

## 이산 웨이브렛 변환을 위한 효율적인 VLSI 구조

## (An Efficient VLSI Architecture for the Discrete Wavelet Transform)

潘聲範\*, 朴來弘\*\*

(Sung Bum Pan and Rae-Hong Park)

## 요 약

본 논문은 1차원 DWT 계산을 위한 효율적인 VLSI 구조를 제안한다. 제안한 구조는  $a_n h_m$ 만을 이용하여 웨이브렛 저주파 및 고주파 성분을 계산한다. 반면에 기존의 구조는  $a_n h_m$ 과  $a_n g_m$ 을 이용하여 계산한다.  $a_n$ ,  $h_m$ , 그리고  $g_m$ 은 각각 입력 시퀀스, 웨이브렛 저주파 및 고주파 계수를 나타낸다. 제안한 구조는 Daubechies 웨이브렛 필터 사이의 계수 관계식을 이용하여 좀 더 효율적으로 Daubechies 웨이브렛을 구할 수 있다. 제안한 구조와 기존 구조의 성능을 비교하여 제안한 구조는 부가적인 블록이 필요없는 반면에 기존의 구조는 부가 블록이 필요함을 제시하였다. 또한 VHDL를 이용하여 모델링하고 시뮬레이션하여 제안한 구조가 정상적으로 동작함을 확인하였다.

## Abstract

This paper proposes efficient VLSI architecture for computation of the 1-D discrete wavelet transform (DWT). The proposed VLSI architecture computes the wavelet lowpass and highpass output sequences using the product term  $a_n h_m$ ,  $n, m \geq 0$ , where  $a_n$  and  $h_m$  denote the input sequence and the wavelet lowpass filter coefficient, respectively. Whereas the conventional architectures compute the lowpass and highpass output sequences using the product terms  $a_n h_m$  and  $a_n g_m$ , respectively, where  $g_m$  denotes the wavelet highpass filter coefficient. The proposed architecture is applied to computation of the Daubechies 4-tap wavelet transform using the relationships between the Daubechies wavelet filter coefficients. Performance comparison of various architectures for computation of the 1-D DWT are presented. Note that the proposed architecture does not require extra processing units whereas the conventional architectures need them. Also it is modeled in very high speed integrated circuit hardware description language (VHDL) and simulated to show its functional validity.

## I. 서 론

DCT (Discrete Cosine Transform)는 정지 영상 표준 JPEG (Joint Photographic Experts Group) 및 동영상 표준인 MPEG (Moving Picture Experts Group)의 핵심 기술로 사용되고 있다. 이러한 변환 부호화 방식은 압축률은 높으나 사용되는 기저 (basis)들이 블록 사이에서 비연속적이므로 복원 연산에서 블록 효과 (blocking effect)가 발생한다. 이러한

\* 學生會員, 西江大學校 電子工學科/韓國電子通信研究院  
(Department of Electronic Engineering, Sogang University / Electronics and Telecommunications Research Institute)

\*\* 正會員, 西江大學校 電子工學科  
(Department of Electronic Engineering, Sogang University)

接受日字:1999年1月15日, 수정완료일:1999年4月30日

블록 효과를 줄이기 위하여 DWT (Discrete Wavelet Transform)가 제시되었다.<sup>[1]-[2]</sup> 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 영상 처리 분야에서 각광을 받기 시작하였다. 또한 표준화가 진행되고 있는 JPEG2000에서는 DCT를 대체하는 핵심 기술로 사용되고 있다.

DWT는 이와 같이 DCT를 대체할 수 있는 유용한 변환임에도 불구하고 계산량이 많기 때문에 실시간 처리의 문제점을 가지고 있다. 이 문제를 극복하기 위하여 여러 가지의 구조가 발표되었다.<sup>[3]-[11]</sup> Knowles의 구조는 중간 결과 값을 저장하기 위한 큰 multiplexer가 필요하다는 단점을 갖고 있다.<sup>[3]</sup> Lee 등의 구조는 MasPar 시스템을 이용하여 구현한 것으로 전역적인 데이터 경로가 필요하다는 단점을 갖고 있다.<sup>[5]</sup> Lang 등의 구조는 DWT 계산 블록과 메모리 블록, 그리고 제어 블록으로 구성되어 있다.<sup>[6]</sup> 이 구조의 DWT 계산 블록은 단순하지만 그 외 블록이 복잡하게 구성된다. Parhi와 Nishitani의 구조는 모든 레벨의 저주파 및 고주파 성분을 각각 하나의 저주파 및 고주파 성분을 계산하는 블록에서 계산하도록 하였다.<sup>[4]</sup> 하나의 데이터가 입력되어 필요한 모든 계산을 하도록 하기 위하여 life time 분석 방법을 사용하여 레지스터의 수를 최소화하였지만 복잡한 라우팅과 제어 위한 부가적인 하드웨어가 필요하다는 단점이 있다. 또한 이 구조는 필터 크기가 변하면 구조가 쉽게 확장되지 않는 단점을 갖고 있다. Vishwanath 등의 구조도 저주파 및 고주파 계산 블록 하나로 모든 레벨의 저주파 및 고주파 성분을 계산하도록 하였다.<sup>[7]</sup> 또한 RPA (Recursive Pyramid Algorithm)를 이용하여 구현하였다. 그러나 DWT를 계산하는 블록은 간단하게 구성되지만 DWT 계산 블록에 데이터를 입력시키고 중간 결과를 저장하는 블록인 라우팅 네트워크가 필요하다는 단점을 갖고 있다.

