



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년02월21일
(11) 등록번호 10-1364206
(24) 등록일자 2014년02월10일

(51) 국제특허분류(Int. Cl.)
G11B 20/10 (2006.01) G11B 27/02 (2006.01)
(21) 출원번호 10-2007-0044670
(22) 출원일자 2007년05월08일
심사청구일자 2012년05월07일
(65) 공개번호 10-2007-0111984
(43) 공개일자 2007년11월22일
(30) 우선권주장
JP-P-2006-00140839 2006년05월19일 일본(JP)
(56) 선행기술조사문헌
US06118928 A*
US20050259957 A1*
W02005101827 A1*
W02004044904 A1
*는 심사관에 의하여 인용된 문헌

(73) 특허권자
소니 주식회사
일본국 도쿄도 미나토구 코난 1-7-1
(72) 발명자
모리모토 나오키
일본 도쿄 미나토-구 코난 1-7-1 소니 가부시끼
가이샤내
아리도메 겐이치로
일본 도쿄 미나토-구 코난 1-7-1 소니 가부시끼
가이샤내
(뒷면에 계속)
(74) 대리인
신관호

전체 청구항 수 : 총 14 항

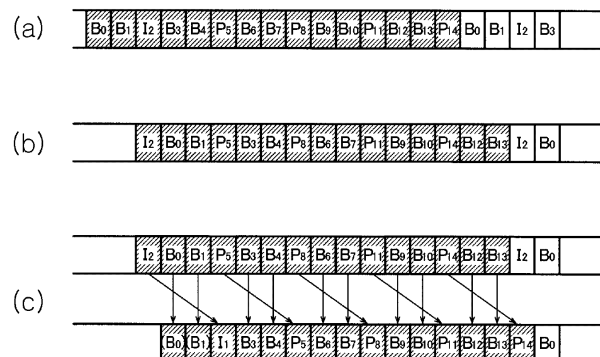
심사관 : 장진환

(54) 발명의 명칭 기록 장치, 기록 방법 및 기록 프로그램, 기록 재생장치, 기록 재생 방법 및 기록 재생 프로그램, 및, 편집 장치, 편집 방법 및 편집 프로그램

(57) 요약

비디오 데이터를 압축 인코딩하여 상기 압축 인코딩된 비디오 데이터를 기록하는 기록 장치는, 적어도 한 개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하는 인코더와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성부와, 상기 인코더에서 압축 인코딩된 비디오 데이터를 기록 매체에 기록하는 기록부를 가지며, 상기 마크 정보 생성부는, 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보를 생성한다.

대표도



(72) 발명자

이소베 유키오

일본 도쿄 미나토-쿠 코난 1-7-1 소니 가부시키 가
이샤내

마에 아츠시

일본 도쿄 미나토-쿠 코난 1-7-1 소니 가부시키 가
이샤내

특허청구의 범위

청구항 1

비디오 데이터를 압축 인코딩하여 기록매체에 기록하는 기록 장치에 있어서,

적어도 1개의 독립적으로 복호가 가능한 픽처를 포함하는 복수개의 픽처로 구성되는 그룹을 단위로 하고, 예측 인코딩에 의한 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩시키는 인코딩부와,

상기 인코딩부에서 압축 인코딩된 비디오 데이터를 기록매체에 기록하는 것을 제어하는 기록제어부와,

상기 비디오 데이터에 대하여 상기 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성부를 가지며,

상기 마크 정보 생성부는,

재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹에 대하여 상기 액세스 위치를 지정할 때에는, 상기 지정된 액세스 위치로부터 상기 그룹의 재생순서의 선두의 위치를 나타내는 상기 마크 정보를 생성하는 것을 특징으로 하는 기록 장치.

청구항 2

제 1항에 있어서,

상기 마크 정보 생성부는,

재생순서에서 직전의 그룹의 정보를 이용하여 복호를 행하는 그룹에 대하여 상기 액세스 위치를 지정할 때에는, 상기 그룹에 속하는 상기 독립적으로 복호가 가능한 픽처의 위치를 나타내는 상기 마크 정보를 생성하는 것을 특징으로 하는 기록 장치.

청구항 3

제 2항에 있어서,

지정하려고 하는 상기 액세스 위치가 포함되는 그룹이 재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹인지 아닌지를 판별하는 판별부를 추가로 가지며,

상기 마크 정보 생성부는,

상기 판별부에 의한 판별 결과에 따라, 상기 그룹의 재생순서의 선두의 위치를 나타내는 상기 마크 정보를 생성하는지, 상기 그룹에 속하는 상기 독립적으로 복호가 가능한 픽처의 위치를 나타내는 상기 마크 정보를 생성하는지를 결정하는 것을 특징으로 하는 기록 장치.

청구항 4

제 3항에 있어서,

상기 마크 정보 생성부는,

기록 개시시에 생성된 상기 그룹의 선두의 위치를 나타내는 상기 마크 정보를 자동적으로 생성하는 것을 특징으로 하는 기록 장치.

청구항 5

제 4항에 있어서,

이용자의 조작을 수신하여 이 조작에 따라 상기 액세스 위치를 지정하는 조작부를 추가로 가지며,

상기 마크 정보 생성부는,

상기 조작부에 대한 조작에 의해 지정된 액세스 위치가 기록 개시 위치 이외의 위치를 나타내고 있다면, 상기 액세스 위치가 포함된 상기 그룹에 속하는 상기 독립적으로 복호가 가능한 픽처의 위치를 나타내는 상기 마크

정보를 생성하는 것을 특징으로 하는 기록 장치.

청구항 6

제 5항에 있어서,

상기 비디오 데이터의 재생 방법을 나타내는 재생 제어 정보를 생성하는 재생 제어 정보 생성부를 더 가지며,

상기 마크 정보는, 상기 재생 제어 정보내에 저장되는 것을 특징으로 하는 기록 장치.

청구항 7

비디오 데이터를 압축 인코딩하여 기록매체에 기록하는 기록 방법에 있어서,

적어도 1개의 독립적으로 복호가 가능한 픽처를 포함하는 복수개의 픽처로 구성되는 그룹을 단위로 하고, 예측 인코딩에 의한 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하는 인코딩 단계와,

상기 인코딩 단계에서 압축 인코딩된 비디오 데이터를 기록매체에 기록하는 것을 제어하는 기록제어 단계와,

상기 비디오 데이터에 대하여 상기 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성의 단계를 가지며,

상기 마크 정보 생성의 단계는,

재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹에 대하여 상기 액세스 위치를 지정할 때에는, 상기 지정된 액세스 위치로부터 상기 그룹의 재생순서의 선두의 위치를 나타내는 상기 마크 정보를 생성하는 것을 특징으로 하는 기록 방법.

청구항 8

비디오 데이터를 압축 인코딩하여 기록매체에 기록하는 기록 방법을 컴퓨터 장치에 의해 실행시키는 기록 프로그램은 포함하는 기록 장치에 있어서,

상기 기록 방법은,

적어도 1개의 독립적으로 복호가 가능한 픽처를 포함하는 복수개의 픽처로 구성되는 그룹을 단위로 하고, 예측 인코딩에 의한 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하는 인코딩 단계와,

상기 인코딩 단계에서 압축 인코딩된 비디오 데이터를 기록매체에 기록하는 것을 제어하는 기록제어 단계와,

상기 비디오 데이터에 대하여 상기 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성의 단계를 가지며,

상기 마크 정보 생성의 단계는,

재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹에 대하여 상기 액세스 위치를 지정할 때에는, 상기 지정된 액세스 위치로부터 상기 그룹의 재생순서의 선두의 위치를 나타내는 상기 마크 정보를 생성하는 것을 특징으로 하는 기록 장치.

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

청구항 12

삭제

청구항 13

삭제

청구항 14

비디오 데이터에 대하여 액세스 위치를 지정하는 마크 정보를 생성하는 편집 장치에 있어서,

적어도 1개의 독립적으로 복호가 가능한 픽처를 포함하는 복수개의 픽처로 구성되는 그룹을 단위로 하고, 예측 인코딩에 의한 프레임간 압축을 이용하여 압축 인코딩된 비디오 데이터를 복호하는 디코더와,

상기 비디오 데이터에 대하여 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성부를 가지며,

상기 마크 정보 생성부는,

재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹에 대하여 상기 액세스 위치를 지정할 때에는, 상기 지정된 액세스 위치로부터 상기 그룹의 재생순서의 선두의 위치를 나타내는 상기 마크 정보를 생성하는 것을 특징으로 하는 편집 장치.

청구항 15

제 14항에 있어서,

상기 마크 정보 생성부는,

재생순서에서 직전의 그룹의 정보를 이용하여 복호를 행하는 그룹에 대하여 상기 액세스 위치를 지정할 때에는, 상기 그룹에 속하는 상기 독립적으로 복호가 가능한 픽처의 위치를 나타내는 상기 마크 정보를 생성하는 것을 특징으로 하는 편집 장치.

청구항 16

제 14항 또는 제 15항에 있어서,

상기 액세스 위치를 포함하는 그룹이 재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹인지 아닌지를 판별하는 판별부를 추가로 가지며,

상기 마크 정보 생성부는,

상기 판별부에 의한 판별 결과에 따라, 상기 그룹의 재생순서의 선두의 위치를 나타내는 상기 마크 정보를 생성하는지, 상기 그룹에 속하는 상기 독립적으로 복호가 가능한 픽처의 위치를 나타내는 상기 마크 정보를 생성하는지를 결정하는 것을 특징으로 하는 편집 장치.

청구항 17

삭제

청구항 18

삭제

청구항 19

비디오 데이터에 대하여 액세스 위치를 지정하는 마크 정보를 생성하는 편집 방법에 있어서,

적어도 1개의 독립적으로 복호가 가능한 픽처를 포함하는 복수개의 픽처로 구성되는 그룹을 단위로 하고, 예측 인코딩에 의한 프레임간 압축을 이용하여 압축 인코딩된 비디오 데이터를 복호하는 복호 단계와,

상기 비디오 데이터에 대하여 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성 단계를 가지며,

상기 마크 정보 생성 단계는,

재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹에 대하여 상기 액세스 위치를 지정할 때에

는, 상기 지정된 액세스 위치로부터 상기 그룹의 재생순서의 선두의 위치를 나타내는 상기 마크 정보를 생성하는 것을 특징으로 하는 편집 방법.

청구항 20

비디오 데이터에 대하여 액세스 위치를 지정하는 마크 정보를 생성하는 편집 방법을 컴퓨터 장치에 의해 실행시키는 프로그램을 기록한 기록 매체에 있어서,

상기 프로그램은,

적어도 1개의 독립적으로 복호가 가능한 픽처를 포함하는 복수개의 픽처로 구성되는 그룹을 단위로 하고, 예측 인코딩에 의한 프레임간 압축을 이용하여 압축 인코딩된 비디오 데이터를 복호하는 복호 단계와,

상기 비디오 데이터에 대하여 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성 단계를 가지며,

상기 마크 정보 생성 단계는,

재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹에 대하여 상기 액세스 위치를 지정할 때에는, 상기 지정된 액세스 위치로부터 상기 그룹의 재생순서의 선두의 위치를 나타내는 상기 마크 정보를 생성하는 것을 특징으로 하는 기록 매체.

청구항 21

비디오 데이터를 압축 인코딩하여 기록매체에 기록하는 기록 장치에 있어서,

적어도 1개의 독립적으로 복호가 가능한 픽처를 포함하는 복수개의 픽처로 구성되는 그룹을 단위로 하고, 예측 인코딩에 의한 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩시키는 인코딩부와,

상기 인코딩부에서 압축 인코딩된 비디오 데이터를 기록매체에 기록하는 것을 제어하는 기록제어부와,

상기 비디오 데이터에 대하여 상기 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성부를 가지며,

상기 마크 정보 생성부는,

재생순서에서 직전의 그룹의 정보를 이용하여 복호를 행하는 그룹에 대하여 상기 액세스 위치를 지정할 때에는, 상기 그룹에 속하는 상기 독립적으로 복호가 가능한 픽처의 위치를 나타내는 상기 마크 정보를 생성하는 기록 장치.

청구항 22

삭제

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

[0039] 본 발명은 2006년 5월 19일 일본 특허청에 제출된 일본특허 JP 2006-140839호에 기재된 주제와 관련되며, 그 전체 내용은 참조로서 여기에 포함되어 있다.

[0040] 본 발명은, AV(Audio/Video) 스트림 데이터를 포함하는 클립에 대한 편집 처리를 실시하는 기록 장치, 기록 방법 및 기록 프로그램, 기록/재생장치, 기록/재생 방법 및 기록/재생 프로그램, 및, 편집 장치, 편집 방법 및 편집 프로그램에 관한 것이다.

[0041] 종래부터, 기록 가능한 기록 재생장치로부터 분리가 가능하고, 기록 용량이 비교적 크고, 비디오 데이터와 오디오 데이터를 포함하는 AV(Audio/Video) 데이터를 기록하는데 적합한 기록 매체로서, 4.7GB(Giga Byte) 이상의 기록 용량을 가지는 DVD(Digital Versatile Disc)가 보급되어 있다. 일본 특허 출원 공개 번호 특개 2004-

350251에는, 기록 가능한 타입의 DVD에서 DVD-Video 포맷으로 AV 데이터를 기록하는 촬상 장치가 나타내져 있다.

[0042] 이러한 기록 가능한 기록 매체내에 기록된 AV 데이터는, 편집 가능하다는 것이 바람직하다. AV데이터에 대한 대표적인 편집 방법의 예로서는, 소정 단위의 AV(Audio/Video) 데이터를 분할 편집하는 것을 들 수 있다. 비디오 데이터 및 오디오 데이터는, 일반적으로는 다중화된 소정의 데이터 단위로서 기록된다.

[0043] 또한, 비디오 데이터는, 데이터 용량이 방대하기 때문에, 소정의 방식으로 압축 인코딩 되어 기록 매체에 기록 되는 것이 일반적이다. 비디오 데이터의 압축 인코딩 방식으로서, MPEG2(Moving Picture Experts Group 2) 방식이 표준적인 방식으로 알려져 있다. 또한, 근래에는, MPEG2의 압축 인코딩 방식을 더욱 발전시킴으로써 효율적인 인코딩이 실행될 수 있도록 하는 ITU-T(International Telecommunication Union-Telecommunication Standardization Sector) 권고 H.264 혹은 ISO(International Organization for Standardization)/IEC(International Electrotechnical Commission) 국제 표준 14496-10(MPEG-4 파트 10) Advanced Video Coding(이하, H.264 | AVC로 약칭한다)도, 역시 널리 보급되어져 있다.

[0044] MPEG2 및 H.264 | AVC에 따르면, 직교변환 등을 이용한 프레임간 부호화를 실시함과 동시에, 이동 보상(motion compensation)을 이용한 예측 인코딩에 의한 프레임간 인코딩을 실시하여, 압축율을 높이고 있다. 이하, MPEG2를 예를 들고, 예측 인코딩에 의한 프레임간 압축에 대해 설명한다.

[0045] 먼저, MPEG2의 데이터 스트림 구조에 대해서, 개략적으로 설명한다. MPEG2는, 이동 보상에 근거한 예측 인코딩과 DCT에 의한 압축 인코딩을 조합한 것이다. MPEG2의 데이터 구조는, 계층 구조를 이루고 있어 하위로부터, 블록층, 매크로 블록층, 슬라이스층, 픽처층, GOP층 및 순서층으로 되어 있다. 블록층은, DCT를 실시하는 단위인 DCT 블록으로 구성된다. 매크로 블록층은, 복수의 DCT 블록으로 구성된다. 슬라이스층은, 헤더부와 한 개 이상의 매크로 블록으로 구성된다. 픽처층은, 헤더부와 한 개 이상의 슬라이스로 구성된다. 픽처는, 1 화면에 대응한다. 각 층의 경계는, 각각 소정의 식별 부호로 식별 가능하게 되어 있다.

[0046] GOP층은, 헤더부와 프레임간 부호화(intraframe coding)에 근거하는 I(Intra-coded) 픽처, 예측 코드에 근거하는 P(Predictive-coded) 픽처와, B(Bi-directionally predictive coded) 픽처로 구성된다. I픽처는, 자신의 정보만으로 디코딩이 가능하고, P 및 B픽처는, 시간적으로 이전 혹은 전후의 화상이 기준 화상과 기준 화상들로서 사용되지 않는다면 단독으로는 디코딩되지 않는다. 예를 들면 P픽처는, 자신보다 시간적으로 이전의 I픽처 또는 P픽처를 기준 화상으로서 이용하여 디코딩된다. 또, B 픽처는, 자신의 전후의 I픽처 또는 P픽처의 2 개의 픽처를 기준 화상으로서 이용해 디코딩된다. 적어도 1개의 I픽처를 포함하는 독립된 그룹을 GOP(Group Of Picture)라고 부르며, MPEG의 스트림내에서 독립적으로 액세스 가능한 최소 단위로 간주된다.

[0047] GOP는, 1 또는 복수의 픽처로 구성된다. 이하에서는, GOP는, 복수의 픽처로 구성되는 것으로 한다. 자체적으로 완전하게 디코딩이 가능하고, 닫혀진 구조(closed structure)를 가지는 폐쇄 GOP와, 코딩의 순서에서 개방 GOP 바로 이전의 GOP에 대한 정보를 이용하여 디코딩 가능한 오픈(open) GOP와 2종류가 있다.

[0048] 도 1a-도 1c를 참조하여, 프레임간 압축 데이터를 디코딩하는 처리에 대해 기술한다. 1개의 GOP가 1개의 I픽처, 4개의 P픽처 및 10개의 B픽처 즉, 합계 15개의 픽처로 구성되어 있는 것으로 한다. I, P 및 B픽처의 표시

순서는, 도 1a에 도시된 바와 같이, [
$$B_0 B_1 I_2 B_3 B_4 P_5 B_6 B_7 P_8 B_9 B_{10} P_{11} B_{12} B_{13} P_{14}$$
]와 같이 된다. 도 1a-도 1c에서는, 첨자는 표시순서를 나타낸다.

[0049] 이 예에서는, 최초의 2개의 B픽처, 즉 B0픽처 및 B1픽처는, 바로 이전의 GOP에 있어서의 최후의 P14 픽처와 이 GOP내의 I2픽처를 이용하여 디코딩로부터 예측되어 얻어진다. GOP내의 최초의 P5픽처는, I2픽처로부터 예측되어 디코딩된 픽처이다. 다른 P8픽처, P11 픽처 및 P14는, 각각 1개전의 P픽처를 이용해 예측되어 디코딩된 픽처이다. 또한, I픽처 이후의 각 B픽처는, 각각 전후의 I 및 / 또는 P픽처로부터 예측되어 디코딩된 픽처이다.

[0050] 각 B픽처는, 시간적으로 전후의 I 또는 P픽처를 이용해 예측되어 디코딩되기 때문에, 스트림이나 기록 매체상에 있어서의 I, P 및 B픽처의 표시순서는, 디코딩 순서를 고려해 결정할 필요가 있다. 즉, B픽처를 디코딩하기 위한 I 및 / 또는 P픽처는, 해당 B픽처보다 항상 먼저 디코딩되어 있지 않으면 안 된다.

[0051] 상술한 예에서는, 스트림이나 기록 매체상의 각 픽처의 배열은, 도 1b에 도시된 바와 같이,

[“I₂B₀B₁P₅B₃B₄P₈B₆B₇P₁₁B₉B₁₀P₁₄B₁₂B₁₃”]와 같이 되므로, 이 순서대로 디코드된다. 첨자는, 도 1a에 대응하며, 표시 순서번호를 나타낸다.

[0052] 디코더에 있어서의 디코드 처리는, 다음과 같다. 도 1c에 도시된 바와 같이, 먼저 I₂픽처를 디코드하고, 디코드된 I₂픽처와 1개전의 GOP에 있어서의 (표시 순서상) 최후의 P₁₄ 픽처를 이용하여 B₀픽처 및 B₁픽처를 예측해 디코드한다. 그리고, B₀픽처 및 B₁픽처가 디코드된 순서에 따라 디코더로부터 출력되며, 다음에 I₂픽처를 출력한다. B₁픽처가 출력되면, 다음에 P₅픽처가 I₂픽처를 이용해 예측되어 디코드된다. 그리고, I₂픽처 및 P₅픽처를 이용해 B₃픽처 및 B₄픽처가 예측되어 디코드된다. 그리고, 디코드된 B₃픽처 및 B₄픽처가 디코드된 순서에 따라 디코더로부터 출력되며, 다음에 P₈픽처를 출력한다.

[0053] 그 후에는, 다음의 처리가 비슷하게 반복 실행된다. B픽처의 예측에 이용하는 P 또는 I픽처를 B픽처보다 먼저 디코드하고, 이 디코드된 P 또는 I픽처를 이용해 B픽처를 예측하여 디코드한다. 디코드된 B픽처를 출력하고 나서, 해당 B픽처를 디코드하기 위해서 이용하는 P 또는 I픽처를 출력하는 처리가 반복된다. 도 1b에 도시된 바와 같이, 기록 매체상이나 스트림에 있어서의 픽처 배열은, 일반적으로 이용되는 것이다.

발명이 이루고자 하는 기술적 과제

[0054] 이와 같이, GOP 구조에 근거하는 프레임간 압축을 실시한 비디오 데이터에 대해서 액세스 위치를 지정함으로써, 비디오 데이터를 분할 편집하려고 하는 경우, GOP의 경계를 지정해 분할하는 제 1의 방법과 GOP내의 I픽처를 지정해 분할하는 제 2의 방법이 생각된다. 제 1의 방법에 의하면, I픽처 이후의 픽처는, 도 1a-도 1c를 참조하여 설명한 것처럼, I픽처를 참조해 차례차례, 디코드될 수 있다. 그러므로, 재생시에, 편집 결과에 근거해 분할된 비디오 데이터의 선두에 액세스 했을 때에, 점프 목적지 프레임의 표시에 지연이 적다는 이점이 있다. 한편, 이 제 1의 방법에서는, 분할 지정된 경계 위치와 GOP의 경계가 일치하지 않게 된다. 그 때문에, 분할된 비디오 데이터의 선두에 액세스 했을 경우, GOP의 선두의 픽처로부터 재생할 수 없다는 문제점이 있었다.

[0055] 또한, 제 1의 방법에 의하면, 분할된 비디오 데이터의 각 분할의 후단부는, GOP경계와 일치하지 않는다. 그 때문에, 예를 들면 분할된 비디오 데이터의 후단의 픽처가 포함되는 GOP가, 선행하는 GOP와는 다른 장면으로서 생성된 GOP인 경우, 재생시에, 해당 분할된 비디오 데이터의 후단부의 몇 개의 프레임 만큼, 관련 없는 장면들이 재생되어 버린다하는 문제점이 있었다.

[0056] 제 2의 방법에서는, GOP의 선두와 편집에 의해 분할 지정된 경계가 일치하므로, 상술의 제 1의 방법의 경우와는 달리, 각 GOP의 선두측에서 재생되지 않는 픽처가 발생하거나, 분할된 비디오 데이터의 후단부에 관련 없는 장면들이 재생되는 문제는 발생하지 않는다.

[0057] 그렇지만, 이 제 2의 방법에서는, 분할된 비디오 데이터의 선두의 프레임이 포함되는 GOP가 오픈 GOP인 경우, 해당 GOP의 표시순서가 I픽처 이전의 B픽처는, 도 1a-도 1c를 참조하여 설명한 것처럼, 해당 GOP의 I픽처와 직전의 GOP에 있어서의 마지막 P픽처를 이용해 디코드된다. 그 때문에, 분할된 비디오 데이터로 점프되었을 때에는, 해당 GOP의 직전의 GOP의 마지막 P픽처도 디코드할 필요가 있으므로, 점프 조작에 따라 영상 출력의 지연이 커져 버린다는 문제점이 있었다.

[0058] 이와 같이, 종래에는, GOP 구조에 근거하는 프레임간 압축을 실시한 비디오 데이터에 대해서 분할하는 편집을 실시하는 경우에, 재생 영상에 문제가 발생하지 않고, 이용자의 편리성을 해치지 않는 방법이 존재하지 않았다고 하는 문제점이 있었다.

[0059] 따라서, 이 발명의 목적은, GOP 구조를 가진 비디오 데이터에 대해서 분할 편집을 실시할 때의 이용자의 편리성을 향상시킬 수 있는 기록 장치, 기록 방법 및 기록 프로그램, 기록/재생장치, 기록 재생 방법 및 기록 재생 프로그램, 및, 편집 장치, 편집 방법 및 편집 프로그램을 제공하는 것에 있다.

[0060] 본 발명의 제 1실시예에 따르면, 비디오 데이터를 압축 인코딩하여 상기 압축 인코딩된 비디오 데이터를 기록하는 기록 장치가 제공되며, 상기 장치는, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하는 인코더와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성부와, 상기 인코더에서 압축 인코딩된 비디오 데이터를 기록 매체에 기록하는 기록부를 가지며, 상기 마크 정보 생성부는, 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호

가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보를 생성한다.

[0061] 본 발명의 제 2실시예에 따르면, 비디오 데이터를 압축 인코딩하여 상기 압축 인코딩된 비디오 데이터를 기록하는 기록 방법이 제공되며, 상기 방법은, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하는 단계와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 단계와, 상기 비디오 데이터를 압축 인코딩하는 단계에서, 상기 압축 인코딩된 비디오 데이터를 기록 매체에 기록하는 단계를 가지며, 상기 마크 정보를 생성하는 단계에서는, 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보가 생성된다.

[0062] 본 발명의 제 3실시예에 따르면, 비디오 데이터를 압축 인코딩하여 상기 압축 인코딩된 비디오 데이터를 기록하는 기록 방법을 컴퓨터 장치에서 실행시키는 기록 프로그램이 제공되며, 상기 기록 방법은, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하는 단계와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 단계와, 상기 비디오 데이터를 압축 인코딩하는 단계에서, 상기 압축 인코딩된 비디오 데이터를 기록 매체에 기록하는 단계를 가지며, 상기 마크 정보를 생성하는 단계에서는, 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보가 생성된다.

[0063] 본 발명의 제 4실시예에 따르면, 비디오 데이터를 압축 인코딩하여 상기 압축 인코딩된 비디오 데이터를 기록 매체내에 기록하고, 상기 기록 매체에 기록된 압축 인코딩된 비디오 데이터를 재생하여 복호화하는 기록/재생장치가 제공되며, 상기 장치는, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하는 인코더와, 상기 인코더에서 압축 인코딩된 비디오 데이터를 기록 매체에 기록하는 기록부와, 상기 기록 매체로부터 압축 인코딩된 비디오 데이터를 읽어내 복호화하는 재생부와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성부를 가지며, 상기 마크 정보 생성부는, 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보를 생성한다.

[0064] 본 발명의 제 5실시예에 따르면, 비디오 데이터를 압축 인코딩하여 상기 압축 인코딩된 비디오 데이터를 기록 매체내에 기록하고, 상기 기록 매체에 기록된 압축 인코딩된 비디오 데이터를 재생하여 복호화하는 기록/재생 방법이 제공되며, 상기 방법은, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하는 단계와, 상기 비디오 데이터를 압축 인코딩하는 단계에서, 상기 압축 인코딩된 비디오 데이터를 기록 매체에 기록하는 단계와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 단계를 가지며, 상기 마크 정보를 생성하는 단계에서는, 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보가 생성된다.

[0065] 본 발명의 제 6실시예에 따르면, 비디오 데이터를 압축 인코딩하여 상기 압축 인코딩된 비디오 데이터를 기록 매체내에 기록하고, 상기 기록 매체에 기록된 압축 인코딩된 비디오 데이터를 재생하여 복호화하는 기록/재생 방법을, 컴퓨터 장치에서 실행시키는 기록/재생 프로그램이 제공되며, 상기 기록/재생 방법은, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하는 단계와, 상기 비디오 데이터를 압축 인코딩하는 단계에서, 상기 압축 인코딩된 비디오 데이터를 기록 매체에 기록하는 단계와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 단계를 가지며, 상기 마크 정보를 생성하는 단계에서는, 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보가 생성된다.

