Realization of Prime-Length Discrete Sine Transform Using Cyclic Convolution

M.N.MURTY
Department of Physics, National Institute of Science and Technology,
Palur Hills,Berhampur-761008, Odisha (INDIA).
mnarayanamurty@rediffmail.com

Abstract:

This paper presents a new algorithm for the implementation of an N-point prime-length discrete sine transform (DST) through cyclic convolution. The proposed algorithm is based on the idea of reformulating prime N-length DST into two $[(N-1)/2]$- point cyclic convolutions. Thus, the hardware complexity can be reduced. This cyclic convolution –based algorithm is used to obtain a simple systolic array for pipelined implementation of the DST. This algorithm preserves all the benefits of very large-scale integration algorithms based on cyclic convolution or circular convolution, such as regular and simple structure. The convolutions play a significant role in digital signal processing due to their nature of easy implementation.

Keywords: Discrete sine transform, Discrete cosine transform, Cyclic convolution, Systolic array, Very large scale integration.

1. Introduction

The Discrete sine transform (DST) was first introduced to the signal processing by Jain[1], and several versions of this original DST were later developed by Kekre *et al.*[2], Jain[3] and Wang *et al.*[4]. There exist four even DST's and four odd DST's, indicating whether they are an even or an odd transform[5]. Ever since the introduction of the first version of the DST, the different DST's have found wide applications in several areas in Digital signal processing (DSP), such as image processing[1,6,7], adaptive digital filtering[8] and interpolation[9]. The performance of DST can be compared to that of the discrete cosine transform (DCT) and it may therefore be considered as a viable alternative to the DCT. For images with high correlation, the DCT yields better results; however, for images with a low correlation of coefficients, the DST yields lower bit rates [10]. Yip and Rao [11] have proven that for large sequence length ($N \geq 32$) and low correlation coefficient ($\rho <$ 0.6), the DST performs even better than the DCT.

Several attempts have been made for efficient implementation of prime-length DST and DCT in systolic hardware through cyclic convolution formulation [12] – [16] due to its remarkable advantage over the others, particularly for efficient input / output and data transfer operations. The convolutions play a significant role in digital signal processing due to their nature of easy implementation. Moreover, the convolution-based algorithms are found to be efficient for read-only-memory (ROM)- based and adder-based very large scale integration (VLSI) implementation.

In this paper, a new prime-length DST algorithm using cyclic convolution is proposed. The N- point prime-length DST is then converted into a pair of $[(N-1)/2]$- point cyclic convolutions. This cyclic convolution-based algorithm is used to obtain a simple linear systolic array for prime-length DST.

The systolic architecture has the following characteristics:

- A massive and non-centralized parallelism
- Local communications
- Synchronous evaluation

The systolic arrays are used in the design and implementation of high performance digital signal processing equipment. Systolic architectures are established as the most popular and dominant class of VLSI structures due to the simplicity of their processing elements(PEs), modularity of their structure, regular and nearest neighbour interconnections between the PEs, High level of pipelinability, small chip area and lower dissipation .In the systolic architectures, the desired data are pumped rhythmically in regular intervals across the PEs for yielding high throughput by fully pipelined processing. The systolic array concept can also be exploited at bit level in the design of individual VLSI chips. The highly regular structure of systolic circuits renders them comparatively easy to design and test.

The rest of the paper is organized as follows. The proposed algorithm for DST is presented in Section-2.The basic idea of cyclic convolution is given in Section-3. Two examples for realizing prime-length DST are presented in Section-4.Conclusion is given in Section-5.

2. Proposed Algorithm for DST

The discrete sine transform for a sequence $\{y(i):i=1,2,\ldots,N\}$ is defined as

$$Y(k) = \sum_{i=1}^{N} y(i) \sin\left[\frac{k(2i-1)\pi}{2N}\right] \qquad \text{for } k = 1,2,\ldots,N \tag{1}$$

The y values represent the input data and the Y values represent the transformed data. Another sequence $\{x(i):i = 1,2,\ldots,N\}$ is defined as

$$\begin{aligned} x(N) &= y(N) \\ x(i) &= y(i) + x(i+1) \qquad \text{for } i = 1,2,\ldots,N-1 \end{aligned} \tag{2}$$

