VOL. 10, NO. 9, MAY 2015 ISSN 1819-6608
ARPN Journal of Engineering and Applied Sciences


www.arpnjournals.com

MODIFIED APPROXIMATE DCT FOR IMAGE COMPRESSION USING EFFICIENT BINARY ADDER

D. Jeyalakshmi and T. Bernatin
Department of Electronics and Telecommunication Engineering, Sathyabama University, Chennai, India
E-Mail: jeyabalaji16@gmail.com

ABSTRACT

Image processing became a big challenge in the multimedia usage for processing the images with low power and high efficiency. For processing images, DCT (Discrete cosine transforms) are utilized for compression because of its advantages over the energy consumption. In existing methods, they used Approximate DCT with fast algorithm and regular pipelining structure for reducing the design complexities. Because of the recursive nature in the existing design, high speed applications are becoming tedious. In our proposed system, we are using Modified Approximate DCT transforms with efficient binary adder circuits. The proposed solution possesses reduced number of gates and logic realization becomes easy for implementation.

Keywords: modified approximate DCT, image compression, efficient binary adder.

1. INTRODUCTION

In Today's world, Digital Multimedia is very popular because of their vital role as well as their fast growth and development. But, contents of Multimedia has large amount of data with it and it needs large storage area. For a better resolution of image, the data size is increasing accordingly.

Like any Fourier related transform, DCT (Discrete Cosine Transform) is a function having data information in terms of a sum of sine wave form signals with various frequencies and amplitudes. In many applications, DCT operate as a function with a finite number of discrete data values similar to that of DFT (Discrete Fourier Transform).

DCT → uses only cosine transform

DFT → uses sine & cosine transform

8-point Approximation based technique was proposed for compressions [1, 3].

Works with 16 point DCT approximations are also performed for some operations [4, 5].

In VLSI, it is a big task for Area reduction and power consumption and DCT plays a main role for these parameters.

The DCT, mainly the DCT-II, is used in signal and image processing for lossless image compression, because it has a good "energy compaction" property: [1, 2] most of the signal data tends to be focused in a low-frequency components of the DCT, considering the KLT transform for signals.KLT (Karhunen-Loeve Transform) has a Linear and not separable property where full matrix multiplication is performed.

DCT utilizes some fixed count of images and by this, fast implementations are possible.DCT also has better correlation characteristics and Energy compaction properties and are also widely used in many applications. DCT can be easily solved by using available DCT formulae and equations, in the form of matrices where the different values of the DCT can be obtained. It is easy to handle these matrices.

The DCT is used for all types of image compression, like MJPEG, MPEG, JPEG formats. The 2D DCT of N× N matrices are calculated and the quantization of values can be done. N takes the value 8 and the DCT formula is applied row-wise and column-wise of the matrices and computations are made.

One of the types of DCT is Approximate DCT which has only Adder circuits. When we use some efficient carry adders instead of normal adder circuits, work efficiency has improved. In the rest of the paper, we are dealing with Approximate DCT and our proposed DCT and the comparative Analysis is found.

2. ANALYSIS OFDCT

In previous systems, the method employed involves multiplication, addition, and shift operation. They have some complex structures and their hardware implementations are not economical. Then we have a approximate DCT which has no multiplications.

A. Approximation DCT

An Approximate DCT is a variety of DCT with the 8 ×8 matrix space and the matrices that possess low computation cost is imposed. Here the cost of a transformation matrix is defined as the number of arithmetic operations like additions or subtractions required for its computation. One of the main advantages is that the matrices do not require any kind of multiplication operations. Approximation DCTs uses additions and does not have multipliers or shifters. Hence, they are very efficiently used in VLSI implementations.

But, Approximate DCT have some of the following disadvantages;

1. Even though they have only added circuits for operations. They involve more number of gates during implementation.
2. The delay is more for their operations.

The table-1 shows the device utilization summary of the existing approximate DCT.

Table-1. Device utilization report.

Device Utilization Summary				
Logic Utilization	Used	Available	Utilization	Note(s)
Number of Slice Flip Flops	107	7,168	1%	
Number of 4 input LUTs	346	7,168	4%	
Logic Distribution				
Number of occupied Slices	188	3,584	5%	
Number of Slices containing only related logic	188	188	100%	
Number of Slices containing unrelated logic	0	188	0%	
Total Number of 4 input LUTs	346	7,168	4%	
Number of bonded IOBs	144	141	102%	OVERMAPPED
IOB Flip Flops	76			
Number of GCLKs	1	8	12%	
Total equivalent gate count for design	3,648			
Additional JTAG gate count for IOBs	6,912			

B. Proposed method

i. Modified approximation DCT

Modified Approximation DCT is a Approximation DCT which uses Efficient binary adders instead of normal adders.

By using efficient binary adders, we have some advantages like reduction in logic implementation and the number of gate counts are reduced .This is proportional to the clock cycles and power consumption.

