

Multiplier-Adder-Converter Linear Piecewise Approximation for Low Power Graphics Applications

Dina M. Ellaithy¹, Magdy A. El-Moursy², Amal Zaki¹, and Abdelhalim Zekry³

¹ Microelectronics Department, Electronics Research Institute (ERI), ²Design Creation Division, Mentor Graphics,

³Electronics & Communications Department, Ain Shams University, Cairo, Egypt

dina_ellaisy@eri.sci.eg, magdy_el-moursy@mentor.com, amalzaki@gmail.com, aaazekry@hotmail.com

Abstract- Multiplication is the core operation used in the linear piecewise approximation. In this paper, the logarithmic number system (LNS) is exploited to design low power and high speed multiplier-adder-converter (MAC). MAC is included in the piecewise linear polynomial approximation to achieve efficient power design with large operand input size. Also, a significant enhancement in the implementation of the linear piecewise approximation is achieved as compared to the conventional architecture. Hardware implementation and synthesis are performed using 90 nm CMOS technology. Up to 30% and 55% reduction in power and delay are attained with the proposed design. MAC considerably improves the power delay product by up to 70% as compared to the well-known multipliers.

I. INTRODUCTION

Recently, logarithmic number system (LNS) has been used to simplify different complex operations. Although logarithmic arithmetic decreases the complexity in some operations, it complicates simple operations such as addition and subtraction [1, 2]. Hence, using LNS for all operations may not be an optimum solution. Piecewise polynomial approximation algorithm has become commonly used algorithm to compute various operations such as reciprocal, square root and trigonometric functions which are used frequently in variety of applications including graphics applications [3-12]. The essential architecture of piecewise linear approximation includes look-up table, multiplier, and adder [4, 5], [9-12]. The core block as demonstrated in [10-12] is the multiplier. The performance of the piecewise linear approximation is determined according to the multiplication scheme. The number of partial products to be summed is the highest constraint that controls the performance of the multiplier. Thus, booth multiplier designs with different radix have been proposed to shrink the partial products [6, 7]. For large operand input sizes, the logic depth and the power requirement increase significantly. Large word size multipliers become more complex to implement and cannot be appropriate for low power portable applications [9-11]. Several approaches tried to optimize the performance by decreasing the partial products. Focusing on high speed, degrades the power dissipation profile for the extra added hardware. LNS is used to directly handle the huge growth in hardware size for large input words. Multiplication process is reduced to addition in the logarithmic domain [13, 14]. Exploiting the advantage of cutting down the process degree from multiplication to addition in the evaluation of the linear piecewise approximation achieves a rapid decrease in the power consumption and delay.

A linear piecewise polynomial approximation design is developed and synthesized in this paper. The traditional multiplier is replaced by the proposed MAC in the linear piecewise approximation as shown in Fig. 1. The paper is arranged as follows; an overview is demonstrated in section II. The MAC associated with the proposed linear piecewise approximation are introduced in section III. In section IV, hardware implementation and results are obtained with comparison with the conventional architectures. Section V is presented with some conclusions.

II. OVERVIEW

According to (1), the computation of the polynomial equation of degree one includes multiplication process. The different conventional parallel multipliers for large word sizes become more complex to implement and cannot be appropriate for low power handheld applications [9-11]. Also, the design becomes more complex with high implementation cost. Large volume of hardware is exhausted by the array multipliers which lead to high power consumption. Wallace tree approach is used to speed up the multiplication process by reducing the number of sequential addition phases. Booth algorithm is one of the most popular algorithms that is proposed with different radix [6, 7] to reduce the number of partial products. As the number of bits of input operand increases, the partial product

generator becomes more complex. Unfortunately, the complexity of the hardware of the piecewise polynomial approximation increases as the input operand increases. Multiplier-adder-converter (MAC) decreases the cost of hardware with large input operand size. To verify the proposed design in this paper, a linear piecewise polynomial approximation for function evaluation is developed and synthesized.

$$f(x) = C_0 + C_1 \times X \quad (1)$$

III. PROPOSED LINEAR PIECEWISE APPROXIMATION WITH MAC

Multipliers are the heart of any degree piecewise polynomial approximation architecture. MAC is utilized in the piecewise polynomial approximation to increase power saving and enhance speed. In the next subsections, MAC scheme and the proposed linear piecewise approximation are presented.

