

## 二進映像處理를 위한 다기능 프로세서 裝置具現에 關한 研究

正會員 李 在 祚\* 正會員 許 允 碩\* 正會員 李 大 寧\*

A Study on the Multi-function Processor Unit  
Implementation for Binary Image ProcessingJae Jo Lee\*, Yoon Seok Heo\*, Dae Young Lee\* *Regular Members*

## 要 約

본 논문에서는 이진영상처리를 위한 다기능 프로세서를 구현하였다. 프로세서는 주소 발생부, 윈도우 파이프 라인, 룩업 테이블, 제어부, 2개의 메모리부로 구성하였다. 본 프로세서는 기존의 SAP(Serial Array Processor)설계기법과 비교하여 구조가 단순하며 처리속도가 향상되었다.

또한 간단한 소프트웨어 선택에 의해서 영상 크기를 선정하며 윤곽검출, 특징점 추출, 세션화, 평활화 등의 기능을 선택적 또는 순차적으로 수행 가능하도록 하였다.

## ABSTRACT

In this paper, a multi-function processor unit is implemented for binary image processing. This unit consists of a set of address generator, window pipeline register, look up table, control unit, and two local memories.

The merits of multi-function processor unit are more simpler than basic SAP and improved disposal speed. A simple software selection give the various choices of image sizes and it can process the function of smoothing, thinning, feature extraction, and edge detection, selectively or sequentially.

## I. 서 론

본 논문은 영상처리의 실용화 과정에서 데이터량의 방대함에 기인한 실시간 처리문제가 가장 중요한

연구과제로 대두됨에 따라서 영상처리를 위한 다기능 프로세서를 구현하므로 이러한 실시간 처리문제를 해결하는데 목적이 있다.

영상처리에 있어서 SIMD(single instruction multiple data)와 파이프라인 구조를 이용한 시스템은 평활화(smoothing), 윤곽검출(edge detection), 세션화(thinning), 특징추출(feature extraction)등과 같은 전처리과정의 고속처리에 적합하며, MIMD(mu-

\*慶熙大學校 工科大學 電子工學科  
Dept. of Electronics Eng. Kyung Hee Univ.  
論文番號 : 93-99

ltiple instruction multiple data) 구조를 이용한 시스템은 영상의 분석이나 인식에 적용하기에 적합하다.

특히 전처리리는 영상 데이터에서 추출할 수 있는 정보의 양을 결정짓는 중요한 단계로서 대부분의 전처리 기법이 윈도우 연산(window operation)으로 처리 가능한 단순 반복적인 알고리즘이지만 영상 데이터의 양이 방대하므로 시간이 많이 소요 된다. 이중 SIMD구조를 이용한 고속 하드웨어 시스템은 다수개의 PE(processing element)에 영상 데이터를 나누어 주고 동일한 처리를 동시에 수행하도록 한 시스템으로 PE의 갯수만큼의 속도 향상과 다양한 처리가 쉽게 구현될 수 있는 장점이 있는 반면, 구조가 복잡하여 시스템을 구현, 상용화 하는데 어려움이 많아 비효율적인 면이 있다. 그러므로 이를 극복하여 저렴한 비용으로 구현 가능하며 병렬성을 충분히 고려한 병렬처리 시스템의 개발이 요구된다. 이러한 요구를 만족시키려면 여러개의 프로세서를 사용하지 않으면서 병렬성을 도출하여야 하므로 공간적인 병렬처리 방식보다는 시간적 병렬처리 방식인 파이프라인(pipeline)기법을 사용하여 한다.[1][2][3][4][6][8][10]

또한 이진 영상에 대한 전처리 과정은 테이블 맵핑(table mapping)기법을 이용한 룩-업 테이블(look-up table; LUT)방식으로 2-메모리 구조가 하드웨어 구현에 적합하다.[7]

따라서 본 논문에서는 파이프라인 구조를 기본하여 LUT방식에 의해 영상표현과 묘사, 패턴인식에서 전처리로 자주 사용되는 평활화, 세선화, 특징추출, 윤곽검출등을 고속으로 처리하기 위한 장치를 2-메모리 구조로 구현하고 이를 응용하여 영상처리 알고리즘을 하드웨어화하여 범용의 영상처리 시스템 구현의 적합성을 타진하고자 한다.

## II. 이진영상처리 알고리즘과 LUT

디지털 영상 패턴의 형태를 인식하기 위한 접근법은 영상 패턴의 구조적 정보가 충분히 유지되면서 보다 데이터 양이 적은 형태로 변환 시키는 것이다. 인식과정은 크게 전처리과정과 특징점 분류과정으로 나눈다. 전처리과정은 대표적으로 잡음제거 및 평활화 과정, 세선화 과정 및 특징점 추출과정, 윤곽 검출 과정을 말한다.

