PROCEEDINGS OF SCIENCE

A Novel Signed Truncated Booth Multiplier based on Probability

Yingran Tan[1]
Tsinghua National Laboratory for Information Science and Technology of Tsinghua University
Beijing, 100084, China
E-mail: `tyr15@mails.tsinghua.edu.cn`

Leibo Liu[2]
Tsinghua National Laboratory for Information Science and Technology of Tsinghua University
Beijing, 100084, China
E-mail: `liulb@tsinghua.edu.cn`

Guiqiang Peng
Tsinghua National Laboratory for Information Science and Technology of Tsinghua University
Beijing, 100084, China
E-mail: `pgq13@mails.tsinghua.edu.cn`

Shouyi Yin
Tsinghua National Laboratory for Information Science and Technology of Tsinghua University
Beijing, 100084, China
E-mail: `yinsy@tsinghua.edu.cn`

Shaojun Wei
Tsinghua National Laboratory for Information Science and Technology of Tsinghua University
Beijing, 100084, China
E-mail: `wsj@mail.tsinghua.edu.cn`

Multiple-input and multiple-output (MIMO) technology has been regarded as the most potential method to achieve both high data rate and high reliability. Novel technology such as internet and mobile network does exert profound effects on nowadays production and living conditions so that data processing is especially important. Meanwhile, multipliers come to critical patches which limit the main frequency, in other words, the throughput rate of integrated circuits, notably in Minimum Mean Square Error (MMSE) algorithm. However, when it comes to signal processing, the accuracy of popular multipliers far exceeds the needs. In this way, a novel truncated multiplier is proposed to meet the demand of decreasing area, reduced power consumption and even shorting the time delay.

ISCC2017
16-17 December 2017
Guangzhou, China

[1] Speaker
[2] Correspongding Author

1. Introduction

With inherent tolerance in the domains of imprecise calculations’ application, the power consumption rather than accuracy is a critical design concern, especially in signal processing domain. When there is no need to retain the whole calculation results, two designs are adopted to do calculation: the post-truncated (P-T) method and the direct-truncated (D-T) method.

Different from P-T as used to calculate accurately the results, D-T is to adopt some methods of estimating to figure out the results [1]. Compared to P-T, although D-T produces a large truncation error, it still has a significant effect on reducing the area and power consumption and shortening the time delay, which is more acceptable for nowadays integrated circuit (IC) design. Due to the noteworthy hardware reduction, the dynamic power dissipation can also be proportionally reduced without resorting to sophisticated power reduction techniques [2].

Numerical calculation includes the fixed-point computation [1]-[8], and the floating-point computation [9]-[10] in MMSE algorithm. From now on, plenty of researches have been done in error-compensated circuits domain in order to improve the accuracy in D-T way. These works have been generalized as Baugh−Wooley (BW) multipliers and Booth multipliers. The Booth multiplier has two remarkable advantages: 1) it can reduce the number of partial products to only about half of the original; 2) the critical-path delay is also less than Baugh−Wooley multiplier (from adaptive low-error fixed-width booth multipliers), which makes this encoding technique widely used in the ASIC-oriented products.

Several works in estimating methods to achieve higher accuracy performance [1]-[8]. A constant-correction truncation (CCT) scheme has been proposed, which has a simple structure and is easy for implementation; however, its problems are also significant [3]. Firstly, the resulting product will have a non-zero DC component. Secondly, to limit the range of the maximum error to be less than a LSB of the data path, the area and power savings cannot be maximized. Based on these problems, a data-dependent variable correction truncation scheme (VCT) has been repersented by using information from the partial product bits of the column adjacent to the truncated least significant bit (LSB) [4]. Then, a pseudo-carry compensated truncation (PCT) scheme has been demostrated by using an adaptive pseudo carry compensation [5]. With an 8 × 8 multiplier, even a 99.4% accuracy for a 16 × 16 multiplication has been shown [6]. While, the performance of different kinds of computing methods such as Wallace, Dadda, Compr. 4:2 have been compared in an accurate and approximate manner [7]. Then, a new approximate Wallace tree multiplier (AWTM) has been presented, which obtained a mean accuracy of 99.85% to 99.965% [8].

