Microprocessors and Microsystems 78 (2020) 103237

Contents lists available at ScienceDirect

Microprocessors and Microsystems

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

High performance compact energy efficient error tolerant adders and multipliers for 16-bit image processing applications

R. Jothin [a,*], M. Peer Mohamed [b], C. Vasanthanayaki [c]

[a] Department of Electronics and Communication Engineering, Infant Jesus College of Engineering, Thoothukudi, Tamilnadu, India
[b] Research Scholar, Anna University, Chennai, Tamilnadu, India
[c] Department of Electronics and Communication Engineering, Government College of Engineering, Salem, Tamilnadu, India

ARTICLE INFO

Keywords:
Compact
Energy Efficiency
CEETA
SAFA2E
PSNR
ASIC
VLSI
SAFA1E
AC

ABSTRACT

Compact design is an extremely important criterion in the recent development error tolerant applications based on the high performance processor core. The performance of the processor core depends upon the data processing sub-system architectures. Area, delay and power reduction in the cost of accuracy have become the critical requirement of high quantity data computing Very Large Scale Integration (VLSI) architectures. In this paper, we proposed Compact Energy efficient Error Tolerant Adders (CEETAs) which have efficient design metrics for data intensive applications. To achieve area and energy efficiency, Simplified gate level Approximate Full Adders (SAFAs) are proposed in the inaccurate part of the CEETA and CEETA1 designs. The simulation result shows that the proposed SAFAs based CEETA1 adder exhibits low power consumption, less Power-Delay Product (PDP), less Area-Delay Product (ADP) and it offers a savings of 51.63%, 43.87%, 48.57%, 36.52%, 36.84%, 15.72%, 18.18% area than the conventional CSLA, SAET-CSLA, ETCSLA, HSETA, HSSSA, HPETA-I, HPETA-II, respectively. Further, the Simplified Approximate Full Adders (SAFA1E and SAFA2E), 4-2 Approximate Compressor (AC) modules based High Performance Error Tolerant Multipliers (HPETMs) are proposed for error tolerant applications. To achieve energy and area efficiency with high speed for the high quantity digital data computation, the propagation delay and the gate count reduction on the carry generation path are proposed in the SAFA and AC designs. The proposed HPETM1 has a significant amount of power and area savings and it exhibits 24.95%, 29.87%, 30.41%, 31.79%, 31.68%, 33.87%, and 35.58% lesser delay than the existing AM1, AM2, SSM, ACM1, ACM2, ACM3 and CDM respectively.

1. Introduction

Minimizing the power consumption of circuits is important for a wide variety of high quantity digital data computing applications, because of the increasing levels of integration and the desire for portability. Since performance is often limited by the speed of arithmetic components, it is also important to maximize the speed. Power reduction has to be addressed at every design level, like gate and transistor-level technology where most of the power can be saved at the high level of abstraction. At the gate level of high performance architectures, an optimized compact design is desired to achieve energy efficiency, high speed, and to be reliable for high quantity digital data computing applications. Good driving capability under different load conditions and balanced output in order to avoid glitches is also important. Since the modules are duplicated in large numbers, layout regularity, and interconnect complexity are vital. The best way to reduce the junction capacitance, as well as the overall gate capacitance, is to optimize the gate count for a particular performance. Several optimization techniques have been proposed to minimize the area of the design while maintaining the performance. They are path based optimization and global optimization in the design. In path-based optimization, gates in the critical paths are upsized to achieve the desired performance, while the gates counts in the off critical paths are reduced to achieve low power consumption. In the global optimization, all gates counts in the architecture are globally optimized for a given delay. The architectures are mostly designed for the highest performance to satisfy the overall system cycle time requirements. They are composed of large and highly parallel architectures with logic regularity. As such, the leakage power consumption is substantial for such architectures. However, every application does not require a fast circuit to operate at the highest

* Corresponding author.
E-mail address: joejothin@gmail.com (R. Jothin).

https://doi.org/10.1016/j.micpro.2020.103237
Received 16 April 2020; Received in revised form 24 June 2020; Accepted 18 August 2020
Available online 20 August 2020

Table 1
Truth table of conventional and existing approximate full adders.

INPUTS			CFA		AFA-2 [2] / IFA [17]			AFA [4]			MFA [12]			MBAFA-I [18]			MBAFA-II [18]		
A	*B*	*C*	*CY*	*S*	*CY*	*S*	*ED*	*CY*	*S*	*ED*	*CY*	*S*	*ED*	*CY*	*S*	*ED*	*CY*	*S*	*ED*
0	0	0	0	0	0	1	+1	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
0	1	0	0	1	0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
0	1	1	1	0	1	0	0	1	0	0	0	0	-2	0	1	-1	0	1	-1
1	0	0	0	1	0	1	0	0	1	0	1	1	+2	1	0	+1	0	1	0
1	0	1	1	0	1	0	0	1	0	0	1	0	0	1	0	0	0	1	-1
1	1	0	1	0	1	0	0	0	1	-1	1	0	0	1	0	0	1	0	0
1	1	1	1	1	1	0	-1	1	0	-1	1	1	0	1	1	0	1	1	0

performance level all the time. Different circuit techniques have been proposed to reduce leakage energy utilizing the timing slack without impacting performance. These techniques can be categorized based on when and how they utilize the available timing slack. Dual threshold voltage statically assigns high threshold voltage to some transistors of logic gates in non-critical paths at the design time so as to reduce leakage current. The techniques, which utilize the slack in runtime, can be divided into two groups depending on reduced standby leakage or active leakage. Standby leakage reduction techniques can put the entire system in a low leakage mode when the computation is not required. Active leakage reduction techniques slow down the system by dynamically changing the threshold voltage to reduce leakage when maximum performance is not needed. In active mode, the operating temperature increases due to the switching activities of transistors. This has an exponential effect on sub-threshold leakage making this the dominant leakage component during the active mode and amplifying the leakage power. The logic gates optimization and switching activity in the critical path basically influences the area, speed, power dissipation, and the wiring complexity of a chip level VLSI system.In a VLSI system, application-specific digital signal processing architecture has been implemented for a high quantity digital data computing applications [29]. The performance of the digital signal processor core is arisen, which depend on the configuration, design parameters and effective utilization of the data path and on-chip memory architectures. The performance of the most critical functional units in the data path unit is totally dependent on being adders. While considering the elementary structure of error tolerant applications, it is a combination of the multipliers and delays, which in turn are the combination of the adders in the data path unit. If adders are too slow or consume more energy, the overall performance of the processor will be degraded. Initially, a Conventional Full Adder (CFA) was used to operate at high accuracy rates for image processing applications [30]. After that in VLSI design, approximation techniques were developed that can be found in parallel or block processing either to increase the effective throughput or to reduce the power consumption at moderate accuracy rates for the human perception. Vision and hearing are the two means by which humans perceive the outside world. It is estimated that 75% of the information received by human is visual. As the human visual system identifies thousands of images based on color, it is one of the most dominant and distinguishable visual features in an image. The visual perception of color starts with achromatic light source, capable of emitting electromagnetic radiation with wavelengths approximately between 400 and 700 nm. Traditionally, the error tolerant application system involves a number of approximation adder modules for parallel processing implementation [31].The critical path delay depends upon the delay component of the propagated carry output signal and relative size of the adders. Carry propagation delay plays a key role to achieve high speed of computation in the adders design. The carry propagation delay introduced by a single bit full adder increases the overall delay component in the multiple bits Ripple Carry Adder (RCA). A Carry Select Adder (CSLA) and Carry Look ahead Adder (CLA) structures are introduced to reduce the carry propagation delay at the cost of area and energy. High speed, compact size, and energy efficiency could be achieved in high quantity digital data computation at the cost of accuracy for portable digital signal processing applications such as video, audio, and image processing. This results in the approximate computation instead of an accurate computation. Therefore, obtaining high accuracy, high speed, energy and area efficiency are a primary challengeable task for designers. So far, many transistor-level Approximate Mirror Adders (AMA) were designed to perform large word size Error Tolerant Adder (ETA) circuits [3]. Approximate XOR/XNOR Adders (AXA) [8], CNFET-based Ternary Full Adders [7] and transmission gate based adders [1], were designed and employed to get the result faster, area and energy efficiency in a multiple-bit error tolerant adder design. The effective approximate full adder design is necessary to achieve high performance, such as high speed, minimum error, minimum error distance, low power consumption, and area efficiency. The optimized gate level is introduced to reduce the switching activity on the sum generation path and the number of logic gate count reduction on the carry propagation path of the proposed gate-level SAFA designs. The proposed gate-level SAFAs are employed in the 16-bit compact energy efficient error tolerant adder (CEETA) designs. Moreover in the High performance Error Tolerant Multiplier (HPETM) designs, the transformation of generated partial products into altered partial products is used [4] for approximation multiplication. Further in the approximation computation (stage 1 to stage 3), high performance arithmetic modules (SAFA2E, SAFA1E, AC) are proposed. The Simplified Approximate Full Adder 1-Error (SAFA1E) improves the pass rate and it assists in achieving better accuracy. The carry propagation delay and the gate count reduction are proposed in the SAFA2E and Approximate Compressor (AC) modules to achieve high speed, area and energy efficiency. The proposed multipliers offer high performance in terms of area, power, speed and accuracy than the existing approximate multipliers.