Using (2) in (1), we obtain

$$\begin{aligned} Y(k) &= \sum_{i=1}^{N-1} \left\{x(i) - x(i+1)\right\} \sin\left[\frac{k(2i-1)\pi}{2N}\right] + x(N) \sin\left[\frac{k(2N-1)\pi}{2N}\right] \\ &= \left\{x(1) + 2P(k)\right\} \sin\left(\frac{k\pi}{2N}\right) \end{aligned} \tag{3}$$

where, $$P(k) = \sum_{i=1}^{N-1} x(i+1) \cos\left(\frac{ik\pi}{N}\right) \qquad \text{for } k = 1,2,\ldots,N \tag{4}$$

3. Basic Idea of Cyclic Convolution

The following is an example of cyclic convolution

$$\begin{bmatrix} u1 \\ u2 \\ u3 \\ u4 \end{bmatrix} = \begin{bmatrix} a & b & c & d \\ d & a & b & c \\ c & d & a & b \\ b & c & d & a \end{bmatrix} . \begin{bmatrix} v1 \\ v2 \\ v3 \\ v4 \end{bmatrix} \tag{5}$$

where $\{v1, v2, v2, v3\}$ are input data, $\{a, b, c, d\}$ are coefficients, and $\{u1, u2, u3, u4\}$ are output data.

Using the commutative property of convolution, (5) can be written as

$$\begin{bmatrix} u1 \\ u2 \\ u3 \\ u4 \end{bmatrix} = \begin{bmatrix} v1 & v2 & v3 & v4 \\ v2 & v3 & v4 & v1 \\ v3 & v4 & v1 & v2 \\ v4 & v1 & v2 & v3 \end{bmatrix} . \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix} \tag{6}$$

Eq. (5) and (6) are both in cyclic convolution formulation. The cyclic convolution formulation has the property that the elements in the coefficient matrix to compute different output elements are same. This property is helpful in reducing the hardware complexity of the circuit during application. Observing (5), we find that different outputs can be computed from the input data with the same set of coefficients $\{a, b, c, d\}$ with rotated order. According to (6), different outputs can be computed using the same input data with rotated order and the same set of coefficients

4. Examples for Realizing Prime-Length DST Using Cyclic Convolution

Two examples for realizing 5-point and 7-point DSTs using cyclic convolution are given below to clarify the proposed algorithm.

4.1. DST for $N = 5$

From (4), $\{P(k) : k = 1, 2, 3, 4, 5\}$ for $N = 5$ is given by

$$\begin{bmatrix} P(2) \\ P(4) \\ P(3) \\ P(1) \end{bmatrix} = \begin{bmatrix} C(2) & -C(1) & C(2) & -C(1) \\ -C(1) & C(2) & -C(1) & C(2) \\ -C(2) & C(1) & C(2) & -C(1) \\ C(1) & -C(2) & -C(1) & C(2) \end{bmatrix} \begin{bmatrix} x(2) \\ x(4) \\ x(5) \\ x(3) \end{bmatrix} \tag{7}$$

$$P(5) = -x(2) + x(3) - x(4) + x(5) \qquad (8)$$

where $C(j) = \cos\left(\frac{j\pi}{N}\right)$

Define

$$Q_2 = \begin{bmatrix} C(2) & -C(1) \\ -C(1) & C(2) \end{bmatrix} \qquad (9)$$

Then (7) can be written as

$$\begin{bmatrix} W_1 \\ W_2 \end{bmatrix} = \begin{bmatrix} Q_2 & Q_2 \\ -Q_2 & Q_2 \end{bmatrix} \begin{bmatrix} X_1 \\ X_2 \end{bmatrix} = \begin{bmatrix} Q_2 & 0 \\ 0 & Q_2 \end{bmatrix} \begin{bmatrix} X_2 + X_1 \\ X_2 - X_1 \end{bmatrix} \qquad (10)$$

$$= \begin{bmatrix} Q_{(N-1)/2} & 0 \\ 0 & Q_{(N-1)/2} \end{bmatrix} \begin{bmatrix} X_2 + X_1 \\ X_2 - X_1 \end{bmatrix} \qquad (11)$$

