

공학석사 학위논문

# 이족보행 로봇을 위한 3D 시뮬레이터

3D Simulator for a Biped Walking Robot

지도교수 최 형 식

2004년 2월

한국해양대학교 대학원

기 계 공 학 과

우 정 재

本 論文을 禹貞在의 工學碩士 學位論文으로 認准함.

委員長 工學博士 鄭 在 鉉 (印)

委 員 工學博士 崔 炯 植 (印)

委 員 工學博士 金 煥 成 (印)

2004 年 2 月

韓國海洋大學校 大學院

機 械 工 學 科

禹 貞 在

# 목 차

Abstract

기호설명

그림목차

표목차

제 1 장. 서론 .....	1
제 2 장. 이족보행 로봇의 동역학 해석 .....	4
2.1. 하체의 순기구학 해석 .....	5
2.2. 하체의 역기구학 해석 .....	7
2.3. 관절 구동부의 동역학 해석 .....	11
2.4. 관절 구동부의 부하 토크 계산 .....	16
제 3 장. 3D 시뮬레이터의 구성 .....	21
3.1. 시뮬레이터의 설계 .....	22
3.2. 3D 이족보행 로봇의 모델링 .....	25
3.3. 물리 엔진의 소개 .....	28
3.4. 제어부의 구성 .....	29
제 4 장. 보행 시뮬레이션 .....	31
4.1. 보행의 안정도 분석 .....	31

4.2. 보행 주기의 결정 .....	34
4.3. 보행을 위한 각 관절의 궤적 생성 .....	37
4.4. 궤적을 이용한 로봇의 구동 .....	48
제 5 장. 결론 .....	49
참고문헌 .....	50

# 3D Simulator of a Biped Walking Robot

Jeong – Jae Woo

Department of Mechanical Engineering  
Graduate School, Korea Maritime University

## Abstract

In this paper, a 3D simulator was developed to reduce experimental problems of dangerous, economical and time element for locomotion of the biped walking robot (BWR) with joint actuators using the Open graphics library (OpenGL) and physical engine so called Open Dynamics Engine (ODE).

The database of 3D models for robot body's elements using the Milkshape 3D ascii format was set up, and using them, 3D locomotion was simulator with mass, length of robot body's elements to 3D. The 3D simulator was develop to have function of calculation of load torque of the drive motor to show the 3D simulator and to have ZMP function for stable walk of robot. The robot model adapted for the 3D simulator shows gesture of walk cycle of 8 steps, and it's trajectories are generated based on the data of joint angles that from the motion capture system. To track the specified trajectories, the PID control algorithm is applied.

Finally, the joint angle from the simulator was applied to the actual BWR, and which showed similar walking results.

## 기 호 설 명

$A_i$	각 관절의 동차 변환 행렬
$l_i$	각 관절의 링크 길이
$\alpha_i$	각 관절의 비틀림
$d_i$	각 관절의 오프셋
$\theta_i$	각 관절의 각도
$T_0^4$	기구의 동차 변환 행렬
$\bar{p}$	말단 장치의 위치 벡터
$S$	왜대칭 행렬
$\bar{\omega}$	관절의 각속도 벡터
$J_v$	선속도 자코비안 행렬
$J_\omega$	각속도 자코비안 행렬
$d_{bi}$	각 관절 구동부의 볼 나사 변위
$\dot{d}_{bi}$	각 관절 구동부의 볼 나사 선 속도
$\ddot{d}_{bi}$	각 관절 구동부의 볼 나사 선 가속도
$a_{i2}, a_{i3}, a_{i4}$	각 관절부의 고정 링크 길이
$\beta_i$	각 관절부의 회전각
$\phi_i$	$d_{bi}$ 와 $a_{i4}$ 가 만나는 점의 회전각
$K_p$	PID 제어기의 비례항 인자
$K_i$	PID 제어기의 적분항 인자

$K_d$	PID 제어기의 미분항 인자
$e(n)$	이산화 된 오차 신호
$u(n)$	이산화 된 제어 신호
$\tau(n)$	이산화 된 토크 제어 신호
$m_i$	각 부재의 질량
$O_{xyz}$	각 부재의 관성 좌표계
$r_i$	각 부재의 질량 중심
$I_i$	각 부재의 질량 중심에 관한 관성력
$f_k$	시스템의 외부 힘
$M_j$	시스템의 모멘트
$s_k$	시스템에서 작용한 힘의 위치
$P_{zmp}$	ZMP의 위치
$f(x_h, y_h, \theta_h)$	허리 관절의 상대 좌표에 따른 관절각의 궤적 함수
$f(x_f, y_f, \theta_f)$	발목 관절의 상대 좌표에 따른 관절각의 궤적 함수

## 그림 목차

그림 2.1 이족보행 로봇의 실제 형상 .....	4
그림 2.2 이족보행 로봇 하체의 D-H 좌표계 .....	5
그림 2.3 이족보행 로봇 하체의 관절 구동부 .....	11
그림 2.4 각 관절 구동기의 부하 토크 .....	16
그림 2.5 각 관절 구동부의 부하 토크 결과 .....	18
그림 3.1 3D 시뮬레이터의 구조 .....	21
그림 3.2 3D 시뮬레이터의 메인 화면 .....	22
그림 3.3 3D 시뮬레이터의 클래스 구조 .....	24
그림 3.4 MilkShape 3D ASCII 파일 포맷의 사양 .....	27
그림 3.5 ODE 에서 활용되는 조인트 .....	28
그림 4.1 강체 (rigid body) 의 시스템.....	32
그림 4.2 이족보행 로봇의 보행 주기 .....	35
그림 4.3 보행을 위한 허리와 발목 관절의 궤적함수 생성 .....	37
그림 4.4 각 관절각의 궤적 .....	39
그림 4.5 동작획득 장치를 이용한 관절각의 궤적 .....	40
그림 4.6 시뮬레이터를 이용한 이족보행 로봇의 보행 동작 .....	46
그림 4.7 ZMP 의 궤적 .....	47

## 표 목 차

표 2.1 이족 보행 로봇 하체의 링크 매개 변수 .....	6
표 2.2 모터 사양의 비교 .....	20
표 4.1 이족보행 로봇 모델 변수 .....	34
표 4.2 궤적을 이용한 로봇의 구동 .....	41

# 제 1 장 서 론

인간을 대신하여 위험적인 요소를 해결하고 경제적인 요소를 절감하기 위하여 개발된 인간과 유사한 구조를 가지는 이족보행 로봇은 1960년대 후반 Vukobratovic [1, 2] 이 모델링을 처음 제시한 이후 많은 이론적 [3-9] 혹은 실험적 [10-15] 연구들이 수행되어져 왔다. 그 동안 연구에만 그쳐왔던 이족보행 로봇이 자동화나 서비스 산업의 가능성을 입증하게 된 계기는 1998년 일본의 혼다 사에서 발표한 P3 [16] 이다. P3는 인체의 크기와 거의 흡사하며 실제 환경에서의 보행을 가능하게 하여 이족보행 로봇에 대한 연구를 더욱 발전시키게 되었다.

그러나 아직까지도 인간과 같은 보행 동작을 구현하기에는 위험적, 경제적 또는 시간적인 어려움이 따르며 이를 해결하기 위해 이족보행 로봇의 관절부에 구동기를 장착하여 보행 동작을 분석하는 하드웨어적인 방법보다 PC 기반으로 한 3D 이족보행 로봇 시뮬레이터를 개발하고 이를 이용하는 방법들이 많이 제시되고 있다.

대표적인 연구로서 1984년 일본의 와세다 대학에서 보행 패턴의 분석과 Zero Moment Point (ZMP) [1] 개념을 도입하여 안정된 보행을 시뮬레이션하기 위한 Walk Master-2 [17] , 1999년 미국의 MIT 대학에서 7개의 링크와 12 자유도로 이루어진 이족보행 로봇이 평지에서의 동적인 보행 동작 [18] 을 분석하기 위한 시뮬레이터 [19] 그리고 2003년 스위스의 Biologically Inspired Robotics 그룹에서 소니 사의 SDR-4X 로봇을 모델로 기반으로 실제 환경과 유사하게 처리하기 위해 물리엔진을 이용하여 개발한 시뮬레이터 [20] 등이 있다.

또한 실제 환경을 PC 기반으로 한 가상의 공간을 구성하는 것 또한 많은 시간을 투자해야 하는 어려움이 따르므로 소형으로 이족보행 로봇을 제작하여 보행 패턴이나 안정성을 실험하는 연구도 많이 병행이 되고 있다. 1998 년 호주의 Western 대학에서 개발하여 불규칙적인 지형을 보행하기 위한 실험 [21] 과 동적인 보행을 연구하기 위한 실험 [22] , 2000 년 와세다 대학에서 개발된 시뮬레이터 Walk Master-2 를 이용하여 시뮬레이션 데이터를 적용한 WABIAN-R II [23] 등이 있다.

그러나 소형의 이족보행 로봇은 서비스적인 기능들은 갖추고 있으나 인체형 이족보행 로봇이 가지는 장점인 인간이 생활하는 작업영역의 적합성과 보행을 위한 지형적인 요소를 분석하기는 힘들다. 그래서 이를 위한 제어연구 [24-26] 들이 활발이 이루어지고 있으며 제어 성능을 가상으로 실험할 수 있는 시뮬레이터를 개발해 실제 로봇에 적용할 수 있게 연구를 하게 되었다.

본 논문은 강제 동역학을 시뮬레이션 하기 위해 물리 엔진인 Open Dynamics Engine (ODE) [27] 을 이용하여 실제환경과 비슷한 물리 법칙이 적용되는 가상의 3차원 공간을 구성하여 개발되어진 이족보행 로봇의 모델 [28] 을 바탕으로 하여 ZMP 를 이용한 안정적인 보행 패턴의 생성과 구동 관절기의 에너지 효율을 최소화하기 위한 부하 토크 분석, 보다 나은 보행 동작을 위한 다양한 제어 알고리즘의 적용, 실제 로봇에 적용하기 위한 관절의 궤적을 생성하기 위한 3D 시뮬레이터를 개발하고 이를 실제 이족보행 로봇에 적용하고자 한다.

본 논문의 구성은 제 2 장에서 3D 시뮬레이터에 적용하기 위한 가상의 이족보행 로봇 하체의 기구적 해석과 관절 구동부의 역학적 해석, 관절 구동기 선정의 타당성을 검증하기 위한 부하 토크 분석을 소개하고 제 3 장에서 개발되어진 3D 시뮬레이터의

동작 환경과 가상 이족보행 로봇 모델의 구성, 실제 환경과 비슷한 물리력을 적용하기 위한 물리엔진을 소개하며 제 4 장에서 보행 시뮬레이션을 위한 제어 알고리즘과 적용된 보행 안정성을 판별하기 위한 ZMP 분석, 보행의 주기와 궤적의 생성, 궤적을 이용한 보행을 로봇에 적용하였다.

## 제 2 장 이족보행 로봇의 동역학 해석

3D 시뮬레이터에 적용하기 위한 이족보행 로봇의 모델은 그림 2.1 과 같이 총 15 자유도의 기구학적 구조로 구성된 로봇 [28] 을 이용하였다. 하체에 적용된 관절 구동부인 사절 링크 구조의 동역학 해석을 하고 보행을 위하여 발 바닥에 위치한 말단 좌표를 순기구학 [29] 을 통해 구하고 이를 자코비안 (Jacobian) 행렬을 이용한 역기구학 해석 [30] 을 이용하여 관절 각을 유도한다. 또한 관절 구동기 선택의 타당성을 확인하기 위하여 부하 토크 (torque) 시뮬레이션을 행한다.

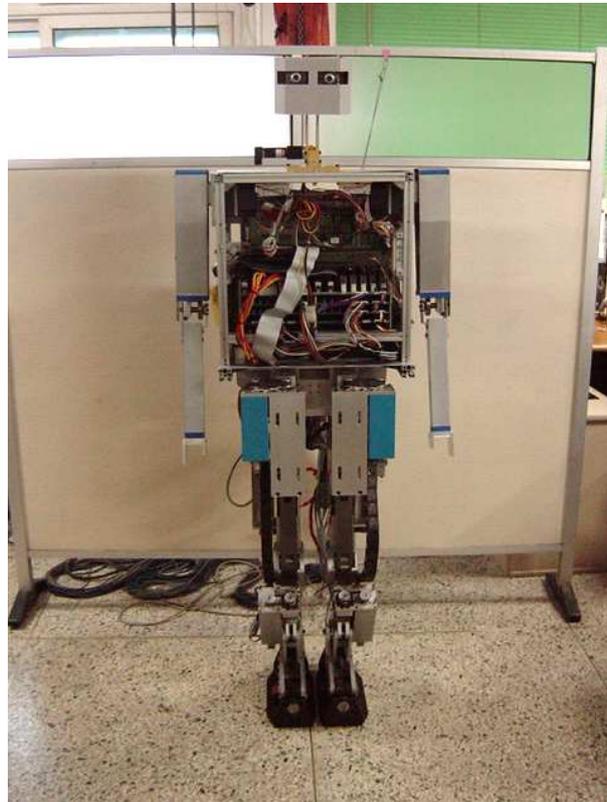


그림 2.1 이족보행 로봇의 실제 형상

## 2.1. 하체의 순기구학 해석

로봇의 관절 변수가 주어졌을 때, 말단 장치인 발바닥의 위치와 방향을 결정하기 위한 이족보행 로봇 하체의 순기구학 방적식은 그림 2.2 과 같이 Denavit-Hartenberg (D-H) 규약에 의해 정의된 좌표계를 이용하여 유도한다.

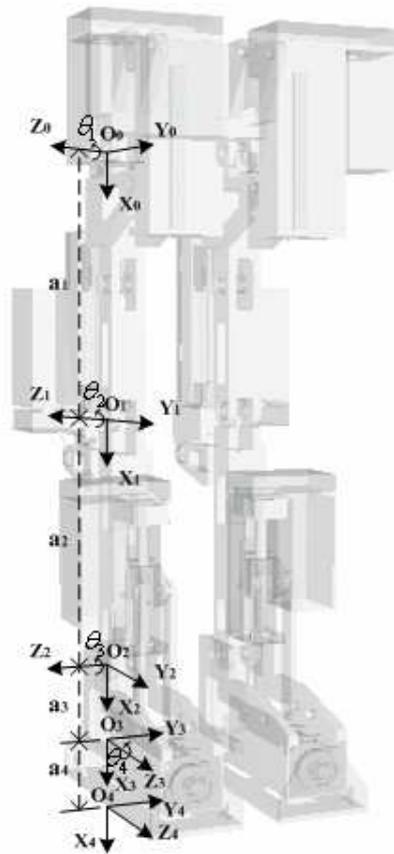


그림 2.2 이족보행 로봇 하체의 D-H 좌표계

D-H 규약은 4개의 기본 변환 행렬의 곱으로 각 관절의 동차 변환 행렬  $A_i$  로 표현하는 것이다. 즉,

$$A_i = Rot_{z,\theta_i} Trans_{z,d_i} Trans_{x,l_i} Rot_{x,\alpha_i}$$

$$= \begin{bmatrix} \cos \theta_i & -\sin \theta_i & 0 & 0 \\ \sin \theta_i & \cos \theta_i & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & l_i \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos \alpha_i & -\sin \alpha_i & 0 \\ 0 & \sin \alpha_i & \cos \alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1.1)$$

$$= \begin{bmatrix} \cos \theta_i & -\sin \theta_i \cos \alpha_i & \sin \theta_i \sin \alpha_i & l_i \cos \theta_i \\ \sin \theta_i & \cos \theta_i \cos \alpha_i & -\cos \theta_i \sin \alpha_i & l_i \sin \theta_i \\ 0 & \sin \alpha_i & \cos \alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

여기서  $l_i$  는 각 관절의 링크 길이 (length),  $\alpha_i$  는 비틀림 (twist),  $d_i$  는 오프셋 (offset),  $\theta_i$  는 각도 (angle) 이다. 그림 2.2 에 의해 설정된 매개 변수는 표 2.1 과 같다.

