SERBIAN JOURNAL OF ELECTRICAL ENGINEERING
Vol. 22, No. 1, February 2025, 75-92
UDC: 004.932:004.315.2-047.82 **DOI: https://doi.org/10.2298/SJEE2501075B**
Original scientific paper

Energy Efficient Design and Implementation of Approximate Adder for Image Processing Applications

Jatothu Brahmaiah Naik[1], Kanagala Sateesh Kumar[2], Kondragunta Rama Krishnaiah[3], Seelam Koteswararao[4]

Abstract: Approximate computing is a new technique that promises to speed up computations while preserving a level of precision suitable for error-tolerant systems such as neural networks and graphics. At the edge, a lot of computationally complex methods are now in use. As such, designing quick and low-energy circuits is crucial. This work presents a novel approximate full adder approach that lowers power consumption and delay at the expense of some output mistakes. To achieve these objectives, the proposed full adder architecture makes use of fundamental gate logic reduction techniques. Evaluations based on the Intel FPGA synthesis tool indicate that the suggested adder surpasses state-of-the-art techniques in terms of power, speed, and propagation delay. The design parameters – area, power dissipation, and latent characteristics of proposed adder are verified by simulation using EDA tools. The results demonstrate that our proposed approximate adder runs faster and requires fewer logic components than earlier equivalent systems. The synthesis reports testify to the fact that compared to other adders currently in use, the suggested adder used less logic elements. Furthermore, suggested approximation adders were used to execute image additions. Using image addition, the image quantitative statistics are used to application-level accuracy metrics analysis. Quantitative results confirm the superior functioning of the full adder cell approximation over comparable models.

Keywords: FPGA, Image Processing, Adder Design, Image Addition, Approximate Computing.

[1]Department of Electronics and Communication Engineering, Narasaraopeta Engineering College, Andhra Pradesh, India, brahmaiahnaik@gmail.com, https://orcid.org/0000-0002-9333-9235
[2]Department of Electronics and Communication Engineering, Sreenidhi Institute of Science and Technology, Telangana, India, sateeshkumarkanagala@gmail.com, https://orcid.org/0000-0001-6233-7790
[3]Department of Computer Science and Engineering, R.K.College of Engineering, Andhra Pradesh, India, kondraguntark@gmail.com, https://orcid.org/0000-0002-9069-766X
[4]Department of Electronics and Communication Engineering, Lingayas Institute of Management and Technology, Andhra Pradesh, India, drskoteswararao.lingayas@limat.edu.in, https://orcid.org/0000-0002-3337-0361

1 Introduction

The modern computer world is moving away from efficiency-oriented design methods and toward environmentally friendly ones in order to bridge the gap between the scaling restrictions with transistor technology and the challenges centered around large data management. Batteries power the majority of portable mobile devices, requiring less hardware area and energy per task [1]. Moreover, Jadav projected that 1030 geophytes worth of information will be generated by three trillion networked devices by 2027, more than the amount of data that can be processed by current outdated systems [2]. Furthermore, modern computing systems are designed to produce exact outcomes at the price of extraordinarily high electricity consumption. However, when data is analyzed using conventional machines, the process of extracting the acquired information from memory and exploring newly created data from tiny devices that come from different kinds of equipment, like sensors and interfaces, is costly in terms of electricity and data gathering. To handle these new types of data, low power and inexpensive electronics are employed; high accuracy results are not expected, but there is a trade-off in lower energy consumption. However, high-throughput applications of these eco-friendly circuits lead to wildly uneven performance and incorrect outputs. Consequently, the phrase most frequently employed in error-silent applications is approximation computation.

Therefore, approximation computing uses fewer hardware circuit parts, uses less power, and operates with the least amount of delay while producing outputs that are accurate enough [3]. For applications such as media, the procedure, search engine mining, smartphone apps that handle real-time sensor data input, etc., precise outcomes are not necessary. It is shown that these inherently error-resistant applications are more appropriate for methods of approximation computation, where an approximation of the result is sufficient. Such applications are useful since human senses naturally have difficulties perceiving declines in the quality of auditory and visual information, duplicate inputs, and imprecise results [4]. The latter approach is applicable in scenarios when intrinsic error resilience is required, as approximation processing is a viable energy-efficient remedy [5].

The proliferation of multimedia applications on modern mobile devices necessitates processing units with minimal complexity and high energy efficiency. Standard design strategies trade-off area, power, and latency; increasing one causes the other to decrease. Therefore, these design solutions cannot simultaneously improve all the parameters. Numerous applications, such as multimedia and large-scale data processing, exhibit strong error resistance and may not require perfectly accurate results [6]. Reduced computational accuracy can result in significant reductions in size and power consumption while maintaining acceptable quality. Approximation computing has so emerged as a

novel approach to creating these applications, as its effectiveness has increased dramatically. Accuracy is viewed in these situations as a new trade-off characteristic to achieve improved design metrics. In many signal processing systems, the arithmetic unit regulates design efficiency and is a significant area and power consumer. An essential arithmetic component in many digital signal processing devices is the adder. However, it also forms the basis for other mathematical operations such as divisions, multipliers, increments, and decrements. The construction of a power and area-efficient adder is essential to the development of energy-efficient signal processing systems [7]. Several design methods have been proposed to obtain highly efficient adder designs. These solutions fall into three groups based on computational accuracy: exact, approximated, and precision-configurable design. Accurate design techniques make advantage of resource sharing and the removal of superfluous circuitry, whereas approximation design strategies utilize approximate calculation to locate and determine the overall number of irrelevant pieces [8].

