An Efficient Hardware Implementation of Activation Functions Using Stochastic Computing for Deep Neural Networks

Van-Tinh Nguyen, Tieu-Khanh Luong, Han Le Duc, and Van-Phuc Hoang
Le Quy Don Technical University, 236 Hoang Quoc Viet Str., Hanoi, Vietnam

Abstract — **In this paper, we present a new approximation method for non-linear activation functions including tanh and sigmoid functions using stochastic computing (SC) logic based on the piecewise-linear approximation (PWL) for the full variable range of [-1, 1]. SC implementations with PWL approximation expansions for non-linear functions are based on a 90nm CMOS process. The implementation results shown that the proposed SC circuits can provide better performance compared with the previous methods such as the well-known Maclaurin expansions based, Bernstein polynomial based and finite-state-machine (FSM) based implementations. The implementation results are also presented and discussed.**

I. INTRODUCTION

Stochastic computing (SC) has attracted many researchers due to its low cost implementation of arithmetic units [1]-[2]. Many applications have employed SC such as artificial neural networks [3], image processing [4], decoding [5] and arithmetic functions computing [6]. Especially, in Deep Neural Networks (DNNs), non-linear activation functions, such as hyperbolic tangent (tanh) and sigmoid, are essential components. In [6], the complex arithmetic functions were implemented in SC by using Horner's rule for Maclaurin expansions. Moreover, piecewise linear (PWL) approximation has been used as an efficient method to approximate the complex arithmetic functions [7]. However, more arithmetic functions with further optimization and more improvements are expected for emerging applications and systems.

Moreover, there has been the remarkable success and growth of high-speed communication systems. By taking the benefits of CMOS technology scaling and advanced digital circuits design techniques, the Analog-to-Digital Converter (ADC) is positioned within the front-end chain closer to antenna, as shifting of the analog blocks (filter, mixer and amplifier) to the digital domain to increase the configurability and flexibility of the receivers and known as Digital Direct-sampling (DDS) receivers or RF direct receivers. These DDS receivers have only one analog front-end and most of the signal processing is performed in digital domain and hence provides scalable and programmable digital tuners.

Recently, with the increasing demand for high speed communications, efficient linearization methods for wideband transmitters are highly required. In [8], the authors have presented a one-step solution for transmitter nonlinearity estimation and linearization control in the presence of modulator imperfections for wideband direct-conversion transmitters with neural networks. By using a two-hidden-layer feedforward neural network, a high level of transmitter linearity can be achieved. An artificial neural network can also be applied for wideband pre-distortion of efficient pico-cell power amplifiers [9]. A. Baccigalupi et al. [10] has proposed a method based on neural network for error compensation in ADCs. However, in the last decade, shifting to all digital background calibration for high speed TIADCs is attracting many researchers working in the field [11]. Moreover, in [12], authors have proposed the method of using an evolutionary-computation algorithm for time-interleaved ADC background calibration which can be employed for wideband receivers. With the continuous development of deep neural networks [13], the application of neural network in TIADC for wideband transceivers is becoming feasible. In [14], we have proposed the approach of combining the stochastic logic with PWL approximation for different complex functions. However, the range of operand is limited to from 0 to 1. In practical systems, the approximation for the full variable range of [-1, 1] is required for activation functions in neural networks.

Therefore, the objective of this work is to propose a new method to approximate the complex arithmetic functions by using the stochastic logic combined with PWL approximation in which the coefficients are optimized to reduce MAE (mean absolute error) with the range of operand is from -1 to 1. The proposed approach could be applied for the neural network based digital assistance in wideband RF transceivers. For example with a 2-channel TIADC as shown in Fig. 1, one ADC (denoted as ADC2) is adjusted by utilizing a deep learning algorithm so that the channel mismatch can be minimized.

