

TCP/IP를 이용한 다중DC모터 제어시스템 구축

김 이철\*, 정 태수\*, 김 승현\*, 박 기현\*  
\*성균관대학교

Development of Multiple DC-Motor Control System using TCP/IP

Yicheal kim\*, Teasoo Jung\*, Seunghyun kim\*, Kiheon Park\*  
\*SungkyungKwan Univ.

**Abstract** - Recently, rapid process of network technology has an effect on the field of control engineering. And it gives effect to network control system(NCS) research that grafts existing control system and network has been studied. According to one-chip development of TCP/IP that is the most useful in LAN server for hardware, it can usefully and conveniently apply to network control system before. This paper construct network control system that can work close-loop control takes advantage of Ethernet that is the most for general data transmission network. Here, protocol uses for data transmission makes use for TCP/IP. This work the basic data transmitter-receive experiment to take advantage of network node was produced. And then, DC-motor can be control plant. Finally, I will prove to build speed control system of multiple DC-motor through Ethernet.

1. 서 론

최근 네트워크 기술의 비약적인 발전은 제어공학 분야에 많은 영향을 주었다. 이를 홈 오토메이션, 공장자동화기기 등과 같이 기존 제어시스템에 네트워크를 적용하려는 연구가 활발히 진행되고 있다. 과거에는 느린 전송 속도로 인한 다수의 데이터 전송 시 트래픽 증가와 패킷 손실 등의 이유로 인해 제어계통에 네트워크를 적용하는 것을 기피하였다. 현재는 네트워크 성능의 눈부신 발전으로 이와 같은 문제점을 충분히 해결할 수 있게 되었다. 네트워크 제어 시스템은 하나의 공유버스를 통해 제어에 필요한 모든 장치들을 연결할 수 있기 때문에 시스템에 필요한 배선을 최소화 할 수 있으므로 설치 및 보수비용 감소, 시스템관리 비용 절감을 꾀할 수 있으며 네트워크를 통한 근거리의 원격제어도 가능하게 만든다는 장점이 있다. 현재 네트워크 제어 시스템 구축에 널리 이용되고 있는 프로토콜은 CAN(Control Area Network), Ethernet(TCP/IP), Control Net등이 있다[1],[2].

본 논문에서는 일반적인 데이터 전송용 네트워크로 보편적으로 많이 사용하고 있는 Ethernet을 이용하여 페루프 제어를 수행할 수 있는 네트워크 제어시스템(NCS)을 구축하고자 한다. 여기서, 데이터 전송에 사용한 프로토콜은 TCP/IP를 이용한다. 먼저, TCP/IP 계층 구조와 데이터 전송방식 등 네트워크 프로토콜에 대한 전반적인 내용을 분석하고 이후 단일 칩으로 구현된 TCP/IP 프로토콜 제어기를 이용하여 Ethernet 상에서 제어에 필요한 각종 정보를 송수신 할 수 있는 네트워크 노드(Network Node)를 제작할 예정이다. 그리고 제작된 네트워크 노드를 이용하여 기본적인 데이터 송수신 실험을 수행하고 DC-Motor를 제어대상 플랜트로 선정할 후 최종적으로 Ethernet을 통한 다중 DC-Motor의 속도제어 시스템을 구축하고자 한다.

2. 본 론

2.1 Ethernet

1976년 Xerox PARC의 Metcalf 등은 원격지에서 상호 데이터를 전송할 수 있는 프로토콜을 발표하였다. 이후, 제록스, 디지털, 인텔 등이 공동 개발한 LAN (Local Area Network)의 표준규격인 IEEE802.3과 호환되는 Ethernet(Version2.0)이 개발되었다. 현재, 전세계적으로 Ethernet망이 형성되어있다. Ethernet망을 제어시스템에 적용할 경우 가장 큰 장점은 최대 10 ~ 100Mbps에 이르는 고속의 데이터 전송대역폭을 이용하여 제어에 필요한 데이터 이외에 다양한 관련 정보들을 실시간 전송 할 수 있으며, 기존의 잘 구축된 인터넷망을 그대로 활용할 수 있어 그 경제적 파급효과가 높다는데 있다. 그러나 Ethernet의 데이터 전송구조상 네트워크의 시간지연(Time Delay)이나 전송데이터의 신뢰성을 보장하지 못한다는 점은 네트워크 제어시스템 설계시 해결할 문제이다[3].

