

# Design and Evaluation of An Approximate Wallace-Booth Multiplier

Liangyu Qian<sup>1</sup>, Chenghua Wang<sup>1</sup>, Weiqiang Liu<sup>1</sup>, Fabrizio Lombardi<sup>2</sup> and Jie Han<sup>3</sup>

<sup>1</sup>College of EIE, Nanjing University of Aeronautics and Astronautics, Nanjing, China, 210016. liuweiqiang@nuaa.edu.cn

<sup>2</sup>Department of ECE, Northeastern University, Boston, MA, USA, 02115. lombardi@ece.neu.edu

<sup>3</sup>Department of ECE, University of Alberta, Edmonton, Alberta T6G 1H9, Canada. jhan8@ualberta.ca

**Abstract**—Approximate or inexact computing has recently attracted considerable attention due to its potential advantages with respect to high performance and low power consumption. This paper presents the design of an approximate multiplier; this approximate multiplier consists of an approximate Booth encoder, an approximate 4-2 compressor and an approximate tree structure. The approximate design is implemented and verified for  $8\times 8$ ,  $16\times 16$  and  $32\times 32$ -bit signed multiplication schemes targeting applications in embedded systems. Simulation results at 45 nm technology are provided and discussed. Compared with an exact Wallace-Booth multiplier as well as other approximate multipliers found in the technical literature, the proposed approximate scheme achieves significant improvements in power consumption, delay and combined metrics. These results show the viability of the proposed design.

**Keywords**—approximate multiplier; inexact computing; low power; delay; error analysis.

## I. INTRODUCTION

High precision and exactness in the operations of digital logic circuits are related to the generally accepted requirement of correctness of information processing. Many applications such as those related to human brain (such as speech, image and video processing) do not require complete or exact results to still be meaningful and useful [1]. Error tolerant applications are extensive in embedded computing; by relaxing the requirement of strict accuracy some measures such as power dissipation and performance can be improved. This design paradigm is generally known as approximate or inexact computing [2].

As the basic operations of an arithmetic processor, addition and multiplication are very important for achieving high performance. Addition has been extensively studied for approximate computing to attain reductions in power consumption and delay. New metrics for evaluating the operation of an approximate adder have been proposed in [3]. The error distance (ED) is defined as the arithmetic distance between an erroneous output and the correct output. The mean error distance (MED) and the normalized error distance (NED) are also used as both the averaging effect of multiple inputs and the normalization of multiple-bit adders must be considered.

Approximate multiplication has not been extensively studied despite its importance in arithmetic processing and systems. Several approximate multipliers that use a truncated

multiplication method have been proposed in [4-7]. In [8], two new approximate 4-2 compressors have been proposed for designing an approximate array multiplier. This design is usually better than a truncated scheme, because it overcomes most of the accuracy and correctness issues due to truncation.

The Wallace-Booth multiplier design is the most popular design solution because it is significantly faster than array multipliers, *i.e.*, it reduces the number of partial products by using a modified Booth encoder (MBE). A Wallace-Booth multiplier mainly consists of three parts: partial product generation, partial product compression and final product generation with a carry propagation adder. Although approximate compressors have been considered in [8], no research has been conducted on the approximate design of the Booth encoder and the tree structure.

In this paper, an approximate Wallace-Booth approximate multiplier is proposed based on utilizing approximate modules in the Booth encoder, the 4-2 compressor (proposed in [8]) and the Wallace tree. Simulation results on area, delay and power consumption at 45 nm CMOS technology show that the proposed approximate multiplier has better performance in terms of power consumption and delay compared with the exact multiplier as well as the approximate Wallace-Booth multiplier that only uses an approximate 4-2 compressor. An error analysis using NED is also provided.

The rest of paper is organized as follows. Section 2 reviews the exact Wallace-Booth multipliers. The approximate Booth encoder and a new tree scheme are proposed in Section 3. Simulation results for the proposed approximate Wallace-Booth multiplier are presented in Section 4. Section 5 concludes the paper.

