

論文94-31B-10-2

DCT/DST/DHT 하드웨어 구현을 위한 2차원 시스톨릭 어레이 (Two-Dimensional Systolic Arrays for DCT/DST/DHT Hardware Implementation)

潘 聲 範*, 朴 來 弘**

(Sung Bum Pan and Rae-Hong Park)

要 約

본 논문에서는 DCT/DST/DHT 계산을 위한 두 종류의 2차원 시스톨릭 어레이 구조를 제안한다. DCT/DST/DHT에 모두 적용 가능한 제안한 첫번째 구조는 N point DCT/DST/DHT를 짝수 주파수 성분과 홀수 주파수 성분으로 나누어 독립적인 계산을 수행하여 병렬처리가 가능하도록 하였다. 그리고 IDCT/IDST/ IDHT에도 사용이 가능하다. 제안한 두번째 구조는 첫번째 구조를 변형하여 DHT/IDHT 계산시 하드웨어 크기와 속도면에서 장점을 갖는 구조이다. 제안한 두 구조에서는 실수 곱셈과 덧셈연산을 하고 출력이 순서적으로 나온다. 기존의 구조와 비교할 때 제안한 구조는 PE의 간단함, 소요시간, 그리고 출력 생성 주기면에서 장점을 갖고 있다. 그리고 제안한 구조를 하드웨어 표준 기술언어인 VHDL을 이용하여 실험하여 효과적으로 동작함을 확인하였다.

Abstract

We propose two architectures using two dimensional systolic arrays for the DCT/DST/DHT. One decomposes the N -point DCT/DST/DHT into even-and odd-numbered frequency samples, and then computes them independently at the same time. In addition, the proposed architecture can be used for the IDCT/IDST/IDHT. Another is the modified version for the DHT/IDHT. Two proposed architectures generate outputs sequentially using real multiplications and additions. As compared to the conventional methods the proposed systolic arrays exhibit many advantages in terms of simplicity of the processing element (PE), latency, and throughput. The simulation results using VHDL, international standard language for hardware description, show the effectiveness of the proposed architecture.

1. 서론

* 準會員, ** 正會員 西江大學校 電子工學科

(Dept. of Elec. Eng., Sogang Univ.)

接受日字 : 1994年 1月 4日

변환 부호화 (transform coding)는 음성과 영상 신호의 처리 및 전송 시스템 등 데이터 감축을 위한

응용 분야에 많이 사용되고 있다. 최근 ISDN (Integrated Services Digital Network)과 HDTV (High Definition TeleVision)로의 발전이 진행되는 동안 영상신호의 실시간 처리 필요성이 크게 부각되었다. 많은 직교변환 중에서 DCT (Discrete Cosine Transform)^[1,2], DST (Discrete Sine Transform), 그리고 DHT (Discrete Hartley Transform)^[3]는 음성 및 영상 신호처리 분야에서 매우 효과적인 변환으로 알려져 있다.

DCT의 성능은 상관 계수가 높은 신호에 대하여 데이터간의 상관관계를 줄이는데 있어 최적 변환인 KLT(Karhunen Loève Transform)의 성능에 근접하고 있다.^[4-7] DST는 상관 계수가 낮은 1차 Markov열 (Markov sequence)에 적용할 때 KLT에 근접한 성능을 보인다. DHT는 복소수 연산이 필요한 DFT (Discrete Fourier Transform)를 대체할 수 있는 변환으로 DCT, DST와 마찬가지로 실수 연산만이 필요하다.^[8-15]

DCT, DST, 그리고 DHT는 butterfly 구조^[4,8,9]와 시스틀릭 어레이 (systolic array) 구조^[5,7,10-14]를 이용하는 두가지 접근 방식이 있다. Butterfly 구조는 전역적 데이터 교환이 필요하기 때문에 계산 시간이 증가하고 하드웨어 구현시 면적이 증가하는 단점을 갖고 있다. Butterfly 구조의 이러한 단점을 제거하기 위해서는 데이터 교환이 국부적으로만 일어나는 구조가 필요하다. 시스틀릭 어레이는 이러한 성질이 만족되는 구조로 하드웨어 구현이 용이하다.^[15-16] 시스틀릭 어레이는 VLSI기술을 이용하여 특정한 알고리즘의 수행 속도를 향상시키기 위해 최대한의 동시실행 (concurrency)을 이룬 전용 하드웨어 구조이다. 이의 특징은 모듈성 (modularity), 규칙성 (regularity), 국부적 연결성 (local interconnection), 고도의 중속 연결성 (pipelining), 잘 동기된 다중처리 (multiprocessing) 등이다.

본 논문에서는 DCT, DST, 그리고 DHT가 계산량이 많아 실시간 처리가 어렵기 때문에 이와 같은 문제점을 해결하기 위하여 이들에 모두 적용이 가능한 시스틀릭 어레이 구조를 제안하였다. 그리고 제안한 구조는 변환뿐만 아니라 역변환에도 적용이 가능하다. 그리고 DHT에 적용 가능한 변형된 2차원 시스틀릭 어레이 구조를 제안하였는데 이 구조는 거의 비슷한 두 종류의 PE(processing element)를 사용하여 앞에서 제안한 구조의 약 1/2개의 PE를 사용하여 DHT를 구할 수 있다. 이 구조도 IDHT (Inverse DHT)에 적용이 가능하다.

제안한 두종류의 2차원 시스틀릭 어레이 구조는 실

수 연산만이 필요한 간단한 PE가 필요하고 기존의 구조와 비교할 때 소요시간 (latency)과 출력 생성 주기 (throughput)가 짧은 장점을 갖고 있다. 제안한 구조의 타당성을 알아보기 위하여 멘토 그래픽스사(Mentor Graphics Corporation)의 VHDL 시뮬레이터인 QuickSimTMII를 이용하여 행위 단계 (behavioral level)로 모델링하였고 시뮬레이션하였다.^[17-18] 멘토 그래픽스사 (Mentor Graphics Corporation)의 AutoLogicTM을 이용하여 제안한 DCT 구조를 합성하였다.