The simplest design for an accurate adder is the ripple carry adder (RCA), despite having a high carry propagation latency. To reduce latency, the carry look-ahead (CLA) adder, for example, determines carry signals in advance, but it has a high area cost. Furthermore, less latency is provided with higher space and power consumption by the carry skip adder (CSKA) and carry select adder (CSLA) [9]. On the other hand, approximate adders work by building approximation sum logic for just a few of least significant bits (LSBs) or by utilizing a simplified complete adder. The overall error is minimized by the construction of the approximation sum logic. Furthermore, a generic approximation carry skip adder is presented in [10] that forecasts the carry-in for the current segment using previous v-segments. Including additional segments in the carry estimate process improves accuracy, but at a large space and latency cost. Though they need a large amount of redesign work to develop an approximate design with an alternative accuracy-performance compromise, the approximate adders provide better design metrics with acceptable error levels.

Different deployment circumstances demand different levels of accuracy; for example, a surveillance system needs to capture crisper images as soon as an object is detected [11]. In order to achieve optimal power savings and acceptable performance, the adder's accuracy for these applications ought to be reprogrammable in real time. Through the integration of the incorrect number into the ensuing pipelined stages, an accuracy customizable adder generates accurate results by estimating the total using sub-adders [12]. It thus provides a balance between processing latency and precision. While typical CLAs carry statements in precise mode, a programmable approximate CLA adder analyzes predetermined w terms (window) of carry expressions while it operates in approximation mode. Although multiplexers are employed to switch between exact and approximation modes, the RAP-CLA operates effectively but uses a lot

of power and space [13]. As a final resort, [14] describes a scaled non-zeroing bit-truncation (NzBt) adder that can switch between accurate and approximation modes via a control signal. In approximation mode, the NzBt sets a specific LSB of its input operands to complimentary values, producing return sum bits of constant logic '1'.

The main contributions are as follows:

1. For image processing applications, we suggested an approximate adder based on a straightforward AND OR logic circuit.
2. A comparison of modern and traditional approximate adders is made based on the logic of error rate complexity.
3. To compare the suggested approximate adder with other adders currently in use in terms of circuit complexity and logic cell utilization, we constructed the adder on an FPGA.
4. We demonstrated the addition of two greyscale images using both suggested and current approximation adders, and we examined the accuracy of the simulation results.

The paper is organized as follows in the remaining sections: The literature on development of low-power, configurable-accuracy adders is reviewed in Section 2. The proposed energy-efficient approximate adder is described in Section 3 along with a detailed analysis of the logical problems with current adders. In Section 4, the simulation and hardware results are compared between the suggested adder and the current adders. Finally, Section 5 presents the conclusion of this work.

2 Related Work

For the last ten years, the demand for high-performing, reasonably priced digital devices has driven research on approximation processing techniques. Approximate adders, an essential component of digital circuits, have garnered a lot of attention because of their capacity to strike a compromise between precision and resource consumption. This literature analysis aims to give an overview of the most recent approximate adder concepts, their applications, and the main challenges faced by professionals in this field. Approximate computing is a concept in development that trades precision for speed and energy efficiency. By reducing the need for exact computation outputs, approximate computing techniques capitalize on the inherent error resistance in a variety of applications, such as image processing, neural networks, and signal processing. Approximation algorithms are essential for computations where small errors are acceptable and large resource savings are feasible. Approximate computing is a novel idea that trades accuracy for higher performance and more energy efficiency. By reducing the need for exact computation outputs, approximate computing techniques

benefit from the innate error resistance in a variety of applications, such as image processing, machine learning, and signal processing. Approximation adders are essential for mathematical operations because they allow for significant energy savings and little error tolerance. Even while portable encryption existed long before the Internet of Things, experts have recently become interested in it because of its rapid growth.

In the work [15], a precision customizable approximate adder was disclosed. A complete adder and a half adder with maskable carry are used in the proposed ripple carry adder (RCA) design. The precision is set via an error-correcting circuit that is activated when needed. The LOA approximate adder is introduced by the authors in [16]. It computes the carry output inserted to the precise section using the 2-input AND gate for the most significant approximation bit and the output Sum for the approximation portion using the 2-input OR gate. If we utilize this adder architecture only for approximating a full adder, it will yield outputs Carry and Sum. The authors of [17] offer a method for identifying a hybrid approximation adder that is affordable to utilize in video and image processing systems. Because the objective of this study is to do multiplications without the usage of traditional processors, the primary techniques are shift-and-add, utilizing parallel prefixed adders. In [18], a practical method for ascertaining the probability errors of the block-based approximation adder is shown, along with a discussion of the statistical error evaluation of approximate adders. A configurable estimated carry look-ahead adder is proposed in [19]. This adder operates in two modes: accurate and approximation, and it has been validated in 15 nm FinFET invention. Using approximate full adders, the authors of [20] presented an approximation of the discrete cosine transform (DCT) for image compression. It replaces the floating-point multipliers with shifts and the addition of integers. The authors of [21] provide a method for obtaining parameterized error models for various approximation adders. Based on this paradigm, signal analysis applications are precise and efficient. In [22], there is a model for creating low-power approximators and a method for computing error variables in linear time. In [23], a nearly full adder based on static CMOS chips has been proposed. Two approximated compressors for the 5:2 and 4:2 ratios were reported in [24] and can be used with approximation multipliers. They are implemented with 7 nm FinFET technology. An approximation of a multiplication algorithm has been proposed in [25] based on the Karatsuba multiplication technique. It has achieved better precision and latency results than the approximate Wallace Tree multiplier.

