Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 4, Issue. 6, June 2015, pg.49 – 56

RESEARCH ARTICLE

High Speed and Energy Efficient Approximate Adder for DSP Application

Sumant Mukherjee[1], Dushyant Kumar Soni[2]

[1]Dr. C.V.Raman University, Department of Engineering Kota, Bilaspur,(C.G.)

[2]Dr. C.V.Raman University, Department of Engineering Kota, Bilaspur,(C.G.)

[1] sjsumant@gmail.com, [2] dushyantsoni.soni@gmail.com

Abstract: In this paper a new high speed Approximate Adder will be introduced, which will reduce the hardware complexity and make justice with SPAA metrics. Adders are one of the key components in arithmetic circuits. Approximation can increase performance or reduce power consumption with a simplified or inaccurate circuit in application contexts where strict requirements are relaxed. For applications related to human senses, approximate arithmetic can be used to generate sufficient results rather than absolutely accurate results. Approximate design exploits a tradeoff of accuracy in computation versus performance and power.

Keywords:- Approximate half Adder(AHA), Approximate full adder(AFA), MAC unit, SPAA(Speed, Power, Area, Accuracy, LUT(Look up Table)

I. Introduction

The addition of two binary numbers is the fundamental and most often used arithmetic operation in microprocessors. In nearly all digital IC designs today, the addition operation is one of the most essential and frequent operations. Instruction sets for DSP's and general purpose processors include at least one type of addition. Other instructions such as subtraction and multiplication employ addition in their operations, and their underlying hardware is similar if not identical to addition hardware. Often, an adder or multiple adders will be in the critical path of the design, hence the performance of a design will be often be limited by the performance of its adders. When looking at other attributes of a chip, such as area or power, the designer will find that the hardware for addition will be a large contributor to these areas. It is therefore beneficial to choose the correct adder to implement in a design because of the many factors it affects in the overall chip. The demand for high speed processing has been increasing as a result of expanding computer and signal processing applications. Higher throughput arithmetic operations are important to achieve the desired performance in many real-time signal and image processing applications.

Digital signal Processing (DSP) is finding its way into more applications [19], and its popularity has materialized into a number of commercial processors [18]. Digital signal processors have different architectures and features than general purpose processors, and the performance gains of these features largely determine the performance of the whole processor. Basic operation found in MAC is the binary addition. Besides of the simple addition of two numbers, addition forms the basis for many processing operations, from counting to multiplication to filtering. But also simpler operations like incrimination and magnitude comparison based on binary addition. Therefore, binary addition is the most important arithmetic operation.

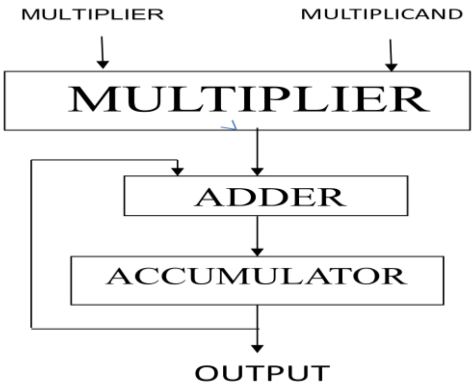


Figure 1 MAC unit

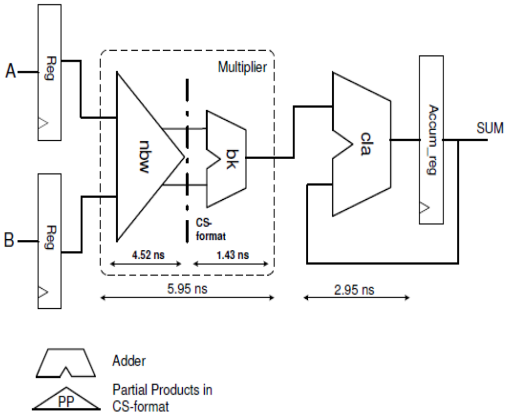


Figure 2 The Benchmark MAC unit

It comparison based on binary addition. Therefore, binary addition is the most important arithmetic operation. It is also a very critical one if implemented in hardware because it involves an expensive carry-propagation step, the evaluation time of which is dependent on the operand word length.

