

1차원 및 2차원 이산 웨이브렛 변환 계산을 위한 새로운 시스톨릭 어레이

(New Systolic Arrays for Computation of the 1-D and 2-D Discrete Wavelet Transform)

潘聲範*, 朴來弘**

(Sung Bum Pan and Rae-Hong Park)

요 약

본 논문에서는 1차원 및 2차원 DWT (Discrete Wavelet Transform) 계산을 위한 시스톨릭 어레이 구조를 제안한다. 1차원 DWT 계산을 위한 제안한 시스톨릭 어레이 구조는 L 개의 PE (Processing Element) 어레이로 구성되어 있다. 여기서 PE 어레이는 DWT의 한 레벨 계산에 필요한 시스톨릭 어레이를 나타낸다. 제안한 PE 어레이는 다음 계산에 필요한 부분만을 계산하고 저주파와 고주파 출력을 번갈아 계산하므로 제안한 구조는 저주파 및 고주파 출력을 한 구조에서 계산한다. 2차원 DWT 계산을 위한 제안한 구조는 2개의 1차원 DWT 계산을 위한 구조와 메모리로 구성되어 있다. 제안한 구조의 계산 시간과 하드웨어 비용은 기존의 구조와 비슷하다. 그러나, 기존의 구조는 추가적인 부분이 필요한 반면에 제안한 구조는 필요 없다. 또한 제안한 구조는 필터 계수만을 바꿔 subband 분할에도 사용할 수 있다.

Abstract

This paper proposes systolic array architectures for computation of the 1-D and 2-D discrete wavelet transform (DWT). The proposed systolic array for computation of the 1-D DWT consists of L processing element (PE) arrays, where the PE array denotes the systolic array for computation of the one level DWT. The proposed PE array computes only the product terms that are required for further computation and the outputs of low and high frequency filters are computed in alternate clock cycles. Therefore, the proposed architecture can compute the low and high frequency outputs using a single architecture. The proposed systolic array for computation of the 2-D DWT consists of two systolic array architectures for computation of the 1-D DWT and memory unit. The required time and hardware cost of the proposed systolic arrays are comparable to those of the conventional architectures. However, the conventional architectures need extra processing units whereas the proposed architectures do not. The proposed architectures can be applied to subband decomposition by simply changing the filter coefficients.

I. 서 론

DCT (discrete cosine transform), DST

(discrete sine transform), 그리고 DHT (discrete Hartley transform) 등의 이산 직교 변환은 음성 및 영상 신호 처리 분야에 효과적으로 사용되었다.^{[1]-[3]}

그러나 이러한 변환 부호화 방식은 압축률은 우수하나 사용되는 기저 (basis)들이 블록 사이에서 비연속적이므로 복원 연산에서 블록 효과가 발생한다. 이러한 블록 효과를 줄이기 위하여 DWT (discrete wavelet

* 學生會員, ** 正會員, 西江大學校 電子工學科

(Department of Electronic Engineering, Sogang University)

接受日字: 1996年10月26日, 수정완료일: 1997年8月26日

transform)가 제시되었다. DWT는 시간과 주파수에 대해 국부성을 가지고 신호를 표현할 수 있어 비정상(nonstationary) 성질을 갖는 신호를 해석함에 유리하고, 이를 이용하여 표현된 영상은 인간 시각 특성과 비슷하여 최근 영상 처리 분야에서 각광을 받기 시작하였다.^{[4]-[7]}

DWT는 유용한 변환임에도 불구하고 계산량이 많기 때문에 실시간 처리의 문제점을 가지고 있다. 이 문제를 극복하기 위하여 최근에는 특정 알고리즘을 위한 전용 컴퓨터(special purpose computer)의 개발이 촉진되어 왔는데, VLSI 기술을 이용한 시스틀릭/웨이브프론트 어레이(systolic/wavefront array)는 그 대표적인 예이다.^{[8]-[9]}

시스틀릭 어레이는 VLSI 기술을 이용하여 특정한 알고리즘의 수행 속도를 향상시키기 위해 최대한의 동시 실행(concurrency)을 이룬 전용 하드웨어 구조로 이의 특징은 모듈성(modularity), 규칙성(regularity), 국부적 연결성(local interconnection), 고도의 종속 연결성(pipelining), 잘 동기된 다중처리(multiprocessing) 등이다. 이 구조는 제어의 간단하고 단위 시간당 처리량이 많아 DSP(digital signal processing) 분야에 많이 사용된다. 그리고 웨이브프론트 어레이는 전역적인 클럭을 사용하지 않고 국부적인 클럭을 사용하는 구조로 시스틀릭 어레이의 문제인 클럭 스쿠(clock skew)를 해결할 수 있는 구조이지만 제어가 어렵다는 단점을 갖고 있다.

1차원 및 2차원 DWT 계산을 위한 여러가지 구조^{[10]-[20]}가 제안되었다. Knowles의 구조^[10]는 중간 결과 값을 저장하기 위한 큰 multiplexer가 필요하다는 단점을 갖고 있다. Parhi와 Nishitani의 구조^[12]는 계산 시간이 짧은 folded architecture를 제안하였지만, 복잡한 라우팅과 컨트롤을 위한 추가적인 하드웨어가 필요하다는 단점을 갖고 있다. 또한 이 구조는 필터 크기가 변하면 구조가 간단하게 확장되지 않는다는 단점을 갖고 있다. Vishwanath 등의 구조^[15]는 RPA(recursive pyramid algorithm)를 이용하여 구현하였지만 라우팅 네트워크가 필요하다는 단점을 갖고 있다. 임은성 등^[19]의 구조는 분석과 합성에서 레지스터를 사용하여 down sampling과 up sampling을 하였고 확장성과 레벨수의 변화에도 유용하게 적용할 수 있지만 전체적인 구조의 제어가 복잡하다는 단점을 갖고 있다.

