

## Speed Control of DC Step Motor using Neural Networks

孫 峻 赫\* · 朴 省 昱\*\* · 徐 輔 煥\*\*\* · 李 相 喆§  
 (Jun-Hyug Son · Seong-Wook Park · Bo-Hyeok Seo · Sang-Chol Lee)

**Abstract** -The present DC Motor or STEP Motor have been used in electronic work and products. There are many papers that let those Motor improved more conveniently in controlling and measuring those than before. By controlling and measuring those, the convenience of users and the functions of products can be improved. In addition, the responding speed of whole system can be increased by improving it of controlling and measuring. Therefore it is necessary that we develop motor motion application. Because of this necessity, neural networks was used to improve the responding speed of controlling and measuring, and a new application was developed for the convenience of users. This paper showed that efficiency of controlling methodology by neural-network is superior to others in correctness and speed. We are intend to verify its practical usefulness though experimentations.

**Key Words** : DC motor, step motor, neural, speed control

### 1. 서 론

직류모터나 스텝모터는 전자제품이나 산업현장에서 많이 사용하고 있으며, 이런 모터들을 보다 편리하게 제어·계측하고, 응답속도를 향상시키기 위해 많은 연구가 있었다[1]. 그 결과 사용자 편의와 제품의 기능을 극대화할 수 있으며, 제어·계측의 응답 속도를 향상시킴으로서 전체 시스템의 진행속도를 향상시켰다. 그러나 더욱 개선된 모터 응답 속도의 향상과 사용이 편리하도록 하기 위해서는 모터의 구동 어플리케이션이 필요한 시점이다.

본 연구에서는 제어·계측의 응답속도 향상을 위한 제어 기법으로 신경망을 이용하였고, 사용자의 편의를 위해 새로운 어플리케이션을 개발하였다. 일반적인 제어 기법은 시스템을 수학적으로 모형화하고 특정 동작점에서 선형화된 선형 모형으로부터 제어를 설계한다. 선형 시불변 시스템에 대해서는 안정성을 위한 필요충분조건들이 지난 세기동안 연구되어져 왔으며 그러한 시스템을 위한 여러 방법들이 개발되었다. 그러나 제어대상이 고차의 미분방정식으로 표현되는데 따른 복잡성과 제어대상 모형화의 부정확성, 특히 제어대상의 비선형성으로 인해 시스템의 안정성을 판별하기 어려울 뿐만 아니라 최적화된 디자인 알고리즘을 찾을 수가 없어서 결국 만족할 만한 제어 성능을 나타내기에는 어려움

이 있다. 이러한 어려움을 극복하고자 신경망을 이용하여 데이터를 학습한 후 그 값을 제어 기법의 새로운 데이터로 사용하였다. 이러한 신경망 제어기는 제어대상의 출력이 원하는 값에 가까워지도록 학습하는 과정이 필요하고 일반적으로 제어대상 출력오차의 제곱합을 목적 함수로 하여 기울기 강하법을 이용하여 이 값을 줄이는 방향으로 학습이 이루어진다. 여기서 학습이란, 시스템의 목적에 맞도록 신경망의 유니트간의 결합하중을 조절하는 것이다.

그러나, 현실에 존재하는 일반적인 제어대상은 입·출력 시에 그 물리적 특성에 의해 처리할 수 있는 입·출력 값의 크기가 제한되어있기 마련이다. 만약 제어하고자 하는 제어대상의 입력이 어떤 범위 안의 값으로 제한될 때 문제점을 가지게 된다[2].

본 연구에서는 이러한 문제점을 없애기 위해 신경망 제어기법을 이용하여 기존의 처리 방식보다 정확하고 빠른 성능을 보여주었고, 실험을 통하여 그 타당성을 검증하고자 한다.

### 2. 직류 모터의 속도제어

직류(DC)모터는 고정자로 영구자석을 사용하고, 회전자(전기자)로 코일을 사용하여 구성한 것으로, 전기자에 흐르는 전류의 방향을 전환함으로써 자력의 반발, 흡인력으로 회전을 생성시키는 장치이다. 그림 1과 같이 고정된 자계속에 전기자인 코일을 놓고 그 코일에 브러시 즉, 정류자를 통해 직류를 흘린다.

그림 1의 왼쪽에서 전기자 a의 부분에 플레밍의 왼손법칙을 적용시키면 위 방향으로 힘이 작용한다. 마찬가지로 원리로 전기자 b의 부분에서는 아래쪽으로 힘이 작용하여 전기자는 회전하게 된다. 가운데 그림에서처럼 전기자가 수직

\* 正 會 員 : 慶北大 工科大学 電氣工學科 博士課程

\*\* 正 會 員 : 龜尾1大學 電氣科 副教授

\*\*\* 正 會 員 : 慶北大 工科大学 電氣電子工學科 教授·工博

§ 正 會 員 : 동서울대학 전기정보제어과 教授·工博  
 接受日字 : 2004年 1月 24日  
 最終完了 : 2004年 2月 14日

으로 되면 전류는 흐르지 않게 되지만 관성 때문에 전기자는 회전을 계속한다. 전기자가 반회전해서 오른쪽 그림의 위치에 오면 전기자 a, b부분이 왼쪽 그림과 반대로 된다. 그러나 정류자에 의해 전류가 반대방향으로 흐르기 때문에 전기자는 같은 힘을 받아 회전을 계속한다.

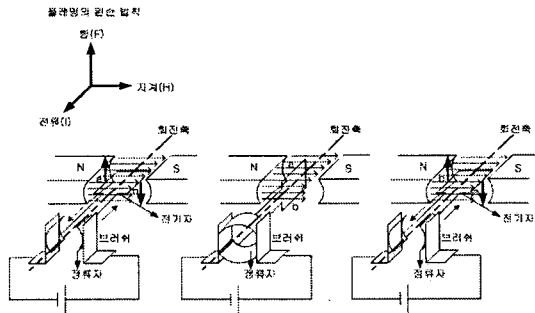


그림 1. 직류모터의 구조  
Fig. 1. Construction of DC Motor

직류모터의 특징은 기동 토크가 크고, 인가전압에 대하여 회전특성이 직선적으로 비례한다. 그리고 입력전류에 대하여 출력 토크가 직선적으로 비례하며 출력 효율이 양호한 것으로 특성은 그림 2와 같다.

