韓國컴퓨터情報學會 論文誌
第16卷 第2號, 2011. 2.

2011-16-2-1-1

명령어 플래시 메모리를 위한 고성능 이중 버퍼 시스템 설계

정 보 성*, 이 정 훈**

The Instruction Flash memory system with the high performance dual buffer system

Bo-Sung Jung*, Jung-Hoon Lee**

요 약

NAND형 플래시 메모리는 저전력, 저렴한 가격, 그리고 대용량 저장매체로 하드디스크 대용을 위하여 많은 연구가 이루어지고 있다. 특히 기존의 캐쉬 구조인 버퍼 시스템을 이용한 플래시 메모리의 성능향상 연구가 이루어지지만 대부분이 데이터 관련 연구이다. 따라서 본 연구에서는 기존의 캐쉬 구조의 버퍼를 이용한 고성능 명령어 플래시 메모리를 구현하였다. 제안된 명령어 플래시 메모리 시스템은 분기 명령어를 위한 시간적 버퍼(victim buffer), 명령어의 대표적인 특징인 순차적 인출을 위한 공간적 버퍼(spatial buffer)로 이루어져 있다. 즉, 제안된 명령어 플래시 메모리의 공간적 버퍼는 큰 페칭 크기를 가지므로 명령어의 순차적 인출에 효과적이며, 작은 페칭 크기를 가지는 시간적 버퍼는 공간적 버퍼에 참조된 명령어를 저장하게 되므로 다시 참조를 위한 분기 명령어에 효과적이다. 시뮬레이션 결과 평균 접근 실패율의 경우 미디어 응용군에 대해 4배 크기의 2-웨이 버퍼, 희생 버퍼, 그리고 2배 크기의 완전연관 버퍼에 비해 평균 77% 감소 효과를 얻을 수 있었다.

▸ Keyword : 낸드 플래시 메모리, 캐쉬 메모리, 시간적-공간적 지역성, 명령어

Abstract

NAND type Flash memory has performing much researches for a hard disk substitution due to its low power consumption, cheap prices and a large storage. Especially, the NAND type flash memory is using general buffer systems of a cache memory for improving overall system performance, but this has shown a tendency to emphasize in terms of data. So, our research is to design a high performance instruction NAND type flash memory structure by using a buffer system. The proposed buffer system in a NAND flash memory consists of two parts, i.e., a fully associative temporal buffer for branch instruction and a fully associative spatial buffer for spatial locality. The spatial buffer with a large fetching size turns out to be effective serial instructions, and the temporal buffer with a small fetching size can achieve effective branch instructions. According to the simulation results, we can reduce average miss ratios by around 77% and the average memory access time can achieve a similar performance compared with the 2-way , victim and fully associative buffer with two or four sizes.

▸ Keyword : NAND flash memory, cache memory, temporal-spatial locality, Instruction

• 제1저자 : 정보성 교신저자 : 이정훈
• 투고일 : 2010. 06. 08, 심사일 : 2010. 08. 12. 게재확정일 : 2010. 11. 12.
* 국립경상대학교 전기전자공학부 박사과정(GyeongSang National University)
** 국립경상대학교 전기전자공학부(공학연구원) 부교수(GyeongSang National University)
※ 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국연구재단의지원을 받아 수행된 기초연구사업임(2009-0065096)

www.kci.go.kr

Ⅰ. 서 론

오늘날 스마트 폰, 포터블 컴퓨터 및 PDA등과 같은 다양한 형태의 정보화/미디어 기기의 사용이 급속히 확대되고 있으며, 더욱이 오늘날 포터블 시스템은 대용량의 정보 처리 저장을 위한 고성능의 요구로 인해 기존의 하위 저장매체를 대신할 새로운 형태의 대용량 저장장치에 대한 필요성의 증대로 이루어지고 있다[1]. 플래시 메모리는 저전력, 강한 충격저항, 높은 집적도 그리고 대용량 저장매체로써 중요성 및 수용에 대한 요구를 증대시키고 있다[2]. 또한 오늘날 하드디스크를 대치하기 위한 플래시 메모리 연구인 SSD(solid-state disk) 연구가 활발히 진행하고 있다. 이러한 SSD의 핵심 모듈은 플래시 메모리가 차지할 것이다[3,4].

플래시 메모리의 쓰기 연산 및 지우기 연산을 줄이는 것은 플래시 메모리의 수명 뿐 아니라 성능향상을 위하여 매우 중요한 요인으로 인하여 현재 많은 연구가 이루어지고 있다. 그 중 기존의 플래시 메모리에 캐쉬 메모리 구조의 버퍼를 이용한 연구[5,6,7]가 시도되어지고 있다. 하지만 대부분의 버퍼를 이용한 NAND형 플래시 메모리의 연구는 상대적으로 비싼 큰 용량의 SRAM 버퍼를 사용하거나 데이터의 쓰기 연산 및 지우기 연산의 개선으로 인한 성능향상 연구들이며, 명령어에 대한 연구가 미비한 상태이다.

일반적으로 NOR 플래시 메모리는 주로 실행을 위한 코드를 저장하는데 사용되고, NAND 플래시 메모리는 주로 데이터를 저장하는데 사용된다. NAND형과 NOR형의 플래시 메모리 읽기 성능은 초기접근시간(initial access time)에서 가장 큰 차이를 보인다. NAND 플래시 메모리의 초기접근시간은 25μs정도이며 NOR 플래시 메모리는 읽기 모드에 따라 50ns ~ 100ns로 약 250배 ~ 500배 정도가 차이가 난다. 이러한 특징은 NAND형과 NOR형 모두 비트를 저장하기 위한 셀의 구조는 동일하지만 셀의 집합과 접근을 어떻게 구성하느냐에 따라 차이를 보이고 있다. NAND형 플래시 메모리는 저장된 비트를 위해 순차적인 페이지 단위의 접근에 비해 NOR형 플래시 메모리는 바이트 단위의 병렬접근을 이룬다.