2.1.1 Ethernet의 계층적 구조

네트워크의 계층구조는 <그림 1>과 같이 OSI 7계층 모델로 표현할 수 있으며, 이 중에서 Ethernet은 2계층에 해당하는 데이터링크(Data Link layer)계층에 해당한다.

Application	
Presentation	
session	
Transport	TCP
Network	IP
Data link	Ethernet
Physical	Physical(PHY)

그림 1. OSI 7계층

데이터링크 계층은 상위계층에서 전달된 메시지를 물리적인 네트워크에 전달하는 계층이며, 버스점유 방식인 CSMA/CD(Carrier Sense Multiple Access / Collision Detect)을 갖는다. 상위계층의 응용계층(Application Layer)은 적절한 응용프로그램을 구성하여 전체 네트워크를 구축한다. 물리계층은 데이터를 송수신 하는 두 스테이션 사이에 실제적인 전기적 신호가 전송되는 경로를 말하며, UTP(Unshield Twisted Pair), 동축케이블, 광섬유 등이 이용되고 있다. Ethernet 기반의 TCP/IP는 각각 전송계층(Transport Layer)과 네트워크계층(Network Layer)으로 나타낼 수 있다 [2],[3].

2.1.2 메시지 전송방식

Ethernet은 버스공유 방식의 CSMA/CD점유 방식을 이용한다. CSMA/CD방식은 먼저, 데이터를 송신하려는 스테이션을 현재 네트워크 버스가 Idle한 상태에 있을 때까지 대기하며, 버스를 사용한 있는 다른 스테이션이 없을 때는 곧바로 데이터 전송을 시작한다. 이 때, 서로 다른 스테이션에서 데이터가 서로 충돌할 경우, 임의의 시간대기 상태에 들어간 후 Exponential Back-off 알고리즘에 의해 데이터를 재전송을 요청하게 된다. 이때 재전송 요청이 지정된 한계를 벗어나면(16회 이상) 에러가 발생했음을 상위계층에 통보하게 된다. 그러나 일반적으로 Ethernet을 이용한 데이터통신은 네트워크 부하가 적고 전송속도가 빠른 경우에는 대단히 높은 데이터 전송 성공률을 보장하는 것으로 알려져 있다. 다음 <그림 2>은 CSMA/CD방식에서 메시지를 전송하는 과정을 나타내는 Flow Chart이다[3].

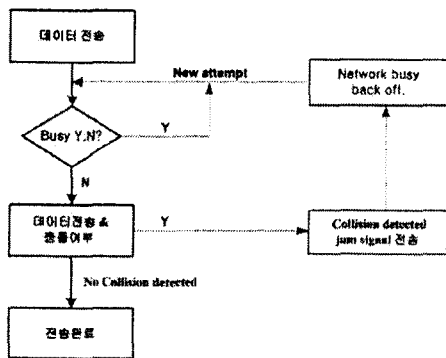


그림 2. CSMA/CD Flow Chart

### 2.2 DC-Motor 모델링

본 논문에서 적용할 네트워크 제어시스템의 제어대상 플랜트로는 DC-motor를 선정하였다. DC-motor는 유도전동기의 기본원리가 되는 회전자계에 의한 회전토크의 발생 원리를 이용한 것으로써, 이를 적용해 본 논문의 직류전동기의 속도제어를 수행하였다. 직류전동기에 대한 일반적인 모델은 <그림 3>와 같다.

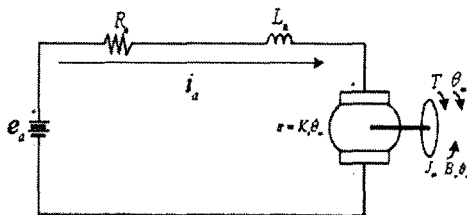


그림 3. 직류전동기의 일반적인 모델

위의 <그림 3>로부터 직류전동기의 토크방정식과 전기방정식을 유도해서 상태방정식과 출력방정식을 유도하면 식(1),(2)과 같다.

