

論文2000-37SP-1-6

## 2차원 이산 웨이블릿 변환을 위한 효율적인 VLSI 구조

## (Efficient VLSI Architectures for the Two-Dimensional Discrete Wavelet Transform)

潘 聲 範 \*, 朴 來 弘 \*\*, 池 龍 \*\*

(Sung Bum Pan, Rae-Hong Park, and Yong Jee)

## 요 약

본 논문은 2차원 DWT 계산을 위한 효율적인 VLSI 구조를 제안한다. 제안한 두 개의 구조는  $M \times N$  ( $N \times M$ ) 블록 단위로 2-D DWT를 계산한다. 각각의 블록에서 2-D DWT의 계산은 행 (열) 방향으로 동시에 계산한다.  $M$ 은 필터 탭 수를 나타내고  $N$ 은 열 (행)을 나타낸다. 그리고 행과 열 방향으로 1차원 DWT를 계산할 때 저주파와 고주파 성분을 하나의 구조에서 번갈아 계산하도록 하였다. 그러므로 제안한 구조는 기존의 구조에 비해 추가적인 처리 유닛이 적게 필요하다. VHDL를 이용하여 모델링하고 시뮬레이션하여 제안한 구조가 정상적으로 동작함을 확인하였다.

## Abstract

This paper proposes efficient VLSI architectures for computation of the 2-D discrete wavelet transform (DWT). The two proposed VLSI architectures for the 2-D DWT are constructed based on block-based computation. Each  $M \times N$  ( $N \times M$ ) block DWT is performed along the row (column) direction simultaneously, where  $M$  and  $N$  denote the number of filter taps and the number of columns (rows), respectively. The proposed architectures compute the lowpass and highpass output sequences of the 1-D DWT along the row and column directions using a single architecture in alternate clock cycles. Therefore the extra processing units required for the proposed architectures are much smaller than those of the conventional architectures. They are modeled in very high speed integrated circuit hardware description language (VHDL) and simulated to show their functional validity.

## 1. 서 론

DCT (Discrete Cosine Transform)는 정지 영상 표

\* 準會員, 西江大學校 電子工學科/ 韓國電子通信硏究員  
멀티미디어硏究部

(Dept. of Electronic Engineering, Sogang University/  
Electronics and Telecommunications Research Institute)

\*\* 正會員, 西江大學校 電子工學科

(Dept. of Electronic Engineering, Sogang University)

接受日字 : 1999年3月15日, 수정완료일 : 1999年10月20日

준 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 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.

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>[3-12]</sup> 기존의 구조를 간단하게 설명하면 다음과 같다. 2차원 DWT 계산을 위한 기존의 구조는 기본적으로 1차원 DWT 구조를 이용하여 행-렬 분해방법을 이용하여 2차원 DWT를 구하는 것이다. Lewis와 Knowles의 2차원 DWT 계산을 위한 구조<sup>[3]</sup>는 Daubechies의 4 탭 DWT인 경우 곱셈기를 사용하지 않고 2차원 DWT를 계산할 수 있다. 그러나 본 구조는 Daubechies의 경우만 계산이 가능하고 다른 DWT 필터인 경우는 계산이 가능하지 않다. Parhi와 Nishitani의 구조<sup>[4]</sup>는 1차원 DWT의 folded 구조를 이용하여 2차원 DWT를 행-렬 분해 방법을 이용하여 구하는 것이지만 하드웨어 비용이 크고 필터 크기 변화에 따라 구조 확장이 어렵다는 단점을 갖고 있다. Paek과 Kim의 구조는 모든 레벨의 2차원 DWT를 두 개의 필터모듈로만 계산하도록 한 구조이다.<sup>[10]</sup> 이 구조는 영상 데이터가 입력되면 1차원 DWT를 계산하고 그 결과는 메모리 모듈을 통하여 1차원 DWT를 다시 한번 수행한다. 그 수행한 결과중 저주파-저주파 성분은 다시 구조에 입력되고 나머지 성분은 출력된다. 본 구조는 1차원 DWT를 계산하는 두 블록 사이에 메모리 블록이 필요하다. Yu와 Chen의 구조는 2차원 DWT 계산시 1차원 DWT 구조를 이용하여 행-렬 분해방법을 이용하지않고 직접 구한다는 특징을 갖고 있다.<sup>[11]</sup>

본 논문에서는 2차원 DWT 계산을 위한 효율적인 시스톨릭 어레이<sup>[13]</sup> 구조를 제안하였다. 제안한 두 개의 구조는  $M \times N(N \times M)$  블록 단위로 2-D DWT를 계산한다. 각각의 블록에서 2-D DWT의 계산은 행 (열) 방향으로 동시에 계산한다. 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 2-D DWT에 대하여 설명하고 2-D DWT 계산을 위한 효율적인 구조를 제안하였다. III장에서는 제안한 구조와 기존 구조의 성능을 비교하였고 IV장에서는 2-D DWT 계산을 위해 제안한 구조의 정상적인 동작을 확인하기 위하여 VHDL (Very high speed integrated circuit Hardware Description Language)<sup>[14]</sup>를 이용하여 전산

모의 실험하였다. 마지막으로 V장에서 본 논문의 결론과 추후 과제를 제시하였다.

## II. DWT의 VLSI 구조

2차원 DWT 계산은 그림 1을 이용하여 구할 수 있다. 여기서 H와 G는 각각 저주파 및 고주파 웨이브렛 필터를 나타내고  $\downarrow 2$ 는 2:1 down sampling을 나타낸다.

