

Draft24 기반의 Mobile IPv6 하드웨어 설계 및 구현

김영희^o 공인엽 이정태

부산대학교 컴퓨터공학과

{ccorange^o, leafgirl, jilee}@pusan.ac.kr

Design and Implementation of H/W Mobile IPv6 based on Draft 24

YoungHee Kim^o InYeup Kong JungTae Lee

Dept. of Computer Engineering, Pusan National University

요 약

Mobile IPv6 기술은 단일의 이동성을 제공하는 기술로서 All-IP기반의 차세대 유무선 통합 망 구축을 위한 핵심 기술이다. 기존의 소프트웨어 Mobile IPv6는 특정 OS에 의존적일 뿐 아니라, OS 구동 시 발생하는 오버 헤드로 인한 높은 전력 소모로 인해 시스템의 성능이 저하되는 문제점이 있다. 이에 본 논문에서는 기존 소프트웨어 기반 Mobile IPv6의 문제점을 해결하고 최신 표준에 따르는 이동성과 고속 통신을 제공하기 위한 하드웨어 Mobile IPv6를 설계 및 구현하였다. 그리고 구현된 모듈은 시험 망 패킷 데이터와 시뮬레이션을 통해 동작을 검증하였다.

1. 서 론

최근 무선 인터넷에 대한 수요가 증가하면서 휴대폰을 이용하여 음성 위주의 이동 통신 서비스만을 이용하는 방식에서 문자 데이터 및 고 음질, 고 화질의 멀티미디어 데이터를 광대역 고속 서비스로 이용하는 형태로 바뀌고 있다. 이러한 요구를 수용하기 위한 기술 중 하나가 Mobile IPv6 기술이다. IP 기반의 차세대 무선 인터넷 망은 궁극적으로 각 기지국에 라우터가 구현되는 형태를 가정하고 있으므로 IP 계층에서의 이동성 제공은 필수적인 요소이다[1].

그러나 기존에 구현된 OS 기반의 Mobile IPv6는 특정 OS에 의존적일 뿐 아니라, OS 구동 시 발생하는 오버 헤드로 인한 높은 전력 소모로 인해 시스템의 성능이 저하되는 문제점을 안고 있다. 또한 이를 해결하고자 했던 기존의 Draft13 기반의 Mobile IPv6 하드웨어 모듈[2]은 메시지 포맷상의 많은 변화로 인해 기존의 Mobile IPv6 단말과의 통신 시 호환이 불가능하다. 따라서 최신 표준 버전인 Draft24[3][4]를 기준으로 한 Mobile IPv6 하드웨어 구현이 필요하게 되었다.

이에 본 논문에서는 Draft24 [3][4]를 기준으로 Mobile IPv6를 하드웨어로 구현함으로써, 기존 Mobile IPv6의 한계를 해결하고, 이동성이 지원되는 고속 통신을 제공할 수 있게 하였다.

본 논문의 전체 구성은 다음과 같다. 먼저 2장에서는 Mobile IPv6의 기본 동작을 설명하고, 3장에서는 하드웨어 Mobile IPv6의 설계 및 구현에 대한 내용을 설명하였고, 4장에서는 모듈의 설계 검증 방법과 시뮬레이션 결과에 대한 내용을 설명하고, 5장에서는 결론 및 향후 과제에 대해 기술하였다.

2. Mobile IPv6 프로토콜

Mobile IPv6 프로토콜은 이동 단말에게 끊임 없는 통신을 위한 이동성을 제공하는 프로토콜이다. 이것은 이동 단말의 이동 시, IP layer에서 Home Address와 Care-of Address를 교환하여 상위 계층으로 전달함으로써, 상위 레이어에게 투명한 이동성을 제공한다.

Mobile IPv6를 하드웨어로 구현하기 위해서는 본 연구실에서 기구현한 TCP/IPv6코어의 IPv6 모듈과 ICMPv6 모듈에 Mobile IPv6 관련 기능들의 수정 및 추가가 필요하다.

IPv6, ICMPv6 메시지를 이용한 Mobile IPv6의 전체 동작 과정은 그림 1과 같다.

