DESIGN OF APPROXIMATE REVERSE CARRY PROPAGATE ADDER FOR DSP APPLICATIONS

[1]Sharron Teena J, [2]Sridevi C

[1]PG Scholar, [2]Assistant Professor

[1,2]Department of Electronics

[1,2]Madras Institute of Technology, Anna University, Chennai, India

Abstract : Adders are the basic hardware unit that is used in most of the VLSI architectures. Adders are capable of replacing multipliers in such a way by performing repeated additions or by shift-add architecture. They are well preferred because of its low area and power consumption. A Full Adder circuit forms the basic circuit of many digital signal processing application. Many full adders have been designed till date of which conventional mirror adder is of prime importance. Inorder to reduce the power and delay of the adder the accuracy of the adder is compromised. Approximate Reverse Carry Propagate Full Adder (RCPFA) is a special kind of adder where the carry signal propagates from the MSB to LSB. Three types of RCPFAs are designed which have proved their efficacy when used in DCT applications and Image Enhancement techniques.

***IndexTerms* - Reverse Carry Propagate Adder, Digital Signal Processing.**

I. INTRODUCTION

The term Adder generally refers to the digital circuit that is used for performing addition of operands. Adders are generally used in the Arithmetic Logic Unit incase of computers, they are also used in different kinds of processors where they are implemented on to the part where address calculation takes place, in increment and decrement operators and in table indices. In any kind of VLSI design adders are considered as the basic building block, they consume low power, low area and are of high speed. Adders are the most significant component in digital platform because of its wide use in basic digital operations like subtraction, multiplication, division. Thus improving the performance of such digital adder would extensively fasten the execution of operation inside any circuit.

A full adder is an adder circuit that adds binary numbers and yields two results namely the sum and the carry. Generally one-bit full adder carries out addition of three inputs that is three one-bit numbers like A, B, C_{in}. Here A and B are the input operands. C_{in} is the bit carried from the LSB stage. S denotes the sum and C_{out} represents the output carry signal. The different combinations of input and their corresponding outputs are denoted in the truth table which is listed in **Table 1**.

Table 1: Truth Table for Full Adder

A	B	C_{in}	S	C_{out}
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Approximate Adders are designed with an aim to reduce the hardware complexity. Generally in the design of any VLSI circuit power, delay and area are the major constraint that is being considered. Tradeoff usually occurs with the performance parameters. Inorder to reduce both the power and the delay the accuracy of the circuit is being compromised, as a result of which we go for approximate adders. They tend to have a slightly modified value than the normal circuits. Sacrificing accuracy of the circuits leads to lowering of delay and power. Approximate adders are used in applications where accuracy is not the major constraint. They can be used in areas where performance parameters alone play a major role. They cannot be used in critical applications since they can cause a large difference and lead to erroneous output. With regard to full adder, approximate adders like Approximate Mirror Adder (AMA), Transmission Gate-based Adder (TGA), Approximate XOR/XNOR based Adder (AXA) have been discussed in the previous works.

1.1 Methodology

In this study, we propose an Approximate Reverse Carry Propagate Adder which can be efficiently used for DSP applications such as Image Enhancement and DCT compression. The RCPFAs are designed and they are implemented for the above applications wherein Image enhancement is performed at Simulink platform and the DCT works are carried out in Altera.

Comparing to the works discussed in the literature, the work presented in this paper is different in the following ways:

(1) Efficient adder with low Area-Delay-Power product is obtained.

(2) The proposed method provides a clear enhancement of the dull image.

II. RELATED STUDY

2.1 Performance Comparison of Adders

The performance of various approximate adders are compared with that of the conventional full adder. Performance metrics namely area, power and delay of 32-bit adders are compared in **Table 2** and simulation results show that Approximate Mirror Adder 5 is best in terms of area and delay, but the accuracy of AMA5 is poor so it is not taken into further comparison. The overall Area-Delay-Power product is calculated wherein the RCPFA1, RCPFA2, RCPFA3 shows better results than the other approximate adders. When higher order adder like 64-bit and 128-bit are used the RCPFAs exhibit high performance than other adders. Error analysis of approximate adders is shown in **Table 3** which states that RCPFAs have higher accuracy and low Mean Error Distance, hence proving that RCPFAs are the best approximate adders.

Table 2: Performance Analysis of various 32 bit Adders.

ADDER	AREA	DELAY (in ms)	POWER (in mW)
FA	66 / 33216	12.189	129.42
AMA 1	66 / 33216	12.189	129.34
AMA 2	33 / 33216	11.970	129.29
AMA 3	33 / 33216	11.970	129.29
AMA 4	33 / 33216	11.275	129.34
AMA 5	0 / 33216	8.984	129.44
TGA 1	33 / 33216	11.275	129.38
TGA 2	33 / 33216	12.036	129.38
AXA 1	33 / 33216	11.996	129.42
AXA 2	66 / 33216	12.357	129.33
AXA 3	66 / 33216	12.189	129.30
RCPFA 1	2 / 33216	9.315	137.32
RCPFA 2	35 / 33216	9.501	137.38
RCPFA 3	35 / 33216	11.283	137.45

Table 3: Error Analysis of various approximate adders

ADDER	ERROR %	μ	σ	MED
AMA1	18.75	0	60.36	34.57
AMA2	12.50	-2.00	85.31	59.69
AMA3	25.00	-1.00	94.03	71.31
AMA4	31.25	31.75	82.62	66.29
TGA1	25.00	-0.25	64.00	44.79
TGA2	25.00	-32.25	45.25	32.25
AXA1	43.75	255.50	128.0	255.50
AXA2	25.00	125.90	145.8	125.90
AXA3	12.50	108.75	167.3	108.75
RCPFA1	25.00	0	26.98	18.12
RCPFA2	12.50	18.12	19.84	18.12
RCPFA3	12.50	-18.12	19.84	18.12

Masoud et al,[1] proposed a Reverse Carry Propagate adder (RCPA) where the carry signal present in the adder propagates in a counter-flow manner that is from the Most Significant Bit to the Least Significant Bit. So here the carry input signal propagated has higher significance than the output carry signal. Three designs of Reverse Carry Propagate Full Adders (RCPFA) are designed with different power, area and delay. These adders are found to have high accuracy than the other approximate adders. Inorder to prove their efficacy RCPFAs are applied in DCT algorithm and in Image filtering. Using RCPFAs in the above applications shows that they consume low power, area and delay than the other approximate adders.

