

논문 2017-54-7-15

NXT Mindstorm을 이용한 교육용 이륜 도립진자 로봇 제작

(Implementation of Educational Two-wheel Inverted Pendulum Robot using NXT Mindstorm)

정 보 환*

(Bo Hwan Jung[©])

요 약

이 논문에서는 모델 기반 설계에 근거한 이륜 도립진자 로봇의 제어이득을 설계하고 NXT 마인드스톰과 RobotC 언어를 이용하여 수업에 활용 가능한 로봇을 제작 및 실험한다. 이륜 도립진자 로봇은 NXT 마인드스톰, 서보 직류전동기, 자이로 센서, 가속도 센서로 구성된다. 직류전동기에 내장된 엔코더를 이용하여 바퀴의 회전각을 검출하며 이동평균을 이용하여 바퀴의 회전각속도를 계산한다. 자이로 센서는 몸체의 피치 각속도를 측정하며 가속도 센서는 몸체의 피치 각도를 측정한다. 자이로와 가속도의 센서 융합을 통해 몸체 각도를 계산한다. 제어기 이득 요소는 휠 각도, 휠 각속도, 몸체 피치 각도, 몸체 피치 각속도에 대한 가중치이다. 이들 제어이득의 변화에 따른 도립진자 로봇의 변화를 실험하며 유용성을 확인한다.

Abstract

In this paper, we propose a controller gain based on model based design and implement the two-wheel inverted pendulum type robot using NXT Lego and RobotC language. Two-wheel inverted pendulum robot consists of NXT mindstorm, servo DC motor with encoder, gyro sensor, and accelerometer sensor. We measurement wheel angle using built-in encoder and calculate wheel angle speed using moving average method. Gyro measures body angular velocity and accelerometer measures body pitch angle. We calculate body angle with complementary filter using gyro and accelerometer sensor. The control gain is a weighted value for wheel angle, wheel angular velocity, body pitch angle, and body pitch angular velocity, respectively. We experiment and observe the effect of two-wheel inverted pendulum with respect to change of control gains.

Keywords : Inverted Pendulum, NXT Mindstorm, Gyro, Accelerometer, Complementary filter, LQR

I. 서 론

도립진자 시스템은 제어대상으로 오랫동안 관심을 받아왔다. 뿐만 아니라 제어 설계 예로써 산업 표준의 하나가 되었다. 도립진자는 막 이륙하려는 우주 발사체, 미사일 등의 자세제어에 대한 모형이라 할 수 있다.

이것은 적절한 제어입력이 작용하지 않으면 항상 넘어지는 불안정한 성질을 가지고 있다. 이 기본적인 성질은 선형화와 선형제어에 대한 주요한 본보기가 된다^[1]. 제어시스템 설계는 종종 상용 개발 툴인 Matlab/Simulink, LabVIEW 등을 활용하여 모델기반 설계(Model Based Design) 방법으로 수행된다. 이를 위해서는 고가의 설계도구와 현대 제어이론을 요구한다.

본 논문에서 제작하는 도립진자 로봇의 제어기 설계는 교수자가 담당한다. 학습자는 표준 C언어와 가장 유사한 RobotC 언어를 이용하여 제어기 구현, 센서 데이터 처리 등을 구현함으로써 임베디드 시스템의 개념을 학습할 수 있다.

* 정회원, 용인송담대학교 정보통신과
(Department of Information and Communication, Yongin Songdam College)

© Corresponding Author (E-mail : bhjung@ysc.ac.kr)

Received ; February 21, 2017 Revised ; April 2, 2017

Accepted ; June 23, 2017

기존의 연구에서 NXT mindstorm을 이용한 이륜 도립진자 로봇 시스템은 LQR 제어 기법을 사용하고 있다^[2-5]. 여기에서도 Matlab/Simulink를 이용하여 LQR 제어 기법을 통해 상태변수 궤환 제어이득을 설계한다. 비선형 수학적 모델의 선형근사화로 인한 시스템 불확실성을 반영하여 실제의 제어기 이득은 튜닝을 통해 제어기 이득을 결정한다. 제어기 이득 요소는 휠 각도 (wheel angle), 휠 각속도(wheel angular velocity), 몸체 피치 각도(body pitch angle), 몸체 피치 각속도 (body pitch angular velocity)에 대한 가중치이다. 이들 제어이득의 변화에 따른 도립진자 로봇의 변화를 실험한다.

II. 시스템 구성



그림 1. 이륜 도립진자 로봇
Fig. 1. Two-Wheel Inverted Pendulum Robot.