This paper is structured as follows; Section 2 describes the related approximate full adder theory, Section 3 exploits the related error tolerant adder and multiplier theory, Section 4 deals with the design methodology of the proposed SAFAs design. The design overview of the proposed Compact Energy efficient Error Tolerant Adders (CEETAs) and High Performance Error Tolerant Multipliers (HPETMs) are presented in Section 5. Section 6 presents the performance analysis of various adders and multipliers in terms of area, delay, accuracy, and power. Finally, the work is concluded in Section 7.

2. Related approximate full adder theory

Table 1 shows the logical behavior of the Conventional Full Adder (CFA) and the existing AFAs. Logic operations of the CFA are given in Eqs. (1) and (2).

$$Carry = (A \oplus B)\cdot C \ + A\cdot B \tag{1}$$

$$Sum = (\mathrm{A} \oplus \mathrm{B}) \oplus C \tag{2}$$

The gate level architecture of CFA has two Conventional Half Adder (CHA) structures and one OR-ed gate. The CFA structure has thirteen basic logic gates (AND, OR and NOT gates), six logic gate delays in the

sum (S) generation output and five logic gate delays in the carry (CY) generation output. The gate-level architecture of CHA, which has six basic logic gates, three logic gate delays on the sum generation path and one logic gate delay on the carry generation path. Approximate full adders exhibit a faster result with energy and area efficiency than the conventional full adder. The gate-level architecture of AFA-2 logic [2], which has six basic logic gates, three logic gate delays for carry generation output and four logic gate delays for sum generation output. The approximate sum output of AFA-2 is derived from the accurate carry output by using inverter gate logic. Similarly, in the CNTFET-based Inexact Full Adder (IFA), the accurate carry output is derived from the approximate sum output by using inverter logic [17]. Hence, both AFA-2 and IFA have two errors in the sum output and no error in the carry output, when all the input bits having uniform distribution as '111' or '000' [2,17] which is shown in Table 1. The carry generation path has two basic logic gates and one of the two XOR gates is replaced with an OR gate on the sum generation path of the AFA. This results in two errors out of eight cases. This provides more simplification in gate count as seven and two logic gate delays in the carry propagation output, while maintaining the difference between the original and the approximate value as one [4]. Logic complexity reduction at the gate-level AFAs is presented in [9-11]. In Modified Full Adder (MFA) [12], carry output has two errors out of eight cases and it has accurate sum output function. The sum generation path has ten basic logic gates and six logic gate delays. The limitation of the design offers more critical path sum delay and area overhead on the sum computation path. If any error exists in the carry output, the accurate value will be generated in the sum output which increases the error distance value as two. The error probability in carry increases with increase in the size of the input bits [14,15]. The MBAFA-I and MBAFA-II are based on the multiplexer logic and these adders have two errors out of eight cases and the error distance value as one. The gate-level architecture of the MBAFA-I has 6 basic logic gates and these gates are employed on the sum logic formulation. The gate-level architecture of MBAFA-II logic has 7 basic logic gates. The six basic logic gates are employed on the sum logic formulation and one basic logic gate is incorporated on the carry generation path. The value of the Error Distance (ED), Pass Rate (PR) and Error Rate (ER) is an important factor to calculate the accuracy in the approximate computation. ED is the arithmetic difference between error and exact outputs. MFA has more error distance to compare with other approximate adders. The pass rate is represented by the number of correct outputs over than the total number of outputs. The error rate is mentioned by a number of inexact outputs over than the total number of outputs [18].

3. Related error tolerant adder and multiplier theory

In an n-bit Error Tolerant Adder (ETA-I), the central point where m-bit is n/2 to the Most Significant Bit (MSB) information is named as accurate part and m-bit to the Least Significant Bit (LSB) information is named as inaccurate part. To avoid large error, CFA based RCA is implemented for normal addition rule in the accurate part. The special function rule is implemented in the inaccurate part to check every bit position from the central position to LSB. When both input values are '1' in the inaccurate part, all sum bits of that appropriate sum to the LSB position is set to '1'. If values are '0' or different, normal XOR function is performed and operation proceeds to the next LSB position [5,6]. CLA and RCA logic are used in the accurate part for improving the area and speed in the ETA-II design. Since there is no carry signal generated, the propagation path will not exist. Energy efficient Low power Area efficient Error Tolerant Adder (ELAETA-I or ELAETA-II) has error sensitive circuit in the most significant bit position of the inaccurate part, which computes the carry and the appropriate addition of carry to the least significant bit position of the accurate part. It increases the accuracy when normal OR operation is performed instead of XOR operation on the inaccurate part of area efficiency over the existing ETA-I or ETA-II, respectively [5,6]. When the input operands values are "255" *or* "127", the 16-bit ETA or ELAETA family exhibits 50% worst case error and the overall computational accuracy is lesser than 50% for low value inaccurate part operands ranging from '0' *to* "225". A carry-select adder has 40% to 90% faster speed than a ripple carry adder by performing additions in parallel and reducing the maximum carry path delay. The 16-bit Gate-level Significance Approximation Error Tolerant Carry Select Adder (SAET-CSLA) architecture was proposed [2], which improves the accuracy. The SAET-CSLA is designed using CFA based CSLA on the accurate part and AFA-2 based CSLA on the inaccurate part to achieve area efficiency at the gate-level logic. High Speed Error Tolerant Adder (HSETA) is implemented using CFA based CSLA on the accurate part and the MFA based RCA logic is incorporated in the inaccurate part to achieve high speed [12]. High Speed Static Segment Adder (HSSSA) is implemented to achieve 100% computational accuracy for the 8-bit lower part logic operands [16]. The MBAFA based HPETAs provide more than 79% computational accuracy for the inaccurate part logic operands value ranging from 0 to 255 and it exhibits more than 99% overall computational accuracy with high speed, area and energy efficiency [18]. The error tolerant multiplication produces inaccurate value. The multiplier that incorporates an error tolerant full adder and 4-2 compressor exhibits acceptable output. The design criteria for an approximate full adder and 4-2 compressor are usually multi fold. Gate count is the primary concern, which determines the system complexity of the data path unit. Hence finding gate count reduction approaches to execute these output functions with minimal error becomes a most important necessity. In this section, some of the prior works in designing approximate multipliers were discussed. An approximate multiplier based on approximate adder named Broken-Array Multiplier (BAM) is an unsigned multiplier similar to array multiplier structure. In BAM, the multiplication is designed based on broken Carry Save Adder (CSA) structure by omitting a few intermediate CSA cells [19]. The approximate signed Booth multiplier with BAM approximation method reduces the power consumption of upto 58.6% and area upto 41.8% for different word lengths compared to regular Booth multiplier. An approximate multiplier using2 $\times$ 2 inaccurate building blocks save power upto 45.4% compared to an accurate multiplier. An approximate signed 32-bit multiplier for speculation purposes in pipelined processors were 20% faster than a full-adder based on tree multiplier with 14% error probability [20]. An Accuracy-Configurable Multiplier Architecture (ACMA) suggested for error-resilient systems, increases its throughput. The ACMA made use of a technique called carry-in prediction that is based on pre-computation logic. Compared to exact multiplier, the proposed approximate multiplication reduces nearly 50% in latency due to shortest critical path [21]. An Approximate Wallace Tree multiplier (AWTM) [22] invokes the carry-in prediction to reduce the critical path. AWTM used in real-time benchmark image application achieves 40% and 30% reductions in power and area, respectively, without loss in image quality compared to an accurate Wallace tree multiplier (WTM). A dynamic range unbiased multiplier (DRUM) selects m-bit segment from leading one bit of input operands and sets the least significant bit (LSB) of truncated values to one. In this structure, the truncated values are multiplied and shifted to left to generate the final output. Approximate 4 $\times$ 4 WTM uses an inaccurate 4:2 counter. In addition, an error correction unit for correcting the outputs has been suggested. To construct larger multipliers, this 4 $\times$ 4 inaccurate WTM has been used in an array structure [23]. In Rounding-Based Approximate multiplier (RoBA), the operands are rounded to nearest exponent of two to remove the computational intensive part of multiplication. This improves the speed and energy consumption with trade off in accuracy [24]. Dynamic segment method (DSM) performs multiplication operation on m-bit segment from leading one bit of the input operands. In DSM, for an n-bit multiplier, two n-bit leading one detectors (LOD) and n/2-bits multiplier schemes are required. Since this approach reduces the hardware complexity than the exact multiplier with some simplified logic. The accuracy of DSM scheme is more. Since this approach is also consuming more area, delay and power but little reduction than the conventional

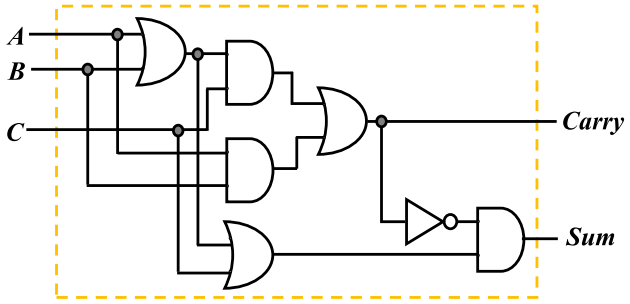

Fig. 1. Gate-level architecture of proposed SAFA1E.

scheme. An alternative approach to DSM is simplified using static segment method (SSM) with different segment sizes chosen by multiplexer based architecture in order to improve the performance. The energy efficient SSM technique for speech signal processing consumes 42% energy/op is reported [25]. Approximate 4-2 compressor has been designed by partitioning the partial products using recursion in multiplication and used in image processing application to evaluate the quality of an image [26]. Two types of approximate 4-2 compressor for multiplier have been applied for image multiplication [27]. Approximate 15-4 compressor based on 5-3 compressor of 16-bit multiplier has better performance and efficiency against existing 4-2 compressor multiplier [28].