A. MAC Scheme

MAC architecture is obtained in Fig. 2. The multiplication process is replaced with addition in the logarithmic field. Log_2 block represents the logarithmic converter while Alog_2 block represents the antilogarithmic converter. The transformation from binary number to logarithmic number is carried out by the logarithmic converter. The antilogarithmic converter is responsible for converting the results from the logarithmic field to the binary field. Two logarithmic converters, one antilogarithmic converter, and an adder are utilized for the multiplication task. The two inputs (multiplier and multiplicand) pass through the logarithmic converters at the beginning as shown in Fig. 2. Subsequently, adding the outputs that are resulted from the two logarithmic converters is performed. At the end, the antilogarithmic converter is used to convert the result from the logarithmic domain to the binary domain. As shown in Fig. 2, the main blocks of the multiplication process in the logarithmic field are the logarithmic and antilogarithmic converters. To enhance the overall performance, wide different approximation techniques have been proposed to trade-off hardware performance with error minimization [13-16]. Among different algorithms, the piecewise linear approximation algorithm which is proposed in [14] gives good trade-off between precision and hardware cost and so the power dissipation. The final addition process in Fig. 2 can be performed by different category of adders. The proposed linear piecewise approximation with MAC is introduced in the next subsection.

B. Proposed Linear Piecewise Approximation

As shown in Fig. 1, the traditional multiplier is replaced with the proposed MAC in the linear piecewise approximation. In this architecture, MAC replaces multiplication by addition. Therefore, large cutting down in the power dissipation is achieved. The input operand X is divided into two portion X_m and X_l . The upper portion X_m is responsible for obtaining the values of the coefficients C_0 and C_1 to be stored in the look-up table. MAC piecewise polynomial approximation achieves saving in hardware with uniform or non-uniform segmentations. Also, large savings in power and delay are achieved in the second-order piecewise approximation with MAC [8] as shown in Fig. 3. In the next section, hardware results are compared with the previous recent work.

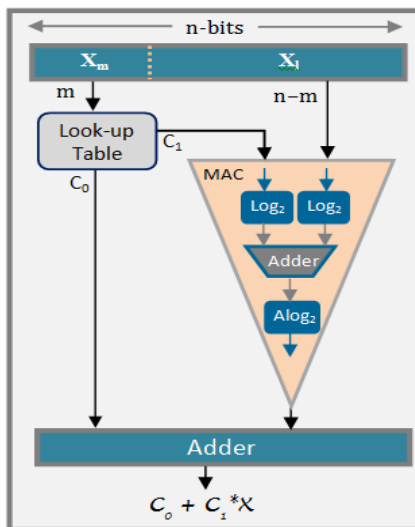


Figure 1. The architecture of the proposed linear piecewise approximation including MAC

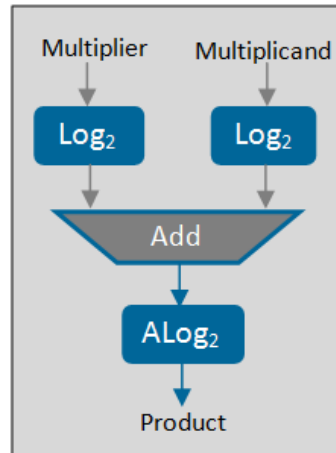


Figure 2. The primary architecture of multiplier-adder-converter (MAC) in logarithmic domain

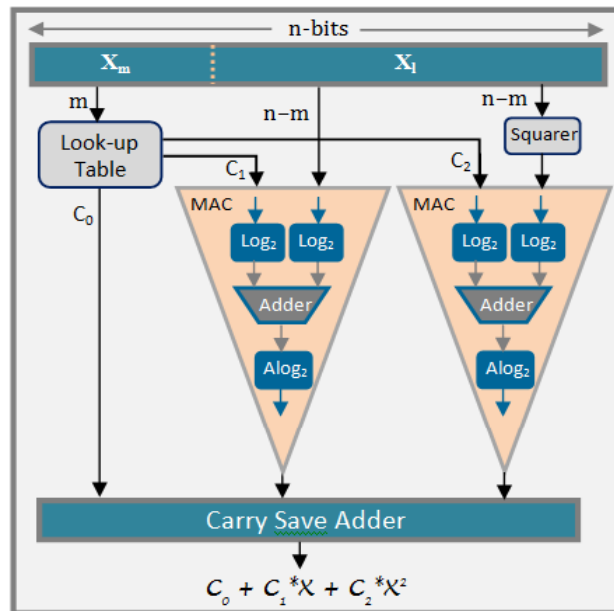


Figure 3. The architecture of the second-order piecewise approximation including MAC