2.1 NXT mindstorm 제어기

NXT는 LEGO사의 로봇 컨트롤러로 ARM7 프로세서 (256 Kbytes FLASH, 64 Kbytes RAM)를 채용한 32비트급 제어기이다. 또한 NXT에 내장된 AVR 마이크로 컨트롤러는 A/D 변환, PWM 신호 생성 및 각종 버튼 조작 기능을 담당한다.

2.2 입력포트와 센서

NXT는 4개의 센서 입력포트를 가지고 있다. 여기에서 사용되는 센서는 각속도를 측정하는 자이로 센서와 기울기 검출에 사용되는 가속도 센서이다. 이 센서들을 이용해 측정 데이터를 궤환시켜 제어기가 자세제어를 수행한다.

2.3 출력포트와 직류전동기

NXT는 PWM으로 모터를 구동하는 3개의 디지털 출

력포트를 가지고 있다. NXT의 직류전동기는 전력효율이 좋으며 크기가 작은 부하에 대해 효과적이다.

2.4 이륜 도립진자 로봇 시스템 모델링

이륜 도립진자 로봇은 그림 2와 같이 모델링 된다. 여기서 ψ 는 몸체 피치 각도, $\theta_{l,r}$ 는 왼쪽/오른쪽 휠 각도, $\theta_{m,l,r}$ 는 왼쪽/오른쪽 모터 각도를 각각 나타낸다. 표 1은 제작된 이륜 도립진자 로봇의 사양을 나타낸다. 여기서 고려하는 이륜 도립진자 로봇 시스템의 물리 상수들은 참고문헌^[2]와 동일하다. 다만, 휠 무게, 휠 반경, 몸체 무게, 몸체 넓이, 몸체 폭, 몸체 높이는 다르다.

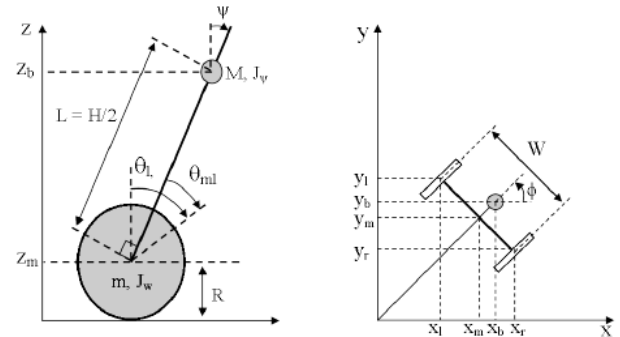


그림 2. 이륜 도립진자 로봇 시스템 모델링^[2]
Fig. 2. Two-Wheel Inverted Pendulum Robot Modelling.

표 1. 이륜 도립진자 로봇의 물리 상수
Table1. Physical constants of Two-Wheel Inverted Pendulum Robot.

parameter	unit	remark
$g = 9.81$	$[m/sec^2]$	중력가속도
$m = 0.017$	$[kg]$	휠 무게
$R = 0.026$	$[m]$	휠 반지름
$J_w = mR^2/2$	$[kgm^2]$	휠 관성모멘트
$M = 0.545$	$[kg]$	몸체 무게
$W = 0.17$	$[m]$	몸체 넓이
$D = 0.045$	$[m]$	몸체 폭
$H = 0.15$	$[m]$	몸체 높이
$L = H/2$	$[m]$	
$J_\psi = ML^2/3$	$[kgm^2]$	몸체 피치모멘트
$J_\phi = M(W^2 + D^2)$	$[kgm^2]$	몸체 요모멘트
$J_m = 1 \times 10^{-5}$	$[kgm^2]$	직류모터모멘트
$R_m = 6.69$	$[\Omega]$	직류모터저항
$K_b = 0.468$	$[Vsec/rad]$	직류모터역기전력
$K_i = 0.317$	$[Nm/A]$	직류모터토크상수
$n = 1$		기어비

2.5 이륜 도립진자 로봇 시스템의 운동방정식과 상태방정식

제어 대상 시스템의 운동방정식은 일반적으로 작용-반작용 원리를 이용한 뉴턴-오일러 방법과 에너지 관점에서 바라보는 오일러-라그랑지안 방법이 있다. 여기서 기술하는 내용은 오일러-라그랑지안 방법을 이용한 참고문헌 [2]에 근거한다. 시각 $t=0$ 에서 이륜 도립진자의 운동방향을 양의 x 축 방향이라 하면 각각의 좌표는 다음과 같다.

