



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2021년07월13일
(11) 등록번호 10-2276912
(24) 등록일자 2021년07월07일

(51) 국제특허분류(Int. Cl.)
G06F 3/06 (2006.01)

(52) CPC특허분류
G06F 3/0643 (2013.01)
G06F 3/0658 (2013.01)

(21) 출원번호 10-2017-0070960

(22) 출원일자 2017년06월07일
심사청구일자 2020년05월12일

(65) 공개번호 10-2018-0133730

(43) 공개일자 2018년12월17일

(56) 선행기술조사문헌
KR1020160150043 A
US20010016925 A1
KR1020110134277 A
JP2001285072 A

(73) 특허권자

삼성전자주식회사

경기도 수원시 영통구 삼성로 129 (매탄동)

연세대학교 산학협력단

서울특별시 서대문구 연세로 50 (신촌동, 연세대학교)

(72) 발명자

이정호

경기도 과천시 별양로 66-11, 343동 803호(별양동, 래미안 슈르)

심호준

경기도 용인시 기흥구 흥덕중앙로105번길 24, 1004동 302호(영덕동, 흥덕마을10단지동원로얄듀크아파트)

(뒷면에 계속)

(74) 대리인

리엔목특허법인

전체 청구항 수 : 총 10 항

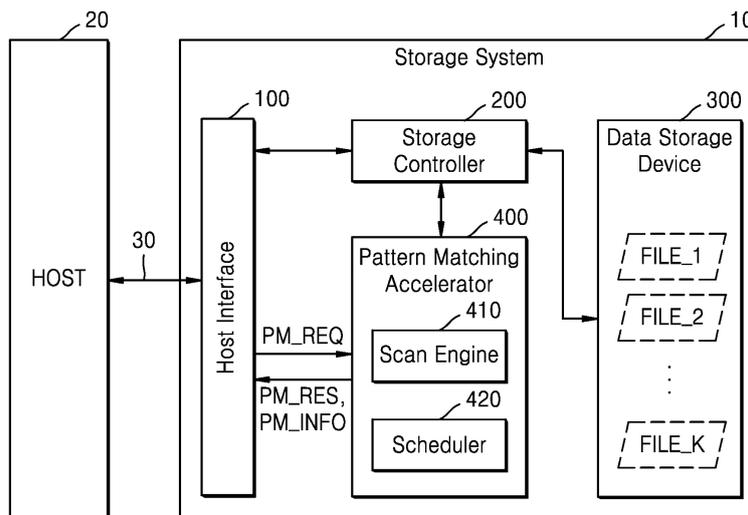
심사관 : 이후락

(54) 발명의 명칭 스토리지 시스템 및 이의 동작 방법

(57) 요약

본 개시의 예시적 실시예에 따른 제거가능한 스토리지 시스템은, 제1 및 제2 파일을 포함하는 복수의 파일들을 저장하는 데이터 스토리지 장치, 호스트로부터 패턴 정보 및 복수의 파일들에 대한 파일 정보를 포함하는 패턴 매칭 요청을 수신하고 복수의 파일들에 대한 패턴 매칭의 결과를 호스트에 송신하는 호스트 인터페이스, 및 패턴 매칭 요청에 응답하여 패턴 매칭을 수행하는 패턴 매칭 가속기를 포함할 수 있다. 패턴 매칭 가속기는 패턴에 기초하여 데이터를 스캔하는 스캔 엔진 및 스캔 엔진이 제1 파일에 대한 스캔을 중지하고 제2 파일에 대한 스캔을 착수하도록 스캔 엔진을 제어하는 스케줄러를 포함할 수 있다.

대표도 - 도1



(52) CPC특허분류

G06F 3/0659 (2013.01)

G06F 3/0679 (2013.01)

(72) 발명자

노원우

서울특별시 강남구 삼성로51길 35, 201동 1202호(대치동, 래미안 대치 팰리스 2단지)

정원섭

서울특별시 마포구 마포대로11길 84, 1동 612호(공덕동, 마포현대아파트)

윤명국

서울특별시 서대문구 신촌로3나길 14, 204호(창천동, 심스빌)

전원

서울특별시 서대문구 연희로14길 23, 301호 (연희동)

명세서

청구범위

청구항 1

제거 가능한 스토리지 시스템으로서,

제1 및 제2 파일을 포함하는 복수의 파일들을 저장하는 데이터 스토리지 장치;

호스트로부터 패턴 정보 및 상기 복수의 파일들에 대한 파일 정보를 포함하는 패턴 매칭 요청을 수신하고, 상기 복수의 파일들에 대한 패턴 매칭의 결과를 상기 호스트에 송신하는 호스트 인터페이스; 및

상기 패턴 매칭 요청에 응답하여 상기 패턴 매칭을 수행하는 패턴 매칭 가속기를 포함하고,

상기 패턴 매칭 가속기는, 패턴에 기초하여 데이터를 스캔하는 스캔 엔진 및 상기 스캔 엔진이 상기 제1 파일에 대한 스캔을 중지하고 상기 제2 파일에 대한 스캔을 착수하도록 상기 스캔 엔진을 제어하는 스케줄러를 포함하는 것을 특징으로 하는 스토리지 시스템.

청구항 2

청구항 1에 있어서,

상기 패턴 매칭 요청은, 상기 패턴 매칭의 완료 시간, 상기 패턴 매칭의 우선순위 및 상기 패턴 매칭의 방식 중 적어도 하나에 대한 요건을 포함하고,

상기 스케줄러는 상기 요건에 기초하여 상기 스캔 엔진을 제어하는 것을 특징으로 하는 스토리지 시스템.

청구항 3

청구항 1에 있어서,

상기 데이터 스토리지 장치는 적어도 하나의 비휘발성 메모리를 포함하고,

상기 스토리지 시스템은, 상기 적어도 하나의 비휘발성 메모리를 관리함으로써 상기 호스트의 상기 데이터 스토리지 장치에 대한 액세스 요청을 처리하는 스토리지 컨트롤러를 더 포함하고,

상기 스케줄러는, 상기 스토리지 컨트롤러로부터 제공되는 상기 데이터 스토리지 장치에 대한 정보에 기초하여 상기 스캔 엔진을 제어하는 것을 특징으로 하는 스토리지 시스템.

청구항 4

청구항 3에 있어서,

상기 스토리지 장치에 대한 정보는 상기 적어도 하나의 비휘발성 메모리로부터 데이터를 독출하는데 걸리는 독출 레이턴시(latency)를 포함하고,

상기 스케줄러는, 상기 제1 파일에 포함된 데이터의 독출 레이턴시가 기준치 이상인 경우 상기 스캔 엔진이 상기 제2 파일에 대한 스캔을 착수하도록, 상기 스캔 엔진을 제어하는 것을 특징으로 하는 스토리지 시스템.

청구항 5

청구항 1에 있어서,

상기 패턴 매칭 가속기는 상기 스캔 엔진의 스캔 결과를 임시적으로 저장하는 리포트 큐를 포함하고,

상기 스케줄러는, 상기 리포트 큐에 상기 제1 파일에 대한 스캔 결과의 저장이 지연되는 경우, 상기 스캔 엔진이 상기 제2 파일에 대한 스캔을 착수하도록 상기 스캔 엔진을 제어하는 것을 특징으로 하는 스토리지 시스템.

청구항 6

청구항 1에 있어서,

상기 패턴 매칭 가속기는 상기 제1 및 제2 파일 각각의 적어도 일부를 저장하는 데이터 버퍼들을 포함하고, 상기 스케줄러는, 상기 스캔 엔진에 의해서 후속하여 스캔될, 상기 데이터 버퍼들에 저장된 상기 제1 파일의 적어도 일부가 기준치 미만 경우, 상기 스캔 엔진이 상기 제2 파일에 대한 스캔을 착수하도록 상기 스캔 엔진을 제어하는 것을 특징으로 하는 스토리지 시스템.

청구항 7

청구항 1에 있어서, 상기 스케줄러는, 상기 스캔 엔진에 의한 상기 제1 파일에 대한 스캔에서 오류가 발생하는 경우, 상기 스캔 엔진이 상기 제2 파일에 대한 스캔을 착수하도록 상기 스캔 엔진을 제어하는 것을 특징으로 하는 스토리지 시스템.

청구항 8

청구항 1에 있어서, 상기 패턴 정보는 복수의 패턴들에 대한 정보를 포함하고, 상기 스케줄러는, 상기 복수의 패턴들 중 적어도 일부를 상기 패턴으로서 상기 스캔 엔진에 동적으로 배정하는 것을 특징으로 하는 스토리지 시스템.

청구항 9

청구항 1에 있어서, 상기 패턴 매칭 가속기는 상기 스캔 엔진의 상태 정보를 저장하는 복수의 컨텍스트 레지스터들을 포함하고, 상기 스케줄러는, 상기 제1 파일에 대한 스캔 중지시 상기 스캔 엔진의 상태 정보를 제1 컨텍스트 레지스터에 저장하고, 상기 제2 파일에 대한 스캔 착수시 제2 컨텍스트 레지스터에 저장된 상태 정보를 상기 스캔 엔진에 로딩하는 것을 특징으로 하는 스토리지 시스템.

청구항 10

제거 가능한 스토리지 시스템으로서, 복수의 파일들을 저장하는 데이터 스토리지 장치; 호스트로부터 패턴 정보 및 상기 복수의 파일들에 대한 파일 정보를 포함하는 패턴 매칭 요청을 수신하고, 상기 복수의 파일들에 대한 패턴 매칭의 결과를 상기 호스트에 송신하는 호스트 인터페이스; 및 상기 패턴 매칭 요청에 응답하여 상기 패턴 매칭을 수행하는 패턴 매칭 가속기를 포함하고, 상기 패턴 매칭 가속기는, 패턴에 기초하여 데이터를 스캔하는 복수의 스캔 엔진들 및 상기 복수의 파일들 중 적어도 일부를 상기 데이터로서 상기 복수의 스캔 엔진들 각각에 동적으로 배정하는 스케줄러를 포함하는 스토리지 시스템.

발명의 설명

기술 분야

[0001] 본 개시의 기술적 사상은 스토리지 시스템에 관한 것으로서, 자세하게는 패턴 매칭을 수행할 수 있는 스토리지 시스템 및 이의 동작 방법에 관한 것이다.

배경 기술

[0002] 스토리지 시스템은 호스트의 요청에 응답하여 호스트로부터 수신된 데이터를 저장하거나 저장된 데이터를 호스트에 제공할 수 있다. 호스트는 스토리지 시스템과 통신함으로써 데이터를 저장하거나 독출할 수 있고, 데이터를 처리함으로써 원하는 동작을 수행할 수 있다. 스토리지 시스템에 저장된 데이터를 이용하는 호스트의 동작들 중 적어도 일부는, 스토리지 시스템에 저장된 데이터에서 특정한 패턴과 매칭되는 데이터를 검색하는 동작을 포함할 수 있다. 스토리지 시스템에 저장된 데이터의 양이 방대할 경우, 이러한 패턴 매칭의 수행은 호스트에 중

대한 부담으로 작용할 수 있고, 결과적으로 호스트 성능의 저하를 유발할 수 있다.

발명의 내용

해결하려는 과제

[0003] 본 개시의 기술적 사상은 스토리지 시스템에 관한 것으로서, 자체적으로 패턴 매칭을 수행할 수 있는 스토리지 시스템 및 이의 동작 방법에 관한 것이다.

과제의 해결 수단

[0004] 상기와 같은 목적을 달성하기 위하여, 본 개시의 기술적 사상의 일측면에 따른 제거가능한 스토리지 시스템은, 제1 및 제2 파일을 포함하는 복수의 파일들을 저장하는 데이터 스토리지 장치, 호스트로부터 패턴 정보 및 복수의 파일들에 대한 파일 정보를 포함하는 패턴 매칭 요청을 수신하고 복수의 파일들에 대한 패턴 매칭 동작의 결과를 호스트에 송신하는 호스트 인터페이스, 및 패턴 매칭 요청에 응답하여 패턴 매칭을 수행하는 패턴 매칭 가속기를 포함할 수 있다. 패턴 매칭 가속기는 패턴에 기초하여 데이터를 스캔하는 스캔 엔진 및 스캔 엔진이 제1 파일에 대한 스캔을 중지하고 제2 파일에 대한 스캔을 착속하도록 스캔 엔진을 제어하는 스케줄러를 포함할 수 있다.

[0005] 또한, 본 개시의 기술적 사상의 일측면에 따른 제거가능한 스토리지 시스템은, 복수의 파일들을 저장하는 데이터 스토리지 장치, 호스트로부터 패턴 정보 및 복수의 파일들에 대한 파일 정보를 포함하는 패턴 매칭 요청을 수신하고 복수의 파일들에 대한 패턴 매칭 동작의 결과를 호스트에 송신하는 호스트 인터페이스, 및 패턴 매칭 요청에 응답하여 패턴 매칭을 수행하는 패턴 매칭 가속기를 포함할 수 있다. 패턴 매칭 가속기는 패턴에 기초하여 데이터를 스캔하는 복수의 스캔 엔진들 및 복수의 파일들 중 적어도 일부를 데이터로서 복수의 스캔 엔진들 각각에 동적으로 배정하는 스케줄러를 포함할 수 있다.

[0006] 또한, 본 개시의 기술적 사상의 일측면에 따른 복수의 파일들을 저장하는 제거가능한 스토리지 시스템의 동작 방법은, 호스트로부터 패턴 정보 및 복수의 파일들에 대한 파일 정보를 포함하는 패턴 매칭 요청을 수신하는 단계, 패턴 매칭 요청에 응답하여 복수의 파일들에 대한 패턴 매칭을 수행하는 단계, 및 패턴 매칭 동작의 결과를 호스트에 전송하는 단계를 포함할 수 있다. 패턴 매칭을 수행하는 단계는, 복수의 파일들 중 적어도 일부를 복수의 스캔 엔진들에 동적으로 배정하는 단계, 및 복수의 스캔 엔진들 각각에 의해서, 패턴에 기초하여 배정된 파일에 포함된 데이터를 스캔하는 단계를 포함할 수 있다.

발명의 효과

[0007] 본 개시의 예시적 실시예에 따라, 스토리지 시스템은 호스트에 대한 패턴 매칭 동작의 부하를 감소시킴으로써 패턴 매칭 동작의 효율성을 증대시킬 수 있다.

[0008] 또한, 본 개시의 예시적 실시예에 따라, 스토리지 시스템은 병렬적으로 처리 가능한 패턴 매칭 동작의 속성을 이용함으로써 높은 효율성의 패턴 매칭 동작을 제공할 수 있다.

도면의 간단한 설명

[0009] 도 1은 본 개시의 예시적 실시예에 따라 호스트와 통신하는 스토리지 시스템을 나타내는 블록도이다.
 도 2a 내지 도 2c는 본 개시의 예시적 실시예들에 따른 도 1의 패턴 매칭 요청의 예시들을 나타내는 도면들이다.
 도 3은 본 개시의 예시적 실시예에 따른 도 1의 패턴 매칭 결과의 예시를 나타내는 도면이다.
 도 4는 본 개시의 예시적 실시예에 따른 도 1의 패턴 매칭 정보의 예시를 나타내는 도면이다.
 도 5는 본 개시의 예시적 실시예에 따라 도 1의 패턴 매칭 가속기의 예시적인 동작을 시간의 흐름에 따라 나타내는 도면이다.
 도 6은 본 개시의 예시적 실시예에 따라 도 1의 스토리지 시스템의 예시적인 구조를 나타내는 블록도이다.
 도 7은 본 개시의 예시적 실시예에 따라 도 6의 패턴 매칭 가속기의 예시적인 구조를 나타내는 블록도이다.
 도 8a 내지 도 8d는 본 개시의 예시적 실시예들에 따라 도 7의 스케줄러가 스캔 엔진에 새로운 파일을 배정하는

동작의 예시들을 나타내는 순서도들이다.

