

이더넷기반의 실시간 제어 통신망 구조의 성능 해석 및 실험

이 성 우[†]

한국 전력 연구원

(2005년 1월 27일 접수, 2005년 4월 8일 채택)

Performance Analysis and Experiment of Ethernet Based Real-time Control Network Architecture

Sung Woo Lee[†]

Korea Electric Power Research Institute, Daejeon, Korea

(Received 27 January 2005, Accepted 8 April 2005)

요 약

본 논문에서는 높은 대역폭과 안정성을 제공하는 DCS용 통신망의 구조를 제시하고 성능을 해석한다. 본 논문에서 제시하는 DCS용 통신망은 DCS 통신망에서 널리 사용되는 리플렉티브 메모리(Reflective Memory) 구조를 채용하며, 이에 따라 링형 토폴로지를 가진다. 물리계층으로 사무용 및 산업용으로 널리 쓰이고 있는 패스트 이더넷(Fast Ethernet) 물리 매체를 사용하여 100 Mbps의 대역폭을 가지고, 링형 토폴로지가 가지는 단점인 각 노드에서의 시간 지연을 줄이기 위해 RED(Ring Enhancement Device)라는 장치를 고안하여 사용한다. 본 논문에서 소개하는 DCS용 통신망을 ERCNet(Ethernet based Real-time Control Network)이라고 명명하며, ERCNet의 구조와 동작에 대해 설명한다. ERCNet의 통신 성능에 대한 수학적 해석을 수행하고 개발된 ERCNet을 이용한 실험을 통하여 해석 결과의 정확성과 통신망 성능을 검증한다.

주요어 : 링 가속장치, 패스트 이더넷, 리플렉티브 메모리, 이더넷기반 실시간 제어통신망

Abstract — This paper describes the implementation of DCS communication network that provides high bandwidth and reliability. The network for DCS in this paper adopts the Reflective Memory (RM) architecture and Fast Ethernet physical media that have 100 Mbps bandwidth. Also, This network uses Ring Enhancement Device (RED) which is invented to reduce the time delay of each node. The DCS network that is introduced in this paper is named as ERCNet (Ethernet based Real-time Control Network). This paper describes the architecture and working algorithms of ERCNet and performs numerical analysis. In addition, the performance of ERCNet is evaluated by experiment using the developed ERCNet network.

Key words : RED, Fast ethernet, Reflective memory, ERCNet

1. 서 론

1960년대에 자동제어 기술이 사용되기 시작하고 산업 현장에서 자동화 기기들이 갈수록 복잡해지고 증가함

에 따라 산업 현장에서는 분산 제어 시스템(Distributed Control System : DCS) 을 널리 사용하게 되었다^{[1][2]}. 산업 현장에서 쓰이는 공정이 복잡해짐에 따라 DCS의 처리 용량은 갈수록 커져가고 있으며 이에 따라 분산된 컴퓨터와 기기들 사이의 데이터를 교환하고 공유하는 기능을 하는 통신망의 중요성이 커지고 있다. DCS의 통신망은 발전하는 산업 환경의 요구 사항에 맞추기 위해서 높은 대역폭을 가져야 하며 산업 환경의 특성상 안정성을 가져야 한다. 특히 원자력 발전소 같은 중대한 기능을 하

[†]To whom correspondence should be addressed.
The Korean Academy of Science and Technology, KAST
Building, Gyunggi-Do 463-808, Korea
Tel: 02-566-3263
E-mail: kast@kast.or.kr

는 환경에서는 더욱 높은 안정성을 요구한다. 또한, 산업 현장에 한번 설치된 통신망은 10년 이상 오랫동안 사용 가능해야 하므로 높은 유지 보수성을 지녀야 한다^[4].

