論文96-33B-2-17

대규모 신경망 시뮬레이션을 위한 칩상 학습가능한 단일칩 다중 프로세서의 구현

(Design of a Single-chip Multiprocessor with On-chip Learning for Large Scale Neural Network Simulation)

金 鐘 文 *, 宋 玧 宣 *, 金 明 源 **

(Jong-moon Kim, Yoonseon Song, and Myung won Kim)

요 약

본 논문은 디지탈 VLSI 방법을 이용하여 구현한 대규모 신경망 모델을 시뮬레이션할 수 있는 신경칩과 이를 사용하여 제작한 신경망 컴퓨터의 설계에 대하여 기술한다. 신경칩은 동일한 네개의 디지탈 뉴럴 프로세서(digital neural processor: DNP-II)를 가지는 단일칩 다중 프로세서(single-chip multiprocessor)이며, 각 DNP-II는 독립된 프로그램 메모리와 데이타 메모리를 가지고 있어서 칩은 MIMD(multiple-instruction, multiple-data) 방식으로 동작한다. DNP-II는 신경망 모델을 고속으로 시뮬레이션할 수 있고 DNP-II상에서 학습을 할 수 있는 명령어를 가지며, 이를 지원하는 연산기, 메모리, 통신, 제어회로의 하드웨어 구조를 가진다. DNP-II는 on-line상태에서 1.28×109 (비트/초)의 속도로 네방향 통신을 하며, 이를 이용하여 확장성을 가지는 대규모 병렬 신경망 컴퓨터에 프로세서로 사용한다. 신경망 컴퓨터는 주 컴퓨터, 프로세서 보드, 그리고 연결 보드로 구성되어 있다. 신경망 컴퓨터는 프로세서 보드를 최대 16개를 가질 수 있으며, 16개의 프로세서 보드를 가지는 신경망 컴퓨터는 최대 약 40 GCPS(giga connection per second)의 성능을 보인다.

Abstract

In this paper we describe designing and implementng a digital neural chip and a parallel neural machine for simulating large scale neural netsorks. The chip is a single-chip multiprocessor which has four digital neural processors(DNP-II) of the same architecture. Each DNP-II has program memory and data memory, and the chip operates in MIMD (mlti-instruction, multi-data) parallel processor. The DNP-II has the instruction set tailored to neural computation, which can be sed to effectively simulate various neural network models including on-chip learning. The DNP-II facilitates four-way data-driven communication supporting the extensibility of parallel systems. The parallel neural machine consists of a host computer, processor boards, a buffer board and an interface board. Each processor board consists of 8×8 array of DNP-II(equivalently 2×2 neural chips). Various parallel structure can be built including linear array, 2-D mesh and 2-D torus. This flexibility supports efficiency of mapping from neural network models into parallel structure. The neural system accomplishes the performance of maximum 40 GCPS(giga connection per second) with 16 processor boards.

*準會員, 韓國電子通信硏究所 基礎基術硏究部
(Research Department, ETRI)
**正會員, 崇實大學校 컴퓨터學部
(Computer Department, Soongsil Univ.)

接受日字: 1995年4月12日, 수정완료일: 1996年1月17日

Ⅰ. 서 론

신경망 모델이 효율적으로 해결할 수 있는 문자인식, 영상처리, 음성인식 등의 문제들은 방대한 데이타의 처리가 필요하므로 범용 컴퓨터에서 시뮬레이션할 경우 많은 비용과 시간이 요구된다. 그래서 이러한 문제들을 빠르게 시뮬레이션하기 위하여 신경망 전용 시스템을 구현하고 있다. 신경망 시스템은 전기 또는 광학적 방법으로 구현되고 있으나 현재의 기술로 실제 문제들에 적용이 가능한 구현 방법은 전기적 방법을 사용하는 것이다. 전기적으로 구현되는 방법은 다시 디지탈와 아날로그로 분류할 수 있다. 아날로그 방법은 값들이 연속적으로 표현되어서 신경망이 가지는 자연 현상을 더욱 잘 모방할 수 있는 방법으로 속도와 집적도가 좋다. 그러나 현재의 기술수준의 한계로 외부 환경에 민감하고, 가변하는 가중치 값의 저장이 어렵고, 데이타 표현의 정밀도가 떨어져서 학습이 어렵다. 그래서 우리는 집적도와 속도는 떨어지나 신경망의 특징인 학습을 할 수 있는 디지탈 방법을 이용하여 구현한다[1]. 디지탈 방법을 이용하여 구현되는 신경망 시스템의 주 프로세서로는 상용되는 transputer, DSP (Digital Signal Processor) 칩 등을 사용할 수 있으나 우리는 신경망에 적합한 구조를 가지는 전용 신경칩을 설계하였다[2,3,4]. 다른 기관에서 신경망 전용으로 구현한 칩은 중앙 제어방법을 사용하는 SIMD(single-instruction, multiple-data) 구조를 가지는 칩으로 X-11, Lneuro, Neuro-Microprocessor, MA16, 등이 있다[5,6,7,8]. 이들은 신경망에 필요한 기능을 구현하여 다양한 신경망 모델을 빠르게 시뮬레이션할 수 있으며, 높은 성능을 가지는 신경망 시스템을 만들 수 있다. X-11, Lneuro, MA16 등은 다중 프로세서 칩으로 전체 시스템이 하나의 제어회로에 의해서 제어되며, 독립된 가중치용 메모리를 가지는 구조를 가진다. 그리고 Neuro-Microprocessor는 하나의 칩에서 동신에 많은 데이타를 처리할 수 있는 여러개의 연산기를 가지는 구조이며, 모든 데이타는 외부에서 오는 구조를 가진다.