도 9는 본 개시의 예시적 실시예에 따라 도 6의 패턴 매칭 가속기의 예시적인 구조를 나타내는 블록도이다.

도 10은 본 개시의 예시적 실시예에 따라 도 6의 패턴 매칭 가속기의 예시적인 구조를 나타내는 블록도이다.

도 11은 본 개시의 예시적 실시예에 따라 도 10의 복수의 스캔 엔진들의 예시적인 동작들을 시간의 흐름에 따라 나타내는 도면이다.

도 12는 본 개시의 예시적 실시예에 따른 스토리지 시스템의 동작의 예시를 나타내는 순서도이다.

도 13은 본 개시의 예시적 실시예에 따른 스토리지 시스템의 동작의 예시를 나타내는 순서도이다.

도 14는 본 개시의 예시적 실시예에 따른 스토리지 시스템을 포함하는 정규 표현식(regular expression) 매칭 시스템을 나타내는 블록도이다.

발명을 실시하기 위한 구체적인 내용

- [0010] 도 1은 본 개시의 예시적 실시예에 따라 호스트(20)와 통신하는 스토리지 시스템(10)을 나타내는 블록도이다. 후술되는 바와 같이, 스토리지 시스템(10)은 호스트(20)의 요청(예컨대, 패턴 매칭 요청)에 응답하여 스토리지 시스템(10)에 저장된 데이터에 대한 패턴 매칭을 자체적으로 수행할 수 있다.
- [0011] 스토리지 시스템(10) 및 호스트(20)는 데이터 처리 시스템을 구성할 수 있다. 예를 들면, 스토리지 시스템(10) 및 호스트(20)는, 데스크탑 컴퓨터, 워크스테이션, 데이터 센터, 인터넷 데이터 센터, 기업형 데이터 저장 시스템, SAN(storage area network), NAS(network attached storage) 등과 같은 고정형(stationary) 데이터 처리 시스템을 구성할 수도 있고, 랩탑 컴퓨터, 휴대폰, 스마트폰 등과 같은 휴대용(mobile) 데이터 처리 시스템을 구성할 수 있다.
- [0012] 스토리지 시스템(10) 및 호스트(20)가 통신하는 인터페이스(30)는 전기적 신호 및/또는 광신호를 사용할 수 있고, 비제한적인 예시로서, SATA(serial advanced technology attachment) 인터페이스, SATAe(SATA express) 인터페이스, SAS(serial attached small computer system interface; serial attached SCSI), PCIe(peripheral component interconnect express) 인터페이스, NVMe(non-volatile memory Express) 인터페이스, AHCI(advanced host controller interface) 또는 이들의 조합으로 구현될 수 있다.
- [0013] 스토리지 시스템(10)은 호스트(20)와 제거가능하게(removable) 결합함으로써 호스트(20)와 통신할 수 있다. 스토리지 시스템(10)은, 비제한적인 예시로서 SSD(solid-state drive or solid-state disk), 임베디드 SSD(embedded SSD; eSSD), 유니버설 플래시 스토리지(universal flash storage; UFS), 멀티미디어 카드(multimedia card; MMC), 임베디드 멀티미디어 카드(embedded multimedia card; eMMC) 등과 같이 반도체 메모리 장치를 포함할 수도 있고, HDD(hard disk drive), 자기테이프 등과 같이 반도체 메모리 장치와 상이한 저장 매체를 포함할 수도 있다. 도 1에 도시된 바와 같이, 스토리지 시스템은, 호스트 인터페이스(100), 스토리지 컨트롤러(200), 데이터 스토리지 장치(300) 및 패턴 매칭 가속기(400)를 포함할 수 있다.
- [0014] 호스트 인터페이스(100)는 인터페이스(30)를 통해서 호스트(20)로부터 요청을 수신할 수 있다. 예를 들면, 호스트 인터페이스(100)는 인터페이스(30)를 통해서 호스트(20)로부터 기입 요청 및 데이터를 수신할 수 있고, 수신된 기입 요청을 스토리지 컨트롤러(200)에 전달할 수 있다. 또한, 호스트 인터페이스(100)는 인터페이스(30)를 통해서, 도 1에 도시된 바와 같이 호스트(20)로부터 패턴 매칭 요청(PM_REQ)을 수신할 수 있고, 수신된 패턴 매칭 요청(PM_REQ)을 패턴 매칭 가속기(400)에 전달할 수 있다. 도 2a 내지 도 2c를 참조하여 후술되는 바와 같이, 패턴 매칭 요청(PM_REQ)은 패턴 매칭을 수행하는데 필요한 정보를 포함할 수 있다.
- [0015] 호스트 인터페이스(100)는 인터페이스(30)를 통해서 호스트(20)의 요청에 대한 응답을 호스트(20)에 제공할 수 있다. 예를 들면, 호스트 인터페이스(100)는 스토리지 컨트롤러(200)가 호스트(20)의 독출 요청에 응답하여 제공하는 데이터를 호스트(20)에 송신할 수도 있다. 또한, 호스트 인터페이스(100)는 인터페이스(30)를 통해서, 도 1에 도시된 바와 같이 패턴 매칭 가속기(400)로부터 패턴 매칭 결과(PM_RES) 및 패턴 매칭 정보(PM_INFO)를 수신할 수 있고, 수신된 패턴 매칭 결과(PM_RES) 및 패턴 매칭 정보(PM_INFO)를 호스트(20)에 송신할 수 있다. 도 3 및 도 4를 참조하여 후술되는 바와 같이, 패턴 매칭 결과(PM_RES)는 매칭된 패턴을 가지는 데이터에 대한 정보를 포함할 수 있고, 패턴 매칭 정보(PM_INFO)는 패턴 매칭 동작에 대한 정보를 포함할 수 있다.
- [0016] 스토리지 컨트롤러(200)는 데이터 스토리지 장치(300)를 관리함으로써 호스트(20)의 데이터 스토리지 장치(300)에 대한 액세스 요청을 처리할 수 있다. 예를 들면, 스토리지 컨트롤러(200)는, 호스트 인터페이스(100)를 통

해서 수신된 호스트(20)의 기입 요청에 응답하여 기입 요청에 수반된 데이터를 데이터 스토리지 장치(300)에 기입할 수도 있고, 호스트 인터페이스(100)를 통해서 수신된 호스트(20)의 독출 요청에 응답하여 데이터 스토리지 장치(300)에 저장된 데이터를 호스트 인터페이스(100)를 통해서 호스트(20)에 제공할 수도 있다. 일부 실시예들에서, 호스트(20)가 스토리지 시스템(10)에 제공하는 어드레스는 데이터 스토리지 장치(300)의 어드레스와 일치하지 아니할 수 있고, 스토리지 컨트롤러(200)에 의해서 호스트(20)가 제공하는 어드레스는 변환될 수 있다.

[0017] 데이터 스토리지 장치(300)는 스토리지 컨트롤러(200)에 의해서 제어될 수 있고, 복수의 파일들(FILE_1 내지 FILE_K)을 저장할 수 있다. 파일은 호스트(20)에서 관리하는 데이터 단위로서, 스토리지 컨트롤러(200)는 호스트(20)의 독출 요청에 응답하여 데이터 스토리지 장치(300)에 저장된 파일의 적어도 일부를 호스트(20)에 제공할 수 있다. 전송된 바와 같이, 데이터 스토리지 장치(300)는 반도체 메모리 장치를 포함할 수도 있고, 반도체 메모리 장치와 상이한 저장 매체를 포함할 수도 있다.

[0018] 패턴 매칭 가속기(400)는 호스트 인터페이스(100)를 통해서 수신된 호스트(20)의 패턴 매칭 요청(PM_REQ)에 응답하여 데이터 스토리지 장치(300)에 저장된 데이터, 예컨대 복수의 파일들(FILE_1 내지 FILE_K) 중 적어도 일부에 대한 패턴 매칭을 수행할 수 있다. 패턴 매칭 가속기(400)는 패턴 매칭을 수행함으로써 패턴 매칭 결과(PM_RES) 및/또는 패턴 매칭 정보(PM_INFO)를 호스트 인터페이스(100)를 통해서 호스트(20)에 제공할 수 있다. 도 1에 도시된 바와 같이, 패턴 매칭 가속기(400)는 스토리지 컨트롤러(200)와 통신함으로써 데이터 스토리지 장치(300)에 저장된 데이터, 예컨대 복수의 파일들(FILE_1 내지 FILE_K) 중 적어도 일부에 액세스할 수 있다. 패턴 매칭 가속기(400)는 논리 합성을 통해서 설계된 하드웨어 블록을 포함할 수도 있고, 프로세서 등에 의해서 수행되는 소프트웨어 블록을 포함할 수도 있다. 도 1에 도시된 바와 같이, 패턴 매칭 가속기(400)는 스캔 엔진(410) 및 스케줄러(420)를 포함할 수 있다.

[0019] 스캔 엔진(410)은 패턴에 기초하여 데이터를 스캔함으로써 데이터에서 패턴에 매칭되는 위치를 판별할 수 있다. 일부 실시예들에서, 스캔 엔진(410)은 DFA(deterministic finite automaton)를 실행할 수 있는 하드웨어를 포함할 수 있고, 패턴 정보로서 STRT(state transition rule table)를 수신할 수 있다. 스캔 엔진(410)은 STRT에 기초하여 데이터를 바이트 단위로 순차적으로 스캔함으로써 STRT에 의해서 정의된 상태들 사이를 천이할 수 있고, 패턴 매칭을 나타내는 상태에 도달하는 경우 도달된 상태에 대한 정보를 제공할 수 있다. 그러나, 스캔 엔진(410)은 전송된 구조에 제한되지 아니하며, 데이터를 스캔함으로써 패턴 매칭을 수행하는 임의의 구조를 가질 수 있는 점은 이해될 것이다.

[0020] 스케줄러(420)는, 스캔 엔진(410)에 의해서 수행되는 작업을 스케줄링함으로써 스캔 엔진(410)의 효율을 상승시킬 수 있다. 도 2a 내지 도 2c를 참조하여 후술되는 바와 같이, 호스트(20)가 제공하는 패턴 매칭 요청(PM_REQ)은 패턴을 검색하는 대상으로서 복수의 파일들에 대한 정보를 포함할 수 있다. 복수의 파일들 각각에 대한 패턴 매칭 동작들은 서로 독립적으로 수행될 수 있는 한편, 패턴 매칭은 데이터를 순차적으로 스캔하는 작업을 필요로 하기 때문에 각각의 파일에 대한 패턴 매칭 동작은 다양한 원인에 의해서 중단될 수 있다. 예를 들면, 도 8a 내지 도 8d를 참조하여 후술되는 바와 같이, 스캔될 파일의 데이터가 데이터 스토리지 장치(300)로부터 독출되지 아니하거나 지연되어 독출되는 경우, 스캔 결과를 저장하는 동작이 지연되는 경우 또는 데이터 스캔의 동작 중 에러가 발생하는 경우, 스캔 동작이 중단될 수 있고, 하나의 파일에 대한 패턴 매칭 동작이 중단될 수 있다. 스캔 동작을 중단시킨 사유가 해소될 때까지 패턴 매칭 동작은 중단될 수 있고, 이에 따라 패턴 매칭 동작의 효율은 감소할 수 있다.

[0021] 스케줄러(420)는, 스캔 엔진(410)이 복수의 파일들 중 하나에 대한 스캔을 중지하고 복수의 파일들 중 다른 하나에 대한 스캔을 착수하도록 스캔 엔진(410)을 제어할 수 있다. 본 명세서에서, 스케줄러(420)가 스캔 엔진(410)을 특정 파일에 대한 스캔을 착수하도록 제어하는 것은 스케줄러(420)가 상기 특정 파일을 스캔 엔진(410)에 배정(assignment)한 것으로 표현될 수 있다. 예를 들면, 스케줄러(420)는 스캔 엔진(410)을 제어함으로써, 스캔 엔진(410)이 제1 파일(FILE_1)에 대한 스캔 동작, 즉 제1 파일(FILE_1)에 포함되는 데이터에 대한 스캔 동작을 중지할 수 있고, 제2 파일(FILE_2)에 대한 스캔 동작, 즉 제2 파일(FILE_2)에 포함되는 데이터에 대한 스캔 동작을 착수하도록 할 수 있다. 스캔 엔진(410)의 스캔 동작을 중단시키는 이벤트가 발생하는 경우, 스케줄러(420)는 현재 스캔 엔진(410)에 배정된 파일에 대한 스캔 동작을 중지시키고 다른 파일에 대한 스캔 동작을 착수시킴으로써 스캔 엔진(410)의 효율을 증대시킬 수 있고, 결과적으로 패턴 매칭 동작의 효율이 향상될 수 있다.

[0022] 도 2a 내지 도 2c는 본 개시의 예시적 실시예들에 따른 도 1의 패턴 매칭 요청(PM_REQ)의 예시들을 나타내는 도면들이다. 도 1을 참조하여 전송된 바와 같이, 호스트(20)는 인터페이스(30)를 통해서 스토리지 시스템(10)에

패턴 매칭 요청(PM_REQ)을 송신할 수 있고, 패턴 매칭 가속기(400)는 호스트 인터페이스(100)를 통해서 패턴 매칭 요청(PM_REQ)을 수신할 수 있다. 일부 실시예들에서, 패턴 매칭 요청(PM_REQ)은 도 1의 인터페이스(30), 예컨대 SATA, SAS, NVMe 등에서 지원하는 커맨드 세트에서 사용되지 않는(reserved) 영역에 신규 커맨드로서 추가될 수 있다. 이하에서 도 2a 내지 도 2c는 도 1을 참조하여 설명될 것이며, 도 2a 내지 도 2c에 대한 설명 중 상호 중복되는 내용은 생략될 것이다.

[0023] 도 2a를 참조하면, 패턴 매칭 요청(PM_REQ)은 패턴 정보(P_INFO)를 포함할 수 있다. 패턴 정보(P_INFO)는 패턴 매칭의 수행에 필요한 패턴을 정의할 수 있다. 예를 들면, 도 2a에 도시된 바와 같이, 패턴 정보(P_INFO)는 STRT를 포함할 수 있다. 도 1을 참조하여 기술된 바와 같이, 스캔 엔진(410)은 DFA를 실행할 수 있는 하드웨어를 포함할 수 있고, 패턴 정보로서 STRT에 기초하여 데이터를 바이트 단위로 순차적으로 스캔할 수 있다. 다만, 도 2a 및 도 2b에 도시된 패턴 정보(P_INFO)는 예시에 불과하며, 다른 실시예들에서 패턴 정보(P_INFO)는 패턴을 정의하는, STRT와 상이한 유형의 데이터를 포함할 수도 있는 점은 이해될 것이다.

