DSP와 FPGA를 이용한 지능 제어기의 하드웨어 구현

Hardware Implementation of an Intelligent Controller with a DSP and an FPGA for Nonlinear Systems

김 성 수, 정 슬*
(Sung-Su Kim and Seul Jung)

Abstract : In this paper, we develop control hardware such as an FPGA based general purposed intelligent controller with a DSP board to solve nonlinear system control problems. PID control algorithms are implemented in an FPGA and neural network control algorithms are implemented in a DSP board. An FPGA was programmed with VHDL to achieve high performance and flexibility. The additional hardware such as an encoder counter and a PWM generator can be implemented in a single FPGA device. As a result, the noise and power dissipation problems can be minimized and the cost effectiveness can be achieved. To show the performance of the developed controller, it was tested for nonlinear systems such as a robot hand and an inverted pendulum.

Keywords : FPGA, VHDL, DSP, intelligent controller

I. 서론

일반적으로 PID 컨트롤러는 모션 제어에 있어서 가장 많이 사용되며 쉽게 적용이 가능한 제어기이다. 선형 시스템에 대해서는 간단한 이득값의 조절로 원하는 출력응답을 얻을 수 있다. 하지만, 시스템의 변수가 변하거나 외부로부터 외란이 있는 경우 등은 PID 제어기로 원하는 응답을 얻기 어렵게 된다. PID 제어기 이득값이 고정되어 있기 때문에 플랜트의 동적 특성이 변화하는 비선형 시스템에 대해서는 제어 성능이 떨어지게 되는 것이다.

이전의 연구에서 FPGA(Field Programmable Gate Array)로 HDL(Hardware Description Language) 기반의 PID 제어기를 구현하여 역진자에 적용하여 실험하여 보았다[1]. 진자의 각은 비교적 제어가 잘 되는 반면, 수레의 위치를 제어하기가 어려웠다. 이는 SIMO 비선형 시스템의 대표적인 현상으로 선형 제어기로는 제어가 어렵다는 것을 보여준다.

따라서 비선형 시스템을 제어하기 위해서는 비선형 시스템의 동특성 변화를 보상할 수 있는 비선형 제어기가 필요하게 된다. 비선형 제어기 중에서 신경회로망 제어기는 시스템의 동특성 변화, 외란과 같은 불확실성을 보상함으로써 비선형 시스템에 대해 좋은 제어 성능을 가진다. 따라서, 많은 연산량이 필요한 신경망 알고리즘을 수행하기 위해서는 빠른 연산 속도가 보장되는 마이크로 컨트롤러가 필요하게 된다. 그래서, DSP와 같이 고속 연산이 가능한 마이크로 프로세서가 주로 많이 사용된다[2]. 하지만, CPU의 처리속도, 하드웨어적인 제약에 의해 센서 인터페이스, 사용자 인터페이스, 다른 프로세서와의 통신 등과 같은 다수의 작업들을 수행하기에는 한계가 있다.

위의 문제점을 해결함과 동시에 비선형 시스템의 제어가 가능하도록 하기 위해 본 논문에서는 비선형 시스템을 위한 DSP와 FPGA를 기반으로 하는 지능제어기를 구현하였다. DSP에는 신경망 제어 알고리즘이 설계되고, FPGA에는 PID 제어기가 설계된다.

FPGA를 이용한 설계 방법 때문에 개발 시간이 단축되며 시스템의 용도, 구성, 목적의 변화에 쉽게 대처 및 변경이 용이한 장점을 갖고 있다. 또한 고용량의 FPGA를 사용함으로써 엔코더 카운터, PWM 발생기와 같은 부가적인 하드웨어도 하나의 FPGA로 설계 가능하기 때문에 시스템 개발비용 면에서도 매우 효과적이다[1]. 신경망[3], 퍼지[4]와 같은 제어기의 하드웨어적 구현, 모터 엔코더와 같은 측정 회로[5], 다관절 로봇 및 이동 로봇에서의 모터 제어[6-8]와 같은 분야가 그 대표적인 예라 할 수 있겠다.

본 논문에서는 위와 같은 특징을 갖는 FPGA를 기반으로 DSP를 이용한 비선형 지능 제어기를 설계함으로써 제어기 보드의 크기 및 설계 비용을 감소시킬 수 있었다[9]. 개발된 제어기의 성능을 검증하기 위해 본 논문에서는 두 가지의 시스템이 사용되었다. 첫 번째는 손가락 로봇 시스템으로서, 3개의 손가락과 각 손가락마다 2개의 관절을 갖는 손가락 로봇이며, 두 번째는 역진자 시스템이다. 손가락 로봇 시스템의 경우, 주기적인 사인파를 로봇의 제어 입력으로 설정하여 로봇의 추종 정도를 살펴 보았고, 역진자 시스템의 경우 진자의 각도와 수레의 위치를 동시에 제어함으로써 제어기의 성능을 검증하였다.

II. 시스템 구성

1. 전체 시스템 구성

본 논문에서 구현하고자 하는 제어기는 그림 1에 보여진 입력 보상 방식의 제어 알고리즘이다. 기존의 제어기가 있고 신경회로망 제어기가 입력에 보상해 줌으로써 공정의 변수의 변화나 불확실성에서 오는 출력 오차를 보상할 수 있게 된다[2, 10]. 용량의 제한 때문에 고속의 연산 처리가 가능한 DSP가 메인 컨트롤러가 되고 FPGA를 하위 컨트롤러로 둠으로써 복잡한 비선형 시스템에 대한 제어가 가능해진다. 그림 1에서 신경망 제어기는 DSP 보드로 구현하고 제어기로는 FPGA로 PID 제어기를 구현하고자 한다.

* 책임저자(Corresponding Author)
논문접수 : 2003. 10. 14., 채택확정 : 2004. 7. 2.
김성수, 정 슬 : 충남대학교 메카트로닉스공학과
(a741103@hanmail.net/jungs@cnu.ac.kr)

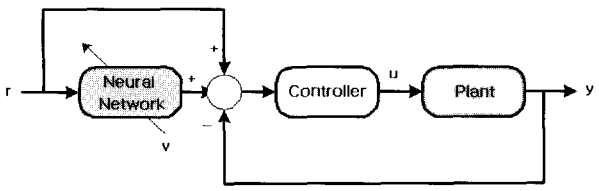


그림 1. 입력 보상 방식 구조.
Fig. 1. Reference compensation control structure.