산업 환경에서 쓰이는 DCS에서는 데이터의 공유를 위해 사용하는 방법 중에 주요한 것으로 리플렉티브 메모리(Reflective Memory : RM) 방식이 있다. 리플렉티브 메모리 방식의 시스템이란 전체 네트워크가 일정한 공유 메모리를 가지고 있고 자신의 고유한 메모리 영역에 데이터를 업데이트하게 되면 전체 네트워크가 이를 자동으로 공유할 수 있도록 하는 방식의 네트워크 시스템이다. 리플렉티브 메모리 방식이 DCS의 통신망으로 사용되는 대표적인 제품으로는 Systran 사의 SCRAMNet이나 VMIC 사의 RTnet 등이 있다. 이러한 제품들은 표준화된 프로토콜을 사용하지 않는 각 회사의 고유한 산업용 통신망 솔루션이다. 그러므로 이러한 통신망을 설치한 경우 유지 보수에 있어서 각 회사에 전적으로 의존해야 하기 때문에 유지 보수에 많은 비용이 들며, 제조 회사 전용의 하드웨어를 사용하기 때문에 가격이 비싸다는 단점이 있다^{[4][5]}.

본 논문에서는 기존의 리플렉티브 메모리 방식의 통신망의 단점을 보완하고 발전하는 산업 환경에 적용할 수 있는 새로운 리플렉티브 메모리 구조의 DCS 통신망을 제안한다. 이를 위해 기존 리플렉티브 메모리 시스템에서 사용되는 통신망의 단점을 보완하기 위해 표준화된 구조를 사용하여 유지 보수성을 높이도록 하는 ERCNet(Ethernet based Real-time Control Network)을 구현하였다. 이 통신망은 기존 리플렉티브 메모리 통신망의 단점을 보완함과 동시에 발전하는 산업 환경에 적용될 수 있도록 하기 위해 상업적으로 널리 사용되고 있는 패스트 이더넷(Fast Ethernet)의 물리 매체를 사용하여 유지 보수성을 높였으며, 상태 기반 동작(state driven) 구조를 사용하고 토큰 패싱 메커니즘을 사용하여 안정성을 높였다. 또한 링형 토폴로지의 통신망에 가지는 노드에서의 시간 지연을 줄이기 위해 RED(Ring Enhancement Device)라는 장치를 고안하였다. 본 논문에서는 ERCNet의 특징적인 구조와 동작에 대해 설명하고 RED의 기능과 동작에 대해 살펴본다. 그리고 ERCNet의 통신 성능에 대한 수학적 해석을 수행하고, ERCNet을 이용하여 데이터가 전체 통신망에 업데이트 되는 시간을 수식으로 표시한다. 또한 개발된 ERCNet 통신망을 이용한 실험을 통하여 해석 결과의 정확성과 ERCNet의 성능을 검증한다.

본 논문의 구성은 다음과 같다. 2장에서는 ERCNet의 기본적인 구조에 대해서 서술한다. ERCNet의 프레임 구조와 데이터 전달 방법에 대해 설명하고 ERCNet의 통신 프로토콜과 RED에 대해서 설명한다. 3장에서는

ERCNet에 대한 성능 해석을 수행한다. 4장에서는 ERCNet 통신망의 테스트베드를 통한 실험을 한다. 5장에서는 결론을 맺고 향후 연구 과제에 대해 설명한다.

2. ERCNet의 기본적인 구조

ERCNet은 링형 토폴로지를 가지고 토큰 패싱 메커니즘과 패스트 이더넷 물리매체를 사용한다. 또한 ERCNet은 성능 향상을 위해 새로이 고안된 장치인 RED(Ring Enhancement Device)를 사용한다. 여기서는 ERCNet의 구조에 대해 설명한다.

2-1. ERCNet의 데이터 타입과 기본적인 전달 방법

ERCNet의 각 노드가 가지는 글로벌 메모리의 크기는 기본적으로 512 kB이다. 글로벌 메모리는 리플렉티브 메모리 방식으로 전체 노드가 공유하게 되는 메모리의 크기이다. ERCNet의 최대 노드 개수가 64개이므로 ERCNet의 모든 노드들은 자신에게 할당된 데이터 영역으로 기본적으로 8kB를 가지게 된다. ERCNet 노드가 가지는 글로벌 메모리 크기는 메모리 확장이 가능하며, 전체 512 kB의 글로벌 메모리는 기본적으로 노드당 8 kB이지만 초기에 환경 설정을 통해 전체 노드수 및 노드가 가지는 데이터양에 따라 노드별로 적절한 크기로 분할되어 할당된다. 각 노드들이 가지는 메모리 영역에 저장되는 데이터는 셀(Cell)이라고 불리는 기본적으로 128 바이트 크기로 할당된 데이터 영역으로 나뉘어져 관리된다. 이렇게 셀이라는 단위를 기본으로 하여 메모리 영역을 관리하는 이유는 통신 프레임을 처리할 때 좀더 빠르게 동작하도록 하고, 메모리 관리를 쉽게 하기 위해서이다. 이후 ERCNet이 동작되기 시작하면 ERCNet은 글로벌 메모리에서 자기 자신에 해당하는 부분을 전체 네트워크로 브로드캐스팅하기 시작한다.

