

이륜 독립진자 로봇의 모델 기반 제어 시스템 설계

具大官*, 池俊根†, 車貴守**

Model Based Control System Design of Two Wheeled Inverted Pendulum Robot

Dae-Kwan Ku, Jun-Keun Ji, and Guee-Soo Cha

요 약

본 논문에서는 LEGO NXT 마인드스톰을 사용하여 이륜 독립진자 로봇을 제작하고 MATLAB/SIMULINK를 이용한 모델 기반 설계 방법으로 제어기를 구현한 임베디드 시스템을 제안하였다. 먼저 이륜 독립진자 로봇의 모델링과 시뮬레이션을 통해 제어기의 성능과 안정성을 확인한 다음 시뮬레이션 프로그램을 직접 코드 변환하여 실험으로 설계된 제어기의 유효성을 확인하였다. 이륜 독립진자 로봇은 독립진자 원리를 이용한 자율 자세 제어를 기본 기능으로 하고 목적지까지 주행이 가능한 로봇이다. 제어기는 2차의 최적제어 방법으로 설계된 상태궤환 제어기를 이용하였다. 2차의 최적제어는 성능 지수 함수 J 를 최소화 하는 이득행렬 K 를 구해 상태궤환 제어기의 이득으로 쓰는 방법으로 이득 설정 방법이 용이하여 이륜 독립진자 로봇의 제어기로 사용할 수 있다. 이륜 독립진자 로봇의 제어기를 설계한 후 실험으로 자세 제어 성능과 외란 응답성, 무선 조종 주행 성능, 라인 주행 및 장애물 회피 성능을 확인하였다.

ABSTRACT

This paper proposes embedded System of two wheeled inverted pendulum robot designed by model based design method, using MATLAB/SIMULINK and LEGO NXT Mindstorms. At first, stability and performance of controller is verified through modeling and simulation. After that direct conversion from simulation model to C code is carried and effectiveness of controller is experimentally verified. Two wheeled inverted pendulum robot has basic function about autonomous balancing control using principle of inverted pedulum and it is also possible to arrive at destination. In this paper, state feedback controller designed by quadratic optimal control method is used. And quadratic optimal control uses state feedback control gain K to minimize performance index function J . Because it is easy to find gain, this control method can be used in the controller of two wheeled inverted pendulum robot. This proposed robot system is experimentally verified with following performances - balancing control, disturbance rejection, remote control, line following and obstacle avoidance.

Key Words : Two wheeled inverted pendulum, Model based control system design, MATLAB/SIMULINK, Quadratic optimal control

1. 서 론

†교신저자 : 정희원, 순천향대 전기통신공학과 교수

E-mail : jkji@sch.ac.kr

*학생회원, 순천향대 전기통신시스템공학과 석사과정

**정회원, 순천향대 전기통신공학과 교수

접수일자 : 2010. 9. 15

1차 심사 : 2010. 11. 11

2차 심사 : 2011. 1. 4

심사완료 : 2011. 1. 11

도립진자는 중력에 의해 항상 떨어지는 진자를 수직으로 세워 지면에 대한 수직 면과 진자가 이루는 경사각 (Pitch)을 중력과 외란에 대해 항상 0도로 제어하는 시스템으로 불안정한 시스템이고 제어기의 성능을 평

가하기 위한 대상으로 많이 다루어졌으며, 2축 보행 로봇이나 미사일의 제어 문제에 많이 응용되었다. 도립진자 시스템은 크게 수레(Cart)형, 원운동(Rotary)형으로 나눌 수 있으며 수레형은 직선 운동으로, 원운동형은 원운동으로 진자를 제어 한다.^[1,2]

이론 도립진자 로봇은 도립진자 몸체가 넘어지는 방향에 직각인 횡방향 양 옆에 전동기로 구동 되는 바퀴를 달고 전동기를 제어하여 몸체의 자세 제어와 동시에 로봇을 원하는 위치로 이동 시킬 수 있는 로봇을 말한다. 미국 자동차 공학회(SAE)의 차량 고정 좌표계(Vehicle Fixed Coordinate System)^[3]에 따르면 x방향의 안정성이 바퀴 2개로 보장되고, y방향은 도립진자의 성질로 인해 불안정한 시스템이 된다. 이론 도립진자는 일반적인 도립진자와 달리 몸체의 경사각을 측정할 수 있는 자이로스코프(Gyroscope) 센서나 경사도(Inclinometer) 센서 또는 가속도 센서를 이용해 관성 측정 장치(Inertial Measurement Unit, IMU)를 구성하여 경사각 측정에 이용한다.^[1,2,4]

기존의 연구로서, 해외에서는 극점 배치 기법을 이용한 상태 궤환 제어기^[10], 극점 배치와 상태 관측기를 이용한 상태 궤환 제어기^[11], LQ(Linear Quadratic) 제어 이론을 이용한 제어기가 연구되었다^[13,16]. 국내에서는 LQ 제어 이론과 외란 관측기를 이용하여 사람이 직접 탈 수 있는 이론 도립진자를 개발하였고^[1], LQ제어 이론을 이용한 시뮬레이션^[14], 단일 가속도 센서 이용시 디지털 필터와 확장 칼만 필터 및 PID 제어기를 이용한 제어기^[4], PID 제어기를 이용한 자세 제어^[12], 퍼지 이론 같은 고급 제어 이론을 이용한 연구 결과도 나와 있다^[15].

본 논문은 LEGO Mindstorms NXT와 MATLAB/SIMULINK 프로그램 개발환경에서 모델 기반 설계(Model Based Design) 기법을 이용한 2차 최적 제어 방법으로 자세 제어기를 설계하여 시뮬레이션 및 실험으로 자세 제어 성능을 확인하였다.

2. 전체 시스템의 구성

2.1 하드웨어 구성

2.1.1 NXT 마인드스톰 제어기

NXT는 32비트 ARM7 마이크로프로세서(256 Kbytes FLASH, 64 Kbytes RAM)를 장착한 소형 컨트롤러이다. 32비트 CPU는 기본 디지털 입출력, 인터럽트, 디스플레이, USB 2.0 및 블루투스 통신 등과 같은 제어 로직을 실행한다. 또한 NXT에 함께 내장된 8

비트 AVR 마이크로프로세서(4 Kbytes FLASH, 512 Byte RAM)는 A/D 변환, PWM 신호발생 및 버튼 조작 등의 기능을 실행한다. 각 프로세서에는 플래쉬(Flash) 메모리와 RAM이 있으며, 프로세서 간에는 I2C 방식으로 통신한다.^[5,6]

2.1.2 NXT 마인드스톰 제어기

NXT는 4개의 센서 입력포트를 갖고 있어 NXT 마인드스톰용으로 나온 터치 센서, 광 센서, 사운드 센서 및 초음파 센서를 연결할 수 있다. 이 센서들을 이용해 측정 데이터를 궤환시켜 프로세서가 피드백 제어 프로그램을 실행시킬 수 있도록 만들 수 있다. 본 논문에서는 자세 제어를 위한 자이로스코프 센서, 라인 주행을 위한 광센서, 전방 장애물 탐지용 초음파 센서를 이용한다.^[5,6]