그림 1에서  $b$ 와  $c$ 는 열 방향의 고주파 및 저주파 웨이브렛 필터링이고  $d$ ,  $e$ ,  $f$  그리고  $g$ 는 행 방향의 고주파 및 저주파 웨이브렛 필터링이다. 그러므로 2차원 DWT 계산을 위해서는 1차원 DWT 계산을 위한 구조를 이용하여 열 방향으로 계산한 후 행 방향으로 계산하면 된다. 즉, 열 방향의 저주파 필터를 통과한  $c$ 는 행 방향의 고주파 및 저주파 필터를 통과하여  $f$ 와  $g$ 를 출력한다. 출력 신호  $d$ ,  $e$ ,  $f$  그리고  $g$ 는 각각 고주파-고주파, 고주파-저주파, 저주파-고주파, 그리고 저주파-저주파 성분이다. 여기서 저주파-저주파 성분인  $g$ 는 두 번째 레벨 웨이브렛 변환을 하게 된다. 결론적으로 그림 1과 같이 세 레벨 웨이브렛 변환을 통해 10개의 성분을 구할 수 있다.

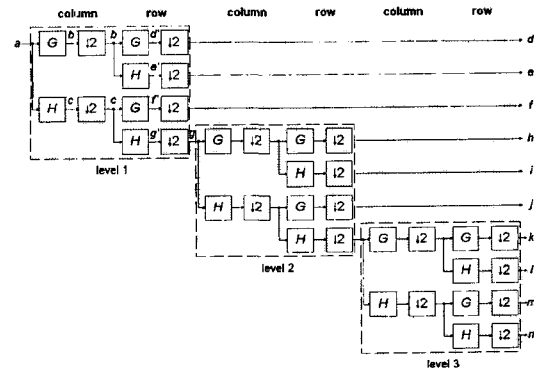


그림 1. 2-D DWT  
Fig. 1. 2-D DWT.

2차원 DWT는

$$Y = (W(WX)^T)^T \quad (1)$$

로 표현할 수 있다. 여기서  $X$ 는  $N \times N$  입력 데이터 행렬, 그리고 위첨자  $T$ 는 행렬 전치를 나타내고  $Y$ 는

2-D DWT 계산 결과를 의미한다. 또한  $W$ 는 1-D DWT 행렬로

$$W = \begin{bmatrix} h_0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ h_2 & h_1 & h_0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 0 & h_3 & h_2 & h_1 & h_0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & & & & & & & & & & & \vdots \\ 0 & \cdots & & & & & & h_3 & h_2 & h_1 & h_0 & 0 \\ g_0 & 0 & 0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ g_2 & g_1 & g_0 & 0 & 0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ 0 & g_3 & g_2 & g_1 & g_0 & 0 & \cdots & 0 & 0 & 0 & 0 & 0 \\ \vdots & & & & & & & & & & & \vdots \\ 0 & \cdots & & & & \cdots & g_3 & g_2 & g_1 & g_0 & 0 & 0 \end{bmatrix} \quad (2)$$

와 같이 주어진다.

예로  $N$ 과  $M$ 이 각각 8과 4라면  $X$ ,  $W$ , 그리고  $Y$ 는 다음과 같이 된다.

$$X = \begin{bmatrix} a_{00} & a_{01} & \cdots & a_{07} \\ \vdots & \vdots & \ddots & \vdots \\ a_{70} & a_{71} & \cdots & a_{77} \end{bmatrix}$$

$$W = \begin{bmatrix} h_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ h_2 & h_1 & h_0 & 0 & 0 & 0 & 0 & 0 \\ 0 & h_3 & h_2 & h_1 & h_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & h_3 & h_2 & h_1 & h_0 & 0 \\ g_0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ g_2 & g_1 & g_0 & 0 & 0 & 0 & 0 & 0 \\ 0 & g_3 & g_2 & g_1 & g_0 & 0 & 0 & 0 \\ 0 & 0 & 0 & g_3 & g_2 & g_1 & g_0 & 0 \end{bmatrix}$$

$$Y = \begin{bmatrix} g_{00} & g_{01} & g_{02} & g_{03} & f_{00} & f_{01} & f_{02} & f_{03} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ g_{30} & g_{31} & g_{32} & g_{33} & f_{30} & f_{31} & f_{32} & f_{33} \\ e_{00} & e_{01} & e_{02} & e_{03} & d_{00} & d_{01} & d_{02} & d_{03} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ e_{30} & e_{31} & e_{32} & e_{33} & d_{30} & d_{31} & d_{32} & d_{33} \end{bmatrix} \quad (3)$$

여기서 출력  $Y$ 는 저주파-저주파 ( $g$ ), 저주파-고주파 ( $f$ ), 고주파-저주파 ( $e$ ), 그리고 고주파-고주파 ( $d$ )로 구성되어 있다.

그러므로 2-D DWT의 계산은 그림 2와 같이 한 레벨 1-D DWT 계산을 위한 구조와 입력/출력 네트워크 즉, 메모리 모듈이 있으면 가능하다. 2-D DWT의 일반적인 계산 과정은 다음과 같다.

- (1) 한 레벨 1-D DWT를 그림 3(a)와 같이 열 방향으로  $N$ 번 계산한다.
- (2) 한 레벨 1-D DWT를 그림 3(b)와 같이 과정 (1)의 결과를 이용하여 행 방향으로 (1)의 저주파 및 고주파 성분을  $N/2$ 번 계산한다.
- (3) 과정 (1)과 (2)를 마지막 레벨까지 반복한다.

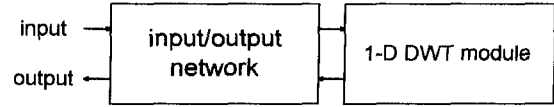


그림 2. 2-D DWT의 기존 시스톨릭 어레이  
Fig. 2. Conventional systolic array for the 2-D DWT.

