



저작자표시-동일조건변경허락 2.0 대한민국

이용자는 아래의 조건을 따르는 경우에 한하여 자유롭게

- 이 저작물을 복제, 배포, 전송, 전시, 공연 및 방송할 수 있습니다.
- 이차적 저작물을 작성할 수 있습니다.
- 이 저작물을 영리 목적으로 이용할 수 있습니다.

다음과 같은 조건을 따라야 합니다:



저작자표시. 귀하는 원저작자를 표시하여야 합니다.



동일조건변경허락. 귀하가 이 저작물을 개작, 변형 또는 가공했을 경우에는, 이 저작물과 동일한 이용허락조건하에서만 배포할 수 있습니다.

- 귀하는, 이 저작물의 재이용이나 배포의 경우, 이 저작물에 적용된 이용허락조건을 명확하게 나타내어야 합니다.
- 저작권자로부터 별도의 허가를 받으면 이러한 조건들은 적용되지 않습니다.

저작권법에 따른 이용자의 권리는 위의 내용에 의하여 영향을 받지 않습니다.

이것은 [이용허락규약\(Legal Code\)](#)을 이해하기 쉽게 요약한 것입니다.

[Disclaimer](#)

공 학 석 사 학 위 논 문

공기압 6자유도 시뮬레이터의 실시간 응답 특성 개선



2009년 2월

부 경 대 학 교 대 학 원

제 어 기 계 공 학 과

차 민 구

공 학 석 사 학 위 논 문

공기압 6자유도 시뮬레이터의
실시간 응답 특성 개선

지도교수 장 지 성

이 논문을 공학석사 학위논문으로 제출함



2009년 2월

부 경 대 학 교 대 학 원

제어기계공학과

차민구

차민구의 공학석사 학위논문을 인준함.

2009년 2월



주 심 공 학 박 사 양 주 호 (인)

위 원 공 학 박 사 고 성 위 (인)

위 원 공 학 박 사 장 지 성 (인)

목 차

Abstract	1
제 1 장 서론	3
제 2 장 6자유도 시뮬레이터의 기구학	5
2.1 시뮬레이터의 좌표계	5
2.2 시뮬레이터의 역기구학	7
제 3 장 시뮬레이터의 구성	9
제 4 장 액추에이터의 구성 방정식 과 제어기	12
제 5 장 실시간 제어	12
5.1 기초 개념	12
5.1.1 Real-time(실시간)	12
5.1.2 Real-time OS	15
5.1.3 Hard real-time and Soft Real-time	16
5.2 Labview real-time 시스템	18
제 6 장 시뮬레이터의 3D CAD 모델링과 VRML	21
제 7 장 실험 및 고찰	22
7.1 응답 속도 비교	22
7.2 시뮬레이터의 제어 성능	24
제 8 장 결 론	29
참고 문헌	30

Improvement of Real-time Response Characteristics of Pneumatic 6 DOF Simulator

Min Gu Cha

Department of Control & Mechanical Engineering,

The Graduate School,

Pukyong National University

Abstract

Stewart platform is a common form of parallel manipulator with 6 linked actuator. It was first introduced in 1965 by Stewart for flight simulator. Stewart platform have shape of a closed loop with a high structural rigidity than serial manipulator. The flight simulator have the advantage of safe and effective training

Recently Stewart platform is used in various fields like aircraft, automobiles, ships, virtual reality games and more. Electric and hydraulic actuator are used primarily rather than air pneumatic actuator using the simulator. Pneumatic cylinder be behind at position control performance, but acceleration and deceleration performance is excellent, in load weight connected at upper platform.

3-axis joystick for the operation with simulator, industrial potentiometer joystick can operate movement of 6dof in a not particular OS. Generally, the voltage, such as signals in the form of sensors can connect directly to digital controller configurations.

In recent years the development of microprocessor-based digital controller is usually done. However, Because of the complexity of the structure than

analog controller, a variety of knowledge for hardware and software is required.

The digital controller must be designed in real-time. Main elements are sensor, controller of the hardware (CPU), software and operating system, control algorithms.

In this paper, to improve Real-time Response Characteristics of digital controller, software programing method and algorithms are reviewed. To reduce the response time, control loop have priority than another task. Labview real-time target system with RT OS can use shared variable for data communication



제 1 장 서 론

병렬형 매니퓰레이터(parallel manipulator)의 대표적인 형태인 스튜어트 플랫폼은 병렬로 연결된 6개의 액추에이터를 이용한 병렬 운동기구로 비행시뮬레이션을 위하여 Stewart에 의해 1965년에 처음 소개되었다. 이 스튜어트 플랫폼은 직렬형 매니퓰레이터(serial manipulator)에 비해 폐루프(closed loop)의 형상을 가지고 있으므로 구조적으로 높은 강성도를 가져, 외부와의 접촉 시 큰 접촉력이 발생하는 경우 혹은 비교적 무거운 물체의 조작 등에 사용되어질 수 있다.⁽¹⁾

병렬형 매니퓰레이터는 상기의 특성을 활용하여 주로 비행시뮬레이션에서 비행기 조종사의 안전하고 효과적인 훈련을 위해 이용되어 왔으며, 최근에 와서는 항공기는 물론, 자동차, 선박, 가상현실 게임 등 다양한 분야에서 이용되어지고 있다.

본문에서는 일반적으로 제작되어지는 전기구동 모터나 유압 액추에이터를 이용한 시뮬레이터가 아닌, 공기압 액추에이터를 이용한 시뮬레이터를 대상으로 하였다. 공기압 액추에이터는 자세제어의 정밀성은 떨어지지만, 저부하의 경우에는 우수한 가·감속 성능으로 인하여 급격한 가속도의 변화에 빠르게 추종할 수 있다.

본 논문에서 이용한 조이스틱은 특정 OS 에서 작동되는 조이스틱이 아닌 산업용 전위차계 조이스틱이므로 제어신호를 일반적 센서와 같이 전압형태로 디지털 제어기에 바로 연결할 수 있으므로 구성에 장점이 있다.

최근에는 이러한 매니퓰레이터를 제어하기 위해서 마이크로프로세서를 이용한 디지털 제어기의 개발이 주류를 이루고 있다. 하지만 제어기의 구조가 기존 아날로그 제어기에 비해 매우 복잡하므로 하드웨어 및 소프트웨어 전반의 다양한 지식이 요구되어진다. 디지털 제어기 시스템 전체가 실시간으로 동작하도록 하기위한 주요 요소는 제어기의 하드웨어(CPU), 그리고 센서, 제어 소프트웨어 및 운영체제, 제어 알고리즘 등이 있다.

본 논문에서는 기존에 제작된 시뮬레이터에 조이스틱을 이용하여 시뮬레이터의 6자유도 운동을 조작할 수 있도록 하였다. 이러한 조이스틱과 연계하여 시뮬레이터의 실시간 응답 특성을 개선하기 위해 소프트웨어와 제어 알고리즘⁽³⁾을 설계하여 검토하였고, 제어기의 응답속도를 최대한 향상시키기 위해, 실시간(Real-time) 운영체제와 함께 실시간 Target과

Host를 이용하였다.



