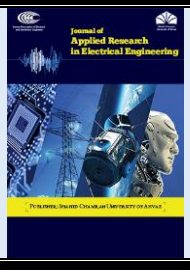
Shahid Chamran University of Ahvaz
Iranian Association of Electrical and Electronics Engineers

Journal of Applied Research in Electrical Engineering

E-ISSN: 2783-2864
P-ISSN: 2717-414X
Homepage: http://jaree.scu.ac.ir/



Research Article

Improving Stochastic Computing Fault-Tolerance: A Case Study on Discrete Wavelet Transform

Shabnam Sadeghi*, and Ali Mahani

Department of Electrical Engineering Shahid Bahonar University of Kerman, Kerman 76169133, Iran

** Corresponding Author: sadeghi.shabnam@eng.uk.ac.ir*

Abstract: The stochastic computing (SC) method is a low-cost alternative to conventional binary computing that processes digital data in the form of pseudo-random bit-streams in which bit-flip errors have a trivial effect on the signal final value because of the highly redundant encoding format of this method. As a result, this computational method is used for fault-tolerant digital applications. In this paper, stochastic computing has been chosen to implement 2-dimensional discrete wavelet transform (2-D DWT) as a case study. The performance of the circuit is analyzed through two different faulty experiments. The results show that stochastic 2-D DWT outperforms binary implementation. Although SC provides inherent fault tolerance, we have proposed four structures based on dual modular redundancy to improve SC reliability. Improving the reliability of the stochastic circuits with the least area overhead is considered the main objective in these structures. The proposed methods are applied to improve the reliability of stochastic wavelet transform circuits. Experimental results show that all proposed structures improve the reliability of stochastic circuits, especially in extremely noisy conditions where fault tolerance of SC is reduced.

Keywords: Stochastic computing, fault-tolerant computation, image processing, discrete wavelet transform.

Article history
Received 28 September 2021; Revised 19 January 2022; Accepted 25 February 2022; Published online 10 December 2022.

How to cite this article
S. Sadeghi, and A. Mahani, "Improving stochastic computing fault-tolerance: A case study on discrete wavelet transform," *J. Appl. Res. Electr. Eng.*, vol. 2, no. 1, pp. 11-18, 2023. DOI: 10.22055/jaree.2022.38634.1036

1. INTRODUCTION

When the size of electronic components shrinks to nanoscale dimensions, the number of expected errors in a circuit increase, and since most error tolerance methods in binary systems increase cost, power consumption, and area, modern computing methods that directly address reliability issues should be considered an alternative to conventional computing. Recently, stochastic computing (SC) has gained attention due to its fault-tolerant capacity and less area requirement as these features are very much attractive for nano-scale CMOS technologies [1]. Therefore, the SC method is used in this paper as a fault-tolerant method.

In SC, a number x is encoded by a random bit stream of 0s and 1s with equal weight for every bit. Irrespective of the length, the ratio of the number of 1s to the length of the bit-stream, i.e., P (X = 1), determines the data value. Error tolerance in stochastic circuits is based on the fact that the occurrence of a single bit error in a bit-stream of length N changes the value of stochastic number (SN) by 1/N because all bits in a stream have the same weight. The larger the bit-stream length is, the more insignificant and smaller the change will be. For example, consider bit stream 00101010 containing three 1s denotes x=p(X=1) =3/8, a single bit-flip changes its value from 3/8 to 4/8 or 2/8, which are the representable numbers closest to the correct result. But, if we consider the same number 3/8 in conventional binary format 0.011, a single bit-flip causes a huge error if it affects the most significant bit. A change from 0.011 to 0.111, for example, changes the result from 3/8 to 7/8.

Additionally, multiple bidirectional errors (i.e., one-to-zero and zero-to-one conversion errors) may even cancel each other out, while the occurrence of errors in binary numbers changes the signal value according to the faulty bit weight [2].

In addition, arithmetic units of SC are very simple, so they have very low power requirements. For instance, the multiplication of two N-bit SNs A and B to form the arithmetic product $A \times B$ can be performed using a single AND gate in

N clock cycles, so SC permits complex computations to be realized using low-cost units in terms of hardware complexity. Fig. 1 shows an example of multiplying two input values, 1/4 and 3/4.

SC provides an accuracy-energy trade-off. Both the accuracy and the energy consumption of the circuit increase with the length N of the stochastic numbers [3], which introduces one of the major drawbacks of SC, i.e., long computing time. While a long stochastic stream may introduce long computational latency, parallel computations can be massively performed [4]. Another new solution to reduce the response time is the SC implementation scheme based on a memristive system [5].

Due to its unique features, the SC method is used to implement randomized algorithms and applications that require large amounts of data. Since small fluctuations are tolerable in such applications but many errors are ruinous, they are suitable to be implemented by stochastic logic [6].

Since image processing operations face severe design limitations in terms of power and area and they do not require high precision [7], several circuit designs have been proposed for different image processing applications in [6,8], including edge detection and gamma correction, which shows that stochastic designs can be significantly smaller, more power-efficient, and noise-tolerant.

