HARDWARE IMPLEMENTATION OF FIR/IIR DIGITAL FILTERS USING INTEGRAL STOCHASTIC COMPUTATION

Arash Ardakani, François Leduc-Primeau and *Warren J. Gross*

Department of Electrical and Computer Engineering
McGill University, Montréal, Québec, Canada

ABSTRACT

Stochastic computing (SC) has received much recent attention due to its inherent fault-tolerance and low implementation cost compared to binary radix representations. SC has been proposed for various signal processing applications such as digital filters. The prior art in stochastic FIR filters can accurately implement the desired filtering function for low-order filters, however, their accuracy degrades as the filter order increases. Moreover, stochastic IIR filters demonstrate high hardware complexity and degraded accuracy. In this paper, we propose an architecture for high-order FIR filters with negligible accuracy loss compared to fixed-point implementation. The proposed architecture requires fewer random number generators. We also describe a novel cascaded second-order direct-form II structure for IIR filters. The implementation results of the proposed design show an improvement in latency and hardware complexity compared to the stochastic architectures reported to date.

Index Terms— Dithered quantization, FIR/IIR filters, stochastic computing, VLSI implementation.

1. INTRODUCTION

In recent years, stochastic computing (SC) has shown promising results for low-cost fault tolerant VLSI implementation for a wide range of applications. Despite its advantages, SC suffers from high processing time and low accuracy. Therefore, SC was viewed as not suitable for applications which require high accuracy such as digital filters. To overcome the aforementioned issue, a lattice structure was proposed for stochastic implementation of digital FIR/IIR filters [1, 2]. However, this approach is restricted to low-order digital filters.

In SC, a real value $x \in [0, 1]$ is represented as a sequence of random bits, $X_i \in \{0, 1\}$, $i \in \{1, 2, \ldots, N\}$, where N denotes the stream length. The number x corresponds to the expected value of an element of the sequence:

$$\mathbb{E}[X_i] = x, \tag{1}$$

where $\mathbb{E}$ denotes the expected value. This stochastic representation is known as the *unipolar* format. The *bipolar* format is also used for stochastic representation of a real number $x \in [-1, 1]$ by setting:

$$\mathbb{E}[X_i] = (x + 1)/2. \tag{2}$$

Many SC operations can be performed with simple circuits. For instance, multiplications can be performed by using the AND gate in unipolar format, and a multiplexer can be used to perform scaled additions. However, the use of scaled adders degrades accuracy, and to overcome this issue, the accumulative parallel counter (APC) was introduced in [3]. The APC uses a binary tree-adder to perform additions without discarding information bits as opposed to the scaled-adder. Moreover, the output of APC is in binary radix domain. Therefore, the APC is restricted to the applications requiring information in binary domain after additions. Note that a binary to stochastic (B2S) converter can be used to convert the binary output of the APC back to the stochastic domain, however, it increases the latency of the stochastic implementation.

To avoid using a B2S in middle of a stochastic circuit, Integral SC was presented in [4] to perform computations on integer stochastic stream while using conventional binary adders to perform additions. Each element S_i of the integer stochastic stream represents a real value $s \in [0, m]$ and can be generated by summing up m binary stochastic streams as follows:

$$S_i = \sum_{j=1}^{m} X_i^j, \tag{3}$$

where X_i^j denotes an element of stochastic stream X^j representing the real value $x^j \in [0, 1]$. Then, the expected value of the sequence element S_i is given by:

$$s = \mathbb{E}[S_i] = \sum_{j=1}^{m} x^j. \tag{4}$$

The integer stochastic stream of a real value s in bipolar format can also be obtained in similar way as follows:

$$S_i = 2 \times \sum_{j=1}^{m} X_i^j - m, \tag{5}$$

and the value represented by the stream is

$$s = \mathbb{E}[S_i] = 2 \times \sum_{j=1}^{m} \mathbb{E}[X_i^j] - m = 2 \times \sum_{j=1}^{m} x^j - m. \tag{6}$$