The correctness of approximation adders has been assessed mathematically in [26] in order to get around the problems. A probabilistic approach to mistake modeling is presented by the authors in [27]. With this paradigm, speculative adders that predict the carrying capacity of sub-adder blocks can be implemented. For embedded medical products, Manikandan et al. [28] provided a basic

encryption method. For use in wireless data transfer, this encryption technique was developed. The authors claimed to have secured the patient's personal information by encrypting the data using a simple, randomly generated key method. Abd El-Latif et al. [29] developed the simple encryption method known as controlled-NOT images using novel logic. They used the NOT image produced by the dynamical map as the basis for their encryption. Using the key matrix construction technique during the implantation procedure is advised to improve the encryption process. To decrypt the encrypted image, you need the logistic map and key matrix. Hardware-dependent encryption methods have emerged recently for greater security and compatibility. Ravichandran et al. [30] developed a low-power medical image encrypting technology on the FPGA using the penta-layer methodology. Five different layers are used to encrypt the healthcare image. Every layer uses a separate technique for mixing and tumbling to produce an image that is properly encoded. High security is attained by developing hardware-dependent cryptography at the fourth layer. Azzam et al. [31] created the equipment-dependent compact steganography technique by taking into account the constraints of hardware resources for embedded applications. The authors examined at the cryptographic systems' device-dependent implementation, which shows resilience against a range of attacks.

Approximation adders described thus far can be changed based on an application. The degree of approximation can be changed as necessary, though, because another family of approximate adders has an adaptive estimator. Furthermore, outputs with varying degrees of accuracy can be generated based on the specific requirements.

3 The Proposed Approximate Full Adder

This section discusses the existing approximate adder, the suggested approximation adder, and the fundamental error-free complete adder's construction and functionality. Furthermore, a thorough discussion of the suggested adders' hardware implementation is included.

3.1 Conventional or error-free adder

A basic digital circuit known as an error-free adder, sometimes known as a full adder, is used in hardware architecture to perform binary addition. C_{in} is the carry input from the last addition phase, and A and B are the two binary inputs that need to be added. These three binary inputs are accepted by the 1-bit full adder. The two primary parts of the accurate adder's output are the carry output (C_{out}) and the sum output (S), which are passed to the subsequent addition stage. The sum output (S) represents the binary sum of the two input bits plus the carry input. If carrying out the current stage requires adding the next higher-order stage, it is indicated by the carry output (C_{out}). A complete adder can be constructed

using a variety of logic gates, including AND, XOR and OR gates. In the circuit diagram of a complete adder, these gates are frequently connected in a certain way to generate the required logic. When it's time to add an N-bit binary number, a cascade of complete adders are coupled together. By joining the carry output of one adder to the carry input of the next adder, multiple-digit numbers can be added. Error-free adders are essential parts of digital logic and arithmetic processes. In digital circuits like CPUs, it forms the foundation for increasingly complex computations.

3.2 Approximate adder

An approximate adder is a kind of digital design that performs addition operations at the expense of accuracy in order to maximize certain design parameters, like speed, energy consumption, or area. Approximate adders are not exact like traditional exact adders; instead, they aim to purposefully introduce controlled flaws into the adding process. These errors are often small enough to be tolerated in applications where giving up some precision is necessary to achieve other goals. Approximate adders are widely used in applications where modest calculation errors can be accepted, such as audio and image processing, machine learning, error-tolerant computations, other different embedded systems. Their energy efficiency and speed advantages make them ideal for scenarios where exact accuracy is not required. This section explains the existing approximate adder's logical operation. By employing transistor reduction approaches, the research works that are currently available offered a variety of approximate adders. The approximate adders now in use are referred to in this study as EAAs. For our performance comparison and study with the suggested adder, we have chosen five distinct approximation adders that currently exist. **Table 1** contains a list of the logical expressions for the current approximate adders.

Table 1

Boolean logical expression of 1-bit full adders.

ADDERS TYPE	SUM OUTPUT	CARRY OUTPUT
Conventional Full Adder	$A \oplus B \oplus C$	$AB + BC + AC$
EAFA1 [52]	$A(B + C) + BC$	$AB + BC + AC$
EAFA2 [53]	$C'_{out}C$	$AB + BC + AC$
AFA1	$(A'B + B'A)C'$	$(A + B)C$
AFA2	$A' + BC$	$(A + BC) + BC$
AFA3	$AB + C'$	A

Table 2 lists the truth tables for the conventional adder and EAFAs along with their accuracy and error values for performance comparison. Furthermore,

for performance analysis, the error rate (ER) and error distance (ED) are computed for both traditional and EAAs. ED typically measures the difference in output between an approximation and error-free adder

$$ED(p,q) = |p-q| = \left| \sum_k p[k]2^k - \sum_l q[l]2^l \right|. \quad (1)$$

The binary bits of the exact and approximate adder are denoted by $p[k]$ and $q[l]$, respectively. The percentage error between exact and approximate adders is calculated to get the error rate, or ER

$$ER = \left(\frac{ED}{2^i} \right) \cdot 100. \quad (2)$$

Table 2
Truth tables of conventional and existing approximate adders.