영상 데이터는 2진 영상을 사용함으로써 물체인 "1"과 배경을 나타내는 "0"으로 구분하며, 사용된 원

도우는 3×3의 국부 연산자로 중심화소 값에 대한 8개의 주위 화소를 조사하여 중심화소의 재구성 여부를 판단하게 된다. 전처리 과정 알고리즘은 이미 결정된 템플릿에 순차적으로 연산을 적용 비교하여 패턴을 재구성하는 방법으로 직접적인 로직 연산에 의한 방법이 아닌 화소점의 재구성 결정값이 룩-업 테이블(LUT)에 저장되어 각 해당하는 조건에 따라 결정값을 취하는 테이블 맵핑법을 도입, 사용함으로써 하드웨어 지향적인 장치 구현에 용이성을 제공하였다.[5][7][10]

그리고 3×3 윈도우 연산은 단지 9화소만을 이용하여 단순 반복적인 처리를 하는 것이기 때문에 처리 영상 데이터 크기와 동일한 크기의 메모리 2개를 사용하여 한쪽에는 원 영상 데이터(raw image data), 다른 한 쪽에는 처리된 결과 데이터를 저장하게 하는 방법을 사용한다면 병렬성이 고려되는 알고리즘이라도 쉽게 하드웨어로 구현 할 수 있다. 데이터 처리 부분이 단순한 룩-업 테이블인 경우의 2-메모리 구조(2-memory structure)를 그림 1에 나타내었다.[7]

결론적으로 패턴 인식의 전처리 과정에서 사용되는 윈도우 연산에 대한 단순하고 병렬성이 가능한 하드웨어 구조는 테이블 맵핑 기법을 이용한 2-메모리 구조를 채택하는 것이 적합하다는 것을 알수 있다. 그림 2는 중심화소 P(i,j)에 대한 재구성 조건을 위한 8방향 이웃에 주소 비트 할당과 이 해당 주소에 대한 룩-업 테이블에 적용 방법의 개략도를 보였다.

이러한 전처리 과정에 대해서 간단히 설명하고 본 실험에서 이용한 알고리즘의 LUT의 일부를 작성하여 도시하였다.

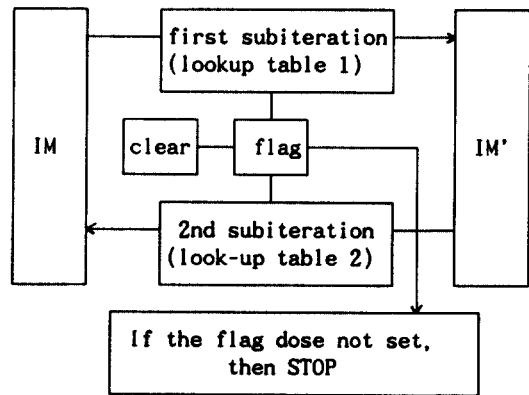
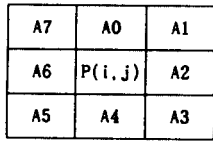


그림 1. 2-메모리 구조  
Fig. 1. 2-Memory structure



(a)

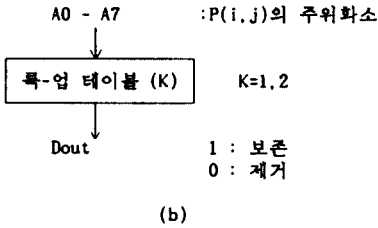


그림 2. (a)8-아웃 화소와 주소 비트  
(b)주소비트의 LUT

Fig. 2. (a)8-neighborhood pixel & address bit.  
(b)LUT diagram of address of address bit.

1. 잡음제거 및 평활화 과정 [11]

영상 입력 장치로 부터 입력된 정보는 일반적으로 인식에 바람직하지 않은 잡음 성분이 섞여있다. 이러한 잡음 성분은 세선화에 의해 제거되지 않고 오히려 특징점으로 오인식되어 인식에 영향을 줄수있다. 따라서 이러한 문제를 없애기 위해 잡음을 제거할 필요가 있다. 또한 이진영상에서는 한개 혹은 두개의 화소가 돌출되거나 들어간 요철성분이 존재한다. 이러한 요철성분을 제거하고, 들어간 요철성분에 데이터를 채워넣는 과정을 평활화 과정이라 한다. 이러한 잡음제거와 평활화를 위한 1st, 2nd 부분 반복 LUT의 일부를 표 1에 보였다.

표 1. 잡음제거 및 평활화를 위한 LUT

Table 1. LUT for noise elimination and smoothing.

low address : A3, A2, A1, A0

테이블A[256]  
1st 부분반복

0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1

high address : A7, A6, A5, A4

low address : A3, A2, A1, A0

테이블B[256]  
2nd 부분반복

0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	
0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1
1	0	0	0	0	0	0	0	0	1	0	0	0	1	0	1	1	1
2	0	0	0	0	0	0	0	1	0	0	0	1	0	1	1	1	1

high address : A7, A6, A5, A4

2. 세선화 과정 [5]

세선화는 이진영상을 단일 화소 두개의 골격 패턴 형태로 변형시키는 것을 말한다. 본 논문에서는 Chen과 Hsu의 병렬세선화 알고리즘을 이용하여 세선화 과정을 처리하였다. 표 2에 1st, 2nd 부분 반복 LUT의 일부를 보였다.

