

# 링 연결구조 기반의 멀티코어 프로세서를 위한 캐시 일관성 유지 기법

박진영<sup>o</sup> 최린

고려대학교 전자컴퓨터공학과

[o2id@korea.ac.kr](mailto:o2id@korea.ac.kr), [lchoi@korea.ac.kr](mailto:lchoi@korea.ac.kr)

## A Preliminary Study on a Cache Coherence Protocol for Multi-Core Processors with Ring Interconnects

Jinyoung, Park<sup>o</sup>, Lynn, Choi

Department of Electronics and Computer Engineering, Korea University

### 1. 서 론

SOC 기술의 발전과 더불어 최근 여러 개의 프로세서를 단일 칩에 집적한 멀티코어 프로세서가 기존 슈퍼스칼라 프로세서 구조에 비하여 보다 에너지 효율적으로 성능을 증가시키는 방안으로 채택되고 있다. 이에 온 칩 프로세서간 캐시 일관성 유지 문제가 시스템의 안정성과 성능에 큰 영향을 미치는 요소로 부각되고 있다. 또한, 종래의 공용 버스 기반의 온-칩 연결구조가 대역폭 향상을 위한 점대점(point-to-point) 구조로 전환되고 있다. 따라서, 이러한 새로운 온 칩 연결구조에 구현 가능한 멀티코어 프로세서에서의 캐시 일관성 유지 기법에 대한 지속적인 연구가 필요하다. 본 논문에서는 종래의 공용 버스가 가지는 문제점과 이를 극복하기 위해 단 방향 링 연결구조 기반의 멀티코어 프로세서에 적합한 캐시 일관성 유지 기법을 제시하고자 한다. 공용 버스 기반의 멀티코어의 경우 프로세서 코어의 개수가 많아질수록 공용 버스에 캐시 일관성 유지 요청이 집중되어 버스를 획득하지 못할 경우가 빈번히 발생할 수 있어 짧은 지연시간의 장점이 점점 줄어들어간다. 반면, 단 방향 링 연결구조 기반의 멀티코어일 경우 각 노드간 연결은 점대점 구조로 처리 될 수 있어 높은 대역폭을 유지할 수 있지만, 캐시 일관성 유지 요청의 처리 순서를 별도로 결정해 줄 필요가 있다. 본 논문에서는 단 방향 링 연결구조에서 데이터 전송 순서에 따라 캐시 일관성 유지 요청의 순서가 결정되는 RING-DATA ORDER 기법을 제안하여, 종래 기술인 ORDERING-POINT 기법의 긴 평균 지연시간 문제, GREEDY-ORDER 기법의 캐시 일관성 유지 채 요청 문제, RING-ORDER 기법의 토큰 관리 문제에 따른 성능 저하를 방지하고 GREEDY-ORDER와 RING-ORDER의 장점을 모두 가지도록 하였다.

### 2. 본 론

최근 몇 년간 단일 칩 멀티프로세서의 급격한 발전이 진행되고 있지만, 멀티프로세서에 관한 연구는 훨씬 먼저 시작 되었다. 공용 버스 기반의 스누핑 프로토콜의 경우 버스에 인가되는 모든 요청은 아비터(arbiter)에 의해 우선순위가 결정되기 때문에 원리는 간단한 반면, 노드의 개수가 많아질 경우 노드당 유효한 대역폭이 감소하고, 노드간에 서로 버스를 점유하기 위한 경쟁을 하게 되어 성능 저하가 발생한다. 디렉터리 프로토콜은 버스뿐 아니라 보다 고속의 다양한 연결구조에서도 사용가능 하지만, 단점으로 요청에 대한 지연시간이 버스 기반의 스누핑 프로토콜 보다 상대적으로 길며, 디렉터리 정보를 유지하기 위한 메모리가 필요하다. Milo M. K. Martin[1]은 캐시 일관성 유지 기법을 정확성과 성능 정책으로 분리하였으며, 성능 정책에 따라 스누핑 프로토콜과 디렉터리 프로토콜의 장점을 적절히 가질 수 있도록 한 토큰 일관성(Token Coherence) 기법을 제안하였다.

링 연결구조 기반의 캐시 일관성 유지를 위한 기법으로 ORDERING-POINT, GREEDY-ORDER, RING-ORDER등이 있다. ORDERING-POINT를 이용한 방법은 링 연결구조의 특정 위치에 ORDERING-POINT를 삽입하여, 링에 있는 여러 노드들의 요청이 ORDERING-POINT에 먼저 도착하는 순서대로 처리 순서가 결정 된다. ORDERING-POINT를 삽입 함으로써 구현이 간단하다는 장점이 있는 반면, 각 노드의 요청이 ORDERING-POINT에 도착하는 홉(hop)의 평균이  $N/2$ 이고( $N$ =전체 노드 개수), ORDERING-POINT로부터 링을 한바퀴 돌고나서( $N$  홉), 요청을 발생시킨 노드에 되돌려 주는  $N/2$ 만큼의 홉이 소요되므로, 평균 적으로 링을 두바퀴 돌아야( $2N$  홉) 요청이 완료 된다. GREEDY-ORDER는 각 노드가 링에 내보낸 요청이 데이터를 먼저 획득하는 순서대로 처리 순서가 결정되는데, 노드간의 경쟁이 발생할 경우 데이터를 획득한 요청은 링을 한바퀴 돌아 완료( $N$

