Computation of DCT From Prime-Factor DHT

Ajit Kumar Patro[1], Dr. Sudhansu Sekhar Nayak[2]

[1]*Research Scholar, Centurion University of Technology and Management, Paralakhemundi, Odisha, India*

[2]*Professor, Centurion University of Technology and Management, Paralakhemundi, Odisha, India*

Abstract - The prime-factor decomposition technique for fast computation of discrete cosine transform (DCT) is popularly used because it is convenient to deal with the resulting groups of small size. The memory for data storage in a DSP processor is expensive. By the prime-factor approach it is possible to implement the long-length DCT by processors of small memory as short-length DCTs are implemented one after the other. In this paper, we have presented the scheme for high throughput computation of prime-factor DCT from DHT. The area-complexity, computation time and VLSI performance measure of the proposed architecture are (N+1)2/4, (N+1) and (N+1)4/4, respectively.

Keywords- Discrete Cosine Transform, Discrete Hartley Transform, Prime-Factor Decomposition, Index Mapping.

I. INTRODUCTION

The prime-factor decomposition technique is popularly used for fast computation of digital convolution and discrete orthogonal transforms. Prime-factor decomposition approach has, therefore, been tried for efficient computation of the DCT. The main theoretical rationale of this technique is to convert N-point DCT into a two-dimensional $(N_1 \times N_2)$-points DCT by employing certain index mapping where $N = (N_1 \times N_2)$. Then we can deal with the resulting groups of small size problems in each dimension. In a DSP processor the memory for data storage is always expensive. By the prime-factor approach it is possible to implement the long-length DCT by processors of small memory as short-length DCTs are implemented one after the other. In addition, when this approach is combined with efficient short-length algorithms the computational complexity is reduced considerably. Cho and Lee [1] derived prime-factor DCT algorithm based on various DFT algorithms which requires complex-number multiplications. Yang and Narasimha [2] proposed a prime-factor DCT algorithm which included only real-number multiplications. However, its index mapping was complicated. Lee [3] presented input and output index mappings for a prime-factor decomposed computation of DCT. However, his input index mapping is realized by constructing and combining two index tables, which occupy additional memory space and would be infeasible in variable-size applications. Chakrabarti and Ja`Ja` [4] developed a systolic architecture for implementing Lee's algorithm. They wanted to compute the DCT from DHT. So they modified the index mappings which are essentially the same as Lee's. However, they did not discuss the actual implementation of these index mappings. Lee and Huang [5] suggested a scheme for prime-factor decomposition of the DCT which involves simpler and more efficient index mapping compared with those of [2, 3], and is devoid of complex arithmetic operations as well. Also they proposed two systolic architectures comprising of two matrix multiplication units and a transposition unit. We have presented the scheme for high throughput computation of prime-factor DCT from DHT.

II. PRIME-FACTOR DECOMPOSITION OF DHT

For the transform length $N = (N_1 \times N_2)$ where N_1 and N_2 are relatively prime, the index k and n may uniquely be mapped into pairs of indices (k_1, k_2) and (n_1, n_2) respectively, according to the following relations [6]

$$k = (k_1 N_2 s_1 + k_2 N_1 s_2) \mod N \quad (1)$$

$$n = (n_1 N_2 + n_2 N_1) \mod N \quad (2)$$

for k_1 and $n_1 = 0, 1, 2, \ldots\ldots, N_1-1$

and k_2 and $n_2 = 0, 1, 2, \ldots\ldots, N_2-1$

where

$$N_2 s_1 = 1 \mod N_1 \quad (3)$$

and

$$N_1 s_2 = 1 \mod N_2 \quad (4)$$

The DHT of sequence $\{x(n),\ n = 0, 1, 2, \ldots\ldots, N-1\}$ may be defined as

$$H(k) = \sum_{n=0}^{N-1} x(n)\left(\cos\frac{2\pi kn}{N} + \sin\frac{2\pi kn}{N}\right) \quad (5)$$

for $k = 0, 1, 2, \ldots\ldots, N-1$

Using equations (1) - (4), equation (5) may be expressed as

$$H(k_1, k_2) = \sum_{n_2=0}^{N_2-1}\sum_{n_1=0}^{N_1-1} x(n_1, n_2)\left[\cos 2\pi\left(\frac{k_1 n_1}{N_1} + \frac{k_2 n_2}{N_2}\right) + \sin 2\pi\left(\frac{k_1 n_1}{N_1} + \frac{k_2 n_2}{N_2}\right)\right] \quad (6)$$

