

동기식 이더넷 시스템용 MAC 프레임 분할 방식과 효율적인 흐름제어 방식의 성능 분석

준희원 최 희 경*, 종신회원 윤 중 호*, 정희원 조 재 현**

Performance of MAC Frame Fragmentation and Efficient Flow Control Schemes for Synchronous Ethernet Systems

Hee-Kyoung Choi* *Associate Member*, Chong-Ho Yoon* *Lifelong Member*
Jae-Hun Cho** *Reguler Member*

요 약

본 논문은 동기식 이더넷 시스템의 구현 시 고려해야 할 다음과 같은 두 가지의 사항에 대한 해결방안을 제시한 것이다. 첫째, 실시간 및 비실시간 트래픽을 모두 지원하기 위하여 일정주기의 사이클 기반으로 동작하는 동기식 이더넷 시스템은 이미 송신이 진행 중인 비실시간 프레임 때문에, 사이클의 시작 시점이 지연됨으로써 실시간 트래픽의 전송이 지연되는 문제점을 가지고 있다. 이 점에 착안하여, 기존 방식의 지터 성능을 분석한 후, 다음 사이클의 시작 시점을 참조하여 비실시간 프레임을 여러 사이클에 분할하여 전송함으로써 실시간 트래픽에 대하여 지터가 전혀 없는 엄격한 전송을 보장하는 방법을 제안하고, 또 다른 지터 억제 방식인 전송보류(hold) 방식과의 프레임 전송지연 특성을 비교 분석하였다. 두 번째는 한 사이클 내의 실시간 트래픽 전송 구간에는 비실시간 프레임을 전송할 수 없기 때문에, 신속히 송신되어야 할 링크 계층에서의 흐름제어용 IEEE 802.3x 중지(pause) 프레임의 전송이 지연됨으로써 과도한 비실시간 프레임의 손실을 초래하는 문제점이 있다. 이 점에 착안하여 실시간 트래픽 전송 구간에서도 흐름제어용 프레임의 전송을 허용하도록 하는 새로운 흐름제어 방법을 제안하고 이에 따른 비실시간 트래픽의 손실률을 분석하였다.

Key Words : Synchronous Ethernet, Superframe, Fragmentation, IEEE 802.3x pause frame.

ABSTRACT

In this paper, we consider the following two issues for implementing the synchronous Ethernet systems. First, a synchronous Ethernet system employs a fixed size superframe which is divided into a synchronous period and an asynchronous one. We note that the starting point of a superframe is not deterministic when an ordinary data frame's transmission is overlapped the superframe boundary. This overlap may be a fatal drawback for strict jitter bounded applications. Circumventing the problem, we propose a frame fragmentation scheme to provide a zero jitter, and compare its delay performance with the hold scheme which also provides the zero jitter. We next concern that IEEE 802.3x pause frames cannot be promptly transmitted in a synchronous period, and thus asynchronous traffics may be dramatically get dropped at the input buffer of a switch. To handle the problem, we propose an efficient flow control by allowing the transmission of the pause frame in a synchronous period, and investigate the blocking probability of the asynchronous traffics by the simulation.

※본 논문은 한국정보통신대학원 BcN 엔지니어링 연구센터와 삼성전자의 지원에 의하여 수행되었음.

* 한국항공대학교 정보통신공학과 응용네트워크 연구실(matrixhk@hau.ac.kr, yoonch@hau.ac.kr),

** 삼성전자 통신연구소 (jaehun.cho@samsung.com)

논문번호 : KICS2005-04-168, 접수일자 : 2005년 4월 19일

I. 서론

대부분의 근거리 통신망에서 사용되는 이더넷은 랜덤한 전송 지연 시간과 캡처 현상에 의한 불공정성 문제 때문에, 엄격한 지터 제약 조건이 요구되는 실시간 서비스를 제공하는데 문제가 있다^{[1][2]}. 이러한 문제를 해결하기 위하여 우선순위 방식, 이더넷 상에서의 토큰 패싱 방법, 시분할 다중화 슬롯 방식 등이 제안되고 있다^{[4][7][10][12]}. 또한, 슈퍼프레임(Superframe)이라고 불리는 고정된 사이클 기간을 분할하여 실시간 및 비실시간 트래픽들을 함께 지원할 수 있는 김슨사의 동기식 이더넷 표준안, 그리고 지멘스사의 ProfiNet 표준안 등도 발표되고 있다^{[3][5]}.

이들 중에서 김슨사의 동기식 이더넷 표준안은 이더넷에서의 서비스 품질(Quality of Service)을 제공하기 위하여 슬롯 기반의 슈퍼프레임 방식으로 동작한다. 최근 IEEE 802.3에서는 이러한 동기식 이더넷 기술에 대한 표준화를 진행하기 위하여 레지던셜 이더넷 스터디 그룹을 구성하였다^[13].

