Computers and Electrical Engineering 87 (2020) 106761

Contents lists available at ScienceDirect

Computers and Electrical Engineering

journal homepage: www.elsevier.com/locate/compeleceng

Process variation-aware approximate full adders for imprecision-tolerant applications

Mohammad Mirzaei[a], Siamak Mohammadi[a,b,*]

[a] Dependable System Design Lab., School of Electrical and Computer Engineering, University of Tehran, Tehran, Iran
[b] School of Computer Science, Institute of Fundamental Sciences (IPM), Tehran, Iran

ARTICLE INFO

Article history:
Received 28 August 2019
Revised 26 June 2020
Accepted 3 July 2020
Available online xxx

Keywords:
Imprecision-tolerant applications
Approximate computing
Full adder
Process variation

ABSTRACT

In imprecision-tolerant applications such as multimedia and signal processing a slightly degraded output quality is acceptable, which could lead to significant power reduction. We have proposed three approximate full adders for such applications. We have also considered the process variation impacts on approximate full adders for the first time. Twelve approximate full adders including ours have been used and analyzed in a ripple carry adder, as well as in a Sobel edge detection application. Experimental results show our design compared to the best existing approximate full adder reduces power consumption, delay and power-delay-product by 6.2%, 42.3% and 46.02%, respectively. Our proposed approximate ripple carry adder has the best performance in terms of error parameters, and the best PSNR (Peak signal-to-noise ratio) when used in Sobel algorithm. When the proposed approximate full adder is used in Sobel algorithm, it exhibits the least variability in terms of delay compared to other counterparts.

1. Introduction

Applications such as image and signal processing, machine learning and computer vision are less sensitive to output quality, as one can afford imprecision when using them. These applications are called imprecision-tolerant. In such applications by employing approximate computing and via slight output quality reduction, considerable area and power consumption reduction and performance improvement can be achieved [1]. For instance, in 3D raytracer application 98% of floating-point operations, and 91% of input data can undergo approximate computing. In k-means clustering algorithms, by reducing 5% of classification precision, 50X energy reduction can be obtained [2,3].

The most important computation in digital systems is the binary computation, where adders are used as the main structure to perform addition, subtraction, multiplication and division. Hence, addition is usually mostly employed in different application. In most imprecision-tolerant applications, approximate adders and multipliers are used. Basic elements in these structures are approximate full adders. Consequently, in this paper, we will analyze and evaluate them as one of the most useful elements in approximate computing [4].

With technology advances, during semi-conductor fabrication, factors such as lithography, chemical mechanical polish and lens defect can change transistor and interconnections parameters, which will modify transistor electrical features such as current, threshold voltage and gate capacitance. Consequently, using these transistors will affect delay, power consump-

[*] Corresponding author.
E-mail addresses: mo.mirzaei@ut.ac.ir (M. Mirzaei), smohamadi@ut.ac.ir (S. Mohammadi).

https://doi.org/10.1016/j.compeleceng.2020.106761
0045-7906/© 2020 Elsevier Ltd. All rights reserved.

tion and performance of the design. In this case, variability becomes a challenge in submicron technology leading to reliability issues [5]. Imprecision-tolerant application imply approximate adders to reduce delay and power consumption overheads of the system, while variation affects these characteristics. Therefore, variation effects of approximate adders must be investigated and diminished.

Variability is divided into two categories: process variation and environment variation [6]. Process variation occurs during fabrication and is permanent. This type of variation is due to processing and masking constraints. Environment variation shows itself over circuitâs lifetime, which includes temperature, activity factor and supply voltage variations. Within-Die (WID) and Die-to-Die (D2D) are two types of process variation. In the former, different factors can occur on the die due to random dopant fluctuation (RDF), causing threshold voltage to be non-identical on different areas of the die. However, D2D variation is identical on the whole area of the die, but differences in a wafer thickness, the threshold voltage of two adjacent dies could be different, but identical within a die.

Therefore, variability appears as an important factor in todayâs technology; however none of the work done in the field of approximate computing has considered the effects of this issue. To the best of our knowledge, we have studied for the first time, the influence of variation on approximate full adders. Based on works in [5] and [7], and the importance of threshold voltage (V_{th}) in variation, we will evaluate D2D and WID V_{th} variation effects on approximate full adders performance.

Our major contributions in this paper are as follows:

(1) We propose three novel approximate full adders, namely AFA1, AFA2 and AFA3, which compared to their existing counterparts exhibit mostly better power consumption, delay and PDP. Also, in terms of error they outperform some other approximate full adders. D2D and WID V_{th} variation effects on power, delay and PDP of nine existing approximate full adders and three proposed approximate full adders are evaluated. HSPICE Monte Carlo simulations are performed in 32nm technology.
(2) We design an approximate ripple carry adder using AFA1 and AFA3 for its different bits, which leads to best error parameters, and when applied in Sobel edge detection algorithm, it gives the best PSNR.

The rest of the paper is as follows: Section 2 describes previous work on approximate components. In the next section, we will evaluate approximate full adders in terms of power, delay, PDP, error and variation. Section 4 analyzes variation and error parameters of an 8-bit RCA based on approximate full adders. Simulation results of a real imprecision-tolerant application are presented in Section 5. Finally, we conclude our paper.

2. Previous work

An approximate adder with configurable accuracy is presented in [8]. To this propose a full adder and a half adder with maskable carry are proposed and used in RCA structure. To adjust the precision, an error correction circuit is designed, which can be activated when necessary. An approach to find energy-efficient hybrid approximate adder for image and video processing applications is presented in [9]. The aim is to perform multiplications without using traditional multipliers, and thus the main operations in this work are done by shift-and-add by using parallel prefix adders.

An efficient method for calculating the error statistics of block-based approximate adders is proposed in [10]. A reconfigurable approximate carry look-ahead adder is introduced in [11]. This adder can work in two approximate and accurate modes, and has been implemented in 15nm Finfet technology. In [12], using approximate full adders, authors have proposed an approximate discrete cosine transform (DCT) for image compression. In this work, floating point multipliers have been eliminated, and instead integer addition and shift are employed. In [13] by truncating the inputs, an approximate multiplier has been proposed. In this paper, n-bit multiplications are converted to multiplications over fewer bits, shift and add operations. Also, the accuracy of the multiplication is controllable at the execution time.

By eliminating some transistors of a mirror full adder in [14] and [15], three and four approximate full adders have been proposed, respectively. Due to the reduction of switching capacitances and the elimination of some transistors, less delay, power and area were obtained compared to exact mirror full adders. However, these designs sometimes produce erroneous outputs.

Three approximate full adders based on XOR and XNOR have been proposed in [16], using pass transistors. Also, in [17] three different approximate full adders were designed using standard gates based on pass transistors. In these two works, in addition of some erroneous results for some inputs, the problem of threshold drop due to pass transistors is seen [18]. Two approximate full adders based on XOR and MUX using transmission gates for their implementation, have been proposed in [19]. They have no threshold drop issue, but consume more power compared to other approximate full adders.

In [20] a half adder, a full adder and a 4:2 compressor, all approximate structures, are proposed for an array multiplier. In the approximate full adder, to produce Sum output an OR gate has been used instead of one of the XOR gates. For the carry AND and OR gates have been employed. In [21] an approximate half adder and an approximate full adder using NAND gates have been designed. Based on the paperâs results, this approximate full adder is acceptable in terms of energy consumption; however it ranks among worse designs in terms of error.

All approximate full adders presented in the aforementioned papers are different from each other in terms of power, delay, precision and area; however none of them has considered the effects of variations. Our goal is to evaluate the performance (power, delay, precision) as well as the variability of approximate full adders. Therefore, we intend to identify or design components with better trade-off in terms of power, delay, precision and variability.

