



(19) 대한민국특허청(KR)
(12) 공개특허공보(A)

(11) 공개번호 10-2014-0111280
(43) 공개일자 2014년09월18일

(51) 국제특허분류(Int. Cl.)
H04N 5/76 (2006.01) H04N 9/82 (2006.01)
H04N 21/43 (2011.01)
(21) 출원번호 10-2014-7019114
(22) 출원일자(국제) 2012년12월20일
심사청구일자 없음
(85) 번역문제출일자 2014년07월09일
(86) 국제출원번호 PCT/US2012/070903
(87) 국제공개번호 WO 2013/106184
국제공개일자 2013년07월18일
(30) 우선권주장
61/584,351 2012년01월09일 미국(US)

(71) 출원인
툼슨 라이선싱
프랑스 이씨레물리노 튀 잔 다르크 1-5 (우:
92130)
(72) 발명자
클레텐저, 브라이언 두에인
미국 인디애나주 47304 먼시 스콰이어 코트 4105
맥클리어리, 마크 앨런
미국 인디애나주 46220 인디애나폴리스 노스 델라
웨어 스트리트 5825
(74) 대리인
특허법인아주양현

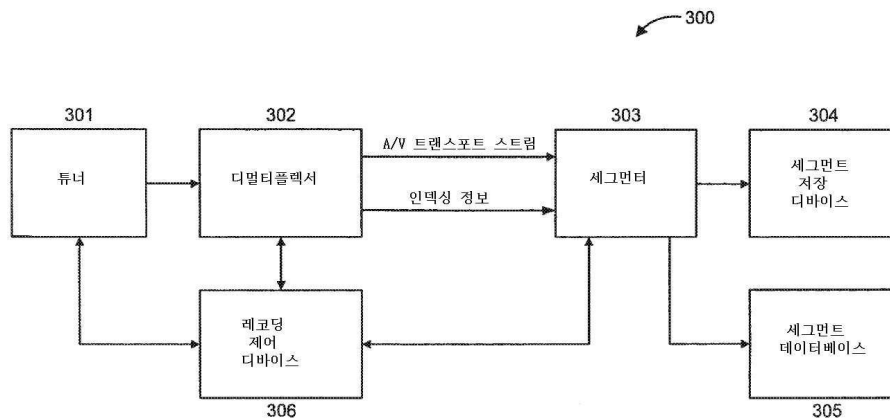
전체 청구항 수 : 총 22 항

(54) 발명의 명칭 시간-시프트 데이터의 관리

(57) 요약

미디어 콘텐츠를 레코딩하고 플레이백하는 각 시스템 및 방법이 제공된다. 미디어 콘텐츠를 레코딩하는 시스템은 상기 미디어 콘텐츠에 대응하는 트랜스포트 스트림 데이터 및 인덱스 데이터를 액세스 포인트 경계에서 주기적인 간격으로 이산 파일로 분할하는 세그먼트(303)를 포함한다. 각 상기 이산 파일은 각 트랜스포트 세그먼트 및 각 인덱스 세그먼트를 포함한다. 상기 시스템은 상기 이산 파일을 저장하는 세그먼트 저장 디바이스(304)를 더 포함한다.

대표도 - 도3



(72) 발명자

르 가르장, 브루노

프랑스 에프-35000 렌 아파트 52에이 뒤 드 브레스
트 20

풀런, 벤자민 엘런

미국 워싱턴주 98052 레드몬드 145 애비뉴 노스웨
스트 6220

특허청구의 범위

청구항 1

미디어 콘텐츠를 레코딩하는 시스템으로서,

상기 미디어 콘텐츠에 대응하는 트랜스포트 스트림 데이터 및 인덱스 데이터를 액세스 포인트 경계에서 주기적인 간격으로 이산 파일들로 분할하는 세그먼트(segmenter)(303)로서, 각 상기 이산 파일들은 각 트랜스포트 세그먼트 및 각 인덱스 세그먼트를 포함하는 것인, 상기 세그먼트(303); 및

상기 이산 파일들을 저장하는 세그먼트 저장 디바이스(304)를 포함하는 미디어 콘텐츠 레코딩 시스템.

청구항 2

제1항에 있어서, 세그먼트 데이터베이스(305)를 더 포함하되, 상기 세그먼트(303)는 트랜스포트 세그먼트 그룹이 시퀀스로 플레이백될 수 있게 하는 정보를 상기 세그먼트 데이터베이스(305)에 저장하고, 각 상기 트랜스포트 세그먼트 그룹은 각 프로그램에 대응하는 것인 미디어 콘텐츠 레코딩 시스템.

청구항 3

제1항에 있어서, 세그먼트 레코드를 저장하기 위한 세그먼트 데이터베이스(305)를 더 포함하되, 각 상기 세그먼트 레코드는 상기 각 트랜스포트 세그먼트에 대응하고, 고유한 세그먼트 식별자, 상기 각 트랜스포트 세그먼트에 대한 트랜스포트 스트림 데이터를 저장하는 파일을 나타내는 트랜스포트 파일 지시자, 상기 각 트랜스포트 세그먼트에 대한 인덱스 정보를 저장하는 파일을 나타내는 인덱스 파일 지시자, 상기 각 트랜스포트 세그먼트의 시작 시간, 및 상기 각 트랜스포트 세그먼트의 종료 시간을 포함하는 것인 미디어 콘텐츠 레코딩 시스템.

청구항 4

제1항에 있어서, 다른 레코딩으로부터 새로운 레코딩의 시작부로 세그먼트를 프리펜드(prepending)하여 프리펜드된 세그먼트를 포함하는 새로운 레코드를 끊임없이 플레이백하는 레코딩 제어 디바이스(306)를 더 포함하는, 미디어 콘텐츠 레코딩 시스템.

청구항 5

제1항에 있어서, 진행중인 레코딩의 시작부로부터 세그먼트를 절단하여 고정된 길이를 가지는 롤링 버퍼(rolling buffer)를 시뮬레이션하는 레코딩 제어 디바이스(306)를 더 포함하는, 미디어 콘텐츠 레코딩 시스템.

청구항 6

제1항에 있어서, 상기 이산 파일들은 기존의 또는 레거시(legacy) 파일 시스템을 사용하여 저장되는 것인 미디어 콘텐츠 레코딩 시스템.

청구항 7

미디어 콘텐츠를 레코딩하는 방법으로서,

상기 미디어 콘텐츠에 대응하는 트랜스포트 스트림 데이터 및 인덱스 데이터를 액세스 포인트 경계에서 주기적인 간격으로 이산 파일들로 분할하는 단계(645)로서, 각 상기 이산 파일들은 각 트랜스포트 세그먼트 및 각 인덱스 세그먼트를 포함하는 것인, 상기 분할하는 단계; 및

상기 이산 파일들을 세그먼트 저장 디바이스에 저장하는 단계(645)를 포함하는 미디어 콘텐츠 레코딩 방법.

청구항 8

제7항에 있어서, 트랜스포트 세그먼트 그룹이 시퀀스로 플레이백될 수 있게 하는 정보를 세그먼트 데이터베이스에 저장하는 단계(645)를 더 포함하되, 각 상기 트랜스포트 세그먼트 그룹은 각 프로그램에 대응하는 것인 미디어 콘텐츠 레코딩 미디어 콘텐츠 레코딩 방법.

청구항 9

제7항에 있어서, 세그먼트 레코드를 세그먼트 데이터베이스에 저장하는 단계(645)를 더 포함하되, 각 상기 세그먼트 레코드는 상기 각 트랜스포트 세그먼트에 대응하고, 고유한 세그먼트 식별자, 상기 각 트랜스포트 세그먼트에 대한 트랜스포트 스트림 데이터를 저장하는 파일을 나타내는 트랜스포트 파일 지시자, 상기 각 트랜스포트 세그먼트에 대한 인덱스 정보를 저장하는 파일을 나타내는 인덱스 파일 지시자, 상기 각 트랜스포트 세그먼트의 시작 시간, 및 상기 각 트랜스포트 세그먼트의 종료 시간을 포함하는 것인 미디어 콘텐츠 레코딩 방법.

청구항 10

제7항에 있어서, 다른 레코딩으로부터 새로운 레코딩의 시작부로 세그먼트를 프리퀀드하여 프리퀀드된 세그먼트를 포함하는 새로운 레코딩을 끊임없이 플레이백하는 단계(1100)를 더 포함하는 미디어 콘텐츠 레코딩 방법.

청구항 11

제7항에 있어서, 진행 중인 레코딩의 시작부로부터 세그먼트를 절단하여 고정된 길이를 가지는 롤링 버퍼를 시뮬레이션하는 단계(1000)를 더 포함하는, 미디어 콘텐츠 레코딩 방법.

청구항 12

제7항에 있어서, 상기 이산 파일들은 기존의 또는 레거시 파일 시스템을 사용하여 저장된 것인 미디어 콘텐츠 레코딩 방법.

청구항 13

컴퓨터 시스템으로 하여금 제7항의 방법을 수행하게 하는 컴퓨터-판독가능한 프로그램 코드를 구비하는 컴퓨터-판독가능한 저장 매체.

청구항 14

미디어 콘텐츠를 플레이백하는 시스템으로서,

이산 파일들을 저장하는 세그먼트 저장 디바이스(402)로서, 각 상기 이산 파일들은 상기 미디어 콘텐츠에 대응하는 트랜스포트 스트림 데이터 및 인덱스 데이터로부터 획득된 각 트랜스포트 세그먼트 및 각 인덱스 세그먼트를 포함하는 것인, 상기 세그먼트 저장 디바이스(402); 및

상기 트랜스포트 세그먼트를 끊임없이 플레이백하기 위해 상기 이산 파일들로부터 트랜스포트 세그먼트를 재조립하는 디세그먼트(desegmenter)(403)를 포함하는 미디어 콘텐츠 플레이백 시스템.

청구항 15

제14항에 있어서, 세그먼트 데이터베이스(401)를 더 포함하되, 상기 디세그먼트(403)는 상기 트랜스포트 세그먼트 그룹이 시퀀스로 플레이백될 수 있게 하는 정보를 상기 세그먼트 데이터베이스(401)로부터 검색하고, 각 상기 트랜스포트 세그먼트 그룹은 각 프로그램에 대응하는 것인 미디어 콘텐츠 플레이백 시스템.

청구항 16

제14항에 있어서, 세그먼트 레코드를 저장하는 세그먼트 데이터베이스(401)를 더 포함하되, 각 상기 세그먼트 레코드는 상기 각 트랜스포트 세그먼트에 대응하고, 고유한 세그먼트 식별자, 상기 각 트랜스포트 세그먼트에 대한 트랜스포트 스트림 데이터를 저장하는 파일을 나타내는 트랜스포트 파일 지시자, 상기 각 트랜스포트 세그먼트에 대한 인덱스 정보를 저장하는 파일을 나타내는 인덱스 파일 지시자, 상기 각 트랜스포트 세그먼트의 시작 시간, 및 상기 각 트랜스포트 세그먼트의 종료 시간을 포함하는 것인 미디어 콘텐츠 플레이백 시스템.

청구항 17

제14항에 있어서, 상기 이산 파일들은 기존의 또는 레거시 파일 시스템을 사용하여 저장된 것인 미디어 콘텐츠 플레이백 시스템.

청구항 18

미디어 콘텐츠를 플레이백하는 방법으로서,

이산 파일들을 세그먼트 저장 디바이스에 저장하는 단계(645)로서, 각 상기 이산 파일들은 상기 미디어 콘텐츠에 대응하는 트랜스포트 스트림 데이터 및 인덱스 데이터로부터 획득된 각 트랜스포트 세그먼트 및 각 인덱스 세그먼트를 포함하는 것인, 상기 저장하는 단계;

상기 트랜스포트 세그먼트를 끊임없이 플레이백하기 위해 상기 이산 파일들로부터 트랜스포트 세그먼트를 재조립하는 단계(1230)를 포함하는 미디어 콘텐츠 플레이백 방법.

청구항 19

제18항에 있어서, 상기 트랜스포트 세그먼트 그룹이 시퀀스로 플레이백될 수 있게 하는 정보를 세그먼트 데이터베이스로부터 검색하는 단계를 더 포함하며, 각 상기 트랜스포트 세그먼트 그룹은 각 프로그램에 대응하는 것인 미디어 콘텐츠 플레이백 방법.

청구항 20

제18항에 있어서, 세그먼트 레코드를 세그먼트 데이터베이스에 저장하는 단계를 더 포함하며, 각 상기 세그먼트 레코드는 상기 각 트랜스포트 세그먼트에 대응하고, 고유한 세그먼트 식별자, 상기 각 트랜스포트 세그먼트에 대한 트랜스포트 스트림 데이터를 저장하는 파일을 나타내는 트랜스포트 파일 지시자, 상기 각 트랜스포트 세그먼트에 대한 인덱스 정보를 저장하는 파일을 나타내는 인덱스 파일 지시자, 상기 각 트랜스포트 세그먼트의 시작 시간, 및 상기 각 트랜스포트 세그먼트의 종료 시간을 포함하는 것인 미디어 콘텐츠 플레이백 방법.

청구항 21

제18항에 있어서, 상기 이산 파일들은 기존의 또는 레거시 파일 시스템을 사용하여 저장되는 것인 미디어 콘텐츠 플레이백 방법.

청구항 22

컴퓨터 시스템으로 하여금 제18항의 방법을 수행하게 하는 컴퓨터-판독가능한 프로그램 코드를 구비하는 컴퓨터-판독가능한 저장 매체.

명세서

기술 분야

[0001] 관련 출원에 대한 상호 참조

[0002] 본 출원은 전체 내용이 본 명세서에 참조 문헌으로 병합된 2012년 1월 9일에 출원된 미국 가출원 번호 61/584,351의 이익을 청구한다.

[0003] 기술 분야

[0004] 본 발명의 원리는 일반적으로 시간-시프트 기능에 관한 것으로, 보다 상세하게는, 시간-시프트 데이터를 관리하는 것에 관한 것이다.

배경 기술

[0005] 디지털 비디오 레코더(DVR)의 핵심 기능은 비디오 콘텐츠(및 이에 대응하는 오디오 콘텐츠, 이는 이후 간략화를 위해 "비디오 콘텐츠"라고 집합적으로 지칭됨)와 같은 방송 멀티미디어 콘텐츠를 레코딩하고 플레이백하는 것이다. 많은 DVR은 시간-시프트 기능을 허용한다. 시간-시프트 기능은 본질적으로 라이브 방송 콘텐츠의 롤링 버퍼(rolling buffer)이다. 이것은 일시정지(pause), 되감기(rewind), 및 고속 전진(fast forward)과 같은 트릭 모드(trick mode) 동작을 라이브 방송 콘텐츠에서 사용될 수 있게 한다. 일반적인 시간-시프트 구현은 마지막 30분의 라이브 방송 콘텐츠의 버퍼를 유지하여, 사용자로 하여금 트릭 모드에서 이 버퍼를 네비게이션할 수 있게 한다.

[0006] 비디오 콘텐츠를 저장하는 데는 상당한 저장 공간을 요구하므로, 시간-시프트 기능에 사용되는 비디오 데이터는 통상적으로 대부분의 구현에서 하드 디스크에 저장된다. 이것은 새로운 데이터가 시간-시프트 레코딩에 추가되

어서, 구 데이터가 중단으로부터 절단되어야 하기 때문에 문제를 야기한다. 대부분의 파일 시스템은 파일의 중단에 새로운 데이터를 효율적으로 삽입할 수 있지만, 대부분의 파일 시스템은 파일의 시작으로부터 데이터를 절단하는 효과적인 메커니즘을 제공하지는 않는다. 일부 DVR은 파일의 시작부로부터 절단하는 것과 파일의 중단에 삽입하는 특별히 설계된 파일 시스템을 사용하는 것에 의해 이 문제를 해결한다. 그러나, 특별히 설계된 파일 시스템을 사용하는 것은 DVR에서 사용되는 표준 파일 시스템에 특별히 설계된 파일 시스템을 통합하는 것을 포함하여 이것으로만 제한되지는 않는 동작이 허용되지 않는다.

