

논문 2011-48SD-8-7

SSD 컨트롤러 최적 설계 기법

(Design Optimization Techniques for the SSD Controller)

이 두 진*, 한 태 희**

(Doo Jin Yi and Tae Hee Han)

요 약

플래시 메모리는 빠른 처리 속도, 비휘발성, 저전력, 강한 내구성으로 인해 최근 다방면에서 활용되는 비중이 점점 커지고 있고, 최근 비트 당 가격이 저렴해지면서 NAND 플래시 기반의 SSD (Solid State Disk)가 기존 기계적 메커니즘의 HDD (Hard Disk Drive)를 대체할 새로운 저장 장치로 주목받고 있다. 특히 모바일 기기에 적용되는 싱글 패키지 SSD 제품의 경우 병렬 처리를 통한 성능 향상을 위해 채널 수를 증가시키면 NAND 플래시 컨트롤러의 면적과 입출력 핀 수가 채널 수 증가에 따라 증가하여 폼팩터 (form factor)에 직접적인 영향을 주게 된다. 본 논문에서는 NAND 플래시 채널 수와 인터페이스의 채널당 FIFO 버퍼 사이즈를 최적화하여 SSD 컨트롤러의 성능을 고려한 면적과 입출력 핀 수를 최소화하고 이를 폼팩터에 반영하는 방법을 제안한다. 이중 버퍼를 채용한 10채널 지원 SSD 컨트롤러에 대해서 실험을 통해 동일한 성능을 유지하면서도 버퍼 블록 사이즈를 73%정도 축소시킬 수 있었고, 컨트롤러 전체 칩 면적으로는 채널 수 감소에 따른 채널별 컨트롤 블록과 입출력 핀 수 감소 등으로 인해 대략 40%정도 축소 가능할 것으로 예상된다.

Abstract

Flash memory is becoming widely prevalent in various area due to high performance, non-volatile features, low power, and robust durability. As price-per-bit is decreased, NAND flash based SSDs (Solid State Disk) have been attracting attention as the next generation storage device, which can replace HDDs (Hard Disk Drive) which have mechanical properties. Especially for the single package SSD, if channel number or FIFO buffer size per channel increases to improve performance, the size of a controller and I/O pin count will increase linearly with channel numbers and form factor will be affected. We propose a novel technique which can minimize form factor by optimizing the number of NAND flash channels and the size of interface FIFO buffer in the SSD. For SSD with 10 channel and double buffer, the experimental results show that buffer block size can be reduced about 73% without performance degradation and total size of a controller can be reduced about 40% because control block per channel and I/O pin count decrease according to decrease channel number.

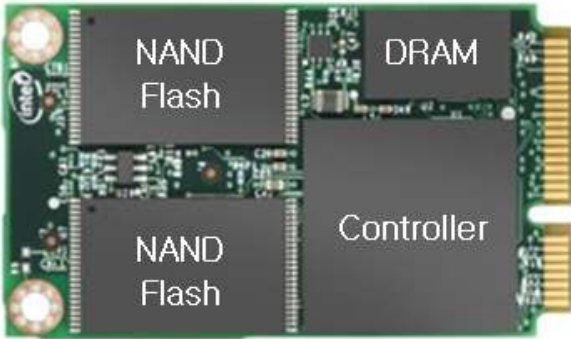
Keywords : SSD, NAND Flash, DRAM, buffer, FIFO, form factor

I. 서 론

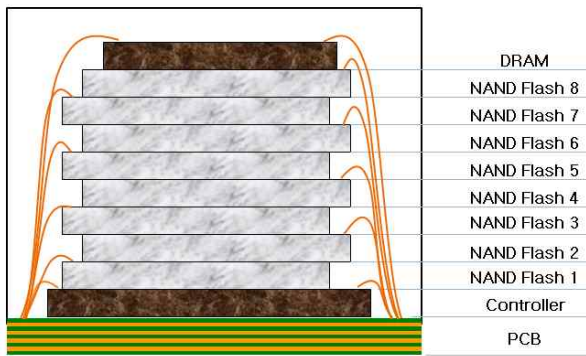
플래시 메모리는 빠른 처리 속도, 비휘발성, 저전력, 강한 내구성으로 인해 최근 다방면에서 활용되는 비중이 점점 커지고 있고, 특히 여러 개의 NAND 플래시 칩과 제어장치로 구성된 SSD^[1]는 동일한 인터페이스

규격일지라도 내부 동작 속도 차이로 인해 전송 속도 면에서 HDD보다 성능이 뛰어나 향후 HDD를 대체할 차세대 저장 장치로 주목받고 있다. 기계적 메커니즘의 HDD와 달리 SSD는 플래시 메모리를 사용하여 데이터를 저장하는 전기적 장치로서 이미 넷북과 태블릿PC 등 모바일 기기에 적용되고 있으며, 최근 출시된 Sandisk사 iSSD^[2]의 경우에는 MCP (Multi-Chip Package)^[3]의 형태로 16 × 20 × 1.85 (mm)로 우표 정도 크기에 64GB의 저장 용량을 갖고 있다. SSD의 구조를 살펴보면 그림 1에서와 같이 다수의 NAND 플래시와 DRAM 버퍼, SSD 컨트롤러로 구성되어 있다. 이 중

* 학생회원, ** 평생회원, 성균관대학교
정보통신공학부
(School of Information Communication Engineering,
Sungkyunkwan University)
접수일자: 2011년4월19일, 수정완료일: 2011년7월20일



