

다관절 휴머노이드 로봇 팔의 제어를 위한 시간지연 제어기의 FPGA 구현 및 실험

FPGA Implementation and Experiment of a Time-Delayed Controller for Humanoid Robot Arm Control

이운규, 전효원, 정슬*
(Woonkyu Lee, Hyo-won Jeon, and Seul Jung)

Abstract : In this paper, a time-delayed controller for position control of humanoid robot arms is designed and implemented on a field programmable gate array(FPGA) chip. The time-delayed control algorithm is simple to implement, and robust to reject disturbances. The time-delayed control method uses the one sample time-delayed previous information to cancel out uncertainties in the system. Since the sampling time is so fast with the current hardware technology, the time-delayed controller can be implemented. However, inertia values should be correctly estimated to have the better performance. The position tracking tasks of humanoid robot arms are tested to compare performances of several control algorithms including the time-delayed controller.

Keywords : FPGA, time-delayed control, robot manipulator

I. 서론

최근에 로봇의 꽃이라 불리는 휴머노이드 로봇에 대한 연구가 활발하다. 일본 아시모의 출현 이후, 국내에서도 HUBO의 출현으로 휴머노이드 로봇에 대한 연구는 절정에 달하고 있다. 오랫동안 해결하지 못한 이족 보행을 완성하는가 싶더니 이제는 이족으로 뛰는 로봇이 나왔다. 하지만, 상대적으로 휴머노이드 로봇 팔에 대한 연구는 활발하지 않다. 그 이유는 산업 로봇에 대한 연구가 오랫동안 진행되어 왔고, 외팔로봇에 대한 연구도 이미 많이 진행되어 온 상태이기 때문이다.

휴머노이드 로봇 팔은 크기와 구동의 제약조건이 있어 기존의 산업로봇과는 구조가 다르다. 상대적으로 작은 팔의 구조는 설계 및 제작을 어렵게 만들 뿐만 아니라 제어도 쉽지 않다. 휴머노이드 로봇 팔은 작은 공간에 구동기가 장착되어야 하고, 여러 개의 관절을 제어해야 하는 제어기가 장착되어야 하기 때문이다. 산업로봇 팔의 경우에는 제어기만해도 독립적으로 큰 부피를 차지하여 고정식으로 사용되고 있다. 휴머노이드의 경우에 있어서는 이동식이므로 제어기도 임베디드 형태로 설계되어야 한다.

임베디드 제어기의 경우 DSP나 MCU를 사용하는 것이 일반적이다. 하지만, 휴머노이드와 같이 제어할 관절이 많은 경우는 각 관절의 모터를 제어할 출력 포트가 충분해야 한다. 두 팔 로봇의 경우 머리에 2개 관절을 포함해 모두 14관절을 제어해야 한다. 따라서, 제어기로 확장성이 우수한 FPGA칩을 사용하는 것이 타당하다. 이는 복잡한 디지털 회로를 구성하는 대신 FPGA 칩 하나로 구성할 수 있으며, 사용자 필요에 따라 특수 기능을 가진 디지털 소자로 설계하여 사용할 수도 있기 때문이다[1,2].

간단한 제어 알고리즘으로는 PD 또는 PID제어 알고리즘이 보편적이다. PID 제어방식은 간단하면서도 성능이 우수하

로 많이 사용된다. 하지만 다관절 로봇의 경우처럼 비선형이면서 커풀된 시스템의 경우에는 정확한 제어를 기대하기 어렵다. 이는 로봇의 불확실성이 존재하여 제어를 어렵게 한다. 모델 기반의 계산 토크방식은 커풀된 비선형 시스템을 디커플시켜서 제어하도록 하는데 모델의 정확한 평가에 따라 시스템의 성능이 달라진다. 또한 모델을 연산하는데 필요한 시간도 고려하여야 한다.

그러므로, 본 논문에서는 시간지연제어기를 사용해서 로봇의 위치 추종 성능을 높이고자 한다. 시간지연제어기는 로봇 팔의 제어나 모터에 적용되어 그 성능을 이미 검증한 바 있다[3-8]. 시간 지연 제어방식은 매우 간단하여 실시간으로 구현이 가능할 뿐만 아니라, 용량이 문제되는 FPGA에서도 연산량을 줄임으로써 구현이 가능하다. 2축 로봇의 위치 추종 시뮬레이션을 통해 시간지연제어기의 성능을 확인하였고, 최적화가 용이한 FPGA에 시간지연제어기를 설계하고 다관절 로봇에 적용하여 실험적으로 그 성능을 확인하였다.

