

박 상 역*, 윤 상 균**

연세대학교 전산학과

*ahsld77@empal.com, **skyun@dragon.yonsei.ac.kr

Hardware Architecture for Efficient IPv6 Address Lookup

*Sang-Eok Park, **Sang Kyun Yun

Department of Computer Science, Yonsei University

요약

인터넷 수요의 급증에 따라서 인터넷 주소에 대한 수요가 급증하였으며 기존의 IPv4의 주소의 고갈이 예상됨에 따라서 이를 해결하기 위해서 IPv6 주소 체계가 제시되었다. 128비트 크기의 IPv6에 주소 검색은 IPv4에 비해서 더 복잡해진다. 그렇지만 IPv6의 주소 할당 체계를 이용하면 주소 검색을 다소 간단하게 할 수 있다. 본 논문에서는 주소의 상위부분 검색은 PLA와 CAM을, 하위부분 검색은 RAM을 사용하는 2단계 구조의 IPv6 주소 검색을 위한 하드웨어 구조를 제시하였다. 제시된 구조는 CAM과 RAM의 확장성과 적절한 정도의 메모리 공간 사용 효율을 가진다. .

1. 서론

현재의 인터넷의 상황은 인터넷 사용의 급격한 보급과 다양한 응용 서비스의 등장에 따라서 신규 IP 주소에 대한 수요가 급증하는 데 비해서 IPv4 주소는 32비트의 길이 제한과 비효율적인 주소 할당으로 인해서 주소의 고갈이 임박해 오고 있다. 이러한 문제를 해결하기 위해서 풍부한 IP주소를 제공하는 인터넷 주소의 차세대 표준인 IPv6[1]가 제시되었으며 인터넷 주소 체계는 IPv4로부터 IPv6으로의 점진적인 교체가 진행되고 있다. 이러한 과정에서 기존의 IPv4 라우터는 이를 수용할 수 없게 되었고 IPv6에 적합한 라우터에 대한 연구가 필요하게 되었다.

라우터에서 그 성능의 결정짓는 가장 큰 요소 중에 하나가 입력된 패킷에서 IP주소의 prefix와 일치하는 라우팅 테이블의 엔트리를 찾아서 출력 포트의 정보를 알아내는 과정인 주소 검색 과정이다. 이러한 주소 검색에 방법에 대한 연구는 활발하게 진행되어 왔으며 검색 방법은 소프트웨어적인 방법과 하드웨어의 지원을 받는 방법으로 구분될 수 있다. 고속의 검색을 위해서는 소프트웨어적인 방법만으로는 한계가 있으며 하드웨어의 지원이 필요하다.

IPv6는 IPv4의 4배인 128비트의 길이를 가지며 기존의 IPv4에서 적용한 방법을 단순히 확장하여 사용하면 라우팅 테이블의 크기가 훨씬 더 커지게 되며 IP주소 검색이 더 복잡해진다. 그렇지만 IPv6는 비효율적인 주소할당 방법을 사용하는 IPv4와는 달리 체계적인 주소 할당 체계를 가지고 있다. IPv6에서는 거대한 인터넷 백본으로부터 작은 조직 내의 개별 서브넷(subnet)까지를 체계적으로 나타낼 수 있도록 서브네팅을 다단계화 하였으며, ISP(internet service provider)에서 상위 ISP로부터 주소의 일부분을 할

당받도록 하는 계층적인 체계로 이루어진다[2].

기존의 IPv6 주소 검색에 대한 연구들은 IPv4 주소 검색 방법을 기반으로 하여 확장한 수준의 연구에 그치고 있으며 IPv6의 주소 할당 체계의 특징을 반영하지 못하고 있다. IPv6와 IPv4의 주소 할당 체계의 차이점으로 인하여 기존의 IPv4의 주소 검색 방법이 IPv6에서는 최상의 방법이 될 수 없으며, IPv6에 적합한 주소 검색 방법이 요구된다. 본 논문에서는 IPv6의 특징을 잘 반영한 CAM과 RAM으로 구성되는 효율적인 주소 검색을 위한 하드웨어 구조를 제안하고자 한다.

본 논문의 구성은 다음과 같다. 2절에서는 일반적인 IP 주소 검색과 IPv6에 관한 주소 검색에 대한 관련 연구를 소개하며, 3절에서는 IPv6 주소 할당 체계의 특징에 대해서 살펴보고, 4절에서는 본 논문에서 제안하는 하드웨어에 기반한 주소 검색 구조에 대해서 설명을 하고, 5절에서는 제안된 구조에 대한 평가를 하고, 마지막으로 6절에서는 결론과 향후 연구 과제에 대해서 제시한다.

2. 관련 연구

최근 몇 년간 인터넷 라우터를 위한 IP주소 검색에 관한 다양한 방법들이 연구되었다. 이러한 방법 중 일반적인 방법과 IPv6에 관한 주소 검색 방법에 대하면 다음과 같다.

첫 번째 검색 방법으로는 trie 구조를 이용한 검색 방법 [3]으로 주소 검색과 관련된 연구 중에 가장 많이 연구되는 방법이다. 트라이는 prefix를 tree형태로 저장하여 주소를 검색하는 것으로 각 노드는 주소의 한 비트에 해당되며 각각의 비트마다 반복적으로 검사하여 가장 긴 prefix