NAND형 플래시 메모리는 NOR형 플래시 메모리에 비해 작은 블록 사이즈로 빠른 쓰기/지우기 동작이 가능하며, 대용량화에 용이한 장점을 보이고 있다. 하지만 NOR형 플래시 메모리에 비해 NAND형 플래시 메모리는 느린 무작위 접근시간을 보이며, 특히 바이트 단위 접근의 NOR형 플래시 메모리에 비해 NAND형 플래시 메모리는 페이지 단위의 쓰기/읽기 동작과 블록 단위의 지우기 동작이라는 단점을 보이고 있다. 그러나 비용적인 측면에서 NAND 플래시 메모리는 NOR 플래시 메모리에 비해 40%정도 저렴한 장점을 보이며, 이에 NAND 플래시에 SRAM/SDRAM을 이용하여 NOR형의 빠른 접근시간의 효과를 얻고자 하는 연구가 활발히 연구가 진행되고 있다[8].

명령어는 프로그램 수행시 공간적 지역성이 강한 특성으로 순차적으로 인출이 일어난다. 기존의 플래시 메모리는 공간적 지역성을 위해 플래시 메모리 단위인 페이지 크기의 읽기/쓰기 레지스터를 가진다. 이는 공간적 지역성에 강한 특성을 가지는 명령어에서 좋은 성능향상을 가져 올 수 있다. 하지만 기존의 명령어 NAND형 플래시 메모리들은 큰 용량의 버퍼와 페칭 크기를 이용하여 공간적 지역성에 좋은 성능을 보이지만, 비용 면에서 단점을 보이고 있다. 명령어는 순차적 인출에 대한 공간적 지역성이 강한 특성을 보이지만, 분기 명령어, 특히 후방분기 명령어에서는 오히려 이러한 공간적 지역성이 전체 성능저하의 원인이 된다. 명령어는 데이터와 달리 읽기 동작으로 구동되므로 공간적 지역성만 높일 경우 분기 명령에 대해 빈번한 플래시 메모리 셀의 접근으로 전체 시스템의 성능저하를 가져 오기 때문이다.

이 논문에서, 우리는 작은 용량의 버퍼 시스템으로 프로그램 수행시 적합한 명령어 특성에 맞는 공간적 지역성과 분기 명령어에 효과적인 시간적 지역성을 이용하여 NAND형 플래시 메모리 셀의 읽기 동작을 줄이므로 NOR형 플래시를 대치할 고성능 명령어 NAND형 플래시 메모리를 설계하였다.

이 논문의 구성은 다음과 같다. Ⅱ장에서는 플래시 메모리 버퍼 시스템의 관련연구에 대해 설명한다. Ⅲ장에서는 제안된 플래시 메모리의 구조와 제안된 알고리즘 및 동작에 대해 설명한다. Ⅳ장에서는 시뮬레이션 결과 및 평가 그리고 Ⅴ장에서 결론을 맺는다.

Ⅱ. 관련 연구

기존의 플래시 메모리 연구는 대부분이 데이터 관련 연구이며, 또한 NAND형 플래시 메모리의 셀의 데이터 관련 매핑 알고리즘인 FTL 혹은 비교적 많은 시간이 소요되는 쓰기 및 지우기 연산을 줄여 성능 향상을 가져오는 연구들이다. 더욱이 플래시 메모리와 캐쉬 구조의 버퍼 시스템을 이용한 NAND 플래시 메모리의 구조적 연구는 미비한 상태이다.

공간적 스마트 버퍼 시스템[6]와 Huang[9]은 기존의 플래시 메모리에 버퍼를 이용하여 플래시 메모리의 가장 큰 문제점인 쓰기 연산과 지우기 연산을 줄이므로 성능 향상을 이루었다. 스마트 버퍼는 프로그램 수행 시 적합한 두 지역성을

www.kci.go.kr

이용한 플래시 메모리로부터 큰 블록을 가지는 공간적 버퍼와 캐쉬 메모리로부터 버려지는 작은 블록을 가지는 시간적 버퍼로 구성되었다. 캐쉬 메모리로부터 버려진 데이터를 시간적 버퍼에 저장하므로 플래시 메모리의 빈번한 쓰기 연산을 줄이며 빠른 메모리 접근 시간을 보장 할 수 있다. 반면 Huang은 하나의 쓰기 버퍼와 2개의 읽기 버퍼를 이용하여 플래시 메모리의 쓰기 연산과 지우기 연산을 줄이므로 성능 향상을 가져왔다. 쓰기 연산의 데이터 요청시 쓰기 버퍼에, 읽기 연산을 읽기 버퍼에 저장하게 된다. 특히 접근은 쓰기 버퍼 접근 후 읽기 버퍼의 접근으로 이루어진다.

Jo[10]는 휴대용 미디어 장치를 위한 FAB(flash-awar-buffer) 정책을 제안하였다. FAB는 버퍼에서 플래시 메모리의 블록들 중 가장 많은 페이지를 가지는 버퍼 블록들을 희생 블록으로 선택한다. 이러한 버퍼의 희생 블록 선택은 한 번의 플래시 메모리 접근으로 많은 수의 쓰기 연산이나 지우기 연산을 줄여 성능 향상을 보였다.

Park[7]은 기존의 명령어를 담당하던 NOR 플래시 메모리를 XIP(Excuse In Place)기반의 SRAM과 NAND 플래시 메모리로 대치하므로 저비용의 플래시 메모리 패키지를 구현하였다. 기존의 플래시 메모리, 희생 캐쉬(victim cache) 구조의 버퍼 그리고 DRAM으로 구성된다. 플래시 메모리 셀의 참조 패턴에 따라 희생 버퍼와 DRAM의 저장 공간을 선택하게 된다. 참조가 많은 플래시 메모리 셀은 희생 버퍼에 저장 되며, 그렇지 못한 페이지는 DRAM에 저장하게 된다. 참조가 빈번한 페이지를 버퍼에 저장하므로 빠른 메모리 접근과 플래시 메모리의 낮은 소비전력을 이끌었다.

