



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2022년08월22일

(11) 등록번호 10-2434917

(24) 등록일자 2022년08월17일

(51) 국제특허분류(Int. Cl.)
G06F 11/10 (2006.01) G06F 3/06 (2006.01)(52) CPC특허분류
G06F 11/1012 (2013.01)
G06F 3/0619 (2013.01)

(21) 출원번호 10-2019-0038752

(22) 출원일자 2019년04월02일

심사청구일자 2022년03월30일

(65) 공개번호 10-2019-0139752

(43) 공개일자 2019년12월18일

(30) 우선권주장
62/682,763 2018년06월08일 미국(US)
16/103,907 2018년08월14일 미국(US)(56) 선행기술조사문헌
US20170161148 A1
(뒷면에 계속)

(73) 특허권자

삼성전자주식회사

경기도 수원시 영통구 삼성로 129 (매탄동)

(72) 발명자

피추마니, 레카

미국 버지니아주 22033 페어팩스 웨이슨 플레이스
3871

기양석

미국 캘리포니아주 94303 팔로알토 알테어 워크
873

(74) 대리인

특허법인 고려

전체 청구항 수 : 총 19 항

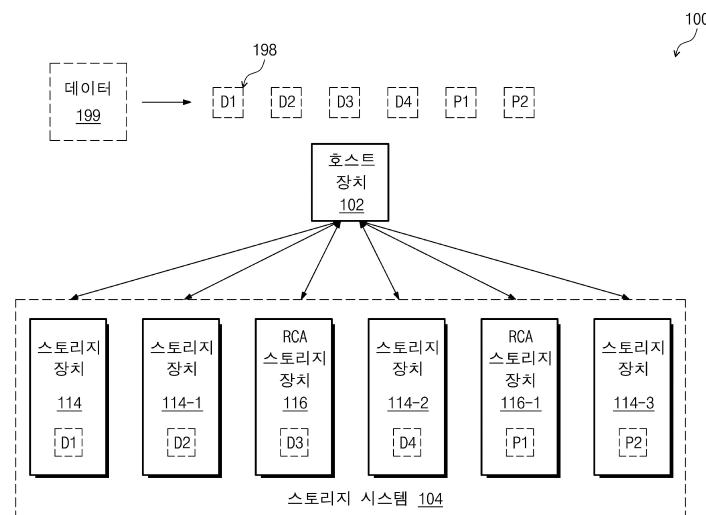
심사관 : 김계준

(54) 발명의 명칭 저대역 데이터 리페어를 보조하는 시스템, 장치, 및 방법

(57) 요약

본 발명의 일 실시 예에 따르면, 장치는 데이터 에러 정정을 위한 적어도 하나의 타입의 데이터 재생성 코드를 연산하도록 구성된 재생성 코드 인식(RCA; regeneration-code-aware) 스토리지 장치를 포함한다. RCA 스토리지 장치는 각각이 데이터 블록들을 포함하는 청크들로 데이터를 저장하도록 구성된 메모리, 외부 호스트 장치에 의해 요청된 경우, 선택된 다수의 데이터 블록들을 기반으로 데이터 재생성 코드를 연산하도록 구성된 프로세서, 및 데이터 재생성 코드를 외부 호스트 장치로 전송하도록 구성된 외부 인터페이스를 포함한다.

대표도 - 도1



(52) CPC특허분류
G06F 3/064 (2013.01)

(56) 선행기술조사문헌
US20130297579 A1
KR1020170129703 A
KR1020170097625 A
KR101609510 B1

명세서

청구범위

청구항 1

장치에 있어서,

데이터 에러 정정을 위한 적어도 하나의 타입의 데이터 재생성 코드를 연산하도록 구성된 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하고,

상기 RCA 스토리지 장치는:

데이터 블록들을 포함하는 청크들로 데이터를 저장하도록 구성된 메모리;

외부 호스트 장치와 연관된 요청을 기반으로, 선택된 다수의 데이터 블록들에 기반된 데이터 재생성 코드를 연산하도록 구성된 프로세서; 및

상기 외부 호스트 장치로 상기 데이터 재생성 코드를 전송하고, 상기 프로세서가 상기 데이터 재생성 코드를 연산하도록 구성하는 커맨드를 상기 외부 호스트 장치로부터 수신하도록 구성된 외부 인터페이스를 포함하는 장치.

청구항 2

제 1 항에 있어서,

상기 RCA 스토리지 장치는:

다른 데이터 재생성 코드를 생성하도록 구성된 명령어들의 하나 또는 그 이상의 세트들을 저장하도록 구성된 코드 메모리를 더 포함하고,

상기 프로세서는, 상기 외부 호스트 장치에 의해, 상기 명령어들의 하나 또는 그 이상의 세트들 중 하나를 선택하여 데이터 재생성 코드를 생성하도록 구성된 장치.

청구항 3

제 2 항에 있어서,

상기 코드 메모리는 상기 외부 호스트 장치에 의해 상기 코드 메모리로 기입된 상기 명령어들의 하나 또는 그 이상의 세트들을 포함하도록 구성된 장치.

청구항 4

제 1 항에 있어서,

상기 외부 인터페이스는 상기 선택된 다수의 데이터 블록들을 기반으로 상기 데이터 재생성 코드의 생성을 가능하게 하는 커맨드를 상기 외부 호스트 장치로부터 수신하도록 구성된 장치.

청구항 5

제 1 항에 있어서,

상기 외부 인터페이스는:

복구 데이터가 요구됨을 가리키고, 상기 데이터 재생성 코드가 연산되어야 하는지를 가리키는 복구 커맨드를 수

신하고;

상기 데이터 재생성 코드를 상기 외부 호스트 장치로 반환하도록 구성되고,

상기 데이터 재생성 코드의 크기는 상기 데이터의 크기보다 작은 장치.

청구항 6

제 1 항에 있어서,

상기 프로세서는, 상기 외부 호스트 장치에 의해 요청된 경우, 데이터 재생성 기법들을 통해 상기 데이터 재생성 코드의 다른 버전을 연산하도록 구성되고,

상기 프로세서의 의해 연산되는 상기 다른 버전은 상기 호스트 장치에 의해 지정되는 장치.

청구항 7

시스템에 있어서,

호스트 장치; 및

분산 스토리지 시스템을 포함하고,

상기 호스트 장치는:

상기 분산 스토리지 시스템에 데이터의 청크들로서 데이터를 저장하고;

데이터의 청크가 에러와 연관되는지 검출하고;

상기 에러의 상기 검출에 응답하여, 데이터 재생성 기법을 통해, 상기 데이터의 상기 청크들을 기반으로 상기 에러와 연관된 상기 데이터의 청크를 재구성하도록 구성되고,

상기 분산 스토리지 시스템은 상기 데이터의 각각의 청크를 저장하도록 구성된 스토리지 장치들을 포함하고,

상기 스토리지 장치들은 적어도 하나의 타입의 데이터 재생성 코드를 내부적으로 연산하도록 구성된 적어도 하나의 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하는 시스템.

청구항 8

제 7 항에 있어서,

상기 RCA 스토리지 장치는:

차례로 데이터 블록들을 포함하는 청크들에 데이터를 저장하도록 구성된 메모리;

선택된 다수의 데이터 블록들을 기반으로 데이터 재생성 코드를 연산하도록 구성된 프로세서; 및

상기 데이터 재생성 코드를 상기 호스트 장치로 전송하도록 구성된 외부 인터페이스를 포함하는 시스템.

청구항 9

제 7 항에 있어서,

상기 호스트 장치는:

데이터 재생성 코드를 내부적으로 연산할 수 있는 스토리지 장치를 판별하고;

상기 스토리지 장치로부터 데이터의 청크 또는 상기 데이터의 청크의 일부를 요청하고, 상기 호스트 장치에 의해 상기 데이터의 청크 또는 상기 데이터의 청크의 일부를 적어도 부분적으로 기반하여, 데이터 재생성 코드를

연산하도록 구성되는 시스템.

청구항 10

제 9 항에 있어서,

상기 호스트 장치는, 상기 스토리지 장치에서 가용한 데이터 재생성 기법, 상기 분산 스토리지 시스템과 연관된 가용한 대역폭의 크기, 상기 데이터의 청크 또는 상기 데이터의 청크의 부분의 크기와 비교되는 상기 데이터 재생성 코드의 크기, 및 상기 호스트 장치 내에 가용한 연산 전력의 크기 중 하나 또는 그 이상에 적어도 부분적으로 기반하여, 상기 스토리지 장치로 상기 데이터 재생성 코드의 상기 연산을 오프로드하는지 판별하도록 구성된 시스템.

청구항 11

제 9 항에 있어서,

상기 호스트 장치는 상기 호스트 장치에 의해 연산된 데이터 재생성 코드들 및 각각의 스토리지 장치들에 연산된 상기 데이터 재생성 코드들을 기반으로 상기 에러가 있는 청크를 재구성하도록 구성된 시스템.

청구항 12

제 9 항에 있어서,

상기 호스트 장치는:

제1 프로토콜을 통해 데이터 재생성 코드를 내부적으로 연산할 수 있는 제1 스토리지 장치와 통신하고,

제2 프로토콜을 통해 데이터 재생성 코드를 내부적으로 연산할 수 없는 제2 스토리지 장치와 통신하도록 구성된 시스템.

청구항 13

제 7 항에 있어서,

상기 호스트 장치는:

각각의 데이터 재생성 코드를 내부적으로 연산할 수 있는 스토리지 장치를 검출하고,

상기 스토리지 장치가 데이터 재생성 기법을 통해 상기 데이터 재생성 코드를 연산하도록 상기 각각의 스토리지 장치에 상기 데이터 재생성 기법과 연관된 명령어들을 저장하도록 구성된 시스템.

청구항 14

제 13 항에 있어서,

상기 호스트 장치는 호스트 장치에 의해 선택된 데이터 재생성 기법을 통해, 상기 데이터 재생성 코드를 연산할 수 있는 스토리지 장치를 검출함으로써, 적어도 부분적으로, 각각의 데이터 재생성 코드를 내부적으로 연산할 수 있는 스토리지 장치를 검출하도록 구성된 시스템.

청구항 15

시스템에 있어서,

호스트 장치; 및

스토리지 시스템을 포함하고,

상기 호스트 장치는:

상기 스토리지 시스템에서, 청크들로 데이터를 저장하고,

청크가 에러와 연관되었는지 검출하고,

상기 에러의 상기 검출에 응답하여, 데이터 재생성 기법을 통해, 상기 청크들을 적어도 부분적으로 기반하여, 상기 데이터를 정정하도록 구성되고,

상기 스토리지 시스템은:

상기 데이터의 각각의 청크를 저장하도록 구성된 스토리지 장치들을 포함하고,

상기 스토리지 장치는 데이터 재생성 코드의 적어도 하나의 타입을 내부적으로 연산하도록 구성된 적어도 하나의 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하고,

상기 RCA 스토리지 장치는:

데이터 블록들을 포함하는 청크들로 데이터를 저장하도록 구성된 메모리;

상기 호스트 장치와 연관된 요청을 기반으로, 선택된 다수의 데이터 블록들을 기반으로 데이터 재생성 코드를 연산하도록 구성된 프로세서;

다른 데이터 재생성 코드들을 생성하도록 구성된 하나 또는 그 이상의 명령어들의 세트들을 저장하도록 구성된 코드 메모리; 및

상기 호스트 장치로 상기 데이터 재생성 코드를 전송하도록 구성된 외부 인터페이스를 포함하는 시스템.

청구항 16

제 15 항에 있어서,

상기 호스트 장치는:

명령어들의 세트를 상기 RCA 스토리지 장치의 상기 코드 메모리로 기입하도록 구성되고,

상기 명령어들의 세트는 상기 RCA 스토리지 장치에 의해, 상기 데이터 재생성 기법을 통해 상기 연산을 가능하도록 구성된 시스템.

청구항 17

제 15 항에 있어서,

상기 호스트 장치는

각각의 RCA 스토리지 장치로 하나 또는 그 이상의 데이터 재생성 코드들의 상기 연산을, 적어도 부분적으로, 동적으로 오프로딩함으로써, 상기 에러를 정정하도록 구성된 시스템.

청구항 18

제 17 항에 있어서,

상기 호스트 장치는 상기 스토리지 시스템에서, RAC 스토리지 장치들이 아닌 스토리지 장치들에 의해 저장된 하나 또는 그 이상의 데이터의 청크들에 대해 데이터 재생성 코드를 상기 호스트 장치에 의해, 연산함으로써, 상기 에러를 정정하도록 구성되고,

상기 호스트 장치에 의한 연산은, 상기 스토리지 장치로부터 상기 데이터의 청크의 적어도 부분을 전송하는 것

을 포함하고,

상기 RCA 스토리지 장치에 의해 연산되고, 상기 호스트 장치로 전송되는 상기 데이터 재생성 코드의 크기는 상기 스토리지 장치로부터 상기 호스트 장치로 전송되는 상기 데이터의 청크들의 상기 적어도 부분의 크기보다 작은 시스템.

청구항 19

제 15 항에 있어서,

상기 스토리지 장치들은 비-재생성-코드-인식 스토리지 장치들을 포함하고,

상기 호스트 장치는:

제1 프로토콜을 통해 상기 RCA 스토리지 장치들과 통신하고,

제2 프로토콜을 통해 상기 비-RCA 스토리지 장치들과 통신하도록 구성되는 시스템.

청구항 20

삭제

발명의 설명

기술 분야

[0001] 본 발명은 데이터 스토리지에 관한 것으로, 더욱 상세하게는 저-대역 데이터 리페어를 보조하는 스토리지 장치를 위한 시스템, 장치, 및/또는 방법에 관한 것이다.