4. Proposed simplified approximate full adders (SAFAs)

The main goal of the SAFA designs is gate level optimization on the sum generation path and carry propagation path to achieve area and energy efficiency for a large word size CEETA and HPETM designs. In full adders, XOR-ed gates occupy more area and it exhibits high delay on the sum generation path. So that in the SAFAs design, the XOR gates are replaced with basic logic gates (AND, OR, NOT) on the sum generation path. The proposed SAFA1E, SAFA2E and SAFA3E logics are made of 2-input AND-ed, 2-input OR-ed, and 1-input inverter gates. The proposed SAFA4E logic is made of 2-input OR-ed gate. Fig. 1 shows the general gate-level architecture of the SAFA1E which has 7 basic logic gates and these gates are employed on the sum and carry logic formulation. The simplified logic functions of the SAFA1E design as given in Eqs. (3-4).

$$Carry = A{\cdot}B + C{\cdot}(A + B) \tag{3}$$

$$Sum = \overline{Carry}\,{\cdot}(A + B + C) \tag{4}$$

This results in one error in the sum bit and no error in the carry bit out of eight cases. The accurate value in carry bit and inaccurate value in sum bit maintain the error distance as one when the input bits are uniformly presented as “111” is shown in Table 2.

$$Sum = \overline{A}{\cdot}(B + C) + B{\cdot}C \tag{5}$$

$$Carry = A \tag{6}$$

$$Sum = \overline{A}.\,(B + C) \tag{7}$$

$$Carry = A \tag{8}$$

$$Sum = B + C \tag{9}$$

$$Carry = A \tag{10}$$

Similarly, Figs. 2-4 show the general gate-level architecture of the proposed SAFA2E, SAFA3E, SAFA4E designs, which have 5, 3, 1 basic logic gates respectively and these gates are employed on the sum logic formulation. The most significant bit (A) of the inputs is directly assigned as a carry output which reduces a node capacitance and improves the speed of computation on the carry propagation path. The proposed SAFA2E, SAFA3E, SAFA4E designs have less carry output delay than the sum output delay. The proposed SAFA2E, SAFA3E and

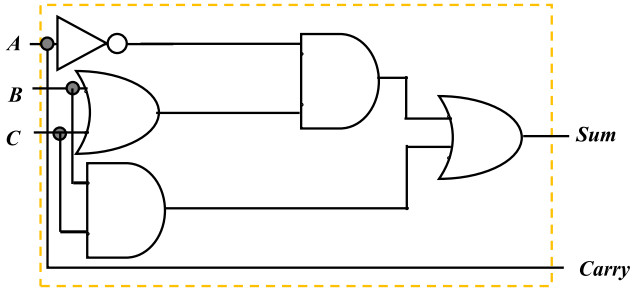

Fig. 2. Gate-level architecture of proposed SAFA2E.

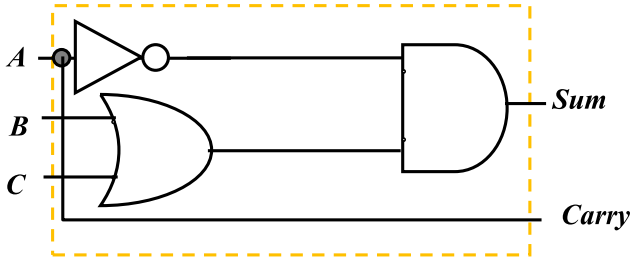

Fig. 3. Gate-level architecture of proposed SAFA3E.

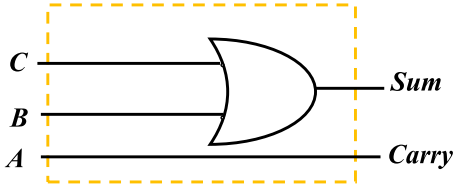

Fig. 4. Gate-level architecture of proposed SAFA4E.

Table 2
Truth table of proposed SAFAs.

INPUTS			Proposed SAFA1E			Proposed SAFA2E			Proposed SAFA3E			Proposed SAFA4E		
A	*B*	*C*	*CY*	*S*	*ED*	*CY*	*S*	*ED*	*CY*	*S*	*ED*	*CY*	*S*	*ED*
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
0	0	1	0	1	0	0	1	0	0	1	0	0	1	0
0	1	0	0	1	0	0	1	0	0	1	0	0	1	0
0	1	1	1	0	0	0	1	-1	0	1	-1	0	1	-1
1	0	0	0	1	0	1	0	+1	1	0	+1	1	0	+1
1	0	1	1	0	0	1	0	0	1	0	0	1	1	+1
1	1	0	1	0	0	1	0	0	1	0	0	1	1	+1
1	1	1	1	0	-1	1	1	0	1	0	-1	1	1	0

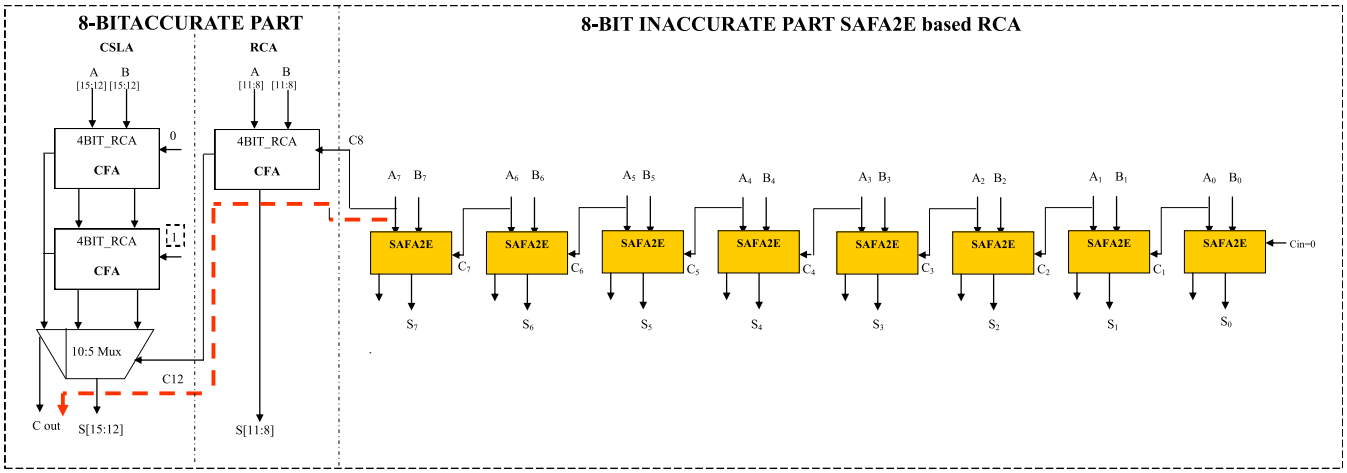

Fig. 5. Block diagram of proposed 16-bit CEETA.

SAFA4E are based on the logic formulations which are given in Eqs. (5-6), (7-8), (9-10) respectively and these adders have two, three, four errors respectively out of eight cases as shown in Table 2. A highlighted gray color denotes that error output. If any error occurs in the carry output, an inaccurate value will be generated instead of an accurate value in the sum output which decreases the error distance value as one and it is presented in Table 2. From the Table 2, the carry propagation of the proposed SAFA2E, SAFA3E, SAFA4E designs have one number of buffer-gate logic. Due to the fast generation of carry output feature and the gate count reduction in the sum generation feature, the proposed SAFA2E, SAFA3E, SAFA4E designs are more favorable than the existing AFA designs for area-energy efficient implementation of large word-size high performance error tolerant adders.