II. Literature Review

In this part we begin with the basic building blocks used for addition, then go through different algorithms.

A. Basic Adder blocks

1. Half Adder

The Half Adder (HA) is a combinational circuit with two binary input and two binary outputs such as sum and carryout. The equation (1) and (2) are the Boolean equations for sum and carryout, respectively.

$$\text{sum} = a \text{ xor } b \qquad (1)$$

$$\text{carryout} = a \text{ and } b \qquad (2)$$

2. Full Adder

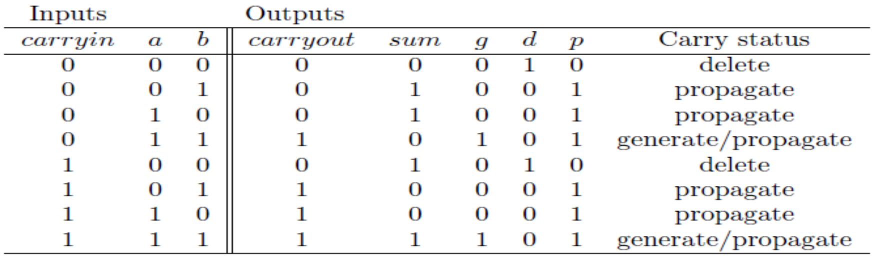
The Full Adder (FA) is a combinational circuit that adds two bits and a carry and outputs a sum bit and a carry bit. Equation (3) , (4) and (5) are the Boolean equations for the full adder sum and full adder carryout, respectively. In both those equations cin means carryin.

$$\text{sum} = \text{a xor b xor cin} \tag{3}$$
$$\text{carryout} = \text{a and b} + \text{b and cin} + \text{a and cin} \tag{4}$$
$$\text{cin} = \text{a and b} + (\text{a} + \text{b})\text{and cin} \tag{5}$$

From the above equations we see that sum and carryout is depends on carryin.

3. Partial Full Adder

The Partial Full Adder (PFA) is a structure that implements intermediate signals that can be used in the calculation of the carry bit. Such as delete, propagate and generate.



Inputs			Outputs					
carryin	*a*	*b*	*carryout*	*sum*	*g*	*d*	*p*	Carry status
0	0	0	0	0	0	1	0	delete
0	0	1	0	1	0	0	1	propagate
0	1	0	0	1	0	0	1	propagate
0	1	1	1	0	1	0	1	generate/propagate
1	0	0	0	1	0	1	0	delete
1	0	1	1	0	0	0	1	propagate
1	1	0	1	0	0	0	1	propagate
1	1	1	1	1	1	0	1	generate/propagate

Table 1: Extended Truth Table for a 1-bit adder

$$\text{generate(g)} = \text{a and b} \tag{6}$$
$$\text{delete(d)} = \bar{\text{a}} \text{ and } \bar{\text{b}} \tag{7}$$
$$\text{propagate(p)} = \text{a and b (or a xor b)} \tag{8}$$
$$\text{sum} = \text{p xor carryin} \tag{9}$$
$$\text{carryout} = \text{g or p and carryin} \tag{10}$$

4. Ripple Carry Adder[14]

In the parallel adder , the carry out of each stage is connected to the carryin of the next stage. The sum and carryout bits of any stage cannot be produced, until some time after the carryin of that stage occurs. This is due to the propagation delay in the logic circuitry , which lead to a time delay in the addition process. The carry propagation delay for each full adder is the time between the application of the carryin and the occurrence of the carryout. The parallel adder in which the carryout of each full adder is the carryin to the next more significant adder is called a ripple carry adder.

5. Carry Look Ahead Adder[15]

In the case of the parallel adder , the speed with which an addition can be performed is governed by the time required for the carries to propagate or ripple through all the stages of the adder. The look ahead carry adder speeds up the process by eliminating this ripple carry delay. It examines all the input bits simultaneaously and also generates the carry in bits for all the stages simultaneously.

