

CAN을 이용한 복수 전동기의 위치 동기화에 관한 연구

이 흥 희, 정 의 현

울산대학교 전기 전자 및 자동화공학부

The Study on Position Synchronization for Multi-motors using Controller Area Network

Hong-Hee Lee, Ui-Hern Jeong

School of Electrical Engineering and Automation, Uiversity of Ulsan, Korea

ABSTRACT

In this paper, we introduce the network based multi-motors control system using CAN (Controller Area Network). The traditional multi-motors control system has many problems in the view of reliability and economy because of the amount and complexity of wiring, noise and maintenance problems, etc.

These problems are serious especially when the motor controllers are separated widely. CAN is generally applied in car networking in order to reduce the complexity of the related wiring harnesses. These traditional CAN application techniques are modified to achieve the real time communication for the multi-motor control system. And also, the position synchronization technique is developed and the proposed methods are verified experimentally.

Key words : position synchronization, real time, multi-motor control, network, CAN

1. 서 론

산업현장에서 제어용 네트워크를 사용하는 기본적인 목적 가운데 하나는 기존의 점대점 (peer-to-peer) 연결방식에서 발생하는 배선구조의 복잡성을 줄이고 공정변화에 따른 배선상의 유연성 확보와 함께 유지 보수능력을 향상시키는 것이다.

이러한 제어용 네트워크의 필요성이 증가함에 따라 다양한 통신 프로토콜이 표준화 작업이 이루어졌다. 이러한 결과로 1980년대 중반부터 Profibus, Interbus-S, P-Net, LON, FIP등과 같은 필드버스

와 센서/액추에이터 프로토콜들이 개발되기 시작하여 상용화되었고 1980년대 후반부터 "autobus" 프로토콜이 등장하여 시제품이 개발되었다^[1].

CAN(Controller Area Network)은 현재 가장 많이 사용되고 있는 "autobus" 프로토콜이다. CAN은 1980년대 초반 독일의 R. Bosch GmbH에 의해 연구, 개발되었으며 1992년에는 Mercedes S-class차량의 엔진 제어기, 기어박스 제어기, dash board사이의 고속 통신이 요구되는 네트워크와 저속 데이터 통신으로도 가능한 에어컨 제어에 처음 적용되었다. 1995년 이후 차량용으로 가장 인정받는 프로토콜로 자리잡았으며 지난 수년간 차량 내의 열악한 환경에서 안정적으로 동작하여 신뢰도를 인정받고 가격대 성능비가 우수하여 공장자동화용 네트워크에 적용되고 있는 추세이다. 대표적 통신망으로는 DeviceNet, CAN Kingdom, CANopen/CAL, SDS등이 있으며 ISO/OSI 7계층 중 응용계층에서 프로토콜을 달리한다^{[2],[4]}.

한편, 여러 대의 전동기를 동시에 제어할 경우 각각의 전동기에 대한 제어명령이나 속도 또는 위치 정보의 교환을 위해 많은 신호선들이 필요하게 되어 배선구조가 복잡해지고 노이즈 및 유지보수의 어려움으로 인해 시스템 신뢰도와 경제적 측면에서 문제가 된다. 특히 제어하고자 하는 전동기의 제어장치가 서로 멀리 떨어져 있을 경우 이러한 문제는 더욱 심각해진다. 이와 같은 단점을 개선하기 위해 본 연구에서는 전동기 제어장치들을 하나의 노드 (node)로 보고 이들을 포함하는 네트워크를 구축하였다. 즉, 기존의 CAN을 이용해 여러 대의 전동기를 제어하기 위해 필요한 각종 제어명령이나 속도 또는 위치정보를 단일 통신선을 이용해 실시간으로 주고받을 수 있는 제어용 네트워크를 구축했다. 또한 제안된 네트워크 시스템을 복수 전동기의 위치 동기화 시스템에 적용하여 네트워크 동작의 타당성을 실험적으로 증명해 보였다.