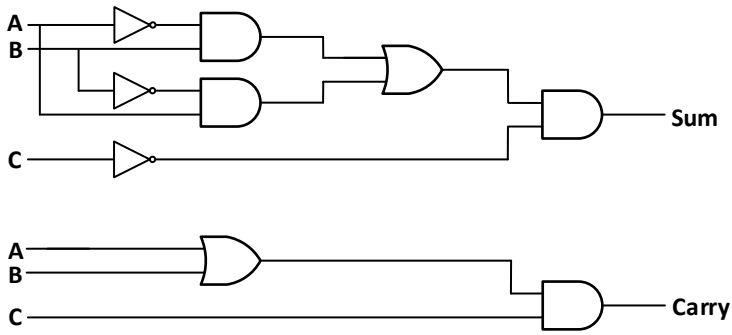
INPUTS			CONVENTIONAL ADDER		EAFA1		EAFA2	
A	B	C	Sum	Carry	Sum	Carry	Sum	Carry
0	0	0	0	0	1 X	0 ✓	0 ✓	0 ✓
0	0	1	1	0	1 ✓	0 ✓	1 ✓	0 ✓
0	1	0	1	0	1 ✓	0 ✓	0 X	0 ✓
0	1	1	0	1	0 ✓	1 ✓	0 ✓	1 ✓
1	0	0	1	0	1 ✓	0 ✓	0 X	0 ✓
1	0	1	0	1	0 ✓	1 ✓	0 ✓	1 ✓
1	1	0	0	1	0 ✓	1 ✓	0 ✓	1 ✓
1	1	1	1	1	0 X	1 ✓	0 X	1 ✓

3.3 Proposed approximate adder

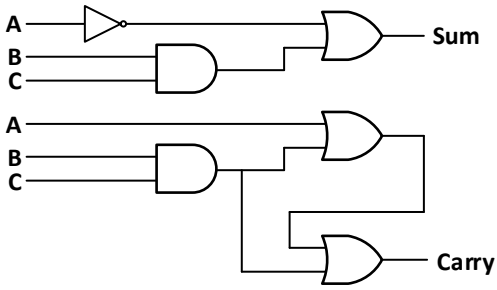
This section discusses the suggested three unique 1-bit approximate adders that make use of basic AND & OR gate logic circuits. This basic approximate adder employs a very reliable approximation mechanism. More sophisticated approximate adders employ a range of calculations and logic gates to consciously choose which bits to approximate or remove. Hereinafter, the developed approximation adders are denoted as AFA1, AFA2, and AFA3.

The fundamental logical formula given in **Table 1** has been used to mathematically derive each suggested adder. These formulas are used to compare the equation complexity of the proposed adder with that of existing adders. The suggested approximate adder has fewer gates than current circuits, according to the logical expression. The fundamental AND, OR logic gates are used to develop

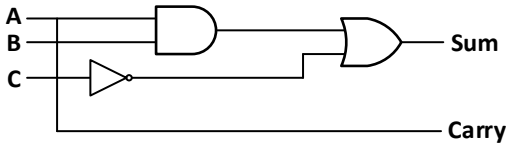
the logical diagram of these adders. Fig. 1 depicts the logical circuits of these adders.



(a)



(b)



(c)

Fig. 1 – *The proposed approximate full adder:* (a) *Design 1 of the proposed approximate full adder;* (b) *Design 2 of the proposed approximate full adder;* (c) *Design 3 of the proposed approximate full adder.*

The number of logical gates or cells as well as the amount of faults in the approximation adders is what we want to decrease. Our proposed adder's carry-out has been made simpler, and the equation complexity of the sum section has been decreased. To aid in comprehension, the logical diagram presents the

number of gates required and their level of complexity. **Table 3** lists the truth tables for the suggested adders.

Table 3
Truth table of proposed approximate adder.

INPUTS			APPROXIMATE ADDER-2		APPROXIMATE ADDER-2		APPROXIMATE ADDER-3	
A	B	C	Sum	Carry	Sum	Carry	Sum	Carry
0	0	0	0 ✓	0 ✓	1 X	0 ✓	0 ✓	0 ✓
0	0	1	0 X	0 ✓	1 ✓	0 ✓	0 X	0 ✓
0	1	0	1 ✓	0 ✓	1 ✓	0 ✓	1 ✓	0 ✓
0	1	1	0 ✓	1 ✓	1 X	1 ✓	0 ✓	0 X
1	0	0	1 ✓	0 ✓	0 X	0 X	1 ✓	1 X
1	0	1	0 ✓	1 ✓	0 ✓	1 ✓	0 ✓	1 ✓
1	1	0	0 ✓	0 X	0 ✓	1 ✓	1 X	1 ✓
1	1	1	1 X	1 ✓	1 ✓	1 ✓	1 ✓	1 ✓

It is required to assess the error rate of basic adder circuits for image processing applications. Since the circuits of these basis adders are entirely responsible for the pixel accuracy of adders. The error and percentage rates for the proposed, existing and conventional approximate adders are listed in **Table 4**. Because the suggested adders are employed for LSB adders in the RAC circuits for image processing addition, some tolerable errors in the carry are introduced.

Table 4
Error rate analysis of proposed adder circuits with existing adders designs.

TYPES OF ADDER	ERROR RATE		ERROR RATE IN PERCENTAGE	
	Sum Output	Carry Output	Sum Output	Carry Output
Conventional	0	0	0	0
EAFA1	2	0	25	0
EAFA2	3	0	37.5	0
AFA1	2	1	25	12.5
AFA2	2	1	25	12.5
AFA3	2	2	25	25

4 Results and Discussion

The specified approximate adder circuit was developed and evaluated through simulation using the electronic design automation tool. The effectiveness

of the developed approximation adder is contrasted with that of the standard adder and the existing approximate adder. Findings from a comprehensive simulation study of approximation adders constructed using Intel FPGA boards. Compared to the current existing approximate adders, the proposed approximation adder performs better in terms of delay and power.

