

VLSI ARCHITECTURE FOR IMAGE COMPRESSION THROUGH ADDER MINIMIZATION TECHNIQUE AT DCT STRUCTURE

N.R. Divya¹ and K. Kannadasan²

Department of Electronics and Communication Engineering, Adhiparasakthi Engineering College, India

E-mail: ¹divyanr32@gmail.com, ²kannadasan.ec@gmail.com

Abstract

Data compression plays a vital role in multimedia devices to present the information in a succinct frame. Initially, the DCT structure is used for Image compression, which has lesser complexity and area efficient. Similarly, 2D DCT also has provided reasonable data compression, but implementation concern, it calls more multipliers and adders thus its lead to acquire more area and high power consumption. To contain an account of all, this paper has been dealt with VLSI architecture for image compression using Rom free DA based DCT (Discrete Cosine Transform) structure. This technique provides high-throughput and most suitable for real-time implementation. In order to achieve this image matrix is subdivided into odd and even terms then the multiplication functions are removed by shift and add approach. Kogge_Stone_Adder techniques are proposed for obtaining a bit-wise image quality which determines the new trade-off levels as compared to the previous techniques. Overall the proposed architecture produces reduced memory, low power consumption and high throughput. MATLAB is used as a funding tool for receiving an input pixel and obtaining output image. Verilog HDL is used for implementing the design, Model Sim for simulation, Quartes II is used to synthesize and obtain details about power and area.

Keywords:

Distributed Arithmetic-Discrete Cosine Transform (DA-DCT), Kogge_Stone_Adder (KSA), Inverse Discrete Cosine Transform (IDCT), Very Large Scale Integrated Circuit (VLSI)

1. INTRODUCTION

In multimedia system the problem of low cost and efficient compression is still almost unsolved. The most significant part of multimedia systems is application involving image or video, which require computationally intensive data processing. Moreover, as the purpose of mobile device increases exponentially, there is a rising need for multimedia application to operate on these portable devices. Multimedia files are big and consume tons of hard disk space. Imaging and video application are one of the fastest developing sectors of the marketplace today, where large quantities of data are required for image transformation due to this memory space usage increased. To overcome this compression state are introduced in image and video application. Compression shrinks files, making them smaller and more practical to store and share. Compression works by removing repetitious or redundant data, effectively summarizing the contents of a file in a path that preserves as much of the original as possible. In order to trim down the multimedia data, data compression technique is widely practiced. Transformation of input images is fed into coefficients, then they are leveled, after this stage reconstructed images are obtained as output.

DCT [1], [2] are most commonly used for compression. Implementation of DA-DCT (Distributed Arithmetic based

Discrete Cosine Transform) multipliers using ROM produce partial product together with adders and that accumulate the partial product, by this way area reduced. By using ROM based DCT, redundancy occurs and so in proposed method ROM-free DA-based DCT with Parallel Prefix Adder (PPA) method is used, where DA-based DCT uses reconfigurable odd DCT and even DCT architectures. Using level parameters, the proposed DCT architecture can dynamically switch from one trade-off level to another with little overhead. When the level control signals are ONE's, the circuit works like a normal DCT processor without any modification of DCT bases. When the level control signals are zeroes, make more or less of the adders zeroes and turn off the adders by this memory reduction occurred. By using 2D-DCT for image compression, adders and multiplier rate are larger and so many digital errors are happening [3]. To overcome this digital complexity Parallel Prefix Adders (Kogge-Stone-Adders) are applied [4]. As compared to other adders PPA are used because of its block compression and fastness.

Execution of the 2D-DCT and Inverse of 2D-DCT in VLSI design so the outcome should be obtained accurately. Here the data's are processed in digital state. Compressed image uses a lossy compression state where the restored image will not be original image. The quality of the reconstructed image can be obtained through the Peak Signal to Noise Ratio for different images. In this paper, we have presented efficient DA based VLSI architecture for DCT by exploiting redundancy [5]. Section 2 explains about design of DA-based DCT architecture. Proposed Parallel Prefix Adder is presented in section 3. Implementation and comparison result for adders, area and power are discussed in section 4. Conclusions are made in section 5.

