

# Mean Circuit Design Using Correlated Random Bitstreams in Stochastic Computing

Feiyu Li, Guangjun Xie, Jie Han, *Senior Member IEEE*, Yongqiang Zhang, *Member, IEEE*

**Abstract**—Stochastic computing (SC) is an approximate computing technique that has been considered in recent years as an alternative to conventional binary encoding and is compatible with CMOS technology. Stochastic circuits have the advantages of low power, small area, and high fault tolerance. However, highly accurate computation often takes a lot of time because of the fluctuations in random numbers and the requirement for decorrelation between random bitstreams. In this work, mean circuits are proposed to exploit the correlation between the random bitstreams for an arbitrary number of inputs. With the proposed mean circuit, high-accuracy and low-cost mean filtering designs are implemented. Experimental results indicate that the implementations of mean filtering show 10% and 93% reductions in area and power, compared with multiplexer-based designs.

**Keywords**—Mean circuit, correlated bitstream, stochastic computing.

## I. INTRODUCTION

Stochastic computing (SC) [1] has recently regained significant attention due to its fault tolerance [2] and low hardware cost of arithmetic units. It has been applied to image processing [3, 4], neural networks (NNs) [5, 6], and digital filters [7, 8]. Stochastic numbers are represented by random bitstreams composed of 0s and 1s. There are two basic number formats in SC: the unipolar and bipolar representations. In the unipolar representation, SC encodes numbers in the range [0, 1]. The number of 1s in a bitstream divided by the length of the bitstream, denoted by  $P_x$  here, encodes the actual value  $X$ . For example, 01111011 represents 6/8. In the bipolar representation, the numerical value of  $X$  is  $2P_x - 1$ , so the range of bipolar format is [-1, 1]. For example, 01111011 represents 4/8. A stochastic number generator (SNG) is usually composed of a liner-feedback shift register (LFSR) and a comparator, as shown in Fig. 1. In each clock cycle, if the binary number  $n$  is greater than the number generated by the LFSR, a 1 or a 0 is generated.

Stochastic arithmetic operations can be performed in a very simple form. For example, the unipolar multiplication can be implemented by an AND gate, as shown in Fig. 2(a); the implementation of unipolar and bipolar scaled addition can be achieved by just a multiplexer, as shown in Fig. 2(b).

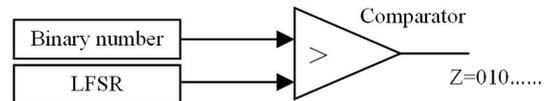


Figure 1. A stochastic number generator (SNG).

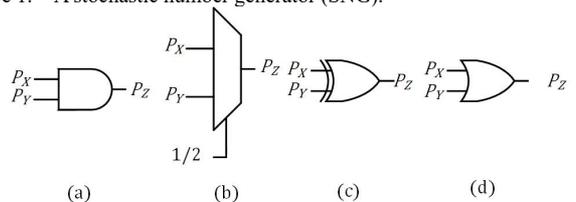


Figure 2. Stochastic arithmetic operations. (a) Unipolar multiplication. (b) Unipolar and bipolar scaled addition. (c) Positively correlated unipolar absolute subtraction. (d) Negatively correlated unipolar addition.

In SC, the computing accuracy depends on the stochastic computing correlation (SCC) [9]. Independent bitstreams are necessary to achieve high accuracy. The relationship between bitstreams is determined by SCC, which indicates the degree of similarity between two bitstreams. When  $SCC=0$ , two bitstreams are said to be independent of each other, while on the occasion that  $SCC=1$  or  $SCC=-1$ , it is called a positive or negative correlation. Positively and negatively correlated bitstreams can be generated by sharing an LFSR, as shown in Figure 3. However, in order to generate independent bitstreams, one way is to use multiple LFSRs and another way is to use a decorrelated circuit [10, 11]. However, utilizing the correlation between bitstreams is also an effective approach [12, 13], as in [14] a positively correlated unipolar absolute subtraction is used, as shown in Fig. 2(c). An OR gate can implement unipolar addition by using negatively correlated bitstreams, as shown in Fig. 2(d). In this paper, we focus on the unipolar correlation mean circuit design in SC and improve the accuracy of the circuit by changing the random number source of the circuit. It is experimentally demonstrated that the proposed mean circuits perform with higher accuracy than previous designs.