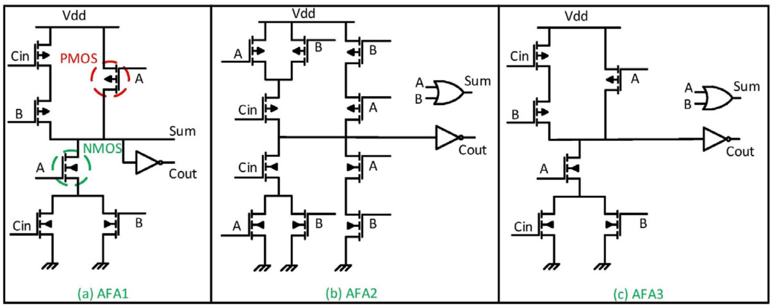

Fig. 1. The proposed AFA1 (a), the proposed AFA2 (b) and the proposed AFA3 (c).

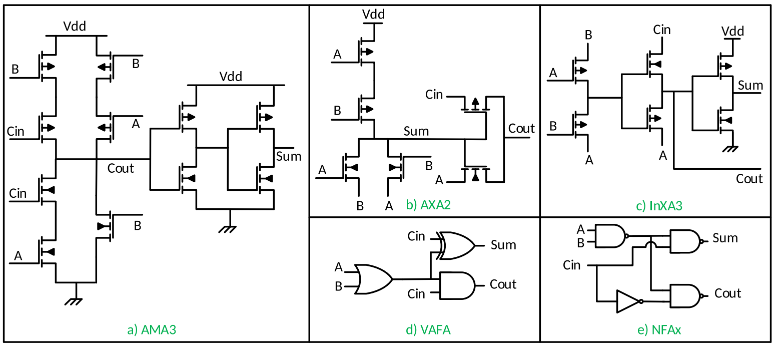

Fig. 2. The existing approximate full adders. (a) AMA3, (b) AXA2, (c) InXA3, (d) VAFA and (e) NFAx.

3. Proposed approximate full adders

In this paper, three approximate full adders, namely approximate full adder 1 (AFA1), approximate full adder 2 (AFA2), and approximate full adder 3 (AFA3) have been proposed. Their transistor level schematics are shown in Fig. 1. The logical equations of these proposed full adders are as follows:

$$AFA1 : Cout = A(B + Cin), Sum = \overline{A(B + Cin)} \tag{1}$$

$$AFA2 : Cout = Cin(A + B) + AB, Sum = A + B \tag{2}$$

$$AFA3 : Cout = A(B + Cin), Sum = A + B \tag{3}$$

As in all cases of exact full adders, except when all inputs are 0 or 1, the value of Sum is the inverse of Cout. Thus, using an inverter at the output Sum to produce Cout, considerably reduces power, delay and area. Therefore, in AFA1 the same logic has been used, and the design of the approximate output Sum, has been based on CMOS logic.

For adding two arbitrary unsigned numbers the input LSB carry in the RCA is zero. Based on our evaluations, for all possible inputs with a uniform distribution, the probability of having ABCin=XX0 is higher than ABCin=XX1 at the inputs of full adders of RCA, where A, B, and Cin are inputs to the full adder. For example for an 8-bit adder, the probability of occurring ABCin=XX0 is 62.44%, whereas that of ABCin=XX1 is 37.56%.

Therefore, designing approximate full adders that almost work correctly in cases where ABCin=XX0, will reduce error parameters. Hence, all approximate full adders AFA2, and AFA3 have been designed to fulfill this purpose. For AFA2, the output Cout is exact. However, to produce Sum, a 2-input OR gate with inputs A and B has been used. Hence, Sum does not need the carry of the previous stage. As soon as A and B are ready, Sum can be generated. AFA3 full adder logic is similar to AFA1 and AFA2 for Cout and Sum, respectively. Therefore, AFA2 and AFA3 have errors in 4 over 8 possible cases, where 3 error cases occur when Cin=1.

The precise mirror full adder proposed in [15] is used as the base design, which we refer to as CMA (Conventional Mirror Adder) in the rest of the paper. The nine approximate full adders are referred to as AMA1, AMA2, AMA3 [15], AXA2, AXA3 [16], InXA2, InXA3 [17],VAFA [20] and NFAx [21]. In Fig. 2, five approximate full adders are presented. In AMAs, AXAs and InXAs families, the best full adder in terms of PDP is presented in Fig. 2.

In order to do a fair comparison, we have compared all approximate full adders in terms of delay, power, PDP and error in presence of variation in three different ways. In the first way, all approximate full adders are compared for all input combinations; whereas in the second one, all of them have been used and analyzed in an 8-bit RCA. In the last way, all approximate full adders are evaluated in a Sobel edge detection application for ten 512x512 images.

Table 1
Precise and approximate full adders truth table.

Inputs			CMA	AMA1	AMA2	AMA3	AXA2	AXA3	InXA2	InXA3	VAFA	NFAx	AFA1	AFA2	AFA3
A	B	Cin	CS	CS	CS	CS	CS	CS	CS	CS	CS	CS	CS	CS	CS
0	0	0	00	00	01	01	01	00	00	01	00	01	01	00	00
0	0	1	01	01	01	01	01	01	01	01	01	10	01	00	00
0	1	0	01	10	01	10	00	00	01	01	01	01	01	01	01
0	1	1	10	10	10	10	10	10	11	10	10	10	01	11	01
1	0	0	01	00	01	01	00	00	01	01	01	01	01	01	01
1	0	1	10	10	10	10	10	10	11	10	10	10	10	11	11
1	1	0	10	10	10	10	11	10	01	01	01	11	10	11	11
1	1	1	11	11	10	10	11	11	11	10	10	11	10	11	11

Table 2
Average power, average delay, average PDP and area values for precise and approximate full adders.

Full adders	Power(μw)	Delay(ps)	PDP(aj)	Area
CMA	4.25	36.15	153.88	28
AMA1	3.08	29.38	91.35	20
AMA2	1.74	25.52	44.39	14
AMA3	1.67	24.51	40.83	11
AXA2	1.11	22.77	25.17	6
AXA3	1.19	27.83	33.13	8
InXA2	1.36	32.53	44.25	8
InXA3	1.15	27.47	31.53	6
VAFA	6.42	46.25	289.98	24
NFAx	2.01	16.86	33.99	14
AFA1	1.04	13.14	13.59	8
AFA2	2.19	19.10	41.97	18
AFA3	1.81	16.47	29.74	14

Table 1 compares all of the above full adders in terms of precision for different inputs. The column CS is the full adder outputs, where the LSB is the sum (S) and the MSB is the carry (C). Whenever the approximate full adder output is incorrect, the corresponding output in the CS column is shown in red. Outputs which are depicted in black are correct outputs.

As seen in Table 1, approximate full adders can be divided in two categories in terms of errors. The first category has an exact Cout but an inexact Sum output; whereas the second one has both inexact outputs. The first category includes AMA2, AXA2, AXA3 and AFA2 full adders. The Sum outputs of AMA2 and AXA3 have errors in 25% of cases; whereas for AXA2 and AFA2, the same output has errors in 50% of cases. The second category itself is divided into three subcategories. In the first one, AMA3, InXA2, InXA3, NFAx and AFA1 circuits exhibit 12.5% and 37.5% error for Cout and Sum, respectively. In the second one, AMA1 and VAFA show 12.5% and 25% error for the same outputs, respectively. Finally, in the third subcategory, AFA3 has 12.5% and 50% error for Cout and Sum, respectively. To adequately compare the error parameters of all approximate full adders, we have employed them in an RCA structure and all possible input combinations have been applied in Section 4.4.

