

Approximate adder design with simplified lower-part approximation

Jungwon Lee¹, Hyoju Seo¹, Yerin Kim¹, and Yongtae Kim^{1, a)}

Abstract This letter presents a novel approximate adder that reduces energy and power consumption by leveraging a simplified lower-part approximation. The proposed scheme reduces hardware costs while providing an acceptable accuracy performance. Implemented in a 32-*nm* CMOS technology, the proposed adder achieves area and power reductions of 67% and 91%, respectively, compare to a conventional adder. In terms of energy, it improves the power-delay and energy-delay products by 13.1% and 17.0%, respectively, compared to the other approximate adders considered herein. In addition, when adopted in a digital image processing application, the proposed adder shows a very promising output quality compared to that produced by an exact adder while providing excellent energy efficiency. **Keywords:** approximate adder, error tolerant adder, energy efficiency **Classification:** Integrated circuits (memory, logic, analog, RF, sensor)

1. Introduction

Addition is one of the heavily used arithmetic operations in many applications, and adders consume a significant amount of power and energy, which leads to hot-spot locations on processors [1]. Computationally intensive applications, such as image processing, machine learning, and data mining, may have inherent error tolerance, and a certain amount of computation error is acceptable in these applications [2, 3, 4, 5, 6]. Therefore, the design of efficient approximate adders that reduce power and energy has drawn great attention, and a large number of approximate adders have been proposed in the recent years [7, 8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20, 21, 22, 23, 24, 25, 26, 27, 28, 29].

The lower-part OR adder (LOA) divides an adder into two: accurate and inaccurate parts [7]. The accurate part uses a precise adder, such as a ripple carry adder (RCA), for normal additions on higher-order bits, whereas the inaccurate part exploits an OR operation to approximately add lower-order input bits. The optimized lower-part constant OR adder (OLOCA) is further improved the LOA by outputting "1" on a few least significant bit (LSB) outputs of the inaccurate part, regardless of the input [8]. The error tolerant adder I (ETAI) is also split into two parts as the LOA and OLOCA do. The key difference with the LOA stems from the inaccurate part where a modified XOR operation is used [9]. Additionally, the ETAI does not include any carry prediction scheme for the precise adder, whereas the LOA does an AND-based carry prediction, which enhances

DOI: 10.1587/elex.17.20200218 Received June 18, 2020 Accepted June 29, 2020 Publicized July 14, 2020 Copyedited August 10, 2020 the overall accuracy. Hence, a couple of ETAI variants that include a carry prediction technique to improve the accuracy were proposed in [10, 11]. The ETAI sequentially checks the input bits from the most significant bit (MSB) position to the LSB; therefore, the delay of the inaccurate part is a bit longer than that of the LOA and its variants [7, 8, 12]. In other words, the critical path delay will likely exist in the inaccurate part when its size is larger than the accurate part, resulting in overall speed degradation and poor energy efficiency.

This letter proposes a new energy and power efficient adder by simplifying the lower-part approximation of the ETAI to reduce hardware cost while maintaining a good computation accuracy. We systematically analyze the proposed design to obtain an optimal adder configuration, and extensively compare our design with other adders in terms of accuracy and hardware performance.

2. Proposed approximate adder design

Fig. 1 shows the general architecture of the proposed *n*-bit adder, termed the simplified error tolerant adder (SETA).



Fig. 1 General architecture of the proposed approximate adder.

The *n*-bit adder consists of a *k*-bit precise adder to accurately add *k* MSB input (*i.e.*, $S_{n-1:n-k} = A_{n-1:n-k} + B_{n-1:n-k}$) and an approximation logic simplified from the ETAI for the remaining lower-order n - k inputs. The lower-part (*i.e.*, inaccurate part) addition is basically achieved through the OR operation, which is similar to that of the LOA and ETAI. The proposed approximation logic checks the two (*i*)th input bits only by using an AND gate. If both bits are "1" (*i.e.*, $A_i = B_i = 1$), then the lower-order output from the $(i - 1)^{th}$ to (0)th bit positions (*i.e.*, $S_{i-1:0}$) are set to "1"; otherwise, the output does not change and keeps the OR gate output (*i.e.*, $S_{n-k-1:0} = A_{n-k-1:0}$ OR $B_{n-k-1:0}$). Note that the bit position *i* can be anywhere in $n-k-1 \le i < 0$. We will seek an optimal position by systematically examining the accuracy and circuit performance in Section 3.