표 2. 세선화를 위한 LUT

Table 2. LUT for thinning.

low address : A3, A2, A1, A0

테이블C[256]  
1st 부분반복

0	0	1	1	0	1	0	0	0	1	1	1	1	0	0	0	0
1	1	1	1	0	1	0	1	0	1	1	1	0	1	0	1	1
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

high address : A7, A6, A5, A4

low address : A3, A2, A1, A0

테이블D[256]  
2nd 부분반복

0	0	1	1	0	1	1	0	0	1	1	1	1	0	1	0	0
1	1	1	1	1	1	1	1	0	1	1	1	0	1	0	1	0
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1

high address : A7, A6, A5, A4

3. 특징점 추출과정 [11]

특징점으로 끝점과 분기점을 추출하고자 한다. 먼저 끝점은 세선화된 영상의 마지막점을 나타낸다. 그리고 분기점은 세선화된 영상에서 가지와 가지가 서로 연결된 점을 나타낸다. 표 3과 표 4는 끝점과 분기점 추출을 위한 LUT의 일부를 보였다.

4. 윤곽 검출과정

이진영상에서 대상물(object)과 배경영역(background)사이의 경계화소를 추출하는 과정을 윤곽검

표 3. 끝점 추출을 위한 LUT

Table 3. LUT for end point extraction.

		low address : A3, A2, A1, A0															
테이블E[256]		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1st 부분반복	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
		high address : A7, A6, A5, A4															

표 4. 분기점 추출을 위한 LUT

Table 4. LUT for branch point extraction..

		low address : A3, A2, A1, A0															
테이블F[256]		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1st 부분반복	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0
	2	0	0	0	0	0	1	0	0	0	1	0	0	0	0	0	0
		high address : A7, A6, A5, A4															

출 과정이라 한다. 이를 추출하기위해서 Sobel연산자 [9]를 사용하였다. 값이 1인 화소가 이웃화소 중 하나라도 "0" 값이 있다면 윤곽점으로 판단한다. 표 5는 윤곽 검출을 위한 LUT의 일부를 보였다.

표 5. 윤곽 검출을 위한 LUT

Table 5. LUT for edge detection.

		low address : A3, A2, A1, A0															
테이블G[256]		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1st 부분반복	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
	2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
		high address : A7, A6, A5, A4															

### III. Serial Array Processor 설계 기법

SAP시스템은 직렬 어레이 처리단들의 연결로 구성시켜 각 단은 단일 클럭 펄스 기간중에 단일 화소의 변환된 값을 생성시키는 일을 수행하도록 한다. 이웃 직렬 어레이 처리(neighborhood serial array processing)단은 이웃논리 변환기(neighborhood logic translator)와 X행의 데이터 입력을 받아 이웃논

리 변환기에 적절한 데이터 배열을 제공하여 국부연산자를 구성시켜 줄 윈도우 파이프라인(window pipeling)과 지연 파이프라인(delay pipeline)으로 구성된다. 이는 영상패턴을 처리하는데 소요되는 시간의 대부분이 각 화소의 주위 화소값을 메모리로부터 읽어오는 시간임을 감안하여, 한번 읽어온 데이터를 연속적으로 사용하도록 구성한것이다.

그림 3은 SAP의 처리 순서에 대한 예로 먼저 그 구성을 설명하면 (A)는 이웃논리 변환기로 적용하고자 하는 처리 기법의 알고리즘을 프로그래밍화 할 수 있으며 파이프라인이 모두 차게되는 순간부터 매 클럭마다 변환된 값을 출력하게 된다. (B)는 윈도우 파이프라인으로 윈도우 레지스터 내용들을 동일 클럭 간격에 중심 화소값의 변형된 값을 계산할 이웃논리 변환기에 병렬로 9개의 값을 투입하게 된다. (C)는 지연 파이프라인으로 메모리에서 열 방향으로 읽혀진 데이터를 윈도우 파이프라인에 각 클럭 간격에서 적절하고 새로운 이웃 구성이 이루어 지도록 한다.

그러나 영상 데이터 배열의 크기가 커지게 된다면 지연 파이프 라인에 해당 윈도우 파이프라인에 적절한 데이터를 입력시키기 위해 영상 데이터 배열 열의 크기에 비례하여 길어져야하는 단점과 파이프라인이 차서 최초의 변환된 데이터가 출력되는 시간이 길어진다는 비효율성을 안고 있다.

본 논문에서는 영상 크기에 상관없이 지연 파이프 라인의 크기를 일정하게하고, 초기화에 필요한 시간을 최소화, 일정화하기 위해 다음과 같이 개선된 SAP 설계 기법을 사용하였다.

개선된 SAP 설계 기법은 메모리 3개를 사용하여 각 메모리에서 영상의 각기 다른 열을 읽어 오게 하는 것이다. 이런 방법을 사용하면 영상의 크기에 시간이 단지 윈도우 파이프 라인의 열(column)수로 고정된다. 여기서는 3클럭으로 윈도우 파이프 라인이 차게 되고 4클럭째 부터는 데이터가 출력되는 장점을 가지게 된다. 이러한 경우 하드웨어는 다소 복잡해지나 영상 크기에 융통성을 부여하고 빠른 처리효과를 볼 수 있다.