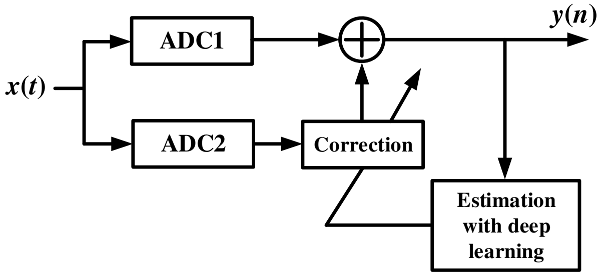

Fig.1. An application of the neural network for the TIADC in the receiver part of wideband RF transceivers.

978-1-5386-6689-0/18/$31.00 ©2018 IEEE
DOI 10.1109/MCSoC2018.2018.00045

The remainder of this paper is organized as follows. Section II presents the proposed method for arithmetic circuits implementation based on stochastic computing by piecewise linear approximation and the implementation results and discussions. Finally, section III concludes the paper.

II. Proposed method

In this work, we extended the approach of combining the stochastic logic with PWL approximation in [14] for the operand range of [-1, 1], optimized the building blocks for the SC circuits and validated the proposed method in 90nm CMOS technology.

A. PWL Aproximation for SC

The complex arithmetic functions *f(x)* is approximated by using PWL method, the range of $x \in [\alpha, \beta)$ is divided into *s* equal segments. In each segment, the function *f(x)* is approximated by a linear function as the following equation:

$$f(x) \approx a_i x + b_i, \quad \frac{i}{s}(\beta - \alpha) \le x < \frac{i+1}{s}(\beta - \alpha), \; i = 0 \div s - 1 \quad (1)$$

Furthermore, the value of coefficients a_i and b_i can be described as (2), in that *B* represents the number of bits of a_i and b_i.

$$a_i, b_i = N \times 2^B; \quad N, B \in \mathbb{Z} \quad (2)$$

The error in the *i*th segment is:

$$\varepsilon_i(x) = f(x) - (a_i x + b_i) \quad (3)$$

in which $\frac{i}{s}(\beta - \alpha) \le x < \frac{i+1}{s}(\beta - \alpha); \; i = 0 \div s - 1.$

The optimized values of coefficients a_i and b_i to minimize the approximation error are obtained by using the Algorithm 1 [14]. All functions are implemented with the range of $x \in [0,1)$, then extended to [-1, 1]. Table I presents the optimized values for a_i and b_i.

Algorithm 1. Optimization for a_i and b_i values.

```
1:  BEGIN: initialize array of $a$
           initialize array of $b$
           initialize array of $i$
2:         for $j$ = 1: length($a$)
               for $k$ = 1:length($b$)
                   for $t$ = 1:length($x$)
                           Calculating $\varepsilon_i(x)$ using (3)
                   end
               end
           end
3:         for $j$ = 1: length($a$)
               for $k$ = 1:length($b$)
                       Calculating $\varepsilon_{imax}$ = max( $\varepsilon_i(x)$)
               end
           end
4:         Calculating min $\varepsilon_{imax}$
5:  END:   Return $a_i$ and $b_i$ values corresponding to min $\varepsilon_{imax}$
```

B. Linear Feedback Shift Register

The computation error in SC caused by several sources such as random fluctuations in the stochastic number representation, correlations and physical errors. To improve the accuracy of SC, we need to produce uncorrelated and sufficiently random numbers. These numbers can be generated by using linear feedback shift registers (LFSR). In this work, *m* flip-flop is created with 2^m-1 different states as shown in Fig.2. The accuracy of SC increases if the number of states increases. In this work, we designed 8-bit registers to make 2^8 distinct states. The LFSR outputs are fed to the AND gate [1] to generate bit streams W_0-W_9 with non-overlapping 1s and weights of 1/2, 1/4, 1/8, 1/16, 1/32, 1/64, 1/128, 1/256, 1/512, and 1/1024 respectively.