제 2 장 6자유도 시뮬레이터의 기구학

2.1 시뮬레이터의 좌표계

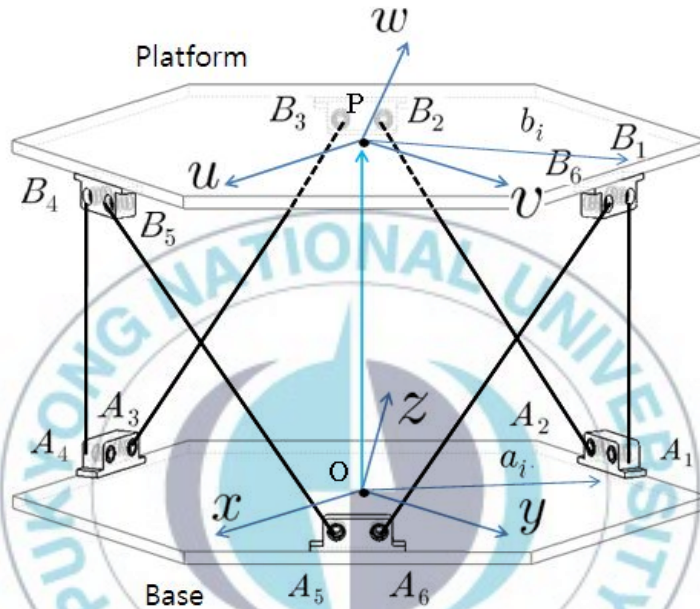


Fig. 1 Coordinate system of 6 DOF simulator

A_i : 하판 조인트 좌표[m], B_i : 상판 조인트 좌표[m], a_i : 하판 조인트의 위치 벡터, b_i : 상판 조인트의 위치 벡터, d_i : 축의 길이[m], O : 하판의 중심 좌표[m], P : 상판의 중심 좌표[m]

$$\begin{aligned} A_i &= [a_{ix} \ a_{iy} \ 0]^T \quad i = 1, 2, \dots, 6 \\ B_i &= [b_{ix} \ b_{iy} \ 0]^T \quad i = 1, 2, \dots, 6 \end{aligned} \quad (2.1)$$

상부에 있는 평판을 플랫폼이라고 하고, 하부에 위치한 평판을 베이스라 한다. 베이스는 고정되어있고 6개의 액추에이터의 의해 6자유도

운동을 한다. Fig. 1은 스튜어트 플랫폼의 좌표계를 보여준다. 고정된 베이스의 직각 좌표계를 베이스 좌표계라고 플랫폼의 직각 좌표계를 플랫폼 좌표계라고 한다. 여섯 개의 액추에이터는 B_i 와 A_i 에서 유니버설(Universal) 조인트에 의해 베이스와 플랫폼을 연결하고 있다. 베이스 좌표계와 플랫폼의 좌표계를 나타내는 Cartesian Coordinate System인 $A(x,y,z)$, $B(u,v,w)$ 를 설정하여, 플랫폼의 X,Y,Z 축 방향의 변진운동을 u,v,w 라 하고, X,Y,Z 축에 대한 회전각을 Roll(r), Pitch (p), Yaw(y) 로 정의한다. 상판과 하판 각각의 중심을 연결하는 위치벡터 P 와 회전변화 행렬(Rotation Matrix) $R(r,p,y)$ 로써 직선 및 회전 운동을 표현한다.

회전변화 행렬(Rotation Matrix)는 다음과 같이 나타낼 수 있다.

$$R(r,p,y) = R(X,r)R(Y,p)R(Z,y) = \begin{bmatrix} u_x & v_x & w_x \\ u_y & v_y & w_y \\ u_z & v_z & w_z \end{bmatrix} \quad (2.2)$$

식 (2.2)에서

$$\begin{aligned} u_x &= \cos(y) \cos(p) \\ v_x &= \cos(y) \sin(p) \sin(r) - \sin(y) \cos(r) \\ w_x &= \cos(y) \sin(p) \cos(r) + \sin(y) \sin(r) \\ u_y &= \sin(y) \cos(p) \\ v_y &= \sin(y) \sin(p) \sin(r) + \cos(y) \cos(r) \\ w_y &= \sin(y) \sin(p) \cos(r) - \cos(y) \sin(r) \\ u_z &= -\sin(p) \\ v_z &= \cos(p) \sin(r) \\ w_z &= \cos(p) \cos(r) \end{aligned} \quad (2.3)$$

위의 식 (2.2)에 나타낸 회전변화 행렬의 각 요소는 다음의 직각조건을 만족해야 한다.

$$\begin{aligned}
u_x^2 + u_y^2 + u_z^2 &= 1 \\
v_x^2 + v_y^2 + v_z^2 &= 1 \\
w_x^2 + w_y^2 + w_z^2 &= 1 \\
u_x v_x + u_y v_y + u_z v_z &= 0 \\
u_x w_x + u_y w_y + u_z w_z &= 0 \\
v_x w_x + v_y w_y + v_z w_z &= 0
\end{aligned} \tag{2.4}$$

고정 플랫폼에 대한 무빙 플랫폼의 중심좌표 p 와 6자유도 운동 q 는 다음 식과 같이 표현된다.

$$\begin{aligned}
p &= [u \ v \ w]^T \\
q &= [u \ v \ w \ r \ p \ y]^T
\end{aligned} \tag{2.5}$$

2.2 시뮬레이터의 역기구학

6 자유도 운동으로부터 각 다리의 길이를 구하는 문제는 플랫폼의 모션으로부터 실린더의 길이를 결정하는 문제이다. 플랫폼의 특정운동을 위한 실린더의 길이를 결정하기 위해 사용된다.

Fig. 1에 각각 $a_i = [a_{ix} \ a_{iy} \ a_{iz}]^T$ 와 $b_i = [b_{ix} \ b_{iy} \ b_{iz}]$ 로 표현되는 베이스와 무빙플랫폼의 조인트 A_i 와 B_i 의 위치벡터는 시뮬레이터의 i 번째 축에 대한 벡터 방정식은 다음과 같다.

$$\overline{A_i B_i} = p + R b_i - a_i \tag{2.6}$$

시뮬레이터의 i 번째 축의 길이 d_i 는 벡터 $\overline{A_i B_i}$ 를 이용해서 다음식과 같이 표현할 수 있다.

$$d_i^2 = [p + Rb_i - a_i]^T [p + Rb_i - a_i], \quad \text{for } i = 1, 2, \dots, 6 \quad (2.7)$$

따라서, 각 액츄에이터의 길이는 $\overline{A_i B_i}$ 의 내적으로 계산해 낼 수있으며 식 (2.7)을 이용하면 각 축의 길이 d_i 를 다음식과 같이 구할 수 있다.

$$d_i^2 = p^T p + b_i^T b_i + a_i^T a_i + 2p^T Rb_i - 2p^T a_i - 2[Rb_i]^T a_i$$

$$d_i = [p^T p + b_i^T b_i + a_i^T a_i + 2p^T Rb_i - 2p^T a_i - 2[Rb_i]^T a_i]^{\frac{1}{2}} \quad (2.8)$$

for $i = 1, 2, \dots, 6$

무빙 플랫폼의 중심좌표 $p(u \ v \ w)$ 와 회전각 $(r \ p \ y)$ 을 실시간으로 모니터를 통해 입력시키면 역기구학으로 계산된 각 축의 길이만큼 무빙 플랫폼이 움직이고 각 축에 연결된 위치 센서로 실제 움직인 축의 길이를 실시간 모니터링 하여 목표치와 실험결과를 비교하였다.