개선된 SAP로 데이터는 다음과 같이 처리된다. 그림 4에서 보는 바와 같이 3클럭이 지난후에는 초기화가 끝난다. 이때부터 출력되는 데이터는 유효하다. 예는 처리할 영상 데이터가 다섯개의 화소폭을 가진 2차원 배열로 표현되었을 경우에 연속적인 3개의 개선된 SAP의 처리 단계를 나타낸 것이다.

단계 1에서 변형된 영상 데이터 7'이 출력되며, 단계 2에서는 8', 단계 3에서는 9'이 순차적으로 출력된다.

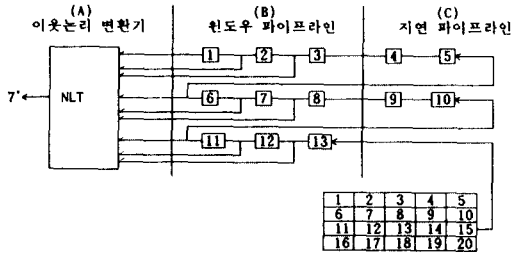
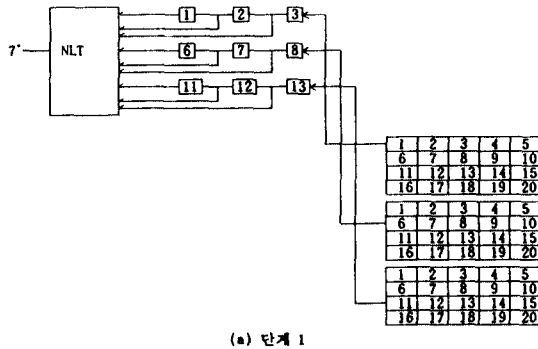
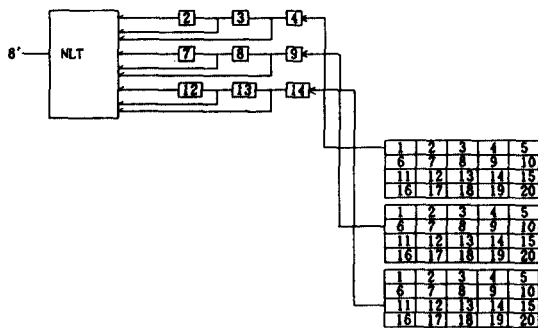


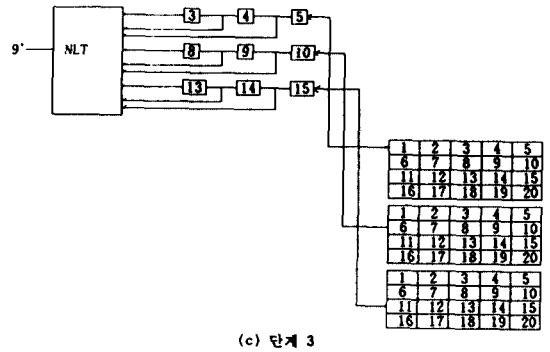
그림 3. SAP 구성도  
Fig. 3. Block diagram of SAP.



(a) 단계 1



(b) 단계 2



(c) 단계 3

그림 4. 개선된 SAP의 처리 순서에 대한 예  
Fig. 4. Example of advanced SAP's processing sequence.  
(a)STEP 1 (b)STEP 2 (c)STEP 3

#### IV. 다기능 프로세서 장치 구조

본 논문에서는 2진화된 영상을 고속으로 처리할 수 있는 장치를 파이프라인 구조 기법을 이용하여 설계하였으며, 이는 IBM-PC에서 전용 프로세서로 동작하도록 하였다. 그림 5에 다기능 프로세서의 구성도를 나타내었다.

다기능 프로세서 장치의 각 부분을 설명하면 다음과 같다.

##### 1. IBM-PC 인터페이스부

인터페이스부는 주 컴퓨터인 IBM-PC에 저장되어 있는 영상 데이터를 다기능 프로세서 장치의 국부 메모리로 이동, 저장시켜 프로세서 장치가 IBM-PC 측에서 볼때 추가된 프로세서로 동작하도록 한다. 8255 포트별로 역할을 구분하여 다기능 프로세서 장치의 초기화, 시작신호등과 같은 제어신호를 제어부에 전달한다. 구현된 장치의 시작신호는 장치내에 자체 클럭발생기를 가지고 있으므로 PC에서 국부 메모리 1로의 데이터 이동은 PC에 의한 외부 클럭으로 동작하게 하고 초기화 후는 클럭 발생기에 절체신호를 줌으로써 장치의 내부 수행이 이루어 지도록 하였다. 또한 기능선택을 통하여 원하는 기능을 선택하여 처리할 수 있게 하였다. 그림 6은 인터페이스부의 블럭 구성도이다.

