



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2022년02월16일
(11) 등록번호 10-2363519
(24) 등록일자 2022년02월11일

(51) 국제특허분류(Int. Cl.)

G06F 3/06 (2006.01)

(52) CPC특허분류

G06F 3/064 (2013.01)

G06F 3/0679 (2013.01)

(21) 출원번호 10-2016-0092495

(22) 출원일자 2016년07월21일

심사청구일자 2021년07월20일

(65) 공개번호 10-2017-0063332

(43) 공개일자 2017년06월08일

(30) 우선권주장

62/261,303 2015년11월30일 미국(US)

15/046,439 2016년02월17일 미국(US)

(56) 선행기술조사문헌

JP2006235960 A

(뒷면에 계속)

전체 청구항 수 : 총 20 항

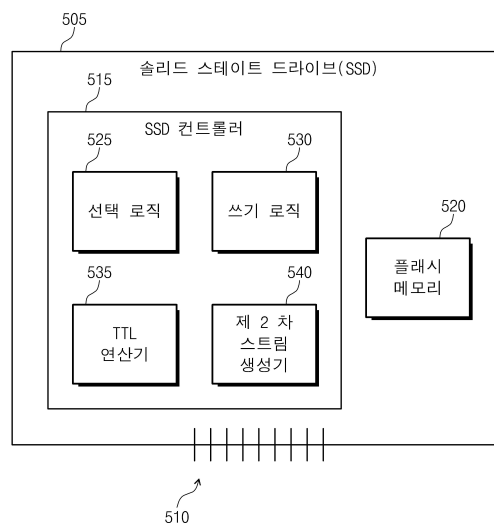
심사관 : 김중기

(54) 발명의 명칭 솔리드 스테이트 드라이브 및 그것의 동작 방법

(57) 요약

본 발명은 솔리드 스테이트 드라이브에 관한 것으로, 제 1 생존 시간을 포함하는 제 1 스트림의 데이터를 수신하고, 제 2 생존 시간을 포함하는 제 2 스트림의 데이터를 수신하는 수신 회로, 제 1 생존 시간 및 제 2 생존 시간을 기반으로 솔리드 스테이트 드라이브의 블록에 기입할 스트림을 선택하는 선택 로직 그리고 제 1 스트림의 데이터 및 제 2 스트림의 데이터를 솔리드 스테이트 드라이브의 블록에 기입하는 쓰기 로직을 포함하되, 제 2 생존 시간은 제 1 생존 시간과 다르다.

대표도 - 도5



(56) 선행기술조사문헌
W02012020544 A1
US08738882 B
US20090119352 A1
US20150293713 A1

명세서

청구범위

청구항 1

솔리드 스테이트 드라이브에 있어서:

제 1 생존 시간을 포함하는 제 1 스트림의 데이터를 수신하고, 제 2 생존 시간을 포함하는 제 2 스트림의 데이터를 수신하는 수신 회로;

상기 제 1 생존 시간을 기반으로 제 1 시간에 상기 솔리드 스테이트 드라이브의 블록에 상기 제 1 스트림의 데이터를 기입할지를 선택하고 및 상기 제 2 생존 시간을 기반으로 제 2 시간에 상기 솔리드 스테이트 드라이브의 상기 블록에 상기 제 2 스트림의 데이터를 기입할지를 선택하는 선택 로직; 및

상기 제 1 스트림의 데이터를 상기 솔리드 스테이트 드라이브의 상기 블록의 제 1 페이지에 기입하고 및 상기 제 2 스트림의 데이터를 상기 솔리드 스테이트 드라이브의 상기 블록의 제 2 페이지에 기입하는 쓰기 로직을 포함하되,

상기 제 2 생존 시간은 상기 제 1 생존 시간과 다르고,

상기 제 1 스트림의 데이터 및 상기 제 2 스트림의 데이터는 상기 블록이 가비지 콜렉션의 대상이 되기 전에 상기 블록에 기입되는 솔리드 스테이트 드라이브.

청구항 2

제 1 항에 있어서,

상기 블록에 저장된 모든 데이터는 상기 제 1 스트림 및 상기 제 2 스트림 중 하나의 스트림의 데이터만이 상기 블록에 기입되는 경우 보다 더 빨리 완료될 것으로 예상되는 솔리드 스테이트 드라이브.

청구항 3

제 1 항에 있어서,

상기 선택 로직은,

상기 제 1 스트림과 연관된 쓰기 오퍼레이션들에 대한 제 1 평균 쓰기 사이즈와 상기 제 2 스트림과 연관된 쓰기 오퍼레이션들에 대한 제 2 평균 쓰기 사이즈를 연산하는 평균 쓰기 사이즈 연산기;

상기 제 1 스트림에 대한 제 1 평균 쓰기 도달 비율과 상기 제 2 스트림에 대한 제 2 평균 쓰기 도달 비율을 연산하는 평균 쓰기 도달 비율 연산기를 포함하는 솔리드 스테이트 드라이브.

청구항 4

제 3 항에 있어서,

상기 선택 로직은 상기 제 1 생존 시간, 상기 제 2 생존 시간, 상기 제 1 평균 쓰기 사이즈, 상기 제 1 평균 쓰기 도달 비율, 상기 제 2 평균 쓰기 사이즈, 상기 제 2 평균 쓰기 도달 비율 중에서 적어도 두 개를 기반으로 상기 솔리드 스테이트 드라이브의 상기 블록에 기입할 스트림을 선택하도록 동작하는 솔리드 스테이트 드라이브.

청구항 5

제 1 항에 있어서,

제 2 차 스트림을 생성하는 제 2 차 스트림 생성기를 더 포함하되,

상기 선택 로직은 상기 제 1 스트림 및 상기 제 2 스트림 중에서 상기 제 2 차 스트림에 기입할 스트림을 선택하는 스트림 선택 로직을 포함하고, 및

상기 쓰기 로직은 상기 제 2 차 스트림을 상기 블록에 기입하도록 동작하는 솔리드 스테이트 드라이브.

청구항 6

솔리드 스테이트 드라이브에 대한 솔리드 스테이트 드라이브 컨트롤러에 있어서:

상기 솔리드 스테이트 드라이브의 블록에 대한 블록 생존 시간(a block Time-to-Live)을 위한 스토리지; 및

제 1 생존 시간을 갖는 제 1 스트림과 제 2 생존 시간을 갖는 제 2 스트림 중에서, 상기 블록 생존 시간, 상기 제 1 생존 시간 및 상기 제 2 생존 시간을 기반으로 상기 블록에 기입할 스트림을 선택하는 선택 로직을 포함하되,

상기 제 1 스트림의 데이터 및 상기 제 2 스트림의 데이터는 상기 블록이 가비지 콜렉션의 대상이 되기 전에 상기 블록에 기입되는 솔리드 스테이트 드라이브 컨트롤러.

청구항 7

제 6 항에 있어서,

상기 블록에 저장된 모든 데이터는 상기 제 1 스트림 및 상기 제 2 스트림 중 하나의 스트림의 데이터만이 상기 블록에 기입되는 경우 보다 더 빨리 완료될 것으로 예상되는 솔리드 스테이트 드라이브 컨트롤러.

청구항 8

제 6 항에 있어서,

상기 선택 로직은 상기 블록 생존 시간, 상기 제 1 생존 시간 및 상기 제 2 생존 시간을 기반으로 제 1 시간에 상기 블록에 기입할 상기 제 1 스트림을 선택하고, 상기 블록 생존 시간, 상기 제 1 생존 시간 및 상기 제 2 생존 시간을 기반으로 제 2 시간에 상기 블록에 기입할 상기 제 2 스트림을 선택하도록 동작하는 솔리드 스테이트 드라이브 컨트롤러.

청구항 9

제 6 항에 있어서,

상기 선택 로직은,

상기 제 1 스트림과 연관된 쓰기 오퍼레이션들에 대한 제 1 평균 쓰기 사이즈 및 상기 제 2 스트림과 연관된 쓰기 오퍼레이션들에 대한 제 2 평균 쓰기 사이즈를 연산하는 평균 쓰기 사이즈 연산기; 및

상기 제 1 스트림에 대한 제 1 평균 쓰기 도달 비율 및 상기 제 2 스트림에 대한 제 2 평균 쓰기 도달 비율을 연산하는 평균 쓰기 도달 비율 연산기를 포함하는 솔리드 스테이트 드라이브 컨트롤러.

청구항 10

제 9 항에 있어서,

상기 선택 로직은 상기 블록 생존 시간, 상기 제 1 생존 시간, 상기 제 2 생존 시간, 상기 제 1 평균 쓰기 사이즈, 상기 제 1 평균 쓰기 도달 비율, 상기 제 2 평균 쓰기 사이즈 및 상기 제 2 평균 쓰기 도달 비율 중 적어도 두 개를 기반으로 상기 솔리드 스테이트 드라이브의 상기 블록에 기입할 스트림을 선택하도록 동작하는 솔리드 스테이트 드라이브 컨트롤러.

청구항 11

제 6 항에 있어서,

상기 선택 로직은 상기 제 1 생존 시간을 갖는 상기 제 1 스트림, 상기 제 2 생존 시간을 갖는 상기 제 2 스트림, 및 제 3 생존 시간을 갖는 제 3 스트림 중에서, 상기 블록 생존 시간, 상기 제 1 생존 시간, 상기 제 2 생존 시간, 및 상기 제 3 생존 시간을 기반으로 상기 블록에 기입할 스트림을 선택하는 솔리드 스테이트 드라이브 컨트롤러.

청구항 12

제 6 항에 있어서,

제 2 차 스트림을 생성하는 제 2 차 스트림 생성기를 더 포함하되,

상기 선택 로직은 상기 제 1 스트림 및 상기 제 2 스트림 중에서 상기 제 2 차 스트림에 기입할 스트림을 선택하는 스트림 선택 로직을 포함하고,

상기 제 2 차 스트림은 상기 블록에 기입되는 솔리드 스테이트 드라이브 컨트롤러.

청구항 13

솔리드 스테이트 드라이브(SSD)의 블록을 식별하는 단계;

제 1 생존 시간을 포함하는 제 1 스트림을 수신하는 단계;

제 2 생존 시간을 포함하는 제 2 스트림을 수신하는 단계;

상기 제 1 생존 시간을 기반으로 제 1 시간에 상기 블록에 상기 제 1 스트림을 기입할지를 선택하는 단계;

상기 제 2 생존 시간을 기반으로 제 2 시간에 상기 블록에 상기 제 2 스트림을 기입할지를 선택하는 단계; 및

상기 제 1 스트림의 데이터를 상기 블록의 제 1 페이지에 기입하고 및 상기 제 2 스트림의 데이터를 상기 블록의 제 2 페이지에 기입하는 단계를 포함하되,

상기 제 2 생존 시간은 상기 제 1 생존 시간과 다르고,

상기 제 1 스트림의 데이터 및 상기 제 2 스트림의 데이터는 상기 블록이 가비지 콜렉션의 대상이 되기 전에 상기 블록에 기입되는 솔리드 스테이트 드라이브의 동작 방법.

청구항 14

제 13 항에 있어서,

상기 블록에 저장된 모든 데이터는 상기 제 1 스트림 및 상기 제 2 스트림 중 하나의 스트림의 데이터만이 상기 블록에 기입되는 경우 보다 더 빨리 완료될 것으로 예상되는 솔리드 스테이트 드라이브의 동작 방법.

청구항 15

제 13 항에 있어서,

상기 블록에 상기 제 1 스트림 또는 상기 제 2 스트림을 기입할지를 선택하는 단계는, 상기 제 1 생존 시간, 상기 제 2 생존 시간, 상기 제 1 스트림에 대한 제 1 평균 쓰기 사이즈, 상기 제 2 스트림에 대한 제 2 평균 쓰기 사이즈, 상기 제 1 스트림에 대한 제 1 평균 쓰기 도달 비율 그리고 상기 제 2 스트림에 대한 제 2 평균 쓰기 도달 비율 중에서 적어도 두 개를 기반으로 상기 블록에 상기 제 1 스트림 또는 상기 제 2 스트림을 기입할지를 선택하는 단계를 포함하는 솔리드 스테이트 드라이브의 동작 방법.

청구항 16

제 13 항에 있어서,

솔리드 스테이트 드라이브(SSD)의 블록을 식별하는 단계는 블록 생존 시간을 상기 블록과 연관시키는 단계를 포함하는 솔리드 스테이트 드라이브의 동작 방법.

청구항 17

제 16 항에 있어서,

상기 블록에 상기 제 1 스트림 또는 상기 제 2 스트림을 기입할지를 선택하는 단계는, 상기 제 1 생존 시간, 상기 제 2 생존 시간, 및 상기 블록 생존 시간을 기반으로 상기 블록에 상기 제 1 스트림 또는 상기 제 2 스트림을 기입할지를 선택하는 단계를 포함하는 솔리드 스테이트 드라이브의 동작 방법.

청구항 18

제 17 항에 있어서,

상기 블록에 상기 제 1 스트림 또는 상기 제 2 스트림을 기입할지를 선택하는 단계는, 상기 제 1 생존 시간 또는 상기 제 2 생존 시간이 상기 블록 생존 시간에 더 가까운지를 기반으로 상기 제 1 스트림 또는 상기 제 2 스트림을 기입할지를 선택하는 단계를 더 포함하는 솔리드 스테이트 드라이브의 동작 방법.

청구항 19

제 13 항에 있어서,

제 2 차 스트림을 생성하는 단계; 및

상기 제 2 차 스트림을 상기 블록과 연관시키는 단계를 더 포함하되,

상기 블록에 상기 제 1 스트림 또는 상기 제 2 스트림을 기입할지를 선택하는 단계는, 상기 제 1 스트림의 데이터 또는 상기 제 2 스트림의 데이터를 상기 제 2 차 스트림에 할당할지를 선택하는 단계를 포함하는 솔리드 스테이트 드라이브의 동작 방법.

청구항 20

제 13 항에 있어서,

제 3 생존 시간을 포함하는 제 3 스트림을 수신하는 단계를 더 포함하되,

상기 제 3 생존 시간은 상기 제 1 생존 시간 및 상기 제 2 생존 시간과 상이하고,

상기 블록에 상기 제 1 스트림 또는 상기 제 2 스트림을 기입할지를 선택하는 단계는, 상기 제 1 생존 시간, 상기 제 2 생존 시간 및 상기 제 3 생존 시간을 기반으로 상기 블록에 상기 제 1 스트림, 상기 제 2 스트림 또는 상기 제 3 스트림을 기입할지를 선택하는 단계를 포함하고,

상기 제 1 스트림의 데이터 및 상기 제 2 스트림의 데이터를 상기 블록에 기입하는 단계는, 상기 제 1 스트림의 데이터, 상기 제 2 스트림의 데이터 그리고 상기 제 3 스트림의 데이터를 상기 블록에 기입하는 단계를 포함하는 솔리드 스테이트 드라이브의 동작 방법.

발명의 설명

기술 분야

[0001] 본 발명은 솔리드 스테이트 드라이브 및 그것의 동작 방법에 관한 것으로, 더 구체적으로는 향상된 멀티 스트림 오퍼레이션들을 갖는 솔리드 스테이트 드라이브 및 그것의 동작 방법에 관한 것이다.

배경 기술

[0002] 오늘날 스토리지 장치들은 많은 기술들을 통합한다. 중요한 기술 중 하나는 낸드(NAND) 플래시 메모리이다. 솔리드 스테이트 드라이브(SSD)들과 같은 스토리지 장치들의 플래시 메모리는 다양한 설계 고려 사항들을 포함할 수 있다. 중요한 고려 사항들은 스토리지 용량을 효율적으로 이용하기 위한 전략들을 포함한다.

[0003] 이전에 저장된 데이터 값들이 플래시 메모리에 업데이트될 때, 데이터 값들을 저장하는 기존 페이지들은 내부적으로 무효(invalid)로 플래그(flagged)되고, 업데이트된 데이터 값들은 새로운 페이지들에 저장된다. 데이터가 삭제될 때, 데이터를 저장하는 페이지들은 유사하게 무효로 플래그된다. 무효 페이지들(invalid pages)은 가비지 컬렉션(garbage collection)으로 불리는 과정을 통해 복구된다. 가비지 컬렉션은 복수의 페이지들을 포함하는 블록 레벨(block level)에서 동작한다. 만약 하나의 블록이 완전히 비워질 때까지(예로, 블록 내 모든 페이지들이 무효로 플래그될 때까지) 가비지 컬렉션이 대기하는 경우, 솔리드 스테이트 드라이브(SSD)는 데이터가 기입될 수 있는 프리 페이지들(free pages)을 갖지 못할 수도 있다. 그러므로, 가비지 컬렉션의 수행은 삭제 대상 블록들(blocks targeted for erasure)로부터 유효 페이지들(valid pages)을 다른 블록들에 이동시킬 것을 요구한다.