One of the most important applications of SC is the implementation of LDPC decoders that mainly require a large number of parallel, fast, and relatively simple operations. In [9], SC has been used to implement an LDPC decoder in that low power requirements, error tolerance, and probabilistic aspects of SC have been exploited to achieve high power efficiency and throughput.

Another area in which SC has been applied with significant progress is artificial neural networks. Due to the resemblance of spike sequences in a biological neural network to stochastic numbers, implementing artificial neural networks continues to be a major application of SC [10]. In [11], authors have shown that stochastic neural networks (SC NN) achieve better area overhead and power consumption than state-of-the-art works by slightly sacrificing accuracy. With recent improvements in SC, the result of SC NNs have become comparable with conventional NNs.

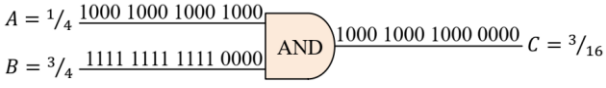
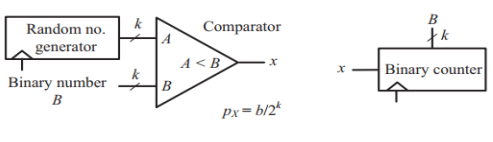


Fig. 1: An example of multiplication using stochastic logic.



(a) (b)

Fig. 2: (a) The binary-to-stochastic and b) stochastic-to-binary converters.

Since hardware implementation of low-cost and fault-tolerant architecture has recently received tremendous attention in modern IC applications, we utilized SC as a fault-tolerant method to implement 2-dimensional discrete wavelet transform (2-D DWT) and improved its reliability by four proposed methods based on error correction and dual modular redundancy.

The remainder of this paper is organized as follows: A brief overview of SC is given in Section 2. The architecture to implement stochastic 2-d wavelet transform is presented in Section 3. Section 4 provides proposed methods to enhance the reliability of stochastic circuits. The proposed methods are implemented and their performance is compared in Section 5, and finally, some conclusions are given in Section 6.

2. STOCHASTIC COMPUTING

In the SC method, operations occur on bit-streams that are interpreted as probabilities. The value of each bit-stream in the unipolar coding format is expressed as the probability of seeing a 1 along the bit-stream. For example, an N-bit stochastic number (SN) X containing N_1 1s and N_0 0s has the value $x=p(X=1) = N_1/N \epsilon [0,1]$. Since the probability of seeing a 1 is a value between 0 and 1, this encoding format is used to display unsigned numbers. In addition to the unipolar format, several alternative SN formats have been proposed in [7] one of which is the bipolar encoding format that deals with the positive and negative numbers in the range [-1,1]. In the scenario of bipolar coding format, the relationship between x and P(X = 1) becomes P(X = 1)= (x+ 1)/2, which enables the stochastic representation for negative numbers and the stochastic value is defined as P(X=1)= (N_1-N_0)/N. Notice that for either unipolar or bipolar coding format, the represented number ranges in [0, 1] or [-1,1]. To represent a number beyond this range, a pre-scaling operation or integer bit-stream-based representation [12] can be used to overcome this limitation. Stochastic circuits consist of three main parts: the stochastic number generator (SNG) of the kind shown in Fig. 2a, which produces a stochastic number. It consists of a pseudo-random number source such as a linear feedback shift register (LFSR) and a magnitude comparator and converts an unsigned k-bit binary number B to an N-bit stochastic bit-stream X. The comparator produces a 1 if the random number is less than B and otherwise a 0. The central unit contains the conventional logic gates and processes the bit-streams, and the last unit converts the output bit-stream to binary values. Fig. 2b shows the structure of the stochastic-to-binary (S2B) converter [7].

Figure 3 illustrates some basic computing units of SC, including multiplication, normalized subtraction, and addition. For instance, multiplication can be performed with an XNOR gate in bipolar coding format since c= 2P(C = 1)– 1 = 2(P(A = 1) P(B= 1) + P(A = 0)P(B = 0))– 1 = (2P(A =1)– 1)(2P(B = 1)– 1) = ab. Another example is addition, which can be simply implemented with a multiplexer in the SC method for c = (C = 1) = 1/2((A= 1) + 1/2(B =1) = 1/2(a + b). Additionally, the addition in the bipolar form uses this multiplexer as well since c = 2P(C = 1)– 1 = 2(1/2(P(A= 1) + 1/2P(B = 1))– 1 = 1/2(2P(A = 1)– 1) + (2P(B = 1)– 1)) = 1/2(a + b), and finally subtraction is easily implemented by

combining a multiplexer and a not gate. Normalization in addition and subtraction aims to make sure that the result is in the range [0,1] or [-1,1] so that it can be treated as a probability [6].

Besides the basic operations of addition and multiplication, the stochastic mean circuit and absolute value subtraction are presented in [13]. SC has also been applied to division [14] and some trigonometric and polynomial arithmetic functions [15, 16].