2.1.3 출력포트

NXT는 상단부에 3개의 디지털 출력포트(A, B, C)를 갖고 있으며, 각각 독립적으로 전동기 구동이 가능하다. "ON" 상태에서는 모터의 "정방향" 또는 "역방향" 구동이 가능하며 "OFF" 상태에서는 모터를 강제로 브레이크 걸거나, 혹은 모터가 자연 감속해서 정지하도록 할 수 있다. 이 포트들은 기본적으로 PWM 전압 변조기법을 통해서 전동기를 구동한다.^[5,6]

2.1.4 직류 전동기

마인드스톰의 서보전동기는 전력효율이 매우 좋으며, 크기가 작은 부하에 대해서는 전동기 자체의 플라이휠 효과가 전동기 전력효율을 좋게 한다. 바퀴 2개를 구동시키기 위해 직류 전동기 2개를 사용한다. 전동기에는 광학식 증분형 엔코더(Optical Incremental Encoder)가 기본으로 장착되어 있다.^[5,6]

2.1.5 이론 도립진자 로봇의 외형

그림 1은 제작된 이론 도립진자 로봇의 전체 모습을 보여준다. 로봇의 각 부분을 기능으로 나누면 크게 제어부, 센서부, 구동부로 나눌 수 있다. 제어부는 NXT 마인드스톰의 ARM7 & AVR 프로세서를 장착한 임베디드 시스템으로 다운로드한 제어 프로그램을 실행하는 역할을 수행한다. 센서부는 제어기가 제어를 하기 위해 필요한 물리량들을 측정하며, 전동기의 엔코더, 몸체의 경사각을 측정용 자이로스코프, 전방 장애물을 탐지하기 위한 초음파 센서, 라인 주행을 위한 광센서를 사용한다.

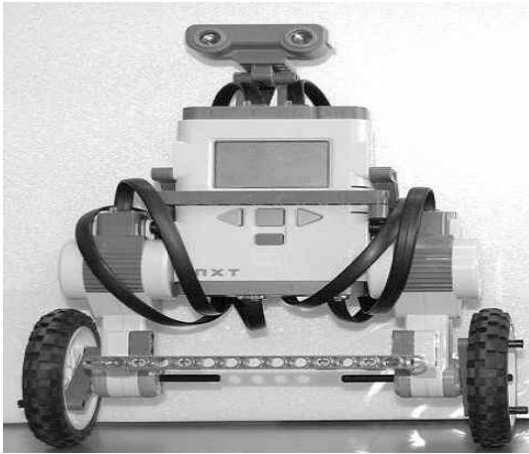


그림 1 이론 도립진자 로봇
Fig. 1 Two wheeled inverted pendulum robot

2.2 소프트웨어 구성

이론 도립진자 로봇을 제어하기 위해 제어프로그램 설계가 필요하다. 전통적인 임베디드 시스템의 제어 프로그램은 C언어를 이용한 문서기반 설계기법을 통해 개발되지만, 본 논문에서는 MATLAB/SIMULINK를 이용한 모델 기반 설계기법으로 제어프로그램을 개발한다. MATLAB/SIMULINK를 이용한 모델 기반 설계기법으로 NXT 마인드스툼 제어프로그램을 설계하기 위해서는 nxtOSEK와 Embedded Coder Robot NXT 소프트웨어가 필요하다^[7]. nxtOSEK는 레고 마인드스툼 NXT를 위한 오픈소스 플랫폼으로 leJOS NXJ 장치 드라이버, TOPPERS/ATK(TOPPERS/OSEK)와 TOPPERS/JSP 실시간 운영체제로 구성되어 있다. Embedded Coder Robot NXT는 nxtOSEK 운영체제하에서 실행될 제어 프로그램을 SIMULINK를 이용한 모델 기반 설계기법으로 설계할 수 있게 해주는 개발 환경이다. 그림 2는 MATLAB/SIMULINK 개발환경과 C코드 생성결과를 보여준다.

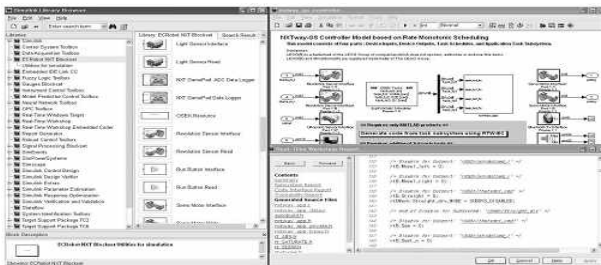


그림 2 MATLAB/SIMULINK 개발환경
Fig. 2 MATLAB/SIMULINK IDE

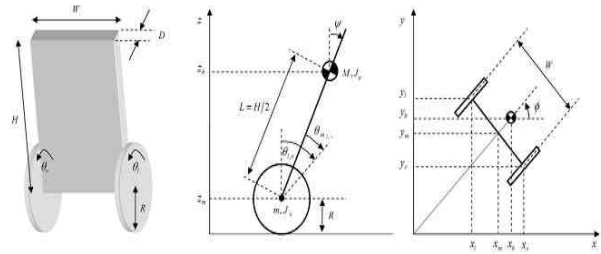


그림 3 이론 도립진자 자유 물체도
Fig. 3 Free body diagram of two wheeled inverted pendulum

표 3.1 이론 도립진자 로봇의 변수

Table 3.1 Variable of two wheeled inverted pendulum

변수	단위	설명
z_b	[m]	몸체 질량 중심의 높이
$\theta_{l,r}$	[rad]	좌우 바퀴의 회전각
$\theta_{ml,r}$	[rad]	직류 전동기의 회전각
x_l, y_l, z_l	[m]	좌측 바퀴의 위치
x_r, y_r, z_r	[m]	우측 바퀴의 위치
x_m, y_m, z_m	[m]	좌우 바퀴를 연결하는 축 중심의 위치
x_b, y_b, z_b	[m]	몸체 질량 중심의 위치
ψ	[rad]	지면에 수직인 면과 이루는 몸체 경사각
ϕ	[rad]	로봇이 조향할 경우 조향각

표 3.2 이론 도립진자 로봇의 상수

Table 3.2 Parameter of two wheeled inverted pendulum robot

상수	단위	설명
$g = 9.8$	[m/sec ²]	중력가속도
$m = 0.03$	[kg]	바퀴 무게
$R = 0.04 = z_m$	[m]	바퀴반경, 바퀴중심의 높이
$J_w = mR^2/2$	[kgm ²]	바퀴 관성모멘트
$M = 0.6$	[kg]	몸체 무게
$W = 0.14$	[m]	몸체 길이
$D = 0.04$	[m]	몸체 폭
$H = 0.144$	[m]	몸체 높이
$L = H/2$	[m]	질량중심과 바퀴 회전축 사이 거리
$J_\psi = ML^2/3$	[kgm ²]	몸체 경사각(ψ) 관성모멘트
$J_\phi = M(W^2 + D^2)/2$	[kgm ²]	몸체 조향각(ϕ) 관성모멘트
$J_m = 1 \times 10^{-5}$	[kgm ²]	직류 전동기 관성모멘트
$R_a = 6.69$	[Ω]	직류 전동기 저항
$K_b = 0.468$	[Vsec/rad]	직류 전동기 역기전력 상수
$K_t = 0.317$	[Nm/A]	직류 전동기 토크 상수
$n = 1$		기어비
$f_m = 0.0022$		몸체와 모터사이 마찰계수
$f_w = 0$		바퀴와 지면사이 마찰계수

3. 이론 독립진자 로봇의 동역학 모델

이론 독립진자 로봇의 운동방정식을 유도하고 상태 방정식을 세우기 위해서 다음 그림 3과 같은 자유 물체도(Free body diagram)를 이용한다^[8].