슈퍼프레임 방식으로 동작하는 동기식 이더넷 시스템의 구현 시에는 다음과 같은 두 가지 문제점을 고려해볼 수 있다. 첫째, 동기식 이더넷에서는 실시간 트래픽의 전송을 위한 동기 구간과 비실시간 트래픽의 전송을 위한 비동기 구간으로 구성된 슈퍼프레임이 1기가비트 이더넷일 경우, 125 μ sec의 주기로 사용된다. 하지만, 이러한 슈퍼프레임 방식에서는 한 슈퍼프레임 기간의 비동기 구간에서 이미 송신이 진행중인 비실시간 프레임의 전송이 비동기 구간 내에 끝나지 못함으로 인해서 다음 슈퍼프레임의 시작 시간이 지연되는 문제가 발생할 수 있다. 이러한 문제 때문에, 슈퍼프레임 방식은 수 μ sec 이내의 엄격한 지터가 요구되는 공장 자동화 시스템이나 데이터 수집 장치 등에서는 사용할 수 없다.

두 번째로는, 전이중 전송 모드 방식으로 동작하는 동기식 이더넷은 비실시간 트래픽의 흐름제어시 전송되는 IEEE 802.3x 중지(pause)프레임도 수용해야 한다^[14]. 하지만 현재 고려되고 있는 동기식 이더넷 시스템에서는, 동기 구간에서 중지프레임의 전송이 필요한 이벤트가 발생하더라도 이 중지프레임의 송신은 동기 구간이 끝날 때까지 지연될 수밖에 없기 때문에, 비실시간 트래픽의 손실률이 증가되는 문제가 발생한다. 이것은 동기 구간에서 비실시간 트래픽에 대한 대역을 할당하지 않았기 때문에, 흐름제어용인 중지프레임과 같은 비실시간 프레임이 전송되지 못하므로 발생하는 현상이다.

본 논문에서는 이러한 두 가지 문제점을 해결하기 위하여, 실시간 서비스를 위한 다양한 이더넷 방식 중에서 최근에 발표된 슬롯 기반의 동기식 이더넷 초안에 근거하여, 해결 방안을 제안하고 성능을 분석한다. 첫 번째로는, 이미 송신이 진행 중인 비실시간 프레임 때문에 실시간 트래픽의 전송이 지연되는 문제점을 해결하기 위하여 기존 방식의 지터 성능을 분석한 후, 다음 사이클의 시작 시점을 참조하여 비실시간 프레임을 여러 사이클에 분할하여 전송하는 방법을 제안하고, 또 다른 지터 억제 방식인 전송보류 방식과의 프레임 처리시간 및 대역폭 활용도를 비교하였다.

두 번째는 한 사이클 내의 실시간 트래픽 전송 구간에는 중지프레임과 같은 비실시간 프레임을 전송할 수 없기 때문에 과도한 비실시간 프레임의 손실이 발생하는 문제점을 해결하기 위한 것으로써, 실시간 트래픽 전송 구간에서도 흐름제어용 중지프레임의 전송을 허용하도록 하는 새로운 흐름제어 방법을 제안하고 이에 따른 비실시간 트래픽의 손실률을 분석하였다.

본 서론에 이어서, 제 2장에서는 동기식 이더넷의 슈퍼프레임 방식의 문제점과 해결 방법을 기술하고, 제 3장에서는 흐름제어 방식의 문제점과 이에 대한 해결 방법을 기술한다. 제 4장에서는 제안된 방법의 프레임 처리시간 및 대역폭 활용도, 비실시간 트래픽의 손실률을 모의실험을 통하여 분석하고, 마지막으로 제 5장에서는 결론을 맺는다.

II. 엄격한 실시간 서비스를 보장하는 프레임 분할 방식

2.1 동기식 이더넷의 기본 동작

슬롯 기반의 동기식 이더넷은 1기가비트 이더넷 상에서 <그림 1>과 같이 사이클이 125 μ sec의 주기로 반복된다. 한 사이클 내에는 192개의 4바이트짜리 실시간 데이터 슬롯들을 이더넷 프레임에 수납한 실시간 프레임을 최대 16개 전송할 수 있다. 그리고 이러한 실시간 프레임 전송이 사이클 내에서 완료되면, 남은 기간 동안에 비실시간 프레임들을 전송할 수 있다. 실시간 트래픽이 전송되는 구간을 동기 구간이라 하며, 비실시간 트래픽이 전송되는 구간은 비동기 구간이라 한다. 또한, 동기 구간과 비동기 구간으로 구성된 하나의 사이클을 슈퍼프레임이라 한다. 이러한 슈퍼프레임의 시작은 첫 번째 실시간 프레임의 동기 헤더에 있는 1로 설정된 동

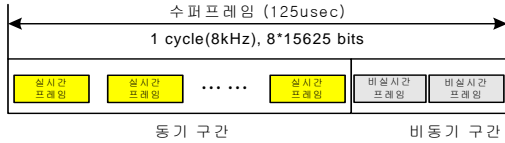


그림 1. 동기식 이더넷의 수퍼프레임 구성

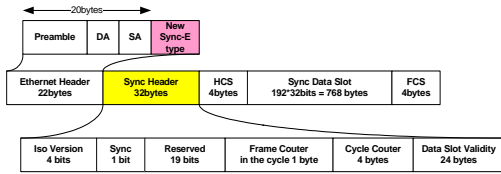


그림 2. 동기식 이더넷의 실시간 프레임 구조

기비트로 지시한다.

