

FPGA Implementation of 4:2 Approximate Compressor Using Water Marking Applications

Abarna S, Joy Roseline Ebenezer D, Krishna Harini R

Department of ECE, K.Ramakrishnan College of Technology, Tamil Nadu, India

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 Compact 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 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.

I. INTRODUCTION

Minimizing 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 tolerant applications. To achieve energy and area efficiency achieve energy efficiency, high speed, and to be reliable for with high speed for the high quantity digital data 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 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 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.

II. EXISTING SYSTEM

It 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

$$(2). \text{Carry} = (A \oplus B) \cdot C + A \cdot B \quad (1)$$

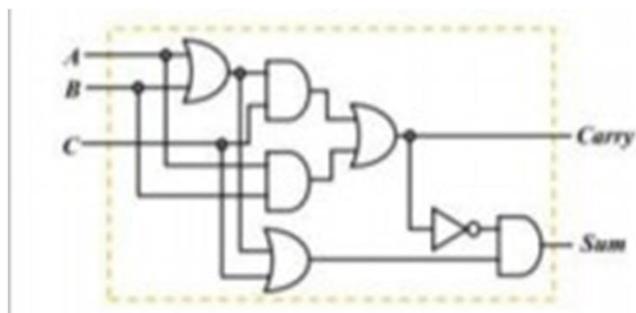
$$\text{Sum} = (A \oplus B) \oplus C \quad (2)$$

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]. 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' which is shown in equations. 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 [1]. Logic complexity reduction at the gate-level AFAs is presented in. In Modified Full Adder (MFA) , carry output has two errors out of eight cases and it has accurate sum output function. 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 . 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.

SIMPLIFIED APPROXIMATE FULL ADDER 1E



A	B	C	CY	S	ED
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	1	0	0
1	0	0	0	1	0
1	0	1	1	0	0
1	1	0	1	0	0
1	1	1	1	0	-1

III. PROPOSED SYSTEM

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).

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

$$\text{Sum} = \text{Carry} \cdot (A + B + C) \quad (4)$$

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.

$$\text{Sum} = A \cdot (B + C) + B \cdot C \quad (5)$$

$$\text{Carry} = A \quad (6)$$

$$\text{Sum} = A \cdot (B + C) \quad (7)$$

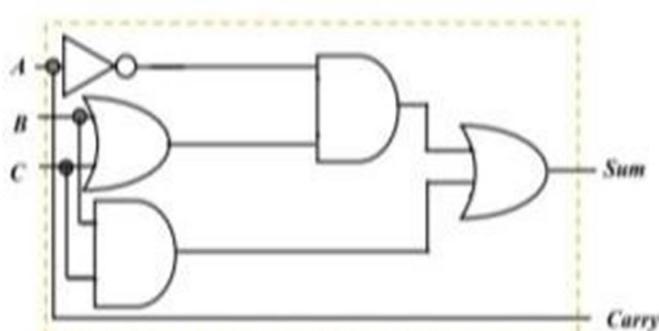
$$\text{Carry} = A \quad (8)$$

$$\text{Sum} = B + C \quad (9)$$

$$\text{Carry} = A \quad (10)$$

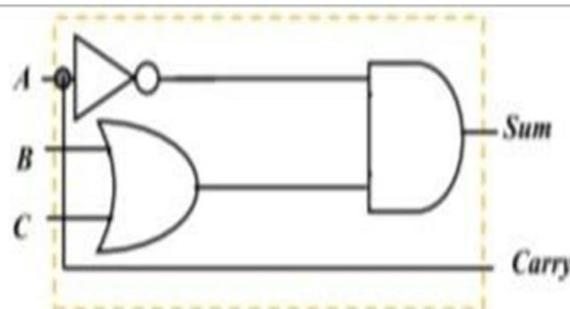
Similarly, Figs. 2-4 show the general gate-level architecture of the proposed SAFA2E, SAFA3E, SAFA4E designs, which have 5, 3, 1 basic logic gates respectively and these gates are employed on the sum logic formulation. The most significant bit (A) of the inputs is directly assigned as a carry output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