Equation (6) may be otherwise be written as

$$H(k_1, k_2) = \sum_{n_2=0}^{N_2-1}\left[\sum_{n_1=0}^{N_1-1} x(n_1, n_2)\cos\frac{2\pi k_1 n_1}{N_1} + \sum_{n_1=0}^{N_1-1} x(n_1, N_2 - n_2)\sin\frac{2\pi k_1 n_1}{N_1}\right]\operatorname{cas}\frac{2\pi k_2 n_2}{N_2} \quad (7)$$

assuming $x(n_1, N_2) = x(n_1, 0)$
Besides, it can be found that,

$$\sum_{n_1=0}^{N_1-1} x(n_1,n_2)\cos\frac{2\pi k_1 n_1}{N_1} = \frac{1}{2}\sum_{n_1=0}^{N_1-1}[x(n_1,n_2) + x(N_1-n_1,n_2)]\,\mathrm{cas}\frac{2\pi k_1 n_1}{N_1} \quad (8)$$

and

$$\sum_{n_1=0}^{N_1-1} x(n_1,N_2-n_2)\sin\frac{2\pi k_1 n_1}{N_1} = \frac{1}{2}\sum_{n_1=0}^{N_1-1}[x(n_1,N_2-n_2) - x(N_1-n_1,N_2-n_2)]\,\mathrm{cas}\frac{2\pi k_1 n_1}{N_1} \quad (9)$$

Using equations (8) and (9) on equation (7), one may have,

$$H(k_1,k_2) = \sum_{n_2=0}^{N_2-1}\sum_{n_1=0}^{N_1-1} y(n_1,n_2)\,\mathrm{cas}\frac{2\pi k_1 n_1}{N_1}\,\mathrm{cas}\frac{2\pi k_2 n_2}{N_2} \quad (10)$$

where

$$y(n_1,n_2) = \frac{1}{2}[x(n_1,n_2) + x(N_1-n_1,n_2) + x(n_1,N_2-n_2) - x(N_1-n_1,N_2-n_2)] \quad (11)$$

III. ALGORITHM FOR SYSTOLIC IMPLEMENTATION OF THE DHT

The arguments of sine and cosine function of equation (6) may be expanded to yield

$$H(k_1,k_2) = \sum_{n_2=0}^{N_2-1} W(k_1,n_2)\left[\cos\frac{2\pi k_2 n_2}{N_2} + \sin\frac{2\pi k_2 n_2}{N_2}\right] \quad (12)$$

where

$$W(k_1,n_2) = \sum_{n_1=0}^{N_1-1} y(n_1,n_2)\,\mathrm{cas}\frac{2\pi k_1 n_1}{N_1} \quad (13)$$

Equation (12) can be expressed as

$$H(k_1,k_2) = \sum_{n_2=0}^{(N_2-1)/2}\left[Z_1(k_1,n_2)\cos\frac{2\pi k_2 n_2}{N_2} + Z_2(k_1,n_2)\sin\frac{2\pi k_2 n_2}{N_2}\right]$$

for $k_1 = 0, 1, 2, \ldots\ldots, (N_1-1)/2$
and $k_2 = 0, 1, 2, \ldots\ldots\ldots, (N_2-1)/2$ (14a)

and

$$H(k_1,N_2-k_2) = \sum_{n_2=0}^{(N_2-1)/2}\left[Z_1(k_1,n_2)\cos\frac{2\pi k_2 n_2}{N_2} - Z_2(k_1,n_2)\sin\frac{2\pi k_2 n_2}{N_2}\right]$$

for $k_1 = 0, 1, 2, \ldots\ldots, (N_1-1)/2$
and $k_2 = 0, 1, 2, \ldots\ldots, (N_2-1)/2$ (14b)

where

$$Z_1(k_1,n_2) = W(k_1,n_2) + W(k_1,N_2-n_2)$$
$$Z_2(k_1,n_2) = W(k_1,n_2) - W(k_1,N_2-n_2)$$