- [0066] 본 발명의 제 7실시예에 따르면, 비디오 데이터에 대해서 액세스 위치를 지정하는 마크 정보를 생성하는 편집 장치가 제공되며, 상기 장치는, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 압축 코드된 비디오 데이터를 복호하는 디코더와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 마크 정보 생성부와, 이용자의 조작에 따라 이용자의 조작을 수신하여 상기 액세스 위치를 지정하는 조작부를 가지며, 상기 마크 정보 생성부는, 조작부의 조작에 응답하여 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보를 생성하며, 상기 마크 정보 생성부는, 이용자의 조작에 응답하여 재생순서에서 직전의 그룹의 정보를 이용하여 복호화된 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 그룹들중 각 그룹에 속하며, 상기 독립적으로 복호가 가능한 픽처의 위치를 나타내는 상기 마크 정보를 생성한다.
- [0067] 본 발명의 제 8실시예에 따르면, 비디오 데이터에 대해서 액세스 위치를 지정하는 마크 정보를 생성하는 편집 방법이 제공되며, 상기 편집 방법은, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 압축 코드된 비디오 데이터를 복호하는 단계와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 단계와, 이용자의 조작에 따라 이용자의 조작을 수신하여 상기 액세스 위치를 지정하는 단계를 구비하며, 상기 마크 정보를 생성하는 단계에서는, 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보가 생성되며, 상기 마크 정보 생성 단계에서는, 이용자의 조작에 응답하여 재생순서에서 직전의 그룹의 정보를 이용하여 복호화된 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에, 상기 그룹들중 각 그룹에 속하며, 상기 독립적으로 복호가 가능한 픽처의 위치를 나타내는 상기 마크 정보가 생성된다.
- [0068] 본 발명의 제 9실시예에 따르면, 비디오 데이터에 대해서 액세스 위치를 지정하는 마크 정보를 생성하는 편집 방법을 컴퓨터 장치에서 실행시키는 편집 프로그램이 제공되며, 상기 편집 방법은, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 압축 코드된 비디오 데이터를 복호하는 단계와, 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성하는 단계와, 이용자의 조작에 따라 이용자의 조작을 수신하여 상기 액세스 위치를 지정하는 단계를 구비하며, 상기 마크 정보를 생성하는 단계에서는, 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보가 생성되며, 상기 마크 정보 생성 단계에서는, 이용자의 조작에 응답하여 재생순서에서 직전의 그룹의 정보를 이용하여 복호화된 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에, 상기 그룹들중 각 그룹에 속하며, 상기 독립적으로 복호가 가능한 픽처의 위치를 나타내는 상기 마크 정보가 생성된다.
- [0069] 상술한 것처럼, 제 1, 제 2 및 제 3의 실시예에 따르면, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩한다. 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보를 생성한다. 그러므로, 기록 매체로부터 비디오 데이터를 재생했을 때에, 재생순서에서 바로 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹으로의 점프를 지정했을 경우, 점프 목적지에서 표시 지연을 줄일 수 있다.
- [0070] 또한, 제 4, 제 5 및 제 6의 실시예에 따르면, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 압축 인코딩된 비디오 데이터를 기록 매체내에 기록한다. 그리고, 기록 매체로부터 압축 인코딩된 비디오 데이터를 읽어 내어 복호화한다. 비디오 데이터에 대해서, 재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹들중 한 그룹에 대해서 액세스 위치를 지정할 경우에는, 그룹의 선두의 위치를 나타내는 마크 정보를 생성하므로, 기록 매체로부터 재생된 비디오 데이터에 대해서, 재생순서에서 바로 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹으로의 점프를 지정했을 경우, 점프 목적지에서 표시 지연을 줄일 수 있다.
- [0071] 또한, 제 7, 제 8 및 제 9의 실시예에 따르면, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이

터를 압축 코드화한다. 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성되어, 재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호화된 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 복호가능한 그룹의 선두의 위치를 나타내낸다. 게다가, 이용자의 조작에 응답하여 재생순서에서 직전의 그룹의 정보를 이용하여 복호가능한 그룹들중 한 그룹에 대해서 액세스 위치를 지정할 경우에는, 각 그룹에 속하는 독립적으로 복호가 가능한 최소한 한 개의 픽처의 위치를 나타내도록 마크 정보가 생성된다. 그러므로, 비디오 데이터에 대한 점프를 지정했을 때에, 점프 목적지에서의 표시 지연을 줄일 수 있다.

[0072] 본 발명의 제 1, 제 2 및 제 3의 실시예에 따르면, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수 개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하며, 그것을 기록매체에 기록한다. 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보를 생성한다. 그러므로, 기록 매체로부터 비디오 데이터를 재생했을 때에, 재생순서에서 바로 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹으로의 점프를 지정했을 경우, 점프 목적지에서 표시 지연을 줄일 수 있다.

[0073] 제 4, 제 5 및 제 6의 실시예에 따르면, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 압축 인코딩된 비디오 데이터를 기록 매체내에 기록한다. 그리고, 기록 매체로부터 압축 인코딩된 비디오 데이터를 읽어내어 복호화한다. 비디오 데이터에 대해서, 재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹들중 한 그룹에 대해서 액세스 위치를 지정할 경우에는, 그룹의 선두의 위치를 나타내는 마크 정보를 생성하므로, 기록 매체로부터 재생된 비디오 데이터에 대해서, 재생순서에서 바로 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹으로의 점프를 지정했을 경우, 점프 목적지에서 표시 지연을 줄일 수 있다.

[0074] 제 7, 제 8 및 제 9의 실시예에 따르면, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 코드화한다. 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 액세스 위치를 나타내는 마크 정보를 생성되어, 재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호화된 그룹들중 한 그룹에 대해서 상기 액세스 위치를 지정할 경우에는, 복호가능한 그룹의 선두의 위치를 나타내낸다. 게다가, 이용자의 조작에 응답하여 재생순서에서 직전의 그룹의 정보를 이용하여 복호가능한 그룹들중 한 그룹에 대해서 액세스 위치를 지정할 경우에는, 각 그룹에 속하는 독립적으로 복호가 가능한 최소한 한 개의 픽처의 위치를 나타내도록 마크 정보가 생성된다. 그러므로, 비디오 데이터에 대한 점프를 지정했을 때에, 점프 목적지에서의 표시 지연을 줄일 수 있다.

[0075] 본 발명의 이러한 목적들과 다른 목적들, 특징들, 및 장점들은 첨부된 도면에 도시된 바와 같이, 다음에 기술된 최선의 실시예를 고려하면 더욱 명백하게 될 것이다.

발명의 구성 및 작용

[0076] 이하, 이 발명의 실시의 한 형태를, 도면을 참조하면서 설명한다. 먼저, 이해를 용이하게 하기 위해서, 이 발명에 적용 가능한 일례의 포맷(이하, AVCHD 포맷이라고 부른다.)에 대해 설명한다. AVCHD 포맷은, 비디오 데이터와 오디오 데이터가 소정의 형태로 다중화된 AV(Audio/Video) 스트림을 기록 가능한 기록 매체에 기록하는 기록 포맷으로서 현재 제안되고 있다. AVCHD 포맷은, 기록 매체에 기록된 AV스트림을, 클립 단위로 플레이리스트를 이용하여 관리가 가능하도록 하고 있다.

[0077] 예를 들면, ITU-T(International Telecommunication Union-Telecommunication Standardization Sector) 권고 H.264 혹은 ISO(International Organization for Standardization)/IEC(International Electrotechnical Commission) 국제 표준 14496-10(MPEG-4 파트 10) Advanced Video Coding(이하, H.264 | AVC로 약칭한다.)에 규정되는 인코딩 방식으로 인코딩되고, MPEG2 시스템에 따라 다중화된 비트 스트림은, 클립 AV스트림(또는 AV스트림)이라고 칭해진다.

[0078] 클립 AV스트림은, 소정의 파일 시스템에 의해 파일로서 디스크에 기록된다. 이 파일을, 클립 AV스트림 파일(또는 AV스트림 파일)로 칭한다. 클립 AV스트림 파일은, 파일 시스템상에서의 관리 단위이며, 이용자에게 있어서 반드시 알기 쉬운 관리 단위라고는 할 수 없다. 이용자의 편리성을 생각했을 경우, 복수의 클립 AV스트림 파일에 분할된 영상 콘텐츠를 하나로 정리해 재생하는 구조나, 클립 AV스트림 파일의 일부만을 재생하는 구

조, 또, 특수 재생이나, 큐(cue)재생 등을 용이하게 실시하기 위한 정보 등을 데이터 베이스로서 디스크에 기록해 둘 필요가 있다.

- [0079] 도 2는, 이 발명에 적용 가능한 AVCHD 포맷에 규정되는 데이터 모델을 개략적으로 나타낸다. AVCHD 포맷에 의하면, 데이터 구조는, 도 2에 도시한 바와 같이, 4층을 포함한다. 가장 최하층은, 클립 AV스트림이 배치되는 층이다(편의상, 클립층이라고 부른다). 클립층 다음으로 두 번 째로 가장 낮은 층은, 클립 AV스트림에 대한 재생 부분을 지정하기 위한, 플레이 리스트(PlayList)와 플레이 아이템(PlayItem)이 배치되는 층이다(편의상, 플레이 리스트 층이라고 부른다). 플레이 리스트 다음으로 세 번째로 가장 낮은 층은, 플레이 리스트에 대해서, 재생 순서를 지정하는 명령을 포함하는 무비 오브젝트(Movie Object) 등이 배치되는 층이다(편의상, 오브젝트층이라고 부른다). 최상층은, 기록 매체에 저장되는 타이틀 등을 관리하는 인덱스 테이블이 배치되는 층이다(편의상, 인덱스층이라고 부른다).
- [0080] 클립 층에 대해 설명한다. 클립 AV스트림은, 비디오 데이터나 오디오 데이터가 MPEG2 TS(트랜스포트 스트림)의 형식 등으로 다중화된 비트 스트림이다. 이 클립 AV스트림에 관한 정보가 클립 정보(Clip Information)로서 파일에 기록된다.
- [0081] 또, 클립 AV스트림에 의해, 자막을 표시하는 그래픽스 스트림인 OB 스트림(Overlay Bitmap stream) 및/또는, 메뉴 표시 등에 이용되는 데이터(버튼 화상 데이터 등)스트림이 되는 MB스트림(Menu Bitmap stream) 스트림을 다중화할 수 있다.
- [0082] 클립 AV스트림 파일과 대응하는 클립 정보가 기록된 클립 정보 파일을 통합 오브젝트라고 간주하며, 클립(Clip)으로 칭한다. 즉, 클립은, 클립 AV스트림과 클립 정보를 포함하는 하나의 오브젝트이다.
- [0083] 파일은, 일반적으로, 바이트 순서로서 간주된다. 클립 AV스트림 파일의 콘텐츠는, 시간축 상에서 전개되며, 클립층의 엔트리 포인트는, 주로 시간으로 지정된다. 소정의 클립에 대한 액세스 포인트가 타임 스탬프로 정해진 경우, 클립 AV스트림 파일내에서 데이터의 읽기를 개시하는 어드레스 정보를 찾아내기 위해서, 클립 정보 파일을 이용할 수 있다.
- [0084] 플레이 리스트 층에 대해 설명한다. 플레이 리스트는, 재생하는 AV스트림 파일의 지정과, 지정된 AV스트림 파일의 재생 부분을 지정하는 재생 개시점(IN점)과 재생 종료점(OUT점)으로 구성된다. 재생 개시점과 재생 종료점의 정보의 1조는, 플레이 아이템(PlayItem)이라고 칭해진다. 플레이 리스트는, 플레이 아이템의 집합으로 구성된다. 플레이 아이템을 재생한다고 하는 것은, 그 플레이 아이템에 참조되는 AV스트림 파일의 일부분을 재생한다고 하는 것이 된다. 즉, 플레이 아이템중의 IN점 및 OUT점 정보에 근거하여, 클립의 대응하는 구간이 재생된다.
- [0085] 오브젝트 층에 대해 설명한다. 무비 오브젝트는, 내비게이션 명령 프로그램과 다른 무비 오브젝트와 협력하는 터미널 정보를 포함한다. 내비게이션 프로그램은, 플레이 리스트의 재생을 제어하기 위한 명령(내비게이션 명령: navigation command)이다.
- [0086] 인덱스 층에 대해 설명한다. 인덱스 층은, 인덱스 테이블(Index Table)을 포함한다. 인덱스 테이블은, 기록 매체에 기록된 콘텐츠의 타이틀을 정의하는, 톱 레벨의 테이블이다. 인덱스 테이블에 저장되어 있는 타이틀 정보에 근거하여, 재생기에 배치된 시스템 소프트웨어의 모듈 매니저에 의해 기록 매체의 재생이 제어된다.
- [0087] 즉, 도 3에 개략적으로 도시된 바와 같이, 인덱스 테이블의 임의의 엔트리는, 타이틀이라고 칭해지며, 인덱스 테이블의 엔트리가 되는 퍼스트 플레이백 타이틀(FirstPlaybackTitle), 메뉴 타이틀(MenuTitle) 및 무비 타이틀(MovieTitle)(#1, #2, ...)은, 모두 타이틀이다. 각 타이틀은, 무비 오브젝트에 대한 링크를 나타낸다.
- [0088] 이해를 용이하게 하기 위해서, 재생 전용의 기록 매체를 예를 들면, 퍼스트 플레이백 타이틀은, 해당 기록 매체에 저장되는 콘텐츠가 영화이면, 영화 본편에 앞서 영화에 나오는 영화 회사의 광고용 영상(예를 들면, 영화 예고편)에 대응한다. 메뉴 타이틀은, 예를 들면 콘텐츠가 영화인 경우, 본편 재생, 챕터 검색, 자막이나 언어 설정, 특정 영상 재생 등을 선택하기 위한 메뉴 화면에 대응한다. 또, 무비 타이틀은, 메뉴 타이틀로부터 선택되는 각 영상이다. 즉, 메뉴 타이틀은, 타이틀도 포함하는 메뉴 화면에 대응한다.
- [0089] 도 4는, 클립 AV스트림, 클립 정보(Stream Attributes), 클립, 플레이 아이템 및 플레이 리스트의 관계를 나타내는 UML(Unified Modeling Language)의 도면이다. 플레이 리스트는, 한 개 또는 복수의 플레이 아이템에 대응하며 한 개 또는 복수의 플레이 아이템은, 한 개의 클립에 대응한다. 한 개의 클립에 대응하여, 각각 개시점 및 / 또는 종료점이 다른 복수의 플레이 아이템을 할당할 수 있다. 한 개의 클립에 의해 한 개의 클립 AV

스트림 파일이 참조된다. 이와 같이, 한 개의 클립에 의해 한 개의 클립 정보 파일이 참조된다. 또, 클립 AV스트림 파일과 클립 정보 파일이란, 1대 1 대응 관계를 가진다. 이러한 구조를 정의함으로써, 클립 AV스트림 파일을 변경하는 일없이, 임의의 부분 만큼을 재생하는 재생 순서가, 비파괴적으로 지정될 수 있다.

[0090] 또, 도 5와 같이, 복수의 플레이 리스트로부터 동일한 클립을 참조할 수도 있다. 또, 한 개의 플레이 리스트로부터 복수의 클립을 지정할 수도 있다. 클립은, 플레이 리스트중의 플레이 아이템에 나타나는 IN점 및 OUT점에 의해, 참조된다. 도 5의 예에서는, 클립(300)은, 플레이 리스트 (310)의 플레이 아이템(320)으로부터 참조됨과 동시에, 플레이 리스트(311)를 구성하는 플레이 아이템(321)으로부터 참조되며, 클립(300)의 구간은, 플레이 아이템(321)내에 포함되는 IN점 및 OUT점에 관한 정보에 근거하여 플레이 리스트(311)를 구성하는 플레이 아이템(321, 322)중 플레이 아이템(321)에 의해 참조된다. 마찬가지로, 클립(301)은, 플레이 아이템(322)내에 포함되는 IN점 및 OUT점에 관한 정보에 근거하여 플레이 리스트(311)를 구성하는 플레이 아이템(322)에 의해 참조된다. 게다가, 클립(301)의 구간은, 플레이 아이템(323)내에 포함되는 IN점 및 OUT점에 관한 정보에 근거하여 플레이 리스트(312)를 구성하는 플레이 아이템(323, 324)중 플레이 아이템(323)에 의해 참조된다. 도 5의 예에서, 클립(301)은, 다른 플레이 리스트에 의해 참조되고 있다.

[0091] 다음에, AVCHD 포맷으로, 기록 매체에 기록되는 파일의 관리 구조에 대해서, 도 6을 이용해 설명한다. 파일은, 디렉토리 구조에 의해 계층적으로 관리된다. 기록 매체상에는, 먼저, 한 개의 디렉토리(도 6의 예에서는 루트(root) 디렉토리)가 작성된다. 이 루트 디렉토리 하부의 계층적인 범위를 한 개의 기록/재생 시스템으로 관리되는 범위로 한다.

[0092] 루트 디렉토리 아래에, 디렉토리 "BDMV"가 배치된다. 게다가, 필요에 따라서 동일한 층내에 디렉토리 "AVCHDTN"가 배치된다. 디렉토리"AVCHDTN"에는, 예를 들면 클립의 대표 화상을 소정 사이즈로 축소한 썸네일 파일(thumbnail file)일이 배치된다. 디렉토리 "BDMV"에는, 도 2를 이용해 설명한 데이터 구조가 저장된다.

[0093] 디렉토리 "BDMV"의 아래에는, 파일 "index.bdmv" 및 파일 "MovieObject.bdmv"의 2개만이 배치될 수 있다. 또, 디렉토리 "BDMV" 아래에는, 디렉토리 "PLAYLIST", 디렉토리 "CLIPINF", 디렉토리 "STREAM" 및 디렉토리 "BACKUP"가 놓여진다. 디렉토리 "BACKUP"에는, 각 디렉토리 및 파일의 백업이 저장된다.

[0094] 파일 "index.bdmv"는, 디렉토리 "BDMV"의 내용에 대해 기술된다. 즉, 이 파일 "index.bdmv"는, 데이터 구조 내에서 최상층이 되는 인덱스 층에 있어서의 인덱스 테이블과 대응한다. 또, 파일 "MovieObject.bdmv"에는, 1개 이상의 무비 오브젝트의 정보가 저장된다. 즉, 이 파일 "MovieObject.bdmv"가 상술한 오브젝트 층에 대응한다.

[0095] 디렉토리 "PLAYLIST"는, 플레이 리스트의 데이터 베이스가 배치되는 디렉토리이다. 즉, 디렉토리 "PLAYLIST"는, 플레이 리스트에 관한 파일인 파일 "xxxxx.mpls"를 포함한다. 파일 "xxxxx.mpls"는, 플레이 리스트의 각각에 대해 작성되는 파일이다. 파일명에서, "."(피리어드)의 전의"xxxxx"는, 5자리수의 숫자로 간주되며, 피리어드의 뒤의 "mpls"는, 이 타입의 파일에 고정된 확장자(extension)이다.

[0096] 디렉토리 "CLIPINF"는, 클립의 데이터 베이스가 배치되는 디렉토리이다. 즉, 디렉토리 "CLIPINF"는, 각 클립 AV스트림 파일에 대응하는 클립 정보 파일인 파일"zzzzz.clpi"를 포함한다. 각 클립 정보 파일명에서는, "."(피리어드)의 전의 "zzzzz"는, 5자리수의 숫자로 간주되며, 피리어드의 뒤의 "clpi"는, 이 타입의 파일에 고정된 확장자(extension)이다.

[0097] 디렉토리 "STREAM"는, 실체로서의 AV스트림 파일이 배치되는 디렉토리이다. 즉, 디렉토리 "STREAM"는, 클립 정보 파일의 각각 대응하는 클립 AV스트림 파일을 포함한다. 각 클립 AV스트림 파일은, MPEG2(Moving Pictures Experts Group 2)의 트랜스포트 스트림(이하, MPEG2 TS로 약칭한다.)으로 구성되며, 파일명 "zzzzz.m2ts"을 가진다. 각 클립 AV 스트림 파일명에서는, 피리어드 "."의 이전의 "zzzzz"는, 대응하는 클립 정보 파일과 동일하게 만들어지며, 클립 정보 파일과 이 클립 AV스트림 파일과의 대응 관계를 용이하게 파악할 수 있다.

[0098] 디렉토리 "AVCHDTN"는, 2 종류의 썸네일 파일 "thumbnail.tidx" 및 "thumbnail.tdt2"가 배치될 수 있다. 썸네일 파일 "thumbnail.tidx"에는, 소정의 방식으로 암호화된 썸네일 화상이 저장된다. 썸네일 파일 "thumbnail.tdt2"에는, 암호화되어 있지 않은 썸네일 화상이 저장된다. 예를 들면, 비디오 카메라로 이용자가 촬영한 클립에 대응하는 썸네일 화상은, 카피 프리이며(copy free), 암호화할 필요가 없다고 생각되므로, 이 썸네일 파일 "thumbnail.tdt2"에 저장된다.

[0099] 도 6에 도시된 각 파일 가운데, 이 발명의 실시예들과 깊은 관련이 있는 것들에 대해서, 보다 상세하게 설명한

다. 먼저, 디렉토리 "BDMV"의 하부에 배치되는 파일 "index.bdmv"에 대해 설명한다. 도 7은, 이 파일 "index.bdmv"의 일례의 구조를 나타내는 선택스(syntax)를 나타낸다. 여기에서, 선택스는 컴퓨터 장치 등의 프로그램 기술 언어로서 이용되는 C언어의 기술법에 근거하여 도시되었다. 이것은, 다른 선택스를 나타내는 도면에도 적용된다.

- [0100] 도 7을 참조하면, 필드 "TypeIndicator"는, 32비트의 데이터 길이를 가지며, 이 파일이 인덱스 테이블인 것을 나타낸다. 필드 "TypeIndicator2"는, 32비트의 데이터 길이를 가지며, 이 파일 "index.bdmv"의 버전을 나타낸다. 필드 "IndexesStartAddress"는, 32비트의 데이터 길이를 가지며, 이 선택스내에 있는 블록 "blkIndexes()"의 개시 어드레스를 나타낸다.
- [0101] 필드 "IndexsStartAddress"는, 32비트의 데이터 길이를 가지며, 이 선택스내에 있는 블록 "blkExtensionData()"의 개시 어드레스를 나타낸다. 블록 "blkExtensionData()"은 소정의 데이터가 저장될 수 있는 블록이다. 필드 "IndexsStartAddress"는, 파일 "index.bdmv"의 최초의 바이트로부터 상대 바이트의 수에 의해 블록 "blkExtensionData()"의 개시 어드레스를 나타낸다. 바이트의 상대적인 수는 "0"에서 개시한다. 만약 필드 ExtensionDataStartAddress의 값이 "0"이라면, 이것은 블록 "blkExtensionData()"가 파일 "index.bdmv"내에 존재하지 않다는 것을 의미한다.
- [0102] 필드 "ExtensionDataStartAddress"에 이어, 데이터 길이가 192바이트의 영역 "reserved"가 배치된다. 영역 "reserved"는, 바이트 얼라인먼트(byte alignment)혹은, 장래적인 필드의 추가 등을 위한 영역이다. 이것은, 이하의 설명에 대해서도 동일하게 적용한다. 블록 "blkA ppInfoBDMV()"는, 콘텐츠 제작자가 임의의 정보를 기술할 수 있는 블록이며, 플레이어의 동작 등에는 영향을 주지 않는다.
- [0103] 블록 "blkIndexes()"는, 이 파일 "index.bdmv"의 실질적인 내용이며, 이 파일 "index.bdmv"에 기술된 내용에 의해, 디스크를 플레이어에 설치했을 때에 재생되는 퍼스트 플레이백 타이틀(FirstPlaybackTitle) 과, 톱 메뉴 상에 호출되는 타이틀(MovieObject)이 지정된다. 인덱스 테이블에 의해 호출된 무비 오브젝트 등에 기술된 명령에 근거하여, 후술하는 플레이 리스트 파일이 읽힌다.
- [0104] 도 8은, 블록 "blkIndexes()"의 일례의 구조를 나타내는 선택스를 나타낸다. 필드 "Length"는, 32비트의 데이터 길이를 가지며, 이 필드 "Length" 직후부터 이 블록 "blkIndexes()"의 마지막까지의 데이터 길이를 나타낸다. 그 뒤에, 블록 "FirstPlaybackTitle()" 및 블록 "MenuTitle()"이 배치된다.
- [0105] 블록 "FirstPlaybackTitle()"은, 퍼스트 플레이백으로 이용되는 오브젝트에 관한 정보가 기술된다. 블록 "FirstPlaybackTitle()"은, 1비트의 데이터 길이를 가지는 영역 "reserved"에 이어 고정치 "1"이 기술된다. 게다가 31비트의 데이터 길이를 가지는 영역 "reserved"를 통해 고정치 "1"이 기술된다. 그리고, 14비트의 데이터 길이를 가지는 영역 "reserved"를 통해, 16비트의 데이터 길이를 가지는 필드 "FirstPlaybackTitleMobjIDRef"가 배치된다. 이 필드 "FirstPlaybackTitleMobjIDRef"는, 퍼스트 플레이백 타이틀로 이용되는 무비 오브젝트의 ID를 나타낸다.
- [0106] 각 무비 오브젝트의 ID는, 예를 들면, 도 9 및 도 10을 참조하여 후술하는 무비 오브젝트의 선택스에 근거하여, 무비 오브젝트의 for-loop 문장내의 루프 변수로서 이용되는 값 "mobj_id"에 의해 표시된다. 도 8의 예에서, 필드 "FirstPlaybackTitleMobjIDRef"에는, 참조하는 무비 오브젝트에 대응하는 값 "mobj_id"가 저장된다.
- [0107] 블록 "blkIndexes()"에 있어서의 블록 "FirstPlaybackTitle()"내의 필드 "FirstPlaybackTitleMobjIDRef"는, 톱 메뉴의 무비 오브젝트를 가리키거나, 타이틀을 가리킨다.
- [0108] 블록 "MenuTitle()"은, 톱 메뉴로 이용되는 오브젝트에 관한 정보가 기술된다. 블록 "MenuTitle()"은, 1비트의 데이터 길이를 가지는 영역 "reserved"의 뒤에 고정치 "1"이 기술된다. 게다가, 31비트의 데이터 길이를 가지는 영역 "reserved"를 통해 고정치 "1"이 기술된다. 그리고, 14비트의 데이터 길이를 가지는 영역 "reserved"를 통해, 16비트의 데이터 길이를 가지는 필드 "MenuTitleMobjIDRef"가 배치된다.
- [0109] 필드 "MenuTitleMobjIDRef"는, 메뉴 타이틀로 이용되는 무비 오브젝트의 ID를 나타낸다. 블록 "MenuTitle()"의 다음의 필드 "NumberOfTitles"는, 16비트의 데이터 길이를 가지며, 이용자가 선택, 재생 가능한 타이틀의 수를 나타낸다. 다음의 for-loop문에 따라, 이 필드 "NumberOfTitles"에 의해 나타나는 횟수만큼, 값 "title_id"를 이용하여, 블록 "MovieTitle[title_id]()"이 인수로서, 기술된다. 블록 "MovieTitle[title_id]()"은, 타이틀 마다의 정보가 기술된다. 값 "title_id"은, "0"으로부터 필드 "NumberOfTitles"로 나타나는 값까지의 수치이며, 타이틀을 식별한다.

- [0110] 블록 "MovieTitle[title_id]()"에서는, 1비트의 데이터 길이를 가지는 영역 "reserved"를 통해 고정치 "1"이 기술되며, 게다가 46비트의 데이터 길이를 가지는 영역 "reserved"를 통해 필드 "MovieTitleMobjIDRef"가 기술된다. 필드 "MovieTitleMobjIDRef"는, 16비트의 데이터 길이를 가지며, 이 타이틀을 위해 이용되는 무비 오브젝트의 ID를 나타낸다. 필드 "MovieTitleMobjIDRef"에 이어서, 32비트의 데이터 길이를 가지는 영역 "reserved"이 배치된다.
- [0111] 도 9는, 디렉토리 "BDMV"의 하부에 배치되는 파일 "MovieObject.bdmv"의 일례의 구조를 나타내는 선택스를 나타낸다. 필드 "TypeIndicator"는, 32비트(4바이트)의 데이터 길이를 가지며, 이 파일이 파일 "MovieObject.bdmv"인 것을 나타낸다. 필드 "TypeIndicator"에는, ISO(International Organization for Standardization) 646에 규정된 인코드 방식으로 인코드 한 4문자를 포함하는 문자열이 기술된다. 이 도 9의 예에서, 필드 "type_indicator"에는, ISO646에 규정된 방식으로 인코드된 4문자의 문자열 "MOBJ"가 기술되며, 이 파일이 파일 "MovieObject.bdmv"이라는 것을 나타낸다.
- [0112] 필드 "TypeIndicator2"는, 32비트(4바이트)의 데이터 길이를 가지며, 이 파일 "MovieObject.bdmv"의 버전 번호를 나타낸다. 이 파일 "MovieObject.bdmv"에서, 필드 "TypeIndicator2"는, ISO646에 규정된 인코드 방식으로 인코딩한 4문자의 문자열 "0(100)"이 대표적으로 기술된다.
- [0113] 필드 "ExtensionDataStartAddress"는, 32비트의 데이터 길이를 가지며, 이 선택스내에 있는 블록 "blkExtensionData()"의 개시 어드레스를 나타낸다. 필드 "ExtensionDataStartAddress"는, 이 파일 "MovieObject.bdmv"의 최초의 바이트로부터의 상대 바이트수로, 블록 "blkExtensionData()"의 개시 어드레스를 나타낸다. 상대 바이트수는, "0"으로부터 개시된다. 이 필드 "ExtensionDataStartAddress"의 값이 "0"이면, 이 파일 "MovieObject.bdmv"내에, 블록 "blkExtensionData()"이 존재하지 않는 것을 가리킨다.
- [0114] 도 9에 도시된 선택스내의 필드 "padding_word"는, 16비트의 데이터 길이를 가지며, 이 파일 "MovieObject.bdmv"의 선택스에 따라 for-loop문에 값(N1, N2)으로 나타나는 회수만큼 삽입된다. 값(N1 또는 N2)은, "0" 또는 임의의 정수의 정수이다. 또, 필드 "padding_word"는, 임의의 값을 이용할 수 있다.
- [0115] 필드 "ExtensionDataStartAddress"에 이어 데이터 길이가 224비트의 영역 "reserved"이 배치되며, 그 다음에, 이 파일 "MovieObject.bdmv"의 본체인 블록 "blkMovieObjects()"이 저장된다.
- [0116] 도 10은, 블록 "blkMovieObjects()"의 일례의 구조를 나타내는 선택스를 나타낸다. 필드 "Length"는, 32비트의 데이터 길이를 가지며, 이 필드 "Length"의 직후부터 이 블록 "blkMovieObjects()"의 마지막까지의 데이터 길이를 나타낸다. 32비트의 데이터 길이를 가지는 영역 "reserved"를 통해 필드 "NumberOfMobjs"가 배치된다. 필드 "NumberOfMobjs"는, 직후의 for-loop문에 따라 저장되는 무비 오브젝트의 수를 나타낸다. for-loop문의 루프 변수로서 이용되는 값 "mobj_id"는, 각 무비 오브젝트를 고유하게 특정한다. 값 "mobj_id"는, "0"으로부터 시작되는 값으로, 무비 오브젝트는, for-loop문중에 기술되는 순서에 의해 정의된다.
- [0117] for-loop문중의 블록 "TerminalInfo()"에는, 고정치 "1"이 기술되며, 다음에 15비트의 데이터 길이를 가지는 영역 ""reserved"가 배치된다. 그 다음에, 16비트의 데이터 길이를 가지는 필드 "NumberOfNavigationCommands[mobj_id]"가 배치된다. 이 필드 "NumberOfNavigationCommands[mobj_id]"는, 값 "mobj_id"에 의해서 지시되는 무비 오브젝트 "MovieObject[mobj_id]()"에 포함되는 내비게이션 명령(NavigationCommand)의 수를 나타낸다.
- [0118] 다음의, 값 "command_id"를 루프 변수로 하는 for-loop문에 의해, 필드 "NumberOfNavigationCommands[mobj_id]"에 나타나는 수만큼, 내비게이션 명령이 기술된다. 즉, for-loop문에 배치되는 필드 "NavigationCommand[mobj_id][command_id]"는, 값 "mobj_id"에 의해서 지시되는 블록 "MovieObject[mobj_id]()"에 포함되는, 값 "command_id"로 표시되는 각 순서내에 내비게이션 명령(NavigationCommand)을 저장한다. 값 "command_id"는, 0으로부터 시작되는 값으로, 내비게이션 명령들이, for-loop문내에 기술되는 순서로 정의된다.
- [0119] 도 11은, 플레이 리스트 파일 "xxxxx.mpls"의 일례의 구조를 나타내는 선택스를 나타낸다. 필드 "TypeIndicator"는, 32비트(4바이트)의 데이터 길이를 가지며, 이 파일이 플레이 리스트 파일인 것을 나타낸다. 필드 "TypeIndicator2"는, 32비트(4바이트)의 데이터 길이를 가지며, 이 플레이 리스트 파일의 버전을 나타낸다. 필드 "PlayListStartAddress"는, 32비트의 데이터 길이를 가지며, 이 선택스중의 블록 "blkPlayList()"의 개시 어드레스를 나타낸다.