IV. HARDWARE IMPLEMENTATION AND RESULTS

Traditional parallel multipliers such as Wallace, booth radix4, and booth radix8 are implemented and compared with the proposed technique MAC. Hardware implementation is performed with synthesis tool using 90 nm CMOS technology, 1.0 V supply voltage standard cell library. The power consumption and delay have been considerably decreased with MAC scheme. The results of comparing MAC and different popular multipliers are listed in Table I and Table II. Table I includes the results of 16-bits multipliers while Table II includes the 24-bits multipliers. Note that, the logarithmic and antilogarithmic converters which are included in MAC architecture (as shown in Fig. 2) are implemented as proposed in [14]. The proposed converters give a compromise between precision and hardware complexity [14]. A carry propagate adder (CPA) is utilized for addition in logarithmic field. Column 1 includes the multiplier scheme. The implementation results in terms of power, area, delay, and power-delay-product are listed in columns 2-5, respectively.

Moreover, the proposed linear piecewise approximation is implemented and synthesized. In Table III and Table IV, the results of comparing the proposed MAC-linear piecewise approximation and previous conventional approaches using same technology are reported. The power, delay, and energy savings have been raised with the MAC scheme. In prior approaches, different functions are implemented such as reciprocal ($1/x$), trigonometric, power-of-two, $\log_2(1+x)$, and logarithmic. The results contain comparison for the reciprocal function ($1/x$) for 24-bit accuracy. Also, the function $\log_2(1+x)$ is implemented using MAC approach and the conventional approaches with 20-bit precision. For higher-bit accuracy, it is better to use MAC scheme to approximate the function due to the high savings in hardware with large operand sizes. According to Masoud [11], 26 and 12 bits are needed for the approximation coefficients C_0 and C_1 for the 24-bit accuracy reciprocal function using first-degree piecewise approximation, respectively. The size of the multiplier that computes $C_1 \times X$ in the linear piecewise approximation is 12 by 13 bits for 24-bit accuracy. The multiplier operation in the piecewise linear approximation is implemented using booth radix4 and booth radix8 multipliers and compared with MAC approach. The final adder is implemented by carry propagate adder (CPA). The results of the function $1/x$ are reported in Table III. The implementation of faithfully rounded piecewise linear approximation of $\log_2(1+x)$ with 20-bit precision requires 25-bit and 12-bit for the approximation coefficients C_0 and C_1 according to De Caro [10]. 12 by 11 bits multiplication process is required [10]. Also, the comparison between MAC and traditional parallel multipliers (radix4 and radix8 booth) is listed in Table IV.

From Table I and Table II, it is shown that MAC scheme achieves high reduction in power dissipation, delay and energy as compared to previous traditional multiplication schemes. Saving in power is up to 14.5% for the MAC in Table I, and is up to 27.7% in Table II as compared with booth radix4 multiplier. The multiplication process is converted to addition by applying MAC which accomplish high power and delay savings.

TABLE I
16-BITS MAC COMPARED WITH DIFFERENT TRADITIONAL PARALLEL MULTIPLIERS IN TERMS OF POWER, AREA, DELAY, AND ENERGY

Scheme	Power (mW)	Area (μm^2)	Delay (ns)	Power-Delay Product (pJ)
Wallace Multiplier	0.57606	6773.76	3.25	1.87219
Booth radix4 Multiplier	0.54603	5395.49	4.27	2.33155
Booth radix8 Multiplier	0.56515	5890.98	3.59	2.02889
MAC	0.46688	7985.040	1.63	0.76101

