

# A LOW POWER HIGH PERFORMANCE APPROXIMATE 16-BIT MULTIPLIER DESIGN

K.Suresh Chowdary<sup>1</sup>, K. Ram Puneeth<sup>2</sup>, K. Sai Pavan<sup>3</sup>, B. Sai Vishal<sup>4</sup>, Asst. Prof. CH. Srigiri<sup>5</sup>,

*Department of Electronics and Communication Engineering, Godavari Institute of Engineering and Technology (An Autonomous Institution) (NBA Accredited & NAAC A+) (Approved by AICTE, Affiliated to JNTUK, Kakinada)NH-16, Chaitanya Knowledge City, Rajahmundry-533294, Andhra Pradesh, India.*

\*\*\*

**Abstract** - Multiplication is a key fundamental function for many error-tolerant applications. Approximate multiplication is considered to be an efficient technique for trading off energy against performance and accuracy. This paper proposes an accuracy-controllable multiplier whose final product is generated by a carry-maskable adder. The proposed scheme can dynamically select the length of the carry propagation to satisfy the accuracy requirements flexibly. The partial product tree of the multiplier is approximated by the proposed tree compressor. An  $8 \times 8$  multiplier design is implemented by employing the carry-maskable adder and the compressor. Compared with a conventional Wallace tree multiplier, the proposed multiplier reduced power consumption by between 47.3% and 56.2% and critical path delay by between 29.9% and 60.5%, depending on the required accuracy. Its silicon area was also 44.6% smaller. In addition, results from an image processing application demonstrate that the quality of the processed images can be controlled by the proposed multiplier design.

**Keywords:** Approximate Computing, Carry Maskable Adder, Multiplier.

## 1. INTRODUCTION

Many increasingly popular applications, such as image processing and recognition, are inherently tolerant of small inaccuracies. These applications are computationally demanding and multiplication is their fundamental arithmetic function, which creates an opportunity to trade off computational accuracy for reduced power consumption. Approximate computing is an efficient approach for error-tolerant applications because it can trade off accuracy for power, and it currently plays an important role in such application domains [1].

Different error-tolerant applications have different accuracy requirements, as do different program phases in an application. If multiplication accuracy is fixed, power will be wasted when high accuracy is not required. This means that approximate multipliers should be dynamically reconfigurable to match the

different accuracy requirements of different program phases and applications.

This paper focuses on an approximate multiplier design that can control accuracy dynamically. A carry-maskable adder (CMA) is proposed that can be dynamically configured to function as a conventional carry propagation adder (CPA), a set of bit-parallel OR gates, or a combination of the two. This configurability is realized by masking carry propagation: the CPA in the last stage of the multiplier is replaced by the proposed CMA. An approximate tree compressor is utilized to reduce the accumulation layer depth of the partial product tree.

Our approach introduces a term representing the power and accuracy requirements which simplifies the partial product reduction (PPR) component as needed. An approximate multiplier is designed using the proposed adder and compressor. This multiplier, together with a conventional multiplier and the previously studied approximate multipliers, was implemented in Verilog HDL using a 45-nm library to evaluate the power consumption, critical path delay, and design area. Compared with the conventional Wallace tree multiplier, the proposed approximate multiplier reduced power consumption by between 47.3% and 56.2% and the critical path delay by between 29.9% and 60.5%, depending on the required computational accuracy. In addition, its design area was 44.6% smaller. Comparisons with the established approximate multipliers, none of which have any dynamic reconfigurability, demonstrate that the proposed multiplier provided the best trade-off of power and delay against accuracy. All the multiplier designs are then evaluated in a real image processing application.

## 1.1 OVERVIEW OF MULTIPLIER

A multiplier appreciably affects the velocity and power intake in a processor. Exact effects are not constantly required in several calculations, for instance, those for order and acknowledgment in statistics processing and Digital Signal Processing (DSP). In this way, multiplier systems are focused on rapid, low location and coffee energy. These parameters are achieved by inexact multipliers. By and big, inexact processing has a crucial consideration as a rising

gadget to control usage. In this paper some other sixteen bit approximate multiplier is designed. The in specific results of the multiplier are adjusted to provide fluctuating likelihood terms. Rationale intricacy of estimate is shifted for the gathering of modified fractional gadgets dependent on their chance.

