



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2014년02월21일

(11) 등록번호 10-1365723

(24) 등록일자 2014년02월14일

(51) 국제특허분류(Int. Cl.)

G11B 20/10 (2006.01) *H04N 5/937* (2006.01)*H04N 5/92* (2006.01) *G11B 20/12* (2006.01)

(21) 출원번호 10-2006-0066330

(22) 출원일자 2006년07월14일

심사청구일자 2011년07월06일

(65) 공개번호 10-2007-0009467

(43) 공개일자 2007년01월18일

(30) 우선권주장

JP-P-2005-00206997 2005년07월15일 일본(JP)

JP-P-2006-00147981 2006년05월29일 일본(JP)

(56) 선행기술조사문헌

US06714826 B1*

US20040179818 A1*

JP2005538440

JP2003244641 A

*는 심사관에 의하여 인용된 문헌

(73) 특허권자

소니 주식회사

일본국 도쿄도 미나토구 코난 1-7-1

(72) 발명자

하또리 시노부

일본국 도쿄도 시나가와구 키타시나가와 6쵸메 7
반 35고 소니가부시끼 가이사내

가토 모토키

일본국 도쿄도 시나가와구 키타시나가와 6쵸메 7
반 35고 소니가부시끼 가이사내

(74) 대리인

신판호

전체 청구항 수 : 총 6 항

심사관 : 장진환

(54) 발명의 명칭 재생장치 및 재생방법, 프로그램, 프로그램 저장매체, 데이터구조, 기록매체, 기록장치 및 기록매체의 제조방법

(57) 요 약

재생되는 스트림 데이터에 오디오 스트림 #1에 합성되는 다른 음성 데이터가 스트림에 포함되어 있는지 아닌지를 판단한다.

PlayList 파일에 포함되는 PlayList()에는, PlayList()에 의해 참조되는 데이터에 오디오 스트림 #1에 합성되는 사운드 데이터가 포함되어 있는지 아닌지의 플래그와, 오디오 스트림 #1에 합성되는 오디오 스트림 #2가 포함되어 있는지 아닌지의 플래그가 기재되어 있다. 재생장치의 컨트롤러는, PlayList에 기재되어 있는 플래그를 참조하여, 자기 자신이 재생 가능한 음성데이터 데이터이며 오디오 스트림 #1에 합성되는 다른 음성 데이터가 스트림에 포함되어 있는지 아닌지를 판단할 수 있다. 본 발명은, 재생장치에 적용할 수 있다.

대 표 도

PlayList-Syntax		
Syntax	No. of bits	Mnemonic
PlayList {		
length	32	uimsbf
is_MixApp_1	1	uimsbf
is_MixApp_2	1	uimsbf
reserved_for_future_use	14	bslbf
number_of_PlayItems	16	uimsbf
number_of_SubPaths	16	uimsbf
for (PlayItem_id=0:		
PlayItem_id<number_of_PlayItems:		
PlayItem_id++) {		
PlayItem()		
}		
for (SubPath_id= 0:		
SubPath_id<number_of_SubPaths:		
SubPath_id++) {		
SubPath()		
}		
}		

특허청구의 범위

청구항 1

부호화(encode)된 스트림 데이터를 포함하는 재생데이터를 취득하는 재생데이터 취득수단과,

상기 스트림 데이터를 디코드하는 디코드수단과,

상기 디코드수단에 의해 디코드된 상기 스트림 데이터에, 상기 스트림 데이터와는 다른 데이터이며, 상기 스트림 데이터에 합성하기 위한 데이터인 합성용 데이터를 합성하는 합성수단과,

상기 스트림 데이터를 상기 디코드수단에 공급하는지, 출력 단자로 출력하는지를 선택하는 선택수단과,

상기 선택수단을 제어하는 제어수단을 갖추어 구성되고,

상기 제어수단은,

상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에 포함되는 재생제어정보로부터, 상기 합성수단에 의한 상기 합성용 데이터의 합성이 이루어지는지 아닌지를 나타내는 판별정보를 취득하고,

상기 판별정보에 근거하여 상기 재생데이터에 상기 합성용 데이터가 포함되지 않았다고 판별되고, 또, 상기 재생데이터 취득수단에 의해 취득된 데이터가 부호화 데이터로서 출력되는 경우, 상기 스트림 데이터를 출력 단자로 출력하도록 상기 선택수단을 제어하는 것을 특징으로 하는 재생장치.

청구항 2

제 1항에 있어서,

상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에는, 상기 재생데이터의 타이틀에 대응한 데이터를 포함하는 소정의 파일이 1개 포함되어 있고,

상기 제어수단은, 상기 판별정보를 상기 소정의 파일로부터 취득하도록 구성된 것을 특징으로 하는 재생장치.

청구항 3

제 1항에 있어서,

상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에는, 상기 재생데이터의 재생순서를 나타내는 정보를 포함하는 소정의 파일이 1개 또는 복수 포함되어 있고,

상기 제어수단은, 상기 판별정보를 상기 소정의 파일로부터 취득하도록 구성된 것을 특징으로 하는 재생장치.

청구항 4

제 1항에 있어서,

상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에는, 상기 재생데이터의 재생순서를 나타내는 정보이며, 상기 재생데이터에 1개 또는 복수 포함되는 제 1 데이터와, 상기 제 1 데이터에 의해 재생순서가 제어되는 데이터의 재생 구간을 나타내는 정보이며, 상기 제 1 데이터에 1개 이상 연결되는(associated) 제 2 데이터가 포함되어 있고,

상기 제어수단은, 상기 판별정보를 상기 제 2 데이터로부터 취득하도록 구성된 것을 특징으로 하는 재생장치.

청구항 5

데이터를 재생출력하는 재생장치의 재생방법에 있어서,

부호화된 스트림 데이터를 포함하는 재생데이터에 있어서, 상기 재생데이터에 포함되는 재생제어정보로부터, 상기 스트림 데이터에 합성하기 위한 합성용 데이터의 합성이 이루어지는지 아닌지를 나타내는 판별정보를 취득하는 단계와,

취득된 상기 판별정보를 근거로, 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함

되어 있는지 아닌지를 판단하는 단계와,

상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되어 있지 않다고 판단되고, 또, 상기 재생장치로부터 출력되는 상기 데이터가 부호화된 데이터인 경우, 상기 스트림 데이터를 출력 단자로 출력하는 단계를 포함하여 구성된 것을 특징으로 하는 재생방법.

청구항 6

데이터를 재생출력하는 처리를 컴퓨터에 실행시키기 위한 프로그램 저장매체에 있어서,

부호화된 스트림 데이터를 포함하는 재생데이터에 있어서, 상기 재생데이터에 포함되는 재생제어정보로부터, 상기 스트림 데이터에 합성하기 위한 합성용 데이터의 합성이 이루어지는지 아닌지를 나타내는 판별정보를 취득하는 단계와,

취득된 상기 판별정보를 근거로, 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되어 있는지 아닌지를 판단하는 단계와,

상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되지 않다고 판단되고, 또, 재생장치로부터 출력되는 상기 데이터가 부호화된 데이터인 경우, 상기 스트림 데이터를 출력 단자로 출력하는 단계를 포함하는 처리를 컴퓨터에 실행시키도록 구성된 것을 특징으로 하는 프로그램 저장매체.

청구항 7

삭제

청구항 8

삭제

청구항 9

삭제

청구항 10

삭제

청구항 11

삭제

명세서

발명의 상세한 설명

발명의 목적

발명이 속하는 기술 및 그 분야의 종래기술

[0049] 본 발명은 재생장치 및 재생방법, 프로그램, 프로그램 저장매체, 데이터구조, 기록매체, 기록장치 및 기록매체의 제조방법에 관한 것이며, 특히, 재생되는 음성 데이터의 합성이 가능하도록 되어 있는 경우에 이용하기 적합한 재생장치 및 재생방법, 프로그램, 프로그램 저장매체, 데이터구조, 기록매체, 기록장치 및 기록매체의 제조방법에 관한 것이다.

[0050] 다른 컨텐츠를 동시에 시청하기 위해, 복수의 컨텐츠의 영상 데이터를 디코드한 후, 비압축의 화상 데이터를 합성하고, 그 후, D/A 변환을 실행해, 영상 출력 단자에 출력하고, 외부의 표시장치에 표시시키는 기술이 있다 (예를 들면, 특허문현 1).

[0051] [특허문현 1] 특개 2005-20242호 공보

상술한 특허문현 1과 같이, 비디오 데이터를 합성하기 위해서는, 압축 부호화된 비디오 데이터를 각각 복호하여, 비압축의 비디오 데이터를 합성할 필요가 있다. 이는, 비디오 데이터뿐만 아니라, 오디오 데이터

에 있어서도 마찬가지이며, 오디오 데이터를 합성하는 경우에는, 비압축의 오디오 데이터를 이용하지 않으면 안 된다.

[0053] 또, 출력을 받는 기기가 가지는 기능, 또는, 출력을 받는 기기와의 접속 방법에 따라서는, 출력 데이터는, 부호화된 데이터가 아니면 안 되는 경우가 있다.

[0054] 주로 출력되는 음성 데이터에 대하여, 다른 음성 데이터가 합성되고, 출력되는 음성 데이터는, 부호화 데이터인 경우의 처리에 대하여, 도 1을 이용하여 설명한다.

[0055] 제 1 음성 데이터 취득부(11)는, 예를 들면, 광디스크로부터 독출되어 공급된 압축 부호화된 제 1 음성 데이터를 취득하고, 복호부(12)에 공급한다. 제 1 음성 데이터는, 그 컨텐츠에 대하여, 상시 재생출력되는 음성 데이터인 것으로 한다. 복호부(12)는, 제 1 음성 데이터 취득부(11)로부터 공급된, 압축 부호화된 제 1 음성 데이터를 복호하고, 비압축의 제 1 음성 데이터를 합성부(14)에 공급한다.

[0056] 제 2 음성 데이터 취득부(13)는, 비압축의 제 2 음성 데이터를 취득하고, 합성부(14)에 공급한다. 합성부(14)는, 제 2 음성 데이터 취득부(13)로부터 비압축의 제 2 음성 데이터의 공급을 받고, 제 2 음성 데이터를 복호부(12)로부터 공급된 비압축의 제 1 음성 데이터와 합성하여, 합성된 음성 데이터를 부호화부(15)에 공급한다.

[0057] 부호화부(15)는, 공급된 음성 데이터를 부호화하고, 디지털 인터페이스(16)에 공급한다. 디지털 인터페이스는, 공급된 부호화데이터를 소정의 네트워크를 통하여 다른 장치 등에 송출한다.

발명이 이루고자 하는 기술적 과제

[0058] 이 비압축의 제 2 음성 데이터는, 그 컨텐츠에 대하여 준비되어 있는 경우와 준비되어 있지 않은 경우가 있다. 즉, 컨텐츠에 따라서는, 제 2 음성 데이터 취득부(13)가 비압축의 제 2 음성 데이터를 취득하지 않도록 되어 있는 경우나, 컨텐츠의 소정의 부분만 제 2 음성 데이터가 준비되고, 제 2 음성 데이터 취득부(13)가 소정의 부분만의 비압축의 제 2 음성 데이터를 취득하도록 되어 있는 경우 등이 있다. 합성부(14)는, 제 2 음성 데이터 취득부(13)로부터 비압축의 제 2 음성 데이터의 공급을 받지 않은 경우, 복호부(12)로부터 공급된 비압축의 제 1 음성 데이터를 부호화부(15)에 공급한다.

[0059] 도 1을 이용하여 설명한 장치에 있어서, 주로 재생출력되는 제 1 음성 데이터에 대하여, 제 2 음성 데이터가 합성되는 컨텐츠와 합성되지 않는 컨텐츠의 양쪽을 재생출력 가능하도록 되어 있는 경우, 또는, 동일 컨텐츠의 소정의 부분만 제 2 음성 데이터를 합성하는 것이 가능하도록 되어 있는 경우, 종래에는, 재생출력하는 장치에 있어서, 제 2 음성 데이터가 합성되는지 아닌지(제 2 음성 데이터가 존재하는지 아닌지)를 검출할 수 없었다.

[0060] 따라서, 제 1 음성 데이터는, 제 2 음성 데이터가 합성되지 않는 컨텐츠, 또는, 제 2 음성 데이터가 합성되지 않는 부분에 있어서도, 제 2 음성 데이터가 합성되는 경우와 마찬가지로 복호되고, 재부호화 되도록 되어 있었다. 그 때문에, 제 1 음성 데이터는, 항상 복호 및 재부호화를 위해 음질이 열화해 버린다.

[0061] 또, 예를 들면, 상술한 특허문헌 1과 같이, 비디오 데이터를 합성하는 경우에 있어서, 부호화 데이터가 출력되도록 되어 있는 경우에도 같은 경우를 생각할 수 있다. 즉, 영상 데이터의 합성을 행하는 것이 가능한 재생 장치가 주로 출력되는 영상 데이터에 합성하는 다른 데이터의 유무를 검출할 수 없으면, 주로 출력되는 영상 데이터는 항상 복호되고, 재부호화 되어 버리기 때문에, 화질이 열화해 버린다.

[0062] 본 발명은 이러한 상황에 감안하여 이루어진 것으로, 필요에 따라 데이터의 합성을 실현하는 것과 동시에, 데이터의 열화를 가능한 한 막을 수 있도록 하는 것이다.

발명의 구성 및 작용

[0063] 본 발명의 제 1 측면의 재생장치는, 부호화된 스트림 데이터를 포함하는 재생데이터를 취득하는 재생데이터 취득수단과, 상기 스트림 데이터를 디코드하는 디코드 수단과, 상기 디코드 수단에 의해 디코드된 상기 스트림 데이터에, 상기 스트림 데이터와는 다른 데이터이며 상기 스트림 데이터에 합성하기 위한 데이터인 합성용 데이터를 합성하는 합성수단과, 상기 스트림 데이터를 상기 디코드 수단에 공급하는지, 출력하는지를 선택하는 선택수단과, 상기 선택수단을 제어하는 제어수단을 갖추고, 상기 제어수단은, 상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터로부터, 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되어 있는지 아닌지에 근거하는 판별정보를 취득하고, 상기 판별정보에 근거하여 상기 재생데이터에 상기 합성용 데이터가 포함되지 않았다고 판별되고, 또, 상기 재생데이터 처리수단에 의해 처리된 데이터가 부호화데이터로

서 출력되는 경우, 상기 스트림 데이터를 출력하도록 상기 선택수단을 제어한다.

[0064] 상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에는, 상기 재생데이터의 타이틀에 대응한 데이터를 포함하는 소정의 파일이 1 파일 포함되는 것으로 할 수 있고, 상기 제어수단에는, 상기 판별정보를 상기 소정의 파일로부터 취득시키도록 할 수 있다.

[0065] 상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에는, 상기 재생데이터의 재생순서를 나타내는 정보를 포함하는 소정의 파일이 1개 또는 복수 포함되는 것으로 할 수 있고, 상기 제어수단에는, 상기 판별정보를 상기 소정의 파일로부터 취득시키도록 할 수 있다.