발명의 내용

- [0007] 종래 기술의 이들 및 다른 단점은 시간-시프트 데이터를 관리하는 것에 관한 본 발명의 원리에 의해 해결된다.
- [0008] 본 발명의 원리의 일 측면에 따르면, 미디어 콘텐츠를 레코딩하는 시스템이 제공된다. 본 시스템은 미디어 콘텐츠에 대응하는 트랜스포트 스트림 데이터 및 인덱스 데이터를 액세스 포인트 경계에서 주기적인 간격으로 이산 파일들로 분할하는 세그먼트(segmenter)를 포함한다. 각 이산 파일들은 각 트랜스포트 세그먼트 및 각 인덱스 세그먼트를 포함한다. 본 시스템은 이산 파일들을 저장하는 세그먼트 저장 디바이스를 더 포함한다.
- [0009] 본 발명의 원리의 다른 측면에 따르면, 미디어 콘텐츠를 레코딩하는 방법이 제공된다. 본 방법은 미디어 콘텐츠에 대응하는 트랜스포트 스트림 데이터 및 인덱스 데이터를 액세스 포인트 경계에서 주기적인 간격으로 이산 파일들로 분할하는 단계를 포함한다. 각 이산 파일들은 각 트랜스포트 세그먼트 및 각 인덱스 세그먼트를 포함한다. 본 방법은 이 이산 파일들을 세그먼트 저장 디바이스에 저장하는 단계를 더 포함한다.
- [0010] 본 발명의 원리의 또 다른 측면에 따르면, 미디어 콘텐츠를 플레이백하는 시스템이 제공된다. 본 시스템은 이산 파일들을 저장하는 세그먼트 저장 디바이스를 포함한다. 각 이산 파일들은 미디어 콘텐츠에 대응하는 트랜스포트 스트림 데이터 및 인덱스 데이터로부터 획득된 각 트랜스포트 세그먼트 및 각 인덱스 세그먼트를 포함한다. 본 시스템은 트랜스포트 세그먼트를 끊임없이 플레이백하기 위해 이산 파일들로부터 트랜스포트 세그먼트를 재조립하는 디세그먼트(desegmenter)를 더 포함한다.
- [0011] 본 발명의 원리의 더 다른 측면에 따르면, 미디어 콘텐츠를 플레이백(play back)하는 방법이 제공된다. 본 방법은 이산 파일들을 세그먼트 저장 디바이스에 저장하는 단계를 포함한다. 각 이산 파일들은 미디어 콘텐츠에 대응하는 트랜스포트 스트림 데이터 및 인덱스 데이터로부터 획득된 각 트랜스포트 세그먼트 및 각 인덱스 세그먼트를 포함한다. 본 방법은 트랜스포트 세그먼트를 끊임없이 플레이백하기 위해 이산 파일들로부터 트랜스포트 세그먼트를 재조립하는 단계를 더 포함한다.
- [0012] 본 발명의 원리의 이들 및 다른 측면, 특징 및 장점은 첨부 도면과 관련하여 판독되는 예시적인 실시예의 이하 상세한 설명으로부터 명백해질 것이다.

도면의 간단한 설명

- [0013] 본 발명의 원리는 이하 예시적인 도면에 따라 더 잘 이해될 수 있을 것이다.
- 도 1은 본 발명의 일 실시예에 따라 본 발명의 원리를 적용할 수 있는 예시적인 처리 시스템(100)을 도시한 도면;
- 도 2는 본 발명의 일 실시예에 따라 디지털 비디오 레코더 기능 및 시간-시프트 기능을 포함하는 예시적인 셋탑 박스(200)를 도시한 도면;
- 도 3은 본 발명의 일 실시예에 따라 본 발명의 원리를 적용할 수 있는 예시적인 레코딩 파이프라인(300)을 도시한 도면;
- 도 4는 본 발명의 일 실시예에 따라 본 발명의 원리를 적용할 수 있는 예시적인 플레이백 파이프라인(300)을 도시한 도면;
- 도 5는 본 발명의 일 실시예에 따라 세그먼트 데이터베이스에 존재하는 4개의 예시적인 레코드 유형(500)을 도시한 도면;
- 도 6은 본 발명의 일 실시예에 따라 레코딩을 시작하는 예시적인 방법(600)을 도시한 도면;
- 도 7은 본 발명의 일 실시예에 따라 레코딩을 중지하는 예시적인 방법(700)을 도시한 도면;
- 도 8은 본 발명의 일 실시예에 따라 레코딩을 삭제하는 예시적인 방법(800)을 도시한 도면;

도 9는 본 발명의 일 실시예에 따라 세그먼트를 가비지 수집(garbage collecting)하는 예시적인 방법(900)을 도시한 도면;

도 10은 본 발명의 일 실시예에 따라 레코딩의 시작부로부터 데이터를 절단하는 예시적인 방법(1000)을 도시한 도면;

도 11은 본 발명의 일 실시예에 따라 레코딩의 시작부에 세그먼트를 첨부(appending)하는 예시적인 방법(1100)을 도시한 도면;

도 12는 본 발명의 일 실시예에 따라 레코딩을 플레이백하는 예시적인 방법(1200)을 도시한 도면;

도 13은 본 발명의 일 실시예에 따라 예시적인 레코딩 트리 인덱스(tree index)(1300)를 도시한 도면.

발명을 실시하기 위한 구체적인 내용

- [0014] 본 발명의 원리는 시간-시프트 데이터를 관리하는 것에 관한 것이다.
- [0015] 이 기술 분야에 통상의 지식을 가진 자라면 본 발명의 원리를 적용할 수 있는 여러 응용을 용이하게 고려할 수 있을 것이지만, 이하 설명은 디지털 비디오 레코더 기능 및 시간-시프트 기능을 갖는 셋탑 박스에 적용된 본 발명의 원리의 실시예에 초점을 둔다. 그러나, 이 기술 분야에 통상의 지식을 가진 자라면 본 명세서에 제공된 본 발명의 원리의 내용을 통해, 본 발명의 원리의 사상을 유지하면서 본 발명의 원리를 적용할 수 있는 다른 디바이스 및 응용을 용이하게 고려할 수 있을 것이다. 예를 들어, 본 발명의 원리는 레코딩 및 시간-시프트 기능과 함께 멀티미디어 플레이백 능력을 구비하는 임의의 디바이스에 통합될 수 있다. 따라서, 본 발명의 원리는 디지털 비디오 레코더(DVR) 능력을 갖는 셋탑 박스, 독립형 DVR, 홈 미디어 서버, 컴퓨터 미디어 스테이션, 태블릿 디바이스, 홈 네트워크 게이트웨이, 멀티-미디어 플레이어, 홈 네트워킹 기기 등에 포함될 수 있다. 디바이스의 이전의 리스트는 단지 예시적인 것일 뿐 이로 제한되는 것은 아니다.
- [0016] 전술된 종래 기술에 대해, 및 시간-시프트 레코딩의 종료시로부터 구 데이터를 절단하면서 새로운 데이터를 시간-시프트 레코딩에 추가하는 문제를 해결하는 특별히 설계된 파일 시스템을 사용하는 것에 대해, 본 출원인은 처음에 특별히 설계된 파일 시스템을 요구하지 않고 이 문제를 해결하는 본 발명의 원리를 설명한다. 유리하게는, 이것은 표준 파일 시스템(본 명세서에서 "기존의" 또는 "레거시(legacy)" 파일 시스템이라고도 지칭됨)이 비디오 저장에 사용될 수 있게 한다. 더욱이, 본 발명의 원리는 중첩 레코딩(overlapping recording)들 사이에 트랜스포트 데이터를 공유할 수 있게 하여, 저장 공간을 감소시킨다.
- [0017] 도 1은 본 발명의 일 실시예에 따라 본 발명의 원리를 적용할 수 있는 예시적인 처리 시스템(100)을 도시한다. 이 처리 시스템(100)은 시스템 버스(104)를 통해 다른 컴포넌트에 동작가능하게 연결된 적어도 하나의 프로세서(CPU)(102)를 포함한다. 판독 전용 메모리(ROM)(106), 랜덤 액세스 메모리(RAM)(108), 디스플레이 어댑터(110), 입력/출력(I/O) 어댑터(112), 사용자 인터페이스 어댑터(114), 및 네트워크 어댑터(198)는 시스템 버스(104)에 동작가능하게 연결된다.
- [0018] 디스플레이 디바이스(116)는 디스플레이 어댑터(110)에 의해 시스템 버스(104)에 동작가능하게 연결된다. 디스크 저장 디바이스(예를 들어, 자기 또는 광 디스크 저장 디바이스)(118)는 I/O 어댑터(112)에 의해 시스템 버스(104)에 동작가능하게 연결된다.
- [0019] 마우스(120) 및 키보드(122)는 사용자 인터페이스 어댑터(214)에 의해 시스템 버스(104)에 동작가능하게 연결된다. 마우스(120) 및 키보드(122)는 시스템(100)에 정보를 입출력하는데 사용된다.
- [0020] 트랜시버(196)는 네트워크 어댑터(198)에 의해 시스템 버스(104)에 연결된다.
- [0021] 물론, 이 처리 시스템(100)은, 이 기술 분야에 통상의 지식을 가진 자라면 용이하게 고려할 수 있는 다른 요소(미도시)를 더 포함할 수 있고, 또는 특정 요소를 생략할 수 있다. 예를 들어, 여러 다른 입력 디바이스 및/또는 출력 디바이스는 이 기술 분야에 통상의 지식을 가진 자라면 용이하게 이해하는 바와 같이 특정 구현에 따라 처리 시스템(100)에 포함될 수 있다. 예를 들어, 여러 유형의 무선 및/또는 유선 입력 및/또는 출력 디바이스들이 사용될 수 있다. 더욱이, 여러 구성으로 추가적인 프로세서, 제어기, 메모리 등이 이 기술 분야에 통상의 지식을 가진 자라면 용이하게 이해할 수 있는 바와 같이 더 사용될 수 있다. 처리 시스템(100)의 이들 및 다른 변형이 본 명세서에 제공된 본 발명의 원리의 내용을 통해 이 기술 분야에 통상의 지식을 가진 자라면 용이하게 고려할 수 있을 것이다.

- [0022] 더욱이, 도 2에 대하여 후술되는 셋탑 박스(200)는 본 발명의 원리의 각 실시예를 구현하기 위한 셋탑 박스(200)인 것으로 이해된다. 나아가, 도 3 및 도 4에 대해 각각 아래에 설명된 레코딩 파이프라인(300) 및 플레이백 파이프라인(400)은 셋탑 박스(200)에 포함되는 것으로 이해된다. 물론, 전술된 바와 같이, 본 발명의 원리는 셋탑 박스로 제한되지 않고, 레코딩 파이프라인(300) 및 플레이백 파이프라인(400)은 본 발명의 사상을 벗어남이 없이 전술된 디바이스를 포함하는 다른 디바이스에 포함될 수 있을 것이다. 처리 시스템(100)의 일부나 전부는 셋탑 박스(200)(또는 다른 디바이스), 레코딩 파이프라인(300), 및/또는 플레이백 파이프라인(400) 중 하나 이상의 요소에 구현될 수 있다.
- [0023] 추가적으로, 처리 시스템(100)은, 예를 들어, 도 6의 방법(600)의 적어도 일부 및/또는 도 7의 방법(700)의 적어도 일부를 포함하는 본 명세서에 설명된 방법의 적어도 일부를 수행할 수 있다. 유사하게, 셋탑 박스(200), 레코딩 파이프라인(300), 및/또는 플레이백 파이프라인(400)의 일부나 전부는 방법(600)의 적어도 일부 및/또는 방법(700)의 적어도 일부를 수행하는데 사용될 수 있다.
- [0024] 도 2는 본 발명의 일 실시예에 따라 디지털 비디오 레코더 기능 및 시간-시프트 기능을 포함하는 예시적인 셋탑 박스(200)를 도시한다. 셋탑 박스(200)의 전체 동작에 필요한 여러 컴포넌트들은 이 기술 분야에 통상의 지식을 가진 자에게 잘 알려져 있는 것이므로 간결함을 위하여 도시되지 않았다.
- [0025] 신호는 입력 신호 수신기(202)에서 셋탑 박스(200)와 인터페이스된다. 입력 신호 수신기(202)는 입력 스트림 프로세서(204)에 연결된다. 입력 스트림 프로세서(204)는 오디오 프로세서(206) 및 비디오 프로세서(210)에 연결된다. 오디오 프로세서(206)는 오디오 인터페이스(208)에 연결되고, 셋탑 박스(200)로부터 오디오 출력 신호를 제공한다. 비디오 프로세서(210)는 디스플레이 인터페이스(218)에 연결되고, 셋탑 박스(200)로부터 비디오 출력 신호를 제공한다. 오디오 프로세서(206) 및 비디오 프로세서(210)는 저장 디바이스(212)에 더 연결된다. 제어기(214)는 입력 스트림 프로세서(204), 오디오 프로세서(206), 및 비디오 프로세서(210) 및 저장 디바이스(212)에 연결된다. 제어 메모리(220)는 제어기(214)에 연결된다. 제어기(214)는 사용자 인터페이스(216) 및 핸드헬드 인터페이스(222)에 더 연결된다.
- [0026] 콘텐츠는 입력 신호 수신기(202)에 수신된다. 입력 신호 수신기(202)는 공중, 케이블, 위성, 이더넷, 섬유(fiber) 및 전화선 네트워크를 포함하는 여러 가능한 네트워크 중 하나를 통해 제공되는 신호를 수신하고 복조하며 디코딩하는데 사용되는 여러 알려진 수신기 회로 중 하나 이상일 수 있다. 입력 신호 수신기(202)는 원하는 방송 입력 신호와 동일한 전달 네트워크 또는 상이한 네트워크, 및/또는 대안적인 셀룰러 또는 무선 네트워크를 통해 전달되는 미디어 콘텐츠 신호뿐만 아니라 데이터 신호를 위한 수신, 복조 및 디코딩 회로를 포함할 수 있다. 전달 네트워크(2)(또는 무선 네트워크)를 통해 수신되는 미디어 콘텐츠 및 데이터는 미디어 콘텐츠 및 전달 네트워크(1)와는 상이할 수 있다. 데이터는 전달 네트워크를 통해 전달되는 미디어 콘텐츠와 관련된 정보뿐만 아니라 스케줄링 변화 및 업데이트와 연관된 정보를 포함할 수 있다. 일 실시예에서, 케이블 방송 신호는 신호 수신기(202) 내 케이블 튜너 회로에서 수신되고 복조되고 디코딩된다. 원하는 방송 입력 신호는 제어 인터페이스(미도시)를 통해 제공되는 사용자 입력에 기초하여 입력 신호 수신기(202)에서 선택되고 검색될 수 있다. 입력 신호 수신기(202)는 양방향 네트워크 연결을 추가적으로 제공하는 IP(Internet protocol) 인터페이스 회로를 더 포함한다.
- [0027] 입력 신호 수신기(202) 내 하나 이상의 회로로부터 오는 디코딩된 출력 신호는 입력 스트림 프로세서(204)에 제공된다. 입력 스트림 프로세서(204)는 최종 신호 선택 및 처리를 수행하고, 콘텐츠 스트림을 위한 오디오 콘텐츠로부터 비디오 콘텐츠를 분리한다. 오디오 콘텐츠는 수신된 포맷(예를 들어, 압축된 디지털 신호)으로부터 다른 포맷(예를 들어, 아날로그 파형 신호)으로 변환하기 위해 오디오 프로세서(206)에 제공된다. 아날로그 파형 신호는 오디오 인터페이스(208)에 제공되고 디스플레이 디바이스(114) 또는 오디오 증폭기(미도시)에 더 제공된다. 대안적으로, 오디오 인터페이스(208)는 고선명 멀티미디어 인터페이스(High-Definition Multimedia Interface: HDMI) 케이블 또는 대안적인 오디오 인터페이스, 예를 들어 SPDIF(Sony/Philips Digital Interconnect Format)를 사용하여 오디오 출력 디바이스 또는 디스플레이 디바이스에 디지털 신호를 제공할 수 있다. 오디오 프로세서(206)는 오디오 신호를 저장하는데 필요한 변환을 더 수행한다.
- [0028] 입력 스트림 프로세서(204)로부터 비디오 출력은 비디오 프로세서(210)에 제공된다. 비디오 신호는 여러 포맷 중 하나일 수 있다. 비디오 프로세서(210)는 입력 신호 포맷에 기초하여 필요에 따라 비디오 콘텐츠를 변환한다. 비디오 프로세서(210)는 비디오 신호를 저장하기 위해 필요한 변환을 더 수행한다.
- [0029] 저장 디바이스(212)는 입력에 수신된 오디오 및 비디오 콘텐츠를 저장한다. 저장 디바이스(212)는 제어기(214)의 제어 하에 및 사용자 인터페이스(216)로부터 수신된 커맨드, 예를 들어, 네비게이션 명령, 예를 들어 고속

