Hybrid Low Radix Encoding-Based Approximate Booth Multipliers

Haroon Waris, Chenghua Wang, and Weiqiang Liu, *Senior Member, IEEE*

***Abstract*—Radix-4 Booth encoding provides ease in the generation of partial products, thus is widely used to achieve power-efficient and low-area signed multipliers. Conversely, the radix-8 Booth encoding exhibits low-performance as it requires generation of odd multiples of the multiplicand. In this brief, this issue is addressed by approximating the odd multiples of radix-8 to their nearest power of two such that the errors complement each other. In the pursuit of an accuracy-energy trade-off, hybrid low radix (HLR) based two approximate Booth multipliers (HLR-BM1 and HLR-BM2) are designed. HLR-BM2, compared to the previous best error-optimized design (ABM1), achieved a reduced energy of 22% with a comparable MRED. Moreover, HLR-BM2 achieves an improvement of 75% in MRED and 11% in energy consumption compared to the previously best energy-optimized design (RAD64). As a case study, performance for image transformation is evaluated. A high peak signal-to-noise ratio (PSNR, close to 50dB) is obtained for the proposed multiplier designs.**

***Index Terms*—Approximate computing, radix-4, radix-8, signed multipliers, partial products reduction.**

I. Introduction

THE DESIGN of energy-efficient embedded systems has gained a significant importance in the recent years. This is because a large number of applications require customized hardware with low-power consumption. Conversely, the amount of data which needs to be processed by these hardware modules has increased drastically; therefore, it is becoming difficult to meet the both requirements. Approximate computing has emerged as an alternative to meet the aforementioned challenge [1]. Approximate circuits generally refer to approximate design of arithmetic units such as adders and multipliers. The recent survey [2] states, little research has been reported on the design of approximate Booth multipliers. Therefore, we believe this problem requires more intensive study. Hence this is the target of the work.

Manuscript received January 17, 2020; accepted February 15, 2020. Date of publication February 19, 2020; date of current version November 24, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 61871216, in part by the Six Talent Peaks Project in Jiangsu Province under Grant 2018-XYDXX-009, and in part by the Fundamental Research Funds for the Central Universities China under Grant NE2019102. This brief was recommended by Associate Editor B.-H. Gwee. *(Corresponding author: Weiqiang Liu.)*

The authors are with the College of Electronic and Information Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 211106, China (e-mail: haroonwaris@nuaa.edu.cn; chwang@nuaa.edu.cn; liuweiqiang@nuaa.edu.cn).

Color versions of one or more of the figures in this article are available online at https://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCSII.2020.2975094

For more than a decade fixed-width Booth multipliers have been studied that were designed using truncation-based approach. Truncation achieves significant hardware gain but incurs a large error; therefore, designs incorporating error compensation modules are also presented. In [3], error compensation is pursued using most significant column of the truncation part. Chen and Chang [4] used a probabilistic estimation to derive an error compensation formula for a fixed-width Booth multiplier. Broken Booth multiplier (BBM) [5] is designed by removing the carry-save adders after the specified vertical break length. In [6], hybrid radix-4/-8 truncated multiplier for mobile GPU applications is proposed. A second approach to design approximate Booth multiplies pursued in the technical literature is using an approximate Booth encoder. Jiang *et al.* [7] have proposed an approximate recoding adder that reduced the long carry propagation for radix-8 Booth multiplier. In [8], truth table modifications are used to propose two approximate partial product generators for radix-4 Booth multipliers. Reference [9] proposed approximate high-radix, (radix-64, radix-256, radix-1024), encoders to further reduce the size of partial product matrix. In, radix-4 based approximate Booth multipliers are proposed that approximates the $\pm$2x multiplicand with that of $\pm$1x multiplicand [10]. Recently, a new double high radix encoding (DRAD) is proposed for the multiplication A$\times$B. According to this technique, except for B, A is also encoded, and thus, the design space increases [11].