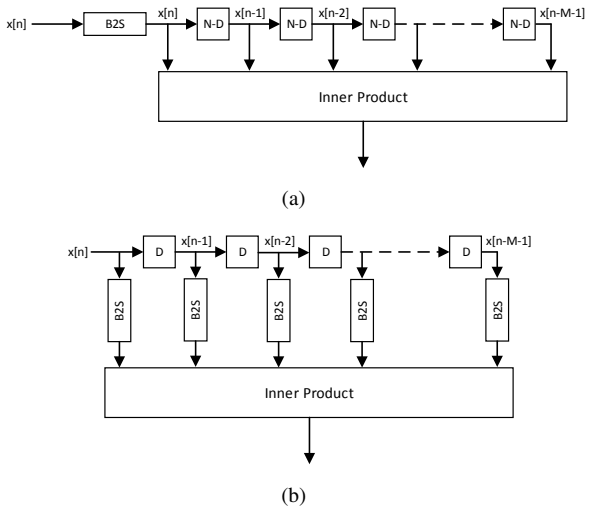

Fig. 1. Two conventional approaches used for stochastic implementation of FIR filters by delaying (a) the stochastic sequence and (b) the binary numbers.

This paper firstly proposes a hardware implementation of a high-order FIR filter using the APC. The proposed architecture requires only two LFSR units independent of filter order. Then, we propose a stochastic implementation of a second order direct-form II structure of an IIR filter based on Integral SC. Therefore, a high-order IIR filter can be achieved by cascading some second-order direct-form II structures.

2. STOCHASTIC IMPLEMENTATION OF FIR FILTERS

A general M-tap FIR filter is formulated as follows:

$$y[n] = b_0 \cdot x[n] + b_1 \cdot x[n-1] + \cdots + b_{M-1} \cdot x[n-M+1]. \quad (7)$$

Previously proposed stochastic implementations of FIR filters normally use two different approaches to generate the delayed version of inputs as shown in Fig. 1. In the first approach, Fig. 1(a), the inputs are converted to the stochastic stream which are then delayed. In contrast, the binary inputs are delayed and then their stochastic streams are generated as illustrated in Fig. 1(b) in the second approach. Therefore, the first approach requires a total of $N \times (M-1)$-bit memory elements in its delay line. The second approach uses more B2S units adopted with different seed values as their initial LFSR value to avoid the correlations among the stochastic sequences. The inner product block denoted in Fig. 1 contains the stochastic elements to perform the additions and multiplications [5]. Moreover, a stochastic lattice implementation of linear-phase FIR filters is presented in [2] to reduce the hardware complexity of traditional stochastic architectures. However, all the aforementioned approaches are restricted to low-order filters while high-order FIR filters are required in many applications. To address this issue, a novel stochastic implementation of FIR filter is proposed in the sequel of this section.

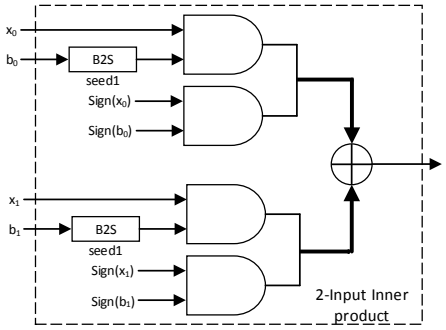

Fig. 2. The proposed inner product architecture.

The additions in SC are traditionally performed by using scaled-adders [6]. This adder consists of a multiplexer in which its selector signal is connected to a stochastic stream with probability of 0.5. The output of the multiplexer is $(\mathbb{E}[A] + \mathbb{E}[B])/2$ where A and B are the inputs of the adder. Therefore, to add a large number of inputs, a large stream length is required for a stochastic implementation of FIR filter to function properly. Note that increasing the stream length results in high latency. To overcome this issue, the APC is proposed in [3], which does not require a random select signal. The APC is a tree-adder followed by an accumulator. Therefore, the APC converts stochastic streams to their binary forms. Our proposed stochastic implementation of a FIR filter uses the high-level architecture illustrated in Fig. 1(b) and the APC to perform additions and convert the stochastic streams to their binary forms as depicted in Fig. 2. Since the high-order filter coefficients are small numbers, the multiplications are performed by using AND gates in unipolar format, which results in more accurate results compared to bipolar multiplications. Moreover, the additions are correlation-free and multiplications only require different seed values for each of its inputs in this architecture. Therefore, two different seed values are required for the whole design independent of filter order.