Lewis와 Knowles의 2차원 DWT 계산을 위한 구조^[11]는 Daubechies의 4 탭 wavelet에만 적용 가능하다는 단점을 갖고 있다. 또한 Parhi와 Nishitani의 구조^[12]는 folded architecture를 제안하였지만 하드웨어 비용이 크고 필터 크기 변화에 따라 구조 확장이 어렵다는 단점을 갖고 있다.

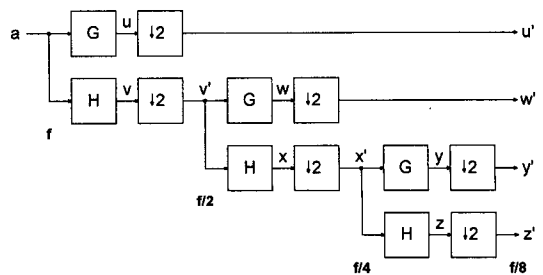
그러므로 본 논문에서는 기존 구조의 단점을 보완한 1차원 및 2차원 DWT 계산을 위한 새로운 시스틀릭 어레이 구조를 제안한다. 2장에서는 1차원 및 2차원 DWT 계산을 위한 시스틀릭 어레이 구조를 설명한다. 그리고 3장에서는 제안한 구조와 기존 구조의 성능을 평가하고 4장에서 결론을 내린다.

II. DWT 계산을 위한 새로운 시스틀릭 어레이

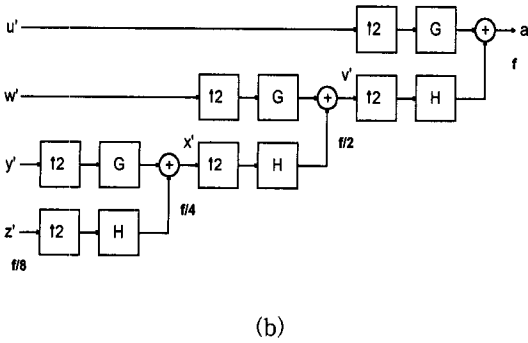
본 장에서는 1차원 및 2차원 DWT에 대해 간단히 설명하고 1차원과 2차원 DWT 계산을 위한 시스틀릭 어레이 구조에 대하여 설명한다. 설명시 편의를 위하여 레벨 수는 3이고 필터 탭 수는 4로 가정하였다.

1. 1차원 DWT

DWT는 서브밴드 코딩의 한 예로 그림 1과 같이 분석(analysis) 과정과 합성(synthesis) 과정으로 이루어져 있다. 그림 1에서 G 는 고주파 웨이브렛 필터이고 H 는 저주파 웨이브렛 필터이다. 그림 1(a)의 3레벨 분석 과정은 신호 a 가 입력되면 G 필터를 통과한 신호 u 는 2:1 down sampling되어 u' 로 출력된다. 반면에 H 필터를 통과하고 down sampling된 신호 v '는 다음 단계에서 G 와 H 필터를 다시 통과하게 된다. 그림 1(b)는 3레벨 합성 과정으로 3레벨 분석 과정의 역이다. 즉, 입력 y' 와 z' 가 1:2 up sampling되어 각각 G 와 H 필터를 통과한 후 합하여 x' 신호가 된다.



(a)



(b)

그림 1. 1차원 DWT

(a) 분석 (b) 합성

Fig. 1. 1-D DWT.

(a) analysis (b) synthesis

그림 1에 나타난 1차원 DWT의 전체적인 블록도에서 다음과 같은 주요 특징을 알 수 있다. 첫째, 각 레벨이 동일하게 구성되어 있다는 것이다. 즉, G 와 H 필터링하는 블록과 down sampling하는 블록이 하나의 단위를 구성하고 있다. 그러나 각 블록이 동일하지만 down sampling에 의해 각 레벨의 입력값의 타이밍은 틀려 하드웨어 구현시 문제가 있다. 즉, 첫 번째 레벨에서는 매 클럭마다 입력되지만 두 번째 레벨에서는 첫 번째 레벨에서의 down sampling에 의해 짝수 클럭에서만 입력되고 세 번째 레벨에서는 네 클럭마다 입력된다. 둘째, 계산량이 많은 부분은 저주파 웨이브렛 필터 H 를 통과하는 부분으로 이 블록을 어떻게 구현하느냐가 중요한 문제가 된다. 셋째, 각각의 레벨이 G 와 H 필터를 통과한 후 down sampling되므로 필터링 계산 결과값이 모두 필요하지 않으므로 하드웨어 구현시는 이 특징을 잘 이용하여야 한다. 즉, 그림 1(a)에서 u_{2n} 과 v_{2n} 만이 필요하고 u_{2n+1} 과 v_{2n+1} 은 계산할 필요가 없다는 것이다.

지금까지 설명한 DWT의 주요 특징을 이용하여 새로운 시스톨릭 어레이 구조를 구현하는 방법에 대하여 설명한다.

고주파 및 저주파 전달 함수 $G(z)$ 와 $H(z)$ 는

$$\begin{aligned} G(z) &= g_0 + g_1z^{-1} + g_2z^{-2} + g_3z^{-3} \\ H(z) &= h_0 + h_1z^{-1} + h_2z^{-2} + h_3z^{-3} \end{aligned} \quad (1)$$

와 같이 나타낼 수 있다. 여기서 $H(z)$ 와 $G(z)$ 는 각각 $\{h_n\}$ 과 $\{g_n\}$ 의 z 변환이며 필터탭수는 4로 가정하였다.

