PalArch's Journal of Archaeology of Egypt / Egyptology

An Efficient Architecture for 8-point Discrete Cosine Transform using Less Number of Multipliers

Reeta Choudhury[1] *Sudhansu Sekhar Nayak*[2]

[1] Research Scholar, [2]Professor in Physics, Centurion University of Technology and Management, Odisha, India

Reeta Choudhury Sudhansu Sekhar Nayak: An Efficient Architecture for 8-point Discrete Cosine Transform using Less Number of Multipliers -- Palarch's Journal Of Archaeology Of Egypt/Egyptology 17(9). ISSN 1567-214x

Keywords: Discrete cosine transform, Discrete Fourier transform, VLSI, Fast Fourier transform. Fast Hartley transform.

ABSTRACT

In this paper, a simple architecture has been presented for direct implementation of 8-point discrete cosine transform (DCT). This architecture is suitable for VLSI implementation and it provides high throughput of computation. Since the architecture uses minimum number of multipliers, its area-and hardware-complexities are less. Its VLSI performance is good. Sub expression sharing technique and sharing the multipliers with same constants have been used in this architecture.

1. Introduction

In the year 1974, Ahmed et al. [1] proposed a real-valued discrete transform called the discrete cosine transform (DCT). It is most popular amongst the discrete orthogonal transforms. It has wide applications in the field of digital signal processing. Many algorithms and architectures have been proposed for efficient implementation of the DCT. It is indicated in [1] that N - point DCT can be computed using $2N$ - point fast Fourier transform (FFT). Malvar [2] showed that DCT could be computed using fast Hartley transform (FHT) of same length. Several algorithms and architectures have been developed for implementation of DCT in VLSI chips [3,4,5]. Aggoun [5] proposed an architecture in which the number of multipliers is reduced from N^2 to $N/2$ for N= 8 by exploiting the symmetries of the cosine functions.

In this paper, we have proposed a simple architecture for direct implementation of 8-point DCT. Multipliers used in an architecture increase area-, hardware–and time– complexities. So, we have proposed an architecture using twentyone multipliers, which is suitable for VLSI implementation. These multipliers share subexpressions.

This paper is organized as follows. In Section-II, the algorithm for implementation of 8-point DCT is given. Section-III deals with the architecture for implementation of DCT. Concluding remarks are provided in Section-IV.

2. ALGORITHM

According to the definition of DCT [1], for a given data sequence $\{x_n ; n = 0, 1, 2,..., N\text{-}1\}$, the DCT data sequence $\{X_k ; k = 0, 1, 2,..., N\text{-}1\}$ is given by the following relation.

$$X_k = \sum_{n=0}^{N=1} x_n \cos\left[\frac{\pi(2n+1)\,k}{2N}\right] \quad (1)$$

For $k = 0,1,2,...., N\text{-}1$

For 8- point DCT

$$X_k = \sum_{n=0}^{7} x_n \cos\left[\frac{\pi\,(2n+1)k}{2N}\right] \quad (2)$$

X_0

$$= [x_0 + x_7] + [x_1 + x_6] + [x_2 + x_5] + [x_3 + x_4] \quad (3)$$

$$X_1 = [x_0 - x_7]\cos\frac{\pi}{16} + [x_1 - x_6]\cos\frac{3\pi}{16} + [x_2 - x_5]\cos\frac{5\pi}{16} + [x_3 - x_4]\cos\frac{4\pi}{16} \quad (4)$$

$$X_2 = [(x_0 + x_7) - (x_3 + x_4)]\cos\frac{2\pi}{16} + [(x_1 + x_6) - (x_2 + x_5)]\cos\frac{6\pi}{16} \quad (5)$$

$$X_3 = [x_0 - x_7]\cos\frac{3\pi}{16} - [x_1 - x_6]\cos\frac{7\pi}{16} - [x_2 - x_5]\cos\frac{\pi}{16} - [x_3 - x_4]\cos\frac{5\pi}{16} \quad (6)$$

$$X_4 = [(x_0 + x_7) - (x_1 + x_6) - (x_2 + x_5) + (x_3 + x_4)]\cos\frac{4\pi}{16} \quad (7)$$

$$X_5 = [x_0 - x_7]\cos\frac{5\pi}{16} - [x_1 - x_6]\cos\frac{\pi}{16} + [x_2 - x_5]\cos\frac{7\pi}{16} + [x_3 - x_4]\cos\frac{3\pi}{16} \quad (8)$$

$$X_6 = [x_0 + x_7]\cos\frac{6\pi}{16} - [x_1 - x_6]\cos\frac{2\pi}{16} + [x_2 + x_5]\cos\frac{2\pi}{16} - [x_3 - x_4]\cos\frac{6\pi}{16} \quad (9)$$

and

$$X_7 = [x_0 - x_7]\cos\frac{7\pi}{16} - [x_1 - x_6]\cos\frac{5\pi}{16} + [x_2 - x_5]\cos\frac{3\pi}{16} - [x_3 - x_4]\cos\frac{\pi}{16} \quad (10)$$

