

# 생산자동화 시스템에서 실시간 물체인식을 위한 디지털 뉴런프로세서의 설계 및 알고리즘 개발\*

홍봉화\*\* · 이승주\*\*\*

## 요 약

본 논문에서는 캐리 전파가 없어 고속연산이 가능한 잉여수계를 이용하여 생산자동화 시스템에서 실시간 물체인식을 위한 디지털 뉴런프로세서의 구현방법을 제안하였다. 설계된 디지털 뉴런프로세서는 잉여수계를 이용한 MAC 연산기와 혼합계수 변환을 이용한 시그모이드 함수 연산 부로 구성되며, 설계된 회로는 C언어 및 VHDL로 기술하였고 Compass 툴로 합성하였다. 최종적으로, LG 0.8 $\mu$ m CMOS 공정을 사용하여 Full Custom 방식으로 설계를 수행하였다. 실험결과, 가장 나쁜 경우일 경우, 약 19nsec의 지연속도와 0.6ns의 연산속도를 보였고, 기존의 실수 연산기에 비하여 약 1/2배정도 하드웨어 크기를 줄일 수 있었다. 본 논문에서 설계한 디지털 뉴런프로세서는 실시간 처리를 요하는 생산자동화 시스템의 물체인식 시스템에 적용될 수 있을 것으로 기대된다.

## 1. 서론

최근 영상신호처리 및 패턴인식 분야에서 대량의 데이터를 실시간으로 처리하여야 하는 필요성이 증가하고 있다. 컴퓨터와 사용자간의 인터페이스 문제에 있어서 실시간 처리는 중요한 해결 수단이다. 이와 같은 응용분야에 신경회로망을 이용하기 위해서는 대량의 데이터를 실시간으로 처리할 수 있는 고속 MAC 연산기가 요구된다[1]-[6].

본 논문에서는 신경회로망을 디지털 회로로 구현함으로써 현재 개발된 BP(Back Propagation) 알고리즘을 이용한 신경회로망의 응용알고리즘을 고속으로 처리할 수 있는 디지털 뉴런 프로

세스를 설계 및 구현하고 구현된 프로세서를 생산자동화 시스템에서 실시간 물체인식에 응용하고자 한다. 신경회로망 모델은 행렬·벡터 연산을 기본으로 하고 있으므로 어레이 프로세서의 구조로 구현이 가능하다[7][8].

잉여수계는 모듈러 간의 캐리 정보가 필요 없으며, 승산이 가산과 거의 같은 속도로 구현될 수 있으므로 고속 MAC연산기의 구현이 가능하다[9]-[12].

또한, 뉴런프로세서 구현 시 문제가 되는 시그모이드(Sigmoid)함수 처리는 잉여수계를 이용한 ROM 연산표(ROM Look-Up Table)방식을 사용함으로써 효율적으로 수행할 수 있다. 잉여수를 이용한 MAC연산기의 설계 시 가산과 승산이 동일한 동형으로 연산하기 위하여 순환군을 사용하며, 연산기에 사용되는 순환 부호는 각각의 부호에 한 비트만을 "1"로 하는 코드를 사용함으로써 처리 시간을 줄일 수 있고, 하드

\*본 연구는 해전대학 교내 연구과제로 수행되었음

\*\* 세종대학교 컴퓨터수리정보학과

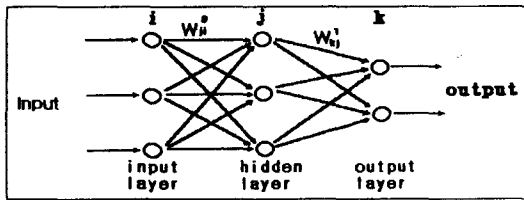
\*\*\* 해전대학 전자과

웨어를 단순하게 구현 할 수 있다[9]-[12].

설계된 MAC 연산기는 200MHZ 이상의 속도로 MAC연산( $AC \leftarrow AC \pm A \times W$ )을 수행할 수 있으며, 이 MAC연산기를 내장한 PE를 어레이 형태로 나열함으로써 대규모의 신경회로망 구현이 가능하다. 또한, 하나의 칩에 64개 정도의 PE를 내장시킴으로써 고속으로 물체를 인식할 수 있는 인식 시스템을 구현할 수 있으므로 로봇의 시각인식 시스템이나, 각종 적응적인 감시 시스템에 적용할 수 있을 것으로 기대된다.

## II. 역전파 알고리즘

역전파 알고리즘은 각층간의 오차를 역전파로 학습하는 모델이며 그림 1과 같은 구조를 갖는다[1]-[6].



(그림 1) 역전파 신경망

(그림 1)에서 j층의 한 노드에 대한 입력( $net_j$ )과 활성화함수를 통과한 후의 출력은 다음 식과 같다.

$$net_j = \sum W_{ji} \cdot O_i = U_j \quad (1)$$

$$O_j = f(U_j) =$$

$$O_j = \frac{1}{1 + \exp(-(net_j + \theta_j)/\theta_0)} \quad (2)$$

( $\theta_j$  : 문턱값,  $\theta_0$  : 기울기)

시그모이드 활성화함수는 출력 값을 0과 1사이의 값으로 제한하며, 다음 단 노드의 입력 값이 된다. 단일노드의 연산은 층의 순서에 의해 식 (1)과 (2)의 연산을 반복 수행하며, 최종 층 k층에서의 출력  $O_k$ 를 출력한다. 여기서 목표 값을  $T_k$ 라하고, 실제 출력 값을  $O_k$ 라 하면, 오차 값 E는 식 (3)의 최소자승오차(LMS: least mean square) 식으로써 구해질 수 있다.