본 논문의 II장에서는 DCT와 DST의 제안한 2차원 시스틀릭 어레이 구조를 보이고 III장에서는 II장의 구조를 이용한 DHT와 이를 변형하여 적은 개수의 PE와 소요시간과 출력 생성 주기가 짧은 새로운 구조를 제안한다. 그리고 IV장에서는 제안한 DCT 구조와 기존 DCT 구조와의 성능을 분석하고 V장에서 결론을 내린다.

II. DCT와 DST의 시스틀릭 어레이 구조

본 장에서는 DCT와 DST 계산을 위한 2차원 시스틀릭 어레이 구조를 제안한다. 그리고, 제안한 구조는 IDCT (Inverse DCT)와 IDST (Inverse DST)에도 적용이 가능함을 보인다.

1. DCT

DCT $Y(k)$ 는 주어진 데이터열 (sequence) $x(n)$, $0 \leq n \leq N-1$,에 대하여 다음과 같이 주어진다.

$$\begin{aligned} Y(k) &= \frac{2}{N} e(k) \sum_{n=0}^{N-1} x(n) \cos \left[\frac{\pi(2n+1)k}{2N} \right] \\ &= \frac{2}{N} e(k) X(k), \quad 0 \leq k \leq N-1. \end{aligned} \quad (1)$$

여기서

$$e(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k = 0 \\ 1, & \text{otherwise} \end{cases}$$

이다. 본 논문에서는 $\frac{2}{N}e(k)$ 가 단순한 상수부분 (scale factor)이므로 식 (2)와 같이 정의되는 정규화 (normalized)된 DCT $X(k)$ 를 사용하기로 한다.

$$X(k) = \sum_{n=0}^{N-1} x(n) C_k^{2n+1} \quad (2)$$

여기서 $C_k^{2n+1} = \cos \left[\frac{\pi(2n+1)k}{2N} \right]$ 이다.

$X(k)$ 를 짝수 주파수 성분 (even-numbered frequency samples)과 홀수 주파수 성분 (odd-numbered frequency samples)으로 나눈다. 짝수 주파수 성분 $X(2r)$ 은 식 (3)과 같이 주어진다.

$$\begin{aligned} X(2r) &= \sum_{n=0}^{N-1} x(n) C_{2r}^{2n+1}, \quad 0 \leq r \leq \frac{N}{2} - 1 \\ &= \sum_{n=0}^{N/2-1} x(n) C_{2r}^{2n+1} + \sum_{n=N/2}^{N-1} x(n) C_{2r}^{2n+1} \\ &= \sum_{n=0}^{N/2-1} x(n) C_{2r}^{2n+1} + \sum_{n=0}^{N/2-1} x(N-n-1) C_{2r}^{2(N-n-1)+1} \\ &= \sum_{n=0}^{N/2-1} [x(n) + x(N-n-1)] C_{2r}^{2n+1}. \end{aligned} \quad (3)$$

그리고, 홀수 주파수 성분 $X(2r+1)$ 도 짝수 주파수 성분과 비슷하게 식 (4)와 같이 나타낼 수 있다.

$$\begin{aligned} X(2r+1) &= \sum_{n=0}^{N-1} x(n) C_{2r+1}^{2n+1} \\ &= \sum_{n=0}^{N/2-1} [x(n) - x(N-n-1)] C_{2r+1}^{2n+1}. \end{aligned} \quad (4)$$

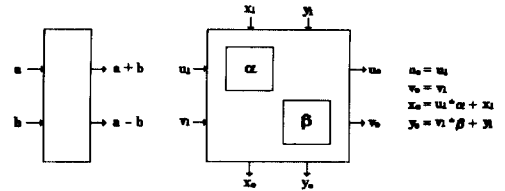
식 (3)과 (4)는 $X_e = C_e x_e$ 와 $X_o = C_o x_o$ 의 벡터, 행렬식으로 표현할 수 있다. 위의 식을 8 point DCT에 적용하여 행렬식으로 표현하면 식 (5), (6)과 같이 나타낼 수 있다.

$$X_e = \begin{bmatrix} X(0) \\ X(2) \\ X(4) \\ X(6) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ \beta & \delta & -\delta & -\beta \\ \alpha & -\alpha & -\alpha & \alpha \\ \delta & -\beta & \beta & -\delta \end{bmatrix} \begin{bmatrix} x(0)+x(7) \\ x(1)+x(6) \\ x(2)+x(5) \\ x(3)+x(4) \end{bmatrix} = C_e x_e \quad (5)$$

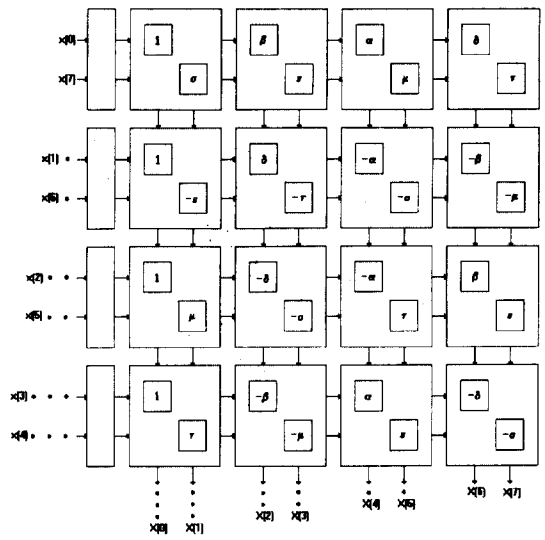
$$X_o = \begin{bmatrix} X(1) \\ X(3) \\ X(5) \\ X(7) \end{bmatrix} = \begin{bmatrix} \sigma & -\varepsilon & \mu & \tau \\ \varepsilon & -\tau & -\sigma & -\mu \\ \mu & -\sigma & \tau & \varepsilon \\ \tau & -\mu & \varepsilon & -\sigma \end{bmatrix} \begin{bmatrix} x(0)-x(7) \\ x(1)-x(6) \\ x(2)-x(5) \\ x(3)-x(4) \end{bmatrix} = C_o x_o \quad (6)$$