for $n_2 = 1, 2, \ldots\ldots, (N_1-1)/2$

$$Z_1(k_1,0) = Z_2(k_1,0) = W(k_1,0) \quad (15)$$

Using equations (13) and (11), equation (15) may be simplified to yield

$$Z_1(k_1,n_2) = \sum_{n_1=0}^{(N_1-1)/2}\left[A(n_1,n_2)\cos\frac{2\pi k_1 n_1}{N_1} + B(n_1,n_2)\sin\frac{2\pi k_1 n_1}{N_1}\right] \quad (16)$$

and

$$Z_2(k_1,n_2) = \sum_{n_1=0}^{(N_1-1)/2}\left[C(n_1,n_2)\cos\frac{2\pi k_1 n_1}{N_1} + D(n_1,n_2)\sin\frac{2\pi k_1 n_1}{N_1}\right] \quad (17)$$

where

$$A(n_1,n_2) = [x(n_1,n_2) + x(N_1-n_1,n_2) + x(n_1,N_2-n_2) - x(N_1-n_1,N_2-n_2)] \quad (18)$$
$$B(n_1,n_2) = [x(n_1,n_2) + x(n_1,N_2-n_2) - x(N_1-n_1,n_2) - x(N_1-n_1,N_2-n_2)] \quad (19)$$
$$C(n_1,n_2) = [x(n_1,n_2) + x(N_1-n_1,n_2) - x(n_1,N_2-n_2) - x(N_1-n_1,N_2-n_2)] \quad (20)$$
$$D(n_1,n_2) = [x(N_1-n_1,n_2) + x(n_1,N_2-n_2) - x(n_1,n_2) - x(N_1-n_1,N_2-n_2)] \quad (21)$$

for $n_1 = 1, 2, \ldots\ldots, (N_1-1)/2$
and $n_2 = 1, 2, \ldots\ldots, (N_2-1)/2$
and

$$A(0,n_2) = B(0,n_2) = x(0,n_2) + x(0,N_2-n_2)$$
$$C(0,n_2) = D(0,n_2) = x(0,n_2) - x(0,N_2-n_2)$$

for $n_2 = 1, 2, \ldots\ldots, (N_2-1)/2$

$$A(n_1,0) = C(n_1,0) = x(n_1,0) + x(N_1-n_1,0)$$
$$B(n_1,0) = D(n_1,0) = x(n_1,0) + x(N_1-n_1,0)$$

for $n_1 = 1, 2, \ldots\ldots\ldots, (N_1-1)/2$

$$A(0,0) = B(0,0) = C(0,0) = D(0,0) = x(0,0) \quad (22)$$

Substituting $(N_1\text{-}k_1)$ for k_1 in equations (14a) and (14b), respectively, one may have

$$H(N_1-k_1,k_2) = \sum_{n_2=0}^{(N_2-1)/2} \left[Z_1(N_1-k_1,n_2)\cos\frac{2\pi k_2 n_2}{N_2} + Z_2(N_1-k_1,n_2)\sin\frac{2\pi k_2 n_2}{N_2}\right]$$

for k_1 = 1, 2, ……, (N_1-1)/2
and k_2 = 1, 2, ……, (N_2-1)/2 (23a)
and

$$H(N_1-k_1,N_2-k_2) = \sum_{n_2=0}^{(N_2-1)/2} \left[Z_1(N_1-k_1,n_2)\cos\frac{2\pi k_2 n_2}{N_2} - Z_2(N_1-k_1,n_2)\sin\frac{2\pi k_2 n_2}{N_2}\right]$$

for k_1 = 1, 2, ………, (N_1-1)/2
and k_2 = 1, 2, ………, (N_2-1)/2 (23b)
where

$$Z_1(N_1-k_1,n_2) = W(N_1-k_1,n_2) + W(N_1-k_1,N_2-n_2) \quad (24)$$

$$Z_2(N_1-k_1,n_2) = W(N_1-k_1,n_2) - W(N_1-k_1,N_2-n_2) \quad (25)$$

for n_1 = 1, 2, ………, (N_1-1)/2

$$Z_1(N_1-k_1,0) = Z_2(N_1-k_1,0) = W(N_1-k_1,0)$$

Using equations (11) and (13), equations (24) and (25) can be simplified to yield

$$Z_1(N_1-k_1,n_2) = \sum_{n_1=0}^{(N_1-1)/2} \left[A(n_1,n_2)\cos\frac{2\pi k_1 n_1}{N_1} - B(n_1,n_2)\sin\frac{2\pi k_1 n_1}{N_1}\right] \quad (26)$$

and

$$Z_2(N_1-k_1,n_2) = \sum_{n_1=0}^{(N_1-1)/2} \left[C(n_1,n_2)\cos\frac{2\pi k_1 n_1}{N_1} - D(n_1,n_2)\sin\frac{2\pi k_1 n_1}{N_1}\right] \quad (27)$$