$$E_p = \frac{1}{2} \sum (T_k - O_k)^2 \quad (3)$$

실제출력과 훈련(training)패턴간의 학습과정은 가중치의 변경에 의해 처리된다. 가중치 변경은 식 (3)에 의해 얻어진 오차 E에 의해서 조절된다. 그림 1에서 j층의 어느 한 노드의 p번째 패턴의 오차 값  $\delta_{pj}$ 는 식 (4)와 같다.

$$\delta_{pj} = O_{pj}(1 - O_{pj})\sigma_{pj} \quad (4)$$

여기서 상위 층인 k층의 오차 값에 의하여  $\sigma_{pj}$ 는 식 (5)처럼 표현된다.

$$\sigma_{pj} = \sum \delta_{pk} W_{kj} \quad (5)$$

최종 층 k에서 p번째 패턴 오차  $\sigma_{pk}$ 는 다음 식에 의해서 구할 수 있다.

$$\sigma_{pk} = O_{pk} - T_{pk} \quad (6)$$

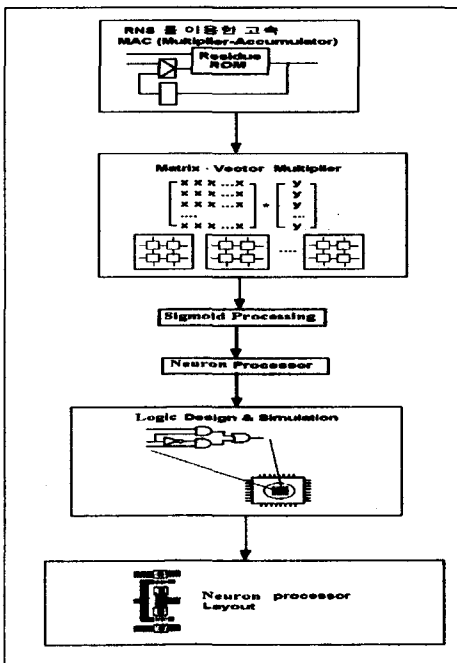
i층과 j층 사이의 t+1단계 가중치 변화량  $\Delta W_{ji}(t+1)$ 은 식 (7)로 구할 수 있다.

$$\Delta W_{ji}(t+1) = \eta(\delta_j O_i) + \alpha \Delta W_{ji}(t) \quad (7)$$

여기에서  $\eta$ 와  $\alpha$ 는 각각 학습계수와 관성계수로서 1회의 가중치 갱신에 따른 비례 값이다. 각 층에서 계산된 오차는 역 방향으로 가중치를 순차적으로 갱신한다.

### III. 디지털 방식을 이용한 역전파 알고리즘의 구현

본 논문에서는 다량의 데이터를 실시간으로 처리할 수 있는 고속 디지털 뉴런 프로세서를 구현하고, 구현된 뉴런프로세서를 생산자동화 시스템에서 실시간 물체인식에 응용함을 목적으로 하며, (그림 2)와 같은 설계단계를 가진다.



(그림 2) 디지털 신경회로망의 설계단계

### 3.1. 어레이 프로세서로 구현

앞 절에서 언급한 역전파 신경회로망의 처리식들은 행렬식을 이용하여 다음과 같이 표현이 가능하다[14]-[22].

$$U = [U_1 \dots U_n]$$

$$O = [O_1 \dots O_n]$$

$$\delta = [\delta_1 \dots \delta_n]$$

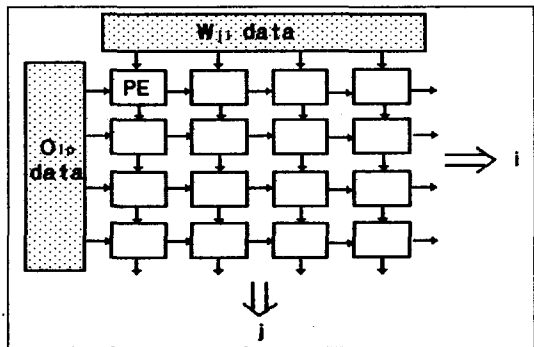
$$T = [T_1 \dots T_n]$$

$$W = \begin{bmatrix} W_{11} & \dots & W_{1n} \\ \vdots & & \vdots \\ W_{n1} & \dots & W_{nn} \end{bmatrix} \text{로 표현되고}$$

$$U = W \cdot O$$

$$O = F(U) \text{이다.}$$

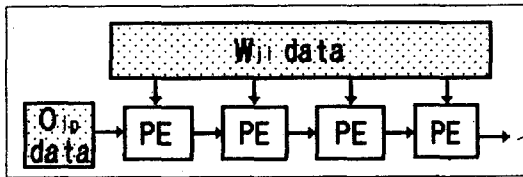
윗 식들은 (그림 3)과 같은 구조의 어레이 프로세서에 의하여 처리될 수 있다.



(그림 3) 신경회로망 모델의 2차원 어레이 프로세서 구현

그러나 (그림 2)와 같이 2차원 어레이 구조를 하드웨어로 구현 시, 속도는 빠르지만 하드웨어 크기가 증대되는 문제가 발생한다. (그림 2)의 구조는 어레이 프로세서 설계 기술에 의하여

(그림 3)과 같이 2차원 구조를 1차원으로 사상하는 것이 가능하며, 이러한 1차원 구조는 대량의 데이터 처리시 분할처리도 가능하기 때문에 본 연구에서는 그림 4의 구조로 설계한다.