이와 같이 신경망 시스템은 현재의 VLSI 기술수준의 제한으로 단일 프로세서를 사용하지 않고 병렬 구조로 구현된다. 그래서 프로세서의 개수에 비례하여 성능이 향상되는 확장성을 가지는 병렬 시스템을 구현하여 디지탈 시스템이 가지는 집적도와 속도의 단점을 보완한다.

병렬 시스템은 특성에 따라서 SIMD와 MIMD(multiple-instruction, multiple-data)로 구분할 수 있다. SIMD 방식의 병렬 시스템은 제어회로가 칩 또는 전체 시스템에 하나만 있으면 되므로 보다 많은 프로세서를 하나의 칩 또는 시스템에 집적할 수 있고 제어회로가 전체 시스템에 하나만 있어서 시스템을 제어하기 쉽다. 그러나 시스템이 동기되어서 동작을 하여야 하므로 클럭 스큐(skew) 현상이 문제가 되는 대규모 시스템으로의 구성이 어렵고, 제어회로가 집중되어 있으므로 유연성(flexibility)과 범용성이 떨어진다. X-11, Lneuro, MA16 등을 이용하여 구성한 시스템은 SIMD 방식의 병렬 시스템이며, 현재 구현되는 많은 신경망 시스템은 SIMD방법을 이용한다. 반면 MIMD 방식 병렬 시스템은 각 칩 또는 프로세서가 독립적으로 동작하므로 원하는 모든 프로세서를 프로그램할 수 있어서 문제의 특성에 따른 프로그램 설계에 유연성을 가지며, 비동기적으로 시스템을 구현할 수 있어서 SIMD 방식보다 대규모로 시스템을 만들 수 있다. 단점은 제어회로가 모든 프로세서에 들어가야 하므로 칩의 집적도가 떨어지며, 모든 칩 또는 프로세서에 각각 프로그램을 올려야 하는 어려운 점이 있다. Neuro-Microprocessor가 MIMD와 유사한 구조를 가지나 하나의 프로세서에서 여러개의 데이타를 동시에 처리할 수 있는 연산기 구조를 가지므로 문제의 특성에 따라서 성능이 변할 수 있다.

우리는 기존의 컴퓨터에서 시뮬레이션하기 어려운 대규모 신경망 모델과 다양한 신경망 모델을 빠르게 시뮬레이션할 수 있는 범용성과 확장성을 가지는 신경망 컴퓨터를 만든다. 그래서 클락스큐의 문제로 대규모 시스템으로의 구현이 어려운 SIMD 방식을 이용하지 않고 MIMD 방식을 이용한다. 그리고 프로세서의 갯수에 비례하여 성능이 향상되는 확장성을 가지도록 프로세서을 설계하여 SIMD보다 떨어지는 집적도의 단점을 보완하고, 프로그램과 제어의 어려운 점은 신경망 컴퓨터의 주변 하드웨어구성과 신경망 모델이 가지는 단순하고 규칙적인 계산을 이용하여 해결한다[9].

본 논문의 2 장에서는 신경칩의 구조와, DNP-II 명령어를 이용하여 작성한 오류역전파 모델과 비선형 함수의 프로그램을 보인다. 3 장에서는 제안하는 신경망 컴퓨터의 구조를 보이고, 성능을 분석하여 DNP-II가 확장성을 가짐을 보인다. 그리고 4 장에서는 결론과

앞으로 연구하여야 할 일들을 소개한다.

Ⅱ. 디지탈 신경칩의 구현

1. 신경칩의 구조

그림 1은 세개의 블럭으로 구성되어 있는 신경칩의 구조를 보인다. 세개의 블럭은 DNP-II, 프로세서간의 통신을 위한 포트(in/out port; IOP) 그리고 전체 칩을 제어하는 칩제어기(chip control block; CCB)이다. DNP-II는 독립된 데이타 메모리와 프로그램 메모리를 가지고 있는 어레이 프로세서(array processor)이다. DNP-II를 네개 가지고 있는 신경칩은 MIMD (multi-instruction, multi-data) 방식으로 동작하는 단일칩 다중 프로세서이다. 신경칩이 사용하는 데이타와 프로그램들은 외부와 연결된 전체 버스(global bus)인 G-Bus를 통해서 오며, on-line 상태에서 동작하는 신경칩들 간의 통신은 IOP를 통하여 지역 버스(local bus)로 사용되는 C1-Bus와 C2-Bus를 이용하여 이루어 진다. CCB는 off-line 상태에서 사용되며 신경칩이 외부로부터 프로그램을 받는 기능과 신경칩과 외부 시스템 간의 데이타 교환을 하는 역활을 한다.

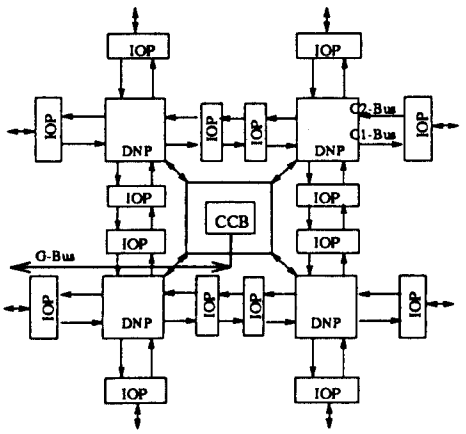


그림 1. 신경칩의 구성도
Fig. 1. Neural chip structure.