This DCT is also 8-point DCT and the DCT is done by means of a butter fly diagram. This is also a multiplier free DCT which has D-flip-flop and efficient binary adder for its operation.Figure-1 shows the block diagram of our proposed transform. In this diagram, 1-D DCT is performed by means of giving a row wise input of a matrix transform in the blocks.DCT operations are performed and the output values show the result of 1-D DCTS.

Also, a 2-D DCT is performed by means of using a transposition buffer. (i.e.) the output of this 1-D DCT is given along with the column values in the Transposition Buffer and finally we get a 2-D DCT output.

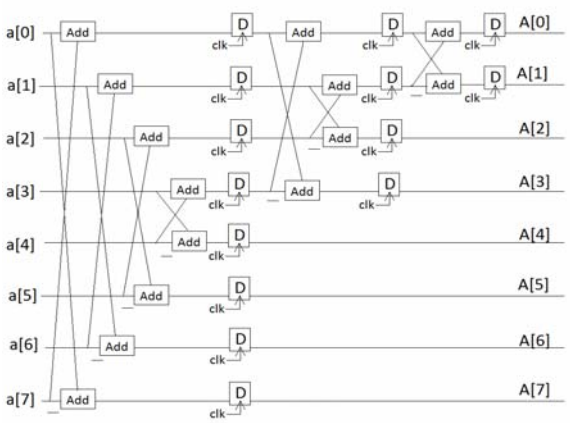

Figure-1. Modified approximate DCT.

ii. Modified approximation IDCT

The modified approximate IDCT is the reverse structure of our proposed DCT what we used above. Figure-2 shows the modified Approximation IDCT.The structure used involves same efficient binary adder.

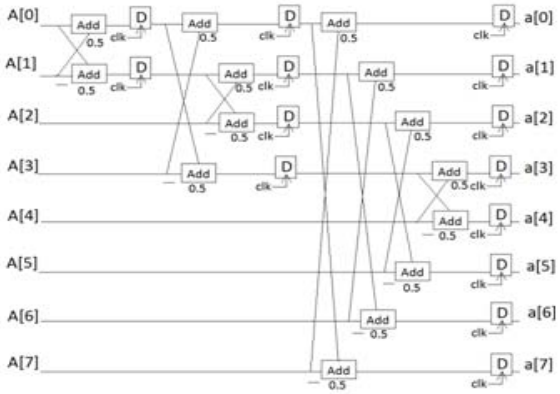

Figure-2. Modified approximate IDCT.

The above diagram shows the IDCT for 1D values to get the 2D, the output of the above is multiplied given to the transpose matrix and the output of transpose is given to the column of the matrix values and 2D DCT is then achieved. The images are converted into the frames using MatLab Simulink software and pixel values are obtained. This pixel values represents the array of the matrix proposed. We can take a 8 point values of the matrix and the row values are given as the input to the proposed DCT structure and a 1D-DCT is performed. Also, a 2-D DCT is performed using a transposition buffer.

Hence, by applying our modified DCT approximation, an image compression takes place. Also, we can retrieve the image by doing image decompression using modified IDCT approximation.

We can synthesize them using Xilinx software and can calculate the gate counts and reduction of gate counts is found using our proposed method.

iii. Efficient binary adder

In our proposed method, we are using the efficient binary adder instead of normal adders used in the existing systems. By using this adder, logic counts & delays are reduced. Figure-3 shows the 3 bit efficient binary adder used in our proposed.

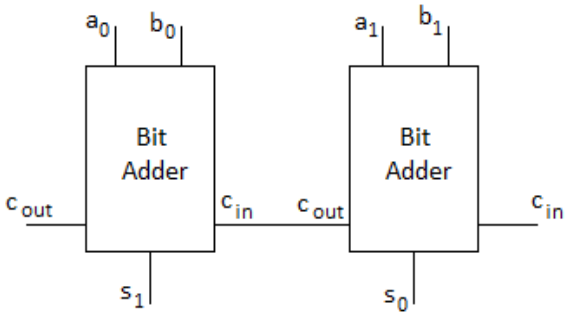

Figure-3. 3 bit efficient binary add.

3. RESULTS AND DISCUSSIONS

Table-2 shows the device utilization summary of the proposed approximate DCT using efficient binary adder.

Table-2. Device utilization report.

Device Utilization Summary				
Logic Utilization	**Used**	**Available**	**Utilization**	**Note(s)**
Number of Slice Flip Flops	107	7,168	1%	
Number of 4 input LUTs	340	7,168	4%	
Logic Distribution				
Number of occupied Slices	179	3,584	4%	
Number of Slices containing only related logic	179	179	100%	
Number of Slices containing unrelated logic	0	179	0%	
Total Number of 4 input LUTs	340	7,168	4%	
Number of bonded IOBs	144	141	102%	OVERMAPPED
IOB Flip Flops	76			
Number of GCLKs	1	8	12%	
Total equivalent gate count for design	3,588			
Additional JTAG gate count for IOBs	6,912			

Table-3 gives a comparative report on the device utilisation for Approximate DCT and our Proposed Method. From the report the total equivalent gate counts are reduced using efficient binary adder.

Table-3. Comparative report.

