

## 2-Dimensional FIR 디지털 필터의 VLSI 시스토크 어레이 구조 실현에 관한 연구

### A Study on the VLSI Systolic Array Implementation of 2-Dimensional FIR Digital Filter

김 수 현\*, 문 대 철\*

(Soo-Hyeon Kim\*, Dai-Tchul Moon\*)

\* 본 연구논문은 1992년도 교육부지원 한국학술진흥재단의 학술연구 조성비 지원에 의해서 연구되었음.

#### 要 約

2-D FIR 필터를 시스토크 어레이 구조로 실현하는 방법을 제시하였다. 시스토크 어레이는 1-D FIR 필터로 부분 실현한 후 병렬연결하여 구현하였다. 부분 실현한 시스토크 어레이의 마지막 입력신호를 다음 단의 입력에 직접 연결시킴으로써 입력 지연에 사용되는 저장요소를 절약 시킨다. 1-D 시스토크 어레이는 지역통신 접근에 의해 DG를 설계한 후 SFG로의 사상을 통해 유도하였다. 유도된 SFG는 DG의 노드가 보다 작은 수의 PE에 사상됨으로써 PE의 이용률을 개선할 수 있다. 유도된 구조는 매우 간단하며, 입력 샘플이 공급되어지면 매 샘플링 기간마다 새로운 출력을 얻는 매우 높은 데이터 비용(data rate)을 갖는다. 시스토크 어레이는 규칙적이고, 모듈성이며, local interconnection, highly synchronized multiprocessing의 특징을 갖기 때문에 VLSI 실현에 매우 적합하다.

PE 셀 구조는 높은 처리율, 최소 계산시간과 최소 파이프라인 주기를 갖도록 설계하였다.

#### ABSTRACT

In this paper, a realization method for 2-D FIR digital filter is presented, it has been derived by applying a systolic procedure to the SFG(signal flow graph). After we realized the 1-D form partial systolic array, we implemented the complete systolic array to be parralled 1-D form. The cascading input signal of partial systolic array reduce the storage elements which used to delay input signal. 1-D systolic array is derived from that DG(dependence graph) is designed through local communication approach and then mapping it to SFG. And in order to improve PE utilization, it is often desirable to map the nodes of the DG onto a few number of PEs.

The derived structure is very simple and has high throughput because while new input sample is supplied, new output is obtained every sampling period, and broadcast input signal is eliminated. Since the derived systolic array has property of regularity, modularity, expandable, local interconnection and highly synchronized multiprocessing, thus it is very suitable for VLSI implementation.

PE cell structure is designed with high throughput rate, minimum computation time and pipelining period.

• 호서대학교 정보통신공학과  
Dept. of Information & Telecommunications Eng.,  
Hoseo Univ.  
접수일자 : 1993년 3월 30일

## I. 서 론

최근 VLSI 기술의 발전으로 하나의 칩속에 수 백만개의 트랜지스터를 집적화 시킬 수 있는 반도체 IC 설계 기술이 급속하게 발전함으로써 신호처리용 하드웨어를 단일 칩에 집적화시키는 것이 가능해 졌다. 2-Dimensional 디지털 필터는 noise 필터링, feature enhancement, video 통신, biomedical 그리고 air reconnaissance와 같은 디지털 영상처리 분야에 폭넓게 이용되어지고 있다. Kung에 의해서 제안된 시스토크 구조는 modularity, regularity하고 높은 동기처리를 행할 수 있기 때문에 VLSI 설계에 매우 적합하다<sup>[2]</sup>. 그래서 본 논문에서는 2-Dimensional FIR 디지털 필터를 시스토크 구조로 유도하기 위한 알고리즘을 개발하고 실현하였다. 2-Dimensional 디지털 필터의 설계에 대한 다른 형태 실현은 참고문헌 [1][2][4][5]에 나타내었다. 본 논문에서 실현할 시스토크 구조는 2-D FIR 디지털 필터의 반복 알고리즘으로부터 1-D 형태의 알고리즘을 유도하여 1-D FIR 디지털 필터의 시스토크 어레이를 구현하고, 그리고 이를 병렬연결 하여 실현하고자 하는 완전한 2-D 시스토크 어레이를 설계하였다. 1-D FIR 디지털 필터의 시스토크 어레이 구조 유도는 계산 상의 종속을 정확히 이해하고, 가능한 많은 설계중 최적의 설계를 할 수 있도록 알고리즘으로부터 DG를 설계한 후 SFG로 사상하여 구현하고자 하는 시스토크 어레이를 실현하는 방법을 이용하였다<sup>[2][6][7]</sup>.

