International Review on Computers and Software (I.RE.CO.S.), Vol. 9, N. 2
ISSN 1828-6003 February 2014

Design of Vedic Architecture for High Speed DCT

M. Thiruveni, D. Shanthi

Abstract – *Discrete Cosine Transform (DCT) is a frequency transform which is extensively used as a transform codec for still, moving image and video compression. The performance of DCT architecture mainly depends on multiplier and adder. In conventional architecture, array multiplier and Ripple Carry Adder (RCA) gives enormous delay when the number of input bits become more. After having detailed literature review, it is decided to design high speed VLSI DCT architecture with Vedic Multiplier and Carry Select Adder (CSA) for better performance. The Vedic multiplier is based on Urdhva Tiryakbhyam, the most efficient Sutra or algorithm which reduces the delay for all types of computation. The functionality of the proposed architecture is simulated using Modelsim and synthesis of verilog HDL code is done using Xilinx ISE. Both the DCT architectures are compared and the result shows that the delay of Vedic multiplier becomes low. The synthesis results show that the combinational delay for Vedic multiplier is reduced by 36% than conventional array multiplier. The combinational delay for CSA is reduced by 35% than RCA. The proposed architecture would give better performance in terms of speed for image and other signal processing applications.*

Keywords*: CSA, DCT, RCA, Urdhva Tiryakbhyam Sutra, Vedic*

I. Introduction

The DCT is a frequency transform used in image processing architectures. It is computation intensive operation. DCT is frequently used in image and signal processing for data compression, because it has a strong energy compaction property [1]-[18]. Low frequency components of DCT carry most of the signal information. DCTs are also closely related to Chebyshev polynomials and fast DCT algorithms. DCT requires large number of adders and multipliers for direct implementation. Multipliers consume more power. The redundancy in direct implementation of DCT is also exploited.

Mathematics is the backbone of engineering applications. The need for a fast and efficient DCT in signal processing applications is of interest. Multiplication is a basic mathematical operation. Many Signal Processing applications such as Convolution, Discrete Fourier Transform, Fast Fourier Transform, Discrete Cosine Transform, and Filtering are implemented with multiplication based operations. So the need for fast and low power multiplier arises. DCT architecture consists of a multiplier along with a fast adder.

Urdhva Tiryakbhyam Sutra or vertically and Crosswise Algorithm develops the architecture of digital multiplier which is similar to the array multiplier architecture. Multiplication algorithm for NxN bit multiplication can be given by this sutra.

The multiplication algorithm reduces an NxN bit multiplication to a 2x2-bit multiplication operation by dividing it into smaller and smaller groups.

A fast multiplier based on Vedic Mathematics can be developed by this methodology.

In digital adders, the delay is increased by the time taken for carry propagation. The sum is calculated in each full adder only after receiving the input carry from the previous full adder. This delay can be reduced by fast carry select adder. Instead of RCA, CSA is used in order to reduce delay. To replace the n-bit RCA, a 3(n/2) bit CSA is required. This paper proposes 8-point DCT Vedic architecture. The above said Vedic multiplier and CSA are used to form the DCT unit.The performance of the vedic multiplier is analysed and found that the delay is reduced drastically, hich results in high speed operation.

II. Literature Survey

The Vedic multiplier [1] is designed using Urdhva Tiryakbhyam Sutra which is suitable for fast multiplication in Digital Signal Processing (DSP) applications. The delay comparison between Vedic and modified booth Wallace tree multiplier proves its efficiency in terms of speed. The Vedic multiplier [2] is designed for computer intensive operations. It is a fast multiplier with reduced complexity. It is applicable for all cases of multiplication. This can be embedded in any signal processing unit. The Vedic multiplier and square architecture [3] is proposed for low power and high speed. All partial products of multiplication and their sum are generated in a single step. It is efficient in terms of speed. It saves time when used for image or video processing applications.

Manuscript received and revised January 2014, accepted February 2014

Low complexity DCT architecture [4] uses vector processing for image compression. It identifies the redundant computations and removes them in matrix operation. This reduction in computational complexity of DCT reduces power consumption.