Vaibhav et al,[2] proposed that power is considered as the most important constraint for any multimedia application. Particularly in low power digital signal processing applications, inorder to reduce the power consumption accuracy of the adders are given up. Accuracy is sacrificed for the fact that humans can interpret information from the incorrect output, incase of multimedia application. Here five different models of Approximate Mirror Adders (AMA) are designed each with different sum and carry expressions. The experimental results prove that these adders consume low power than the conventional full adder. Inorder to prove the efficacy of these mirror adders, architectures have been developed for image and video compression. Simulation result shows an improvement of 69% in power consumption than the existing accurate adders.

Zhixi et al,[3] presented two different types of Transmission Gate-based Adders (TGA). Inorder to reduce the power dissipation in CMOS circuits, the accuracy these adders are sacrificed. These adders are used in applications where exactness is not the major criteria. Transmission gates are made use in the design of these adders and it also results in reducing the critical path delay. The designed adder is implemented in Image sharpening algorithm to prove its efficacy in power and delay. Experimental results reveal that TGA2 exhibits well in case of accuracy, while TGA1 shows better performance in image sharpening.

Swaroop et al,[4] presented three different designs of Approximate XOR/XNOR based Adders (AXA). All the three designs are based on XOR/XNOR gates with multiplexers and are implemented by using pass transistors. These adders are used in applications where accuracy is not the most important constraint. The performance metrics namely the area, delay and power are compared with the accurate XOR/XNOR adder and results show that the proposed adders consume low power and delay. Simulation results reveal that AXA1 has the better performance, AXA2 utilizes smaller area, AXA3 has shortest error distance and consumes low power. Additional drivers are required at the output of the adder to reduce the noise margin caused due to usage of pass transistors.

III. Overview

3.1 Approximate Reverse Carry Propagate Adder

Approximate Reverse Carry Propagate Full Adder (RCPFA) is one kind of approximate adder where it differs from the ordinary full adder in a way that the carry signal propagates in counter-flow manner that is from Most significant bit to Least significant bit. In conventional adders output carry signal is of higher importance as it propagates from Least significant bit to Most significant bit, whereas in RCPFA input carry signal is of more importance because of the reverse propagation. This propagation from Most Significant Bit to Least Significant Bit leads to higher stability. Three different designs of RCPFAs are designed each with different accuracy level, power, area and delay. The three RCPFAs are designed using cell libraries like the And-Or-Invert and the Or-And-Invert cells inorder to reduce the area. The performance metrics are compared with the other approximate adders and conventional Full Adder and the simulation results show that for higher order bits RCPFAs delay and power are lower than conventional adders. As the bit size increases the performance of RCPFAs are far better than the other approximate adders and conventional full adder in terms of power and delay. The truth tables of the three RCPFAs namely RCPFA1, RCPFA2, RCPFA3 are listed in **Table 4, 5** and **6**.

Table 4: Truth Table for RCPFA 1

A	B	C	F_i	S	Ca	F_{i+1}
0	0	0	0	0	0	0
0	0	0	1	1	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	1	0	0
0	1	0	1	1	0	0
0	1	1	0	0	1	0
0	1	1	1	0	1	0
1	0	0	0	1	0	1
1	0	0	1	1	0	1
1	0	1	0	0	1	1
1	0	1	1	0	1	1
1	1	0	0	1	0	1
1	1	0	1	1	0	1
1	1	1	0	0	0	1
1	1	1	1	1	1	1

Table 5: Truth Table for RCPFA 2

A	B	C	F_i	S	Ca	F_{i+1}
0	0	0	0	0	0	0
0	0	0	1	1	1	0
0	0	1	0	0	0	0
0	0	1	1	1	1	0
0	1	0	0	1	0	0
0	1	0	1	1	0	0
0	1	1	0	0	0	0
0	1	1	1	1	1	0
1	0	0	0	1	0	0
1	0	0	1	1	0	0
1	0	1	0	0	0	0
1	0	1	1	1	1	0
1	1	0	0	1	0	1
1	1	0	1	1	0	1
1	1	1	0	0	0	1
1	1	1	1	1	1	1

Table 6: Truth Table for RCPFA 3

A	B	C	F_i	S	Ca	F_{i+1}
0	0	0	0	0	0	0
0	0	0	1	1	1	0
0	0	1	0	0	1	0
0	0	1	1	0	1	0
0	1	0	0	0	0	1
0	1	0	1	1	1	1
0	1	1	0	0	1	1
0	1	1	1	0	1	1
1	0	0	0	0	0	1
1	0	0	1	1	1	1
1	0	1	0	0	1	1
1	0	1	1	0	1	1
1	1	0	0	0	0	1
1	1	0	1	1	1	1
1	1	1	0	0	0	1

1	1	1	1	1	1	1

IV. Results

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = 96969696, carry = E8E8E8E8