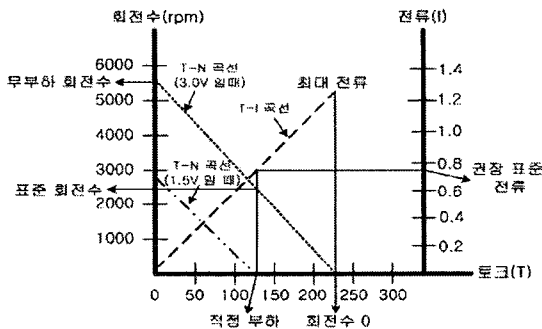


그림 2. 직류모터의 특성 곡선  
Fig. 2. Characteristic curve of DC motor

그림 2의 T-I(토크 대 전류) 특성은 인가 전류에 대해 분명하게 직선적으로 토크가 비례하고, T-N(토크 대 회전수) 특성은 토크에 대하여 회전수는 직선적으로 반비례한다. 그리고 인가전압에 대해서도 비례하며, 그림과 같이 평행하게 이동시킨 그래프로 된다. 이와 같은 2가지 특성은 서로 연동하고 있기 때문에 3가지 요소는 이 그래프에서 관계를 지을 수 있다. 즉, 이들 특성에서 알 수 있는 것은 회전수나 토크를 일정하게 하는 제어를 하려는 경우에는 전류를 제어하면 양자를 제어할 수 있다는 것을 나타내고 있다. 이것은 제어회로나 제어방식을 생각할 때, 매우 단순한 회로나 방식으로 할 수 있는 것이다.

그러나, 직류모터의 큰 결점으로는 구조상 브러시(brush)와 정류자(commutator)에 의한 기계적 접점이 있다는 점이다. 이것으로 인해 전류를 인가할 때 전기불꽃, 회전 소음, 수명 저하라는 형태로 나타난다. 또한 마이크로컴퓨터 제어를 하려는 경우에 노이즈가 발생하게 된다. 이 노이즈 대책을 위해서는 각 단자와 케이스 사이에 0.01[μF]~0.1[μF] 정도

의 세라믹 콘덴서를 직접 부착한다. 이것으로 정류자에서 발생하는 전기불꽃을 흡수하여 노이즈를 억제할 수 있다.

이 논문에서 모터의 속도를 검출하기 위해 인크리멘털 인코더를 이용했고, 여기서 검출된 속도와 PC상의 어플리케이션에 표시된 속도를 비교한다. 이 과정에서 속도 검출기와 속도 지령이 필요하다. 위치 결정 제어 시스템의 경우에 있어서는 속도 검출기로서 테코제너레이터가 광범위하게 사용되고 있으나, 위치 결정 제어를 실행하지 않는 경우 그림 3과 같은 인크리멘털 인코더를 이용하여 이 출력을 F/V 컨버터로 변환해서 속도 정보를 얻어낸다. 이 방식을 사용해서 직류 테코제너레이터와 거의 동일한 속도 정보(전압)를 얻기 위해서는 그림 4와 같이 F/V변환 회로, 회전 방향 판별 회로, 출력 전압 극성부가 회로 등이 필요하다. 이 논문에서는 1회전 100펄스의 인크리멘털 인코더를 사용하였다. 속도 지령은 외부 인가 전압에 의해 실행하는 방법과 회로 내부에 부착된 포텐셜-메터에 의해 실행하는 방법, 회로 내에 내장된 DA 컨버터를 이용해서 실행하는 방법의 3종류의 방식을 사용하였다.

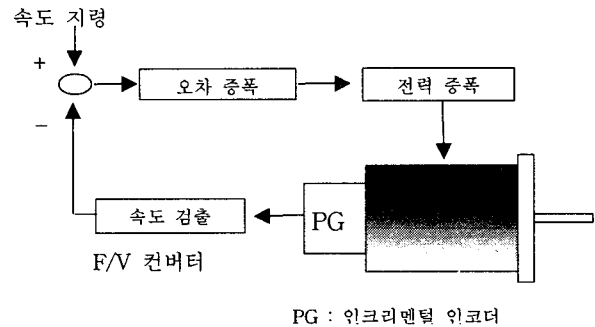


그림 3. 인크리멘털 인코더를 이용한 속도제어  
Fig. 3. Speed control using the incremental encoder

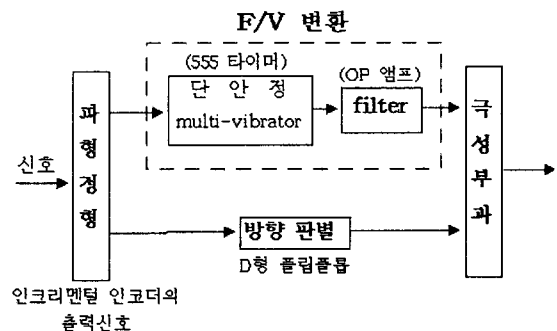


그림 4. 회전속도 검출기의 블록선도  
Fig. 4. Block diagram of rotation speed observer

그림 5 - 7은 직류 모터 속도제어 회로이다. 직류모터를 구동하기 위해 CPU에 내장된 펄스폭 변조기와 병렬 I/O 8255의 PC 상위 포트를 사용한다. 이 신호를 L298 선형 IC에 입력하여 제어 신호와 전류 증폭을 거쳐 직류 모터에 입력한다. 모터의 순방향 혹은 역방향으로 회전을 할 수 있으며, 속도 제어 회로에서는 회전속도를 검출하기 위해 포토커플러(Photo-Coupler)를 사용한다. 이 포토커플러에서 발생하는 신호를 고속 입력 장치 HSO0에 입력하여 소프트웨어적으로 회전 속도를 알아낼 수 있게 하였다[3].

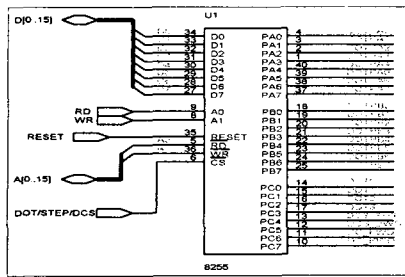


그림 5. 직류모터 속도 제어의 제어부  
Fig. 5. Control circuit of DC motor speed control

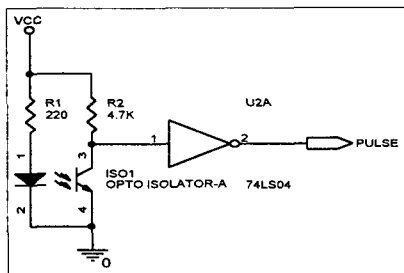


그림 6. 직류모터 속도 제어의 신호부  
Fig. 6. Display circuit of DC motor speed control

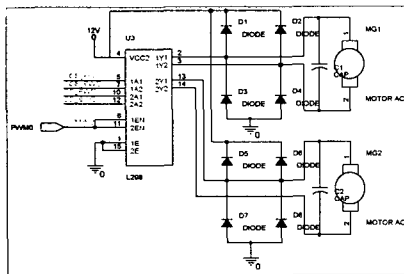


그림 7. 직류모터 속도 제어의 구동부  
Fig. 7. Drive circuit of DC motor speed control

### 3. 스텝 모터의 속도제어

스텝모터는 그림 8과 같이 코일 L1에 전류가 흐르면 내부에 회전하는 영구자석은 L1위치에 N극이 끌려오고, L1에 흐르는 전류는 차단하고 L2에 흐르게 하면 회전자가 L2의 위치로 이동하게 된다. 이때 이동하는 회전각은 코일의 극수에 의해 결정되며, 회전속도는 코일에 인가되는 펄스신호의 주기에 의해 결정되므로 전자회로만으로 초정밀 위치, 속도, 가속도 제어가 가능한 모터가 된다. 그리고 다른 종류의 모터와 달리 정지된 상태에서도 머물러 있도록 하는 정지토크를 가지고 있으며, 정확한 각도제어에 유리하다.