ERCNet은 각 노드가 자신에게 할당된 데이터 영역을 다른 노드들에게 일정한 시간 주기안에 브로드캐스팅하여 다른 노드들이 그 노드의 데이터 영역을 계속해서 업데이트 하도록 하는 방식으로 동작한다. 이는 기존의 RM 시스템이 대개 사건기반 동작(Event Driven) 방식으로 구동하는 것에 비해 상태기반 동작(Status Driven) 방식으로 동작하는 것을 의미한다. 이러한 방식으로 설계한 이유는 안정성을 중시하여 설계한 결과이다. 각 노드는 수신된 데이터를 읽고 이 데이터의 위치를 판단하여 자신의 글로벌 메모리의 정해진 위치에 데이터를 써넣게 된다. 데이터를 전송한 노드는 자신이 전송할 데이터를 다 보내고 난 후 토큰 프레임을 다음 노드로 넘겨주게 된다. 이러한 데이터 전달 과정을 시간에 따라 나타내면 그림 1과 같다.

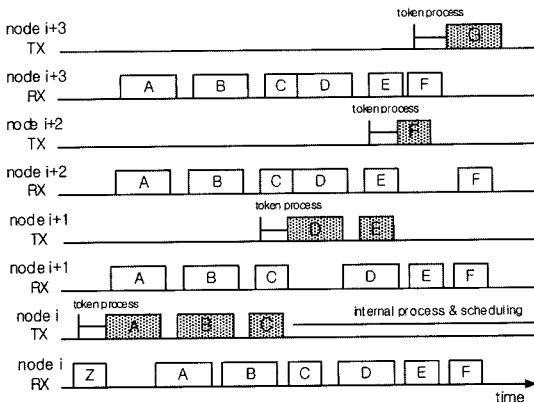


그림 1. ERCNet의 데이터 전달 프로세스.

2-2. ERCNet 통신 프로토콜

ERCNet은 OSI 7 계층 중에서 물리계층, 데이터링크 계층, 응용 계층의 세 계층만을 사용한다. 물리 계층으로는 패스트 이더넷 트랜시버를 사용하고 데이터 링크 계층으로는 토큰 패싱 메커니즘을 사용한다.

패스트 이더넷은 현재 가장 널리 쓰이고 있는 LAN 프로토콜중 하나로서 안정성 면에서 검증되어 있고, 유지 보수가 용이하다는 장점을 가진 프로토콜이다. ERCNet의 물리계층은 이 패스트 이더넷의 물리 매체를 그대로 사용한다. ERCNet은 패스트 이더넷 표준 중에서 광케이블 100Base-FX 표준의 물리매체를 사용하여 신호를 송수신 한다. ERCNet에서는 패스트 이더넷 표준의 Full-duplex 모드를 변형하여 사용한다. Full-duplex 모드는 본래 두 노드 사이를 일대일로 연결할 때 주로 사용하는 모드이지만, ERCNet에서는 현재 노드의 Tx 라인을 다음 노드의 Rx 라인과 연결하고 이전 노드의 Tx 라인을 현재 노드의 Rx 라인에 연결한다. 이러한 방법으로 케이블링을 하면 ERCNet의 각 노드들은 링 형태로 연결된다.

ERCNet의 데이터 링크 계층은 토큰 패싱 메커니즘을 사용한다. ERCNet의 데이터 링크 계층에서 토큰은 전송 권한과 같은 뜻으로 토큰을 가진 노드만이 전송을 할 수 있고, 토큰을 가진 노드가 전송을 마치면 다음 노드로 토큰을 넘겨주어서 다음 노드가 데이터를 전송할 수 있도록 한다. 이러한 방식으로 토큰은 전체 노드를 순환하게 된다. ERCNet에서는 이러한 토큰을 패스트 이더넷 프레임 포맷에서 Type/Length 필드를 사용하여 표시한다. ERCNet은 산업용 네트워크이기 때문에 Type 필드를 사용하지 않아도 된다. 그러므로 ERCNet에서는 일반적인 데이터 프레임의 경우 이 필드를 데이터 프레임의 길이를 나타내는데 사용하며, 토큰 프레임을 나타

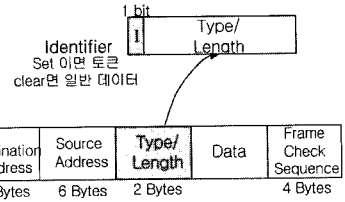


그림 2. ERCNet의 프레임 포맷.

