ISSN: 2456-043X

International Journal of Engineering Science and Generic Research (IJESAR)
Available Online at www.ijesar.in
Volume 2; Issue 5; September-October; 2016; Page No. 100-107

DESIGN AND IMPLEMENTATION OF RECONFIGURABLE 64-POINT DISCRETE COSINE TRANSFORM (DCT) ARCHITECTURE

Meesala Shiva Kumar [1], Dr. G V Maha Lakshmi [2]

[1]Department of Electronics and Communication Engineering, Sreenidhi Institute of Science and Technology, Affiliated to Jawaharlal Nehru Technological University, Hyderabad, Telangana, India.

[2]Department of Electronics and Communication Engineering, Sreenidhi Institute of Science and Technology, Affiliated to Jawaharlal Nehru Technological University, Hyderabad, Telangana, India.

Email id: meesalashivakumar@gmail.com, mahalakshmi@sreenidhi.edu.in

Abstract

The existing algorithms for approximation of DCT targets only on the DCT of small transform lengths, the main objective is reducing the power and calculation time. Multiplications are the operations in DCT which consumes majority of time and power and it is very complex to calculate the values of DCT. Approximation is needed in DCT for higher transform lengths as computational complication increases non-linearly with higher size lengths DCT. To offer lower circuit complexity and superior compression performance Multiplier-free approximate DCTs have been implemented which can be easily implemented in VLSI hardware by using only addition operation and subtraction operations. Thus, compared to integer and conventional DCTs, approximated DCTs result in reduction of the chip area as well as in power consumption.

In this paper, here an algorithm is presented for approximation of DCT where an approximate DCT of length N could be derived from pair of DCTs of length (N/2) at cost of N additions. This algorithm is highly scalable for hardware as well as software implementation of DCT of higher lengths and it make use of the existing approximation of 8-point DCT to obtain approximate DCT of any power of 2 length{N>8}.It involves lower arithmetic complexity compared with the other existing approximation algorithms. It provides better image and video compression performance then the existing approximation methods. A fully scalable reconfigurable parallel architecture for computation of approximate 32-point DCT based on algorithm is implemented, The interesting feature of this architecture is that it could be configured for the computation of a 32-point DCT (or) for parallel computation of two 16-point DCTs (or) for parallel computation of four 8-point DCTs with less control overhead. A fully scalable reconfigurable parallel architecture for computation of approximate 32-point DCT extended to 64-point DCT based on same algorithm, it also could be configured for the computation of a 64-point DCT (or) for parallel computation of two 32-point DCTs (or) for parallel computation of four 16-point DCTs (or) for parallel computation of eight 8-point DCTs with less control overhead. A reconfigurable 32-point discrete cosine transform (DCT) and 64-point discrete cosine transform (DCT) architecture is simulated and synthesized by Xilinx 14.2 tool.

KEYWORDS: DCT. Approximation of DCT, Scalable DCT, Reconfigurable DCT.

I. INTRODUCTION

THE DISCRETE COSINE TRANSFORM (DCT) expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies. DCTs are important to numerous applications in science and engineering, from lossy compression of audio (e.g. MP3) and images (e.g. JPEG) (where small high-frequency components can be discarded) The use of cosine rather than sine functions is critical for compression, since it turns that fewer cosine functions are needed to approximate a typical signal, DCT is mainly used in Image and Video compression. 'Image data compression has been an active research area for image processing over the last decade and has been used in a variety of applications. Image and video data compression refers to a process in which the amount of data used to represent image and video is reduced to meet a bit rate requirement

Corresponding authors: Meesala Shiva Kumar, Dr. G V Maha Lakshmi

(below or at most equal to the maximum available bit rate), while the quality of the reconstructed image or video satisfies a requirement for a certain application and the complexity of computation involved is affordable for the application.

Recently, new transforms have been presented for 8-point DCT approximation: Cintra have presented a low-complexity 8-point approximate DCT based on integer functions and Potluri have presented a novel 8-point DCT approximation that requires only 14 additions. Bouguezel have proposed two methods for multiplication-free approximate form of DCT. The first method is for length N=8, 16 and 32, and is based on the appropriate extension of integer DCT. Cintra have proposed a new 16X16 matrix also for approximation of 16-point DCT. Several methods for approximation of DCT have been achieved in the literature. These approximation methods are very close to exact DCT coefficients with requiring only lower computations. Approximation methods are free of multipliers with eliminating all floating-point operations. Hence, these approximation methods replace the exact DCT method providing low power, low cost ,economical hardware and . high performance.