자유 물체도의 변수와 상수들을 다음 표 3.1과 표 3.2에 정리하였다.

본 논문에서는 그림 3의 자유 물체도 좌표계에 따라 라그랑지안 방법을 이용하여 로봇의 운동 방정식을 유도할 것이다. 라그랑지안 방법은 운동 방정식을 세울 대상 시스템에 대해 일반화된 좌표계를 정의하고 대상의 운동 에너지, 위치 에너지등의 에너지 관계식으로부터 운동 방정식을 유도하기 때문에 작용하는 힘을 일일이 계산해야 하는 뉴턴 방법과 달리 계산적인 측면에서 간단하다. 일반화 좌표는 시스템의 운동을 나타낼 수 있는 서로 독립된 좌표 변수를 말한다. 라그랑지안 방법은 다음 식 (1)을 이용한다.

$$L = T - V$$

$$\frac{d}{dt} \frac{\partial}{\partial \dot{q}_j} T - \frac{\partial}{\partial q_j} T = Q_j, \quad j = 1, 2, \dots, s \quad (1)$$

T , V 는 각각 대상의 운동 에너지, 위치 에너지이고, L 은 라그랑지안이라고 한다. 이 라그랑지안 L 에 대해 일반화 좌표라고 하는 q_j 와 시간 t 에 대해 편미분 연산을 하면 일반화 힘 Q_j 를 얻을 수 있다. 일반화 힘은 대상의 운동 상태를 기술한다.

이론 독립진자 로봇의 운동에너지는 병진 운동에너지와 회전 운동에너지로 구성 된다. 병진 운동에너지는 로봇이 시간에 따라 지면에서 변위를 가지는 운동을 할 때 생긴다. 이 에너지는 다음 식 (2)와 같이 좌우 바퀴, 몸체의 병진 운동에너지로 구성된다.

$$T_1 = \frac{1}{2} m (\dot{x}_l^2 + \dot{y}_l^2 + \dot{z}_l^2) + \frac{1}{2} m (\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2)$$

$$+ \frac{1}{2} M (\dot{x}_b^2 + \dot{y}_b^2 + \dot{z}_b^2) \quad (2)$$

이론 독립진자는 회전하는 물체를 모두 5개 가지고 있다. 전동기 2개, 바퀴 2개와 몸체이다. 그러므로 회전 운동에너지는 다음 식 (3)과 같다.

$$T_{wheel} = \frac{1}{2} J_w \dot{\theta}_l^2 + \frac{1}{2} J_w \dot{\theta}_r^2, \quad T_{body} = \frac{1}{2} J_\psi \dot{\psi}^2 + \frac{1}{2} J_\phi \dot{\phi}^2$$

$$T_{motor} = \frac{1}{2} n^2 J_m (\dot{\theta}_l - \dot{\psi})^2 + \frac{1}{2} n^2 J_m (\dot{\theta}_r - \dot{\psi})^2 \quad (3)$$

$$T_2 = T_{wheel} + T_{body} + T_{motor}$$

다음으로 로봇의 위치에너지를 구할 수 있다. 바퀴와 몸체의 위치에너지 및 전체 위치에너지는 다음 식 (4)와 같다.

$$V_{wheel} = mg(z_r + z_l)$$

$$V_{body} = Mg z_b \quad (4)$$

$$V = V_{wheel} + V_{body}$$

로봇의 운동에너지와 위치에너지를 구하였으므로 로봇의 운동을 완전히 기술할 수 있는 일반화 좌표를 정해 일반화 힘을 구할 수 있다. 본 논문에서 정한 일반화좌표는 다음과 같다.

$$\theta, \psi, \phi \quad (5)$$

라그랑지안 방정식은 다음 식 (6)과 같다.

$$L = T_1 + T_2 - V$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = F_\theta, \quad \frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\psi}} \right) - \frac{\partial L}{\partial \psi} = F_\psi \quad (6)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi} = F_\phi$$

여기서, F_θ, F_ψ, F_ϕ 는 일반화 힘이며 계산 결과는 각각 다음 식 (7)과 같다.

$$F_\theta = [(2m + M)R^2 + 2J_w + 2n^2 J_m] \ddot{\theta}$$

$$+ (MLR \cos \psi - 2n^2 J_m) \ddot{\psi} - MLR \dot{\psi}^2 \sin \psi$$

$$F_\psi = (MLR \cos \psi - 2n^2 J_m) \ddot{\theta} + (ML^2 + J_\psi + 2n^2 J_m) \ddot{\psi}$$

$$- MgL \sin \psi - ML^2 \dot{\phi}^2 \sin \psi \cos \psi \quad (7)$$

$$F_\phi = \left[\frac{1}{2} m W^2 + J_\phi + \frac{W^2}{2R^2} (J_w + n^2 J_m) \right]$$

$$+ ML^2 \sin^2 \psi \ddot{\phi} + 2ML^2 \dot{\psi} \dot{\phi} \sin \psi \cos \psi$$

이론 독립진자 로봇의 제어목표는 몸체 경사각 ψ 를 항상 0° 로 수렴시키는 것이다. 이러한 특징을 이용해서 비선형 미분방정식의 선형화가 가능하다. 그러므로 $\lim_{\psi \rightarrow 0} : \sin \psi \rightarrow \psi, \cos \psi \rightarrow 1, \dot{\psi}^2 \rightarrow 0$ 이 되고, 다음과 같은 식 (8)로 정리된다.

$$F_\theta = [(2m + M)R^2 + 2J_w + 2n^2 J_m] \ddot{\theta} + (MLR - 2n^2 J_m) \ddot{\psi}$$

$$F_\psi = (MLR - 2n^2 J_m) \ddot{\theta} + (ML^2 + J_\psi + 2n^2 J_m) \ddot{\psi} - MgL \psi \quad (8)$$

$$F_\phi = \left[\frac{1}{2} m W^2 + J_\phi + \frac{W^2}{2R^2} (J_w + n^2 J_m) \right] \ddot{\phi}$$

지금까지 구한 일반화 힘 3개를 다음 식 (9)와 같이 좌우 바퀴의 힘으로 기술할 수 있다.

$$F_\theta = F_l + F_r, \quad F_\psi = F_\psi, \quad F_\phi = \frac{W}{2R}(F_r - F_l) \quad (9)$$

또한 좌우 바퀴의 힘은 다음 식 (10)과 같이 전동기의 발생 토크, 몸체와 전동기의 상호 작용, 지면 마찰력으로 표현 할 수 있다.