2. CAN의 기본구조

2.1 CAN 프로토콜

CAN은 CSMA/CD+AMP (Carrier Sense Multiple Access/Collision Detection + Arbitration on Message Priority) 프로토콜을 가진다. 이는 IEEE 802.3 CSMA/CD 프로토콜과 유사하며 ISO/OSI 7 계층 중 물리계층과 데이터 링크 계층만을 지원한다. 이러한 CAN 프로토콜을 간단히 설명하면, 연결된 모든 노드는 전송해야 할 메시지가 있는 경우 버스의 상태를 감지해 버스 비활성화 상태에서 이를 전송한다. 만약 하나 이상의 노드에서 버스사용권을 요구했을 경우 메시지나 노드에 부여된 식별자(ID)를 이용해 우선 순위가 높은 노드에서 버스 사용권을 얻는다. (이러한 버스사용권 획득방법은 NBA(Non-deductive Bitwise Arbitration)라 불린다.) 우선 순위가 낮아 버스 사용권을 획득하지 못한 노드는 버스의 상태를 감지하다 비활성화 상태가 되면 메시지를 재전송한다.^{[5],[6]}

2.1.1 CAN 데이터 프레임

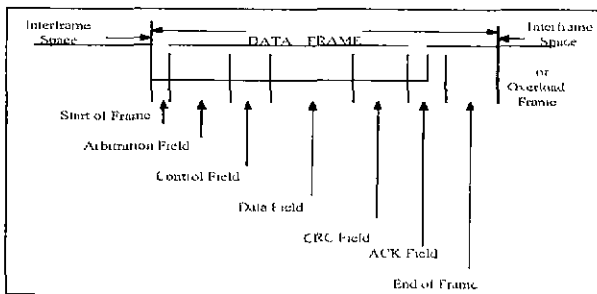


그림 1 CAN 데이터 프레임

CAN 2.0A와 2.0B가 제안되어 있으며 2.0A의 경우 11bits, 2.0B의 경우 29bits 길이의 식별자를 가지고 있으며, 2.0B를 지원하는 제품의 경우 확장된 기능들이 지원되고 있다.^[3]

2.2 CAN 노드의 구성

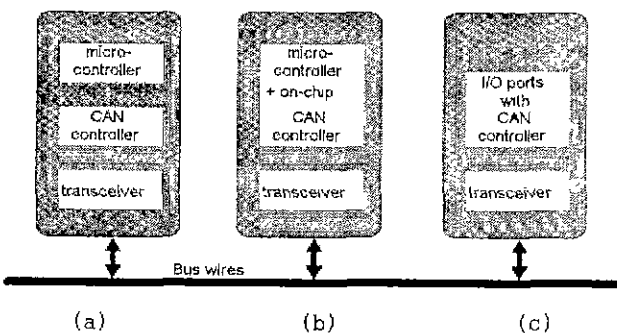


그림 2 CAN 노드 구성 예

CAN통신 기능을 포함하는 노드(제어기)를 구성하는 방법으로 그림2에 나타난 세 가지 방법이 사용되고 있다. 일반적으로 Stand alone CAN 제어기 형태로 구성되고 있으며 여기에 적용되는 마이크로컨트롤러는 대부분 범용 프로세서가 사용되고 있는데 이는 시스템의 구조가 변경되어도 재사용이 가능하다는 장점이 있다.

Integrated (micro-controller based) CAN 제어기 형태로 구성할 경우 구조가 간단해지지만 현재 제공되는 컨트롤러의 종류가 적어 요구되는 제어기 사양을 맞출 수 없는 경우가 발생할 수 있다. 또한, Serial Linked I/O Device는 CAN제어기에 디지털/아날로그 입출력 기능을 부여한 것으로 마이크로컨트롤러나 기타 입출력 장치의 거리를 넓히는데 사용되고 있으나 저속(125Kbps)이라는 단점이 있다.^{[6],[8]}

