

논문 2017-12-19

# 지역성을 이용한 하이브리드 메모리 페이지 교체 정책 ( Page Replacement Policy of DRAM&PCM Hybrid Memory Using Two Locality )

정 보 성, 이 정 훈\*

(Bo-Sung Jung, Jung-Hoon Lee)

Abstract : To replace conventional DRAM, many researches have been done on nonvolatile memories. The DRAM&PCM hybrid memory is one of the effective structure because it can utilize an advantage of DRAM and PCM. However, in order to use this characteristics, pages can be replaced frequently between DRAM and PCM. Therefore, PCM still has major problem that has write-limits. Therefore, it needs an effective page management method for exploiting each memory characteristics dynamically and adaptively. So we aim reducing an average access time and write count of PCM by utilizing two locality for an effective page replacement. We proposed a page selection algorithm which is recently requested to write in DRAM and an algorithm which uses two locality in PCM. According to our simulation, the proposed algorithm for the DRAM&PCM hybrid can reduce the PCM write count by around 22% and the average access time by 31% given the same PCM size, compared with CLOCK-DWF algorithm.

Keywords : Hybrid memory, Page management method, Temporal/spatial locality, Memory characteristics

## I. 서 론

오늘날 컴퓨팅 시스템의 주 메모리로 빠른 접근 시간과 낮은 가격의 장점을 가지는 DRAM이 대표적이다. 하지만 집적도의 한계와 더불어 데이터 유지를 위한 높은 에너지 소비를 가지는 단점을 가진다. 더욱이 오늘날 휴대용 스마트 기기 시장의 급속한 성장과 소셜 네트워크의 등장으로 개인용 데이터 저장 요구와 수요가 지속적으로 증가하고 있다 [1]. 따라서 기존의 메인 메모리인 DRAM을 대체할 새로운 메모리가 필요하다.

현재 DRAM을 대체할 차세대 메모리로 비휘발

성 메모리가 주목받고 있다. 이러한 비휘발성 메모리가 DRAM을 대체 할 경우, 기존의 DRAM의 가장 큰 단점인 누설전류 및 집적도의 한계를 극복 할 수 있다. 더욱이, 비휘발성 메모리에 적재된 커널 및 코드 (code)는 전원이 없어도 유지되기 때문에 빠른 컴퓨터 부팅 작업이 이루어 질수 있다. 또한 의도치 않은 전원 차단으로 인한 데이터 손실 역시 피할 수 있다.

이러한 DRAM을 대체할 차세대 비휘발성 메모리로 Phase Change Memory (PCM), Resistive RAM (RRam), 그리고 Spin- Transfer Torque RAM (STT-RAM)과 같은 고집적도 저전력 메모리가 주목받고 있다 [2].

특히 DRAM을 대체할 차세대 메모리로 PCM이 가장 주목 받고 있다. 다른 차세대 비휘발성 메모리와 같이 PCM은 데이터를 위한 리플래시 동작 (Refresh operation)이 필요 없으며, DRAM에 비해 낮은 대기 전력 (Standby power)을 가진다. 또한 DRAM에 비해 약 4배의 높은 직접도를 가지며, DRAM과 같이 바이트-주소 접근이 가능하다. 특히 PCM이 DRAM의 대체 메모리로 주목 받는 이유 중

\*Corresponding Author (leejh@gsnu.ac.kr)

Received: May 12 2017, Revised: May 22 2017,

Accepted: May 25 2017.

B.S. Jung: GyeongSang National University

J.H. Lee: GyeongSang National University (ERI)

※ 이 논문은 2014년도 정부 (교육부)의 재원으로 한국연구재단의 지원을 받아 수행된 기초연구사업 임 (No. NRF-2014R1A1A4A01008504).

하나가 PCM은 DRAM 생산 공정과 거의 유사하게 생산할 수 있기 때문에 다른 비휘발성 메모리에 비해 설비투자자와 개발비용의 관점에서 유리하다 [3].

하지만, PCM이 이러한 장점을 가짐에도 불구하고 컴퓨터 시스템에서 메인 메모리로 DRAM을 대체하기 위해 큰 문제점을 가지고 있다. 먼저, PCM은 낸드 플래시에 비해 긴 수명을 가지고 있지만, 여전히 제한적인 쓰기 동작 횟수 ( $10^7 - 10^8$ )를 가지고 있다. 둘째, PCM은 DRAM에 비해 읽기/쓰기 동작에서 높은 지연시간을 가진다. PCM의 읽기 동작 시간은 DRAM에 비해 약 1~2배정도 느리며, 쓰기 동작은 약 5~10배 느린 접근시간을 가진다 [4].