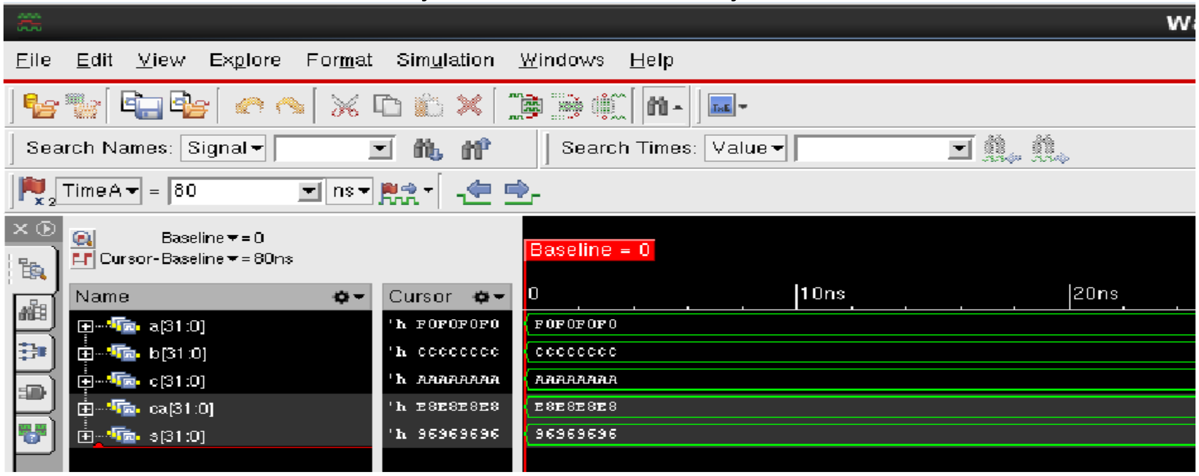

Figure 1: Simulation Waveform of 32-bit Full adder

Figure 1 represents the addition of three variables a, b and c using 32-bit Full adder, where a, b and c are the input variables. Sum and Carry of the FA are represented by s and ca respectively

a = F0F0F0F0, b = C`CCCCCCC, c = AAAAAAAA yields Sum = 82828282, carry = ECECECEC

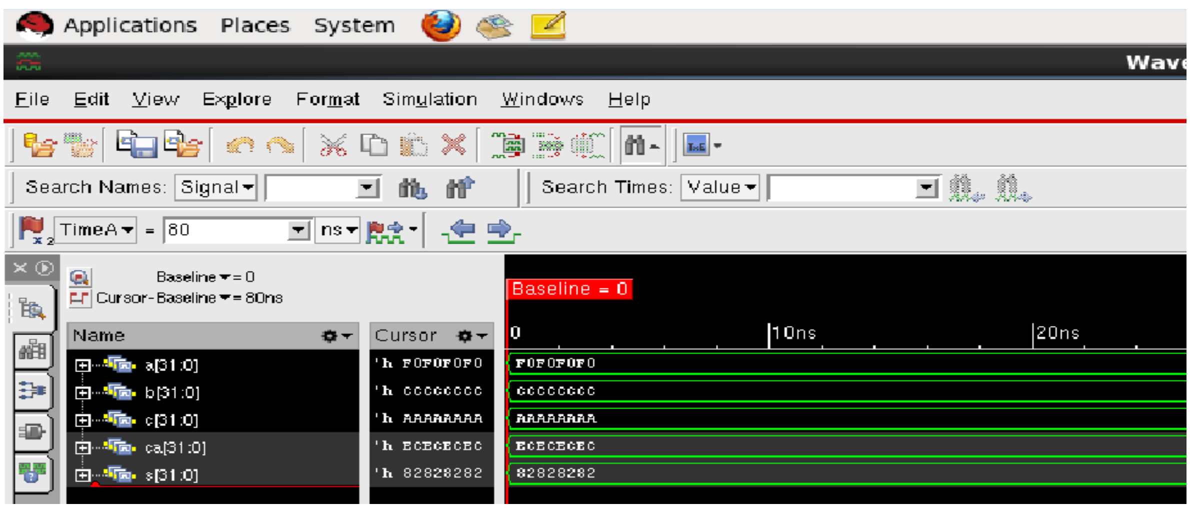

Figure 2: Simulation Waveform of 32-bit Approximate Mirror Adder 1 (AMA1)

Figure 2 represents the addition of three variables a, b and c using 32-bit Approximate Mirror adder 1, where a, b and c are the input variables. Sum and Carry of AMA1 are represented by s and ca respectively.

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = 17171717, carry = E8E8E8E8

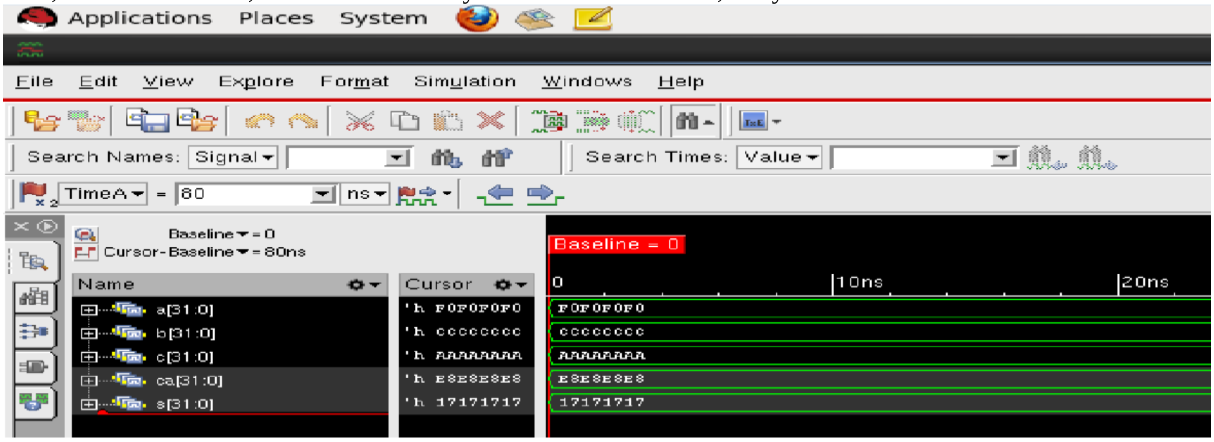

Figure 3: Simulation Waveform of 32-bit Approximate Mirror Adder 2 (AMA2)

Figure 3 represents addition of three variables a, b and c using 32-bit Approximate Mirror adder 2, where a, b and c are the input variables. Sum and Carry of AMA2 are represented by s and ca respectively.

