

보급형 센서를 장착한 자율주행로봇의 위치제어기 설계 및 제작

° 홍순학*, 최병홍**, 한석균*, 박종현***, 김용일*

* 한국과학기술연구원 기전연구부
**한국과학기술인 정보및 통신공학과
***한양대학교 정밀기계공학과

Abstract

This paper presents the design and implementation of an industrial controller for an autonomous guided vehicle(AGV) with economic sensors. A guidance scheme provides accurate tracking and achieves faster minimizing of tracking error. A sensor at the center provides the position and orientation of the vehicle relative to the track. Control laws that make use of this information have been devised to achieve accurate and fast tracking. The gains are modified on-line to achieve proper tracking. The simulation and implementation results are provided for the illustration of the implemented controller.

1. 서론

본 논문에서는 자기테이프를 따라 진행하는 자율주행로봇을 위한 보급형 센서를 이용한 위치제어기를 설계 및 제작하였다. 트랙 추종방식 제어기의 목표는 자율주행로봇이 주행을 하면서 외부의 외란에 대하여 트랙으로부터 벗어나지 않게 하는 것이 목적이다. 추종센서(tracking sensor)로부터의 위치신호를 제어 알고리즘에 적용하여 빠른 시간 안에 추정오차가 줄어들게 하는 알고리즘을 개발했다. 제어기는 16bit 단일칩 컨트롤러(single chip controller)를 사용하여 제작하였다.

자율주행로봇의 유도방식은 크게 유도형과 자율형이 있다. 현재 산업현장에서 사용되는 자율주행로봇은 유도형이 주류를 이루고 있으나 점차적으로 자율형에 관한 관심이 고조되고 있다. 유도형에는 크게 광학식과, 자기 자성체식, 전자식으로 나눌 수 있다.

유도형 방식 중에서 자기 자성체식은 경로변경이 용이하고, 보수가 거의 필요 없다. 일반적으로 자율주행로봇의 경로제어는 트랙을 추종하는 알고리즘에 의해 많이 수행되어져 왔다[1]. 이러한 트랙추종방식은 현재까지 많은 연구가 진행되어 왔으며, 실제로 산업현장에서 많이 쓰이고 있다[3].

트랙을 추종하는 알고리즘은 트랙을 감지하는 센서의 종류와 센서가 장착된 위치에 따라 다양하게 개발되어왔다. 가장 일반적인 방법은 하나의 센서를 이용하여 트랙을 추종하는 방법이며, 자율주행로봇의 앞,뒤에 두개의 센서를 부착하여 트랙을 추종하는 방식도 있다[1]. 자율형 자율주행로봇에서는 초음파센서 또는 비전센서를 이용한 방식이 일반적이다[2].

2. 제어시스템의 설계

2.1 제어시스템의 사양

제어시스템의 제어기는 단일칩 컨트롤러인 80C196KC를 사용하였다. 80C196KC는 16비트(bit) 데이터 처리가 가능하고, 5개의 I/O포트를 가지고 있다. 이중 2개의 I/O포트를 메모리를 제어하기 위한 시스템버스로 사용하고, 나머지 포트들은 외부의 주변기기들을 제어하는데 사용하였다. 메모리는 32kbyte 롬(Read Only Memory)과 32kbyte 램(Random Access Memory)을 사용하였다[그림 1].

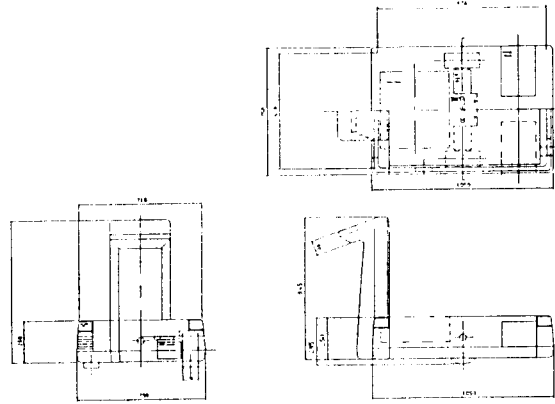
센서는 트랙을 추종하기 위하여 추종센서와 자이로센서(Gyro sensor)를 사용하였다. 추종센서는 16비트의 디지털신호를 출력하는데, 자기테이프를 감지하면 디지털 신호 "1"을, 감지하지 못하면 디지털 신호 "0"을 나타낸다. 추종센서는 분해능이 1cm이므로, 폭이 5cm인 자기테이프로부터 수직으로 3cm떨어졌을 때 5~6비트의 디지털신호 "1"을 나타낸다. 자이로센서는 각속도에 비례하여 0.5~4.5V의 아날로그신호를 발생시킨다. 자이로센서가 측정할수 있는 각속도

의 범위는 $-90^{\circ} \sim 90^{\circ}$ 이다. 자이로센서는 각속도가 없을 때는 2.5V의 전압을 발생한다.