II. EXISTING METHODS

A. APPROXIMATION OF 8-POINT DCT

The 8-point discrete cosine transform (DCT) is a key step in many image and video processing applications. This particular block length is widely adopted in several image and video coding standards, such as JPEG, MPEG-1 and MPEG-2. This is mainly due to its good energy compaction properties.

The basic block for the computation of 8-point DCT is shown in figure 1.This 8-point DCT is used as the basis for the proposed system for 16, 32 and 64 point DCTs. This existing transformation matrix contains only zeros and ones, multiplications and bit shift operations are absent.

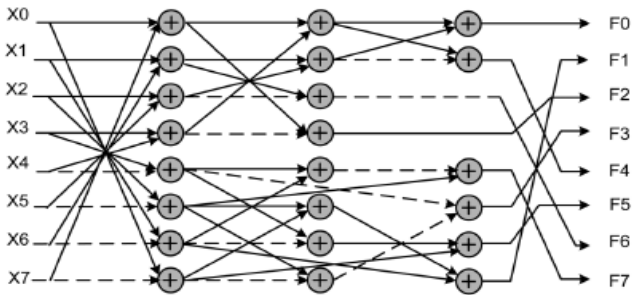

Figure 1. Design Flow Graph of 8-point DCT

B. SCALABLE DESIGN

The block diagram of 16-point DCT is illustrated in Figure. 2

For 16-point DCT first the inputs are fed to adder of 16-point unit and the outputs of this adder unit is now the inputs for two 8-point DCTs. The results for 16-point DCT is computed by same 8-point DCT as calculated above and at the output side the results are arranged to get the ordered DCT of 16-point.

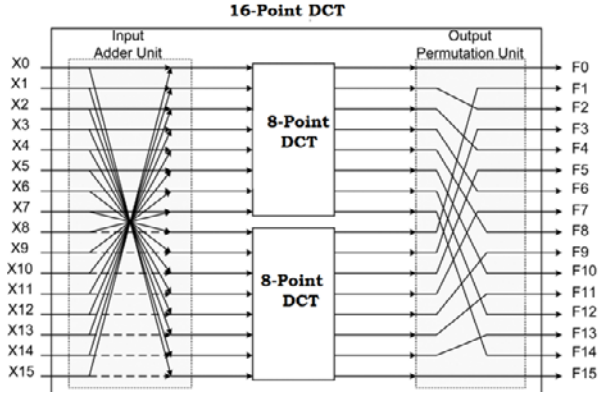

Figure. 2. Scalable Design of 16-point DCT

III. RECONFIGURABLE ARCHITECTURE FOR DCT COMPUTAION

In this section, Reconfigurable 16-point DCT Architecture, 32-point DCT Architecture and 64-point DCT Architecture is discussed.

A. RECONFIGURABLE 16-POINT DCT ARCHITECTURE

The reconfigurable architecture for the implementation of approximated 16-point DCT is shown in Figure. 3. It consists of three computing units, namely two 8-point approximated DCT units and a 16-point input adder unit that generates a(i) and b(i), i belongs to[1:7]. The input to the first 8-point DCT approximation unit is fed through 8 MUXes that select either[a(0),a(1),.....a(7)] or [X(0),X(1),..... X (7)], depending on whether it is used for 16-point DCT calculation or 8-point DCT calculation. Similarly, the input to the second 8-point DCT unit is fed through 8 MUXes that select either[b(0),b(1),.....b(7)] or [X(8),X(9),..... X (15)], depending on whether it is used for 16-point DCT calculation or 8-point DCT calculation. On the other hand, the output permutation unit uses 14 MUXes to select and re-order the output depending on the size of the selected DCT. Sel16 is used as control input of the MUXes to select inputs and to perform permutation according to the size of the DCT to be computed. Specifically, Sel16 = 1 enables the computation of 16-point DCT and Sel16 = 0 enables the computation of a pair of 8-point DCTs in parallel. Consequently, the architecture of Figure. 3 allows the calculation of a 16-point DCT or two 8-point DCTs in parallel.