(a)

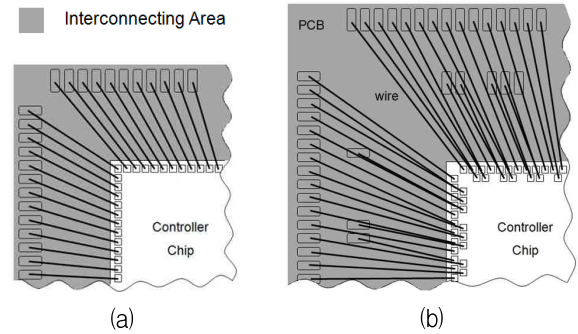


(b)

그림 1. SSD의 구조, (a) mSATA SSD 구조, (b) 싱글 패키지 SSD 수직 구조
 Fig. 1. SSD structure, (a) mSATA SSD structure, (b) vertical structure of single package SSD.

로직 공정을 사용하는 SSD컨트롤러는 비용 문제 등으로 최첨단 미세 공정을 사용하는 NAND 플래시나 DRAM보다 이전 세대 공정으로 제조되고 또한 여러 칩들을 컨트롤하기 위한 컨트롤 블록과 많은 입출력 핀 수로 인해 사이즈가 큰 경우가 대부분이다. 이로 인해 iSSD와 같은 싱글 패키지인 경우에는 컨트롤러 칩이 폼팩터를 결정하는 가장 큰 요인이 된다.

SSD는 성능을 향상시키기 위한 방법으로 멀티채널 구조와 이중 버퍼^[4]를 적용하고 있는데, 병렬 처리를 통한 성능 향상을 위해 채널 수를 8채널에서 10채널, 16채널로 증가시키면 채널 당 필요한 NAND 플래시 입출력 핀, 컨트롤 신호 핀, 파워, 그라운드 핀 등의 증가로 컨트롤러의 입출력 핀 수가 증가하게 되고 NAND 플래시 컨트롤러의 면적도 증가하게 된다. 만일 컨트롤러 면적은 증가하지 않고 채널 수만 증가시킬 수 있다고 해도 채널 수 증가에 따라 입출력 핀 수는 증가할 수밖에 없고 칩과 PCB 연결을 위한 wire간 간섭 문제, PCB 회로 패턴 공정, PCB상 신호 간섭 문제 등으로



(a)

(b)

그림 2. 입출력 핀 수에 따른 내부 연결 공간 비교, (a) 입출력 핀 수 증가 전, (b) 입출력 핀 수 증가 후

Fig. 2. Comparison of interconnecting area according to I/O pin count, (a) before increase of I/O pin count, (b) after increase of I/O pin count.

내부 연결 공간 (Interconnecting Area)이 증가하게 된다. 그림 2와 같이 8채널에서 16채널로 증가하는 경우 입출력 핀수는 대략 500개에서 1000개 수준으로 두 배 정도 증가하고 내부 연결 공간은 기존 대비 30% 이상이 더 요구되어 채널 수 증가는 폼팩터에 직접적인 영향을 주게 된다. 이러한 문제를 해결하기 위하여 본 논문에서는 인터페이스 채널 수와 채널당 FIFO 버퍼 사이즈 최적화를 통해 성능을 고려하여 폼팩터 최소화할 수 있는 SSD 컨트롤러 설계 기법을 제안한다.

이후 본 논문의 구성은 다음과 같다. II장에서는 SSD의 구조와 동작, 문제점에 대해서 알아보고, III장에서는 문제점을 해결하기 위해 플래시 채널 수 및 인터페이스 FIFO 버퍼 사이즈를 최적화하는 방법을 설명한다. IV장에서는 시뮬레이션 환경을 구성한 후 데이터 전송 시간을 비교하여 성능을 실험하였으며, V장에서는 이 실험 결과를 비교 분석하여 결론을 도출하였다.

II. SSD의 내부 구조와 문제점

SSD는 여러 개의 NAND 플래시 메모리들이 병렬로 연결되는 형태의 저장장치로서 멀티채널 병렬 처리를 통한 빠른 읽기 및 쓰기 동작이 가능하다.

1. 인터페이스와 멀티채널 구조

제조사마다 다소 차이는 있지만, 일반적인 SSD의 내부 구조는 그림 3과 같다. Host 인터페이스를 통하여 읽기 또는 쓰기에 대한 요청이 들어오면 DRAM 버퍼를 거쳐서 컨트롤부인 프로세서와 SRAM이 데이터의

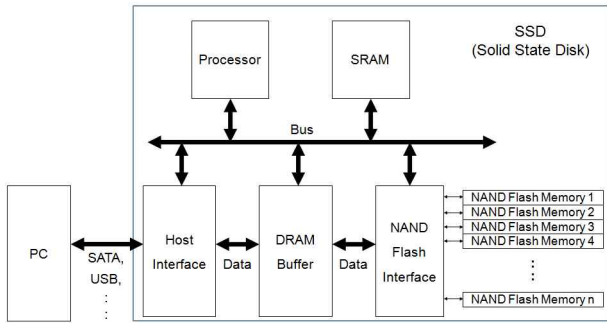


그림 3. SSD 내부 구조
Fig. 3. SSD internal structure.

