

전력선 기반의 홈 오토메이션용 제어 통신망의 성능 분석

조성국, 김동성, 이재민, 명관주, 권옥현
 서울대학교 전기컴퓨터공학부

Performance Evaluation of Powerline based Control Systems for Home Automation

Cho Sung Kug, Kim Dong-sung, Lee Jae Min, Myoung Kwan Joo, Kwon Wook Hyun
 School of Electrical Engineering and Computer Science, Seoul National University

Abstract - 본 논문에서는 홈 오토메이션용 제어 통신망에 널리 사용되는 Ethernet, CEBus, Lonworks 통신 규약에 대해서 성능을 분석하고 이를 평가한다. 각 통신 시스템의 MAC 규약을 살펴보고 시간 해석에 관한 분석을 제시한다. 본 논문에서는 평균 지연 시간, 네트워크 이용률, 전송 성공 확률 등을 성능분석의 지표로 사용한다. 이를 이용하여 각 통신 규약에 대해 시뮬레이션 결과를 비교, 분석한다.

1. 서 론

네트워크 기반의 분산 제어 시스템은 홈 자동화를 위한 제어 시스템에도 사용되고 있다. 일반적으로 데이터나 A/V 통신망은 많은 패킷이 상대적으로 비주기적으로 전송되게 되지만, 이에 반해 제어 통신망은 데이터 양이 작지만 상대적으로 많은 노드를 가지게 되며, 시간 제한 특성이 요구된다.

제어 네트워크 시스템에 요구되는 성능 평가 지표에는 접근 지연 시간, 전송 시간, 응답 시간, 데이터 지연 시간, 데이터 충돌, 네트워크 이용률 등이 있다 [1]. 제어 시스템에서는 한정된 지연 시간과 보장된 전송의 주요한 두 가지 기준을 만족시켜야 한다. 메시지가 한정된 지연 시간 내에서 성공적으로 전송되어야 한다는 점이 가장 중요하다고 볼 수 있다 [2]. 이러한 제어 시스템에 요구되는 지표는 홈 자동화를 위해 제안된 규약에도 마찬가지로 적용된다. 지금까지 제안된 규약을 살펴보면, Ethernet(IEEE 802.3:CSMA/CD), X10, Lonworks, CEBus등이 있다 [3][4][5][6].

Ethernet, Lonworks, CEBus의 성능 평가에 대한 연구가 있었지만[7][8][9], 개별적인 규약에 관한 성능 평가만을 기술하였을 뿐, 각 규약의 장단점을 언급한 바가 없다. 또, 옥내라는 특성을 시간해석에 적용한 사례도 찾아 보기 어렵다.

본 논문에서는 Ethernet (CSMA/CD), Lonworks, CEBus의 성능을 옥내라는 특성을 적용하여 비교 분석하기 위한 성능 해석 지표를 제안한다. 2장에서는 각 통신 규약의 MAC 매커니즘에 대하여 기술하고 3장에서는 성능분석을 위한 시간해석에 대하여 설명한다. 4장에서는 위의 자료를 토대로 시뮬레이션한 결과를 보이고 향후 연구와 함께 결론을 맺는다.

2. 제어 통신망의 소개

본 장에서는 세 가지 홈 네트워크를 위한 제어 통신망의 MAC 규약에 대하여 살펴보기로 한다. MAC은 네트워크 전체의 응답시간 제한을 만족시킬 수 있는 알고리즘을 제시하고, 네트워크 노드간의 통신 신뢰성을 제공해야 하는 역할을 담당한다.

2.1 Ethernet(CSMA/CD)

Ethernet은 MAC 규약으로 CSMA/CD(Carrier Sense Multiple Access with Collision Detection)을 사용한다 [7]. 어떤 노드가 전송을 하기를 원하면, 그 노드는 네트워크 상황을 살펴본다. 만약에 네트워크가 사용중이면, 네트워크가 휴지상태가 될 때까지 대기한다. 이 때, 물 이

상의 노드가 대기상태에서 전송모드로 전송을 시작하면 충돌이 발생한다. 충돌이 발생했다는 사실을 발견하면, 전송을 하고 있던 노드를 랜덤시간만큼 대기한 후 재전송하게 된다. 이 랜덤타임은 표준 BEB(Binary Exponential Backoff)를 통해 구해진다. 재전송하기 전의 시간은 0과 $(2^I - 1)$ 사이에서 랜덤하게 선택된다. (I : 노드에서 발견한 충돌 메시지의 순서) 그러나, 16번의 충돌이 발생되면, 컨트롤러는 전송시도를 중단하고 노드의 마이크로 컨트롤러에 전송 실패 사실을 통보한다. Ethernet의 프레임 구조는 그림 1과 같다.

