

# 인메모리 컴퓨팅을 위한 최적의 메모리 구성 및 채널 개수에 대한 연구

김봉정\*, 김영규\*, 문병인\*\*  
\*경북대학교 전자전기컴퓨터학부  
\*\*경북대학교 IT 대학 전자공학부  
e-mail : leadcain@ee.knu.ac.kr

## A Study on Optimal Memory Configuration and the Number of Channels for In-Memory Computing

Bong-jeong Kim\*, Young-Kyu Kim\*, Byungin Moon\*\*

\*School of Electrical Engineering and Computer Science, Kyungpook National University

\*\* School of Electronics Engineering, Kyungpook National University

### 요 약

DRAM 가격의 하락으로 인메모리 컴퓨팅에 대한 연구 및 개발이 다시 활발해지고 있으나 효율적인 메모리 시스템 구성을 위한 연구는 아직 부족한 실정이다. 이에 본 논문은 64 비트 멀티프로세서와 대용량의 메모리로 구성되는 인메모리 컴퓨팅 시스템을 모델링하고, 메모리 크기 및 채널 개수에 따른 시스템의 성능을 시뮬레이션 하였다. 그리고 처리된 트랜잭션의 수를 성능평가의 기준으로 하여 메모리의 크기와 채널 개수에 따른 비용을 고려한 최적의 인메모리 컴퓨팅 메모리 시스템 구조를 제안하였다.

### 1. 서론

컴퓨터 시스템의 메인 메모리로 사용되는 DRAM의 가격이 큰 폭으로 하락하고 64 비트 프로세서가 보급되면서 서버급 시스템뿐만 아니라 일반 PC에 이르기까지 64 비트 시스템의 보급이 점차 확대되고 있다<sup>[1]</sup>. 이러한 64 비트 시스템의 메인 메모리 크기는 이론상 최대 128 TB 까지 확장이 가능하며 메모리의 용량이 충분히 큰 경우, 시스템에서 수행 중인 모든 프로그램을 메인 메모리에 적재할 수 있게 된다. 이는 기존의 32 비트 시스템과 비교했을 때, 시스템 전체 성능의 병목현상을 일으키는 페이지 교환과 페이지 부재에<sup>[2]</sup> 대해서 고려할 필요가 없는 이점을 가진다. 즉, 메인 메모리와 프로세서 사이에서만 데이터 전송이 이루어짐으로써, 긴 탐색시간을 가지는 하드디스크로의 접근이 불필요하게 되어 시스템 전체 성능이 비약적으로 증가하게 한다. 이러한 이유로 기존에 이론으로 정립되었던 인메모리 컴퓨팅(in-memory computing)에 대한 연구 및 개발이 최근에 다시 활발하게 이루어지고 있다<sup>[1]</sup>.

인메모리 컴퓨팅은 하드디스크를 사용하지 않고 모든 데이터를 메인 메모리에 저장하는 시스템으로, 데이터의 전송량이 많은 데이터베이스 서버 또는 엔터프라이즈 서버 등에 적용할 경우 기존의 하드디스크 기반의 서버 시스템보다 우수한 성능을 기대할 수 있

기 때문에 그 중요성이 크게 증가하고 있다<sup>[1]</sup>. 하지만 인메모리 컴퓨팅에 대한 최근의 연구 동향은 소프트웨어 기반의 데이터베이스 시스템을 위한 연구가 대부분이며 효율적인 하드웨어 구조에 대한 연구는 부족한 실정이다.

이에 본 논문은 인메모리 컴퓨팅에서 효율적인 메모리 시스템의 구조를 알아보기 위해, 메모리 채널의 수, 메모리의 페이지 정책, 메모리 컨트롤러의 구조 등을 메모리 시스템 시뮬레이터인 DRAMSim2를 수정하여 모델링 하였고<sup>[3]</sup>, 시뮬레이션을 통하여 메모리의 크기와 채널 개수에 따른 시스템 성능을 평가 하였다.