3. Proposed Stochastic Circuit Design

3.1. Wavelet Transform Overview

Due to the wide application of wavelet transform and its computational complexity, the study of VLSI implementation of discrete wavelet transform (DWT) has become significant and unavoidable. A small and straightforward architecture will be advantageous, especially in image processing applications. Furthermore, since error and noise rates are high in most image processing applications, error resistance is also essential for these structures. For example, a fault-tolerance method is discussed in [17] to deal with silent data corruption errors on DWT. Therefore, having the advantages of SC, an error-resistant low-cost design is presented for two-dimensional DWT.

Wavelet transform is a mathematical tool that can decompose signals into different sub-bands of well-defined time-frequency characteristics. This conversion uses various methods to analyze the signal and adjust the accuracy in both time and frequency domains [18]. These benefits lead to extensive use of DWT in different areas such as medicine for processing medical images and diagnosing disorders using a computer [19], data transmission through the internet [20], and noise detection in data collected with a sensor [21].

Compared with the standards JPEG and JPEG-LS, the JPEG2000 standard not only offers a superior image compression ratio but also benefits from better image reconstruction performance [18]. One of the most widely used types of DWT is the 5/3 method used in the JPEG2000 standard to implement lossless image compression. The 5/3 wavelet transform can be implemented by using mathematical notations as follows [22]:

$$H(n) = -\frac{1}{2}[X(2n) + X(2n+2)] + X(2n+1) \quad (1)$$

$$L(n) = \frac{1}{4}[H(n-1) + H(n)] + X(2n) \quad (2)$$

where *H(n)* and *L(n)* represent the high-frequency (detail coefficients) and low-frequency (approximation coefficients) components of the input signal, respectively and *X(n)* represents the *n*th input sample.

The data flow related to these equations is shown in Fig. 4. To implement them in binary mode, we need a structure similar to Fig. 5. As shown in Fig. 5, this structure requires four adders and two multipliers for each decomposition level.

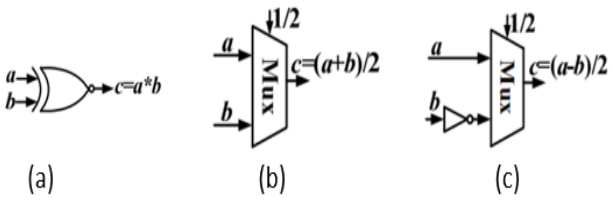


Fig. 3: a) The multiplication of bipolar format, b) scaled addition, b) scaled subtraction of bipolar format.

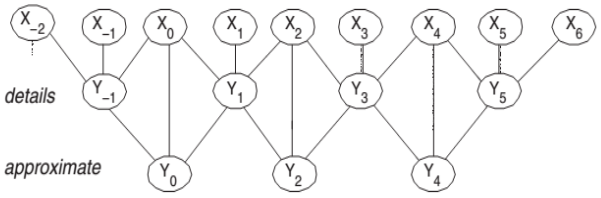


Fig. 4: The data flow of 5/3 wavelet transform.

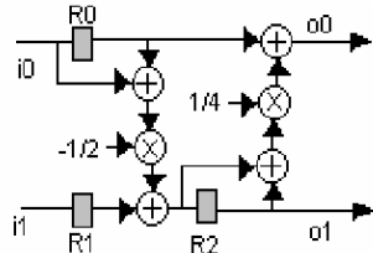


Fig. 5: The one-dimensional structure of binary DWT.

3.2. Stochastic Architecture for the Lifting Structure

As mentioned earlier, operations occur on bit-steams that are interpreted as probabilities in SC. Since we are dealing with positive and negative numbers in wavelet transform, the bipolar coding format is used in this article.

The blocks used in the lifting-based DWT include multipliers and adders, which are easily implemented in the BP stochastic method using an XNOR gate and a two-way multiplexer. Because the coefficients in the 5/3 wavelet transform equations are all multiples of 0.5, they are implemented using the coefficient of 0.5 in the stochastic adder, and there is no need to use a stochastic multiplier (XNOR gate). Since stochastic addition and subtraction are normalized, the implementation of equations (1) and (2) by stochastic logic leads to equations (3) and (4).

$$\mathrm{H'(n)} = \frac{-\frac{1}{2}[\mathrm{X(2n)} + \mathrm{X(2n+2)}] + \mathrm{X(2n+1)}}{2} = \frac{\mathrm{H(n)}}{2} \quad (3)$$

$$\mathrm{L'(n)} = \frac{\frac{1}{2}[\mathrm{H'(n-1)} + \mathrm{H'(n)}] + \mathrm{X(2n)}}{2} = \frac{\frac{1}{2}\left[\frac{\mathrm{H(n-1)}}{2} + \frac{\mathrm{H(n)}}{2}\right] + \mathrm{X(2n)}}{2} = \frac{\frac{1}{4}[\mathrm{H(n-1)} + \mathrm{H(n)}] + \mathrm{X(2n)}}{2} = \frac{\mathrm{L(n)}}{2} \quad (4)$$

By implementing equations (3) and (4) with stochastic addition blocks, the structure of Fig. 6 is obtained for a one-dimensional 5/3 wavelet transform.