사상 방법에 의해 다양한 설계가 가능한 SFG는 SFG 사상 조건에 맞도록 retiming을 고려하여 설계하였다. 실현된 1-D FIR 디지털 필터의 시스토크 어레이를 병렬 연결하여 완전한 2-D FIR 디지털 필터의 시스토크 어레이를 구현할 경우, 입력신호들은 각 차수에 비례하는 양만큼의 저장요소를 필요로 하므로 저장 및 지연소자를 줄이기 위해서 입력신호를 직접 연결시켜 2-D FIR 디지털 필터를 실현하였다.

## II. 2-D FIR 디지털 필터 알고리즘

2-D FIR 디지털 필터의 전달함수는 다음과 같다.

$$H(z_1, z_2) = \sum_{i=0}^N \sum_{j=0}^N h(i, j) z_1^{-i} z_2^{-j} \quad (1)$$

여기서  $h(i, j)$ 는 필터 계수이다.

식(1)의 입·출력 관계 recursive 방정식은 다음과

같이 정리할 수 있다.

$$\begin{aligned} y(n, m) &= \sum_{i=0}^N \sum_{j=0}^N h(i, j) \cdot x(n-i, m-j) \quad (2) \\ &+ \sum_{j=0}^N h(0, j) \cdot x(n-0, m-j) \\ &+ \sum_{j=0}^N h(1, j) \cdot x(n-1, m-j) \\ &+ \sum_{j=0}^N h(2, j) \cdot x(n-2, m-j) \\ &\vdots \\ &+ \sum_{j=0}^N h(N, j) \cdot x(n-N, m-j) \\ &= \sum_{i=0}^N y_i(n, m) \end{aligned}$$

$$\therefore y_i(n, m) = \sum_{j=0}^N h(i, j) \cdot x(n-i, m-j) \quad (3)$$

$x(n-i, m-j)$ 에서 인덱스가  $n-i < 0$ , 또는  $m-j < 0$  일 경우에는  $x(0, 0)$  이전의 값이므로 모두 0으로 취급한다. 식(2)의 2-D FIR 디지털 필터는 식(3)의 1-D 형태의 FIR 필터를 설계하여 이를 병렬로 연결하면 2-D FIR 디지털 필터가 실현됨을 보여주고 있다.

식(3)은 형태로 보아  $i$ -index에 무관하다. 따라서 1-D 형태로 취급할 수 있기 때문에 1-D 선형위상 FIR 필터처럼 생각하여, 시스토크 어레이의 부분실현을 위해 식(3)을 다음과 같이 수정한다.

$$y_i(n) = \sum_{k=0}^N h(k) \cdot x(n-k) = \sum_{k=0}^N h(k) \cdot x(n-1-k) \quad (4)$$

여기서,

$$\begin{aligned} y_i(n, m) &= y_i(n), \quad h(i, j) = h(k) \\ x(n-k) &= x(n-i, m-j) \end{aligned}$$

## III. 시스토크 어레이 실현

2-D FIR 디지털 필터의 시스토크 어레이 구조를 실현하기 위해서는 우선 식(4)를 DG, SFG로 사상하여 시스토크 구조를 유도한다.

### 3.1 1-D FIR 필터의 DG 설계

DG(dependence graph)는 알고리즘의 계산상의 종속, 즉 모든 변수들의 종속을 보여주는 그래프이다. 이 단계에서는 알고리즘을 space-time을 기초로 하여 homogeneity하고 broadcast되는 입력과 출력