전진(fast-forward: FF) 및 되감기(rewind: Rew)에 기초하여 콘텐츠를 차후에 검색하고 플레이백할 수 있게 한다. 저장 디바이스(212)는 하드 디스크 드라이브, 하나 이상의 대용량 집적 전자 메모리, 예를 들어 정적 RAM(SRAM), 또는 동적 RAM(DRAM), 교환가능한 광 디스크 저장 시스템, 예를 들어 콤팩트 디스크 드라이브 또는 디지털 비디오 디스크 드라이브, 또는 셋탑 박스(200)의 외부에 있고 이 셋탑 박스에 의해 액세스가능한 저장 매체일 수 있다.

[0030] 입력으로부터 또는 저장 디바이스(212)로부터 발신되는 비디오 프로세서(210)로부터 오는 변환된 비디오 신호는 디스플레이 인터페이스(218)에 제공된다. 디스플레이 인터페이스(218)는 디스플레이 신호를 전송된 유형의 디스플레이 디바이스에 더 제공한다. 디스플레이 인터페이스(218)는 RGB(red-green-blue)와 같은 아날로그신호 인터페이스이거나, 또는 디지털 인터페이스(예를 들어, HDMI)일 수 있다.

[0031] 제어기(214)는 입력 스트림 프로세서(202), 오디오 프로세서(206), 비디오 프로세서(210), 저장 디바이스(212), 사용자 인터페이스(216), 및 핸드헬드 인터페이스(222)를 포함하는 셋탑 박스(200) 중 여러 컴포넌트에 버스를 통해 상호 연결된다. 제어기(214)는 입력 스트림 신호를 저장 디바이스에 저장하거나 또는 디스플레이를 하기 위한 신호로 변환하는 변환 공정을 관리한다. 제어기(214)는 저장된 콘텐츠를 검색하거나 플레이하는 것을 더 관리한다. 제어기(214)는 제어기(214)를 위한 정보와 명령 코드를 저장하는, RAM, SRAM, DRAM, 판독 전용 메모리(ROM), 프로그래밍가능한 ROM, 전자적으로 프로그래밍가능한 ROM(EPROM), 전자적으로 소거가능한 프로그래밍가능한 ROM(EEPROM), 플래시 메모리 등)을 포함하는 제어 메모리(220)(예를 들어, 휘발성 또는 비-휘발성 메모리)에 더 연결된다. 나아가, 메모리(220)를 구현하는 것은 단일 메모리 디바이스 또는, 대안적으로, 공유된 또는 공통 메모리를 형성하도록 서로 연결된 2개 이상의 메모리 회로와 같은 여러 가능한 실시예를 포함할 수 있다. 더 나아가, 메모리는 더 큰 회로에서 버스 통신 회로의 일부와 같은 다른 회로에 포함될 수 있다.

[0032] 사용자 인터페이스 요소 및 디스플레이 디바이스와 인터페이스하는 것에 더하여, 셋탑 박스(200)는 핸드헬드 인터페이스(222)를 통해 태블릿과 같은 핸드헬드 디바이스와 인터페이스할 수 있다. 이 핸드헬드 디바이스는 추가적인 제어부를 갖는 디스플레이 스크린을 포함하거나 또는 터치 스크린을 포함할 수 있다. 비디오 프로세서(210)로부터 오는 비디오 신호와, 예를 들어 온스크린 디스플레이 메시지 및 메시지 프롬프트 리턴과 같은 다른 데이터는 제어기(214)와 핸드헬드 인터페이스(222) 사이에 라우팅될 수 있다. 핸드헬드 인터페이스(222)는 무선 주파수 통신 링크, 예를 들어, Wi-Fi, 블루투스(Bluetooth), 또는 IEEE(Institute of Electrical and Electronics Engineers) 표준 802.11을 사용하여 핸드헬드 디바이스 또는 태블릿과 신호와 데이터를 송수신할 수 있다. 핸드헬드 인터페이스(222)는 대안적으로 적외선 인터페이스를 사용하여 핸드헬드 디바이스 또는 태블릿과 신호와 데이터를 송수신할 수 있다.

[0033] 동작시, 셋탑 박스(200)는 아래에 더 상세히 설명되는 바와 같이 셋탑 박스 또는 홈 게이트웨이와 같은 클라이언트 디바이스에서 미디어 콘텐츠 데이터베이스를 업데이트하고 관리하고 서치(search)하는 공정을 구현한다. 알고리즘 또는 기능의 물리적인 구현은 비디오 프로세서(210)와 관련된 이산 회로와 같은 하드웨어, 또는 소프트웨어, 예를 들어 제어 메모리(220)에 상주하고 제어기(214)에 의해 판독되어 실행되는 소프트웨어에서 수행될 수 있다. 본 방법은, 미디어 콘텐츠와 연관된 이벤트 데이터를 수신하는 단계로서, 이 이벤트 데이터는 고유한 이벤트 식별자를 포함하는 것인, 상기 수신하는 단계, 이 이벤트 식별자를 검사(examining)하여 상기 이벤트 데이터를 캐시 메모리 내 어레이에 저장하는 단계, 상기 수신된 이벤트 데이터 내 이벤트 식별자가 어레이에 현재 있지 않다면 이벤트 카운트 값을 업데이트하는 단계, 및 상기 캐시 메모리의 스왑(swap)이 트리거된 경우, 예를 들어 이벤트 카운트 값의 적어도 하나가 미리 결정된 임계값을 초과하고 시간 기간 값이 미리 결정된 임계값을 초과하는 경우, 어레이 내 이벤트 데이터로 제1 이벤트 데이터베이스를 업데이트하는 단계를 포함한다.

[0034] 미디어 저장 및 인터페이스 디바이스로 사용되는 셋탑 박스(200)와 같은 디바이스의 동작 측면은 일반적으로 미디어 콘텐츠뿐만 아니라 콘텐츠를 식별하는 데이터베이스 엔트리 또는 레코드를 저장, 유지, 서치, 검색하는 것을 포함한다. 데이터베이스 정보 및 엔트리를 업데이트하는 것은 주기적인 또는 연속적인 콘텐츠 정보 업데이트를 포함할 수 있는 동적 시스템에서 중요하다. 예를 들어, 새로운 데이터베이스 레코드 또는 정보는 수신되어 데이터베이스에 제공될 수 있다. 추가적으로, 콘텐츠와 연관된 현재 저장된 콘텐츠 및 엔트리 또는 정보는 제거되거나 또는 소거될 수 있다. 콘텐츠는 사용자 제어로부터 디바이스에 대한 입력을 통해 주기적으로 또는 대안적으로 제거되거나 소거될 수 있다. 동시에, 또는 거의 동시에, 데이터베이스 서치 요청과 같은 요청이 이루어질 수 있다. 일반적으로, 콘텐츠 및 정보의 모든 분류와 형태는 다수의 소스 및 입력으로부터 데이터베이스로 수신될 수 있다. 안정적인 데이터베이스가 서치 요청에 이용가능한 것이 중요하다. 데이터 및 정보는 서치 능력을 계속 유지하고 가능한 한 높이 서치 결과와 서치 성능을 유지하면서 데이터베이스에 추가될 필요가 있다. 데이터베이스에서 업데이트와 서치를 개선하는 것이 간단한 데이터베이스 구조(예를 들어, 구조화된 질문 언어

(structured query language: SQL) 데이터베이스로 구현된 구조)에서 훨씬 더 바람직하다.

- [0035] 본 명세서에 설명된 바와 같이 미디어 콘텐츠 데이터베이스를 업데이트하고 관리하고 서치하는 기능과 제어는 게이트웨이 디바이스(예를 들어, 셋탑 박스(200))와 연관된 동작 코드 또는 펌웨어의 일부로 포함될 수 있다. 공정은 프로그래밍 언어(예를 들어, 자바(Java) 또는 HTML(hypertext markup language))로 기록된 동작 명령을 포함할 수 있다. 애플리케이션은 (예를 들어, 서버 또는 인터넷 사이트로부터) 미리 로딩되거나 다운로드될 수 있고, 호스트 디바이스의 메모리에 저장될 수 있다. 일 실시예에서 명령은 도 2의 제어 메모리(220)에 저장되고 여기서 명령이 검색되고 제어기(214)에 의해 실행될 수 있다. 다른 실시예에서, 메모리 및 이에 대응하여 처리를 수행하는 프로세서 또는 제어기는 별개의 독립형 집적 회로(예를 들어, 디지털 처리 프로세서(DSP) 또는 애플리케이션 특정 집적 회로(ASIC))로 집적될 수 있다.
- [0036] 도 3은 본 발명의 일 실시예에 따라 본 발명의 원리를 적용할 수 있는 예시적인 레코딩 파이프라인(300)을 도시한다. 일 실시예에서, 레코딩 파이프라인(300)은 도 2의 셋탑 박스(200) 내에 포함된다. 그러나, 전술한 바와 같이, 레코딩 파이프라인(300)은 독립형 DVR, 홈 미디어 서버, 컴퓨터 미디어 스테이션, 태블릿 디바이스, 홈 네트워크 게이트웨이, 멀티-미디어 플레이어, 홈 네트워킹 기기, 등을 포함하지만 이로 제한되지 않는 다른 디바이스에 포함될 수 있다.
- [0037] 레코딩 파이프라인(300)은 튜너(301), 디멀티플렉서(demux)(302), 세그먼트(303), 세그먼트 저장 디바이스(304), 세그먼트 데이터베이스(305), 및 레코딩 제어 디바이스(306)를 포함한다.
- [0038] 방송 비디오는 튜너(301)에 의해 수신된다. 튜너(301)는 네트워크 인터페이스(미도시)를 통해 트랜스포트 스트림을 수신하는 하드웨어 튜너, 예를 들어, DVB(digital video broadcasting) 또는 ATSC(advanced television systems committee) 튜너 또는 IPTV(Internet protocol television) 튜너일 수 있다. 튜너(301)에 의해 수신된 트랜스포트 스트림은 디멀티플렉서(302)에 송신된다.
- [0039] 멀티-프로그램 트랜스포트 스트림의 경우에, 디멀티플렉서(302)는 원하는 프로그램을 선택하고, 인덱스 정보와 함께 단일 프로그램 트랜스포트 스트림을 세그먼트(303)에 송신할 수 있다. 단일 프로그램 트랜스포트 스트림의 경우에, 디멀티플렉서(302)는 수신된 트랜스포트 스트림을 변경 없이 세그먼트(303)에 투명하게 송신할 수 있지만, 연관된 스트림에 대한 인덱스 정보를 추가적으로 제공할 수 있다. 두 경우에, 인덱스 정보는 스트림 내 액세스 포인트의 리스트와 그 연관된 프리젠테이션 시간 스탬프(presentation time stamp: PTS)를 포함한다. 액세스 포인트는 디코딩이 시작할 수 있는 스트림 내 위치를 표시한다.
- [0040] 세그먼트(303)는 트랜스포트 스트림 데이터 및 인덱스 데이터를 이산 세그먼트들로 분해(break)하고 이 데이터를 세그먼트 저장 디바이스(304)에 기록한다. 세그먼트 저장 디바이스(304)는 일반적으로 파일 시스템을 갖는 하드 디스크일 수 있다. 물론, 세그먼트 저장 디바이스(304)는 본 명세서에 제공된 본 발명의 원리의 개시 내용을 통해 이 기술 분야에 통상의 지식을 가진 자에 의해 용이하게 이해할 수 있는 바와 같이 파일 시스템을 갖는 상이한 유형의 메모리 디바이스에서 구현될 수 있다.
- [0041] 각 트랜스포트 세그먼트 및 인덱스 세그먼트는 파일 시스템에 고유한 파일로서 저장된다. 세그먼트의 사이즈는 특정 시스템에 대한 성능 고려사항에 기초하여 조절될 수 있으나, 일반적인 세그먼트 사이즈는 대략 1 내지 5분의 데이터를 포함할 수 있다. 각 세그먼트는 기록되므로, 세그먼트(303)는 자기가 생성하는 세그먼트에 관한 정보로 세그먼트 데이터베이스(305)를 업데이트한다.
- [0042] 레코딩 제어 디바이스(306)는 튜너(301), 디멀티플렉서(302), 및 세그먼트(303)와 상호작용하며 레코딩을 제어한다. 레코딩 제어 디바이스(306)는 튜너(301)를 제어하거나 및/또는 다른 방식으로 지시하여 레코딩될 서비스에 튜닝한다. 레코딩 제어 디바이스(306)는 디멀티플렉서(302)에 트랜스포트 스트림에서 어느 프로그램이 레코딩할지를 제어하거나 및/또는 지시하고, 세그먼트(303)에 레코딩된 세그먼트와 연관된 플레이리스트의 리스트를 제공한다.
- [0043] 도 4는 본 발명의 일 실시예에 따라 본 발명의 원리를 적용할 수 있는 예시적인 플레이백 파이프라인(300)을 도시한다. 플레이백 파이프라인(400)은 레코딩된 세그먼트를 플레이백한다. 일 실시예에서, 플레이백 파이프라인(400)은 도 2의 셋탑 박스(200) 내에 포함된다. 그러나, 전술한 바와 같이, 플레이백 파이프라인(400)은 독립형 DVR, 홈 미디어 서버, 컴퓨터 미디어 스테이션, 태블릿 디바이스, 홈 네트워크 게이트웨이, 멀티-미디어 플레이어, 홈 네트워킹 기기 등을 포함하지만 이들로 제한되지 않는 다른 디바이스에 포함될 수 있다.
- [0044] 플레이백 파이프라인(400)은 세그먼트 데이터베이스(401), 세그먼트 저장 디바이스(402), 디세그먼트(403), 디멀티플렉서(404), 플레이백 제어 디바이스(405), 자막 디코더(406), 비디오 디코더(407), 오디오 디코더(408),

오디오 렌더링 디바이스(209), 및 비디오 렌더링 디바이스(210)를 포함한다.