본 논문은 DWT의 실시간 처리를 위해 Kung이 제안한 시스톨릭 어레이 구조<sup>[12]</sup>를 이용하여 VLSI 구조를 제안하였다. 또한 제안한 구조는 웨이브렛 packet, subband 분해 (decomposition) 등의 여러 종류 필터 뱅크 (filter bank) 구조에 적용할 수 있다.

본 논문의 구성은 다음과 같다. 먼저 II장에서

DWT 계산을 위한 시스톨릭 어레이 구조에 관해 연구하기 위하여 DWT에 대해 알아보았고 DWT의 저주파와 고주파 필터 사이의 관계식을 이용하여 하드웨어 비용에서 효율적인 구조를 제안하였다. 그리고 III장에서 제안한 구조와 기존 구조의 성능을 비교하였다. 또한 IV장은 DWT 계산을 위해 제안한 구조의 정상적인 동작을 확인하기 위하여 VHDL (Very high speed integrated circuit Hardware Description Language)<sup>[13]</sup>를 이용하여 전산 모의 실험하였다. 마지막으로 V장에서 본 논문의 결론과 추후 과제를 제시하였다.

## II. DWT의 VLSI 구조

DWT는 서브밴드 코딩의 한 예로 그림 1과 같이 분석 (analysis) 과정과 합성 (synthesis) 과정으로 이루어져 있다. 그림 1에서  $G$ 는 고주파 웨이브렛 필터,  $H$ 는 저주파 웨이브렛 필터이고  $\downarrow 2$ 와  $\uparrow 2$ 는 각각 2:1 down sampling과 1:2 up sampling을 나타낸다.

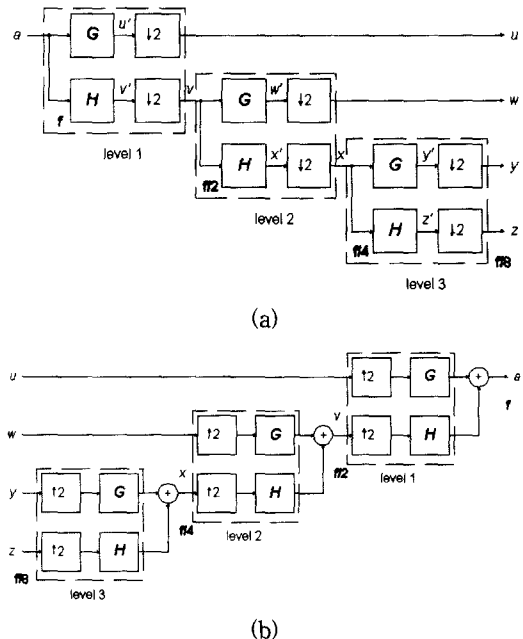


그림 1. 1-D DWT (a) 분석 (b) 합성  
Fig. 1. 1-D DWT. (a) Analysis (b) Synthesis.

그림 1(a)의 3레벨 분석 과정은 신호  $a$ 가 입력되면  $G$  필터를 통과한 신호  $u'$  즉, 고주파 출력은 2:1 down sampling되어  $u$ 로 출력된다. 반면에  $H$  필터

를 통과하고 down sampling된 저주파 신호  $u$ 는 다음 단계에서  $G$ 와  $H$  필터를 다시 통과하여 2번째 레벨 고주파 및 저주파 출력  $w, x$ 를 구하게 된다. 같은 방법으로 최종 고주파 및 저주파 출력  $y, z$ 를 구한다. 그림 1(b)는 3-레벨 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.