$$(\theta, \phi) = \left(\frac{1}{2}(\theta_l + \theta_r), \frac{R}{W}(\theta_r - \theta_l) \right) \quad (1)$$

$$(x_m, y_m, z_m) = \left(\int \dot{x}_m dt, \int \dot{y}_m dt, R \right) \quad (2)$$

$$(\dot{x}_m, \dot{y}_m) = (R\dot{\theta} \cos \phi, R\dot{\theta} \sin \phi) \quad (3)$$

$$(x_l, y_l, z_l) = \left(x_m - \frac{W}{2} \sin \phi, y_m + \frac{W}{2} \cos \phi, z_m \right) \quad (4)$$

$$(x_r, y_r, z_r) = \left(x_m + \frac{W}{2} \sin \phi, y_m - \frac{W}{2} \cos \phi, z_m \right) \quad (5)$$

$$(x_b, y_b, z_b) = \left(x_m + L \sin \psi \cos \phi, y_m + L \sin \psi \sin \phi, z_m + L \cos \psi \right) \quad (6)$$

병진 운동에너지 T_1 , 회전 운동에너지 T_2 , 위치 에너지 U 는 각각 다음과 같다.

$$T_1 = \frac{1}{2} m(\dot{x}_l^2 + \dot{y}_l^2 + \dot{z}_l^2) + \frac{1}{2} m(\dot{x}_r^2 + \dot{y}_r^2 + \dot{z}_r^2) + \frac{1}{2} M(\dot{x}_b^2 + \dot{y}_b^2 + \dot{z}_b^2) \quad (7)$$

$$T_2 = \frac{1}{2} J_w \dot{\theta}_l^2 + \frac{1}{2} J_w \dot{\theta}_r^2 + \frac{1}{2} J_\psi \dot{\psi}^2 + \frac{1}{2} J_\phi \dot{\phi}^2 + \frac{1}{2} n^2 J_m (\dot{\theta}_l - \dot{\phi})^2 + \frac{1}{2} n^2 J_m (\dot{\theta}_r - \dot{\phi})^2 \quad (8)$$

$$U = mgz_l + mgz_r + Mgz_b \quad (9)$$

T_2 의 다섯 번째와 여섯 번째 항은 왼쪽/오른쪽 DC 모터의 전기자(armature) 회전 운동에너지이다.

라그랑지안 방정식($L = T_1 + T_2 - U$)과 운동 상태를 표현하는 일반화 좌표는 오른쪽/왼쪽 휠의 평균 각도(θ), 몸체 피치 각도(ψ), 몸체 요 각도(ϕ)이다. 일반화 좌표를 이용하여 일반화 힘을 구하면 비선형방정식으로 유도된다. 이륜 도립진자시스템의 평형 점에서 운동 방정식을 선형화함으로써 현대 제어이론에 근거한 상태방정식을 유도할 수 있다. 즉, 이륜 도립진자시스템의 제어목표는 몸체 피치 각도(ψ)을 0° 로 유지시키는 것이다. 이를 ($\lim_{\psi \rightarrow 0} : \sin \psi \rightarrow \psi, \cos \psi \rightarrow 1,$

$\dot{\psi}^2 \rightarrow 0$) 이용하면 선형화가 가능하며 운동 방정식은 다음과 같이 근사화 된다. 이에 대한 보다 자세한 내용은 참고문헌 [2]에 기술되어 있다.

$$\begin{aligned} ((2m+M)R^2+2J_w+2n^2J_m)\ddot{\theta}+(MLR-2n^2J_m)\ddot{\psi} &= F_\theta \\ (MLR-2n^2J_m)\ddot{\theta}+(ML^2+J_\psi+2n^2J_m)\ddot{\psi}-MgL\psi &= F_\psi \\ \left(\frac{1}{2}mW^2+J_\phi+\frac{W^2}{2R^2}(J_w+n^2J_m)\right)\ddot{\phi} &= F_\phi \end{aligned} \quad (10)$$

식 (10)의 첫째 및 둘째 항은 θ 와 ψ 를 각각 모두 가지는 반면 셋째 항은 단지 ϕ 로만 표현된다. 그러므로 상태방정식을 다음과 같이 분리하여 표현할 수 있다.

$$\mathbf{x}_1 = [\theta \ \psi \ \dot{\theta} \ \dot{\psi}]^T, \quad \mathbf{x}_2 = [\phi \ \dot{\phi}]^T, \quad \mathbf{u} = [v_l \ v_r]^T \quad (11)$$

따라서, 이륜 도립진자 시스템의 상태 방정식은 각각 다음과 같이 표현된다.

$$\dot{\mathbf{x}}_1 = A_1 \mathbf{x}_1 + B_1 \mathbf{u} \quad (12)$$

$$\dot{\mathbf{x}}_2 = A_2 \mathbf{x}_2 + B_2 \mathbf{u} \quad (13)$$

식 (12)는 이륜 도립진자 시스템의 자세 제어 운동, 식 (13)은 조향 운동을 각각 표현한다[2].