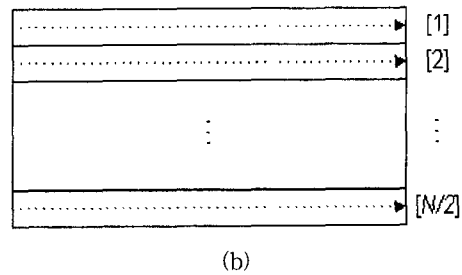
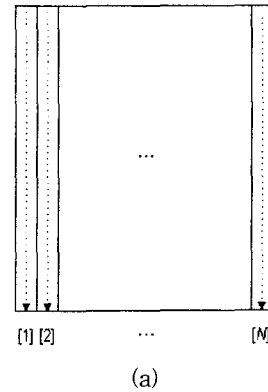


그림 3. 기존 구조의 계산 과정  
(a) 열 방향  
(b) 행 방향

Fig. 3. Computational procedure of the conventional architecture.  
(a) Column direction.  
(b) Row direction.

과정 (1)~(3)의 과정을 그림 3에 나타내었다. 그림 3(a)는 과정 (1)을 설명하는데 행 방향으로 1차원 DWT를 계산하는 것으로 그림 3(a)의 (1), (2), (3), ..., (N-1), (M)의 열을 순차적으로 계산한다. 과정 (1)이 완료되면 행 방향의 1차원 DWT를 저주파 및 고주파 성분을 계산하기 위해 순차적으로 그림 3(b)처럼 행한다. 이 때 과정 (1)에서 열 방향의 DWT를 계산할 때 down sampling에 의해 행의 크기가 1/2이 되어 그림 3(b)의 행의 크기가  $N/2$ 이 된다. 계산 과정은 그림 3(a)의 과정과 그림 3(b)를 마지막 레벨까지 계산한다.

그러나 이러한 계산 방법은 하나의 1-D DWT 계산

을 위한 하드웨어 구조가 필요하다는 장점이 있지만 계산 시간이 많이 걸리고 중간 결과를 저장하는 프레임 메모리가 필요하다는 단점을 갖고 있다. 그리고 컨트롤이 복잡하게 되는 문제점을 갖고 있다.

그러므로 일반적으로 위의 문제를 해결하기 위해서는 그림 4처럼 블록 단위로 처리할 수 있다. 그러나 이때 영상 전체에 대하여 2-D DWT 자체의 성능을 유지하기 위해서는 영상 자체의 블록보다 필터 탭 크기만큼의 블록이 그림 4의 점선처럼 영상 밖으로 필요하게 된다. 그러나 이 구조는 점선 사이의 중복되는 부분의 블록이 많이 존재하여 효율적이라 할 수 없다.

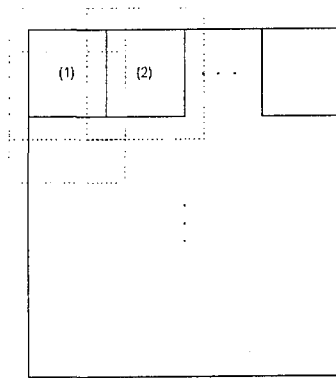


그림 4. 블록 단위 계산 과정  
Fig. 4. Block-based computational procedure.

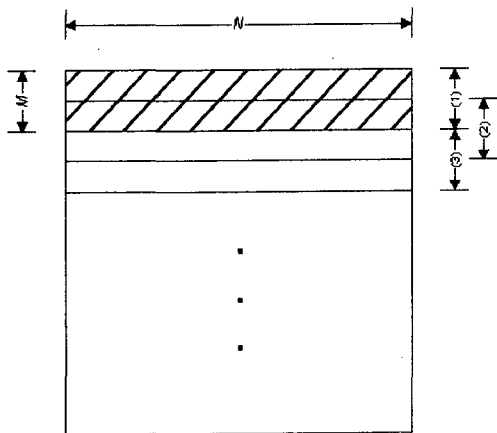


그림 5. 제안한 구조의 계산 과정  
Fig. 5. Computational procedure of the proposed architecture.

그러므로 본 논문에서는 그림 5와 같이  $M \times N$  크기의 블록 단위로 2차원 DWT를 처리하는 구조를 제안

한다. 여기서  $M$ 은 필터 탭 수를 나타내고  $N$ 은 행 방향의 화소수를 나타낸다. 본 구조도 블록의 위 아래로 중복되는 부분이 존재하지만 그림 4보다 효율적이다. 그림 5에서 각 점선 블록단위로 계산을 하게 된다. 즉 [1], [2], [3],... 의 순서로 순차적으로 계산한다. 각 블록은 2행 단위로 밑으로 진행하는데 그 이유는 down sampling에 의해 홀수 행에서는 계산의 결과가 그 다음 레벨에 필요 없기 때문이다.

제안한 구조의 2-D DWT의 계산 과정은 그림 5와 같이 필터 탭 수만큼의 블록을 기본으로 하여 2-D DWT의 한 레벨을 계산한다. 2-D DWT의 한 레벨 계산은 한 블록을 행 방향으로 진행하면서 계산하여야 하므로 기존의 구조인 1-D DWT를 2번 행과 열 방향으로 계산하는 것은 불가능하다.

그러므로 본 논문에서는 2-D DWT를 직접적으로 구현하는 두 종류의 구조 (Type I, Type II)를 제안한다. Type I은 1차원 DWT 계산을 위하여 PE array를 이용하여 열 방향의 DWT를 계산하고 곱셈기 PE와 덧셈기 PE를 이용하여 행 방향의 DWT를 계산한다. 반면에 Type II는 곱셈기 PE와 덧셈기 PE를 이용하여 열 방향으로 계산하고 두 개의 PE array를 이용하여 행 방향의 결과를 구한다. 특히 제안한 구조는 저주파-저주파, 저주파-고주파, 고주파-저주파, 그리고 고주파-고주파를 동시에 구할 수 있다.