알맞은 위치를 파악하고 다시 DRAM 버퍼를 통해 NAND 플래시 메모리에 데이터를 기록하거나 읽어오게 된다. 컨트롤부를 제외하고 데이터 버스 부분만을 고려한 SSD의 인터페이스와 FIFO의 구조는 그림 4와 같다. SSD 컨트롤러 내부에는 Host와 데이터를 주고받는 Host 인터페이스와 NAND 플래시와 데이터를 주고받는 플래시 인터페이스가 존재하고 각각의 인터페이스 내부에는 FIFO 버퍼가 존재하여 데이터들이 버스트 모드에 맞춰 전송될 수 있도록 임시로 데이터를 저장하는 역할을 하게 된다. DRAM 버퍼에는 읽기와 쓰기를 위한 데이터뿐만 아니라 메타 데이터도 저장하게 되는데 Host에서 플래시로 또는 플래시에서 Host로 이동되는 데이터는 모두 DRAM 버퍼를 거쳐야만 한다. 이로 인해 DRAM 버퍼의 대역폭은 SSD의 성능을 결정하는 중요한 요인이 되고, Arbiter는 동시에 DRAM 버퍼에 액세스하려는 인터페이스들을 중재하는 역할을 한다.^[5]

SSD의 성능을 향상시키기 위해 플래시 메모리에는 멀티채널 구조가 적용되며, 이 구조에서는 여러 개의 메모리에 대한 동시 액세스가 가능하고 한 번에 읽고 쓰는 데이터양이 많아져 전체적으로 빠른 읽기 및 쓰기가 가능하게 된다. 예를 들어 쓰기인 경우 한 개 채널의 최대 성능이 40MByte/s이라고 하면 2채널인 경우 80MByte/s가 최대 성능이 되지만, 채널 수를 N개로 증가한다면 40MByte/s × N 채널의 성능을 얻을 수 있다. 하지만, 플래시 채널 수가 증가해도 DRAM 버퍼 대역폭이 인터페이스 전체 대역폭보다 작은 경우에는 DRAM 버퍼에서의 병목 현상으로 인해 채널 수에 비례하는 성능 향상은 이루어지지 않는다.

플래시에서 읽어오거나 저장하기 위한 데이터는 모두 DRAM 버퍼를 거쳐야하므로 DRAM 버퍼의 대역폭이 작은 경우 플래시 채널 수가 증가하더라도 DRAM

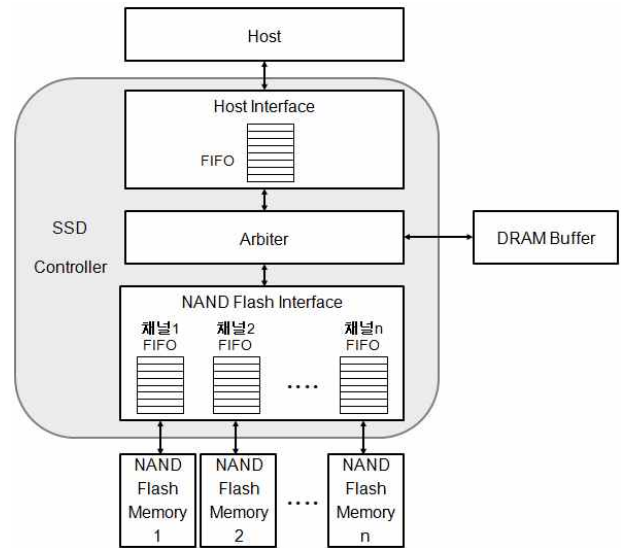


그림 4. SSD 컨트롤러 내부의 인터페이스 FIFO 버퍼
Fig. 4. Interface FIFO buffers in the SSD controller.

버퍼에서의 병목 현상으로 인해 채널 수에 비례하는 컨트롤러의 성능 향상은 이루어지지 않는다. 반대로 채널 수가 감소하면 여러 개의 플래시에 동시에 읽고 쓰는 데이터양 자체가 감소되어 컨트롤러의 성능은 저하된다. 이러한 문제를 DRAM 버퍼 대역폭을 고려한 플래시 채널 수를 최적화하여 해결하고자 하며, 이를 통해 컨트롤러의 입출력 핀 수도 최적화할 수 있다.

2. 인터페이스 FIFO 버퍼

데이터를 전송하는 동작은 크게 읽기와 쓰기 두 가지로 구분되며, 쓰기는 Host의 데이터를 DRAM 버퍼를 거쳐 플래시로 이동시켜 저장하는 것, 읽기는 플래시에 저장된 데이터를 DRAM 버퍼를 거쳐 Host로 불러오는 것을 의미한다. 데이터 쓰기의 경우, 플래시에 저장하기 위한 데이터가 Host 인터페이스 FIFO 버퍼에 임시로 저장되고, 저장된 데이터의 사이즈가 한 번에 전송 가능한 버스트 데이터 크기만큼 되면 DRAM 버퍼로 데이터를 전송하기 위해 Arbitrator에 DRAM 버퍼 액세스 권한을 요청하게 된다. 플래시 인터페이스 FIFO 버퍼를 비롯한 여러 인터페이스 FIFO 버퍼들이 동시에 DRAM 버퍼 액세스 권한을 요청하는 경우 Arbitrator는 중재 기법을 통해 그 중 하나의 인터페이스 FIFO 버퍼에 권한을 부여하고, Arbitrator를 통해 권한을 부여받은 FIFO 버퍼는 DRAM 버퍼에 데이터를 전송하게 된다. DRAM 버퍼에 저장된 데이터는 플래시 인터페이스 FIFO 버퍼가 비어있는 경우 Arbitrator를 통해 다

시 권한을 부여받아 FIFO 버퍼에 데이터를 저장하고 버스트 데이터 단위로 데이터를 플래시로 전송하게 된다. 여러 인터페이스 FIFO 버퍼들이 동시에 DRAM 액세스 권한을 요청하는 경우 Arbiter에서는 Fixed-Priority 기법이나 Round-Robin 기법과 같은 중재 방식을 적용하여 순서를 조정하여 액세스를 허용하도록 처리한다.