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = 13131313, carry = ECECECEC

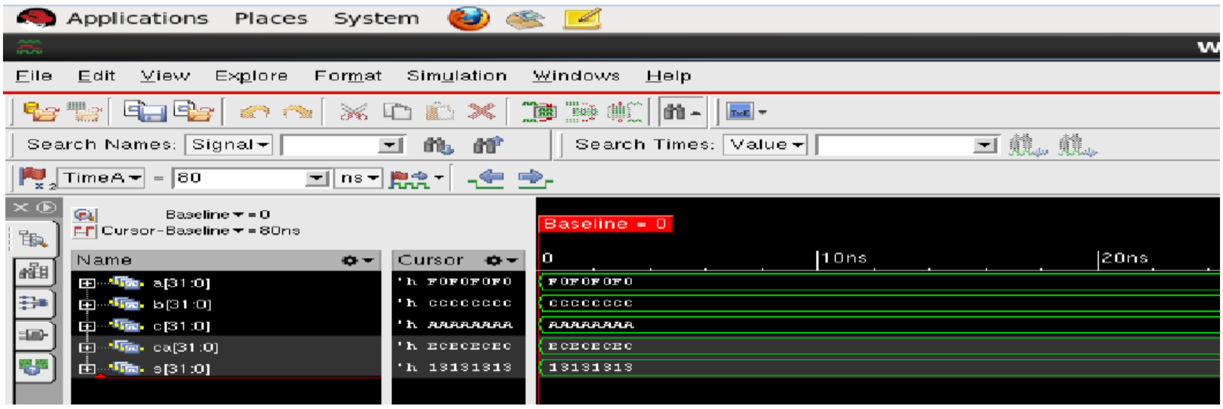

Figure 4: Simulation Waveform of 32-bit Approximate Mirror Adder 3 (AMA3)

Figure 4 represents addition of three variables a, b and c using 32-bit Approximate Mirror adder 3, where a, b and c are the input variables. Sum and Carry of AMA3 are represented by s and ca respectively.

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = 8A8A8A8A, carry = F0F0F0F0

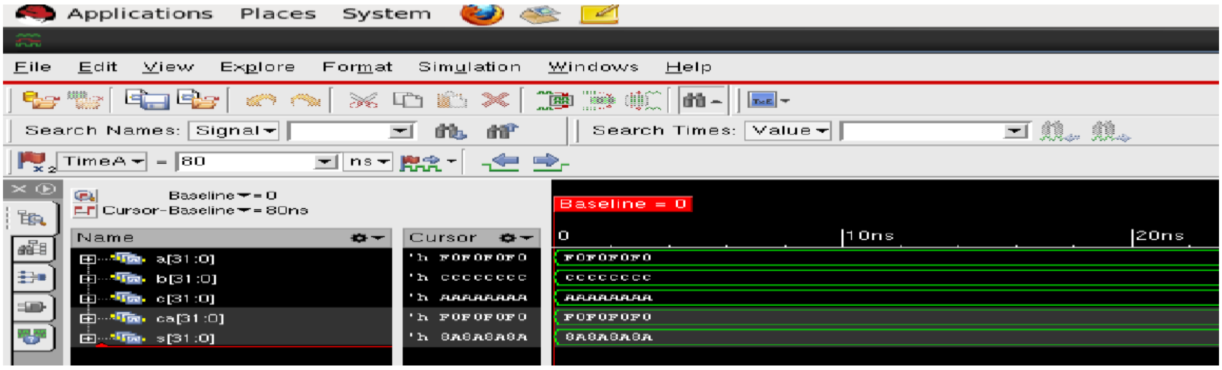

Figure 5: Simulation Waveform of 32-bit Approximate Mirror Adder 4 (AMA4)

Figure 5 represents addition of three variables a, b and c using 32-bit Approximate Mirror adder 4, where a, b and c are the input variables. Sum and Carry of AMA4 are represented by s and ca respectively.

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = CCCCCCCC, carry = F0F0F0F0

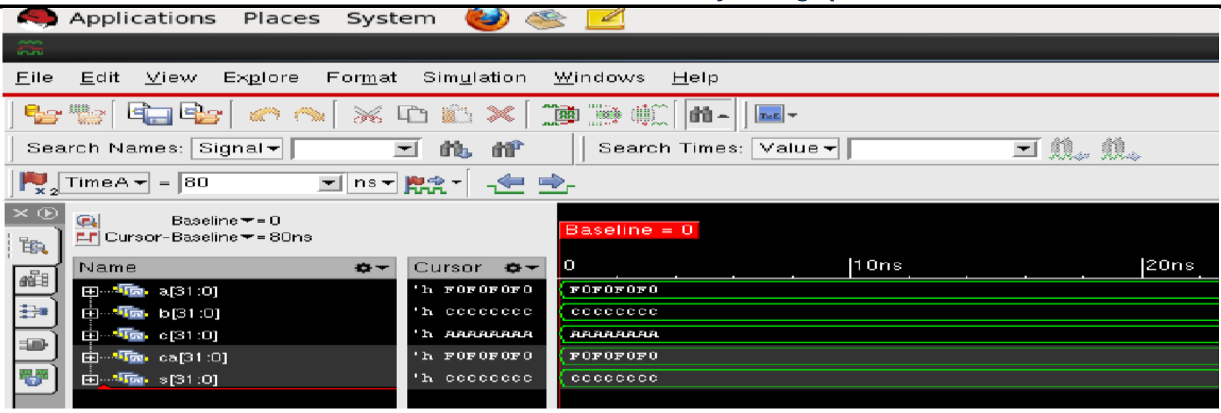

Figure 6: Simulation Waveform of 32-bit Approximate Mirror Adder 5 (AMA5)

Figure 6 represents addition of three variables a, b and c using 32-bit Approximate Mirror adder 5, where a, b and c are the input variables. Sum and Carry of AMA5 are represented by s and ca respectively.