5. Proposed SAFAs based CEETAs and HPETMs

5.1. Proposed SAFAs based compact energy efficient error tolerant adders (CEETAs)

This section describes the design features of the proposed CEETAs. The dividing strategy is chosen to design proposed CEETA adder with area efficiency and minimum acceptable accuracy to be more than 99%. Having considered the above, the 16-bit adder is divided by 8-bit or 7-bit in to the accurate part and 8-bit or 9-bit in to the inaccurate part CEETA or CEETA1 respectively. If the accurate part has less than 8-bit, the minimum acceptable accuracy and the acceptance probability will be decreased. If the accurate part has more than 8-bit, the minimum acceptable accuracy and the acceptance probability will be slightly more than 99% and area will be inefficient. The normal addition rule is implemented from the LSB to MSB of the proposed CEETAS when compared to the special function rule in the inaccurate part of the ETA family. To preserve the correct output in the most significant bits and speed up the computation of a 16-bit CEETA, CFA based CSLA structure is employed in the accurate part and SAFA2E cell based parallel adder is proposed in the inaccurate part to achieve area and energy efficiency at the cost of accuracy as shown in Fig. 5. To achieve high speed in the proposed CEETA, carry propagation input of the each SAFA2E is obtained from the previous input operand A in the inaccurate part as given as $C_{i+1} = A_i$ where i varies from 0 to 7. Hence the inaccurate part design offers parallel addition instead of the ripple carry adder addition. The proposed SAFA2E cell based parallel adder implementation in the inaccurate part reduces the critical path delay (Red color). This leads to a high speed and it offers an accurate part computational delay. The input-carry (C_8) delay of the accurate part is assumed to be zero and the delay of final-carry (C_{out}) represents the adder critical path delay. The critical path delay is the sum of 4-bit CFA based RCA delay and a multiplexer delay of the accurate part.

Further, the goal of this research is to improve CEETA systematically. First, we generalize the CEETA design in the form of an 8-bit accurate part and 8-bit inaccurate part; then, we obtain an optimal architecture for the presented design focusing on computational accuracy, area, power and delay. The proposed method will be referred to as optimized CEETA1 is shown in Fig. 6. Conventional Half Adder (CHA) and CFA based structure is employed in the Most Significant Bit (MSB) accurate

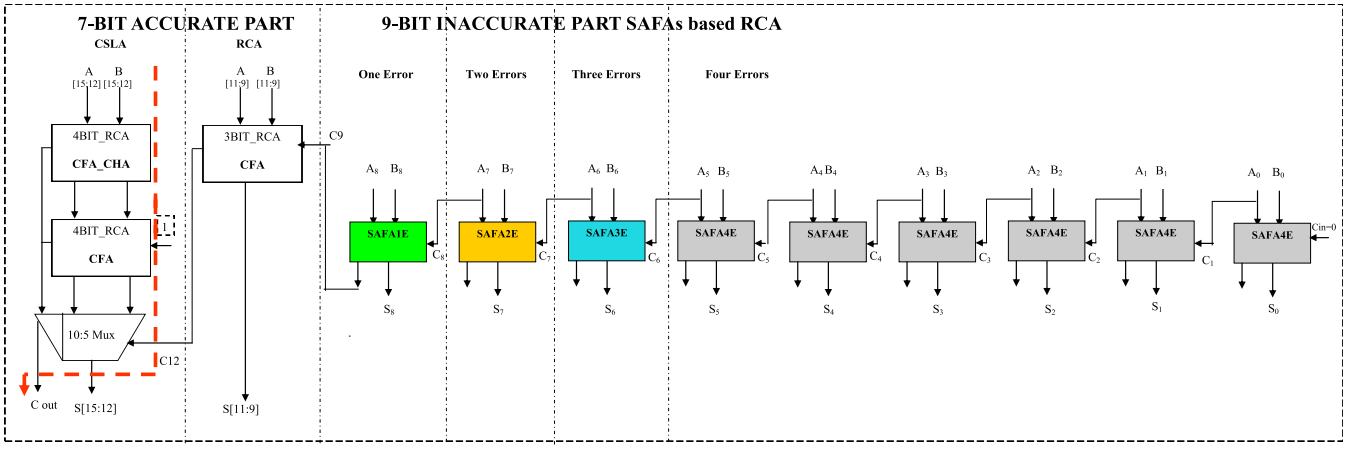

Fig. 6. Block diagram of proposed 16-bit CEETA1.

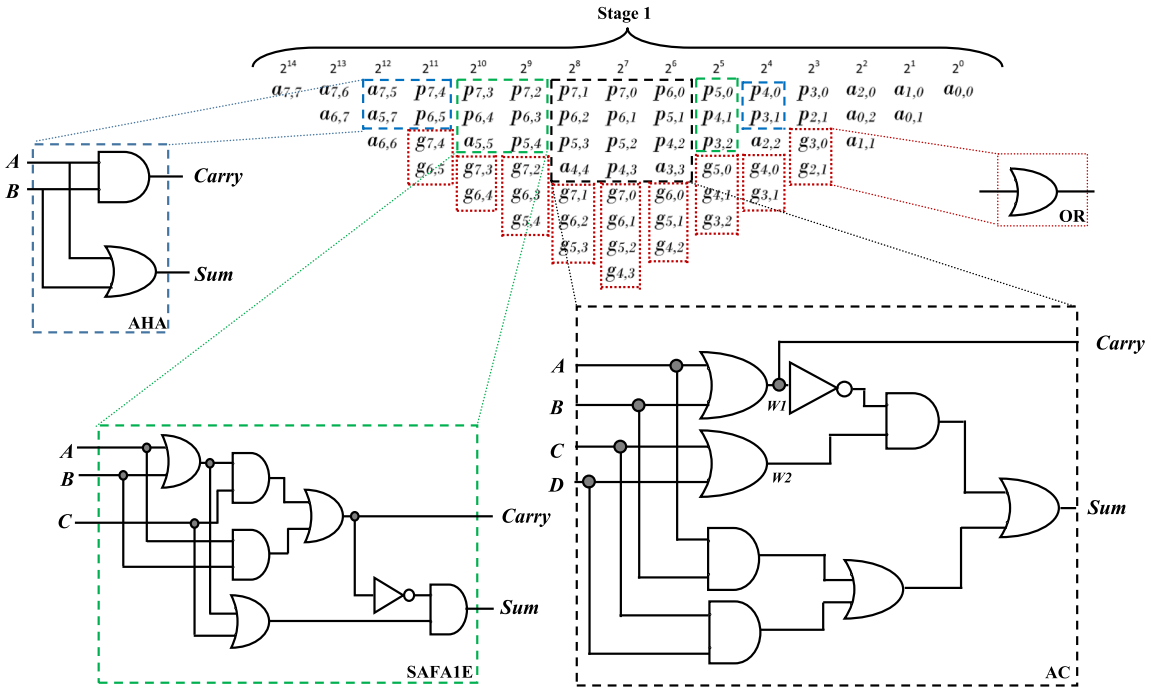

Fig. 7. Stage 1 of RTPP.

part (7-bit), the SAFA1E based structure is proposed in the mid-segment MSB part (1-bit), SAFA2E is used in the mid-segment middle part (1-bit), SAFA3E is used in the mid-segment lower part (1-bit), and SAFA4E structure is incorporated in the least significant part (6-bit). The optimized CEETA1 design outperforms all the existing approximate adders, considering the trade-off between design metrics and computational accuracy. The critical path delay reduction improves the speed of operation and the gate count reduction reduces the switching activity in the critical path for area and energy efficient implementation of the proposed 16-bit CEETAs. The main advantages of the proposed CEETAs is that the normal addition method is performed from LSB of the inaccurate part to MSB of the accurate part and it takes the accurate part computational delay only when compared to the existing transistor-level ETA-I, ETA-II, ELAETA-I and ELAETA-II architectures.

5.2. *Proposed AC and SAFAs based high performance error tolerant multipliers (HPETMs)*

The main goal of the proposed HPETM designs is to reduce the number of gate count reduction on the carry propagation path to achieve high speed of computation with area and energy efficiency. The implementation of 8-bit multiplier has generation of Transformed Partial Products (TPP) and three stages to produce final product from the TPP. The unsigned two 8-bits input operands u, v are considered. The partial product $a_{x,y} = u_x \cdot v_y$ is the result of AND operation between the bits of u_x and v_y. The column has more than three partial products (PP), the PP $a_{x,y}$ and $a_{y,x}$ are integrated to form propagate $p_{x,y}$ and generate $g_{x,y}$ signals of the TPP [4] as given in Eqs. (11) and (12).

$$p_{x,y} = a_{x,y} + a_{y,x} \tag{11}$$

$$g_{x,y} = a_{x,y} \cdot a_{y,x} \tag{12}$$

The TPPs and the remaining PPs are used to produce approximation computation in the stage 1 of Reduction in Transformed Partial Product (RTPP) matrix for 8×8 multiplier is shown Fig. 7. SAFA1E and AC are proposed with the Approximate Half Adder (AHA) [4] and OR gates for their accumulation. In adders and compressors, XOR gates occupy more area and it exhibits high delay. The main goal of the proposed AC design is to reduce the number of gate count reduction on the carry propagation path sum generation path to achieve high speed of computation with area and energy efficiency as shown in Fig. 7. The simplified logic functions of the proposed AC design as given in Eqs. (13-16). The corresponding behavior of the AC is given in Table 3. From Table 3, it shows five errors with minimum error distance as one out of sixteen cases.

