이더넷 전송장치에 있어서 최대 전송속도에서의 비동기로 인한 패킷손실 개선

Reducing the Packet Loss Due to Asynchronization At the Maximum Link Speed Between Ethernet Transmission Systems

안 정 균*, 김 성 수, 권 용 식, 엄 종 훈
(Jeong-Gyun Ahn, Sung-Su Kim, Yong-Sik Kwon and Jong-Hun Eum)

Abstract : 본 논문은 이더넷이 비동기식으로 전송됨으로 인해, 동일한 전송속도를 가진 장비라 할지라도 링크가 제공하는 명목상의 최대속도로 전송될 경우, 상호 접속한 장비간의 전송클럭 차이로 인해 프레임의 손실이 발생한다. 본 논문에서는 PHY에서 복원된 수신 클럭과 송신 클럭의 차이를 비교하고 동시에 프레임버퍼에 쌓인 큐를 참조하여 프레임 손실이 방생할 수 있는 임계치를 넘어설 경우, 전송프레임의 프리엠블 길이를 조정함으로써 이더넷 장비에서 전송클럭의 차이로 인한 프레임손실을 줄일 수 있음을 확인 하였다.

Keywords: Ethernet, asynchronization, packet loss, preamble

I. 서론

90년대 말 초고속 인터넷의 급속한 보급과 더불어 전송속도와 가격경쟁력 측면에서 우수한 이더넷이 거의 모든 초고속 인터넷 단말을 점유하였으며, 스위치와 같은 가입자 전송장치들 또한 이더넷 전송장치가 대부분을 차지하고 있는 실정이다.

이러한 현상은 그림 1에서 보듯이 전송방식별 초고속 인터넷가입자의 추이에서도 뚜렷이 보여주고 있다.[1]

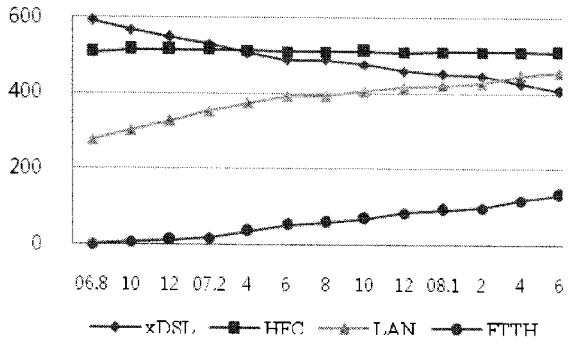

그림 1. 초고속인터넷 전송방식별 가입자 현황

특히, 전통적인 통신사를 중심으로 보급된 XDSL 계열의 가입자 전송장비가 큰 폭으로 감소하는 대신, 이더넷 랜 방식과 E-PON 방식의 FTTH가 같은 기간에 뚜렷하게 증가하는 모습을 볼 수 있다. 이는 국내의 주거 형태가 상당부분 아파트로 형성되어 이더넷 랜방식으로 서비스를 제공하기에 적합한 구조이기 때문이며, 또한 ADSL과 같은 전송성능이 낮은 전송방식이 FTTH로 전환하고 있는 모습을 보여주고 있다.

* 안정균(Corresponding Author)
논문접수 : 2008.7.29., 채택확정 : 2008. 8. 1.
김성수, 권용식, 엄종훈: KT 미래기술연구소 연구전문그룹
(jkahn@kt.com, loach@kt.com, yongsik@kt.com, jheom@kt.com)

더불어 높아진 전송대역폭을 기반으로 대용량의 IPTV 서비스를 비롯하여 VoIP와 같은 높은 수준의 QoS가 요구되는 서비스가 빠르게 확산되고 있다. 그러나 Best-effort 방식을 기반으로 전송하는 이더넷은 높은 수준의 서비스품질을 요구하는 양방향 멀티미디어 서비스의 전송품질을 유지하기가 매우 어려운 것이 사실이다.