을 localized하게 DG를 만드는 과정이다. DG를 설계하기 위해서 우선 식(4)를 단일할당코드(single assignment code)로 변형하여 순환 반복형으로 만들면 다음과 같이 된다.<sup>[6][7]</sup>

$$y_n^k = y_n^{k-1} + h(k) \cdot x(n-k) \quad (5)$$

$$y(n) = y_n^N, y_n^{-1} = 0, k = 0, 1, \dots, N$$

식(5)의 DG는 그림1(a)와 같이 된다. 노드들의 아크는 설계자의 선택에 따라 다양하며 또한 다음 단계에서 설계될 SFG의 구조와 이것으로부터 실현될 시스토크 어레이의 구조를 결정한다. 본 연구에서는 계산상의 종속과 설계될 구조의 명확성을 위해 알고리즘의 변수들의 인덱스 매칭과 아크들의 방향을 그림1(a)에서 보여주는 바와 같이 적절하게 결정하였다. 또한 그림1(a)에서 보여주는 바와 같이 입력신호인 입력 데이터  $x(n)$ 과 필터계수  $h(k)$ 는 broadcast하여 localized하지 않기 때문에 입력신호가 broadcast되는 것을 방지하기 위해서는 식(5)를 localized한 순환 반복형으로 다음과 같이 수정 변경하여야 한다.

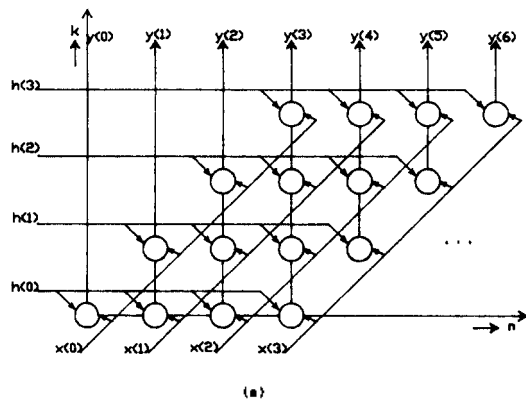
$$\text{즉, } y_n^k = y_{n-1}^{k-1} + h_n^k \cdot x_n^k \quad (6)$$

여기서,

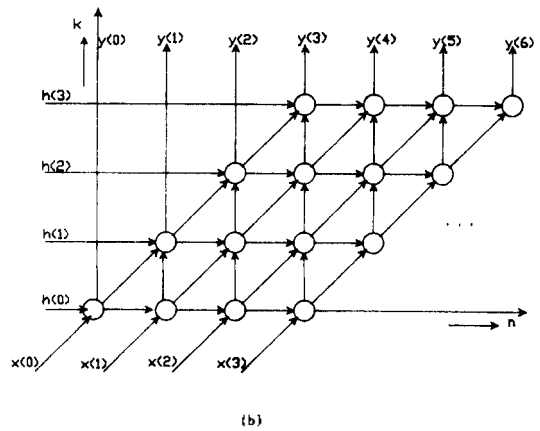
$$x_n^k = x_{n-1}^{k-1}, x_n^0 = x(n)$$

$$h_n^k = h_{n-1}^{k-1}, h_n^0 = h(k)$$

따라서, 식(6)으로부터 localized DG는 그림1(b)와 같이 나타낼 수 있다.



(a)



(b)

그림 1. 종속 그래프 (a)포괄적인 DG (b)지역화된 DG  
Fig. 1. Dependence graph (a)global (b)localized

### 3.2 SFG 설계

DG에서 한개의 노드에 각각 한개의 PE로 지역 반복 알고리즘에 대한 어레이 구조를 결정하면 비동률적인 PE의 응용화를 이룰 수 있다. 따라서 이번 단계는 이용률이 개선된 PE의 응용화를 위해 DG의 각 노드를 보다 적은 PE로 사상시킨 어레이 구조와 DG의 중간 단계인 SFG(signal flow graph)를 설계하는 것이다. 이것은 보다 구체적이며 하드웨어 설계에 가까워 언어적 시스토크 어레이의 일반 형태를 나타낸다. 여기서 설계될 SFG는 2-인덱스 공간이 1-D 프로세서 공간과 1-D 지연공간의 합으로 분해된다. DG로부터 SFG로의 사상은 두 단계로 나눌 수 있다.

첫 단계는 규칙적인 프로세서 배열이다. 즉, 한 직선에서 DG의 노드를 프로세서 어레이 PE로 투영시키는 투영벡터(projection vector)  $\vec{d}$ 를 결정하여 모든 노드들을  $\vec{d}$ 와 평행한 직선상의 하나의 프로세서 배열로 구성한다.