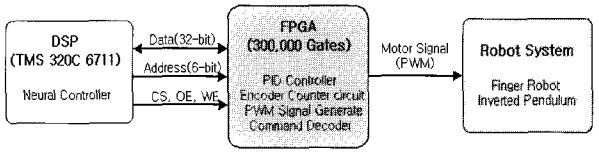


그림 2. 시스템의 전체 구성.
Fig. 2. Overall system structure.

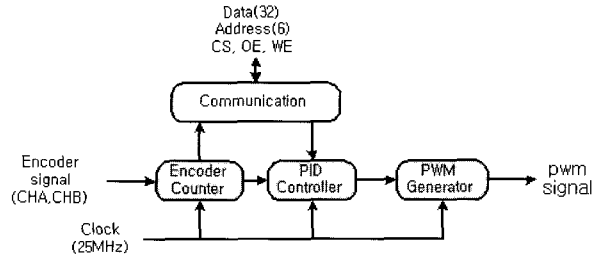


그림 3. FPGA에 설계된 PID 제어기의 구조.
Fig. 3. PID controller structure embedded in FPGA.

그림 2는 제어기의 하드웨어적 구성을 보여주고 있다. 고속 연산 처리가 가능한 DSP는 신경망 알고리즘이 적용되고, FPGA에는 PID 제어기가 설계된다. 엔코더의 신호 처리, 모터 구동을 위한 PWM(Pulse Width Modulation) 신호 발생 회로, 그리고 DSP로부터의 명령을 해독하는 명령 해독기 회로가 FPGA에 포함된다.

DSP와 FPGA간의 통신은 32비트 데이터 버스를 이용하여 이루어지며 제어를 위해 사용되는 각각의 명령들은 어드레스에 의해 구분된다.

2. FPGA 기반의 PID제어기 설계

2.1 PID 제어기의 구조

FPGA제어기의 설계는 그림 3에서 보듯이 DSP와 통신하기 위한 통신 블록, 엔코더 카운터 블록, PID 컨트롤러 블록, 그리고 모터 신호 발생 블록으로 구성된다. 입력 신호로는 DSP와의 통신을 위한 데이터 버스 32 비트, 어드레스 버스 6비트, 그리고 CS, OE, WE 이 있으며, 엔코더 신호, 그리고 25㎒의 클록이 입력된다. 그리고 모터 구동을 위한 PWM 신호가 출력된다. 이전의 연구에서는 이 FPGA 기반의 PID 제어기를 사용하여 역진자를 구동하는 실험을 해 보았다[1].

2.2 Encoder Counter 블록

모터 엔코더 신호의 처리는 모터 제어에서 매우 중요하다. 정확한 모터의 속도 및 포지션 제어를 위해서는 엔코더 신호

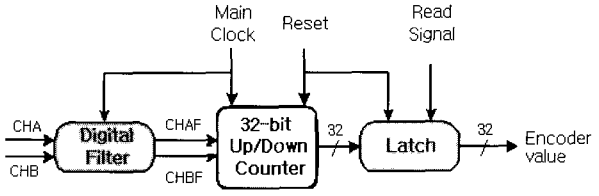


그림 4. 엔코더 카운터 구조.
Fig 4. Encoder counter block.

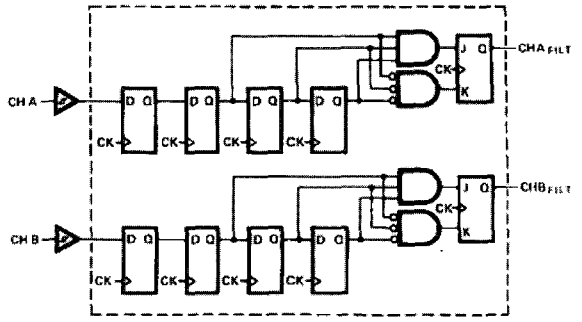


그림 5. 디지털 필터의 구조.
Fig. 5. Digital filter structure.

의 정확하고 정밀한 계수가 필수적이다. 본 논문에서 디자인된 엔코더 카운터는 A, B 두 상의 엔코더 신호의 위상차를 검출하여 정, 역 카운트가 가능하며, 엔코더 신호 한 주기 당 4채배로 계수된다.

그림 4는 엔코더 카운터 블록의 구조를 나타낸다. 모터 엔코더 카운터 블록은 모터와 같은 기계적인 시스템과 연결되기 때문에 잡음 신호가 엔코더 신호에 포함될 수 있다. 이러한 잡음을 제어하기 위한 디지털 필터가 그림 5처럼 포함되었다.

설계된 엔코더 카운터는 32비트의 크기를 가지며 정, 역 방향으로 계수가 가능하다. DSP에서 발생되는 엔코더 read 신호에 의해 카운터 값이 출력된다.

2.3 PID 제어기 블록

설계된 PID 제어기는 다음과 같이 나타낸다.

$$u(t) = K_P e(t) + K_I \int e(t)dt + K_D \dot{e}(t) \quad (1)$$

여기에서 K_P, K_I, K_D 는 각각 PID 이득값이며, $e(t)$ 는 기준값과 출력값의 오차로 진자 각의 오차 $e_\theta(t)$ 또는 $e_x(t)$ 를 나타낸다. 여기에서는 위 식을 이산화하여 다음과 같이 사용하였다.

$$u(k) = K_P e(k) + K_I \sum e(k) + K_D (e(k) - e(k-1)) \quad (2)$$

이 블록에서는 제어 주기마다 오차를 계산하여 PID 제어기의 출력을 계산한다.

2.4 PWM 발생 블록

PID 제어기에서 생성되는 모터 제어량을 모터의 입력 신호로 변환한다. 제어기 입력 $u(k)$ 를 모터의 입력 신호인

표 1. 어드레스 버스에 따른 명령 구분.
Table 1. Instructions on address bus.

