Accuracy-Configurable Radix-4 Adder With a Dynamic Output Modification Scheme

Kun-Lin Tsai, *Member, IEEE*, Yen-Jen Chang, *Member, IEEE*, Chien-Ho Wang, and Cheng-Tse Chiang

***Abstract*— Approximate computing is an efficient approach for reducing computational costs. This method involves a trade-off between computational accuracy and the circuit's power consumption, delay, and area. However, the accuracy requirements may differ for different applications. In some situations, precise results are required. Therefore, this paper proposes an accuracy-configurable radix-4 adder (ACRA), which uses the power gating technique to turn on or turn off the partial logic gates of an adder element dynamically to compute accurate or approximate results. When the ACRA operates in the approximate mode, the partial sum of one adder element is modified to reduce the error distance between the approximate and accurate results. A comparison of the ACRA with two state-of-the-art accuracy-configurable adders indicated that the ACRA achieved the best trade-off between the power–delay product and computational accuracy. In addition, in the image processing experiment conducted in this study, under half-approximate computation, the peak signal-to-noise ratio of the ACRA was less than 1 dB and its structural similarity index measure was maintained above 0.99. These results are superior to those obtained for other accuracy-configurable adders.**

***Index Terms*— Accuracy-configurable, approximate computing, image quality, power gating, radix-4 adder.**

I. Introduction

AN INCREASING number of image-related applications (e.g., image processing, image recognition, machine learning, and digital signal processing), many of which require considerable computational effort [1]–[3], have been developed recently. The two major circuit design challenges faced by these applications are power consumption and performance. However, in most cases, due to the limited human perception ability, inaccurate calculations can also generate meaningful results for humans [4]. Approximate calculation [5]–[7], which is a technology used in error-tolerant applications, is a design method that involves a trade-off between the design cost and the computational accuracy. By simplifying the circuit and sacrificing the computational accuracy, approximate calculation can reduce the circuit's power consumption [8], delay time [9], and area [10] and thus, address the drawbacks of traditional computations.

An adder, which is a basic component of arithmetic operations, is widely used in integrated circuits. For an error-tolerant application, an approximate adder can be used to generate acceptable results. Many studies [10]–[14] have designed approximate adders by adopting a fixed accuracy method. However, these adders cannot be used to calculate accurate values and their accuracy cannot be changed. Moreover, despite their benefits, such as small circuit area and high performance, approximate adders face difficulty in satisfying the different accuracy requirements of different applications [15]. For example, image compression requires relatively low computational accuracy, whereas image recognition requires high computational accuracy [16]. In addition, in some cases, a general-purpose processor in a digital system must be able to perform both approximate and accurate computations [17]. In certain cases, the accuracy required is lower than that set by the adder, which can cause unnecessary energy consumption [1]. In addition, devices with a power budget must be able to adjust dynamically the power consumed by calculations.

Compared with the ripple carry adder (RCA) and carry-lookahead adder (CLA), which calculate one bit at a time, the radix-4 adder (RD4A) [11], [18] calculates two bits simultaneously to improve the power–delay product (PDP). Therefore, this paper proposes an accuracy-configurable RD4A (ACRA) to reduce the circuit's power consumption in approximate calculations. In the ACRA, the power gating technique is used to carry-in related logic gates and the partial sum of one RD4A is dynamically modified to minimize the error distance between the approximate and accurate results. The ACRA operates in two modes: the accurate and approximate modes; thus, depending on the application type, the ACRA can generate an accurate or approximate result in the run time. The results of the experiments conducted using the TSMC 45-nm CMOS technology indicated that the ACRA exhibited a lower power consumption and propagation delay time than did the RCA and CLA. Moreover, according to the obtained peak signal-to-noise ratio (PSNR) and structural similarity index measure (SSIM) for image processing, the ACRA significantly outperformed other accuracy-configurable adders.

The contributions of this paper are as follows.

1) The ACRA can be configured as an accurate or approximate adder by controlling the power supply of specific logic gates in the modified RD4A elements.
2) In the approximate mode, the computational results of the ACRA can be controlled within a small error

Manuscript received March 7, 2021; revised May 8, 2021; accepted May 24, 2021. Date of publication June 14, 2021; date of current version July 13, 2021. This work was supported in part by the Ministry of Science and Technology, Taiwan, under Grant MOST 108-2221-E-005-018. This article was recommended by Associate Editor T. Hanyu. *(Corresponding author: Yen-Jen Chang.)*

Kun-Lin Tsai is with the Department of Electrical Engineering, Tunghai University, Taichung 407224, Taiwan, and also with the Research Center for Smart Sustainable Circular Economy, Tunghai University, Taichung 407224, Taiwan.

Yen-Jen Chang, Chien-Ho Wang, and Cheng-Tse Chiang are with the Department of Computer Science and Engineering, National Chung Hsing University, Taichung 402202, Taiwan (e-mail: ychang@cs.nchu.edu.tw).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSI.2021.3085572.

Digital Object Identifier 10.1109/TCSI.2021.3085572

distance, which is suitable for image processing-related applications.

3) The computation result obtained by the ACRA exhibits an outstanding trade-off among power consumption, propagation delay time, and error distance.

The rest of this paper is organized as follows. The related studies and RD4A are described in section II. The circuit structure of the ACRA is introduced in section III. The experimental results are presented in section IV. Finally, the conclusions of this study are provided in section V.

II. Preliminary

This section first introduces the RD4A and then describes the related studies.