표 2.1 이족보행 로봇 하체의 링크 매개 변수

링크	$l_i$	$\alpha_i$	$d_i$	$\theta_i$
1	$l_1$	0	0	$\theta_1$
2	$l_2$	0	0	$\theta_2$
3	$l_3$	$-90^\circ$	0	$\theta_3$
4	$l_4$	0	0	$\theta_4$

A 행렬들은 식 (2.1.1) 로 부터 다음과 같이 얻어 진다.

$$A_1 = \begin{bmatrix} \cos \theta_1 & -\sin \theta_1 & 0 & l_1 \cos \theta_1 \\ \sin \theta_1 & \cos \theta_1 & 0 & l_1 \sin \theta_1 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_2 = \begin{bmatrix} \cos \theta_2 & -\sin \theta_2 & 0 & l_2 \cos \theta_2 \\ \sin \theta_2 & \cos \theta_2 & 0 & l_2 \sin \theta_2 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.1.2)$$

$$A_3 = \begin{bmatrix} \cos \theta_3 & 0 & -\sin \theta_3 & l_3 \cos \theta_3 \\ \sin \theta_3 & 0 & \cos \theta_3 & l_3 \sin \theta_3 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad A_4 = \begin{bmatrix} \cos \theta_4 & -\sin \theta_4 & 0 & l_4 \cos \theta_4 \\ \sin \theta_4 & \cos \theta_4 & 0 & l_4 \sin \theta_4 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

따라서 변환 행렬  $T$  행렬은 식 (2.1.2) 의 행렬들에 의해 다음과 같이 주어진다.

$$T_0^4 = A_1 A_2 A_3 A_4 \tag{2.1.3}$$

$$= \begin{bmatrix} \cos(\theta_1 + \theta_2 + \theta_3) \cos \theta_4 & -\cos(\theta_1 + \theta_2 + \theta_3) \sin \theta_4 & -\sin(\theta_1 + \theta_2 + \theta_3) & P_x \\ \sin(\theta_1 + \theta_2 + \theta_3) \cos \theta_4 & -\sin(\theta_1 + \theta_2 + \theta_3) \sin \theta_4 & \cos(\theta_1 + \theta_2 + \theta_3) & P_y \\ -\sin \theta_4 & -\cos \theta_4 & 0 & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

식 (2.1.3) 에서의 발바닥의 위치 벡터  $p$  는 다음과 같다.

$$\begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} = \begin{bmatrix} l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2 + \theta_3) + l_4 \cos(\theta_1 + \theta_2 + \theta_3) \cos \theta_4 \\ l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3) + l_4 \sin(\theta_1 + \theta_2 + \theta_3) \cos \theta_4 \\ -l_4 \sin \theta_4 \end{bmatrix} \tag{2.1.4}$$

## 2.2. 하체의 역기구학 해석

앞에서 개발된 이족보행 로봇 하체의 발바닥의 위치와 방향을 관절 변수로 나타내는 방법을 설명하였다. 이번에는 말단 장치의 위치와 방향에서 관절 변수를 역으로 찾아내기 위하여 보행 시 각 관절의 부드러운 궤적을 생성하기 위한 자코비안 행렬을 유도하고 관절각의 역속도 방정식을 유도한다.

식 (2.1.3) 에서 정의된 이족보행 로봇의 기구학 해석을 다시 정의 하면 다음과 같다.

$$T_0^4(\theta) = \begin{bmatrix} R_0^4(\theta) & d_0^4(\theta) \\ 0 & 1 \end{bmatrix} \quad (2.2.1)$$

여기서,  $\theta = [\theta_1 \ \theta_2 \ \theta_3 \ \theta_4]^T$  는 관절 변수의 벡터이다. 말단 장치의 좌표계로부터 기저 좌표계로의 변환을 나타내면 이족보행 로봇이 움직임에 따라 관절 변수  $\theta_i$  와 말단 장치 위치 벡터  $d_0^4$  와 방향  $R_0^4$  는 시간의 함수가 된다. 여러 개의 링크로 연결되어 있는 이족보행 로봇의 보행 시 각각의 링크는 각기 다른 선속도와 각속도를 가진다. 따라서 말단 장치의 선속도와 각속도를 기저 좌표계에 대한 관절 속도의 벡터  $\dot{\theta}(t)$  와 연관 지어야 한다.

$$S(\omega_0^4) = \dot{R}_0^4 (\dot{R}_0^4)^T \quad (2.2.2)$$

식 (2.2.2) 에서  $S(\omega)$  는 왜대칭 행렬이며 이 식이 각속도 벡터  $\omega_0^4$  를 정의한다.

$$v_0^4 = \dot{d}_0^4 \quad (2.2.3)$$

식 (2.2.3) 이 말단 장치의 선속도를 나타낸다. 이때 식 (2.2.2) 과 (2.2.3) 에 의해 다음과 같은 관계를 구할 수 있다.

$$v_0^4 = J_v \dot{\theta} \quad (2.2.4)$$

$$\omega_0^4 = J_\omega \dot{\theta} \quad (2.2.5)$$

식 (2.2.4) 과 (2.2.5) 에서  $3 \times 4$  행렬인  $J_v, J_\omega$  는 선속도 자코비안 행렬과 각속도 자코비안 행렬이다. 또한 두 식을 같이 쓰면 다음과 같이 된다.

$$J_0^4 = \begin{bmatrix} J_v \\ J_\omega \end{bmatrix} \quad (2.2.6)$$

식 (2.2.6) 에서  $6 \times 4$  행렬  $J_0^4$  는 이족 보행 로봇의 자코비안 행렬이며 이를 벡터의 성분으로 나타내면 다음과 같다.

$$J_1 = \begin{bmatrix} -(l_1 \sin \theta_1 + l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3) + l_4 \sin(\theta_1 + \theta_2 + \theta_3) \cos \theta_4) \\ l_1 \cos \theta_1 + l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2 + \theta_3) + l_4 \cos(\theta_1 + \theta_2 + \theta_3) \cos \theta_4 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (2.2.7)$$

$$J_2 = \begin{bmatrix} -(l_2 \sin(\theta_1 + \theta_2) + l_3 \sin(\theta_1 + \theta_2 + \theta_3) + l_4 \sin(\theta_1 + \theta_2 + \theta_3) \cos \theta_4) \\ l_2 \cos(\theta_1 + \theta_2) + l_3 \cos(\theta_1 + \theta_2 + \theta_3) + l_4 \cos(\theta_1 + \theta_2 + \theta_3) \cos \theta_4 \\ 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \quad (2.2.8)$$

$$J_3 = \begin{bmatrix} -l_4 \sin \theta_4 \cos(\theta_1 + \theta_2 + \theta_3) \\ -l_4 \sin \theta_4 \sin(\theta_1 + \theta_2 + \theta_3) \\ \left( -\sin(\theta_1 + \theta_2 + \theta_3)(l_3 \sin(\theta_1 + \theta_2 + \theta_3) + l_4 \sin(\theta_1 + \theta_2 + \theta_3) \cos \theta_4) \right) \\ \left( -\cos(\theta_1 + \theta_2 + \theta_3)(l_3 \cos(\theta_1 + \theta_2 + \theta_3) + l_4 \cos(\theta_1 + \theta_2 + \theta_3) \cos \theta_4) \right) \\ -\sin(\theta_1 + \theta_2 + \theta_3) \\ \cos(\theta_1 + \theta_2 + \theta_3) \\ 0 \end{bmatrix} \quad (2.2.9)$$

$$J_4 = \begin{bmatrix} -l_4 \sin \theta_4 \cos(\theta_1 + \theta_2 + \theta_3) \\ -l_4 \sin \theta_4 \sin(\theta_1 + \theta_2 + \theta_3) \\ \left( -\sin(\theta_1 + \theta_2 + \theta_3)l_4 \sin(\theta_1 + \theta_2 + \theta_3) \cos \theta_4 \right) \\ \left( -\cos(\theta_1 + \theta_2 + \theta_3)l_4 \cos(\theta_1 + \theta_2 + \theta_3) \cos \theta_4 \right) \\ -\sin(\theta_1 + \theta_2 + \theta_3) \\ \cos(\theta_1 + \theta_2 + \theta_3) \\ 0 \end{bmatrix} \quad (2.2.10)$$

식 (2.2.4) 와 (2.2.5) 에서는 각 관절의 선속도를 자코비안 행렬로 정의를 했다. 이를 역으로 정의를 하면 이족보행 로봇의 직교좌표에서 각 관절의 각속도로 변환하는 방정식을 얻을 수 있으며 다음과 같다.

$$\dot{\theta} = J_v^{-1} v_0^4 \quad (2.2.11)$$

$$\dot{\theta} = J_\omega^{-1} \omega_0^4 \quad (2.2.12)$$

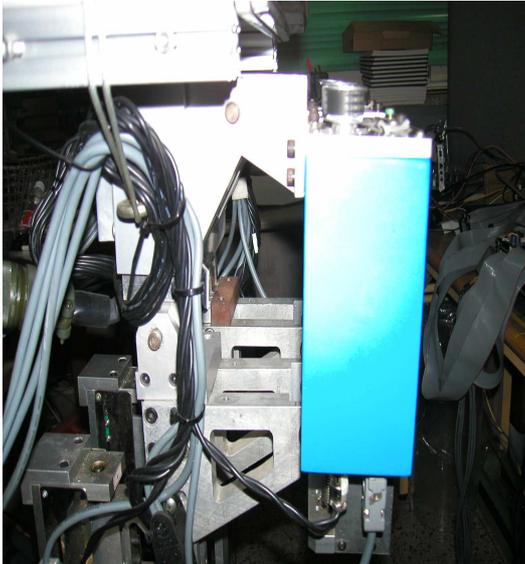
식 (2.2.11) 과 (2.2.12) 를 만족하기 위해서는  $\det J_0^4 \neq 0$  이어야 한다. 또한 자코비안

행렬의 역을 구하기 위한 필요충분조건은  $\dot{\theta} \in R^4$  일 때 첨가 행렬  $[J_0^4 | \dot{\theta}]$  의 계

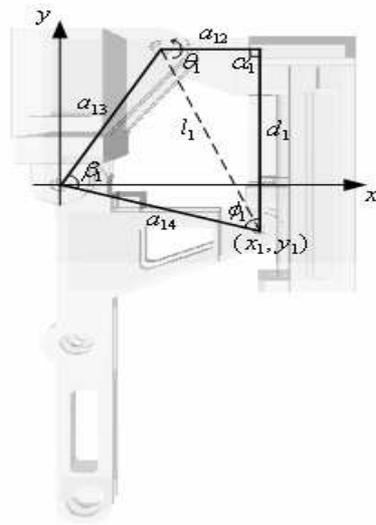
수가 자코비안 행렬  $J_0^4$ 의 계수가 같아질 경우이다.

### 2.3. 관절 구동부의 동역학 해석

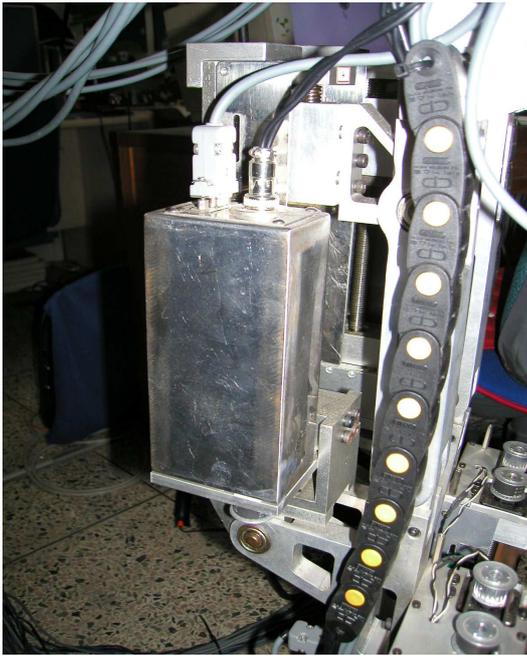
그림 2.3 과 같이 각 관절의 기구적 구조는 볼 나사 구동기의 변화율에 따른 볼 나사의 변위  $d_{bi}$  와 고정 링크  $a_{i2}, a_{i3}, a_{i4}$  로 구성된다.



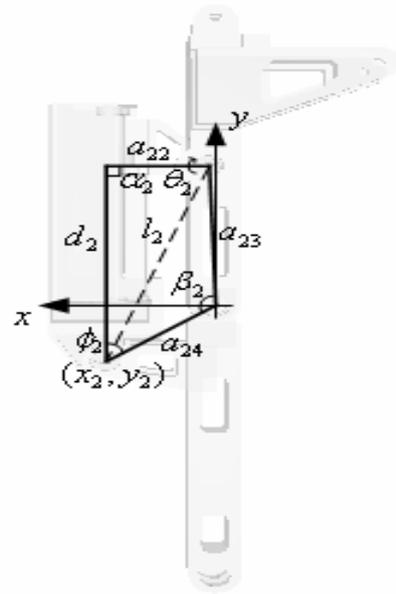
(a) 허리 관절의 실제 형상



(b) 허리 관절의 사절 링크 모델링



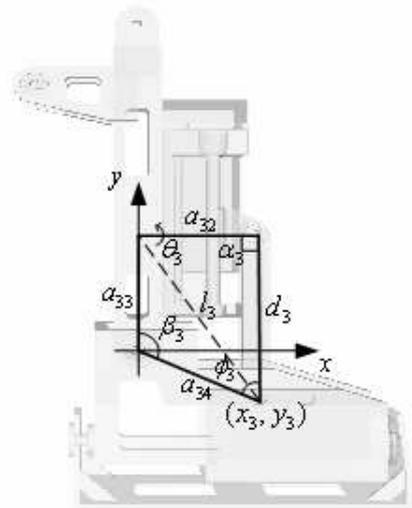
(c) 무릎 관절의 실제 형상



(d) 무릎 관절의 사절 링크 모델링



(e) 발목 관절의 실제 형상



(f) 발목 관절의 사절 링크 모델링

그림 2.3 이족보행 로봇 하체의 관절 구동부

그림 2.3 의 사절 링크에서  $l_i (i=1,2,3)$  에 대한 코사인법칙을 적용하면 다음과 같다.

$$l_i = a_{i2}^2 + d_{bi}^2 - 2a_{i2}d_{bi} \cos \alpha_i = a_{i3}^2 + a_{i4}^2 - 2a_{i3}a_{i4} \cos \beta_i$$

$$d_{bi} = \frac{C_i + [C_i^2 + 4(A_i + B_i \cos \beta_i)]^{\frac{1}{2}}}{2} \quad (2.3.1)$$