제 3 장 6자유도 시뮬레이터의 구성

3.1 시뮬레이터의 기본사양

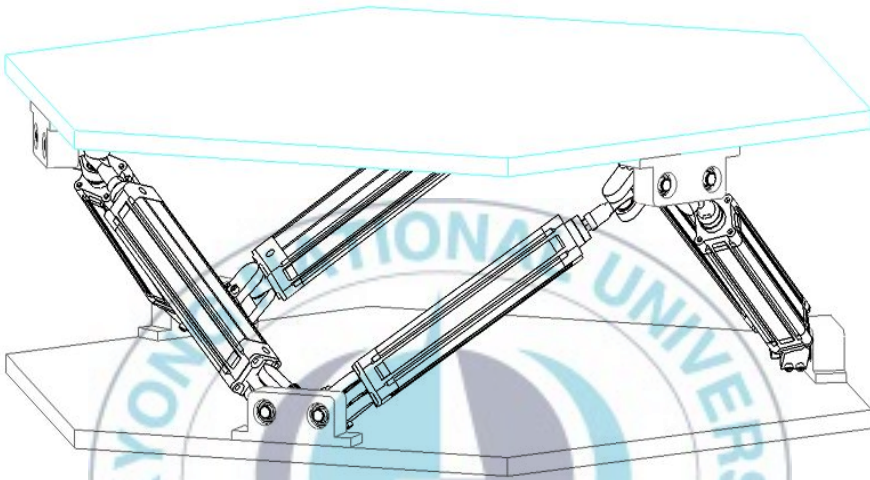


Fig. 2 3D CAD of 6 DOF simulator

시뮬레이터는 Fig. 1과 같이 한 면이 810[mm]인 알루미늄 정육면체 플랫폼과 베이스를 연결하는 여섯개의 공압 실린더로 구성되어 있다.

조이스틱(Fig. 3 참조)과, 센서가 PXI의 DAQ카드와 연결되어 조이스틱 입력신호를 받아 목표위치를 설정한다. 실린더가 구동하면 플랫폼이 6자유도를 하게되고 실린더의 위치센서에서 신호를 PXI로 보낸다.

PXI는 NII사의 Real-time OS로 구동되며 Real-time Target이 되어 Host인 컴퓨터로 데이터를 전송하여 모니터링을 가능하게 한다.

시뮬레이터의 기본사양을 Table 1에 나타낸다. Fig. 4 는 시뮬레이터의 사진을 나타낸다.

Table 1 Specification of simulator

플랫폼	Maker : SNI - Quality of material : aluminum alloy - Form : regular hexa-hedron
공기압 실린더	Model : TCDA2CD63-300, Maker : TPC - Pressure range : 0 ~ 9.9 kgf/cm ²
위치 센서	Model : LT-M-300-s, Maker : GEFTRAN - Displacement speed 0 ~ 5 m/s - Maximum applicable voltage : 10V
유량 제어 밸브	Model : MPYE-5-1/4-010-B, Maker : FESTO - Electrical setpoint : 0 ~ 10 V DC
PXI	Maker : NI Model : PXI-1042Q DAQ device : NI 6229 Resolution(in/out) : 16 bits
제어 및 계측 S/W (Labview 8.5)	Maker : NI Model : Labview 8.5 Real-time Module Labview Real-time OS
Joystick	Maker : CH Products Model : PXI-1042Q Axis : X, Y, Z

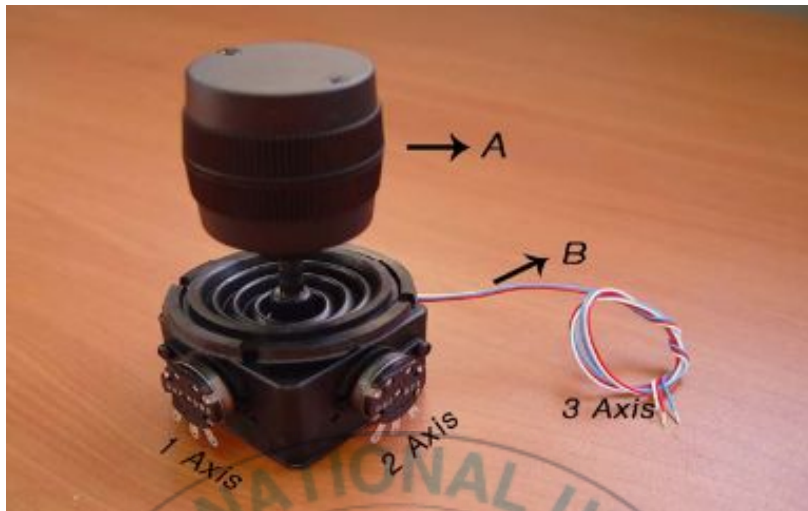


Fig. 3 Potentiometer Joystick



Fig. 4 6 DOF simulator

제 4 장 시뮬레이터 구동계의 구성 방정식과 제어기

4.1 구동계의 구성 방정식

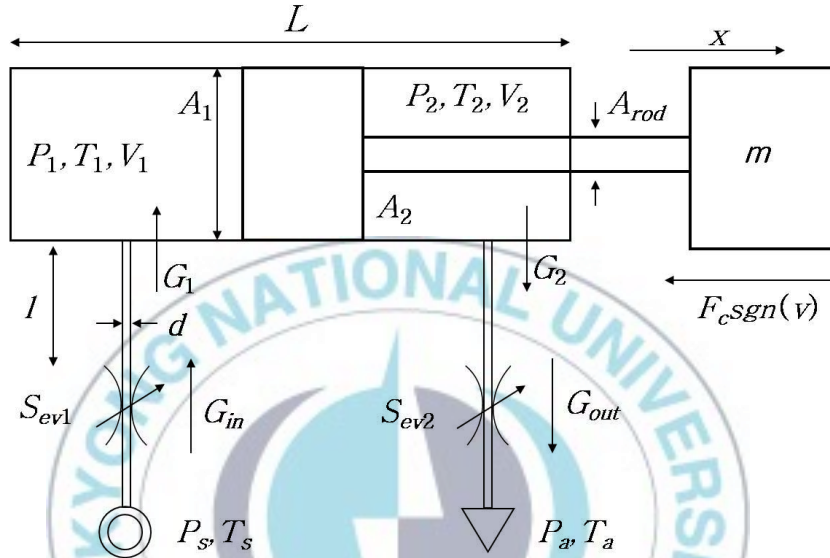


Fig. 5 Schematic diagram of a pneumatic cylinder to control 6 DOF simulator

플랫폼을 제어하기 위하여 사용한 공기압 실린더 구동장치의 전달함수를 구하면 다음과 같다.

$$\frac{x(s)}{u(s)} = \frac{k_{nc} w_{nc}^2}{s(s^2 + 2\zeta_c w_{nc} s + w_{nc}^2)} = H(s)$$

$$k_{nc} = \frac{(A_1 + A_2) k_q R T_a}{(P_{10} A_1^2 + P_{20} A_2^2)}, \quad 2\zeta_c w_{nc} = \frac{b}{m}$$

$$w_{nc}^2 = \frac{\kappa (P_{10} A_1^2 + P_{20} A_2^2)}{m (V_d + V_0)} \quad (4.1)$$

u : 제어 입력[V], k_q : 제어 입력·유량 계수[(kg/s)/V], 첨자 0: 평형

상태

시물레이터를 제어하기 위한 공기압 구동장치의 물리 파라미터를 Table 2에 표시하였다.