where

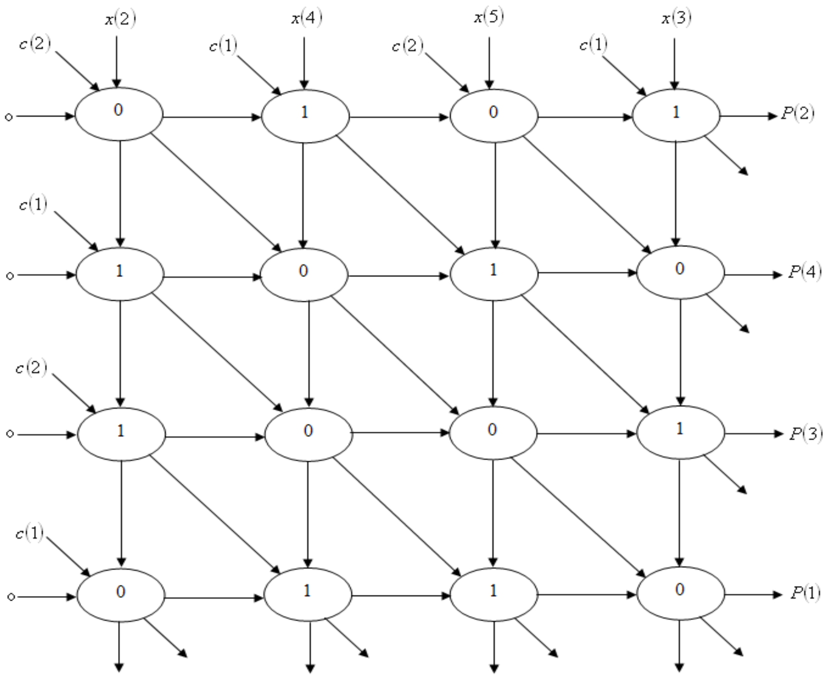
$$W_1 = \left[P(2) \quad P(4)\right]^T$$

$$W_2 = \left[P(3) \quad P(1)\right]^T \qquad (12\ a)$$

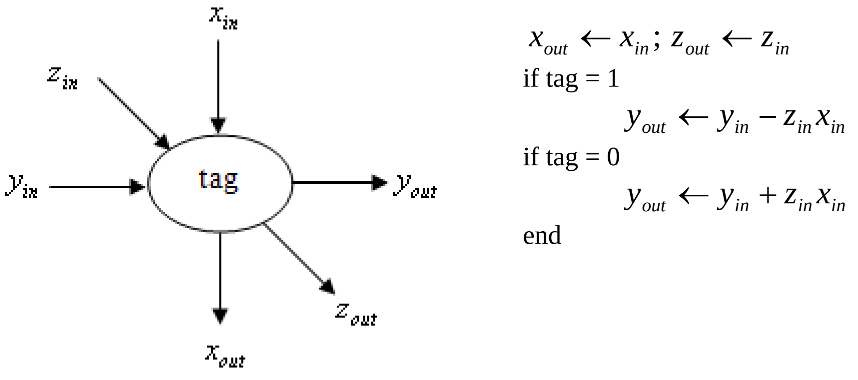
$$X_1 = \left[x(2) \quad x(4)\right]^T$$

$$X_2 = \left[x(5) \quad x(3)\right]^T \qquad (12\ b)$$

Eq. (7) is in cyclic convolution form. The dependent graph (DG) for (7) is shown in Fig.1(a). Function of each node of the DG is described in Fig.1(b). $P(5)$ can be realized using (8). Eq. (11) shows that the implementation of the cyclic convolution (7) is equivalent to two [(N-1) /2] – point cyclic convolutions. This means the implementation of prime N- length DST has been transformed into two [(N-1) /2] – point cyclic convolutions with the same kernel $Q_{(N-1)/2}$. Eq. (10) is realized using a systolic array shown in Fig. 2(a). The input values to a processing element (PE) are staggered by one cycle-period with respect to the preceding PE to maintain the data dependency requirement. The function of each PE of this systolic array is shown in Fig.2(b). Each PE performs two multiplications and two additions in each cycle-period.

