

케이블 감속을 이용한 소형 로봇의 개발과 실시간 제어에 관한 연구

A Study of Development and Real Time Control of Small Size Robot by Cable Reduction

홍 종 성* 이 정 완**
Hong, Jong-Sung Lee, Jung-Wan

Abstract

In this thesis, a three degrees of freedom robot, which is able to provide sufficient precision for various robot researches, has been developed. The cable mechanism is used as a basic transmission of robot joints. Based on an optimal design strategy, link and joint parameters are determined and then overall geometry of the robot is designed. As an architecture of robot control, real time control system using real time linux and RtiC-Lab(Real Time Controls Laboratory) is developed. This system, written in C and based on Linux O/S, includes text editor, compiler, downloader, and real time plotter running in host computer for developing control purpose. Using these hardware and software, simple PD position control is implemented, the results shows the effectiveness of the system.

키워드 : 실시간 제어, 케이블 감속, 소형로봇
Keywords : *real time control, cable reduction method, small size robot*

1. 서론

로봇은 예전부터 다양한 형태와 기능으로 제작되어 왔다. 로봇의 응용 분야는 학교, 교실에서의 미니로봇에서부터 자동차 공장의 용접 로봇, 우주 왕복선의 원격 조정 로봇에 이르기까지 매우 다양하다. 소비자 지향적인 로봇의 대부분은 실용적이기보다는 진기한 상품 가치를 갖고 있는데 반해, 공장에 도입된 로봇은 이미 제조공정에 큰 영향을 주고 있다. 이러한 환경에서 교육용이나 연구용, 소형의 여러 가지 적은 비용의 로봇들이 로봇 교

육의 실습실이나 연구실에서 사용될 수 있도록 개발되었다. 하지만 값이 싸기 때문에 신뢰도가 낮고 부정확하며 센서가 없고 소프트웨어가 적절하지 못하다. 가정용 로봇은 현재의 문제점을 찾는 해결책이다. 1980년의 전반기 동안 많은 가정용 로봇이 시장에 소개되었으나 이들은 감지 기술이 부족하여 조심스럽게 가르쳐준 길을 따라 가는 것 이상으로는 할 수 없었다. 그래서 현재에는 교육용으로 많이 사용되고 있으며 일부는 전시장에서 사람들의 호기심을 끄는 목적으로 사용되고 있다. 하지만 다른 심각한 문제점은 이러한 로봇들은 다른 프로그램이나 매니플레이터와의 확장성이 없다는 것이고 가격이 매우 비싸다는 단점이 있다.[1][2]

현재 국내외의 여러 업체에서 교육용 또는 연구용 소형로봇을 제작하고 판매하고 있다. 이러한 상품의 제품은 대부분이 내부가 보이지 않게 되어있

* 강원대학교 대학원 기계 메카트로닉스공학과
** 강원대학교 기계메카트로닉스공학부 조교수, 공학박사

고, 기어를 이용해서 움직이고 감속하게 설계되어 있다. 이것을 교육용으로 사용할 때 학생들이 로봇의 내부구조를 이해하기 매우 힘들 것이다. 또한 가격이 저렴한 것은 정밀도가 많이 떨어지고 기어를 정밀하게 만들려면 생산단가가 올라가고 기어 자체의 특성상 소음과 백래쉬의 큰 단점을 지니고 있다. 이로 인하여 결과적으로 정밀도가 매우 떨어지며, 실제 구동을 하는데 있어서 복잡한 작업을 수행할 수 없으며 또한 연구용으로는 적절히 사용될 수 없는 문제점을 동시에 지니고 있다. 그리고 현재 나와 있는 상용의 로봇들은 대부분 크기가 커서 연구실의 공간을 많이 차지한다. 이에 로봇의 움직임과 내부구조를 쉽게 이해할 수 있고, 정밀도가 향상되고 교육용이나 연구용으로 사용할 수 있으며 공간을 많이 차지하지 않는 소형 로봇이 필요하다.

본 연구에서는 소음과 백래쉬의 문제가 거의 없고 강성을 크게 보장해주는 감속장치로 알려진 케이블 감속 기법과 최적 설계의 개념을 이용하여 소형 고정밀 로봇을 설계 및 제작하고 컴퓨터 인터페이스 및 실시간 제어시스템을 개발하는데 그 목표를 둔다.

케이블감속기법을 이해하기 위해서 같은 기법을 사용하고 있는 Barrett사와 여러 케이블감속에 관한 자료를 수집하고 분석했으며 케이블감속기법을 이용한 바 있는 한국과학기술원의 자료와 자문을 구했다. 그 결과를 토대로 분석과 연구를 통해서 단점을 보완하고 몇 번의 설계변경을 통해 케이블 감속기법, 조인트 및 링크길이 형태 등이 정해졌으며 이를 바탕으로 해서 제작하였다.

로봇의 구동 및 제어에는 기계부와 전기부로 나누어 생각하기로 했다. 기계부의 제작에서 로봇의 주재료선정, 모터용량 및 크기선정, 외형설계, 힘속도를 고려했고, 로봇의 제어를 담당하는 전기부에서는 기존에 있는 s626 DSP보드와 RT-linux를 사용하여 컴퓨터와의 인터페이스를 구축했다.

2. 로봇의 설계와 제작

2.1 다양한 동력전달 기법

기계의 동력전달 기법은 Fig.1과 같이 여러 가지 방법이 있다. 첫 번째 그림이 케이블을 이용한 동력전달 방법이며, 각각 기어, 주축, 공압 및 유압, 마찰, 링크를 이용한 동력 전달방법을 나타낸 것이다. 유압은 대형이나 큰 하중을 들어올릴 수 있는 로봇에 유리하고, 동력/무게비가 가장 크다. 강성 시스템, 고정밀도, 응답속도가 빠르고 감속기어가 필요 없다. 또 넓은 범위의 속도로 작업을 할 수 있고 손실 없이 위치를 유지 가능하나 단점은 누수가 있을 수 있고 청결이 필요한 곳은 적합하지