구분	A3	A2	A1	A0	기능
Write	0	0	0	1	P gain write
	0	0	1	0	I gain write
	0	0	1	1	D gain write
	0	1	0	0	Motor enable
	0	1	0	1	Motor disable
	0	1	1	0	Desired value write
	0	1	1	1	FPGA reset
	1	0	0	0	신경망 제어기 출력 write
Read	0	0	0	1	엔코더 카운터read, PID 제어시작

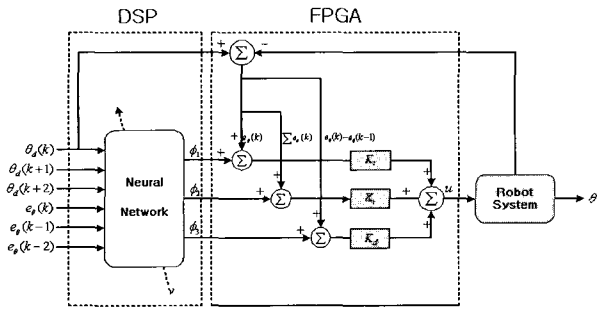


그림 6. 로봇손 제어를 위한 신경망 제어 방식의 구조.
Fig. 6. Neural control structure for robot hand.

PWM으로 만들며, 그 주파수는 12㎑이다. 이 PWM 신호는 모터 드라이버에 입력되어 모터를 구동하게 된다.

2.5 통신 블록

통신 블록은 주로 DSP와 FPGA 간의 동기화 문제를 다룬다. 한 샘플동안 DSP 연산이나 PID 연산이 이루어 져야 하므로 동기화가 필요하다. FPGA의 PID 제어 주기와 DSP의 신경망 제어 주기를 동기화하기 위해서 DSP로부터 명령을 통해 FPGA의 PID 제어 시간을 결정한다.

DSP의 신경망 제어와 FPGA의 PID 제어가 각각 독립적으로 진행되기 때문에 서로간에 데이터가 공유되어야 한다. 그러기 위해 32비트 데이터 통신을 이용해서 DSP에서 FPGA로 명령을 전달하는 방식을 사용하였다. 명령의 구분은 어드레스 버스를 이용했으며, 표 1에서 그 내용을 나타내고 있다.

실질적인 명령 구분은 A3~A0까지만 사용된다. 그리고 A5, A4는 FPGA에 여러 개의 PID 제어기를 설계할 경우 구분하기 위한 핀이다. 따라서 이 경우 4개의 PID 제어기를 설계할 수 있다.

위의 표 1에서 DSP에서 FPGA로의 명령 전달에 관해 좀 더 세부적으로 설명하면 다음과 같다.

• FPGA 초기화 : FPGA 내부 변수들을 초기화한다(Write)(엔코더 카운터, PID 제어기의 이득값 등).

• PID 이득값 설정 : FPGA에 설계된 PID 제어기에서 사용할 PID 이득값을 설정한다(Write).

• 기준 값 설정 : 샘플링 시간마다 PID 제어기에서 사용할 기준값을 설정한다(Write).

• 신경망 제어 출력 값 설정 : 샘플링 시간마다 DSP에서 연산된 신경망 출력을 FPGA에서 전송하고, FPGA에서는 PID 제어기의 출력과 더해서 모터 신호를 생성한다(Write).

• 엔코더 값 read : 샘플링 시간마다 계수된 엔코더 값을 읽어 온다. 그와 동시에 FPGA내의 PID 제어가 한번 실행된다. 이 기능을 통해서 DSP의 신경망과 FPGA의 PID 제어기가 동기화를 이룰 수 있다.

III. 신경망 제어기 구현

1. 입력 보상 방법

신경회로망은 비선형 함수의 맵핑 능력, 일반화 능력 및 학습 능력이 있어 다양한 분야에서 사용되고 있다[11]. 이러한 신경회로망은 제어할 공정의 정확한 모델을 실제적으로 얻기 어려운 제어 시스템 및 정확하지 않은 대략적인 동적 방정식이 주어진 제어 시스템 등의 불확실성과 부정확성을 해결하기 위해 기존의 PID 또는 PD 제어기와 더불어 보조 제어기 역할로 사용되어 정상상태에서의 오차를 영으로 수렴하도록 한다. 본 논문에서 사용되는 신경회로망 제어방식은 그림 6과 같이 입력 경로에 신호를 보상하는 입력보상 방식과 PID 제어기를 사용한다. 이 방식의 특징은 기존의 제어기를 수정하지 않고 시스템 외부에서 내부의 불확실성을 보상할 수 있다는 것이다.

신경망 제어기는 제어 회로 밖에서 선 필터의 기능을 수행한다. 즉 신경 회로망이 PID 제어기로 들어가는 경로 오차의 입력 신호를 보상해주는 방식이다. 이렇게 함으로써 시스템의 내부 제어기는 수정하지 않고 외부에서 제어가 가능하게 된다. 이렇게 보상된 신호는 PID 제어기에서 이득값과 곱해져서 증폭이 된다. 이러한 방식을 사용함으로써 신경망이 공정대상의 역모델을 학습하고 신경망 제어기의 안정성을 확보할 수 있다. 오차 신호는 다음의 식과 같다.

$$e_\theta = \theta_d - \theta \tag{3}$$

진자의 각을 제어하기위한 PID 제어기는

$$u_\theta = k_{P\theta} e_\theta(t) + k_{I\theta} \int e_\theta(t)dt + k_{D\theta} \dot{e}_\theta(t) + k_{P\theta}\phi_1 + k_{I\theta}\phi_2 + k_{D\theta}\phi_3 \tag{4}$$

역진자의 경우에는 진자의 각 뿐만 아니라 수레의 위치도 제어하므로 수레의 위치 오차는 다음과 같다.

$$e_x = x_d - x \tag{5}$$

수레를 위한 PID제어기는

$$u_x = k_{Px} e_x(t) + k_{Ix} \int e_x(t)dt + k_{Dx} \dot{e}_x(t) + k_{Px}\phi_4 + k_{Ix}\phi_5 + k_{Dx}\phi_6 \tag{6}$$

결과적으로 역진자 시스템의 제어 입력은 다음과 같다.

$$u = u_x + u_\theta \tag{7}$$

2. 신경망 구조

그림 7은 로봇 손을 제어하기 위한 신경 회로망 구조를 나타낸다.