여기서

$$A_1 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -643.55 & -386.45 & 386.45 \\ 0 & 260.99 & 109.97 & -109.97 \end{bmatrix}, \quad (14)$$

$$B_1 = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 375.60 & 375.60 \\ -106.89 & -106.89 \end{bmatrix}$$

$$A_2 = \begin{bmatrix} 0 & 1 \\ 0 & -262.26 \end{bmatrix}, \quad B_2 = \begin{bmatrix} 0 & 0 \\ -77.97 & 77.97 \end{bmatrix} \quad (15)$$

이다. 상태 궤환 이득은 LQR 방법을 이용하며 가중 행렬 Q, R 은 시행착오에 의해 식 (16), (17)과 같이 정했다. 그 결과, 상태변수 궤환 제어이득은 식 (18), (19)와 같다.

$$Q_1 = \begin{bmatrix} 300 & 0 & 0 & 0 \\ 0 & 2.5 \times 10^3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad Q_2 = \begin{bmatrix} 200 & 0 \\ 0 & 10^4 \end{bmatrix} \quad (16)$$

$$R_1 = \begin{bmatrix} 2500 & 0 \\ 0 & 2500 \end{bmatrix}, R_2 = \begin{bmatrix} 500 & 0 \\ 0 & 500 \end{bmatrix} \quad (17)$$

$$lqr_1 = [-0.2449, -39.7554, -1.0922, -3.9157] \quad (18)$$

$$lqr_2 = [-0.3162, -1.9005] \quad (19)$$

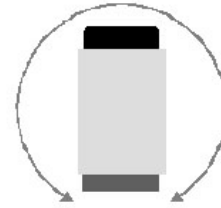


그림 4. 자이로 센서^[6]
Fig. 4. Gyro Sensor.

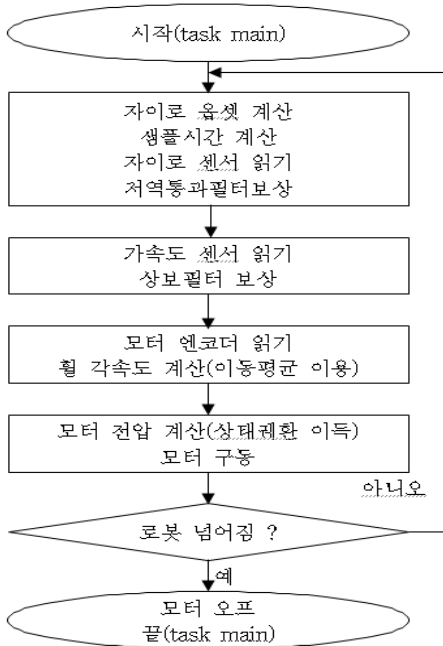


그림 3. 이륜 도립진자 로봇 소프트웨어 순서도
Fig. 3. Flowchart of Two-Wheel Inverted Pendulum Robot Software.

위의 상태변수 제환 제어이득을 바탕으로 이륜 도립진자 로봇을 구현하기 위한 소프트웨어 순서도는 그림 3과 같다.

III. 이륜 도립진자 로봇시스템의 성능 실험

3.1 자이로 센서 처리

NXT 자이로 센서는 $\pm 360^\circ / \text{sec}$ 까지의 각속도를 측정한다. NXT 자이로 센서를 사용하기 전에 윗셋 값을 계산하여 보정하여야 한다. 아래는 자이로 센서 측정 방향과 RobotC 언어로 구현된 자이로 데이터 처리의 일부분이다.

```

for(i=0; i<OFFSET_SAMPLES; i++){
    Raw_Omega = SensorValue(GYRO);
    TotalCounts++;
    Sum_RawOmega += Raw_Omega;
}
Omega_Bias = Sum_RawOmega / TotalCounts;
    
```

3.2 가속도 센서 처리

NXT 가속도 센서는 3축이며 $-2g \sim +2g$ 범위의 가속도뿐만 아니라 기울기를 측정할 수 있다. 아래는 가속도 센서 측정 방향과 RobotC 언어로 구현된 가속도 데이터 처리의 일부분이다.

