(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2023년02월03일
(11) 등록번호 10-2494568
(24) 등록일자 2023년01월27일

(51) 국제특허분류(Int. Cl.)
G06F 3/06 (2006.01) *G06F 13/28* (2006.01)
(52) CPC특허분류
G06F 3/0659 (2013.01)
G06F 13/28 (2013.01)
(21) 출원번호 10-2017-0174318
(22) 출원일자 2017년12월18일
심사청구일자 2020년12월18일
(65) 공개번호 10-2018-0119464
(43) 공개일자 2018년11월02일
(30) 우선권주장
62/489,991 2017년04월25일 미국(US)
15/655,738 2017년07월20일 미국(US)
(56) 선행기술조사문헌
KR1020170007103 A
(뒷면에 계속)
전체 청구항 수 : 총 19 항

(73) 특허권자
삼성전자주식회사
경기도 수원시 영통구 삼성로 129 (매탄동)
(72) 발명자
카차르, 람다스 피.
미국 캘리포니아주 95014 쿠퍼티노 노르망디 웨이 7665
이동기
경기도 용인시
(뒷면에 계속)
(74) 대리인
특허법인 고려

심사관 : 김종기

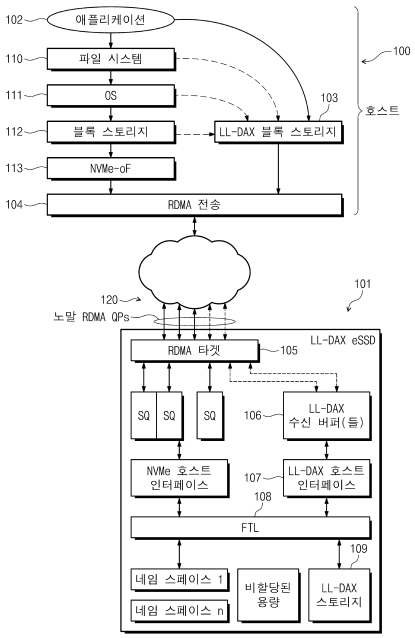
(54) 발명의 명칭 **NVMe-oF 이더넷 SSD에서의 낮은 레이턴시 직접 액세스 블록 스토리지**

(57) 요 약

이더넷을 통해 연결된 적어도 하나의 NVMe-oF SSD 장치 내 비-휘발성 플래시 메모리로의 낮은-레이턴시 직접 데이터 액세스 방법이 제공된다. 상기 방법은 호스트에서, 낮은-레이턴시 직접 액세스(LL-DAX) 블록 저장 소프트웨어 계층으로부터, 플래시 메모리로 원격 직접 메모리 액세스(RDMA) 쓰기 요청을 송신하는 단계를 포함한다. 상기

(뒷면에 계속)

대 표 도 - 도2



RDMA 쓰기 요청은 데이터, 스토리지 어드레스, 데이터 전송 동작의 길이 및 연산 코드를 포함한다. 또한 상기 방법은 상기 호스트에서, 상기 데이터가 상기 플래시 메모리에 지속되었음을 나타내는 RDMA 레벨 확인 응답을 수신하는 단계를 포함한다. 또한 상기 방법은 상기 LL-DAX 블록 스토리지 소프트웨어 계층으로부터, 스토리지 어드레스, 데이터 전송의 길이, 및 연산 코드를 포함하는 RDMA 읽기 요청을 상기 플래시 메모리에 전송하는 단계를 포함한다. 또한 상기 방법은 상기 호스트에서, 상기 플래시 메모리로부터 상기 RDMA 읽기 요청에 대응하는 RDMA 읽기 응답 데이터 패킷들을 수신하는 단계를 포함한다.

(52) CPC특허분류
G06F 3/0626 (2013.01)
G06F 3/0652 (2013.01)
G06F 3/0661 (2013.01)
G06F 3/067 (2013.01)
G06F 3/0679 (2013.01)

(72) 발명자
순다르 라지, 아제이
미국 캘리포니아주 95134 산호세 리오 로블레스 이스트 135 #169
울리, 프레드
미국 캘리포니아주 95129 산호세 그린 드라이브 1471

(56) 선행기술조사문헌
비특허문헌
US20020152327 A1
US20160085718 A1
US20140279941 A1

명 세 서

청구범위

청구항 1

이더넷을 통해 연결된 적어도 하나의 NVMe-oF(non-volatile memory express over fabric) SSD(solid-state drive)로의 비-휘발성 플래시 메모리에 대한 낮은-레이턴시 직접 데이터 액세스 방법에 있어서:

호스트에서, 낮은-레이턴시 직접 액세스(LL-DAX) 블록 스토리지 소프트웨어 계층으로부터, LL-DAX 프로토콜을 사용하여 상기 플래시 메모리로 데이터, 스토리지 어드레스, 데이터 전송 동작의 길이, 및 연산 코드를 포함하는 원격 직접 메모리 액세스(RDMA) 쓰기 요청을 전송하는 단계;

상기 LL-DAX 블록 스토리지 소프트웨어 계층의 상기 호스트에서, NVMe-oF 계층으로부터 NVMe-oF 프로토콜을 사용하여 상기 플래시 메모리로 RDMA 쓰기 요청을 전송하는 것과 동시에 상기 LL-DAX 블록 스토리지 소프트웨어 계층으로부터 상기 플래시 메모리로 상기 RDMA 쓰기 요청을 전송하는 단계;

상기 호스트에서, 상기 데이터가 상기 플래시 메모리에 지속되었음을 나타내는 RDMA 레벨 확인 응답을 수신하는 단계;

상기 LL-DAX 블록 스토리지 소프트웨어 계층으로부터, 상기 플래시 메모리로 스토리지 어드레스, 데이터 전송의 길이, 및 연산 코드를 포함하는 RDMA 읽기 요청을 전송하는 단계; 및

상기 호스트에서, 상기 플래시 메모리로부터 상기 RDMA 읽기 요청에 대응하는 RDMA 읽기 응답 데이터 패킷들을 수신하는 단계를 포함하되,

상기 LL-DAX 프로토콜을 사용한 상기 RDMA 쓰기 요청 및 상기 NVMe-oF 프로토콜을 사용한 상기 RDMA 쓰기 요청은 모두 상기 호스트로부터 발생하는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 2

제 1 항에 있어서,

상기 RDMA 쓰기 요청의 상기 스토리지 어드레스, 상기 데이터 전송의 상기 길이, 및 상기 연산 코드와 상기 RDMA 읽기 요청의 상기 스토리지 어드레스, 상기 데이터 전송의 상기 길이, 및 상기 연산 코드는 각 RDMA 헤더 필드들로 전달되는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 3

제 1 항에 있어서,

상기 RDMA 쓰기 요청을 전송하는 단계는 복수의 RDMA 큐 쌍들(QPs)을 사용하여 상기 RDMA 쓰기 요청을 전송하는 단계를 포함하고,

상기 RDMA 읽기 요청을 전송하는 단계는 복수의 RDMA 큐 쌍들(QPs)을 사용하여 상기 RDMA 읽기 요청을 전송하는 단계를 포함하는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 4

제 3 항에 있어서,

둘 이상의 RDMA 큐 쌍들(QPs)이 호스트 명령들을 가지며, 상기 방법이 LL-DAX 호스트 인터페이스(HIF) 로직에 의해 중재를 수행하는 단계를 더 포함하되, 상기 중재는 실행을 위해 상기 호스트 명령들 중 하나를 선택하는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 5

제 3 항에 있어서,

상기 복수의 RDMA 큐 쌍들(QPs) 중 제1 RDMA 큐 쌍(QP)은 제1 명령과 관련되고, 상기 복수의 RDMA 큐 쌍들

(QPs) 중 제2 RDMA 큐 쌍(QP)은 상기 제1 명령과 다른 제2 명령과 관련되는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 6

제 5 항에 있어서,

상기 제1 명령은 쓰기 명령이고, 상기 제2 명령은 읽기 명령인 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 7

제 1 항에 있어서,

상기 호스트에서, 상기 LL-DAX 블록 스토리지 소프트웨어 계층으로부터 식별 명령을 상기 플래시 메모리로 전송하는 단계를 더 포함하고,

상기 플래시 메모리는 상기 식별 명령에 응답하여, 상기 플래시 메모리에 관한 정보를 상기 호스트에 전송하는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 8

제 7 항에 있어서,

상기 정보는 상기 플래시 메모리의 지원되는 특성들, 성능들 및 특징들을 포함하는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 9

제 7 항에 있어서,

상기 식별 명령은 상기 RDMA 읽기 요청과 함께 전송되는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 10

제 1 항에 있어서,

상기 호스트의 상기 LL-DAX 블록 스토리지 소프트웨어 계층으로부터, 상기 플래시 메모리로 삭제 명령을 전송하는 단계를 더 포함하며,

상기 삭제 명령은 블록 어드레스들의 범위의 시작 블록 어드레스 및 종료 블록 어드레스 범위를 포함하는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 11

제 10 항에 있어서,

상기 삭제 명령은 상기 RDMA 쓰기 요청과 함께 전송되는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 12

제 1 항에 있어서,

상기 RDMA 읽기 및 상기 RDMA 쓰기 요청들은 RoCE v1, RoCE v2 및 iWARP로 구성된 RDMA 전송 프로토콜의 그룹으로부터 선택된 전송 프로토콜을 사용하여 전송되는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 13

제 3 항에 있어서,

상기 LL-DAX 블록 스토리지 소프트웨어 계층은 타임아웃 값을 포함하고, 상기 RDMA 큐 쌍들(QPs)은 상기 RDMA 쓰기 요청을 전송하는 단계 동안 타임아웃을 방지하도록 구성된 수신 버퍼를 포함하는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 14

제 1 항에 있어서,

상기 플래시 메모리의 LL-DAX 스토리지 용량은 NVMe-oF 네임스페이스들과 공유되는 낮은-레이턴시 직접 데이터 액세스 방법.