III. Discrete Cosine Transform

DCT expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies. DCT is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using only real numbers. DCT can be expressed as:

$$X(k) = e(k)\sum_{n=0}^{N-1} x(n)\cos\left[\frac{(2n+1)k\pi}{2N}\right] \quad (1)$$

$$k = 0,1,...,N-1$$

$$x(n) = \frac{2}{N}\sum_{n=0}^{N-1} e(k)\,X(k)\cos\left[\frac{(2n+1)k\pi}{2N}\right] \quad (2)$$

$$n = 0,1,...,N-1$$

where:

$$e(k) = \begin{cases} \frac{1}{\sqrt{2}} \text{ if } k = 0 \\ 1 \quad \text{otherwise} \end{cases} \quad (3)$$

The 8-point DCT [5] can be written in the matrix form based on the following Eq. (4):

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \\ X(6) \\ X(7) \end{bmatrix} = \begin{bmatrix} c4 & c4 & c4 & c4 & c4 & c4 & c4 & c4 \\ c1 & c3 & c5 & c7 & c9 & c11 & c13 & c15 \\ c2 & c6 & c10 & c14 & c18 & c22 & c26 & c30 \\ c3 & c9 & c15 & c21 & c27 & c1 & c7 & c13 \\ c4 & c12 & c20 & c28 & c4 & c12 & c20 & c28 \\ c5 & c15 & c25 & c3 & c13 & c23 & c1 & c11 \\ c6 & c18 & c30 & c10 & c22 & c2 & c14 & c26 \\ c7 & c21 & c3 & c17 & c31 & c13 & c27 & c9 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}$$

where:

$$c_i = cos\ (i\pi/16) \quad (5)$$

Using trigonometric properties based on Eq. (5), the matrix can be rewritten as the following Eq. (6):

$$\begin{bmatrix} X(0) \\ X(1) \\ X(2) \\ X(3) \\ X(4) \\ X(5) \\ X(6) \\ X(7) \end{bmatrix} = \begin{bmatrix} c4 & c4 & c4 & c4 & c4 & c4 & c4 & c4 \\ c1 & c3 & c5 & c7 & -c7 & -c5 & -c3 & -c1 \\ c2 & c6 & -c6 & -c2 & -c2 & -c6 & c6 & c2 \\ c3 & -c7 & -c1 & -c5 & c5 & c1 & c7 & -c3 \\ c4 & -c4 & -c4 & c4 & c4 & -c4 & -c4 & c4 \\ c5 & -c1 & c7 & c3 & -c3 & -c7 & c1 & -c5 \\ c6 & -c2 & c2 & -c6 & -c6 & c2 & -c2 & c6 \\ c7 & -c5 & c3 & -c1 & c1 & -c3 & c5 & -c7 \end{bmatrix} \begin{bmatrix} x(0) \\ x(1) \\ x(2) \\ x(3) \\ x(4) \\ x(5) \\ x(6) \\ x(7) \end{bmatrix}$$

The algorithm architecture transformations can be used to derive efficient DCT implementation where the number of multiplications can be reduced [15]. It works in a hierarchical way to adapt architecture to a given algorithm or change the algorithmic description in a systematic way. The number of multiplications is reduced from 56 to 13 based on Eq. (6).

The algorithm-architecture mapping for the 8-point DCT can be carried out in three steps.

ii) DCT algorithm is modified and the architecture is designed using trigonometric properties that are simpler to implement based on Eq. (6) (see Fig. 1).

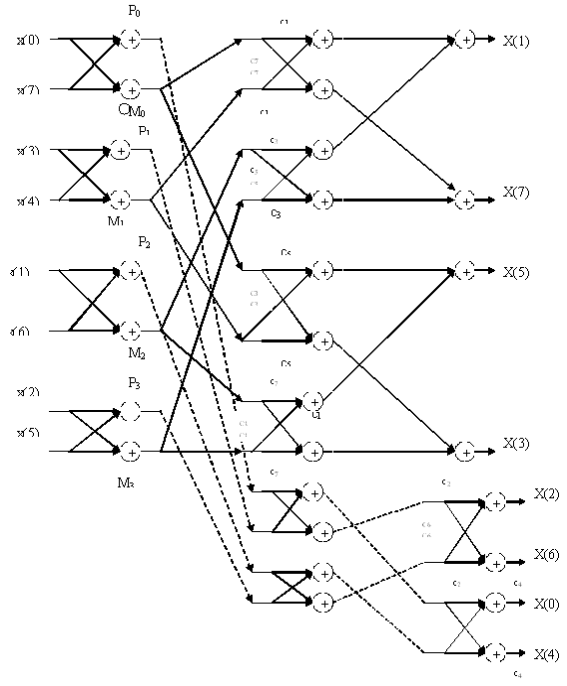

Fig. 1. Matrix Implementation of 8-point DCT

ii) The DCT structure shown in Fig. 1 is transformed into a modified structure (see Fig. 2) in which different functional units are grouped into blocks.