동기 구간에 전송되는 실시간 프레임의 형식은 <그림 2>와 같다. 이더넷 헤더 22바이트와 동기 헤더 32바이트, 헤더 체크섬 4바이트, 192개의 4바이트짜리 실시간 데이터 슬롯, 프레임 체크 시퀀스 4바이트로 구성된다. 또한, 동기 헤더에는 버전 정보 비트와 수퍼프레임의 시작을 표시하는 동기비트, 한 수퍼프레임안의 실시간 프레임 수를 나타내는 카운터, 수퍼프레임의 카운터, 실시간 데이터 슬롯의 유효성을 표시하는 비트열과 예약 비트로 구성된다.

이러한 실시간 프레임은 새로운 이더넷 타입 값으로 식별된다. 반면에 비실시간 프레임은 기존의 이더넷 프레임 형식과 동일하다.

2.2 수퍼프레임 방식의 문제점

표준안에 따르면, 한 수퍼프레임의 비동기 구간에 비실시간 프레임이 적어도 한 개 이상 전송될 수 있도록 하기 위해서 한 수퍼프레임 내의 실시간 프레임의 개수는 최대 16개로 제한된다. 이 경우, 비동기 구간은 최대 2153바이트의 비실시간 프레임이 전송될 수 있는 길이를 갖는다.

하지만, 현재 고려되고 있는 수퍼프레임 방식의 동기식 이더넷은 이러한 비동기 구간에서 이미 전송이 진행 중인 비실시간 프레임으로 인하여 다음 수퍼프레임의 시작이 지연될 수 있는 문제점이 있다. 최악의 경우, 수퍼프레임의 지연 시간은 이더넷 최대 프레임 길이인 1518바이트의 전송 시간 동안 지연될 수도 있다. 이렇게 비실시간 프레임으로 인해 수퍼프레임의 시작이 빈번하게 지켜지지 않을 경우, 엄격한 실시간 서비스를 보장할 수 없다. 특히 수퍼프레임의 시작이 지연되는 경우, 다음 수퍼프레임의 동기 구간이 축소되는 문제도 발생할 수

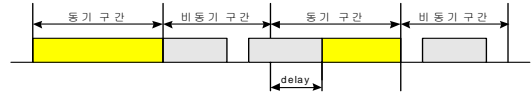
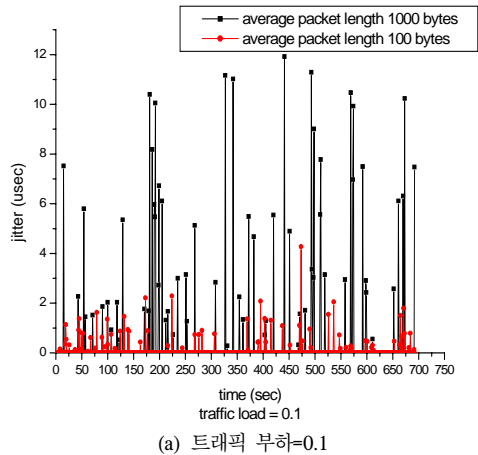


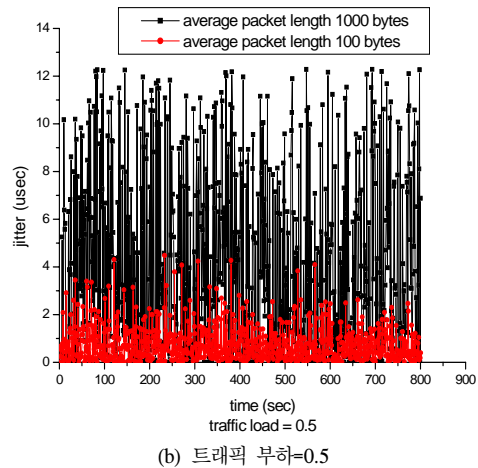
그림 3. 수퍼프레임의 지연과 동기 구간의 축소

있다. <그림 3>은 이미 전송이 진행 중인 비실시간 프레임으로 인하여 다음 수퍼프레임의 시작이 지연되는 것과 이로 인해 동기 구간이 축소되는 예를 도시한 것이다.

이러한 수퍼프레임의 지연 시간 특성을 알아보기 위하여, 본 논문에서는 모의실험을 통하여 <그림 4>와 같은 결과를 도출하였다. 수퍼프레임의 최대 지연 시간은 비실시간 프레임의 최대 길이만큼 지연될 수도 있으며, 비실시간 트래픽의 부하가 클수록 이러한 지연이 빈번하게 발생하는 것을 확인할 수 있다.



(a) 트래픽 부하=0.1



(b) 트래픽 부하=0.5

그림 4. 비실시간 트래픽 부하에 따른 수퍼프레임의 지연 시간 변화

2.3 엄격한 실시간 서비스를 보장하는 프레임 분할 방식

이미 전송이 진행중인 비실시간 프레임으로 인하여 수퍼프레임의 시작이 지연되어 지터가 발생하는 문제를 해결하기 위한 가장 단순한 방법은, 다음 수퍼프레임의 예상 시작 시간까지 단말이 전송해야 할 비실시간 프레임의 전송이 완료될 수 없는 프레임 길이를 가진 경우, 해당 단말은 이 프레임의 전송을 유보하고 다음 수퍼프레임의 비동기 구간 시작 시점에 이 프레임을 전송하는 것이다. <그림 5>와 같이 전송보류(hold)하는 방식은 엄격하게 수퍼프레임의 시작을 보장할 수 있는 장점이 있지만, 대역을 낭비하는 문제점이 있다.