이러한 전송품질을 향상하기 위해 혼잡상황에서도 중요한 프레임의 손실을 막기 위해 802.1Q와 같은 우선순위에 따른 스케쥴링 기술이 일반적으로 브릿지에서는 적용되고 있다.[2] 또한, IEEE802.1 AVB 태스크 그룹에서는 L2계층에서 스트림을 예약할 수 있는 기술을 논의하고 있다.[3]

그러나 이더넷 스위치의 경우, 혼잡상황이 발생하지 않는 최대전송속도에서도 이더넷시스템 간의 비동기로 인해 프레임손실이 발생할 수 있으나 대부분의 시스템에서는 혼잡상황과 같이 다루어 프레임 손실을 허용하고 있다. 이러한 비동기로 인한 문제를 근원적으로 해결하기 위해 IEEE 1588과 IEEE 802.1AS 표준 태스크그룹에서는 정밀한 시간동기를 위한 표준화가 이루어지고 있다.[4,5]

본 논문에서는 링크가 허용하는 최대 속도로 전송되고 있는 이더넷 스위치에서, 특정포트로 인입된 프레임이 다른 포트로 전송될 때, 수신클럭의 속도가 송신클럭의 속도 보다 높을 경우 프레임 손실이 발생하게 된다. 이러한 비동기로 인한 프레임 손실이 발생하는 현상과 비동기 상황에서 프레임 손실을 줄일 수 있는 구조를 설계하고 시험을 통해 검증하였다.

II. 본론

1. 비동기에 의한 이더넷 전송

이더넷 장비는 동일한 전송속도를 가진 장비라 할지라도 상호 접속한 장비간에 각 장비가 보유한 클럭이 상호 동기되어 있지 않기 때문에 전송클럭(Tx_Clk)간의 차로 인해 다른 전송시스템에 비해 전송클럭이 낮은 시스템에서는 최고속도로 전송시 필연적인 프레임의 손실이 발생한다.

IEEE802.3 의 물리계층 표준에 의하면 제공되는 송신클럭

은 100Mbps 의 경우 속도의 25%에 해당하는 25MHz ± 100ppm 을 정의하고 있으며, 1Gbps 는 125MHz ± 100ppm 을 정의하고 있다. [6] 100Mbps 패스트이더넷은 그림. 2 에서 보는 바와 같이 물리계층(PHY)에서 수신된 프레임의 프리엠블로부터 수신클럭(Rx Clock)과 바이트(Byte) 동기를 복원한다. 이 복원된 수신 클럭은 장치 자신의 클럭이 아니라 상대측 장비의 클럭이며 장치가 프레임을 전송시에 사용하는 송신 클럭은 장치의 PLL 로부터 직접 받은 송신클럭(Reference Clock, Tx Clock)을 이용한다.

이러한 이유로 인해 100Mbps 나 1Gbps 등의 장치가 이웃하는 장치와 동일한 클럭을 갖지 않을 경우, 동일한 최대 속도로 전송하더라도, 각 시스템 자체가 보유한 클럭의 차이로 인해 전송속도가 낮은 장치에서는 필연적으로 프레임의 손실이 발생한다.

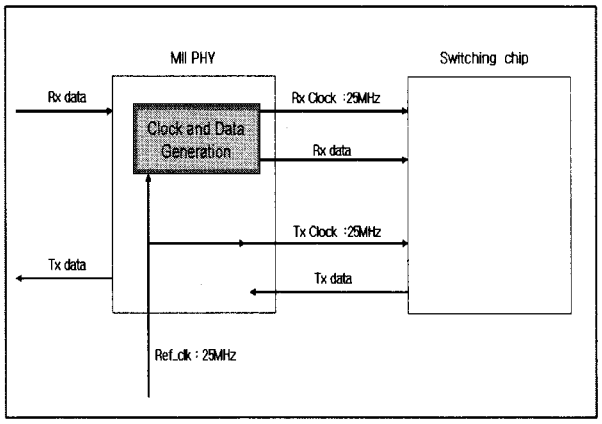

그림 2. 이더넷 장치의 클럭구조

예를 들어서 그림 3 과 같이 차례로 100Mbps 로 전송할 경우, 100Mbps 이더넷 장비에서 사용하는 25MHz ± 100PPM 의 클럭허용 값은 최대 100Mbps ± 10Kbps(0.001%)의 속도차이를 낼 수도 있다.