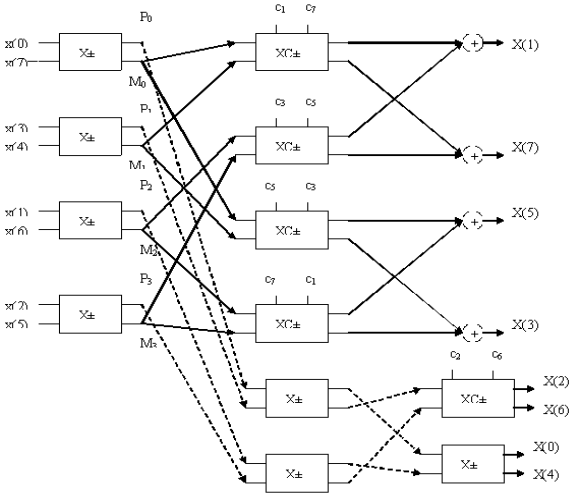

Fig. 2. Modified structure of 8-point DCT

iii) The operations are realized by implementation of various blocks (see Fig. 3). The block XC± can be realized using 3 multiplications and 3 additions instead of using 4 multiplications and 2 additions (see Fig. 4).

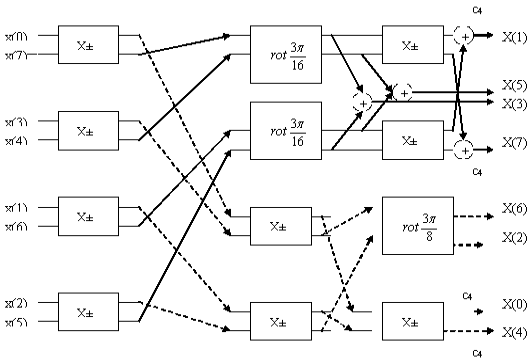

Fig. 3. Reduced structure of 8-point DCT

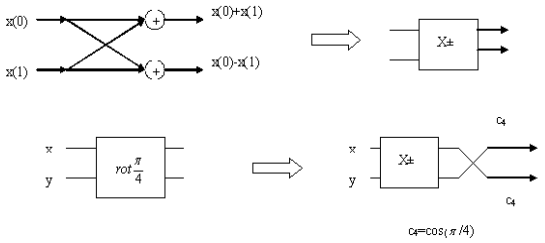

Fig. 4. Basic Block Definitions

IV. Vedic Sutras

The word Veda derives the word "Vedic", Knowledge store. The 16 Sutras [6] dealing with various branches of mathematics like arithmetic, geometry, algebra, etc forms the Vedic Mathematics. These Sutras along with their brief meanings are enlisted below alphabetically (Table I).

TABLE I
LIST OF SUTRAS

Sutra	Meaning
(Anurupye) Shunyamanyat	If one is in ratio, the other is zero.
Chalana-Kalanabyham	Dissimilarities and Similarities
EkadhikinaPurvena	One greater than preceding one
EkanyunenaPurvena	One smaller than the preceding one.
Gunakasamuchyah	The factors of the sum are equal to the sum of the factors.
Gunitasamuchyah	The product of the sum is equal to the sum of the product.
Nikhilam Navatashcaramam Dashatah	All from 9 and last from 10.
ParaavartyaYojayet	Reverse and Regulate
Puranapuranabyham	By the completion or non completion.
Sankalana	By addition and by subtraction.
Shesanyankena Charamena	The remainders by the last digit.
Shunyam Saamyasamuccaye	When the sum is the same that sum is zero.
Sopaantyadvayamantyam	The last and twice of the last but one.
Urdhva-tiryagbhyam	Vertically and crosswise
Vyashtisamanstih	Part and Whole
Yaavadunam	Whatever the extent of its shortage

The Vedic mathematics is of interest in reducing complex calculations into simple one, because it approaches and solves the calculation in a natural way like how the human mind works. Various branches of engineering such as computing and digital signal processing make use of this algorithm.