¹ School of Computer Science and Engineering, Kyungpook National University, Daegu 41566, South Korea

^{a)} yongtae@knu.ac.kr

The ETAI successively investigates the input bits from the MSB to LSB of the inaccurate part; thus, its delay $t_{inaccurate, ETAI}$ is obtained by [10]

$$\mathbf{t}_{\text{inaccurate,ETAI}} = t_{AND} + (n - k - 3) \cdot t_{OR}.$$
(1)

where *n* is the size of the entire adder, *k* is that of the precise adder, and t_{AND} and t_{OR} are the delays of a two-input AND gate and an OR gate, respectively.

In the proposed adder, only two gate delays are required to produce the output of the inaccurate part. As a result, the delay $t_{inaccurate,SETA}$ is further reduced by

$$\mathbf{t}_{\text{inaccurate,SETA}} = \max(t_{AND}, t_{OR}) + t_{OR}.$$
 (2)

The proposed SETA produces an incorrect output when both inputs of any bit position of the inaccurate part are "1". Hence, the error rate (ER) of the proposed adder under random input patterns \mathbf{ER}_{SETA} is obtained by

$$\mathbf{ER}_{\mathbf{SETA}}(n,k) = 1 - \left(\frac{3}{4}\right)^{n-k}.$$
 (3)

 ER_{SETA} is independent of *i*, and the ERs of the LOA, ETAI, and the proposed adder are identical because an error occurs at the same condition.

3. Experimental Results

The proposed adder together with an accurate adder (RCA) and other three approximate adders (LOA, OLOCA, and ETAI) were designed in Verilog HDL and synthesized with a 32-*nm* CMOS technology to determine the circuit performance in terms of area, delay, and power. The 16-bit adders were implemented using an 8-bit RCA based precise adder (*i.e.*, n=16, k=8). We also developed a simulator to evaluate the accuracy by extracting error metrics including the ER and normalized mean error distance (NMED). These metrics were estimated under 10 million (*i.e.*, 10^7) input pairs, each of which was uniformly generated random number.

 Table I
 Power, NMED, and power-NMED product (PNP) with various i.

i	7	6	5	4	3
Power (μW)	30.60	30.53	30.45	30.38	30.31
NMED (1 <i>e</i> -3)	2.81	2.87	2.90	2.92	2.93
PNP (1 <i>e</i> -3)	86.08	87.71	88.42	88.67	88.87

First, we varied the design parameter *i* from 7 to 3 to seek the best tradeoff between accuracy and power. Table I shows the performance of the proposed approximate adder with various values of *i*. The power decreases as *i* decreases because a smaller *i* requires less OR gates. In contrast, the accuracy degrades. Note that the delay and ER does not change as *i* varies. The power-NMED product (PNP) can be used to jointly evaluate the accuracy and power of the adder [30]. According to the product in Table I, the adder shows the best accuracy and power tradeoff at *i* = 7. Hence, our design with *i* = 7 is considered for comparison with the other adders. Note that they have a similar trend with *i* when *k* varies.

Table II summarizes the performances of the proposed adder and the other four adders. The RCA has the longest

Table II Performance summary of various adders with *n*=16 and *k*=8.

Design	Area (µm ²)	Delay (ns)	Power (µW)	ER (%)	NMED (1e-3)
RCA	190.4	1.79	58.5	-	-
LOA	115.8	0.88	33.4	89.99	1.71
OLOCA	102.1	0.88	30.9	99.12	1.77
ETAI	131.2	0.85	33.5	89.99	2.74
SETA	114.2	0.85	30.6	89.99	2.81

delay caused by the bit-by-bit carry propagation and consumes the most area and power. The lack of carry prediction allows the ETAI and SETA to be better in speed than the LOA and OLOCA; however, it degrades the NMED performance. The LOA occupies slightly more area than the SETA because the prediction requires a full adder at the LSB of the accurate part, whereas a half adder is formed at the corresponding bit in the SETA. Although the ER of the OLOCA reaches over 99%, the NMED is similar to the LOA. The accuracy of our adder is comparable to that of the ETAI, and it exhibits a better performance in area and power.