Table 2 Physical parameter of a pneumatic cylinder driving apparatus

Parameter	Value	Parameter	Value
m	16[kg]	b	50[N/(m/s)]
A_1	$3.1172 \cdot 10^{-3}[\text{m}^2]$	P_s	501300[Pa]
A_2	$2.8030 \cdot 10^{-3}[\text{m}^2]$	P_{10}	334200[Pa]
x_0	0.142[m]	P_{20}	319279[Pa]
L	0.3[m]	S_e	Max. 18.46[mm ²]
V_0	$4.426 \cdot 10^{-4}[\text{m}^3]$	k_q	$4.376 \cdot 10^{-3}[\text{kg/s/V}]$
R	287[J/kg·K]	T_a	293[K]
d	0.0065[m]	κ	1.4

4.2 제어기

PDD²를 이용하여 실시간 제어 시스템을 설계하였다.

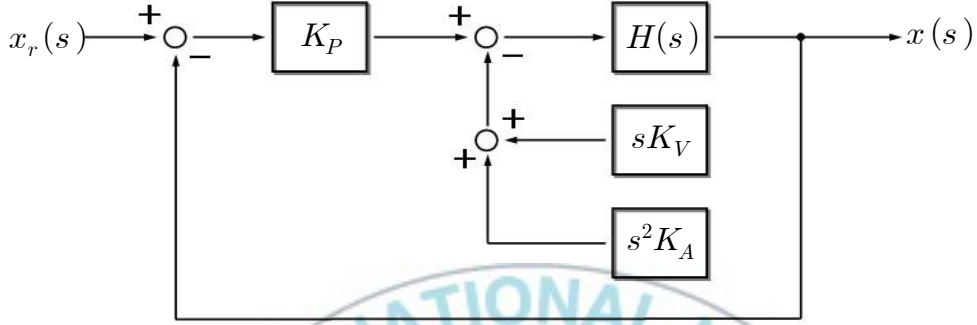


Fig. 6 PDD² controller

식 (4.1)과 PDD²제어기를 이용하면 제어계의 폐루프 전달함수를 다음 식으로 표현할 수 있다.

$$\frac{x(s)}{x_r(s)} = \frac{K_P K_{nc} w_{nc}^2}{s^3 + s^2(2\zeta_c w_{nc} + K_A K_{nc} w_{nc}^2) + s(w_{nc}^2 + K_V K_{nc} w_{nc}^2) + K_P K_{nc} w_{nc}^2} \quad (4.2)$$

$$= H_{ctrl}(s)$$

식 (4.1)에서 K_P [V/m]는 비례 이득, K_V [V/(m/s)]는 속도 이득, K_A [V/(m/s²)]는 가속도 이득을 나타낸다.

비례 이득 K_P 를 이용해서 실린더 구동계의 이득 여유와 위상여유를 각각 12[dB], 60[deg]로 설정하면 속도 이득 K_V 와 가속도 이득 K_A 를 다음 식으로 표시할 수 있다.⁽¹⁰⁾

$$K_V = \left(3 \cdot \sqrt[3]{(k_n w_n^2 K_P)^2} - w_n^2 \right) / (k_n w_n^2) \quad (4.2)$$

$$K_A = \left(2 \cdot \sqrt[3]{(k_n w_n^2 K_P)^2} - 2\zeta w_n \right) / (k_n w_n^2) \quad (4.3)$$

제 5 장 실시간 제어

5.1 기초개념

5.1.1 Real-time(실시간) 응답특성

실시간 응답특성은 응답시간(Response Time)에 대한 요구사항을 피제어 대상이 만족시키는 특성이라고 정의할 수 있다. 즉, Real-time은 ‘정확한’이나 ‘즉시’라는 의미로 이해할 수 있다.

피제어대상이 실시간 응답 특성을 만족시키지 못한다면 그 시스템은 어떤 형식으로든 실패(Failure)하게된다. 즉 실시간 시스템이란 어떤 시스템의 기능이 정확성뿐만 아니라 시간적인 요소도 함께 고려되는 시스템이다. 실시간 시스템의 핵심은 ‘시간’이고 ‘적정시간’ 또는 ‘마감시간’을 준수하느냐 하지 못하느냐 하는 것이 성공과 실패의 중요한 갈림길이다.

실시간 응답특성을 구현하기 위한 핵심적인 내용은 일정한 시간주기로 제어동작을 수행하며, 주어진 시간 안에 제어 동작을 위한 준비를 완료하는 것이다. 이와 같은 조건은 제어대상 시스템 및 제어 요구 사항에 따라 엄격하게 적용될 수도 있고 약간의 오차를 허용할 수도 있다.

디지털 제어기의 경우 설계자는 시스템 전체가 실시간성을 보장하도록 설계하여야 한다. 여기에 포함되는 주요 요소는 센서, 제어기의 하드웨어(CPU) 소프트웨어 및 운영체제, 제어 알고리즘 등이며, 전체 시스템에 대한 성능 요구 사항도 중요한 설계 기준이된다.

5.1.2 Real-time OS

일반적인 OS에서는 자원(예를 들어, 메모리, 하드 디스크 등등)을 얼마나 효율적으로 낭비 없이 쓸 것인가에 초점이 맞춰져 있다. 그러나 real-time OS에서는 효율적인 것도 중요하지만 그 속도에 좀 더 신경을 쓴다. 즉, 일반 OS에서는 메모리라는 자원을 좀 더 효율적으로 쓰려는 의도가 있는 것이고 Real-time OS에서는 약간의 메모리 낭비를 감수하고서라도 시간 제약을 맞추려는 의도가 있는 것이다

5.1.3 Hard real-time and Soft Real-time

real-time system중에는 어떤 시간 안에 반드시 처리해야 하며 그 시간이 지난 후의 결과 값은 정확해도 소용이 없는 경우가 있다. 적정시간 또는 한계시간을 초과해서 응답이 나온다면 시스템을 실패한 것으로 간주하다. 이런 시스템을 hard real-time system이라고 한다. 이에 반해서 어떤 시간 안에 처리하면 좋지만 그렇지 못한 경우에 그 시간에서 약간 경과한 후의 값도 인정하는 경우를 soft real-time system이라고 한다.



Fig. 7 Hard real-time VS soft real-time

Hard real-time에 비해 Soft-real-time 시스템은 수행결과가 적정시간, 한계시간을 초과한다고 해도 정도를 따져 성공과 실패 사이에 4단계를 나누어 결과를 평가한다.

실제 제어기에 실행 중 샘플링 타임과 컨트롤 루프 실행주기의 시점 차이가 발생할 수 있다. 이러한 시점차이를 흔히 Jitter(Fig 8 참조)라고 부른다.