낼 때만 이 필드의 가장 첫번째 비트를 토큰 식별자(identifier)로 삼아 1로 Set하여 토큰 프레임을 나타낸다. ERCNet의 프레임 구조는 그림 2와 같다.

2-3. RED (Ring Enhancement Device)

ERCNet은 링 형태의 토폴로지에서는 각 노드를 경유할 때 마다 시간 지연이 발생한다는 문제점이 생긴다. 이러한 문제점을 보완하기 위해 ERCNet에서는 새로이 고안한 특수한 장치를 사용한다. 이 장치는 RED(Ring Enhancement Device)라고 불리는 장치로 패스트 이더넷 트랜시버에서 수신된 아날로그 신호를 노드의 CPU에 보내줌과 동시에 패스트 이더넷 트랜시버를 통해 송신하는 장치이다. 이 장치는 CPU에서 제어 신호를 받아서 노드가 수신 상태에 있는 경우, 수신된 신호를 CPU와 송신 장치에 보낸다. 노드가 토큰을 받아서 송신 모드가 되면, 수신과 송신 사이의 연결을 끊고 보내고자 하는 데이터를 송신하는 방식으로 동작한다. 이러한 장치를 통해 ERCNet에서는 링형 토폴로지가 가지는 문제점 중 하나인 브로드캐스팅 할 때 각 노드를 거치면서 생기는 시간 지연을 최소화한다. 이 장치는 수신된 신호를 재전송할 때 CPU의 처리를 거치지 않고 패스트 이더넷 트랜시버를 통해 수신된 아날로그 신호를 바로 스위칭하기 때문이다. 따라서, 수신 모드의 노드들을 거치는데 걸리는 시간은 패스트 이더넷 트랜시버를 통과하는 시간과 아날로그 스위치를 통과하는 시간 밖에 걸리지 않는다.

3. ERCNet 성능 해석

이 섹션에서는 ERCNet의 데이터 업데이트 주기를 수식으로 표현한다. 이를 위한 파라미터들은 다음과 같다.

- ◇ 노드 개수 : N
- ◇ i번째 노드의 짧은 주기 데이터 양 : P_i
- ◇ i번째 노드의 긴 주기 데이터 양 : Q_i
- ◇ 한 프레임의 데이터크기 : F_a=1024 B
- ◇ Cell 크기 : M_c=128 B

- ◇ 헤더크기(M_{header}) : 16 B
- ◇ 한 프레임에 들어가는 cell의 수 : $N_{cell}=F_d/M_c=8$
- ◇ i 번째 노드에서 $i+1$ 번째 노드 사이 거리 : L_i
- ◇ 1 bit time : T_b
- ◇ 광전변환시간 : T_{oc}
- ◇ 전광변환시간 : T_{eo}
- ◇ Interframe gap : $T_g=96T_b$
- ◇ 전체 노드에서의 업데이트주기 : T_{up}
- ◇ 노드의 토큰전달시간 : T_{th}
- ◇ 노드의 토큰인식지연시간 : T_{Ath}
- ◇ 노드의 데이터 프레임 생성 시간 : T_{Dcr}
- ◇ 노드의 토큰프레임 생성시간 : T_{Tcr}
- ◇ 노드의 토큰프레임전송지연시간 : T_{Dt}

이때 ERCNet에서 어떤 번째 노드가 전송할 짧은 주기 셀의 개수와 긴 주기 셀의 개수를 각각 N_s , N_l 이라고 표시하면,

$$N_s=[P_i/M_c]$$

$$N_l=[Q_i/5 \times M_c]$$

라고 나타내어질 수 있다. 이를 통해 짧은 주기 프레임의 개수 N_{F_s} 와 긴 주기 프레임의 개수 N_{F_l} 는 다음과 같이 표시할 수 있다.