- [0120] 필드 "PlayListMarkStartAddress"는, 32비트의 데이터 길이를 가지며, 이 선택스중의 블록 "blkPlayListMark()"의 개시 어드레스를 나타낸다. 필드 "ExtensionDataStartAddress"는, 32비트의 데이터 길이를 가지며, 이 선택스중의 블록 "blkExtensionData()"의 개시 어드레스를 나타낸다. 필드 "ExtensionDataStartAddress"의 값은, 블록 "blkExtensionData()"의 개시 어드레스를, 파일 "xxxxx.mpls"의 최초의 바이트로부터의 상대 바이트 수에 의해 나타낸 값이다. 상대 바이트수는, "0"으로부터 개시된다. 이 필드 "ExtensionDataStartAddress"의 값이 "0"이면, 이 파일 "xxxxx.mpls"내에, 블록 "blkExtensionData()"이 존재하지 않는 것을 가리킨다.
- [0121] 160비트의 데이터 길이를 가지는 영역 "reserved"를 통해 블록 "blkAppInfoPlayList()"이 배치된다. 블록 "blkAppInfoPlayList()"은, 다음의 블록 "blkPlayList()"에 기술되는 플레이 리스트의 타입, 재생 제한 등의 정보가 기술된다. 블록 "blkPlayList()"에는, 플레이 리스트가 기술된다. 블록 "blkPlayListMark()"에는, 챕터 등에 의한 점프 목적지가 기술된다. 블록 "blkExtensionData()"은, 소정의 확장 데이터가 저장될 수 있는 블록이다.
- [0122] 도 12는, 블록 "blkPlayList()"의 일례의 구조를 나타내는 선택스를 나타낸다. 필드 "Length"는, 32비트의 데이터 길이를 가지며, 이 필드 "Length"의 직후부터 블록 "blkPlayList()"의 끝까지의 데이터 길이를 나타낸다. 필드 "Length"에 이어 16비트의 데이터 길이를 가지는 영역 "reserved"이 배치되며, 다음에 필드 "NumberOfPlayItems"가 배치된다. 필드 "NumberOfPlayItems"는, 16비트의 데이터 길이를 가지며, 이 블록 "blkPlayList()"에 포함되는 플레이 아이템의 수를 나타낸다. 필드 "NumberOfSubPath"는, 이 블록 "blkPlayList()"에 포함되는 서브 패스의 수를 나타낸다.
- [0123] 다음의 for-loop문에 따라, 필드 "NumberOfPlayItems"로 표시되는 수만큼, 플레이 아이템이 기술되는 블록 "blkPlayItem()"이 기술된다. for-loop문에 근거하는 카운트 수가 블록 "blkPlayItem()"의 식별자 "PlayItem_id"가 된다. 다음의 for-loop문에 따라, 필드 "NumberOfSubPath"로 나타나는 수만큼, 블록 "blkSubPath()"이 기술된다. for-loop문에 근거하는 카운트 수가 블록 "blkSubPath()"의 식별자 "SubPath_id"가 된다.
- [0124] 서브 패스는, 주로 재생되는 플레이 아이템에 대응하는 메인 패스에 대해서, 서브 플레이 아이템에 대응하도록 배치될 수 있다. 서브 패스는, 예를 들면, 2개의 영상을 합성할 때에, 플레이 아이템으로 지정되는 클립과 동기하여 재생되는 부영상(sub-picture)이 된다.
- [0125] 도 13은, 블록 "blkPlayItem()"의 일례의 구조를 나타내는 선택스를 나타낸다. 필드 "Length"는, 16비트의 데이터 길이를 가지며, 이 필드 "Length"의 직후부터 블록 "blkPlayItem()"의 끝까지의 데이터 길이를 나타낸다.
- [0126] 필드 "ClipInformationFileName"는, 40비트(5바이트)의 데이터 길이를 가지며, 블록 "blkPlayItem()"에 의해 참조되는 클립 정보 파일의 파일명을 나타낸다. 이 블록 "blkPlayItem()"에는, 필드 "ClipInformationFileName"에 의해 표시되는 파일명내의 클립 정보 파일의 파일명이 관독된다. 필드 "ClipCodecIdentifier"는, 32비트(4바이트)의 데이터 길이를 가지며, 이 블록 "blkPlayItem()"에 의해 표시되는 플레이 아이템에 대해서 이용되는 클립 AV 트림의 코덱 방식을 나타낸다.
- [0127] 12비트의 데이터 길이를 가지는 영역 "reserved"를 통해, 필드 "ConnectionCondition"가 배치된다. 필드 "ConnectionCondition"는, 4비트의 데이터 길이를 가지며, 클립간의 접속 상태에 관한 정보를 나타낸다. 기록 용도의 기록 매체에 대해서는, 필드 "ConnectionCondition"의 값으로서 "1", "5" 또는 "6"이 이용된다. 필드 "ConnectionCondition"의 값이 "1"이면, 그 플레이 아이템에 의해 참조되고 있는 클립은 바로 이전의 플레이 아이템에 의해 참조되고 있는 클립과 심리스 접속을 하지 않고 있다는 것을 가리키며, 필드 "ConnectionCondition"의 값이 "5" 또는 "6"이라면, 그 플레이 아이템에 의해 참조되고 있는 클립과 앞의 플레이 아이템에 의해 참조되고 있는 클립이 심리스 접속하고 있다는 것을 나타낸다. 심리스 접속이란, 클립과 다음의 클립이 프레임 타이밍에 연속적으로 재생되도록, 클립간의 재생 제어를 실시하는 것을 말한다.
- [0128] 필드 "ConnectionCondition"의 값이 "5"라면, 해당 플레이 아이템이 참조하는 클립내에서, 오디오 데이터의 기록 길이가 비디오 데이터의 기록길이 보다 더욱 길게 설정된다(도 14a 참조). 이에 의해, 클립과 클립을 접속할 때에, 오디오 데이터의 페이드아웃(fadeout) 처리가 가능하게 된다. 예를 들면, 이용자에 의한 기록 정지 조작에 의해 클립이 클로즈(close)되는 경우에, 필드 "ConnectionCondition"의 값이 "5"로 설정된다. 이하, 이 필드 "ConnectionCondition"의 값이 "5"로 나타나는 클립의 접속 방법을, 제 1의 심리스 접속이라고 부른다.

- [0129] 필드 "ConnectionCondition"의 값이 "6"이라면, 해당 플레이 아이템이 참조하는 클립내에서, 오디오 데이터의 기록 길이가 비디오 데이터의 기록길이와 동일하게 설정된다(도 14b 참조). 이에 의해, 클립과 클립과의 접속을 심리스하게 실시하는 것이 가능하게 된다. 예를 들면, 이용자 조작에 대응하는 기록 정지 이외의 이유, 예를 들면 시스템 요인에 근거하여 클립이 클로уз 되는 경우에, 필드 "ConnectionCondition"의 값이 "6"으로 설정된다. 이하, 이 필드 "ConnectionCondition"의 값이 "6"으로 나타나는 클립의 접속 방법을, 제 2의 심리스 접속이라고 부른다.
- [0130] 필드 "RefToSTCID"는, 8비트의 데이터 길이를 가지며, 시스템 타임 베이스(STC)의 불연속점에 관한 정보를 나타낸다. 필드 "INTime" 및 필드 "OUTTime"는, 각각 32비트의 데이터 길이를 가지며, 메인 클립 AV스트림의 재생 범위를 나타낸다. 필드 "INTime"가 개시점(IN점)을 나타내며, 필드 "OUTTime"가 종료점(OUT점)을 나타낸다.
- [0131] 블록 "blkUOMaskTable()"은, 이용자 입력의 접속 제한이 설정되는 테이블이다. 1비트의 데이터 길이를 가지는 플래그 "PlayItemRandomAccessFlag"는, 이 블록 "blkPlayItem()"에 의해 표시되는플레이 아이템에 대해서 랜덤 액세스를 허가할 것인지 아닌지를 한정하는 것이다. 이어서, 7비트의 데이터 길이를 가지는 영역 "reserved"를 통해 필드 "StillMode"가 배치된다. 필드 "StillMode"는, 8비트의 데이터 길이를 가지며, 블록 "blkPlayItem()"에 의해 표시되는 플레이 아이템에서, 마지막에 표시한 영상을 정지화면으로 표시할 것인지 아닌지를 나타낸다. 필드 "StillMode"의 값이 "0x01"(바이너리)이면, if문에 근거하여, 16비트의 데이터 길이를 가지는 필드 "StillTime"에 의해 정지 시간이 표시된다. 필드 "StillMode"의 값이 "0x01"이외의 값이라면, 해당 16비트의 데이터 길이를 가지는 영역이 영역 "reserved"로 사용된다.
- [0132] 수치 표시 "0x"는, 그 수치가 16진수로 표시되었다는 것을 나타낸다.이것은, 이하의 동일한 표기에 대해 공통이다. 블록 "blkSTNTable()"은, 이 블록 "blkPlayItem()"에 의해 표시되는 플레이 아이템이 관리하고 있는 클립 AV스트림의 속성, PID 번호, 기록 매체상에서의 클립 AV스트림의 기록 위치등이 관리되는 테이블이다.
- [0133] 도 15는, 블록 "blkPlayListMark()"의 일례의 구조를 나타내는 선택스를 나타낸다. 필드 "Length"는, 32비트의 데이터 길이를 가지며, 이 필드 "Length"의 직후부터 블록 "blkPlayListMark()"의 끝까지의 데이터 길이를 나타낸다.
- [0134] 필드 "NumberOfPlayListMarks"는, 16비트의 데이터 길이를 가지며, 이 블록 "blkPlayListMark()"에 포함되는 플레이 리스트 마크의 수를 나타낸다. 다음의 for-loop문에 따라, 필드 "NumberOfPlayListMarks"로 표시되는 수만큼 플레이 리스트 마크의 정보가 기술된다.
- [0135] for-loop문내에서, 8비트의 데이터 길이를 가지는 영역 "reserved"에 이어 필드 "MarkType"가 배치된다. 필드 "MarkType"는, 8비트의 데이터 길이를 가지며, 마크의 타입을 나타낸다. 플레이 리스트 마크에는, 엔트리 마크(Entry Mark) 및 링크 포인트(Link Point)의 2 타입이 정의되고 있으며, 이 필드 "MarkType"에 의해, 어느 쪽의 타입인지를 알 수 있다. 챕터(chapter)를 정의하기 위해서는, 엔트리 마크를 이용한다. 링크 포인트는, 이 발명과 관련성이 별로 없기 때문에, 설명을 생략한다. 상술한 필드 "NumberOfPlayListMarks"는, 엔트리 마크 및 링크 포인트를 합계한 값을 나타낸다.
- [0136] 필드 "RefToPlayItemID"는, 16비트의 데이터 길이를 가지며, 마크가 주어지는 플레이 아이템을 참조하는데 이용되는 식별 정보 "PlayItem_id"가 기술된다. 필드 "MarkTimeStamp"는, 32비트의 데이터 길이를 가지며, 마크가 주어지는 포인트를 나타내는 타임 스탬프가 기술된다. 필드 "EntryESPID"는, 16비트의 데이터 길이를 가지며, 마크에 의해서 지시되는 엘리먼트리(elementary) 스트림을 포함하고 있는 TS패킷의 PID의 값을 나타낸다. 필드 "Duration"은, 45kHz의 클럭 구간들에서 측정되며, 32비트의 데이터 길이를 가지는 부호없는 정수이다. 이 필드 "Duration"에 저장되는 값이 "0"이면, 이 필드 "Duration"은, 의미가 없게 된다.
- [0137] 도 16은, 클립 정보 파일의 일례의 구조를 나타내는 선택스를 나타낸다. 필드 "TypeIndicator"는, 32비트(4바이트)의 데이터 길이를 가지며, 이 파일이 클립 정보 파일이라는 것을 나타낸다. 필드 "TypeIndicator2"는, 32비트(4바이트)의 데이터 길이를 가지며, 이 클립 정보 파일의 버전을 나타낸다.
- [0138] 이 클립 정보 파일은, 블록 "blkClipInfo()", 블록 "blkSequenceInfo()", 블록 "blkProgramInfo()", 블록 "blkCPI()", 블록 "blkClipMark()" 및 블록 "blkExtensionData()"을 가지며, 각각 32비트의 데이터 길이를 가지는 필드 "SequenceInfoStartAddress", 필드 "ProgramInfoStartAddress", 필드 "CPIStartAddress", 필드 "ClipMarkStartAddress" 및 필드 "ExtensionDataStartAddress"는, 각각 대응하는 블록의 개시 어드레스를 나타낸다.

- [0139] 필드 "ExtensionDataStartAddress"는, 이 클립 정보 파일의 최초의 바이트로부터의 상대 바이트 수에 의해, 블록 "blkExtensionData()"의 개시 어드레스를 나타낸다. 상대 바이트 수는, "0"으로부터 개시된다. 이 필드 "ExtensionDataStartAddress"의 값이 "0"이면, 이 파일 "index.bdmv"내에, 블록 "blkExtensionData()"이 존재하지 않는 것을 가리킨다.
- [0140] 블록 "blkClipInfo()"은, 이러한 개시 어드레스를 나타내는 필드에 계속 되는, 96비트의 데이터 길이를 가지는 영역 "reserved"의 다음으로부터 개시된다. 블록 "blkClipInfo()"내에는, 이 클립 정보 파일이 관리하는 클립 AV스트림에 관한 정보가 기술된다. 블록 "blkSequenceInfo()"내에는, STC나 ATC(arrival time base : 도착 시간 기준)가 통합된 순서로서 연속적인 순서를 관리하는 정보가 기술된다. 블록 "blkProgramInfo()"내에는, 이 클립 정보 파일에 의해 관리되는 클립 AV 스트림의 인코드 방식, 클립 AV스트림중의 비디오 데이터의 어스펙트비(aspect ratio) 등의 정보가 기술된다. 블록 "blkCPI()"내에는, 랜덤 액세스 개시점과 같은, AV스트림내의 특징적인 부분을 나타내는 특징점 정보("CPI")에 관한 정보가 저장된다.
- [0141] 또, 블록 "blkClipMark()"에는, 캡터 위치와 같은, 각 클립에 할당된 큐(cue) 재생을 위한 인덱스점(점프 포인트)이 기술된다. 블록 "blkExtensionData()"은, 확장 데이터를 저장할 수 있는 영역이다. 클립 정보 파일내의 블록 "blkClipMark()" 및 블록 "blkSequenceInfo()"은, 이 발명과의 관련성이 별로 없기 때문에, 상세한 설명을 생략한다.
- [0142] 도 17은, 클립 정보 파일에 있어서의 블록 "blkCPI()"의 일례의 구조를 나타내는 선택스를 나타낸다. MPEG 스트림과 같은, 프레임간 압축을 실시하고 있는 인코드 스트림에 대해서는, 디코드가 가능한 부분들이, 각 GOP(Group Of Picture)의 선두와 같은 부분들중 한 부분으로 종종 한정된다. CPI(Characteristic Point Information)란, 디코드가 가능한 개시점의 위치의 정보를 모은 데이터 베이스이며, 재생 시각과 파일내 어드레스가 대응되어 있는 테이블 형태이다. 즉, CPI는, 각 디코드 단위의 선두 위치를 나타내는 정보가 테이블 형태로 기록된 데이터 베이스이다.
- [0143] 이와 같이 데이터 베이스를 한정하여, 예를 들면, 임의의 시각부터 스트림을 재생하고 싶은 경우, 재생 시각을 바탕으로 CPI를 참조하여 재생 위치의 파일내 어드레스를 알 수 있다. 이 파일내 어드레스는, 디코드 단위의 선두가 되므로, 이용자는 플레이어를 이용하여, 그 어드레스의 데이터를 읽어내고 디코드하여, 재빠르게 화상을 표시할 수 있다.
- [0144] 이 CPI에 저장되는, 디코드 단위의 선두 위치(이 예에서는 GOP의 선두 위치)를, EP(Entry Point) 엔트리라고 칭한다.
- [0145] 도 17에서는, 필드 "Length"는, 32비트의 데이터 길이를 가지며, 이 필드 "Length"의 직후부터 블록 "blkCPI()"의 끝까지의 데이터 길이를 나타낸다. 다음의 if문에 따라, 필드 "Length"의 값이 0이 아니면, 12비트의 데이터 길이를 가지는 영역 "reserved"를 통해 필드 "CPIType"가 배치된다. 필드 "CPIType"는, 4비트의 데이터 길이를 가지며, CPI의 종류를 나타낸다. 다음의 블록 "blkEPMMap()"은, 대응하는 클립 AV스트림 파일에 있어서의 PTS치와 바이트 어드레스가 대응되는 관련성을 표시한 테이블을 저장한다.
- [0146] 도 18은, 블록 "blkEPMMap()"의 일례의 구조를 나타내는 선택스를 나타낸다. 8비트의 데이터 길이를 가지는 영역 "reserved"를 통해 필드 "NumberOfStreamPIDEntries"가 배치된다. 필드 "NumberOfStreamPIDEntries"는, 8비트의 데이터 길이를 가지며, 블록 "blkEPMMap()"에 있어서의 블록 "blkEPMMapForOneStreamPID"의 엔트리수를 나타낸다. for-loop에 따라, 값 [k]를 루프 변수로서 가지는 필드 "NumberOfStreamPIDEntries"에 표시되는 만큼, 엔트리 포인트에 관한 정보가 기술된다.
- [0147] for-loop문에서, 필드 "StreamPID[k]"는, 16비트의 데이터 길이를 가지며, 블록 "blkEPMMap()"내에서 [k]번째에 엔트리가 되는 블록 "blkEPMMapForOneStreamPID"(이하, [k]번째의 블록 "blkEPMMapForOneStreamPID"라고 기술한다)에 의해서 참조되는 엘리먼트리 스트림을 전송하는 트랜스포트 패킷의 PID의 값을 나타낸다.
- [0148] 10비트의 데이터 길이를 가지는 영역 "reserved"를 통해 필드 "EPStreamType[k]"가 배치된다. 필드 "EPStreamType[k]"는, 4비트의 데이터 길이를 가지며, [k]번째의 블록 "blkEPMMapForOneStreamPID"에 의해서 참조되는 엘리먼트리 스트림의 타입을 나타낸다. 필드 "NumberOfEPCoarseEntries[k]"는, 16비트의 데이터 길이를 가지며, [k]번째의 블록 "blkEPMMapForOneStreamPID"내에 있는 개략적인 검색용의 서브 테이블(EP coarse table)의 엔트리수를 나타낸다. 필드 "NumberOfEPFineEntries[k]"는, 18비트의 데이터 길이를 가지며, [k]번째의 블록 "blkEPMMapForOneStreamPID"안에 있는 정밀한 검색용의 서브 테이블(EP fine table)의 엔트리수를 나타낸다. 필드 "EPMMapForOneStreamPIDStartAddress[k]"는, 32비트의 데이터 길이를 가지며, 블록 "blkEPMMap()"내에

서 [k]번째의 블록 "blkEMapForOneStreamPID"이 시작되는 상대 바이트 위치를 나타낸다. 이 값은, 블록 "blkEMap()"의 제 1바이트째로부터의 바이트수에 의해 표시된다.

[0149] 상술의 for-loop문에 의해 기술된 뒤에, 16비트의 정수배의 데이터 길이를 가지는 필드 "padding_word"에 걸쳐 기술된 for-loop문에 따라, 값[k]를 루프 변수로서 사용하는 필드 "NumberOfStreamPIDEntries"에 의해 표시된 수만큼, 블록 "blkEMapForOneStreamPID(EPStreamType[k], NumberOfEPCoarseEntries[k], NumberOfEPFineEntries[k])"이 저장된다. 즉, 인수 "NumberOfEPCoarseEntries[k]"는, 서브 테이블(EP coarse table)에 저장되는 엔트리 "PTSEPCoarse" 및 엔트리 "SPNEPCoarse"의 수를 나타낸다. 이와 같이, 인수 "NumberOfEPFineEntries[k]"는, 서브 테이블(EP fine table)에 저장되는 엔트리 "PTSEPFine" 및 엔트리 "SPNEPFine"의 수를 나타낸다. 이하에서는, 인수 "NumberOfEPCoarseEntries[k]" 및 인수 "NumberOfEPFineEntries[k]"를, 각각 적절하게 엔트리수 "Nc" 및 엔트리수 "Nf"라고 부른다.

[0150] 도 19는, 블록 "blkEMapForOneStreamPID(EP_stream_type, Nc, Nf)"의 일례의 구조를 나타내는 신택스를 나타낸다. 블록 "blkEMapForOneStreamPID(EP_stream_type, Nc, Nf)"의 시멘틱스(semantics)를 설명하기 위해서, 먼저, 블록 "blkEMapForOneStreamPID(EP_stream_type, Nc, Nf)"에 데이터가 저장되는 근거가 되는 엔트리가 되는, 엔트리 "PTSEPStart" 및 엔트리 "SPNEPStart"의 의미에 대해 설명한다.

[0151] 엔트리 "SPNEPStart"와 관련된 엔트리 "PTSEPStart"와 엔트리 "PTSEPStart"는, 각각 AV스트림상의 엔트리 포인트를 가리킨다. 그리고, 엔트리 "PTSEPFine"와 관련된 엔트리 "PTSEPFine"과 엔트리 "PTSEPCoarse"는, 동일한 엔트리 "PTSEPStart"로부터 유도된다. 또, 엔트리 SPNEPFine과 관련된 엔트리 "SPNEPFine"과 엔트리 "SPNEPCoarse"는, 동일한 엔트리 "SPNEPStart"로부터 유도된다.

[0152] 도 20은, 엔트리 "PTSEPCoarse" 및 엔트리 "PTSEPFine"의 일례의 포맷에 대해 가리킨다. PTS, 즉 엔트리 "PTSEPStart"는, 데이터 길이가 33비트의 값이다. MSB의 비트를 제 32비트, LSB의 비트를 제 0비트로 할 때, 이 도 20의 예에서는, 개략적인 검색을 실시할 때에 이용되는 엔트리 "PTSEPCoarse"는, 엔트리 "PTSEPStart"의 제 32비트로부터 제 19비트까지의 14비트가 이용된다. 엔트리 "PTSEPCoarse"에 의해, 해상도가 5.8초로, 26.5 시간까지의 범위에서 검색이 가능하다. 또, 보다 정밀한 검색을 행하기 위한 엔트리 "PTSEPFine"는, 엔트리 "PTSEPStart"의 제 19비트로부터 제 9비트까지의 11비트가 이용된다. 엔트리 "PTSEPFine"에 의해, 해상도가 5.7밀리초와, 11.5초까지의 범위에서 검색이 가능하다. 제 19비트는, 엔트리 "PTSEPCoarse"와 엔트리 "PTSEPFine"에 공통적으로 이용된다. 또, LSB측의 제 0비트로부터 제 8비트까지의 9비트는, 엔트리 "PTSEPCoarse"와 엔트리 "PTSEPFine"를 위해 이용되지 않는다.

[0153] 도 21은, 엔트리 "SPNEPCoarse" 및 엔트리 "SPNEPFine"의 일례의 포맷에 대해 나타내고 있다. 소스 패킷 번호, 즉 엔트리 "SPNEPStart"는, 데이터 길이가 32비트의 값이다. MSB의 비트를 제 31비트, LSB의 비트를 제 0비트로 할 때, 이 도 21의 예에서는, 개략적인 검색을 실시할 때에 이용되는 엔트리 "SPNEPCoarse"는, 엔트리 "SPNEPStart"의 제 31비트로부터 제 0비트까지의 모든 비트가 이용된다. 또, 보다 정밀한 검색을 행하기 위한 엔트리 "SPNEPFine"는, 엔트리 "SPNEPStart"의 제 16비트로부터 제 0비트까지의 17비트가 이용된다. 엔트리 "SPNEPFine"에 의해, 예를 들면 거의 25MB(Mega Byte)의 AV스트림 파일까지의 범위내에서, 검색이 가능하다.

[0154] 소스 패킷 번호의 경우에서도, 엔트리 "SPNEPCoarse"에 대해 MSB측의 소정 비트수의 값만 이용하도록 해도 괜찮다. 예를 들면, 엔트리 "SPNEPCoarse"에 대해서는, 엔트리 "SPNEPStart"의 제 31비트로부터 제 16비트까지의 17비트를 이용하고, 엔트리 "SPNEPFine"에 대해서는, 엔트리 "SPNEPStart"의 제 16비트로부터 제 0비트까지의 17비트를 이용한다.

[0155] 상기에 의해, 엔트리 "PTSEPStart" 및 엔트리 "SPNEPStart"는, 다음과 같이 정의된다.

[0156] 엔트리 "PTSEPStart"는, 도 20에 도시된 바와 같이, 데이터 길이가 33비트의 부호 없음 정수이며, AV스트림중에서, 랜덤 액세스가 가능한 픽처(예를 들면 IDR(Instantaneous Decoding Refresh) 픽처나 I(Intra) 픽처)로부터 개시하는 비디오 액세스 유니트내의 33비트의 데이터 길이를 가지는 PTS를 나타낸다.

[0157] 엔트리 "SPNEPStart"는, 도 21에 도시된 바와 같이, 32비트의 부호 없음 정수이며, 엔트리 "PTSEPStart"와 관련된 비디오 액세스 유니트의 제 1바이트째를 포함하는 소스 패킷중, AV스트림내의 어드레스를 나타낸다. 엔트리 "SPNEPStart"는, 소스 패킷 번호의 단위로 표현되며, 값 "0"을 초기치로 설정한 상태에서, AV스트림 파일중 최초의 소스 패킷으로부터, 소스 패킷마다 1씩 증가하는 값으로 카운트된다.

[0158] 도 19를 참조하면, 블록 "blkEMapForOneStreamPID(EP_stream_type, Nc, Nf)"은, 제 1의 for-loop문에 의해 개략적인 검색을 행하기 위한 서브 테이블(EP coarse table)이 기술되며, 제 2의 for-loop문에 의해 정밀 검색을

위한 서브 테이블(EP fine table)이 기술된다.