Using

A = x_0+x_7, B = x_1+x_6, C = x_2+x_5, D = x_3+x_4,

$A' = x_0 - x_7$, $B' = x_1 - x_6$, $C' = x_2 - x_5$, $D' = x_3 - x_4$, equations (3) to (10) can be written as

$$X_0 = A + B + C + D \quad (11)$$

$$X_1 = A'\cos\frac{\pi}{16} + B'\cos\frac{3\pi}{16} + C'\cos\frac{5\pi}{16} + D'\cos\frac{7\pi}{16} \quad (12)$$

$$X_2 = (A - D)\cos\frac{2\pi}{16} + (B - C)\cos\frac{6\pi}{16} \quad (13)$$

$$X_3 = -C'\cos\frac{\pi}{16} + A'\cos\frac{3\pi}{16} - D'\cos\frac{5\pi}{16} - B'\cos\frac{7\pi}{16} \quad (14)$$

$$X_4 = (A - B - C + D)\cos\frac{4\pi}{16} \quad (15)$$

$$X_5 = -B'\cos\frac{\pi}{16} + D'\cos\frac{3\pi}{16} + A'\cos\frac{5\pi}{16} + C'\cos\frac{7\pi}{16} \quad (16)$$

$$X_6 = (-B' + C)\cos\frac{2\pi}{16} + (A - D')\cos\frac{6\pi}{16} \quad (17)$$

And

$$X_7 = -D'\cos\frac{\pi}{16} + C'\cos\frac{3\pi}{16} - B'\cos\frac{5\pi}{16} + A'\cos\frac{7\pi}{16} \quad (18)$$

8 –point DCT can be implemented straight forward using equations (11) to (18).

3. PROPOSED ARCHITECTURE

The primary aim of this paper is to propose a simple architecture for direct implementation of DCT. The proposed architecture for implementation of 8- point DCT is shown in Fig. I. It consists of 14 adders and 16 subtractors. In a VLSI design, multipliers introduce large area–and hardware–complexities and significant delays. The proposed architecture uses only 21 multipliers, contained in M_1 to M_7 to implement 21 multiplications. M_1, M_3, M_5 and M_7 contain four multipliers each. M_2 and M_6 contain two multipliers each and M_4 contains only one multiplier. Common multiplicands of multipliers of M_1, M_3, M_5 and M_7 are $\cos\frac{\pi}{16}$, $\cos\frac{3\pi}{16}$, $\cos\frac{5\pi}{16}$ and $\cos\frac{7\pi}{16}$ respectively. The multiplicands of multipliers M_2 and M_6 are $\cos\frac{2\pi}{16}$ and $\cos\frac{6\pi}{16}$, respectively. The multiplicand of M_4 is $\cos\frac{4\pi}{16}$. Hardware–complexity is significantly reduced by using sub expression sharing technique and sharing of multipliers with a constant. The hardware complexity of the proposed architecture is much less compared to those of existing architectures. Eight inputs x_0 ,..... and x_7 are available simultaneously. In order to obtain the transformed output components in an order, a number of delays should be used in the architecture.