한 레벨의 저주파 DWT 계산 과정을 표시하면 다음과 같다.

$$\begin{aligned} u_0 &= a_0h_0 \\ u_1 &= a_1h_0 + a_0h_1 \\ u_2 &= a_2h_0 + a_1h_1 + a_0h_2 \\ u_3 &= a_3h_0 + a_2h_1 + a_1h_2 + a_0h_3 \\ u_4 &= a_4h_0 + a_3h_1 + a_2h_2 + a_1h_3 \\ u_5 &= a_5h_0 + a_4h_1 + a_3h_2 + a_2h_3 \\ &\vdots \end{aligned} \quad (2)$$

여기서 $\{a_n\}$ 은 입력 sequence를 나타내고 $\{u_n\}$ 은 저주파 출력 sequence를 나타낸다.

식 (2)에서 v_{2n+1} , $n \geq 0$,은 다음 레벨에서 사용되지 않으므로 계산할 필요가 없다. 그러므로 $a_{2n}h_{2m}$ 과 $a_{2n+1}h_{2m+1}$ 의 계산만이 필요하다. 또한, 한 레벨의 고주파 DWT 계산 과정을 표시하면 식 (2)에서 h_n 대신 g_n 으로 바꾸면 되고 출력 v_n 을 u_n 으로 바꾸면 된다. 그러므로, 고주파 계산 부분에서도 u_{2n+1} , $n \geq 0$,은 다음 레벨에서 사용되지 않으므로 계산할 필요가 없다. 즉 $a_{2n}g_{2m}$ 과 $a_{2n+1}g_{2m+1}$ 의 계산만이 필요하다. 위와 같은 특징을 이용하면 저주파 출력과 고주파 출력을 다른 하드웨어에서 구하지 않고 식 (3)과 같이

$$\begin{aligned} v'_0 &= u_0 = a_0h_0 \\ u'_0 &= u_0 = a_0g_0 \\ v'_1 &= u_2 = a_2h_0 + a_1h_1 + a_0h_2 \\ u'_1 &= u_2 = a_2g_0 + a_1g_1 + a_0g_2 \\ v'_2 &= u_4 = a_4h_0 + a_3h_1 + a_2h_2 + a_1h_3 \\ u'_2 &= u_4 = a_4g_0 + a_3g_1 + a_2g_2 + a_1g_3 \\ &\vdots \end{aligned} \quad (3)$$

같은 하드웨어에서 같이 구할 수 있다. 이러한 계산 과정은 모든 레벨에서 가능하다.

이러한 특징을 이용하여 한 레벨의 분석 DWT 계산을 위한 구조인 Type I과 Type II를 각각 그림 2(a), (b)에 나타내었다. 그림 2의 구조에서 $\{h_n\}$ 과 $\{g_n\}$ 은 프로세싱 엘리먼트 (PE: processing element)에 저장되어 있고 입력 a_n 이 좌측으로부터 입력되어 출력 v_{2n} 과 u_{2n} 을 번갈아가며 계산하도록 하였다. 즉, 필요하지 않은 값인 u_{2n+1} 과 v_{2n+1} 을 계산하지 않고 u_{2n} 과 v_{2n} 을 계산하도록 하였다. 그림 2(a)

의 Type I에서 A_n ($n = 1, 2, 3$) PE는 두 입력의 합을 계산하는 것이고 D PE는 입력을 한 클럭 지연시키는 것을 의미한다. 여기서 α , β , 그리고 γ 는 지연의 수를 의미하는 것으로

$$\begin{aligned} \alpha &= 2^{l-1} \\ \beta &= 2^l - 2 \\ \gamma &= \alpha + \beta \end{aligned} \quad (4)$$

로 구할 수 있다. 여기서 l 은 레벨의 수를 나타낸다. 식 (4)는 그림 2(a)에 입력되는 데이터의 타이밍이 달라 필요한 수식을 정의한 것으로 한 클럭은 한 번의 합과 곱을 하는 시간이다. l 레벨에서는 P_0 PE로 $t=0$ 에 입력 a_0 , $t=2 \times 2^l$ 에 a_2 가 입력되고 P_1 PE로 $t=2^l$ 에 a_1 이 입력되므로 α , β , γ 는 각각 식 (4)와 같이 정의되어 그림 2(a)가 정상적으로 동작한다. 그러므로 식 (4)는 모든 레벨에서 사용이 가능합니다. 예로 첫 번째, 두 번째, 그리고 세 번째 레벨의 α 는 1, 2, 그리고 4, β 는 0, 2 그리고 6이고 γ 는 1, 4 그리고 10이 된다. Type II의 구조에서는 D PE와 A PE는 필요하지 않다.

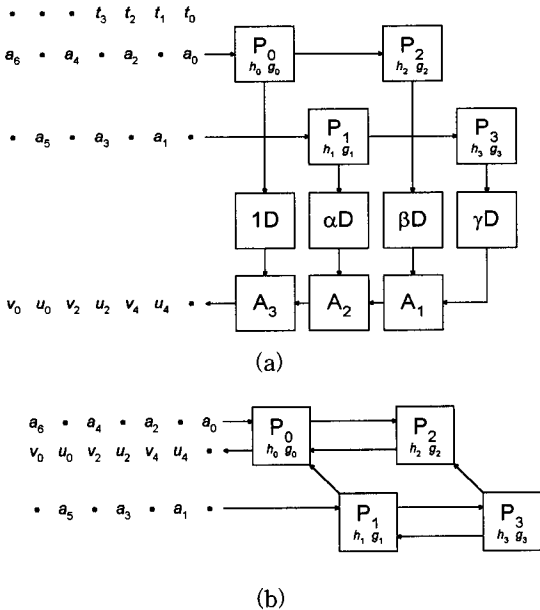


그림 2. 한 레벨 분석 웨이브렛 필터의 시스톨릭 어레이 구조

(a) Type I (b) Type II

Fig. 2. Systolic array architecture for a 1-level analysis wavelet filter.