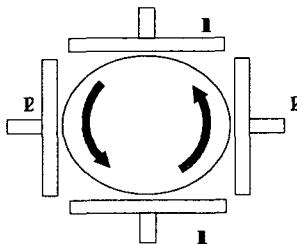


그림 8. 스텝 모터의 구조  
Fig. 8. Construction of STEP Motor

스텝모터는 모터의 회전각이 입력 펄스 수와 모터 속도가 1초간 입력 펄스 수에 비례하며, 1 step 당 각도오차가 +5[%]이내 이므로 회전각 오차가 누적이 없다는 특징이 있다. 스텝모터의 장점은 디지털 신호를 직접 open-loop 제어가 가능하므로 시스템 전체가 간단하고, 펄스 신호의 주파수에 비례하여 회전하므로 속도제어가 광범위하다. 그리고 구동, 정지, 정·역회전 변속이 용이하고, 회전각 검출을 위한 계환이 불필요하여 제어계가 간단하다. 그러나, 고속 운전시 탈조가 쉽고, 공진 현상이 발생할 수 있으며 펄스비가 상승함에 따라 토크가 저하되는 단점이 있다.

이 논문에서는 스텝모터를 80196KC가 내장된 고속 출력 장치를 이용하여 구동을 하므로 별다른 복잡한 펄스 발생기를 갖고 있지 않다. 고속 출력 장치에서 나오는 신호를 ULN2064로 전류를 증폭하여 스텝모터에 전달한다. 2상 방식의 스텝모터가 장착했다. 여기에 인가되는 고속 출력 신호를 LED를 사용하여 모니터 할 수 있도록 4개의 HSO가 동시에 연결되어 있다. 그림 9는 이 논문에서 사용된 스텝모터의 구동회로와 LED 모니터 회로이다.

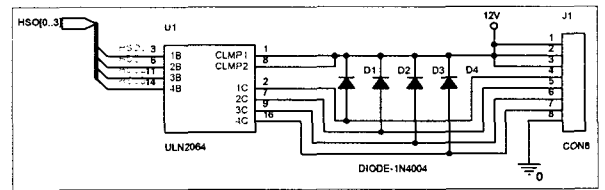


그림 9. 스텝 모터 구동 회로와 LED 모니터 회로  
Fig. 9. STEP motor circuit and LED monitor circuit

### 4. 신경망과 학습과정

이 논문에서 학습에 이용된 알고리즘은 널리 알려진 역전파법을 사용했다[4-7]. 일반적으로 다층의 신경망이 있다. 이 논문에서는 3층의 신경망을 역전파법으로 학습시키는 방법을 사용하였다.

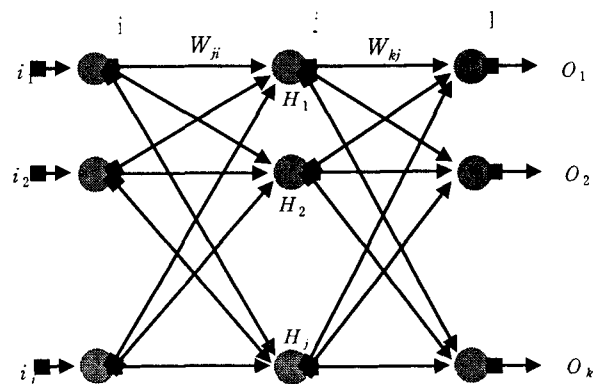


그림 10. 3층의 신경망  
Fig. 10. Neural network with three layers

이 논문에서는 역전파 신경망을 이용하여 학습하였다. 역전파 신경망은 가장 많이 알려져 있는 교사 학습 신경망 모형 중 하나로 일반적으로는 여러 개의 은닉층을 가지고 있지만, 출력층과 하나의 은닉층, 입력층으로 충분히 비선형 함수를 표현한다[8].

신경망 학습을 위한 목적함수는 식(1)과 같이 정의한다.

$$E = \frac{1}{2} \sum_{k=1}^K \{ \alpha(k) - d(k) \}^2 \quad (1)$$

여기서,  $\alpha(k)$ 는  $k$  스텝에서의 신경망 출력,  $d(k)$ 는  $k$  스텝에서의 기준신호이다. 이 신경망의 수학적인 모형은 다음 식과 같다.

$$\alpha(k) = \sum_j W_{jk}^i H_j(k) \quad (2)$$

$$H_j(k) = f_a(S_j(k)) \quad (3)$$

$$S_j(k) = \sum_i W_{ji}^i i_i(k) \quad (4)$$

여기서,  $W_{jk}^i$ 는 출력층과 은닉층의  $k$ 번째 뉴런간의 연결 강도,  $W_{ji}^i$ 는 은닉층의  $j$ 번째 뉴런과 입력층의  $i$ 번째 뉴런간의 연결 강도이다. 그리고, 활성화 함수는  $f_a(x) = \tanh(x) = \frac{1 - e^{-2x}}{1 + e^{-2x}}$  를 사용한다. 가중치의 학습은 경사 학습법에 의해 목적 함수를 감소시키는 방향으로 갱신된다.

$$\begin{aligned} W(k+1) &= W(k) + \eta \left( -\frac{\partial E}{\partial W} \right) \\ &= W(k) + \Delta W(k+1) \end{aligned} \quad (5)$$

여기서, 가중치에 대한 목적 함수의 변화량은 식(6)과 같다.

$$\frac{\partial E}{\partial W} = \frac{\partial E}{\partial \alpha(k)} \frac{\partial \alpha(k)}{\partial W} = e(k) \frac{\partial \alpha(k)}{\partial W} \quad (6)$$

여기서,  $e(k) = \alpha(k) - d(k)$ 이고, 각 가중치에 대한 신경망 출력의 변화량은 다음과 같다.

$$\frac{\partial \alpha(k)}{\partial W_{jk}^i} = H_j \quad (7)$$

$$\frac{\partial \alpha(k)}{\partial W_{ji}^i} = W_{jk}^i f_a(S_j(k)) i_i(k) \quad (8)$$

여기서,  $f_a(\cdot)$ 는 활성화 함수를 미분한 값이다.