않으며 펌프, 저수조, 모터, 호스가 필요하고 규모가 커진다. 그리고 매우 고가이며 잡음이 심하고, 유지보수도 필요하며 유체점성이 온도에 따라 변하고 유체에 이물질이 있어도 사용가능, 구동기의 토크가 세고 압력이 높으며 관성이 크다는 특성이 있다. 다음으로 공압의 특성을 보면 누수 방전이 없고 저가이며 단순하고 압력이 유압보다 낮다. 부품집기나 배치작업에 유리하며 순응적 시스템이다. 그러나 잡음이 있고, 공압과 필터가 필요하며 선형 위치제어가 어렵고 하중이 실리면 변형되고 강성도가 매우 낮고 부정확한 응답특성을 가진다. 동력/무게비가 가장 낮다. 기어는 힘의 전달이 좋은 반면에 잡음이 있고 백래쉬가 있다. 또 윤활을 필요로 하고, 정밀한 기어를 제작하기가 어렵다.[3] 정밀한 기어는 매우 고가이다. 링크식 전달방법은 공간의 제약이 많고 곡선구간에서의 힘 전달이 어렵다는 단점을 가지고 있다. 각각의 전달방법에는 장점과 단점이 있지만 본 논문에서는 위에 열거한 여러 가지 전달방법 중에 기어기법과 비교해서 백래쉬가 거의 없고 소음이 적고 강성이 크고 각각의 모터를 기저로 가져올 수 있는 장점을 지니고 있는 케이블을 이용한 동력전달 방법을 사용한다.[4]

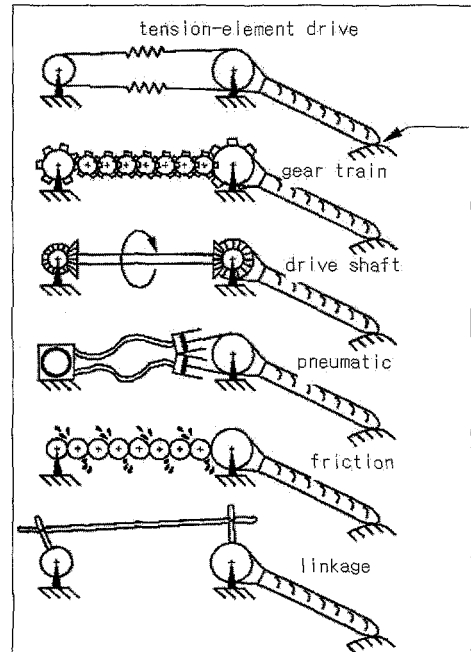


Fig.1 Various types of transmissions

2.2 케이블감속기법 설계

케이블 감속기법의 기본적인 원리는 Fig. 2에서

와 같다. 즉 케이블을 이용하여 동력을 전달하고 감속은 작은 원통과 큰 원통의 지름비가 감속비가 되는 것이다. 작은 원통의 지름이 10mm 이고 큰 원통의 지름이 50mm이면 감속비는 5:1이 되는 것이다. 이것은 기어감속기법의 원리와 유사하나 각각의 원통에 정해진 길이의 케이블을 고정시켜서 동력을 전달하는 방법은 기어감속기법과는 다르다고 할 수 있다. Fig.2의 그림은 케이블을 이용해서 전체적인 감속이 어떻게 발생하는지를 보여주는 그림이다. 모터에 연결된 케이블이 다른 풀리에 연결되고 이것이 또 다른 풀리에 연결되면서 두 번의 감속이 일어나게 되는 것을 도식화 한 것이다.

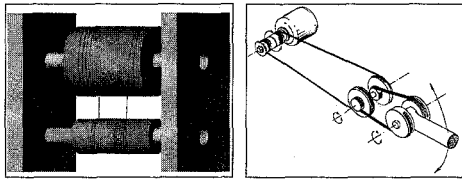


Fig.2 Cable reduction method.

Fig. 3에서는 케이블감속 기법을 응용한 것의 예를 보여준 그림이다. 그림을 보면 1번 모터 풀리에 2, 3번 케이블이 감겨져 있고 2번 모터 풀리에 1, 4번 케이블이 감겨져 있다. 그림의 좌측에서 봤을 때 좌측 풀리는 시계방향으로 회전하고 우측의 풀리가 반 시계방향으로 회전한다고 가정하면 우측의 풀리에는 1번 케이블이 감기게 되고 좌측의 풀리에는 2번 케이블이 감기게 된다. 결과적으로 중간의 풀리는 반 시계방향으로 물운동을 하게 된다. 또 우측과 좌측의 풀리가 동시에 같은 방향으로 반 시계방향으로 회전을 하게 된다면 우측 풀리에는 1번 케이블이 감기고 좌측 풀리에는 3번 케이블이 감기게 되어서 결과적으로 중간의 풀리는 전면으로 피치운동을 하게 되는 것이다. 케이블은 장력을 많이 주면 줄수록 좋지만 너무 강하게 힘을 가하면 축에 무리가 갈 염려가 있기 때문에 주의할 기울여야 한다. 이렇게 함으로써 백래쉬가 없고, 마찰이 거의 없으며 기어보다 조용하고 부드럽고 견고하며 강한 장점이 나타나게 된다.[4]

케이블의 재료는 강한 강성과 탄성이 적은 케이블을 사용했으며 이것은 가는 케이블을 여러 가닥을 꼬아서 만든 것으로 강한 강성을 가진 케이블이다. 케이블의 특성상 직경은 감기는 축 직경의 1/20 이상이 되면 안 된다. 왜냐하면 1/20 이상이 되면 케이블 자체 강성에 의해서 자연적으로 풀리려하는 성질 때문에 쉽게 느슨해지기 때문이다. 이것은 탄성이 다른 여러 가지 케이블을 비교해보면 쉽게 이해할 수 있을 것이다. 그리고 풀리의 표면은 테프론(teflon)을 입힌 강한 세락믹 표면으로 해야 가장 이상적[4]이지만 본 논문에서의 로봇 제작

은 전체적으로 가볍고 강성이 좋은 두랄루민을 사용하였다.

여기에서 양쪽옆의 풀리의 직경비와 중간 풀리의 직경비는 같아야 한다. 만약 두 풀리의 직경비가 일정하지 않다면 풀리가 몇 번의 움직임을 반복하면 케이블이 조금씩 느슨하게 되면서 결과적으로는 로봇의 정밀도에 직접적인 영향을 주게 된다.