배경 기술

[0002] 코딩 이론(coding theory)에서, 소거 코드(erasure code)는 원본 메시지가 n 개의 심볼들의 일부로부터 복원될 수 있도록 k 개의 심볼들의 메시지를 n 개의 심볼들의 더 긴 메시지(코드 워드(code word))로 변환하는 비트 소거들(비트 에러들 대신에)의 가정하에서의 순방향 에러 정정(FEC; forward error correction) 코드이다. $r=k/n$ 의 분수는 코드율(code rate)라 불린다. k'/k 의 분수(이 때, k' 는 복원을 위해 요구되는 심볼들의 개수를 가리킴.)는 수신 효율(reception efficiency)이라 불린다.

[0003] 재생성 코드들은 기존의 인코딩된 조각들(fragments)로부터 손실된 인코딩된 조각들을 재구성(또는 복구(repairing)라 불림.)하는 이슈를 해결한다. 좀 더 상세하게는, 재생성 코드들은 종래의 MDS(maximum distance separable) 코드의 스토리지 효율성은 유지하면서, 복구 동안 다운로드의 크기를 감소시키는 것을 목표로 하는 코드들의 클래스이다. 이러한 이슈는 인코딩된 리던던시를 유지하는 통신이 문제가 되는 분산 스토리지 시스템들에서 발생한다.

[0004] 분산 스토리지 시스템(distributed storage system)은 일반적으로 정보가 하나 이상의 노드들 또는 장치들에 종종 복제 방식으로 저장되는 컴퓨터 네트워크이다. 분산 스토리지 시스템은 사용자들이 다수의 노드들에 정보를 저장하는 분산 데이터 베이스 또는 사용자들이 다수의 피어 네트워크 노드들에 정보를 저장하는 컴퓨터 네트워크 중 하나를 나타내는데 사용된다. 분산 스토리지 시스템들은 일반적으로 에러 검출 및 정정 기법을 사용한다. 일부 분산 스토리지 시스템은, 파일의 부분들이 손상되거나 또는 가용할 수 없는 경우, 순방향 에러 정정 기법들(forward error correction techniques)을 사용하여 원본 파일, 청크 또는 부분을 복구한다. 다른 사용자들은 다른 미러로부터 파일을 다운로드하려고 다시 시도한다.

발명의 내용

해결하려는 과제

[0005] 본 발명의 목적은 본 발명의 실시 예들에 따르면, 향상된 신뢰성 및 향상된 성능을 갖는 저대역 데이터 리페어

를 보조하는 시스템, 장치, 및 방법을 제공하는데 있다.

과제의 해결 수단

- [0006] 본 발명의 일 실시 예에 따르면, 장치는 데이터 에러 정정을 위한 적어도 하나의 타입의 데이터 재생성 코드를 연산하도록 구성된 재생성 코드 인식(RCA; regeneration-code-aware) 스토리지 장치를 포함할 수 있다. 상기 RCA 스토리지 장치는 각각이 데이터 블록들을 포함하는 청크들로 데이터를 저장하도록 구성된 메모리를 포함할 수 있다. 상기 RCA 스토리지 장치는 외부 호스트 장치에 의해 요청된 경우, 선택된 다수의 데이터 블록들을 기반으로 데이터 재생성 코드를 연산하도록 구성된 프로세서를 포함할 수 있다. 상기 RCA 스토리지 장치는 상기 데이터 재생성 코드를 상기 외부 호스트 장치로 전송하도록 구성된 외부 인터페이스를 포함할 수 있다.
- [0007] 본 발명의 일 실시 예에 따르면, 시스템은 호스트 장치 및 분산 스토리지 시스템을 포함할 수 있다. 상기 호스트는 상기 분산 스토리지 시스템 사이에서 복수의 청크들로서 데이터를 저장하고, 적어도 하나의 청크가 에러와 연관된 경우를 검출하고, 상기 에러에 응답하여, 데이터 재생성 기법을 통해, 적어도 상기 데이터의 상기 복수의 청크들을 기반으로 상기 데이터와 연관된 상기 적어도 하나의 청크를 재구성하도록 구성될 수 있다. 상기 분산 스토리지 시스템은 복수의 스토리지 장치들을 포함할 수 있다. 상기 복수의 스토리지 장치들 각각은 적어도 상기 데이터의 각각의 청크를 저장하도록 구성될 수 있다. 상기 복수의 스토리지 장치들은 적어도 하나의 재생성 코드 인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하고, 상기 적어도 하나의 RCA 스토리지 장치 각각은 데이터 재생성 코드의 적어도 하나의 타입을 내부적으로 연산하도록 구성된다.
- [0008] 본 발명의 일 실시 예에 따르면, 시스템은 호스트 장치, 및 스토리지 시스템을 포함할 수 있다. 상기 호스트 장치는 상기 스토리지 시스템 사이에서 복수의 청크들에 데이터를 저장하고, 적어도 하나의 청크가 에러와 연관된 것을 검출하고, 상기 에러에 응답하여, 데이터 재생성 기법을 통해 상기 데이터의 상기 복수의 청크들을 기반으로 상기 에러를 정정하도록 구성될 수 있다. 상기 스토리지 시스템은 복수의 스토리지 장치들을 포함할 수 있다. 상기 복수의 스토리지 장치들 각각은 상기 데이터의 적어도 각각의 청크를 저장하도록 구성되고, 상기 복수의 스토리지 장치들은 적어도 하나의 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함할 수 있다. 상기 RCA 스토리지 장치 각각은 적어도 하나의 타입의 데이터 재생성 코드를 내부적으로 연산하도록 구성될 수 있다. 상기 RCA 스토리지 장치는 청크들에 데이터를 저장하도록 구성된 메모리를 포함할 수 있다. 상기 청크들 각각은 데이터 블록들을 포함할 수 있다. 상기 RCA 스토리지 장치는 상기 호스트 장치에 의해 요청된 경우, 선택된 다수의 데이터 블록들을 기반으로 데이터 재생성 코드를 연산하도록 구성된 프로세서를 포함할 수 있다. 상기 RCA 스토리지 장치는 복수의 명령어들의 세트들을 저장하도록 구성된 코드 메모리를 포함할 수 있다. 상기 복수의 명령어들의 세트들 각각은 다른 데이터 재생성 코드를 생성할 수 있다. 상기 RCA 스토리지 장치는 상기 데이터 재생성 코드를 상기 호스트 장치로 전송하도록 구성된 외부 인터페이스를 포함할 수 있다.
- [0009] 하나 또는 그 이상의 구현들의 상세한 설명들은 첨부된 도면들 및 이하의 상세한 설명에서 제시된다. 다른 특징들은 상세한 설명 및 도면들 그리고 특허청구범위로부터 나타날 것이다.
- [0010] 실질적으로 도면들 중 적어도 하나와 연관되어 설명되거나 또는 도시되고, 특허청구범위에서 좀더 완전하게 나타나는 바와 같이, 본 발명은 데이터 스토리지를 위한 시스템 및/또는 방법, 좀 더 상세하게는 스토리지 장치에 의한 저-대역폭 데이터 복구에 관한 것이다.

발명의 효과

- [0011] 본 발명의 실시 예들에 따르면, 향상된 신뢰성 및 향상된 성능을 갖는 저대역 데이터 리페어를 보조하는 시스템, 장치, 및 방법이 제공된다.

도면의 간단한 설명

- [0012] 도 1은 본 발명의 실시 예에 따른 시스템의 예시적인 실시 예의 블록도이다.
- 도 2a는 본 발명의 실시 예에 따른 시스템의 예시적인 실시 예의 블록도이다.
- 도 2b는 본 발명의 실시 예에 따른 시스템의 예시적인 실시 예의 블록도이다.
- 도 2c는 본 발명의 실시 예에 따른 시스템의 예시적인 실시 예의 블록도이다.
- 도 2d는 본 발명의 실시 예에 따른 시스템의 예시적인 실시 예의 블록도이다.

도 3은 본 발명의 실시 예에 따른 기법의 예시적인 실시 예의 순서도이다.

도 4는 본 발명의 실시 예의 원리들에 따라 형성된 장치들을 포함할 수 있는 정보 처리 시스템의 개념적인 블록도이다.

다양한 도면들에서 유사한 참조 번호는 유사한 요소들을 가리킨다.

발명을 실시하기 위한 구체적인 내용

- [0013] 다양한 예시적인 실시 예들이 일부 예시적인 실시 예들을 보여주는 첨부된 도면들을 참조하여 더욱 상세하게 설명된다. 그러나, 본문에 제시된 내용은 다양한 다른 형태로 구현될 수 있고, 본문에 설명된 예시적인 실시 예들에 제한되는 것으로 해석되어서는 안 된다. 대신에, 이러한 예시적인 실시 예들은 상세한 설명이 완전하며, 당업자에 의해 본 발명의 기술적 사상이 완전하게 전달될 수 있도록 제공된다. 도면들에서, 계층들 및 영역들의 크기들 및 상대적인 크기들은 명확성을 위하여, 과장된다.
- [0014] 구성 또는 계층이 다른 구성 또는 계층과 "연결된(on, connected to, or coupled to)" 것으로 지칭되는 경우, 이는 직접적으로(directly) 다른 구성 또는 계층과 연결될 수 있거나 또는 하나 또는 그 이상의 중간(intervening) 구성들 또는 계층들이 존재할 수 있음이 이해될 것이다. 대조적으로, 구성이 다른 구성 또는 계층과 "직접적으로 연결된(directly on, directly connected to, or directly coupled to)" 것으로 지칭되는 경우, 중간 구성들 또는 계층들이 존재하지 않는다. 유사한 참조 번호는 전체적으로 유사한 구성들을 지칭한다. 본문에서 사용되는 바와 같이 "및/또는(and/or)"은 연관되어 나열된 목록들 중 하나 또는 그 이상의 조합들의 일부 및 전부를 포함한다.
- [0015] "제1(first)", "제2(second)", "제3(third)" 등과 같은 용어들은 다양한 요소들, 구성들, 영역들, 계층들, 및/또는 구역들을 설명하기 위하여 본문에서 사용되나, 이러한 요소들, 구성들, 영역들, 계층들, 및/또는 구역들은 이러한 용어들에 한정되지 않음이 잘 이해될 것이다. 이러한 용어들은 하나의 요소, 구성, 영역, 계층, 또는 구역을 다른 하나의 요소, 구성, 영역, 계층, 또는 구역으로부터 구분하기 위해서만 사용된다. 즉, 이하에서 기재되는 제1 요소, 구성, 영역, 계층, 또는 구역은 본 발명의 사상 및 범위로부터 벗어남 없이, 제2 요소, 구성, 영역, 계층, 또는 구역으로 불릴 수 있다.
- [0016] "~의 아래(beneath, below, lower, under)", "~의 위(above, upper)" 등과 같은 공간적으로 상대적인 용어들(spatially relative terms)은 도면들에서 도시된 다른 하나의 요소(들) 또는 특징(들)과 하나의 요소 또는 특징들과의 관련성을 용이하게 설명하기 위하여 본문에서 사용될 수 있다. 공간적으로 상대적인 용어들은 도면에 도시된 지향성에 추가적으로 동작 또는 사용에서 장치의 다른 지향성(orientations)을 포함하는 것으로 의도됨이 잘 이해될 것이다. 예를 들어, 도면들에서 장치가 뒤집어지는 경우, 다른 요소들 또는 특징들의 "아래(below or beneath or under)"로 설명된 요소들은 다른 요소들 또는 특징들의 "상부(above)"로 향할 것이다. 즉, "아래(below, under)"의 예시적인 용어들은 위 및 아래의 방향들을 모두 포함할 수 있다. 장치가 다른 방향(예를 들어, 90도로 회전하거나 또는 다른 방향)을 지향할 수 있고, 본문에서 사용되는 공간적으로 상대적인 설명들은 이에 따라 해석되어야 한다.
- [0017] 마찬가지로, "하이(high)", "로우(low)", "풀-업(pull up)", "풀 다운(pull down)", "1", 및 "0" 등과 같은 전기적인 용어들 도면에서 도시된 바와 같이, 다른 전압 레벨들 또는 다른 구성들 또는 특징들과 상대적인 전압 레벨 및 전류를 나타내는 설명의 편의를 위하여 상세한 설명에서 사용될 수 있다. 전기적으로 상대적인 용어들은 도면들에 도시된 전압들 또는 전류들에 추가적으로 사용 또는 동작에서 장치의 다른 기준 전압들을 포함하는 것으로 의도된다. 예를 들어, 도면들에서 장치 또는 신호들이 반전되거나 또는 다른 기준 전압들, 전류들, 또는 전하들을 사용하는 경우, "하이(high)" 또는 "풀 업(pull up)"으로 설명되는 구성들은 새로운 기준 전압 또는 전류와 비교하여 "로우(low)" 또는 풀 다운(pull down)"일 수 있다. 즉, "하이(high)"의 예시적인 용어는 상대적으로 낮거나 또는 높은 전압 또는 전류를 모두 포함할 수 있다. 장치는 다른 전기적은 프레임들의 기준에 기반될 수 있고, 상세한 설명에서 사용되는 전기적으로 상대적인 설명들에 따라 해석될 수 있다.
- [0018] 상세한 설명에서 사용되는 용어들은 오직 특정 실시 예에 대한 설명의 목적을 위한 것이며, 본 발명의 한정을 의도하는 것은 아니다. 상세한 설명에서 사용되는 바와 같이, 단수 형태들(singular forms)은, 명확히 다르게 정의되지 않는 한, 복수 형태들(plural forms)을 포함하도록 의도된다. "포함하다(comprise)" 용어는, 상세한 설명에 사용되는 경우, 열거된 특징들, 단계들, 동작들, 요소들, 및/또는 구성들의 존재를 특정하나, 하나 또는 그 이상의 다른 특징들, 단계들, 동작들, 요소들, 구성들, 및/또는 그것들의 그룹들의 추가 또는 존재를 배제하

지 않는다.