여기서 $\alpha = \cos \frac{\pi}{4}$, $\beta = \cos \frac{\pi}{8}$, $\delta = \sin \frac{\pi}{8}$, $\sigma = \cos \frac{\pi}{16}$, $\varepsilon = \cos \frac{3\pi}{16}$, $\mu = \sin \frac{3\pi}{16}$, 그리고 $\tau = \sin \frac{\pi}{16}$ 이다. 식 (5)와 (6)에서 보듯이 짝수 주파수 성분 $X(2r)$ 과 홀수 주파수 성분 $X(2r+1)$ 은 독립적으로 계산되므로 병렬처리가 가능하다. 그림 1은 $N = 8$ 인 경우의 제안한 시스틀릭 어레이 구조와 PE 구조를 나타낸다. 그림 1(a), (b)는 사용한 PE의 구조를 나타낸다. 그림 1(a)는 두 입력의 합과 차를 출력시키는 PE로 2 point DCT 과정과 같다. 그림 1(b)는 짝수, 홀수 주파수 성분을 병렬처리하는 PE로서 입력 u_i 와 v_i 는 x_i 와 v_i 로 전파되고 입력 x_i 와 y_i 는 x_o 와 y_o 로 출력된다.

x_o 와 y_o 는 입출력 관계식에 나타나있듯이 u_i , v_i 와 PE에 저장되어 있는 α , β 곱하여 x_i , y_i 와 더하여 출력된다. PE안에는 DCT kernel 값이 저장되어 있다. 그림 1(c)에서의 동작은 좌측에서 들어온 입력은 우측으로 전파되고 위에서 들어온 입력은 아래로 전파된다. 그러므로 DCT를 구하기 위해서는 좌측으로부터 입력을 받아 아래단으로 DCT 최종 결과가 출력된다. 좌측에서 들어온 입력은 맨 앞단의 PE, 즉 두개의 입력의 합과 차를 출력시키는 PE를 통과하여 식 (6)과



(a) (b)



(c)

그림 1. 제안한 8-point DCT 2차원 시스틀릭 어레이 구조
(a) 2-point DCT
(b) PE의 기본 기능
(c) 아키텍처

Fig. 1. Proposed two-dimensional systolic array for the 8-point DCT.
(a) 2-point DCT,
(b) Functional definition of the basic PE,
(c) Architecture.

(7)에 나타나 있듯이 입력의 합과 차를 구하게 된다. 구한 입력의 합과 차는 좌측에서 우측으로 전파되고 각 PE에서의 결과는 아래의 PE로 전파된다. 결과적으로 최종적인 결과는 맨 아래단으로부터 순서적으로 출력된다. 짝수 주파수 성분 $X(2r)$ 과 홀수 주파수 성분 $X(2r+1)$ 이 독립적으로 계산되므로 하나의 PE에서 계산할 수 있다. 그러므로 최종적인 결과는 한 클럭에 짝수, 홀수 주파수 성분의 두개의 결과값이 출력된다.

그림 1에 나타난 구조로 N -point DCT를 구할 때는 $N^2/4$ ($= N/2 \times N/2$)개의 PE가 필요하다. 즉, 8 point DCT를 구할 때는 16 ($= 4 \times 4$)개의 PE가 필요하다. PE에서 소요되는 시간은 동시에 계산되므로 한번의 곱셈과 덧셈하는 시간이다. 즉 한 클럭을 $T(= T_m + T_a)$ 로 정의한다. 여기서 T_m 과 T_a 는 실수 곱셈과 덧셈에 소요되는 시간이다. 그러므로 DCT의 첫번째 결과 $X(0)$, $X(1)$ 은 $NT/2$ 에 출력되고 마지막 결과 $X(n-2)$, $X(n-1)$ 은 $(N-1)T$ 에 출력된다.

2. DST

제안한 구조는 PE안에 저장되어 있는 kernel 값만 바꾸어서 또한 DST에 적용할 수 있다. DST $Z(k)$ 는 주어진 데이터열 $x(n+1)$, $0 < n < N-1$,에 대하여 식 (7)과 같이 주어지는데 상수 부분인 $2e(k)/N$ 을 제외하고 생각한다. 이것을 DCT에서 행한 수식 전개와 같이 전개하면 결과적으로 식 (8), (9)와 같다.

$$Z(k) = \frac{2}{N} e(k) \sum_{n=0}^{N-1} x(n+1) \sin \left[\frac{\pi k (2n+1)}{2N} \right]$$

$$= \frac{2}{N} e(k) X(k), \quad 1 \leq k \leq N. \quad (7)$$

여기서

$$e(k) = \begin{cases} \frac{1}{\sqrt{2}}, & k = N \\ 1, & \text{otherwise} \end{cases}$$

이다. DCT 경우와 같이 $X(k)$ 를 다시 나타내면

$$X(2r) = \sum_{n=0}^{N-1} x(n+1) S_{2r}^{2n+1}, \quad 1 \leq r \leq \frac{M}{2}$$

$$= \sum_{n=0}^{N/2-1} [x(n+1) - x(N-n)] S_{2r}^{2n+1} \quad (8)$$

$$X(2r+1) = \sum_{n=0}^{N-1} x(n+1) S_{2r+1}^{2n+1}$$

$$= \sum_{n=0}^{N/2-1} [x(n+1) + x(N-n)] S_{2r+1}^{2n+1} \quad (9)$$

로 쓸 수 있다. 여기서 $S_k^{2n+1} = \sin \left[\frac{\pi(2n+1)k}{2N} \right]$ 이다.

식 (8), (9)는 식 (3), (4)와 같은 모양이므로 제안한 구조를 사용할 수 있다. 그림 2는 8 point DST 2차원 시스톨릭 어레이 구조이다. 사용한 PE는 DCT 구조에서 사용한 것과 같고 PE안에 저장되어 있는 kernel 값만 DCT 경우와 다르다. DST 계산 과정에서 짝수, 홀수 주파수 성분이 동시에 출력되고 첫번째 결과 $X(1)$, $X(2)$ 는 $NT/2$ 에 출력되고 마지막 결과 $X(N-1)$, $X(N)$ 은 $(N-1)T$ 에 출력된다.

