

# DSP를 이용한 로봇 제어시스템 개발

## Development of Robot Control System Using DSP

이 보 희, 김 진 결  
(Bo-Hee Lee and Jin-Geol Kim)

**Abstract** : In this paper, the design and the implementation of the controller for an articulate robot, which is developed in our Automatic Control Laboratory, are mainly discussed. The controller reduces software computational load via distributed processing method using multiple CPU's, and simplifies hardware structures by the time-division control with TMS320C31 DSP chip. The method of control is based on the fuzzy-compensated PID control with scale factor, which compensates for the influence of load variation resulting from the various postures of the robot with conventional PID scheme. The application of the proposed controller to the robot system with DC servo-motors shows some excellent control capabilities. Also, the response characteristics of system for the various trajectory commands verify the superiority of the controller.

**Keywords** : robot controller, multiple CPU's, TMS320C31 DSP Chip, scale factor, fuzzy-compensated PID control, DC servo-motors

### I. 서론

오늘날 산업 현장에서는 3D 업종을 기피하는 경향 및 임금 인상과 인력난으로 인해 자동화에 대한 관심이 집중되면서 자동화에 근간이 되는 로봇에 대한 연구가 활발하게 진행되고 있다. 사람을 대신하여 쉬지 않고 작업을 계속하는 로봇의 꿈을 실현시킨 산업용 로봇의 시초는 1962년 미국 Unimation사에 의해서이며, 그 후 다양한 종류의 로봇트가 개발되어 왔다. 1980대초 만하더라도 산업용 로봇은 물류운반, 용접과 같은 단순 반복적인 조작에만 고려했으나 현재에 와서는 조립과 같은 정교한 작업을 대상으로 하기 때문에 고속화 및 정밀화가 요구되고 있으며, 이를 위한 제어도 매우 복잡하고 다양하게 발전이 되어 왔다. 그러나 현대 제어 이론의 비약적인 발달에도 불구하고 실제 현장에서는 아직도 실현이 간편한 PID제어기가 많이 사용되고 있으나 로봇트 시스템이 불감대(deadzones), 포화(saturation), 히스테리시스(hysteresis) 등과 같은 비선형 요소들을 포함하고 있고 모델링의 불확실 혹은 동작 조건의 변화에 따라 만족할 만한 제어 성능을 얻을 수 없는 문제점을 가지고 있다[1]. 그래서 최적 제어나 적응 제어 방식과 같은 고급 제어가 사용되긴 하지만 제어 알고리즘이 복잡하고 많은 계산량이 요구되어 실시간 제어에는 어려움이 따른다[2][5]-[8]. 퍼지제어이론은 인간의 추론과 같이 근사적이고 정성적인 면을 기술하는 데에 적합하며, 시스템의 특성이 복잡하여 기존의 정량적인 방법으로는 해석할 수 없거나, 얻어지는 정보가 부정확하고 불확실한 경우에 효과적으로 적용될 수 있다[3]-[7]. 이러한 방법으로 연구된 퍼지논리 제어기(fuzzy logic controller: FLC)는 기존의 제어 알고리즘에서 필수적인 수학적 모델이 잘 정의되지 않는 공정에 대해서 효과적인 방법이며 로봇트와 같이 비선형성이 강하고 관성이 복잡하게 결합되어 있는 시스템 제어에 적합하다고 할 수 있다[4]. 그러나 이 퍼지알고리즘은 자체가 가지는 불확실한 면으로 인한 제어 성능 발현이 어려우므로 기존의 제어알고리즘에 퍼지논리를 연관시키는 방법이 제시되고 있다. 이에

관한 연구 중에서 기존의 퍼지제어기와 PI제어기와와의 관계를 설정한 Tang[9]의 연구와, Macvicar-Whelan[10]의 퍼지 제어 원리를 이용하여 PID이득 조정에 연관을 지은 Tzafestas[11]의 연구가 있으나, 이러한 방법들은 퍼지 준위의 세분화가 증가할수록 추론 법칙 수가 증가하여 실시간 제어가 어렵게 된다는 단점이 있다. Procyk, Mamdani[12]는 스스로 퍼지법칙을 수정하여 성능을 개선시키는 자동 조정 제어기(self organizing controller: SOC)에 관해 연구하였다. 그러나 이 경우에는 좋은 성능평가표를 만들기 위해서는 많은 시행착오를 거쳐야 하며 계산 시간이 많이 소요된다는 문제점이 있다. 본 논문은 이미 설계되어진 PID제어 시스템의 성능을 개선시킬 때 퍼지 보상기를 이용하여 기존의 제어 시스템을 재구성할 필요가 없이 경제적이고, 왜란이나 잡음에도 강인한 특성을 가진 제어기를 제시하였다. 퍼지보상기에 대한 연구에는 부하증가시 비례하는 비선형성의 영향에 따른 성능 저하를 보상하는 Kim[13], Chong[14]의 연구와 합성된 퍼지제어기의 안정도 해석을 한 Shieh[15]의 연구가 있는데, Kim, Chong, Shieh는 단일 기준 입력(reference input)에 대한 출력오차와 출력 오차의 변화량을 퍼지화하였으므로 가변 기준 입력을 입력으로 할 때에는 크기 변환 요소(scaling factor)를 재조정해야 될 필요가 있다. 이러한 점은 단순 퍼지 논리 제어기의 문제점이기도 한데, 최적 응답은 제한된 범위의 입력에 대해서만 가능하고 입력의 범위가 달라지면 제어기를 재조정해야 하는 문제가 있다[9]. 이러한 점을 해결하기 위하여, 측정된 출력 오차와 출력 오차의 변화량을 크기 변환할 때 고정된 상수 값이 아닌 기준 입력과의 상대적인 크기를 크기 변환 요소로 선택하여 기준 입력이 가변하더라도 크기 변환 요소를 다시 조정할 필요가 없게 하였다. 또한 실시간 제어를 위해 퍼지보상기의 추론 부분은 결정표(Look-Up Table) 방식을 사용하였는데, Rutherford[16]는 디지털 제어 시스템에서 이러한 방식의 필요성을 강조하였으며, Li[9]는 정밀한 제어를 위하여 결정표를 거친(coarse)제어와 미세(fine)제어의 두 부분으로 나누는 방법을 제안하였지만 기억용량이 두배로 늘어난다는 단점이 있다. 따라서 본 논문에서는 PID 제어 방식에서의 적절한 계수 조절의 어려움을 극복하기 위해 단지 크기 변환 요소만을 변화시켜서 제어기의 성능을 개선하는 방법을 선택하였다 즉, 기준 입력 근처에서의 정밀한 제

접수일자 : 1995. 6. 15.