PCM의 성능향상을 위한 대표적인 방법으로 PCM 셀의 자체 수명 연장하는 쓰기 동작 평준화 기법 (Wear Leveling)과 구조적인 방법인 하이브리드 구조가 대표적이다. Wear Leveling 기법은 PCM에 빈번하게 쓰기 동작을 요청하는 특정 페이지 혹은 블록에 대하여 쓰기 횟수를 제한하므로 PCM의 수명 연장을 도울 수 있다 [5-7]. 하지만, 이 기법은 단지 PCM의 제한된 쓰기 동작에 효과적이지만, 여전히 쓰기 및 읽기 동작에 대한 성능개선은 이루어지지 않는다. 반면, 하이브리드 메모리 구조는 PCM에 빠른 수행 시간을 위해 기존 DRAM 및 SRAM을 추가적으로 사용하는 방법으로, PCM의 단점인 높은 지연시간 및 제한적인 쓰기 횟수를 효과적으로 줄일 수 있다 [8-12]. 하지만, 이러한 하이브리드 메모리 구조에서 PCM과 DRAM의 빈번한 페이지 (4KByte) 교체는 오히려 시스템의 성능 저하를 가져 올 수 있다. 따라서 하이브리드 메모리 구조에서는 효과적인 페이지 교체 정책이 중요하다.

## II. 관련 연구

### 1. DRAM과 PCM의 하이브리드 메모리 구조

차세대 메인 메모리 구조인 하이브리드 메모리 구조는 DRAM을 PCM의 버퍼 용도로 사용하는 방법과 동일한 계층의 메모리로 사용하는 방법으로 그림 1과 같이 크게 두 가지 방법으로 나누어진다.

그림 1 (a)는 기존 DRAM을 PCM의 버퍼 캐시 (buffer cache)로 활용하는 하이브리드 메모리 구조이다. 이러한 버퍼 시스템 구조는 PCM에 빈번하게 접근하는 블록을 DRAM에 저장하므로 전체적인 PCM의 접근과 쓰기 횟수를 줄일 수 있다. 특히 DRAM 버퍼는 다양한 블록 크기를 가질 수 있으므로 이러한 하이브리드 메모리 구조는 프로그램

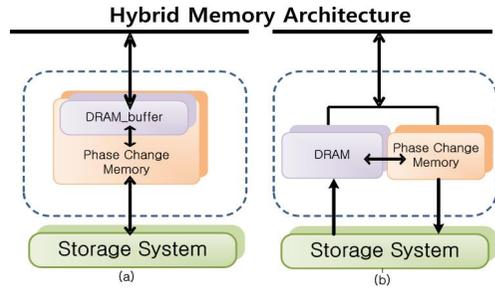


그림 1. DRAM&PCM 하이브리드 구조

Fig. 1 DRAM&PCM Hybrid structure

수행 시, 적합한 두 지역성을 효과적으로 이용할 수 있다. 하지만 이러한 구조는 DRAM 버퍼 관리를 위해 추가적인 하드웨어가 필요하다.

그림 1 (b)는 PCM과 DRAM이 동일한 메인 메모리로 구성된 하이브리드 구조이다. 이러한 구조는 기존의 페이지 관리를 위한 페이지 테이블 (Page Table)을 이용할 수 있다. 특히, DRAM과 PCM이 동일한 메인 메모리로 구동이 될 경우, 최하위 계층에서 요청된 페이지를 DRAM에 저장할 수 있으므로, 참조가 빈번하게 발생하는 특정 페이지를 PCM에 쓰기를 지연 시키므로 PCM의 쓰기 제한 횟수 및 높은 읽기/쓰기 동작 시간을 줄일 수 있다. 하지만, 이러한 구조의 하이브리드 메모리는 PCM에서 빈번한 페이지 쓰기를 줄이기 위해 DRAM과 페이지 교체가 필수적이다. 따라서 PCM과 DRAM의 빈번한 페이지 교체는 오히려 전체 메모리 시스템의 성능 저하와 PCM의 수명 단축의 원인이 된다.

### 2. 하이브리드 메모리 관련 연구

연구 [5-7]은 Wear leveling 기법으로 PCM의 전체 수명을 개선하였지만, 여전히 읽기/쓰기 지연 시간에 대한 성능개선을 이루지 못한다.