청구항 15

비-휘발성 플래시 메모리에 대한 낮은-레이턴시 직접 데이터 액세스를 위한 시스템 있어서:

상기 비-휘발성 플래시 메모리를 포함하고, 이더넷을 통해 연결되는 적어도 하나의 NVMe(non-volatile memory express) SSD(solid-state drive) 장치; 및

프로세서, NVMe-oF(Non-Volatile Memory Express Over Fabric) 계층, 및 내부에 저장된 명령어들을 포함하는 낮은-레이턴시 직접 액세스(LL-DAX) 블록 스토리지 소프트웨어 계층을 포함하는 호스트 장치를 포함하되,

상기 명령어들이 상기 프로세서에 의해서 실행될 때, 상기 명령어들은 상기 프로세서가,

LL-DAX 프로토콜을 사용하여 상기 비-휘발성 플래시 메모리로 데이터, 스토리지 어드레스, 데이터 전송 동작의 길이, 및 연산 코드를 포함하는 원격 직접 메모리 액세스(RDMA) 쓰기 요청을 전송하고,

상기 데이터가 상기 비-휘발성 플래시 메모리에 지속되었음을 나타내는 확인 응답을 수신하고,

스토리지 어드레스, 데이터 전송의 길이, 및 연산 코드를 포함하는 RDMA 읽기 요청을 상기 비-휘발성 플래시 메모리로 전송하고, 그리고

상기 비-휘발성 플래시 메모리로부터 상기 RDMA 읽기 요청에 대응하는 데이터 패킷들을 수신하도록 하고,

상기 NVMe-oF 계층은 상기 NVMe-oF 계층에 저장된 명령어들을 가지며, 상기 명령어들이 상기 프로세서에 의해 실행될 때, 상기 프로세서가 상기 LL-DAX 블록 스토리지 소프트웨어 계층으로부터 상기 비-휘발성 플래시 메모리로 NVMe-oF 프로토콜을 사용하여 상기 RDMA 쓰기 요청과 동시에 상기 RDMA 쓰기 요청을 상기 비-휘발성 플래시 메모리로 전송하도록 하고,

상기 LL-DAX 프로토콜을 사용한 상기 RDMA 쓰기 요청 및 상기 NVMe-oF 프로토콜을 사용한 상기 RDMA 쓰기 요청은 모두 상기 호스트 장치로부터 발생하는 시스템.

청구항 16

제 15 항에 있어서,

상기 NVMe SSD 장치들은 2개 이상의 RDMA 큐 쌍들(QPs) 내 호스트 명령들 사이의 중재하도록 구성된 LL-DAX 호스트 인터페이스(HIF) 로직을 포함하는 시스템.

청구항 17

제 16 항에 있어서,

상기 2개 이상의 RDMA 큐 쌍들(QPs) 중 제1 RDMA 큐 쌍(QP)은 제1 명령 전용이고, 상기 2개 이상의 RDMA 큐 쌍들(QPs) 중 제2 RDMA 큐 쌍(QP)은 상기 제1 명령과 다른 제2 명령 전용인 시스템.

청구항 18

제 15 항에 있어서,

상기 LL-DAX 블록 스토리지 소프트웨어 계층은 타임아웃 값을 포함하고, 상기 적어도 하나의 NVMe SSD 장치는 상기 RDMA 쓰기 요청을 전송하는 동안 타임아웃을 방지하도록 구성된 수신 버퍼를 포함하는 시스템.

청구항 19

제 15 항에 있어서,

상기 LL-DAX 블록 스토리지 소프트웨어 계층에 저장된 상기 명령어들이 상기 프로세서에 의해서 실행될 때, 상기 명령어들은 상기 프로세서가 상기 비-휘발성 플래시 메모리로 식별 명령을 전송하거나 또는 상기 비-휘발성 플래시 메모리로 삭제 명령을 전송하도록 하는 시스템.

청구항 20

삭제

청구항 21

삭제

발명의 설명

기 술 분 야

[0001] 본 발명은 솔리드-스테이트 드라이브들을 위한 일반적인 데이터 액세스 방법들 및 시스템들에 관한 것이다.

배 경 기 술

[0002] 솔리드 스테이트 드라이브(SSD)들은 급속히 현대의 IT 인프라스트럭쳐들(infrastructures)의 주요 저장 요소들이 되고 있으며, 종래의 하드 디스크 드라이브(HDD)들을 대체하고 있다. SSD들은 매우 낮은 레이턴시(latency), 높은 데이터 읽기/쓰기 처리량 및 신뢰적인 사용자 데이터 스토리지를 제공한다. NVMe-oF(Non-volatile memory express over fabric)는 수백 또는 수천 개의 NVMe(non-volatile memory express) SSD들이 이더넷(Ethernet)을 통해 연결될 수 있게 해주는 새로운 기술이다. NVMe-oF 프로토콜은 rDAS(remote Direct Attach Storage) 구현을 가능하게 한다. 이는 많은 수의 SSD들이 원격 호스트에 연결될 수 있게 한다. NVMe-oF 프로토콜은 RDMA(remote direct memory access) 프로토콜을 사용하여 NVMe 명령들, 데이터 및 응답들을 전달하는 신뢰성 있는 전송 서비스를 제공한다. iWARP(Internet wide area RDMA protocol), RoCE(RDMA over converged Ethernet) v1 및 RoCE v2는 RDMA 서비스들을 제공하는 전송 프로토콜들 중 일부이다.

[0003] 도 1은 종래의 NVMe-oF 이더넷 SSD 스토리지에 사용되는 시스템 아키텍처의 예를 도시한다. 도 1에 도시된 바와 같이, 서버들에서 실행되는 애플리케이션들은 일반적으로 시스템 소프트웨어 스택을 통해 스토리지를 액세스한다. 일반적인 스토리지 시스템 소프트웨어 스택은 애플리케이션(들), 오퍼레이팅 시스템(OS)/파일 시스템(FS)의 NVMe-oF 드라이버들, 블록 스토리지 계층(도 1에 도시되지 않음) 및 RDMA 전송 계층(도 1에 도시되지 않음) 등과 같은 많은 계층들을 포함한다. 시스템 소프트웨어의 이러한 수많은 계층들은 애플리케이션들에 의한 데이터 액세스에 많은 양의 레이턴시를 잠재적으로 추가할 수 있다. 저장 매체 기술(예를 들면, Z-NAND) 및 네트워킹 기술(예를 들면, 100Gbps 이더넷)의 발전으로 인해 미디어 및 네트워크로 인한 레이턴시는 크게 감소하고 있다. 따라서 시스템 소프트웨어 레이턴시는 전반적인 데이터 액세스 레이턴시의 중요한 구성 요소가 되고 있다.

[0004] 이 배경 기술 분야에서 개시된 상기 정보는 본 발명의 배경 지식을 향상시키기 위해 제공되며, 따라서 종래 기술에 구성되지 않은 정보를 포함할 수 있다.

발명의 내용

해결하려는 과제

[0005] 본 발명의 목적은 이더넷을 통해 연결된 NVMe(non-volatile memory express) SSD (Solid-State Drive) 장치에서 적어도 하나의 비-휘발성 플래시 메모리에 낮은-레이턴시 직접 데이터 액세스의 다양한 방법들을 제공하는데 있다.

과제의 해결 수단

[0006] 본 발명은 이더넷을 통해 연결된 NVMe(non-volatile memory express) SSD (Solid-State Drive) 장치에서 적어도 하나의 비-휘발성 플래시 메모리에 낮은-레이턴시 직접 데이터 액세스의 다양한 방법을 개시한다. 일 실시예에서, 상기 방법은 호스트에서, 낮은-레이턴시 직접 액세스(LL-DAX) 블록 저장 소프트웨어 계층으로부터, 플래시 메모리로 원격 직접 메모리 액세스(RDMA) 쓰기 요청을 송신하는 단계를 포함한다. 상기 RDMA 쓰기 요청은 데이터, 스토리지 어드레스, 데이터 전송 동작의 길이 및 연산 코드를 포함한다. 상기 방법은 상기 호스트에서, 상기 데이터가 상기 플래시 메모리에 지속되었음을 나타내는 RDMA 레벨 확인 응답을 수신하는 단계를 더 포함한다. 또한 상기 방법은 상기 LL-DAX 블록 스토리지 소프트웨어 계층으로부터, RDMA 읽기 요청을 상기 플래시 메모리에 전송하는 단계를 더 포함한다. 상기 RDMA 읽기 요청은 스토리지 어드레스, 데이터 전송의 길이, 및 연산 코드를 포함한다. 상기 방법은 상기 호스트에서, 상기 플래시 메모리로부터 상기 RDMA 읽기 요청에 대응하는

RDMA 읽기 응답 데이터 패킷들을 수신하는 단계를 더 포함한다.

[0007] 상기 RDMA 쓰기 요청 및 상기 RDMA 읽기 요청의 상기 스토리지 어드레스, 상기 데이터 전송의 길이 및 상기 연산 코드는 RDMA 헤더 필드들로 전달될 수 있다.

[0008] 상기 방법은 NVMe-oF(non-volatile memory express over fabric) 프로토콜을 사용하여 상기 호스트로부터 상기 플래시 메모리로 RDMA 쓰기 요청 및 RDMA 읽기 요청을 전송하는 단계를 더 포함할 수 있다.

[0009] 상기 RDMA 쓰기 요청을 전송하는 단계 및 상기 RDMA 읽기 요청을 전송하는 단계는 복수의 RDMA 큐 쌍들(QPs)로 전송하는 단계를 더 포함할 수 있다. 또한 상기 방법은 둘 이상의 RDMA QPs가 호스트 명령들을 가질 때 LL-DAX 호스트 인터페이스 로직에 의해 중재를 수행하는 단계를 더 포함할 수 있다. 상기 중재는 실행을 위해 상기 호스트 명령들 중 하나를 선택한다. 상기 일련의 RDMA QPs 중 제1 RDMA QPs는 제1 명령 전용이고, 상기 일련의 RDMA QPs 중 제2 RDMA QPs는 상기 제1 명령과 다른 제2 명령 전용일 수 있다. 상기 제1 명령은 쓰기 명령이고, 상기 제2 명령은 읽기 명령일 수 있다.

[0010] 또한 상기 방법은 상기 호스트에서, 상기 LL-DAX 블록 스토리지 소프트웨어 계층으로부터 식별 명령을 상기 플래시 메모리로 전송하는 단계를 포함하고, 상기 플래시 메모리는 상기 식별 명령에 응답하여, 상기 플래시 메모리에 관한 정보를 호스트에 전송한다. 상기 정보는 상기 플래시 메모리의 지원되는 특성들, 성능들 및 특징들을 포함할 수 있다. 상기 식별 명령은 상기 RDMA 읽기 요청과 함께 전송될 수 있다.