로봇 손 제어를 위해 사용한 신경망은 그림 7에 나타난 바와 같이 입력층, 은닉층 그리고 출력층으로 구성된 다층 역전파 신경 회로망을 사용하였다. 신경망 입력층의 입력으로는 기준 각도 값, 각도 오차를 사용하였다. 신경망이 시스템의 동적 행동을 학습할 수 있도록 2step 다음까지의 기준값과 2step 이전까지의 오차값이 같이 입력된다. 그리고 비선형 함수로는 다음과 같은 hyperbolic tangent함수를 사용하였다.

$$f(x) = \frac{1-\exp(-x)}{1+\exp(-x)} \tag{8}$$

역진자 제어의 경우에는 입력층, 은닉층 그리고 출력층의 뉴론 개수는 진자의 각도, 수레의 위치의 기준값과 오차값이 입력으로 사용되기 때문에 각각 12, 12, 6개이다.

3. 신경망 학습

여기에서는 역전파 알고리즘을 사용하여 시스템 자코비안이 필요 없이 on-line 학습이 가능하도록 하였다. 신경망의 출력을 다음과 같이 설정하자.

$$\Phi = k_P\phi_1 + k_I\phi_2 + k_D\phi_3 \tag{9}$$

만약 시스템의 동력학식이 $f(\theta,\dot{\theta},\ddot{\theta})$ 이면 (4)와 (6)은 다음 식으로 유도될 수 있다.

$$k_P e + k_I \int e dt + k_D \dot{e} = f(\theta,\dot{\theta},\ddot{\theta}) - \Phi \tag{10}$$

이 때 신경망 학습 신호를 (11)과 같이 설정하고 목적함수를 (12)로 정하여 목적함수가 영으로 수렴하도록 신경망을 학습시킨다. 이렇게 신경망이 학습이 되면 $\Phi \cong f(\theta,\dot{\theta},\ddot{\theta})$ 이 되어 시스템의 inverse dynamics를 수행한다.

$$v = k_P e + k_I \int e dt + k_D \dot{e} \tag{11}$$

오차를 줄이는 목적함수는 다음과 같다.

$$E = \frac{1}{2}v^2 \tag{12}$$

(12)를 가중치에 대해서 미분하면 목적함수 E 의 그래디언트를 (13)처럼 구할 수 있다. (10)으로부터

$$\frac{\partial E}{\partial w} = \frac{\partial E}{\partial v}\frac{\partial v}{\partial w} = v\frac{\partial v}{\partial w} = -v\frac{\partial \Phi}{\partial w} \tag{13}$$

여기서,

$$\frac{\partial \Phi}{\partial w} = K_P\frac{\partial \phi_1}{\partial w} + K_I\frac{\partial \phi_2}{\partial w} + K_D\frac{\partial \phi_3}{\partial w} \tag{14}$$

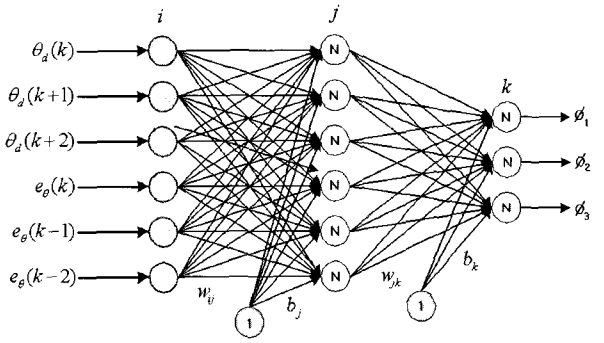


그림 7. 신경망 구조.

Fig. 7. Neural network structure.

위에서 구한 (13), (14)를 이용하여 역전파 알고리즘에 사용하면 다음과 같다.

$$\Delta w(t) = \eta\frac{\partial \psi}{\partial w}v + \Delta w(t-1) \tag{15}$$

$$w(t+1) = w(t) + \Delta w(t) \tag{16}$$

4. 신경망 제어기 하드웨어 구현

사용된 하드웨어는 DSP는 TI사의 TMS320C6711이고 FPGA는 Altera사의 APEX II EP20K300EQC240이며, 그림 8과 같이 구성되었다.

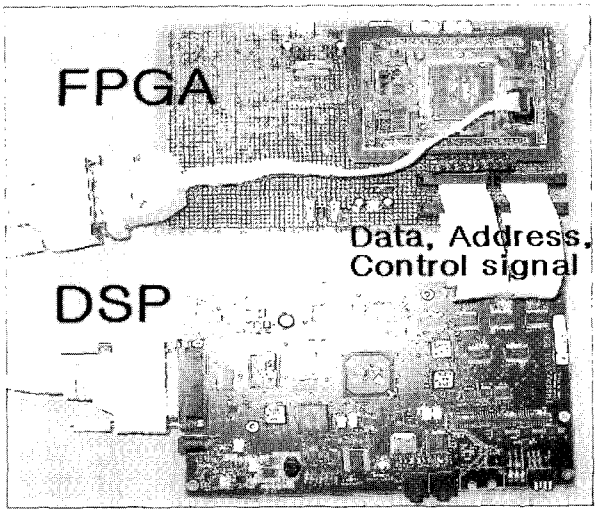


그림 8. DSP-FPGA 신경망 제어기 시스템 사진.

Fig. 8. Neural network control hardware.

IV. 실험 결과

손가락 로봇과 역진자 시스템을 이용하여 DSP와 FPGA를 이용한 모션 제어기의 성능을 검증하였다. 손가락 로봇의 경우, 로봇의 경로로 사인파를 생성하여 로봇의 제어 상태를 확인하였다. 그리고 역진자 시스템의 경우 역진자의 각도와 위치를 동시에 제어하는 실험을 하였다.