## II. EXACT WALLACE-BOOTH MULTIPLICATION

Multiplication can be thought as a series of shifted additions. In the past, it has been implemented sequentially using adders and multiple cycles, hence at a lower speed. With advancements in the VLSI technology, multiplication recoding and the Wallace tree are widely utilized in a multiplier design at significantly better performance. A multiplier consists of three steps: partial product generation, partial product compression and final product generation. The usual focus of an approximate multiplier design is on the first two steps due to that the multiplication recoding and tree structure can be improved. Modified Booth Encoder and 4-2

---

This work is supported by grants from National Natural Science Foundation of China (61401197) and Natural Science Foundation of Jiangsu Province (BK20151477).

compressor are widely utilized for their relatively mature structures.

### A. The Modified Booth Encoder

The generation of the partial products is the first step of multiplication, and Booth encoding is very efficient for this process. Booth encoding reduces the number of rows for the partial products ( $PP_j$ ) in a multiplier. The complexity of a Booth encoder significantly affects the delay and power consumption of the entire multiplier, because it determines the number of partial products.

The modified Booth encoding (MBE) algorithm was introduced in [9]; the MBE algorithm is easier to implement and has a lower delay than the original Booth encoder. Let  $X$  be the multiplicand and  $Y$  the multiplier. The output of the Booth encoder is given by

$$PP_j = (X_{2i} \oplus X_{2i-1})(X_{2i+1} \oplus Y_j) + \overline{(X_{2i} \oplus X_{2i-1})}(X_{2i+1} \oplus X_{2i})(X_{2i+1} \oplus Y_{j-1}) \quad (1)$$

Fig. 1 shows the gate level design of the modified Booth algorithm [9].

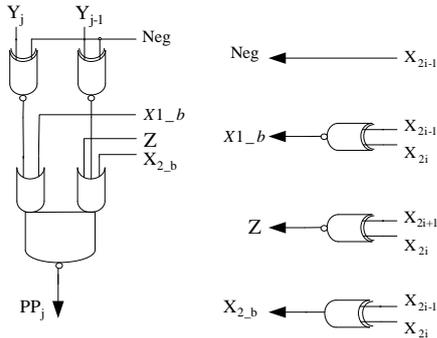


Fig. 1. The encoder circuit of the MBE scheme of [9].

### B. Exact 4-2 Compressor

Generally,  $N-2$  compressors are widely used in the design of digital multipliers. However, not every  $N-2$  compressor is suitable for partial product reduction in a multiplier.

The 4-2 compressor is very efficient and widely used in computer arithmetic. Two full adders are combined into the compressor [10]; therefore, it has five inputs (*i.e.*,  $C_{in}$ ,  $P_1$ ,  $P_2$ ,  $P_3$ ,  $P_4$ ) and three outputs (*i.e.*,  $Sum$ ,  $C_{out}$ ,  $Carry$ ).  $C_{in}$ ,  $P_1$ ,  $P_2$ ,  $P_3$ ,  $P_4$  and  $Sum$  have the same weight of 1, while  $C_{out}$  and  $Carry$  have a weight of 2.

## III. APPROXIMATE WALLACE-BOOTH MULTIPLIER

In this section, an approximate Wallace-Booth multiplier is proposed. This approximate multiplier consists of an approximate modified Booth encoder, an approximate 4-2 compressor (that was proposed in [8]) and an approximate Wallace tree.

### A. The Approximate MBE Algorithm

As shown in (1), five inputs generate a partial product. The Karnaugh map of the MBE is shown in Table I.

TABLE I. KARNAUGH MAP OF MBE

$X_{2i+1}X_{2i}X_{2i-1}$	$Y_j Y_{j-1}$							
1	000	001	011	010	110	111	101	100
00	0	0	0	0	1	0	1	1
01	0	0	1	0	1	0	1	0
11	0	1	1	1	0	0	0	0
10	0	1	0	1	0	0	0	1

The complexity of the  $PP_j$  function is related to the symmetry of the K-map; the design of the new approximate modified Booth encoder (AMBE) is achieved by replacing 1 with 0 (Table I), thus making the K-map of the MBE more symmetrical. The function in the proposed approximate Booth encoder is as follows:

$$PP_j = (X_{2i} \oplus X_{2i-1})(X_{2i+1} \oplus Y_j) \quad (2)$$

By using (2) only four inputs are used to produce one partial product. Table II shows the truth table of the AMBE algorithm. It also shows the difference between the outputs of exact Booth encoding ( $PP_j$ ) and the outputs of approximate Booth encoding ( $APP_j$ ). As shown in Table II, the proposed AMBE only introduces two incorrect outputs out of sixteen outputs (*i.e.*, error rate of 12.5%).

