

論文96-33B-3-22

# 학습 기능을 내장한 신경 회로망의 하드웨어 구현

## (Implementation of Artificial Neural Network with On-chip Learning Circuitry)

崔明烈 \*

(Myung-Ryul Choi)

## 요 약

신경회로망 칩 구현을 위한 개선된 학습 규칙을 소개하고, 학습 규칙 전용 회로를 내장한 점진형 신경회로망 (FANNs : Feedforward Artificial Neural Nets)을 표준형 아날로그 CMOS 기술을 사용하여 구현한다. 학습 규칙으로는 점진형 학습 규칙으로 널리 사용되고 있는 오차역전파 (EBP - Error Back Propagation) 학습규칙을 하드웨어로 구현하기 적합하도록 수정된 오차역전파 (MEBP - Modified EBP) 학습 규칙을 사용하였다. 시냅스 회로로는 입력과 출력을 모두 전압을 단위로 하는 아날로그 4상한 벡터 내적곱을 행하는 선형 곱셈기를 채용하였다. 두 가지의 학습 기능을 내장한  $2 \times 2 \times 1$  FANN 회로가 구현되었다. 각각의 구현된 신경회로망 회로는 주어진 입출력 학습 데이터를 가지고 PSPICE 회로 시뮬레이터를 사용하여 모의실험을 수행함으로써 그것의 동작성을 입증하였다.

## Abstract

A modified learning rule is introduced for the implementation of feedforward artificial neural networks with on-chip learning circuitry using standard analog CMOS technology. Learning rule, is modified from the EBP(Error Back Propagation) rule which is one of the well-known learning rules for the feedforward artificial neural nets(FANNs). The employed MEBP (Modified EBP) rule is well - suited for the hardware implementation of FANNs with on-chip learning rule. As a synapse circuit, a four - quadrant vector - product linear multiplier is employed, whose input/output signals are given with voltage units. Two  $2 \times 2 \times 1$  FANNs are implemented with the learning circuitry. The implemented FANN circuits have been simulated with learning test patterns using the PSPICE circuit simulator and their results show correct learning functions.

## 1. 서 론

인조 신경 회로망(ANNs)은 많은 연구자들에 의해 다양한 ANN 모델과 가지고 소프트웨어나 하드웨어에 의해 구현되어지고 있다. 소프트웨어에 의한 구현 방법은 보통 신경 회로망의 구조와 기능에 근거를 둔

알고리즘을 선택하여 기존의 디지털 컴퓨터 상에서 프로그램에 의해 구현된다. 이러한 소프트웨어에 의한 구현 방법은 이미 구현된 신경 회로망으로부터 다른 신경망 모델이나 다른 학습 규칙(learning rule)을 사용하여 바꿀 필요가 있을 경우, 단지 알고리즘을 프로그램 상에서 바꿈으로써 쉽게 구현할 수 있기 때문에 높은 융통성(flexibility)을 제공한다. 하지만, 신경 회로망 모델과 학습 알고리즘이 미분 방정식을 포함하는 경우 계산 시간(computation time)이 매우 많이 걸린다는 단점을 갖는다.

\* 正會員, 漢陽大學校 制御計測工學科

(Dept. of Control and Inst. Eng., Hanyang Univ.)

接受日字: 1995年2月17日, 수정완료일: 1996年2月15日

이러한 단점들은 하드웨어에 의한 구현 방법에 의해 해결되어질 수 있다. 즉, 하드웨어적 구현은 다이내믹 연속적 프로세싱(dynamic continuous processing)에 따른 빠른 계산 속도를 제공할 수 있다.

신경 회로망의 하드웨어 구현에 있어서 가장 중요한 것은 시냅스(synapse)의 구현에 있다. 시냅스는 보통 선형 곱셈기(linear multiplier)를 이용하여 구현되는데, 사실은 꼭 선형 곱셈기 이어야 할 이유는 없으며, 시냅스가 비선형(nonlinear)인 신경 회로망의 경우에도 그레디언트 시스템(Gradient system) 조건을 만족하여 시스템은 안정하다는(stable) 것이 수학적으로 증명되어있다<sup>11)</sup>.