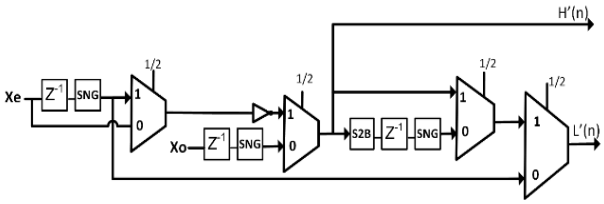


Fig. 6: The stochastic one-dimensional 5/3 wavelet transform structure.

In the proposed one-dimensional architecture, input signals pass through the delay elements (Z^{-1}) in 2's complement format and then each signal from the delay element is converted to a stochastic bit-stream. Each delay requires an 8-bit shift register instead of using a 256-bit delay element used in [23]. Also, for the intermediate delay element, an 8-bit memory element is used so the mux2 output is converted from stochastic to binary format before being stored in the delay element and then back to stochastic again. Since stochastic addition is correlation-insensitive [24] and only one LFSR has been used for these SNGs to reduce the hardware cost. In each clock cycle, one bit of high-frequency output coefficients and one bit of low-frequency output coefficients are generated. As it can be seen in Fig. 6, instead of using complex binary multiplier and adder blocks, only a multiplexer is used in the SC design.

To run wavelet, transform on the images, it is necessary to apply the 1-D DWT in both vertical and horizontal directions of the image. The proposed 2-D architecture is based on three 1-D DWT structures, which operate in parallel and communicate through shared memory. The input image is fed to the architecture bit-by-bit using row-by-row scanning. In each clock cycle, a single bit is fed. In the row module, 1-D DWT of each row is computed to yield the low- and high-frequency components of each row, and the results are stored in memory to utilize in column transform.

To reduce the memory required, the row transform results are converted to binary format before being stored in memory. We employ shift registers for intermediate data storage. Low-frequency outputs and high-frequency outputs are stored in two distinct register files. Since three data are required to compute equation 3, the column transform process starts as soon as the first data in the third row of memory is available.

Simultaneously with the column transform, the rest of the data required for the column operation are prepared by the row transform and overwritten in the memory. So, the capacity of the register file is considered 3*N/2 coefficients for the N x N image.

The column operation begins by converting the stored binary numbers to stochastic ones, which can also eliminate autocorrelation [24] and improve the accuracy of stochastic computation, then performing a step of computation on these three rows in the column direction.

Two column transform modules are also employed and work in parallel to increase the design speed. One module performs the column direction wavelet transform on high-frequency coefficients of the row transform module and the other on low-frequency coefficients. Fig. 7 illustrates the stochastic 2-D 5/3 wavelet transform structure. The section marked with a dotted line in Fig. 7 shows the structure of a 1-D DWT. The other two modules perform the column transforms. Although this architecture is presented for the 5/3 case, the method can be applied to all lifting schemes that rely on single or double lifting steps.

4. Methods To Improve Fault Tolerance Of Stochastic Circuits

Although the SC method is inherently fault-tolerant, so far no action has been taken to improve its reliability and its error tolerance. Therefore, in this paper, four methods based on dual modular redundancy (DMR) are proposed to improve the reliability of stochastic circuits. Furthermore, since one of the significant advantages of SC is the small size of the stochastic circuits, in order to maintain this feature, the goal has been set to improve reliability with minimum area overhead in all proposed methods.

As the most important computational part of stochastic wavelet transform is multiplexers, these methods are only applied to multiplexers so that improved reliability does not lead to much hardware overhead. These methods are explained in the next subsections.

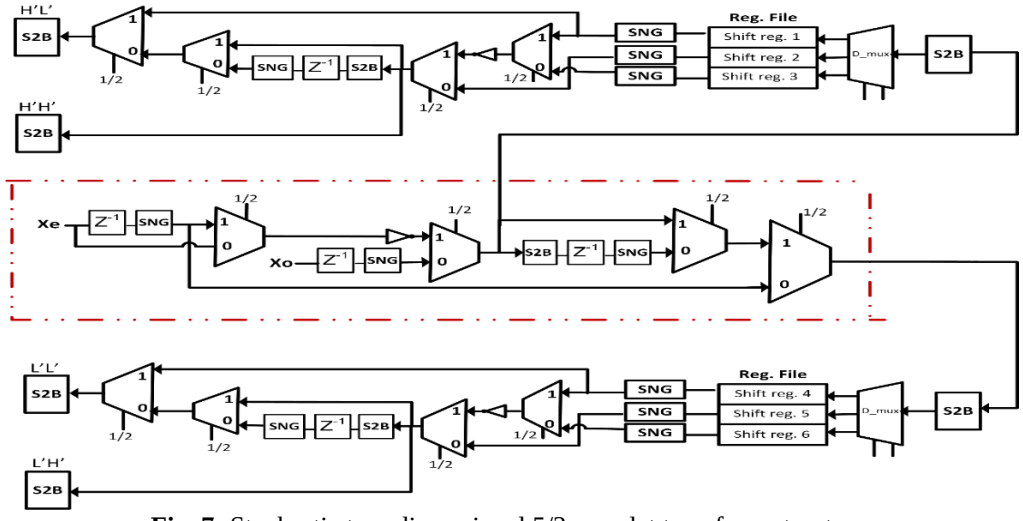


Fig. 7: Stochastic two-dimensional 5/3 wavelet transform structure.