[0004] 솔리드 스테이트 드라이브(SSD)들은 데이터가 도달하는 순서에 따라 도달한 데이터를 페이지들에 기입할 수 있다. 이것은 전혀 관계가 없는 파일들로부터 데이터((심지어 솔리드 스테이트 드라이브(SSD)가 멀티 클라이언트 시스템으로 사용되는 경우, 전혀 관계가 없는 클라이언트들에 의해 사용되는 데이터))가 솔리드 스테이트 드라이브(SSD)에 인접할 수도 있다. 솔리드 스테이트 드라이브(SSD)는 주위 판독 헤드(a read head around)를 이동

시키는 것과 관련되는 대기 시간(latency)을 요구하지 않기 때문에(전통적인 하드 디스크 드라이버들과는 상반됨), 솔리드 스테이트 드라이브(SSD)의 읽기 및 쓰기 액세스 시간들은 데이터 쓰기 전략에 의해 영향을 받지 않는다.

[0005] 솔리드 스테이트 드라이브(SSD)의 다른 동작들에 대한 가비지 콜렉션 동작들의 영향을 최소화하기 위한 방안에 대한 요구는 여전히 남아 있다.

발명의 내용

해결하려는 과제

[0006] 본 발명의 목적은 향상된 성능을 갖는 솔리드 스테이트 드라이브 및 그것의 동작 방법을 제공하는데 있다.

과제의 해결 수단

[0007] 본 발명의 일 실시 예에 따른 솔리드 스테이트 드라이브는 수신회로, 선택 로직, 쓰기 로직을 포함할 수 있다. 수신 회로는 제 1 생존 시간을 포함하는 제 1 스트림의 데이터를 수신하고, 제 2 생존 시간을 포함하는 제 2 스트림의 데이터를 수신한다. 선택 로직은 제 1 생존 시간 및 제 2 생존 시간을 기반으로 솔리드 스테이트 드라이브의 블록에 기입할 스트림을 선택한다. 쓰기 로직은 제 1 스트림의 데이터 및 제 2 스트림의 데이터를 솔리드 스테이트 드라이브의 블록에 기입한다. 그리고 제 2 생존 시간은 제 1 생존 시간과 다르다.

[0008] 본 발명의 일 실시 예에 따른 솔리드 스테이트 드라이브의 로직은 스토리지, 선택 로직을 포함한다. 스토리지는 솔리드 스테이트 드라이브의 블록에 대한 블록 생존 시간(a block Time-to-Live)을 위한 것이다. 선택 로직은 블록 생존 시간, 제 1 생존 시간 및 제 2 생존 시간을 기반으로 제 1 생존 시간을 갖는 제 1 스트림과 제 2 생존 시간을 갖는 제 2 스트림 중에서 블록에 기입할 스트림을 선택한다.

[0009] 본 발명의 일 실시 예에 따른 솔리드 스테이트 드라이브의 동작 방법은 솔리드 스테이트 드라이브(SSD)의 블록을 식별하는 단계, 제 1 생존 시간을 포함하는 제 1 스트림을 수신하는 단계, 제 2 생존 시간을 포함하는 제 2 스트림을 수신하는 단계, 제 1 생존 시간 및 제 2 생존 시간을 기반으로 블록에

[0010] 제 1 스트림 또는 제 2 스트림을 기입할지를 선택하는 단계 그리고 제 1 스트림의 데이터 및 제 2 스트림의 데이터를 블록에 기입하는 단계를 포함하되, 제 1 생존 시간은 제 2 생존 시간과 구별된다.

발명의 효과

[0011] 본 발명의 실시 예에 따르면, 향상된 성능을 갖는 솔리드 스테이트 드라이브 및 그것의 동작 방법이 제공된다.

도면의 간단한 설명

[0012] 도 1은 일반적인 시스템에서 생존 시간과 연관된 스트림의 데이터를 수신하는 블록을 보여준다.

도 2a 내지 도 2d는 본 발명의 일 실시 예에 따른, 다른 생존 시간들을 갖는 복수의 스트림들과 연관된 데이터를 수신하는 도 1의 블록의 일 실시 예를 보여준다.

도 3a 내지 도 3d는 다른 생존 시간들을 갖는 복수의 스트림들과 연관된 데이터를 수신하는 도 1의 블록의 다른 실시 예를 보여준다.

도 4는 제 2 차 스트림의 실시 예들을 도시한다.

도 5는 본 발명의 실시 예들에 따라, 다른 스트림들의 데이터를 도 1의 블록에 기입하는 솔리드 스테이트 드라이브를 보여준다.

도 6은 도 5의 선택 로직의 세부 사항을 보여준다.

도 7은 도 6의 평균 쓰기 사이즈 연산기의 세부 사항을 보여준다.

도 8은 도 6의 평균 쓰기 도달 비율 연산기의 세부 사항을 보여준다.

도 9는 도 6의 스트림 선택 로직의 세부 사항을 보여준다.

도 10은 도 5의 생존 시간 연산기의 세부 사항을 보여준다.

도 11은 도 5의 솔리드 스테이트 드라이브를 포함하는 머신의 세부 사항을 보여준다.

도 12는 본 발명의 일 실시 예에 따라, 다른 스트림들의 데이터를 도 1의 블록에 기입하는 도 5의 솔리드 스테이트 드라이브의 동작 과정에 대한 순서도이다.

도 13은 본 발명의 일 실시 예에 따라, 제 2 차 스트림을 이용하여 다른 스트림들의 데이터를 도 1의 블록에 기입하는 도 5의 솔리드 스테이트 드라이브의 동작 과정의 순서도를 보여준다.

발명을 실시하기 위한 구체적인 내용

- [0013] 전술한 특성 및 이하 상세한 설명은 모두 본 발명의 설명 및 이해를 돕기 위한 예시적인 사항이다. 즉, 본 발명은 이와 같은 실시 예에 한정되지 않고 다른 형태로 구체화될 수 있다. 다음 실시 형태들은 단지 본 발명을 완전히 개시하기 위한 예시이며, 본 발명이 속하는 기술 분야의 통상의 기술자들에게 본 발명을 전달하기 위한 설명이다. 따라서, 본 발명의 구성 요소들을 구현하기 위한 방법이 여럿 있는 경우에는, 이들 방법 중 특정한 것 또는 이와 동일성 있는 것 가운데 어떠한 것으로든 본 발명의 구현이 가능함을 분명히 할 필요가 있다.
- [0014] 본 명세서에서 제 1, 제 2 등과 같은 용어들은 다양한 구성 요소들을 설명하기 위해 이용될 수 있으나, 이러한 구성 요소들은 이러한 용어들에 의해 제한되지 않는다. 이러한 용어들은 오직 다른 구성 요소들로부터 하나의 구성 요소를 구별하기 위해 이용된다. 예를 들어, 본 발명의 범위에서 제 1 모듈은 제 2 모듈로 일컬어질 수 있고, 이와 유사하게 제 2 모듈은 제 1 모듈로 일컬어질 수도 있다.
- [0015] 본 명세서에서 어떤 구성이 특정 요소들을 포함한다는 언급이 있는 경우, 또는 어떤 과정이 특정 단계들을 포함한다는 언급이 있는 경우는, 그 외 다른 요소 또는 다른 단계들이 더 포함될 수 있음을 의미한다. 즉, 본 명세서에서 사용되는 용어들은 특정 실시 형태를 설명하기 위한 것일 뿐이고, 본 발명의 개념을 한정하기 위한 것이 아니다. 나아가, 발명의 이해를 돕기 위해 설명한 예시들은 그것의 상보적인 실시 예도 포함한다.
- [0016] 앞서 언급된 것과 같이, 솔리드 스테이트 드라이브(SSD, solid state drive)들은 데이터를 페이지들에 기입한다. 페이지들은 순차적으로 블록들을 구성한다. 데이터가 업데이트될 때, 기존 데이터를 저장하는 페이지들은 랜덤 액세스 메모리(RAM random access memory)로 복사되고, 수정된 후, 솔리드 스테이트 드라이브(SSD)의 프리 페이지에 기입된다. 이어, 원래 페이지는 무효(invalid)로 표시되고 새로운 페이지는 유효(valid)로 표시된다.
- [0017] 시간이 경과함에 따라, 무효 페이지들(invalid pages)의 수는 증가한다. 결국 무효 페이지들(invalid pages)이 가비지 콜렉션의 수행을 통해서 복구되지 않는다면, 솔리드 스테이트 드라이브(SSD)는 더 이상의 프리 페이지들(free pages)을 갖지 못할 것이다. 가비지 콜렉션은 대상 블록(target block, 혹은 super block)을 가져오는 것, 대상 블록 내 모든 유효 페이지들을 새로운 블록(new block)에 카피하는 것, 그리고 원래의 대상 블록을 소거하는 것을 포함한다. 이러한 유효 데이터(the valid-data)의 복사는 시간과 에너지를 요구한다. 게다가 플래시 메모리는 제한된 수의 쓰기만을 지원하기 때문에, 가비지 콜렉션은 솔리드 스테이트 드라이브(SSD)의 수명에 부정적인 영향을 끼친다.
- [0018] 플래시(Flash)의 가비지 콜렉션은 스토리지 장치의 성능(performance), 응답성(responsiveness) 및 수명(lifespan)에 상당한 영향을 끼치기 때문에, 가비지 콜렉션의 효율을 최적화하는데 도움이 되는 다양한 방법들이 존재한다. 이러한 방법들 중 하나는 멀티 스트림(Multi-stream)이라 언급되는 것으로, 이것은 컴퓨팅 시스템들(computing systems)이 데이터 기입 활동(data write activity)을 분류하는 것을 허용하는 방식이다.
- [0019] 멀티 스트리밍(Multi-streaming)은 데이터 쓰기 동작들(data write operations)을 스트림들(streams)로 합치기 위한 솔리드 스테이트 드라이브(SSD)의 동작 방법을 제공한다. 데이터 쓰기 동작은 기대(expectation)를 기반으로 복수의 스트림들 중 어느 하나의 스트림과 연관된다. 기대(expectation)는 스트림과 연관된 모든 데이터는 비슷한 생존 시간(TTL, Time-To-Live)을 갖는다는 것을 의미한다. 이것은 스토리지 장치가 좁고(narrow) 예측 가능한(predictable) 타임 프레임(timeframe) 내에서 집중적으로 데이터가 무효가 된다는 희망(hope)을 갖고 플래시 매체(Flash media) 내 데이터를 함께 저장하도록 한다. 가비지 콜렉션된 블록의 어떠한 유효 데이터(valid data)도 다른 블록에 저장될 필요가 없으므로, 성공적일 경우, 이런 배치 전략(placement strategy)은 가비지 콜렉션의 동작 집중도(the operational intensiveness)를 매우 줄여준다.
- [0020] 도 1은 일반적인 시스템에서 생존 시간과 연관된 스트림의 데이터를 수신하는 블록을 보여준다. 도 1을 참조하면, 스토리지 장치는 하나의 구체적인 스트림(stream)에 대한 복수의 쓰기 요청들을 실행한다. 실시 예로, 스토리지 장치는 우선 이용 가능한 미사용 블록을 스트림(stream)에 할당(assigning)한 후 전용(dedicating)시킴으

로써 스트림 요청들(stream requests)을 실행할 것이다. 그리고 나서, 스토리지 장치는 최하위 위치 어드레스에서 시작하여 더 상위 위치 어드레스로 진행시켜 가면서 표시 데이터 값들(presented data values)을 블록 내 제 1 이용 가능 영역(예를 들면, 페이지(들))에 기입할 것이다. 블록이 가득 찬 경우, 또 다른 블록이 스트림(stream)에 할당되고, 전용된다. 추후 블록이 가비지 콜렉션에 의해 완전히 리클레임(completely reclaimed)될 때, 같은 스트림 혹은 다른 스트림에 대한 후속 할당(subsequent assignment)을 위해 블록은 이용 가능 블록 풀(an Available-Block pool)로 반환된다.

[0021] 도 1에서, 가장 빨리 기입된 데이터 값들(the earliest-written data values) 중 일부는 삭제되거나 업데이트되어, 가장 빨리 기입된 데이터 값들의 스토리지 위치들(storage locations)이 무효(invalid)로 되게 한다. 예를 들어, 블록(103)은 스트림에 전용되었다. 스트림 쓰기들(106, 109, 112)는 페이지들(115, 118, 121)에 기입되었다. 결국, 페이지들(115, 118, 121)은 전부 무효가 되었다. 한편, 프리(free) 상태를 유지하는 다른 페이지들(136, 139, 14, 145)을 갖고, 가장 최근의 스트림 쓰기들(124, 127)이 페이지들(130, 133)에 기입되었다. 만약 페이지들(115, 118, 121)로부터의 데이터 값이 업데이트되었다면, 신규 업데이트된 값은 블록 내 더 상위 어드레스에 놓일 수 있다. 예를 들어, 상위 어드레스는 페이지들(148, 151, 130, 133)일 수 있다. 다르게는, 신규 값(new value)은 스트림에 후속으로 할당되고, 전용된 다른 블록에 놓일 수 있다.

[0022] 도 1을 참조하면, 잔여 생존 시간 값들(154, 157, 160, 163, remaining TTL values)을 갖는 유효 페이지들(148, 151, 130, 133)이 보여진다. 이러한 값들은 솔리드 스테이트 드라이브(SSD)에 실제로 저장되진 않지만, 페이지 내 데이터가 만료될 것으로 예상될 때까지의 시간을 나타낸다. 이 값은 스트림 내 데이터의 생존 시간(TTL)과 데이터가 솔리드 스테이트 드라이브(SSD)에 얼마나 오래 유지되었는지 사이의 차이를 기반으로 연산될 수 있다. 예를 들어, 생존 시간 값들(154)은 페이지(148) 내 데이터가 1분 내에 만료할 것으로 예상됨을 보여주고, 생존 시간 값(157)은 페이지(151) 내 데이터가 2분 내에 만료할 것으로 예상됨을 보여주고, 생존 시간 값(160)은 페이지(130) 내 데이터가 59분 내에 만료할 것으로 예상됨을 보여주고, 생존 시간 값(163)은 페이지(133) 내 데이터가 60 분 내에 만료할 것으로 예상됨을 보여준다. 예를 들어, 스트림이 60분의 생존 시간(TTL)을 갖는다면, 이런 상황이 발생할 것이다. 그리고 블록(103)의 하나의 페이지를 차지하는 데이터는 매 분마다 스트림(stream)에 도달한다.

[0023] 만약 블록(103)이 페이지(133)로 종료하면, 생존 시간 값(163)을 갖는 페이지(133)는 만료하는데 60분을 요구하는 만료할 데이터를 갖는 마지막 페이지일 수 있다. 그러므로, 솔리드 스테이트 드라이브(SSD)가 블록(103)에 가비지 콜렉션의 수행할 준비가 될 때, 솔리드 스테이트 드라이브(SSD)는 페이지(133) 내 데이터가 만료할 때(데이터가 쓰여진 이후로 60분)까지 대기할 것이다. 또는 솔리드 스테이트 드라이브(SSD)는 블록(103)을 지우기(erase) 위해서 블록(103)으로부터 유효 데이터(valid data)를 또 다른 블록에 복사(copy)할 것이다.

[0024] 도 1에 도시된 종래(traditional)의 멀티 스트리밍 스토리지(multi-streaming storage)에서, 단일 블록(single block)은 단일 스트림(single stream)과 연관된다. 블록(103)에 기입되는 데이터는 예측 가능한 생존 시간(TTL)을 가질 수 있다. 그러나, 블록을 채우기 위해 요구되는 시간으로 인해, 블록(103)의 페이지들은 “웨이브(wave)”에서 무효로 될 수 있다. 즉, 블록(103) 내 페이지들은 시간에 걸쳐 순차적으로 만료될 것이다. 이것은 솔리드 스테이트 드라이브(SSD)를 “어려운(difficult)” 위치(position)에 둘 것이다. 즉, 솔리드 스테이트 드라이브(SSD)는 “자유 공간(freeable space)”이었을 것을 활용하지 못하는 타임 패널티(time penalty)를 겪으면서 모든 데이터가 만료될 때까지 기다려야 한다. 또는, 솔리드 스테이트 드라이브(SSD)는 널리 알려진 가비지 콜렉션 패널티(garbage collection penalty)를 겪으면서 블록(103)에 가비지 콜렉션을 수행하기 위해 일부 유효 페이지들을 다른 블록에 복사(copy)해야만 할 것이다.