이러한 신경 회로망의 하드웨어 구현 시, 또 하나의 고려해야 할 사항은 칩 외부에서 학습된 학습 가중치(learning weights)를 칩 안에 단지 프로그램만을 하느냐, 혹은 칩 안에 학습 규칙 전용 회로(on-chip learning circuitry)를 포함시키느냐 하는 것이다.

아날로그 회로를 이용한 신경 회로망의 구현 방법은 온 칩(on-chip) 학습 회로를 칩 위에 내장 시키는 것이 매우 흥미로운 구현 방법 중의 하나이다. 그 이유는 같은 학습 규칙을 사용하여 학습시키려는 데이터에 대응하는 시냅스 가중치 값이 얻어질지라도, 컴퓨터에서 계산된 시냅스 가중치(synapse weight)와 온 칩(on-chip) 학습 회로를 사용하여 얻어진 시냅스 가중치가 일치되는 것이 보장될 수 없기 때문이다. 따라서, 학습된 가중치가 칩에 적용되어 사용될 때, 올바른 동작을 보증하는 것이 매우 어렵게 된다. 즉, 컴퓨터에서 얻어진 가중치는 모든 뉴런 셀들이 같은 특성을 가지고 모두 아무런 결점(fault)없이 동작한다는 가정하에서 얻어지게 되나, 하드웨어상에서는 서로 조금씩 다른 특성을 갖기도 하는 뉴런 셀들과 결점(fault)으로 인하여 동작하지 않는 뉴런 셀들로 구성된 회로에 의해 가중치 값이 결정된다. 이러한 경우 컴퓨터에서 얻어진 시냅스 가중치는 하드웨어와 대응(mapping)하기 어렵고, 여기에서 더 큰 차이점이 발생될 수도 있기 때문에 기대된 신경 회로망 동작이 더욱 어렵게 될 수도 있다.

본 논문에서는 FANN(Feedforward Artificial Neural Network)용 회로구현에 적합한 학습 알고리즘인 수정된 오차역전파(MEBP - Modified Error Back Propagation) 학습 법칙을 소개하고<sup>11)</sup>, 아날로그 4상한 벡터 내적곱을 행하는 선형 곱셈기를 시냅스

회로<sup>12)</sup>를 사용하여 학습 기능을 내장한 FANN을 구현하고, PSPICE 시뮬레이션을 통하여 그것의 동작성을 입증한다<sup>13,4)</sup>.

## II. FANN 칩 구현을 위한 수정된 오차역전파(MEBP) 학습 규칙

신경회로망 칩 구현을 위한 가장 중요한 연구항목의 하나로 학습 규칙(Learning Rule) 회로의 구현을 꼽을 수 있다. 본 장문에서는 아날로그 CMOS 기술을 이용하여 학습기능을 내장한 FANN(Feedforward Artificial Neural Network) 칩 구현을 위하여 오차역전파(EBP) 학습규칙을 개선한 수정된 오차역전파(MEBP) 학습규칙을 소개한다<sup>11)</sup>.

FANN에서의 각 뉴런의 출력 방정식은 다음과 같이 표현된다<sup>15)</sup>.

$$\bar{y}_j = S_j \left( \sum_{i=1}^N \bar{w}_{ji} y_i + \bar{\theta}_j \right) \quad (1)$$

여기에서  $\bar{y}_j$ 는 현재층(present layer)내의 j번째 뉴런의 출력,  $y_i$ 는 전층(previous layer)의 i번째 뉴런으로부터의 출력,  $\bar{w}_{ji}$ 는  $y_i$ 와 현재층(present layer) 내의 j번째 뉴런의 입력단을 연결하는 시냅스 가중치(synapse weight)를,  $\bar{\theta}_j$ 는 현재 층(present layer)내 j번째 뉴런의 문턱 가중치(threshold weight)를,  $S_j$ 는 시그모이드 함수(sigmoid function)를,  $N$ 은 전층(previous layer)의 출력 개수를 의미한다.