$$\begin{aligned} F_l &= nK_t i_l - f_m(\dot{\psi} - \dot{\theta}_l) - f_w \dot{\theta}_l \\ F_r &= nK_t i_r - f_m(\dot{\psi} - \dot{\theta}_r) - f_w \dot{\theta}_r \end{aligned} \quad (10)$$

위 식에서 $i_{l,r}$ 은 직류 전동기 전기자 전류이다. 경사각 방향힘 F_ψ 는 F_θ 에서 지면 마찰력이 0이고 반대 방향인 것과 같다.

$$F_\psi = -F_\theta|_{f_w=0} = -nK_t(i_l + i_r) - f_m[2\dot{\psi} - (\dot{\theta}_l + \dot{\theta}_r)] \quad (11)$$

좌우 전동기에 로봇 몸체가 연결된 직류 전동기 동역학 모델은 다음과 같다.

$$\begin{aligned} L_a \dot{i}_{l,r} &= V_{l,r} - R_a i_{l,r} - K_b(\dot{\psi} - \dot{\theta}_{l,r}) \\ K_b &= k_e \phi_f = K_e \lambda_f \left[\frac{V_{sec}}{Rad} \right] \end{aligned} \quad (12)$$

여기서 L_a 는 작기 때문에 해석상의 편의를 위해 0으로 무시하고 전기자 전류에 대해 정리하면 다음과 같다.

$$i_{l,r} = \frac{V_{l,r} - K_b(\dot{\psi} - \dot{\theta}_{l,r})}{R_a} \quad (13)$$

이 전기자 전류 식을 식 (9)에 대입하면 다음과 같이 정리 된다.

$$\begin{aligned} F_\theta &= \alpha(v_l + v_r) + 2\dot{\psi}\beta - 2(\beta + f_w)\dot{\theta} \\ F_\psi &= -\alpha(v_l + v_r) + 2\beta\dot{\theta} - 2\beta\dot{\psi} \\ F_\phi &= \frac{W}{2R}\alpha(v_r - v_l) - \frac{W^2}{2R^2}(\beta + f_m)\dot{\phi} \end{aligned} \quad (14)$$

이 식에서 α, β, θ 는 다음과 같다.

$$\alpha = \frac{nK_t}{R_a}, \quad \beta = \frac{nK_t K_b}{R_a} + f_m, \quad \theta = \frac{\theta_l + \theta_r}{2} \quad (15)$$

로봇의 일반화 힘을 라그랑지안 방법과 전동기 동역

학 모델로 구했으며, 두 결과에는 다음 식 (16)과 같은 항등관계가 성립한다.

$$\begin{aligned} \alpha(v_l + v_r) &= [(2m + M)R^2 + 2J_w + 2n^2 J_m] \ddot{\theta} \\ &\quad + (MLR - 2n^2 J_m) \dot{\psi} + 2(\beta + f_w) \dot{\theta} - 2\dot{\psi}\beta \\ -\alpha(v_l + v_r) &= (MLR - 2n^2 J_m) \dot{\theta} + (ML^2 + J_\psi + 2n^2 J_m) \ddot{\psi} \\ &\quad - MgL\dot{\psi} - 2\beta\dot{\theta} + 2\beta\dot{\psi} \\ \frac{W}{2R}\alpha(v_r - v_l) &= \left[\frac{1}{2}mW^2 + J_\phi + \frac{W^2}{2R^2}(J_w + n^2 J_m) \right] \ddot{\phi} \\ &\quad + \frac{W^2}{2R^2}(\beta + f_m)\dot{\phi} \end{aligned} \quad (16)$$

이 관계를 이용해 상태 방정식을 세울 수 있는데, F_θ, F_ψ 에 관한 식들을 다음과 같이 식 (17)의 행렬로 표현할 수 있다.

$$H \begin{bmatrix} v_l \\ v_r \end{bmatrix} = E \begin{bmatrix} \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} + F \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix} + G \begin{bmatrix} \theta \\ \psi \end{bmatrix} \quad (17)$$

계수 행렬은 다음 식 (18)과 같다.

$$\begin{aligned} H &= \begin{bmatrix} \alpha & \alpha \\ -\alpha & -\alpha \end{bmatrix}, \quad F = 2 \begin{bmatrix} \beta + f_w & -\beta \\ -\beta & \beta \end{bmatrix}, \quad G = \begin{bmatrix} 0 & 0 \\ 0 & -MgL \end{bmatrix} \\ E &= \begin{bmatrix} (2m + M)R^2 + 2J_w + 2n^2 J_m & MLR - 2n^2 J_m \\ MLR - 2n^2 J_m & ML^2 + J_\psi + 2n^2 J_m \end{bmatrix} \end{aligned} \quad (18)$$

또한 F_ϕ 는 다음과 같이 식 (19)로 나타낼 수 있다.

$$K(v_r - v_l) = I\ddot{\phi} + J\dot{\phi} \quad (19)$$

여기서, 계수 K, I, J 는 다음 식 (20)과 같다.

$$\begin{aligned} K &= \frac{W}{2R}\alpha, \quad I = \frac{1}{2}mW^2 + J_\phi + \frac{W^2}{2R^2}(J_w + n^2 J_m) \\ J &= \frac{W^2}{2R^2}(\beta + f_m) \end{aligned} \quad (20)$$

이제 식 (21)과 같이 다음 변수들에 따라 상태방정식을 만들 수 있다.

$$x_1 = [\theta, \psi, \dot{\theta}, \dot{\psi}]^T, \quad x_2 = [\phi, \dot{\phi}]^T, \quad u = [v_l, v_r]^T \quad (21)$$

상태방정식을 세우면 다음 식 (22)와 같다.

$$\begin{aligned} \dot{x}_1 &= A_1 x_1 + B_1 u \\ \dot{x}_2 &= A_2 x_2 + B_2 u \end{aligned} \quad (22)$$

먼저 상태변수방정식 \dot{x}_1 을 구할 수 있다.

$$\begin{bmatrix} \dot{\theta} \\ \dot{\psi} \\ \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ A_1(3,1) & A_1(3,2) & A_1(3,3) & A_1(3,4) \\ A_1(4,1) & A_1(4,2) & A_1(4,3) & A_1(4,4) \end{bmatrix} \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ B_1(3,1) & B_1(3,2) \\ B_1(4,1) & B_1(4,2) \end{bmatrix} \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (23)$$

식 (17)을 $\ddot{\theta}, \ddot{\psi}$ 에 대해서 정리하면,

$$E \begin{bmatrix} \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = -G \begin{bmatrix} \theta \\ \psi \end{bmatrix} - F \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix} + H \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (24)$$

식 (24)에 $E^{-1} = \begin{bmatrix} a & b \\ c & d \end{bmatrix}$ 를 양변에 곱하면

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = -E^{-1}G \begin{bmatrix} \theta \\ \psi \end{bmatrix} - E^{-1}F \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix} + E^{-1}H \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (25)$$

그러므로 아래의 식 (26)을 계산하면 미지수 A_1, B_1 을 알 수 있다.