즉, 두 장비간에 동일한 100Mbps 를 전송한다고 하더라도 최대 20Kbps 만큼의 차이가 발생할 수도 있다. 최대 20Kbps 의 차이는 전송장비의 버퍼에 프레임이 쌓이게 되고, 따라서 일정시간 후 버퍼가 가득 차게 되거나 혹은 큐가 부족하게 됨에 따라 프레임의 손실이 발생한다.

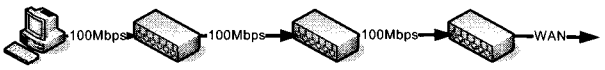

그림 3. 100Mbps 전송네트워크

프레임 손실이 발생하는 구조는 장치의 설계에 따라 다르지만 대부분의 장치에서는 주로 큐의 부족에 의해 프레임 손실이 발생한다. 이런 경우, 송수신 간의 클럭차이로 인해 프레임 손실이 발생하는 시간을 일반화하여 정리하여 보면 다음 식 (1)과 같이 간략하게 표현할 수 있다.

$$T_{\text{frame loss}} = (\#\text{ of Queues}) \times (\text{Frame Size}) / (\Delta\text{ppm} \times 10^{-6} \times B_{max}) \quad (1)$$

$T_{\text{frame loss}}$ 는 수신 장비에서 프레임 손실이 최초로 발생하는 시간(sec)이고, # of Queues 는 장치의 큐메머리에 할당 가능한 큐의 개수, Δppm 은 장비간의 100 만개당 장치간의 클럭차이, 그리고 B_{max} 는 장치의 최대 전송속도(bps)를 나타낸다.

따라서 지속적으로 최고의 전송속도로 일정시간을 전송하게 되면 위의 식에서 보듯이 장치가 보유한 클럭의 차이로 인해 $T_{\text{frame loss}}$ 의 시간이 경과되면 프레임 손실이 일어나고, 이것은 비동기식으로 전송하는 이더넷장치의 특성이기도 하다. 베스트에포트(Best Effort)에 기반한 이더넷 전송장치는 기본적으로 혼잡에 의한 프레임의 손실과 비동기로 인한 프레임의 손실을 전제로 하지만, 프레임의 손실은 프레임의 재전송 시도를 하게 되고 따라서 네트워크의 혼잡이 가중되게 된다. 이러한 프레임 손실은 고품질의 실시간 방송이나 실시간 통화 등과 같은 서비스의 품질을 저하시키는 주요한 원인이 되고 네트워크의 혼잡을 가중시키기 때문에 프레임의 손실은 가능하면 최대한 방지할 수 있어야 한다.

2. 프레임 손실을 방지하기 위한 구조설계

비동기로 인한 프레임손실을 방지하기 위해서는 송수신 클럭의 차이로 인한 데이터 전송속도를 조정할 수 있는 구조를 전송장비가 가져야 하며 본 논문에서 제안하는 전송속도를 조절하기 위한 구조를 간략하게 표현하면 다음과 같다.