오차역전파(EBP) 학습 규칙<sup>15)</sup>에서 오류 함수(error function)는 보통 다음과 같이 정의하고 있다.

$$E = \sum_p E_p$$

$$E_p = \frac{1}{2} \sum_{j=1}^n (t_{pj} - y_{pj})^2 \quad (2)$$

여기에서 목표 패턴(target pattern)  $t_p$ 가 주어졌을 때,  $y_{pj}$ 는 출력층(output layer)의 j번째 실제 출력을 나타내고,  $t_{pj}$ 는  $y_{pj}$ 의 목표 패턴(desired target)이며,  $n$ 은 출력층(output layer)의 뉴런 수를,  $p$ 는 목표 패턴의 색인(index)을 의미한다. 이 때 임의의 시냅스 가중치  $w_{ji}$ 에 대한 수정된 오차역전파(MEBP) 학습 규칙은 다음과 같이 표현된다<sup>11)</sup>.

$$\dot{w}_{ji} = -\eta \left( \frac{\partial S_j}{\partial u_j} \right)^{-1} \sum_p \frac{\partial E_p}{\partial w_{ji}} - \mu_{ji} w_{ji} = \eta \sum_p e_{pj} y_{pi} - \mu_{ji} w_{ji} \quad (3)$$

여기에서  $u_j$ 는 시그모이드 함수  $S_j$ 의 입력이며,  $\mu_{ji}$ 는 기생 소자에 의해 발생하는 매우 작은 값이다. 식 (3)에서 만일 뉴런  $j$ 가 출력층(output layer)내에 존재한다면,

$$e_{pj} = t_{pj} - \bar{y}_{pj} \quad (4)$$

만일 뉴런  $j$ 가 중간층(hidden layer)에 속해 있다면,

$$e_{pj} = \sum_k e_{pk} \bar{w}_{kj} \quad (5)$$

여기에서  $\eta$ 는 학습 가중치를 의미한다. 식 (3) - (5)을 다시 정리하면 다음과 같은 식을 얻을 수 있다.

먼저 뉴런  $k$ 가 출력층(output layer)에 있다면,

$$\dot{w}_{jk} = \eta \sum_p (t_{pk} - \bar{y}_{pk}) y_{pj} - \mu_{kj} \bar{w}_{kj} \quad (6)$$

이며, 뉴런  $j$ 가 중간층(hidden layer)에 있다면,

$$\dot{w}_{ji} = \eta \sum_k \bar{w}_{jk} \sum_p (t_{pk} - \bar{y}_{pk}) x_{pi} - \mu_{ji} w_{ji} \quad (7)$$

여기에서 식 (6)와 (7)는 그레디언트형 학습 규칙 (gradient-type update law)이다. 식 (6)와 (7)에서  $\mu_{ji}$ 가 0일때 식 (6)와 (7)의 학습 규칙은 식 (2)의 오류 함수(error function)가 최소화되게 동작한다. 하지만,  $\mu_{ji}$ 는 실제 회로에서 기생 소자에 의해 발생하는 매우 작은 값으로 존재하게 되며, 이러한 매우 작은  $\mu_{ji}$ 는 학습 규칙 식 (6)와 (7)의 안정성(stability)에 기여하게 되며, 식 (3)은 이론 적으로도 식 (2)의 오류 함수(error function)를 최소화하는 학습 규칙으로 사용될 수 있음을 보여준다<sup>[11]</sup>.

### III. 학습기능을 내장한 3단 FANN 회로

위에서 설명한 수정된 오차역전파 학습규칙을 이용한 2개의 3단(3-layer) FANN 회로를 구현한다<sup>[3,4]</sup>. 3단 FANN이란 입력층(input layer), 중간층(hidden layer), 출력층 (output layer) 각각 한 개씩을 가진 FANN을 의미한다.

#### 1. 학습기능을 내장한 FANN 회로 1

첫번째 구현한  $2 \times 2 \times 1$  FANN 회로는 그림 1에서

보여주고 있는데,  $2 \times 2 \times 1$ 이란 입력층 (input layer)에 2개의 뉴론을, 중간층(hidden layer)에 2개의 뉴론을, 출력층(output layer)에 한 개의 뉴론을 가진 것을 의미한다.