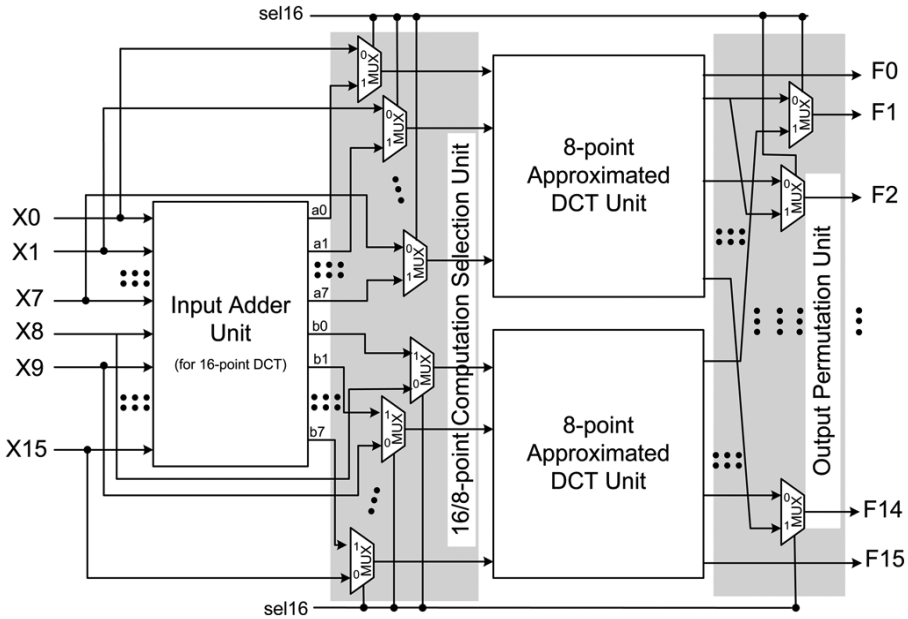

Figure 3.Reconfigurable 16-Point DCT Architecture

B. RECONFIGURABLE 32-POINT DCT ARCHITECTURE

A reconfigurable design for the computation of 32-pomt, 16-point, and 8-point DCTs is presented in Figure. 4. It performs the calculation of a 32-point DCT or two 16-point DCTs in parallel or four 8-point DCTs in parallel. The architecture is composed of 32-point input adder unit, two 16-point input adder units, and four 8-point DCT units. The reconfigurability is achieved by three control blocks composed of 64 2:1 MUXes along with 30 3:1 MUXes. The first control block decides whether the DCT size is of 32 or lower. If Sel32 = 1, the selection of input data is done for the 32-point DCT, otherwise, for the DCTs of lower lengths. The second control block decides whether the DCT size is higher than 8. If Sel16 = 1 the length of the DCT to be computed is higher than 8 (DCT length of 16 or 32), otherwise, the length is 8. The third control block is used for the output permutation unit which re-orders the output depending on the size of the selected DCT .Sel32 and Sel16 are used as control

signals to the 3:1 MUXes. Specifically, for $\{Sel32, Sel16\}_2$ equal to {00}, {01} or {11} the 32 outputs correspond to four 8-point parallel DCTs, two parallel 16-point DCTs, or 32-point DCT, respectively.