3. STOCHASTIC IMPLEMENTATION OF IIR FILTERS

A transfer function of a M-tap IIR filter is represented by:

$$H(z) = \frac{b_0 + b_1 \cdot z^{-1} + \cdots + b_{M-1} \cdot z^{-(M-1)}}{1 + a_1 \cdot z^{-1} + \cdots + a_{M-1} \cdot z^{-(M-1)}}. \quad (8)$$

The practical way to implement a M-tap IIR filer is to use direct-form II structure, which only requires M delay units. However, direct-form II structure has a disadvantage. In [7], it is shown that as the quality factor increases, the round-off noise of this structure increases without bound. To overcome this problem, high-order IIR filter are realized as a cascaded series of second-order direct-form II structure. The general

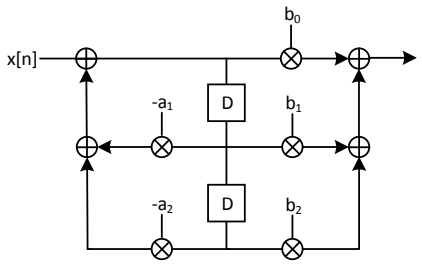

Fig. 3. The direct-form II structure for a second-order IIR filter.

transfer function of this structure is given by:

$$H(z) = \frac{b_0 + b_1 \cdot z^{-1} + b_2 \cdot z^{-2}}{1 + a_1 \cdot z^{-1} + a_2 \cdot z^{-2}}. \quad (9)$$

Fig. 3 shows a general second-order direct-form II structure for a hardware realization. Assuming that the input and coefficients lie in $[-1, 1]$, the output and internal values may be out of this range. Therefore, all the coefficients are first scaled down to prevent the occurrence of the aforementioned issue in the previously proposed stochastic implementation of a direct-form I structure of an IIR filter, leading to the huge performance loss in terms of error-to-signal power ratio [5]. Moreover, cascaded series of second-order direct-form I structure also result in high latency since the output of each stage is converted back into the binary domain [5]. In [1], a stochastic implementation of an IIR filter using the lattice structure is proposed to improve the performance. However, this architecture uses several binary multipliers which potentially increase the hardware complexity. Therefore, an efficient stochastic architecture for IIR filter is still missing.

In this paper, the Integral SC method is used for stochastic implementation of the cascaded series of second-order direct-form II structure of an IIR filter as depicted in Fig. 4. The motivation for using this method is the fact that internal values and output of each stage lie to the range out of $[-1, 1]$ interval. Therefore, computations are performed by using Integral SC computational elements introduced in [4]. In [4], it was shown that additions are performed by using binary adders, and multiplications can be performed by binary multipliers. The second-order direct-form II structure coefficient values mostly lie in $[-2, 2]$ and multiplications are then performed using a multiplexer as shown in Fig. 5. For those values which lie in $[-1, 1]$, multiplications can be performed by AND gate similar to conventional SC [4]. Note that if the coefficient values are out of $[-2, 2]$, they should be scaled down to fit the mentioned interval. Otherwise, using a binary multiplier is an inevitable approach.

The proposed architecture also raises a challenge for generating the integer stochastic stream. To generate an integer stochastic stream S representing a real value $s \in [-m, m]$ in Integral SC, the real value s is first scaled down by $2^{\lceil \log_2(|s|) \rceil}$

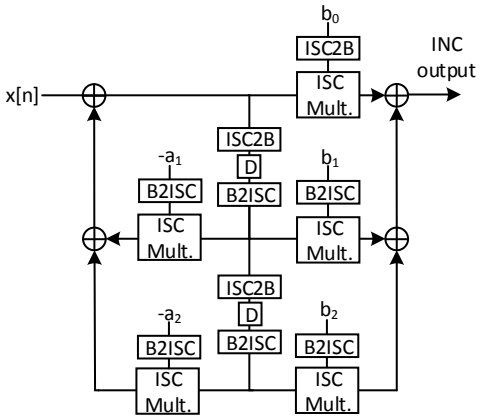

Fig. 4. The proposed stochastic architecture of a second-order IIR filter using Integral SC.

