Design of fully scalable reconfigurable parallel architecture for the computation of approximate DCT

T. Kavitha[1], S.V.R. Manimala[2]

[1] Associate Professor, Department of ECE, MVSR Engineering College, Hyderabad, India

[2] Associate Professor, Department of ECE, MVSR Engineering College, Hyderabad, India

ABSTRACT

This paper presents a fully scalable reconfigurable parallel architecture for the computation of approximate DCT based on the algorithm. One uniquely interesting feature of the existing design is that it could be configured for the computation of 32-point DCTs for parallel computation of two 16-pointDCTs, four 8-point DCTs. We have proposed the computation of 64-point DCTs for parallel computation of two 32-point DCTs, four 16-point DCTs and eight 8-point DCTs. The Reconfigurable Architecture for 64-point DCT is simulated and synthesized by Xilinx 13.2 tool.

Index Terms: DCT approximation, Discrete Cosine Transform (DCT)

Introduction

The Discrete cosine transform (DCT) is popularly used in image and video compression. Since the DCT is computationally intensive, several algorithms have been proposed in the literature to compute it efficiently [1]–[3]. Recently, significant work has been done to derive approximate of 8-point DCT for reducing the computational complexity [4]–[9].The main objective of the approximation algorithms is to get rid of multiplications which consume most of the power and computation-time, and to obtain meaningful estimation of DCT as well. Haweel [8] has proposed the signed DCT (SDCT) for 8 blocks where the basis vector elements are replaced by their sign, i.e, 1. Bouguezel-Ahmad-Swamy (BAS) have proposed a series of methods. They have provided a good estimation of the DCT by replacing the basis vector elements by 0, 1/2, 1 [7]. In the same vein, Bayer and Cintra [5], [6] have proposed two transforms derived from 0 and 1 as elements of transform kernel, and have shown that their methods perform better than the method in [7], particularly for low- and high-compression ratio scenarios. The need of approximation is more important for higher-size DCT since the computational complexity of the DCT grows nonlinearly. On the other hand, modern video coding standards such as high efficiency video coding (HEVC) [10] uses DCT of larger block sizes (up to 32x32) in order to achieve higher compression ratio. But, the extension of the design strategy used in H264 AVC for larger transform sizes, such as 16-point and 32-point is not possible. Besides, several image processing applications such as tracking and simultaneous compression and encryption require higher DCT sizes.

II.PROPOSED DCT APPROXIMATION

To find these approximated sub matrices we take the smallest size of DCT matrix to terminate the approximation procedure to 8, since 4-pointDCT and 2-point DCT can be implemented by adders only. Consequently, a good approximation of C_N, where N is an integral power of two, for N$\geq$8, leads to proper approximations of C_8 and S_8.For approximation of C_8 we can choose the 8-pointDCT.The first one is to approximate $S_8$by null matrix, which implies all even-indexed DCT coefficients are assumed to be zero. The transform obtained by this approximation is far from the exact values of even-indexed DCT coefficients, and the odd coefficients do not contain any information. The second solution is obtained by approximating S_8 by an 8x8 matrix where each row contains one 1 and all other elements are zeros. Here, elements equal to 1 correspond to the maximum of elements of the exact DCT in each row. The approximate transform in this case is closer to the exact DCT than the solution obtained by null matrix. The