그림 1에 나타난 1차원 DWT의 전체 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. 둘째, 각각의 레벨이  $G$ 와  $H$  필터를 통과한 후 down sampling되므로 필터링 계산 결과 값이 모두 필요하지 않으므로 하드웨어 구현 시는 이 특징을 이용하여야 한다. 즉, 그림 1(a)에서  $u_{2n}'$ 과  $u_{2n+1}'$ 만이 필요하고  $u_{2n+1}'$ 과  $u_{2n+1}'$ 은 계산할 필요가 없다. 셋째, DWT는 저주파 필터와 고주파 필터 사이의 직교성에 의한 관계식이 있다.

이러한 DWT의 주요 특징을 이용하여 효율적인 시스틀릭 어레이 구조를 구현하는 방법에 대하여 설명한다.

고주파 및 저주파 전달 함수  $G(z)$ 와  $H(z)$ 는

$$\begin{aligned} G(z) &= g_0 + g_1 z^{-1} + \dots + g_{M-1} z^{-(M-1)} \\ H(z) &= h_0 + h_1 z^{-1} + \dots + h_{M-1} z^{-(M-1)} \end{aligned} \quad (1)$$

와 같이 나타낼 수 있다. 여기서  $G(z)$ 와  $H(z)$ 는 각각  $g_m$ 과  $h_m$ 의  $z$  변환이며  $M$ 은 필터 탭 수를 나타낸다.

한 레벨의 저주파 DWT 계산 과정을 표시하면 다음과 같다.

$$\begin{aligned} v_0' &= a_0 h_0 \\ v_1' &= a_1 h_0 + a_0 h_1 \\ v_2' &= a_2 h_0 + a_1 h_1 + a_0 h_2 \\ v_3' &= a_3 h_0 + a_2 h_1 + a_1 h_2 + a_0 h_3 \\ v_4' &= a_4 h_0 + a_3 h_1 + a_2 h_2 + a_1 h_3 \\ v_5' &= a_5 h_0 + a_4 h_1 + a_3 h_2 + a_2 h_3 \\ &\vdots \end{aligned} \quad (2)$$

여기서  $a_n$ 은 입력 sequence를 나타내고  $u_n'$ 은 저주

파 출력 sequence를 나타낸다.

식 (2)에서  $u_{2n+1}'$ ,  $n \geq 0$ , 은 앞에서 언급한 두번째 특징과 같이 다음 레벨에서 사용되지 않으므로 계산할 필요가 없다. 그러므로 식 (2)의 입력  $a_0$ 와의 곱셈은  $h_0$  및  $h_2$ 만이 필요하고 입력  $a_1$ 과의 곱셈은  $h_1$  및  $h_3$ 만이 필요하다. 그러므로 필요한 계산은  $a_{2n} h_{2m}$ 과  $a_{2n+1} h_{2m+1}$ 이고 그 외의 계산은 필요 없다. 또한, 한 레벨의 고주파 DWT 계산 과정을 표시하면

$$\begin{aligned} u_0' &= a_0 g_0 \\ u_1' &= a_1 g_0 + a_0 g_1 \\ u_2' &= a_2 g_0 + a_1 g_1 + a_0 g_2 \\ u_3' &= a_3 g_0 + a_2 g_1 + a_1 g_2 + a_0 g_3 \\ u_4' &= a_4 g_0 + a_3 g_1 + a_2 g_2 + a_1 g_3 \\ u_5' &= a_5 g_0 + a_4 g_1 + a_3 g_2 + a_2 g_3 \\ &\vdots \end{aligned} \quad (3)$$

과 같으며 식 (3)은 식 (2)에서  $h_m$  대신  $g_m$ 으로 대체한 것과 같다. 그러므로, 고주파 계산 부분에서도  $u_{2n+1}'$ ,  $n \geq 0$ , 은 계산할 필요가 없으므로  $a_{2n} g_{2m}$ 과  $a_{2n+1} g_{2m+1}$ 의 계산만이 필요하다. 위와 같은 특징을 이용하면 저주파 출력과 고주파 출력을 다른 하드웨어에서 구하지 않고 식 (4)와 같이

$$\begin{aligned} v_0 &= v_0' = a_0 h_0 \\ u_0 &= u_0' = a_0 g_0 \\ v_1 &= v_2' = a_2 h_0 + a_1 h_1 + a_0 h_2 \\ u_1 &= u_2' = a_2 g_0 + a_1 g_1 + a_0 g_2 \\ v_2 &= v_4' = a_4 h_0 + a_3 h_1 + a_2 h_2 + a_1 h_3 \\ u_2 &= u_4' = a_4 g_0 + a_3 g_1 + a_2 g_2 + a_1 g_3 \\ &\vdots \end{aligned} \quad (4)$$

같은 하드웨어에서 같이 구할 수 있다. 즉, 식 (2)에서 필요 없는 계산 클럭  $v_1', v_3', \dots$ 에 식 (3)에서 필요한 부분  $u_0', u_2', \dots$ 을 삽입하면 하나의 하드웨어에서 저주파와 고주파 출력을 구할 수 있다.