II. 로봇팔의 PD 제어

일반적인 n축 로봇의 동력학 모델은 다음과 같다.

$$D(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + f_f(\dot{q}) = \tau \quad (1)$$

여기서, q 는 $n \times 1$ 벡터의 조인트, τ 는 $n \times 1$ 벡터의 토크, $D(q)$ 는 $n \times n$ 관성 행렬, $C(q, \dot{q})\dot{q}$ 은 $n \times 1$ 벡터의 Coriolis와 원심력, $G(q)$ 는 $n \times 1$ 벡터의 중력, $f_f(\dot{q})$ 은 $n \times 1$ 벡터의 마찰력을 나타낸다.

로봇의 안정성을 보장하는 PD제어의 경우 제어 법칙은 다음과 같다.

$$\tau = K_p e + K_D \dot{e} \quad (2)$$

여기서 $e = q_d - q$ 이고 $\dot{e} = \dot{q}_d - \dot{q}$ 이다.

(1), (2)를 등식으로 놓고 양변에 $D(q)\ddot{q}_d$ 를 더한 후 정리하면 다음과 같은 페루프 오차 방정식을 얻을 수 있다.

* 책임저자(Corresponding Author)

논문접수 : 2006. 6. 29., 채택확정 : 2007. 3. 30.

이운규, 전효원, 정슬 : 충남대학교 BK21 메카트로닉스 그룹

(seuspace@hotmail.co.kr;atlantis44@naver.com;jungs@cnu.ac.kr)

$$\ddot{e} + D^{-1}K_D\dot{e} + D^{-1}K_Pe = D^{-1}(D\ddot{q}_d + C + G + f_f) \quad (3)$$

(3)의 우항은 안정상태에서 오차를 나타낸다. 따라서, 우항의 불확실성 변수들을 없애기 위해 시간 지연제어 방식을 사용한다. PD 제어의 경우 중력토크가 보상되면 (3)의 안정성을 보장할 수 있다.

III. 시간지연제어

시간지연 제어 방식은 이전의 정보를 사용해서 불확실성을 없애는 강건제어 방식으로 그 구조가 간단하여 로봇 제어에 많이 사용된다. (1)의 동역학 식은 다음과 같이 간단하게 나타내어 진다.

$$D(q)\ddot{q} + h = \tau \quad (4)$$

여기서 $h = C(q, \dot{q})\dot{q} + G(q) + f_f(\dot{q})$.

시간지연제어 방식의 제어 법칙은 다음과 같다.

$$\tau = \hat{D}u + \hat{h} \quad (5)$$

여기서 \hat{D}, \hat{h} 는 각각 로봇 모델 D, h 의 평가치이다. 제어입력은 다음과 같다.

$$u = \ddot{q} \quad (6)$$

(4)로부터 h 는 다음과 같이 표현된다.

$$h = \tau - D(q)\ddot{q} \quad (7)$$

(7)은 같은 샘플링 시간의 값을 계산해야 하므로 계산이 불가능함을 알 수 있다. 하지만 짧은 시간의 샘플링 시간을 사용하여 이전의 값을 저장했다가 사용하면 $h(t) \cong h(t - \lambda)$ 으로 나타낼 수 있다.

따라서 (7)로부터 실제 얻을 수 있는 모델은 다음과 같이 이전의 정보를 사용해서 대략적인 값으로 나타낼 수 있다.

$$\bar{h} = \tau(t - \lambda) - \bar{D}\ddot{q}(t - \lambda) \quad (8)$$

여기서 λ 는 샘플링 시간이고 \bar{D} 는 관성행렬의 평가치이다. 결과적으로 (5)와 (8)로부터 PD제어기를 사용하는 시간지연 제어법칙은 다음과 같다.

$$\tau = \bar{D}(\ddot{q}_d + K_D\dot{e} + K_Pe) + \bar{h} \quad (9)$$

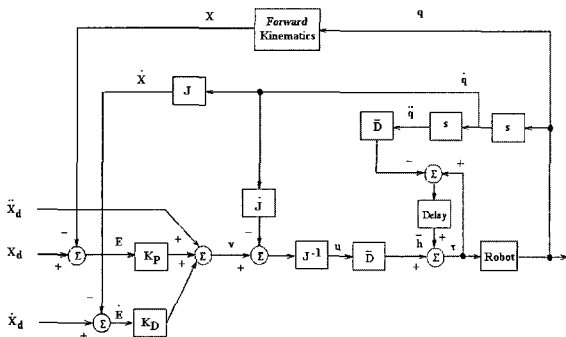


그림 1. 시간지연제어방식구조.
Fig. 1. Time-delayed control structure.