2-D DWT 계산을 위한 제안한 구조의 계산 과정은 다음과 같다.

- (1) 식 (1)의  $(WX)^T$ 를 열 방향으로 계산한다.
- (2) 식 (1)의  $(W(WX)^T)^T$ 를 과정 (1)의 결과를 이용하여 행 방향으로 계산한다.
- (3) 과정 (1)과 (2)를 마지막 행 또는 열까지 반복한다.

여기서 과정 (3)의 마지막 행 또는 열까지 반복하는 이유는 과정 (1)에서 그림 5와 같이 행 방향의 블록으로 구성되면 과정 (3)에서는 마지막 행까지 반복하여야 하고 반대로 그림 5와 달리 열 방향으로 블록이 구성되면 과정 (3)에서는 마지막 열까지 반복하여야 하기 때문이다.

제안한 구조의 이해를 위하여 그림 1의  $g_{22} = g'_{41}$ 를 구하는 예를 가지고 먼저 설명한다.

$$\text{Type I : } \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \Rightarrow \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ c'_{41} & c'_{42} & c'_{43} & c'_{44} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \Rightarrow \begin{bmatrix} \vdots & \vdots & \vdots & \vdots \\ g'_{44} \\ \vdots & \vdots & \vdots & \vdots \end{bmatrix} \quad (4)$$

식 (4)는 제안한 구조 Type I을 설명하는 것으로  $a_{11}, a_{21}, a_{31}, a_{41}$ 를 이용하여 저주파 성분  $c'_{41}$ 를 구하고 각 열이 위의 과정을 반복 계산하면 식 (4) 가운데 행렬처럼  $c'_{42}, c'_{43}, c'_{44}$ 를 구한다. 그리고 최종적으로 구하고자 하는  $g'_{44}$ 는 가운데 행렬에서  $c'_{41}, c'_{42}, c'_{43}, c'_{44}$ 를 이용하여 구한다. 본 구조는 그림 5의 블록과 달리 블록이 행 방향으로 구성되는 구조이다.

$$\text{Type II : } \begin{bmatrix} \cdots & a_{11} & a_{12} & a_{13} & a_{14} & \cdots \\ \cdots & a_{21} & a_{22} & a_{23} & a_{24} & \cdots \\ \cdots & a_{31} & a_{32} & a_{33} & a_{34} & \cdots \\ \cdots & a_{41} & a_{42} & a_{43} & a_{44} & \cdots \end{bmatrix} \Rightarrow [\cdots c'_{41} \ c'_{42} \ c'_{43} \ c'_{44} \ \cdots] \Rightarrow [\cdots g'_{44} \ \cdots] \quad (5)$$

식 (5)는 제안한 구조 Type II를 설명하는 것으로 식 (4)의 계산과정과 같게, 먼저 열 방향으로 한 레벨 1-D DWT를 계산하여 가운데 행렬처럼 구한다. 그리고  $c'_{41}, c'_{42}, c'_{43}, c'_{44}$ 를 이용하여  $g'_{44}$ 를 구한다. 식 (5)의 계산과정은 그림 5와 같은 블록을 단위로 계산되므로  $c'_{44}$  이후에는  $c'_{45}, c'_{46}$ 의 값이 계산된다.

식 (4)에서 설명한 첫 번째 구조는 그림 6과 같이 구현할 수 있다. 그림 6(a)는 Type I 구조를 나타내고 그림 6(b)는 제안한 구조에서 사용한 PE로, 입력이 되면 저장하고 있던 저주파 및 고주파 계수값과 곱하여 출력한다. 출력이 굵은 실선은 출력이 2개임을 의미한다.

그림 7은 그림 6(a)의 PE array의 출력을 나타내는 것으로  $\alpha, \beta, \gamma, \delta$ 가 각각 저주파 및 고주파 성분을 출력함을 확인할 수 있다. 이 중간 결과값이 그림 6(a)의 곱셈기와 덧셈기 PE를 통과하여 최종 결과를 출력함을 또한 그림 6(a)에 나타내었다. 그림 6(a)에서 사용한 PE array는 [12]에서 제안한 구조이다. 즉, 1차원 DWT의 한 레벨 계산을 위한 구조이고 이 구조를 이

용하면 저주파 및 고주파 성분을 번갈아 가면서 구할 수 있다. 그리고 구한 저주파 성분을 곱셈기 PE와 덧셈기 PE를 이용하면 저주파-저주파 성분  $g$ 와 저주파-고주파 성분  $f$ 를 구할 수 있다. 그 다음 클럭에는 구한 고주파 성분을 이용하여 고주파-저주파 성분  $e$ 와 고주파-고주파 성분  $d$ 를 구할 수 있다. 그러므로 제안한 구조는 저주파-저주파, 저주파-고주파, 고주파-저주파, 그리고 고주파-고주파를 하나의 과정에서 구할 수 있다.