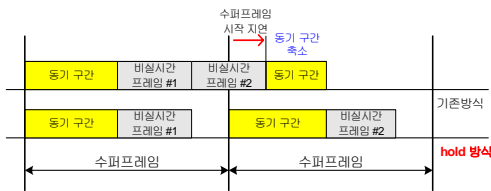


그림 5. 전송보류 방식

본 논문에서는 이러한 전송보류 방식의 대역 낭비 문제를 해결하면서도, 엄격하게 수퍼프레임의 시작을 보장할 수 있는 링크계층에서의 프레임 분할 방식을 제안한다. 프레임 분할 방식은 <그림 6>과 같이 전송해야 할 비실시간 프레임에 대하여, 남은 비동기 구간에 맞게 프레임을 분할하여 전송하고, 프레임의 나머지 부분은 다음 수퍼프레임의 비동기 구간에 전송하는 방식이다.

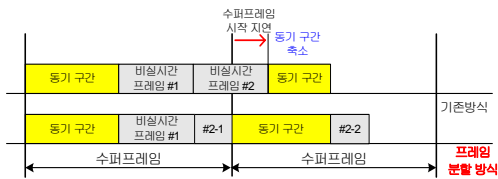


그림 6. 프레임 분할 방식

프레임 분할 방식은 IP 계층에서 수행되는 IP 분할 방식과 유사하지만, <그림 7>의 순서도처럼 다음과 같은 조건 하에 실행된다. 남은 비동기 구간이 이더넷 헤더 22바이트와 IFG(Inter Frame Gap) 12바이트, 프레임 체크 시퀀스 4바이트, 그리고 프레임 분할 정보를 나타내는 2바이트, 총 40바이트를 보장할 수 있는 전송 시간 보다 클 경우에만 프레임을 분할하여 전송하고, 남은 비동기 구간이 40바

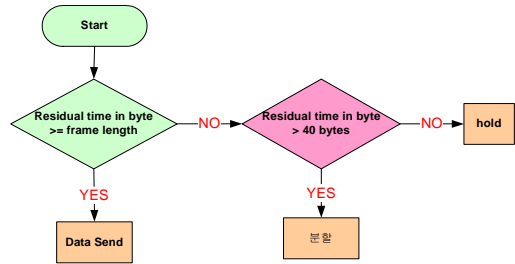


그림 7. 프레임 분할 방식의 순서도

이트의 전송 시간을 보장하지 못 할 경우에는 프레임을 전송하지 않고 다음 수퍼프레임의 비동기 구간에 전송하는 방식이다.

<그림 8>은 분할된 비실시간 프레임의 구조를 자세히 도시한 것이다. 프레임을 분할 할 경우에는, 분할된 프레임의 새로운 이더넷 타입 값을 규정하며, 분할된 정보를 나타내기 위해 2바이트의 분할 정보(Fragmentation Control)영역을 추가한다. 또한, 수신측에서의 재조립을 위하여 분할된 나머지 프레임의 영역에 대해서는 분할되기 이전의 프레임에 부착된 이더넷 헤더와 분할 정보 2바이트를 재사용한다. 분할 정보의 2바이트는 프레임이 분할되어 전송됨을 나타내기 위한 *more* 1비트와 *sequence* 15비트로 구성된다. 즉, 동일한 순서(sequence) 번호를 가지며, 분할된 프레임의 마지막 영역을 제외한 프레임들은 모두 *more* 비트를 1로 설정하여 전송하고, 마지막 프레임의 경우에만 *more* 비트를 0으로 설정하여 수신측 MAC에서 이 프레임을 재조립하여 상위 계층으로 전달할 수 있도록 한다. 이렇게 함으로써 엄격한 사이클을 제공할 뿐만 아니라, 낭비되는 대역을 감소시킬 수 있다. <그림 9>는 분할된 프레임의 분할 정보 영역의 상세를 도시한 것이다.

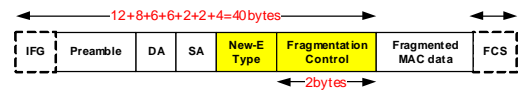


그림 8. 분할된 비실시간 프레임

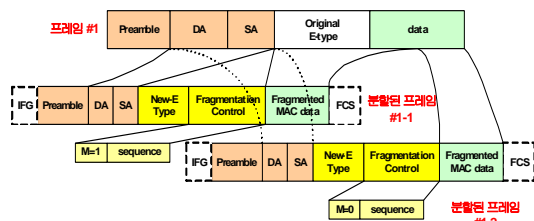


그림 9. 분할된 프레임의 분할 정보 영역의 구조

III. 효율적인 흐름제어 방식

3.1 링크계층에서의 명시적인 흐름제어 과정

기존의 전이중 이더넷에서는 링크계층에서의 흐름제어를 위하여 명시적인 흐름제어용 제어프레임인 중지(pause)프레임을 사용한다¹⁴⁾¹⁵⁾. 중지프레임은 상대방으로부터 수신되는 프레임의 양보다 자신이 처리하는 속도가 느려서 수신 버퍼의 임계치에 도달했을 때, 상대방의 프레임 송신을 일정시간 즉, 중지타이머 기간 동안 중지시키기 위한 목적으로 송신된다. 중지프레임은 중지타이머 기간을 지시하는 중지기간(pause duration)필드를 포함하며, 중지기간은 512 비트 단위로 0부터 65535의 값을 가진다. 1기가비트 이더넷일 경우, 실제 중지타이머의 최대값은 33.6msec이다¹⁶⁾.