Types of DCT	No. of slice flip-flops	No. of 4 input LUT's	Total equivalent Gate count
Approximate DCT	107	346	3648
Proposed method	107	340	3588

The below bar graph shows the difference in delay values between the existing and proposed DCTs.

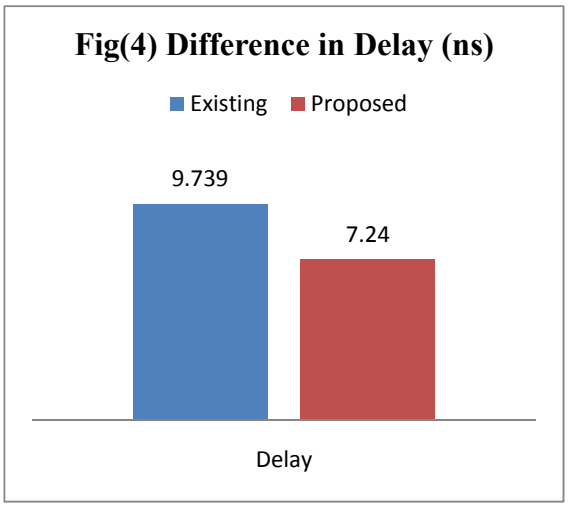

Figure-4. Difference in delay (ns).

4. COMPARITIVE ANALYSIS

Table-4 shows a comparative report for the different types of DCT showing Mean Square Error (MSE)values, Peak Signal to Noise Ratio(PSNR)values and Efficiency values. The types of DCT present in the table shows the all existing methods in DCT. A survey is made with the existing DCTs to understand the comparative performance. From the Table-4, we infer on the performance of our modified approximate DCT with all the types of DCT.

Table-4. Comparative analysis.

Types of DCT	MSE($X10^{-4}$)	Efficiency η	PSNR
Exact DCT [1]	0.000	93.991	28.336
Fast multiplier less DCT [3]	7.102	85.380	26.902
BAS 2008 [6]	2.378	86.863	27.245
CB-2011 [9]	0.980	87.432	27.369
Approximate DCT [8]	7.899	80.597	25.726
Proposed method	7.778	84.766	25.706

5. CONCLUSIONS

By using our proposed method (modified approximate DCT), we aim at a better outfit of reducing the number of gate counts and hence logic implementation realized reduces and correspondingly implementing using VLSI can be made possible with reduced complexities. This propose methods is used for further future enhancement techniques like CORDIC based algorithm, which normally uses normal DCT for its operations. It gives an efficient throughput when using our proposed DCT for implementing CORDIC.

REFERENCES

[1] T. Bernatin, G. Sundari. 2014. "Video compression based on Hybrid transform and quantization with Huffman coding for video codec" International conference on control, instrumentation, communication and computational technologies (ICCICCT), pp 476-480, July, IEEE.

[2] T. Bernatin, G. Sundari. 2014, "A Survey of Video Compression Using Video Codec" IETE 45th Mid Term Symposium on Broadband-Technologies and Services for Rural India MTS 14, 4th and 5th April.

[3] U. S. Potluri, A. Madanayake, F. M. Bayer, R. J. Cintra, A. Edirisuriya and S. Kulasekera. 2013. "Improved 8-point approximate DCT for image and video compression requiring only 14 additions", IEEE Transactions on Circuits and Systems I.

[4] F. M. Bayerand R. J. Cintra. 2012. "DCT-like transform for image compression requires 14 additions only", Electronics Letters, Vol. 48, pp. 919-921.

[5] F. M. Bayer, R. J. Cintra, A. Madanayake and U. S. Potulri. 2013. "Multiplierless approximate 4-pointDCT VLSI architectures for transform block coding", Electronics Letters, Vol. 49, pp. 1532-1534.

[6] R. J. Cintra. 2011. "An integer approximation method for discrete sinusoidal transforms", Journal of Circuits, Systems, and Signal Processing, Vol. 30, pp. 1481-1501.

[7] R. J. Cintraand F. M. Bayer. 2011. "A DCT approximation for image compression", IEEE Signal Processing Letters, pp. 579-582.

[8] R. J. Cintra, F. M. Bayer, and C. J. Tablada. 2014. "Low-complexity 8-point DCT approximations based on integer functions", Signal Processing, Vol. 99, pp. 201-214.

[9] P. Meher, S. Y. Park, B. Mohanty, K. S. Lim, and C. Yeo. 2014. "Efficient integer DCT architectures for HEVC'', IEEE Transactions on, Vol. 24, pp. 168-178.

[10] H. Schwarz, T. Wiegand, T. K. Tan, J.R. Ohm&G. J. Sullivan. 2012. "Comparison of the coding efficiency of video coding standards including HEVC", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 22, pp. 1669-1684.

[11] W.J. Han,J.R. Ohm, T. Wiegand & G. J. Sullivan. 2012."Overview of the HEVC (high efficiency video coding) standard", IEEE Transactions on Circuits and Systems for Video Technology, Vol. 22, pp. 1649-1668.