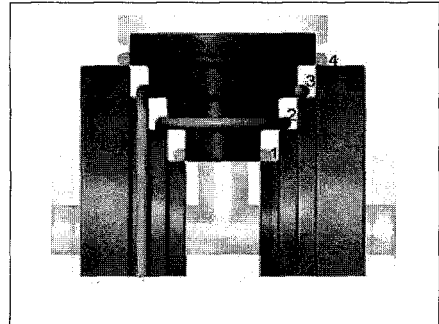


Fig.3 Application of Cable reduction method

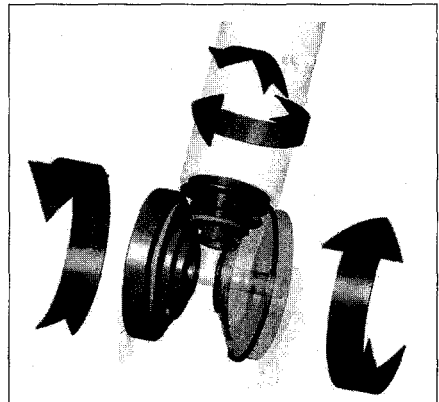


Fig.4 Application of Cable reduction method

Fig. 4는 케이블 감속기법의 응용 예로서 2개의 모터로 로봇 팔의 풀과 피치 운동을 유발시키는 원리를 보인 것이다. 위에서도 언급했듯이 모터1 이나 모터2가 반대로 회전을 한다면 한 풀리당 1쌍의 케이블의 감김과 풀림에 의해서 중간의 풀리는 회전을 하게 될 것이고 링크는 물운동이 유발될 것이다. 또한, 모터1,2가 동시에 같은 방향으로 회전하면 우측과 좌측 풀리에 고정되어있는 1쌍의 케이블은 동시에 같은 방향으로 감기고 풀리기 때문에 중간링크는 결과적으로 피치 운동이 일어나게 된다.

2.3 로봇의 링크길이, 형태설계

본 연구의 로봇의 전체적인 형태와 크기는 Fig. 5와 같이 정하였다. 여기에는 3개의 회전 조인트가 있으며, 3개의 모터가 장착된다. 최대 전체 높이는 870mm 이고 폭은 250×250mm로 했다. 링크1과 링크2의 재질은 두랄루민재질의 각각 직경이 50mm와 70mm이고 길이는 270mm 과 240 mm인 중공축을 썼고, 링크1의 무게를 600g, 링크2의 무게를 370g으로 결정했으며, 로봇의 맨 끝단에 걸리는 예하중은 500g으로 결정했다.

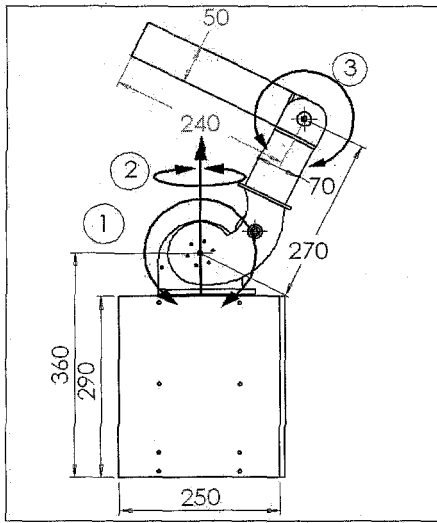


Fig.5 Schematic of robot

2.4 로봇의 조인트설계

Fig.5에서 정한 여러 가지 물성치를 바탕으로 로봇의 조인트를 설계했다. 전체적인 로봇의 모양과 모터의 위치는 Fig.6에 나타난 것과 같다.

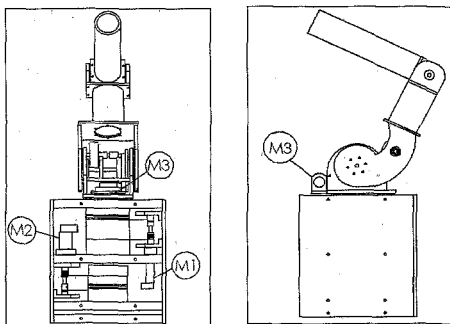


Fig.6. Whole schematic of robot with motor

모터1(M1)은 Fig.6과 같이 로봇팔의 전체 즉, 링크

1 과 링크2의 팔을 회전시켜야 하기 때문에 팔의 처짐을 고려하여 설계하였다. 그리고 모터2(M2)의 회전은 중심의 폴리에 연결된 케이블에 의해서 동력이 전달되고, 이로 인해서 링크1의 피치운동을 유발하게 한다. 모터3(M3)의 회전은 여러 번의 폴리를 거쳐서 감속과정을 거친 후 최종적으로 링크2의 피치운동을 유발시킨다.

Fig.7의 (a)는 링크1과 링크2의 관절부분으로 모터를 결합하고 베어링의 결합도를 개략적으로 나타낸 것이고 (b)는 결합한 실제 사진을 보인 것이다. 베어링의 선정은 축 방향과 경 방향의 하중에 양호한 볼 베어링으로 선정했으며 링크1의 기저부에는 축 방향 하중에 강한 추력베어링을 선정했다.[5] 그리고 로봇의 처짐을 방지하기 위해서 하우징고정에 주의를 기울였다. 특히 이 부분의 설계는 로봇전체의 케이블 기법 중에 제일 중요한 곳으로 주의를 요하는 부분이다. Fig.3에서도 언급했듯이 1번 폴리와 2번 폴리가 직접적으로 접촉하는 것은 피하고 어느 정도의 정확한 간격을 두어야 케이블기법의 장점을 살릴 수 있다. 간격의 범위를 크게 주면 폴리에서 케이블이 이탈 될 수 있다. 또한 간격의 범위가 작으면 두 폴리의 마찰로 인하여 정밀도가 떨어지고 케이블기법의 장점이 감소될 수 있다.