4.1 Hardware implementation of proposed approximate adder

This section discusses the synthesis results and hardware implementation of proposed as well as existing approximate adders. It is necessary to develop an 8-bit ripple carry adder (RCA) circuit for image processing applications in order to carry out fundamental image operations. Using a traditional complete adder as the 4-bit MSB adder and suggested approximate adders as the 4-bit LSB adder, we have built and implemented an 8-bit RCA circuits to add two images. Maintaining the error rate for image processing applications is the primary motivation behind the implementation of both approximation and traditional adders. Using an Intel Cyclone IV FPGA board, the existing approximate & conventional adders are developed in order to demonstrate the energy efficiency of the proposed approximation adders. Fig.2 illustrates the block diagram of the suggested RCA adder for an image processing application. To examine the synthesis report of the suggested approximate adders, both traditional and current approximate adders are used. **Table 5** lists the FPGA synthesis reports. The suggested approximation adders use less power than the others, according to the FPGA resource performance.

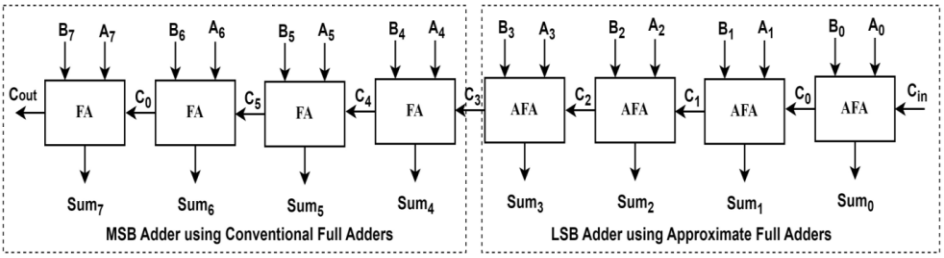
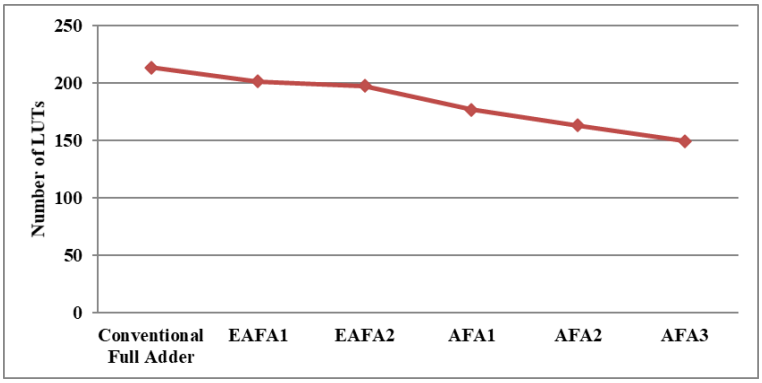
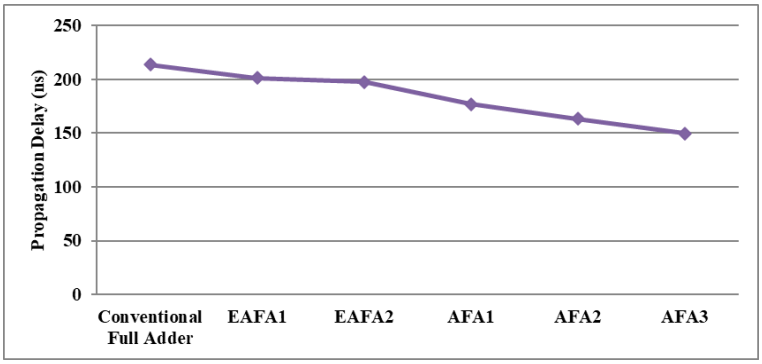


Fig. 2 – *The proposed RCA adder using proposed approximate adders.*

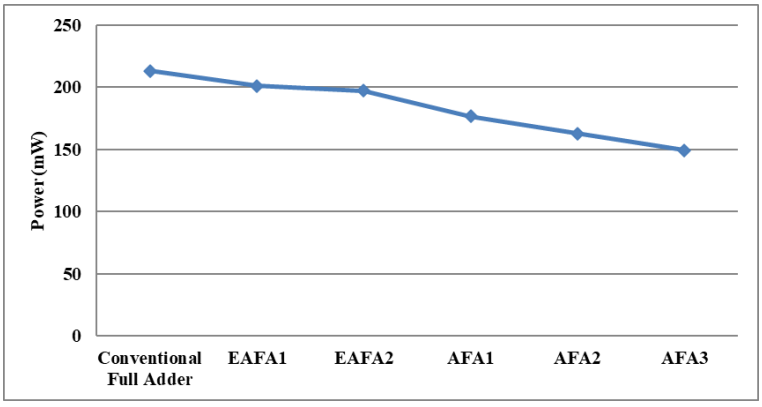
Compared to existing adders, the suggested approximate adder requires fewer LUTs. Fig. 3a shows the results of the LUTs comparison between the suggested and existing adders. Figs. 3b and 3c, respectively, showed the additional hardware performances, such as propagation delays and power consumption. The suggested adders perform better in terms of power, LUTs, and propagation delay with an improved performance of 17-30%, 28-45% and 18-33% respectively according to the synthesis reports. Using the image addition operation, the suggested adder's performance is examined in the next section.



(a)



(b)



(c)

Fig. 3 – *Analysis of resource usage proposed AFA with existing adders:*
(a) *Resource usage analysis based on the number of LUTs;*
(b) *Delay analysis of proposed adders with existing adders;*
(c) *Power analysis of proposed adders with existing adders.*

Table 5
FPGA implementation results of the conventional and approximate adders.

