International Journal of Science and Research (IJSR)
ISSN (Online): 2319-7064
Index Copernicus Value (2013): 6.14 | Impact Factor (2013): 4.438

Systematic Approach of Low Power Truncation-Error-Tolerant (TET) Adder

Khiali Pooja Sen[1], Vishal G. Puranik[2]

[1, 2] Savitribai Phule Pune University, Department of E &TC, JSPM's BSIOTR, Wagholi, Pune, Maharashtra, India

Abstract: ***In modern VLSI technology, the presence of all kinds of errors has become unavoidable. By adopting a rising concept in VLSI design and testing, error tolerance (ET), an error-tolerant-adder (ETA) is implemented. The ETA is able to ease the strict restriction on accuracy, and at the same time attain enormous improvements in both the power consumption and speed execution. We can compare it to its ceremonious counterparts, the implemented ETA is able to achieve more than 65% improvement in the Power-Delay-Product (PDP). There is one advantage of the implemented ETA is that it can tolerate certain amount of errors. This paper compares the performance of the ETA in terms of accuracy, delay and power consumption with that of conventional adders.***

Keywords: VLSI, ET, ETA, PDP.

1. Introduction

The addition of two binary numbers is the most fundamental and widely used arithmetic operation. This operation is used in microprocessors, digital signal processors, data processing application specific integrated circuits and many more. There are many adders designed till now. ETA is one such efficient adder which speeds up binary addition. ETA is the Error Tolerant Adder which consumes less power and delay.

In conventional digital VLSI design, we can assume that a usable circuit or system should always provide definite and accurate results. But these perfect operations are seldom needed in our non digital worldly experiences. Now the existing world accepts "analog computation," which gives "good enough" results rather than completely accurate results. The data treated by many digital systems can already hold errors. In some applications, such as, a communication system, the signal (analog) coming from the outside world must first be sampled before being converted to digital data. Then the digital data processed and transmitted in a noisy channel before converting back to an analog Signal. While in this process, errors can occur anywhere. Based on the characteristic of digital VLSI design, some new concepts and design techniques have been suggested. It includes the concept of error tolerance (ET) and the PCMOS technology. The definition says, a circuit is error tolerant if:

1) It comprises defects that cause internal and external errors and
2) The system that comprises this circuit produces acceptable results. Also, the "imperfect" property looks to be not appealing. The need for the error-tolerant circuit was foretold in the 2003 International Technology Roadmap for Semiconductors (ITRS). Therefore, to deal with error-tolerant problems, some of the truncated adders/multipliers have been reported, but are not able to perform well in its area, speed, power, or accuracy. Naturally, not all digital systems can occupy the error-tolerant concept. Now a day in digital systems such as control systems, the precision of the output signal is extremely important, and this refuses the use of the error tolerant circuit. For many digital signal processing (DSP) systems that process signals relating to human senses such as sight, smell, hearing, and touch, for e.g. we can take, the image processes and speech processing systems, where the error-tolerant circuits may be applicable.

2. Literature Survey

Conventional Adders

Ripple-Carry Adder (RCA): The n-bit adder built from n one-bit full adders is known as a ripple carry adder, because of the way the carry is computed. Each full adder inputs a C_{in}, which is the Cout of the previous adder. This kind of adder is a ripple carry adder, since each carry bit "ripples" to the next full adder. Block diagram of Ripple Carry Adder is as in Fig. 1.