1차 수정 : 1995. 9. 4., 2차수정 : 1995. 9. 14

이보희 : 인하대학교 자동화공학과

김진결 : 인하대학교 자동화공학과

어를 위하여 오차가 일정한 범위 내에 들어오게 되면 크기 변환 요소가 거친 제어 모드에서 미세 제어 모드로 변환되게 하였다. 또한 상기에 제시한 제어의 유용성을 증명하기 위해 로봇 시스템을 실제로 제작하여 실험을 하였다. 실험 대상에 사용된 로봇은 5축 수직 다관절 형태를 가지고 있으며, 관절 구동원은 직류 모터를 사용하였다. 또한 부하 실험 및 궤적 추적 실험을 하기 위해 로봇 손(gripper)을 장착할 수 있게 설계하였다.

II. 시스템 구성

1. 시스템 개요

본 연구에 사용된 전체 로봇 시스템의 구성도는 그림1과 같이 온라인 시뮬레이터(On-line Simulator:이하 모의실험기로 약칭함), 제어기, 그리고 5축 다관절 로봇 몸체의 세부분으로 나누어진다.

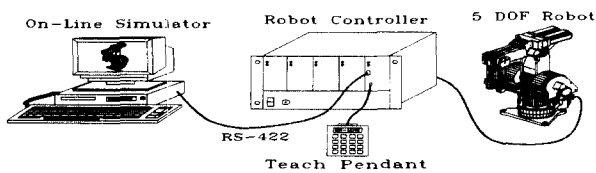


그림 1. 로봇 시스템 구성도.

Fig. 1. Robot System Configuration.

모의실험기는 PC 및 슬롯에 장착되는 RS-422 통신보드로 구성되며, 로봇 언어를 이용한 궤적 계획, 정기구학 및 역기구학 계산을 수행하고 계산된 관절각을 로봇 제어기로 송신하는 기능 및 로봇 제어기에서의 실제로 계측된 관절의 위치각을 수신하여 저장하고 모니터 상에 실제의 로봇 자세를 표시하는 기능을 가지고 있다. 제어기는 입출력 제어기, 양방향 메모리, 위치 제어기, 모터 구동기, 엔코더로 구성되며, 모의실험기에서 계산된 관절각을 받아 모터의 각축 구동 및 위치 제어를 수행하고 엔코더를 이용하여 관절각을 계측하여 모의실험기로 송신하는 기능을 가지고 있다. 로봇 몸체는 허리 구동부, 어깨 구동부, 팔꿈치 구동부, 피치 구동부, 롤 구동부로 구성되어 있으며 구동원은 직류 서보 모터를 사용하였다. 기어의 감속비는 각각 40, 264, 180, 60, 1로 하였다. 또한 로봇손을 장착할 수 있게 설계하여 물체의 이동과 실험에 사용된 펜을 잡을 때 사용하였다. 로봇을 제어할 때는 자체 개발한 로봇 언어를 이용하여 손쉽게 명령할 수 있게 하였으며[17] 직선 보간(PTP) 동작, 원호 보간(CP) 동작, 화면 제어, 기구학 해석 및 데이터 수집에 많은 시간이 소요되기 때문에 IBM PC 486 DX2-66 이상급에서 동작 하였다. 연산된 각 관절의 위치 명령 값은 매 20ms 주기마다 56700 bps의 통신속도를 가진 RS-422 방식으로 로봇 제어기로 송신하고 모의실험기는 각 관절의 위치 측정값을 기구학 해석하여 모니터 상으로 표현하여 실제 동작시 로봇의 운동 상황을 나타내었다. 또한 로봇 제어기는 모의실험기가 연산한 각 관절의 위치 명령 값을 일괄 제어하여야 하는데 기존의 범용 마이크로 프로세서로는 실시간 실현이 불가능 하므로 TI사의 부동소수점 DSP인 TMS320C31을 사용하여 시분할 방식에 의해 각 관절을 제어함으로써 기존의 각 관절의 한 개씩의 마이크로 프로세서가 제어하는 복잡한 방식의 하드웨어 및 소프트웨어 구조를 대폭 간소화하였다[18][19][20].

2. 로봇 설계

실험에 사용된 로봇의 구동원은 금성기전의 회토류계

직류 서보 모터를 사용하였으며 감속기는 유성치차열을 사용한 방식으로 큰 토크 전달 및 소형 경량화 하였다. 각관절부의 구동에 의한 영향을 최소화 하기 위하여 기저부에는 전체 자중의 60%가 집중되도록 설계 하였다. 로봇 팔의 형상은 보통 구동원의 위치에 따라 직접 구동형, 체인에 의한 원격 구동형과 4절 링크에 의한 원격 구동형으로 분류되는데 이 세가지 형상은 장단점을 가지고 있으나 제어의 관점에서 보면 코리올리힘에 의한 토크항의 상쇄와 토크 계산식의 비선형 항을 제거할 수 있는 4절 링크 형상이 보다 큰 장점을 갖는다[21]. 따라서 본 논문에 설계된 4절링크 형상을 택하여 설계 하였다. 각 관절의 축은 중공 및 테이퍼(taper) 형상을 갖도록 하여 미소 편심을 제거하고 배선시 편리하게 했다. 손목 관절은 토크 이득 및 작동 범위의 확장을 고려하여 차동치차열을 이용하여 구동 되도록 설계 하였으며, 구동원은 어깨 및 팔꿈치 관절 및 부하를 줄이기 위해 팔꿈치 관절 상에 위치 하여 타이밍 벨트를 이용하여 원격 구동할 수 있게 했다. 각 구동원 및 계측기의 배선은 중공축 및 커넥터를 이용, 작동 범위에 영향을 주지않고 보수가 용이하도록 배치하였으며 고장진단이 용이하도록 주위의 배선이 모두 집결되도록 설계하였다. 또한 어깨 구동부와 팔꿈치 구동부에는 전자 제동 장치를 장착하여 정전시에 대한 대비를 하였다. 로봇의 전체 사양은 표1과 같다.

표 1. 로봇의 사양.

Table 1. Robot Specification.

형상 Construction	수직다관절(밀폐형) Vertical articulated arm (Closed structure)	반복정밀도 Repeatability	± 0.1 mm		
자유도 Degree of Freedom	5축과 공구부 5axes plus end effector	구동원 Actuators	직류서보모터 DC servo motor		
5.단거리 Reach (Wrist Centre)	508.5mm	중량 Weight	6639kg		
관절작동범위 Joint Operation Range	Base	360° (+178°)	최소중량부 Incremental Movement	Base	17.45 × 10 <sup>-3</sup> rad
	Shoulder	180° (+ 90°)		Shoulder	5.83 × 10 <sup>-3</sup> rad
	Elbow	140° (+ 70°)		Elbow	9.52 × 10 <sup>-3</sup> rad
	Pitch	200° (+100°)		Pitch	39.27 × 10 <sup>-3</sup> rad
	Roll	360° (+178°)		Roll	39.27 × 10 <sup>-3</sup> rad
성적최대속도 Maximum Speed	Base	200 deg/sec	가변하중 Pay Load	유효 Effective	30kg (66lbs)
	Shoulder	66.73 deg/sec			최대 Maximum
	Elbow	109.09 deg/sec			
	Pitch	-150 deg/sec			
	Roll	150 deg/sec			

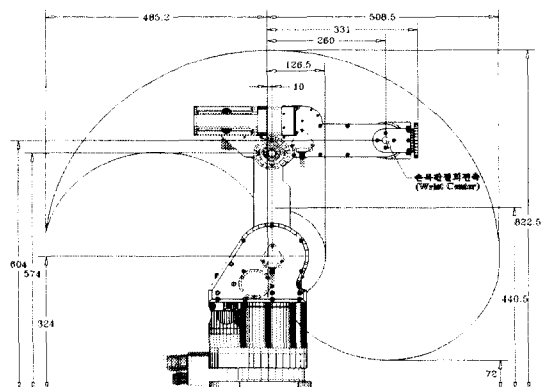


그림 2. 로봇의 작업 공간.