$$\begin{bmatrix} \ddot{\theta} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & bMgl \\ 0 & dMgl \end{bmatrix} \begin{bmatrix} \theta \\ \psi \end{bmatrix} + \begin{bmatrix} a\alpha - b\alpha & a\alpha - b\alpha \\ c\alpha - d\alpha & c\alpha - d\alpha \end{bmatrix} \begin{bmatrix} v_l \\ v_r \end{bmatrix} + E^{-1}H \begin{bmatrix} \dot{\theta} \\ \dot{\psi} \end{bmatrix} \quad (26)$$

이제 상태변수방정식 \dot{x}_2 를 구하면 다음 식 (27)과 같이 나타낼 수 있다.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ A_2(2,1) & A_2(2,2) \end{bmatrix} \begin{bmatrix} \phi \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ B_2(2,1) & B_2(2,2) \end{bmatrix} \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (27)$$

여기서 식 (19)을 이용하면, 다음 식 (28)을 구할 수 있다.

$$\ddot{\phi} = \frac{J}{I}\dot{\phi} + \frac{K}{I}(v_r - v_l) \quad (28)$$

식 (27)과 (28)을 정리하면 다음 식 (29)와 같다.

$$\begin{bmatrix} \dot{\phi} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & J \\ 0 & I \end{bmatrix} \begin{bmatrix} \phi \\ \phi \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ -\frac{K}{I} & \frac{K}{I} \end{bmatrix} \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (29)$$

위에서 구한 상태방정식에서, 식 (23)은 이론 도립진자 로봇의 자세 제어 운동을 기술하고 있으며, 식 (29)는 로봇의 조향운동을 기술한다. 여기서 식 (23)과 (29)의 상태방정식을 통합하고 표 3.2의 상수들을 이용해 계산하면 다음 식 (30)과 같다.

$$\begin{bmatrix} \dot{\theta} \\ \dot{\psi} \\ \ddot{\theta} \\ \ddot{\psi} \\ \ddot{\phi} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & -409.7184 & -162.1273 & 162.1273 & 0 & 0 \\ 0 & 269.6273 & 78.1496 & -78.1496 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -95.5684 \end{bmatrix} \begin{bmatrix} \theta \\ \psi \\ \dot{\theta} \\ \dot{\psi} \\ \phi \\ \phi \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 157.5798 & 157.5798 \\ -75.9576 & -75.9576 \\ 0 & 0 \\ -53.0787 & 53.0787 \end{bmatrix} \begin{bmatrix} v_l \\ v_r \end{bmatrix} \quad (30)$$

제어기를 설계하기 전에 상태변수 방정식에서 식 (23)을 이용하여 바퀴 각도 외물체 경사각에 대해 MATLAB의 `ctrb`, `obsv` 명령어로 가제어성 (Controllable)과 가관측성(Observable)을 알아본 결과 Rank가 4로써 제어가 가능하고 관측이 가능함을 알 수 있다. 한편 식 (29) 대해서는 Rank가 2로써 제어가 가능하나, Rank가 1로써 관측이 불가능하므로 조향 각 가속도는 관측이 불가능함을 알 수 있다.

4. 제어 시스템 설계

4.1 제어기 설계

본 논문의 이론 도립진자 로봇은 좌우 직류 전동기에 엔코더, 몸체에 자이로스코프 센서가 설치되어 있고 앞 장에서 제어 대상의 가제어성을 확인했기 때문에 4개 상태변수 $[\theta \ \psi \ \dot{\theta} \ \dot{\psi}]^T$ 를 모두 측정하여 케환시킬 수 있다. 따라서 로봇의 제어기로 상태케환 제어기가 유용하다고 볼 수 있다. 그림 4는 논문에서 적용한 상태케환 제어기의 블록도이다.

제어기 이득 K_1, K_2 는 2차 최적제어 방법으로 결정하였다. 이 방법은 다음 식 (31)을 최소화 하는 행렬 K 를 구해서 제어기 이득으로 쓰는 것이다.

$$J = \int_0^{\infty} (x^T Q x + u^T R u) dt \quad (31)$$

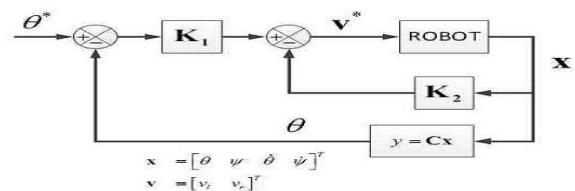


그림 4 상태케환 제어기 블록도
Fig. 4 Block diagram of state feedback controller

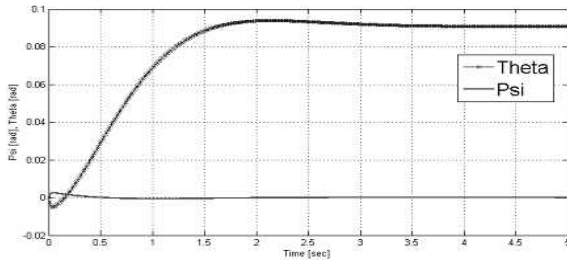


그림 5 θ, ψ 의 단위계단응답 시뮬레이션
Fig. 5 Time response of θ, ψ at $\psi = 0^\circ$

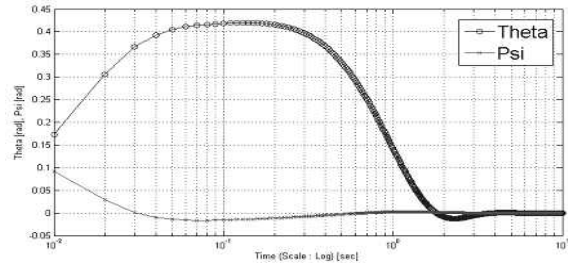


그림 6 θ, ψ 의 초기조건 시뮬레이션
Fig. 6 Time response of θ, ψ at $\psi = 10^\circ$

이 식의 J 는 성능지수(Performance Index)라고 한다. 식 (31)의 가중치 행렬 Q, R 을 정하면 이득 행렬 K 를 구할 수 있다. 이 방법의 장점은 이득을 체계적으로 계산할 수 있다는 점이다^[9]. 다음 식 (32)의 행렬 Q 와 R 을 시뮬레이션과 실험을 통해 시행착오로 정하고, $[\theta, \psi, \dot{\theta}, \dot{\psi}]^T$ 에 대한 상태변수 방정식 (23)의 행렬 A_1, B_1 을 이용하여 MATLAB의 lqr함수를 통해서 이득 행렬 K 를 결정했다.

$$Q = \begin{pmatrix} 10^2 & 0 & 0 \\ 0 & 10^5 & 0 \\ 0 & 0 & 10 \\ 0 & 0 & 0 \end{pmatrix}, R = \begin{pmatrix} 100 \\ 0 \end{pmatrix} \quad (32)$$

$$\begin{aligned} \text{Gain} &= -(0.9534, 34.2069, 0.7641, 1.7537) \\ K_1 = lqr_1 &= -0.9535 \\ K_2 = lqr_2 &= -(0, 34.2069, 0.7641, 1.7537) \end{aligned}$$

다음 그림 5는 이득 행렬 K 와 상태변수 방정식 행렬을 이용해 MATLAB의 step 명령어를 써서 θ, ψ 의 단위계단 응답 그래프를 나타낸 것이다. 초기치는 0이며 약 3.9초 부근에서 정상상태에 도달하였다.