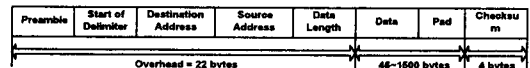


그림 1 Ethernet (CSMA/CD) 프레임 구조

2.2 Lonworks

Lonworks의 매체접근제어 규약, 즉 Lontalk는 Predictive p-persistent CSMA(Carrier Sense Multiple Access)를 사용한다. 네트워크 load를 예상하고 랜덤하게 접근함으로써 접근 회피를 이루는 방법이다.

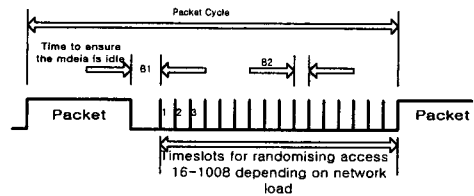


그림 2 Predictive p-persistent CSMA protocol

그림 2는 b1 주기의 단일 패킷 주기와 랜덤한 시간 슬롯의 구조를 보여주고 있다. 한 노드가 우선 순위 없이 전송을 원하면 버스의 상태를 살펴본다. 버스가 사용중이면, 노드는 버스의 휴지기가 될 때까지 기다린다. 버스가 사용중이지 않으면, 노드는 랜덤 시간 슬롯만큼 대기한다. 랜덤시간만큼 대기한 후에도 버스가 사용중이지 않으면 전송을 시작한다. 그렇지 않으면, 노드는 버스 접근을 다시 시도한다. 따라서, 한 노드는 주어진 랜덤 시간 슬롯 내에서 확률 p로 한 패킷을 전송하게 된다. p는 네트워크 데이터 양에 따라서 동적으로 조정된다. 데이터가 많은 경우에는 랜덤 슬롯을 증가시켜 충돌을 감소시켜 충돌이 발생할 확률이 작아진다. 랜덤 타임 슬롯의 수는 16개부터 worst case에 최대 1008개까지 가변적이다. 네트워크 데이터 양이 증가함에 따라 각 노드들은 랜덤 슬롯의 수를 증가시키기 위해 노드들의 backlog 상태를 상호 교환한다. Overload상태에서 네트워크는 최대 전송 용량을 계속 사용한다. 매체의 throughput은 충돌로 인해 감소되지 않는다. 이 알고리즘은 충돌의 횟수를 감소시켰지만, 둘 상의 노드가 동일한 랜덤 슬롯 타임에 전송하여 충돌이 발생하는 것을 완전히 제거한 것은 아니다. 그림 3은 Lonworks의 프레임 구조를 보여주고

있다.

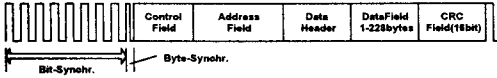


그림 3 Lonworks의 프레임 구조

2.3 CEBus

CEBus는 CSMA/CDCR(Carrier Sense, Multiple Access/with Collision Detection and Collision Resolution)을 MAC 규약으로 사용한다. 전송하기 전 각 노드는 버스가 사용 중인지 살펴본다. 사용중이면 사용이 끝날 때까지 대기한다. 사용이 끝나면, 노드는 전송하기 전 일정 시간을 대기하고 전송을 시작한다. 전송 전에 대기하는 시간은 전송하려는 패킷의 우선 순위와 랜덤하게 결정되는 채널 접근 지연시간에 따라 다르다. 랜덤하게 결정되는 지연시간을 무시하고 생각할 때, 두 노드가 동시에 버스가 사용 가능하다는 사실을 인식하게 되고, 전송하려는 패킷의 우선 순위가 같으면 두 노드 모두 전송을 시도하게 되고 충돌이 발생한다. 전송이 시작될 때, 각 노드는 Preamble을 먼저 보내게 된다. Preamble이 충돌 없이 전달되었다는 사실을 확인한 후, 패킷의 나머지 부분도 전송하게 된다. 만약 Preamble의 전송 시 충돌이 일어나면, 전송은 중단되고, 프로세서가 다시 시작된다. 그림 4은 CEBus의 프레임의 구조를 보여주고 있다.

