

Efficient design and analysis of Error tolerant approximate computing-ETAC through Partial Product Perforation.

A.Balamanikandan¹, K.Krishnamoorthi²

¹Electrical and Electronics Engineering, Adhiyamaan college of Engineering, Hosur (TN), 635 109 India.

²Electrical and Electronics Engineering, Sona college of Technology, Salem (TN), 636 005 India.

Abstract: In modern VLSI technology, we get useful information accompanied with slightly erroneous outputs. To reduce the error, a novel error-tolerant approximate computing (ETAC) is proposed. The objective of this research is to develop a new approximate multiplier and utilize modified 4:2 compressor and discussed its performance. The usage of compressor in the proposed model will improve the efficiency and minimizes the processing time, since the power consumption and performance is generally depends on the area required for doing the operation. Using OR gate in the accumulation of column wise generate elements in the altered partial product matrix provides exact result in most of the cases. The power consumption and its silicon area reduced dramatically by 9.24% and 4.88% in the proposed multiplier compared with conventional Wallace multipliers.

Keywords: Approximate computing, Compressor, ETA, Low power, Multiplication.

1. INTRODUCTION

Digital Signal processing (DSP) blocks plays a significant role in most of the multimedia applications. The absolute output is always embedded with error and human beings have perceptual abilities to accept these erroneous outputs [1]. Since large number of applications is error tolerant, approximate computing is widely used. Approximate computing is emerged as a promising solution for digital systems to enhance performance by limiting its energy consumption, processing time and manufacturing costs. It can be applied at both software and hardware level [2].

Hardware level approximate circuits mostly targets adders and multipliers in DSP applications. The commonly used techniques to build arithmetic circuits are truncation of lower order bits, aggressively scaling voltage and simplification of logic complexity (i.e.) alteration of truth table [2]. Several researches have been conducted so far on approximate adders which yields significant results in terms of reduction in area and power along with some error. However, researches on design of approximate multiplier have received less attention

due to lack of approximate techniques in partial product generation.

Studies on multipliers have to be focused since the speed of multipliers determines the processor's running speed in most of the digital systems such as digital filters, general purpose microprocessors, 3-D graphics applications, motion estimation accelerators, etc. The multiplier architecture consists of mainly three stages (i) partial product generation stage (ii) partial product reduction stage and (iii) the final addition stage [3]. Among these, the second stage contributes most to the overall delay, power consumption and high fraction of area. This delay of partial product accumulation is minimized by employing 4-2 compressors. These 4-2 compressors are facile to construct due to their regular interconnection and simple structure [4].

The approximation can be applied to the arithmetic units at various design abstraction levels such as circuit, logic, and architecture levels as well as algorithm and software layers. The approximation can be done on several techniques including some timing violations (voltage over scaling or over clocking) and function approximation methods (modifying the Boolean function of a circuit) or combining both the techniques [5]. In the later method, adders and multipliers are used as building blocks.

A new approximate multiplier is constructed by simply modifying the adder and compressor. The power consumption and design area of the new multiplier was implemented in Verilog HDL using a180-nm library. The proposed approximate multiplier reduces the power consumption by 9.24% as compared to the conventional Wallace tree multiplier. Additionally its design area minimized to 4.88%. In this paper, we proposed a new partial product generation technique called ETAC.

➤ It consists of accurate part (higher order bits) and inaccurate part (lower order bits).

➤ An inaccurate 4:2 compressor can be used to build energy efficient 4x4 Wallace multiplier.

➤ The lower order partial product bits of the multiplier (inaccurate part) require a special addition mechanism.

2. PREVIOUS WORK

The conventional ripple carry adder is not suitable when the inputs are very large owing to its delay in carry propagation. Hence researches focused on developing fast adders such as the carry-skip adder (CSK), carry-select adder (CSL), carry-look-ahead adder (CLA) [6, 7] and error tolerance adder (ETA) have been motivated along with many different low power design techniques. Moreover, there have always been tradeoffs between both area and power. This problem can be overcome by a suitable error tolerant design called ETAC. By utilizing this ETAC, we can achieve better performance in terms of both power and area by compensating some accuracy.

In order to reduce the number of signals generated, OR gate is used in the altered partial product matrix. The probability of error (P_{err}) is listed in Table I. From the table it is clear that the probability of mismatch is very low. The error probability increases linearly with the number of generate signals. On the other hand, the error value also rises. Hence the generate signals has to be minimized is the better solution to reduce the error. Thus by using OR gate, the maximum number of generate signals to be grouped. For a column having m generate signals, $m/4$ OR gates are used.