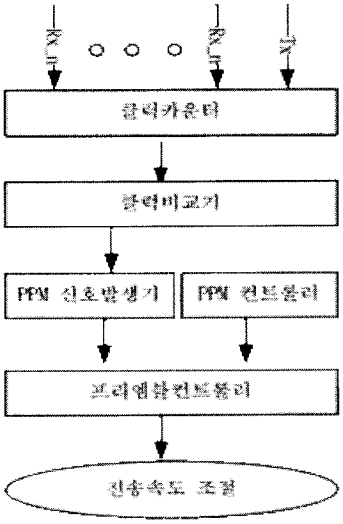

그림 4. 전송속도조절 구조

위 그림의 전송속도조절을 위한 구조를 간략하게 설명하면, 다른 이더넷 시스템으로 받은 프레임의 프리앰블로부터 복원한 수신클럭(Rx_Clk)의 수를 자체클럭(Ref_Clk)을 기준으로 일정 시간 동안 측정하는 클럭카운터(Tx or Rx_Clk Counter), 전송포트로 인입되는 수신포트별 Stream 대역폭 점유율을 측정하는 플로우 웨이팅기(Flow weighting), 단위 시간당 전송포트(Tx 포트)로 들어오는 수신포트의 클럭을 대역폭 점유율로 환산하고 전송클럭과 차이(ppm, part per million)를 비교하는 클럭비교기, 클럭비교기의 PPM 결과를 바탕으로 PPM 조정을 위해 신호를 발생하는 PPM 신호발생기, 스케쥴러의 큐에 쌓여있는 프레임의 량을 측정하여 전송속도를 높일 것인지 아닌지를 판단하는 PPM 컨트롤러, 그리고 PPM 신호발생기와 PPM 컨트롤러로부터 받은 결과를 바탕으로

전송속도의 조절을 최종 판단하는 프리엠블컨트롤러(Preamble Controller) 등으로 구성된다.

이렇게 설계된 전송속도조절 장치의 동작은 먼저, 복원한 수신클럭(Rx_Clk)과 자체 전송클럭(Ref_Clk or Tx_Clk)을 일정시간 동안 측정하여 클럭비교기에서 두 클럭간의 차이를 ppm(part per million) 단위로 계산하여 전송속도를 높여야 할지를 1 차적으로 판단한다.

다음으로 무조건적인 전송속도 조정을 방지하기 위해 전송을 위해 대기중인 큐(Queue)에 충분한 공간이 있는지를 판단하기 위하여 큐에 일정한 상한치 (Up threshold) 이상으로 프레임이 쌓이면 ppm 컨트롤러가 전송속도를 높이기 위한 신호(PPM Enable)를 보내고, 또 일정 하한치(Down Threshold) 이하로 전송대기 프레임이 줄어들면 정상적인 속도로 복귀하기 위한 신호(PPM Disable)를 출력하도록 한다.

클럭비교기에서 전송클럭이 수신클럭 보다 낮아서 프레임 손실이 예상되어 PPM 신호발생기로부터 전송속도를 높이는 신호를 받고, 큐에서도 프레임이 충분히 쌓여 프레임손실이 일어날 우려가 발생하는 상한치(Up threshold)를 넘어서는 조건이 동시적으로 형성되면 프리엠블 컨트롤러(Preamble controller)는 즉각적으로 프리엠블 7 바이트 중에서 1 바이트를 삭제하여 전송속도를 높여 큐에 쌓인 프레임을 줄여서 프레임 손실을 방지한다. 전송속도를 높여 큐에 누적된 프레임 수가 하한치(Down threshold) 이하로 내려가면 PPM 컨트롤러는 프리엠블컨트롤러의 동작을 정지시켜 본래의 전송속도를 유지한다.

이러한 절차를 반복적으로 수행함으로써 전송속도를 1518byte 의 경우 0.06%에서 64byte 에서는 최대 1.2% 정도 순간적으로 전송속도를 높임으로써 장비의 최대속도로 전송시에도 비동기로 인한 프레임의 손실을 방지 할 수 있다.

전송속도조절 장치의 세부 모듈의 기능을 설명하면 다음과 같다.

가. 클럭비교기(Clock Comparator)

수신속도와 전송속도를 비교하기 위해서는 먼저 수신클럭과 송신클럭을 비교할 수 있어야 한다.

그림 5 는 MII 또는 GMII 물리계층(PHY) 인터페이스를 통해 전송포트(Tx 포트)로 들어오는 프레임의 클럭들을 분석 비교하는 클럭비교기의 구조이다. 다른 이더넷 시스템으로부터 프레임을 받은 수신포트(RX 포트)에서는 10101010 으로 계속되는 7 바이트 프리엠블(preamble)의 비트열에서 수신동기클럭(Rx_Clk)를 추출하여 수신데이타(Rx_data)를 복원한다. 이와 같이 각각의 수신포트에서 추출한 수신동기클럭(Rx_Clk)을 각각 백만개 단위로 수신클럭카운터(Rx_Clk Counter_n)가 측정하여 클럭비교기로 보낸다.