(9)를 (5)에 대입하면 페루프 오차 방정식을 얻게 되고 시스템의 안정성에 관한 조건을 얻게 된다[5]. 그림 1에 시간지연 제어방식의 구조가 잘 나타나있다.

IV. FPGA의 설계

1. FPGA의 전체 구조

본 논문에 사용된 FPGA는 Altera사의 APEX-20K300EQC이다. 이 FPGA는 ROM방식으로 외부 ROM이나 PC로 프로그래밍함으로써 초기화가 가능하며 전원이 공급되지 않으면 프로그래밍 내용이 전부 지워진다. 이 FPGA는 11,520개의 로직셀을 가진다.

설계된 FPGA의 구조는 그림 2와 같다. FPGA는 제어기로서 외부에서 전송되는 각 관절의 각도와 로봇의 각 관절에 부착된 엔코더를 이용하여 로봇을 제어한다. 이를 위해 FPGA에는 주 컨트롤러와 PWM 생성기가 포함되어 있으며, 외부와의 통신을 위한 RS232 신호 변환기도 포함하고 있다. 여기서 주 컨트롤러는 앞서 설명한 시간지연제어기이다. 총 12개의 관절을 제어하기 위해 엔코더 카운터, 컨트롤러, PWM 생성기는 12개가 병렬 구조로 설계된다.

그림 3은 전체적인 구조를 블록다이어그램으로 간단하게 나타냈다. 외부에서 PD이득, 관절각도, 엔코더 신호를 제어기로 입력하고 여기서 최종 출력되는 신호는 DC모터의 제어를 위한 PWM신호가 된다.

그림 4는 그림 3에서 설명한 기능을 모두 포함하는 FPGA 프로그램이다. 각 영역을 주요 기능별로 나누었다. 각 영역별 역할은 다음과 같다.

- A: PD Gain 값
- B: Serial To Parallel Converter
- C: Parallel To Serial Converter
- D: Encoder Signal Input
- E: PWM Signal Output
- F: Controller

A는 PD 제어를 위한 Gain 값으로 상수이다. B와 C는 외부와 Serial 형태로 통신하기 위한 데이터 형 변환기이다. 이

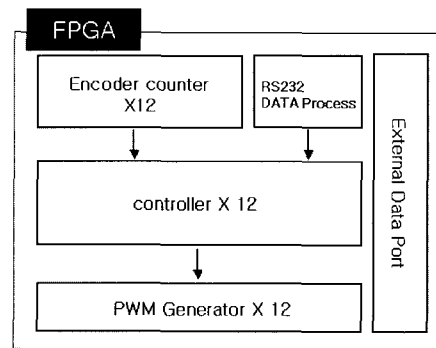


그림 2. FPGA의 구조.

Fig. 2. FPGA structure.

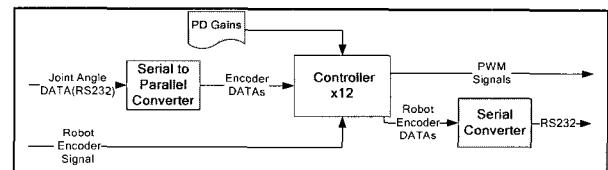


그림 3. 시간지연제어기의 FPGA 블록다이어그램.
Fig. 3. FPGA block diagram of time-delayed controller.