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = B2B2B2B2, carry = CCCCCCCC

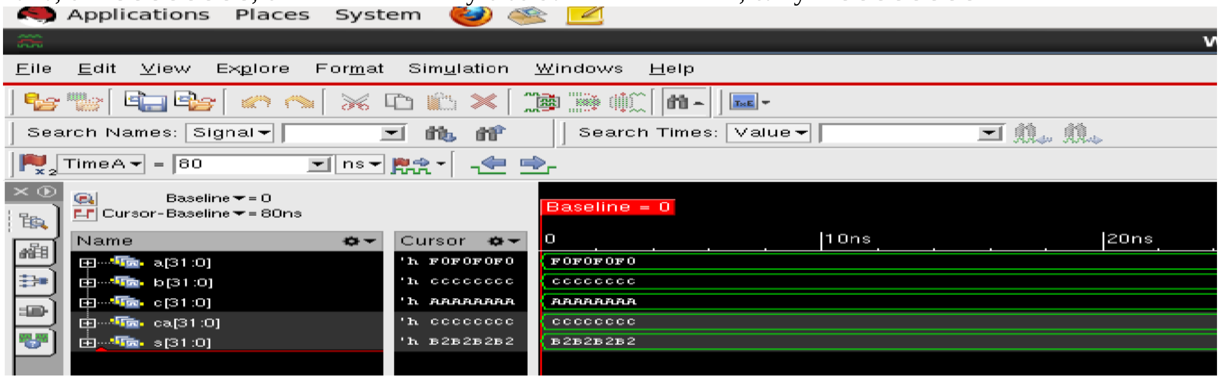

Figure 7: Simulation Waveform of 32-bit Transmission Gate-based Adder 1 (TGA1)

Figure 7 represents addition of three variables a, b and c using 32-bit Transmission Gate-based adder 1, where a, b and c are the input variables. Sum and Carry of TGA1 are represented by s and ca respectively.

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = 82828282, carry = FCFCFCFC

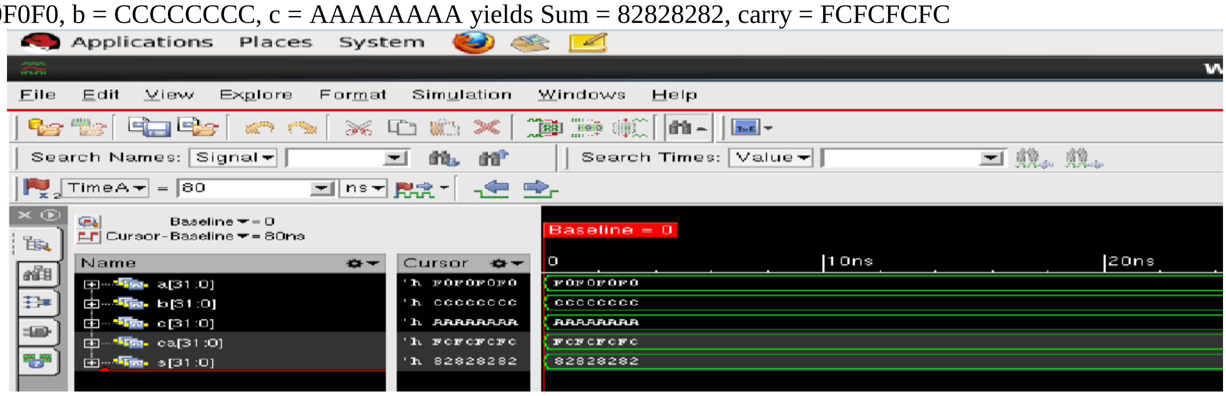

Figure 8: Simulation Waveform of 32-bit Transmission Gate-based Adder 2 (TGA2)

Figure 8 represents addition of three variables a, b and c using 32-bit Transmission Gate-based adder 2, where a, b and c are the input variables. Sum and Carry of TGA2 are represented by s and ca respectively.

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = AAAAAAAA, carry = 80808080

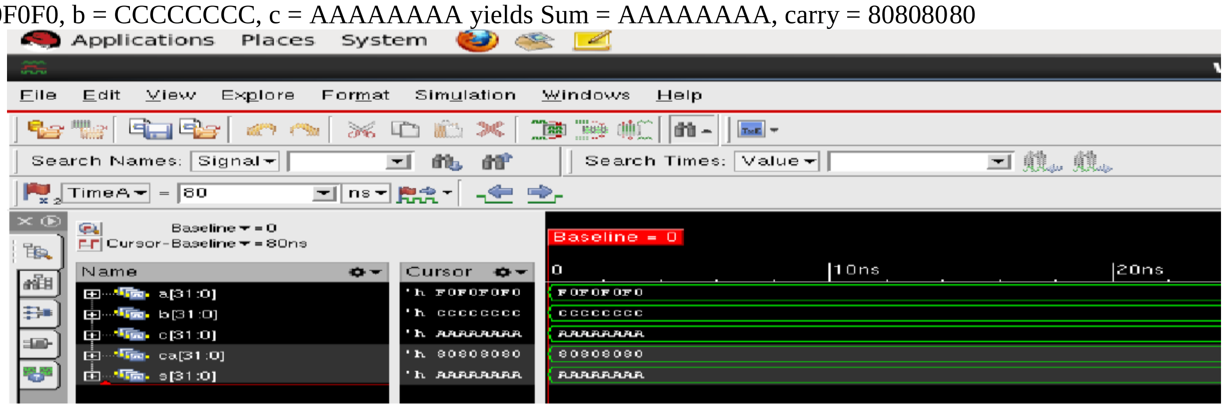

Figure 9: Simulation Waveform of 32-bit Approx. XOR/XNOR-based Adder 1 (AXA1)

Figure 9 represents addition of three variables a, b and c using 32-bit Approximate XOR/XNOR-based adder 1, where a, b and c are the input variables. Sum and Carry of AXA1 are represented by s and ca respectively.