Table 3
Truth table of Proposed 4-2 AC.

INPUTS				Proposed 4-2 AC		
A	*B*	*C*	*D*	*CY*	*S*	*ED*
0	0	0	0	0	0	0
0	0	0	1	0	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	-1
0	1	0	0	1	0	+1
0	1	0	1	1	0	0
0	1	1	0	1	0	0
0	1	1	1	1	1	0
1	0	0	0	1	0	+1
1	0	0	1	1	0	0
1	0	1	0	1	0	0
1	0	1	1	1	1	0
1	1	0	0	1	1	+1
1	1	0	1	1	1	0
1	1	1	0	1	1	0
1	1	1	1	1	1	-1

$$W1 = A + B \tag{13}$$

$$W2 = C + D \tag{14}$$

$$Sum = \overline{W1} \cdot W2 + A \cdot B + C \cdot D \tag{15}$$

$$Carry = W1 \tag{16}$$

In Fig. 7, three AHAs, three SAFA1E modules, three 4-2 ACs are needed to exhibit sum and carry outputs, S_i and C_i and the outputs of OR gates are grouped and marked as G_i corresponding to column i. The resultant elements are placed appropriately for the approximate

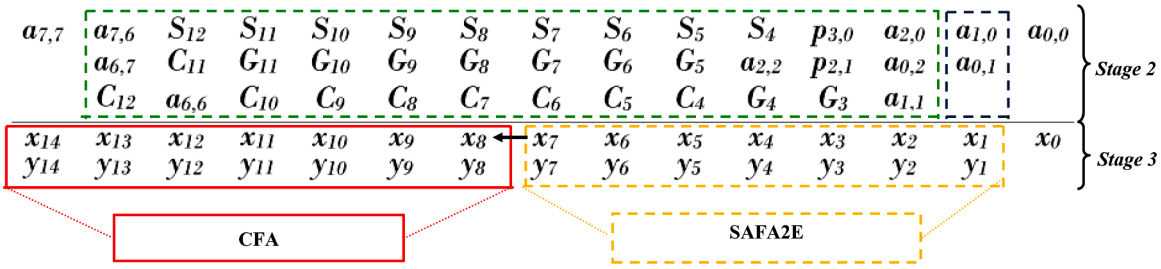

Fig. 8. Stage 2 and 3 of RTPP.

Table 4
logic level comparison of adders.

Adder Type	Area (gate count)	Sum (logical delay)	Carry (logical delay)
CFA	13	6	5
AFA-2 [2]	6	4	3
AFA [4]	7	4	2
MFA [12]	10	6	0
MBAFA-I [18]	6	4	0
MBAFA-II [18]	7	5	1
Proposed SAFA1E	7	5	3
Proposed SAFA2E	5	3	0
Proposed SAFA3E	3	2	0
Proposed SAFA4E	1	1	0

addition in the stage 2 of RTPP as shown in Fig. 8. The elements in the stage 2 are reduced using one AHA and twelve SAFA1E modules producing two operands x_i and y_i to be fed to Error Tolerant Ripple Carry Adder structure (ETRCA) in the stage 3 as shown in Fig. 8 for the approximate computation of the product output. The ETRCA has accurate and inaccurate part. To achieve high speed, the proposed SAFA2E modules are used in (n-1) least significant columns of RTPP. To achieve high speed in the ETRCA, carry propagation input of the each SAFA2E is obtained from the previous input operand x_{i-1} in the inaccurate part as given as $C_{i+1} = x_i$ where i varies from 2 *to* $n-1$. Hence the inaccurate part design offers parallel addition instead of the ripple carry adder addition. The proposed SAFA2E cell based parallel adder implementation in the inaccurate part reduces the critical path delay. This leads to a high speed and it offers an accurate part computational delay in the stage 3. Two variant of multipliers are proposed. In the HPETM1, approximation is applied in all columns of stage 1, 2 and ETRCA is used in the stage 3, whereas in HPETM2, the appropriate approximation modules are used in $n-1$ columns of RTPP. The critical path delay reduction improves the speed of operation and the gate count reduction reduces the switching activity in the critical path for area and energy efficient implementation of the proposed HPETMs.

6. Simulation results and discussion

6.1. Analysis of existing and proposed approximate full Adders

6.1.1. Logical gate count and logical gate delay analysis

The fundamental structural analysis in terms of gate count and logic delay for the existing AFAs and the proposed SAFAs adders as shown in Table 4. Gate-level logic comparison result shows that the proposed SAFA2E design has eight gate count savings with respect to the existing CFA. But in this proposed SAFAs and the existing MBAFA-I, and MFA, the overall carry propagation delay is considerably less (one buffer delay is included in the synthesized process) when compared to the existing AFAs. The proposed SAFA2E has 1, 2, 5, 1, 2 gate count savings when compared to the existing two error cases AFA-2, AFA, MFA, MBAFA-I, MBAFA-II respectively.

6.1.2. Design metrics analysis

All modules are coded in Very-High-Speed Integrated Circuits Hardware Description Language (VHDL), analysis the functional behavior in the Xilinx 14.2 ISE software and these designs are synthesized in the Cadence Encounter (R) RTL compiler using the Application Specific Integrated Circuits (ASIC) based TSMC 90-nm standard cell library of which 1.0, 0.9V, 25 °C are fixed as the standardized process corner, operating voltage, ambient temperature respectively for the area, delay, static power and dynamic power investigation. The net list obtained is then processed in the RTL compiler. After placement and route, the area, delay and power are reported by the Cadence Encounter (R) RTL compiler. Table 5 displays the design metrics of the proposed SAFA2E and the existing two error cases AFAs.

CFA consumes more power, high area utilization and larger delay than the SAFA2E and the existing AFAs. The proposed SAFA2E design has 47.1%, 10.0%, 35.7%, 18.2%, 10.0% and 30.77%, area savings than the CFA, AFA-2, AFA, MFA, MBAFA-I and MBAFA-II respectively. The proposed SAFA2E design involves significantly less number of basic logic gates on the sum generation path. Therefore the proposed designs consume less power. The proposed SAFA2E design offers a savings of 67.4%, 11.2%, 49.4%, 58.2%, 6.1%, 33.6% power than the CFA, AFA-2, AFA, MFA, MBAFA-I, MBAFA-II respectively. The proposed SAFA2E design has lesser delay on the sum generation path than the CFA, AFA, MFA, MBAFA-I respectively. Carry propagation plays a vital role in the critical path delay of the large word size adders.

Table 5
Design metrics comparison of two errors full adders.

Adder Type	Cell Area (μm^2)	Leakage Power (nW)	Dynamic Power (nW)	Total Power (nW)	Sum Delay (ps)	PDP (aJ)	ADP (μm^2.ps)
CFA	17	51.390	314.024	365.414	276	100.9	4692
AFA-2 [2]	10	20.907	113.247	134.154	150	20.1	1500
AFA [4]	14	36.921	198.688	235.609	227	53.5	3178
MFA [12]	11	41.108	244.259	285.367	183	52.2	2013
MBAFA-I [18]	10	18.945	107.910	126.855	152	19.3	1520
MBAFA-II [18]	13	27.457	151.951	179.408	144	25.8	1872
Proposed SAFA2E	09	18.132	101.021	119.153	150	17.9	1350

Table 6
Design features of 16-bit adders.

Adder Type	Design Features and Gate Count	
Con. CSLA	424 gate count	
HSSSA [16]	Static Segment CSLA 329 gate count	
	Accurate Part	Inaccurate Part
SAET-CSLA [2]	CFA based CSLA 248 gate count	AFA-2 [2] based CSLA 92 gate count
ETCSLA		AFA [4] based CSLA 104 gate count
HSETA [12]		MFA [12] based RCA 80 gate count
HPETA-I [18]	CFA based CSLA & RCA 176 gate count	MBAFA-I based RCA 48 gate count
HPETA-II [18]		MBAFA-II based RCA 56 gate count
Proposed CEETA		SAFA2E based RCA 40 gate count
Proposed CEETA1	CFA based CSLA & RCA 156 gate count	SAFAs based RCA 21 gate count

The existing AFA-2 and MBAFA-II designs have more delay on the carry formulation path than that of the proposed SAFA2E design. Carry propagation delay, power consumption, area-delay product and power-delay product of the proposed SAFA2E design are considerably lesser than the existing AFAs designs. Hence, SAFA2E based n-bit error tolerant adder leads to provide area and energy efficiency. The Proposed SAFA2E offers a savings of 71.2%, 10.0%, 57.5%, 32.9%, 11.2%, 27.9% ADP than the CFA, AFA-2, AFA, MFA, MBAFA-I, MBAFA-II respectively. The proposed design SAFA2E is more energy efficient than the conventional and other approximate full adders. The Proposed SAFA2E consumes 82.3%, 10.9%, 66.5%, 65.7%, 7.3%, 30.6% lesser PDP than the CFA, AFA-2, AFA, MFA, MBAFA-I, MBAFA-II respectively.