2. DESIGN OF DA-BASED DCT STRUCTURE

Discrete cosine transform (DCT) is unitary of the major compression schemes owing to its near optimal performance and delivers energy compaction efficiency greater than any other transform. The transformation algorithm is presented in [6].

2.1 DCT ARCHITECTURE

By using DCT architecture, then DWT is that there is higher throughput, lesser complexity and also no need to manipulate complex number. When computing 2D DCT, a greater number of multipliers and adders are required for enforcing the compression organization in harder, which shows the most time-consuming process, it can be completely avoided in the proposed DA-based DCT architecture with Kogge_Stone_Adder. A minimum number of additions are used to the DCT based on the Distributed Arithmetic.

2.2 DA-BASED DCT

Distributed Arithmetic (DA) is an efficient method for computing inner products. It uses look up tables and replaced the accumulators instead of multipliers for computing inner products in DCT. DA-based DCT architecture is considerably known for VLSI implementation due to its reducing ROM size, by this area reduced [8]. DA-based DCT uses even-odd frequency decomposition of the DCT along with memory reduction. The 1D 8-point DCT are constructed using a DA-Butterfly-Matrix that has even and odd processing elements and Parallel Prefix Adder (Kogge_Stone_Adder) are show in Fig.1.

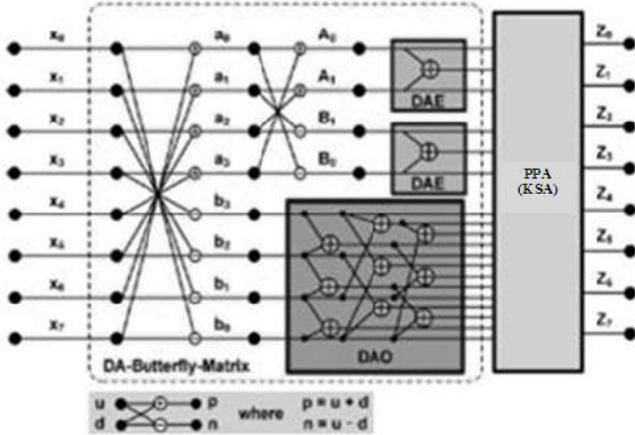


Fig.1. Architecture of 1D 8-point DA-DCT

The 1D-DCT employs the DA-based architecture [6] and proposed Kogge_Stone_Adder to achieve a high-speed, small area and low power design. The 1D 8-point DCT can be expressed as follows in Eq.(1).

$$z_n = \frac{1}{2} k_n \sum_{m=0}^7 x_m \times \cos\left(\frac{(2m+1)np}{16}\right) \quad (1)$$

where, x_m denotes the input data;
 z_n denotes the transform output;

$$k_n = \frac{1}{\sqrt{2}} \text{ for } n = 0;$$

$$0 \leq n \leq 7; k_n = 1 \text{ for other } n \text{ values.}$$

By neglecting the scaling factor $\frac{1}{2}$, the 1D 8-point DCT in Eq.(1) can be divided into odd and even parts as presented in [7].

The DA-based DCT operation performs even, odd decomposition of input pixels and the representation of cosine basis in Canonical Sign Digit (CSD) [7]. Image compression operations are taking place as, the input image is broken into 8×8 block and they are multiplied by DCT matrix. After multiplication, addition process has taken place.

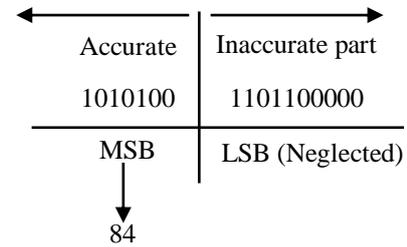
For instance, take the input pixel value 120 is multiplied with the DCT fraction value 0.707 answers as 84.84, where the complexity is more and also delay increases due to the carry part. To overcome this complexity and delay state in this paper at DA-based DCT structure Canonical Sign Digit are used [6] by neglecting the unwanted LSB's this is accomplished by reproducing the input pixels with larger number as 210 are shown below.

$$2^{10} \times 0.707 = 723.968$$

By multiplying pixel value with larger number we are getting,

$$724 \times 120 = 86880$$

For 86880 the binary values are 10101001101100000, and by neglecting the 10 LSB original values are held as indicated below.