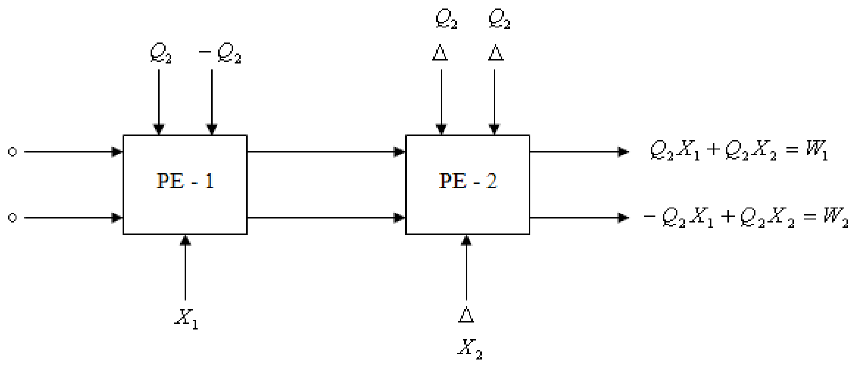


(a)

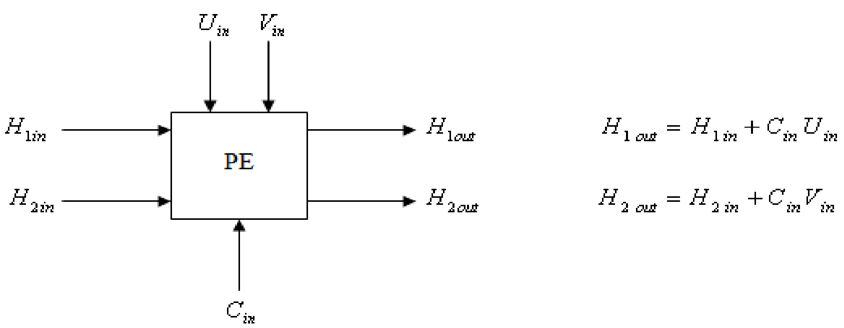


(b)

Fig. 1. (a) Dependent graph for (7) of DST algorithm of length $N = 5$. (b) Functions of nodes in (a).



(a)



(b)

Fig. 2. The linear systolic array for computing cyclic convolution (10). (a) The linear array. (b) Function of each PE. Δ stands for unit delay.

From(3), we obtain

$$\begin{bmatrix} Y(1) \\ Y(2) \\ Y(3) \\ Y(4) \\ Y(5) \end{bmatrix} = \begin{bmatrix} (2P(1)+x(1))S(1) \\ (2P(2)+x(1))S(2) \\ (2P(3)+x(1))S(3) \\ (2P(4)+x(1))S(4) \\ (2P(5)+x(1))S(5) \end{bmatrix} \tag{13}$$

where

$$S(n) = \sin\left(\frac{n\pi}{2N}\right) \tag{14}$$

The DST $\{Y(k): k = 1,2,3,4,5\}$ can be realized by using (7) and (8) in (13). Fig.3 shows the flow diagram for realization of DST components $\{Y(k): k = 1,2,3,4\}$.