[0011] 상기 방법은 상기 호스트의 상기 LL-DAX 블록 스토리지 소프트웨어 계층으로부터, 플래시 메모리로 삭제 명령을 전송하는 단계를 더 포함할 수 있으며, 상기 삭제 명령은 블록 어드레스들의 범위의 시작 블록 주소 및 종료 블록 주소 범위를 포함한다. 상기 삭제 명령은 상기 RDMA 쓰기 요청과 함께 전송될 수 있다.

[0012] 상기 RDMA 읽기 및 상기 RDMA 쓰기 요청들은 RoCE v1, RoCE v2 및 Iwarp와 같은 RDMA 전송 프로토콜로 전송될 수 있다.

[0013] 상기 LL-DAX 블록 스토리지 소프트웨어 계층은 타임아웃 값을 포함할 수 있고, 상기 LL-DAX QPs는 상기 RDMA 쓰기 요청을 전송하는 단계동안 타임아웃을 방지하도록 구성된 수신 버퍼를 포함할 수 있다.

[0014] 상기 플래시 메모리의 LL-DAX 스토리지 용량은 NVMe-oF 네임스페이스들과 공유될 수 있다.

[0015] 본 발명은 비-휘발성 플래시 메모리에 대한 낮은-레이턴시 직접 데이터 액세스를 위한 시스템의 다양한 실시예들을 개시한다. 일 실시예에서, 상기 시스템은 플래시 메모리를 포함하고, 이더넷을 통해 연결되는 적어도 하나의 NVMe-oF(non-volatile memory express over fabric) SSD(solid-state drive) 장치 및 프로세서 및 낮은-레이턴시 직접 액세스(LL-DAX) 블록 저장 소프트웨어 계층을 포함하는 호스트 장치를 포함한다. 상기 LL-DAX 블록 스토리지 소프트웨어 계층은 내부에 저장된 명령들을 포함하며, 상기 명령들이 상기 프로세서에 의해서 실행될 때, 상기 명령들은 상기 프로세서가, 상기 플래시 메모리로 데이터, 스토리지 어드레스, 데이터 전송 동작의 길이 및 연산 코드를 포함하는 원격 직접 메모리 액세스(RDMA) 쓰기 요청을 송신하고, 상기 데이터가 상기 플래시 메모리에 지속되었음을 나타내는 RDMA 레벨 확인 응답을 수신하고, 스토리지 어드레스, 데이터 전송의 길이, 및 연산 코드를 포함하는 RDMA 읽기 요청을 상기 플래시 메모리에 전송하고, 그리고 상기 플래시 메모리로부터 상기 RDMA 읽기 요청에 대응하는 RDMA 읽기 응답 데이터 패킷들을 수신하도록 한다.

[0016] 상기 호스트 장치는, 상기 프로세서에 의해서 실행될 때, 상기 프로세서가 NVMe-Of(non-volatile memory express over fabric) 프로토콜을 사용하여 RDMA 쓰기 요청 또는 RDMA 읽기 요청을 상기 비-휘발성 플래시 메모리로 전송하도록 하는 명령들을 더 포함한다. 상기 NVMe-Of 프로토콜 및 상기 LL-DAX 모드는 배타적으로 뜨는 동시에 함께 동작할 수 있다.

[0017] 상기 NVMe SSD 장치들은 2개 이상의 RDMA 큐 쌍들(QPs) 내 호스트 명령들 사이의 중재를 위한 LL-DAX 호스트 인터페이스 로직을 포함할 수 있다. 상기 2개 이상의 RDMA QPs 중 제1 RDMA QPs는 제1 명령 전용이고, 상기 복수의 RDMA QPs 중 제2 RDMA QPs는 상기 제1 명령과 다른 제2 명령 전용일 수 있다.

[0018] 상기 LL-DAX 블록 스토리지 소프트웨어 계층은 타임아웃 값을 포함하고, 상기 적어도 하나의 NVMe SSD 장치는 상기 RDMA 쓰기 요청을 전송하는 동안 타임아웃을 방지하도록 구성된 수신 버퍼를 포함할 수 있다.

[0019] 상기 LL-DAX 블록 스토리지 소프트웨어 계층에 저장된 상기 명령들은, 상기 프로세서에 의해서 실행될 때, 상기 프로세서가 상기 비-휘발성 플래시 메모리로 식별 명령을 전송하거나 또는 상기 비-휘발성 플래시 메모리로 삭제 명령을 전송하도록 하도록 할 수 있다.

[0020] 상기 구성의 요약은 본 발명의 실시예들의 특징들 및 개념들의 선택을 소개하기 위해 제공되는 것으로, 본 발명의 실시예들은 이하 상세한 설명에서 더 설명된다. 상기 구성의 요약은 청구된 주제의 핵심 또는 필수적인 특징을 식별하기 위한 것이 아니며 청구된 주제의 범위를 제한하려는 것이 아니다. 설명된 특징들 중 하나 이상은 동작 가능한 장치를 제공하기 위해 하나 이상의 다른 설명된 특징들과 결합될 수 있다.

발명의 효과

[0021] 본 발명의 시스템 및 방법은 종래의 데이터 액세스 방법에 이용되는 스토리지 소프트웨어 스택 계층들 중 일부를 바이패스하여 데이터 스토리지에 대한 낮은 낮은 레이턴시를 제공하고 현존하는 스토리지 소프트웨어 스택들의 복잡성 및 비용을 감소시키도록 구성된다. 또한, 본 개시의 다양한 시스템들 및 방법들은 임의의 추가적인 중재 스토리지 하드웨어 인프라스트럭처를 필요로 하지 않는다.

도면의 간단한 설명

[0022] 본 개시의 실시예들의 특징들 및 이점들은 이하 도면들과 관련하여 고려될 때 다음의 상세한 설명을 참조함으로써 더욱 명백해질 것이다. 도면들에서, 동일한 참조 번호들은 동일한 특징들 및 구성 요소들을 참조하기 위해 도면 전체에 걸쳐 사용된다. 도면들은 반드시 비율에 맞게 그려지는 것은 아니다.

도 1은 NVMe-oF(non-volatile memory express over fabric) 이더넷 SSD 스토리지에 이용되는 종래의 시스템 아키텍처를 도시하는 개략도이다.

도 2는 본 발명의 일 실시예에 따른 NVMe-oF 이더넷 SSD 스토리지에 이용되는 시스템 아키텍처를 도시하는 개략도이다.

도 3은 LL-DAX 쓰기 명령의 포맷을 도시하는 도면이다.

도 4는 LL-DAX 읽기 및 LL-DAX 식별 명령의 포맷을 나타내는 도면이다.

도 5는 LL-DAX 삭제 명령의 포맷을 나타내는 도면이다.

도 6은 본 발명의 일 실시예에 따른 eSSD에서의 LL-DAX 동작의 흐름도이다.

발명을 실시하기 위한 구체적인 내용

[0023] 본 발명은 NVMe-oF(non-volatile memory express over fabric) 이더넷 SSD 스토리지에서 낮은-레이턴시 데이터 액세스를 위한 다양한 시스템들 및 방법들에 관한 것이다. 본 개시의 다양한 시스템들 및 방법들은 LL-DAX 프로토콜, LL-DAX 인에이블된 eSSD 및 LL-DAX 블록 스토리지 소프트웨어를 이용한다. 데이터 스토리지에 대한 낮은 레이턴시 액세스를 위한 다양한 시스템들 및 방법들은 고성능 및 비용 효율적인 솔루션을 달성하도록 구성된다. 본 개시의 다양한 시스템 및 방법은 종래의 데이터 액세스 방법에 이용되는 스토리지 소프트웨어 스택 계층들 중 일부를 바이패스하여 더 낮은 레이턴시를 제공하고 현존하는 스토리지 소프트웨어 스택들의 복잡성 및 비용을 감소시키도록 구성된다. 또한, 본 개시의 다양한 시스템들 및 방법들은 임의의 추가적인 중재(intervening) 스토리지 하드웨어 인프라스트럭처를 필요로 하지 않는다(즉, 본 개시의 다양한 시스템들 및 방법들은 스토리지 인프라스트럭처에 대한 임의의 새로운 하드웨어 요건들을 요구하지 않으며, 따라서 이들 시스템들 및 방법들은 설치된 하드웨어 인프라스트럭처에 영향을 주지 않음).

[0024] 이하, 첨부된 도면들을 참조하여 본 발명의 바람직한 실시예들에 대하여 상세히 설명한다. 그러나, 본 발명은 다양한 형태들로 구체화될 수 있으며, 여기에 도시된 실시예들에만 한정되는 것으로 해석되어서는 안된다. 오히려, 이들 실시예들은 본 개시가 철저하고 완전하게 이루어 지도록 당업자에게 본 발명의 양상 및 특징을 충분히 전달할 수 있도록 예로서 제공된다. 따라서, 본 발명의 양상들 및 특징들의 완전한 이해를 위해 당업자에게 불필요한 프로세스들, 요소들, 및 기술들은 설명되지 않을 수 있다. 다른 언급이 없는 한, 첨부된 도면들 및 상세한 설명 전반에 걸쳐 동일한 참조 부호들은 동일한 요소들을 나타내며, 따라서 그 설명은 반복되지 않을 수 있다.

[0025] 도면들에서, 엘리먼트들, 계층들 및 영역들의 상대적 크기들은 명확성을 위해 과장되거나 그리고/또는 간략화될 수 있다. "아래에", "하부에", "낮은", "상부", "위에" 등과 같은 공간적으로 관련된 용어들은 도면들에 도시된 바와 같이, 하나의 엘리먼트 또는 특징과 다른 엘리먼트(들) 또는 특징(들) 간의 관계를 용이하게 설명하기 위해 사용될 수 있다. 공간적으로 관련된 용어들은 도면들에 도시된 방위에 추가하여, 사용 또는 동작 중에 장치의 상이한 방위들을 포함하도록 의도된 것으로 이해될 것이다. 예를 들어, 도면들의 장치가 뒤집힌다면, 다

른 엘리먼트들 또는 특징들의 "아래" 또는 "아래에" 또는 "하부의"로 기술 된 요소는 다른 엘리먼트들 또는 특징들의 "위로" 향할 것이다. 따라서, "아래" 및 "아래에"의 예시적인 용어는 위와 아래의 방향 모두를 포함할 수 있다. 장치는 다른 방향으로 향할 수 있고(예를 들어, 90도 또는 다른 방향으로 회전될 수 있음), 공간적으로 상대적인 기술 용어들은 그에 따라 해석되어야 한다.