By this proposed method the decimal values are completely carried away and complexity reduces largely by left shifting as described the cosine basis in [6].

3. PARALLEL PREFIX ADDER

The comparative study of the different kind of adders is shown in [7]. The complexity problem can be completely swept over by using Parallel Prefix Adder (PPA). It is one of the fastest adders which compute the carry_i for each bit in a tree structure [9]. The different types of PPA adders are available where Brent-Kung and Kogge-Stone are very popular. In this paper Kogge-Stone Adder (KSA) is used in DA-based DCT structure due to its fastness.

3.1 KOGEE_STONE_ADDER

Adder does the work by adding 2 additional signals often part of other arithmetic components, like sum-of-products, multiplier etc. Here the KSA is the component of Parallel Prefix form carry look-ahead adder. It generates carry signals in $\log_2 n$ levels. In KSA, carriers are computed fast by computing them in parallel at the monetary value of increased area as $(n \cdot \log_2 n - n + 1)$. In this adder, generate and propagate takes main part. These signals are given by the logic equations as for each bit i of adder generate (G_i) as shown in Eq.(2).

$$G_i = a_i \cap b_i \quad (2)$$

where, G_i indicates whether a carry is generated from that bit or not, and also for each bit i of adder propagate P_i as presented in Eq.(3).

$$P_i = a_i \oplus b_i \quad (3)$$

where, P_i indicates whether a carry is Propagated from that bit or not. G and P blocks comprising state are equally expressed in Fig.2, then the operation of 4bit Kogge Stone Adder are as indicated in Fig.3 where the carry look ahead state are as input as presented in [4].

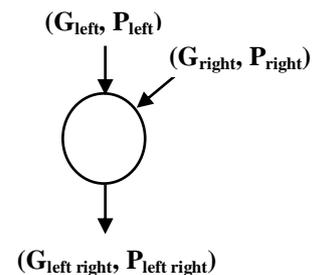


Fig.2. KSA carry operator

Carry look ahead network differentiates KSA from other adders and is the main force behind its high performance. This step involves computation of carries corresponding to each bit. It uses group propagate and generate as intermediate signals which are given by the logic Eqs.(4) and (5) as mentioned below:

$$P_i : j = P_i : k+1 \text{ and } P_k : j \tag{4}$$

$$G_i : j = G_i : k+1 \text{ or } (P_i : k+1 \text{ and } G_k : j) \tag{5}$$

Post processing is the final step it involves computation of sum bits. Sum bits are computed by the logic given below in Eq.(6).

$$S_i = P_i \oplus C_{i-1} \tag{6}$$

For establishing an efficient trade-off between computational complexity and image quality, this approach could achieve minimum degradation in the image quality and also reduced the total number adders required for addition function. By this way the adders are computed at DCT for different levels. In case of image compression after performing 2D DCT quantization table is applied to remove AC coefficients by this complexity will remain same for whole ranges of compression rate. So, the parameterizable level is used to achieve quantization table less adjustable image compression. By evaluating this adjustable hardware as per our requirement is obtained.

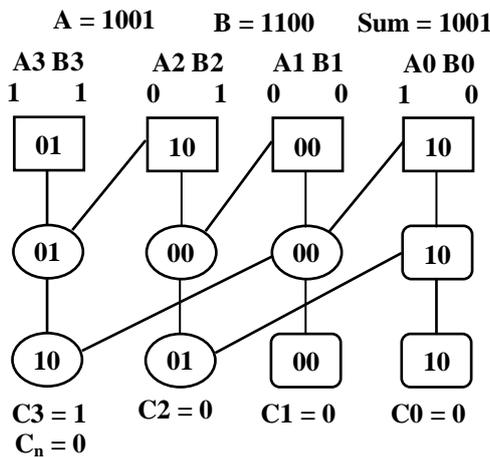


Fig.3. 4 Bit Kogge_Stone_Adder

Table.1. Comparison outputs of adders

ADDER TYPE	HARDWARE COMPLEXITY	SPEED (MHz)
Ripple Carry Adder	98	164.58 MHz
Carry Save Adder	97	205.21 MHz
Carry Look Ahead Adder	99	256.61 MHz
Kogge_Stone_Adder	83	317.97 MHz