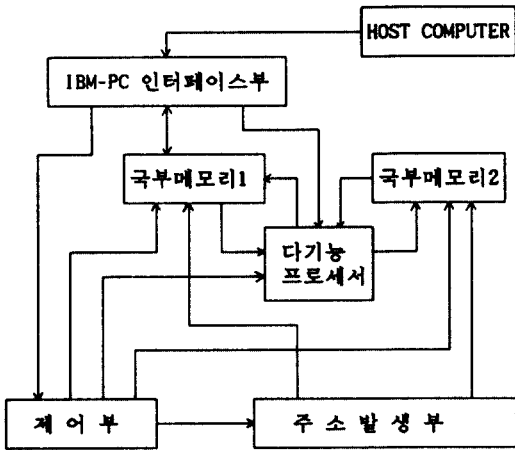


그림 5. 다기능 프로세서 장치 구성도.  
Fig. 5. The structure of multi-function processor.

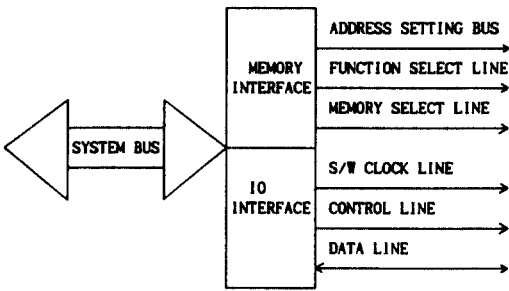


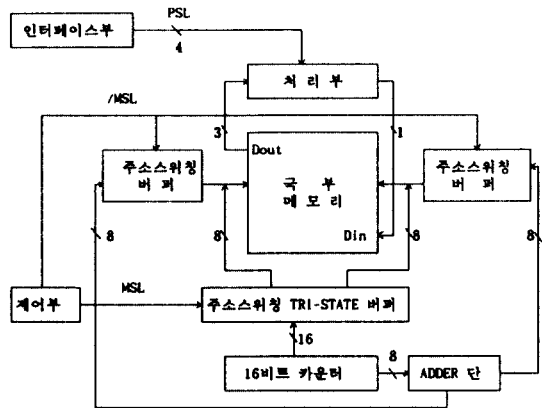
그림 6. 인터페이스 구성도  
Fig. 6. Block diagram of interface.

### 2. 국부 메모리부

국부 메모리 1은 PC측으로 부터 초기 영상 데이터를 입력받아 저장하고 제어부의 시작 신호 후에는 국부 메모리 2와 상호 데이터를 교환하며 LUT에서 출력되는 결과를 저장하게 된다. 열에 따라 해당 행방향을 고려하며 윈도우 레지스터에 데이터를 입력시키기 위해서 행의 크기와 같은 3개의 국부 메모리 소자가 한조를 이루도록 하였다. WRITE시에는 3개의 국부메모리 소자에 동시에 결과 데이터가 기입되지만 READ시에는 3개의 국부메모리 소자가 해당 행번지 데이터만을 윈도우 레지스터에 입력하도록 한다.

### 3. 주소 발생부

1클럭 주기마다 1씩 증가하는 카운터로 이루어지며 2개의 메모리부분에 적절한 주소를 공급하도록 주소 스위칭 역할의 tri-state 버퍼를 사용하였다. 4비트 이진 카운터의 조합에서 발생하는 16비트 주소는 제어부의 조절을 받아 국부메모리에서 데이터 값을 읽어서 윈도우 레지스터로 공급하는 메모리에는 원 카운터의 주소비트를 제공하고 결과값의 데이터를 저장하는 메모리에는 [영상최대열의 크기+처리를 수행하는 열의 크기]에 기입할 수 있도록 Adder단을 통한 조절된 주소 비트를 제공하도록 한다. 국부 메모리부와 주소 발생에 대한 기본 구성은 그림 7에 보였다.



PSL: PROCESSING SELECT LINE.  
MSL: MEMORY SELECT LINE, (READ/WRITE)  
/MSL: REVERSE MEMORY SELECT LINE.

그림 7. 국부 메모리부와 주소발생부 구성도

Fig. 7. Block diagram of local memory and address generation unit.

### 4. 다기능 프로세서부

프로세서부는 처리가 행해지는 부분으로 메모리로부터 읽어온 데이터를 1클럭 주기마다 1쉬프트 데이터 전송을 하기 위해 쉬프트 레지스터로 구성시켜 윈도우 파이프라인 역할을 담당하게 하였으며, 인터페이스부의 Processing Select Line(A7-A4)의 데이터에 따라 각 processing에 해당하는 LUT의 주소 비트를 설정하게 하고 결과값을 LUT로부터 출력 받을 수 있도록 기본 논리 게이트를 조합시켜 주었다. LUT는 2개의 2732A EPROM에 세그먼트를 분할하여 저장시켰으며, 표 6에 각 세그먼트의 처리기능을

표시하였다.

각 EPROM은 16개의 세그먼트로 분할 할 수 있으므로 본 하드웨어 시스템에서는 최대 16개의 기능을 가질 수 있다. 그림 8에 처리부의 구조를 보였다.

표 6. 각 세그먼트의 처리기능

Table 6. The processing function of each segment.