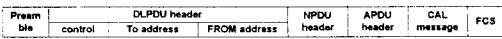


그림 4 CEBus의 프레임 구조

3. 제어 통신망의 성능 분석

제어 통신망의 성능 분석을 실시한다. 제어 통신망은 다음과 같은 파라미터를 가질 수 있다.

표 1 네트워크 성능 분석을 위한 파라미터

N_{data}	데이터 크기(bytes)	T_{frame}	네트워크 매체에서의 Frame 시간
N_{ovhd}	오버헤드 크기(bytes)	T_{max}	최대 시뮬레이션 시간
N_{pad}	padding 프레임(bytes)	T_{prop}	전송 지연 시간
N_{stuff}	Stuffing 프레임(bytes)	T_{queue}	소스 노드에서의 Queuing 시간
P_{util}	네트워크 이용률(%)	T_{resid}	Residual 시간
T_{block}	소스노드에서의 Blocking 시간	T_{tx}	전송 시간
T_{bit}	Bit Time	T_{wait}	소스 노드에서의 대기 시간
T_d	데이터의 지연 시간	T_{peri}	데이터 주기

3.1 평균 지연 시간

T_{max} = 10 s의 동작시간으로 각 통신망에 대해 각 노드에서의 평균 지연시간을 계산하였다.

$$T_d^{avg} = \frac{1}{N} \sum_{i \in N_{node}} \left[\frac{\sum_{j=1}^{M^{(i)}} T_d^{(i,j)}}{M^{(i)}} \right] \quad (1)$$

$$M^{(i)} = \lfloor T_{max} / T_{peri} \rfloor \quad (2)$$

N 은 총 노드의 수이며, N_{node} 는 노드의 집합이며,

$M^{(i)}$ 는 노드 i 에서 요청 받은 데이터의 개수이다. 본 논문에서는 모든 데이터는 주기적으로, 총 데이터의 개

수는 데이터의 주기로 나눈 값으로 가정한다. 즉, 식(2)와 같다. 평균지연시간은 식(1)에서처럼 각 노드에 대해서도 구할 수 있다.

각 노드에서의 지연시간 $T_d \approx T_{wait} + T_{tx}$ 로 근사화할 수 있으며, 이때, T_{wait} 는 큐잉 시간 T_{queue} 와 블로킹 시간 T_{block} 의 합으로 표현된다. T_{queue} 는 전송하기 전 소스 노드에서 대기하는 지연시간이며, 이는 큐의 앞선 데이터의 블로킹 시간과 그 데이터의 우선 순위에 의해 결정된다. T_{block} 은 각 통신망의 충돌회피기법과 재전송 알고리즘 등에 따라 달라지며, 이를 정리하면 다음과 같다.

(Ethernet Blocking Time)

$$E(T_{block}) = [(g_1 + g_2 + \dots + g_{15})Q_r + g_{16}]T_{slot} + T_{resid} \quad (3)$$

g_k : 데이터 전송을 요청 받은 시간부터 k 번째 backoff시간까지의 경과시간

Q_r : 현재 노드가 주어진 슬롯에서 backlogged 데이터를 전송하려고 시도하는 유일한 노드일 확률

T_{slot} : Round-trip 전송에 요구되는 시간

T_{resid} : 노드 1에서 네트워크가 idle하다는 사실을 알 때 까지의 경과시간

(Lonworks Blocking Time)

$$E(T_{block}) = \sum_{j=N_{max}}^{k-1} \left[\frac{T_{block}^{(k-1)} + T_{bit}}{T_{peri}^{(j)}} \right] T_{tx} + T_{resid} \quad (4)$$

(CEBus Blocking Time)

$$E(T_{block}) = [w_{queued}Q_r + w_{unqueued}]T_{slot} + T_{resid} \quad (5)$$

식(5)에서 w_{queued} 는 데이터의 우선 순위가 queued일 때의 대기 시간을 나타내며, $w_{unqueued}$ 는 데이터의 우선 순위가 unqueued일 때의 대기 시간을 나타낸다. CEBus 표준은 HIGH, STANDARD, DEFERRED의 세 가지 우선 순위를 정의하고 있으며, w_{queued} 와 $w_{unqueued}$ 는 이 우선 순위에 따라 그 값이 다르다. HIGH에 경우 queued에서는 4-8 UST(Unit Symbol Times)의 대기 시간을 가지며, queued에서 충돌이 발생하여, unqueued가 되면 8-12 UST의 대기 시간을 갖는다.