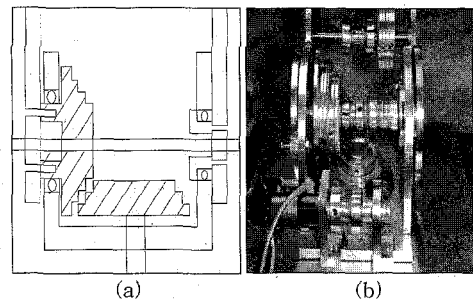


Fig.7 Joint of 2nd axis

케이블기법의 로봇을 설계할 때 중요한 것은 항상 케이블이 적당한 장력을 유지하고 있는 것이다. 그러나 로봇이 여러 번의 작업을 반복하다 보면 케이블의 장력이 유지되지 못하는 것을 발견했다. 이것은 케이블의 문제와 로봇 부품의 정밀성의 문제 때문에 생긴 것으로 예상하고 있다. 이런 문제는 케이블기법의 로봇에서는 쉽게 나타나는 현상으로 알려져 있다. 그래서 최초 로봇의 조립단계에서의 장력을 그대로 유지하고, 또한 케이블의 단선과 느슨해진 케이블을 쉽게 초기상태로 만드는 작업을 쉽게 하기 위해서는 어떠한 특별한 장치가 필요했다. 그래서 본 연구에서 여기에 적당한 장치를 개발했다. Fig.8은 고안한 장력 조절장치를 보

인 것이다. 이것은 케이블의 장력이 감소했을 때 초기의 장력을 주기 위해서 고안한 것으로서 만약 케이블이 느슨해지면 줄을 다시 쉽게 당길 수 있도록 설계한 것이다. 이것의 원리는 케이블이 느슨해졌을 때 윗 부분의 풀리와 아랫부분의 풀리를 따로 분리함으로써 장력을 유지하는 장치이다. 고안한 장치를 사용하여 케이블을 감는 원리는 지렛대의 원리를 이용했고, 여기에 추가로 베벨 기어를 달아서 고정을 시킨다면 더욱더 좋은 장력 조절장치가 될 것이다.

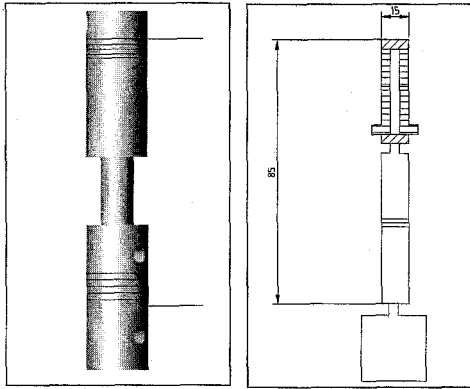


Fig.8 Tension regulator

Fig.9는 로봇내부 전체의 케이블감속의 형태를 투시하여 도시화한 것이다. 그림에서 모터1(M1)의 구동은 로봇전체를 회전하게 한다. 모터2(M2)의 구동은 링크1의 움직임을 유발하고, 모터3(M3)는 여러 번의 풀리를 거쳐서 감속을 하고 링크2를 움직이게 한다. 여기서 로봇의 기저는 로봇의 어떤 운동에도 로봇의 중심을 잡아야 하기 때문에 질량이 많이 나가게 하기 위해서 중심의 모터1,모터2에 연결되는 풀리는 중공축을 사용하지 않았다. 또한 모터3(M3)가 링크2로 전달되는 거리가 길기 때문에 케이블의 길이도 다른 축보다는 더욱 많이 소요되고, 케이블의 처짐을 예방하기 위해서 중간에 더욱 많은 풀리를 사용했다. 모터2의 구동으로 링크1의 움직임을 담당하는 풀리부분은 특히 많은 힘을 받으므로 다른 부분에 사용한 케이블 0.5mm보다 두꺼운 0.8mm의 것을 사용하였다. 또한 로봇의 부피를 줄이기 위해서 케이블이 감기는 축과 축의 간격을 최소화했으며, 전체적인 풀리의 고정에는 6각 렌치볼트를 사용했으며 하나의 풀리에 4개 이상의 볼트를 사용했다.

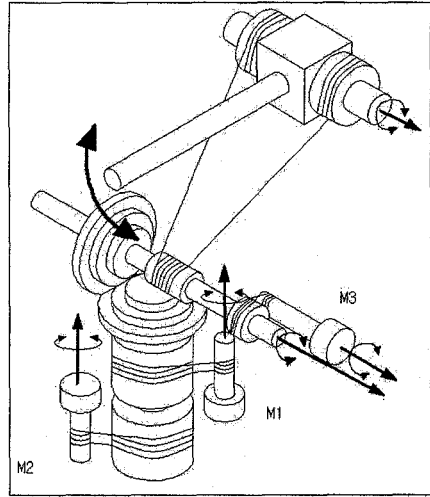


Fig.9 Cable reduction method of whole robot

2.5 로봇의 힘, 속도 해석 및 모터의 용량 선정

2.5.1 로봇의 힘, 속도 해석

Fig.10의 (a)는 본 연구에서 설계한 로봇의 좌표계를 나타낸 것이다. 2개의 링크와 3개의 관절부로 구성되어 있다. 기저부에서 링크1과 링크2의 회전축, 로봇전체의 회전을 θ_1 으로 가정하고, 기저와 링크1의 거리를 L_1 으로 가정하고 링크1과 링크2의 길이를 각각 L_2 와 L_3 로 가정했다. 또 2번째 관절의 회전각을 θ_2 로 나타냈고 3번째 관절의 회전각을 θ_3 로 가정했다. 이것을 바탕으로 로봇의 위치, 속도, 힘을 해석하기 위한 Denavit-Hartenberg Parameter를 구한 결과를 Table1에 나타냈다. Fig.10의 (b)는 이것을 바탕으로 로봇 전체의 형태를 Solid Works로 모델링한 그림이다.