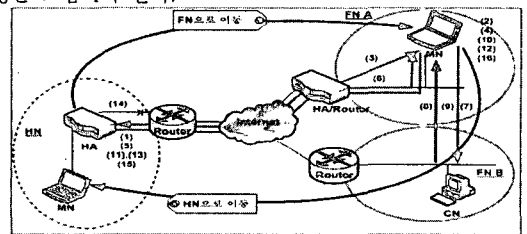


그림 1. Mobile IPv6 기본 동작

이동 단말인 MN(Mobile Node)는 HN(Home Network)에서 CN(Correspondent Node)과 통신하던 중, 다른 네트워크(Foreign Network)로 이동한다. MN는 이동 후, 30초 동안 HN의 Router로부터 RA 메시지를 수신 대기한다. 30초 경과 후, RA(Router Advertisement) 메시지를 수신하지 못하는 경우, Router의 연결 상태를 확인하기 위해 NS(Neighbor Solicitation) 메시지를 송신한다(1). 3초 경과 후에도, NA(Neighbor Advertisement) 메시지를 수신하지 못한 경우, IPv6의 NUD(Neighbor Unreachable Detection: 이웃 노드의 연결 상태 확인 매커니즘)를 수행한다. 이 과정에서 요청한 NS 메시지에 대한 응답으로, NA 메시지를 수신하지 못한 경우 MN는 FN으로의 이동을 검출하게 된다. 만약 NA 메시지를 수신하는 경우, 이동하지 않았다고 판단한다(2).

이동 감지 후, IPv6의 Auto-configuration 기능을 활용하여 FN의 라우터로부터 수신한 RA 메시지(3)를 통해 획득한

Prefix 정보를 통해 COA(Care-of Address)를 생성한다(4). 그리고, MN는 자신의 새로운 위치 정보를 알려주기 위해 HA에게 BU(Binding Update) 메시지를 송신하고(5), 이에 대한 응답으로 BA(Binding Acknowledgement) 메시지를 수신한다(6). BA 수신을 통해 적절한 Binding 정보를 갱신한 후, CN과의 BU송신(7), BA수신(8) 과정을 수행함으로써, HA를 경유하지 않는 직접 통신을 시도한다(9).

MN가 다시 FN에서 HN으로 이동하는 경우에도 MN는 자신의 새로운 위치 정보를 알리고, CN과 직접 통신을 하기 위한 HA, CN과의 Binding과정을 수행한다. 이 때, FN에서 수행했던, 그림 1의 (1)~(9)과정을 동일하게 수행하되, Binding 생성의 의미가 아닌, Binding 해제의 의미를 갖는다. 이것은 FN에서 유효했던, CoA의 사용을 무효화하고, HoA(Home Address)를 다시 사용함을 알리는 과정이다. (1)~(9)의 과정은 MN의 네트워크 이동 시, 기본적으로 발생하는 이벤트인 반면, (10)~(16)의 과정은 상황에 따라 수행되는 과정이다. 단, MN의 Home Address는 Prefix의 lifetime만기로 인해 변경될 수 있다. 따라서 이 때, HN의 HA가 송신한 UMPA(UnSolicited Mobile Prefix Advertisement) 메시지를 수신하고(10) 이에 대한 응답으로 MPS(Mobile Prefix Solicitation) 메시지를 송신(11), HA의 (11)에 대한 응답으로 MPA 메시지를 송신함으로써, HA와의 동적 Home Address 갱신 과정은 완료된다(13). 또한 HA의 링크 결함 또는 손실(14)로 인해 HA에 접근할 수 없을 경우, MN는 DHAAD(Dynamic Home Agent Address Discovery) Request 메시지를 송신하고(15), DHAAD Reply 메시지를 수신함으로써 동적으로 HA의 Address를 갱신하는 과정을 수행한다(16).

3. Mobile IPv6 프로토콜의 설계 및 구현

3.1 프로토콜별 기능 규격

표 1은 Mobile IPv6 표준 중에서 이동 단말이 통신시 필요한 기능과 발생 빈도 수가 높은 예외 사항들을 정리한 내용이다.