We obtain the power-delay product (PDP) and energydelay product (EDP) to further evaluate the energy efficiency of our adder when compared with those of the other adders.



Fig. 2 PDP and EDP of the approximate adders with n=16 and k=8.

Fig. 2 shows the PDP and EDP of the proposed and three approximate adders. The LOA exhibits the worst energy efficiency, whereas our design is the most efficient. The OLOCA is comparable to the ETAI in both the PDP and EDP. The proposed adder improves the PDP and EDP by 13.1% and 17.0%, respectively, over the LOA. Clearly, our adder outperforms the others in terms of energy efficiency.

We applied an exact adder and our adder to a 5×5 Gaussian filter to build an accurate and an inaccurate filters and determine the impact of the approximation errors on real applications [23]. The peak signal-to-noise ratio (PSNR) is commonly used to evaluate the output image quality [19, 20], and was measured with the two adders. Fig. 3 depicts the output images with an exact adder (*e.g.*, RCA) and the proposed adder. The PSNRs of the output processed by both adders are greater than 20dB. The exact and proposed adders produce visually indistinguishable output images, proving that the approximation errors generated by our adder negligibly affect the quality of the digital image processing application.

4. Conclusions

This letter proposed a new energy efficient approximate adder that adopts a simplification of the existing approximation scheme to reduce hardware costs. Implemented in a 32-*nm* CMOS process, our adder reduced the area and



Fig. 3 Original image and Gaussian filter output with two adders.

power by 67% and 91%, respectively, over the RCA. Furthermore, the proposed design showed 13.1% and 17.0% better PDP and EDP, respectively, compared to the LOA. The effectiveness of the proposed adder was shown through a digital image processing application. Therefore, the proposed adder is highly suitable for designing energy efficient applications.

Acknowledgments

This research was supported in part by Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Education (NRF-2019R1I1A3A01061266) and in part by the BK21 Plus Project (SW Human Resource Development Program for Supporting Smart Life) funded by the Ministry of Education, School of Computer Science and Engineering, Kyungpook National University, Korea (21A20131600005).

References

- S. Ghosh, *et al.*: "Voltage salable high-speed robust hybrid arithmetic units using adaptive clocking," IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 18 (2010) 1301 (DOI: 10.1109/TVLSI.2009.2022531).
- [2] S. Mittal: "A survey of techniques for approximate computing," ACM Comput. Survey 48 (2016) 62 (DOI: 10.1145/2893356).
- [3] J. Chen, *et al.*: "A new area and power efficient DCT circuits using sporadic logarithmic shifters," IEICE Electron. Express **16** (2019) 1 (DOI: 10.1587/elex.16.20190317).
- [4] Y. Kim, et al.: "A reconfigurable digital neuromorphic processor with memristive synaptic crossbar for cognitive computing," ACM J. Emerg. Technol. Comput. Syst. 11 (2015) 38 (DOI: 10.1145/ 2700234).
- [5] S. Zhang, *et al.*: "Thread: towards fine-grained precision reconfiguration in variable-precision neural network accelerator," IEICE Electron. Express 16 (2019) 1 (DOI: 10.1587/elex.16.20190145).
- [6] J. Chang and J. Sha: "An efficient implementation of 2D convolution in CNN," IEICE Electron. Express 14 (2017) 1 (DOI: 10.1587/elex. 13.20161134).
- [7] H. Mahdiani, *et al.*: "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Trans. Circuits Syst. I, Reg. Papers **57** (2010) 850 (DOI: 10.1109/ TCSI.2009.2027626).
- [8] A. Dalloo, *et al.*: "Systematic design of an approximate adder: the optimized lower part constant-OR adder," IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **26** (2018) 1595 (DOI: 10.1109/TVLSI. 2018.2822278).
- [9] N. Zhu, et al.: "Design of low-power high-speed truncation-errortolerant adder and its application in digital signal processing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst. 18 (2010) 1225 (DOI: 10.1109/TVLSI.2009.2020591).
- [10] Y. Kim: "An accuracy enhanced error tolerant adder with carry prediction for approximate computing," IEIE Trans. Smart Process. Comput. 8 (2019) 324 (DOI: 10.5573/IEIESPC.2019.8.4.324).
- [11] Y. Kim: "A novel approximate adder with enhanced low-cost carry

prediction for error tolerant computing," IEIE Trans. Smart Process. Comput. **8** (2019) 506 (DOI: 10.5573/IEIESPC.2019.8.6.506).