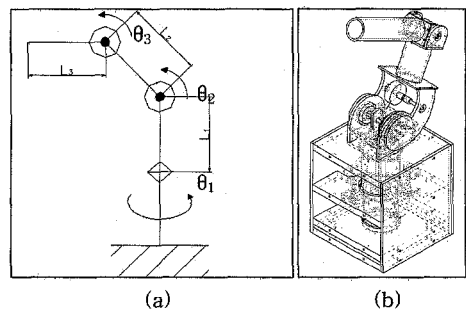


Fig.10 Definition of coordinate frames for Robot and modeling of Robot

Table1 Denavit-Hartenberg Parameter (D-H Parameter)

Coordinate Frame Number	Link Twist (Radian)	Link Length	Link Offset	Joint Angle
i	a_{i-1}	a_{i-1}	d_i	θ
1	0	0	0	θ_1
2	$-\frac{\pi}{2}$	0	L_1	θ_2
3	0	L_2	0	θ_3
4	0	L_3	0	0

식 (1)은 계 $i-1$ 에서 계 i 까지의 일반적 변환행렬을 정의한 식을 나타낸 것이다.[6]

$${}_{i-1}^i T = \begin{bmatrix} C\theta_i & -S\theta_i & 0 & a_{i-1} \\ S\theta_i C a_{i-1} & C\theta_i C a_{i-1} & -S a_{i-1} & -S a_{i-1} d_i \\ S\theta_i S a_{i-1} & C\theta_i S a_{i-1} & C a_{i-1} & C a_{i-1} d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

여기에서 $S\theta_i = S_i$, $C\theta_i = C_i$ 로 정의하기로 한다. (1)식을 이용하여 정기구학을 구하면 아래와 같이 구할 수 있다. 식(2)는 기저에서 링크1 까지의 변환을 나타낸 것이다.

$${}^0_1 T = \begin{bmatrix} C_1 & -S_1 & 0 & 0 \\ S_1 & C_1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

식(3)은 첫 번째 프레임에 대한 위치(x, y, z)를 나타낸 것이고, 식(4)는 기저부에 대한 위치를 나타내는 식을 나타낸 것으로써 로봇의 각 관절의 각 θ 를 알면 말단부의 로봇의 위치를 알 수 있는 식을 나타낸 것이다.

$${}^1 P = \begin{bmatrix} C_2 L_3 + C_2 L_2 \\ L_1 \\ -S_2 L_2 - S_2 L_3 \\ 1 \end{bmatrix} \quad (3)$$

$${}^0 P = {}^0_1 T {}^1 P$$

$${}^0 P = \begin{bmatrix} -S_1 L_1 + C_1 C_2 L_2 + L_3 C_1 C_2 \\ C_1 L_1 + C_2 S_1 L_2 + L_3 S_1 C_2 \\ -L_2 S_2 - S_2 L_3 \end{bmatrix} \quad (4)$$

역기구학을 구해보면 아래의 식과 같다.

$$\begin{aligned} \theta_1 &= \text{Atan}(x, y) = \text{Atan}(C_1, S_1) \\ \theta_2 &= \text{Atan}(r, s) - \text{Atan}(a_2 + a_3 C_3, a_3 S_3) \\ &= \text{Atan}(\sqrt{p_x^2 + p_y^2}, p_z) - \text{Atan}(L_2 + L_3 C_3, L_3 S_3) \\ \theta_3 &= \tan^{-1} \frac{S_3}{C_3} = \text{Atan2}(S_3, C_3) \end{aligned} \quad (5)$$

식(5)를 이용해서 로봇의 위치를 알면 각 관절의 각 θ 를 알 수 있다.

시간에 대한 기구 요소들의 기하학적 표현인 자코비안을 구해보면 아래의 식과 같고 식(7)은 최종적으로 자코비안 행렬을 나타낸 것이다. 자코비안 행렬을 이용하여 직교공간의 속도를 관절속도로 나타낼 수 있고 반대로 관절속도를 직교공간속도로 나타낼 수 있는 것이다.

$$\begin{aligned} \dot{x} &= J \dot{\theta} \\ \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} &= \begin{bmatrix} -C_1 L_1 - S_1 (C_2 L_2 + L_3 C_2) \\ -S_1 L_1 + C_1 (L_2 C_2 + L_3 C_2) \\ 0 \\ -C_1 (L_2 S_2 + L_3 S_2) - L_3 C_1 S_2 \\ -S_1 (L_2 S_2 + L_3 S_2) - S_1 L_3 S_2 \\ L_3 C_2 + C_2 L_1 & L_3 C_2 \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \end{aligned} \quad (6)$$

$$\begin{aligned} J &= \begin{bmatrix} -C_1 L_1 - S_1 (C_2 L_2 + L_3 C_2) \\ -S_1 L_1 + C_1 (L_2 C_2 + L_3 C_2) \\ 0 \\ -C_1 (L_2 S_2 + L_3 S_2) - L_3 C_1 S_2 \\ -S_1 (L_2 S_2 + L_3 S_2) - S_1 L_3 S_2 \\ L_3 C_2 + C_2 L_1 & L_3 C_2 \end{bmatrix} \end{aligned} \quad (7)$$

본 연구에서 설계한 로봇전체의 높이는 870mm 이며 폭은 250×250mm 이고 링크의 길이는 $L_1 = 70$ mm, 링크1(L_2) = 270 mm, 링크2(L_3) = 240 mm 이다.

2.5.2 모터의 용량 선정

각 부분에 사용될 모터를 선정하기 위해서 로봇 전체의 링크의 무게와 관절에 걸리는 토크의 세기를 계산해야 한다. 식(8)은 본 연구에서 제작되는 로봇의 링크가 원통형상의 두랄루민관을 사용하므로 원통형상을 가진 질량의 관성 모멘트를 구하는 식을 나타낸 것이다.[7]

$$\sum T = J\ddot{\theta} \quad (8)$$

$$I_x = \frac{1}{2} mr^2$$

$$I_y = I_z = \frac{1}{12} m(3 \times r^2 + L^2)$$

식(9)는 토크평형 방정식(torque balance equation)을 나타낸 것으로써, N은 감속비, I_l 은 부하의 관성, I_m 은 모터의 관성, b_m 과 b_l 은 모터와 부하에 대한 마찰점성계수(viscous coefficients of friction)를 나타낸다. 이 두 식을 통하여 각각의 joint에 작용되는 토크를 구할 수 있다.[3]

$$\begin{aligned} T_m &= I_m \ddot{\theta}_m + b_m \dot{\theta}_m + \frac{1}{N} T_l \\ &= I_m \ddot{\theta}_m + b_m \dot{\theta}_m + \frac{1}{N} (I_l \ddot{\theta}_l + b_l \dot{\theta}_l) \\ &= I_m \ddot{\theta}_m + b_m \dot{\theta}_m + \frac{1}{N^2} (I_m \ddot{\theta}_m + b_l \dot{\theta}_m) \end{aligned} \quad (9)$$

위의 공식을 이용하여 본 연구에서 설계한 로봇의 초기값인 링크 1개당 질량을 0.37 Kg으로 했고 요구 가속도는 0.5 m/s^2 으로 가정했다. 링크 외경(D)은 0.05m, 로봇 끝단에 걸리는 하중은 0.5Kg으로 결정하고 이에 대해 각 관절에 걸리는 토크를 계산한다.