3.1. Full adder performance evaluation

In this section, we will compare a precise mirror full adder and twelve approximate full adders in terms of power, delay, PDP and area. The area is reported by the number of transistors in the full adder. All simulations are done in 32 nm PTM technology [22] with HSPICE. To evaluate power, delay and PDP, we consider all possible states due to 3 input signals. In this paper the base inverter has the following sizes: Wn = 64 nm, Wp = 128 nm and Lp = Ln = 32 nm. The other gates and more complex designs sizes are an integer factor of the base size. For instance, in the Nand2 gate we have Wn = 128 nm, Wp = 128 nm and Lp = Ln = 32 nm. For the load capacitance we have considered 4 inverters at the output of all approximate full adders, which means each full adder drives 4 base inverters. Also, we have changed all full addersâ inputs every 250ps.

In Table 2 all the average values of power, delay and PDP in 32 nm PTM technology are presented. Note that between Sum and Cout delays, we choose the greatest value for the reported delay. Based on Table 2, AFA1 has the least power, delay and PDP. To compute Cout, we have used 8 transistors in CMOS logic which have smaller gate and output capacitances

Table 3
Average D2D variation of V_{th} on full adders' power, delay and PDP.

Full adders	Power(w)			Delay(s)			PDP(j)		
	$\mu(10^{-6})$	$\sigma(10^{-6})$	Cv(%)	$\mu(10^{-12})$	$\sigma(10^{-12})$	Cv(%)	$\mu(10^{-18})$	$\sigma(10^{-18})$	Cv(%)
CMA	4.95	1.81	36.49	39.90	12.99	32.56	186.58	50.39	27.01
AMA1	3.61	1.39	38.42	32.16	9.82	30.53	110.97	30.01	27.04
AMA2	2.10	0.96	45.71	28.01	8.78	31.35	55.35	18.01	32.54
AMA3	1.99	0.89	44.70	26.88	7.69	28.59	51.24	18.17	35.46
AXA2	1.34	0.67	50.27	25.94	18.94	73.02	31.19	27.92	89.53
AXA3	1.43	0.75	52.35	31.52	10.78	34.18	45.39	28.36	62.48
InXA2	1.63	0.86	52.91	36.71	44.51	121.24	53.33	83.44	156.47
InXA3	1.39	0.91	65.36	31.12	30.42	97.72	40.37	72.81	180.36
VAFA	7.33	2.51	34.23	49.47	12.33	24.92	342.61	104.22	30.42
NFAx	2.45	1.16	47.34	18.09	4.56	25.21	42.34	15.94	37.65
AFA1	1.28	0.64	49.77	14.38	3.82	26.59	17.63	8.38	47.56
AFA2	2.65	1.17	44.23	20.69	6.44	31.11	51.61	16.29	31.57
AFA3	2.22	1.09	49.19	17.69	4.53	25.60	37.17	14.00	37.67

compared to other approximate full adders. AFA1 compared to AXA2 shows that both of them have almost the same gate and output capacitances; however, as AXA2 has a threshold drop problem, charge and discharge of output capacitances with weak signals (smaller than 0.9v or greater than 0v) require a longer time leading to larger delay. As the output capacitance of AFA1 is less than the other designs, thus its load capacitance is less and its dynamic power will be less too. Also, as designs in CMOS logic have smaller leakage current compared to other logics, AFA1 exhibits less static power as well [18]. The same conclusion can be drawn for AFA3, however as it has one more OR2 gate compare to AFA1, therefore its power, delay and PDP are higher than AFA1.

Compared to CMA, power, delay and PDP of AFA1 have decreased by 76%, 64% and 91%, respectively. Also, compared to the best approximate full adder (AXA2), AFA1 power, delay and PDP are reduced by 6%, 42% and 46%, respectively. AFA2 full adder compared to CMA reduces power, delay and PDP by 48%, 47% and 72%, respectively. Also, AFA3 compared to CMA reduces power, delay and PDP by 57%, 54% and 80%, respectively. The proposed approximate full adders AFA1 and AFA3 always show the least delay compared to other approximate full adders.

3.2. Evaluation of variability effects on approximate full adders

Here, all the designs mentioned in last section will be evaluated with regard to D2D and WID variations. The threshold voltage (V_{th}) is the most important variation parameter in recent technologies [5,7]. Therefore, all evaluations of the full adders focus on this parameter. As mentioned previously, all simulations are done in 32nm PTM technology using HSPICE, for which we have used 1024 Monte Carlo points.

The variation of V_{th} relative to its nominal value is 20% based on a Gaussian distribution [5]. For instance, for each of scenarios a 1024-point Monte Carlo simulation is performed, and the amount of variation is calculated (we will explain how we calculate variation later in the same section). The average of variations of these scenarios will be considered.

To evaluate variation impacts we use some mathematical equations [6]. For this purpose, by using Monte Carlo simulation and Gaussian distribution for V_{th} parameter, power, delay and PDP are calculated. Based on these equations, first the Mean $\mu(x)$, the variance Var(x), and the Standard Deviation $\sigma(x)$ of each of power, delay and PDP, based on simulation results are calculated, and then using Eq. (4) variation coefficient Cv is obtained. For two designs with identical Mean values, smaller variation coefficient shows smaller variation impact on the full adder, which means that full adder is robust against variability.

$$C_v = \frac{\sigma(x)}{\mu(x)} \tag{4}$$

$$\mu(x) = \frac{\sum_i^n x_i}{n} \tag{5}$$

$$Var(x) = \frac{\sum_i^n (x_i - \mu(x))^2}{n-1} \tag{6}$$

$$\sigma(x) = \sqrt{Var(x)} \tag{7}$$

First 1024 Monte Carlo iterations are done for input scenarios. For example, for scenario one 1024 values are obtained for each of power, delay and PDP. To calculate the power variation in this scenario, we first calculate the Mean value (Eq. (5)), and the Standard Deviation (Eq. (7)). Then using Eq. (4), we obtain Cv. The same procedure is used for delay and PDP. Once this is done for all input scenarios, we calculate the average of power Cv, delay Cv and PDP Cv. These values are reported in Tables 3 and 4 for D2D and WID variations.

Table 4
Average WID variation of V_{th} on full adders' power, delay and PDP.

Full adders	Power(w)			Delay(s)			PDP(j)		
	$\mu(10^{-6})$	$\sigma(10^{-6})$	Cv(%)	$\mu(10^{-12})$	$\sigma(10^{-12})$	Cv(%)	$\mu(10^{-18})$	$\sigma(10^{-18})$	Cv(%)
CMA	4.80	0.67	14.00	43.69	7.56	17.31	209.24	45.28	21.64
AMA1	3.53	0.52	14.83	34.04	7.76	22.81	121.31	34.66	28.57
AMA2	2.05	0.45	22.13	28.79	4.96	17.21	58.86	16.95	28.80
AMA3	1.93	0.41	21.07	27.48	4.31	15.69	53.19	13.49	25.37
AXA2	1.39	0.47	33.49	29.71	20.99	70.64	40.71	36.70	90.14
AXA3	1.49	0.63	42.57	35.83	16.93	47.25	56.47	38.39	67.99
InXA2	1.68	0.87	51.57	41.51	45.96	110.72	80.75	128.01	158.52
InXA3	1.44	0.54	37.43	35.39	29.30	82.79	55.90	56.90	101.78
VAFA	7.23	0.86	11.89	50.68	10.17	20.07	358.84	89.06	24.82
NFAx	2.41	0.54	22.41	19.84	3.67	18.50	47.31	12.85	27.16
AFA1	1.25	0.40	32.34	15.12	3.22	21.28	18.65	6.45	34.57
AFA2	2.56	0.44	17.33	22.27	4.40	19.75	57.11	13.96	24.45
AFA3	2.19	0.69	31.39	18.80	3.44	18.30	41.17	14.45	35.10

As a matter of fact, based on HSPICE manual the relation between the relative error and the number of Monte Carlo iterations is as follows [23]:

$$RelativeError = \frac{1}{\sqrt{NumberofMonteCarloIterations}} \tag{8}$$

Based on Eq. (8) and NumOfMC=1024, the relative error is about 3% which is relatively small. According to this manual, if the circuit operates correctly for all 1024 iterations, there is a 99 percent probability that over 96% of all possible component values also operate correctly. Based on simulations, when 10,000 Monte Carlo iterations are performed, Mean, Standard Deviation, and Cv values have between 10^{-3} to 10^{-4} difference with 1024 Monte Carlo iterations. Hence, the accuracy of results is almost the same in both cases; however, the simulation time of 10,000 Monte Carlo iterations is 10X longer.