두번째 단계는 규칙적인 프로세서 배열이 이루어지면 스케줄링을 행하여 SFG 구성을 결정한다. 스케줄링은 모든 PE의 연산 순서를 설명한다. 스케줄 벡터(schedule vector)  $\vec{s}$ 는 DG에서의 평행 hyperplane 노드가 같은 시간 단계에서 처리되도록 결정된다.  $\vec{s}$ 는 다음 조건을 만족한다.<sup>[6][7]</sup>

$$\vec{s} \cdot \vec{e} > 0 \quad \forall \vec{e} \quad (7)$$

$$\vec{s} \cdot \vec{d} > 0 \quad (8)$$

여기서,  $\vec{e}$ 는 알고리즘의 신호 흐름도에서 임의의

아크(arc)를 나타낸다. 식(7)은 hyperplane을 가로질러 같은 방향으로 종속 아크가 흐른 것을 설명하며 이것은 스케줄링 기법에서 인과관계를 확실하게 한다. 식(8)은 hyperplane이 투영 벡터에 평행할 수 없음을 설명한다. 이것은 같은 hyperplane 위의 노드가 동일 프로세서로 고안될 수 없음을 보여주는 것이다. 그림 2는 위의 조건을 만족하는 시스토크 스케줄을 나타내었다.

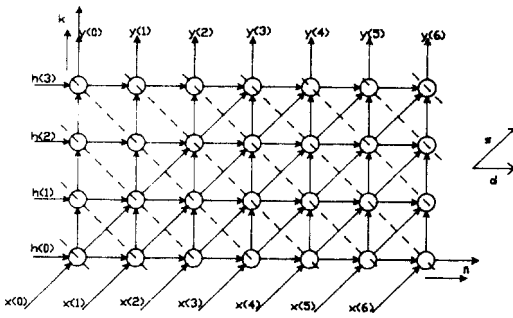


그림 2. FIR 디지털 필터에 대한 시스토크 스케줄.  
 $\vec{s} = [1 1], \vec{d} = [1 0]$

Fig. 2. A systolic schedule for FIR digital filter algorithm,  $\vec{s} = [1 1], \vec{d} = [1 0]$

그림 2의 hyperplane은 다른 순간의 시간을 나타낸다. 일반적인 투영과정은 다음의 3단계로 나누어진다.<sup>[2][7]</sup>

1. 어: 투영 방향에 대해서도 프로세서 공간은 투영방향에 적자이다. 프로세서 어레이는 프로세서 공간에 인덱스 점을 투영함으로써 얻어진다.
2. DG의 아크를 그에 대응하는 프로세서 사이의 zero 또는 nonzero의 지연 엣지로 대치한다.
3. 각 노드가 PE에 투영되고 각 입·출력 데이터가 노드에 연결되기 때문에, 이제 입력과 출력 데이터들 그에 대응하는 프로세서에 인입하여 나타낸다.

위의 과정으로 그림 2의 투영 벡터  $\vec{d}$ 와 스케줄 벡터  $\vec{s}$ 에 의해 나타난 SFG는 그림 3과 같다. 각 수직 선의  $\vec{d}$ 와 평행한 노드들이 SFG의 한 노드로 사상되었다. 스케줄 벡터에 의해 필터의 입력신호가 지연되었다.  $\vec{d} = [1 0]$ 이므로  $\vec{d}$ 에 직각인 프로세서 공간을 만들어 여기에 인덱스점을 투영하여 각 필터계수가 하나의 PE로 할당되었다. 스케줄은 식(7)과 (8)을

만족하도록 하였으며 이 스케줄은 시스토크 스케줄이다.