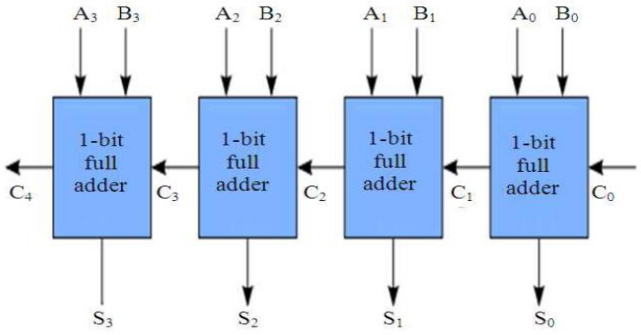

Figure 1: Block diagram of 4 Bit Ripple Carry

The layout of ripple carry adder is simple, which allows for fast design time; however, the ripple carry adder is relatively slow, since each full adder must wait for the carry bit to be calculated from the previous full adder. The gate delay can easily be calculated by inspection of the full adder circuit. Each full adder requires three levels of logic. In a 32-bit (ripple carry) adder, there are 32 full adders, so the critical path (worst case) delay is 31 * 2(for carry propagation) + 3(for sum) = 65 gate delays. Table 1 and 2 shows the result obtained for RCA.

Carry-look-ahead adder CLA: Carry look ahead logic uses the concepts of generating and propagating carries. The addition of two 1-digit inputs A and B is said to generate if

the addition will always carry, regardless of whether there is an input carry. In the case of binary addition, A+B generates if and only if both A and B are 1. The addition of two 1-digit inputs A and B is said to propagate if the addition will carry whenever there is an input carry. The propagate and generate are defined with respect to a single digit of addition and do not depend on any other digits in the sum. In the case of binary addition, A+B propagates if and only if at least one of A or B is 1. Sometimes a slightly different definition of propagate is used. By this definition, A+B is said to propagate if the addition will carry whenever there is an input carry, but will not carry if there is no input carry. For binary arithmetic, or is faster than xor and takes fewer transistors to implement. However, for a multiple-level carry look ahead adder, it is simpler to use. Block Diagram of 4bit carry-look-ahead adder is as in Fig. 2. The carry look ahead adder represents the most widely used design for high-speed adders in modern Computers. The advantage of using a look-ahead design over a ripple carry adder is that the Look-ahead is faster in computing the solution. The carry-in values in a carry look-ahead design are calculated independent of each other through a series of logic circuits.

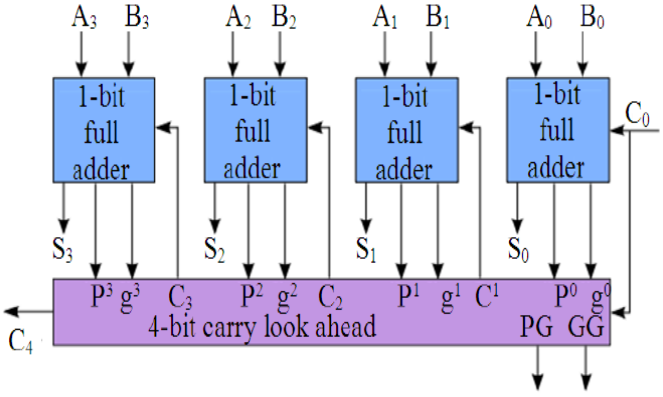

Figure 2: Block diagram of 4bit carry-look-ahead adder

Carry look ahead depends on two things:

- Calculating, for each digit position, whether that position is going to propagate a carry if one comes in from the right.
- Combining these calculated values so as to be able to deduce quickly whether, for each group of digits, that group is going to propagate a carry that comes in from the right.
- Supposing that groups of 4 digits are chosen.

Then the sequence of events goes something like this:

- All 1-bit adders calculate their results. Simultaneously, the look ahead units perform their calculations.
- Suppose that a carry arises in a particular group. Within at most 3 gate delays, that carry will emerge at the left-hand end of the group and start propagating through the group to its left.
- If that carry is going to propagate all the way through the next group, the look ahead unit will already have deduced this. Accordingly, before the carry emerges from the next group the look ahead unit is immediately (within 1 gate delay) able to tell the next group to the left that it is going to receive a carry- and, at the same time, to tell the next look ahead unit to the left that a carry is on its way Table 1 and 2 shows the result obtained for CLA.