4.1. Multiplexer-Based Error Correction

In the first method, DMR and multiplexer-based error correction structures are proposed to increase the reliability of stochastic circuits. Fig. 8 shows the structure of the error detection and correction circuit for a multiplexer. This multiplexer has a selection input with a constant value of 0.5 in which case 50% of the mux1 output and 50% of the mux2 output are randomly transferred to the final output**.**

4.2. C-Element Based Error Correction

The second method proposed to improve the reliability of stochastic circuits is based on DMR and a C-element. In this case, when the two main modules have an equally logical value, the same value is placed on the output, but if the two values are different, the output retains its previous value. The proposed circuit structure is shown in Fig. 9.

4.3. Error Correction Method based on Repeating the Operation for Faulty Bits

In this method, the multiplexers are placed in the form of DMR and an XOR gate is used for error detection. The clock signal in this circuit is blocked by an AND gate. When there is no error in the circuit, the output of the two multiplexers is equal and as a result, the XOR gate has a value of 0, which causes the next edge of the clock signal to be seen by the system and the operation to be performed on the next bit, but when an error occurs on one of the multiplexers, the output of the XOR gate becomes 1, the edge of the clock signal is not passed, and the operation is repeated on the same bit until the system has no error. Fig. 10 shows a part of the wavelet transform circuit optimized using this method.

Since the critical path in SC circuits is short, which means that it can be run with extremely high clock frequency [7], this method will be more efficient in stochastic circuits than in binary ones and the overall latency can be greatly mitigated by exploiting a special property of SC, which is called progressive precision [8] or by using parallel computational elements.

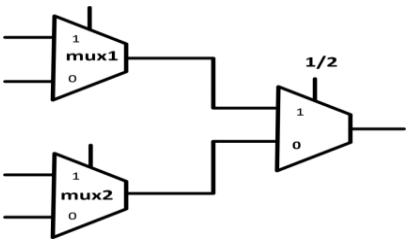


Fig. 8: The mux_based error correction circuit structure.

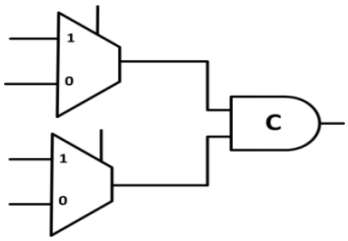


Fig. 9: The C_element-based error correction circuit structure.

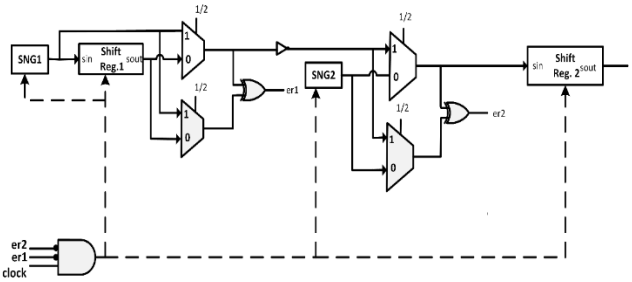


Fig. 10: The partial structure of a wavelet transform optimized by the fourth method.

4.4. Error Correction Method based on Stochastic Value of the Signal

As mentioned in Section 2, the value of each bit-stream in SC is defined as the number of 1s in the bit-stream divided by the total number of bits. In other words, the probability of occurrence of a logical one in the bit-stream indicates the stochastic number. Using this definition, the probability of the output bit of each module is clear. For example, the probability of output of the stochastic adder being one is obtained $Py = \frac{1}{2}(Px_1 + Px_2)$ and the probability of being zero is $1 - Py$. We suggest an algorithm to correct the bit-flip errors based on this SC feature. Consider an error occurs in the adder block. After the error detection to correct the error, that bit becomes one with the probability of Py and zero with the probability of $1 - Py$.

5. Experimental Results

In this section, the results of implementing the proposed stochastic circuit designs are presented. These results include the estimated hardware cost and analyzed performance under two different faulty situations.

The classic Lena image was used as the input source of the wavelet transform circuit. The input is a 256 × 256-pixel grey-scale image. Each pixel is represented by an 8-bit binary number. The length of the stochastic bit-stream is 256, which corresponds to 8-bit precision for conventional binary design.

The wavelet transform circuit presented in Section 3 was coded in VHDL and synthesized with Xilinx ISE on Virtex5 (XC5VLX110T) FPGA device. The results obtained in Table 1 were compared with the binary wavelet transform circuits [25-27].