표 1. Mobile IPv6 하드웨어 모듈의 프로토콜별 기능 규격

프로토콜	기능	메시지
ICMPV6	이동 검출	<ul style="list-style-type: none"> RA 메시지 수신 <ul style="list-style-type: none"> - RA Interval Option - HA Information Option - HA Interval Option
	주소 생성	RS, RA, NS, NAD메시지 송신
	HoA 동적 생성	MPS, MPA 메시지 수신
	HA Address 동적 생성	DHAAD Request/ Reply메시지
IPv6 확장 헤더	Binding 정보 유지 관리	<ul style="list-style-type: none"> Home Address Option Type2 Routing Header Mobility Header <ul style="list-style-type: none"> - BU, BA, BE, BRR -Binding Refresh Advice Option 데이터 구조 구현: BU List

표 1에 제시된 기능들을 프로토콜별로 설계 및 구현한 상세 내용은 3.2절과 같다.

3.2 전체 구조

본 설계는 기 구현된 TCP/IPV6 모듈을 기반으로하며, 전체 구조는 그림 2와 같다[5]. Mobile IPv6 하드웨어 모듈은 Mobile

IPv6 세부 기능 모듈들간의 동작을 제어하는 MIP6CORE 모듈, 기존의 ICMPV6모듈에 새롭게 Mobility 기능을 추가 구현한 ICMPV6SEND 모듈/ICMPV6 RECV모듈, Mobility기능 지원을 위한 IPV6EXTSEND 모듈/IPV6EXT RECV 모듈로 구성된다.

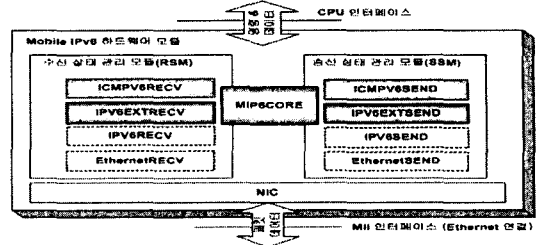


그림 2. Mobile IPv6 하드웨어 모듈의 전체 구성도

그림 2의 EthernetSEND 모듈/ EthernetRECV 모듈, IPV6SEND 모듈/IPV6RECV 모듈은 각각 Ethernet 송신, Ethernet 수신, IPv6 송신 모듈/IPv6 수신 모듈은 하드웨어로 구현한 TCP/IPV6 코어의 기존 모듈을 그대로 사용한다. ICMPV6SEND 모듈과 ICMPV6RECV모듈은 기존 ICMPV6 기능에 표 1의 Mobile IPv6 관련 ICMPV6 메시지 생성, 분석, 송수신 기능을 추가 구현한 모듈이다. IPV6EXTSEND 모듈과 IPV6EXTRECV 모듈은 표 1의 Mobile IPv6 관련 IPV6 확장 헤더 생성, 분석, 송수신 기능을 추가 구현한 모듈이다. 그리고 RSM(Receive State Manager: 수신 상태 관리 모듈)은 Ethernet 수신 모듈, IPv6 수신 모듈, ICMPV6 수신 모듈을 제어하여 각 프로토콜에 대한 수신 처리를 담당하는 모듈이다 SSM(Send State Manager: 송신 상태 관리 모듈)은 세부 모듈인 Ethernet 송신 모듈, IPv6 송신 모듈, ICMPV6 송신 모듈을 제어하여 각 프로토콜에 대한 송신 처리를 담당하는 모듈이다[5].

MIP6CORE 모듈은 Mobile IPv6 기능을 지원하는 각 세부 기능 모듈간의 동작 연계 제어 및 ICMPV6SEND 모듈/ICMPV6RECV 모듈 또는 IPV6EXTSEND 모듈/IPV6EXTRECV 모듈과의 인터페이스를 제공한다. 3.2절에서는 MIP6CORE 모듈의 Mobile IPv6의 세부 기능 모듈간의 동작 제어를 중심으로 상세히 설명하도록 한다.

3.3 MIP6CORE 모듈의 설계 및 구현

그림 4는 MIP6CORE 모듈의 세부 모듈 구성 및 외부 모듈과의 관계를 보여준다.