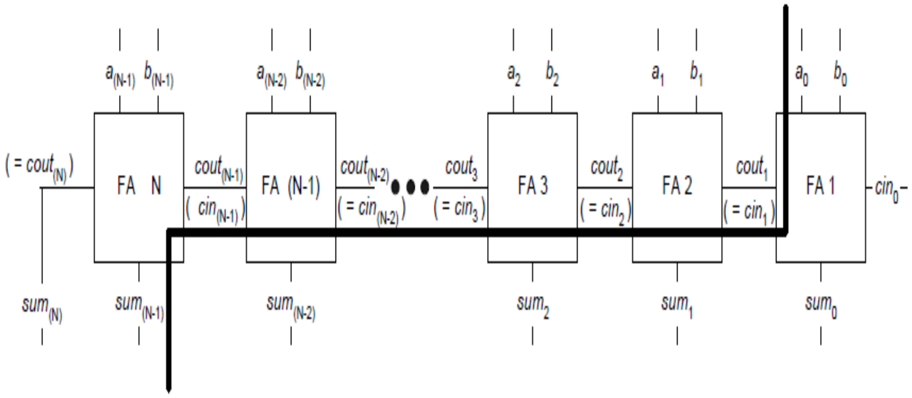
6. Carry save Adder

A carry-save adder is a type of digital adder, used in computer micro architecture to compute the sum of three or more *n*-bit numbers in binary. It differs from other digital adders in that it outputs two numbers of the same dimensions as the inputs, one which is a sequence of partial sum bits and another which is a sequence of carry bits.

III. Problem Identification

From the adder circuit we understand that the carry propagation is the main issue. In the ripple carry adder the carry out of each stage is connected to the carryin of the next stage. The sum and carryout bits of any stage cannot be produced, until some time after the carryin of that stage occurs. the time for this implementation of the adder is expressed in below Equation, where tRCAcarry is the delay for the carryout of a FA and tRCAsum is the delay for the sum of a FA.

Propagation Delay (tRCAprop) = (N - 1) . tRCAcarry + tRCAsum



.

Figure 3 Critical Path for an N-bit Ripple Carry Adder

In the multiplier, after partial product we again have to add that partial product by using adders. So if we want to speed up MAC unit we have to minimize carry propagation delay.

IV. Proposed Architecture of 8 Bit approximate adder

Here we proposed a new architecture of half adder and full adder as we know for 8 bit addition there is total 7 full adder and 1 half adder is require. But in proposed approach we propose a new novel 8 bit architecture where we can put some error on lsb bit of adder. Here in approximate half and full adder there is no any carry generation unit. So on first LSB bit we are using proposed approximate half adder and on second LSB bit we use one approximate full adder for next third bit there is no any carry generate so there is no need to use one full adder so at the place of full adder we are using one half adder and after that we use 5 full adder. So as we can see with small error generation we can reduce the hardware requirement and we can make justice with SPAA matrices.