[0066] 상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에는, 상기 재생데이터의 재생순서를 나타내는 정보이며 상기 재생데이터에 1개 또는 복수 포함되는 제 1 데이터와, 상기 제 1 데이터에 의해서 재생순서가 제어되는 데이터의 재생 구간을 나타내는 정보이며 상기 제 1 데이터에 1개 또는 복수 관련지어지는 제 2 데이터가 포함되는 것으로 할 수 있고, 상기 제어수단에는, 상기 판별정보를 상기 제 2의 데이터로부터 취득시키도록 할 수 있다.

[0067] 본 발명의 제 1 측면의 재생방법, 프로그램 및 프로그램 저장매체에 저장되어 있는 프로그램은, 부호화된 스트림 데이터를 포함하는 재생데이터로부터 상기 재생데이터에 상기 스트림 데이터에 합성되는 합성용 데이터가 포함되어 있는지 아닌지를 나타내는 판별정보를 취득하는 단계와, 취득된 상기 판별정보를 근거로, 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되어 있는지 아닌지를 판단하는 단계와, 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되지 않다고 판단되고, 또, 상기 재생장치로부터 출력되는 상기 데이터가 부호화된 데이터인 경우, 상기 스트림 데이터를 출력하는 단계를 포함한다.

[0068] 본 발명의 제 2 측면의 데이터구조 또는 기록매체에 기록되어 있는 데이터는, 스트림 데이터의 재생순서를 관리하는 제 1 정보를 포함하고, 상기 제 1 정보는, 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터의 유무에 근거하는 제 2 정보를 포함한다.

[0069] 본 발명의 제 2 측면의 기록장치는, 재생장치에 대하여 재생가능한 데이터를 기록매체에 기록하는 기록장치에 있어서, 스트림 데이터의 재생순서를 관리하는 제 1 정보를 포함하고, 상기 제 1 정보는, 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터의 유무에 근거하는 제 2 정보를 포함하는 데이터구조를 가지는 데이터를 취득하는 취득수단과, 상기 취득수단에 의해 취득된 상기 데이터를 기록매체에 기록하는 기록수단을 갖춘다.

[0070] 본 발명의 제 2 측면의 기록매체의 제조방법은, 재생장치에 대하여 재생가능한 데이터가 기록되는 기록매체의 제조방법에 있어서, 스트림 데이터의 재생순서를 관리하는 제 1 정보를 포함하고, 상기 제 1 정보는, 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터의 유무에 근거하는 제 2 정보를 포함하는 데이터를 생성하는 단계와, 생성된 데이터를 기록매체에 기록하는 단계를 포함한다.

[0071] 본 발명의 제 1 측면에 있어서는, 부호화된 스트림 데이터를 포함하는 재생데이터로부터, 상기 재생데이터에 상기 스트림 데이터에 합성되는 합성용 데이터가 포함되어 있는지 아닌지에 근거하는 판별정보가 취득되고, 취득된 상기 판별정보를 근거로, 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되어 있는지 아닌지가 판단되어, 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되지 않다고 판단되고, 또, 상기 재생장치로부터 출력되는 상기 데이터가 부호화된 데이터인 경우, 상기 스트림 데이터가 출력된다.

[0072] 본 발명의 제 2 측면의 데이터구조 또는 기록매체에 기록되어 있는 데이터에 있어서는, 스트림 데이터의 재생순서를 관리하는 제 1 정보를 포함하고, 상기 제 1 정보는 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터의 유무에 근거하는 제 2 정보가 포함된다.

[0073] 본 발명의 제 2 측면의 기록장치에 있어서는, 스트림 데이터의 재생순서를 관리하는 제 1 정보를 포함하고, 상기 제 1 정보는 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터의 유무에 근거하는 제 2 정보를 포함하는 데이터가 취득되어, 취득된 데이터가 기록매체에 기록된다.

[0074] 본 발명의 제 2 측면의 기록매체의 제조방법에 있어서는, 재생장치에 대하여 재생되는 데이터의 데이터구조에 있어서, 스트림 데이터의 재생순서를 관리하는 제 1 정보를 포함하고, 상기 제 1 정보는 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터의 유무에 근거하는 제 2 정보를 포함하는 데이터구조를 가지는 데이터가 생성되어, 생성된 데이터가 기록매체에 기록된다.

- [0075] 네트워크란, 적어도 2개의 장치가 접속되고, 어느 장치로부터 다른 장치에 대하여 정보의 전달을 할 수 있도록 한 구조를 말한다. 네트워크를 통하여 통신하는 장치는, 독립한 장치끼리여도 좋고, 1개의 장치를 구성하고 있는 내부 블록 끼리여도 좋다.
- [0076] 또, 통신이란, 무선통신 및 유선통신은 물론, 무선통신과 유선통신이 혼재한 통신, 즉, 어느 구간에서는 무선통신이 행해지고, 다른 구간에서는 유선통신이 행해지는 것이어도 좋다. 더욱이, 어느 장치로부터 다른 장치로의 통신이 유선통신으로 행해지고, 다른 장치로부터 어느 장치로의 통신이 무선통신으로 행해지는 것이어도 좋다.
- [0077] 이하에 본 발명의 실시의 형태를 설명하나, 본 발명의 구성요건과 발명의 상세한 설명에 기재된 실시의 형태의 대응관계를 예시하면 다음과 같이 된다. 이 기재는 본 발명을 지원하는 실시의 형태가 발명의 상세한 설명에 기재되어 있는 것을 확인하기 위한 것이다. 따라서, 발명의 상세한 설명 안에는 기재되어 있으나, 본 발명의 구성요건에 대응하는 실시의 형태로서 여기에는 기재되지 않은 실시의 형태가 있었다고 해도, 그것은, 그 실시의 형태가 그 구성요건에 대응하는 것이 아닌 것을 의미하는 것은 아니다. 반대로, 실시의 형태가 구성요건에 대응하는 것으로서 여기에 기재되어 있었다고 해도, 그것은, 그 실시의 형태가 그 구성요건 이외의 구성요건에는 대응하지 않는다는 것을 의미하는 것도 아니다.
- [0078] 본 발명의 제 1 측면의 재생장치(예를 들면, 도 37의 재생장치(20-1), 도 41의 재생장치(20-2), 도 45의 재생장치(20-3))는, 부호화된 스트림 데이터(예를 들면, 오디오 스트림#1 또는 도 45의 비디오 디코더(72)로 디코드되는 비디오 스트림)를 포함하는 재생데이터를 취득하는 재생데이터 취득수단(예를 들면, 도 37, 도 41, 도 45의 재생데이터 취득부(31))과, 상기 스트림 데이터를 디코드하는 디코드 수단(예를 들면, 도 37의 오디오 디코더(75), 또는, 도 41 또는 도 45의 오디오 디코더(75-1))과, 상기 디코드 수단에 의해 디코드된 상기 스트림 데이터에, 상기 스트림 데이터와는 다른 데이터이며 상기 스트림 데이터에 합성하기 위한 데이터인 합성용 데이터(예를 들면, 사운드 데이터 또는 사운드 스트림#1, 또는, 비디오 디코더(72)로 디코드되는 비디오 스트림에 합성되는 다른 비디오 스트림 또는 화상(영상)데이터)를 합성하는 합성수단(예를 들면, 도 37, 도 41 또는 도 45의 핵심 처리부(97), 또는, 도 41 또는 도 45의 핵심 처리부(102))과, 상기 스트림 데이터를 상기 디코드 수단에 공급하는지, 출력하는지를 선택하는 선택수단(예를 들면, 도 37, 도 41, 도 45의 스위치(61))과, 상기 선택수단을 제어하는 제어수단(예를 들면, 도 37의 컨트롤러(34-1), 도 41의 컨트롤러(34-2), 또는, 도 45의 컨트롤러(34-3))을 갖추고, 상기 제어수단은, 상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터로부터, 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되어 있는지 아닌지에 근거하는 판별정보(예를 들면, is_MixApp, is_MixApp_1, 또는, is_MixApp_2)를 취득하고, 상기 판별정보에 근거하여 상기 재생데이터에 상기 합성용 데이터가 포함되어 있지 않다고 판별되고, 또, 상기 재생데이터 처리수단에 의해 처리된 데이터가 부호화데이터로서 출력되는 경우, 상기 스트림 데이터를 출력하도록, 상기 선택수단을 제어한다.
- [0079] 본 발명의 제 1 측면의 재생장치에 있어서, 상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에는, 상기 재생데이터의 타이틀에 대응한 데이터를 포함한 소정의 파일(예를 들면, Index 파일)이 포함되어 있고, 상기 제어수단은, 상기 판별정보를 상기 소정의 파일로부터 취득할 수 있다.
- [0080] 본 발명의 제 1 측면의 재생장치에 있어서, 상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에는, 상기 재생데이터의 재생순서를 나타내는 정보(예를 들면, PlayList)를 포함하는 소정의 파일(예를 들면, 도 8의 PLAYLIST 파일에 대응하는 도 13의 XXXXX.mpls)이 1개 또는 복수 포함되어 있고, 상기 제어수단은, 상기 판별정보를 상기 소정의 파일로부터 취득할 수 있다.
- [0081] 본 발명의 제 1 측면의 재생장치에 있어서, 상기 재생데이터 취득수단에 의해 취득되는 상기 재생데이터에는, 상기 재생데이터의 재생순서를 나타내는 정보이며 상기 재생데이터에 1개 또는 복수 포함되는 제 1 데이터(예를 들면, PlayList)와, 상기 제 1 데이터에 의해 재생순서가 제어되는 데이터의 재생구간을 나타내는 정보이며 상기 제 1 데이터에 1개 또는 복수 관련지어지는 제 2 데이터(예를 들면, PlayItem)가 포함되어 있고, 상기 제어수단은, 상기 판별정보를 상기 제 2 데이터로부터 취득할 수 있다.
- [0082] 본 발명의 제 1 측면의 재생방법 및 프로그램 및 프로그램 저장매체에 저장되어 있는 프로그램은, 부호화된 스트림 데이터(예를 들면, 오디오 스트림#1 또는 도 45의 비디오 디코더(72)로 디코드되는 비디오 스트림)를 포함하는 재생데이터로부터, 상기 재생데이터에 상기 스트림 데이터(예를 들면, 사운드 데이터 또는 사운드 스트림#1, 또는, 비디오 디코더(72)로 디코드되는 비디오 스트림에 합성되는 다른 비디오 스트림 또는 화상(영상)데이터)에 합성되는 합성용 데이터가 포함되어 있는지 아닌지를 나타내는 판별정보(예를 들면, is_MixApp, is_MixApp_1, 또는, is_MixApp_2)를 취득하고(예를 들면, 도 38의 단계(S2), 도 39의 단계(S64), 도 40의 단계

(S135), 도 42의 단계(S202), 도 43의 단계(S264), 도 44의 단계(S335)의 처리), 취득된 상기 판별정보를 근거로, 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되어 있는지 아닌지를 판단하여(예를 들면, 도 38의 단계(S3), 도 39의 단계(S65), 도 40의 단계(S136), 도 42의 단계(S203), 도 43의 단계(S265), 도 44의 단계(S336)의 처리), 상기 재생데이터에 상기 스트림 데이터에 합성되는 상기 합성용 데이터가 포함되어 있지 않다고 판단되고, 또, 상기 재생장치로부터 출력되는 상기데이터가 부호화된 데이터인 경우, 상기 스트림 데이터를 출력하는(예를 들면, 도 38의 단계(S11), 도 39의 단계(S71), 도 40의 단계(S141), 도 42의 단계(S211), 도 43의 단계(S271), 도 44의 단계(S341)의 처리) 단계를 포함한다.

[0083] 본 발명의 제 2 측면의 데이터구조는, 재생장치(예를 들면, 도 37의 재생장치(20-1), 도 41의 재생장치(20-2), 도 45의 재생장치(20-3))에 대하여 재생되는 데이터의 데이터구조에 있어서, 도 43의 스트림 데이터의 재생순서를 관리하는 제 1 정보(예를 들면, 도 8의 PLAYLIST에 대응하는 도 13의 XXXXX.mpls)를 포함하고, 상기 제 1 정보는, 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터(예를 들면, 사운드 데이터 또는 사운드 스트림#1, 또는, 비디오 디코더(72)로 디코드되는 비디오 스트림에 합성되는 다른 비디오 스트림 또는 화상(영상) 데이터)의 유무에 근거하는 제 2 정보(예를 들면, is_MixApp, is_MixApp_1, 또는, is_MixApp_2)를 포함한다.

[0084] 본 발명의 제 2 측면의 기록매체에 기록되어 있는 데이터는, 재생장치(예를 들면, 도 37의 재생장치(20-1), 도 41의 재생장치(20-2), 도 45의 재생장치(20-3))에 대하여 재생되는 데이터에 있어서, 스트림 데이터의 재생순서를 관리하는 제 1 정보(예를 들면, PlayList)를 포함하고, 상기 제 1 정보는, 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터(예를 들면, 사운드 데이터 또는 사운드 스트림 #1, 또는, 비디오 디코더(72)로 디코드되는 비디오 스트림에 합성되는 다른 비디오 스트림 또는 화상(영상) 데이터)의 유무에 근거하는 제 2 정보(예를 들면, is_MixApp, is_MixApp_1, 또는, is_MixApp_2)를 포함한다.

[0085] 본 발명의 제 2 측면의 기록장치는, 스트림 데이터의 재생 순서를 관리하는 제 1 정보(예를 들면, PlayList)를 포함하고, 상기 제 1 정보는, 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터(예를 들면, 사운드 데이터 또는 사운드 스트림#1, 또는, 비디오 디코더(72)로 디코드되는 비디오 스트림에 합성되는 다른 비디오 스트림 또는 화상(영상) 데이터)의 유무에 근거하는 제 2 정보(예를 들면, is_MixApp, is_MixApp_1, 또는, is_MixApp_2)를 포함하는 데이터구조를 가지는 데이터를 취득하는 취득수단(예를 들면, 도 48의 CPU(501), 통신부(509), 또는, 드라이브(501))과, 상기 취득수단에 의해 취득된 상기 데이터를 기록매체에 기록하는 기록수단(예를 들면, 도 48의 드라이브(501))을 갖춘다.

[0086] 본 발명의 제 2 측면의 제조방법은, 재생장치(예를 들면, 도 37의 재생장치(20-1), 도 41의 재생장치(20-2), 도 45의 재생장치(20-3))에 대하여 재생가능한 데이터가 기록되는 기록매체의 제조방법에 있어서, 도 43의 스트림 데이터의 재생순서를 관리하는 제 1 정보(예를 들면, PlayList)를 포함하고, 상기 제 1 정보는, 상기 스트림 데이터에 합성되는 상기 스트림 데이터와는 다른 합성용 데이터(예를 들면, 사운드 데이터 또는 사운드 스트림 #1, 또는, 비디오 디코더(72)로 디코드되는 비디오 스트림에 합성되는 다른 비디오 스트림 또는 화상(영상) 데이터)의 유무에 근거하는 제 2 정보(예를 들면, is_MixApp, is_MixApp_1, 또는, is_MixApp_2)를 포함하는 데이터구조를 가지는 데이터를 생성하는 단계와, 생성된 데이터를 기록매체에 기록하는 단계를 포함한다.