[0024] 패턴 매칭 요청(PM_REQ)은 파일 정보(F_INFO)를 포함할 수 있다. 파일 정보(F_INFO)는 패턴 매칭의 대상이 되는 적어도 하나의 데이터 단위를 정의할 수 있다. 예를 들면, 도 2a에 도시된 바와 같이, 파일 정보(F_INFO)는 적어도 하나의 파일에 대한 정보들을 포함할 수 있고, 하나의 파일에 대한 정보는 파일 식별자(F_ID) 및 적어도 하나의 어드레스 정보(LBA) 및 크기 정보(SIZE)의 쌍({LBA, SIZE})을 포함할 수 있다. 파일 식별자(F_ID)는 어드레스 정보(LBA) 및 크기 정보(SIZE)에 대응하는 데이터가 포함되는 파일을 식별하는데 사용될 수 있다. 예를 들면, 스케줄러(420)는 상이한 파일 식별자들에 각각 포함되는 데이터는 서로 독립적으로 처리될 수 있는 것으로 인식할 수 있고, 그에 따라 스캔 엔진(410)을 제어할 수 있다. 어드레스 정보(LBA) 및 크기 정보(SIZE)의 쌍({LBA, SIZE})에 의해서 데이터 스토리지 장치(300)에 저장된 데이터가 정의될 수 있다. 일부 실시예들에서, 하나의 파일에 대응하는 어드레스 정보(LBA) 및 크기 정보(SIZE)의 쌍({LBA, SIZE})들은, 패턴 매칭이 수행될 (또는 스캔될) 순서대로 정렬될 수 있고, 스케줄러(420)는 어드레스 정보(LBA) 및 크기 정보(SIZE)의 쌍({LBA, SIZE})들의 순서에 따라 스캔 엔진(410)이 스캔 동작을 수행하도록 제어할 수 있다. 도 2a에 도시된 바와 같이 어드레스 정보(LBA) 및 크기 정보(SIZE)의 쌍({LBA, SIZE})에 의해서 데이터를 정의하는 것은 예시에 불과하며, 예를 들면 파일 정보(F_INFO)는 시작(start) 어드레스 정보 및 끝(end) 어드레스 정보의 쌍을 포함할 수도 있다.

[0025] 도 2b를 참조하면, 일부 실시예들에서 패턴 매칭 요청(PM_REQ)은 요건 정보(R_INFO)를 포함할 수 있다. 요건 정보(R_INFO)는 호스트(20)가 스토리지 시스템(10)에서 수행되는 패턴 매칭에 대한 적어도 하나의 요건(requirement)을 포함할 수 있다. 예를 들면, 도 2b에 도시된 바와 같이, 요건 정보(R_INFO)는 스토리지 시스템(10)에서 수행되는 패턴 매칭 동작을 완료하는데 걸리는 시간에 대한 요건인 시간 요건(T_LIM)을 포함할 수도 있고, 패턴 매칭 요청(PM_REQ)의 우선순위를 나타내는 우선순위(PRI)를 포함할 수도 있으며, 패턴 매칭 방식을 결정하는 스킴(scheme) 요건(SCH)을 포함할 수도 있다. 도 2b에 도시된 요건 정보(R_INFO)에 포함되는 요건들은 예시에 불과하며, 요건 정보(R_INFO)는 하나의 요건을 포함할 수도 있고, 도 2b에 도시되지 아니한, 패턴 매칭 동작에 대한 요건이 요건 정보(R_INFO)에 더 포함될 수도 있는 점은 이해될 것이다.

[0026] 도 2c를 참조하면, 일부 실시예들에서 패턴 매칭 요청(PM_REQ)은 도 2a 및 도 2b의 패턴 정보(P_INF) 및/또는 요건 정보(R_INFO)를 포함하지 아니할 수 있다. 즉, 호스트(20)는 도 2a 또는 도 2b의 패턴 매칭 요청(PM_REQ)에 후속하여 파일 정보(F_INFO)를 독립적으로 스토리지 시스템(10)에 전송할 수 있다. 예를 들면, 호스트(20)는 패턴 정보(P_INFO)를 포함하는 패턴 매칭 요청(PM_REQ)을 스토리지 시스템(10)에 전송함으로써, 스토리지 시스템(10)으로 하여금 패턴 매칭 동작을 착수하도록 한 이후, 매회 상이한 데이터를 가리키는 파일 정보(F_INFO)를 추가적으로 적어도 1회 이상 스토리지 시스템(10)에 전송할 수 있다. 스토리지 시스템(10)은 패턴 매칭 요청(PM_REQ)을 수신한 이후, 패턴 매칭 요청(PM_REQ)에 포함된 패턴 정보(P_INFO) 및 후속하여 수신되는 파일 정보(F_INFO)에 기초하여 패턴 매칭 동작을 지속할 수 있다.

[0027] 도 3은 본 개시의 예시적 실시예에 따른 도 1의 패턴 매칭 결과(PM_RES)의 예시를 나타내는 도면이다. 도 1을 참조하여 기술된 바와 같이, 패턴 매칭 가속기(400)는 패턴 매칭을 수행함으로써 매칭된 패턴을 가지는 데이터에 대한 정보를 포함하는 패턴 매칭 결과(PM_RES)를 호스트 인터페이스(100)를 통해서 호스트(20)에 제공할 수 있다. 이하에서, 도 3은 도 1을 참조하여 설명될 것이다.

[0028] 도 3을 참조하면, 패턴 매칭 결과(PM_RES)는 적어도 하나의 데이터 단위에 대한 정보를 포함할 수 있다. 예를 들면, 도 3에 도시된 바와 같이, 패턴 매칭 결과(PM_RES)는 적어도 하나의 데이터 단위에 대한 정보로서 파일 식별자(F_ID) 및 적어도 하나의 상태 식별자(S_ID) 및 오프셋(OFF)의 쌍({S_ID, OFF})을 포함할 수 있다. 파일

식별자(F_ID)는 패턴이 검색된 데이터가 포함되는 파일을 나타낼 수 있다. 상태 식별자(S_ID)는 스캔 엔진(410)이 패턴 매칭 요청(PM_REQ)에 포함된 패턴 정보(P_INFO)(예컨대, STRT)에 따라 상태들 사이를 천이함으로써 데이터를 스캔할 수 있고, 상태 식별자(S_ID)는 매칭된 패턴이 검색된 상태를 나타낼 수 있다. 오프셋(OFF)은 데이터에서 패턴이 검색된 위치를 나타낼 수 있다. 도 3에 도시된 패턴 매칭 결과(PM_RES)는 예시에 불과하며, 패턴 매칭 결과(PM_RES)는 도 3에 도시된 바와 상이하게 패턴이 검색된 데이터의 위치를 나타내는 정보를 포함할 수도 있는 점은 이해될 것이다.

[0029] 도 4는 본 개시의 예시적 실시예에 따른 도 1의 패턴 매칭 정보(PM_INFO)의 예시를 나타내는 도면이다. 도 1을 참조하여 전송된 바와 같이, 패턴 매칭 가속기(400)는 패턴 매칭 동작에 대한 정보를 포함하는 패턴 매칭 정보(PM_INFO)를 호스트 인터페이스(100)를 통해서 호스트(20)에 제공할 수 있다. 이하에서 도 4는 도 1을 참조하여 설명될 것이다.

[0030] 도 4를 참조하면, 패턴 매칭 정보(PM_INFO)는 패턴 매칭을 수행하는 패턴 매칭 가속기(400)(예를 들면, 스케줄러(420))에서 생성될 수 있고, 다양한 패턴 매칭 동작에 대한 정보를 포함할 수 있다. 예를 들면, 도 4에 도시된 바와 같이, 패턴 매칭 정보(PM_INFO)는, 패턴 매칭을 완료하는데 추정되는 시간 정보, 스캔 엔진의 점유율, 스캔 완료된 파일들의 정보를 포함할 수 있다. 패턴 매칭 가속기(400)는, 패턴 매칭 정보(PM_INFO)를 요청하는 호스트(20)의 요청에 응답하여 패턴 매칭 정보(PM_INFO)를 호스트(20)에 제공할 수도 있고, 호스트(20)의 패턴 매칭 요청(PM_REQ)에 포함된 요건 정보(R_INFO)를 충족하기 어렵다고 판단되는 경우 패턴 매칭 정보(PM_INFO)를 호스트(20)에 제공할 수도 있고, 패턴 매칭 요청(PM_REQ)에 따라 패턴 매칭을 수행하는 과정에서 주기적으로 패턴 매칭 정보(PM_INFO)를 호스트(20)에 제공할 수도 있다. 도 4에 도시된 패턴 매칭 정보(PM_INFO)에 포함된 정보는 예시에 불과하며, 패턴 매칭 정보(PM_INFO)는 하나의 정보만을 포함할 수도 있고, 도 4에 도시되지 아니한 정보를 더 포함할 수도 있는 점은 이해될 것이다.

[0031] 도 5는 본 개시의 예시적 실시예에 따라 도 1의 패턴 매칭 가속기(400)의 예시적인 동작을 시간의 흐름에 따라 나타내는 도면이다. 구체적으로, 도 5는, 시간 t11에서 스캔 엔진(410)의 스캔 동작을 중단시키는 이벤트가 발생할 때, 스케줄러(420)의 제어 없이 스캔 엔진(410)이 복수의 파일들에 대한 스캔 동작을 수행하는 예시(CASE_A) 및 스케줄러(420)의 제어 하에서 스캔 엔진(410)이 복수의 파일들에 대한 스캔 동작을 수행하는 예시(CASE_B)를 나타낸다. 도 1을 참조하여 전송된 바와 같이, 스케줄러(420)는 스캔 엔진(410)에 의해서 수행되는 작업을 스케줄링함으로써 스캔 엔진(410)의 활용도를 상승시킬 수 있다. 이하에서 도 5는 도 1을 참조하여 설명될 것이다.

[0032] 도 5의 'CASE_A'를 참조하면, 스캔 엔진(410)은 시간 t11에서 'F_ID=X'인 데이터의 스캔 동작을 이벤트에 기인하여 중단할 수 있다. 도 5에 도시된 바와 같이, 시간 t11에 발생한 이벤트의 원인이 해소되지 아니하는 경우, 스캔 엔진(410)에 의한 스캔 동작이 중단된 상태가 지속될 수 있고, 결과적으로 패턴 매칭 동작의 효율이 저하될 수 있다.

[0033] 도 5의 'CASE_B'를 참조하면, 스케줄러(420)는 스캔 엔진(410)의 스캔 동작의 중단을 유발하는 이벤트를 모니터링할 수 있다. 이에 따라, 스케줄러(420)는 시간 t11에서 발생한 이벤트를 인식할 수 있고, 스캔 엔진(410)이 'F_ID=X'인 데이터의 스캔 동작을 중지하도록 스캔 엔진(410)을 제어할 수 있다. 그 다음에, 도 5에 도시된 바와 같이, 스케줄러(420)는 스캔 엔진(410)의 컨텍스트(context)를 저장할 수 있다. 예를 들면, 도 7을 참조하여 후술되는 바와 같이, 스케줄러(420)는 스캔 엔진(410)의 복수의 컨텍스트들을 저장할 수 있는 레지스터들을 포함할 수 있고, t11에서 중지된 'F_ID=X'인 데이터의 스캔 동작이 추후 재개될 수 있도록 t11에서의 스캔 엔진(410)이 가지는 컨텍스트를 레지스터들 중 하나에 저장할 수 있다.

[0034] 시간 t12에서, 스케줄러(420)는 스캔 엔진(410)의 컨텍스트의 저장을 완료할 수 있고, 스캔 엔진(410)에 다른 컨텍스트를 로딩할 수 있다. 시간 t12에서 스캔 엔진(410)에 로딩되는 컨텍스트는, 스캔 엔진(410)이 'F_ID=Y'인 데이터의 스캔을 착수하는데 사용될 수 있다. 예를 들면, 전송된 바와 같이, 스케줄러(420)는 스캔 엔진(410)의 복수의 컨텍스트들을 저장하는 레지스터들 중 'F_ID=Y'인 데이터의 스캔 동작시의 스캔 엔진(410)의 컨텍스트를 저장하는 레지스터에 액세스할 수 있고, 레지스터로부터 컨텍스트를 독출하여 스캔 엔진(410)에 로딩할 수 있다.

[0035] 시간 t13에서, 스케줄러(420)는 스캔 엔진(410)에 컨텍스트의 로딩을 완료할 수 있고, 스캔 엔진(410)이 'F_ID=Y'인 데이터의 스캔 동작을 착수하도록 스캔 엔진(410)을 제어할 수 있다. 이에 따라, 스캔 엔진(410)은 'F_ID=Y'인 데이터의 스캔 동작을 착수할 수 있는 한편, 스케줄러(420)는 스캔 엔진(410)의 스캔 동작의 중단을 유발할 수 있는 이벤트를 다시 모니터링할 수 있다. 결과적으로, 'CASE_B'에서 스캔 엔진(410)은, 시간 t11에서

'F_ID=X'인 데이터의 스캔 동작을 중단시킨 이벤트의 발생에도 불구하고, 'F_ID=Y'인 데이터의 스캔 동작을 착수할 수 있고, 'CASE_A'와 비교할 때 스캔 엔진(410)의 효율 및 그에 따른 패턴 매칭 동작의 효율이 향상될 수 있다. 이하에서, 본 개시의 예시적 실시예들은 설명의 편의를 위하여, 도 5에 도시된 바와 같이, 이벤트 발생에 기인하여 스캔 엔진(410)이 'F_ID=X'인 데이터의 스캔 동작을 중지하고 'F_ID=Y'인 데이터의 스캔 동작을 착수하는 예시를 인용하여 설명될 것이나, 본 개시의 기술적 사상이 이에 제한되지 아니하는 점은 이해될 것이다.

[0036] 도 6은 본 개시의 예시적 실시예에 따라 도 1의 스토리지 시스템(10)의 예시적인 구조를 나타내는 블록도이다. 도 1을 참조하여 전술된 바와 같이, 도 6의 스토리지 시스템(10')은 호스트(20)와 통신할 수 있고, 패턴 매칭 가속기(400')는 호스트(20)의 패턴 매칭 요청(PM_REQ)에 응답하여 패턴 매칭을 수행함으로써 패턴 매칭 결과(PM_RES) 및/또는 패턴 매칭 정보(PM_INFO)를 호스트(20)에 제공할 수 있다. 이하에서 도 6은 도 1을 참조하여 설명될 것이며, 도 6에 대한 설명 중 도 1에 대한 설명과 중복되는 내용은 생략될 것이다.

[0037] 도 6의 예시에서 스토리지 시스템(10')은 호스트(20)로부터 제공된 데이터를 저장하는 반도체 메모리 장치를 포함할 수 있다. 도 6에 도시된 바와 같이, 스토리지 시스템(10')은 호스트 인터페이스(100'), 스토리지 컨트롤러(200'), 데이터 스토리지 장치(300'), 패턴 매칭 가속기(400'), 버퍼 메모리(500') 및 NVM(non-volatile memory) 인터페이스(600') 및 버스(700')를 포함할 수 있고, 호스트 인터페이스(100'), 스토리지 컨트롤러(200'), 패턴 매칭 가속기(400'), 버퍼 메모리(500') 및 NVM 인터페이스(600')는 버스(700')를 통해서 서로 통신할 수 있다.

[0038] 데이터 스토리지 장치(300')는 복수의 비휘발성 메모리(non-volatile memory; NVM)들을 포함할 수 있다. 비휘발성 메모리(NVM)는 전력 공급이 차단되더라도 저장하고 있는 데이터를 소실하지 아니하는 메모리를 지칭할 수 있고, 데이터 스토리지 장치(300')는 비휘발성 메모리(NVM)으로서, 예컨대, EEPROM (Electrically Erasable Programmable Read-Only Memory), 플래시 메모리(flash memory), PRAM(Phase Change Random Access Memory), RRAM(Resistance Random Access Memory), NFGM (Nano Floating Gate Memory), PoRAM(Polymer Random Access Memory), MRAM (Magnetic Random Access Memory), FRAM(Ferroelectric Random Access Memory) 등을 포함할 수 있다.

[0039] NVM 인터페이스(600') 및 데이터 스토리지 장치(300')는 복수의 채널들(CH_1 내지 CH_L)을 통해서 서로 통신할 수 있다. 예를 들면, 데이터 스토리지 장치(300')에 포함된 비휘발성 메모리(NVM)의 제1 그룹(301)은 제1 채널(CH_1)을 통해서 NVM 인터페이스(600')와 통신할 수 있고, 비휘발성 메모리(NVM)의 제2 그룹(302)은 제2 채널(CH_2)을 통해서 NVM 인터페이스(600')와 통신할 수 있고, 비휘발성 메모리(NVM)의 제3 그룹(303)은 제L 채널(CH_L)을 통해서 NVM 인터페이스(600')와 통신할 수 있다. 비휘발성 메모리(NVM)의 제1 내지 제3 그룹(301 내지 303)은 서로 독립적으로 NVM 인터페이스(600')와 통신할 수 있다. 예를 들면, 상이한 채널들을 통해서 NVM 인터페이스(600')와 통신하는, 즉 상이한 비휘발성 메모리(NVM)의 그룹들에 각각 포함된 비휘발성 메모리(NVM)들은 병렬적으로 데이터를 독출하거나 기입할 수 있다.