Corresponding author: T. Kavitha | E-mail: kavitati2k3@yahoo.com

third solution consists of approximation of S_8 by $C\hat{}_8$. Since C_8 as well as S_8 are sub-matrices of C_{16} and operate on matrices generated by sum and differences of pixel pairs at distance of 8, approximation of S_8 by $C\hat{}_8$ has attractive computational properties: regularity of the signal-flow graph, orthogonality. Since $C\hat{}_8$ is orthogonalizable and good compression efficiency, other than scalability and scope for reconfigurable implementation. We have not done exhaustive search of all possible solutions. So there could be other possible low-complexity implementation of $S\hat{}_8$. But other solutions are not expected to have the potential for reconfigurablity what we achieve by replacement of $S\hat{}_8$ by $C\hat{}_8$. Based on this third possible approximation of S_8.
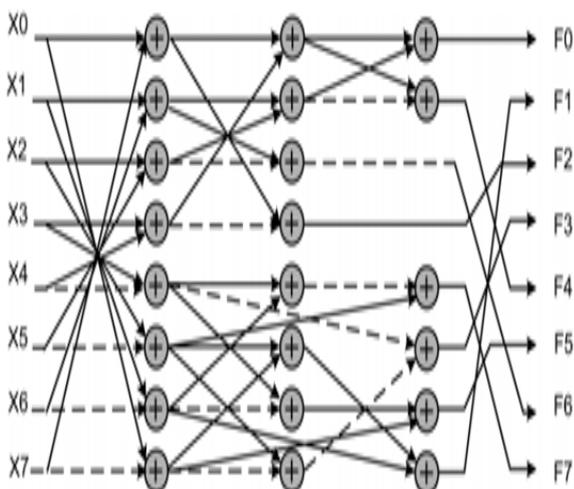
Fig1. Signal flow graph (SFG) of ($C\hat{}_8$). Dashed arrows represent multiplications by 1.

III. SCALABLE AND RECONFIGURABLE ARCHITECTURE FOR DCT COMPUTATION

In this section, we discuss the proposed scalable architecture for the computation of approximate DCT of N = 16 and 32.We have derived the theoretical estimate of its hardware complexity and discuss the reconfiguration scheme.

A. Proposed Scalable Design

The basic computational block of algorithm for the proposed DCT approximation, $C\hat{}_8$ is given in [6]. The block diagram of the computation of DCT based on $C\hat{}_8$ is shown in Fig. 1. For a given input sequence {X (n)}, n belongs to [0, N-1], the approximate DCT coefficients are obtained by $C\hat{}_N$ $\cdot X^t$. An example of the block diagram of $C\hat{}_{16}$ is illustrated in Fig. 2, where two units for the computation of $C\hat{}_8$ are used along with an input

adder unit and output permutation unit. The functions of these two blocks are shown respectively in(8) and (6).Note that structures of 16-point DCT of Fig. 2 could be extended to obtain the DCT of higher sizes. For example, the structure for the computation of 32-point DCT could be obtained by combining a pair of 16-point DCTs with an input adder block and output permutation block.
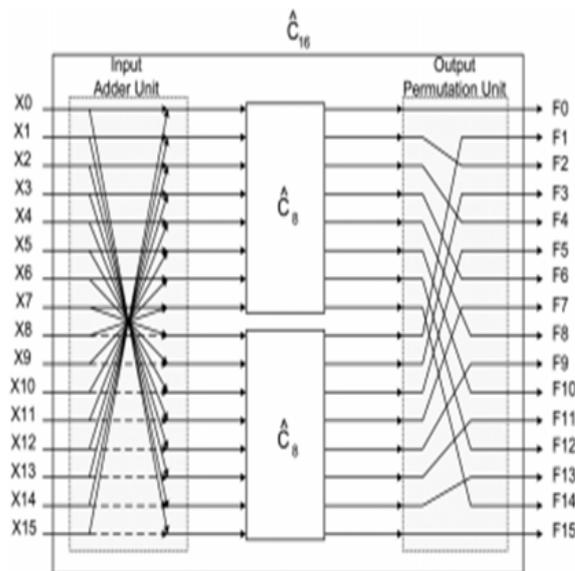
Fig. 2. Block diagram of the proposed DCT for N = 16 ($C\hat{}_n$).
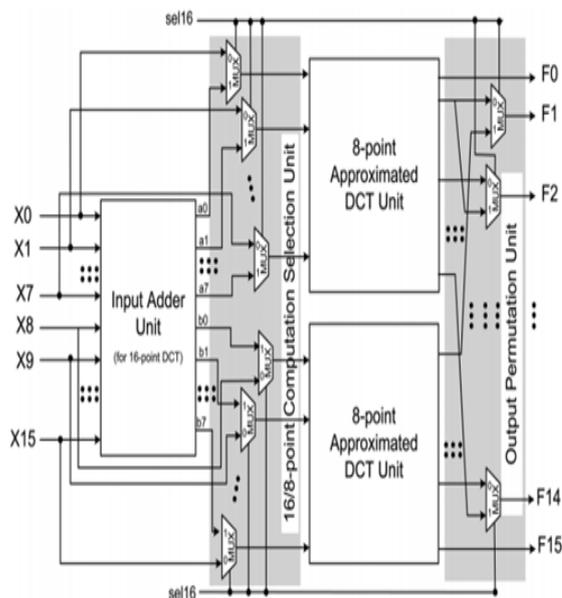
Fig. 3. Proposed reconfigurable architecture for approximate DCT of lengths N = 8 and 16.