Fig. 2. Workspace of Robot.

허리 구동부는 도립형상으로도 작업이 가능하도록 테이퍼 롤러 베어링을 이용하여 축방향 하중에 견딜 수 있도록 하였으며, 팔꿈치 관절에 오프셋을 주어 작업 반경이 증대 되도록 하였다. 최대 반경은 약 1m이며 작업영역은 그림2

와 같다

로봇의 기준 좌표계는 그림3과 같이 설정하였으며 이때 D-H 표시법에 의한 좌표값은 표2와 같다.

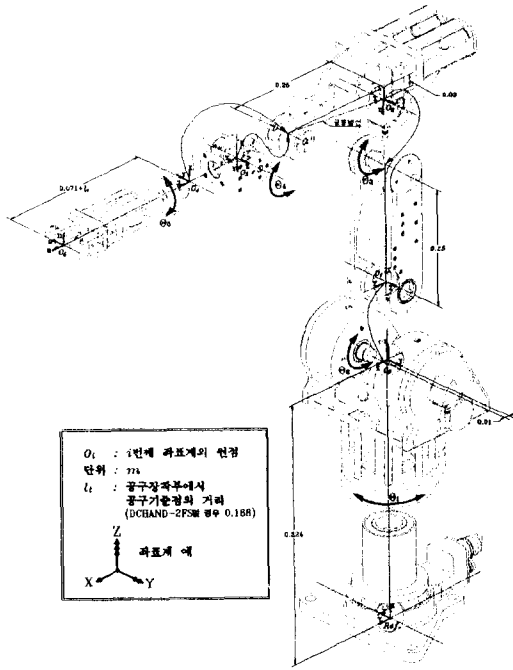


그림 3. 로봇 기준 좌표계.  
Fig. 3. Robot Reference Coordinates.

표 2. D-H 변수.

Table 2. D-H Parameters.

Joint i	$\theta_i$	Initial Joint Value (단위:rad)	$d_i$ (단위:m)	$a_i$ (단위:m)	$\alpha_i$ (단위:rad)	Joint Range (단위:degree)
하리 구동부	$\theta_1$	0	0	0.01		356 (±178°)
어깨 구동부	$\theta_2$	$-\pi/2$	0	0.25	0	180 (±90°)
팔꿈치 구동부	$\theta_3$	$\pi/2 - 0.119428926$	0	0.2517935662	0	140 (±70°)
피치 구동부	$\theta_4$	0.119428926	0	0	$-\pi/2$	200° (±100°)
물 구동부	$\theta_5$	0	0.071+it	0	0	356° (±178°)

로봇 관절각을 기준좌표계에 대하여 공구좌표계로 표현하면 (1)과 같은 동차변환 행렬  ${}^{base}A_{tool}$  를 구할 수 있다.

$${}^{base}A_{tool} = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

(1)에서  $[n, s, a]$  는 공구 좌표계의 방향 성분이며  $p_x, p_y, p_z$  는 기준 좌표계에 대한 위치 벡터 성분으로 (2)와 같이 표현된다.

$$\begin{aligned} n_x &= C_5 C_1 C_{234} + S_1 S_5, & s_x &= C_5 S_1 C_{234} - C_1 C_{234} S_5, & a_x &= -C_1 S_{234} \\ n_y &= C_5 C_1 C_{234} - C_1 S_5, & s_y &= -C_1 C_5 - S_1 S_{234} S_5, & a_y &= -S_1 S_{234} \\ n_z &= S_{234}, & s_z &= S_{234} S_5, & a_z &= -C_{234} \\ p_x &= 0.01 C_1 - (0.071 + l_i) C_1 S_{234} + 0.25 C_1 C_2 + 0.2517935662 C_1 C_{234} \\ p_y &= 0.01 S_1 - (0.071 + l_i) S_1 S_{234} + 0.25 C_2 S_1 + 0.2517935662 S_1 C_{234} \end{aligned} \quad (2)$$

$$p_z = -0.2517935662 S_{234} - (0.071 + l_i) C_{234} - 0.25 S_2$$

이때  $S_i$  는  $\sin \theta_i$  를,  $C_i$  는  $\cos \theta_i$  를 의미하며  $l_i$  는 공구 부착면에서 공구 중심까지의 거리이고  $S_{ij}$  과  $S_{ijk}$  는  $\sin(\theta_i + \theta_j)$ ,  $\sin(\theta_i + \theta_j + \theta_k)$  이다. 또한, 각 관절의 공구좌표계에 대한 관절 회전각은 아래와 같이 주어진다.

$$\begin{aligned} \theta_1 &= \text{Atan2}(p_x, p_y) \\ \theta_2 &= \text{Atan2}(\alpha, \beta) \\ \theta_3 &= \cos^{-1}(\gamma / 0.1258967831) \\ \theta_4 &= \text{Atan2}(a_x C_1 + a_y S_1, a_z) - \theta_2 - \theta_3 \\ \theta_5 &= \text{Atan2}(n_x C_1 - n_y S_1, s_x C_1 - s_y S_1) \end{aligned} \quad (3)$$

(3)에서  $\alpha, \beta, \gamma$  는 다음과 같다.

$$\begin{aligned} \alpha &= 0.2517935662 S_3 [ (p_x C_1 + p_y S_1 - 0.01) - (0.071 + l_i)(a_x C_1 + a_y S_1) ] \\ &\quad + 0.2517935662 C_3 [ p_z - (0.071 + l_i) a_z ] \\ &\quad + 0.25 [ p_z - (0.071 + l_i) a_z ] \\ \beta &= 0.2517935662 C_3 [ (p_x C_1 + p_y S_1 - 0.01) - (0.071 + l_i)(a_x C_1 + a_y S_1) ] \\ &\quad + 0.25 [ (p_x C_1 + p_y S_1 - 0.01) - (0.071 + l_i)(a_x C_1 + a_y S_1) ] \\ &\quad - 0.2517935662 S_3 [ p_z - (0.071 + l_i) a_z ] \\ \gamma &= p_z^2 + (p_x C_1 + p_y S_1 - 0.01)^2 \\ &\quad + (0.071 + l_i)^2 - 0.1259 + 2(0.071 + l_i) \cdot [ p_z C_{234} + (p_x C_1 + p_y S_1 - 0.01) S_{234} ] \end{aligned} \quad (4)$$

### 3. 로봇 제어기 구성

로봇 제어기는 그림4와 같이 크게 중앙 연산 처리부(CPU), 모터 구동부, 엔코더의 세부분으로 나눈다.