3. 제어용 네트워크 시스템

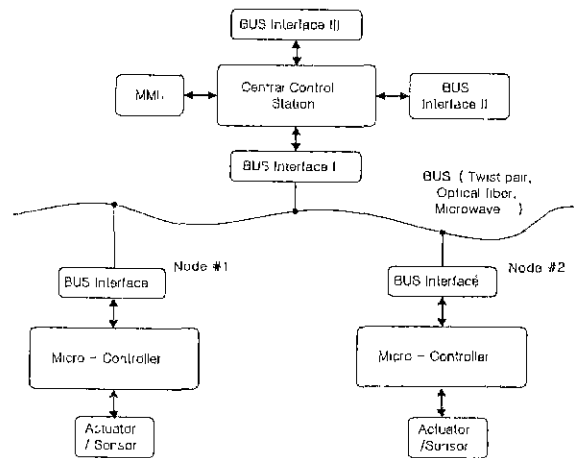


그림 3 분산제어기를 연결한 네트워크 시스템

그림3은 분산 제어기들을 네트워크를 이용해 연결한 시스템의 전형적인 예이다. 이 경우 국부 제어기는 전체 공정에 따른 분산제어를 기능을 수행하고 알고리즘에 따라 공정의 센서신호등을 분석하고 그 결과에 따라 액추에이터를 제어하기도 한다. 또한 같은 계층의 다른 국부제어기 및 상위제어기로부터 각종 정보나 제어 상태를 수집하여 제어기능을 수행하거나 또 다른 상위 제어기에 정보를 전달하는 기능을 수행한다.

제어기들 사이에서 이루어지는 정보교환이 빈번하게 발생하지 않아 버스 사용률이 낮은 시스템일 경우는 직렬통신망을 사용하는 것이 가능하다. 이러한 경우 공정 변화에 따른 설치상의 유연성을 확보가 가능하고 전체 시스템의 구성이 간단해지므로 배선작업으로 인한 경제적 손실을 막을 수 있고 유지보수가 용이해진다. CAN 이용해 네트워크를 구

성할 경우 시스템의 요구사항에 따라 broadcasting 방식을 사용하거나 노드 또는 메시지에 특정 어드레스를 부여방식을 사용할 수 있다. Broadcasting 방식은 현재 메시지를 전송하고 있는 노드가 마스터 역할을 수행하고 네트워크상의 나머지 노드는 슬레이브가 되어 메시지를 수신한다. 이러한 기능을 이용하여 전체 시스템에 초기화 정보나 기타 유용한 정보를 모든 노드에 전송할 수 있다.

그림 4는 본 연구에서 네트워크 구현에 사용된 CAN 노드의 블록도를 나타내고 있는데 Stand alone 형태로 구현하였다.

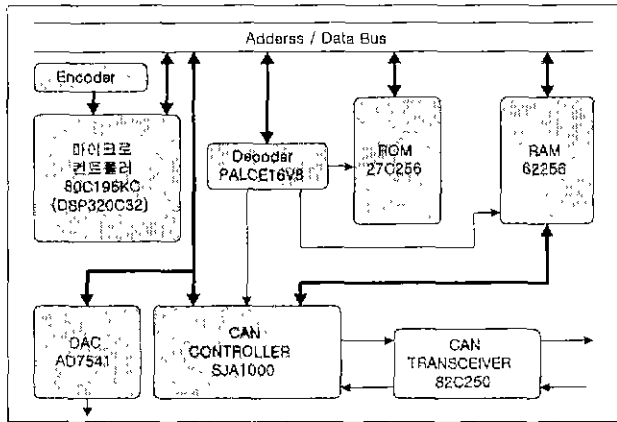


그림 4 범용노드 블록도

4. 실험

제안한 네트워크 시스템의 타당성을 고찰하기 위해 전동기 2대를 사용하고 있는 위치동기 시스템에 적용했다. 전동기의 위치동기 알고리즘은 다양한 방법이 있으나 본 연구에서는 네트워크 특성을 살펴보기 위해 그림 5와 같은 간단한 위치 동기 시스템을 채택했다.

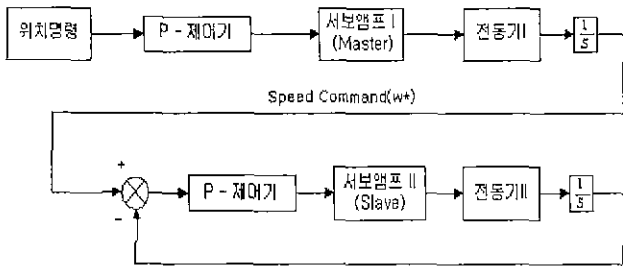


그림 5 위치제어 시스템 블록도

그림과 같은 두 대의 전동기 제어시스템에서 전동기 속도(위치)는 상호의존적이라고 가정한다. 알고리즘의 간략화를 위해 한쪽 전동기를 마스터로 두고 다른 하나는 이를 추종하는 슬레이브로 둔다.