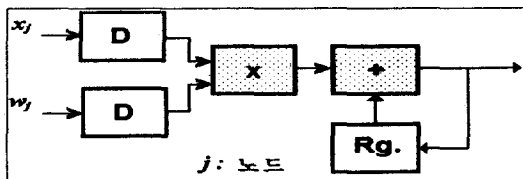


(그림 4) 신경회로망 모델의 1차원 어레이 프로세서 구현

(그림 4)에서 각 PE의 기본연산은 식 (8)과 같은 MAC연산이다.

$$AC \leftarrow AC \pm A \times W \quad (8)$$

식 (8)과 같은 연산을 수행하는 MAC의 기본 구조는 그림 5와 같다.



(그림 5) MAC의 기본구조

그러나, 식 (8)을 현재의 연산회로(정수형과 부동 소수점 연산회로)를 가지고 구현 할 경우, 하드웨어 크기의 증대 및 속도가 저하되는 문제점이 발생됨으로 새로운 방식의 MAC 연산회로 설계와 이를 이용한 어레이 프로세서의 설계방식이 연구되어야 한다.

### 3.2. 잉여수계를 이용한 디지털 뉴런프로세서의 설계

#### 3.2.1. 잉여수의 특징

일반적인 2진 정수계에 의한 연산기는 캐리 정보로 인하여 가산기 및 승산기 설계시 문제가 된다. 특히, 대량의 입력 데이터를 처리하는 영상신호처리, 패턴인식 분야의 경우, 연산기의 크기 및 처리속도 향상에 많은 어려움이 있다. 잉여수계는 가중치가 없는 수체계이며, 각 모듈간에 독립성을 가지므로 캐리 정보가 필요 없고 승산과 가산이 거의 동일한 시간에 이루어질 수 있다는 큰 장점을 갖기 때문에 대량의 데이터를 처리하는 고속의 병렬처리 연산에 유용하다 [9]-[12].

#### 3.2.2. 잉여수계의 기본연산

정수의 잉여수계 표현 방법은 다음과 같다. 서로소인 모듈리 P를 선택하고  $P = \{ m_1, m_2, m_3 \}$ 일때 정수 X의 표시 가능한 범위는 다음과 같다.

$$0 \leq X \leq M \quad (M = \prod_{i=1}^N ) \text{ 이다.}$$

잉여수계 정수 X는 n 테이블로 표시할 때

$$X \xrightarrow{RNS} ( X_1, X_2, X_3, \dots, X_n ) \text{로 된다.}$$

(단,  $X_i = X \bmod m_i = |X|_{m_i}$ )

예를 들면,  $P = \{2, 3, 5\}$ 일 경우, 정수의 잉여수 표현 예를 표1에 나타내었다. 정수  $X = 25$  는  $|X|_{m_1} = 1, |X|_{m_2} = 1, |X|_{m_3} = 0$  즉 (1, 1, 0)로 표현된다.

기본연산은 다음과 같다.

$Z = X \circ Y$  (' $\circ$ ' 연산자: \* + -)가 성립한다.

즉  $|Z|_{m_i} = |X|_{m_i} \circ |Y|_{m_i} = |X \circ Y|_{m_i}$

예를 들면  $X = 12, Y = 5$ 의 가산의 경우

$P = (2, 3, 5)$ 일 때

	mod 2	mod 3	mod 5	
+	0	0	2	: $x = 12$ : $y = 5$
	1	2	0	
	1	2	2	

연산 결과는 <표 1>을 참조하면 17임을 알 수 있다.

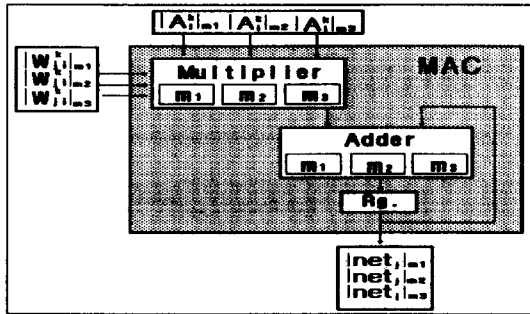
<표 1> 모듈리 (2, 3, 5)의 경우 잉여 수 표현 예

정수 X	잉여수 표현 m1 m2 m3	정수 X	잉여수 표현 m1 m2 m3
0	0 0 0	16	0 1 1
1	1 1 1	17	1 2 2
2	0 2 2	18	0 0 3
3	1 0 3	19	1 1 4
4	0 1 4	20	0 2 0
5	1 2 0	21	1 0 1
6	0 0 1	22	0 1 2
7	1 1 2	23	1 2 3
8	0 2 3	24	0 0 4
9	1 0 4	25	1 1 0
10	0 1 0	26	0 2 1
11	1 2 1	27	1 0 2
12	0 0 2	28	0 1 3
13	1 1 3	29	1 2 4
14	0 2 4	30	0 0 0
15	1 0 0		

3.2.3. 잉여수계를 이용한 MAC 연산기의 구현

잉여수계를 이용한 MAC연산회로의 구성은 (그림 6)과 같다.