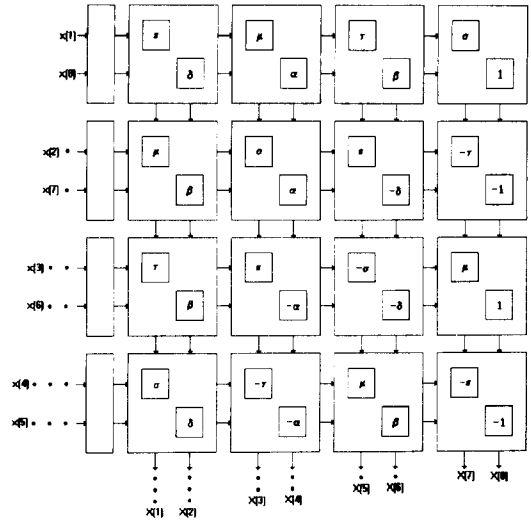


그림 2. 제안한 8-point DST 2차원 시스톨릭 어레이 구조

Fig. 2. Proposed two-dimensional systolic array for the 8-point DST.

그리고 본 구조는 IDCT와 IDST에 적용이 가능하다. IDCT는 식 (10)과 같이 주어지고 그림 1, 2의 구조와의 차이점은 맨 앞단에 있는 두 입력의 합과 차를 구하는 PE가 필요없다는 것이다. 그 외의 과정은 앞의 구조와 동일하다.

$$x(n) = \sum_{k=0}^{N-1} Y(k) \cos \left[\frac{\pi(2n+1)k}{2N} \right] \quad (10)$$

식 (11)은 IDST로서 IDCT 구조와 동일하고 PE에 있는 kernel 값만이 다르다.

$$x(n+1) = \sum_{k=1}^N e(k) Z(k) \sin \left[\frac{\pi(2n+1)k}{2N} \right] \quad (11)$$

그림 3은 8 point DCT 실험 결과이다. 전산모의 실험은 맨토 그래픽스사의 VHDL을 사용하였으며 행위 단계 (behavioral level)에서 모델링하여 시뮬레이션하였다. 실험 결과는 입력과 출력만을 표시하

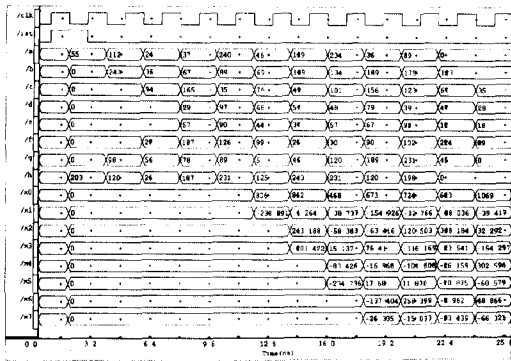
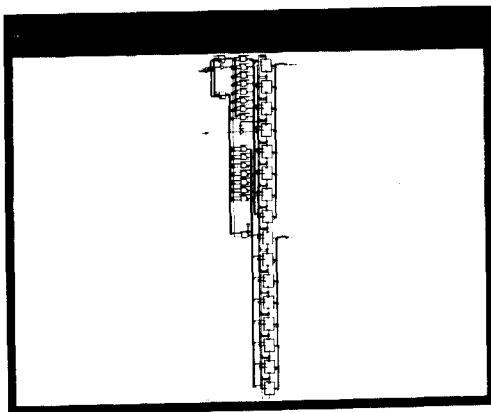


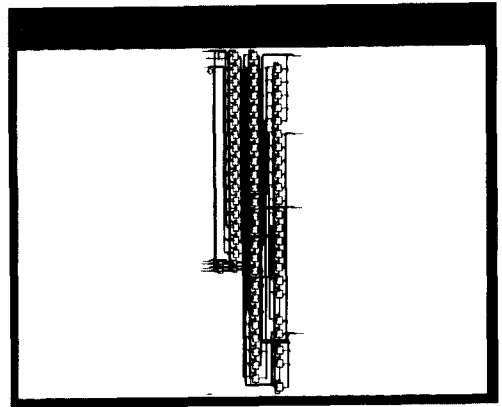
그림 3. 8-point DCT VHDL 실험 결과
Fig. 3. 8-point DCT simulation results using VHDL.

였으며 그림 3에서 a-h는 입력이고 x0-x7는 DCT 최종 결과이다. 각 PE에서 소요되는 한 클럭을 2 ns로 하고 실험하였다. 예로 55, 243, 94, 29, 57, 27, 98, 203의 DCT를 구한다고 할 때 2 ns에 첫번째 입력 a, h 즉, 55, 203이 들어가고 그 다음 클럭 즉, 4 ns에 b, g의 값인 243, 98 순서로 계속적으로 입력된다. 그리고 출력은 12 ns에 X0, X1 값인 806, -238.188이 출력되고 그 다음 클럭에 X2, X3의 순서로 계속적으로 출력된다. 그러므로 위의 데이터열이 계산되는데 그림 3에서 볼 수 있듯이 한 데이터열의 결과가 출력된 후 그 다음 클럭에 그 다음 데이터열의 결과가 출력됨을 알 수 있다. 그러므로 제안한 구조가 정상적으로 동작함을 알 수 있다.

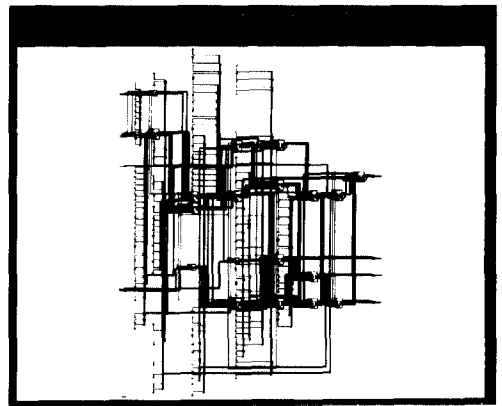
그림 4는 맨토 그래픽스사의 AutoLogic™을 이용하여 8 point DCT 구조인 그림 1을 합성한 결과이다. 그림 4(a)는 그림 1(a)에 나타난 2 point DCT 합성 결과이다. 그림 4(b)는 그림 1(b)를 합성한 것이다. 그리고 그림 4(c)는 제안한 구조를 합성한 것이다.