$$\begin{bmatrix} \frac{di_a}{dt} \\ \frac{d\omega_m}{dt} \end{bmatrix} = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K_b}{L_a} \\ \frac{K_t}{J_m} & -\frac{B_m}{J_m} \end{bmatrix} \begin{bmatrix} i_a \\ \omega_m \end{bmatrix} = \begin{bmatrix} 1 \\ 0 \end{bmatrix} \frac{1}{L_a} e_a \quad (1)$$

$$y = [0 \ 1] \begin{bmatrix} i_a \\ \omega_m \end{bmatrix} \quad (2)$$

네트워크 제어시스템은 전체적으로 볼 때 이산화 제어시스템이라 말할 수 있다. 식(1),(2)의 연속플랜트를 샘플링 주기  $h[\text{sec}]$ 일때 이산화 하면,

$$x(k+1) = A_d x(k) + B_d u(k)$$

$$y = C_d x(k)$$

$$A_d = e^{Ah} \quad B_d = \left( \int_0^h e^{A\tau} d\tau \right) B \quad (3)$$

$$C_d = C \quad D_d = D$$

$$x(k) = [i_a(k), \omega_m(k)]^T$$

위 식(3)에서 이산화된  $A_d, B_d, C_d, D_d$  파라메타들을 식(4)에 의해 제어기에 삽입될 이산적인 전달함수를 구한다.

$$H(s) = C(SI - A)^{-1} B + D \quad (4)$$

$$= \frac{\det(SI - A + BC) - (SI - A)}{\det(SI - A)}$$

위의 식(4)에 의해서 구해진 전달함수를 이용하여 시뮬레이션 블록선도는 <그림 4>와 같이 나타낼 수 있다.

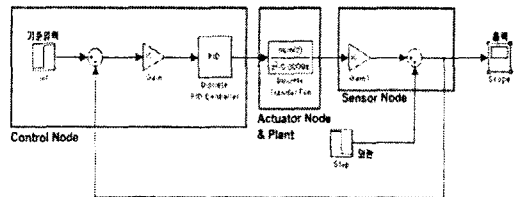


그림 4. 시뮬레이션 블록선도

네트워크 Control Node에서 계산된 제어입력은 플랜트에 연결된 Actuator Node에 전송된다. 그리고 Sensor Node에서는 현재 DC-Motor의 회전속도를 측정(Tachometer)하여 이를 제어노드에 전송하는 역할을 수행한다. Actuator Node에서는 수신한 제어입력을 이용하여 DC-Motor구동에 필요한 입력을 플랜트에 인가한다.

### 2.3 네트워크 노드 구성

본 논문에서 Ethernet 네트워크를 이용하여 제어를 수행할 수 있는 네트워크 제어시스템을 구축하고자 한다. 먼저 Ethernet 네트워크상에 제어에 필요한 데이터를 전송할 수 있는 네트워크 노드를 구성해야 하며, 본 논문에서 제작된 Node의 구성도는 <그림 5>와 같다.

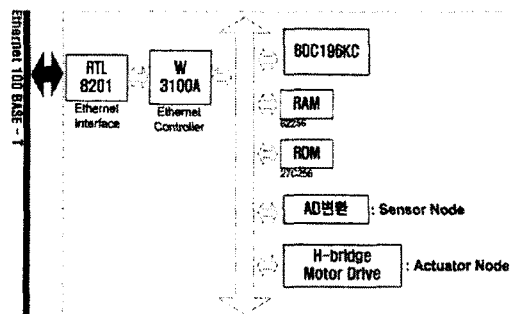


그림 5. 네트워크 노드 구성도

각 네트워크 노드의 구성 부분은 다음과 같다.

- 전원부  
National, LM2575 : PWM방식, +5V정전원 공급
- 통신부  
Maxim, RS-233C : 기본네트워크 환경 설정
- 메모리부  
SGS- THOMSON,27C256 : EPROM  
ISSI, IS61256AH : RAM
- 제어부  
Intel, 80196KC20 : Main Controller
- 액츄에이터부  
SGS-THOMSON, L298 : Motor 구동 Drive
- 네트워크부  
Realtek, RTL8210 : Ethernet interface

위의 네트워크 구성을 실제 하드웨어로 구현한 네트워크 노드 사진은 <그림 6>, <그림 7>과 같다.

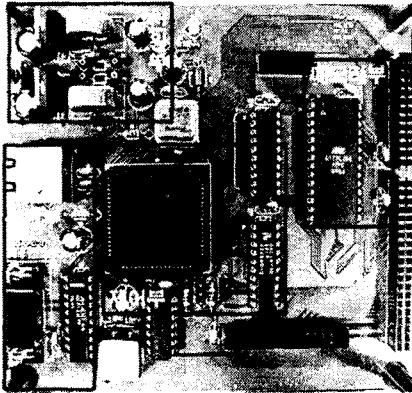


그림 6. 네트워크 노드(MCU)

