

IPv6-IPv4 프로토콜 변환기를 위한 ALG 모듈의 H/W 설계 및 구현

이중렬^o 공인엽 이경렬 이정태
부산대학교 네트워크 연구실
{blueirix^o, leafgirl, idking96, jtee}@pusan.ac.kr

H/W Design and Implementation of ALG Module for IPv6-IPv4 Translator

JoongLyul Lee^o InYeup Kong KyoungLyul Lee JungTae Lee
Dept. of Computer Engineering, Pusan National University

요 약

최근 IPv6의 도입에 있어서 중요한 과제인 IPv6-IPv4 프로토콜 변환기에 대한 연구 및 구현이 활발히 이루어지고 있다. 그러나 기존에 구현된 IPv6-IPv4 프로토콜 변환기는 운영 체제에 종속적인 소프트웨어로서, 소프트웨어 TCP/IP 자체가 가지는 오버헤드와 비효율적인 메모리 접근 방식으로 인하여 성능이 제한되는 문제점이 있다. 이러한 문제를 해결하기 위하여 성능 분석을 통해 고성능 IPv6-IPv4 프로토콜 변환기의 구현 방법에 대한 연구가 진행되었고, 이에 대한 결과로서 하드웨어 기반의 IPv6-IPv4 프로토콜 변환기인 64Translator가 제안되었다. 이러한 64Translator는 다양한 응용을 지원하기 위해 DNS와 FTP에 대한 응용 프로토콜 변환 모듈을 필요로 하는데, 이는 하나의 모듈로 통합되기 위해 하드웨어로 구현되어야 한다. 이에 본 논문에서는 DNS와 FTP에 대한 응용 프로토콜 변환 모듈을 하드웨어로 설계 및 구현하였고, 이를 64Translator에 통합하여 시뮬레이션과 시험망에서의 테스트를 수행함으로써 기능을 검증하였다.

1. 서 론

인터넷의 급격한 보급과 무선인터넷, 정보가전 등의 신규IP 주소 수요급증과 비효율적인 할당으로 인하여 IPv4 주소는 고갈 위기를 맞고 있다. 이러한 문제를 해결하기 위하여 IETF에서는 IPv6를 표준화하였다. IPv6는 128비트의 풍부한 주소 공간을 기본으로 하여 IP 주소의 부족문제를 근본적으로 해결할 뿐만 아니라 자동 주소 설정, 품질 보장, 보안 등의 다양한 기능을 제공하는 차세대인터넷 프로토콜이다. 그러나, IPv6를 도입하기 위해서는 호스트와 라우터의 IP 프로토콜 스택을 교체해야 하므로 상당기간 동안 IPv4와 IPv6가 상호 공존하는 형태가 될 것으로 예상되고 있다. 이러한 환경에서 IPv4와 IPv6 Network 간의 통신을 가능하게 하는 IPv6-IPv4 프로토콜 변환기에 관한 연구가 활발히 이루어지고 있다[1][2].

현재까지의 IPv6-IPv4 프로토콜 변환기는 소프트웨어로 구현되고 있는데, 이는 운영 체제에 종속적으로 동작한다. 이러한 프로토콜 변환기는 운영 체제 내의 TCP/IP 프로토콜이 가지는 근본적인 오버헤드와 비효율적인 메모리 액세스로 인하여 네트워크 간의 트래픽을 처리하기에는 성능이 떨어지는 문제점이 있다[3].

따라서, 운영 체제에 독립적으로 프로토콜 변환 기능을 제공할 수 있는 형태로 구현된 IPv6-IPv4 프로토콜 변환기가 필요하다. 이를 위하여 고성능 IPv6-IPv4 프로토콜 변환기의 구현에 관한 연구[3]를 바탕으로 하여 IP 계층에서 프로토콜 변환을 제공하는 NAPT-SIIT(Network Address Port Translation / Stateless IP/ICMP Translation) 기술을 사용한 하드웨어 IPv6-IPv4 프로토콜 변환기(64Translator)를 개발하였다[4].