(a) Type I (b) Type II

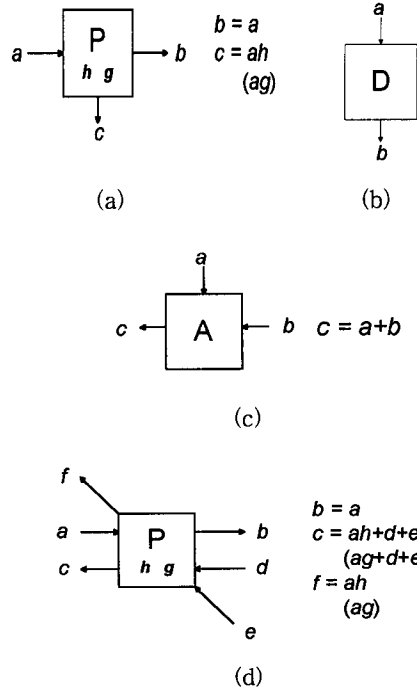


그림 3. 그림 2에서 사용된 PE의 기능도

(a) MI PE (Type I) (b) D PE (Type I)

(c) A PE (Type I) (d) MII PE (Type II)

Fig. 3. Function definition of the PE's used in Fig. 2.

(a) MI PE (Type I) (b) D PE (Type I)

(c) A PE (Type I) (d) MII PE (Type II).

그림 3에 그림 2의 Type I과 Type II에서 사용한 PE의 기본 기능을 나타내었다. 그림 3(a)의 M_I PE는 저장되어 있는 h 와 입력 a 를 곱하여 출력시키고 그 다음 클럭에는 g 와 입력 a 를 곱하여 출력시키는 PE이고 그림 3(b)의 D PE는 입력을 한 클럭 지연시키는 기능을 한다. 그림 3(c)의 A PE는 두 입력을 더하는 PE이고 그림 3(d)의 M_{II} PE는 한 클럭에 a 를 전달하고 $ah+d+e$ 및 ah 를 계산하고 그 다음 클럭에는 a 를 전달하고 $ag+d+e$ 및 ag 를 계산하는 블록으로 위의 계산을 번갈아 한다.

그림 2(a)의 구조의 이해를 위하여 표 1에 데이터 흐름을 나타내었다. 표 1에서 time은 클럭 수를 나타내고 P_n 은 $h_n(g_n)$ 을 저장하고 있는 PE의 출력을 의미한다. 그리고 A_n 은 A_n PE의 출력을 의미한다. A_3 의 결과 즉, 최종 출력은 v'_n 과 u'_n 이 번갈아 출력됨을 알 수 있다.

표 1. 한 레벨 분석 1-D 웨이브렛 변환 계산을 위한 데이터 흐름도

Table 1. Data flow for computation of a 1-level analysis of the 1-D wavelet transform.

time	P ₀	P ₁	P ₂	P ₃	A ₁	A ₂	A ₃ (출력)
1	a ₀ h ₀						
2	a ₀ g ₀	a ₁ h ₁	a ₀ h ₂				
3	a ₂ h ₀	a ₁ g ₁	a ₀ g ₂	a ₁ h ₃	a ₀ h ₂		a ₀ h ₀ = v' ₀
4	a ₂ g ₀	a ₃ h ₁	a ₂ h ₂	a ₁ g ₃	a ₀ g ₂	a ₁ h ₁ +a ₀ h ₂	a ₀ g ₀ = u' ₀
5	a ₄ h ₀	a ₃ g ₁	a ₂ g ₂	a ₃ h ₃	a ₂ h ₂ +a ₁ h ₃	a ₁ g ₁ +a ₀ g ₂	a ₂ h ₀ +a ₁ h ₁ +a ₀ h ₂ = v' ₁
6	a ₄ g ₀	a ₅ h ₁	a ₄ h ₂	a ₃ g ₃	a ₂ g ₂ +a ₁ g ₃	a ₃ h ₁ +a ₂ h ₂ +a ₁ h ₃	a ₂ g ₀ +a ₁ g ₁ +a ₀ g ₂ = u' ₁
7	a ₆ h ₀	a ₅ g ₁	a ₄ g ₂	a ₅ h ₃	a ₄ h ₂ +a ₃ h ₃	a ₃ g ₁ +a ₂ g ₂ +a ₁ g ₃	a ₄ h ₀ +a ₃ h ₁ +a ₂ h ₂ +a ₁ h ₃ = v' ₂
8	a ₆ g ₀	a ₇ h ₁	a ₆ h ₂	a ₅ g ₃	a ₄ g ₂ +a ₃ g ₃	a ₅ h ₁ +a ₄ h ₂ +a ₃ h ₃	a ₄ g ₀ +a ₃ g ₁ +a ₂ g ₂ +a ₁ g ₃ = u' ₂
9	a ₈ h ₀	a ₇ g ₁	a ₆ g ₂	a ₇ h ₃	a ₆ h ₂ +a ₅ h ₃	a ₅ g ₁ +a ₄ g ₂ +a ₃ g ₃	a ₆ h ₀ +a ₅ h ₁ +a ₄ h ₂ +a ₃ h ₃ = v' ₃
:	:	:	:	:	:	:	:

3-레벨 웨이브렛 분석 필터를 구성하기 위해서는 3개의 PE 어레이를 그림 4와 같이 연결해야 한다. 그림 4의 PE 어레이는 그림 2에서 설명한 한 레벨 DWT 계산을 위한 시스틀릭 어레이 구조로 Type I과 Type II중 어떠한 것으로 사용하여도 된다. 그림 4의 입력은 a 이고 첫 번째 레벨의 고주파 출력 u' 는 출력되고 저주파 출력 v' 는 두 번째 PE 어레이로 입력된다. 이러한 방법으로 수행하면 세 번째 PE 어레이에서 저주파와 고주파 출력 y' 와 z' 가 출력된다.

$$\begin{aligned}
 a_1 &= u'_0g_1 + v'_0h_1 = u_0g_1 + v_0h_1 \\
 a_2 &= u'_1g_0 + u'_0g_2 + v'_1h_0 + v'_0h_2 = u_2g_0 + u_0g_2 \\
 &\quad + v_2h_0 + v_0h_2 \\
 a_3 &= u'_1g_1 + u'_0g_3 + v'_1h_1 + v'_0h_3 = u_2g_1 + u_0g_3 \\
 &\quad + v_2h_1 + v_0h_3 \\
 a_4 &= u'_2g_0 + u'_1g_2 + v'_2h_0 + v'_1h_2 = u_4g_0 + u_2g_2 \\
 &\quad + v_4h_0 + v_2h_2 \\
 a_5 &= u'_2g_1 + u'_1g_3 + v'_2h_1 + v'_1h_3 = u_4g_1 + u_2g_3 \\
 &\quad + v_4h_1 + v_2h_3 \\
 &\quad \vdots
 \end{aligned} \tag{5}$$

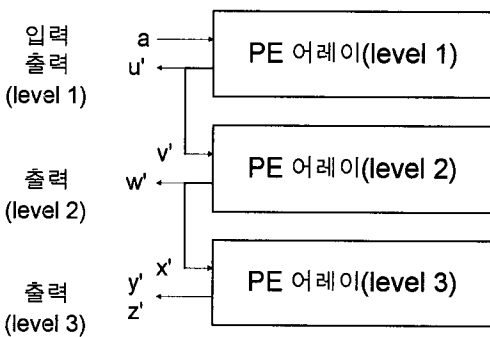


그림 4. DWT 계산을 위한 시스틀릭 어레이 (3-레벨)
Fig. 4. Systolic array architecture for 3-level analysis wavelet decomposition.

지금까지 설명한 분석 DWT 구조를 이용하여 합성 DWT도 구할 수 있음을 설명한다. 첫 번째 레벨 합성 웨이브렛 필터 계산은 다음과 같이 계산된다.

$$a_0 = u'_0g_0 + v'_0h_0 = u_0g_0 + v_0h_0$$

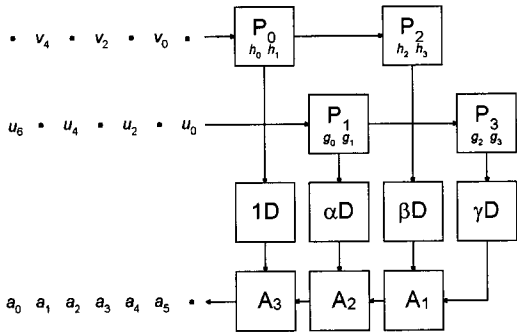
여기서도 분석 과정과 마찬가지로 $u_{2n+1}g_m$ 과 $v_{2n+1}h_m$, $n \geq 0$,은 필요하지 않으므로 $u_{2n}g_m$ 과 $v_{2n}h_m$ 만을 계산하도록 하여 그림 5의 Type I과 Type II와 같이 한 레벨의 시스틀릭 어레이 구조를 나타낼 수 있다. 저주파와 고주파 필터 계수는 PE에 저장되도록 하였고 입력이 좌측으로부터 들어오도록 하였다. 그림 5의 Type I과 II 구조는 그림 2의 Type I과 II 구조와의 차이점은 입력값과 PE에 저장되어 있는 필터 계수값만이다. 즉, 분석 DWT 구조에서는 필터 계수값이 PE에 저주파와 고주파 필터 값이 하나씩 저장되어 있는 반면에 합성 DWT에서는 하나의 PE에 두 개의 저주파 필터 값 또는 고주파 필터 값이 저장되도록 되어 있다. 그림 5(a)의 구조의 이해를 위하여 표 2에 데이터 흐름을 나타내었다. 표 2에서 P_n 은 필터 계수를 저장하고 있는 PE의 출력을 의미하고 A_n 은 A_n PE의 출력을 의미한다. A_3 의 결과 즉, 최종 출력 a_n 이 출력됨을 알 수 있다. 또한 3-레벨 웨이브렛 합성

표 2. 한 레벨 합성 1-D 웨이브렛 변환 계산을 위한 데이터 흐름도

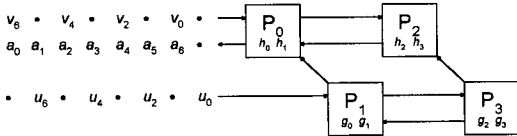
Table 2. Data flow for computation of a 1-level synthesis of the 1-D wavelet transform.