[0087] 이하, 도면을 참조하여, 본 발명의 실시의 형태에 대하여 설명한다.

[0088] 도 2를 이용하여, 본 발명을 적용한 재생장치에 대하여 설명한다.

[0089] 재생장치(20)는, 장착된, 예를 들면, 광디스크 등의 기록매체(21)에 기록되어 있는 정보, 네트워크(22)를 통하여 공급된 정보, 또는, 자기 자신의 내부의 기록매체(예를 들면, HDD 등)에 기록되어 있는 정보를 재생할 수 있다. 재생장치(20)는, 재생된 데이터를 유선 또는 무선에 의해 접속되어 있는 표시·음성 출력장치(23)에 공급하고, 화상을 표시시켜, 음성을 출력시킬 수 있다. 재생장치(20)는 또한, 재생된 데이터를, 네트워크(22)를 통하여 다른 장치에 송출할 수 있다. 재생장치(20)는, 예를 들면, 리모트카랜더(23) 외, 본체에 설치된 버튼 등의 입력 디바이스에 의해, 사용자로부터의 조작 입력을 받을 수 있다.

[0090] 기록매체(21)는, 재생장치(20)에 의해 재생가능한 영상 및 음성을 포함하는 데이터를 소정의 포맷으로 기록하고 있다. 기록매체(21)는, 광디스크의 외, 예를 들면, 자기 디스크나 반도체 메모리여도 좋다.

[0091] 표시·음성 출력장치(23)가, 디지털의 비압축 데이터의 입력을 받도록 되어 있는 경우, 재생장치(20)는, 기록매체(21) 등에 기록되어 있는 부호화데이터를 복호하고, 비압축 데이터를 표시·음성 출력장치(23)에 공급하도록 되어 있으나, 표시·음성 출력장치(23)가 복호기능을 가지고, 압축 데이터의 입력을 받도록 되어 있는 경우, 재

생장치(20)는, 압축 데이터를 표시·음성 출력장치(23)에 공급하도록 되어 있다. 또, 표시·음성 출력장치(23)가, 비압축의 아날로그 데이터의 입력을 받도록 되어 있는 경우, 재생장치(20)는, 기록매체(21) 등에 기록되어 있는 부호화데이터를 복호하고, 비압축 데이터를 D/A 변환하여 아날로그 신호를 생성하고, 표시·음성 출력장치(23)에 공급하도록 되어 있다. 재생장치(20)는 또한, 기록매체(21) 등에 기록되어 있는 데이터를 재생하여, 압축 데이터 상태로, 네트워크(22)에 송출한다.

[0092] 도 3은, 도 2의 재생장치(20)에 의해서 재생가능한 정보, 즉, 재생장치(20)에 장착되는 기록매체(21)에 기록되어 있는 데이터, 네트워크(22)를 통하여 공급되는 데이터, 또는, 재생장치(20)의 내부의 기록매체에 기록되어 있는 데이터의 어플리케이션 포맷의 예를 나타내는 도면이다.

[0093] 어플리케이션 포맷은, AV(Audio Visual) 스트림의 관리를 위해, PlayList와 Clip의 2개의 레이어를 가지고 있다. 여기에서는, 1개의 AV 스트림과 거기에 부수하는 정보인 Clip 인포메이션의 쌍(pair)을 1개의 오브젝트로 생각하고 그것들을 한꺼번에 Clip이라 칭한다. 이하, AV 스트림을 AV 스트림 파일이라고도 칭한다. 또, Clip 인포메이션을 Clip 인포메이션 파일이라고도 칭한다.

[0094] 일반적으로, 컴퓨터 등에서 이용되는 파일은 바이트 열로서 다루어지지만, AV 스트림 파일의 컨텐츠는 시간축상에 전개되고, Clip의 액세스 포인트는, 주로, 타임 스탬프로 PlayList에 의해 지정된다. 즉, PlayList와 Clip은, AV 스트림의 관리를 위한 레이어이다.

[0095] Clip 내의 액세스 포인트가 타임 스탬프로 PlayList에 의해 나타내지고 있는 경우, Clip Information 파일은, 타임 스탬프로부터, AV 스트림 파일 내의 디코드를 개시해야 할 주소정보를 찾아내기 위해 사용된다.

[0096] PlayList는, AV 스트림의 재생구간의 모임이다. 어느 AV 스트림 중의 하나의 재생구간은 PlayItem이라 불리고, 그것은, 시간축 상의 재생구간의 IN점(재생 개시점)과 OUT점(재생 종료점)의 쌍으로 나타내진다. 따라서, PlayList는, 도 3에 나타낸 바와 같이 1개, 또는 복수의 PlayItem에 의해 구성된다.

[0097] 도 3에 있어서, 왼쪽에서 1번째의 PlayList는 2개의 PlayItem으로 구성되고, 그 2개의 PlayItem에 의해, 좌측의 Clip에 포함되는 AV 스트림의 전반부분과 후반부분이 각각 참조되고 있다. 또, 왼쪽에서 2번째의 PlayList는 1개의 PlayItem으로 구성되고, 그것에 의해, 우측의 Clip에 포함되는 AV 스트림 전체가 참조되고 있다. 더욱 이, 왼쪽에서 3번째의 PlayList는 2개의 PlayItem으로 구성되고, 그 2개의 PlayItem에 의해, 좌측의 Clip에 포함되는 AV 스트림이 있는 부분과 우측의 Clip에 포함되는 AV 스트림이 있는 부분이 각각 참조되고 있다.

[0098] 네비게이션 프로그램은, PlayList의 재생순서나, PlayList의 인터랙티브한 재생을 컨트롤하는 기능을 가진다. 또, 네비게이션 프로그램은, 각종 재생의 실행을 사용자가 지시하기 위한 메뉴화면을 표시하는 기능 등도 가진다. 이 네비게이션 프로그램은, 예를 들면, Java(등록상표) 등의 프로그램 언어로 기술된다.

[0099] 예를 들면, 도 3의 네비게이션 프로그램에 의해, 그때의 재생 위치를 나타내는 정보로서, 왼쪽에서 1번째의 PlayList에 포함되는 좌측의 PlayItem이 지정되었을 경우, 그 PlayItem이 참조하는 좌측의 Clip에 포함되는 AV 스트림의 전반부분의 재생이 행해진다. 이와 같이, PlayList는, AV 스트림 파일의 재생을 관리하기 위한 재생관리정보로서 이용된다.

[0100] 네비게이션 프로그램은, 인덱스 테이블(Index table)과 인덱스 테이블에 의해 독출되는 네비게이션 오브젝트(Navigation Object)에 의해 구성된다. 도 4를 이용하여, 인덱스 테이블과 네비게이션 오브젝트에 대하여 설명한다.

[0101] 인덱스 테이블은, 컨텐츠의 타이틀이나 메뉴의 정의를 실시하는 것이며, 각 타이틀이나 메뉴의 엔트리 포인트를 보관하고 있다. FirstPlayback에는, 이 데이터를 기록하고 있는 기록매체(21)가 재생장치(20)에 장착되었을 때 등, 인덱스 테이블이 읽어 들여졌을 때에 최초로 자동적으로 실행되는 NavigationObject에 관한 정보가 포함되어 있다. TopMenu에는, 사용자가, 컨텐츠를 모두 재생한다, 특정의 챕터만 재생한다, 반복재생한다, 초기 메뉴를 표시한다 등의 내용의 항목을 포함하는 재생메뉴 화면을 표시시키는 경우 등에 호출되는 TopMenu의 NavigationObject에 관한 정보가 포함되어 있다. 각각의 Title은, 소정의 Title ID로 구별되는 타이틀마다 할당되어 호출하는 것이 가능한 NavigationObject에 관한 정보가 포함되어 있다. 이 도 4에 있어서는, 1개의 Title에 붙여 1개의 Navigation Command가 존재하고 있다.

[0102] 네비게이션 오브젝트는, 실행 가능한 커맨드인 Navigation command로 완성되어 있다. Navigation Command는, 예를 들면, PlayList의 재생이나, 다른 Navigation Object의 호출 등의 각종 커맨드를 포함한다. 예를 들면, Navigation Command#3에 PlayList#1을 재생하기 위한 명령문이 포함되는 경우, Navigation

Command#3을 실행하면 PlayList#1이 재생된다.

[0103] 인덱스 테이블을 포함하는 데이터 파일인 Index 파일에 대하여는, 도 9 내지도 11을 이용하여 후술한다.

[0104] 본 실시의 형태에서는, PlayList 중에서, 1개 이상의 PlayItem의 열에 의해(연속하는 PlayItem에 의해) 만들어지는 재생경로를 메인패스(Main Path)라 칭하고, PlayList 중에서, Main Path에 병행(병렬)하고, 1개 이상의 Sub Path의 열에 의해(비연속이라도 좋고, 연속해도 좋은 SubPlayItem에 의해) 만들어지는 재생경로를 서브패스(Sub Path)라 칭한다. 즉, 재생장치(20)로 재생가능한 데이터의 어플리케이션 포맷은, 메인패스에 관련지어(맞추어) 재생되는 서브패스(Sub Path)를 PlayList 내에 가지도록(관련지어지도록) 되어 있다.

[0105] 도 5는, 메인패스와 서브패스의 구조를 설명하는 도면이다. PlayList는, 1개의 메인패스와 1개 이상의 서브패스를 가질 수 있다. 1개의 메인패스는, 1개 이상의 PlayItem의 열에 의해 만들어지고, 1개의 서브패스는, 1개 이상의 SubPlayItem의 열에 의해 만들어진다.

[0106] 도 5의 예의 경우, PlayList는, 3개의 PlayItem의 열에 의해 만들어지는 1개의 메인패스와 3개의 서브패스를 가지고 있다. 메인패스를 구성하는 PlayItem에는, 선두로부터 차례로 각각 ID(Identification)가 첨부되어 있다. 구체적으로는, 메인패스는, PlayItem_id=0, PlayItem_id=1 및 PlayItem_id=2의 PlayItem으로 이루어진다. 또, 서브패스에도 선두로부터 차례로 Subpath_id=0, Subpath_id=1 및 Subpath_id=2로 각각 ID가 첨부되어 있다. Subpath_id=0의 서브패스에는 1개의 SubPlayItem이 포함되고, Subpath_id=1의 서브패스에는 2개의 SubPlayItem이 포함되고, Subpath_id=2의 서브패스에는 1개의 SubPlayItem이 포함된다.

[0107] Subpath_id=0의 서브패스에 포함되는 SubPlayItem이 참조하는 스트림은, 예를 들면, 영화의 일본어 더빙의 음성으로 상정되고, Mainpath에 의해 참조되는 AV 스트림 파일의 오디오 스트림으로 바꾸어 재생되는 경우가 생각될 수 있다. 또, Subpath_id=1의 서브패스에 포함되는 SubPlayItem이 참조하는 스트림은, 예를 들면, 영화의 디렉터즈 컷으로 상정되고, Main Path에 의해 참조되는 AV 스트림 파일의 소정의 부분에만 영화감독 등의 코멘트가 들어있는 경우가 생각될 수 있다.

[0108] 1개의 PlayItem이 참조하는 Clip AV 스트림 파일에는, 적어도 비디오 스트림 데이터(메인 화상 데이터)가 포함된다. 또, Clip AV 스트림 파일에는, Clip AV 스트림 파일에 포함되는 비디오 스트림(메인 화상 데이터)과 같은 타이밍으로(동기 하여) 재생되는 오디오 스트림이 1개 이상 포함되어도 좋고, 포함되지 않아도 좋다. 더욱이, Clip AV 스트림 파일에는, Clip AV 스트림 파일에 포함되는 비디오 스트림과 같은 타이밍으로 재생되는 비트맵 자막 스트림이 1개 이상 포함되어도 좋고 포함되지 않아도 좋다. 또한, Clip AV 스트림 파일에는, Clip AV 스트림 파일에 포함되는 비디오 스트림과 같은 타이밍으로 재생되는 인터랙티브 그래픽스 스트림이 1개 이상 포함되어도 좋고, 포함되지 않아도 좋다. 그리고, Clip AV 스트림 파일에 포함되는 비디오 스트림과, 비디오 스트림과 같은 타이밍으로 재생되는 오디오 스트림, 비트맵 자막 스트림 파일, 또는 인터랙티브 그래픽스 스트림은 다중화되어 있다. 즉, 1개의 PlayItem이 참조하는 Clip AV 스트림 파일에는, 비디오 스트림 데이터와, 그 비디오 스트림에 맞추어 재생되는 0개 이상의 오디오 스트림, 0개 이상의 비트맵 자막 스트림 데이터 및 0개 이상의 인터랙티브 그래픽스 스트림 데이터가 다중화되어 있다.

[0109] 즉, 1개의 PlayItem이 참조하는 Clip AV 스트림 파일에는, 비디오 스트림, 오디오 스트림, 비트맵 자막 스트림 파일, 또는, 인터랙티브 그래픽스 스트림 등의 복수의 종류의 스트림이 포함되어 있다.

[0110] 또, 1개의 SubPlayItem은, PlayItem이 참조하는 Clip AV 스트림 파일과는 다른 스트림(별개의 스트림)의 오디오 스트림 데이터나 자막 데이터를 참조한다.

[0111] 메인패스만을 가지는 PlayList를 재생하는 경우, 사용자에 의한 음성전환이나 자막전환이라고 하는 조작은, 그 메인패스가 참조하는 Clip에 다중화되어 있는 오디오 스트림과 서브 핵쳐 스트림 중에서밖에 음성이나 자막을 선택할 수 없다. 이에 대하여, 메인패스와 서브패스를 가지는 PlayList를 재생하는 경우, 그 메인패스가 참조하는 Clip AV 스트림 파일에 다중화되어 있는 오디오 스트림과 서브 핵쳐 스트림에 더하여, SubPlayItem이 참조하는 Clip의 오디오 스트림이나 서브 핵쳐 스트림을 참조할 수 있다.

[0112] 이와 같이, 1개의 PlayList 안에 SubPath를 복수 포함하여 각각의 SubPath가 각각 SubPlayItem을 참조하는 구성을 했으므로, 확장성이 높고, 또, 자유도가 높은 AV 스트림을 실현할 수 있다. 즉, MainPath로 참조되는 Clip AV 스트림에 더하여, 나중에, SubPlayItem을 추가할 수 있는 구성을 할 수 있다.

[0113] 도 6은, 메인패스와 서브패스의 예를 설명하는 도면이다. 도 6에 있어서는, 메인패스와 같은 타이밍으로(AV 동기하여) 재생되는 오디오의 재생패스를, 서브패스를 사용하여 나타내고 있다.

