

---

# 10Gbps 이더넷 응용을 위한 MAC 코어의 설계 및 검증

손승일\*

Design and Verification of MAC Core for 10Gbps Ethernet Application

Seung-il Sonh\*

---

이 논문은 2006년도 한신대학교 연구비 지원에 의해 수행되었음.

---

## 요 약

최근 대부분의 전송기술(LAN 뿐만 아니라 MAN과 WAN까지)이 이더넷으로 통일되는 경향에 힘입어, 예전에 비하여 이더넷은 대단한 주목을 받게 되었다. 하드웨어 설계를 위해 10Gbps 이더넷 Data Link 계층의 MAC 코어를 C언어를 이용하여 성능평가를 실시하여 내부 FIFO의 크기를 도출하였다. 본 논문에서는 VHDL 언어와 Xilinx ISE 6.2i 툴을 이용하여 상위 계층 인터페이스, 전송엔진, 플로우 컨트롤 블록, 수신엔진, 정합 부계층(Reconciliation Sublayer), 초기설정 블록, 상태전송 블록, XGMII 인터페이스 블록으로 구성되는 10Gbps 이더넷용 MAC(Media Access Control) 코어를 설계하고 Model\_SIM 5.7G 시뮬레이터를 이용하여 검증하였다. 10Gbps 이더넷의 권고안에서는 10Gbps를 지원하기 위해 64비트 데이터 패스를 갖는 MAC 코어는 156.25MHz를 지원해야 하는데, 설계된 MAC 코어는 64비트의 데이터를 처리하고 168.549MHz를 지원하여 최대 10.78Gbps의 데이터 처리를 지원한다. 이는 10Gbps 이상의 고속의 데이터처리가 요구되는 응용분야에 적합하다.

## ABSTRACT

Ethernet has been given a greater attention recently due to tendency of unifying most of transmission technique(not only LAN, but MAN and WAN) to ethernet. Performance evaluation was performed using C language for 10Gbps ethernet Data Link to design the optimum hardware, then internal FIFO size was evaluated. In this paper, MAC core for 10Gbps ethernet which contains high layer interface, transmit engine, flow control block, receive engine, reconciliation sublayer, configuration block, statistics block, and XGMII interface block was designed using VHDL language and Xilinx 6.2i tool and verified using Model\_SIM 5.7G simulator. According to the specification of 10Gbps ethernet, MAC core with 64-bit data path should support 156.25MHz in order to support 10Gbps. The designed MAC core that processes 64-bit data, operates at 168.549MHz and hence supports the maximum 10.78Gbps data processing. The designed MAC core is applicable to an area that needs a high-speed data processing of 10Gbps or more.

## 키워드

10Gbps Ethernet, MAC, XGMII, VHDL

## I. 서론

초기에 공유매체를 기반으로 개발된 이더넷 시스템은 링크들의 대역폭 수요 증대에 부합하기 위하여 100Mbps의 패스트(Fast) 이더넷을 채택하였으나, 매체 공유방식으로 인하여 발생하는 트래픽의 문제를 적절히 해결할 수 없었다. 이후 이더넷 시스템은 노드에 일정한 대역폭을 제공할 수 있는 스위치 개념의 시스템으로 발전하였고, LAN 백본망에는 트래픽의 병목현상을 제거하기 위하여 기가비트 이더넷 스위치 시스템이 개발되어 적용되고 있다. 또한 이더넷 기술은 기가비트 속도로 근거리 통신뿐만 아니라 장거리 통신을 위한 방법으로 발전되고 있으며 현재는 10Gbps 이더넷의 표준화가 완성되어, 이러한 표준안을 채용한 제품군의 출시가 이루어지고 있는 단계에 있다. MAC 인터페이스는 IEEE 802.3ae-2002 Spec에서 권고한 규격[1,2,6]이다.

본 연구에서는 10Gbps 이더넷을 통한 프레임 전송을 위한 물리계층소자와 네트워크계층소자 간의 인터페이스인 MAC 인터페이스 모듈을 설계하는 것이다. 이를 위해 C언어를 이용하여 성능 평가하여, 여러 가지의 PHY 방식을 적용하기 위한 IFG(InterFrame Gap)의 변화와 MAC 프레임 크기의 변화에 따른 MAC 인터페이스의 적용 정도에 맞는 MAC FIFO의 크기[7-8]를 도출하였다. 이러한 기존의 연구[9-11]를 바탕으로 하드웨어 설계언어인 VHDL을 이용하여 회로 설계를 수행하였고, Xilinx ISE 6.2i을 이용하여 합성하였으며, Model\_SIM 5.7G 시뮬레이터를 이용하여 시뮬레이션을 통해 검증을 완료하였다.

## II. 10Gbps 이더넷용 MAC 개요

10Gbps 이더넷은 상업용으로 개발되어 급속하게 확산되고 있다. 특히, 고속의 라우터나 스위치[12]에서 10Gbps 이더넷 인터페이스를 채용하여 설계되고 있다. 10Gbps 이더넷의 기능 및 요구 사항은 다음과 같다. 전이중 통신만을 지원하는 MAC을 가지며, XGMII 인터페이스를 통해 10Gbps 데이터 전송속도를 지원한다. 뿐만 아니라 전송선로가 10Gbps 미만인 SONET과 같은 WAN 전송방식도 지원하므로 MAC에서 전달되는 10Gbps 속도의 데이터와 9.95Gbps급 WAN 전송부와의 속도 조절을 위한 104

비트의 ifsStreshRatio 값이 추가 되었다. 그리고 BER 값이 10-12 이하의 값을 가져야 한다.