time	P ₀	P ₁	P ₂	P ₃	A ₁	A ₂	A ₃ (출력)
1		u ₀ g ₀					
2	v ₀ h ₀	u ₀ g ₁		u ₀ g ₂			
3	v ₀ h ₁	u ₂ g ₀	v ₀ h ₂	u ₀ g ₃		u ₀ g ₀	
4	v ₂ h ₀	u ₂ g ₁	v ₀ h ₃	u ₂ g ₂	u ₀ g ₂ +v ₀ h ₂	u ₀ g ₁	u ₀ g ₀ +v ₀ h ₀ = a ₀
5	v ₂ h ₁	u ₄ g ₀	v ₂ h ₂	u ₂ g ₃	u ₀ g ₃ +v ₀ h ₃	u ₂ g ₀ +u ₀ g ₂ +v ₀ h ₂	u ₀ g ₁ +v ₀ h ₁ = a ₁
6	v ₄ h ₀	u ₄ g ₁	v ₂ h ₃	u ₄ g ₂	u ₂ g ₂ +v ₂ h ₂	u ₂ g ₁ +u ₀ g ₃ +v ₀ h ₃	u ₂ g ₀ +u ₀ g ₂ +v ₂ h ₀ +v ₀ h ₂ = a ₂
7	v ₄ h ₁	u ₆ g ₀	v ₄ h ₂	u ₄ g ₃	u ₂ g ₃ +v ₂ h ₃	u ₄ g ₀ +u ₂ g ₂ +v ₂ h ₂	u ₂ g ₁ +u ₀ g ₃ +v ₂ h ₁ +v ₀ h ₃ = a ₃
8	v ₆ h ₀	u ₆ g ₁	v ₄ h ₃	u ₆ g ₂	u ₄ g ₂ +v ₄ h ₂	u ₄ g ₁ +u ₂ g ₃ +v ₂ h ₃	u ₄ g ₀ +u ₂ g ₂ +v ₄ h ₀ +v ₂ h ₂ = a ₄
9	v ₆ h ₁	u ₈ g ₀	v ₆ h ₂	u ₆ g ₃	u ₄ g ₃ +v ₄ h ₃	u ₆ g ₀ +u ₄ g ₂ +v ₄ h ₂	u ₄ g ₁ +u ₂ g ₃ +v ₄ h ₁ +v ₂ h ₃ = a ₅
:	:	:	:	:	:	:	:

과정도 3-레벨 분석 과정과 마찬가지로 세 개의 PE 어레이를 연결하면 된다.



(a)



(b)

그림 5. 한 레벨 합성 웨이브렛 계산을 위한 시스톨릭 어레이

(a) Type I (b) Type II

Fig. 5. Systolic array architecture for a 1-level synthesis wavelet filter.

(a) Type I (b) Type II.

2. 2차원 DWT

2차원 DWT 계산은 그림 6을 이용하여 구할 수

있다. 여기서 H와 G는 1-D DWT와 같다. 그림 6에서 b와 c는 열 방향의 고주파 및 저주파 웨이브렛 필터링이고 d, e, f, 그리고 g는 행 방향의 고주파 및 저주파 웨이브렛 필터링이다. 그러므로 2차원 DWT 계산을 위해서는 그림 4와 같은 구조를 이용하여 열 방향으로 계산하고 그 결과를 이용하여 행 방향으로 계산하면 된다.

2차원 DWT는

$$Y = WXW^T \tag{6}$$

로 표현할 수 있다. 여기서 W는 1-D DWT 행렬, X는 N×N 입력 데이터 행렬, 그리고 T는 행렬 전치를 나타낸다. Y는 2-D DWT 계산 결과를 의미한다. 그러므로 2-D DWT의 계산은 그림 7(a)와 같이 1-D DWT 계산을 위한 구조와 입력/출력 네트워크 즉, 메모리 모듈이 있으면 가능하다. 2-D DWT의 일반적인 계산 과정은 다음과 같다.

- (1) 1-D DWT를 행 방향으로 N번 계산한다.
- (2) 1-D DWT를 (1)의 결과를 이용하여 열 방향으로 N/2번 계산한다.
- (3) (1)과 (2)의 과정을 마지막 레벨까지 반복한다. 그러나 이러한 계산 방법은 하나의 1-D DWT 계산을 위한 하드웨어 구조가 필요하다는 장점이 있지만 계산 시간이 많이 소요되고 중간 결과값을 저장하는 메모리가 많이 필요하다는 단점을 갖고 있다.

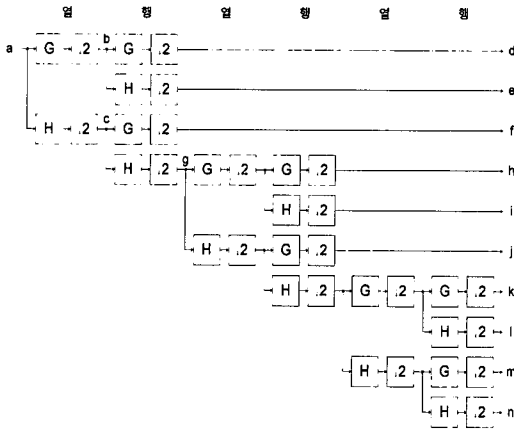
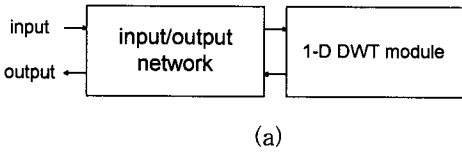
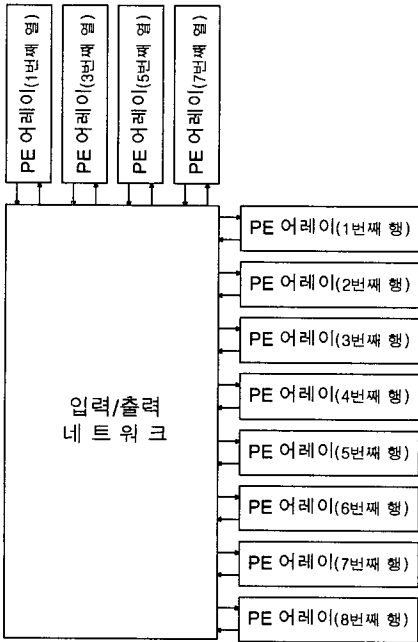


그림 6. 2차원 DWT 분석 필터
Fig. 6. 2-D DWT analysis filter.