As we know, NMOS transistors pass 0 logic correctly, but reduce 1 logic by V_{thn}. On the contrary, PMOS transistors increase 0 logic by $|V_{thp}|$ [18]. Therefore, all pass transistor-based full adders (AXA2, AXA3, InXA2, InXA3) observe a threshold drop at their outputs Sum and Cout for certain inputs. For instance, in InXA2 for inputs A=1 and B=Cin=0, the value of Cout is 0.16v instead of 0v, which is the same as $|V_{thp}|$ value in 32nm technology. The value of Sum is 0.74v instead of 0.9v, which is the same as $V_{dd} - V_{thn}$ (V_{dd}=0.9v and V_{thn}=0.16v). Hence, putting these full adders in series (like RCA) or in tree (like array multipliers), taking into account threshold drop, may change a 0 logic to a 1 logic and vice versa.

Tables 3 and 4 show the effect of V_{th} D2D and WID variations on power, delay and PDP of the full adders. In Tables 3 and 4, green values show the least variability, blues values are the second least variability, and red values depict the most variability. In terms of power and PDP variations, VAFA is ranked first and second, respectively. In terms of D2D delay variation, VAFA is ranked first and in terms of WID delay variation, AMA3 is ranked first. The good performance of VAFA in terms of variability is due to its higher average power consumption and average delay; however this leads to decreasing Cv. Hence, this approximate full adder is not a suitable choice for imprecision-tolerant applications. Based on Table 1, AFA1 and InXA3 have almost the same precision. However, based on Tables 3 and 4, in terms of D2D and WID variations for all power, delay and PDP parameters, AFA1 is far better than InXA3. For instance for PDP parameter, D2D and WID variations in AFA1 are 47.56% and 34.57% respectively; whereas for InXA3 the same values are 180.36% and 101.78%. In addition of better variability, AFA1 has no threshold drop issue compared to InXA3. As Tables 3 and 4 show, pass transistor-based full adders are the most sensitive to variation, due to the threshold drop at their outputs.

From the statistical point of view, for two designs with identical Mean values, the one that has greater Standard Deviation is more sensitive to variations. From the circuit point of view this means that for two similar designs, there exist some input scenarios that activate some paths in the design causing a significant increase or decrease of delay or power. For instance, in pass-transistor based designs for some input scenarios a threshold drop could occur, causing a sharp increase of delay and power. This leads to an increase of Standard Deviation and subsequently to a greater sensitivity to variations. Therefore, if we want to design a robust design against variations, we must have a design which for different input scenarios has power and delay values close to their average values, leading to a small standard deviation and thus less variability.

4. Evaluation of approximate RCA using approximate full adders

Here, we will investigate the effects of twelve approximate full adders described in the previous section to implement RCA adders. To evaluate an approximate adder performance, error and variation effects we have implemented an 8-bit RCA adder, where for different bits an approximate full adder has been used. We have considered four scenarios for implementing the RCA, where NAB is defined as the number of approximate bits. In the first one, namely NAB1, an approximate full adder is employed only for the least significant bit. In the second one, namely NAB2, approximate full adders are employed

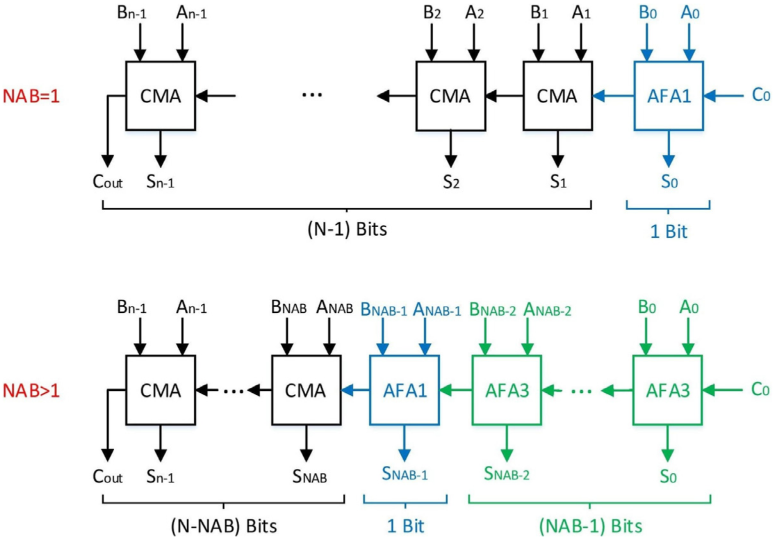

Fig. 3. The structure of N bit BestAFA.

for the two least significant bits. Similarly, in the fourth scenario, namely NAB4, approximate full adders are used for the four least significant bits. We have considered not only RCA structures based on one type of approximate full adders, but also structures based on multiple types of full adders from one family. Families of full adders could be AFAs (AFA1, 2, 3) or AMAs (AMA1, 2, 3), etc.

4.1. Proposed best RCA structure based on AFAs family

In this section, we have considered the combination of two AFAs and AMAs families, as these families have better performance in terms of power, delay, precision and variability compared to others. To find the best configuration for RCA with different NAB and N values, we have evaluated all combinations of approximate full adders in terms of power, delay and error parameters. According to our analyses, combining different AMAs full adders will always increase error parameters compared to the case where only one type of AMAs is used (error parameters are described in Section 4.4). For example for NAB=4, the best combination of AMAs has the normalized mean error distance (NMED) equal to 0.00498, and mean relative error distance (MRED) equal to 0.01412; whereas AMA1 has NMED=0.00479 and MRED=0.01362. In contrast, using a given combination of different types of AFAs will always cause error parameters reduction. We have called this optimized RCA structures ”BestAFA” in this paper. The structure of N bit BestAFA is shown in Fig. 3. In the proposed N-bit BestAFA adder, for NAB=1, an AFA1 full adder is used for the least significant bit and for the rest of the bits the exact full adder CMA is employed. For NAB $>$ 1, BestAFA is built from AFA3, AFA1, and CMA. For bits 0 to NAB-2 AFA3, for bit NAB-1 AFA1, and for bits NAB to N-1 CMA have been used. As a matter of fact, another structure similar to BestAFA in terms of error parameters, built with another combination of AFAs, has also existed and could have been considered; however its power, delay, PDP, and area were higher than BestAFA, and thus we ignored it in this paper.

4.2. Approximate RCA performance evaluation

In this section, we will evaluate power, delay, PDP and area of exact and approximate RCA adders. As discussed earlier, pass transistor-based approximate full adders encounter threshold drop issue at the output, and specifically for Cout of next stages. To alleviate this problem, we have used a minimum size buffer at the output, which increases power, delay and area of these adders.

To evaluate the adders performance, we have applied input scenarios that activate the critical path and cause the carry propagation from the first to the eighth bit. Table 5 shows NAB3 results for power, delay, PDP and area. The area is reported by the number of transistors in the adder. The approximate adder based on AFA1 has the least power, delay, PDP and area. The adder made with VAFA has the highest power, delay, PDP and area. We can also see that for NAB3, BestAFA is ranked first, second and third in terms of delay, PDP and power, respectively.

4.3. Approximate RCA variability evaluation

Four 8-bit RCA adders, where from first bit to the fourth bit approximate full adders have been used, are considered. To evaluate V_{th} variation impacts, different input scenarios have been conceived, where carry propagates from first bit to the

Table 5
Average power, average delay, average PDP, area, NMED and $PDP \times Area \times NMED$ values for 8-bit precise and approximate RCAs (NAB3).