A. RD4A

The RD4A, which operates two bits simultaneously, reduces the propagation delay and power consumption of a traditional adder. Fig.1 presents the architecture of a conventional 2-bit RD4A, which has five inputs–C_{in}, A_i, A_{i+1}, B_i, and B_{i+1}–and three outputs–C_{out}, Sum_i, and Sum_{i+1}. The parameter C_{in} represents the carry input of the RD4A; A_i and A_{i+1} represent the first and second bits of augend A, respectively; and B_i, and B_{i+1} represent the first and second bits of addend B, respectively. The parameters C_{out}, Sum_i, and Sum_{i+1} indicate the carry output and 2 adjacent sum bits of the RD4A, respectively. According to the design rule of the RD4A, the output Boolean function can be formularized as presented in Eq.(1) to Eq. (3). To reuse the logic gate, as displayed in Fig. 1, an XOR gate is divided into one AND gate and two NOR gates.

$$C_{out} = A_{i+1}B_{i+1} + (A_iB_i)(A_{i+1} + B_{i+1}) + C_{in}((A_i + B_i)(A_{i+1} + B_{i+1})) \quad (1)$$

$$Sum_{i+1} = (A_{i+1} \oplus B_{i+1}) \oplus (A_iB_i + C_{in}A_i + C_{in}B_i) \quad (2)$$

$$Sum_i = (A_i \oplus B_i) \oplus C_{in} \quad (3)$$

B. Related Studies

Many studies have been conducted on approximate adders. For example, Seo *et al.* [14] proposed an energy-efficient approximate adder with the hybrid error reduction scheme to increase the computational accuracy and performance of a power circuit. This adder is divided into two parts: the precise and approximate parts; however, its partition has a fixed design and its accuracy cannot be configured at run time.

Design methods for an accuracy-adjustable adder can be classified into two types: methods based on error correction and carry prediction. Accuracy-adjustable adders based on error correction, (e.g., [16]) use an approximate adder to calculate an inaccurate result; then, they use an error correction circuit to correct the inaccurate result. However, such a design has some disadvantages. To improve the accuracy of an approximate adder, several overlapping subadders, which are used for error detection and correction, are added to it. The error correction process is performed from the least significant bit (LSB) to the most significant bit (MSB) so that the computational result can be fixed gradually.

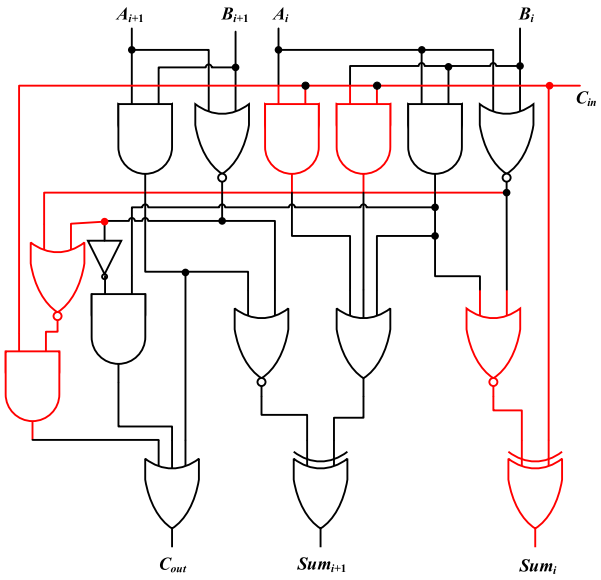

Fig. 1. The architecture of a conventional 2-bit Radix-4 adder.

However, the redundant circuits induce high power consumption and increase the overall circuit size.

To address this problem, Kanani *et al.* [19] introduced a carry-selection-based accuracy-configurable approximate adder, which uses the carry prediction scheme for the next adder block's carry input to shorten the propagation delay time of the carry chain. Ye *et al.* [20] proposed a reconfiguration-oriented approximate adder composed of multiple nonoverlapping subadders, each of which contains one carry prediction circuit. This adder adjusts the computational accuracy by selecting the predicted or calculated carry output. The aforementioned adder requires neither a redundant circuit nor an error detection circuit. Because the carry signals of high-order bits are not related to low-order bits, the computational accuracy can be instantly improved during the adjustment process of the aforementioned adder.

Sakthivel and Padma [21] proposed an RCA-based reconfigurable approximate adder by using a gate diffusion logic passing cell, which is a simple CMOS connected with the previous carry value and approximated value in an adder chain. Although they claimed that the proposed adder is suitable for high-speed application, because of being based on RCA, it incurs a long propagation delay time. Afzali-Kusha *et al.* [22] designed a low-power accuracy-configurable block-based carry look-ahead adder (AC-CLA), which uses voltage scaling and approximate blocks to reduce the energy consumption. Under a given accuracy level, some blocks of the AC-CLA operate in the approximate mode, whereas other blocks operate in the accurate mode. However, the block structure with an overscaled voltage effectively reduces the adder's energy consumption. To shorten the carry chain, Xu *et al.* [15] proposed a simple accuracy-reconfigurable adder (SARA), in which the accuracy can be reconfigured by dynamically adjusting the carry chain. Both configurable and traditional full adders were used to implement the SARA in [15]. The SARA has a simple circuit structure and short carry chain; thus, it can achieve high

energy efficiency. However, when performing accurate calculations, the circuit delay is dominated by the critical path of the RCA and the multiplexer between the subadders. Accordingly, the circuit delay of the SARA is longer than that incurred by other accuracy-reconfigurable adders.

Frustaci *et al.* [1] designed an energy-quality-scalable adder (EQSA), which only performs operations on higher bits of the augend and addend and not on lower bits. The basic concept is to minimize the number of operations by truncating the unnecessary bits before the augend's and addend's signals are input to the adder. The adder proposed in [1] sets the truncated bit to a fixed value of 1 and the corresponding carry-out to 0. Compared with other accuracy-reconfigurable adders, Frustaci's adder has a smaller error distance and is easier to implement. Moreover, the power consumption and circuit delay of the aforementioned adder are lower than those of traditional accuracy-reconfigurable adders because the truncation bit is equivalent to no operation.

III. Accuracy-Configurable RD4A

This section first describes the approximate addition process of the RD4A. It then describes how this process is used in the ACRA to achieve a trade-off among power consumption, circuit delay, computational accuracy, and circuit overhead.

A. RD4A Addition

The ACRA operates in two modes: the approximate mode, in which it operates as an approximate adder, and the accurate mode, in which it operates as a normal adder. To reduce the overall design cost of an accuracy-configurable adder, the computational accuracy and circuit overhead, including circuit area, power consumption, and propagation delay time, should be considered in the design.