B.32-Point reconfigurable architecture for approximate DCT

As specified in the recently adopted HEVC [10], DCT of different lengths such as N = 8, 16, 32 are required to be used in video coding applications.

Therefore, a given DCT architecture should be potentially reused for the DCT of different lengths instead of using separate structures for different lengths. We propose here such reconfigurable DCT structures which could be reused for the computation of DCT of different lengths. The reconfigurable architecture for the implementation of approximated 16-point DCT is shown in Fig. 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 (Fig.3) 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 Fig. 3 allows the calculation of a 16-point DCT or two 8-point DCTs in parallel.
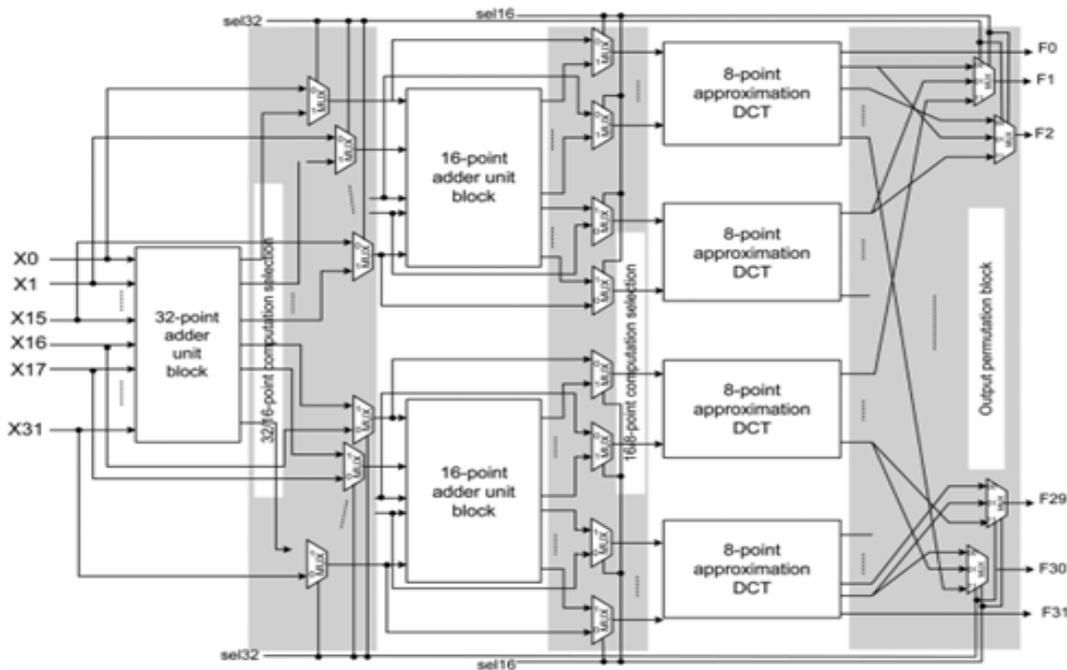
Fig. 4. Proposed reconfigurable architecture for approximate DCT of lengths N = 8, 16 and 32