(a)



(b)

그림 7. 2차원 DWT 계산을 위한 시스톨릭 어레이 구조
(a) 기존의 구조 (b) 제안한 구조

Fig. 7. Systolic array architecture for computation of the 2-D DWT. (a) Conventional architecture (b) Proposed architecture

그러므로 본 논문에서는 2-D DWT의 빠른 계산을 위해 $3N'$, $0 \leq N' \leq N/2$, 개의 1-D DWT 계산을 위한 하드웨어를 이용하여 2-D DWT 계산을 하는 하드웨어 구조를 그림 7(b)에 나타내었다. 그림 7(b)의 오른쪽 $2N'$, $N'=4$, PE 어레이는 행 방향으로 1-D DWT를 계산하고 위쪽의 N' PE 어레이는 열 방향의 1-D DWT를 계산한다. 열 방향으로는 N' 개의 PE 어레이만을 이용한 것은 행 방향으로의 계산 과정에서 데이터 크기가 1/2로 되기 때문이다.

그림 7(b)의 2-D DWT 계산을 위한 계산과정은 다음과 같다. (1) $2N'$ 개의 1-D DWT를 행 방향으로 동시에 $\frac{N}{2N'}$ 번 계산한다. (2) N' 개의 1-D DWT를 (1)의 결과를 이용하여 열 방향으로 $\frac{N}{2N'}$ 번 계산한다. (3) (1)과 (2)의 과정을 마지막 레벨까지 반복한다. 그러므로 본 구조는 기존의 구조 그림 7(a)보다 빠르게 동작한다.

III. 성능 분석

본 장에서는 제안한 구조와 기존의 구조를 계산 시간과 하드웨어 비용 등에서 비교한다.

표 3은 1차원 DWT 계산을 위한 기존 구조와 제안한 구조를 나타낸 것으로 N , L , 그리고 M 은 각각 sequence 크기, 레벨 수, 그리고 필터 탭 수를 나타낸다. PE 복잡도 (complexity)는 곱셈기와 덧셈기 그리고 약간의 부가적인 하드웨어로 모든 구조가 비슷하다. 제안한 구조는 LM 개의 곱셈 PE가 필요한 반면에 기존의 구조인 Lee 등의 구조^[13]와 Vishwanath 등의 구조^[15]는 각각 $2LM$, $2M$ 개의 곱셈 PE가 필요하다. 계산 시간과 필요한 PE의 개수에서는 Parhi와 Nishitani의 구조^[12]가 가장 성능이 좋지만 메모리 블럭이 필요하다는 단점을 갖고 있다. 제안한 구조의 성능이 PE의 수에서는 기존 구조와 비슷하지만 기존 구조는 모두 컨트롤 블럭, 메모리 블럭, 또는 라우팅 네트워크 등의 블럭이 추가적으로 필요하지만 제안한 구조는 추가 하드웨어가 필요 없다.

제안한 구조의 PE의 이용률은 첫번째 레벨에서는 100%이고 두번째 레벨에서는 50%이고 세번째 레벨에서는 25%이다. 또한 그림 2(a)와 (b)의 입력 핀 수는 2개이고 출력 핀 수는 1이다. 그러므로 레벨 수가 3인 그림 4의 전체 구조에서의 입력 수는 2이고 출력

수는 3이다.

표 3. 1차원 DWT 계산을 위한 시스틀릭 어레이 구조의 성능 비교

Table 3. Performance comparison of various systolic arrays for computation of the 1-D DWT.

	PE 복잡도	PE의 수	계산 시간	비고
Lang 등	×, +	M	2T	메모리 블록 필요 컨트롤 블록 필요
Lee 등	×, +	2LM	LT	전역적 데이터 교환 필요
Parhi와 Nishitani	×, +	2M	T	메모리 블록 필요 컨트롤 블록 필요
Vishwanath 등	×, +	2M	2T	라우팅 네트워크 필요
제안한 방법	×, +	LM	2T	

표 4. 2차원 DWT 계산을 위한 시스틀릭 어레이 구조의 성능 비교

Table 4. Performance comparison of various systolic arrays for computation of the 2-D DWT.

	PE 복잡도	PE의 수	계산 시간	비고
Parhi와 Nishitani	×, +	6LM		시스틀릭 어레이 아님 메모리 블록 필요 컨트롤 블록 필요
Charkrabarti와 Vishwanath	×, +	4M	2T	메모리 블록 필요
Vishwanath 등	×, +	2M	LT	라우팅 네트워크 필요 메모리 블록 필요
제안한 방법	×, +	2LM	T	메모리 블록 필요

표 4는 2-D DWT 계산을 위한 기존의 구조와 제안한 구조의 성능을 비교한 것으로 Parhi와 Nishitani의 구조^[12]는 시스틀릭 어레이 구조가 아니다. 표 4에서 제안한 구조의 성능이 PE의 수에서는 기존의 구조와 비슷하지만 추가적인 하드웨어가 적게 필요함을 알 수 있다. 그러므로 제안한 구조를 이용하여 1차원 및 2차원 DWT 계산을 위한 하드웨어를 구현할 때 기존 구조보다 유리하다.