2. DNP-II의 구조

DNP-II는 신경망 계산에 적합한 명령어 및 구조를 가지는 범용의 16 비트 RISC(reduced instruction set computer)형 마이크로 프로세서와 유사하다. 특히 오류역전파 모델의 연산을 기본으로 명령어를 구성하고, 명령어를 빠르게 수행하기 위하여 RISC 프로세서에서 필요한 기능중 경제성을 고려하여 신경칩의 하드웨어에 구현하였다. 구현된 기능은 연산 파이프라인 기능, 명령어 파이프라인기능, 신경망 지향적인 명령어, 명령어의 반복적인 수행 등이 있다. 그림 2는 연산기, 메모리, 레지스터 파일, 그리고 제어 블럭으로 구성되어 있는 DNP-II의 구조를 보인다. 모든 데이타는 2의 보수형태로 표시되며, 크기는 학습을 할 수 있는 16비트이다[10]. 연산기 블럭은 16 비트 병렬-병렬 곱셈기와 16 비트 병렬 덧셈/뺄셈기, 누산기, 그리고 신경망에 필요한 논리 회로기로 구성되어 있다.

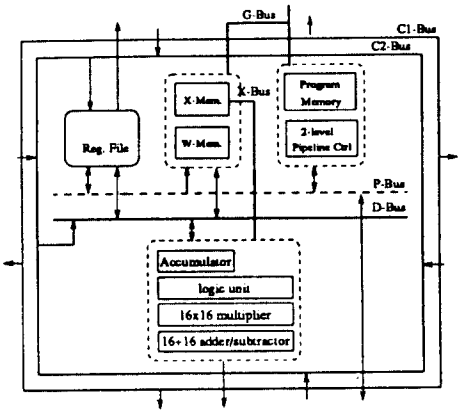


그림 2. DNP-II의 구조
Fig. 2. DNP-II architecture.

연산기는 곱셈기-덧셈/뺄셈기가 직렬로 연결되어서 3-단계의 연산 파이프라인 동작을 한다. 연산 파이프라인 동작을 이용하여 하나의 시스템 사이클(system cycle)에 하나의 연산(가중치와 입력의 곱에 합의 연산)을 수행한다. 연산 파이프라인 동작을 지원하기 위해서는 계산에 사용되는 데이타의 원활한 공급이 있어야 하며, 두개의 메모리와 독립된 D-bus와 X-bus가 이를 지원한다. 연산이 수행된 후에 덧셈/뺄셈의 결과는 하위값은 떨어지고 상위 16 비트가 선택되며, 곱셈의 결과는 32 비트중 하위 15 비트가 떨어진다. 그리고 계산중에 오버플로우(overflow)가 발생하면 오퍼랜드의 값에 따라서 최대값 또는 최소값으로 결과를 보낸다.

메모리는 526×16 비트의 WM(W-Memory)와 128×16 비트의 XM(X-Memory)가 있으며, 이들의 값은 각각 독립된 D-bus와 X-bus를 통하여 출력된다. WM은 신경망 모델에서 가중치를 가지고 있으며, XM

은 입력값을 가지고 있다. 가중치 외에 입력 데이타를 XM에 올려놓고 반복 사용하여 병렬 시스템에서 문제가 되는 통신 시간을 줄이고 칩상 학습을 빠르게 한다. 두개의 메모리는 독립된 주소 발생 장치와 독립된 여러개의 주소지정용 레지스터를 가지고 있다.

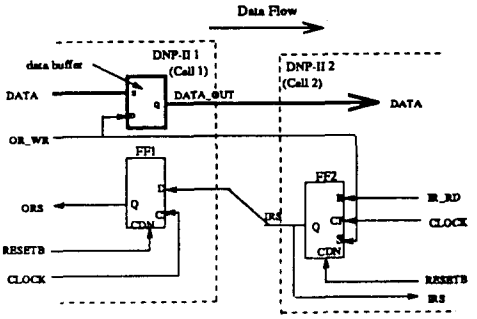
신경망 계산이 분산처리되면 하나의 프로세서가 모든 입력값을 가지지 못하여(입력값을 가지는 메모리의 크기가 제한됨) 하나의 뉴런 출력을 위한 부분 결과들이 계산되고 결과를 다른 프로세서로 보내어서 하나의 뉴런 출력을 만든다. 그래서 계산용 데이타를 가리키는 주소지정용 레지스터와 계산 결과를 저장시키는 주소지정용 레지스터를 분리시키면 하나의 주소지정용 레지스터를 사용하여 필요할때마다 필요한 주소값을 가져와서 사용하는데 소비되는 시간을 줄이수 있으므로 동작속도를 높이는데 유용하다.

레지스터 파일에는 범용 레지스터와 특수 목적으로 사용하는 레지스터가 있다. 범용 레지스터에는 학습에 필요한 변수들을 임시로 저장하여서 메모리를 억세서 하는데 소비되는 시간과 주소지정 레지스터의 사용을 줄인다. 특수한 목적으로 사용하는 레지스터는 다음과 같다. DNP-II가 통신하려고 하는 방향을 가지고 있는 IOPR 레지스터, WM 메모리의 주소를 계산하기 위한 변위를 가지고 있는 ICR 레지스터, 그리고 명령어를 반복해서 수행하여야 할 횟 수를 가지고 있는 Rc 레지스터가 있다.

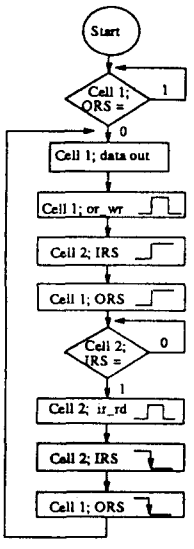
제어블럭은 2-단계 명령어 파이프라인 동작을 하도록 설계되어 있다. 대부분의 명령어는 2 시스템 클럭에 수행되도록 구성되어 있으므로 하나의 시스템 클럭에 하나의 명령어가 실행된다. 전체 명령어가 수행되는 평균 속도는 1.36 (명령어/시스템 클럭)이다.

