

# IP over ATM 프로토콜의 VHDL 구현

## (VHDL Implementation of IP over ATM Protocol)

崔秉台\*, 崔竣均\*\*, 金在根\*\*, 高聖濟\*

(Byung-Tae Choi, Jun-Kyun Choi, Jae-Guen Kim, and Sung-Jea Ko)

### 요 약

본 논문에서는 ATM (Asynchronous Transfer Mode)네트워크에서 기존 인터넷에서 사용하는 TCP/IP (Transmission Control Protocol/Internet Protocol) 프로토콜을 수용할 수 있는 방안으로 IETF (Internet Engineering Task Force)에서 제시한 IP over ATM 규격을 따르는 IP (Internet Protocol) 프로토콜을 VHDL을 사용하여 탐다운 방식으로 설계하였다. 설계한 IP 프로토콜은 송신부와 수신부를 완전히 분리함으로써 전이중(full-duplex) 방식으로 동작할 수 있으며, STM-1 (Synchronous Transfer Mode-1)급에 해당하는 155.52Mbps의 속도로 송수신 데이터를 처리할 수 있다. 또한 병렬 처리구조를 채택하여 처리시간을 줄였으며, 테이블 검색에 걸리는 시간을 최소화하기 위하여 라우팅 테이블 검색과 ATMARP (ATM Address Resolution Protocol) 테이블 검색을 결합하여 하나의 LANCAM (Local Area Network Content Addressable Memory) 소자를 한 번만 검색하도록 하는 구조를 제안하였다. 설계한 IP 프로토콜 회로를 논리 합성하고 시뮬레이션한 결과 전체 IP 데이터그램에 대하여 송신부와 수신부에서 각각 48.5클럭과 29.5클럭의 지연시간으로 155Mbps의 송수신 데이터를 처리할 수 있음을 확인하였다.

### Abstract

In this paper, a VHDL implementation method for the Internet Protocol (IP) placed on top of ATM, so called IP over ATM, is presented. The proposed implementation method employs a parallel processing architecture to reduce the processing time and offers 155.52Mbps (STM-1) interface with the full-duplex mode for the ATM-based network. Furthermore, in order to minimize the search time for the table look-up, a LANCAM-based structure combining the routing table with the ATMARP table is proposed. The VHDL simulation results show that this proposed method can transmit (receive) at 155.52Mbps with delays of 48.5 clocks (29.5 clocks).

### I. 서 론

정보화 시대를 맞이하여 데이터 통신의 수요는 폭발적으로 증가하고 있다. 더욱이 정보통신의 수단이 멀티미디어화됨에 따라서 통신 데이터들이 다양화되어 갈 뿐만 아니라 데이터 통신의 대역폭도 급격히 증가하게

되었다. 이러한 요구에 부응하기 위하여 고속 멀티미디어 통신을 위한 연구들이 진행되고 있다. 데이터 통신을 위한 트랜스포트/인터넷워킹 프로토콜로서 현재까지는 TCP/IP가 그 주류를 이루어 왔는데<sup>[11]-[14]</sup>, 최근 IP 주소공간의 부족을 해결하고 장래를 위하여 몇 가지 문제점을 보완한 차세대 IP (next generation IP 혹은 IP version 6)가 IETF에 의해 규정되어 기존의 IP를 교체하려는 움직임이 있다. 또한, 고속 멀티미디어 통신을 위한 새로운 트랜스포트 및 네트워크 계층 프로토콜들에 대한 연구들도 진행되고 있으나 오랫동안 지배적으로 확산되어 있는 TCP/IP를 새로운 멀티미디어 프로토콜이 완전히 자리바꿈하기는 매우 어려운 것

\* 正會員, 高麗大學校 電子工學科  
(Dept. of Elec. Eng., Korea University)

\*\* 正會員, 韓國電子通信研究所 通信網構造研究室  
(Electronics and Telecomm. Research Institute)  
接受日字: 1996年8月7日, 수정완료일: 1997年1月14日

이다. 따라서, 앞으로 전개될 B-ISDN과의 연동으로 초고속 정보고속도로를 구축하는데 있어서도 TCP/IP는 여전히 핵심 프로토콜로서 그 역할을 담당하게 될 것이며 이를 위해서는 TCP/IP의 고속화가 절실히 필요한 실정이다.

한편, 최근에는 B-ISDN이 도입되기 시작함에 따라 기존의 다양한 데이터 통신 서비스들을 ATM망을 통하여 서비스하는 방안들이 제시되고 있다. 그 중에서 ATM Forum에서 제시한 LAN Emulation에서는 Ethernet이나 토큰 링이 LAN-to-LAN Converter를 통하여 ATM망에 의해 서비스되게 함으로써, 기존의 데이터망에 아무런 수정 없이 ATM망과 연동할 수 있게 한다<sup>[5], [6]</sup>. 또, IETF의 RFC1577에서는 Classical IP over ATM을 제시하였는데<sup>[6]-[8]</sup>, 이 접근 방법은 프로토콜의 수행효율을 높일 수 있으나 기존의 데이터망을 그대로 수용할 수 없다는 진화상의 문제점이 있다. 이 두 가지의 접근방법중에서 어떤 것이 최종적으로 지배적으로 될 지는 미지수이나 기존의 TCP/IP를 이용한 데이터망이 고속의 ATM망에 연동되어 사용되는 것만은 확실하다. 이 경우 LAN-to-ATM 연동 지점에서의 라우터나 ATM망내의 데이터 통신 서비스 노드에서는 고속의 프로토콜 처리능력이 필요하게 된다. 즉, 현재의 TCP/IP를 사용한 데이터망들을 가정할 때 고속의 TCP/IP 수행능력이 필요하게 된다.