[0040] 도 1을 참조하여 전술된 바와 같이, 스토리지 컨트롤러(200')는 데이터 스토리지 장치(300')를 관리함으로써 도 1의 호스트(20)의 데이터 스토리지 장치(300')에 대한 액세스 요청, 예컨대 데이터 독출 요청, 데이터 기입 요청 등을 처리할 수 있다. 예를 들면, 스토리지 컨트롤러(200')는 어드레스 맵핑 테이블을 포함할 수 있고, 어드레스 맵핑 테이블을 참조하여 호스트 인터페이스(100')를 통해서 호스트(20)로부터 수신되는 어드레스, 즉 논리적 어드레스를, 데이터 스토리지 장치(300')의 액세스에 사용되는 어드레스, 즉 물리적 어드레스로 변환할 수 있다. 일부 실시예들에서, 도 6에 도시된 바와 같이, 스토리지 컨트롤러(200')는 호스트 인터페이스(100')를 통해서 패턴 매칭 요청(PM_REQ)을 수신할 수 있고, 패턴 매칭 요청(PM_REQ)에 포함된 파일 정보(F_INFO)로부터 논리적 어드레스를 추출할 수 있으며, 논리적 어드레스를 물리적 어드레스로 변환할 수 있다. 또한, 스토리지 컨트롤러(200')는 호스트(20) 및 데이터 스토리지 장치(300') 사이에서 오류 정정(error correction), 가비지 컬렉션(garbage collection), 재프로그래밍(reprogram) 등의 동작을 수행할 수도 있다. 일부 실시예들에서, 스토리지 컨트롤러(200')는 호스트(20) 및 데이터 스토리지 장치(300') 사이에서 FTL(flash translation layer)을 제공할 수 있다.

[0041] 스토리지 컨트롤러(200')가 데이터 스토리지 장치(300')를 관리함으로써 데이터 스토리지 장치(300'), 즉 비휘발성 메모리(NVM)들에 대한 정보가 생성될 수 있다. 스토리지 컨트롤러(200')는 도 6에 도시된 바와 같이, 버스(700')를 통해서 데이터 스토리지 장치(300')에 대한 정보를, 스토리지 정보(S_INFO)로서 패턴 매칭 가속기(400')에 제공할 수 있고, 패턴 매칭 가속기(400')는 스토리지 정보(S_INFO)에 기초하여 패턴 매칭을 수행할 수 있다. 예를 들면, 패턴 매칭 가속기(400')는 도 1을 참조하여 전술된 바와 같이, 스케줄러 및 스캔 엔진을 포함

할 수 있고, 스케줄러는 스토리지 컨트롤러(200')로부터 제공된 스토리지 정보(S_INFO)에 기초하여 스캔 엔진을 제어할 수 있다. 패턴 매칭 가속기(400')가 스토리지 정보(S_INFO)에 기초하여 패턴 매칭을 수행하는 동작은 도 8a 내지 도 8d를 참조하여 후술될 것이다.

[0042] 버퍼 메모리(500')는 버스(700')를 통해서 수신되는 데이터를 임시적으로 저장할 수 있고, 저장된 데이터를 스토리지 시스템(10')의 다른 구성요소들에 제공할 수 있다. 예를 들면, 버퍼 메모리(500')는 호스트 인터페이스(100')를 통해서 호스트(20)로부터 수신된 데이터를 저장할 수도 있고, 데이터 스토리지 장치(300')로부터 독출된 데이터를 저장할 수도 있다.

[0043] 일부 실시예들에서, 버퍼 메모리(500')는 도 6에 도시된 바와 같이, 패턴 매칭 가속기(400')에 파일 데이터(F_DATA)를 제공할 수 있고, 패턴 매칭 가속기(400')는 파일 데이터(F_DATA)를 스캔함으로써 패턴 매칭을 수행할 수 있다. 파일 데이터(F_DATA)는 스토리지 컨트롤러(200')에 의해서 또는 패턴 매칭 가속기(400')에 의해서 패턴 매칭 요청(PM_REQ)에 포함된 파일 정보(예컨대, 도 2a의 F_INFO)에 기초하여 데이터 스토리지 장치(300')로부터 독출되어 버퍼 메모리(500')에 저장된 데이터일 수 있고, 예컨대 데이터 스토리지 장치(300')에 포함된 비휘발성 메모리(NVM)의 페이지 단위일 수 있다. 패턴 매칭 가속기(400')는 버퍼 메모리(500')에 액세스함으로써 파일 데이터(F_DATA)를 획득할 수 있다. 일부 실시예들에서, 패턴 매칭 가속기(400')는 도 6에 도시된 바와 같이, NVM 인터페이스(600')를 통해서 데이터 스토리지 장치(300')로부터 파일 데이터(F_DATA)를 직접 독출할 수도 있다. 도 7 등을 참조하여 후술되는 바와 같이, 패턴 매칭 가속기(400')에 수신된 파일 데이터(F_DATA)는 패턴 매칭 가속기(400') 내부 저장 공간에 저장될 수 있다.

[0044] 일부 실시예들에서, 패턴 매칭 가속기(400')는 복수의 패턴 매칭 방식들을 지원할 수 있다. 예를 들면, 패턴 매칭 가속기(400')는 파일 데이터(F_DATA)가 수신되는 경향에 기초하여 복수의 패턴 매칭 방식들 중 하나를 선택하여 패턴 매칭을 수행할 수도 있고, 도 2b를 참조하여 전술된 바와 같이 패턴 매칭 요청(PM_REQ)의 요건 정보(R_INFO)에 포함된 스킴 요건에 기초하여 복수의 패턴 매칭 방식들 중 하나를 선택하여 패턴 매칭을 수행할 수도 있다.

[0045] 도 7은 본 개시의 예시적 실시예에 따라 도 6의 패턴 매칭 가속기(400')의 예시적인 구조를 나타내는 블록도이다. 도 6을 참조하여 전술된 바와 같이, 도 7의 패턴 매칭 가속기(400a)는 패턴 매칭 정보(PM_INFO), 스토리지 정보(S_INFO) 및 파일 데이터(F_DATA)를 수신할 수 있고, 패턴 매칭 결과(PM_RES) 및 패턴 매칭 정보(PM_INFO)를 출력할 수 있다. 도 7에 도시된 바와 같이, 패턴 매칭 가속기(400a)는 스캔 엔진(410a), 스케줄러(420a), 리포트 큐(430a), 데이터 버퍼들(440a) 및 천이 테이블(450a)을 포함할 수 있다. 이하에서, 도 7은 도 6을 참조하여 설명될 것이다.

[0046] 데이터 버퍼들(440a)은 적어도 하나의 파일에 대응하는 파일 데이터(F_DATA)를 수집하여 저장할 수 있다. 즉, 데이터 버퍼들(440a)은 파일 정보(F_INFO)에 포함된 파일 식별자들 중 적어도 일부에 대응하는 파일 데이터(F_DATA)를 저장할 수 있다. 이에 따라, 후술되는 바와 같이, 스케줄러(420a)에 의해서 스캔 엔진(410a)에 새로운 파일이 배정될 때, 배정된 파일에 대응하는 파일 데이터(F_DATA)가 데이터 버퍼들(440a)로부터 스케줄러(420a)에 제공될 수 있고, 기존에 배정되어 있던 파일에 대응하는 파일 데이터(F_DATA)는 데이터 버퍼들(440a)에 유지될 수 있다.

[0047] 스케줄러(420a)는 스토리지 정보(S_INFO) 및 파일 정보(F_INFO)를 수신할 수 있고, 데이터 버퍼들(440a)로부터 파일 데이터(F_DATA)를 수신할 수 있다. 스케줄러(420a)는 스토리지 정보(S_INFO) 및 파일 정보(F_INFO)에 기초하여, 스캔 엔진(410a)에 파일을 배정할 수 있다. 예를 들면, 스케줄러(420a)는 스토리지 정보(S_INFO) 및 파일 정보(F_INFO)에 기초하여 스캔 엔진(410a)의 동작(예컨대, 스캔 동작의 중지, 착수 등)을 제어하기 위한 제어 신호(CTR)를 생성할 수 있고, 배정된 파일에 대응하는 파일 데이터(F_DATA)를 데이터 버퍼들(440a)로부터 획득할 수 있으며, 획득된 파일 데이터(F_DATA)로부터 (예컨대, 바이트 단위로) 데이터 스트림(STR)을 생성하여 스캔 엔진(410a)에 제공할 수 있다. 또한, 스케줄러(420a)는 스토리지 정보(S_INFO), 파일 정보(F_INFO), 파일 데이터(F_DATA) 및 스캔 엔진(410a)으로부터 수신된 컨텍스트 신호(CXT) 중 적어도 하나에 기초하여 패턴 매칭 정보(PM_INFO)를 생성하여 출력할 수 있다.

[0048] 스케줄러(420a)는 도 7에서 점선으로 표시된 바와 같이, 패턴 매칭 가속기(400a)의 다른 구성요소들, 즉 스캔 엔진(410a), 리포트 큐(430a) 및 데이터 버퍼들(440a)로부터 신호를 수신할 수 있고, 스캔 엔진(410a), 리포트 큐(430a) 및 데이터 버퍼들(440a)에서 발생한 이벤트를 모니터링할 수 있다. 스케줄러(420a)는 이벤트를 모니터링함으로써 이벤트 발생시 스캔 엔진(410a)에 현재 배정된 파일과 다른 파일을 배정할 수 있다. 스케줄러(420a)가 스캔 엔진(410a)에 새로운 파일을 배정하는 동작의 예시들은 도 8a 내지 도 8d를 참조하여 후술될 것이다.

- [0049] 도 7에 도시된 바와 같이, 스케줄러(420a)는 컨텍스트 레지스터들(421)을 포함할 수 있다. 스케줄러(420a)는 스캔 엔진(410a)으로부터 수신된 컨텍스트 신호(CXT)에 기초하여 스캔 엔진(410a)의 컨텍스트를 컨텍스트 레지스터들(421) 중 하나에 저장할 수도 있고, 컨텍스트 레지스터들(421) 중 하나에 저장된 컨텍스트에 기초하여 컨텍스트 신호(CXT)를 스캔 엔진(410a)에 제공할 수 있다. 예를 들면, 스케줄러(420a)가 스캔 엔진(410a)에 배정된 파일에 대한 스캔 동작을 중지하는 경우, 스케줄러(420a)는 스캔 엔진(410a)로부터 수신된 컨텍스트 신호(CXT)에 기초하여 스캔 엔진(410a)의 컨텍스트를 컨텍스트 레지스터들(421) 중 하나에 저장할 수 있다. 또한, 스케줄러(420a)가 스캔 엔진(410a)에 새로운 파일을 배정하는 경우, 스케줄러(420a)는 컨텍스트 레지스터들(421) 중 배정할 파일에 대응하는 컨텍스트를 저장하는 레지스터에 저장된 컨텍스트로부터 컨텍스트 신호(CXT)를 생성하여 스캔 엔진(410a)에 제공함으로써 스캔 엔진(410a)에 새로운 컨텍스트를 로딩할 수 있다.
- [0050] 스캔 엔진(410a)은 천이 테이블(450a)을 참조하여 스케줄러(420a)로부터 수신되는 데이터 스트림(STR)을 처리함으로써 파일 데이터(F_DATA)를 스캔하는 동작을 수행할 수 있다. 예를 들면, 패턴 정보(P_INFO)는 도 2a 등을 참조하여 전술된 바와 같이 STRT를 포함할 수 있고, 도 7에 도시된 바와 같이 천이 테이블(450a)은 패턴 정보(P_INFO)를 저장할 수 있다. 스캔 엔진(410a)은 천이 테이블(450a)을 참조함으로써 데이터 스트림(STR)에 기초하여 천이할 상태를 결정할 수 있고, 결정된 상태로 천이할 수 있다. 패턴 매칭에 대응하는 상태로 스캔 엔진(410a)이 천이된 경우, 스캔 엔진(410a)은 결과 신호(RES)를 생성하여 리포트 큐(430a)에 제공할 수 있다. 예를 들면, 결과 신호(RES)는 스캔 엔진(410a)의 상태에 대한 정보를 포함할 수 있다.
- [0051] 리포트 큐(430a)는 스캔 엔진(410a)로부터 결과 신호(RES)를 수신할 수 있고, 결과 신호(RES)에 기초하여 스캔 결과를 임시적으로 저장할 수 있다. 또한, 리포트 큐(430a)는 스캔 엔진(410a) 또는 스케줄러(420a)로부터, 파일 정보 및 데이터에서 패턴이 매칭된 위치에 대한 정보를 수신할 수 있고, 결과 신호(RES), 파일 정보 및 데이터에서 패턴이 매칭된 위치에 대한 정보는 총괄적으로 스캔 결과로서 지칭될 수 있다. 리포트 큐(430a)는 스캔 결과를 임시적으로 저장할 수 있고, 스캔 결과를 포함하는 패턴 매칭 결과(PM_RES)를 출력할 수 있다.
- [0052] 도 8a 내지 도 8d는 본 개시의 예시적 실시예들에 따라 도 7의 스케줄러(420a)가 스캔 엔진(410a)에 새로운 파일을 배정하는 동작의 예시들을 나타내는 순서도들이다. 구체적으로, 도 8a 내지 도 8d는 스캔 엔진(410a)에 현재 배정되어 있는 'F_ID=X'인 파일 데이터(F_DATA)에 대한 스캔 동작을 중지하고, 'F_ID=Y'인 파일 데이터(F_DATA)에 대한 스캔 동작을 착수하는 예시들을 나타낸다. 이하에서, 도 8a 내지 도 8d는 도 7을 참조하여 설명될 것이다.
- [0053] 도 8a를 참조하면, 단계 S11에서 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)를 스캔할 수 있다. 단계 S12에서, 스케줄러(420a)는 후속 파일 데이터(F_DATA)의 독출 레이턴시(latency)를 획득할 수 있다. 독출 레이턴시는 데이터 스토리지 장치(예컨대, 도 6의 300')로부터 파일 데이터(F_DATA)를 독출하는데 걸리는 시간을 지칭할 수 있고, 스토리지 컨트롤러(예컨대, 도 6의 200')로부터 제공되는 스토리지 정보(S_INFO)는 독출 레이턴시를 포함할 수 있다. 도 6을 참조하여 전술된 바와 같이, 스토리지 컨트롤러(200')에 의해서 수행되는 동작(예컨대, 에러 정정, 가비지 컬렉션 등) 및/또는 데이터 스토리지 장치(300')에 포함된 비휘발성 메모리(NVM)의 상태(예컨대, 호스트의 기입 요청에 따른 기입 동작 처리 상태 등)에 따라 파일 데이터(F_DATA)가 데이터 스토리지 장치(300')로부터 독출되는데 걸리는 시간은 가변될 수 있다.
- [0054] 단계 S13에서, 스케줄러(420a)는 독출 레이턴시가 기준치 이상인지 판단할 수 있다. 독출 레이턴시가 기준치 미만인 경우 단계 S11에 따라 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)를 스캔하는 동작을 계속할 수 있는 한편, 독출 레이턴시가 기준치 이상인 경우 스케줄러(420a)의 제어에 따라, 단계 S14에서 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)의 스캔을 중지할 수 있고, 그 다음에 단계 S15에서 'F_ID=Y'인 파일 데이터(F_DATA)의 스캔을 착수할 수 있다. 즉, 스캔 엔진(410a)에 의해서 후속하여 스캔될 'F_ID=X'인 파일 데이터(F_DATA)의 적어도 일부를 위한 독출 레이턴시가 큰 경우, 스캔 엔진(410a)의 스캔 동작은 중단될 수 있다. 스케줄러(420a)는 독출 레이턴시에 기초하여 'F_ID=X'인 파일 데이터(F_DATA)의 스캔 동작이 중단될 것을 판단할 수 있고, 판단 결과에 따라 다른 파일, 즉 'F_ID=Y'인 파일을 스캔 엔진(410a)에 배정할 수 있다.
- [0055] 도 8b를 참조하면, 단계 S21에서 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)를 스캔할 수 있다. 단계 S22에서, 스케줄러(420a)는 후속 파일 데이터(F_DATA)가 데이터 버퍼들(440a)에 충분한지 여부를 판단할 수 있다. 예를 들면, 스케줄러(420a)는 데이터 버퍼들(440a)에 저장된 'F_ID=X'인 파일 데이터(F_DATA)의 크기, 현재 스캔 엔진(410a)에 제공되는 데이터 스트림(STR)의 오프셋 및 임의의 기준치에 기초하여 데이터 버퍼들(440a)에 후속 파일 데이터(F_DATA)가 충분한지 여부를 판단할 수 있다. 후속 파일 데이터(F_DATA)가 충분한 경우 단계 S21에 따라 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)를 스캔하는 동작을 계속할 수 있는 한편, 후속

파일 데이터(F_DATA)가 충분하지 아니한 경우 스케줄러(420a)의 제어에 따라, 단계 S23에서 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)의 스캔을 중지할 수 있고, 그 다음에 단계 S24에서 'F_ID=Y'인 파일 데이터(F_DATA)의 스캔을 착수할 수 있다.