현재 Xilinx CS201[13]은 10Gbps 이더넷용 MAC 코어로 출시되었으며, 이 밖에 일부 연구소 및 업체가 MAC 코어를 확보하고 있는 것으로 알려져 있으나 이를 공식적인 논문을 통해 발표한 사례는 거의 없는 실정이다.

그림 1은 10Gbps 이더넷의 아키텍처[14]를 보여주고 있다. 먼저 네트워크 계층에서 TCP/IP 프로토콜을 처리한 패킷은 버스 인터페이스를 통해 데이터 링크 계층에 전달되고 본 논문에서 설계하고자 하는 MAC 계층으로 전달된다. PCS(Physical Coding Sublayer)에서 64b/66b 코딩을 수행한다. 이후 점선으로 표시된 WIS(WAN Interface Sublayer)에서는 LAN PHY가 아닌 WAN PHY와 인터페이스를 지원할 경우에만 선택적으로 추가되는 부계층이다. 이후 물리적 매체의 부착에 관여하는 PMA(Physical Medium Attachment)와 물리적 매체에 따라 종속적인 PMD(Physical Medium Dependent)로 구성된다.

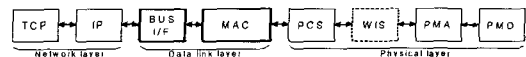


그림 1. 10Gbps 이더넷 아키텍처  
Fig. 1 10Gbps ethernet architecture

상위계층 패킷(LLC 또는 IP/TCP)을 수신측 MAC으로 오류없이 전달하기 위하여, 일반적인 MAC 프레임은 주소 부분이 있는 헤더와 상위계층 패킷을 담은 정보영역 그리고 오류검사를 위한 트레일러 부분으로 구성된다. 각 프레임의 도착을 알리는 7바이트의 프리앰블은 각 프레임이 송신된 속도로 수신될 수 있도록 송신스테이션과 수신스테이션의 클럭을 동기화시킨다. 프리앰블 다음에 오는 1바이트의 프레임 시작 구분자는 수신스테이션에 프레임의 실제 부분이 시작된다는 것을 알린다. 수신스테이션 주소, 송신스테이션 주소와 종류/길이 필드를 합쳐서 일반적으로 이더넷 헤더라고 부른다. 이더넷 헤더는 이더넷이 송신지, 수신지, 크기와 데이터 필드에 있는 상위계층 데이터 패킷의 프로토콜을 아는 데 사용하는 제어 정보를 가지고 있다. 헤더 필드 바로 뒤에 따라오는 데이터 필드는 길이가 46에서 1500바이트까지 담길 수 있다. 프레임의 다른 모든 필드는 길이가 정해져 있다. 프레임 검사열은 이더넷 프레임의 마지막에 있으며 프레임이 전송 중에 에러가 발생하지 않았다는 것을 확인하는데 사용될 체크섬 값을 가지고 있다. 체크섬은 프레임의 다른 모든 필드에

서 나온 비트 값을 계산한 결과이다. 802.1Q의 VLAN[3][5]인 경우, 다음 그림 2와 같이, 4바이트의 Qtag가 추가되어, 최대길이는 1522바이트로 증가되었다. 그림 2는 MAC으로 입력되는 프레임 정보를 보여주고 있다.

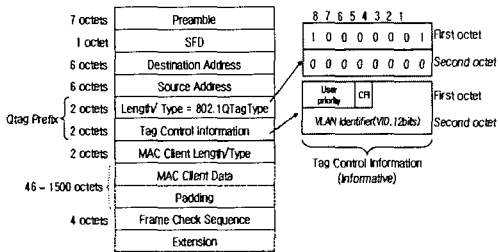


그림 2. MAC 프레임 정보  
Fig. 2 MAC frame information

### III. 하위 계층 인터페이스

#### 3.1 정합부계층

정합 부계층은 MAC과 PCS간의 인터페이스인 XGMII와의 인터페이스를 위해 추가되었다. 10Gbps 이더넷에서는 기존의 정합 부계층에 있던 RX\_DV, RX\_ER, TX\_EN, TX\_ER, COL 등의 신호선들은 모두 RXC(RX Control), TXC(TX Control)로 대체되었다. 또한 충돌감지 기능이 없기 때문에 간단한 신호선으로 MAC과 PCS를 접속할 수 있도록 하였다. 그림 3은 정합 부계층과 XGMII[13][16-18]의 구성을 보여주고 있다.