Multiplication is a mathematical operation that performs addition of an integer to itself through multiple instances. A number (multiplicand) is brought by wide variety of times as followed by means of any other quantity (multiplier) to form a end result (product). Multipliers play an major position in today's digital sign processing and various other programs. Multiplier design ought to provide high pace and coffee strength consumption. Multiplication entails specially three steps.

- Partial product generation
- Partial product reduction
- Final addition

## 2. BLOCK DIAGRAM

Wallace tree multiplier is fast multiplier compared to the available multiplier as they are used carry save addition algorithm for the final product addition. In multipliers if there is a small increase in speed will improve the operating frequency of a digital signal processor. Hence many attempts are done on multipliers to make it faster. When designing a multiplier, huge amount power and delay are generated, to minimize that, adders and compressor are used. The proposed Wallace tree multiplier uses higher order compressors such as 3-2, 5-3 and 7-3 compressors to reduce delay and to achieve high-speed. The higher order compressors are developed by merging binary counter property with compressor property.

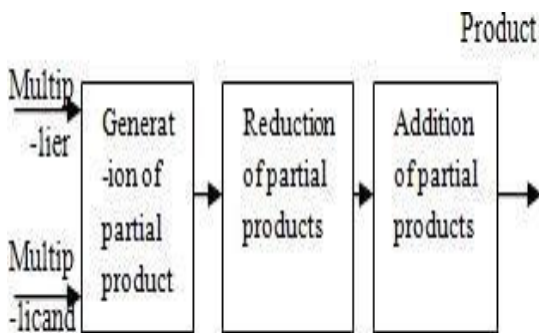


Fig.1 Block diagram of wallace tree Multiplier.

In multiplier design if speed is not a problem the design complexity can be reduced by adding partial

products serially. For high speed multipliers, consider 16 bit partial products are generally added using Wallace tree method. In this approach generated partial products are compressed at each column into two or more bits.

## 3. MULTIPLICATION PROCESS

The diagram showing the multiplication process is shown in figure.

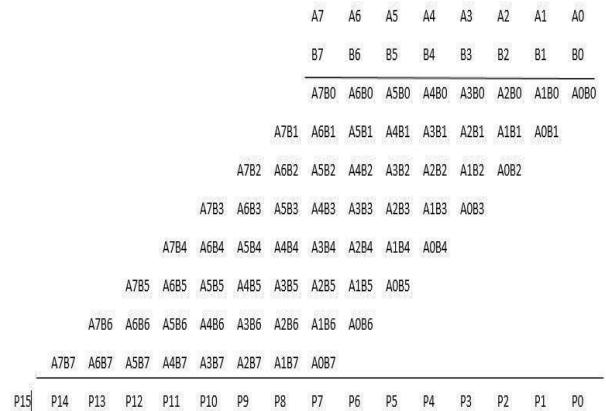


Fig.2 Multiplication Process.

Multiplication is done using two operands one called as multiplicand and the other as multiplier. Each bit of multiplicand is multiplied with single bit of multiplier and it represents the final product. To arrive at the final product, partial product addition is performed. The bits of partial products are fed to adders column wise in order to obtain final product.

## 4. COMPRESSOR DESIGN

The 4-2 Compressor layout has 5 inputs A, B, C, D and Cin to get three outputs as Sum, Carry and Carry out. The four inputs A, B, C and D and the output Sum are having the same weight. The input Cin is the output from a previous least enormous bit compressor and the Cout output is for the compressor within the subsequent level.