[0056] 도 8c를 참조하면, 단계 S31에서 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)를 스캔할 수 있다. 단계 S32에서, 스케줄러(420a)는 스캔 엔진(410a)의 동작에 오류가 발생하였는지 여부를 판단할 수 있다. 비제한적인 예시로서, 스캔 엔진(410a)이 참조하는 천이 테이블(450a)에 저장된 패턴 정보(P_INFO)에서 정의되지 아니한 상태로 천이하거나 패턴 정보(P_INFO)에서 정의되지 아니한 천이 조건이 발생하는 등의 오류가 발생할 수 있다. 오류가 발생하지 아니하는 경우 단계 S31에 따라 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)를 스캔하는 동작을 계속할 수 있는 한편, 오류가 발생한 경우 스케줄러(420a)의 제어에 따라, 단계 S33에서 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)의 스캔을 중지할 수 있고, 그 다음에 단계 S34에서 'F_ID=Y'인 파일 데이터(F_DATA)의 스캔을 착수할 수 있다.

[0057] 도 8d를 참조하면, 단계 S41에서 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)를 스캔할 수 있다. 단계 S42에서, 스케줄러(420a)는 리포트 큐(430a)에 스캔 결과의 저장이 지연되는지 여부를 판단할 수 있다. 예를 들면, 리포트 큐(430a)는 유한한 저장 공간을 가질 수 있고, 스캔 엔진(410a)이 출력하는 결과 신호(RES)에 따라 리포트 큐(430a)에 스캔 결과가 저장되는 속도 및 패턴 매칭 결과(PM_RES)가 출력되는 속도에 따라 리포트 큐(430a)에서 비어있는 저장 공간의 크기가 변경될 수 있다. 리포트 큐(430a)에 비어있는 저장 공간이 없거나 부족한 경우, 리포트 큐(430a)에 스캔 결과의 저장이 지연될 수 있고, 리포트 큐(430a)에 이미 저장되어 있는 스캔 결과들의 소실을 방지하기 위하여 스캔 엔진(410a)의 스캔 동작은 중단될 수 있다. 따라서, 스케줄러(420a)는 리포트 큐(430a)에 스캔 결과의 저장이 지연되는지 여부를 판단함으로써, 리포트 큐(430a)에 스캔 결과의 저장이 지연되지 아니하는 경우 단계 S41에 따라 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)를 스캔하는 동작을 계속할 수 있는 한편, 리포트 큐(430a)에 스캔 결과의 저장이 지연되는 경우 스케줄러(420a)의 제어에 따라, 단계 S43에서 스캔 엔진(410a)은 'F_ID=X'인 파일 데이터(F_DATA)의 스캔을 중지할 수 있고, 그 다음에 단계 S44에서 'F_ID=Y'인 파일 데이터(F_DATA)의 스캔을 착수할 수 있다. 이에 따라, 'F_ID=X'인 파일 데이터(F_DATA)에 대한 스캔 동작은 컨텍스트가 저장됨으로써 추후 재개될 수 있고, 리포트 큐(430a)에 비어 있는 저장 공간이 발생할 때까지 스캔 엔진(410a)은 'F_ID=Y'인 파일 데이터(F_DATA)에 대한 스캔 동작을 수행할 수 있다.

[0058] 이상 도 8a 내지 도 8d를 참조하여 기술된, 스캔 엔진(410a) 및 스케줄러(420a)의 동작들은 예시들에 불과한 점이 유의된다. 즉, 본 개시의 예시적 실시예에 따른 스케줄러(420a)는 스캔 엔진(410a)의 스캔 동작의 중단을 유발하는 다양한 이벤트들을 모니터링할 수 있고 이벤트 발생시 스캔 엔진(410a)에 새로운 파일을 배정할 수 있음은 이해될 것이다.

[0059] 도 9는 본 개시의 예시적 실시예에 따라 도 6의 패턴 매칭 가속기(400')의 예시적인 구조를 나타내는 블록도이다. 도 9에 도시된 바와 같이, 패턴 매칭 가속기(400b)는 스캔 엔진(410b), 스케줄러(420b), 리포트 큐(430b), 데이터 버퍼들(440b) 및 천이 테이블들(450b)을 포함할 수 있다. 도 7의 패턴 매칭 가속기(400a)와 비교할 때, 도 9의 패턴 매칭 가속기(400b)는 복수의 패턴들에 대한 정보를 저장하는 천이 테이블들(450b)을 포함할 수 있다. 이하에서, 도 9에 대한 설명 중 도 7에 대한 설명과 중복되는 내용은 생략될 것이다.

[0060] 천이 테이블들(450b)은 복수의 패턴에 대응하는 패턴 정보(P_INFO)를 수집하여 저장할 수 있다. 일부 실시예들에서, 패턴 매칭 요청(PM_REQ)은 복수의 패턴들에 대응하는 복수의 STRT들을 포함하는 패턴 정보(P_INFO)를 포함할 수 있고, 천이 테이블들(450b)은 복수의 패턴들에 대응하는 패턴 정보(P_INFO)를 저장할 수 있다. 다른 일부 실시예들에서, 도 2a를 참조하여 기술된 바와 같이, 패턴 매칭 요청(PM_REQ)은 하나의 패턴에 대응하는 STRT를 포함하는 패턴 정보(P_INFO)를 포함할 수 있고, 천이 테이블들(450b)은 호스트로부터 제공된 복수의 패턴 매칭 요청(PM_REQ)들로부터 복수의 패턴들에 대응하는 패턴 정보(P_INFO)를 수집하여 저장할 수 있다. 이에 따라, 스캔 엔진(410b)의 컨텍스트는 천이 테이블들(450b) 중 하나를 나타내는 테이블 정보를 포함할 수 있고, 컨텍스트 신호(CXT)를 통해서 스케줄러(420b) 및 스캔 엔진(410b) 사이에서 전달될 수 있다.

[0061] 도 10은 본 개시의 예시적 실시예에 따라 도 6의 패턴 매칭 가속기(400')의 예시적인 구조를 나타내는 블록도이다. 도 10에 도시된 바와 같이, 패턴 매칭 가속기(400c)는 스캔 엔진들(410c), 스케줄러(420c), 리포트 큐들(430c), 데이터 버퍼들(440c) 및 천이 테이블들(450c)을 포함할 수 있다. 도 7 및 도 9의 패턴 매칭 가속기들(400a, 400b)과 비교할 때, 도 10의 패턴 매칭 가속기(400c)는 복수의 스캔 엔진들(410c)을 포함할 수 있고, 복수의 스캔 엔진들(410c)이 각각 참조하는 복수의 천이 테이블들(450c) 및 복수의 스캔 엔진들(410c)에 의해서

병렬적으로 생성되는 스캔 신호들(RESs)을 저장하기 위한 리포트 큐들(430c)을 포함할 수 있다. 또한, 데이터 버퍼들(440c)은 복수의 스캔 엔진들(410c)에 파일 데이터(F_DATA)를 공급하기 위하여 복수의 파일들에 대응하는 파일 데이터(F_DATA)를 수집하여 저장할 수 있다.

- [0062] 도 10을 참조하면, 스케줄러(420c)는 복수의 스캔 엔진들(410c)을 위하여, 제어 신호들(CTRs) 및 데이터 스트림들(STRs)을 생성할 수 있고, 복수의 스캔 엔진들(410c)과 컨택스트 신호들(CXTs)을 주고받을 수 있다. 스케줄러(420c)가 복수의 스캔 엔진들(410c)을 제어하는 동작은 도 11을 참조하여 후술될 것이다.
- [0063] 패턴 매칭 가속기(400c)가 복수의 스캔 엔진들(410c)을 포함하고, 복수의 스캔 엔진들(410c)은 스캔 동작이 중지되는 구간이 최소화되도록 스케줄러(420c)에 의해서 제어됨으로써 패턴 매칭 동작의 효율은 향상될 수 있다. 복수의 스캔 엔진들(410c) 각각은, 서로 독립적으로 자신에 배정된 파일에 대한 스캔 동작을 수행할 수 있고, 스케줄러(420c)에 의해서 새로운 파일을 배정받을 수 있다. 스케줄러(420c)는 복수의 스캔 엔진들(410c) 각각의 스캔 동작의 중단을 유발하는 이벤트들을 모니터링함으로써, 복수의 파일들 중 적어도 일부를 복수의 스캔 엔진들(410c) 각각에 (예컨대, 상호 배타적으로) 동적으로 배정할 수 있다. 이에 따라, 패턴 매칭 동작의 효율은, 복수의 스캔 엔진들(410c)에 의한 병렬적인 스캔 동작들뿐만 아니라 스케줄러(420c)에 의한 복수의 스캔 엔진들(410c) 각각의 상승된 효율에 기인하여, 향상될 수 있다.
- [0064] 도 11은 본 개시의 예시적 실시예에 따라 도 10의 복수의 스캔 엔진들(410c)의 예시적인 동작들을 시간의 흐름에 따라 나타내는 도면이다. 구체적으로, 도 11은 3개의 파일들에 대한 2개의 스캔 엔진들(SE1, SE2)의 스캔 동작들을 나타낸다. 비록 도 11은 2개의 스캔 엔진들(SE1, SE2)의 예시적인 동작들을 도시하나, 본 개시의 예시적 실시예에 따라 3개 이상의 스캔 엔진들의 스캔 동작 역시 도 11과 유사하게 가능한 점은 이해될 것이다. 설명의 편의상 도 11에서 2개의 스캔 엔진들(SE1, SE2)의 컨택스트들이 저장되고 로딩되는 구간은 생략되며, 이하에서 도 11은 도 10을 참조하여 설명될 것이다.
- [0065] 도 11을 참조하면, 제1 스캔 엔진(SE1)은 'F_ID=X'인 파일 데이터(F_DATA)의 스캔 동작을 수행할 수 있고, 제2 스캔 엔진(SE2)은 'F_ID=Y'인 파일 데이터(F_DATA)의 스캔 동작을 수행할 수 있다. 즉, 스케줄러(420c)는 시간 t21 이전에, 제1 스캔 엔진(SE1)에 'F_ID=X'인 파일을 배정하고, 제2 스캔 엔진(SE2)에 'F_ID=Y'인 파일을 배정할 수 있다.
- [0066] 시간 t21에서, 제1 스캔 엔진(SE1)의 스캔 동작의 중단을 유발하는 이벤트가 발생할 수 있다. 스케줄러(420c)는 제1 스캔 엔진(SE1)의 스캔 동작을 중지할 수 있고, 제1 스캔 엔진(SE1)에 'F_ID=Z'인 파일을 배정할 수 있다. 이에 따라, 제1 스캔 엔진(SE1)은 'F_ID=X'인 파일 데이터(F_DATA)에 대한 스캔 동작에 후속하여 'F_ID=Z'인 파일 데이터(F_DATA)에 대한 스캔 동작을 수행할 수 있다.
- [0067] 시간 t22에서, 제2 스캔 엔진(SE2)의 스캔 동작의 중단을 유발하는 이벤트가 발생할 수 있다. 스케줄러(420c)는 제2 스캔 엔진(SE2)의 스캔 동작을 중지할 수 있고, 제2 스캔 엔진(SE2)에 'F_ID=X'인 파일을 배정할 수 있다. 이에 따라, 제2 스캔 엔진(SE2)은 'F_ID=X'인 파일 데이터(F_DATA)에 대한 스캔 동작을 수행할 수 있다. 즉, 시간 t21에서 제1 스캔 엔진(SE1)이 중지한, 'F_ID=X'인 파일 데이터(F_DATA)에 대한 스캔 동작이, 시간 t22에서 제2 스캔 엔진(SE2)에 의해서 재개될 수 있다.
- [0068] 시간 t23에서, 제1 스캔 엔진(SE1)의 스캔 동작의 중단을 유발하는 이벤트가 발생할 수 있다. 스케줄러(420c)는 제1 스캔 엔진(SE1)의 스캔 동작을 중지할 수 있고, 제1 스캔 엔진(SE1)에 'F_ID=Y'인 파일을 배정할 수 있다. 이에 따라, 제1 스캔 엔진(SE1)은 'F_ID=Y'인 파일 데이터(F_DATA)에 대한 스캔 동작을 수행할 수 있다. 즉, 시간 t22에서 제2 스캔 엔진(SE2)이 중지한, 'F_ID=Y'인 파일 데이터(F_DATA)에 대한 스캔 동작이, 시간 t23에서 제1 스캔 엔진(SE1)에 의해서 재개될 수 있다. 이에 따라, 도 11에 도시된 바와 같이, 제1 및 제2 스캔 엔진(SE1, SE2)은 스케줄러(420c)에 의해서 실질적으로 연속적으로 스캔 동작을 수행할 수 있다.
- [0069] 도 12는 본 개시의 예시적 실시예에 따른 스토리지 시스템의 동작의 예시를 나타내는 순서도이다. 예를 들면, 도 12의 동작은 도 1의 스토리지 시스템(10)에 의해서 수행될 수 있다. 이하에서 도 12는 도 1을 참조하여 설명될 것이다.
- [0070] 도 12를 참조하면, 단계 S410a에서 패턴 매칭 요청(PM_REQ)을 호스트로부터 수신하는 동작이 수행될 수 있다. 예를 들면, 스토리지 시스템(10)에 포함된 패턴 매칭 가속기(400)는 호스트 인터페이스(100)를 통해서 호스트(20)로부터 패턴 매칭 요청(PM_REQ)을 수신할 수 있고, 패턴 매칭 요청(PM_REQ)은, 예컨대 도 2a 내지 도 2c를 참조하여 전송된 바와 같이 패턴 정보(P_INFO) 및/또는 파일 정보(F_INFO)를 포함할 수 있다.
- [0071] 단계 S420a에서, 복수의 파일들 중 하나를 스캔 엔진에 배정하는 동작이 수행될 수 있다. 예를 들면, 패턴 매칭

가속기(400)에 포함된 스케줄러(420)는 복수의 파일들 중 하나를 스캔 엔진에 배정할 수 있다. 즉, 스케줄러(420)는 배정된 파일에 대응하는 파일 데이터(F_DATA)로부터 데이터 스트림(STR)을 스캔 엔진(410)에 제공할 수 있고, 스캔 동작을 착수하도록 제어 신호(CTR)를 스캔 엔진(410)에 제공할 수 있다. 그 다음에 단계 S430a에서, 파일 데이터(F_DATA)를 스캔하는 동작이 수행될 수 있다. 예를 들면, 스캔 엔진(410)은 스케줄러(420)로부터 배정된 파일에 대응하는 파일 데이터(F_DATA)를 패턴 정보(P_INFO)에 기초하여 스캔할 수 있다.