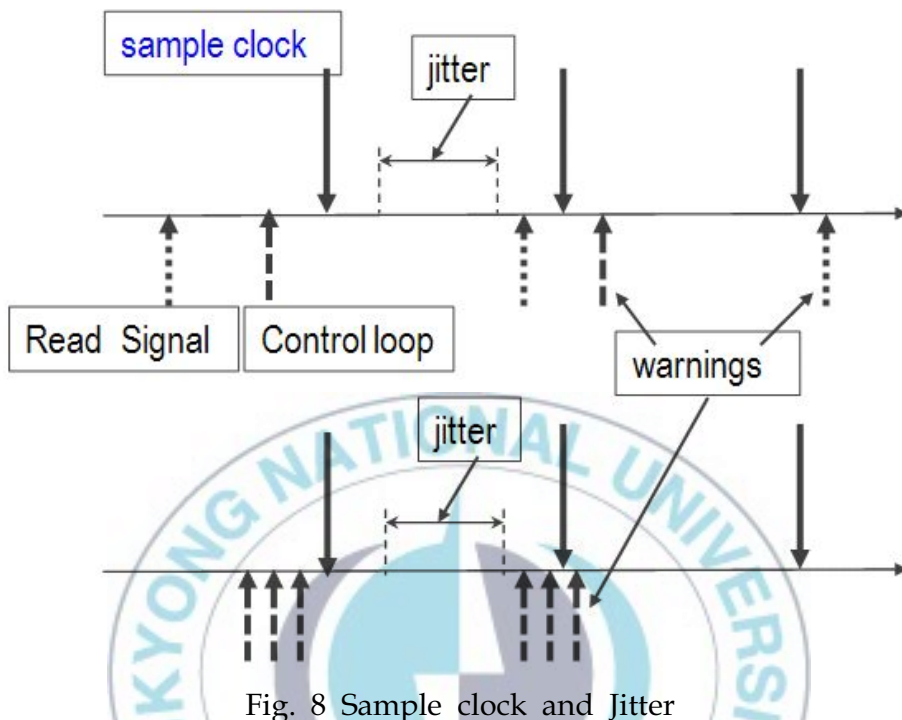


Fig. 8 Sample clock and Jitter

이러한 Jitter를 줄이기 위해서 특정 루프에 우선순위를 부여하는 방법이 있다. 우선순위가 부여된 루프에는 루프에 지정된 특정 시간에 동작하도록 다른 실행 task(실행을 위한 최소단위)들을 정지 시키고 특정 task만을 실행하게 할 수 있다. Fig. 8에서는 우선순위로 인한 Jitter의 예방 방법을 알아 볼 수 있다.

본 논문에서는 고정 우선순위 스케줄링 방식 (static priority scheduling method)을 가지는 soft-time 실시간 제어시스템을 구성하였다.

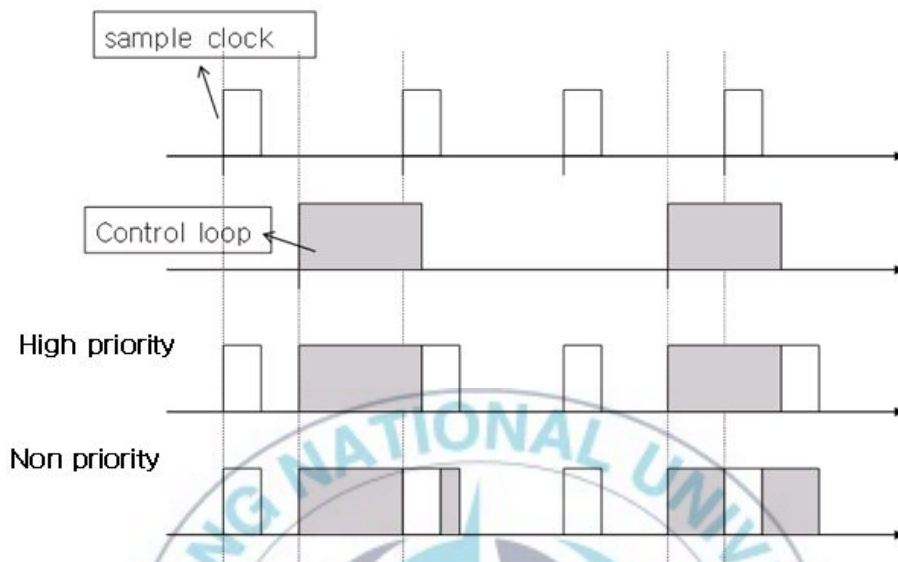


Fig. 9 High priority and Non priority

5.2 Labview real-time 시스템

Real-time control system의 구성을 Fig. 10 에 나타내었다. PXI가 Real-time Target이 되어 입출력 신호를 DAQ 디바이스를 이용해 처리하고, 모니터링을 위한 결과 Data는 공유변수 엔진을 통해 NI Publish-Subscribe Protocol를 이용 RT Target에서 Host computer로 전송되거나, Target 내에 프로그램 사이에 데이터 전송과 처리에 이용되어진다. 조이스틱 입출력을 담당하는 프로그램과 컨트롤 루프를 포함하는 프로그램을 분리해서 설계하여 각각의 우선순위를 부여할 수 있도록 하였다. 그리하여 컨트롤루프는 높은 우선순위를 갖게 되어 원하는 주기와 응답시간을 기대할 수 있게된다.

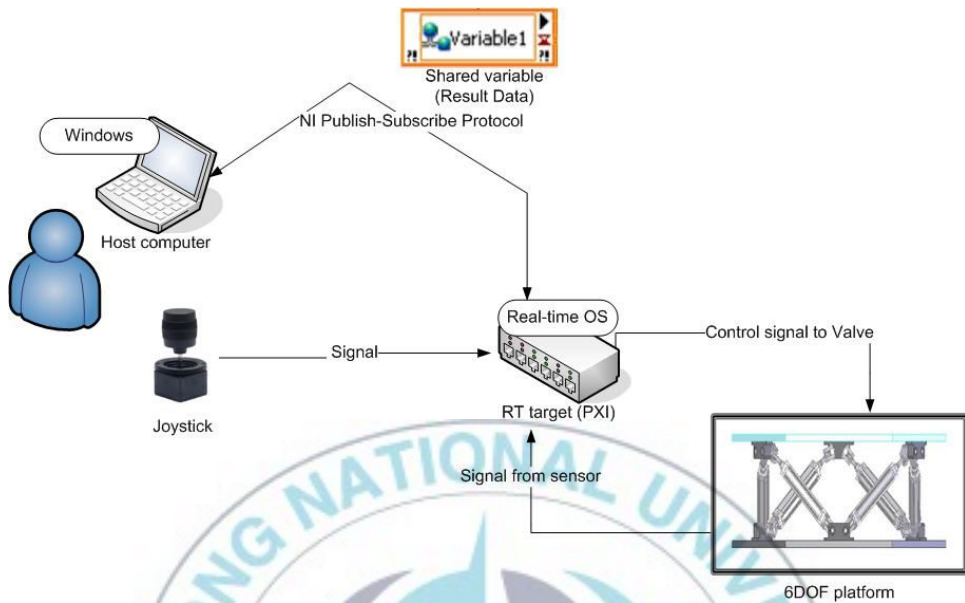


Fig. 10 schematic diagram of Real-time control System



Fig. 11 Real-time Execution Trace Toolkit

Fig.11은 실제 프로그램 실행 후 Real-time Execution Trace Toolkit을 이용하여 실행된 VI와 쓰레드들의 실행순서 및 실행시간 등을 확인한 결과를 나타낸다. 그림으로부터 스케줄링을 쉽게 확인할 수 있다.

제 6 장 시뮬레이터의 3D CAD 모델링과 VRML

Solidworks를 이용하여 시뮬레이터를 각 Part별 CAD로 디자인하여 전체 6자유도 시뮬레이터를 3D CAD 모델링하였다.