T_{tx} 는 전송에 소요되는 시간을 나타내며, $T_{tx} = T_{frame} + T_{prop}$ 로 표현된다. (T_{frame} 은 프레임 시간, T_{prop} 은 전송지연 시간을 나타낸다.) T_{frame} 은 데이터의 크기, 오버헤드, 패딩, 전송률에 의해 결정된다. 이를 식으로 나타내면 다음과 같다.

$$T_{frame} = [N_{data} + N_{ovhd} + N_{pad} + N_{stuff}] \times 8 \times T_{bit}$$

T_{prop} 은 전송률과 소스로부터 목적지까지의 실제거리에 의해 결정된다. 본 논문에서 각 통신망에서 최대 허용 전송 지연시간은 무시한다. 옥내의 상황을 고려할 때, 소스로부터 목적지까지는 수 미터 내외로 전송에 소요되는 시간에 거의 영향을 미치지 않는다.

3.2 통신망의 이용률

통신망의 이용률 P_{util} 은 전체 동작 시간과 데이터 전송에 사용된 전체 시간의 비율로 나타난다. 즉, 식(6)과 같다.

$$P_{util} = \frac{\sum_{i \in N_{node}} \sum_{j=1}^{M^{(i)}} T_{tx}^{(i,j)}}{T_{sum}^{(d)}} \quad (6)$$

3.3 전송 실패 패킷의 수

제어 통신망에서는 성공적으로 전송이 이루어졌다는 정보를 즉시 알아야 한다. 이러한 정보가 원활히 전달되

지 않으면, 네트워크 전체의 성능이 저하되고 시스템이 unstable해 질 수 있다. 따라서 전송이 실패한 패킷의 수도 네트워크 구약을 평가하는데 중요한 요소이다.

4. 시뮬레이션 결과

4.1 시뮬레이션 시나리오

각 통신망에 대해 5개의 노드를 가진 시스템을 고려하였다. 각 노드는 매 주기마다 8바이트의 데이터를 전송하도록 하였고 시뮬레이션은 각각 0.5 sec까지 동작하도록 하였다. 데이터 전송을 시작할 때, 모든 노드가 t=0일 때 전송을 시작하도록 하였다. 이후에는 각각의 알고리즘에 따라 결정되는 랜덤한 대기 시간을 가지고 전송이 이루어지게 된다. CEBus의 경우 우선 순위 데이터의 비율을 달리하여서 시뮬레이션 하였다.

4.2 시뮬레이션 결과

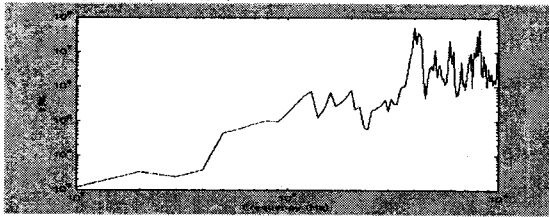


그림 5 Ethernet의 평균 지연 시간

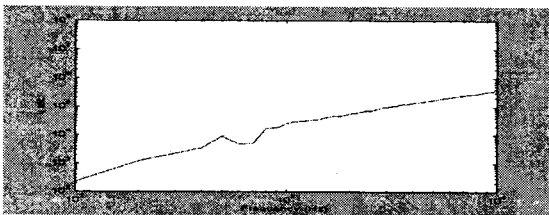


그림 6 Lonworks의 평균 지연 시간

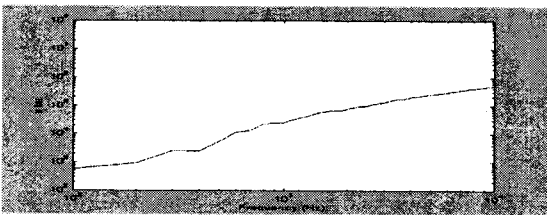


그림 7 CEBus의 평균 지연 시간

(HIGH: 20%, STANDARD: 40%, DEFERRED: 40%)

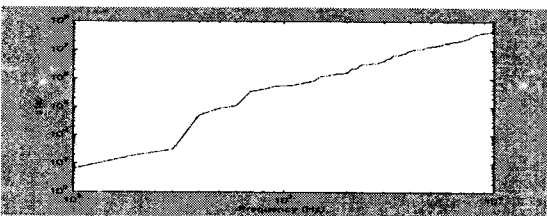


그림 8 CEBus의 평균 지연 시간

(HIGH: 80%, STANDARD: 10%, DEFERRED: 10%)