Cin	X4	X3	X2	X1	Cout	Carry	sum
0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	1
0	0	0	1	0	0	0	1
0	0	0	1	1	1	0	0
0	0	1	0	0	0	0	1
0	0	1	0	1	1	0	0
0	0	1	1	0	1	0	0
0	0	1	1	1	1	0	1
0	1	0	0	0	0	0	1
0	1	0	0	1	0	1	0
0	1	0	1	0	0	1	0
0	1	0	1	1	1	0	1
0	1	1	0	0	0	1	0
0	1	1	0	1	1	0	1
0	1	1	1	0	1	0	1
0	1	1	1	1	1	1	0
1	0	0	0	0	0	0	1
1	0	0	0	1	0	1	0
1	0	0	1	0	0	1	0
1	0	0	1	1	1	0	1
1	0	1	0	0	0	1	0
1	0	1	0	1	1	1	0
1	0	1	1	0	0	1	0
1	0	1	1	1	1	1	0
1	1	0	0	0	0	1	0
1	1	0	0	1	0	1	1
1	1	0	1	0	0	1	1
1	1	0	1	1	1	1	0
1	1	1	0	0	0	1	1
1	1	1	0	1	1	1	0
1	1	1	1	0	1	1	0
1	1	1	1	1	1	1	1

Truth Table of 4-2 Compressor

The fundamental objective multi-operand bring-shop addition or parallel addition is to lessen the multi operands to two numbers consequently n-2 compressors are widely used in laptop mathematics. A n-2 blower is generally a streamlined circuit that diminishes n numbers to two numbers whilst legitimately recreated. The deliver bit from the state of affairs to the privilege is indicated as cin even as the deliver bit into the higher role is meant as cout. The essential shape of a four-2 blower is practiced with the aid of utilizing full-snake (FA) cells. The adjusted plan incorporates of 3 XOR-XNOR entryways, one XOR and 2-1 multiplexers are utilized.

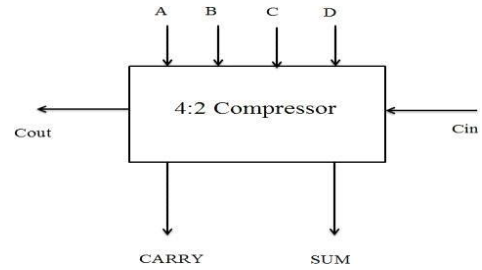


Fig.3 Adder Compressor.

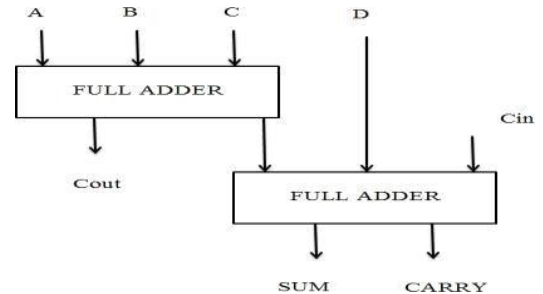


Fig.4 Adder Compressor Implemented with Full Adders.

## 5. PROPOSED ACCURACY-CONTROLLABLE MULTIPLIER

A typical multiplier consists of three parts: (i) Partial product generation using an AND gate; (ii) PPR using an adder tree; and (iii) Addition to produce the final result using a CPA. Power consumption and circuit complexity are dominated by the PPR, and the multiplier's critical path is dominated by the propagated carry chain in the CPA.

This section is organized as follows. Section III-A explains how the partial product layer is simplified by the approximate tree compressor. Section III-B introduces the CMA. Finally, Section III-C presents the overall structure of the accuracy- controllable approximate multiplier, which uses the proposed adder and tree compressor.

### 5.1 Approximate Tree Compressor

Figure 1(a) shows an accurate half adder, for which the following equation can be obtained:

$$\{c, s\} = a + b = 2c + s = (c + s) + c,$$

where {,} and + denote concatenation and addition, respectively. The value c is generated by a AND b and s is generated by a XOR b, so (c+

s) can be generated by a OR b. Based on the above, consider the basic logic cell shown in Fig. 1(b), for which the following equations can be obtained:

$$p = c + s, q = c,$$

$$\{c, s\} = a + b = p + q.$$

This is called an incomplete adder cell (iCAC). Table I shows the truth tables for an accurate half adder and an iCAC. Note that the bit position of c and that of s, p, and q are different. As can be seen, q is equal to c. While p is not equal to s, the precise sum can be obtained by adding p and q, so the iCAC is not an approximate adder but an element of a precise adder.

By extending the above equation to N bits, the following equation can be obtained:

$$S = A + B = P + Q.$$

where A, B, P, and Q are N-bit values, the bits of which correspond to a, b, p, and q, respectively. A row of eight iCACs, used for 8-bit inputs, is shown in Fig. 2.