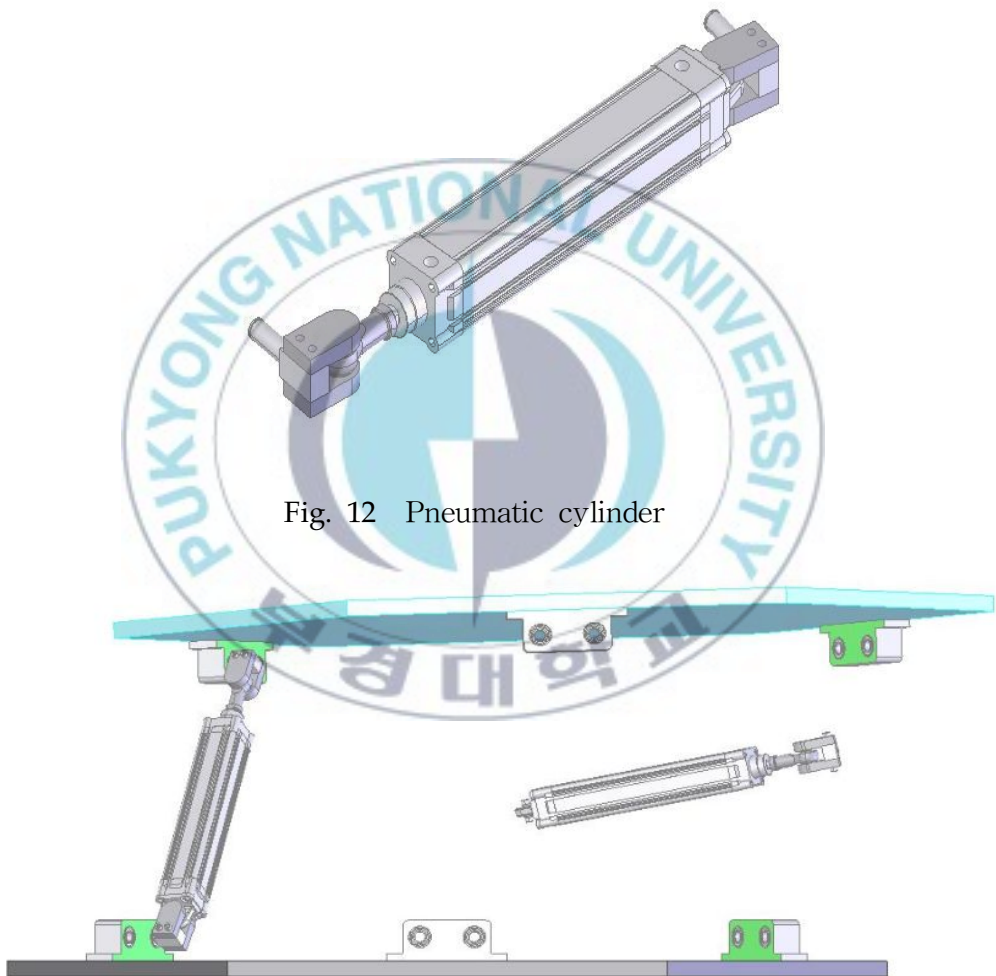


Fig. 12 Pneumatic cylinder

Fig. 13 Pneumatic cylinder with plate

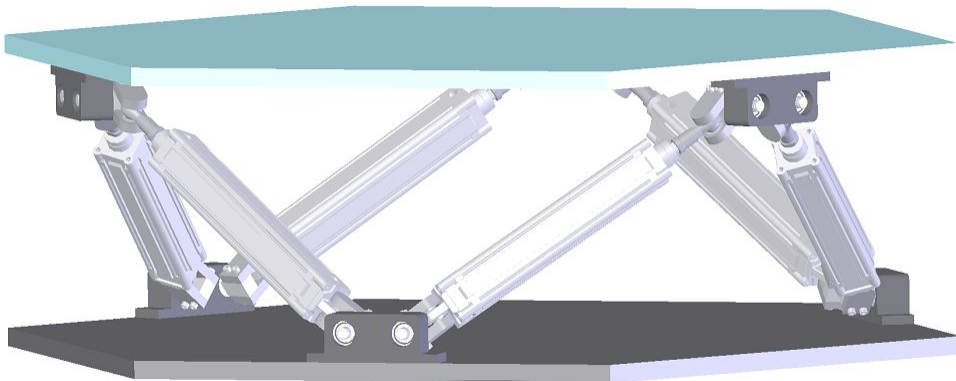


Fig. 14 6 DOF simulator with pneumatic cylinder

완성된 CAD파일을 VRML파일로 바꾸어, 시뮬레이터의 시스템 출력 Display에 이용하였다. VRML은 virtual reality modeling language의 약자로 인터넷상에서 3차원 가상현실 공간을 구현하기 위해서 개발되었다. Text기반으로 작성되어져 처리가 용이하며, 영상의 빠른 구현을 가능하게 해준다.

제 7 장 실험 및 고찰

7.1 응답 속도 비교

본 절에서는 기존에 제작된 컨트롤 루프의 응답 시간과 본 논문에서 설계한 컨트롤 루프의 응답시간을 비교해 보았다. Fig. 15에 나타난 결과와 같이 기존에 제작된 컨트롤 루프의 응답시간을 설정된 샘플링 시간에 불구하고 1ms 와 3ms 사이에 넓게 퍼져있는 것을 알 수 있다. 이러한 현상은 실제 제어기의 실행 시 응답시간이 급격히 증가하는 형태로 나타난다. Fig. 15에서는 특별히 컨트롤 루프의 기준이 되는 응답 시간을 찾을 수가 없다.