Table I. Truth Table of Approximate Full adder

Inputs			Exact output		Approximate output		Absolute difference
X1	X2	X3	CARRY	SUM	CARRY	SUM	
0	0	0	0	0	0✓	0✓	0
0	0	1	0	1	0✓	1✓	0
0	1	0	0	1	0✓	1✓	0
0	1	1	1	0	1✓	0✓	0
1	0	0	0	1	0✓	1✓	0
1	0	1	1	0	1✓	0✓	0
1	1	0	1	0	0✗	1✗	1
1	1	1	1	1	1✓	0✗	1

Approximate half-adder, full-adder, and 4-2 compressor are utilized for their accumulation. Among the two outputs carry and sum, Carry has higher weight of binary bit. So, carry has higher contribution of error in circuits. Approximation is defined that the difference between exact and approximate output is one. Hence we approximated Carry outputs only where Sum is approximated.

XOR gates tend to occupy high area and responsible for delay in adders and compressors. Thus, OR gate is used in place of XOR gate for approximating half-adder. This results in one error out of eight cases as listed in table below. A tick mark represents that approximate output and exact output are correctly matched and cross mark represents mismatch [8].

$$\text{Sum} = x1 + x2 \quad (1)$$

$$\text{Carr} y = x1 \cdot x2. \quad (2)$$

In 4-2 compressor, we require three bits only when all the four inputs are one which is possible only once out of all 16 cases. The output "100" (value of 4) has four inputs equal to one has changed to output "11" (value of 3). Therefore by changing the output value the minimal error difference is preserved as one. This property is used to remove one of the three output bits in the compressor. Among the three XOR gates, one XOR gate is changed with OR gate for sum computation. An additional circuit $x1 \cdot x2 \cdot x3 \cdot x4$ is added along with Sum expression in order to keep the sum matching to the case where all inputs are ones as one. This results in five errors out of all 16 cases. Carry is simplified according to eq (4). The truth table is given in Table II

Table II. Truth Table of Approximate 4-2 Compressor

INPUTS				Approximate output		Absolute difference
X1	X2	X3	X4	CARRY	SUM	
0	0	0	0	0✓	0✓	0
0	0	0	1	0✓	1✓	0
0	0	1	0	0✓	1✓	0
0	0	1	1	1✓	0✓	0
0	1	0	0	0✓	1✓	0
0	1	0	1	0✗	1✗	1
0	1	1	0	0✗	1✗	1
0	1	1	1	1✓	1✓	0
1	0	0	0	0✓	1✓	0
1	0	0	1	0✗	1✗	1
1	0	1	0	0✗	1✗	1
1	0	1	1	1✓	1✓	0
1	1	0	0	1✓	0✓	0
1	1	0	1	1✓	1✓	0
1	1	1	0	1✓	1✓	0
1	1	1	1	1✗	1✗	1

$$W1 = x1 \cdot x2 \quad (3)$$

$$W2 = x3 \cdot x4 \quad (4)$$

$$Sum = (x1 \oplus x2) + (x3 \oplus x4) + W1 \cdot W2$$

$$Carr y = W1 + W2.$$

3. PROPOSED ERROR TOLERANT APPROXIMATE COMPUTING

The proposed multiplier is designed to keep probability of error at minimal rate. A binary multiplier generally made up of three parts.

- (i) First part uses OR gate for Partial product generation
- (ii) Second part uses adder tree for Partial product reduction
- (iii) Final results are produced by using Carry propagation adder

Power consumption, delay and circuit complexity are mainly contributed by the PPR in the design of the multiplier [9]. Compressors are used to reduce these problems. Therefore an approximate multiplier is used in place of the exact compressor with minimum circuit complexity, power consumption and delay by sacrificing some accuracy.

Figure 1 shows the 8 by 8 bit Wallace multiplier. It is split into two parts as accurate part -2^6 to 2^{14} (Most significant part) and approximate part -2^0 to 2^5 (Least significant part). The approximate compressor is used in the least significant bits to reduce power consumption and circuit area and accurate compressor is used in the most significant bits to minimize the loss of accuracy. And the key idea here is the elimination or curtails the carry propagation which is solely responsible for the speed performance of the multiplier. The addition process starts from the middle joining point of the two parts.

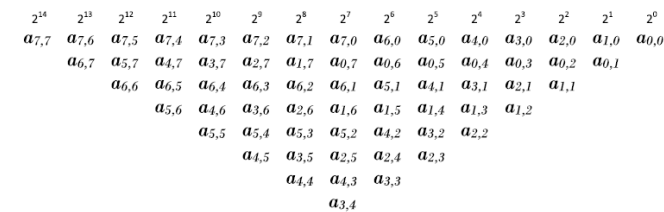


FIGURE 1. WALLACE MULTIPLIER (PARTIAL PRODUCT REDUCTION).