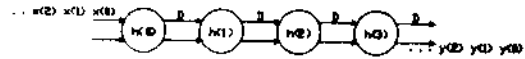


그림 3. 종속 그래프로부터 유도된 SFG  
 Fig. 3. Derived SFG from DG

### 3.3 SFG로 부터 시스토크 어레이 실현

얻어진 SFG는 공간적으로는 localized 하지만 시간적으로는 localized 하지는 않다. 즉, DG로 부터 스케줄 벡터에 의해 결정된 타이밍은 설계될 시스토크 어레이의 타이밍과는 차이가 있다. 이를 해결하기 위해 컷셋(cut-set) 시스토크화 과정등의 retiming을 통하여 최적 pipelining rate, high performance, PE (processor element) 이용율을 높이고, 최소의 지연소자를 갖도록 완전한 시스토크 어레이를 구현하는 것이다. 그림 4는 일반적인 시스토크 어레이의 구조 블록도이다.<sup>[1][2][7]</sup>

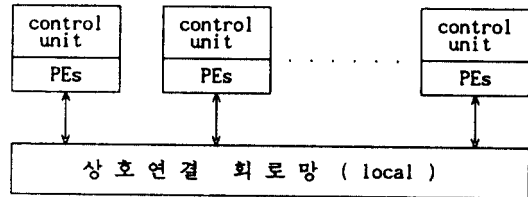


그림 4. 시스토크 어레이 일반구조  
 Fig. 4. General architecture of systolic array

그림 3의 SFG에서는 지연없이 PE로 입력되는 아크들은 지연을 갖지 않으며 따라서 한 샘플링 주기동안 전송 아크를 통해 마치 broadcast되듯 전 PE로 전송되어야만 하는데 이것들이 최소의 지연을 갖도록 하는 것이 retiming이다.

이 과정의 목적은 시간적으로 localized하게 만들어, modular section의 모든 엣지는 적어도 하나의 지연을 갖도록 하는 것이다. SFG의 컷셋은 최소의 엣지를 가지도록 두 부분으로의 분할이다. 컷셋 retiming 과정은 다음 두 규칙을 따른다<sup>[2][5][7]</sup>.

규칙 1. 시간·스케일링(time-scaling): 모든 지연 D는 파이프 라인 주기인 양의 정수  $\alpha$ 와 지연단위 D'로 표시하여  $\alpha D'$ 로 스케일된다.

규칙 2. 지연-전송(delay-transfer) : 컷셋의 방향에 따라 엣지들은 내부경계(in-bound) 엣지와 외부경계(out-bound) 엣지로 묶을 수 있다. 내부경계와 외부경계의 모든 엣지들을 지연단위에 상수배를 해줄 수 있고, 컷셋의 같은 면에 존재하는 입·출력이 다르면 지연은 재조정되며 그렇지 않으면 타이밍은 같다.

retiming을 위한 시스토크화는 1. 기본연산 모듈 선택 2. retiming 규칙 적용 3. 지연과 연산 모듈 조합의 과정에 의해 수행된다. 위의 규칙과 파상에 따라 그림 3의 SFG를 그림 5와 같이 컷셋을 가진 SFG로 나타낼 수 있다.

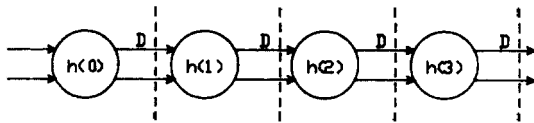


그림 5. 컷셋 retiming 과정을 위한 SFG  
Fig. 5. SFG for cut-set retiming procedure

그림 5에서 컷셋의 동일한 면에 두개의 엣지 방향이 같고 각 엣지가 최소 하나의 지연을 갖도록 해야 하므로 각각 하나의 지연 단위를 더해 주어야 한다. 여기에서  $\alpha=1$ 이다. 이의 과정을 거친 retiming후의 시스토크 어레이는 계수와와의 연산을 위한 입력신호는 2개의 지연을 가지며 각 PE 셀의 출력신호는 다음 PE 셀로의 입력을 위해 하나의 지연을 갖는다. 각각의 PE 셀의 지연 소자가 입력되는 필터계수는 전송선로를 가지지 않고 PE내에 저장되어 연산시마다 적절히 공급되던 된다. 따라서 계수를 위한 저장요소가 필요하다. 그림 6은 컷셋 retiming 과정에 의해서 유도된 시스토크 어레이와 PE 셀의 구조를 보여주고 있다.