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = C3C3C3C3, carry = E8E8E8E8

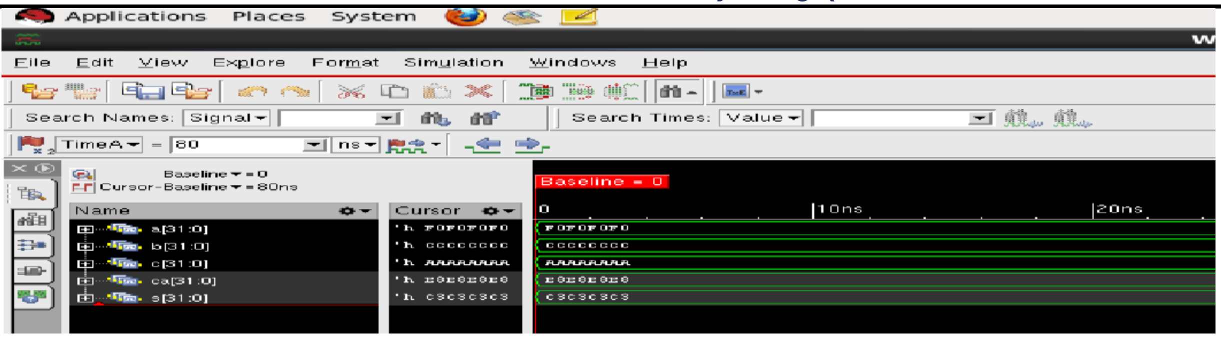

Figure 10: Simulation Waveform of 32-bit Approx. XOR/XNOR-based Adder 2 (AXA2)

Figure 10 represents addition of three variables a, b and c using 32-bit Approximate XOR/XNOR-based adder 2, where a, b and c are the input variables. Sum and Carry of AXA2 are represented by s and ca respectively.

a = F0F0F0F0, b = CCCCCCCC, c = AAAAAAAA yields Sum = 82828282, carry = E8E8E8E8

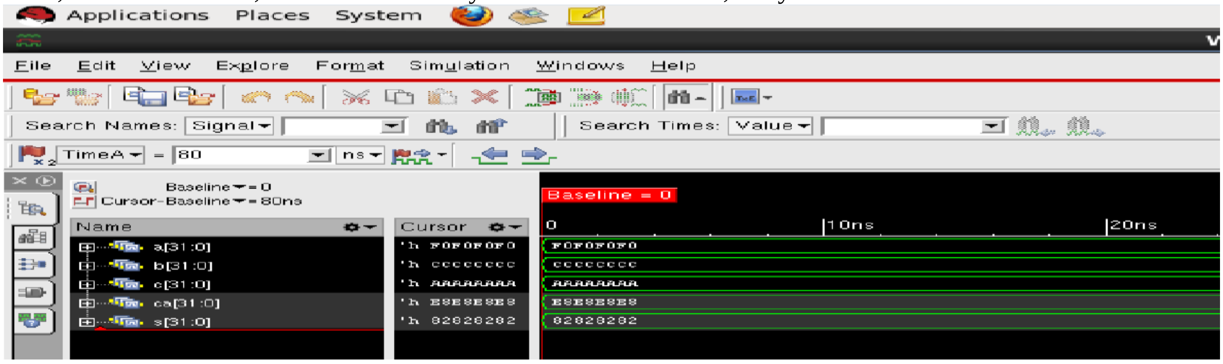

Figure 11: Simulation Waveform of 32-bit Approx. XOR/XNOR-based Adder 3 (AXA3)

Figure 11 represents addition of three variables a, b and c using 32-bit Approximate XOR/XNOR-based adder 3, where a, b and c are the input variables. Sum and Carry of AXA3 are represented by s and ca respectively.

a = FF00FF00, b = F0F0F0F0, c = CCCCCCCC, f_i = AAAAAAAA yields Sum = B332B332, carry = 8CCE8CCE

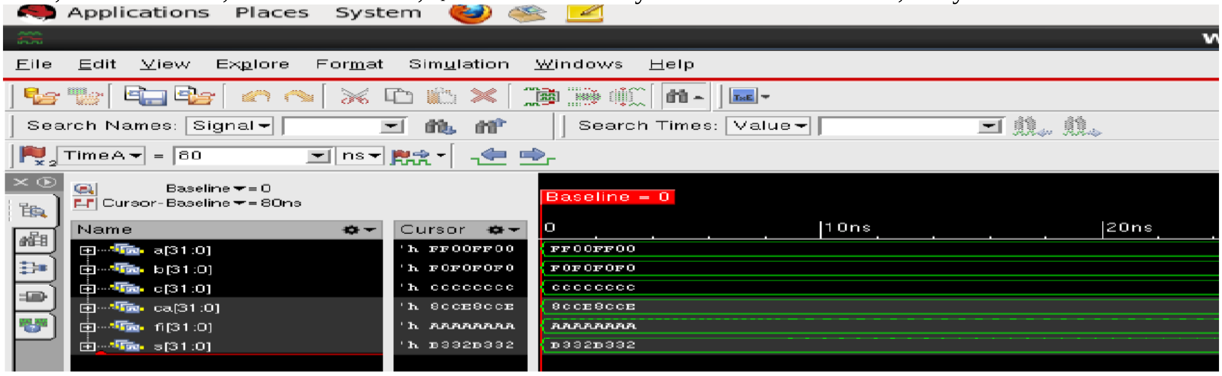

Figure 12: Simulation Waveform of 32-bit Approximate Reverse Carry Propagate Full Adder 1 (RCPFA1)

Figure 12 represents addition of three variables a, b and c and a forecast signal f_i using 32-bit Approximate Reverse carry propagate adder 1, where a, b, c and f_i are the input variables. Sum and Carry of RCPFA1 are represented by s and ca respectively.

a = FF00FF00, b = F0F0F0F0, c = CCCCCCCC, f_i = AAAAAAAA yields Sum = FFFFFFFE, carry = 00000000