이러한 배경에서 본 논문에서는 현재의 TCP/IP를 사용한 기존의 데이터망이 고속의 ATM망에 연동되어 사용될 것을 가정하고, 고속화된 광대역 통신망에서 IP 혹은 TCP/IP 프로토콜이 그 역할을 원활히 수행할 수 있도록 고속화하기 위한 연구를 수행하였다. 기존의 TCP/IP 프로토콜이 ATM망과 연동하는 방법으로는 IP over ATM 규격을 선택하고, IP over ATM 규격을 따르는 IP 프로토콜을 하드웨어로 설계하였다. 내부 구조에서 처리 시간이 많이 걸리는 부분에 대해서는 병렬 처리 구조를 제시하였으며, IP 데이터그램을 송신하는 경우에 사용하는 라우팅 테이블 처리과정과 ATMARP 테이블 검색과정을 결합하여 처리 속도를 높일 수 있는 구조를 아울러 제안하였다. 설계 언어는 IEEE 표준의 하드웨어 기술 언어인 VHDL을 사용하여<sup>[9], [10]</sup> 호환성을 확보할 수 있었다. 본 논문에서 설계한 IP 프로토콜은 STM-1급의 ATM망과 속도의 저하없이 바로 인터페이스될 수 있다. 최종적으로는 설

계된 IP 프로토콜 하드웨어를 시뮬레이션해 봄으로써 설계된 회로가 정상적으로 동작하는 것을 확인하였다.

본 논문의 구성의 다음과 같다. II장에서는 Classical IP over ATM 모델에 대해서 설명하고, III장에서는 본 논문에서 제안하는 병렬 처리 IP 프로토콜 구조와 고속의 라우팅 테이블 및 ATMARP 테이블 검색구조에 대해서 설명하고, IP over ATM 규격을 따르는 IP 프로토콜 전체의 하드웨어 설계에 대해서 다루었다. IV장에서는 본 논문에서 설계한 IP 프로토콜에 대한 회로 합성 및 시뮬레이션을 통한 동작검증을 설명하였으며, 끝으로 V장에서 결론을 맺었다.

## II. IP over ATM 프로토콜 규격

IETF RFC 1577<sup>[7]</sup>에서는 IP 프로토콜을 사용하는 단말과 LAN 세그먼트, 그리고 IP 라우터를 연결하기 위하여 ATM을 사용할 수 있도록 하는 Classical IP over ATM을 규정하고 있다. 여기에서는 기존의 IP 서브넷 아키텍처를 그대로 유지하면서 ATM을 단지 넓은 대역폭을 제공하는 하나의 링크 서비스로서 간주한다. 이것은 IP 서브넷간의 상호 연결이 기존의 IP 라우터에 의해서 행해지며, IP 패킷의 전달은 이미 잘 확립되어 있는 라우팅 기술과 프로토콜에 의해서 이루어진다는 것을 의미한다. 즉, IP over ATM을 사용함으로써 기존의 TCP/IP 프로토콜은 그대로 사용하면서 ATM망의 빠른 전송속도를 이용할 수 있는 것이다.

IP over ATM 표준 모델은 ATM망위에서 기존 IP 호스트의 요구사항을 만족시키기 위해서 IP 서브넷에 대응하는 논리적 개념인 LIS (Logical IP Subnetwork)를 도입한다. 여기에 하나의 LIS내에 적어도 하나 이상의 ATMARP (ATM Address Resolution Protocol) 서버 및 ATM 라우터가 존재하여, 이들이 각각 IP 어드레스와 ATM 어드레스간의 매핑 및 다수의 LIS간의 IP 패킷 라우팅을 수행하는 것을 골격으로 한다. 물론 이러한 방식은 서로다른 LIS에 존재하는 IP 호스트간의 직접적인 ATM 연결을 제공하지 못하고 디폴트로 ATM 라우팅 서버를 경유해야 하는 단점이 없지 않다. 이를 개선하기 위하여 IETF의 ROLC 그룹에서 NHRP (Next Hop Resolution Protocol)에 관한 작업을 하고 있지만 아직 완전한 스펙이 제시되지 않았기 때문에 본 연구에

서는 RFC 1577의 Classical IP over ATM을 기본으로 한다.

1. LIS

IP over ATM은 LIS라고 하는 논리적인 IP 서브넷을 기반으로 운용되는데, LIS는 기존의 LAN처럼 호스트와 라우터로 구성된다. 각 LIS는 동일한 ATM 망에 있는 다른 LIS들과 서로 독립적으로 동작하며 통신한다. 즉, 동일 LIS내의 두 단말 시스템간에는 직접 데이터를 주고받을 수 있지만 다른 LIS내의 IP멤버와는 반드시 라우터를 통해야만 한다. 이 라우터는 ATM 망에 접속된 ATM 단말로서 하나 이상의 LIS들의 구성원이 된다. 이러한 구성을 통하여 동일 ATM망 위에 여러 개의 독립된 LIS들이 존재하게 된다.

2. LLC/SNAP (Logical Link Control/ Sub-network Access Protocol) 인캡슐레이션

IP over ATM에서는 IEEE 802.2 LLC/SNAP 인캡슐레이션 방식을 사용하고<sup>[71,111]</sup>, 인캡슐레이션된 PDU (Protocol Data Unit)들은 AAL5의 CPCS (Common Part Convergence Sublayer) PDU의 유효부하 (payload) 필드를 통하여 전달된다. LLC 인캡슐레이션을 사용하는 이유는 하나의 VC위에서 여러 개의 프로토콜 데이터들을 전달하기 위해서이다. 그림 1에서는 LLC/SNAP 인캡슐레이션에서의 헤더 형식을 보여주고 있는데, SNAP내의 PID 필드의 값들을 통하여 서로 다른 프로토콜들을 구별함으로써 AAL5 상위에서 하나의 VC에 대해 여러개의 프로토콜을 사용할 수 있게된다.

필드	값 (16진수)	크기 (바이트)	의미	
LLC	AAAA03	3	SNAP헤더임을 명시	
SNAP	OUI*	000000	3	라우팅 방식 사용
	PID (Protocol ID)	0800	2	프로토콜 구별
		0835 0806		
PDU		최대 9180	데이터 부분	

그림 1. LLC/SNAP 인캡슐레이션 헤더의 형식.  
Fig. 1. Format of LLC/SNAP Encapsulation header (\*: Organization Unique Identifier).