- [0114] 도 6의 PlayList에는, 메인패스로서 PlayItem_id=0인 1개의 PlayItem과, 서브패스로서 1개의 SubPlayItem이 포함되어 있다. 메인패스의 PlayItem_id=0인 1개의 PlayItem()은, 도 6의 메인 AV 스트림을 참조하고 있다. SubPlayItem()에는, 이하에 나타내는 데이터가 포함된다. 우선, SubPlayItem()에는, PlayList 내의 Sub Path(서브패스)가 참조하는 Clip을 지정하기 위한 Clip_Information_file_name이 포함된다. 도 6의 예의 경우, SubPlayItem에 의해, SubClip_entry_id=0의 Auxiliary audio stream(오디오 스트림)이 참조되고 있다. 또, SubPlayItem()에는, 지정된 Clip에 포함되는 스트림(여기서는, Auxiliary audio stream) 내의 Sub Path의 재생구간을 지정하기 위한 SubPlayItem_IN_time과 SubPlayItem_OUT_time이 포함된다. 더욱이, SubPlayItem()에는, Main path의 시간축상에서 Sub Path가 재생개시하는 시각을 지정하기 위한 sync_PlayItem_id와 sync_start PTS_of_PlayItem이 포함된다. 도 6의 예의 경우, sync_PlayItem_id=0이 되고, sync_start PTS_of_PlayItem=t1으로 되어 있다. 이것에 의해, 메인패스의 PlayItem_id=0의 시간축상에서 Sub Path가 재생을 개시하는 시각(t1)을 지정할 수 있다. 즉, 도 6의 예의 경우에는, 메인패스의 재생 개시시각(t1)과 서브패스의 개시시각(t1)이 동일시각인 것을 나타내고 있다.
- [0115] 여기서, Sub Path에 참조되는 오디오의 Clip AV 스트림은, STC 불연속점(시스템 타임 베이스의 불연속점)을 포함해서는 안 된다. 서브패스에 사용되는 Clip에 포함되는 스트림의 오디오 샘플의 클럭은, 메인패스에 사용되는 Clip에 포함되는 스트림의 오디오 샘플의 클럭에 고정되어 있다.
- [0116] 환언하면, SubPlayItem()에는, Sub Path가 참조하는 Clip을 지정하는 정보, Sub Path의 재생구간을 지정하는 정보 및 Main path의 시간축 상에서 Sub Path가 재생을 개시하는 시각을 지정하는 정보가 포함되어 있다. Sub Path에 사용되는 Clip AV 스트림이 STC를 포함하지 않으므로, SubPlayItem()에 포함되는 정보(Sub Path가 참조하는 Clip을 지정하는 정보, Sub Path의 재생구간을 지정하는 정보 및 Main path의 시간축상에서 Sub Path가 재생을 개시하는 시각을 지정하는 정보)에 근거하여, 메인패스가 참조하는 Clip AV 스트림(메인 AV 스트림)과는 다른 Clip AV 스트림의 오디오 스트림을 참조하고, 재생할 수 있다.
- [0117] 이와 같이, PlayItem과 SubPlayItem은, Clip AV 스트림 파일을 각각 관리하는 것이며, 여기서는, PlayItem이 관리하는 Clip AV 스트림 파일(메인 AV 스트림)과 SubPlayItem이 관리하는 Clip AV 스트림 파일은 다른 파일이 된다.
- [0118] 또한, 도 6의 예와 같이 하여, 메인패스와 같은 타이밍으로 재생되는 자막 스트림 재생경로를 서브패스를 사용하여 나타낼 수도 있다.
- [0119] 도 7은, 메인패스와 서브패스의 다른 예를 설명하는 도면이다. 도 7에 있어서는, 메인패스와 같은 타이밍으로(AV 동기하여) 재생되는 오디오의 재생경로를, 서브패스를 사용하여 나타내고 있다. 여기서, 메인패스의 PlayItem이 참조하는 메인 AV 스트림 파일은, 도 6과 같은 것으로 생략하고 있다.
- [0120] 예를 들면, 메인패스에 의해 참조되는 Clip AV 스트림을 1개의 영화의 컨텐츠(AV 컨텐츠)로 하고, 서브패스의 오디오 패스에 의해 참조되는 Auxiliary audio stream을 그 영화에 대한 감독의 코멘트로 하고, 메인패스로 참조되는 Clip AV 스트림의 오디오 스트림에 서브패스의 오디오 패스로 참조되는 Auxiliary audio stream을 믹싱(중첩하여) 재생하는 경우에, 이러한 구성이 이용된다. 즉, 사용자가, 영화를 보면서, 그 영화에 대한 감독의 코멘트를 맞추어 듣는 지령을 재생장치(플레이어, player)에 입력했을 경우, 예를 들면, 메인패스로 참조되는 Clip AV 스트림의 음성과 서브패스의 오디오 패스로 참조되는 Auxiliary audio stream을 믹싱하여 재생시키는 경우에 이용된다.
- [0121] 도 7에 있어서는, 메인패스에 3개의 PlayItem이 PlayItem_id=0, 1, 2로 배치되고, 서브패스(Subpath_id=0)에 2개의 SubPlayItem이 배치되어 있다. 그리고, Subpath_id=0의 Sub Path(도 19에서 후술한다)로 호출하는 SubPlayItem(도 21에서 후술한다)은, Auxiliary audio stream(SubClip_entry_id=0의 영어의 Auxiliary audio stream의 클립 및 SubClip_entry_id=1의 일본어의 Auxiliary audio stream의 클립)의 Sub path의 재생구간을 지정하기 위한, SubPlayItem_IN_time과 SubPlayItem_out_time을 포함한다.
- [0122] 도 7과 도 6을 비교함에서, 도 7에 있어서는, SubPlayItem에 의해, SubClip_entry_id=0, 1의 Auxiliary audio stream(영어 또는 일본어의 오디오 스트림)을 참조할 수 있다. 즉, SubPlayItem을 이용하여, 복수의 오디오 스트림 파일을 참조하는 구조를 가지고, 이 SubPlayItem을 재생하는 경우에는, 복수의 오디오 스트림 파일로부터 1개의 오디오 스트림 파일이 선택되어 재생된다. 도 7의 예의 경우, 영어의 오디오 스트림 파일과 일본어의 오디오 스트림 파일 중, 1개의 오디오 스트림 파일이 선택되어 재생된다. 구체적으로는, subClip_entry_id=0, 1 중에서(사용자의 지령에 근거하여) 1개가 선택되고, 그 ID가 참조하는 Auxiliary audio

stream이 재생된다. 또, 이것에 더하여, 또한 메인패스에 의해 참조되는 오디오 스트림과의 맵싱에 의한 재생이 선택되었을 경우(재생하는 오디오 스트림으로서 2개의 오디오 스트림이 선택되었을 경우), 예를 들면, 메인패스로 참조되는 오디오 스트림 파일과 서브패스의 오디오 패스로 참조되는 오디오 스트림 파일이 맵싱되어 재생된다.

[0123] 도 8은, 재생장치(20)에 대하여 재생가능한 데이터 파일의 파일시스템의 예를 나타내는 도면이다. 도 8에는, 일례로서 재생장치(20)에 대하여 재생가능한 데이터 파일이 왕디스크 등의 기록매체(21)에 의해 공급되고 있는 경우를 나타내고 있고, 이 파일 시스템은 디렉토리 구조를 가지고 있다.

[0124] 이 파일 시스템에 있어서, 「root」 아래에는 「BDMV」의 이름이 설정된 디렉토리가 준비되고, 그 디렉토리에, 「Index.bdmv」의 이름이 설정된 파일과, 「NavigationObject.bdmv」의 이름이 설정된 파일이 저장되어 있다. 이하, 적절하게, 이러한 파일을 각각 Index 파일, NavigationObject 파일이라 칭한다. 또, 적절하게, 각 파일에 대하여는, 「파일명」에 「파일」을 부가한 형태로, 또는, 각 디렉토리에 대하여는, 「디렉토리명」에 「디렉토리」를 부가한 형태로 칭한다.

[0125] Index 파일은, 상술한 인덱스 테이블이 기재된 파일이며, 재생장치(20)에 대하여 재생가능한 데이터 파일을 재생하는 메뉴에 관한 정보를 포함한다. 재생장치(20)는, 예를 들면, 재생장치(20)에 대하여 재생가능한 데이터 파일에 포함되는 컨텐츠를 모두 재생한다, 특정의 챕터만 재생한다, 반복재생한다, 초기메뉴를 표시한다 등의 내용의 항목을 포함하는 재생메뉴 화면을 Index 파일에 근거하여 표시장치에 표시시킨다. Index 파일의 인덱스 테이블에는 각 항목이 선택되었을 때에 실행하는 NavigationObject를 설정할 수 있고, 사용자에 의해 재생메뉴 화면으로부터 1개의 항목이 선택되었을 경우, 재생장치(20)는 Index 파일의 인덱스 테이블로 설정되어 있는 NavigationObject의 커맨드를 실행한다.

[0126] NavigationObject 파일은, NavigationObject를 포함하는 파일이다. NavigationObject는, 재생장치(20)에 대하여 재생가능한 데이터 파일에 포함되어 있는 PlayList의 재생을 제어하는 커맨드를 포함하고, 예를 들면, 재생장치(20)는, 이 파일 시스템에 포함되어 있는 NavigationObject중에서 1개를 선택하고, 실행함으로써, 컨텐츠를 재생시킬 수 있다.

[0127] BDMV 디렉토리에는 또, 「BACKUP」의 이름이 설정된 디렉토리(BACKUP 디렉토리), 「PLAYLIST」의 이름이 설정된 디렉토리(PLAYLIST 디렉토리), 「CLIPINF」의 이름이 설정된 디렉토리(CLIPINF 디렉토리), 「STREAM」의 이름이 설정된 디렉토리(STREAM 디렉토리), 「AUXDATA」의 이름이 설정된 디렉토리(AUXDATA 디렉토리)가 설정되어 있다.

[0128] BACKUP 디렉토리에는, 재생장치(20)에 대하여 재생가능한 파일이나 데이터를 백업하기 위한 파일이나 데이터가 기록된다.

[0129] PLAYLIST 디렉토리에는, PlayList 파일이 저장된다. 각 PlayList 파일에는, 도면 내에 나타낸 바와 같이 5자리 수의 숫자로 이루어지는 파일명에 확장자(extension) 「.mpls」를 부가한 명칭이 명명된다.

[0130] CLIPINF 디렉토리에는, Clip Information 파일이 저장된다. 각 Clip Information 파일에는, 도면 내에 나타낸 바와 같이 5자리 수의 숫자로 이루어지는 파일명에 확장자(extension) 「.clpi」를 부가한 명칭이 명명된다.

[0131] STREAM 디렉토리에는, Clip AV 스트림 파일이나 서브 스트림 파일이 저장된다. 각 스트림 파일에는, 도면 내에 나타낸 바와 같이 5자리 수의 숫자로 이루어지는 파일명에 확장자(extension) 「.m2ts」를 부가한 명칭이 명명된다.

[0132] AUXDATA 디렉토리에는, Clip AV 스트림 파일이나 서브 스트림 파일에 포함되지 않고, Clip AV 스트림 파일이나 서브 스트림 파일로부터 참조되는 데이터나, Clip AV 스트림 파일이나 서브 스트림 파일과는 독립하여 이용되는 데이터 등의 파일이 저장된다. 도 8의 예에 있어서는, AUXDATA 디렉토리에는, 「11111.otf」의 이름이 붙여져 있는 자막의 폰트 파일, 「sound.bdmv」의 이름이 설정된 효과음 등의 사운드 데이터가 저장되어 있다.

[0133] 또, 재생장치(20)에 대하여 재생가능한 데이터 파일이 왕디스크에 의해 배포되고 있는 경우, 예를 들면, 제작회사나 영화의 배급회사 등, 이 컨텐츠 제작원이나, 또는, 이 기록매체의 공급원인 타이틀 저자(author)를 식별하기 위해 각 타이틀 저자에 할당된 식별자인 author_id 및 author_id에 나타나는 타이틀 저자에 대하여 제작된 왕디스크의 종류를 식별하기 위해 할당된 식별자인 disc_id가, 사용자 등에 의해 고쳐 쓸 수 없는 보안 전자 데이터로서, 또는, 물리적으로 피트에 의해 기록되어 있다.

- [0134] 또, 재생장치(20)에 대하여 재생가능한 데이터 파일이 광디스크 이외의 리무버를 기록매체에 기록되어 있는 경우나, 네트워크를 통하여 다운로드 되고 있는 경우도, author_id 및 author_id에 상당하는 id가 할당되어 각각 구별 가능하게 되고, 도 8에 나타내는 경우와 같은 디렉토리 구조를 가지고 있으면 적합하다. 또, 재생장치(20)에 대하여 재생가능한 데이터 파일에는, author_id 및 author_id에 상당하는 id가 할당될 수 없다고 해도, 도 8을 이용하여 설명한 경우와 마찬가지로, 「Index.bdmv」의 이름이 설정된 파일, 「NavigationObject.bdmv」의 이름이 설정된 파일이 포함되어 있고, 「BACKUP」의 이름이 설정된 파일군(群), 「PLAYLIST」의 이름이 설정된 파일군, 「CLIPINF」의 이름이 설정된 파일군, 「STREAM」의 이름이 설정된 파일군, 「AUXDATA」의 이름이 설정된 파일군 가운데, 적절하게 필요한 파일군이 포함되어 있다.
- [0135] 그런데, 재생장치(20)는, 음성 데이터를 비압축 데이터로서 출력하는 경우와 압축 부호화된 데이터로서 출력하는 경우가 있다. 또, 재생장치(20)는, 음성 데이터를 필요에 따라 합성할 수 있다. 재생장치(20)는, 예를 들면, 주로 재생 출력되는 음성 데이터에 대하여 효과음을 합성하는 기능을 가질 수 있고, 또, 효과음뿐만 아니라, 제 2 음성(부음성 등)을 주로 재생 출력되는 음성 데이터에 합성하는 기능을 가질 수 있다. 환언하면, 재생장치(20)가 음성 데이터의 합성을 행하는지 아닌지는, 재생되는 스트림에, 재생장치(20)가 가지는 기능에 있어서 합성가능한 음성 데이터가 포함되어 있는지 아닌지에 따라서 결정된다.
- [0136] 이하, 주로 재생 출력되는 음성 데이터를 오디오 스트림#1이라 칭하고, 예를 들면, 사용자의 조작입력에 대응하여 발생하는 클릭음이나 효과음 등 상술한 AUXDATA 디렉토리에 sound.bdmv로서 기재되어 있는 음성 데이터를 사운드 데이터라 칭하는 것으로 한다. 사운드 데이터는 오디오 스트림#1에 합성가능하다. 또, 오디오 스트림#1과는 다른 스트림이며, 예를 들면, 부음성 등 오디오 스트림#1에 합성가능한 음성 데이터를 오디오 스트림#2라 칭하는 것으로 한다. 또한, 오디오 스트림#1과는 다른 스트림이며, 오디오 스트림#1에 합성가능한 음성 데이터는 오디오 스트림#2 이외에도 복수 존재해도 좋고, 그 경우, 예를 들면, 오디오 스트림#3, 오디오 스트림#4 . . . 등으로 하여, 각각 구별가능하도록 하면 좋다.
- [0137] 오디오 스트림#1이 압축 부호화데이터로서 출력되는 경우라도, 오디오 스트림#2나 사운드 데이터와의 음성 합성이 행해지는 경우, 오디오 스트림#1은, 일단 복호되고, 합성처리가 행해진 후, 재부호화 되어 출력된다. 이것에 대하여, 오디오 스트림#1이 압축 부호화데이터로서 출력되고, 오디오 스트림#2나 사운드 데이터와의 음성 합성이 행해지지 않는 경우, 오디오 스트림#1은, 디코드 되지 않고 그대로 출력되는 편이 음질이 열화하지 않고, 적합하다.
- [0138] 따라서, 재생장치(20)에 있어서는, 오디오 스트림#1에 대하여 복호처리를 실시하는지 아닌지를 결정하기 위해, 출력의 형태와, 이 데이터 파일에 포함되어 있는 음성 데이터의 종별(오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지)을 인식할 필요가 있다. 이 때문에, 재생장치(20)에 의해 재생가능한 데이터 파일에는, 사운드 데이터가 포함되어 있는지 아닌지를 나타내는 플래그나, 오디오 스트림#2 가 포함되어 있는지 아닌지를 나타내는 플래그가 소정의 위치에 기재되어 있다.
- [0139] 또한, 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 기재되는 위치로서는, 예를 들면, Index 파일, 플레이 리스트(Playlist) 파일, 플레이 아이템(PlayItem) 등이 있다. 이러한 플래그가 Index 파일에 기재되어 있는 경우, 도 8을 이용해 설명한 데이터구조에 포함되는 모든 데이터의 재생에 있어서, 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있다고 취급되는지, 포함되어 있지 않다고 취급되는지가 규정된다. 마찬가지로, 이러한 플래그가 플레이 리스트에 기재되어 있는 경우, 대응하는 플레이 리스트를 근거로 재생되는 데이터에 포함되는 오디오 스트림#1에 대하여 합성되는 다른 음성 데이터가 포함되어 있다고 취급되는지, 포함되어 있지 않다고 취급되는지가 규정되고, 이러한 플래그가 플레이 아이템에 기재되어 있는 경우, 대응하는 플레이 아이템에 대응하는 클립에 오디오 스트림#1에 대하여 합성되는 다른 음성 데이터가 포함되어 있다고 취급되는지, 포함되어 있지 않다고 취급되는지가 규정된다.
- [0140] Index 파일(Index.bdmv)의 구문(syntax)을 도 9에 나타낸다.
- [0141] Type_indicator는, ISO 646에 근거하여 부호화된 "INDX"의 값이 기술되는 필드이다.
- [0142] version_number는, Index.bdmv의 버전 넘버를 나타내는 4 문자의 문자열이며, ISO 646에 근거하여 부호화된 "0089"의 값이 기술된다.
- [0143] Indexes_start_address는, Indexes() 블록의 시작주소가 기재되는 필드이다.
- [0144] AppInfoBDMV()에는, 이 Index 파일을 포함하는 데이터 파일의 공급원을 나타내고, 개별의 식별자 등이 기재되어