The partial product addition of the higher order bits (accurate part) of the input operands is performed by normal addition method from right to left (LSB to MSB) to preserve its correctness. Because the higher order bits plays a vital role than the lower order bits

[6]. The partial product addition of the lower order bits (approximate part) requires special addition method.

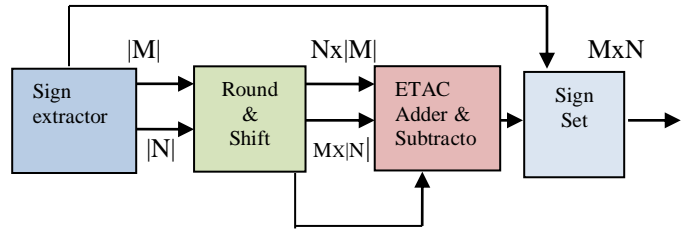


FIGURE 2. BLOCK DIAGRAM OF THE PROPOSED MULTIPLIER

3.1 Different blocks of the multiplier

3.1.1 Sign extractor

This block helps to change the sign of the input values by taking two's complement of the MSB bit. Then it gives output by reversing the input for negative values while positive ones remains constant.

3.1.2 Round/shift

This block helps to round off the inputs to the nearest value and the output are obtained in the form of $2n$. The rounded bit can be considered as one for the whole process.

3.1.3 ETAC adder

The idea of the ETAC is based on tolerating some error in the lower order bits while maintaining the accuracy in the higher order bits using the normal addition. In the inaccurate part, ETAC is performed from right to left. If any carry is generated in the MSB part of inaccurate part, it is shifted to the LSB of the accurate part

Accurate Part				Inaccurate Part			
		Carry in					
MSB	LSB	MSB	LSB	MSB	LSB	MSB	LSB
1	0	1	1	0	0	1	1
0	1	1	0	1	0	0	0
0	1	1	0	1	0	0	0
1	0	0	1	1	1	0	1
Normal Addition				ETAC Addition			

In the example, the two 15 bit operands, $A = 101100111110011$ (23027) and $B = 011010011011000$ (13528) are divided into two parts each for the accurate and inaccurate parts. If input bits are 0 or different (0 & 1) normal addition is performed. If both the inputs are 1, carry is

generated and the carry is shifted from MSB of the inaccurate part to the LSB of the accurate part. Thus, we can able to achieve better accuracy. By normal addition the addition of A & B yields $(A+B) = 36555$ and by using ETAC addition $(A+B) = 36603$. The accuracy (ACC) of the adder is $(1 - 48/36555) \times 100 = 99.87\%$.

By shifting the carry from inaccurate to the accurate part, overall delay time is greatly reduced and therefore the power consumption.

3.1.4 Sign set

This block applies to fix the sign of the output by takes two's complement of the given data. When the extracted sign for the two input values are different, the output of the Subtractor is reversed [10].

4. RESULT AND ANALYSIS

4.1. Hardware Implementation

4.1.1 Area

With shrinking system size, ASIC should be capable to accommodate an increased functionality in a low-area. The designer will specify area constraint and cadence tool is utilized to implement the area performance. The area is improved by having lesser number of cells and by replacing several cells with a Single cell that includes both functionalities.

4.1.2 Power

The implementation of hand-held systems has led to a decrease of battery-size due to low- power consuming systems. Low-power consumption has become a main requirement for a lot of designers.

Table III shows the comparison of the area, power for different technologies such as 180nm. In existing method, the normal approximate was employed to perform the multiplication operation, which occupied more area but the ETAC method required less area. Due to this ETAC adder, the area and power are minimized than the conventional method.

Table III. Comparison of ASIC performance for existing method and ETAC Method.

Technology	Method	Area (um ²)	Power (nW)
180nm	APROXIMATE MULTIPLIER	5994	963186.491
	ETAC	4146	874233.810

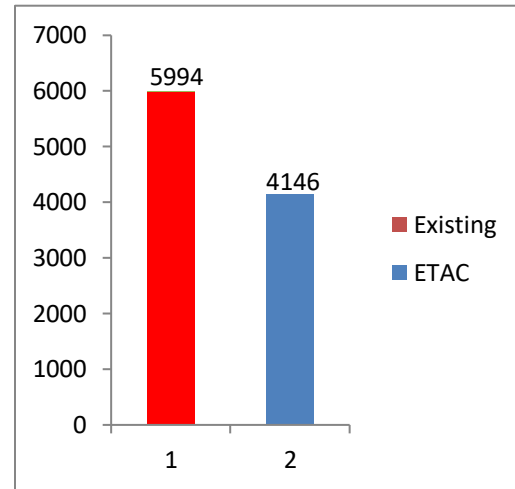


FIGURE 3. Comparison of the area performance for existing and ETAC method.