3. PVC 사용 환경에서의 주소 해석

ATM LIS내의 주소 해석은 ATMARP 및 InATMARP를 사용하여 이루어진다<sup>[111],[112]</sup>. ATMARP는 ARP프로토콜을 기본으로 하여, 유니캐스트 서버 ATM환경에서 ARP를 지원할 수 있는 확장 프로토콜이다. InATMARP는 ATM망에 적용되는 것 이외에는 원래의 InARP프로토콜과 동일하다. 이들 프로토콜의 사용은 PVC (Permanent Virtual Circuit)를 사용하는지 SVC (Switched Virtual Circuit)를 사용하는지에 따라 다르다.

PVC환경에서는 모든 IP멤버들은 LLC/SNAP 인캡슐레이션을 사용하는 VC에 대하여 InATMARP를 사용하여야 한다. 그림 2에서는 PVC로 연결된 ATM망 환경에서 InATMARP를 이용한 주소 해석 과정을 보여주고 있다. 먼저 ATM단말 A에서 B로 InATMARP 요구 패킷을 PVC를 통하여 송신한다. 수신측 단말 B는 패킷으로부터 송신측의 IP 주소를 알아내고 VC와 연결시키며, InATMARP 응답 패킷을 만들어 단말 A로 돌려보낸다. 응답 패킷을 수신한 단말 A는 B에 대하여 VC와 IP로 구성되는 어드레스 테이블을 만들 수 있게 된다.

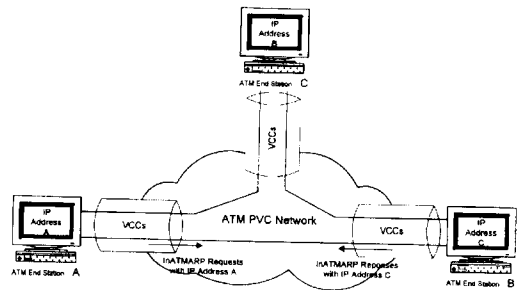


그림 2. PVC 네트워크에서의 주소 해석  
Fig. 2. Address resolution in ATM PVC network (VCCs : Virtual Channel Connections).

III. 제안한 IP 프로토콜 하드웨어

1. IP 하드웨어의 구조 및 설계 규격

본 연구에서는 ATM망이 PVC환경에서 STM-1급에 해당하는 155.52Mbps의 속도로 동작하며, IP 상위의 프로토콜은 소프트웨어로 동작한다고 가정하고, 이러한 가정 위에서 IP over ATM 규격을 따르는 IP 프로토콜을 IEEE 표준 하드웨어 기술 언어인 VHDL을 이용하여 탐다운 방식으로 설계하였다. 여기서 탐

운 설계방식이란 시스템의 특성을 상위 레벨의 개념적인 단계로부터 기술하기 시작하여 점차적으로 가장 기본적인 회로 구성 요소에까지 점점 세분화된 서브 모듈들을 통해서 단계적으로 표현하는 기법을 말한다.

IP 프로토콜 하드웨어는 그림 3에서 보여주는 것처럼 크게 호스트 CPU와의 통신을 담당하는 DMA 인터페이스부, IP 프로토콜 데이터그램을 처리하는 IP 송수신부, 그리고 LLC/SNAP 인캡슐레이션 처리부로 나누어 볼 수 있다. 그림 3에서 점선 내부 블록이 본 논문에서 설계하는 IP 프로토콜을 나타낸다. 설계하는 IP 프로토콜 하드웨어는 전이중 방식으로 동작할 수 있도록 하기 위하여 송수신부를 분리하였으며, 송신부와 수신부에서 각각 동시에 처리할 수 있는 부분에 대해서는 병렬 처리 구조를 채택하여 처리속도를 높일 수 있도록 하였다. 그림에서 화살표는 데이터의 흐름을 나타낸다.

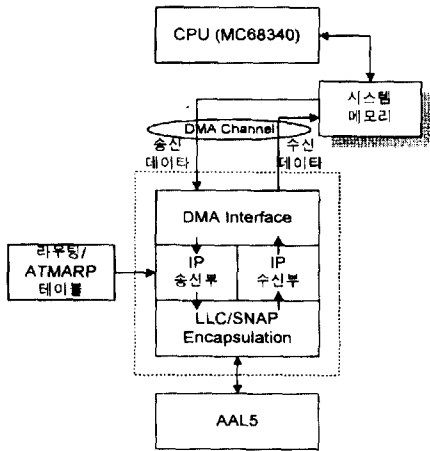


그림 3. IP 하드웨어 전체 구성도  
Fig. 3. Block diagram of IP hardware.

표 1. IP 프로토콜의 하드웨어 설계 규격  
Table 1. Hardware design specification of IP protocol.

항목	규격
데이터 처리 속도	STM-1 (155.52Mbps)
ATM 연결 형태	PVC
데이터그램 송수신 방식	전이중 동작
IP 내부 데이터 처리 단위	8 비트
설계언어	VHDL
호스트 CPU	MC68340
CPU-IP간 통신 방식	16비트 DMA 방식
라우팅 · ATMARP 테이블	MU9C1480 LANCAM

표 1에서는 본 논문에서 설계한 IP 프로토콜의 설계 규격을 보여주고 있다.

## 2. 송신부 설계

VHDL 구현에 있어서 세부 모듈들을 통합하는 송신부 통합 모듈은 Structural 모델로 설계하고, 각각의 세부 모듈들은 Behavioral 모델 및 Dataflow 모델로 설계하였다. 각 모듈은 VHDL 엔티티(Entity) 하나로 구성되어 있다.

그림 4에서는 송신부의 모듈 구성을 나타내고 있으며, 그림내의 화살표는 데이터패스를 보여주고 있다. 송신부는 DMA Input Interface Handler 모듈에서 시작하여 LLC/SNAP Encapsulation Handler까지 모두 10개의 세부 모듈로 이루어져 있다. 이들 중에서 처리 시간을 많이 구하는 모듈은 체크섬 계산<sup>[14]</sup>하는 모듈과 라우팅을 처리하는 모듈인데, 이들 모듈들은 서로 입출력 데이터를 주고 받지 않고 시간상으로 데이터를 독립적으로 처리할 수 있다. 따라서 라우팅 처리 모듈 (Routing & ATMARP Handler)과 체크섬 계산을 담당하는 모듈 (Checksum Calculator), IP 헤더를 만드는 모듈 (Header Organizer)에 대하여 병렬 구조로 입력을 동시에 받아서 데이터를 처리하도록 설계하였다. 이렇게 처리된 결과는 Header Merger 및 Data Adder 모듈과 Destination Demultiplexer 모듈에서 통합되게 된다.