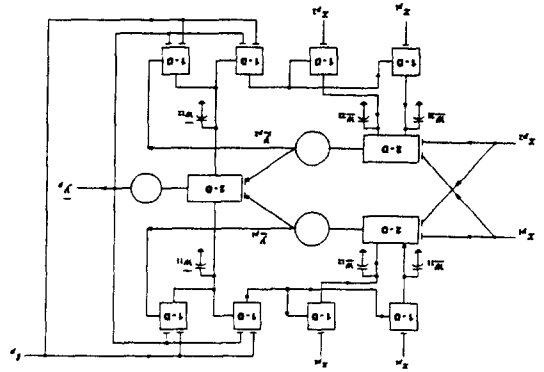


그림 1. 연속형 학습 회로를 내장한 FANN 회로 #1

Fig. 1. The Feedforward ANN with the sequential learning circuit #1.

그림에서  $x_{p1}, x_{p2}$ 는 입력,  $t_p$ 는 목표 패턴(target pattern),  $\bar{y}_p$ 는  $2 \times 2 \times 1$  FANN의 출력층의 출력,  $y_{p1}, y_{p2}$ 는 각각 중간층 뉴런 1과 2의 출력,  $w_{ij}$ 는 입력  $j$ 에서 중간층 뉴런  $i$ 로의 시냅스 가중치,  $\bar{w}_{ij}$ 는 중간층 뉴런  $j$ 의 출력에서 출력층 뉴런  $i$ 로의 시냅스 가중치를 표현하며, 1-D는 스칼라 곱(scalar product)을, 2-D는 2차원 벡터의 내적 곱(inner product)을 계산해내는 곱셈기이다. 그림에서 점진형 신경망회로는 3개의 2-D 곱셈기와 3개의 시그모이드 함수 회로로 구성되고, 학습 회로는 8개의 1-D 곱셈기와 시냅스 가중치를 저장하는 6개의 커패시터로 구성되어 있다. 그림 1의 회로에서 얻어지는 각 뉴론의 출력 방정식을 살펴보면 다음과 같다.

$$y_{p1} = S(k_1 \sum_j w_{1j} x_{pj}) \quad (8)$$

$$y_{p2} = S(k_2 \sum_j w_{2j} x_{pj}) \quad (9)$$

$$\bar{y}_p = S(k_3 \sum_j \bar{w}_{1j} y_{pj}) \quad (10)$$

여기에서  $k_i$ 이란  $1-D$ (차원 벡터 내적) 곱셈기의 상수에 해당한다. 그리고  $\bar{y}_p$ 와  $x_{pj}$ 는 목표 패턴(target pattern)  $p$ 에 대한 실제 출력과 입력에 해당한다.

그림 1의 학습 회로에 대한 다이내믹 학습(dynamic update) 방정식을 살펴보면 다음과 같다.

$$RC \dot{\bar{w}}_{11} = k_1 [(t_p - \bar{y}_p) y_{p1}] - \bar{w}_{11} \quad (11)$$

$$RC \dot{\bar{w}}_{12} = k_1 [(t_p - \bar{y}_p) y_{p2}] - \bar{w}_{12} \quad (12)$$

$$RC \dot{\bar{w}}_{21} = k_1 \bar{w}_{11} [(t_p - \bar{y}_p) x_{p1}] - \bar{w}_{21} \quad (13)$$

$$RC \dot{\bar{w}}_{22} = k_1 \bar{w}_{11} [(t_p - \bar{y}_p) x_{p2}] - \bar{w}_{22} \quad (14)$$

$$RC \dot{\bar{w}}_{21} = k_1 \bar{w}_{12} [(t_p - \bar{y}_p) x_{p1}] - \bar{w}_{21} \quad (15)$$

$$RC \dot{\bar{w}}_{22} = k_1 \bar{w}_{12} [(t_p - \bar{y}_p) x_{p2}] - \bar{w}_{22} \quad (16)$$

여기에서 R은 1-D 곱셈기의 출력 저항을, C는 시냅스 가중치를 저장하는 커패시터를 의미한다. 식 (8)~(10)과 (11)~(16)는 식 (1), (3)과 같은 꼴임을 알 수 있다.

2. 학습기능을 내장한 FANN 회로 2

두번째 구현된 FANN회로는 그림 1의 회로에서 문턱 가중치(threshold weight)를 학습 패턴에 따라 변화시키는 회로를 첨부한 2×2×1 회로이며 그림 2에서 보여주고 있다<sup>1,3,4)</sup>.