또한 실시간 내장형 시스템에 적합한 요구 페이징(demand paging)을 버퍼를 이용하여 그림자(shadowing) 기법[11] 알고리즘 연구가 이루어졌다. 이 연구는 실시간 내장형 시스템에서 코드 및 데이터를 NOR 플래시의 대체인 SRAM/DRAM과 NAND 플래시 메모리에서의 요구 페이징 기법의 최악의 응답 시간의 분석을 위한 연구이다.

공간적 스마트 버퍼, Huang, 그리고 Jo 등의 연구는 플래시 메모리에서 데이터 관련 연구로 버퍼를 이용하여 플래시 메모리에서 가장 큰 문제점인 플래시 메모리 셀에 쓰기 연산과 지우기 연산을 줄이므로 전체 성능향상을 이루었다. 반면, Pack의 연구와 버퍼를 이용한 그림자 기법 알고리즘의 경우 플래시 메모리에서 명령어에 관련 연구이다. 하지만, Pack의 연구는 상대적으로 면적 대 비용이 높은 큰 용량의 SRAM 사용과 이로 인한 높은 소비전력의 단점을 보이고 있다. 또한 페이지 단위의 저장으로 명령어의 특징인 공간적 지역성에 효과적이지만, 큰 페칭 크기로 인해 페칭 블록의 수가 줄어들므로 한번 참조된 명령어를 다시 참조 되는 하나의 분기 명령어 대해 큰 단점을 보이고 있다.

이와 같이 기존의 NAND형 플래시 메모리에 버퍼의 이용은 성능 향상에 좋은 장점을 보이고 있다. 하지만 데이터와 달리 명령어에 대한 연구는 미비한 수준이다. 더욱이 NAND형 플래시 메모리는 오늘날 내장형 시스템의 저장매체로 사용하기 위하여 연구가 활발하게 이루어지고 있다. 따라서 작은 용량의 저전력을 위한 새로운 구조의 버퍼 시스템이 필요하다.

III. 명령어 낸드 플래시 메모리 버퍼시스템

이 장에서는 연구 개발 동기에 대하여 설명하고 SRAM 버퍼를 내장한 명령어 NAND 플래시 메모리에 대한 구조적 모델 및 공간적 지역성에 효과적인 구조와 메커니즘의 구체적인 동작에 대해서 설명한다.

3.1 고성능 명령어 버퍼시스템의 제안 동기 및 방법

본 연구의 목적은 기존 플래시 메모리에 캐쉬 메모리 구조의 버퍼를 이용하여 고성능 명령어 플래시 메모리 설계이다. 명령어를 위한 새로운 구조의 플래시 메모리 시스템을 설계하기 위하여 기존 플래시 메모리에 다양한 구조의 버퍼 구조와 버퍼 크기 그리고 페칭 크기를 이용하여 최적의 명령어를 위한 NAND형 플래시 메모리 버퍼 구조와 버퍼 크기 그리고 페칭 크기를 선택하였다. 선행연구에서 기존의 플래시 메모리와 캐쉬 메모리 버퍼의 접목에서 최적의 버퍼 구조는 완전연관 버퍼로 16Kbyte의 버퍼 크기와 2Kbyte 페칭 크기에서 가장 좋은 버퍼 접근 실패율을 보였으며, 32byte에서는 가장 좋은 평균 메모리 접근 시간을 보였다. 버퍼 접근 실패율은 프로그램 수행시 순차적 접근에 효과적인 큰 페칭 크기가 효과적이다. 하지만 전체 시스템의 성능평가 지표인 평균 메모리 접근 시간은 플래시 메모리 셀의 접근에 민감하게 반영된다. 특히, 한번 참조된 명령어를 다시 참조하기 위한 분기 명령어는 버퍼에서 참조된 명령어를 오랜 시간 저장 될수록 플래시 메모리 셀의 접근을 줄여 전체 시스템의 성능향상을 이룰 수 있다. 하지만 낮은 버퍼 접근 실패율은 공간적 지역성을 위해 큰 페칭 크기를 가지므로 버퍼 효용성은 높으나, 분기 명령어에 대한 시간적 지역성을 이용하지 못하여 플래시 메모리 셀의 빈번한 접근으로 성능저하의 원인이 된다. 분기 명령어에 대한 시간적 지역성을 위해 작은 페칭 크기의 버퍼를 사용할 경우 버퍼의 효용성이 떨어져 높은 버퍼 접근 실패

www.kci.go.kr

율을 야기 시킨다.

그림 1은 기존 플래시 메모리와 다양한 버퍼 시스템을 가지는 명령어 플래시 메모리에서 미디어 벤치[12]에서의 평균 메모리 접근 시간을 나타낸 그림이다. 각 버퍼 시스템은 직접사상 버퍼(dirtect-mapped buffer), 2-웨이(2-way associative buffer), 희생 버퍼(victim buffer) 그리고 완전연관 버퍼(fully associative buffer)이다. 각 버퍼 시스템의 크기는 4Kbyte의 버퍼 크기와 32byte의 페칭 크기를 가지는 구조이며, 참조되어진 명령어는 바로 버퍼에 저장하게 된다.