DNP-II는 IOP를 이용하여 네방향으로 비동기 통신을 할 수 있으며 동시에 16비트의 데이타를 다른 DNP-II에서 받아들이고, 다른 DNP-II로 출력시킬 수 있어서 최대 32 (비트/시스템 클럭), 40MHz로 동작하면 1.28×10^{9} (비트/초)의 통신 대역폭(communication bandwidth)을 가진다. 그래서 네개의 DNP-II를 가지는 신경칩은 5.12×10^{9} (비트/초)의 용량을 가진다.

DNP-II의 버스는 명령어를 수행하기 쉽도록 구성되어 있다. 버스는 16 비트로 구성되어 있고, 그의 역할에 따라서 두가지 분류할 수 있다. 첫번째는 외부와 연결되는 버스로 C1-Bus, C2-Bus와 G-Bus가 있다. 두번째는 DNP-II 내부에 있는 P-Bus, D-Bus와 X-Bus가 있다. 외부와 연결되는 G-Bus는 DNP-II가 off-line 상태에서 외부로부터 데이타를 받는 버스이다. 지역 버스로 사용되는 C1-Bus와 C2-Bus는 다른 DNP-II와 통신을 할 수 있는 inter-processor 버스이다. DNP-II는 C1-Bus를 이용하여 데이타를 외부로 보내고 C2-bus를 이용하여 외부에서 데이타를 받아 들인다. DNP-II의 내부에서 데이타 이동을 위해서 사용하는 두개의 D-Bus와 X-Bus의 데이타 버스와 프로그램을 위한 P-Bus 버스와는 독립되어 있어서 동시에 프로그램 메모리와 데이타 메모리를 억세스할 수 있다.



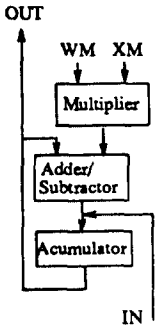
(a)



(b)

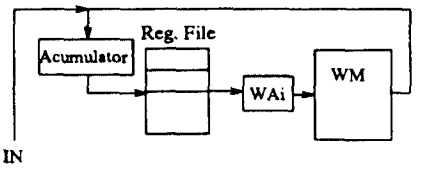
그림 3. 통신회로
(a) Data driven 통신 회로 (b) 동작 흐름도

Fig. 3. Communication circuit.
(a) Data driven communication circuit (b) Operation flow chart



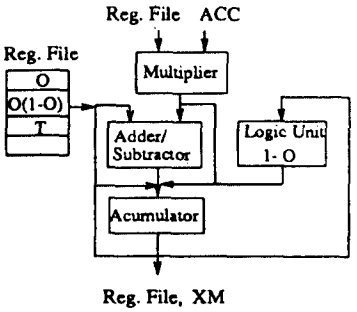

```
INA IOp          ; partial sum of XM*WM
MADD WM, XM, A ; partial sum of Σ XM*WM
OUTA IOp         ; output of partial value of ΣXM*WM
```

(a)



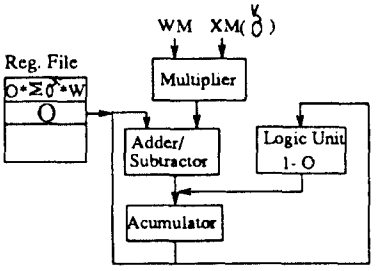

```
INA IOp
MOV A RFn
MOV RFn WAi
MOV WM, A
```

(b)




```
MOV RFn, A  ; reading out value(O)
NOT A       ; (1-O) -> ACC
MUL RFn, A  ; O * ACC -> ACC
MOV A RFn   ; writing to RF2
MOV RFn A ; reading target value(T)
SUB RFn A   ; (T-O) -> ACC
MUL RFn A   ; RF2 * ACC
OUTA IOP    ;
```

(c)




```
INX IOp XM      ; δ -> XM
INA IOp         ; partial sum of Σ xm*wm->ACC
MADD WM XM A ; Σ δ*WM -> ACC
MUL RFn A       ; O * ACC -> ACC
MOV A RFn       ; writing to RF1
MOV RFn A       ; reading O
NOT A           ; (1-O) -> ACC
MUL RFn A       ; ACC * RF1
```

(d)

그림 4. 오류 역전파 모델의 시뮬레이션과 프로그램
(a) 분류과정의 데이터 경로와 프로그램 (b) 비선형 함수의 데이터 경로와 프로그램
(c) 중간층 학습과정의 데이터 경로와 프로그램 (d) 출력층 학습과정의 데이터 경로와 프로그램

Fig. 4. Simulation and program of error back propagation.
(a) Data path for forward path, and its program (b) Data path for non-linear function, and its program (c) Data path for hidden layer learning, and its program (d) Data path for output layer learning, and its program

3. Data-Driven 통신

동시에 많은 데이타를 처리하는 것을 장점으로 하는 병렬 시스템에서 데이타를 교환하는 프로세서간에 on-line 통신은 병렬 시스템의 성능과 구조를 결정하는 중요한 기능이다. DNP-II는 on-line 상태에서 핸드쉐이킹(hand shacking)방법으로 네방향 통신을 한다. 그리고 전달될 데이타의 발생에 의해서 통신이 시작되는 data-driven으로 동작한다[11,12]. 시스템 전체는 비동기식으로 동작하나, 통신을 하는 두개의 DNP-II는 핸드쉐이킹으로 통신 프로토콜을 맞추고 동기되어서 통신을 한다. 2개의 시스템 사이클에 하나의 데이타를 보낼 수 있고, 명령어의 반복 수행을 이용하면 하나의 시스템 사이클에 하나의 데이타를 반복해서 전송할 수 있어서 확장성을 가지는 병렬 시스템에 사용할 수 있다. 그리고 네방향 통신을 할 수 있어서 네방향 통신 이하의 통신을 필요로하는 병렬 구조에 주 프로

세서로 사용할 수 있다.

