

# 제한된 Copyback 사용을 통한 가비지 컬렉션 성능 향상 기법

홍두원<sup>o</sup>, 신슬기, 김지홍

서울대학교 컴퓨터공학부

{dwhong, sgshin, jihong}@davinci.snu.ac.kr

## Improving performance of Garbage Collection by using Restricted Copyback

Duwon Hong<sup>o</sup>, Seulgi Shin, Jihong Kim

Department of Computer Science and Engineering, Seoul National University

### 요 약

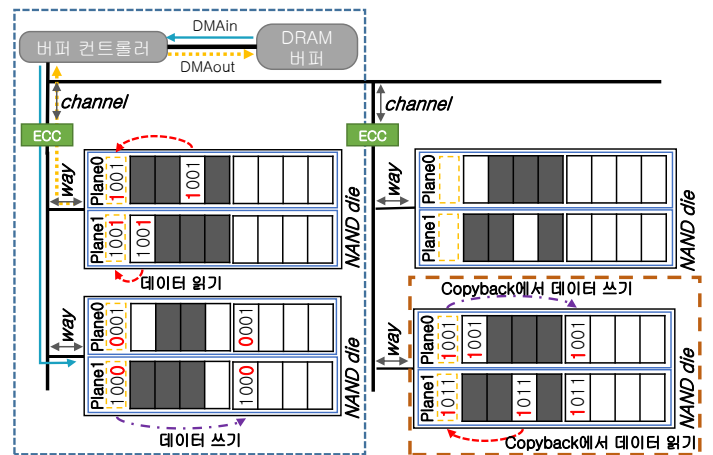
가비지 컬렉션의 수행 시간을 최소화 하는 것은 SSD 성능에 매우 중요하다. Copyback 명령을 사용하면 가비지 컬렉션 수행 시간을 최소화 할 수 있지만 ECC에 의한 에러 정정을 수행하지 않아 데이터 신뢰성에 악영향을 끼치는 한계점이 있다. 본 논문에서는 copyback 수행 시 실제 NAND칩의 오류 전파 특성을 파악하여 데이터 신뢰성이 문제가 되지 않는 copyback 한계 횡수를 규명하고, 제한된 횡수까지만 copyback을 활용하는 rcopyback 모델을 정의한다. 그리고 제한된 횡수의 rcopyback을 활용하는 FTL을 제안한다. 제안된 FTL은 데이터 신뢰성 문제를 일으키지 않으면서도 가비지 컬렉션 과정의 DMA 비용을 최소화함으로써 copyback을 사용하지않는 baseline 대비 평균 I/O 처리량이 54% 개선되었다.

### 1. 서 론

플래시 기반 Solid State Drive (SSD) 저장장치는 물리적 특성으로 인한 erase-and-write 제한때문에 가비지 컬렉션이 필수적이다. 하지만 가비지 컬렉션이 일어나게 되면 사용자 수준의 응답시간에 많은 악영향을 끼치기 때문에 이를 효율적으로 수행하는 것은 SSD의 성능에 매우 중요하다.

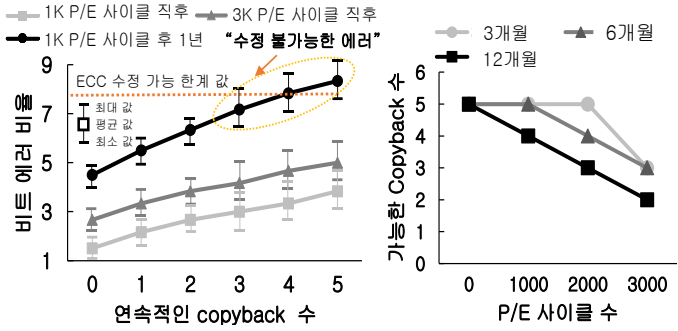
그림 1(a)는 가비지 컬렉션이 수행될 때 off-칩 데이터 복사의 경로를 그린 것이다. Off-칩 데이터 복사는 SSD 내부의 DRAM 버퍼를 이용해서 가비지 컬렉션의 지움 대상 블록의 유효 데이터를 새로운 블록에 복사해 준다. 그림에 나타난 Off-칩 데이터 복사에 걸리는 시간은  $t_{COPY} = t_R + t_{DMAout} + t_{DMAin} + t_{PROG}$  로 나타낼 수 있다.  $t_R$  은 NAND 셀에서 plane 레지스터로 데이터를 읽어오는 시간을 나타내고  $t_{DMAout}$ ,  $t_{DMAin}$ 는 DRAM버퍼와 레지스터 사이에서 데이터를 옮기는 시간을 나타내고  $t_{PROG}$ 는 데이터가 써지는 시간이다. 예전의 SSD는  $t_{PROG}$ 가 데이터 복사 시간의 대부분을 차지했지만 NAND 최적화에 의한  $t_{PROG}$ 의 감소[1]와 다수의 channel과 way로 구성된 고성능 SSD에서의 높은 병렬성은  $t_{COPY}$ 에서 차지하는  $t_{DMA}$ 의 비율을 높여서  $t_{DMA}$  가 새로운 병목이 되었다.