In this paper, the error distance (ED) and error rate (ER) are regarded as the indicators of computational accuracy. The ED is defined as the arithmetic distance between the approximate and accurate results under a given input. The expression of ED is presented in Eq.(4), in which A is the approximate result; C is the accurate result; and a_i and c_i represent the ith bits of A and C, respectively. For example, if A is 1101 and C is 1000, the ED is 5. Furthermore, the ER can be defined as a ratio of error input patterns to all possible input patterns, as expressed in Eq.(5).

$$ED(A, C) = |A - C| = |\sum_i a_i \times 2^i - \sum_j c_j \times 2^j|. \quad (4)$$

$$ER = \frac{\text{Error input pattern}}{\text{Total input pattern}} \quad (5)$$

The maximum ED for an approximate RD4A is $2^{k+2} + 2^{k+1} + 2^k$; for example, the accurate computational result of (C_{out}, Sum_{i+1}, Sum_i) is (1, 1, 1), and its approximate result is (0, 0, 0). The closer is the k value to the MSB, the higher is the ED. Therefore, to minimize the overall error distance, an attempt is made to control the ED of one approximate RD4A to within 2^k. The basic concept is to turn off the C_{in}-related logic gates, which are labeled in red in Fig. 1. The circuit function after turning off C_{in}-related logic gates is equivalent to the conventional RD4A with $C_{in} = 0$. Therefore, in the approximate mode, when C_{in} is 0, the equivalent circuit has the same function as that of the conventional RD4A. If C_{in} is 1, the adder can only generate the result for C_{in} being 0 because the C_{in}-related logic gates are turned off; thus, C_{in} cannot be calculated, and its weight is 2^k.

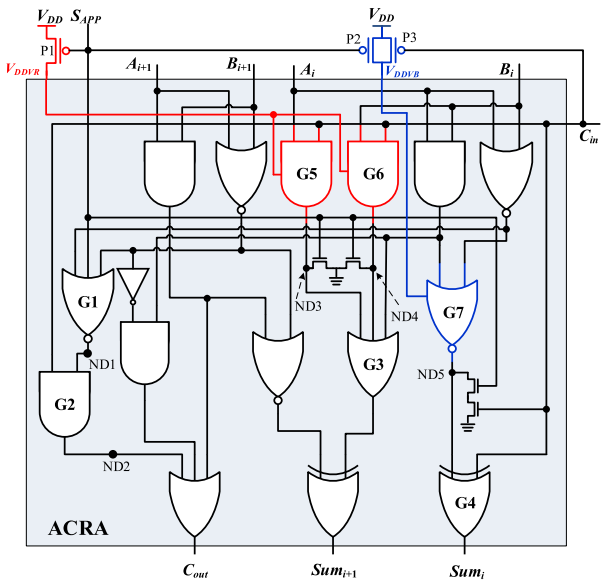

Fig. 2. The logic diagram of the proposed ACRA.

Table I presents the truth table of an approximate RD4A. To reduce the ER, when the adder is operated in the approximate mode and C_{in} is 1, the approximate output Sum_0 is automatically set to 1, as indicated in red in Table I. Thus, only eight approximate outputs differ from the accurate outputs. Consequently, the ER can be reduced from 50% to 25%. Among the eight approximate outputs, only four types of C_{out} signals (marked in gray) are incorrect (i.e., when (C_{in}, A_{i+1}, A_i, B_{i+1}, B_i) is (1, 0, 0, 1, 1), (1, 0, 1, 1, 0), (1, 1, 0, 0, 1), or (1, 1, 1,0, 0)).

According to Table I, the Boolean functions of the R4DA can be formulated as presented in Eq. (6) to Eq. (8).

$$C_{out} = A_{i+1}B_{i+1} + (A_iB_i)(A_{i+1} + B_{i+1}) \quad (6)$$

$$Sum_{i+1} = (A_{i+1} \oplus B_{i+1}) \oplus (A_iB_i) \quad (7)$$

$$Sum_i = (A_i \oplus B_i) + C_{in} \quad (8)$$

B. Functioning of the ACRA

Power gating technology is the core mechanism of the proposed ACRA for reconfiguring the computational accuracy. To minimize the circuit overhead, only power gating technology is used to control the switching transistors and output isolation elements, and no additional logic gate is required.

The logic diagram of the proposed ACRA is illustrated in Fig. 2, where G5–G7 are controlled by the power gating under different conditions and the remaining logic gates are always-on logics.

The input signal S_{APP} in Fig. 2 is a control signal that controls the mode in which the ACRA should operate.

TABLE I
TRUTH TABLE OF APPROXIMATE RADIX-4 ADDER

Inputs					Approximate Outputs			Accurate Outputs		
C_{in}	A_{i+1}	A_i	B_{i+1}	B_i	C_{out}	Sum_{i+1}	Sum_i	C_{out}	Sum_{i+1}	Sum_i
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1	0	0	1
0	0	0	1	0	0	1	0	0	1	0
0	0	0	1	1	0	1	1	0	1	1
0	0	1	0	0	0	0	1	0	0	1
0	0	1	0	1	0	1	0	0	1	0
0	0	1	1	0	0	1	1	0	1	1
0	0	1	1	1	1	0	0	1	0	0
0	1	0	0	0	0	1	0	0	1	0
0	1	0	0	1	0	1	1	0	1	1
0	1	0	1	0	1	0	0	1	0	0
0	1	0	1	1	1	0	1	1	0	1
0	1	1	0	0	0	1	1	0	1	1
0	1	1	0	1	1	0	0	1	0	0
0	1	1	1	0	1	0	1	1	0	1
0	1	1	1	1	1	1	0	1	1	0
1	0	0	0	0	0	0	1	0	0	1
1	0	0	0	1	0	0	1	0	1	0
1	0	0	1	0	0	1	1	0	1	1
1	0	0	1	1	0	1	1	1	0	0
1	0	1	0	0	0	0	1	0	1	0
1	0	1	0	1	0	1	1	0	1	1
1	0	1	1	0	0	1	1	1	0	0
1	0	1	1	1	1	0	1	1	0	1
1	1	0	0	0	0	1	1	0	1	1
1	1	0	0	1	0	1	1	1	0	0
1	1	0	1	0	1	0	1	1	0	1
1	1	0	1	1	1	0	1	1	1	0
1	1	1	0	0	0	1	1	1	0	0
1	1	1	0	1	1	0	1	1	0	1
1	1	1	1	0	1	0	1	1	1	0
1	1	1	1	1	1	1	1	1	1	1