RCA structures	Power(μw)	Delay(ps)	PDP(fj)	Area	NMED	$PDP \times Area \times NMED$
CMA	23.06	269.10	6.21	224	–	–
AMA1	20.32	201.40	4.09	200	0.002757	2.26
AMA2	18.85	244.90	4.62	182	0.00337	2.83
AMA3	18.50	201.40	3.73	173	0.004412	2.85
AXA2	17.41	245.10	4.27	182	0.005882	4.57
AXA3	18.55	253.90	4.71	188	0.004902	4.34
InXA2	21.77	255.03	5.55	188	0.003431	3.58
InXA3	22.56	259.60	5.86	182	0.005147	5.49
VAFA	26.56	276.16	7.33	212	0.003431	5.33
NFAx	19.01	231.23	4.39	182	0.005760	4.60
AFA1	17.12	201.33	3.45	164	0.004044	2.29
AFA2	19.33	230.87	4.46	194	0.004289	3.71
AFA3	18.31	201.33	3.69	182	0.003431	2.30
BestAFA	17.93	201.33	3.61	176	0.002696	1.71

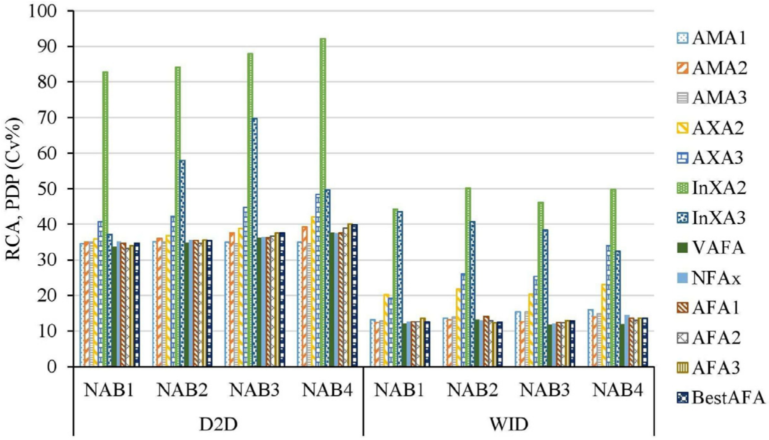

Fig. 4. D2D and WID PDP variation impacts on V_{th} for RCA adders for various NAB values.

last one. These scenarios can be different from each other according to the type of approximate full adders that have been used. For each scenario, we calculate its variation based on equations presented in Section 3. Then, the average of variations in all scenarios of a given adder is presented as its variation. Figs. 4 to 5 show PDP and delay variation impacts on V_{th} for different approximation bits in an 8-bit RCA adder. As expected RCA designs based on pass transistors are more sensitive to D2D and WID variations compared to other adders, which have closer sensitivities to each other.

Based on Fig. 4, pass transistor-based full adders have the highest PDP variation specifically in WID state, whereas other full adders (AFAs and AMAs) have almost similar variabilities. Fig. 5 illustrates the delay sensitivity of different full adders in presence of V_{th} variation. Specifically in WID state, InXA2 and InXA3 are three times more sensitive compared to AMAs and AFAs full adders. Therefore, these full adders in presence of variation could have a far higher delay compared to the critical path delay with no variation; thus they are not very reliable.

Based on Fig. 4, in terms of D2D PDP variation the least average Cv belongs to AMA3 (34.7%), and in terms of WID PDP variation the values are 12.3%, 12.7% and 12.9% for VAFA, AFA2 and BestAFA. Also according to Fig. 5, in terms of D2D delay variation the least average Cvs belongs to NFAx (28.13%), and in terms of WID delay variation the value is 9.09% for NFAx.

4.4. Approximate RCA error evaluation

To evaluate an approximate adder errors, we consider the error rate (ER), the normalized mean error distance (NMED), and the mean relative error distance (MRED) as three parameters defined in [17]. These parameters are formulated as follows, where n shows the total number of input states for an 8-bit adder.

$$ER = \frac{NumberofErroneosOutputs}{n} \tag{9}$$

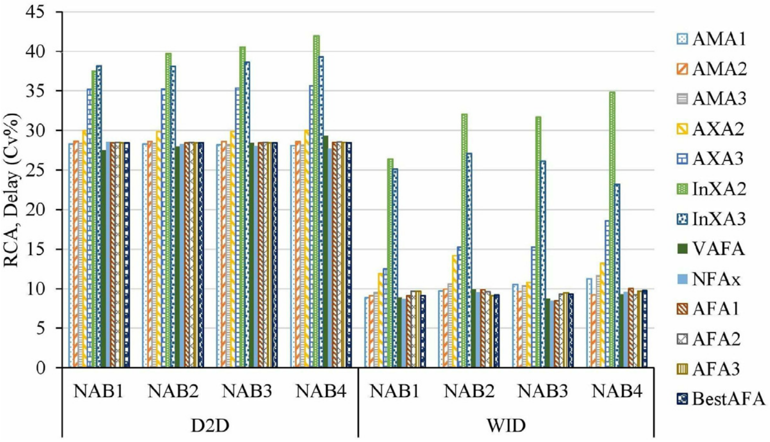

Fig. 5. D2D and WID Delay variation impacts on V_{th} for RCA adders for various NAB values.

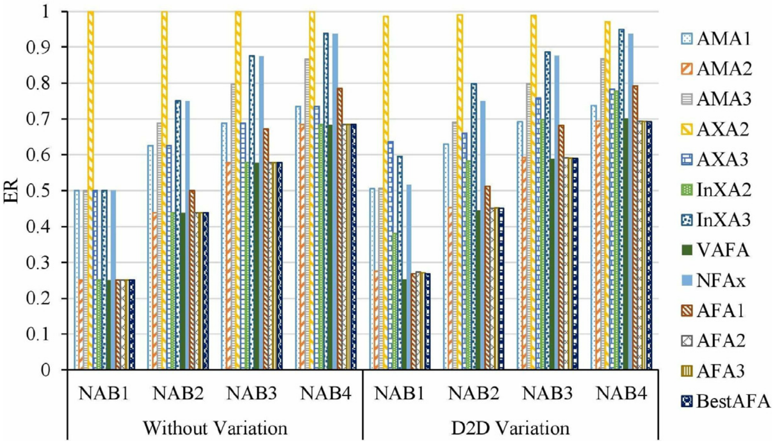

Fig. 6. ER of different 8-bit adders for various NAB values (without variation and with D2D variation).

$$NMED = \frac{\frac{1}{n}\sum_{i=1}^{n} |ExactOutput_i - ApproximateOutput_i|}{ExactOutput_{Max}} \tag{10}$$

$$MRED = \frac{1}{n}\sum_{i=1}^{n} \frac{|ExactOutput_i - ApproximateOutput_i|}{ExactOutput_i} \tag{11}$$

We have simulated an 8-bit unsigned adder for all possible inputs. Results are shown in Figs. 6 to 8. In order to take into account the delay variation effects of full adders in the evaluation of error parameters (ER, NMED, MRED), we consider a maximum delay $Delay_{\max} = \alpha * Delay_{CP}$ for each adderâs delay, where $Delay_{CP}$ is the critical path delay of each adder. α is the ratio of the maximum delay of an exact RCA in presence of process variation, to the maximum delay of an exact RCA in absence of process variation. In the Monte Carlo simulation results, the outputs that cause this value to be exceeded are considered as error. The average number of errors for all input scenario is considered as the adderâs error percentage in presence of variation. Hence, the evaluation of error parameters is based on variability, where the results are shown in Figs. 6 to 8. Based on Fig. 5, as D2D variation effect on delay is almost 2.5 times greater than that of WID variation, we have depicted only the D2D variation effect on error in Figs. 6 to 8. In these figures, the left-hand side illustrates error parameters with no variability, and the right-hand side with D2D variation.