- [0159] 제 1의 for-loop문의 바로 직전에, 필드 "EPFineTableStartAddress"가 배치된다. 필드 "EPFineTableStartAddress"는, 32비트의 데이터 길이를 가지며, 제 2의 for-loop에 있어서의 필드 "reservedEPFine[EP_fine_id]"의 제 1바이트의 개시 어드레스를, 블록 "blkEPMAPForOneStreamPID(EP_stream_type, Nc, Nf)"의 제 1바이트로부터의 상대 바이트수로 나타낸다. 상대 바이트수는, 값 "0"으로부터 개시한다.
- [0160] 제 1의 for-loop문은, 루프 변수[i]를 이용하여, 서브 테이블(EP coarse table)의 엔트리수(Nc)까지 반복되므로, 엔트리수(Nc)의 수만큼 필드 "RefToEPFineID[i]", 엔트리 "PTSEPCoarse[i]" 및 엔트리 "SPNEPCoarse[i]"가 저장된다. 제 1의 for-loop문에서, 필드 "RefToEPFineID[i]"는, 18비트의 데이터 길이를 가지며, 필드 "RefToEPFineID[i]"에 계속되는 필드 "PTSEPCoarse[i]"가 가리키는 엔트리 "PTSEPCoarse"와 관련된 엔트리 "PTSEPFine"를 가지는 서브 테이블(EP fine table)내의 엔트리 번호를 나타낸다. 엔트리 "PTSEPFine"와 이 엔트리 "PTSEPFine"와 관련된 엔트리 "PTSEPCoarse"는, 동일한 엔트리 "PTSEPStart"로부터 유추된다. 필드 "RefToEPFineID[i]"는, 제 2의 for-loop문에 기술되는 순서에 정의된 루프 변수 [EP_fine_id]의 값이 주어진다.
- [0161] 제 1의 for-loop문의 뒤에, 패딩 워드 "padding_word"를 사이에 두고 제 2의 for-loop문에 의한 기술이 이루어진다. 제 2의 for-loop문은, 루프 변수[EP_fine_id]를 이용하여, 서브 테이블("EPfine table")의 엔트리수(Nf)까지 반복된다. 엔트리수(Nf)의 수만큼, 1비트의 데이터 길이를 가지는 필드 "reservedEPFine[EP_fine_id]"와 3비트의 데이터 길이를 가지는 필드 "IEndPositionOffset[EP_fine_id]"와 11비트의 데이터 길이를 가지는 필드 "PTSEPFine[EP_fine_id]"와 17비트의 데이터 길이를 가지는 필드 "SPNEPFine[EP_fine_id]"가 저장된다. 이러한 필드들중에서, 필드 "PTSEPFine[EP_fine_id]" 및 필드 "SPNEPFine[EP_fine_id]"는, 루프 변수[EP_fine_id]에 근거하여 서브 테이블(EPfine table)로부터 참조되는 엔트리 "PTSEPFine" 및 엔트리 "SPNEPFine"로서 각각 저장된다.
- [0162] 엔트리 "PTSEPCoarse" 및 엔트리 "PTSEPFine", 및, 엔트리 "SPNEPCoarse" 및 엔트리 "SPNEPFine"는, 다음과 같이 유도된다.
- [0163] 서브 테이블(EPfine table)내의 각 관련된 데이터 "SPNEPStart"의 값이 증가하는 순서대로 Nf개의 엔트리가 있다고 가정한다. 각각의 엔트리 "PTSEPFine"는, 대응하는 엔트리 "PTSEPStart"로부터, 식(1)과 같이 유도된다.
- [0164] 식 (1)
- [0165]
$$PTSEPFine[EP_fine_id] = (PTSEPStart[EP_fine_id] \gg 9) / 2^{11} \dots\dots\dots (1)$$
- [0166] 엔트리 "PTSEPCoarse"와 대응하는 엔트리 "PTSEPFine"와의 관계는, 식(2)와 (3)과 같다.
- [0167]
$$PTSEPCoarse[i] = (PTSEPStart[RefToEPFineID[i]] \gg 19) / 2^{14} \dots\dots\dots (2)$$
- [0168]
$$PTSEPFine[RefToEPFineID[i]] = (PTSEPStart[RefToEPFineID[i]]$$
- [0169]
$$\gg 9) / 2^{11} \dots\dots\dots (3)$$
- [0170] 각각의 엔트리 "SPNEPFine"는, 대응하는 엔트리 "SPNEPStart"로부터, 식(4)와 같이 유도된다.
- [0171]
$$SPNEPFine[EP_fine_id] = SPNEPStart[EP_fine_id] / 2^{17} \dots\dots\dots (4)$$
- [0172] 엔트리 "SPNEPCoarse"와 대응하는 엔트리 "SPNEPFine"와의 관계는, 식(5)와, (6)과 같다.
- [0173]
$$SPNEPCoarse[i] = SPNEPStart[RefToEPFineID[i]] \dots\dots\dots (5)$$
- [0174]
$$SPNEPFine[RefToEPFineID[i]]$$
- [0175]
$$= SPNEPStart[RefToEPFineID[i]] / 2^{17} \dots\dots\dots (6)$$
- [0176] 상술의 식(1)~(6)에서, 기호 ">>x"는, 데이터의 LSB측으로부터 x비트를 넘는 자리수로부터 비트를 이용하는 것을 의미한다.
- [0177] 다음에, 이 발명의 실시의 한 형태에 대해 설명한다.

- [0178] 이 발명에서는, 플레이 리스트에 대해서 플레이 리스트 마크가 할당되며, 클립의 분할 편집을 실시할 때에, 플레이 리스트 마크가 할당된 위치에 대응하는 GOP가 클로즈드 GOP라면, 어드레스 정보, 즉 GOP 경계에 플레이 리스트 마크를 할당하고, 해당 GOP가 오픈 GOP이면, 재생 시각 정보, 즉 상술한 엔트리 "PTSEStart"(도 21 참조)에 의해 표시되는 위치에 플레이 리스트 마크를 할당한다.
- [0179] 예를 들면, 기록 개시 위치는, 그 이전의 GOP의 정보를 이용할 수 없기 때문에, 통상은 클로즈드 GOP 구조를 가진다. 그 때문에, 기록 개시 위치에서는, 플레이 리스트 마크를 GOP경계에 할당한다. 기록 개시 위치 이외에서는, 플레이 리스트 마크를 엔트리 "PTSEStart"로 표시되는 위치에 할당한다. 또, 기록 개시 위치 이외에도, 플레이 리스트 마크를 할당하려고 하는 위치에 대응하는 GOP가 클로즈드 GOP 및 오픈 GOP 중 어느 쪽일지를 판별할 수 있으면, 판별 결과에 따라, GOP 경계 또는 엔트리 "PTSEStart" 중 적합한 쪽에 플레이 리스트 마크를 할당한다.
- [0180] 이와 같이 제어하여, 챕터 점프시에, 점프 목적지의 디코드 범위를 최소한으로 하는 것이 가능해지며, 챕터 경계와 본래의 기록 개시 위치가 일치하도록 할 수 있다.
- [0181] 이 발명의 실시의 한 형태에서는, 챕터 분할 위치를 지정하기 위한 플레이 리스트 마크는, 모두 타입이 엔트리 마크인 것으로 가정한다.
- [0182] 이하, 이 발명의 실시의 한 형태에 의한 챕터 분할에 대해서, 보다 상세하게 설명한다. 이하에서는, 플레이 리스트 마크의 종류가 엔트리 마크에 고정되어 있다고 가정한다.
- [0183] 먼저, 챕터는, 이용자에게 보이는 최소의 재생 구간 단위이며, 인접하는 플레이 리스트 마크의 사이의 구간을 가리킨다. 여기서, 플레이 리스트 마크는, 플레이 리스트에 의해 표시되는 최후미의 위치에는 할당되지 않는다. 또, 각 플레이 리스트의 선두의 위치에는, 반드시 플레이 리스트 마크가 할당된다.
- [0184] 게다가, 이 실시의 한 형태에서는, 분할 편집의 시간적인 입도(granularity)는, GOP 단위로 가정한다. 또, 기록 개시 위치가 포함되는 GOP 이외의 GOP는, 오픈 GOP인 것으로 가정한다. 이것은, 보다 고화질을 실현하기 위한 것이다.
- [0185] 먼저, 플레이 리스트에 플레이 리스트 마크를 삽입하는 처리에 대해 설명한다. 도 22는, 상술한 본 발명에 적용 가능한 포맷에 있어서의, 플레이 리스트 이하의 일례의 파일 구조를 나타낸다. 도 22의 보기에서는, 플레이 리스트 #10은, 값 "PlayItem_id"에 의해 참조되는 3개의 플레이 아이템 #0, #1 및 #2를 포함하다(도 12 참조). 플레이 아이템 #0 및 #1이 클립 #20의 구간을 각각 참조하며, 플레이 아이템 #2가 클립 #3의 한 구간을 참조하고 있다. 또, 플레이 리스트 #11은, 2개의 플레이 아이템 #0 및 #1을 포함하며, 플레이 아이템 #0 및 #1이 클립 #3의 한 개의 구간을 각각 참조하고 있다.
- [0186] 상술한 것처럼, 플레이 아이템은, 각 클립을 구성하는 클립 정보에 포함되는 블록 "blkEPMap()"에 기술되는 엔트리 포인트를 참조하며, 대응하는 클립 AV스트림 파일내의 어드레스를 지정한다. 이에 의해, 플레이 리스트로부터 클립 AV스트림 파일의 부분을 지정하여 소정의 순서로 재생시킬 수 있다.
- [0187] 게다가, 도 22의 예에서는, 플레이 리스트 #10에, 4개의 플레이 리스트 마크 PLM#0, PLM#1, PLM#2 및 PLM#3이 할당된다. 각 플레이 리스트에, 클립내의 점프 가능한 액세스 위치를 나타내는 마크를 설정할 수 있다. 이 마크를, 플레이 리스트 마크(PlayListMark: PLM으로 줄여서 부른다.)라고 부른다. 이 플레이 리스트 마크는, 플레이 리스트 파일중의 블록 "blkPlayListMark()"내의 값 "PL_mark_id"에 의해 참조되어 필드 "MarkTimeStamp"에 의해 시각 정보로 표시된다(도 15 참조).
- [0188] 다음에, 도 23a-도 23c를 참조하여, 플레이 리스트를 이용한 클립의 분할 편집에 대해서, 개략적으로 설명한다. 도 23a는, 플레이 리스트에 의해서 클립이 참조되는 상태를 나타낸다. 기록 매체상에 기록된 클립 #1 및 클립 #2는, 각각, 플레이 리스트 #1의 플레이 아이템 #1 및 플레이 아이템 #2로부터 재생 개시점(IN점) 및 재생 종료점(OUT점)을 지정함으로써 참조된다. 이 도 23a-도 23c의 예에서, 플레이 아이템 #1은, 클립 #1의 선두(위치 a)는 IN점이며, 후단(위치 b)은 OUT점이라는 정보에 근거하여 클립 #1을 참조하고 있다. 마찬가지로, 플레이 아이템 #2도, 클립 #2의 선두(위치 c) 및 후단(위치 d)이 각각 IN점 및 OUT점이 된다는 정보에 근거하여 클립 #2를 참조하고 있다. 플레이 리스트 #1에서는, 플레이 아이템 #1 및 플레이 아이템 #2의 재생순서를 지정한다.
- [0189] 따라서, 이 도 23a의 예에서는, 플레이 리스트 #1의 명령에 응답하여, 클립 #1이 위치 a로부터 위치 b까지 재생되며, 클립 #2는 클립 #1의 위치 b에 이어지는 위치 c로부터 재생되어 위치 d에서 재생이 종료한다. 즉,

이 도 23a의 예에서는, 플레이 리스트 #1로부터의 명령에 응답하여, 클립 #1의 선두로부터 클립 #2의 끝까지가 연속적으로 재생된다.

- [0190] 또, 도 23a의 예에서는, 클립 #1의 선두의 위치 a와 클립 #2의 선두의 위치 c에, 플레이 리스트 마크 PLM#1 및 PLM#2가 각각 설정되어 있다. 이 경우, 예를 들면 플레이 리스트 #1에 따라 클립 #1의 위치 a와 b 사이의 임의의 위치에서 AV스트림 파일이 재생하고 있을 때, 점프가 지정된다면, 재생 방향에서 현재의 재생 위치와 가장 인접하게 설정된 플레이 리스트 마크의 위치(위치 a로부터 위치 b를 향해서 재생하고 있는 경우, 플레이 리스트 마크 PLM#2의 위치)로, 재생 위치가 점프되어, 플레이 리스트 마크 PLM#2의 위치로부터 재생이 계속되게 된다.
- [0191] 클립의 도중에 플레이 리스트 마크를 설정하여, 해당 클립을 가상적으로 분할할 수 있다. 예를 들면, 도 23b에 일례가 도시된 바와 같이, 플레이 리스트 #1에 대해, 클립 #2의 소망하는 위치에 플레이 리스트 마크 PLM#3을 할당한다. 이에 의해, 도 23c에 도시된 바와 같이, 클립 #2의 플레이 리스트 마크 PLM#3에 대응하는 위치 e를 점프 위치로서 지정하는 것이 가능해져, 클립 #2가 위치 e로 가상적으로 분할된 것이라고 볼 수 있게 된다.
- [0192] 이용자측에서 보았을 경우, 플레이 리스트에 의해 표시되는 구간은, 챕터에 따라 설정된다. 챕터는, 플레이 리스트에 각 플레이 리스트 마크를 할당함으로써 설정될 수 있다. 예를 들면, 플레이 리스트로 표시되는 구간의 선두에만 플레이 리스트 마크가 제공되어 있거나, 플레이 리스트에 플레이 리스트 마크가 전혀 설치되지 않으면, 해당 플레이 리스트 전체가 한 개의 챕터를 구성한다.
- [0193] 또, 상술한 도 23a의 예에 도시된 바와 같이, 클립 #1의 선두의 위치 a에 대응하는 위치에 플레이 리스트 마크 PLM#1이 설치되며, 클립 #2의 선두의 위치 c에 대응하는 위치에 플레이 리스트 마크 PLM#2가 설치되어 있다면, 플레이 리스트 마크 PLM#1로부터 PLM#2까지의 구간이 챕터 #1로 간주되며, 플레이 리스트 마크 PLM#2로부터 플레이 리스트 #1의 마지막 위치까지의 구간이 챕터 #2로 간주된다. 이 경우, 플레이 리스트 #1에 나타나는 전체 길이가 플레이 리스트 마크 #2에 의해 두 개의 챕터로 분할된다.
- [0194] 게다가, 상술한 도 23c의 예의 같이, 클립 #2의 중간의 소망하는 위치 e에 대응하는 플레이 리스트 마크 PLM#3이 설치되었을 경우, 도 23a의 챕터 #2가 플레이 리스트 마크 PLM#3의 위치에서 더욱 분할된다. 즉, 플레이 리스트 마크 PLM#2로부터 플레이 리스트 마크 PLM#3까지의 구간이 챕터 #2로 간주되며, 플레이 리스트 마크 PLM#3으로부터 플레이 리스트 #1의 마지막 위치까지의 구간이 챕터 #3으로 간주된다.
- [0195] 도 24는, 플레이 리스트 마크를 추가 삽입하는 일례의 처리를 나타내는 플로차트이다. 도 24의 처리에서는, 플레이 리스트내에 플레이 리스트 마크를 저장하는 블록 "blkPlayListMark()"(도 15 참조)을 검색하여, 현재 재생 위치와 이미 설정되어 있는 플레이 리스트 마크와의 위치 관계를 구하고, 그 결과에 근거해 플레이 리스트 마크를 추가로 삽입함과 동시에, 블록 "blkPlayListMark()"의 정보를 고쳐 쓴다.
- [0196] 스텝(S50)에서, 카운트치 i를 값 0으로서 초기화하고, 다음의 스텝(S51)에서, 현재 재생중의 플레이 아이템(도 24에서는 PI로 표기)의 번호가 i번째의 플레이 리스트 마크(도 24에서는 PLM으로 표기)에 참조되는 플레이 아이템의 번호보다 큰가 아닌가가 판단된다.
- [0197] 예를 들면, 도 12의 블록 "blkPlayList()"에서, 루프 변수 "PlayItem_id"를 이용하는 for-loop문내에 기술되는 블록 "blkPlayItem()" 가운데, 현재 재생중의 플레이 아이템에 대응하는 값 "PlayItem_id"이 연산되며, 연산된 값 "PlayItem_id"와 블록 "blkPlayListMark()"에 있어서의 i번째의 값 "RefToPlayItemID"이 비교된다. 현재 재생중의 플레이 아이템의 번호가 크다고 판단되면, 카운트치 i가 1만큼 증가되어, 처리가 스텝(S51)으로 돌아간다.
- [0198] 스텝(S51)에서, 현재 재생중의 플레이 아이템의 번호가 i번째의 플레이 리스트 마크에 의해 참조되는 플레이 아이템의 번호보다 크지 않다면, 즉, 현재 재생중의 플레이 아이템과 i번째의 플레이 리스트 마크에 의해 참조되는 플레이 아이템의 번호가 동일한지, 혹은, 현재 재생중의 플레이 아이템의 번호보다 i번째의 플레이 리스트 마크에 의해 참조되는 플레이 아이템의 번호가 크다고 판단되면, 처리는 스텝(S52)으로 진행된다.
- [0199] 스텝(S52)에서는, 현재 재생중의 플레이 아이템의 번호와 i번째의 플레이 리스트 마크에 의해 참조되는 플레이 아이템의 번호가 동일한가 아닌가가 판단된다. 동일하다고 판단되면, 처리는 다음의 스텝(S53)으로 진행된다. 동일하지 않다고 판단되면, 처리는 후술하는 스텝(S54)으로 진행된다.
- [0200] 스텝(S53)에서는, 현재의 재생 시각이 i번째의 플레이 리스트 마크가 지시하는 재생 시각보다 다음에 있는지 아

난지가 판단된다. 현재의 재생 시각은, 예를 들면 비디오 스트림을 디코드하는 디코더로부터 취득할 수 있다.

또, i 번째의 플레이 리스트 마크가 지시하는 재생 시각은, 도 15를 참조하여 얻어질 수 있으며, 블록 "blkPlayListMark()"에 있어서의 i 번째의 값 "MarkTimeStamp"에 근거한다.

[0201] 현재의 재생 시각이 i 번째의 플레이 리스트 마크가 지시하는 시각보다 다음에 있다고 판단되면, 카운트치 i 가 1만큼 증가되고, 처리가 스텝(S53)으로 돌아간다. 현재의 재생 시각이 i 번째의 플레이 리스트 마크가 지시하는 재생 시각보다 늦지 않다고 판단되면, 즉, 현재의 재생 시각이 i 번째의 플레이 리스트 마크가 지시하는 재생 시각과 동일한지, 혹은, 현재의 재생 시각이 i 번째의 플레이 리스트 마크가 지시하는 재생 시각보다 이전이면, 처리는 스텝(S54)으로 진행된다.

[0202] 스텝(S54)에서는, i 번째의 위치에 플레이 리스트 마크(이 경우는 엔트리 마크)를 추가 삽입하는 처리를 한다. 여기에서는, 블록 "blkPlayListMark()"에서, 기존의 i 번째의 플레이 리스트 마크와 $(i-1)$ 번째의 플레이 리스트 마크와의 사이에, 플레이 리스트 마크가 추가 삽입되게 된다. 기존의 i 번째의 플레이 리스트 마크는, 플레이 리스트 마크의 추가 삽입 처리 후에는, $(i+1)$ 번째의 플레이 리스트 마크로 간주된다.

[0203] 보다 상세하게는, 블록 "blkPlayListMark()"에서는, 현재 재생중의 플레이 아이템에 대응하는 값 "PlayItem_id"를, 필드 "RefToPlayItemID"의 값으로 설정한다. 또, 필드 "MarkTimeStamp"의 값은, 현재의 재생 시각을 나타내는 값으로 간주된다. 게다가 필드 "NumberOfPlayListMarks"의 값이 1만큼 증가되어 필드 "Length"의 값이 재계산된다.

[0204] 실제로, 블록 "blkPlayListMark()"은 플레이 리스트 파일의 일부이므로, 도 11에 일례가 도시된 바와 같이, 플레이 리스트 파일의 섹스에서는, 필드 "ExtensionDataStartAddress"의 값이, 블록 "blkPlayListMark()"에 있어서의 필드 "Length"의 값에 따라 고쳐 쓸 수 있게 된다.

[0205] 도 25a와, 도 25b를 이용해 보다 구체적으로 플레이 리스트 마크 추가 삽입의 처리를 설명한다. 도 25a 및 도 25b는, 상술한 도 22와 같은 구조이므로, 상세한 설명은 생략한다. 여기에서는, 도 25a를 참조하여, 플레이 리스트 마크 PLM#2 및 PLM#3이 재생되는 동안에, 위치 a에 플레이 리스트 마크를 추가 삽입하는 경우에 대해 설명한다.

[0206] 카운트치 i 가 초기화되어 값이 "0"으로 되며, 위치 a에 대응하는 플레이 아이템은, 플레이 아이템 #2이므로, 현재 재생중의 플레이 아이템의 번호가 2로 간주된다. 한편, i 번째, 즉 0번째의 플레이 리스트 마크 #0은, 플레이 아이템 #0을 참조하고 있으므로, i 번째의 플레이 리스트 마크가 참조하는 플레이 아이템 번호가 0으로 간주된다. 스텝(S51)의 판단에 근거하여, 현재 재생중의 플레이 아이템의 번호가 i 번째의 플레이 리스트 마크가 참조하는 플레이 아이템 번호보다 크기 때문에, 카운트치 i 가 1만큼 증가되어 값이 "1"로 되며 다시 스텝(S51)에서 판단이 이루어진다.

[0207] 스텝(S51)의 두 번째의 판단에서는, i 번째, 즉 1번째의 플레이 리스트 마크 PLM#1이 참조하는 플레이 아이템 번호와 현재 재생중의 플레이 아이템 번호가 비교된다. 첫 번째의 플레이 리스트 마크 PLM#1은, 플레이 아이템 #1을 참조하고 있어, 참조하는 플레이 아이템 번호가 1로 된다. 현재 재생중의 플레이 아이템의 번호가 i 번째의 플레이 리스트 마크가 참조하는 플레이 아이템 번호보다 크기 때문에, 스텝(S51)의 판단에 근거하여, 카운트치 i 가 1만큼 증가되어 값이 "2"로 되며, 다시 스텝(S51)에서 판단이 이루어진다.

[0208] 스텝(S51)의 세 번째 판단에서는, i 번째, 즉 2번째의 플레이 리스트 마크 PLM#2는, 플레이 아이템 #1로 지정되는 구간내에 있으며, 2번째의 플레이 리스트 마크 PLM#2가 참조하는 플레이 아이템 번호가 1로 간주된다. 따라서, 현재 재생중의 플레이 아이템 번호가, i 번째의 플레이 리스트 마크가 참조하는 플레이 아이템 번호보다 크다고 간주되어, 카운트치 i 가 1만큼 증가되어 값이 "3"으로 되고, 다시 스텝(S51)에서 판단이 이루어진다.

[0209] 스텝(S51)의 네 번째 판단에서는, i 번째, 즉 3번째의 플레이 리스트 마크 PLM#3은, 플레이 아이템 #2로 지정되는 구간내에 있으며, 3번째의 플레이 리스트 마크가 참조하는 플레이 아이템 번호가 "2"로 간주된다. 따라서, 현재 재생중의 플레이 아이템 번호가 i 번째의 플레이 리스트 마크가 참조하는 플레이 아이템 번호와 동일하다고 간주되며, 처리는 스텝(S52)으로 진행된다. 스텝(S52)에서는, 3번째의 플레이 리스트 마크 PLM#3이 참조하는 플레이 아이템 번호와 현재 재생중의 플레이 리스트 마크의 번호가 동일하기 때문에, 처리는 스텝(S53)으로 진행된다.

[0210] 카운트치 i 는 "3"이므로, i 번째의 플레이 리스트 마크인 플레이 리스트 마크 PLM#3의 설정된 시각이 현재 재생중의 위치 a의 시각보다 이후가 된다. 따라서, 스텝(S53)에서는, 현재의 재생 시각을 나타내는 값이 i 번째의 플레이 리스트 마크의 설정된 시각을 나타내는 값보다 작다고 간주되어, 처리는 스텝(S54)으로 진행되어, 위치

a에 대응하는 플레이 리스트 마크가 설정된다.

- [0211] 즉, 참조하는 플레이 아이템 번호가 "2"가 되며, 위치 a의 재생 시각을 나타내는 플레이 리스트 마크가 3번째의 플레이 리스트 마크 PLM#3으로서 추가 삽입된다. 또, 플레이 리스트 마크수가 1만큼 증가되어, 이 플레이 리스트 마크 PLM#3이 추가 삽입되기 직전의 해당 플레이 리스트 마크 PLM#3이 가리키는 재생 시각 이후에 설치된 플레이 리스트 마크의 번호가 각각 1만큼 증가된다. 즉, 도 25a와 도 25b의 예에서, 도 25a에 도시된 플레이 리스트 마크 PLM#3은, 플레이 리스트 마크의 추가 삽입 처리에 의해 도 25b에 도시된 바와 같이, 플레이 리스트 마크 PLM#4로 간주되며, 추가 삽입된 플레이 리스트 마크가, 플레이 리스트 마크 PLM#3으로 간주된다.
- [0212] 스텝(S52)에서는, 현재 재생중의 플레이 아이템 번호와 i번째의 플레이 리스트 마크의 참조하는 플레이 아이템 번호가 동일하지 않은 지, 환언하면, 현재 재생중의 플레이 아이템 번호가 i번째의 플레이 리스트 마크의 참조하는 플레이 아이템 번호보다 작은지 큰지가 판단되면, 각각 대응하는 구간에 플레이 리스트 마크가 설치되어 있는 2개의 플레이 아이템의 사이에 있고, 플레이 리스트 마크가 설치되지 않은 플레이 아이템에, 새롭게 플레이 리스트 마크를 추가 삽입할 수 있다.
- [0213] 일례로서 도 26에 도시된 바와 같이, 플레이 아이템 #0, 플레이 아이템 #1 및 플레이 아이템 #2의 3개의 플레이 아이템을 포함한 플레이 리스트 #12를 생각한다. 여기서, 플레이 아이템 #0의 선두에 대응하는 위치와 플레이 아이템 #2로 나타나는 구간내에, 플레이 리스트 마크 PLM#0 및 플레이 리스트 마크 PLM#1이 각각 설정되어 있는 것으로 가정한다.
- [0214] 이러한 상태에서, 플레이 아이템 #1로 나타나는 구간내의 위치 b에 플레이 리스트 마크를 추가 삽입한다. 카운트치 i가 "1"일때에, 현재 재생중의 플레이 아이템 번호가 "1", i번째의 플레이 리스트 마크가 참조하고 있는 플레이 아이템 번호가 "2"가 되므로, 스텝(S51)에서는, 현재 재생중의 플레이 아이템 번호보다 i번째의 플레이 리스트 마크가 참조하고 있는 플레이 아이템 번호가 크다고 판단하고, 처리가 스텝(S52)으로 이행된다. 스텝(S52)에서는, 현재 재생중의 플레이 아이템 번호가 "1", i번째의 플레이 리스트 마크가 참조하고 있는 플레이 아이템 번호가 "2"이므로, 양자가 동일하지 않다고 간주된다. 이 경우, 위치 b에 대응하는 플레이 아이템 #1에는, 다른 플레이 리스트 마크가 존재하지 않기 때문에, 스텝(S53)에서는, 현재 재생 시각을 나타내는 값이 i번째의 플레이 리스트 마크가 설정된 시각을 나타내는 값보다 적은지 아닌지를 판단할 필요가 없다. 따라서, 처리는 스텝(S52)으로부터 스텝(S54)으로 이행되어, 위치 b에 대응하는 플레이 리스트 마크의 추가 삽입 처리가 이루어진다.
- [0215] 이 발명의 개념과 관련되는, 플레이 리스트 마크를 마련하는 위치의 결정 방법에 대해 설명한다. 이미 설명한 것처럼, 엔트리 "PTSEPSstart"에 의해 나타나는 위치에 플레이 리스트 마크를 할당하는 제 1의 방법과 GOP 경계의 위치에 플레이 리스트 마크를 할당하는 제 2의 방법의 두 가지 방법이 생각된다. 이하에, 각각의 경우에 있어서의 장점, 결점 및, 개선방법에 대해 설명한다.
- [0216] 예를 들면 MPEG2 방식에 따라, I픽처, P픽처 및 B픽처에 의해 GOP가 구성되는 경우, 엔트리 "PTSEPSstart"로 나타나는 위치는, I픽처의 위치에 대응시킬 수 있다. 또, 예를 들면 MPEG4나 H.264 | AVC 방식에 따라, IDR 픽처를 추가로 이용하는 경우에는, IDR 픽처의 위치에 대응시킬 수 있다.
- [0217] 도 27a와 도 27b는, 제 1 및 제 2의 방법에 있어서의 각각의 캡처 경계의 예를 나타낸다. 플레이 리스트 마크 PLM#1 및 플레이 리스트 마크 PLM#2에 의해 캡처의 경계가 표시되며, 플레이 리스트 마크 PLM#1 및 PLM#2의 사이가 캡처#2로 간주된다. 도 27은, 제 1의 방법에 있어서의 일례의 캡처 경계를 나타낸다. 기록이 개시되는 위치는, GOP의 선두가 되므로, 엔트리 "PTSEPSstart"로 나타나는 위치에 플레이 리스트 마크를 배치하면, GOP의 선두와 플레이 리스트 마크에 근거하는 각 캡처의 선두와의 사이에, 도 27a에 도시된 바와 같이, 불일치 구간이 발생한다.
- [0218] 따라서, 캡처 선두부의 불일치 구간의 한 개 내지 복수 프레임이 재생되지 않게 된다. 예를 들면, 기록 개시 위치에 플레이 리스트 마크를 배치하려고 했을 경우, 기록 개시 위치로부터 개시되는 GOP내에 있는, 최초의 I픽처(MPEG2 방식의 경우)보다 시간적으로 이전에 있는, 예를 들면 한 개 내지 복수의 B픽처가 재생되지 않는 문제점이 생긴다.
- [0219] 또, 캡처의 후단부에 있어서는, 불일치 구간의 한 개 내지 복수 프레임은, 불일치 구간의 직전의 GOP와는 다른 GOP에 속한다. 그 때문에, 해당 불일치 구간의 한 개 내지 복수 프레임은, 해당 불일치 구간의 직전의 프레임과 내용적으로 관련성이 없는 경우가 있다. 예를 들면, 불일치 구간의 직전의 GOP는, 기록 정지에 따라 기록된 GOP이며, 불일치 구간을 포함한 GOP는, 다음의 혹은 다른 기록 개시에 기록된 GOP인 경우 등이다. 이러한

경우, 챕터의 후단부에 있고, 관련성이 없는 프레임이 몇 개의 프레임이 재생되어 버리는 문제가 생긴다.