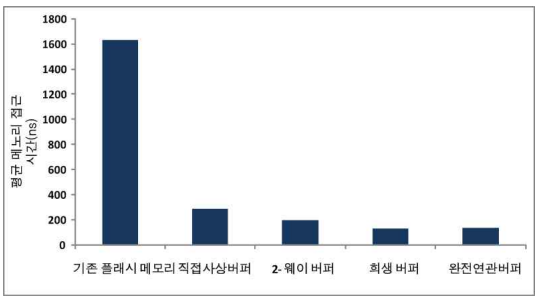

그림 1. 기존 플래시 메모리와 다양한 버퍼시스템의 명령어 플래시 메모리 평균 메모리 접근 시간.
Fig. 1. Average memory access time of the conventional instruction flash with buffer systems

그림 1.에서 보듯이 작은 용량의 버퍼 시스템을 이용할 경우 기존의 플래시 메모리에 비해 좋은 성능향상을 가지는 것을 알 수 있다. 동일한 크기의 버퍼와 페칭 크기에서 완전연관 버퍼가 가장 좋은 성능향상을 보이고 있다. 이는 캐쉬 메모리에서처럼 충돌 접근 실패를 많이 줄이기 때문이다.

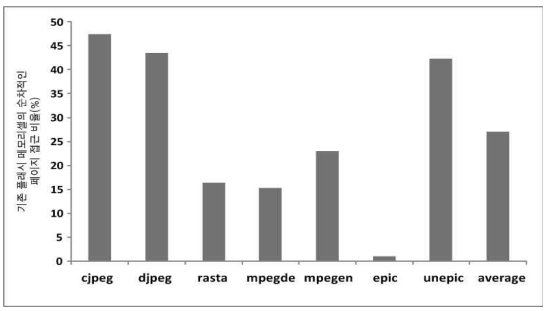

그림 2. 기존 NAND형 플래시 메모리셀의 순차적인 명령어 접근 비율
Fig. 2.Serial access pattern ratios of a conventional NAND flash memory

그림 2는 기존의 NAND형 플래시 메모리에서 순차적인 플래시 메모리 셀의 접근 비율을 나타낸 그림이다. cjpeg, djpeg, unepic는 전체 40%가 넘는 비율이 순차적인 페이지 접근을 하고 있다. 평균적으로 미디어벤치에서 플래시 메모리 셀에 약 25%가 순차적인 접근이 이루어진다. 그림 1에서 버퍼 구조를 가지는 플래시 메모리가 기존의 플래시 메모리 보다 더 좋은 성능을 보이는 이유는 그림 2의 순차적인 플래시 메모리 셀의 페이지 접근이 대부분 분기 명령에 의해 다시 참조되는 명령어가 버퍼에 저장되기 때문으로 추정 할 수 있다.

특히, 한 페이지를 초과하는 분기 명령어에서는 이러한 버퍼 시스템은 더욱 효과적이라고 할 수 있다. 그림. 1에서 충돌 실패(conflict miss)에 가장 효과적인 구조인 완전연관 버퍼에서 좋은 성능을 보이고 있다. 하지만 그림. 2와 그림. 3에서 알 수 있듯이 접근 실패율과 평균 메모리 접근 시간은 페칭 크기에서 서로 상반된 결과를 보여주고 있다.

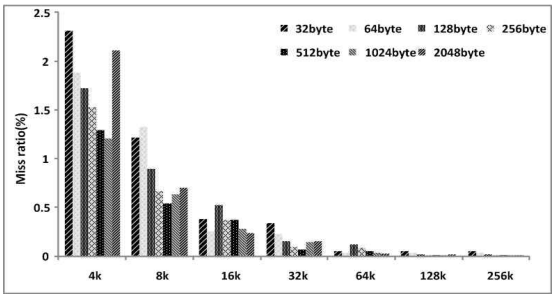

그림 3. 다양한 버퍼 크기와 페칭 크기에서 완전연관 버퍼 접근 실패율
Fig. 3.Fully associative buffer miss ratios with various buffer sizes and fetching sizes

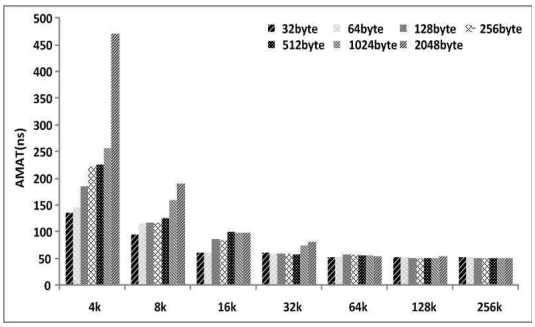

그림 4 다양한 버퍼 크기와 페칭 크기에서 완전연관 버퍼 평균 메모리 접근 시간
Fig. 4.Fully associative buffer average memory access time with various buffer and fetching sizes

그림 3은 완전연관 버퍼에 대하여 버퍼 접근 실패율을 나타낸 그림이며, 그림 4는 완전연관 버퍼의 평균 메모리 접근 시간을 나타낸 그림이다. 완전연관 버퍼는 그림 1에서 가장 좋은 성능을 보이는 버퍼 구조이다. 그림 3과 그림 4는 완전연관 버퍼에서 버퍼의 크기와 페칭 크기를 증가 시켰을 때의 각각의 성능을 변화를 나타내고 있다.

www.kci.go.kr

버퍼의 크기는 4Kbyte에서 256Kbyte까지 증가 시켰으며, 페칭 크기는 32byte에서 2048byte인 한 페이지 크기까지 증가 시켜서 성능을 평가하였다. 그림 3에서 보듯이 페칭 크기와 버퍼 크기가 증가 할수록(2048byte 페칭 크기 제외) 낮은 접근 실패율을 보이고 있다. 그림 3에서 작은 용량에서 2048byte의 페칭 크기가 높은 접근 실패율을 보이는 것은 순차적인 명령어 인출 보다는 분기 명령에서 불리한 조건이기 때문이라고 판단 할 수 있다. 하지만, 그림 4의 경우 작은 용량에서 오히려 작은 페칭 크기에서 더 좋은 성능을 보이고 있다. 이는 작은 용량의 버퍼 크기에서 작은 페칭 크기를 가지는 버퍼는 그림 1과 그림 2에서 보았듯이 분기 명령에 대한 시간적 지역성을 효과적으로 사용할 수 있기 때문이다.

