Scientific
Research
Publishing

FPGA Implementation of Approximate 2D Discrete Cosine Transforms

M. Thiruveni Raguraman, D. Shanthi Saravanan

Department of ECE, Department of CSE, PSNA College of Engineering and Technology, Dindigul, India
Email: thiruveniraguraman@gmail.com, dshan71@gmail.com

Abstract

Discrete cosine transform (DCT) is frequently used in image and video signal processing due to its high energy compaction property. Humans are able to perceive and identify the information from slightly erroneous images. It is enough to produce approximate outputs rather than absolute outputs which in turn reduce the circuit complexity. Numbers of applications like image and video processing need higher dimensional DCT algorithms. So the existing architectures of one dimensional (1D) approximate DCTs are reviewed and extended to two dimensional (2D) approximate DCTs. Approximate 2D multiplier-free DCT architectures are coded in Verilog, simulated in Modelsim to evaluate the correctness, synthesized to evaluate the performance and implemented in virtexE Field Programmable Gate Array (FPGA) kit. A comparative analysis of approximate 2D DCT architectures is carried out in terms of speed and area.

Keywords

Discrete Cosine Transform, Energy Compaction, Field Programmable Gate Array, Dimension

1. Introduction

The increase in use of computers increases the use of digital signal processing (DSP). In DSP, three domains are used to represent the signals. They are time domain/spatial domain (for one-dimensional signals for multidimensional signals respectively), frequency domain, and wavelet domains. Signal can be represented in any one of the domain which represents the essential characteristics of the signal. Frequency domain also called spectrum- or spectral analysis makes partitioning of spectral components to propose a small and meaningful form of signal representation. There are many frequency domain transformations. Due to its strong "energy compaction" property DCT is frequently used in signal and image processing. It is also used in a multitude of compression standards. For multimedia applications, video processing systems such as High Efficiency Video Coding (HEVC)

need fast and compact blocks. Approximation of DCT transform becomes efficient by the vast improvement in fast algorithms.

A Discrete Cosine Transform (DCT) [1] gives a finite number of points in terms of addition of cosine functions oscillating at different frequencies. Discrete Fourier transforms (DFT) using only real numbers becomes DCT, a Fourier-related transform.

DCT can be expressed as

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

where

$$e(k) = \begin{cases} \dfrac{1}{\sqrt{2}} & \text{if } k = 0 \\ 1 & \text{otherwise} \end{cases} \tag{2}$$

1D-DCT is utilized for changing one dimensional signal like audio. But image and video signal needs 2D-DCT for its handling. Number of uses requiring higher dimensional DCT calculations are increasing. So much importance is given for the algorithm which can be extended for higher dimensional readily.

The individual product of all dimensions of 1D-DCT is used to produce multidimensional DCT. For example, the product of 1D-DCT along the rows and columns form the 2D-DCT of an image. The computation of 2D-DCT from 1D-DCTs across all dimension is known as a row-column algorithm.

The expression for 2D-DCT is given by

$$S(u,v) = \frac{2}{\sqrt{nm}} C(u) C(v) \sum_{y=0}^{m-1} \sum_{x=0}^{n-1} s(x,y) \cos\frac{(2x+1)u\pi}{2n} \cos\frac{(2y+1)u\pi}{2m} \quad u = 0,\cdots,n; v = 0,\cdots,m \tag{3}$$

where

$$c(u) = \begin{cases} \dfrac{1}{\sqrt{2}} & \text{for } u = 0 \\ 1 & \text{otherwise} \end{cases} \tag{4}$$

In conventional DCT, an 8-point 1D- DCT requires 64 multiplications and 56 additions and 8-point 2D-DCT requires 1024 multiplications and 896 additions. It is computation intensive and also occupies more area. So the approximate DCTs are preferred. Humans are able to perceive and identify the information from slightly erroneous images. It is enough to produce approximate outputs rather than absolute outputs.

The idea of this paper is three-fold: First, reviewing architecture of approximate DCTs; Second, extending the architecture of 1D approximate DCT to 2D approximate DCT and third, proposes implementation of 2D approximate DCT in virtexE FPGA; The workflow is shown in **Figure 1**.

2. Methods and Materials

The idea of using the approximation algorithm for DCT is to eliminate computation intensive and power consuming operations like multiplications and also to get significant evaluation of DCT. It is more suitable for large DCTs to reduce the computations of DCT which increases randomly. The available methods are not suitable for extension. But sizes such as 16-point and 32-point DCTs are needed for many image processing applications like biomedical signal processing, satellite communication, etc.

Approximate DCT transforms have been proposed with no cost of multiplication gives better compression performance. Realization of the approximations in digital VLSI hardware requires only additions and subtractions which reduces chip area and power consumption than conventional DCTs transforms. The 8-point approximate DCT manipulation requires only addition and no multiplication. So computational complexity is brought down. A reconfigurable video standard like HEVC uses the best DCT approximation. The transformation matrix cost is equal to the number of arithmetic operations in its computation.
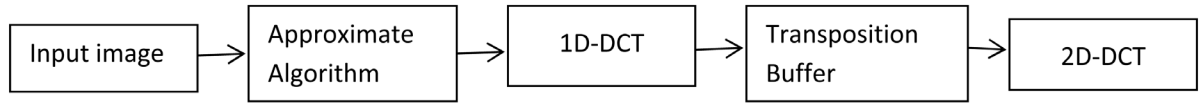
Figure 1. Flow graph of 2D DCT architecture.

The number of reserved coefficients in the transform domain is the main constraint of image compression process. The performance of the DCT approximations is often a trade-off between accuracy and computational density of a given algorithm.

The reduced computational complexity, orthogonality, small error energy extendable of DCT is the main features of approximate DCT.

$$\text{Transformation matrix} = [\text{Diagonal matrix}] \times [\text{low-complexity matrix}]. \tag{5}$$

The diagonal matrix typically includes irrational numbers in the form $1/\sqrt{m}$, where m is a small positive integer. Normally, the irrational numbers in the diagonal matrix requires more computations. Since the elements of the matrix comprise only powers of two $\{0, \pm 1/2, \pm 1, \pm 2\}$, no multiplication is required.