When S_{APP} is 1, the ACRA operates in the approximate mode, and when S_{APP} is 0, the ACRA operates in the accurate mode. The power supply of the ACRA is depicted in the top part of Fig. 2. Two slave power supplies, namely V_{DDVR} and V_{DDVB}, are used. These power supplies are derived from the master power supply V_{DD}. In Fig. 2, G5 and G6 are powered by V_{DDVR}, G7 is powered by V_{DDVB}, and the remaining logic gates are directly powered by V_{DD}. When the ACRA is operating in the accurate mode (i.e., $S_{APP} = 0$), the transistors P1 and P2 are turned on to connect V_{DD} to V_{DDVR} and V_{DDVB}. By contrast, when the ACRA is operating in the approximate mode (i.e., $S_{APP} = 1$), P1 and P2 are turned off; thus, V_{DDVR} is disconnected from V_{DD}. The on or off status of transistor P3 depends on the C_{in} signal. When C_{in} is 0, transistor P3 is turned on and the voltage level of V_{DDVB} is the same as that of V_{DD}. However, when C_{in} is 1, V_{DDVB} and V_{DD} are disconnected. Therefore, G7 is powered off when the ACRA operates in the approximate mode and C_{in} is 1.

In Fig. 2, G1 and G2 are two logic gates on the carry chain and the ACRA uses the S_{APP} signal to control the output of G1. When the ACRA operates in the accurate mode, S_{APP} is 0, the signal at ND1 is $(A_{i+1} + B_{i+1})(A_i + B_i)$, and the signal at ND2 is $C_{in}(A_{i+1} + B_{i+1})(A_i + B_i)$; thus, C_{out} can be obtained using Eq. (1). When the ACRA operates in the approximate mode (i.e., S_{APP} is 1), both the ND1 and ND2 signals are 0; thus, C_{out} can be obtained using Eq. (6). As indicated in Fig. 2, C_{out} can achieve the function designing goal of this paper in different operating modes. In addition, when the ACRA is performing approximate calculations, the internal signal-switching activities of G1 and G2 stop, which decreases the power consumption.

In Fig. 2, G3 and G4 are the next logic gates of G5–G7. When G5–G7 are turned off, G3 and G4 must be assigned an exact potential (i.e., GND) so that they can operate as normal. In the accurate mode, the value of ND3 is $C_{in}A_i$, the value of ND4 is $C_{in}B_i$, the output of G3 is $A_iB_i + C_{in}A_i + C_{in}B_i$, and Sum_{i+1} can be obtained using Eq. (2). As shown in Fig. 3(a), since G5 and G6 are powered by V_{DDVR}, in the approximate mode ($S_{APP} = 1$), the power of G5 and G6 is turned off and the values of ND3 and ND4 are decreased to 0 due to the two NMOS turn-on situations; thus, the output of G3 in Fig. 2 is A_iB_i and Sum_{i+1} can be calculated using Eq. (7). Because the power of G5 and G6 is turned off, no shoot-through current in the approximate mode is occurred.

In Fig. 2, G4 is a logic gate that generates the Sum_i signal. When S_{APP} is 0, the value of ND5 is the same as the output of G7 and Sum_i can be obtained using (3). When S_{APP} is 1, the output of G4 is dependent on the input value of C_{in}. When C_{in} is 0, the value of ND5 remains the same as the output of G7 and $Sum_i = A_i \oplus B_i$. When S_{APP} is 1 and C_{in} is 1, as shown in Fig. 3(b), G7 is turned off, and the value of ND5 is decreased to 0 by the NMOS, so that the output Sum_i in Fig. 2 is equal to 1. In this situation, no shoot-through current will occur in ND5. Thus, the value of Sumi is

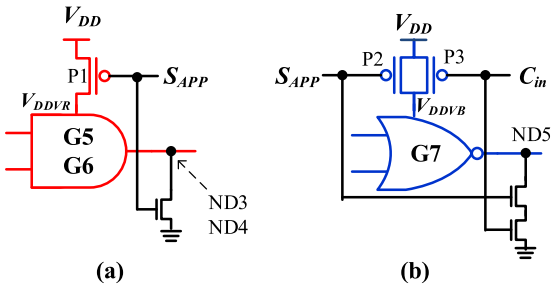

Fig. 3. The operations of gate G5 to G7 in approximate mode.

dynamically controlled by S_{APP} and C_{in}. The aforementioned operation represents the so-called dynamic output modification scheme.

The equivalent circuit diagrams of the ACRA in the accurate and approximate modes are presented in Figs. 4(a) and 4(b), respectively. The architecture displayed in Fig. 4(a) is completely equivalent to the RD4A architecture depicted in Fig. 1.

The ACRA modifies the original RD4A architecture and enables it to adjust the computational accuracy. In the approximate mode, the ACRA cuts off the power supply of up to three logic gates to suspend the switching activity of G1, G2, and G4 so that the power consumption can be reduced. It also cuts the carry chain so that the C_{out} signal is independent of the C_{in} signal. Thus, the ACRA reduces the delay time of a multibit adder connected in series while maintaining an ED of 2^k and ER of 25%.

A single ACRA is a 2-bit adder. To implement a multibit adder, one RD4A and multiple ACRAs are added to an ACRA-based completely configurable adder architecture, as presented in Fig. 5. A total of $n/2$ adders are required for an n-bit completely configurable adder. Only the lowest bit element is an RD4A, whereas the remaining $n/2-1$ adder elements are ACRAs.