8-BIT RCA DESIGN	NUMBER OF LUTs	PROPAGATION DELAY(ns)	POWER [mW]
Conventional Full Adder	64	22.35	213.32
EAFA1	58	20.42	201.23
EAFA2	54	19.48	197.31
AFA1	46	17.91	176.78
AFA2	38	15.25	162.98
AFA3	35	14.74	149.56

4.2 Application of proposed approximate adders for image addition

In this section of the paper, the suggested AFAs are put into implementation. Many image processing applications use image addition, including motion detection (via reduction) and enhancing/masking particular areas of an image. This study evaluates the suggested AFAs for improving images. In this instance, two test images are combined to create a new image. Test Image 1 and Test Image 2, which are displayed in Figs. 4a and 4b, respectively, are selected for addition. The two test images are M×N-sized JPEG files, where M = N = 256. These test images were selected because they demonstrate the opposite characteristics that are required for image processing. The total number of pixels in each test image is 256×256 = 65536.

The value of each pixel is an 8-bit unsigned integer with a maximum value of '255' and a starting value of '0'. A pixel-by-pixel addition is used to merge two test images. An 8-bit binary RCA is used to add the pixels of the pertinent test images because each pixel has an 8-bit size. The structural layout of RCA for inserting the test images is shown in Fig. 2. It is composed of two adders: the most significant (MSB) adder and the least significant (LSB) adder. Both the MSB and LSB additions are performed using AFAs. The SCILAB software is used to convert the test images into pixel values before they are added. Using RCA's Verilog code, each image pixel by pixel is added in the next phase. The generated pixels are then converted back into images using the SCILAB program. The further results of test images using existing, recommended AFAs and conventional are shown in Figs. 4c to 4h.

The performance of the adders for image addition was compared using the following metrics:

(a) The quality of resulting image is measured by the Peak Signal-to-Noise Ratio (PSNR), which compares it to a ground-truth result using traditional full adders.
(b) The average inaccuracy brought about by the approximation is assessed using the Mean Squared Error (MSE).

(c) The perceived resemblance of output to the ground-truth image is measured by the Structural Similarity Index Measure (SSIM).

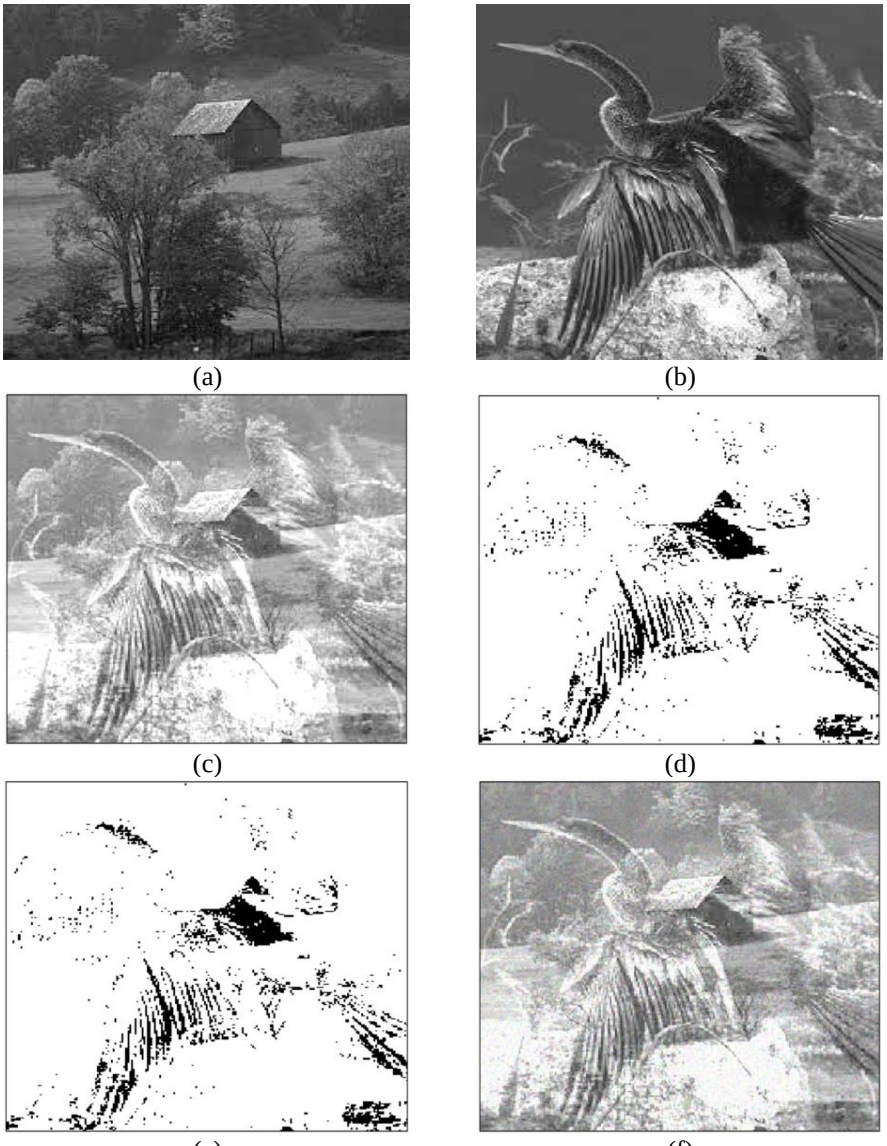


Fig. 4 – *Comparative analysis of Image addition of proposed AFA with existing adders:* (a) *The test image 1 for addition;* (b) *The test image 2 for addition;* (c) *Addition result of two images using conventional RCA adder;* (d) *Addition of two images using EAFA1 RCA adder;* (e) *Addition of two images using EAFA2 RCA adder;* (f) *Addition of two images using AFA1 RCA adder.*

Performance evaluation of proposed designs with existing adders in terms of above metrics is presented in **Table 6**. The MSE gradually rises as a result of approximation. Even though AFA3 has the highest error, error-resilient applications can still tolerate it. With increasing approximation level (e.g., AFA1 to AFA3), the PSNR value falls, indicating considerably lower output image quality. However, values remain above 35 dB, which is appropriate for most image processing applications. Even with AFA3, SSIM values stay above 0.85, suggesting that the structural integrity of image is maintained, despite a modest drop with increasing approximation.