그림 1(b)는 가비지 컬렉션이 copyback[2]을 이용해서 수행될 때의 경로이다. Copyback은 plane의 레지스터에 가비지 컬렉션 대상 페이지를 읽어내고 그것을 같은 plane의 다른 페이지에 바로 써주는



(a) Off-칩 데이터 복사 (b) copyback  
그림 1. 가비지 컬렉션 데이터 이동 경로

순서로 진행된다. Copyback은 블록 간 복사를 수행할 때 DRAM 버퍼로의 DMA를 수행하지 않아  $t_{COPY}$  시간을 줄인다. 하지만 copyback은 NAND에서 발생한 에러를 정정하는 플래시 메모리 컨트롤러의 Error Correction Code (ECC) 모듈을 우회하여 에러 비트가 누적되는 문제가 있다. 그래서 Copyback이 반복적으로 수행되는 경우 비트 에러 비율(Bit Error Rate, BER)의 누적량이 ECC 모듈의 정정 능력을 넘게 되면 데이터를 읽을 수 없는 문제가 생기게 된다.



(a) BER 변화 (b) 가용 copyback 수 변화  
 그림 2. copyback 데이터 신뢰성 평가

본 논문에서는 동일 데이터에 대해 Copyback을 연속적으로 수행할 때 발생할 수 있는 데이터 신뢰성 문제를 해결하기 위해 연속적인 copyback의 사용 횟수가 제한된 rcopyback을 제안한다. Rcopyback은 제한된 횟수 이후에 ECC 모듈을 통해서 누적된 에러들을 정정시키는 것을 빼고는 copyback과 동일하다. 그리고 Rcopyback 기법을 활용하는 FTL을 구현하여 다양한 워크로드에서 I/O 처리능력이 평균적으로 54% 개선된 것을 확인했다.

논문의 구성은 다음과 같다. 2장에서 copyback의 오류 전파 특징을 살펴보고 데이터 신뢰성 문제없이 copyback을 활용하는 rcopyback을 제안한다. 그리고 3장에서 rcopyback 모델링을 활용한 FTL 구현을 설명한다. 4장에서는 구현된 FTL의 성능 실험을 통해 제안한 기법의 효과를 설명하고 5장에서 결론을 맺는다.

## 2. Rcopyback

### 2.1 copyback 오류 전파(Error Propagation) 특징

연속적인 copyback에 의한 오류 전파 특성을 평가를 위해 64개의 워드 라인(Word Line, WL)을 가진 멀티 레벨 셀(Multi-Level Cell, MLC) NAND칩 20개를 이용하여 총 81,920 페이지에 대해서 동일 데이터에 대한 연속적인 copyback을 수행하는 실험을 진행했다. 실험 결과 오류 전파 특성이 각 워드 라인 별로 다를 수 있었는데, 62 번 워드 라인의 MSB 페이지에서 오류 전파가 가장 많이 발생했다. 이는 외각의 워드 라인이 핫 캐리어 효과(hot carrier effect)와 GIDL (gate incuded drain leakage) 같은 노이즈에 취약하고, Vpass 교란 효과가 크게 나타나기 때문이다. 우리의 관심사는 안전하게 연속적으로 copyback을 활용 가능한 한계를 파악하는 것이기 때문에 copyback을 수행했을 때 BER의 값이 가장 크게 누적될 수 있는 조합인 가비지 컬렉션 대상 페이지와 복사할 페이지가 모두 WL 62에 있을 때를 대상으로 실험을 진행했다. 실험을 통해 측정된 누적 BER 값은 그림 2(a)에서 나타냈다. 각 블록의 누적 BER의 값은 P/E (Program/Erase) 사이클이 0번 수행된 블록에서 보존 시간이 0인 시점의 BER 값을 1로 두고 정규화