- [0220] 도 27b는, 제 2의 방법에 있어서의 일례의 챕터 경계를 나타낸다. 이 제 2의 방법에서는, 챕터 경계가 GOP 경계와 일치하고 있으므로, 상술한 제 1의 방법의 경우와 같은 문제는 발생하지 않는다.
- [0221] 도 28a와 도 28b는, 제 1 및 제 2의 방법에 있어서의 디코드 구간의 예를 나타낸다. 플레이 리스트 마크 PLM# 1 및 PLM#2의 사이의 구간은 표시해야 할 구간이다. 도 28a는, 제 1의 방법의 경우의 일례의 디코드 구간을 나타낸다. 챕터 선두에서는, 엔트리 "PTSEPEnd"로 나타나는 위치에 플레이 리스트 마크가 설치되어 있으므로, 디코드 개시 위치는, 이 엔트리 "PTSEPEnd"로 나타나는 위치에 대응하는 픽처가 포함되는 GOP의 선두로부터 시작된다. 즉, 각 챕터의 선두측에서는, GOP 선두로부터 플레이 리스트 마크 PLM#1까지의 구간이, 표시해야 할 구간에 대해서 여분으로 디코드할 필요가 있는 구간이 된다. 재생시에는, 이 여분으로 디코드된 구간은, 마스크 된다.
- [0222] 한편, 제 2의 방법의 경우는, 도 28b에 도시된 바와 같이, 플레이 리스트 마크가 설치된 위치의 직전의 GOP로부터 디코드를 실시할 필요가 있다. 이것은, GOP가 오픈 GOP 및 클로즈드 GOP의 어느 쪽인지를 판별할 수 없는 경우, 해당 GOP의 바로 이전의 GOP의 정보를 이용하여 그 GOP를, 오픈 GOP로서 디코드할 필요가 있기 때문이다. 즉, GOP 경계에서, 표시해야 할 구간에 대해서 한 개의 GOP분 만큼, 여분으로 디코드하는 구간이 발생한다.
- [0223] 또, 시각 정보(time information)로서는, 엔트리 "PTSEPEnd"로 나타나는 위치에 대응한 정보만이 이용된다. 그러므로, 플레이 리스트 마크는, 실제로는, 해당 플레이 리스트 마크가 설치된 위치의 전후의 엔트리 "PTSEPEnd"로 나타나는 위치동안에 존재한다는 사실만을 인식하게 된다. 이러한 의미에서 볼 때, 플레이 리스트 마크가 설치된 위치 전의 엔트리 "PTSEPEnd"로 나타나는 위치의 픽처가 포함되는 GOP로부터 디코드를 실시할 필요가 있다.
- [0224] 디코드 종료 위치는, 제 1의 방법에 대해서는, 챕터 종단의 프레임이 포함되는 GOP의 종단까지가 된다. 또, 제 2의 방법에 대해서는, 상술한 이유와 같이, 챕터 종단을 나타내는 플레이 리스트 마크의 위치가 그 전후의 어느 쪽의 GOP에 포함되는지를 인식할 수 없기 때문에, 해당 플레이 리스트 마크의 직후에 나타나는 엔트리 "PTSEPEnd"로 나타나는 위치에 대응하는 픽처를 포함한 GOP의 종단까지 디코드를 실시할 필요가 있다. 이와 같이, 결과적으로는, 제 1의 방법과 제 2의 방법으로 같은 위치까지 디코드를 실시할 필요가 있다.
- [0225] 도 29a-도 29c는, GOP가 오픈 GOP 및 클로즈드 GOP의 어느 쪽인지를 판별 가능한 경우에, 제 1 및 제 2의 방법에 따르는 디코드 구간의 예를 나타낸다. 플레이 리스트 마크 PLM# 1 및 PLM#2의 사이의 구간은 표시해야 할 구간이다. 도 29a는, 플레이 리스트 마크를 엔트리 "PTSEPEnd"로 나타나는 위치에 설치했을 경우의 예이며, 상술한 도 28a와 동일한 도면이다. 또, 도 29b는, 플레이 리스트 마크를 오픈 GOP의 GOP 경계에 설치했을 경우의 예이다.
- [0226] 이미 설명한 것처럼, 클로즈드 GOP는, 그 GOP내에서 디코드가 완결되어, 바로 직전의 GOP의 정보를 이용할 필요가 없다. 그 때문에, 도 29c에 도시된 바와 같이, GOP 경계에 플레이 리스트 마크를 설치했을 경우에도, 플레이 리스트 마크를 설치한 위치로부터 디코드를 개시할 수 있다. 즉, 플레이 리스트 마크가 설치된 GOP가 클로즈드 GOP인 것이라고 판별 가능한 경우에는, 표시해야 할 구간에 대해서 여분으로 디코드하는 구간이 발생하지 않는다.
- [0227] 또, 기록이 정지된 후에 다시 기록이 개시되었을 경우, 기록 정지시의 오픈 GOP의 다음의 GOP는, 기록 개시에 따라, 클로즈드 GOP가 되므로, 기록 개시 위치인 GOP경계에 플레이 리스트 마크가 설치된다. 도 29b에 도시된 바와 같이, 기록의 정지 부분에 있어서 표시해야 할 구간의 종단과 디코드 구간의 종단이 일치하므로, 이 플레이 리스트 마크에 근거하는 챕터의 종단에는 내용적으로 관련성이 없는 프레임이 표시되지 않는다.
- [0228] GOP가 오픈 GOP 및 클로즈드 GOP의 어느 쪽인지를 판별하는 수단으로서, 몇 개의 방법이 생각된다. 이 발명의 실시의 한 형태에서는, 기록 개시 시에는, 각 GOP는, 반드시 클로즈드 GOP 구성으로 간주되므로, 기록 개시 위치에 대응하는 위치에서 챕터 분할을 실시하는 경우에는, GOP 경계에 플레이 리스트 마크를 마련하도록 한다.
- [0229] 이것에 한정하지 않고, 예를 들면 MPEG2 방식으로는, 비디오 데이터의 엘리먼트 스트림의 헤더내에, GOP가 클로즈드 GOP인가 아닌가를 나타내는 플래그가 저장되므로, 이것을 이용하는 것을 생각한다.
- [0230] 또, 기록을 실시한 장치와 동일한 장치에 의해 챕터 분할 편집을 실시하고, 동일한 장치로 재생을 실시하는 경우에는, 클로즈드 GOP로 기록을 실시한 위치를 나타내는 정보를 장치내의 메모리 등에 기억해 두는 방법이 생각된다. 게다가 장치의 사양으로서, 클로즈드 GOP를 이용하고 기록을 실시하도록 한다면, 동일 기종의 장치로

챕터 분할을 실시하는 경우에는, 플레이 리스트 마크를 항상 GOP 경계에 마련하도록 할 수 있다.

- [0231] 기중명을 나타내는 정보는, 예를 들면 인덱스 파일, 무비 오브젝트 파일, 플레이 리스트 파일 및 클립 정보 파일에 각각 마련할 수 있는 확장 데이터 영역(예를 들면, 블록 blkExtensionData())에 저장할 수 있다. 오픈 GOP 및 클로즈드 GOP의 어느 쪽을 이용하고 있는지를 나타내는 정보를, 이러한 확장 데이터 영역에 대해서 저장하는 방법도 생각할 수 있다.
- [0232] 도 30은, 도 27a-도 29c를 이용하여 설명한 각 조건에 관한 평가를 나타낸다. 도 30에서, "○"(원)이 첨부된 항목은, 유용한 모드이며, 「×」가 첨부된 항목은, 바람직하지 않은 모드이다. 또, 「-」(가로줄)이 첨부된 항목은, 실질적으로 의미를 만들어내지 않는 모드이다.
- [0233] 도 30에서는, 챕터 분할을 실시하기 위해 지정된 위치가 기록 개시 또는 정지 위치에 대응하는 위치인지 아닌지에 대해서, 각각, 기록과 재생과에 있어서의 챕터 경계의 불일치가 존재하는지의 여부와, 챕터 선두부 및 후단부의 각각으로 읽기와 디코드가 표시 프레임 보다 많이 필요한 구간에서, 플레이 리스트 마크를 엔트리 "PTSEPEnd"의 위치와 GOP 경계 위치와에 마련했을 경우에 대해 평가하고 있다. 또, GOP 경계 위치에 플레이 리스트 마크를 마련하는 경우에, 기록, 편집 및 재생을 동일한 장치로 실시해 장치 독자적인 정보를 이용할 수 있는 경우(도 30의 "고유 정보를 이용하는 장치"로 칭한다.)와 기록 및 편집을 실시한 장치와 재생을 실시하는 장치가 다른 경우("일반 재생기"로 칭한다.)의 각각에 대해 평가하고 있다.
- [0234] 도 30에서, M은, 기준 픽처로부터 다음의 기준 픽처까지의 픽처의 이동수, N치는, 한 개의 GOP내의 픽처수를 나타낸다. 예를 들면, GOP가 「 $I_2B_0B_1P_5B_3B_4P_8B_6B_7P_{11}B_9B_{10}P_{14}B_{12}B_{13}$ 」의 15매의 픽처로 구성되는 경우, M=3, N=15이다.
- [0235] 기록 개시 또는 정지의 경계 위치에 대응하는 위치에서 챕터 분할을 실시하도록 지시했을 경우에 대해 설명한다. 이 때, 기록과 재생에 있어서의 챕터 위치의 차이에 관해서는, 도 27a와 도 27b를 이용하여 설명한 것처럼, 플레이 리스트 마크를 엔트리 "PTSEPEnd"의 위치에 마련하는 경우에 차이가 생기고, GOP 경계 위치에 마련하는 경우에는, 차이가 생기지 않는다.
- [0236] 디코드 구간에서, 디코드 개시 위치에 관해서는, 도 29a-도 29c를 이용하여 설명한 것처럼, 플레이 리스트 마크를 엔트리 "PTSEPEnd"의 위치에 마련하는 경우에는, (M-1) 프레임 분이 표시 프레임보다 많이 필요하다. GOP 경계 위치에 마련하는 경우에는, 기록 개시 위치는 반드시 클로즈드 GOP가 되므로, 일반 재생 장치 및 고유 정보를 이용하는 장치의 어느 쪽에 대해서도, 표시 프레임 구간과 디코드 구간이 일치한다. 또, 디코드 종료 위치에 관해서는, 플레이 리스트 마크를 엔트리 "PTSEPEnd"의 위치에 마련하는 경우에는, (N-M+1) 프레임 분이 표시 프레임 보다 필요하며, GOP 경계 위치에 마련하는 경우에는, 일반 재생 장치 및 고유 정보를 이용하는 장치의 어느 쪽에 대해서도, 표시 프레임 구간과 디코드 구간이 일치한다.
- [0237] 기록 개시 또는 정지의 경계 위치가 아닌 위치에서 챕터 분할을 실시하도록 지시했을 경우에 대해 설명한다. 이 경우, 기록과 재생에 있어서의 챕터 위치의 불일치에 관한 평가는 의미가 없으므로, 도 30의 「-」(가로줄)로 표시된다.
- [0238] 디코드 구간에서는, 도 29a-도 29c를 이용하여 설명한 것처럼, 디코드 개시 위치에 관해서는, 플레이 리스트 마크를 엔트리 "PTSEPEnd"의 위치에 마련하는 경우에는, (M-1) 프레임 분이 표시 프레임보다 많이 필요하다. 한편, GOP 경계 위치에 플레이 리스트 마크를 마련하는 경우에는, 일반 재생 장치에 대해 직전의 한 개의 GOP분, 즉 N프레임 분이 표시 프레임보다 많이 필요하며, 고유 정보를 이용하는 장치에 대해서는, 플레이 리스트 마크가 설치되는 GOP가 오픈 GOP라고 판별되면, 한 개의 GOP분, 즉 N프레임 분이 표시 프레임 보다 많이 필요하게 된다.
- [0239] 또, 고유 정보를 이용하는 장치에서는, 해당 GOP가 클로즈드 GOP라고 판별되면, 표시 개시 프레임과 디코드 개시 프레임이 일치한다.
- [0240] 게다가, 디코드 종료 위치에 관해서는, 플레이 리스트 마크를 엔트리 "PTSEPEnd"의 위치에 마련하는 경우에는, (N-M+1) 프레임 분이 표시 프레임 보다 많이 필요하다. 한편, GOP 경계 위치에 플레이 리스트 마크를 마련하는 경우에는, 일반 재생 장치에 대해 직전의 한 개의 GOP분, 즉 N프레임 분이 표시 프레임보다 많이 필요하며, 고유 정보를 이용하는 장치에 대해서는, 표시 종료 프레임과 디코드 종료 위치가 일치한다.
- [0241] 이와 같이, 기록 개시 또는 정지의 경계 위치에서 챕터 분할을 실시하는 경우에는, GOP 경계에 플레이 리스트

마크를 설치한다. 기록 개시 또는 정지의 경계 위치 이외에서, 챕터 분할을 실행하는 경우에는, 엔트리 "PTSEPStart"에 나타나는 위치에 플레이 리스트 마크를 마련하여, 이용자의 편리성을 최대로 할 수 있다.

[0242] 도 31은, 챕터 분할 편집을 실시할 때에, 플레이 리스트 마크의 일례의 삽입 방법을 나타내는 플로차트이다. 예를 들면 플레이 리스트에 따라 클립을 재생중에 챕터 분할이 지시되면, 먼저, 최초의 스텝(S60)에서, 현재 재생중의 위치에 대응하는 GOP가 클로즈드 GOP라고 판정 가능한가 아닌가가 판단된다. 현재 재생중의 위치는, 예를 들면 비디오 스트림을 디코딩하는 디코더로부터 취득할 수 있다.

[0243] 스텝(S60)에서, 그 GOP가 클로즈드 GOP라고 판정될 수 없다면, 처리는 스텝(S61)으로 진행하고, 플레이 리스트 마크를 설치한 현재 시각으로서 엔트리 "PTSEPStart"의 값을 이용한다. 스텝(S60)에서, GOP가 클로즈드 GOP라고 판정 가능하다면, 처리는 스텝(S63)으로 이행되어, 플레이 리스트 마크를 설치한 현재 시각으로서 GOP 경계의 값을 이용한다.

[0244] 스텝(S61) 또는 스텝(S63)의 처리 뒤에, 처리는 스텝(S62)으로 진행되어, 스텝(S61) 또는 스텝(S63)에서 결정된 현재 시각에 근거하여 플레이 리스트 마크를 설치하는 처리가 이루어진다. 즉, 이 스텝(S62)에서, 상술한 도 24의 스텝(S54)에서 설명한, 블록 "blkPlayListMark()"내의 필드 "MarkTimeStamp"의 값이, 스텝(S61) 또는 스텝(S63)의 결과에 따라 결정된다.

[0245] GOP 경계의 위치는, 예를 들면 비디오 스트림을 디코딩하는 디코더로부터 취득할 수 있다. 이것에 한정하지 않고, 예를 들면 상술한 엔트리 "PTSEPStart"의 값에 근거해 구할 수 있다. 이 경우에는, 엔트리 "PTSEPStart"의 위치로부터 GOP의 구조에 따라 소정 프레임 분만큼 거슬러 올라간 위치를, GOP의 선두 위치로 한다.

[0246] 이용자의 기록 개시 조작에 따라, 각 기록 개시 위치에 자동적으로 플레이 리스트 마크를 마련하는 기록 장치도 생각할 수 있다. 기록 장치를 이러한 구성으로 설계함으로써, 재생시에, 클립 분할 편집 처리를 실시하지 않아도 기록 개시 및 정지시에 챕터 점핑을 실시하는 것이 가능하게 된다.

[0247] 도 32는, 이러한, 기록 개시에 따라 기록 개시 위치에 자동적으로 플레이 리스트 마크를 마련하는 경우의 일례의 처리를 나타내는 플로차트이다. 예를 들면 이용자에 의해 기록 장치에 대해서 기록 개시 조작이 이루어지면, 클립의 기록이 개시됨과 동시에, 플레이 리스트 마크를 마련하는 현재 시각으로서 GOP 경계의 값을 이용하도록 결정된다. 그리고, 다음의 스텝(S71)에서, 스텝 S70의 결과에 근거해, 플레이 리스트 마크를 마련하는 처리를 한다. 즉, 이 스텝(S71)에서, 기록이 개시되어 최초로 생성된 GOP의 선두에 대응하는 재생 시각이 블록 "blkPlayListMark()"내의 필드 "MarkTimeStamp"의 값으로 설정되어, 플레이 리스트 마크가 설치된다.

[0248] 기록 개시 위치, 즉 기록이 개시되어 최초로 생성된 GOP의 선두에 대응하는 시각은, 예를 들면 플레이 리스트에 대해서 최초로 기록되는 클립에 관해서는, 값 "0"에 소정의 오프셋을 더한 값으로 설정된다. 기록 개시 및 정지에 의해 생성된 클립을 추가 기입한 사양의 경우에, 그 시각은, 예를 들면 이미 기록된 클립의 누적적인 재생 시각에 소정의 오프셋을 더한 값에 의해 얻어지는 값이 된다. 오프셋은, 예를 들면, 디코더에 스트림의 공급이 개시되고 나서 해당 GOP의 선두의 프레임이 표시되는 시간에 대응하는 값이며, 장치의 사양 등에 대응하는 고유의 값이다.

[0249] 이것은 다음의 경우에 적합한 방법이다. 예를 들면 한 개의 클립이 여러 차례의 기록 개시 및 정지 조작을 포함할 수 있는 경우에, 클립 AV스트림 파일, 클립 정보 파일, 및 해당 클립 정보 파일에 대응하는 플레이 아이템을 한 개씩 이용하면서, 플레이 리스트 마크만으로, 기록 개시 및 정지 조작에 대응하는 챕터 경계를 제공한다

[0250] 한 개의 플레이 리스트 파일에 할당가능한 플레이 리스트 마크의 수에 대해서 상한을 설정할 수 있다. 플레이 리스트 마크수의 상한을 설정하여, 무제한으로 플레이 리스트 파일의 용량이 증대하는 것이 방지된다. 또, 예를 들면 편집기나 재생 장치에 있어서의 챕터 표시 수에, 예를 들면 3자리수 등의 제한이 설치되어 있는 경우에, 플레이 리스트 마크의 상한을 설정하는 일도 생각할 수 있다. 상술한 도 31 및 도 32를 이용해 설명한 처리는, 각각, 현재 설정하려고 하는 플레이 리스트 마크의 수가 이 플레이 리스트내에 설정 가능한 플레이 리스트 마크 수의 상한을 넘지 않는가 아닌가의 판단을 실시한 다음, 상한을 넘지 않는다는 판단이 이루어진 후에, 행해진다.

[0251] 다음에, 이 발명의 실시의 한 형태에 적용 가능한 일례의 기록 재생장치에 대해 설명한다. 먼저, 기록 재생장치의 설명에 앞서, 가상 플레이어에 대해서, 개략적으로 설명한다. 상술한 데이터 구조를 가지는 디스크가 플레이어에 장착되면, 플레이어는, 디스크로부터 판독한 무비 오브젝트 등에 기술된 명령을, 플레이어 내부의 하드웨어를 제어하기 위한 고유의 명령으로 변환할 필요가 있다. 플레이어는, 이러한 변환을 행하기 위한 소프트

웨어를, 플레이어에 내장되는 ROM(Read Only Memory)에 미리 기억하고 있다. 이 소프트웨어는, 디스크와 플레이어 중개하고, 플레이어에 이 포맷의 규정에 따른 동작을 시키며, 가상 플레이어라고 칭해진다.

[0252] 도 33a와 도 33b는, 이 가상 플레이어의 동작을 개략적으로 나타낸다. 도 33a는, 디스크의 로딩시의 동작의 예를 나타낸다. 디스크가 플레이어에 장착되어, 디스크에 대한 이니셜 액세스가 되면(스텝 : S30), 한 개의 디스크에 대해 공유적으로 이용되는 공유 파라미터가 기억되는 레지스터가 초기화된다(스텝 : S31). 그리고, 다음의 스텝(S32)에서, 무비 오브젝트 등에 기술된 프로그램이 디스크로부터 판독되어 실행된다. 이니셜 액세스는, 디스크 장착시와 같이, 디스크의 재생이 처음으로 행해지는 것을 말한다.

[0253] 도 33b는, 플레이어가 정지상태로부터 이용자에 의해 예를 들면 플레이 키가 가압되어, 재생이 지시받았을 경우의 동작의 예를 나타낸다. 최초의 정지상태(스텝 : S40)에서는, 이용자에 의해, 예를 들면 원격 조정 명령(UO : User Operation) 등을 이용하여 재생을 지시한다. 재생이 지시받으면, 먼저, 레지스터, 즉 공통 파라미터가 초기화되고(스텝 : S41), 다음의 스텝(S42)에서, 무비 오브젝트 실행 단계로 이행한다.

[0254] 무비 오브젝트의 실행 단계에 있어서의 플레이 리스트의 재생에 대해서, 도 34를 이용하여 설명한다.

[0255] UO등에 의해, 타이틀 번호 #1을 가지는 콘텐츠를 재생 개시하는 지시가 있었을 경우에 대해 생각한다. 플레이어는, 콘텐츠의 재생 개시 지시에 따라, 상술한 도 2에 나타나는 인덱스 테이블(Index Table)을 참조하고, 타이틀 #1의 콘텐츠 재생에 대응하는 오브젝트의 번호를 취득한다. 예를 들면 타이틀 #1의 콘텐츠 재생을 실현하는 오브젝트의 번호가 #1이었다고 하면, 플레이어는, 무비 오브젝트 #1의 실행을 개시한다.

[0256] 도 34의 예에서는, 무비 오브젝트 #1에 기술된 프로그램은 2행으로부터 시작된다. 1행째의 명령이 "Play PlayList(1)"이라고 하면, 플레이어는, 플레이 리스트 #한 개의 재생을 개시한다. 플레이 리스트 #1은, 1이상의 플레이 아이템으로부터 구성되어 플레이 아이템이 차례차례 재생된다. 플레이 리스트 #1중의 플레이 아이템의 재생이 종료하면, 무비 오브젝트 #한 개의 실행으로 돌아와, 2행째의 명령이 실행된다. 도 34의 예에서는, 2행째의 명령이 "jump MenuTitle"이면, 이 명령이 실행되어 인덱스 테이블에 기술된 메뉴 타이틀(MenuTitle)을 실현하는 무비 오브젝트의 실행이 개시된다.

[0257] 삭제

[0258] 삭제

[0259] 삭제

[0260] 삭제

[0261] 도 35는, 이 발명의 실시의 한 형태에 적용 가능한 기록 재생장치의 일례의 구성을 개략적으로 가리킨다. 도 34에 예시되는 기록 장치는, 외부로부터 입력되는 비디오 데이터 및 오디오 데이터를 기록 매체에 기록하고, 기록 매체에 기록된 비디오 데이터 및 오디오 데이터를 재생하는, 단독의 기록 재생장치로서 이용할 수도 있고, 광학계나 촬상 소자등을 갖춘 카메라 블록과 조합해 촬상한 촬상 신호에 근거하는 비디오 데이터를 기록 매체에 기록하는, 비디오 카메라 장치의 기록 블록으로서 이용할 수도 있다.

[0262] 기록/재생 장치에 적용 가능한 압축 인코드나 다중화의 방식으로서, 여러가지가 생각된다. 예를 들면, H.264 | AVC에 규정되는 방식을, 이 발명의 실시의 한 형태의 압축 인코드로서 적용할 수 있다. 이것에 한정하지 않고, MPEG2 방식에 근거해 압축 인코드를 실시하도록 해도 괜찮다. 또, 다중화 방식은, 예를 들면 MPEG2 시스템즈를 적용할 수 있다. 이하에서는, 비디오 데이터의 압축 인코드를 H.264 | AVC에 규정되는 방식으로 실시하고, 비디오 데이터 및 오디오 데이터의 다중화를, MPEG2 시스템즈에 규정되는 방식으로 실시하는 것으로 가정하여 설명한다.

[0263] 제어부(30)는, 예를 들면 CPU(Central Processing Unit), RAM(Random Access Memory), ROM(Read Only Memory) 및 기타(도시하지 않는다)를 포함한다. 제어부(30)는, RAM을 작업 메모리로서 이용하여 기록/재생 장치의 (10)의 각부를 제어한다. 제어부(30)를 기록/재생 장치(10)의 각부에 접속하는 경로는, 번잡함을 피하기 위

해서, 도 35에서는 생략하고 있다.

- [0264] UI(User Interface)부(31)는, 이 기록/재생 장치(10)의 동작을 이용자가 조작하도록 조작자를 제어하는 조작자(operator)를 구비하며, 이용자의 조작에 대응하는 제어 신호를 출력한다. 이 제어 신호는, 제어부(30)에 공급된다. 제어부(30)는, 이용자 조작에 따라 UI부(31)로부터 공급된 제어 신호에 근거하여 프로그램에 의해 실행되는 처리에 의해, 기록/재생 장치(10)의 각부의 동작을 제어한다. 또, UI부(31)는, 간단한 표시부를 가지며, 후술하는 관리 정보처리부(16)로부터 공급되는 정보에 근거하여, 소정의 표시, 예를 들면 기록/재생장치에 있어서의 기록이나 재생 상태를 나타내는 표시를 실시할 수 있다.
- [0265] 예를 들면, UI부(31)에서 실행된 조작에 따라, 제어부(30)는, 기록/재생장치를 제어하여, 기록 매체(32)에 대해서 데이터를 기록하는 동작의 개시 및 정지의 동작이나, 기록 매체(32)로부터 데이터를 재생하는 재생동작을 하도록 한다. 게다가, 편집 동작중에는, 데이터가 기록 매체(32)로부터 재생되며, UI부(31)의 조작에 따르는 위치에서 캡처 분할 처리가 이루어진다.
- [0266] 비디오 엔코더(11)는, 복수 프레임의 비디오 데이터를 저장할 수 있는 버퍼 메모리를 가지며, 공급된 베이스밴드 디지털 비디오 데이터를 버퍼 메모리에 축적하고, 소정의 방식으로, 디지털 비디오 데이터를 압축 인코드 한다. H.264 | AVC에 규정되는 방식으로 압축 인코드가 되는 본 예에서는, 예를 들면, DCT(Discrete Cosine Transform)와 화면내 예측에 근거하여 프레임간 압축을 실시함과 동시에, 움직임 벡터를 이용한 프레임간 압축을 실시하고, 엔트로피(entropy) 인코드를 실시해 압축 효율을 높인다. 비디오 엔코더(11)에 의해 압축 인코드된 디지털 비디오 데이터는, H.264 | AVC 엘리먼트리 스트림(ES)으로서 출력된다.
- [0267] 비디오 디코더(20)는, 복수 프레임의 비디오 데이터를 저장할 수 있는 버퍼 메모리를 가지며, 공급된 압축 비디오 데이터를 버퍼 메모리에 축적하고, 압축 인코드 방식으로 대응하는 복호화 방식에 따라, 압축 비디오 데이터를 디코드하고, 베이스밴드 디지털 비디오 데이터로서 출력한다. 예를 들면, 비디오 엔코더(11)가, H.264 | AVC에 규정되는 방식으로 압축 부호화를 실시하는 이 예에서, 비디오 디코더(20)도 비디오 엔코더(11)에 대응하는 H.264 | AVC에 규정되는 방식으로 디코드 처리를 실시한다. 비디오 디코더(20)는, 후술하는 멀티플렉서 / 디멀티플렉서(13)(이하, MUX / DEMUX(13))에 의해 추출되며, DTS(Decoding Time Stamp) 및 PTS(Presentation Time Stamp)로 표시되는 시각에 근거하여, 디코드 및 출력을 실시할 수 있다. 비디오 디코더(20)에 의해 디코드되어 얻을 수 있는 베이스밴드 디지털 비디오 데이터는, 단자(42)로부터 출력된다.
- [0268] 오디오 엔코더(12)는, 단자(41)로부터 공급된 베이스밴드 디지털 오디오 데이터를 소정의 압축 인코드 방식, 예를 들면 돌비 디지털(Dolby Digital : 등록상표) 방식에 의해 압축 인코드 한다. 오디오 데이터의 압축 인코드 방식은, 이것에 한정되는 것이 아니며, 게다가 오디오 데이터를 압축 인코드 하지 않고, 베이스밴드 데이터가 현재 상태대로 이용되는 일도 생각할 수 있다.
- [0269] 오디오 디코더(21)는, 공급된 압축 오디오 데이터를 압축 인코드 방식에 대응하는 복호화 방식으로 디코드하고, 베이스밴드 디지털 오디오 데이터로서 출력한다. 오디오 엔코더(12)가 돌비 디지털 방식에 의해 압축 인코하는 예에서는, 오디오 디코더(21)도 오디오 엔코더(12)에 대응하는 돌비 디지털 방식에 따르는 복호화 방식으로 압축 오디오 데이터를 디코드한다. 오디오 엔코더(12)는, 비디오 디코더(20)로부터 출력되는 비디오 데이터와 동기하여, 디코드된 오디오 데이터를 단자(43)로부터 출력한다.
- [0270] MUX / DEMUX(13)는, 소정의 다중화 방식에 따라, 각각 압축 인코드 되어 공급된 디지털 비디오 데이터 및 디지털 오디오 데이터를 소정의 압축 인코딩 방식으로 다중화하고, 한 개의 데이터 스트림으로서 출력하는 멀티플렉서 기능과, 디지털 비디오 데이터와 디지털 오디오 데이터가 소정의 다중화 방식에 따라, 다중화되어 있는 데이터 스트림을, 디지털 비디오 데이터와 디지털 오디오 데이터로 분리하여, 디지털 비디오 데이터와 디지털 오디오 데이터를 각각 추출하는, 디멀티플렉서 기능을 가진다.
- [0271] 멀티플렉서 기능은, 예를 들면, MPEG2 시스템에 따라 다중화를 하는 예에서는, MPEG2의 트랜스포트 스트림을 이용하여, 공급된 압축 비디오 데이터 및 압축 오디오 데이터를 시분할로 다중화한다. 예를 들면, MUX / DEMUX(13)는, 버퍼 메모리를 가지며, 공급된 압축 비디오 데이터 및 압축 오디오 데이터를 일단 버퍼 메모리에 저장한다. 버퍼 메모리에 저장된 압축 비디오 데이터는, 소정 사이즈로 각각 분할되어, 헤더가 부가되고, PES(Packetized Elementary Stream) 패킷으로 제공된다. 압축 오디오 데이터도, 이와 같이, 소정 사이즈로 분할되어, 헤더가 부가되며, PES 패킷으로 제공된다. 각 헤더에는, 각 패킷에 저장되는 데이터의 재생 시각을 나타내는 PTS와 복호 시각을 나타내는 DTS(Decoding Time Stamp)와 같은, MPEG2 시스템에 규정되는 소정의 정보가 저장된다. PES 패킷은, 더욱 분할되어 각 패킷은, 트랜스포트 패킷(TS패킷)의 페이로드(payload)내에

포함된다. TS패킷의 헤더에는, 페이로드에 포함된 데이터 종별 등을 식별하기 위한 PID(Packet Identification)이 저장된다.

[0272] 디멀티플렉서 기능은, 멀티플렉서 기능과 반대의 처리를 실시하며, 패킷으로부터 압축 비디오 데이터 및 압축 오디오 데이터를 추출한다. 예를 들면, 공급된 TS패킷의 헤더로부터 PID를 검출하고, TS패킷을 페이로드에 저장되는 데이터에 따라 분류한다. 그리고, 분류된 TS패킷의 각각에 대해, 페이로드(payload)에 저장된 데이터를 추출하고, PES 패킷을 재구성한다. 게다가 PES 패킷의 페이로드에 저장된 압축 비디오 데이터나 압축 오디오 데이터를 추출하고, PES 헤더에 저장된 정보 등에 기초하여 헤더 정보 등을 추출된 압축 비디오 데이터와 압축 오디오 데이터에 부가하고, 비디오 데이터와 오디오 데이터의 각각을 한 개의 엘리먼트리 스트림으로서 출력한다.