- [0019] 예시적인 실시 예들이 이상적인 실시 예들(및 중간 구조들)의 예시적인 도면인 단면도를 참조하여 상세한 설명에서 설명된다. 이와 같이, 예를 들어 제조 기술 및/또는 허용 오차와 같은 결과로서의 도면들의 형상으로부터의 변형이 예상되어야 한다. 즉, 예시적인 실시 예들은 상세한 설명에 도시된 특정 형상의 영역들에 한정되는 것으로 해석되어서는 안되며, 예를 들어 제조로부터 초래되는 형상의 편차를 포함해야 한다. 예를 들어, 직사각형으로 도시된 주입된 영역은 일반적으로, 주입된 영역에서부터 주입되지 않은 영역으로의 이진 변화 라기보다는, 둥근 또는 곡선의 피쳐 및/또는 모서리에서의 주입 농도의 기울기를 가질 것이다. 마찬가지로, 주입에 의해 형성된 매립 영역은 매립 영역과 주입이 일어나는 표면 사이의 영역에 약간의 주입을 초래할 수 있다. 따라서, 도면들에 도시된 영역들은 본질적으로 개략적이며, 그 형상들은 디바이스의 영역의 실제 형상을 설명하기 위한 것이 아니며 본 발명의 범위를 제한하려는 것이 아니다.
- [0020] 다르게 정의되지 않는 한, 본문에서 사용되는 모든 용어들(기술적 및 과학적 용어들을 포함함)은 당업자에 의해 공통적으로 이해될 수 있는 의미를 갖는다. 또한, 일반적으로 사용되는 사전에서 정의된 용어와 같은 용어는 관련 기술 및/또는 본 명세서와 관련하여 그 의미와 일치하는 의미를 갖는 것으로 해석되어야 하며, 본문에서 정의되지 않는 한 이상적이거나 지나치게 형식적인 의미로 해석되지 않아야 한다.
- [0021] 이하에서, 예시적인 실시 예들이 첨부된 도면들을 참조하여 상세하게 설명된다.
- [0022] 도 1은 본 발명의 실시 예에 따른 시스템(100)의 예시적인 실시 예의 블록도이다. 도시된 실시 예에서, 시스템(100)은 복수의 노드들 또는 스토리지 장치들을 통해 데이터를 저장하는 분산 스토리지 시스템(104)을 포함할 수 있다.
- [0023] 분산 스토리지 시스템들은 대규모 신뢰성 스토리지(large-scale reliability storage)를 제공하는데 종종 사용된다. 종종, 이는 리던던시(redundancy) 또는 여러 정정(예를 들어, 패리티(parity))를 다수의 노드들 또는 스토리지 장치들에 걸쳐 분산시킴으로써 달성된다. 그러나 노드 또는 스토리지 장치가 오프-라인이 된 경우(예를 들어, 네트워크 에러, 하드웨어 고장 등에 기인함), 데이터가 손상될 가능성이 있거나 또는 적어도 리던던시의 레벨이 감소될 수 있다. 스토리지 시스템이 분산될수록, 이러한 상황이 좀 더 자주 발생한다.
- [0024] 이러한 상황을 방지하기 위하여 다양한 기법들(예를 들어, 미러링, 리드-솔로몬 인코딩)이 사용될 수 있으나, 본 발명의 실시 예들은 재생성 인코딩에 중점을 둔다. 이러한 실시 예에서, 데이터의 누락된 부분(missing piece)(청크(chunk)은 데이터의 나머지 부분들에 기반된 방식(formular)을 사용하여 재생성되거나 또는 재구성된다.
- [0025] 도시된 실시 예에서, 시스템(100)은 분산 스토리지 시스템(104)을 관리하도록 구성된 하나 또는 그 이상의 호스트 장치들(102)을 포함할 수 있다. 호스트 장치(102)는 스토리지 시스템(104)을 읽고 쓰는 컴퓨팅 장치(예를 들어, 컴퓨터, 서버, 가상 머신)를 포함할 수 있다. 에러(예를 들어, 데이터의 누락된 청크)가 발생하는 경우, 호스트 장치(102)는 일반적으로 에러를 검출하고, 가능하다면 에러를 정정할 책임이 있다.
- [0026] 도시된 실시 예에서, 각 데이터 세트(199)는 호스트 장치(102)에 의해 복수의 더 작은 부분들의 데이터 또는 청크들(198)로 쪼개지거나 또는 세분화(fragmented)될 수 있다. 도시된 실시 예에서, 데이터(199)는 청크들(198)(D1, D2, D3, D4)로 분할된다. 더욱이, 다양한 실시 예들에서, 호스트 장치(102)는 패리티 청크들(P1, P2)(패리티 청크들 또한 청크들이므로, 198의 참조번호로 표기됨.)과 같은 일부 형태의 리던던시를 데이터 청크들(198)로 적용할 수 있다.
- [0027] 본문에서, 원본 데이터 청크들(198)(D1, D2, D3, D4)의 개수는 변수(K 또는 k)로 설명된다. 마찬가지로, 리던던트 데이터 청크들(198)(P1, P2)의 개수들은 변수(R 또는 r)로 설명된다. 이러한 청크들의 전체 개수는 K+R개이다. 도시된 실시 예에서, K는 4이고, R은 2이고, K+R은 6이나, 이는 단순히 예시적인 실시 예이며, 본 발명이 이에 한정되지 않음이 이해될 것이다.
- [0028] 도시된 실시 예에서, 호스트 장치(102)는 이러한 청크들(198)(원본 및 리던던트 모두)의 각각을 스토리지 시스템(104)의 각각의 노드들 또는 스토리지 장치들에 저장한다. 도시된 실시 예에서, 스토리지 장치(114)는 청크(198)(D1)를 저장하고, 스토리지 장치(114-1)는 청크(198)(D2)를 저장하고, 스토리지 장치(116)는 청크(198)(D3)를 저장하고, 스토리지 장치(114-2)는 청크(198)(D4)를 저장하고, 스토리지 장치(116-1)는 청크(198)(P1)를 저장하고, 스토리지 장치(114-3)는 청크(198)(P2)를 저장한다. 다양한 실시 예들에서, 스토리지 장치들(114/116)의 개수는 청크들(198)의 개수와 동일하지 않을 수 있다.

- [0029] 다양한 실시 예들에서, 청크(198)가 소실될 수 있거나(예를 들어, 네트워크 또는 하드웨어 고장), 또는 그렇지 않다면 예러와 연관될 수 있다. 도시된 실시 예에서, 청크(198)(D3)(및 스토리지 장치(116))가 갑자기 사용될 수 없는 것으로 가정한다. 호스트 장치(102)는, 예러가 검출된 경우, 청크(198)(D3)를 재생성하거나 또는 그렇지 않다면 예러를 정정하도록 시도할 수 있다.
- [0030] 이러한 실시 예에서, 하나의 청크(예를 들어, 청크(198)(D3))가 고장(fail)이고, 원본 데이터(199)에 K(예를 들어, 4)개의 전체 청크들이 존재한다면, 고장인 청크(예를 들어, 청크(198)(D3))를 복구하기 위하여, 적어도 K(예를 들어, 4)개의 노드들 또는 스토리지 장치들(114/116)은 호스트 장치(102)로 정보를 전송하여야 한다. 이러한 K(예를 들어, 4)개의 청크들은 K+R(예를 들어, 6)개의 청크들 중 어느 것으로부터 올 수 있다. 예를 들어, 청크들(198)(D1, D2, D4, P1)이 청크(198)(D3)를 재생성하는데 사용될 수 있다.
- [0031] 재생성 코드들(regeneration codes)은 D개의 노드들(단, 일반적으로 $D > K$)로부터 정보를 전체 청크 크기 정보보다 덜 전송함으로써 복구 대역폭(repair bandwidth)을 감소시킨다. 다시 말해서, 현명한 방식(clever formula)의 사용에 의해, 호스트 장치(102)는 전체 청크들(198)(D1, D2, D4, P1)을 사용하지 않고, 청크들(198)(D1, D2, D4, P1, P2) 중 오직 일부만을 사용함으로써, 누락된 청크(198)(D3)를 재생성할 수 있다. 재생성 코드는 일반적으로 더 많은 스토리지 장치들(114/116)로부터 정보를 획득하나, 비-재생성 코드가 취하는 것 보다 더 적은 정보를 각각의 스토리지 장치(114/116)로부터 획득한다.
- [0032] 예를 들어, 데이터의 6개 청크들이 사용되고($K=6$), 리턴던시의 6개 청크들이 사용되고($R=6$, $K+R=12$), 각 청크의 크기가 16MB 인 경우, 표준 RS(Reed-Solomon) 예러 정정 인코딩 방식은 누락된 16MB의 청크를 정정하기 위해 6개(K)의 16MB의 청크들이 호스트로 전송되거나 또는 96MB의 데이터가 전송되는 것을 필요로 한다. 이와 반대로, 재생성 기법이 사용되는 경우, 모든 12개(이 경우, $K+R$ 또는 D)의 청크들의 부분들이 읽어지나, 오직 각 청크의 부분만 사용되기 때문에(예를 들어, 2.7MB), 호스트 장치로 전송된 전체 크기는 작아질 수 있다(예를 들어, 29.7MB).
- [0033] 재생성 코드들은 스토리지 및 대역폭의 트레이드 오프를 수반한다. 다양한 실시 예들에서, 일반적으로 재생성 코드들의 2개의 클래스들 또는 그룹들이 존재한다. 스토리지 오버헤드가 최소인 경우, 그것들은 MSR(Minimum Storage Regeneration) 코드들이라 불린다. 복구 대역폭이 추가된 스토리지 오버헤드에 대하여 최소인 경우, 그것들은 MBR(Minimum Bandwidth Regeneration) 코드들이라 불린다. 이러한 넓은 카테고리들에서, 다양한 특정 기법들 또는 방식들이 재생성 코드들을 수행하는데 사용될 수 있다. 상술된 내용은 단순한 일부 예시적인 실시 예들이며, 본 발명이 이에 한정되지 않음이 이해될 것이다.
- [0034] 도 1을 다시 참조하면, 도시된 실시 예에서, 스토리지 시스템(104)은 복수의 스토리지 장치들(114, 116)을 포함할 수 있다. 각 스토리지 장치(114, 116)는 청크들로 또는 다르게 데이터를 저장하도록 구성될 수 있다. 도시된 실시 예에서, 스토리지 장치들(114)은 하드 디스크 드라이브들, 솔리드 스테이트 드라이브들, 또는 휘발성 메모리들과 같은 상대적으로 종래의 스토리지 장치들일 수 있다.
- [0035] 그러나 도시된 실시 예에서, 스토리지 시스템(104)은 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치들(116)을 포함할 수 있다. 이러한 실시 예에서, 종래의 또는 비-RCA 스토리지 장치들(114)과 달리, RCA 스토리지 장치들(116)은 데이터 재생성 코드의 연산을 돕도록 구성되거나 또는 그렇게 하는 구성 요소들을 포함할 수 있다. 이하에서 좀 더 상세하게 설명되는 바와 같이, 호스트 장치(102)는 데이터 재생성 코드의 연산의 일부를 RCA 스토리지 장치(116)로 동적으로 오프로드(offload)시킬 수 있다. 다양한 실시 예들에서, 이는 호스트 장치(102) 및 스토리지 시스템(104) 사이에서 송수신되는 메시지의 크기, 호스트 장치(102) 및 스토리지 시스템(104) 사이에서 전송되는 데이터의 크기, 및/또는 호스트 장치(102)에서의 연산 부하를 감소시킬 수 있다. 상술된 내용은 단순히 일부 예시적인 실시 예들이며, 본 발명의 범위가 이에 한정되지 않음이 이해될 것이다.
- [0036] 다양한 실시 예들에서, RCA 스토리지 장치들(116)은 호스트 장치(102)가 최신 또는 의도한 재생성 코드 방식 또는 기법으로 그것들을 업데이트할 수 있도록 프로그램될 수 있다. 이러한 실시 예에서, RCA 스토리지 장치들(116)은 다양한 재생성 기법들을 저장할 수 있고, 재생성 기법들 중 하나가 호스트 장치(102)에 의해 동적으로 또는 반-고정적으로(semi-statically) 선택되도록 할 수 있다. 이러한 실시 예에서, 호스트 장치(102)는 현재 어떤 재생성 기법이 사용될 수 있는지 선택할 수 있다.
- [0037] 다양한 실시 예들에서, 스토리지 시스템(104)은 분산될 수 있다. 이러한 실시 예에서, 스토리지 장치들(114/116)은 서로 물리적으로 이격될 수 있고, 네트워크 프로토콜을 통해 통신할 수 있다. 다른 실시 예에서, 스토리지 장치들(114/116)은 상대적으로 국부적으로 (예를 들어, 서버 팜 또는 동일한 건물 내에) 배치될 수 있

으나, 여전히 네트워크 프로토콜을 통해 통신할 수 있다. 또 다른 실시 예에서, 스토리지 시스템(104)은 분산되지 않을 수 있다. 이러한 실시 예에서, 본 발명의 실시 예는 네트워크 프로토콜을 사용하지 않는(예를 들어, USB, SATA) 로컬 장치(예를 들어, 동일한 머신)를 위하여 사용될 수 있다. 상술된 내용은 단순히 일부 예시적인 실시 예들이며, 본 발명이 이에 한정되지 않음이 이해될 것이다.