#### 3.2 XGMII

XGMII는 기존의 1000Mbps용 GMII 기능과 유사한 MAC부분과 물리적 장치(PCS)를 연결하는 인터페이스이다. 기존의 1000Mbps급 GMII에서는 송수신시 8비트 단위로 전달되는데 비해 XGMII에서는 32비트 단위로 송수신 스트림이 전달되는 점이 특이하며, 전이중방식만 지원한다. 또한, 정합 부계층은 링크 폴트 감지기능도 수행하는데, 수신 상태를 감시하여, 고장난 링크에 대하여 해당 단말에게 알린다. 그리고 기본적으로 MAC과 PHY 칩들을 연결시 XGMII의 접속 길이는 7Cm가 최대인데, XAUI를 사용하면 이러한 XGMII의 접속길이를 대략 50cm까지 연장할 수 있다. 그림 3에서 XAUI는 \*로 표시하였는데, 이는 MAC이 PHY 칩들과의 거리가 멀리 떨어져 있을 경우에는 사용되는 계층을 의미한다.

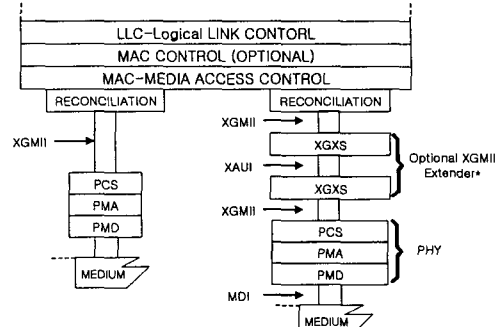


그림 3. 정합 부계층과 XGMII의 구성  
Fig. 3 Reconciliation Sublayer and XGMII

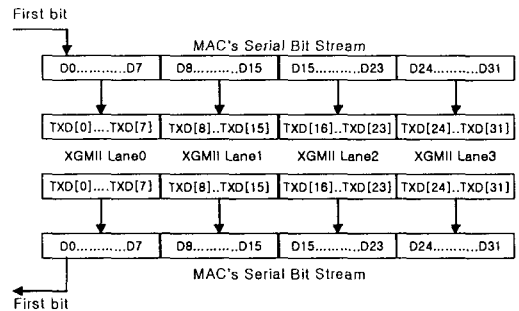


그림 4. 데이터 레인과 MAC 시리얼 비트 패턴과의 관계  
Fig. 4 Relation between data lanes and MAC serial bit pattern

표 1. 컨트롤에 따른 허용된 제어문자 인코딩  
Table 1. Encoding of escape character

TXC	TXD	Description	Data,Request parameter
0	00~FF	일반적인 데이터 전송	
1	07	Idle	
1	9C	Sequence (0번라인에만존재)	Normal inter-frame
1	FB	Start (0번라인에만존재)	
1	FD	Terminate	Data Complete
1	FE	Transmit Error propagation	Normal inter-frame
1	etc.	Reserved	

32개의 데이터라인과 4개의 제어 신호선은 4개의 레인으로 나누어지며, 모두 동일한 TX\_CLK이나 RX\_CLK에 맞추어진다. 이 4개의 레인에 있는 데이터들은 그림 4와 같이 라운드로빈 방식에 의해 각 바이트씩 처리된다. TX\_CLK은 상승 엣지와 하강 엣지 모두에서 데이터가 전달되는 더블 엣지 클럭킹을 사용하므로, 데이터 폭이 32비트로 156.25 MHz의 클럭 속도를 지원해도 10Gbps의 데

이더 전송속도를 지원한다

TXC는 4개의 레인에 대한 송신을 제어하는 신호선으로서 데이터 송신 중이 아닌 제어용 문자를 송신하는 기간에만 활성화된다. 표 1은 XGMII의 송수신용으로 규정된 데이터와 제어문자[13]를 보여주고 있다.

#### IV. MAC 코어의 설계

##### 4.1 고속 데이터 처리용 MAC 모듈의 제안

입력 프레임의 처리에 대해 Cut\_Through 방식을 지원하여 고속 데이터 처리를 지원하였다. 그리고 전송 MAC FIFO에서 64비트의 데이터와 7비트의 컨트롤 데이터를 연속적으로 입력받아 프레임의 시작인지 끝인지를 프레임 체크와 프레임의 길이정보를 카운트하여 프레임의 전송이 종료 되는 시점부터 IFG만큼 감소하면서 Idle 컨트롤 정보를 전송 한다. IFG를 수행한 후 프레임의 ACK 신호가 발생하면서 연속적인 프레임 전송하고 Data Path를 거쳐 DDR (Double Data Rate) 인터페이스를 통하여 데이터를 전송한다. 프레임 전송시 프레임의 상태정보를 상위계층에 보내준다. 그리고 하위 계층의 수신 FIFO에서 DDR 인터페이스를 통하여 데이터와 제어신호를 보내고 받은 데이터의 오류 체크를 한 후 오류가 발생하지 않았으면 데이터 패스를 거쳐 64비트의 데이터와 7비트의 제어신호를 MAC FIFO에 전송하고 에러가 발생 하였으면 해당 프레임을 체크하고 프레임의 최대/최소 길이에 어긋나거나 DA(Destination Address)가 자신의 주소가 아닌 경우나 FCS가 틀린 경우, 프레임을 버린다. 하위 계층에서 받은 클럭을 이용하여 DCM (Digital Clock Management) 모듈에서 내부에서 사용가능한 클럭으로 재생성하여 사용하고 내부 상태와 전송된 데이터의 정보를 상태 저장 벡터에서 상위 계층으로 알려주는 기능을 가지고 있다. 그리고 휴지 상태 컨트롤을 통한 플로우 컨트롤을 지원한다. 고속 데이터 처리를 위해 데이터 처리 지연요소가 가장 큰 데이터 패스에 대해 전송부는 4단 파이프라인, 수신부는 3단 파이프라인 방식을 제안하여 최대 지연요소를 해소 하였다. 그림 5는 고속 데이터 처리를 지원하는 MAC 코어의 블록도를 보여주고 있다.