또한 세 번째 특징인 DWT의 저주파 필터와 고주파 필터 사이의 직교성 (orthogonality)에 의해

$$g_m = (-1)^m h_{M-1-m} \quad (5)$$

의 관계가 있다. 위의 특징을 이용하여 식 (4)를 다시 쓰면

$$\begin{aligned} v_0 &= v_0' = a_0 h_0 \\ u_0 &= u_0' = a_0 h_3 \\ v_1 &= v_2' = a_2 h_0 + a_1 h_1 + a_0 h_2 \\ u_1 &= u_2' = a_2 h_3 - a_1 h_2 + a_0 h_1 \\ v_2 &= v_4' = a_4 h_0 + a_3 h_1 + a_2 h_2 + a_1 h_3 \\ u_2 &= u_4' = a_4 h_3 - a_3 h_2 + a_2 h_1 - a_1 h_0 \\ &\vdots \end{aligned} \quad (6)$$

이 된다. 저주파 DWT를 계산하기 위해서는  $a_{2n}h_{2m}$ 과  $a_{2n+1}h_{2m+1}$ 이 필요하고 고주파 DWT를 계산하기 위해서는  $a_{2n}h_{2m+1}$ 과  $a_{2n+1}h_{2m}$ 이 필요하다. 그러므로 저주파 및 고주파 DWT를 계산하기 위해서는  $a_n h_m$ 이 필요하다.

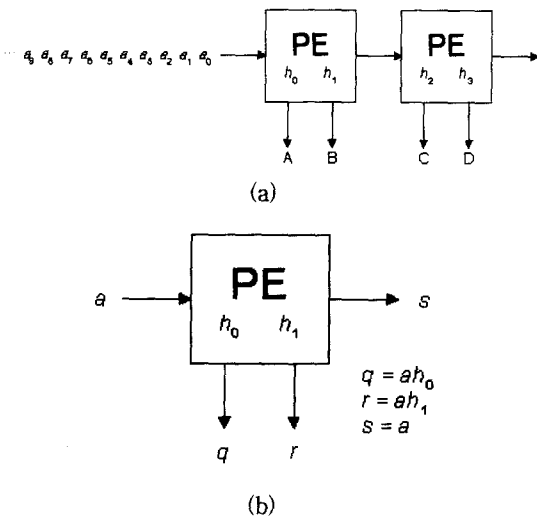


그림 2. 1-D DWT의 효율적인 시스톨릭 어레이 (분석, 1-레벨) (a) 시스톨릭 어레이 구조 (b) PE의 기능

Fig. 2. An efficient systolic array for the 1-D DWT (Analysis, 1-level). (a) Systolic array architecture (b) Function definition of the PE.

그림 2(a)는  $a_n h_m$ 만으로 저주파 및 고주파 DWT를 계산이 가능함을 나타낸 것으로 그림 2(b)에 보인

사용한 PE는 필터 계수값을 2개 저장하고 있으면서 입력되면 입력 값과 계수값을 곱하여 출력하는 구조이다.

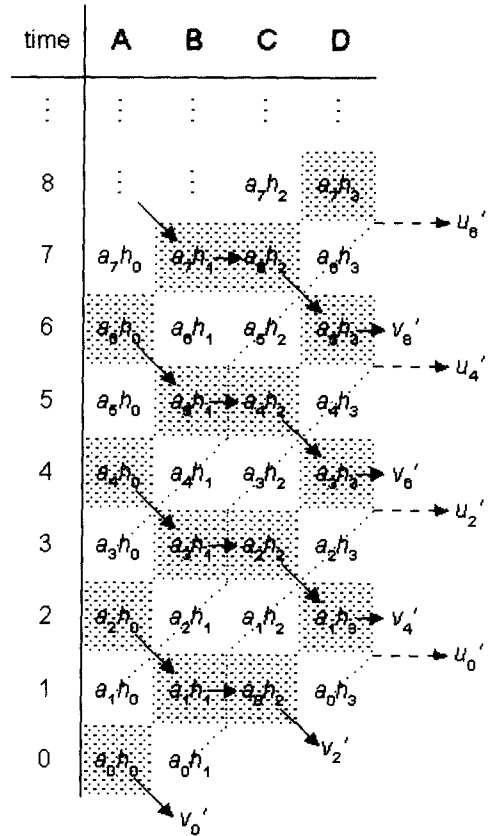


그림 3. 제안한 1-D DWT의 시스톨릭 어레이의 데이터 흐름도 (분석, 1-레벨)

Fig. 3. Data flow of the proposed systolic array for the 1-D DWT (Analysis, 1-level).