As shown in the previous section, in the design of stochastic DWT, a multiplexer is only used instead of using complex multipliers and adder blocks, and as Table 1 shows, the proposed 5/3 stochastic DWT has a lower area than the other existing conventional binary architectures. Since new computing methods must be able to meet severe constraints such as very small size and low power consumption, SC can be a suitable alternative to conventional binary computing to design circuits that require fault-tolerant procedures on large amounts of data (e.g., various image processing operations).

Table 1: Area comparison.

architecture	Device	Slice LUTs	Slice registers
[25]	XC5VLX110T	494	633
[26]	XC5VLX110T	9424	301
[27]	6VLX760FF1760-2	361/433	411/511
stochastic	XC5VLX110T	215	408

Table 2: The comparison of fault tolerance.

Method	average error					
	Noise at input			Noise at computational blocks		
	5%	10%	15%	5%	10%	15%
SC method	0.030	0.038	0.043	0.082	0.139	0.174
Conventional method	0.329	0.353	0.390	0.315	0.350	0.397

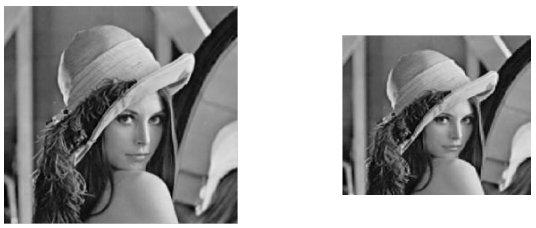
Fig. 11 is an image obtained from a stochastic 2-D DWT applied to the Lena image, which shows approximation coefficients for one decomposition level. The result of the wavelet transform processing by stochastic structure has some errors mainly due to the random fluctuations of SC [24] and the output MSE will be 0.0031. Since digital circuits are easily affected by manufacturing defects and transient errors, this inherent error is negligible compared to the high error rate in digital circuits.

To demonstrate the fault tolerance of the proposed architecture, we performed two kinds of experiments. In the first one, the inputs contain noise and in the second one, the circuit computational components are considered noisy and unreliable. Experiments were performed by injecting soft errors including flipping the bits by different injected noise ratios and evaluating the output average error in pixel values. Table 2 shows these experimental results for conventional implementations compared to stochastic 2-D DWT.

As can be seen in Table 2, the stochastic method shows high resistance to bit-flip errors in both faulty experiments and gives acceptable results even in the case of 15% noise, but the performance of the binary method decreases dramatically with increasing error rate. Also, the average error drop rates for these experiments in the stochastic method are less than the binary one while the error rate is increasing from 5% to 15%.

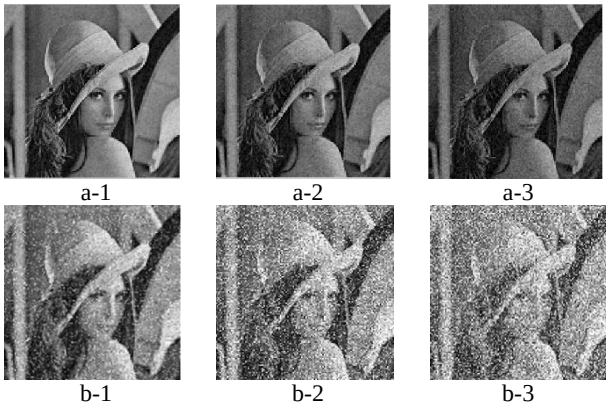
Fig. 12 illustrates the output image of the wavelet transform implemented by the stochastic method and the binary method with different noise ratios. As can be seen in Fig. 12 when errors are injected at the rate of 15%, the image transformed by stochastic DWT is still recognizable and becomes greyer, while the image generated by the conventional method is full of noisy pixels.

In the following, the hardware area overhead and latency overhead are investigated for proposed error correction methods using stochastic DWT and to evaluate the robustness of our methods we analyze the effect of randomly injected errors by measuring the corresponding average output error for each implementation. Area overhead and latency overhead are given in Table 3. As in the third proposed method, the delay overhead changes according to the error rate, so the delay for this method is reported for each error rate in Table 3. The latency overhead is calculated for operations on 256 bits.



(a) (b)

Fig. 11: The result of applying stochastic wavelet transform to Lena image, (a) Original image, (b) Approximate coefficients of stochastic DWT.



a-1 a-2 a-3

b-1 b-2 b-3

Fig. 12: The output of (a) stochastic and (b) conventional DWT with noise level of (1) 5%, (2) 10%, and (3)15%.

Table 3: Area and latency overheads of error correction methods to stochastic DWT circuit.

Method	Area overhead	Latency overhead
multiplexer-based error correction	8.14	34.50
C-element based error correction	21.46	55.71
Error correction method based on repeating the operation	15.59	Noise level 5% 10% 15% 25.75 31.85 37.95

Table 4 summarizes the performance of the proposed error correction methods at various noise levels of the inputs. As shown in Table 4, all the proposed methods improve the reliability of stochastic DWT according to average output error. In these methods, the average error is significantly improved and their efficiency increases with increasing error rate. In the multiplexer-based error correcting method, the least overhead (only 8%) is applied to the circuit and has acceptable error correction performance. The use of C_element to improve the reliability of stochastic circuits will improve average error so that it is more effective than the previous method but it has more latency and area overheads.