[0072] 단계 S440a에서, 스캔 엔진에 파일을 동적으로 배정하는 동작이 수행될 수 있다. 구체적으로, 단계 S441a에서, 이벤트의 발생 여부를 판단하는 동작이 수행될 수 있다. 예를 들면, 스케줄러(420)는 스캔 엔진(410)의 스캔 동작의 중단을 유발하는 이벤트를 모니터링함으로써 이벤트의 발생 여부를 판단할 수 있다. 이벤트가 발생한 경우 단계 S442a가 후속해서 수행될 수 있는 한편, 이벤트가 발생하지 아니한 경우 단계 S450a가 후속해서 수행될 수 있다.

[0073] 단계 S442a에서, 스캔 엔진에 다른 파일을 배정하는 동작이 수행될 수 있다. 예를 들면, 스케줄러(420)는 스캔 엔진(410)에 배정된 파일과 다른 파일을 스캔 엔진에 배정할 수 있다. 스케줄러(420)는 스캔 엔진(410)의 스캔 동작을 중지시키고 스캔 엔진(410)의 컨텍스트를 저장한 후, 다른 파일의 스캔 동작에 대응하는 컨텍스트를 스캔 엔진(410)에 로딩하고 스캔 엔진(410)이 스캔 동작을 착수하도록 제어할 수 있다. 도 12에 도시된 바와 같이, 단계 S442a에 후속하여 단계 S430a가 수행될 수 있다.

[0074] 단계 S450a에서, 패턴 매칭이 발생하였는지 여부를 판단하는 동작이 수행될 수 있다. 예를 들면, 스캔 엔진은(410)은 파일 데이터(F_DATA)를 스캔 패턴 정보(P_INFO)에 기초하여 스캔함으로써 패턴 매칭의 발생 여부를 판단할 수 있다. 패턴 매칭이 발생하지 아니한 경우 단계 S430a가 후속하여 수행될 수 있는 한편, 패턴 매칭이 발생한 경우 단계 S460a가 후속하여 수행될 수 있다.

[0075] 단계 S460a에서, 패턴 매칭 결과(PM_RES)를 호스트에 송신하는 동작이 수행될 수 있다. 예를 들면, 패턴 매칭 가속기(400)는, 도 3을 참조하여 전송된 바와 같이, 파일 식별자(F_ID) 및 파일 데이터(F_DATA)에서 매칭된 패턴이 검색된 위치를 나타내는 정보를 포함하는 패턴 매칭 결과(PM_RES)를 호스트 인터페이스(100)를 통해서 호스트(20)에 송신할 수 있다. 비록 도 12는 패턴 매칭이 발생하는 경우 패턴 매칭 결과(PM_RES)가 호스트에 송신되는 예시를 도시하였으나, 일부 실시예들에서, 패턴 매칭 결과(PM_RES)는, 2이상의 패턴 매칭들이 발생한 경우 호스트에 송신될 수도 있고, 그리고/또는 단계 S470a에 후속하여 패턴 매칭이 완료된 후 호스트에 송신될 수도 있다.

[0076] 단계 S470a에서, 패턴 매칭이 완료되었는지 여부를 판단하는 동작이 수행될 수 있다. 예를 들면, 스케줄러(420)는 패턴 매칭 요청(PM_REQ)에 포함된 적어도 하나의 파일의 파일 데이터(F_DATA)에 대한 스캔 동작이 스캔 엔진(410)에 의해서 완료되었는지 여부를 판단할 수 있다. 패턴 매칭이 완료되지 아니한 경우 단계 S430a가 후속하여 수행될 수 있는 한편, 패턴 매칭이 완료된 경우 패턴 매칭 요청에 응답하여 착수된 스토리지 시스템의 동작은 종료될 수 있다.

[0077] 도 13은 본 개시의 예시적 실시예에 따른 스토리지 시스템의 동작의 예시를 나타내는 순서도이다. 예를 들면, 도 13의 동작은 도 10의 패턴 매칭 가속기(400c)를 포함하는 스토리지 시스템에 의해서 수행될 수 있다. 이하에서 도 13은 도 10을 참조하여 설명될 것이다.

[0078] 도 13을 참조하면, 단계 S410b에서 패턴 매칭 요청(PM_REQ)을 호스트로부터 수신하는 동작이 수행될 수 있다. 예를 들면, 패턴 매칭 가속기(400c)는 호스트 인터페이스(예컨대, 도 1의 100)를 통해서 호스트로부터 패턴 매칭 요청(PM_REQ)을 수신할 수 있고, 패턴 매칭 요청(PM_REQ)은 패턴 정보(P_INFO) 및 파일 정보(F_INFO)를 포함할 수 있다.

[0079] 단계 S420b에서, 복수의 파일들 중 적어도 일부를 복수의 스캔 엔진들 각각에 배정하는 동작이 수행될 수 있다. 예를 들면, 스케줄러(420c)는 복수의 파일들 중 하나를 복수의 스캔 엔진들(410c) 중 하나에 배정할 수 있다. 그 다음에 단계 S430b에서, 파일 데이터(F_DATA)를 병렬적으로 스캔하는 동작이 수행될 수 있다. 예를 들면 복수의 스캔 엔진들(410c)은 서로 독립적으로 스캔 동작을 수행할 수 있다.

[0080] 단계 S440b에서, 복수의 파일들 중 적어도 일부를 복수의 스캔 엔진들 각각에 동적으로 배정하는 동작이 수행될 수 있다. 구체적으로, 단계 S441b에서, 이벤트의 발생 여부를 판단하는 동작이 수행될 수 있다. 예를 들면, 스케줄러(420c)는 복수의 스캔 엔진들(410c)의 스캔 동작들 중 적어도 하나의 중단을 유발하는 이벤트를 모니터링함으로써 이벤트의 발생 여부를 판단할 수 있다. 이벤트가 발생한 경우 단계 S442b가 후속해서 수행될 수 있는 한편, 이벤트가 발생하지 아니한 경우 단계 S450b가 후속해서 수행될 수 있다.

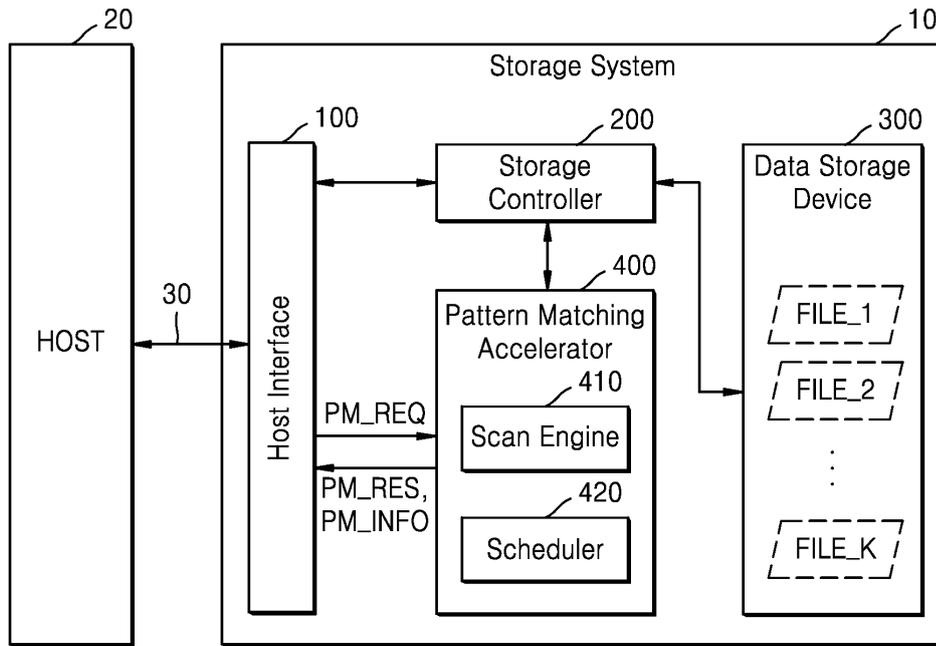
- [0081] 단계 S442b에서, 이벤트와 관련된 스캔 엔진에 다른 파일을 배정하는 동작이 수행될 수 있다. 예를 들면, 스케줄러(420c)는 복수의 스캔 엔진들(410c) 중, 이벤트에 의해서 스캔 동작이 중단되거나 중단될 스캔 엔진의 컨택스트를 저장한 후, 다른 파일의 스캔 동작에 대응하는 컨택스트를 해당 스캔 엔진에 로딩하고 해당 스캔 엔진이 스캔 동작을 착수하도록 제어할 수 있다. 도 13에 도시된 바와 같이, 단계 S442b에 후속하여 단계 S430b가 수행될 수 있다.
- [0082] 단계 S450b에서, 패턴 매칭이 발생하였는지 여부를 판단하는 동작이 수행될 수 있다. 예를 들면, 스캔 엔진은 (410)은 파일 데이터를 파일 데이터(F_DATA)를 스캔 패턴 정보(P_INFO)에 기초하여 스캔함으로써 패턴 매칭의 발생 여부를 판단할 수 있다. 패턴 매칭이 발생하지 아니한 경우 단계 S430b가 후속하여 수행될 수 있는 한편, 패턴 매칭이 발생한 경우 단계 S460b가 후속하여 수행될 수 있다.
- [0083] 단계 460b에서, 패턴 매칭 결과(PM_RES)를 호스트에 송신하는 동작이 수행될 수 있다. 예를 들면, 패턴 매칭 가속기(400c)는, 파일 식별자(F_ID) 및 파일 데이터(F_DATA)에서 매칭된 패턴이 검색된 위치를 나타내는 정보를 포함하는 패턴 매칭 결과(PM_RES)를 호스트 인터페이스(예컨대 도 1의 100)를 통해서 호스트에 제공할 수 있다. 비록 도 13은 패턴 매칭이 발생하는 경우 패턴 매칭 결과(PM_RES)가 호스트에 송신되는 예시를 도시하였으나, 일부 실시예들에서, 패턴 매칭 결과(PM_RES)는, 2이상의 패턴 매칭들이 발생한 경우 호스트에 송신될 수도 있고, 그리고/또는 단계 S470b에 후속하여 패턴 매칭이 완료된 후 호스트에 송신될 수도 있다.
- [0084] 단계 S470b에서, 패턴 매칭이 완료되었는지 여부를 판단하는 동작이 수행될 수 있다. 예를 들면, 스케줄러(420c)는 패턴 매칭 요청(PM_REQ)에 포함된 적어도 하나의 파일의 파일 데이터(F_DATA)에 대한 스캔 동작이 복수의 스캔 엔진들(410c)에 의해서 완료되었는지 여부를 판단할 수 있다. 패턴 매칭이 완료되지 아니한 경우 단계 S430b가 후속하여 수행될 수 있는 한편, 패턴 매칭이 완료된 경우 패턴 매칭 요청에 응답하여 착수된 스토리지 시스템의 동작은 종료될 수 있다.
- [0085] 도 14는 본 개시의 예시적 실시예에 따른 스토리지 시스템을 포함하는 정규 표현식(regular expression) 매칭 시스템(2)을 나타내는 블록도이다. 도 14에 도시된 바와 같이, 정규 표현식 매칭 시스템(2)은 본 개시의 예시적 실시예에 따른 스토리지 시스템을 포함할 수 있다.
- [0086] 사용자 시스템(1)은 정규 표현식(R_EXP) 및 데이터 정보(D_INFO)를 정규 표현식 매칭 시스템(2)에 제공할 수 있고, 정규 표현식 매칭 시스템(2)으로부터 매칭 결과(M_RES)를 수신할 수 있다. 정규 표현식은 특정한 패턴을 정의하는 문자열을 지칭할 수 있고, 정규 표현식에 의해서 특정한 규칙을 가진 문자열의 집합이 정의될 수 있다. 사용자 시스템(1)은, 비제한적인 예시로서 사용자가 사용하는 단말, 최종 사용자에게 검색 서비스를 제공하는 검색 시스템 등과 같이, 데이터 정보(D_INFO)에 의해 정의되는 데이터에서 정규 표현식(R_EXP)에 매칭되는 부분을 구하고자 하는 임의의 엔티티(entity)일 수 있다.
- [0087] 도 14에 도시된 바와 같이, 정규 표현식 매칭 시스템(2)은 복수의 스토리지 시스템들(31 내지 33) 및 호스트(40)를 포함할 수 있고, 복수의 스토리지 시스템들(31 내지 33)은 호스트(40)와 통신할 수 있다. 복수의 스토리지 시스템들(31 내지 33)은 도 14에 도시된 바와 같이 SSD일 수 있고, 전술된 본 개시의 예시적 실시예들에 따라 호스트(40)의 패턴 매칭 요청에 응답하여 패턴 매칭을 수행할 수 있다. 예를 들면, 제1 SSD(31)는 패턴 매칭 가속기를 포함할 수 있고, 패턴 매칭 가속기는 호스트(40)로부터 수신되는 패턴 매칭 요청(PM_REQ)에 응답하여 스캔 엔진에 파일을 동적으로 배정함으로써 향상된 효율성을 가지는 패턴 매칭을 제공할 수 있다. 또한, 패턴 매칭 가속기는 복수의 스캔 엔진들을 포함할 수도 있고, 복수의 파일 들 중 적어도 일부를 복수의 스캔 엔진들 각각에 상호 배타적으로 동적으로 배정할 수도 있다.
- [0088] 호스트(40)는 규칙 테이블 생성기(41) 및 실행 관리자(42)를 포함할 수 있다. 규칙 테이블 생성기(41)는 사용자 시스템(1)으로부터 정규 표현식(R_EXP)을 수신할 수 있고, 정규 표현식(R_EXP)에 기초하여 DFA를 생성하여 실행 관리자(42)에 제공할 수 있다. DFA는 정규 표현식(R_EXP)에 대응하는 패턴을 나타낼 수 있고, 실행 관리자(42)를 통해서 STRT의 형태로 복수의 스토리지 시스템들(31 내지 33) 중 적어도 하나에 전달될 수 있다. 실행 관리자(42)는 데이터 정보(D_INFO)에 포함된 파일 정보에 기초하여, 복수의 스토리지 시스템들(31 내지 33) 중 파일이 저장된 것을 판별할 수 있고, 해당 파일의 정보 및 DFA를 포함하는 패턴 매칭 요청(PM_REQ)을 판별된 스토리지 시스템에 제공할 수 있다. 또한, 실행 관리자(42)는 패턴 매칭 요청(PM_REQ)에 응답하여 복수의 스토리지 시스템들(31 내지 33) 중 적어도 일부가 패턴 매칭을 수행함으로써 제공하는 패턴 매칭 결과(PM_RES)를 수신할 수 있고, 수신된 패턴 매칭 결과(PM_RES)에 기초하여 매칭 결과(M_RES)를 생성하여 사용자 시스템(1)에 제공할 수 있다.