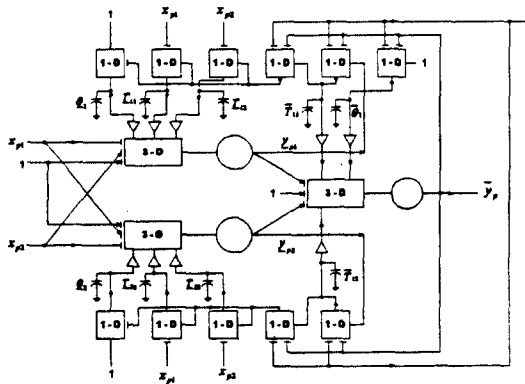


그림 2. 연속형 학습 회로를 내장한 FANN 회로 #2

Fig. 2. The Feedforward ANN with the sequential learning circuit #2.

이 회로에서는 문턱 가중치와 시냅스 가중치는 시그모이드(sigmoid) 함수를 취하는 회로를 첨부하였다. 시그모이드 함수를 통과하기 전의 문턱 가중치와 시냅스 가중치는  $\theta_i$ 와  $T_{ij}$ 이며, 시그모이드 함수를 통과한 후에는  $\theta_i$ 와  $w_{ij}$ 로 표현한다. 그림에서 점진형 신경망은 3개의 3-D 곱셈기와 3개의 시그모이드 회로로 구

성되고, 학습 회로는 2개의 1-D 곱셈기와 10개의 커패시터, 10개의 시그모이드 함수 회로로 구성된다.

이 회로의 다이내믹 학습 시냅스 가중치(dynamic update synapse weight) 방정식을 살펴보면 다음과 같다.

$$RC \dot{\bar{T}}_{11} = k_1 (t_p - \bar{Y}_p) y_{p1} - \bar{T}_{11} \quad (17)$$

$$RC \dot{\bar{T}}_{12} = k_1 (t_p - \bar{Y}_p) y_{p2} - \bar{T}_{12} \quad (18)$$

$$RC \dot{\bar{T}}_{21} = k_1 \bar{w}_{11} (t_p - \bar{Y}_p) x_{p1} - \bar{T}_{21} \quad (19)$$

$$RC \dot{\bar{T}}_{22} = k_1 \bar{w}_{11} (t_p - \bar{Y}_p) x_{p2} - \bar{T}_{22} \quad (20)$$

$$RC \dot{\bar{T}}_{21} = k_1 \bar{w}_{12} (t_p - \bar{Y}_p) x_{p1} - \bar{T}_{21} \quad (21)$$

$$RC \dot{\bar{T}}_{22} = k_1 \bar{w}_{12} (t_p - \bar{Y}_p) x_{p2} - \bar{T}_{22} \quad (22)$$

$$w_{ij} = S(T_{ij}) \quad (23)$$

또한 문턱 가중치 학습(threshold update weight) 방정식은 다음과 같이 얻어진다.

$$RC \dot{\theta}_1 = k_1 \bar{w}_{11} (t_p - \bar{Y}_p) 1 - \theta_1 \quad (24)$$

$$RC \dot{\theta}_2 = k_1 \bar{w}_{12} (t_p - \bar{Y}_p) 1 - \theta_2 \quad (25)$$

$$RC \dot{\theta}_1 = (t_p - \bar{Y}_p) 1 - \theta_1 \quad (26)$$

$$\theta_i = S(\theta_i) \quad (27)$$

그리고, 각 뉴런의 출력 방정식은 다음과 같이 나타난다.

$$Y_{p1} = S_n(k_3 (\sum_j w_{1j} x_{pj} + \theta_1)) \quad (28)$$

$$Y_{p2} = S_n(k_3 (\sum_j w_{2j} x_{pj} + \theta_2)) \quad (29)$$

$$\bar{Y}_p = S_n(k_3 (\sum_j \bar{w}_{1j} y_{pj} + \bar{\theta}_1)) \quad (30)$$

IV. FANN회로 모의 실험 결과

그림 1의 회로는 그것의 동작을 살펴보기 위하여 PSPICE 회로 시뮬레이터를 사용하여 회로 시뮬레이션을 하였다. 본 논문에서 구현된 FANN에서는 입력과 출력을 모두 전 압을 단위로 하는 4상한 벡터 내적 곱을 행하는 n-D 아날로그 선형 곱셈기를 사용하였다<sup>1,3,6)</sup>.