6.2. Analysis of existing and proposed 16-bit error tolerant adders

6.2.1. Logical gate count analysis

The existing 16-bit conventional CSLA has 424 basic logic gates. In 16-bit dual ripple carry adder it requires 7 numbers of 4-bit CFA based RCA and 3 numbers of 10:5 multiplexer. Each CFA has 13 logic gates and a 2:1 multiplexer has 4 logic gates. *The total number of gate count* = $(7*4*13)+\ (3*5*4) = 424$. Table 6 exploits the design structure of the proposed and the existing error tolerant adders. The proposed 16-bit CSLA based CEETA has 176 gates in the accurate part and 40 basic logic gates in the inaccurate part. Moreover, the proposed 16-bit CEETA1 has 156 gates in the accurate part and 21 basic logic gates in the inaccurate part. From Table 6, the proposed CEETA has 208, 124, 136, 112, 113, 8, 16 gate count savings and the proposed CEETA1 offers 247, 163, 175, 151, 152, 47, 55 gate count savings when compared to the conventional CSLA, SAET-CSLA, ETCSLA, HSETA, HSSSA, HPETA-I, HPETA-II respectively.

6.2.2. Design metrics analysis

The investigated design metrics of the 16-bit proposed and existing adders are shown in Table 7. From the Table 7, the proposed adder exhibits better performances than the existing error tolerant adder approaches. As expected, CEETAs have area and energy efficiency owing by the parallel implementation of the SAFAs based logic in the inaccurate part. Therefore, critical path delay of the CEETA and CEETA1 involve accurate part computation delay and these have 31.44%, 23.21%, 13.28%, 7.11%, 25.66%, 0.00%, 2.98% lesser critical path delay. The proposed CEETA offers a savings of 44.43%, 35.63%, 41.02%, 27.20%, 27.57%, 3.37%, 6.17% area and the proposed CEETA1 offers a savings of 51.63%, 43.87%, 48.57%, 36.52%, 36.84%, 15.72%, 18.18% area than the conventional CSLA, SAET-CSLA, ETCSLA, HSETA, HSSSA, HPETA-I, HPETA-II, respectively. Whereas CEETA consumes 36.59%, 33.67%, 33.26%, 29.42%, 18.87%, 0.71%, 1.41% lesser power and the CEETA1 consumes 44.06%, 41.48%, 41.12%, 37.73%, 28.43%, 12.40%, 13.03% lesser power than the conventional CSLA, SAET-CSLA, ETCSLA, HSETA, HSSSA, HPETA-I, HPETA-II, respectively.

The proposed CEETA and CEETA1 have lesser ADP and PDP than the conventional and other existing approximate adders. The proposed CEETA consumes 56.54%, 49.07%, 37.92%, 34.46%, 39.70%, 0.64%, 4.11% lesser PDP and the proposed CEETA1 consumes 61.66%, 55.07%, 45.28%, 42.17%, 46.80%, 12.34% and 15.40% lesser PDP than the conventional CSLA, SAET-CSLA, ETCSLA, HSETA, HSSSA, HPETA-I, HPETA-II, respectively. The proposed CEETA consumes 61.96%, 50.52%, 48.82%, 32.30%, 46.10%, 3.10%, 8.71% lesser ADP and CEETA1 consumes 66.80%, 56.82%, 55.33%, 40.91%, 52.95%, 15.41%, 20.32% lesser ADP than the conventional CSLA, SAET-CSLA, ETCSLA, HSETA, HSSSA, HPETA-I, HPETA-II, respectively. Finally from Table 7, the proposed CEETA1 has 12.80% less area utilization, 11.78% low power consumption, 11.78% less PDP, 12.72% less ADP than the proposed CEETA.

6.2.3. Quality metrics analysis

6.2.3.1. FPGA hardware based co-simulation for image processing application. To analyze the overall computational accuracy of existing and proposed designs, a number of input operand pairs (65536*3 (*RGB*)) are to be examined, so that these designs are incorporated to implement an image blending application to investigate the quality metrics as given in Eqs. (17) and (18).

$$PSNR = 20\log_{10}\left(V_{max} \Big/ \sqrt{MSE}\right) \tag{17}$$

V_{max} and MSE are the maximum dynamic range of the pixel values and mean square error, respectively for image processing applications.

$$MSE = \frac{1}{M\ X\ N}\sum_{i=1}^{M}\sum_{j=1}^{N}\left(X_{ij} - Y_{ij}\right)^2 \tag{18}$$

FPGA hardware co-simulation [13] of the proposed 16-bit CEETA1 user-defined module is integrated with MATLAB Simulink predefined modules by setting or detecting target board information such as the board name (test), system clock (Clock Frequency 20Mhz, Clock Pin E10), JTAG options (Position 1, IR lengths 6) and configuration of targetable device family, part, speed and package (Spartan6, xc6slx16,-2 and ftg256) in the compilation of Xilinx system generator for image

Table 7
Design metrics comparison of 16-bit adders.

Adder Type	Cell Area (μm^2)	Leakage Power (nW)	Dynamic Power (nW)	Total Power (nW)	Delay (ps)	PDP (fJ)	ADP (μm^2.ns)
Con.CSLA	521	1689.778	13356.885	15046.663	1428	21.49	744
SAET-CSLA	449	1795.995	12587.527	14383.522	1275	18.34	572
ETCSLA	490	1672.732	12622.280	14295.012	1129	15.06	553
HSETA	397	1761.783	11755.102	13516.885	1054	14.25	418
HSSSA	399	1696.380	10063.340	11759.720	1317	15.49	525
HPETA-I	299	1244.771	8363.539	9608.310	979	9.40	292
HPETA-II	308	1315.372	8362.239	9677.611	1006	9.74	310
Proposed CEETA	289	1211.223	8329.502	9540.725	979	9.34	283
Proposed CEETA1	252	1173.112	7243.303	8416.415	979	8.24	247

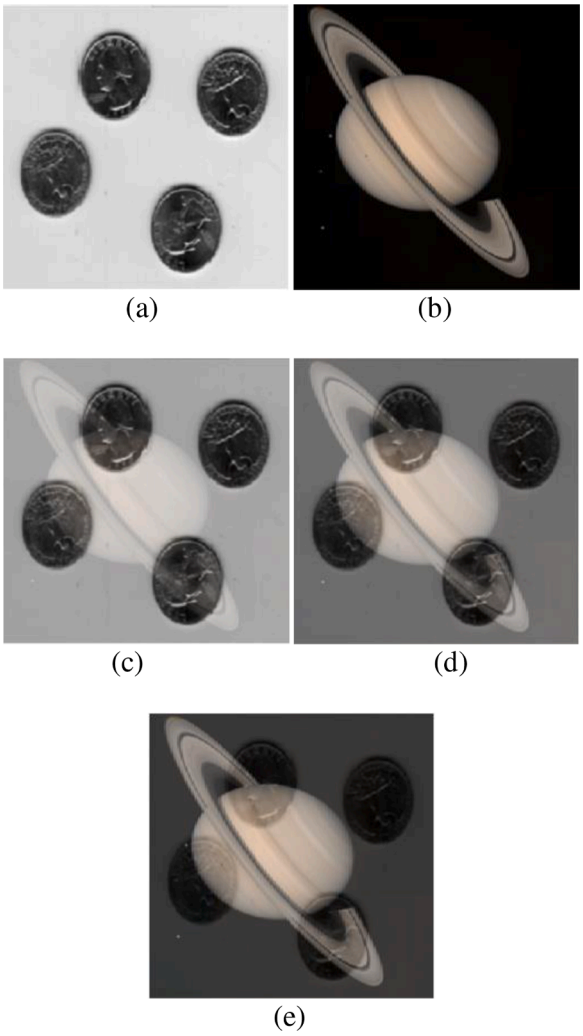

Fig. 9. Input images (a) f1, (b) f2, Output images g of CEETA1 (c) α = 0.25, (d) α = 0.50, (e) α = 0.75.

blending application to find out the PSNR value of the output image. The input and output images of the proposed 16-bit CEETA1 for various blending factors are represented in Fig. 9. The two test color images $f_1(x, y)$ and $f_2(x, y)$ are chosen from the database for image blending application as shown in Fig. 9 (*a*) *and* (*b*). The size of the each color images is 256*256. The image has three dimensional RGB pixels. The total number of pixels in each dimension is 65536. The depth of each pixel is 8-bit. The 8-bit of each pixel value of input images are mapped from (0 −255) *to* (0 −65535) 16-bit value for 16-bit data addition. Image blending operation is performed based on image addition principle as per the Eq. (19). The corresponding pixel value of one image $f_2(x, y)$ is multiplied by a blending factor (α) and another one image $f_1(x, y)$ is multiplied by $(1 - \alpha)$ and is added to avoid the saturation overflow.

$$g(x, y) = (1 - \alpha) \cdot f1(x, y) + \alpha \cdot f2(x, y) \tag{19}$$

where, α is the blending or a transparency factor for varying the contrast and brightness influence of foreground ($f_1(x, y)$) and background ($f_2(x, y)$) of the blended image $g(x, y)$. The different weights ($\alpha = 0.25,\ 0.5,\ 0.75$) are given for input images for varying the contrast and brightness influence of foreground (75%, 50%, 25%) and background (25%, 50%, 75%) of the blended images as shown in Fig. 9 (*c*), (*d*) *and* (*e*) respectively. Similar to pixel addition, the value of each pixel in the output image is a linear combination of the corresponding pixel values in the input images. The investigated quality metrics of the proposed and existing adders are shown in Table 8.