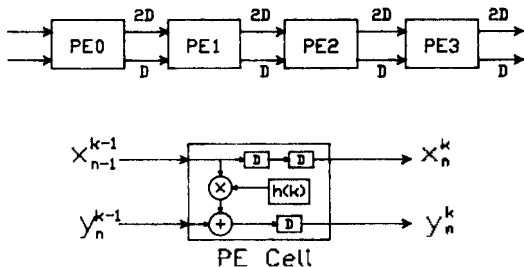


그림 6. 1-D FIR 디지털 필터의 시스토크 어레이 구조  
Fig. 6. Systolic array architecture of 1-D FIR digital filter

각 PE는 필터 계수를 포함하고 매 입력시마다 공급된다. 또한 PE는 부분 출력 데이터의 저장과 지연을 위한 저장요소를 가지고 있으며 이들 모든 저장요소들은 처음 입력되기 전에 클리어된다. PE 셀 구조는 지연소자, 승산기, 가산기 및 저장소자로 구성되어지며, 그림 6에서 각 PE는 특별한 출력 샘플에 대한 모든 부분공의 곱셈 동작과 부분합을 더하여 출력으로 발생하기위한 덧셈동작을 수행한다. 이 구조는 높은 처리율을 갖고, 최소 계산시간과 pipelining 수기를 갖도록 설계되었다<sup>(6)(8)(9)</sup>. 입력은 낮은 필터계수를 가진 노드로 차례로 공급된다. 그리고 매 입력 추가마다 출력이 얻어진다. 유도된 시스토크 어레이 구조는 모듈라하고 결국 구현하고자 하는 고차 필터에 대하여 직렬 접속될 수 있다.

3.5 2-D FIR 필터의 시스토크 어레이

그림 6처럼 구현된 1-D 시스토크 어레이를 이용하여 실현하고자 하는 완전한 2-D FIR 디지털 필터의 시스토크 어레이는 다음과 같은 과정에 의해서 유도된다.

2-D FIR 필터에 대한 알고리즘은 식(1)이고 그림 6의 시스토크 어레이의 출력에 대응하는 알고리즘은 식(3)이다. 따라서 식(4)로 부터 그림 6의 시스토크 어레이를 병렬연결 사킴으로써 쉽게 2-D FIR 디지털 필터의 시스토크 어레이를 설계할 수 있다. 2D 시스토크 어레이는 다음 부분 어레이의 입력을 위한 지연소자로 입력신호의 각 행의 크기만큼 저장요소가 입력 데이터의 열의 크기 수만큼 필요하다. 따라서 입력신호 행의 크기를 필터 차수의 2배로 하여 지연소자를 제거하였다. 이것의 연결도 마치 각각의 PE들을 연결하여 설계하듯 대응하는 입력의 신호들만 고려하면 매우 간단하다. 즉, 병렬연결시에 필요한 지연소자들의 크기가 그 이전의 PE들 내에 설계된 지연소자와 같으므로 마지막 PE의 출력을 다음 단계의 입력으로 연결하면 지연을 위한 별개의 소자가 필요하지 않다. 각 첫번째 PE의 입력지연 소자들은 각 행이 스캔되기 전에 클리어 된다.

2-D 시스토크 어레이의 연산동작 타이밍을 위해 새로운 입력행이 공급되면 그 다음 시간부터 차례로 두번째, 세번째, 마지막 PE의 계수입력을 0로 공급한다. 0 계수입력공급 순서는: Time(8 \* N + 1) : PE \* 1, Time(8 \* N + 2) : PE \* 2, Time(8 \* N + 3) : PE \* 2 : PE \* 3, Time(8 \* N + 4) : PE \* 3, Time(8 \* N + 5) : PE \* 3 이다.

질곡 최종 실현된 2-D FIR 필터의 구현을 그림 7에 나타냈다. 시스토크 어레이의 장점인 modularity로 인해, PE들의 동기화된 다중처리를 위한 제어장치와 각 제어신호는 모든 order에 그리고 모든 PE에 동일하게 입력된다. 따라서 하드웨어 시스템 설계시에 제어부의 설계 자체도 간단하게 된다.