있다.

[0145] Indexes()에는, 예를 들면, 메뉴를 독출하거나, 타이틀을 검색하거나, 소정의 타이틀로 점프하거나, 또는, 이 Index 파일을 포함하는 데이터 파일이 기록되어 있는 광디스크 등의 기록매체(21)가 재생장치(20)에 장착되었을 경우에 자동적으로 기동하는 처리를 실행하는 어플리케이션(NavigationObject)과의 링크가 기재되어 있다. Indexes()의 구문에 대하여는, 도 10 내지 도 12를 이용하여 후술한다.

[0146] Padding_word는, Index.bdmv의 구문에 근거하여 삽입되는 것이며, 0 또는 양의 정수로 나타내진다.

[0147] Indexes()의 구문의 제 1 예를 도 10에 나타낸다. 도 10은, Indexes()에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 Index()에 기재되어 있지 않은 경우의 구문이다.

[0148] length는, Indexes()에 몇 바이트의 정보가 기재되어 있는지를 나타내는 필드이다.

[0149] FirstPlayback은, 이 파일 시스템의 데이터가 재생되는 경우에 최초로 실행되는 NavigationObject에 관한 정보가 포함되어 있는 데이터 블록이다. 구체적으로는, 이 파일 시스템의 데이터가 기록된 광디스크 등의 기록매체(21)가 재생장치(20)에 장착되었을 때에 최초로 자동적으로 실행되는 NavigationObject에 관한 정보가 포함되어 있는 데이터 블록이다.

[0150] FirstPlayback_mobj_id_ref는, 최초로 실행되는 NavigationObject의 mobj_id의 값을 지정하기 위한 필드이다. mobj_id는, NavigationObject에 각각 고유하게 주어져 있는 ID이다. 또, 재생개시시에 자동적으로 실행되는 NavigationObject가 설정되어 있지 않은 경우, 즉, 재생개시시에는 어플리케이션은 어느 것도 실행되지 않고, 사용자에 의한 지령을 받아, 어느 어플리케이션이 실행되도록 되어 있는 경우, 이 필드에는, "0xFFFF"가 기재된다.

[0151] TopMenu()는, 사용자가 메뉴화면을 표시시키는 경우 등에 호출되는 TopMenu의 NavigationObject에 관한 정보가 포함되어 있는 데이터 블록이다.

[0152] TopMenu_mobj_id_ref는, TopMenu의 NavigationObject의 mobj_id의 값을 지정하기 위한 필드이다. TopMenu가 설정되지 않은 경우, 이 필드에는, "0xFFFF"가 기재된다.

[0153] number_of_Titles는, Indexes()에 기재되어 있는 타이틀의 수를 나타내는 필드이다.

[0154] Title[title_id]()는, title_id에 의해 고유하게 구별되어 있는 각각의 타이틀에 관한 정보가 기재되는 블록이다. title_id는 0부터 할당할 수 있다.

[0155] Title_playback_type[title_id]는, 그 title_id에 의해 지정되는 타이틀의 재생타입을 나타내는 정보이며, 예를 들면, 재생되는 동영상과 음성에 의한 일반적인 컨텐츠인 Movie Title이나, 사용자의 조작입력에 의해 재생되는 동영상이나 음성이 변경되는 등의 쌍방향의 교환이 가능한 컨텐츠인 Interactive Title 등의 타입이 표시된다. 예를 들면, 그 타이틀이 Movie Title인 경우, Playlist에 근거한 재생처리가 실행된다.

[0156] Title_access_type[title_id]는, 그 title_id에 의해 지정되는 타이틀이, Title_Search를 이용하여 재생하는 것이 허가되어 있는지 아닌지를 나타내는 정보이다.

[0157] reserved_for_future_use는, 장래의 확장에 대비하기 위해 데이터의 기술(記述) 내용이 미설정으로 되어 있는 29비트의 필드이다.

[0158] Title_mobj_id_ref[title_id]는, 그 title_id에 의해 지정되는 타이틀이 엔트리 되어 있는 NavigationObject의 mobj_id의 값을 지정하기 위한 필드이다.

[0159] 다음으로, Indexes()의 구문의 제 2 예를 도 11에 나타낸다. 도 11의 예에 있어서는, Indexes()에 의해 참조되는 데이터에는, 사운드 데이터만 합성되도록(오디오 스트림#2는 포함되지 않음) 되어 있는 경우의 것이므로, Index()에는, Indexes()에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 사운드 데이터가 포함되어 있는지 아닌지의 플래그가 기재되어 있다.

[0160] 도 11에 나타내는 Indexes()의 구문의 제 2 예에 있어서는, Indexes()에 의해 참조되는 데이터에 사운드 데이터가 포함되어 있는지 아닌지를 나타내는 1비트의 플래그인 is_MixApp가 새롭게 기재되고, reserved_for_future_use가 29비트에서 28비트로 변경되어 있는 이외에는, 도 10을 이용하여 설명한 Indexes()의 구문의 제 1 예와 동일한 구성을 가지고 있다. 또한, 이 is_MixApp를, Indexes()에 의해 참조되는 데이터

터에 오디오 스트림 #1에 합성될 수 있는 사운드 데이터 또는 오디오 스트림 #2 중 적어도 어느 한쪽이 포함되어 있는지 아닌지를 나타내는 플래그로서 정의해도 좋다. 이 경우에는, 이 1개의 플래그를 확인하는 것만으로 오디오 데이터의 합성이 행해지는지 아닌지, 즉, 디코드 처리의 필요성을 신속히 인식하는 것이 가능해진다.

[0161] 다음으로, Indexes()의 구문의 제 3 예를 도 12에 나타낸다. 도 12의 예에 있어서는, Index()에는, Indexes()에 의해 참조되는 데이터에 오디오 스트림 #1에 합성되는 사운드 데이터가 포함되어 있는지 아닌지의 플래그와, 오디오 스트림 #1에 합성되는 오디오 스트림 #2가 포함되어 있는지 아닌지의 플래그의 2 종류가 기재되어 있다.

[0162] 도 12에 나타내는 Indexes()의 구문의 제 3 예에 있어서는, Indexes()에 의해 참조되는 데이터에 오디오 스트림 #2가 포함되어 있는지 아닌지를 나타내는 1비트의 플래그인 is_MixApp_1과, Indexes()에 의해 참조되는 데이터에 사운드 데이터가 포함되어 있는지 아닌지를 나타내는 1비트의 플래그인 is_MixApp_2의 2 종류의 플래그가 새롭게 기재되고, reserved_for_future_use가 29비트에서 27비트로 변경되어 있는 이외에는, 도 10을 이용하여 설명한 Indexes()의 구문의 제 1 예와 동일한 구성을 가지고 있다.

[0163] 또, 오디오 스트림 #1에 합성해야 할 데이터(오디오 스트림 #2, 또는 사운드 데이터 중 적어도 어느 한쪽)가 Indexes()에 의해 참조되는 데이터에 포함되어 있는지 아닌지를 나타내는 플래그가 아니라, 합성처리가 행해질 수 있는가, 행해지지 않는가를 나타내는 플래그를 정의하는 것으로 해도 좋다. 이 경우에는, 정의되는 플래그는, 재생순서를 관리하는 Index, PlayList 등에 근거하여 재생을 행할 때, 합성 데이터가 그 PlayList에 대하여 적용될 수 있는지, 또는, 적용되지 않는지를 나타내는 플래그가 된다.

[0164] 도 13에, PlayList 파일의 데이터구조를 나타낸다. PlayList 파일은, 재생장치(20)에 장착된 기록매체, 또는, 재생장치(20) 내부의 기억부(예를 들면, 하드디스크 등으로 이루어지는 로컬 스토리지)에 있어서, PLAYLIST 디렉토리에 저장되고, 확장자(extension) 「.mpls」를 가지는 데이터 파일이다.

[0165] type_indicator는, 이 파일의 종류를 나타내는 정보가 기재되어 있는 것이다. 즉, 이 필드에는, 이 파일의 종류가 영상의 재생을 관리하는 재생관리 정보인 PlayList(MoviePlayList)인 것을 나타내는 정보가 기재된다.

[0166] version_number는, 이 xxxx.mpls(MoviePlayList)의 버전 넘버를 나타내는 4개의 캐릭터 문자를 나타내는 것이다.

[0167] PlayList_start_address는, PlayList 파일의 선두의 바이트로부터의 상대 바이트 수를 단위로 하여, PlayList()의 선두주소를 나타내는 것이다.

[0168] PlayListMark_start_address는, PlayList 파일의 선두의 바이트로부터의 상대 바이트 수를 단위로 하여, PlayListMark()의 선두주소를 나타내는 것이다.

[0169] ExtensionData_start_address는, PlayList 파일의 선두의 바이트로부터의 상대 바이트 수를 단위로 하여, ExtensionData()의 선두주소를 나타내는 것이다.

[0170] AppInfoPlayList()에는, 예를 들면, 재생제한 등, PlayList의 재생 컨트롤에 관한 파라미터가 저장되어 있다. AppInfoPlayList()의 상세한 것에 대하여는 도 14 또는 도 15를 이용하여 후술한다.

[0171] PlayList()에는, PlayList의 메인패스나 서브패스 등에 관한 파라미터가 저장되어 있다. PlayList()의 상세한 것은 도 16 내지 도 18을 이용하여 후술한다.

[0172] PlayListMark()에는, PlayList의 마크정보, 즉, 캡터 점프 등을 지령하는 사용자 오퍼레이션 또는 커맨드 등에 대한 점프 목적지(점프 포인트)인 마크에 관한 정보가 저장되어 있다.

[0173] ExtensionData()에는, 프라이빗(private) 데이터가 삽입될 수 있도록 되어 있다.

[0174] 다음으로, 도 14를 참조하여, AppInfoPlayList()의 제 1 예에 대하여 설명한다. 도 14의 AppInfoPlayList()는, Indexes()가, 도 10을 이용하여 설명한 것일 때, 즉, Indexes()에 의해 참조되는 데이터에 오디오 스트림 #1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 Index()에 기재되어 있지 않은 경우에 적용할 수 있다.

[0175] length는, length 이하의 구문의 길이, 즉, 이 length 필드의 직후부터 reserved_for_future_use의 끝까지의 AppInfoPlayList()의 바이트 수를 나타내는 것이다. 이 length의 뒤에는, 8비트의 reserved_for_future_use 가 준비된다.