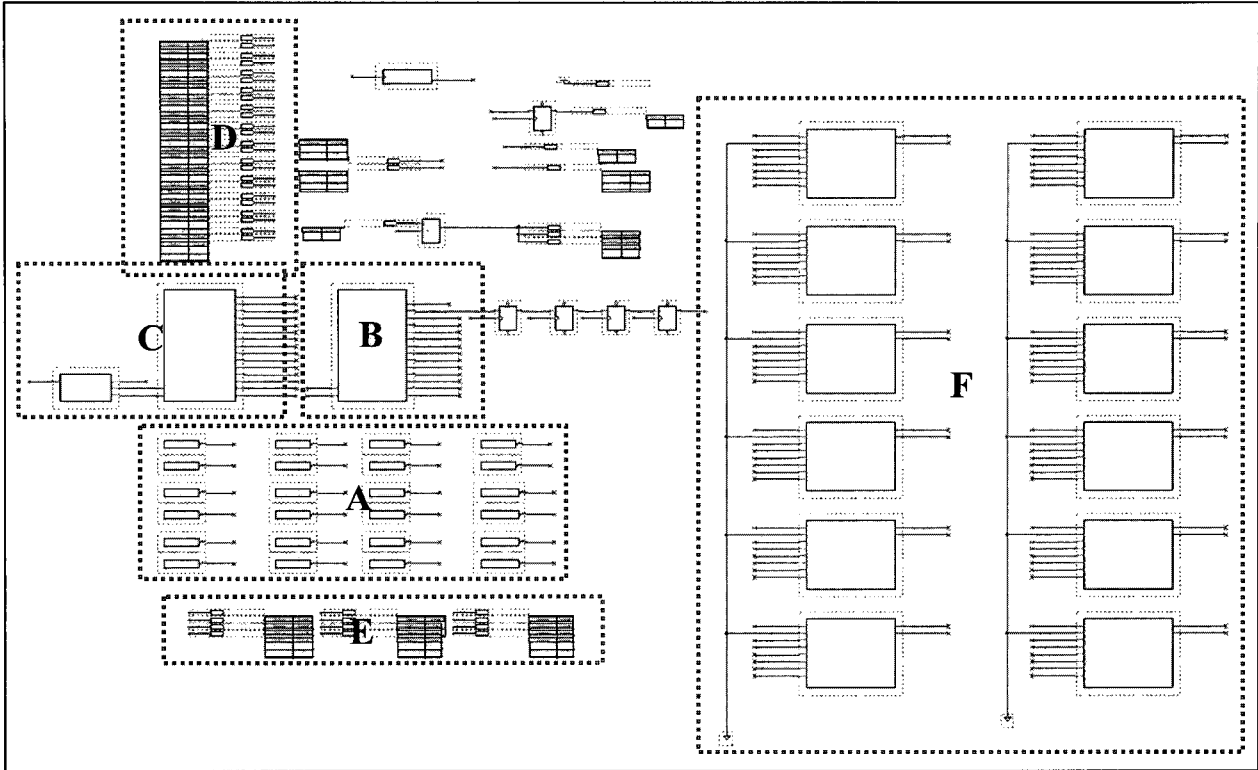


그림 4. FPGA 프로그램.
Fig. 4. FPGA program blocks.

FPGA는 38400BPS의 속도로 외부와 시리얼 통신이 가능하다. D는 로봇에 부착된 엔코더 신호의 입력부이다. E는 제어기 연산이 완료된 PWM 값의 출력부이다. 마지막으로 F는 이 프로그램의 중심이라고 할 수 있는 제어기이다. 12관절의 로봇 팔을 제어하기 위한 같은 기능을 가진 12개의 제어기로 구성되어 있다.

2. 시간지연 제어기의 구조

제어기의 구조는 그림 5와 같다. 이 제어기는 엔코더 카운터, PWM 생성기가 기본적으로 포함된 범용 DC 모터 제어기이다. 여기에 주 제어기인 시간지연제어기가 포함되어 있다.

그림 6은 FPGA에 사용된 PID제어기 프로그램 내용을 보여준다. 엔코더 카운터는 16bit 카운터이다. 그리고, 제어기로 입력되는 각 관절의 각도 값의 크기도 16bit로 전체 연산은 16bit로 수행된다. 하지만, 최종 출력되는 PWM 신호의 범위는

0~255로 8bit이다. 제어기는 그림 7의 시간지연 제어기로 상세하게 나타내어진다.

시간지연제어기는 앞서 설명한 시간지연제어의 (9)를 적용했다. 하지만, FPGA의 특성상 소수점연산이 불가능하고 모든 연산은 매 주기에 병렬로 수행되므로 이를 고려하여 프로그래밍했다. 그림 7의 프로그램을 보면 FPGA에서 시간지연제어를 위해서는 복잡한 연산을 필요로 하는 것을 알 수 있다. 다음 표 1에 시간지연제어기에 사용되는 연산기의 수를 나타냈다.

PC나 DSP 등의 프로세서의 경우 많은 양의 연산을 수행하므로 연산시간이 증가한다. 하지만, FPGA는 모든 연산을 병렬로 수행한다. 병렬 연산을 수행하기 위해서는 1회의 연산당 그에 해당하는 연산기를 FPGA에 설계해야 한다. 물론 주연산기를 따로 설계하고 이를 스케줄링하여 모든 연산을 순차적으로 수행할 수도 있지만 설계가 어렵고 병렬 처리에 의한 빠른 연산이 불가능하다. 따라서 FPGA를 설계할 때 대부분의 경우 병렬연산이 가능하도록 설계한다.