IV.1. Vedic Multiplication

The proposed work is the extension of the Vedic multiplication formula for the binary number system [6] which is traditionally used for the multiplication of decimal numbers. It makes the design compatible with the digital hardware.

IV.2. Urdhva Tiryakbhyam Sutra

The multiplier is based on Urdhva Tiryakbhyam Sutra or ―Vertically and crosswise algorithm of ancient Indian Vedic Mathematics. Partial products are generated simultaneously and adder concurrently. The multiplier has the advantage that as the number of bit increases, gate delay and area increases very slowly as compared to other multipliers. Therefore it is time efficient. It is demonstrated that this architecture is quite efficient in terms of speed. A 4×4 bit multiplication is simplified into four 2×2 bit multiplications that can be performed in parallel (see Fig. 5). This reduces the number of stages of logic and thus reduces the delay of the multiplier. This example illustrates a better and parallel implementation style of Urdhva Tiryakbhyam Sutra.

The beauty of this approach is that larger bit steams (of say N bits) can be divided into (N/2 = n) bit length, which can be further divided into n/2 bit streams and this can be continued till we reach bit streams of width 2, and they can be multiplied in parallel, thus providing an increase in speed of operation.

Multiplication of 2 digit numbers using Urdhva Tiryakbhyam Sutra (1101 × 1010).

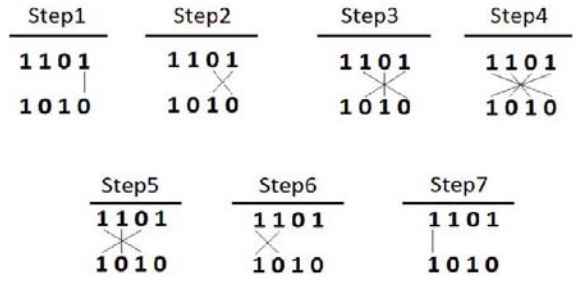

Fig. 5. Urdhva Tiryakbham for binary numbers

IV.3. Ripple Carry Adder

The ripple carry adders are used in the existing works which results in more delay since each bit addition depends on the previous bit carry. The 16 bit RCA uses 16 full adder for its construction and it adds two 16 bit numbers with input carry (normally 0) (see Fig. 6).

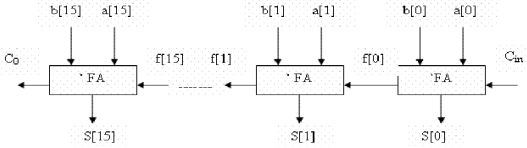

Fig. 6. Hardware Realization of 16 bit Ripple Carry Adder

V. Proposed Work

DCT requires large number of multipliers and adders for direct implementation. In the proposed architecture, multiplication can be performed with the Vedic multiplier which is efficient than array multiplier. Since multiplier is fast, it makes the DCT architecture fast. The ripple carry adder used for n -bit addition is replaced by carry select adder which results in less delay.

The design starts with 2×2 bit multiplier. Urdhva Tiryakbhyam Sutra or vertically and Crosswise Algorithm for multiplication has been effectively used to develop digital multiplier architecture [7]. Its modularity builds higher blocks from smaller blocks.

V.1. Vedic Multiplier

1) 2×2 Bit Multiplier

The 2x2 bit multiplier is the basic block that is used in 4x4 bit multiplier. The structure of 2×2 bit multiplier (see Fig. 7). In 2×2 bit multiplier, the multiplicand has 2 bits each and the result of multiplication is of 4 bits.