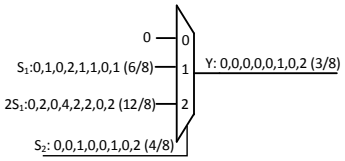

Fig. 5. The Integral SC multiplier.

to fit in $[-1, 1]$ interval. Then, $2^{\lceil \log_2(|s|) \rceil}$ number of the conventional B2S units are used to generate different stochastic streams. Finally, the generated streams are summed up together to create an integer stochastic stream. Therefore, $2^{\lceil \log_2(|s|) \rceil}$ B2S units are required in this approach, which potentially increases the complexity of the proposed stochastic architecture.

To overcome the aforementioned issue, we used the *dithered quantization* technique to generate the integer stochastic streams. The dithered quantization is used in the audio and video processing to randomize quantization errors [8]. In general, a real number $s \in [-m, m]$ is split down into two parts, an integer and a fractional part. The integer and fractional parts can be obtained as $\lfloor s \rfloor$ and $s - \lfloor s \rfloor$, respectively. A stochastic stream of the fractional part is generated by using conventional B2S unit. The integer stochastic stream of the real value s is then obtained by adding the integer part, i.e., $\lfloor s \rfloor$ with each element of generated stochastic stream as depicted in Fig. 6. Therefore, an integer stream can be easily generated by using a single B2S and an adder.

4. EXPERIMENTAL RESULTS

In this section, the simulation results of the proposed stochastic implementations of FIR/IIR filters are provided. Moreover, the proposed stochastic architectures were synthesized in a 65 nm CMOS technology for a hardware realization.

4.1. Simulation Results

To measure the accuracy of the proposed stochastic FIR filter, a mixture of several sinusoidal waves with different frequen-

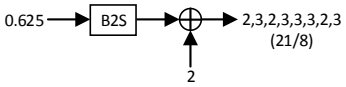

Fig. 6. The integer stochastic stream generation of the real value 2.625.

cies and a white noise are used as the inputs of filters. To this end, high-pass and low-pass filters with two different filter orders and four different cut-off frequencies are considered. A stream length of 1024 is used for generation of stochastic streams and 1000 input samples are used to test the proposed designs. Table 1 shows the accuracy of the proposed stochastic FIR filter in the form of error-to-signal power ratio with the aforementioned conditions. The simulation results show that the proposed architecture functions properly for high-order filters and the error remains roughly constant as the filter order increases as opposed to the previously proposed stochastic architectures.

To illustrate the performance of the proposed IIR filter based on Integral SC, we studied the 6^{th}-order IIR filter presented in [1] for a fair comparison. The transfer function of the low-pass filter is given by:

$$H(z) = \frac{0.0007378(1+6z^{-1}+15z^{-2}+20z^{-3}+15z^{-4}+6z^{-5}+6z^{-6})}{1-3.183z^{-1}+4.622z^{-2}-3.769z^{-3}+1.791z^{-4}-0.4593z^{-5}+0.0453z^{-6}}, \quad (10)$$

and the transfer function of the high-pass filter is given by:

$$H(z) = \frac{0.0007378(1-6z^{-1}+15z^{-2}-20z^{-3}+15z^{-4}-6z^{-5}+6z^{-6})}{1+3.183z^{-1}+4.622z^{-2}+3.769z^{-3}+1.791z^{-4}+0.4593z^{-5}+0.0453z^{-6}}. \quad (11)$$

Each of the above equations are first decomposed to three second-order direct-form II structures. The performance results of the proposed stochastic implementations of the IIR filters are summarized in Table 2. The simulations are performed under the same condition as the proposed FIR filter. For a fair comparison, the simulation results of the direct-form I structure was regenerated in this work. As shown in Table 2, the proposed architecture results in a roughly 4 orders of magnitude reduction in error compared to the direct-form I structure presented in [5]. Note that the lattice structure presented in [3] showed only 2 orders of magnitude improvement compared to the direct-form I structure.