$i=1$  이면 허리 관절,  $i=2$  이면 무릎 관절,  $i=3$  이면 발목 관절이며 식 (2.3.1) 에서

$$A_i = a_{i3}^2 + a_{i4}^2 - a_{i2}^2$$

$$B_i = -2a_{i3}a_{i4}$$

$$C_i = 2a_{i2} \cos \alpha_i$$

이다. 그리고 링크 길이  $a_{i2}, a_{i3}, a_{i4}$  와 링크 각  $\alpha_i$  는 고정된 값이다. 식 (2.3.1) 의 미끄럼 변위를 시간에 관해 미분하면 다음과 같이 속도 및 가속도를 구할 수 있다.

$$\dot{d}_{bi} = -[C_i^2 + 4(A_i + B_i \cos \beta_i)]^{-\frac{1}{2}} B_i \sin \beta_i \dot{\beta}_i \quad (2.3.2)$$

$$\ddot{d}_{bi} = -2[C_i^2 + 4(A_i + B_i \cos \beta_i)]^{-\frac{3}{2}} B_i^2 \sin^2 \beta_i \dot{\beta}_i^2$$

$$- [C_i^2 + 4(A_i + B_i \cos \beta_i)]^{-\frac{1}{2}} (B_i \cos \beta_i \ddot{\beta}_i + B_i \sin \beta_i \dot{\beta}_i^2) \quad (2.3.3)$$

회전각  $\beta_i$  와  $d_{bi}$  의 속도 및 가속도 관계식을 구하면 다음과 같다.

$$\beta = \arccos \left[ \frac{d_{bi}^2 - A_i - C_i d_{bi}}{B_i} \right] \quad (2.3.4)$$

$$\dot{\beta}_i = R_{i1} \dot{d}_{bi} \quad (2.3.5)$$

$$\ddot{\beta}_i = R_{i2} \dot{d}_{bi}^2 + R_{i3} \ddot{d}_{bi} \quad (2.3.6)$$

식 (2.3.5) 와 (2.3.6) 에서

$$R_{i1} = \frac{[C_i^2 + 4(A_i + B_i \cos \beta_i)]^{\frac{1}{2}}}{B_i \sin \beta_i}$$

$$R_{i2} = -2[C_i^2 + 4(A_i + B_i \cos \beta_i)]^{-1} B_i \sin \beta_i R_{i1}^2 + \frac{\cos \beta_i}{\sin \beta_i} R_{i1}^2$$

$$R_{i3} = -\frac{[C_i^2 + 4(A_i + B_i \cos \beta_i)]^{\frac{1}{2}}}{B_i \sin \beta_i}$$

이다. 그림 2.3 에서  $d_{bi}$  와  $a_{i4}$  가 만나는 점에서 좌표  $(x_i, y_i)$  를 표현하여  $\phi_i$  와  $d_i$  의 관계식을 유도하면 다음과 같다.

$$x_i = d_{bi} \cos(\phi_i + N_i) + a_{i2} \cos \theta_i = a_{i4} \cos N_i - a_{i3} \cos q_i \quad (2.3.7)$$

$$y_i = d_{bi} \sin(\phi_i + N_i) - a_{i2} \sin \theta_i = a_{i4} \sin N_i + a_{i3} \sin q_i \quad (2.3.8)$$

식 (2.3.7) 과 (2.3.8) 에서

$$\theta_i = \pi - \phi_i - (\alpha_i + N_i)$$

이다. 식 (2.3.7) 의 좌변에  $\theta_i$  를 대입하여  $\phi_i$  에 대해 정리를 하면 다음과 같다.

$$\begin{aligned} & d_{bi}(\cos \phi_i \cos N_i - \sin \phi_i \sin N_i) + a_{i2}[\cos(\pi - \phi_i) \cos(\alpha_i + N_i) + \sin(\pi - \phi_i) \sin(\alpha_i + N_i)] \\ &= [d_{bi} \cos N_i - a_{i2} \cos(\alpha_i + N_i)] \cos \phi_i - [d_{bi} \sin N_i - a_{i2} \sin(\alpha_i + N_i)] \sin \phi_i \end{aligned} \quad (2.3.9)$$

$$\begin{aligned} X_i &= d_{bi} \cos N_i - a_{i2} \cos(\alpha_i + N_i) \\ Y_i &= d_{bi} \sin N_i - a_{i2} \sin(\alpha_i + N_i) \end{aligned} \quad (2.3.10)$$

식 (2.3.9) 에서 식 (2.3.10) 과 같이 치환하고 재정의하면 다음과 같다.

$$X_i \cos \phi_i - Y_i \sin \phi_i = x_i \quad (2.3.11)$$

또한, 식 (2.3.8) 의 좌변에  $\theta_i$  를 대입하여  $\phi_i$  에 대해 정리를 하면 다음과 같다.

$$\begin{aligned} & d_{bi}(\sin \phi_i \cos N_i - \cos \phi_i \sin N_i) - a_{i2}[\sin(\pi - \phi_i) \cos(\alpha_i + N_i) - \cos(\pi - \phi_i) \sin(\alpha_i + N_i)] \\ &= [d_{bi} \cos N_i - a_{i2} \cos(\alpha_i + N_i)] \sin \phi_i - [d_{bi} \sin N_i - a_{i2} \sin(\alpha_i + N_i)] \cos \phi_i \end{aligned} \quad (2.3.12)$$

마찬가지로 식 (2.3.12) 를 재정의하면 다음과 같다.

$$Y_i \cos \phi_i - X_i \sin \phi_i = y_i \quad (2.3.13)$$

식 (2.3.11) 과 식 (2.3.13) 을 이용하여  $\phi_i$  를 구하면 다음과 같다.

$$\phi_i = a \cos\left(\frac{X_i x_i + Y_i y_i}{X_i^2 + Y_i^2}\right) \quad (2.3.14)$$

식 (2.3,14) 에 구한  $\phi_i$  는 다음의 2.4 장에서 다룰 각 관절 구동기의 토크 계산에 사용된다.

## 2.4. 관절 구동부의 부하 토크 계산

앞선 2.3 장에서 각 관절 구동기의 사절 링크에 대한 동역학 해석을 해보았다. 사절 링크 모델링을 설명하는 그림 2.3 을 각 관절 구동기의 부하토크 방정식을 유도하기 위해 도식화 하면 그림 2.4 와 같다. 이는 로봇의 보행 시 각 관절부에 작용하는 부하 토크를 시뮬레이션 함으로써 선택된 관절 구동기의 타당성을 검증하려는 것이다.

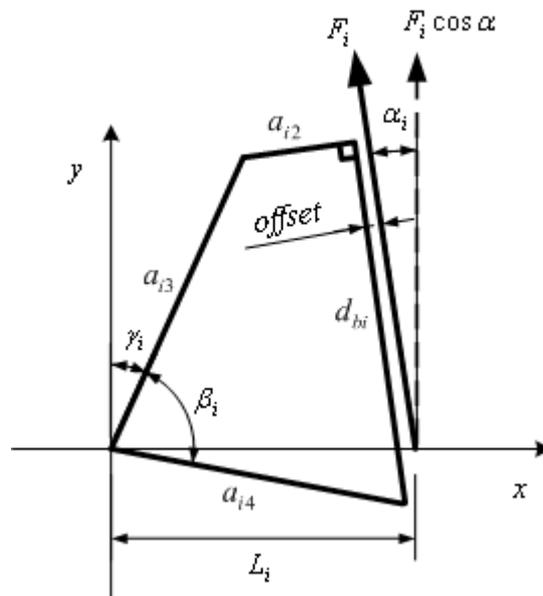


그림 2.4 각 관절 구동기의 부하 토크

식 (2.3.1) 에서 보듯이  $\beta_i$  가 변하게 되면 볼 나사의 길이가 바뀌고 그림 2.4 와 같이  $\alpha_i$  가 변하게 되므로 볼 나사 축 방향의 부하 외력  $F_i$  의 방향이 바뀐다. 회전 중심에 대한 모멘트 합이 원리에 의해 외력에 대한 각 관절부의 관계식을 구하면 다음과 같다.

$$F_1 = \frac{F_{\text{sup port}} \cdot L_{\text{thigh}} \cdot \sin \gamma_1}{L_1 \cdot \cos \alpha_1} \quad (2.4.1)$$

$$F_2 = \frac{F_{\text{sup port}} \cdot L_{\text{thigh}} \cdot \sin \alpha_2}{L_2 \cdot \sin \gamma_2} \quad (2.4.2)$$

$$F_3 = \frac{F_{\text{sup port}} \cdot L_{\text{fibula}} \cdot \sin \alpha_3}{L_3} \quad (2.4.3)$$

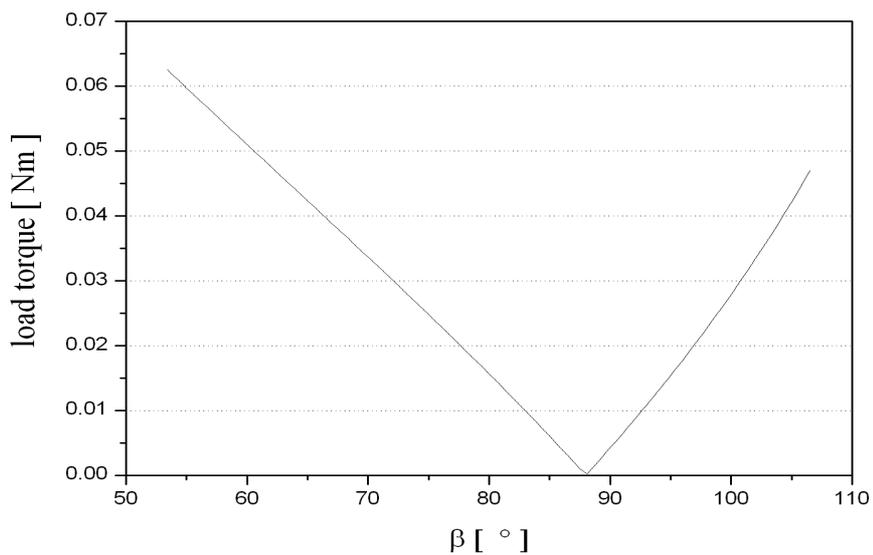
식 (2.4.1) 은 허리 관절부, (2.4.2) 는 무릎 관절부, (2.4.3) 은 발목 관절부에 해당이 되며 식에서  $F_{\text{sup port}}$  는 한 쪽 다리가 지면에 지지할 때의 자중으로 허리 관절에서는 70kgf, 무릎 관절에서는 75kgf, 발목 관절에서는 78kgf 이다. 그리고  $L_i$  는 볼 나사가 위치한 *offset* 과 고정 링크  $a_{i4}$  를  $x$  축에 사상 시킨 길이이며  $L_{\text{thigh}}, L_{\text{fibula}}$  는 로봇의 허벅지와 종아리의 길이이다.  $\gamma$  는 상대 좌표계의  $y$  축과 고정 링크  $a_{i3}$  의 각이며  $\alpha_i$  는 볼 나사  $d_{bi}$  를  $y$  축에 사상 시키기 위한 각이다.

각 관절부의 볼 나사에 작용하는 축 방향의 부하 하중에 대한 토크를 계산하는 방정식은 다음과 같다.

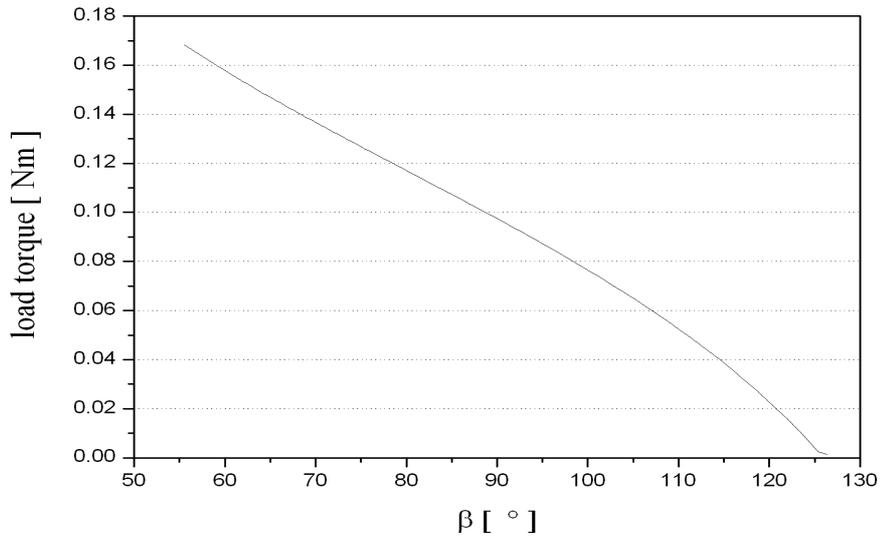
$$\tau_{load,i} = \frac{F_i \cdot l_b}{2\pi \cdot \eta} \cdot A \quad (2.4.4)$$

여기서,  $l_b$ 는 볼 나사의 리드 (lead) 길이로 2mm 이고  $\eta$ 는 볼 나사의 효율로 90% 로 가정하며  $A$ 는 감속비율로  $\frac{1}{5.8}$ 을 가진다. 이는 0.077Nm의 정격 토크와 90W의 용량을 가지는 감속기가 장착된 소형 DC 모터를 기준으로 하였다.

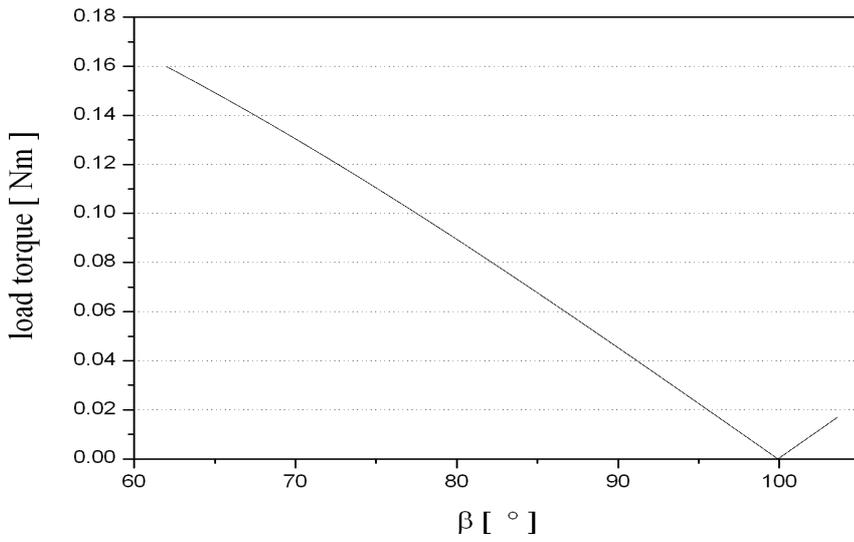
식 (2.4.1), (2.4.2), (2.4.3) 의 각 관절부를 (2.4.4) 식을 이용하여 시뮬레이션 한 결과는 그림 2.5 와 같다.