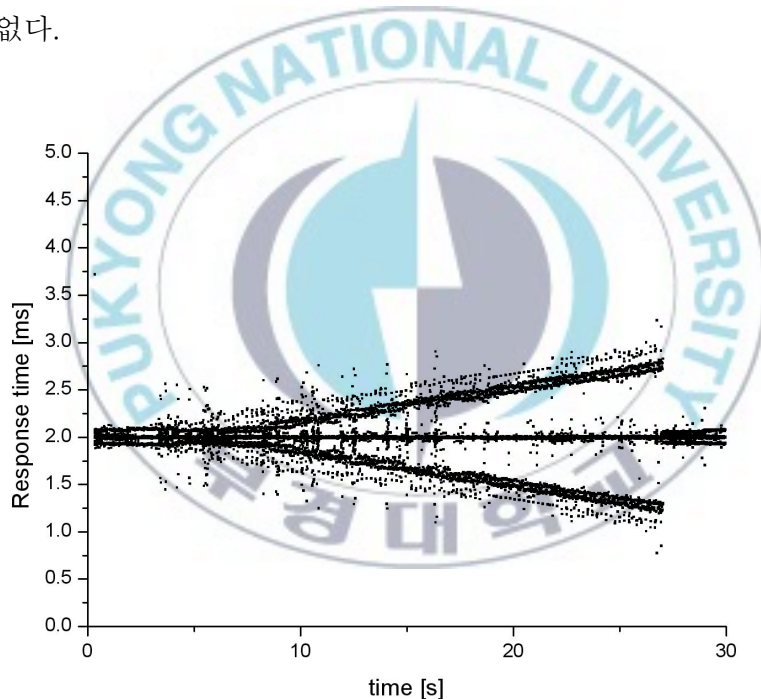


Fig. 15 Response time of Non priority control loop

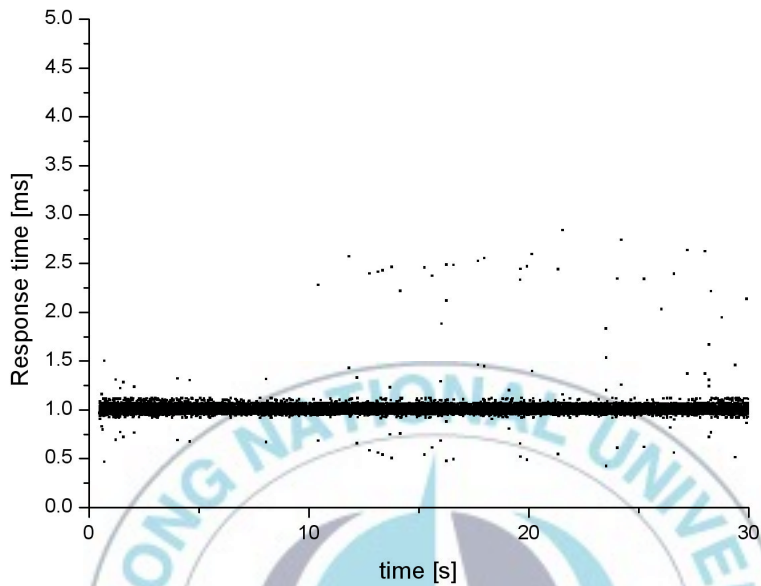


Fig. 16 Response time of High priority control loop

Fig. 16은 본 논문에서 설계한 예서는 높은 우선 순위를 부여한 컨트롤 루프의 응답시간을 보여주고 있다. 샘플링 시간은 1ms로 설정하였다. 그림으로부터 응답시간이 1ms를 중심으로 집중되어 있으며 간간히 발생하는 시간 초과는 보여지지만 매우 미미한 수준임을 알 수 있다. 엄격한 마감시간을 적용하지 않는 Soft real-time 시스템으로서는 충분한 응답 특성을 가지고 있음을 알 수 있다.

7.2 시뮬레이터의 제어 성능

X, Y, Z 축을 $(0, 0, 0.75)[m]$ 로 고정한 채 롤(Roll) 방향으로 $0.13 [rad]$ 회전시켜서 기존 제어기와 비교해 Fig 17 ~ 22 에 나타내었다. 역기구학을 이용하여 액추에이터의 길이로 변화시킨 후 이를 목표치로 제어한 결과를 나타낸다.