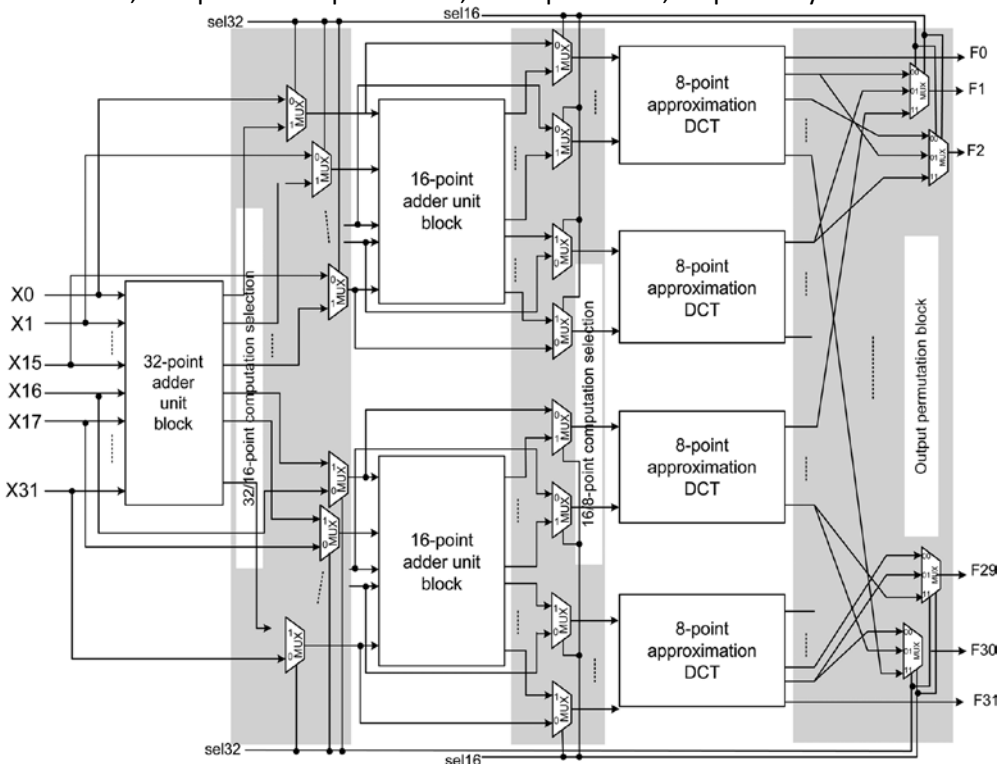

Figure 4 Reconfigurable 32-point DCT Architecture

C. RECONFIGURABLE 64-POINT DCT ARCHITECTURE

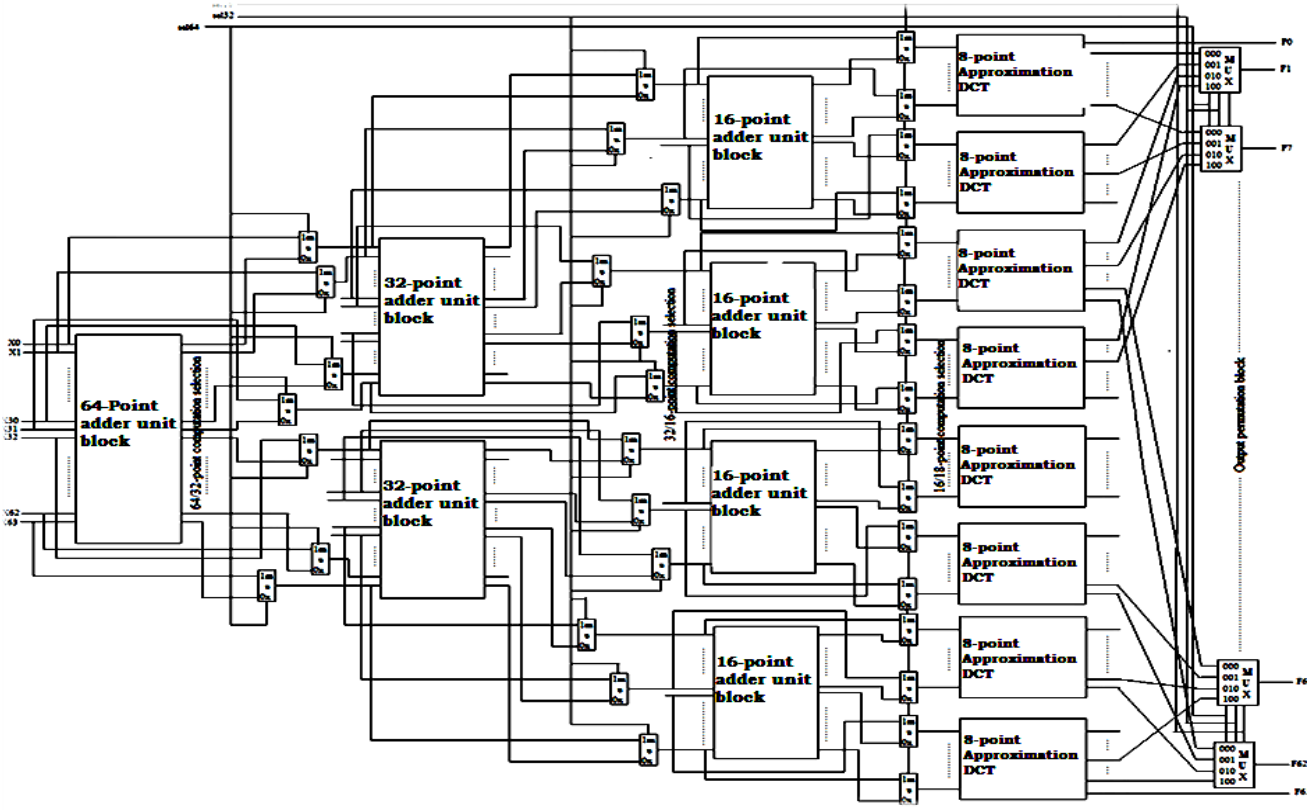

Figure 5 Reconfigurable 64-point DCT Architecture

A reconfigurable design for the computation of 64-point, 32-point, 16-point, and 8-point DCTs is presented in Figure. 5. It performs the calculation of a 64-point DCT or two 32-point DCTs in parallel or four 16-point DCTs in parallel or eight 8-point DCTs in parallel. The architecture is composed of 64-point input adder unit, two 32-point input adder units, four 16-point DCT units, and eight 8-point DCT units. The reconfigurability is achieved by three control blocks composed of 128 2:1 MUXes along with 62 4:1 MUXes. The first control block decides whether the DCT size is of 64 or lower. If Sel64 = 1, the selection of input data is done for the 64-point DCT, otherwise, for the DCTs of lower lengths. The second control block decides whether the DCT size is higher than 32. If Sel32 = 1 the length of the DCT to be computed is higher than 16 (DCT length of 32 or 64), otherwise, the length is 16. The third control block decides whether the DCT size is higher than 16. If Sel16 = 1 the length of the DCT to be computed is higher than 88 (DCT length of 32 or 64), otherwise, the length is 8. The fourth control block is used for the output permutation unit which re-orders the output depending on the size of the selected DCT .Sel64, Sel32 and Sel16 are used as control signals to the 4:1 MUXes. Specifically, for { Sel64, Sel32, Sel16}$_2$ equal to {000}, {001}, {011} or {111} the 64 outputs correspond to eight 8-point parallel DCTs, four parallel 16-point DCTs, two parallel 32-point DCTs or 64-point DCT, respectively.