(a)



(b)



(c)

그림 4. 8-point VHDL합성 결과
(a) 2-point DCT 합성결과
(b) Basic PE 합성결과
(c) 8-point DCT 합성결과

Fig. 4. 8-point synthesis results using VHDL.

- (a) 2-point DCT synthesis result,
- (b) Basic PE synthesis result,
- (c) 8-point DCT synthesis result.

III. DHT의 시스틀릭 어레이 구조

본 장에서는 II장에서 제안한 구조를 이용하여 DHT를 구하는 과정과 제안한 구조를 변형하여 제안한 구조보다 약 1/2개의 PE를 사용하고 속도면에서

도 빠른 구조를 제안한다.

DHT $H(k)$ 은 주어진 데이터열 (sequence) $x(n)$, $0 < n < N-1$,에 대하여 다음과 같이 주어진다.

$$H(k) = \sum_{n=0}^{N-1} x(n) \text{cas} \left(\frac{2\pi kn}{N} \right), 0 \leq k \leq N-1. \quad (12)$$

여기서 $\text{cas } \theta \triangleq \cos \theta + \sin \theta$ 이다.

$H(k)$ 를 II장에서 전개한 것과 같이 짝수 주파수 성분과 홀수 주파수 성분으로 분리한다. 짝수 주파수 성분인 $H(2r)$ 은 식 (13)과 같이 표현된다.

$$\begin{aligned} H(2r) &= \sum_{n=0}^{N-1} x(n) \text{cas} \left(\frac{2\pi(2r)n}{N} \right) \\ &= \sum_{n=0}^{M/2-1} x(n) \text{cas} \left(\frac{2\pi(2r)n}{N} \right) \\ &\quad + \sum_{n=0}^{M/2-1} x(n+N/2) \text{cas} \left(\frac{2\pi(2r)(n+N/2)}{N} \right) \\ &= \sum_{n=0}^{M/2-1} [x(n) + x(n+N/2)] \text{cas} \left(\frac{2\pi rn}{N/2} \right), \quad 0 \leq r \leq N/2-1 \end{aligned} \quad (13)$$

유사하게 홀수 주파수 성분 $H(2r+1)$ 도 식 (14)와 같이 나타낼 수 있다.

$$\begin{aligned} H(2r+1) &= \sum_{n=0}^{N-1} x(n) \text{cas} \left(\frac{2\pi(2r+1)n}{N} \right) \\ &= \sum_{n=0}^{N/2-1} [x(n) - x(n+N/2)] \text{cas} \left(\frac{2\pi(r+1/2)n}{N/2} \right), \quad 0 \leq r \leq N/2-1. \end{aligned} \quad (14)$$

식 (13), (14)에서 볼 수 있듯이 $H(2r)$ 과 $H(2r+1)$ 은 $N/2$ point DHT와 MDHT (Modified DHT)이다. 그리고 II장에서 설명한 DCT 구조와 마찬가지로 식 (13), (14)도 서로 독립적이므로 병렬처리가 가능하다. 식 (13)과 (14)는 DCT 구조와 마찬가지로 $H_e = C_e x_e$ 와 $H_o = C_o x_o$ 로 표현할 수 있다. 식 (13)과 (14)를 이용하여 4 point DHT $H(2r)$ 과 MDHT $H(2r+1)$, $0 \leq r \leq 3$,을 행렬식으로 표현하면 다음과 같다.

$$H_e = \begin{bmatrix} H(0) \\ H(2) \\ H(4) \\ H(6) \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ 1 & -1 & 1 & -1 \\ 1 & -1 & -1 & 1 \end{bmatrix} \begin{bmatrix} x(0) + x(4) \\ x(1) + x(5) \\ x(2) + x(6) \\ x(3) + x(7) \end{bmatrix} = c_e x_e \quad (15)$$

$$H_o = \begin{bmatrix} H(1) \\ H(3) \\ H(5) \\ H(7) \end{bmatrix} = \begin{bmatrix} 1 & 2\beta & 1 & 0 \\ 1 & 0 & -1 & 2\beta \\ 1 & -2\beta & 1 & 0 \\ 1 & 0 & -1 & -2\beta \end{bmatrix} \begin{bmatrix} x(0) - x(4) \\ x(1) - x(5) \\ x(2) - x(6) \\ x(3) - x(7) \end{bmatrix} = c_o x_o \quad (16)$$

여기서 $\beta = \cos \frac{\pi}{4}$ 이다.

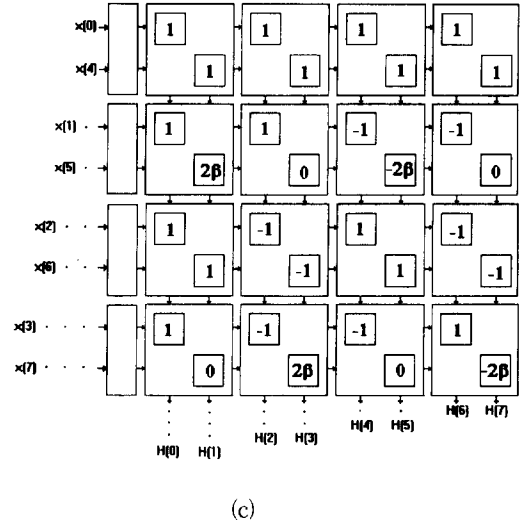
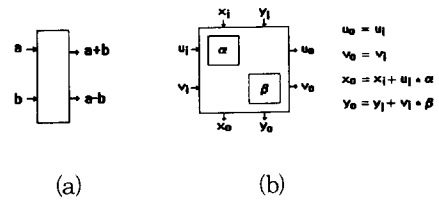


그림 5. 제안한 8-point 2차원 시스톨릭 어레이 구조 I

- (a) 2-point DHT
- (b) PE의기본 기능
- (c) 아키텍처

Fig. 5. Proposed two-dimensional systolic array I for the 8-point DHT.