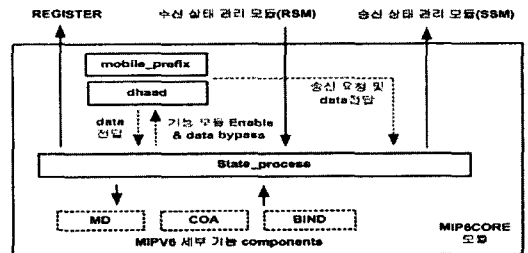


그림 3. MIP6CORE모듈의 세부 모듈 구성

MIP6CORE 모듈은 표 1의 기능들을 구현하기 위해 각각 3개의 Process와 3개의 component들로 구성된다. 3개의 process는 MIP6CORE모듈의 전체 동작 상태를 결정하고 세부 기능 모듈들의 동작을 제어하는 state process, Prefix discovery 기능을 제공하는 mobile_prefix process, Dynamic

Home Agent Address Discovery의 간단한 메시지 송수신 과정을 처리하는 dhaad process로 구성된다. 또한 Mobile IPv6 세부 기능 모듈들은 MD 모듈, CoA 모듈, BIND 모듈로 구성된다. MIP6CORE 모듈은 RSM으로부터 데이터를 수신하여 데이터를 분석, 내부 세부 기능 모듈들의 동작을 제어한다. 그리고 내부 세부 기능 모듈로부터 데이터를 수신하여 레지스터에 저장하거나 SSM으로 데이터를 bypass한다.

그림 4는 MN이 HN에서 FN으로 이동한 경우, MIP6CORE 모듈의 내부 세부 기능 모듈들의 동작 제어 흐름을 보여준다.

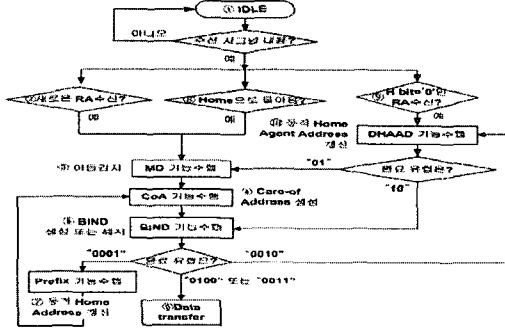


그림 4. MIP6CORE 모듈의 동작 제어 흐름도

MIP6CORE 모듈은 TCP/IPv6기반의 하드웨어 칩과 함께 초기화 되어 IDLE 상태에서 MN의 위치 이동에 상관없이 현재 네트워크의 정보(Router MAC, IPv6 주소, MAC, IPv6 주소)를 임시 저장한다(①). 그리고 RSM으로부터 새로운 RA 수신 알림 신호를 수신한 후(②), MD 모듈을 Enable한다(③). 이때, MD 모듈로부터 수신한 NS 메시지에 대한 송신 요청과 메시지 타입을 전달받아 SSM 모듈에게 ICMPV6SEND 모듈과 IPV6SEND 모듈의 동작을 지시하고, 해당 메시지 유형에 대한 정보를 Bypass한다. CoA 주소 생성 과정은 기존의 IPv6 Auto-configuration 매커니즘과 같은 ICMPv6 메시지 송수신 과정 없이 FN에서 MIP6CORE 모듈에 의해 임시 저장된 FN의 네트워크 정보를 전달받아 CoA를 생성한다(④). 생성한 주소는 주소 유형에 따라 IPv6 레지스터 또는 Binding Update List에 저장된다. 이후, Binding 생성을 위한 BIND 모듈을 Enable하고(⑤), BIND 모듈로부터 BU 메시지를 구성하는 필드의 데이터 값들과 목적지 주소 정보를 전달받아 SSM에게 IPV6EXTSEND 모듈과 IPV6SEND 모듈의 동작을 지시하고, 해당 메시지 유형에 대한 정보를 Bypass한다. 이후, RSM으로부터 BA를 수신하여 데이터 정보를 BIND 모듈에게 전달한 후, BIND 모듈의 완료 유형에 따라 각각 DHAAD 모듈, Prefix 모듈, Data Transfer 상태 전이하여 다음 동작 과정을 수행한다. 이 때 동작 과정은 'RSM으로부터 데이터 수신→분석→세부 모듈 Enable 및 데이터 전달→SSM에게 해당 송신 모듈 동작 지시 및 데이터 전달→완료 신호 수신→다음 동작 상태 전이'의 흐름으로 MD 또는 CoA 주소 생성 과정과 동일하게 제어한다. MIP6CORE 모듈을 그림2의 나머지 모듈들의 설계에 있어서도 3.2절에서 설명한 세부 모듈 구성과 동작 흐름 방식으로 설계하였다.