This brief puts forward a novel algorithm in compensation by probability statistics (PS). It counts the number of 1 in multiplicator X (in binary), then this number would be used to figure out the probability of 1 in X. Although PS method has lower accuracy compared to MLCP one, the circuit area, timing delay; and the power consumption of PS method is superior to those of MLCP method. Above all, according to the tradeoff between accuracy and circuit area, the PS method provides an excellent performance within the tolerance of accuracy.

The rest of this paper is organized as follows. In Section II, an introduction of Booth encording, decording, and the saving method of bit extension. This novel algorithm and the theoretical analysis basis are given in Section III. Section IV depicts the comparisons of accuracy, area, power and further demonstrates the performance of multiplier with the proposed compensation circuit. Finally, conclusion of this paper is drawn in Section V.

2. Booth Multiplier Implement

Modified Booth encoding is widely used in ASIC-oriented products because of its reduction in the number of partial products [1].

2.1 Based-4 Booth Encoding

The n-bit product P can be expressed in two complement representations as follows:

$$X=\sum_{i=0}^{n-1}(y_i+2^i); \tag{2.1}$$

$$Y=\sum_{i=1}^{n-1}(y_i+2^i)=\sum_{i=1}^{n-1}(y_i+2^i-2\times y_{2i+1})\cdot 2^i; \tag{2.2}$$

$$P=X\times Y=X\times\sum_{i=1}^{n-1}(y_i+2^i-2\times y_{2i+1})\cdot 2^i; \tag{2.3}$$

In this encoding process, when the n is even, the binary multiplicator Y needs to be shifted to left for 1 bit, then the top digit should be repeated twice in the two front bits of Y; otherwise, when the n is odd, it needs to be shifted to left for 1 bit, then the top digit should be repeated only once in the two front bits of Y. Assuming that the number of multiplicator Y is n bits (n is a even integer). It is obviously to find that, the number of partial products of P from the n reduce to (n/2) + 1.

y_{2i+1}	y_{2i}	y_{2i-1}	YE_i
0	0	0	+0
0	0	1	+1
0	1	0	+1
0	1	1	+2
1	0	0	-2
1	0	1	-1
1	1	0	-1
1	1	1	-0

Table 1: Based-4 Booth Encoding

Table 1 lists three concatenated inputs y_{2i+1}, y_{2i}, and y_{2i-1} mapped into YE_i using a Booth encoder. Table 2 shows the partial products with corresponding YE_i for a 12-bit Booth encoder.

YE_i	p_{12i}	p_{11i}	p_{10i}	p_{9i}	p_{8i}	p_{7i}	p_{6i}	p_{5i}	p_{4i}	p_{3i}	p_{2i}	p_{1i}	p_{0i}	E_i	S_i
0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0
+1	x_{11}	x_{11}	x_{10}	x_9	x_8	x_7	x_6	x_5	x_4	x_3	x_2	x_1	x_0	$\overline{x_{11}}$	0
+2	x_{11}	x_{10}	x_9	x_8	x_7	x_6	x_5	x_4	x_3	x_2	x_1	x_0	0	$\overline{x_{11}}$	0
-2	$\overline{x_{11}}$	$\overline{x_{10}}$	$\overline{x_9}$	$\overline{x_8}$	$\overline{x_7}$	$\overline{x_6}$	$\overline{x_5}$	$\overline{x_4}$	$\overline{x_3}$	$\overline{x_2}$	$\overline{x_1}$	$\overline{x_0}$	0	x_{11}	1
-1	$\overline{x_{11}}$	$\overline{x_{11}}$	$\overline{x_{10}}$	$\overline{x_9}$	$\overline{x_8}$	$\overline{x_7}$	$\overline{x_6}$	$\overline{x_5}$	$\overline{x_4}$	$\overline{x_3}$	$\overline{x_2}$	$\overline{x_1}$	$\overline{x_0}$	x_{11}	1
-0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