완전연관 버퍼뿐 아니라 다른 직접사상 버퍼, 집합연관 버퍼 그리고 희생 버퍼에서도 버퍼 크기와 페칭 크기 변화에서 비슷한 성능 변화를 보이고 있다.

하지만 작은 용량에서 높은 접근 실패율을 가지므로 오히려 플래시 메모리에 빈번한 접근을 발생하게 된다. 이러한 접근이 순차적인 접근(읽기/쓰기 레지스터 접근)성공 일지라도 전체 플래시 메모리의 구동을 요구하므로 많은 소비전력을 요구하게 될 것이다. 따라서 작은 용량의 버퍼에서 낮은 접근 실패율과 평균 메모리 접근 시간을 위한 명령어를 위한 새로운 고성능의 버퍼 구조가 필요하다.

본 논문에서는 명령어를 위한 버퍼 시스템으로 명령어의 순차적인 인출에 강한 큰 페칭 크기를 가지는 공간적 버퍼와 한번 참조된 명령어를 다시 참조하기 위한 분기 명령어에 효과적인 작은 페칭 크기를 가지는 이중 구조의 명령어 버퍼 시스템을 제안한다.

3.2 구조적 특징과 동작모델

그림 5는 본 연구에서 제안된 새로운 명령어 플래시 메모리구조이다. 제안된 명령어 플래시 메모리는 크게 3부분으로 나누어진다. 플래시 메모리 셀과 2KB 읽기/쓰기 레지스터를 가진 기존의 NAND 플래시 메모리와 명령어의 순차적인 인출에 효과적인 공간적 버퍼, 그리고 분기 명령어에서 효과적인 시간적 버퍼로 나누어진다. NAND 플래시 메모리는 일반 상용 플래시와 동일한 구성이며, 대용량 추세에 따라 한 페이지가 2Kbyte로 가정하였다. 명령어 플래시 메모리를 위한 버퍼 시스템의 이중 버퍼 구조는 완전연관 버퍼로 구성된 공간적 버퍼와 시간적 버퍼로 나누어진다. 공간적 버퍼는 명령어의 기본 특성인 공간적 지역성을 효과적으로 활용하기 위해 큰 페칭 크기를 가지며, 시간적 버퍼는 공간적 버퍼에 참조된 페칭 크기를 저장하게 된다. 이러한 시간적 버퍼는 한번 참조된 명령어를 다시 참조하기 위한 분기 명령어 이후 순차적인 명령어 접근에서 효과적인 성능 향상을 가져다 줄 수 있다.

공간적 버퍼의 한 페칭 크기는 시간적 버퍼의 페칭 크기의 n배로 구성된다. 만약 시간적 버퍼의 페칭 크기가 8byte로 구성된다면 공간적 버퍼의 한 페칭 엔트리의 크기는 n*8byte가 된다. 또한 공간적 버퍼에는 시간적 버퍼에 명령어를 저장하기 위한 n개의 접근 성공비트(hit_bit)를 가지고 있다. 즉, 공간적 지역성의 한 페칭 크기가 512byte 그리고 시간적 버퍼의 페칭 크기는 명령어 크기로 8byte라면, 공간적 버퍼의 한 페칭 엔트리에는 64개의 시간적 버퍼를 위한 접근 성공 비트를 가지게 된다. 공간적 버퍼에 접근 성공이 이뤄지면 n개의 접근 비트 중 하나를 골라 접근 비트의 갱신이 이루어진다. 시간적 접근 성공이 이뤄지면 기존의 버퍼 접근과 같이 동일하게 접근 성공만 일어난다. 만약 버퍼 접근 실패시, 읽기/쓰기 레지스터 혹은 플래시 메모리 셀로부터 공간적 버퍼에 저장하게 된다. 만약 공간적 버퍼의 모든 엔트리가 유효하다면 FIFO동작으로 가장 먼저 들어온 공간적 버퍼의 엔트리 중 참조가 발생한 m개의 작은 페칭 크기를 시간적 버퍼에 저장하게 된다.

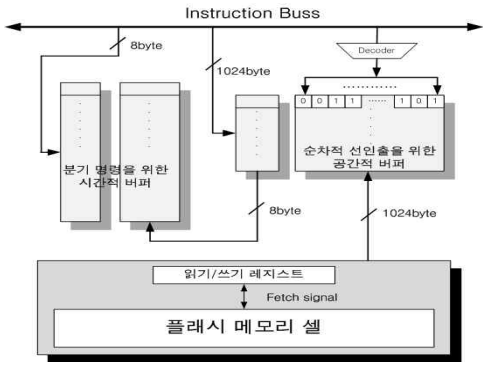

그림 5. 제안된 명령어 플래시 메모리 이중 버퍼 시스템 구조
Fig. 5.The proposed instruction flash memory with dual buffer system

3.2.1 공간적 버퍼 혹은 희생 버퍼에서 접근 성공일 경우

CPU가 메모리 접근을 수행할 때 접근 실패가 일어난다면, 제안된 명령어 플래시 메모리 시스템에서 이중 버퍼인 공간적 버퍼와 시간적 버퍼의 접근은 동시 접근이 이루어진다. 만약 공간적 버퍼에서 접근 성공이 발생하면, 공간적 버퍼의 참조된 페칭 크기내의 요청된 명령어의 n개 중 하나의 접근 성공 비트를 갱신하게 되다. 시간적 버퍼의 접근 성공이 발생하면, 기존의 하위계층의 접근 성공과 동일한 동작을 하게 된다.