2×2×1 FANN회로는 2개의 입력을 가지므로 4개의 입력 패턴을 갖게 된다. 4개의 입력 패턴과 각각의

입력에 대한 목표 패턴(target pattern)을 입출력 쌍으로 하는 4개의 입출력 패턴.  $(x_{p1}, x_{p2}, t_p) = \{(0, 0, 0), (0, 1, 1), (1, 0, 1)(1, 1, 0)\}$ 에 대한 PSPICE 시뮬레이션을 수행하였다.

표 1. 그림 1의 FANN 회로의 학습 패턴에 대한 PSPICE 모의 실험 결과

Table 1. PSPICE simulation results of FANN circuitry in Fig. 1.

$x_{p1}$	0.5	0.5	4.5	4.5				
$x_{p2}$	0.5	4.5	0.5	4.5				
$t_p$	1.0	4.5	4.5	1.0				
$y_p$	1.81	5.0	5.0	1.81				
$t_p - y_p$	-0.81	-0.5	-0.5	-0.81				
weights	init.	s.s	init.	s.s	init.	s.s	init.	s.s
$w_{11}$	-0.5	0.053	-0.5	0.02	-0.5	-0.019	-0.5	-0.052
$w_{12}$	0.6	0.053	0.6	-0.019	0.6	0.02	0.6	-0.052
$w_{21}$	-0.5	0.053	-0.5	0.02	-0.5	-0.019	-0.5	-0.052
$w_{22}$	0.5	0.053	0.5	-0.019	0.5	0.02	0.5	-0.052
$w_{11}$	0.7	0.276	0.7	0.168	0.7	0.168	0.7	0.276
$w_{12}$	0.5	0.276	0.5	0.168	0.5	0.168	0.5	0.276

그림 1의 PSPICE 시뮬레이션 결과는 표 1에 요약하였다<sup>[3,4]</sup>. 여기에서 논리 0의 값은 0볼트에 근접한 아날로그 값을, 논리 1의 값은 5볼트에 근접한 아날로그 값을 사용하였다. 즉, 첫번째 입출력 패턴( $x_{p1}, x_{p2}, t_p$ )은 아날로그 전압 값 (0.5 0.5 1.0)의 논리치 (0 0 0)와 초기 시냅스 가중치 ( $w_{11}, w_{12}, w_{21}, w_{22}, w_{11}, w_{12}$ )를 (-0.5 0.6 -0.5 0.5 0.7 0.5)볼트로 주었을 때의 PSPICE 모의 실험이다. 실제 출력은 1.81볼트로 목표 논리치인 논리 0의 결과로 학습에 성공하였으며, 학습 시냅스 가중치 (0.053, 0.053 0.053 0.053 0.276 0.276)의 수렴치와 목표 출력과 -0.81볼트의 오류 전압을 나타내었다.

표 1은 그림 1의 FANN 회로가 각각의 입출력 패턴에 대하여 성공적인 학습을 수행한 것을 보여준다. 이어진 PSPICE 시뮬레이션에서 각각의 입출력 패턴은 연속형 (sequential-type)으로 실시되었다. 즉, 하나의 입출력 패턴에 대한 학습을 하고 그 다음 패턴에 대한 학습을 실시했을 때, 바로 전의 입출력 패턴에 대한 정보는 더 이상 보유하지 않는 PSPICE 시뮬레이션 결과를 보여주었다. 즉 매 번의 학습 시 그 전의 학습 패턴은 모두 잊어버렸다. 이것은 하드웨어 구현에서

는 당연한 예측된 결과이다.

표 2. 그림 2의 FANN 회로의 학습 패턴에 대한 PSPICE 모의 실험 결과

Table 2. PSPICE simulation results of FANN circuitry in Fig. 2.