(a) 허리 관절부의 부하 토크



(b) 무릎 관절부의 부하 토크



(c) 발목 관절부의 부하 토크

그림 2.5 각 관절부의 부하 토크 결과

그림 2.5 의 결과는 각 관절부의  $\beta_i$ 의 변화율에 따른 부하 토크로 최대 0.17Nm의 정격 토크가 필요함을 알 수 있다. 현재 사용되는 모터보다 사양을 넘기 때문에 구동 실험 시 과전류에 의한 모터의 문제가 발생할 가능성이 있다. 따라서 표 2.2 에 보는 바와 같이 사용 중인 모터와 동일한 회사의 제품인 0.181Nm의 정격 토크와 150W의 용량을 가지는 모터로 교환해야 안정적인 구동기를 설계할 수 있다.

표 2.2 모터 사양의 비교

구분	구동 전원[V]	정격 토크[Nm]	최대 전류[A]	용량[W]
사용 중인 모터	15V	0.077	4	90
교체할 모터	24V	0.181	6	150

### 제 3 장. 3D 시뮬레이터의 구성

이족보행 로봇을 위한 3D 시뮬레이터는 그림 3.1 과 같은 구조를 가지며 윈도우즈 환경에서 동작하고 Microsoft Visual C++ 6.0 컴파일러와 3D 그래픽을 구현하기 위해서 Silicon Graphics 사의 Open Graphics Library (OpenGL) [31] 를 이용한다. 이족보행 로봇의 3D 모델은 OpenGL 로 파싱 (parsing) 하기 용이한 MilkShape 3D ASCII 파일 포맷으로 이루어진 데이터 베이스를 가진다. 그리고 실물에 가깝게 하기 위해서 Jpeg 그림 파일을 이용한 텍스처 맵핑 (texture mapping) 으로 이족보행 로봇의 3D 모델과 배경 화면을 렌더링 (rendering) 한다.

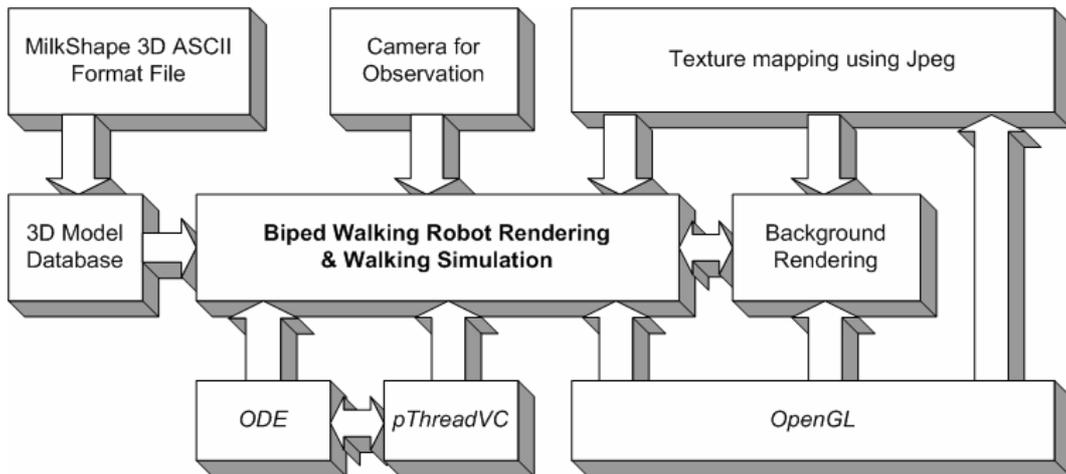


그림 3.1 3D 시뮬레이터의 구조

### 3.1. 시뮬레이터의 설계

이족보행 로봇을 위한 3D 시뮬레이터 화면의 구성은 그림 3.2 와 같이 7 가지의 기능으로 분류할 수가 있다.

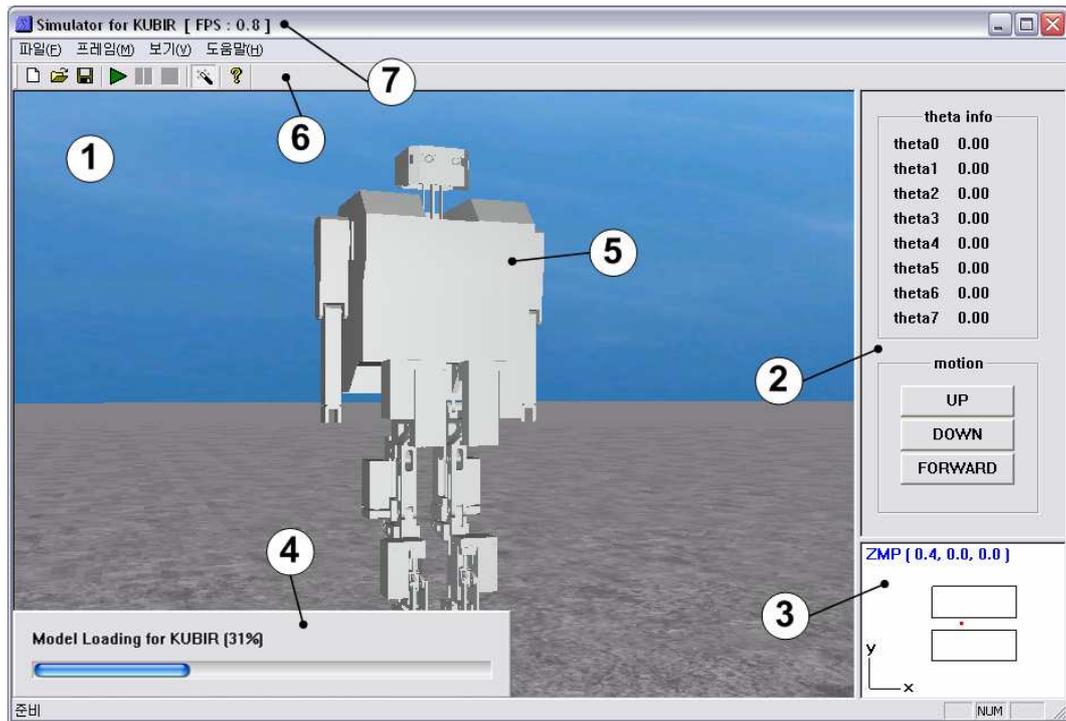


그림 3.2 3D 시뮬레이터의 메인 화면

기능들을 소개하면 다음과 같다.

① 배경 화면 : 가상의 환경으로 이루어진 배경 화면은 로드 (load) 되어진 ⑤ 의 3D 로봇 모델에 물리력이 적용된다. 로봇의 보행 알고리즘을 적용할 경우 보행 동작이 애니메이션화 되어 3차원적인 시각으로 동작을 볼 수 있다.

② 제어 패널 (control panel) : 로봇의 동작을 수치적으로 확인하기 위한 화면으로 “theta info” 그룹은 ⑤ 의 로봇 모델에서 구성된 각 관절의 각을 피드백 받아서 모니터링을 해주고 “motion” 그룹은 원하는 동작을 구동할 수 있게 해준다.

③ ZMP 화면 : 모멘트의 합이 ‘0’ 이 되는 위치를 보여준다. 로봇의 안정성을 판별하기 위한 화면으로 ZMP 의 값이 지면에 지지 된 발바닥 면에 놓여지는지 시각적, 수치적으로 확인을 할 수 있다. ZMP 에 관해서는 4 장의 4.1 절에서 자세히 다룬다.

④ 진행 바 (progress bar) : 프로그램이 실행될 경우 데이터량이 많은 모델이 로딩되는 경과를 알려준다.

⑤ 로봇 모델 : 파일로부터 로드 된 데이터를 강제 동역학을 시뮬레이션 하기 위한 ODE 엔진을 사용하여 구성되어있으며 프로그램의 실행 시 물리력이 적용되어 진다.

⑥ 툴 바 (tool bar) : ① 의 배경 화면을 Bitmap 구조의 그림파일로 캡처 하거나 시뮬레이션의 시작, 멈춤, 정지를 하기 위한 프레임 (frame) 관리를 해준다.

⑦ 타이틀 바 (title bar) : 프로그램의 제목과 현재 실행되고 있는 프로그램의 프레임 변화율 (frame per second : FPS) 을 표시해준다.

위와 같은 기능을 구성하기 위해서 그림 3.1 과 같이 객체별로 설계를 하였다. 응용 프로그램의 하위 레벨은 물리력의 적용과 로봇과 배경화면의 렌더링을 위하여 ODE 물리 엔진과 OpenGL 라이브러리를 사용하고 이를 동기화 하기 위해서 리눅스 환경에서 사용하는 멀티 스레드 라이브러리인 POSIX 쓰레드 [32] 를 윈도우즈 환경에 동작할 수 있게 포팅 (porting) 한 pThreadVC 라이브러리를 사용한다.



그리고 각 기능별 객체를 그림 3.3 과 같이 구조화 시키고 클래스별 멤버함수를 만들었다. 로봇의 보행을 위한 제어 알고리즘과 안정성을 계산하는 부분은 CKubir 클래스가 담당을 하며 프로그램의 실행 시 생성자 (constructor) 에 의해서 물리력을 적용하기 위한 클래스 CCodeThread 로 부터 가상의 3 차원 공간을 구성한 다음 모델링 되어진 로봇의 각 부재의 데이터 베이스를 관리하는 클래스 CMilkshapeAscii 를 이용하여 배경화면에 로드 시켜준다. 각 관절의 연결은 ODE 물리 엔진을 사용하여 구성을 하고 로봇 외의 데이터를 추가할 것을 고려하여 IModel 추상 클래스를 구성한다. 추상클래스를 사용한 것은 다중 상속을 할 때 발생하는 문제를 해결하기 위한 것이다.

시뮬레이터의 그래픽 부분은 CGraphicsSystemOpenGL 클래스가 관리를 하며 IGraphicsSystem 추상 클래스로부터 상속을 받아서 사용을 한다. 개발된 시뮬레이터는 위의 선조 클래스를 이용하여 배경화면을 이루는 하늘 (CSky) 과 땅 (CGround), 로봇 (CKubir) 을 렌더링 하고 최종적으로 보여지는 화면은 CKubirView 클래스를 이용하여 애니메이션화 시킨다. 또한 로봇이 보행 할 때 시뮬레이터에서 설정된 좌표를 벗어날 수 가 있고 동작하는 것을 정밀하게 관측할 수 있도록 CCamera 클래스를 이용한다.

### 3.2. 3D 이족보행 로봇 모델링

OpenGL 라이브러리는 3D 프로그래밍을 하는데 있어서 매우 좋은 기능을 가진 언어이지만 복잡한 3D 모델을 만드는 툴로서는 상당히 부족한 면이 많다. 그래서 3D

이족보행 로봇의 모델은 MilkShape 3D 라는 모델링 툴을 이용하여 만들었다.

MilkShape 3D 는 chUmbaLum soft 에서 개발한 폴리곤 (polygon) 모델링 툴로서 Quake 라는 게임의 캐릭터 (character) 를 모델링 하는데 사용되었으며 여러 가지 장점으로 인해 3D 게임 제작에 많이 이용된다. 또한 이 툴에서 지원하는 파일 포맷은 그림 3.4 와 같으며 Milkshape 3D ASCII 포맷은 정점 (vertex) 와 삼각형 (triangle) 으로 잘 구분이 되어 있기 때문에 OpenGL 프로그래밍으로 파싱하기에 좋은 구조를 가지고 있다.

포맷의 구조를 보면 프레임 (frame), 메쉬 (mesh), 재질 (material) 과 관절 (joint) 의 4 가지 영역으로 나눌 수 있다. 시뮬레이터에서는 제어 알고리즘을 통해 각 관절 각의 변화로 동작을 하므로 프레임을 사용하지 않는다. 프레임은 MilkShape 3D 모델링 툴에서 애니메이션으로 제작하여 이를 사용하고자 할 때 각 프레임에 해당하는 메쉬의 정보를 가지고 동작을 한다. 메쉬는 정점과 노멀 벡터 (normal vector), 삼각형 정보로 구성이 되는데 CMikshapeAscii 클래스에서 이 정보를 가져와 로봇의 부재를 구성한다. 또한 클래스에서는 재질의 정보도 가져온다. 재질은 알루미늄과 같은 속성을 가지고 로봇의 부재에 구성되며 따로 텍스처 맵핑을 이용하지 않는다. 관절은 부재 하나를 데이터 파일로 가지고 있어서 프로그램의 생성 시에 부재 하나씩 가져와 전체 로봇의 모델을 구성하므로 사용되지 않는다.

3D 시뮬레이터에서는 로봇의 각 부재에 해당하는 정보를 가지는 데이터는 OpenGL 을 이용하여 필요한 정보만을 파싱하며 각 부재는 ODE 물리 엔진의 관절을 구성하는 함수를 이용하여 구성을 한다.

```

// MilkShape 3D ASCII

// total frames
Frames: 30
// current frame
Frame: 1

// number of meshes
Meshes: 1
// mesh: name, flags, material index
"Box01" 0 0
// number of vertices
20
// vertex: flags, x, y, z, u, v, bone index
0 -10.375000 10.500000 13.750000 0.000000 0.000000 -1
:
0 10.375000 -10.250000 13.750000 1.000000 0.000000 -1
// number of normals
6
// normal: x, y, z
0.000000 0.000000 1.000000
:
0.000000 -1.000000 0.000000
// number of triangles
12
// triangle: flags, vertex index1, vertex index2, vertex index3, normal index1, normal index 2, normal index 3, smoothing group
0 0 1 2 0 0 1
:
0 13 7 19 5 5 3

// number of materials
Materials: 1
// material: name
"Material01"
// ambient
0.200000 0.200000 0.200000 0.800000
// diffuse
0.000000 0.501961 0.752941 0.800000
// specular
0.752941 0.752941 0.752941 0.800000
// emissive
0.000000 0.000000 0.000000 0.800000
// shininess
63.000000
// transparency
0.800000
// color map
"image2.jpg"
// alphasmap
"image1.jpg"

// number of joints
Bones: 1
// name
"joint1"
// parent name
""
// joint: flags, posx, posy, posz, rotx, roty, rotz
0 0.250000 0.000000 0.000000 0.224024 1.570796 0.000000
// number of position keys
2
// position key: time, posx, posy, posz
1.000000 0.000000 0.000000 0.000000
30.000000 0.000000 0.000000 0.000000
// number of rotation keys
2
// rotation key: time, rotx, roty, rotz
1.000000 0.000000 0.000000 0.000000
30.000000 0.000000 0.000000 0.000000

```

그림 3.4 MilkShape 3D ASCII 파일 포맷의 사양

### 3.3. 물리 엔진의 소개

개발된 시뮬레이터는 실제와 유사한 물리 법칙을 필요로 한다. 물리학 관점에서 로봇은 여러 개의 조인트를 연결한 강체의 집합이다. 하나의 부재에 적용되는 힘과 토크는 연결된 부재에 영향을 미치고 반발력이 작용할 수 있다. 따라서 모든 부재는 중력에 영향을 받게 되므로 가상의 강체 동역학을 시뮬레이션 하기 필요하다. 강체를 시뮬레이션 할 수 있는 라이브러리는 많이 있지만 가격이 비싸다. 그래서 무료이면서 검증되어진 Open Dynamics Engine (ODE) 이라는 물리 엔진을 사용한다.