To the author's best knowledge, [7] is the only research done so far on the design of approximate radix-8 Booth multipliers, despite the fact that, it generates a small number of partial products and thus requires fewer adders to accumulate partial products compared to radix-4. This is because, generation of $\pm$3x multiplicand in a radix-8 encoder requires a preliminary processing; therefore, a higher delay of 10% to 20% is observed than radix-4 algorithm. In this brief, this problem is alleviated by approximating the $\pm$3x multiplicand to its nearest power of two (thus can easily be generated by shifting). Moreover, the said approximation is handled such that errors complement each other, in contrast to the high-radix approximation [9], where error always increases exponentially. Additionally, the concept of hybrid lower-radix encoding is introduced to explore the accuracy-energy trade-off for approximate Booth multipliers. In the proposed approach, least-significant bits of the multiplicand are encoded by the approximate radix-8 encoder and most-significant bits are encoded using exact radix-4 encoder. Later, further reduction in the time is achieved by using Wallace tree

TABLE I
ACCURATE AND APPROXIMATE RADIX-4 BOOTH ENCODERS

$d_{2i+1}d_{2i}d_{2i-1}$	sign	$\times1$	$\times2$	PP_i	R4ABM1	ABM1
000	0	0	0	0	0	0
001	0	1	0	+1C	+1C	+1C
010	0	1	0	+1C	+1C	+1C
011	0	0	1	+2C	0	+1C
100	1	0	1	-2C	0	-1C
101	1	1	0	-1C	-1C	-1C
110	1	1	0	-1C	-1C	-1C
111	1	0	0	0	0	0

for the partial products accumulation. The main contributions of this brief are summarized as follows:

1) The application of radix-8 Booth encoder for the design of energy-efficient approximate multipliers is proposed. In the proposed radix-8 Booth encoder errors complement each other and approximation is handled in such a way that the error distance is always maintained as one.
2) The hybrid low-radix encoding based approximate Booth multipliers are proposed that can be reconfigured per application energy-error trade-off.
3) The proposed HLR-BM2 multiplier outperforms prior state-of-the-art error-energy Pareto fronts, achieving 22% less energy dissipation for comparable error values.

The rest of this brief is organized as follows. Section II presents the exact and existing approximate Booth encoders. The proposed approximate Booth multipliers are described in Section III. In Section IV, evaluation and comparison with the state-of-the-art approximate multipliers are presented. Case study of image transformation is performed in Section V. Finally, Section VI concludes this brief.

II. BOOTH ENCODERS

Booth multiplier consists of Booth encoding, partial product generator, partial product accumulation and final addition. In this section, accurate and existing approximate designs of Booth encoders are presented. Booth multiplier takes two signed inputs C and D of length M and gives an output S_{out} of length 2M.

A. Radix-4 Booth Encoder

In the radix-4 encoder, three consecutive bits $\{d_{2i+1}, d_{2i}, d_{2i-1}\}$ are encoded using three signals sign, $\times1$, $\times2$ and the corresponding partial product is selected from the set ($\pm$ 2C, $\pm$ C or $\pm$ 0) as shown in Table I. Sign signifies the negative partial product, $\times1$ specifies the original partial product, and $\times2$ indicates the shifted partial product. In the technical literature two approximate radix-4 encoders have already been proposed (Table I), in R4ABM1 [8] and ABM1 [10], $\pm$2x multiplicand is replaced by 0 and $\pm$1x, respectively. The said approximation not only simplifies the encoder but also reduce the logic complexity of the partial product generator.

B. Radix-8 and High-Radix Booth Encoders

In the radix-8 encoder, the input D is grouped using overlapping four bits $\{d_{i+2}, d_{i+1}, d_i, d_{i-1}\}$. Sign, $\times1$ and $\times2$ signals are used for the same objective as in radix-4 whereas $\times3$ signifies the need of odd multiplicand and $\times4$ specifies the 4C partial product (Table II). Higher-radix encoding reduces the number of partial products significantly; hence, partial product accumulation can be pursued by simpler tree architectures. Leon *et al.* [9] have proposed approximate high-radix encoders where all values are rounded to their nearest power of two.