4. EXPERIMENTAL RESULTS

The image is converted into pixels using MATLAB and the values are stored as a text file. The text file is accessed by the MODELSIM ALTERA and the corresponding 2D DCT coefficients are calculated. These values are then fed to the IDCT module which returns the spatial data sequence.

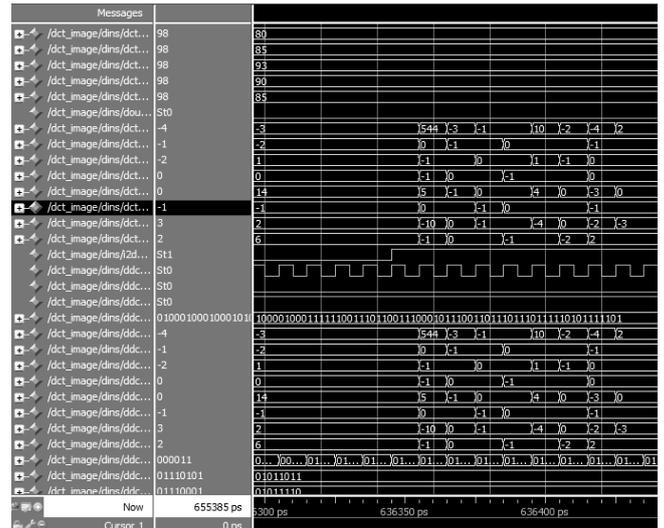


Fig.4. 2D DCT coefficient

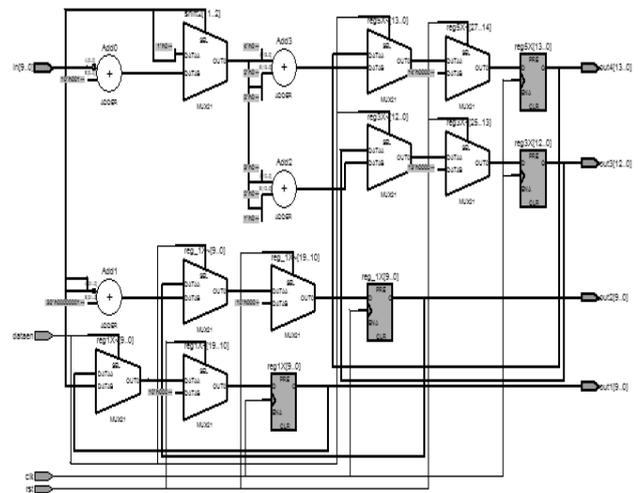


Fig.5. RTL for 2D DCT

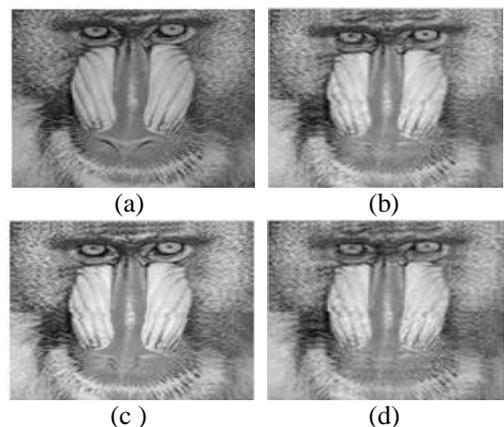


Fig.6(a). Original Image, (b). Image reconstruction by level 0, (c). Image reconstruction by level 1, (d). Image reconstruction by level 2

These data are written to a text file, then image reconstructed from the text file using MATLAB coding. Finally hardware and

speed optimizations are measure by using QUARTUS II EDA tool. The simulated results are shown above in Fig.4 and Fig.5.