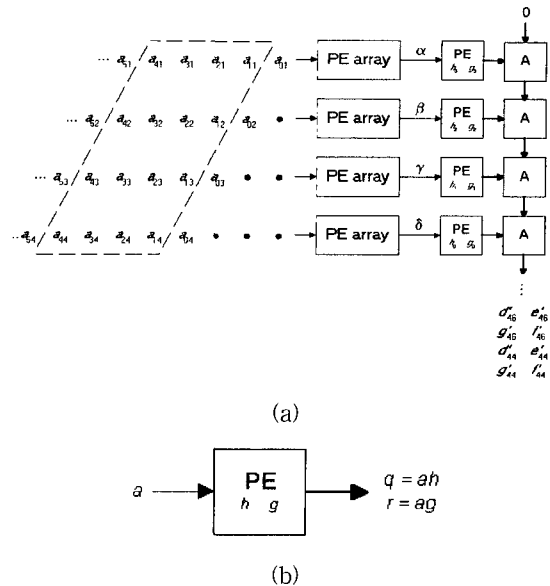


그림 6. 제안한 2-D DWT의 시스톨릭 어레이 Type I  
(a) 시스톨릭 어레이 구조  
(b) PE의 기능

Fig. 6. Proposed Type I systolic array for the 2-D DWT.

- (a) Systolic array architecture.
- (b) Function definition of the PE.

예로 저주파-저주파  $g'_{44}$ , 저주파-고주파  $f'_{44}$ , 고주파-저주파  $e'_{44}$ , 그리고 고주파-고주파  $d'_{44}$ 를 구하는 과정을 설명한다. 그림 6(a)에서 첫번째 PE array를 이용하여 저주파 성분  $c'_{41}$ 과  $b'_{41}$ 를 구한다. 그리고 그 외의 PE array를 이용하여 그림 7과 같이 열 방향의 저주파와 고주파 성분을 구한다. 즉, 그림 6(a)의 점선 블록 안에 있는 데이터로부터 저주파 성분  $c'_{41}, c'_{42}, c'_{43}$  및  $c'_{44}$ 과 고주파 성분  $b'_{41}, b'_{42}, b'_{43}$  및  $b'_{44}$ 을 구한다. 저주파-저주파  $g'_{44}$ 와 저주파-고주파  $f'_{44}$ 는

그림 7에 표시한  $c'_{41}$ ,  $c'_{42}$ ,  $c'_{43}$  및  $c'_{44}$ 를 네 개의 덧셈기 PE와 세 개의 덧셈기 PE를 이용하여 구한다. 그 다음 클럭에는  $b'_{41}$ ,  $b'_{42}$ ,  $b'_{43}$  및  $b'_{44}$ 를 이용하여 고주파-저주파  $e'_{44}$ 와 고주파-고주파  $d'_{44}$ 를 구한다.

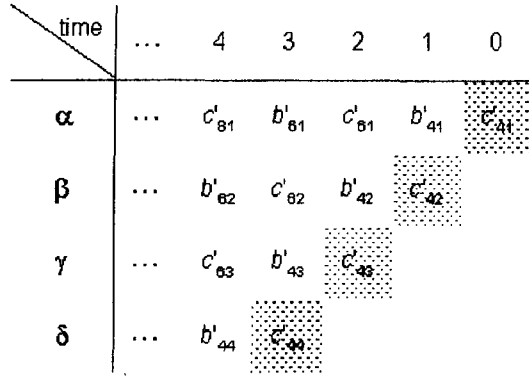


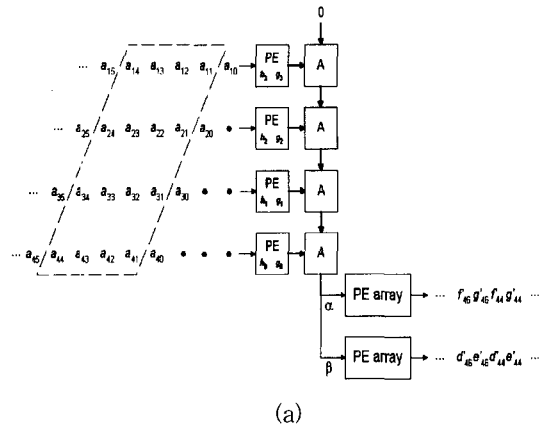
그림 7. Type I의 데이터 흐름도  
Fig. 7. Data flow of the Type I.

식 (5)에서 설명한 두 번째 구조는 그림 8과 같이 구현할 수 있다. 그림 8(a)에서 곱셈기 PE와 덧셈기 PE를 이용하여 저주파 및 고주파 성분을 동시에 계산한다. 그리고 구한 저주파 성분과 고주파 성분을 PE array를 이용하여 저주파-저주파 성분  $g$ 와 고주파-저주파 성분  $e$ 를 구할 수 있다. 그 다음 클럭에는 저주파-고주파 성분  $f$ 와 고주파-고주파 성분  $d$ 를 구할 수 있다. 그러므로 Type II도 Type I의 구조와 마찬가지로 저주파-저주파, 저주파-고주파, 고주파-저주파, 그리고 고주파-고주파를 하나의 과정에서 구할 수 있다.

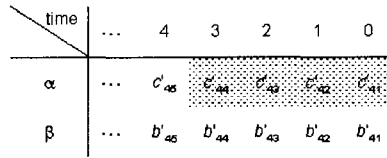
예로 저주파-저주파  $g'_{44}$ , 저주파-고주파  $f'_{44}$ , 고주파-저주파  $e'_{44}$ , 그리고 고주파-고주파  $d'_{44}$ 를 구하는 과정을 설명한다. 그림 8(a)에서 곱셈기 PE와 덧셈기 PE를 이용하여 저주파 성분  $c'_{41}$ 과  $b'_{41}$ 를 동시에 구한다. 그리고 순차적으로  $c'_{42}$ ,  $c'_{43}$  및  $c'_{44}$ 과  $b'_{42}$ ,  $b'_{43}$  및  $b'_{44}$ 를 그림 8(b)와 같이 구한다. 저주파-저주파  $g'_{44}$ 와 저주파-고주파  $f'_{44}$ 는 그림 7에 표시한  $c'_{41}$ ,  $c'_{42}$ ,  $c'_{43}$  및  $c'_{44}$ 를 첫번째 PE array를 통과시켜 구한다. 그와 동시에 두 번째 PE array에서는  $b'_{41}$ ,  $b'_{42}$ ,  $b'_{43}$  및  $b'_{44}$ 를 이용하여 고주파-저주파  $e'_{44}$ 와 고주파-고주파  $d'_{44}$ 를 구한다.