By the error correction method, which is based on repeating the operation, with an area overhead of 13.6%, errors are completely corrected. The latency overhead of this method varies according to the error rate, but the correction in the other methods is performed with constant overhead. Since the clock period of stochastic circuits is small, this method will be more efficient in stochastic designs than in conventional architectures. In addition, because of stochastic circuits' simplicity, parallel processing approaches and some methods based on the progressive precision property [28, 8] can be used to further reduce the latency.

Table 4: Fault tolerance comparison of error correction methods.

Method	Improved average error (%)					
	Noise at input			Noise at computational blocks		
	5%	10%	15%	5%	10%	15%
Multiplexer-based error correction	36.94	41.86	49.06	39.59	46.69	55.03
C-element based error correction	39.28	48.9	57.21	46.10	54.15	63.8
Error correction method based on stochastic value of the signal	53.25	58.73	66.19	60.78	66.09	71.99
Error correction method based on repeating the operation	100	100	100	100	100	100

The error correction method based on the stochastic value of the signal improves reliability better than other methods because it performs the correction according to the actual value of the signal, but the other methods work randomly. Similar to other methods, the efficiency of this method is improved by increasing the error rate, which makes these methods more suitable for environments with very high noise levels where the inherent fault tolerance of SC is reduced.

6. Consultation

This study presents a fault-tolerant and low-area architecture for 2-D lifting-based DWT based on SC. The proposed architecture not only is much more tolerant of soft errors but also requires less area than the conventional implementation of this algorithm. To make the SC designs more robust to soft errors, we introduced four error correction methods based on DMR. Our experimental results show that the proposed methods had low hardware costs and all the proposed methods improved the reliability of the stochastic circuits according to average error. The remarkable note about all the proposed methods is that in all the methods, the performance of the proposed methods improves by increasing the error rate, which in turn makes these methods suitable for highly noisy environments where the inherent fault tolerance of SC is reduced. Future work will focus on how to generate low-cost probabilities to use in the error correction method based on the stochastic value of the signal.

Credit Authorship Contribution Statement

Shabnam Sadeghi: Conceptualization, Data curation, Formal analysis, Investigation, Methodology, Resources, Software, Validation, Visualization, Roles/Writing - original draft, Writing - review & editing. **Ali Mahani**: Conceptualization, Funding acquisition, Project administration, Supervision, Validation.

Declaration Of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper. The ethical issues; including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, redundancy has been completely observed by the authors.

References

[1] V. V. Mahesh, and T. K. Shahana, "Design and synthesis of FIR filter banks using area and power efficient Stochastic Computing," in *2020 Fourth World Conference on Smart Trends in Systems, Security and Sustainability*, 2020, pp. 662-666.

[2] T.-H. Chen, A. Alaghi, and J. P. Hayes, "Behavior of stochastic circuits under severe error conditions," *it-Information Technology*, vol. 56, pp. 182-191, 2014.

[3] A. Alaghi, W. T. J. Chan, J. P. Hayes, A. B. Kahng, and J. Li, "Optimizing stochastic circuits for accuracy-energy tradeoffs," in 2015 IEEE/ACM International Conference on *Computer-Aided Design*, 2015, pp. 178-185.

[4] K. Papachatzopoulos, C. Andriakopoulos, and V. Paliouras, "Novel Noise-Shaping Stochastic-Computing Converters for Digital Filtering," in *2020 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2020, pp. 1-5.

[5] T. Li, S. Duan, J. Liu, and L. Wang, "Memristive combinational logic circuits and stochastic computing implementation scheme," *Circuit World*, 2021.

[6] P. Li, D. J. Lilja, W. Qian, K. Bazargan, and M. D. Riedel, "Computation on stochastic bit-streams digital image processing case studies," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 22, no. 3, pp. 449-462, 2014.

[7] A. Alaghi, W. Qian, and J. P. Hayes, "The promise and challenge of stochastic computing," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 37, no. 8, pp. 1515-1531, 2018.

[8] R. Seva, P. Metku, K. K. Kim, Y. -B. Kim, and M. Choi, "Approximate stochastic computing (ASC) for image processing applications," in *2016 International SoC Design Conference (ISOCC)*, 2016, pp. 31-32.

[9] A. Hussein, M. Elmasry, and V. Gaudet, "On the fault tolerance of stochastic decoders," in *2017 IEEE 47th International Symposium on Multiple-Valued Logic (ISMVL)*, 2017, pp. 219-223.

[10] J. P. Hayes, "Introduction to stochastic computing and its challenges," in *Proceedings of the 52nd Annual Design Automation Conference*, 2015, pp. 59.

[11] Y. Liu, S. Liu, Y. Wang, F. Lombardi, and J. Han, "A survey of stochastic computing neural networks for machine learning applications," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 7, pp. 2809-2824, 2021.