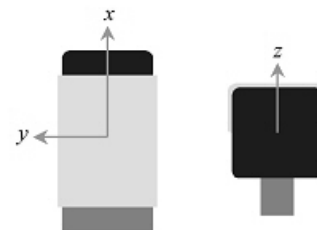


그림 5. 가속도 센서^[6]
Fig. 5. Accelerometer Sensor.

```

if (!HTACreadAllAxes(HTAC, xaxis, yaxis, zaxis)) {
    nxtDisplayTextLine(2, "ERROR occurred !!");
    wait1Msec(1000);
    break;
}
pitch = atan2(y_axis, -x_axis)*180 / PI;
    
```

3.3 자이로와 가속도 센서 데이터 융합

자이로 센서를 적분한 결과로 나타나는 각도는 고주파 영역에서 응답특성이 좋기 때문에 고역통과필터를 적용시킨다. 반면, 가속도 센서에서 계산된 각도는 저주파 영역에서 응답특성이 좋았기 때문에 저역통과필터를

적용시킨다. 센서 융합을 위해 칼만 필터가 많이 이용되지만 상보필터는 칼만 필터 보다 훨씬 단순하고 설계하기가 쉽다. 또한 발산할 염려도 없다. 대신 융합할 센서의 주파수 특성이 서로 보완적인 특성을 갖는 경우에만 적용가능하다. 위에서 언급한 자이로와 가속도 센서는 서로 보완적인 특성을 가지고 있어서 각각의 장점을 이용할 수 있다. 아래는 1차 상보필터 구성도와 RobotC 언어로 구현된 상보필터 데이터 처리의 일부분이다.

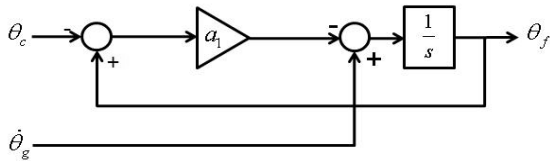


그림 6. 상보 필터 구조
Fig. 6. Complementary filter.

```
Comp_PitchAngle = 0.93*(Comp_PitchAngle +
    Gyro_Speed*dt) + 0.07*pitch;
Body_PitchAngle = Comp_PitchAngle;
```

3.4 모터 엔코더 센서 처리

직류전동기에 내장된 엔코더를 이용하여 바퀴의 회전 각도를 검출하며 이동평균을 이용하여 바퀴의 회전 각속도를 계산한다. 아래는 RobotC 언어로 구현된 엔코더 센서 데이터 처리의 일부분이다.

```
Left_Motor = nMotorEncoder(LEFT_MOTOR);
Right_Motor = nMotorEncoder(RIGHT_MOTOR);
Motor_Theta = (Left_Motor + Right_Motor) / 2;
Motor_ThetaDelta = Motor_Theta - Motor_ThetaPre;
Motor_Angle += Motor_ThetaDelta;
Motor_Rate = (Motor_ThetaDelta +
    Motor_ThetaDeltaPre1 + Motor_ThetaDeltaPre2
    + Motor_ThetaDeltaPre3) / (4*dt);
Motor_ThetaDeltaPre3 = Motor_ThetaDeltaPre2;
Motor_ThetaDeltaPre2 = Motor_ThetaDeltaPre1;
Motor_ThetaDeltaPre1 = Motor_ThetaDelta;
Motor_ThetaPre = Motor_Theta;
```

3.5 제어이득 변화에 대한 도립진자 로봇 성능 실험

상태변수 궤환을 위한 제어이득에는 4가지 종류가 있다. 몸체 피치 각도(KBODYPITCHANGLE), 몸체 피치 각속도(KBODYPITCHRATE), 휠 각도(KWHANGLE),

휠 각속도(KWHANGLERATE) 등이 있다. 여기에서 제작 실험한 로봇의 제어이득은 LQR 방법을 통해 얻은 식 (19)를 기반으로 튜닝을 통해 아래와 같이 설정하였다. 여러 번의 반복 실험을 통한 결과 로봇은 휠 각도와 휠 각속도이득의 변화에 보다 민감한 특성을 보였다.

$$lqr_1 = [-0.25, -37.5, -0.55, -2.5] \quad (20)$$

위의 제어이득을 바탕으로 모터에 인가되는 전압은 배터리의 양에 따른 상대적인 비율로 [-100%~100%] 까지 전압 레벨을 조절할 수 있으며 아래의 식 (21)로 표현된다.