5. SIMULATION RESULT

The multipliers are simulated by Xilinx 14.7 ISE tool. The given input and the outputs obtained by the 8x8 Wallace multiplier and ETAC is shown in the table IV.

Table IV. Comparison between normal approximation method and ETAC method.

Parameters	Normal approximation Method	ETAC Method
Slices	78	66
FFs	64	64
LUTs	134	118
Gate Count	1956	1579
Power(mW)	197	152
Delay(ns)	8.664	10.047

Figure 4 shows the variation of the power consumption and delay with different size of the ETAC adder. It is observed that the delay time and

power consumption increases with increase in size of the bits. The result suggests that the power consumption reduces by 9.24 % for 8 bit multiplier and increases slightly with increase in size of the multiplier. Initially, delay time increases from 10% for 8 bit multiplier. As we increase the bit size further, the delay time increase slightly whereas for 64 bit multiplier, the delay time tends to reduce.

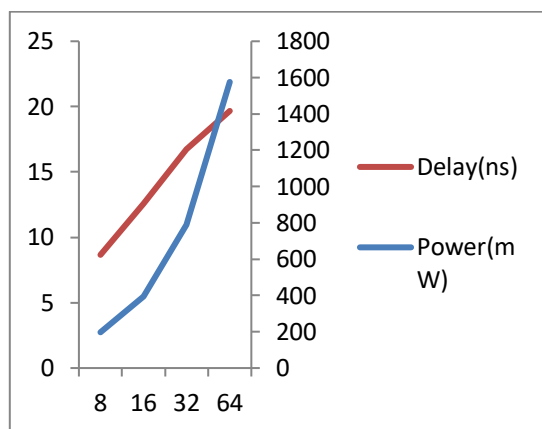


FIGURE 4. Variation of power consumption and delay with the size of ETAC multiplier.

6. CONCLUSION

In this paper, a novel Error Tolerant Approximate computing multiplier technique is developed which has low power consumption and device area as compared with traditional multipliers. To evaluate the performance of the multiplier, two adders are implemented. The first one is exact adder for accurate part and the OR adder for inaccurate part. Hence, the proposed multiplier provides proper balance between accuracy and the overall performance. This new multiplier delivers significant reduction in power consumption and area by 9.24% and 30.8%. As a conclusion, this new

multiplier can be suitable for Digital signal processing applications.

REFERENCES

1. V. Gupta, D. Mohapatra, A. Raghunathan, "Low power Digital signal processing using Approximate Adders" IEEE Tran's computer aided design of integrated circuits and systems, Vol.32, no.1, Jan.2013.
2. G. Zervakis, K. Tsoumanis, S. Xydis, D. Soudris, K. Pekmetzi, "Design-Efficient Approximate multiplication Circuits through Partial Product Perforation."
3. A. Chandrakala, A. Sreeramulu, L. Srinivas, "Design and Implementation of 4-2 Compressor Design with XOR-XNOR" International Advanced Research Journal in science, Engg and Tech Vol.3, Issue 7, Jul. 2016.
4. P. Apiarist, D. Koozehkanani, F. Nazari, "An Ultra High Speed Digital 4-2 Compressor in 65 nm CMOS" International Journal of Computer Theory and Engg, Vol. 5, no. 4, Aug. 2013.
5. R. Venkatesan, A. Agarwal, K. Roy, A. Raghunathan, "MACACO: Modeling and Analysis of Circuits for Approximate Computing".
6. N. Zhu, Wang L. Goh, W. Zhang, K. Seng Yeo, Z. H. 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 (vlsi) systems, Vol. 18, no. 8, Aug. 2010.
7. R. Sivaraman, R. Parameshwaran, "Minimization of Area and Power in Digital System Design for Digital Combinational Circuits", Indian Journal of Science and Technology, Vol. 9(29), Aug. 2016
8. S. Venkatachalam, S.B. Ko, "Design of Power and Area Efficient Approximate Multipliers", IEEE Trans on very large scale integration (vlsi) systems
9. T. Yang, T. Ukezono, T. Sato, "A Low power high speed accuracy controllable approximate multiplier design".
10. M. Ostaa, A. Ibrahima, M. Vallea, H. Chible, "Approximate multipliers based on inexact adders for energy efficient data processing", 2017.