이 논문에서 사용한 신경망 구조는 표 1과 같고, 신경망의 학습율은 0.035675, 학습횟수는 5000번이다.

표 1. 신경망 구조

Table 1. Construction of neural networks

종류	층 수	값
입력층	3	전압, 전류, error
은닉층	11	—
출력층	1	속도

## 5. 신경망을 이용한 직류 스텝모터의 속도제어

### 5.1 시스템 전체 구조

이 논문에서 사용된 직류 스텝 모터 장치와 PC를 연결하는 전체적인 구조도는 그림 11과 같다. PC와 직류 스텝 모터 장치의 MAIN BOARD를 시리얼 포트로 연결하여 RS232를 이용한 통신으로 명령을 내리고, 속도를 제어한다.  $\mu$ -processor로는 80196KC를 탑재하였으며, ROM과 RAM도 지원한다. MAIN BOARD와 직류 MOTOR ENCODER, STEP MOTOR ENCODER의 상호 전달에 의해 속도제어

명령을 수행하고, 그 결과 값을 각 MOTOR ENCODER의 값과 비교하여 오차를 비교한다.

PC의 애플리케이션에서 내린 명령은 MAIN BOARD를 통해 DC MOTOR DRIVE와 STEP MOTOR DRIVE에 전달되어 각 MOTOR를 구동하게 된다. PC의 애플리케이션 뿐만 아니라 KEY PAD로도 MOTOR 장치에 속도제어 명령을 내릴 수 있다. 그림 11은 시스템의 전체 구조도이다.

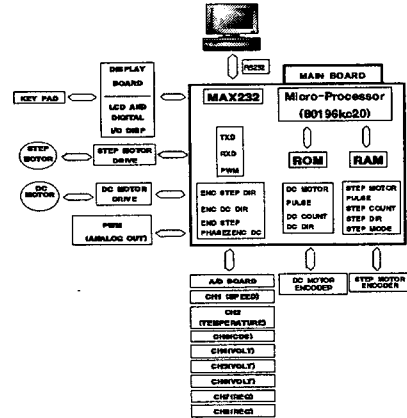


그림 11. 시스템 구조도

Fig. 11. System architecture

그림 12는 본 실험에 사용한 직류 스텝모터 장치이며, PC와 시리얼 통신을 통하여 사용자가 목적에 맞게 직류 모터와 스텝 모터를 구동시킨다. 직류 모터는 사용자가 원하는 각각의 파라미터 설정에 따라 직류 모터를 구동할 수 있고, 그에 따른 RPM Data에 의해 직류 모터 속도를 측정할 수 있다. 또한 스텝 모터 역시 파라미터 설정에 따라 구동할 수 있으며 각도와 회전수를 측정할 수 있다.

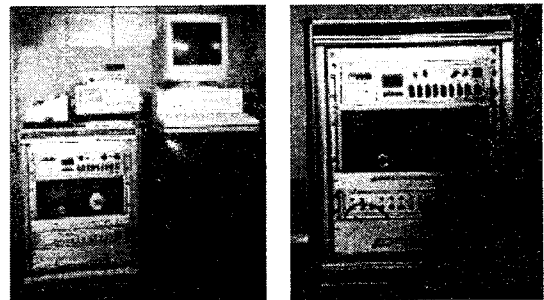


그림 12. 직류 스텝모터 장치

Fig. 12. DC STEP motor equipment

그림 13은 8bit 데이터를 전송함에 있어서 시리얼 방식을 나타낸 것이다. 시리얼 방식의 경우는 각 비트의 데이터를 인코딩(encoding)해서 특정한 주기에 맞추어 하나의 선으로 신호를 전송한다. 그러면 이 신호를 리시버(receiver)에서는 디코더(decoder)를 이용하여 필요한 한 byte의 데이터로 바꾸게 된다. 80196은 다른 장치간에 양방향 통신을 위해 트랜스미터(transmitter)와 리시버(receiver)가 필요하다.

이 논문에서 사용한 모터장치의 MAIN BOARD에 탑재된 80196 역시 양쪽 모두 트랜스미터(transmitter)와 리시버

(receiver)를 구축하여 양방향 통신이 되도록 만들었다. 그림 14는 이 논문에서 사용한 시리얼 통신의 회로도이고, 그림 15는 속도제어 시스템의 흐름도이다.

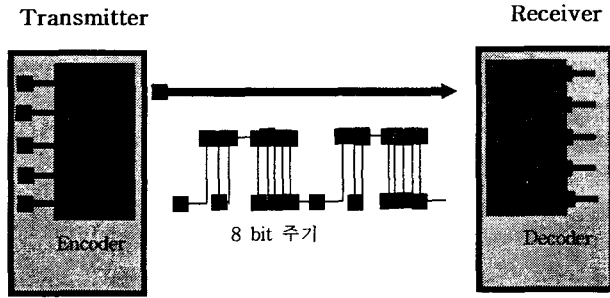


그림 13. 시리얼 통신  
Fig. 13. Serial communication

모터 메뉴에서 원하는 모터를 설정 후 동작시킨다. 그림 16은 직류 스텝 모터 어플리케이션이다.

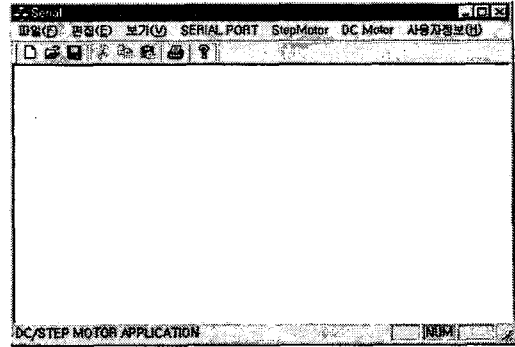


그림 16. 직류 스텝 모터 어플리케이션  
Fig. 16. DC STEP motor application

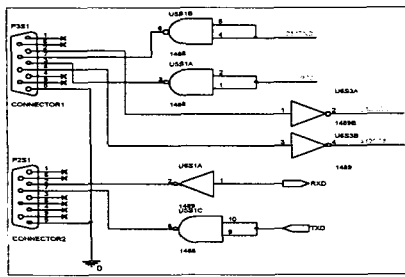


그림 14. 시리얼 통신 RS232 회로  
Fig. 14. Serial communication RS232 circuit

5.2.1 어플리케이션의 시리얼 포트 설정 메뉴

시리얼 포트 메뉴에는 포트 설정과 포트 초기화가 있으며, 포트 설정에서 COM 포트와 바운드를 설정한다. COM 포트 설정은 모터장치의 시리얼 포트와 PC의 COM 통신 포트 위치를 설정하는 하는 곳이다. 이 논문에서 사용된 어플리케이션은 COM1, COM2, COM3, COM4의 4개의 통신 포트까지 설정 가능하도록 되어있다. 바운드를 설정은 통신 바운드를 설정하는 곳으로 이 논문에서는 57600, 38400, 19200의 3가지 바운드를까지 설정 가능하도록 되어있다. 그림 17은 시리얼 포트 설정 메뉴이다.