TABLE II
ACCURATE AND APPROXIMATE RADIX-8 BOOTH ENCODERS

$d_{i+2}d_{i+1}d_id_{i-1}$	PP_i	R8ABE1	ED	R8ABE2	ED
0000	0	0	0	0	0
0001	+1C	+1C	0	+1C	0
0010	+1C	+1C	0	+1C	0
0011	+2C	+2C	0	+2C	0
0100	+2C	+2C	0	+2C	0
0101	+3C	+2C	1	+2C	1
0110	+3C	+2C	1	+4C	1
0111	+4C	+4C	0	+4C	0
1000	-4C	-4C	0	-4C	0
1001	-3C	-2C	1	-4C	1
1010	-3C	-2C	1	-2C	1
1011	-2C	-2C	0	-2C	0
1100	-2C	-2C	0	-2C	0
1101	-1C	-1C	0	-1C	0
1110	-1C	-1C	0	-1C	0
1111	0	0	0	0	0

Remarks: Partial product generation logic can be simplified by introducing approximation in the Booth encoder. The approximation is to be handled in such a way that the resultant partial product can easily be obtained by shifting. Moreover, the approximation in the radix-8 Booth encoder has not yet been explored for the possible accuracy-energy trade-off.

III. APPROXIMATE BOOTH MULTIPLIERS

In this section, two approximate R8ABEx encoders are proposed and used to design hybrid low radix (HLR) Booth multipliers. 16-bit signed multiplication is considered to explore the effectiveness of hybrid low radix encoding. For the *k* bit multiplier, R8ABEx is used to encode *n* least-significant bits whereas exact radix-4 encoder is used for the remaining *n-k* most-significant bits. The amount of approximation introduced is configurable by the parameter n; therefore, the number of approximate radix-8 encoders to be used in the Booth multiplier can be selected based on the application requirement. Error distance (ED) is a well-known error metric to measure the accuracy of approximate logic circuits and is defined as $ED = |O' - O|$ where, O' and O represents approximate and accurate values, respectively. The said metric is used to characterize the accuracy of proposed approximate radix-8 encoders.

A. HLR-BM1 Booth Multiplier

In HLR-BM1, the nine least significant bits of the multiplicand are encoded using the approximate radix-8 encoder whereas radix-4 encoding is used for the seven most significant bits as shown in Fig. 1(a). This results in seven partial products in total, out of which, three are generated using approximate R8ABE1 whereas remaining four using exact radix-4 encoder. R8ABE1 is designed with a concept to keep the absolute

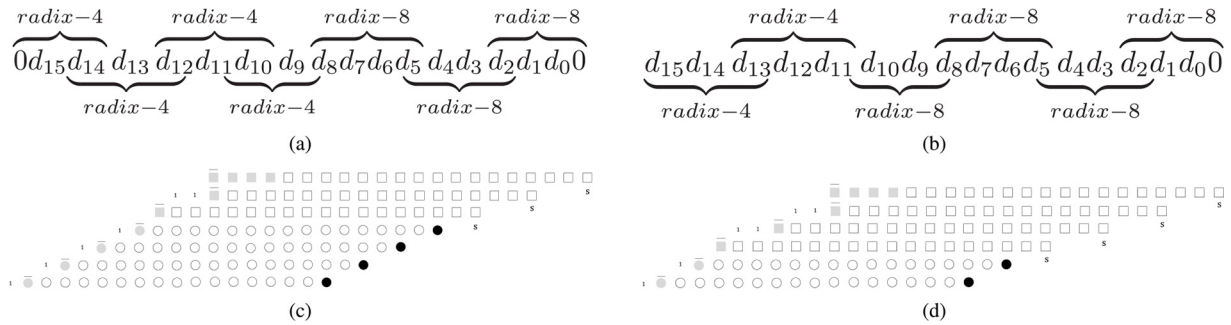

Fig. 1. Hybrid low radix encoding of multiplier bits (a)(c) HLR-BM1 using R8ABE1 (b)(d) HLR-BM2 using R8ABE2. ○: exact partial product bit; □: approximate partial product bit; ●: sign bit; $\bar{\bullet}$: sign bit inversion; s: 0/1 (partial product positive/negative); ●: OR gate result of correction term and least-significant partial product bit.