위의 초기값을 대입하여 계산한 결과로 로봇 전체의 회전을 담당하는 모터1의 요구토크는 0.1697Nm, 링크1을 구동하는 모터2의 요구토크는 4.4072Nm, 링크2를 구동시키는 모터3의 요구토크는 1.623Nm가 나왔으며, 이것을 토대로 아래의 Table2와 같은 계산결과를 얻었다. Motor는 Maxon사의 DC모터를 선정하였으며[8] 특히, 모터1의 토크는 케이블감속기법의 특성상 모터2와 모터3의 토크를 모두 합산하여 계산하였다. 그 결과 6.1999 Nm의 토크를 얻을 수 있었다.

Table. 2는 각각의 모터의 토크를 구한 표이다. 최종 토크는 ①과 ②를 곱한 값에 모터의 토크를 곱하면 된다. 표에서 요구토크는 안전 여유계수2를 곱한 값이다. 이것은 로봇의 재료를 가공하거나 부품을 제작할 때 예상치 못한 변수에 대비하기 위해서 여유를 두는 것이다. 모터1은 20W, 모터2는 6W 그리고 모터3는 5W로 선정했다.

여기에서 공간의 제약과 용량의 제약으로 일정 부분의 감속은 Maxon사의 기어박스로 대체 하였다. 모터에 기어 감속기를 사용한 이유는 케이블 감속의 방법은 많은 감속비를 얻으려면 그만큼의 풀리가 더 필요하고 그 풀리가 위치할 공간이 필요한데 본 연구의 로봇의 크기는 이를 충족시키지 못하고 어느 정도의 공간의 제약 때문에 많은 양의 감속을 케이블 감속만으로 하기에는 무리가 되었기 때문이다.

Table 2. Capacity of motor and deceleration ratio

Motor	Torque demand	Motor torque	Gear reduction	Cable reduction
Motor1	12.400 Nm	0.240 Nm	20 : 1	6 : 1
Motor2	8.814 Nm	0.0517 Nm	30 : 1	5 : 1
Motor3	3.246 Nm	0.026 Nm	20 : 1	6.5 : 1

3. 실시간 로봇 제어시스템

3.1 실시간 제어시스템

Fig.11은 RT-linux를 바탕으로 설계된 실시간 제어시스템의 전체적인 기능을 나타낸 것이다. 실시간 제어시스템은 크게 3가지로 나뉜다. 첫 번째는 soft real time monitoring으로 이것은 제어 상태, 플랜트(plant)의 입력과 출력을 모니터링하는 기능을 가진다. 두 번째는 soft real time updating으로 이것은 제어기의 이득과 수치를 갱신하는 기능을 한다. 마지막으로 hard real time task가 있는데 이것은 A/D, D/A, 엔코더, 제어루틴, 기구학, 동역학등의 하드웨어와 관련된 작업을 한다.

이런 시스템의 장점은 다른 동일한 기능의 시스템에 비해서 가격이 싸고 사용자가 쉽게 배우고 사용할 수 있으며, 조작자가 원격지에서 특히, 위험하거나 접근하기 어려운 곳에서 모든 제어기의 정보와 변수들을 쉽게 알 수 있는 원격 모니터링을 할 수 있다. 또한 이 시스템은 하드웨어의 종류와의 독립성을 보장함으로 확장성이 좋다. 즉, 컴퓨터의 CPU나 계산기기가 바뀌더라도 쉽게 교체 가능하다. 마지막으로 도시화 되어있어 사용자가 보기 쉽다는 것이다. 사용자가 로봇이나 어떤 플랜트와 인터페이스를 할 때 자료나 각종 정보들을 쉽게 볼 수 있고 명령 또한 쉽게 내릴 수 있도록 화면의 환경이 도시화되어 있다.

Fig.12는 RT-linux 시스템을 전체적으로 표현한

블록 다이어그램이다. 로봇, RT-Linux 그리고 사용자와의 자료이동과 인터페이스를 간단하게 도시화 한 그림으로 로봇은 RT-Linux와 하드웨어 인터럽트를 하고 RT-Linux는 Linux Kernel의 장치 드라이버와 소프트웨어 인터럽트를 한다. RT-Scheduler는 Hard Real Time Task와 Linux Kernel을 스케줄링한다. Hard Real Time Task는 공유메모리를 통해서 Soft Real Time Task로 자료를 전송하고 다시 Soft Real Time Task는 FIFO로 Hard Real Time Task로 자료를 전송한다.

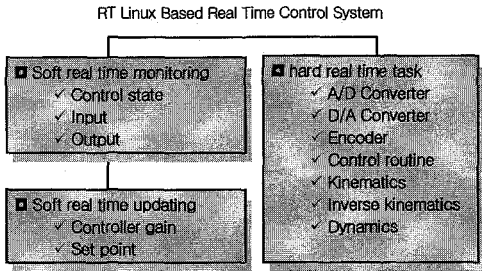


Fig.11 Whole function of Real Time Control system

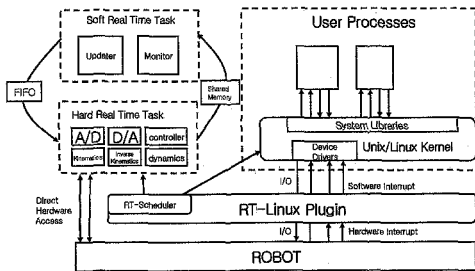


Fig.12 Block Diagram of Real Time linux system

3.2 전체시스템의 하드웨어구성

로봇의 전체 시스템의 구성은 로봇, 구동 드라이버, 엔코더 드라이버, 전원, 인터페이스카드, 컴퓨터 시스템으로 구성된다.