일반적인 인터페이스 FIFO 버퍼의 사이즈는 한 번에 전송 가능한 버스트 데이터 크기의 두 배이며, 이는 인터페이스 FIFO 버퍼의 데이터가 비워지는 동안에도 다음 전송을 위한 데이터를 미리 인터페이스 FIFO 버퍼에 채워 기다리는 시간을 최소화하여 효율을 향상시키기 위한 것이다. 그러나 멀티채널 구조의 SSD에서는 각 인터페이스의 대역폭의 합과 DRAM 버퍼의 대역폭 차이로 인해 대역폭이 상대적으로 작은 부분에서 병목 현상이 발생하고 이로 인해 FIFO 버퍼 사이즈 증가에 따른 성능 향상은 이루어지지 않는다. 동시에 여러 인터페이스 FIFO 버퍼들이 Arbiter에게 DRAM 버퍼 액세스 권한을 요청하는 경우 Arbiter는 한 번에 하나의 인터페이스에만 DRAM 버퍼 액세스 권한을 부여하게 되는데 이때 권한을 부여받지 못한 인터페이스 FIFO 버퍼들은 권한을 부여받을 때까지 데이터 전송이 지연되며, 결국 DRAM 버퍼 대역폭에 의한 병목 현상으로 인해 인터페이스 FIFO 버퍼 사이즈와 상관없이 DRAM 버퍼 대역폭에 의해 전송 성능은 한계가 발생하게 된다. 이와는 반대로 만일 DRAM 버퍼의 대역폭이 전체 인터페이스들의 대역폭 합보다 상대적으로 크다면 DRAM 버퍼가 처리할 수 있는 대역폭은 크지만, 인터페이스의 대역폭이 작아서 인터페이스 FIFO 버퍼에 버스트 데이터 크기만큼 데이터가 채워지기 위해 시간이 지연되고 DRAM 버퍼가 기다리는 경우가 생기게 된다.

이처럼 인터페이스 FIFO 버퍼에 이중 버퍼를 적용해도 충분히 효율적으로 활용되지 못하는 문제를 FIFO 버퍼의 사이즈를 최적화하고 면적 대비 성능 효율을 향상시켜 해결하고자 한다.

III. 제안하는 컨트롤러 최적화 방법

앞서 분석한 바와 같이 NAND 플래시 인터페이스의 채널당 FIFO 버퍼 사이즈와 채널 수를 최적화하면 컨트롤러 면적 대비 성능 효율을 향상시킬 수 있고 입출력 핀 수를 최소화할 수 있다.

1. 대역폭에 따른 채널 수 최적화

멀티채널 구조에서 DRAM 버퍼의 대역폭과 각 인터페이스 대역폭 합의 차이가 큰 경우에는 인터페이스 FIFO 버퍼가 DRAM 버퍼 액세스 권한을 요청한 후 기다리는 시간이 증가하거나 DRAM 버퍼가 인터페이스를 통해 데이터를 전송받지 못한 채 기다리는 시간이 증가하게 되는데, NAND 플래시 채널 수를 최적화하여 Host 인터페이스와 NAND 플래시 인터페이스 대역폭의 합을 DRAM 버퍼 대역폭과 유사한 수준으로 유지하면 대역폭 차이에 따른 성능 저하 문제를 해결할 수 있다.

DRAM 버퍼의 대역폭은 동작 주파수와 1클록 당 전송 가능한 데이터 크기의 곱으로 표현된다. 단, DDR2인 경우에는 1클록 사이클 당 2배의 데이터 전송이 가능하므로 대역폭도 2배가 된다. 식 (1)에서 BW_{DRAM} 은 DRAM 버퍼 대역폭, f_{DRAM} 은 DRAM 버퍼 동작 주파수, 32bit는 1클록 당 전송 가능한 데이터 크기를 의미한다.

$$BW_{DRAM}(Mbps) = f_{DRAM}(MHz) \times 32(bit) \quad (1)$$

전체 인터페이스 대역폭은 호스트 인터페이스 대역폭과 NAND 플래시 인터페이스 대역폭의 합이며, 전체 NAND 플래시 인터페이스 대역폭은 각 채널당 대역폭과 채널 수의 곱이다. 식 (2)에서 $BW_{NANDtotal}$ 는 NAND 플래시 인터페이스 대역폭의 합, f_{NAND} 는 NAND 플래시 동작 주파수, 8bit는 1클록 당 전송 가능한 데이터 크기, $N_{channel}$ 은 NAND 플래시 채널 수를 의미한다.

$$BW_{NANDtotal}(Mbps) = f_{NAND}(MHz) \times 8(bit) \times N_{channel} \quad (2)$$