ODE 물리 엔진에서 사용되는 부재는 질량 중심에 해당하는 위치 벡터, 선속도, 부재의 방향, 각속도 벡터와 같은 성질을 가진다. 이와 같은 성질들은 시간에 따라 변하는 동적인 것이며 정적인 것은 부재의 질량, 질량의 중심 위치와 관성 행렬이다. 또한 그림 3.5 와 같이 볼 (ball) 과 소켓 (socket), 힌지 (hinge), 유니버설 (universal), 슬라이더 (slider) 등의 다양한 조인트를 사용할 수 있다.

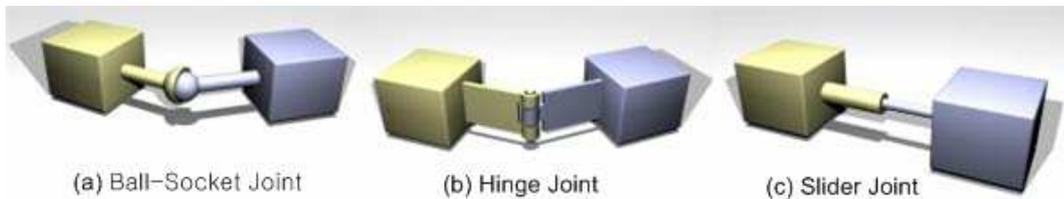


그림 3.5 ODE 에서 활용되는 조인트

3D 이족보행 로봇 모델은 총 15 자유도를 가지므로 15 개의 힌지 조인트와 하체의 허리, 무릎, 발목의 관절을 구동하기 위한 6 개의 슬라이더 조인트로 이루어져 있

고 그 외에 부재를 연결하기 위한 고정과 힌지 조인트로 구성된다.

### 3.4. 제어부의 구성

이족보행 로봇의 관절부에 제어기를 구성하지 않으면 중력의 영향으로 시뮬레이션 수행 시 곧바로 로봇이 땅에 떨어질 것이다. 하드웨어적으로는 모터의 브레이크 기능이 있어 로봇이 서있을 수 있으나 소프트웨어적인 처리는 이러한 기능이 없으므로 제어 알고리즘을 적용해야만 한다. 또한 보행을 하기 위해서는 각 관절의 시간에 따른 궤적 (trajectory) 을 활용해야 하는데 이를 위해서 개발된 시뮬레이터에 PID 제어 알고리즘을 적용하였다.

PID 제어는 자동제어 방식 가운데서 가장 흔히 사용되는 제어방식으로 비례 ( $K_p$ ), 적분 ( $K_i$ ), 미분 ( $K_d$ ) 의 3 가지 인자를 조합으로 유연한 제어가 가능해진다. 이는 오차신호  $e(t)$  를 피드백 (feedback) 받아서 제어신호  $u(t)$  를 계산한다.

$$u(t) = K_p e(t) + K_i \int e(t) dt + K_d \frac{de(t)}{dt} \quad (3.4.1)$$

원래 PID 제어는 식 (3.4.1) 과 같이 연속한 아날로그량을 제어하는 것이 기본으로 되어 있다. 그러나 시뮬레이터의 데이터 입출력은 일정시간 간격을 가지고 미/적분 연산의 빠른 데이터 처리를 위해 고성능의 컴퓨터가 필요하게 되므로 식 (3.4.2) 와 같이 이산화를 해야 한다.

$$u(n) = u(n-1) + \Delta u(n) \tag{3.4.2}$$

$$\Delta u(n) = K_p[e(n) - e(n-1)] + K_i e(n) + K_d[(e(n) - e(n-1)) - (e(n-1) - e(n-2))]$$

시뮬레이터에서는 부재의 지지를 위해 각 관절의 모터부에 토크 출력 값이 필요하므로 관절의 모터부를 제어하기 위해서 식 (3.4.2) 를 이용해 다음과 같은 토크  $\tau$  제어 신호를 얻을 수 있다.

$$\tau(n) = \tau(n-1) + \Delta \tau(n) \tag{3.4.3}$$

$$\Delta \tau(n) = K_p[\theta(n) - \theta(n-1)] + K_i \theta(n) + K_d[(\theta(n) - \theta(n-1)) - (\theta(n-1) - \theta(n-2))]$$

## 제 4 장 보행 시뮬레이션

로봇이 보행을 할 때에 로봇의 안정성을 판별하는 기준 조건으로 사용하는 Zero Moment Point (ZMP) 의 개념은 Vukobratovic 에 의해 처음으로 제시되었다. 이 개념을 사용하여 Gruver [33] 는 물리적 로봇에 대해 실험적으로 ZMP 값의 궤적을 분석하였고 ZMP 궤적에 근거하여 사용되어진 매개 변수의 영향을 찾아내었다. 실시간으로 ZMP 값을 계산하기 위해서는 각각의 구성요소의 가속도를 계산해야만 한다. Stark [34] 는 각 부재의 가속도를 계산하기 위한 독창적인 근사화를 제시하였다.

### 4.1. 보행의 안정도 분석

보행 로봇의 물리적 허용을 규정하기 위하여 Vukobratovic 은 로봇에 작용하는 힘과 모멘트의 합이 평형이 되는 점을 ZMP 로 정의하였다. [1] 과 [7] 에서는 모든 힘 (중력, 반력, 관성력) 이 평형이면 로봇의 운동이 물리적으로 실행될 수 있다고 말하고 있다. 다시 말해 바닥에 접촉된 발바닥에 임의의 기준 좌표 계를 설정해 놓고 그 점에 대해서 모멘트 합 조건을 적용했을 때 로봇 발 위쪽 링크 (link) 들의 전체 운동량을 보상하기 위한 반발력이 발바닥의 어느 한 점에 존재할 경우 보행 로봇은 넘어지지 않고 보행 할 수가 있는 것이다.

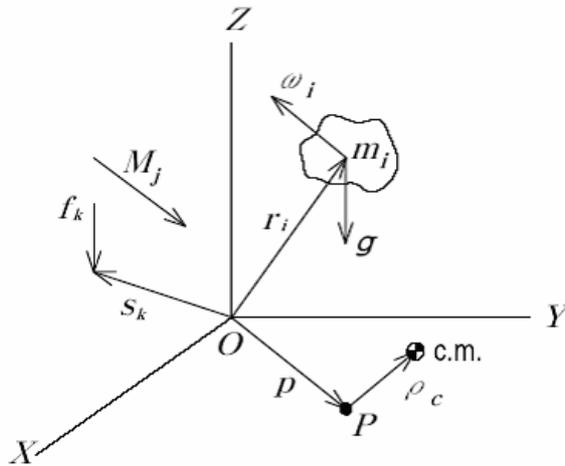


그림 4.1 강체 (rigid body) 의 시스템

그림 4.1 에서  $i$  번째 부재가 질량  $m_i$ , 관성 좌표계  $O_{xyz}$  에서 상대적인 질량의 중심  $r_i = [x_i, y_i, z_i]^T$ , 질량 중심에 관한 관성력  $I_i$  그리고 각속도  $\omega_i$  를 가진다고 가정하고 시스템의 외부 힘과 모멘트를 각각  $f_k$  와  $M_j$  라고 한다. D'Alembert 의 원리에 의해 임의의 한 점 P(P의 위치벡터  $p = [x_p, y_p, 0]^T$  에 관한 시스템의 운동 방정식은 식 (4.1.1) 과 같다.

$$\begin{aligned}
 & \sum_i (r_i - p) \times m_i (\ddot{r}_i + g - \ddot{p}) + \rho_c \times m \ddot{p} + \sum_i [I_i \cdot \dot{\omega}_i + \omega_i \times I_i \cdot \omega_i] \\
 & - \sum_j M_j - \sum_k (s_k - p) \times f_k = M_p
 \end{aligned} \tag{4.1.1}$$

여기서,  $g = [0, 0, g_z]^T$  는 중력가속도,  $s_k$  는 작용한 힘의 위치,  $m = \sum_i m_i$  를 의

미한다. 식 (4.1.1) 에서의  $\rho_c$  는 다음과 같다.

$$\rho_c = \frac{1}{m} \sum_i m_i (r_i - p) \quad (4.1.2)$$

식 (4.1.2) 로 부터, 식 (4.1.1) 에서의  $\rho_c \times m\ddot{p}$  는

$$\rho_c \times m\ddot{p} = \frac{1}{m} \sum_i m_i (r_i - p) \times m\ddot{p} = \sum_i (r_i - p) \times m_i \ddot{p} \quad (4.1.3)$$

이다. 그러므로, 식 (4.1.1) 은 아래와 같이 된다.

$$\sum_i (r_i - p) \times m_i (\ddot{r}_i + g) + \sum_i [I_i \cdot \dot{\omega}_i + \omega_i \times I_i \cdot \omega_i] - \sum_j M_j - \sum_k (s_k - p) \times f_k = M_p \quad (4.1.4)$$

여기서, 외부 힘과 모멘트는 작용하지 않는다. 그래서

$$\sum_i (r_i - p) \times m_i (\ddot{r}_i + g) + \sum_i [I_i \cdot \dot{\omega}_i + \omega_i \times I_i \cdot \omega_i] = M_p \quad (4.1.5)$$

식 (4.1.5) 에서  $p$  를 포함하지 않는  $T_i = (I_i \cdot \dot{\omega}_i + \omega_i \times I_i \cdot \omega_i)$  로 치환하면, 다음과 같은 방정식을 얻을 수 있다.

$$\sum_i (r_i - p) \times m_i (\ddot{r}_i + g) + \sum_i T_i = M_p \quad (4.1.6)$$

식 (4.1.6) 에서 ZMP의 정의에 의해  $p = p_{zmp} (= [x_{zmp}, y_{zmp}, 0]^T)$  일 때  $M_p = [0, 0, M_z]^T$  이다. 마지막으로 식 (4.1.7) 과 같은 ZMP의 구성 요소를 얻을 수

있다.

$$x_{zmp} = \frac{\sum_i m_i (\ddot{z}_i + g) x_i - \sum_i m_i \ddot{x}_i z_i - \sum_i (T_y)_i}{\sum_i m_i (\ddot{z}_i + g)},$$

$$y_{zmp} = \frac{\sum_i m_i (\ddot{z}_i + g) y_i - \sum_i m_i \ddot{y}_i z_i - \sum_i (T_x)_i}{\sum_i m_i (\ddot{z}_i + g)}$$

(4.1.7)

식 (4.1.7) 을 실제 표 4.1 과 같은 개발되어진 이족보행 로봇의 모델 변수를 시뮬레이터에 적용하였다.

표 4.1 이족보행 로봇 모델 변수

링크	길이 [m]	질량 [Kg]
발	0.181	11.16
종아리	0.35	5.469
허벅지	0.35	5.369
상체	0.829	39.022

## 4.2. 보행 주기의 결정

시뮬레이터를 이용하여 이족보행 로봇의 보행을 구현하기 위해서 그림 4.3 과 같이 8 단계의 보행 주기를 만들었다.

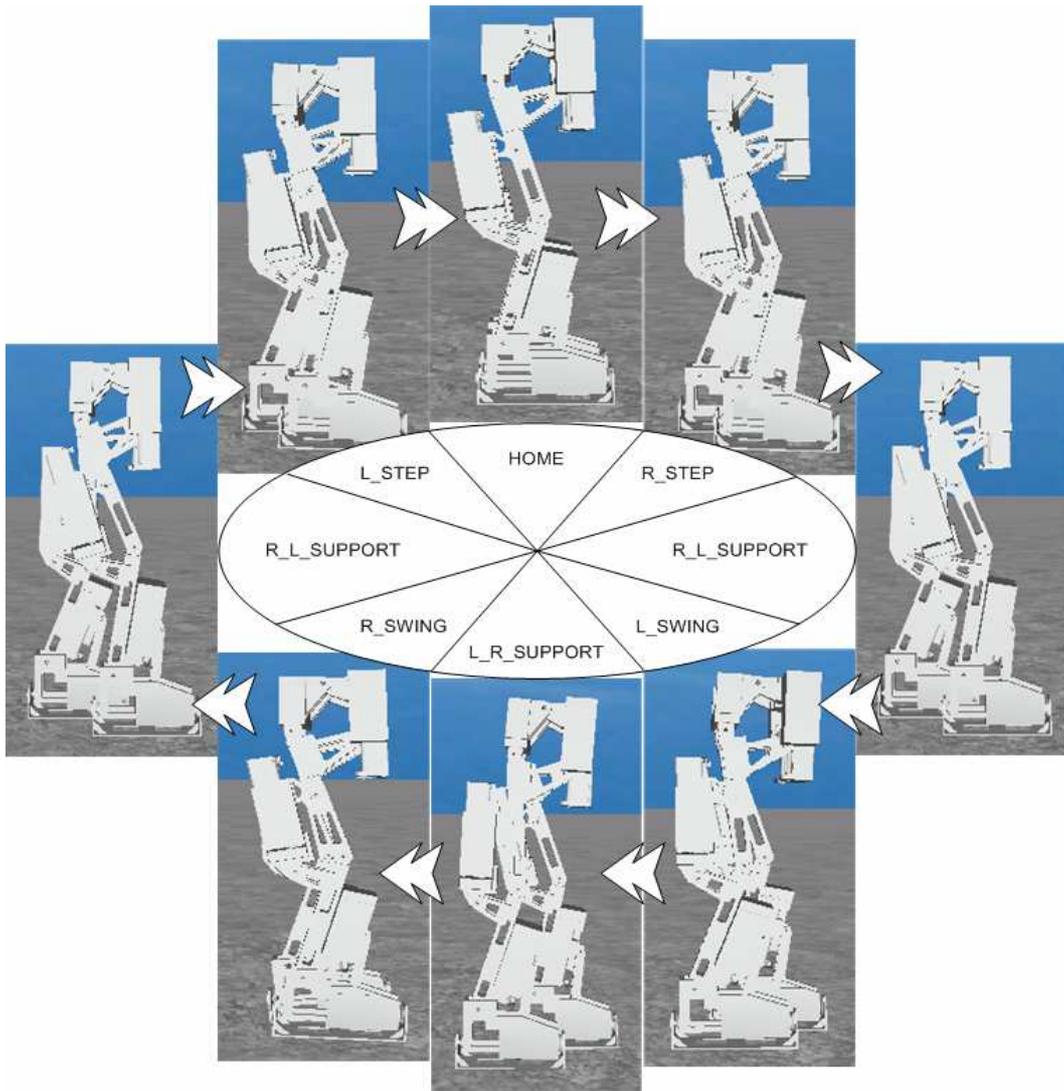


그림 4.2 이족보행 로봇의 보행 주기

한 보행 주기를 각 단계별로 정의를 하면 다음과 같으며 보행 시 오른쪽 발을 먼저 내미는 경우로 가정한다.