- [0045] 플레이백 제어 디바이스(405)는 플레이되어야 플레이리스트를 디세그먼트(403)에 제어하거나 및/또는 지시한다. 플레이백 제어 디바이스(405)는 디세그먼트(403)에 지정된 시간 오프셋에서 플레이백을 시작하고 지정된 속도에서 플레이백할 것을 제어하거나 및/또는 지시한다. 예를 들어, 1.0의 플레이백 속도는 정상 플레이백에 대응하고, 4.0는 4x 고속 전진이며, -4.0는 4x 되감기이고, 0.0는 일시정지일 수 있다.
- [0046] 디세그먼트(403)는 세그먼트 데이터베이스(401)에 질문하여 플레이백에 요구되는 트랜스포트 및 인덱스 세그먼트 파일을 발견한다. 디세그먼트(403)는 세그먼트 저장 디바이스(402)로부터 이들 파일을 판독한다. 인덱스 정보는 디세그먼트(403)에 의해 사용되어 플레이백이 세그먼트에서 일부 시간 오프셋으로 시작하는 경우에 세그먼트에 한정된 오프셋으로 스킵하게 한다. 디세그먼트(403)는 데이터(예를 들어, 트랜스포트 스트림)를 세그먼트 저장 디바이스(402)로부터 디멀티플렉서(404)로 송신한다.
- [0047] 디멀티플렉서(404)는 트랜스포트 스트림을 오디오, 비디오, 및 자막 기본 스트림으로 분할하고 이들 스트림을 오디오 디코더(408), 비디오 디코더(407), 및 자막 디코더(406)로 각각 송신한다. 오디오 디코더(408)로부터의 출력은 오디오 렌더링 디바이스(409)로 송신되고, 이 오디오 렌더링 디바이스는 오디오를 최종적으로 렌더링하는 일을 한다. 비디오 디코더(407) 및 자막 디코더(406)로부터 오는 출력은 비디오 렌더링 디바이스(410)로 송신되고 이 비디오 렌더링 디바이스는 비디오를 최종적으로 렌더링하는 일을 한다.
- [0048] 도 5는 본 발명의 일 실시예에 따라 세그먼트 데이터베이스(예를 들어, 세그먼트 데이터베이스(305) 및 세그먼트 데이터베이스(401))에 존재하는 4개의 예시적인 레코드 유형(500)을 도시한다. 이 레코드 유형은 다음 사항: 플레이리스트 레코드(501); 플레이리스트세그먼트 레코드(502); 세그먼트 레코드(503); 및 세그먼트삭제 레코드(504)를 포함한다. 각 플레이리스트 레코드(501)는 레코딩을 나타낸다. 각 세그먼트 레코드(403)는 디스크에 저장된 레코딩된 세그먼트를 나타낸다. 각 플레이리스트세그먼트 레코드(402)는 세그먼트 레코드(503)를 플레이리스트 레코드(501)와 연관시킨다. 최종적으로, 세그먼트삭제 레코드(504)는 삭제 표시된 레코딩된 세그먼트를 추적하는데 사용된다.
- [0049] 이제 본 발명의 일 실시예에 따라 본 발명의 원리를 적용할 수 있는 일부 예시적인 시나리오에 관한 설명이 제공된다.
- [0050] 사용자가 라이브 방송 콘텐츠를 시청하며 레코딩을 시작하기로 결정하면, 사용자가 종종 이벤트의 시작 후까지 레코딩 요청을 하지 않았더라도 이용가능한 시간-시프트 데이터를 사용하여 이벤트의 전체 레코딩을 생성하는 것이 바람직하다. 일례로서, 사용자는 8:00pm에서 시작하는 이벤트를 시청하고 있다고 가정하자. 8:10pm에서, 사용자는 이벤트를 레코딩할 것을 요청한다. 시간-시프트 버퍼가 8:00pm 내지 8:10pm의 콘텐츠를 포함하는 경우, 전체 이벤트의 전체 레코딩을 생성하는 것이 가능하여야 한다. 본 발명은 새로운 레코딩을 형성하기 위해 시간-시프트 데이터를 복사할 필요 없이 시간-시프트 데이터를 레코딩으로 변환하는 수단을 제공한다.
- [0051] 일부 경우에 다수의 레코딩이 동일한 콘텐츠의 일부를 공유하는 것이 가능하다. 예를 들어, 이벤트 A가 8:00PM에서 시작해서 8:30PM에서 종료하고 동일한 채널에서 이벤트 B는 8:30PM에서 시작해서 9:00PM에서 종료하는 것으로 가정하자. 사용자는 이벤트 A 및 이벤트 B를 모두 레코딩할 것을 요청할 수 있다. 나아가, 사용자의 레코딩 선호도는 모든 레코딩이 5 분 일찍 시작해서 5 분 늦게 종료하여야 하는 것을 지시할 수 있다. 이 시나리오에서 레코딩 A의 마지막 10 분이 레코딩 B의 처음 10분과 동일한 콘텐츠를 포함하는 중첩이 있다. 본 발명의 원리는 공유된 데이터를 복제하는 대신에 레코딩의 중첩 부분을 하나의 위치에 저장하고 2개의 레코딩에서 공유하게 하는 것에 의해 이 중첩 레코딩의 저장을 최적화하는 방식을 제공한다.
- [0052] 대부분의 DVR에서, 사용자는 레코딩할 특정 이벤트를 선택하거나, 또는 시간 범위 및 채널에 기초하여 레코딩을 스케줄링할 수 있다. 사용자가 시간 및 채널에 의해 스케줄링할 때, 대부분의 DVR는 "채널 99에서 8pm - 10pm에서 수동 레코딩"과 같은 것을 말하는 일반 엔트리로 레코딩을 보여준다. 레코딩은 다수의 이벤트를 포함할 수 있으나, 레코딩이 무엇을 포함하는지를 아는 방법은 없다. 특정 이벤트를 픽아웃(pick out)을 하고 이를 플레이하거나, 또는 이벤트를 레코딩에 유지하고 나머지를 삭제하는 방법은 없다. 예를 들어, 전송된 수동 레코딩은 사용자가 8:30pm-9pm에서 유지하기를 원하지만, 사용자가 나머지 부분은 삭제하기를 원하는 프로그램 A의 에피소드를 포함할 수 있다. 본 발명의 원리는 이 문제에 대한 해법을 제공한다.
- [0053] 이제 본 발명의 일 실시예에 따라 레코딩을 생성하는 것에 관한 설명이 제공된다.
- [0054] 레코딩이 시작될 때, 제일 먼저 새로운 레코딩을 나타내는 새로운 플레이리스트 레코드(501)가 세그먼트 데이터베이스에 추가된다. 플레이리스트 레코드에는 새로운 고유한 플레이리스트ID가 할당되고 시간시프트용 필드()는

이 플레이리스트가 정상 레코딩을 위한 것인지 또는 시간-시프트를 위한 것인지를 나타내도록 설정된다.

- [0055] 플레이리스트가 생성된 후, 레코딩 제어는 레코딩을 시작할 레코딩 파이프라인(300)을 구성한다. 세그먼트(303)는 세그먼트 데이터베이스(305)에서 새로운 세그먼트 레코드(503) 및 플레이리스트세그먼트 레코드(502)를 생성하고 제1 트랜스포트 파일 및 인덱스 파일을 디스크에 기록하기 시작한다.
- [0056] 세그먼트 레코드(503)는 다음 값으로 생성된다:
- [0057] 세그먼트ID - 세그먼트에 대한 고유한 ID
- [0058] 트랜스포트파일 - 이 세그먼트에 대한 트랜스포트 스트림 데이터를 저장하는데 사용된 파일
- [0059] 인덱스파일 - 이 세그먼트에 대해 인덱스 정보를 저장하는데 사용된 파일
- [0060] 레퍼런스카운트(ReferenceCount) - 이것은 이 세그먼트를 참조하는 플레이리스트세그먼트 레코드의 번호의 카운트이다. 현재 레코딩이 하나의 플레이리스트에만 적용되는 경우, 이것은 1로 설정된다.
- [0061] 서비스/프로그램ID - 이것은 레코딩되는 서비스(채널)를 식별한다.
- [0062] 시작시간 - UTC에서 세그먼트의 시작 시간
- [0063] 종료시간 - 이것은 세그먼트의 종료 시간이지만, 세그먼트가 완료될 때까지는 미지이다. 초기에 이 값은 0으로 설정된다.
- [0064] 시작PTS - 이것은 세그먼트에서 제1 PTS이다. 제1 PTS가 세그먼트 생성시에 미지인 경우, 이 값은 초기에 0으로 설정된다.
- [0065] 종료PTS - 이것은 세그먼트에 대한 종료PTS 값이지만, 이것은 초기에는 미지이고 0으로 설정된다.
- [0066] 불연속성 - 이것은 세그먼트 내에 PTS 불연속성이 존재하는 것을 나타내는 것을 플래그이다. 이것은 초기에 거짓(FALSE)으로 설정된다.
- [0067] 플레이리스트세그먼트 레코드(502)는 다음 값으로 생성된다:
- [0068] 플레이리스트ID - 이 세그먼트와 연관된 플레이리스트 레코드에 대한 플레이리스트ID
- [0069] 세그먼트ID - 세그먼트 레코드의 세그먼트ID
- [0070] 시퀀스번호 - 이것은 플레이리스트에 추가된 각 세그먼트를 증분하는 정수이다. 초기 값은 0이다.
- [0071] 시작오프셋 - 이것은 초기에 0으로 설정된다.
- [0072] 종료오프셋 - 이것은 초기에 0으로 설정된다.
- [0073] 레코딩할 때, 플레이리스트세그먼트 레코드(502)는 각 세그먼트에 대한 각 플레이리스트에 대해 생성된다. 따라서, 데이터가 3개의 플레이리스트에 레코딩되고 있었다면, 각 세그먼트 레코드는 3개의 플레이리스트세그먼트 레코드(502)를 구비할 수 있다. 또한, 각 세그먼트 레코드는 3으로 설정된 레퍼런스 카운트를 구비할 수 있는데, 그 이유는 세그먼트를 참조하는 3개의 플레이리스트세그먼트 레코드(502)가 있기 때문이다.
- [0074] 일반적인 세그먼트 사이즈는 약 1 내지 5 분의 데이터를 포함할 수 있다. 각 세그먼트가 완료된 후, 세그먼트(303)는 세그먼트에 대한 최종 데이터로 세그먼트 레코드(503)를 업데이트한다. 이것은 종료시간, 종료PTS, 불연속성 플래그, 및 레코딩의 시작시에 알려져 있지 않았다면 시작PTS를 포함한다. 세그먼트(303)는 그 다음 세그먼트를 기록하기 시작하여, 새로운 세그먼트 레코드(503) 및 플레이리스트세그먼트 레코드(502)를 생성한다. 이 공정은 레코딩이 완료될 때까지 연속한다.
- [0075] 도 6은 본 발명의 일 실시예에 따라 레코딩을 시작하는 예시적인 방법(600)을 도시한다. 단계(610)에서, 플레이리스트가 생성된다. 단계(620)에서, 레코딩 파이프라인이 이미 원하는 서비스에 대해 존재하는지 여부가 결정된다. 만약 그렇다면, 방법(600)은 단계(615)로 진행한다. 그렇지 않다면, 방법(600)은 단계(660)로 진행한다. 단계(615)에서, 플레이리스트는 기존의 레코딩 파이프라인의 세그먼트(303)에 추가된다.
- [0076] 단계(620)에서, 세그먼트(303)가 세그먼트의 중간에 있는지 여부가 결정된다. 만약 그렇다면, 방법(600)은 단계(625)로 진행한다. 그렇지 않다면, 방법(600)은 단계(645)로 진행한다.
- [0077] 단계(625)에서, 세그먼트(303)는 현재 위치에 대해 조절된 시작 오프셋을 갖는 새로운 플레이리스트에 대한 플

레이리스트세그먼트 레코드(502)를 생성한다.

- [0078] 단계(630)에서, 세그먼트에 대한 레퍼런스 카운트가 증분된다.
- [0079] 단계(635)에서, 세그먼트가 완료되었는지 여부가 결정된다. 만약 그렇다면, 방법(600)은 단계(640)로 진행한다. 그렇지 않다면, 방법(600)은 단계(650)로 진행한다.
- [0080] 단계(640)에서, 세그먼트 레코드(503)는 최종 시간스탬프로 업데이트된다.
- [0081] 단계(645)에서, 세그먼트(303)는 (새로운 세그먼트에 대해) 새로운 세그먼트 및 인덱스 파일을 기록하고, 세그먼트 레코드(503)를 생성하며, 각 플레이리스트에 대한 플레이리스트세그먼트 레코드(502)를 생성한다.
- [0082] 단계(660)에서, 레코딩 파이프라인이 생성된다.
- [0083] 단계(650)에서, 레코딩이 완료되었는지 여부가 결정된다. 만약 그렇다면, 방법(600)은 단계(655)로 진행한다. 그렇지 않다면, 방법(600)은 단계(635)로 리턴한다.
- [0084] 단계(655)에서, 레코딩은 정지한다(도 7 참조).
- [0085] 도 7은 본 발명의 일 실시예에 따라 레코딩을 정지하는 예시적인 방법(700)을 도시한다. 단계(705)에서, 플레이리스트가 세그먼트(303)로부터 제거된다. 단계(710)에서, 세그먼트(303)가 세그먼트의 중간에 있는지 여부가 결정된다. 만약 그렇다면, 방법(700)은 단계(715)로 진행한다. 그렇지 않다면, 방법(700)은 단계(720)로 진행한다.
- [0086] 단계(715)에서, 마지막 플레이리스트세그먼트 레코드(502)에 대한 종료 오프셋이 조절된다.
- [0087] 단계(720)에서, 세그먼트가 하나 이상의 플레이리스트를 포함하는지 여부가 결정된다. 만약 그렇다면, 방법(700)은 종료한다. 그렇지 않다면, 방법(700)은 단계(725)로 진행한다.
- [0088] 단계(725)에서, 레코딩 파이프라인은 분해(tear down)된다.
- [0089] 도 8은 본 발명의 일 실시예에 따라 레코딩을 삭제하는 예시적인 방법(800)을 도시한다. 단계(805)에서, 플레이리스트 세그먼트 레코드(502)가 질문되고, 여기서 플레이리스트ID = 세그먼트의 리스트를 형성하도록 삭제될 플레이리스트이다. 단계(810)에서, 방법(800)은 리스트 내 제1 세그먼트에서 시작한다. 단계(815)에서, 세그먼트 레코드(503)에 대한 레퍼런스 카운트가 감분된다. 단계(820)에서, 레퍼런스 카운트 = 0인지 여부가 결정된다. 만약 그렇다면, 방법(800)은 단계(825)로 진행한다. 그렇지 않다면, 방법(800)은 단계(840)로 진행한다.
- [0090] 단계(825)에서, 세그먼트삭제 레코드(504)가 세그먼트ID로 생성된다.
- [0091] 단계(830)에서, 세그먼트가 더 있는지 여부가 결정된다. 만약 그렇다면, 방법(800)은 단계(835)로 진행한다. 그렇지 않다면, 방법(800)은 단계(845)로 진행한다.
- [0092] 단계(835)에서, 방법(800)은 리스트에서 그 다음 세그먼트로 이동한다.
- [0093] 단계(845)에서, 가비지 수집이 트리거된다.
- [0094] 도 9는 본 발명의 일 실시예에 따라 세그먼트를 가비지 수집하는 예시적인 방법(900)을 도시한다. 단계(905)에서, 세그먼트삭제 레코드(504)가 있는지 여부가 결정된다. 만약 그렇다면, 방법(900)은 단계(910)로 진행한다. 그렇지 않다면, 방법(900)은 종료한다. 단계(910)에서, 트랜스포트 및 인덱스 파일이 삭제된다. 단계(915)에서, 세그먼트 레코드(503)가 삭제된다. 단계(920)에서, 세그먼트삭제 레코드(504)가 삭제된다.
- [0095] 도 10은 본 발명의 일 실시예에 따라 레코딩의 시작부로부터 데이터를 절단하는 예시적인 방법(1000)을 도시한다. 단계(1005)에서, 최소 세그먼트 레코드(503)를 갖는 플레이리스트세그먼트 레코드(502)가 질문되고, 여기서 플레이리스트ID = 세그먼트 레코드를 취득하도록 절단될 플레이리스트이다. 단계(1010)에서, 현재 시간 - 종료 시간 \geq 시간-시프트 길이인지 여부가 결정된다. 만약 그렇다면, 방법(1000)은 단계 1015로 진행한다. 그렇지 않다면, 방법은 단계(1040)로 진행한다.
- [0096] 단계 1015에서, 세그먼트 레코드(503)에 대한 레퍼런스 카운트가 감분된다. 단계(1020)에서, 레퍼런스 카운트 = 0인지 여부가 결정된다. 만약 그렇다면, 방법(1000)은 단계(1025)로 진행한다. 그렇지 않다면, 방법(1000)은 단계(1035)로 진행한다.
- [0097] 단계(1025)에서, 세그먼트삭제 레코드(504)는 세그먼트ID로 생성된다. 단계(1030)에서, 가비지 수집이 트리거된

다.