$$N_{F_s}=[N_s/N_{cell}]$$

$$N_{F_l}=[N_l/N_{cell}]$$

그리고 각 노드 사이의 전송 선로에 의한 전송지연(Propagation delay)을 전체 네트워크에서 나타내는 값인 T_{prop} 는

$$T_{prop} = \sum_{i=0}^N 5L_i$$

와 같이 나타내어진다. 이때 짧은 주기 한 프레임을 보내는데 걸리는 시간 T_{F_s} 은

$$T_{F_s}=(N_{F_s}-1)\{(F_d+M_{header}+M_{CRC})T_b+T_g\}$$

이고 긴 주기 한 프레임을 보내는 데 걸리는 시간 T_{F_l} 은

$$T_{F_l}=(N_{F_l}-1)\{(F_d+M_{header}+M_{CRC})T_b+T_g\}$$

이 된다. 하지만 마지막 프레임의 경우 N_{cell} 만큼의 크기로 보내지 않을 수도 있으므로 짧은 주기 마지막 프레임을 보낼 때의 시간은

$$T_{F_s}=\{(N_s \bmod N_{cell}) \times M_c + M_{header} + M_{CRC}\} T_b$$

가 되고 긴 주기 마지막 프레임을 보낼 때의 시간은

$$T_{F_l}=\{(N_l \bmod N_{cell}) \times M_c + M_{header} + M_{CRC}\} T_b$$

여기서 한 노드가 자신의 통신 프레임을 전송하는데 걸린 시간 $T_{D_{cell}}$ 은

$$T_{D_{cell}}=T_{F_s}+T_{F_s}+T_{Tg}+T_{Tl}+T_{oc}+T_{eo}+T_{prop}$$

이 되고 전체 네트워크에서의 업데이트 주기 T_{up} 은

$$T_{up} = \sum_{i=1}^N T_{Hi} = \sum_{i=1}^N \{T_{Ath} + (N_{F_s} + N_{F_l})T_{D_{cr}} + T_{D_{cell}} + T_{Tcr} + T_{Dt}\}$$

로 표시할 수 있다. 이 식을 이용하여 계산된 결과를 통해 정해진 데이터양과 노드 수가 주어질 경우 ERCNet 통신망의 업데이트 주기를 예측할 수 있다.

4. 실험

4개의 노드를 이용하여 위의 식을 테스트해 본 측정 결과가 그림 3에 나타나 있다. 그림 3은 토큰 홀드 시간 측정결과를 스캔한 것이다. User LED가 On일 때는 Low 신호이고, Off일 때는 High 신호이다. LED 신호가 High일 때 매니저 노드가 토큰을 받고부터 데이터를 전송하고 난 후 토큰을 다음 노드로 전송하기 까지(토큰을 홀드 시간(T_{th} : 400 μ s))의 시간을 나타낸 실험 값이다. 실험 환경과 실험을 통해 나온 결과의 값을 표 1로 나타내어 보면 다음과 같다.

이러한 실험치를 위의 식에 대입해보면,

$$T_{is}=(1024+16+32) \times 8\text{bit} \times 10 \text{ ns/bit}=85760 \text{ ns}=85.76 \mu\text{s}$$

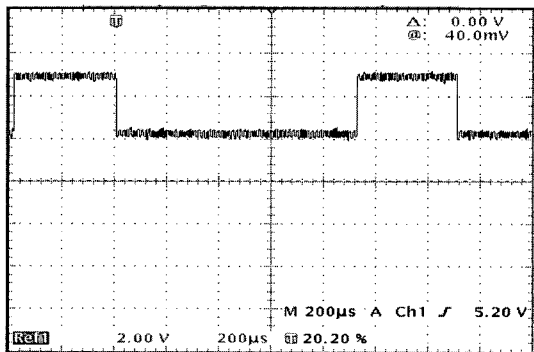


그림 3. 토큰 홀드 시간.

표 1. ERCNet 실험 데이터.