- ① 단계 (HOME :  $t_h$ ) : 시뮬레이터에서 로봇을 생성할 때 각 관절각이  $0^\circ$ 로 되어 있어 사람이 서있는 자세와 다르므로 개발되어진 동작 획득 장치 [34]를 이용하여 각 관절각의 정보를 이용해 동기화 시킨다.
- ② 단계 (R\_STEP :  $t_{rs}$ ) : 보행을 위해 오른쪽 발을 내미는 동작 주기를 나타낸다.
- ③ 단계 (R\_L\_SUPPORT :  $t_{rls}$ ) : 오른쪽 발이 내밀어진 다음 왼쪽 발을 내밀기 위한 전 동작으로 오른쪽 발이 무게 중심의 앞쪽에 왼쪽 발이 무게 중심의 뒤쪽으로 발이 땅에 지지 되는 동작 주기를 나타낸다.
- ④ 단계 (L\_SWING :  $t_{lsw}$ ) : 지지 된 오른쪽 발을 거쳐서 왼쪽 발을 내미는 동작 주기를 나타낸다.
- ⑤ 단계 (L\_R\_SUPPORT :  $t_{lrs}$ ) : ③ 단계와 반대로 왼쪽 발이 무게 중심의 앞쪽에 오른쪽 발이 무게 중심 뒤쪽으로 발이 땅에 지지 되는 동작 주기를 나타낸다.
- ⑥ 단계 (R\_SWING :  $t_{rsw}$ ) : ④ 단계와 반대로 지지 된 왼쪽 발을 거쳐서 오른쪽 발을 내미는 동작 주기를 나타낸다.
- ⑦ 단계 (R\_L\_SUPPORT :  $t_{rls}$ ) : ③ 와 같다.
- ⑧ 단계 (L\_STEP :  $t_{ls}$ ) : 한 보행을 마무리 하는 동작으로 ① 단계로 돌아오기 위하여 왼쪽 발을 오른쪽 발과 같이 y축에 오도록 하는 동작 주기를 나타낸다.

위의 각 단계를 이용하여 한 보행을 하기 위해서는 ①→②→③→④→⑤→⑥→⑦→⑧→① 과 같은 순서로 동작을 구현한다.

### 4.3. 보행을 위한 각 관절의 궤적 생성

그림 4.3 과 같은 보행 주기를 이용하여 동작을 하기 위해서는 식 (3.4.3) 을 이용한 제어 함수를 가지고 각 관절을 제어할 필요가 있다. 각 관절의 위치를 결정하기 위해서는 각 보행 주기에 해당하는 궤적을 생성해야 한다.

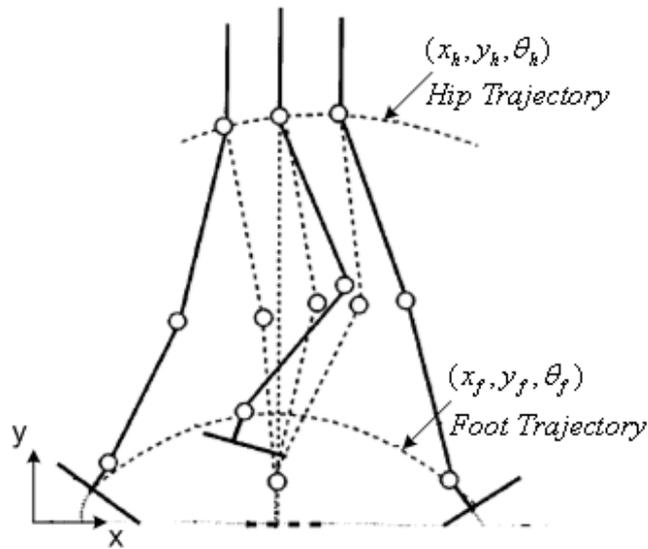


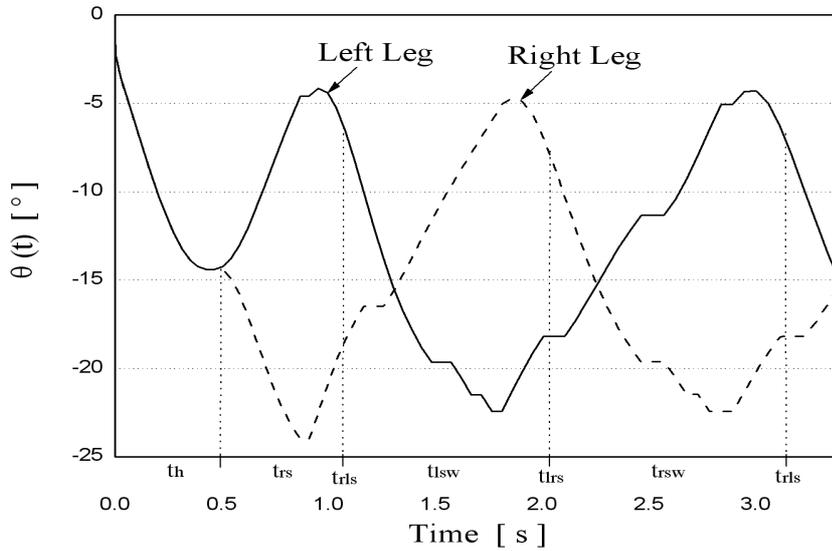
그림 4.3 보행을 위한 허리와 발목 관절의 궤적함수 생성

궤적을 생성하기 위한 궤적 함수는 주기가  $t_{rsw}$  일 때의 궤적을 나타내는 그림 4.4 와 같이 허리 관절과 발목 관절의 궤적 함수를 만들고 역기구학 방정식을 이용하면

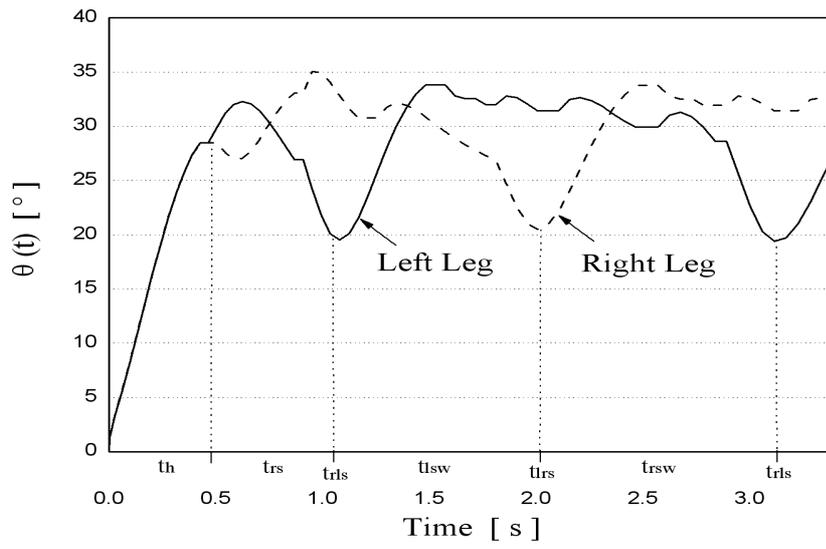
각 관절의 각을 얻을 수 있다. 주기별 관절 함수는 주기 함수를 이용하여 다음과 같다.

$$\begin{aligned}
 x_h(t) &= t, & t = t_{rs} = t_{ts} = t_{rsw} = t_{lsw} \\
 y_h(t) &= \begin{cases} h + \left\| 0.0025 \times \sin(t) \right\|, & t = t_{rs} = t_{ls} \\ h + \left\| 0.0025 \times \sin\left(\frac{t}{2}\right) \right\|, & t = t_{rsw} = t_{lsw} \end{cases} \\
 x_f(t) &= t, & t = t_{rs} = t_{ts} = t_{rsw} = t_{lsw} \\
 y_f(t) &= \begin{cases} \left\| 0.01 \times \sin(t) \right\|, & t = t_{rs} = t_{ls} \\ \left\| 0.01 \times \sin\left(\frac{t}{2}\right) \right\|, & t = t_{rsw} = t_{lsw} \end{cases}
 \end{aligned} \tag{4.3.1}$$

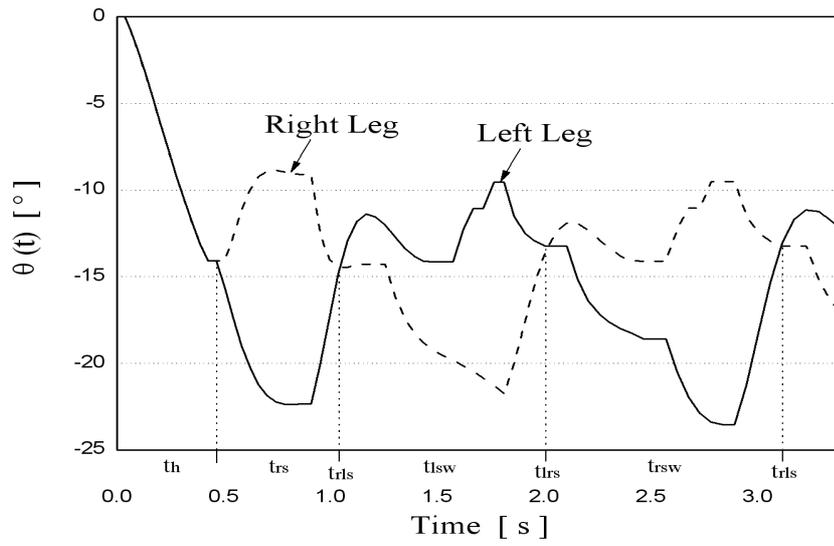
여기서  $h$ 는 HOME 단계에서 허리 관절의 중심 위치와 지면의 높이 차이이다.



(a) 허리 관절각의 궤적



(b) 무릎 관절각의 궤적



(c) 발목 관절각의 궤적

그림 4.4 각 관절각의 궤적

식 (4.3.1) 을 이용하여 획득한 그림 4.4 와 같은 관절 각의 궤적의 타당성을 검증하기 위하여 개발되어진 동작획득 장치 [35] 를 이용하여 얻은 관절각의 궤적은 그림 4.5 와 같으며 허리와 발목의 궤적은 유사함을 알 수 있다. 하지만 무릎 관절의 궤적은 다소 차이가 있는데 이는 보행 시 지면에 닿는 순간에 생기는 반발력으로 인한 요소로 시뮬레이터에 적용된 물리력이 실제 환경과 똑같지 않기 때문이다.

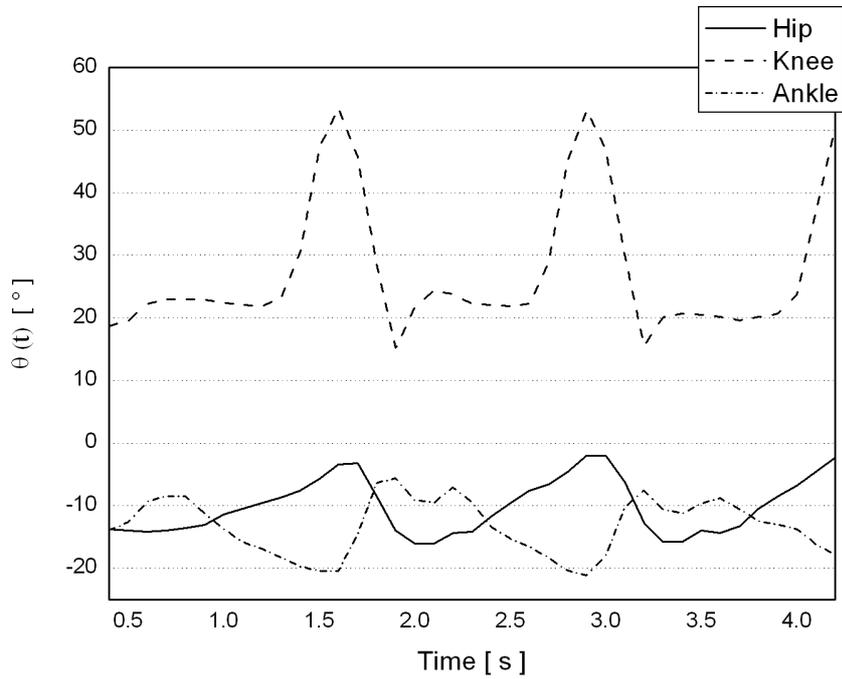
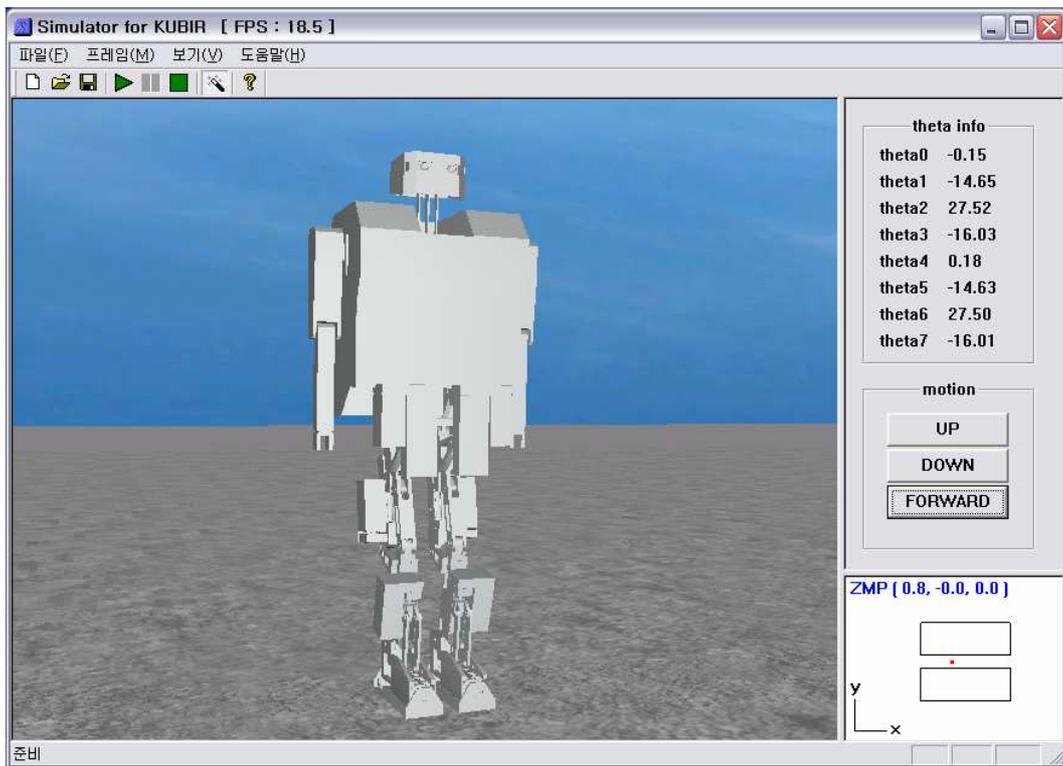
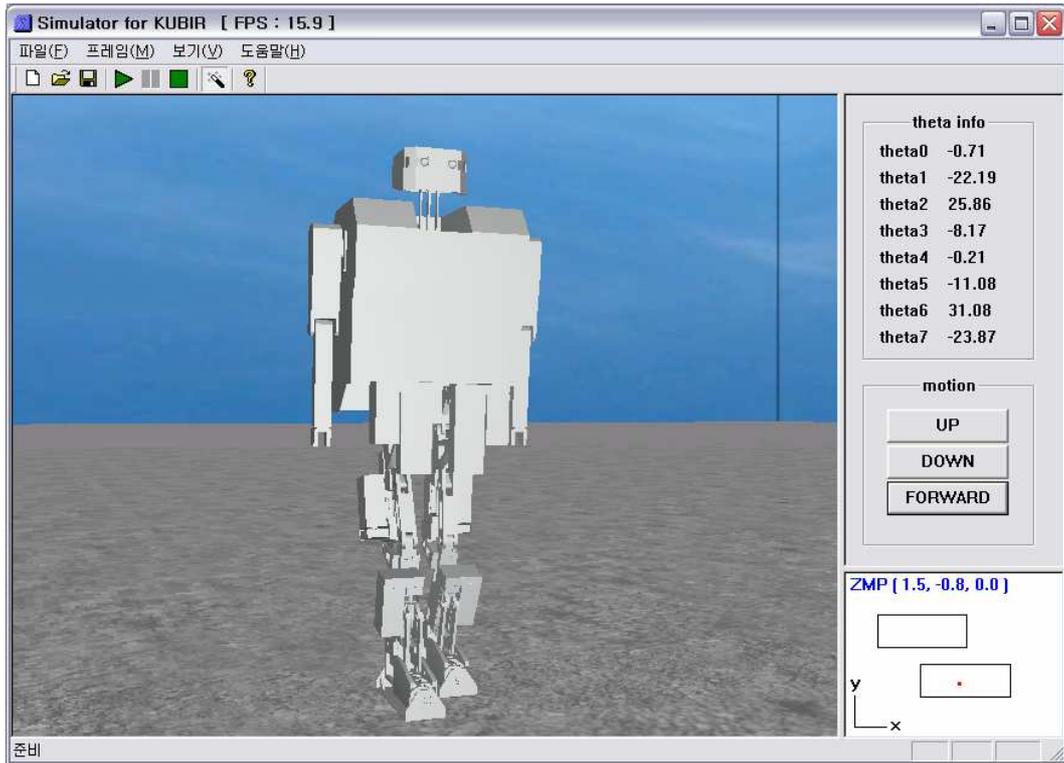