- [0098] 단계(1035)에서, 세그먼트 레코드(503)가 삭제된다.
- [0099] 단계(1040)에서, 지연 = 시간 - 시프트 길이 - (현재 시간 - 종료시간) 만큼 지연시킨다.
- [0100] 도 11은 본 발명의 일 실시예에 따라 레코딩의 시작부에 세그먼트를 부착하는 예시적인 방법(1100)을 도시한다. 방법(1100)은 시간-시프트 데이터를 레코딩의 시작부에 프리펜드(prepend)하는데 사용될 수 있다. 단계(1105)에서, 레코딩이 시작된다(도 6 참조). 단계(1110)에서, 세그먼트 레코드(503)가 질문되고, 여기서 서비스 = 레코딩하고 있는 서비스이고, 시작시간 ≤ 원하는 시작시간, 및 종료시간 > 시작시간 내림차순으로 분류된 원하는 시작 시간이다. 단계(1115)에서, 세그먼트에 시간 갭이 있는지 여부가 결정된다. 만약 그렇다면, 방법(1100)은 단계(1120)로 진행한다. 그렇지 않다면, 방법은 단계(1125)로 진행한다.
- [0101] 단계(1120)에서, 갭에 절단이 수행되어 리스트에 갭이 더 이상 없다.
- [0102] 단계(1125)에서, 방법(1100)은 시퀀스번호 = 0으로 리스트 내 제2 세그먼트에서 시작한다.
- [0103] 단계(1130)에서, 플레이리스트 세그먼트 레코드(502)가 생성되고, 레퍼런스 카운트가 증분된다. 단계(1135)에서, 리스트에 세그먼트가 더 있는지 여부가 결정된다. 만약 그렇다면, 방법(1100)은 단계(1140)로 진행한다. 그렇지 않다면, 방법(1100)은 종료된다. 단계(1140)에서, 방법(1100)은 그 다음 세그먼트로 이동하고, 시퀀스번호를 감분한다.
- [0104] 도 12는 본 발명의 일 실시예에 따라 레코딩을 플레이백하는 예시적인 방법(1200)을 도시한다. 단계(1205)에서, 최소 시퀀스번호를 갖는 플레이리스트 세그먼트 레코드(502)가 질문되고, 여기서 플레이리스트ID = 플레이할 플레이리스트이다. 단계(1210)에서, 시작오프셋 = 0인지 여부가 결정된다. 만약 그렇다면, 방법(1200)은 단계(1215)로 진행한다. 그렇지 않다면, 방법(1200)은 단계(1250)로 진행한다.
- [0105] 단계(1215)에서, 시작 바이트 위치는 0으로 설정된다.
- [0106] 단계(1220)에서, 종료오프셋 = 0인지 여부가 결정된다. 만약 그렇다면, 방법(1200)은 단계(1225)로 진행한다. 그렇지 않다면, 방법(1200)은 단계(1255)로 진행한다.
- [0107] 단계(1225)에서, 종료 바이트 위치는 트랜스포트 파일의 사이즈와 동일하게 설정된다.
- [0108] 단계(1230)에서, 트랜스포트 파일은 시작 바이트 위치로부터 종료 바이트 위치로 판독되고 데이터는 디멀티플렉서(404)로 송신된다. 단계(1235)에서, 시퀀스번호가 증분된다. 단계(1240)에서, 플레이리스트ID = 플레이할 플레이리스트이고 시퀀스 번호를 가지는 플레이리스트 세그먼트가 존재하는지 여부가 결정된다. 만약 그렇다면, 방법(1200)은 단계(1245)로 진행한다. 그렇지 않다면, 방법(1200)은 단계(1260)로 진행한다.
- [0109] 단계(1245)에서, 플레이리스트 세그먼트 및 세그먼트가 검색된다.
- [0110] 단계(1250)에서, 인덱스 파일이 판독되어 시작오프셋에 대한 시작 바이트 위치를 발견한다.
- [0111] 단계(1255)에서, 인덱스 파일이 판독되어 종료오프셋에 대한 종료 바이트 위치를 발견한다.
- [0112] 단계(1260)에서, 플레이백이 완료된다.
- [0113] 이제 본 발명의 일 실시예에 따라 레코딩 동작 동안 플레이리스트를 추가/제거하는 것에 관한 설명이 제공된다.
- [0114] 동일한 데이터가 다수의 플레이리스트에 사용되고 있는 경우에, 레코딩 동작 동안 세그먼트(303)로부터 플레이리스트를 추가 또는 제거하는 것이 종종 필요하다. 플레이리스트세그먼트 레코드(502)에서 시작오프셋 및 종료오프셋 필드는 이 목적을 위해 사용된다. 본 시스템이 5 분의 세그먼트를 사용하는 것으로 가정한다. 나아가, 제2 플레이리스트가 레코딩에 추가되는 세그먼트가 2분이라고 가정한다. 이 점에서 새로운 플레이리스트세그먼트 레코드(502)가 생성되고, 이 세그먼트 레코드의 레퍼런스 카운트가 증분되고, 플레이리스트세그먼트에서 필드는 시작오프셋 필드를 제외하고는 이전과 같이 채워진다. 시작오프셋 필드는 120 초로 설정되어 이 세그먼트에 대한 플레이백이 세그먼트에서 120초의 오프셋에서 시작하는 것을 나타낸다. 마찬가지로, 데이터가 다수의 플레이리스트에 레코딩되고 있고 레코딩이 단일 플레이리스트에 대해 중지된 경우, 종료오프셋 필드가 플레이백이 중지하는 세그먼트의 종료로부터 오프셋으로 설정된다.
- [0115] 이제 본 발명의 일 실시예에 따라 레코딩을 삭제하는 것에 관한 설명이 제공된다. 이 공정은 또한 도 8에 대해 전술되어 있다.

- [0116] 각 레코딩은 플레이리스트에 대응한다. 레코딩이 삭제되면, 다음 액션이 세그먼트 데이터베이스에 수행된다:
- [0117] 1. 레코딩을 위한 플레이리스트 레코드(501)가 삭제된다.
- [0118] 2. 삭제된 플레이리스트ID를 참조하는 모든 플레이리스트세그먼트 레코드(502)가 삭제된다.
- [0119] 3. 각 삭제된 플레이리스트세그먼트 레코드(502)에 대해, 삭제된 플레이리스트세그먼트 레코드(502)에서 세그먼트ID에 대응하는 세그먼트 레코드(503)가 1씩 감분된 레퍼런스카운트 필드를 구비한다.
- [0120] 4. 레퍼런스카운트가 이제 0인 각 세그먼트 레코드(503)에 대해, 새로운 세그먼트삭제 레코드(504)가 생성되고, 여기서 세그먼트ID는 0 레퍼런스카운트를 가지는 세그먼트로 설정된다.
- [0121] 위 4개의 동작은 데이터베이스에 원자적(atomically)으로 수행되어 동작 동안 전력 단전이 일어나는 경우, 데이터베이스는 일관된 상태에 유지될 수 있다.
- [0122] 위 도 9에 대해 설명된 바와 같이, 배경 공정은 세그먼트삭제 레코드(504)에 주기적으로 질문하고, 대응하는 세그먼트 레코드(503)를 룩업(look up)하며, 인덱스파일 및 트랜스포트파일을 삭제하고, 세그먼트 레코드(503) 및 세그먼트삭제 레코드(504)를 제거한다. 이것은 삭제 직후에 수행되거나, 또는 이 동작은 디스크 공간이 필요할 때까지 지연될 수 있다.
- [0123] 이제 본 발명의 일 실시예에 따라 시간-시프트를 위한 플레이리스트를 생성하는 것에 관한 설명이 제공된다.
- [0124] 시간-시프트를 위한 플레이리스트를 생성하는 것은 2개의 예외를 제외하고는 레코딩할 플레이리스트를 생성하는 것과 동일하다:
- [0125] 1. 플레이리스트 레코드에서 시간시프트용 플래그는 참(TRUE)으로 설정된다. 셋탑 박스(200)가 부팅될 때마다, 참으로 설정된 시간시프트용 필드를 갖는 모든 플레이리스트는 시간-시프트 플레이리스트가 리셋을 통해 보존될 필요가 없어서 삭제된다.
- [0126] 2. 레코딩의 시작부로부터 플레이리스트세그먼트가 주기적으로 삭제된다.
- [0127] 이 공정은 전체 플레이리스트를 삭제하는 것과 동일한 공정을 따르고 플레이리스트 레코드만이 삭제되지 않고 지정된 플레이리스트세그먼트 레코드(502)만이 제거된다. 플레이리스트세그먼트를 주기적으로 삭제하면 시간-시프트 버퍼의 사이즈를 한정할 수 있다. 예를 들어, 30 분의 시간-시프트 버퍼를 유지하기를 원하고 세그먼트 사이즈가 5 분이었다면, 플레이리스트에 대해 6개 이하의 완료 세그먼트를 유지할 수 있다. 7번째 세그먼트가 완료될 때, 제1 플레이리스트세그먼트를 삭제하여 플레이리스트에 대한 세그먼트의 총 번호를 다시 6으로 되가게 한다.
- [0128] 이제 본 발명의 일 실시예에 따라 시간-시프트 데이터를 레코딩으로 변환하는 것에 관한 설명이 제공된다. 이 공정은 또한 도 10에 대해 전술되어 있다.
- [0129] 레코딩이 이미 진행중인 이벤트에 요청되면, 이벤트에 대한 시간 범위와 서비스와 매칭하는 세그먼트 레코드(503)에 대해 세그먼트 데이터베이스에 질문된다. 이것은 시작시간, 종료시간, 및 서비스/프로그램ID 필드에 질문하는 것에 의해 수행된다. 요청된 이벤트의 서비스 및 시간 범위에 대응하는 세그먼트가 발견되면, 새로운 플레이리스트세그먼트 레코드(502)가 이들 세그먼트 각각에 대해 생성되어 시퀀스 번호에 삽입되어 레코딩의 시작부에 적절한 시퀀스로 나타난다. 제1 세그먼트의 시작오프셋은 이벤트에 대한 시작 시간과 매칭하는데 필요에 따라 조절된다. 나아가, 각 추가된 세그먼트에 대해 레퍼런스카운트 필드가 증분된다. 이 공정은 트랜스포트 또는 인덱스 파일을 실제 복사하거나 변경함이 없이 시간-시프트에 사용된 기존의 데이터를 레코딩에 추가할 수 있게 한다.
- [0130] 이제 본 발명의 일 실시예에 따라 레코딩이나 시간-시프트를 플레이백하는 것에 관한 설명이 제공된다. 이 공정은 또한 도 12에 대해 전술되어 있다.
- [0131] 기본 플레이백은 도 4에 대해 이미 설명되어 있다. 이제 플레이백 동작 동안 디세그먼트(403)에 의해 수행되는 데이터베이스 동작을 보다 상세히 설명한다.
- [0132] 시작부로부터 레코딩을 플레이백하면, 디세그먼트(403)는 최저 시퀀스 번호를 갖는 플레이리스트세그먼트 레코드(502)에 질문한다. 디세그먼트(403)는 대응하는 세그먼트 레코드(503)에 질문하여 트랜스포트파일 및 인덱스파일을 발견한다. 플레이리스트세그먼트 레코드에 대한 시작오프셋이 0인 경우, 플레이백은 트랜스포트 파일의 시작시에 시작한다. 시작오프셋이 비-제로(non-zero)라면, 오프셋은 세그먼트의 시작에 대한 베이스 시간으로

시작시간 및 시작PTS 필드를 사용하여 PTS 오프셋으로 변환된다. 인덱스 파일은 스캐닝되어 지정된 PTS 오프셋에 대응하는 트랜스포트 파일에서 바이트 위치를 발견하여 이 위치로부터 플레이백이 시작한다.

[0133] 플레이리스트에서 지정된 오프셋에 탐색을 수행하기 위해, 디세그먼트터(403)는 플레이리스트세그먼트 레코드(502)에 질문하여 플레이리스트에 대한 세그먼트ID의 리스트를 취득한다. 디세그먼트터(403)는 최저 시퀀스 번호를 갖는 세그먼트 레코드(503)에 질문하여 시작시간을 판독한다. 이것은 플레이리스트에 대한 베이스 시작 시간이다. 디세그먼트터(403)는 지정된 오프셋을 베이스 시간에 추가하여 원하는 위치의 절대 시간을 계산한다. 최종적으로, 디세그먼트터(403)는 플레이리스트의 일부인 세그먼트 레코드(503)에 대한 시작시간 및 종료시간 필드를 비교하는 것에 의해 지정된 시간을 포함하는 세그먼트 레코드(503)를 서지한다. 이것은 원하는 위치를 포함하는 트랜스포트 및 인덱스 파일을 제공한다. 원하는 절대 시간은 세그먼트에 대한 베이스 시간으로서 세그먼트 레코드(503)에 있는 시작시간 및 시작PTS 필드를 사용하고 나서 원하는 PTS 오프셋을 계산하는 것에 의해 PTS 시간 스탬프로 변환된다. 인덱스 파일은 스캐닝되어 원하는 PTS 오프셋에 대한 트랜스포트 파일에서 바이트 오프셋을 발견한다. 디세그먼트터(403)는 트랜스포트 파일에서 이 점으로부터 플레이백을 시작한다.

[0134] 불연속적인 트릭 모드(고속 전진 및 되감기)는 탐색 동작과 동일한 방식으로 수행된다. 예를 들어, 4x 고속 전진을 구현하기 위해 디세그먼트터(403)는 매 0.5 초마다 플레이리스트에서 전진으로 2초를 탐색할 수 있다. 이것은 4x 정상 플레이백의 속도로 플레이리스트를 통해 스캐닝하는 효과를 제공한다.