<그림 10>은 이러한 흐름제어 방식이 동기식 이더넷 스위치에 적용될 경우의 비실시간 트래픽에 대한 흐름제어 방법을 도시하고 있다. 동기식 이더넷 스위치가 여러 개의 입력 포트로 비실시간 프레임을 수신하여 하나의 업링크로 중계할 때, 과도한 비실시간 프레임 수신에 의해 비실시간 수신 버퍼의 임계치가 초과되는 일이 발생하게 된다. 이러한 경우, 동기식 이더넷 스위치는 해당 단말들에게 다운링크를 통해 중지프레임을 송신하여, 해당 단말로부터 중지타이머가 만기될 때까지 송신이 중지되도록 해야 한다.

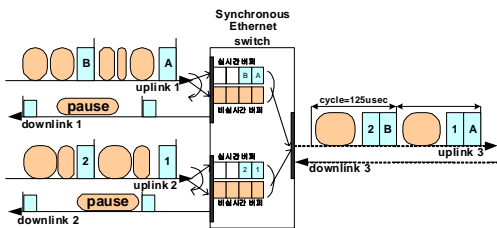


그림 10. 동기식 이더넷 스위치에 적용된 명시적인 흐름제어 과정

3.2 동기식 이더넷의 흐름제어 방식의 문제점

현재 고려되고 있는 동기식 이더넷의 흐름제어의 문제점은 비실시간 트래픽의 흐름제어용 중지프레임의 전송이 수퍼프레임의 비실시간 트래픽 전송 구간인 비동기 구간에서만 가능하다는 점이다. 즉, 동기 구간에서는 비실시간 트래픽의 전송이 허용되지 않기 때문에 중지프레임의 전송이 불가능하다. 만약, 동기 구간에서 중지프레임의 전송이 요구되는 상황이 감지된다면, 기존의 비동기 전용 스위치와는 달

리 어쩔 수 없이 중지프레임의 전송은 비동기 구간의 시작 시점까지 지연되므로, 수신 버퍼에서 비실시간 프레임의 손실이 발생한다. 이것은 동기 구간에서 비실시간 트래픽에 대한 대역을 할당하지 않았기 때문에, 중지프레임과 같은 비실시간 프레임이 전송되지 못하므로 발생하는 현상이다.

<그림 11>은 이러한 동기식 이더넷에서의 비실시간 트래픽에 대한 흐름제어의 문제점을 도시한 것이다. 이 예에서 동기식 이더넷 스위치 포트 1의 비실시간 트래픽 수신 버퍼의 임계치 초과 이벤트가 발생한 시점은 다운링크 1의 동기 구간이다. 동기 구간에서는 중지프레임을 전송하지 못하기에, 다운링크 1의 동기 구간이 끝날 때까지 이 중지프레임의 전송은 지연되고, 그 후 다운링크 1의 비동기 구간이 시작될 때 전송된다. 따라서, 수신 버퍼의 임계치 초과 이벤트가 발생한 시점부터 중지프레임이 전송된 시간까지 수신 버퍼에서 비실시간 프레임은 버려지게 된다. 최악의 경우, 다운링크 1의 동기 구간의 시작 시점에서 임계치 초과 이벤트가 발생했다면 많은 양의 비실시간 프레임이 버려지게 될 수도 있다.

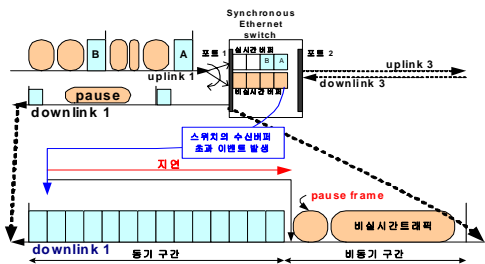


그림 11. 동기식 이더넷의 비효율적인 흐름제어

3.3 동기 구간에서의 중지프레임 전송 허용 방안

본 논문에서는 비실시간 트래픽의 효율적인 흐름제어를 위해서 동기 구간에서도 중지프레임을 전송할 수 있는 다음과 같은 방식을 제안한다. 즉, 동기 구간에 전송되는 실시간 데이터 슬롯 중 하나 이상의 슬롯을 비실시간 트래픽의 흐름제어를 위한 제어프레임 슬롯으로 사용하는 것을 허용함으로써 중지프레임이 가급적 신속하게 전송될 수 있도록 한다. 이렇게 동기 구간에서 비실시간 프레임인 제어프레임의 전송을 허용하는 것은 흐름제어를 통한 비실시간 트래픽의 손실률을 줄일 수 있으며, 또한 신속한 처리가 요구되는 OAM(Operation, Administration, and Management) 정보 또는 기타 제어용으로도 활용할 수 있다.