3.2.2 공간적 버퍼 혹은 희생 버퍼에서 접근 실패일 경우

www.kci.go.kr

공간적 버퍼와 시간적 버퍼 모두 접근 실패가 발생할 경우, 플래시 메모리의 2KB 읽기/쓰기 레지스터나 플래시 메모리 셀로부터 요청된 공간적 버퍼의 페칭 크기인 512byte를 가지고 온다. 이때 플래시 메모리로부터 이동된 공간적 버퍼의 페칭 크기내의 참조된 명령어의 접근 성공비트를 갱신하게 된다.

공간적 버퍼에 무효 엔트리가 존재한다면, 페칭 크기를 무효 엔트리에 저장하며 접근 성공된 작은 페칭 크기(8byte)에 접근 성공비트를 갱신하게 된다. 만약, 공간적 버퍼의 모든 엔트리가 모두 유효한 엔트리라면 FIFO 동작으로 공간적 버퍼의 명령어를 시간적 버퍼의 명령어로 이동하게 된다. 이때 이동 명령어의 선택은 공간적 버퍼에서 접근 성공이 발생한 작은 페칭 만을 이동하게 된다. 즉, 성공비트가 1인 작은 페칭만을 시간적 버퍼에 저장하게 된다.

IV. 시뮬레이션을 통한 성능평가

이 장에서는 시뮬레이션 환경 및 성능 평가에 대해 자세히 설명한다. 사용된 벤치마크는 멀티미디어 응용을 대표하는 미디어벤치마크[12]를 이용하여 성능 평가를 하였다. 이 벤치마크들의 트레이스를 생성하기 위해 Simplescalar3.0[13]을 사용하여 각각 1억 개의 명령을 수행하는 동안 데이터 참조 주소를 모니터링 하여 사용하였다. 시뮬레이션을 위한 시스템으로 본 연구에서는 다양한 버퍼 시스템을 가지는 플래시 메모리 자체 성능을 평가하였다. 기본 시뮬레이션 환경 변수는 표 1과 같다.

표 1. 시뮬레이션 변수들
Table 1. Simulation Parameters

System Parameters	Value
Random read time	20us
Serial read time	20ns
Buffer access time	50ns

4.1 제안된 고성능 명령어 버퍼시스템 성능평가

본 연구에서는 메모리 시스템의 대표적인 성능 평가 지표인 접근 실패율(miss ratio)과 평균 메모리 접근 시간(average memory access time)을 제안된 플래시 시스템과 다른 버퍼시스템을 가진 플래시 메모리 시스템을 비교, 측정하기 위해서 사용하였다. 여기서 접근 실패율은 요청된 명령어가 버퍼 시스템 내에서 일치하는 페칭 크기가 없을 때 전체 명령어 요청 중 버퍼 접근 실패율을 의미한다. 또한 평균 메모리 접근 시간 역시 버퍼 시스템의 접근 실패로부터 요청한 데이터가 버퍼 시스템에 업로드 될 때까지 지연된 시간을 의미한다. 플래시 메모리 시스템에서 버퍼 크기는 접근 실패율을 결정하는 중요한 요소이다. 그러므로 이러한 버퍼 크기는 실제 설계 시 중요한 변수로써 작용되어진다. 그러나 버퍼 시스템은 빠른 접근 시간을 위해 SRAM을 사용하므로 큰 용량에서의 버퍼 시스템은 비용적인 측면에서 무시하지 못할 변수로 작용한다. 시뮬레이션에서 제안된 구조의 버퍼 크기는 8byte 페칭 크기를 갖는 4KB 완전연관 버퍼인 시간적 버퍼와 공간적 지역성을 위한 공간적 버퍼는 512byte의 페칭 크기를 갖는 4KB 완전연관 버퍼를 사용하였다.

비교 버퍼 시스템으로 직접사상 버퍼, 2-way 버퍼, 희생 버퍼 그리고 완전연관 버퍼를 비교 버퍼구조로 사용하였다. 직접사상 버퍼를 제외한 모든 버퍼들은 캐쉬 메모리에서 충돌 접근 실패에 강한 버퍼 구조로써 낮은 메모리 접근 실패율을 보이는 버퍼 구조이다. 각 버퍼의 페칭 크기는 전체 시스템의 성능 평가인 평균 메모리 접근 시간이 가장 좋은 32byte로 선택하였다. 또한 버퍼의 크기는 비용 대 성능향상을 고려하여 선정하였다. 즉 직접사상 버퍼의 경우 버퍼 크기는 64Kbyte(DM_64K), 2-Way 버퍼와 희생 버퍼는 각각 32Kbyte(2W_32K, VT_32K) 그리고 완전연관 버퍼는 16Kbyte(FA_16K)의 버퍼 크기에서 성능 평가를 비교하였다.

그림 6는 제안된 버퍼 시스템과 비교 버퍼 시스템의 버퍼 접근 실패율을 나타낸 그림이다. 전체적으로 제안된 플래시 메모리의 버퍼 시스템이 좋은 성능을 보이고 있다. 제안된 버퍼 시스템은 8배 크기의 직접사상 버퍼에 비해 68%의 낮은 버퍼 접근 실패율을 보이며, 4배 크기의 2-way버퍼와 희생 버퍼에서도 각각 79%, 75%의 낮은 버퍼 접근 실패율을 보이고 있다. 특히 충돌 접근 실패에 강한 2배 크기의 완전연관 버퍼에 비해 80%의 낮은 접근 실패율을 가진다. 특히 'rasta'에서 제안된 플래시 메모리의 버퍼구조가 다른 버퍼구조에 비해 가장 좋은 성능 향상을 보이고 있다. 'rasta'의 경우 그림 2에서 보듯이 연속적인 페이지 접근이 15%를 차지하고 있다. 다소 낮은 순차적 접근을 보이면서도 제안된 버퍼가 다른 버퍼구조에 비해 'rasta'에서 좋은 성능을 보이는 것은 분기 명령어를 위한 많은 엔트리를 보유하는 시간적 버퍼에 의한 것으로 추정할 수 있다. 이는 완전연관 버퍼에 비해 많은 엔트리를 보유한 직접사상 버퍼, 2-Way 버퍼 그리고 희생 버퍼에서도 완전연관 버퍼에 비해 좋은 성능을 보이기 때문이다. 즉 'rasta'의 경우 많은 버퍼 엔트리에 접근 성공한 명령어를 오랫동안 저장하므로 좋은 성능 효과를 볼 수 있다. 제안된 플래시 메모리에서 참조된 명령어를 저장하는 시간적 버퍼에서 'rasta' 경우 많은 접근 성공이 발생하였다. 즉,