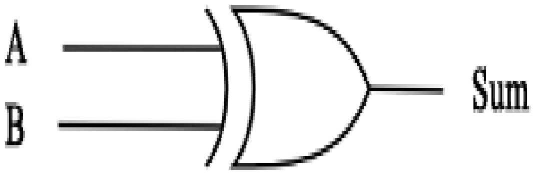


Figure 4 Proposed Approximate Half Adder

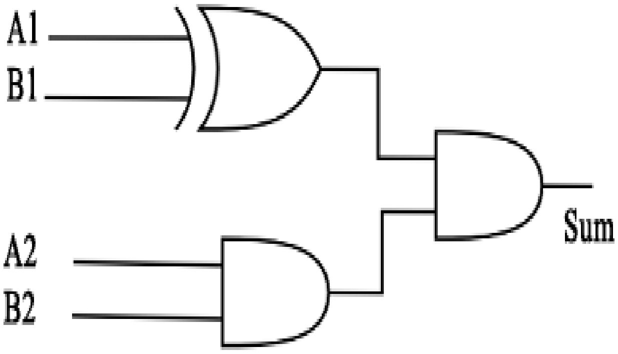


Figure 5 Proposed Approximate Full Adder

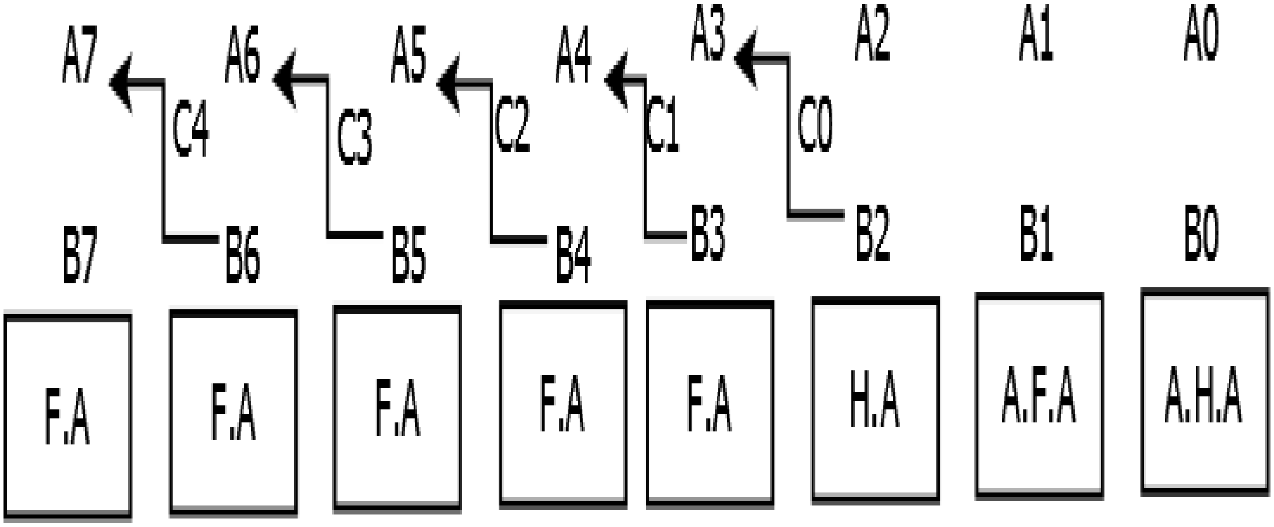


Figure 6 Proposed Architecture of 8 Bit Approximate Adder

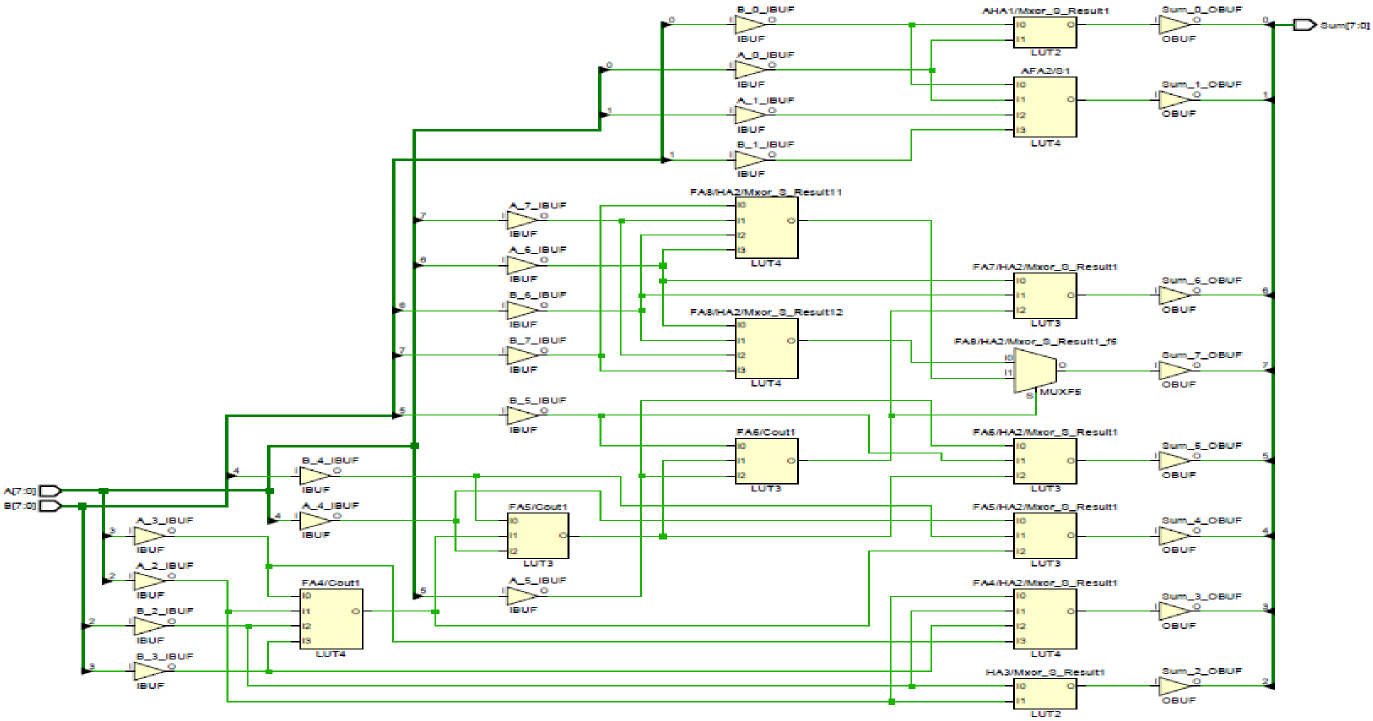