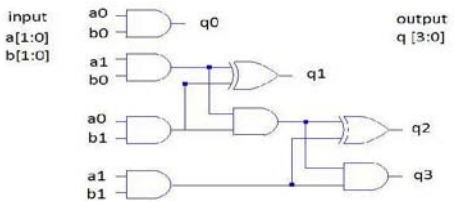

Fig. 7. Hardware Realization of 2×2 block

2) 4×4 Bit Multiplier

The 4×4 bit multiplier is the basic block that is used in 8×8 bit multiplier. The block diagram of 4×4 bit multiplier (see Fig. 8). In 4×4 bit multiplier, the multiplicand has 4 bits each and the result of multiplication is of 8 bits.

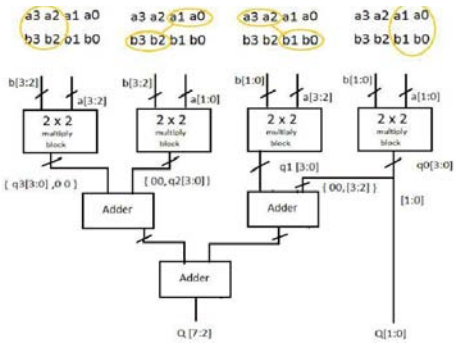

Fig. 8. Hardware Realization of 4×4 block

3) 8×8 Bit Multiplier

The 8×8 bit multiplier is the basic block that is used in 16×16 bit multiplier. The block diagram of 8×8 bit multiplier (see Fig. 9). In 8×8 bit multiplier, the multiplicand has 8 bits each and the result of multiplication is of 16 bits [8]-[13].

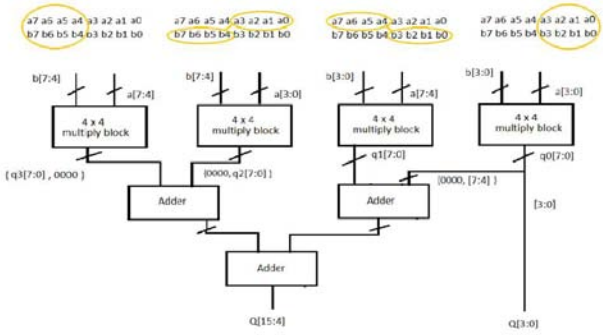

Fig. 9. Hardware Realization of 8x8 block

V.2. Carry Select Adder

In functional architecture of CSA(see Fig. 10), the 16 bit number is divided into two 8 bit numbers and the 8-bit LSB is added using 8 bit RCA with input carry as 0 [14]. Parallely, the 8 bit MSB is added in two 8 bit RCA's with input carry as 0 and 1 respectively. A 2×1 multiplexer is used to choose the sum from the two 8 bit RCA's with output carry of the 8 bit LSB RCA as select line.

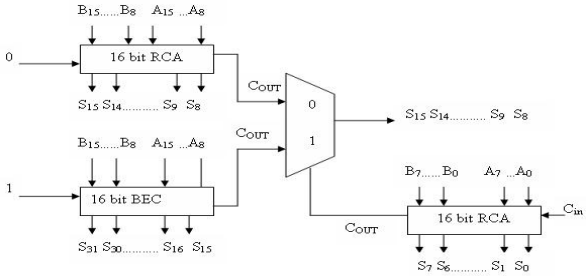

Fig. 10. Hardware Realization of CSA

V.3. Vedic Discrete Cosine Transform

The multipliers and adders of the proposed DCT architecture is replaced by the Vedic multiplier and CSA which reduces the delay of DCT architecture. The DCT architecture delay is reduced from 1125.29ns to 1040.20ns and reduced DCT architecture delay is reduced from 598.71ns to 314.12ns.It shows that the delay of DCT is reduced by 7.5% and Reduced DCT is reduced by 14%.

The reduction in delay is represented in percentage by the formula given in Eq. (7):

$$\%Delay = \frac{Conventional\ DCT - Vedic\ DCT}{Conventional\ DCT} \times 100 \quad (7)$$

VI. Results and Discussions

The proposed Vedic multiplier and the existing array multiplier architectures are designed using Verilog hardware description language (HDL) and synthesized using Xilinx ISE. Both the multiplier architectures have been designed for five different input bit lengths.

Even though 8-point DCT requires 8x8 bit multiplier, design extends up to 32x32 bit multiplier in order to show that the delay of Vedic multiplier increases very slowly with increase in the number of input bits that as shown in Table II and Fig. 11.