1. 손가락 로봇 시스템

로봇의 좌표계는 첫 번째 손가락의 경우 그림 9와 같이

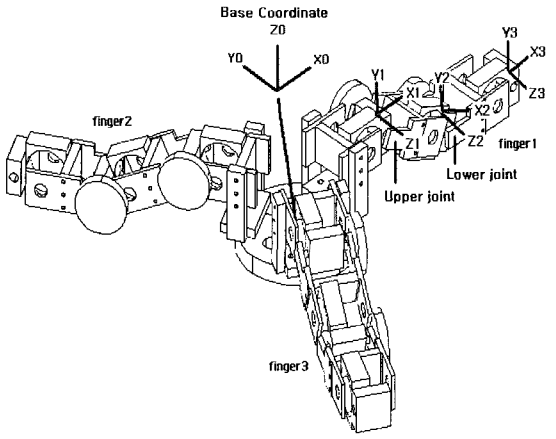


그림 9. 손가락 로봇의 좌표계 정의.
Fig. 9. Robot hand coordinates.

표 2. D-H 변수값.
Table 2. D-H parameters.

Joint		θ_i	d_i	a_i	α_i
Finger1	1	0	37.5	90	62
	2	-25	0	0	55
	3	25	0	0	55
Finger2	1	120	37.5	90	62
	2	-25	0	0	55
	3	25	0	0	55
Finger3	1	240	37.5	90	62
	2	-25	0	0	55
	3	25	0	0	55

설정되었으며, 두 번째, 세 번째 손가락의 경우 역시 같은 방법으로 정의된다. 단지 두 번째 손가락의 경우 기준 좌표계를 중심으로 120도, 세 번째 손가락은 240도가 각각 회전하였다. 그림 9는 손가락 로봇의 좌표계 설정을 나타내고, 그때의 D-H 변수들은 표 2와 같다.

본 실험에서는 신경망-PID 제어기를 이용한 로봇 제어와 PID 제어기를 이용한 로봇 제어의 결과를 비교함으로써 개발된 제어기의 성능을 확인할 수 있었다. 그 결과를 그림 10~12에 나타내고 있다. 그림에서 보듯이 약 0.5초~1.25초까지가 손가락 로봇이 중력 방향으로 움직이는 경우이다. 이 경우 모터의 토크, 중력에 의해서 진동이 발생하는 것을 볼 수 있다. 이 영역에서 PID 제어기만을 사용할 경우 최대 약 ± 0.7도 정도의 오차 범위 내에서 진동이 일어나고 있지만, 신경망-PID를 같이 사용할 경우 약 ± 0.3도 이내의 오차 범위를 갖고 있음을 볼 수 있다. 뿐만 아니라 전반적인 모든 영역에서 신경망과 PID를 같이 사용할 경우 PID만 사용한 경우보다 좋은 결과를 얻을 수 있었다.

2. 역진자 시스템 실험

그림 13에서 처럼 본 실험에서는 역진자의 위치와 각도를 동시에 제어하는 실험을 수행하였다. 역진자의 각도는 영을 유지하고 위치를 원하는 값으로 제어함으로써 개발된 비선형 제어기의 성능을 검증하였다. 본 실험에서는 위치를 영으로 하는 실험과 사인파의 궤적을 추종하는 실험이 수행되었다.

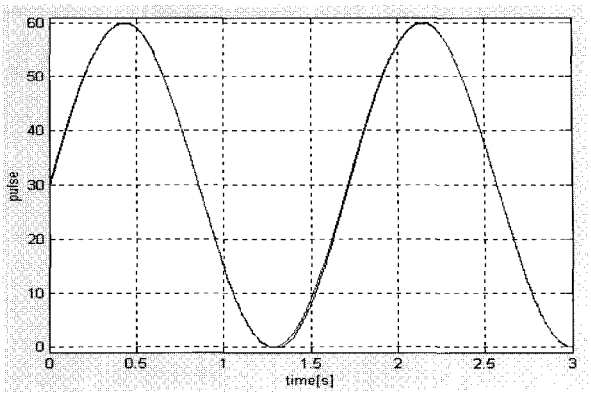


그림 10. 신경망-PID 제어기의 로봇의 위치 추종.
Fig. 10. Position tracking of neural controller.

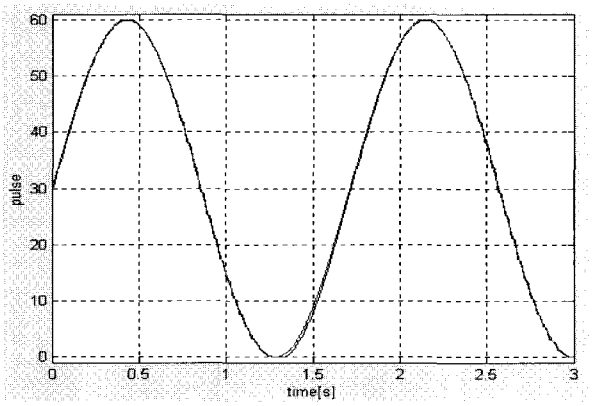


그림 11. PID 제어기를 이용할 경우 로봇의 위치 추종.
Fig. 11. Position tracking of PID controller.

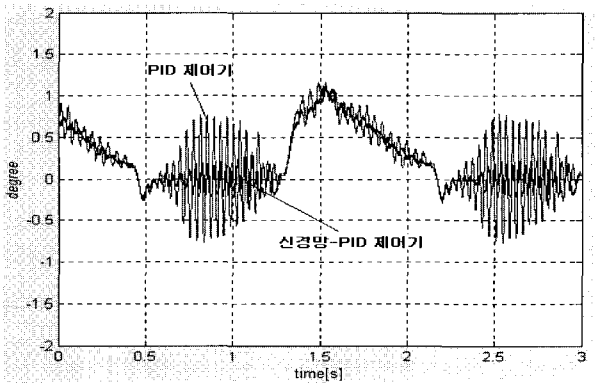


그림 12. 신경망-PID 제어기와 PID 제어기의 에러비교.
Fig. 12. Positional errors of neural-PID and PID.