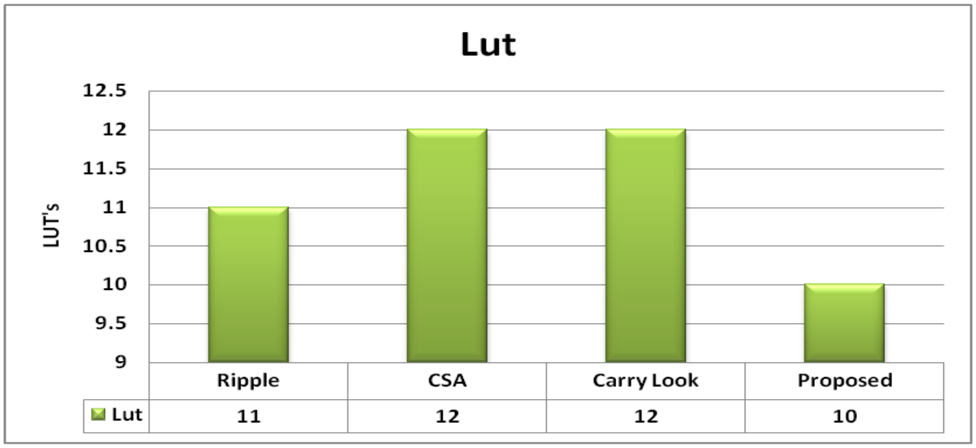


Figure 7 Implementation Detail of proposed 8 Bit approximate adder

V. Result & Hardware Analysis

Approximate Adder Accuracy Level = 95%

The FPGA comparison analysis of proposed and accurate are shown below, here hardware analysis is done on Vertix 6 FPGA which is 45nm based technology.