[0135] 탐색 동작에 요구되는 데이터베이스 동작의 효율성을 보장하기 위해, 데이터베이스(401)는 적절한 인덱스를 제공하여 요구되는 질문을 신속히 실행할 것을 보장하는 것으로 예상된다. 나아가, 디세그먼트터(403)는 성능을 개선하기 위해 데이터베이스(401)로부터 데이터를 캐시(cache)하거나 프리페치(pre-fetch)할 수 있다. 이들 상제는 특정 데이터베이스 구현에 특정된 것이어서 본 발명의 원리의 범위를 넘는다.

[0136] 이제 본 발명의 일 실시예에 따라 레코딩 및 서브 레코딩을 생성하고 관리하는 것에 관한 설명이 제공된다.

[0137] 본 명세서에 설명된 원리는 사용자로 하여금 한정된 시간 범위 동안 채널에서 레코딩을 생성하고 레코딩된 콘텐츠를 연속적인 레코딩으로 및 레코딩된 블록 내에서 이산 이벤트로 관리할 수 있게 한다.

[0138] 사용자가 본 발명의 원리를 사용하여 시간 및 채널에 의하여 레코딩을 스케줄링할 때, 전자 프로그램 가이드(EPG) 데이터를 포함하는 데이터베이스에 질문하여 레코딩에 포함된 이벤트를 결정한다. 시스템은 다수의 레코딩을 생성한다. 부모 레코딩은 사용자에게 의해 요청된 시간 범위를 커버하는 연속적인 레코딩일 수 있다. 부모 레코딩과 병렬로, 시스템은 EPG 데이터에 의해 리포트된 바와 같이 부모 레코딩이 포함하는 각 이벤트에 대한 레코딩을 생성한다. 아래 표 1은 이의 예를 도시한다.

표 1

[0139]

수동 레코딩 - 채널 99에서 8pm 내지 10pm		
프로그램 A	프로그램 B	뉴스 스페셜(New Special)--대통령 선거 토론(Presidential Debate)

[0140] 따라서, 이 예에서, 본 발명의 원리는 사용자가 채널 99에서 8pm 내지 10pm 레코딩만을 요청하였더라도 4개의 레코딩을 생성한다. 본 시스템은 이것을 하기 위해 추가적인 데이터를 실제 레코딩하지 않는다. 본 발명의 원리는 본 명세서에 설명된 세그먼트된 레코딩 아키텍처를 사용한다.

[0141] 레코딩 사이의 부모-자녀 관계로 인해, 수동 레코딩은 트리의 루트 노드인 부모 레코딩에 대해 확장되거나 축소될 수 있는 트리 구조로 사용자에게 제시될 수 있다. 도 13은 본 발명의 일 실시예에 따라 예시적인 레코딩 트리 인덱스(1300)를 도시한 도면이다. 따라서, 상기 예에서, 레코딩은 도 13에 도시된 바와 같이 사용자에게 디스플레이될 수 있다. "수동 레코딩 - 채널 99에서 8PM 내지 10PM"은 트리 인덱스(1300)의 루트(1310)에서 도시된다. "프로그램 A", "프로그램 B", 및 "뉴스 스페셜 - 대통령 선거 토론"은 분기 및/또는 가지(1311, 1312 및 1313)으로 도시된다.

[0142] 사용자는 수동 레코딩을 플레이할 것을 선택하고, 연속적인 블록으로 모든 것을 볼 수 있다. 사용자는 레코딩 내에 프로그램 A만을 플레이하는 것으로 결정하고, 플레이백은 프로그램 A의 에피소드의 시작시에 시작하고 에피소드의 종료시에 종료한다.

[0143] 사용자는 수동 레코딩을 삭제할 수 있으나, 수동 레코딩 내에 포함되게 특정 에피소드를 유지하도록 선택할 수 있다. 예를 들어, 사용자는 프로그램 A의 에피소드는 유지하지만 그 밖의 다른 것은 삭제하기를 원할 수 있다.

실제 레코딩된 비디오 및 오디오 데이터는 부모 레코딩 및 자녀 레코딩 사이에 공유되고 참조 카운트되어, 리스트 내 레코딩에 의해 더 이상 참조되지 않을 때 각 세그먼트의 데이터가 디스크로부터 제거된다.

- [0144] 본 설명은 본 발명의 원리를 예시한다. 따라서, 이 기술 분야에 통상의 지식을 가진 자라면 본 명세서에 명시적으로 기술되거나 도시되지는 않았을지라도 본 발명의 사상과 범위 내에 포함되고 본 발명의 원리를 구현하는 여러 배열을 고안할 수 있을 것이라는 것을 이해할 수 있을 것이다.
- [0145] 본 명세서에 언급된 모든 예시와 조건적 언어들은 이 기술을 개선하려고 발명자(들)가 기여한 본 발명의 원리와 개념을 독자들이 이해하는 것을 돕기 위한 설명을 위한 목적으로 의도된 것이며 그러한 구체적으로 언급된 예시와 조건으로 제한하는 것으로 해석되어서는 아니된다.
- [0146] 나아가, 본 발명의 원리, 측면 및 실시예뿐만 아니라 특정 예시를 언급하는 모든 진술은 구조적 및 기능적으로 균등한 것을 포함하는 것으로 의도된 것이다. 추가적으로, 그러한 균등물은 현재 알려진 균등물뿐만 아니라 미래에 개발될 균등물, 즉 구조에 상관없이 동일한 기능을 수행하는 개발된 임의의 요소를 포함하는 것으로 의도된다.
- [0147] 따라서, 예를 들어, 이 기술 분야에 통상의 지식을 가진 자라면 본 명세서에 제시된 블록도가 본 발명의 원리를 구현하는 예시적인 회로의 개념적인 도식을 제시하는 것임을 이해할 수 있을 것이다. 이와 유사하게, 임의의 흐름도, 흐름 선도, 상태 전이도, 의사코드 등은 컴퓨터나 프로세서가 명시적으로 도시되지 않았을 지라도 컴퓨터로 판독가능한 매체에 실질적으로 제공되고 컴퓨터나 프로세서에 의해 실행될 수 있는 여러 공정을 나타낸다는 것을 이해할 수 있을 것이다.
- [0148] 도면에 도시된 여러 요소의 기능은 적절한 소프트웨어와 연관된 소프트웨어를 실행할 수 있는 하드웨어뿐만 아니라 전용 하드웨어의 사용을 통해 제공될 수 있다. 프로세서에 의해 제공될 때, 그 기능은 단일 전용 프로세서에 의해, 단일 공유 프로세서에 의해 또는 일부가 공유될 수 있는 복수의 개별 프로세서에 의해 제공될 수 있다. 나아가, "프로세서" 또는 "제어기"라는 용어의 명시적인 사용이 소프트웨어를 실행할 수 있는 하드웨어를 배타적으로 말하는 것으로 해석되어서는 안되며, 제한없이 디지털 신호 프로세서("DSP") 하드웨어, 소프트웨어를 저장하는 판독 전용 메모리("ROM"), 랜덤 액세스 메모리("RAM") 및 비휘발성 저장장치를 암시적으로 포함할 수 있다.
- [0149] 종래의 것이든 및/또는 주문형이든 상관없이 다른 하드웨어가 또한 포함될 수 있다.
- [0150] 이와 유사하게 이 도면에 도시된 임의의 스위치는 단지 개념적인 것이다. 그 기능은 프로그램 논리의 동작을 통해, 전용 논리를 통해, 프로그램 제어 및 전용 논리의 상호작용을 통해 또는 심지어 수동으로 수행될 수 있으며, 특정 기술은 문맥으로부터 보다 구체적으로 이해되는 바와 같이 구현하는 자에 의해 선택될 수 있다.
- [0151] 본 명세서에서, "연결된"이라는 어구는 직접 연결된 것이거나 또는 하나 이상의 중간 컴포넌트를 통해 간접 연결된 것을 의미하는 것으로 이해된다. 이러한 중간 컴포넌트는 하드웨어 및 소프트웨어 기반 컴포넌트를 포함할 수 있다.
- [0152] 특허청구범위에서, 특정 기능을 수행하는 수단으로 표시된 임의의 요소는 예를 들어 a) 그 기능을 수행하는 회로 요소의 조합이나 b) 그 기능을 수행하는 소프트웨어를 실행하는 적절한 회로와 결합된 펌웨어, 마이크로 코드 등을 포함하는 임의의 형태의 소프트웨어를 포함하여 그 기능을 수행하는 임의의 방법을 포함하는 것으로 의도된다. 특허청구범위에 의해 한정된 본 발명의 원리는 여러 언급된 수단으로 제공된 기능이 특허청구범위가 요청하는 방식으로 서로 결합된 것에 존재한다. 따라서, 그 기능을 제공할 수 있는 임의의 수단은 본 명세서에 도시된 것과 균등한 것이라고 간주된다.
- [0153] 명세서에서 본 발명의 원리의 "일 실시예" 또는 "실시예"라는 언급과 그 다른 변형 어구의 언급은 실시예와 관련하여 기술된 특정 특징, 구조, 특성 등이 본 발명의 원리의 적어도 하나의 실시예에 포함되는 것을 의미한다. 따라서, 명세서 전체에 걸쳐 여러 곳에 나타나는 "일 실시예에서" 또는 "실시예에서"라는 어구의 표현과 그 임의의 다른 변형 어구는 동일한 실시예를 모두 언급하는 것이 아닐 수 있다.
- [0154] 예를 들어 "A/B", "A 및/또는 B" 그리고 "A 및 B 중 적어도 하나"에 있는 "/", " 및/또는" 및 "~ 중 적어도 하나" 중 어느 하나의 사용은 예를 들어 처음 나열된 옵션(A)만을 선택하거나 두 번째 나열된 옵션(B)만을 선택하거나 두 개의 옵션(A와 B)을 모두 선택하는 것을 포함하는 것으로 의도된 것이라는 것을 이해하여야 한다. 다른 예로서, "A, B 및/또는 C" 그리고 "A, B 및 C 중 적어도 하나"의 경우에서 이 어구는 처음 나열된 옵션(A)만을 선택하거나, 두 번째 나열된 옵션(B)만을 선택하거나, 세 번째 나열된 옵션(C)만을 선택하거나, 처음 및 두 번

제 나열된 옵션(A 와 B)만을 선택하거나, 처음과 세 번째 나열된 옵션(A와 C)만을 선택하거나, 두 번째와 세 번째 나열된 옵션(B와 C)만을 선택하거나, 3개의 옵션(A와 B와 C)을 모두 선택하는 것을 포함하는 것으로 의도된 것이다. 이것은 이 기술 분야 및 관련 기술 분야의 통상의 지식을 가진 자에게는 명백한 바와 같이 많은 항목이 나열된 것으로 확장될 수 있다.

[0155] 본 발명의 원리의 내용이 하드웨어, 소프트웨어, 펌웨어, 특수 목적 프로세서 또는 이들의 조합의 여러 형태로 구현될 수 있는 것으로 이해된다.

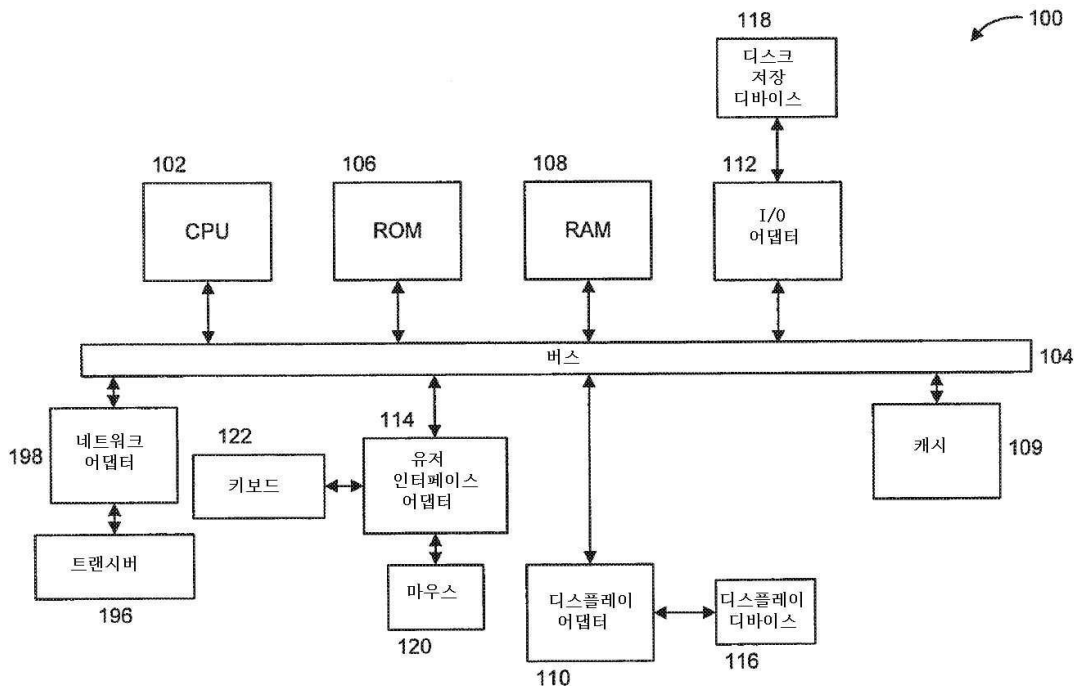
[0156] 가장 바람직하게는 본 발명의 원리의 내용은 하드웨어와 소프트웨어의 조합으로 구현된다. 나아가, 소프트웨어는 프로그램 저장 유닛에 유형적으로 구현된 애플리케이션 프로그램으로 구현될 수 있다. 애플리케이션 프로그램은 임의의 적절한 이기택처를 포함하는 기계에 업로딩되고 이 기계에 의해 실행될 수 있다. 바람직하게는, 이 기계는 하나 이상의 중앙 처리 장치("CPU"), 랜덤 액세스 메모리("RAM") 및 입력/출력("I/O") 인터페이스와 같은 하드웨어를 구비하는 컴퓨터 플랫폼에서 구현된다. 컴퓨터 플랫폼은 운영 시스템 및 마이크로명령 코드를 더 포함할 수 있다. 본 명세서에 기술된 여러 공정과 기능은 CPU에 의해 실행될 수 있는 마이크로명령 코드의 부분이나 애플리케이션 프로그램의 부분 또는 이들의 임의의 조합일 수 있다. 나아가, 여러 다른 주변 장치들이 추가적인 데이터 저장 장치와 프린팅 장치와 같은 컴퓨터 플랫폼에 연결될 수 있다.

[0157] 첨부 도면에 도시된 구성 시스템 요소 및 방법의 일부는 바람직하게는 소프트웨어로 구현될 수 있으므로, 시스템 성분이나 공정 기능 블록들 사이의 실제 연결은 본 발명의 원리가 프로그래밍되는 방식에 따라 상이할 수 있는 것으로 이해된다. 본 명세서의 내용에 따라 이 기술 분야에 통상의 지식을 가진 자라면 본 발명의 원리의 이들 및 이와 유사한 구현이나 구성을 구상할 수 있을 것이다.