따라서, 대역폭이 최적화되는 경우는 DRAM 버퍼 대역폭과 모든 인터페이스 대역폭의 합이 같을 때이고, 그 때의 NAND 플래시 채널 수는 식 (3)을 만족하는 가장 근접한 정수이다. 식 (3)에서 $N_{channel}$ 은 NAND 플래시 채널 수, BW_{DRAM} 은 DRAM 버퍼 대역폭, BW_{host} 는 Host 대역폭, BW_{NAND} 는 NAND 플래시 인터페이스 채널당 대역폭을 의미한다.

$$N_{channel} = \frac{BW_{DRAM} - BW_{host}}{BW_{NAND}} \quad (3)$$

200MHz DRAM, 40MHz NAND 플래시가 적용되고,

Host 인터페이스 대역폭이 300MByte/s인 SSD의 최대 성능은 식 (3)을 통해 NAND 플래시가 12채널일 경우이지만, 200MHz DDR2 DRAM, 166MHz NAND 플래시로 변경되면 8채널일 때 성능은 최적화가 된다.

2. 인터페이스 FIFO 버퍼 사이즈 최적화

인터페이스 FIFO 버퍼는 데이터 전송 시 버스트 모드를 고려하여 버스트 데이터 크기의 두 배를 적용하고 있는데, 이것은 버스트 모드 전송 효율을 극대화하기 위한 것이다. 하지만, 인터페이스 FIFO 버퍼의 사이즈가 증가한다고 해도 인터페이스와 DRAM 버퍼의 대역폭 차이에 의한 병목 현상으로 인해 FIFO 버퍼 사이즈에 비례하는 성능 향상은 이루어지지 않는다.

DRAM 버퍼와 인터페이스 대역폭에 따라 세 가지 경우로 나눌 수가 있는데, (1) DRAM 버퍼 대역폭이 인터페이스 대역폭의 합과 유사한 경우, (2) DRAM 버퍼 대역폭이 상대적으로 훨씬 큰 경우, 반대로 (3) DRAM 버퍼 대역폭이 상대적으로 작은 경우이다.

(1) 인터페이스 대역폭의 합과 DRAM 버퍼의 대역폭이 유사한 경우에는 데이터 전송 시 병목 현상은 발생하지 않고 DRAM 버퍼 대역폭만큼의 최대 성능이 나오게 된다. NAND 플래시의 채널당 대역폭이 100MByte/s이고 4채널인 경우 인터페이스 대역폭의 합은 400MByte/s이고, DRAM 버퍼의 대역폭이 400MByte/s라고 하면, 대역폭이 동일하기 때문에 인터페이스 FIFO 버퍼가 버스트 데이터 크기의 두 배라고 해도 그 공간을 모두 채울 수가 없고 버스트 데이터 크기만큼만 FIFO 버퍼를 반복해서 사용하게 된다.

(2) 인터페이스 대역폭의 합이 DRAM 버퍼 대역폭보다 큰 경우에는 데이터 전송 시 DRAM 버퍼에서의 병목 현상으로 인해 최대 성능은 대역폭이 작은 DRAM 버퍼 대역폭에 의해 결정된다. NAND 플래시의 채널당 대역폭이 100MByte/s이고 8채널인 경우 인터페이스 대역폭의 합은 800MByte/s이고 DRAM 버퍼의 대역폭이 400MByte/s라고 하면, 인터페이스 대역폭의 합이 DRAM 버퍼의 대역폭보다 훨씬 크기 때문에 데이터를 전송 중인 채널을 제외하고 다른 채널에서는 빠른 속도로 인터페이스 FIFO 버퍼를 금방 채우게 되고, 데이터가 채워진 상태에서 DRAM 버퍼 액세스 권한이 승인될 때까지 대기하게 된다. 대기 시간 발생은 데이터 전송 지연을 의미하며 이중 버퍼에 의한 성능 향상은 이루어지지 않는다.

(3) 인터페이스 대역폭의 합이 DRAM 버퍼 대역폭보다 작은 경우에는 최대 성능은 대역폭이 작은 인터페이스에 의해 결정된다. 채널당 대역폭이 100MByte/s이고 4채널인 경우 인터페이스 대역폭의 합은 400MByte/s이고 DDR2 DRAM 버퍼의 대역폭이 1600MByte/s라고 하면, DRAM 버퍼의 대역폭이 훨씬 크기 때문에 하나의 채널에서 FIFO 버퍼가 버스트 데이터 크기만큼 채워져 권한을 요청하게 되면 DRAM 버퍼에서는 FIFO 버퍼를 채우는 속도보다 빠른 속도로 FIFO 버퍼를 비우게 된다. 이에 모든 채널의 FIFO 버퍼는 거의 대부분 비어있는 상태를 유지하며 버스트 데이터 크기만큼의 FIFO 버퍼만 사용하게 된다.

모든 경우에 있어서 인터페이스 채널당 FIFO 버퍼의 사이즈는 버스트 데이터 크기만큼만 존재하면 되고, 최대 효율을 위해서는 FIFO 버퍼에 채워진 데이터가 전송되는 시간동안에도 FIFO 버퍼를 채울 수 있을 만큼 여분의 공간이 더 있으면 된다. 모든 채널에 순차적으로 동등한 액세스 권한이 부여되는 Round-Robin 방식에 의해 중재된다고 가정하면 인터페이스 FIFO 버퍼의 사이즈는 다음 식으로 결정된다.

$$S_{NANDbuffer} = S_{burst} + \frac{S_{burst} \times BW_{NAND}}{BW_{total}} \quad (4)$$

식 (4)에서 $S_{NANDbuffer}$ 는 플래시 인터페이스 FIFO 버퍼 사이즈, S_{burst} 는 버스트 데이터 크기, BW_{NAND} 는 NAND 플래시 채널당 대역폭, BW_{total} 은 모든 인터페이스 대역폭의 합을 의미한다. 수식을 통해 버스트 데이터 크기가 128Byte이고, NAND 플래시 대역폭이 40MByte/s, 전체 인터페이스 대역폭의 합이 320MByte/s인 경우에는 채널당 버퍼 사이즈가 144Byte, 전체 인터페이스 대역폭의 합이 160MByte/s인 경우에는 채널당 버퍼 사이즈가 160Byte일 때 컨트롤러 효율은 최적화가 된다.