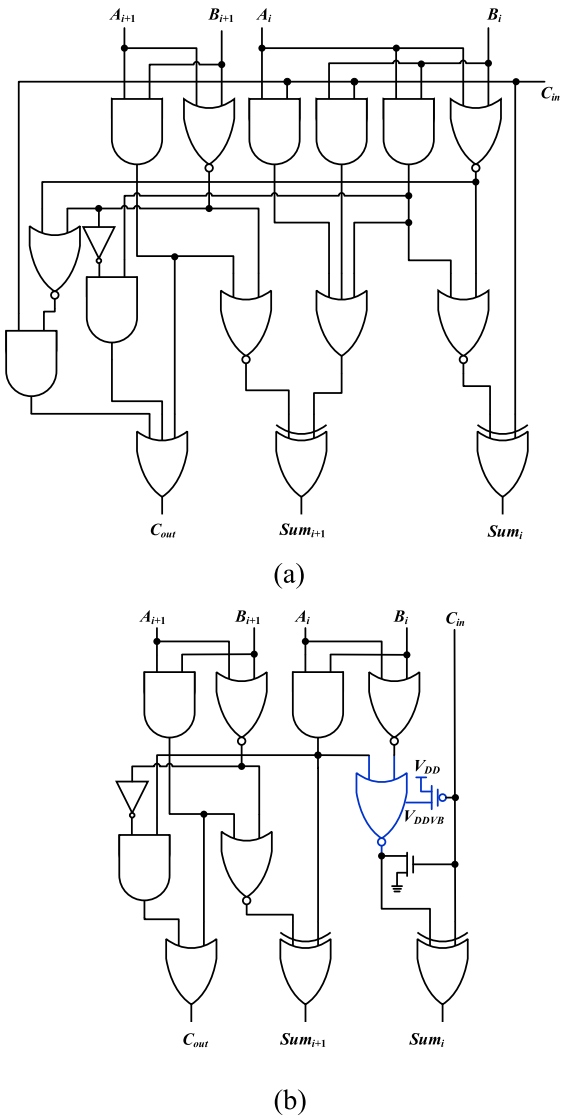

Fig. 4. Equivalent circuit diagram of ACRA in (a) accurate mode and (b) approximate mode.

IV. Experimental Results

This section presents the power consumption, delay time, PDP, and circuit size of the ACRA and two previously proposed accuracy-configurable adders. Moreover, to examine the effects of the computational accuracy on image quality, the results of image processing are used to compare the output quality of the aforementioned three accuracy-configurable adders.

A. Experimental Setting

To verify the feasibility and effectiveness of the ACRA, three accuracy-configurable adders (i.e., the ACRA, SARA [15], and EQSA [1]) and three conventional adders (i.e., the RCA, CLA, and RD4A) were implemented using TSMC 45 nm CMOS technology with $V_{DD} = 1.0$ V. All the parameters, including interconnects and parasitics, were determined from the circuit layout. A total of 100 random input patterns with uniform distribution were used for the simulation, and the average power consumption was then calculated. The propagation delay time was measured on the basis of the worst case of each adder. A simulation was performed using the HSPICE simulation tool [23]. Three bit lengths, namely 8, 16, and 32 bits, were designed for the aforementioned six adders. Furthermore, for the three accuracy-configurable adders, four operation modes were designed (Fig. 6): the accurate mode (_A), quarter-approximate mode (_Q), half-approximate mode (_H), and three-quarter-approximate mode (_TQ).

B. Analyses of the Power Consumption, Propagation Delay, and PDP

The power consumption, propagation delay time, and PDP of the different adders are listed in Table II, where the RCA, CLA, and RD4A are conventional accurate adders and do not possess three approximate modes (_Q, _H, and _TQ). The ACRA, SARA, and EQSA use 4 bits as one module to control the power consumption during implementation; thus, 8 bits cannot be divided by _Q and _TQ (NA entries in Table II). Without considering the three conventional adders, the best result for each mode is marked in gray color in Table II.

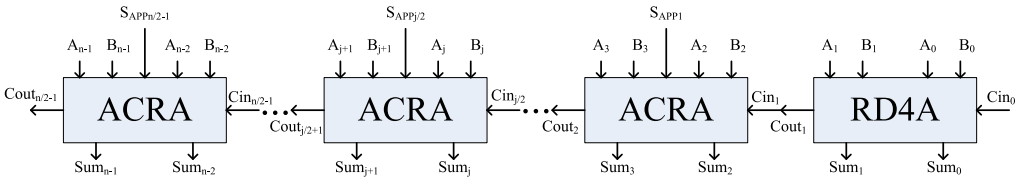

Fig. 5. n-bit ACRA based fully configurable adder.

TABLE II
THE POWER CONSUMPTION, PROPAGATION DELAY, AND POWER-DELAY PRODUCT (PDP) OF DIFFERENT ADDERS

Adder	8 bits			16 bits			32 bits		
	Power (μW)	Delay (ps)	PDP (zJ)	Power (μW)	Delay (ps)	PDP (zJ)	Power (μW)	Delay (ps)	PDP (zJ)
RCA_A	20.47	151.3	3.1	43.65	294.7	12.86	88.29	581.8	51.36
CLA_A	25.83	102.8	2.66	54.99	188.8	10.38	110.3	365.8	40.35
RD4A_A	21.66	111.9	2.32	44.83	197.6	8.87	92.16	370.1	34.11
ACRA_A	21.01	116.5	2.48	45.41	205.7	9.48	93.67	384.4	36.47
ACRA_Q	NA	NA	NA	43.8	183.9	8.05	88.74	317.7	28.19
ACRA_H	19.24	94.7	1.82	40.05	139.2	5.57	80.58	228.5	18.41
ACRA_TQ	NA	NA	NA	36.17	94.7	3.43	73.95	139.2	10.29
SARA_A	21.66	169.1	3.66	47.55	348.4	16.56	97.77	706.5	69.07
SARA_Q	NA	NA	NA	45.14	284.6	12.85	93.7	559.4	52.41
SARA_H	19.54	105.5	2.06	43.67	195.1	8.52	89.29	374.1	33.4
SARA_TQ	NA	NA	NA	41.72	105.5	4.4	85.16	195.1	16.61
EQSA_A	26.09	159.2	4.15	55.24	302.7	16.72	110.7	589.8	65.29
EQSA_Q	NA	NA	NA	40.45	231	9.34	81.64	446.2	36.43
EQSA_H	11.77	83.7	0.99	26.3	159.2	4.19	53.53	302.8	16.21
EQSA_TQ	NA	NA	NA	13.15	83.7	1.1	25.15	155.5	3.91

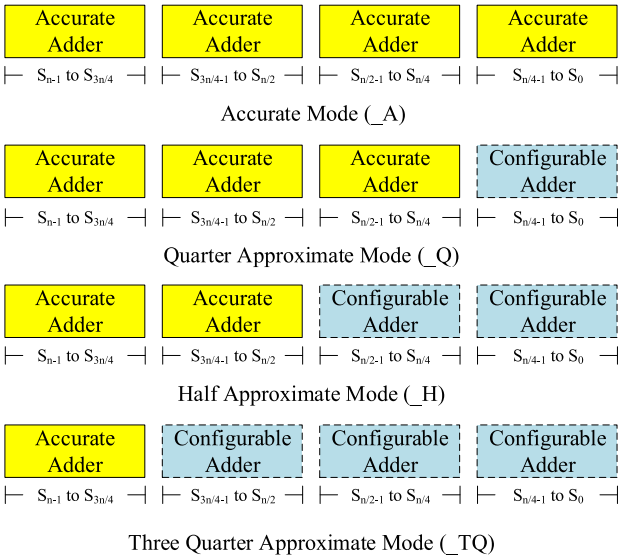

Fig. 6. Four operation modes for accuracy configurable adders ACRA, SARA, and EQSA.