The performance of the proposed method with DA-based DCT in terms of various levels is shown in Fig.6. Proposed levels are user defined that’s as per the user requirement they are consumed. If levels are increased complexity is reduced but the image quality goes down at level 0 its opposite complexity increased but the image quality gained high. This is clearly calculated by using Peak Signal to Noise Ratio (PSNR) as shown in Table.2.

Table.2. PSNR and F_{max} values for the reconstructed image

LEVELS	PSNR VALUES	FMAX
Level 0	5.3250	322.68 MHz
Level 1	5.3246	314.76 MHz
Level 2	5.3232	429.55 MHz

Then the comparison of area and power consumption is shown below in Table.3 and Table.4. Various frequency levels are taken place at Table.2.

Table.3. Area analysis for the reconstructed images

Parameter	Total	Level 0	Level 1	Level 2
Total logic elements	15408	543	742	175
Logic register	15408	226	243	52
Registers	15408	243	226	52
Total pins	347	171	171	171
Total memory bits	516096	0	0	0
PLL	4	0	0	0

Table.4. Power analysis for the reconstructed images

Powers	Level 0	Level 1	Level 2
Thermal Power Dissipation	329.79mW	329.79mW	80.06mW
Dynamic Power Dissipation	0.00mW	0.00mw	0.00mW
Static Power Dissipation	303.03mW	303.03mW	51.80mW
I/O Thermal Power	26.76mW	26.76mW	28.26mW

5. CONCLUSION

The concept of DA-based DCT and IDCT architectures which take on the algorithmic strength reduction technique to cut the device utilization pulling the power consumption, low have thus also been planned and introduced in VLSI design. The DCT computation is as well done by DA based with sufficiently high precision, yielding an acceptable quality by way of using the Kogge_Stone_adder by eliminating the carry propagation. The proposed DA-based DCT architecture achieves a maximum efficiency over multiplier based approach. Therefore the proposed architecture is suitable for a high compression rate at area and power.

REFERENCES

- [1] Yao Wang, J. Ostermann and Ya-Qin Zhang, “Video Processing and Communications”, First Edition, Prentice-Hall, 2002.
- [2] Gilbert Strang, “The Discrete Cosine Transform”, *Society for Industrial and Applied Mathematics Review*, Vol. 41, No. 1, pp. 135-147, 1999.
- [3] G. K. Wallace, “The JPEG still picture compression standard”, *IEEE Transactions on Consumer Electronics*, Vol. 38, No. 1, pp. xviii–xxxiv, 1992.
- [4] Dhanya Geethanjali Sasidharan and Aarathy Iyer, “Comparison of Multipliers Based on Modified Booth Algorithm”, *International Journal of Engineering Research and Applications*, Vol. 3, No. 1, pp. 1513-1516, 2013.
- [5] V. Muralitharan and M. Jagadeeswari, “An Enhanced Carry Elimination Adder for Low Power VLSI Implementation”, *International Journal of Engineering Research and Applications*, Vol. 2, No. 2, pp. 1477-1482, 2012.
- [6] N. R. Divya and K. Kannadasan, “Logic Complexity Reduction and VLSI Architecture for Image Compression Using Conventional Adders”, *International Journal of Electrical and Communication Engineering for Applied Research*, Vol. 2, pp. 31-35, 2014.
- [7] N. R. Divya and K. Kannadasan, “Image compression using DA-DCT Logic Through VLSI structure with Power Enhancement Scheme”, *International Journal of Graphics and Image Processing*, Vol. 4, No. 1, pp. 32-37, 2014.
- [8] A. M. Shams, A. Chidanandan, W. Pan and M. A. Bayoumi, “NEDA: A low-power high-performance DCT architecture”, *IEEE Transactions on Signal Processing*, Vol. 54, No. 3, pp. 955-964, 2006.
- [9] Young-Ho Seo and Dong-Wook Kim, “A New VLSI Architecture of Parallel Multiplier Accumulator Based on Radix-2 Modified Booth Algorithm”, *IEEE Transactions on Very Large Scale Integration Systems*, Vol. 18, No. 2, pp. 201-208, 2010.