2.1. One-Dimensional Digital Architectures of Approximate DCT

In [2], a low complexity approximate was introduced by Bouguezel *et al.* is shown in **Figure 2** and called BAS-2008 Approximation.

$$C_1 = D_1 \cdot \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 1 & 1/2 & -1/2 & -1 & -1 & -1/2 & 1/2 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 \\ 1/2 & -1 & 1 & -1/2 & -1/2 & 1 & -1 & 1/2 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{6}$$

$C_1 = D_1 \cdot T_1$ gives the mathematical structure where $D_1 = \text{diag}\left(1/\sqrt{8}, 1/\sqrt{4}, 1/\sqrt{5}, 1/\sqrt{2}, 1/\sqrt{8}, 1/\sqrt{4}, 1/\sqrt{5}, 1/\sqrt{2}\right)$. The computation requires only 18 additions and 2 shifts.

An 8-point orthogonal DCT transform proposed by Bouguezel-Ahmad-Swamy in 2011 [3] contains a single parameter "a". It is shown in **Figure 3**.

$$C_2 = D_2 \cdot \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 1 & a & -a & -1 & -1 & -a & a & 1 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 1 & -1 \\ a & -1 & 1 & -a & -a & 1 & -1 & a \end{bmatrix} \tag{7}$$

The mathematical structure is $C_2 = D_2 \cdot T_2$ where $D_2 = \text{diag}\left(1/\sqrt{8}, 1/\sqrt{2}, 1/\sqrt{(4 + 4a^2)}, 1/\sqrt{2}, 1/\sqrt{8}, 1/\sqrt{2}, 1/\sqrt{2}, 1/\sqrt{(4 + 4a^2)}\right)$. The computation requires only 16 additions.

In CB-2011 [4], DCT approximation was obtained by rounding-off the elements of exact DCT matrix. The resulting matrix is orthogonal and contains elements only in $\{0, \pm 1\}$. It own very low arithmetic complexity. The CB-2011 architecture is shown in **Figure 4**.
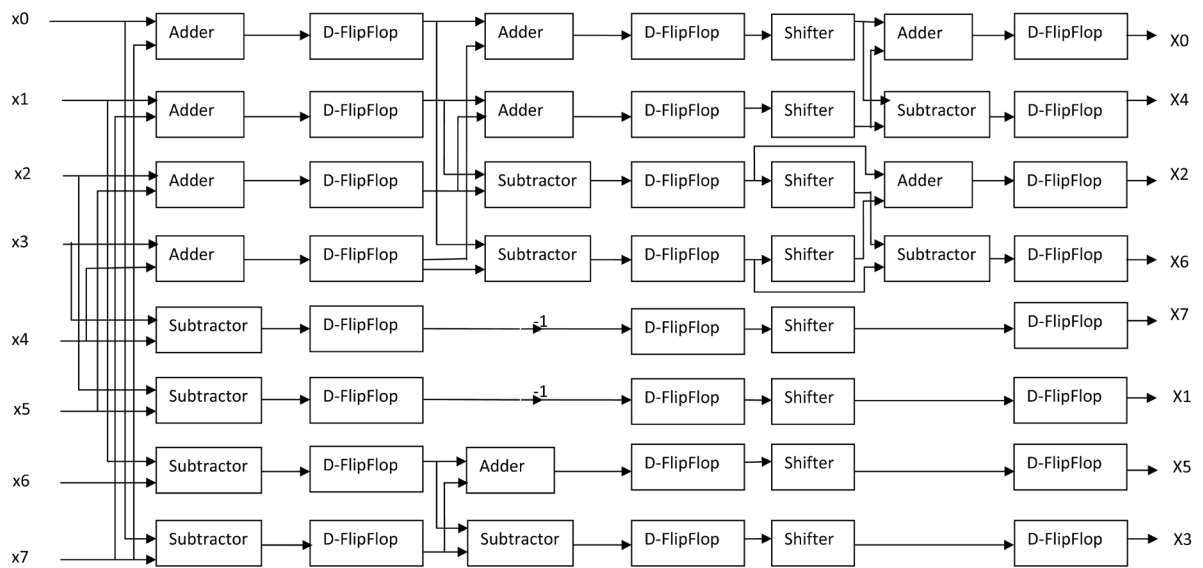
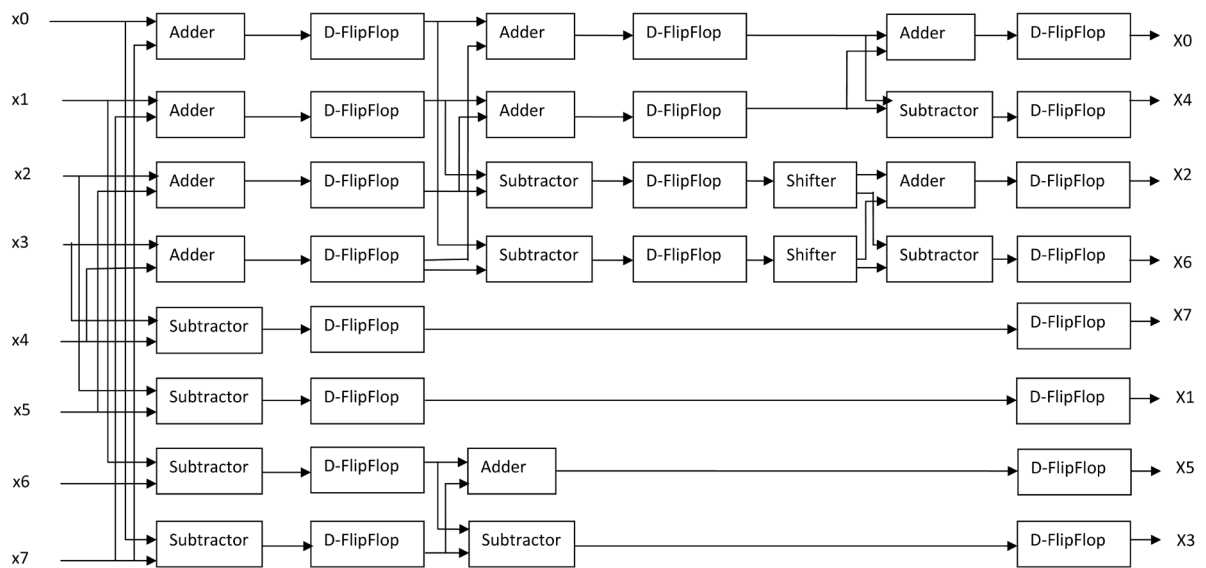**Figure 2.** Digital architecture of BAS-2008 approximate 1D-DCT.