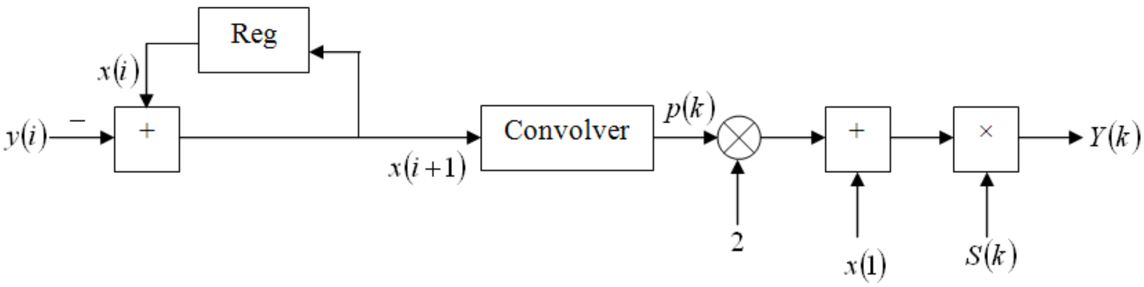


Fig. 3. Flow diagram for the implementation of the DST

4.2. DST for *N* = 7

From (4), $\{P(k): k = 1,2,3,4,5,6,7\}$ for *N* = 7 is written as

$$\begin{bmatrix} P(4) \\ P(2) \\ P(6) \\ P(3) \\ P(5) \\ P(1) \end{bmatrix} = \begin{bmatrix} -C(3) & -C(1) & C(2) & -C(3) & -C(1) & C(2) \\ C(2) & -C(3) & -C1) & C(2) & -C(3) & -C(1) \\ -C(1) & C(2) & -C(3) & -C(1) & C(2) & -C(3) \\ C(3) & C(1) & C(2) & -C(3) & -C(1) & -C(2) \\ -C2) & C(3) & -C(1) & C(2) & -C(3) & C(1) \\ C(1) & -C(2) & -C(3) & -C(1) & C(2) & C(3) \end{bmatrix} \begin{bmatrix} x(2) \\ x(6) \\ x(5) \\ x(7) \\ x(3) \\ x(4) \end{bmatrix} \tag{15}$$

$$P(7) = -x(2) + x(3) - x(4) + x(5) - x(6) + x(7) \tag{16}$$

where $C(j) = \cos\left(\frac{j\pi}{N}\right)$

Define

$$Q_3 = \begin{bmatrix} -C(3) & -C(1) & C(2) \\ C(2) & -C(3) & -C(1) \\ -C(1) & C(2) & -C(3) \end{bmatrix}, \quad R_3 = \begin{bmatrix} C(3) & C(1) & C(2) \\ -C(2) & C(3) & -C(1) \\ C(1) & -C(2) & -C(3) \end{bmatrix}$$

(17)

$$\text{and} \quad A_3 = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Then (15) can be expressed as

$$\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} Q_3 & Q_3 \\ R_3 & -R_3 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} \tag{18}$$

Since $R_3 = Q_3 A_3$, (18) can be written as

$$\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} Q_3 & Q_3 \\ Q_3 A_3 & -Q_3 A_3 \end{bmatrix} \begin{bmatrix} U_1 \\ U_2 \end{bmatrix} = \begin{bmatrix} Q_3 & 0 \\ 0 & Q_3 \end{bmatrix} \begin{bmatrix} U_1 + U_2 \\ A_3(U_1 + U_2) \end{bmatrix} \tag{19}$$

$$\begin{bmatrix} T_1 \\ T_2 \end{bmatrix} = \begin{bmatrix} Q_{(N-1)/2} & 0 \\ 0 & Q_{(N-1)/2} \end{bmatrix} \begin{bmatrix} U_1 + U_2 \\ A_3(U_1 - U_2) \end{bmatrix} \tag{20}$$

where

$$T_1 = \begin{bmatrix} P(4) & P(2) & P(6) \end{bmatrix}^T$$

$$T_2 = \begin{bmatrix} P(3) & P(5) & P(1) \end{bmatrix}^T \qquad (21a)$$

$$U_1 = \begin{bmatrix} x(2) & x(6) & x(5) \end{bmatrix}^T$$

$$U_2 = \begin{bmatrix} x(7) & x(3) & x(4) \end{bmatrix}^T \qquad (21b)$$

Eq. (15) is in cyclic convolution form. $P(7)$ can be realized using (16). Eq. (18) can be realized by the systolic array. Eq. (20) shows that the implementation of the cyclic convolution (15) is equivalent to two [(N-1) /2] – point cyclic convolutions. This means the implementation of prime N- length DST has been transformed into two [(N-1) /2] – point cyclic convolutions with the same kernel $Q_{(N-1)/2}$.