TABLE II
COMPARISON OF VEDIC MULTIPLIERS

	Array Multiplier		Vedic Multiplier	
Size of Multiplier	Delay (ns)	No. of Slices	Delay (ns)	No. of Slices
2×2 bit	8.104	1	8.49	2
4×4 bit	20.203	14	14.04	15
8×8 bit	31.304	73	19.97	74
16×6 bit	41.732	308	26.505	319
32×32 bit	50.420	598	31.413	623

The delay is analysed for all the 10 circuits as shown in Table II. The result (see Fig. 11) clearly shows that the delay of Vedic multiplier increases very slowly with increase in the number of input bits. The delay of array multiplier increases linearly with increase in the number of input bits.

TABLE III
COMPARISON OF ADDERS

ADDER	Delay (ns)	No. of Slices
RCA	26.104	18
CSA	19.695	23

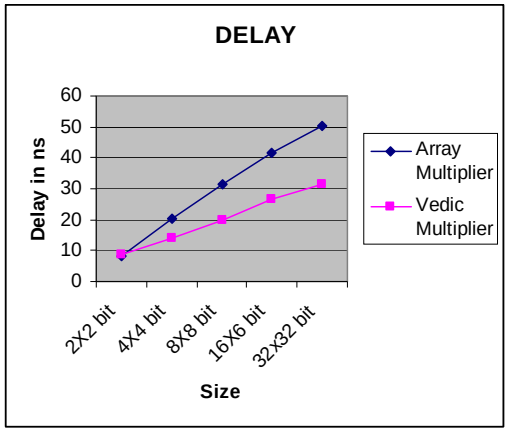

Fig. 11. Comparison of Delay in Vedic and Array Multipliers

The area is also analysed for all the 10 circuits as shown in Table II. The result (see Fig. 12) clearly shows that the area of Vedic multiplier is slightly increasing when compared to the area of array multiplier.

It is clear that Vedic multiplier is much faster than array multiplier with slight increase in area. It is more efficient for high speed architectures. The results of RCA versus CSA are obtained by simulation and synthesis.

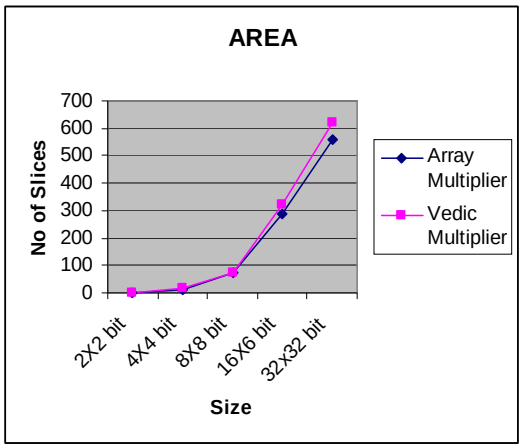

Fig. 12. Comparison of area in Vedic and Array Multipliers

The delay and area are compared as shown in Table III. The result (see Fig. 13) clearly shows that the delay of CSA is less when compared to the delay of RCA where area required for RCA architecture is less when compared to CSA. So CSA can be chosen for high speed architectures and RCA can be used for compact architectures. The delay is reduced drastically in CSA .The reduction in delay will be more for large CSA with higher order of input bits.

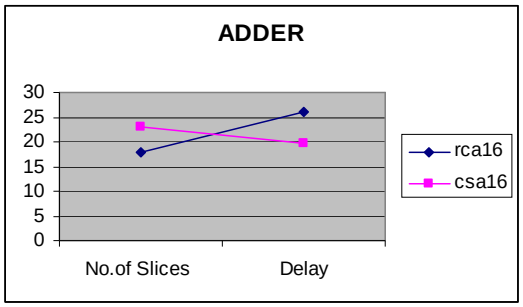

Fig. 13. Comparison of Delay in RCA and CSA

The design of conventional DCT architecture shown in Fig. 1, reduced DCT architecture shown in Figure 2 and its equivalent Vedic DCT architectures are developed in verilog HDL and synthesized. The delay and area for all the four architectures are compared in Table IV.