[0026] "제1", "제2", "제3"등의 용어는 본 명세서에서 다양한 구성 요소들, 엘리먼트들, 영역들, 계층들 및/또는 섹션들을 설명하기 위해 사용될 수 있지만, 이들 구성 요소들, 엘리먼트들, 영역들, 계층들 및/또는 섹션들은 이들 용어들에 의해 제한되어서는 안 된다. 이들 용어들은 하나의 요소, 엘리먼트, 영역, 층 또는 섹션을 다른 요소, 엘리먼트, 영역, 층 또는 섹션과 구별하기 위해 사용된다. 따라서, 이하에서 설명되는 제1 요소, 엘리먼트, 영역, 층 또는 섹션은 본 발명의 사상 및 범위를 벗어남이 없이 제2 요소, 엘리먼트, 영역, 계층 또는 섹션으로 지칭될 수 있다.

[0027] 엘리먼트 또는 계층이 다른 엘리먼트 또는 계층에 "연결된" 또는 "결합된" 것으로 언급될 때, 이는 다른 엘리먼트 또는 계층에 직접적으로 연결되거나, 결합될 수 있거나, 또는 하나 이상의 개재된 엘리먼트들 또는 계층들이 존재할 수 있다. 또한, 하나의 엘리먼트 또는 계층이 2 개의 엘리먼트들 또는 계층들의 "사이에" 있는 것으로 언급될 때, 2 개의 엘리먼트들 또는 층들 사이의 유일한 엘리먼트 또는 계층, 또는 하나 이상의 중간에 있는 엘리먼트들 또는 계층들이 존재할 수도 있다.

[0028] 본 명세서에서 사용되는 용어는 특정 실시 양태를 설명하기 위한 것일 뿐 본 발명을 제한하려는 것은 아니다. 본원에서 사용된 단수 형태는 문맥 상 다르게 지시하지 않는 한 복수 형태를 포함하는 것으로 의도된다. 본 명세서에서 사용된 "포함하는", "포함된", "구비하는" 및 "구비된"의 용어는 명시된 특징들, 정수들, 단계들, 동작들, 엘리먼트들 및/또는 구성 요소들의 존재를 특정하며, 하나 이상의 다른 특징들, 정수들, 단계들, 동작들, 엘리먼트들, 구성 요소 및/또는 이들의 그룹의 존재 또는 추가를 배제하지 않음이 잘 이해할 것이다. 본 명세서에 사용된 바와 같이, "및/또는" 이라는 용어는 하나 이상의 관련 열거 된 항목의 임의 및 모든 조합을 포함한다. "적어도 하나"와 같은 표현들은 엘리먼트들 앞에서 엘리먼트들의 전체 목록을 수식하고 목록의 개별 엘리먼트들을 수식하지 않는다.

[0029] 본 명세서에 사용된 바와 같이, 용어 "실질적으로", "약" 및 유사한 용어들은 근사의 용어로서 사용되며 정도(degree)의 용어로서 사용되지 않으며, 측정된 또는 계산된 값의 고유한 변화를 설명하기 위한 것이며, 이는 당업자에 의해 인식될 것이다. 또한, 본 발명의 실시예들을 설명할 때 "할 수 있다"를 사용하는 것은 "본 발명의 하나 이상의 실시예들"을 의미한다. 본 명세서에서 사용된 바와 같이, 용어 "사용하다", "사용하는" 및 "사용되는"은 "활용하다", "활용하는" 및 "활용되는"과 동의어로 간주될 수 있다. 또한 "예시적인"의 용어는 사례 또는 예시를 나타낸다.

[0030] 달리 정의되지 않는 한, 본 명세서에서 사용되는 모든 용어(기술 용어 및 과학 용어 포함)는 본 발명이 속하는 기술 분야의 당업자에 의해 일반적으로 이해되는 것과 동일한 의미를 갖는다. 또한, 일반적으로 사용되는 사전에서 정의된 용어들과 같은 용어들은 관련 기술 및/또는 본 명세서와 관련하여 그 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 이상적이거나 지나치게 형식적인 의미로 해석되지 않아야 한다.

[0031] **LL-DAX eSSD 프로토콜**

[0032] 도 2를 참조하면, 본 발명의 일 실시예에 따른 LL-DAX 스토리지 및 데이터 액세스 시스템은 호스트 장치(100) 및 LL-DAX eSSD(101)(즉, 이더넷을 통해 연결된 일련의 NVMe SSD 장치들)을 포함한다. LL-DAX eSSD(101)는 LL-DAX 특성을 추가적으로 지원하는 표준 NVMe-oF eSSD이다. 호스트 장치(100)는 애플리케이션(102), LL-DAX 블록 스토리지 소프트웨어(103) 및 RDMA 전송 계층(104)를 포함한다. 도시된 실시예에서, LL-DAX eSSD(101)는 RDMA 타겟 계층(105), RDMA 타겟 계층(105)에 연결된 LL-DAX 수신 버퍼(106), LL-DAX 수신 버퍼(106)에 연결된 LL-DAX 호스트 인터페이스(I/F)(107), LL-DAX 호스트 인터페이스(I/F)에 연결된 LL-DAX 호스트 인터페이스(FTL)(108) 및 플래시 변환 계층(FTL)(108)에 연결된 LL-DAX 스토리지(109)를 포함한다. 아래에서 보다 상세히 설명되는 바와 같이, 호스트 장치(100) 내의 LL-DAX 블록 스토리지 소프트웨어(103)는 LL-DAX 스토리지(109) 내 저장된 데이터에 대한 낮은-레이턴시 직접 액세스를 획득하기 위해 LL-DAX 프로토콜을 이용하여 호스트 명령들(예를 들어, RDMA 읽기(READ) 및 RDMA 쓰기(WRITE) 명령들)을 LL-DAX eSSD(101) 내 RDMA 타겟(105)으로 전송한다(예를 들어, LL-DAX 블록 스토리지 소프트웨어(103)는 호스트(100)에서 애플리케이션(102) 또는 다른 시스템 소프트웨어 계층에 저장 서비스를 제공하고, LL-DAX eSSD들 내 LL-DAX 스토리지로/로부터 데이터를 전송하기 위해 RDMA 읽기 및 RDMA 쓰기 요청들을 이용한다). 이러한 방식으로, 본 발명의 시스템은 호스트 장치(100)의 파일 시스템 계층(110), 오퍼레이팅 시스템(OS) 계층(111), 블록 스토리지 계층(112) 및 NVMe-oF 계층(113)을 바

이패스하고, LL-DAX eSSD(101)의 LL-DAX 스토리지 장치(109)에 저장된 데이터에 대한 낮은-레이턴시 직접 액세스를 획득하도록 구성된다.

[0033] 또한, 하나 이상의 실시예들에서, LL-DAX eSSD(101)는 표준을 따르는 NVMe-oF 이더넷 SSD로서 동작 할 수 있다. LL-DAX 데이터 액세스 모드는 옵션 기능이다. 따라서, 애플리케이션들은 원하는 대로 표준 블록 스토리지 모드 또는 LL-DAX 모드를 사용할 수 있다. 또한, 하나 이상의 실시예들에서, LL-DAX eSSD(101)는 동시에 액세스 모드들(즉, LL-DAX 및 NVMe-oF)을 모두 지원할 수 있다.

[0034] LL-DAX 프로토콜은 데이터 전송들(즉, LL-DAX 쓰기 및 LL-DAX 읽기 명령들이 데이터 액세스를 위해 이용됨)을 수행하기 위해 RDMA 읽기 및 RDMA 쓰기 요청들을 포함하는 RDMA(remote direct memory access) 전송을 사용한다. RDMA 읽기 요청들은 eSSD(101)로부터 데이터를 검색하거나 페치(fetch)하는데 이용된다. RDMA 쓰기 요청들은 호스트(100)로부터 eSSD(101)로 데이터를 전송하는데 이용된다. 아래의 표 1에 도시된 바와 같이, RDMA 읽기 및 RDMA 쓰기 각각의 요청은 LL-DAX 스토리지에 관련된 연산부호(opcode)(OPC) 필드, 어드레스 필드(ADDR) 및 길이 필드(LEN)를 포함한다.

표 1

[0035]

필드 이름	M/O	폭(비트)	설명
연산 코드(OPC)	M	8	eSSD에 대한 명령
어드레스(ADDR)	M	64	저장 위치, 블록 어드레스
길이(LEN)	M	64	전송 길이, 블록 사이즈 단위

[0036] 아래의 표 2는 본 개시의 일 실시예에 따른 RDMA 쓰기 요청 및/또는 RDMA 읽기 요청의 연산 코드(OPC) 필드에 대한 필수 명령 세트(M) 및 선택적인 명령 세트(O)를 포함하는 명령 세트(즉, 인코딩 세트)를 보여준다. 도 3은 LL-DAX 쓰기 명령의 포맷을 도시하는 도면이고, 도 4는 LL-DAX 읽기 명령의 포맷을 도시하는 도면이다.

표 2

[0037]

연산 코드 값(8-bit)	M/O	인코딩
0	M	LL-DAX 호스트 쓰기
1	M	LL-DAX 호스트 읽기
2	M	LL-DAX 식별
3	O	LL-DAX 삭제
4-255	NA	RSVD

[0038] 상술한 LL-DAX 명령 필드들(즉, 연산 코드(OPC), 어드레스(ADDR) 및 길이(LEN))은 기존의 RDMA 헤더 필드들에 전달된다. 즉, 일반적으로 원격 버퍼 세부 사항들에 사용되는 기존 RDMA 헤더 필드들은 LL-DAX 명령 필드들(즉, 연산 코드(OPC), 어드레스(ADDR) 및 길이(LEN))을 전달하기 위해 용도를 변경해 사용된다. iWARP(Internet wide area RDMA protocol), RoCE(RDMA over converged Ethernet) v1 및 RoCE v2와 같은 여러 RDMA 전송 프로토콜들이 있으므로, LL-DAX 사용을 위해 용도 변경되는 실제 필드 이름들은 RDMA 전송 프로토콜들에 따라 변경될 수 있다.

[0039] 아래 표 3은 RoCE v1 및 RoCE v2 요청들에 대한 하나의 가능한 매핑을 나타낸다.