From (3), we get

$$\begin{bmatrix} Y(1) \\ Y(2) \\ Y(3) \\ Y(4) \\ Y(5) \\ Y(6) \\ Y(7) \end{bmatrix} = \begin{bmatrix} (2P(1)+x(1))S(1) \\ (2P(2)+x(1))S(2) \\ (2P(3)+x(1))S(3) \\ (2P(4)+x(1))S(4) \\ (2P(5)+x(1))S(5) \\ (2P(6)+x(1))S(6) \\ (2P(7)+x(1))S(7) \end{bmatrix} \qquad (22)$$

where

$$S(n) = \sin\left(\frac{n\pi}{2N}\right)$$

The DST $\{Y(k) : k = 1,2,3,4,5,6,7\}$ can be realized by using (15) and (16) in (22).

5. Conclusion

A new algorithm is derived to compute prime-length DST of size N from a pair of $[(N-1)/2]$- point cyclic convolutions. This reduces the hardware cost. A simple linear array is presented for systolic implementation of these cyclic convolutions. The convolutions play a significant role in digital signal processing due to their nature of easy implementation. Moreover, the convolution-based algorithms are found to be efficient for read-only-memory (ROM)- based and adder-based very large scale integration (VLSI) implementation.

References

[1] A.K. Jain, "A fast Karhunen-Loeve transform for a class of random processes," IEEE Trans. Commun., vol. COM-24, pp 1023-1029, September 1976.

[2] H.B. Kekre and J.K. Solanka, "Comparative performance of various trigonometric unitary transforms for transform image coding," Int. J. Electron., vol. 44, pp 305-315, 1978.

[3] A.K. Jain, "A sinusoidal family of unitary transforms," IEEE Trans. Patt. Anal. Machine Intell., vol. PAMI-I, pp 356-365, September 1979.

[4] Z. Wang and B. Hunt, "The discrete W transform," Applied Math Computat., vol. 16, pp 19-48, January 1985.

[5] S. Poornachandra, V. Ravichandran, and N.Kumarvel, "Mapping of discrete cosine transform (DCT) and discrete sine transform (DST) based on symmetries" IETE Journal of Research, Vol. 49, no. 1, pp 35-42, January-February 2003.

[6] S. Cheng, "Application of the sine transform method in time of flight positron emission image reconstruction algorithms," IEEE Trans. BIOMED. Eng., vol. BME-32, pp 185-192, March 1985.

[7] K. Rose, A. Heiman, and I. Dinstein, "DCT/DST alternate transform image coding," Proc. BLOBE COM 87, vol. I, pp. 426-430, November 1987.

[8] J.L. Wang and Z.Q. Ding, "Discrete sine transform domain LMS adaptive filtering," Proc. Int. Conf. Acoust., Speech, Signal Processing, pp 260-263, 1985.

[9] Z. Wang and L. Wang, "Interpolation using the fast discrete sine transform," Signal Processing, vol. 26, pp 131-137, 1992.

[10] A. K. Jain, Fundamentals of Digital Image Processing. Englewood Cliffs, NJ: Prentice - Hall, 1989.

[11] P. Yip and K.R. Rao, "On the computation and the effectiveness of discrete sine transform", Comput. Electron., vol. 7, pp. 45-55, 1980.

[12] D.F.Chiper, "A new systolic array algorithm for memory-based VLSI array implementation of DCT," in Proc. 2nd IEEE Symp. Comput. Commun., Jul.1997, pp.297-301.

[13] C.Cheng and K.K.Parthi, "A novel systolic array structure for DCT", IEEE Trans. Circuits Syst.II, Exp. Briefs, vol.52, no.7, pp.366-369, Jul.2005.

[14] D.F.Chiper, M.N.S. Swamy, M.O.Ahmad, and T.Stouraitis,“ A systolic array architecture for the discrete sine transform,” IEEE Trans. Signal Process., vol.50, no.9, pp.2347-2354, Sep.2002.

[15] D.F.Chiper, M.N.S. Swamy, M.O.Ahmad, and T.Stouraitis, “Systolic algorithms and a memory-based design approach for a unified architecture for the computation of DCT/DST/IDCT/IDST,” IEEE Trans. Circuits Syst.I, Reg. Papers, vol.52, no.6, pp.1125-1137, Jun.2005.

[16] P.K.Meher, ”Systolic designs for DCT using a low-complexity concurrent convolutional formulation,” IEEE Trans. Circuits Syst. Video Technol., vol.16, no.9, pp.1041-1050, Sep.2006.