Table II indicates that in the accurate mode (_A), the power consumption of the RCA was lower than that of other adders. This result was obtained mainly due to the simple structure of the RCA. However, a traditional RCA has a long propagation delay time and is not practical for modern systems. The CLA used several logic gates to improve the propagation delay time and consumed more power than did the other adders. For a bit length of 32 bits, compared with the RD4A, the ACRA consumed only 1.64% ($= \frac{93.67-92.16}{92.16}$) more power and had only a 3.86% ($= \frac{384.4-370.1}{370.1}$) longer propagation delay time. This result was obtained because G1 (Fig. 2) is located on the critical path and is a two-input NOR gate in the RD4A but is a three-input NOR gate in the ACRA. Consequently, the power consumption and propagation delay time of the ACRA were marginally higher and longer, respectively, than those of the RD4A. In the accurate mode, the power consumption of the ACRA was 4.19% ($= \frac{97.77-93.67}{97.77}$) and 15.38% ($= \frac{110.7-93.67}{110.7}$) lower than that of the SARA and EQSA, respectively, for a bit length of 32 bits. Moreover, in the accurate mode, the propagation delay time of the ACRA was shorter than those of the SARA and EQSA, respectively, because these adders were implemented on the basis of the RCA. Table II indicates that for a bit length of 32 bits, the propagation delay time of the ACRA was 45.60% ($= \frac{706.5-384.4}{706.5}$) and 34.83% ($= \frac{589.8-384.4}{589.8}$) shorter than those of the SARA and EQSA, respectively.

In the approximate mode, because the SARA uses multiplexers to control the computational accuracy and its circuit overhead is considerably larger than those of the other adders, it consumed more power and had a longer propagation delay time than the ACRA and EQSA did. For a bit length of 32 bits, the SARA consumed approximately 5% - 13% more power and had a 40.2% – 76.1% longer delay time than did the proposed ACRA. The EQSA truncated the unnecessary bits so that the switching activity of the truncated logic gates could be suspended. The switching activity was the main factor influencing dynamic power consumption, and no switching activity indicated no operation and consequently no dynamic power consumption. Therefore, in the three approximate modes, the EQSA had a considerably lower power consumption than did the other adders. As presented in Table II, the power consumed by the EQSA was lower than that consumed by the ACRA. However, the relative computational accuracy of the EQSA decreased significantly as the power consump-

TABLE III
TOTAL NUMBER OF TRANSISTORS IN THE DIFFERENT ADDERS

	8 bits	16 bits	32 bits
RCA	336	672	1344
CLA	472	944	1888
RD4A	348	740	1524
ACRA	375	803	1659
SARA	356	732	1484
EQSA	424	872	1744

tion reduced; thus, although the ACRA had a marginally higher power consumption than the EQSA, the ACRA exhibited higher computational accuracy (discussed in detail in section 4.5). The propagation delay time of the EQSA was affected by the number of nontruncated bits because the truncated bits in the EQSA were not used to produce the carry output. However, the nontruncated bits of the EQSA were implemented on the basis of the RCA; thus, the overall propagation delay time of the EQSA was longer than that of the ACRA.

We also used the PDP to evaluate the overall performance of the circuit. The smaller the PDP, the better is the circuit's performance. As presented in Table II, the PDP of the ACRA was considerably smaller than that of the SARA in both the accurate and approximate modes. For a bit length of 32 bits, the PDP of the ACRA was 38% – 47% lower than that of the SARA. Because the power consumption of the EQSA was lower than that of the ACRA in the approximate mode, after multiplying the propagation delay time with the power consumption, the PDP of the EQSA in the _H and _TQ modes was still lower than that of the ACRA; however, in the _A and _Q modes, the PDP of the EQSA was higher than that of the ACRA.

In summary, the proposed ACRA had a lower power consumption than did the SARA in both the accurate and approximate modes. Moreover, the ACRA had a lower and higher power consumption than did the EQSA in the accurate and approximate modes, respectively.

C. Circuit Size

We used the number of transistors to evaluate the circuit size. Table III lists the total number of transistors required for the different adders at different bit lengths. The aforementioned table indicates that the CLA required the maximum number of transistors, whereas the RCA required the least number of transistors. The ACRA required fewer transistors than the EQSA did but marginally more transistors than the SARA did. This result indicates that many additional circuits must be used when designing an EQSA.

D. Image Quality Analysis

To indicate the effects of different computational accuracies, in this study, the image smoothing was performed on three image benchmarks, namely Lena, Peppers, and Mandrill so that the computational results of the 16-bit ACRA, SARA, and EQSA can be compared. The adopted image processing

Fig. 7. Lena's image processing results under different accuracy configurable adders.

procedure is based on the method proposed in [24], in which a 5×5 kernel matrix is used to perform convolution on the original image and obtain a feature image. For fair comparison, only additions are used for the image processing.

In this study, the PSNR and SSIM were used to evaluate the processed image. The PSNR is an objective value for evaluating the image quality and can be defined by the mean square error (MSE). The formulas of the MSE and PSNR are presented in Eq. (9) and Eq. (10), respectively, in which m and n represent the length and width of an image, respectively; $I(i, j)$ and $K(i, j)$ indicate the pixel values at position (i, j) for two images; and MAX_I indicates the maximum signal strength. The unit of the PSNR is dB. The larger the PSNR, the smaller is the distortion between two images. In general, PSNR values higher than 30 dB and less than 10 dB indicate a suitable and unacceptable image quality, respectively.

$$MSE = \frac{1}{mn}\sum_{i=0}^{m-1}\sum_{j=0}^{n-1}(I(i, j) - K(i, j))^2 \tag{9}$$

$$PSNR = 20 \times \log_{10}(\frac{MAX_I}{\sqrt{MSE}}) \tag{10}$$

The SSIM is used to measure the similarity between two images; thus, the SSIM is more in line with the human eye's perception of image quality than the PSNR is. The SSIM compares the brightness, contrast, and structure between

Fig. 8. Peppers' image processing results under different accuracy configurable adders.