As seen on the left-hand side of Fig. 6, the error rates of the six adders based on AMA2, InXA2, VAFA, AFA2, AFA3, and BestAFA are equal and smaller than that of other adders. As depicted on the left-hand side of Fig. 7, and Fig. 8, for NAB=1, 2, 3, BestAFA adder and for NAB=4 AMA1 adder exhibit the least NMED and MRED.

In AXA2 for all the cases where Cin=0, the output Sum is erroneous (see Table 1). Thus, in the adder based on AXA2, as the input carry is 0 for the first bit, when adding two arbitrary unsigned numbers, at least the first bit becomes erroneous. This is why this adder has the most ER, NMED, and MRED (see Figs. 6 to 8).

Referring to the right-hand side of Figs. 6 to 8, D2D variations increase error in different adders; however for pass transistor-based adders the amount of error is greater, as these adders are more sensitive to variations. In presence of

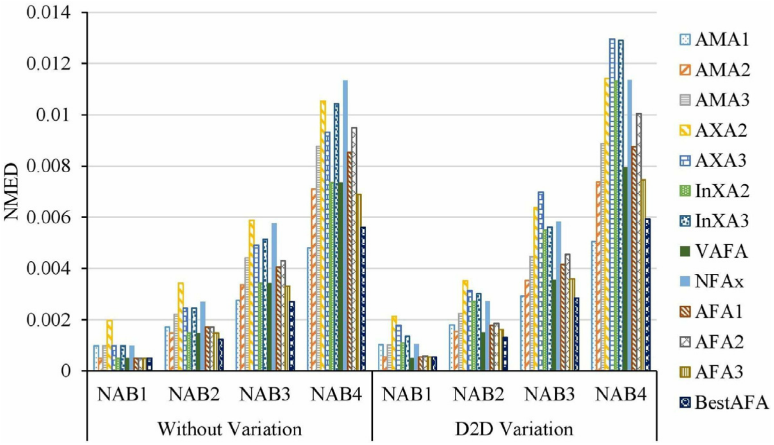

Fig. 7. NMED of different 8-bit adders for various NAB values (without variation and with D2D variation).

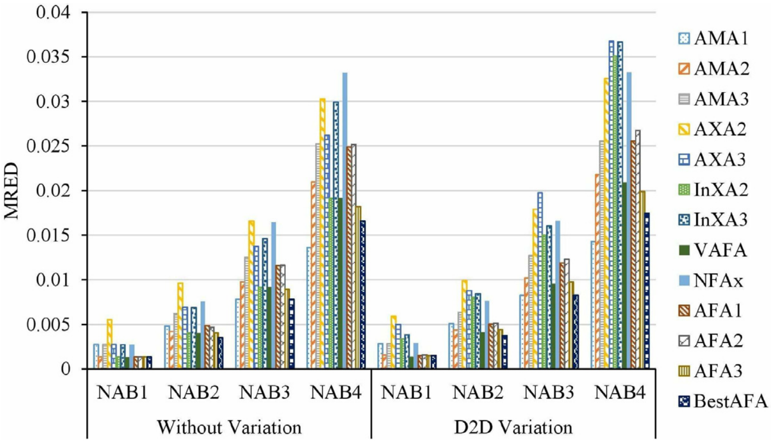

Fig. 8. MRED of different 8-bit adders for various NAB values (without variation and with D2D variation).

variations, VAFA has the least error rate. In terms of NMED and MRED, for NAB=1 VAFA, for NAB=2, 3 BestAFA and for NAB=4 AMA1 have the least errors.

As seen in Table 5, the approximate RCA based on AFA1 has the least power, delay, PDP and area. Also, BestAFA has the lowest delay and NMED. When all parameters namely power, delay, PDP, area and NMED are considered simultaneously ($PDP \times Area \times NMED$), BestAFA is the best. Based on $PDP \times Area \times NMED$ criterion, comparing RCAs in presence and absence of variations, for NAB=1, 2, 3 BestAFA and for NAB=4 AMA1 have always the best performance, whereas InXA3 has the worst performance. Therefore, in terms of power, delay, area and error parameters in presence and absence of variations, BestAFA is the best trade-off.

In Fig. 9, the Pareto chart is illustrated for an 8-bit RCA with NAB=3 in presence of D2D variation. The X-axis and Y-axis show the normalized values of PDP and NMED in presence of D2D variation, respectively. PDP values are normalized to the PDP value of the adder based on VAFA, and NMED values are normalized to the NMED value of the adder based on AXA3. According to Fig. 9, in terms of PDP-NMED Pareto, BestAFA has the best performance and AMA1, AFA3, and AFA1 are ranked from second to fourth place, respectively. InXA3 exhibits the worst performance.

5. Imprecision-tolerant real application simulation results

As a real application we have considered the Sobel edge detection algorithm. In this algorithm, two 3x3 windows are convolved with the original image. In the Axbench benchmark a hardware implementation of this algorithm is proposed using, addition, subtraction and shift operations [24]. In this paper, we have used the same implementation for 512x512 images.

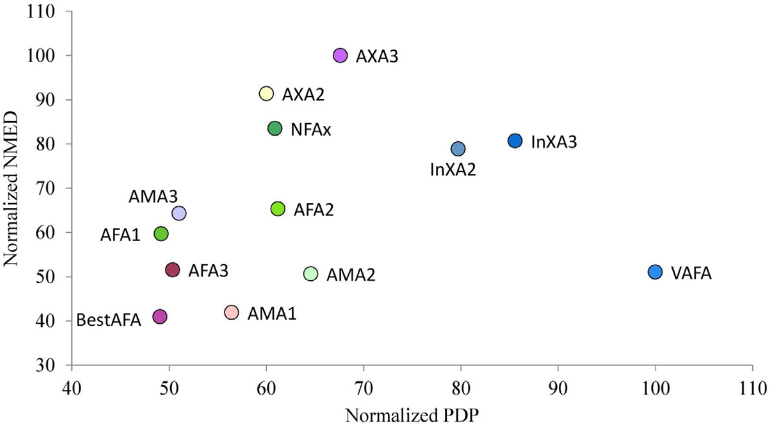

Fig. 9. Normalized PDP-NMED Pareto of different 8-bit adders (NAB3 with D2D variation).

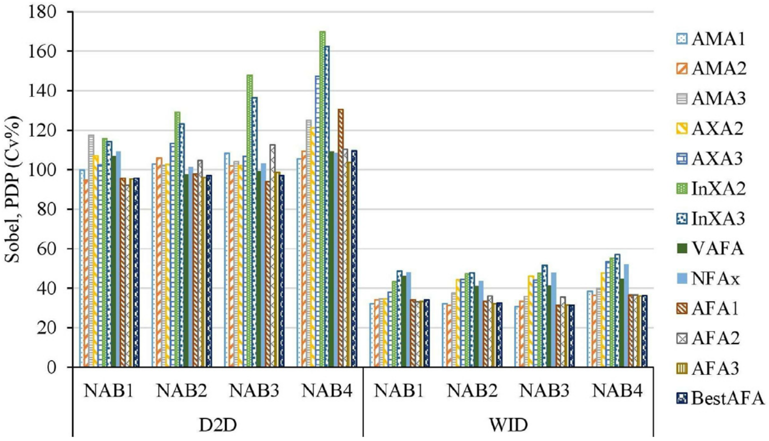

Fig. 10. D2D and WID PDP variation impacts on V_{th} for Sobel benchmark for various NABs.

5.1. Full adders variability evaluation with sobel edge detection benchmark

We have implemented the Sobel algorithm in the Axbench benchmark in HSPICE. Instead of the existing adders and subtractors in this algorithm, we have used our own approximate adders and subtractors based on RCA. To evaluate this algorithm we have used ten 512x512 images, and the average result of these ten images is presented as the variability value of this algorithm. As in the previous sections, we have only considered D2D and WID V_{th} 20% variations, on Sobel edge detection algorithm.