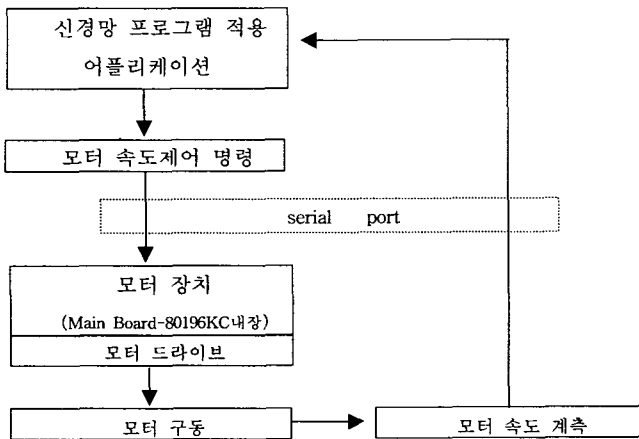


그림 15. 속도 제어 시스템의 흐름도  
Fig. 15. Flow chart of speed control system

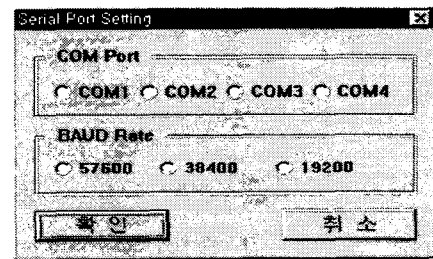


그림 17. 시리얼 포트 설정 메뉴  
Fig. 17. Serial port setting menu of application

5.2 PC에서의 어플리케이션

이 어플리케이션은 PC에서 직류·스텝 모터 장치의 구동 명령을 하는 곳이며, 사용자가 여러 가지 작업을 할 수 있게 한다. 작업의 기본 메뉴인 파일메뉴와 편집메뉴, 보기메뉴 등의 윈도우의 기본 메뉴들을 사용할 수 있다. 보기메뉴에서는 파일열기·파일저장·인쇄·잘라내기·붙여넣기 등의 기능이 있다.

사용자의 모터 제어를 위한 명령을 위해 우선 시리얼 포트 메뉴에서 포트 설정을 해야 하고, 스텝 모터 메뉴와 직류

5.2.2 어플리케이션의 스텝 모터 메뉴

스텝 모터 메뉴에서는 파라미터 설정을 이용하여 위치, 속도, 제어 모드, 회전 방향을 설정하여 명령을 지시하면 구동 결과가 어플리케이션의 Data View에 표시된다.

Parameter Setup에는 Position과 Velocity, 제어모드, 회전 방향을 설정하게 된다. Position 설정과 Velocity 설정은 0~100까지의 값으로 설정하며, 제어모드설정은 Full, Full+Half, Half의 모드로 설정할 게 지원한다. 그리고, 회전방향은 반시계방향(CCW)와 시계방향(CW)로 설정할 수 있어, 원하는 방향으로 회전설정을 하면 된다. Data View에는 모터가 구동된 결과로서 모터의 Pulse와 각도, 회전수가 결과 값으로 어플리케이션에 표시된다. 이 값과 실제 인크리멘탈 인코더에서의 값을 비교함으로써 결과 값의 오차정도를 확인한다. 그림 18은 스텝모터 메뉴이다.

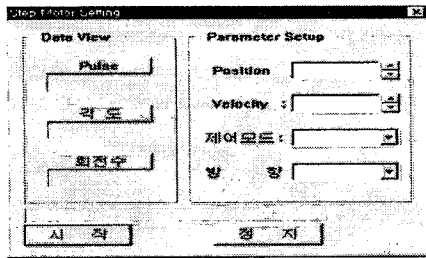


그림 18. 스텝모터 메뉴  
Fig. 18. STEP motor menu of application

5.2.3 어플리케이션의 직류 모터 메뉴

직류 모터 메뉴에서는 파라미터 설정을 이용하여 P(Proportional) gain과 I(Integral) gain, D(Derivative) gain, 정지 시간, 회전 방향, REF.(Reference) RPM(Recycle Per Minute)를 설정하여 명령을 지시하면 구동 결과가 어플리케이션에 나타난다.

이 논문에 사용된 어플리케이션의 변수설정에서는 모터에 인가할 P이득(kp), I이득(ki), D이득(kd)과 정지시간을 0~255의 값으로 설정할 수 있게 했으며, 회전 방향은 CCW와 CW로 설정할 수 있어, 원하는 방향으로 회전설정을 하면 된다. 그리고, 기준속도(REF. RPM)을 0~1000의 값으로 설정할 수 있게 하여 다양한 속도로 구동할 수 있게 하였다. 그림 19는 직류 모터 메뉴이다.

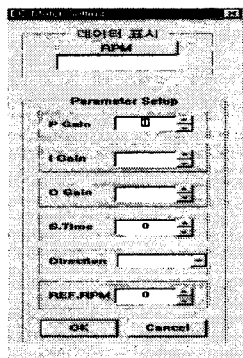


그림 19. 직류모터 메뉴  
Fig. 19. DC motor menu of application

6. 결과 및 검토

직류 스텝 모터 장치와 PC를 연결하고 속도 지령 명령을 내려, 속도를 실제 측정값과 이 논문에서 개발된 어플리케이션에서 측정된 값을 비교한다. 직류 모터의 속도 측정은 직류모터에 부착되어 있는 인크리멘탈 인코더와 어플리케이션에서 측정된 값을 표 2에 비교하였다. 레퍼런스 RPM를 설정하고, 인크리멘탈 인코더에서 측정된 RPM과 어플리케이션에서 측정된 RPM를 비교하면, 어플리케이션에서 측정된 값이 더 오차가 적다는 것을 알 수 있다. 스텝모터의 속도제어는 레퍼런스 값을 주고, 그 값에 얼마나 정확히 구동되었는가를 이 논문의 어플리케이션에 값으로 나타낸다. 레퍼런스 값과 어플리케이션에서 측정된 값이 오차가 적음을 표 3에 비교하였다. 이 때, 레퍼런스의 측정된 값은 스텝모터의 회전수로 하였다. 속도 지령의 응답 속도를 향상시킨 결과를 그림 20-22처럼 100rpm, 800rpm, 1000rpm으로 나누