하지만, 연산기의 수가 늘어나게 되면 연산에 사용되는 로직셀이 증가하게 된다. 특히 곱셈기의 경우 연산에 소요되는 로직셀의 용량이 상당히 크다. Altera의 라이브러리에서 기본적으로 제공하는 곱셈기의 경우, 16bit 곱셈 1회당 약 400개의 로직셀을 사용하게 된다. 48회의 곱셈연산을 위해서는 19,200개의 로직셀이 소요된다. 본 연구에 사용된 FPGA의 로직셀은 11,520개이므로 곱셈 연산만으로도 그 용량을 초과한다. 그러므로 라이브러리로 제공되는 기본 곱셈기가 아닌 그림 8에 보여진 시리얼곱셈기를 이용해 곱셈연산을 수행했다.

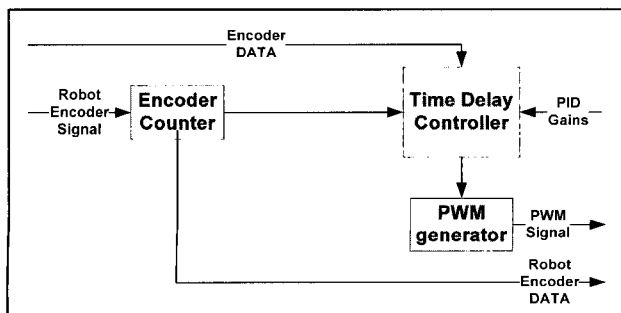


그림 5. 제어기의 구조.
Fig. 5. Time delayed controller structure.

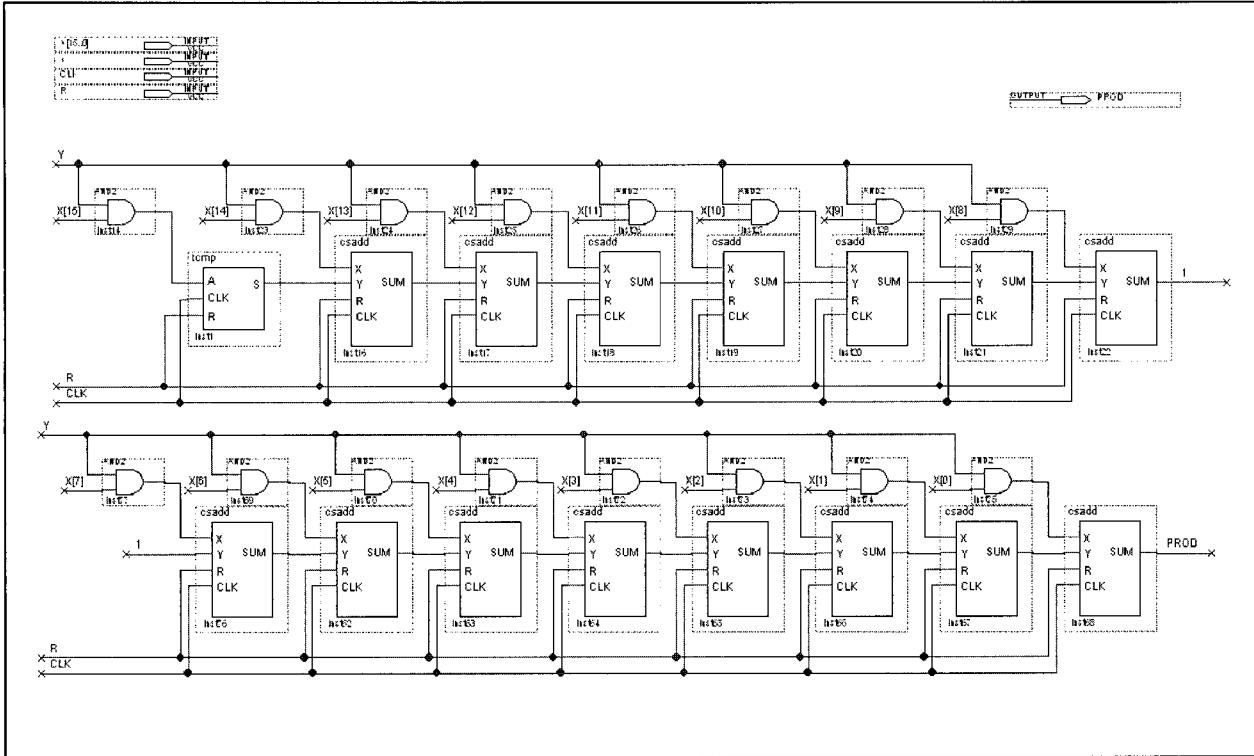


그림 8. 시리얼 곱셈기.