A reconfigurable design for the computation of 32-, 16-, and 8-point DCTs is presented in Fig. 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} 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. Note that the throughput is of 32 DCT coefficients per cycle irrespective of the desired transform size.
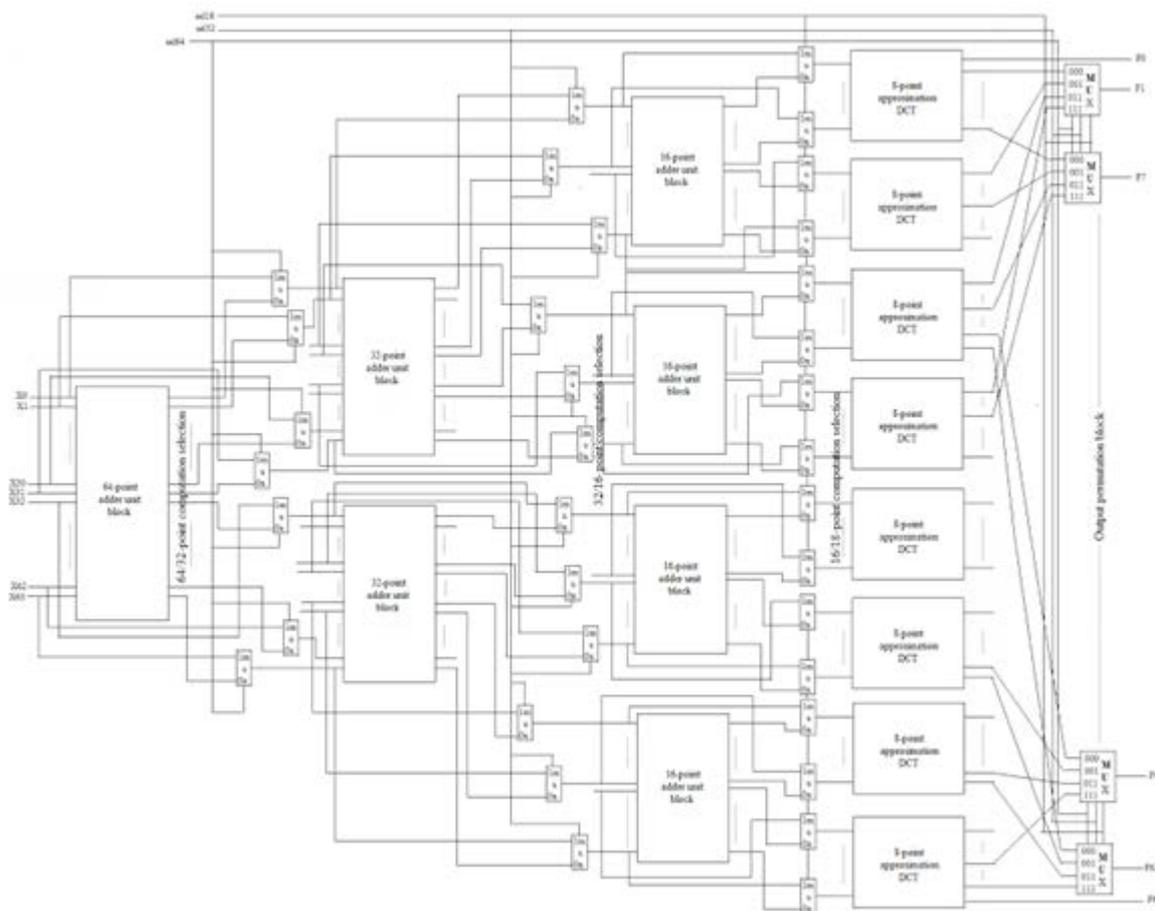
Fig. 5. Proposed reconfigurable architecture for approximate DCT of lengths N = 8, 16, 32 and 64.

A reconfigurable design for the computation of 64-, 32-, 16-, and 8-point DCTs is presented in Fig. 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 units, 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 624:1 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} 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. Note that the throughput is of 64 DCT coefficients per cycle irrespective of the desired transform size.