그림 6은 $\psi = 10^\circ$ 를 초기 조건으로 설정하고 MATLAB의 initial 명령어로 10초까지 시뮬레이션을 한 결과이다. ψ 의 변화를 보기 쉽게 하기 위해 시간축 스케일을 로그로 설정하였다. 약 4.7초 부근에서 θ, ψ 가 정상상태에 도달하였다. 경사각 초기치가 있기 때문에 단위계단 응답 결과의 정상상태 도달 시간보다 길어졌다는 것 알 수 있다. 이륜 도립진자 로봇은 경사각 ψ 제어를 위해 전동기를 구동하기 때문에 위치 θ 의 변화가 있다는 것을 확인할 수 있다.

4.2 제어 프로그램

다음 그림 7은 이륜 도립진자 로봇 제어 프로그램의 전체적인 구성을 보여준다. 좌측 블록들은 입력 장치(Device Input)로 주로 물리량을 측정해 마이크로프로

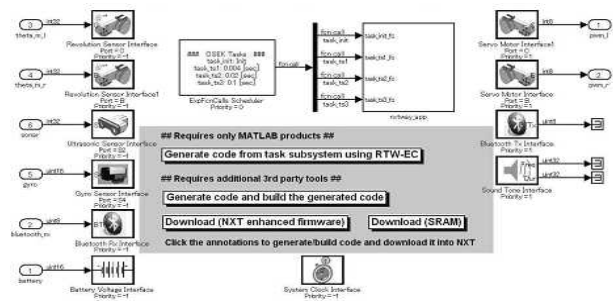


그림 7 제어 프로그램의 전체 구성
Fig. 7 The overall composition of the control program

세서로 보내주는 계측기들이다. 우측 블록들은 출력 장치(Device output)로서 전동기, 블루투스, 스피커 같은 것들이다. 본 논문에서 구현한 이륜 도립진자 로봇은 PC에 연결한 조이 스틱을 이용해 무선으로 주행 제어를 할 수 있으며, 또한 독립적인 라인 주행 운전을 할 수 있다. 두 경우 모두 자율적으로 몸체 경사각 제어를 하며 주행이 가능하다.

중앙 좌측의 Exported Function-Calls Scheduler는 제어주기를 정하고, 제어주기마다 호출해 수행해야 하는 서브시스템이 연결된 블록이다. 호출해야 하는 서브시스템은 중앙 우측에 nextway_app 블록안에 프로그램되어있다. 중앙 하단부에 있는 버튼들은 SIMULINK로 만든 제어 프로그램을 C코드로 변환해 NXT 마인드스톰으로 다운로드 하기 위한 것들이다.

그림 8은 제어 프로그램의 상태케환제어기 부분이다. Cal Reference 블록은 Make Command 블록에서 생성되는 $\dot{\theta}_{cmd}$ 지령값을 계산하고, Cal x1 블록은 자이로센서와 전동기 엔코더로부터 측정값을 받아서 케환시켜야 하는 상태변수를 계산한다.

그림 9는 상태변수 지령을 계산하는 블록이다. 바퀴 속도지령 $\dot{\theta}_{cmd}$ 의 급격한 변화를 막기 위해 디지털 저역통과 필터를 거쳐서 지령신호를 계산한다.

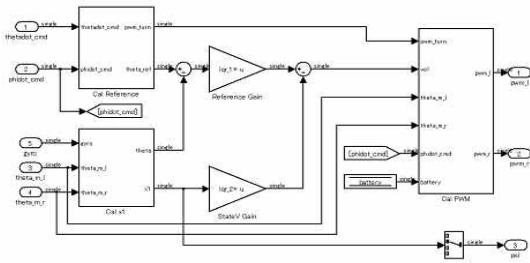


그림 8 상태궤환 LQR 제어기
Fig. 8 State feedback LQR Controller

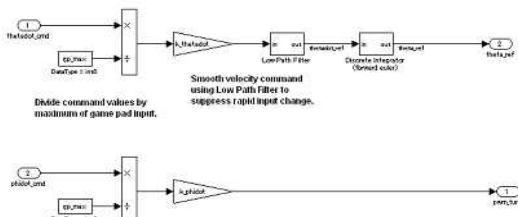


그림 9 지령 생성
Fig. 9 Reference generator

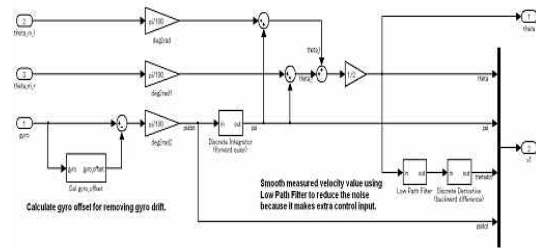


그림 10 현재 상태 변수 계산
Fig. 10 Calculate present state variable

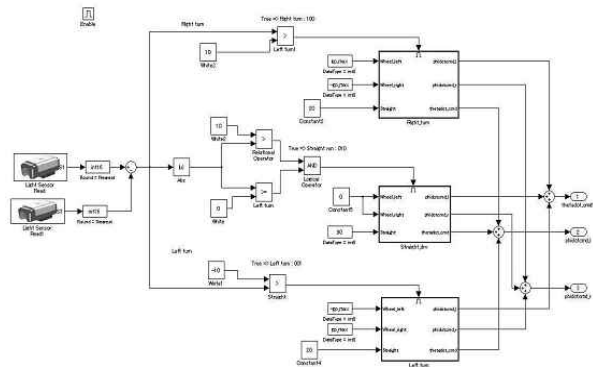


그림 11 라인트레이싱 - Line_drv 블록
Fig. 11 Line Tracing - Line_drv Block

그림 10은 로봇의 센서를 통한 계측 데이터를 바탕으로 현재 상태변수를 계산하는 블록이다. Gyro Calibration에서 자이로의 오프셋을 계산하였지만 주행 및 제어 프로그램 수행단계에서도 계속 오프셋이 발생하기 때문에 디지털 저역통과 필터로 오프셋을 계산해 내어 자이로 계측값에서 빼주어야 한다. 이 블록에서도 $\dot{\theta}$ 을 계산할 때 디지털 저역통과 필터를 사용해야 한다.

그림 11은 라인주행 알고리즘에 따라 설계한 Line_drv 블록 내부 프로그램이다. 여기서는 경계값을 10으로 설정하였다. 두 광센서 계측값 차이의 절대값이 10보다 작을 경우 직선라인으로 인식해 직선주행을 하게 되고 계측값 차이가 10보다 크고 -10보다 작을 경우 곡선라인으로 인식해 커브를 돌게 된다.

그림 12는 라인 주행중 장애물 탐지시 장애물을 회피하여 반대방향으로 주행하는 알고리즘을 나타낸다. 장애물 탐지 거리는 사용자가 설정할 수 있다. 장애물 회피 알고리즘은 그림 12와 13을 따른다. 초음파 센서로 30cm 이내의 장애물을 탐지하면 (그림 13-a), 정지 후 왼쪽 바퀴를 구동하여 그림 13-b의 상태로 만든 다음에 정지 후 다시 오른쪽 바퀴만 후진으로 구동하여 제자리에서 반대 방향으로 위치시킨 뒤에(그림 13-c) 라인 검사 후 반대 방향으로 주행을 한다. 그림 13의 각 단계별 상황 인식은 두 전동기의 엔코더 측정값 차이를 이용하였다.