(그림 6)에서 잉여수계를 이용한 MAC연산기의 설계 시, 가산과 승산이 동일한 속도로 수행된다. 잉여수계를 이용한 가산과 승산은 순환 균을 형성하기 때문에 바렐 쉬프터(barrel shifter)



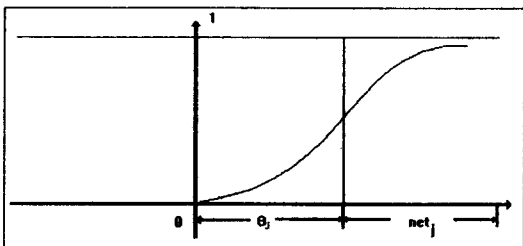
(그림 6) 잉여수계를 이용한 MAC 연산기의 구성

등을 사용하여 고속의 연산회로를 설계할 수 있다[16]-[22].

3.2.4. 잉여수계를 이용한 시그모이드 함수 처리

시그모이드 함수처리는 1절의 식(2)와 같이 표현되며 이 부분은 신경회로망의 디지털회로 구현 시 문제가 되는 부분이다.

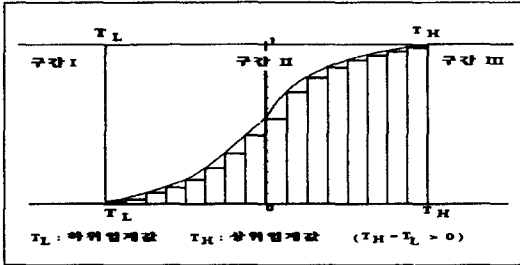
본 연구에서는 이 부분을 잉여수계의 MRC (Mixed Radix Conversion)을 사용하여 구간을 분할하여 처리하고자 한다. 식 (2)에서  $\theta_1, \theta_0$ 에 따라 시그모이드 함수는 그림 7과 같은 특성을 갖는다.



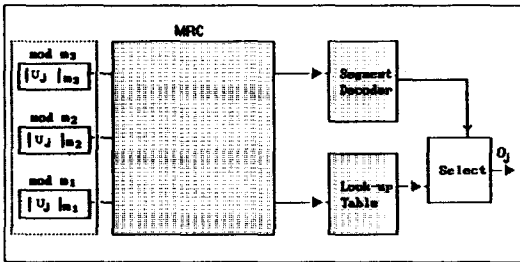
(그림 7)  $\theta_1$  와  $\theta_0$ 에 의한 시그모이드 함수의 이동과 기울기

(그림 7)을 (그림 8)과 같이 3구간으로 분할하

여 구간 분할시 MRC를 사용하며, 시그모이드 함수 처리부분을 (그림 9)와 같이 구성하였다.



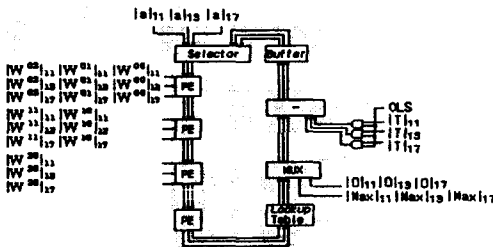
(그림 8) 시그모이드 함수의 구간분할



(그림 9) 시그모이드 함수의 연산처리과정

### 3.2.4. 잉여수계를 이용한 디지털 뉴런프로세서의 설계

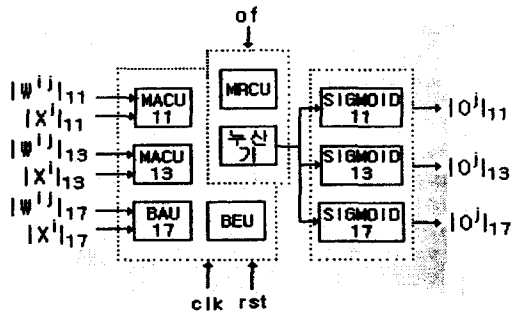
고속의 디지털 신경회로망을 구현하기 위하여 잉여수계와 시스틀릭 어레이 프로세서 구조를 적용하여 (그림 10)과 같이 뉴런프로세서를 구성하였다.



(그림 10) 시스틀릭 어레이 프로세서 기법을 이용한 뉴런 프로세서의 구성

## IV. 디지털 신경 프로세서의 구현

(그림 10)에는 잉여수계를 이용한 신경프로세서의 구성도를 나타내었다. 잉여수계를 이용한 신경 프로세서는 시스틀릭 어레이 구조로 설계되었고 외부에서 인가되는 입력(연결강도 행렬과 입력벡터)은 행렬·벡터 연산을 수행하는 1개 이상의 PE(Processing Element)부, 시그모이드 함수 값을 출력하는 연산표(Lookup Table), 시그모이드 함수 구간에 따라 출력 값을 선택하는 다중채널(Multiplexor), 출력층의 출력값과 목표값의 차이를 계산하는 감산기 및 신경 프로세서의 입력 값과 은닉층의 출력값 중 하나를 선택하는 선택기로 구성된다. 본 논문에서 설계한 신경프로세서는 잉여수계의 모듈러 11, 13, 17를 사용하며, 내부적인 연산에 있어 0~2431 범위의 수로 연산을 수행하고 출력 범위는 0~16값을 가진다.