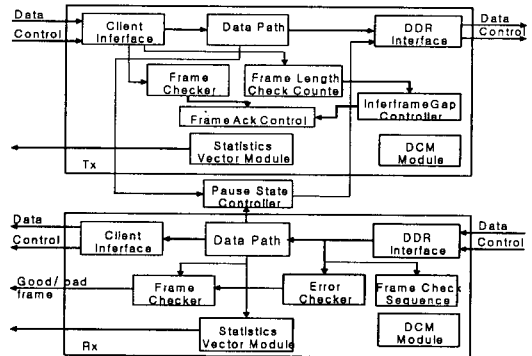


그림 5. 고속 데이터 처리를 위한 MAC 코어의 블록도

Fig. 5 Block diagram for MAC core for high-speed data processing

##### 4.2 제안한 모듈의 성능 평가

4.1절에서 제안한 MAC 코어의 내부 FIFO 적응도를 판별하기 위하여 C언어를 이용해 전체 블록도를 모델링하였다. 입력되는 트래픽에 다양한 Offered Load를 주어 각 Offered Load에 의한 FIFO 크기와 전송률을 확인하였다. FIFO에 대한 적응도를 판별하기 위하여 초기화를 시킬 때 10만개의 초기화 주기와 100만개의 유효주기를 제공하여 검증하였다. 그림 6은 패킷크기별 FIFO 크기변화를 보여주고 있다. 검증을 위해 4가지의 상이한 패킷 길이를 가지는 패킷들이 랜덤 유니폼하게 유입되도록 하였다. 예를 들어 (1488, 438, 1372, 724)의 경우에 패킷 길이가 1488, 438, 1372, 724바이트 등 총 4가지 종류로 이루어졌다는 것을 의미한다. 그림 6에서 보이는 것과 같이 평균 패킷 크기의 변화에 따른 FIFO의 변화량을 확인한 결과 FIFO 크기가 512워드일 경우에는 95%에서 96%정도의 Offered Load에서 패킷 손실이 없이 동작이 가능하였고, FIFO 크기가 1024워드일 경우에는 대략 97%의 Offered Load에서 패킷 손실없이 동작하는 것을 확인하였다. 그림 7은 패킷 크기의 변화에 따른 쓰루풋(Throughput)을 시뮬레이션한 것이다. 시뮬레이션 결과 평균 패킷 크기가 커질수록 쓰루풋 또한 증가하는 것을 확인하였다.

성능 평가를 토대로 본 논문에서는 면적 대 성능비 등을 고려하여 FIFO 크기를 512워드 크기로 설정하여 설계를 수행하였다.

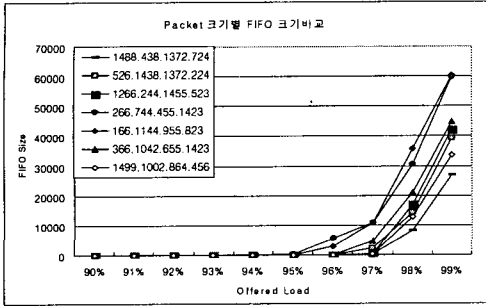


그림 6. 패킷 크기별 FIFO 크기의 변화  
Fig. 6 FIFO sizes according to packet sizes

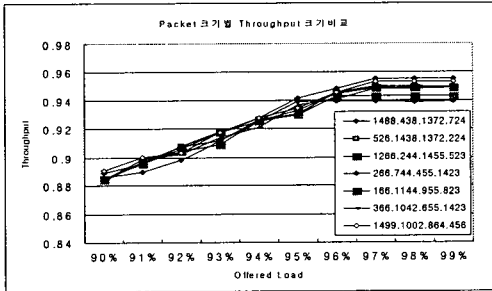


그림 7. 패킷 크기 변화에 따른 쓰루풋의 비교  
Fig. 7 Comparison of Throughput according to packet sizes

### 4.3 전송부의 설계

MAC 코어는 64비트의 데이터와 8비트의 컨트롤 데이터를 연속적으로 입력받아 프레임의 시작인지 끝인지 체크하고 프레임의 길이 정보를 확인한다. 프레임 체크 블록에서 프레임의 끝을 확인하여 IFG(Inter Frame Gap) 블록을 실행하여 IFG만큼의 Idle 프레임을 생성하여 전송하게 한다. IFG를 수행한 후 MAC 코어의 전송부는 데이터를 수신하여 전송 가능한 대기 상태에 존재하게 되고 상위 MAC FIFO로부터 START 신호를 확인하여 데이터를 받아 처리하게 된다. FCS(Frame Check Sequence)[19-20]를 요구할 경우 MAC 코어 내부에서 CRC값을 삽입하여 전송하게 된다. 전송되는 데이터는 XGMII에서 요구하는 컨트롤 정보와 데이터 정보를 조합해서 전송하게 된다. 하위 계층에서 받은 클럭을 이용하여 DCM모듈에서 내부에서 사용가능한 클럭으로 재생성하여 사용한다. 휴지 상태 컨트롤을 통한 플로우 컨트롤을 지원한다.