본 장에서 제안하는 컨트롤러 최적화 방법은 다음의 두 가지로 정리된다.

(1) DRAM 버퍼 대역폭과 모든 인터페이스 대역폭의 합이 같아지도록 NAND 플래시 채널 수를 최적화하면 대역폭 차이에 의한 컨트롤러 성능 저하 문제를 해결할 수 있다.

(2) 인터페이스 FIFO 버퍼 사이즈 대비 성능 효율이 최적화 되는 경우는 FIFO 버퍼에 채워진 데이터를

DRAM 버퍼에 전송하는 동안에도 FIFO 버퍼에는 다음 전송을 위한 데이터가 지속적으로 채워지는 경우로, 이중 버퍼를 적용하여도 충분히 효율적으로 활용되지 못하므로 최적화된 인터페이스 FIFO 버퍼 사이즈는 식 (4)로 결정할 수 있다.

IV. 실험 및 결과

1. 테스트 환경 및 조건

테스트 환경은 DMA 모드에서 동일한 크기의 데이터 전송 요청에 따른 전체 데이터 전송 시간 비교를 위해 Host 인터페이스, 멀티채널 NAND 플래시 인터페이스, DRAM 버퍼로 구성된 SSD 시스템을 가정하였다. 삼성전자의 40Mbps SDR 32Gb NAND 플래시인 K9LBG08U0M^[6]와 133Mbps ToggleDDR 32Gb NAND 플래시인 K9GBGD8U0A^[7]을 기준으로 FIFO 버퍼 구조의 이중 버퍼를 가진 NAND 플래시 인터페이스를 모델링하였고, DRAM은 삼성전자의 133MHz DDR 1Gb Mobile DRAM인 K4X1G323PC와 200MHz DDR2 1Gb Mobile DRAM인 K4Q1G323BE^[8]을 기준으로 모델링하였으며, Host 인터페이스는 SATA2 규격인 대역폭 300MByte/s로 동작하도록 구성하였다. SSD 시스템은 Verilog HDL로 기술되었고, 삼성 65nm CMOS 라이브러리를 이용하여 합성하였다.

조건에 따른 SSD 컨트롤러의 효율을 비교하기 위하여 전체 데이터 전송량은 48섹터인 24KByte (512 Byte × 48섹터)로 고정하였고, DMA 모드에서의 데이터 전체 전송 시간을 비교, 분석하였다. 실제 상황에서는 DMA 모드로 연속 액세스를 통해 대용량의 데이터 전송이 이루어지게 되거나 랜덤 액세스를 통해 작은 용량의 데이터 전송이 이루어지게 되는데, 랜덤 액세스의 경우에는 논리 블록 주소 (Logical Block Address, LBA)를 검색하는 시간과 펌웨어에서 NAND 플래시 주소를 관리하는 정책에 따라 성능이 크게 달라지기 때문에 시뮬레이션 환경은 하드웨어 측면 최적화에 따른 성능 향상 결과를 비교하기 위해 연속 액세스 조건을 적용하였다.

2. 실험 결과

그림 5는 DRAM 버퍼 대역폭이 1600MByte/s, Host 대역폭이 300MByte/s일 때 NAND 플래시 채널 수에 따른 데이터 전송 시간을 비교한 결과이다. 식 (3)에 의

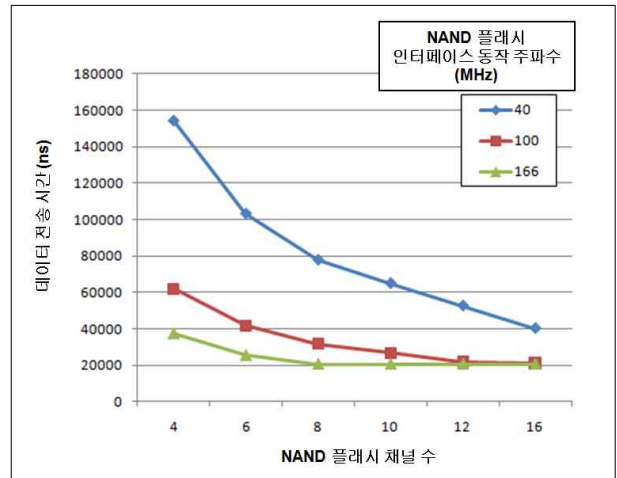


그림 5. 플래시 채널 수에 따른 성능 비교
Fig. 5. Performance according to number of flash channel.