표 3

[0040]

LL-DAX 필드 이름	읽기/쓰기
연산 코드(OPC)	RETH.R-Key[7:0]
어드레스(ADDR)	RETH.VA[63:0]
길이(LEN)	RETH.DMALen[31:0]

표 4

[0041]

LL-DAX 필드 이름	읽기	쓰기
연산 코드(OPC)	RDMAP.SrcSTag[7:0]	DDP.Sink STag[7:0]
어드레스(ADDR)	RDMAP.SrcTO[63:0]	DDP.Sink Tagged Offset[63:0]

길이(LEN)	RDMAP.RDMARDSZ[31:0]	

[0042] 또한, LL-DAX 프로토콜은 eSSD(101)로부터 4KB 데이터 구조를 페치하기 위해 LL-DAX 식별(IDENTIFY) 명령을 사용한다. LL-DAX 식별(IDENTIFY) 명령을 수신한 것에 응답하여, eSSD(101)는 LL-DAX 스토리지(109)에 관한 정보 예를 들면, 지원되는 특징들(features), 성능들(capabilities) 및 특성들(characteristics)을 보고한다. LL-DAX 식별(IDENTIFY) 명령의 포맷은 LL-DAX 읽기 명령의 포맷과 동일하며, LL-DAX 도 4에 도시되어 있다. 아래의 표 5는 RDMA 읽기 및 쓰기 명령들에 의한 액세스를 위해 예약된 LL-DAX 스토리지 용량(예, 1TB), eSSD(101) 상에 존재하는 수신 버퍼 크기, eSSD(101)에 의해 지원되는 최대 RDMA 쓰기 크기, 및 블록 크기를 포함하는 LL-DAX 식별(IDENTIFY) 명령에 응답하여 보고된 LL-DAX 스토리지(109)에 관한 정보를 파악한다. 하나 이상의 실시예들에서, LL-DAX 스토리지(109)는 LL-DAX 애플리케이션들 및 사용자들에 대한 eSSD(101) 내부의 전용 저장 용량일 수 있다. 하나 이상의 실시예들에서, LL-DAX 및 비-LL-DAX(non-LL-DAX) 애플리케이션들은 파일 시스템 또는 다른 시스템 스택 계층들을 통해 eSSD(101) 내부에서 동일한 저장 용량을 공유할 수 있다(예를 들어, LL-DAX 및 비-LL-DAX 애플리케이션들은 서로 데이터를 교환할 수 있다). LL-DAX 애플리케이션들과 레거시(legacy) 비-LL-DAX) 애플리케이션들 간의 저장 공간 공유는 SSD 컨트롤러 레벨에서 수행될 수도 있다. 예를 들어, 하나 이상의 실시예들에서, LL-DAX 스토리지(109)는 NVMe-oF 네임스페이스(NameSpace)들과 공유될 수 있다. LL-DAX 스토리지(109)가 NVMe-oF 네임스페이스들과 공유되는) 하나 이상의 실시 예에서, 액세스 조정은 호스트 소프트웨어의 상위 레벨에서 수행된다.

표 5

[0043]

필드 이름	M/O	폭(비트)	설명
용량	M	128	LL-DAX 용량 (바이트)
수신 버퍼 크기	M	32	LL-DAX에 대한 eSSD 상에 존재하는 수신 버퍼 크기 (KB)
최대 쓰기 크기	M	32	RDMA 쓰기 캡슐을 지원하는 최대 크기 (KB)
블록 크기	M	32	블록 크기 (bytes)
RSVD	M	4K-28 Bytes	나중에 사용될 것으로 예약된 나머지 공간

[0044] LL-DAX 프로토콜은 선택적으로 LL-DAX 삭제(DELETE) 명령을 사용한다. LL-DAX 삭제 명령은 하나 이상의 삭제 어드레스 범위 엔트리들을 전달한다. 각각의 삭제 범위 엔트리는 삭제 범위 엔트리의 시작 블록 어드레스 및 종료 블록 어드레스를 포함하는 16 바이트(Byte) 데이터 구조이다. 도 5는 LL-DAX 삭제 명령의 포맷을 나타내는 도면이다. 다음 표 6은 삭제 범위 엔트리의 포맷을 보여준다.

표 6

[0045]

필드 이름	M/O	폭(비트)	설명
시작 어드레스	M	64	범위의 시작 블록 어드레스
종료 어드레스	M	64	범위의 종료 블록 어드레스

[0046] LL-DAX 프로토콜에 대한 명령 세트는 확장 가능하며, 예를 들어 제어 및 상태 레지스터(CSR) 액세스 및 상태 로그(Health log)와 같은 추가 명령들이 LL-DAX 프로토콜에 추가될 수 있다.

[0047] 본 발명의 일 실시예에 따라 이더넷을 통해 접속된 NVMe SSD 장치들에서 플래시 메모리로의 낮은-레이턴시 데이터 액세스 방법은 RDMA 쓰기 요청을 호스트(100)에서의 LL-DAX 블록 스토리지 소프트웨어 계층(103)으로부터 송신 태스크(task), 데이터, 저장 어드레스, 데이터 전송 동작의 길이 및 연산 코드를 포함하는 플래시 메모리로 전송하는 작업을 포함한다. RDMA 쓰기 요청은 식별된 스토리지 어드레스에서 플래시 메모리에 데이터를 유지한다. 또한 이 방법은 RDMA 읽기 요청을 LL-DAX 블록 스토리지 소프트웨어 계층으로부터, 스토리지 어드레스, 데이터 전송의 길이 및 연산 코드를 포함하는 플래시 메모리로 전송하는 태스크를 포함한다. RDMA 읽기 요청은 플래시 메모리의 특정 스토리지 어드레스에 있는 데이터를 검색한다. RDMA 쓰기 요청 및 RDMA 읽기 요청의 스토리지 어드레스, 데이터 전송의 길이 및 연산 코드는 RDMA 헤더 필드들에서 전달된다. 또한 이 방법은 호스트에서의 LL-DAX 블록 스토리지 소프트웨어 계층으로부터, LL-DAX 식별(IDENTIFY) 명령을 플래시 메모리에 전송하여 플래시 메모리에 관한 정보, 예를 들어, 지원되는 특징들, 성능들 및 특성들을 획득하는 태스크를 포함할 수 있다. LL-DAX 식별은 RDMA 읽기 요청과 함께 전송된다. 또한 이 방법은 호스트에서의 LL-DAX 블록 스토리지 소프

트웨어 계층으로부터, 삭제(DELETE) 명령을 플래시 메모리에 전송하여 하나 이상의 특정 어드레스 범위들에서 플래시 메모리 상에 유지된 데이터를 삭제하는 태스크를 포함할 수 있다. 삭제(DELETE) 명령은 RDMA 쓰기 요청과 함께 전송된다.

[0048] LL-DAX eSSD Operation

[0049] 다시 도 2를 참조하면, LL-DAX는 eSSD가 내장형 RDMA 네트워크 인터페이스를 지원하기 때문에 표준 NVMe-oF 이더넷 SSD의 선택적 기능이다. NVMe-oF 프로토콜은 RDMA 큐 쌍들(queue pairs, QPs)을 사용하여 명령들, 데이터 및 완료들(completions)을 전송한다. NVMe-oF 호스트 드라이버는 RDMA 송신(SEND) 요청을 사용하여 eSSD에 명령들을 전송한다. eSSD는 데이터 전송을 위해 RDMA 읽기(READ) 및 RDMA 쓰기(WRITE) 요청을 사용한다. 또한 eSSD는 RDMA 송신(SEND) 요청을 사용하여 호스트에 완료들(completions)(예를 들면, 데이터 지속성 확인 응답들(acknowledgements))을 게시(post)한다

[0050] LL-DAX는 eSSD(101)에 존재하는 동일한 RDMA 인터페이스를 이용한다. LL-DAX는 LL-DAX 액세스를 위해 하나 이상의 개별 및 전용 큐 쌍들(QPs)(120)을 이용한다. LL-DAX 호스트는 RDMA 읽기 및 RDMA 쓰기 요청을 사용하여 eSSD(101)에 데이터를 전송하고, 명령을 송신한다. eSSD(101)는 LL-DAX 호스트의 RDMA 요청들(예를 들면, RDMA 읽기 및 RDMA 쓰기 요청들)에만 응답하고, 자체적으로 RDMA 요청들을 시작하지 않는다. RDMA 확인 응답들은 데이터 전송 완료들로 동작하며 NVMe-oF 완료들과는 달리 별도의 완료들이 없다. 즉, NVMe-oF에는 명시적 명령 완료들이 있지만 LL-DAX에는 이러한 명시적 완료가 없다. 대신에, LL-DAX는 명령 완료와 같은 목적으로 RDMA 레벨 확인 응답들을 사용한다. 하나 이상의 실시예들에서, NVMe-oF 프로토콜 모드 및 LL-DAX 모드는 배타적으로 또는 동시에 동작될 수 있다(예컨대, eSSD는 NVMe-oF 프로토콜 모드 및 LL-DAX 프로토콜 모드를 모두 지원하여 NVMe-oF 동작 모드는 LL-DAX 동작 모드와 공존할 수 있다).