상위 계층인 MAC\_FIFO로부터 수신한 64비트의 데이터와 8비트의 컨트롤 정보를 수신하여 데이터를 처리하게 되는데, MAC\_FIFO에서 전송할 데이터가 존재함을 확

인하여 TX\_START 신호를 활성화하게 된다. MAC 코어는 현재 데이터 전송이 가능한 때에 TX\_ACK 신호를 활성화하여 MAC\_FIFO로부터 연속적인 프레임을 수신하게 된다. 수신된 데이터의 유효 프레임 정보를 확인하여 프레임의 시작과 끝을 확인하게 된다. 유효 프레임 정보를 확인하여 XGMII 블록으로 전송할 컨트롤 정보를 생성하게 되고 DDR 인터페이스를 거쳐 32비트의 데이터 정보와 4비트의 컨트롤 정보를 하위 계층으로 전송하게 된다. 그림 8은 MAC 코어의 전송부의 DataPath의 블록도를 보여주고 있다. 4단 파이프라인으로 구성되어 있고, MAC 코어의 데이터 처리가 Cut-Through 방식으로 데이터 처리하기 때문에 프레임의 끝을 확인하지 않고 바로 지연없이 전송하게 된다.

Pause 요구가 유입되었을 때 프레임 처리중일 경우 프레임 처리가 종료될때 까지 대기한 후 Pause 프레임을 생성하여 전송하게 된다.

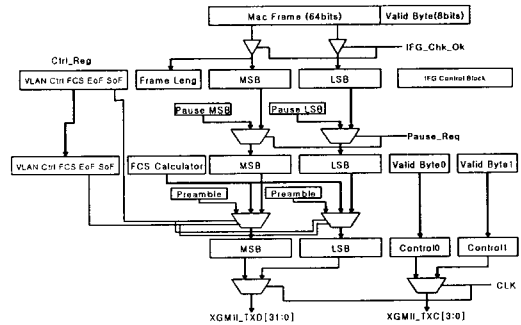


그림 8. MAC 코어의 데이터패스 블록도  
Fig. 8 Block diagram of data path for MAC

Pause 프레임 처리중일 경우 TX\_START 신호가 활성화되었을 때 IFG 모듈에서 플로우 컨트롤을 수행하여 데이터 전송을 일시 중지 시키게 동작한다. 플로우 컨트롤이 종료한 후 ACK 신호를 통하여 데이터 전송을 요구하게 된다.

### 4.4 수신부의 설계

유입되는 32비트의 데이터와 4비트의 컨트롤 정보를 DDR 인터페이스에서 단일 데이터율(Single Data Rate)로 변환하여 64비트의 데이터와 8비트의 컨트롤 정보로 변환한다. 변환된 데이터 정보와 컨트롤 정보를 디코딩하여 프레임의 시작과 프레임의 끝, 그리고 프레임 전송도중 발생한 에러정보를 추출하게 되고, FCS연산을 수행하여

FCS연산에 오류가 발생하였는지 확인하게 되고 64비트의 데이터와 8비트의 데이터 유효정보로 변환하여 전송하게 된다. 전송되는 데이터의 정보를 수집하고 하나의 프레임이 전송 종료 되었을 때 상태 정보를 상위 계층으로 전송하여 수신되는 데이터를 확인하게 된다. 하위 계층으로부터 수신한 XGMII\_RX\_CLK를 받아서 DCM 모듈에서 MAC Rx 모듈에서 사용 가능한 클럭으로 변환하여 사용하게 된다. 수신되어 프레임으로 정합한 후 자체 FCS연산을 통하여 에러 유무를 확인하게 된다. 확인 후 정상 프레임으로 확인될 경우 RX\_Good\_Frame 신호가 활성화하게 되고, 에러가 발생하였을 경우, 상위에서 프레임을 버릴수 있도록 RX\_Bad\_Frame를 활성화하여 상위 계층에 전송하게 된다.

MAC RX 모듈에서는 컨트롤 정보와 입력되는 데이터를 상시 확인하여 프레임의 시작을 알리는 컨트롤 정보와 컨트롤 데이터를 확인하여 프레임 수신을 시작하게 된다. 수신된 데이터의 프리앰블 필드를 제외한 모든 필드를 파이프라인 시키면서 MAC\_FIFO에서 요구하는 64비트의 데이터로 재조립 해주며 8비트의 컨트롤 정보를 8비트의 데이터 유효 정보로 변환해준다. 3단 파이프라인으로 구성되어 있으며, 각 파이프라인에서는 현재 프레임의 에러 정보와 컨트롤 정보를 가지고 있으며 최종적으로 MAC\_FIFO로 전송하고자 할 때 데이터 유효 정보를 생성하는데 사용한다. 그림 9는 XGMII에서 DDR로 데이터를 수신하여 데이터를 처리하는 데이터 패스의 상세 회로를 보여주고 있다.