A7	A6	A5	A4	세그먼트	1st 부분반복	2nd 부분반복	테이블 크기
0	0	0	0	0	평활화(1)	평활화(2)	256×2 byte
0	0	0	1	1	세선화(1)	세선화(2)	256×2 byte
0	0	1	0	2	끝점추출(1)		256×1 byte
0	0	1	1	3	분기점추출(1)		256×1 byte
0	1	0	0	4	윤곽 검출(1)		256×1 byte
*	*	*	*	*	*	*	*

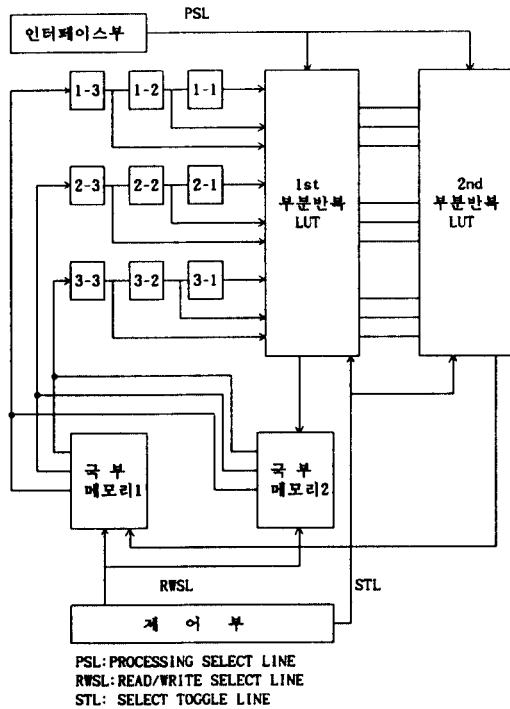


그림 8. 다기능 프로세서부 구성도

Fig. 8. Block diagram of multi-function processor unit.

5. 제어부

장치 초기화 이후 인터페이스부를 통해 PC로부터 모든 제어에 관한 역할을 넘겨 받은 제어부는 주소

발생부에서 생성된 주소 비트를 2개의 구분된 메모리들에 적절히 공급하기 위해서 주소 스위칭 버퍼에 tri-state를 발생시켜 줌과 동시에 해당 반복 부분 수행에 따른 메모리의 상태를 조절하여 준다.

또한 병렬 알고리즘의 특성상 2개의 LUT를 구성 하였으므로 이들 LUT의 선택신호도 발생시켜 주며, 국부 연산자의 이동에 따른 M×N 2차원 영상 배열에서의 행렬 변화를 고려하여 i=1 or i=M or j=1 or j=N인 경우는 중심 화소의 값에 변화를 주지 않도록 논리를 구성하였다.

제어부는 tri-state 버퍼와 플립플롭, 기본 논리 게이트, 단안정 멀티 바이브레이터, 멀티플렉서등 다수의 논리회로로 구성시켰으며 수행과정이 종료되었을 시에는 최종 결과 데이터를 국부메모리1로 이동시켜 PC쪽으로 영상 데이터를 넘겨주게 된다. 그림 9에 제어부에 대한 구성도를 보였다.

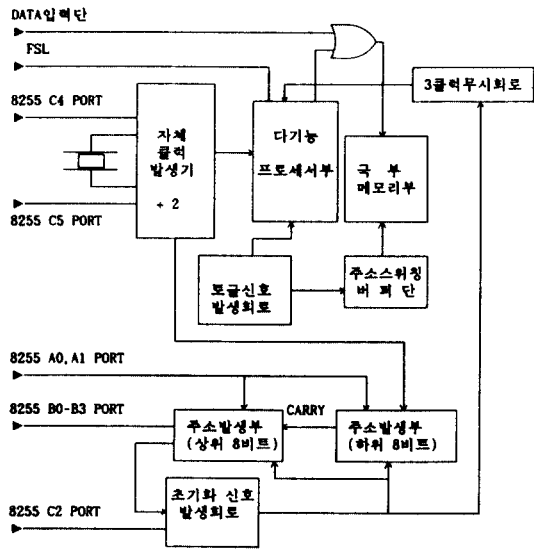


그림 9. 제어부 구성도

Fig. 9. Block diagram of control unit.

V. 실험 및 결과 고찰

본 실험에서는 IBM-PC 확장 슬롯에 인터페이스 카드를 장착하고 프로세서장치를 연결하여 다기능 프로세서 장치로 동작하도록 하였다. 본 연구에서 구현한 다기능 프로세서 장치를 그림 10에 보였다.

구현된 장치에는 LS급 TTL소자를 사용하였으며 소자교유의 전파지연시간(propagation delay time)을 감안하여 4MHz를 50% 듀티(duty)로 2분주 한 500nsec를 1클럭 주기로 자체 클럭 발생부에서 공급하였다.

파이프라인 구조의 장점은 파이프라인이 차게되면 단수에 관계없이 1클럭 주기 마다 하나의 변형 결과를 출력하게 된다는 것이다.