그러므로 그림 6과 식(2)에 의하여 병렬구조 형태인 kernel이 3×3인 경우에 대한 2-D FIR 디지털 필터

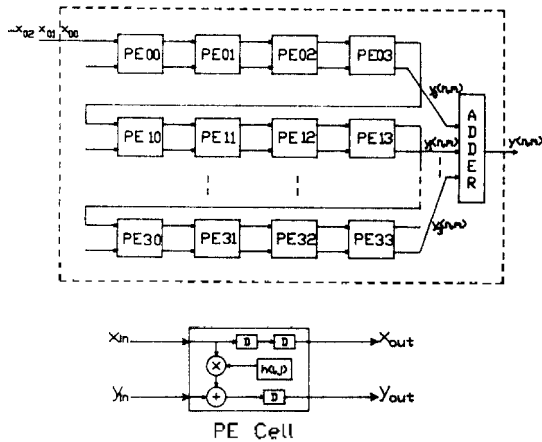


그림 7. 2-D FIR 디지털 필터의 시스토크 어레이 구조  
Fig. 7. Systolic array architecture of 2-D FIR digital filter

표 1. 신호처리를 위한 타이밍도

Table 1. Timing diagram for signal process

(a) Time 1 - Time 8 과정에서 PE00, PE01, PE02, PE03 셀 신호처리 과정

PE 셀 타입	PE 00	PE 01	PE 02	PE 03
1	$x_0e x_0o$			
2	$x_0e x_0o$	$x_0e x_1o$		
3	$x_0e x_0o$	$x_0e x_1o + x_0e x_1e$	$x_0e x_1o$	
4	$x_0e x_0o$	$x_0e x_1o + x_0e x_1e$	$x_0e x_1o + x_0e x_1e$	$x_0e x_1o$
5	$x_0e x_0o$	$x_0e x_1o + x_0e x_1e$	$x_0e x_1o + x_0e x_1e + x_0e x_2o$	$x_0e x_1o + x_0e x_1e$
6	$x_0e x_0o$	$x_0e x_1o + x_0e x_1e$	$x_0e x_1o + x_0e x_1e + x_0e x_2o$	$x_0e x_1o + x_0e x_1e + x_0e x_2o$
7	$x_0e x_0o$	$x_0e x_1o + x_0e x_1e$	$x_0e x_1o + x_0e x_1e + x_0e x_2o$	$x_0e x_1o + x_0e x_1e + x_0e x_2o$
8	$x_0e x_0o$	$x_0e x_1o + x_0e x_1e$	$x_0e x_1o + x_0e x_1e + x_0e x_2o$	$x_0e x_1o + x_0e x_1e + x_0e x_2o$

(b) Time 9 - Time 16 과정에서 PE00, PE01, PE02, PE03 셀 신호처리 과정

PE 셀 타입	PE 00	PE 01	PE 02	PE 03
9	$x_1e x_0o$	$x_1e x_0o + x_1e x_0e$	$x_1e x_0o + x_1e x_0e + x_1e x_0e$	$x_1e x_0o + x_1e x_0e + x_1e x_0e$
10	$x_1e x_0o$	$x_1e x_0o$	$x_1e x_0o + x_1e x_0e + x_1e x_0e$	$x_1e x_0o + x_1e x_0e + x_1e x_0e$
11	$x_1e x_0o$	$x_1e x_0o + x_1e x_0e$	$x_1e x_0o$	$x_1e x_0o + x_1e x_0e + x_1e x_0e$
12	$x_1e x_0o$	$x_1e x_0o + x_1e x_0e$	$x_1e x_0o + x_1e x_0e$	$x_1e x_0o$
13	$x_1e x_0o$	$x_1e x_0o + x_1e x_0e$	$x_1e x_0o + x_1e x_0e + x_1e x_0e$	$x_1e x_0o + x_1e x_0e$

(c) Time 9 - Time 16 과정에서 PE10, PE11, PE12, PE13 셀 신호처리 과정

PE 셀 타입	PE 10	PE 11	PE 12	PE 13
9	$x_0e x_1e$			
10	$x_0e x_1e$	$x_0e x_1e$		
11	$x_0e x_1e$	$x_0e x_1e + x_0e x_1o$	$x_0e x_1e$	
12	$x_0e x_1e$	$x_0e x_1e + x_0e x_1o$	$x_0e x_1e + x_0e x_1o$	$x_0e x_1e$
13	$x_0e x_1e$	$x_0e x_1e + x_0e x_1o$	$x_0e x_1e + x_0e x_1o + x_0e x_1o$	$x_0e x_1e + x_0e x_1o$

터 시스토크 어레이 구조를 그림 7과 같이 설계할 수가 있다. 이 시스토크 어레이에 대한 신호처리 과정은 타이밍도 해석에 의해서 표1과 같이 나타내었다.