흡)되지만, 데이터를 획득하지 못할 경우 재 요청을 반복하게 되므로 평균 지연 시간은  $N(1+M)$ 흡(M:재 요청 회수)이 된다. 토큰 일반성에 기반한 RING-ORDER[2]의 경우 경쟁이 발생할 경우 특수토큰(priority token)을 먼저 획득한 노드의 요청부터 처리되는데, 특수토큰을 가진 노드는 자기가 필요로 하는 만큼의 토큰을 확보할 때까지 경쟁 관계의 요청이 오면 블로킹 하게 되는데, 지연시간 측면에서는 길어질 수 있지만, 시스템의 대역폭 효율성 측면에서는 모든 요청이 링을 한 바퀴만 돌기 때문에(N흡) 효율성은 높다. 반면, 각 노드에서 메모리 블록을 내쫓을 경우(eviction case) 해당 노드가 가지고 있는 토큰을 다른 노드로 안전하게 이동 시켜야 하는 문제가 있다.

본 논문에서 제안하는 RING-DATA ORDER는 요청을 내보낸 노드에 도착하는 데이터 전송을 동반하는 모든 요청을 블로킹하고 있다가, 자기가 내보낸 요청이 링을 한 바퀴 돌아 자신에게 도착하면 수행완료(N흡)하고, 블로킹한 요청을 링에 다시 내보내어 계속 수행되도록 한다. RING-DATA ORDER는 지연시간 측면에서 RING-ORDER와 같은 성능을 가지는 반면, RING-ORDER의 단점인 T개의 토큰을 유지하거나, 특히 메모리 블록을 내쫓을 경우 토큰을 안전하게 백업하는 등의 문제가 없으며, GREEDY-ORDER의 문제점인 재 요청이 발생하지 않으므로, 링 연결 구조의 대역폭 효율성이 높다.

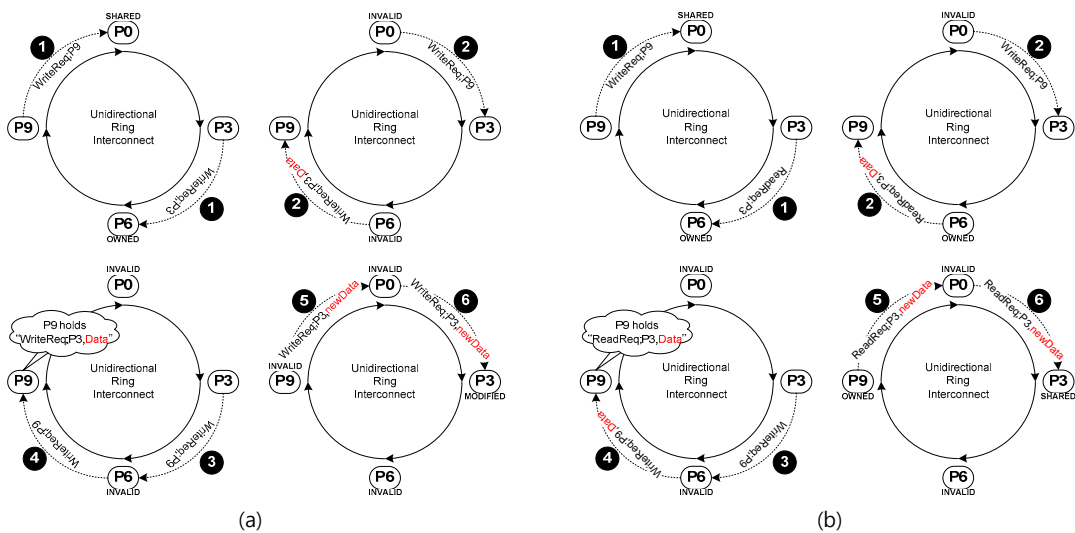


그림1. RING-DATA ORDER 기법의 예

종류	성능	장점	단점
ORDERING-POINT	2N	간단한 구조	긴 지연시간
GREEDY-ORDER	$N(1+M)$	대역폭 효율	재 요청 (M)
RING-ORDER	N	높은 대역폭 효율	토큰 관리, 충분한 버퍼 필요
RING-DATA ORDER	N	높은 대역폭 효율	충분한 버퍼 필요

표 1. 성능비교

### 3. 결 론

단 방향 링 연결구조 기반의 캐시 일관성 유지 방법인 RING-DATA ORDER는 링에 동시에 존재하는 요청의 처리 순서를 각 노드의 위치와 데이터 전송 동반 여부에 따라 결정 함으로써 시스템의 캐시 일관성을 유지 하도록 하였다. 단 방향의 링 연결구조를 사용 함으로써 각 노드는 점대점 연결을 하게 되어 고속의 동작이 가능하며, 버퍼를 이용하여 경쟁 관계의 요청들에 대한 순서를 결정함으로써 RING-ORDER의 장점과 GREEDY-ORDER의 장점을 모두 가지게 된다. 단 방향 링 연결구조 기반의 RING-DATA ORDER기법의 캐시 일관성 유지 방법은 공용 버스기반의 멀티코어보다 높은 대역폭 효율성을 달성하면서 캐시 일관성을 유지하기 위한 오버헤드가 크지 않으므로 링 연결구조 기반의 멀티코어 프로세서에 적합한 일관성 유지 기법이다.

### 참 고 문 헌

[1] Milo M.K. Martin, Token Coherence, PhD Thesis, University of Wisconsin-Madison, 2003  
 [2] Michael R. Marty and Mark D. Hill, Coherence Ordering for Ring-based Chip Multiprocessors, International Symposium on Microarchitecture, 2006.