(그림 11) 잉여수계를 이용한 신경프로세서의 PE 구성도

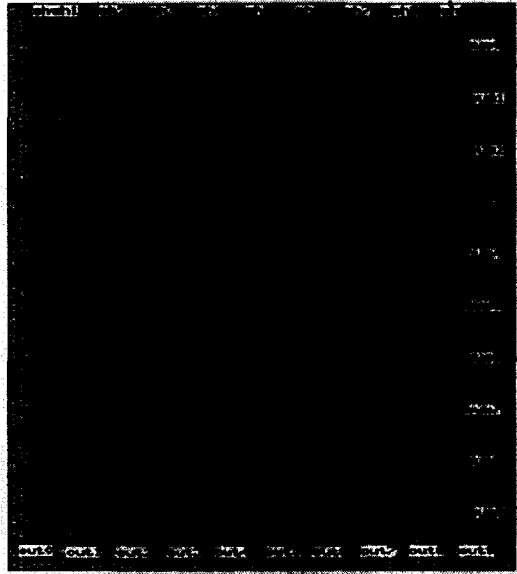
입력벡터  $|a_{11}|$ ,  $|a_{13}|$ ,  $|a_{17}|$ 이 신경 프로세서로 입력되면 선택기는 입력벡터를 선택하여 PE부로 전달되며, PE부는 입력벡터와 연결강도 행렬  $|W_{11}|$ ,  $|W_{13}|$ ,  $|W_{17}|$ 과의 연산을 통해 각 신경의 출력을 발생하고 연산표와 다중선택기를 통해 시그모이

드 함수값을 출력한다. 또한, 은닉층의 출력값과 목표값과의 차이를 감산기가 연산하여 연결강도의 갱신에 사용한다. 이러한 신경프로세서를 이용하여 다층형태로 구성된 신경회로망을 구성하면 (그림 11)과 같이 구성할 수 있다.

(그림 11)에 나타낸 신경 프로세서의 PE는 MAC(Multiplication Accumulation)연산부, 뭉 연산부, 시그모이드 연산부로 구성된다. 여기에서 MAC연산부는 각 모듈리 11의 MACU (Multiplication Accumulation Unit), 모듈리 13의 MACU13, 모듈리 17의 BAU17(Basic Arithmetic Unit) 및 BEU(Base Extension Unit)으로 구성되고 뭉 연산부는 MRCU(Mixed Radix Conversion Unit)와 누산기로 구성된다. 또한, 시그모이드 연산부는 연산표와 멀티플렉서(Multiplexer)로 구성된다. 이러한 구성을 가지는 PE는 각 모듈리 별로 입력되는 입력벡터와 연결강도 행렬의 시냅스 연산을 수행한다. 일반적인 신경회로망의 시냅스 연산은 승산과 가산을 수행하며, 시냅스 연산 결과를 이용하여 시그모이드 함수의 출력 값을 결정한다.

### 4.1. MAC 연산부

MAC연산부는 각 모듈리 11, 13, 17의 입력벡터와 연결강도 행렬의 승산 후 누적연산을 수행한다. 이 경우, 누적되는 값이 모듈리 11, 13, 17로 표현할 수 있는 수의 범위를 넘어 오버플로우가 발생할 수 있으므로 MACU11, MACU13의 출력으로 BEU가 기본확장(Base Extension)을 수행하여 모듈리 17값을 보정하여 BAU17로 입력한다. 모듈리 11, 13, 17의 승산기와 가산기는 (그림 12)에 나타낸바와 같이 작은 규모의 바렐 쉬프터(Barrel shifter)로 구성할 수 있다.



(그림 12) 10×10 Barrel Shifter를 이용한 MAC 연산기의 구현

(그림 12)에 나타낸 Barrel Shifter를 이용하여 연산기를 구성할 경우, 쉬프트하는데 소요되는 시간은 트랜지스터 하나를 통과하는데 걸리는 시간과 같다. 따라서, 바렐쉬프터를 이용하여 연산기를 구성할 경우, 고속연산이 가능하며, 부호정합은 입력선의 재배열로 수행되므로 논리소자가 소요되지 않는다.

### 4.2. 뭉 연산부

MAC연산부에서 출력된 모듈리 11, 13의 값과 오버플로우 방지를 위하여 사용되는 모듈리 17의 연산과정을 거쳐 생성된 값은 MRCU에 의해 스케일링되며, 스케일 된 뭉은 누산기에 의해 누적된다.

### 4.3. 시그모이드 연산부

시그모이드 연산부에서는 시그모이드 출력값

을 연산표에 저장하고 있으며, 몫 연산부에서 스케일되어 누적된 값에 따라 시스모이드 함수 출력을 얻는다.

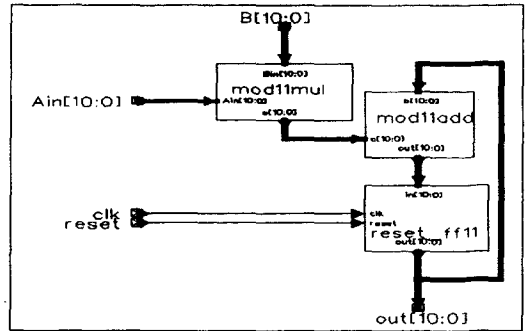
시스모이드 함수값은 "0" 기준으로 대칭으로 지수함수적 변화를 가지는 함수로 "0" 부근에서의 값은 고정된 값을 가지므로 연산표는 "0" 부근의 변화하는 함수값만을 저장하고 멀티플렉서는 몫 연산부의 출력값의 구간을 판단하여 시스모이드 함수 값을 선택하여 출력함으로써 연산표에 저장되는 시스모이드 함수값의 양을 줄일 수 있다.