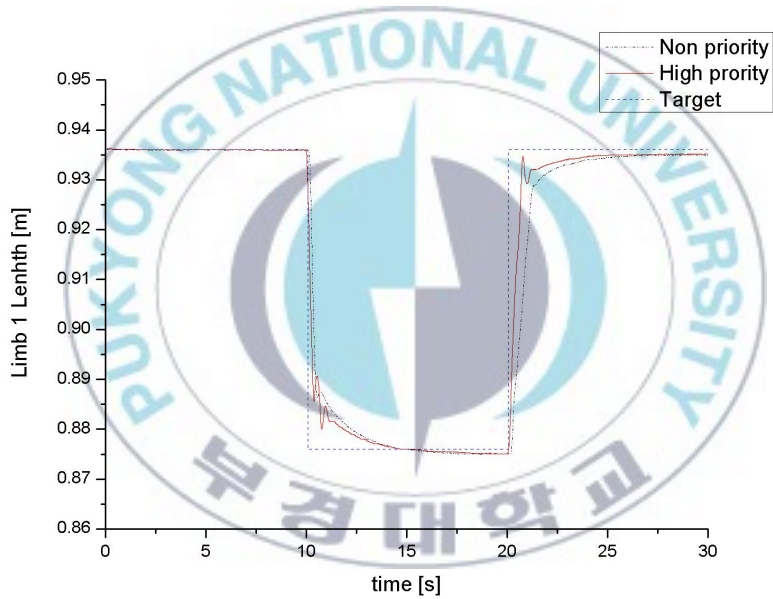


Fig. 17 Length following performance of Limb 1

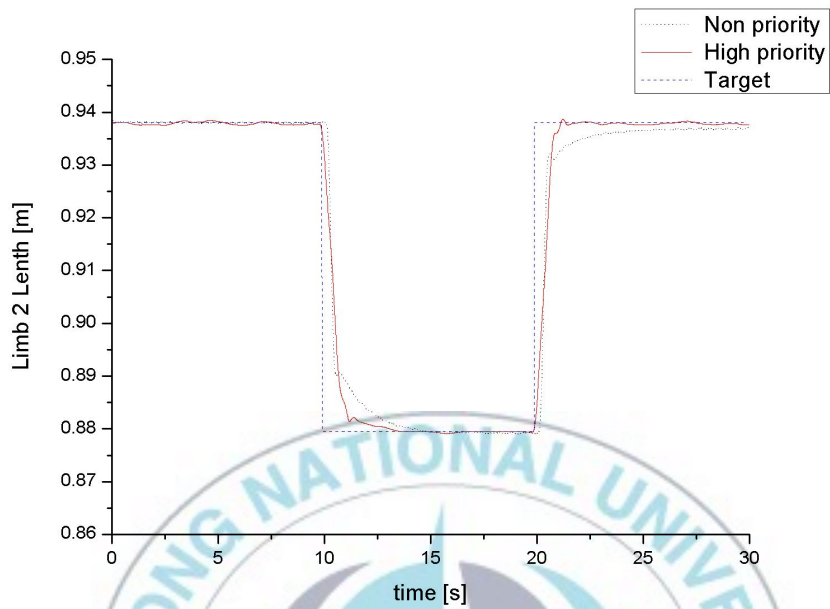


Fig. 18 Length following performance of Limb 2

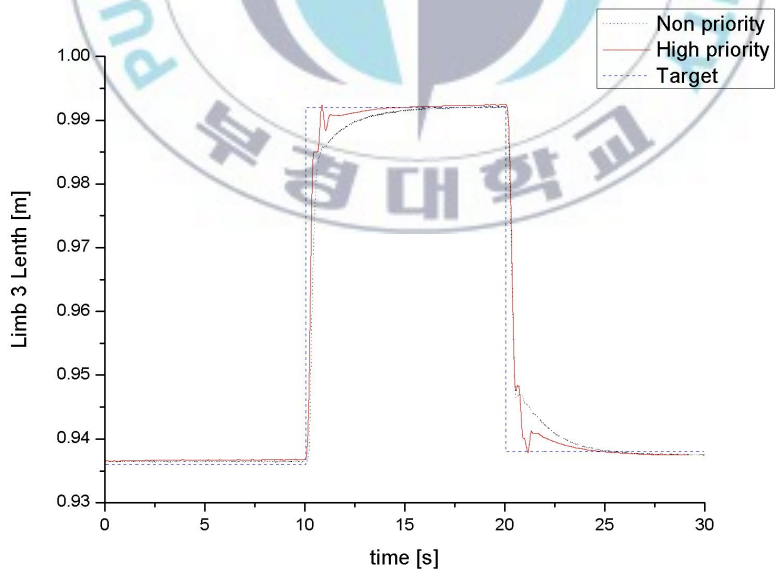


Fig. 19 Length following performance of Limb 3

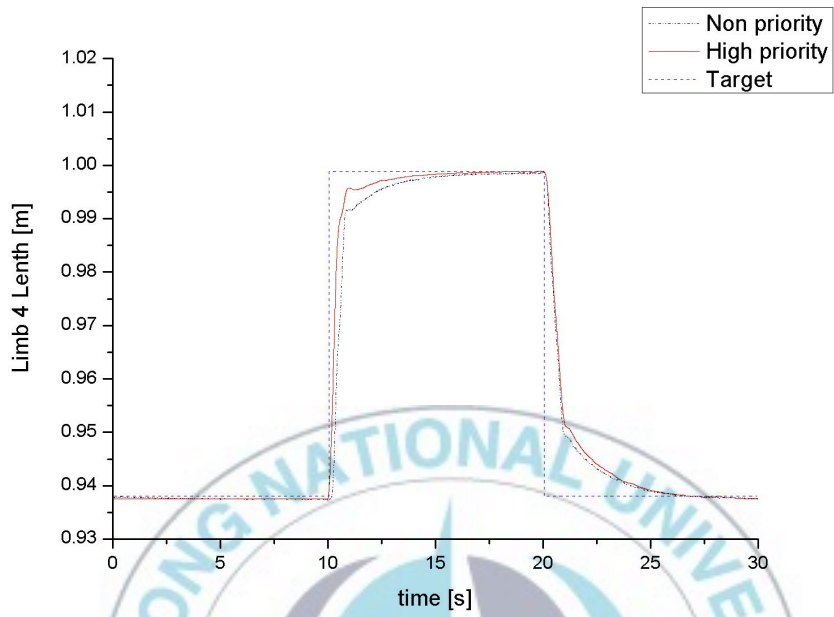


Fig. 20 Length following performance of Limb 4

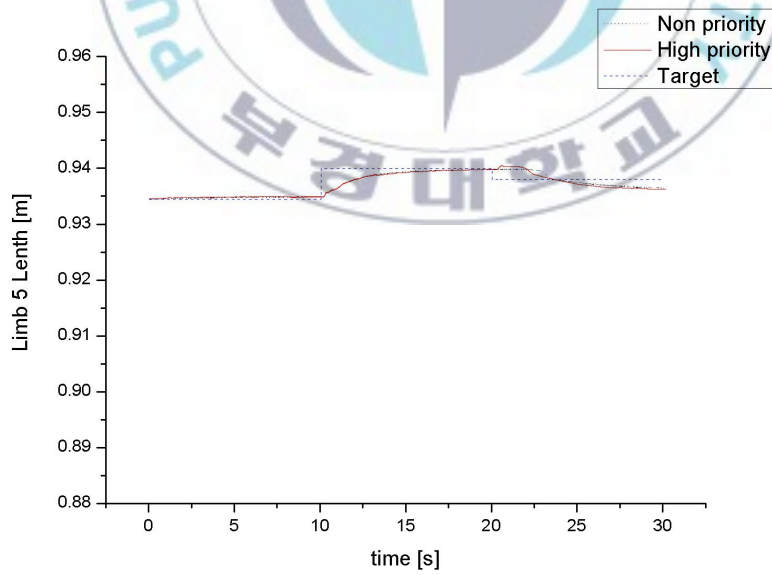


Fig. 21 Length following performance of Limb 5

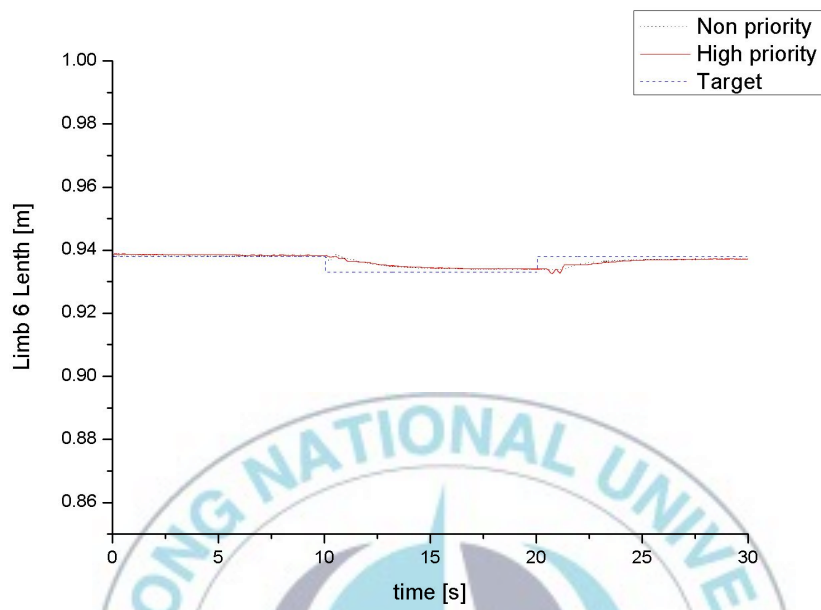


Fig. 22 Length following performance of Limb 6

제 8 장 결론

Z축으로 플랫폼을 일정 높이로 상승시켜, Roll 방향의 회전운동을 실험하였다. 공기압 액추에이터의 특성상 목표치에 빠른 변화에 따른 큰 가속에도 빠른 추종을 보였다.

그러나, 제어초기에 약간의 진동이 발생하고 있고, 6개의 실린더에서 동일한 제어성능을 얻지 못하는 경우가 발생하였다. 그러므로 본 논문에서 적용한 제어기법기법과 공기압 액추에이터의 비선형성을 보상할 수 있는 제어기를 적용한다면 6자유도 시뮬레이터의 제어성능이 더욱 향상될 것으로 판단된다.



참고문헌

- 1) 구상화, 손권, “스튜어트 플랫폼의 기구학적 교정기법에 관한연구,” 제어 · 자동화 · 시스템공학 논문지, Vol. 7, No. 2, pp.168~172, 2001.
- 2) 정지현, 지상원, “공기압실린더 구동장치를 이용한 6자유도 시뮬레이터의 자세제어에 관한 연구” 한국동력기계공학회지, 제11권, 제3호, pp.59~65, 2007.
- 3) Tukamoto, N., Kawakami, Y. and Nakano, K., “An Application of Gain-Scheduling Control to a Pneumatic Servo System”, *Transactions of the Japan Fluid Power System Society*, Vol. 33, No. 1, pp.15~20, 2002.
- 4) Do, W. Q. D. and Yang, D. C. H., “Inverse Dynamic Analysis and Simulation of a Platform Type of Robot”, *J. Robot. Syst.*, Vol.5, No.3, pp.209~227, 1988.
- 5) Hanafusa, H., “Design of Electrohydraulic Servo Control Systems for Articulated Robot Control”, *The Japan Hydraulics & Pneumatics Society*, Vol. 20, No.7, pp.429~436, 1982.
- 6) Dasgupta, B. and Mruthyunjaya, T. S., “A Canonical Formulation of the Direct Position Kinematics Problem for a General 6-6 Stewart Platform”, *Mechanism and Machine Theory*, Vol. 29, No.6, pp.819~827, 1994.
- 7) Hermann Kopetz, Kluwer, “Real-Time systems (Design Principles for Distributed Embedded Applications)”, Academic Publishers, 1997.
- 8) 한재현, 신민석, “실시간 제어시스템의 확률적 응답시간 해석에 관한 연구”, *Transactions of the KSAE*, Vol.16, No.2, pp.8~17, 2006.