송신부는 호스트 CPU의 DMA 채널을 통하여 데이터를 수신하면 동작을 시작하게 된다. Protocol Demultiplexer 모듈에서는 수신 데이터의 헤더를 조사하여 만약 InATMARP 데이터패킷이면 Protocol Multiplexer로 바로 전달하고, IP 데이터그램을 위한 데이터이면 IP 프로토콜 처리를 시작하게 된다. 라우팅 처리와 IP 헤더 구성 및 체크섬 계산은 병렬로 처리되고, 그 결과를 Header Merger & Data Adder 모듈에서 통합한다. 이때 IP 데이터그램에서의 순수 데이터 부분은 IP 프로토콜에서 직접 처리하는 부분이 아니므로 위의 병렬 처리 모듈 상위의 모듈에서부터 직접 입력받는다. Destination Demultiplexer 모듈에서는 원격지 호스트로 보내는 데이터그램과 로컬 호스트로 보내는 데이터그램을 구별하여, 로컬 호스트 데이터그램이면 곧바로 수신부로 보내준다. 이때 만약 수신부에서 원격지로부터의 입력 데이터그램을 처리중이라면 데이터를 출력하지 않으므로써, 수신부에서의 데이터 충돌

을 방지한다. 원격지로 보내는 데이터그램은 Protocol Multiplexer모듈에서 InATMARP패킷과 다중화되고 마지막으로 LLC/SNAP Encapsulation Handler모듈에서 LLC/SNAP헤더를 붙여서 ATM의 AAL5계층으로 보내지게 된다.

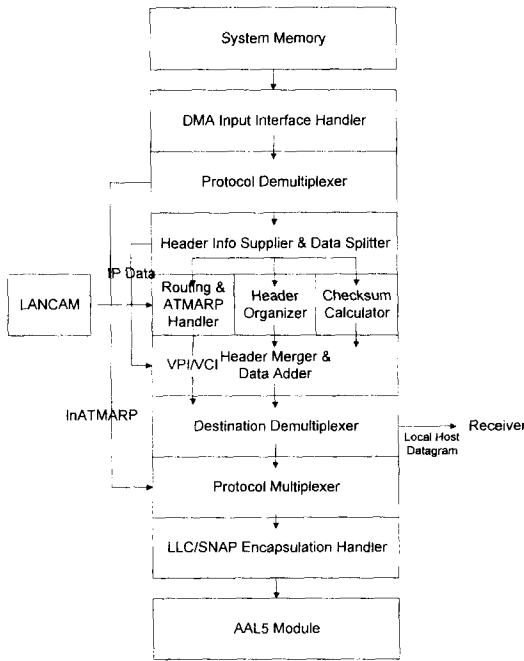


그림 4. 송신부 모듈 구성도  
Fig. 4. Block diagram of sending part.

전체 IP 하드웨어에서 가장 많은 처리시간을 요구하는 부분은 송신부의 라우팅 처리 부분이다. IP프로토콜을 소프트웨어로 구현하는 경우에는 라우팅 처리시에 테이블을 순차적으로 검색해야하기 때문에 많은 시간이 소모되며, 또한 검색에 걸리는 시간이 일정하지 않게 된다. 따라서 이 부분을 하드웨어로 구현함에 있어서 큰 문제가 된다. 본 논문에서는 테이블내의 모든 엔트리에 대해서 병렬로 한 번에 테이블 내의 모든 엔트리를 검색할 수 있는 LANCAM (Local Area Network Content Addressable Memory) 소자를 사용하여<sup>13)</sup> 테이블 처리 시간을 크게 줄일 수 있었다. 또한 소프트웨어로 구현하는 경우에는 검색에 걸리는 시간이 경우에 따라 다르지만, 본 연구에서는 LANCAM을 사용함으로써 정해진 시간에 검색 결과를 얻을 수 있었다.

한편, PVC를 사용하는 IP over ATM에서는 라우팅 결과로 알아낸 IP주소를 ATMARP 프로토콜 처리

를 통하여 해당하는 VC값으로 바꾸어주어야 한다. 이를 위하여 ATMARP프로토콜은 IP 주소를 VC값으로 바꾸어주는 ATMARP 테이블을 유지해야 한다. 이 ATMARP 테이블은 VC값에 해당하는 IP 주소를 찾아주는 InATMARP 프로토콜을 이용하여 구성한다. 본 논문에서는 이러한 라우팅 처리와 ATMARP테이블 찾기를 한 단계로 결합함으로써 테이블을 검색하는 시간을 절반으로 줄일 수 있는 방법을 제안하였다. 이러한 과정은 Routing and ATMARP Handler모듈에서 처리하는데, 이를 요약하면 그림 5와 같다.

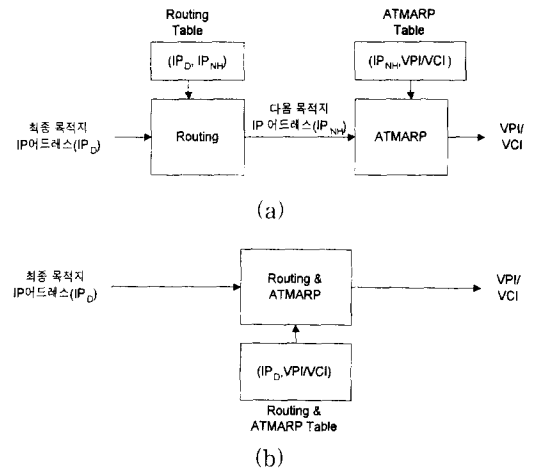


그림 5. Routing and ATMARP Handler 모듈의 동작, (a) 2단계 테이블 검색 방법 (b) 제안된 1단계 검색 방법  
Fig. 5. Operation of Routing and ATMARP Handler module, (a) 2-step searching method (b) proposed single step searching method.