연구 [8, 9]는 DRAM을 버퍼로 이용하는 하이브리드 메모리 구조를 제안하였다. 그리고 연구 [10-12]는 DRAM과 PCM이 동일한 메모리 이용하는 하이브리드 메모리 구조를 제안하였다. 위와 같은 기존의 하이브리드 메모리 구조의 연구들은 PCM의 쓰기 횟수 감소 및 성능향상에 효과적인 방법이다. 하지만, 대부분이 빈번한 블록 접근에 대한 프로그램 특성을 고려하지 못하고 있다. 비록 연구 [9]가 빈번하게 접근하는 특정 블록 크기를 고려하고 있지만, 너무 많은 다양한 블록 크기의 버퍼들을 사용하고 있다. 연구 [10]은 쓰기 요청과 읽기 요청 페이지를 구분하여 페이지를 저장하는 반면,

PCM의 쓰기 요청이 발생하면 DRAM과 빈번한 페이지 교체가 발생 하는 문제가 있다. 또한 연구 [11]의 경우, DRAM과 PCM의 빈번한 페이지 교체를 줄이기 위해 지연 시간을 고려하였지만, 이는 단순히 쓰기 요청 횟수만을 고려하고 있다.

### III. 본 론

본 논문의 주목적은 DRAM과 PCM의 하이브리드 구조에 효과적인 페이지 관리를 위한 새로운 알고리즘을 제안함으로써 차세대 고성능 메인 메모리 구현에 그 목적을 두고 있다.

#### 1. 제안 동기 및 방법

DRAM과 PCM이 동일한 메인 메모리 구조로 운용되는 하이브리드 메모리 구조는 기존의 페이지 테이블을 이용하므로 전체적인 하이브리드 메모리의 페이지를 관리에 효과적인 구조이다. 더욱이 DRAM은 PCM과 달리 쓰기 제한이 없기 때문에 최하위 계층으로부터 요청된 페이지를 우선적으로 DRAM에 저장 및 프로그램 수행시 다양한 특성을 판단 할 수 있기 때문에 PCM의 가장 큰 단점인 쓰기 동작 횟수 효과적으로 줄일 수 있다. 따라서 하이브리드 메모리 구조에서 DRAM은 최하위 계층으로부터 요청된 페이지 및 빈번하게 쓰기 동작이 발생하는 페이지를 저장하는 것이 효과적이며, PCM의 읽기 동작 시간이 DRAM과 비슷하기 때문에 읽기 요청이 발생하는 페이지를 저장하는 것이 효과적이다.

따라서 하이브리드 메모리의 DRAM을 효과적으로 운용하기 위해서는 PCM에서 빈번하게 쓰기 동작이 발생하는 페이지를 DRAM의 참조 가능성이 낮은 페이지와 교체가 이루어져야 한다. 하지만, DRAM의 효과적인 운용을 위한 DRAM과 PCM의 빈번한 페이지 교체는 오히려 전체 메모리 시스템의 성능 저하를 가져 올수 있다.

표 1은 DRAM과 PCM의 특성을 보여주고 있다. 표 1.에서 보듯이 64Byte에서 DRAM과 PCM의 읽기 동작 시간은 비슷하지만, 쓰기 동작 시간은 약 PCM이 10배정도 느리다. 기존 메인 메모리의 기본 페이지인 4KByte를 고려시, DRAM과 PCM 하이브리드 메모리의 페이지 교체는 약32us의 지연시간이 필요하다. 이는 하이브리드 메모리구조에서 PCM의 쓰기 요청이 발생할 경우, 바로 DRAM으로 옮기는 동작은 실제 추가적인 지연시간을 요구하게 된다.

표 1. DRAM과 PCM 특성  
Table 1. DRAM& PCM Characteristics

	DRAM	PCM
Cell size	6-10F <sup>2</sup>	4-10F <sup>2</sup>
Access Granularity	64B	64B
Read latency	50ns	50ns
Write latency	50ns	500ns
Read Energy	~0.1nJ/b	~0.1nJ/b
Write Energy	~0.1nJ/b	~0.5nJ/b
Endurance	>10 <sup>15</sup>	10 <sup>8</sup> - 10 <sup>9</sup>
standby Power	Refresh	X

반면, 이러한 지연시간을 줄이기 위해서 PCM에서 DRAM으로 데이터 이동을 지연시키는 것은 PCM의 특정 페이지에 대한 수명을 단축뿐 아니라 데이터 오염을 야기 시킬 수 있다.

기존 캐시 메모리에서 성능향상을 위한 효과적인 방법 중 하나는 프로그램 수행시 적합한 두 지역성 (two locality)을 이용하는 것이다. 본 논문에서 공간 지역성 (spatial locality)은 최근 쓰기 참조가 발생한 데이터의 이웃 데이터가 참조 되어질 확률이 높다는 것을 나타내며, 시간 지역성 (temporal locality)은 최근 쓰기 참조된 데이터가 가까운 시간에 다시 참조되어질 확률이 높다는 것으로 정의하였다. 따라서 공간 지역성이 높은 페이지는 PCM에 쓰기 동작을 수행하는 것이 DRAM과 PCM의 페이지 교체를 줄이므로 전체 적인 지연시간을 줄일 수 있다. 시간 지역성인 높은 페이지는 DRAM에서 관리하는 것이 더 효과적이다.