[0025] 도 1은 멀티 스트리밍 스토리지(multi-streaming storage) 없이 솔리드 스테이트 드라이브(SSD)가 블록들을 채우는 방식을 나타낸다. 솔리드 스테이트 드라이브(SSD)가 멀티 스트리밍 스토리지(multi-streaming storage) 없이 동작할 때, 각 페이지는 근본적으로 무작위 생존 시간(TTL)을 갖는다. 그리고, 블록(103)의 임의 데이터가 만료되는 시기를 예측하는 것이 불가능하다. 순차적인 생존 시간(TTL) 값들(154, 157, 160, 163)을 랜덤한 값들로 교환(exchanging)하고 그리고 그에 따른 유효 페이지들(valid pages)과 무효 페이지들(invalid pages)을 섞는 것은 종래의 비 멀티 스트리밍 스토리지(non-multi-streaming storage)를 반영하는 것이다.

[0026] 그러나 종래 멀티 스트리밍 시스템들(traditional multi-streaming systems)은 특정한 스트림들(specific streams)을 스토리지 블록들(storage blocks)에 할당하는 반면에, 스토리지 블록들(storage blocks)은 특정한 스트림들(specific streams)에 할당될 것이 필요하지 않다. 대신 전반적인 수명(overall lifetime)이 블록에 할당될 수 있다. 그리고 가변적인 생존 시간(TTL)들의 스트림(stream)들로부터의 데이터는 목적(aim)을 갖고 블

록에 기입될 수 있다. 이 경우 목적은 블록 내 데이터가 단일(single)하고 정의된(defined) 시간에 만료하는 것을 의미한다. 스토리지 장치 지능(Using storage device intelligence)과 과거의 데이터(historical data)를 이용하여, 솔리드 스테이트 드라이브(SSD)는 현재 스트림(current stream)의 데이터를 블록에 계속 기입하는 것 또는 다른 스트림(different stream)의 데이터를 블록의 잔여 페이지들(remaining pages)에 기입하는 것 중 어느 것이 더 효율적인지를 판단할 수 있다. 물론 동일한 논리가 블록의 후속 할당되는 스트림(later-assigned stream)에도 적용된다. 특히, 스토리지 장치는 제 3 스트림의 데이터를 블록에 기입하는 것이 제 2 스트림의 데이터를 블록에 기입하는 것보다 효율적인지를 그 후에 판단한다. 물론 블록에서 전환된(switched away) 스트림들의 데이터는 다른 블록들에 기입될 데이터를 포함할 수 있다.

[0027] 다른 해결 방안은 “제 2 차 스트림들(second-order streams)”을 이용하는 것이다. 앞서 언급된 스트림들은 제 1 차 스트림들(first-order streams)으로 생각될 수 있다. 즉, 각 스트림은 유사한 생존 시간(TTL) 특성을 갖는 데이터를 보유할 수 있다. 다만, 제 1 차 스트림들(first-order streams)의 데이터를 블록들에 기입하는 대신 제 1 차 스트림들(first-order streams)의 데이터는 제 2 차 스트림들(second-order streams)에 기입될 수 있다. 제 2 차 스트림들(second-order streams)은 자기 자신만의 생존 시간(TTL)을 갖는다. 그리고 제 2 차 스트림(second-order stream)에 전달된 데이터는 블록들에 기입된다. 다른 제 1 차 스트림들(first-order streams)은 제 2 차 스트림(second-order stream)의 수명이 얼마나 남았는지에 따라 데이터를 제 2 차 스트림(second-order stream)에 제공할 수 있다. 본 발명의 실시 예들에서, 블록 내에서 데이터의 더 일관적인 만료(consistent expiration)를 여전히 달성하면서, 제 2 차 스트림(second-order stream)과 솔리드 스테이트 드라이브(SSD)의 블록들 사이의 연관성(association)은 유지된다.

[0028] 도 2a 내지 도 2d는 다른 생존 시간(TTL)들을 갖는 복수의 스트림들과 연관된 데이터를 수신하는 도 1의 블록의 일 실시 예를 보여준다. 도 2a 내지 도 2d는 복수의 스트림들(multiple streams)의 데이터를 단일 블록(single block)에 기입하는 것의 이점을 보여준다. 도 2a 내지 도 2d에서, 솔리드 스테이트 드라이브(SSD)에 대한 블록들이 1MB의 크기라고 가정한다. 그리고 솔리드 스테이트 드라이브(SSD)는 두 개의 스트림들(two streams)을 수신하고 있다고 가정한다. 스트림(205)은 60분의 예측 생존 시간(expected TTL)을 갖고, 한번에 평균적으로 24KB의 데이터를 기입하고, 평균적으로 일분에 한번씩 파일을 기입한다. 스트림(210)은 45분의 예측 생존 시간(expected TTL)을 갖고, 한번에 평균 256KB의 데이터를 기입하고, 5분에 한번씩 파일을 기입한다.

[0029] 도 2a는 블록(103)에 쓰여진 모든 데이터가 단일 스트림에 연관된 상황을 다시 보여준다. 스트림(205)에 대한 데이터의 평균적인 파일 크기는 24KB이고, 1분에 하나의 파일의 평균 쓰기 도착률(average write arrival rate)로 주어질 때, 블록(103)의 1MB 용량을 채우는데 대략 45분이 걸린다. 그러므로 블록(103)이 채워진 이후, 블록(103)에 기입되는 첫 번째 데이터는 15분의 잔여 생존 시간 값(215, remaining TTL value)을 갖는다. 이와 달리 블록(103)에 기입되는 마지막 데이터(last data)는 60분의 잔여 생존 시간 값(220)을 갖는다. 그리고 블록(103)은 총 1시간 45분 동안의 유효 데이터를 갖게 될 것이다. (마지막 데이터가 기입될 때까지 45분 더하기 마지막 데이터의 생존 시간으로 1시간)

[0030] 만약 블록에 마지막 데이터(last data)가 추가된 이후로 1시간 이전에 대상 블록(target block)이 임의의 시점에 가비지 콜렉션을 수행하도록 명령 받는다면, 가비지 콜렉션 동안 다른 블록으로 이동(transfer)을 요구하는 블록 내 유효 데이터가 여전히 존재하게 된다. 게다가, 솔리드 스테이트 드라이브(SSD)가 블록에 가비지 콜렉션을 더 일찍 수행할수록, 솔리드 스테이트 드라이브(SSD)의 다른 블록으로 재위치(relocated)시켜야 할 더 많은 유효 데이터(valid data)가 남아 있게 된다.

[0031] 대신 도 2b를 참조하면, 스트림(205)은 오직 22분 동안 블록(103)에 데이터를 기입한다. (여기서는, 블록(103)은 절반만 채워진다) 0:22 시점에서, 기입되는 첫 번째 데이터는 38분의 잔여 생존 시간 값(215)을 갖는다. 그리고 기입되는 마지막 데이터(last data)는 60분의 잔여 생존 시간 값(225)을 갖는다.

[0032] 도 2c에 도시된 것처럼, 0:22 시점 이후 블록(103)은 스트림(210)과 연결한다. 0:27 시점에서, 45분의 생존 시간 값(230)을 갖는 데이터를 추가하면서, 스트림(210)은 데이터를 블록(103)에 기입한다. 생존 시간 값들(215, 225)은 첫 데이터를 기입하는 스트림(210)을 위해 대기(waiting)하는 5분만큼 감소하는 것에 유의해야 한다.

[0033] 마지막으로 0:32에서, 스트림(210)은 제 2 파일을 블록(103)에 기입한다. 주어진 스트림(210)에 대한 평균 쓰기 사이즈(average write size)와 쓰기 도달 비율(write arrival rate)에 따라, 이러한 제 2 파일은 블록(103)에 전부 기입된다. 이 시점에서, 스트림(205)로부터 기입된 데이터는 변경된 잔여 생존 시간(remaining TTL)들을 갖는다. 잔여 생존 시간(remaining TTL)들은 생존 시간 값(215)은 28분 그리고 생존 시간 값(225)은 50분으로 변화한다. 그리고 스트림(210)으로부터 기입된 데이터는 각각 40분 및 45분의 생존 시간 값들(230, 235)

찾는다. 0:32 시점에서 블록(103)은 완전하게 채워졌고, 그리고 가장 긴 잔여 생존 시간(225)을 갖는 데이터는 50분 뒤에 만료될 것으로 기대되기 때문에, 블록(103)은 도 2a에서 발생한 시점보다 23분이 빠른 1:22 시점에서 가비지 콜렉션을 수행할 가능성이 높다. 게다가, 더 많은 데이터가 동시에 만료될 가능성이 있기 때문에, 모든 데이터가 만료되기 전에 블록(103)은 가비지 콜렉션을 위한 바람직하지 않은 후보가 될 것이다.

- [0034] 심지어 더 많은 스트림들, 그리고 필 레이트(fill rate), 생존 시간(TTL) 및 데이터 크기(data size)를 위한 더 많은 수의 옵션들과 함께, 스트림들의 페이지들이 모두 한꺼번에 “무효화(invalidate)” 되는 방식으로 블록들을 채우는 스트림들을 전환(switch)하는 능력이 향상되는 것에 유의해야 한다. 각각이 임의의 생존 시간(TTL), 평균 쓰기 사이즈 및 평균 쓰기 도달 비율을 갖는 임의의 수의 스트림들이 있을 수 있다.
- [0035] 도 3a 내지 도 3d는 도 1의 블록(103)이 다른 생존 시간들을 갖는 복수의 스트림들과 연관된 데이터를 수신하는 다른 실시 예를 보여준다. 도 3a 내지 도 3d에서, 스트림들간의 주요 차이점(primary difference)은 생존 시간(TTL)이기 때문에, 동일한 데이터 필 레이트(fill rate)와 동일한 평균 쓰기 사이즈(average write size)를 갖는 단지 4개의 스트림(stream)이 있다고 가정한다. 도 3a 내지 도 3d에 도시된 4개의 스트림들은 각각 60분, 45분, 30분 그리고 15분의 생존 시간(TTL) 값들을 갖는다.
- [0036] 도 3a에서, 스트림(305)은 60분의 생존 시간(240, TTL)을 갖는 블록(103)에 기입하고 있다. 스트림(305)은 60분의 생존 시간(310)을 갖는다. 그리고 15분 이내로 60분의 가장 긴 잔여 생존 시간(315)을 갖는 데이터를 갖고 블록(103)의 1/4을 채우도록 충분한 데이터가 기입된다. 이 시점에서, 도 3b에 도시된 것과 같이 솔리드 스테이트 드라이브(SSD)는 스트림(305)을 스트림(320)으로 전환한다. 스트림(320)은 45분의 생존 시간(325, TTL)을 갖는다. 다시, 15분 이내로 블록(103)의 다른 1/4을 채우도록 스트림(320)으로부터 충분한 데이터가 기입된다. 이 시점에서, 블록(103)은 45분 이내로 만료될 것이 예상되는 데이터(315, 330)로 절반이 채워진다.
- [0037] 도 3c에 도시된 것과 같이 솔리드 스테이트 드라이브(SSD)는 스트림(335)로 전환할 수 있다. 스트림(335)은 30분의 생존 시간(340, TTL)을 갖는다. 다시 15분 이내로 스트림(335)은 블록(103)의 또 다른 1/4를 채우기 위해 충분한 데이터를 기입할 수 있다. 이 경우 블록(103)은 잔여 생존 시간(315, 330, 345)가 30분인 데이터로 3/4이 채워진다. 이 시점에서, 도 3d에 도시된 것과 같이 솔리드 스테이트 드라이브(SSD)는 스트림(350)으로 전환한다. 스트림(350)은 15분의 생존 시간(355, TTL)을 갖는다. 또 다른 15분 이후로 스트림(350)은 블록(103)을 완전히 채우기에 충분한 데이터를 기입한다. 그리고 블록(103)의 데이터(315, 330, 345, 360)는 15분 내 만료할 것으로 기대된다. 그러므로, 도 3a의 스트림(305)으로부터 처음 데이터가 블록(103)에 기입된 이후로 1:15 시점에서, 블록(103)의 모든 데이터는 만료될 것으로 기대된다. 그리고 전체 블록은 가비지 콜렉션 될 수 있다.
- [0038] 종래 시스템들과 비교하면, 도 2b 내지 도 2d와 도 3a 내지 도 3d의 예들은 향상된 가비지 콜렉션 성능을 제공한다. 멀티 스트리밍 스토리지(multi-streaming storage)를 이용함으로써, 블록은 감소하는 생존 시간(TTL) 값들을 갖는 데이터들로 순차적으로(sequentially) 채워진다. 블록들의 데이터는 전반적으로 더 빨리 만료될 것이다. 그리고 더 많은 데이터가 동시에 만료하는 경향을 갖는다. 그 결과, 가비지 콜렉션이 수행될 때, 블록에 대해 가비지 콜렉션을 수행하기 전에 모든 유효 데이터를 다른 블록에 복사할 필요가 없이 블록들은 전부 만료 데이터를 더 많이 갖게 될 수 있다. 게다가, 더 많은 데이터가 동시에 만료될 경향을 갖기 때문에, 블록의 모든 데이터가 만료되기 전에 블록은 가비지 콜렉션을 위해 덜 선택될 수 있다.
- [0039] 도 2b 내지 도 2d와 도 3a 내지 도 3d를 비교하면, 선택할 수 있는 스트림들이 더 많이 있을 때, 블록의 모든 데이터가 만료될 때까지의 시간이 덜 소요된다. 예를 들어, 도 2b 내지 도 2d에서, 블록의 모든 데이터는 1:32 시점에서 만료될 것으로 기대된다. 도 3a 내지 도 3d에서, 블록의 모든 데이터는 1:15 시점에서 만료될 것으로 기대된다. 블록의 데이터는 또한 대략 비슷한 시점에 만료될 데이터를 가지는 더 많은 경향을 갖고, 블록이 여전히 유효 데이터를 갖는 동안에 가비지 콜렉션을 위해 블록이 더 적게 선택되도록 한다.
- [0040] 도 2a 내지 도 2d에서, 2개의 스트림들이 블록(103)에 데이터를 기입하기 위해 설명되고 이용된다. 도 3a 내지 도 3d에서, 4개의 스트림들이 블록(103)에 데이터를 기입하기 위해 설명되고 이용된다. 그러나 이러한 예들은 블록(103)에 데이터를 기입하기 위해 이용되는 모든 스트림들을 보여주나, 본 발명의 다른 실시 예는 오직 이용 가능한 스트림들의 서브세트(subset)만을 이용할 수 있다. 예를 들어, 도 2a 내지 도 3d에서, 솔리드 스테이트 드라이브(SSD)로 기입되는 20개의 스트림들이 있을 수 있고, 이 중 몇몇의 스트림들만이 블록(103)에 기입될 수 있다. 그리고 다양한 스트림들은 제한 없이 다른 생존 시간(TTL)들, 쓰기 사이즈들(write sizes) 및 쓰기 도달 비율들(write arrival rates)을 포함할 수 있다.
- [0041] 최적의 가비지 콜렉션을 위해 한 블록이 얼마나 일찍 준비가 되는지 대한 하한(lower limit)이 존재한다. 블록

(103)에 기입된 각 페이지는 시작된(originated) 스트림에 따라 각각의 생존 시간(TTL)을 갖는다. 최악의 경우, 블록은 해당 블록을 가득 채우는 시간과 블록에 기입된 데이터 중 가장 큰 생존 시간만큼을 더한 시간 동안 유효 데이터(valid data)를 가질 수 있다. 더욱 정확하게는 블록(103)은 일부 유효 데이터를 $\max(\text{time written}(\text{page}) + \text{TTL}(\text{page}))$ 까지 보유할 것이다. 빨리 기입된 페이지는 더 긴 생존 시간(TTL)을 가질 수 있고, 그리고 빨리 기입된 페이지에 대한 만료 시간(expiration time)이 더 짧은 생존 시간(TTL)을 갖는 늦게 기입된 페이지에 대한 만료 시간보다 늦어질 수 있는 경우가 발생 할 수 있다는 것에 유의 하여야 한다. 그러나 빈번하게, 블록 내 모든 데이터에 대한 만료 시간은 블록에 기입된 마지막 데이터에 대한 만료 시간이 될 것이다. 그러므로, 만약 하나의 블록을 완전히 채우는데 한 시간이 걸리고 마지막에 기입된 데이터가 15분의 생존 시간(TTL)을 갖는 경우, 블록 내 모든 데이터는 대략 1:15 시간에 만료될 것으로 예상될 것이다.