IV. COMPUTATION OF PRIME-FACTOR DCT FROM DHT

The DCT of sequence {$x(n)$, n = 0, 1, 2, ……, N-1} may be defined as [8]

$$X(k) = \frac{2}{N}\varepsilon(k)\sum_{n=0}^{N-1} x(n)\cos\left[\frac{\pi(2n+1)k}{2N}\right] \quad (28)$$

and the inverse discrete cosine transform (IDCT) is given by

$$x(n) = \frac{2}{N}\sum_{k=0}^{N-1} \varepsilon(k)X(k)\cos\left[\frac{\pi(2n+1)k}{2N}\right] \quad (29)$$

for k = 0, 1, 2, ……, N-1

where

$$\varepsilon(k) = \begin{cases} (2)^{-1/2} & for\ k=0 \\ 1 & for\ 1 \le k \le N-1 \end{cases}$$

Since ε(k) effects only the amplitude of X(0) component, we shall take ε(k) as unity with X(0) scaled up by √2.
It is shown in [7] that the DCT defined by equation (28) can be expressed as

$$X(k) = \frac{1}{2}[H(k)\,\mathrm{cas}(-k\pi/2N) + H(N-k)\,\mathrm{cas}(k\pi/2N)] \quad (30)$$

where {$H(k)$} represents N-point DHT of {$\bar{x}(n)$} given that

$$\bar{x}(n) = \begin{cases} x(2n) & 0 \le n \le \left(\frac{N}{2}\right)-1 \\ x(2N-2n-1) & \left(\frac{N}{2}\right) \le n \le N-1 \end{cases} \quad (31)$$

The prime-factor DCT may thus be obtained from DHT as

$$X(k_1,k_2) = \frac{1}{2}[H(k_1,k_2)\,\mathrm{cas}(-k\pi/2N) + H(N_1-k_1,N_2-k_2)\,\mathrm{cas}(k\pi/2N)] \quad (32)$$

where k is given by equation (1).

V. PROPOSED 2-D ARCHITECTURE

The proposed 2-D architecture for computation of ($N_1 \times N_2$)-point DHT consists (N_2+1)/2 linear arrays as shown Fig. 1. Each linear array consists of (N_1+1)/2 number of locally connected identical PEs. The function of each PE is depicted in Fig. 2. The elements of n_2th column of $A(n_1, n_2)$, $B(n_1, n_2)$, $C(n_1, n_2)$, and $D(n_1, n_2)$ are fed to the (n_2+1)th array staggered by one time-step with respect to the input of n_2th array. In the first (n_1+1)/2 time-steps, each PE makes the first stage of computation to provide the intermediate results [$Z_1(k_1, n_2)$], [$Z_1(N_1-k_1, n_2)$], [$Z_2(k_1, n_2)$], and [$Z_2(N_1-k_1, n_2)$] of size [(N_1+1)/2 × (N_2+1)/2]. In the next (N_2+1)/2 time-steps, the second stage of computation yields the desired DHT components. From the PEs of the last array, the structure provides four DHT components $H(k_1, k_2)$, $H(N_1-k_1, k_2)$, $H(N_1, N_2-k_2)$, and $H(N_1-k_1, N_2-k_2)$ simultaneously. Therefore, the DCT can conveniently be computed from the output of the structure according to equation (32).