실험에 사용된 신경망과 PID 이득값은 표 3과 같다. PID 제어기의 이득값은 실험적인 시행착오 과정을 거쳐 어느 정도 최적화 되었다. 마찬가지로 학습율과 모멘텀 상수도 시행착오를 거쳐 설정되었다. D게인은 잡음에 민감하므로 디지털 필터를 사용해서 처리하였다[12].

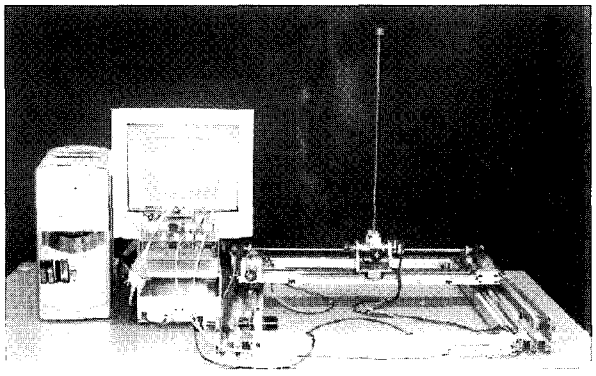


그림 13. 역진자 시스템 사진.
Fig. 13. Inverted pendulum system.

표 3. 신경망과 PID 이득값.
Table 3. Neural network parameters and PID gains.

종류		이득값
각도	k_p	-5.5
	k_i	0.002
	k_d	-1.3
위치	k_p	-1.5
	k_i	-0.05
	k_d	-1.3
학습률(η)		0.04
모멘트(α)		0.3

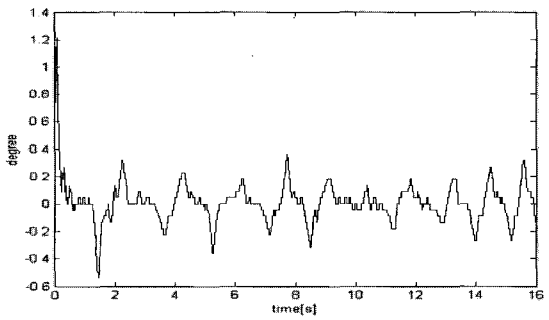


그림 14. PID제어기로 제어된 역진자의 각도.
Fig. 14. The angle of inverted pendulum of a PID controller.

2.1 역진자 자세 제어

2.1.1 PID 제어

그림 14와 15는 PID제어기로 제어한 역진자의 움직임을 나타낸다. 역진자의 균형이 잘 유지됨을 알 수 있다. 하지만 수레의 위치 추종에 있어서는 계속적으로 움직이는 결과를 보여주고 있다. 결과적으로 수레의 위치 제어가 잘 되어지지 않고 계속 움직이고 있음을 나타낸다.

2.1.2 신경망 제어

먼저 역진자의 제어 입력으로 각도와 위치를 영으로 하는 실험의 결과를 그림 16과 17에 보여 주고 있다.

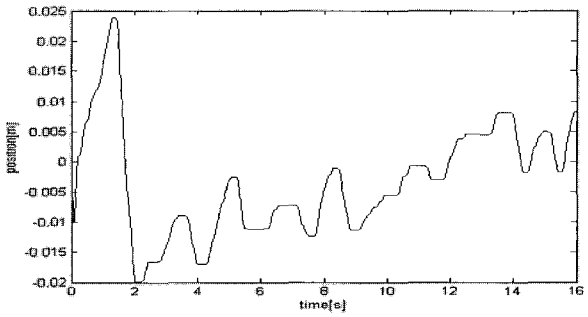


그림 15. 수레의 위치.
Fig. 15. The position of a cart.

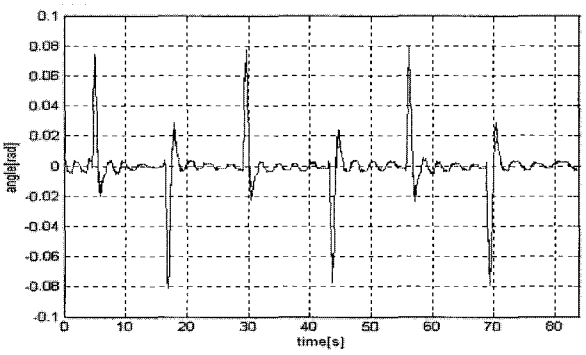


그림 16. 위치 및 각도 실험에서의 역진자의 각도.
Fig. 16. Angle of the pendulum.

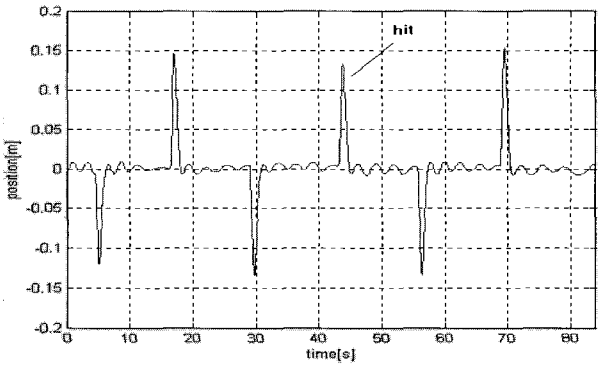


그림 17. 위치 및 각도 실험에서의 역진자의 위치.
Fig. 17. Position of the cart.

역진자를 세워 놓고 그 각도를 원점으로 초기화 한 후, 외란으로 임의로 진자를 쳤을 때 다시 원점으로 돌아오도록 하는 실험을 하였다. 그 결과 그림 17에서 보여지는 것처럼 수레의 경우에 진자를 칠 경우 원점으로 돌아오는 것을 볼 수 있다. 그리고 수레위치는 ± 1㎝ 이내의 영역에서 계속 유지하고 있음을 볼 수 있다. 각도는 그림 16에서 보여주고 있다. 각도 역시 진자를 손으로 쳤을 경우 각도가 커졌다가 다시 영으로 수렴하는 것을 볼 수 있다. 각도는 ± 0.005rad 이내의 영역에서 일정하게 유지되고 있다. 외란이 있을 시에도 제어기의 강건함을 보여준다.