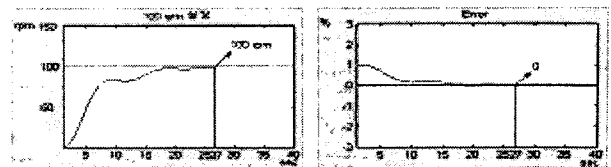
어 실험하여 비교하여 나타냈고, 표 4와 같이 신경망을 적용시킨 시스템에서의 응답속도가 향상됨을 알 수 있다. 여기서 사용된 신경망 구조에서 가중치는 56개이며, 표 5에서 각 rpm간의 학습된 개수의 절대값 편차를 비교했다.

표 2. 직류 모터의 RPM 비교  
Table 2. RPM comparison of DC motor

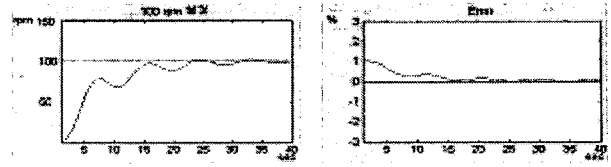
레퍼런스 RPM	인크리멘탈 인코더에서 측정된 RPM	어플리케이션에서 측정된 RPM
100	101.7	100.2
300	298.1	300.7
500	501.4	500.3
800	801.3	799.7
1000	998.4	999.8

표 3. 스텝 모터의 회전수 비교  
Table 3. Movement comparison of STEP motor

레퍼런스 값(회전수)	어플리케이션에서 측정된 회전수
10	10
40	40
70	69
100	99
150	148
250	249

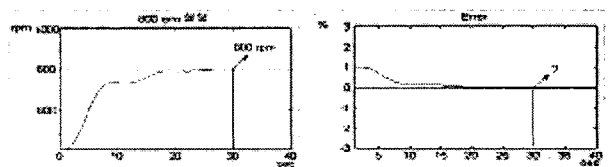


(a) 신경망 적용 제어기(neural network controller)

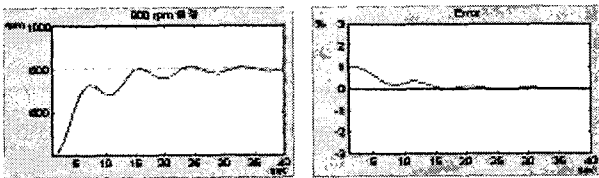


(b) 기존 PID 제어기 적용(general PID controller)

그림 20. 100 rpm 일 때 응답 속도 비교  
Fig. 20. Comparison of response speed (100 rpm)



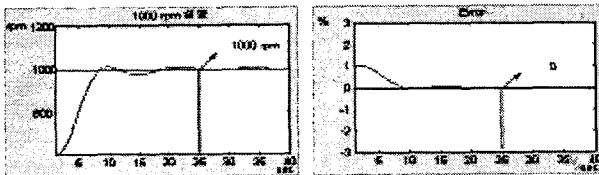
(a) 신경망 적용 제어기(neural network controller)



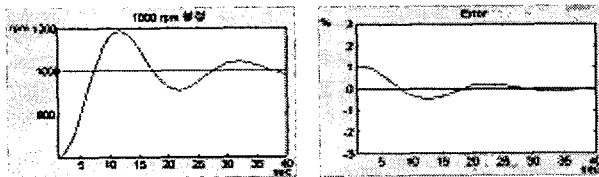
(b) 기존 PID 제어기 적용(general PID controller)

그림 21. 800 rpm 일 때 응답 속도 비교

Fig. 21. Comparison of response speed (800 rpm)



(a) 신경망 적용 제어기(neural network controller)



(b) 기존 PID 제어기 적용(general PID controller)

그림 22. 1000 rpm 일 때 응답 속도 비교

Fig. 22. Comparison of response speed (1000 rpm)

표 4. 신경망 적용 제어기와 기존 PID 제어기의 수렴시간 비교  
Table 4. Convergence comparison of neural network controller and general PID controller

rpm	신경망 적용 제어기	기존 PID 제어기
100	27 sec	40 sec 이상
800	30 sec	40 sec 이상
1000	25 sec	40 sec 이상

표 5. rpm간의 가중치 편차 비교

Table 5. Comparison weight deviation of rpm

각 rpm의 비교	각 rpm간의 학습된 개수의  편차
100 - 800	0.2
800 - 1000	0.3
100 - 1000	1

그림 23은 개개의 rpm 학습 구조에서 개선된 구조인 모든 rpm에 대해 학습하는 신경망 구조이다. 이 구조에서는 각각의 rpm 오차  $e_1, e_2, e_3$ 을 평균한 평균오차를 입력으로 한다. 그림 24는 각각의 rpm 오차  $e_1, e_2, e_3$ 를 나타낸 것인데 오차의 감소가 PID 제어기보다 향상되었다. 표 6에서는 기존 PID 제어기의 각 rpm(100 rpm, 300 rpm, 500 rpm, 800 rpm, 1000 rpm)에 대한 이득값과 신경망을 이용한 전체 rpm에 대한 PID 이득값, 신경망을 이용한 각 rpm에 대한 PID 이득값을 비교했다. 그리고, 각 rpm에 대해 한번에 학습한 결과를 recall함으로써 결과를 비교할 수 있으며, 표 7에서는 제어기의 각 rpm에 따른 오차를 비교했다.