[0038] 다양한 실시 예들에서, 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치(116)는 재생성 코드들의 다른 타입들 또는 버전들을 연산하는 능력을 포함할 수 있다. 이러한 실시 예에서, 원하는 재생성 코드의 타입 또는 버전은 호스트 장치(102)에 의해 동적으로 선택될 수 있다. 일부 실시 예들에서, RCA 스토리지 장치(116)는 데이터를 더 작은 블록들 또는 패킷들로 분할하고, 소거 코드 또는 그것들의 부분을 연산하고, 다른 고장인 청크의 복구를 위해 데이터 청크(들)를 처리 것 등을 할 수 있다.

[0039] 다양한 실시 예들에서, 통신 프로토콜은 재생성 코드들 또는 기법들을 사용하여 데이터 신뢰성을 구현하기 위해 호스트 장치(102) 및 RCA 스토리지 장치(116) 사이에 존재할 수 있다. 이러한 실시 예들에서, 프로토콜은 재생성 기법을 선택하는 것, 입력들을 통과시키는 것, 의도된 기법의 동작들을 지시하는 것, 및 출력들을 회수하는 것을 허용할 수 있다. 일부 실시 예들에서, 프로토콜은 RCA 및 비-RCA 스토리지 장치들(116/114) 모두를 포함하는 조합된 환경에서 호스트가 동작할 때의 호스트 작용(host behavior) 및 양쪽 모두와 상호 작용하는 방법을 정의할 수 있다. 다양한 실시 예들에서, 호스트 시스템(102)은 프로토콜을 사용하여 RCA 스토리지 장치(116)를 셋업하고, 사용자 데이터를 인코딩하고/읽고/쓰고, 데이터 복구 동안 연산을 오프로드하고, 데이터 트래픽을 감소시키고, 연산을 가속하고, RCA 스토리지 장치의 기능을 사용하여 원본 데이터를 재구성할 수 있다.

[0040] 도 2a는 본 발명의 실시 예에 따른 시스템(201)의 예시적인 실시 예의 블록도이다. 도시된 실시 예에서, 시스템(201)은 제1 타입(Type 1) 재생성 코드를 연산하는 호스트 장치(210) 및 스토리지 장치(212) 사이의 상호 작용(interaction)을 보여준다. 다양한 실시 예들에서, 시스템(201)은 종래의 또는 비-RCA 스토리지 장치들에 사용될 수 있으며, RCA 기능이 사용되지 않는다면 RCA 스토리지 장치들에서도 사용될 수 있다.

[0041] 도시된 실시 예에서, 시스템(201)은 호스트 장치(210) 및 스토리지 장치(212)를 포함할 수 있다. 이러한 실시 예에서, 호스트 장치(210)는 명령어들을 실행하고 연산들을 수행하는 프로세서(232), 데이터 또는 데이터의 일부를 적어도 임시적으로 저장하는 메모리(234), 및 스토리지 장치(212) 또는 좀 더 일반적으로 스토리지 시스템(미도시)과 통신하는 인터페이스(236)를 포함할 수 있다. 이러한 실시 예에서, 스토리지 장치(212)는 데이터를 저장하도록 구성된 메모리(224)를 포함할 수 있다. 다양한 실시 예들에서, 이러한 메모리(224)는 불휘발성 또는 휘발성일 수 있다.

[0042] 도시된 실시 예에서, 청크들(214)은 블록들(216)로 다시 분할된다. 이러한 실시 예에서, 호스트 장치(210)는 스토리지 장치(212)에 저장된 하나 또는 그 이상의 청크들(214)로부터 (그리고 다른 스토리지 장치들에 저장된 K-1개의 청크들로부터) 블록들(216)을 획득하고 재생성 코드(218)(R1)를 연산할 수 있다.

[0043] 이러한 재생성 코드 기법(Type 1)에서, 블록들(216)은 더 작은 패킷들(미도시)로 구성될 수 있다. 각 노드 또는 스토리지 장치(212)에 대하여, 호스트 장치(210)는, 다양한 패킷들을 사용하여, 패리티 패킷 또는 재생성 코드(218)를 연산한다. 각 스토리지 장치의 각각의 재생성 코드(218)는 누락되거나 또는 오류가 난 청크(missing or errored chunk)를 재구성하는데 사용된다. 일반적으로, 제1 타입(Type 1) 재생성 코드 기법들에 대하여, 연산은 선형이고 고장인 청크들에 의존한다. 다시 전송된 데이터의 크기는 서브-패킷화(sub-packetization) 레벨 및 기능에 의존한다.

[0044] 도시된 실시 예에서, 호스트 장치(210)가 에러를 검출한 경우, 호스트(210)는 데이터 읽기 요청 또는 커맨드(242A)를 스토리지 장치(212)로 전송할 수 있다. 데이터 읽기 커맨드(242A)는 어떤 청크(214)가 읽어들지에 대한 정보(예를 들어, 청크(214C))를 포함할 수 있다. 이후에, 스토리지 장치(212)는, 데이터 읽기 요청 또는 메시지(244A)를 통해, 의도된 청크(214)를 호스트 장치(210)로 전송한다. 다양한 실시 예들에서, 이것은 종래의 호스트와 스토리지 장치(212)사이의 프로토콜(예를 들어, SATA)을 사용하여 모두 행해질 수 있다.

[0045] 의도된 청크(214C)가 수신된 경우, 호스트 장치(210)는 인터페이스(236)에 의해 청크(214C) 또는 블록들(216)을 메모리(234)에 저장할 수 있다. 프로세서(232)는, 이후에, 의도된 재생성 코드 기법(287)을 수행할 수 있다. 재생성 코드 기법(287)은 단순한 덧셈 또는 불 방식의 XOR 연산(Boolean XORing)으로 도시되나, 이는 단순한 일부 예시적인 실시 예들이며, 본 발명이 이에 한정되지 않음이 이해될 것이다. 상술된 바와 같이, 다양한 실시 예들에서, 이것은 블록들(216)을 더 작은 패킷들로 다시 분할하는 것을 포함할 수 있다. 재생성 코드 기법(287)은 재생성 코드(218)(R1)를 연산하거나 또는 생성할 수 있고, 재생성 코드(218)(R1)는 다른 청크들 또는 스토리지

장치들과 연관된 재생성 코드들과 함께 오류가 난 청크를 재구성하거나 또는 복구하는데 사용될 수 있다.

- [0046] 도 2b는 본 발명의 실시 예에 따른 시스템(203)의 예시적인 실시 예의 블록도이다. 도시된 실시 예에서, 시스템(203)은 제1 타입(Type 1) 재생성 코드를 연산하는 호스트 장치(210) 및 RCA 스토리지 장치(252) 사이의 상호작용을 보여 준다. 다양한 실시 예들에서, 시스템(203)은 오직 RCA 스토리지 장치들에 대해서만 사용될 수 있고, 비-RCA 장치에 대해서는 사용되지 않는다.
- [0047] 도시된 실시 예에서, 시스템(201)은 호스트 장치(210) 및 RCA 스토리지 장치(252)를 포함할 수 있다. 이러한 실시 예에서, 호스트 장치(210)는 명령어들을 실행하고 연산들을 수행하는 프로세서(232), 데이터 또는 데이터의 부분들을 적어도 임시적으로 저장하는 메모리(234), 및 RCA 스토리지 장치(252) 또는 좀 더 일반적으로 스토리지 시스템(미도시)와 통신하는 인터페이스(236)를 포함할 수 있다.
- [0048] 이러한 실시 예에서, RCA 스토리지 장치(252)는 데이터를 저장하도록 구성된 메모리(224)를 포함할 수 있다. 다양한 실시 예들에서, 메모리(224)는 불휘발성 또는 휘발성일 수 있다. 더욱이, 다양한 실시 예들에서, RCA 스토리지 장치(252)는, 호스트 장치(210)(일반적으로 스토리지 장치의 외부에 위치함.)에 의해 요청된 경우, 선택된 다수의 데이터 블록들(216)을 기반으로 데이터 재생성 코드(218)를 연산하도록 구성된 프로세서(222)를 포함할 수 있다. 다양한 실시 예들에서, 프로세서(222)는 프로그램 가능한 게이트 어레이(예를 들어, FPGA), 그래픽 프로세서 유닛(GPU; graphic processor unit), 범용 프로세서(예를 들어, CPU), 컨트롤러 프로세서, 또는 시스템-온-칩(SoC; system-on-chip)을 포함할 수 있다. 상술된 내용은 단순히 일부 예시적인 실시 예들이며, 본 발명이 이에 한정되지 않음이 이해될 것이다. RCA 스토리지 장치(252)는 명령어들의 복수의 세트들을 저장하도록 구성된 코드 메모리(228)를 포함할 수 있다. 명령어들의 각 세트(229)는 다른 재생성 코드 기법을 수행하는 방법에 대한 명령어들 또는 다른 데이터 재생성 코드를 생성한다. 다양한 실시 예들에서, 명령어들의 세트(229)는 스토리지 장치(252)로 미리 구성될 수 있거나 또는 구동 중에 (예를 들어, 호스트 장치(210)에 의해) 동적으로 추가/조절될 수 있거나 또는 그것들의 조합의 형태로 구현될 수 있다. RCA 스토리지 장치(252)는 적어도 호스트 장치(210)와 통신하도록 구성된 외부 인터페이스(226)를 포함할 수 있다.
- [0049] 도시된 실시 예에서, 호스트 장치(210)는 스토리지 장치(252)가 데이터 재생성 코드를 내부적으로 연산할 수 있는지 또는 일반적으로 RCA 스토리지 장치인지 판별할 수 있다. 만약 그렇다면, 호스트 장치(210)는 RCA 스토리지 장치(252)가 의도된 재생성 코드 기법을 수행할 수 있는지 또는 의도된 재생성 코드 기법을 수행하도록 (코드 메모리(228)를 통해) 프로그램되어 있는지 판별할 수 있다. 만약 그렇지 않다면, 도 2a에 도시된 기법이 사용될 수 있다.
- [0050] RCA 스토리지 장치(252)가 의도된 재생성 코드 기법을 수행할 수 있는 경우, 호스트 장치(210)는 복구를 위한 읽기 커맨드(242B)(Read for Repair command)를 발행할 수 있다. 다양한 실시 예들에서, 복구를 위한 읽기 커맨드(242B)는 의도된 재생성 또는 복구 기법의 지시(indication), 의도된 패킷 또는 블록 크기, 의도된 재생성 코드 또는 복구 기법에 대한 파라미터들, 데이터 또는 청크 어드레스, 고장인 청크 넘버 중 하나 또는 그 이상을 포함하거나 또는 가리킬 수 있다. 이는 단순히 일부 예시적인 실시 예들이며, 본 발명이 이에 한정되지 않음이 이해될 것이다.
- [0051] 커맨드(242B)에 응답하여, 프로세서(222)는 의도된 블록(216) 또는 청크(214C)를 회수할 수 있다. 프로세서(222)는 의도된 재생성 또는 복구 기법과 연관된 명령어들의 세트(229)를 회수할 수 있다. 프로세서(222)는 의도된 재생성 기법(287)을 수행하고, 데이터 재생성 코드(DRC)(218)(R1)를 연산할 수 있다.
- [0052] 이후에, RCA 스토리지 장치(252)는 호스트 장치(210)로, 인터페이스(226)를 통해, 데이터 재생성 코드(218)(R1)(메시지(244B))를 전송할 수 있다. 이러한 실시 예에서, 데이터 재생성 코드(218)(R1)는 도 2a의 메시지(244A)를 통해 전송되는 데이터보다 더 작은 크기를 갖거나 또는 더 적은 대역폭을 사용할 수 있다.
- [0053] 도시된 실시 예에서, 메시지들(242B, 244B)은 메시지들(242A, 244A)에 대하여 사용된 것과 다른 프로토콜을 필요로 할 수 있다. 메시지들(242A, 244A)은 종래의 스토리지 장치 프로토콜에 의해 허용될 수 있는 반면에, 메시지들(242B, 244B)은 추가적이고 다른 정보를 필요로 할 수 있고, 이에 따라 새로운 메시징 프로토콜 또는 적어도 새로운 커맨드들을 필요로 할 수 있다.
- [0054] 도시된 실시 예에서, 호스트 장치(210)는 다른 RCA 스토리지 장치들(미도시)에 의해 제공되거나 또는 호스트 장치(210) 자체에 의해 생성된 추가적인 데이터 재생성 코드들과 함께 데이터 재생성 코드(218)(R1)를 사용하여 오류가 난 데이터의 청크를 재생성 할 수 있다.
- [0055] 도 2c는 본 발명의 실시 예에 따른 시스템(205)의 예시적인 실시 예의 블록도이다. 도시된 실시 예에서, 시스템

(205)은 제2 타입(Type 2) 재생성 코드를 연산하는 호스트 장치(210) 및 스토리지 장치(212) 사이의 상호 작용을 보여준다. 다양한 실시 예들에서, 시스템(205)은 종래의 또는 비-RCA 스토리지 장치들에 대하여 사용될 수 있으며, RCA 기능이 사용되지 않는 경우, RCA 스토리지 장치들에 대해서도 사용될 수 있다.