Figs. 10 and 11 show PDP and delay variations for the Sobel algorithm. In these figures, in terms of D2D variation AFAs family has the least variability for delay and PDP, whereas InXA2 has the highest variability. Based on Fig. 10, in terms of D2D PDP variation the least average Cvs belong to AFA3, and BestAFA (98.4%, 99.8%, respectively), and in terms of WID PDP variation the values are 33.3% and 33.4% for AFA3 and AMA1. Also Fig. 11 shows that in terms of D2D and WID delay variations, the least average Cvs belong to AFA3 and BestAFA (46.3%, 46.6% for D2D, and 29.3%, 29.6% for WID, respectively). Knowing that in real applications such as Sobel, the output quality depends on the input image, and that these images have values with different ranges with possible non-uniform distribution, in terms of variability VAFA and NFAx do not have the best performance; however, their results are close to the best results.

5.2. Full adders error evaluation with sobel edge detection benchmark

In image processing algorithms, such as Sobel, some of the most important parameters to compare different methods are PSNR, MSE and MSSIM [17,25], explained as follows:

Mean Square Error (MSE):

$$MSE = \frac{1}{n \times m}\sum_{i=1}^{m}\sum_{j=1}^{n}(P_{i,j} - \widehat{P_{i,j}})^2 \tag{12}$$

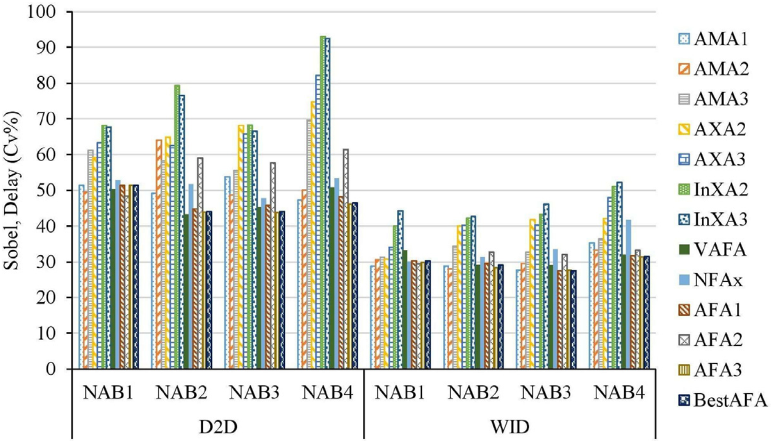

Fig. 11. D2D and WID Delay variation impacts on V_{th} for Sobel benchmark for various NABs.

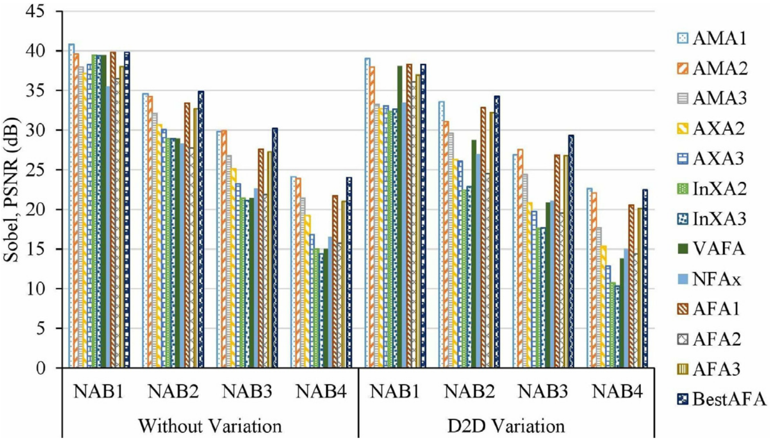

Fig. 12. PSNR of different adders in Sobel edge detection application for various NAB values (without variation and with D2D variation).

Peak Signal to Noise Ratio (PSNR):

$$PSNR = 10Log\frac{255^2}{MSE} \tag{13}$$

Mean Structural SIMilarity Index (MSSIM):

$$MSSIM = \frac{1}{n \times m}\sum_{i=1}^{m}\sum_{j=1}^{n} SSIM(P_{i,j}, \widehat{P_{i,j}}) \tag{14}$$

In the above equations, $P_{i,j}$ expresses the exact pixel value in the ith row and jth column of the produced exact image; whereas $\widehat{P_{i,j}}$ expresses the approximate pixel value in the ith row and jth column of the produced approximate image. m and n are the size of row and column of the image, respectively.

Again, to compare approximate full addersâ errors in Sobel algorithm, ten 512x512 images are used and their average PSNR and MSSIM are presented as the error of each full adder. To evaluate variability effects on PSNR and MSSIM in Sobel algorithm, we have used D2D delay variations (see Fig. 11). Figs. 12 and 13 illustrate the average PSNR and MSSIM in the absence and presence of D2D variation. According to Fig. 12, for all approximate RCAs and different NABs, D2D variations reduce PSNR by 9.8% in average. The least PSNR reduction in average belongs to AFAs family (4.5%) and the most increase in average belongs to InAXs family (21%). AFA3 has the least average (2.5%), whereas InAX2 has the most average (21.7%) PSNR reductions. According to Fig. 13, for all approximate RCAs and different NABs, D2D variations reduce MSSIM by 3.5% in average. The least MSSIM reduction in average belongs to AFAs family (2.3%) and the most increase in average belongs to InAXs family (5.7%). AFA2 has the least average (1.2%), whereas InAX2 has the most average (6.1%) MSSIM reductions.

Figs. 12 and 13 show AMA1 and BestAFA always exhibit the highest PSNR and MSSIM in Sobel algorithm. With or without variability, BestAFA for NAB=2, 3, and AMA1 for NAB=1, 4 have the highest PSNR and MSSIM. According to Table 5, BestAFA

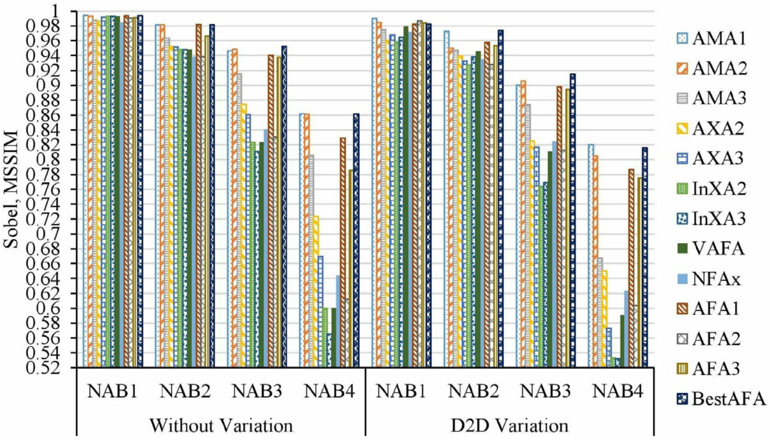

Fig. 13. MSSIM of different adders in Sobel edge detection application for various NAB values (without variation and with D2D variation).

is better than AMA1 in terms of power, delay, PDP and area. Therefore overall, BestAFA is the best implementation choice for the Sobel algorithm.

6. Conclusion

We analyzed D2D and WID V_{th} variation impacts for seven approximate full adders, and proposed three new approximate full adders, namely AFA1, AFA2 and AFA3. Also, we proposed a new approximate RCA, BestAFA, based on AFA1 and AFA3. According to Monte Carlo simulation results, all CMOS approximate full adders have shown less variability, whereas full adders based on pass transistors are very sensitive to variations. In Sobel algorithm, BestAFA is among the best adders with regard to PSNR and MSSIM, and D2D and WID variations. We have concluded that overall BestAFA presents the best trade-off in terms of power, delay, PDP, area and PSNR in presence or absence of variations. As a future work, we can use the existing methods to reduce the variation effects on the approximate adders.