64Translator의 구현에 있어서 프로토콜 변환 기능의 구현과 더불어 고려되어야 할 점은 응용 계층에서의 변환 문제이다. DNS나 FTP와

같은 응용의 경우, IP 버전에 따라 응용 프로토콜 규격이 다르고, 응용 프로토콜 헤더에 IP 주소와 포트 번호를 포함하는 특징을 가지고 있다. 이러한 응용에 대해서는 IPv4와 IPv6 간에 응용 프로토콜을 변환해주는 ALG(Application Level Gateway)가 필수적이다.

그러나, 현재 구현된 ALG는 소프트웨어로 구현되어 있어서 하드웨어로 구현된 64Translator와 통합될 수 없는 문제점이 있다. 이에 본 논문에서는 DNS_ALG와 FTP_ALG 기능을 하드웨어로 설계 및 구현하였다. 구현된 모듈은 64Translator와 통합하여 소프트웨어 시뮬레이션과 시험망 테스트를 수행함으로써 기능을 검증하였다.

본 논문의 구성은 다음과 같다. 2절에서는 ALG 기술동향을 알아보고, 3절에서 IPv6-IPv4 프로토콜 변환기의 전체 구성도와 ALGTrans의 상세설계 및 구현에 대해 기술하고, 4절에서는 동작환경과 테스트에 관한 내용을 설명하고, 5절에는 결론 및 향후 계획에 관한 내용을 적었다.

2. ALG 기술 동향

ALG(응용 Level Gateway)는 IPv6와 IPv4 호스트 간의 통신을 가능하게 하는 응용 에이전트로서, NAPT-PT와 결합되어 Payload에 네트워크 주소들을 나눌 수 있도록 제공한다. 다음과 같은 특징을 가지는 응용의 경우, ALG가 필수적이다.

- ① IPv6 응용 표준이 해당 응용의 IPv4 표준을 수정 또는 확장한 경우, (예: DNSv4에서의 Record Type은 A, DNSv6에서의 Record Type은 AAAA임.)
- ② 응용 프로토콜 헤더 내에 IP 주소 또는 포트 번호를 포함하는 경우. (예: DNS Answer 메시지에 DNS Query 시에 요청되었던 도메인 이름에 대한 IP 주소가 포함됨.)

ALG가 필요한 대표적인 응용으로는 DNS, FTP, SIP(Session Initiation

Protocol) 등이 있는데, DNS와 FTP는 호스트에 기본적으로 포함되어 있는 응용으로서, 이미 이에 해당하는 ALG가 RFC로 확정되어 있다 [5][6]. 따라서 본 논문에서는 DNS_ALG, FTP_ALG에 대해 연구하였고, 이를 하드웨어로 구현하였다.

2.1 DNS_ALG

IPv6용 DNS 표준(DNSv6)은 기존 IPv4에서의 DNS 표준의 틀을 그대로 사용하면서, IPv6 주소를 표현하는 Record Type인 AAAA Type과 A6 Type을 추가로 확장하였다[5].

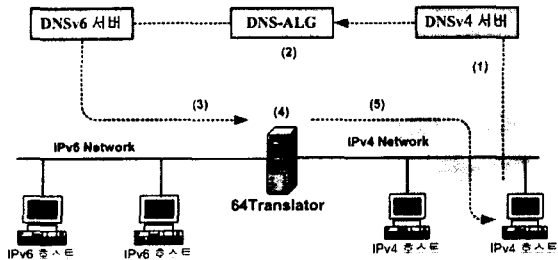


그림 1. DNS_ALG의 동작 원리

IPv4 호스트가 IPv6 호스트의 IP 주소를 얻기 위한 DNS 과정을 통하여 DNS_ALG의 동작 원리를 설명하면 다음과 같다. 먼저 IPv4 호스트는 접속하고자 하는 IPv6 호스트의 IP 주소를 알아내기 위해 DNS Query 메시지를 전송하고, 이는 DNSv4 서버를 통해 64Translator의 DNS-ALG에 전달된다(1). DNS-ALG는 이 DNS Query에 대하여 표 1에 제시한 바와 같이 변환을 수행한다(2).