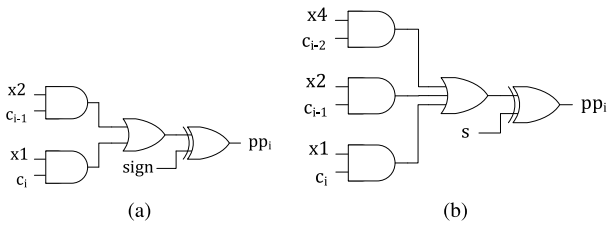

Fig. 2. Partial product generators used in HLR-BMx multipliers based on: (a) exact radix-4 encoder, (b) approximate radix-8 encoder.

TABLE III
PARTIAL PRODUCT MATRIX IN APPROXIMATE BOOTH DESIGNS

Approximate Design	Partial Product Matrix	Error Distance
HLR-BM1	7	1
HLR-BM2	6	1
ABM1 [10]	8	1
RAD64 [9]	6	8
R4ABM1 [8]	8	2

value of approximate product less than the exact counterpart. In R8ABE1, ± 3x multiplicands are approximated to ± 2x multiplicand; therefore, the proposed approximations generate result that is smaller than the exact value. Moreover, to keep the error bounded, error difference is always kept as one against all approximated values (Table II).

B. HLR-BM2 Booth Multiplier

HLR-BM2 is proposed using R8ABE2, where R8ABE2 is designed on the phenomena that positive and negative errors in a logic circuit complement each other [8], [10]. This allows further reducing the height of partial product matrix, the twelve least significant bits of the multiplicand are encoded using the approximate radix-8 encoder whereas remaining four bits are encoded using radix-4 encoders as shown in Fig. 1(b).

R8ABE2 is based on a simple yet efficient approximation approach; therefore, as opposed to R8ABE1, the input combinations are approximated to different values. Inputs {(0101, 1010) and (0110, 1001)} are approximated to ±2 × and ±4 × multiplicand, respectively (Table II). This method complements the errors in the partial product reduction process and still maintains the error difference of one against approximated values. Therefore, if two Booth multipliers are designed using R8ABE1 and R8ABE2, respectively, then the error in R8ABE2 based Booth multiplier will always be less than R8ABE1 based Booth multiplier.

C. Approximate Partial Product Generator

As mentioned in Section II an approximate Booth encoder leads to a design of a simpler partial product generator. For both R8ABE1 and R8ABE2, this approximation simplifies the Booth encoder as ×3 signal is no more needed. Moreover, in the partial product generation, ±3C multiplicand is not required. This leads to a simple radix-8 partial product generator. Fig. 2 shows the simplified partial product generators used for the HLR-BMx multipliers.

D. Partial Product Matrix

The partial product tree for the 16 bit exact (using radix-8 encoder) and proposed HLR-BMx (using hybrid low-radix encoder) multipliers is shown in Fig. 1(c). In the approximate designs, LSBs are encoded using R8ABEx encoders whereas MSBs are encoded using exact radix-4 encoder. Moreover, to further reduce the height of partial product tree, correction term of the exact partial products is handled by OR-ing recoding sign factor with its respective vertical partial product bit, shown as a black filled circle in Fig. 1(d). The exact partial product bit is represented by an empty dot whereas an approximate partial product bit is denoted by square, a gray dot represents the sign bit while the inverting operation is designated by bars on the gray dots (sign extension elimination method is used). The overall multiplication time is reduced by using a Wallace tree for partial product accumulation and the final addition is performed by a prefix (fast) adder.

Table III shows the partial product reduction achieved by the proposed HLR-BMx multipliers. HLR-BM1 has the less number of partial products as compared to R4ABM1 and ABM1. HLR-BM2, DRAD and RAD64 have the similar number of partial products but HLR-BM2 has the least ED, bounded to one. Moreover, the use of R8ABE2, in HLR-BM2, that complements the error makes it a more favorable choice with respect to RAD64.