[12] M. Hasani Sadi, and A. Mahani, "Accelerating deep convolutional neural network base on stochastic computing," *Integration*, vol. 76, pp. 113-121, 2021.

[13] M. H. Najafi, and M. E. Salehi, "A fast fault-tolerant architecture for sauvola local image thresholding algorithm using stochastic computing," *IEEE Transactions on Very* Large *Scale Integration (VLSI) Systems*, vol. 24, pp. 808-812, 2016.

[14] T. Chen and J. P. Hayes, "Design of division circuits for stochastic computing," in *2016 IEEE Computer Society Annual Symposium on VLSI*, 2016, pp. 116-121

[15] W. Qian, and M. D. Riedel, "The synthesis of robust polynomial arithmetic with stochastic logic," in *2008 45th ACM/IEEE Design Automation Conference*, 2008, pp. 648-653.

[16] Y. Liu, and K. K. Parhi, "Computing complex functions using factorization in unipolar stochastic logic," in *2016 International Great Lakes Symposium on VLSI*, 2016, pp. 109-112.

[17] Ch. Bao, and Sh. Zhang. "Algorithm-based fault tolerance for discrete wavelet transform implemented on GPUs." *Journal of Systems Architecture*, vol. 108, pp. 101823, 2020.

[18] B.-F. Wu, and C.-F. Lin, "A high-performance and memory-efficient pipeline architecture for the 5/3 and 9/7 discrete wavelet transform of JPEG2000 codec," *IEEE Transactions on Circuits and Systems for Video Technoogy*, vol. 15, pp. 1615-1628, 2005.

[19] R. Starosolski, "Hybrid adaptive lossless image compression based on discrete wavelet transform," *entropy*, vol. 22, no. 7, pp. 751, 2020.

[20] S. H. Farghaly, and S. M. Ismail. "Floating-point discrete wavelet transform-based image compression on FPGA," *AEU-International Journal of Electronics and Communications*, vol. 124, pp. 153363, 2020.

[21] T.-B. Dang, D.-T. Le, M. Kim, and H. Choo, "DWT-PCA combination for noise detection in wireless sensor networks." in *Proc. of the Korea Information Processing Society Conference*, 2020, pp. 144-146.

[22] C. Xiong, J. Tian, and J. Liu, "Efficient architectures for two-dimensional discrete wavelet transform using lifting scheme," *IEEE Transactions on Image Processing*, vol. 16, pp. 607-614, 2007.

[23] S. A. Salehi, and D. D. Dhruba, "Efficient Hardware Implementation of Discrete Wavelet Transform Based on Stochastic Computing," in *2020 IEEE Computer Society Annual Symposium on VLSI*, 2020, pp. 422-427.

[24] A. Alaghi, P. Ting, V. T. Lee, and J. P. Hayes, "Accuracy and correlation in stochastic computing," *Gross W.*, Gaudet *V. (eds) Stochastic Computing: Techniques and Applications*, pp. 77-102, 2019.

[25] A. D. Darji, S .S. Kushwah, S. N. Merchant, and A. N. Chandorkar, "High-performance hardware architectures for multi-level lifting-based discrete wavelet transform," *EURASIP Journal on Image and Video Processing*, vol. 2014, pp. 47, 2014.

[26] S. S. Bhairannawar, S. Sarkar, and K. B. Raja, "Implementation of fingerprint based biometric system using optimized 5/3 DWT architecture and modified CORDIC based FFT," *Circuits, Systems, Signal Process*, vol. 37, no. 1, pp. 342–366, 2018.

[27] MR. Lone, "A high speed architecture for lifting-based 2-D Cohen-Daubechies-Feauveau (5, 3) discrete wavelet transform used in JPEG2000". *International* Journal *of Advances in Telecommunications, Electrotechnics, Signals and Systems*, vol. 24, no. 9, pp. 9-24, 2017.

[28] T. Chen, P. Ting, and J. P. Hayes, "Achieving progressive precision in stochastic computing," in *2017 IEEE Global Conference on Signal and Information Processing (GlobalSIP)*, 2017, pp. 1320-1324.

BIOGRAPHY

Shabnam Sadeghi received her B.Sc. degree in electronic engineering from Shahid Bahonar University, Kerman, Iran in 2013. Since then, she was with the RSS Lab at Shahid Bahonar University for 3 years. Her research interests are fault-tolerant designs, stochastic computing, and data analysis.

Ali Mahani received his B.Sc. degree in Electronic Engineering from Shahid Bahonar University, Kerman, Iran in 2001, and his M.Sc. and Ph.D. degrees both in Electronic Engineering from the Iran University of Science and Technology (IUST), Tehran, Iran in 2003 and 2009, respectively. Since then, he has been with the Electrical Engineering Department of Shahid Bahonar University where he is currently an associate professor. His research interests focus on fault-tolerant designs, FPGA-based accelerators, approximate digital circuits, stochastic computing, and networked systems.