### 2. 시뮬레이션을 위한 메모리 시스템 모델링

싱글 프로세서 시스템의 메모리 접근방법에 대한 연구들 대부분은 page mode policy의 성능을 극대화할 수 있는 open page 기반의 정책 연구가 주를 이루었다<sup>[4][5]</sup>. 하지만 고성능 멀티프로세서의 보급으로 인해 효율적인 병렬처리에 대한 중요성이 강조되었고, 데이터의 공간적 지역성(spatial locality)을 활용하는 정책들보다 프로세서들의 병렬성을 향상시킬 수 있는 close page 기반 정책들에 더욱 주목하게 되었다<sup>[6]</sup>. 이에 본 논문에서는 64 비트 멀티프로세서 기반의 인메모리 컴퓨팅 시스템을 목표로 하여 모델링 하였으며, 메인 메모리의 페이지 정책은 close page policy로 설정하여 시뮬레이션 하였다<sup>[3]</sup>.

Close page policy를 효과적으로 활용하기 위해서는

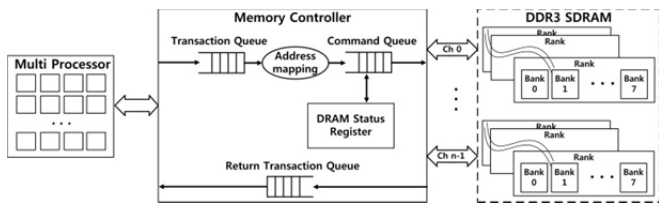
본 논문은 경북대-삼성전자 반도체 산학협력위원회 연구과제에 의해 지원된 연구 결과입니다.

물리적 어드레스 또는 캐시 어드레스를 DRAM 접근을 위한 메모리 어드레스로 변환하는 어드레스 맵핑 스킴(address mapping scheme)의 선정 또한 중요하다. 일반적으로, 공간적 지역성의 원칙에 따라 현재 메모리에 접근하는 물리적 어드레스의 비트들과 다음 어드레스 비트들의 변화율을 각각 비교했을 때 물리적 어드레스의 LSB 에 가까이 위치한 비트들일수록 MSB 측면에 위치한 비트들보다 더욱 빈번한 변화율을 보인다. 그러므로 논리적 어드레스의 하위 비트를 물리적 어드레스의 채널, 랭크, 뱅크의 필드로 결정하면 메모리의 인터리빙(interleaving)을 최대한 활용할 수 있게 된다<sup>7)</sup>. 이러한 사실을 고려하여 본 논문에서는 DRAMSim2 에서 제공하는 어드레스 맵핑 스킴들 중 인터리빙을 최대한 활용할 수 있는 어드레스 맵핑 스킴을 사용하였으며 어드레스의 각 필드들은 그림 1 과 같이 구성된다.

Row	Column	Bank	Rank	CH
-----	--------	------	------	----

(그림 1) 어드레스 맵핑 스킴

본 논문에서는 가장 보편적인 메모리 시스템을 모델링 하고 시뮬레이션 하기 위하여 하나의 메모리 컨트롤러와 공유 메모리를 사용하는 구조로 DRAMSim2 를 수정하여 모델링 하였으며 아래의 그림 2 는 본 논문에서 모델링 한 메모리 시스템의 구조를 보여준다.



(그림 2) 메모리 시스템의 모델 구조

멀티 프로세서는 64 비트 프로세서 32 개로 구성되어 있고 메모리 접근 요청을 위한 트랜잭션을 생성하는 기능을 한다. 메모리 컨트롤러는 프로세서들로부터 수집된 트랜잭션을 각각의 메모리로 전달하는 역할을 수행하며, 앞에서 설명했던 page 정책 및 어드레스 맵핑 스킴이 모두 메모리 컨트롤러에서 처리된다. 메모리 컨트롤러는 프로세서에서 생성되는 트랜잭션들을 매 사이클마다 하나씩 획득하여 트랜잭션 큐에 저장한 후 순차적으로 커맨드 큐로 내보내면서 메모리 접근을 위한 어드레스로 변환한다. 커맨드 큐에서는 메모리 컨트롤러의 레지스터에 저장되어 있는 메모리의 상태정보를 이용하여 뱅크보다 랭크를 우선순위로 둔 라운드 로빈 스케줄링 기법을 이용하여 트랜잭션을 메모리로 전송한다. 메모리에서 읽혀진 데이터는 리턴 트랜잭션 큐로 전송된 후, 다시 프로세서로 전송된다.