TABLE IV
COMPARISON OF DCT ARCHITECTURES

	DCT	Reduced DCT	Vedic DCT	Vedic Reduced DCT
Delay(ns)	1125.29	598.71	1040.20	314.12
Slices	670	150	753	160

The result (see Fig. 14) clearly shows that the delay of Vedic reduced DCT is less when compared to the delay of DCT, reduced DCT and Vedic DCT architectures.

It is clear that Vedic reduced DCT is much faster with slight increase in area than conventional DCT architectures (see Fig. 15) Vedic reduced DCT is the efficient high speed DCT architecture.

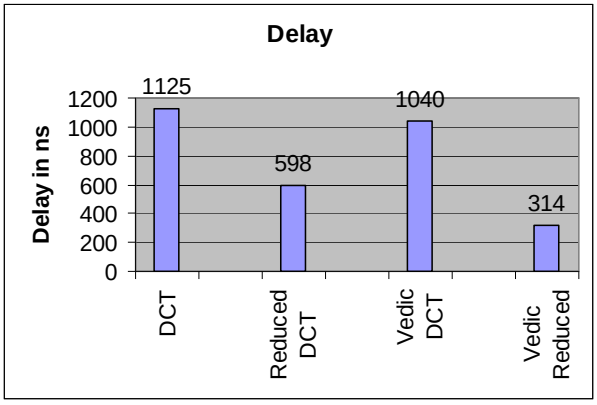

Fig. 14. Comparison of Delay in DCT Architectures

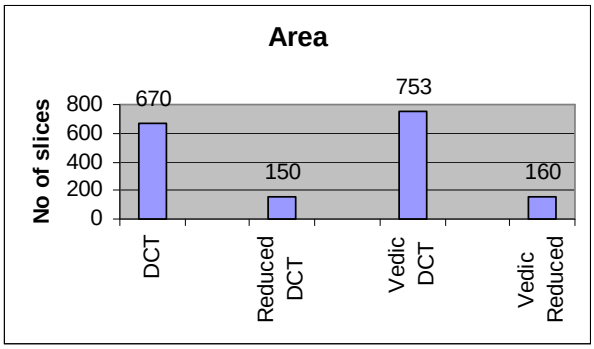

Fig. 15. Comparison of area in DCT Architectures

VII. Conclusion

In this paper, high speed architecture for DCT has been proposed. This has been achieved by replacing the conventional array multiplier with Vedic multiplier and RCA with CSA. In the proposed architecture the delay in the multiplier has been reduced in to 19.97 nano seconds from 31.304 nano seconds and in adder the delay has been reduced to 19.69 nano seconds from 26.10 nano seconds. There is only one limitation that the no of slices in the multiplier and CSA slightly increases when we go for higher number of bits.

References

[1] Ramesh Pushpangadan, Vineeth sukumaran, Rino innocent,Dinesh sasikumar, Vaisak Sundar, "High Speed Vedic Multiplier for Digital Signal Processors", *IET E Journal of Research, Vol . 55*, Issue 6, pp 282-286, 2009.

[2] J. M Rudagi, Vishwanath Ambli, Vishwanath Munavalli, Ravindra Patil, Vinaykumar Sajjan, "Design and Implementation of efficient multiplier using Vedic mathematics", *proceedings of Int . Conf. on Advances in Recent Technologies in communication and computing*, 2011.

[3] Vijaya Prakash. A.M, K.S. Gurumurthy, "A Novel VLSI Architecture for Image Compression Model Using Low power Discrete Cosine Transform", *World Academy of Science, Engineering and Technology*, 2010.

[4] Mu-Huo Cheng and Yu-Hsin Hsu, "Fast IMDCT and MDCT Algorithms A Matrix Approach", *IEEE Transactions on signal processing, vol. 51*, no. 1, 2003

[5] M.Thiruveni, M.Deivakani "Design of Analog VLSI Architecture for DCT", *International Journal of Engineering and Technology, vol 2*, no.8, pp 1475-81, 2012.

[6] Jagadguru Swami, Sri Bharati Krisna, Tirthaji Maharaja (1986), "Vedic Mathematics or Sixteen Simple Mathematical Formulae from the Veda, Delhi (1965)", Motilal Banarsidas, Varanasi, India.

[7] Pushpalata Verma, K. K. Mehta (2012), " Implementation of an Efficient Multiplier based on Vedic Mathematics Using EDA Tool", *International Journal of Engineering and Advanced Technology (IJEAT), vol.1*, no.5, pp 76-9, 2012.