[0056] 도시된 실시 예에서, 시스템(205)은 호스트 장치(210) 및 스토리지 장치(212)를 포함할 수 있다. 호스트 장치(210) 및 스토리지 장치(212) 모두는 앞서 설명되고 도시된 구성 요소들을 포함할 수 있다.

[0057] 이러한 재생성 코드 기법(Type 2)에서, 데이터 재생성 코드들은 더 적은 패킷들(미도시) 또는 블록들(216)이 읽히도록 연산된다. 그러나 이는 의도된 블록들(216) 또는 패킷들이 사전에 완전히 알려지나, 연산이 수행됨에 따라 단편적으로 요청됨을 의미한다. 이러한 타입의 재생성 기법은 네트워크 대역폭 및 데이터 읽기 모두를 이론상 감소시키나, 하나의 큰 읽기를 복수의 더 작은 읽기들로 변환하며, 이는 성능에 좋지 않다.

[0058] 도시된 실시 예에서, 호스트 장치(210)는, 의도된 재생성 기법의 일부(288)를 사용하여, 블록(E1)이 에러와 연관되어 있는 경우, 블록들(B1, B3)(또는 그것들의 패킷들)이 에러가 난 블록(E1)을 교정하는데 필요한 것으로 계산한다. 이러한 실시 예에서, 호스트 장치(210)가 블록(B1)이 필요한 것으로 검출한 경우, 호스트 장치(210)는 데이터 읽기 요청 또는 커맨드(242C)를 스토리지 장치(212)로 전송할 수 있다. 데이터 읽기 커맨드(242C)는 어떤 블록(216)이 읽어질지(예를 들어, 블록(B1))에 대한 정보를 가리킬 수 있다. 스토리지 장치(212)는, 이후에, 의도된 블록(216)(B1)을, 데이터 읽기 응답 또는 메시지(244C)를 통해, 호스트 장치(210)로 전송한다. 다양한 실시 예들에서, 이는 종래의 호스트 및 스토리지 장치 사이의 프로토콜(예를 들어, SATA)을 사용하여 모두 행해질 수 있다.

[0059] 이러한 실시 예에서, 호스트 장치(210)가 블록(B3)이 필요한 것을 검출한 경우, 호스트 장치(210)는 데이터 읽기 요청 또는 커맨드(246C)를 스토리지 장치(212)로 전송할 수 있다. 이는 일반적으로 블록(B1)을 요청하는 것과 별도로 제2 데이터 요청으로서 수행된다. 데이터 읽기 커맨드(246C)는 어떤 블록(215)이 읽어질지(예를 들어, 현재 블록(B3))에 대한 정보를 가리킬 수 있다. 스토리지 장치(212)는, 이후에, 데이터 읽기 응답 또는 메시지(248C)를 통해 의도된 블록(216)(B3)을 호스트 장치(210)로 전송한다. 다양한 실시 예들에서, 이는 종래의 호스트 및 스토리지 장치 사이의 프로토콜(예를 들어, SATA)을 사용하여 모두 행해질 수 있다.

[0060] 의도된 블록들(216)이 수신된 경우, 인터페이스(236)에 의해 호스트 장치(210)는 블록들(216)을 메모리(234)에 저장할 수 있다. 프로세서(232)는, 이후에, 의도된 재생성 코드 기법(부분(289)에 의해 도시됨)을 수행할 수 있다. 재생성 코드 기법(또는 부분(289))은 재생성 코드(219)(R1)를 연산하거나 또는 생성할 수 있고, 재생성 코드(219)(R1)는 다른 청크들 또는 스토리지 장치들과 연관된 재생성 코드들과 함께 에러가 난 청크를 재구성하거나 복구하는데 사용될 수 있다.

[0061] 도 2d는 본 발명의 실시 예에 따른 시스템(207)의 예시적인 실시 예의 블록도이다. 도시된 실시 예에서, 시스템(207)은 제2 타입(Type 2)의 재생성 코드를 연산하는 호스트(210) 및 RCA 스토리지 장치(252) 사이의 상호 작용을 보여준다. 다양한 실시 예들에서, 시스템(207)은 RCA 스토리지 장치들에 대해서만 사용될 수 있으며, 비-RCA 스토리지 장치들에 대해서는 사용되지 않는다.

[0062] 도시된 실시 예에서, 시스템(207)은 호스트 장치(210) 및 스토리지 장치(252)를 포함할 수 있다. 호스트 장치(210) 및 스토리지 장치(252) 모두는 앞서 설명되고 도시된 구성 요소들을 포함할 수 있다.

[0063] 도시된 실시 예에서, 호스트 장치(210)는 스토리지 장치(252)가 데이터 재생성 코드를 내부적으로 연산할 수 있는지, 또는 일반적으로 RCA 스토리지 장치인지 판별할 수 있다. 만약 그렇다면, 호스트 장치(210)는 RCA 스토리지 장치(252)가 의도된 재생성 코드 기법을 수행할 수 있는지 또는 의도된 재생성 코드 기법을 수행하도록 (코드 메모리(228)를 통해) 프로그램될 수 있는지 판별할 수 있다. 만약 그렇지 않다면, 도 2C에 도시된 기법이 사용될 수 있다.

[0064] RCA 스토리지 장치(252)가 의도된 재생성 코드 기법을 수행할 수 있는 경우, 호스트 장치(210)는 복구를 위한 읽기 커맨드(242D)(Read for Repair command)를 발행할 수 있다. 다양한 실시 예들에서, 복구를 위한 읽기 커맨드(242D)는 의도된 재생성 또는 복구 기법의 지시(indication), 의도된 패킷 또는 블록 크기, 의도된 재생성 또는 복구 기법을 위한 파라미터들, 데이터 또는 청크 어드레스, 및 고장인 청크 넘버(예를 들어, 블록(E1)) 중 하나 또는 그 이상을 포함하거나 또는 가리킬 수 있다. 상술된 내용은 단순히 일부 예시적인 실시 예들이며, 본 발명의 범위가 이에 한정되지 않음이 이해될 것이다.

[0065] 커맨드(242D)에 응답하여, 프로세서(222)는 의도된 재생성 또는 복구 기법과 연관된 명령어들의 세트(229)를 회수할 수 있다. 프로세서(222)는 의도된 재생성 기법 또는 그것의 일부(288)를 수행할 수 있다. 이러한 실시

예에서, 프로세서(222)는 의도된 블록들이 B1 및 B3임을 연산할 수 있다. 이러한 실시 예에서, 이러한 블록들(B1, B3)은 RCA 스토리지 장치(252)에 의해 연산된 데이터 재생성 코드에 포함될 수 있다. 이러한 실시 예에서, 이러한 블록들은 복구를 위한 읽기 커맨드(242D)의 응답의 일부로 간주될 수 있다.

[0066] RCA 스토리지 장치(252)는, 이후에, 인터페이스(226)를 통해 의도된 블록들(B1, B3)(예를 들어, 메시지(244D)을 호스트 장치(210)로 전송할 수 있다. 이러한 실시 예에서, 데이터 재생성 코드 또는 의도된 블록들(B1, B3)은 더 작은 크기를 갖거나 또는 더 적은 대역폭을 사용하거나 또는 적어도 더 작은 메시지를 포함할 수 있고, 이에 따라 도 2c의 메시지들(244C, 248C)을 통해 전송된 데이터보다 오버헤드가 적을 수 있다.

[0067] 도시된 실시 예에서, 메시지들(242D, 244D)은 메시지들(242C, 244C, 246C, 248C)을 위해 사용되는 것과 다른 프로토콜을 필요로 할 수 있다. 메시지들(242C, 244C, 246C, 248C)은 종래의 스토리지 장치 프로토콜에 의해 허용될 수 있으나, 메시지들(242D, 244D)은 추가적이고 다른 정보를 필요로 할 수 있고, 이에 따라 새로운 메시지 프로토콜, 또는 적어도 새로운 커맨드들이 요구될 수 있다.

[0068] 도시된 실시 예에서, 호스트 장치(210)는, 이후에, 데이터 재생성 코드 또는 블록들(B1, B3)을 다른 RCA 스토리지 장치들(미도시)에 의해 제공되거나 또는 호스트 장치(210) 자체에 의해 생성된 추가적인 재생성 코드들 또는 데이터와 함께 사용하여 에러가 난 데이터(E1)를 재생성할 수 있다.

[0069] 도 3은 본 발명의 실시 예에 따른 기법(300)의 예시적인 실시 예의 순서도이다. 다양한 실시 예들에서, 기법(300)은 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 것들과 같은 시스템들에 의해 사용되거나 또는 생산될 수 있다. 상술된 내용은 단순히 일부 예시적인 실시 예들이며, 본 발명이 이에 한정되지 않음이 이해될 것이다. 본 발명이 기법(300)에 의해 도시된 다수의 동작들의 순서에 한정되지 않음이 이해될 것이다.

[0070] 도시된 실시 예에서, 설명의 편의를 위하여, 기법(300)은 스토리지 시스템의 모든 장치들이 RCA 스토리지 장치들 또는 비-RCA 스토리지 장치들 중 하나인 실시 예(즉, 동종 스토리지 시스템(homogeneous storage system))를 보여준다. 조합되거나 또는 이종 스토리지 시스템(heterogeneous storage system)에 대해서는, 본 발명이 속하는 기술 분야의 당업자가 단순화된 기법(300)이 개별적인 스토리지 장치들로 적용되도록 확장될 수 있는 방법을 잘 이해할 것이다.

[0071] 일 실시 예에서, 블록(302)은 데이터의 청크와 연관된 에러가 검출될 수 있음을 보여준다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0072] 일 실시 예에서, 블록(304)은, 상술된 바와 같이, 데이터 재생성 코드(DRC)가 호스트 장치에 의해 또는 각각의 RCA 스토리지 장치들에 의해 연산된 것인지가 판별되는 것을 보여준다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0073] 일 실시 예에서, 블록(306)은 RCD가 호스트에 의해 좀 더 일반적인 방식으로 연산되는 경우, RCD를 연산하는데 충분한 기존 데이터가 존재하는지가 판별될 수 있는 것을 보여준다. 이러한 실시 예에서, 이것은 K개의 청크들이 K+R개의 데이터 청크들로부터(out of the K+R data chunks) 가용한지 판별하는 것을 포함할 수 있다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0074] 일 실시 예에서, 블록(309)은 RCD를 연산하는데 에러-프리 청크들이 충분히 존재하지 않는 경우, 에러인 데이터의 청크들의 재생성을 넘어 다른 형태의 에러 처리가 발생할 수 있음을 보여준다. 다양한 실시 예들에서, 이것은 단순히 데이터가 오류를 갖거나 또는 가용할 수 없음을 리포팅하는 것이다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0075] 일 실시 예에서, 블록(308)은 요구된 개수의 청크들(예를 들어, K개의 청크들)이 다양한(예를 들어, K+R개) 스토리지 장치들로부터 읽어들일 수 있음을 보여준다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0076] 일 실시 예에서, 블록(310)은, 상술된 바와 같이, 호스트 장치가 에러-프리 청크들(예를 들어, K개의 청크들)을 사용하여 에러가 난 청크를 재구성하거나 또는 재생성할 수 있음을 보여준다. 다양한 실시 예들에서, 이러한 블

록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0077] 일 실시 예에서, 블록(350)은, 상술된 바와 같이, 예러-프리 청크들(예를 들어, D개의 청크들)이 DRC를 연산하는데 충분히 존재하는지가 판별될 수 있는 것을 보여준다. 만약 그렇지 않다면, 다양한 실시 예들에서, 기법(300)은 블록(306)에서 시작하는 비-RCA 장치 경로를 시도하는 것에 기댈 수 있다. 그 외에는, 기법(300)은 블록(352)으로 계속될 수 있다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0078] 일 실시 예에서, 블록(352)은, 상술된 바와 같이, 복구를 위한 읽기 커맨드(read for repair command)가 전체(예를 들어, K+R개) 스토리지 장치들 중 필요한 개수(예를 들어, D개)로 발행될 수 있음을 보여준다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0079] 일 실시 예에서, 블록(354)은, 상술된 바와 같이, 복수의 버전들 또는 타입들의 DRC 기법 중 어느 것이 사용될지가 판별될 수 있음을 보여준다. 도시된 실시 예에서, DRC 기법들의 버전들 및 타입들은, 상술된 바와 같이, 제1 타입(Type 1) 및 제2 타입(Type 2)으로 일반화된다. 그러나 이러한 타입들은 단순히 일부 예시적인 실시 예들이며, 본 발명이 이에 한정되는 것은 아니며, 또한 넓은 범위의 타입들에서, 상술된 바와 같이, 많은 서브-타입들이 존재할 수 있음이 이해될 것이다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0080] 일 실시 예에서, 블록(356)은 제1 타입 DRC 기법(Type 1 DRC technique)이 선택된 경우, 상술된 바와 같이, 복구 기능(repair function)이 RCA 스토리지 장치들 내에서 청크들로 적용될 수 있음을 보여준다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0081] 일 실시 예에서, 블록(358)은 제2 타입 DRC 기법(Type 2 DRC technique)이 선택된 경우, 상술된 바와 같이, 복구를 위해 요구되는 블록들(또는 패킷과 같은 다른 서브-부분들)이 연산될 수 있음을 보여준다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0082] 일 실시 예에서, 블록(360)은 DRC 또는 요구되는 블록들이 연산된 경우, 상술된 바와 같이, DRC 또는 블록들이 호스트 장치로 전송될 수 있음을 보여준다. 다양한 실시 예들에서, 이것은, 상술된 바와 같이, 비-RCA 경로보다 더 작은 크기의 데이터 또는 더 작은 개수의 메시지들을 포함할 수 있다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0083] 일 실시 예에서, 블록(362)은, 상술된 바와 같이, 호스트 장치가 DRC 또는 반환된 블록들을 사용하여 에러가 난 청크를 재구성하거나 또는 재생성할 수 있음을 보여준다. 다양한 실시 예들에서, 이러한 블록에 의해 도시된 하나 또는 그 이상의 동작(들)은, 상술된 바와 같이, 도 1, 도 2a, 도 2b, 도 2c, 및 도 2d의 시스템들 또는 장치들에 의해 수행될 수 있다.