- [0042] 본 발명의 다른 실시 예에서, 모든 1차 스트림들(first-order streams)은 앞서 도시된 것처럼 분석될 것이다. 그러나 블록들에 직접 기입하는 것을 대신하여 제 1 차 스트림들(the first-order streams)은 제 2 차 스트림들(time-limited second-order streams)의 잔여 시간을 기반으로 시간 제약을 받는 제 2 차 스트림들(time-limited second-order streams)에 기입하게 된다.
- [0043] 도 4는 제 2 차 스트림의 일 실시 예를 도시한다. 간단한 경우로서, 도 3a 내지 도 3d에서와 동일한 1차 스트림들을 가정한다.
- [0044] 스트림(305)은 1시간의 생존 시간(TTL)을 갖는다.
- [0045] 스트림(320)은 45분의 생존 시간(TTL)을 갖는다.
- [0046] 스트림(335)은 30분의 생존 시간(TTL)을 갖는다.
- [0047] 스트림(350)은 15분의 생존 시간(TTL)을 갖는다.
- [0048] 또한, 1시간의 생존 시간(410, TTL)을 갖는 단일한 제 2 차 스트림(405, a single second-order stream)을 가정한다.
- [0049] 제 2 차 스트림(405)이 생성될 때, 스트림(305)으로 부터 데이터가 처음 할당된다. 제 2 차 스트림이 시간이 지나서 수명(lifespan)이 45분에 도달하면, 제 2 차 스트림(405)은 스트림(305)으로 부터 데이터를 수신하는 것을 중지하고 스트림(320)으로 부터 데이터를 수신하기 시작한다. 마찬가지로, 제 2 차 스트림(405)의 수명이 30분에 도달하면, 스트림(335)으로 부터 데이터를 수신한다. 그리고 수명이 단지 15분이 남게 되면, 스트림(350)으로 부터 데이터를 수신한다.
- [0050] 이러한 접근 방식 하에서, 제 2 차 스트림(405)에 할당된 블록(103)은 모든 데이터가 약 15분 내에 무효화(invalidate)되도록 할 것이다(이는 제 1차 스트림들의 증설 단위(granularity)인 것에 유의). 전체 블록의 최대 수명은 대략 1:15가 될 것이다. 이 예에서, 꼭 그럴 필요는 없지만 이러한 동작은 앞서 도시된 도 3a 내지 도 3d의 실시 예와 유사하다는 것에 유의하여야 한다. 데이터 필 레이트(data fill rate)들이 데이터의 수명 또는 블록의 잔여 크기(remaining size)에 비해 더 높은 예들에서, 제 2 차 스트림의 실시에는 더 많은 유연성을 제공할 수 있다. 만약 하나의 블록이 만료되기 전에 제 2 차 스트림이 블록을 채우면, 제 2 차 스트림은 새로운 블록으로 이동될 수 있다(남은 생존 시간을 갖고서). 한편, 기존 블록(old block)은 여전히 대략 같은 시기에 만료하는 모든 데이터를 갖게 될 것이다(제 1차 스트림들의 증설 단위(granularity)내에서). 즉, 제 2 차 스트림 메커니즘에서는, 수명의 끝에 다가서는 블록을 채우는 낮은 수명(low-lifespan) 또는 빠른 필 레이트(fast-fill rate)의 스트림들을 찾기 위한 시스템의 부담(strain)은 존재하지 않는다. 만료하는 제 2 차 스트림(expiring second-order stream)에 의해 채워지지 않은 임의의 블록은 이미 블록에 저장된 데이터의 잔여 생존 시간(remaining TTL)과 유사한 수명을 갖는 새로운 스트림(new stream)에 의해 채워질 수 있다는 것에 또한 유의하여야 한다.
- [0051] 어느 방법에서든, 더 많은 제 1 차 스트림들을 가질수록, 더 작은 시간 증설 단위들이 얻어질 수 있다. (페이지들이 서로의 더 인접한 시간들 내에서 무효화되는 경향을 갖게 된다)
- [0052] 앞서 보여진 제 1 차의 실시 예들과 같이, 많은 제 2 차 스트림들은 동시에 동작할 수 있다. 제 1 차 스트림들의 속성들에 따라 다른 2 차 스트림들은 다른 수명들(different lifespans)을 갖도록 생성될 수 있다.
- [0053] 어느 방법에서든, 솔리드 스테이트 드라이브(SSD)가 다른 스트림으로 전환될 때(이는 블록에 기입하는 다른 제

1 차 스트림 혹은 제 2 차 스트림에 할당된 다른 제 1 차 스트림일 수 있다), 솔리드 스테이트 드라이브(SSD)는 새로운 스트림을 선택하기 위해 임의의 바람직한 알고리즘을 이용할 수 있다. 새로운 제 1 차 스트림을 선택하기 위한 하나의 알고리즘은 블록 또는 제 2 차 스트림에 대한 잔여 생존 시간(remaining TTL)보다 크면서 동시에 가장 작은 생존 시간(TTL)을 갖는 스트림을 선택하는 것이다.(또는, 만약 블록 또는 제 2 차 스트림에 대한 잔여 생존 시간보다 큰 생존 시간을 갖는 스트림이 존재하지 않는다면, 가장 큰 생존 시간을 선택한다) 그러므로, 도 3a로 되돌아가면, 블록(103)은 60분의 블록 생존 시간(240)을 갖기 때문에, 스트림(305)이 60분의 생존 시간(310)을 갖도록 하는 것이 최고의 선택이다. 도 3b에서, 블록 생존 시간(240)이 45분으로 떨어질 때, 알고리즘을 이용하여 스트림(320)이 45분의 생존 시간(325)을 갖도록 하는 것이 최고의 선택이 된다. 도 3c에서, 블록 생존 시간(240)이 30분으로 떨어질 때, 알고리즘을 이용하여 스트림(335)이 30분의 생존 시간(340)을 갖도록 하는 것이 최고의 선택이 되며, 도 3d에서도 동일한 방식으로 선택된다. 도 4의 제 2 차 스트림(405)과 연관된 제 1 차 스트림을 선택할 때, 같은 전략이 적용될 수 있다.

[0054] 다른 실시 예로, 솔리드 스테이트 드라이브(SSD)는 블록에 대한 잔여 생존 시간(remaining TTL)에 가장 가까운 생존 시간(TTL)을 갖는 제 1 차 스트림을 선택할 수 있다. 그러므로, 도 3a로 되돌아가면, 60분의 생존 시간(310)을 갖는 스트림(305)은 60분의 블록 생존 시간(240)에 가장 가깝다. 그리고 스트림(305)은 0:07.5 시간까지 사용하기 위해 유지된다. 그 시간 이후로, 45분의 생존 시간(325)을 갖는 스트림(320)은 남아 있는 52 $\frac{1}{2}$ 분의 블록 생존 시간(240)에 가장 가깝다. 그리고 스트림(320)은 0:22.5 시간까지 유지된다. 0:22.5 시간에서, 30분의 생존 시간(340)을 갖는 스트림(335)은 남아 있는 37 $\frac{1}{2}$ 분의 블록 생존 시간(240)에 가장 가깝다. 그 다음도 동일한 방식으로 진행된다.

[0055] 도 5는 본 발명의 실시 예들에 따라 다른 스트림들로부터 데이터를 도 1의 블록에 기입하는 솔리드 스테이트 드라이브(SSD)를 보여준다. 도 5에서, 솔리드 스테이트 드라이브(505)가 보여진다. 솔리드 스테이트 드라이브(505)는 회로(510)를 포함한다. 회로(510)는 정보를 전송하고 수신하기 위해 이용된다. 예를 들어, 정보는 오퍼레이션들(operations) 또는 데이터일 수 있다. 솔리드 스테이트 드라이브(505)는 솔리드 스테이트 드라이브 컨트롤러(515, 이하 SSD 컨트롤러)와 플래시 메모리(520)를 포함한다. SSD 컨트롤러(515)는 솔리드 스테이트 드라이브(505)의 동작을 제어할 수 있다. 플래시 메모리(520)는 데이터를 저장할 수 있다. (즉, 플래시 메모리(520)는 블록들 중에서 도 1의 블록(103)을 저장할 수 있다)

[0056] SSD 컨트롤러(515)는 다른 구성 요소들 중에서 선택 로직(525), 쓰기 로직(530) 그리고 생존 시간 계산기(535)를 포함할 수 있다. 선택 로직(525)은 도 1의 블록(103)에 기입하기 위해 이용되는 스트림을 선택할 수 있다. 또는 선택 로직(525)은 도 4의 제 2 차 스트림(405)을 할당하기 위해 이용되는 스트림을 선택할 수 있다(도 4의 제 2 차 스트림(405)이 이용되는 경우). 쓰기 로직(530)은 데이터를 플래시 메모리(520)에 기입할 수 있다(아마도 도 4의 제 2 차 스트림(405)의 명령하에서). 더 구체적으로, 쓰기 로직(530)은 데이터를 플래시 메모리(520)의 페이지에 기입할 수 있다. 예를 들어, 플래시 메모리(520)의 페이지는 도 1의 블록(103)의 페이지일 수 있다. 생존 시간 계산기(535)는 스트림 또는 블록에 대한 생존 시간(TTL)을 계산할 수 있다. 이 경우 스트림은 도 2a 내지 도 2d의 생존 시간들(245, 250) 그리고 도 3a 내지 도 3d의 생존 시간들(310, 325, 340, 355)일 수 있다. 이 경우 블록은 도 2a 내지 도 3d의 블록 생존 시간(240)일 수 있다.

[0057] 제 2 차 스트림들을 이용한 본 발명의 실시 예들에서, SSD 컨트롤러(515)는 제 2 차 스트림 생성기(540)를 또한 포함할 수 있다. 제 2 차 스트림 생성기(540)는 도 4의 블록(103)과 연관될 수 있는 도 4의 제 2 차 스트림(405)을 생성할 수 있다.

[0058] 도 6은 도 5의 선택 로직(525)의 세부 사항을 보여준다. 도 6에서, 선택 로직(525)은 비교기(605), 평균 쓰기 사이즈 연산기(610), 평균 쓰기 도달 비율 연산기(615), 스트림 선택 로직(620) 및 스토리지(625)를 포함할 수 있다. 스토리지(625)는 블록 생존 시간(240) 및/또는 제 2 차 스트림 생존 시간(410)을 저장할 수 있다. 그러나 블록 생존 시간(240)과 제 2 차 스트림 생존 시간(410)은 선택 로직(525)의 외부에 저장될 수도 있다. 본 발명의 일 실시 예에 따른 비교기(605)는 블록 생존 시간(240) 또는 제 2 차 스트림 생존 시간(410)을 이용해 도 2a 내지 도 2d의 생존 시간들(245, 250)(도 2a 내지 도 2d의 스트림들(205, 210)에 대한)과 도 3a 내지 도 3d의 생존 시간들(310, 325, 340, 355)(도 3a 내지 도 3d의 스트림들(305, 320, 335, 350)에 대한)을 비교할 수 있다. 비교(comparison)를 기반으로, 본 발명의 일 실시 예에 따른 스트림 선택 로직(620)은 어떤 스트림이 도 1의 블록(103)에 기입할지 또는 어떤 스트림이 도 4의 제 2 차 스트림(405)에 할당될지를 선택할 수 있다.

[0059] 선택 로직(525)은 임의의 원하는 시간에 동작할 수 있다. 예를 들어, 선택 로직(525)은 주기적으로(예를 들어, 5분마다) 이용될 수 있다. 또는, 새로운 쓰기 오퍼레이션(new write operation)이 도 5의 솔리드 스테이트 드라이브

이브(SSD)에 전달될 때마다, 선택 로직(525)이 이용될 수 있다. 또는, 블록 생존 시간(240) 또는 제 2 차 스트림 생존 시간(410)이 도 2a 내지 도 2d의 생존 시간(245, 250) 또는 도 3a 내지 도 3d의 생존 시간들(310, 325, 340, 355)보다 낮아지게 될 때마다, 선택 로직(525)이 이용될 수 있다. 이 경우 도 2a 내지 도 2d의 생존 시간(245, 250) 또는 도 3a 내지 도 3d의 생존 시간들(310, 325, 340, 355)은 도 1의 블록(103) 또는 도 4의 제 2 차 스트림에 현재 기입 중인 스트림의 생존시간들을 의미한다. 선택 로직(525)은 다른 정보를 반영할 수 있다. 예를 들어, 다른 정보는 하나의 스트림을 선택할 때, 각 스트림의 평균 쓰기 사이즈(average write size) 및 평균 쓰기 도달 비율(average write arrival rate)일 수 있다. 예를 들어, 도 2a 내지 도 3d에 도시된 60분의 생존 시간(240)을 갖는 블록(103)을 다시 고려한다. 그리고 도 2a 내지 도 3d의 블록(103)이 60 페이지들을 포함한다고 가정한다. 60분의 생존 시간(TTL)을 갖고 1분에 한 페이지를 기입하는 스트림은 블록(103)이 120분 동안 유효 데이터(valid data)를 보유할 수 있음을 의미한다. (마지막 페이지가 기입될 때까지 60분 그리고 페이지가 만료할 때까지 60분) 그러나 만약 이용 가능한 유일한 다른 스트림이 15분의 생존 시간(TTL)을 갖고 하나의 페이지를 5분 마다 기입하고 그리고 선택 로직(525)이 0:45 시간에 그러한 다른 스트림으로 전환된다면, 블록(103)을 채우는데 120분의 시간이 소요될 것이다. 나아가, 블록(103)의 마지막 데이터가 만료되기 전 15분이 더해진다. 그러므로, 평균 쓰기 사이즈(average write size)와 평균 쓰기 도달 비율(average write arrival rate)은 스트림들을 변화시킬 최적의 시간에 영향을 미칠 수 있다. 또한, 선택 로직(525)을 이용하기 위한 다른 스케줄들이 제한 없이 이용될 수 있다.

[0060] 도 7은 도 6의 평균 쓰기 사이즈 연산기(610)의 세부 사항을 보여준다. 도 7에서, 평균 쓰기 사이즈 연산기(610)는 주어진 스트림의 쓰기 오퍼레이션들(106, 109, 112, 124, 127)에 대한 정보를 수신할 수 있다. 그리고 평균 쓰기 사이즈 연산기(610)는 이러한 쓰기들에 대한 평균 쓰기 사이즈(705)를 연산할 수 있다. (즉, 쓰여진 데이터 양의 합 나누기 수행된 쓰기 오퍼레이션들의 개수를 의미한다) 앞서 설명된 것과 같이, 평균 쓰기 사이즈 연산기(610)는 스트림에 대한 이용 가능한 모든 정보, 또는 가장 최근의 n개의 쓰기 오퍼레이션들, 또는 스트림에 대한 마지막 n개의 쓰기 오퍼레이션들 중 가장 빠른 k개, 또는 마지막 t분 동안 발생한 스트림에 대한 쓰기 오퍼레이션들, 또는 스트림에 대한 쓰기 오퍼레이션들을 선택하기 위해 바람직한 임의의 접근 방식을 이용할 수 있다.

[0061] 도 8은 도 6의 평균 쓰기 도달 비율 연산기(615)의 세부 사항을 보여준다. 도 8에 있어서, 평균 쓰기 도달 비율 연산기(615)는 주어진 스트림에 대해 쓰기 오퍼레이션들(106, 109, 112, 124, 127)에 대한 정보를 수신할 수 있다. 그리고 평균 쓰기 도달 비율 연산기(615)는 이러한 쓰기들에 대한 평균 쓰기 도달 비율(805)을 연산할 수 있다. (즉, 첫 쓰기 오퍼레이션과 마지막 쓰기 오퍼레이션 사이의 시간의 양 나누기 그 간격(interval)에서 수행된 쓰기 오퍼레이션들의 개수를 의미한다) 앞서 설명된 것과 같이, 평균 쓰기 도달 비율 연산기(615)는 스트림에 대한 모든 이용 가능한 정보, 또는 스트림에 대한 가장 최근의 n개의 쓰기 오퍼레이션들, 또는 스트림에 대한 마지막의 n개의 쓰기 오퍼레이션들 중 가장 빠른 k개, 또는 마지막 t분 안에 발생한 스트림에 대한 쓰기 오퍼레이션, 또는 스트림에 대한 쓰기 오퍼레이션들을 선택하는 임의의 다른 바람직한 접근 방법을 이용할 수 있다.