그림 4.5 동작획득 장치를 이용한 관절각의 궤적

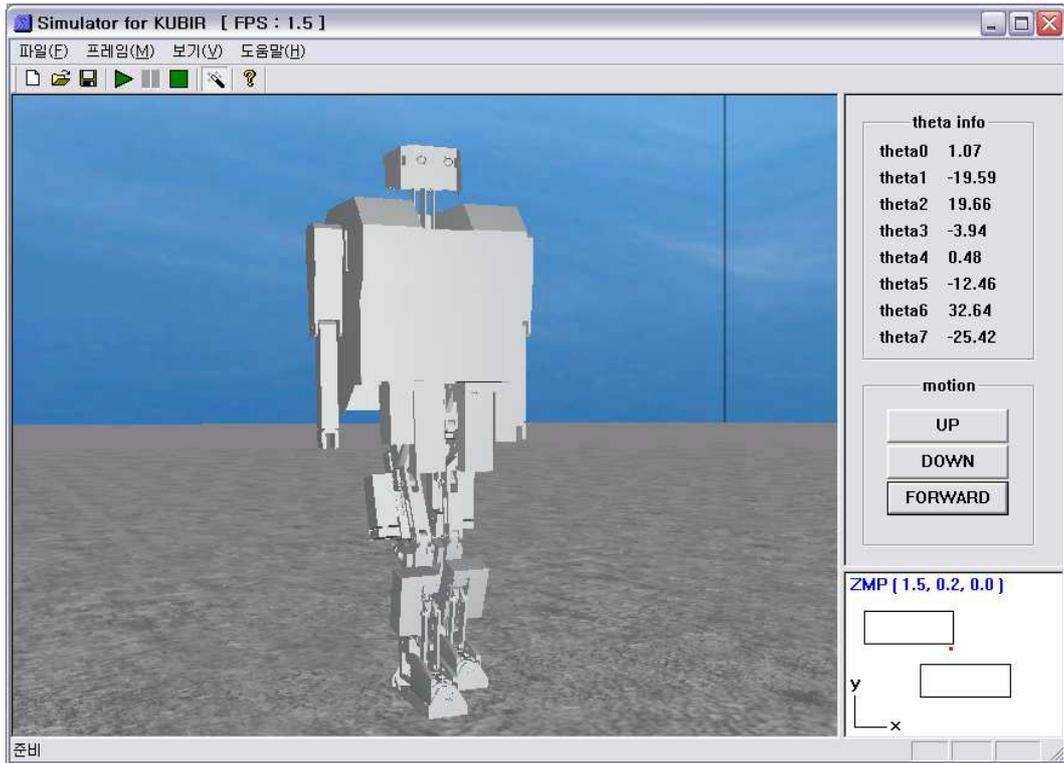
또한 획득한 관절 각을 식 (3.4.3) 제어기의 목표 입력으로 주면 그림 4.6 과 같은 시뮬레이터 결과를 얻을 수 있다.



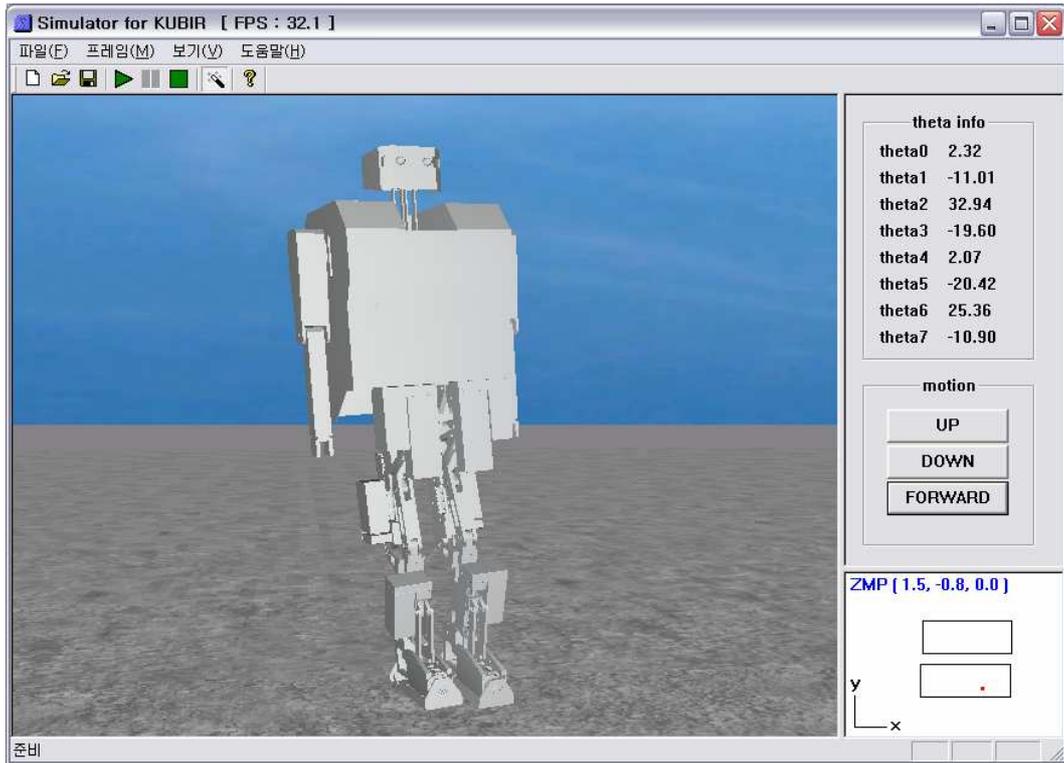
(a) 주기  $t_h$  일 때의 보행 동작



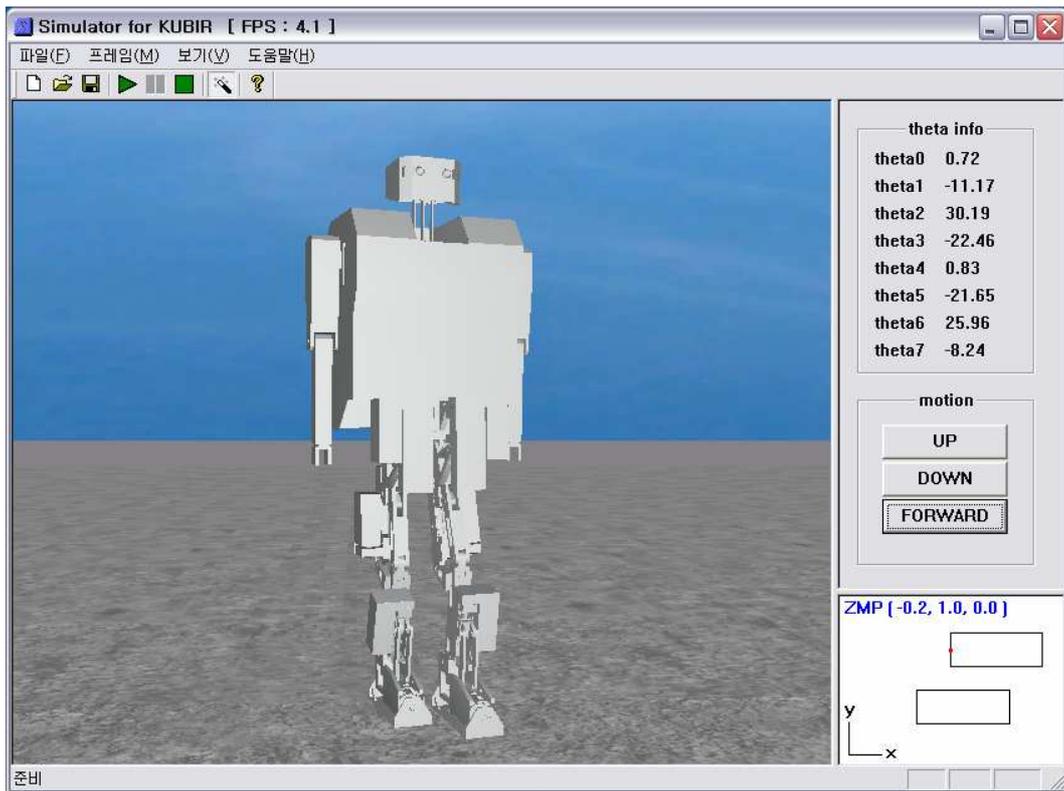
(b) 주기  $t_{rs}$  일 때의 보행 동작



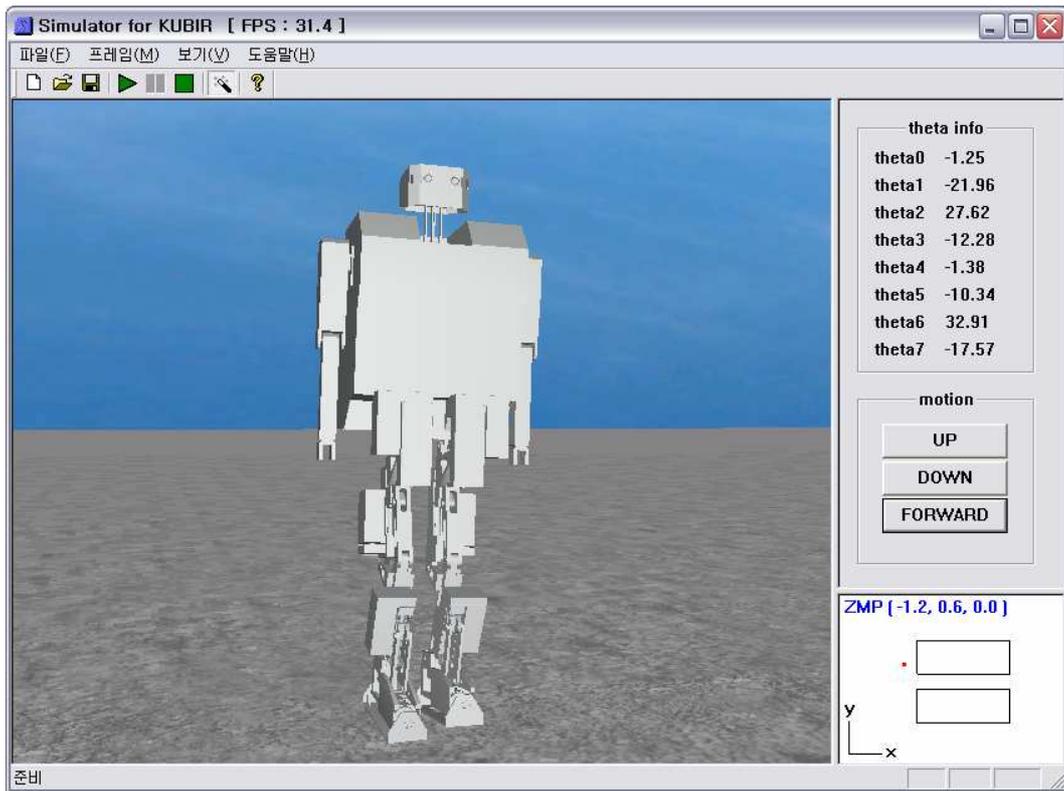
(c) 주기  $t_{rls}$  일 때의 보행 동작



(d) 주기  $t_{lsw}$  일 때의 보행 동작



(e) 주기  $t_{lrs}$  일 때의 보행 동작



(f) 주기  $t_{rsv}$  일 때의 보행 동작

그림 4.6 시뮬레이터를 이용한 이족보행 로봇의 보행 동작

또한 시뮬레이터를 사용하여 그림 4.4 의 궤적을 식 (4.1.7) 의 안정도 분석에 적용한 결과는 그림 4.7 과 같다. 시뮬레이터의 물리 엔진에서는 ZMP 를 계산하기 위해서 부재의 속도 값을 출력할 수 있도록 함수로 지원을 해준다. 이 속도 함수와 프레임간의 시간 간격차 (time interval) 를 이용하여 각 부재의 가속도를 계산하여 적용하였다.