구동부는 앰프, 모터, 하모닉드라이브, 구동륜으로 구성된다. 앰프는 제어기로부터 기준전압을 입력 받아 모터를 구동한다. 앰프는 자체에 진류피드백 루프를 가지고 있어 모터가 일정한 토크를 발생시킨다. 또한 높은 주파수에서도 좋은 성능을 내기 위하여 질연개이트 바이폴라트랜지스터(IGBT)를 사용하였다[그림 2].

자율주행모터를 구동하기위해 사용된 모터의 소비 전력은 300W이다. 이 모터의 정격전압은 75V이고, 분당회전수는 3000rpm이다. 하모닉드라이브의 감속비는 50 : 1 이다.

자율주행로봇의 중심축에는 양쪽에 구동륜이 두개 있고, 앞과 뒤에는 구동성이 없는 보조륜이 있다. 즉, 양쪽에 있는 구동륜의 속도에 따라 자율주행로봇의 진행속도와 진행방향이 결정된다. 이 자율주행로봇의 가반하중은 500kg이고, 평균 주행속도는 36m/min이다. [그림 3]과 [그림 4]는 자율주행로봇과 자율주행로봇의 설계도를 나타내고있다.



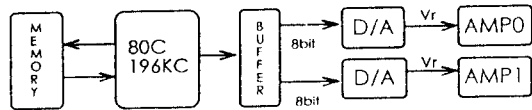
[그림 4] 자율주행로봇의 설계도

2.2 제어시스템의 설계

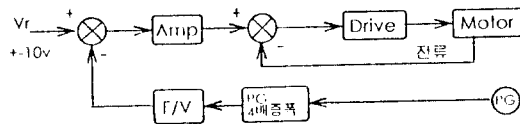
[그림 5]는 전체적인 제어시스템의 블럭도를 나타낸다. 제어 알고리즘에서 기준 속도값을 계산하면, 이 값은 디지털-아날로그 변환기를 통하여 전압을 출력한다. 그러면 앰프들이 이 전압 값을 받아 모터를 구동하게 되고, 자율주행로봇은 트랙을 따라 주행을 하게 된다.

자율주행로봇은 주행을 하면서 일정한 시간 (Sampling time)마다 트랙으로부터의 이탈여부를 감지한다. 주행 중에 바닥의 표면상태, 자율주행로봇의 비선형성, 트랙의 특성등 여러 가지 외란에 의해서 자율주행로봇이 트랙을 벗어나게 되고, 이것을 추종 센서가 감지한다. 제어기는 이 센서로부터 신호를 입력 받게 된다. 제어기는 이 센서로부터 신호를 입력 받게 된다. 제어기는 추종오차의 크기에 비례하여 제어 이득을 계산하고 추종오차를 줄이기 위하여 제어법칙을 통하여 속도값을 계산한다.

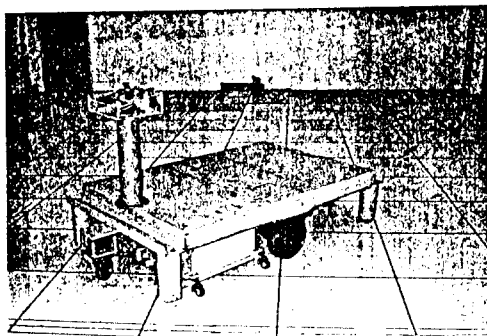
제어기에서 속도값을 계산할 때 주의해야 할 점은 구동부의 비선형구간을 고려해야하는 것이다. 즉, 자율주행로봇이 초기상태에서 동작하기 시작할 때 속도값이 너무 작으면 바퀴는 회전하지 않게 되고, 속도값이 어느 값 이상이 되면 바퀴의 회전수는 변하지 않게 되므로 제어할 수 있는 속도값 구간을 설정하여야 한다.