[0062] 도 9는 도 6의 스트림 선택 로직(620)의 세부 사항을 보여준다. 도 9에서, 스트림 선택 로직(620)은 다양한 정보를 수신할 수 있다. 예를 들어, 다양한 정보는 도 2a 내지 도 3d의 블록 생존 시간(240), 도 2a 내지 도 2d의 스트림들(205, 210)로부터 도 2a 내지 도 2d의 생존 시간들(245, 250), 도 3a 내지 도 3d의 스트림들(305, 320, 335, 350)로부터 도 3a 내지 도 3d의 생존 시간들(310, 325, 340, 355), 각 스트림에 대한 평균 쓰기 사이즈(705), 그리고 각 스트림에 대한 평균 쓰기 도달 비율(805)일 수 있다. 스트림 선택 로직(620)은 도 2b 내지 도 3d의 블록(103)에 기입할 스트림(905)을 선택하기 위해 이러한 정보를 이용할 수 있다. 스트림 선택 로직(620)은 도 2b 내지 도 3d의 블록(103) 또는 도 4의 제 2 차 스트림(405)에 기입할 스트림(905)을 선택하기 위한 임의의 바람직한 전략을 이용할 수 있다. 예를 들어, 스트림 선택 로직(620)은 도 2a 내지 도 3d의 블록(240)에 대한 잔여 생존 시간(remaining TTL)보다는 작으면서 동시에 그 중 가장 높은 생존 시간(TTL)을 갖는 스트림을 선택할 수 있다. 또는, 스트림 선택 로직(620)은 앞서 도 2a 내지 도 2d를 참조하여 예를 든 것처럼, 데이터 만료를 최적화하기 위해 스트림들을 전환하는 시점을 연산할 수 있다. 또한 스트림 선택 로직(620)은 다른 선택 전략들을 이용할 수 있다. 앞서 설명된 것처럼, 스트림 선택 로직(620)은 스트림을 선택하기 위해 제공된 모든 정보, 또는 그 정보 중 일부를 이용하여 스트림(905)을 선택할 수 있다. 본 발명의 일 실시 예에 따르면, 만약 스트림 선택 로직(620)이 도 9에 도시된 모든 정보를 이용하지 않는다면, 스트림 선택 로직(620)은 도 9에 도시된 모든 정보를 제공받을 필요가 없다.

[0063] 도 10은 도 5의 생존 시간 연산기(535)의 세부 사항을 보여준다. 도 10에서, 생존 시간 연산기(535)는 주어진

스트림의 쓰기 오퍼레이션들(106, 109, 112, 124, 127)에 대한 정보를 수신할 수 있다. 그리고 생존 시간 연산기(535)는 이러한 스트림들에 대한 생존 시간(310, 325, 340, 355)을 연산할 수 있다. (즉, 데이터가 쓰여진 시기와 그 데이터가 변경되거나 삭제된 시기 사이의 지속 시간을 측정하고, 그리고 나서 이러한 지속 시간들을 합하고 그리고 영향받은 페이지들의 개수로 나눔으로써) 앞서 설명된 바와 같이, 생존 시간 연산기(535)는 스트림에 대한 모든 이용 가능한 정보, 또는 스트림에 대한 가장 최근의 n개의 쓰기 오퍼레이션들, 또는 스트림에 대한 마지막 n개의 쓰기 오퍼레이션들 중 가장 앞선 k개, 또는 마지막 t분에 발생한 스트림에 대한 쓰기 오퍼레이션들, 또는 스트림에 대한 쓰기 오퍼레이션들을 선택하기 위한 임의의 다른 바람직한 접근 방식을 이용할 수 있다.

[0064] 도 11은 도 5의 솔리드 스테이트 드라이브(SSD)를 포함하는 머신의 세부 사항을 보여준다. 도 11을 참조하면, 머신(machine) 또는 머신들(1105)은 적어도 하나의 프로세서들(1110)을 포함한다. 적어도 하나의 프로세서들(1110)은 메모리 컨트롤러(1115), 클락(1120)을 포함한다. 적어도 하나의 프로세서들(1110)은 머신(machine) 또는 머신들(1105)의 구성 요소들의 오퍼레이션들을 조정(coordinate)하기 위해 이용될 수 있다. 예시적으로 프로세서들(1110)은 랜덤 액세스 메모리(RAM), 읽기 전용 메모리(ROM) 또는 다른 상태 보존 매체(other state preserving media)를 포함하는 메모리(1125)에 연결될 수 있다. 프로세서들(1110)은 스토리지 장치들(505)과 네트워크 커넥터(1130)와 연결될 수 있다. 예를 들어, 네트워크 커넥터(1130)는 이더넷 커넥터(Ethernet connector)일 수 있다. 프로세서들(1110)은 또한 버스(1135)에 연결될 수 있다. 버스(1135)는 다른 구성 요소들 중에서도, 유저 인터페이스(1140)와 입출력 엔진(1145)을 이용하여 관리될 수 있는 입출력 인터페이스 포트들에 연결될 수 있다.

[0065] 도 12는 본 발명의 일 실시 예에 따라 다른 스트림들로부터 데이터를 도 1의 블록에 기입하는 도 5의 솔리드 스테이트 드라이브(SSD)의 동작 과정에 대한 순서도이다. 도 12를 참조하면, S1205 단계에서, 도 5의 솔리드 스테이트 드라이브(SSD, 505)는 도 1의 블록(103)을 식별할 수 있다. S1210 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 도 2a 내지 도 3d의 블록 생존 시간(240)을 도 1의 블록(103)과 연관시킬 수 있다. S1215 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 도 2a 내지 도 2d의 스트림들(205, 210)과 도 3a 내지 도 3d의 스트림들(305, 320, 335, 350)을 수신할 수 있다. S1220 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 도 1의 블록(103)에 기입할 스트림(905)을 선택할 수 있다. S1225 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 선택된 스트림(905)으로부터 데이터를 도 1의 블록(103)에 기입할 수 있다. 점선(1230)으로 도시된 것과 같이, 도 12의 순서도는 더 늦은 시점에 도 5의 솔리드 스테이트 드라이브(SSD)가 도 1의 블록(103)에 기입할 다른 스트림을 선택하는 것이 가능하도록 선택적으로 S1220 단계로 되돌아 갈 수 있다.

[0066] 도 13은 본 발명의 일 실시 예에 따라 제 2 차 스트림을 이용하여 다른 스트림들의 데이터를 도 1의 블록에 기입하는 도 5의 솔리드 스테이트 드라이브(SSD)의 동작 과정의 순서도를 보여준다. 도 13을 참조하면, S1305 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 도 1의 블록(103)을 식별할 수 있다. S1310 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 도 4의 제 2 차 스트림(405)을 생성할 수 있다. S1315 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 도 4의 생존 시간(410)을 도 4의 제 2 차 스트림(405)에 할당할 수 있다. S1320 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 도 2a 내지 도 2d의 스트림들(205, 210)을 수신하거나 도 3a 내지 도 3d의 스트림들(305, 320, 335, 350)을 수신할 수 있다. S1325 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 도 4의 제 2 차 스트림(405)에 기입할 스트림(905)을 선택할 수 있다. S1330 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 선택된 스트림(905)으로부터 데이터를 도 4의 제 2 차 스트림(405)에 기입할 수 있다. S1335 단계에서, 도 5의 솔리드 스테이트 드라이브(505)는 도 4의 제 2 차 스트림(405)으로부터 데이터를 도 1의 블록(103)에 기입할 수 있다. 점선(1340)으로 도시된 것과 같이, 도 13의 순서도는 더 늦은 시점에 도 5의 솔리드 스테이트 드라이브(505)가 도 4의 제 2 차 스트림(405)에 기입할 다른 스트림을 선택하는 것이 가능하도록 선택적으로 S1325 단계로 되돌아 갈 수 있다. 도 12 및 도 13에서, 본 발명의 사상에 따른 일부 실시예들이 나타나 있다. 그러나 당업자는 단계들의 순서를 변경하고, 단계들을 제외하고, 또는 도면들에서 보이지 않는 링크들 포함 시킴으로써 발명의 사상에 따른 다른 실시예들이 가능하다는 것을 인식 할 수 있을 것이다. 순서도의 모든 이러한 변경들은 명시적인 설명 여부에 관계없이 본 발명의 사상에 따른 실시예들로 고려되어야 한다.

[0067] 하기의 설명은 본 발명의 개념의 특정한 관점들이 구현될 수 있는 적합한 머신(machine) 또는 머신들(machines)의 간략하고 일반적인 설명을 제공하기 위한 것이다. 머신(machine) 또는 머신들(machines)의 적어도 일부는 다른 장치들로부터 수신된 지시들(directives)뿐만 아니라 키보드들, 마우스들과 같은 통상적인 입력 장치들로부터의 입력(input)에 의해 제어될 수 있다. 장치 또는 장치들의 적어도 일부는 가상 현실(VR, virtual

reality) 환경과 상호 작용, 바이오 피드백(bio feedback) 또는 다른 입력 신호들에 의해서도 제어될 수 있다. 본 명세서에서 사용된 용어인 “머신(machine)”은 단일 머신(single machine), 가상 머신(virtual machine) 또는 통신 접속 머신들의 시스템(a system of communicatively coupled machines), 가상 머신들(virtual machine) 또는 함께 동작하는 장치들을 넓게 포함하는 것으로 의도된다. 예시적인 머신들은 컴퓨팅 장치들(computing devices) 및 수송 장치들(transportation devices)을 포함한다. 예를 들어, 컴퓨팅 장치들은 퍼스널 컴퓨터들(personal computers), 워크 스테이션들(workstations), 서버들, 휴대용 컴퓨터들, 휴대용 장치들, 휴대폰, 태블릿들(tablets) 등과 같을 수 있다. 예를 들어, 수송 장치들은 개인 또는 대중 교통 수단(예를 들어, 자동차, 기차, 택시 등)일 수 있다.

[0068] 머신(machine) 또는 머신들(machines)은 임베디드 컨트롤러들(embedded controllers)을 포함할 수 있다. 예를 들어, 임베디드 컨트롤러들은 프로그램 가능한(또는 프로그램 불가능한) 로직 디바이스들 또는 어레이들, 주문형 반도체들(ASICs, Application Specific Integrated Circuits), 임베디드 컴퓨터들(embedded computers), 스마트 카드들(smart cards) 및 이와 유사한 것들일 수 있다. 네트워크 인터페이스(network interface), 모뎀 또는 다른 통신 연결(communicative coupling)을 통해, 머신(machine) 또는 머신들(machines)은 하나 이상의 원격 머신들(remote machines)에 대해 하나 이상의 접속들(connections)을 이용할 수 있다. 머신들(machines)은 인터넷, 근거리 네트워크들(LAN, local area networks), 광역 네트워크들(WAN, wide area networks) 등과 같은 물리적인 및/또는 논리적인 네트워크를 통하여 상호 접속될 수 있다. 통상의 기술자는 네트워크 통신이 다양한 유선 및/또는 무선 근거리 또는 장거리 캐리어들(carriers) 및 프로토콜들을 이용할 수 있음을 이해할 수 있을 것이다. 이 경우 다양한 유선 및/또는 무선 근거리 또는 장거리 캐리어들 및 프로토콜들은 무선 주파수(radio frequency), 위성(satellite), 마이크로파(microwave), 미국 전기 전자 학회(IEEE) 802.11, 블루투스(bluetooth)등일 수 있다.

[0069] 본 발명에 따른 실시 예들은 기능들(functions), 절차들(procedures), 데이터 구조들(data structures), 응용 프로그램들(application programs) 등을 포함하는 연관 데이터(the associated data)를 함께 참조하여 설명될 수 있다. 이 경우 연관 데이터(the associated data)는 머신에 의해 액세스될 때 태스크들(tasks)을 수행하거나 추상 데이터 형식들(abstract data types) 또는 로우 레벨의 하드웨어 컨텍스트들(low-level hardware contexts)을 정의하는 머신 구동의 결과에 따른 것이다. 예를 들어, 연관 데이터(associated data)는 휘발성 및/또는 불휘발성 메모리(예로, RAM, ROM 등) 또는 다른 스토리지 장치들 및 연관 저장 매체(associated storage media)에 저장될 수 있다. 예를 들어, 연관 저장 매체(associated storage media)는 하드 드라이브들(hard-drives), 플로피 디스크들(floppy disks), 광학 저장 장치(optical storage), 테이프(tapes), 플래시 메모리(flash memory), 메모리 스틱(memory sticks), 디지털 비디오 디스크들(digital video disks), 생물학적 저장 장치(biological storage) 등을 포함할 수 있다. 연관 데이터(associated data)는 전송 환경들(transmission environments)을 통해 전달될 수 있다. 예를 들어, 전송 환경들은 패킷 형식(the form of packets), 직렬 데이터 형식(the form of serial data), 병렬 데이터 형식(the form of parallel data, 전파 신호 형식(the form of propagated signals) 등을 갖는 물리 및/또는 논리 네트워크를 포함한다. 그리고 연관 데이터(associated data)는 압축(compressed)되거나 암호화(encrypted)된 포맷(format)으로 사용될 수 있다. 연관 데이터(associated data)는 분산 환경(distributed environment)에서 사용될 수 있고, 머신 액세스(machine access)를 위해 로컬적으로 및/또는 원격적으로 저장될 수 있다.

[0070] 본 발명의 실시 예들은 하나 이상의 프로세서들에 의해 실행되는 지시어들(instructions)을 포함하는 유형의(tangible) 비일시적인(non-transitory) 머신-판독 가능 매체(machine-readable medium)를 포함할 수 있다. 이러한 지시어들은 본 명세서에서 설명된 본 발명의 구성요소들을 수행하기 위한 지시어들을 포함한다.

[0071] 도시된 실시 예들을 참조하여 본 발명의 개념의 원리를 설명하고 예시하였다. 도시된 실시 예들은 이러한 원리들로부터 벗어나지 않고 배열 및 세부 사항에 있어서 변형될 수 있고, 임의의 방식으로 조합될 수 있음은 이해될 것이다. 그리고, 비록 전술한 논의는 특정한 실시 예들에 집중한 것이지만, 다른 구성들이 고려될 수 있다. 특히, 이러한 “본 발명의 일 실시 예에 있어서” 또는 이와 유사한 표현이 본 명세서에서 사용되었으나, 이러한 문구들은 일반적으로 실시 가능성(embodiment possibilities)을 참조하기 위한 것이고, 본 발명의 개념을 특정 한 실시 예의 구성들에 한정하기 위한 것이 아니다. 본 명세서에서 사용된 것과 같이, 이러한 용어들은 다른 실시 예들에 결합될 수 있는 동일한 또는 다른 실시 예들을 참조할 수 있다.

[0072] 전술한 실시 예들은 본 발명의 개념을 제한하는 것으로 해석되지 않는다. 비록 몇몇의 실시 예들이 설명되었으나, 이러한 통상의 기술자들은 이러한 실시 예들에 대해 실질적으로 신규한 교시(novel teaching) 및 본 발명의 이점들(advantages)로부터 벗어나지 않으면서 많은 변형들(many modifications)이 가능하다는 것을 쉽게 이

해할 것이다. 따라서, 이러한 모든 변형들은 청구 범위들에 기재된 본 발명의 개념의 범위 내에 포함되는 것으로 의도된다.