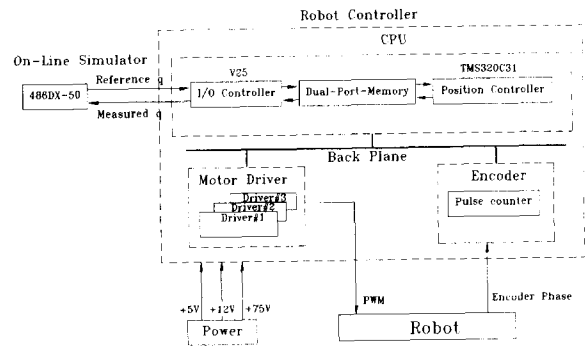


그림 4. 로봇 제어기 블록선도.  
Fig. 4. Block Diagram of Robot Controller.

첫째, CPU는 모의실험기에서 전달된 각 관절의 위치 명령값을 해독하여 모터의 구동각으로 환산하여 양방향 메모리로 전송하여 위치 제어를 수행시키고 교시상자(teach pendant)를 통한 사용자의 명령을 해독하고 실행하는 기능을 가지며 로봇 동작에 관련된 입출력을 관장한다. CPU는 입출력제어기, 양방향 메모리, 위치 제어기로 구성되며 각 부분의 특징은 다음과 같다. 입출력 제어기는 NEC사의 V25 원칩 CPU를 사용하였으며, 모의 실험기로부터 각각의 모터 위치 명령값을 받아서 양방향 메모리에 전달하는 기능과 양방향 메모리에서 각각의 모터 위치 측정값을 주 컴퓨터로 전달하는 동시에 교시상자의 입,출력을 처리하여 수동으로 로봇트를 조작할 수 있는 기능을 가지게 하였다. 양방향 메모리는 CY7C132(Cypress, 25ns)를 사용하였으며, 입출력 제어기와 위치 제어기의 정보 전달 기능과 각각의 모터 위치 명령 및 위치 측정값을 상호 저장에 사용된다. 위치 제어기는 DSP를 사용하며 시분할 방식으로 5축의 위치 제어를 수행하여 하드웨어의 간략화 및 소프트웨어의 효

율화를 기하였다. 제어 수행 주기는 전체를 1ms 단위로 하였으며 이때 각관절의 회전각을 측정하여 양방향 메모리로 전달하는 기능도 가지고 있다. 각축을 구동하는 위치 제어기의 전체 구성은 그림5와 같다.

위치 제어기에서 출력한 제어 값은 D/A변환기를 거쳐 아날로그 값으로 변경한 후 모터를 구동하며 실제의 위치 값을 엔코더를 통하여 계수한후 모의실험기에서 전달된 명령치와 비교하고 위치제어기는 본논문에서 제시한 알고리즘을 이용하여 제어값을 환산하고 전과정을 반복한다. 이때 모터의 부하에 대한 토크를 보상하기 위해 하드웨어로 내부에 전류 보상부품을 두었다. 본 논문에서 사용한 모의 알고리즘은 C-언어[21]를 사용하였으며 부분적으로 속도가 요구되는 부분과 하드웨어의 제어에 밀접한 관계가 있는 시스템 설정과 인터럽트 수행하는 부분에는 어셈블러를 삽입시켜 호환성이 있고 실장이 편리하며 디버깅이 용이하게 하였다. 둘째, 모터구동부는 연산된 제어 값을 실제 모터를 구동하기 위한 신호로 변환하고 구동시키는 부분이며 전체 구성은 그림5와 같다.

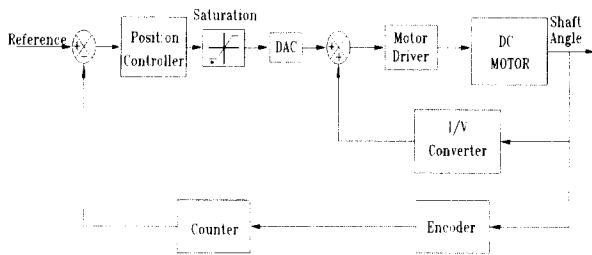


그림 5. 위치 제어 시스템 블록선도.  
Fig. 5. Block Diagram of Position Control System.

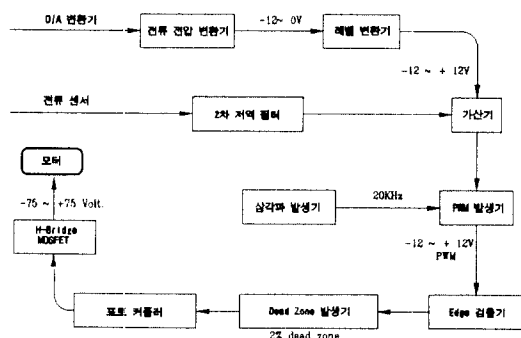


그림 6. 모터 구동기 블록선도.  
Fig. 6. Block Diagram of Motor Driver.

CPU로부터 연산된 제어 신호는 D/A 변환기와 전류 전압 변환기를 거쳐 모터 지령 아날로그 전압으로 변경된다. 이 전압 지령치는 모터구동시 전력 소모를 줄이기 위하여 펄스폭 변조(pulse width modulation)변환기를 거쳐 구동 펄스 폭으로 변증된다. 모터를 직접 구동하는 스위칭방식은 전력 구동 소자인 MOSFET를 사용한 혼합 구동 방식(H-bridge)을 사용했으며, 고 전력을 단락하는데 발생하기 쉬운 노이즈가 디지털회로에 환류되는 것을 방지하기 위해 각각의 구동부를 전기적으로 격리 시켰다. 또한 구동시 MOSFET의 상측과 하측 부분이 동시에 단락 되는 것을 방지하기 위해 상(edge)검출기와 데드타임 발생기를 사용하여 약간의 지연이 일어나게 하였다. 또한 MOSFET가 고속으로 스위칭함으로써 발생하는 EMI 노이즈를 줄이고 소자 보호를 위해 각 구동부에는 콘덴서와 다이오우드, 저항을 병렬로 연결한 스너버(snubber)를 장착하였다. 셋째, 엔코더는

직류 서보모터의 종단부에 장착되어 회전 위치각을 측정하는 검출기이다. 로봇 구동부에 장착된 엔코더는 회전당 1000개의 펄스를 발생시키며 회전 방향에 따라 90°의 위상차를 갖는 A상과 B상의 2가지 펄스를 발생하고 1회전 중에 절대 위치를 검출하기 위한 영점 신호인 C상을 출력하는 증분형을 사용했다. 검출된 정보는 회전 방향 및 회전속의 회전 각도이며 보다 정밀한 위치 회전속도 정보를 얻기 위해서 분해능을 4배로 늘려 주는 회로를 설계 제작하여 정밀도를 높혔고, 엔코더로 부터 출력된 펄스는 바이폴라 읽기 전용 메모리를 거쳐 에지를 검출하여 계수용 펄스를 만든후 카운터 IC(Intel, 82C54)를 사용하여 계수하고 DSP에서는 그 값을 읽도록 하였다.