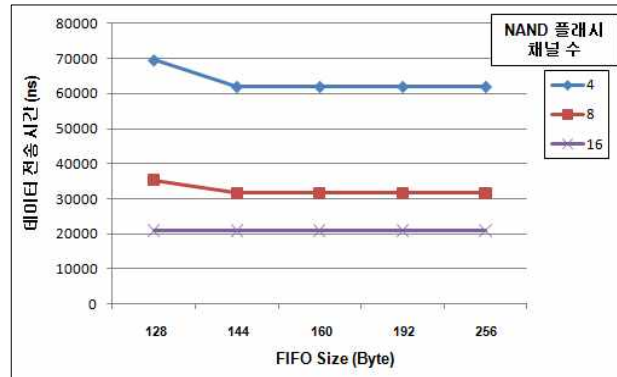


그림 6. FIFO 버퍼 사이즈에 따른 성능 비교
Fig. 6. Performance according to size of FIFO buffer.

해서 NAND 플래시 동작 주파수가 40MHz이면 32채널, 166MHz이면 8채널일 때 성능은 최적화가 되는데, 이것은 실험 결과와도 일치한다. NAND 플래시 동작 주파수가 40MHz이면 32채널일 때 최적화되므로 4채널에서 16채널로 채널 수가 증가할수록 성능도 향상되지만, 166MHz인 경우에는 8채널일 때 최적화되므로 채널 수가 8채널 이상으로 증가하면 채널 수에 따른 성능향상은 일어나지 않았다.

그림 6은 FIFO 버퍼 사이즈에 따른 성능 비교 결과이다. 버스트 데이터 크기가 128Byte일 때, NAND 플래시가 4채널, 8채널인 경우에 인터페이스의 버퍼 크기가 144Byte보다 증가하는 경우에는 성능 향상은 없었고, 16채널인 경우에는 FIFO 버퍼 사이즈가 버스트 데이터 크기인 128Byte일 때와 그보다 증가하는 경우에도 성능 차이는 없었다. 즉, 전체 전송 데이터의 크기가 랜덤 액세스에 비해 상대적으로 큰 연속 액세스인

경우에 식 (4)에 의해 예상했던 바와 마찬가지로 FIFO 버퍼 사이즈에 비례하는 성능 향상은 일어나지 않음을 실험을 통해 입증되었다.

300MByte/s의 SATA2 Host 인터페이스 규격을 지원하고 40Mbps NAND 플래시를 채용하고 있는 SSD의 경우 현재 성능 향상을 위해 10채널 또는 16채널로 구성되고 있으나, 향후 600MByte/s의 SATA3 Host 인터페이스 규격을 지원하고 개발 중인 200Mbps NAND 플래시를 채용하게 되면 식 (3)에 의해 5채널인 경우 성능이 최적화되어 기존의 10채널 또는 16채널 지원하는 컨트롤러를 그대로 적용하는 대신 5채널로 최적화된 컨트롤러를 적용하는 것이 가능하다. 또한 인터페이스 FIFO 버퍼 사이즈는 이중 버퍼 적용시의 256Byte에서 식 (4)에 의해 버스트 데이터 크기 수준인 144Byte로 축소시키는 것이 가능하다. 표 1은 기존 방식과 제안한 방법을 적용하여 최적화한 경우의 버퍼 블록 합성 결과인데, 10채널에서 5채널로 축소하고 인터페이스 FIFO 버퍼 사이즈를 256Byte에서 144Byte로 축소하는 경우 컨트롤러 버퍼 블록 사이즈는 최적화하기 전보다 73% 정도 축소시키는 것이 가능하였다.

제안한 방법을 적용하면 채널 수 감소에 따른 채널별 컨트롤러 블록 감소와 입출력 핀 수 감소 등으로 인해 컨트롤러 전체 칩 면적으로는 현재 출시되어 있는 SSD 컨트롤러들의 채널 수와 면적의 관계를 고려하였을 때 대략 40%정도 축소 가능할 것으로 예상된다. 또한 폼팩터 축소는 그림 7과 같이 컨트롤러 칩 면적뿐 아니라 채널 수와도 연관이 있는데, (b) 채널 수는 변경 없이 칩 면적만 감소되는 경우 입출력 핀 수에는 변함이 없어 폼팩터를 감소시키는 것에는 한계가 있으나 (c) 칩

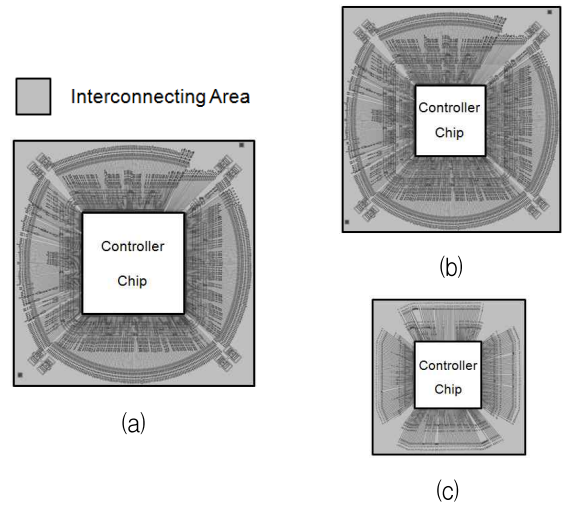


그림 7. 컨트롤러 칩 사이즈와 채널 수에 따른 폼팩터 비교, (a) 기존 10채널 컨트롤러, (b) 10채널에서 칩 사이즈만 40% 축소된 경우, (c) 칩 사이즈는 40% 축소되고 5채널로 감소된 경우
 Fig. 7. Form factor comparison according to chip size and channel number of controller, (a) 10 channel controller, (b) chip size 40% shrinkage for 10 channel controller, (c) chip size 40% shrinkage for 5 channel controller.