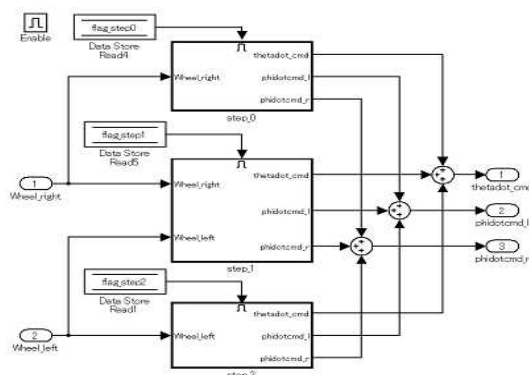


그림 12 장애물 탐지시 회피 프로그램
Fig. 12 Obstacle avoidance Program

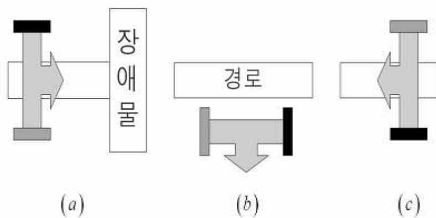


그림 13 장애물 탐지시 회피동작
Fig. 13. Motion of Obstacle avoidance

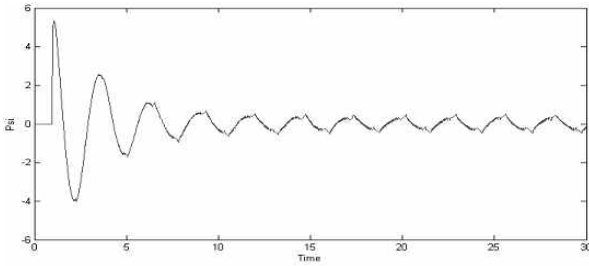


그림 14 몸체 경사각 제어 시뮬레이션 결과
Fig. 14 Simulation result of ψ

5. 시뮬레이션 및 실험 결과

5.1 시뮬레이션 결과

5.1.1 경사각 제어 시뮬레이션

실제 실험 전에 시뮬레이션 모델을 이용하여 시뮬레이션을 수행하였다. 그림 14는 시간에 대한 몸체 경사각의 시뮬레이션 결과를 보여준다. 30초 동안 시뮬레이션을 수행했으며 약 10초후에 정상상태에 도달하였다. 약 $0^\circ \pm 0.5^\circ$ 의 범위를 가지고 자세 제어를 수행한다.

5.1.2 외란 제어 시뮬레이션

다음 그림 15는 정지 상태 로봇에 경사각 방향의 외란을 주어 외란에 대한 응답 특성을 시뮬레이션한 결과이다. 100초의 시간동안 7번 외란을 주었고, 몸체 경사각 ψ 그래프의 화살표가 외란의 방향을 나타내는데 외란이 있어도 경사각 제어가 유지되는 것을 알 수 있다. 경사각 제어 시뮬레이션 결과와 같이 $0^\circ \pm 0.5^\circ$ 의 범위를 가지고 자세제어를 수행하며 로봇 위치는 외란에 대해 큰 변화가 나타나지 않았다.

5.2 실험 결과

5.2.1 경사각 제어 시뮬레이션

그림 16은 정지 상태 시 자세 제어 실험 결과이다. 그림 14의 시뮬레이션 결과와 달리 경사각 스파이크가 생긴 것을 볼 수 있으나 추이는 비슷하고 자세 제어가 가능하였다.

5.2.2 외란 시뮬레이션

그림 17은 로봇이 정지한 상태에서 로봇을 앞 또는 뒤로 밀어서 외란을 준 실험 결과이다. 약 100초의 시간동안 7번 외란을 주었고, 몸체 경사각 그래프의 화살표가 외란의 방향을 의미한다. 15~22초 영역에서는 그림 15의 시뮬레이션 결과와 달리 외란에 대해 위치

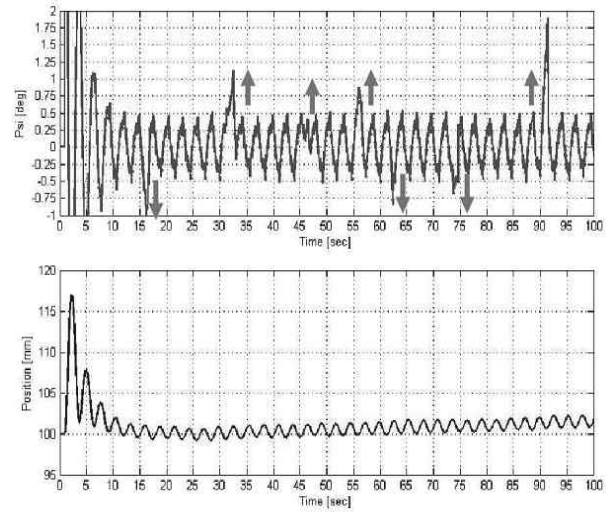


그림 15 외란에 대한 경사각과 위치 그래프(θ, ψ)
Fig. 15 Simulation result of disturbance response(θ, ψ)

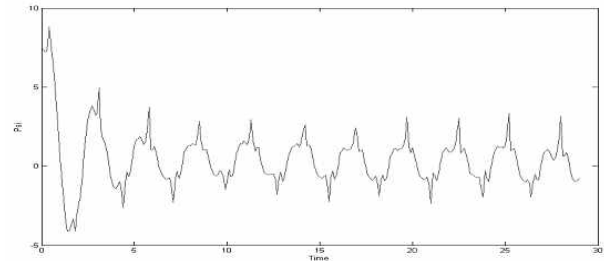


그림 16 정지 상태시 자세 제어
Fig. 16 Experimental result - balancing control

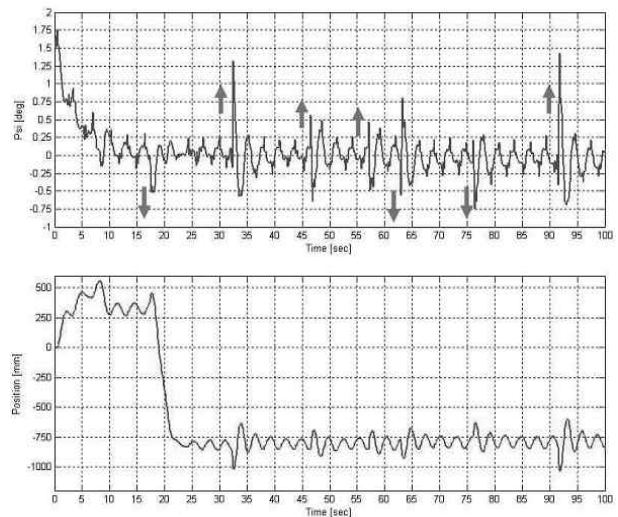


그림 17 외란에 대한 경사각 제어 실험(θ, ψ)
Fig. 17 Experimental result of disturbance response(θ, ψ)