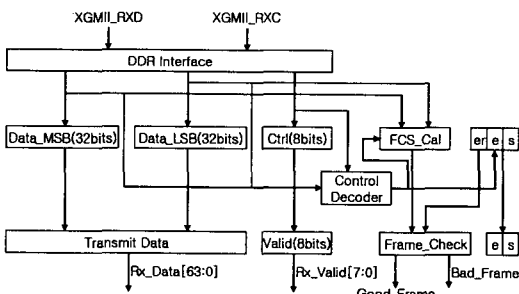


그림 9. 데이터 패스의 상세 회로도  
Fig. 9 Detailed circuit for data path

5.5 상태 정보수집 모듈 설계

MAC 송/수신모듈은 하위계층으로부터 수신된 프레임에 대한정보를 상위계층으로 전송하게 된다. 전송되는

시점은 프레임이 종료되는 시점으로 정보 유효 신호를 활성화시키고 데이터를 전송하게 된다. 그림 10은 프레임 상태정보를 확인하는 블록도를 보여주고 있다. MAC 송/수신모듈은 수신되는 프레임의 종료시점을 확인하여 상태 정보를 수집하여 STATISTICS\_VALID신호와 같이 상위 계층으로 전송하게 된다.

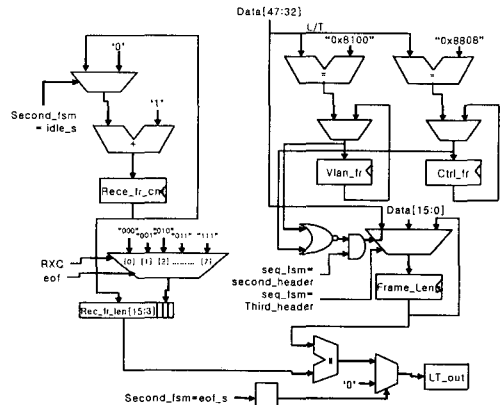


그림 10. MAC 상태 정보수집 블록도  
Fig. 10 Status information gathering block for MAC

상태 정보 중 Byte\_valid 정보는 프레임이 전송되는 도중에만 유효한 값을 가지고 그 외에는 값을 가지지 않는다. 프레임의 정보를 확인하기 위해서는 MAC 프레임의 헤더 정보를 분석하여야 하는데 총 2~3 클럭 사이클에 MAC 프레임 헤더 정보가 송/수신되게 된다. 프레임 헤더를 확인하기 위해 프레임이 시작됨과 동시에 동기되는 프레임 헤더 상태머신을 두어 프레임 상태정보를 확인하였다. 프레임 헤더 카운트 상태머신에 의해 각 사이클 별로 주소 정보와 프레임 길이, VLAN Tag 등 여러 종류를 확인하여 프레임의 송/수신이 종료 되었을 때의 값과 비교하여 상태정보를 설정하게 된다.

V. 시뮬레이션을 통한 검증 및 설계 분석

5.1 전송부의 파형 분석

그림 11은 전송부 블록의 시뮬레이션 파형을 보여주고 있다. 먼저 상위 계층에서는 WrClk 클럭 신호를 사용하여 WrData 버스상으로 데이터와 데이터의 유효성을 통보하는 WrVData 버스에 실어 보내고 FIFO 쓰기 인에이블신

호인 Wren 신호를 이용하여 FIFO에 데이터를 쓴다. FIFO는 이에 대한 응답으로 WrAck 신호를 비롯하여 FIFO에 데이터가 채워져 있는 상태인 AlmostFull, Full, Overflow 신호를 상위 계층에 전달한다. FIFO에 유효 데이터가 도착하면 Cut-Through 모드로 XGMII\_tx\_clk, XGMII\_tx\_data, XGMII\_txc 출력 포트를 통해 외부로 신호를 전송하게 된다.

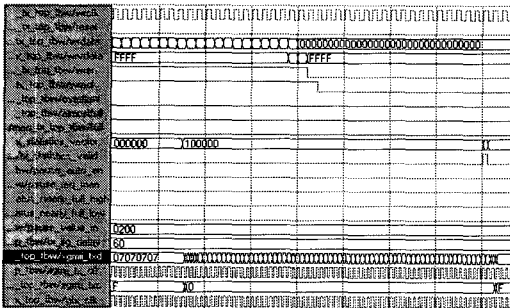


그림 11. MAC 전송부 블록의 시뮬레이션 파형  
Fig. 11 Simulation waveform for MAC sending part

전송 중에 수신부로부터 Pause 요청을 받을 경우에는 전송부는 Pause\_auto\_en 신호를 활성화하고 전송을 중지하며, 전송부 측에서 필요시에는 인위적인 Pause\_req\_man 신호를 발생하여 전송을 중지할 수 있도록 한다. 전송부가 XGMII 인터페이스를 통해 완전한 패킷을 전송하면 statistics\_valid 신호를 활성화하면서 Tx\_statistics\_vector를 통해 패킷의 전송 상태를 상위 계층에 통보하게 된다.