Figure 3. Digital architecture of BAS-2011 approximate 1D DCT.

$$C_3 = D_3 \cdot \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 0 & 0 & -1 & -1 & -1 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 1 & 0 & -1 & -1 & 1 & 1 & 0 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -1 & 0 & 1 & -1 & 0 & 1 & -1 \\ 0 & -1 & 1 & 0 & 0 & 1 & -1 & 0 \\ 0 & -1 & 1 & -1 & 1 & -1 & 1 & 0 \end{bmatrix} \quad (8)$$

The transformation matrix $C_3 = D_3 \cdot T_3$ where $D_3 = \mathrm{diag}\left(1/\sqrt{8}, 1/\sqrt{6}, 1/2, 1/\sqrt{6}, 1/\sqrt{8}, 1/\sqrt{6}, 1/\sqrt{2}, 1/\sqrt{6}\right)$. The computation of matrix requires only 22 additions.
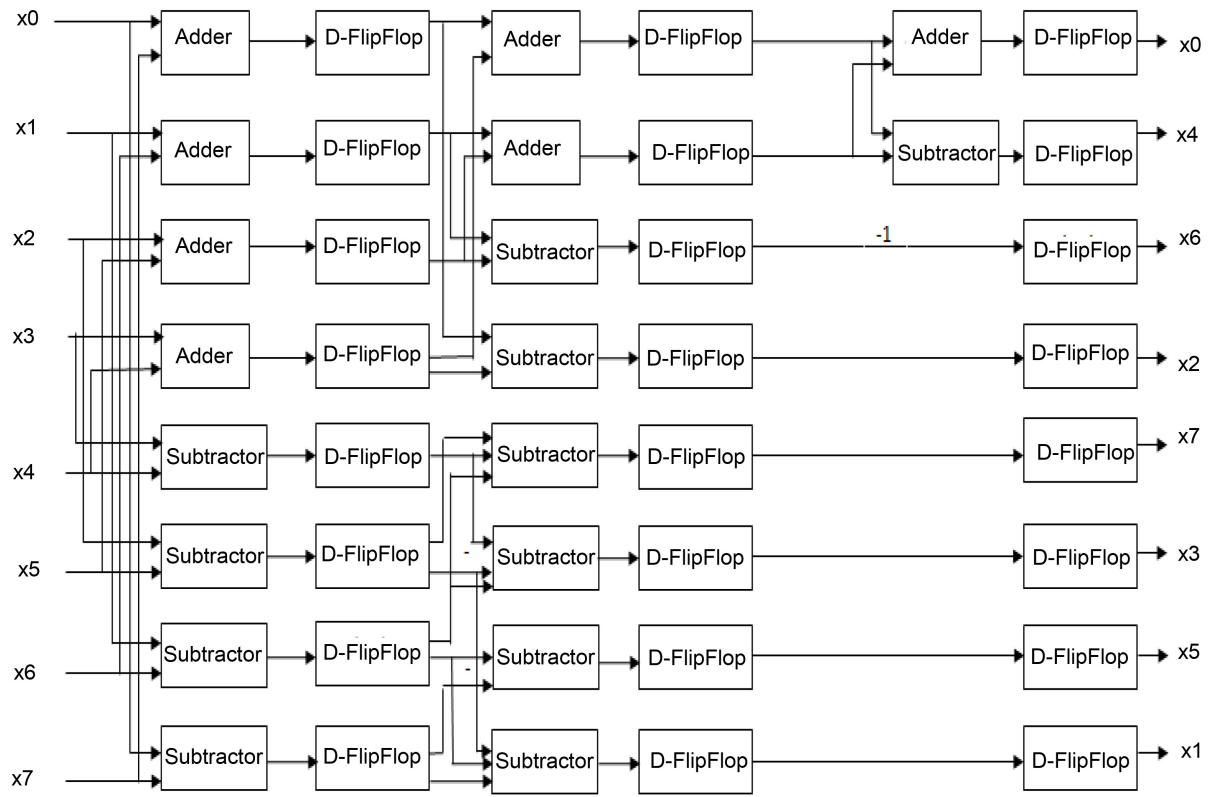
Figure 4. Digital architecture of CB-2011 approximate 1D DCT.

$$C_4 = D_4 \cdot \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 0 & -1 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & -1 & 1 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & -1 & 1 & 0 & 0 & 0 \end{bmatrix} \tag{9}$$

The replacement of CB-2011[5] matrix elements with zeros results in modified CB-2011which is shown in **Figure 5**. The transformation matrix $C_4 = D_4 \cdot T_4$ where $D_4 = \text{diag}\left(1/\sqrt{8}, 1/\sqrt{2}, 1/2, 1/\sqrt{2}, 1/\sqrt{8}, 1/\sqrt{2}, 1/2, 1/\sqrt{2}\right)$. It needs only 14 addition for matrix computation.

In [6], a DCT approximation in **Figure 6** is suitable for radio-frequency (RF) application.