www.kci.go.kr

'rasta'의 경우, 일정한 페칭 크기로 명령어 접근 성공이 발생하며, 그 접근 성공된 명령어는 일정 시간뒤 다시 접근이 일어나는 빈도가 높다고 할 수 있다. 또한 djpeg나 unepic의 경우 역시 그림 2에서 많은 비율의 순차적인 접근을 보이고 있다. 역시 'djpeg'와 'unepic'에서도 제안된 버퍼 구조가 다른 버퍼구조에 비해 낮은 접근 실패율을 보이고 있다. 하지만 높은 순차적 접근을 보이는 'cjpeg'의 경우 제안된 버퍼구조나 비교 버퍼구조에서 다른 벤치마크에 비해 큰 변화가 없다. 이는 'cjpeg'의 순차적인 플래시 메모리 접근은 대부분 다른 벤치마크에 비해 비교적 짧은 거리의 분기 명령으로 추정 할 수 있다.

실제 djpeg와 unepic의 경우 다른 비교 버퍼에서 짧은 거리의 분기 명령은 모두 접근 성공이 가능하다. 하지만 비교적 먼 거리의 분기 명령어에 대해 접근 성공을 보장 할 수 없다. 기본적으로 비교 버퍼는 32byte의 페칭 크기로 명령어에서 볼 때, 접근이 발생하지 않는 명령어까지 가질 수 있으므로 효과적인 버퍼의 엔트리를 사용할 수 없기 때문이다.

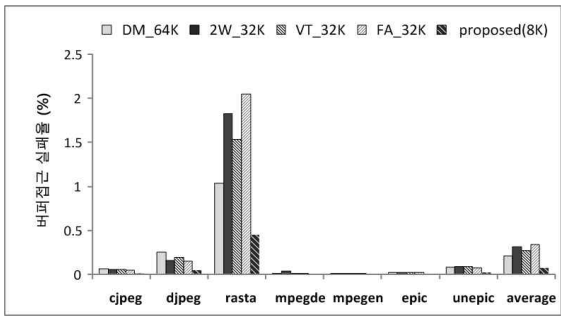

그림 6. 각 버퍼 시스템의 버퍼 접근 실패율
Fig. 6. Miss ratio of various buffer systems

하지만 제안된 플래시 메모리에서 버퍼 구조는 8byte의 페칭 크기의 시간적 버퍼로 구성되어 있다. 따라서 공간적 버퍼에서 접근이 발생한 명령어를 엔트리의 손실 없이 사용할 수 있기 때문에 다른 비교 버퍼 접근 실패율보다 좋은 성능향상을 이룰 수 있었다.

그림 7은 제안된 버퍼 시스템과 비교 버퍼 시스템들의 평균 메모리 접근 시간을 나타 낸 그림이다. 2-Way 버퍼를 제외하고는 모든 버퍼 구조에서 비슷한 성능을 보이고 있다. 하지만 버퍼 크기 면에서 제안된 버퍼의 크기는 전체 8Kbyte임에 비해 직접사상 버퍼는 8배, 2_Way 버퍼와 희생버퍼는 4배 그리고 완전연관 버퍼는 2배의 크기를 보이고 있다. 따라서 제안된 버퍼구조가 다른 버퍼구조에 비해 좋은 성능을 가짐을 알 수 있다.

실제, 제안된 버퍼구조가 그림 6에서 버퍼 접근 실패율이 가장 좋은 성능을 보임에도 그림 7에서 보듯이 평균 메모리 접근 시간이 비슷하게 나타나는 이유는 다른 버퍼구조에서 순차적인 접근 성공(읽기/쓰기 레지스터 접근 성공)이 많이 발생하기 때문이다. 특히 'rasta'의 경우 제안된 버퍼 구조는 다른 버퍼구조에 비해 버퍼 접근 실패율이 평균 75%의 성능 향상을 보임에도, 평균 메모리 접근 시간은 평균 11%만의 성능 향상을 보이고 있다. 이는 플래시 메모리 셀 접근(20us)에 비해 순차적인
접근(70ns, 버퍼 접근 시간 + 순차적인 접근 시간)이 매우 짧기 때문이다. 하지만 순차적인 접근은 플래시 메모리의 접근이므로 플래시 메모리의 자체 구동으로 이루어져야 한다. 따라서 소비 전력 측면에서도 제안된 버퍼 구조를 가지는 플래시 메모리가 다른 비교 버퍼를 가지는 플래시 메모리보다 더 효과적인 버퍼 구조라는 것을 예측 할 수 있다.

