# Carry based approximate full adder for low power approximate computing

Manickam Ramasamy
FETBE
UCSI University
Malaysia
ManickamRamasamy@ucsiuniversity.edu.my

G.Narmadha
Department of Electrical Engineering
Sethu Institure of Technology
Madurai, India
gnarmadhame@gmail.com

S. Deivasigamani
FECT
AIMST University
Malaysia
deivasigamani@aimst.edu.my

***Abstract*— Approximate computing can reduce the design complication with an increase in performance and power efficiency for error resilient applications. In most multimedia applications, we can gather valuable information from slightly erroneous outputs. Therefore, we do not need to produce accurate outputs. This brief deals with a new gate level logic modification approach for approximation of full adder, to take advantage of the relaxation of numerical exactness. The sum term of the conventional full adder is altered to reduce an area complexity by proposing carry based approximation adder(CBAA) to avoid critical XOR operation in conventional one. We demonstrate this concept by proposing various imprecise approximate type full adders with reduced complexity at the gate level, and utilize them to design approximate multi-bit adders. Simulation results indicate up to 98% power savings using the proposed approximate adders, when compared o existing implementations using accurate adders.**

***Keywords—carry based approximate adder, error %, low power.***

## I. INTRODUCTION

In multimedia signal processing and data mining applications, the exact computing units are not always necessary, which can tolerate error and can be replaced with approximate units. They can be replaced with their approximate counterparts. Research on inexact computing for error tolerant applications is on the rise. Adders and multipliers shape the key parts in these applications. The quick development of interest for portable and wireless computing frameworks is driving the requirement for ultra-low power frameworks.

As an essential logic component, a multiplier part significantly contributes the overall power consumption in microprocessor and signal processing systems. In that system performance dominated and degraded by speed of the multiplication operation is executed. Adder is an elementary block of multiplier and speed limiting element of multiplier as well. Therefore, there is a huge need for careful optimization of adder design.

The paper is structured as follows: Section II addresses the related works of approximate computing in adders whereas, section III explains the structure of proposed adder with various approximating types. The obtained simulation results for the proposed approximates are discussed in section IV. Finally, the conclusions are summarized in section V.

## II. RELATED WORKS

Previous works on low power approximate computing [1]–[3] focused on VOS (Voltage Over Scaling) in which supply voltage was varied beyond critical voltage and corresponding performance degradation was compensated by various algorithms like Algorithmic Noise Tolerance (ANT).

In [4][5], authors investigate the effect of lowering threshold voltage and supply voltage to design low power CMOS circuits and develop test generation procedure to characterize acceptable errors.

The author Y. Kim et al. [6] proposed an error magnitude reduction scheme to design an approximate adder with moderate error rate. In [7], proposed an error tolerant adder by splitting the operation by accurate and inaccurate parts. The author Wanget al .[8] presents 2's complement based multiplication algorithm using radix 4 booth algorithm to reduce a switching activity of partial product generation. In [9], author sintroduce a low-power carry speculative adder which separates adder as a carry generator and sum generator. To reduce a critical path delay and area overhead only one sum adder generator is used in block adder. In[10],authors show how to reduce logic complexity of a conventional mirror adder (MA) by reducing the number of transistors and internal node capacitances and demonstrate the proposed approximate adders in two digital signal processing architectures with quality constraints. In [11], two designs of approximate 4-2 compressors are introduced and used in partial product reduction tree of four variants of 8 × 8Dadda multiplier. The author Suganthi Venkatachalam et al [12], proposed an approximate of multipliers. Based on probability terms, the partial product of the multiplier altered and performance evaluated with an image processing application. In [13], various adder structures are analyzed based on the area and time consumption and time efficient carry select adder was proposed. The Sum based adder was proposed in [14] and its performance parameters are also compared with the existing structures.