2.2 역진자 위치 추종 제어

역진자의 두 번째 실험으로는 진자가 서도록 한 상태에

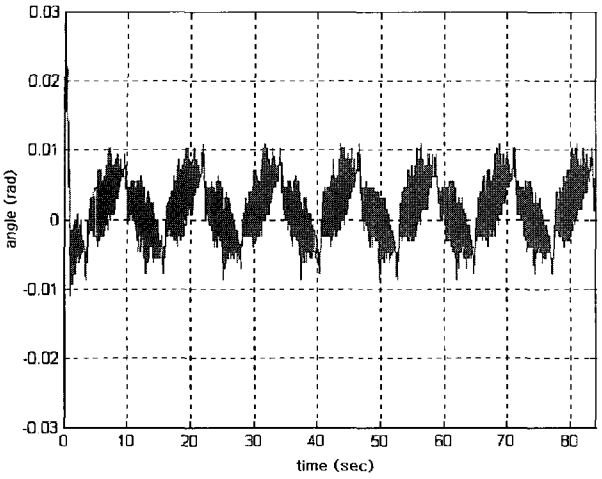


그림 18. 주기가 4π sec 일 때의 진자 각도.
Fig. 18. Angle of pendulum when T= 4π sec .

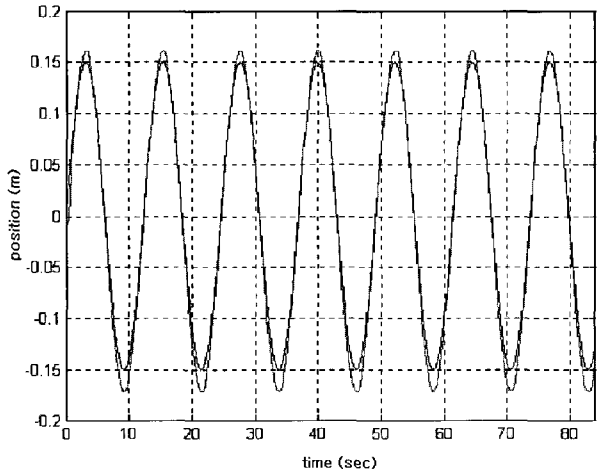


그림 19. 주기가 4π sec 일 때의 수레의 위치 추종.
Fig. 19. Position of the cart when T= 4π sec .

서 수레가 사인파의 위치 경로를 추종하는 것이다. 위치 제어의 기준 입력으로는 사인함수를 설정하였다. 본 실험에서는 주기가 다른 두 사인파를 제어 입력으로 사용하여 실험하였다. 그림 18에서 그 결과를 나타내고 있다. x(t)는 수레의 위치 경로이다.

① $x(t) = 0.15 \sin \frac{t}{2}$

먼저 주기가 4π sec 일 때의 사인파 경로를 추종하는 경우이다. 그림 18은 진자의 각도를 나타내는데 0도를 잘 유지하고 있음을 알 수 있다. 그림 19는 수레의 위치 추종을 나타내는데 방향이 바뀌는 부분에서는 동적 현상에 의한 오버슈트가 발생함을 볼 수 있다. 그림 20은 위치추종의 오차를 보여준다.

② $x(t) = 0.15 \sin \frac{t}{3}$

다른 경로로 주기를 더 크게 하여 속도를 늦추어 보았다. 그림 21에서는 진자 오차를 나타내는데 앞의 경우보다

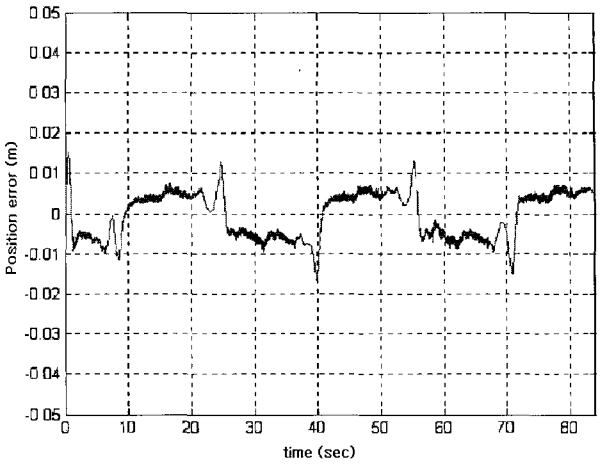


그림 20. 주기가 4π sec 일 때의 수레 위치 오차.
Fig. 20. Positional error when T= 4π sec .

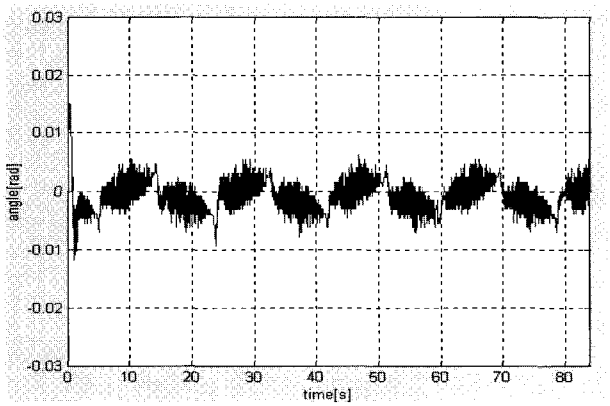


그림 21. 주기가 6π sec 일 때의 진자 각도.
Fig. 21. Angle of pendulum when T= 6π sec .

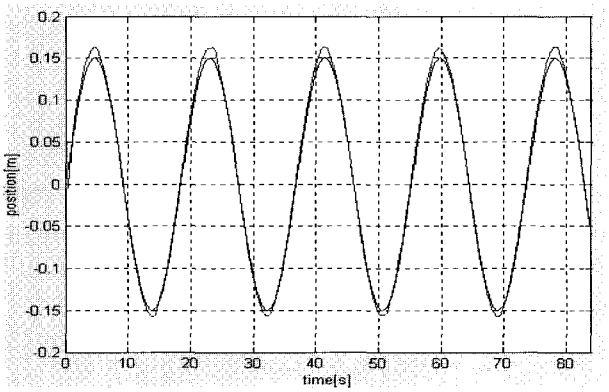


그림 22. 사인파의 주기가 6π sec 일 때의 수레 위치 추종.
Fig. 22. Positional tracking when T= 6π sec .