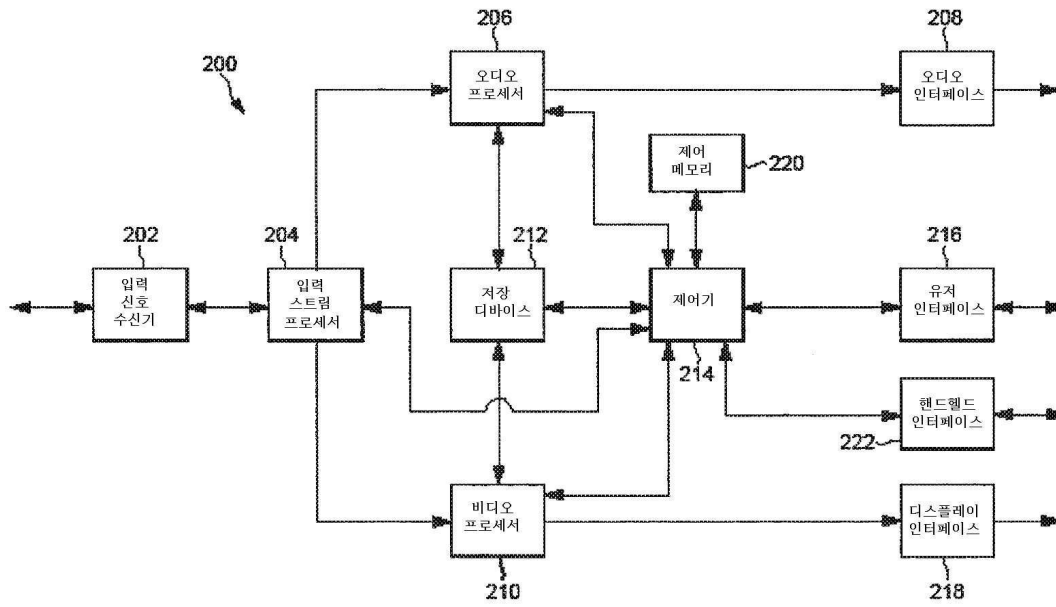
[0158] 첨부 도면을 참조하여 본 명세서에서 예시적인 실시예들이 기술되었으나, 본 발명의 원리는 이들 정확한 실시예로 제한되는 것은 아니고 본 발명의 원리의 범위나 사상을 벗어남이 없이 이 기술 분야에 통상의 지식을 가진 자라면 여러 변경과 변형이 이루어질 수 있다는 것을 이해할 수 있을 것이다. 그러므로, 모든 이러한 변경과 변형은 첨부된 청구범위에 개시된 본 발명의 원리의 범위 내에 포함된 것으로 의도된다.