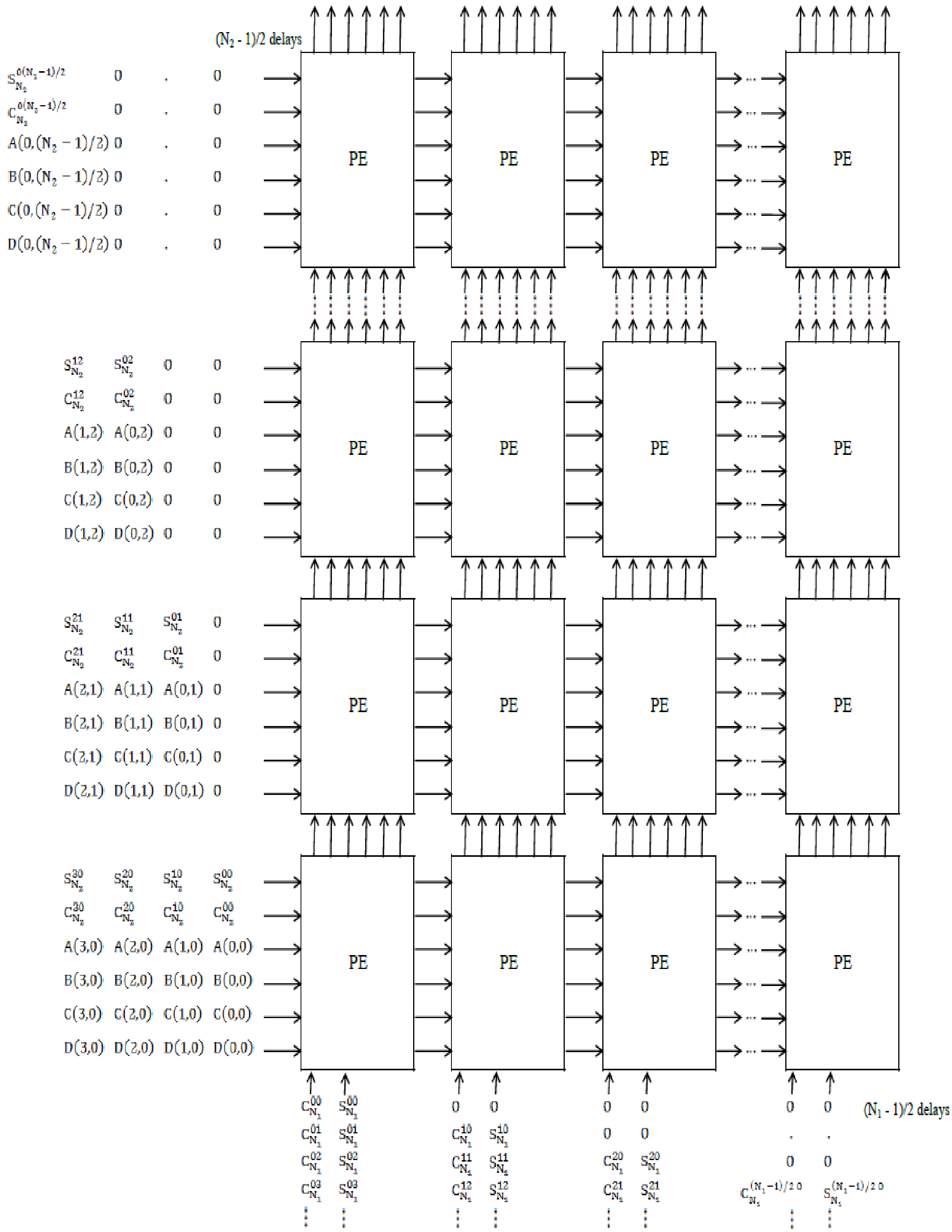

Fig.1: Structure of the DHT Architecture

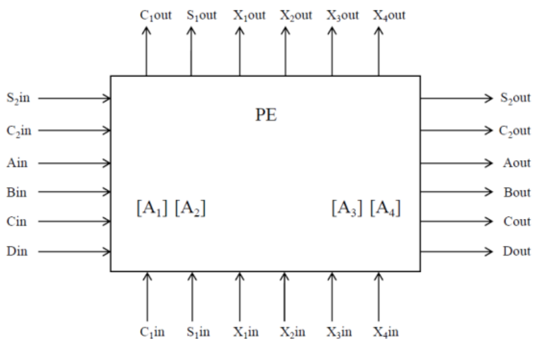

Fig.2: Function of a Processing Element

1. $P_1 = A_1 + A_2$
   $P_2 = A_3 + A_4$
   $P_3 = A_1 - A_2$
   $P_4 = A_3 - A_4$
   $A_1 = A_2 = A_3 = A_4 = 0$
   Count = 0
2. $A_1 = A_1 + Ain \cdot C_1in$
   $A_2 = A_2 + Bin \cdot S_1in$
   $A_3 = A_3 + Cin \cdot C_1in$
   $A_4 = A_4 + Din \cdot S_1in$
   Aout = Ain
   Bout = Bin
   Cout = Cin
   Dout = Din
   Count = Count + 1
   if (Count = $(N_1+1)/2$) then
   goto 1
   else
   goto 2
   end if