TABLE IV
COMPARATIVE PERFORMANCE ANALYSIS OF EXACT AND APPROXIMATE BOOTH MULTIPLIERS

Design	Area (um^2)	Delay (ns)	Power (uW)	Energy (uW.ns)	MRED (%)	NMED (10^{-5})	PRED (%)
Exact radix-8	4387	0.85	4825	4101.25	-	-	-
HLR-BM1	2978	0.74	4060	3004.40	0.03	0.79	98.35
HLR-BM2	2651	0.69	3587	2475.30	0.02	0.66	99.12
DRAD (6,6) [11]	2604	0.68	3564	2423.52	0.15	5.33	98.77
ABM1-12 [10]	3054	0.75	4215	3161.25	0.01	0.54	99.68
ABM2-08 [10]	1854	0.70	2964	2074.80	2.68	108.7	86.17
RAD64 [9]	3179	0.71	3840	2726.40	0.08	4.79	99.58
RAD256 [9]	2536	0.67	3287	2202.29	0.28	16.78	98.31
RAD1024 [9]	2412	0.63	3219	2027.97	0.93	44.35	93.26
R4ABM1-12 [8]	3751	0.72	4493	3234.96	0.03	0.84	98.51
R8ABM1 [7]	3958	0.76	4617	3508.92	0.04	1.93	98.95

IV. EVALUATION AND COMPARISON

The proposed HLR-BM1 and HLR-BM2 are evaluated in terms of hardware (area, delay and, power) and error (PRED, NMED and, MRED) metrics. For the fair comparison, both exact radix-8 and state-of-the-art approximate Booth multipliers are considered. Comparison multipliers include R8ABM1 proposed by [7], R4ABM1 proposed by Liu *et al.* [8], RAD multipliers proposed by [9], (ABM1,ABM2) of [10] and DRAD proposed by [11]. R4ABM1 has the better MRED with respect to R4ABM2 [8], thus chosen in our comparison. Similarly, ABM1 is selected as it outperforms design presented in [9] with a lower MRED. It is worth mentioning that the parameter k for R4ABM1-k, ABM1-k and ABM2-k refers to the number of columns in the partial product matrix to which approximation is applied whereas for RAD and DRAD multipliers k refer to the high-radix encoder.

Verilog HDL is used to implement all the examined designs and simulated using Synopsys VCS. Later, Synopsys Design Compiler with a TSMC 45-nm standard cell library is used for the synthesis. SAIF (Switching Activity Interchange Format) and VCD (Value Change Dump) files are used as an input to Synopsys PrimeTime-PX tool for the power estimation. For the error analysis, MATLAB is used to calculate NMED, MRED, PRED of the approximate designs against random 16-bit signed inputs (uniformly-distributed). The NMED is defined as the normalized mean error distance by the maximum output of the accurate design. MRED is the average value of all possible REDs, where, RED is defined as the error distance over the absolute accurate result. PRED is the probability of obtaining a RED smaller than a specific percentage value that is assumed to be 2% throughout this brief.

Table IV shows the hardware and error metrics of proposed HLR-BMx and existing approximate Booth multipliers. HLR-BMx multipliers achieve power savings of 18% to 31% over the exact radix-8 multiplier. HLR-BM1 has the least delay due to the reduced partial product tree compared to ABM1 and R8ABM1 designs. Moreover, HLR-BM1 has the power improvement of 7% to 14% with its comparative designs having similar error metrics. HLR-BM2 has the lesser NMED than HLR-BM1 because R8ABE2 introduce both positive and negative errors. The errors complement each other and HLR-BM2 realized better error metrics. An image transformation result

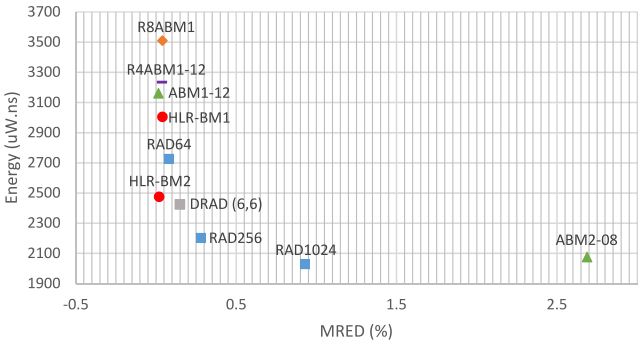

Fig. 3. MRED-Energy trade-off of approximate multipliers.