표1. P/E 사이클에 따른 copyback 한계 횟수

P/E 사이클	1-1000	1001-2000	2001-3000
Copyback 값	4	3	2

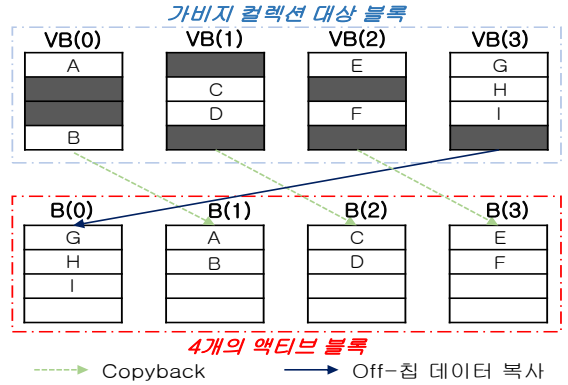


그림 4. 가비지 컬렉션 수행 예시

하였다. 만약 블록의 P/E값이 3000인 조건에서 1년의 데이터 보존 기간이 요구된다면 동일 데이터에 대해 copyback을 3번 이상 사용했을 때 ECC모듈이 수정 가능한 BER을 넘게 되어 신뢰성 문제가 생길 수 있음을 확인할 수 있다.

### 2.2 Rcopyback 모델링

오류 전파의 특성 연구를 통해 x번의 P/E 사이클이 수행된 블록에서 t달 동안 데이터 보존이 필요할 때 신뢰성 문제없이 수행할 수 있는 copyback 한계 횟수  $CT(x, t)$ 를 정의했다. 그림 2(b)는 MLC NAND 칩에서 다양한 P/E주기 및 보존 시간 요구사항에 따른 copyback 한계 횟수의 변화를 보여준다. 같은 P/E사이클에서 보존 시간 요구가 증가하면 가능한 연속적인 copyback 최대 수는 줄어든다. 그리고 같은 보존 시간 요구조건을 갖는 경우 P/E 값이 증가할수록 copyback 수행 가능 횟수는 줄어든다. 30°C에서 1년의 데이터 보존이 요구되었을 때 P/E사이클이 0에서 3000으로 증가하면 copyback의 한계 수행 횟수가 5에서 2로 줄어들게 된다. 이것은 3000번의 P/E 사이클이 수행된 블록에서 copyback이 최대 2번만 수행될 수 있다는 것을 의미한다. Copyback의 한계 횟수가 2일 때 세 번째 데이터 이동이 같은 페이지에 요구되면 off-칩 데이터 복사를 수행해야한다. 이 과정에서 누적된 비트 에러가 FMC의 ECC 모듈에서 수정되며 이로 인해 데이터 신뢰성에 문제가 생기지 않는다. 표1은 1년의 데이터 보존이 요구되었을 때 P/E 사이클에 따른 copyback 한계 횟수를 나타낸 것이다.

## 3. Rcopyback을 활용한 FTL 구현

2장에서 제안된 rcopyback 모델링을 활용하기 위한 FTL을 구현하였다. copyback이 한계 횟수 이상 수행될 수 없도록 강제 하려면 각 페이지의 copyback 누적 횟수를 저장하고 관리해야 한다. 가장 쉬운 구현 방법은

각 페이지 별로 메모리를 할당하여 copyback의 누적 횟수를 관리하는 방법이다. 하지만 각 페이지 별로 copyback 누적 수행 수를 저장하는 것은 많은 메모리 용량이 필요한 비효율적인 방법이다. 예를 들어 4KB 페이지 매핑의 16TB SSD에는 1.4GB의 메모리가 copyback 누적 수행 수를 관리하기 위해 추가로 필요하다. 또한 각 페이지 별로 copyback 누적 수행 수를 관리하는 방식은 FTL 프로그램의 수행 시간을 증가시키는 오버헤드를 발생시킬 수 있다. 이러한 오버헤드를 피하기 위해 한 블록 내의 모든 페이지들이 동일한 copyback 누적 수를 가질 수 있도록 하여 하나의 값으로 여러 페이지들을 대표 하도록 했다. 블록단위로 copyback 누적 수행 수를 관리하면 페이지 단위로 관리할 때보다 필요한 메모리 공간도 감소하고 컴퓨팅 오버헤드도 최소화 되는 장점이 있다.

구현된 FTL은 한 블록에 같은 copyback 누적 수를 가진 페이지들을 관리하기 위해 다수의 액티브 블록을 유지한다. Copyback 한계 횟수가 N인 조건에서는 관리하는 액티브 블록 수가 N+1로 정해진다. 그림 4는 copyback 한계 횟수가 3일 때 가비지 컬렉션 수행 예시이다. VB(n)은 가비지 컬렉션 대상이 된 블록, B(n)은 제한된 수의 copyback 수행을 위해 유지되는 액티브 블록을 의미한다. 만약 copyback 누적 횟수가 0인 VB(0)이 가비지 컬렉션 지움 대상 블록으로 선택되면 copyback을 통해 VB(0)에 있는 유효 데이터 A와 B는 액티브 블록인 B(1)에 옮겨진다. 그리고 VB(3)이 가비지 컬렉션 지움 대상 블록으로 선택되면 off-칩 데이터 복사를 이용해서 유효 페이지들이 B(0)로 옮겨진다.