그림 3은 그림 2(a)의 데이터 흐름을 나타낸 것으로 빗금친 부분의 데이터는 저주파 DWT를 계산하는데 사용되고 그 외의 부분은 고주파 결과를 계산하는데 사용된다. 한 예로 저주파 출력  $u_4'$ 에 필요한 데이터는  $a_4 h_0, a_3 h_1, a_2 h_2$ , 그리고  $a_1 h_3$ 으로 빗금친 부분에 있다.

그림 4는 위의 특징을 이용하여 한 레벨 DWT를 구현한 것으로 입력이 PE로 되면 저주파  $v_n$ 은 왼쪽으로 출력되고 고주파 출력  $u_n$ 은 오른쪽으로 출력된다. 그림 4에서  $\bullet$ 는 한 클럭 지연을 나타내고 사용한 PE는 그림 2(b)와 A-D PE를 구성하기 위하여 덧셈기와 지연기 PE를 사용한 것이다. 본 구조에서의 PE의 사용도는 첫 번째 레벨에서는 100%이고 두 번째는

50%이다. 그리고 세 번째는 25%이다. 그러므로 본 구조는 두 번째 이하의 레벨을 각각의 구조에서 구하지 않고 하나의 구조에서 구할 수 있다. 그러나 이때 데이터 컨트롤이 복잡하다는 문제가 발생하여 본 논문에서는 각각의 구조를 적용하였다. 이 부분은 추후 데이터 컨트롤 등의 문제를 해결하면 좀더 효율적인 구조를 구할 수 있을 것이다.

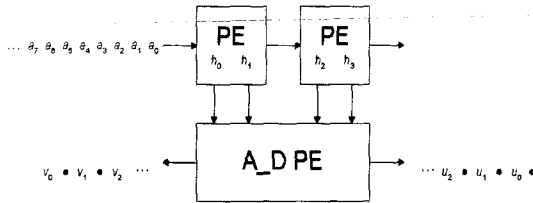


그림 4. 1-D DWT의 시스톨릭 어레이 (분석, 1-레벨)  
Fig. 4. Systolic array for the 1-D DWT (Analysis, 1-level).

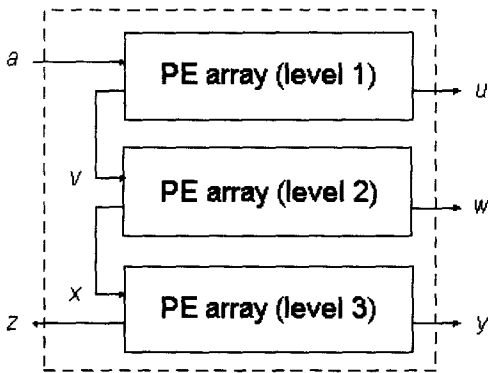


그림 5. 1-D DWT 계산을 위한 시스톨릭 어레이 (분석, 3-레벨)  
Fig. 5. Proposed VLSI architecture for the 1-D DWT (Analysis, 3-level).

3-레벨 웨이브렛 분석 필터를 구성하기 위해서는 그림 5와 같이 3개의 PE 어레이를 연결하면 된다. 즉, 그림 4가 한 레벨 DWT 계산을 위한 PE 어레이가 된다.

Daubechies의 4-tap 필터인 경우는 앞에서 설명한 구조보다 효율적으로 VLSI 구조를 구할 수 있다. Daubechies의 저주파 필터 계수는

$$\begin{aligned}
 h_0 &= \frac{1+\sqrt{3}}{4\sqrt{2}} & h_1 &= \frac{3+\sqrt{3}}{4\sqrt{2}} \\
 h_2 &= \frac{3-\sqrt{3}}{4\sqrt{2}} & h_3 &= \frac{1-\sqrt{3}}{4\sqrt{2}}
 \end{aligned}
 \tag{7}$$

이므로 정규화 상수를 무시하면  $h_0$ 와  $h_1$ 사이에는  $h_1$ 은  $h_0+2$ 로 구할 수 있으며  $h_2$ 는  $h_3+2$ 로 구할 수 있다. 그러므로 그림 2(b)의 PE를 그림 6(b)처럼 변경하여 그림 4에 적용하면 그림 6(a)처럼 된다. Daubechies의 DWT의 경우 그림 6(a)의 제안한 구조를 사용하면 곱셈기의 수를 1/2로 줄일 수 있다.