- [0074] 본 발명의 실시 예들은 제한(limitation) 없이 하기의 구절들(statements)로 확장할 수 있다.
- [0075] 제 1 구절로, 본 발명의 일 실시 예에 따른 솔리드 스테이트 드라이브는 제 1 생존 시간을 포함하는 제 1 스트림으로부터 데이터를 수신하고, 제 2 생존 시간을 포함하는 제 2 스트림으로부터 데이터를 수신하는 수신 회로, 제 1 생존 시간 및 제 2 생존 시간을 기반으로 솔리드 스테이트 드라이브의 블록에 기입할 스트림을 선택하는 선택 로직 및 솔리드 스테이트 드라이브의 블록에 제 1 스트림 및 제 2 스트림으로부터 데이터를 기입하는 쓰기 로직을 포함하되, 제 2 생존 시간은 제 1 생존 시간과 다르다.
- [0076] 제 2 구절로, 본 발명의 일 실시 예는 제 1 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 제 1 스트림 및 제 2 스트림 중 어느 하나의 데이터만이 블록에 기입되는 경우, 블록에 저장된 데이터는 만료될 것으로 예상된다.
- [0077] 제 3 구절로, 본 발명의 일 실시 예는 제 1 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 제 1 시간에 제 1 생존 시간 및 제 2 생존 시간을 기반으로 상기 블록에 기입할 상기 제 1 스트림을 선택하고, 제 2 시간에 상기 제 1 생존 시간 및 상기 제 2 생존 시간을 기반으로 상기 블록에 기입할 상기 제 2 스트림을 선택하도록 동작한다.
- [0078] 제 4 구절로, 본 발명의 일 실시 예는 제 1 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 선택 로직은 제 1 생존 시간 및 제 2 생존 시간을 기반으로 솔리드 스테이트 드라이브의 블록에 기입할 스트림을 주기적으로 선택하도록 동작한다.
- [0079] 제 5 구절로, 본 발명의 일 실시 예는 제 1 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 수신 회로가 제 1 스트림 또는 제 2 스트림으로부터 데이터를 수신할 때, 선택 로직은 제 1 생존 시간 및 제 2 생존 시간을 기반으로 솔리드 스테이트 드라이브의 블록에 기입할 스트림을 선택하도록 동작한다.
- [0080] 제 6 구절로, 본 발명의 일 실시 예는 제 1 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 선택 로직은 제 1 생존 시간 및 제 2 생존 시간을 블록과 연관된 블록 생존 시간과 비교하는 비교기를 포함한다.
- [0081] 제 7 구절로, 본 발명의 일 실시 예는 제 1 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 선택 로직은 제 1 스트림과 연관된 쓰기 동작들에 대한 제 1 평균 쓰기 사이즈와 제 2 스트림과 연관된 쓰기 동작들에 대한 제 2 평균 쓰기 사이즈를 연산하는 평균 기입 사이즈 연산기, 제 1 스트림에 대한 제 1 평균 쓰기 도달 비율 및 제 2 스트림에 대한 제 2 평균 쓰기 도달 비율을 연산하는 평균 기입 도달 비율 연산기를 포함한다.
- [0082] 제 8 구절로, 본 발명의 일 실시 예는 제 7 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 제 1 생존 시간, 제 2 생존 시간, 제 1 평균 쓰기 사이즈, 제 1 평균 쓰기 도달 비율, 제 2 평균 쓰기 사이즈, 제 2 평균 쓰기 도달 비율 중에서 적어도 두 개를 기반으로 솔리드 스테이트 드라이브의 블록에 기입할 스트림을 선택한다.
- [0083] 제 9 구절로, 본 발명의 일 실시 예는 제 7 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 제 1 스트림에 대한 제 1 생존 시간과 제 2 스트림에 대한 제 2 생존 시간을 연산하는 생존 시간 연산기를 더 포함한다.
- [0084] 제 10 구절로, 본 발명의 일 실시 예는 제 7 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 제 2 차 스트림을 생성하는 제 2 차 스트림 생성기를 더 포함하고, 선택 로직은 상기 제 1 스트림 및 상기 제 2 스트림 중에서 제 2 차 스트림에 기입할 스트림을 선택하는 스트림 선택 로직을 포함하고, 쓰기 로직은 블록에 제 2 차 스트림을 기입하도록 동작한다.
- [0085] 제 11 구절로, 본 발명의 일 실시 예는 제 1 구절에 따라 솔리드 스테이트 드라이브(SSD)를 포함하되, 제 1 스트림, 제 2 스트림 그리고 제 3 생존 시간을 포함하는 제 3 스트림으로부터 데이터를 수신하는 수신 회로, 제 1 생존 시간, 제 2 생존 시간 및 제 3 생존 시간을 기반으로 솔리드 스테이트 드라이브의 블록에 기입할 스트림을 선택하도록 동작하는 선택 로직 및 솔리드 스테이트 드라이브의 블록에 제 1 스트림, 제 2 스트림 그리고 제 3 스트림으로부터 데이터를 기입하는 쓰기 로직을 포함한다.
- [0086] 제 12 구절로, 솔리드 스테이트 드라이브에 대한 로직에 있어서, 솔리드 스테이트 드라이브의 블록에 대한 블록 생존 시간(a block Time-to-Live)을 위한 스토리지 및 블록 생존 시간, 제 1 생존 시간 및 제 2 생존 시간을 기반으로 제 1 생존 시간을 갖는 제 1 스트림과 제 2 생존 시간을 갖는 제 2 스트림 중에서 블록에 기입할 스트림

을 선택하는 선택 로직을 포함한다.

- [0087] 제 13 구절로, 본 발명의 일 실시 예는 제 12 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 제 1 스트림 및 제 2 스트림 중 오직 하나의 스트림으로부터 데이터가 블록에 기입되는 경우, 블록에 저장된 데이터는 만료될 것으로 예상된다.
- [0088] 제 14 구절로, 본 발명의 일 실시 예는 제 12 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 선택 로직은 제 1 시간에 블록 생존 시간, 제 1 생존 시간 그리고 제 2 생존 시간을 기반으로 제 1 스트림을 선택하도록 동작하고, 제 2 시간에 블록 생존 시간, 제 1 생존 시간 그리고 제 2 생존 시간을 기반으로 제 1 스트림을 선택하도록 동작한다.
- [0089] 제 15 구절로, 본 발명의 일 실시 예는 제 12 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 선택 로직은 블록 생존 시간, 제 1 생존 시간 및 제 2 생존 시간을 기반으로 솔리드 스테이트 드라이브의 블록에 기입할 스트림을 주기적으로 선택하도록 동작한다.
- [0090] 제 16 구절로, 본 발명의 일 실시 예는 제 12 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 솔리드 스테이트 드라이브(SSD)가 제 1 스트림 또는 제 2 스트림으로부터 데이터를 수신할 때, 선택 로직은 블록 생존 시간, 제 1 생존 시간 및 제 2 생존 시간을 기반으로 솔리드 스테이트 드라이브(SSD)의 블록에 어떤 스트림을 기입할지 선택하도록 동작한다.
- [0091] 제 17 구절로, 본 발명의 일 실시 예는 제 12 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 선택 로직은 제 1 생존 시간 및 제 2 생존 시간을 블록 생존 시간과 비교하는 비교기를 포함한다.
- [0092] 제 18 구절로, 본 발명의 일 실시 예는 제 12 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 선택 로직은 제 1 스트림과 연관된 쓰기 오퍼레이션들에 대한 제 1 평균 쓰기 사이즈 및 제 2 스트림과 연관된 쓰기 오퍼레이션들에 대한 제 2 평균 쓰기 사이즈를 연산하는 평균 쓰기 사이즈 연산기 및 제 1 스트림에 대한 제 1 평균 쓰기 도달 비율 및 제 2 스트림에 대한 제 2 평균 쓰기 도달 비율을 연산하는 평균 쓰기 도달 비율 연산기를 포함한다.
- [0093] 제 19 구절로, 본 발명의 일 실시 예는 제 18 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 선택 로직은 블록 생존 시간, 제 1 생존 시간 및 제 2 생존 시간, 제 1 평균 쓰기 사이즈, 제 1 평균 쓰기 도달 비율, 제 2 평균 쓰기 사이즈 및 제 2 평균 쓰기 도달 비율 중 적어도 두 개를 기반으로 상기 솔리드 스테이트 드라이브의 블록에 기입할 스트림을 선택하도록 동작한다.
- [0094] 제 20 구절로, 본 발명의 일 실시 예는 제 12 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 제 1 스트림에 대한 제 1 생존 시간과 제 2 스트림에 대한 제 2 생존 시간을 연산하는 생존 시간 연산기를 더 포함한다.
- [0095] 제 21 구절로, 본 발명의 일 실시 예는 제 12 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 로직은 제 2 차 스트림을 생성하는 제 2 차 스트림 생성기를 더 포함하고, 선택 로직은 제 1 스트림 및 제 2 스트림 중에서 제 2 차 스트림에 기입할 스트림을 선택하는 스트림 선택 로직을 포함하고, 제 2 차 스트림은 블록에 기입된다.
- [0096] 제 22 구절로, 본 발명의 일 실시 예는 제 12 구절에 따라 솔리드 스테이트 드라이브에 대한 로직을 포함하되, 선택 로직은 블록 생존 시간, 제 1 생존 시간, 제 2 생존 시간 그리고 제 3 생존 시간을 기반으로 제 1 생존 시간을 갖는 제 1 스트림, 제 2 생존 시간을 갖는 제 2 스트림 그리고 제 3 생존 시간을 갖는 제 3 스트림 중에서 블록에 기입할 스트림을 선택하도록 동작한다.
- [0097] 제 23 구절로, 본 발명의 일 실시 예는 동작 방법을 포함하고, 본 발명의 동작 방법은 솔리드 스테이트 드라이브의 블록을 식별하는 단계, 제1 생존 시간을 포함하는 제 1 스트림을 수신하는 단계, 제 2 생존 시간을 포함하는 제 2 스트림을 수신하는 단계, 제 1 생존 시간 및 제 2 생존 시간을 기반으로 블록에 제 1 스트림 또는 상기 제 2 스트림을 기입할지를 선택하는 단계 그리고 그리고 제 1 스트림과 제 2 스트림 모두로부터 데이터를 블록에 기입하는 단계를 포함하되, 제 1 생존 시간은 제 2 생존 시간과 다르다.
- [0098] 제 24 구절로, 본 발명의 일 실시 예는 제 23 구절에 따른 동작 방법을 포함하고, 제 1 스트림 및 제 2 스트림 중 오직 하나의 스트림의 데이터가 블록에 기입되는 경우, 블록에 저장된 데이터는 곧 만료될 것으로 예상된다.
- [0099] 제 25 구절로, 본 발명의 일 실시 예는 제 23 구절에 따른 동작 방법을 포함하고, 블록에 제 1 스트림 또는 제

제 2 스트림을 기입할지 선택하는 단계는, 제 1 생존 시간 및 제 2 생존 시간을 기반으로 제 1 시간에 제 1 스트림을 블록에 기입하도록 선택하는 단계 및 제 1 생존 시간 및 제 2 생존 시간을 기반으로 제 2 시간에 제 2 스트림을 블록에 기입하도록 선택하는 단계를 포함한다.

[0100] 제 26 구절로, 본 발명의 일 실시 예는 제 23 구절에 따른 동작 방법을 포함하고, 블록에 제 1 스트림 또는 제 2 스트림을 기입할지 선택하는 단계는,

[0101] 제 1 생존 시간 및 제 2 생존 시간을 기반으로 제 1 스트림 또는 제 2 스트림을 블록에 기입할지 여부를 주기적으로 선택하는 단계를 포함한다.

[0102] 제 27 구절로, 본 발명의 일 실시 예는 제 23 구절에 따른 동작 방법을 포함하고, 블록에 제 1 스트림 또는 제 2 스트림을 기입할지 선택하는 단계는 제 1 스트림 또는 제 2 스트림의 데이터가 솔리드 스테이트 드라이브에 기입될 때마다, 제 1 생존 시간 또는 제 2 생존 시간을 기반으로 제 1 스트림 또는 제 2 스트림을 기입할지 여부를 선택하는 단계를 포함한다.

[0103] 제 28 구절로, 본 발명의 일 실시 예는 제 23 구절에 따른 동작 방법을 포함하고, 블록에 제 1 스트림 또는 제 2 스트림을 기입할지 선택하는 단계는,

[0104] 제 1 생존 시간, 제 2 생존 시간, 제 1 스트림에 대한 제 1 평균 쓰기 사이즈, 제 2 스트림에 대한 제 2 평균 쓰기 사이즈, 제 1 스트림에 대한 제 1 평균 쓰기 도달 비율 그리고 제 2 스트림에 대한 제 2 평균 쓰기 도달 비율 중 적어도 두 개를 기반으로 블록에 제 1 스트림 또는 제 2 스트림을 기입할지를 선택하는 단계를 포함한다.

[0105] 제 29 구절로, 본 발명의 일 실시 예는 제 23 구절에 따른 동작 방법을 포함하고, 솔리드 스테이트 드라이브(SSD)의 블록을 식별하는 단계는 블록 생존 시간을 블록과 연관시키는 단계를 포함한다.

[0106] 제 30 구절로, 본 발명의 일 실시 예는 제 29 구절에 따른 동작 방법을 포함하고, 블록에 제 1 스트림 또는 제 2 스트림을 기입할지 선택하는 단계는, 제 1 생존 시간, 제 2 생존 시간 그리고 블록 생존 시간을 기반으로 블록에 제 1 스트림 또는 제 2 스트림을 기입할지를 선택하는 단계를 포함한다.

[0107] 제 31 구절로, 본 발명의 일 실시 예는 제 30 구절에 따른 동작 방법을 포함하고, 블록에 제 1 스트림 또는 제 2 스트림을 기입할지 선택하는 단계는, 제 1 생존 시간 또는 제 2 생존 시간 중 블록 생존 시간과 더 가까운 생존 시간을 기반으로 블록에 제 1 스트림 또는 제 2 스트림을 기입할지를 선택하는 단계를 더 포함한다.

[0108] 제 32 구절로, 본 발명의 일 실시 예는 제 23 구절에 따른 동작 방법을 포함하고, 동작 방법은, 제 2 차 스트림을 생성하는 단계 그리고 제 2 차 스트림과 블록을 연관시키는 단계를 포함하되, 블록에 제 1 차 스트림 또는 제 2 차 스트림을 기입할지 선택하는 단계는 제 1 스트림의 데이터 또는 제 2 스트림의 데이터 중 제 2 차 스트림에 할당할 데이터를 선택하는 단계를 포함한다.

[0109] 제 33 구절로, 본 발명의 일 실시 예는 제 23 구절에 따른 동작 방법을 포함하고, 동작 방법은, 제 3 생존 시간을 포함하는 제 3 스트림을 수신하는 단계를 더 포함하되, 제 3 생존 시간은 제 1 생존 시간 및 제 2 생존 시간과 상이하고, 블록에 제 1 스트림 또는 제 2 스트림을 기입할지를 선택하는 단계는 제 1 생존 시간, 제 2 생존 시간 그리고 제 3 생존 시간을 기반으로 제 1 스트림, 제 2 스트림 또는 제 3 스트림 중 블록에 기입할 스트림을 선택하는 단계를 포함하고, 제 1 스트림 및 제 2 스트림의 데이터를 블록에 기입하는 단계는 제 1 스트림의 데이터, 제 2 스트림의 데이터 그리고 제 3 스트림의 데이터를 블록에 기입하는 단계를 포함한다.

[0110] 본 발명의 상세한 설명에서는 구체적인 실시 예에 관하여 설명하였으나, 본 발명의 범위에서 벗어나지 않는 한도 내에서 여러 가지 변형이 가능하다. 그러므로, 본 발명의 범위는 상술한 실시 예에 국한되어 정해져서는 안 되며 후술하는 특허청구범위뿐만 아니라 이 발명의 특허청구범위와 균등한 것들에 의해 정해져야 한다.