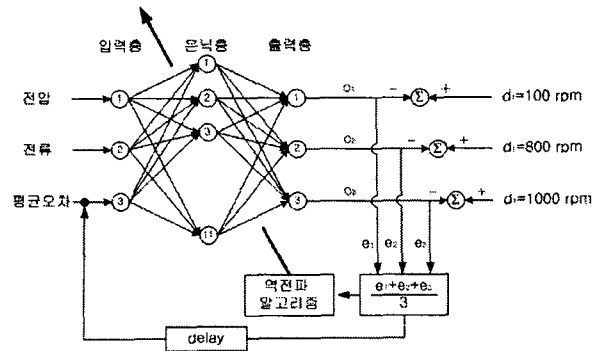
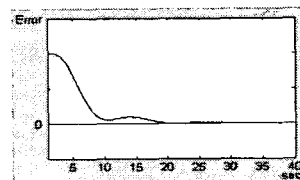
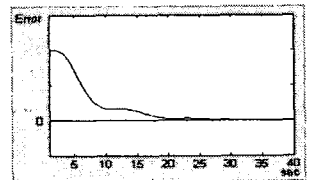


그림 23. 적용된 신경망의 구조

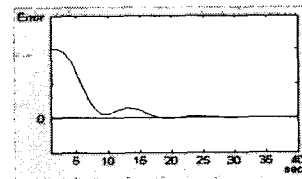
Fig. 23. Construction of application neural networks



(a) e1 의 오차(100 rpm)



(b) e2 의 오차(800 rpm)



(c) e3 의 오차(1000 rpm)

그림 24. 오차의 비교

Fig. 24. Comparison of error

표 6. 기존 PID 제어기와 신경망을 이용한 PID 제어기의 Kp, Ki, Kd 이득값 비교

Table 6. Comparison Kp, Ki, Kd gain of general PID controller and neural network PID controller

이득종류	기존 PID 제어기의 각 rpm에 대한 PID 이득값					신경망을 이용한 전체 rpm에 대한 PID 이득값	신경망을 이용한 제어기의 각 rpm에 대한 PID 이득값				
	100	300	500	800	1000		100	300	500	800	1000
Kp	3.567	5.367	4.364	7.128	6.346	6.346	2.345	4.253	4.035	6.194	5.629
Ki	29321	43565	12455	23893	8723	23450	18923	38472	9871	21056	7213
Kd	0.345	1.896	0.986	3.145	5.234	2.873	0.174	1.452	0.539	2.071	4.091

표 7. 각 rpm에서의 제어기 오차 비교  
Table 7. Comparison control error of several rpm

rpm	제어기 오차			
	PID	각각의 신경망 학습	전체 신경망 학습	신경망을 이용한 PID 이득수정
100	1.742	0.237	0.312	0.349
300	1.928	0.716	0.541	0.482
500	1.453	0.361	0.272	0.528
800	1.319	0.322	0.584	0.592
1000	1.693	0.242	0.416	0.598

이 논문에서는 직류·스텝 모터의 속도 지령을 PC상의 어플리케이션으로 함으로써 사용자의 편의를 제공하고 있으며, 각 모터의 속도를 다시 PC상의 어플리케이션에 표시되어 속도 계측의 측면에서도 역시 사용자의 편의를 고려하였다. 속도 지령 명령을 받아 그 값을 신경망에 의한 학습으로 오차를 감소하였으며, 통신 속도 역시 향상시키는 결과를 보여주었다. 그리하여, 모터 장치와 PC의 어플리케이션과의 통신 속도가 기존보다 향상되어 빠른 제어속도를 요하는 기계장치나 장비에서 사용한다. 제어기 설계시 신경망 제어기의 유무에 따라 시스템 사양에 적합한 이득값이 다소 차이가 발생하는 데, 이 차이로 인해 전체 시스템의 성능과 응답 속도의 저하를 가져온다. 또한 신경망을 이용하여 PID 제어기의 제어 사양 최적의 이득값을 추정함으로써 제어기뿐만 아니라 전체 시스템의 성능과 응답 속도를 향상시킬 수 있다. 그리고, 기존의 PID제어기의 구조에 신경망 제어기를 전방향 제어기로 사용하는 구조를 제안하여 제어기 제어 사양 최적의 이득값을 추정할 수 있다. 이와 유사한 제어기로서 Kawato[9]등이 제안한 제어기 구조가 있으나 제안된 구조에서는 플랜트의 역을 학습하는 형태가 아니고, 플랜트의 정방향 동력학을 학습한 후, 그 신경망의 역을 구하여 제어하는 것이다. PID알고리즘은 신경망의 학습오차를 보완해주는 역할을 하고, 신경망 PID제어기는 PID제어기의 성능을 학습에 의하여 꾸준히 향상시키는 역할을 한다[10]. 이런 신경망 PID제어기에서 추정된 최적의 이득값을 사례연구하여 그림 25-26와 표 8에 나타내었다. 결과에서 볼 수 있듯이 기존 PID 제어기보다 시스템의 제어 사양에 부합되는 최적의 이득값이 추정됨을 확인할 수 있었다[11-12]. 여기서, 전체 시스템 구조도에서 시스템 부분으로는 차수가 4차이고, 지연(delay)이 -0.2로 설정하여 실험했다. 그리고 시스템의 제어 사양으로 첨두치가 30%이하, 정정시간까지의 오차제곱면적과 제어입력의 제곱을 최소화하는 것을 목적으로 한다. 이때, ISE는 Integral of Square Error 이고, IE는 Input Error 이다. 이 논문의 실험에서 측정 오차 비교에서 볼 수 있듯이 기존의 속도 계측 장비보다 정확한 속도를 제어·계측할 수 있었다. 앞으로 모터의 속도 제어와 계측에 더욱 향상된 알고리즘과 제어 기법을 통하여 더욱 정확한 측정값과 응답 속도 향상에 대한 연구가 되어야 한다.

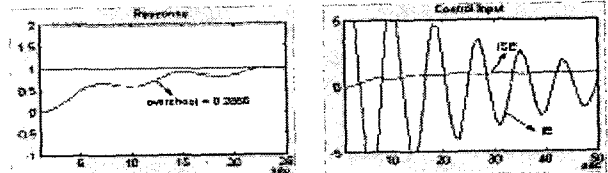


그림 26. PID 제어기의 첨두치와 제어입력  
Fig. 26. Overshoot and control input of PID controller

표 8. 신경망 PID 제어기와 기존 PID제어기의 비교  
Table 8. Comparison of neural networks PID controller and general PID controller

제어기	신경망	PID
이득, 첨두치, 목적함수		
PID 이득	kp=7.69 kd=0.6 ki=100000	kp=8.37 kd=0.83 ki=1300000
첨두치(overshoot)	0.0023	0.2856
오차제곱면적과 제어입력의 제곱	0.0482	0.0876

7. 결 론

이 논문에서는 신경망을 이용한 직류·스텝모터의 속도제어이다. 신경망을 이용하면 기존의 모터 제어기보다 정확한 속도제어가 가능하며 응답속도 역시 빠른 결과가 나타났다. 입력에 대한 출력이 빠르게 응답하기 위해서는 전기적인 측면보다 기계적인 측면에서 주로 문제가 된다. 모터와 그 축에 연결된 부분들은 관성에 직접적인 관계가 있으며 관성이 커지는 경우 응답속도가 늦어지며 정상상태 도달시간이 비례적으로 지연된다. 그리고, 시스템의 이득이 커지면 응답속도가 빨라지지만 오버슈트가 비례적으로 커지는 단점이 있으며, 오차신호 증폭기의 이득이 너무 작으면 응답속도가 느려지고 위치오차가 많이 난다. 시스템의 속도궤환이 너무 클 경우 역시 응답속도가 저하되고 특히 관성 부하에서는 궤환 위상차로 인한 불안정한 특성이 발생한다.