그림 5의 (a)에서는 라우팅 테이블 검색과 ATMARP테이블 검색을 두 번에 걸쳐서 수행하는 과정을 보여주고 있으며, (b)에서는 다음 목적지 어드레스인 IP<sub>NH</sub>을 매개로 하여 최종 목적지 어드레스 IP<sub>D</sub>와 VPI/VCI (Virtual Path Identifier/Virtual Channel Identifier) 값을 직접 연결하는 테이블을 이용하여 한 번의 테이블 검색으로 최종 VPI/VCI값을 얻음으로써 테이블 검색 시간을 반으로 줄일 수 있음을 보여주고 있다.

### 3. 수신부 설계

그림 6에서는 수신부에서의 모듈 구성을 나타내었다. 수신부는 AAL5모듈로부터 수신 데이터그램을 입력받

는 LLC/SNAP Decapsulation Handler에서부터 DMA Output Interface Handler에서 DMA채널을 통해 출력할 때까지 모두 10개의 내부 모듈로 구성되어 있다.

InATMARP 수신 패킷과 다중화되어 DMA채널을 통해 호스트 CPU로 전달된다. 수신부 전체 처리 지연 시간은 AAL5 계층에서 데이터가 입력되어 DMA Output Interface 모듈에서 데이터를 출력할 때까지 걸리는 시간으로, 이 값을 최소로 하도록 설계하였다.

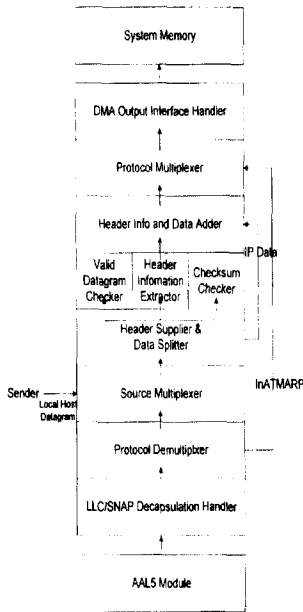


그림 6. 수신부 모듈 구성  
Fig. 6. Block diagram of receiving part.

수신부 구조는 송신부에서 데이터를 처리하는 순서와 정반대로 이루어져 있다. 먼저 AAL5계층에서 수신 데이터를 입력받아 Protocol Demultiplexer 모듈에서 IP 데이터그램과 InATMARP 프로토콜 패킷을 분리한다. Valid Datagram Checker 모듈은 입력 데이터그램이 유효한지를 체크하고, Header Information Extractor 모듈은 데이터그램 헤더중에서 상위 프로토콜로 전해주어야 하는 정보를 추출해내며, Checksum Checker 모듈은 헤더의 체크섬 부분에 에러가 없는지를 확인하는 모듈이다. 이들 세 모듈은 서로 주고받는 데이터가 없이 동시에 처리될 수 있기 때문에 각각 병렬로 동작하여 수신 데이터그램을 처리하고, 처리된 최종 데이터는 Protocol Multiplexer 모듈에서 InATMARP 수신 패킷과 다중화되어 DMA채널을 통해 호스트 CPU로 전달된다. 수신부에서의 전체 처리 지연 시간은 AAL5 계층에서 데이터가 입력되어

DMA Output Interface Handler 모듈에서 데이터를 출력할 때까지 걸리는 시간으로, 이 값을 최소로 하도록 설계하였다.

IV. VHDL 회로 합성 및 시뮬레이션 결과

본 논문에서 설계한 IP 하드웨어는 kg6000 라이브러리와 Synopsys 툴을 이용하여 회로를 합성하고, Verilog 시뮬레이션 툴로 시뮬레이션을 수행하였다. 회로 합성 및 시뮬레이션 과정은 다음과 같다. 먼저 각 모듈별로 회로를 합성한 후 시뮬레이션을 통하여 모듈별로 동작을 확인하였다. 다음으로는 송신부와 수신부 전체를 각각 통합하여 합성한 후, IP프로토콜과 InATMARP프로토콜 데이터에 대하여 시뮬레이션을 수행하였다. 마지막으로 송신부와 수신부를 하나로 합친 전체 IP 프로토콜 하드웨어에 대한 회로를 합성하고, 송수신부가 동시에 동작하는 경우에 대해서 시뮬레이션을 통하여 설계된 하드웨어가 정상적으로 동작함을 확인하였다.

표 2에서는 IP 프로토콜에 대한 시뮬레이션 결과로써 송수신부에서의 프로토콜 처리 시간을 나타내었다. 여기에서 필요지연시간이란 송신부의 경우에는 IP프로토콜에서 추가되어야 하는 헤더에 대한 처리시간을 말하며, 수신부의 경우에는 수신한 헤더 중에서 상위 프로토콜로 보내지 않고 버리는 헤더를 말한다. 송신의 경우에는 순수한 데이터부분이 지연되는 시간에서 헤더가 추가됨으로 인하여 발생하는 지연시간을 뺀 값이 유효 지연시간이라 할 수 있고, 수신에 있어서는 전체 데이터그램의 처리지연시간으로부터 IP내부에서 필요없는 헤더를 제거하는데 걸리는 시간을 뺀 값을 유효 지연시간이라 할 수 있다 (그림 7 참고). 결과적으로 전체 유효지연시간은 이 두 값을 더한 시간이 된다.

표 2. IP 프로토콜 데이터그램의 처리 지연 시간

Table 2. Processing delay time of IP protocol datagram.