## III. PROPOSED WORK

The objective of this work is to propose a new gate level modification, for assessing adder designs with respect to area and power efficiency for inexact computing. Figure 1 shows a conventional full adder(CFA) for one bit (precise); it is obtained using 2 XOR gates, 2 AND gates and an OR gate. The Boolean expressions used are,

$$\text{Sum} = a \oplus b \oplus \text{Cin} \quad (1)$$

$$\text{Carry} = (a \cdot b) + (a \oplus b.\ \text{Cin}) \quad (2)$$

978-1-7281-1557-3/19/$31.00 ©2019 IEEE

978-1-7281-1557-3/19/$31.00 ©2019 IEEE

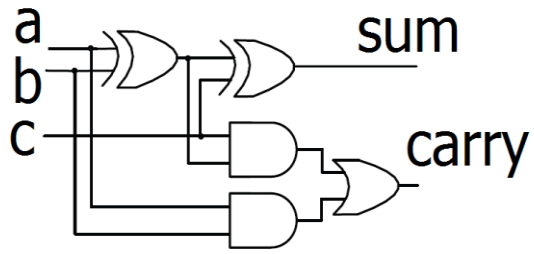


Figure 1. Conventional adder

Differently from conventional designs (as defined by its truth table), an approximate logic implementation alters some entries in the truth table. Propose a different type of approximate adders by calculating sum term from carry term.

In our approximation, one XOR gate, two AND gates and one OR gates are replaced with single OR gate and AND gate for carry calculation. Sum term is calculated by inverting carry term by using an inverter. This results in error in the four cases out of sixteen cases both in sum and carry terms. This provides more simplification, while maintaining the difference between original and approximate value as similar one for more cases. The truthtable of approximate full-adder is given in Table I. Four types of carry based approximations as follows.

*A. Carry based approximate adder1(CBAA)-type I*

It is obtained using one AND gate, one OR gate and a single inverter gate as shown in Figure2.The Boolean expressions are,

Carry1 = (a.b)+c; (3)

Sum1=~carry1 (4)

In this case, there is one error in Carry1 and 3 errors in Sum1 with total error distance(ED) of 3 as shown in Table I.

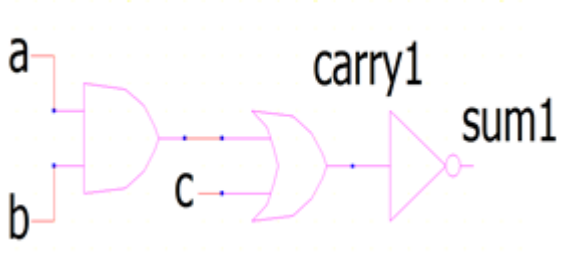


Figure 2. Carry Based approximate Adder1(CBA1)

*B. Carry based approximate adder2-Type-II*

It is obtained using one AND gate, one OR gate and single inverter gate as shown in Figure 3.The Boolean expressions are

Carry2 = (b.c)+a (5)

Sum2=~carry2 (6)

In this case, there is one error in Carry2 and 3 errors in Sum2 with total error distance(ED) of 3 as shown in Table I

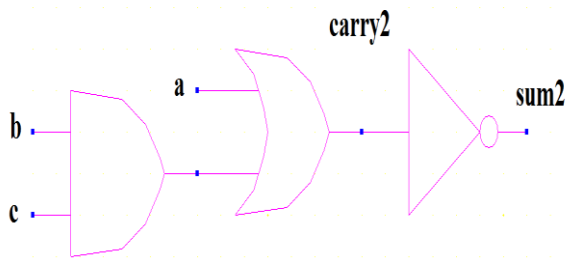


Figure 3. Carry Based approximate Adder2 (CBA2)

*C. Carry based adder2-Type-III*

It is obtained using one AND gate, one OR gate and single inverter gate as shown in Figure 4. The Boolean expressions are

Carry3 = (a.b) ⊕ c (7)

Sum3=~carry3 (8)

In this case, there are 2 errors in Carry3 and 1 error in Sum3 with total error distance(ED) of 3as shown in Table I.