또한, 그림 3에서와 같이 마스터는 제어기에서 출력되는 임의의 속도(위치)명령에 따라 구동되고 슬레이브는 마스터의 실제 속도(위치)를 받아 중앙제어기(호스트)에서 계산된 속도(위치)명령에 의해 구동된다고 가정했다. 이러한 가정을 바탕으로 본 연구에서 CAN을 이용한 복수전동기 동기화 시스템을 그림 6과 같이 구성하였다.

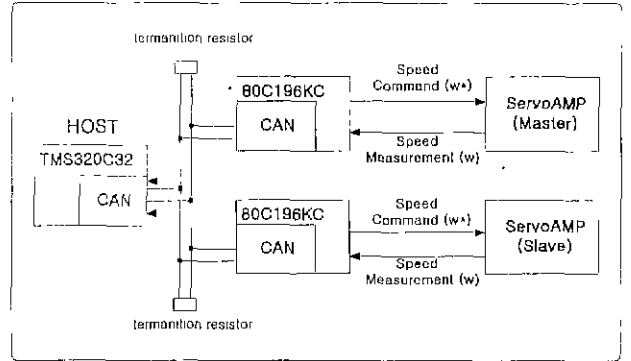


그림 6 복수 전동기 동기화 시스템

전동기 제어를 위한 제어기는 그림 4와 같으며 호스트 제어기는 TMS320C32에 Stand alone 형태의 CAN 제어기를 이용하여 버스 인터페이스를 구현하였다. 데이터 흐름도에서 알 수 있는 바와 같이 마스터 전동기의 속도(위치)정보를 CAN 네트워크를 통해 이를 호스트로 전송하고 호스트에서는 마스터의 속도나 위치정보를 이용해 슬레이브의 속도 명령을 출력하여 슬레이브 전동기가 마스터의 속도나 위치를 추종하도록 하고 있다. 시스템 구현에 사용된 주요사항은 다음과 같다.

표 1 시스템 자원 및 변수값

CAN Spec.	2.0A
데이터 전송속도	1Mbps
호스트 메시지 길이	2Bytes
I/O 보드메시지 길이	4Bytes
제어주기	1ms
DAC	12bits

그림 7은 1Mbps로 전송되는 CAN 데이터의 비트 타이밍을 보여주고 있는데 본 실험에서 전송 주기가 1Mbps이므로 데이터 비트의 길이가 $1\mu s$ 임을 알 수 있다. 그림 8은 버스상의 트래픽과 데이터를 수신하였을 때 CAN 제어기에서 각 마이크로컨트롤러로 보내어지는 인터럽트 신호를 보여준다. 즉, 채널1은 CAN 버스 데이터를, 채널2, 3, 4는 각각의 CAN controller로부터 마스터, 슬레이브 및 호스트 프로세서에 인가되는 인터럽트 신호를 보여주고 있다. CAN의 경우 식별자는 메시지의 우선 순위 결정과 메시지 필터링에 사용될 수 있는데 필터링 기능은 노드(메시지)의 수가 많을 경우 빈번한

인터럽트의 발생으로 인한 시스템 성능저하를 방지하는데 사용될 수 있다.^[5]