$$C_5 = D_5 \cdot \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 2 & 1 & 1 & 0 & 0 & -1 & -1 & -2 \\ 2 & 1 & -1 & -2 & -2 & -1 & 1 & 2 \\ 1 & 0 & -2 & -1 & 1 & 2 & 0 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 1 & -2 & 0 & 1 & -1 & 0 & 2 & -1 \\ 1 & -2 & 2 & -1 & -1 & 2 & -2 & 1 \\ 0 & -1 & 1 & -2 & 2 & -1 & 1 & 0 \end{bmatrix} \tag{10}$$
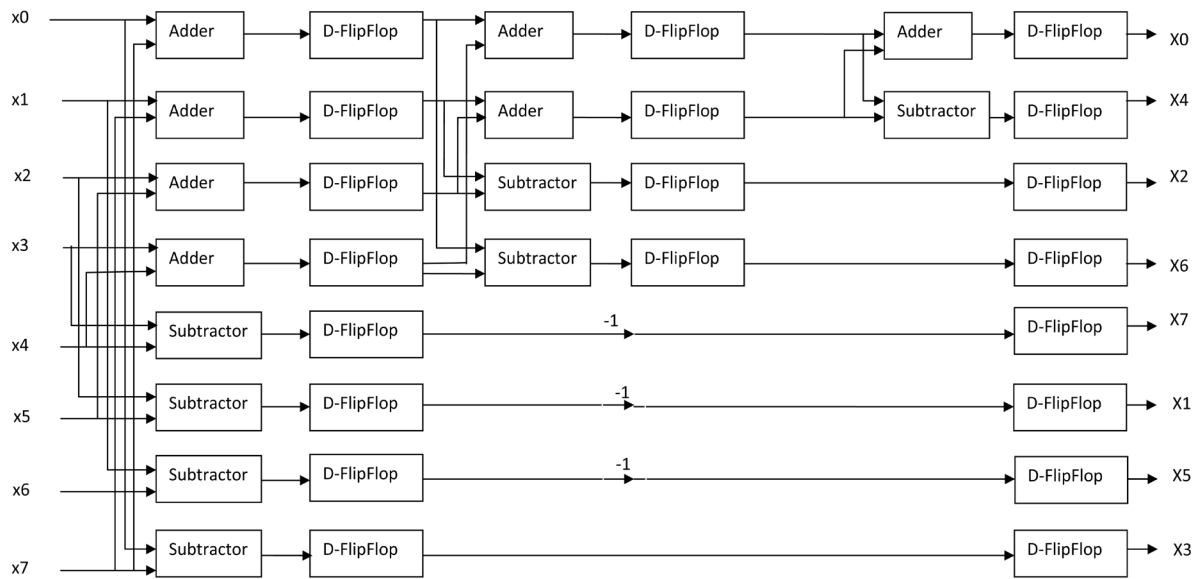
Figure 5. Digital architecture of modified CB-2011 approximate 1D DCT.
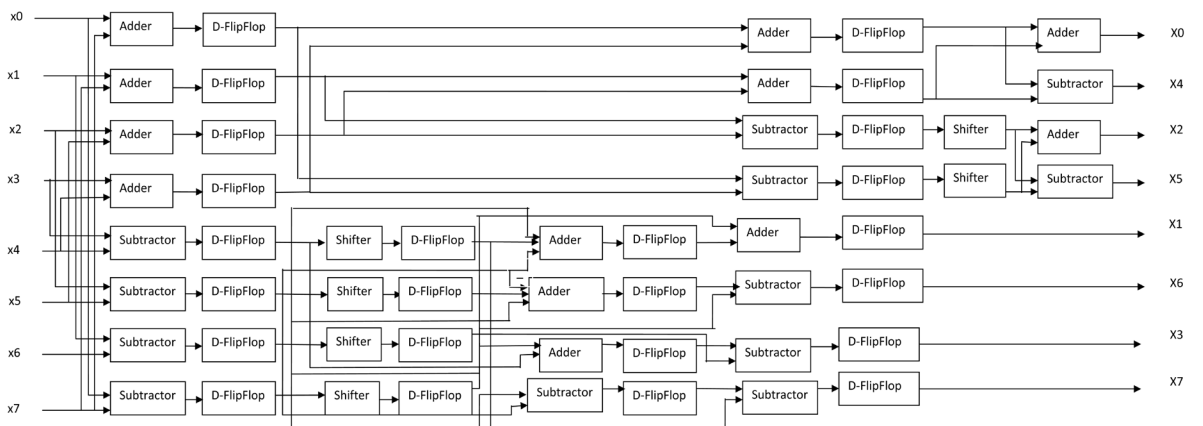

Figure 6. Digital architecture of potluri-2012 approximate 1D DCT.

The transformation matrix $C_5 = D_5 \cdot T_5$ where $D_5 = 1/2 \cdot \mathrm{diag}\left(1/\sqrt{2}, 1/\sqrt{3}, 1/\sqrt{5}, 1/\sqrt{3}, 1/\sqrt{2}, 1/\sqrt{3}, 1/\sqrt{5}, 1/\sqrt{3}\right)$. It requires only 24 additions and 6 shifts for its computation.

In [7], approximate DCT with fewer computations in **Figure 7** describe the cost of a transformation matrix as the quantity of number-crunching operations required for its calculation. The multiplicative complexity is null because of the matrix elements lies in {0, ±1, ±2}.

$$C_6 = D_6 \cdot \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 & -1 & 0 \\ 1 & 0 & 0 & -1 & -1 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \\ 1 & -1 & -1 & 1 & 1 & -1 & -1 & 1 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & -1 & 1 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \end{bmatrix} \tag{11}$$
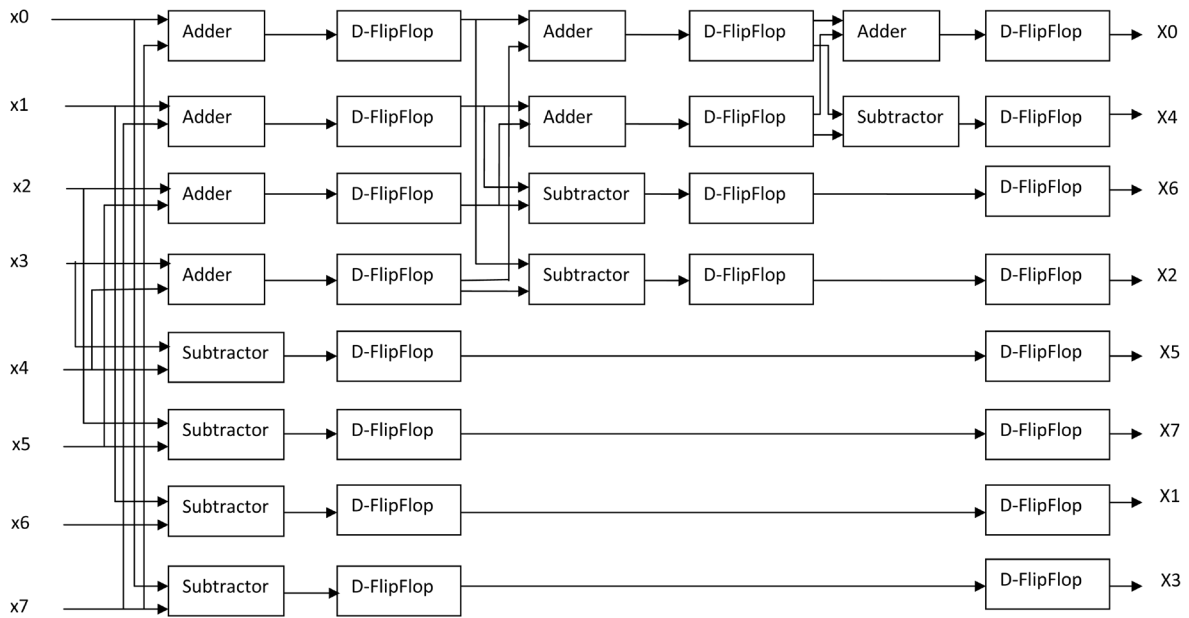
Figure 7. Digital architecture of Potluri-2014 approximate 1D DCT.