[0051] 도 6은 본 발명의 일 실시예에 따른 eSSD에서 LL-DAX 동작을 나타내는 플로우차트이다. 도 6에 도시된 바와 같이, 동작(201)에서, eSSD는 호스트 장치에서 LL-DAX 블록 스토리지 소프트웨어 계층(103)(도 2 참조)으로부터 RDMA 읽기 요청 또는 RDMA 쓰기 요청을 수신한다. 동작(202)에서, RDMA 읽기 요청 또는 RDMA 쓰기 요청이 LL-DAX QP 상에 존재하는 지의 여부가 판별된다. eSSD가 LL-DAX QP 상에 존재하지 않은 RDMA 읽기 또는 RDMA 쓰기 요청을 수신하면, RDMA 읽기 또는 RDMA 쓰기 명령은 동작(203)에 도시된 바와 같이, eSSD를 액세스하기 위한 표준 NVME-oF 프로토콜을 따른다. eSSD가 호스트 장치의 LL-DAX 블록 스토리지 소프트웨어 계층(103)으로부터 LL-DAX QP 상의 RDMA 읽기 또는 RDMA 쓰기 요청을 수신하면, RDMA 요청은 LL-DAX 호스트 인터페이스(HIF) 로직(107)(도 2 참조)으로 전송된다. 동작(204)에서, 둘 이상의 LL-DAX QPs(120)가 호스트 명령들을 가질 때, LL-DAX 호스트 인터페이스 로직(107)은 이들 LL-DAX QPs(120) 사이의 중재를 수행해서 실행을 위한 호스트 명령들 중 하나를 선택한다. LL-DAX 호스트 인터페이스 로직(107)에 의해 수행되는 중재는 예를 들어, 간단한 라운드-로빈(round-robin) 중재 또는 호스트 제어 설정들로 구현된 복잡한 중재 스킴일 수 있다. 하나 이상의 실시예들에서, 상이한 LL-DAX QPs(120)는 상이한 명령들 사이의 실행을 구별하기 위해 사용될 수 있다. 예를 들어, 하나의 LL-DAX QP(120)는 쓰기 명령들만을 전송하기 위해 사용될 수 있고(예를 들어, 하나의 LL-DAX QP(120)는 배타적으로 쓰기 명령들을 전송하는데 전념할 수 있고), 반면 다른 LL-DAX QP 120는 읽기 명령들만을 전송하기 위해 사용될 수 있다(예를 들어, 하나의 LL-DAX QP(120)는 배타적으로 읽기 명령들을 전송하는데 전념할 수 있다). LL-DAX 호스트 인터페이스 로직(107)은 선택된 명령을 분석하고, RDMA 명령으로부터 관련된 LL-DAX 필드들뿐만 아니라 연산 코드를 추출한다. LL-DAX 호스트 인터페이스 로직(107)은 수신된 LL-DAX 명령을 실행하고 호스트로 RDMA 요청 패킷에 확인 응답(acknowledge)한다. 하나 이상의 실시예들에서, 별도의 명령 완료들이 없다. RDMA 요청 패킷 확인 응답들은 명령들의 완료들로 사용된다.

[0052] 계속해서 도 6을 참조하면, 동작(205)에서, LL-DAX 호스트 인터페이스 로직(107)(도 2 참조)은 호스트 명령이 RDMA 읽기 요청인지 또는 RDMA 쓰기 요청인지를 판별한다. 호스트 명령이 RDMA 쓰기 명령인 것으로 판별되면, 동작(206)에서, LL-DAX 호스트 인터페이스 로직(107)은 호스트 명령의 연산 코드 값을 검사하는 것에 의해 RDMA 쓰기 명령이 LL-DAX 쓰기 명령 또는 LL-DAX 삭제 명령(LL-DAX 삭제 명령은 RDMA 쓰기 명령과 함께 전송됨)인 지를 판별한다. 동작(207)에서, (예를 들어, 명령의 연산 코드 값이 0일 때) LL-DAX 호스트 인터페이스 로직(107)이 호스트 명령이 LL-DAX 쓰기 명령인 것으로 판별하면, 동작(208)에서 LL-DAX 호스트 인터페이스 로직(107)은 먼저 수신된 데이터를 플래시 미디어(예를 들어, 도 2에 도시된 LL-DAX 스토리지(109)) 또는 캐시에 지속시키고, 동작(209)에서, LL-DAX 호스트 인터페이스 로직(107)은 RDMA 쓰기에 확인 응답한다(예를 들어, LL-DAX 쓰기 명령의 확인 응답(acknowledgement)은 LL-DAX 호스트 인터페이스 로직(107)에 의해 호스트로 전송된다). 만일 eSSD가 전력 손실 보호 버퍼를 구비하면, 수신된 쓰기 데이터는 캐시에 쓰여질 수 있고, RDMA 쓰기 패킷은

확인 응답될 수 있다.

[0053] 데이터 지속 기간으로 인해 호스트 측에서 RDMA 쓰기 타임아웃(timeout)을 방지하기 위해, 호스트 RDMA 스택은 충분히 큰 타임아웃 값으로 프로그래밍 될 수 있다. 또한, LL-DAX QP 수신 버퍼(106)(도 2 참조)는 데이터 지속 기간으로 인해 일정 양의 쓰기 버스트들(write bursts)을 허용하도록 적절하게 크기 조정될 필요가 있다. eSSD 플래시 트랜지션 계층(FTL)이 사용중(busy)이고, 수신 버퍼를 비울 수 없는 경우, 쓰기 버스트들의 흡수는 호스트 성능을 향상시킨다. 예를 들어, 하나 이상의 실시예들에서, 50MB의 수신 버퍼는 40Gbps 속도에서 10ms 버스트가 호스트에 대한 역압(backpressure)없이 흡수될 수 있게 한다. 사용되는 역압 메커니즘은 표준 RDMA NACK 패킷들이다.

[0054] 도 6을 참조하면, LL-DAX 삭제 명령은 RDMA 쓰기 요청을 사용하여 전송된다. 동작(210)에서, LL-DAX 호스트 인터페이스 로직(107)은 호스트 명령이 LL-DAX 삭제 명령(예를 들어, 명령어의 연산 코드 값이 16 일 때)인 것으로 판별될 때, 먼저 동작(211)에서, LL-DAX 호스트 인터페이스 로직(107)은 플래시 변환 계층(FTL) 맵 테이블들의 업데이트를 시작한 후, 동작(212)에서 LL-DAX 호스트 인터페이스 로직(107)은 RDMA 쓰기 패킷을 확인 응답한다(예를 들어, LL-DAX 삭제 명령을 전송하기 위해 사용되는 LL-DAX 쓰기 명령의 확인 응답은, 동작(212)에서 LL-DAX 호스트 인터페이스 로직(107)에 의해 호스트로 전송된다). LL-DAX 삭제 명령을 전송하기 위해 사용되는 RDMA 쓰기 요청의 페이로드(payload)는 도 6에 도시된 바와 같이, 애플리케이션들에 의해 삭제되거나 할당 해제되는 하나 이상의 블록 어드레스 범위들을 전달한다. 할당 해제된 어드레스 정보는 플래시 미디어 관리를 위해 eSSD에 의해서 사용된다.

[0055] 계속해서 도 6을 참조하면, 동작(205)에서 LL-DAX 호스트 인터페이스 로직(107)는 호스트 명령이 RDMA 읽기 명령인 것으로 판별되면, 동작(213)에서 LL-DAX 호스트 인터페이스 로직(107)은 호스트 명령의 연산 코드 값을 검사(inspecting)하는 것에 의해 RDMA 읽기 명령이 LL-DAX 읽기 명령 또는 (RDMA 읽기 명령과 함께 전송되는) LL-DAX 식별 명령인 지를 판별한다. 동작(214)에서, LL-DAX 호스트 인터페이스 로직(107)은 호스트 명령이 LL-DAX 읽기 명령인 것으로 판별되면(예를 들면, 명령의 연산 코드 값이 1일 때), LL-DAX 호스트 인터페이스 로직(107)은 먼저 동작(215)에서, 플래시 미디어(예를 들면, LL-DAX 스토리지(109) 또는 캐시로부터 요청된 데이터를 페치하고, 동작(216)에서 요청된 데이터를 RDMA 읽기 응답(READ RESPONSE) 패킷들과 함께 호스트로 전송한다. 하나 이상의 다른 실시예들에서, 네트워크 최대 전송 단위(maximum transmission unit, MTU)에 따라서 다중 응답 패킷들이 있을 수 있다. 마지막 RDMA 읽기 응답 패킷은 LL-DAX 읽기 명령을 확인 응답한다.

[0056] 도 6에 도시된 바와 같이, LL-DAX 식별 명령은 RDMA 읽기 요청을 사용하여 전송된다(즉, LL-DAX 식별 명령은 RDMA 읽기 요청의 일부로서 수신된다). LL-DAX 식별 명령은 LL-DAX 읽기 명령과 유사한 실행 흐름을 갖는다. 동작(217)에서 LL-DAX 호스트 인터페이스 로직(107)은 호스트 명령이 LL-DAX 식별 명령인 것으로 판별되면(예를 들어, 명령의 연산 코드 값이 2일 때), 동작(218)에서, LL-DAX 호스트 인터페이스 로직(107)은 먼저 플래시 미디어(예를 들어, LL-DAX 스토리지(109)) 또는 캐시로부터 식별 데이터 구조를 페치하고, 동작(219)에서 RDMA READ 응답 패킷들을 사용하여 호스트에 식별 데이터 구조를 전송한다.

[0057] **LL-DAX Block Storage Software Operation**

[0058] 다시 도 2를 참조하면, LL-DAX 블록 스토리지 소프트웨어 계층(103)은 애플리케이션들 또는 다른 시스템 소프트웨어 계층들에 대한 스토리지 액세스를 제공한다. LL-DAX 소프트웨어(103)는 발견, 애플리케이션 측 인터페이스, 데이터 전송들 및 에러 처리를 포함하는 DAX 스토리지와 관련된 다양한 기능들을 수행한다.

[0059] 초기에, LL-DAX 소프트웨어(103)는 LL-DAX eSSD(101)를 발견하고 LL-DAX eSSD(101)와 통신 채널들을 설정해야 한다. LL-DAX eSSD(101)의 네트워크 어드레스는 예를 들어, LL-DAX eSSD(101)의 네트워크 어드레스를 포함하는 스토리지 관리자에 의해 생성된 텍스트 파일과 같은 임의의 적합한 방법으로 LL-DAX 소프트웨어(103)에 제공될 수 있다. 발견 서비스는 스토리지 관리자에 의해 관리될 수 있다. 이러한 발견 방법들은 기존 스토리지 인프라스트럭처들(infrastructures)에서 일반적으로 사용된다.

[0060] 일단 LL-DAX 소프트웨어(103)가 LL-DAX eSSD(101)와의 통신 채널들을 발견하고 설정(establish)하면, LL-DAX 소프트웨어(103)는 원하는 수의 새로운 큐 쌍들(QPs)(120)을 LL-DAX eSSD(101)에 설정하는 것을 RDMA 전송 계층(104)에 요청한다. 일단 하나 이상의 QPs(120)가 설정되면, LL-DAX 소프트웨어(103)는 RDMA 읽기 요청을 사용하여 LL-DAX 식별 명령을 발행한다. LL-DAX 스토리지(109)에 관한 정보(예를 들어, LL-DAX 스토리지(109)의 식별 파라미터들) 예를 들어, LL-DAX 스토리지(109)의 다양한 성능들, 특징들 및 특성들을 전송함으로써 LL-DAX eSSD(101)는 LL-DAX 식별 명령에 응답한다. 따라서, LL-DAX 식별 명령 다음에, LL-DAX 드라이버는 데이터 스토

리지의 정상 동작동안 이용되는 근본적인 LL-DAX 스토리지(109)의 다양한 성능들, 특징들 및 특성들을 이해한다.