$$\begin{aligned} \text{Voltage_level} = & \text{KPITCHRATE} * \text{Body_PitchRate} + \\ & \text{KPITCHANGLE} * \text{Body_PitchAngle} + \\ & \text{KMOTORANGLE} * \text{Motor_Angle} + \\ & \text{KMOTORRATE} * \text{Motor_Rate}; \quad (21) \end{aligned}$$

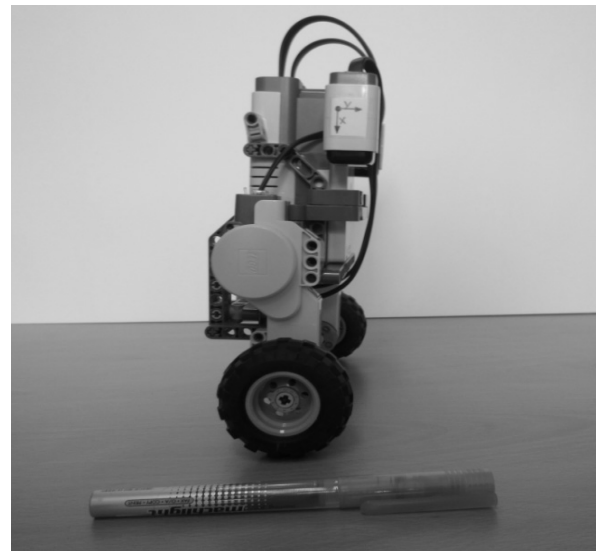


그림 7. 균형을 잡는 로봇
Fig. 7. Image of self-balancing Robot.

그림 7은 로봇이 스스로 균형을 수행하는 작업 중 한 장면을 보여 준다.

IV. 결 론

본 논문에서는 NXT 마인드스툼, 서보 직류전동기, 자이로 센서, 가속도 센서를 이용하여 이륜 도립 진자 로봇 시스템을 제작 및 실험하였다. 첫째, 회전각을 검출하는 대표적인 센서인 엔코더를 이용하여 바퀴의 회전 각도를 검출하며 이동평균을 이용하여 바퀴의 회전 속도를 계산하였다. 둘째, 회전 각속도를 측정하는 자이

로 센서 실험 및 자이로 센서 바이어스를 계산하였다. 셋째, 충격뿐만 아니라 물체의 기울기를 측정할 수 있는 가속도 센서를 실험하였다. 넷째, 자이로와 가속도 센서 각각의 장점을 활용하기 위한 센서 융합을 실험하였다. 마지막으로, 상태변수 궤환 제어이득 변화를 통해 이륜 도립진자 로봇에 미치는 성능 영향을 실험을 통해 확인하였다.

NXT 마인드스톰을 이용한 도립진자 로봇은 이미 준비되어있는 모터, 센서 등을 이용하여 로봇을 쉽게 구현할 수 있는 교육용 로봇 도구이다. 이를 통해 학습자가 제어 대상에 대한 모델 기반 설계 기법을 배우고 모터 및 센서 데이터 처리를 효과적으로 학습할 수 있음을 본 논문을 통해 확인하였다. 본 논문에서는 제어를 구형인 NXT를 사용하였다. 향후, 보다 진보된 EV3 제어기 또는 일반적인 프로세서로 구현된 도립진자 로봇과의 비교 및 분석을 수행할 계획이다.

REFERENCES

- [1] Katsuhiko Ogata, Modern Control Engineering, 1997.
- [2] Yori-hisa Yamamoto, NXTway-GS Model-Based Design - Control of self-balancing two-wheeled robot built with LEGO Mindstorms NXT. First Edition, 29 Feb., 2008.
- [3] Steven J. Witzand, Coordinated LEGO Segways, 2009.
- [4] Dae-Kwan Ku, Jun-Keun Ji and Gueesoo Cha, "Embedded Control System of Segway Robot using Model Based System", The Korea Academia-Industrial cooperation Society, Vol. 11, No. 8, pp. 2975-2982, 2010.
- [5] D. Lofaro, T. Giang, Paul Oh, Mechatronics Education: From Paper Design to Product Prototype Using LEGO NXT Parts, 2009.
- [6] <http://www.hitechnic.com>

저 자 소 개

정 보 환(정회원)

2006년 경북대학교 전자공학과 박사 졸업.

1998년~현재 용인송담대학교 정보통신과 교수