	Column #1	Column #2	Column #3	Column #4	Column #5
$x_{p1}$	4.5	1.0	1.0	4.5	4.5
$x_{p2}$	4.5	1.0	4.5	1.0	4.5
$t_p$	0.5	0.5	4.5	4.5	0.5
$y_p$	0.5002	0.4999	4.501	4.501	0.5002
$t_p - y_p$	-0.002	0.001	-0.001	0.001	-0.002
$T_{11}$	4.210E-04	-2.511E-04	-2.901E-04	3.817E-04	4.565E-04
$T_{12}$	4.210E-04	-2.511E-04	3.817E-04	-2.901E-04	4.565E-04
$T_{21}$	4.210E-04	-2.511E-04	-2.901E-04	3.817E-04	4.565E-04
$T_{22}$	4.210E-04	-2.511E-04	3.817E-04	-2.901E-04	4.565E-04
$\bar{T}_{11}$	3.310E-03	3.411E-03	3.410E-03	3.410E-03	3.410E-03
$\bar{T}_{12}$	3.310E-03	3.411E-03	3.410E-03	3.410E-03	3.410E-03
$\theta_1$	4.210E-04	3.304E-04	3.817E-04	3.817E-04	4.565E-04
$\theta_2$	4.210E-04	3.304E-04	3.817E-04	3.817E-04	4.565E-04
$\bar{\theta}_1$	-3.269E-03	2.739E-03	-2.723E-03	-2.723E-03	-3.771E-03
$w_{11}$	-1	-1	-1	-1	-1
$w_{12}$	-1	-1	-1	-1	-1
$w_{21}$	-1	-1	-1	-1	-1
$w_{22}$	-1	-1	-1	-1	-1
$w_{11}$	-0.1393	0.1369	-0.1663	-0.1663	-0.1663
$w_{12}$	-0.1393	0.1369	-0.1663	-0.1663	-0.1663
$\theta_{11}$	-1	-1	-1	-1	-1
$\theta_{12}$	-1	-1	-1	-1	-1
$\bar{\theta}_1$	-1	-1	-1	-1	-1

다음에는 그림 2의 회로를 PSPICE 회로 시뮬레이터를 사용하여 시뮬레이션 하였다<sup>[3,4]</sup>. 이것의 PSPICE 모의 실험 결과는 표 2에 요약되었다. 표 2(a)는 PSPICE transient 분석을 행한 결과인데, 초기치는 ( $T_{11}, T_{12}, T_{21}, T_{22}, \bar{T}_{11}, \bar{T}_{12}, \theta_1, \theta_2, \bar{\theta}_1$ ) = (0.5 -0.5 -0.5 0.5 0.5 0.5 -0.5 -0.5 0.5)로 주어졌다. 이 초기치를 사용한 PSPICE 모의 실험 결과가 Column #1에 보여지고 있으며, 수렴된 시냅스 가중치는 다음 학습 패턴의 초기치로 사용되고, 그것의 PSPICE 모의 실험 결과가 Column #2에 요약되었다. 이와 같은 절차가 계속되어 Column #3, #4, #5가 얻어졌다. 여기에서 학습 결과는 매우 작은 출력 오류를 보여 주고 있으며, Column #5와 #1은 같은 학습 패턴과 다른 초기치에 대하여 같은 학습 가중치로 수렴함을 보여준다.

표 3은 각각의 수렴된 가중치들을 사용한 PSPICE dc 분석 결과이다. 이 시뮬레이션 결과도 역시 연속형으로 행하여졌기 때문에 각각의 패턴에 대해서는 성

표 3. 표 2의 수렴된 가중치들을 사용한 PSPICE dc 분석 결과  
Table 3. PSPICE DC analysis using the converged synapse weights in Table 2.