일반적인 병렬처리 구조를 이용한 시스템의 성능 평가 중에서 수행시간(processing time :PT)에 대하여 영상크기 64×64, 반복 횟수 5회인 문자영상 패턴을 계산한다.

$$PT = (N + I) \times T \times R \quad (1)$$

여기서 “N”은 영상크기, “I”는 초기화에 의한 윈도우 파이프라인이 차게되는 지연시간을 표현하며, “T”는 클럭 사이클(500n sec) 또는 화소율(pixel rate)이라 부르며, “R”은 반복수행 연산의 수를 의미한다.

따라서 본 구현 장치에 대한 수행시간은

$$PT = (4096 + 3) \times 500 \text{ n sec} \times 5 = 0.01 \text{ sec} \text{가 된다.}$$

실험은 장치에서의 한글 문자 패턴, 논리회도로, 지문영상에 대한 세선화 처리시간을 소프트웨어 수행시간과 비교하였다. 소프트웨어 수행시간은 여러 가지 시험패턴에 대한 세선화 처리과정을 C언어로 IBM-PC 386(co-processor : 80387)상에서 측정하였다.

표 7. 세선화 처리의 소프트웨어 수행시간과 하드웨어 수행시간의 비교

Table 7. Comparison of S/W processing time and H/W processing time for thinning process.

영상크기	시험패턴	S/W 수행시간 (ms)	H/W 수행시간 (ms)	비율
64×64	디	274	18	13.7
	지	329	20	16.5
	틀	329	18	18.3
128×128	지문영상	659	40	16.5
	논리회로도	659	49	13.5
256×256	지문영상	7143	393	18.2
	논리회로도	2473	196	12.6
평 균				15.6

표 7에서 하드웨어 수행시간은 소프트웨어의 수행시간 보다 약 15배정도 빠른 것으로 나타났다. 이것은 소프트웨어 수행시간은 실제 명령어를 디코더 하는데 걸리는 시간이 있고 또 값이 “1”인 화소의 갯수에 의존하지만, 하드웨어 수행시간은 명령어를 디코더하는 시간이 없고, 값이 “1”인 화소의 갯수에 상관없이 부분 반복수에 의존하기 때문이다.

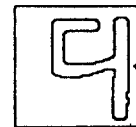
그림 11(a)은 한글 문자 패턴에 대한 입력 영상과 평활화 후의 에지검출 결과영상을 보였고, 그림 11(b)에 지문영상에 대한 입력영상과 평활화 후의 세선화, 특징점 추출 결과 영상을 보였다.



그림 10. 다기능 프로세서 장치  
Fig. 10. Equipment of multi-function processor.



(1)입력 영상



(2)에지 검출 영상

(a)



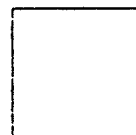
(1)입력 영상



(2)평활화 영상

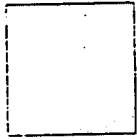


(3)세선화 영상



(4)가지점 추출 영상





(5) 끝점 추출 영상

(b)

그림 11. (a) 문자 실험 영상  
(b) 지문 실험 영상

Fig. 11. (a) Character experiment image.  
(b) Fingerprint experiment image.

고속 연산 수행이 가능한 본 구현 장치의 장점은 다음과 같다.

- (1) 기존의 프로세서 장치가 CPU를 처리부로 사용하여 많은 수행 명령어와 이를 디코드하는데 소요되었던 시간적 손실을 본 장치에서는 단지 클럭펄스 공급에 의해서만 수행하며 CPU를 채택하지 않았다는 것이다.
  - (2) 간단한 I/O 포트 비트 조작으로써 영상 크기와 원하는 기능을 선택할 수 있으며, 처리를 선택적 또는 순차적으로 하여 인식을 위한 전처리 프로세서로 이용할 수 있다. 또한 프로세서는 최대 16가지 기능을 가질 수 있다.
  - (3) 병렬처리 시스템이 안고 있었던 구조의 복잡도를 배제하고 순수한 하드웨어 지향 구조를 지니고 있으며, 소자 구입이 용이하여 실용화가 가능하다.
- 나아가 본 장치의 성능 향상을 위해서는 전파지연 시간이 짧은 소자로의 대체와 주소 발생부에 대한 일부 회로 추가, 변경 및 메모리 용량 확장에 의해 고해상도의 영상처리가 고려되어야 한다. 또한 현재의 VLSI 기술을 도입 시킨다면 본 설계와 제작 장치가 CUSTOM IC로 상용화가 쉬우며 그 처리 능력은 5배 이상(10MHz)의 향상을 기할 수 있다.

## VI. 결 론

본 논문에서는 Stanley R. Sternberg가 제안한 파이프 라인 기법을 개선하여 영상 분석 및 패턴 인식 등의 전처리 단계로 자주 이용되는 평활화, 세션화, 특징추출, 윤곽검출 등을 고속 처리할 수 있는 다기능 프로세서 장치를 구현하였다.

장치는 IBM-PC 인터페이스부, 국부메모리부, 주

소발생부, 다기능 프로세서부, 제어부로 구성된다. 다수개의 CPU를 채택하는 병렬처리 시스템 구조의 복잡성을 피하기 위해 테이블 매핑 기법을 이용한 2-메모리 구조를 이용하였다. 영상 크기에 융통성을 부여 하였으며 5가지 기능을 선택적 또는 순차적으로 처리할 수 있다.