The DCT coefficient matrix $C_6 = D_6 \cdot T_6$ where $D_6 = \mathrm{diag}\left(1/\sqrt{8}, 1/\sqrt{2}, 1/2, 1/\sqrt{2}, 1/\sqrt{8}, 1/\sqrt{2}, 1/2, 1/\sqrt{2}\right)$. The computation requires only 14 additions.

The butterfly structure [8], is modified into a low-complexity approximate DCT in **Figure 8**. The redundant computations are shared or removed in DCT matrix. The matrix elements lies in {0, ±1} makes the multiplicative complexity null.

$$C_7 = D_7 \cdot \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 & -1 & -1 & 0 & 1 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 & -1 & -1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & -1 \end{bmatrix} \tag{12}$$

The transformation matrix $C_7 = D_7 \cdot T_7$ where $D_7 = 1/2 \cdot \mathrm{diag}(1,1,1,1,1,1,1,1)$. The computation of above matrix requires only 12 additions.

Table 1 shows the number of additions, multiplications and bit-shift operations required for the existing transforms. The approximate DCT by Vaithyanathan *et al*. [8] saves computation by 16.7% than Modified Cintra and Bayer [5] and Potluri [7], 33.3%, 66% and 83% than Bouguezel *et al*. [3], Bouguezel *et al*. [2] and Modified Cintra and Bayer [4] respectively.

2.2. Two Dimensional Approximate Transform

For real-time implementation of approximate algorithms, the proposed digital architectures are custom designed. The 1-D approximate DCT block is implemented using suitable algorithm chosen from the existing architectures [9]-[11]. The row wise transformation of the input image, followed by a column wise transformation of the intermediate result forms the 2D-DCT transformation as shown in **Figure 9**.

Multidimensional DCT from 1D DCT

➢ row wise transformation of the input image of1D DCT
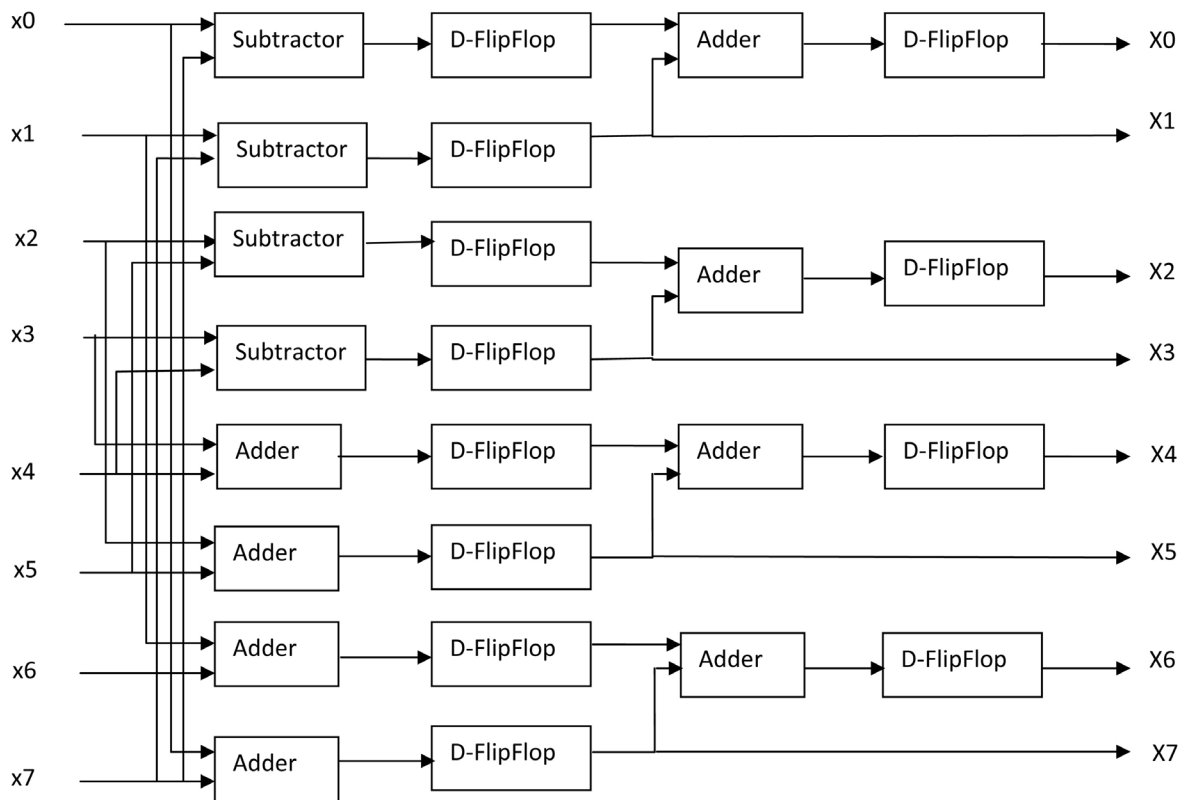➢ Column wise transformation of resultant row wise transformation above.