- [0176] `Playlist_playback_type`은, 연속재생, 랜덤 억세스, 셔플 중 어느 재생을 행하는 Playlist인가를 나타내는 정보를 기재하는 필드이다.
- [0177] `Playback_count`는, 랜덤 억세스, 셔플을 행하는 PlayList의 경우, 재생에 이용하는 PlayItem의 수를 나타내는 정보를 기재하는 필드이다.
- [0178] 그리고, `UO_mask_table()`은, 일시정지, 챕터 검색(search), 점프, 빨리감기, 되감기(후방 고속재생) 등의 특수 재생, 또는, 표시 등에 관한 사용자 조작의 제한을 결정하는 정보이다.
- [0179] 그리고, `PlayList_random_access_flag`는, 다른 PlayList로부터의 점프재생을 제어하기 위한 플래그 정보이다. 여기서는, `PlayList_random_access_flag=1`로 설정되어 있는 경우, 다른 PlayList로부터의 점프재생을 금지하는 것으로 한다.
- [0180] 예를 들면, 재생장치(20)에 있어서는, 예를 들면, 현재의 PlayList에 의해 참조되고 있는 재생위치로부터, 다른 PlayList에 의해 참조되는 소정의 PlayItem의 선두 또는 챕터의 위치로의 재생위치의 점프를 지령하는 사용자 오퍼레이션 등, 장치 독자적으로 확장적인 사용자 오퍼레이션을 준비할 수 있다. `PlayList_random_access_flag`는, 예를 들면, 이러한 사용자 오퍼레이션에 의해, 다른 PlayList에 의해 참조되는 Clip AV Stream이 있는 재생 위치로부터 이 PlayList에 의해 참조되는 Clip AV Stream으로의 점프(재생위치의 이동)가 지령된 경우, 이 사용자 오퍼레이션을 제한하는지 아닌지를 설정하기 위한 플래그 정보이다.
- [0181] 또한, 사용자 오퍼레이션에 의해 점프(재생위치의 이동)가 지령되는 것이 아니라 네비게이션 커맨드에 의해 점프(재생위치의 이동)가 지령된 경우, `PlayList_random_access_flag`는 무시된다(커맨드는 실행되고, 커맨드의 발생에 수반하여 재생위치가 이동된다).
- [0182] `Is_MixApp`는, 이 Playlist 파일에 의해 재생되는 스트림이 음성이나 효과음의 합성을 행하는지 아닌지의 플래그이다. 즉, 이 `is_MixApp`는, 이 PlayList에 의해 참조되는 데이터에, 오디오 스트림#1에 합성될 수 있는 사운드 데이터 및 오디오 스트림#2가 포함되어 있는지 아닌지를 나타내는 플래그로서 정의되어 있다. 이 경우는, 이 1개의 플래그를 확인하는 것만으로, 오디오 데이터의 합성이 행해지는지 아닌지, 즉, 디코드 처리의 필요성을 신속하게 인식하는 것이 가능해진다.
- [0183] 또, `Lossless_may_bypass_mixer_flag`는, 무손실(lossless)의 음성의 재생에 관한 플래그이다. 이 다음에는, 13비트의 `reserved_for_future_use`가 준비된다.
- [0184] 도 10에 나타낸 바와 같이, `Indexes()`에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 `Indexes()`에 기재되어 있지 않은 경우에 있어서는, 도 14를 이용하여 설명한 바와 같이, `AppInfoPlayList()`에 `Is_MixApp`를 기재하고, 이 PlayList가 음성이나 효과음의 합성을 행하는지 아닌지를 나타내도록 해도 좋으나, 예를 들면, `AppInfoPlayList()`에 `Is_MixApp`를 기재하는 것이 아니라 `PlayList()` 또는 `PlayItem`에 `Is_MixApp`를 기재하는 것으로 해도 좋다.
- [0185] 다음으로, 도 15를 참조하여, `AppInfoPlayList()`의 제 2 예에 대하여 설명한다.
- [0186] 도 15에 나타내는 `AppInfoPlayList()`의 제 2 예는, 이 PlayList가 음성이나 효과음의 합성을 행하는지 아닌지의 플래그인 `Is_MixApp`가 기재되어 있지 않은 것 이외에는, 도 14를 이용하여 설명한 `AppInfoPlayList()`의 제 1 예와 동일한 구성을 가지고 있다. 즉, 도 15의 `AppInfoPlayList()`의 제 2 예는, `Indexes()`가 도 10을 이용하여 설명한 것이나, `Is_MixApp` 등의 플래그가 후술하는 `PlayList()` 또는 `PlayItem`에 기재되어 있을 때, 즉, `Indexes()`에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 `Index()`에 기재되지 않고, `PlayList()` 또는 `PlayItem`에 `Is_MixApp`가 기재되는 경우, 또는, `Indexes()`가 도 11 또는 도 12를 이용하여 설명한 것일 때, 즉, `Indexes()`에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 `Index()`에 기재되어 있는 경우에 적용된다.
- [0187] 다음으로, 도 16은, `PlayList()`의 구문의 제 1 예를 나타내는 도면이다. 도 16은, 오디오 스트림#1에 합성되는 다른 음성 데이터가 이 플레이 리스트를 참조하는 것에 의해 재생되는 데이터에 포함되어 있는지 아닌지의 플래그가 `PlayList()`에 기재되지 않은 경우의 구문이다.
- [0188] 즉, 도 16의, `PlayList()`의 구문의 제 1 예는, `Indexes()`가 도 10을 이용하여 설명한 것일 때, 즉, `Indexes()`에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 `Index()`에 기재되어 있지 않으나, 후술하는 `PlayItem`에 기재되어 있는 경우, 또는, `Indexes()`가 도 11 또는

도 12를 이용하여 설명한 것일 때, 즉, Indexes()에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 Index()에 기재되어 있는 경우에 적용된다.

[0189] length는, 이 length 필드의 직후부터 PlayList()의 끝까지의 바이트 수를 나타내는 32비트의 부호 없는 정수이다. 즉, reserved_for_future_use로부터 Playlist의 끝까지의 바이트 수를 나타내는 필드이다. 이 length의 다음에는 16비트의 reserved_for_future_use가 준비된다. reserved_for_future_use는 장래의 확장에 대비하기 위하여 데이터의 기술내용이 미설정으로 되어 있다. number_of_PlayItems는 PlayList 내에 있는 PlayItem의 수를 나타내는 16비트의 필드이다. 예를 들면, 도 5의 예의 경우 PlayItem의 수는 3개이다. PlayItem_id의 값은 PlayList 내에서 PlayItem()이 나타내는 차례대로 0부터 할당된다. 예를 들면, 도 5 및 도 7에 나타낸 바와 같이, PlayItem_id=0, 1, 2가 할당된다.

[0190] number_of_SubPaths는 PlayList 내에 있는 SubPath의 수(엔트리 수)를 나타내는 16비트의 필드이다. 예를 들면, 도 5의 예의 경우, SubPath의 수는 3개이다. SubPath_id의 값은 PlayList 내에서 SubPath()가 나타내는 차례대로 0부터 할당된다. 예를 들면, 도 5에 나타낸 바와 같이, Subpath_id=0, 1, 2가 할당된다. 그 후의 for문에서는 PlayItem의 수만큼 PlayItem이 참조되고, SubPath의 수만큼 SubPath가 참조된다.

[0191] 다음으로, 도 17은 PlayList()의 구문의 제 2 예를 나타내는 도면이다. 도 17의 예에 있어서는, 이 플레이리스트를 참조하는 것에 의해 재생되는 오디오 스트림은 오디오 스트림#1 뿐이고(오디오 스트림#2는 포함되지 않음), PlayList()에는 PlayList()를 참조하여 재생되는 데이터에 오디오 스트림#1에 합성되는 사운드 데이터가 포함되어 있는지 아닌지의 플래그가 기재되어 있다.

[0192] 즉, 도 17의 PlayList()의 구문의 제 2 예는, Indexes()가 도 10을 이용하여 설명한 것일 때, 즉, Indexes()에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 Index()에 기재되지 않고, 또, 후술하는 PlayItem에도 플래그가 기재되지 않은 경우에 적용된다.

[0193] 도 17에 나타내는 PlayList()의 구문의 제 2 예에 있어서는, PlayList()에 의해 참조되는 데이터에 사운드 데이터가 포함되어 있는지 아닌지를 나타내는 1비트의 플래그인 is_MixApp가 새롭게 기재되고, reserved_for_future_use가 16비트에서 15비트로 변경되어 있는 이외에는, 도 16을 이용하여 설명한 PlayList()의 구문의 제 1 예와 동일한 구성을 가지고 있다. 또한, 이 is_MixApp를, PlayList()에 의해 참조되는 데이터에 오디오 스트림#1에 합성될 수 있는 사운드 데이터 및 오디오 스트림#2가 포함되어 있는지 아닌지를 나타내는 플래그로서 정의해도 좋다. 이 경우에는, 이 1개의 플래그를 확인하는 것만으로, 오디오 데이터의 합성이 행해지는지 아닌지, 즉, 디코드 처리의 필요성을 신속히 인식하는 것이 가능해진다.

[0194] 다음으로, PlayList()의 구문의 제 3 예를 도 18에 나타낸다. 도 18의 예에 있어서는, PlayList()에는, PlayList()에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 사운드 데이터가 포함되어 있는지 아닌지의 플래그와, 오디오 스트림#1에 합성되는 오디오 스트림#2가 포함되어 있는지 아닌지의 플래그가 기재되어 있다.

[0195] 즉, 도 18의, PlayList()의 구문의 제 3 예도, Indexes()가 도 10을 이용하여 설명한 것일 때, 즉, Indexes()에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 Index()에 기재되어 있지 않고, 또, 후술하는 PlayItem에도 플래그가 기재되어 있지 않은 경우에 적용된다.

[0196] 도 18에 나타내는 PlayList()의 구문의 제 3 예에 있어서는, PlayList()에 의해 참조되는 데이터에 오디오 스트림#2가 포함되어 있는지 아닌지를 나타내는 1비트의 플래그인 is_MixApp_1과, PlayList()에 의해 참조되는 데이터에 사운드 데이터가 포함되어 있는지 아닌지를 나타내는 1비트의 플래그인 is_MixApp_2의 2종류의 플래그가 새롭게 기재되고, reserved_for_future_use가 16비트에서 14비트로 변경되어 있는 이외에는, 도 16을 이용하여 설명한 PlayList()의 구문의 제 1 예와 동일한 구성을 가지고 있다.

[0197] 도 19는, SubPath()의 구문의 예를 나타내는 도면이다.

[0198] length는, 이 length 필드의 직후부터 SubPath()의 끝까지의 바이트 수를 나타내는 32비트의 부호 없는 정수이다. 즉, reserved_for_future_use로부터 Playlist의 끝까지의 바이트 수를 나타내는 필드이다. 이 length의 뒤에는 16비트의 reserved_for_future_use가 준비된다. reserved_for_future_use는, 장래의 확장에 대비하기 위하여 데이터의 기술내용이 미설정으로 되어 있다. SubPath_type은, Sub Path의 어플리케이션 종류를 나타내는 8비트의 필드이다. SubPath_type은, 예를 들면, Sub Path가 오디오인지, 비트맵 자막인지, 텍스트 자막인지 등의 종류를 나타내는 경우에 이용된다. 이 SubPath_type에 대하여는, 도 20을 참조하여 후술한다. SubPath_type의 다음에는 15비트의 reserved_for_future_use가 준비된다. is_repeat_SubPath는,

Sub Path의 재생방법을 지정하는 1비트의 필드이며, 메인패스의 재생 동안에 Sub Path의 재생을 반복하여 행하는지, 또는 Sub Path의 재생을 1회만 행하는지를 나타내는 것이다. 예를 들면, 메인 AV 스트림과 서브패스가 지정하는 Clip의 재생타이밍이 다른 경우(메인패스를 정지화상의 슬라이드 쇼로 하고, 서브패스의 오디오 패스를 메인패스의 BGM(배경음악)으로서 사용하는 경우 등)에 이용된다. Is_repeat_SubPath의 다음에는, 8비트의 reserved_for_future_use가 준비된다. number_of_SubPlayItems는, 1개의 Sub Path 안에 있는 SubPlayItem의 수(엔트리 수)를 나타내는 8비트의 필드이다. 예를 들면, number_of_SubPlayItems는, 도 5의 SubPath_id=0의 SubPlayItem은 1개이며, SubPath_id=1의 SubPlayItem은 2개이다. 그 후의 for문에서는, SubPlayItem의 수만큼 SubPlayItem이 참조된다.

[0199] 도 20은, SubPath_type(서브패스의 타입)의 예를 설명하는 도면이다. 즉, Sub Path의 종류는, 예를 들면, 도 20에 나타낸 바와 같이 정의되어 있다.

[0200] 도 20에 있어서, SubPath_type=0, 1은, 예약됨(reserved)으로 되어 있다. SubPath_type=2는, 브라우저블 슬라이드쇼의 오디오 프레젠테이션 패스(Audio presentation path of the browsable slideshow)로 되어 있다. 예를 들면, SubPath_type=2는, 플레이 리스트 내에 있어서, 서브패스로 참조되는 오디오 프레젠테이션 패스와, 플레이 아이템으로 참조되는 메인패스가 비동기인 것을 나타내고 있다.

[0201] SubPath_type=3은, 인터랙티브 그래픽스의 프레젠테이션 메뉴(Interactive graphics presentation menu)로 되어 있다. 예를 들면, SubPath_type=3은, 플레이 리스트 내에 있어서, 서브패스로 참조되는 인터랙티브 그래픽스의 메뉴와 플레이 아이템으로 참조되는 메인패스가 비동기인 것을 나타내고 있다.

[0202] SubPath_type=4는, 텍스트 자막의 프레젠테이션 패스(Text subtitle presentation path)로 되어 있다. 예를 들면, SubPath_type=4는, 플레이 리스트 내에 있어서, 서브패스로 참조되는 텍스트 자막의 프레젠테이션 패스와 플레이 아이템으로 참조되는 메인패스가 동기되어 있는 것을 나타내고 있다.

[0203] SubPath_type=5는, 2번째의 오디오 프레젠테이션 패스(2nd Audio Presentation path)(2번째 오디오 스트림을 참조하기 위한 패스)로 되어 있다. 구체적으로는, SubPath_type=5는, 플레이 리스트 내에 있어서, 서브패스로 참조되는 2번째의 오디오 프레젠테이션 패스와 플레이 아이템으로 참조되는 메인패스가 동기되어 있는 것을 나타내고 있다. 예를 들면, 이 서브패스로 참조되는(2번째의) 오디오 스트림은 영화에 대한 감독의 코멘트(음성)이다. 도 7에 나타낸 바와 같은 Subpath_id의 경우, 도 19의 SubPath_type은 이 SubPath_type=5가 된다.

[0204] SubPath_type=6은, 2번째의 비디오 프레젠테이션 패스(2nd Video Presentation path)(2번째 비디오 스트림을 참조하기 위한 패스)로 되어 있다. 구체적으로는, SubPath_type=6은, 플레이 리스트 내에 있어서, 서브패스로 참조되는 2번째의 비디오 프레젠테이션 패스와 플레이 아이템으로 참조되는 메인패스가 동기되어 있는 것을 나타내고 있다. 예를 들면, 이 서브패스로 참조되는(2번째의) 비디오 스트림은 영화에 대한 감독의 코멘트(동영상)이다.

[0205] SubPath_type=7 내지 255는, 예약(reserved) 되어 있다.

[0206] 도 21은, SubPlayItem(i)의 구문을 나타내는 도면이다.

[0207] length는, 이 length 필드의 직후부터 SubplayItem()의 끝까지의 바이트 수를 나타내는 16비트의 부호 없는 정수이다.

[0208] 도 21에 있어서는, SubPlayItem이 1개의 Clip을 참조하는 경우와, 복수의 Clip을 참조하는 경우로 나누어져 있다.

[0209] 먼저, SubPlayItem이 1개의 Clip을 참조하는 경우에 대하여 설명한다.

[0210] SubPlayItem에는, Clip을 지정하기 위한 Clip_Information_file_name[0]이 포함된다. 또, Clip의 코덱 방식을 지정하는 Clip_codec_identifier [0], reserved_for_future_use, 멀티 클립의 등록의 유무를 나타내는 플래그인 is_multi_Clip_entries, STC 불연속점(시스템 타임 베이스의 불연속점)에 관한 정보인 ref_to_STC_id [0]을 포함한다. is_multi_Clip_entries의 플래그가 성립되어 있는 경우, SubPlayItem이 복수의 Clip을 참조하는 경우의 구문이 참조된다. 또, Clip 내에 있는 Sub Path의 재생구간을 지정하기 위한 SubPlayItem_IN_time과 SubPlayItem_OUT_time을 포함한다. 더욱이, main path의 시간축상에서 Sub Path가 재생개시하는 시각을 지정하기 위한 sync_PlayItem_id와 sync_start PTS_of_PlayItem을 포함한다. 이 sync_PlayItem_id와 sync_start PTS_of_PlayItem은, 상술한 바와 같이, 도 6과 도 7의 경우(메인 AV 스트림과

서브패스에 의해 나타내지는 파일의 재생 타이밍이 같은 경우)에 사용되고, 메인 AV 스트림과 서브패스에 의해 나타내지는 파일의 재생 타이밍이 다른 경우(예를 들면, 정지화상에 의해 구성되는 슬라이드 쇼의 BGM과 같이, 메인패스에 의해 참조되는 정지화상과 서브패스에 의해 참조되는 오디오가 동기하지 않는 경우)에는 사용되지 않는다. 또, SubPlayItem_IN_time, SubPlayItem_OUT_time, sync_PlayItem_id, sync_start PTS_of_PlayItem 은, SubPlayItem이 참조하는 Clip에 대하여 공통으로 사용된다.