4. 소프트웨어 구조

전체 프로그램은 PC에서 동작하는 실시간 모의실험기와 로봇 제어기에서 동작하는 구동부 위치 제어 프로그램의 두 부분으로 구성된다. 첫째, 모의실험기는 로봇의 원하는 궤적을 표현하는 언어 형태의 지령치를 해독하여 관절이 움직일 지령치를 발생하고, 실제로 로봇과 연결되지 않고도 모의실험을 할 수 있게 되어 있어서 궤적 계획시 발생할 수 있는 오차를 줄일 수 있게 구성 하였다. 또한 각 관절의 실측치를 계속하여 모니터에 표시 함으로써 실제 관절의 움직임을 알 수 있게 하였고 화일 작업을 통하여 관절의 움직임을 분석할 수 있게 하였다. 모의 실험기에서 관절을 구동하는 전체 흐름도는 그림7과 같다.

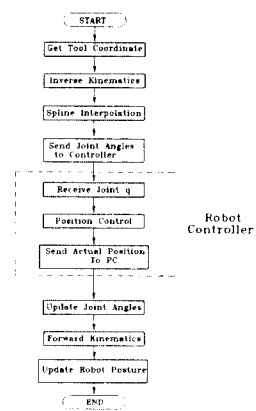


그림 7. 전체 프로그램 흐름도.  
Fig. 7. Flow Chart of Total Program.

로봇 손이 경유해야 할 경유점을 입력하면 역기구학 해석에 의해 관절의 구동량을 계산하고 5차 보간법에 의한 궤적 생성을 한 후 얻어진 관절 구동각을 주 제어기로 전달한다. 입출력 제어기는 전달된 구동각을 수집하여 모터가 동작할 수 있는 값으로 바꾸어 위치 제어기로 전달한다. 위치 제어기는 제안된 피지제어 알고리즘을 이용하여 정해진 시간에 정확하게 위치 제어한 후 모터 종단에 붙어 있는 엔코더의 값을 읽어 다시 모의실험기에 전달하고 로봇제어기는 데이터를 저장하는 동시에 순 기구학 해석을 통하여 로봇의 손 좌표계로 변환한 후 모니터에 각 관절값 및 로봇 손의 궤적을 표시한다. 둘째, 위치 제어 프로그램은 관절의 구동부를 담당하는 프로그램으로 관절값을 일정값으로 유지하게 한다. 정해진 시간에 정확히 제어하기 위하여 인터럽트를 이용하여 정확히 1ms 간격으로 수행하게 하였다. 전체 흐름도는 그림8과 같다.

먼저 관절 구동 명령값의 유무를 확인한 후 명령값이 있으면 읽어서 모터 제어 알고리즘의 위치 기준치에 복사한다. 위치 제어 알고리즘은 모터의 현재 위치값을 측정 한 후

기준치와 비교하여 오차와 오차의 변화치를 구하고 퍼지 보상 알고리즘을 수행하여 제어값을 결정한다. 연산된 제어값은 D/A변환부를 거쳐 모터 제어를 행하고 1ms후 다시 모터의 위치각을 계측후 경신한 후 위치제어 알고리즘을 반복한다. 여기서 위치 제어 알고리즘은 정확한 시간 간격으로 수행하기 위해 인터럽트 방식으로 하였다. 또한 계측된 모터의 위치각은 20ms 주기로 모의실험기로 보내져 구동부의 현재상태 값을 정확히 알 수 있게 한다.

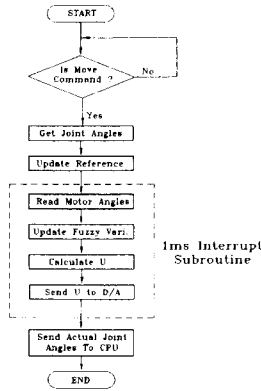


그림 8. 위치 제어 프로그램 흐름도.  
Fig. 8. Flow Chart of Position Control Program.

III. 제어기 구성

본 논문에서 사용된 퍼지 보상 제어기는 그림9와 같이 PID 제어기를 이용한 기존의 위치 제어 시스템에 퍼지 논리 제어기를 합성하는 형태를 가지고 있다[23].

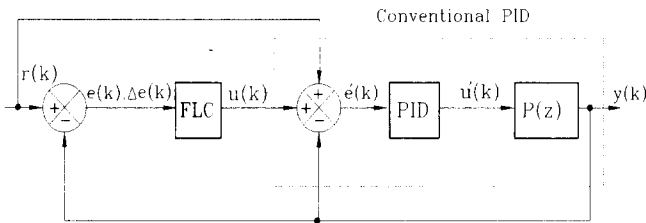


그림 9. 퍼지보상 PID제어기의 블럭선도.  
Fig. 9. Block Diagram of Fuzzy-compensated PID Controller.

그림8 에서 각 신호들은 다음과 같이 표현된다.

$$\begin{aligned}
 u(k) &= FLC[ e(k), \Delta e(k) ] \\
 e'(k) &= u(k) - y(k) + r(k) \\
 u'(k) &= u'(k-1) + K_1 \cdot e'(k) + K_2 \cdot e'(k-1) + K_3 \cdot e'(k-2) \\
 y(k) &= P(z) \cdot u'(k)
 \end{aligned}
 \tag{5}$$

여기서  $P(z)$ 는 플랜트이고  $e'(k)$ 는 오차이며,  $u'(k)$ 는 PID 제어기 출력으로  $K_p$ 와  $T$ 를 비례 상수와 샘플링 주기,  $T_I$ 와  $T_D$ 는 각각 적분 시간과 미분시간을 나타낼때  $K_1 = K_p(1 + T_D/T)$ ,  $K_2 = -K_p(1 + 2T_D/T - T/T_I)$ ,  $K_3 = K_p \cdot T_D/T$ 로 표현된다.  $FLC[ e(k), \Delta e(k) ]$ 는 퍼지 논리 제어기 이며 구조는 그림10과 같다