The rest of the paper proceeds as follows. Section II presents the background. Section III describes the design principles of the proposed circuits. Section IV shows the experimental results and application of mean circuits. Section V gives a summary.

This work was supported by the Fundamental Research Funds for the Central Universities of China (Grant No. JZ2020HGTA0085, No. JZ2020HGQA0162), and by the Natural Sciences and Engineering Research Council (NSERC) of Canada (Project Number: RES0048688).

F. Li, G. Xie, and Y. Zhang are with the School of Microelectronics, Hefei University of Technology, Hefei 230009, China (e-mail: 535413508@qq.com; gjxie8005@hfut.edu.cn; ahzhangyq@hfut.edu.cn)

J. Han is with the Department of Electrical and Computer Engineering, University of Alberta, Edmonton, AB T6G 1H9, Canada (e-mail: jhan8@ualberta.ca)

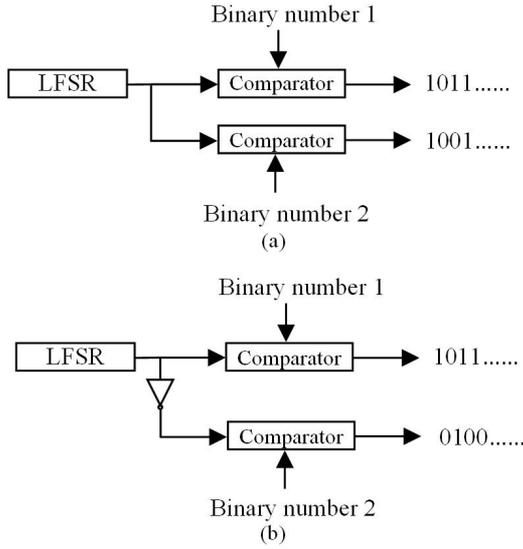


Figure 3. The generation of (a) positively and (b) negatively correlated stochastic bitstreams.

## II. BACKGROUND

### A. Stochastic Mean Circuits (SMCs)

A 2-input multiplexer (MUX) can realize the function  $P_Z = P_S P_X + (1 - P_S) P_Y$ , where  $P_X$  and  $P_Y$  are the inputs,  $P_S$  is the select signal. When  $P_S = 0.5$ ,  $P_Z = 0.5(P_X + P_Y)$ , so it is a 2-to-1 SMC. Fig. 4 shows some conventional SMCs [15], which compute the average of  $2^n$  input bitstreams. Such a circuit uses one multiplexer with  $2^n$  inputs and  $n$  uncorrelated bitstreams as select signals. By changing the input as well as the select signal, a mean circuit for different inputs can be obtained. For example, a 16-to-1 SMC can be changed to a 9-to-1 SMC, where its first 8 ports are for different inputs, the last 8 ports are for the same inputs, and the 4 select signals are set to 0.5, 0.5, 0.5, and 0.1111. The bitstreams for the select signals must be of mutual independence.

### B. Absolute Valued Subtraction

Under certain specific conditions, correlation can be used to design stochastic computation elements with correlated input bitstreams. An XOR gate with independent inputs performs the function  $z = x_1(1 - x_2) + x_2(1 - x_1)$ . However, for positively correlated inputs where  $x_1$  and  $x_2$  have a maximum overlap of 1s, the circuit computes  $z = |x_1 - x_2|$  [13]. If  $x_1$  is greater than  $x_2$ , the output  $z = x_1 - x_2$ .

### C. Correlation Addition

An OR gate with independent inputs performs the function  $z = x_1 + x_2 + x_1 x_2$ . If negatively correlated inputs are used, the circuit computes  $z = \max(1, x_1 + x_2)$ . If  $x_1 + x_2$  is less than 1, the output is  $x_1 + x_2$ .