Table I. The value of coefficients a_i and b_i of generated by using algorithm 1.

tanh			sigmoid		
Segment	a_i	b_i	Segment	a_i	b_i
0	255×2^{-8}	0	0	64×2^{-8}	128×2^{-8}
1	247×2^{-8}	2×2^{-8}	1	64×2^{-8}	128×2^{-8}
2	232×2^{-8}	5×2^{-8}	2	63×2^{-8}	128×2^{-8}
3	213×2^{-8}	12×2^{-8}	3	63×2^{-8}	128×2^{-8}
4	189×2^{-8}	24×2^{-8}	4	57×2^{-8}	131×2^{-8}
5	165×2^{-8}	39×2^{-8}	5	57×2^{-8}	131×2^{-8}
6	141×2^{-8}	57×2^{-8}	6	52×2^{-8}	135×2^{-8}
7	117×2^{-8}	78×2^{-8}	7	50×2^{-8}	137×2^{-8}
$\varepsilon_{\max} = 0.0012$			$\varepsilon_{\max} = 7.1\times10^{-4}$		

Table II. Bit-streams generated by LFSR.

Signal	**Bit-streams**											**Value**
L7	0	0	0	1	…	0	1	…	0	0	0	512/1024
L6	0	0	0	0	…	1	0	…	0	0	0	512/1024
L5	0	0	0	0	…	1	0	…	0	0	0	512/1024
L4	0	0	0	0	…	0	1	…	0	0	0	512/1024
L3	0	0	0	0	…	0	0	…	0	0	0	512/1024
L2	0	0	0	0	…	1	0	…	1	0	0	512/1024
L1	0	0	0	0	…	0	1	…	0	1	0	512/1024
L0	0	0	0	0	…	0	0	…	0	0	1	512/1024
W7	0	0	0	1	…	0	1	…	0	0	0	128/1024
W6	0	0	0	0	…	0	0	…	0	0	0	64/1024
W5	0	0	0	0	…	0	0	…	0	0	0	32/1024
W4	0	0	0	0	…	0	0	…	0	0	0	16/1024
W3	0	0	0	0	…	0	0	…	0	0	0	8/1024
W2	0	0	0	0	…	0	0	…	1	0	0	4/1024
W1	0	0	0	0	…	0	0	…	0	1	0	2/1024
W0	0	0	0	0	…	0	0	…	0	0	1	1/1024
X	0	1	0	1	…	0	1	…	1	1	0	755/1024

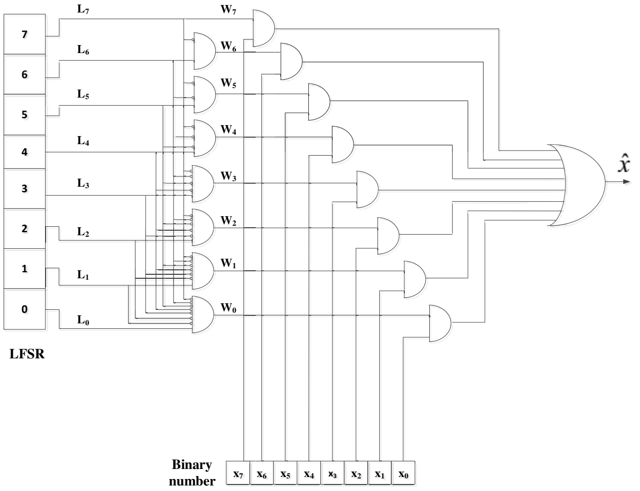

Fig.2. Stochastic number generator using LFSR.

This approach can be used to design the stochastic multiplication more accuracy. The Table II shows bit-streams generated by the circuit of Fig. 2 that are discussed in [1] with the initial states of register is 1100000111.

C. Hardware implementation for activation functions

This section presents the proposed SC-PWL hardware architectures. Moreover, MAE can be improved by adjusting the value of coefficients b_i in a ROM memory (called ROM-B as presented later). As a result, the proposed approximation method reaches a high level of accuracy. In the proposed approach, the complex arithmetic functions are approximated by multiple segments in equal intervals. The traditional multiplier occupied a lot of hardware resources. The stochastic logic for multiplier is implemented by using only AND gates [1]. The hardware complexity, therefore, decreases dramatically and so does the computation delay. The novelty of the proposed method is that arithmetic functions are computed by using stochastic logic combined with the optimized PWL approximation. Firstly, the complex arithmetic functions of tanh(x) and sigmoid(x) are approximated with PWL as:

$$f(x) = -a_i x + b_i,\ i = 0 \div 7 \qquad (4)$$

The hardware implementation of these functions requires multipliers and adders. The multiplier is implemented by the stochastic logic. However, the addition is implemented in the direct binary representation to reduce MAE by adjusting the coefficients b_i in the ROM-B as shown in Fig. 3. Two stochastic number generators (SNG) are used [6]. Three MSB of input (x) are used to select the value of coefficients in the ROM-A and ROM-B depending on the range of x.