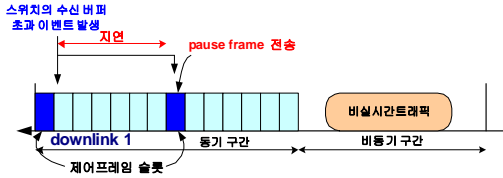


그림 12. 개선된 흐름 제어 방법

<그림 12>는 제안하는 흐름제어 기법을 도시한 것으로 다운링크 1의 동기 구간에서 두 개의 제어프레임 슬롯을 사용한 예이다. 수신 버퍼의 임계치 초과 이벤트가 다운링크 1의 동기 구간에서 감지됐을 때, 다운링크 1의 비동기 구간의 시작 시점까지 지연되는 대신에 동기 구간의 두 번째 제어프레임 슬롯에 중지프레임이 전송되는 것을 보여준다. 이것은 <그림 11>의 중지프레임 전송 지연 시간보다 감소된 것을 알 수 있으며, 감소된 지연 시간만큼 비실시간 트래픽의 손실률도 감소된다. 물론, 이러한 제어프레임 슬롯을 동기 구간에 많이 할당한다면, 중지프레임이 더욱 신속히 전송될 수 있기 때문에 비실시간 트래픽의 손실률도 더욱 감소할 것이다.

IV. 성능 분석

4.1 엄격한 실시간 서비스를 보장하는 동기식 이더넷의 성능 분석

본 논문에서는 슈퍼프레임의 시작 시간을 엄격하게 보장하기 위해 제안한 프레임 분할 방식의 성능을 분석하기 위하여, SIMULA를 사용한 모의실험을 통해 전송보류 방식과의 성능을 비교한다^[11]. 모의실험의 환경모델은 <그림 13>을 따르며, 사용한 파라미터는 <표 1>과 같다. 비실시간 프레임 길이는 지수분포를 가지며, 비실시간 프레임 발생은 포아송 프로세스에 따르도록 하였다.

<그림 14>는 본 논문에서 제안하는 프레임 분할 방식에 대하여 전송보류 방식과의 평균 프레임 처리 시간을 비교한 것이다. 슈퍼프레임 내의 동기 구간 비율(R_s)을 30%, 50%, 70%로 각각 변화시킬 때, 비실시간 트래픽 부하에 따른 평균 프레임 처리 시간을 도시한 것이다. 동기 구간이 30%이고 비실시간 트래픽의 부하가 0.68일 경우, 전송보류 방식의 평균 프레임 처리 시간은 약 72.19 μsec 이지만, 프레임 분할 방식은 약 54.80 μsec 으로써, 제안된 프레임 분할 방식이 전송보류 방식보다 평균 프레임 처리 시간 면에서 우수함을 알 수 있으며, 비실시간 트래픽의 부하가 커질수록 평균 프레임 처리

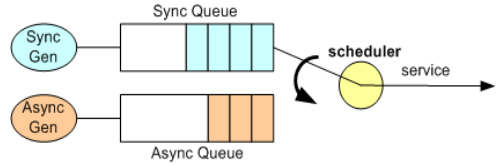


그림 13. 프레임 분할 방식 모의실험의 환경모델

표 1. 프레임 분할 방식 모의실험의 파라미터

| 파라미터 | 값 |
|----------------------------|---------------------|
| 대역폭 | 1 Gbps |
| 최대 큐 길이 | 100 |
| 수퍼프레임 길이 | 125 μsec |
| 최대 프레임 길이 | 1518 바이트 |
| 최소 프레임 길이 | 64 바이트 |
| 평균 프레임 길이 | 1250 바이트 |
| 수퍼프레임 내의 동기 구간 비율(R_s) | 30, 50, 70(%) |

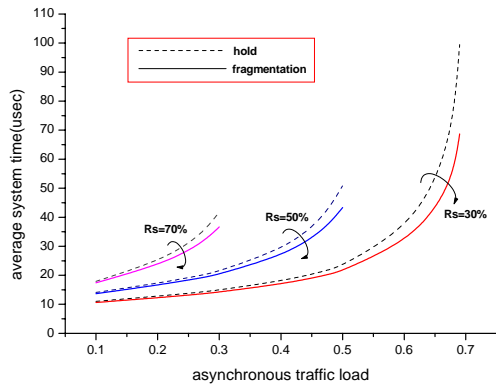


그림 14. 비실시간 트래픽부하에 따른 평균 프레임 처리 시간 비교

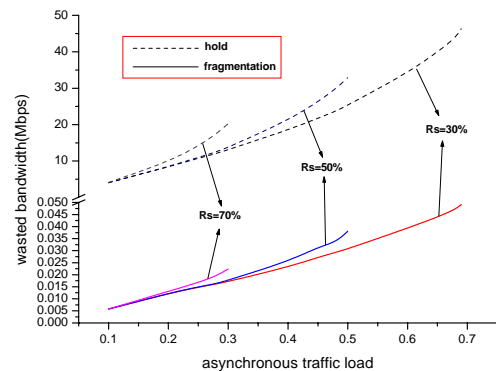


그림 15. 비실시간 트래픽부하에 따른 낭비된 대역폭의 비교

시간이 급격히 차이가 남을 알 수 있다.

<그림 15>는 프레임 분할 방식과 전송보류 방식에 대하여 슈퍼프레임 내의 동기 구간 비율을 30%, 50%, 70%로 각각 변화시킬 때, 비실시간 트래픽의

부하에 따른 낭비된 대역폭을 비교한 것이다. 프레임 분할 방식과 전송보류 방식 모두 비실시간 트래픽의 부하가 클수록 낭비된 대역폭은 증가한다. 하지만, 동기 구간의 비율이 30%이고 비실시간 트래픽의 부하가 0.65일 때, 전송보류 방식의 낭비된 대역폭은 약 40Mbps인 반면에 제안된 프레임 분할 방식의 낭비된 대역폭은 약 0.043Mbps로써, 프레임 분할 방식의 낭비된 대역폭이 아주 작음을 알 수 있다. 이것은 제안된 프레임 분할 방식이 전송보류 방식에 비하여 효율적으로 대역폭을 사용하는 것을 보여준다.