Fig. 8. Serial multiplier.

표 1. 연산횟수.

Table 1. Number of calculation.

연산종류	연산기 당 횟수	총 횟수
덧셈기(뺄셈기)	10	120
곱셈기	4	48
나눗셈기	2	24

시리얼 곱셈기는 16bit 연산 1회당 총 80개의 로직셀을 소요한다. 이는 일반 곱셈기의 1/5수준으로 많은 양의 곱셈 연산이 필요로 하는 경우 효율적이다.

시리얼 곱셈기의 구조는 그림 8과 같다. 시리얼 곱셈기는 AND 게이트와 전가산기의 조합으로 구성되어 있으며, 곱셈 연산을 위한 두 개의 입력은 각각 병렬과 직렬로 입력된다. 그리고, 연산 결과는 직렬신호로 출력된다. 그러므로, 16bit 병렬값을 직렬신호로 변환하는 변환기가 부가적으로 필요하다. 그림 8의 구조를 보면, 16bit 곱셈 연산을 완료하고 그 결과가 출력되기 위해서는 최소 16clock 이상의 시간이 필요하다. 일반적인 곱셈기의 경우 약 2clock 이후에 연산 결과가 출력되는 것에 비해 연산 속도에 있어서 약간 느리다고 볼 수 있다. 하지만, FPGA를 사용한 제어기의 경우 많은 연산이 병렬로 수행되기 때문에 총 연산 속도는 일반적인 마이크로프로세서나 DSP에 비해 빠르다.

V. 시뮬레이션 결과

2축 로봇을 대상으로 시뮬레이션을 하였다. 그림 9는 두

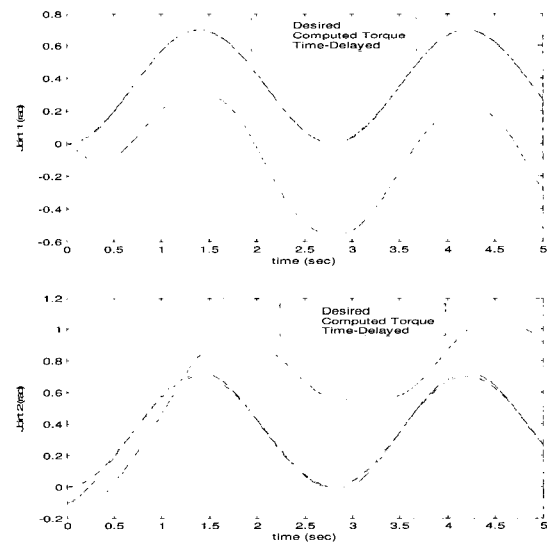


그림 9. 사인파 경로 추정 결과.

Fig. 9. Sinusoidal trajectory tracking result.

팔 로봇의 경우 사인파를 추종하는 시뮬레이션 결과를 나타낸다. 각 조인트에 다음과 같은 경로를 추종하도록 하였다.

$$A = 20 * \sin(\text{time} * 3.141592 / 1.4 - \pi / 2) + 20 \tag{10}$$

모델의 오차가 50%이고 각 조인트에 마찰력이 불확실모델로 주어졌을 경우에 계산 토크 방식과 시간 지연 제어기와의 성능을 비교하였다. 그림의 추종 결과를 비교해 볼 때 시간 지연 제어기가 우수함을 알 수 있다.

VI. 실험 결과

1. 실험환경

그림 10은 본 연구에 사용된 로봇이다. 한 팔 6개의 관절에 (10)의 사인과 형태로 각도를 변화시키며 그 결과를 확인하여 실험하였다. 6개의 관절 중 3, 4, 5번 3개 관절의 제어 결과를 그림 11에 나타내었다.

2. PD제어

먼저 간단한 PD제어방식을 구현하였다. 이 절에서 사용하는 PD 이득은 표 2와 같이 설정하였다.

그림 11에는 로봇 팔의 위치추종 제어 결과와 오차가 나타나 있다. PD제어기 만으로도 위치를 잘 추종하는 것을 볼 수 있다. 위치추종 오차는 대략 조인트 4에서 2.5도 정도가 가장 큰 것으로 나타난다.

3. 시간지연제어

다음으로 시간지연제어기를 적용하여 실험하였다. PD 이득은 앞 절의 PD제어와 같이 설정했다. 그림 12에서 보면 각 조인트의 추종오차가 현저하게 줄어든 것을 볼 수 있다. 특히 조인트 4같은 경우, 1도 이하로 줄어들었다. 조인트 3과 5는 오차가 0.5도로 줄어들었다.