- (a) 2-point DHT,
- (b) Functionbal definition of the basic PE,
- (c) Architecture.

그림 5는 $N = 8$ 인 경우의 PE 구조와 시스톨릭 어레이 구조를 나타낸다. 그림 5(a)는 PE 구조로서 DCT 구조에서 사용한 PE와 같다. 즉, 2-point DHT를 구하는 PE이다. 그리고 그림 5(b)도 DCT에서 사용한 PE와 같다. 그림 5(c)는 식 (15), (16)을 이용하여 구현한 8-point DHT의 2차원 시스톨릭 어레이 구조이다. 제안한 구조에서 볼 수 있듯이 8 point DHT 시스톨릭 어레이 구조는 II장에서 보인 DCT, DST 구조와 같다. 동작 과정도 DCT 구조에서 동작하는 것과 같다. 두 구조의 차이점은 입력이 들어가는 순서가 다르다는 것과 PE안에 저장되어 있

는 kernel 값이 다르다는 것이다. N-point DHT를 구할 때 소요되는 PE 수와 PE에서 소요되는 시간 등은 II장에서 소요되는 것과 같고 그림 5는 IDHT에도 적용이 가능하다. 그림 6은 8-point DHT 실험 결과로 제안한 구조가 DCT와 같이 정상적으로 동작함을 확인할 수 있다. 결론적으로 제안한 구조는 DCT/DST/DHT에 모두 적용이 가능하다.

그림 5의 구조를 변형하여 계산 시간과 PE 수 등에서 장점을 갖는 구조를 제안한다. 식 (13)의 짝수 주파수 성분은 식 (17)과 같이 다시 분해할 수 있다.

$$H(2r) = \sum_{n=0}^{N/2-1} [x(n) + x(n + N/2)] \text{cas}\left(\frac{2\pi r n}{N/2}\right)$$

$$H(2r + N/2) = \sum_{n=0}^{N/2-1} [x(n) + x(n + N/2)] \text{cas}\left(\frac{2\pi (r + N/4)n}{N/2}\right)$$

$$= \sum_{n=0}^{N/2-1} (-1)^n [x(n) + x(n + N/2)] \text{cas}\left(\frac{2\pi r n}{N/2}\right),$$

$$0 \leq r \leq N/4 - 1. \quad (17)$$

여기서 $H(2r)$ 과 $H(2r + N/2)$ 은 동시에 계산이 가능하다. 그리고 식 (14)의 홀수 주파수 성분도 유사하게 식 (18)과 같이 분해할 수 있고 이것도 동시에 계산이 가능하다.

$$H(2+1r) = \sum_{n=0}^{N/2-1} [x(n) - x(n + N/2)] \text{cas}\left(\frac{2\pi (r + 1/2)n}{N/2}\right)$$

$$H(2r + 1 + N/2) = \sum_{n=0}^{N/2-1} (-1)^n [x(n) - x(n + N/2)] \text{cas}\left(\frac{2\pi (r + 1/2)n}{N/2}\right).$$

$$0 \leq r \leq N/4 - 1. \quad (18)$$

그러므로 결론적으로 $H(2r)$, $H(2r + 1)$, $H(2r + N/2)$, 그리고 $H(2r + 1 + N/2)$ 을 동시에 계산할 수 있으므로 병렬처리가 가능하다.

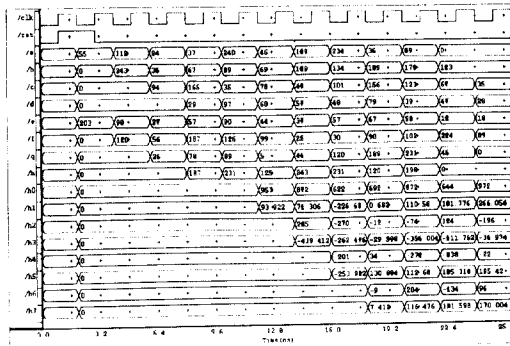
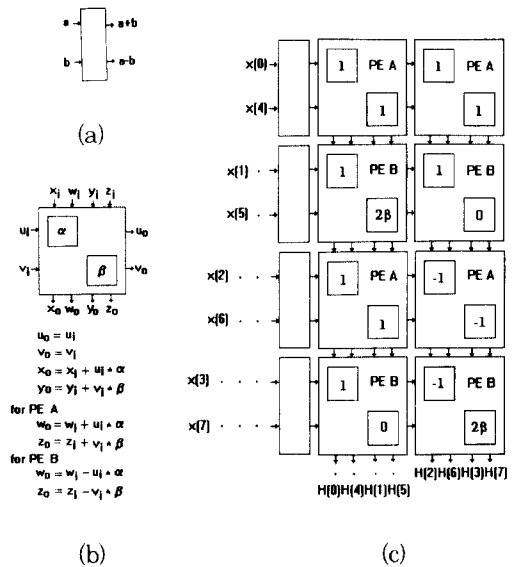


그림 6. 8-point DHT VHDL 실험 결과 (시스톨릭 어레이 I)

Fig. 6. 8-point DHT simulation results using VHDL (Systolic array I).