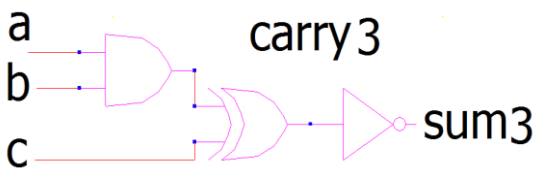


Figure 4. Carry Based approximate Adder3 (CBA3)

*D. Carry based approximate adder4-Type IV*

It is obtained using three AND gate, two OR gate and single inverter gate as shown in Figure 5. The Boolean expressions are,

Carry4 = a.b+b.c+c.a (9)

Sum4=~carry4 (10)

In this case, there is no error in Carry4 and 2 errors in Sum4 with total error distance(ED) of 2 as shown in Table I

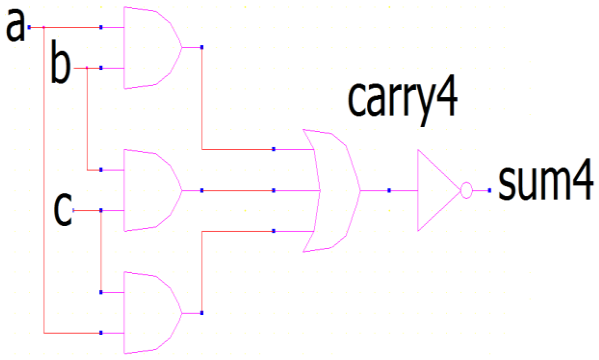


Figure 5. Carry Based approximate Adder4 (CBA4)

**Table 1: Truth Table for Conventional adder and Carry Based Approximate Adders 1–4**

| Inputs | | | Accurate outputs | | Approximate outputs | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | | CBA1 | | | CBA2 | | | CBA3 | | | CBA4 | | |
| A | B | C | Sum | Carry | Sum 1 | Carry 1 | ED | Sum 2 | Carry 2 | ED | Sum 3 | Carry 3 | ED | Sum 4 | Carry 4 | ED |
| 0 | 0 | 0 | 0 | 0 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 0 | 0 | 1 | 0 | 1 |
| 0 | 0 | 1 | 1 | 0 | 0 | 1 | 2 | 1 | 0 | 0 | 0 | 1 | 2 | 1 | 0 | 0 |
| 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 |
| 0 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 2 | 1 | 0 | 0 | 1 | 0 | 0 |
| 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 0 | 0 | 1 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 | 0 | 1 | 0 |
| 1 | 1 | 1 | 1 | 1 | 0 | 1 | 1 | 0 | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 1 |

## IV PERFORMANCE ANALYSIS

The conventional and proposed carry based adders are simulated by using Synopsis software. The performance is compared based on the trade off parameters such as delay, area and power consumption. Error percentage of approximate adders is also computed and is given in Table 2. The performance is analyzed for various bit sizes such as 16, 32, 64 and 128 bits.

## V CONCLUSION

In this brief, a carry based approximate adder can be used for low power applications. The proposed carry based area and power cost metrics in comparison with conventional one. The compared results show that our 128 bits carry based adder type I has significantly reduce the delay, area and power by 23.56 %, 61.63% and 98.128% respectively. Therefore, the proposed approximate architecture results in less area, lower delay, simple and power efficient implementation for hardware structures.

**Table 2: Performance Comparison**

| Bit size | Type | Delay (nS) | Area (µm) | Power (µW) | Error(%) |
|---|---|---|---|---|---|
| 16 | Conventional | 1.45 | 684.15 | 1940 | 0 |
| | CBAA-Type I | 0.86 | 262.61 | 60.79 | 6.2 |
| | CBAA-Type II | 1.01 | 262.61 | 84.49 | 6.2 |
| | CBAA-Type III | 2.02 | 410.92 | 818.21 | 4 |
| | CBAA-Type IV | 1.37 | 403.50 | 503.14 | 3.2 |
| 32 | Conventional | 2.52 | 1368.15 | 2880 | 0 |
| | CBAA-Type I | 1.69 | 525.08 | 94.72 | 4 |
| | CBAA-Type II | 2 | 525.08 | 169.58 | 4 |
| | CBAA-Type III | 3.71 | 821.71 | 1890 | 3 |
| | CBAA-Type IV | 2.52 | 806.87 | 1010 | 2 |
| 64 | Conventional | 4.61 | 2736.27 | 7760 | 0 |
| | CBAA-Type I | 3.46 | 1049.97 | 159.14 | 1.2 |
| | CBAA-Type II | 4.08 | 1049.97 | 367.04 | 0.6 |
| | CBAA-Type III | 6.98 | 1643.31 | 4130 | 0.65 |
| | CBAA-Type IV | 4.91 | 1613.56 | 2050 | 0.062 |
| 128 | Conventional | 8.87 | 5472.29 | 15540 | 0 |
| | CBAA-Type I | 6.78 | 2099.85 | 290.88 | 0.005 |
| | CBAA-Type II | 8.05 | 2099.85 | 707.31 | 0.005 |
| | CBAA-Type III | 13.73 | 3286.45 | 8660 | 0.001 |
| | CBAA-Type IV | 9.52 | 3227.02 | 4070 | 0.002 |