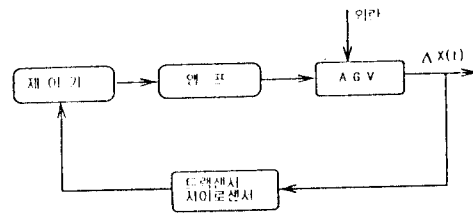
[그림 1] 제어기 블럭도



[그림 2] 앰프 블럭도

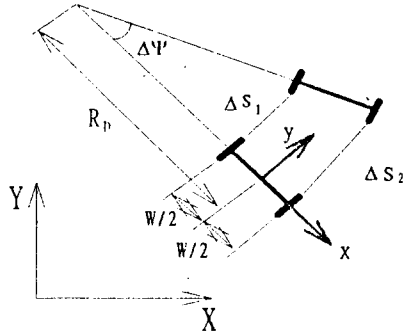


[그림 3] 자율주행로봇



[그림 5] 제어시스템 블럭도

3. 제어 알고리즘



[그림 6] 자율주행로봇의 주행 거리계

단위 시간동안 두 바퀴가 움직인 거리를 각각 Δs_1 , Δs_2 라고 하고 자율주행로봇의 폭을 W 라하면,

$$\begin{aligned} \Delta s_1 &= (R_D - \frac{W}{2})\Delta\psi \\ \Delta s_2 &= (R_D + \frac{W}{2})\Delta\psi \\ R_D &= \frac{\Delta s_1 + \Delta s_2}{2\Delta\psi} \end{aligned} \quad (1)$$

또한 두 바퀴의 움직인 거리의 차와 W 를 이용해서 $\Delta\psi$ 를 나타내면,

$$\Delta\psi = \frac{\Delta s_2 - \Delta s_1}{W} \quad (2)$$

식(1),(2)에서,

$$R_D = \frac{W}{2} \left(\frac{\Delta s_2 + \Delta s_1}{\Delta s_2 - \Delta s_1} \right) \quad (3)$$

이동 좌표계에서,

$$\begin{aligned} \Delta x &= R_D(1 - \cos(\Delta\psi)) \\ \Delta y &= R_D \sin(\Delta\psi) \end{aligned} \quad (4)$$

Δx , Δy 를 기준 좌표계의 ΔX , ΔY 로 나타내면,

$$\begin{bmatrix} \Delta X \\ \Delta Y \end{bmatrix} = \begin{bmatrix} \cos \Delta\psi & -\sin \Delta\psi \\ \sin \Delta\psi & \cos \Delta\psi \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta y \end{bmatrix} \quad (5)$$

식(5)로부터 트랙으로부터 벗어난 절대위치 ΔX 를 계산할 수 있다[5]. 이 ΔX 로부터 식(6)을 이용하여 제어 이득을 계산할 수 있다.

$$K(t) = K_f \Delta X(t) + T_D \times \Delta \dot{X}(t) + \frac{1}{T_I} \times \int \Delta X(t) \quad (6)$$

식(6)에서 $\Delta X(t)$ 는 센서가 샘플링 주기마다 트랙으로부터 벗어난 거리를 측정해낸 값이다. 식(6)에서 K_f, T_D, T_I 는 Ziegler-Nichols 조정방법[7]에 의하여 결정하였다.

$\Delta X(t)$ 가 양수이면 주행방향을 기준으로 볼 때 트랙의 오른쪽으로 자율주행로봇이 벗어난 것이므로 오른쪽 구동륜의 속도를 증가시키고, 왼쪽 구동륜의 속도를 감소시켜야 트랙으로 다시 돌아올 수 있다. 그러므로 자율주행로봇의 제어 알고리즘은 식(7)에 의해서 결정된다.

$$\begin{aligned} R_PWM(t) &= R*(1+K(t)) \\ L_PWM(t) &= R*(1-K(t)) \end{aligned} \quad (7)$$

위 식에서 R 은 제어가 만들어내는 기준 속도값이고, $R_PWM(t)$ 과 $L_PWM(t)$ 은 오른쪽 바퀴와 왼쪽 바퀴를 구동하는 속도값이다. 식(6)에서 구해진 $K(t)$ 를 식(7)에 대입함으로써 $\Delta X(t)$ 를 줄이기 위한 속도값을 계산할 수 있다.