[0061] LL-DAX 소프트웨어(103)는 LL-DAX 스토리지(109)에 액세스하기 위한 애플리케이션들에게 인터페이스를 제공한다. LL-DAX는 커널 모듈이며 입/출력 제어(input/output control, IOCTL) 시스템 호출을 사용하여 읽기/쓰기 인터페이스를 제공한다. LL-DAX 소프트웨어(103)는 로드 스토어 시맨틱스(load store semantics)에서 동작하지 않는다. LL-DAX 드라이버는 LL-DAX 드라이버와 관련된 장치 핸들을 사용하여 애플리케이션들에 액세스할 수 있다. 애플리케이션들은 이러한 핸들과 LL-DAX 스토리지 어드레스들 사용하여 읽기 및 쓰기 동작들을 발행할 수 있다. 이러한 애플리케이션 요청들은 LL-DAX 소프트웨어(103)에 의해 적절한 RDMA 계층 요청들로 변환된다. RDMA 요청들은 LL-DAX 소프트웨어(103)에 의해 완료되도록 발행되고 추적된다. 요청들이 완료되면, 애플리케이션 호출들이 리턴된다. LL-DAX 스토리지(109)에 할당된 데이터 구조가 삭제되거나 할당 해제되면, LL-DAX 소프트웨어(103)는 이들 이벤트들을 LL-DAX 삭제 명령들로 변환하여 이들 명령들을 eSSD(101)에 전송한다. LL-DAX 삭제 명령은 eSSD들을 위해 사용되어 eSSD 내 백그라운드 작업들과 관련된 가비지 콜렉션(garbage collection)을 감소시킨다.

[0062] 또한, LL-DAX 소프트웨어(103)는 파일 시스템과 유사한 일부 기본 스토리지 공간 관리 기능들을 구현할 수 있다. 예를 들어, 하나 이상의 실시 예들에서, LL-DAX 소프트웨어(103)는 저장 공간 할당 및 할당 해제 기능들을 구현할 수 있다. 하나 이상의 실시예들에서, LL-DAX 소프트웨어(103)는 LL-DAX 스토리지(109)를 사용하여 애플리케이션들에 대한 기본 액세스 제어들을 수행할 수 있다. 애플리케이션이 일정량의 스토리지를 필요로 할 때, 애플리케이션은 원하는 양의 스토리지의 할당을 요청한다. LL-DAX 소프트웨어(103)는 LL-DAX 스토리지(109)에 대한 할당 테이블을 유지할 수 있다. LL-DAX 소프트웨어(103)는 이용 가능한 공간으로부터 요구된 스토리지 공간을 할당하고 그 공간에 대한 스토리지 핸들을 요구자에게 리턴한다. 그 다음 스토리지 핸들은 애플리케이션들에 의해서 후속 데이터 읽기/쓰기 액세스를 위해 사용된다. 또한, 하나 이상의 실시예들에서, LL-DAX 블록 스토리지 소프트웨어(103)는 더 많은 유연성, 데이터 접근성 및 공간 관리를 제공하기 위해 파일 정보 변환 기능들을 포함 할 수 있다. 따라서, 하나 이상의 실시예들에서, LL-DAX 애플리케이션은 자신의 LL-DAX 데이터를 다른 애플리케이션(예를 들어, 비-LL-DAX 애플리케이션)과 공유할 수 있다.

[0063] 본 발명은 예시적인 실시예들을 참조하여 설명되었지만, 당업자라면 본 발명의 사상 및 범위를 벗어나지 않고 서술된 실시예들에 대한 다양한 변경 및 수정이 수행될 수 있다는 것을 인식할 것이다. 또한, 다양한 기술 분야의 당업자는 여기에 설명된 본 발명이 다른 애플리케이션들에 대한 다른 태스크들 및 적응들에 대한 해결책을 제안함을 인식할 것이다. 출원인의 의도는 청구범위에 의해서 본 발명의 이러한 모든 용도, 및 본원에서 개시의 목적으로 선택된 본 발명의 예시적인 실시 예에 대해 행해질 수 있는 변경 및 수정이 모두 본 출원인의 의도이며, 본 발명의 사상 및 범위를 벗어나지 않는다. 따라서, 본 발명의 예시적인 실시 예는 모든면에서 예시적인 것으로서 제한적이지는 않으며, 본 발명의 사상 및 범위는 첨부 된 청구범위 및 그 균등물에 의해 나타내져야 한다.

부호의 설명

[0064] 100: 호스트 장치

101: LL-DAX eSSD

102: 애플리케이션

103: LL-DAX 블록 스토리지 소프트웨어

104: RDMA 전송 계층

105: RDMA 타겟 계층

106: LL-DAX 수신 버퍼

107: LL-DAX 호스트 인터페이스

108: LL-DAX 호스트 인터페이스

109: LL-DAX 스토리지

110: 파일 시스템 계층

111: 오퍼레이팅 시스템 계층

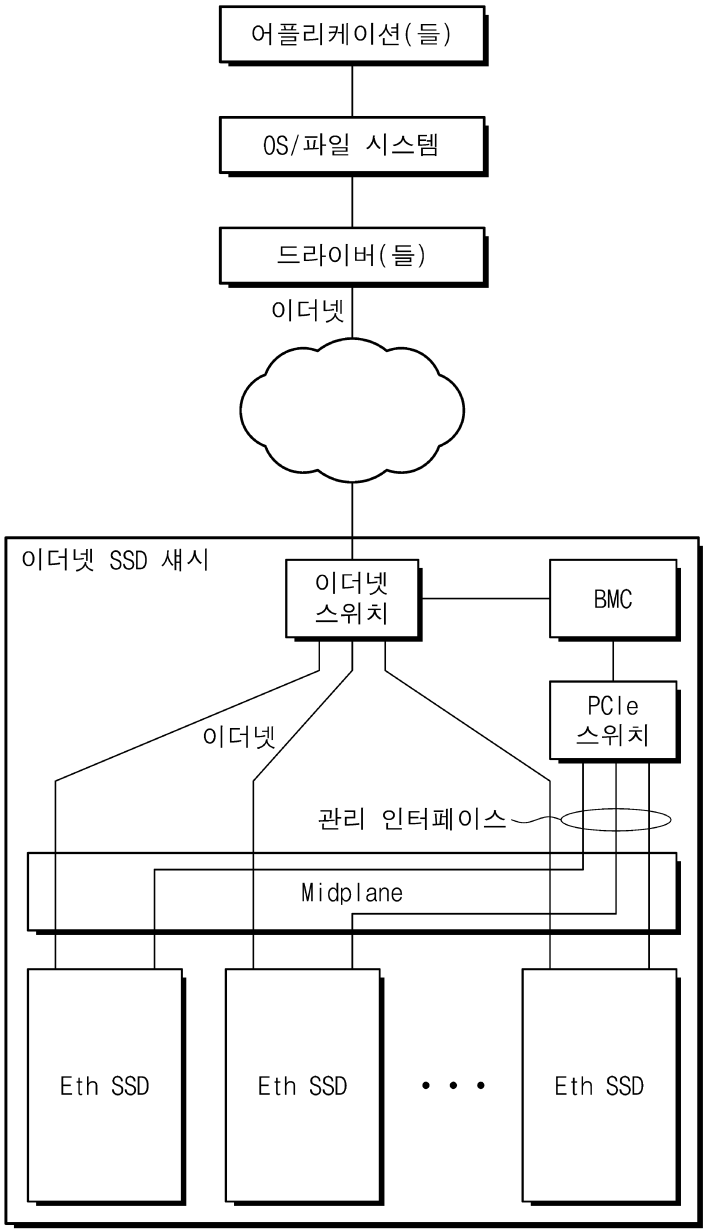
112: 블록 스토리지 계층

113: NVMe-oF 계층

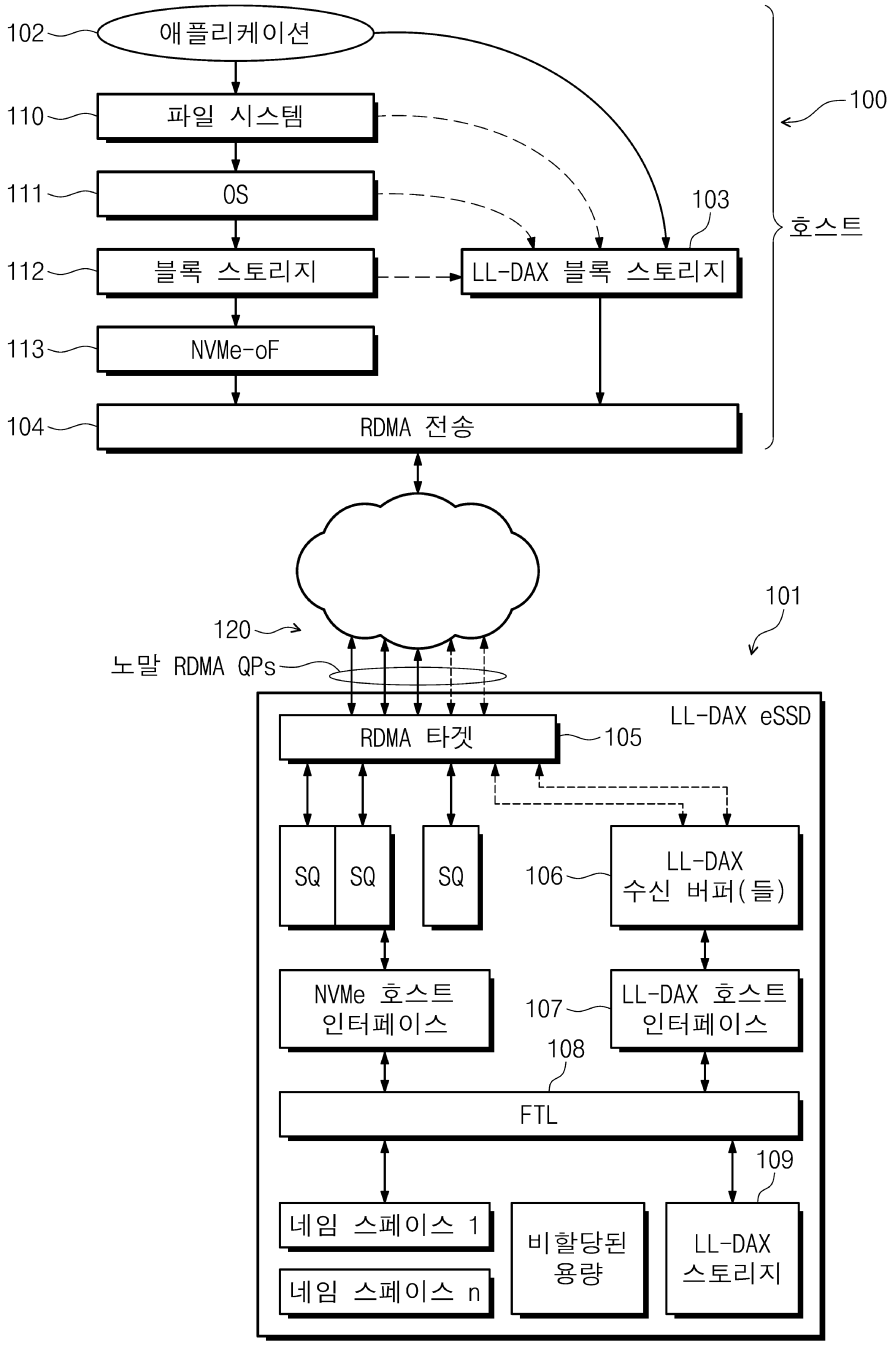
120: 큐 쌍들(QPs)