N	4	N_{F_l}	0.25
P_i	1920	T_{Ath}	10 μ s
Q_i	1152	$T_{D_{cr}}$	20.2 μ s
N_s	15	T_{Tcr}	1.1 μ s
N_l	2	T_{Dt}	34 μ s
N_{F_s}	2	T_{th}	400 μ s

$$T_{rs}=(7*128+16+32)*8\text{bit}*10\text{ ns/bit}=75520\text{ ns}=75.52\text{ }\mu\text{s}$$

$$T_{tr}=(1024+16+32)*8\text{bit}*10\text{ ns/bit}=85760\text{ ns}=85.76\text{ }\mu\text{s}$$

$$T_{tr}=(1*128+16+32)*8\text{bit}*10\text{ ns/bit}=14080\text{ ns}=14.08\text{ }\mu\text{s}$$

$$T_{D_{\text{cell}}}=85.76\text{ }\mu\text{s}+75.52\text{ }\mu\text{s}+85.76\text{ }\mu\text{s}+14.08\text{ }\mu\text{s} \\ +0.2\text{ }\mu\text{s}+0.15\text{ }\mu\text{s}=261.47\text{ }\mu\text{s}$$

$$T_{H_i}=10\text{ }\mu\text{s}+(2+0.25)(20.2\text{ }\mu\text{s})+261.47\text{ }\mu\text{s} \\ +1.1\text{ }\mu\text{s}+34\text{ }\mu\text{s}=352.02\text{ }\mu\text{s}$$

실험을 통해서 나온 T_{H_i} 값은 $400\text{ }\mu\text{s}$ 이고, 수식을 통해 나온 T_{H_i} 값은 $352\text{ }\mu\text{s}$ 로서 거의 비슷함을 알 수 있다.

5. 결 론

발전하는 산업환경에서 성능과 안정성은 갈수록 중요한 설계 요소가 되며, ERCNet은 이를 만족시킬 수 있는 성능을 발휘하고 안정성이 높도록 설계된 통신망이다. ERCNet은 100 Mbps 의 속도를 가지고, 토큰 패싱 메카니즘을 사용하여 안정적으로 데이터를 전송하며 RED라는 장치를 이용하여, 각 노드에서의 시간 지연을 최소화하였다. 본 논문에서는 리플렉티브 메모리 방식의 DCS용 네트워크로 개발된 ERCNet의 구조를 설명하고, ERCNet의 성능의 지표 중 하나인 업데이트 주기에 대한 수학적 해석을 행한 후 실험을 통해 그 정확성을 입증하였다. 본 논문에서 제시한 해석 결과를 통해 사용하는 데이터의 양과 셀의 크기에 따라 ERCNet에서의 데이터 업데이트 주기를 산출할 수 있으며, 이를 통해 실제 설치하기 전에 ERCNet을 이용한 DCS 시스템에서 데이터 업데이트 주기를 미리 알아볼 수 있다. 앞으로 더욱 많은 노드들을 가지고 ERCNet의 성능에 대해 검증해 볼 것이다.

참고문헌

1. Moon, H.-J. "Performance Analysis and Design of a Communication Network for Industrial Automation : A Case Study of the Mini-Map Network", Ph. D Thesis, Seoul National University, 1998.
2. Dobrivoje Popovic; Vijay P. Bhatkar. Distributed Computer Control for Industrial Automation, Marcell Dekka Inc., 1990.
3. Lee, J.M.; Park, T.R.; Kim, H.S.; Shin, Jae S.Y.; Choi, Y.; Kwon, W.H.; Lee, S.W.; Song, S.I. "A Communication Network with High Safety, Maintainability, and User Convenience for Digital I&C Systems of Nuclear Power Plants", 8th IEEE International Conference on Emerging Technologies and Factory Automation, Proceedings, 2001, 353-359.
4. Moorthy, V.; Jacunski, M.G.; Pillai, M.; Ware, P.P.; Panda, D.K.; Page, T.W., Jr.; Sadayappan, P.; Nagarajan, V.; Daniel, J. "Low-latency Message Passing on Workstation Clusters using SCRAMNet", IEEE 13th International and 10th Symposium on Parallel and Distributed Processing, April 1999, 148-152.
5. Chia Shen; Ichiro Mizunuma. "RT-CRM: Real-Time Channel-Based Reflective Memory", IEEE Transaction on Computers, November 2000, 49(11), 1202-1214.
6. IEEE Standards 802.3, 1998 Clause 3.2.6 Length/Type Field specifications and IEEE 802.1H-1995.