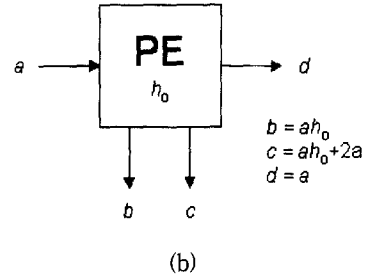
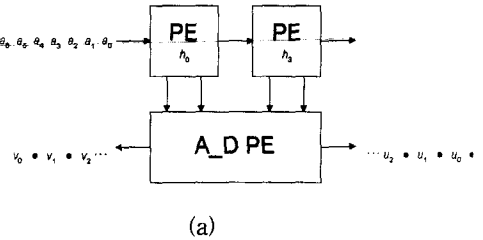


그림 6. 1-D Daubechies DWT의 시스톨릭 어레이 구조 (분석, 1-레벨) (a) 시스톨릭 어레이 구조 (b) PE의 기능

Fig. 6. Systolic array architecture for the 1-D Daubechies DWT (Analysis, 1-level). (a) Systolic array architecture (b) Function definition of the PE.

### III. 성능 평가

본 장에서는 제안한 구조와 기존의 구조를 PE 복잡도 (complexity), PE의 수, 소요시간 (latency), 그리고 출력 생성 주기 (throughput) 등을 비교한다.

표 1은 1-D 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 복잡도는 곱셈기와 덧셈기 그리고 약간의 부가적인 하드웨어로 모든 구조가 비슷하다. 사용하는 클럭은 제안한 구조와 기존 구조 모두 같다. 제안한 구조의 곱셈기 수가 2이지만 동시에 곱하므로 기존 구조와 같은 클럭을 사용한다.

필요한 PE의 수에서 제안한 구조는  $LM/2$ 개의 곱

샘 PE가 필요한 반면에 기존의 구조인 Lee 등의 구조<sup>[5]</sup>와 Vishwanath 등의 구조<sup>[7]</sup>는 각각  $2LM$ ,  $2M$ 개의 곱셈 PE가 필요하다. 계산 시간과 필요한 PE의 개수에서는 Parhi와 Nishitani의 구조<sup>[4]</sup>가 가장 성능이 좋지만 메모리 블록이 필요하다는 단점을 갖고 있다. 그리고 이 구조는 시스톨릭 어레이 구조가 아니므로 다른 구조와의 성능 비교에 약간의 문제를 갖고 있다. 또한 제안한 구조의 PE의 수가 레벨의 수에 비례하는 문제를 갖고 있는데 이 문제는 앞에서 언급한 것과 같이 2번째 이하의 레벨을 하나의 구조를 이용하여 구할 수 있으므로 해결할 수 있다.

표 1의 소요시간은 각각 한 데이터 블록의 첫 번째 데이터가 입력되어 그 데이터 블록에 대한 마지막 결과가 출력되는 시간으로 정의된다. 출력 생성 주기는 한 데이터 블록에 대한 첫 번째 결과가 출력된 후 그 다음 데이터 블록에 대한 첫 번째 결과가 출력되기까지 소요되는 시간의 비를 나타낸다. 그러므로 소요시간은 작을수록 출력 생성 주기는 클수록 좋은 성능을 나타낸다.

제안한 구조의 성능이 PE의 수에서는 기존 구조와 비슷하지만 기존 구조들은 모두 컨트롤 블록, 메모리 블록, 또는 라우팅 네트워크 등의 블록이 추가적으로 필요하지만 제안한 구조는 추가 하드웨어가 필요 없다.

또한 소요시간과 출력 생성 주기면에서 다른 구조에 비교해 좋은 성능을 나타냄을 확인할 수 있다.

#### IV. VHDL 모델링 및 시뮬레이션

본 장에서는 II장에서 제안한 1차원 DWT 계산을 위한 VLSI 구조의 정상적인 동작의 확인 및 추후 실제 칩의 구현을 위하여 대표적인 하드웨어 기술 언어인 VHDL을 이용한 전산 모의 실험에 대하여 설명한다.