TABLE II. TRUTH TABLE OF AMBE

$Y_j$	$X_{2i+1}$	$X_{2i}$	$X_{2i-1}$	$PP_j$	$APP_j$	$Diff.$
0	0	0	0	0	0	0
0	0	0	1	0	0	0
0	0	1	0	0	0	0
0	0	1	1	0	0	0
0	1	0	0	1	0	-1
0	1	0	1	1	1	0
0	1	1	0	1	1	0
0	1	1	1	0	0	0
1	0	0	0	0	0	0
1	0	0	1	1	1	0
1	0	1	0	1	0	0
1	0	1	1	0	0	0
1	1	0	0	1	0	-1
1	1	0	1	0	0	0
1	1	1	0	0	0	0
1	1	1	1	1	0	0

The gate level design of the proposed AMBE is shown in Fig. 2. The conventional design of a MBE (Fig. 1) [9] consists of four XNOR gates, one NOR gate, two AND gates and one OR gate. Therefore, the critical path of an exact MBE has a delay of  $4\tau$ , where  $\tau$  is the unitary delay through any gate in the design. The proposed design of AMBE consists of only two XOR gates and one AND gate. Therefore, its critical path delay is reduced to  $2\tau$ .

### B. Approximate 4-2 Compressor

Two approximate 4-2 compressors have been proposed in [8]; they are both designed by changing few values in the

truth table of the compressor. However, the very first approximate design generates fifteen inexact outputs out of thirty two. The error rate of this design is rather high, *i.e.*, 46.9%. The second 4-2 compressor is a very efficient design and is used in this work. The logic functions of this approximate 4-2 compressor are given by [8]:

$$Sum' = \overline{(P_1 \oplus P_2 + P_3 \oplus P_4)} \quad (3)$$

$$Carry' = \overline{(P_1 P_2 + P_3 P_4)} \quad (4)$$

At gate level, the critical path delay of this design is  $2\tau$ ; this design requires only two XNOR gates, two NAND gates, one NOR gate and one OR gate.

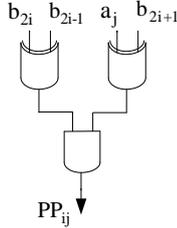


Fig. 2. The encoder circuit of AMBE.

### C. Approximate Wallace Tree

A tree is used to perform the reduction of the partial product rows until only two rows remain. Fig. 3 shows the MBE partial product arrays for an  $8 \times 8$  multiplier [9], where *Neg* and the sign extension term appear in the partial product array. *Neg* is determined by the LSB of the *y* signal combined with the *x* signals. The sign extension term is determined by the MSB of the *y* signal combined with the *x* signals; refer to [9] for more details on these terms.

Due to the presence of the fifth row *Neg* term which is represented as  $\Delta$ , two stages are needed in the exact compression of the partial product array for an  $8 \times 8$  bit multiplier; however, in the design of an approximate Wallace tree,  $\Delta$  can be ignored because it is the least significant bit of the final partial product vector; therefore, sixteen carry save adders are not needed in the approximate tree structure.

The equation for  $\Delta$  is given by

$$PP_{n/2-1} = b_n \overline{b_{n-1} b_{n-2}}, \quad (5)$$

where  $n$  is the size of multiplier and  $b$  is the multiplicand of the multiplier. The error rate is then equal to 37.5%.

### D. Approximate Wallace-Booth Multiplier

The design of an approximate Wallace-Booth multiplier is divided into two parts by the position of  $\Delta$ . AMBE and the approximate 4-2 compressor are used in the  $n/2$  least significant bits (LSBs) (*i.e.*, there are 8 LSBs for the  $8 \times 8$  bit multiplier), the most significant bits (MSBs) use an exact MBE and 4-2 compressors.  $\Delta$  is also ignored.

As the approximate (exact) designs are used in the least (most) significant columns, improvements in delay and power consumption can be achieved at reasonable accuracy.