## V. 실험 및 고찰

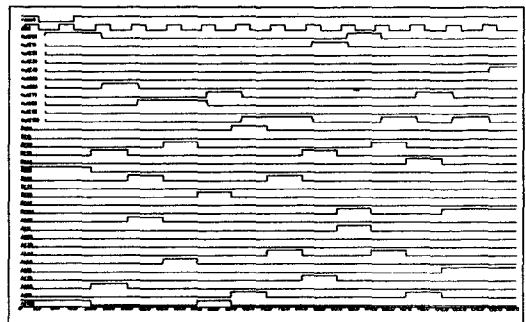
본 논문에서 설계한 고속 디지털 뉴런프로세서는 C 언어 및 VHDL을 이용하여 논리적인 검증을 수행하였으며, 최종적으로 LG 0.8 $\mu$ m CMOS 공정을 사용하여 Full Custom 방식으로 설계를 진행하였다. 모의 실험의 실행조 건으로 입력 값과 연결 강도를 각각 0~10, -32~32의 정수 값으로 하였으며, 그림 8의 시스모이드 함수처리를 위하여 구간II를 9, 11, 15등분하여 실험하였다.

### 5.1. VHDL를 이용한 뉴런프로세서 구현

(그림 13)은 (그림 12)의 바렐쉬프터를 이용한 MAC연산기를 VHDL(Very High Speed Integrated Circuit Hardware Description Language)로 합성한 결과를 나타내며 (그림 14)는 VHDL로 합성한 MAC연산기의 입출력 파형을 나타낸다.



(그림 13) VHDL를 이용한 MAC 연산기의 합성결과

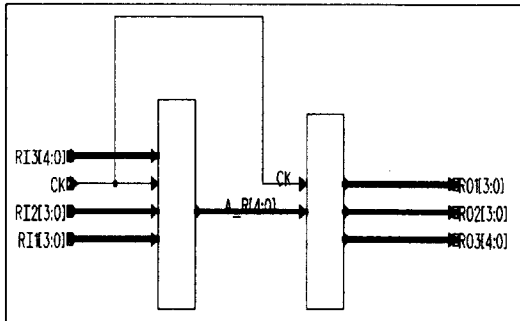


(그림 14) VHDL로 합성한 MAC연산기의 입출력신호

(그림 13)은 모듈리가 11인 경우  $|out_{j+1}|_{11} = A_{j+1}|_{11} | B_{j+1}|_{11} + |out_j|_{11}$  연산을 수행하며,  $|A_{j+1}|_{11}, |B_{j+1}|_{11}, |out_j|_{11}$ 가 각각  $(11^2 + 10^2 + 44)$  비트로 표현되기 때문에 일반적인 정수 연산회로에 비해 작은 하드웨어로 고속의 연산이 가능하다. 그림 13의 MAC연산기의 연산속도를 텍스트 벡터로 작성하여 실험한 결과, 0.6ns안에 연산 결과가 출력되었다. 이 결과를 각층의 노드가 4개인 뉴런프로세서에 적용할 경우, 처음 연산 결과를 얻는데 걸린 시간은 4.8(0.6x2x4)ns가 된다. (그림 15)는 모듈리를 11, 13, 17로 할 경우 모듈리 17를 이용하여 시스모이드 함수 전체 구간을 17개로 분할하고 구간 I를 3개 ( $a_3 = 0, 1, 2$ ), 구간 III를 3개( $a_3=14, 15, 16$ )로 0과 1을 출력하고,

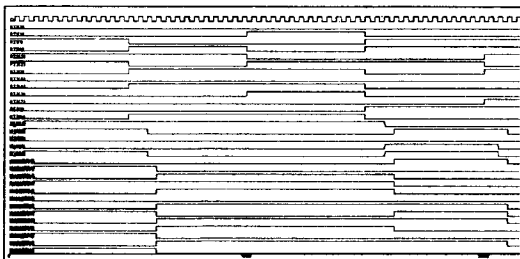


구간 II를 11개로 정하여 시그모이드 함수값을 출력하는 시그모이드 함수처리부의 VHDL 합성 결과를 나타낸다.



(그림 15) VHDL로 합성한 시그모이드 함수 처리부의 논리회로

(그림 16)은 VHDL로 합성한 시그모이드 함수 처리부의 입출력 신호를 나타내며, 설계된 시그모이드 함수처리부는 약 16.9ns의 연산 속도와 7704여개의 Tr수를 보였다. 따라서, 바렐 쉬프트를 이용하여 MAC연산기를 구성할 경우 최대 200Mhz로 동작하는 연산기를 구현 할 수 있다.



(그림 16) VHDL로 합성한 시그모이드 함수 처리부의 입출력 신호

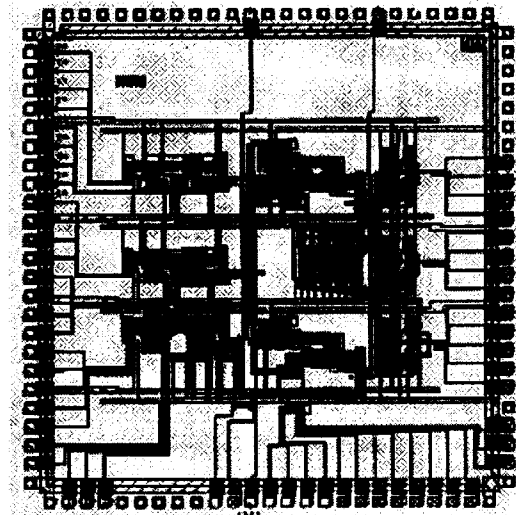
## 5.2. CMOS를 이용한 뉴런프로세서의 구현

C 및 VHDL에 의한 알고리즘 검증이 끝난

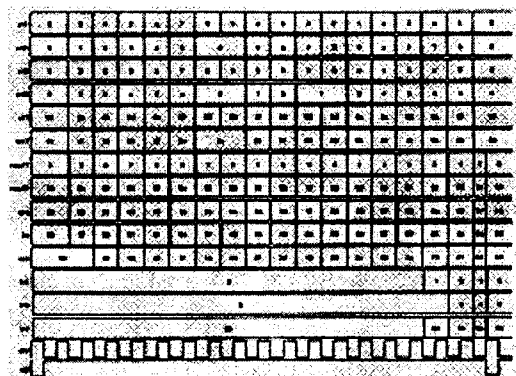
후, Magic 4.1.3 Layout Tool로 레이아웃을 진행하였고 Compass V8r9로 DRC를 수행하였다. 또한 Spice로 각 연산부에 대한 동작 검증을 마친 후, Irsim로 전체 동작을 실험하였다.