### 3. 실험환경

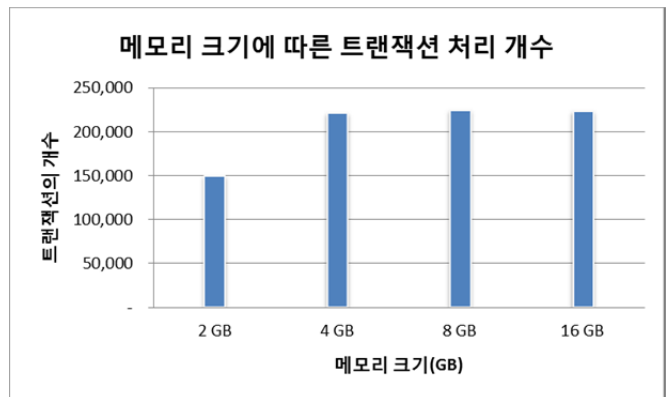
표 1 은 2 장에서 설명한 메모리 본 연구의 시뮬레

이션을 위한 시스템 환경을 보여준다.

<표 1> 시뮬레이션을 위한 시스템 환경

프로세서 개수	64 비트 프로세서 32 개
메인 메모리	Micron DDR3 MT41J256M4JP-125
메모리 채널	2, 4, 8, 16, 32
페이지 정책	Close page Policy
수행 사이클	1,000,000

본 논문은 대용량의 메모리를 사용하는 시스템을 대상으로 하기 때문에 메모리의 인터리빙 정도가 시스템의 성능향상에 미치는 한계를 조사하여, 메모리의 용량 증가로 인해 시스템의 성능이 향상되는 범위의 메모리 크기는 실험환경에서 제외 하였다. 이를 위해 싱글 채널에서 메모리의 크기에 따른 트랜잭션 처리량을 비교하는 시뮬레이션을 하였고, 시뮬레이션 결과는 그림 3 과 같이 메모리의 크기가 4 GB 인 경우에 랭크와 뱅크 간의 인터리빙을 통한 성능향상은 거의 최대로 나타났고, 8 GB 에서 그 한계점에 도달했음을 확인할 수 있었다. 따라서 8 GB 이상에서 메모리의 크기는 트랜잭션의 처리량에 영향을 주지 않기에 메모리의 최소 크기를 64 GB 로 두고 시뮬레이션을 진행하였다.



(그림 3) 메모리 크기에 따른 트랜잭션의 처리 개수

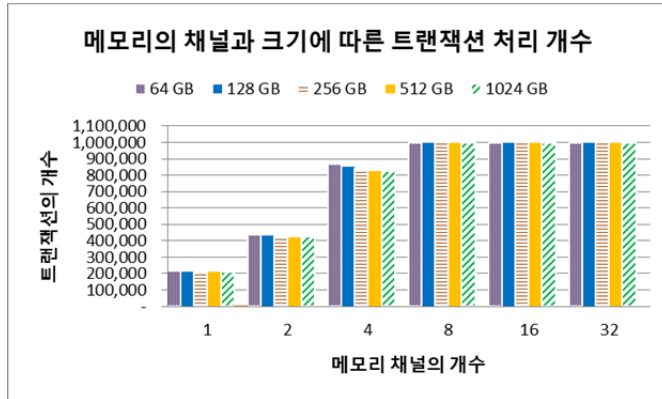
표 2 는 메모리 크기 및 채널에 따른 랭크의 개수를 보여주며, 본 연구에서 제안된 구조를 시뮬레이션 하기 위해 메모리의 채널과 크기를 정리한 것이다. 메모리는 DRAMSim2 에서 제공되는 Micron DDR3 MT41J256M4JP-125 를 사용하였으며<sup>8)</sup>, 한 채널에 연결되는 랭크의 개수는 메모리 채널의 개수 및 메모리의 크기에 의해 결정된다.