4.2 동기 구간에서의 중지프레임 전송 허용 방안의 성능 분석

본 논문에서는 동기식 이더넷의 흐름제어 방식인 비동기 구간에서만 중지프레임의 전송이 허용되는 기존 방식과, 동기 구간에서 제어프레임 슬롯을 지정하여 중지 정보를 전송하는 제안된 방식과의 성능을 비교하였다. 본 모의실험의 환경모델은 <그림 16>과 같으며, 사용한 파라미터는 <표 2>와 같다. 제안된 방식의 제어프레임 슬롯은 <그림 12>와 같이 두 개를 사용하였다. 단, 스위치와 단말간의 전파지연시간은 무시하였다.

<그림 17>은 <그림 11>과 같이, 중지프레임을 전송하는 동기식 이더넷 스위치의 다운링크 1에서의 동기 구간 비율(R_s)을 30%, 50%, 70%로 각각

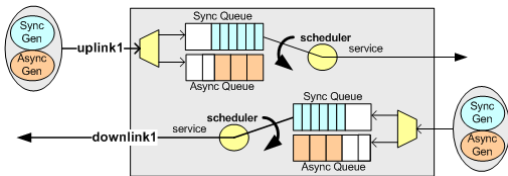


그림 16. 흐름제어 모의실험의 환경모델

표 2. 흐름제어 모의실험의 파라미터

| 파라미터 | 값 |
|---------------------------|---------------|
| 대역폭 | 1 Gbps |
| 수퍼프레임 길이 | 125 μ sec |
| 최대 프레임 길이 | 1518 바이트 |
| 최소 프레임 길이 | 64 바이트 |
| 평균 프레임 길이 | 1250 바이트 |
| 다운링크 1의 동기 구간 비율(R_s) | 30, 50, 70(%) |
| 업링크 1의 비실시간 트래픽 수신 버퍼 | 5 |
| 중지 타이머 기간 | 50 μ sec |
| 동기구간의 제어프레임 슬롯의 개수 | 2개 |

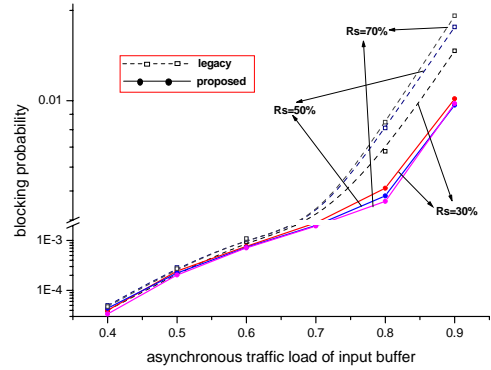


그림 17. 다운링크 1의 동기 구간 비율(R_s)에 따른 비실시간 트래픽의 손실률

변화시킬 때, 업링크 1의 입력 버퍼에서 수신되는 비실시간 트래픽 부하에 따른 비실시간 트래픽의 손실률을 비교한 것이다.

모의실험 결과, 다운링크 1의 동기 구간 비율이 70%이고, 업링크 1의 입력 버퍼에서 수신되는 비실시간 트래픽의 부하가 0.8일 경우, 기존 흐름제어 방식의 비실시간 트래픽의 손실률은 0.84%이고 제안된 흐름제어 방식의 비실시간 트래픽의 손실률은 0.46%로써, 프레임의 손실률이 개선됨을 확인하였다.

본 논문에서 제안하는 흐름제어 방식은 기존의 흐름제어 방식보다 비실시간 트래픽의 손실률이 감소하는 것을 알 수 있다. 또한, 동기 구간 내에 제어프레임 슬롯이 많을수록, 비실시간 트래픽의 손실률은 더욱 감소할 것이다.

V. 결론

본 논문에서는 슬롯 기반의 동기식 이더넷 시스템에서의 두 가지 문제점에 대한 해결 방안을 제안하였으며, 이에 대한 성능 분석을 통하여 제안 방식이 우수함을 확인하였다. 동기식 이더넷 시스템은 실시간 및 비실시간 트래픽을 모두 지원하기 위하여, 동기 구간과 비동기 구간으로 구성된 수퍼프레임이라 하는 일정 주기의 사이클 기반으로 동작한다. 하지만, 이미 송신이 진행 중인 비실시간 프레임 때문에 수퍼프레임의 시작 시점이 지연됨으로써 실시간 트래픽의 전송이 지연되는 문제점 있다. 이를 해결하기 위하여, 기존 방식의 지터 성능을 분석한 후, 전송해야 할 비실시간 프레임이 다음 수퍼프레임의 시작 시점을 초과한다면 다음 수퍼프레임의 시작 시점을 참조하여 비실시간 프레임을 여러 사이클에 분할하여 전송하는 방법인 링크 계층에서의

프레임 분할 방식을 제안하였고, 또 다른 지터 억제 방식인 전송보류 방식과의 프레임 처리시간 및 대역폭 활용도를 모의실험을 통하여 비교하였다. 제안된 프레임 분할 방식은 실시간 트래픽에 대하여 지터가 전혀 없는 엄격한 전송을 보장하였으며, 전송을 포기하는 전송보류 방식보다 프레임 처리 시간이 적으며 대역폭 활용도가 큰 것을 확인하였다.