설계된 회로는 약 12300개의 Tr로 구성되며, Pad를 포함하여 5m×5m 크기의 die size를 가지며 100핀 QFP 패키지로 제작 구현되었다.

(그림 17)은 잉여수계를 이용한 신경 프로세서의 PE의 레이아웃을 나타내며, 그림 18은



(그림 17) 디지털 신경 프로세서의 레이아웃



(그림 18) 디지털 신경프로세서의 출력 파형

Irsim에 의한 동작 결과를 나타낸다. 역전파 신경회로망의 구현 시, 시스모이드 함수 처리부는 칩 크기를 고려하여 칩 내부에 구현하였다.

또한, <표 2>는 실수 연산기를 이용한 뉴런 프로세서와 잉여수(모듈리 11, 13, 17)를 이용하여 뉴런프로세서를 구현할 경우, 하드웨어의 크기 비교를 나타낸다.

<표 2>에서 알 수 있듯이 수의 범위를 1024로 할 경우, 실수 연산기를 이용한 뉴런프로세서는 총 20652 여개의 Tr로 구성되며, 잉여수(모듈리 11,13,17)를 이용한 뉴런프로세서는 6300 개의 Tr로 구성된다. 따라서 잉여수계를 이용한 방법이 일반적인 실수 연산기를 이용하여 구현한 뉴런프로세서에 비하여 하드웨어 크기가 1/2 이상 감소됨을 고찰하였다. 또한, 승산기, 가산기 및 시그 모이드 함수 처리부의 속도를 각각  $t_m$ ,  $t_a$ ,  $t_s$ 라 할 경우 뉴런프로세서의 속도( $t_N$ )는  $t_N = \max(t_m, t_a, t_s)$ 가 된다.

스 문제에 있어서 실시간 처리는 중요한 해결 수단이다. 이와 같은 응용 분야에 신경회로망을 이용하기 위해서는 대량의 데이터를 실시간으로 처리할 수 있는 고속의 MAC 연산기와 시그모이드 함수처리를 위한 연산기가 요구된다.

잉여수 연산은 정수연산을 수행하며, 수를 각각의 모듈리로 분리하여 연산함으로 모듈리간에 캐리 정보가 필요치 않다. 그러므로 연산기 크기가 감소하며, 고속의 연산기 설계가 가능하다. 순환군은  $\text{mod } p-1$ ( $p$ :소수) 승산이  $\text{mod } p$ 의 가산과 동형이므로 부호정합에 의하여 승산기 설계가 가능하다. 그러므로 본 논문에서는 순환 부호를 이용한 잉여수 연산시, 소수만을 모듈리로 사용함으로써 수 범위가 확장되는 단점을 감소시켰다. 또한, 신경회로망 구현시 문제가 되는 시그모이드 함수 처리는 MRC를 이용하여 활성 영역을 세 구간으로 분할하고 제 II구간을 최대 모듈리를 이용하여 등분한 표본 값을 연산표

<표 2> 잉여수계를 이용한 뉴런프로세서와 실수 연산기를 이용한 뉴런프로세서의 크기비교

연산기의종류	연산기	연산기의 크기(Tr)		표현수의 범위	Tr수
		승산기(10x10)	가산기(20bit)		
실수 연산기를 이용	MAC 연산기	승산기(10x10)	8748	210=1024	20652
		가산기(20bit)	928		
		레지스터(20bit)	828		
	시그모이드 처리부	10148(10x17)			
잉여 수계를이용	MAC 연산기	승산기(10x10)	2316	11x13 x17=2431	12308
		가산기(20bit)	2000		
		MUX( 2 : 1 )	288		
	시그모이드 처리부	7704			

## VI. 결론

영상신호처리 및 패턴인식 분야에서 대량의 데이터를 실시간으로 처리하여야 하는 필요성이 증가하고 있다. 컴퓨터와 사용자간의 인터페이

(Look Up Table)에 저장하여 두고 이용함으로써 연산표의 크기 및 연산속도의 향상을 기대할 수 있다.

즉, 디지털 신경회로망의 고속화 및 하드웨어 크기를 줄이기 위하여 역전파 신경회로망의 전방향 연산을 수행하는 단일 뉴런 프로세서를 잉

여수계를 적용하여 설계하였으며, 알고리즘의 타당성 및 동작 검증을 위하여 C언어 및 VHDL을 이용하여 회로를 기술하고 Compass tool로 합성하여 논리검증을 수행하였다. 최종적으로 LG 0.8 $\mu$ m CMOS 공정을 사용하여 Full Custom 방식으로 설계를 수행하였다.