IV.SIMULATION RESULTS
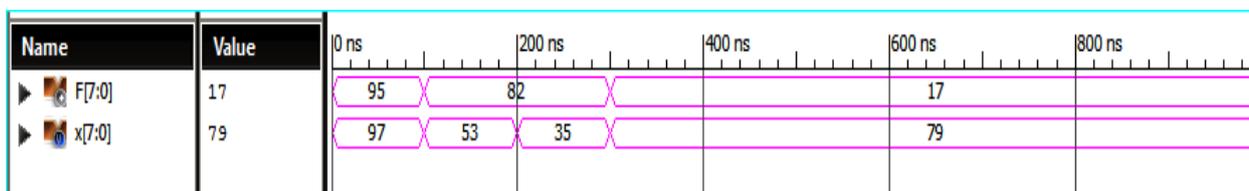
Figure 6: 8-point DCT
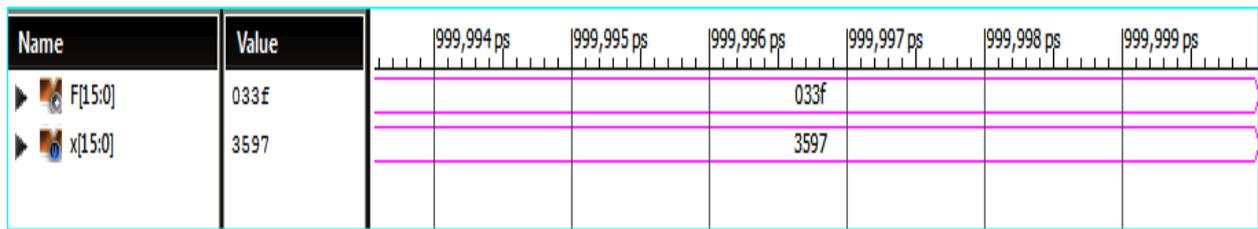
Figure 7: 16-point DCT
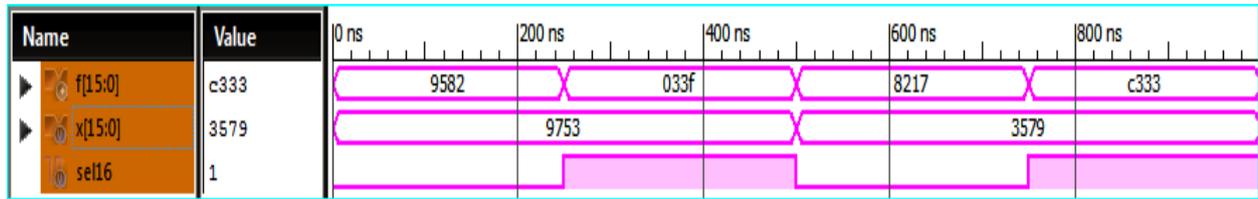
Figure 8: 16-point reconfigurable architecture DCT
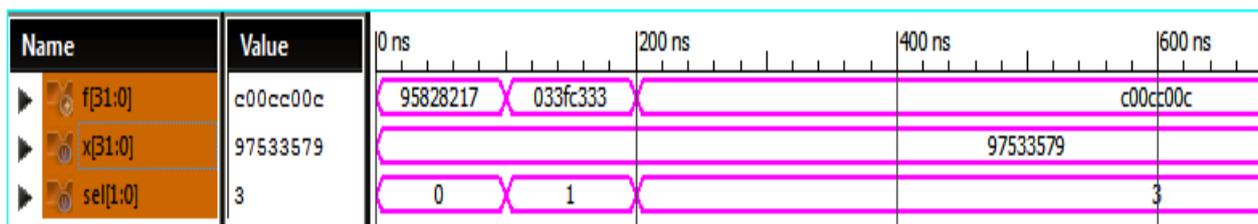
Figure 8: 32-point reconfigurable architecture DCT
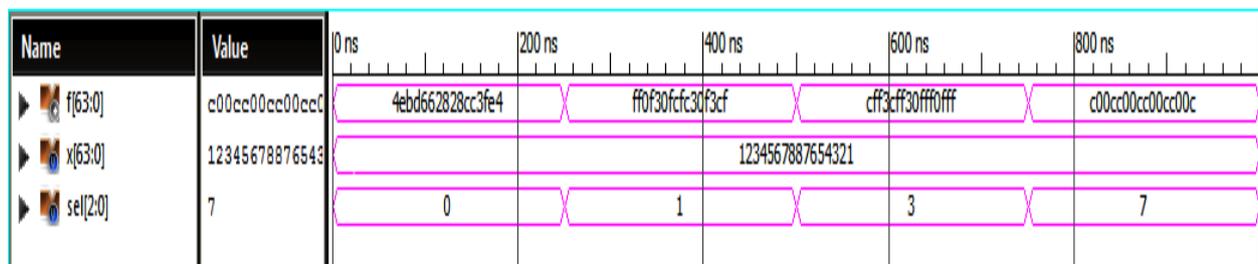
Figure 8: 64-point reconfigurable architecture DCT