Table 8
PSNR Comparison of 16-bit Approximate Adders.

Adder Type	α = 0.25 PSNR (dB)	α = 0.50 PSNR (dB)	α = 0.75 PSNR (dB)
SAET-CSLA	56.68	56.37	56.79
ETCSLA	59.01	58.68	57.96
HSETA	53.57	53.79	54.05
HSSSA	52.55	54.16	53.53
IFA-CSLA	56.68	56.37	56.79
HPETA-I	59.59	59.81	60.07
HPETA-II	62.71	63.54	62.96
Proposed CEETA	59.59	59.81	60.07
Proposed CEETA1	57.75	57.36	57.89

Table 9
Quality and Error metrics Comparison of 16-bit adders.

Adder Type	Mean Relative Error Distance	Mean Error Distance	Normalized Mean Error Distance	Computational Accuracy (%)
SAET-CSLA	$1.6703e^{-007}$	0.0064	$0.0488e^{-006}$	99.8437
ETCSLA	$1.4647e^{-007}$	0.0069	$0.0526e^{-006}$	99.8629
HSETA	$2.0163e^{-007}$	0.0095	$0.0725e^{-006}$	99.8113
HSSSA	$1.0935e^{-007}$	0.0051	$3.8910e^{-008}$	99.8976
IFA-CSLA	$1.6703e^{-007}$	0.0064	$0.0488e^{-006}$	99.8437
HPETA-I	$1.0081e^{-007}$	0.0048	$0.0366e^{-006}$	99.9056
HPETA-II	$7.4770e^{-008}$	0.0034	$0.0259e^{-006}$	99.9300
Proposed CEETA	$1.0081e^{-007}$	0.0048	$0.0366e^{-006}$	99.9056
Proposed CEETA1	$1.5901\ e^{-007}$	0.0060	$0.04578e^{-006}$	99.8513

From the Table 8, it is found that the proposed CEETA provides slightly less PSNR when compared to the HPETA-II. But the proposed CEETA has lesser delay, power, area, PDP and ADP than the existing HPETA-II adder. The proposed CEETA1 exhibits slightly less PSNR when compared to the ETCSLA, HPETA-I, HPETA-II. But the proposed CEETA and CEETA1 have lesser power, area, PDP and ADP than the existing adders.

6.2.3.2. Test Vector based simulation. For any approximate adder, the Error Distance (*ED*) is the difference between exact sum output (*E*) and the predicted approximate sum output (*P*), where $ED = |E - P|$. The Relative Error Distance (*RED*) is the ratio of Error Distance (*ED*) to the accurate output (*E*), $RED = \frac{ED}{E} = \frac{|E-P|}{E}$. The Error Rate (*ER*) is the ratio of incorrect outputs with respect to the total number of outputs. For any N-bit approximate adder, the Mean RED (*MRED*) is, $MRED = \frac{\sum_{i=1}^{2^{2N}-1} RED}{2^{2N}}$. Mean Error Distance (MED) is the average of error distance, where $MED = \frac{\sum ED}{2^{2N}}$. Normalized MED is, $NMED = \frac{MED}{S_{max}} = \frac{\sum_{i=0}^{2^{2N}-1} ED}{\frac{2^{2N}}{S_{max}}}$, Where

Table 10
Quality and Error metrics Comparison of 8-bit inaccurate part of adders.

8-bit inaccurate part only	Mean Relative Error Distance	Mean Error Distance	Normalized Mean Error Distance	Computational Accuracy (%)
SAET-CSLA	0.4534	57.7408	0.1132	54.6573
ETCSLA	0.2116	63.7500	0.1250	78.8392
HSETA	0.4088	89.5753	0.1756	59.1214
HSSSA	0.0000	0.0000	0.0000	100.0000
IFA-CSLA	0.4534	57.7408	0.1132	54.6573
HPETA-I	0.2044	44.7867	0.0878	79.5607
HPETA-II	0.1189	31.7500	0.0623	88.1073
Proposed CEETA	0.2044	44.7867	0.0878	79.5607

Table 11
Design Metrics Comparison of 16-bit Multipliers.

Label	Multiplier Type	Area (um^2)	Power (nW)	Delay (ps)
HPETM1	Proposed HPETM1	3900	222400	1197
HPETM2	Proposed HPETM2	4421	311413	1215
AM1	Approximate Multiplier1 [4]	4214	234843	1595
AM2	Approximate Multiplier2 [4]	5258	476384	1707
SSM	Static Segment Multiplier [25]	3715	495207	1720
ACM1	Approximate 4-2 Compressor [26]	4596	578583	1755
ACM2	Approximate 4-2 Compressor [27]	4717	600251	1752
ACM3	Approximate 15-4 Compressor [28]	6382	613814	1810
CDM	Conventional Dadda Multiplier [28]	7550	744645	1858

Table 12
Quality and Error Metrics Comparison of 16-bit Multipliers.

Label	Mean Relative Error Distance	Mean Error Distance	Normalized Mean Error Distance	Computational Accuracy (%)
HPETM1	0.0720	944.6692	0.0145	92.7952
HPETM2	0.0224	104.4189	0.0016	97.7643
AM1	0.0787	1.6771 e+03	0.0258	92.1264
AM2	0.0163	97.2334	0.0015	98.3680
SSM	0.1479	1.7421e+03	0.0268	85.2123
ACM1	0.3067	5.3799e+03	0.0828	69.3293
ACM2	0.2848	4.6499e+03	0.0715	71.5248
ACM3	0.2074	3.3298e+03	0.0508	79.2621
CDM	0.0000	0.0000	0.0000	100.00

S_{max} is the maximum value of sum that can be obtained from an N-bit accurate adder, $S_{max} = (2^N - 1) + (2^N - 1)$. To evaluate the functional behavior of the existing and the proposed 16-bit adders, 4,58,850 test vectors are randomly applied. The investigated quality and error metrics are listed in Table 9 for comparison. The HPETA-II exhibits high computational accuracy and less mean error distance when compared to other adders. Similarly, the quality and error metrics on the inaccurate part operands (0 *to* 255) of the existing and the proposed adders are investigated by using $(2^{2N} - 1)$ *where* $(N = 8)$ test vectors. From the investigated results, the existing HSSSA provides 100% computational accuracy for the 8-bit lower part logic operands is shown in Table 10. Finally from Tables 7 and 9, the proposed CEETA1 has 12.80% less area utilization, 11.78% low power consumption, 11.78% less PDP, 12.72% less ADP, 0.054% less computational accuracy than the proposed CEETA.

6.3. Analysis of existing and proposed 16-bit Error Tolerant Multipliers

The 16-bit multipliers are designed and implemented in VHDL and synthesized in Cadence Encounter (R) RTL compiler for design metrics reports.

6.3.1. Design metrics analysis

Table 11 compares all designs in terms of area, power and delay. From Table 11, the proposed HPETM1 exhibits 24.95%, 29.87%, 30.41%, 31.79%, 31.68%, 33.87%, and 35.58% lesser delay than the AM1, AM2, SSM, ACM1, ACM2, ACM3 and CDM respectively. The proposed design HPETM1 is more energy efficient than the conventional and other approximate multipliers. The Proposed HPETM1 consumes 5.30%, 53.31%, 55.09%, 61.56%, 62.95%, 63.77% and 70.13% lesser power than the AM1, AM2, SSM, ACM1, ACM2, ACM3 and CDMrespectively.

6.3.2. Quality metrics analysis

To evaluate the functional behavior of the existing and proposed 16-bit multipliers, 18,35,400 test vectors are randomly applied. The investigated quality and error metrics are listed in Table 12 for comparison. From the Table 11 and XII, it is observed that the AM2 exhibits 0.62% high computational accuracy and it offers 40.49%, 18.93% and 52.97% more delay, more area utilization and more power consumption respectively than the proposed HPETM2.