4.2. Hardware Implementation Results

The proposed FIR/IIR stochastic filters were synthesized using `Cadence Encounter RTL Compiler` in a 65 nm CMOS technology. For a fair comparison, binary traditional implementations of FIR/IIR filters were also implemented in the same framework. The implementation results respectively show 90% and 79% reductions in area compared to the fixed-point implementations of FIR and IIR filters, which are summarized in Table 3. The numbers for traditional binary implementations are quantized to 10 bits, and the stream length of $L = 1024$ is used for hardware implementation of the proposed stochastic architectures. Note that the stream length can be reduced to $L = 64$ while maintaining a reasonably small error-to-signal power ratio. In that case, the worst error-to-signal power ratios of the proposed design become 0.036 for the FIR filters and 0.078 for the IIR filters.

Table 1. The output error-to-signal power ratio of the proposed stochastic FIR filters

Filter	Low-pass Cut-off Frequency			
Order	0.2π	0.4π	0.6π	0.8π
45	0.0014	0.0012	2.9×10^{-4}	5.3×10^{-4}
55	0.0012	0.0014	4.28×10^{-4}	5.3×10^{-4}
Filter	High-pass Cut-off Frequency			
Order	0.2π	0.4π	0.6π	0.8π
46	0.0012	0.0011	0.0012	0.0021
56	2.8×10^{-4}	8.1×10^{-4}	0.0011	0.0018

Table 2. The output error-to-signal power ratio of the proposed stochastic IIR filters

Filter Type	Direct Form [5]	Proposed
Low-pass	42.5721	0.0030
High-pass	43.3597	0.0011

Table 3. The hardware implementation of the proposed stochastic architectures for IIR/FIR filters in a 65 nm CMOS technology @ 400 MHz for a stream length of L.

Filter Type	FIR		IIR	
Implementation Type	ISC	Binary	ISC	Binary
Filter Order	56	56	6	6
Area (μm^2)	22,526	218,905	7,620	36,921
Latency (ns)	$2.5 \times L$	2.5	$2.5 \times L$	2.5

5. CONCLUSIONS

In this paper, firstly, a novel stochastic implementation of a FIR filter is proposed. The proposed architecture uses the APC and AND gate as its main computational units to perform the additions and multiplications required for filtering. It is shown that the error rate of the proposed architecture remains constant as the filter order increases as opposed to traditional stochastic hardware implementations. Secondly, a VLSI architecture for a second-order direct-form II structure of an IIR filter is proposed using the Integral SC. Therefore, a high-order IIR filter can be obtained by cascading series of second-order direct-form II structures. The error-to-signal power ratio results of the proposed IIR filter showed a roughly 2 orders of magnitude improvement compared to the lattice structure.

6. REFERENCES

[1] K.K. Parhi and Yin Liu, "Architectures for IIR digital filters using stochastic computing," in *Proc. 2014 IEEE International Symposium on Circuits and Systems (ISCAS)*, June 2014, pp. 373–376.

[2] Yin Liu and K.K. Parhi, "Lattice FIR digital filter architectures using stochastic computing," in *Proc. 2015 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, April 2015, pp. 1027–1031.

[3] Pai-Shun Ting and J.P. Hayes, "Stochastic Logic Realization of Matrix Operations," in *2014 17th Euromicro Conference on Digital System Design (DSD)*, Aug 2014, pp. 356–364.

[4] A. Ardakani, F. Leduc-Primeau, N. Onizawa, T. Hanyu and W. J. Gross, "VLSI Implementation of Deep Neural Network Using Integral Stochastic Computing," *CoRR*, vol. abs/1509.08972, 2015. [Online]. Available: http://arxiv.org/abs/1509.08972.

[5] Yun-Nan Chang and K.K. Parhi, "Architectures for digital filters using stochastic computing," in *Proc. 2013 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2013, pp. 2697–2701.

[6] B.R. Gaines, "Stochastic Computing Systems," in *Advances in Information Systems Science*, JuliusT. Tou, Ed., Advances in Information Systems Science, pp. 37–172. Springer US, 1969.

[7] Leland B. Jackson, "On the interaction of roundoff noise and dynamic range in digital filters," *Bell System Technical Journal*, vol. 49, no. 2, pp. 159–184, Feb 1970.

[8] K. Pohlmann, *Principles of Digital Audio*, McGraw-Hill Companies Inc., 2005.