Table 6

Performance evaluation of conventional and approximate adders for image addition.

8-BIT RCA DESIGN	MSE	PSNR(dB)	SSIM
Conventional Full Adder	2.01	45.8	0.99
EAFA1	3.21	43.5	0.97
EAFA2	4.76	41.2	0.94
AFA1	6.12	39.8	0.92
AFA2	7.89	38.6	0.90
AFA3	9.35	37.4	0.89

5 Conclusion

In order to reduce the cascade effect, we designed and evaluated 8-bit full adders that are error-tolerant, high-speed, and low-energy and area-consuming. More dependability and the shortest error distance are feasible with the suggested approximation adders, which also operate quicker and consume less power than comparable inexact full adders. In order to assess the proposed approximation adders' effectiveness, they were compared against other multi-bit adders based on conventional RCA architectures, such as the approximate and conventional adders. One-bit approximation full adder architectures without the cascading effect provide RCA speed, area, and power advantages for image processing applications. In this study, we introduced a unique approximate adder by combining error-reduced LSB and traditional MSB adders. In comparison to the present approximate adder, the suggested approximate adders accomplish an error rate reduction in the carry output, increasing the accuracy of image addition. The proposed gate reduction technique boosts the overall calculation accuracy by reducing the error rate. We carefully examined our design to determine the optimal ratio between hardware costs and accuracy, then modified the adder design parameter. Compared to the current approximation adders and traditional adders, the proposed design requires less logic parts when it is implemented in the Intel Cyclone IV FPGA board. Our suggested method outperforms the current approximate adder-based RCA design in terms of speed and power consumption.

An implementation of the proposed adder in a digital image processing application shows that it seldom deviates from the output image quality that is closest to the accuracy of the adder. We also examined our adder's performance in picture addition, and the findings showed that it performed better than other approximate adders. Therefore, the proposed adder is well suited for applications such as computer vision, image processing and neural computing that demand error tolerance and energy efficiency.

6 References

[1] A. Tirupathireddy, M. Sarada, A. Srinivasulu: Energy-Efficient Approximate Adders for DSP Applications, Analog Integrated Circuits and Signal Processing, Vol. 107, No. 3, June 2021, pp. 649 – 657.

[2] H. N. Fakhouri, S. Alawadi, F. M. Awaysheh, I. B. Hani, M. Alkhalaileh, F. Hamad: A Comprehensive Study on the Role of Machine Learning in 5G Security: Challenges, Technologies, and Solutions, Electronics, Vol. 12, No. 22, , November 2023, p. 4604.

[3] H. Yu, G. Yuan, D. Kong, L. Lei, Y. He: An Optimized Method for Nonlinear Function Approximation Based on Multiplierless Piecewise Linear Approximation, Applied Sciences, Vol. 12, No. 20, October 2022, p. 10616.

[4] G. Armeniakos, G. Zervakis, D. Soudris, J. Henkel: Hardware Approximate Techniques for Deep Neural Network Accelerators: A Survey, ACM Computing Surveys, Vol. 55, No. 4, April 2023, p. 83.

[5] S. E. Fatemieh, M. R. Reshadinezhad, N. Taheri Nejad: Approximate In-Memory Computing Using Memristive IMPLY Logic and its Application to Image Processing, Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS), Austin, USA, May 2022, pp. 3115 – 3119.

[6] E. Ozen, A.Orailoglu: Boosting Bit-Error Resilience of DNN Accelerators Through Median Feature Selection, IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, Vol. 39, No. 11, November 2020, pp. 3250 – 3262.

[7] M. Hassan, F. Awwad, M. Atef, O. Hasan: Approximate Computing-Based Processing of MEA Signals on FPGA, Electronics, Vol. 12, No. 4, February 2023, p. 848.

[8] A. Cherezov, A. Vasiliev, H. Ferroukhi: Acceleration of Nuclear Reactor Simulation and Uncertainty Quantification Using Low-Precision Arithmetic, Applied Sciences, Vol. 13, No. 2, January 2023, p. 896.

[9] V. Vijay, M. Sreevani, E. Mani Rekha, K. Moses, C. S. Pittala, K. A. Sadulla Shaik, C. Koteshwaramma, R. Jashwanth Sai, R. R. Vallabhuni: A Review on N-Bit Ripple-Carry Adder, Carry-Select Adder and Carry-Skip Adder, Journal of VLSI Circuits and Systems, Vol. 4, No. 1, June 2022, pp. 27 – 32.

[10] S. Patel, B. Garg, S. K. Rai: A Power and Area Efficient Approximate Carry Skip Adder for Error Resilient Applications, Turkish Journal of Electrical Engineering and Computer Sciences, Vol. 28, No. 1, January 2020, pp. 443 – 457.

[11] T. Nomani, M. Mohsin, Z. Pervaiz, M. Shafique: xUAVs: Towards Efficient Approximate Computing for UAVs – Low Power Approximate Adders with Single LUT Delay for FPGA-Based Aerial Imaging Optimization, IEEE Access, Vol. 8, June 2020, pp. 102982 – 102996.