	Ripple	CSA	Carry Look	Proposed
🟩 Lut	11	12	12	10

Figure 8 Comparison analysis of Luts of Accurate and Proposed Adder

From the above graphs we can see that 35% reduction in logic block is achieved

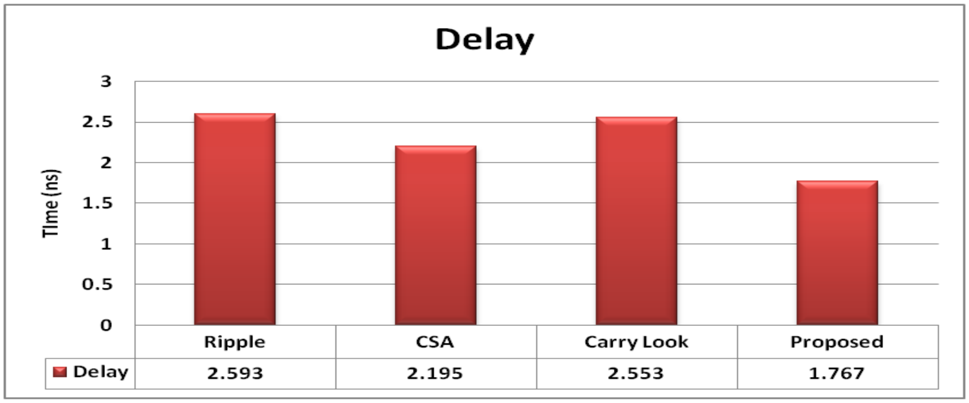


Figure 9 Comparison analysis of delay of Accurate and Proposed Adder

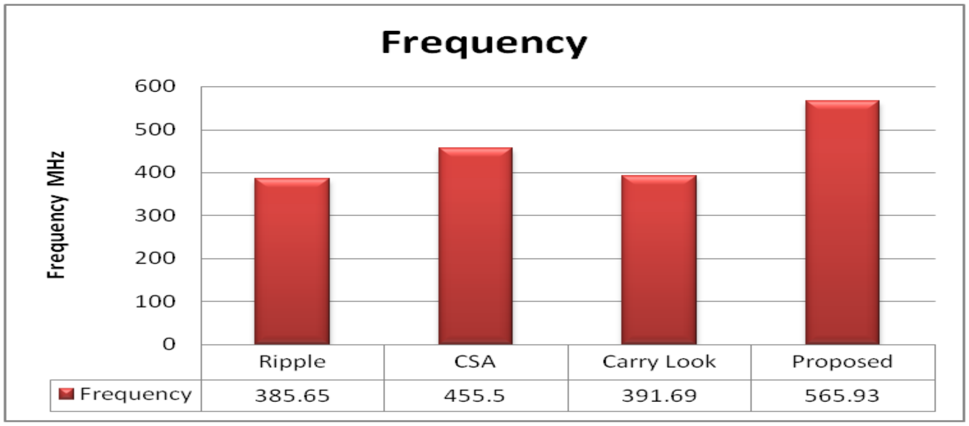


Figure 10 Comparison analysis of frequency of an accurate and Proposed Adder

VI. Conclusion

This paper present a approximate Adder. There is small degradation in image quality which is tolerable by human eye. The overall area and Delay and Frequency analysis are presented and compared. From the results we can depict that approximately up to 25 to 35% of reduction at all levels are achieved. So due to this we use approximation , which will minimize delay . This design is particularly useful in computation-intensive applications which are robust to small errors in computation. The potential applications of this approximate Adder 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 the accuracy. One example of such applications is in the DSP application for portable devices such as cell phones and laptops.