Fig.13는 전체적인 로봇제어 시스템을 나타낸 것이다. 로봇은 전체적으로 두랄루민재질의 3 DOF 로봇이며 여기에는 3개의 모터와 3개의 엔코더가 설치되어 있다. 각각의 모터의 용량을 보면 기저의 모터는 20W, 중간링크의 구동모터는 6W를 사용하였으며, 마지막 링크의 구동을 하는 모터는 5W를 사용하였다. 블록 다이어그램을 보면 로봇이 움직이면서 생기는 위치신호는 로봇에 달린 3개의 엔코더에 의해서 인식된 후에 인터페이스

카드를 통해 컴퓨터로 받아들여져 처리하게 되고 반대로 컴퓨터에서 조작자가 어떤 명령을 내리면 이 명령은 인터페이스 카드를 통해서 로봇으로 전달하는데 그 중간 역할을 s626 DSP보드가 맡고 있다. s626 Card는 48개의 디지털 I/O채널을 가지고 있고 16개의 14 bit 아날로그 입력, 4개의 13 bit 아날로그 출력을 가지고 있다.

s626 DSP보드에서는 -10 ~ +10 까지의 전압을 내 보낸다. 이것을 모터 드라이버에서 받아 모터를 구동하게 되고 엔코더 값이 다시 s626 DSP보드로 돌아와서 위치를 감지하는 것이다.

컴퓨터는 IBM 펜티엄 PC를 사용하였으며 O/S는 RT-linux를 사용했으며, 프로그램은 Emacs 에디터를 사용하여 C언어를 사용하여 코딩했다. 컴파일러는 gcc를 사용했고, 디버거는 gdb를 사용했다.

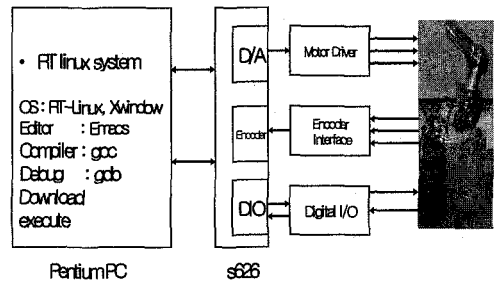


Fig.13 System Block Diagram

3.3 전체시스템의 소프트웨어구성

본 연구에서는 RT-linux를 기반으로 하는 로봇 제어 시스템을 개발하였다. Fig.17은 본 연구에서 개발한 소프트웨어의 구성을 블록 다이어그램으로 나타낸 것이다. GUI RT-Updater에서 사용자가 명령을 주면 제어루틴에서 D/A 컨버터와 디지털 I/O를 통해서 로봇에게 명령이 전달되고, 로봇의 현재 데이터 값들은 파일이나 Standard Out 또는 User Module로 표시된다. 반대로 로봇의 위치 자료는 엔코더 루틴이나 디지털 I/O를 통해서 컨트롤 루틴으로 전달되고 이때의 데이터값들 역시 파일이나 Standard Out 또는 User Module로 표시된다. Fig.14의 괄호 안의 파일명들은 각각의 프로그램 파일명들을 나타낸 것이다.

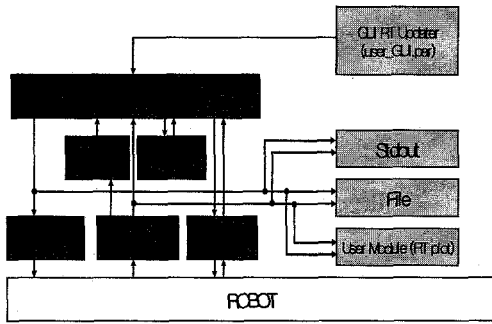


Fig.14 Organization of RT-Software

Fig.15는 GUI RT Updater의 실행 화면을 나타낸 것으로 여기에서는 엔코더 초기화, 위치 초기화, DA Test, PD 제어, PID 제어, TDC 제어 등 여러 가지 작업을 실시간으로 User가 줄 수 있다. 또 Data TX는 데이터 샘플링간격을 정하는 것으로 여기에서 사용자가 원하는 값으로 넣을 수 있다.

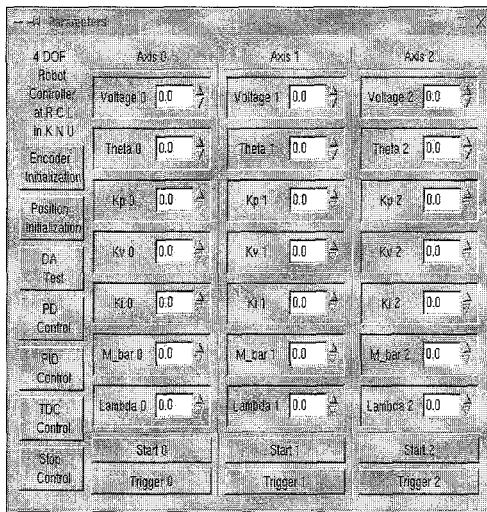


Fig.15 Parameters Window of RT-Window

Fig.16은 데이터의 출력형태를 나타낸 것이다. 샘플링타임을 400Hz로 엔코더에서 나오는 신호와 D/A 컨버터에서 나오는 신호가 실시간으로 표시되는 그림으로 첫 번째 열은 시간(sec)을 나타내며 3, 4, 5 열은 각각 모터1, 모터2, 모터3에 부착된 엔코더의 신호를 나타내는 것이며 7, 8, 9 열은 각각 모터1, 모터2, 모터3에 걸리는 전압 즉 토크를 나타낸다.

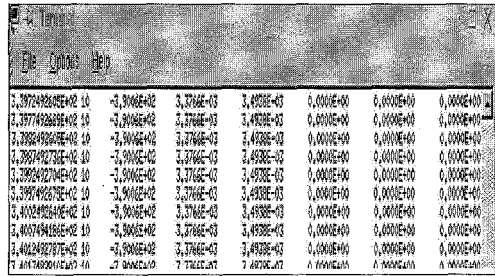


Fig.16 Data Window

Fig.17은 User Module(RT-Plot)으로 설정한 상태의 그림을 나타낸 것으로써 실시간으로 전송되는 Data를 그래프로 나타내기 위해서 만든 것이다. 각각의 로봇의 조인트가 움직이면 움직인 변위만큼 그래프로 연속적으로 표시되는 것이다. 첫 번째, 두 번째, 세 번째 그림들은 각각 모터1, 모터2, 모터3의 변위를 나타내는 그래프들이다. Start를 누르면 그래프가 그려지고 Pause는 일시정지, Stop를 누르면 그려지던 그래프가 끝나게 된다.

