제19회 한국정보처리학회 춘계학술발표대회 논문집 제10권 제1호 (2003.5)

IPv4와 IPv6의 변환을 위한 SIIT 하드웨어 설계

박상원, 송문빈, 이두영, 임재청, 정연모
경희대학교 전자공학과
e-mail : chung@khu.ac.kr

Hardware design of SIIT for IPv4 to IPv6 Protocol Translation

Sangwon Park, Moonvin Song, Dooyoung Yi, Jaechung Lim, Yunmo Chung
Dept. of Electronic Engineering, Kyung-Hee University

요 약

현재는 IPv4의 주소 체계를 사용하여 많은 단말기들이 인터넷에 연결되고 있다. 32비트 주소 체계인 IPv4는 앞으로 유비쿼터스 환경에서 모든 단말기에 주소를 할당 할 수 없는 문제점을 가지고 있다. 이러 문제점을 해결하기 위해 새로운 주소 체계인 IPv6가 연구되고 있다. 현재의 주소 체계를 사용하고 있는 대부분의 단말기들과 IPv6의 주소 체계를 사용하는 단말기들을 직접 연결하는 것은 불가능하다. IPv4에서 IPv6의 주소 체계로 넘어가는 과도기적 단계에서 두 프로토콜 간의 상호 변환이 필요하다. 본 논문에서는 *IPv4와 IPv6의 변환 기술인 SIIT(Stateless IP/ICMP Translator)*를 하드웨어로 설계하기 위하여 VHDL로 모델링 하였으며 FPGA에서 검증하였다.

1. 서론

현재 인터넷 환경을 통한 시스템 간의 통신을 제공하는 IPv4 주소 체계는 몇 가지 문제점을 가지고 있다. 첫째, IPv4의 주소 개수가 한정되어 있어 많은 단말기가 생길 경우 인터넷 연결에 제한이 있다. 둘째, 고품질의 실시간 오디오 및 비디오 전송이 어렵다. 셋째, 암호화 및 인증 등을 수용하지 못한다. 마지막으로 클래스 개념을 사용하여 주소를 분할하므로 주소 공간을 비효율적으로 사용한다.

IPv4 에서는 위와 같은 문제를 해결하기 위해 서버에서 주소를 동적으로 할당하는 DHCP(Dynamic Host Configuration Protocol) 서버를 운영하거나, NAT(Network Address Translation) 서브넷팅, 수퍼네팅 등의 방법을 사용한다[1]. 그러나 이러한 방법은 위에서 제시한 문제점들을 해결하지 못하므로 해결 방법으로 IPv6 연구가 진행되고 있다. 현재 IPv4 사용되고 있으므로 IPv6로 대체 되는 과정에서 두 버전이 같이 사용 된다. 두 프로토콜을 동시에 사용하기 위해서는 주소 체계를 변환 시켜주는 시스템이 필요하다[6]. 본 논문에서는 두 프로토콜간 변환 방법인 SIIT을 하드웨어로 구현하였다.

2. IPv4

IPv4는 RFC 791 및 MIL-STD 1777에 정의되어 있다. IPv4의 주요 역할은 호스트의 주소 분석과 데이터 전송을 위한 패킷의 단편화 처리이다. 그러나 종단간에 전송되는 메시지의 안정성이나 흐름 제어 등의 역할은 수행하지 않는다. 이러한 역할은 상위 프로토콜에서 수행된다[5].

IPv4의 특징을 정리하면 다음과 같다. 첫째, 호스트의 인터넷 주소는 32비트를 사용한다. 둘째, 호스트간의 통신은 비연결성 프로토콜이다. 셋째, 여러 프레임으로 단편화하여 전송을 하거나 단편화 된 프레임을 하나의 데이터 그램으로 만들어 상위 레이어로 전송한다. 이때 데이터 그램의 최대 크기는 헤더를 포함한 65,535 바이트이다. 넷째, IP에서는 헤더의 체크썸(checksum) 검사만을 수행한다. 다섯째, Time To Live 필드를 두어 패킷의 가용 시간을 기

록 한다[3]. 인터넷 프로토콜이 [표 1]과 같이 네 가지 프로토콜을 포함하며 이들 상호 간에 작용을 하면서 하위 레이어와 상위 레이어의 교량 역할을 하게 된다.

표 1. IP 구성 프로토콜

프로토콜	기능
ARP	IP주소를 물리 주소로 변환
RARP	물리 주소를 IP 주소로 변환
ICMP	패킷의 문제점을 송신자에게 알림
IGMP	수신자들이 메시지를 동시에 전송

[그림 1]은 IPv4의 데이터 그램의 구조를 나타내며 패킷의 앞은 20~60 바이트의 헤더 부분이다. 최대 65,535 바이트인 패킷의 길이에서 헤더를 뺀 나머지 부분을 모두 데이터로 채운다. 또한 데이터 부분은 최소 46 바이트 이상이다. 만약 데이터가 46 바이트보다 작을 경우에는 나머지 부분에 '0' 으로 채워 넣는다[4].