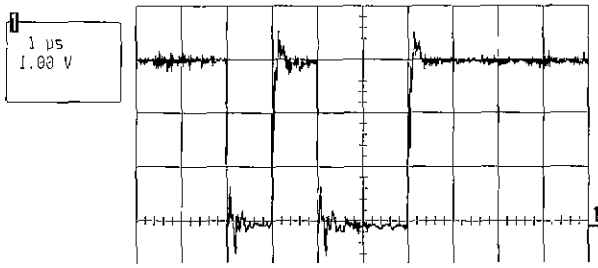


그림 7 CAN 데이터 파형

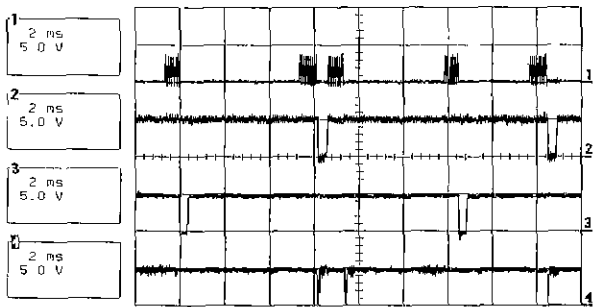


그림 8 네트워크상의 트래픽 및 인터럽트

그림 9는 마스터 및 슬레이브에 인가되는 속도 지령치를 나타내고 있는데 마스터에 일정한 속도지령치를 인가 한 후 이 속도지령치를 호스트가 읽어들이 슬레이브에 마스터와 같은 속도지령치를 인가한 경우이다. 그림 10은 그림 5와 같은 제어알고리즘으로 전동기를 구동할 경우 마스터 및 슬레이브의 실제 속도를 나타내고 있다.

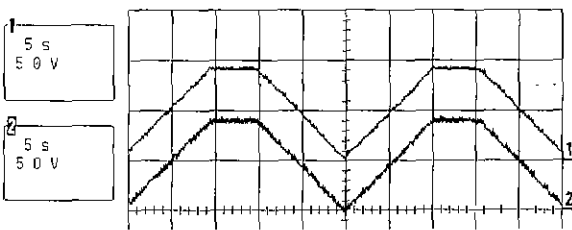


그림 9 마스터/슬레이브 속도지령치

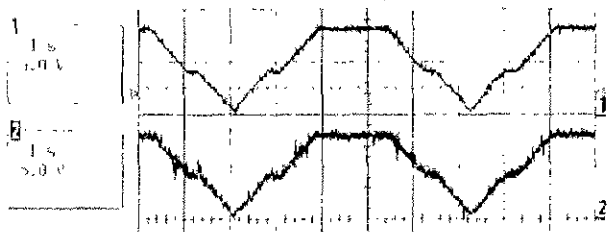


그림 10 동기화 알고리즘에 의한 전동기 속도

실험 결과에서 알 수 있는 바와 같이 본 연구에서 구현한 네트워크 시스템으로 전동기의 속도 또는 위치동기 제어를 위한 실시간 데이터 전송이 가능함을 알 수 있다.

5. 결론

본 논문에서는 복수 전동기를 구동할 경우 복잡한 배선작업에서 발생하는 문제를 해결하기 위해 CAN을 사용해 간단한 제어용 네트워크를 제안하였다. 실험에서 두 대의 전동기로 구현된 시스템의 제어주기를 1KHz로 하였으나 좀 더 정확한 시간 정보를 이용한 시분할 방식으로 데이터를 전송하거나 각 제어기간의 동기화가 이루어진다면 좀 더 빠른 제어주기를 얻을 수 있다. 이러한 결과에서 복수 전동기 구동시스템에 CAN을 이용한 제어용 네트워크의 적용은 안정적인 실시간 통신을 제공할 수 있었다. 이를 토대로 시스템의 성능을 개선하기 위해서는 별도로 추가되는 하드웨어 없이 각 제어기들을 동기화시킬 수 있는 알고리즘 개발이 향후과제로 남는다.

이 논문은 정보통신부의 연구비 지원에 의하여 연구되었음

참고 문헌

- [1] Wolfhard Lawrenz, "CAN System Engineering From Theory to Practical Application", Springer, 1997.
- [2] G. Cena and A. Valenzano, "An Improved CAN Fieldbus for Industrial Application", *IEEE Trans. On Industrial Electronics*, Vol.44, No 4, Aug. 1997.
- [3] CiA, CAN Specification 2.0 PartA, B <http://www.can-cia.de>, 1997
- [4] CiA, CAN Application Layer, www.can-cia.de, 1997
- [5] CiA, CAN Datalink Layer, www.can-cia.de, 1997
- [6] CiA, CAN Physical Layer, www.can-cia.de, 1997
- [7] Philips, "SJA1000 Stand-alone CAN Controller", Philips Semiconductor, 1997 Nov.
- [8] Philips, "CAN Serial Linked I/O device(SLIO)", Philips Semiconductor, 1996 Jun.