## REFERENCES

[1] R. Hegde and N. Shanbhag, “Soft digital signal processing,” IEEE Trans.on Very Large Scale Integration Systems, vol. 9, no. 6, pp. 813–823, Dec.2001.

[2] G. Varatkar and N. Shanbhag, “Error-resilient motion estimation architecture,”IEEE Trans. on Very Large Scale Integration Systems, vol. 16,no. 10, pp. 1399–1412, Oct. 2008.

[3] G. Karakonstantis, D. Mohapatra, and K. Roy, “System level DSP synthesis using voltage over scaling, unequal error protection and adaptive quality tuning,” in Proc. IEEE Workshop on Signal Processing Systems,Oct. 2009, pp. 133–138.

[4] S. Shahidi and S. Gupta,“A theory of error-rate testing,” in Proc. IEEEInt. Conf. on Computer Design, Oct. 2006, pp. 438–445.

[5] R. Gonzalez, B. Gordon, and M. Horowitz, “Supply and Threshold voltage scaling for low power CMOS,” IEEE Trans. on Solid-State Circuits, vol. 32, no. 8, pp. 1210–1216, Aug. 1997.

[6] 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. on Computer-Aided Design, Nov. 2013, pp. 130–137.

[7] N. Zhu, W. L. Goh, W. Zhang, K. S. Yeo, and Z. H. Kong, “Design of low-power high-speed truncation-error-tolerant adder and its application in digital signal processing,” IEEE Trans. on Very Large Scale Integration Systems, vol. 18, no. 8, pp. 1225–1229, Aug. 2010.

[8] O. T. C. Chen, S. Wang, and Y. W. Wu, “Minimization of switching activities of partial products for designing low-power multipliers,” IEEE Trans. on Very Large Scale Integration Systems, vol. 11, no. 3, pp. 418–433, Jun. 2003.

[9] I. C. Lin, Y. M. Yang, and C. C. Lin, “High-performance low-power carry speculative addition with variable latency,” IEEE Trans. on Very Large Scale Integration Systems, pp. 1–1, 2014.

[10] V. Gupta, D. Mohapatra, A. Raghu Nathan, and K. Roy, “Low-power digital signal processing using approximate adders,” IEEE Trans. OnComputer-Aided Design of Integrated Circuits and Systems, vol. 32,no. 1, pp. 124–137, Jan. 2013.

[11] A. Momeni, J. Han, P. Montuschi, and F. Lombardi, “Design and analysis of approximate compressors for multiplication,” *IEEE Trans.Comput.*, vol. 64, no. 4, pp. 984–994, Apr. 2015.

[12] Suganthi Venkatachalam and Seok-Bum Ko,” Design of Power and Area Efficient Approximate Multipliers”, IEEE Transactions on Very Large Scale Integration (VLSI) Systems (Volume: 25, Issue: 5, May 2017 )

[13] G.Narmadha, Dr. K. Balasubadra , S. Deivasigamani “Design and Implementation of Time Efficient Carry Select Adder using FPGA”, International Journal of Applied Engineering Research, ISSN 0973-4562, Volume 10, Issue Number 3 (2015), pp. 7215-7227.

[14] G. Narmadha, Dr. K. Balasubadra, “An optimized Montgomery Multiplier for Public Key Cryptography”, Vol. VII/Issue I/Jan.-March, 2016.