IV. 결론

본 논문에서는 1차원 및 2차원 DWT 계산을 위한 효율적인 시스틀릭 어레이 구조를 제안하였다. 1차원

DWT 계산을 위한 제안한 시스틀릭 어레이 구조는 레벨수만큼의 PE 어레이로 구성되어 있고 제안한 PE 어레이는 $a_{2n}h_{2m}$ 과 $a_{2n+1}h_{2m+1}$ 의 계산만을 하도록 되어 있다. 또한 저주파와 고주파 출력을 번갈아 계산하도록 되어 있다. 2차원 DWT 계산을 위한 제안한 구조는 2개의 1차원 DWT 계산을 위한 구조와 메모리로 구성되어 있다.

1차원 및 2차원 DWT 계산을 위한 제안한 구조는 웨이브렛의 특성과 sequence 크기 등에 무관하게 동작하여 어떤 종류의 DWT도 계산할 수 있다. 제안한 구조의 성능은 기존의 구조와 비슷하게 동작하고 기존 구조 모두가 가지고 있는 추가적인 하드웨어 비용이 필요 없도록 설계되어 있다. 또한 본 구조는 필터 계수만을 바꿈으로써 DWT 분석과 합성을 같은 구조로 수행할 수 있고 subband 분할 등에도 바로 사용할 수 있다.

※ 본 연구는 1996년도 서강대학교 교내 연구비 지원에 의하여 이루어졌음.

참고 문헌

- [1] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete cosine transform," *IEEE Trans. Commun.*, vol. COM-23, no. 1, pp. 90-93, Jan. 1974.
- [2] K. R. Rao and P. Yip, *Discrete Cosine Transform: Algorithms, Advantages, and Applications*. New York: Academic Press, 1990.
- [3] R. N. Bracewell, *The Hartley Transforms*. England: Oxford University Press, 1986.
- [4] S. Mallat, "Multifrequency channel decompositions of images and wavelet models," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-37, no. 12, pp. 2091-2110, Dec. 1989.
- [5] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Processing Magazine*, vol. 8, no. 4, pp. 14-38, Oct. 1991.
- [6] M. Vetterli and C. Herley, "Wavelets and filter banks: Theory and design," *IEEE*

- Trans. Signal Processing*, vol. SP-40, no. 9, pp. 2207-2232, Sept. 1992.
- [7] I. Daubechies, "The wavelet transform time-frequency localization and signal analysis," *IEEE Trans. Inform. Theory*, vol. IT-36, no. 5, pp. 961-1005, Sept. 1990.
- [8] H. T. Kung, "Why systolic architectures?," *IEEE Computer*, vol. 15, no. 1, pp. 37-46, Jan. 1982.
- [9] S. Y. Kung, *VLSI Array Processors*. Englewood Cliffs, NJ: Prentice-Hall, 1988.
- [10] G. Knowles, "VLSI architecture for the discrete wavelet transform," *Electron. Lett.*, vol. 26, no. 15, pp. 1184-1185, July 1990.
- [11] A. S. Lewis and G. Knowles, "VLSI architecture for 2-D Daubechies wavelet transform without multipliers," *Electron. Lett.*, vol. 27, no. 2, pp. 171-173, Jan. 1991.
- [12] K. K. Parhi and T. Nishitani, "VLSI architectures for discrete wavelet transform," *IEEE Trans. VLSI Systems*, vol. 1, no. 2, pp. 191-202, June 1993.
- [13] H. J. Lee, J. C. Liu, A. K. Chan, and C. K. Chui, "Parallel implementation of wavelet decomposition/reconstruction algorithms," in *Proc. SPIE Wavelet Applications*, Orlando, FL, vol. 2242, pp. 248-259, Apr. 1994.
- [14] R. Lang, E. Plesner, H. Schoder, and A. Spray, "An efficient systolic architecture for one-dimensional wavelet transform," in *Proc. SPIE Wavelet Applications*, Orlando, FL, vol. 2242, pp. 925-935, Apr. 1994.
- [15] M. Vishwanath, R. M. Owens, and M. J. Irwin, "VLSI architectures for the discrete wavelet transform," *IEEE Trans. Circuits Systems*, vol. CAS-42, no. 5, pp. 305-316, May 1995.
- [16] C. Chakrabarti and M. Vishwanath, "Efficient realizations of the discrete continuous wavelet transforms: From single chip implementations to mappings on SIMD array computers," *IEEE Trans. Signal Processing*, vol. SP-43, no. 3, pp. 759-771, Mar. 1995.
- [17] M.-H. Sheu, S.-F. Cheng, and M.-D. Shieh "A pipelined VLSI with module structure design for discrete wavelet transform," in *Proc. IEEE Int. Symp. Ciccuits Systems*, Atlanta, GA, pp. 352-355, May 1996.
- [18] C. Chakrabarti and C. Mumford "Efficient realizations of analysis and synthesis filters based on the 2-D discrete wavelet transform," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Atlanta, GA, pp. 3256-3259, May 1996.
- [19] 임은성, 이문호, 양근호 "일차원 이산 웨이브렛 변환을 위한 시스틀릭 어레이 구조" 제7회 신호처리합동학술대회, 제7권 제1호, pp. 398-401, 1994년 10월
- [20] S. B. Pan and R.-H. Park, "New systolic arrays for computation of the 1-D discrete wavelet transform," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Munich, Germany, pp. 4113-4116, Apr. 1997.

 저자 소개

潘聲範(學生會員) 第31卷 B編 第7號 參照
 현재 서강대학교 전자공학과 대학
 원 재학중.

朴來弘(正會員) 第31卷 第7號 參照
 현재 서강대학교 전자공학과 교수.