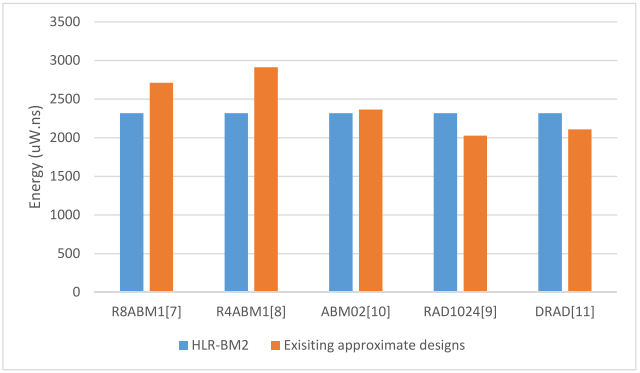

Fig. 4. Energy gains of the HLR-BM2 w.r.t. previous state-of-the-art multipliers.

shown in Section V further demonstrates this phenomena. Fig. 3 shows the comparative performance of all approximate Booth multipliers by considering both energy and MRED. ABM1 has the least MRED among all proposed designs as ± 2 x multiplicand values are replaced by $\pm$ 1 values. However, HLR-BM1 having a comparable MRED to the previous best design (ABM1) has less energy consumption. Moreover, HLR-BM2 achieves reduced energy of 22% compared to the previous best error-optimized design, i.e., ABM1. Similarly, HLR-BM2 has an improvement of 75% and 11% in MRED and energy, respectively, with reference to the previous best energy-optimized design (RAD64).

DRAD shows a reduced energy consumption compared to previously Pareto front formed by RAD64 whereas with respect to HLR-BM2 it has a higher MRED. The reason to

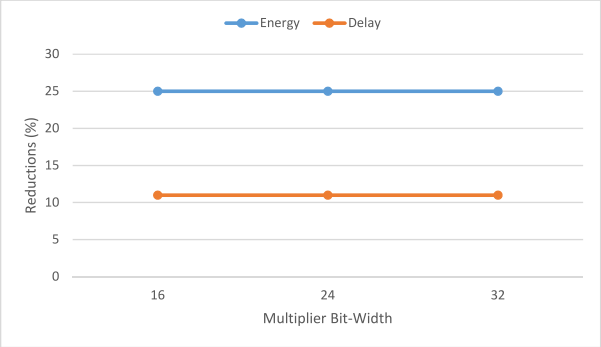

Fig. 5. Energy and delay gains for scale-up variants of HLR-BM2.

(a) (b) 0.95/50.09 (c) 0.96/51.16

(d) 0.97/53.78 (e) 0.85/39.46 (f) 0.91/48.31

Fig. 6. SSIM/PSNR of output images using (a) Accurate multiplier, (b) HLR-BM1, (c) HLR-BM2, (d) ABM1-12, (e) RAD256, (f) R4ABM1.

this lower energy consumption can be associated to the fact as, in this technique, except for B, A is also encoded and rationale behind higher MRED is, DRAD does not have an error compensation property; therefore, as approximation factor is increased, error increases. However, HLR-BM2 encoder has an error compensation phenomenon which keeps MRED small. Fig. 4 shows the energy gains of HLR-BM2 with respect to previous state-of-the art multipliers for large-error values. We select 0.93% (RAD1024) as an error bound for all multipliers. HLR-BM2 performs better from all designs except RAD1024 and DRAD. The use of high-radix encoders in RAD1024 and DRAD generates less number of partial products; therefore, for large-error values these designs show better error-energy tradeoff. The scaling behavior of HLR-BM2 is also analyzed; scaled-up variants of HLR-BM2 (16-bit and 24-bit) have a constant improvement in energy savings and delay of 25% and 11%, respectively, while maintaining small MRED as shown in Fig. 5.