The hardware architectures for these activation functions are implemented in a 90nm CMOS library using Synopsys Design Compiler tool. The length of the random number sequence is 256 with 8-bit LFSRs. We also implemented the functions with 8-bit LFSR for the Horner's rule for Maclaurin expansions based implementations [6] for the fair comparison.

Table III shows that the implementation results of functions ln(1+x), e^{-x} and tanh(x) with the proposed SC-PWL approach outperforms the previous study [6]. Table III shows synthesis results of various arithmetic functions in a 90nm CMOS library. The proposed approach results in lower hardware complexity than the approach based on Horner's rule for Maclaurin expansions. The critical path delays are also smaller than that in [6].

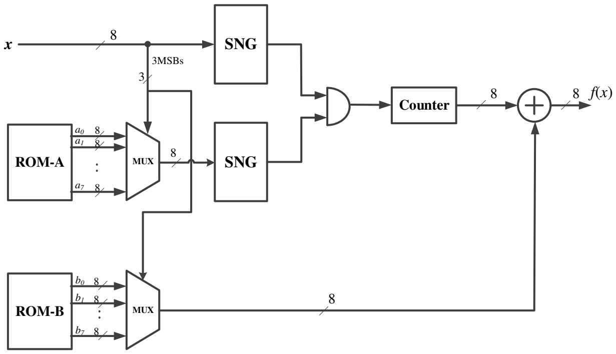

Fig. 3. The hardware architecture of stochastic implementation of tanh and sigmoid functions using PWL approximation.

Moreover, Table IV shows that the MAE results of the proposed implementations for tanh and sigmoid functions are better than the approaches based on Horner's rule for Maclaurin expansions, Bernstein polynomial and FSM as presented in [6]. However, in our implementations, 8-bit LFSRs are used instead of 10-bit ones [6]. For the accuracy adjustment, the value of coefficients of b_i can be adjusted to minimize error in the output [14].

In addition, the power consumption of the proposed circuits is reduced dramatically compared with the Horner's rule for Maclaurin expansions based implementations. For the hardware performance comparison, the same 8-bit LFSRs are used for the proposed architectures and Maclaurin expansions based implementations [6]. Based on these results, the proposed circuits can be used for low-power applications.

III. CONCLUSIONS

A new stochastic logic based approach for computing activation functions in neural networks, such as DNNs, using optimized PWL approximation and several circuit optimization techniques was presented in this paper. The implementation results have clarified the improvements of the proposed method. The implementation results for tanh and sigmoid functions were presented. However, the proposed method could be extended for other activation functions. Future works will focus on the stochastic logic implementations of the different arithmetic functions for the artificial neural networks using PWL approximation combined with the pipeline technique for high speed applications.

ACKNOWLEDGMENT

This research is funded by Vietnam National Foundation for Science and Technology Development (NAFOSTED) under grant number 102.02-2016.12.

Table III. The hardware complexity and critical path delay (ns) of stochastic implementations for arithmetic functions using proposed method, Maclaurin expansions method with different order in 90nm CMOS ASIC library.

Functions	Proposed			Horner's rule for Maclaurin expansions [6]			
	Area ($\times 10^3 \mu m^2$)	Delay (ns)	Power (μW)	Order	Area ($\times 10^3 \mu m^2$)	Delay (ns)	Power (μW)
tanh	3.650	1.72	81.62	7	4.933	1.74	83.51
sigmoid	3.982	1.73	104.48	5	5.350	1.74	189.17

Table IV. MAE results of stochastic implementations for various complex arithmetic functions using the proposed method compared with Maclaurin expansions, Bernstein polynomial and FSM methods.