5.2 수신부의 파형 분석

그림 12는 MAC 수신부 블록의 시뮬레이션 파형을 보여주고 있다. XGMII\_rx\_clk, XGMII\_RXC, XGMII\_rxd, XGMII\_rxc 버스 신호를 통해 XGMII 인터페이스를 통해 패킷을 수신하기 시작한다.

수신되는 패킷 데이터는 내부 데이터를 통해 내부 FIFO에 데이터를 저장한다. 수신을 완료하였을 때 수신된 패킷의 정상 여부에 따라 Rx\_Good\_Frame 신호나 Rx\_Bad\_Frame 신호를 발생하며, Rx\_statistics\_vector를 통해 프레임의 형태와 에러에 대한 정보를 상위 계층에 전달한다. 정상적으로 FIFO에 쓰여진 데이터는 Rx\_Data, Rx\_Data\_Valid, RdClk, Rden 신호를 사용하여 상위 계층에서 읽어가게 된다. 이 때 내부 FIFO는 FIFO의 내부 상태를 RdAck, Underflow, Empty, AlmostEmpty 등의 신호를 사용하여 상위 계층에 통보하는 역할을 수행하게 된다.

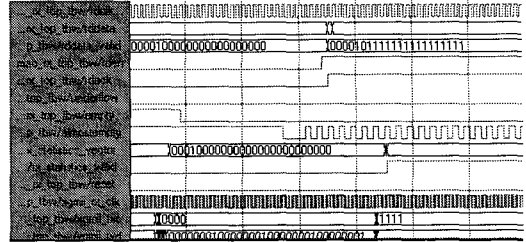


그림 12. MAC 수신부 블록의 시뮬레이션 파형  
Fig. 12 Simulation waveform for MAC receiving part

5.3 설계 분석

설계된 MAC 코어는 사용자 인터페이스는 64비트 데이터 버스와 하위 계층과의 데이터 교환시에는 32비트 DDR 인터페이스를 통해 이루어진다. 표준안에서 요구하는 MAC 코어의 주파수는 156.25MHz 이상을 요구하여 10Gbps 이더넷을 지원하게 된다.

표 2. MAC 코어의 설계 결과 비교  
Table 2. Comparison for designed result of MAC

비교모델 비교값	표준안 요구사항	Xilinx CS201	Our Design
Internal Clock	156.25MHz	156.25MHz	168.54MHz
XGMII Interface	156.25MHz	156.25MHz	168.54MHz
Slices	n/a	1061	950
Data Path	n/a	n/a	Tx : 4단 pipeline Rx : 3단 pipeline
Data Forwarding	n/a	Cut-Through	Cut-Through

설계한 모듈은 라인당 168.549MHz를 지원하여 CS201 MAC보다 나은 성능을 보여주고 있으며, 고속의 응용분야에 활용에 효과적이다. 또한 본 논문에서 고속화 처리를 위해 전송부 데이터 패스는 4단 파이프라인을 적용하였고, 수신부 데이터 패스는 3단 파이프라인을 적용하여 설계하였다. 그리고, 상위 계층으로부터 완전한 데이터 패킷을 수신한 후 전송하는 저장 및 포워딩 방식을 사용하지 않고 효과적인 전송이 가능한 패킷의 일부만 수신하였을 때 XGMII 인터페이스를 통해 전송을 개시하는 Cut-Through 방식을 채택하여 설계에 적용하였다. 이 밖에도 VLAN을 지원하고, IEEE 803.3-2002에서 규정한 최대 프레임 사이즈인 1518바이트 패킷보다 큰 패킷을 에러없이 보낼 수 있는 Jumbo 프레임 모드를 지원하여 전송의 효율성을 재고하였다. 표 2는 MAC 코어의 설계 모듈과 Xilinx

에서 설계된 CS201 모듈[13]의 비교를 보여주고 있다.

## VII. 결 론

전 세계 데이터 전송의 대부분이 이더넷 연결로 시작되고 끝나게 되는 시대를 맞이하였다. 오늘날 우리는 전통적인 네트워킹 정설에 문을 열고 급격한 E-Business와 저가의 IP 서비스에 대한 요구에 의해 자극 받고 있는 이더넷 르네상스의 중앙에 있다. 서비스 제공자는 매우 높은 레벨의 안정성을 유지하면서도 네트워킹 연결의 총 비용을 간단하게 하고 줄일 수 있고, 그리하여 이익이 되는 서비스 차별화를 가능케 하는 고용량의 솔루션을 찾고 있다.

이더넷은 이제 더 이상 LAN에서만 설계되지 않는다. 10Gbps 이더넷은 속도와 거리에서 잘 정립된 IEEE 802.3 표준의 전형적인 발전이다. 10Gbps 이더넷은 이더넷의 증명된 가격과 경제성을 대도시 또는 원거리 통신망까지 확장될 것으로 예견되고 있다. 이에 발맞춰 본 논문에서는 10Gbps 이더넷의 핵심 모듈인 MAC 코어를 설계하여 검증을 수행하였다. 10Gbps 이더넷에서는 전 이중방식의 통신만 지원하므로, 전송 중 충돌감지 기능이 요구되지 않고, 다만 속도정합을 위한 ifSresh 등이 추가 되었다. WAN 전송 기술인 SONET/SDH 전송 기법은 9.958Gbps를 요구하며, PCS 부계층 아래 WIS 계층을 추가함으로써 10Gbps 이상을 지원하는 본 연구의 MAC 코어를 사용하여 응용이 가능하다.