부호의 설명

[0111] 103: 블록

106, 109, 112, 124, 127: 쓰기 오퍼레이션들

115, 118, 121: 무효 페이지들

148, 151, 130, 133: 유효 페이지들

136, 139, 14, 145: 무효 페이지들

240: 블록 TTL

410: 제 2 차 스트림 생존 시간

505: 솔리드 스테이트 드라이브

510: 회로

515: SSD 컨트롤러

520: 플래시 메모리

525: 선택 로직

530: 쓰기 로직

535: 생존 시간 계산기

540: 제 2 차 스트림 생성기

605: 비교기

610: 평균 쓰기 사이즈 연산기

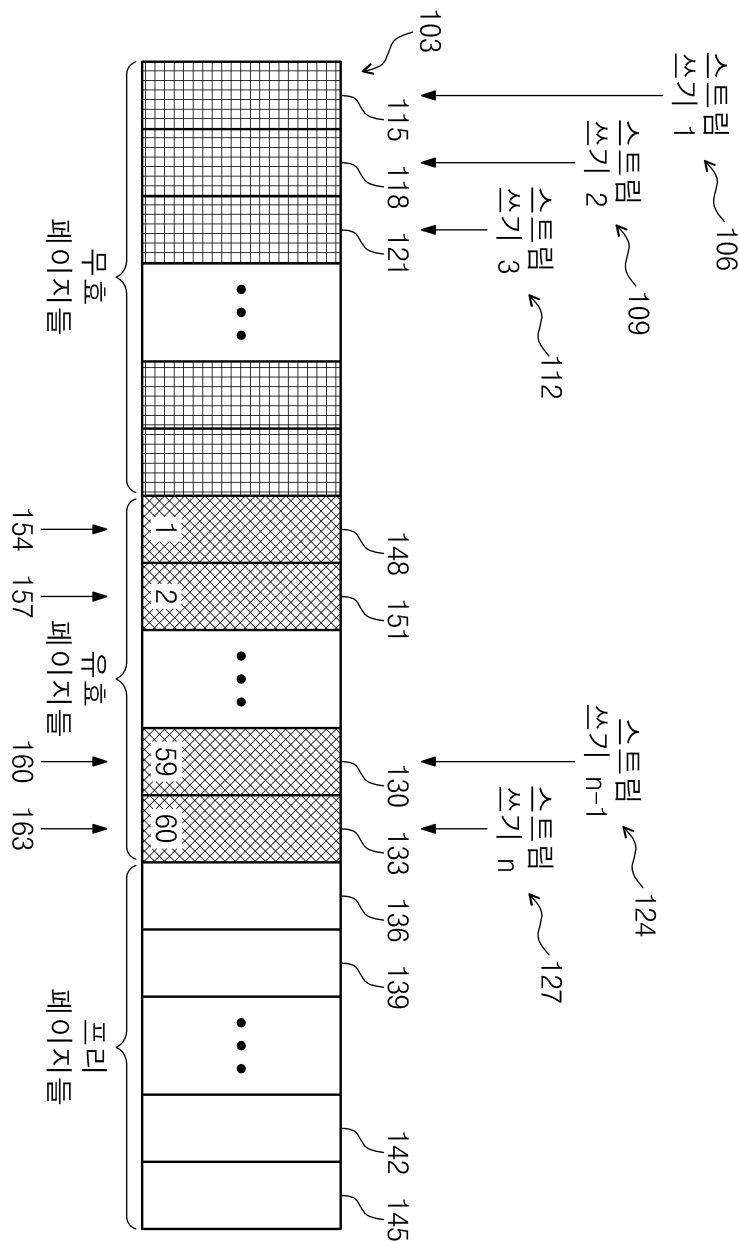
615: 평균 쓰기 도달 비율 연산기

620: 스트림 선택 로직

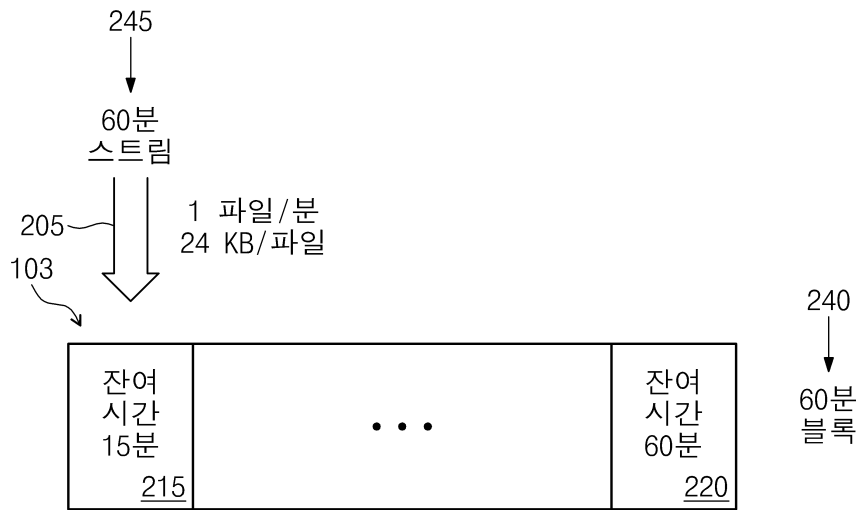
625: 스토리지

도면

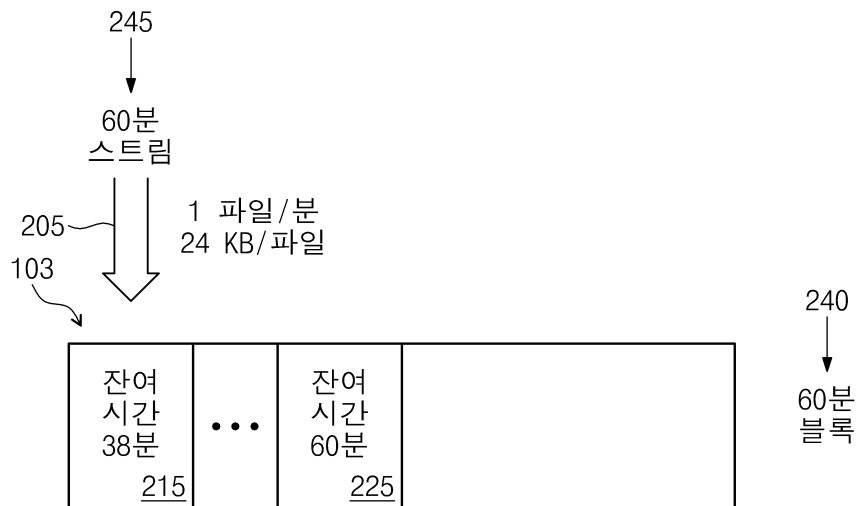
도면1



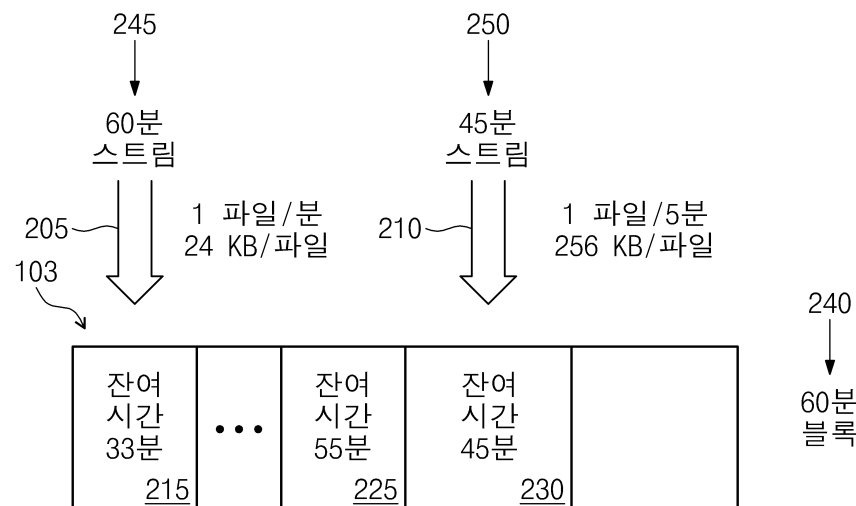
도면2a



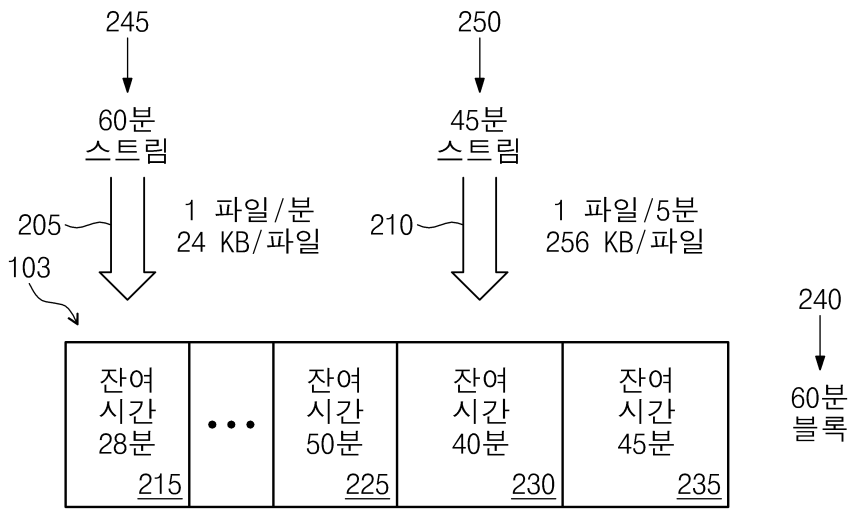
도면2b



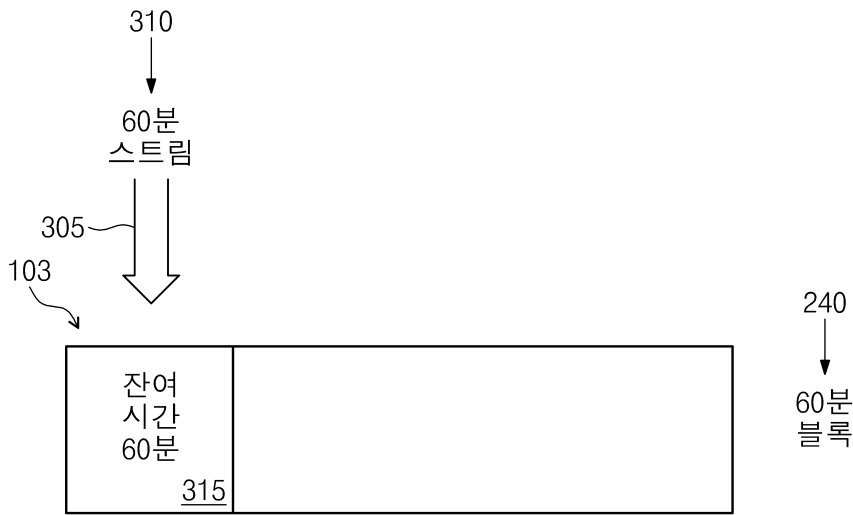