## III. THE PROPOSED MEAN CIRCUITS

In this section, the stochastic mean circuits are presented by using correlated input bitstreams, so it is referred to as CORSMCs. The design of the proposed mean circuits consists of three steps: the first step is to generate coefficients, the

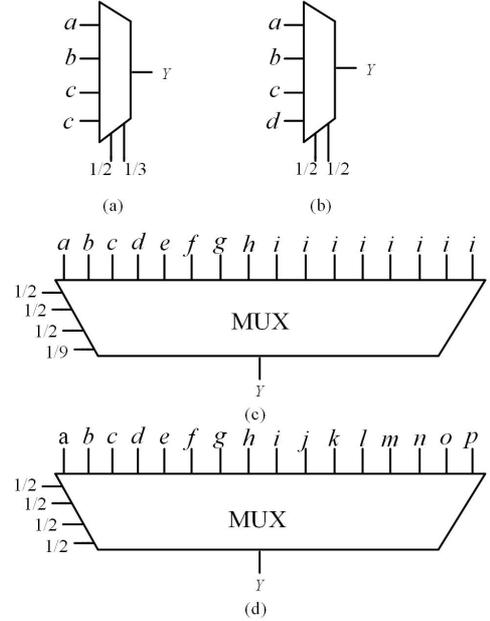


Figure 4. Stochastic mean circuits. (a) 3-to-1 SMC. (b) 4-to-1 SMC. (c) 9-to-1 SMC. (d) 16-to-1 SMC.

second step realizes the multiplication of the coefficients and inputs, and the third step realizes the summation of the products.

Absolute valued subtraction is used in the first step of the design [14]. If  $x_1$  is larger than  $x_2$ , the circuit computes  $z = x_1 - x_2$ , and the result shows that the output bitstreams  $z$  is negatively correlated with the input bitstreams. By sharing a random number source, the SNG first generates  $P_n = 1, N-1/N, N-2/N, \dots, 1/N$ . These  $n$  bitstreams are all positively correlated. Each bitstream needs to be subtracted from the neighboring bitstreams on the right to obtain  $n$  bitstreams with the value  $1/N$  whose sections of '1's are interleaved. The XOR gate is used to implement subtraction. These  $n$  negatively correlated bitstreams are used to encode the mean coefficients.

The second step consists of using some AND gates. The  $n$  input bitstreams are first generated by other random number sources. After that, the products of the coefficients and the inputs are realized using AND gates. The last step is to implement the negatively correlated addition. Since the sum of coefficients is 1 and the multiplication use AND gates, the result must be less than the maximum value of inputs. Thus, the result of the addition is less than or equal to 1. The correlation-based stochastic mean circuit (CORSMC) is shown in Fig. 5.

This design needs to generate two independent bitstreams, so two different random number sources are needed. For the random number sources, two LFSRs can be taken, or a counter can be used as the random number source, while the other random number source takes the inverse of the counter to produce a Halton sequence [16, 17]. For example, if  $X_1 = 001, 010, 011, \dots$  in a 3-bit bipolar encoding format,  $X_2 = \text{flip}(X_1)$  becomes  $100, 010, 110, \dots$ . This can greatly improve the computing accuracy of the final results.