[0089]

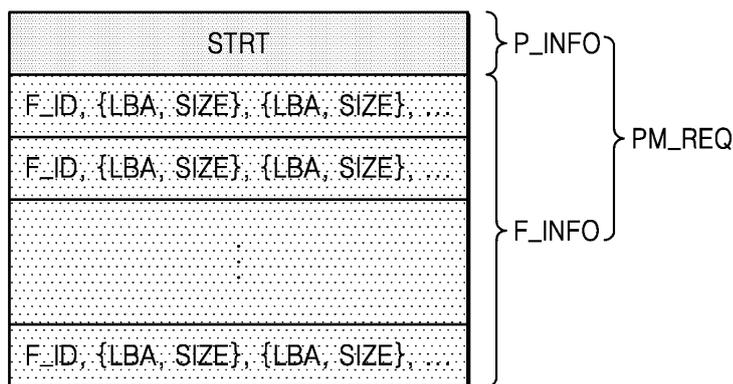
이상에서와 같이 도면과 명세서에서 예시적인 실시예들이 개시되었다. 본 명세서에서 특정한 용어를 사용하여 실시예들을 설명되었으나, 이는 단지 본 개시의 기술적 사상을 설명하기 위한 목적에서 사용된 것이지 의미 한정이나 특허청구범위에 기재된 본 개시의 범위를 제한하기 위하여 사용된 것은 아니다. 그러므로 본 기술분야의 통상의 지식을 가진 자라면 이로부터 다양한 변형 및 균등한 타 실시예가 가능하다는 점을 이해할 것이다. 따라서, 본 개시의 진정한 기술적 보호범위는 첨부된 특허청구범위의 기술적 사상에 의해 정해져야 할 것이다.