#### 4. 실험 결과

제안된 기법의 검증을 위해서 SSD 에뮬레이터[3] 환경을 이용하여 호스트 레벨 FTL에 rcopyback 모델링을 적용하였다. 표2는 제안된 기법이 구현된 SSD 에뮬레이터 환경을 나타낸 것이다. 전체 물리적 영역에서 90%의 영역을 논리공간으로 할당하였다. 실험을 위해서 Filebench[4]와 Sysbench[5]를 이용해서 네 가지 I/O trace를 생성했다. 표3에서 실험을 진행한 I/O trace의 읽기-쓰기 비율과 WAF값을 확인할 수 있다. 여러 I/O trace에서 rcopyback 모델링이 적용된 FTL의 세가지 버전인 FTL2, FTL3, FTL4의 I/O 처리량을 측정했다. FTLn에서 n은 copyback 한계 횟수를 나타낸다. 모든 측정 값은 가비지 컬렉션에서 off-칩 데이터 복사만 이용하는 페이지 레벨 매핑 FTL의 결과 값을 1로 두고 정규화했다. 그림5는 각각 다른 copyback 한계 횟수를 가진 FTL에서 도출한 I/O 처리량이다. Copyback의 한계 횟수가 커질수록 copyback을 이용한 데이터 복사를 더 많이 할 수 있기 때문에 I/O 처리량이 증가한다. 그래서 실험결과에서 기존의 FTL보다 구현된

표 2. SSD 에뮬레이터 환경

용량	NAND칩	채널	집당 블록 수	블록당 페이지수
64GB	8	8	1024	64

표 3. I/O trace의 읽기쓰기비율과 WAF

	OLTP	NTRX	Fileserver	Varmail
읽기쓰기비율	7:3	0.5:9.5	4:6	4:6
WAF	2.17	2.11	3.08	1.8

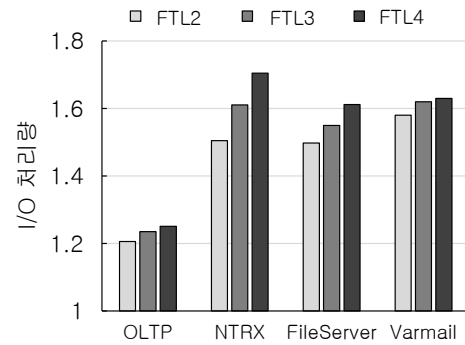


그림 5. Copyback 한계 횟수에 따른 I/O 처리량 비교

FTL의 성능이 평균적으로 FTL2에서는 41%, FTL4에서는 54% 향상된 것을 확인할 수 있다. 반대로 Varmail같이 데이터가 순차적으로 업데이트되는 경우에 가비지 컬렉션시 데이터 이동이 많지 않아서 copyback의 한계 횟수 증가의 효과가 적었다.

#### 5. 결론

이 논문에서는 동일 데이터에 대해 연속적으로 수행되는 Copyback에 의해 발생할 수 있는 데이터 신뢰성 문제를 해결하기 위해 copyback의 횟수가 제한된 rcopyback을 제안했다. 실험을 통해 실제 NAND칩의 오류전파 특성을 파악하여 데이터 신뢰성 문제가 생기지 않는 copyback의 한계 횟수를 찾고 rcopyback 모델링에 사용했다. 그리고 rcopyback 모델을 활용하는 FTL을 구현했다. 실험을 통해 구현된 FTL이 copyback을 지원하지 않는 FTL보다 평균적으로 54%의 성능향상을 보인다는 것을 확인했다.

#### 참고문헌

- [1]Seungjae Lee et al., "A 128Gb 2b/cell NAND flash memory in 14nm technology with tPROG= 640μs and 800MB/s I/O rate", *IEEE International Solid-State Circuits Conference*, 2016.
- [2]NAND Flash Performance Improvement Using Internal Data Move. Technical Note, pp.29-15.
- [3]Sang-Woo Jun et al., "Bluedbm: An appliance for big data analytics", *International Symposium on Computer Architecture*, 2015.
- [4]Filebench. <http://filebench.sourceforge.net>.
- [5]Sysbench. <http://github.com/akopytov/sysbench>.