클럭비교기는 그림 5 에서 보는 바와 같이 다수조의 수신클럭카운터로부터 수신포트에서 추출한 수신동기클럭을 단위시간당 백만개단위(ppm)로 받아들이고, 또 포트별 플로우 웨이팅기(Flow Weighting)로부터 전송포트(Tx 포트)의 대역폭을 점유하는 포트별 스트림 점유율과 스트림의 총점유대역(Total Bandwidth, B_{TB})을 전송받는다.

클럭비교기는 다수의 포트로부터 수신되는 데이터가 최대 전송속도보다 많은 데이터가 인입되고 있는 혼잡상황인지, 혹은, 최대전송속도로 전송되고 있는지를 판단하여야 한다. 클럭비교기는 플로우 웨이팅기로부터 받은 총점유대역(B_{TB})이 전송포트의 명목상의 최대대역폭(B_{max})과 비교하여 표준에서 정의된 클럭의 최대 허용치인 ± 100PPM 을 초과할 경우, 즉 표준허용대역폭(B_{std_max}= B_{max} x $(1 + 2x10^{-4})$)을 초과할 경우는 인입되는 프레임들이 혼잡상황이라는 것을 의미하며,

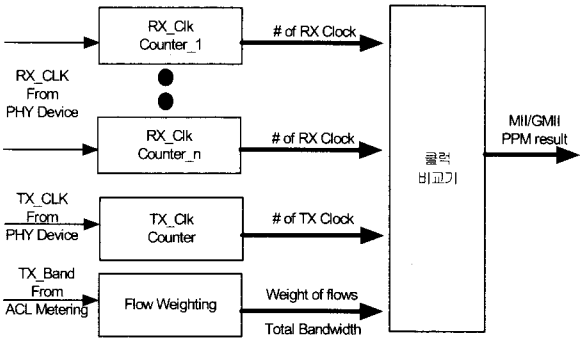

그림 5. MII/GMII 의 송수신 클럭비교기

PPM 결과값으로 '0'을 출력하여 혼잡에 의한 프레임 손실을 전송속도를 높여 해결하고자 하는 것을 방지한다.

그러나 총점유대역(B_{TB})이 B_{max} 와 B_{std_max} 사이인 경우에는 다음과 같이 클럭의 차이(Δppm)를 계산한다.

$$\Delta ppm = Weighted\,Rx_Clk \;-\; Tx_Clk \qquad (2)$$

여기서,

$$Weighted\,Rx_{Clk} \qquad \sum_{i=1}^{n}(WB_i \;\mathbf{x}\; Rx_Clk_i \qquad (3)$$

이다.

여기서 Δppm 은 가중치를 곱한 수신클럭(*Weighted Rx_Clk*)과 전송클럭(*Tx_Clk*)과의 차이를 백만개 단위로 표현한 값이고, *Tx_Clk* 은 백만개 단위의 전송클럭이다. i 는 전송포트로 프레임을 보내는 수신포트 번호이고, n 은 수신포트의 총수이다. WB_i는 i 번째 포트의 스트림이 총점유대역폭을 점유하는 점유율 가중치이고, *Rx_Clk_i* 는 i 번째 포트의 수신클럭이다. 따라서 *Weighted Rx_Clk* 은 전송포트를 점유한 포트별 점유율의 정도를 반영하여 수신클럭에 미치는 영향을 표현하였다.

특히, 1 개의 수신포트로 들어와서 1 개의 송신포트로 전송될 경우, 즉 한 개의 수신포트로부터 인입된 모든 스트림이 송신포트 전체의 대역폭을 점유할 경우 위의 식(2)는 다음과 같이 간략하게 표현된다.

$$\Delta ppm \;=\; Rx_Clk - Tx_Clk \qquad (4)$$

식(2)와 식(3)에서 Δppm 의 값이 양수가 되면 수신클럭(*Rx_Clk*)이 더 빠르다는 것을 의미한다.

클럭 비교기에서 *Rx_Clk* 과 *Tx_Clk* 의 클럭속도를 비교한 다음, PPM 결과를 그림 6 의 PPM 신호발생기로 결과 값을 전송한다.

나. PPM 신호발생기

그림 6 에서 보는 바와 같이 PPM 신호발생기는 PPM 결과가 양수이면(PPM Result > 0), PPM 신호를 '1'로 펄스를 발생하여 프리엠블 컨트롤러로 보낸다.