그림 6과 8에서 2차원 DWT 계산을 위한 제안한 구조는 웨이브렛 packet, subband 분해 (decomposition)

등의 여러 종류 필터 बैं크 (filter bank) 구조에 필터 계수값과 약간의 수정을 통하여 적용할 수 있다.



(a)



(b)

그림 8. 제안한 2-D DWT의 시스톨릭 어레이 Type II  
(a) 시스톨릭 어레이 구조  
(b) 데이터 흐름도

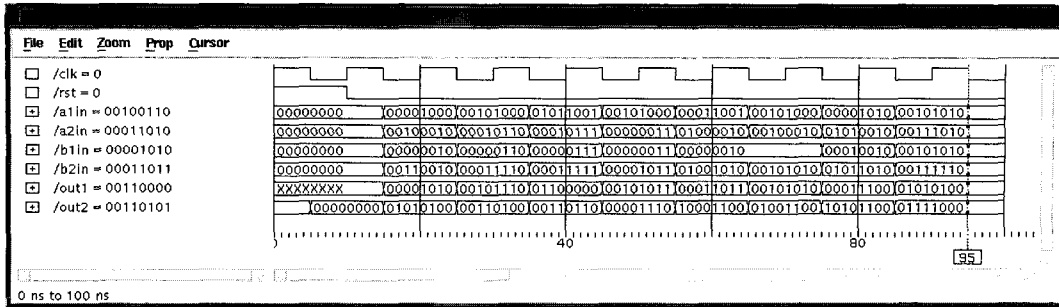
Fig. 8. Proposed Type II systolic array for the 2-D DWT.

(a) Systolic array architecture.  
(b) Data flow

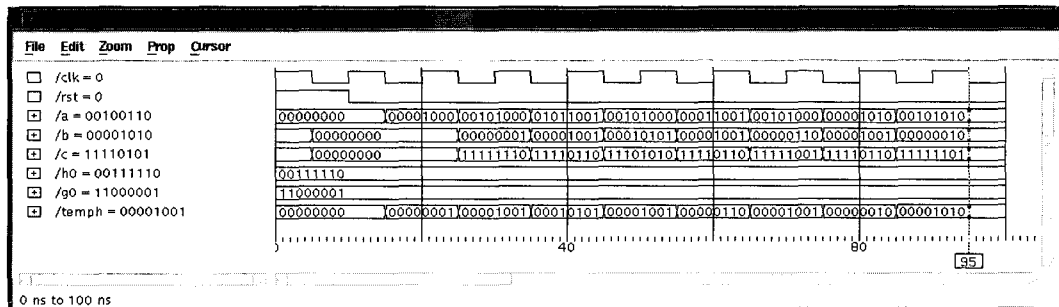
### III. 성능 평가

본 장에서는 제안한 구조와 기존의 구조를 PE 복잡도와 PE의 수 등을 비교한다. 표 1은 2-D DWT 계산을 위한 기존 구조와 제안한 구조를 나타낸 것으로  $L$ 과  $M$ 은 각각 레벨 수 및 필터 탭 수를 나타낸다. PE 복잡도는 곱셈기와 덧셈기 그리고 약간의 부가적인 하드웨어로 모든 구조가 비슷하다.

제안한 구조 Type I (Type II)의 PE의 수에서  $M^2 + M$  ( $3M$ )는 [12]의 PE array를 사용한 경우이다. 표 1에서 제안한 구조의 성능이 PE의 수에서는 기존의 구조와 비슷하지만 부가적인 하드웨어가 적게 필요함을 알 수 있다. 그러므로 제안한 구조를 이용하여 2차원 DWT 계산을 위한 하드웨어를 구현할 때 기존 구조보다 유리함을 확인할 수 있다.



(a)



(b)

그림 9. PE의 VHDL 실험 결과

(a) A PE (b) M PE

Fig. 9. VHDL simulation result of PE.

(a) A PE (b) M PE

표 1. 2차원 DWT의 시스톨릭 어레이 성능 비교

Table 1. Performance comparison of various systolic arrays for the 2-D DWT.

	PE 복잡도	PE의 수	비고
Parhi와 Nishitani	×, +	6LM	시스톨릭 어레이 구조가 아님 메모리 블록 필요 복잡한 컨트롤 블록 필요
Chakrabarti와 Vishwanath	×, +	4M	메모리 블록 필요
Vishwanath 등	×, +	2M	라우팅 네트워크 필요 메모리 블록 필요
제안한 방법 (Type I)	×, +	$M^2+M$	
제안한 방법 (Type II)	×, +	3M	

또한 제안한 구조는 기존 구조와 달리 행-렬 분해 방법을 사용하지 않아 중간 결과를 저장하는 메모리와 복잡한 제어 블록이 필요 없어 제안한 구조의 하드웨