V. Case Study: Image Transformation

Image brightening is a practical signal processing application, thus pursued here to evaluate the proposed approximate multipliers. 16-bit image is considered for that purpose, the pixel values are uniformly shifted from [0, 65535] to [−32768, 32767]. A predefined constant is multiplied by an input image (pixel-by-pixel) to get the transformed (brighten) image. PSNR and structural similarity index (SSIM) metrics are used to assess the output image quality. Fig. 6 shows the output images processed using different approximate multipliers. RAD256 has the worst quality metrics as it has a large NMED. R4ABM1 approximates ± 2 x multiplicand to zero; therefore, has a smaller SSIM. The proposed approximate Booth multipliers achieve a PSNR close to 50dB; however, HLR-BM2 has a better PSNR compared to HLR-BM1.

VI. Conclusion

In this brief, we have proposed two hybrid low radix based approximate Booth multipliers (HLR-BM1 and HLR-BM2). For the k bit multiplier, the n least significant bits are encoded using approximate radix-8 encoder whereas the exact radix-4 encoder is used for the remaining n-k most-significant bits. The so-called approximate radix-8 encoders (R8ABE1 and R8ABE2) are purposely designed with an aim to keep the error minimal. Moreover, the use of hybrid approach, to use both exact radix-4 and approximate radix-8 encoders resulted in the reduction of partial product tree. Compared with the state-of-the-art approximate designs, HLR-BMx multipliers shows a better error and hardware characteristics, an improvement of 75% and 22% in MRED and energy are achieved. Image transformation is used to evaluate the proposed multipliers, a PSNR greater than 50dB is achieved; therefore, can be used in error-resilient applications.

References

[1] S. Xu and B. C. Schafer, "Toward self-tunable approximate computing," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 4, pp. 778–789, Apr. 2019.

[2] H. Jiang, C. Liu, L. Liu, F. Lombardi, and J. Han, "A review, classification, and comparative evaluation of approximate arithmetic circuits," *ACM J. Emerg. Technol. Comput. Syst. (JETC)*, vol. 13, no. 4, pp. 1–34, 2017.

[3] J.-P. Wang, S.-R. Kuang, and S.-C. Liang, "High-accuracy fixed-width modified Booth multipliers for lossy applications," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 19, no. 1, pp. 52–60, Jan. 2011.

[4] Y.-H. Chen and T.-Y. Chang, "A high-accuracy adaptive conditional-probability estimator for fixed-width Booth multipliers," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 59, no. 3, pp. 594–603, Mar. 2012.

[5] F. Farshchi, M. S. Abrishami, and S. M. Fakhraie, "New approximate multiplier for low power digital signal processing," in *Proc. 17th CSI Int. Symp. Comput. Architect. Digit. Syst. (CADS)*, 2013, pp. 25–30.

[6] S. Choi, G. Kim, H.-J. Yoo, and B.-G. Nam, "Hybrid radix-4/-8 truncated multiplier for mobile GPU applications," *Electron. Lett.*, vol. 50, no. 23, pp. 1680–1682, Nov. 2014.

[7] H. Jiang, J. Han, F. Qiao, and F. Lombardi, "Approximate radix-8 Booth multipliers for low-power and high-performance operation," *IEEE Trans. Comput.*, vol. 65, no. 8, pp. 2638–2644, Aug. 2016.

[8] W. Liu, L. Qian, C. Wang, H. Jiang, J. Han, and F. Lombardi, "Design of approximate radix-4 Booth multipliers for error-tolerant computing," *IEEE Trans. Comput.*, vol. 66, no. 8, pp. 1435–1441, Aug. 2017.

[9] V. Leon, G. Zervakis, D. Soudris, and K. Pekmestzi, "Approximate hybrid high radix encoding for energy-efficient inexact multipliers," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 3, pp. 421–430, Mar. 2018.

[10] S. Venkatachalam, E. Adams, H. J. Lee, and S.-B. Ko, "Design and analysis of area and power efficient approximate Booth multipliers," *IEEE Trans. Comput.*, vol. 68, no. 11, pp. 1697–1703, Nov. 2019.

[11] V. Leon, K. Asimakopoulos, S. Xydis, D. Soudris, and K. Pekmestzi, "Cooperative arithmetic-aware approximation techniques for energy-efficient multipliers," in *Proc. 56th ACM/IEEE Design Autom. Conf. (DAC)*, Jun. 2019, pp. 1–6.