클럭비교기로부터의 PPM 결과가 양수일 경우, 송수신 클럭의 차이에 의해 프레임의 손실이 발생할 우려가 있는 조건이 형성된다. 따라서 PPM 신호발생기는 PPM 결과값을 ± 100ppm 을 적절한 단계로 세분화 하여 펄스주기를 달리한다. 본 연구에서는 16 단계로 세분화 하여 펄스를 생성하도록 하였다. 즉, 송수신간의 클럭차이가 큰 경우 잦은 펄스를 발생하도록하고 차이가 적은 경우 낮은 주기로 펄스를 생성하도록 설계하였다.

다. PPM 컨트롤러

송수신 클럭의 차이가 있더라도 송신포트에서 충분히 프레임을 전송할 수 있는 경우, 즉 최대대역폭(B_{max})에 다다르지 않았을 경우에는 전송속도를 높일 필요가 없다. 따라서 전송포트에 큐가 쌓인 상태를 확인할 필요가 있다.

전송을 위해 대기중인 큐(Queue)에 충분한 공간이 있는지를 판단하기 위하여 큐가 일정한 상위임계치(Up threshold) 이상으로 프레임이 쌓이면 PPM 컨트롤러에서 전송속도를 높이도록 명령(PPM Enable=1)을 내리고, 큐에 쌓인 프레임이 줄어들어 하위임계치(Down threshold)이하로 내려오면 PPM 컨트롤러는 전송속도를 복원하는 명령(PPM Enable = 0)를 내린다.

라. 프리엠블 컨트롤러

프리엠블컨트롤러는 다음과 같은 조건이 만족되어야 전송속도를 높이는 직접적인 동작을 취한다. 첫째, 클럭비교기에서 전송포트로 인입되는 플로의 총점유대역(Total Bandwidth, B_{TB})이 B_{max} 와 B_{std_max} 사이의 대역폭이고, 둘째, 클럭의 차이 Δppm 이 양수로 PPM 펄스 발생기로부터 '1'의 펄스를 받은 경우, 그리고 셋째, 큐에 쌓여있는 프레임의 량이 상위임계치(Up Threshold) 이상으로 PPM 컨트롤러로부터 'enable=1'의 신호를 동시에 만족하여야 한다.

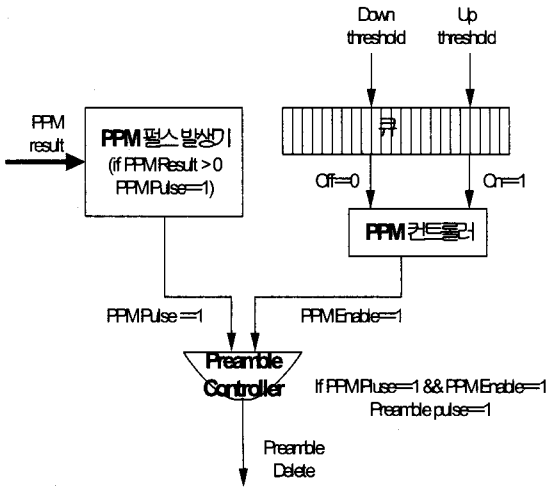

그림 6. 전송속도 조절 구조

전송속도를 높이기 위한 세 가지의 동작조건이 만족되면, 프리엠블 컨트롤러는 프레임에 대해 1Byte 씩 프리엠블을 삭제하는 명령(Preamble Pulse =1)을 내려 프리엠블을 삭제한다. 프리엠블은 7byte 의 '10101010'계속되는 비트열로 수신클럭의 동기와 바이트 동기를 제공한다. IEEE802.3 표준에서는 7Byte 를 정의하고 있으나 실제 시스템에서는 프리엠블의 길이가 짧아도 동기클럭을 추출할 수 있다. 프리엠블 1 바이트를 삭제함으로써 IFG(Inter Frame Gap)의 변경이나 동기클럭의 복원에 지장 주지 않으면서 이더넷프레임의 총 길이를 1byte 줄임으로써 전송속도를 높일 수 있다.