따라서 본 논문에서는 고성능 하이브리드 메모리를 위한 새로운 페이지 운용 정책을 제안하였다. 제안된 방법은 최하위 계층으로부터 요청된 페이지 및 쓰기 요청의 시간 지역성을 가지는 PCM의 페이지는 DRAM에서 관리하게 되며, PCM은 쓰기 요청의 공간 지역성 및 DRAM으로부터 추출된 페이지들을 관리하게 된다.

#### 2. 제안된 페이지 관리 방법

제안된 하이브리드 메모리에서 DRAM은 추가적인 상태비트를 가진다. Reference 비트 (DRAM\_R, DR)는 DRAM에서 접근 성공 여부를 나타낸다. DR 비트 만약 DRAM의 페이지에 접근 성공이 발생하면, DR비트는 '1'의 값으로 갱신이 일어난다. Write\_hit 비트 (W)는 2비트로 (W[0], W[1]) 구성 되어진다. W[0]비트는 DRAM에 최근 쓰기 요청이

발생한 페이지를 나타내며, W[1]은 이전에 쓰기 요청이 발생한 페이지를 나타낸다. 만약 DRAM이 페이지에 쓰기 요청이 발생하면, 최근 쓰기 접근 성공을 나타내는 W[0]비트만 '1'의 값으로 갱신된다. 마지막으로 Frequence 비트는 DRAM에서 쓰기 요청이 시간적으로 빈번하게 발생한 페이지는 나타낸다. 제안된 알고리즘에서 DRAM의 빈번한 쓰기 요청은 DRAM에 쓰기 접근 성공시 W[0]와 W[1]가 모두 '1'인 상태로 정의하였다. 따라서 DRAM의 페이지에 쓰기 요청이 발생하고, W[0]비트와 W[1]의 비트가 값이 모두 '1'이면, F비트는 '1'로 갱신된다.

만약 하이브리드 메모리에서 접근 실패가 발생하고, DRAM에 무효 페이지가 없다면, DRAM은 각 페이지의 상태 비트를 체크하게 된다. 만약 상태비트들이 모두 '0'이면 그 페이지는 희생 페이지로 선택되며, PCM으로 이동하게 된다. 그리고 새롭게 저장되어진 페이지의 상태비트들의 값은 모두 '0'으로 갱신된다. 반면 상태비트들이 모두 '0'이 아니면, 각 비트 값 들은 다른 상태비트들로 이동하게 된다. 즉, W[1]비트는 W[0]의 값으로 갱신되며, W[0]은 F 비트 값으로 갱신된다. 그리고 F 비트는 '0'으로 갱신이 된다. 이러한 동작은 희생 페이지가 선택 될 때까지 DRAM에서 순환적으로 이루어진다.

제안된 하이브리드 메모리의 PCM은 참조가 발생한 페이지를 나타내는 Reference 비트 (PCM\_R, PR), 공간적 지역성의 쓰기 요청 접근 횟수를 나타내는 Dirty\_Count 비트 (DC) 그리고 페이지 내의 쓰기 요청이 발생한 작은 블록을 나타내는 블록 Dirty 비트 (Block\_Dirty ,BD)들로 구성된다. 만약 제안된 쓰기 요청 블록이 64byte라면 한 페이지에 64개의 DB 비트들을 가지며, DC는 8비트로 구성되어진다.

PR은 PCM에서 접근 성공이 발생하면 '1'로 갱신이 일어난다. 만약 PCM에 쓰기 접근 성공이 발생하고, BD 비트가 '0'이면 BD 비트는 '1'로 갱신이 되며, 이때 DC 비트의 값이 증가하게 된다. 반면, 이미 DB 비트의 값이 '1'이면, 제안된 알고리즘에 의해 그 페이지는 시간적 지역성을 가진다고 판단하고, DRAM의 희생 페이지와 교체가 이루어진다. 이때 DRAM으로부터 PCM에 저장된 페이지의 PR비트에 DR 값이 저장되며, 그 외 나머지 상태비트들은 모두 '0'으로 갱신이 된다. 만약, PCM에 무효 블록이 없다면, PCM은 PR 비트와 DC 비트를 체크하게 된다. 만약 PR 비트가 '1'이면, PR 비트는 '0'으로 갱신이 일어나고 다음 페이지를 체크하게 된다. 반면, PR 비트가 '0'일 경우, DC 비트를 체크