Authorship statement

Conception and design of study: M. Mirzaei, S. Mohammadi; acquisition of data: M. Mirzaei; analysis and/or interpretation of data: M. Mirzaei, S. Mohammadi.

Drafting the manuscript: M. Mirzaei, S. Mohammadi; revising the manuscript critically for important intellectual content: M. Mirzaei, S. Mohammadi.

Approval of the version of the manuscript to be published (the names of all authors must be listed): M. Mirzaei, S. Mohammadi.

Declaration of Competing Interest

The authors certify that they have NO affiliations with or involvement in any organization or entity with any financial interest (such as honoraria; educational grants; participation in speakers bureaus; membership, employment, consultancies, stock ownership, or other equity interest; and expert testimony or patent-licensing arrangements), or non-financial interest (such as personal or professional relationships, affiliations, knowledge or beliefs) in the subject matter or materials discussed in this manuscript.

Acknowledgment

This research was in part supported by a grant from the Institute for Research in Fundamental Sciences (IPM) (Grant No. CS1398-4-14).

References

[1] Laurenzano MA, Hill P, Samadi M, Mahlke S, Mars J, Tang L. Input responsiveness: using canary inputs to dynamically steer approximation. ACM SIGPLAN Notices 2016;51(6):161–76. doi:10.1145/2908080.2908087.

[2] Esmaeilzadeh H, Sampson A, Ceze L, Burger D. Architecture support for disciplined approximate programming. In: ACM SIGPLAN Notices, 47. ACM; 2012. p. 301–12. doi:10.1145/2150976.2151008.

[3] Mittal S. A survey of techniques for approximate computing. ACM Comput Surv (CSUR) 2016;48(4):62. doi:10.1145/2893356.

[4] Jiang H, Liu C, Liu L, Lombardi F, Han J. A review, classification, and comparative evaluation of approximate arithmetic circuits. ACM J Emerg Technol ComputSyst (JETC) 2017;13(4):60. doi:10.1145/3094124.

[5] Mirzaei M, Mosaffa M, Mohammadi S. Variation-aware approaches with power improvement in digital circuits. Integration VLSI J 2015;48:83–100. doi:10.1016/j.vlsi.2014.07.001.

[6] Mirzaei M, Mosaffa M, Mohammadi S, Trajkovic J. Power and variability improvement of an asynchronous router using stacking and dual-vth approaches. In: Digital system design (DSD), 2013 euromicro conference on. IEEE; 2013. p. 327–34. doi:10.1109/DSD.2013.41.

[7] Adl SMT, Mirzaei M, Mohammadi S. Elastic buffer evaluation for link pipelining under process variation. IET Circuits, Devices Syst 2018;12(5):645–54. doi:10.1049/iet-cds.2017.0394.

[8] Yang T, Ukezono T, Sato T. A low-power configurable adder for approximate applications. In: 2018 19th international symposium on quality electronic design (ISQED). IEEE; 2018. p. 347–52. doi:10.1109/ISQED.2018.8357311.

[9] Soares LB, da Rosa MMA, Diniz CM, da da Costa EAC, Bampi S. Design methodology to explore hybrid approximate adders for energy-efficient image and video processing accelerators. IEEE Trans Circuits Syst I Regul Pap 2019. doi:10.1109/TCSI.2019.2892588.

[10] Wu Y, Li Y, Ge X, Gao Y, Qian W. An efficient method for calculating the error statistics of block-based approximate adders. IEEE Trans Comput 2019;68(1):21–38. doi:10.1109/TC.2018.2859960.

[11] Akbari O, Kamal M, Afzali-Kusha A, Pedram M. Rap-cla: a reconfigurable approximate carry look-ahead adder. IEEE Trans Circuits Syst II Express Briefs 2018;65(8):1089–93. doi:10.1109/TCSII.2016.2633307.

[12] Almurib HA, Kumar TN, Lombardi F. Approximate dct image compression using inexact computing. IEEE Trans Comput 2018;67(2):149–59. doi:10.1109/TC.2017.2731770.

[13] Vahdat S, Kamal M, Afzali-Kusha A, Pedram M. Letam: a low energy truncation-based approximate multiplier. Comput Electr Engineering 2017;63:1–17. doi:10.1016/j.compeleceng.2017.08.019.

[14] Gupta V, Mohapatra D, Park SP, Raghunathan A, Roy K. Impact: imprecise adders for low-power approximate computing. In: Proceedings of the 17th IEEE/ACM international symposium on Low-power electronics and design. IEEE Press; 2011. p. 409–14. http://dl.acm.org/citation.cfm?id=2016802.2016898.

[15] Gupta V, Mohapatra D, Raghunathan A, Roy K. Low-power digital signal processing using approximate adders. IEEE Trans Comput Aided Des Integr Circuits Syst 2013;32(1):124–37. doi:10.1109/TCAD.2012.2217962.

[16] Yang Z, Jain A, Liang J, Han J, Lombardi F. Approximate xor/xnor-based adders for inexact computing. In: Nanotechnology (IEEE-NANo), 2013 13th IEEE conference on. IEEE; 2013. p. 690–3. doi:10.1109/NANO.2013.6720793.

[17] Almurib HAF, Kumar TN, Lombardi F. Inexact designs for approximate low power addition by cell replacement. In: Proceedings of the 2016 Conference on Design, Automation & Test in Europe. San Jose, CA, USA: EDA Consortium; 2016. p. 660–5. ISBN 978-3-9815370-6-2. http://dl.acm.org/citation.cfm?id=2971808.2971962

[18] Weste NH, Harris D. CMOS VLSI Design: a circuits and systems perspective. Pearson Education India; 2015.

[19] Yang Z, Han J, Lombardi F. Transmission gate-based approximate adders for inexact computing. In: Nanoscale architectures (NANOARCH), 2015 IEEE/ACM international symposium on. IEEE; 2015. p. 145–50. doi:10.1109/NANOARCH.2015.7180603.

[20] Venkatachalam S, Ko S-B. Design of power and area efficient approximate multipliers. IEEE Trans Very Large Scale Integr VLSI Syst 2017;25(5):1782–6. doi:10.1109/TVLSI.2016.2643639.

[21] Waris H, Wang C, Liu W. High-performance approximate half and full adder cells using nand logic gate. IEICE Electron Express 2019:16–20190043. doi:10.1587/elex.16.20190043.

[22] Nanoscale integration and modeling (nimo) group, predictive technology model (ptm), note = http://ptm.asu.edu/, last accessed may 8, 2019.

[23] Basic Simulation and Analysis. Synopsys, HSPICE User Guide; 2013. https://www.synopsys.com/.

[24] Yazdanbakhsh A, Mahajan D, Esmaeilzadeh H, Lotfi-Kamran P. Axbench: a multiplatform benchmark suite for approximate computing. IEEE Design & Test 2017;34(2):60–8. doi:10.1109/MDAT.2016.2630270.

[25] Wang Z, Bovik AC, Sheikh HR, Simoncelli EP, et al. Image quality assessment: from error visibility to structural similarity. IEEE Trans Image Process 2004;13(4):600–12. doi:10.1109/TIP.2003.819861.

Mohammad Mirzaei received his BSc. and MSc. degrees from the University of Mazandaran and the University of Tehran in 2010 and 2013, respectively, all in computer engineering. He is currently a Ph.D. student at the University of Tehran. His research interests include approximate computing, process variation, low power and asynchronous system design, verification and on-chip interconnects in GALS NoCs.

Siamak Mohammadi received his BSc, MSc and PhD degrees from the University of Paris-Sud in 1990, 1992 and 1996, respectively, all in electrical engineering. From 1997 to 1999 he was a Research Associate at the University of Manchester. Then, he moved to Canada and worked in industry until 2005. Currently, he is an Associate Professor at the University of Tehran.