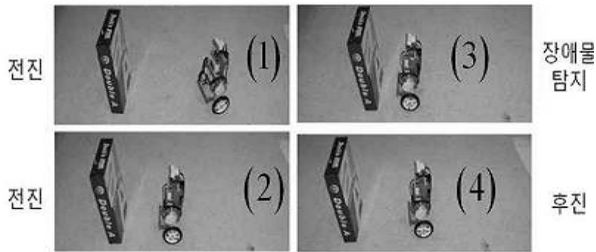


그림 18 장애물 탐지 및 회피

Fig. 18 Obstacle contact and avoidance

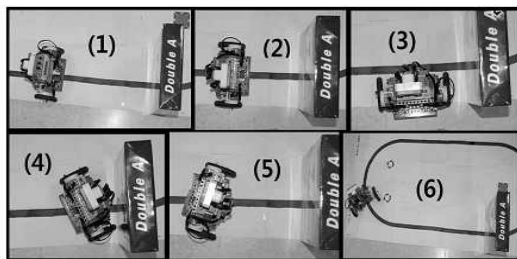


그림 19 라인주행 및 장애물 회피

Fig. 19 Line Tracing and obstacle avoidance

변화가 심했지만, 25초 이후로는 외란에 대한 위치 변화가 시뮬레이션 추이와 비슷한 것을 알 수 있다. 외란의 $d\psi/dt$ 가 클수록 위치 변화량이 크다.

5.2.3 경로 주행 및 장애물 회피 실험

그림 18은 조이스틱으로 무선조종을 하는 실험 결과를 보여준다. 초음파로 약 30cm 이내의 전방 장애물을 탐지할 수 있다. 장애물 탐지 시 조이스틱의 지령보다 장애물 회피가 우선하여 20ms 마다 수행되는 장애물 탐지 프로그램이 장애물 탐지를 못할 때까지 로봇은 후진한다. 또한 그림 19는 로봇이 독립적으로 라인주행 중 장애물을 탐지하면 장애물 회피 알고리즘대로 회피한 후 반대방향으로 진행하는 것을 볼 수 있다.

6. 결 론

본 논문에서는 LEGO NXT 마인드스톰을 사용하여 이론 도립진자 로봇 하드웨어를 제작하고, MATLAB/SIMULINK를 이용한 모델 기반 설계 방법으로 제어기를 설계하여 실제로 구현한 임베디드 시스템을 제안하였다. 먼저는 모델링을 통한 시뮬레이션을 수행함으로써 이론 도립진자 로봇 제어기의 안정성과 성능을 확인한 후, NXT 마이크로프로세서로 프로그램을 다운로드하고 실험을 수행하여 유도한 운동방정식을 토대

로 한 상태변수 방정식의 유효성 및 제어기의 안정성과 성능을 확인하였다. 본 논문에서 제어기로 사용한 2차 최적제어 상태회환제어기는 시스템 파라미터의 정확함이 보장되고 Q, R 이 결정되면 MATLAB을 이용하여 이득 설정이 용이한 특징이 있다. 이러한 최적제어 상태회환제어기를 통해 무선 조종, 장애물탐지 및 회피, 라인 주행 등 기존의 x, y 방향에서 안정성이 확보된 다른 로봇에서 구현되었던 기능이 이론 도립진자 로봇에서도 적용가능하다는 것을 알 수 있었다.

참 고 문 헌

- [1] 최동일, "2륜자기균형이동차의 개발 및 외란 추정기를 이용한 강인제어", 한국과학기술원 기계공학전공 석사학위논문, 2007.
- [2] 신영훈, "관성 센서를 이용한 두 바퀴 도립 진자 제어에 관한 연구", 울산대학교 대학원 전기전자정보시스템공학과 석사학위논문, 2004.
- [3] T. D. Gillespie, "Fundamentals of Vehicle Dynamics", SAE International, 1992.
- [4] 하현욱, 이장명, "단일 가속도 센서에 의한 모바일 역진자 제어", 제어 로봇 시스템 학회 논문지, 제16권 제5호, pp. 440-445, 2010.
- [5] 김수환, 장순호, "LEGO MINDSTORM을 이용한 로봇 구현", 순천향대학교 정보기술공학부 학사학위논문, 2008.
- [6] 홍선학, 송선미, "C언어로 즐기는 LEGO MINDSTORMS NXT", 이지테크, 2007.
- [7] Yori-hisa Yamamoto, "NXTway-GS Model-Based Design", 2009.
- [8] Takashi Chikamasa, Tomoki Fukuda, "Embedded Coder Robot NXT Modeling Tips", 2009.
- [9] 오가타 가즈히코, "현대제어공학", 제4판, 사이텍미디어, 2003.
- [10] F. Grasser, A. D'Arrigo, S. Colombi, A.C. Rufer, "JOE: a mobile, inverted pendulum", *IEEE Trans. Ind. Electron.*, Vol. 49, pp. 107-114, 2002.
- [11] N. Shiroma, O. Matsumoto, S. Kajita, K. Tani, "Cooperative behavior of a wheeled inverted pendulum for object transportation", *Intelligent Robots and Systems '96*, Vol. 2, pp. 396-401, 1996.
- [12] 윤재무, 이재경, 이장명, "모바일 역진자의 효율적 수평 유지 기법", 제어·자동차·시스템공학 논문지, 제13권, 제 7호, pp. 656-663, 2007.
- [13] Y Kim, S H Kim, and YK Kwak, "Improving driving ability for a two-wheeled inverted-pendulum-type autonomous vehicle", Proc. of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering, Vol. 220, No. 2, pp. 165-175, 2005.

- [14] 이세한, 박종근, "차륜형 역진자 자세 제어기 구현을 위한 시뮬레이션", 대한기계학회 춘추학술대회, pp. 191-196, 2006.
- [15] 엄기환, 이규윤, 이현관, 김주웅, "PD+I 제어 방식을 적용한 Two-wheel Balancing Mobile Robot", 전자공학회 논문지, 제45권 SC편, 제1호, pp. 1-8, 2008.
- [16] Seong Hee Jeong, T. Takahashi, "Wheeled inverted pendulum type assistant robot: inverted mobile, standing, and sitting motions", *Intelligent Robots and Systems*, pp. 1932-1937, 2007.

저 자 소 개



구대관(具大官)

1986년 2월 5일생. 2010년 순천향대 정보기술공학부 졸업. 현재 동 대학원 전기통신시스템공학과 석사과정.



지준근(池俊根)

1964년 8월 13일생. 1986년 서울대 전기공학과 졸업. 1988년 동대학원 전기공학과 졸업(석사). 1994년 동대학원 전기공학과 졸업(공박). 현재 순천향대 전기통신공학과 교수. 당학회 평의원 및 JPE 편집위원.



차귀수(車貴守)

1956년 7월 18일생. 1978년 서울대 공대 공업교육과 졸업. 1987년 동대학원 전기공학과 졸업(공박). 현재 순천향대 전기통신공학과 교수 및 Post B.K. 전기통신인력양성사업단장.