[0084] 다양한 실시 예들에서, 호스트 장치는 데이터 재생성 코드를 연산하는 RCA 스토리지 장치(또는 그것에 포함된 프로세서)의 기능을 온 또는 오프시키는 커맨드를 RCA 스토리지 장치로 제공할 수 있다. RCA 스토리지 장치(또는 그것에 포함된 프로세서)는, 커맨드에 응답하여, 도 2b 또는 도 2d에 도시된 RCA 스토리지 장치와 같이 동작하거나 또는 도 2a 또는 도 2c에 도시된 종래의 또는 비-RCA 스토리지 장치와 같이 동작할 수 있다.

[0085] 도 4는 본 발명의 기술적 사상에 따라 형성된 반도체 장치들을 포함할 수 있는 정보 처리 시스템(400)의 개념적인 블록도이다.

[0086] 도 4를 참조하면, 정보 처리 시스템(400)은 본 발명의 기술적 사상에 따라 구성된 하나 또는 그 이상의 장치들을 포함할 수 있다. 다른 실시 예에서, 정보 처리 시스템(400)은 본 발명의 기술적 사상에 따른 하나 또는 그 이상의 기법들을 사용하거나 또는 실행할 수 있다.

[0087] 다양한 실시 예들에서, 정보 처리 시스템(400)은, 예를 들어, 랩탑(laptop), 데스크 탑, 워크스테이션, 서버,

블레이드 서버, 개인용 디지털 보조 장치(PDA; personal digital assistant), 스마트폰, 태블릿, 및 다른 적절한 컴퓨터들과 같은 컴퓨팅 장치 또는 가상 머신 또는 그것들의 가상 컴퓨팅 장치를 포함할 수 있다. 다양한 실시 예들에서, 정보 처리 시스템(400)은 사용자(미도시)에 의해 사용될 수 있다.

[0088] 본 발명에 따른 정보 처리 시스템(400)은 중앙 처리 유닛(CPU; central processing unit), 로직, 또는 프로세서(410)를 더 포함할 수 있다. 일부 실시 예들에서, 프로세서(410)는 하나 또는 그 이상의 기능 유닛 블록들(FUBs) 또는 조합 로직 블록들(CLBs)(415)을 포함할 수 있다. 이러한 실시 예들에서, 조합 로직 블록은 다양한 불 방식의 로직 동작들(예를 들어, NAND, NOR, NOT, XOR), 안정화 로직 장치들(예를 들어, 플립-플롭들, 래치들), 다른 로직 장치들, 또는 그것들의 조합을 포함할 수 있다. 이러한 조합 로직 동작들은 단순하거나 또는 복잡한 방식으로 입력 신호들을 처리하여 의도된 결과를 달성하도록 구성될 수 있다. 동기식 조합 로직 동작들의 일부 예시적인 실시 예들이 설명되었으나, 본 발명이 이에 한정되는 것은 아니며, 비동기식 동작들, 또는 그것들의 조합을 포함할 수 있음이 이해될 것이다. 일 실시 예에서, 조합 로직 동작들은 복수의 CMOS(complementary metal oxide semiconductors) 트랜지스터들을 포함할 수 있다. 다양한 실시 예들에서, 이러한 CMOS 트랜지스터들은 로직 동작들을 수행하는 게이트들로 정렬될 수 있으나, 단, 본 발명의 범위에 속하는 다른 기술들이 사용될 수 있음이 이해된다.

[0089] 본 발명에 따른 정보 처리 시스템(400)은 휘발성 메모리(420)(예를 들어, 랜덤 액세스 메모리(RAM; random access memory))를 더 포함할 수 있다. 본 발명에 따른 정보 처리 시스템(400)은 불휘발성 메모리(430)(예를 들어, 하드 드라이브, 광 메모리, NAND 또는 플래시 메모리)를 더 포함할 수 있다. 일부 실시 예들에서, 휘발성 메모리(420), 불휘발성 메모리(430), 또는 그것들의 조합 또는 일부들은 "저장 매체(storage medium)"으로 지칭될 수 있다. 다양한 실시 예들에서, 휘발성 메모리(420) 및/또는 불휘발성 메모리(430)는 반-영구적인 또는 실질적으로 영구적인 형태로 데이터를 저장하도록 구성될 수 있다.

[0090] 다양한 실시 예들에서, 정보 처리 시스템(400)은 정보 처리 시스템(400)이 통신 네트워크의 일부분 이도록 또는 통신 네트워크를 통해 통신하도록 구성된 하나 또는 그 이상의 네트워크 인터페이스(440)를 포함할 수 있다. Wi-Fi 프로토콜의 예시들은 IEEE(Institute of Electrical and Electronics Engineers) 802.11g, IEEE 802.11n을 포함할 수 있으나, 이에 한정되지 않는다. 셀룰러 프로토콜의 예시들은 IEEE 802.16m (일명, Wireless-MAN (Metropolitan Area Network) Advanced), LTE(Long Term Evolution) Advanced, Enhanced Data rates for GSM(Global System for Mobile Communications) Evolution (EDGE), Evolved High-Speed Packet Access(HSPA+)을 포함할 수 있으나, 이에 한정되지 않는다. 유선 프로토콜의 예시들은 IEEE 802.3(일명, Ethernet), 파이버 채널(Fibre Channel), 전력 라인 통신(Power Line communication)(예를 들어, 홈 플러그, IEEE 1901)를 포함할 수 있으나, 이에 한정되지 않는다. 상술된 내용은 단순히 일부 예시적인 실시 예들이며, 본 발명이 이에 한정되지 않음은 이해될 것이다.

[0091] 본 발명에 따른 정보 처리 시스템(400)은 사용자 인터페이스 유닛(450)(예를 들어, 디스플레이 어댑터, 햅틱 인터페이스, 휴먼 인터페이스 장치)를 더 포함할 수 있다. 다양한 실시 예들에서, 이러한 사용자 인터페이스 유닛(450)은 사용자로부터 입력을 수신하거나 또는 사용자로 출력을 제공하도록 구성될 수 있다. 다른 종류의 장치들이 사용자와의 상호 작용을 제공하는데 사용될 수 있으며, 예를 들어, 사용자로 제공되는 피드백은 감각 피드백(sensory feedback)의 형태(예를 들어, 시각 피드백, 청각 피드백, 또는 촉각 피드백)형태일 수 있고, 사용자로부터의 입력은 소리, 음성, 또는 촉각 입력의 형태로 수신될 수 있다.

[0092] 다양한 실시 예들에서, 정보 처리 시스템(400)은 하나 또는 그 이상의 다른 장치들 또는 하드웨어 구성들(460)(예를 들어, 디스플레이 또는 모니터, 키보드, 마우스, 카메라, 지문 인식기, 비디오 프로세서)를 포함할 수 있다. 상술된 내용은 단순히 일부 예시적인 실시 예이며, 본 발명이 이에 한정되지 않음은 이해될 것이다.

[0093] 본 발명에 따른 정보 처리 시스템(400)은 하나 또는 그 이상의 시스템 버스들(405)을 더 포함할 수 있다. 이러한 실시 예에서, 시스템 버스(405)는 프로세서(410), 휘발성 메모리(420), 불휘발성 메모리(430), 네트워크 인터페이스(440), 사용자 인터페이스 유닛(450), 및 하나 또는 그 이상의 하드웨어 구성들(460)을 통신적으로 연결하도록 구성될 수 있다. 프로세서(410)에 의해 처리된 데이터 또는 불휘발성 메모리(430)의 외부로부터 입력된 데이터는 불휘발성 메모리(430) 또는 휘발성 메모리(420) 중 어느 하나에 저장될 수 있다.

[0094] 다양한 실시 예들에서, 정보 처리 시스템(400)은 하나 또는 그 이상의 소프트웨어 구성들(470)을 포함하거나 또는 실행할 수 있다. 일부 실시 예들에서, 소프트웨어 구성들(470)은 운영 체제(OS; operating system) 및/또는 애플리케이션을 포함할 수 있다. 일부 실시 예들에서, OS는 하나 또는 그 이상의 서비스들을 애플리케이션으로 제공하고, 애플리케이션 및 정보 처리 시스템(400)의 다양한 하드웨어 구성들(예를 들어, 프로세서(410), 네트워크

크 인터페이스(440)) 사이의 중재자(intermediary)로서 관리하거나 또는 동작할 수 있다. 이러한 실시 예에서, 정보 처리 시스템(400)은 지역적으로(예를 들어, 불휘발성 메모리(430) 내에) 설치될 수 있고 프로세서(410)에 의해 직접적으 실행되고 OS와 직접적으로 상호 작용하도록 구성된 하나 또는 그 이상의 네이티브 애플리케이션들을 포함할 수 있다. 이러한 실시 예들에서, 네이티브 애플리케이션들은 미리-컴파일된 머신 실행 가능한 코드(pre-compiled machine executable code)를 포함할 수 있다. 일부 실시 예들에서, 네이티브 애플리케이션들은 소스 또는 객체 코드를 프로세서(410)에 의해 실행되는 실행 가능한 코드로 변환하도록 구성된 VM(virtual execution machine)(예를 들어, the Java Virtual Machine, the Microsoft Common Language Runtime) 또는 스크립트 해석기(script interpreter)(예를 들어, csh(C shell), AppleScript, AutoHotkey) 또는 포함할 수 있다.

[0095] 상술된 반도체 장치들은 다양한 패키징 기법들을 사용하여 캡슐화될 수 있다. 예를 들어, 본 발명의 기술적 사상에 따라 구성된 반도체 장치들은 POP(package on package) 기법, BGA(ball grid array) 기법, CSP(chip scale package) 기법, PLCC(plastic leaded chip carrier) 기법, PDIP(plastic dual in-line package) 기법, 다이 인 와플 팩 기법(die in waffle pack technique), 다이 인 웨이퍼 형성 기법(die in wafer form technique), COB(chip on board) 기법, CERDIP(ceramic dual in-line package) 기법, PMQFP(plastic metric quad flat package) 기법, PQFP(plastic quad flat package) 기법, SOIC(small outline package) 기법, SSOP(shrink small outline package) 기법, TSOP(thin small outline package) 기법, TQFP(thin quad flat package) 기법, SIP(system in package) 기법, MCP(multi-chip package) 기법, WFP(wafer-level fabricated package) 기법, WSP(wafer-level processed stack package) 기법, 또는 당업자에 의해 잘 알려진 다른 기법들 중 어느 하나를 사용하여 캡슐화될 수 있다.

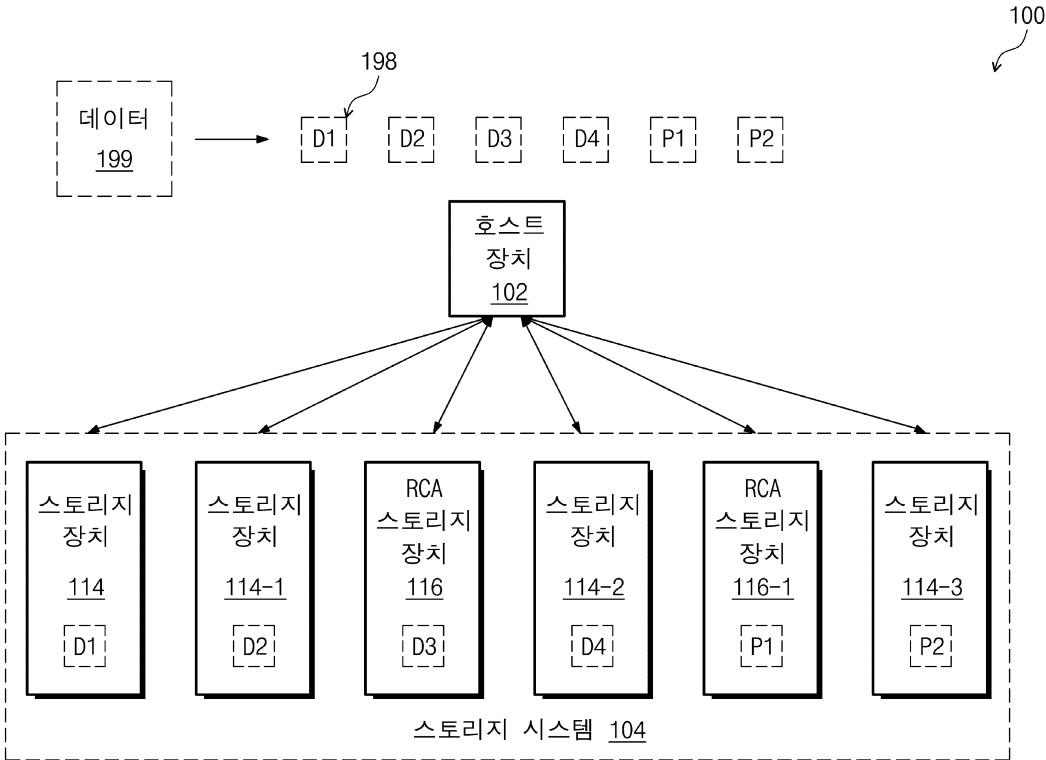
[0096] 방법의 단계들은 컴퓨터 프로그램을 실행하여 입력 데이터에 대해 동작하거나 또는 출력을 생성함으로써 기능들을 수행하는 하나 또는 그 이상의 프로그램 가능상 프로세서들에 의해 수행될 수 있다. 방법의 단계들은 특정 목적 로직 회로, 예를 들어, FPGA(field programmable gate array) 또는 ASIC(application-specific integrated circuit)에 의해 수행될 수 있으며, 장치들은 특정 목적 로직 회로, 예를 들어, FPGA(field programmable gate array) 또는 ASIC(application-specific integrated circuit)으로 구현될 수 있다. .