도면

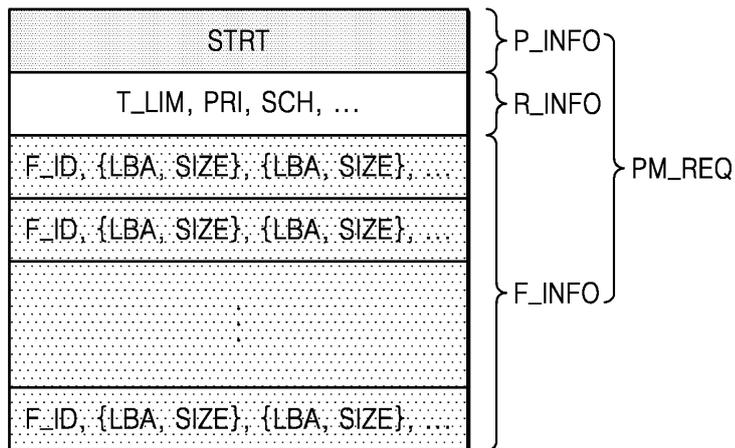
도면1



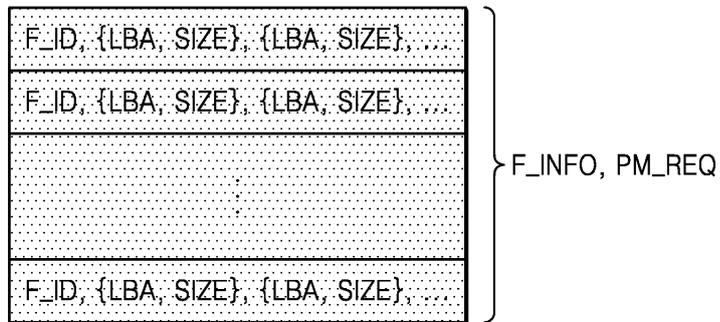
도면2a



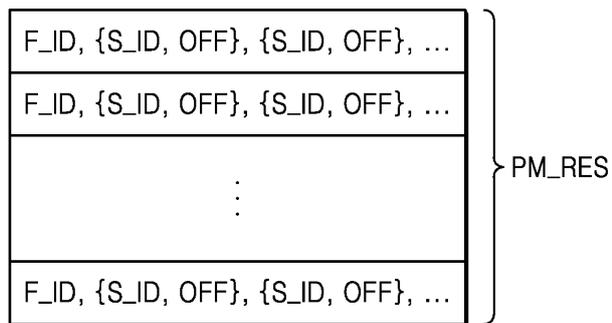
도면2b



도면2c

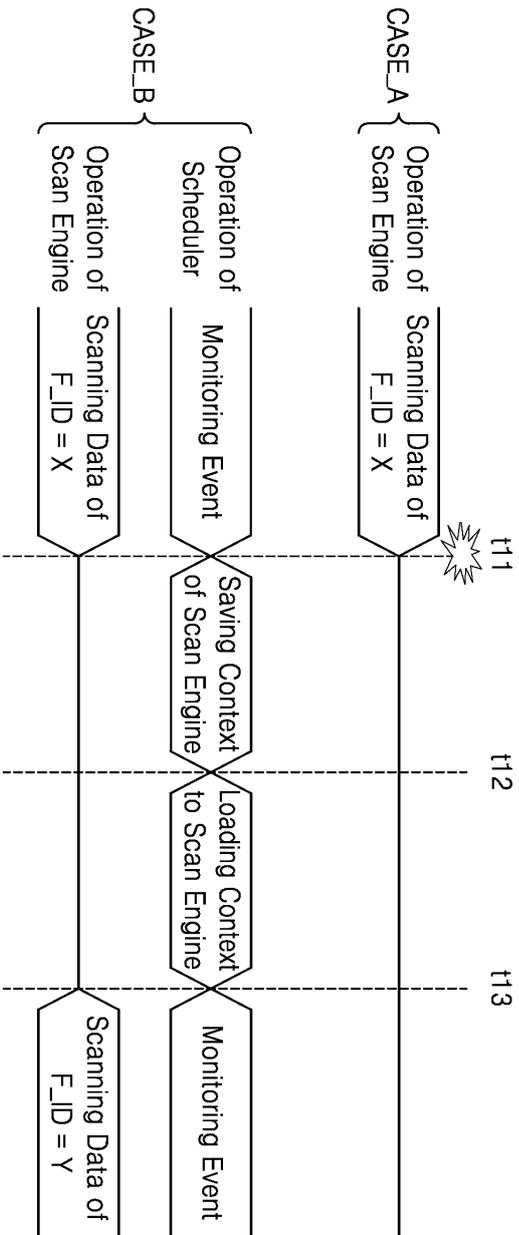


도면3



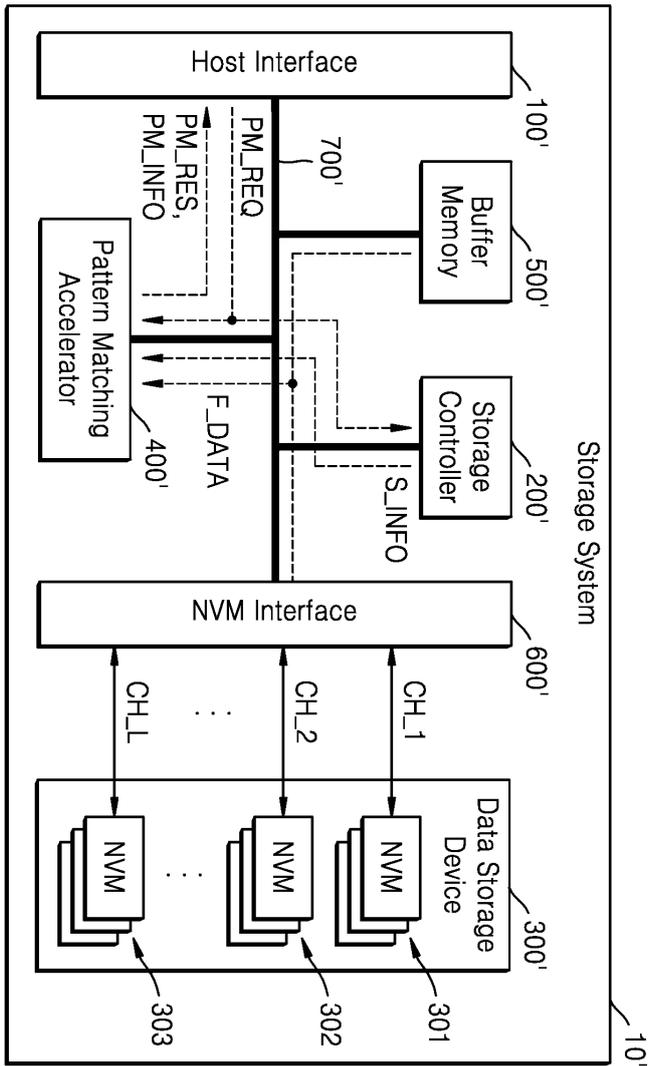
도면4



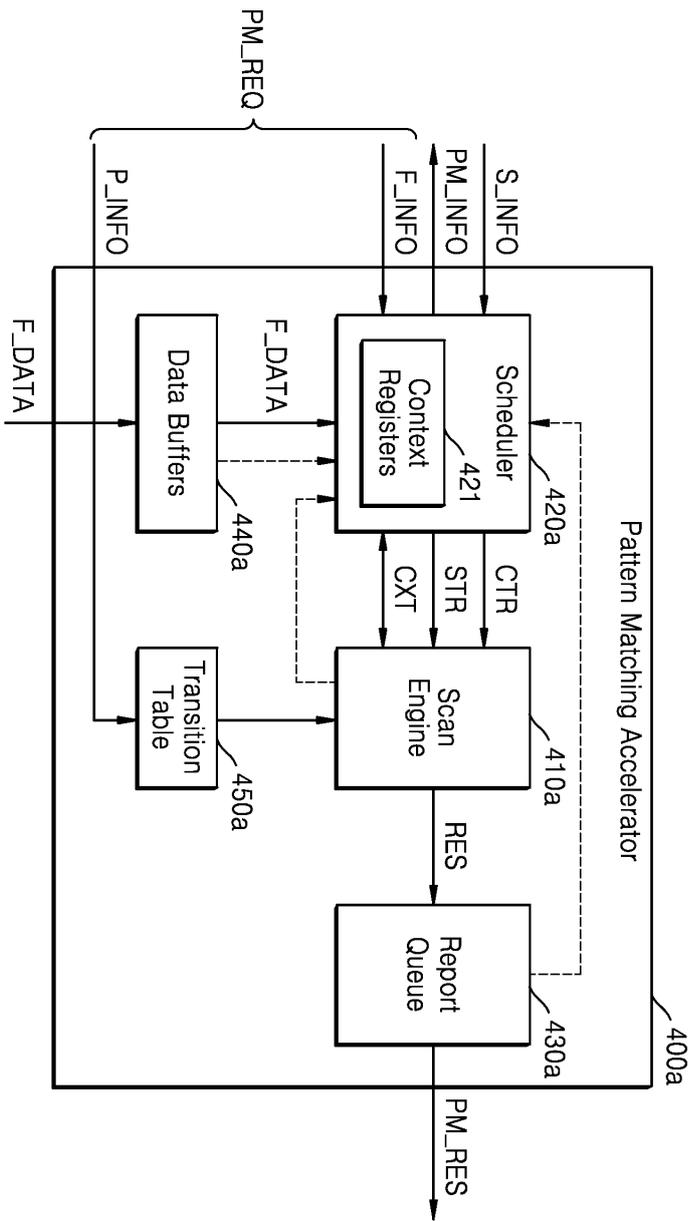


도면5

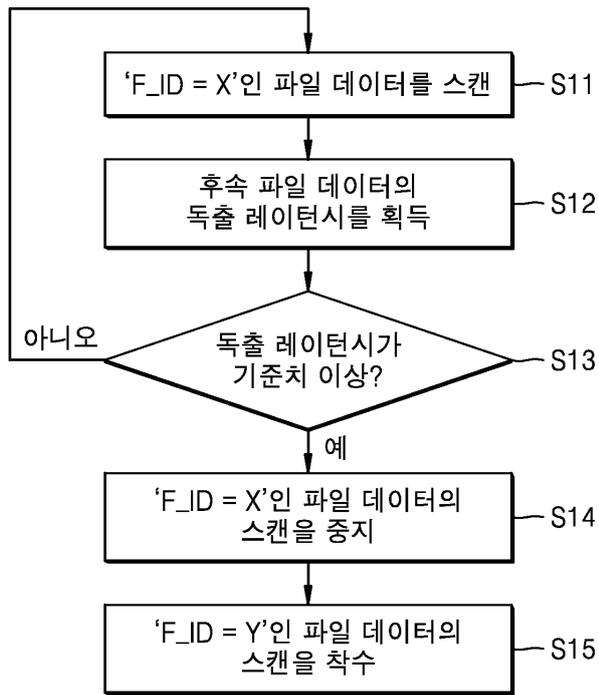
도면6



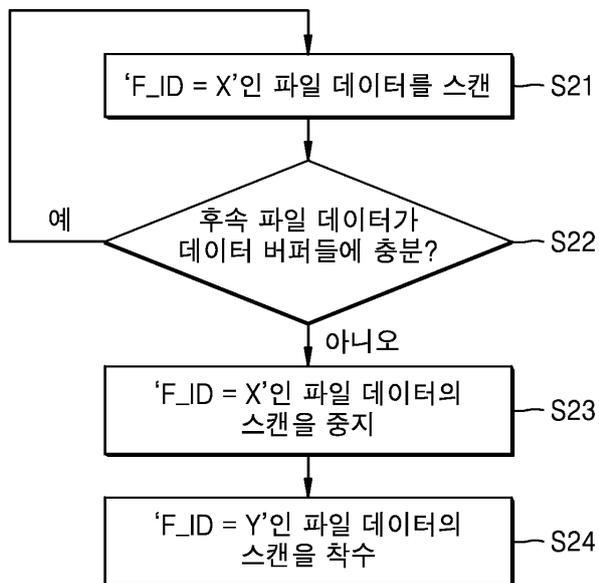
도면7



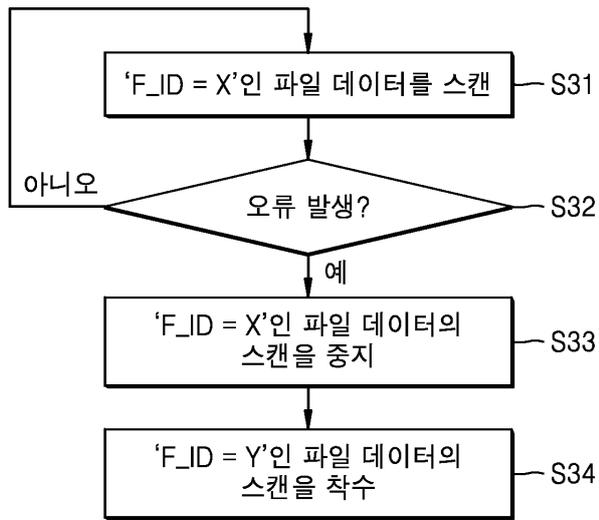
도면8a



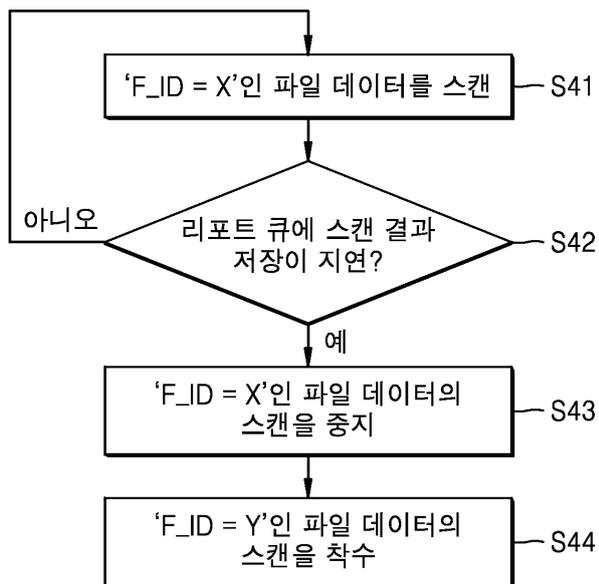
도면8b



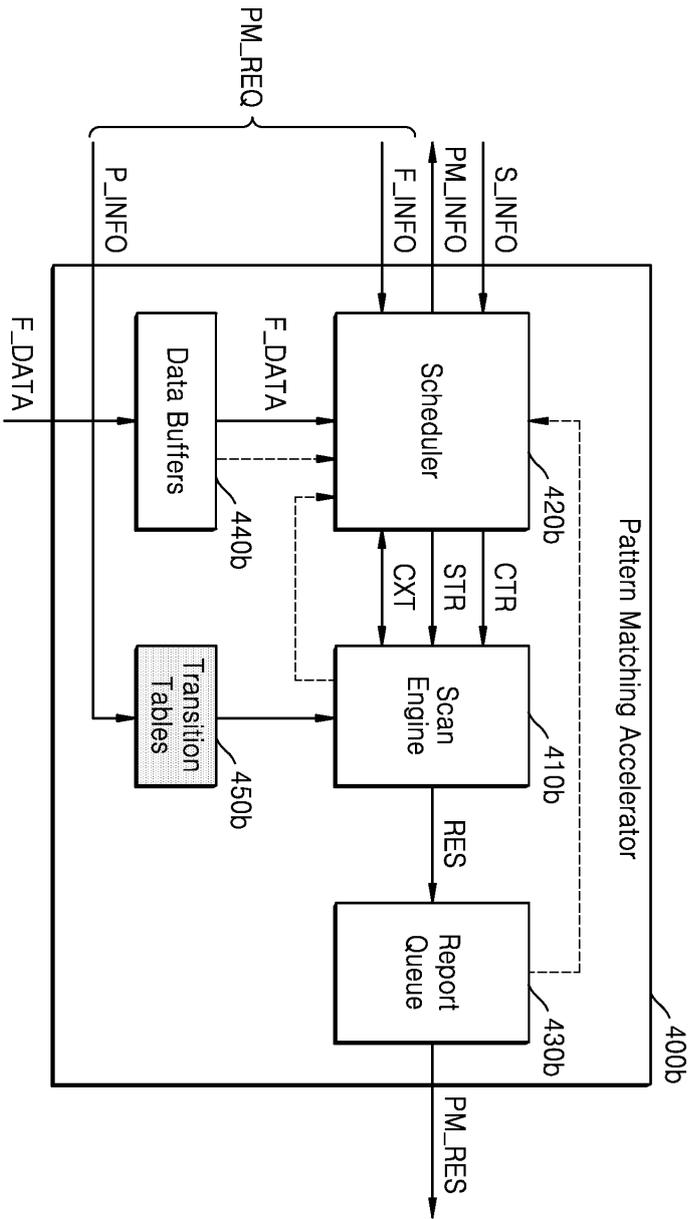
도면8c



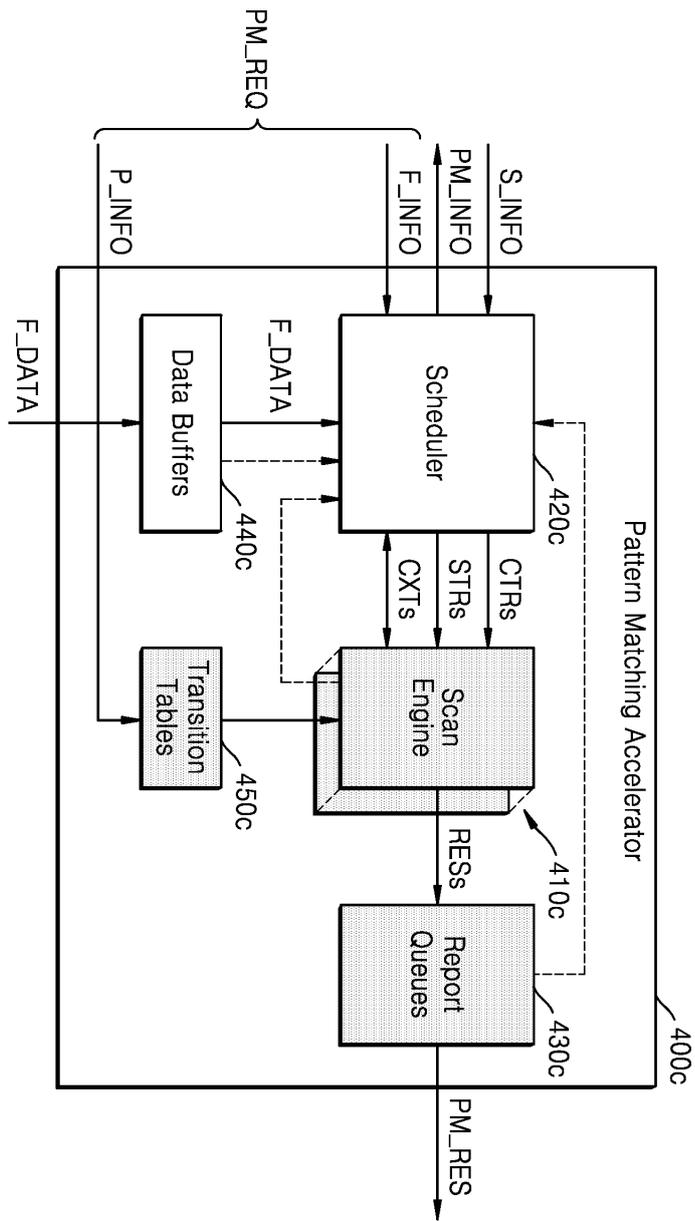
도면8d



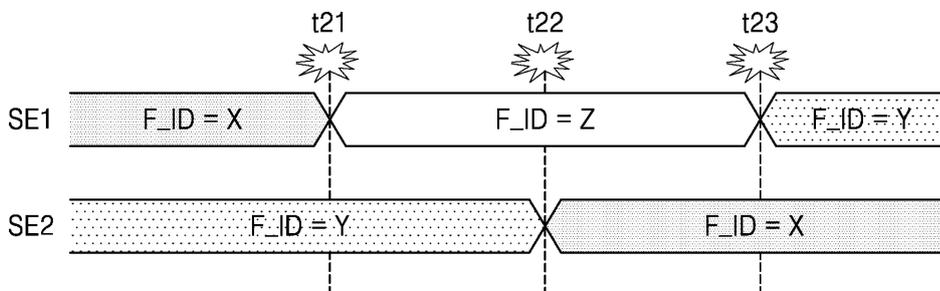
도면9



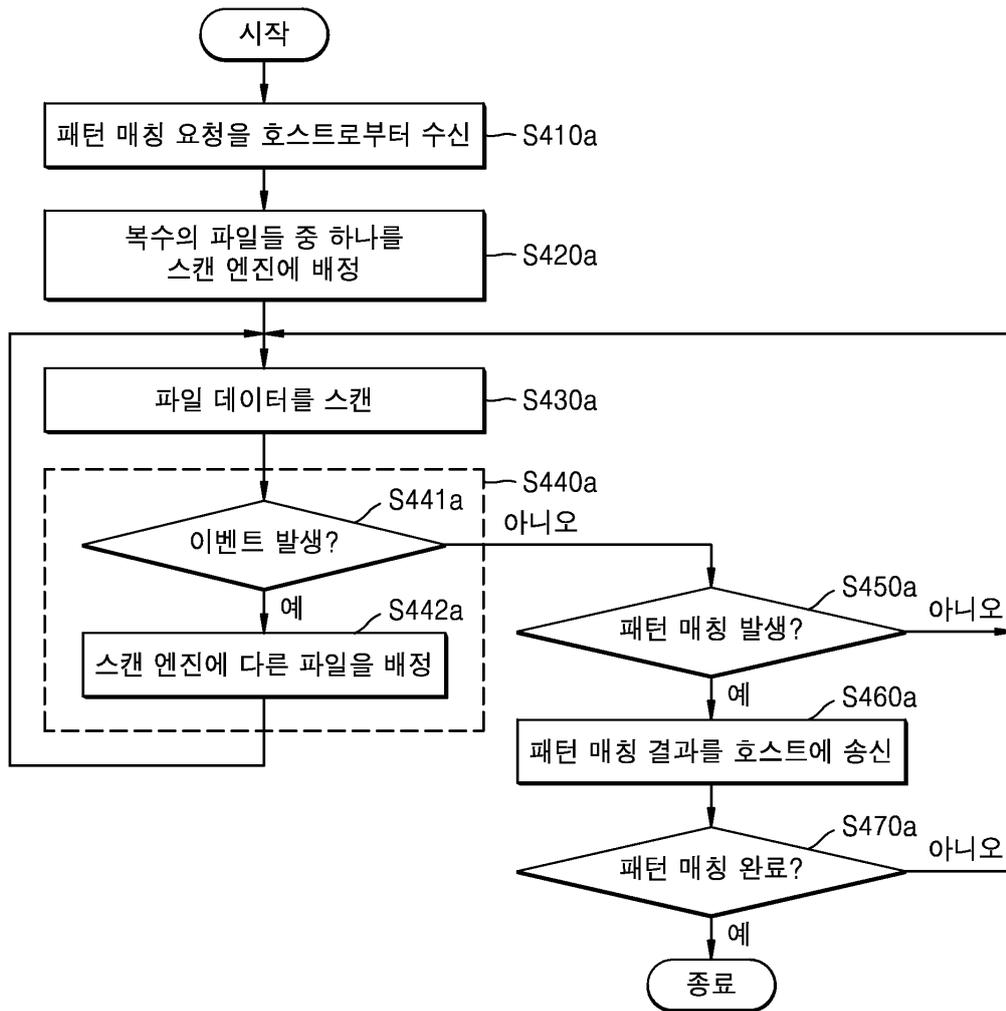
도면10



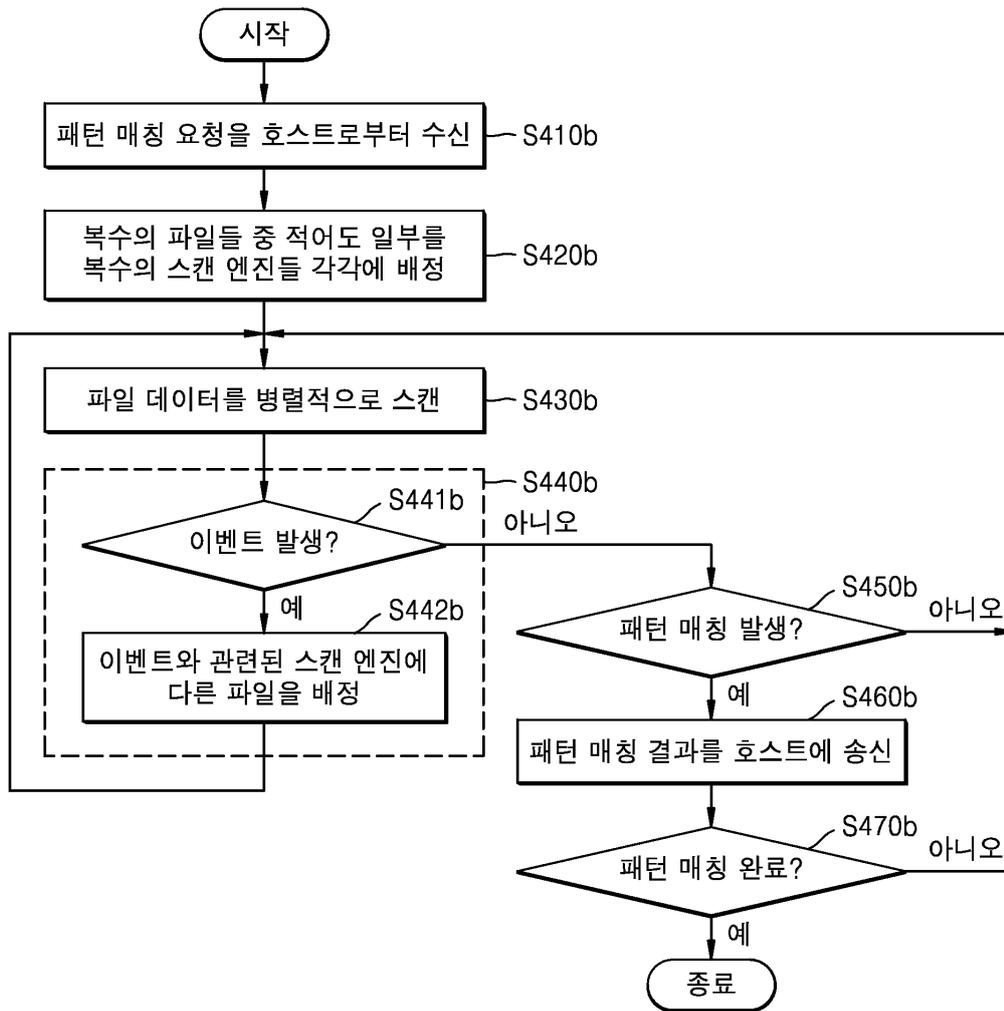
도면11



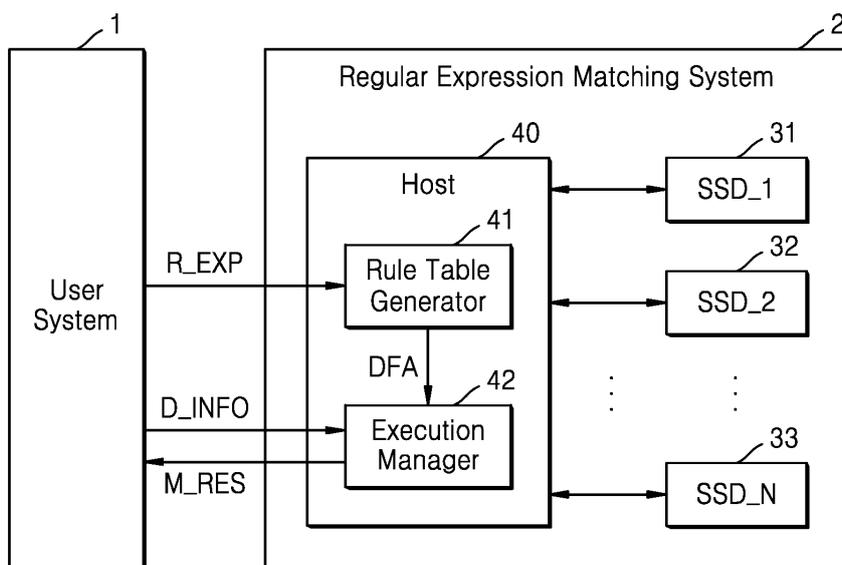
도면12



도면13



도면14



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 청구항 9

【변경전】

청구항 1에 있어서,

상기 패턴 매칭 가속기는 상기 스캔 엔진의 상태 정보를 저장하는 복수의 컨택스터 레지스터들을 포함하고,

상기 스케줄러는, 상기 제1 파일에 대한 스캔 중지시 상기 스캔 엔진의 상태 정보를 제1 컨택스트 레지스터에 저장하고, 상기 제2 파일에 대한 스캔 착수시 제2 컨택스트 레지스터에 저장된 상태 정보를 상기 스캔 엔진에 로딩하는 것을 특징으로 하는 스토리지 시스템.

【변경후】

청구항 1에 있어서,

상기 패턴 매칭 가속기는 상기 스캔 엔진의 상태 정보를 저장하는 복수의 컨택스트 레지스터들을 포함하고,

상기 스케줄러는, 상기 제1 파일에 대한 스캔 중지시 상기 스캔 엔진의 상태 정보를 제1 컨택스트 레지스터에 저장하고, 상기 제2 파일에 대한 스캔 착수시 제2 컨택스트 레지스터에 저장된 상태 정보를 상기 스캔 엔진에 로딩하는 것을 특징으로 하는 스토리지 시스템.