표 1. DNS 메시지의 Record Type 변환

	변환 전	변환 후
Name to address query request	A	AAAA 또는 A6
Address to name query request	IN-ADDR.ARPA	IP6.INT

DNSv6에 맞게 변환된 DNS Query 메시지는 DNSv6 서버에 의해 처리되고, DNSv6 서버는 IPv6 호스트의 IPv6 주소를 포함하는 DNS Response 메시지를 생성하여 64Translator로 전달한다(3). 이에 64Translator의 DNS-ALG에서는 표 1에서 제시된 규칙을 역으로 적용하여 DNS Response 메시지를 변환하는데, 이와 더불어 DNS Response 메시지에 포함된 IPv6 주소를 IPv4 주소로 변환해야 한다(4). DNSv4에 맞게 변환된 DNS Response 메시지는 해당 DNSv4 서버를 통해 IPv4 호스트에 전달되고, 이로써 IPv4 호스트는 기존의 DNSv4 메커니즘을 사용하여 IPv6 호스트의 IP 주소를 획득하게 된다.

2.2 FTP_ALG

FTP에서는 FTP 제어세션이 데이터 세션에 대해 TCP포트 정보와 IP주소를 전달하기 때문에 이 부분을 변환해주는 FTP_ALG가 필요하다. 기존의 IPv4 노드에서 운영되는 FTP 응용에서는 Port Command 변수와 PASV Response 변수가 사용되는데, 이는 IPv6에서 EPRT Command 와 EPSV Response로 확장되었다[6]. FTP_ALG는 이러한 Command 간의 변환을 담당하고, FTP 헤더에 포함되는 IP 주소와 포트 번호에 대한 변환 또한 담당한다.

FTP 세션을 생성한 IPv4 호스트가 PORT Command 또는 PASV Response를 사용한다면 FTP_ALG는 이를 IPv6 호스트에 전달하기 위해 EPRT Command 와 EPSV Response 형식으로 변환을 해야 한다.

EPRT Command와 PORT Command 형식은 아래와 같다.

표 2. FTP Command 변환

EPRT	EPRT<space><d><net-prt><d><net-addr><d><tcp-port><d>
PORT	PORT h1,h2,h3,h4,p1,p2

IPv4의 PORT Command를 IPv6의 EPRT Command로 변환을 위해서는 EPRT의 <net-prt> 필드를 AF #2로 설정하고, IPv4 호스트의 IP 주소 (h1,h2,h3,h4)를 IPv6-Pv4 프로토콜 변환기가 할당한 IPv6 주소의 String notation으로 변환함으로써 Command를 변환한다. 이러한 형식으로 IPv6의 EPSV Response는 IPv4의 PASV Response로 변환된다.

3. ALG 모듈의 H/W 설계 및 구현

3.1 전체 모듈 구성도

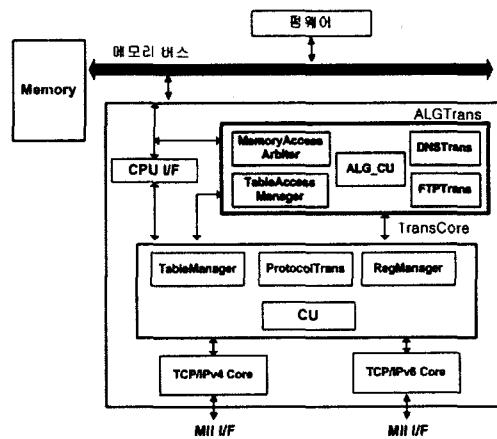


그림 2. 64Translator의 전체 구성도

64Translator는 TCP/IPv4 Core, TCP/IPv6 Core, TransCore, ALGTrans 모듈로 구성된다. TCP/IPv4와 TCP/IPv6 Core 모듈은 각각 해당 프로토콜을 하드웨어로 구현한 모듈이고, MII I/F는 Ethernet과 연결하기 위한 표준 인터페이스로서, 연구실에서 기구현된 모듈을 사용하였다. TCP/IPv4 Core, TCP/IPv6 Core는 64Translator가 운영 체제 없이 단독으로 동작할 수 있도록 제공한다. CPU I/F는 펌웨어와의 인터페이스를 제공하는 모듈로서, TransCore 모듈에 포함되어 구현되었다.

TransCore 모듈은 NAT-PT/SIIT 기능을 구현한 모듈로서, 내부 레지스터를 관리하는 RegisterManager 프로토콜을 변환하는 ProtocolTrans, Mapping Table을 관리하는 TableManager, TransCore의 세부 모듈을 제어하는 CU로 구성된다[2]. ALGTrans 모듈은 본 논문에 의한 모듈로서 DNS_ALG와 FTP_ALG 기능을 하드웨어로 구현한 모듈이다.