3. ETA – Error-Tolerant Adder

Before giving description about the ETA, the definitions of some normally used terminologies shown in this paper are given as follows:-

a) **OE: (Overall error):-** OE = | Rc – Re |, Here Re is the result obtained by the adder, and Rc denotes the correct result.
b) **Accuracy (ACC):** It is defined as: ACC = (1-(OE/Rc)) × 100%. Its value ranges from 0% to 100%.
c) **Minimum acceptable accuracy (MAA):** Minimum acceptable accuracy is just that threshold value. Hence, the result we have obtained, whose accuracy is higher than the minimum acceptable accuracy is called acceptable result.
d) **Acceptance probability (AP):** Acceptance probability is the probability that the accuracy of an adder is higher than the minimum acceptable accuracy. It can be expressed as AP = P (ACC > MAA), where its value ranging from 0 to 1.

4. Arithmetic Addition

In a conventional circuit like adder, the delay is mainly assigned to the carry propagation chain along the critical path, from the least significant bit (LSB) to the most significant bit (MSB). A significant proportion of the power consumption of an adder is due to the glitches that are caused by the carry propagation. Hence, if the carry propagation can be eliminated or restricted, a great improvement in speed and power consumption can be attained. In this paper, we propose, an innovative and new addition arithmetic that can attain great saving in speed and power consumption. The new type of arithmetic addition can be illustrated in Fig. 3 via an example. First of all we split the input operands into two parts: an accurate part that includes several higher order bits and the inaccurate part that is made up of the remaining lower order bits. It is not necessary that the length of each part to be equal. The addition of each part in this process starts from the middle (joining point of the two parts) toward the two opposite directions at the same time. From the figure we can take the example, it has the two 16-bit input operands, A = “1011001110011010” (45978) and B = “0110100100010011” (26899), are bifurcated or split up equally into 8 bits each for the accurate and inaccurate parts.

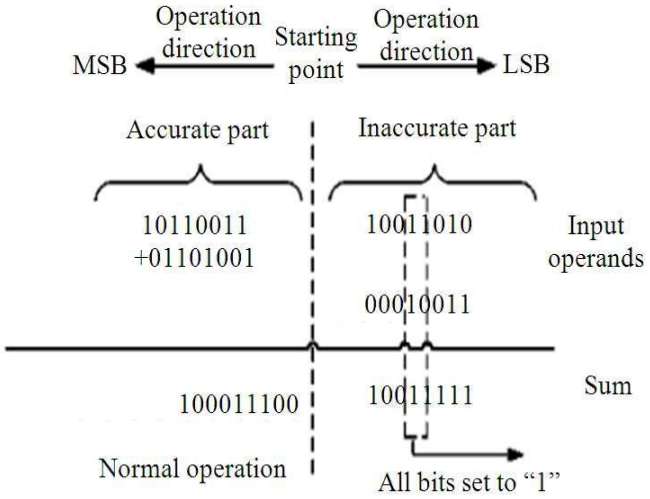

Figure 3: Arithmetic addition

5. Requirement for Error-Tolerant-Adder (ETA)

Increasingly huge data sets and the need for instant response need the adder to be large and fast. The traditional adder like ripple-carry adder (RCA) is thus no longer suitable for large adders because of its low-speed execution. There are different types of adders, like CSK (carry-skip adder), CSL (carry-select adder), and CLA (carry-look-ahead adder), are developed. Likewise, there are so many low-power adder design techniques that have been proposed. Even so, there are always trade-offs between speed and power. For this problem the error-tolerant design can be a possible solution. By sacrificing some accuracy, the error tolerance adder can achieve great improvement in both the power consumption and speed performance.