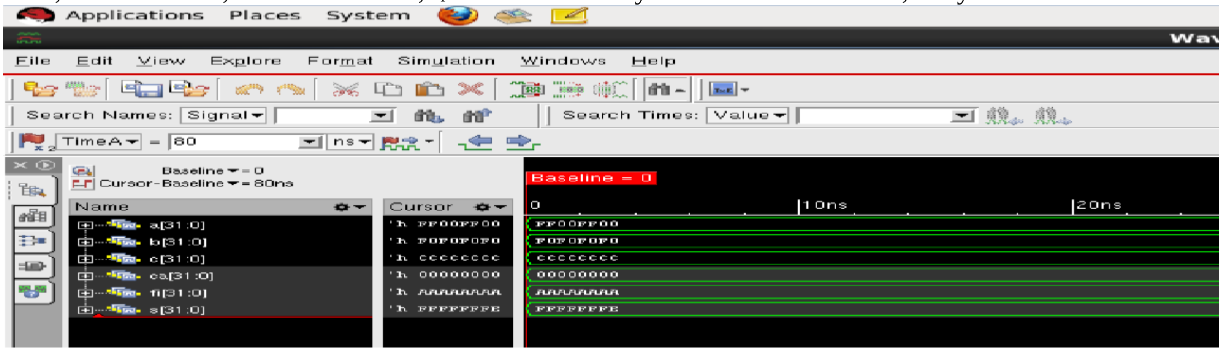

Figure 13: Simulation Waveform of 32-bit Approximate Reverse Carry Propagate Full Adder 2 (RCPFA2)

Figure 13 represents addition of three variables a, b and c and a forecast signal f_i using 32-bit Approximate Reverse carry propagate adder 2, where a, b, c and f_i are the input variables. Sum and Carry of RCPFA2 are represented by s and ca respectively.

a = FF00FF00, b = F0F0F0F0, c = CCCCCCCC, f_i = AAAAAAAA yields Sum = A222A222, carry = AEEEAEEE

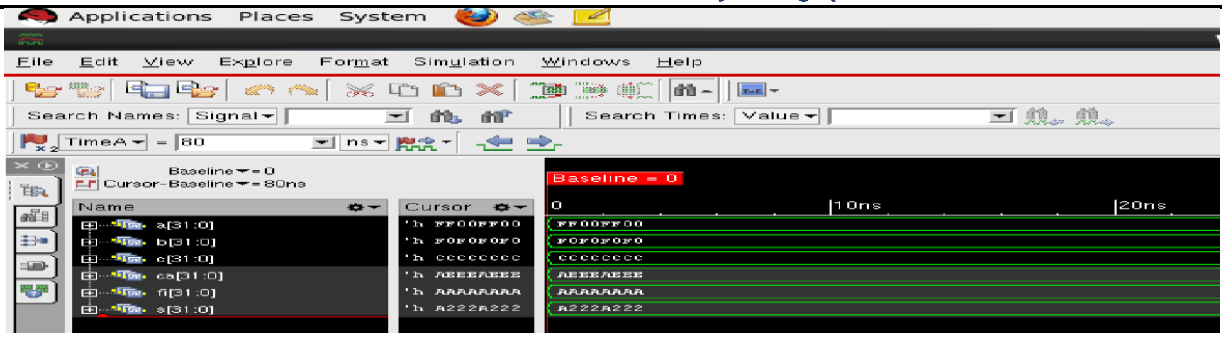

Figure 14: Simulation Waveform of 32-bit Approximate Reverse Carry Propagate Full Adder 3 (RCPFA3)

Figure 14 represents addition of three variables a, b and c and a forecast signal f_i using 32-bit Approximate Reverse carry propagate adder 3, where a, b, c and f_i are the input variables. Sum and Carry of RCPFA3 are represented by s and ca respectively.