본 실험에서 VHDL로 기술할 때, 각각의 PE 등 최소단위의 블록은 행위 기술로 블록을 기술하고 이들을 구조기술을 통하여 최종 시스템을 구성하였다. 각 블록의 기술은 PE의 기능을 정의한 것을 바탕으로 기술하였다. VHDL 시뮬레이션은 Mentor사의 VHDL 시뮬레이터를 사용하였다.

그림 7은 1-D DWT 계산을 위한 제안한 구조의 VHDL 시뮬레이션 결과이다. 입력 데이터는 /ain이고 최종 결과인 저주파 출력 /vout과 고주파 출력 /uout이 출력된다. 그리고 그림 2(a)의 출력 A, B, C, 그리고 D는 각각 /aout, /bout, /cout, 그리고 /dout이다. DWT의 저주파 계수는 각각 /mpe0/h0, /mpe0/h1, /mpe1/h2, 그리고 /mpe1/h3에 저장되어 있다. 예로

표 1. 1-D DWT의 시스톨릭 어레이 성능 비교

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

	PE 복잡도	PE의 수	소요시간		출력 생성 주기		비고
			1-level	3-level	1-level	3-level	
Lang 등 <sup>[6]</sup>	×, +	$M$	$2N$	$6N$	$\frac{1}{2N}$	$\frac{1}{6N}$	메모리 블록 필요 컨트롤 블록 필요
Lee 등 <sup>[5]</sup>	×, +	$2LM$	$N$	$3N$	$\frac{1}{N}$	$\frac{1}{N}$	전역적 데이터 교환 필요
Parhi와 Nishitani <sup>[4]</sup>	×, +	$2M$	$N$	$N$	$\frac{1}{N}$	$\frac{1}{N}$	시스톨릭 어레이 구조가 아님 메모리 블록 필요 컨트롤 블록 필요
Vishwanath 등 <sup>[7]</sup>	×, +	$2M$	$N$	$2N$	$\frac{1}{N}$	$\frac{1}{2N}$	라우팅 네트워크 필요
제안한 방법	$2\times$ , +	$LM/2$	$N$	$2N$	$\frac{1}{N}$	$\frac{1}{N}$	

/ain에 45ns에 "01101000"이 입력되면 다음 클럭 55ns에 /aout과 /bout에 각각 "00011001"과 "00101011"이 출력되고 그 다음 클럭 65ns에 /cout과 /dout에 각각 "00001011"과 "00000110"이 출력된다. 그림 7의 그 외의 신호인 /vdlya00, /vdlya01, ... 등은 중간 결과를 나타낸다.

/bout에 각각 "00011001"과 "11101001"이 출력되고 그 다음 클럭 65ns에 /cout과 /dout에 각각 "11010000"과 "00000000"이 출력됨을 확인할 수 있다. 그리고 최종 결과인 저주파 출력 /vout과 고주파 출력 /uout이 출력된다.

V. 결 론

본 논문에서는 1차원 DWT 계산을 위한 효율적인 시스톨릭 어레이 구조를 제안하였다. 1-D DWT의 계산을 위한 제안한 구조는 1-D DWT 특징을 찾아 저주파 및 고주파 웨이브렛을 계산시 down sampling에 의해 그 다음 레벨에서는 필요 없는 계산이 존재한다. 그러므로 저주파 계산시 필요 없는 계산을 하는 클럭에 고주파 웨이브렛을 하여 저주파 및 고주파 웨이브렛 계산을 다른 구조 또는 순차적으로 하지 않고 하나의 VLSI 구조에서 계산하도록 하였다. 그리고 또한 저주파 웨이브렛 계수와 고주파 웨이브렛 계수 사이의 관계를 이용하여 하드웨어가 단순화되도록 하였다. 그리고 제안한 구조는 Daubechies 웨이브렛 필터의 경우에는 좀 더 효율적인 구조를 구할 수 있음을 제시하였다.

1차원 DWT 계산을 위한 제안한 구조와 Daubechies 계산을 위한 제안한 구조를 VHDL 모델링하고 VHDL 시뮬레이션하여 제안한 구조가 정상적으로 동작함을 확인하였다.

추후 연구 과제로는 2-D DWT 계산을 위한 효율적인 구조에 관한 연구가 진행되어야 하고 본 논문에서 제안한 구조의 정상적인 동작의 확인을 위하여 VHDL로 전산 모의 실험하였는데 실제 chip 설계를 위한 실험이 계속 진행되어야 한다.

참 고 문 헌

[1] 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.