Table I Comparison between 14 bit adder and 22 bit adder

	Bit Size	No. of Slices	No. of LUT's
DCT Using 14 Adders	8	4	7
	16	4	7
	16r	21	39
	32r	79	143
	64r	168	307
DCT Using 22 Adders	8	6	10
	16	8	15
	16r	28	51
	32r	89	160
	64r	244	434

V. CONCLUSION

In this paper, we have proposed a recursive algorithm to obtain orthogonal approximation of DCT where approximate DCT of length N could be derived from a pair of DCTs of length N/2 at the cost of N additions for input preprocessing. The proposed approximated DCT has several advantages, such as of regularity, structural simplicity, lower-computational complexity, and scalability. We have proposed a fully scalable reconfigurable architecture for approximate DCT computation where the computation of 64-point DCT could be configured for parallel computation of two 32-point DCTs, four 16-point DCTs and eight 8-point DCTs. The Reconfigurable Architecture for 64-point DCTis simulated and synthesized by Xilinx 13.2 tool.

VI.REFERENCES

1. 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.
2. 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.
3. 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.
4. S. Bouguezel, M. O. Ahmad, and M. N. S. Swamy, "Binary discrete cosine and Hartleytransforms," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 60, no. 4, pp. 989–1002, Apr. 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 8x8 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," *SignalProcess.*, vol. 81, no. 11, pp. 2309–2319, Nov. 2001.
9. V. Britanak, P. Y. Yip, and K. R. Rao, *Discrete Cosine and Sine Transforms: General Properties, Fast Algorithms and Integer Approximations*. London, U.K.: Academic, 2007.
10. G. J. Sullivan, J.-R. Ohm, W.-J. Han, and T. Wiegand, "Overview of the high efficiency video coding (HEVC) standard," *IEEE Trans. CircuitsSyst. Video Technol.*, vol. 22, no. 12, pp. 1649–1668, Dec. 2012.

AUTHOR'S Biography

T. Kavitha *is an Associate Professor at M.V.S.R.Engineering college, Hyderabad in ECE Department. She received her B.E degree in Electronics and Communication Engineering from MVSREngineering college, Hyderabad and M.Tech degree in Digital Systems and Computer Electronics from J.N.T.U.,Hyderabad. Her research interest is Signal Processing and Communications.*
Email: tkavitha_ece@mvsrec.edu.in

S.V.R.manimala is an Associate Professor at M.V.S.R.Engineering college, Hyderabad in ECE Department. She received her B.E degree in Electronics and Communication Engineering from Sir C R Reddy college of Engineering , Elluru and M.Tech degree in Digital Systems and Computer Electronics from J.N.T.U.,Hyderabad. Her research interest is Sparse Signal Processing.
Email: svrm.73@gmail.com