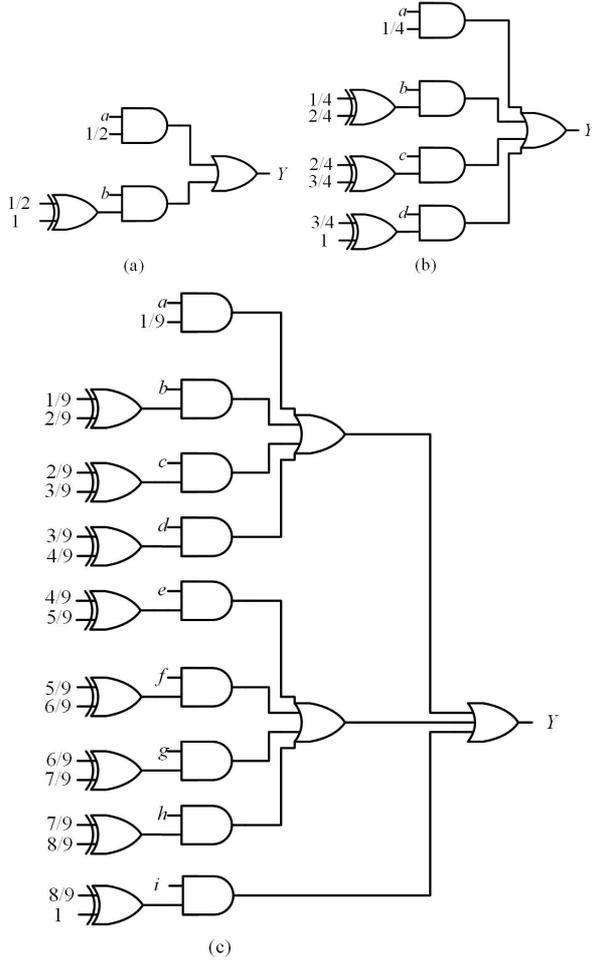


Figure 5. The proposed (a) 2-to-1 CORSMC. (b) 4-to-1 CORSMC. (c) 9-to-1 CORSMC.

#### IV. EXPERIMENTAL RESULTS

##### A. Computing Accuracy

This subsection compares the errors of the SMC and CORSMC by measuring the mean absolute error (MAE). MAE is defined as

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i|. \quad (1)$$

Three-, four-, and nine-input mean circuits are compared. Bitstream lengths of 16, 64, 128, 256, 512, and 1024 are considered. One thousand sets of random inputs are selected for the experiment. The input numbers are randomly generated. The experiment is repeated for different bitstream lengths. For the CORSMC, a random number source, a counter, and LFSRs are used respectively. The SMC uses LFSRs as random number sources. The experimental results are shown in Fig. 6.

##### B. Mean Filter

Mean filtering uses simple algorithms and has the advantage of fast operation. Mean filtering usually takes a window of  $n \times n$ . In this paper, a window of  $3 \times 3$  is used, as shown in Fig. 7. This design takes an average of 9 pixels for replacing the pixel values at the center. The experiment selects a gray-scale image with an 8-bit representation for each pixel

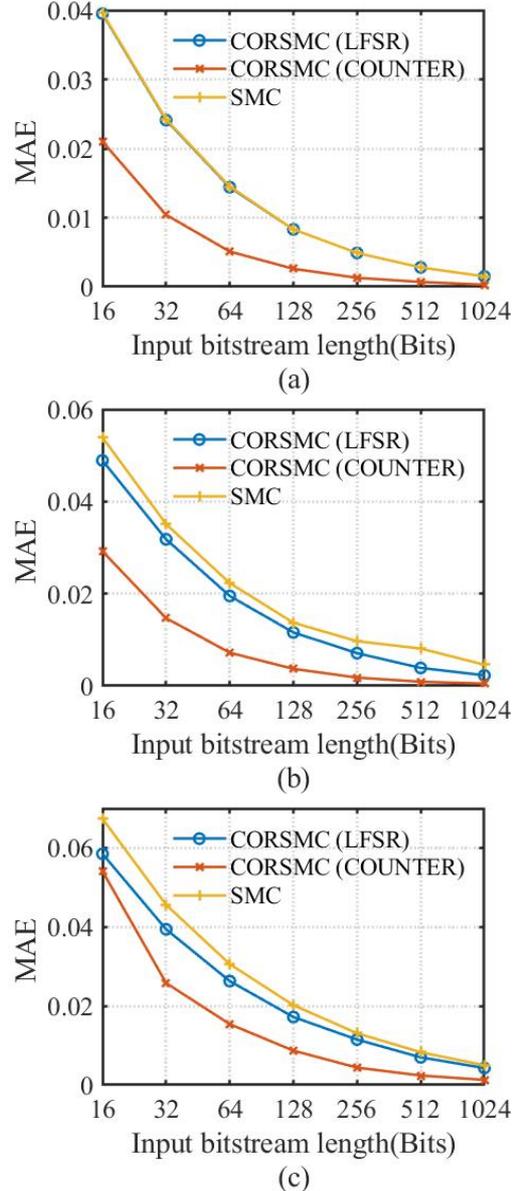


Figure 6. Mean absolute error for (a) 2-to-1 CORSMC (LFSR), 2-to-1 CORSMC (COUNTER), and 2-to-1 SMC. (b) 4-to-1 CORSMC (LFSR), 4-to-1 CORSMC (COUNTER), and 4-to-1 SMC. (c) 9-to-1 CORSMC (LFSR), 9-to-1 CORSMC (COUNTER), and 9-to-1 SMC.