모의 실험결과 목표 값과의 오차가 0.005 이상인 경우 실수 연산기에 비해 3배 이상의 빠른 수렴 결과를 보였으며, 실제 회로 합성 결과에서는 0.6ns의 연산속도와 약 12300개의 Tr수를 보임으로써 약 1/2 정도 하드웨어 크기를 줄일 수 있었다. 또한, 2진 연산과정의 캐리 전달을 고려할 경우, 본 논문에서 설계한 연산기가 기존의 실수 연산기에 비하여 유리함을 확인 할 수 있었다. 따라서 본 논문에서 설계한 디지털 신경회로망은 고속의 처리를 요하는 생산자동화 시스템에서 실시간으로 물체인식을 위한 응용분야에 적용될 수 있을 것으로 기대된다.

앞으로의 연구방향은 본 논문에서 구현한 뉴런프로세서를 실제 응용화하기 위한 연구가 계속되어야 할 것이다.

## 참고문헌

- [1] D.E. Rumelhart, J.L. McClelland, et al., *Parallel Distributed Processing, Vol.1*, MIT Press, 1986.
- [2] *DARPA neural network study*, AFCEA International Press, 1987.
- [3] You-Han Pao, *Adaptive Pattern Recognition and Neural Networks*, Addison Wesley Publishing Company Inc. 1989.
- [4] B. Widrow and M.A. Lehr, 30years of adaptive neural networks: Perceptron, madaline, and back propagation, *Proceedings of the IEEE, Vol.78*, pp. 1415-1442, September 1990.
- [5] R.P. Lippman, An introduction to computing with neural nets, *IEEE Acoustics, Speech, and Signal Processing*, pp.4-22, April 1987.
- [6] K. Fukushima, A neural network for visual pattern recognition, *IEEE Computers, Vol.21*, No.3, pp.65-75, March 1988.
- [7] C.A. Mead and M.A. Mahowald, A silicon model of early visual processing, *Neural Networks, Vol.1*, pp.91-98, 1988.
- [8] Judith E. Dayhoff, *Neural network architecture*, Van Nostrand Reinhold, 1990.
- [9] Nicholas S. Szabo, M.S. & Richard I. Tanaka, Ph.D, *Residue arithmetic and its applications to computer technology*, McGRAW-Hill Book Company, 1987. 2.
- [10] James A Freeman & David M. Skapura, *Neural networks algorithms, applications, and programming techniques*, Addison-Wesley Publishing Company, Inc. 1991.
- [11] K.H.O. Keefe, A note on fast base extension for residue number systems with three moduli, *IEEE Transaction on Computers, Vol.C-24*, pp.1132-1133, November 1975.
- [12] D.K. Banerji and J.A. Brzozowski, On translation algorithm in residue number systems, *IEEE Transaction on Computers, Vol. C-21*, pp.1281-1285, December 1972.
- [13] G.A. Carpenter, Neural network models

- for pattern recognition and associative memory, *Neural Networks, Vol.2*, pp. 243-257, 1989.
- [14] L.E. Arlas and Y. Suzuki, Digital systems for artificial neural networks, *IEEE Circuits and Device Magazine*, pp.20-24, July 1990.
- [15] H.P. Graf and L.D. Jackel, Analog electronic neural network circuits, *IEEE Circuits and Devices Magazine*, pp. 44-55, July 1989.
- [16] 조원경, "RNS를 이용한 연산 프로세서의 설계에 관한 연구", 한양대학교 박사학위 논문, 6, 1986.
- [17] 정윤돈, "디지털 신경회로망의 시그모이드 함수 연산 회로 설계에 관한 연구", 경희대학교 대학원 전자공학과 석사학위논문, 1992.
- [18] 홍봉화 외 3, "디지털 신경 회로망 실현을 위한 어레이 프로세서의 설계", 인공지능 신경망 및 퍼지 시스템 워크샵, pp.199-208, 11. 1991.
- [19] 윤현식 외 1, "잉여수계를 이용한 디지털 신경회로의 실현", 전자공학회 논문집 제 30권, B편 제2호, 1993.
- [20] 윤현식, "잉여수계를 이용한 고속 디지털 신경망의 설계", 경희대학교대학원 전자공학과 박사학위논문, 1994.
- [21] 홍봉화 외 1, "잉여수계를 이용한 역전파 신경회로망 구현", 「정보학 연구」, 제2권 제2호, 한국정보기술전략혁신학회 pp.145-161, 1999.
- [22] 홍봉화 외 1, "시그모이드 함수의 디지털 구현에 관한 연구", 「정보학 연구」, 제4권 제3호, 한국정보기술전략혁신학회 pp. 155-163, 2001.

# Design of the Digital Neuron Processor and Development of the Algorithm for the Real Time Object Recognition in the Making Automatic System

Bong-Wha Hong\* · Seung-Joo Lee\*\*

## Abstract

We proposes that Design of the Digital Neuron Processor and Development of the Algorithm for the real time object recognition in the making Automatic system which uses the residue number system making the high speed operation possible without carry propagation, in this paper.

Consisting of MAC(Multiplication and Accumulation) operator unit using Residue number system and sigmoid function operator unit using Mixed Residue Conversion is designed, The Designed circuits are described by C language and VHDL and synthesized by Compass tools. finally, the designed processor is fabricated in  $0.8\mu\text{m}$  CMOS process.

Result of simulations shows that critical path delay time is about 19nsec and the operation speed is 0.6nsec and the size can be reduced to 1/2 times compared to the neural networks implemented by the real number operation unit. The proposed design the digital neuron processor can be implemented of the object recognition in the making Automatic system with desired real time processing.

---

\* Dept. of Computer aided Mathematical Information Science, Semyung Univ.

\*\* School of Hi-Techmaterials & electronics, Hyejeon College