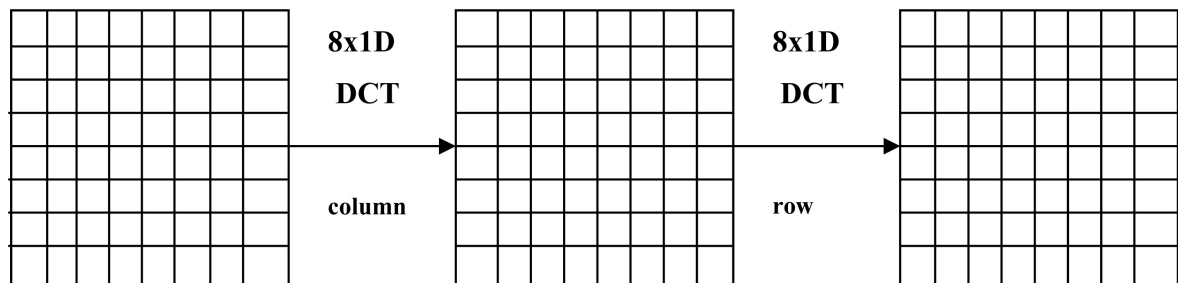**Figure 8.** Digital architecture of Vaithyanathan 1D approximate DCT.

Figure 9. Computation of 2D DCT.

Table 1. Computation complexity of approximate DCTs.

DCT	Multiplications	Additions	Shifts	Total
Conventional	64	56	0	120
BAS-2008	0	18	2	20
BAS-2011	0	16	0	16
CB-2011	0	22	0	22
Modified CB-2011	0	14	0	14
Potluri-2012	0	24	6	30
Potluri-2014	0	14	0	14
Vaithyanahan-2014	0	12	0	12

➢ Or alternatively vertical to Horizontal.

The row and column wise transforms can be any of the existing DCT approximations. In other words, there is no restriction for both row and column wise transforms to be the same. However, for simplicity, identical transforms for both steps are adopted. It employs two parallel realizations of DCT approximation blocks, as shown in **Figure 10**.

2.3. Transposition Buffer

Between the approximate DCT blocks a real-time row parallel transposition buffer circuit is required. Such block ensures data ordering for converting the row transformed data from the first DCT approximation circuit to a transposed format as required by the column transform circuit. The transposition buffer block is detailed in **Figure 11**.

3. FPGA Implementations and Discussions

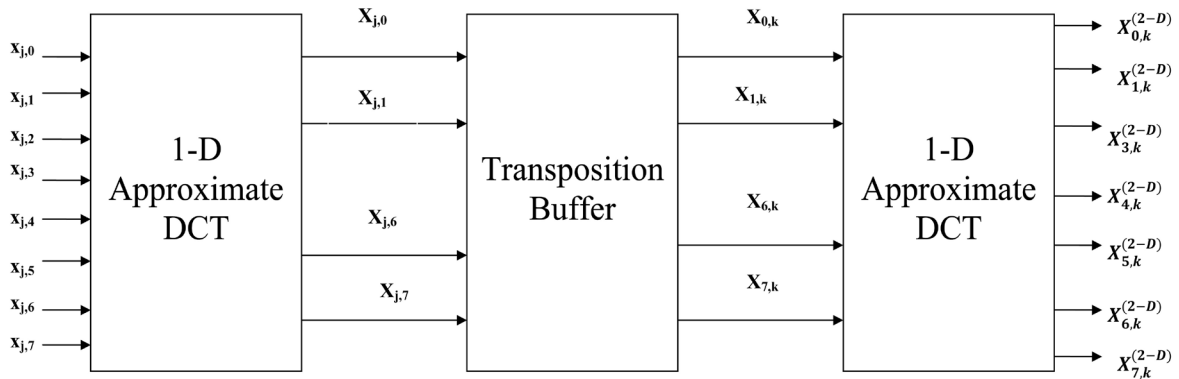The 2D approximate DCT architectures are prototyped on a FPGA for analysis and tested using on-chip hardware.

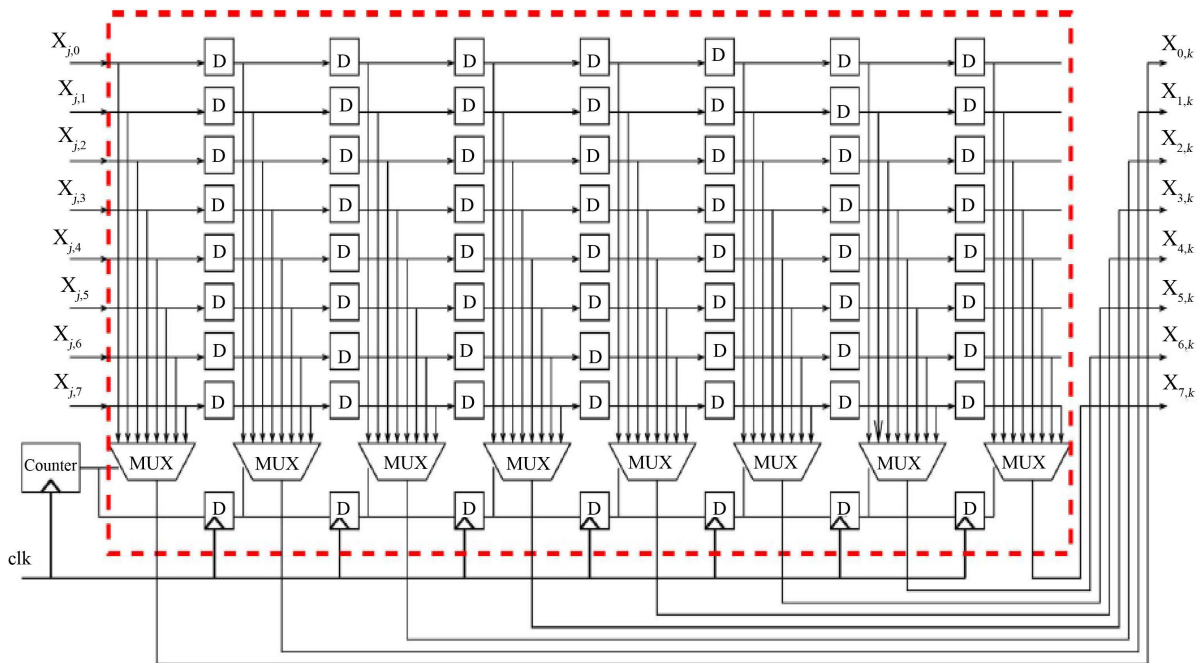

Figure 10. Block diagram of 2D DCT.

Figure 11. Detailed circuit of the transposition buffer block.