Functions		Proposed	[6]			Bernstein polynomial [6]			FSM [6]
tanh	Order	-	3	5	7	3	5	7	8 states
	MAE	0.0029	0.0178	0.0175	0.0140	0.0182	0.0110	0.0082	0.0351
sigmoid	Order	-	5			-			8 states
	MAE	0.0024	0.0046			-			0.0198

REFERENCES

[1] A. Alaghi and J. P. Hayes, "Survey of stochastic computing," *ACM Transactions on Embedded Computing Systems*, vol.12, no.2s, pp. 92.2-92.19, May 2013.

[2] W. Qian, X. Li, M. D. Riedel, K. Bazargan and D. J. Lilja, "An Architecture for Fault-Tolerant Computation with Stochastic Logic," in *IEEE Transactions on Computers*, vol.60, no.1, pp. 93-105, Jan. 2011.

[3] B. D. Brown and H. C. Card, "Stochastic neural computation. I. computational elements," *IEEE Transactions on Computers*, vol. 50, no. 9, pp. 891-905, Sept. 2001.

[4] W. Qian, M. D. Riedel, H. Zhou, and J. Bruck, "Transforming probabilities with combinational logic," *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, vol. 30, no. 9,pp.1279-1292, Aug. 2011.

[5] A. Naderi, S. Mannor, M. Sawan, and W. J. Gross, "Delayed stochastic decoding of LDPC codes," *IEEE Transactions on Signal Processing*, vol. 59, no. 11, pp. 5617-5626, Nov. 2011

[6] K. Parhi; Y. Liu, "Computing Arithmetic Functions Using Stochastic Logic by Series Expansion," *IEEE Transactions on Emerging Topics in Computing* ,vol.PP, no.99, pp.1-13, Oct. 2016.

[7] V. P. Hoang, C. K. Pham, "Novel Quasi-Symmetrical Approach for Efficient Logarithmic and Anti-logarithmic Converters," Proc. 8th Conference on Ph.D. Research in Microelectronics & Electronics (PRIME2012), pp.1-4, Jun. 2012.

[8] H. Le Duc *et al.*, "Hardware implementation of all digital calibration for undersampling TIADCs," in *Circuits and Systems (ISCAS), 2015 IEEE International Symposium on*, May 2015, pp. 2181–2184.

[9] A. Baccigalupi, A. Bernieri and C. Liguori, "Error compensation of A/D converters using neural networks," in *IEEE Transactions on Instrument*ation and Measurement, vol. 45, no. 2, pp. 640-644, Apr 1996.

[10] D. Zhou, C. Talarico, J. Silva-Martinez, "A digital-circuit-based evolutionary-computation algorithm for time-interleaved ADC background calibration," 29th IEEE International System-on-Chip Conference (SOCC), pp. 13-17, Sept. 2016.

[11] Weibo Liu, Zidong Wang, Xiaohui Liu, Nianyin Zeng, Yurong Liu, Fuad E. Alsaadi, "A survey of deep neural network architectures and their applications," *Neurocomputing*, vo. 234, pp.11-26, Apr. 2017.

[12] M. Rawat and F. M. Ghannouchi, "A Mutual Distortion and Impairment Compensator for Wideband Direct-Conversion Transmitters Using Neural Networks," in *IEEE Transactions on Broadcasting*, vol. 58, no. 2, pp. 168-177, June 2012.

[13] M. K. Ngwar and J. S. Wight, "An artificial neural network for wideband pre-distortion of efficient pico-cell power amplifiers," *2014 6th International Symposium on Communications, Control and Signal Processing (ISCCSP)*, Athens, 2014, pp. 562-565.

[14] Van-Tinh Nguyen, Van-Phuc Hoang, Van-Thuan Sai, Tieu-Khanh Luong, Minh-Tu Nguyen, Han Le Duc, "A New Approach of Stochastic Computing for Arithmetic Functions in Wideband RF Transceivers," 2017 IEEE 60th International Midwest Symposium on Circuits and Systems (MWSCAS2017), pp. 1525-1528, Aug. 2017.