7. Conclusion

This paper presents the design and evaluation of gate-level 16-bit adder and multipliers based on the proposed approximation modules. In comparison with other approximate full adder modules, SAFA2E shows lesser PDP and ADP. The efficacy of the proposed CEETA and HPETM are compared to the existing adders and multipliers. The proposed structure CEETA consumes 36.59%, 33.67%, 33.26%, 29.42%, 18.87%, 0.71%, 1.41% lesser power and CEETA1 consumes 44.06%, 41.48%, 41.12%, 37.73%, 28.43%, 12.40%, 13.03% lesser power than the conventional CSLA, SAET-CSLA, ETCSLA, HSETA, HSSSA, HPETA-I, HPETA-II, respectively.. Moreover, the proposed HPETM1 exhibits lesser delay and it consumes 5.30%, 53.31%, 55.09%, 61.56%, 62.95%, 63.77% and 70.13% lesser power than the AM1, AM2, SSM, ACM1, ACM2, ACM3 and CDM respectively. Therefore, the proposed CEETA and HPETM modules are more suitable for the high speed, area and energy efficient error tolerant applications.

Declaration of Competing Interest

None.

References

[1] Z. Yang, J. Han, F. Lombardi, Transmission gate-based approximate adders for inexact computing, in: Proc. NANOARCH, IEEE/ACM International Symposium on. IEEE, 2015, pp. 145–150.
[2] R. Jothin, C. Vasanthanayaki, High performance significance approximation error tolerance adder for image processing applications, J. Electronic Test. 32 (3) (2016) 377–383.
[3] V. Gupta, D. Mohapatra, A. Raghunathan, K. Roy, Low-power digital signal processing using approximate adders, in: IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems 32, 2013, pp. 124–137.
[4] S. Venkatachalam, S.-.B. Ko, Design of power and area efficient approximate multipliers, IEEE Trans. Very Large Scale Integr. Syst. 25 (5) (May 2017) 1782–1786.
[5] R. Sakthivel, H.M. Kittur, Energy efficient low area error tolerant adder with higher accuracy, Circ. Syst. Signal Process. 33 (8) (2014) 2625–2641.
[6] N. Zhu, W. Goh, W. Zhang, K. Yeo, Z. Kong, Design of low-power high-speed truncation-error-tolerant adder and its application in digital signal processing, IEEE Trans. VLSI Syst. 18 (8) (2010) 1225–1229.
[7] F. Sharifi, A. Panahi, M.H. Moaiyeri, H. Sharifi, K. Navi, High performance CNFET-based ternary full adders, IETE J. Res. 64 (1) (2018) 108–115, https://doi.org/10.1080/03772063.2017.1338973.
[8] Z. Yang, A. Jain, J. Liang, J. Han, F. Lombardi, Approximate XOR/XNOR-based adders for inexact computing, in: Nanotechnology (IEEE-NANO), 2013 13th IEEE Conference on, IEEE, 2013, pp. 690–693.
[9] D. Kelly, B. Phillips, Arithmetic data value speculation, in: Proc. Asia-Pacific Comput. Syst. Architect. Conf, 2005, pp. 353–366.
[10] S.-L. Lu, Speeding up processing with approximation circuits, Computer 37 (3) (2004) 67–73. Mar.
[11] Y.V. Ivanov, C.J. Bleakley, Real-time h.264 video encoding in software with fast mode decision and dynamic complexity control, ACM Trans. Multimed. Comput. Commun. Appl 6 (2010), 5:1–5:21Feb.
[12] S. Geetha, P. Amritvalli, High speed error tolerant adder for multimedia applications, J. Electron. Test. 33 (5) (2017) 675–688.
[13] R. Borade, A. Dimber, D. Gharpure, S. Ananthakrishnan, Design and development of FPGA-based spectrum analyzer, IETE J. Educ. 59 (1) (2018) 5–17, https://doi.org/10.1080/09747338.2018.1450648.
[14] H.R. Mahdiani, A. Ahmadi, S.M. Fakhraie, C. Lucas, Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications, IEEE Trans. Circ. Syst. Part I 57 (4) (2010) 850–862.
[15] M.K. Ayub, O. Hasan, M. Shafique, Statistical error analysis for low power approximate adders, in: *Design Automation Conf. (DAC)*, 2017 54th ACM/EDAC/IEEE, IEEE, 2017, pp. 1–6.
[16] R. Jothin, C. Vasanthanayaki, High speed energy efficient static segment adder for approximate computing, J. Electron. Test. 33 (1) (2017) 125–132.

[17] Y.S. Mehrabani, R.F. Mirzaee, Z. Zareei, S.M. Daryabari, A novel high-speed, low-power CNTFET-based inexact full adder cell for image processing application of motion detector, J. Circ. Syst. Comput. 26 (05) (2017).
[18] R. Jothin, C. Vasanthanayaki, High performance error tolerant adders for image processing applications, IETE J. Res. (2018), https://doi.org/10.1080/03772063.2018.1535920.
[19] H.R. Mahdiani, A. Ahmadi, S.M. Fakhraie, C. Lucas, Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications, IEEE Trans. Circ. Syst. I 57 (4) (2010) 850–862.
[20] C. Liu, J. Han, F. Lombardi, A low-power, high-performance approximate multiplier with configurable partial error recoveryProceedings of the Conference on Design, Automation & Test in Europe, Eur. Des. Autom. Assoc. (2014) 95.
[21] K. Bhardwaj, P.S. Mane, ACMA: accuracy-configurable multiplier architecture for error-resilient system-on-chip, in: Reconfigurable and Communication-Centric Systems-on-Chip (ReCoSoC), 2013 8th International Workshop on, IEEE, 2013 pages 1–6.
[22] K. Bhardwaj, P.S. Mane, J. Henkel, Power-and area-efficient approximate wallace tree multiplier for error-resilient systems, in: Quality Electronic Design (ISQED), 2014 15th International Symposium on, IEEE, 2014, pp. 263–269.
[23] S. Hashemi, R. Iris Bahar, S. Reda, Drum: a dynamic range unbiased multiplier for approximate applications., in: *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*, ICCAD '15, IEEE Press, Piscataway, NJ, USA, 2015, pp. 418–425.
[24] R. Zendegani, M. Kamal, A.A.-.K. MiladBahadori, M. Pedram, Roba multiplier: a rounding-based approximate multiplier for high-speed yet energy-efficient digital signal processing, IEEE Trans. Very Large Scale Integr. Syst. 25 (2) (2017) 393–401.
[25] S. Narayanamoorthy, H. AsghariMoghaddam, Z. Liu, T. Park, N.S. Kim, Energy-efficient approximate multiplication for digital signal processing and classification applications, IEEE Trans. Very Large Scale Integr. Syst. 23 (6) (2015) 1180–1184.
[26] N. Maheshwari, Z. Yang, J. Han, F. Lombardi, A design approach for compressor based approximate multipliers, in: VLSI Design (VLSID), 2015 28th International Conference on, IEEE, 2015, pp. 209–214..
[27] A. Momeni, J. Han, P. Montuschi, F. Lombardi, Design and analysis of approximate compressors for multiplication, IEEE Trans. Comput. 64 (4) (2015) 984–994.
[28] R. Marimuthu, Y. Elsie Rezinold, P.S. Mallick, Design and analysis of multiplier using approximate 15-4 compressor, IEEE Access 5 (2017) 1027–1036.
[29] K.K. Parhi, VLSI Digital Signal Processing Systems: Design and Implementation, John Wiley & Sons, 2007.
[30] S.V. Vaseghi, Multimedia Signal Processing: Theory and Applications in Speech, Music and Communications, John Wiley & Sons, 2007, pp. 1–674.
[31] G.V. Varatkar, N.R. Shanbhag, Energy-efficient motion estimation using error-tolerance, in: ISLPED'06 Proceedings of the 2006 International Symposium on Low Power Electronics and Design, 2006, pp. 113–118.

R. Jothin received his Diploma in Electronics and Communication Engineering from Sankar Institute of Polytechnic, Tirunelveli, India and received his B.E. and M.E. (VLSI Design) degrees in Electronics and Communication Engineering from Government College of Technology, Coimbatore, India. He received his Ph.D. degree in the Information and Communication Engineering from Anna University, Chennai, India. He has 20 years of industry experience and 7 years of academic experience.

M. Peer Mohamed, Research Scholar in the Department of ICE at Anna University, Chennai. He was graduated in Electronics and Communication Engineering at Anna University in 2010. He received his M.E. degree in VLSI Design at Government College of Technology, Coimbatore in the year 2013. He worked as an Assistant Professor in the Department of ECE at SCAD Institute of Technology, Coimbatore. He has great interest in low power VLSI design for Image processing applications. He is working as a software engineer at Robert Bosch Engineering and business solutions private limited, Coimbatore.

C. Vasanthanayaki received her B.E. degree in Electronics and Communication Engineering and M.E. degree in Computer Science, from Government College of Engineering, Tirunelveli, India. She received her Ph.D. degree in the Information and Communication Engineering. Since 1988, she has been with the Government College of Technology and Engineering, Tamilnadu, India.