그림 5에서 Ethernet은 어느 순간 이후부터는 데이터 양의 증가에 따라 평균 지연 시간이 급격히 증가함을 알 수 있는데, 이는 Ethernet의 BEB 알고리즘으로 인해 지연시간이 기하급수적으로 증가하기 때문에 나타나는 현상으로 보인다. 또, Ethernet의 경우 데이터 양이 증가

함에 따라 전송되지 못한 메시지의 수도 어느 수준까지 증가하는 현상을 나타냈다. 이러한 결과로 볼 때 Ethernet의 안정도는 홈 오토메이션을 제어 통신망에 사용하기 부적합하다고 볼 수 있다.

그림 6,7에서 Lonworks와 CEBus는 비슷한 지연 시간을 나타내며, 네트워크가 어느 정도 포화상태에 이르면 Ethernet에 비해 안정적으로 지연시간이 일정한 수준에 머물러 있음을 알 수 있다. 그러나 그림 8에서 CEBus의 경우 HIGH 우선 순위의 데이터의 비율이 높고, 그 양이 증가함에 따라 지연시간이 기하급수적으로 증가하는 모습을 볼 수 있다. 즉, CEBus는 데이터의 우선 순위의 비율에 따라 성능이 크게 좌우되는 현상을 살펴볼 수 있다. 결과적으로 일반적으로 사용되는 Ethernet에 비해 홈 오토메이션을 위한 통신 규약인 Lonworks와 CEBus가 보다 안정적인 성능을 나타낸다는 사실을 알 수 있다.

5. 결 론

본 논문에서는 홈 오토메이션 용 제어 통신망의 성능을 비교 분석하였다. 성능 평가를 위한 시간 해석 결과를 제시하였으며, 시뮬레이션을 통한 지연시간을 통한 성능 분석 결과를 보였다. 일반적으로 쓰이는 Ethernet의 경우 보다 홈 오토메이션을 위한 통신 규약인 Lonworks와 CEBus가 보다 안정적인 성능을 내는 것으로 결과가 나타났다.

향후 연구 과제로는 실제 구현한 장치를 이용한 실험과 이를 바탕으로 한 스케줄링 방법에 대한 연구 등이 필요하다.

[참고 문헌]

- [1] Feng-Li Lian, Moyne, J.R., Tilbury, D.M, "Performance evaluation of control networks: Ethernet, ControlNet, and DeviceNet", IEEE Control Systems Magazine, Volume: 21 Issue: 1, Pages: 66 -83, Feb. 2001
- [2] Feng-Li Lian, Moyne, J.R., Tilbury, D.M, "Performance evaluation of control networks for manufacturing systems", Proceeding of the ASME Dynamic Systems and Control Division, DSC-Vol 67., Nov. 1999
- [3] Hubertus Schweins, Donal Heffernan, "Retrofitting a deterministic access control policy to a Non-deterministic Control Network", ISSC Conference 1998, Jun, 1998
- [4] Jaesoo Yang, Constantine N.Manikopoulos, "Performance comparisons of the CEBus with other protocols", IEEE Transactions on Consumer Electronics, Vol. 39, No. 4, Nov, 1993
- [5] Altaf Hussain, Anthony D. Robbi, "Delay performances of standard and modified CEBus schemes", IEEE Transactions on Consumer Electronics, vol. 38, No.2, Pages: 77-84, May 1992
- [6] Echelon Corporation, Lontalk Protocol Specification Version 3.0, 1994
- [7] Brian R. Bertan, "Simulation of MAC-layer queueing and priority strategies of CEBus", IEEE Transactions on Consumer Electronics, vol. 35, No. 3, Pages: 557-563, Aug 1989
- [8] Sairaj R. Pakkam, Constantine N. Manikopoulos, "Performance evaluation of the consumer electronic bus", IEEE Transactions on Consumer Electronics, vol. 36, No 4, Pages: 949-953, Nov 1990
- [9] E. Vazquez J. Vinyes, T. Miguel, "Through analysis of p-CSMA based LonTalk protocol for building management systems", IEEE Conference on Consumer Electronics, 1996
- [10] Dimitri Bertsekas, Robert Gallager, Data Networks 2nd edition, Prentice Hall, 1992