[12] B. Sakthivel, A. Padma: Area and Delay Efficient GDI Based Accuracy Configurable Adder Design, Microprocessors and Microsystems, Vol. 73, March 2020, p. 102958.

[13] S. K. Patel, B. Garg , S. K. Rai: An Efficient Accuracy Reconfigurable CLA Adder Designs Using Complementary Logic, Journal of Electronic Testing, Vol. 36, No. 1, February 2020, pp. 135 – 142.

[14] M. B.Veena, N. S. Khanum, D. H. Soundaryya, P. M. Sarathi: Energy Scalable Brent Kung Adder with Non-Zeroing Bit Truncation, Proceedings of the IEEE 3rd Global Conference for Advancement in Technology (GCAT), Bangalore, India, October 2022, pp. 1 – 6.

[15] K.- L. Tsai, Y.- J. Chang, C.- H. Wang, C.- T. Chiang: Accuracy-Configurable Radix-4 Adder with a Dynamic Output Modification Scheme, IEEE Transactions on Circuits and Systems I: Regular Papers, Vol. 68, No. 8, August 2021, pp. 3328 – 3336.

[16] A. Kanani, J. Mehta, N. Goel: ACA-CSU: A Carry Selection Based Accuracy Configurable Approximate Adder Design, Proceedings of the IEEE Computer Society Annual Symposium on VLSI (ISVLSI), Limassol, Cyprus, July 2020, pp. 434 – 439.

[17] J. Lee, H. Seo, Y. Kim, Y. Kim: Approximate Adder Design with Simplified Lower-Part Approximation, IEICE Electronics Express, Vol. 17, No. 15, July 2020, p. 20200218.

[18] L. B. Soares, M. M. Azevedo da Rosa, C. M. Diniz, E. A. César da Costa, S. Bampi: Design Methodology to Explore Hybrid Approximate Adders for Energy-Efficient Image and Video Processing Accelerators, IEEE Transactions on Circuits and Systems I: Regular Papers, Vol. 66, No. 6, June 2019, pp. 2137 – 2150.

[19] S. Reda, M. Shafique: Approximate Circuits, Methodologies and CAD, Springer, Cham, 2019.

[20] A. J. Sanchez-Clemente, L. Entrena, R. Hrbacek, L. Sekanina: Error Mitigation Using Approximate Logic Circuits: A Comparison of Probabilistic and Evolutionary Approaches, IEEE Transactions on Reliability, Vol. 65, No. 4, December 2016, pp. 1871 – 1883.

[21] V. Camus, J. Schlachter, C. Enz: Energy-Efficient Inexact Speculative Adder with High Performance and Accuracy Control, Proceedings of the IEEE International Symposium on Circuits and Systems (ISCAS), Lisbon, Portugal, May 2015, pp. 45 – 48.

[22] Y. Wu, Y. Li, X. Ge, Y. Gao, W. Qian: An Efficient Method for Calculating the Error Statistics of Block-Based Approximate Adders, IEEE Transactions on Computers, Vol. 68, No. 1, January 2019, pp. 21 – 38.

[23] H. Afzali-Kusha, M. Kamal, M. Pedram: Low-Power Accuracy-Configurable Carry Look-Ahead Adder Based on Voltage Overscaling Technique, Proceedings of the 21st International Symposium on Quality Electronic Design (ISQED), Santa Clara, USA, March 2020, pp. 67 – 72.

[24] M. Deivakani, S. V. Sudheer Kumar, N. Udaya Kumar, E. Fantin Irudaya Raj, V. Ramakrishna: VLSI Implementation of Discrete Cosine Transform Approximation Recursive Algorithm, Journal of Physics: Conference Series, Vol. 1817, March 2021, p. 012017.

[25] C. Dharmaraj, V. Vasudevan, N. Chandrachoodan: Optimization of Signal Processing Applications Using Parameterized Error Models for Approximate Adders, ACM Transactions on Embedded Computing Systems (TECS), Vol. 20, No. 2, January 2021, p. 12.

[26] J. Lee, H. Seo, H. Seok, Y. Kim: A Novel Approximate Adder Design Using Error Reduced Carry Prediction and Constant Truncation, IEEE Access, Vol. 9, August 2021, pp. 119939 – 119953.

[27] S. E. Fatemieh, S. S. Farahani, M. R. Reshadinezhad: LAHAF: Low-Power, Area-Efficient, and High-Performance Approximate Full Adder Based on Static CMOS, Sustainable Computing: Informatics and Systems, Vol. 30, June 2021, p. 100529.

[28] N. Manikandan, R. Muthaiah, Y. Teekaraman, R. Kuppusamy, A. Radhakrishnan: A Novel Random Error Approximate Adder-Based Lightweight Medical Image Encryption Scheme for Secure Remote Monitoring of Health Data, Security and Communication Networks, Vol. 2021, January 2021, p. 3570904.

[29] D. Ravichandran, S. Rajagopalan, H. N. Upadhyay, J. B. Balaguru Rayappan, R. Amirtharajan: Encrypted Biography of Biomedical Image - A Pentalayer Cryptosystem on FPGA, Journal of Signal Processing Systems, Vol. 91, No. 5, May 2019, pp. 475 – 501.

[30] S. N. Al Azzam, F. A. Al-Garni: The Use of Binary Digit Mapping on ASCII Characters to Create a High-Capacity, Undetectable Text Steganography, Journal of Advanced Sciences and Engineering Technologies, Vol. 5, No. 2, April 2023, pp. 49 – 57.