IV. COMPARISION OF EXISTING SYSTEMS AND IMPLEMENTED SYSTEM

Table 1 illustrates computation and delay comparison of existing method and implemented architecture. The overall complexity for calculation of different DCT lengths as 16-point, 32-point and 64-point DCT approximations requires 60, 152, and 368 additions, respectively. The implemented structure for DCT requires less number of addition operations and does not involve any shift operations.

Normally, the calculation complexity of N -point DCT is equivalent to N (log2N-(1/4)) additions. Since, for the calculation of higher size DCTs of different lengths, the design of scalable and regular structures is used, the computation time is found to be (log2N) TA where TA is the addition-time.

Table 1 Comparison of existing system and implemented System

PARAMETER	EXISTING SYSTEM	IMPLEMENTED SYSTEM
Computational complexity is low	22 adders for 8-point 72 adders for 16-point 160 adders for 32-point 368 adders for 64-point	22 adders for 8-point 60 adders for 16-point 152 adders for 32-point 384 adders for 64-point
Delay is less	22 adders for 8-point 72 adders for 16-point 160 adders for 32-point 368 adders for 64-point	22 adders for 8-point 60 adders for 16-point 152 adders for 32-point 384 adders for 64-point

V. SIMULATION RESULTS

Figure 6. illustrates the output of 8-point Discrete Cosine Transform Architecture

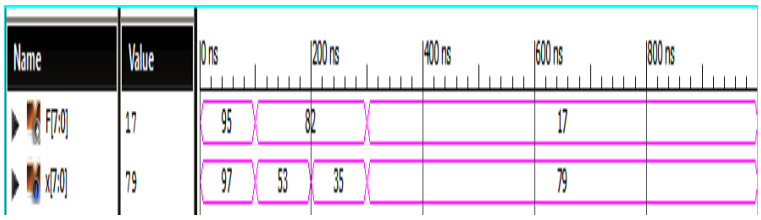

Figure 6: 8-point DCT

Figure 7. illustrates the output of 16-point Discrete Cosine Transform Architecture