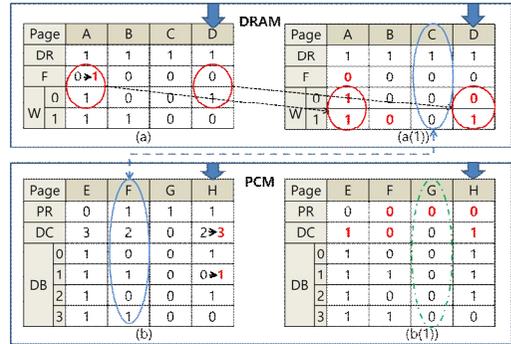


그림 2. 제안된 알고리즘 동작

Fig. 2 The proposed algorithm operation

하며, DC 비트가 '0'이 아니면 DC 비트의 값이 감소하고, 다음 페이지를 체크 하게 된다. 이러한 동작은 어떤 페이지의 PR 비트와 DC 비트가 '0'의 값을 가질 때 까지 PCM에서 순환적으로 일어난다. 따라서 PCM에서 희생 페이지 선택은 PR 비트와 DC 비트가 모두 '0'일 경우 그 페이지를 선택하여 최하위 계층 메모리에 저장하게 된다.

그림 2는 제안된 하이브리드 메모리의 구체적인 동작을 나타내는 그림이다. 그림 2 (a)에서 DRAM의 상태를 나타내고 있으며, 그림 2 (b)는 PCM의 상태를 나타내고 있다. DRAM과 PCM은 4개의 페이지를 가진다고 가정하였으며, PCM의 쓰기 접근 블록은 4개의 블록이 한 페이지를 이룬다고 가정하였다 그리고 모든 페이지는 읽기 혹은 쓰기 요청이 한번 (DR='1', PR='1')은 발생하였다고 가정하였다.

1) DRAM 접근 성공

만약, 그림 2 (a)에서 페이지'A'에 쓰기 접근이 성공한다면, 페이지'A'의 W[0] 비트와 W[1] 비트가 모두 '1'이므로, 페이지'A'는 빈번한 쓰기 요청 페이지로 정의 되어 F 비트가 '1'로 갱신이 된다. 만약 페이지'A'를 제외한 다른 페이지에서 쓰기 접근이 성공한다면, W[0] 비트만 '1'로 갱신이 된다.

2)PCM 접근 성공

만약 PCM에 읽기 요청의 접근 성공이 발생하면, PCM은 단지 PR 비트만을 '1'로 갱신이 된다. 그림 2 (b)에서, 만약 페이지'H'의 DB1에서 쓰기 접근이 성공하면, 페이지'H'의 DB1이 '0'이므로 DB1이 '1'로 갱신되며, DC값이 1증가하게 된다. 반면 페이지'F'의 DB3에 쓰기 접근이 성공하면, DB3이 '1'이기 때문에 시간 지역성 페이지로 정의

되고 DRAM의 희생 페이지와 교체가 일어난다.

2.1) DRAM 희생페이지 선택 : 그림 2 (a)에서 희생 페이지 선택을 위한 체크는 페이지'D'부터 이루어진다. 페이지'D'의 W[0]비트가 '1'이므로 희생 페이지 후보에서 제외되며, 다음 페이지인 페이지'A'의 상태비트들을 체크하게 된다. 이때 페이지'D'의 W[0] 비트는 W[1]에 저장되며, F비트는 W[0]에 저장된다. 페이지'A'역시 상태비트들(F,W[0],W[1])이 모두 '0'이 아니므로, 희생 페이지 후보에서 제외되며, 페이지'B'를 체크하게 된다. 이때 역시 페이지'A'의 상태비트 값들은 갱신이 된다. 페이지'B' 역시 W[1] 비트에 의해 희생 페이지 후보에서 제외되며, 상태 비트 값들이 갱신된다. 페이지'C'는 F,W[0] 그리고 W[1]이 모두 '0'의 값으로 희생 페이지로 선택되어지며, PCM의 페이지'F'와 페이지 교체가 이루어진다. 이때, 페이지'C'가 PCM에 저장 될 때, DR비트 값 역시 PCM에 저장되며, 그 외 상태비트들은 모두 '0'으로 갱신된다. 역시 페이지'F'가 DRAM에 저장될 때, 모든 상태비트들은 '0'으로 갱신이 된다. 그리고 다음 희생 페이지 선택을 위한 시작 페이지는 새롭게 저장된 페이지'F'의 다음 페이지인 페이지'D'가 된다. 그림 2의 a (1)은 DRAM의 희생페이지 선택후 각 페이지들의 상태 비트들을 나타낸 그림이다.