4. 구현 및 검증

구현된 TCP/IPv6 기반의 Mobile IPv6 하드웨어 모듈의 VHDL 코드를 통해 VHDL 코드 내 문법상의 오류를 정정하였다. 본 논문에서는 VHDL을 사용하여 3장에서 설명한 설계 내용을 구현하였고, 소프트웨어 시뮬레이션 테스트를 통해 시뮬레이션 시에는 그림 5에서 보는 바와 같이 소프트웨어 Mobile IPv6 시험 망에서 수집한 송수신 패킷을 테스트 벡터와 Draft24

[3][4]를 기준으로 예상되는 결과를 비교, 분석하여 시뮬레이션 하였다.

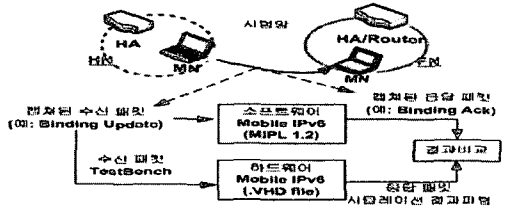


그림 5. 소프트웨어 시뮬레이션 검증 방법

그림 6은 그림 5의 개념을 바탕으로 BIND 모듈의 BU/BA 송수신 과정에 대한 시뮬레이션 결과를 예로 제시한 것이다.

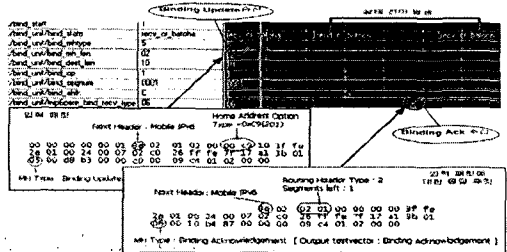


그림 6. 시뮬레이션 결과의 예 : BU/BA 송수신

5. 결론 및 향후 과제

Mobile IPv6는 단말의 이동성 지원을 위해 필수적인 프로토콜이지만 현재 소프트웨어로 구현된 모듈은 특정 OS에 의존적이며, 이로 인해 많은 전력을 소모하며 성능이 저하되는 문제점이 있었다.

이에 본 논문에서는 이동 단말을 위해 OS에 독립적인 고성능의 하드웨어 Mobile IPv6를 설계 및 구현하였다. 또한 소프트웨어 Mobile IPv6 시험 망에서 다양한 시나리오를 통해 수집한 테스트 벡터를 이용한 소프트웨어 시뮬레이션을 수행하였다.

향후에는 구현한 VHDL 코드를 FPGA 보드에 탑재하여 다양한 환경에서 에뮬레이션 테스트함으로써, 기능을 안정화하고, 코드 최적화 연구를 진행할 예정이다.

참고문헌

- [1] 한연희, "IPv6 이동성 지원 기술", TTA 저널 91호, 2004
- [2] 황보 문, "Mobile IPv6 프로토콜용 하드웨어 설계 및 구현", 한국정보과학회 가을학술대회 발표논문집(III), 제29권, 제2호, pp.37-39, 2002.
- [3] C. Perkins, "Mobility Support in IPv6", RFC 3775, 2004.
- [4] 박재홍 외 4명, "Mobile IPv6 Draft24", IPv6 포럼 기술문서, 2003.
- [5] 이정태 외 5명, "TCP/IP 하드웨어 칩 모듈화 및 64변환기 타당성 검토", wiznet, Dec.2002.