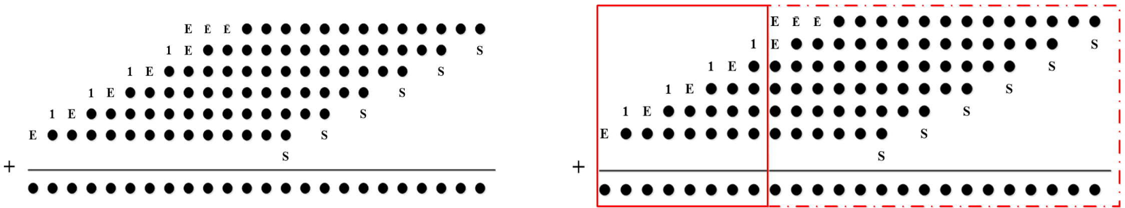
Table 2: Tweleve-bit Booth Decoding

In addition, when the multiplicator Y is encoded, as the top digit repeats twice, the result of this encoded part must be +0 or -0, partial products must be 0 and the bottom line of partial products can be omitted.

2.2 Booth Decoding

Table 2 shows the partial products with corresponding YE_i for a 12-bit Booth encoder. Just shown in Table 2, n+1 bits are set for storing the partial products each row. After encoding, the partial product array with an even width n contains n/2 rows. If the result of Booth encoding of multiplicator Y is positivewe definite S = 1, otherwise, S = 0. Table II shows an example of how to definite the E and S, when n = 12.

2.3 Saving Method of Bit Extension



(a) Partial Product Array

(b) Accurate Part and Estimated Part

Figure 1 : A 12-Bit Booth Multiplier.

Fig. 1 (a) shows the partial product array in a Booth multiplier when n = 12.

Each partial product row needs n+1 bits for partial products, the extended sign bits as brought by Booth encoding, can be simplified as the combination of Es and 1s.

If the multiplicand and the result of Booth encoding of multiplicator have different sign bits, or when the result of Booth encoding is -0, there is a definition of E: E = 0; if the result of Booth encoding of multiplicator have the same sign bit, or when the result of Booth encoding is +0, there is a definite of E: E = 1.

3. New Truncated Multiplier

3.1 Estimation Method based on Probability

The left red frame of Fig. 1 (b) represents the precise calculation of the partial productions by using the Wallace-tree algorithm in this design. Otherwise, the right side of Fig. 1 would rather be estimated by the method, which will be introduced bellow, than be calculated in the interest of obtain the contribution that the right part to the left one. The probability of 1 of the binary multiplicator X is defined as proX, just as the formula below:

$$E_{\bar{X}_i=1}(i=0,1,2,\ldots,n-1)=proX=\frac{\sum_{i=0}^{n-1}X_i}{n}; \tag{3.1}$$

Assuming that zero and one in the binary multiplicand Y have the same probability, and the probability of 1 of the binary multiplicator Y is defined as proY , that is:

$$E[YE_i=1]=proY=0.5; \tag{3.2}$$

The probability of 1 of the partial product pi is defined as $proX$ ($proY=0.5$), just as the formula below

$$proP=proX\times proY=0.5\cdot proX; \tag{3.3}$$

So, when it comes to the situation of multibit, such as p_n to p_k , if the first k bits are supposed to be stored, the possibility of carry which the later one has to the one-bit forward one can be defined as $0.5\times proX\times proY$ (only when the two partial products that equal to 1 meet the one carry will come into being). In this way, each bit is needed to multiply the correlation-weighted indexes as $2^{-(k-j+1)}$:

$$carry_k=\sum_{j=0}^{k-1} proP\times n_j\times 2^{-(k-j+1)} \tag{3.4}$$

The above comprehensive calculation requires estimate of the number of partial products and its location contribution to carry, and in the 12-bit multiplier the result is $proP\times 5.2917$:

$$carry_{16}[n=12]\approx proP\times 5.2917\cdot proP \tag{3.5}$$

3.2 Hardware Implementation

3.2.1 Two Approximation Methods

Considering the hardware design, two approximation methods have been adopted in this multiplier.