References

1. Leem, L.; Hyungmin Cho; Bau, J.; Jacobson, Q.A.; Mitra, S, "ERSA: Error Resilient System Architecture for probabilistic applications," *Design, Automation & Test in Europe Conference & Exhibition (DATE), 2010* , vol., no., pp.1560,1565, 8-12 March 2010
2. Ning Zhu; Wang-Ling Goh; Kiat-Seng Yeo, "An enhanced low-power high-speed Adder For Error-Tolerant application," *Integrated Circuits, ISIC '09. Proceedings of the 2009 12th International Symposium on* , vol., no., pp.69,72, 14-16 Dec. 2009
3. Kahng, A.B.; Seokhyeong Kang, "Accuracy-configurable adder for approximate arithmetic designs," *Design Automation Conference (DAC), 2012 49th ACM/EDAC/IEEE* , vol., no., pp.820,825, 3-7 June 2012
4. Rudagi, J M; Ambli, Vishwanath; Munavalli, Vishwanath; Patil, Ravindra; Sajjan, Vinaykumar, "Design and implementation of efficient multiplier using Vedic Mathematics," *Advances in Recent Technologies in Communication and Computing (ARTCom 2011), 3rd International Conference on* , vol., no., pp.162,166, 14-15 Nov. 2011
5. Abdelgawad, A.; Bayoumi, M., "High Speed and Area-Efficient Multiply Accumulate (MAC) Unit for Digital Signal Prossing Applications," *Circuits and Systems, 2007. ISCAS 2007. IEEE International Symposium on* , vol., no., pp.3199,3202, 27-30 May 2007
6. Mottaghi-Dastjerdi, M.; Afzali-Kusha, A.; Pedram, M., "BZ-FAD: A Low-Power Low-Area Multiplier Based on Shift-and-Add Architecture," *Very Large Scale Integration (VLSI) Systems, IEEE Transactions on* , vol.17, no.2, pp.302,306, Feb. 2009
7. Tung Thanh Hoang; Sjalander, M.; Larsson-Edefors, P., "A High-Speed, Energy-Efficient Two-Cycle Multiply-Accumulate (MAC) Architecture and Its Application to a Double-Throughput MAC Unit," *Circuits and Systems I: Regular Papers, IEEE Transactions on* , vol.57, no.12, pp.3073,3081, Dec. 2010
8. Lomte, R.K.; Bhaskar, P.C., "High Speed Convolution and Deconvolution Using Urdhva Triyagbhyam," *VLSI (ISVLSI), 2011 IEEE Computer Society Annual Symposium on* , vol., no., pp.323,324, 4-6 July 2011
9. Abdelgawad, A., "Low power multiply accumulate unit (MAC) for future Wireless Sensor Networks," *Sensors Applications Symposium (SAS), 2013 IEEE* , vol., no., pp.129,132, 19-21 Feb. 2013
10. Saokar, S.S.; Banakar, R. M.; Siddamal, S., "High speed signed multiplier for Digital Signal Processing applications," *Signal Processing, Computing and Control (ISPCC), 2012 IEEE International Conference on* , vol., no., pp.1,6, 15-17 March 2012
11. Gandhi, D.R.; Shah, N.N., "Comparative analysis for hardware circuit architecture of Wallace tree multiplier," *Intelligent Systems and Signal Processing (ISSP), 2013 International Conference on* , vol., no., pp.1,6, 1-2 March 2013
12. Prakash, A.R.; Kirubaveni, S., "Performance evaluation of FFT processor using conventional and Vedic algorithm," *Emerging Trends in Computing, Communication and Nanotechnology (ICE-CCN), 2013 International Conference on* , vol., no., pp.89,94, 25-26 March 2013
13. Itawadiya, A.K.; Mahle, R.; Patel, V.; Kumar, D., "Design a DSP operations using vedic mathematics," *Communications and Signal Processing (ICCSP), 2013 International Conference on* , vol., no., pp.897,902, 3-5 April 2013
14. Khan, S.; Kakde, S.; Suryawanshi, Y., "VLSI implementation of reduced complexity wallace multiplier using energy efficient CMOS full adder," *Computational Intelligence and Computing Research (ICCIC), 2013 IEEE International Conference on* , vol., no., pp.1,4, 26-28 Dec. 2013
15. Yu-Ting Pai; Yu-Kumg Chen, "The fastest carry lookahead adder," *Field-Programmable Technology, 2004. Proceedings. 2004 IEEE International Conference on* , vol., no., pp.434,436, 28-30 Jan. 2004
16. ZhouWang; Bovik, A.C.; Sheikh, H.R.; Simoncelli, E.P., "Image quality assessment: from error visibility to structural similarity," Image Processing, IEEE Transactions on , vol.13, no.4, pp.600,612, April 2004 doi: 10.1109/TIP.2003.819861
17. Xue, W.; Zhang, L.; Mou, X.; Bovik, A., "Gradient Magnitude Similarity Deviation:A Highly E_cient Perceptual Image Quality Index," Image Processing, IEEE Transactions on , vol.PP, no.99, pp.1,1
18. Itawadiya, A.K.; Mahle, R.; Patel, V.; Kumar, D., "Design a DSP operations using vedic mathematics," *Communications and Signal Processing (ICCSP), 2013 International Conference on* , vol., no., pp.897,902, 3-5 April 2013
19. Saokar, S.S.; Banakar, R. M.; Siddamal, S., "High speed signed multiplier for Digital Signal Processing applications," *Signal Processing, Computing and Control (ISPCC), 2012 IEEE International Conference on* , vol., no., pp.1,6, 15-17 March 2012