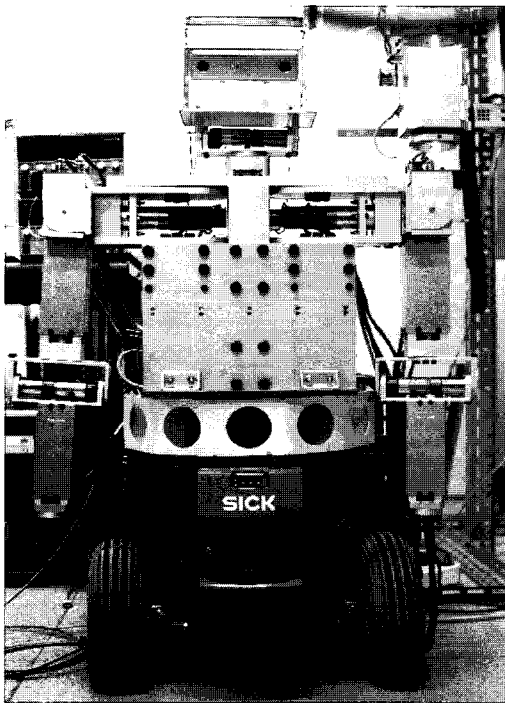
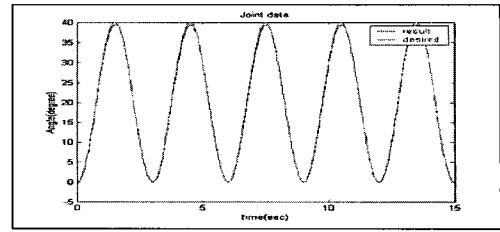


그림 10. 다관절 로봇 팔.
Fig. 10. Humanoid robot arms(ROBOKER).

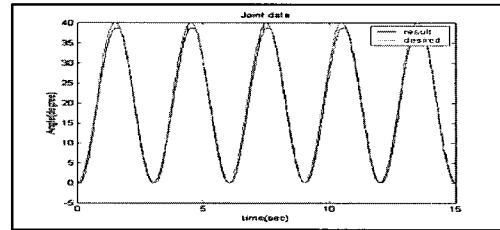
표 2. PD 이득값.

Table 2. PD gains.

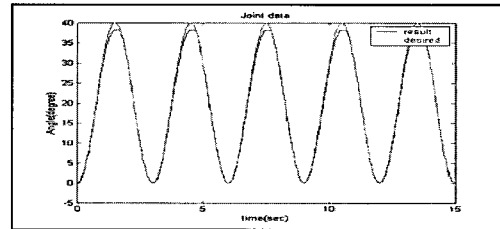
Joint	P Gains	D Gains
1	25	2
2	33	3
3	33	3
4	8	1
5	15	1
6	10	1



(a) Joint 3

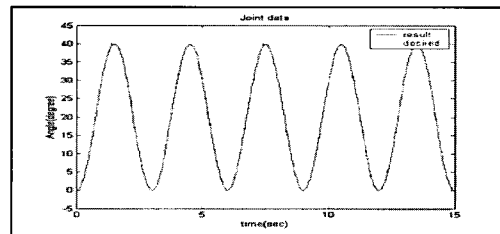


(b) Joint 4

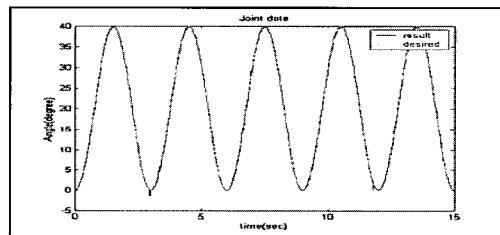


(c) Joint 5

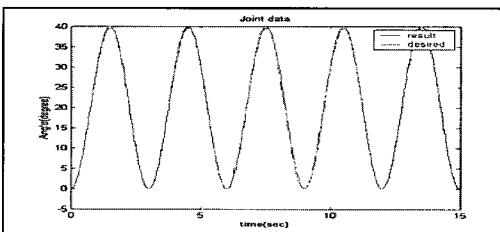
그림 11. PD제어 경로추종 결과.
Fig. 11. Tracking results of PD control.



(a) Joint 3



(b) Joint 4



(c) Joint 5

그림 12. 시간지연제어기 경로 추종 결과.
Fig. 12. Tracking results of a time-delayed controller.