모터 제어의 주된 특징은 과도 응답 동안에 시스템의 구조가 끊임없이 변하는 것이다. 모터 제어에서 시스템의 구조는 미리 설정한 알고리즘이나 구조적인 변화의 법칙에 따라 임의로 가변된다. 이와 같이 구조의 변화가 일어나는 시간은 오차 신호와 오차 신호의 미분 값에 따라 결정된다. 일반적으로 선형시스템에서는 시스템의 응답 속도를 높이면 정정시간은 길어지고, 정정시간을 줄이면 응답속도가 느려지는 특성이 있다. 과도 응답 동안에 변화되는 시스템의 구조

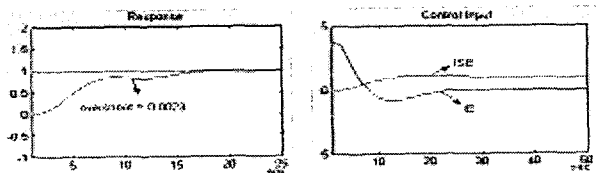


그림 25. 신경망 PID 제어기의 첨두치와 제어입력  
Fig. 25. Overshoot and control input of neural networks PID controller



는 기존의 자동제어에서 발생하는 응답 속도와 정정시간 사이의 상반되는 요구조건을 해결할 수 있다.

기존 시스템에서는 장치와 PC 사이의 응답 속도가 느리며, 사용자가 직류·스텝 모터 장치의 어플리케이션을 통해 제어명령을 하는데 사용이 복잡하고 어려웠다.

그래서, 이 논문에서는 직류 스텝 모터 장치와 PC 사이에 제어·계측의 응답 속도를 개선하였으며, PC에서의 직류 스텝 모터 장치 어플리케이션을 개발함으로써 사용자의 사용을 편리하게 했다. 실제 산업 현장에서 모터 장치의 응답 속도를 개선함으로써 제어기뿐 만 아니라 전체 시스템의 성능을 향상시켰고, 사용자 편리성 중심으로 개발된 PC에서의 어플리케이션은 작업 능력을 향상시킬 수 있다. 향후 신경망을 이용한 모터 장치의 개발은 산업 현장의 제어기 성능 향상뿐 만 아니라 가정용, 실험용 모터장치에서도 성능 향상이 기대된다.

**참 고 문 헌**

- [1] 정도영, 박영록, 심낙순, 박종국, 허규환, “모터 제어·감시를 위한 Monitoring device 개발”, 산업과학기술연구 논문집, Vol.-6, pp. 17-22, 2001.
- [2] 김동희, 이시일, 김성식, 유동완, 서보혁, “신경망을 이용한 입력 제한 플랜트의 제어”, 2000년도 대한전기학회 하계학술대회 논문집, pp. 2693-2695, 2000.
- [3] 이정일, 조영빈, 김진현, “알기쉬운 80196 마이크로컨트롤러”, 도서출판 차송, 1999.
- [4] J. M. Jurada, “Introduction to Artificial Neural Systems”, PWS, pp. 163-206, 1992.
- [5] K. S. Narendra, K. Parthasarathy, “Identification and Control of Dynamical systems Using Neural Networks”, IEEE trans. on Neural Networks, Vol. 1, No. 1, pp. 4-27, 1990.
- [6] S. Haykin, “Network: A Comprehensive Foundation”, Prentice-Hall International, 1990.
- [7] J. Hertz, A. Krogh, B. Lautrup, T. Lehmann, “Nonlinear Backpropagation: Doing Backpropagation Without Derivative of the Activation Function”, IEEE trans. on Neural Networks, Vol.-8, No.-6, pp. 1321-1326, 1997.
- [8] P. D. Wasserman, “Advanced Methods in Neural Computing”, Van Nostrand Reinhold, New York, 1993.
- [9] H. Gomi, M. Kawato, “Learning control for a closed loop system using feedback-error-learning”, Proceedings of the 29th IEEE Conference on 5-7 Dec. Vol.-6, pp. 3289-3294, 1990.
- [10] Jeong-Woo Lee, “Inversion Control of Nonlinear Systems with Neural Network Modelling”, IEE Proceeding D(Control System and Application) Vol.-9, pp. 481-488, 1997.
- [11] 손준혁, 서보혁, “신경망을 이용한 직류·스텝모터의 속도제어”, 2003년도 대한전기학회 하계학술대회논문집 pp. 2245-2247, 2002.

- [12] 손준혁, 서보혁, “신경망을 이용한 PID 제어기의 제어 사양 최적의 이득값 추정”, 2003년도 대한전기학회 시스템 및 제어부문학회 논문집, pp. 717-719, 2003.

저 자 소 개

**손 준 혁 (孫 峻 赫)**



1978년 3월 29일생. 2001년 9월~2003년 8월 경북대학교 공과대학 전자전기공학부 정보기술인력양성사업단 참여. 2003년~현재 경북대학교 대학원 전기공학과 박사과정.  
Tel : 053-940-8604(8804)  
Fax : 053-950-5505  
E-mail : d7101194@palgong.knu.ac.kr

**박 성 옥 (朴 省 爠)**



1964년 11월 16일생. 1987년 경북대 공대 전기공학과 졸업. 1989년 경북대 대학원 졸업(석사). 1991년 국방과학연구소 근무. 1992년~현재 구미1대학 컴퓨터응용전기 전공 부교수.  
Tel : 054-440-1203  
Fax : 054-440-1209  
E-mail : swpark@kumi.ac.kr

**서 보 혁 (徐 輔 赫)**



1952년 3월 11일생. 1975년 서울대 공대 전기공학과 졸업(학사). 1980년 동 대학원 전기공학과 졸업(석사). 1985년 동 대학원 전기공학과 졸업(박사). 현재 경북대학교 공대 전기전자공학부 교수

**이 상 철 (李 相 喆)**



1952년 8월 13일생. 숭실대학교 전기공학과 졸업. 1981년 동 대학원 전기공학과 졸업(석사). 1991년 동 대학원 전기공학과 졸업(박사). 1981~현재 동서울대학교 전기정보제어과 교수.  
Tel : 031-720-2065  
Fax : 031-720-2262,  
E-mail : scllee@dsc.ac.kr