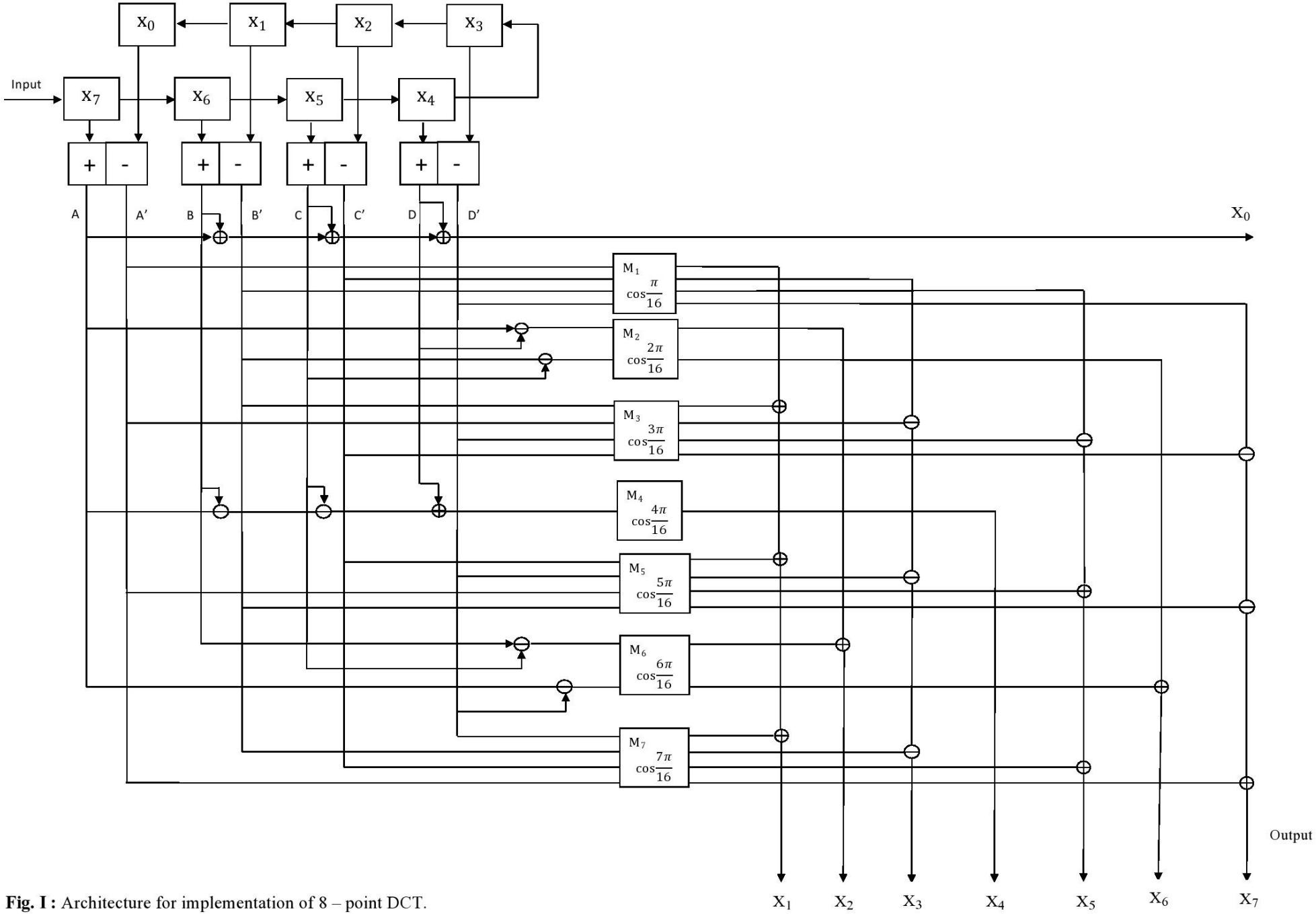

Fig. I : Architecture for implementation of 8 – point DCT.

Table : *Comparison of number of multipliers, number of adders and number of substractors used in proposed achitecture with those of [5] for 8 – point DCT.*

No. of Multipliers		**No. of adders**		**No. of subtractors**	
Architecture [5]	Proposed Architecture	Architecture [5]	Proposed Architecture	Architecture [5]	Proposed Architecture
32	21	36	14	4	16

4. CONCLUSION

In this paper, a simple architecture has been presented for implementation of 8-point DCT. It uses 14 adders, 16 subtractors and 32 multipliers which are significantly less compared to existing architectures. The architecture proposed in [5] requires 32 multipliers and 40 adders and substractors for 8–point DCT. Area - and hardware - complexitive of proposed architecture are small compared to those of [5].

References

N. Ahmed, T. Natarajan, and K.R. Rao, "Discrete cosine transforms", IEEE Trans. on Computers, vol. C-23, pp. 90-93, Jan. 1974.

H.S. Malvar, "Fast computation of discrete cosine transform through fast Hartley transform", Electron Lett., vol. 22, pp. 353-354, March 1986.

S. Boussakta and O. Alshibami, "Fast algorithm for the 3–D DCT–II", IEEE Trans. Signal Processing, vol. 52. pp. 992-1001, April 2004.

S. Saponara, "Real– time and low– power processing of 3D direct/ inverse discrete cosine transform for low– complexity video codec", J Real – Time Image Proc., vol. 7, pp. 43 – 53, 2012.

A. Aggoun, "Three dimensional DCT / IDCT architecture", IJAET, pp. 648 - 658, May 2013.