통신은 데이타를 보내는 DNP-II가 주인(master)이 되고 데이타를 받는 DNP-II는 하인(slave)이 된다. 그림 3은 통신 포트의 구조와 동작을 보인다. 그림 3은 주인인 DNP-II 1이 하인인 DNP-II 2로 데이타를 보내는 단일 방향(uni-direction) 통신 동작을 보이며 ORS, IRS는 각각 출력 및 입력포트의 상태를 알리는 상태 플래그이다. 통신은 주인만이 상태 플래그를 확인하고 시작할 수 있다. DNP-II간에 통신되는 데이타는 데이타 버퍼에 저장되어 있는데 동작의 연속성을 위해서 여러개의 데이타 버퍼를 두어서 하인이 데이타를 받아가지 않아도 주인이 데이타 버퍼의 크기 만큼에 데이타를 계속적으로 보낼 수 있도록 할 수 있다[11]. 그러나 신경망 모델은 동일하고 규칙적인 연산이 대규모로 일어나므로 DNP- II의 연산 부담을 균등히 하면, 전체 동작의 타이밍을 벗어나서 일정 크기의 데이타 버퍼를 요구하는 동작은 발생하지 않으므로 여러개의 데이타 버퍼를 가져야하는 하드웨어 부담이 없다[13].

4. DNP-II의 프로그램

본 절은 신경칩이 신경망 모델에 적합한 명령어로 구성되어 있고 명령어에 적합하게 하드웨어가 설계된 것을 보이기 위해서 DNP-II의 명령어를 이용하여 오류역전파 모델을 프로그램하고 이들을 지원하는 하드웨어의 구성을 설명한다. 그림 4는 프로그램과 데이타 경로(data path)를 보인다. DNP-II의 하드웨어는 그림 4에서 보인 데이타 경로를 모두 만족시킨다. 프로그램은 여러개의 DNP-II에 신경망 모델이 분산처리되도록 구성되어 있다. 그림 4에서 A는 누산기에 저장된 값이며, WM은 W-Memory에 저장된 값이며, XM은 X-Memory에 저장된 값을 나타낸다. WM은 가중치값을 XM은 입력값을 가지고 있다. 식(1),(2),(3),(4)는 프로그램에 사용한 다층 오류역전파 모델을 보인다. 식 (1)은 전방향 경로(forward path), 식 (2),(3),(4)는 후방향 경로(backward path)와 관련된 연산식이다. W_{ji}은 가중치를, X_j은 입력값을, O_{pj}는 출력값을, t_{pj}는 원하는 출력값을, η는 학습율을 그리고 α는 모우멘트항을 표시한다.

$$O_{pi} = F(\sum_{j=1}^{n} W_{ji} \times X_j) \quad (1)$$

$$\delta_{pj} = (t_{pj} - O_{pj}) \times O_{pj} \times (1 - O_{pj}) \quad (2)$$

$$\delta_{pj} = (1 - O_{pj}) \times \sum_{k=1}^{m} \delta_{ik} \times W_{kj} \quad (3)$$

$$W_{ji}^{+} = \eta \times \delta_{pj} \times O_{pj} + \alpha \times W_{ji} \quad (4)$$

그림 4(a)는 식 (1)을 수행하는 데이타 경로(data path)와 프로그램을 보인다. INA IOp는 데이타를 외부에서 받고 누산기에 저장시킨다. OUTA IOp는 누산기의 값을 외부의 다른 프로세서로 보낸다. MADD WM, XM, A는 $WM \times XM + A$의 연산을 수행한다. INA IOp를 통해서 식 (1)의 j=1에서 n까지의 전체 연산 중 일부의 결과를 다른 DNP-II에서 전달 받고 ($\sum_{j=1}^{k} W_{ji} \times X_j (K < n)$), MADD WM, XM, A를 사용하여서 일부의 계산을 수행한 후에($\sum_{j=1}^{k} W_{ji} \times X_j (K < L < n)$), OUTA IOp를 사용하여서 결과를 다른 DNP-II로 전달한다. 몇개의 DNP-II를 이용하여서 이러한 동작을 수행하면 식 (1)의 계산에서 j=1에서 n까지의 연산이 수행된다. 그림 4(b)는 메모리를 이용하여 식 (1)의 비선형 함수를 구현하는 프로그램을 보인다. WM메모리에 비선형 함수의 테이블이 있어서 입력으로 들어온 값을 메모리의 주소 저장 레지스터인 WAi로 보내면, 그 주소에 들어있는 WM 값이 비선형 함수값이 되어 출력된다. 그림 4.(c)는 출력 노드의 가중치를 변화시키기 위해서 식 (2)의 δ_{pj} 값을 얻는 프로그램이다. 그림 4.(d)는 은닉층의 가중치를 변화시키기 위해서 식 (3)의 δ_{pj}값을 얻는 프로그램을 보인다. 이렇게 얻은δ_{pj}을 식 (4)에 넣어서 W_{ji}^{+}을 계산하며 학습을 시킨다.

그리고 DNP-II는 독립적으로 프로그램할 수 있는 MIDM 구조를 가지고 있어서 보여준 오류 역전파 모델외에도 다른 신경망 모델들을 시뮬레이션할 수 있다.

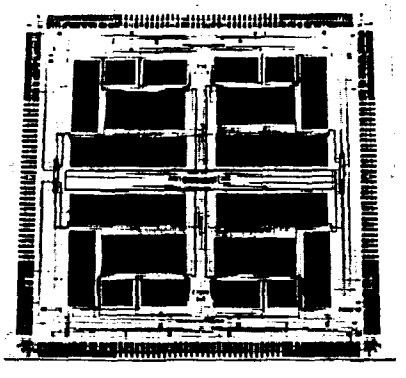


그림 5. 신경칩의 layout 사진
Fig. 5. Photocopy of neural chip layout.