그림 7에 식 (17), (18)을 이용하여 8-point DHT를 구하는 시스톨릭 어레이 구조와 PE의 동작을 나타내고 있다. 그림 7(a)는 2-point DHT를 계산하는 PE이고 그림 7(b)는 앞에서 사용한 PE를 약간 변형한 구조이다. 제안한 시스톨릭 어레이 구조는 두 종류의 PE, 즉 PE A와 PE B가 존재하는데 차이점은 그림 7(b)에 나타나있듯이 w_0 와 z_0 에 있다. 즉, PE A에서는 출력 $w_0(z_0)$ 가 $u_i(v_i) \times \alpha(\beta)$ 와 $w_i(z_i)$ 의 합이 출력되고 PE B에서는 출력 $w_0(z_0)$ 가 $u_i(v_i) \times \alpha(\beta)$ 와 $w_i(z_i)$ 의 차가 출력된다. 전체적인 동작과정은 앞의 구조와 같고 PE A와 PE B는 매 행마다 번갈아 쓰인다. DHT 결과는 맨 아래의 PE에서 출력되며 한번의 클럭에 $H(0)$, $H(1)$, $H(4)$, 그리고 $H(5)$ 등으로 4개의 DHT 결과가 출력된다. 그리고 첫번째 결과 $H(0)$, $H(1)$, $H(4)$, 그리고 $H(5)$ 는 $(N/2 + 1)T$ 에 출력되고 한 데이터 열의 마지막 결과는



(b) (c)

그림 7. 제안한 8-point DHT 2차원 시스톨릭 어레이 구조 II

- (a) 2-point DHT
- (b) PE의 기본 기능
- (c) 아키텍처

Fig. 7. Proposed two-dimensional systolic array II for the 8-point DCT.

- (a) 2-point DHT,
- (b) Functional definition of the basic PE,
- (c) Architecture.

3NT/4에 출력된다. 본 구조를 VHDL을 이용하여 실험하였는데 그림 8에서 볼 수 있듯이 a-h에서 정하 여진 순서대로 입력되고 출력은 처음에 h0, h4, h1, 그리고 h5가 출력되고 그 다음 클럭에 h2, h6, h3, 그리고 h7이 출력됨을 확인할 수 있다.

본 구조로 N-point DHT를 구할 때 사용되는 PE의 수는 N/2 point DHT와 MDHT를 위한 N²/8개의 PE와 2-point DHT를 계산하는 N/2개의 PE이다.

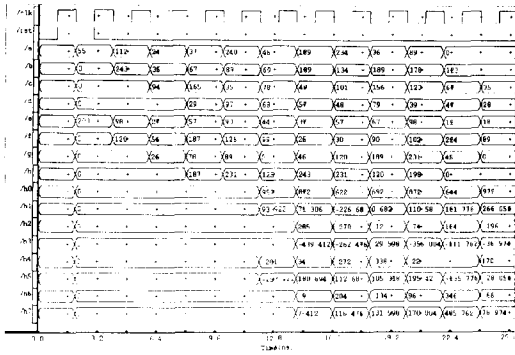


그림 8. 8-point DHT VHDL 실험 결과 (시스톨릭 어레이 II)

Fig. 8. 8-point DHT simulation results using VHDL (Systolic array II)

IV. 기존 DCT 구조와의 성능 비교

본 장에서는 제안한 DCT 구조에서 필요한 PE 수, 소요시간, 그리고 출력 생성 주기 등을 기존의 DCT 구조와 비교한다. 클럭은 II장에서 정의한대로 $T (= T_m + T_a)$ 로 한 PE에서 소요되는 실수 곱셈과 덧셈하는데 소요되는 시간으로 정의한다. 소요시간 (latency)은 한 데이터 블록의 첫번째 데이터가 입력되어 그 데이터 블록에 대한 마지막 결과가 출력되는 시간으로 정의되고 출력 생성 주기는 한 데이터 블록에 대한 첫번째 결과가 출력된 후 그 다음 데이터 블록에 대한 첫번째 결과가 출력되기까지 소요되는 시간으로 정의한다.

그림 1, 2, 그리고 5에서 나타낸 DCT/DST/DHT 계산을 위한 시스톨릭 어레이 구조에 소요되는 PE 수는 N²/4이다. 그리고 소요시간은 NT이고 출력 생성 주기는 T이다. 그림 6에 나타낸 DHT 계산을 위한 시스톨릭 어레이 구조는 N²/8개의 PE가 필요하고 소요시간은 3NT/4이다. 그리고 출력 생성 주기는 T 즉, 한 클럭이 소요된다.

표 1. N-point DCT 시스톨릭 어레이 구조의 성능 비교

Table 1. Performance comparison of various systolic array architectures for the N-point DCT.

	Cho and Lee [5]	Chang and Wu [6]	Lee [7]	Proposed
PE의 실수 곱셈기 수	4	2	8	2
필요한 PE 수	N-1	N-1	N	N ² /4
소요시간	2N(T _m +T _a)	(2N-2)(T _m +T _a)	(2(N ² -1)+N ² -2)(T _m +T _a)	(N-1)(T _m +T _a)
출력 생성 주기	N(T _m +T _a)	N(T _m +T _a)	N ² (T _m +T _a)	(T _m +T _a)

DCT/DST/DHT에 적용이 가능한 그림 1의 구조와 DCT 계산을 위한 기존의 구조와의 비교를 표 1에 나타냈다. 표에서 T_m'와 T_a'는 복소수 곱셈과 덧셈에 필요한 시간을 나타낸다. 그러므로 복소수 연산이 필요한 구조는 클럭이 실수 연산을 하는 구조보다 길게 되고 PE의 면적이 증가하게 된다. 1차원 DCT를 계산하는 기존의 구조중 [5]와 [6]의 구조는 1차원 시스톨릭 어레이를 이용한 구조이고 [7]의 구조는 2차원 시스톨릭 어레이를 이용하였다. 일반적으로 1차원 시스톨릭 어레이 구조의 면적은 O(N)에 비례하고 2차원 시스톨릭 어레이는 O(N²)에 비례한다. 반면에 속도면에서는 표 1에서 볼 수 있듯이 2차원 시스톨릭 어레이가 1차원 시스톨릭 어레이보다 2배 이상 빠르게 동작한다. 제안한 구조도 위의 일반적인 특징과 같이 속도면에서는 상당한 장점을 갖고 있지만 PE의 개수에서는 구하는 데이터열이 길어 질 경우 기존의 구조보다 많은 PE가 소요되는 단점을 갖고 있다. 면적 크기에 큰 영향을 미치는 곱셈기에서 실수 곱셈이 본 구조는 2개가 필요한 반면에 기존의 구조는 2개에서 8개까지 필요하다. 그리고, 출력 생성 주기는 제안한 구조가 (T_m + T_a)인데 반하여 기존의 구조는 N(T_m + T_a), N(T_m + T_a), 또는 N^{1.2}(T_m + T_a)으로 제안한 구조가 한 데이터열의 결과가 출력된 후 그 다음 데이터열의 결과가 출력되는데 소요되는 시간이 한 클럭인데 반하여 기존의 구조는 N클럭 또는 $\frac{N}{2}$ 클럭이 소요된 후 그 다음 데이터 열의 결과가 출력되어 출력 생성 주기가 빠름을 알 수 있다.