[0211] 다음으로, SubPlayItem이 복수의 Clip을 참조하는 경우(if(is_multi_Clip_entries==1b)인 경우, 즉 멀티 클립의 등록을 하고 있는 경우)에 대하여 설명한다. 구체적으로는, 도 7에 나타낸 바와 같이, SubPlayItem이 복수의 Clip을 참조하는 경우를 나타낸다.

[0212] num_of_Clip_entries는, Clip의 수를 나타내고 있고, Clip_Information_file_name[SubClip_entry_id]의 수가, Clip_Information_file_name[0]을 제외하고, Clips를 지정한다. 즉, Clip_Information_file_name[0]을 제외하고, Clip_Information_file_name[1], Clip_Information_file_name[2] 등의 Clip을 지정한다. 또, SubPlayItem은, Clip의 코덱 방식을 지정하는 Clip_codec_identifier [SubClip_entry_id], STC 불연속점(시스템 타임 베이스의 불연속점)에 관한 정보인 ref_to_STC_id[SubClip_entry_id] 및 reserved_for_future_use를 포함한다.

[0213] 또한, 복수의 Clip 사이에서, SubPlayItem_IN_time, SubPlayItem_OUT_time, sync_PlayItem_id 및 sync_start PTS_of_PlayItem은 공통으로 사용된다. 도 7의 예의 경우, SubPlayItem_IN_time, SubPlayItem_OUT_time, sync_PlayItem_id 및 sync_start PTS_of_PlayItem은, SubClip_entry_id=0과 SubClip_entry_id=1의 사이에 공통되어 사용되는 것이며, 선택된 SubClip_entry_id에 대한 텍스트 기반 자막(Text based subtitle)이 이 SubPlayItem_IN_time, SubPlayItem_OUT_time, sync_PlayItem_id 및 sync_start PTS_of_PlayItem에 근거하여 재생된다.

[0214] 여기서, SubClip_entry_id의 값은, SubPlayItem 내에 있는 Clip_Information_file_name[SubClip_entry_id]가 나타내는 차례대로 1부터 할당된다. 또, Clip_Information_file_name[0]의 SubClip_entry_id는 0이다.

[0215] 도 22는, PlayItem()의 구문의 제 1 예를 나타내는 도면이다. 도 22는 오디오 스트림#1에 합성되는 다른 음성 데이터가 이 PlayItem()에 대응하는 데이터에 포함되어 있는지 아닌지의 플래그가 기재되어 있지 않은 경우의 구문이다.

[0216] 즉, 도 22의, PlayItem()의 구문의 제 1 예는, Indexes()가 도 11 또는 도 12를 이용하여 설명한 것일 때, 즉, Indexes()에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 Index()에 기재되어 있거나, AppInfoPlayList()가 도 14를 이용하여 설명한 것일 때, 즉, PlayList에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 AppInfoPlayList()에 기재되어 있거나, 혹은, PlayList()가 도 17 또는 도 18을 이용하여 설명한 것일 때, 즉, PlayList에 의해 참조되는 데이터에 오디오 스트림#1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 기재되어 있는 경우에 적용된다.

[0217] length는, 이 length 필드의 직후부터 PlayItem()의 끝까지의 바이트 수를 나타내는 16비트의 부호 없는 정수이다. Clip_Information_file_name[0]은 PlayItem이 참조하는 Clip을 지정하기 위한 필드이다. 도 6의 예의 경우, Clip_Information_file_name[0]에 의해 메인 AV 스트림이 참조된다. 또, Clip의 코덱 방식을 지정하는 Clip_codec_identifier [0], 장래의 확장에 대비하기 위해 데이터의 기술내용이 미설정으로 되어 있는 11비트의 reserved_for_future_use, 멀티앵글 재생에 대응하고 있는지 아닌지를 나타내는 플래그인 is_multi_angle을 포함하고, 또한, connection_condition이나, STC 불연속점(시스템 타임 베이스의 불연속점)에 관한 정보인 ref_to_STC_id [0]을 포함한다. 더욱이, Clip 내의 PlayItem의 재생구간을 지정하기 위한 IN_time과 OUT_time을 포함한다. 도 6의 예의 경우, IN_time과 OUT_time에 의해 메인 Clip AV 스트림 파일의 재생범위가 나타내진다. 또, UO_mask_table(), PlayItem_random_access_mode, still_mode를 포함한다. is_multi_angle이 복수 있는 경우에 대하여는, 본 발명과 직접적으로는 관계없으므로 그 설명을 생략한다.

[0218] PlayItem() 내의 STN_table()은 대상의 PlayItem과 거기에 관련지어 재생되는 1개 이상의 Sub Path가 준비되어 있는 경우에, 사용자에 의한 음성 전환이나 자막 전환이라는 조작이 그 PlayItem이 참조하는 Clip과 이러한 1개 이상의 Sub Path가 참조하는 Clips 중에서 선택할 수 있는 구조를 제공하는 것이다. 또, STN_table()은, 2개의 오디오 스트림의 믹싱 재생을 선택할 수 있는 구조를 제공하는 것이다.

[0219] 다음으로, 도 23은 PlayItem()의 구문의 제 2 예를 나타내는 도면이다. 도 23의 예에 있어서는, 이

PlayItem()에 대응하는 데이터에 오디오 스트림 #2는 포함되지 않고, PlayItem()에는 PlayItem()에 대응하는 데이터에 오디오 스트림 #1에 합성되는 사운드 데이터가 포함되어 있는지 아닌지의 플래그가 기재되어 있다.

[0220] 즉, 도 23의 PlayItem()의 구문의 제 2 예는, Indexes()가 도 10을 이용하여 설명한 것일 때, 즉, Indexes()에 의해 참조되는 데이터에 오디오 스트림 #1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 Index()에 기재되어 있지 않고, AppInfoPlayList()가 도 15를 이용하여 설명한 것일 때, 즉, PlayList에 의해 참조되는 데이터에 오디오 스트림 #1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 AppInfoPlayList()에 기재되어 있지 않고, PlayList()가 도 16을 이용하여 설명한 것일 때, 즉, PlayList에 의해 참조되는 데이터에 오디오 스트림 #1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 기재되어 있지 않은 경우에 적용된다.

[0221] 도 23에 나타내는 PlayItem()의 구문의 제 2 예에 있어서는, PlayItem()에 의해 참조되는 데이터에 사운드 데이터가 포함되어 있는지 아닌지를 나타내는 1비트의 플래그인 is_MixApp가 새롭게 기재되고, reserved_for_future_use가 11비트에서 10비트로 변경되어 있는 이외에는, 도 22를 이용하여 설명한 PlayItem()의 구문의 제 1 예와 동일한 구성을 가지고 있다. 또한, 이 is_MixApp를, PlayItem()에 의해 참조되는 데이터에 오디오 스트림 #1에 합성될 수 있는 사운드 데이터 및 오디오 스트림 #2가 포함되어 있는지 아닌지를 나타내는 플래그로서 정의해도 좋다. 이 경우에는, 이 1개의 플래그를 확인하는 것만으로 오디오 데이터의 합성이 행해지는지 아닌지, 즉, 디코드 처리의 필요성을 신속히 인식하는 것이 가능해진다.

[0222] 다음으로, PlayItem()의 구문의 제 3 예를 도 24에 나타낸다. 도 24의 예에 있어서는, PlayItem()에는, PlayItem()에 대응하는 데이터에 오디오 스트림 #1에 합성되는 사운드 데이터가 포함되어 있는지 아닌지의 플래그와, 오디오 스트림 #1에 합성되는 오디오 스트림 #2가 포함되어 있는지 아닌지의 플래그가 기재되어 있다.

[0223] 즉, 도 24의 PlayItem()의 구문의 제 3 예도, Indexes()가 도 10을 이용하여 설명한 것일 때, 즉, Indexes()에 의해 참조되는 데이터에 오디오 스트림 #1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 Index()에 기재되어 있지 않고, AppInfoPlayList()가 도 15를 이용하여 설명한 것일 때, 즉, PlayList에 의해 참조되는 데이터에 오디오 스트림 #1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 AppInfoPlayList()에 기재되어 있지 않고, PlayList()가 도 16을 이용하여 설명한 것일 때, 즉, PlayList에 의해 참조되는 데이터에 오디오 스트림 #1에 합성되는 다른 음성 데이터가 포함되어 있는지 아닌지의 플래그가 기재되어 있지 않은 경우에 적용된다.

[0224] 도 24에 나타내는 PlayItem()의 구문의 제 3 예에 있어서는, PlayItem()에 대응하는 데이터에 오디오 스트림 #2가 포함되어 있는지 아닌지를 나타내는 1비트의 플래그인 is_MixApp_1과, PlayItem()에 대응하는 데이터에 사운드 데이터가 포함되어 있는지 아닌지를 나타내는 1비트의 플래그인 is_MixApp_2의 2 종류의 플래그가 새롭게 기재되고, reserved_for_future_use가 11비트에서 9비트로 변경되어 있는 이외에는, 도 22를 이용하여 설명한 PlayItem()의 구문의 제 1 예와 동일한 구성을 가지고 있다.

[0225] 도 25는, STN_table()의 구문을 나타내는 도면이다. STN_table()은, PlayItem의 속성으로서 설정되어 있다.

[0226] length는, 이 length 필드의 직후부터 STN_table()의 끝까지의 바이트 수를 나타내는 16비트의 부호 없는 정수이다. length의 다음에는 16비트의 reserved_for_future_use가 준비된다. number_of_video_stream_entries는 STN_table() 내에서 엔트리 되는(등록되는) video_stream_id가 주어지는 스트림 수를 나타낸다. video_stream_id는 비디오 스트림을 식별하기 위한 정보이며, video_stream_number는 비디오 전환에 사용되고, 사용자에 보이는 비디오 스트림 번호이다.

[0227] number_of_audio_stream_entries는 STN_table() 내에서 엔트리 되는 audio_stream_id가 주어지는 1번째의 오디오 스트림의 스트림 수를 나타낸다. audio_stream_id는 오디오 스트림을 식별하기 위한 정보이며, audio_stream_number는 음성 전환에 사용되는 사용자에 보이는 오디오 스트림 번호이다. number_of_audio_stream2_entries는 STN_table() 내에서 엔트리 되는 audio_stream_id2가 주어지는 2번째의 오디오 스트림의 스트림 수를 나타낸다. audio_stream_id2는 오디오 스트림을 식별하기 위한 정보이며, audio_stream_number는 음성 전환에 사용되는 사용자에 보이는 오디오 스트림 번호이다. 구체적으로는, STN_table()에서 엔트리 되는 number_of_audio_stream_entries의 오디오 스트림은 후술하는 도 28의 재생장치(20)의 제 1 오디오 디코더(75-1)로 디코드되는 오디오 스트림이며, STN_table()에서 엔트리 되는 number_of_audio_stream2_entries의 오디오 스트림은 후술하는 도 28의 재생장치(20)의 제 2 오디오 디코더(75-2)로 디코드되는 오디오 스트림이다. 이와 같이, 도 25의 STN_table()에서는, 2개의 오디오 디코더의 각

각에 디코드시키는 오디오 스트림을 엔트리 할 수 있다.

[0228] 또한, 이하에 있어서, 도 28의 재생장치(20)의 제 1 오디오 디코더(75-1)로 디코드되는 number_of_audio_stream_entries의 오디오 스트림을 오디오 스트림 #1이라 칭하고, 도 28의 재생장치(20)의 제 2 오디오 디코더(75-2)로 디코드되는 number_of_audio_stream2_entries의 오디오 스트림을 오디오 스트림 #2라 칭한다. 또, 오디오 스트림 #1은 오디오 스트림 #2보다 우선되는 오디오 스트림인 것으로 한다.

[0229] number_of_PG_txtST_stream_entries는 STN_table() 내에서 엔트리 되는 PG_txtST_stream_id가 주어지는 스트림 수를 나타낸다. 이 안에서는, DVD의 서브 텍스트와 같은 비트맵 자막을 런 령스(run length) 부호화한 스트림(PG, Presentation Graphics stream)과 텍스트 자막 파일(txtST)이 엔트리 된다. PG_txtST_stream_id는, 자막 스트림을 식별하기 위한 정보이며, PG_txtST_stream_number는 자막 전환에 사용되는 사용자에 보이는 자막 스트림 번호(텍스트 서브 타이틀 스트림의 번호)이다.

[0230] number_of_IG_stream_entries는, STN_table() 내에서 엔트리 되는 IG_stream_id 가 주어지는 스트림 수를 나타낸다. 이 안에서는, 인터랙티브 그래픽스 스트림이 엔트리 된다. IG_stream_id는 인터랙티브 그래픽스 스트림을 식별하기 위한 정보이며, IG_stream_number는 그래픽스 전환에 사용되는 사용자에 보이는 그래픽스 스트림 번호이다.

[0231] 여기서, stream_entry()의 구문에 대하여 도 26을 참조하여 설명한다.

[0232] length는 이 length 필드의 직후부터 stream_entry()의 끝까지의 바이트 수를 나타내는 8비트의 부호 없는 정수이다. type은 상술한 스트림 번호가 주어지는 스트림을 한가지로 특정하기 위해 필요한 정보의 종류를 나타내는 8비트의 필드이다.

[0233] type=1에서는, PlayItem에 의해 참조되는 Clip(Main Clip) 안에 다중화되어 있는 복수의 엘레멘터리 스트림 중에서 1개의 엘레멘터리 스트림을 특정하기 위해, 16비트의 패킷 ID(PID)가 지정된다. ref_to_stream_PID_of_mainClip이, 이 PID를 나타내고 있다. 즉, type=1에서는 메인 Clip AV 스트림 파일 안의 PID를 지정하는 것만으로 스트림이 결정된다.

[0234] type=2에서는 SubPath가 한 번에 복수 개의 Clips를 참조하고, 각각의 Clip이 복수의 엘레멘터리 스트림을 다중화하는 경우에, SubPath에 의해 참조되는 1개의 Clip(SubClip)의 복수의 엘레멘터리 스트림 중에서 1개의 엘레멘터리 스트림을 특정하기 위해, 그 SubPath의 SubPath_id, Clip id, 및 패킷 ID(PID)가 지정된다. ref_to_SubPath_id가 이 SubPath_id를 나타내고, ref_to_SubClip_entry_id가 이 Clip id를 나타내고, ref_to_stream_PID_of_SubClip가 이 PID를 나타내고 있다. SubPlayItem 내에서 복수의 Clip이 참조되고, 또한 이 Clip에 복수의 엘레멘터리 스트림이 참조되고 있는 경우에 이용된다.