5. 칩의 구현

그림 5은 제작된 칩의 layout 사진이다. 0.8μm

CMOS, cell-base 방법을 사용하여 제작하였다. 실리콘의 크기는 $11.5\times11.5mm^2$이며 네개의 동일한 DNP-II가 들어있는 구조를 보인다. 메모리를 제외한 전체 케이트수는 60000여개이다. 칩의 핀수는 299개이다. 동작 속도는 40 MHz이며, 최대 2 W의 전력을 소비한다.

Ⅲ. 신경망 컴퓨터

1. 신경망 컴퓨터의 구조

본 절은 신경칩을 이용하여 구성한 신경망 컴퓨터을 보인다. 신경망 컴퓨터는 주 컴퓨터(host computer), 프로세서 보드, 연결 보드(interface board), 그리고 버퍼 보드(buffer board)로 구성되어 있다. 그림 6은 제안하는 신경망 컴퓨터의 구성을 보이며, 각 보드의 기능은 다음과 같다.

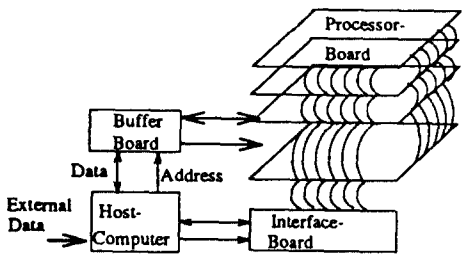


그림 6. 신경망 시스템의 구성
Fig. 6. The overview of neural system.

주 컴퓨터는 전체 시스템을 제어하는 목적으로 사용된다. Off-line 상태에서 프로세서보드로 프로그램과 데이타를 보내고 프로세서 보드로부터 데이타를 받을 수 있다. On-line 상태에서 프로세서보드와 데이타 교환을 하고 외부로부터 데이타를 받는다. 버퍼 보드는 주 컴퓨터와 프로세서 보드사이에 off-line 데이타 교환을 위하여 필요하다. 주 컴퓨터에서 받은 데이타를 프로세서 보드로 전달하는 중간 부분으로 프로세서 보드에 있는 많은 칩들로 연결된 입력(fan in)을 담당한다. 그리고 프로세서 보드에서 받은 데이타를 주 컴퓨터에 전달하는 역할도 수행한다. 연결 보드는 프로세서 보드와 주 컴퓨터사이의 on-line 데이타 교환을 수행한다. 그리고 프로세서 보드에서 필요로 하는 데이타를 가지고 있는 off-chip 메모리가 있다. 프로세서 보드는 신경 칩들이 장착된 보드이다. 하나의 프로세서 보드에는 16개의 신경칩이 장착되어 있으며, 신경망 컴퓨터에서 프로세서 보드의 개수를 변화시킬 수 있다. 최대 16개의 프로세서 보드를 가진다. 4개의 프로세서 보드와 이들간의 연결이 그림 7에 나와 있다. 다른 프로세서 보드와는 커넥터를 이용하여 연결하며 이러한 커넥터의 연결에 따라서 다른 병렬 구조를 가질 수 있다. 그림 7에서 A-C, B-D를 연결하면 토러스(torus)구조가 되며 A-B를 연결하면 우리가 다른 논문에서 제안한 PASF(processor array with switchable feedback) 구조가 된다[13]. 필요하다면 2번 보드에서 a1-a2, a6-a7, a3-a4을 연결하여 일차원 병렬구조를 만들 수 있다. 그래서 시뮬레이션하려고 하는 신경망 모델에 맞는 병렬 구조와 시스템 규격을 선택하여 사용할 수 있으며, 앞에서 보인 오류역전파 모델은 은닉층의 개수에 관계없이 2 차원 구조에서 효과적으로 시뮬레이션할 수 있다. DNP-II의 확장성을 가지는 네방향의 통신 포트가 이러한 병렬 구조들을 지원한다.

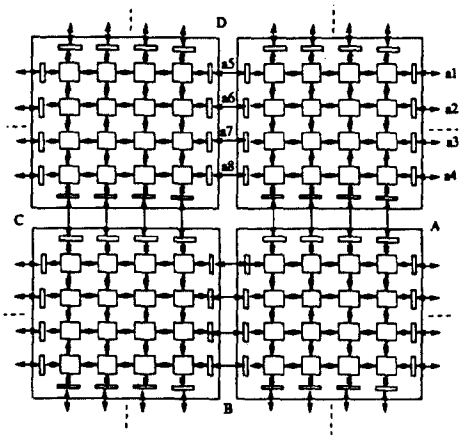


그림 7. 프로세서 보드의 연결구조
Fig. 7. The connection structure of processor board.

2. 신경망 컴퓨터의 성능

본 절에서는 신경망 시스템의 성능을 분석하여 최적의 신경망 시스템을 찾아보고, 이를 이용하여 신경칩이 확장성을 가지는 것을 보인다. 신경망 시스템의 성능을 분석하기 위한 프로세서 보드의 연결 구조는 토러스(torus)이며, 사용되는 신경망 모델은 다층 오류 역전파 모델(error back propagation model)이다. 뉴런의 개수와 DNP-II의 개수에 따라서 전방향 경로에 소비되는 시간(Tw)을 식으로 표시하면 다음과 같다[9]

$$T_w = \sum_{l=1}^{L-1}([\frac{N_{l+1}}{n}](t_{cp}[\frac{N_l}{n}] + t') + t_{cm}[\frac{n_l}{n}] + t'n) \quad (5)$$

여기서 L은 층수를, N_l은 L층의 뉴런 수를, n은 2차원 격자 구조에서 한변에 놓인 DNP-II의 개수를, t_{cm}은 데이타 하나를 통신하는데 필요한 시스템 클럭수를, t_{cp}은 비선형 함수를 계산하는데 필요한 시스템 클럭수를, 그리고 t'은 한개의 데이타에 곱의 합을 계산하는데 소비되는 시간이다. T_w의 단위는 시스템 클럭수이다.