그림10 에서  $R_1-R_n$ 과 같은 제어 규칙은 IF-THEN 형태로 표현되며 추론은 Mamdani[22]가 제시한 min-max 연산

법을 이용 하였다. 퍼지 언어 변수(linguistic variable)는 NB(negative big), NM(negative medium), NS(negative small), ZO(zero), PS(positive small), PM(positive medium), PB(positive big) 의 7단계로 선택하였다. 따라서 퍼지 집합은 전체 집합  $U$ 를  $U = \{-6, -5, -4, -3, -2, -1, 0, 1, 2, 3, 4, 5, 6\}$ 와 같이  $[-6 \sim +6]$ 사이의 정수로 이산화한 13개의 양자화 준위들을 갖게 된다. 비퍼지화는 가장 널리 사용되고 있는 면적중심법(center of area method)을 사용하였다[13]. 또한 그림 10의  $U(k)$ ,  $E(k)$ ,  $\Delta E(k)$ 는 (6)과 같다.

$$\begin{aligned}
 U(k) &= f[ E(k), \Delta E(k) ] \\
 E(k) &= q[ C_1 \cdot e(k) ] \\
 \Delta E(k) &= q[ C_2 \cdot \Delta e(k) ]
 \end{aligned}
 \tag{6}$$

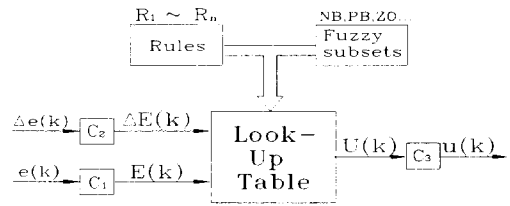


그림 10. FLC의 블럭선도.  
Fig. 10. Block Diagram of FLC.

이때  $f$ 는 퍼지관계를 나타내며,  $q$ 는 양자화 함수를 나타낸다. (6)과 같이 추정된 출력 오차  $e$ 와 출력 오차의 변화량  $\Delta e$ 에 적당한 크기 변환 요소  $C_1$ 과  $C_2$ 를 곱함으로써 전체 집합의 양자화 준위  $E(k)$ ,  $\Delta E(k)$ 로 크기 변환 시킬 수 있다.  $C_1$ ,  $C_2$ 는 출력 오차의 최대치와 출력 오차에 대한 양자화 준위의 최대치, 출력 오차 변화량의 최대치와 출력 오차 변화량에 대한 양자화 준위의 최대치의 비에 의해 구하며 이에 따른 결정표는 표3와 같다.

표 3. 결정표.  
Table 3. Look Up Table.

		출력오차의 변화량(ΔE)												
		6	5	4	3	2	1	0	1	2	3	4	5	6
출력 오차 (E)	-6	-6	-6	-6	-6	-6	-6	-6	-5	4	3	-2	-1	0
	-5	-6	-6	-6	-6	-6	-5	-5	-4	-4	-2	-1	0	1
	-4	6	6	6	6	6	5	4	4	4	-2	0	1	2
	-3	6	6	6	5	5	-1	3	2	2	0	1	2	3
	2	6	6	6	5	4	3	2	-1	0	1	2	3	4
	1	6	5	5	3	2	1	1	1	3	3	4	4	5
	0	-6	-5	-4	-2	0	0	0	3	6	6	6	6	6
	1	-5	4	3	1	0	1	3	4	5	5	6	6	6
	2	4	3	2	1	0	3	6	5	4	5	6	6	6
	3	-3	-2	1	0	2	4	6	5	5	5	6	6	6
	4	-2	-1	0	2	4	5	6	6	6	6	6	6	6
	5	1	0	1	2	1	5	6	6	6	6	6	6	6
	6	0	1	2	3	4	5	6	6	6	6	6	6	6

양자화된 제어기 출력  $U(k)$ 는 크기 변환 요소  $C_3$ 를 곱함으로써 실제 적용 가능한 비퍼지 제어기 출력  $u(k)$ 를 (7)과 같이 구할 수 있다. 여기서  $dq$ 는 비양자화 함수를 나

타내며  $C_3$ 는 제어 입력에 대한 양자화 준위의 최대치와 제어 입력의 최대치의 비에 의해 구한다.

$$u(k) = dq [ C_3 \cdot U(k) ] \quad (7)$$

또한 그림11과 같이 기준 입력 주변 영역을 모터축의 위치가 기준 입력 부근 영역 ( $|e(k)| < M$ )으로 들어왔을 경우에 크기 변환 요소만을 I에서 II로 변환될 수 있게 하여 두개 영역으로 제어함으로써 보다 효율적인 제어를 하였다.

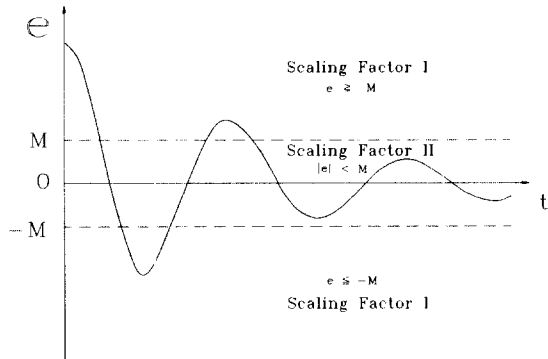


그림 11. 변환 비율 선택.  
Fig. 11. Selection of Scale Factor.

**IV. 실험 결과**

본 논문에서 제안한 퍼지 보상 PID 제어 기법을 로보트 제어기의 위치 제어 시스템에 적용하여 실험을 하였다. 그림12는 본 실험에서 사용한 다관절 로보트, 제어장치 및 컴퓨터의 전체 모습이다.

구동부에 사용된 모터는 100 watt 직류 서보모터이며 자세한 사양은 표4와 같다.



그림 12. 실험 장치.  
Fig. 12. Experimental Equipments.

표 4. 직류 서보모터 사양.  
Table 4. DC Servo-motor Specification.

Parameters	Symbios	Values	Units
Moment of Inertia	J	0.28	g-cm-sec <sup>2</sup>
Mechanical Time Constant	$\tau_m$	4.01	msec
Electrical Time Constant	$\tau_e$	0.3	msec
Back EMF Constant	$K_b$	22.2210%	V/K rpm
Torque Constant	$K_t$	2.17	kg cm
Armature Inductance	$L_a$	2.0	mH
Armature Resistance	$R_a$	6.6	$\Omega$
Power Rate	P	100	Watt

PID제어기의 계수는 Ziegler-Nichols의 임계값을 이용한 동조계수 결정 방법으로 구하였으며 표5와 같이 주어진다.

표 5. PID 계수.  
Table 5. PID Parameters.

계수 구동부	$K_p$	$K_i$	$K_d$
허리 구동부	8.5	0.052	0.062
어깨 구동부	4.2	0.1	0.092
팔꿈치 구동부	9.8	0.122	0.122
피치구동부	8.7	0.062	0.045
롤 구동부	6.3	0.069	0.023

퍼지 보상 PID제어기에서 퍼지 보상의 크기변환요소는 기준입력과의 상대적인 크기로 결정되어지며 표6과 같다. 미세 제어로 변환되는 오차 범위의 크기는 기준 입력의 2.5%로서  $M = |r \cdot 0.025|$ 으로 선택하였다.