다소 오차가 줄어듦을 알 수 있다. 그림 22와 23은 수레의 위치 추종 오차를 나타내는데 앞의 예제보다 오차가 줄어듦을 알 수 있다.

위의 실험에서도 역시 모든 경우에 대해서 각도는 0.007rad 이내의 오차 범위를 가지며, 위치는 1㎝의 오차 내에서 유지

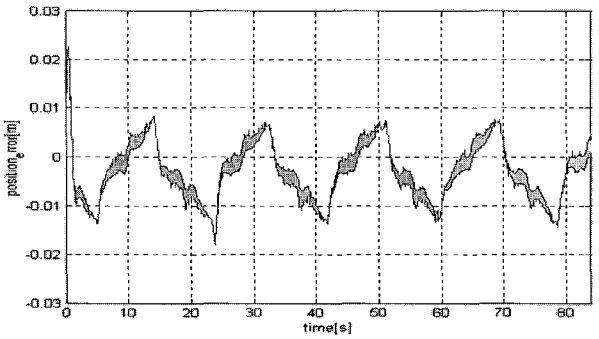


그림 23. 사인파의 주기가 6π sec 일 때의 위치 오차.
Fig. 23. Positional error when T= 6π sec .

되는 것을 볼 수 있다. 위의 실험을 통해 본 논문에서 개발된 제어기가 복잡한 비선형 시스템 제어에 매우 효과적임을 알 수 있다. 심지어 손으로 진자를 쳐서 외란을 주었을 경우에도 진자는 균형을 잘 유지하고 수레는 위치를 잘 추종함을 실험적으로 확인하였다.

V. 결론

본 논문에서는 DSP와 FPGA를 이용한 모션 컨트롤러의 개발에 관한 내용에 대해 설명하였다. 일반적으로 가장 많이 사용되는 DC 모터 제어를 기본으로 하였으며, DSP와 FPGA를 이용하여 DSP에서는 신경망 제어기, FPGA에서는 PID 제어기를 설계함으로써 복잡한 비선형 시스템 제어를 위한 모션 컨트롤러를 개발하였다.

손가락 로봇과 역진자를 이용하여 개발된 모션 컨트롤러를 실험하였으며, 그 성능이 효과적임을 입증하였다. 또한 FPGA를 이용한 설계 방식과 FPGA의 특성에 의해 제어기 전체의 부피가 현저히 감소하였으며, 대상 시스템에 변화에 따른 제어기의 수정이 간단함을 확인하였다. 추후 연구로는 FPGA 퍼지제어기를 적용하여 특성을 파악하여 비교하는 것이다.

참고문헌

[1] 김성수, 정슬, "FPGA를 이용한 범용 모션 컨트롤러의 개발", 제어·자동화·시스템공학 논문지, 제10권 1호 pp. 73-80, 2004.

[2] H. T. Cho and S. Jung, "Balancing and position control of an inverted pendulum on an X-Y Plane using decentralized neural networks", *Proceedings of the 2003 International Conference on Advanced Intelligent Mechatronics*, pp. 181- 186, 2003.

[3] M. Krips, T. Lammert, A. Kummert, "FPGA implementation of a neural network for a real-time hand tracking system", *Proceedings of the First IEEE International Workshop on Electronic Design*, Test and Applications, pp. 313 317, 2002.

[4] M. Cristea, J. Khor, M. McCormick; "FPGA fuzzy logic controller for variable speed generators", *Proceedings of the 2001 IEEE International Conference on Control Applications*, pp. 301 304, 2001.

[5] A. K. Oudjida et al, "A reconfigurable counter controller for digital motion control application", *Microelectronics Journal*, vol. 28, no. 6-7, 1997.

[6] F. Thomas et al, "Design and implementation of a wheel speed measurement circuit using field programmable gate arrays in a spacecraft", *Microprocessors and Microsystems*, pp. 553-560, 1999.

[7] Oh et al, "Design of a biped robot using DSP and FPGA", *Proceeding 2002 FIRA Robot World Congress*. 698-701.

[8] A. Kongmunvattana, P. Chongstivatana, "A FPGA-based behavioral control system for a mobile robot", *The 1998 IEEE Asia-Pacific Conference on Circuits and Systems*, pp. 759-762, 24-27, Nov, 1998.

[9] "LM628/LM629 precision motion controller", *National Semiconductor Corporation*, Nov, 1999.

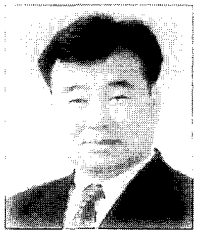
[10] S. Jung and T. C. Hsia, "Neural network inverse control techniques for PD controlled robot manipulator", *ROBOTICA*, pp. 305-314, vol. 19, no 3, 2000

[11] W. T. Miller, R. S. Sutton, and P. J. Werbos, "Neural networks for control", *The MIT Press*, 1991.

[12] H.-T. Cho and S. Jung, "Neural network position tracking control of an inverted pendulum by an X-Y table robot", *IEEE/RSJ International Conference on Intelligent Robots and System*, pp. 1210 – 1215, 2003.

김 성 수

1974년 12월 22일생. 2001년 경일대학교 제어계측공학과 졸업. 2003년 충남대학교 메카트로닉스공학과 석사. 관심분야는 S.o.C 제어기 설계, DSP 및 마이크로프로세서 응용, 로보틱스.



정 슬

1964년 9월 11일 생. 1988년 미국 웨인주립대 전기 및 컴퓨터 공학과 졸업. 1991 미국 캘리포니아대 데이비스 전기공학과 석사. 동대학 박사. 1997년~현재 충남대학교 메카트로닉스공학과 부교수. 관심분야는 지능 제어 알고리즘 및 하드웨어 구현, 로봇의 인간의 지능적인 상호 작용, 필드 로봇 시스템.