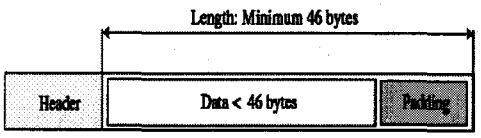

그림 1. IPv4 데이터 그램 구조

[그림 2]는 IPv4의 패킷 구조를 나타낸 그림이며 VER을 통해 IP 버전을 확인한다. 다음에 오는 HLEN은 헤더의 길이를 나타내며 Service Type은 라우터에서 사용하는 정보를 갖는다. Total Length를 통해 패킷의 총 길이를 알 수 있다. Identification, Flags, Fragmentation Option들은 패킷의 단편화를 위해서 사용하는 부분이다. Time To Live 부분을 통해 패킷의 수명시간을 결정하고 Protocol을 통해서 목적지 상위 프로토콜을 알 수 있다. Header Checksum을 사용하여 헤더의 오류를 알 수 있다. Source IP Address을 통해 송신자 IP 주소를 알고 Destination IP Address을 통해 수신자 IP 주소를 나타낸다. 마지막으로 Option을 통하여 네트워크를 시험하거나 디버그 할 수 있다.

3. IPv6

IPv6는 IPv4로부터 많은 개념을 도입하여 세부 사항들을 정의하였다. IPv4와 마찬가지로 비연결적이며 각 데이터 그램은 도착지 주소를 포함하고, 독립적으로 라우팅 된다. 또한 다양한 길이의 Option 필드를 갖는 단일 헤더 대신에 선택적 정보를 처리하기 위하여 고정된 길이의 여러 헤더들을 사용한다.

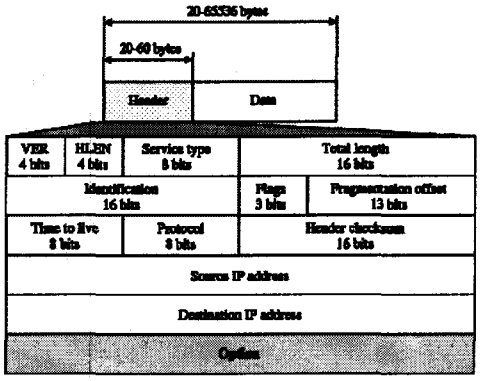

그림 2. IPv4 패킷 구조

IPv6가 IPv4와 비교하여 개선된 부분은 다음과 같다. 첫째, 128비트로 확장된 주소 체계를 가지고 있으므로 많은 수의 단말기를 연결할 수 있다. 둘째, 헤더 형식의 변화이다. 가변길이의 단일 헤더가 아닌 고정된 이중 헤더 구조를 가진다. 셋째, 고품질의 오디오, 비디오 데이터를 전송하기 위한 영역을 따로 만들어 두었다. 넷째, 보안성을 위해 MD5 암호화을 사용할 수 있다. 다섯째, IPv4에서 사용한 ARP 및 IGMP 프로토콜은 IPv6에서는 ICMPv6에 흡수 통합 하였으며 RARP 프로토콜은 삭제되었다. 여섯째, BOOTP 프로토콜이 RARP의 기능을 수행한다[8].

[그림 3]은 IPv6 데이터 그램의 구조를 나타낸 것이며 기본 헤더와 1개의 Payload로 구성 된다. Payload는 확장 헤더 영역과 상위 계층으로부터 받은 데이터 영역으로 구성된다.

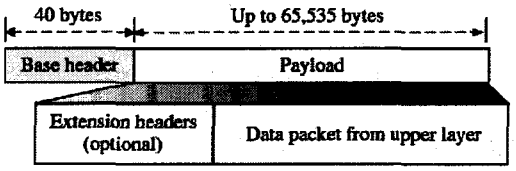

그림 3. IPv6 데이터 그램 구조

IPv6의 패킷 형식은 [그림 4]와 같으며 기본 헤더는 8개의 기본 필드로 이루어져 있다. VER은 4비트 IP 버전을 나타내며, PRI 필드는 동시 접속에 대한 우선순위를 정의하고, Flow label에서는 특정 흐름들을 다룬다. Payload Length는 기본 헤더를 제외한 IP 데이터 그램의 전체 길이를 정의하고 Next header는 확장 헤더 이거나 상위 계층 프로토콜을 위한 헤더이다. Hop limit 필드는 패킷의 수명 시간을 정의 하는데 사용된다.