3) 하이브리드 메모리 접근 실패

만약 제안된 하이브리드 메모리에서 접근 실패가 발생하면, 최하위 계층 메모리로부터 새로운 페이지를 DRAM에 저장하게 된다. 그림 2 (a (1))에서 희생 페이지'C'는 PCM에 저장된다. 그림 2 (b)에서 희생 페이지를 선택하게 되며, 페이지'H'부터 상태비트를 체크하게 된다. 페이지'H'의 PR 비트와 DC비트가 모두 '0'이 아니므로 희생 페이지 후보에서 제외된다. 그리고 PR비트는 '0'으로 갱신되며, DC 값은 감소하게 된다. 그리고 다음 페이지인 페이지'E'의 상태비트를 체크하게 된다. 비록 페이지'E'의 PR 비트가 '0'이지만 DC 비트가 '3'으로 희생 페이지 후보에서 제외되며, DC 비트 값은 '2'로 갱신이 일어난다. 다음 페이지인 페이지'F' 역시 PR비트와 DC비트에 의해 희생 페이지에서 제외되며 PR비트와 DC비트가 갱신이 된다. 페이지'G'는 DC비트가 '0'이지만, PR 비트가 '1'이므로 역시 희생 페이지 후보에서 제외되며, PR비트가 '0'으로 갱신이 된다. PCM의 모든 페이지를 검사후 희생 페이지를 선택하지 못했다면, 희생 페이지가 선택 될 때

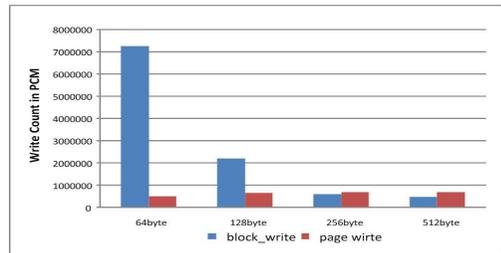


그림 3. PCM의 블록 및 페이지쓰기 횟수  
Fig. 3 Write Count of Block and Page in PCM

까지 모든 페이지의 상태비트들을 체크하게 된다. 본 예에서는 페이지'G'가 PCM의 희생 페이지로 선택되어진다. 그림 2 (b (1))은 PCM의 희생 페이지'G'가 선택되어진 후 상태비트들의 결과를 나타내고 있다.

IV. 성능 평가

본 논문에서는 성능평가를 위하여 Spec CPU 2006벤치마크를 Cachegrind [13]를 수정하여 메모리에 접근하는 1억 개의 데이터의 주소를 사용하였다. DRAM과 PCM의 파라미터는 표. 1과 같다.

성능 평가를 위하여 본 논문에서는 제안된 하이브리드 메모리의 크기는 4Gbit의 PCM을 사용하였다. DRAM은 PCM과 동일한 페이지 크기를 가지는 1Gbit를 사용하였다. 본 논문에서 PCM에 제안된 효과적인 공간 지역성을 위한 쓰기 요청의 블록 크기 선택하기 위해 성능평가를 하였다. 그림 3은 제안된 하이브리드 메모리의 PCM 페이지 내의 블록 쓰기 (block\_write) 횟수 와 페이지 쓰기 (page\_wirtle) 횟수의 평균을 나타낸 그림이다. 여기서 블록 쓰기란 PCM의 한 페이지는 n개의 작은 블록으로 구성되며, 구성된 작은 블록에 한번만 쓰기 요청이 수행된 것을 의미하며, 페이지 쓰기는 DRAM과 PCM의 페이지 교체 및 DRAM으로부터 PCM에 페이지 단위로 쓰기가 수행되는 것을 의미한다. 쓰기 접근 블록은 캐시 메모리 블록 크기인 64byte부터 하드디스크 섹터 크기인 512byte까지 시행 하였다.

그림 3에서 보듯이 제안된 하이브리드 메모리에서 64byte 블록 크기에서 페이지 쓰기는 다른 블록 크기에 비해 평균 25%을 줄인 반면, 블록 쓰기는 256byte와 512byte에 비해 64byte 블록이 약92% 많이 발생 하였으며, 128byte에 비해 70% 더 많이 발생하였다. 이는 제안된 페이지 교체 알고리즘에서 64byte의 블록 크기가 PCM과 DRAM의 페이지 교체 효과적으로 줄이는 크기를 의미한다.

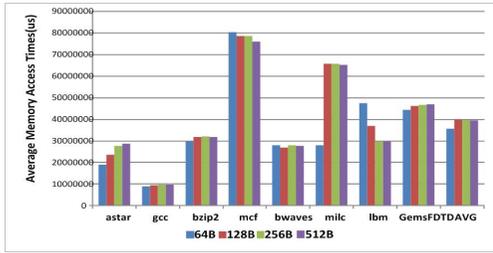


그림 4. 다양한 PCM 블록에 대한 평균 메모리 접근 시간

Fig. 4 Average Memory Access Time by the different block size of PCM

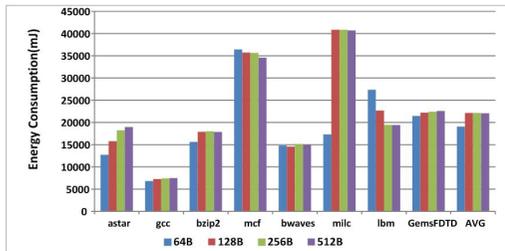


그림 5. 다양한 PCM 블록 크기에 대한 에너지 소비  
Fig. 5 Energy Consumption by the different block size of PCM