			Column #1,5				Column #2				Column #3,4			
$x_{\rho 1}$	$x_{\rho 2}$	$t_p$	$y_{\rho 1}$	$y_{\rho 2}$	$y_p$	$t_p - \bar{y}$	$y_{\rho 1}$	$y_{\rho 2}$	$\bar{y}_p$	$t_p - \bar{y}$	$y_{\rho 1}$	$y_{\rho 2}$	$\bar{y}_p$	$t_p - \bar{y}$
1.0	1.0	0.5	5.0	5.0	0.0	0.5	5.0	5.0	0.4995	0.005	5.0	5.0	0.0	0.5
1.0	4.5	4.5	0.0	0.0	0.5002	3.998	0.0	0.0	0.0	4.5	0.0	0.0	4.501	-0.001
4.5	1.0	4.5	0.0	0.0	0.5002	3.998	0.0	0.0	0.0	4.5	0.0	0.0	4.501	-0.001
4.5	4.5	0.5	0.0	0.0	0.5002	-0.002	0.0	0.0	0.0	0.5	0.0	0.0	4.501	-0.001

공적으로 학습을 하였으나, 그 전 패턴에 대하여는 모두 잊어버리는 것을 보여주었다. 이와 같은 모의 실험은 여러 학습 패턴에 대하여 수행되었으며, 그것들의 결과는 표 2와 같은 결과를 나타내었다(더 많은 모의 실험 결과는 참고 문헌<sup>[31]</sup>을 참고 바람).

### V. 결론

본 논문에서는 학습 회로를 내장한 점진형 신경회로망(FANN)을 구현하기 위한 수정된 오차역전파(MEBP) 학습규칙을 소개하였고, 그것을 이용한 FANN을 아날로그 CMOS 기술에 의한 n-D (n 차원) 내적곱 선형 곱셈기를 사용하여 구현하였다. 구현된 FANN은 많은 아날로그 곱셈기, 커패시터, 시그모이드 함수 회로 등을 포함한 복잡한 회로로 구성된 학습 회로부를 내장한 점진형 신경회로망(FANN)이다.

구현된 FANN회로의 동작을 점검하기 위하여 두 개의 2×2×1 프로토타입 FANN 회로를 구성하여 PSPICE 회로 모의 실험기로 모의 실험을 하였다. 모의 실험 결과 2×2×1 회로는 성공적인 학습을 수행하였다. 하지만 연속형 학습일 경우 바로 전에 학습된 사항은 더 이상 기억하지 못하는 결과를 보여주었으며, 이것은 하드웨어 상에서 당연한 예측 결과이다. 이 문제는 동시형 학습 방법을 적용함으로써 해결될 수 있다<sup>[31]</sup>.

### 참고 문헌

[1] F. M. A. Salam, "A Modified Learning

Rule for FANNs for Analog Implementation", Memorandum No. MSU/EE/S 90/02, Dept. of EE, Michigan State University, E. Lansing, MI, 26 Jan., 1990.

[2] D. E. Rumelhart et. al, "Parallel Distributed Processing - Explorations in the Microstructure of cognition", Vol. 1, MIT Press, 1986.

[3] M. R. Choi, "Implementation of FANNs with learning using Standard CMOS Technology", Ph. D. Thesis, Michigan State University, 1991.

[4] M. R. Choi, F. M. A. Salam, "Implementation of FANNs with Learning using standard CMOS Technology", IEEE ISCAS, pp. 1509-1512, 1991.

[5] M. R. Choi, "A Simple Nonlinear synapse circuit for ANNs", 2nd Int. Conf. on Fuzzy Logic and Neural Nets, Japan, pp. 505-508, 1992.

[6] 최명렬, "신경 회로망의 시냅스 구현을 위한 아날로그 벡터 곱셈기", 한양대학교 생산공학 논문집, Vol. 2, No. 1, 1993년 9월, pp. 13-21

[7] 최명렬, "신경회로망 구현을 위한 간단한 비선형 시냅스 회로", 한양대학교 산업과학 논문집, 1994년

---

 저 자 소 개
 

---



崔明烈(正會員)

1960年 9月 26日生. 1983년 2월 한양대학교 전자공학과(학사). 1985년 12월 미시간 주립대학교 전기공학과(M.S). 1991년 3월 미시간 주립대학교 전기공학과(B.S). 1991년 3월 ~ 1992년

8월 생산기술연구원, 전자정보실용화센터 조교수 겸 선임연구원. 1992년 9월 ~ 현재 한양대학교 제어계측공학과 조교수. 주관심분야는 Neural Network, VLSI Design, DSP H/W 응용, Smart IC Card, Intelligent Transportation System