For continuous processing in every time-step
$C_2out = C_2in$
$S_2out = S_2in$
$X_1out = X_1in + P_1 \cdot C_2in$
$X_2out = X_2in + P_2 \cdot S_2in$
$X_3out = X_3in + P_3 \cdot C_2in$
$X_4out = X_4in + P_4 \cdot S_2in$
$C_1in = \cos(2\pi k_1 n_1/N_1)$
$C_2in = \cos(2\pi k_2 n_2/N_2)$
$S_1in = \sin(2\pi k_1 n_1/N_1)$
$S_2in = \sin(2\pi k_2 n_2/N_2)$

TABLE 1 COMPARISON OF AREA-COMPLEXITY, COMPUTATION TIME AND VLSI PERFORMANCE MEASURE OF THE PROPOSED STRUCTURE WITH THE STRUCTURE OF [5].

Structures	**Area Complexity (A)**	**Computation Time (τ)**	**VLSI Performance Measure $t(A\tau^2)$**
Structure of [5]	$3N^2$	$3N$	$27N^4$
Structure of the DHT architecture (Fig. 1)	$(N+1)^2/4$	$(N+1)$	$(N+1)^4/4$

VI. HARDWARE AND THROUGHPUT CONSIDERATIONS

The proposed DHT architecture (Fig. 1) requires $(N_1+1)(N_2+1)/4$ number of PEs. There are eight multipliers, eight adders and four accumulators in each PE. It gives the first DHT component after $\left[\frac{1}{2}(N_1+N_2)+1\right]$ time-steps. First set of DHT component is obtained in (N_1+N_2-1) time-steps. However, successive sets of DHT are obtained in every $(N_1+1)/2$ time-steps. The throughput rate of the structure would, therefore, be $2(N_1+1)/T$ where T is the duration of the time-step. The transposition of the intermediate output is avoided here so as to save the hardware for transposition, to reduce the chip area and latency.

VII. CONCLUSION

In this paper we have presented a scheme for high throughput computation of prime-factor DCT from DHT. The area-complexity, computation time and VLSI performance measure of the proposed structure with the structure of [5] are listed. The transposition of the intermediate output is also avoided in the DHT structure (Fig. 1) so as to save the hardware for transposition, to reduce the chip area and latency.

VIII. REFERENCES

[1]. Cho Nam Ik, Lee Sang Uk. DCT Algorithms for VLSI Parallel Implementations. IEEE Transactions on Acoustics, Speech, and Signal Processing. 1990 January, 38 (1), pp. 121-27.
[2]. Yang P P N, Narasimha M J. Prime Factor Decomposition of the Discrete Cosine Transform and its Hardware Realization. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP). 1985 April, 10, pp. 772-75.
[3]. Lee Byeong G. Input and Output Index Mappings for a Prime-Factor Decomposed Computation of Discrete Cosine Transform. IEEE Transactions on Acoustics, Speech and Signal Processing. 1989 February, 37 (2), pp. 237-44.
[4]. Chakrabarti C, Ja`Ja` J. Systolic Architectures for the Computation of the Discrete Hartley and Discrete Cosine Transforms based on Prime Factor Decomposition. IEEE Transactions on Computers. 1990 November, 39 (11), pp. 1359-68.
[5]. Lee Pei-Zong, Huang Fang-Yu. An Efficient Prime-Factor Algorithm for the Discrete Cosine Transform and its Hardware Implementations. IEEE Transactions on Signal Processing. 1994 August, 42 (8), pp. 1996-2005.
[6]. Burrus C S. Index Mappings for Multidimensional Formulation of the DFT and Convolution. IEEE Transactions on Acoustics, Speech, and Signal Processing. 1977 June, 25 (3), pp. 239-42.
[7]. Malvar H S. Fast Computation of Discrete Cosine Transform through Fast Hartley Transform. Electronics Letters . 1986 March, 22 (7), pp. 352-53.
[8]. Ahmed N, Natarajan T, Rao K R. Discrete Cosine Transform. IEEE Transactions on Computers. 1974 January, C-23 (1), pp. 90-93.