- [12] H. Seo, *et al.*: "Design and analysis of approximate adder with hybrid error reduction," Electronics **9** (2020) 471 (DOI: 10.3390/ electronics9030471).
- [13] M. Pashaeifar, *et al.*: "Approximate reverse carry propagation adder for energy-efficient DSP applications," IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **26** (2018) 2530 (DOI: 10.1109/TVLSI.2018. 2859939).
- [14] W. Liu, *et al.*: "Design and analysis of inexact floating-point adders," IEEE Trans. Comput. **65** (2016) 308 (DOI: 10.1109/TC.2015. 2417549).
- [15] K. Du, et al.: "High performance reliable variable latency carry select addition," Proc. Design, Autom., Test Eur. (DATE) (2012) 2157 (DOI: 10.1109/DATE.2012.6176685).
- [16] A. Kahng and S. Kang: "Accuracy-configurable adder for approximate arithmetic designs," Proc. Design Autom. Conf. (DAC) (2012) 820 (DOI: 10.1145/2228360.2228509).
- [17] M. Shafique, *et al.*: "A low latency generic accuracy configurable adder," Proc. Design Autom. Conf. (DAC) (2015) 86 (DOI: 10.1145/2744769.2744778).
- [18] S.-L. Lu: "Speeding up processing with approximation circuits," Computer 37 (2004) 67 (DOI: 10.1109/MC.2004.1274006).
- [19] O. Akbari, *et al.*: "RAP-CLA: a reconfigurable approximate carry loo-ahead adder," IEEE Trans. Circuits Syst. I, Reg. Papers 65 (2018) 1089 (DOI: 10.1109/TCSII.2016.2633307).
- [20] F. Ebrahimi-Azandaryani, et al.: "Block-based carry speculative approximate adder for energy-efficient applications," IEEE Trans. Circuits Syst. II, Exp. Briefs 67 (2020) 137 (DOI: 10.1109/TCSII.2019. 2901060).
- [21] Y. Kim, *et al.*: "An energy efficient approximate adder with carry skip for error resilient neuromorphic VLSI systems," Proc. Int. Conf. Comput.-Aided Design (ICCAD) (2013) 130 (DOI: 10.1109/ICCAD. 2013.6691108).
- [22] Y. Kim, *et al.*: "Energy efficient approximate arithmetic for error resilient neuromorphic computing," IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **23** (2015) 2733 (DOI: 10.1109/TVLSI.2014. 2365458).
- [23] T. Sato, *et al.*: "Trading accuracy for power with a configurable approximate adder," IEICE Trans. Electron. E102-C (2019) 260 (DOI: 10.1587/transele.2018CDP0001).
- [24] H. Waris, *et al.*: "High-performance approximate half and full adder cells using NAND logic gate," IEICE Electron. Express **16** (2019) 1 (DOI: 10.1587/elex.16.20190043).
- [25] T. Yang, et al.: "An accuracy-configurable adder for low-power applications," IEICE Trans. Electron. E103-C (2020) 68 (DOI: 10.1587/ transele.2019LHP0002).
- [26] P. Balasubramanian and D.L. Maskell: "Hardware optimized and error reduced approximate adder," Electronics 8 (2019) 1212 (DOI: 10.3390/electronics8111212).
- [27] B.K. Mohanty: "Efficient fixed-width adder-tree design," IEEE Trans. Circuits Syst. II, Exp. Briefs 66 (2019) 292 (DOI: 10.1109/TCSII. 2018.2849214).
- [28] N.-C. Huang, *et al.*: "Sensor-based approximate adder design for accelerating error tolerant and deep-learning applications," Proc. Design, Autom., Test Eur. (DATE) (2019) 692 (DOI: 10.23919/DATE.2019.8714949).
- [29] D. Celia, *et al.*: "Optimizing power-accuracy trade-off in approximate adders," Proc. Design, Autom., Test Eur. (DATE) (2018) 1488 (DOI: 10.23919/DATE.2018.8342248).
- [30] J. Liang, *et al.*: "New metrics for the reliability of approximate and probabilistic adders," IEEE Trans. Comput. **62** (2013) 1760 (DOI: 10.1109/TC.2012.146).