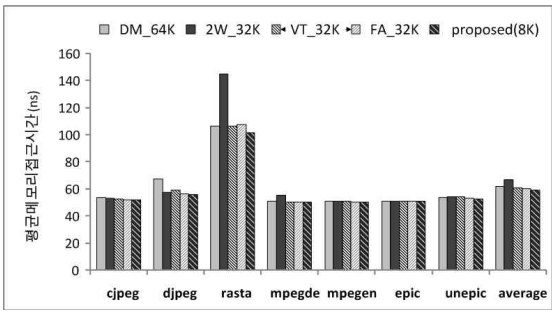

그림 7. 버퍼 시스템의 평균 메모리 접근 시간
Fig. 7. Average memory access time of various buffer systems

V. 결론

본 연구에서는 명령어에 효과적인 작은 용량의 새로운 플래시 메모리 버퍼 시스템의 설계를 목적으로 하였다. 기존의 캐쉬 메모리 구조를 플래시 메모리에 접목시켜 명령어를 위한 고성능 버퍼 구조와 버퍼 크기 및 페칭 크기를 선택하였다.

제안된 플래시 메모리는 이중 버퍼 구조로써, 명령어의 순차적인 인출에 적합한 공간적 지역성에 용이한 큰 페칭 크기를 가지는 공간적 버퍼와 참조된 명령어를 다시 참조하기 위한 분기 명령어에 적합한 작은 페칭 크기를 가지는 시간적 버퍼로 구성된다. 시간적 버퍼의 명령어는 공간적 버퍼에서 시간적 버퍼의 페칭 크기의 접근을 선택하므로 이루어진다. 즉, 공간적 버퍼의 접근 성공에서 시간적 버퍼의 페칭 크기를 참조 성공 비트를 이용하여 참조된 작은 페칭 크기만을 시간적 버퍼에 저장하게 된다. 공간적 버퍼는 512byte의 큰 페칭 크

www.kci.go.kr

기로 버퍼에서 낮은 실패 접근율을 보장하며, 시간적 버퍼는 8byte의 페칭 크기로 이루어진다. 시간적 버퍼는 작은 페칭 크기로 많은 엔트리 수를 가지므로 비교적 먼 거리의 분기 명령까지 저장 할 수 있기 때문이다.

결과적으로 제안된 8Kbyte의 버퍼를 가지는 플래시 메모리는 8배의 크기를 가지는 직접사상 버퍼에 비해 버퍼 접근 실패율을 68%, 4배의 크기를 가지는 2-Way 버퍼, 희생 버퍼 그리고 완전연관버퍼에서는 각각 79%, 75%, 79%의 성능 향상을 보이고 있다. 이는 소비전력이 측면에서 플래시 메모리 접근을 줄임으로써 매우 효과적이라 할 수 있다. 또한 평균 메모리 접근 시간에서는 8배 크기의 직접사상 버퍼, 4배 크기의 희생버퍼와 완전연관 버퍼 그리고 2-웨이 버퍼에서 비슷한 성능을 보이고 있으며 이는 성능 면에서도 작은 버퍼의 사용으로 고성능을 보장하는 구조라 할 수 있다.

참고문헌

[1] N. Ballard, "State of PDAs and Other Pen-Based Systems," In Pen Computing Magazine, pp. 14-19, Aug. 1994.

[2] B. March, F. Douglis, and P. Krishnan, "Flash Memory File Caching for Mobile Computer," In Proc. of the 27 Hawaii International conference on System Science, pp. 451-460, Jan. 1994.

[3] S. Park, J. Park, and S. Kim, "Short-Random Request Absorbing Structure with Volatile DRAM Buffer and Nonvolatile NAND Flash Memory," In Proc. of the CEA09, pp.257-262, Jan. 2009.

[4] Computerworld, "Solid-State Disk (SSD) - The future in hard drives," *http://whatsonmypc.wordpress.com /2009/01 /17/ssd/*

[5] J. Lee, G. Park, and S. Kim, "A New NAND-type Flash Memory Package with Smart Buffer System for Spatial and Temporal Localities," Journal of System Architecture, Vol.51 No.2 pp. 111-123, Feb. 2005.

[6] B. Jung, J. Lee, "The Smart buffer Nand Flash memory System for a hard-disk substitution." Journal of The Korea Society of Computer and Information, Vol 14, No 3, pp. 41-50, Mar. 2009.

[7] C. Park, J. Seo, S. Bae, H. Kim, S. Kim and Bumsoo Kim, "A low-cost memory architecture with NAND XIP for mobile embedded systems," In Proc. of the 1st CODES-ISSS'03, pp.138-143, Oct. 2003.

[8] Samsung Elec., "NAND-type Flash Memory," *Http://www.samsungelectronics.com/semiconductor/flash/Falsh.thml*

[9] W. Huang, C. Chen, C. Chen, and C. Chen, " Energy-Efficient Buffer Architecture of Flash Memory," In Proc. of the Multimedia and Ubiquitous Engineering, pp.5 43-546, Apr. 2008.

[10] H. S. Jo, J. U Kang, S. Y Pack, "FAB: Flash-aware buffer management policy for protable media players," Comsumer Electronics, IEEE Transactions on, pp.485-493, Apr. 2006

[11] Y. Lee, S. Lim, "Worst Case Response Time Analysis for Demand Paging on Flash Memory," Journal of The Korea Society of Computer and Information, Vol 11, No 6, pp. 113-123, Dec. 2006.

[12] Mediabench, *http://euler.slu.edu/~fritts/mediabench/*

[13] D. Burger and T. M. Austin, "The SimpleScalar tool set, version 2.0, Technical Report TR-97- 1342," University of Wisconsin-Madison, 1997.

저 자 소 개

정 보 성

2008년 2월 : 경상대학교 제어계측공학과(석사)

2008년~현재 : 경상대학교 제어계측공학과 박사과정

관심분야 : 마이크로프로세서, 캐쉬 및 플래시 메모리

E-mail : blueking80@gnu.ac.kr

이 정 훈

2004년 2월 : 연세대학교 컴퓨터과학과(박사)

2004년~현재 : 경상대학교 제어계측공학과 부교수

관심분야 : 고성능 컴퓨팅, 내장형 시스템 및 SOC 시스템

E-mail : leejh@gnu.ac.kr

www.kci.go.kr