[8] M.Prathan, R.Panda (2010), "Design and Implementation of Vedic Multiplier", *A.M.S.E Journal Series D, Computer Science And Statistics, vol.15*, Issue2, pp.1-19, 2010.

[9] M.Prathan, R.Panda, S.Kumarsahu(2011), "MAC implementation using Vedic multiplication algorithm" , *International journals of computer applications , Vol. 21*,No.7, pp 26-8, May 2011.

[10] Parth Mehta, Dhanashri Gawale, "Conventional versus Vedic Mathematical Method for Hardware Implementation of a Multiplier", *International Conference on Advances in computing, control and Telecommunication Technologies* Trivandrum, Kerala , India, 2009.

[11] M.Prathan, R.Panda ,S.Kumarsahu (2011), "Speed comparison of 16×16 Vedic Multipliers" , *International journals of computer applications, vol.21*, no.6, pp.16 -9, 2011

[12] A. Abelgawad, Magdy Bayoumi, "High speed and area-efficient Multiply Accumulate (MAC) unit for digital signal processing applications", *IEEE International symposium on circuits and systems*, 2007.

[13] C.Senthilpari, "A Low-power and High-performance Radix-4 Multiplier Design Using a Modified Pass-transistor Logic Technique", *IET E Journal of Research, Vol 57*, no.2 , pp. 149-155, 2011.

[14] B. Ramkumar, Harish M kittur (2012), "Low-Power and Area-Efficient Carry Select Adder", *IEEE Transactions on Very Large Scale Integration (VLSI) systems, Vol. 20*, no. 2, pp371-6, 2012.

[15] Abhishek, A.K., Aneesh, M.U., Arun, B.V., Yaradoni, D.K.S., Manikantan, K., Ramachandran, S., Circular sector DCT based feature extraction for enhanced Face Recognition with image segmentation as a pre-processing step, (2012) *International Review on Computers and Software (IRECOS)*, 7 (5), pp. 1954-1968.

[16] Senhaji, S., Aarab, A., A new and robust image watermarking technique using contourlet-DCT domain and decomposition model, (2013) *International Review on Computers and Software (IRECOS)*, 8 (3), pp. 747-752.

[17] Jokar, E., Pourghassem, H., Kidney region extraction in ultrasound images based on gradient descent method and Curvelet transform-based enhancement, (2012) *International Review on Computers and Software (IRECOS)*, 7 (1), pp. 132-142.

[18] Balaji, V.R., Subramanian, S., A discrete fractional cosine transform based speech enhancement system through Adaptive Kalman filter Combined with perceptual weighting filter with pitch synchronous analysis, (2013) *International Review on Computers and Software (IRECOS)*, 8 (9), pp. 2288-2295.

Authors' information

M. Thiruveni born in palani, Dindigul in 1980.She received her B.E. Electronics and Communication Engineering in 2002 from Jayaram College of Engineering and Technology, Trichy, Tamil Nadu and M.E. VLSI Design from Kongu Engineering College, Tamil Nadu and Pursuing PhD in VLSI Design. Her research interests include VLSI, Signal Processing, Digital Circuits and Architecture Design.
Address: Associate Professor, Department of ECE, PSNACET, Dindigul, Tamilnadu, India-624622.
E-mail: thiruveniraguraman@gmail.com

Shanthi Saravanan born in Dindigul in 1974.She received her B.E. Computer Science and Engineering in 1992 from Thiagarajar College of Engineering, Madurai, Tamil Nadu and M.E.Computer science and Engineering in from Manonmaniam Sundaranar University, Tamil Naduand PhD in Soft Computing from Birla Institute Of Technology, Ranchi. Her research interests include Genetic Algorithms, Neural Networks, Intelligent Systems, Image Processing, Embedded System, M-Learning and Green Computing. She has published more than 25 papers in International Conferences and Journals and also published 5 books in computing and applications. Dr. Shanthi Saravanan is member of various professional societies like IEEE, CSTA, IAENG and IACSIT.

Address: Professor, Department of CSE, PSNACET, Dindigul, Tamilnadu, India-624622.
E-mail: dshan71@gmail.com