뿐만 아니라, 실시간 트래픽의 전송 구간인 동기 구간에서도 신속하게 처리되어야 할 흐름제어용 프레임의 전송을 허용함으로써 비실시간 트래픽의 손실률을 낮추는 흐름제어 방식을 제안하였으며, 모의실험을 통하여 비실시간 트래픽의 손실률 성능을 분석하였다. 모의실험 결과, 제안된 흐름제어 방식은 비실시간 트래픽의 손실률을 감소시키는 효율적인 흐름제어 방식임을 확인하였다.

본 논문에서 제안된 방법인 링크 계층에서의 프레임 분할 방법은 무선 LAN용 MAC인 IEEE 802.11에서도 이미 활용되는 기술이며, 중지프레임의 신속한 전송 방법도 용이하게 구현할 수 있는 장점이 있다. 더불어, 제안된 방식들은 앞으로 표준화가 활발히 진행될 레지던셜 이더넷 표준화에 활용 가능하며 관련된 제품의 개발에도 활용될 수 있을 것이다.

참 고 문 헌

[1] K.K.Ramakrishnan and H.Yang, The Ethernet capture effect: analysis and solution, In *Proc. 19th Conf. on Local Computer Networks*, pp. 228-240, Minneapolis, MN USA, Oct., 1994.

[2] Carrier Sense Multiple Access with Collision Detection (CSMA/CD), IEEE Std 802.3-1990 Edition (ISO/DIS8802-3), York,

[3] Elias Keshishoglou, Nathan Yeakel, Alexei Beliaev, hri Arora, "Synchronous Ethernet Specification Draft v0.39", Nov., 2003.

[4] F. A. Tobagi, Carrier sense multiple access with message based priority functions. *IEEE Trans. on Commun.*, pp.185-200, Jan., 1982.

[5] IEEE 802.9: Integrated Services (IS) LAN Interface at the Medium Access Control (MAC) and Physical (PHY), Dec., 1996.

[6] Henry Juszkwicz et al., "Media-accelerated Global Information Carrier Engineering Specification Revision 3.0c", May, 2003.

[7] T.Chiueh, "Rether A Software-Only Real-time Ethernet for PLC network", *Proc. of the Embedded Systems Workshop*, March, 1999.

[8] F. Carreiro et al., "Virtual Token-Passing Ethernet-VTPE," *5th International Conf. on Fieldbus Systems and their Applications*, July, 2003.

[9] S.Varadarajan and T. Chiueh, "EtheReal; A host-transparent real-time Fast Ethernet Switch," *Proc. ICNP*, Oct., 1998.

[10] P. Pedreiras et al., "The FTT Ethernet protocol: Merging flexibility, timeliness, and efficiency," *Proc. of 14th Euromicro Conference on Real-Time Systems*, Viena, June, 2001.

[11] SIMULA, available at <http://www.volny.cz/petr-novak/cim>.

[12] A. S. Krishnakumar, "EQUB-Ethernet Quality of Service Using Black Bursts", Bell Laboratories, Lucent Technologies.

[13] IEEE 802.3 Residential Ethernet Study Group, http://www.ieee802.org/3/re_study/.

[14] Feuser, O, Wenzel, A, "On the effects of the IEEE 802.3x flow control in full-duplex Ethernet LANs", *Local Computer Networks*, pp. 160-161, Oct., 1999.

[15] Jing-Fei Ren, Landry, R, "Flow control and congestion avoidance in switched Ethernet LANs", *IEEE International Conference Volume 1* pp. 508-512, June, 1997.

[16] 윤중호, "최신이더넷", (주)교학사 pp.451-462, 2002.

[17] 윤중호, "기가비트 이더넷", 한국통신학회 고속통신망(HSN) 워크샵 제8권, 1998.

최 희 경 (Hee-kyoung Choi)

준회원



2004년 2월 한국항공대학교 정보통신공학과 졸업(공학사)
 2004년 3월~현재 한국항공대학교 정보통신공학과 석사과정
 <관심분야> 유무선 데이터 통신, 광통신 공학

윤 증 호 (Chong-ho Yoon)

중신회원



1977년 3월~1984년 2월 한양대학교 전자공학과 졸업(공학사)

1984년 3월~1986년 2월 한국과학기술원 전기 및 전자공학과 졸업 (공학석사)

1986년 3월~1990년 8월 한국과학기술원 전기 및 전자공학과

졸업 (공학박사)

1991년 8월~현재 한국한공대학교 항공전자정보통신공학부 교수

<관심분야> MAC 기술 및 BcN 전달망 기술

조 재 헌 (Jae-hun Cho)

정회원



2000년 2월 경희대학교 전자공학과 졸업(공학사)

2002년 2월 경희대학교 전자공학과 졸업(공학석사)

2002년~현재 삼성전자 통신연구소 방송통신기술랩 선임연구원

<관심분야> 레지던셜 인터넷, BcN 기술 분야