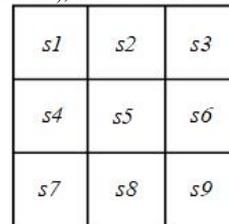


Figure 7. A  $3 \times 3$  window.

and injects salt & pepper noise with a noise density of 0.01. Subsequently, we normalize the pixel values to the range  $[0, 1]$  so as to be processed in the SC domain. Peak Signal to Noise Ratio (PSNR) is the metric used to assess the accuracy. Simulation results under different methods are shown in TABLE I. Fig. 8 shows the original and processed images using different methods.

TABLE I. ACCURACY IN PSNR OF THE 3×3 MEAN FILTER USING DIFFERENT MEAN CIRCUITS

$N=2^k$	$2^4$	$2^5$	$2^6$	$2^7$	$2^8$	$2^9$	$2^{10}$
SMC	27.54	30.16	33.58	37.60	40.61	43.12	46.51
CORSMC (LFSR)	28.85	32.60	36.94	39.18	45.29	50.34	55.58
CORSMC (COUNTER)	29.25	33.75	38.28	44.30	49.34	55.37	59.51

TABLE II. THE AREA, POWER, DELAY, ADP, PDP, AND EDP FOR DIFFERENT MEAN CIRCUITS

	Area ( $\mu\text{m}^2$ )	Power ( $\mu\text{W}$ )	Delay (ns)	ADP ( $\mu\text{m}^2 \times \text{ns}$ )	PDP ( $10^{-3} \text{pJ}$ )	EDP ( $10^{-24} \text{Js}$ )
SMC	23.34	5.83	0.52	12.13	3.03	1.57
CORSMC	20.93	0.43	0.49	9.00	0.21	0.11

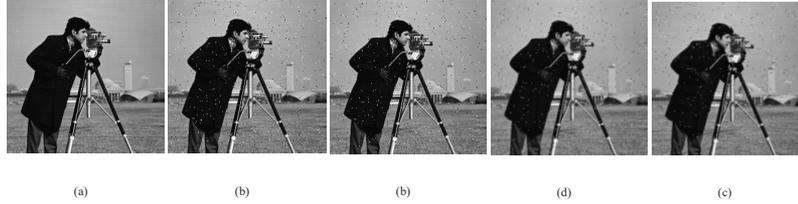


Figure 8. Mean filtering with a 3×3 kernel for sequence length of 256 bits. (a) Original image. (b) Noisy image with salt & pepper noise with density 0.01. (c) Filtered image using MATLAB. (d) Processed image using 9-to-1 SMC and (e) 9-to-1 CORSMC.

### C. Hardware Evaluation

Mean circuits consist of an SNG and a bitstream processing module. TABLE II lists the hardware measurements of the proposed design and the previous design by using the Synopsys design compiler and the TSMC 40 nm technology library. The area, power, delay, area-delay product (ADP), power-delay product (PDP), and energy-delay product (EDP) of 9-to-1 mean circuits are shown. This data just includes the bitstream processing units without SNGs. The CORSMC only needs two random number sources, because it requires only two types of independent bitstreams. The SMC needs at least two random number sources because it requires at least two types of uncorrelated bitstreams. So, the CORSMC requires a smaller number of random number sources than the SMC. Specifically, the experimental results show that the CORSMC reduces 10% in area, 93% in power, 6% in latency, 25% in ADP, 93% in PDP, and 92% in EDP compared to the SMC.

## V. CONCLUSION

In this paper, mean circuits are proposed by utilizing correlated bitstreams. Experimental results show that the performance of the proposed mean circuits is higher than that of the previous design. Meanwhile, the proposed design overcomes the limitation on the number of inputs. Finally, the implementation of a mean filter using the proposed 9-to-1 CORSMC is realized. Future work will focus on the design of mean circuits with a larger number of inputs and other related applications.