Firstly, in the carry estimation, the matlab method has been used to simulate the error rate of this multiplier. There is a distinction from the theoretical value and simulation value. When the weight of X's probability equal to 5.25-5.375, the error rate comes to the minimum 1.6358. In consideration of hardware structure, $5.25\cdot proP$ is used to take the place of the $proP$, which means shifting the multiplicand X two-bit to the left and plus the other two part: the original multiplicand X, and shifting the multiplicand X two-bit to the right.

Secondly, there is a problem that how many bits of the binary $proP$ should be stored at least. Considering the extreme condition, when all bits of X in binary are 1, it means $proP=0.5$ and the result of carry is calculated as 2.625, equaling to 10.101 in binary. However, the only integer part is needed. In this way, it only needs to keep the two bits of the result of $proP$ at least. There are only three conditions for n_x (whichis the number of 1 in X, and n_x = 0,1, ... ,12): when n_x ranges from 0 to 5, it has been figured out that $proP=00$; when it ranges from 6 to 10, $proP=01$; and when it ranges from 11 to 12, $proP=10$.

Both of the two approximation methods also help save areas because of storage.

3.2.2 Other Area Saving Way

In addition to those methods mentioned above, when it comes to the proX generation, MUX-13 has been used at first to implement the divider; however, it needs too large area. Thus this approach has been withdrawn and several OR gates have been used instead.

4. Comparisons and Discussion

This section presents a comparison of the accuracy, area cost, power consumption and computation delay of several Booth multipliers.

4.1 Accuracy

In this brief, the average absolute error $|\bar{\varepsilon}|$ is defined as:

$$|\bar{\varepsilon}|=E[|P-P_q|]/2^L; \tag{4.1}$$

Method	W=1	W=2	W=3	W=L
D-T	0.2511			
PS	1.6358			
MLCP[1]	0.3567	0.2713	0.2559	0.2499

Table 3: Comparison of the $|\bar{\varepsilon}|$ for various methods.

Table 3 shows $|\bar{\varepsilon}|$ for D-T, the proposed PS estimation method, and previous works MLCP respectively [1]. $|\bar{\varepsilon}|$ is the most crucial metric for comparing the accuracy of a multiplier. Table 3 shows that the proposed truncated Booth multiplier sacrifices high performance compared to various multipliers.

In this part, it shows that there is 1.640-accuracy-sacrifice, which certainly has influence on next parts of calculation. Besides, when it comes to many times of iteration, the error rate could be superposed as multiplying. However, as long as the number of iterations is limited, the error rate could be in control and acceptable, especially in nowadays error-tolerance electronic current design.

4.2 Power Consumption Performance

Method		Delay	Area	F-Delay	F-Area
D-T		1.33	2201	0.88	3079
		1.41	2144		
		1.42	2170		
		1.63	2161		
PS		1.33	1908	0.86	3379
		1.41	1904		
		1.42	1930		
		1.63	1835		
MLCP[1]	W=1	1.33	3933	1	6442
	W=2	1.41	4194	0.98	6616
	W=3	1.42	4101	1	5931
	W=L	1.63	5215	1.26	6606

Table 4: Circuit Performance Values.

Method		Delay	Power	F-Delay	F-Power
D-T		1.33	50	0.88	69
		1.41	50		
		1.42	50		
		1.63	53		
PS		1.33	33	0.86	55
		1.41	32		
		1.42	34		
		1.63	32		
MLCP[1]	W=1	1.33	65	1	115
	W=2	1.41	67	0.98	115
	W=3	1.42	66	1	101
	W=L	1.63	86	1.26	117

Table 5: Power Consumption perfomance.

The area cost and the computation delay are critical in Booth multiplier designs. Table 4 lists the area and delay of the proposed truncated Booth multiplier and previous designs with various L and W values, in the case of the fastest one and the delay referred in the paper [1]. The area and delay information was implemented by using the Synopsys design compiler with a TSMC 65-nm CMOS standard cell library to synthesize the RTL design.