[0273] 스트림 버퍼(14)는, MUX / DEMUX(13)(기록시) 또는 기록 재생/제어부 (15)(재생시)로부터 공급된 TS패킷을 일시적으로 저장한다. 스트림 버퍼(14)에 대한 TS패킷의 판독 및 기록의 타이밍을 소정의 방식으로 제어함으로써, 기록 매체(32)에 대한 액세스 속도는 오디오 및 비디오 데이터 인코더 또는 디코드의 신호 처리 속도와 정합된다(matched).

[0274] 기록/재생 제어부(15)는, 기록 매체(32)에 대한 데이터의 기록과 기록 매체(32)로부터의 데이터의 재생을 제어한다. 즉, 기록/재생 제어부(15)는, 예를 들면 제어부(30)와 같은 더 높은 상위 레벨의 단위로부터의 명령에 근거하여, 지정된 어드레스에서 기록 매체(32)에 대한 데이터의 기록과 판독을 실시한다.

[0275] 기록 매체(32)로서는, 예를 들면 기록 가능한 타입의 DVD(Digital Versatile Disc)를 이용할 수 있다. 이것에 한정하지 않고, 기록 매체(32)로서 하드 디스크 드라이브를 이용해도 괜찮고, 반도체 메모리를 기록 매체(32)에 적용하는 것도 가능하다. 또, 기록 매체(32)로서 보다 대용량을 실현한 Blu-rayDisc(블루 레이 디스크: 등록상표)를 적용하는 것도 생각할 수 있다.

[0276] 관리 정보처리부(16)는, 예를 들면 CPU, 작업 메모리로서의 RAM 및 소정의 프로그램(도시하지 않음)에 관한 데이터가 미리 기억되는 ROM으로 구성된다. 이것에 한정하지 않고, 관리 정보처리부(16)에서는, 예를 들면 제어부(30)에 있어서의 프로그램 처리에 의해 관리 정보처리부(16)의 기능을 실현하는 것도 가능하다. 이 경우, 예를 들면, 제어부(30)내에 포함된 RAM이 휘발성 메모리(17)로서 이용됨과 동시에, 불휘발성 메모리(18)가 제어부(30)에 접속된다.

[0277] 관리 정보처리부(16)는, 예를 들면 휘발성 메모리(17)를 작업 메모리로서 이용하고, 상술한 인덱스 파일 "index.bdmv", 무비 오브젝트 파일 "MovieObject.bdmv", 플레이 리스트 파일 "xxxxx.mpls" 및 클립 정보 파일 "zzz zz.clpi"에 관한 처리를 실시한다. 예를 들면, 이 발명의 실시의 한 형태에 의한 플레이 리스트 마크의 추가 삽입 처리는, 제어부(30)의 제어에 근거하여 이 관리 정보처리부(16)에 의해 행해진다.

[0278] 이러한 구성을 가지는 기록 재생장치(10)에 있어서의, 기록시의 동작에 대해 설명한다. 또한, 이하에서는, 비디오 데이터는, H.264 | AVC에 준한 방식으로 압축 인코딩되며, 오디오 데이터는, AAC 방식으로 압축 인코딩되는 것으로 가정한다. 베이스밴드 디지털 비디오 데이터가 단자(40)로부터 기록/재생 장치(10)에 입력되어 비디오 엔코더(11)에 공급된다. 예를 들면, UI부(31)에 대해서 기록 개시의 조작이 이루어지면, 비디오 엔코더(11)는, 공급된 디지털 비디오 데이터의 압축 인코딩을 개시한다. 비디오 엔코더(11)는, 베이스밴드 디지털 비디오 데이터를 압축 인코딩하여, H.264 | AVC의 엘리먼트리 스트림(ES)으로서 출력한다. 이 엘리먼트리 스트림은, MUX / DEMUX(13)에 공급된다.

[0279] 베이스 밴드 디지털 오디오 데이터가 단자(41)로부터 기록/재생 장치(10)에 입력되어, 오디오 엔코더(12)에 공급된다. 오디오 엔코더(12)는, 상술의 UI부(31)에 대한 기록 개시의 조작에 대응하는 타이밍에 공급된 오디오 데이터의 압축 인코딩을 개시한다. 오디오 엔코더(12)에서 압축 인코딩된 디지털 오디오 데이터는, MUX / DEMUX(13)에 공급된다.

[0280] MUX / DEMUX(13)는, 각각 압축 인코딩되어 공급된 디지털 비디오 데이터 및 디지털 오디오 데이터를 소정의 방식으로 다중화하고, 1개의 데이터 스트림으로서 출력한다. 예를 들면, MUX / DEMUX(13)는, 버퍼 메모리를 가지며, 공급된 압축 비디오 데이터 및 압축 오디오 데이터를 일단 버퍼 메모리에 저장한다.

[0281] 버퍼 메모리에 저장된 압축 비디오 데이터는, 소정 구조로 분할되어 헤더가 부가되고, PES(Packetized Elementary Stream) 패킷화 된다. 압축 오디오 데이터도 이와 같이 소정 사이즈로 분할되어 헤더가 부가되어 PES 패킷화 된다. 헤더에는, 패킷에 저장되는 데이터의 재생 시각을 나타내는 PTS나 복호 시각을 나타내는 DTS(Decoding Time Stamp)와 같은, MPEG2 시스템에 규정되는 소정의 정보가 저장된다. PES 패킷은, 더욱 분

할되어 트랜스포트 패킷(TS패킷)의 페이로드에 포함된다. TS패킷의 헤더에는, 페이로드에 포함된 데이터를 식별하기 위한 PID(Packet Identification)가 저장된다. 멀티플렉서로 기능하는 MUX / DEMUX(13)로부터 출력된 TS패킷은, 스트림 버퍼(14)에 일단 축적된다.

[0282] MUX / DEMUX(13)는, 실제로는, 소정 사이즈의 소정의 헤더를 TS패킷에 부가하여, 헤더가 부가된 TS패킷들을 출력한다. 각 TS패킷에 대해 소정의 헤더를 부가한 패킷을, 소스 패킷이라고 부른다.

[0283] 기록/재생 제어부(15)는, 스트림 버퍼(14)에 축적된 데이터량을 감시하고, 스트림 버퍼(14)에 소정량 이상의 데이터가 축적되면, 스트림 버퍼(14)로부터 기록 매체(32)의 기록 단위마다 데이터를 판독하고, 기록 매체(32)에 그 판독한 데이터를 기록한다.

[0284] 관리 정보처리부(16)는, 기록 데이터에 근거하여, 휘발성 메모리(17)를 작업 메모리로서 이용함으로써, 상술한 인덱스 파일 "index.bdmv", 무비 오브젝트 파일 "MovieObject.bdmv", 플레이 리스트 파일 "xxxxx.mpls" 및 클립 정보 파일 "zzzzz.clpi"에 저장하기 위한 정보를 생성한다. 생성된 정보는, 소정의 타이밍에서 기록 매체(32)에 기록된다.

[0285] 일례로서, 관리 정보처리부(16)는, MUX / DEMUX(13)로부터 기록 데이터의 시각 정보를 취득함과 동시에, 기록/재생 제어부(15)로부터 기록 데이터의 기록 매체(32)에 있어서의 어드레스 정보를 취득하고, 취득된 이러한 시각 정보 및 어드레스 정보에 근거하여 엔트리 "PTSEStart"가 생성된다.또, UI부(31)에 대한 기록 개시, 기록 종료의 조작에 따라 제어부(30)로부터 출력되는 제어 신호와 MUX / DEMUX(13) 및 기록/재생 제어부(15)로부터의 기록 데이터에 관한 정보에 근거하여, 플레이 리스트 파일 "xxxxx.mpls"의 생성 또는 갱신, 클립 정보 파일 "zzzzz.clpi"의 생성 등이 행해진다.

[0286] 기록이 개시될 때에, 기록 개시 위치에 플레이 리스트 마크를 할당하도록 기록/재생장치가 구성되어 있는 경우, UI부(31)에 대한 기록 개시 조작에 따라 생성되는 선두의 프레임에 대응하는 재생 시각을, 필드 "MarkTimeStamp"의 값으로 가지는 플레이 리스트 마크가, 관리 정보처리부(16)에 의해 작성되며, 플레이 리스트에 그 생성된 플레이 리스트 마크를 추가한다.

[0287] 기록/재생 장치(10)의 재생시의 동작에 대해 설명한다. 기록 매체(32)가 예를 들면 기록 재생장치에 대해서 장착되면, 기록 매체(32)로부터, 인덱스 파일 "index.bdmv", 무비 오브젝트 파일 "MovieObject.bdmv"이 판독되어, 관리 정보처리부(16)로 전달된다. 관리 정보처리부(16)는, 이러한 정보를, 휘발성 메모리(17)에 기억한다. 제어부(30)는, 관리 정보처리부(16)로부터 휘발성 메모리(17)에 기억된 정보를 취득하고, 취득된 정보에 근거하여, 예를 들면, UI부(31)내에 포함된 표시부에, 기록 매체(32)에 기록되어 있는 클립에 관한 정보를 표시시킨다. 이용자는, 이 정보에 근거하여, UI부(31)에 대해서 소정의 조작을 실시하여, 기록 매체(32)에 기록되어 있는 클립의 재생을 지시할 수 있다. 여기에서는, 간편하게, 클립의 선두로부터 재생을 개시시키는 것으로 가정한다.

[0288] 제어부(30)는, UI부(31)에 대한 조작에 따라, 관리 정보처리부(16)로부터 플레이 리스트 파일의 정보를 취득하고, 이 플레이 리스트 파일에 근거하여, 기록/재생 제어부(15)에 대해서, 해당 플레이 리스트 파일에 저장되는 플레이 아이템에 의해 참조되는 클립 정보 파일과 대응하는 클립 AV스트림 파일을 기록 매체(32)로부터 판독하도록 명령을 내린다. 기록/재생 제어부(15)는, 이 명령에 따라, 기록 매체(32)로부터 클립 정보 파일과 클립 AV스트림 파일을 판독한다. 클립 AV스트림 파일은, 기록 매체(32)로부터 각 TS패킷에 대해 판독되며, 기록/재생 제어부(15)를 통해 스트림 버퍼(14)에 축적된다.

[0289] 실제로는, 기록 매체(32)로부터 판독되는 TS패킷은, 본래의 TS패킷에 대해서 소정 사이즈의 헤더가 부가되어 얻어지는 소스 패킷이다.

[0290] MUX / DEMUX(13)는, 스트림 버퍼(14)에 축적된 데이터량을 감시하고, 스트림 버퍼(14)에 소정량 이상의 TS패킷이 축적되면, 스트림 버퍼(14)로부터 비디오 디코더(20)가 디코드에 필요로 하는 양의 데이터를, TS패킷 단위로 판독한다. 판독된 TS패킷은, MUX / DEMUX(13)에 공급되며, 버퍼 메모리에 일단 저장된다. MUX / DEMUX(13)는, PID에 근거하여 데이터의 종류에 따라 TS패킷을 분류하며 각 TS패킷내의 페이로드에 저장된 데이터로부터 PES 패킷을 재구성한다. 게다가, MUX / DEMUX(13)는, PES 패킷의 페이로드로부터 데이터를 추출하고, PES 헤더의 정보에 근거하여, DTS와 PTS와 같은 디코드 시각과 재생 시각을 지정하는 정보를 포함하는, 소정의 헤더 정보 등을 데이터에 부가하여, 압축 인코딩된 비디오 데이터 및 오디오 데이터(이후로는, "비디오 데이터 ES" 및 "오디오 데이터 ES"라고 칭한다.)를 엘리먼트리 스트림으로 각각 생성한다.

[0291] MUX / DEMUX(13)는, 압축 인코딩된 비디오 데이터의 엘리먼트리 스트림을 비디오 디코더(20)에 공급한다. 비디오

오 디코더(20)는, 공급된 압축 비디오 데이터를 버퍼 메모리에 축적하고, 소정 픽처 분의 데이터가 축적되면, 버퍼 메모리에 축적된 데이터에 대해서 디코드 처리를 개시한다. 비디오 디코더(20)는, 디코드된 비디오 데이터를, 예를 들면 도시되지 않는 시스템 시계로부터 공급되는 STC(System Time Clock)에 근거하여, PTS에 따라 프레임 타이밍에서 차례차례 출력한다.

[0292] 한편, MUX / DEMUX(13)는, 오디오 데이터 ES 즉, 압축 오디오 데이터를, 오디오 디코더(21)에 공급하고, 오디오 디코더(21)는, 소정의 디코드 처리를 압축된 오디오 데이터에 실행한다. 오디오 디코더(21)는, 비디오 디코더(20)로부터 출력되는 비디오 데이터와 동기하여, 디코드된 오디오 데이터를 출력한다.

[0293] 기록/재생장치는, 기록 매체(32)에 기록된 클립에 대한 챕터 분할 편집을 실시할 수 있다. 위에서 설명한 바와 같이, 기록 매체(32)로부터 클립을 재생하는 동안에, UI부(31)에 대해서 소정의 조작을 실시하고, 챕터 분할을 실시하는 위치를 지정한다. 챕터 분할의 위치를 지정하는 방법으로서, 클립을 통상대로, 예를 들면, 순방향으로 1배속 재생으로 재생하면서, 소망하는 위치에서 클립 분할의 조작을 실시하는 방법, 고속 재생, 슬로우 재생, 또는 프레임 단위의 재생과 같은 가변속 재생을 이용하여 분할 위치를 프레임 단위로 검색하여 챕터 분할의 조작을 실시하는 방법이 생각될 수 있다.

[0294] 분할 위치가 지정되면, 도 24를 참조하여 이미 기술된 처리에 따라, 지정된 분할 위치에 대응하는 현재의 재생 시각과 플레이 리스트중의 다른 플레이 리스트 마크와의 위치 관계가 판단된다. 그와 함께, 도 31을 이용하여 이미 설명한 처리에 따라, 플레이 리스트 마크를 할당하는 값으로서 엔트리 "PTSEStart" 및 GOP 경계의 어느 쪽의 값을 이용할 것인지를 판단한다.

[0295] 플레이 리스트 마크를 할당하기 위한 현재 시각으로서, 엔트리 "PTSEStart"의 값을 이용하고, 분할 위치를 프레임 단위로 임의로 지정할 때, 지정된 만큼 비율 위치는 반드시 엔트리 "PTSEStart"로 표시되는 프레임이라고 할 수 없다. 이 경우, 예를 들면 지정된 위치에 가장 가까운 엔트리 "PTSEStart"의 값을, 지정된 분할 위치 이전 또는 이후에 대응하는 "PTSEStart"의 값으로서 이용할 수 있다. 게다가, 이에 한정하지 않고, 지정된 위치의 바로 이전 또는 이후의 엔트리 "PTSEStart"의 값은, 지정된 분할 위치에 대응하는 엔트리 "PTSEStart"의 값으로서 이용할 수 있다.

[0296] 현재 재생중의 위치에 대응하는 GOP가 클로즈드(closed) GOP라고 판단된 경우, 클로즈드 GOP의 선두의 프레임의 재생 시각은, 예를 들면 MUX / DEMUX(13)로부터 취득할 수 있다. 이에 한정하지 않고, 엔트리 "PTSEStart"의 값을 기점으로 하여, 비디오 디코더(20)의 버퍼 메모리에 축적된 데이터와 STC와에 근거하여, 클로즈드 GOP의 선두의 프레임의 재생 시각을 구하는 일도 생각할 수 있다.

[0297] 도 36은, 위에서 설명한 바와 같이 하여 설치된 플레이 리스트 마크에 근거하는 챕터 점핑 처리의 일례의 방법을 나타내는 플로차트이다. 일례로서, 위에서 설명한 바와 같이, 플레이 리스트 마크가 설치되어 챕터 분할 편집된 플레이 리스트를 재생하는 동안에, 예를 들면 UI부(31)에 대해서 챕터 점핑을 실시하는 조작이 이루어진다. 제어부(30)는, 이 조작에 따라, 먼저, 현재 재생중의 위치에 대응하는 시각 정보를 구한다(스텝 : S80). 예를 들면, 제어부(30)는, MUX / DEMUX(13)와 통신을 실시하며, MUX / DEMUX(13)에 의해 분리되는 비디오 데이터의 엘리먼트 스트림에 포함되는 PTS와 MUX / DEMUX(13)의 STC에 근거하여, 비디오 디코더(20)에 의해 디코드되는 임의의 프레임의 재생 시각에 관한 정보를 취득할 수 있다.

[0298] 현재 재생중의 위치에 대응하는 시각 정보를 구하는 방법은, 이것에 한정되지 않는다. 예를 들면, 재생중에, 항상 클립 정보 파일중의 블록 "blKEPMap()" (도 18참조)의 정보를 이용하는 엔트리 "PTSEStart"에 근거하여 어드레스 변환을 실시하는 방법이 생각된다. 재생중에는, 다음의 GOP의 데이터를 관독할 필요가 있으므로, 이 방법에 의하면, 다음의 GOP의 데이터를 용이하게 취득하는 것이 가능해진다. 또, 현재의 재생 위치와 플레이 리스트 마크의 위치와의 관계를 항상 갱신하면서 재생을 실시하여, UI부(31)의 표시 화면에 현재 재생중의 챕터의 정보를 표시하는 것이 가능해진다.

[0299] 현재 재생중의 위치에 대응하는 시각 정보가 구해지면, 제어부(30)에 의해, 다음의 스텝(S81)에서 현재 재생되는 플레이 리스트의 블록 "blkPlayListMark()"으로부터 시각 정보에 의해 표시되는 시각과 가장 가까운 플레이 리스트 마크를 재생 방향에서 검색한다.

[0300] 가장 인접한 플레이 리스트 마크가 존재한다고 간주되면(스텝 : S82), 처리는 스텝(S83)으로 이행되어, 제어부(30)에 의해, 스텝(S81)에서 검색된 플레이 리스트 마크에 근거하여, 해당 플레이 리스트 마크에 대응하는 엔트리 "PTSEStart"가 검색된다.

[0301] 이 실시의 한 형태에서는, 플레이 리스트 마크는, 엔트리 "PTSEStart"의 위치 또는 GOP경계 위치의 어느 쪽인

가에 설치된다. 따라서, 플레이 리스트 마크에 대응하는 엔트리 "PTSEPEnd"는, 플레이 리스트 마크의 재생 시각을 나타내는 값 "MarkTimeStamp"와 값이 일치하는 엔트리 "PTSEPEnd"가 존재하면, 플레이 리스트 마크에 대응하는 엔트리 "PTSEPEnd"는 값 "MarkTimeStamp"와 값이 일치하는 엔트리 "PTSEPEnd"가 된다. 한편, 플레이 리스트 마크의 재생 시각을 나타내는 값 "MarkTimeStamp"와 값이 일치하는 엔트리 "PTSEPEnd"가 존재하지 않고, 플레이 리스트 마크가 GOP 경계에 설치되며, GOP가 클로즈드 GOP라는 것을 인식하는 경우에는, 값 "MarkTimeStamp"로부터 재생 시각을 재생 방향으로 진행할 때에, 플레이 리스트 마크에 대응하는 엔트리 "PTSEPEnd"는 최초로 나타나는 엔트리 "PTSEPEnd"로 간주된다. 게다가, 플레이 리스트 마크의 재생 시각을 나타내는 값 "MarkTimeStamp"와 값이 일치하는 엔트리 "PTSEPEnd"가 존재하지 않는 경우에, 플레이 리스트 마크가 GOP 경계에 설치되어 있지 않거나, 또는, GOP가 클로즈드 GOP인 것을 모르는 경우에는, 재생 방향과는 역방향으로 진행하고 다음에 나타나는 엔트리 "PTSEPEnd"로 간주된다.

[0302] 다음의 스텝(S84)에서, 제어부(30)는, 재생중의 플레이 아이템에 대응하는 클립 정보 파일내의 블록 "blkEPMAP()" (도 18-도 21참조)에 근거하여, 스텝(S83)에서 검색된 엔트리 "PTSEPEnd"에 대응하는 어드레스 정보를 취득한다. 그리고, 제어부(30)는, 기록/재생 제어부(15)에 대해서, 기록 매체(32)로부터 해당 어드레스 정보에 의해 표시되는 어드레스에서 데이터의 판독을 실시하도록 명령한다.

[0303] 기록 매체(32)로부터 판독된 데이터는, 기록/재생 제어부(15)로부터 스트림 버퍼(14)를 통해 MUX / DEMUX(13)에 공급되고, 패킷을 이용하여 시분할 다중화된 스트림을 데이터 종류에 따라 분리한다. 분리된 데이터중에서, 비디오 데이터, 즉 압축 비디오 데이터의 엘리먼트리 스트림은, 비디오 디코더(20)에 공급된다. 비디오 디코더(20)는, 공급된 압축 비디오 데이터의 엘리먼트리 스트림의 디코드를 개시한다(스텝 : S85). 그리고, 플레이 리스트 마크로 나타나는 프레임이 디코드되어 출력 가능해지면, 그 프레임으로부터의 출력이 이루어져 표시가 개시된다(스텝 : S86).

[0304] 이 때, 플레이 리스트 마크가 엔트리 "PTSEPEnd" 위치에 설치되고 있는 경우에는, GOP내의 해당 엔트리 "PTSEPEnd"에 대응하는 픽처보다 전의 픽처는, 디코드되지만 표시는 실시하지 않게 된다. 플레이 리스트 마크가 GOP 경계 위치에 설치되어 있는 경우에는, GOP의 선두 위치로부터 표시가 개시된다.

[0305] 재생 시각의 마지막에 설치된 플레이 리스트 마크보다 시간적으로 늦은 위치로 챕터 점핑 명령이 전달될 수 있다. 이 경우, 현재 재생중의 위치에 대해서 재생 방향을 향해서 가장 가까운 플레이 리스트 마크가, 재생중의 플레이 리스트중에 존재하지 않게 된다. 이 경우, 챕터 점핑 지시가 무시된다. 이에 한정하지 않고, 예를 들면 도 36의 스텝(S87)에 일레가 도시된 바와 같이, 해당 플레이 리스트의 종단까지 점프하도록 해도 괜찮다.

[0306] 다음에, 이 발명의 실시의 한 형태의 다른 예에 대해 설명한다.

[0307] 상술에서는, 이 발명이 단체의 기록 장치에 적용된 예에 대해 설명했다(도 35 참조). 이에 대해, 이 실시의 한 형태의 다른 예에서는, 이 발명을, 촬상 소자와 피사체로부터의 빛을 촬상 소자에 입사시키는 광학계를 가지며, 촬상 소자로 촬상된 촬상 신호에 근거해 비디오 데이터를 기록 매체에 기록하는, 비디오 카메라 장치에 적용했다.

[0308] 도 37은, 이 발명의 실시의 한 형태의 다른 예에 의한 비디오 카메라 장치(100)의 일례의 구성을 나타낸다. 비디오 카메라 장치(100)에서, 기록계의 구성은, 도 35를 이용해 설명한 기록 장치의 구성을 거의 그대로 적용할 수 있으므로, 도 35와 공통되는 부분에는 동일한 부호를 부여하고, 상세한 설명을 생략한다.

[0309] 도 37의 구성에서, 카메라부(50)는, 영상 신호에 관한 구성으로서 광학계(51), 촬상 소자(52), 촬상 신호 처리부(53), 카메라 제어부(54), 비디오 신호 처리부(58) 및 표시부(55)를 가지며, 음성 신호에 관한 구성으로서 마이크로폰(MIC)(56), 음성 신호 처리부(57) 및 스피커부(SP60)를 가진다. 제어부(30)는, 카메라부(50)의 각 부와의 사이에 각종 제어 신호나 정보의 교환을 실시하고, 카메라부(50)의 동작을 제어한다. 또, 제어부(50)는, 이용자 조작에 따라 UI부(31)로부터 공급되는 제어 신호에 근거하여, 카메라부(50)의 동작을 제어한다.

[0310] 한 장치가 비디오 카메라 장치(100)로서 구성되는 경우, 기록 개시 조작 및 기록 정지 조작은, 예를 들면, UI부(31)에 설치된 단일의 기록 스위치를 이용하여 해당 기록 스위치가 가압될 때마다 기록 개시 및 기록 정지가 교대로 지시받는 것이 일반적이다. 또, 이 비디오 카메라 장치(100)에서는, 기록 매체(20)로서 Blu-ray Disc 나 기록 가능한 타입의 DVD와 같은 디스크 기록 매체를 적용하는 것으로 가정한다.

[0311] 카메라부(50)에서, 광학계(51)는, 피사체로부터의 빛을 촬상 소자(52)로 유도하는 렌즈계, 조임 조정 기구, 포커스 조정 기구, 줌 기구, 셔터 기구 등을 갖춘다. 조임 조정 기구, 포커스 조정 기구, 줌 기구 및 셔터 기구

의 동작은, 제어부(30)로부터 공급되는 제어 신호에 근거하여, 카메라 제어부(54)에 의해 제어된다.

- [0312] 촬상 소자(52)는, 예를 들면 CCD(Charge Coupled Device)로 구성되며, 광학계(51)를 통해 조사된 빛을 광전 변환에 의해 전기신호로 변환하고, 소정의 신호 처리를 실시하여 촬상 신호로 출력한다. 촬상 신호 처리부(53)는, 촬상 소자로부터 출력된 촬상 신호에 대해서 소정의 신호 처리를 실시하여, 베이스밴드 디지털 비디오 데이터로서 출력한다. 예를 들면, 촬상 신호 처리부(53)는, 촬상 소자(52)로부터 출력된 촬상 신호로부터, CDS(Correlated Double Sampling)회로를 이용하여 화상 정보를 가지는 신호만을 샘플링 함과 동시에, 샘플된 신호로부터 노이즈를 제거하고, AGC(Auto Gain Control) 회로에 의해 이득을 조정한다. 그리고, 촬상 신호 처리부(53)는, A/D변환에 의해 디지털 신호로 변환한다.
- [0313] 게다가, 촬상 신호 처리부(53)는, 촬상 소자(52)로부터 출력된 촬상 신호의 정보를 제어부(30)에 보낸다. 제어부(30)는, 이 정보에 근거하여 광학계(51)를 제어하기 위한 제어 신호를 생성하고, 카메라 제어부(54)에 공급한다. 카메라 제어부(54)는, 이 제어 신호에 근거하여 포커스 조정 기구나 조임 조정 기구 등의 제어를 실시한다.
- [0314] 비디오 신호 처리부(58)는, 공급된 디지털 신호에 대해서 소정의 신호 처리를 실시한다. 예를 들면, 비디오 신호 처리부(58)는, 공급된 디지털 신호에 대해서 감지와 관련된 신호 처리를 실시하고, R(적색), G(녹색) 및 B(청색)의 각 색의 성분을 추출한다. 그리고, 추출된 각 색성분에 근거하여, γ 보정이나 화이트 밸런스 보정 등의 처리를 실시하여, 최종적으로 1개의 베이스밴드 디지털 비디오 데이터로서 출력한다.
- [0315] 표시부(55)는, 예를 들면 LCD(Liquid Crystal Display)를 표시 소자로서 이용하여, 비디오 신호 처리부(58)로부터 공급된 디지털 비디오 데이터에 근거하는 표시를 실시할 수 있다. 표시부(55)는, 촬영시에는 촬영 화상의 모니터로서 이용되며, 재생시에는, 재생 화상을 표시할 수 있다.
- [0316] 음성 신호 처리부(57)는, 예를 들면, 마이크로폰(MIC)(56)로부터 공급되는 아날로그 음성 신호를 A/D변환하여, 디지털 오디오 데이터로 바꾸며, 노이즈 제거나 음질 보정 등과 같은 소정의 음성 신호 처리를 실시하여, 베이스밴드 디지털 오디오 데이터로서 출력한다. 또, 음성 신호 처리부(57)는, 공급되는 디지털 오디오 데이터에 대해서 음질 보정이나 음량 조정 등과 같은 소정의 음성 신호 처리를 실시하여, D/A변환을 통해 아날로그 음성 신호로 변환하고, 증폭 처리 등을 아날로그 음성 신호에 실시하여, 최종 아날로그 음성 신호를 스피커부(SP)(60)에 공급한다.
- [0317] 촬영시에는, 광학계(51)를 통해 촬상 소자(52)에 입사된 빛이 광전 변환에 의해 전기신호로 변환되어 촬상 신호로서 출력된다. 촬상 신호는, 촬상 신호 처리부(53)에서 소정의 신호 처리를 거쳐, A/D변환되어 디지털 비디오 신호로서 출력된다. 이 디지털 비디오 신호는, 비디오 신호 처리부(58)에 공급된다. 비디오 신호 처리부(58)는, 공급된 디지털 비디오 신호에 대해서 화질 보정 등의 소정의 신호 처리를 실시하며 디지털 비디오 데이터로서 출력한다. 이 디지털 비디오 데이터는, 기록/재생 장치(10)에 공급되어 단자(40)에 입력된다.
- [0318] 또, 비디오 신호 처리부(58)는, 촬상 신호 처리부(53)로부터 공급된 디지털 신호에 근거하여, 표시부(55)에 표시하기 위한 디지털 비디오 데이터를 생성한다. 게다가, 비디오 신호 처리부(58)는, 제어부(30)와 통신을 실시하며, 제어부(30)에 의해 소정의 방식으로 생성된 표시 제어 신호에 근거하는 화상을 생성할 수 있다. 이 디지털 비디오 데이터나 화상은, 표시부(55)에 공급되어 표시된다.
- [0319] 한편, 마이크로폰(56)으로부터 출력된 음성 신호는, 음성 신호 처리부(57)에 의해 노이즈 제거, 리미터 처리, 음질 보정 등과 같은 소정의 신호 처리를 거치며, A/D변환을 통해 디지털 오디오 데이터로서 변환되어 출력된다. 이 디지털 오디오 데이터는, 기록/재생 장치(10)에 공급되어 단자(41)에 입력된다.
- [0320] 기록 정지상태에서 UI부(31)에 설치된 기록 스위치가 가압되면, 기록 개시를 지시하는 제어 신호가 UI부(31)로부터 제어부(30)에 공급되며, 제어부(30)의 제어에 근거하여, 카메라부(50)로부터 출력된 베이스밴드 디지털 비디오 데이터 및 디지털 오디오 데이터의 기록이 기록 매체(32)에 개시된다.
- [0321] 즉, 도 35를 이용해 이미 설명한 것처럼, 제어부(30)의 제어에 근거하여, 비디오 엔코더(11) 및 오디오 엔코더(12)의 동작이 개시되어 비디오 데이터 및 오디오 데이터가 각각 비디오 엔코더(11) 및 오디오 엔코더(12)로 압축 인코드되며, 멀티플렉서(13)는 최종 비디오와 오디오 데이터를 패킷화하고 다중화하여 AV스트림 데이터로 만든다. AV스트림 데이터는, 스트림 버퍼(14)를 통해, 기록 제어부(15)에 공급되어, 클립 AV스트림 파일로서 기록 매체(32)에 기록된다.
- [0322] 여기서, 비디오 카메라 장치(100)가 기록 개시에 따라 기록 개시 위치에 자동적으로 플레이 리스트를 할당하도

록 구성되어 있는 경우, 도 32를 이용해 설명한 처리에 따라, 기록 개시 위치에 대응하는 위치에 자동적으로 플레이 리스트 마크가 할당된다. 실제로는, 플레이 리스트 파일에 저장되는 정보는, 예를 들면 휘발성 메모리(17)에 의해 생성된다. 이 플레이 리스트 마크도, 이 시점에서는, 휘발성 메모리(17)내의 정보이다.