10Gbps의 전송 속도를 지원하기 위해 MAC 코어는 156.25MHz 이상의 내부 클럭이 지원되어야 한다. 그리고 휴지상태의 관리와 IFG를 통한 플로우 컨트롤이 지원되고, 내부의 FCS를 통한 에러검출이 가능하도록 설계하였다.

본 논문에서는 10Gbps 이더넷의 데이터 링크 계층에서 적용 가능한 MAC 코어를 C언어를 이용한 성능평가를 토대로 하드웨어 설계 언어인 VHDL 언어를 이용하여 설계하여 FPGA 칩인 Xilinx Vertex 1000E에 다운로드하여 칩 수준의 테스트를 실시하였다. 설계된 모듈의 내부 지연요소가 가장 많은 데이터 패스는 전송부 경우는 4단 파이프라인, 수신부 경우에는 3단 파이프라인을 적용하여 고속의 데이터 처리를 구현하였다. 내부 주파수 168.549MHz에 동작하여 고속의 네트워크 장비에 효과적으로 적용가능

하다. 그리고, 설계된 MAC 코어는 10Gbps 이더넷의 스위칭 장비의 인터페이스로 응용이 가능하고, 10Gbps 이상의 고속의 데이터 처리가 요구되는 응용 분야에 적합할 것으로 사료된다.

## 참고문헌

- [1] 이찬구, 김대영, "10기가비트 이더넷 기술동향," 한국통신학회논문지 16권 112호 pp.59-69, 1999년 12월
- [2] 김대영, 성기순, "초고속 이더넷," Telecommunications Review 제10권 1호, pp93-105, Jan. 2000.
- [3] David G. Cunningham, Ph.D. & William G. Lane, Ph.D. *GIGABIT ETHERNET NETWORKING*, Macmillan Technical publishing, 1999.
- [4] 10GEA, "10Gigabit Ethernet White Paper," May. 2002.
- [5] IEEE Draft P802.3ae/D5.0, "Media Access Controp (MAC) Parameters, Physical Layer, and Management Parameters for 10Gb/s Operation," May. 2002.
- [6] IEEE Std 802.3ae/D5.0, "Supplement to Carrier Sense Multiple Access with Collision Detection (CSMA/CD) A ccess Method & Physical Layer Specifications," 2002.
- [7] 김준영, 손승일, "고속 이더넷 MAC 설계를 위한 성능 파라미터에 대한 연구," 한국해양정보통신학회 2004 춘계 학술대회, Vol.8 No.1, pp674-677, May 2004.
- [8] 이동훈, 손승일, "MAC용 TX FIFO 인터페이스 블록의 설계," 한국인터넷정보학회 2004 춘계 학술대회, Vol.5 No.1, pp253-256, May 2004.
- [9] 이동훈, 손승일, "MAC에 적용 가능한 Receive FIFO 블록의 설계," 한국해양정보통신학회 2004 춘계 학술대회, Vol.8 No.1, pp647-650, May 2004.
- [10] 박노식, 손승일, "10GbE용 MAC Core의 전송부 설계," 한국해양정보통신학회 2004 추계 학술대회, Vol.8 Num.2, pp457-460, Oct. 2004.
- [11] 박노식, 손승일, "10GbE용 MAC Core의 수신부 설계," 한국해양정보통신학회 2004 추계 학술대회, Vol.8 No.2, pp1061-1064, Oct. 2004.
- [12] T. Yazaki, T. Kanetaki, "High-Speed IPv6 Router/Switch Architecute," Proceedings of SAINTW'04, 2004
- [13] Xilinx, "10-Gigabit Ethernet MAC with XGMII or XAUI V4.0," Dec. 2003.



- [14] 10 Gigabit Ethernet Alliance (10GEA). <http://www.10gea.org/>
- [15] Mark Norris, *Gigabit Ethernet Technology and Applications*, Artech House, 2003.
- [16] Xilinx, "XGMII Using the DDR Registers, DCM, and SecectI/O-Ultra Features," July, 2002.
- [17] Xilinx, "XAUI Core v4.0", Dec. 2003.
- [18] Xilinx, "Ten Gigabit Ethernet MAC FIFO," May. 2003.
- [19] Shu Lin, Daniel J./ Costello, Jr., *Error Control Coding : Fundamentals and Applications*, Prentice Hall, 1983.
- [20] [http://www.repairfaq.org/filipg/LINK/F\\_crc\\_v32.html](http://www.repairfaq.org/filipg/LINK/F_crc_v32.html)

### 저자소개



손 승 일(Seung-II Sonh)

1989년 연세대학교 전자공학과(학사)

1991년 연세대학교 대학원 전자공학과(석사)

1998년 연세대학교 대학원 전자공학과(박사)

2002년~현재 한신대학교 정보통신학과 부교수

※ 관심분야 : ATM 통신 및 보안, ASIC 설계, 영상신호 처리칩