[0097] 다양한 실시 예들에서, 컴퓨터 판독 가능한 매체는 명령어들을 포함할 수 있고, 명령어들이 실행되는 경우, 장치는 방법의 단계들의 적어도 일부를 수행할 수 있다. 일부 실시 예들에서, 컴퓨터 판독 가능한 매체는 자기 매체(magnetic medium), 광 매체(optical medium), 다른 매체, 또는 그것들의 조합(예를 들어, CD-ROM, 하드 드라이브, 읽기-전용-메모리, 플래시 드라이브)에 포함될 수 있다. 이러한 실시 예에서, 컴퓨터 판독 가능한 매체는 명확하게 그리고 비-일시적으로 형상화된 제조품일 수 있다.

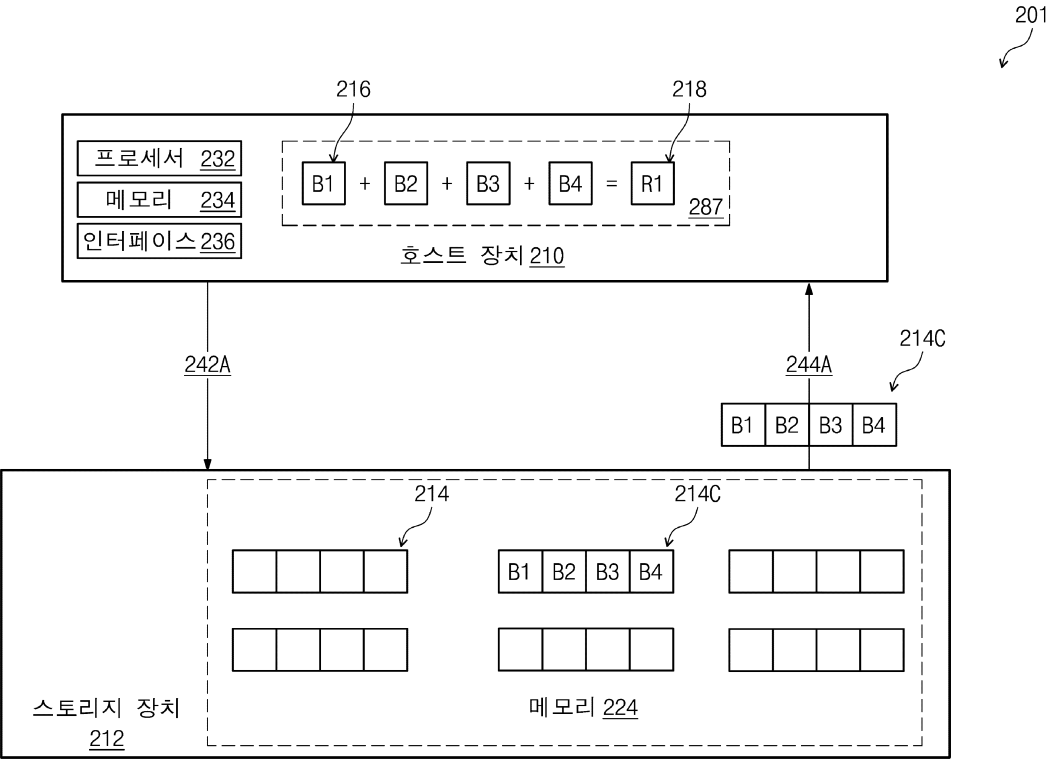
[0098] 본 발명의 기술적 사상이 예시적인 실시 예들을 참조하여 설명되었으나, 당업자는 본 발명의 사상 및 범위로부터 벗어남 없이 다양한 변형들 및 변화들을 행할 수 있을 것이다. 그러므로 상술된 실시 예들은 오직 설명을 위한 것이며, 제한되지 않음이 이해될 것이다. 본 발명의 범위는 이하의 특허청구범위 및 그것의 균등물들의 가장 넓게 허용되는 해석에 의해 결정되어야 하고, 앞선 설명들에 의해 제한되거나 또는 한정되어서는 안된다. 그러므로 첨부된 특허청구범위는 실시 예들의 범위 내에서 이러한 변형들 및 수정들 모두를 포함하는 것으로 의도됨이 이해될 것이다.