[0323] UI부(31)에 제공된 기록 스윗치가 다음에 가압되면, 기록이 정지되어 클립 정보 파일의 작성이나, 플레이 리스트 파일의 갱신을 실행한다. 관리 정보처리부(16)는, 멀티플렉서(13) 및 기록/재생 제어부(15)로부터의 정보에 근거하여, 기록 매체(32)에 기록되는 각각의 클립 AV스트림 파일에 대응하는 클립 정보 파일을 작성한다. 또, 관리 정보처리부(16)는, 해당 클립 정보 파일을 참조하는 플레이 아이템을 생성하고, 이미 플레이 리스트가 존재하는 경우에는, 생성된 플레이 아이템을 해당 플레이 리스트에 추가함과 동시에, 플레이 리스트에 플레이 리스트 마크의 정보를 저장한다.

[0324] 재생시에, 기록 매체(32)가 비디오 카메라 장치(100)에 장착되면, 기록 매체(32)에 기록되어 있는 인덱스 파일, 무비 오브젝트 파일 등의 파일이 판독되어 관리 정보처리부(16)에 공급된다. 제어부(30)는, 관리 정보처리부(16)로부터 인덱스 파일의 정보를 취득하고, 취득된 정보에 근거하여 소정의 메뉴 화면을 표시하기 위한 표시 제어 신호를 생성한다. 이 표시 제어 신호는, 비디오 신호 처리부(58)에 공급되어 표시부(55)에 표시된다. 이 표시에 따라 UI부(31)에 대해서 이용자가 소정의 조작을 실시하면, 예를 들면, 기록 매체(32)로부터 재생을 지시받은 플레이 리스트 파일이 판독되며, 이 플레이 리스트 파일의 기술에 따라, 기록 매체(32)에 기록된 클립의 재생이 이루어진다.

[0325] 즉, 도 35를 이용해 이미 설명한 것처럼, 제어부(30)는, UI부(31)에 대한 조작에 따라 관리 정보처리부(16)로부터 플레이 리스트 파일의 정보를 취득하고, 취득된 정보에 근거하여, 기록/재생 제어부(15)에 대해서 기록 매체(32)로부터 클립 정보 파일이나 클립 AV스트림 파일을 판독하도록 명령을 내린다.

[0326] 기록 매체(32)로부터 판독된 클립 AV스트림 파일은, 스트림 버퍼(14)를 통해 MUX / DEMUX(13)에 공급되며, 패킷의 헤더 정보 등에 근거하여 분리되어, 압축 비디오 데이터와 압축 오디오 데이터로서 추출된다. 압축 비디오 데이터는, 비디오 디코더(20)에 공급되어 디코딩되고, 예를 들면 PTS에 따라 단자(42)로부터 출력된다. 압축 오디오 데이터는, 오디오 디코더(21)에 공급되어 디코딩되어, 비디오 디코더(20)로부터 출력되는 비디오 데이터와 동기하여 단자(43)로부터 출력된다.

[0327] 단자(42)로부터 출력된 비디오 데이터는, 비디오 신호 처리부(58)에 의해 소정의 신호 처리를 거치며 표시부(55)에 공급된다. 표시부(55)는, 공급된 비디오 데이터에 근거하여 화상을 표시한다. 단자(43)로부터 출력된 오디오 데이터는, 음성 신호 처리부(57)에 의해 소정의 신호 처리를 거치며, 증폭 처리 등도 거친 후에, 스피커부(60)에 공급된다.

[0328] 비디오 카메라 장치(100)는, 클립의 챕터 분할 편집을 실시할 수 있다. 챕터 분할 편집 처리는, 상술한 실시의 한 형태에 의한 처리 동일하므로, 번잡함을 피하기 위해서 설명을 생략한다. 또, 챕터 점핑 처리에 대해서도, 상술한 실시의 한 형태에 의한 처리와 변함없기 때문에, 설명을 생략한다.

[0329] 상술한 실시예에서는, 도 35에 도시한 기록 장치나 도 37에 도시한 비디오 카메라 장치(100)의 기록/재생 장치(10)가 하드웨어적으로 구성되도록 설명했지만, 이것은 하드웨어에만 한정되지 않는다. 즉, 기록/재생 장치(10)는, 소프트웨어로서 구성하는 것도 가능하다. 이 경우, 소프트웨어는, 예를 들면 제어부(30)가 가지는 도시되지 않는 ROM에 미리 기억된다. 이것에 한정하지 않고, 기록/재생 장치(10)를, 퍼스널 컴퓨터 등의 컴퓨터 장치에 구성하는 것도 가능하다. 이 경우에는, 기록/재생 장치(10)를 컴퓨터 장치에 실행시키는 소프트웨어는, CD-ROM나 DVD-ROM라고 하는 기록 매체에 기록되어 제공된다. 컴퓨터 장치가 네트워크에 접속 가능한 경우, 인터넷 등의 네트워크를 통해 해당 소프트웨어를 제공하는 것도 가능하다.

[0330] 게다가, 상기한 설명에서는, 본 발명이 기록 매체에 콘텐츠 데이터 등을 기록하는 기록 장치 또는 기록 재생장치에 적용되도록 설명했지만, 이에도 한정되지 않는다. 본 발명은, 기록 매체에 대한 콘텐츠 데이터의 기록 수단이나 기록 매체로부터의 콘텐츠 데이터의 재생 수단을 가지지 않는, 예를 들면 콘텐츠 데이터 및 해당 콘텐츠 데이터의 재생 제어 정보(인덱스 파일, 무비 오브젝트, 플레이 리스트 및 클립 정보 파일 등)를 편집하는 편집 장치 등에도 적용 가능한 것이다. 이 경우에는, 예를 들면, 기록 매체로부터의 데이터의 판독이나 기록 매체에 대한 기록은, 편집 장치에 접속된 기록 재생장치에 의해 실시될 수 있다.

[0331] 본 발명의 첨부된 청구항과 그와 동등한 것들의 범위내에서 여러 가지 수정과 결합, 소결합 및 변경들이 설계요구 및 다른 인자에 따라 당업자들에 의해 발생할 수 있는 것은 물론이다.

발명의 효과

- [0332] 본 발명의 제 1, 제 2 및 제 3의 실시예에 따르면, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 인코딩하며, 그것을 기록매체에 기록한다. 재생순서에서 가장 인접한 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹들중 한 그룹에 대해서 상기 역세스 위치를 지정할 경우에는, 상기 복호가능한 그룹의 선두의 위치를 나타내는 상기 마크 정보를 생성한다. 그러므로, 기록 매체로부터 비디오 데이터를 재생했을 때에, 재생순서에서 바로 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹으로의 점프를 지정했을 경우, 점프 목적지에서 표시 지연을 줄일 수 있다.
- [0333] 제 4, 제 5 및 제 6의 실시예에 따르면, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 압축 인코딩된 비디오 데이터를 기록 매체내에 기록한다. 그리고, 기록 매체로부터 압축 인코딩된 비디오 데이터를 읽어내 복호화한다. 비디오 데이터에 대해서, 재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호가능한 그룹들중 한 그룹에 대해서 역세스 위치를 지정할 경우에는, 그룹의 선두의 위치를 나타내는 마크 정보를 생성하므로, 기록 매체로부터 재생된 비디오 데이터에 대해서, 재생순서에서 바로 직전의 그룹의 정보를 이용하지 않고 복호 가능한 그룹으로의 점프를 지정했을 경우, 점프 목적지에서 표시 지연을 줄일 수 있다.
- [0334] 제 7, 제 8 및 제 9의 실시예에 따르면, 적어도 1개의 독립적으로 복호가 가능한 픽처를 포함한 복수개의 픽처들로 각각 구성되는 그룹들의 각각에 대해, 예측 인코딩에 근거하여 프레임간 압축을 이용하여 비디오 데이터를 압축 코딩화한다. 상기 비디오 데이터에 대해서 상기 각 그룹 단위의 역세스 위치를 나타내는 마크 정보를 생성되어, 재생순서에서 직전의 그룹의 정보를 이용하지 않고 복호화된 그룹들중 한 그룹에 대해서 상기 역세스 위치를 지정할 경우에는, 복호가능한 그룹의 선두의 위치를 나타내준다. 게다가, 이용자의 조작에 응답하여 재생순서에서 직전의 그룹의 정보를 이용하여 복호가능한 그룹들중 한 그룹에 대해서 역세스 위치를 지정할 경우에는, 각 그룹에 속하는 독립적으로 복호가 가능한 최소한 한 개의 픽처의 위치를 나타내도록 마크 정보가 생성된다. 그러므로, 비디오 데이터에 대한 점프를 지정했을 때에, 점프 목적지에서의 표시 지연을 줄일 수 있다.

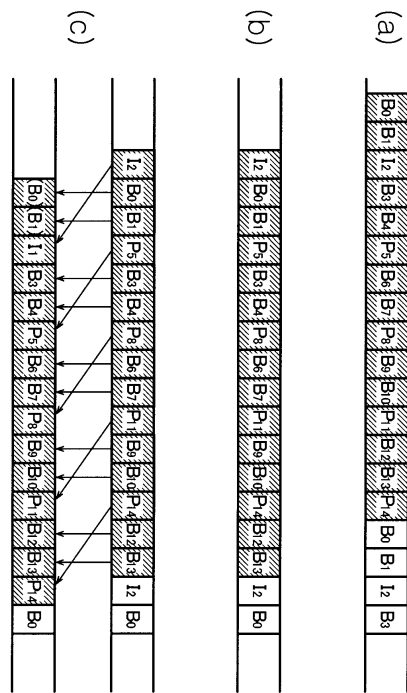
도면의 간단한 설명

- [0001] 도 1a, 도 1b, 도 1c는, 프레임간 압축을 이용하여 데이터의 디코딩 처리에 대해 설명하기 위한 개략도이다.
- [0002] 도 2는, 본 발명에 적용 가능한 AVCHD 포맷에 규정되는 데이터 모델을 개략적으로 나타내는 개략도이다.
- [0003] 도 3은, 인덱스 테이블을 설명하기 위한 개략도이다.
- [0004] 도 4는, 클립 AV스트림, 클립 정보, 클립, 플레이 아이템 및 플레이 리스트의 관계를 나타내는 UML 도면이다.
- [0005] 도 5는, 복수의 플레이 리스트로부터 동일한 클립을 참조하는 방법을 설명하기 위한 개략도이다.
- [0006] 도 6은, 기록 매체에 기록되는 파일의 관리 구조를 설명하기 위한 개략도이다.
- [0007] 도 7은, 파일 "index.bdmv"의 일례의 구조를 나타내는 선택스(syntax)를 나타내는 개략도이다.
- [0008] 도 8은, 블록 "blkIndexes()"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0009] 도 9는, 파일 "MovieObject.bdmv"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0010] 도 10은, 블록 "blkMovieObjects()"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0011] 도 11은, 플레이 리스트 파일 "xxxxx.mpls"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0012] 도 12는, 블록 "blkPlayList()"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0013] 도 13은, 블록 "blkPlayItem()"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0014] 도 14a와, 도 14b는 제 1 및 제 2의 심리스 접속(seamless connection)을 설명하기 위한 개략도이다.
- [0015] 도 15는, 블록 "blkPlayListMark()"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0016] 도 16은, 클립 인포메이션 파일의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.

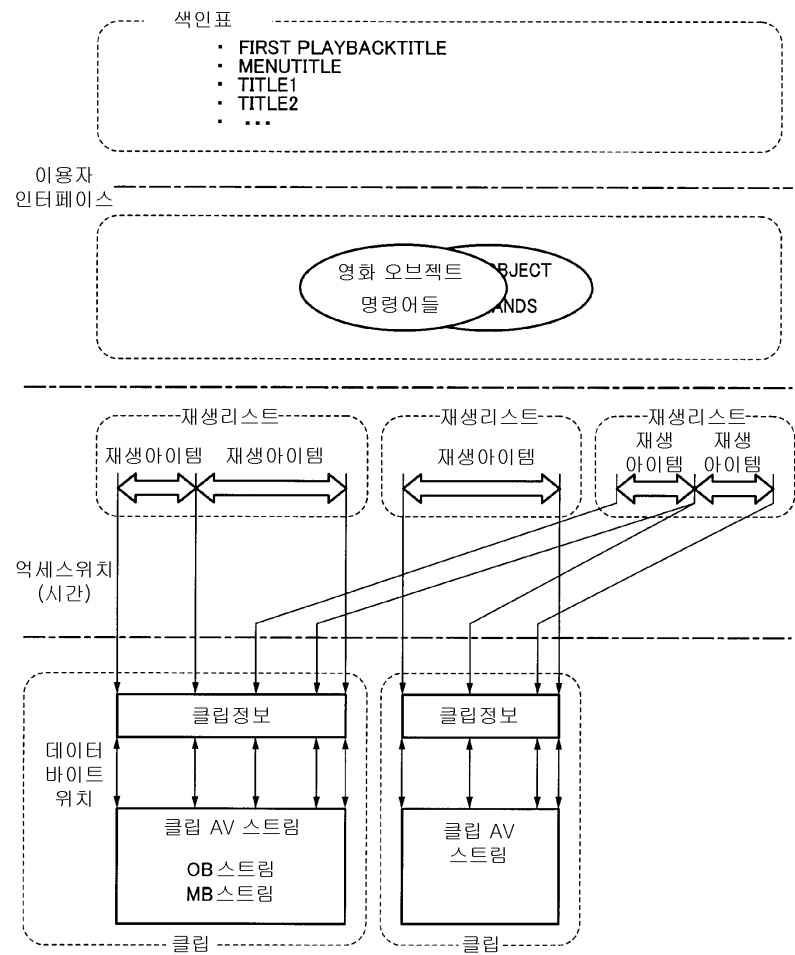
- [0017] 도 17은, 블록 "blkCPI()"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0018] 도 18은, 블록 "blkEPMMap()"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0019] 도 19는, 블록 "blkEPMMapForOneStreamPID(EP_stream_type, Nc, Nf)"의 일례의 구조를 나타내는 선택스를 나타내는 개략도이다.
- [0020] 도 20은, 엔트리 "PTSEPCoarse" 및 엔트리 "PTSEPFine"의 일례의 포맷을 나타내는 개략도이다.
- [0021] 도 21은, 엔트리 "SPNEPCoarse" 및 엔트리 "SPNEPFine"의 일례의 포맷을 나타내는 개략도이다.
- [0022] 도 22는, 플레이 리스트 및 이하의 일례의 파일 구조를 나타내는 개략도이다.
- [0023] 삭제
- [0024] 도 23a, 도 23b와, 도 23c는, 플레이 리스트를 이용한 클립 분할 및 편집을 개략적으로 설명하기 위한 개략도이다.
- [0025] 도 24는, 플레이 리스트 마크를 추가 삽입하는 처리를 설명하기 위한 플로차트이다.
- [0026] 도 25a와 도 25b는, 플레이 리스트 마크를 추가 삽입하는 처리를 설명하기 위한 개략도이다.
- [0027] 도 26은, 플레이 리스트 마크를 추가 삽입하는 처리를 설명하기 위한 개략도이다.
- [0028] 도 27a와 도 27b는, 플레이 리스트 마크를 엔트리 "PTSEPStart"로 표시된 각 위치에 할당하는 방법과, GOP 경계의 위치에 할당하는 방법에서의 챕터(chapter) 경계의 예를 각각 나타내는 개략도이다.
- [0029] 도 28은, 플레이 리스트 마크를 엔트리 "PTSEPStart"로 표시된 각 위치에 할당하는 방법과, GOP 경계의 위치에 할당하는 방법에서의 각각의 디코드 구간의 예를 나타내는 개략도이다.
- [0030] 도 29a, 도 29b와 도 29c는, GOP가 오픈 GOP 혹은 폐쇄 GOP로 판정될 수 있는 경우에, 플레이 리스트 마크를 엔트리 "PTSEPStart"로 표시된 각 위치에 할당하는 방법과 GOP 경계의 위치에 할당하는 방법에서의 각각의 디코드 구간의 예를 나타내는 개략도이다.
- [0031] 도 30은, 플레이 리스트 마크(PlaylistMarks)의 설정을 각 조건하에서 평가한 내용을 나타내는 테이블이다.
- [0032] 도 31은, 챕터 분할 편집을 실시할 때의, 플레이 리스트 마크의 일례의 삽입 방법을 나타내는 플로차트이다.
- [0033] 도 32는, 기록 개시에 따라 기록 개시 위치에 자동적으로 플레이 리스트를 할당하는 경우의 일례의 처리를 나타내는 플로차트이다.
- [0034] 도 33a와 도 33b는, 가상 플레이어에 의해 실행되는 동작을 개략적으로 나타내는 플로차트이다.
- [0035] 도 34는, 가상 플레이어에 의해 실행되는 동작을 개략적으로 나타내는 개략도이다.
- [0036] 도 35는, 본 발명의 실시의 한 형태에 적용 가능한 기록 재생장치의 일례의 구성을 개략적으로 나타내는 블록도이다.
- [0037] 도 36은, 플레이 리스트 마크에 근거하여 챕터 처리의 일례의 방법을 나타내는 플로차트이다.
- [0038] 도 37은, 본 발명의 실시의 한 형태의 다른 예에 의한 비디오 카메라 장치의 일례의 구성을 나타내는 개략도이다.

도면

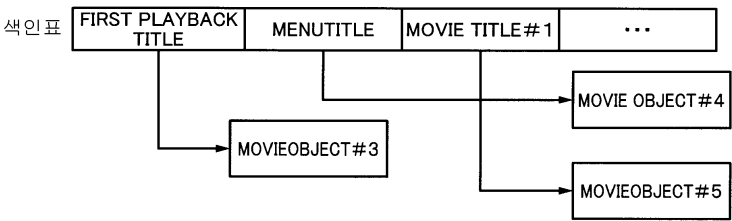
도면1



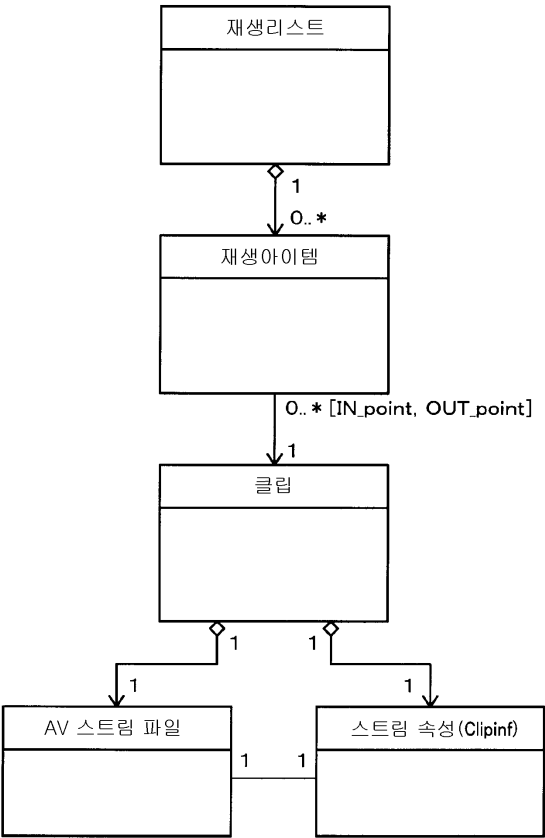
도면2



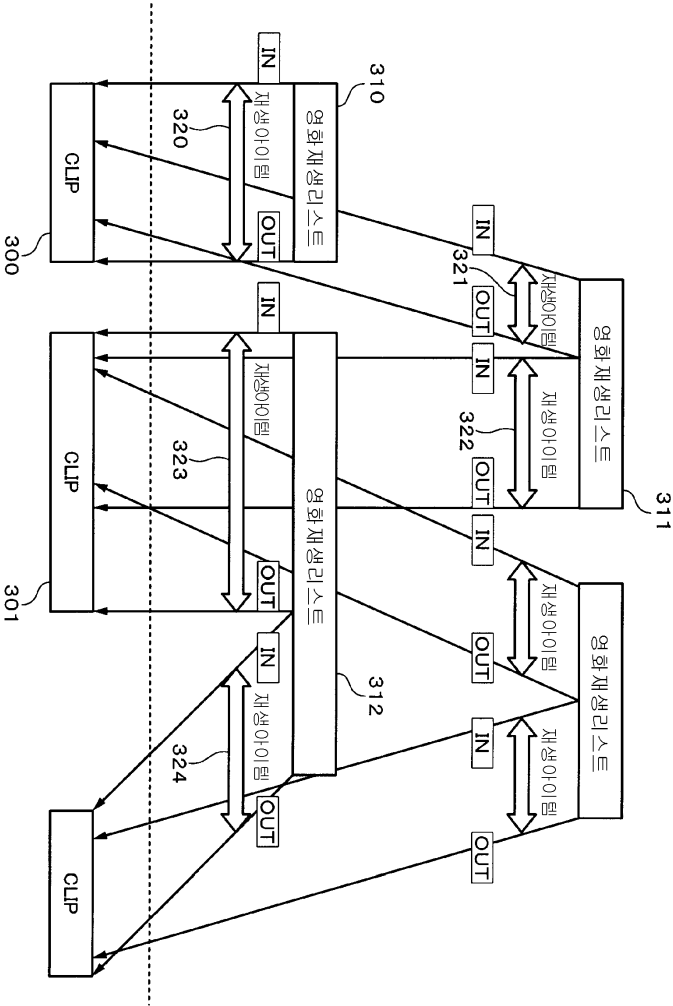
도면3



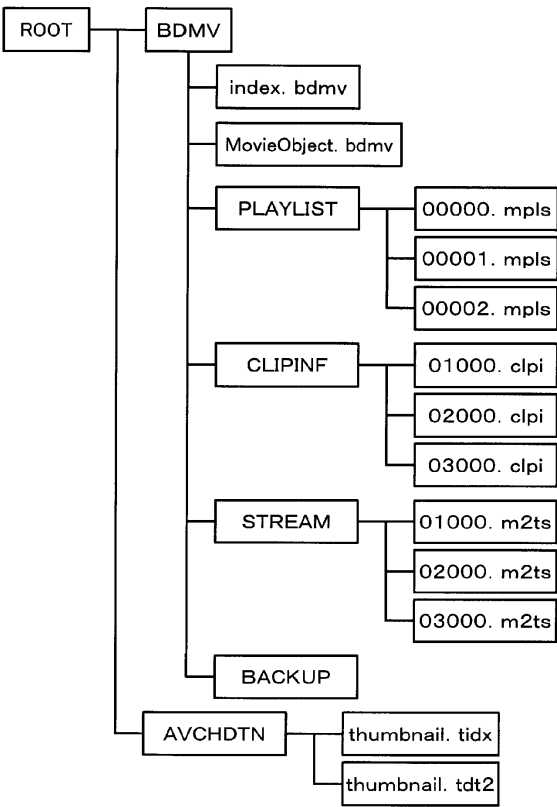
도면4



도면5



도면6



도면7

선택스	데이터길이 (비트)	비모닉
Index table file {		
TypeIndicator	8 * 4	bslbf
TypeIndicator2	8 * 4	bslbf
IndexesStartAddress	32	uimsbf
ExtensionDataStartAddress	32	uimsbf
reserved	192	bslbf
blkAppInfoBDMV()		
for (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
blkIndexes()		
for (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
blkExtensionData()		
for (i=0; i<N3; i++) {		
padding_word	16	bslbf
}		
}		

도면8

신택스	데이터길이 (비트)	니모닉
blkIndexes() {		
Length	32	uimsbf
FirstPlaybackTitle() {		
reserved	1	bslbf
'1'	1	bslbf
reserved	31	bslbf
'1'	1	bslbf
reserved	14	bslbf
FirstPlaybackTitleMobIDRef	16	uimsbf
reserved	32	bslbf
}		
MenuTitle() {		
reserved	1	bslbf
'1'	1	bslbf
reserved	31	bslbf
'1'	1	bslbf
reserved	14	bslbf
MenuTitleMobIDRef	16	uimsbf
reserved	32	bslbf
}		
NumberOfTitles	16	uimsbf
for (title_id=0; title_id < NumberOfTitles; title_id++) {		
MovieTitle[title_id]() {		
reserved	1	bslbf
'1'	1	bslbf
reserved	46	bslbf
MovieTitleMobIDRef[title_id]	16	uimsbf
reserved	32	bslbf
}		
}		
}		

도면9

신택스	데이터길이 (비트)	니모닉
MovieObject file {		
TypeIndicator	8 * 4	bslbf
TypeIndicator2	8 * 4	bslbf
ExtensionDataStartAddress	32	uimsbf
reserved	224	bslbf
blkMovieObjects()		
for (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
blkExtensionData()		
for (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
}		

도면10

신택스	데이터길이 (비트)	비고
bklMovieObjects() {		
Length	32	uintsbf
reserved	32	bslbf
NumberOfMobjs	16	uintsbf
for (mobl_id=0; mobl_id<NumberOfMobjs; mobl_id++) {		
MovieObject[mobl_id]() {		
TerminalInfo() {		
'1'	1	bslbf
reserved	15	bslbf
}		
NumberOfNavigationCommands[mobl_id]	16	uintsbf
for (command_id=0;		
command_id<NumberOfNavigationCommands[mobl_id];		
command_id++) {		
NavigationCommand[mobl_id][command_id]	96	bslbf
}		
}		
}		
}		

도면11

신택스	데이터길이 (비트)	니모닉
Movie PlayList file {		
TypeIndicator	8 * 4	bslbf
TypeIndicator2	8 * 4	bslbf
PlayListStartAddress	32	uimsbf
PlayListMarkStartAddress	32	uimsbf
ExtensionDataStartAddress	32	uimsbf
reserved	160	bslbf
blkAppInfoPlayList()		
for (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
blkPlayList()		
for (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
blkPlayListMark()		
for (i=0; i<N3; i++) {		
padding_word	16	bslbf
}		
blkExtensionData()		
for (i=0; i<N4; i++) {		
padding_word	16	bslbf
}		
}		

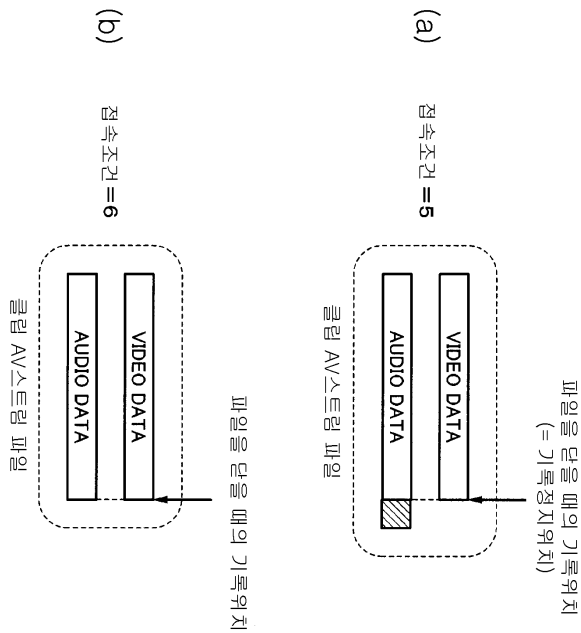
도면12

신택스	데이터길이 (비트)	니모닉
blkPlayList() {		
Length	32	uimsbf
reserved	16	bslbf
NumberOfPlayItems	16	uimsbf
NumberOfSubPaths	16	uimsbf
for (PlayItem_id=0; PlayItem_id<NumberOfPlayItems; PlayItem_id++) {		
blkPlayItem()		
}		
for (SubPath_id=0; SubPath_id<NumberOfSubPaths; SubPath_id++) {		
blkSubPath()		
}		
}		

도면13

신택스	데이터길이 (바이트)	니모닉
blkPlayItem() {		
Length	16	uimsbf
ClipInformationFileName	8 * 5	bslbf
ClipCodecIdentifier	8 * 4	bslbf
reserved	12	bslbf
ConnectionCondition	4	bslbf
RefToSTCID	8	uimsbf
INTime	32	uimsbf
OUTTime	32	uimsbf
blkUOMaskTable()		
PlayItemRandomAccessFlag	1	bslbf
reserved	7	bslbf
StillMode	8	bslbf
if (StillMode== 0x01){		
StillTime	16	uimsbf
} else {		
reserved	16	bslbf
}		
blkSTNTTable()		
}		