V. 결론

본 논문에서는 DCT/DST/DHT에 모두 적용이 가능한 2차원 시스톨릭 어레이 구조를 제안하였고 DHT에는 제안한 구조를 변형하여 적은 면적과 빠른 동작을 하는 2차원 시스톨릭 어레이 구조를 제안하였

다. 그리고 제안한 구조는 IDCT/IDST/IDHT에 적용이 가능하다.

제안한 구조는 2개의 곱셈기와 덧셈기가 있는 PE가 $N^2/4$ 개 필요하고 소요시간은 $(N-1)T$ 가 된다. 출력 생성 주기는 한 클럭이 소요된다. DHT에 적용 가능한 변형된 구조는 거의 같은 기능의 PE가 필요하다. PE에는 두개의 실수 곱셈기와 덧셈기가 있다. 이 구조에 사용된 PE 수는 $\frac{N^2}{8}$ 이고 소요시간은 $3NT/4$ 이다. 그리고 출력 생성 주기는 T 즉, 한 클럭이다.

제안한 DCT/DST/DHT에 모두 사용 가능한 2차원 시스톨릭 어레이 구조와 기존의 구조를 \mathbb{N} 장에서 비교하였듯이 제안한 구조가 소요시간, 그리고 출력 생성 주기가 빠르므로 실시간 처리면에서 기존의 구조보다 유리하다. 그리고 PE가 간단하고, PE 사이에 국부적 연결만이 필요하며 데이터 교환이 규칙적이므로 하드웨어 구현이 용이하다. 그러나 제안한 구조는 데이터 열이 길어지면 기존의 구조보다 PE의 개수가 많아지는 단점이 있으므로 1차원 어레이 구조에 대한 연구가 진행되어야 하고 또한 2차원 DCT/DST/DHT에 적용 가능한 시스톨릭 어레이 구조에 관한 연구가 진행되어야 하겠다.

參考文獻

- [1] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Commun.*, vol. COM 23, pp. 90-93, Jan. 1974.
- [2] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, and Applications*. New York, NY: Academic Press, 1990.
- [3] R. N. Bracewell, *The Hartley Transforms*. England: Oxford University Press, 1986.
- [4] H. S. Hou, "A fast recursive algorithm for computing the discrete cosine transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP 35, pp. 1455-1461, Oct. 1987.
- [5] N. I. Cho and S. U. Lee, "DCT algorithms for VLSI parallel implementations," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-38, pp. 121-127, Jan. 1990.
- [6] L.-W. Chang and M. C. Wu, "A unified systolic array for discrete cosine and sine transforms," *IEEE Trans. Signal Process.*, vol. SP-39, pp. 192-194, Jan. 1991.
- [7] M. H. Lee, "On computing 2-D systolic algorithm for discrete cosine transform," *IEEE Trans. Circuits Systems*, vol. CAS 37, pp. 1321-1323, Oct. 1990.
- [8] H. V. Sorensen, D. L. Jones, C. S. Burrus, and M. T. Heideman, "On computing the discrete Hartley transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, pp. 1231-1238, Oct. 1985.
- [9] H. S. Malvar, "Fast computation of the discrete cosine transform and the discrete Hartley transform," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, pp. 1484-1485, Oct. 1987.
- [10] J.-L. Wu and S.-C. Pei, "The vector split-radix algorithm for 2-D DHT," *IEEE Trans. Signal Process.*, vol. SP-41, pp. 960-965, Feb. 1993.
- [11] C. Chakrabarti and J. a J. a J, "Systolic architectures for the computation of the discrete Hartley and the discrete cosine transforms based on prime factor decomposition," *IEEE Trans. Comput.*, vol. C-39, pp. 1359-1368, Nov. 1990.
- [12] A. S. Dhar and S. Banerjee, "An array architecture for fast computation of discrete Hartley transform," *IEEE Trans. Circuits Syst.*, vol. CAS-38, pp. 1095-1098, Sep. 1991.
- [13] J.-H. Hsiao, L.-G. Chen, T.-D. Chiueh, and C.-T. Chen, "Novel systolic array design for the discrete Hartley transform with high throughput rate," in *Proc. Int. Conf. Circuits Syst.*, pp. 1567-1570, Minneapolis, Minnesota, May 1993.
- [14] J.-I. Guo, C.-M. Liu, and C.-W. Jen, "A CORDIC-based VLSI array for computing 2-D discrete Hartley

- transform," in *Proc. Int. Conf. Circuits Syst.*, pp. 1571-1574, Minneapolis, Minnesota, May 1993.
- [15] H. T. Kung, "Why systolic architectures?," *IEEE Computer*, vol. 15, pp. 37-46, Jan. 1982.
- [16] S. Y. Kung, *VLSI Array Processors*. Englewood Cliffs, NJ: Prentice-Hall Int. Ed., 1988.
- [17] D. L. Perry, *VHDL*. McGraw Hill, Int. Ed., 1991.
- [18] *V8 QuickSim™ II Training Workbook*, Mentor Graphics Co., 1993.

— 著 者 紹 介 —

潘 聲 範 (準會員) 第 31卷 B編 第 7號 參照
 현재 서강대학교 전자공학과 대학
 원 재학중

朴 來 弘 (正會員) 第 23卷 第 6號 參照
 현재 서강대학교 전자공학과 교수