구분	송수신	지연 시간		필요지연 시간		유효 지연 시간		전체
		송신	수신	송신	수신	송신	수신	
원격지 데이터그램	전체 데이터	20.5	28	48.5	· 16	· 12	·	31.5
	순수 데이터	40.5	10	50.5	21	· 19.5	·	
로컬 호스트 데이터그램	전체 데이터	11.5	18	29.5	· 8	· 10	·	30.5
	순수 데이터	30.5	8	38.5	10	· 20.5	·	

(단위 : 클럭, 1 클럭 = 51.44ns = 1/155.52MbpsX8)

원격지 데이터그램에 대한 표 2의 데이터를 보면, 송신에서 순수 데이터의 필요지연 시간이 21클럭이고, 전체 데이터의 송신에서의 필요지연 시간이 16클럭임을 보여주고 있다. 이로부터 유효지연시간을 계산하면 송신과 수신에서 각각 19.5클럭과 12클럭이 되고, 따라서 전체 유효 지연시간은 31.5 클럭, 즉 약 1.620  $\mu$ s가 됨을 알 수 있다. 표 3에서는 InATMARP 프로토콜에 대한 처리 지연시간을 나타내었다.

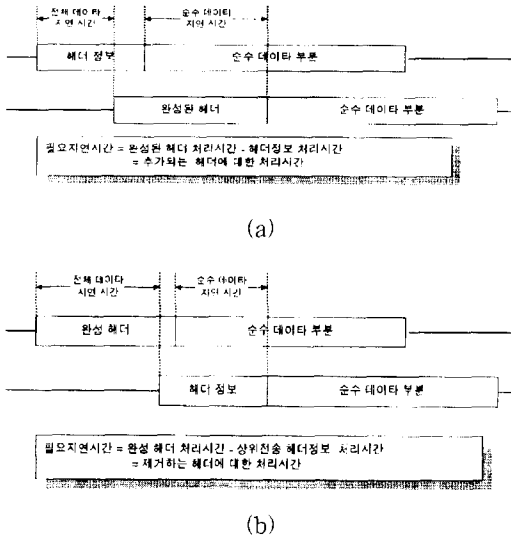


그림 7. 처리 지연 시간의 정의, (a) 송신부에서의 지연시간 (b) 수신부에서의 지연시간.  
Fig. 7. Definition of processing delay time, (a) delay time in sending part (b) delay time in receiving part.

표 3. InATMARP 프로토콜 처리 지연 시간  
Table 3. Processing delay time of InATMARP protocol packet.

구분	송수신	지연 시간			필요지연 시간			유효 지연 시간		
		송신	수신	전체	송신	수신	송신	수신	전체	
원격지 데이터그램	전체 데이터	5.5	14	19.5	·	8	·	6	·	10.5
	순수 데이터	12.5	7	19.5	8	·	4.5	·		

(단위 : 클럭, 1 클럭 = 51.44ns = 1/155.52Mbps\*8)

1. 송신부 시뮬레이션 결과

송신부에서는 전체 IP 데이터그램의 경우 20.5클럭, InATMARP 데이터패킷에 대해서는 5.5클럭의 지연시간으로 송신을 처리할 수 있었다 (표 2,3참조). 송신부 모듈 시뮬레이션의 예로 그림 8에서는 송신부에서 가장 처리시간이 긴 라우팅 및 ATMARP 처리 모듈

에 대한 시뮬레이션 결과를 나타내었다. 그림에서 보면, 이 모듈의 입력 데이터인 목적지의 IP주소 (i\_data [7.0], 십육진수 "12345678")에 대한 처리 결과로 라우팅 테이블 및 InATMARP 주소해석 테이블로 사용하는 LANCAM소자의 출력인 VPI/VCI값이 o\_data [7.0]에서 십육진수 "125712"값으로 출력됨을 볼 수 있다. 이때 입력 데이터 "12345678"이 모두 들어와서 출력 데이터 "125712"가 처음으로 나오는데 걸리는 시간이 9클럭임을 그림에서 알 수 있다.

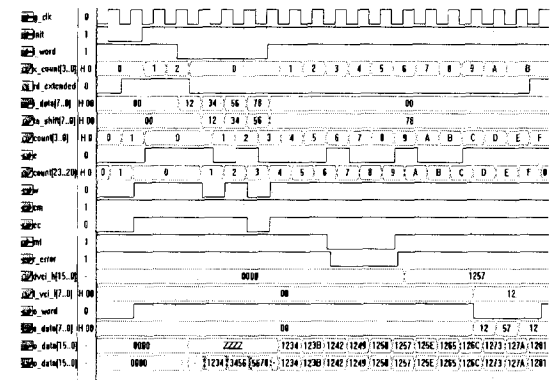


그림 8. 송신부 Routing & ATMARP Handler 모듈의 시뮬레이션 결과  
Fig. 8. Simulation result of Routing & ATMARP Handler module of sending part.

2. 수신부 시뮬레이션 결과

수신부에서는 전체 IP 데이터그램의 경우 28클럭, InATMARP 데이터패킷에 대해서는 14클럭의 지연시간으로 수신 데이터를 처리할 수 있었다 (표 2,3참조).

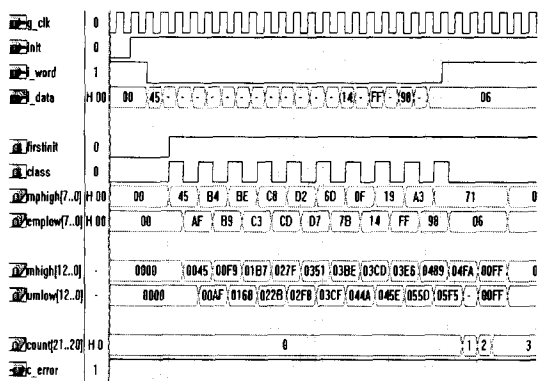


그림 9. 수신부 Checksum Checker 모듈의 시뮬레이션 결과  
Fig. 9. Simulation result of Checksum Checker module of receiving part.