Consider the example of an 8-bit adder with the two inputs A = 01011111 and B = 00110110. The accurate sum S is 10010101, while the row of iCACs produces P = 01111111 and Q =

00010110. Again, it is evident that the following holds:

$$S = P + Q.$$

While S is obtained from P and Q, P can be used as an approximation for S, and Q can be used as an error recovery vector for the approximate sum P.

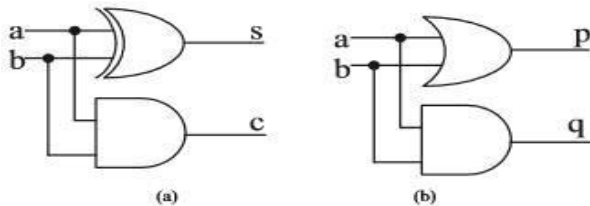


Fig.5 (a) Accurate half adder and (b) incomplete adder cell.

Inputs		Outputs			
		Accurate half adder		iCAC	
a	b	c	s	q	p
0	0	0	0	0	0
0	1	0	1	0	1
1	0	0	1	0	1
1	1	1	0	1	1

Truth Tables for Accurate Half Adder and Incomplete Adder Cell.

### 5.2 Carry Maskable Adder

A CMA is proposed to control the accuracy flexibly and dynamically. A K-bit CMA comprises (K-1) carry-maskable full adders and one carry-maskable half adder, and its structure is similar to that of a K-bit CPA.

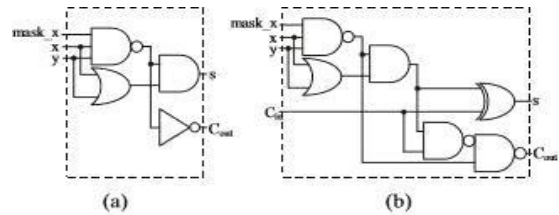


Fig.6 (a) Carry-maskable half adder, (b) Carry-maskable full adder.

The structures of the proposed carry-maskable half and full adders are shown in Fig. 4. In the proposed half adder, when mask\_x is 0, S is equal to x OR y and Cout is equal to 0. Otherwise, when mask\_x is 1, S is equal to x XOR y and Cout is equal to x AND y. In other words, the operation of the proposed half adder can be controlled by the active-low signal mask\_x. When mask\_x is disabled (=1), it functions as an accurate half adder, and when mask\_x is enabled (=0), Cout is masked to 0 and it functions as an OR gate with output S. The operation of the proposed full adder is similar to the half adder: when mask\_x is disabled (=1), it functions as an accurate full adder, and when mask\_x is enabled (=0), Cout is equal to Cin and S is the output of an OR gate.

Multiplication of two 16-bit numbers. The numbers are denoted by A and B where represents the bits of multiplicand A with as its least significant bit and B as its most significant bit. The product of the two 16-bit numbers is denoted by P which is of 32-bit with as the least significant bit and as the most significant bit. Fig shows the basic multiplication of two numbers and thus producing the result, Now the use of half adders and full adders is explained in next Fig(1).

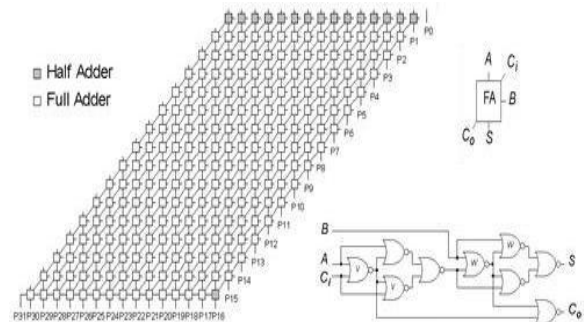


Fig.7 Structure of 16-bit wallace tree Multiplier.

### 6. RESULTS AND DISCUSSION

The 16-bit Approximate Multiplier is designed via the usage of Verilog HDL language and synthesized the usage of XILINX tool. From the Xilinx, we get vicinity, put off, and dynamic and static power intake. The generated partial products are generated and compressed by means of the use approximate half adder, full adder, and carry maskable adder designs to produce the final result. The proposed multiplier achieves the better performance as compared with the previous approximate multipliers.