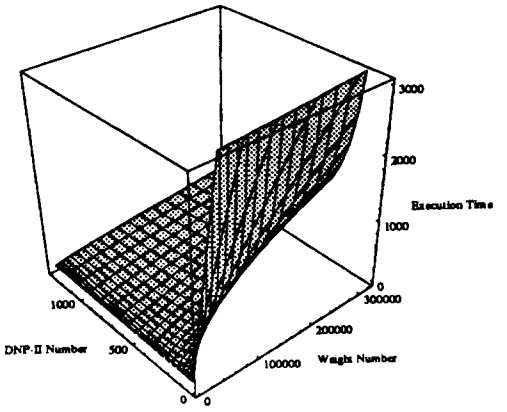


그림 8. 신경망 시스템의 수행시간
Fig. 8. The execution time of neural system.

그림 8는 위의 식을 신경망 모델의 크기와 DNP-II의 개수의 함수로 시뮬레이션한 결과이다. 사용하는 신경망 모델은 3층이며 모든 층은 같은 수의 뉴런을 가진다. X-축은 DNP-II의 개수이며 Y-축은 뉴런의 총 개수이고, Z-축은 실행시간으로 시스템 클럭수를 나타낸다. 그림 8를 이용하여 문제의 크기(가중치의 개수)가 정해지면 그에 맞는 시스템의 크기를 결정하여 사용할 수 있다. 그림 8에서 가중치의 갯 수를 고정하고 DNP-II의 갯 수를 변화시키면서 그린 그래프가 그림 9에 나와있다. 그림 9에서 실선은 문제가 큰 경우(총 135200개 가중치)로 프로세서의 개수가 증가함에 따라서 CPS가 선형적으로 증가하여 확장성이 있음을 보인다. 점선은 문제가 작은 경우(총 7200개 가중치)로 DNP-II의 개수를 증가시켜도 일정값 이상 CPS가 증가하지 않고 오히려 감소하는 것을 보인다. 그러나 프로세서의 개수가 작은 영역에서는 CPS가 선형적으로 증가하는 것을 보인다. 그래서 비용과 성능을 비교하여 문제에 맞는 최적의 시스템을 선택한다. 16개의 프로세서보드(1024개의 DNP-II)를 사용하는 신경망 컴퓨터의 최대 성능은 신경망 모델이 프로세서의 갯 수에 비교하여 매우 큰 경우로 $[\frac{N_{l+1}}{n}][\frac{N_l}{n}] \gg (\frac{N_{l+1}}{n})$가 되어서 식 (5)는 다음과 같이 근사화 된다.

$$T_w = \sum_{l=1}^{L-1}([\frac{N_{l+1}}{n}](t_{cp}[\frac{N_l}{n}]) \quad (6)$$

그래서 16개의 프로세서보드(1024개의 DNP-II)를 가지는 신경망 컴퓨터가 40 MHz로 동작하면 최대 약 40 GCPS의 성능을 보인다.

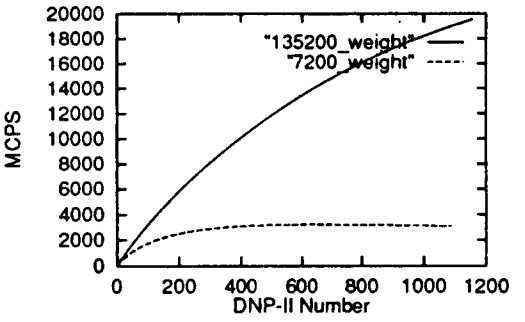


그림 9. 신경망 시스템의 성능
Fig. 9. The performance of neural system.

Ⅳ. 결 론

본 논문은 신경칩과 신경칩을 사용하여 구현한 병렬 신경망 컴퓨터에 대하여 기술하였다. 신경칩은 네개의 신경망 프로세서(DNP-II)가 장착된 MIMD형 단일칩 다중프로세서이다. 신경칩은 대규모 병렬 시스템에서 주 프로세서로 사용할 수 있으며, 대규모 병렬 시스템에서 문제가 되는 프로세서간의 통신을 해결하기 위하여 1 개의 시스템 사이클에 1 개의 데이타를 전송할 수 있는 통신 구조와 통신 횟수를 줄이기 위한 칩상 학습 가능한 구조를 가진다. 그리고 동작 주파수를 높이기 위해서 하나의 시스템 사이클에 데이타가 흘러가야할 경로를 짧게되도록 명령어를 설계하였으며, 성능 향상을 위해서 하나의 시스템 사이클에 하나의 연산을 수행하도록 하는 연산 파이프라인 기능, 명령어 파이프라인 기능, 명령어 반복 수행 기능 등이 있다. 그리고 DNP-II의 data-driven 방식 네방향 통신을 이용하여 다양한 병렬 구조에 주 프로세서로 사용할 수 있다. 신경망 컴퓨터는 주 컴퓨터(host competr), 프로세서

보드, 연결 보드(interface board), 버퍼 보드(buffer board)로 구성되어 있다. 신경망 컴퓨터에 프로세서 보드의 갯 수는 신경칩의 확장성을 이용하여 성능의 감소없이 최대 16개까지 가질 수 있으며, 프로세서 보드들의 연결 구조를 변경시킬 수 있어서 신경망 모델의 특성에 맞는 시스템의 크기와 구조를 선택하여 사용한다. 병렬 시스템의 성능은 프로세서의 개수에 비례하여 증가하는 확장성이 있음을 보였으며, 16개 프로세서 보드를 가지는 신경망 컴퓨터는 최대 40 GCPS 정도의 성능을 보인다.

