals," Technologies, vol.10, no.1, p.14, 2022.
[66] D.A. Johns and D.M. Lewis, "Sigma-delta based IIR filters," Proc. 34th Midwest Symposium on Circuits and Systems, vol.1, pp.210–213, 1991.
[67] B.R. Owen and D.A. Johns, "A single-column structure for delta-sigma based IIR filters," Proc. 1992 IEEE International Symposium on Circuits and Systems, vol.5, pp.2413–2416, 1992.
[68] D.A. Johns and D.M. Lewis, "Design and analysis of delta-sigma based IIR filters," IEEE Trans. Circuits Syst. II, Analog Digit. Signal Process., vol.40, no.4, pp.233–240, 1993.
[69] D.A. Johns, D.M. Lewis, and D. Cherepacha, "Highly selective 'analog' filters using $\delta\sigma$ based IIR filtering," 1993 IEEE International Symposium on Circuits and Systems, vol.2, pp.1302–1305, 1993.
[70] N. Saraf, K. Bazargan, D.J. Lilja, and M.D. Riedel, "IIR filters using stochastic arithmetic," 2014 Design, Automation & Test in Europe Conference & Exhibition (DATE), pp.1–6, 2014.
[71] K. Hayashi, T. Katao, H. Fujisaka, T. Kamio, and K. Haeiwa, "Piecewise linear circuits operating on first-order multi-level and second-order binary sigma-delta modulated signals," 2007 18th European Conference on Circuit Theory and Design, pp.683–686, IEEE, 2007.
[72] D. Zrilic, G. Petrovic, and W. Tang, "Novel solutions of a delta-sigma-based rectifying encoder," IEEE Trans. Circuits Syst. II, Exp. Briefs, vol.64, no.10, pp.1242–1246, 2017.
[73] M.H. Najafi, P. Li, D.J. Lilja, W. Qian, K. Bazargan, and M. Riedel, "A reconfigurable architecture with sequential logic-based stochastic computing," ACM Journal on Emerging Technologies in Computing Systems (JETC), vol.13, no.4, pp.1–28, 2017.
[74] Y. Hidaka, H. Fujisaka, M. Sakamoto, and M. Morisue, "Piecewise linear operations on sigma-delta modulated signals," 9th International Conference on Electronics, Circuits and Systems, vol.3, pp.983–986, IEEE, 2002.
[75] A. Alaghi and J.P. Hayes, "Exploiting correlation in stochastic circuit design," 2013 IEEE 31st International Conference on Computer Design (ICCD), pp.39–46, 2013.
[76] M. Lunglmayr, D. Wiesinger, and W. Haselmayr, "Design and analysis of efficient maximum/minimum circuits for stochastic computing," IEEE Trans. Comput., vol.69, no.3, pp.402–409, 2020.
[77] N. Temenos and P.P. Sotiriadis, "Stochastic computing max & min architectures using markov chains: Design, analysis, and implementation," IEEE Transactions on Very Large Scale Integration (VLSI) Systems, vol.29, no.11, pp.1813–1823, 2021.
[78] H. Fujisaka, T. Kamio, C.-J. Ahn, M. Sakamoto, and K. Haeiwa, "Sorter-based arithmetic circuits for sigma-delta domain signal processing—part i: Addition, approximate transcendental functions, and log-domain operations," IEEE Trans. Circuits Syst. I, Reg. Papers, vol.59, no.9, pp.1952–1965, 2012.
[79] P. Li, D.J. Lilja, W. Qian, K. Bazargan, and M. Riedel, "The synthesis of complex arithmetic computation on stochastic bit streams using sequential logic," 2012 IEEE/ACM Proc. International Conference on Computer-Aided Design (ICCAD), pp.480–487, 2012.
[80] P. Li, D.J. Lilja, W. Qian, M.D. Riedel, and K. Bazargan, "Logical computation on stochastic bit streams with linear finite-state machines," IEEE Trans. Comput., vol.63, no.6, pp.1474–1486, 2014.
[81] N. Kouvaras, "Operations on delta-modulated signals and their application in the realization of digital filters," Radio and Electronic Engineer, vol.48, no.9, pp.431–438, 1978.
[82] K.F. Cheung and P.Y.H. Tang, "Sigma-delta modulation neural networks," IEEE International Conference on Neural Networks, vol.1, pp.489–493, 1993.
[83] W. Qian, X. Li, M.D. Riedel, K. Bazargan, and D.J. Lilja, "An architecture for fault-tolerant computation with stochastic logic," IEEE Trans. Comput., vol.60, no.1, pp.93–105, 2011.
[84] A. Alaghi and J.P. Hayes, "Survey of stochastic computing," ACM Transactions on Embedded computing systems (TECS), vol.12, no.2s, pp.1–19, 2013.

Takao Waho received the B.S., M.S., and Ph.D. degrees in physics from Waseda University in 1973, 1975, and 1978, respectively. He joined Electrical Communications Laboratories, Nippon Telegraph and Telephone Public Corporation (now NTT) in 1975, where he investigated III-V compound semiconductor device technology, including MBE growth, GaAs surface passivation, resonant tunneling diodes (RTDs), and their circuit applications. He also served as secretary of the Japan Electronic Industry Development Association (JEIA) Technical Committee on Advanced Information Processing Integrated Systems from 1997 to 1999. Since 1999, he has been a professor at the Faculty of Science and Technology, Sophia University, Tokyo, and is currently a professor emeritus. His current research interests include analog IC design and signal processing, such as analog-to-digital converters, delta-sigma modulators, and multiple-valued logic circuits. He received the Distinctive Contribution Paper Award in 1996 and 2020, and the Long Term Contribution Award in 2020 from the IEEE Computer Society. Dr. Waho served as Editor of the IEICE Transactions on Electronics (Japanese Edition) from 2002 to 2004, as Technical Committee (TC) Chair of IEICE Electron Devices from 2005 to 2007, and as TC Chair of IEEE Computer Society Multiple-Valued Logic from 2008 to 2009. He also served as General Chair of the Topical Workshop on Heterostructure Microelectronics (TWHM) held in Nagano, Japan, in August 2011 and of the IEEE International Symposium on Multiple-Valued Logic (ISMVL) held in Toyama, Japan, in May 2013.

Akihisa Koyama received the B.E. degree in Electrical Engineering from Sophia University in 2023. He is now investigating signal processing using delta-sigma modulated bitstreams towards his master degree thesis.

Hitoshi Hayashi received the B.E. degree in Electronics Engineering from the University of Tokyo in 1990. He joined NTT Radio Communication Systems Laboratories in 1992. He was engaged in the research and development of wireless communication systems. During 2000 - 2001, he was a Visiting Scientist with Massachusetts Institute of Technology, USA. In April 2012, he joined the Faculty of Science and Technology, Sophia University, Tokyo. He was the recipient of the 2012 IEICE Communications Society Excellent Paper Award.