타임 8 후에

첫번째 PE셀 PE00, PE10, PE20, PE30의 입력신호에 대한 두번째 지연소자가 클리어 된다.

### IV. 결 론

2-D FIR 디지털 필터에 대한 시스토크 어레이 실현 방법을 나타내었다. 우선 1-D FIR 필터의 어레이를 실현한 후 이를 병렬연결하여 최종 설계를 하였다. 시스토크 어레이 구조 유도는 계산 상의 종속을 정확히 이해하고, 가능한 많은 설계중 최적의 설계를 할 수 있도록 알고리즘으로 부터 DG를 설계한 후

SFG로 사상하여 구현하고자 하는 시스톱릭 어레이를 실현하는 방법을 사용하였으며, 유도된 구조는 매우 간단하며, skew 현상을 피할 수 있도록 모든 입력 데이터들이 localized하게 전송되고 스캔입력이 유도된 구조에 맞도록 제한되면 입력신호의 지연을 위해 PE 내부 지연소자를 사용하지 않게 되어 외부 지연소자는 필요하지 않다. 또한 입력 샘플이 공급되어지면 배 샘플링 기간마다 새로운 출력을 얻는 매우 높은 데이터율(data rate)을 갖는다. 유도된 시스톱릭 어레이는 규칙적이고 modularity, local interconnection, highly synchronized multiprocessing의 특징을 갖기 때문에 VLSI 실현에 매우 적합하다.

### 참 고 문 헌

1. M.A. Sid-Ahmed, "A Systolic Realization for 2-D Digital Filters," IEEE Trans. Signal processing, vol. 37, No.4, pp.560-565 Apr. 1989.
2. S.Y. Kung, "On Supercomputing with Systolic/Wavefront Array Processor," Invited paper, Proceedings of the IEEE, vol.72, No.7
3. M.A. Sid-Ahmed, "Serial Architectures for the Implementation of 2-D Digital Filters and for Template Matching in digital Images," IEEE Trans. Signal processing, vol.38, No.5, pp.853-857, May 1990.
4. Chun-Hsien Chou, "VLSI Architectures for High Speed and Flexible Two-Dimensional Digital Filters," IEEE Trans. Signal processing, Vol.39, No.11, Nov. 1991.
5. S.Y. Kung and J.N. Hwang, "Systolic Array designs for Kalman Filtering" IEEE Trans. Signal Processing, Vol.39, No.1, Jan. 1991.
6. F. EL-Guibaly, S. Sunder, and A. Antoniou, "Systolic Implementation of FIR Filters" Signal processing V: Theories and App. 1423-1426, 1990.
7. S. Y.Kung, VLSI Array Processors, Prentice-Hall, Inc. 1988.
8. 문대철외 1인, "실시간 신호처리용 고속 Multiplier 단 일칩화에 관한 연구," 한국통신학회, 15권 7호 pp. 628-637, 1990.
9. 문대철외 1인, "Systolic array를 이용한 2-Dimension Convolution 설계 및 제작에 관한 연구," 한국통신학회, 15권 10호, pp.819-828, 1990.
10. 문대철, 차규혁, "Digital Filter One Chip화 및 제작," Proceeding of autumn meeting of KITE, pp. 743-746, 1987.
11. I Chen Wu, "A Fast 1-D Serial-Parallel systolic Multiplier," IEEE Trans. Computers, Vol.C-36, No. 10, pp.1234-1247 Oct. 1987.
12. J. V. McCanny, J. G. McWhirter, and S.Y. Kung, "The Use of Data Dependence Graphs in the Design of Bit-Level Systolic Arrays," IEEE Trans. Acoustics, Speech, and Signal Processing, Vol.38, No.5, pp.787-793, May. 1990.

### ▲文 大 哲

현재 : 호서대학교 정보통신과 교수(12권 1E호 참조)

### ▲金 秀 炫



1992년 2월 : 충북대학교 컴퓨터 공학과 졸업

1993년 8월 ~ 현재 : 호서대학교 정보통신공학과 대학원 재학중

1993년 8월 ~ 현재 : 호서대학교 정보통신공학과 실험조교