TABLE II
24-BITS MAC COMPARED WITH DIFFERENT TRADITIONAL PARALLEL MULTIPLIERS IN TERMS OF POWER, AREA, DELAY, AND ENERGY

Scheme	Power (mW)	Area (μm^2)	Delay (ns)	Power-Delay Product (pJ)
Wallace Multiplier	1.8635	15574.160	4.96	9.2430
Booth radix4 Multiplier	1.1815	13087.312	6.23	7.3607
Booth radix8 Multiplier	1.4065	13143.760	5.53	7.7780
MAC	0.85416	14494.576	2.53	2.1610

TABLE III
COMPARISON BETWEEN THE PROPOSED LINEAR PIECEWISE APPROXIMATION FOR THE $1/x$ FUNCTION IN TERMS OF POWER, AREA, DELAY, AND ENERGY AT 100 MHZ

Technique	Power (mW)	Area (μm^2)	Delay (ns)	Power-Delay Product (pJ)
Masoud et al. [11] [Booth radix4 Multiplier]	0.8295	6192.82	5.14	4.2636
Reduction (%)	30.05%	-14.93%	56.81%	64.58%
Masoud et al. [11] [Booth radix8 Multiplier]	1.019	8405.6	4.29	4.3715
Reduction (%)	52.40%	13.40%	48.25%	65.46%
Proposed Linear with MAC	0.6802	7279.44	2.22	1.5100

TABLE IV
COMPARISON BETWEEN THE PROPOSED LINEAR PIECEWISE APPROXIMATION FOR THE $\log_2(1+x)$ FUNCTION IN TERMS OF POWER, AREA, DELAY, AND ENERGY AT 100 MHZ

Technique	Power (mW)	Area (μm^2)	Delay (ns)	Power-Delay Product (pJ)
De Caro et al. [10] [Boothradix4 Multiplier]	0.7105	4405.6	4.70	3.3393
Reduction (%)	29.91%	-21.66%	55.53%	68.83%
De Caro et al. [10] [Booth radix8 Multiplier]	0.9560	6729.7	3.87	3.6997
Reduction (%)	47.91%	16.44%	46%	71.88%
Proposed Linear with MAC	0.4980	5623.4	2.09	1.0408

When compared with the work in [11] and [10], the MAC linear piecewise approximation achieves high reductions in power, delay, and energy. In Table III, for the reciprocal function with 24-bit accuracy, the proposed architecture attains reductions of 30%, 56%, and 64% in power consumption, delay, and power-delay-product as compared to booth radix4 approach, respectively. The reductions which are obtained by the MAC linear piecewise approximation are at least 52%, 48%, and 65% in power, delay, and energy as compared to booth radix8 multiplier, respectively. The comparison in Table IV for the function $\log_2(1+x)$ demonstrates saving in power and delay by around 30% and 55% for 20-bit precision as compared to conventional radix4 booth approach. Also, comparison with radix8 booth multiplier linear piecewise approximation exhibits up to 47%, 46%, and 71% decrease in power dissipation, delay, and energy, respectively. From the results in Table III and Table IV, as the input operand size increases, the saving by

using MAC approach increases in power, delay, and energy. The proposed MAC piecewise linear approximation is significantly energy efficient.

V. CONCLUSIONS

The performance of the linear piecewise approximation can be improved by enhancing the performance of the multiplier which is the core operation in the piecewise-polynomial approximation. Recently, logarithmic number system (LNS) has gained the attention of researchers for its simplicity in basic arithmetic calculations such as multiplication and division. In this paper, Multiplier-Adder-Converter (MAC) is proposed. MAC achieves less power consumption, delay, and energy as compared to traditional multipliers scheme. Up to 27%, 59%, and 70% reductions in power, delay, and power-delay-product are attained with MAC as compared to the radix4 booth multiplier. Hardware requirements are decreased by using MAC in piecewise linear approximation. Compared to the conventional architecture, the proposed approach exhibits saving in power, delay, and energy by up to 52%, 48%, and 65% for the reciprocal function, respectively.