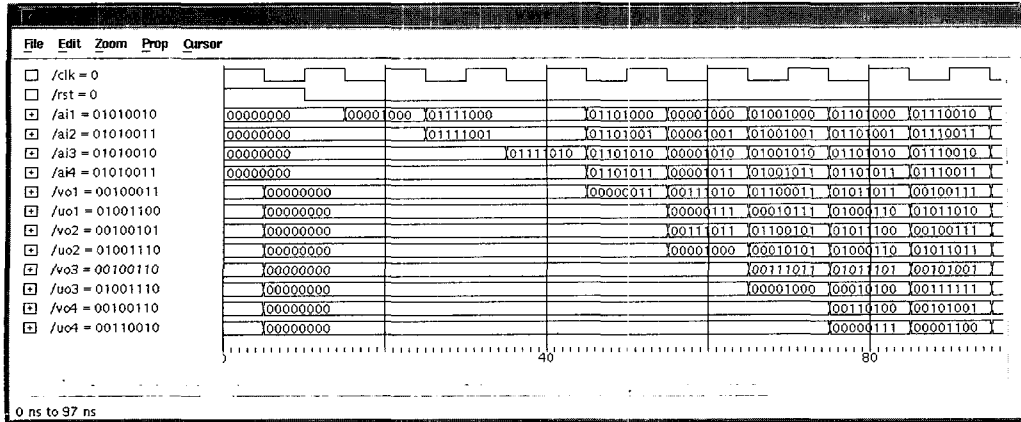
어 구현이 용이하다.

Yu와 Chen의 구조<sup>[11]</sup>도 제안한 구조와 같이 행-렬 분해 방법을 사용하지 않았다. Yu와 Chen의 구조는 곱셈기가  $M^2/2$ 이 필요한 반면에 제안한 구조는 레벨수 만큼 Type I과 Type II의 구조를 사용하면  $M^2+M$  또는  $3M$ 이 필요하여 M이 크면 제안한 구조가 좋은 성능을 나타냄을 확인할 수 있다.

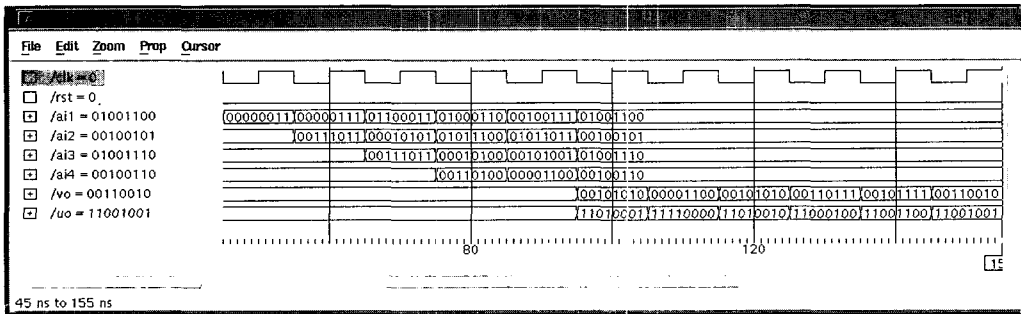
#### IV. VHDL 모델링 및 시뮬레이션

본 장에서는 II장에서 제안한 2차원 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. VHDL 시뮬레이션은 Mentor사의 VHDL 시뮬레이터를 사용하



(a)



(b)

그림 10. Type I의 VHDL 실험 결과

(a) 중간 결과 (b) 최종 결과

Fig. 10. VHDL simulation result of the Type I.

(a) Intermediate result. (b) Final result.

였다.

2차원 DWT 계산을 위하여 제한한 구조인 Type I과 II 구조 (그림 6과 8)를 VHDL 시뮬레이션하기 위하여 사용된 PE인 A PE와 그림 6(b)의 M PE를 VHDL 모델링하였다.

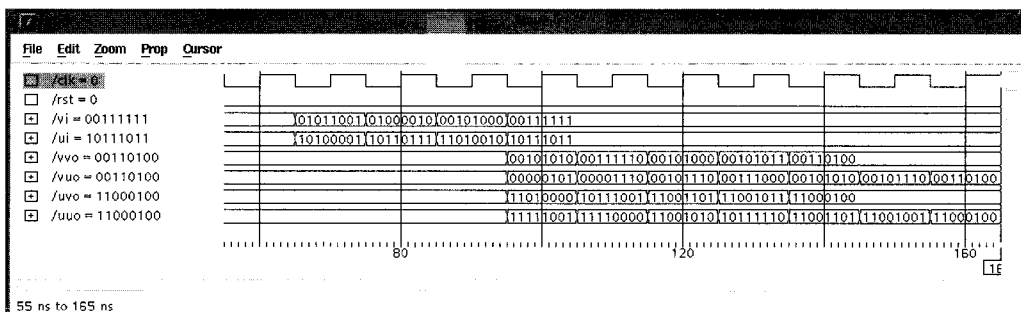
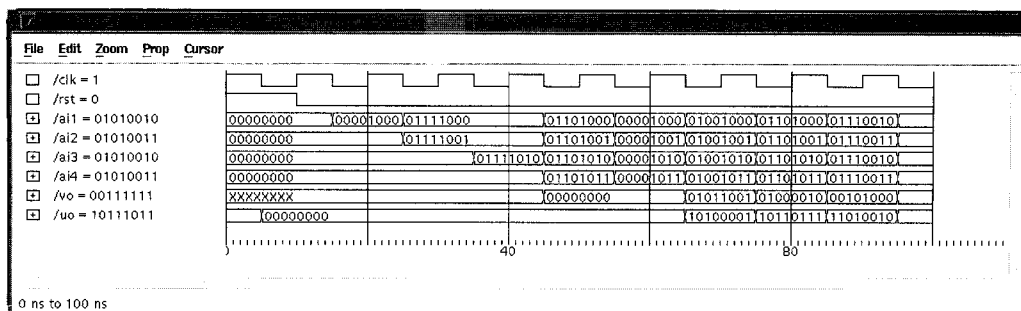
그림 9는 A PE와 M PE의 시뮬레이션 결과로 그림 9(a)의 A PE는 4 개의 값을 입력받아 두개의 합을 출력하는 것이다. 그림 9(b)는 곱셈기 PE의 시뮬레이션 결과로 DWT 저주파 및 고주파 계수를 각각 /h0와 /g0에 "00111110"와 "11000001"로 저장하고 있다. 데이터 /a가 입력되면 출력 데이터 /b는 /h0와 곱셈을 하여 출력하고 /c는 /h1과 곱셈을 하여 출력한다. 예로 클럭 45 ns에 입력 /a에 "00101000"이 입력되면 그 다음 클럭 55ns에 /b와 /c에 각각 /h0와 /g0의 곱셈을 하여 "00001001"과 "11110110"을 출력한다.