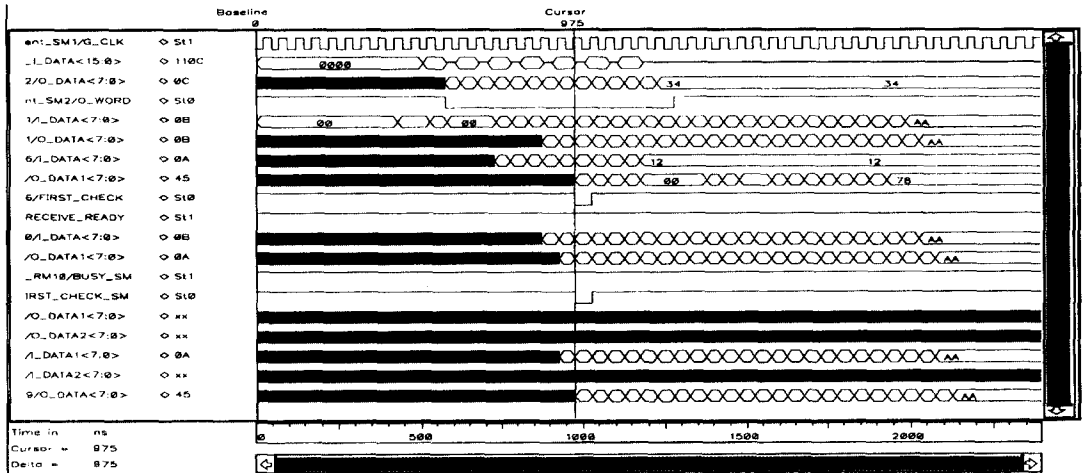


그림 10. 원격지 데이터그램이 로컬호스트 데이터그램보다 먼저 입력될 때의 시뮬레이션 결과.  
 Fig. 10. Simulation result when remote datagram arrives earlier than local host datagram.

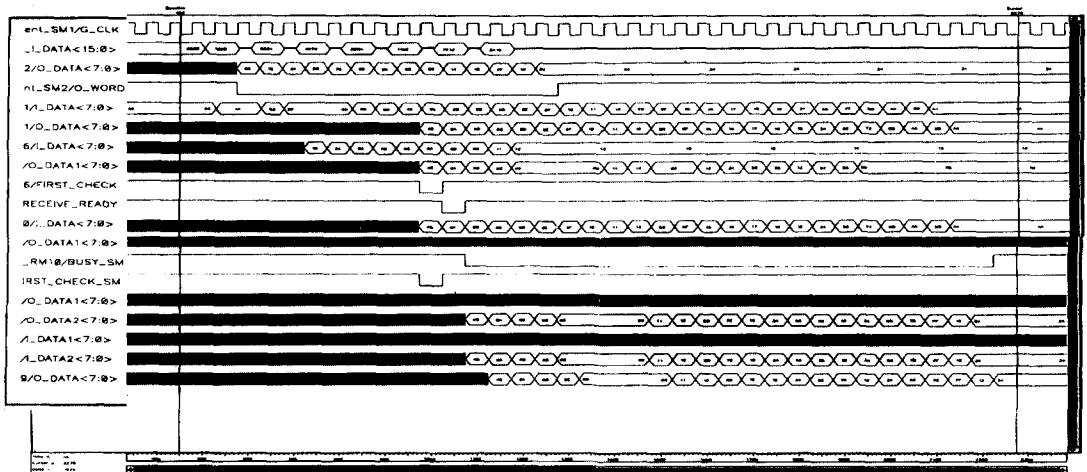


그림 11. 로컬호스트 데이터그램이 원격지 데이터그램보다 먼저 입력될 때의 시뮬레이션 결과  
 Fig. 11. Simulation result when local host datagram arrives earlier than remote datagram.

수신부 모듈 시뮬레이션의 예로 그림 9에서는 수신부에서 가장 복잡한 모듈인 체크섬 확인 모듈에 대한 시뮬레이션 결과를 보여주고 있다. 입력 데이터인 IP 헤더(i\_data [7..0])에 대한 체크섬 확인 결과, 체크섬 결과를 저장하는 내부 신호인 sumhigh [12..0], sumlow [12..0]의 값이 계산결과 십육진수 “FF”값을 가지므로 이 모듈의 최종 출력인 c\_error에서 에러가 출력되지 않음을 확인할 수 있다.

3. 전체 시뮬레이션 결과

설계한 IP 프로토콜 하드웨어는 송신 데이터와 수신 데이터를 동시에 처리할 경우에도 정상적으로 동작해

야 한다. 이러한 예로써, 그림 10과 그림 11에서는 로컬 호스트 데이터그램과 원격지로부터의 데이터그램 수신을 동시에 처리하는 경우의 시뮬레이션 결과를 나타내고 있다.

송신부에서의 로컬 호스트 데이터그램의 입력과 원격지로부터 수신하는 데이터그램의 다중화 기능은 수신부의 Source Multiplexer 모듈에서 처리한다 (그림 6. 수신부 모듈 구성 참조). 그림 10에서 아래 부분의 /A1\_DATA1<7:0>과 /A1\_DATA2<7:0>는 수신부 Source Multiplexer 모듈의 두 입력, 즉 원격지 데이터그램과 로컬 호스트 데이터그램의 입력을 각각 나타



내며, 9/O\_DATA<7:0>는 이 모듈의 출력을 나타낸다. 그림 10의 경우에는 원격지로부터의 데이터그램이 먼저 입력되는데, 로컬 호스트 데이터그램의 송신부에서는 IRST\_CHECK\_SM에서 이를 확인하고 데이터그램을 출력하지 않는다 (/L\_DATA2<7:0>참조). 그러나, 원격지 데이터그램 /L\_DATA1<7:0>은 그대로 입력되어 출력단 9/O\_DATA<7:0>에서 데이터가 출력되고 있음을 알 수 있다.

그림 11은 그림 10과는 반대의 경우로, 로컬 호스트 데이터그램이 원격지 데이터그램보다 먼저 입력되는 경우이다. 그림의 아래 부분에서 알 수 있듯이 원격지 데이터그램을 수신하는 모듈에서는 \_RM10/BUSY\_SM의 값을 읽어서 이 값이 액티브 상태를 알 수 있으므로 수신 데이터를 Source Multiplexer모듈로 출력하지 않는다. 따라서 Source Multiplexer 모듈에서 원격지 데이터그램의 입력 /L\_DATA1<7:0>에서는 입력되는 데이터가 없고, 로컬 호스트 데이터그램의 입력단 /L\_DATA2<7:0>에서는 데이터그램이 입력된다. 그림에서 Source Multiplexer 모듈의 최종 출력은 로컬 호스트 데이터그램임을 알 수 있다.