전송속도를 0.06%에서 최대 1.2% 정도 전송속도를 순간적으로 높여 프레임 손실을 방지하고, 큐에 쌓인 프레임이 하한치(Down threshold) 이하로 내려가면 PPM 컨트롤러는 프리엠블컨트롤러의 동작을 정지시켜 본래의 전송속도를 유지한다.

이러한 전송속도 조절 구조의 설계로 혼잡이 발생하지 않은 상황에서 장비의 최대속도로 전송시, 장비간의 클럭속도 차이에 의해 발생하는 프레임의 손실을 방지할 수 있다.

3. 시험구성 및 결과

비동기로 인한 이더넷시스템에서의 프레임손실을 줄이기 위해 설계된 전송속도조절장치의 동작을 확인하기 위해 단순하게 전송클럭의 차이가 있는 이더넷 스위치를 아래 그림 7 과 같이 구성하였다. 네트워크 분석기에서 64byte, 128byte, 256byte, 512byte, 1024byte, 1518byte 의 프레임을 10 분간 지속적으로 100Mbps 로 이더넷스위치로 전송하고 다시 스위치에서 다른 포트를 통해 네트워크 분석기로 전송하였다.

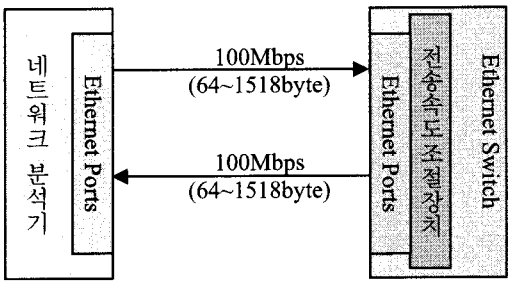

그림 7. 전송속도 조절 시험구성도

클럭비교기에 의해 분석된 스위치에서 복원된 네트워크 분석기의 전송속도와 스위치 자체의 전송속도 간에는 11.76ppm 의 클럭차이가 있었다. 이를 속도로 환산할 경우 1176bps 정도 네트워크 분석기가 스위치에 비해 전송속도가 빠르다는 것을 알 수 있다.

시험에서 사용한 스위치의 전송포트당 동일순위의 큐 갯수가 40 개이고, 프레임메머리는 1.5Mbit 이지만 192kbyte 단위의 셀로 관리되어 1000 개의 프리셀에 프레임이 버퍼링 되었다가 전송된다. 따라서 전송속도 차이에 의해 저장되는 프레임이 프레임메리보다 할당된 큐가 부족하여 손실이 발생하기 시작한다. 즉, 40 개의 특정 우선순위의 큐가 소진된 이후부터 프레임 손실이 발생한다.

프레임이 최대속도로 전송된 후 최초 프레임손실이 발생

하기 시작하는 시간은 식(1)에 따라 계산되며, 프레임의 길이가 길어짐에 따라 최초 손실발생 시간이 늘어나고 프레임 손실은 줄어들고 있지만, 시간이 길어지면 반드시 프레임손실이 발생한다는 것을 표 1.로부터 예측할 수 있다. 식(1)에 의하면 64Byte 의 가장 짧은 프레임이 전송될 경우 약 23 초 후부터 비동기로 인한 클럭차이로 프레임 손실이 발생하기 시작하고, 가장 긴 1518Byte 를 전송할 경우에는 419 초 이후에나 최초의 프레임손실이 예상된다. 그러나 1518Byte 에서는 10 분간의 전송시험결과가 예측치와 다소간의 차이가 있어서 프레임 발생이 되지 않았다. 그러나 시간을 연장하여 시험하였을 때, 프레임 손실이 발생하는 것을 확인하였다.

표 1. 프레임 손실예측 및 결과값

frame size (Bytes)	예측치	Drop Frame	최초손실 발생시간 (sec)
64	1,010	768	23
128	556	446	40
256	280	264	75
512	126	124	145
1,024	44	65	284
1,518	17	0	419

예측치와 실제 프레임손실간의 차이는 시험에 사용된 스위치의 특정구조로 인한 측면이 있지만 전체적인 경향에서 어느 정도 예측되고 있음을 볼 수 있다.