Fig. 4 shows the approximate Wallace-Booth multiplier. Consider the  $8 \times 8$  multiplier as an example. A box with a solid line represents the use of an exact 4-2 compressor; a box with the dotted line represents an approximate 4-2 compressor, while the exact partial product is represented by  $\blacktriangle$  and the approximate partial product is represented by  $\blacksquare$ . In the first stage,  $\Delta$  is ignored; in the second stage, a 16-bit carry look-ahead adder is used to generate the final product.

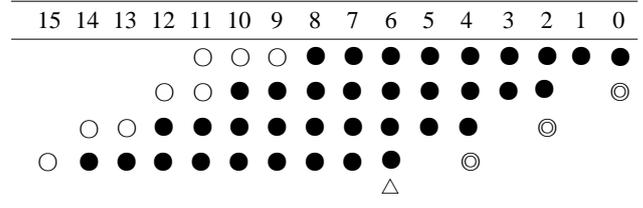


Fig. 3. A conventional  $8 \times 8$  MBE partial product array, where  $\bullet$  denotes the  $pp_{ij}$  term,  $\circ$  denotes the sign extension term,  $\odot$  denotes the *Neg* term and  $\Delta$  is the ignored *Neg* term.

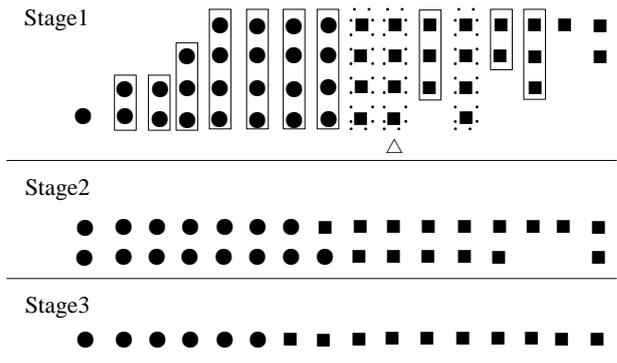


Fig. 4. Diagram of an approximate  $8 \times 8$  multiplier. ( $\bullet$  represents the exact data, and  $\blacksquare$  represents the approximate data.)

## IV. SIMULATION RESULTS

The proposed approximate Wallace-Booth multiplier is initially designed at gate level in verilog HDL; then it is synthesized by the Synopsys Design Compiler using the Nangate 45nm Open Cell Library. In the simulation of each design, a supply voltage of 1.25V and room temperature are assumed; the power consumption is evaluated by using the Synopsys Power Compiler.

Table III summarizes the delay, area and power consumption of exact and approximate multipliers.

- The first design is the exact Wallace-Booth multiplier (EWBM) [9].
- The second design is the approximate Wallace-Booth multiplier (AWBM-I) that only use the approximate 4-2 compressors in  $n/2$  LSBs.
- The third design is the approximate Wallace-Booth multiplier (AWBM-II) proposed in Section III.

### Delay

AWBM-I has the same number of reduction stages as the exact multiplication; however the use of the approximate 4-2 compressors in  $n/2$  LSBs reduces the delay. In AWBM-II, the approximate tree structure reduces the number of

reduction stages, while the approximate Booth encoding and the approximate 4-2 compressors also contribute to a reduction of the delay. So as shown in Table III, AWBM-II has the best performance in terms of delay.

#### Power Consumption

The power consumption of each multiplier is dependent on the number and type of the Booth encoding cells and the 4-2 compressors. In Table III, AWBM-I has the power consumption significantly lower than EWBM by using the approximate 4-2 compressors. AWBM-II has better performance in power consumption than AWBM-I due to the approximate Booth encoding and the approximate tree.

TABLE III. MULTIPLIER DESIGNS AND COMPARISON (USING NANGATE 45NM OPEN CELL LIBRARY)

<i>N</i> -bit	Type of Multiplier	Delay (ns)	Area ( $\mu\text{m}^2$ )	Power ( $\mu\text{W}$ )	PDP (aJ)
8	EWBM [9]	1.26	699	173	0.218
	AWBM-I	1.13	629	159	0.180
	AWBM-II	1.04	457	110	0.114
16	EWBM [9]	1.53	2615	629	0.962
	AWBM-I	1.26	2183	526	0.663
	AWBM-II	1.19	1613	373	0.444
32	EWBM [9]	1.97	10225	2540	5.003
	AWBM-I	1.63	8920	2208	3.599
	AWBM-II	1.54	6776	1684	2.593

#### Error Distance

Signed approximate multipliers ( $8 \times 8$ ,  $16 \times 16$  and  $32 \times 32$  bit) are analyzed using the NED as the metric [3]. NED is defined as the average distance over all inputs and normalized by the maximum possible error. The maximum high (low) NED is also defined in [8]. AWBM-II multipliers use AMBE in  $n/2$  LSBs, the approximate compressors, and the approximate tree structure. Both AWBM-I and AWBM-II are compared using NED. Table IV shows the NED of the  $8 \times 8$  bit AWBM-I and AWBM-II multipliers.