도면14



도면15

신택스	데이터길이 (비트)	니모닉
blkPlayListMark() {		
Length	32	uimsbf
NumberOfPlayListMarks	16	uimsbf
for(PL_mark_id=0; PL_mark_id< NumberOfPlayListMarks; PL_mark_id++) {		
reserved	8	bslbf
MarkType	8	bslbf
RefToPlayItemID	16	uimsbf
MarkTimeStamp	32	uimsbf
EntryESPID	16	uimsbf
Duration	32	uimsbf
}		
}		

도면16

신택스	데이터길이 (비트)	니모닉
Clip information file {		
TypeIndicator	8 * 4	bslbf
TypeIndicator2	8 * 4	bslbf
SequenceInfoStartAddress	32	uimsbf
ProgramInfoStartAddress	32	uimsbf
CPIStartAddress	32	uimsbf
ClipMarkStartAddress	32	uimsbf
ExtensionDataStartAddress	32	uimsbf
reserved	96	bslbf
blkClipInfo()		
for (i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
blkSequenceInfo()		
for (i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
blkProgramInfo()		
for (i=0; i<N3; i++) {		
padding_word	16	bslbf
}		
blkCPI()		
for (i=0; i<N4; i++) {		
padding_word	16	bslbf
}		
blkClipMark()		
for (i=0; i<N5; i++) {		
padding_word	16	bslbf
}		
blkExtensionData()		
for (i=0; i<N6; i++) {		
padding_word	16	bslbf
}		
}		

도면17

신택스	데이터길이 (비트)	니모닉
blkCPI() {		
Length	32	uimbsf
if(Length !=0) {		
reserved	12	bslbf
CPIType	4	bslbf
blkEMap()		
}		
}		

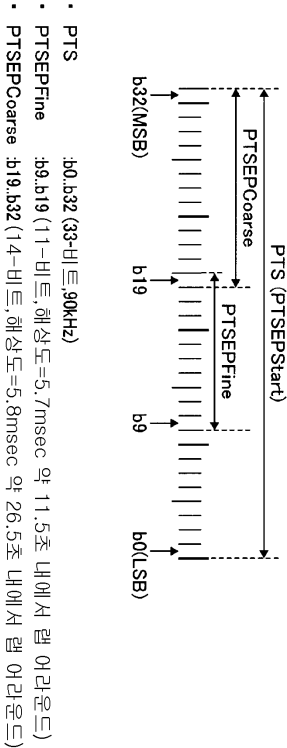
도면18

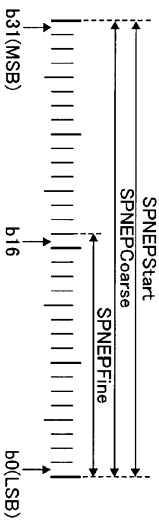
신택스	데이터길이 (비트)	니모닉
blkEMap(){		
reserved	8	bslbf
NumberOfStreamPIDEntries	8	uimbsf
for (k=0; k<NumberOfStreamPIDEntries; k++) {		
StreamPID[k]	16	bslbf
reserved	10	bslbf
EPStreamType[k]	4	bslbf
NumberOfEPCoarseEntries[k]	16	uimbsf
NumberOfEPFineEntries[k]	18	uimbsf
EMapForOneStreamPIDStartAddress[k]	32	uimbsf
}		
for (i=0; i<X; i++) {		
padding_word	16	bslbf
}		
for (k=0; k<NumberOfStreamPIDEntries; k++) {		
blkEMapForOneStreamPID(EPStreamType[k], NumberOfEPCoarseEntries[k], NumberOfEPFineEntries[k])		
for (i=0; i<Y[k]; i++) {		
padding_word	16	bslbf
}		
}		
}		

도면19

신택스	데이터길이 (비트)	니모닉
blkEPMaPForOneStreamPID(<i>EP_stream_type</i> , <i>Nc</i> , <i>Nf</i>) {		
EPFineTableStartAddress	32	uimbsf
for (i=0; i< <i>Nc</i> ; i++) {		
// EP coarse table		
RefToEPFineID[i]	18	uimbsf
PTSEPCoarse[i]	14	uimbsf
SPNEPCoarse[i]	32	uimbsf
}		
for (i=0; i< <i>X</i> ; i++) {		
padding_word	16	bslbf
}		
for (<i>EP_fine_id</i> =0; <i>EP_fine_id</i> < <i>Nf</i> , <i>EP_fine_id</i> ++) {		
// EP fine table		
ReservedEPFine[EP_fine_id]	1	bslbf
IEndPositionOffset[EP_fine_id]	3	bslbf
PTSEPFine[EP_fine_id]	11	uimbsf
SPNEPFine[EP_fine_id]	17	uimbsf
}		
}		

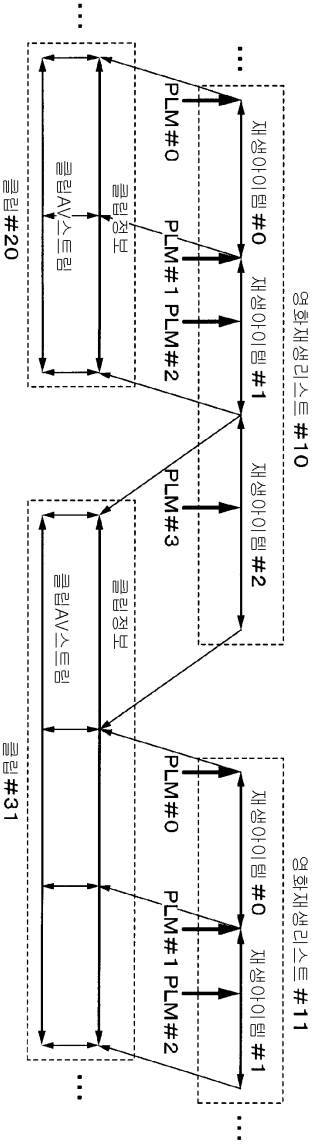
도면20



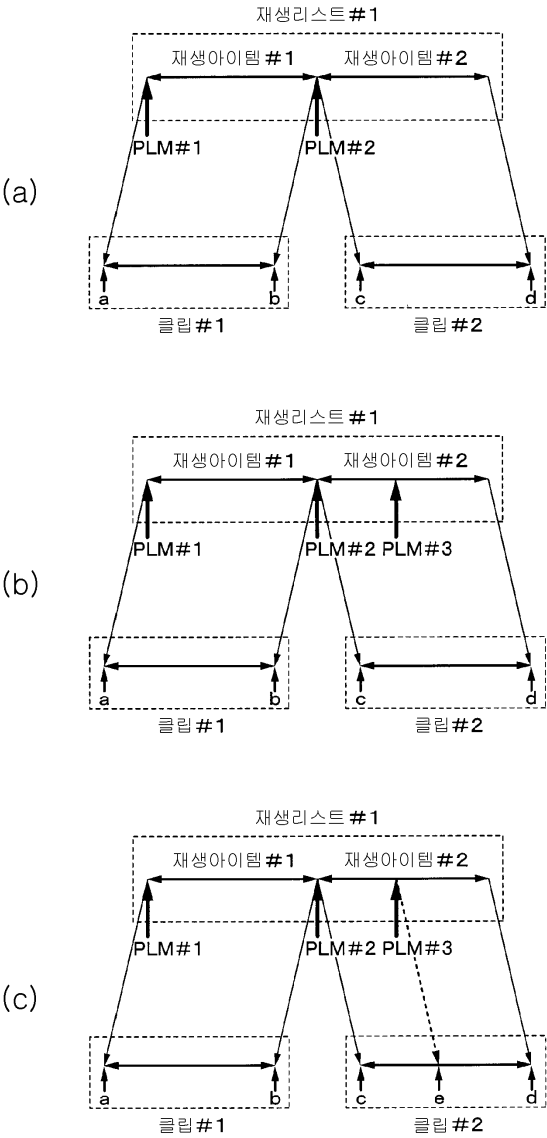


- SPNEPStart b0..b31 (32-비트)
- SPNEPFine b0..b16 (17-비트, AV스트림 파일내에서 약 25Mbyte내에서 랩 아라운드(wrap around))
- SPNEPCoarse b0..b31 (32-비트)

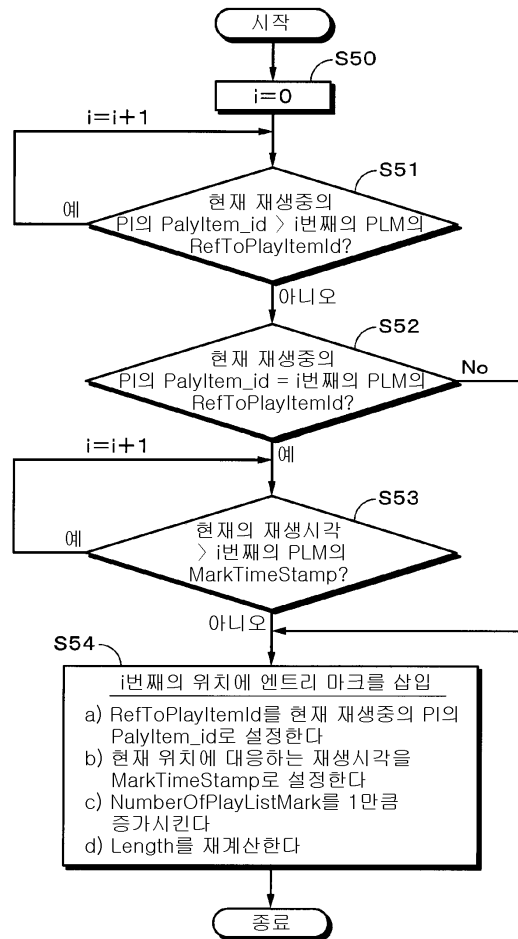
도면22



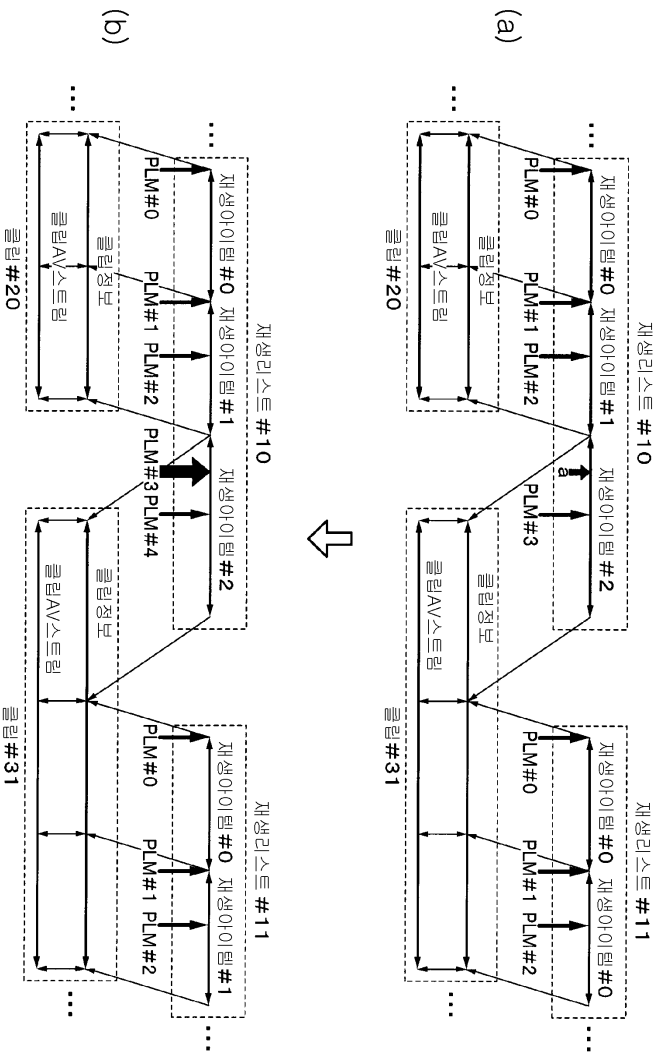
도면23



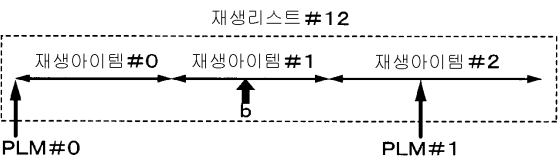
도면24



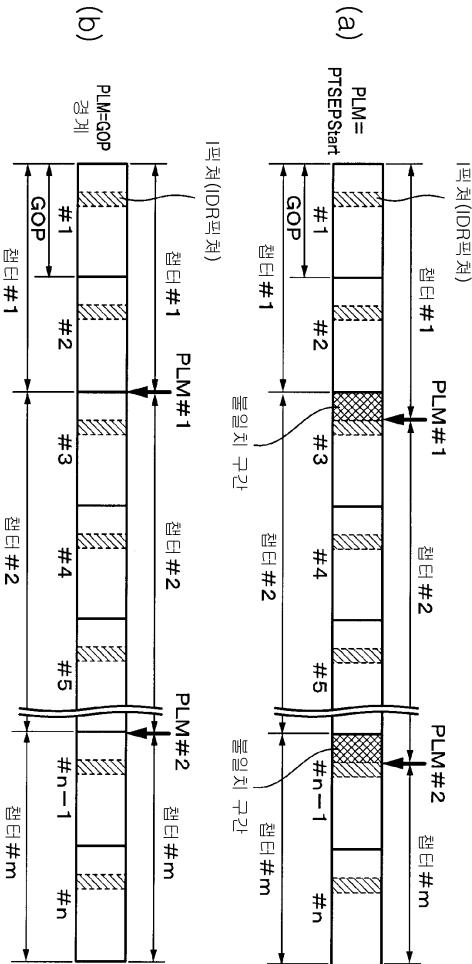
도면25



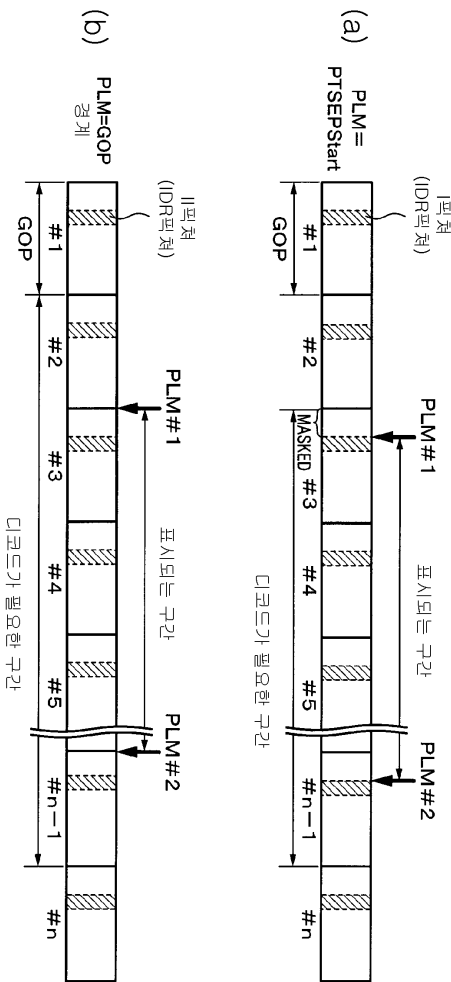
도면26

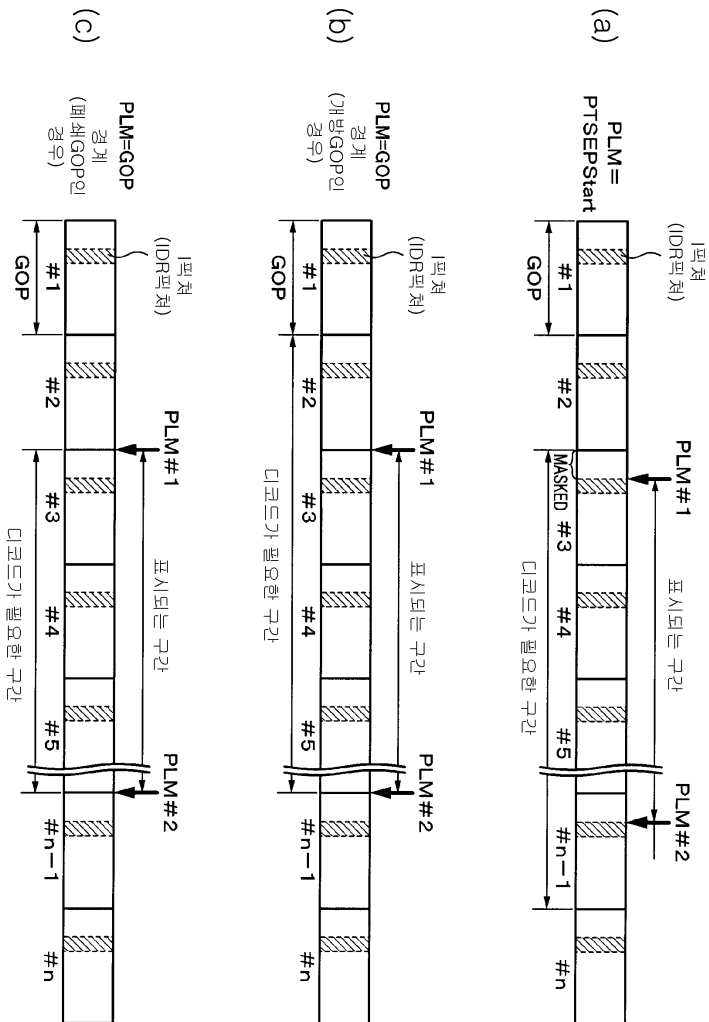


도면27



도면28

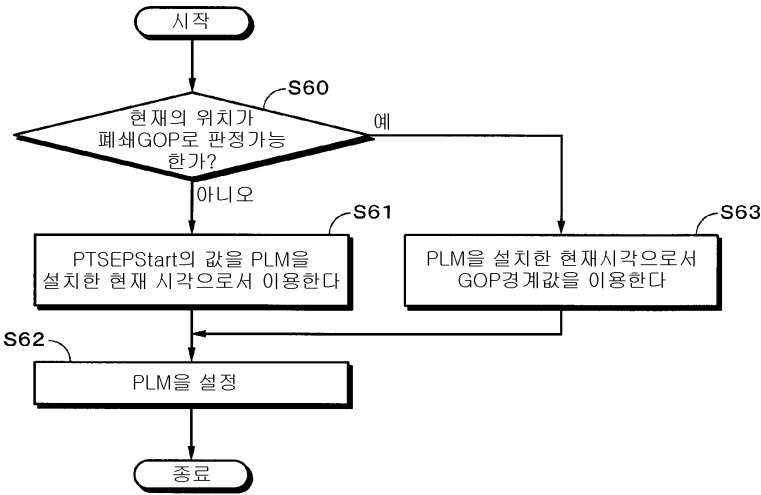




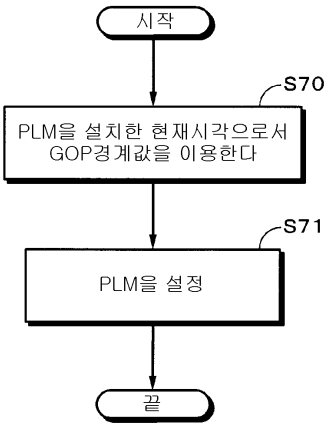
도면30

실제의 위치 지정위치	항목	PTSEPEnd	GOP경계 (일반재생기)	GOP경계 (독자정보 이용기)
지정 위치가 기록 시작/ 종지의 경계가 되는 경우	기록과 재생에 있어서 헛터경계의 차이가 있는가?	x	o	o
	개시위치에서 판독/디코딩을 위해 표시 프레임보다 더 많은 프레임이 필요한 구간	o (M-1)개의 프레임이 필요	단행진 GOP 라면 불필요	단행진 GOP 라면 불필요
	중문위치에서 판독/디코딩을 위해 표시 프레임보다 더 많은 프레임이 필요한 구간	x (N-M-1)개의 프레임이 필요	불필요	불필요
지정 위치가 기록 시작/ 종지의 경계가 아닌 경우	기록과 재생에 있어서 헛터경계의 차이가 있는가?	-	-	-
	개시위치에서 판독/디코딩을 위해 표시 프레임보다 더 많은 프레임이 필요한 구간	o (M-1)개의 프레임이 필요	x N개의 프레임 이 필요	단행진 GOP 라면 N개의 프레임이 필요
	중문위치에서 판독/디코딩을 위해 표시 프레임보다 더 많은 프레임이 필요한 구간	x (N-M-1)개의 프레임이 필요	x N개의 프레임 이 필요	불필요

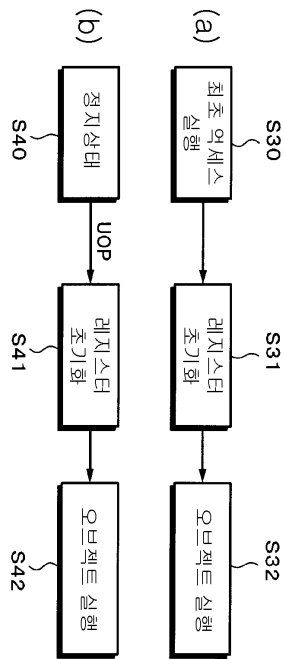
도면31



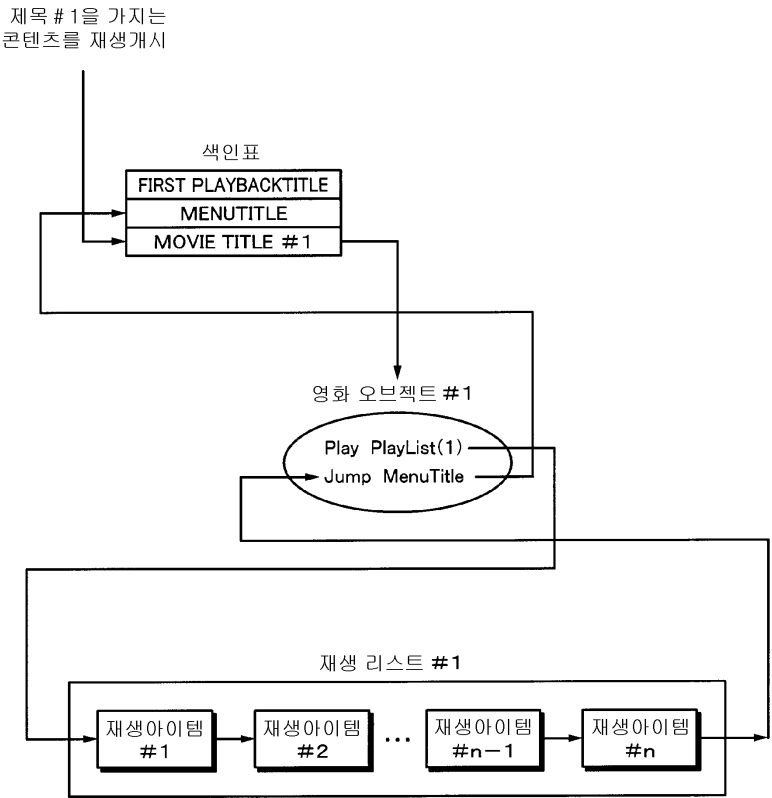
도면32



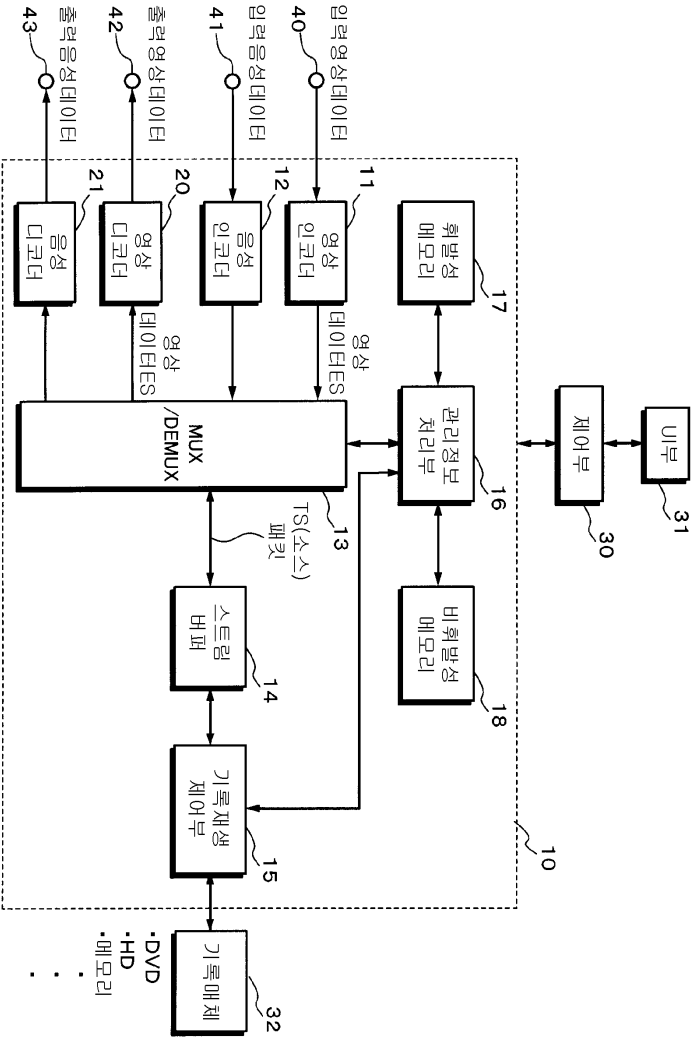
도면33



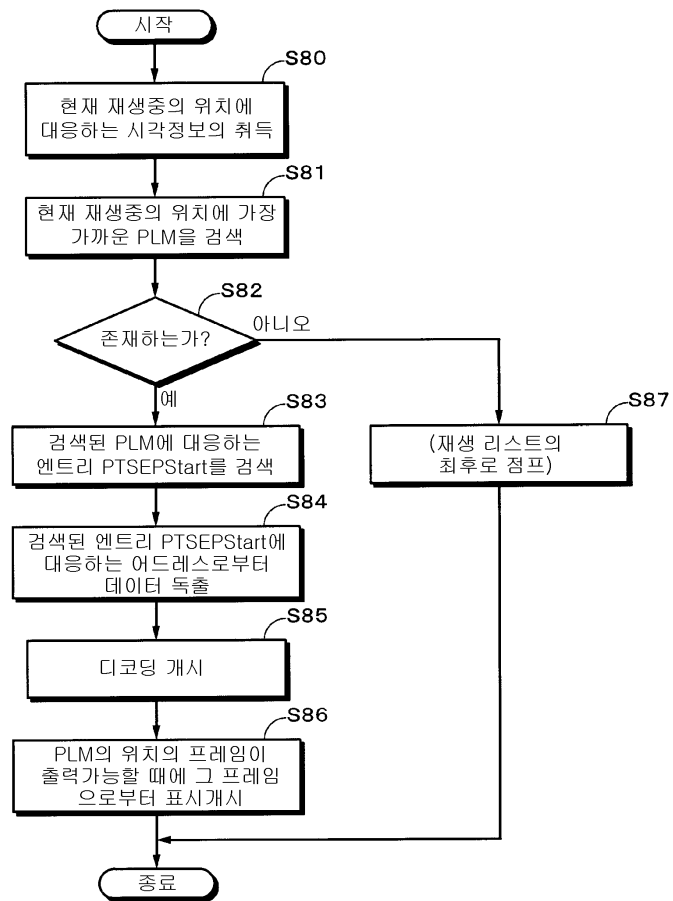
도면34



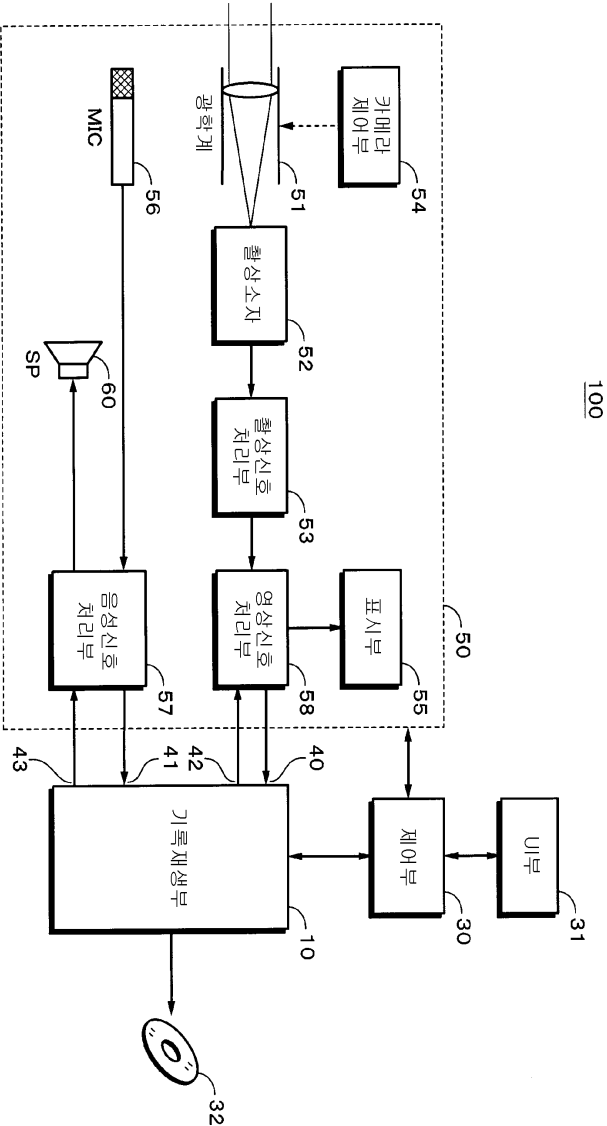
도면35



도면36



도면37



【심사관 직권보정사항】

【직권보정 1】

【보정항목】 청구범위

【보정세부항목】 제8항

【변경전】

상기 기록매체에 기록하는

【변경후】

기록매체에 기록하는