표 6. 변환 계수.  
Table 6. Scale Factor.

계수 구동부	미세제어			저원제어		
	$C_1$	$C_2$	$C_3$	$C_1$	$C_2$	$C_3$
허리 구동부	0.006	0.0001	0.001	0.0004	0.0075	633.33
어깨 구동부	0.005	0.020	0.0001	0.002	0.005	633.33
팔꿈치 구동부	0.001	0.050	0.001	0.001	0.005	633.33
피치구동부	0.001	0.050	0.005	0.001	0.001	633.33
롤 구동부	0.001	0.050	0.010	0.001	0.001	633.33

본 논문에서 제안한 제어기법의 효율성을 보이기 위해 로보트 시스템에 각각  $10^\circ$ 의 계단 함수를 인가한 후 응답을 보았다. 그림 13과 같이 정착 시간(settling time)은 허리구동부, 어깨 구동부, 팔꿈치 구동부, 피치 구동부, 롤 구동부 각각 0.17초, 0.28초, 0.24초, 0.06초, 0.05초로 비교적 빠르고 안정된 응답을 얻었다. 정상 상태 오차는 각각  $-0.0447^\circ$ ,  $+0.0067^\circ$ ,  $-0.0955^\circ$ ,  $+0.02^\circ$ ,  $+0.04^\circ$ 로 정밀한 제어임을 보여준다.

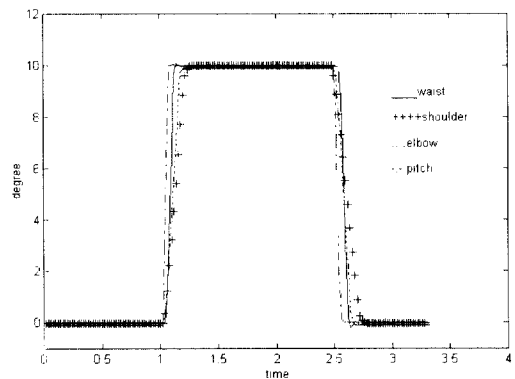


그림 13.  $10^\circ$  계단 응답.  
Fig. 13. Step Response with 10 Degree.

이번에는 로보트 손을 장착하고 사각형 그림을 그리는 실험을 하였다. 그리는 순서는 x, y 각각 10 cm 사각형 궤적을 A부터 F까지 그리는 실험을 하였다. 여기서 A, B, C는 사각형을 그리기 위해 로보트손이 지정된 곳으로 이동할 때의 x, y 방향의 궤적을 나타냈다.

그림14와 같이 각모서리 부분은 약간의 진동이 있으며 전체적으로는 주어진 값을 잘 추적함을 알 수 있다. 그러나 이때 궤적값은 관절의 모터축 종단부의 엔코더 값을 계측하

여 모의실험기로 보낸 후 순기구학을 풀어 공구좌표계로 변환된 값이므로 실제 기구에서 나타나는 오차 성분 즉, 백래쉬등과 같은 성분이 고려되지 않았으므로 실제 로봇 손에 펜을 장착하여 그려보았다.

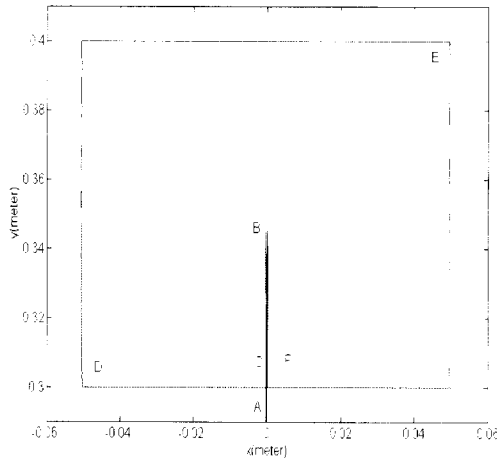


그림 14. 사각형 궤적 추적.  
Fig. 14. Tracking of a Square Trajectory.

그런 결과 각 모서리에는 백래쉬가 존재함을 알 수 있고 중력 방향의 영향을 받고 있지 않은 허리 구동부에서 방향의 전환시 비교적 큰 백래쉬가 존재함을 알 수 있다. 이러한 백래쉬는 비선형 제어를 설계하여 적절하게 없앨 수 있으나 본 실험에서는 백래쉬가 일어나는 값을 실험으로 측정하여 보정하는 방법에 의해 보상하였다. 로봇 손목관절에 펜을 장착하고 지면에 종이를 놓고 사각형 추적 명령을 준 후 그린 실제의 궤적은 그림15와 같다.

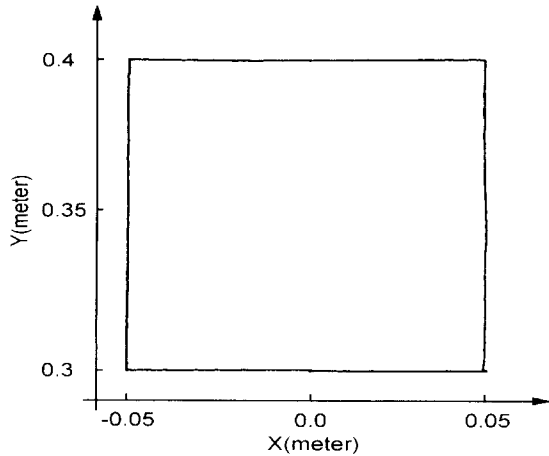


그림 15. 사각형 궤적 실제 추적.  
Fig. 15. Actual Tracking of a Square Trajectory.

그림 14의 결과처럼 주어진 궤적을 잘 추적함을 알 수 있다. 이번에는 지름 10cm인 원을 그리는 실험을 하였다. 그림16의 (a)는 3차원으로 표현한 로봇손의 궤적 추적 그림이며 (b)는 2차원 추적 그림이고 (c)는 x 방향으로 오차 (d)는 y 방향으로 오차이다. 그림에서 알 수 있듯이 전체 오차는 x방향으로는 2.5mm 이하 y방향으로는 5mm이하의 오차를 가지고 있다. 이때의 각 축의 오차에 대한 궤적은 그림 17의 (a)-(d)와 같다. 롤 구동부는 명령을 추적하는데 필요치 않으므로 생략하고 5초까지의 궤적은 그리는 위치까지 이동할 때의 궤적이다. 실험 결과 모든 관절부의 최대 오차

가 1° 미만으로 정밀한 제어가 됨을 알 수 있다.