현재는 신경망 시스템을 이용한 문자인식과 음성인식 소프트웨어를 작성하고 있으며, 시스템과 소프트웨어를 결합한 신경망 시스템(E-MIND)를 제작하고 있다. 그리고 병렬 신경망 시스템에 신경망 모델을 쉽게 사상(mapping)할 수 있도록 도와주는 시스템 소프웨어의 작성과 신경망 모델의 병렬 시스템으로의 사상에 관해서 더욱 연구를 해야한다.

감사의 글

본 연구는 정보통신부의 출연금으로 수행중인 기초기술연구과제에서 이루어진 것으로 정보통신부에 감사드리며, 아울러 기초기술연구부장인 이일항 박사의 배려에도 감사드린다.

참 고 문 헌

[1] D.A. Orrey, D.J. Myers and J.M. Vincent, "A High Performance Digital Processor for Implementing Large Artificial Neural Networks", IEEE 1991 Custom Integrated Circuits Conference", pp. 16. 3. 1-16. 3. 4, 1991.

[2] Jacob M.J. Murre, "Transputers and Neural Networks: An Analysis of Implementation Constraints and Performance", IEEE Transactions on Neural Networks, Vol. 4, No. 2, pp. 284-292, Mar, 1993.

[3] K.S. Gugel, J.C. Principe and S. Venkumahanti, "Analysis of Parallel Architectures for Artificial Neural Processing", World Congress on Neural Networks, Portland, Oregon, Vol. IV, pp. 787-790, July, 1993.

[4] U. Ramacher, W. Raab, J. Anlauf, U. Hachmann, J. Beichter, N. Bruls, M. Webeling and E. Sicheneder, "Multiprocessor and Memory Architecture of the Neurocomputer SYNAPSE-1", World Congress on Neural Networks, Vol. IV, pp. 775-778, july, 1993.

[5] Hal McCartor, "Back Propagation Implementation on the Adaptive Solutions CNAPS Neurocomputer Chip", Advances in Neural Information Processing Systems 3, pp. 1028-1031, Morgan Kaufmann Publishers, Inc., 1991.

[6] Nicoals Maudit, Marc Duranton, and Jean Gobert, "Lneuro 1.0: A Piece of Hardware LEGO for Building Neural Network Systems", IEEE Transactions on Neural Networks, Vol. 3, No. 3, pp. 414-421, May, 1993.

[7] John Wawrzynek, Krste Asanovic, and Nelson Morgan, "The Design of Neuro-Microprocessor", IEEE Transactions on Neural Networks, Vol. 4, No. 3, pp. 394-399, May, 1993.

[8] J. Beichter, N. Bruls, U. Ramaccher, E. Sichenedeer, H. Klar, "A VLSI Array Processor For Neural Network Algorithms", Custom Integrated Circuits Conference, pp. 4.6.1-4.6.3, 1993.

[9] Myung Won Kim, Youngjik Lee, Chong Moon Kim and Yoonseon Song, "A Wavefront Array Processing Architecture for Real-Time Simulation of Large Scale Neural Networks", International Joint Conference on Neural Networks, Nagoya, JAPAN, Vol. 2, pp. 1959-1962, Oct, 1993.

[10] Jordan L. Holt and Jeng-Neng Hwang, "Finite Precision Error Analysis of Neural Network Hardware Implements", IEEE Transactions on Computers, Vol. 42, No. 3, pp. 281-290, March, 1993.

[11] Ulrich Schmidt and Knut Caesar, "Datawave: A Single-Chip Multiproces-

sor for Video Applications", IEEE MICRO, pp. 22, Jun, 1991.
[12] S.Y. Kung, "VLSI Array Processors", pp. 295-359, Prentice-Hall Int., 1988.
[13] Myung Won Kim, et.al, "E-MIND: an Implementation of a Digital Neurocomputer and Its Application to Handwritten Digit Recognition", International Joint Conference on Neural Networks, Vol. III, pp. 258-263, Nov. 1992.

저 자 소 개

金 鐘 文(準會員)

1988年 연세대학교 금속공학과 졸업. 1989年 ~ 현재 한국전자통신연구소 기초기술연구부 연구원. 주관심분야는 신경회로망 이론과 구현, DSP 칩설계 등임.

金 明 源(正會員)

1972年 서울대학교 공과대학 응용수학과 졸업. 1981年 Univ. of Massachusetts (Amherst). Computer Science(석사). 1986年 Univ. of Texas(Austin). Computer Science(박사). 1975年 ~ 1978年 한국과학기술연구소 연구원. 1985年 ~ 1987年 AT&T Bell Labs. 연구원. 1987年 ~ 1994年 한국전자통신연구소 기초기술연구부, 책임연구원. 1994年 ~ 현재 숭실대학교 컴퓨터학부 교수. 1992年-1993年 한국신경회로망연구회 회장. 1993年 ~ 1995年 정보과학회 뉴로컴퓨팅연구회 위원장을 역임. 1993年 ~ 현재 IEEE Neural Networks Council 한국 지부장. 주관심분야는 신경회로망, 퍼지시스템, 진화알고리즘, 패턴인식, 디지탈 신경회로망의 구현 등임.

宋 玧 宣(準會員)

1990年 한국과학기술대학 컴퓨터학과 졸업. 1992年 한국과학기술원 전산학과 석사 졸업. 1992年 ~ 현재 한국전자통신연구소 기초기술연구부 연구원. 주관심분야는 신경회로망, 패턴인식, 유전자알고리즘, 퍼지이론 등임.