4. 제어알고리즘 성능 시뮬레이션 및 실험

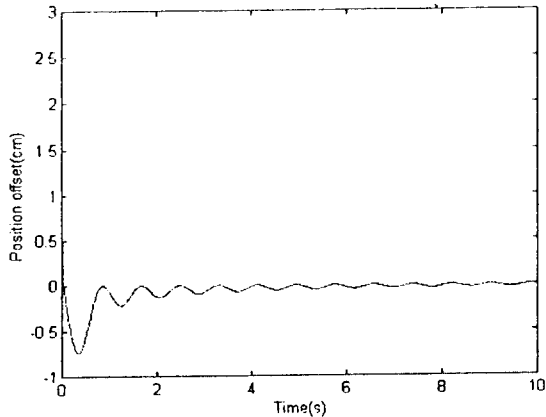
4.1 시뮬레이션 및 실험 방법

위에서 제안된 제어 알고리즘을 가지고 자율주행로봇의 추종성능을 시뮬레이션 하였다. 기본 속도값을 제어기에 입력하여 주행을 시작하고, 초기상태에서 위치 오차는 3cm로 주었다. 여기서 오차는 자율주행로봇이 트랙으로 벗어난 거리 $\Delta X(t)$ 을 말한다. 제어기의 샘플링주기는 10ms이다. 알고리즘을 시뮬레이션한 후 알고리즘을 롬에 저장하여 실제 자율주행로봇을 실험하였다. 실제 실험을 할 때 주의할 점은 양쪽 모터에 똑같은 속도 값을 제어기가 출력하더라도 구동륜의 물리적인 특성에 따라서 실제 바퀴의 속도는 차이가 나게 되므로 이를 고려하여 속도 값을 계산하여야 한다.

4.2 시뮬레이션 및 실험 결과

[그림 7]에서 나타나듯이 시뮬레이션 결과를 고찰하면 0.1sec안에 추종오차가 최소화되는 것을 알 수 있다. 트랙센서의 분해능이 1cm이므로, 시뮬레이션 결과에 나타난 0.1sec 후의 추종오차들은 자율주행로봇이 트랙을 따라 실제로 주행할 때에는 트랙을 벗어 나지 않은 것으로 인식된다. 시뮬레이션 후 이 알고리즘을 실제 자율주행로봇에 실험한 결과 트랙을 따

라 잘 추종하는 것을 볼 수 있었다. 위 알고리즘의 실험결과를 고찰하면 위치제어는 잘되었지만 자세제어가 잘되지 않는 것을 알 수 있었다. 이 결과는 자세에 대한 고려가 제어 알고리즘을 구성하는데 포함되지 않은 결과라고 생각된다.



[그림 7] 시뮬레이션 결과

5.결 론

본 논문에서는 자율주행로봇을 설계하고, 위치를 제어하기 위해 제어를 설계하고 제어알고리즘을 구성한 후 시뮬레이션과 실험을 하였다.

2차원에서 자율주행로봇을 제어하는데 있어서는 위치와 자세제어가 중요한데 위 논문에서는 위치제어만 좋은 결과를 낼 수 있었다. 자율주행로봇의 자세 제어를 위해서는 자이로센서로부터 자율주행로봇의 각속도($\dot{\theta}$)를 측정하여 자세 제어를 위한 제어 알고리즘을 구성하여 자세와 위치를 동시에 제어할 수 있는 알고리즘을 구성하여야 한다.

자율주행로봇의 위치 및 자세제어가 완성된 후에는 장애물감지, 주 통제실과의 통신, 다중 자율주행로봇의 주행 등 자율주행로봇의 성능향상에 관한 연구를 계속 진행할 것이다.

감사의 글

본 연구진행에 있어서 애플설계 및 제작을 하시고, 전 사회로에 관한 여러 가지 많은 도움을 주신 송태오 선생님과 우성우씨에게 진심으로 감사를 드린다.

또한 AGV 설계 및 제작에 많은 도움을 주신 노덕문 선생님과 김상조씨에게 진심으로 감사를 드린다.

참고문헌

1. Rajagopalan, R., Cheng, R.M.H., Lequoc, S., "A guidance control scheme for accurate track following of AGVs", IEEE International Conference on Robotics and Automation, Vol.1, p.188-193, 1992.
2. U. Raschke, J. Borenstein, A. "Comparison of Grid-type Map-building Techniques by Index of performance", IEEE International Conference on Robotics and Automation, p.1828-1832, 1990.
3. T. Tsumura, "Survey of Automated Guided Vehicle in Japanese Factory", IEEE International Conference on Robotics and Automation, p.1329-1334, 1986.
4. Intel 16bit Embedded controller handbook, 1990
5. Barry Steer, "Trajectory Planning for a mobile Robot", The International journal of Robotics Research, Vol.8, No.5, October 1989.
6. James L. Crowley, "Asynchronous Control of Orientation and Displacement in a Robot Vehicle", IEEE International Conference on Robotics and Automation, p.1277-1282, 1989.
7. Gene F. Franklin, J. David Powell, Abbas Emami-Naeini, "Feedback control of dynamic systems", Addison Wesley, 3rd, 1994.
8. Karl J. Åström, Björn Wittenmark, "Computer-controlled system", Prentice-Hall, 2nd, 1990.