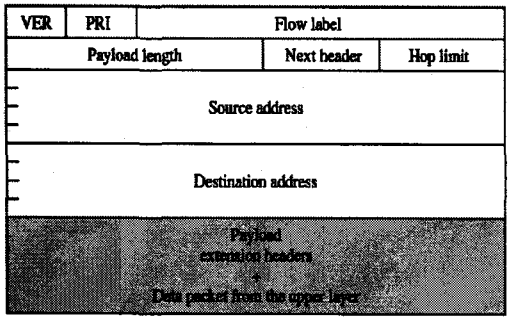

그림 4. IPv6 패킷 구조

4. 설계 및 검증

IPv4와 IPv6간 변환 기술은 계층에 따라 세 가지로 분류할 수 있다. 첫째, 헤더 변화 방식이다. IP 계층에서의 변환 방법은 IPv4 패킷 헤더를 IPv6 패킷 헤더로 변환하거나 그 반대로 IPv6를 IPv4로 변환한다. 둘째, 수송계층 릴레이 방식이다. 각각 IPv4와 IPv6 각각의 TCP와 UDP 세션을 중간에서 릴레이 하는 방법이다. 셋째, 응용 계층 게이트웨이(ALG) 방식이다. 트랙잭션 서비스를 위한 ALG는 사이트 정보를 숨기고 캐시 메커니즘으로 서비스의 성능을 향상시키기 위한 방법이다.

위에서 설명한 방법 중 첫 번째 방법인 SIIT를 본 논문에서는 하드웨어로 설계하였다. 이 방법은 변환속도가 빠르며 다음과 같은 두 단계의 과정을 거친다. 첫째, IPv4 와 IPv6의 헤더를 변환하는 부분이다. 둘째, IPv4의 ARP, RARP, ICMP등의 프로토콜을 ICMPv6로 바꾸거나 또는 그 반대로 바꾸는 단계를 거친다.[2].

[그림 5]는 네트워크에서 IPv4와 IPv6의 변환기의 위치를 나타낸 것으로 라우터들은 각 네트워크에 연결된 호스트의 IP 버전과 주소를 알고 있다. 다른 네트워크로 전송되는 같은 버전 패킷 이면 다른 라우터로 패킷을 전송하고 패킷이 다른 버전의 IP인 경우는 패킷 변화기로 그 데이터를 보낸다. 이때 변환기는 서로 다른 IP 버전의 주소와 스택을 지원 할 수 있어야 한다.

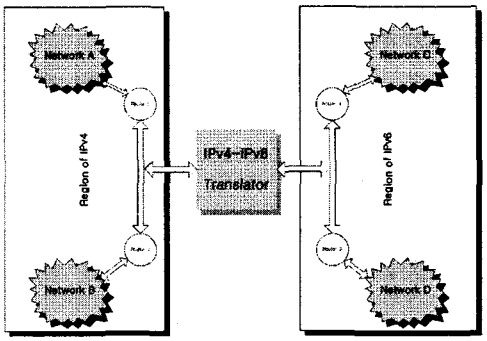

그림 5. IPv4-IPv6 네트워크에서 변환기 위치

본 논문에서 설계한 변환기는 크게 두 단계의 과정을 거친다. [그림6]은 변환기의 첫 번째 단계를 나타낸 것이다. 수신된 패킷의 버전을 확인하며 그 패킷의 목적지를 확인한다. 먼저 다른 버전의 네트워크로 가는 패킷이면 헤더를 변환하기 위해 다음 모듈로 전송을 하고 그렇지 않는 경우는 내부 네트워크의 다른 모듈로 전송을 한다.

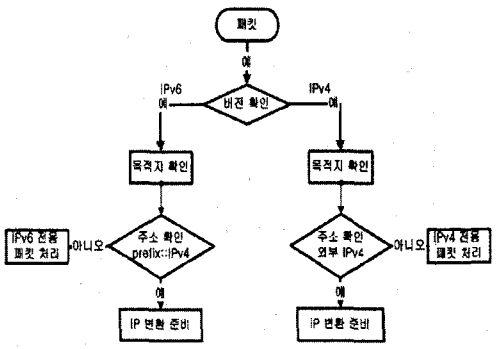

그림 6. IPv4-IPv6 변환 1 단계

[그림 7]은 변환기의 두 번째 단계의 흐름을 나타낸 것으로 프로토콜을 확인하여 값에 따라 다음 프로토콜 모듈로 패킷을 전송한다.