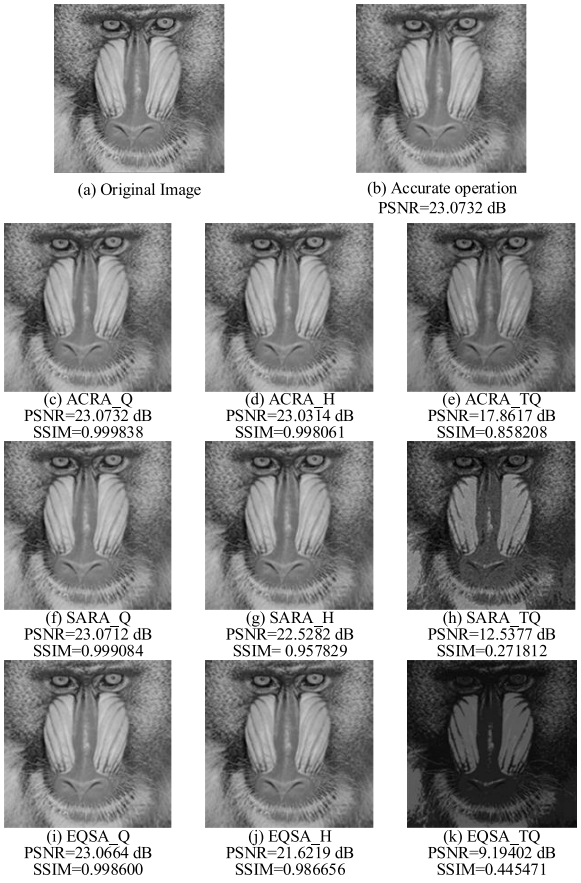

Fig. 9. Mandrill's image processing results under different accuracy configurable adders.

two images. If the brightness, contrast, and structure weights are set equal, the SSIM can be expressed using Eq. (11), in which μ_x and σ_x^2 represent the mean and variance of all pixels in an image x, respectively; σ_{xy} is the covariance of all pixels in images x and y; and c_1 and c_2 are two constants used to maintain the stability of the brightness and contrast, respectively. The SSIM ranges from 0 to 1, and the closer the SSIM is to 1, the more similar are the structures of two images. When the SSIM exceeds 0.95, the image quality is acceptable for the human eye. Moreover, an SSIM value exceeding 0.98 indicates that two images are almost indistinguishable by the human eye.

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)} \qquad (11)$$

Figs. 7–9 depict the image processing results for the Lena, Peppers, and Mandrill benchmarks, respectively, under the application of different accuracy-configurable adders. When the three adders operated in the _TQ mode, compared with the ACRA image, the SARA and EQSA images were unacceptable. For operation in the _Q and _H modes, except for some details, the qualities of the ACRA, SARA, and EQSA images were generally similar. The ACRA had the highest PSNR and SSIM values among the three accuracy-configurable adders. In the _H mode, the PSNR of the ACRA was less than 1 dB lower than that obtained for the accurate calculation and the SSIM was maintained above 0.99. The SARA and EQSA had significantly lower PSNR and SSIM values than the ACRA did in the _H mode, which indicates that the quality of the SARA and EQSA images significantly declined in this mode. In the _TQ mode, the PSNR values of the SARA and EQSA were more than 20 dB lower than that obtained for the accurate calculation, and the corresponding SSIM was less than 0.5. As displayed in Figs. 7–9, under the _H and _TQ modes, the PSNR of the SARA was higher than that of the EQSA but the SSIM of the SARA was lower than that of the EQSA. This result indicates that the PSNR and SSIM use different standards for measuring the image quality. The PSNR focuses on the distortion caused by noise, whereas the SSIM compares the structures of two images.

V. Conclusion

This paper proposes an ACRA to adjust the computational accuracy. The power gating and control signal forcible settings are used in the proposed ACRA to reduce the circuit's overhead. According to the experimental results, the ACRA outperforms the SARA in terms of power consumption, propagation delay time, and computational accuracy. Although the EQSA consumes less power than the ACRA, the ACRA outperforms the EQSA in terms of propagation delay time and computational accuracy. The EQSA exhibits low power consumption at the cost of computational accuracy; thus,

the EQSA has limited use in applications that require high computational accuracy. In the accurate mode, the ACRA has a lower PDP than did the RCA and CLA. In the _Q and _H modes, the ACRA obtained almost the same image quality as that obtained in an accurate operation but had a 45%–80% lower PDP than a conventional RCA. Thus, the proposed ACRA performs well in both the accurate and approximate modes.

ACKNOWLEDGMENT

The authors would like to thank to Synopsys Inc., and Dr. Charles Chiang of Synopsys Inc., for the donation of the Tunghai University Electronic Design Automation Education Project to enhance the education of electronic design in Tunghai University. They would also like to acknowledge the Taiwan Semiconductor Research Institute (TSRI), Taiwan, for providing circuit design and verification tools and cell library.

REFERENCES

[1] F. Frustaci, S. Perri, P. Corsonello, and M. Alioto, "Energy-quality scalable adders based on nonzeroing bit truncation," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 27, no. 4, pp. 964–968, Apr. 2019.

[2] M. Biasielli, L. Cassano, and A. Miele, "An approximation-based fault detection scheme for image processing applications," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2020, pp. 1331–1334.

[3] L. Jin and H. Liang, "Deep learning for underwater image recognition in small sample size situations," in *Proc. OCEANS Aberdeen*, Jun. 2017, pp. 1–4.

[4] Y.-J. Chang, Y.-C. Cheng, Y.-F. Lin, S.-C. Liao, C.-H. Lai, and T.-C. Wu, "Imprecise 4-2 compressor design used in image processing applications," *IET Circuits, Devices Syst.*, vol. 13, no. 6, pp. 848–856, 2019.