본 논문에서는 제안된 페이지 교체 알고리즘의 효과적인 성능평가를 위해 평균 메모리 접근 시간과 에너지 소비를 측정하였다. 하이브리드 메모리의 평균 메모리 접근 시간을 위해 접근 실패시, 최하위 계층 메모리 접근 시간은 톨 [14]을 이용하여 구하였으며, 본 논문에서는 15ms로 정의 하였다. 그림 4는 제안된 하이브리드 메모리의 다양한 PCM 블록에 대한 평균 메모리 접근 시간을 나타낸 그림이다.

그림 5는 PCM의 다양한 블록을 가지는 제안된 하이브리드의 에너지 소비를 나타낸 그림이다.

그림 4와 그림 5에서 보듯이 제안된 하이브리드 메모리 구조에서 mcf와 lbm을 제외한 나머지 벤치마크에서 PCM이 64Byte의 블록 크기 접근이 이루어질 때 다른 블록 크기에 비해 좋은 성능향상을 보이고 있다. 시뮬레이션 결과, 64byte 블록 크기에서 mcf와 lbm은 쓰기 요청에 대해 높은 쓰기 공간 지역성과 더불어 읽기 요청에 대한 높은 접근을 보였다. 따라서 64byte 블록을 가지는 구조에서는 다른 블록 크기에 비해 PCM에서 많은 쓰기 요청 후 DRAM으로 페이지 교체가 발생하여 다른 블록에 비해 높은 쓰기 지연 시간이 발생하였다.

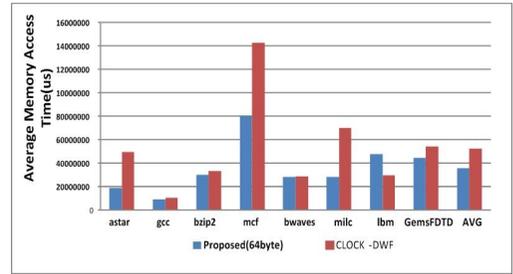


그림 6. 평균 메모리 접근 시간

Fig. 6 Average Memory Access Time

더욱이 64byte 블록 크기를 가지는 구조는 그 페이지를 PCM에 오래 저장하게 된다. 이는 다른 블록 크기에 비해 쓰기 요청된 페이지가 PCM에서 희생 페이지로 선택될 확률이 더 높다. 시뮬레이션 결과, 64byte의 블록 크기가 다른 블록 크기에 비해 다소 높은 메모리 접근 실패율을 보였다.

그림 6은 제안된 하이브리드 메모리의 성능평가를 위해 제안된 하이브리드 구조와 동일한 목적과 구조를 가지는 CLOCK-DWF와 64Byte의 PCM 접근을 가지는 제안된 하이브리드 메모리와 평균 메모리 접근 시간에 대한 성능평가를 나타내고 있다.

제안된 하이브리드 메모리는 CLOCK-DWF에 비해 약 평균 30%의 메모리 접근 시간을 줄였다. 시뮬레이션 결과, 제안된 하이브리드 메모리 구조가 CLOCK-DWF에 비해 평균 페이지 쓰기 동작을 약 22% 줄였으며, 전체 메모리 접근 실패율에서는 평균 5%의 성능 향상을 이루었다. 그림 6에서 보듯이 lbm을 제외한 모든 벤치마크에서 좋은 성능을 보이고 있다. lbm에서 CLOCK-DWF보다 낮은 성능을 보이는 것은 앞서 언급한 것처럼, lbm의 특성에 의한 것으로, 비슷한 특성을 가지는 mcf에 비해 블록 크기변화에 lbm이 더 민감하기 때문이다. 실제 제안된 구조에서 lbm은 CLOCK-DWF에 비해 약 40%의 높은 페이지 쓰기 동작을 가졌다. 반면, astar, mcf 그리고 milc에서는 효과적인 성능향상을 보이고 있다. 결론적으로 이러한 성능향상은 각각 CLOCK-DWF에 비해 약 70%, 23% 그리고 67%의 PCM의 페이지 쓰기를 줄인 것으로 나타났다.