3.2 ALG 모듈 설계 및 구현

ALGTrans 모듈은 ALG_CU, TableAccessManager, MemoryAccessArbiter, DNS_Trans, FTP_Trans 모듈로 구성된다.

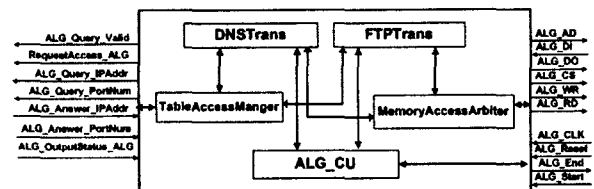


그림 3. ALGTrans 모듈의 세부 모듈 구성

그림 3에서 보는 바와 같이 ALGTrans 모듈은 ALG_CU, TableAccessManager, MemoryAccessArbiter, DNSTrans, FTPTrans 모듈로 구성되며, 각 세부 모듈을 설명하면 다음과 같다.

ALG_CU 모듈은 ALGTrans 모듈을 전체적으로 제어하는 모듈로서, ALGTrans 모듈의 초기화를 담당하고, TransCore 모듈의 요청에 따라 해당하는 ALG 모듈을 동작시킨다.

TableAccessManager 모듈은 TransCore 모듈의 TableManager 모듈로부터 DNS 메시지의 변환에 필요한 IP 주소, FTP Command의 변환에 필요한 IP주소와 포트 번호에 해당하는 변환 정보에 대하여 검색, 추가 및 삭제를 요청하고, 이에 대한 응답을 받아 처리하는 모듈이다.

MemoryAccessArbiter 모듈은 외부 메모리에 저장된 Payload정보를 읽어와서 내부 레지스터에 저장하고, ALG 변환에 필요한 정보를 제공한다. 그리고 변환을 수행한 후 변경된 Payload의 크기를 TransCore 모듈로 알려준다.

DNSTrans 모듈은 DNS 메시지의 Query, Response 메시지를 분석하여 IP주소와 Record Type을 변환하는 모듈로서, [5]의 표준에 따라 구현하였다.

FTPTrans 모듈은 FTP Command를 변환하고, 포트 번호와 IP 주소의 변환을 담당하는 모듈이다. 이는 [6]의 표준에 따라 구현하였으며, 동작원리는 그림 4와 같다.

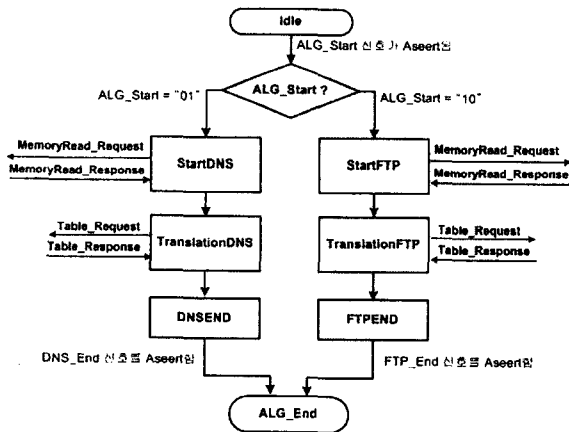


그림 4. ALGTrans 모듈의 동작 순서도

초기에 ALGTrans 모듈은 Idle 상태에 있다(ALG_Start는 "00"임). 이 상태에서 TransCore 모듈에서 ALG의 동작을 필요로 하는 경우 ALG_Start 신호의 값이 "00"(Idle)이 아닌 값으로 Assert된다. ALG_Start 신호가 "01"이면 DNSTrans 모듈을 구동하고, "10"이면 FTPTrans 모듈을 구동한다. DNS-ALG 요청의 경우, StartDNS 상태로 전이되는데, 이때 MemoryAccessArbiter 모듈을 통해 외부 메모리로부터 Payload 정보를 읽어온다. 다 읽은 후에는 TranslationDNS 상태로 전이되며, 이 상태에서는 읽어 온 DNS Header를 분석한 후 변환을 수행하게 되는데 이때 TableAccessManager 모듈을 통해 필요한 변환 정보를 TransCore 모듈로부터 받게 된다. 변환과정이 완료되면 DNS_End 상태로 전이되며, 이 상태에서는 DNS_End 신호를 '1'로 Assert함으로써, DNS-ALG 수행의 완료를 알린다. 그리고 변환된 DNS 메시지는 외부 메모리의 해당 영역으로 저장된다.