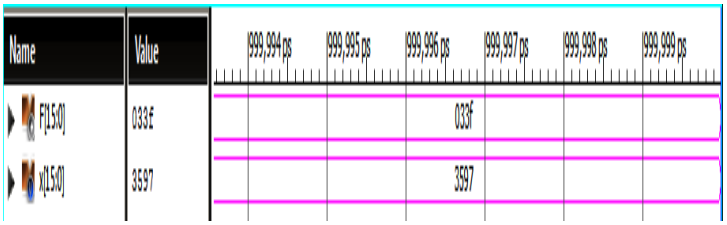

Figure 7: 16-point DCT

Figure 8. illustrate the output of 16-point Reconfigurable architecture of Discrete Cosine Transform

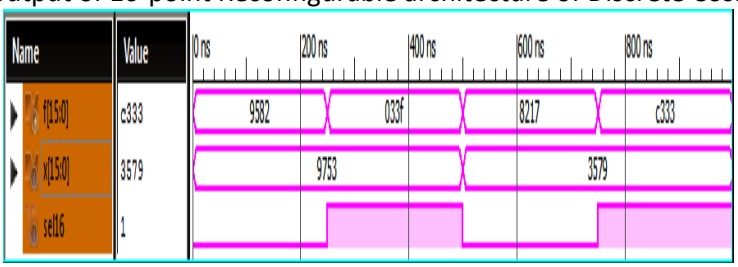

Figure 8: 16-point reconfigurable architecture DCT

Figure 9. illustrate the output of 32-point Reconfigurable Architecture of Discrete Cosine Transform

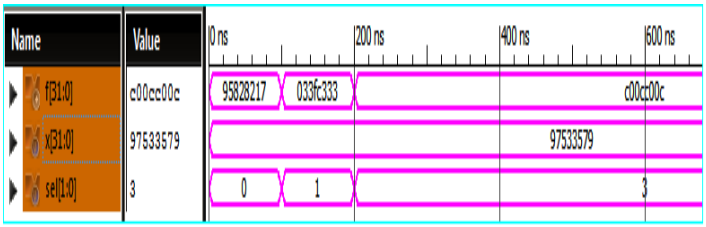

Figure 9: 32-point reconfigurable architecture DCT

Figure 10. illustrate the output of 64-point Reconfigurable Architecture of Discrete Cosine Transform

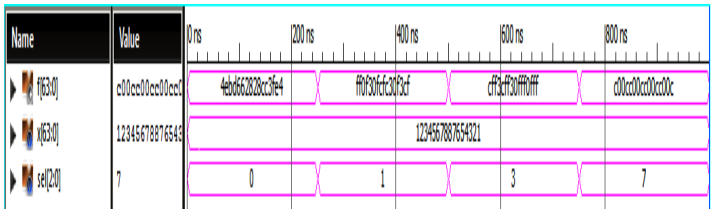

Figure 10: 64-point reconfigurable architecture DCT

VI. CONCLUSION AND FUTURE SCOPE

In this paper, Reconfigurable 32-point DCT Architecture It could be configured for computation 32-point DCT (or) for parallel computation of two 16-point DCTs (or) for parallel computation of four 8-point DCTs and also Reconfigurable 64-point DCT, it could be configured for computation of 64-point DCT (or) for parallel computation of two 32-point DCTs (or) for parallel computation of four 16-point DCTs (or) for parallel computation of eight 8-point DCTs architectures are implemented by using an existing 8-point approximation DCT. This architectures are simulated and synthesized by Xilinx 14.2 tool.

Future enhancement can be implementation of Reconfigurable 128-point DCT architecture, it could be configured for computation of 128-point DCT (or) for parallel computation of two 64-point DCTs (or) for parallel computation of four 32-point DCTs (or) for parallel computation of eight 16-point DCTs (or) for parallel computation of sixteen 8-point DCTs architectures.