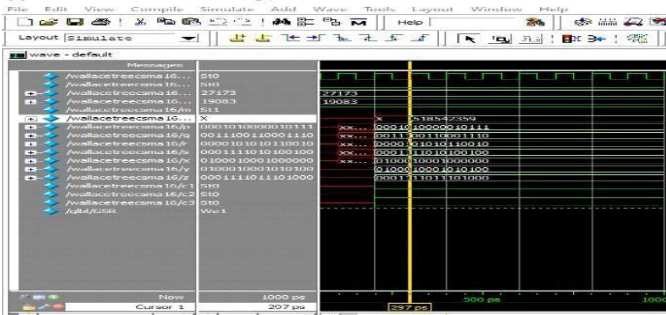


Fig.8 Simulation Results for 16-bit Wallace Input Combination.

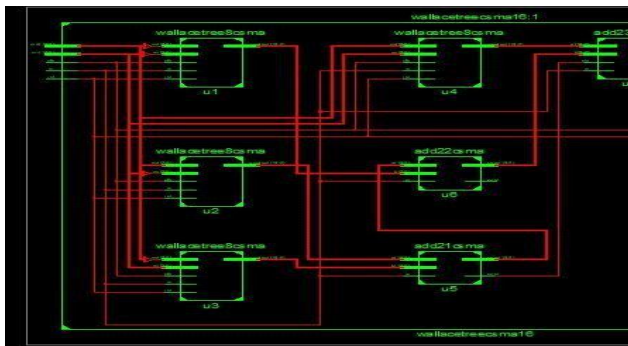


Fig.9 RTL Schematic of the 16 bit Approximate Multiplier.

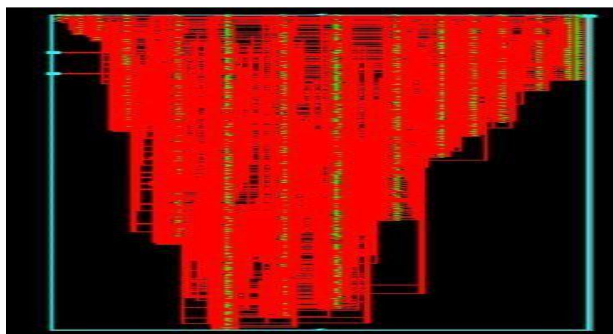


Fig.10 Technology Schematic of the 16 bit Approximate Multiplier.

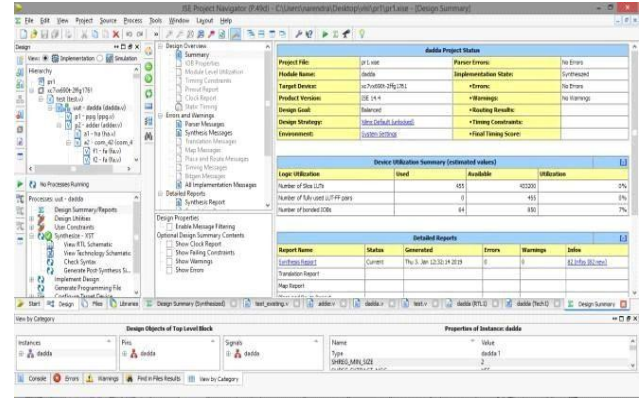


Fig.11 Summary Report of the 16 bit Approximate Multiplier.

### 7. CONCLUSION AND FUTURE WORK

Hence a 16-bit Wallace multiplier using a compression technique for adding the generated the partial products by using carry maskable adders and the compressor shows a less delay and Area. An accuracy-controllable approximate multiplier has been proposed in this paper that consumes less power and has a shorter critical path delay than the conventional design. Its dynamic controllability is realized by the proposed CMA. The multiplier was evaluated at both the circuit and application levels. The experimental results demonstrate that the proposed multiplier was able to deliver significant power savings and speedups while maintaining a significantly smaller circuit area than that of the conventional Wallace tree multiplier. Further more, for the same accuracy, the proposed multiplier delivered greater improvements in both power consumption and critical path delay than other previously studied approximate multipliers. Finally, the ability of our proposed multiplier to control accuracy was confirmed by an application- level evaluation.