도면

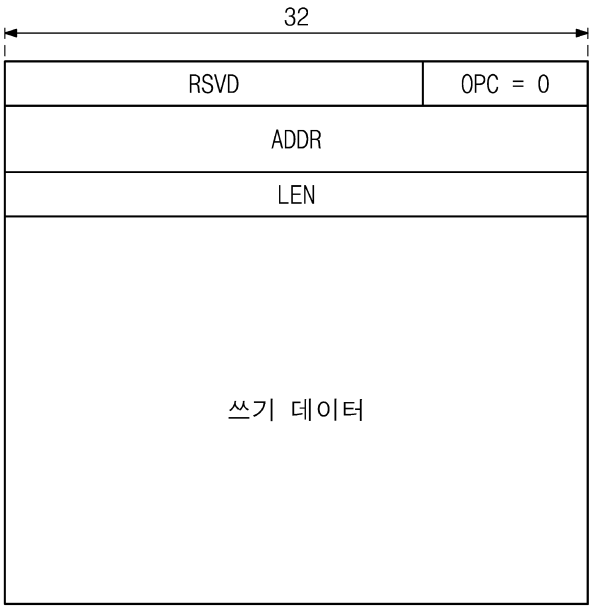
도면1



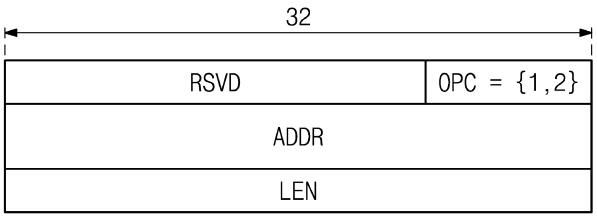
도면2


102
애플리케이션
110
파일 시스템
111
OS
112
블록 스토리지
113
NVMe-oF
104
RDMA 전송
103
LL-DAX 블록 스토리지
100
호스트
120
노말 RDMA QPs
101
LL-DAX eSSD
RDMA 타겟
105
SQ
SQ
SQ
106
LL-DAX
수신 버퍼(들)
NVMe 호스트
인터페이스
107
LL-DAX 호스트
인터페이스
108
FTL
109
네임 스페이스 1
네임 스페이스 n
비할당된
용량
LL-DAX
스토리지

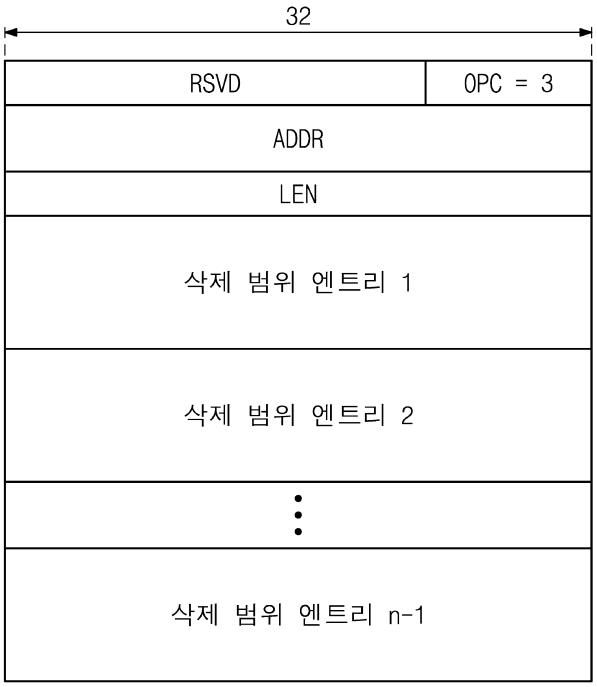
도면3


32
RSVD
OPC = 0
ADDR
LEN
쓰기 데이터

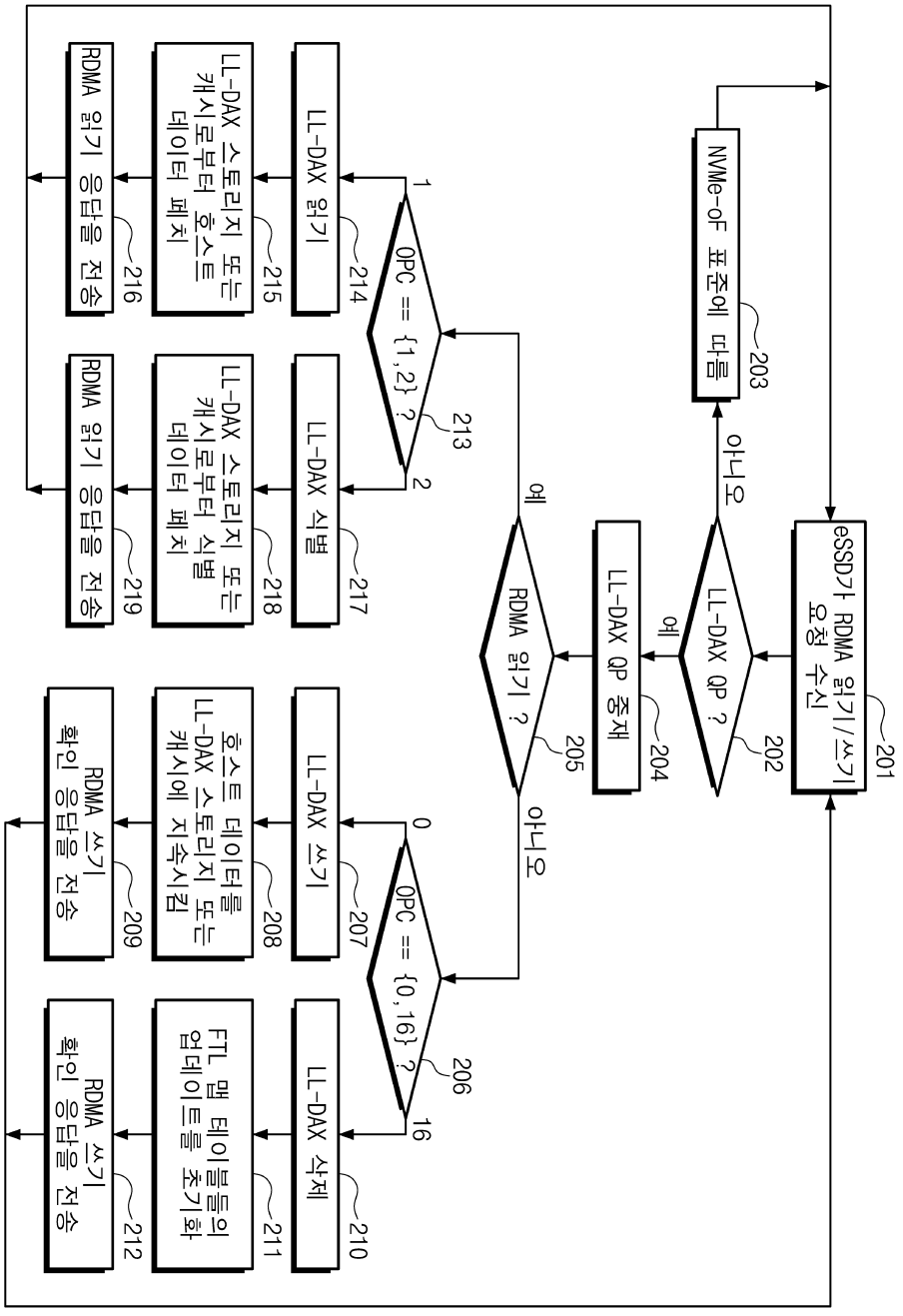
도면4


32
RSVD
OPC = {1,2}
ADDR
LEN

도면5


32
RSVD
OPC = 3
ADDR
LEN
삭제 범위 엔트리 1
삭제 범위 엔트리 2
삭제 범위 엔트리 n-1

도면6


eSSD가 RDMA 읽기/쓰기 요청 수신
201
LL-DAX QP ?
202
아니오
NVMe-oF 표준에 따름
203
예
LL-DAX QP 중재
204
RDMA 읽기 ?
205
예
QPC == {1,2} ?
213
1
LL-DAX 읽기
214
LL-DAX 스토리지 또는 캐시로부터 호스트 데이터 페치
215
RDMA 읽기 응답을 전송
216
2
LL-DAX 식별
217
LL-DAX 스토리지 또는 캐시로부터 식별 데이터 페치
218
RDMA 읽기 응답을 전송
219
아니오
QPC == {0,16} ?
206
0
LL-DAX 쓰기
207
호스트 데이터를 LL-DAX 스토리지 또는 캐시에 지속시킴
208
RDMA 쓰기 확인 응답을 전송
209
16
LL-DAX 삭제
210
FTL 맵 테이블들의 업데이트를 초기화
211
RDMA 쓰기 확인 응답을 전송
212