TCP와 UDP는 네트워크 레이어에서 처리를 하지 않고 보다 상위 레이어 처리 모듈로 보낸다. ICMP 모듈은 받은 메시지의 변환 여부를 분석하여

변환이 가능 하면 메시지에 변환된 헤더를 추가하여 전송 모듈로 보내고 변환이 불가능 한 경우는 에러 메시지를 패킷의 발신자에게 전송한다.

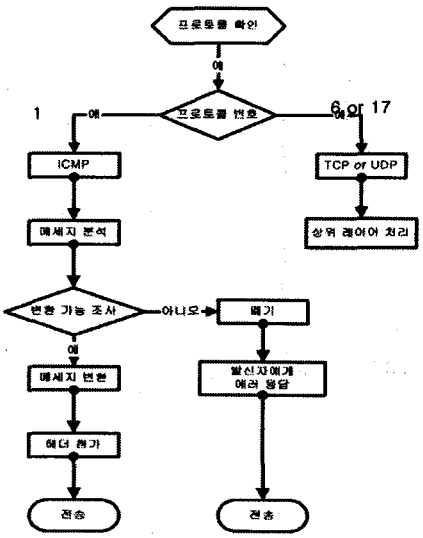

그림 7. IPv4-IPv6 변환기 2 단계

[그림 8]은 설계된 IPv4-IPv6 변환기의 전체 블록 다이어그램으로 두개의 전처리 블록은 변환기의 물리적인 레이어의 전송 특성에 따라 구성이 결정된다.

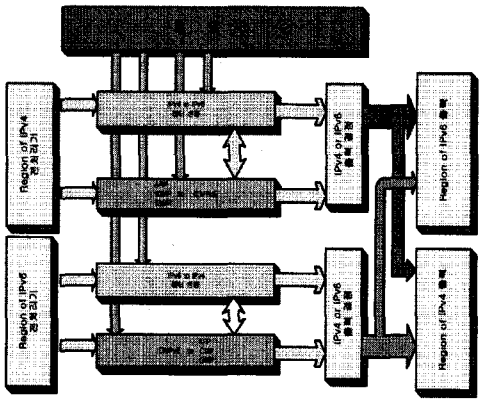

그림 8. IPv4-IPv6 변환 블록 다이어그램

전처리 단계를 지나온 IPv4 패킷은 IPv4 to IPv6 헤더 변환블록을 통해서 헤더를 변환하고 ICMPv4 to ICMPv6 블록을 거치면서 여러 프로토콜로 변환한다. 만약 변환을 하지 못하는 경우에는 [그림 7]에서 제시하는 흐름에 따라 발신자에게 에러 메시지를 IPv4 출력 블록을 통하여 전송을 하게 된다. 마찬가지로 IPv6 패킷은 IPv6 to IPv4 블록에서 헤더 변환을 수행한다.

5. 결론

[그림 9]는 전체 블록을 합성한 그림이다.

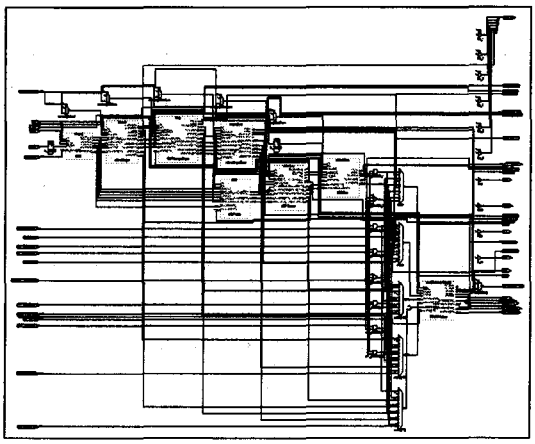

그림 9. IPv4-IPv6 변환 모듈

본 논문에서는 IPv4에서 IPv6로 변화하는 단계에서 필요한 IPv4-IPv6 변환기 SIIT를 하드웨어로 설계를 위한 구조를 제시하고 VHDL로 모델링 하였다. 또한 FPGA를 이용하여 실제로 그 기능을 검증하였다.

참고 문헌

[1] The Linley group. *A guide to Network Processors*. The Linley group, 2002.
[2] Jag Bolaria *A guide to Storge and TCP Processors*. The Linley group, 2002.
[3] Net Technology Laboratory. *IPv6*. 미래컴, 2000.
[4] Jean Walrand. 통신 네트워크. 교보문고, 2000.
[5] Behrouz A. Forouzan. *TCP/IP*. 미래컴, 2000.
[6] 조경산. 컴퓨터 네트워크와 인터넷. 그린, 1995.
[7] 유상렬. TCP/IP 인터넷. 성안당, 1995.
[8] IPv6 포럼 코리아, 차세대 인터넷 프로토콜 IPv6. IPv6 포럼 코리아, 2002.