As an attempt to develop arithmetic algorithm and architecture level optimization techniques for low power multiplier design, the research presented in this dissertation has achieved good results and demonstrated the efficiency of high level optimization techniques. However, there are limitations in our work and several future research directions are possible.

### REFERENCES

- Swami Bharati Krishna Tirthaji Maharaja, "Vedic Mathematics", MotilalBanarsidass Publishers, 1965.
- Rakshith T R and Rakshith Saligram, "Design of High-Speed Low Power Multiplier using Reversible logic: a Vedic Mathematical Approach", International Conference on Circuits, Power and Computing Technologies (ICCPCT-2013), ISBN: 978-1-4673-4922-2/13, pp.775-781.

- M.E. Paramasivam and Dr R.S. Sabeenian, "An Efficient Bit Reduction Binary Multiplication Algorithm using Vedic Methods", IEEE 2nd International Advance Computing Conference, 2010, ISBN: 978-1-4244-4791-6/10, pp. 25-28.
- Sushma R. Huddar, Sudhir Rao Rupanagudi, Kalpana M and Surabhi Mohan, "Novel High Speed Vedic Mathematics Multiplier using Compressors", International Multi conference on Automation, Computing, Communication, Control and Compressed Sensing(iMac4s), 22-23 March 2013, Kottayam, ISBN: 978-1-4673-5090-7/13, pp.465-469.
- L. Sriraman and T. N. Prabakar, "Design and Implementation of Two Variables Multiplier Using KCM and Vedic Mathematics", 1st International Conference on Recent Advances in Information Technology (RAIT -2012), ISBN: 978- 1-4577-0697-4/12.
- Prabir Saha, Arindam Banerjee, Partha Bhattacharyya and Anup Dandapat, "High Speed ASIC Design of Complex Multiplier Using Vedic Mathematics", Proceeding of the 2011 IEEE Students' Technology Symposium 14-16 January,2011, IIT Kharagpur, ISBN: 978-1-4244- 8943-5/11, pp.237-241.
- Soma BhanuTej, "Vedic Algorithms to develop green chips for future", International Journal of Systems, Algorithms & Applications, Volume 2, Issue ICAEM12, February 2012, ISSN Online: 2277-2677.
- Gaurav Sharma, Arjun Singh Chauhan, Himanshu Joshi and Satish Kumar Alaria, "Delay Comparison of 4 by 4 Vedic Multiplier based on Different Adder Architectures using VHDL", International Journal of IT, Engineering and Applied Sciences Research (IJIEASR), ISSN: 2319-4413, Volume 2, No. 6, June 2013, pp. 28-32.
- Aniruddha Kanhe, Shishir Kumar Dasand Ankit Kumar Singh, "Design and Implementation of Low Power Multiplier using Vedic Multiplication Technique", International Journal of Computer Science and Communication, Vol. 3, No. 1, June 2012, pp. 131-132.
- Anju and V.K. Agrawal, "FPGA Implementation of Low Power and High-Speed Vedic Multiplier using Vedic Mathematics", IOSR Journal of VLSI and Signal Processing (IOSR-JVSP) Volume 2, Issue 5 Jun. 2013, ISSN: 2319 - 4200, pp. 51-57.
- A Low-Power High-Speed Accuracy- Controllable Approximate Multiplier Design Tongxin Yang<sup>1</sup> Tomoaki Ukezono<sup>2</sup> Toshinori Sato<sup>3</sup>  
<sup>1</sup> Graduate School of Information and Control Systems, Fukuoka University, Japan  
<sup>2,3</sup>Department of Electronics Engineering and Computer Science, Fukuoka University, Japan 1.
- S. Venkataramani, V. K. Chippa, S. T. Chakradhar, K. Roy, and A. Raghunathan. "Quality programmable vector processors for approximate computing," 46th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), pp. 1-12, Dec. 2013.
- 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 Transactions on Circuits and Systems I: Regular Papers, vol. 57, no. 4, pp. 850-862, Apr. 2010.
- C. Liu, J. Han, and F. Lombardi, "A Low- Power, High-Performance approximate multiplier with configurable partial error recovery," Design, Automation & Test in Europe Conference & Exhibition (DATE), Mar. 2014.