[2] O. Rioul and M. Vetterli, "Wavelets and signal processing," *IEEE Signal Processing Magazine*, vol. 8, no. 4, pp. 14-38, Oct. 1991.

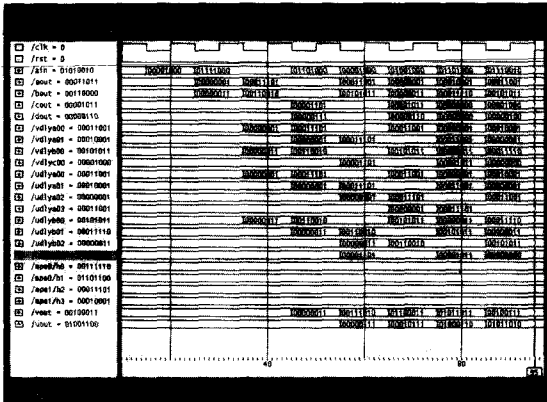


그림 7. 1-D 제안한 1-D DWT의 시스톨릭 어레이의 VHDL 실험 결과  
Fig. 7. VHDL simulation result of the proposed systolic array for the 1-D DWT.

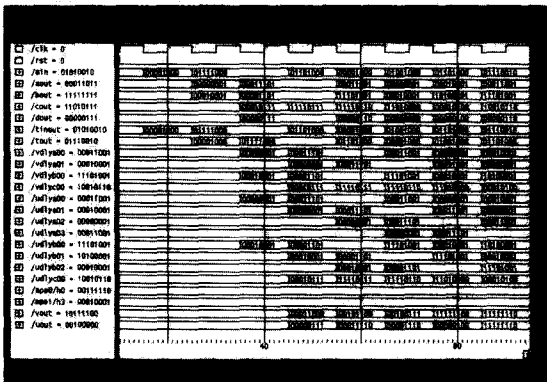


그림 8. 1-D DWT의 VHDL 실험 결과 (Daubechies)  
Fig. 8. VHDL simulation result for the 1-D DWT (Daubechies).

그림 8은 Daubechies의 1-D DWT 계산을 위한 제안한 구조의 최종 VHDL 시뮬레이션 결과이다. 입력 데이터는 /ain이고 중간 결과는 각각 /aout, /bout, /cout, 그리고 /dout 이다. 그리고 DWT의 저주파 계수는 /mpe0/h0와 /mpe1/h3에 각각 "00111110"과 "00010001"에 저장되어 있다. 예로 /ain에 45ns에 "01101000"이 입력되면 다음 클럭 55ns에 /aout과

- [ 3 ] G. Knowles, "VLSI architecture for the discrete wavelet transform," *Electron. Lett.*, vol. 26, no. 15, pp. 1184-1185, July 1990.
- [ 4 ] 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.
- [ 5 ] 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, USA, vol. 2242, pp. 248-259, Apr. 1994.
- [ 6 ] 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, USA, vol. 2242, pp. 925-935, Apr. 1994.
- [ 7 ] 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.
- [ 8 ] 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. Circuits Systems*, Atlanta, GA, USA, pp. 352-355, May 1996.
- [ 9 ] 장시중, 김대용, 김순영, 이문호, "정규직교 이산웨이브렛을 위한 효율적인 VLSI 구조," 한국통신학회논문지, 제 23권, 제 1호, pp. 239-252, 1998년 1월
- [ 10 ] T. C. Denk and K. K. Parhi, "VLSI architectures for lattice structure based orthonormal discrete wavelet transforms," *IEEE Trans. Circuits Syst. II*, vol. CAS-44, no. 2, pp. 129-132, Feb. 1997.
- [ 11 ] 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.
- [ 12 ] H. T. Kung, "Why systolic architectures?," *IEEE Computer*, vol. 15, no. 1, pp. 37-46, Jan. 1982.
- [ 13 ] D. L. Perry, *VHDL*. 2nd Ed., Singapore: McGraw-Hill, 1991.

---

 저 자 소 개
 

---

潘 肇 範(準會員) 第 31 卷 B 編 第 7 號 參照

1991년 2월 서강대학교 전자공학과 졸업 (공학사). 1995년 2월 서강대학교 전자공학과 졸업 (공학석사). 1999년 2월 서강대학교 전자공학과 졸업 (공학박사). 1999년 2월 ~ 현재 한국전자통신연구원 멀티미디어연구부 (선임연구원). 주관심분야는 영상 처리, VLSI 신호처리 등

朴 來 弘(正會員) 第 33 卷 B 編 第 9 號 參照

현재 서강대학교 전자공학과 교수