그림 10은 Type I 구조의 VHDL 시뮬레이션 결과가

다. 그림 10(a)는 중간 결과로 그림 6(a)의 PE array를 통과한 결과를 나타낸다. 네 개의 PE array의 입력은 /ai1, /ai2, /ai3, 그리고 /ai4이고 첫 번째 PE array의 저주파 출력은 /vo1, 고주파 출력은 /uo1 등으로 나타내었다. 즉 그림 10의 출력은 그림 7을 나타낸다. 그림 10(b)는 최종 결과로 그림 10(a)의 출력을 입력으로 하여 그림 6의 최종 결과를 /vo와 /uo에 출력한다.

그림 11은 Type II 구조의 VHDL 시뮬레이션 결과이다. 그림 11(a)는 중간 결과로 그림 8(a)의 M PE와 A PE를 통과한 결과를 나타낸다. 즉, 그림 8(b)에 나타난 것이다. 네 개의 M PE의 입력은 /ai1, /ai2, /ai3, 그리고 /ai4이고 M PE와 A PE의 저주파 출력은 /vo이고 고주파 출력은 /uo로 나타내었다. 예로 그림 11(a) 입력의 첫 번째 값인 /ai1의 "00001000", /ai2의 "01111001", /ai3의 "01111010", 그리고 /ai4의 "01101011"을 이용하여 출력 /vo와 /uo는 각각 "01011001"과





(b)

그림 11. Type II의 VHDL 실험 결과  
(a) 중간 결과 (b) 최종 결과

Fig. 11. VHDL simulation result of the Type II.  
(a) Intermediate result. (b) Final result.

“10100001”로 출력됨을 확인할 수 있다. 그림 11(b)는 최종 결과로 그림 10(a)의 출력을 입력으로 하여 그림 6의 최종 결과인 저주파-저주파 /vvo, 저주파-고주파 /vuo, 고주파-저주파 /uvo, 그리고 고주파-고주파 /uuo를 출력한다.

### V. 결론

본 논문에서는 2차원 DWT 계산을 위한 효율적인 VLSI 구조를 제안하였다. 제안한 구조중 Type I은 1차원 DWT 계산을 위하여 제안한 PE array를 이용하여 열 방향의 DWT를 계산하고 곱셈기 PE와 덧셈기 PE를 이용하여 행 방향의 DWT를 계산한다. Type II는 곱셈기 PE와 덧셈기 PE를 이용하여 열 방향으로 계산하고 두 개의 PE array를 이용하여 행 방향의 결과를 구한다. 특히 제안한 구조는 저주파 및 고주파 성분을 동시에 구하도록 설계되어 Type I과 Type II를 통과하면 저주파-저주파, 저주파-고주파, 고주파-저주파, 그리고 고주파-고주파를 동시에 구할 수 있다. 제안한 구조

는 기본적으로 행-렬 분해 방법을 사용하지 않고 열 방향의 DWT를 계산하고 그 결과를 바로 행 방향으로 DWT를 계산하므로 기존 구조와 달리 중간 결과를 저장할 필요가 없고 제어가 간단하게 된다.

추후 연구 과제로는 제안한 구조의 정상적인 동작의 확인을 위하여 VHDL로 전산 모의 실험하였는데 실제 chip 설계를 위한 실험이 계속 진행되어야 한다.

### 참고 문헌

- [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.
- [3] 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.
- [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] 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.
- [6] 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.
- [7] 장시중, 김대용, 김순영, 이문호, "정규직교 이산웨이블릿을 위한 효율적인 VLSI 구조," 한국통신학회논문지, 제 23권, 제 1호, pp. 239-252, 1998년 1월
- [8] 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.
- [9] J. Fridman and E. S. Manolakos, "Discrete wavelet transform: Data dependence analysis and synthesis of distributed memory and control array architectures," *IEEE Trans. Signal Processing*, vol. SP-45, no. 5, pp. 1291-1308, May 1997.
- [10] S.-K. Paek and L.-S. Kim, "2D DWT VLSI architecture for wavelet image processing," *Electron. Lett.*, vol. 34, no. 6, pp. 537-538, Mar. 1998.
- [11] C. Yu and S.-J. Chen, "VLSI implementation of 2-D discrete wavelet transform for real-time video signal processing," *IEEE Trans. on Consumer Elec.*, vol. 43 no. 4, pp. 1270-1279, Nov. 1997.
- [12] 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.
- [13] H. T. Kung, "Why systolic architectures?," *IEEE Computer*, vol. 15, no. 1, pp. 37-46, Jan. 1982.
- [14] D. L. Perry, *VHDL*. 2nd Ed., Singapore: McGraw-Hill, 1991.

## 저 자 소 개

潘 聲 範(正會員) 第 31卷 B編 第 7號 參照  
현재 한국전자통신연구원 멀티미디어 연구부 선임연구원

朴 來 弘(正會員) 第 33卷 B編 第 9號 參照  
현재 서강대학교 전자공학과 교수

池 龍(正會員) 第 31卷 A編 第 3號 參照  
현재 서강대학교 전자공학과 부교수