Method	W=1	W=2	W=3	W=L
D-T	0.2511			
PS	1.6358			
MLCP[1]	0.3567	0.2713	0.2559	0.2499

All the multipliers are implemented by using the Wallace Tree architecture in Fig. 4 with their own compensated circuit. The reference multiplier is synthesized directly by the DC tool. Obviously, even compared with this, the system can tolerate the accuracy of the sacrifice in exchange for the delay and the area of ascension is definitely huge. The MLCP methods presented used exhaustive simulation to design the compensated circuit, and therefore required a long simulation time to establish compensated circuit [1]. The proposed methods have greatly reduced the simulation time. Although one design outperformed other circuits in $|\overline{\varepsilon}|$ values; at the same times, it has a largest circuit area and delay as well [1].

4.3 Circuit Performance

The power consumption is also a major judgement in Booth multiplier designs. Table 5 reveals the minimum area and the power consumption performance of the proposed estimator and previous designs.

The compensated circuit designs generally include a tradeoff between accuracy and area. The novel truncated Booth method in the accuracy tolerance range reduces the power consumption, calculation delay and circuit area to a great extant. Therefore, the proposed Booth multiplier can fulfill the demands of booth throughput and miniaturization.

5. Conclusion

In this paper a novel truncated Booth multiplier is proposed for inherent tolerance to digital signal processing. Approximate computing appears as a most potential solution to reduce both power and area dissipation as well as the delay. It pays 1.6358 accuracy-sacrifice as cost in return for 11%-13% to the multiplier by DC Synopsys design tool and 51%-64% to MLCP method in area scaling down, at least 2% for time delay decrease, and 36%-39% to the multiplier by DC Synopsys design tool and 48%-64% to MLCP method in power consumption cutting down, which is an excellent breakthrough.

The performances of the proposed circuits are extensive when compared with previous architectures. It is found that the proposed topologies, among most of the investigated cases, are Pareto-optimal, given the best trade-off between accuracy and hardware complexity. We have also presented experimental results and obtained from a simple multiply-accumulate unit implemented with a TSMC 65-nm CMOS standard cell library.

References

[1] Y. H. Chen, "*An Accuracy-Adjustment Fixed-Width Booth Multiplier Based on Multilevel Conditional Probability*, "IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 23, Jun. 2015.

[2] C. H. Chang, and R. K. Satzoda, "*A Low Error and High Performance Multiplexer-Based Truncated Multiplier*," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 18, Dec. 2010.

[3] M. J. Schulte, and E. E. Swartzlander, Jr., "*Truncated Multiplication with Correction Constant,*" in IEEE Workshop VLSI Signal Process. VI, Oct. 1993, pp. 388–396.

[4] E. J. King, and E. E. Swartzlander, Jr., "*Data-Dependent Truncation Scheme for Parallel Multipliers*," in Proc. 31st Asilomar Conf. Signals, Syst. Comput., Nov. 1997, vol. 2, pp. 1178–1182.

[5] B. Parhami, *Computer Arithmetic: Algorithms and Hardware Design*. Oxford, U.K.: Oxford Univ. Press, 2000.

[6] S. Narayanamoorthy, H. A. Moghaddam, and Z. Liu, "*Energy-Efficient Approximate Multiplication for Digital Signal Processing and Classification Applications*," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 23, no. 6, pp. 1180–1184, June. 2015.

[7] G. Zervakis, and K. Tsoumanis, "*Design-Efficient Approximate Multiplication Circuits Through Partial Product Perforation*," IEEE Trans. Very Large Scale Integr. (VLSI) Syst., vol. 24, no. 10, pp. 3105–3117, Oct. 2016.

[8] K. Bhardwaj, P. S. Mane, and J. Henkel, "*Power- and Area-Efficient Approximate Wallace Tree Multiplier for Error-Resilient Systems*," 15th International Symposium on Quality Electronic Design (ISQED), 2014.

[9] H. Thapliyal,and M. B. Srinavas, "*A Novel Time Area-Power Efficient Single Precision Floating Point Multiplier*," Proceeding MAPLD, 2005.

[10] N. Souto, and R. Dinis, "*MIMO Detection and Equalization for SingleCarrier Systems Using the Alternating Direction Method of Multipliers*," IEEE Signal Processing Letters, vol. 23, Issue. 12, pp. 1751–1755, Dec. 2016