면적과 함께 채널 수가 절반으로 감소하게 되면 입출력 핀 수도 채널 수에 비례하여 감소하게 되어 단순히 칩 면적만 감소된 경우보다 폼팩터를 30%정도 더 축소시키는 것이 가능하다. 이러한 컨트롤러 최적화 설계는 DRAM과 Host 인터페이스, NAND 플래시 대역폭에 따른 채널 수를 결정하고 싱글 패키지 SSD의 경우 성능에 영향을 주지 않는 패키지 내부의 불필요한 공간을 최소화함으로써 성능을 고려한 컨트롤러 최적화 설계를 가능하게 하여 폼팩터를 최소화하는 것이 가능하다.

표 1. SSD 컨트롤러 버퍼 블록 합성 결과
 Table 1. Result of synthesis for SSD buffer block.

	SSD 컨트롤러 버퍼 블록	
	전체 면적 (um ²)	로직 게이트 (gate)
16채널 256Byte FIFO	104405	81567
10채널 256Byte FIFO	65253	50979
5채널 256Byte FIFO	32627	25490
5채널 144Byte FIFO	17774	13886

V. 결 론

모바일 기기에 적용되는 싱글 패키지 SSD 제품의 경우 컨트롤러의 면적과 채널 수에 의해 폼팩터가 결정되는 경우가 많은데, 병렬 처리를 통한 성능 향상을 위해 채널 수를 증가시키면 채널 당 필요한 입출력 핀 수가 증가하게 되고 NAND 플래시 컨트롤러의 면적도 증가하여 폼팩터에 영향을 주게 된다. 이에 본 논문에서는 Host와 플래시 인터페이스를 포함한 전체 인터페이스 대역폭의 합이 DRAM 버퍼 대역폭과 동일한 수준이 되도록 플래시 채널 수를 최적화하여 대역폭 차이에 따른 성능 저하 문제를 해결하고 입출력 핀 수를 최소화

할 수 있었다. 또한 인터페이스 FIFO 버퍼 사이즈는 이중 버퍼가 아닌 버스트 데이터 크기 수준으로 감소시켜 동일한 성능을 유지하면서도 컨트롤러 칩 면적을 최소화할 수 있었다. 제안한 방법을 적용하여 실험한 결과, 600MByte/s의 SATA3 Host 인터페이스 규격을 지원하고 개발 중인 200Mbps NAND 플래시를 채용하게 되면 기존의 이중 버퍼를 채용한 10채널 지원 SSD 컨트롤러를 그대로 적용하는 대신에 채널 수는 5채널로 축소하고 채널당 FIFO 버퍼 사이즈는 최적화하여 동일한 성능을 유지하면서도 버퍼 블록 사이즈를 73%정도 축소시키는 것이 가능하였다. 이는 전체 칩 면적으로 대략 40%정도 축소 가능할 것으로 예상되며 제안한 방법을 적용한 컨트롤러 최적화 설계는 성능을 고려한 폼팩터 최소화를 가능하게 하고 SSD의 장점인 경박단소에도 큰 기여를 할 수 있다.

참 고 문 헌

- [1] Samsung Electronics, <http://www.samsung.com/us/business/oem-solutions/storage-devices/ssd/index.html>
- [2] <http://www.sandisk.com/business-solutions/ssd/issd>
- [3] Micron Technology, Inc., "Multi-chip Stacked Devices", US Patent, no. 5,291,061, 1994.
- [4] J. U. Kang, J. S. Kim, C. Park, H. Park, and J. Lee, "A multi-channel architecture for high-performance NAND flash-based storage system", Journal of Systems Architecture, vol. 53, no. 9, pp. 644-658, 2007.
- [5] H. Shim, B. K. Seo, J. S. Kim, S. Maeng, "An adaptive partitioning scheme for DRAM-based cache in Solid State Drives", IEEE 26th Symposium on Mass Storage Systems and Technologies, pp.1-12, 2010.
- [6] Samsung Electronics, http://www.samsung.com/global/business/semiconductor/products/flash/Products_NANDFlash.html
- [7] Samsung Electronics, http://www.samsung.com/global/business/semiconductor/products/flash/Products_Toggle_DDR_NANDFlash.html
- [8] Samsung Electronics, http://www.samsung.com/global/business/semiconductor/products/dram/Products_MobileSDRAM.html

저 자 소 개



이 두 진(학생회원)
2004년 8월 고려대학교
재료공학부 학사 졸업.
2004년 7월~현재 삼성전자
선임연구원.
2010년 3월~현재 성균관대학교
반도체디스플레이공학과
석사과정.

<주관심분야 : 반도체 Package 설계, SOC 설계, SSD 컨트롤러>



한 태 희(평생회원)-교신저자
1992년 KAIST 전기 및
전자공학과 학사.
1994년 KAIST 전기 및
전자공학과 석사.
1999년 KAIST 전기 및
전자공학과 박사.

1999년 3월~2006년 8월 삼성 전자 통신연구소
책임 연구원.

2006년 9월~2008년 2월 한국산업기술대학교
전자공학과 조교수.

2008년 3월~현재 성균관대학교 정보통신공학부
반도체시스템공학 전공 부교수.

<주관심분야 : IT SoC 설계 및 설계 방법론, 단
말 시스템, IT 융합 기술>