[5] S. Natarajan, *Imprecise and Approximate Computation*, vol. 318. New York, NY, USA: Springer, 1995.

[6] G. Paim, L. M. G. Rocha, G. M. Santana, L. B. Soares, E. A. C. da Costa, and S. Bampi, "Power-, area-, and compression-efficient eight-point approximate 2-D discrete Tchebichef transform hardware design combining truncation pruning and efficient transposition buffers," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 66, no. 2, pp. 680–693, Feb. 2019, doi: 10.1109/TCSI.2018.2868513.

[7] Z. Zhang, Y. He, and B. Zhang, "A time-efficient automatic circuit approximation method," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 9, pp. 3047–3055, Sep. 2020, doi: 10.1109/TCSI.2020.2984526.

[8] M. Ramasamy, G. Narmadha, and S. Deivasigamani, "Carry based approximate full adder for low power approximate computing," in *Proc. 7th Int. Conf. Smart Comput. Commun. (ICSCC)*, Jun. 2019, pp. 1–4.

[9] N. Van Toan and J.-G. Lee, "FPGA-based multi-level approximate multipliers for high-performance error-resilient applications," *IEEE Access*, vol. 8, pp. 25481–25497, 2020.

[10] N.-C. Huang, S.-Y. Chen, and K.-C. Wu, "Sensor-based approximate adder design for accelerating error-tolerant and deep-learning applications," in *Proc. Design, Autom. Test Eur. Conf. Exhib. (DATE)*, Mar. 2019, pp. 692–697.

[11] C. Yang and H. Jiao, "Low power Karnaugh map approximate adder for error compensation in loop accumulations," in *Proc. Int. Conf. IC Design Technol. (ICICDT)*, Jun. 2019, pp. 1–4.

[12] V. Gupta, D. Mohapatra, A. Raghunathan, and K. Roy, "Low-power digital signal processing using approximate adders," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 32, no. 1, pp. 124–137, Jan. 2013.

[13] D. Esposito, D. De Caro, E. Napoli, N. Petra, and A. G. M. Strollo, "On the use of approximate adders in carry-save multiplier-accumulators," in *Proc. IEEE Int. Symp. Circuits Syst. (ISCAS)*, May 2017, pp. 1–4.

[14] H. Seo, Y. S. Yang, and Y. Kim, "Design and analysis of an approximate adder with hybrid error reduction," *Electronics*, vol. 9, no. 3, pp. 1–13, Mar. 2020.

[15] W. Xu, S. S. Sapatnekar, and J. Hu, "A simple yet efficient accuracy-configurable adder design," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 26, no. 6, pp. 1112–1125, Jun. 2018.

[16] A. B. Kahng and S. Kang, "Accuracy-configurable adder for approximate arithmetic designs," in *Proc. 49th Annu. Design Autom. Conf. (DAC)*, Jun. 2012, pp. 820–825.

[17] O. Akbari, M. Kamal, A. Afzali-Kusha, and M. Pedram, "RAP-CLA: A reconfigurable approximate carry look-ahead adder," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 65, no. 8, pp. 1089–1093, Aug. 2018.

[18] C.-H. Lai, Y.-C. Cheng, T.-C. Wu, and Y.-J. Chang, "Radix-4 adder design with refined carry," in *Proc. IEEE Conf. Dependable Secure Comput.*, Aug. 2017, pp. 300–304.

[19] A. Kanani, J. Mehta, and N. Goel, "ACA-CSU: A carry selection based accuracy configurable approximate adder design," in *Proc. IEEE Comput. Soc. Annu. Symp. VLSI (ISVLSI)*, Jul. 2020, pp. 434–439.

[20] R. Ye, T. Wang, F. Yuan, R. Kumar, and Q. Xu, "On reconfiguration-oriented approximate adder design and its application," in *Proc. IEEE/ACM Int. Conf. Comput.-Aided Design (ICCAD)*, Nov. 2013, pp. 48–54.

[21] B. Sakthivel and A. Padma, "Area and delay efficient GDI based accuracy configurable adder design," *Microprocessors Microsyst.*, vol. 73, Mar. 2020, Art. no. 102958.

[22] H. Afzali-Kusha, M. Kamal, and M. Pedram, "Low-power accuracy-configurable carry look-ahead adder based on voltage overscaling technique," in *Proc. 21st Int. Symp. Qual. Electron. Design (ISQED)*, Mar. 2020, pp. 67–72.

[23] *HSPICE Circuit Simulator*, Synopsys, Inc., Mountain View, CA, USA, 2019.

[24] O. Akbari, M. Kamal, A. Afzali-Kusha, and M. Pedram, "Dual-quality 4:2 compressors for utilizing in dynamic accuracy configurable multipliers," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 25, no. 4, pp. 1352–1361, Apr. 2017.

Kun-Lin Tsai (Member, IEEE) received the Ph.D. degree in electrical engineering from National Taiwan University, Taipei, Taiwan, in 2006. He was a Post-Doctoral Researcher with the National Taiwan University of Science and Technology in 2007. He is currently an Associate Professor with the Department of Electrical Engineering, Tunghai University, Taichung, Taiwan. From 2015 to 2018, he was the Department Chair. His research interests include low-power system design, digital integrated circuit design, information security, and the Internet-of-Things. He is a member of IICM, CCISA, and CIE.

Yen-Jen Chang (Member, IEEE) received the Ph.D. degree in computer science and information engineering from National Taiwan University, Taipei, Taiwan, in 2003. He joined as a Faculty Member with the Department of Computer Science and Engineering, National Chung Hsing University, Taichung, Taiwan, in 2004, where he is currently a Professor. He has been serving as the Department Chair, since August 2019. His research interests include VLSI design, low-power memory design, embedded systems, and system-on-chip design.

Chien-Ho Wang received the M.S. degree in computer science and engineering from National Chung Hsing University (NCHU), Taichung, Taiwan, in 2019. He is currently pursuing the Ph.D. degree with the Department of Computer Science and Engineering, NCHU. His research interests include low power memory design, deep neural networks, and deep learning.

Cheng-Tse Chiang received the M.S. degree in computer science and engineering from National Chung Hsing University, Taichung, Taiwan, in 2020. His research interests include FinFET technique and low-power memory design.