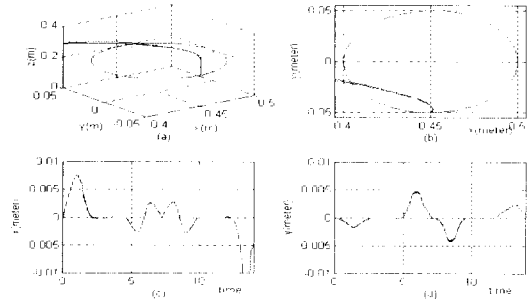


그림 16. 지름 0.01m 인 원추적 궤적.  
(a) 3차원 궤적 (b) 2차원 궤적  
(c) x방향 오차 (d) y방향 오차  
Fig. 16. Circle Tracking Trajectory with 10cm Diameter.  
(a) 3-D Tracking (b) 2-D Tracking  
(c) x Direction Error (d) y Direction Error

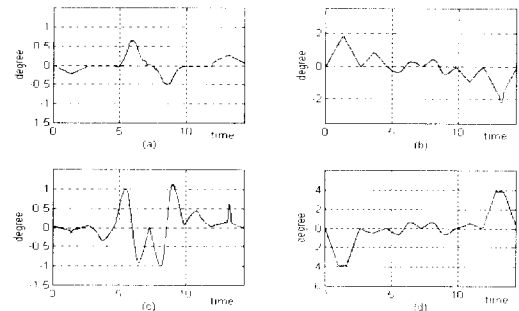


그림 17. 원 추적 오차.  
(a) 허리구동부 (b) 어깨구동부  
(c) 팔꿈치구동부 (d) 피치구동부  
Fig. 17. Tracking Error of a Circle Trajectory.  
(a) Waist (b) Shoulder (c) Elbow (d) Pitch

**V. 결론**

본 논문에서는 로봇트를 실제 제작 하면서 고려된 설계 방법을 다루었다. 제작된 로봇트의 제어 방법으로는 기존 PID 제어기에 크기 변환 요소를 가진 퍼지 보상 제어를 첨가한 기법을 사용하고 다양한 형태의 작업 명령을 주고 응답을 분석해 봄으로써 퍼지 보상 PID 제어기의 효율성 및 로봇트의 정교함을 입증 하였다. 앞으로 연구할 분야는 제작된 로봇트에 보다 포괄적인 실험을 하기 위해 발달효과 장치를 다양한 형태로 만들고 힘 제어 및 역학이 고려된 제어 기법의 연구가 필요하며, 특히 신경회로망과 같은 지능 제어의 적용이 요구된다.

**참고 문헌**

- [1] H. Asada and J. J. E. Slotine, *Robot Analysis and Control*, John Wiley & Sons Inc, 1986.
- [2] F. L. Lewis and C. T. Abdallah, *Control of Robot Manipulator*, Macmillan Publishing Company, 1993.
- [3] S. Banerjee and P. Y. Woo, "Fuzzy Logic Control of Robot Manipulator," *Proc. IEEE Conf. Control Applications*, Vancouver, Canada, pp. 87-88, 1993.
- [4] H. B. Kuntze and M. Sajidman, "A Fuzzy Logic Concept for Highly Fast and Accurate Position Control of Industrial Robots," *IEEE Int. Conf. Robotics and Automation*, vol. 1, pp. 1184-1190, 1995.
- [5] Y. F. Li and C. C. Lau, "Development of fuzzy algorithms for servo systems," *IEEE Contr. Syst. Mag.*, vol. 9, no. 3, pp. 65-72, 1989.

- [6] G. Tao and P. V. Kokotović, "Adaptive control of plants with unknown dead-zones," *IEEE Trans. Auto. Contr.*, vol. 39, no. 1, 1994.
- [7] J. E. Slotine and W. Li, *Applied Nonlinear Control*, Englewood Cliffs : Prentice-Hall, 1991.
- [8] L. A. Zadeh, "Outline of a new approach to the analysis of complex systems and decision processes," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-3, pp. 28-44, 1973.
- [9] K. L. Tang and R. J. Mulholland, "Comparing fuzzy logic with classical controller designs," *IEEE Trans. Syst. Man Cybern.*, vol. SMC-17, no. 6, Nov./Dec. 1987.
- [10] P. J. Macvicar-Whelan, "Fuzzy sets for man-machine interaction," *Int. J. Man-Mach. Stud.*, vol. 8, pp. 687-697, 1976.
- [11] S. Tzafestas and N. P. Papanikolopoulos, "Incremental fuzzy expert PID control," *IEEE Trans. on Indus. Elec.*, vol. 37, no. 5, pp. 365-371, Oct. 1990.
- [12] T. J. Procyk and E. H. Mamdani, "A linguistic self-organizing process controller," *Automatica*, vol. 15, pp. 15-30, 1979.
- [13] J. H. Kim, S. W. Lee, K. C. Kim and E. K. P. Chong, "Fuzzy precompensation of PID controllers," *IEEE Conf. Cont. Applic.*, pp. 183-188, 1993.
- [14] J. H. Kim, J. H. Park, S. W. Lee and E. K. P. Chong, "Control of systems with deadzones using PD controllers with fuzzy precompensation", *Proc. of the 1993 Inter. Symp. on Intelligent Control*, pp. 451-456, August, 1993.
- [15] M. Y. Shieh and T. H. S. Li, "Integrated fuzzy logic controller design," *IECON 93*, vol. 1, pp. 279-284, 1993.
- [16] D. A. Rutherford and G. C. Bloore, "The implementation of fuzzy algorithms for control," *Proc. IEEE*, pp. 572-573, 1976.
- [17] 한솔(주), LPR593DS 사용자 길잡이, 1995
- [18] Texas Instrument, *TMS320C3X User's guide*, 1993
- [19] Rulph Chassaing, *Digital signal processing with C and the TMS320C30*, John Wiley & Sons, 1992
- [20] Texas Instrument, *TMS320 Floating-point DSP optimizing C compiler user's guide*, 1993
- [21] Mark W. Spong and M. Vidyasagar, *Robot dynamic and control*, John Wiley & Sons, pp. 177-180, 1989
- [22] Ronald R. Yager and Dimitar P. Filev, *Essential of fuzzy modeling and control*, Wiley Interscience, pp. 265-311, 1994.
- [23] 김진걸, 이혁재, "개선된 퍼지보상 PID제어기 설계에 관한 연구," 한국정밀학회논문지, 12권 48호, pp. 111-118, 1995.



이 보 희

1963년 4월 20일생. 1985년 2월 ~ 1987년 12월 삼성반도체 통신연구소. 1987년 12월 ~ 1990년 7월 쌍용컴퓨터시스템 연구소. 1992년 7월 인하대학교 자동화 공학과 졸업(공학석사). 1993년 2월 ~ 현재 인하대학교 자동화공학과 박사과정. 주관심분야 : Robot Control, Walking Robot



김 진 걸

1955년 7월 4일생. 1978년 2월 서울대학교 전기공학과(공학사). 1984년 8월 미국 IOWA대 전기 및 컴퓨터공학과(공학석사). 1986년 2월 미국 IOWA대 수학과(이학석사). 1988년 5월 미국 IOWA대 전기 및 컴퓨터공학과(공학박사). 1988년 8월 ~ 인하대학교 자동화공학과 교수