그림 8 에서 볼 수 있듯이 10 분간 최대속도로 송수신할 때, 송수신 장비간의 비동기로 인해 발생하던 프레임손실이 전송속도조절 기능을 활성화 하였을 경우에는 프레임 손실이 전혀 발생하지 않았다.

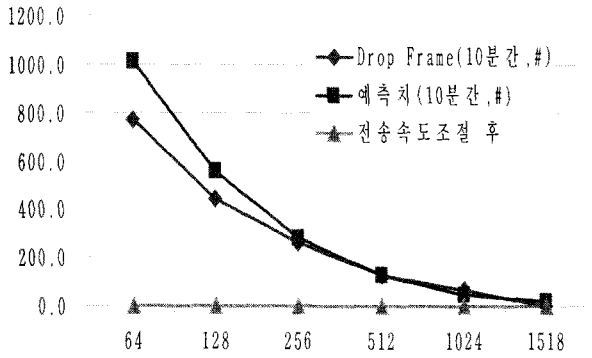

그림 8. 전송속도 조절결과

이는 순간적으로 프리엠블 컨트롤러로 프리엠블의 길이를 조절하여 전송속도를 높임으로써 송수신 장비간의 클럭차이에 의한 전송속도의 차이를 없앰으로써 프레임손실 없이 전송하였다는 것이다.

III. 결론

본 논문에서는 링크가 허용하는 최대 속도로 전송되는 이더넷 스위치에서, 특정포트로 인입된 프레임이 다른 포트로 전송될 때, 수신클럭의 속도가 송신클럭의 속도 보다 높을 경우 프레임 손실이 발생하게 된다. 이러한 비동기로 인한 프레임 손실의 발생을 예측하고, 이를 줄이기 위해 전송속도조절 구조를 설계하였다.

전송속도조절을 위한 구조는 클럭의 속도를 측정하는 클럭카운터, 송수신 클럭간의 차이를 계산하는 클럭비교기, 수신속도가 송신속도에 비해 빠를 때 펄스를 발생시키는 ppm 신호발생기, 큐의 상태를 확인하고 임계치 이상의 프레임이 쌓일 경우 프레임손실이 발생할 수 있음을 알리는 ppm 컨트롤러, 그리고 프레임손실을 줄이기 위해 프리엠블의 길이를 조절하는 프리엠블 컨트롤러 모듈 등으로 구성하였다. 전송속도의 조절은 클럭의 차이를 계산하고, 큐의 상태를 확인하여 프레임의 손실이 예측될 경우, 프리엠블의 길이를 조정하여 전송속도를 조절하였다.

그리고 송수신간의 클럭차이가 있는 장치에 링크가 허용하는 최대 대역폭으로 프레임을 전송하였을 때 프레임의 손실을 관찰하고, 전송속도조절 장치를 동작시켜 프리엠블의 길이를 순간적으로 조절함으로써 비동기로 인한 프레임 손실이 발생하지 않음을 확인 하였다.

참고문헌

[1] 방송통신위원회, http://www.mic.go.kr home page

[2] IEEE Std 802.1Q-2005 Standard for Local and metropolitan area network-Virtual Bridged Local Area Networks, May, 2006

[3] IEEE P802.1Qat/D1.1 Draft standard for Local Area Networks-Amendment9: Streaming Reservation Protocol(SRP), IEEE Computer Society, March, 2008

[4] IEEE 1588 Committee, IEEE P1588TM/D1-F Draft Standard for a Precision Clock Synchronization Protocol for Networked Measurement and Control Systems, IEEE WG IM/ST Committee, March 2007

[5] IEEE P802.1AS/D2.0 Draft standard for Local and Metropolitan Area Networks-Timing and Synchronization for Time - Sensitive Applications in Bridged Local Area Networks, IEEE Computer Society, Feb,2008.

[6] IEEE Std 802.3TM-2005 Standard for Local and metropolitan area network-Part3:Carrier sense multiple access with collision detection(CSMA/CD) access method and physical layer specifications, Dec, 2005