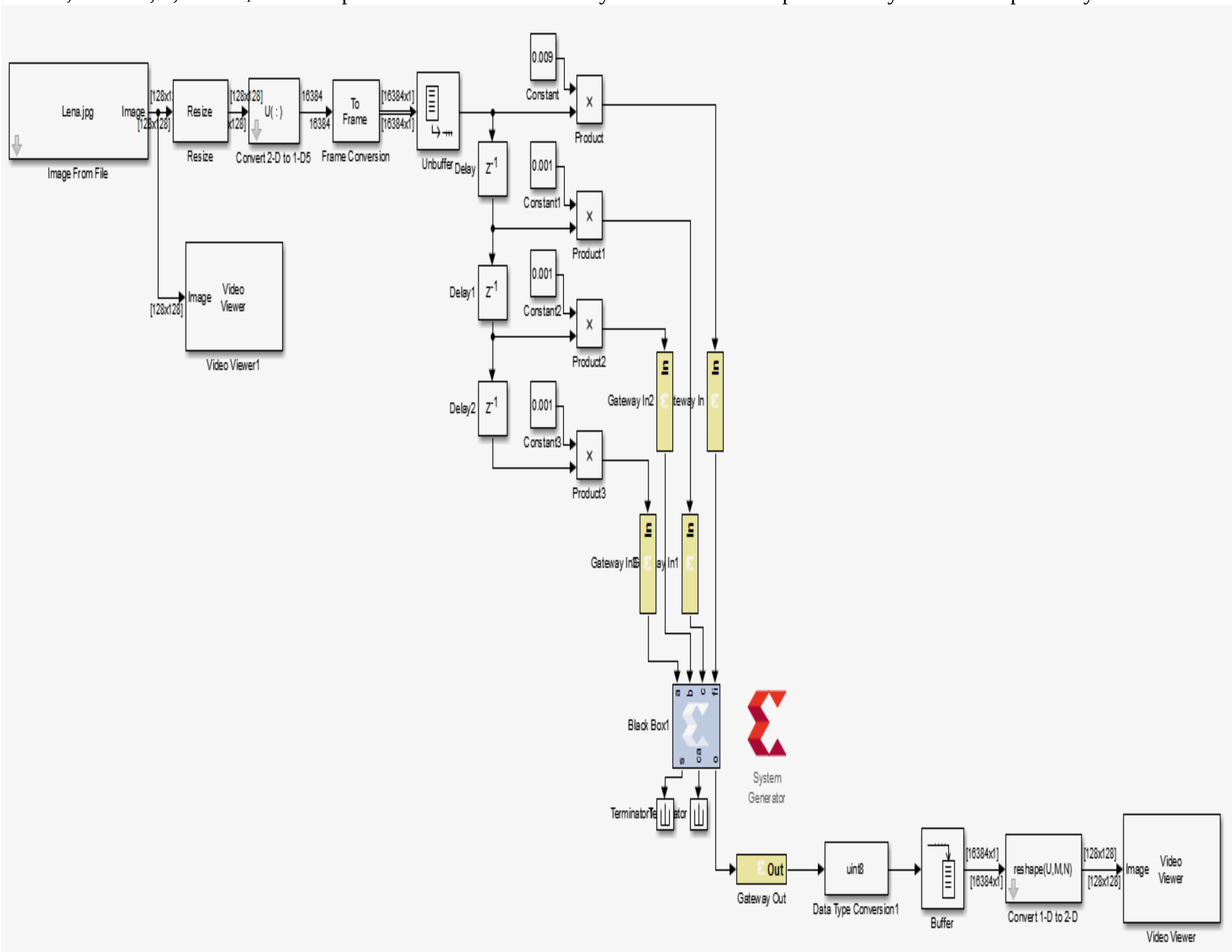

Figure 15: Simulink model of RCPFA1 generated using blackbox

Figure 15 represents the simulink model wherein the image is preprocessed for converting it from 2-D to 1-D, then the image is passed through the filter after which the reverse process namely conversion from 1-D to 2-D i.e the post processing is done to recover the image. The output image from various adders are listed in Fig IV.16

Figure 16: Images a. Original Image b. RCPFA1 c. RCPFA2 d. RCPFA3 e. AMA1 f.AMA2 g. AMA3 h. AMA4 i. AMA5 j. AXA1 k. AXA2 l. AMA3 m. TGA1 n.TGA2

V. CONCLUSION

Approximate computations holds for any application where exactness is not of much importance. Approximate adders are mainly used to lower the delay and power in any digital circuit. Accuracy is sacrificed in such circuits to improve the performance. Three designs of RCPFAs are implemented each of different accuracy, area, power and delay. Each of these adders are analyzed with variable bit size and simulation results reveal that RCPFAs show good performance for higher bit size. The efficacy of these adders are proved by implementing them in DCT algorithm and also by using them for Image Enhancement Applications. Results show that RCPFAs are efficient enough that they produce accurate output and good performance. These adders can also be used in DCT block of the JPEG compression technique for Image Compression. RCPFAs can be optimized such that they result in higher accuracy levels and better performance. RCPFAs can be used along with accurate adders to form hybrid adders and those hybrid adders can be used in DCT algorithm and in Image enhancement applications so as to improve the performance of the approximate adders. This work can be extended to Color images as well to utilize the approximate adders.

REFERENCES

[1] Masoud Pashaeifar, Mehdi Kamal, Ali Afzali-Kusha, Massoud Pedram, "Approximate Reverse Carry Propagate Adder for Energy-Efficient DSP Applications", IEEE Transactions on Very Large Scale Integration Systems, July 2018.

[2] Vaibhav Gupta, Debabrata Mohapatra, Anand Raghunathan, Kaushik Roy,"Low-Power Digital Signal Processing Using Approximate Adders", IEEE Transactions on Computer Aided Design on Integrated Circuits and Systems, Volume 32, January 2013.

[3] Zhixi Yang, Jie Han, Fabrizio Lombardi, "Transmission gate-based adders for inexact computing", IEEE/ACM International symposium on Nanoscience Architecture (NANOARCH), July 2015.

[4] Swaroop Ghosh, Debabrata Mohapatra, Georgios Karakonstantis, Kaushik Roy, "Voltage Scalable high-speed robust hybrid Arithmetic units using adaptive clocking", IEEE Transactions on Very Large Scale Integration Systems, Volume 18, September 2010.

[5] Ning Zhu, Wang Ling Goh, Weija Zhang, Kiat Seng Yeo, Zhi Hui Kong, "Design of low-power high-speed Truncation-Error-Tolerant adder and its application in Digital signal Processing", IEEE Transactions on Very Large Scale Integration Systems, Volume 18, August 2010.

[6] J. Kung, D. Kim, S. Mukhopadhyay, "On the impact of energy-accuracy tradeoff in a digital cellular neural network for image processing", IEEE Transactions on Computer Aided Design on Integrated Circuits and Systems, Volume 34, July 2015.

[7] Z. Yang, A. Jain, J. Liang, J. Han, F. Lombardi,"Approximate XOR/XNOR-based adders for inexact computing", IEEE International Conference on Nanotechnology (NANO), August 2013.

[8] I. C. Lin, Y. M. Yang, C. C. Lin, "High-performance low-power carry speculative addition with variable latency", IEEE Transactions on Very Large Scale Integration Systems, Volume 23, September 2015.

[9] T. Moreau, A. Sampson, and L. Ceze, "Approximate computing: Making mobile systems more efficient," IEEE Pervasive Comput., vol. 14, no. 2, pp. 9–13, Apr. 2015.

[10] A. Madanayake et al., "Low-power VLSI architectures for DCT/DWT: Precision vs approximation for HD video, biomedical, and smart antenna applications," IEEE Circuits Syst. Mag., vol. 15, no. 1, pp. 25–47, 1st Quart., 2015.

[11] H. R. Mahdiani, A. Ahmadi, S. M. Fakhraie, and C. Lucas, "Bio-inspired imprecise computational blocks for efficient VLSI implementation of soft-computing applications," IEEE Trans. Circuits Syst. I, Reg. Papers, vol. 57, no. 4, pp. 850–862, Apr. 2010.

[12] H. A. F. Almurib, T. N. Kumar, and F. Lombardi, "Inexact designs for approximate low power addition by cell replacement," in Proc. Design, Autom. Test Eur. (DATE), Mar. 2016, pp. 660–665.

[13] M. Shafique, W. Ahmad, R. Hafiz, and J. Henkel, "A low latency generic accuracy configurable adder," in Proc. ACM/EDAC/IEEE Design Autom. Conf. (DAC), Jun. 2015, pp. 1–6.

[14] Y. Kim, Y. Zhang, and P. Li, "An energy efficient approximate adder with carry skip for error resilient neuromorphic VLSI systems," in Proc. IEEE/ACM Int. Conf. Comput.-Aided Design, Nov. 2013, pp. 130–137.

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