VII. REFERENCES

1. Maher Jridi Ayman Alfalou and Pramod Kumar Meher, "A Generalized Algorithm and Reconfigurable Architecture for Efficient and Scalable Orthogonal Approximation of DCT" IEEE Trans. Circuits and systems , vol. 62, no. 2, february 2015
2. M. Shams, A. Chidanandan,W. Pan, and M. A. Bayoumi, "NEDA: A low-power high-performance DCT architecture," *IEEE Trans. Signal Process.*, vol. 54, no. 3, pp. 955–964, 2006.
3. C. Loeffler, A. Lightenberg, and G. S. Moschytz, "Practical fast 1-D DCT algorithm with 11 multiplications," in *Proc. Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, May 1989, pp. 988–991.
4. M. Jridi, P. K. Meher, and A. Alfalou, "Zero-quantised discrete cosine transform coefficients prediction technique for intra-frame video encoding," *IET Image Process.*, vol. 7, no. 2, pp. 165–173, Mar. 2013.
5. F. M. Bayer and R. J. Cintra, "DCT-like transform for image compression requires 14 additions only," *Electron. Lett.*, vol. 48, no. 15, pp. 919–921, Jul. 2012.
6. R. J. Cintra and F. M. Bayer, "A DCT approximation for image compression," *IEEE Signal Process. Lett.*, vol. 18, no. 10, pp. 579–582, Oct. 2011.
7. S. Bouguezel, M. Ahmad, and M. N. S. Swamy, "Low-complexity 8 8 transform for image compression," *Electron. Lett.*, vol. 44, no. 21,pp. 1249–1250, Oct. 2008.
8. T. I. Haweel, "A new square wave transform based on the DCT," *Signal Process.*, vol. 81, no. 11, pp. 2309–2319, Nov. 2001.
9. G. J. Sullivan, J.-R. Ohm,W.-J.Han, and T.Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.
10. F. Bossen, B. Bross, K. Suhring, and D. Flynn, "HEVC complexity and implementation analysis," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 22, no. 12, pp. 1685–1696, 2012.
11. Alfalou, C. Brosseau, N. Abdallah, andM. Jridi, "Assessing the performance of a method of simultaneous compression and encryption of multiple images and its resistance against various attacks," *Opt. Express*, vol. 21, no. 7, pp. 8025–8043, 2013.
12. R. J. Cintra, F. M. Bayer, and C. J. Tablada, "Low-complexity 8-point DCT approximations based on integer functions," *Signal Process.*, vol. 99, pp. 201–214, 2014.
13. U. S. Potluri, A.Madanayake, R. J. Cintra, F.M. Bayer, S. Kulasekera, andA. Edirisuriya, "Improved 8-point approximate DCT for image and video compression requiring only 14 additions," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 61, no. 6, pp. 1727–1740, Jun. 2014.
14. S. Bouguezel, M. Ahmad, and M. N. S. Swamy, "A novel transform for image compression," in *Proc. 2010 53rd IEEE Int. Midwest Symp. Circuits Syst. (MWSCAS)*, pp. 509–512.
15. K. R. Rao and N. Ahmed, "Orthogonal transforms for digital signal processing," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process. (ICASSP)*, Apr. 1976, vol. 1, pp. 136–140.
16. Z. Mohd-Yusof, I. Suleiman, and Z. Aspar, "Implementation of two dimensional forward DCT and inverse DCT using FPGA," in *Proc. TENCON 2000*, vol. 3, pp. 242–245.

17. "USC-SIPI image database," Univ. Southern California, Signal and Image Processing Institute [Online]. Available: http://sipi.usc.edu/ database/, 2012

18. Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: From error visibility to structural similarity," *IEEE Trans. Image Process.*, vol. 13, no. 4, pp. 600–612, 2004.

19. Z. Wang and A. C. Bovik, "A universal image quality index," *IEEE Signal Process. Lett.*, vol. 9, no. 3, pp. 81–84, 2002.

AUTHOR DETAILS

Meesala Shiva Kumar received B.Tech Degree in Electronics and Communication Engineering from ACE Engineering College, Ghatkesar, R.R Dist Affiliated to Jawaharlal Nehru Technological University and pursuing M.Tech in Digital Systems and Computer Electronics in Sreenidhi Institute of Science and Technology, Yamnampet, R.R Dist, Affiliated to Jawaharlal Nehru Technology University.