## V. 결 론

본 논문에서는 ATM망 위에서 기존의 TCP/IP 프로토콜을 수용할 수 있는 방안으로써 IETF에서 제시한 IP over ATM 규격을 따르는 IP 프로토콜을 IEEE 표준인 VHDL을 사용하여 하드웨어로 설계하였다. 송수신 부분을 완전히 분리함으로써 전이중 방식으로 동작할 수 있도록 하였으며, 송수신부에서 처리 시간이 많이 걸리지만 모듈간에 서로 주고 받는 데이터가 없어 동시에 동작할 수 있는 부분에 대해서는 병렬 처리 구조를 채택하여 처리 속도를 향상시킬 수 있었다. 또한 라우팅 테이블 검색과 ATMARP테이블 검색을 결합한 구조를 제안하여 테이블 검색시간을 줄일 수 있도록 설계하였다. 설계된 회로는 kg6000 ASIC라이브러리와 회로합성 툴인 Synopsys를 사용하여 합성하고 시뮬레이션을 수행하였다. IP 데이터그램에 대한 시뮬레이션 결과, 전체 데이터그램에 대해서는 송신부와 수신부에서 각각 48.5클럭과 29.5클럭의 지연시간을 나타내었다. 이 결과로부터 송신부에서 31.5클럭, 수신부에서 30.5클럭의 유효처리지연시간으로 155Mbps의 송수신 데이터를 처리할 수 있음을 확인하였다. 본 연구

논문의 결과는 IP over ATM 라우터나 TCP 및 UDP프로토콜을 하드웨어로 설계하고 구현하는데 있어서 실질적인 도움을 줄 수 있을 것으로 기대한다.

## 참 고 문 헌

- [1] J. Postel, "Internet Protocol", RFC791, Network Working Group, Sep. 1981.
- [2] J. Postel, "Internet Control Message Protocol", RFC792, Network Working Group, Sep. 1981.
- [3] T. Socolofsky and C. Kale, "A TCP/IP Tutorial", RFC1180, Network Working Group, Jan. 1991.
- [4] R. Braden, "Requirements for Internet Hosts - Communication Layers", RFC1122, Network Working Group, Internet Engineering Task Force, Oct. 1989.
- [5] P. Newman, "ATM Local Area Networks", *IEEE Communications*, vol. 32, no. 3, pp. 86-98, Mar. 1994.
- [6] H. Chao, D. Ghosal, D. Saha, and S. Tripathi, "IP on ATM Local Area Networks", *IEEE Communications*, vol. 32, no.8, pp. 52-59, Aug. 1994.
- [7] M. Laubach, "Classical IP and ARP over ATM", RF1577, Information Sciences Institute, Jan. 1994.
- [8] M. Laubach, "Classical IP and ARP over ATM Update", Internet-Draft, Network Working Group, Aug. 1995.
- [9] J. Bhasker, *A VHDL Primer*, Prentice Hall, 1995.
- [10] R. Airiau, J. Berge, and V. Olive, *Circuit Synthesis with VHDL*, Kluwer Academic Publishers, 1994.
- [11] J. Heinanen, "Multiprotocol Encapsulation over ATM Adaptation Layer 5", RFC1483, Network Working Group, Jul. 1993.
- [12] R. Handel, *ATM Networks Concepts, Protocols, Applications*, Addison Wesley, Jun. 1991.
- [13] *The MU9C1480 LANCAM Handbook*, Music Semiconductor.
- [14] R. Braden, D. Borman, and C. Partridge,

“Computing the Internet Checksum”,  
RFC1071, Network Working Group, Sep.  
1988.

[15] D. E. Comer and D. L. Stevens,  
*INTERNETWORKING WITH TCP/IP*,

Vol I, II, Prentice Hall International  
Editions, 1994.

[16] *MC68340 INTEGRATED PROCESSOR  
WITH DMA USER'S MANUAL*,  
Motorola.

저 자 소 개



崔 秉 台(正會員)

1994년 2월 : 고려대학교 전자공학과(공학사). 1996년 2월 : 고려대학교 전자공학과(공학석사). 1996년 3월 ~ : 고려대학교 전자공학과 박사과정 재학중. 1996년 11월 : IEEE APCCAS Best Paper Award 수상.

관심분야 : 영상 신호처리, 멀티미디어 통신



高 聖 濟(正會員)

1980년 2월 : 고려대학교 전자공학과(공학사). 1986년 5월 : 미국 State Univ. of New York at Buffalo, 전기 및 컴퓨터 공학(공학석사). 1988년 8월 : 미국 State Univ. of New York at Buffalo,

전기 및 컴퓨터 공학(공학박사). 1981년 8월 ~ 1983년 12월 : 대한전선 중앙연구소 연구원. 1998년 8월 ~ 1992년 5월 : 미국 The University of Michigan-Dearbon, 전기 및 컴퓨터 공학과 조교수. 1992년 3월 ~ 현재 : 고려대학교 전자공학과 부교수. 1996년 11월 : IEEE APCCAS Best Paper Award 수상. 관심분야 : 신호 처리 및 영상 처리, 영상 압축 및 통신, 멀티미디어 통신 등

崔 竣 均(正會員)

1959년 10월 22일생. 1982년 2월 서울대학교 전자공학과(학사). 1985년 8월 한국 과학기술원, 전기 및 전자공학과 (공학석사). 1988년 2월 한국 과학기술원, 전기 및 전자공학과 (공학박사). 1990년 8월 ~ 1991년 8월 캐나다 토론토 대학 교환 연구원. 1986년 6월 ~ 현재 한국 전자통신연구소 선임연구원. 통신망구조연구실 실장.



金 在 根(正會員)

1980년 2월 고려대학교 전자공학과 졸업. 1983년 2월 고려대학교 대학원 전자공학과 졸업(석사). 1990년 8월 고려대학교 대학원 전자공학과 졸업(박사). 1979년 12월 현재 한국 전자통신연구소 근무. 광대역통신망

연구부, 책임연구원