The architectures are designed for digital realization with verilog code using modelsim and Xilinx System with synthesis options. The proposed 2D approximate DCT architectures were physically realized on Xilinx VirtexE device. Comparison of hardware resource consumption and speed of architectures are analysed by implementing them in Xilinx VirtexE device. **Table 2** gives the overall comparison analysis. For various approximate DCTs, the No. of slices occupied is shown in **Figure 12**, No. of Flip-flop required is shown in **Figure 13**, No. of LUTs needed is shown in **Figure 14** and comparison of maximum frequency for different 2-D DCT is shown by **Figure 15**.

The approximate DCT by vaithyanathan *et al*. [8] reduces the computational complexity by 40.11% , 24.83%, 38.42%, 34.34%, 45.5% and 34.34% than Bouguezel *et al*. [2], Bouguezel *et al*. [3], Cintra and Bayer [4], Modified Cintra and Bayer [5], Potluri [6] and Potluri [7] respectively. It also increase the maximum frequency of operation by 24.7%, 23.5%, 12.36%, 10.51%, 13.29% and 4.91% than Bouguezel *et al*. [2], Bouguezel *et al*. [3], Cintra and Bayer [4], Modified Cintra and Bayer [5], Potluri [6] and Potluri [7] respectively.

From **Table 2**, it is evident that vaithyanathan *et al*. [8] transform requires less hardware resource and have highest frequency of operation than remaining approximations. The delay and computational complexity are reduced in this transform.
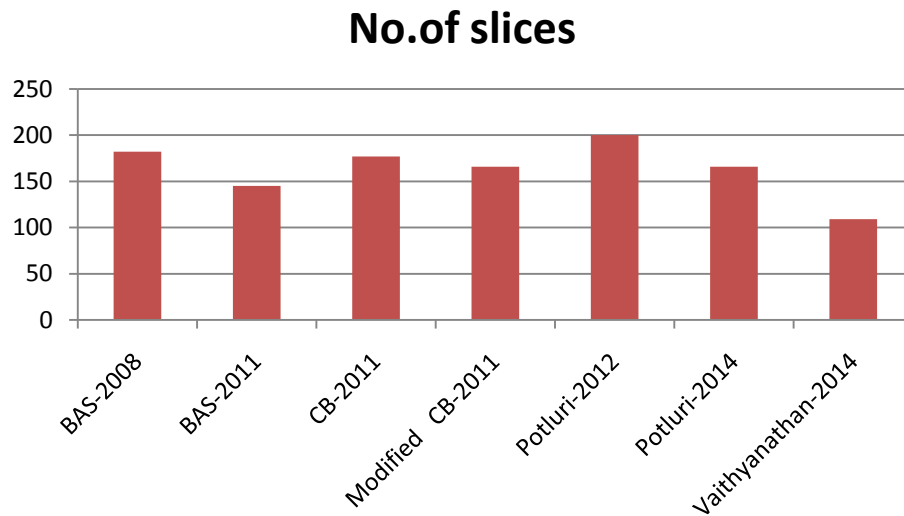

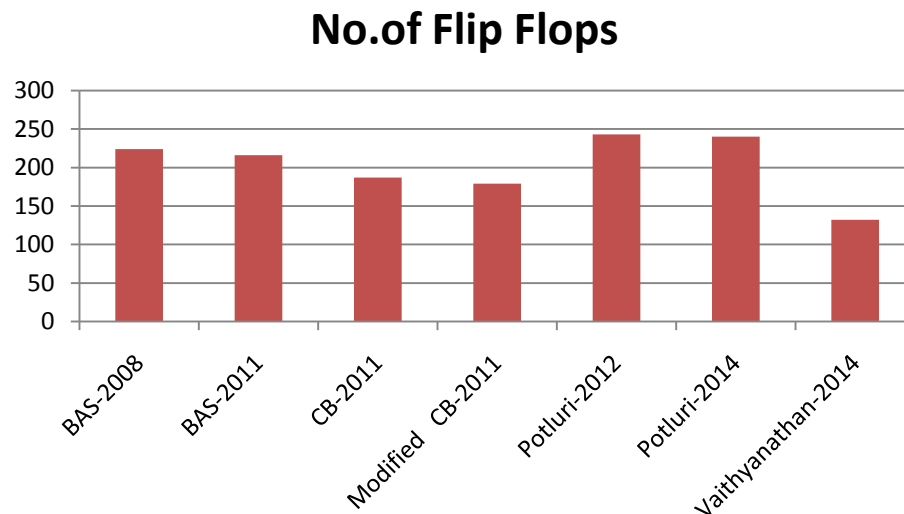**Figure 12.** Comparison of number of slices for different 2-D DCT.

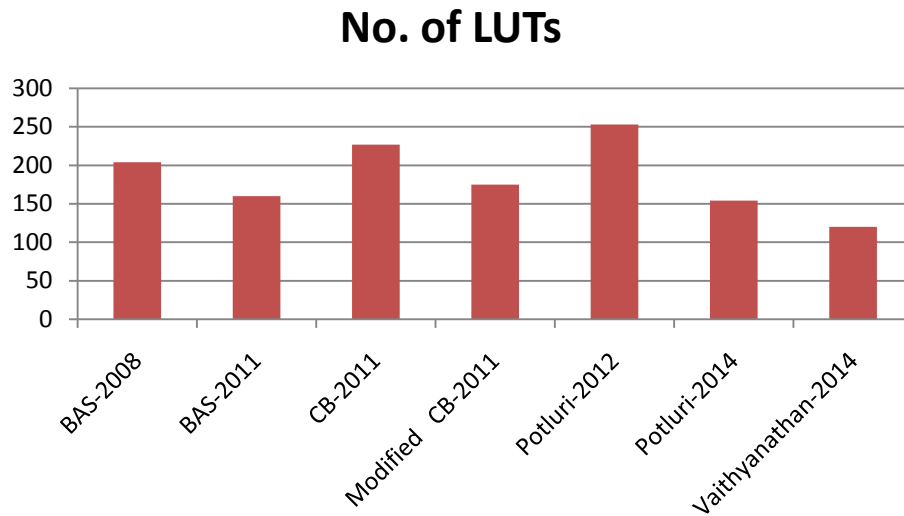**Figure 13.** Comparison of number of flip-flops for different 2-D DCT.