REFERENCES

- [1] J. N. Coleman, and R. Che Ismail, "LNS with Co-transformation Competes with Floating-Point," *IEEE Transactions on Computers*, vol. 65, no. 1, pp. 136-146, January 2016
- [2] Siti Zarina Md Naziri, Rizalafande Che Ismail, and Ali Yeon Md Shakaff, "An Analysis of Interpolation Implementation for LNS Addition and Subtraction Function in Positive Region," *IEEE International Conference on Computer and Communication Engineering (ICCCCE)*, pp. 499-504, July 2016.
- [3] Davide De Caro, Nicola Petra, and Antonio G. M. Strollo, "High-Performance Special Function Unit for Programmable 3-D Graphics Processors," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 56, no. 9, pp. 1968-1978, September 2009.
- [4] Shen-Fu Hsiao, Hou-Jen Ko, Yu-Ling Tseng, Wen-Liang Huang, Shin-Hung Lin, and Chia-Sheng Wen, "Design of Hardware Function Evaluators Using Low-Overhead Nonuniform Segmentation With Address Remapping," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 21, no. 5, pp. 875-886, May 2013.
- [5] Shen-Fu Hsiao, Hou-Jen Ko, and Chia-Sheng Wen, "Two-Level Hardware Function Evaluation Based on Correction of Normalized Piecewise Difference Functions," *IEEE Transactions on Circuits and Systems II: Express Briefs*, vol. 59, no. 5, pp. 292-296, April 2012.
- [6] Abd-Elrahman G. Qoutb, Abdullah M. El-Gunidy, Mohammed F. Tolba, and Magdy El-Moursy, "High Speed Special Function Unit for Graphics Processing Unit," *International Design & Test Symposium*, pp. 24-29, December 2014.
- [7] Jose-Alejandro Pineiro, Stuart F. Oberman, Jean-Michel Muller, and Javier D. Bruguera, "High-Speed Function Approximation Using a Minimax Quadratic Interpolator," *IEEE Transactions on Computers*, vol. 54, no. 3, pp. 304-318, March 2005.
- [8] Dina M. Ellaithy, Magdy A. El-Moursy, Amal Zaki and Abdelhalim Zekry, "Dual Channel Multiplier for Piecewise-Polynomial Function Evaluation for Low-Power 3D Graphics Applications,"
- [9] Masoud Sadeghian, and James E. Stine, "Optimized Low-Power Elementary Function Approximation for Chebyshev Series Approximations," *Conference Record of the Forty Sixth Asilomar on Signals, Systems and Computers (ASILOMAR)*, pp. 1005-1009, November 2012.
- [10] Davide De Caro, Ettore Napoli, Darjn Esposito, Gerardo Castellano, Nicola Petra, and Antonio G. M. Strollo, "Minimizing Coefficients Wordlength for Piecewise-Polynomial Hardware Function Evaluation With Exact or Faithful Rounding," *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 64, no. 5, pp. 1187-1200, May 2017.
- [11] Masoud Sadeghian, James E. Stine, and E. George Walters, "Optimized Linear, Quadratic and Cubic Interpolators for Elementary Function Hardware Implementations," *Electronics*, vol. 5, no. 2, pp. 1-25, April 2016.
- [12] Shen-Fu Hsiao, Chia-Sheng Wen, and Po-Han Wu, "Compression of Lookup Table for Piecewise Polynomial Function Evaluation," *17th Euromicro Conference on Digital System Design (DSD)*, pp. 279-284, August 2014.
- [13] Davide De Caro, Mariangela Genovese, Ettore Napoli, Nicola Petra, and Antonio G. M. Strollo, "Accurate Fixed-Point Logarithmic Converter," *IEEE Transactions On Circuits And Systems*, vol. 61, no. 7, pp. 526-530, July 2014.
- [14] Dina M. Ellaithy, Magdy A. El-Moursy, Ghada Hamdy, Amal Zaki and Abdelhalim Zekry, "Double Logarithmic Arithmetic Technique for Low-Power 3-D Graphics Applications," *IEEE Transactions On Very Large Scale Integration*, vol. 25, no. 7, pp. 2144-2152, July 2017.
- [15] Chao-Tsung Kuo, and Tso-Bing Juang, "Design of fast logarithmic converters with high accuracy for digital camera application," *Springer Microsystem Technologies*, pp. 1-9, August 2016.
- [16] R. Rachel Selina, "VLSI implementation of Piecewise Approximated antilogarithmic converter," *The Proceedings of Communications and Signal Processing Conference*, pp. 763 - 766, April 2013.