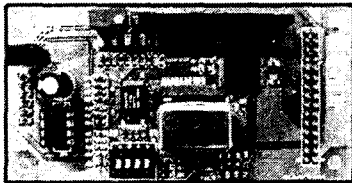


그림 7. 네트워크 노드(Network)

여기서, 80196KC는 두 가지 역할을 한다. 첫째는, RS-233을 이용하여 PC와 노드간에 직렬통신을 함으로써, 통신에 필요한 Subnet Mask, Gateway, IP Address, 그리고 MAC address를 네트워크부에 있는 Serial EEPROM에 저장하는 역할을 하며, 둘째, Control Node로부터 기준신호를 입력받아 DC-Motor를 제어하는데 있어 제어입력을 계산하는 역할을 한다. 계산된 제어입력은 A/D컨버터에 의해 DC-Motor를 구동시킨다. 네트워크부에서는 직접버스I/F모드의 기본방식을 채택하고 Full Duplex하게 설계하였다. 또한, Ethernet 물리층 Device는 RTL8201과 Ethernet Controller W3100A에 데이터 전송을 용이하게 하기 위해서 MII(Media Independent Interface)를 이용하였다. MII는 Serial MII방식과 Nibble MII방식을 사용할 수 있으나, 구현한 방식은 Nibble MII방식으로써 4bit의 TXD[3:0]과 RXD[3:0]으로 구성 되어있다. 그리고 시스템 클럭을 2.5MHz, 25MHz 두 가지 사이클을 이용하여 전송속도를 10Mbps~100Mbps를 수신할 수 있도록 설정하였다. 본 논문에서

서는 25MHz를 사용함으로써 최대100 Mbps까지 전송 속도를 가지도록 네트워크 노드를 설계하였다[5],[6].

### 3. 결 론

본 연구에서 Ethernet을 이용하여 다중 DC-Motor를 제어하기 위해 구축된 전체 시스템의 구성도는 <그림 8>과 같다.

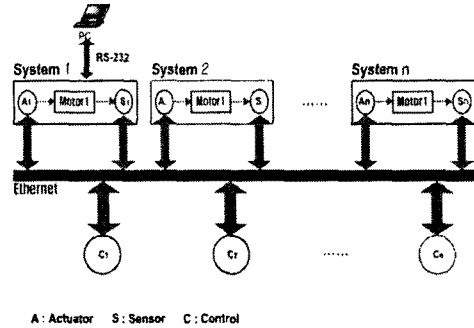


그림 8. 전체 시스템 구성도

모터제어에서는 네트워크에서 발생하는 시간지연이나 패킷 손실이 크게 문제되지 않았지만, 다수의 제어를 구현한다면 두 가지 현상에 의한 제어성능 변화가 예측된다. 그러나 본 논문에서는 현재 가장 많이 활용되고 있는 Ethernet 기반의 TCP/IP 프로토콜을 이용하여 네트워크 제어시스템을 구성하였다. 또한, TCP/IP 프로토콜을 구현하기 위해 하드웨어적인 단일 칩을 이용하여 프로그램 개발비용과 시간을 줄일 수 있도록 하였으며, Ethernet 네트워크상에서 데이터를 전송할 수 있는 독립적인 네트워크 노드를 직접 설계, 제작하였다. 이를 통해, 설계된 네트워크 노드가 일반적인 인터넷을 이용하여 제어시스템을 구축할 수 있음을 입증하였다. 향후, 본 연구에서 구축된 다중DC-Motor 제어 시스템을 직접 구현하고, 네트워크를 사용하지 않은 시스템과 비교 분석하여 효율성을 입증하고자 한다.

### [참 고 문 헌]

- [1] 정준홍, 최수영, 이종성, 박기현, "Network Control System의 안정도 분석에 관한 연구", 대한전기학회 하계 학술대회는문집, pp.2231-2233, 7, 2002
- [2] W. Richard Stevens, "TCP/IP Illustrated Volume1, 2(1), 2(2), 3", 1994
- [3] James F. Kurose, Keith W. Ross, "Computer Networking", 2001
- [4] 강신영, 김은주, 위석오, 김광현, 임영철, "인터넷기반 모터 원격제어 및 모니터링", Trans KIEE.Vol.50D, no.7 pp.279-285, 2002
- [5] Realtek, "Realtek single chip port 10/100Mbps fast ethernet by receiver RTL8210", 2002
- [6] i2ChipW3100A, Wiznet Datasheet, Available: <http://www.wiznet.co.kr>, 1999