No. of LUTs

Figure 14. Comparison of number of LUTs for different 2-D DCT.
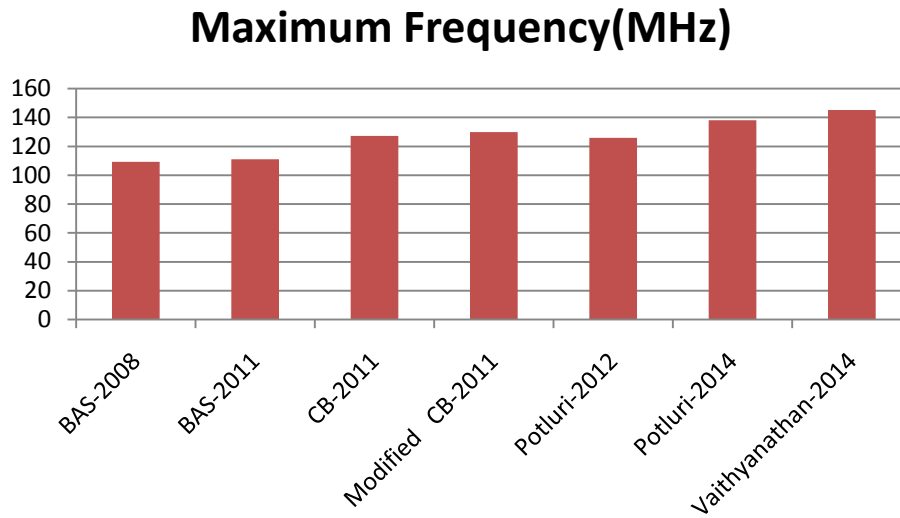
Maximum Frequency(MHz)

Figure 15. Comparison of maximum frequency for different 2-D DCT.

Table 2. Synthesis results of approximate 2D DCTs.

DCT	slices	FF	LUTs	time	Max.Freq (MHz)
BAS-2008	182	224	204	9.15ns	109.289
BAS-2011	145	216	160	9.01ns	110.975
CB-2011	177	187	227	7.86ns	127.194
Modified CB-2011	166	179	175	7.69ns	129.887
Potluri-2012	200	243	253	7.94ns	125.849
Potluri-2014	166	240	154	7.24ns	138.007
Vaithyanathan-2014	109	132	120	6.89ns	145.138

4. Conclusion

In this paper, we proposed 1) VLSI Architectures for Approximate 2D DCTs and 2) hardware implementation of

proposed approximate 2D DCT transforms. All proposed approximate 2D DCT transforms perform well. However, Vaithyanathan *et al*. transform offers lower computational complexity and faster than all other transforms. In terms of image compression, the approximate transforms could outperform the conventional transforms. Hence the proposed transforms are the best approximation for the DCT in terms of computational complexity and speed. Introduced implementations address 2-D approximate DCTs. All the approximations were digitally simulated, prototyped and implemented using modelsim, VirtexE FPGA kit and Xilinx. The proposed architectures are suitable for image and video processing, being candidates for improvements in several standards including the HEVC. In future, the approximate versions for the 16-, 32- and 64-point DCT will be developed.

References

[1] Ahmed, N., Natarajan, T. and Rao, K.R. (1974) Discrete Cosine Transform. *IEEE Transactions on Computers*, **23**, 90-93. http://dx.doi.org/10.1109/T-C.1974.223784

[2] Bouguezel, S., Ahmad, M.O. and Swamy, M.N.S. (2008) Low-Complexity 8×8 Transform for Image Compression. *Electronics Letters*, **44**, 1249-1250. http://dx.doi.org/10.1049/el:20082239

[3] Bouguezel, S., Ahmad, M.O. and Swamy, M.N.S. (2011) A Low-Complexity Parametric Transform for Image Compression. *Proceedings of IEEE International Symposium on Circuits and Systems*, Rio de Janeiro, 15-18 May 2011, 2145-2148.

[4] Cintra, R.J. and Bayer, F.M. (2011) A DCT Approximation for Image Compression. *IEEE Signal Processing Letters*. **18**, 579-582. http://dx.doi.org/10.1109/LSP.2011.2163394

[5] Bayer, F.M. and Cintra, R.J. (2012) DCT-Like Transform for Image Compression Requires 14 Additions Only. *Electronics Letters*, **48**, 919-921. http://dx.doi.org/10.1049/el.2012.1148

[6] Potluri, U.S., Madanayake, A., Cintra, R.J., Bayer, F.M. and Rajapaksha, N. (2012) Multiplier-Free DCT Approximations for RF Multi-Beam Digital Aperture-Array Space Imaging and Directional Sensing. *Measurement Science and Technology*, **23**, 1-15. http://dx.doi.org/10.1088/0957-0233/23/11/114003

[7] Potluri, U.S., Madanayake, A., Cintra, R.J., Bayer, F.M., Kulasekera, S. and Edirisuriya, A. (2014) Improved 8-Point Approximate DCT for Image and Video Compression Requiring Only 14 Additions. *IEEE Transactions on Circuits and Systems—I*: *Regular Papers*, **61**, 1727-1740. http://dx.doi.org/10.1109/TCSI.2013.2295022

[8] Dhandapani, V. and Ramachandran, S. (2014) Area and Power Efficient Architecture for Image Compression. *EURASIP Journal on Advances in Signal Processing*, **180**, 1-9. http://dx.doi.org/10.1186/1687-6180-2014-180

[9] Edirisuriya, A., Madanayake, A., Cintra, R.J. and Bayer, F.M. (2012) A Multiplication-Free Digital Architecture for 16×16 2-D DCT/DST Transform for HEVC. *IEEE* 27*th Convention of Electrical and Electronics Engineers in Israel*, Eilat, 14-17 November 2012, 1-5.

[10] Jridi, M., Alfalou, A. and Meher, P.K. (2015) A Generalized Algorithm and Reconfigurable Architecture for Efficient and Scalable Orthogonal Approximation of DCT. *IEEE Transactions on Circuits and Systems—I*: *Regular Papers*, **62**, 449-457. http://dx.doi.org/10.1109/TCSI.2014.2360763

[11] Lengwehasatit, K. and Ortega, A. (2004) Scalable Variable Complexity Approximate Forward DCT. *IEEE Transactions on Circuits Systems for Video Technology*, **14**, 1236-1248. http://dx.doi.org/10.1109/TCSVT.2004.835151