도면2c



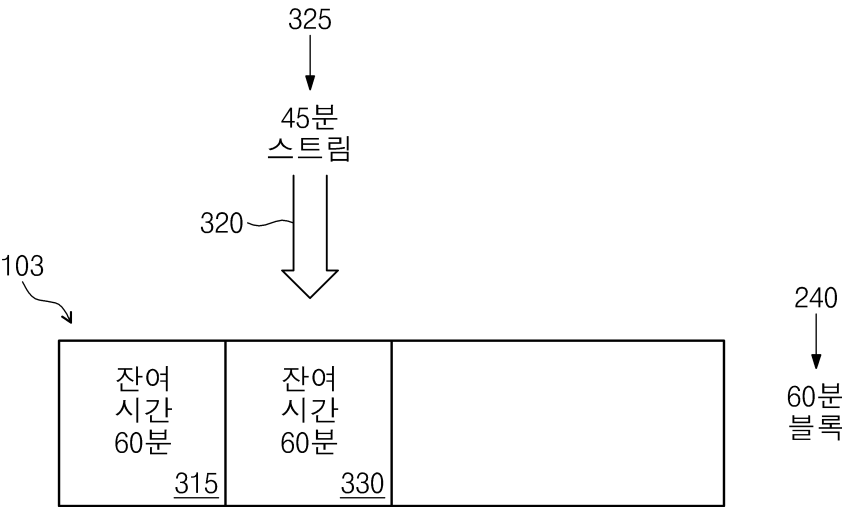
도면2d



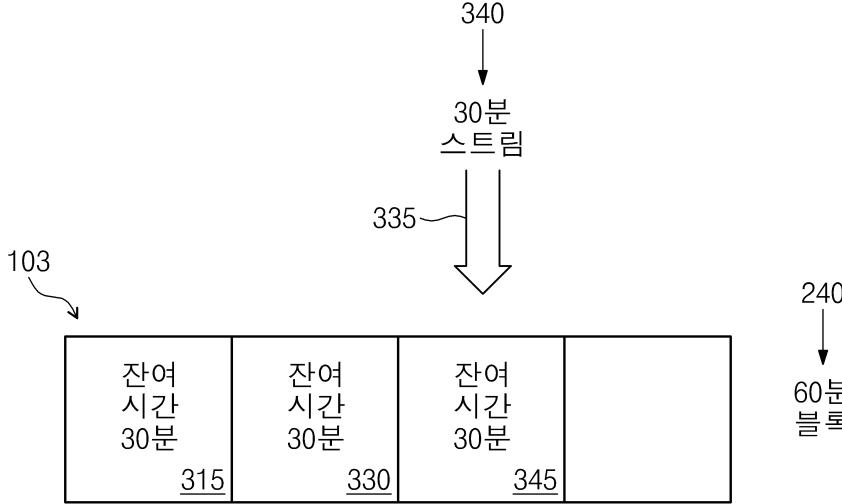
도면3a



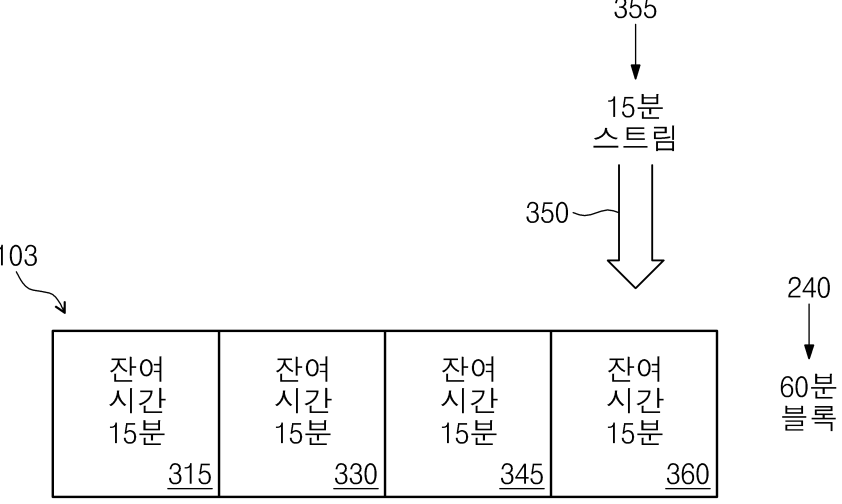
도면3b



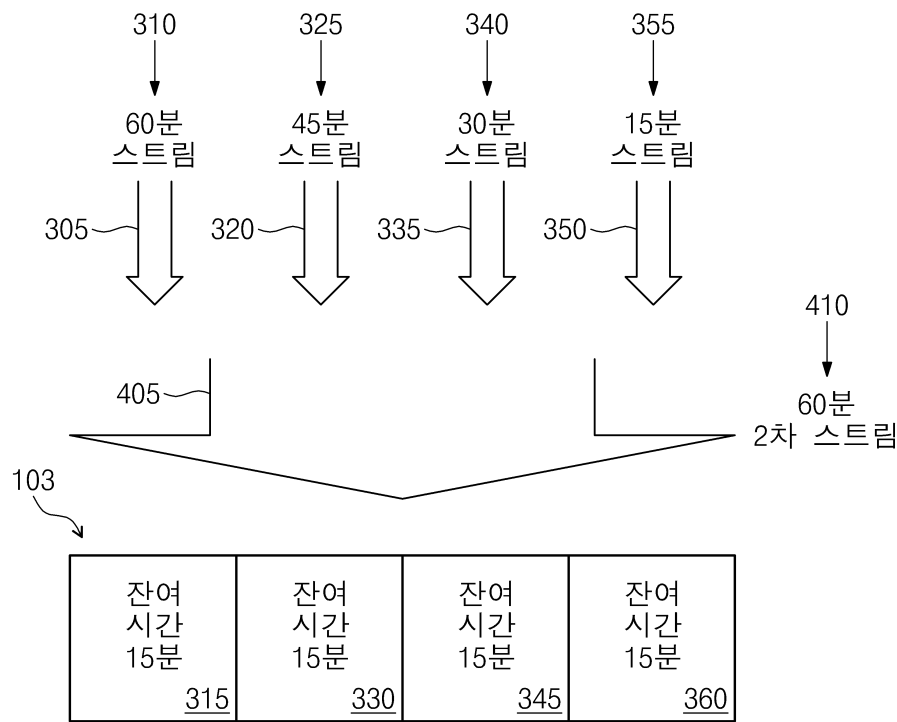
도면3c



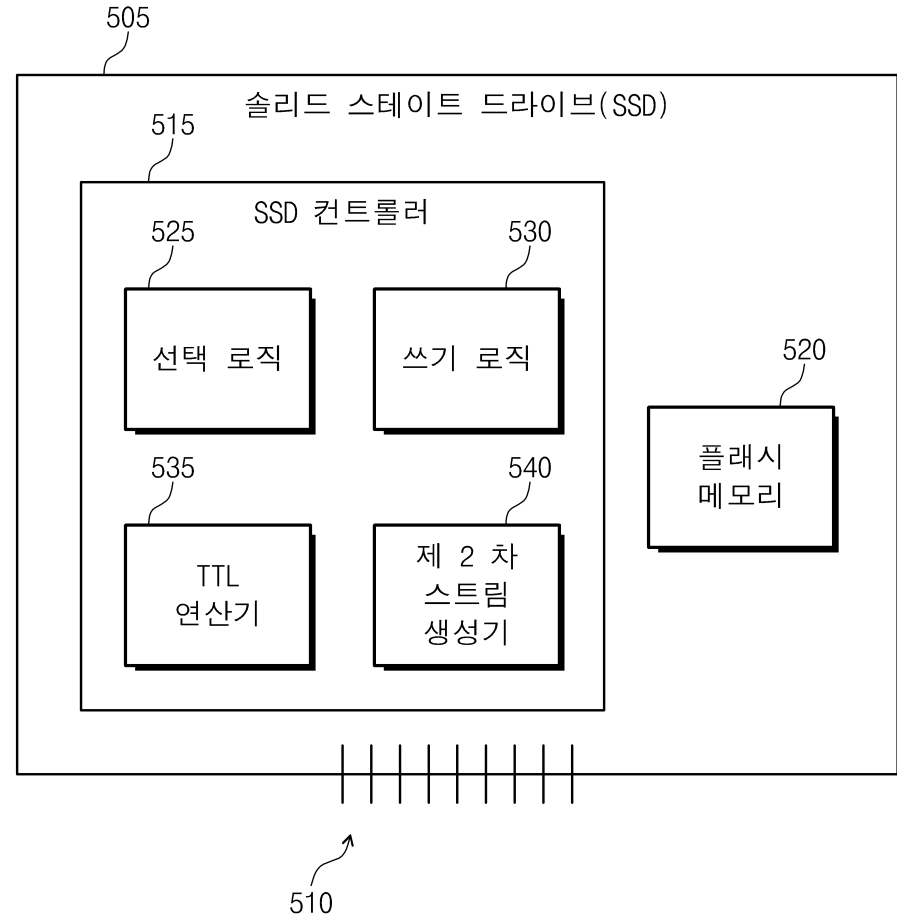
도면3d



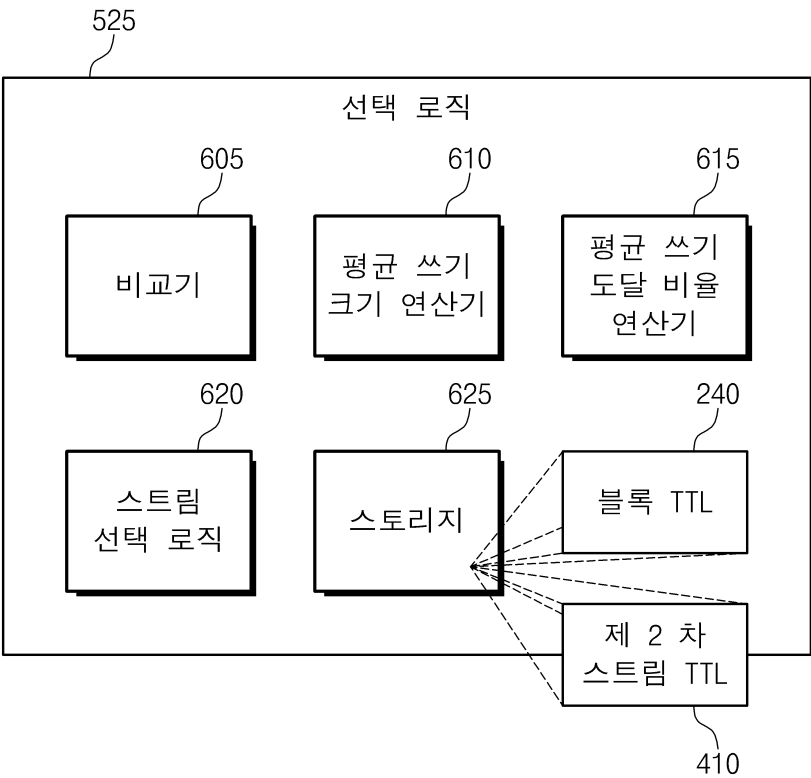
도면4



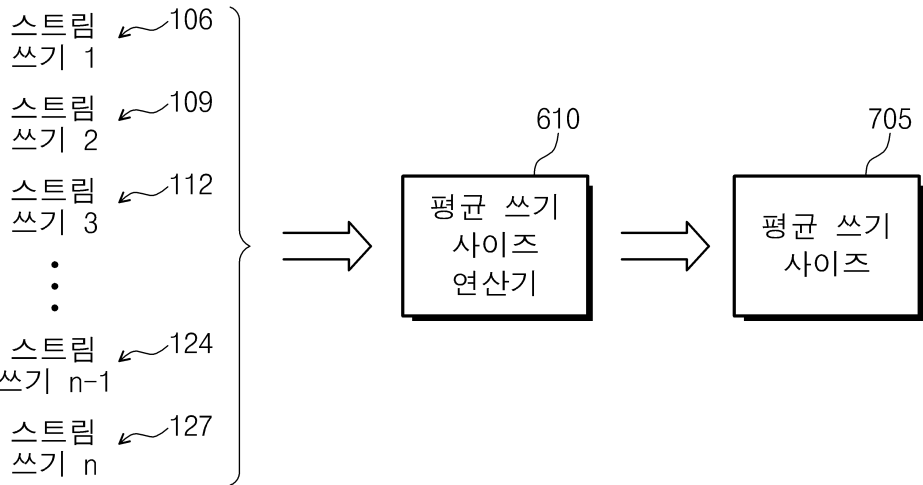
도면5



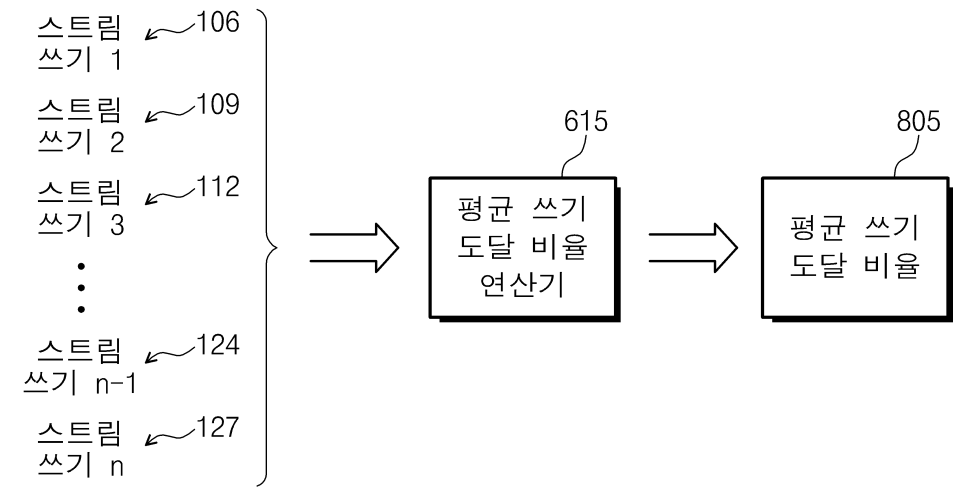
도면6



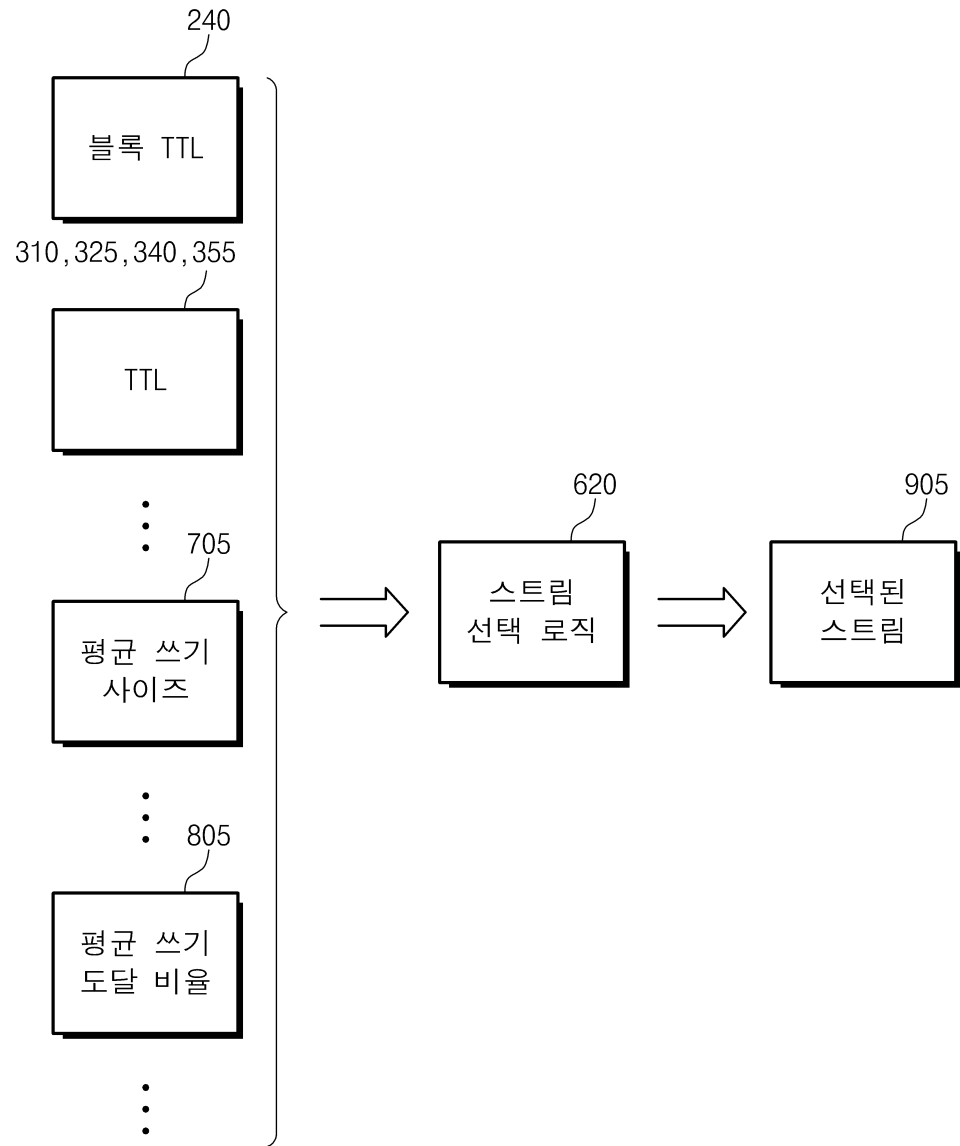
도면7



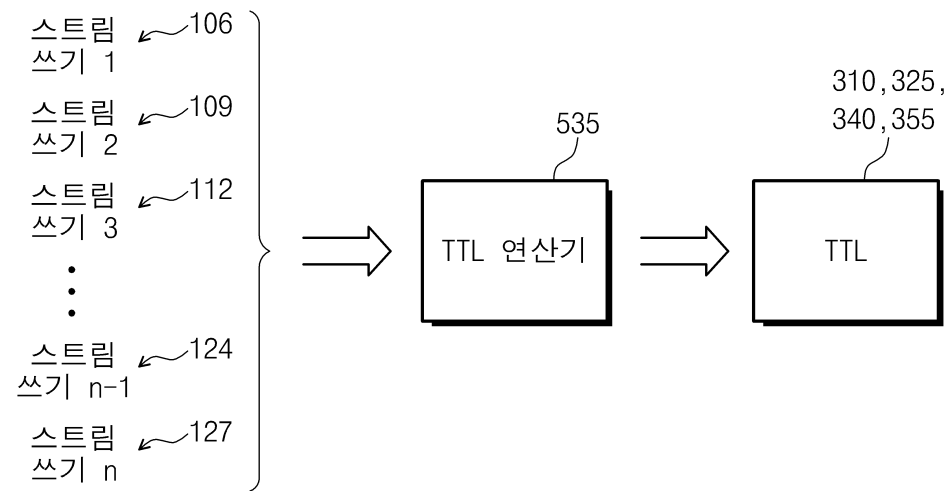
도면8



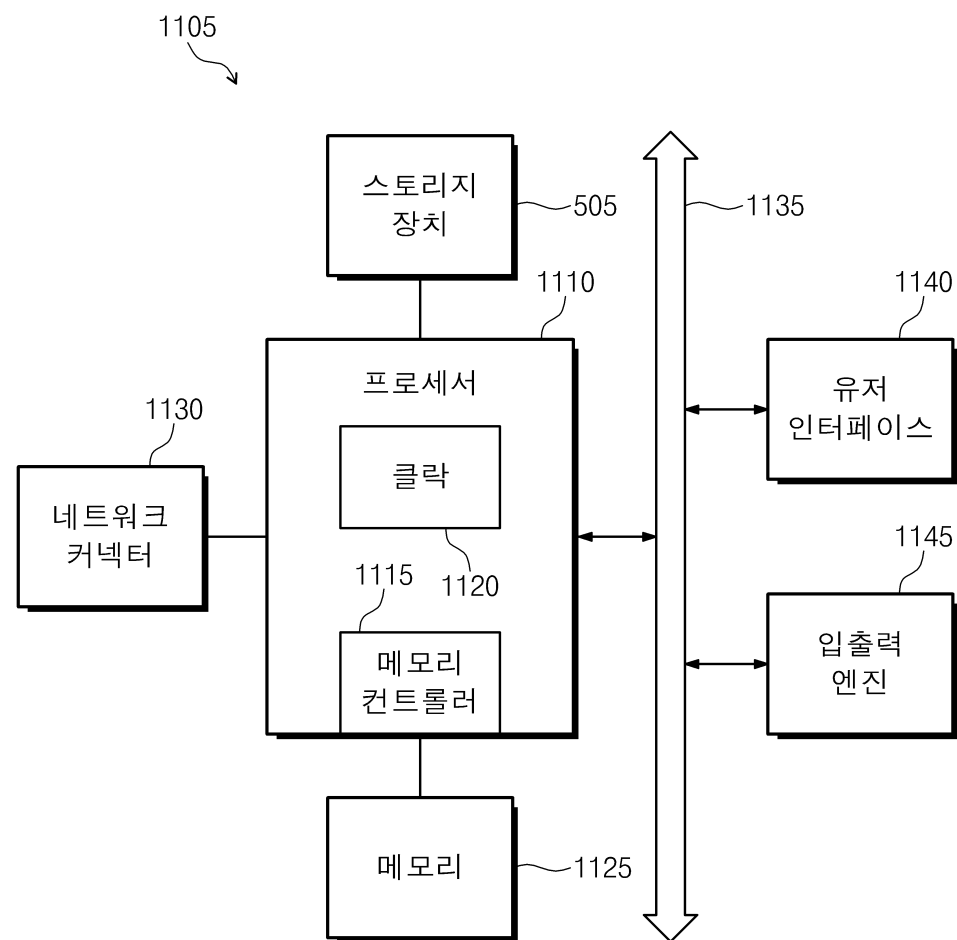
도면9



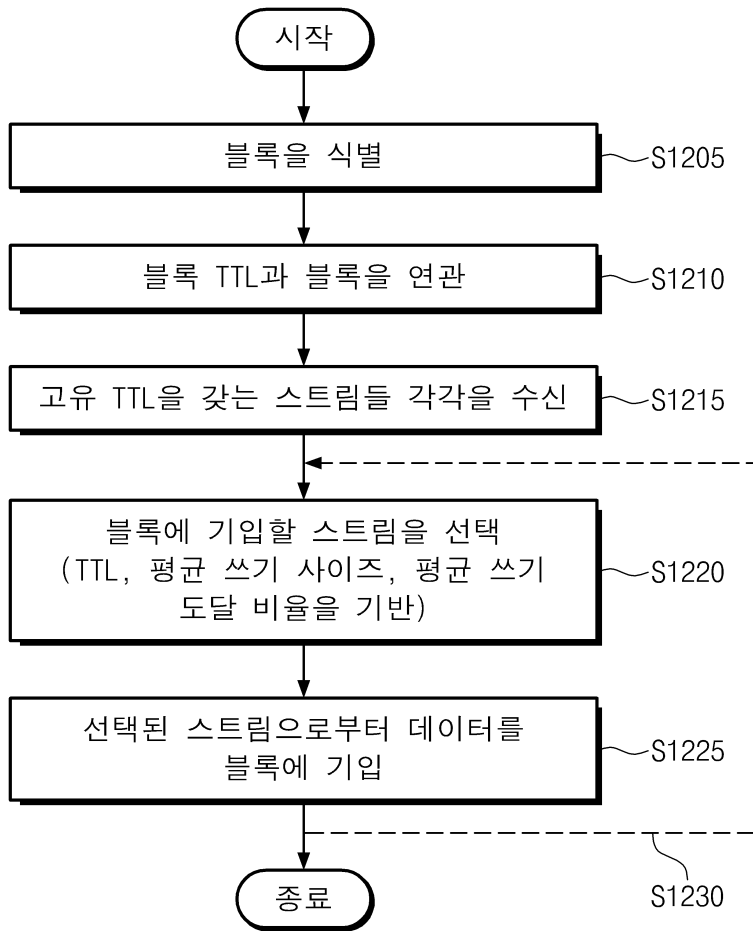
도면10



도면11



도면12



도면13