SIMPLIFIED APPROXIMATE FULL ADDER 2E



A	B	C	CY	S	ED
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	0	1	-1
1	0	0	1	0	+1
1	0	1	1	0	0
1	1	0	1	0	0
1	1	1	1	1	0

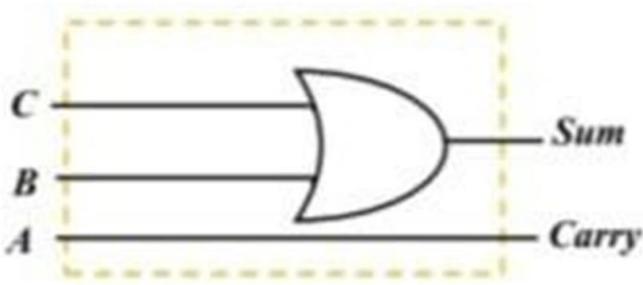
SIMPLIFIED APPROXIMATE FULL ADDER 3E



A	B	C	CY	S	ED
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	0	1	-1
1	0	0	1	0	+1
1	0	1	1	0	0
1	1	0	1	0	0
1	1	1	1	0	-1

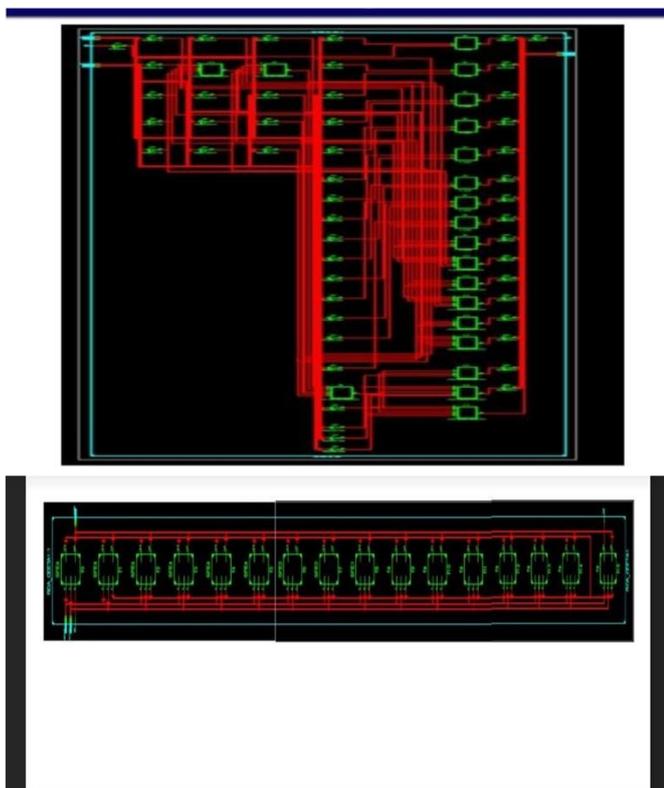
SIMPLIFIED APPROXIMATE FULL ADDER 4E

IV. CONCLUSION



A	B	C	CY	S	ED
0	0	0	0	0	0
0	0	1	0	1	0
0	1	0	0	1	0
0	1	1	0	1	-1
1	0	0	1	0	+1
1	0	1	1	1	+1
1	1	0	1	1	+1
1	1	1	1	1	0

IMPLEMENTATION RESULTS



Here we can design a high speed area efficient 4:2 compressor architecture and by introducing error we can gain area efficiency i.e by reducing components (gates) to get reduced size of the circuit.

V. 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[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.
- [11].S. Geetha, P. Amritvalli, High speed error tolerant adder for multimedia applications, J. Electron. Test. 33 (5) (2017) 675–688.
- [12].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>.
- [13].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.
- [14].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.