#### IV. 결론

본 논문에서는 현재 차세대 메모리로 주목받고 있는 비휘발성 메모리인 PCM을 DRAM 대체 메모리로 사용하기 위한 PCM&DRAM 하이브리드 메모

리의 페이지 정책을 제안하였다. 하이브리드 메모리에서 PCM이 가장 고려되어야 할 문제는 쓰기 제한이다. 이를 해결하기 위한 최선의 방법중 하나가 프로그램 수행시 적합한 지역성을 이용한 페이지 교체 정책이다. 본 논문에서는 PCM과 DRAM의 효과적인 페이지 교체를 위하여, DRAM에 최하위 계층에서 요청된 페이지와 PCM에서 쓰기 요청에 대한 시간 지역성을 가지는 페이지를 저장하게 하였으며, PCM에서는 DRAM과 빈번한 페이지 교체를 줄이기 위하여 쓰기 요청에 대한 공간 지역성이 높은 페이지를 제한된 블록단위로 쓰기를 수행하였다. 이러한 페이지 교체 알고리즘을 제안함으로 PCM & DRAM 하이브리드 메모리의 성능향상을 도모하였다.

## References

- [1] Next Generation Nonvolatile Memory Semiconductor Technology Trend, <http://www.epnc.co.kr/news/articleView.html?idxno=58446>.
- [2] Y. Xie, "Future Memory and Interconnect Technologies," Proceedings of Design, Automation and test in Europe Conference and .Exhibition, pp. 964-969, 2013.
- [3] S. Mittal, J. S. Vetter, "A Survey of Software Techniques for Using Non-volatile Memories for Storage and Main Memory Systems," IEEE Transactions on Parallel and Distributed Systems, Vol. 27, No. 5, pp. 1537-1550, 2016.
- [4] B. Jung, J. Lee, "Analysis on the Effectiveness of the Filter Buffer for Low Power NAND Flash Memory," IEMEK J. Embed. Syst. Appl., Vol. 7, No. 4, pp. 201-207, 2012 (in Korea).
- [5] A. N. Jacobvite, R. Calderbank, D. J. Sorin, "Coset Coding to Extend the Lifetime of Memory," Proceedings of IEEE International Symposium on High Performance Computer Architecture, pp. 222-233, 2013.
- [6] R. Maddah, S. M. Seyedzadeh, "CAFO: Cost Aware Flip Optimization for Asymmetric Memories," Proceedings of the IEEE 21st International Symposium on High Performance Computer Architecture, pp. 320-330, 2015.
- [7] S. Im, D. Shin, "Differentiated Space Allocation for Wear Leveling on Phase-change Memory-based Storage Device," IEEE Transactions on Consumer Electronics, Vol. 60, No. 1, pp. 45-51, 2014.
- [8] M. K. Qureshi, V. Srinivasan, J. A. Rivers, "Scalable High Performance Main Memory System Using Phase Change Memory Technology," ACM SIGARCH Computer Architecture News, Vol. 37, No. 3, pp. 24-33, 2009.
- [9] K. Park, S. Yoon, S. Kim, "Selective Data Buffering Module for Unified Hybrid Storage System," Proceedings of the IEEE/ACIS 14th International Conference on Computer and Information Science, pp. 173-178, 2015.
- [10] G. Dhiman, R Ayoub, T. Rosing, "PDRAM: A Hybrid PRAM and DRAM Main Memory System," Proceedings of Design Automation Conference, pp. 664-669, 2009.
- [11] S. Lee, H. Bahn, S. H. Noh, "CLOCK-DWF: A Write-History-Aware Page Replacement Algorithm for Hybrid PCM and DRAM Memory Architecture," IEEE Transactions on Computers, Vol. 63. No. 9, pp. 2187- 2200, 2013.
- [12] M. Lee, D.H. Kang, J. King, "M-CLOCK: Migration-optimized Page Replacement Algorithm for Hybrid DRAM and PCM Memory Architecture," Proceedings of the ACM Symposium on Applied Computing, pp. 2001-2006, 2015.
- [13] N. Nethercote, J. Seward, "Valgrind: A Program Supervision Framwork," Elsevier Electronc Notes in Theoretical Computer Science, Vol. 89, No. 2, pp. 44-66, 2003.
- [14] HD Tune Pro, <http://www.hdtune.com>

**Bo-Sung Jung (정 보 성)**

He received B.S. and M.A. degrees from GyeongSang National University in 2006 and 2008 respectively. He is currently a Ph.D. student in Control and Instrument Engineering at GyeongSang National University. His research interests include cache memory, and PCM memory.

Email: blueking80@gnu.ac.kr

**Jung-Hoon Lee (이 정 훈)**

He received the M.S. and Ph.D. degree in Computer Science from Yonsei University, Seoul, Korea, in 2001 and 2004, respectively. He is currently a professor in GyeongSang National University. His research interests include advanced computer system and next flash memory.

Email: leejh@gsnu.ac.kr