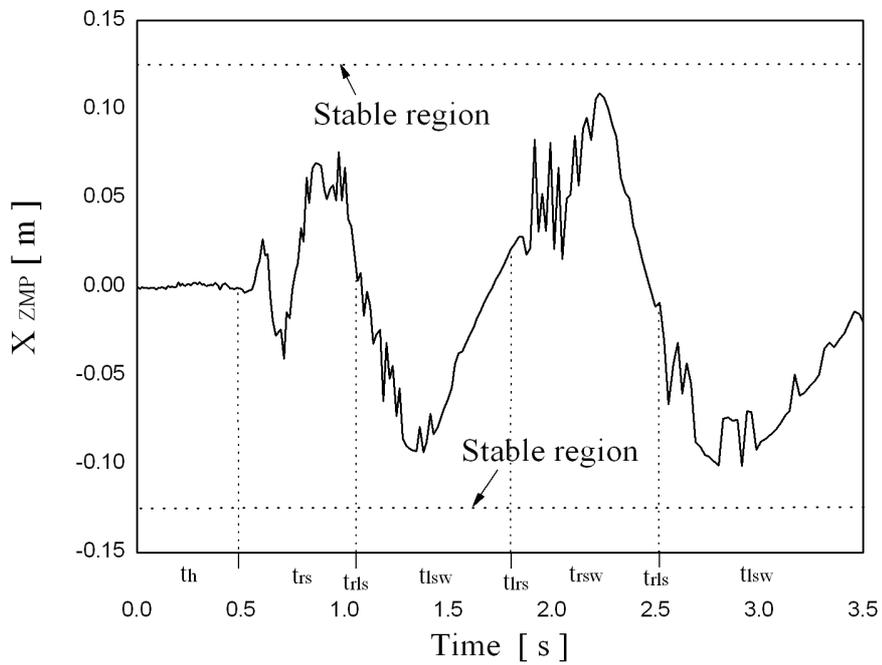
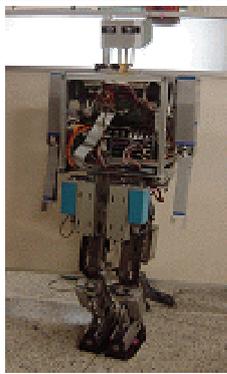
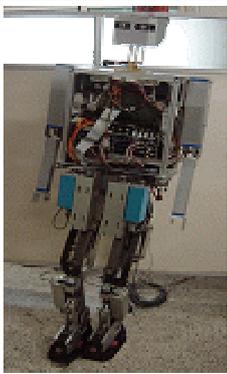
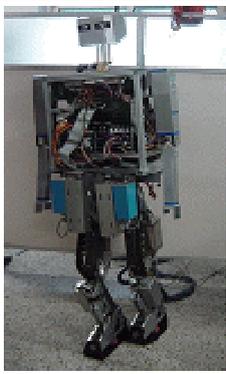
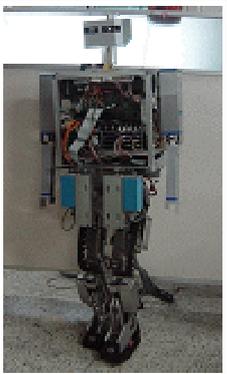
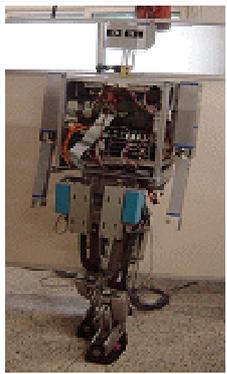
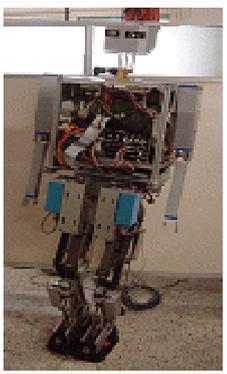
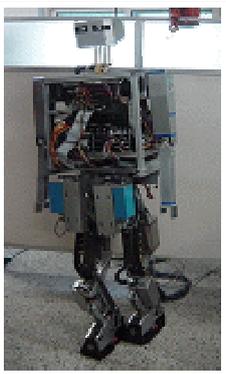
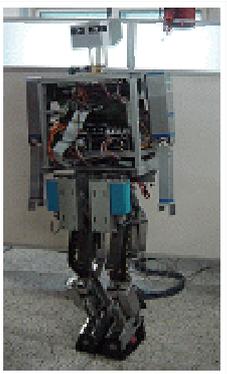


그림 4.7 ZMP 의 궤적

#### 4.4. 궤적을 이용한 로봇의 구동

개발된 이족보행 로봇의 제어기에 그림 4.4 의 값을 입력으로 한 결과는 표 4.2 처럼 실제 구동으로도 안정한 보행이 가능한 것을 확인할 수가 있다.

표 4.2 궤적을 이용한 로봇의 구동

단계	① HOME	② R_STEP	③ R_L_SUPPORT	④ L_SWING
자세				
단계	⑤ L_R_SUPPORT	⑥ R_SWING	⑦ R_L_SUPPORT	⑧ L_STEP
자세				

## 제 5 장 결 론

본 논문에서는 실제 이족보행 로봇의 보행을 위하여 관절 구동기를 장착한 하드웨어적 실험의 문제점인 위험적, 경제적, 시간적인 요소를 줄이는 목적으로 OpenGL 라이브러리와 ODE 물리 엔진을 이용한 3D 시뮬레이터를 개발하였다. 실제 로봇의 부재에 해당하는 3D 모델을 Milkshape 3D Ascii 포맷으로 데이터 베이스화하고 실제 로봇의 질량과 길이를 적용하여 3 차원적으로 보행을 실험하였다. 보행 시뮬레이션 시에 실제 구동부의 모터 변수를 입력하여 부하 토크의 계산을 이용해 에너지 효율을 확인하고 로봇의 안전한 보행을 위해 ZMP 분석을 하였으며 관절을 구동하기 위해 이산화 된 PID 제어 알고리즘을 적용하여 인간과 흡사한 보행을 만들기 위해 8 단계의 보행 주기를 결정하여 각 관절각의 궤적을 생성하였다.

생성된 각 관절각의 정보를 바탕으로 하여 개발된 실제 이족보행 로봇의 제어 시스템에 적용하여 시뮬레이터에서 얻은 결과와 유사한 보행 동작을 만들 수가 있었으므로 목적인 바와 같이 하드웨어적 실험의 문제점을 보완할 수 있음을 입증하였다. 또한 향후 추가될 관절, 다양한 보행 패턴과 지형적인 요소를 첨가할 수 있게 시뮬레이터를 설계하였으므로 실제 기구부를 재구성하기 전 시뮬레이션 결과를 보고 시스템의 안정성과 설계의 정확성을 확인할 수가 있다.

향후 보행 동작의 다양한 패턴과 독립적인 로봇의 개발을 위한 에너지 효율의 최적화와 지형적인 환경에 적응할 수 있는 이족보행 로봇을 개발하기 위해 강화 학습과 같은 지능적인 알고리즘을 적용하고 또한 동작 획득 장치의 개선을 통해 인간과 같은 동작 주기를 결정해야 하는 문제가 남아있다.

## 참고문헌

- [1] M. Vukobratovic, A. A. Frank, and D. Juricic. *On the Stability of Biped Locomotion*. Proc. IEEE Transactions on Biomedical Engineering, Vol. BME-17, No.1, pp. 25-36, 1970.
- [2] M. Vukobratovic. *Mathematical Models of Legged Locomotion Robots*. Mihailo Pulpin Institute Beograd, 1975.
- [3] A. Takanishi, M. Ishida, Y. Yamazaki, I. Kato. *The Realization of Dynamic Walking by The Biped Walking Robot WL-10RD*. Journal of the Robotics Society of Japan, Vol.3, No.4, pp.325-336, 1985.
- [4] F. Gubina, H. Hemami., R. B. McGhee. *On the Dynamic Stability of Biped Locomotion*. IEEE trans. on Biomd. Engineering, Vol, BME-21, pp.102-108, 1974.
- [5] T. T. lee and J. Liao. *Trajectory Planning and Controlk of a 3-link Biped Robot*. IEEE Robotics and Automation, pp. 820-823, 1988.
- [6] M. Cotsatis and C. Vibet. *Decoupled control for 2D N-link Biped Walking System*. Robotics and Autonomous Systems 5 pp.97-107, 1989.
- [7] J. Furusho, A. Sano. *Sensor-Based Control of a Nine-Link Biped*. Int. J. Robot. Res., vol.9, No.2, pp.83-98, 1990.
- [8] C. L. Golliday, H. Hemami. *An Approach to Analyzing Biped Locomotion Dynamics and Designing Robot Locomotion Controls*. IEEE, Transactions on Automatic Control, vol.AC-22, No.6, pp.963-972, 1977.
- [9] T. Chang and Y. Hurmuzlu. *Sliding Control without Reaching Phase and its Application to Bipedal Locomotion*. J. of Dynamic Systems, Measurement, and Control, vol.447-455, 1993.

- [10] M. Vukobratovic and J. Stepanenko. *Mathematical Models of General Anthropomorphic Systems*. Mathematical Biosciences 17, pp.191-242, 1973.
- [11] S. H. Lim and J. G. Kim. *Biped Locomotion with Mechanical Compliance*. Korea Robotics and Automation Workshop Conference, pp.21-25, 1992.
- [12] Kawamura S. et al. *Realization of Biped Locomotion by Motion Pattern Learning*. Journal of Robot Society of Japan vol.3, No.3, pp.177-180, 1985.
- [13] K.Ohnishi, M.Shibata & T.Murakami. *Motion Control for Advanced Mechatronics*. IEEE.ASME Trans. On Mechatronics, Vol.1, No.1, pp56-67, 1996
- [14] C. L. Shih. *Analysis of the Dynamics of a Biped Robot with Seven Degrees of Freedom*. IEEE International Conference on Robotics and Automation, pp.3008-3013, 1996.
- [15] T. A, Egusa. Y, Tochizawa. M, Takeya. M, and Kato. I. *Realization of Dynamic Biped Walking Stabilized with Trunk Motion*. RoManSy 7: Proc. Seventh CISM-IFTOMM Symposium on Theory and Practice of Robots and Manipulators, A.Morecki, G.Bianchi and KoJaworek, Eds. Hermes, Paris, pp.68-79, 1990.
- [16] Hirai, K., et al. 1998. *The development of Honda humanoid robot*. Proceedings of ICRA 2:1321-1326.
- [17] Ichiro Kato. *Development of Waseda Robot*. Humanoid Robotics Institute, Waseda University, 2000.
- [18] J. Pratt and G. Pratt. *Exploiting natural dynamics in the control of a*

- planar bipedal walking robot*. Proceedings of the Thirty-Sixth Annual Allerton Conference on Communication, Control, and Computing, pages 739–748, 1998.
- [19] Pratt, Jerry, Pratt, Gill 1999. *Exploiting Natural Dynamics in the Control of a 3D Bipedal Walking Simulation*. Proceedings of the International Conference on Climbing and Walking Robots (CLAWAR99), Portsmouth, UK, September 1999.
- [20] Stéphane Mojon. *Realization of a Physic Simulation for a Biped Robot*. Swiss Federal Institute of Technology Lausanne (EPFL), School of Computer and Communication Sciences, 27th of June 2003.
- [21] Joon Ng. *An Anthropomorphic Bipedal Robot*. Centre for Intelligent Information Processing Systems Department of Electrical/Electronic Engineering. The University of Western Australia. Final Year Project Report 1998.
- [22] Elliot Nicholls. *Bipedal Dynamic Walking in Robotics*. The University of Western Australia Department of Electrical and Electronic Engineering. October 26, 1998.
- [23] Hun-ok Lim and Atsuo Takanishi. *Waseda Biped Humanoid Robots Realizing Human-like Motion*. Department of Mechanical Engineering School of Science and Engineering, Waseda University Advanced Research Institute for Science and Engineering, Waseda University. IEEE 2000.
- [24] Jong H. Park and Kyoung D. Kim. *Biped Robot Walking Using Gravity-Compensated Inverted Pendulum Mode and Computed Torque Control*.

- IEEE International Conference on Robotics & Automation Leuven, Belgium 1 May 1998.
- [25] Yasutaka Fujimoto, Atsuo Kawamura. *Simulation of an Autonomous Biped Walking Robot Including Environmental Force Interaction*. IEEE Robotics and Automation Magazine, Vol. 5, No. 2, June 1998.
- [26] Yutaka Nakamura, Masa-aki Sato, and Shin Ishii. *Reinforcement Learning for Biped Robot*. Proceedings of the 2nd International Symposium on Adaptive Motion of Animals and Machines, Kyoto, March.4-8, 2003.
- [27] Russell Smith. *Open Dynamics Engine v0.039 User Guide*. <http://opende.sourceforge.net/>. Saturday 9 August, 2003.
- [28] Hyung-Sik Choi. *Development of Human - Sized Biped Robot*. Korean Society of Precision Engineering. pp.15-18, 1225-9071. 2000.
- [29] Mark W. Spong, M. Vidyasagar. *Robot Dynamics and Control*. John Wiley & Sons. Pages 74 – 85. 1989.
- [30] Gaurav Tevatia, Stefan Schaal. *Inverse Kinematics for Humanoid Robots*. IEEE International Conference on Robotics and Automation. 2000.
- [31] Richard S. Wright JR & Michael Sweet. *OpenGL Superbible*, The Waite Group, 1996.
- [32] Ulrich Drepper, Ingo Molnar. *The Native POSIX Thread Library for Linux*. Red Hat, Inc. January 30, 2003.
- [33] C. L. Shih, Y.Z. Li, S. Churng, T. T. Lee, and W. A. Gruver. *Trajectory Synthesis and Physical Admissibility for a Biped Robot During the*

*Single-Support Phase*. Proc. IEEE International Conference on Robotics and Automation, Vol.3, pp. 1646 - 1652, 1990.

- [34] K. Stark. *Simulation of a Two-Legged Locomotion*. University of Wisconsin-Madison, Mechanical Engineering Department, Dec. 1993.
- [35] Jeong-Jae Woo. *Development of Motion Capture System*. Korean Society of Precision Engineering. pp.139-146, Oct. 2000.

## 감사의 글

책장에 꽂혀있는 책들을 바라보며...

어지럽게 진열되어 있는 무수한 책들이 낡고 지저분해져 있지만 보내온 시간을 대변해 주는 듯 합니다. 시커멓게 변해버린 키보드와 마우스 그리고 데스크 탑 역시 3년이라는 세월을 열심히 잘해왔다는 듯 책상에 놓여져 있습니다. 이렇듯 긴 시간을 보내오며 대학원을 마무리하는 시점에 본 논문을 지필 하는데 많은 노력과 격려를 해주신 분들에게 감사의 말을 전하러 합니다.

내가 나아가야 할 진로와 올바른 길을 걸어 올 수 있도록 언제나 버팀목이 되어주시고 부족한 저에게 칭찬을 아껴주시지 않고 걱정해주신 최 형식 교수님과 논문 수정에 많은 도움을 주신 정 재현 교수님, 김 환성 교수님 그리고 대학원 수업할 때 성심껏 지도해주신 조 석제 교수님, 김 재훈 교수님에게 깊이 고개 숙여 감사하다는 말씀을 드리고 싶습니다.

연구실에서 같이 생활하면서 단체라는 것에 대해 많은 생각을 하게 해주신 박 용현, 이 호식, 전 대원, 주 상혁 형들과 많은 가르침을 주고 언제나 연구실 발전을 위해 노력하시는 김 영식 형 그리고 2년 동안 동고동락해온 정 경식, 오 정민, 김 유신, 백 창열 형들에게도 감사한 마음을 전하고 싶습니다. 또한 앞으로 연구실을 지탱할 김 무경, 이 창만, 황 규득 형들과 전 창훈 그리고 저와 같이 웃음을 나눈 손 석봉, 손 정호, 오 주환 형들과 상준 이에게 언제나 서로 도와가며 연구실 발전에 힘써달라는 부탁의 말과 좋은 일들만 가득하기를 바랍니다.

다른 연구실에서 열심히 연구하시는 종식이 형, 민수 형, 종욱이 형, 성윤이 형과 학부 생활 동안 언제나 좋은 말씀과 관심을 가져주신 냉동공학과 95 학번 형들에게 앞으로 좋을 일만 가득했으면 합니다.

논문을 쓰면서 어려운 부분과 제가 공부하는데 많은 도움을 주신 계명대학교의 배인호 형과 항상 동생처럼 챙겨주시는 ITCComputer의 조성돈 형과 서울에 계신 신경희 형, 그리고 OepnGL 동호회의 운영진 분들에게 고마움과 언제나 건강하시길 바랍니다.

죽마고우인 휘철이와 성국이, 언제나 물심 양면으로 저를 돌봐주신 저의 어머니와 형님 그리고 멀리 떨어져 있지만 항상 나를 걱정해준 사랑하는 미라에게도 고마운 마음을 전합니다.

끝으로 이 글을 읽지는 못하지만 저를 태어나게 해주신 아버님께 이 논문을 바칩니다.

2004년 1월 5일

우정재