위와 같이 설계한 모듈을 각각의 모듈별로 VHDL을 사용하여 구현하였고, 각 모듈의 동작은 소프트웨어 시뮬레이션으로 테스트하였다.

4. 동작테스트

전체 모듈의 동작은 ALGTrans 모듈을 TransCore 모듈에 통합한 후 전용 FPGA 보드에 탑재하여 DNS_ALG와 FTP_ALG의 동작을 테스트하였다. 이를 위해 사용된 시험망 환경은 그림 5에 제시된 바와 같다.

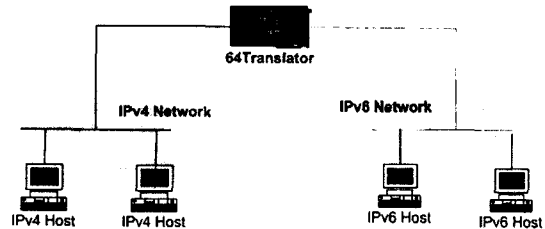


그림 5. ALGTrans 모듈의 테스트 환경

이러한 테스트 환경에서 다음과 같이 ALG의 동작을 검증하였다. 먼저, DNS_ALG의 테스트는 DNS Name lookup을 호스트에서 실행하여 이에 대한 응답을 체크함으로써 기능을 검증하였다. 다음으로 FTP_ALG의 테스트는 FTP 클라이언트 프로그램을 통해 FTP 서버로 FTP Connection을 연결하여 IPv4 호스트와 IPv6 호스트와의 연결 설정과 파일 송수신을 테스트함으로써 기능을 검증하였다.

5. 결론

풍부한 주소 공간과 자동 주소 할당, 보안 기능 등의 향상된 인터넷 프로토콜인 IPv6의 도입을 위해서는 IPv6 Network와 IPv4 Network 간의 원활한 통신을 제공하는 IPv6-IPv4 프로토콜 변환기가 필수적이다. 그러나 기존에 구현된 IPv6-IPv4 프로토콜 변환기는 운영 체제 기반의 소프트웨어로 구현되어 있어 성능 상의 한계가 있다.

이러한 문제를 해결하기 위해 고성능 IPv6-IPv4 프로토콜 변환기로 하드웨어 기반의 64Translator가 제안되었다. 그런데 64Translator가 다양한 응용 환경에서 동작하기 위해서는 응용 프로토콜 변환을 제공하는 ALG 모듈이 필수적이다. 이에 본 논문에서는 64Translator를 위한 ALG 모듈을 하드웨어로 설계 및 구현하였다. 구현된 ALG 모듈은 64Translator의 변환 모듈과 통합되어 기능 검증을 수행하였다. 이로써 64Translator는 기본적인 응용 환경에서 IPv6 Network과 IPv4 Network 간의 고속 통신을 제공할 수 있다.

향후 과제로는 기구현 모듈에 대한 성능을 분석함으로써 코드를 최적화하고, SIP에 대한 ALG를 추가 구현하는 방향으로 진행할 예정이다.

참고 문헌

- [1] 이승민, "NAT-PT, SIIT Technical Report", 광운대학교, 2000.
- [2] 이승민 외 3명 "IPv4와 IPv6의 연동과 호환을 위한 NAT-PT에 관한 연구", 광운대학교, 2003.
- [3] 공인엽 외 3명, "고성능 IPv6-IPv4 프로토콜 변환기의 구현에 관한 연구", 정보과학회 2003년 추계학술대회 제출.
- [4] 이경렬 외 3명 "IPv6-IPv4 프로토콜 변환기의 하드웨어 설계 및 구현", 정보과학회 2003년 추계학술대회 제출.
- [5] P. Srisuresh 외 3명, "DNS extensions to Network Address Translators (DNS_ALG)", IETF RFC 2694, 1999.
- [6] M. Allman 외 2명, "FTP Extensions for IPv6 and NATs", IETF RFC 2428, 1998.