도면

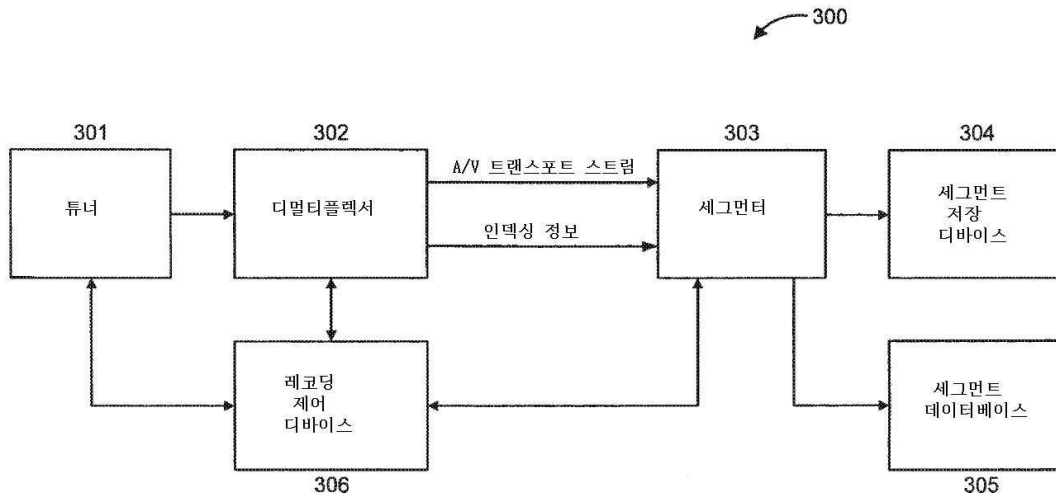
도면1



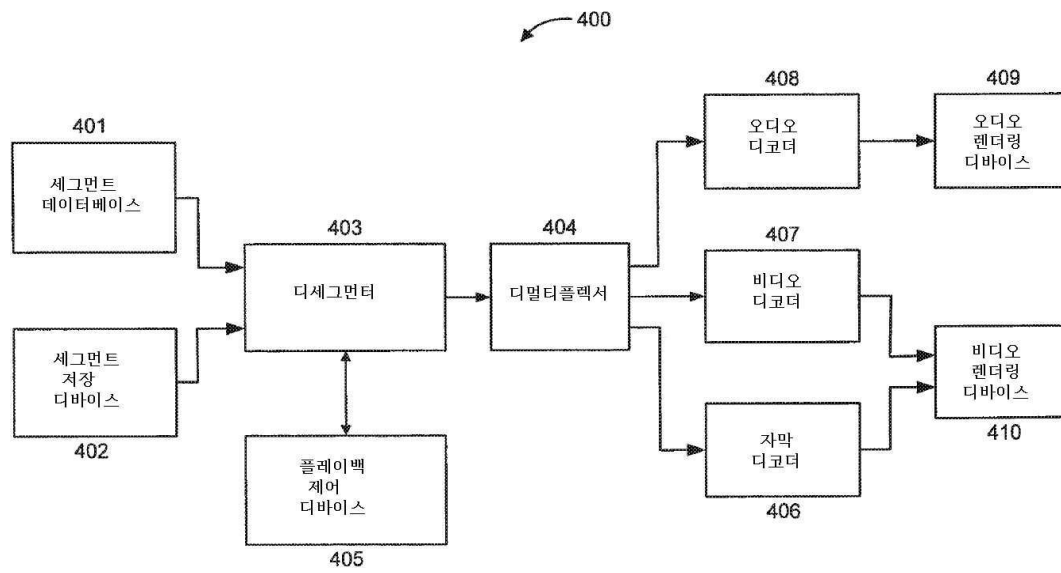
도면2



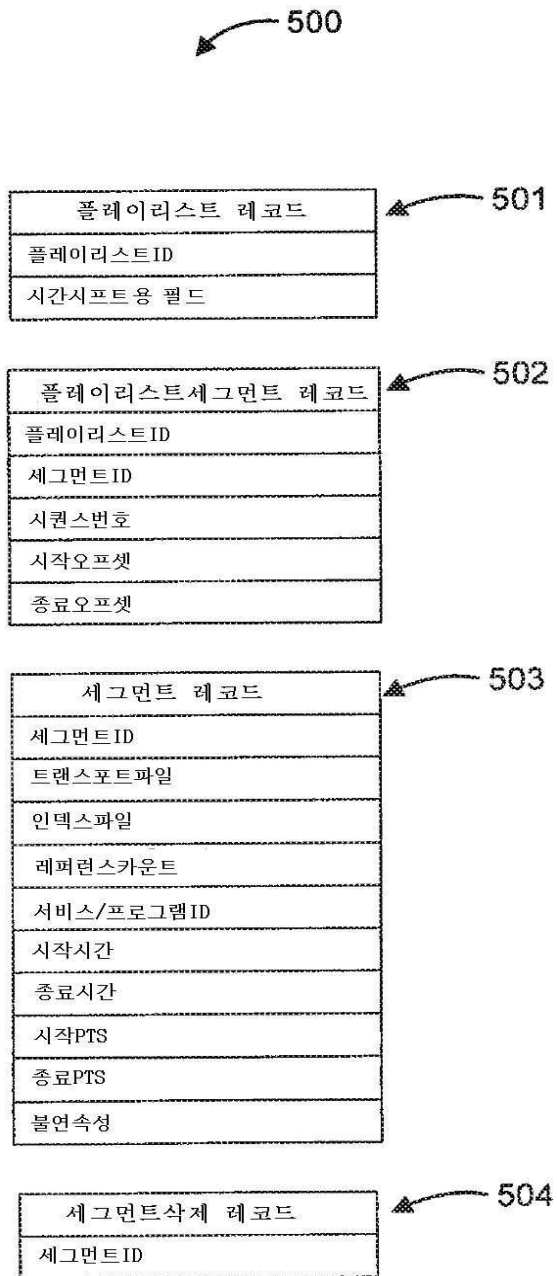
도면3



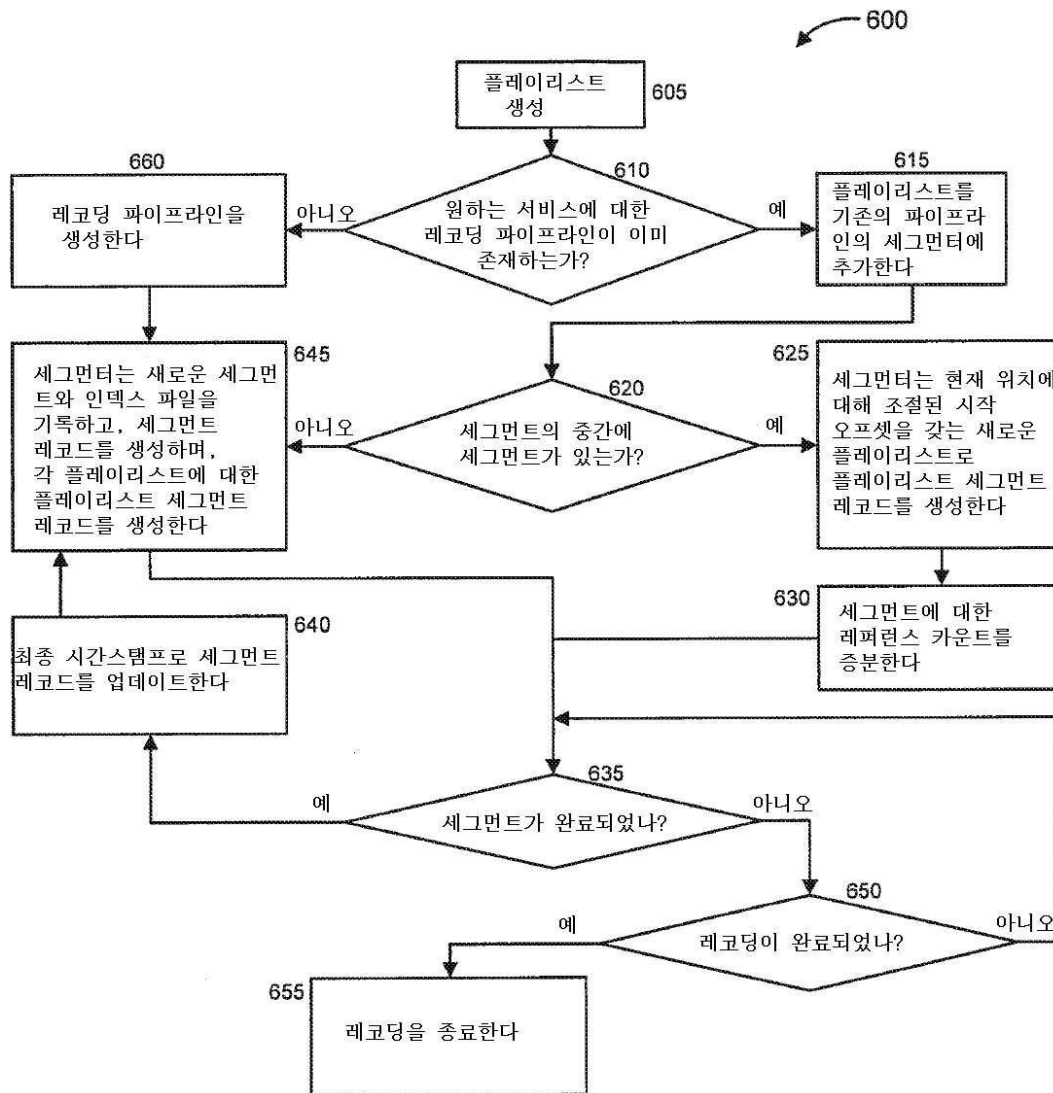
도면4



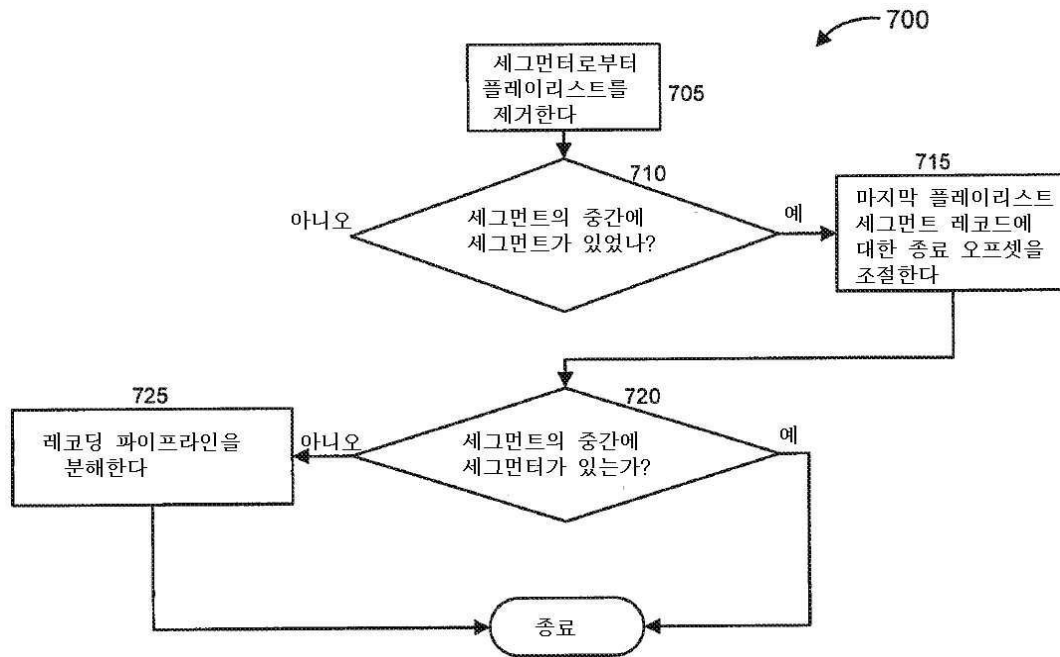
도면5



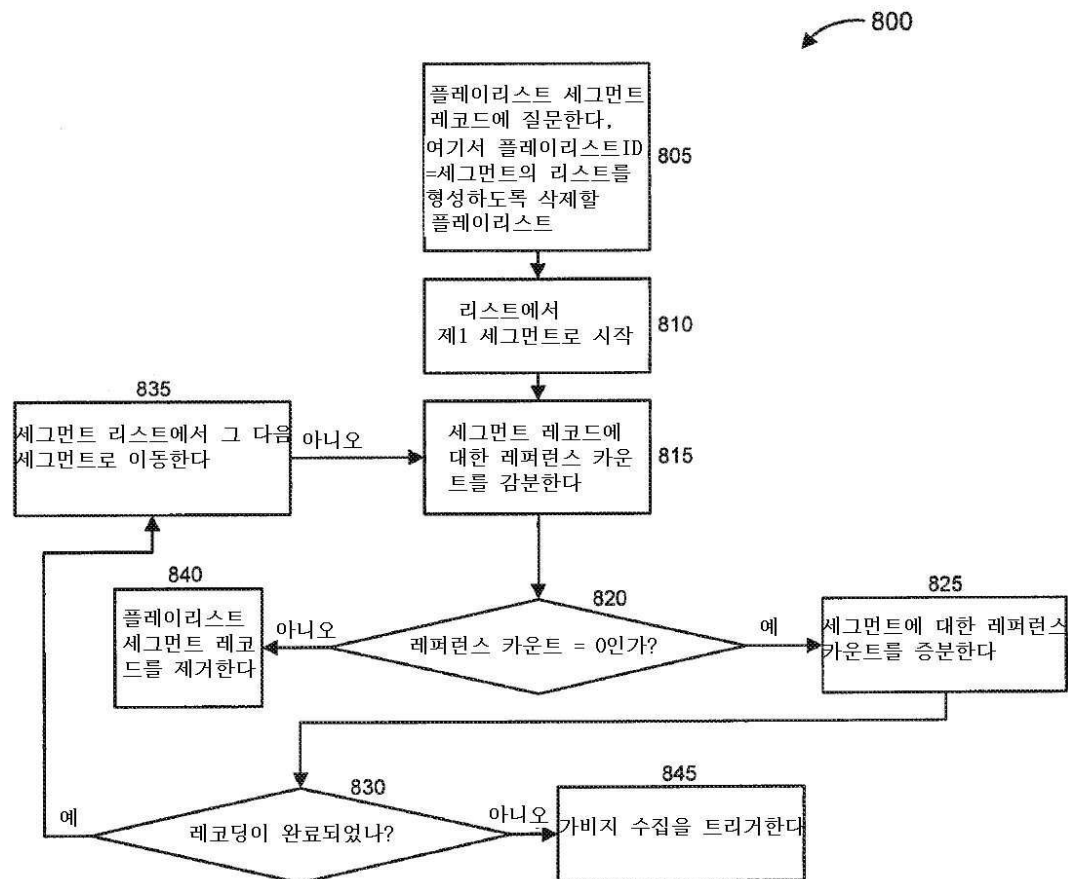
도면6



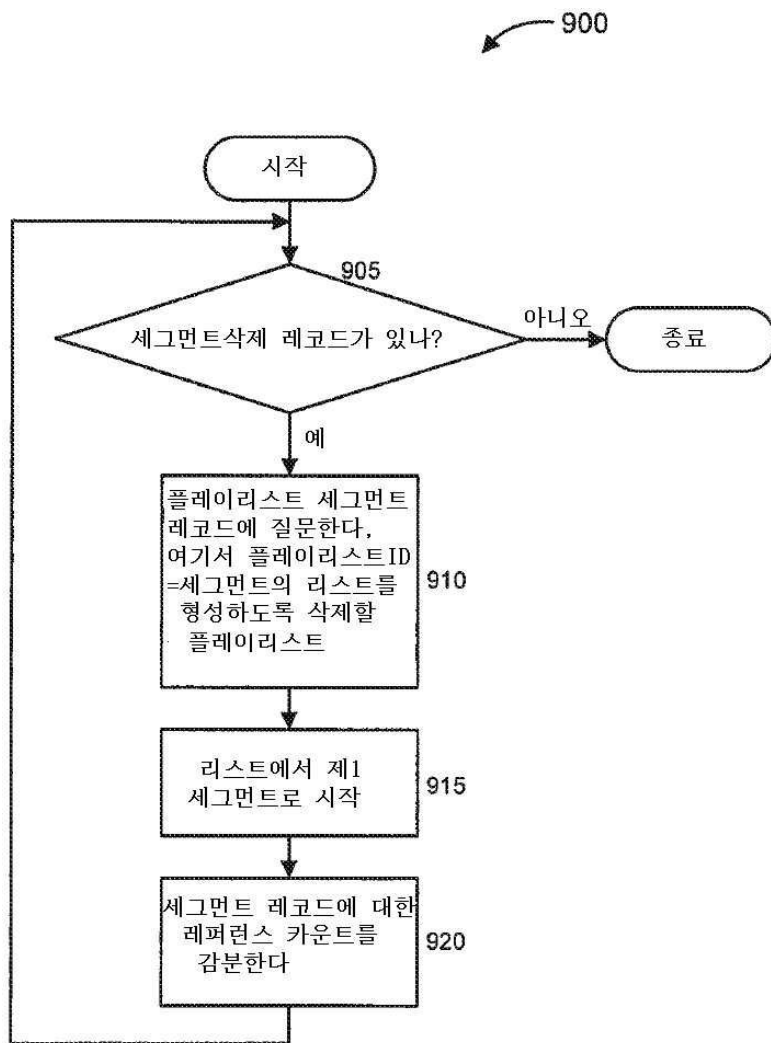
도면7



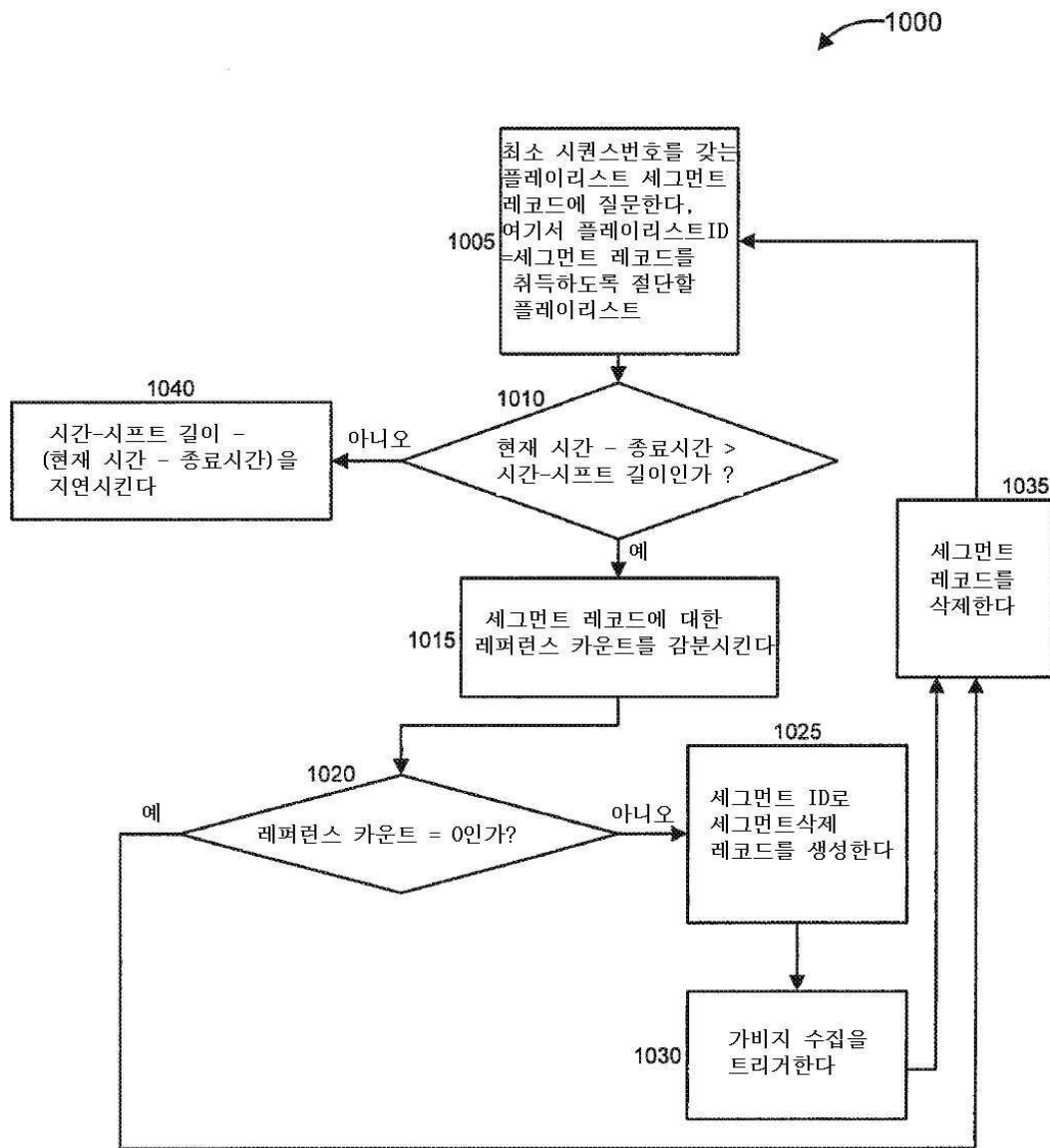
도면8



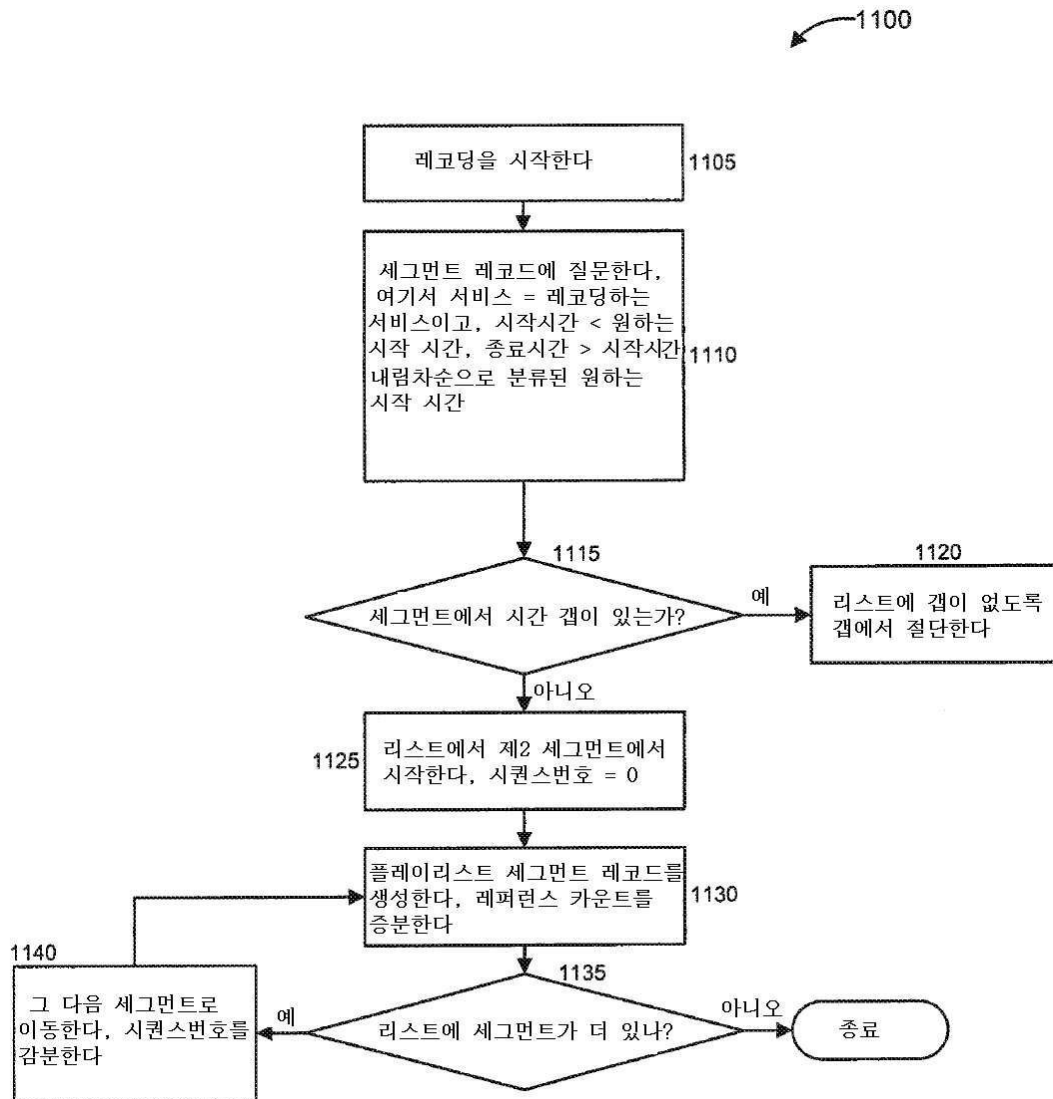
도면9



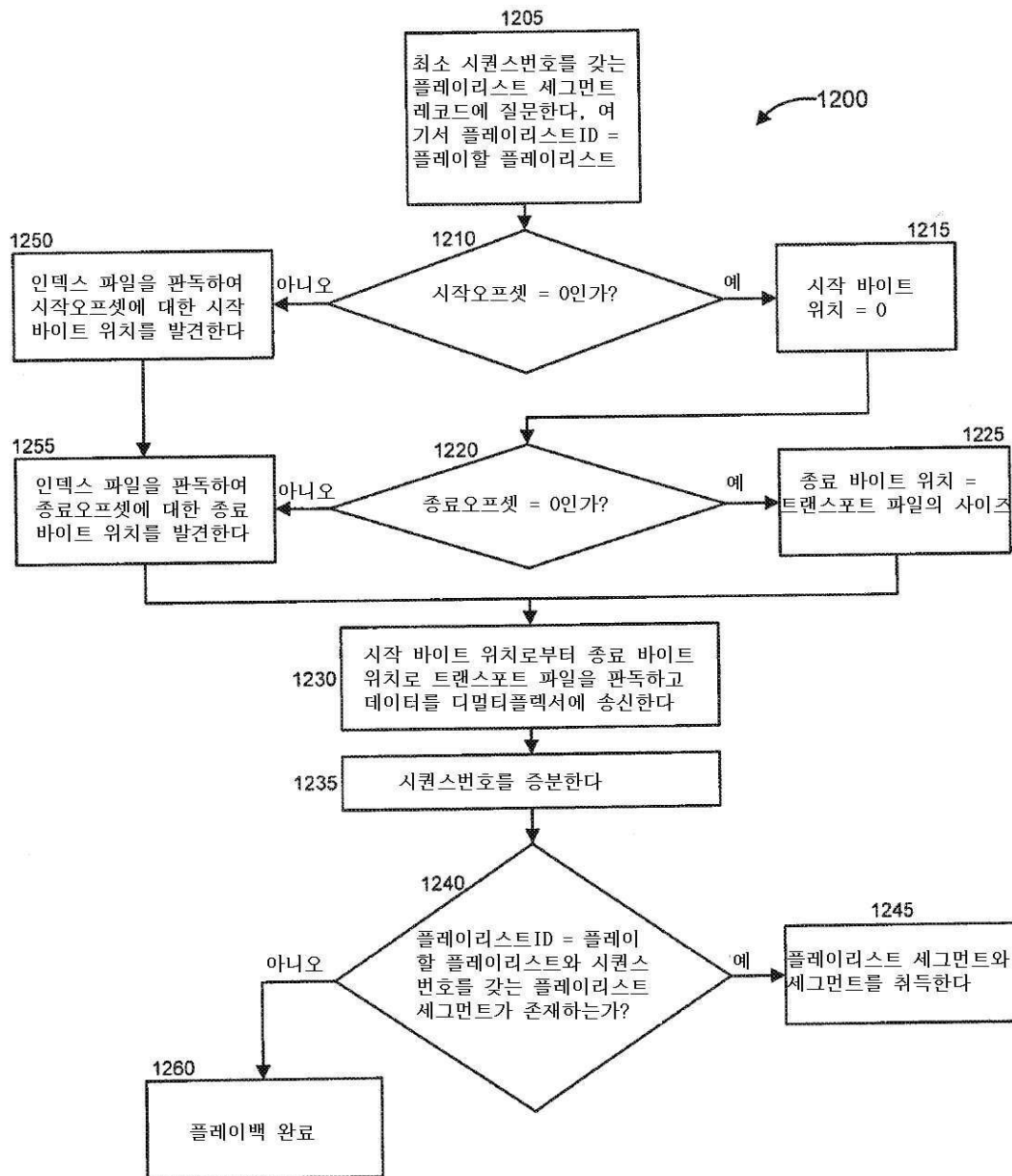
도면10



도면11



도면12



도면13