도면

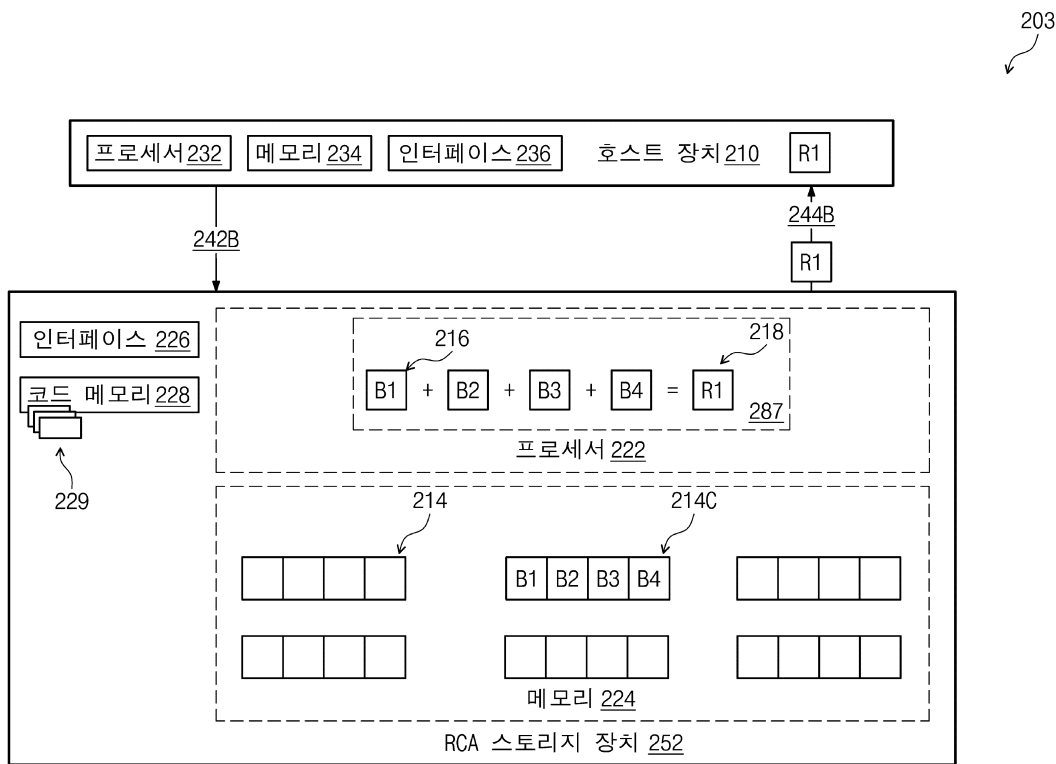
도면1



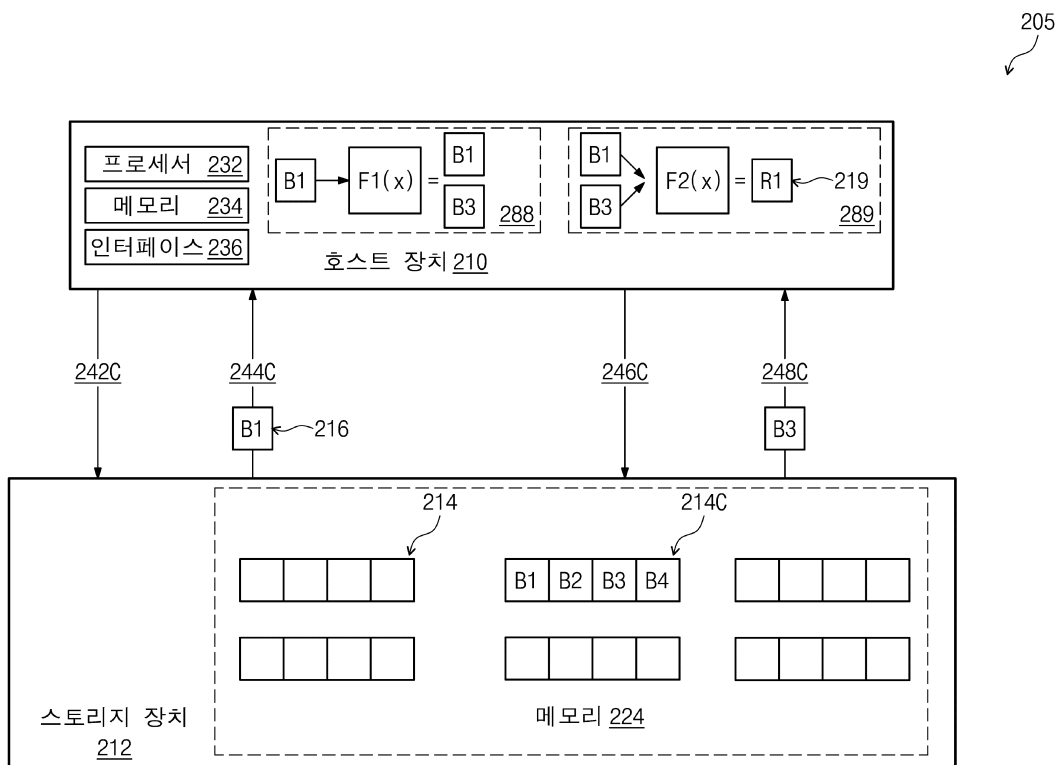
도면2a



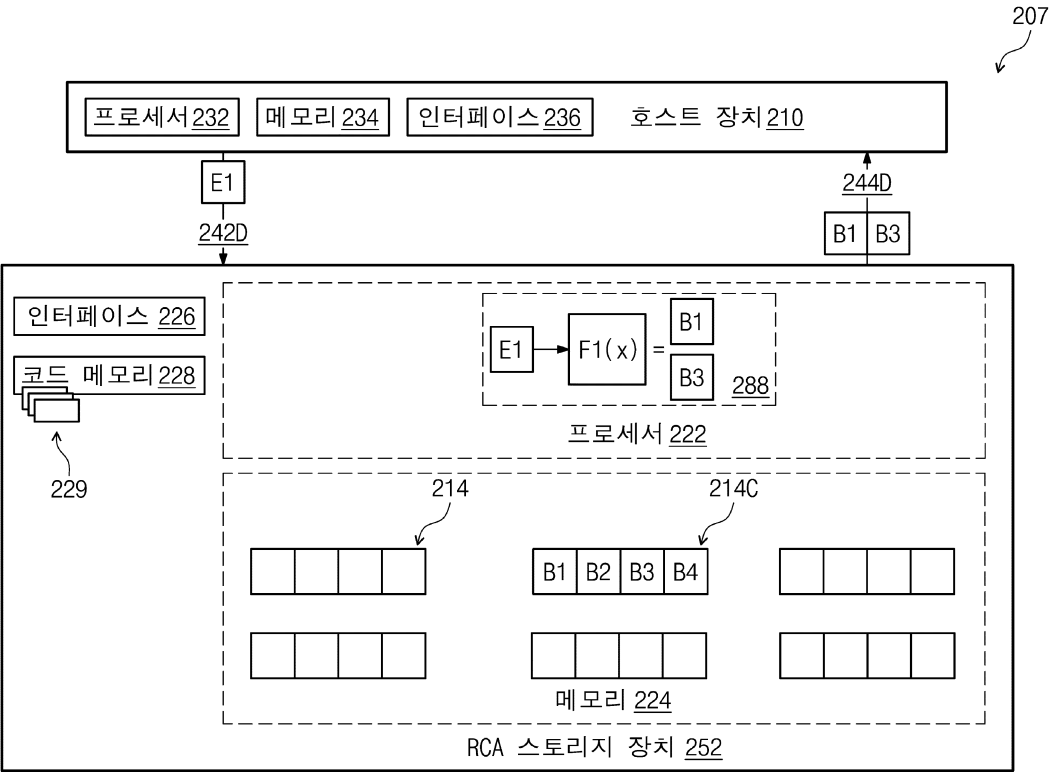
도면2b



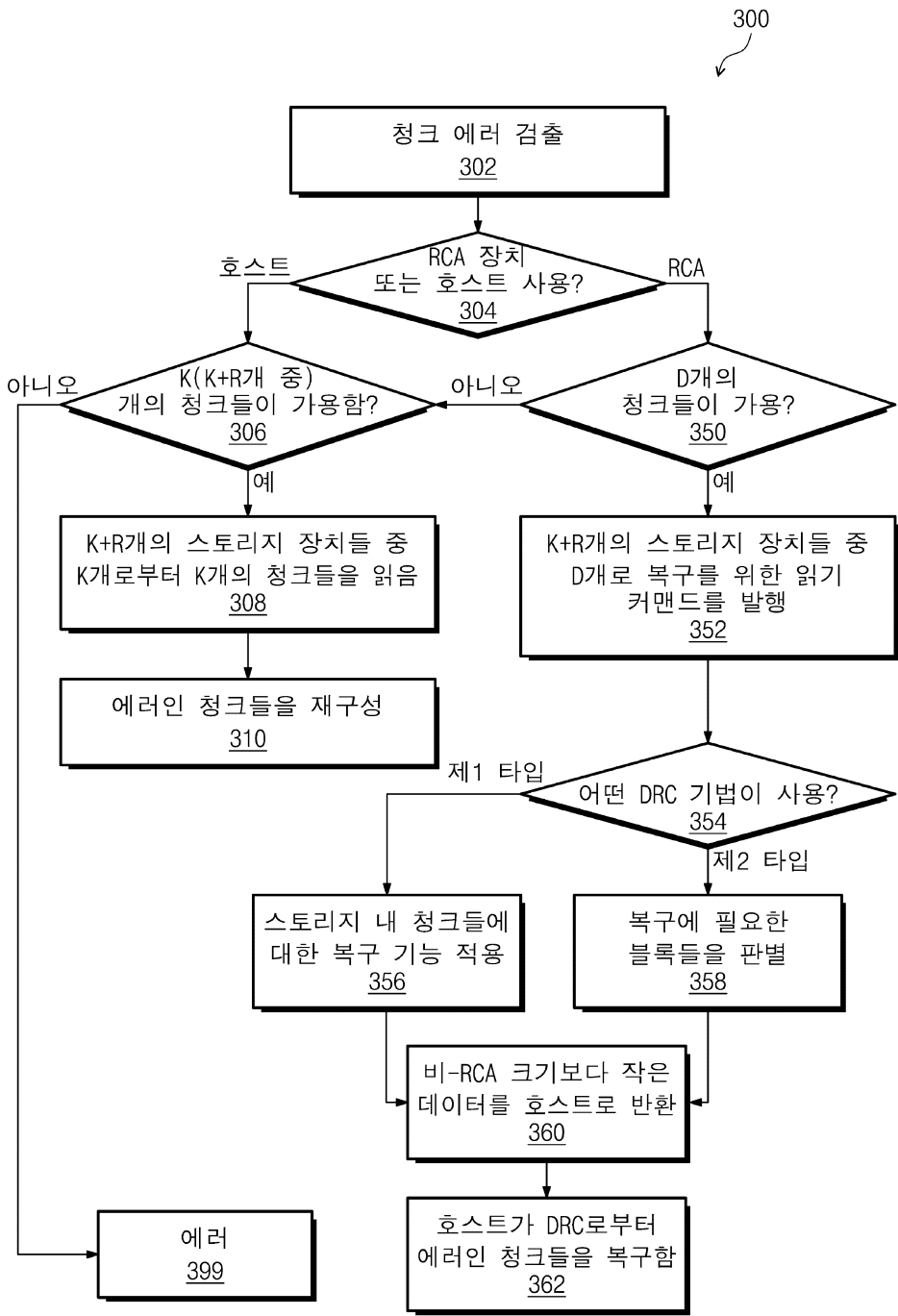
도면2c



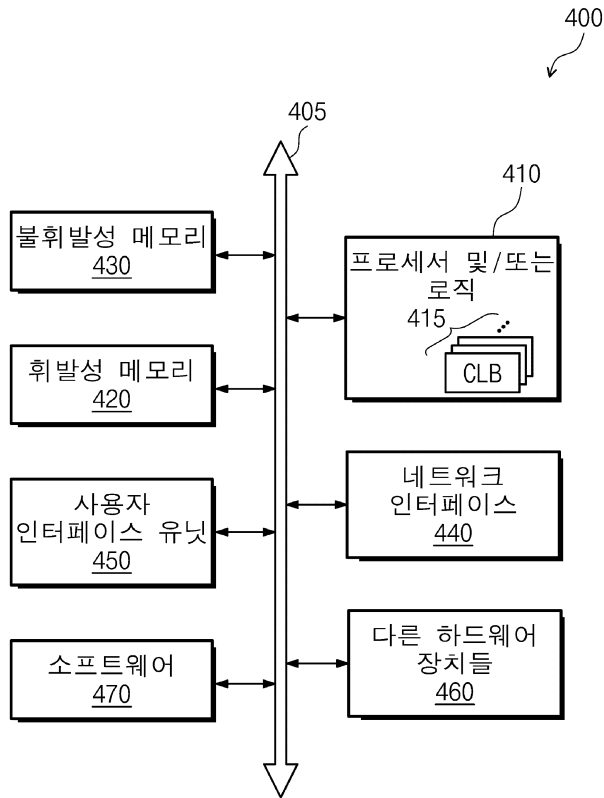
도면2d



도면3



도면4



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 1

【변경전】

장치에 있어서,

데이터 에러 정정을 위한 적어도 하나의 타입의 데이터 재생성 코드를 연산하도록 구성된 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하고,

상기 RCA 스토리지 장치는:

데이터 블록들을 포함하는 체크들로 데이터를 저장하도록 구성된 메모리;

외부 호스트 장치와 연관된 요청을 기반으로, 선택된 다수의 데이터 블록들에 기반된 데이터 재생성 코드를 연산하도록 구성된 프로세서; 및

상기 외부 호스트 장치로 상기 데이터 재생성 코드를 전송하고, 상기 프로세서가 상기 데이터 재생성 코드를 연산하도록 구성하는 커맨드를 상기 외부 호스트 장치로부터 수신하도록 구성된 외부 인터페이스를 포함하는 장치.

【변경후】

장치에 있어서,

데이터 에러 정정을 위한 적어도 하나의 타입의 데이터 재생성 코드를 연산하도록 구성된 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하고,

상기 RCA 스토리지 장치는:

데이터 블록들을 포함하는 청크들로 데이터를 저장하도록 구성된 메모리;

외부 호스트 장치와 연관된 요청을 기반으로, 선택된 다수의 데이터 블록들에 기반된 데이터 재생성 코드를 연산하도록 구성된 프로세서; 및

상기 외부 호스트 장치로 상기 데이터 재생성 코드를 전송하고, 상기 프로세서가 상기 데이터 재생성 코드를 연산하도록 구성하는 커맨드를 상기 외부 호스트 장치로부터 수신하도록 구성된 외부 인터페이스를 포함하는 장치.

【직권보정 2】

【보정항목】 청구범위

【보정세부항목】 청구항 6

【변경전】

제 1 항에 있어서,

상기 프로세서는, 상기 외부 호스트 장치에 의해 요청된 경우, 데이터 재생성 기법들을 통해 상기 데이터 재생성 코드의 다른 버전을 연산하도록 구성되고,

상기 프로세서의 의해 연산되는 상기 다른 버전은 상기 호스트에 의해 지정되는 장치.

【변경후】

제 1 항에 있어서,

상기 프로세서는, 상기 외부 호스트 장치에 의해 요청된 경우, 데이터 재생성 기법들을 통해 상기 데이터 재생성 코드의 다른 버전을 연산하도록 구성되고,

상기 프로세서의 의해 연산되는 상기 다른 버전은 상기 호스트 장치에 의해 지정되는 장치.

【직권보정 3】

【보정항목】 청구범위

【보정세부항목】 청구항 7

【변경전】

시스템에 있어서,

호스트 장치; 및

분산 스토리지 시스템을 포함하고,

상기 호스트는:

상기 분산 스토리지 시스템에 데이터의 청크들로서 데이터를 저장하고;

데이터의 청크가 에러와 연관되는지 검출하고;

상기 에러의 상기 검출에 응답하여, 데이터 재생성 기법을 통해, 상기 데이터의 상기 청크들을 기반으로 상기 에러와 연관된 상기 데이터의 청크를 재구성하도록 구성되고,

상기 분산 스토리지 시스템은 상기 데이터의 각각의 청크를 저장하도록 구성된 스토리지 장치들을 포함하고,

상기 스토리지 장치들은 적어도 하나의 타입의 데이터 재생성 코드를 내부적으로 연산하도록 구성된 적어도 하나의 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하는 시스템.

【변경후】

시스템에 있어서,

호스트 장치; 및

분산 스토리지 시스템을 포함하고,

상기 호스트 장치는:

상기 분산 스토리지 시스템에 데이터의 청크들로서 데이터를 저장하고;

데이터의 청크가 에러와 연관되는지 검출하고;

상기 에러의 상기 검출에 응답하여, 데이터 재생성 기법을 통해, 상기 데이터의 상기 청크들을 기반으로 상기 에러와 연관된 상기 데이터의 청크를 재구성하도록 구성되고,

상기 분산 스토리지 시스템은 상기 데이터의 각각의 청크를 저장하도록 구성된 스토리지 장치들을 포함하고,

상기 스토리지 장치들은 적어도 하나의 타입의 데이터 재생성 코드를 내부적으로 연산하도록 구성된 적어도 하나의 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하는 시스템.

【직권보정 4】

【보정항목】 청구범위

【보정세부항목】 청구항 15

【변경전】

시스템에 있어서,

호스트 장치; 및

스토리지 시스템을 포함하고,

상기 호스트 장치는:

상기 스토리지 시스템에서, 청크들로 데이터를 저장하고,

청크가 에러와 연관되었는지 검출하고,

상기 에러의 상기 검출에 응답하여, 데이터 재생성 기법을 통해, 상기 청크들을 적어도 부분적으로 기반으로, 상기 데이터를 정정하도록 구성되고,

상기 스토리지 시스템은:

상기 데이터의 각각의 청크를 저장하도록 구성된 스토리지 장치들을 포함하고,

상기 스토리지 장치는 데이터 재생성 코드의 적어도 하나의 타입을 내부적으로 연산하도록 구성된 적어도 하나의 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하고,

상기 RCA 스토리지 장치는:

데이터 블록들을 포함하는 청크들로 데이터를 저장하도록 구성된 메모리;

상기 호스트 장치와 연관된 요청을 기반으로, 선택된 다수의 데이터 블록들을 기반으로 데이터 재생성 코드를 연산하도록 구성된 프로세서;

다른 데이터 재생성 코드들을 생성하도록 구성된 하나 또는 그 이상의 명령어들의 세트들을 저장하도록 구성된 코드 메모리; 및

상기 호스트로 상기 데이터 재생성 코드를 전송하도록 구성된 외부 인터페이스를 포함하는 시스템.

【변경후】

시스템에 있어서,

호스트 장치; 및

스토리지 시스템을 포함하고,

상기 호스트 장치는:

상기 스토리지 시스템에서, 청크들로 데이터를 저장하고,

청크가 에러와 연관되었는지 검출하고,

상기 에러의 상기 검출에 응답하여, 데이터 재생성 기법을 통해, 상기 청크들을 적어도 부분적으로 기반하여, 상기 데이터를 정정하도록 구성되고,

상기 스토리지 시스템은:

상기 데이터의 각각의 청크를 저장하도록 구성된 스토리지 장치들을 포함하고,

상기 스토리지 장치는 데이터 재생성 코드의 적어도 하나의 타입을 내부적으로 연산하도록 구성된 적어도 하나의 재생성-코드-인식(RCA; regeneration-code-aware) 스토리지 장치를 포함하고,

상기 RCA 스토리지 장치는:

데이터 블록들을 포함하는 청크들로 데이터를 저장하도록 구성된 메모리;

상기 호스트 장치와 연관된 요청을 기반으로, 선택된 다수의 데이터 블록들을 기반으로 데이터 재생성 코드를 연산하도록 구성된 프로세서;

다른 데이터 재생성 코드들을 생성하도록 구성된 하나 또는 그 이상의 명령어들의 세트들을 저장하도록 구성된 코드 메모리; 및

상기 호스트 장치로 상기 데이터 재생성 코드를 전송하도록 구성된 외부 인터페이스를 포함하는 시스템.

【직권보정 5】

【보정항목】 청구범위

【보정세부항목】 청구항 18

【변경전】

제 17 항에 있어서,

상기 호스트 장치는 상기 스토리지 시스템에서, RAC 스토리지 장치들이 아닌 스토리지 장치들에 의해 저장된 하나 또는 그 이상의 데이터의 청크들에 대한 데이터 재생성 코드를 상기 호스트 장치에 의해, 연산함으로써, 상기 에러를 정정하도록 구성되고,

상기 호스트에 의한 연산은, 상기 스토리지 장치로부터 상기 데이터의 청크의 적어도 부분을 전송하는 것을 포함하고,

상기 RCA 스토리지 장치에 의해 연산되고, 상기 호스트로 전송되는 상기 데이터 재생성 코드의 크기는 상기 스토리지 장치로부터 상기 호스트 장치로 전송되는 상기 데이터의 청크들의 상기 적어도 부분의 크기보다 작은 시스템.

【변경후】

제 17 항에 있어서,

상기 호스트 장치는 상기 스토리지 시스템에서, RAC 스토리지 장치들이 아닌 스토리지 장치들에 의해 저장된 하나 또는 그 이상의 데이터의 청크들에 대해 데이터 재생성 코드를 상기 호스트 장치에 의해, 연산함으로써, 상기 에러를 정정하도록 구성되고,

상기 호스트 장치에 의한 연산은, 상기 스토리지 장치로부터 상기 데이터의 청크의 적어도 부분을 전송하는 것을 포함하고,

상기 RCA 스토리지 장치에 의해 연산되고, 상기 호스트 장치로 전송되는 상기 데이터 재생성 코드의 크기는 상기 스토리지 장치로부터 상기 호스트 장치로 전송되는 상기 데이터의 청크들의 상기 적어도 부분의 크기보다 작은 시스템.