시간 지연제어의 결과를 보면 PD 제어보다 좋은 결과를 얻을 수 있음을 알 수 있다. 오차는 최대 0.5~1.0도 정도로 PD제어와 비교했을 때 반 정도로 줄어들었다. 시간지연제어에서 \bar{D} 의 값이 결과에 영향을 끼친다는 것을 확인할 수 있다. 하지만, PD 이득값에 상관없이 \bar{D} 값을 크게 한다고 해서 좋은 결과를 얻을 수 있는 것은 아니다. \bar{D} 의 값은 안정성을 만족하는 범위 안에서 설정해야 한다.

VII. 결론

본 논문에서는 구현이 간단한 강건제어방식인 시간지연제어를 FPGA에 설계하고 휴머노이드 로봇 팔을 제어해 봄으로써 그 성능을 확인했다. 그 결과 FPGA를 활용한 고성능의 제어기 설계 및 활용도 충분히 가능함을 확인했다. FPGA를 활용하면 PC등의 일반적인 제어 시스템보다 최적화된 제어기를 저 비용으로 구현할 수 있다는 장점이 있다. 하지만, 소수점 연산이 불가능하고 디지털 레벨에서 설계해야 하므로 프로그래밍이 복잡하고 어렵다는 단점이 있다. 또한, 연산용량의 제한이 있어 FPGA의 설계 시 시스템의 연산량도 고려해야 한다. 또한 시간 지연 제어기의 관성행렬과 가속도의 평가치를 얼마나 정확하게 구하는가 하는 것이 성능에 매우 중요하게 작용하는 것을 볼 수 있었다.

참고문헌

- [1] R. K. Dueck, "Digital design with CPLD: Applications and VHDL," DELMAR, Thomson Learning, 2001.
- [2] 김성수, 정슬, "비선형 시스템을 위한 DSP와 FPGA를 이용한 지능제어기의 하드웨어 구현," 제어·자동화·시스템공학 논문지, 제 10 권 제 10 호, pp. 922-929, 2004.
- [3] T. C. Hsia and L. S. Gao, "Robot manipulator control using decentralized linear time-invariant time-delayed joint controllers," *IEEE Conf. on Robotics and Automation*, pp. 2070-2075, 1990.
- [4] K. Yousef-Toumi and O. Ito, "A time delay controller for systems with unknown dynamics," *Journal of Dynamic Systems, measurement, and Control*, vol. 112, pp. 133-142, 1990.
- [5] K. Y. Toumi, "Stability of uncertain linear system with time delay," *Trans. of ASME*, vol. 113, pp. 558-567, Dec. 1991.
- [6] K. Y. Toumi, "Analysis of linear time invariant system with time delay," *Trans. of ASME*, vol. 114, pp. 544-555, Dec. 1992.
- [7] S. Jung, T. C. Hsia, and R. G. Bonitz, "Force tracking impedance control for robot manipulators with an unknown environment: theory, simulation, and experiment," *Journal of Robotics Research*, vol. 29, no. 9, pp. 765-774, 2001.
- [8] P. W. Jeon and S. Jung, "Teleoperated control of mobile robot using exoskeleton type motion capturing device through wireless communication," *IEEE/ASME Advanced Intelligent Mechatronics*, pp. 1107-1112, 2003.



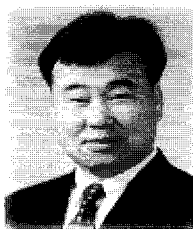
이 윤 규

1978년 2월 25일생. 2003년 충남대학교 메카트로닉스공학과 졸업. 2006년 충남대학교 메카트로닉스공학과 석사 졸업. 현재 도담시스템 연구원. 관심분야는 임베디드 시스템 설계, 로봇공학, 무인 차량 로봇 제어.



전 효 원

1984년 1월 8일생. 2006년 충남대학교 메카트로닉스공학과 졸업. 2006년~현재 충남대학교 메카트로닉스공학과 석사 과정. 관심분야는 임베디드 제어 시스템 설계, 인간과 로봇의 상호작용을 위한 힘제어 응용공학.



정 슬

1964년 9월 11일생. 1988년 미국 웨인 주립대 전기 및 컴퓨터 공학과 졸업. 1991년 미국 캘리포니아대 데이비스 전기 및 컴퓨터 공학과 석사. 동 대학 박사. 1997년~현재 충남대학교 메카트로닉스공학과 부교수. 관심분야는 지능 제

어 및 지능 로봇 시스템, 로봇과 인간의 상호작용, 무인 로봇의 원격제어.