## REFERENCES

- [1] B. Gaines, "Stochastic computing," in The Proceedings of the Spring Joint Computer Conference on - AFIPS'67, Atlantic City, New Jersey, 1967, pp. 149-156.
- [2] W. Qian, X. Li, M. Riedel, K. Bazargan, and D. Lilja, "An architecture for fault-tolerant computation with stochastic logic," *IEEE Trans. Comput.*, vol. 60, no. 1, pp. 93-105, Jan. 2011.
- [3] P. Li, D. Lilja, W. Qian, K. Bazargan, and M. Riedel, "Computation on stochastic bit streams digital image processing case studies," *IEEE Trans. Very Large Scale Integr. VLSI Syst.*, vol. 22, no. 3, pp. 449-462, Mar. 2014.
- [4] H. Joe, and Y. Kim, "Novel stochastic computing for energy-efficient image processors," *Electronics*, vol. 8, no. 6, pp. 1-11, Jun. 2019.
- [5] 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, Jul. 2021.
- [6] Y. Zhang, R. Wang, X. Zhang, Y. Wang, and R. Huang, "Parallel hybrid stochastic-binary-based neural network accelerators," *IEEE Trans. Circuits Syst. II Express Briefs*, vol. 67, no. 12, pp. 3387-3391, Dec. 2020.
- [7] H. Ichihara, T. Sugino, S. Ishii, T. Iwagaki, and T. Inoue, "Compact and accurate digital filters based on stochastic computing," *IEEE Trans. Emerging Top. Comput.*, vol. 7, no. 1, pp. 31-43, Mar. 2019.
- [8] Y. Liu, and K. K. Parhi, "Linear-phase lattice FIR digital filter architectures using stochastic logic," *J. Signal Process. Syst. Signal Image Video Technol.*, vol. 90, no. 5, pp. 791-803, May. 2018.
- [9] A. Alaghi, and J. Hayes, "Exploiting correlation in stochastic circuit design," in the 2013 IEEE 31st International Conference on Computer Design (ICCD), Asheville, NC, USA, 2013, pp. 39-46.
- [10] Y. Sakamoto, and S. Yamashita, "Efficient methods to generate constant sns with considering trade-off between error and overhead and its evaluation," *IEICE Trans. Inf. Syst.*, vol. E103d, no. 2, pp. 321-328, Feb. 2020.
- [11] P. Ting, and J. Hayes, "Isolation-based decorrelation of stochastic circuits," in 2016 IEEE 34th International Conference on Computer Design (ICCD), Scottsdale, AZ, USA, 2016, pp. 88-95.
- [12] R. Budhwani, R. Ragavan, and O. Sentieys, "Taking advantage of correlation in stochastic computing," in the 2017 IEEE International Symposium on Circuits and Systems (ISCAS), Baltimore, MD, USA, 2017, pp. 1-4.
- [13] D. Wu, R. Yin, and J. Miguel, "In-stream correlation-based division and bit-inserting square root in stochastic computing," *IEEE Des. Test*, vol. 38, no. 6, pp. 53-59, Dec. 2021.
- [14] M. Ranjbar, M. E. Salehi, and M. H. Najafi, "Using stochastic architectures for edge detection algorithms," in 2015 23rd Iranian Conference on Electrical Engineering, Tehran, Iran, 2015, pp. 723-728.
- [15] M. Najafi, and M. Salehi, "A fast fault-tolerant architecture for Sauvola local image thresholding algorithm using stochastic computing," *IEEE Trans. Very Large Scale Integr. VLSI Syst.*, vol. 24, no. 2, pp. 808-812, Feb. 2016.
- [16] I. Dalal, D. Stefan, and J. Harwayne-Gidansky, "Low discrepancy sequences for monte carlo simulations on reconfigurable platforms," in 2008 International Conference on Application-Specific Systems, Architectures and Processors, Leuven, Belgium, 2008, pp. 108-113.
- [17] S. Liu, and J. Han, "Toward energy-efficient stochastic circuits using parallel Sobol sequences," *IEEE Trans. Very Large Scale Integr. VLSI Syst.*, vol. 26, no. 7, pp. 1326-1339, Jul. 2018.