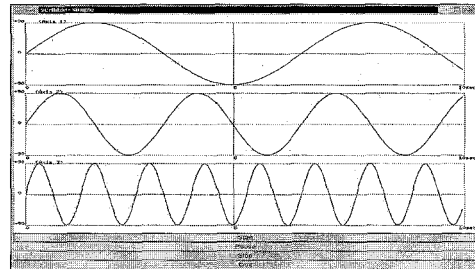


Fig.17 Real Time Plot

4. 로봇제어 실험

Fig.18은 로봇의 위치를 나타낸 그림이며 실험은 Fig.18에 나타난 순서대로 했다. 제어실험은 PD 제어를 했으며 이득값은 모터1, 모터3는 $K_p=2.0$, $K_v=0.1$ 를 주었고 모터2는 $K_p=2.0$, $K_v=0.2$ 를 주어서 실험을 했다.

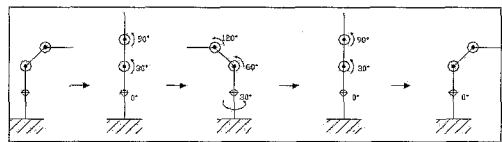


Fig.18 Position of robot

Fig.19는 시간에 대한 로봇의 위치를 나타낸 그래프인데 처음에 초기위치($\theta_1=0, \theta_2=0, \theta_3=0$)에서 임의로 정한 위치 초기상태인 $\theta_1=0, \theta_2=30, \theta_3=90$ 인 위치로 로봇은 위치를 잡는다. 그리고 다시 $\theta_1=30, \theta_2=60, \theta_3=120$ 으로 이동하고 다시 $\theta_1=0, \theta_2=30, \theta_3=90$ 인 상태로 왔다가 $\theta_1=0, \theta_2=0, \theta_3=0$ 인 상태로 되돌아가는 과정을 그래프로 나타낸 것이다. 모터1의 그래프에서 미세한 그래프의 떨림을 볼 수 있는데 이것은 케이블 장력의 영향으로 보인다.

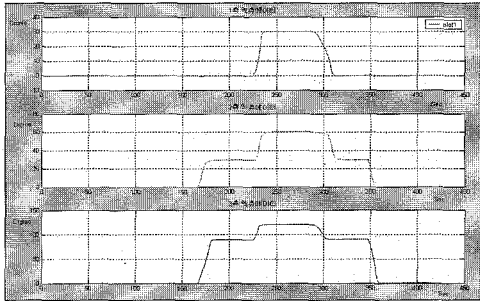


Fig.19 Position graph of robot

Fig.20은 로봇의 모터1, 모터2, 모터3에 가해지는 토크를 나타낸 그래프로 각각의 구동이 있을 때 모터에 걸리는 토크를 나타낸 그래프이다. 실험결과에서도 보이듯이 처음에 움직일 때와 정지할 때 힘을 가장 많이 필요로 하는 것을 알 수 있다.

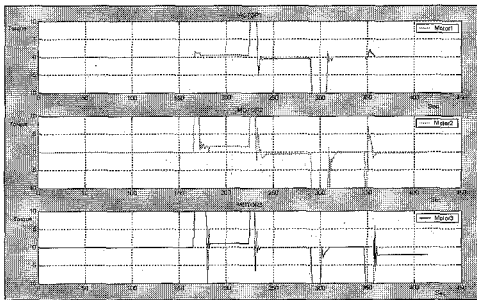


Fig.20 Torque graph of robot

5. 결론

본 연구에서는 조인트 감속기법 중에서 무게가 작으면서 강성과 정밀도를 보장하는 케이블 감속기법을 이용하여 3 자유도 로봇을 설계하고 제작하였다. 본 로봇은 로봇 내부의 구조와 동작을 쉽게 이해할 수 있도록 되어 있으며 크기 또한 250mm×250mm×870mm이고 두랄루민재질의 미니 로봇의 요건을 충족시키고 실제 연구와 교육에 효과적으로 적용될 수 있는 로봇으로 현실적으로 설

계하고 제작하였다. 강성이 향상된 케이블을 사용하고 로봇 부품의 가공 오차를 좀더 줄이면 완전히 해결되지 않은 백래쉬의 문제는 해결되리라 본다.

본 연구에서는 로봇과 연동해서 사용할 수 있는 실시간 제어시스템을 개발하였다. 이 시스템은 실시간 운용에 체계적인 RT-linux 환경 하에서 개발한 것으로 사용자가 편리하게 사용할 수 있는 범용 제어기를 설계하였다. 이번에 개발한 실시간 제어시스템의 특징은 본 연구에서 개발한 로봇뿐만 아니라 다른 기계에도 적용할 수 있으며, 인터넷을 통한 원격제어도 가능하고 리눅스 환경하에서 개발했기 때문에 기존의 상용 실시간 제어시스템에 비해서 경제적인 측면에서도 많은 효과를 거둔 결과를 가져왔다.

본 연구는 실용 및 적용에 관한 논문의 특성을 지니며, 곧바로 현장에 적용될 수 있는 매우 유용한 결과를 얻었다고 생각한다.

참고 문헌

- [1] 박종현, 송재복, 한창수(공역), *로봇공학*, 기술, pp.212-222, 1998.
- [2] James G, Keramas, *로봇공학*, pp 106-127, 2000.
- [3] Saeed B. Niku, *Introduction to Robotics Analysis, Systems, Applications*, Pearson Education, pp.176-185, 2001. ISBN 0 13 061309 6.
- [4] William T. Townsend, *The Effect of Transmission Design on Force-Controlled Manipulator Performance*, MIT Artificial Intelligence Laboratory, pp.161 -170, 1994
- [5] KBC Rolling Bearings Catalogue, (주)HANWHA MACHINERY CO.,LTD., pp.32-68,1995.
- [6] Joha J. Craig, *Introduction to Robotics, MECHANICS AND CONTROL*, Addison Wesley, pp.70-114, 1989. ISBN 0 201 09528 9
- [7] Beer,Johnston, *동역학, 회중당*, pp.550-551, 1992.
- [8] Maxon, *Maxon motor Main Catalogue*, pp.55-10,1 2000.
- [9] NASA Kennedy Space Center as the Final Report, *"Intrinsically Safer Robots"*,1995.
- [10] 홍장표, *기계설계*, (주)복스힐, pp.406-415, 2001.