<표 2> 메모리 크기 및 채널에 따른 랭크의 개수

채널\크기	64 GB	128 GB	256 GB	512 GB	1024 GB
1	32	64	128	256	512
2	16	32	64	128	256
4	8	16	32	64	128
8	4	8	16	32	64
16	2	4	8	16	32
32	1	2	4	8	16

#### 4. 실험결과

앞서 언급한 표 2 와 같이 채널의 수를 1, 2, 4, 8, 16, 32 개로 두고 각각의 채널에 대해 메모리의 크기를 64 GB, 128 GB, 256 GB, 512 GB, 1024 GB 로 변경하면서 1,000,000 사이클 동안 각각 시뮬레이션을 수행하였다.



(그림 4) 메모리의 채널과 크기에 따른 트랜잭션 처리 개수

그림 4 는 메모리의 크기와 채널의 개수에 따른 트랜잭션 처리량을 보여준다. 채널의 개수가 4 개 미만 일 때는 채널의 개수가 증가함에 따라 트랜잭션 처리량이 두 배 가까이 향상되었지만, 채널의 개수가 4 개에서 8 개로 증가할 때는 트랜잭션 처리량이 대략 18% 정도만 향상 되었다. 그리고 채널의 수가 8 개 이상인 경우에는 더 이상의 성능 향상은 되지 않았는데, 이는 본 논문의 모델링이 매 사이클 마다 하나 이상의 트랜잭션은 처리 될 수 없는 시스템이기 때문에 8 개 이상의 채널 개수에서 보여주는 트랜잭션 처리량은 모델링 한 시스템의 최고 성능에 근접한 결과로 볼 수 있다.

#### 5. 결론

본 논문은 인메모리 컴퓨팅 환경에서 최적의 메모리 시스템 구조를 알아보기 위하여 메모리 시스템을 모델링한 후, 시뮬레이션하여 메모리의 크기와 채널의 개수에 따른 트랜잭션 처리량을 분석하였다. 시뮬레이션 결과, 메모리 채널의 수에 따른 트랜잭션의 처리량은 최대 5 배까지 증가하였으며, 채널의 수가 메모리 시스템의 성능에 큰 영향을 준다는 것을 확인 하였다. 그리고 채널 개수가 8 개인 경우에 병렬화를 통한 성능향상이 최대로 이루어졌으나, 채널이 4 개인 경우와 채널이 8 개인 경우의 트랜잭션 처리량 차이 고려한다면 채널의 개수가 4 개인 경우가 하드웨어 비용 대비 성능이 가장 우수함을 확인 하였다.

#### 참고문헌

- [1] H. Plattner and A. Zeier, 'In-Memory Data Management: An Inflection Point for Enterprise Applications', Springer, Berlin Heidelberg, 2011.
- [2] A. Siberschatz and P. B. Galvin, 'Operating System Concepts', 4th ED., Addison-Wesley Longman Publishing Co., Inc., Boston, MA, 1993.

- [3] R. Paul, C. Elliott and J. Bruce, "DRAMSim2: A Cycle Accurate Memory System Simulator," Computer Architecture Letters, vol.10, no.1, pp.16-19, 2011.
- [4] Z. Zhang, Z. Zhu and X. Zhang, "A permutation-based page interleaving scheme to reduce row-buffer conflicts and exploit data locality," In Proceedings of 33rd Annual IEEE/ACM International Symposium on Microarchitecture, pp.32-41, 2000.
- [5] V. Stankovic and N. Milenkovic, "Access Latency Reduction in Contemporary DRAM Memories," Facta University, Series: Electronics and Energetics, vol.17, No.1, pp.81-97, 2004.
- [6] S. kshittij, C. Niladris, N. David, A. Manu and B. Rajeev, "micro-pages:increasing DRAM efficiency with locality-aware data placement," In Proceedings of the fifteenth International Conference on Architectural Support for Programming Languages and Operating Systems, pp.219-230, 2010.
- [7] S. Jun and B. T. Davis, "The bit-reversal SDRAM address mapping," In Proceedings of Workshop Software Compilers Embedded Systems, pp.62-71, 2005.
- [8] Micron 1Gb: x4, x8, x16 DDR3 SDRAM Features, 2006. <http://www.micron.com/parts/dram/ddr3-sdram/mt41j256m4jp-125 MT41J256M4JP-125>.