#### Overall Performance

The product of the power-delay product and the average NED (PDPE) is used as a combined measure of an approximate multiplier. The PDPE of AWBM-I is 0.054, while the PDPE of AWBM-II is 0.025 for the  $8 \times 8$  bit multiplier. AWBM-II using the proposed approximate Booth encoder reduces the PDPE by 53.7% compared with AWBM-I, thus confirming the advantages of the proposed AWBM-II multiplier in terms of overall performance.

TABLE IV. NED OF TWO APPROXIMATE WALLACE-BOOTH MULTIPLIERS FOR  $N=8$ .

Design	Ave NED	Max High NED	Max Low NED	Correct Outputs
AWBM-I	0.30	0.178	0.445	3976
AWBM-II	0.18	0.1018	0.2598	2358

## V. CONCLUSION

The paper has proposed an approximate Wallace-Booth multiplier; this design has shown considerable savings for power consumption and delay while incurring a modest loss in accuracy. Approximate Booth encoders, approximate 4-2 compressors and approximate trees are used in the  $n/2$  LSBs of the multiplier. Compared with an exact signed multiplier, the new approximate multiplier has up to 22% improvement in delay for the  $16 \times 16$  bit scheme and up to 37% improvement in power consumption for the  $8 \times 8$  bit. Schemes employing different types of inexact circuits have been analyzed. The proposed  $8 \times 8$  bit approximate Wallace-Booth multiplier reduces the combined metric of PADNP by over 50% compared with AWBM-I that only uses a previously proposed approximate 4-2 compressor.

## REFERENCES

- [1] S. Venkataramani, S. T. Chakradhar, K. Roy. "Computing approximately, and efficiently," Design, Automation & Test in Europe Conference & Exhibition (DATE), 2015, pp.748-751.
- [2] S. Venkataramani, V. K. Chippa, S. T. Chakradhar. "Quality programmable vector processors for approximate computing," Proceedings of the 46th Annual IEEE and ACM International Symposium on Microarchitecture ACM, 2013, pp.1-12.
- [3] J. Liang, J. Han, and F. Lombardi, "New metrics for the reliability of approximate and probabilistic adders," IEEE Transactions on Computers, vol. 63, pp. 1760-1771, Sep. 2013.
- [4] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Transactions on Circuits and Systems-I: Regular Papers, vol. 57, no. 4, pp. 850-862, Apr. 2010.
- [5] M. J. Schulte and E. E. Swartzlander Jr., "Truncated multiplication with correction constant," in Proc. Workshop on VLSI Signal Processing, 1993, pp. 388-396.
- [6] E. J. King and E. E. Swartzlander Jr., "Data dependent truncated scheme for parallel multiplication," in Proc. 31st Asilomar Conference on Signals, Systems and Computers, 1998, pp. 1178-1182.
- [7] P. Kulkarni, P. Gupta, and M. D. Ercegovic, "Trading accuracy for power in a multiplier architecture," Journal of Low Power Electronics, vol. 7, pp. 490-501, 2011.
- [8] A. Momeni, J. Han, P. Montuschi, and F. Lombardi, "Design and analysis of approximate compressors for multiplication". IEEE Transactions on Computers, 2015, pp. 984- 994.
- [9] W. C. Yeh, and C. W. Jen. "High-speed booth encoded parallel multiplier design." IEEE Transactions on Computers, pp.692- 701, 2000.
- [10] C. Chang, J. Gu, and M. Zhang, "Ultra low-voltage low- power CMOS 4-2 and 5-2 compressors for fast arithmetic circuits," IEEE Transactions on Circuits and Systems-I: Regular Papers, vol. 51, pp. 1985-1997, Oct. 2004.