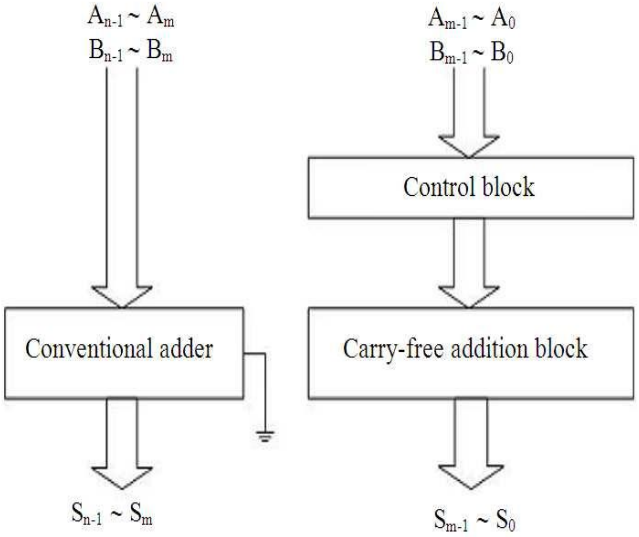

Figure 4: Hardware implementation of the ETA

6. Result

Comparing the simulation results of our proposed ETA with those of the conventional adders (see Table I), it is evident that the ETA performed the best in terms of power consumption, delay, and Power-Delay Product (PDP). The PDP of the ETA is noted to be 66.29%, 77.44%, 83.70%, and 75.21% better than the RCA, CSK, CSL, and CLA, respectively. As for transistor count, the proposed ETA is almost as good as the RCA.

Table 1: Simulation Result for ETA versus Conventional Adders

Type of Adder	Power (mW)	Delay (ns)	PDP (pJ)	PDP saving (%)	Transistor Count
RCA	0.22	4.04	0.89	66.29	896
CSK	0.46	2.90	1.33	77.44	1728
CSL	0.60	3.06	1.84	83.70	2176
CLA	0.51	2.37	1.21	75.21	2208
ETA	0.13	2.29	0.30	N.A.	1006

7. Conclusion

In this paper, error tolerance is introduced in VLSI design. A new type of adder, the error-tolerant adder, which have certain amount of accuracy for significant power saving and performance improvement, is proposed. The possible applications of the ETA fall mainly in areas where there is no strict requirement on accuracy or where super low power consumption and high-speed performance are more important than accuracy. We can take an example, in the DSP application for portable devices such as cell phones and laptops.

8. Acknowledgement

I would like to express my sincere gratitude to my guide Prof. Vishal G. Puranik for his continuous support, patience, motivation, enthusiasm, and immense knowledge. His guidance helped me in all the time of research and writing of this paper.

References

[1] Ning Zhu, Wang Ling Goh, Zhi Hui Kong, Weija zhang, Kiat Seng Yeo, "Design of Low-Power High-Speed Truncation Error Tolerant Adder and its Applications in Digital Signal Processing," august 2010, IEEE transactions on VLSI (very large scale integration) vol. 18, no. 8,.

[2] K. V. Palem, "Energy aware computing through probabilistic switching: A study of limits," September 2005, IEEE Transaction Computer, vol. 54, no. 9, pp. 1123–1137.

[3] M. A. Breuer, T.M. Mak and S.K. Gupta, "Design and error-tolerance in the presence of massive numbers of defects," May-Jun. 2004, IEEE Des. Test Computer, vol. 24, no. 3, pp. 216–227.

[4] International Technology Roadmap for Semiconductors [Online]. Available: http://public.itrs.net/

[5] Rajesh Mehra and Ruchika Sharma, "Design of Error-Tolerant CMOS Adder Using optimized Transistor Count," IJARCCE (International Journal of Advanced Research in Computer and Communication Engineering), July 2014, Volume 3, Issue 7, ISSN (Online) : 2278-1021 ISSN (Print) : 2319-5940.

[6] A. B. Melvin and Z. Haiyang, "Error-tolerance and multi-media," in Proc. 2006 Int. Conf. Intell. Inf. Hiding and Multimedia Signal Process, 2006, pp. 521–524.

Author Profile

Khiali Pooja Sen received the Bachelors degree (B.E.) Computer Engineering in 2012 from SSIPMT, Raipur, Chhattisgarh. She is now pursuing Masters Degree E & TC (VLSI and Embedded System), from JSPM's BSIOTR, Wagholi, Pune, Maharashtra.

Prof. Vishal G. Puranik received his M.E(Electronics) in 2007 from Aurangabad University, and now he is working as Assistant professor, Department of E & TC, JSPM's Bhivarabai Sawant Institute of Technology & Research, Wagholi, Pune, Maharashtra.. His research is Embedded Systems.