다기능 프로세서부는 직렬 어레이 프로세서(SAP)의 단점인 지연 파이프라인과 윈도우 파이프라인을 더한 수가 영상의 열의 크기와 같아야 한다는 단점을 보완하기 위하여 본 논문에서는 지연 파이프 라인을 제거하고 윈도우 파이프 라인만으로 처리부를 구성하는 설계 기법을 이용하였다. 이러한 설계기법을 사용할 경우 초기화에 필요한 시간이 영상의 크기에 상관없이 일정하게 되어 처리시간을 단축시키는 효과를 얻게되었다.

또한 LUT의 세그먼트의 초기 번지를 각각 선택하여 원하는 처리를 할 수 있으며, 본 장치는 최대 16가지 처리 기능을 가질 수 있으므로 앞으로 기능이 확장될 수 있다. 메모리부는 두개의 독립적 저장 장소로 구성시켜 주소 발생부에서 발생하는 주소를 tri-state를 이용하여 양 방향성 데이터 흐름을 이루도록 하였다.

본 장치는 전처리 과정에서 소요되는 시간적 소모를 최소화 하는 실시간 처리부를 구성, 적용시켜 본 것으로 만족할 만한 처리 속도를 얻었다. 구현된 장치 제어가 간단한 논리에 의한 하드웨어 지향적 시스템으로 디지털 영상처리 알고리즘의 고속화를 이룩할 수 있는 범용의 영상처리 시스템 구현 가능성을 확인 하였고, 앞으로 인식 프로세서의 개발에 한 부분을 담당하리라 본다.

## 참 고 문 헌

1. Hwang K., Briggs F.A., COMPUTER ARCHITECTURE AND PARALLEL PROCESSING, pp.1-51, Mcgraw Hill, 1984.
2. Anthony P. Reeves, "Parallel Computer Architectures for image Processing" CVGIP Vol.25, pp.65-88, 1984.
3. D. Lavie, W.K. Taylor, "A micorprocessor-controlled real-time image processor," IEE PROCEEDINGDS, Vol.130, Pt. E.No.5, SEPTEMBER 1983.

4. H.F. Li, C.M. Tsang, "Pattern recognition for automated wire bonding" IEE PROCEEDING-DS, Vol.131, Pt. E.No.1, JANUARY 1984.
5. Chen Y.S., Hus W.H., "A modified fast parallel algorithm for thinning digital patterns," Pattern Recognition Letters 7, pp.99-106, Feb. 1988.
6. Stanley R. Sternberg, "Pipeline Architecture for Image Processing," Multicomputer and Image processing, pp.291-305, ACADEMIC PRESS, 1982.
7. Chen, Y.S., W.H. Hsu, "A multi-function parallel processor for binary image processing," Proceeding of National Computer Symposium 1985.
8. C.D McIlroy, "Hardware for real-time image processing," IEE PROCEEDING, Vol.131, Pt. E. No.6, NOVEMBER 1984.
9. N. Ranganathan, "A VLSI Architecture for Dynamic Scene Analysis," CVGIP: IMAGE UNDERSTANDING, Vol.53, March. pp.189-197, 1991.
10. 허윤석, "영상의 고속 세션화 장치 구현에 관한 연구," 경희대학교 대학원 석사학위논문, 1990.
11. 남호원, "영문자 인식 및 전처리용 신경칩의 설계," 한국통신학회 논문지 '90-6 Vol.15 No.6.



**李在 聃(Jae Jo Lee) 正會員**  
 1968年 8月 12日生  
 1990年 2月 : 경희대학교 문리대학 물리학과 졸업(이학사)  
 1992年 2月 : 경희대학교 대학원 전자공학과 졸업(공학석사)

1992年 5月 ~ 1993年 7月 현재 : 한국전기연구소 전력전자 연구부 전력통신실 연구원 근무  
 ※주관심분야 : 영상처리, 패턴인식, 병렬처리 등임.



**許 允 碩(Yoon Seok Heo) 正會員**  
 1964年 2月 26日生  
 1987年 2月 : 경희대학교 공과대학 전자공학과 졸업(공학사)  
 1990年 2月 : 경희대학교 대학원 전자공학과 졸업(공학석사)

1989年 12月 ~ 1993年 7月 현재 : (주)신도리코 기술연구소 주임연구원 근무  
 ※주관심분야 : 영상처리, 병렬처리, 패턴인식 등임.

**李 大 寧(Dae Young Lee) 正會員**  
 1940年 3月 18日生  
 1968年 9月 ~ 1970年 3月 : 캘리포니아 주립대학원 졸업(공학석사)  
 1976年 9月 ~ 1979年 9月 : 연세대학교 대학원 전자공학과 졸업(공학박사)  
 1971年 9월 : 경희대학교 공과대학 전자공학과 조교수  
 1977年 3월 : 경희대학교 공과대학 전자공학과 부교수  
 1982年 3월 ~ 현재 : 경희대학교 공과대학 전자공학과 정교수  
 1988年 1월 ~ 현재 : 한국 통신학회 이사