[0235] 이와 같이, type(type=1과 type=2의 2개의 type)을 사용하는 것으로, PlayItem과 거기에 관련지어 재생되는 1개 이상의 SubPath가 준비되어 있는 경우에, 이 PlayItem이 참조하는 Clip과 1개 이상의 SubPath가 참조하는 Clip 중에서 1개의 엘레멘터리 스트림을 특정할 수 있다. 또한, type=1은 Main Path가 참조하는 Clip(메인 Clip)을 나타내고 있고, type=2는 Sub Path가 참조하는 Clip(서브 Clip)을 나타내고 있다.

[0236] 도 25의 STN_table()의 설명으로 돌아가서, 비디오 스트림 ID (video_stream_id)의 for 루프 내에서, 차례로 stream_entry() 마다 특정되는 1개의 비디오 엘레멘터리 스트림에 0부터 video_stream_id가 주어진다. 또한, 비디오 스트림 ID(video_stream_id) 대신에, 비디오 스트림 번호(video_stream_number)를 이용하도록 해도 좋다. 이 경우, video_stream_number는, 0이 아니라 1부터 주어진다. 즉, video_stream_id의 값에 1을 가산한 것이 video_stream_number이다. 비디오 스트림 번호는, 비디오 전환에 사용되고, 사용자에 보이는 비디오 스트림 번호이므로 1부터 정의된다.

[0237] 마찬가지로, 오디오 스트림 ID(audio_stream_id)의 for 루프 내에서, 차례로 stream_entry() 마다 특정되는 1개의 오디오 엘레멘터리 스트림에 0부터 audio_stream_id가 주어진다. 또한, 비디오 스트림의 경우와 마찬가지로, 오디오 스트림 ID(audio_stream_id) 대신에, 오디오 스트림 번호(audio_stream_number)를 이용하도록 해도 좋다. 이 경우, audio_stream_number는 0이 아니라 1부터 주어진다. 즉, audio_stream_id의 값에 1을 가산한 것이 audio_stream_number이다. 오디오 스트림 번호는 음성 전환에 사용되고, 사용자에 보이는 오디오 스트림 번호이므로 1부터 정의된다.

[0238] 마찬가지로, 오디오 스트림 ID2(audio_stream_id2)의 for 루프 내에서, 차례로 stream_entry() 마다 특정되는 1개의 오디오 엘레멘터리 스트림에 0부터 audio_stream_id2가 주어진다. 또한, 비디오 스트림의 경우와 마찬

가지로, 오디오 스트림 ID2(audio_stream_id2) 대신에, 오디오 스트림 번호2(audio_stream_number2)를 이용하도록 해도 좋다. 이 경우, audio_stream_number2는 0이 아니라 1부터 주어진다. 즉, audio_stream_id2의 값에 1을 가산한 것이 audio_stream_number2이다. 오디오 스트림 번호2는 음성 전환에 사용되고, 사용자에 보이는 오디오 스트림 번호2이므로, 1부터 정의된다.

[0239] 즉, 도 25의 STN_table()에 있어서는, number_of_audio_stream_entries(오디오 스트림#1)의 오디오 스트림과 number_of_audio_stream2_entries (오디오 스트림#2)의 오디오 스트림이 정의된다. 환언하면, STN_table()을 이용하여 오디오 스트림#1과 오디오 스트림#2를 엔트리 할 수 있으므로, 사용자는, 동기하여 재생하는 오디오 스트림을 2개 선택할 수 있다.

[0240] 마찬가지로, 자막 스트림 ID(PG_txtST_stream_id)의 for 루프 내에서, 차례로 stream_entry() 마다 특정되는 1개의 비트맵 자막 엘레멘터리 스트림 또는 텍스트 자막에, 0부터 PG_txtST_stream_id가 주어진다. 또한, 비디오 스트림의 경우와 동일하게, 자막 스트림 ID(PG_txtST_stream_id) 대신에, 자막 스트림 번호(PG_txtST_stream_number)를 이용하도록 해도 좋다. 이 경우, PG_txtST_stream_number는 0이 아니라 1부터 주어진다. 즉, PG_txtST_stream_id의 값에 1을 가산한 것이 PG_txtST_stream_number이다. 자막 스트림 번호는 자막전환에 사용되는 사용자에 보이는 자막 스트림 번호(텍스트 서브 스트림의 번호)이므로, 1부터 정의된다.

[0241] 마찬가지로, 그래픽스 스트림 ID(IG_stream_id)의 for 루프 내에서, 차례로 stream_entry() 마다 특정되는 1개의 인터랙티브 그래픽스 엘레멘터리 스트림에 0부터 IG_stream_id가 주어진다. 또한, 비디오 스트림의 경우와 마찬가지로, 그래픽스 스트림 ID(IG_stream_id) 대신에, 그래픽스 스트림 번호(IG_stream_number)를 이용하도록 해도 좋다. 이 경우, IG_stream_number는 0이 아니라 1부터 주어진다. 즉, IG_stream_id의 값에 1을 가산한 것이 IG_stream_number이다. 그래픽스 스트림 번호는 그래픽스 전환에 사용되는 사용자에 보이는 그래픽스 스트림 번호이므로, 1부터 정의된다.

[0242] 다음으로, 도 25의 STN_table()의 stream_attribute()에 대하여 설명한다.

[0243] reserved_for_future_use의 뒤의 for문에서는, video stream의 분량만큼 video stream이 참조되고, 이 데이터의 제공자(예를 들면, 기록매체(21)의 제조자 등)가 오디오 스트림에 대하여 Main Path 및 Sub Path를 설정한 분량만큼 audio stream이 참조되고, PG textST stream의 분량만큼 PG textST stream이 참조되고, IG stream의 분량만큼 IG stream이 참조된다.

[0244] 비디오 스트림 ID(video_stream_id)의 for 루프 내의 stream_attribute()는 stream_entry() 마다 특정되는 1개의 비디오 엘레멘터리 스트림의 스트림 속성정보를 부여한다. 즉, 이 stream_attribute()에는 stream_entry() 마다 특정되는 1개의 비디오 엘레멘터리 스트림의 스트림 속성정보가 기술되어 있다.

[0245] 마찬가지로, 오디오 스트림 ID(audio_stream_id)의 for 루프 내의 stream_attribute()는 stream_entry() 마다 특정되는 1개의 오디오 엘레멘터리 스트림의 스트림 속성정보를 부여한다. 즉, 이 stream_attribute()에는 stream_entry()마다 특정되는 1개의 오디오 엘레멘터리 스트림의 스트림 속성정보가 기술되어 있다. 예를 들면, 도 26의 stream_entry()의 type=1 또는 type=2로 특정되는 오디오 엘레멘터리 스트림은 1개이므로, stream_attribute()는 그 1개의 오디오 엘레멘터리 스트림의 스트림 속성정보를 부여한다.

[0246] 마찬가지로, 오디오 스트림 ID2(audio_stream_id2)의 for 루프 내의 stream_attribute()는 stream_entry() 마다 특정되는 1개의 오디오 엘레멘터리 스트림의 스트림 속성정보를 부여한다. 즉, 이 stream_attribute()에는 stream_entry() 마다 특정되는 1개의 오디오 엘레멘터리 스트림의 스트림 속성정보가 기술되어 있다. 예를 들면, 도 26의 stream_entry()의 type=1 또는 type=2로 특정되는 오디오 엘레멘터리 스트림은 1개이므로, stream_attribute()는 그 1개의 오디오 엘레멘터리 스트림의 스트림 속성정보를 부여한다.

[0247] 마찬가지로, 자막 스트림 ID(PG_txtST_stream_id)의 for 루프 내의 stream_attribute()는 stream_entry() 마다 특정되는 1개의 비트맵 자막 엘레멘터리 스트림 또는 텍스트 자막 엘레멘터리 스트림의 스트림 속성정보를 부여한다. 즉, 이 stream_attribute()에는 stream_entry() 마다 특정되는 1개의 비트맵 자막 엘레멘터리 스트림의 스트림 속성정보가 기술되어 있다.

[0248] 마찬가지로, 그래픽스 스트림 ID(IG_stream_id)의 for 루프 내의 stream_attribute()는 stream_entry() 마다 특정되는 1개의 인터랙티브 그래픽스 엘레멘터리 스트림의 스트림 속성정보를 부여한다. 즉, 이 stream_attribute()에는 stream_entry() 마다 특정되는 1개의 인터랙티브 그래픽스 엘레멘터리 스트림의 스트림 속성정보를 부여한다.

속성정보가 기술되어 있다.

[0249] 여기서, `stream_attribute()`의 구문에 대하여 도 27을 참조하여 설명한다.

[0250] `length`는 이 `length` 필드의 직후부터 `stream_attribute()`의 끝까지의 바이트 수를 나타내는 16비트의 부호 없는 정수이다.

[0251] `stream_coding_type`은 도 28에 나타낸 바와 같이, 엘레멘터리 스트림의 부호화 타입을 나타낸다. 엘레멘터리 스트림의 부호화 타입으로서는, MPEG-2 비디오 스트림(MPEG-2 video stream), HDMV LPCM 오디오(HDMV LPCM audio), 돌비 AC-3 오디오(Dolby AC-3 audio), dts 오디오(dts audio), 프리젠테이션 그래픽스 스트림(Presentation graphics stream), 인터랙티브 그래픽스 스트림(Interactive graphics stream) 및 텍스트 서브 타이틀 스트림(Text subtitle stream)이 기술된다.

[0252] `video_format`은 도 29에 나타낸 바와 같이, 비디오 엘레멘터리 스트림의 비디오 포맷을 나타낸다. 비디오 엘레멘터리 스트림의 비디오 포맷으로서는, 480i, 576i, 480p, 1080i, 720p, 및 1080p가 기술된다.

[0253] `frame_rate`는 도 30에 나타낸 바와 같이, 비디오 엘레멘터리 스트림의 프레임율(frame rate)을 나타낸다. 비디오 엘레멘터리 스트림의 프레임율(frame rate)로서는, 24000 / 1001, 24, 25, 30000 / 1001, 50, 및 60000 / 1001이 기술된다.

[0254] `aspect_ratio`는 도 31에 나타낸 바와 같이, 비디오 엘레멘터리 스트림의 화면비(aspect ratio) 정보를 나타낸다. 비디오 엘레멘터리 스트림의 화면비 정보로서는 4 : 3(4 : 3 display aspect ratio) 및 16 : 9(16 : 9 display aspect ratio)가 기술된다.

[0255] `audio_presentation_type`은 도 32에 나타낸 바와 같이, 오디오 엘레멘터리 스트림의 프레젠테이션 타입 정보를 나타낸다. 오디오 엘레멘터리 스트림의 프레젠테이션 타입 정보로서는, 단일 모노 채널(single mono channel), 이중 모노 채널(Dual mono channel), 스테레오(stereo)(2-channel) 및 멀티 채널(multi-channel)이 기술된다.

[0256] `sampling_frequency`는 도 33에 나타낸 바와 같이, 오디오 엘레멘터리 스트림의 샘플링 주파수를 나타낸다. 오디오 엘레멘터리 스트림의 샘플링 주파수로서는 48kHz 및 96kHz가 기술된다.

[0257] `audio_language_code`는 오디오 엘레멘터리 스트림의 언어 코드(일본어, 한국어, 중국어 등)를 나타낸다.

[0258] `PG_language_code`는 비트맵 자막 엘레멘터리 스트림의 언어 코드(일본어, 한국어, 중국어 등)를 나타낸다.

[0259] `IG_language_code`는 인터랙티브 그래픽스 엘레멘터리 스트림의 언어 코드(일본어, 한국어, 중국어 등)를 나타낸다.

[0260] `textST_language_code`는 텍스트 자막 엘레멘터리 스트림의 언어 코드(일본어, 한국어, 중국어 등)를 나타낸다.

[0261] 삭제

[0262] `character_code`는, 도 34에 나타낸 바와 같이, 텍스트 자막 엘레멘터리 스트림(elementary stream)의 캐릭터 코드를 나타낸다. 텍스트 자막 엘레멘터리 스트림의 캐릭터 코드로서는 유니코드(Unicode) V1.1(ISO 10646-1), 쉬프트 JIS(Shift JIS)(Japanese), KSC 5601-1987(KSC 5601-1987 including KSC 5653 for Roman character(Korean)), GB 18030-2000(Chinese), GB 2312(Chinese) 및 BIG5(Chinese)가 기술된다.

[0263] 이하에, 도 27의 `stream_attribute()`의 구문에 대하여, 도 27과 도 28 내지 도 34를 이용하여 구체적으로 설명한다.

[0264] 엘레멘터리 스트림의 부호화 타입(도 27의 `stream_coding_type`)이 MPEG-2 비디오 스트림(video stream)(도 28)인 경우, `stream_attribute()`에는 그 엘레멘터리 스트림의 비디오 포맷(도 29), 프레임율(frame rate)(도 30) 및 화면비 정보(도 31)가 포함된다.

[0265] 엘레멘터리 스트림의 부호화 타입(도 27의 `stream_coding_type`)이 HDMV LPCM audio, 돌비(Dolby) AC-3 audio, 또는 dts audio(도 28)인 경우, `stream_attribute()`에는 그 오디오 엘레멘터리 스트림의 프레젠테이션 타입 정보(도 32), 샘플링 주파수(도 33) 및 언어코드가 포함된다.

[0266] 엘레멘터리 스트림의 부호화 타입(도 27의 `stream_coding_type`)이 프리젠테이션 그래픽스 스트림(presentation

graphics stream)(도 28)인 경우, stream_attribute()에는, 그 비트맵 자막 엘레멘터리 스트림의 언어코드가 포함된다.

[0267] 엘레멘터리 스트림의 부호화 타입(도 27의 stream_coding_type)이 인터랙티브 그래픽스 스트림(Interactive graphics stream)(도 28)인 경우, stream_attribute()에는 그 인터랙티브 그래픽스 엘레멘터리 스트림의 언어코드가 포함된다.

[0268] 엘레멘터리 스트림의 부호화 타입(도 27의 stream_coding_type)이 텍스트 서브타이틀 스트림(Text subtitle stream)(도 28)인 경우, stream_attribute()에는 그 텍스트 자막 엘레멘터리 스트림의 캐릭터코드(도 34), 언어코드가 포함된다.

[0269] 또한, 이러한 속성정보는 이것으로 한정되지 않는다.

[0270] 이와 같이, PlayItem과 그것에 관련지어 재생되는 1개 이상의 서브 패스(SubPath)가 준비되어 있는 경우에, 이 PlayItem이 참조하는 Clip과 1개 이상의 서브 패스가 참조하는 Clip 중에서, stream_entry()에 의해서 특정된 1개의 엘레멘터리 스트림의 속성정보를 stream_attribute()에 의해 알 수 있다.

[0271] 재생장치는 이 속성정보(stream_attribute())를 조사하는 것으로, 그 엘레멘터리 스트림을 자기 자신이 재생하는 기능을 가지고 있는지 아닌지를 조사할 수 있다. 또, 재생장치는 이 속성정보를 조사하는 것에 의해, 재생장치의 언어설정의 초기정보에 대응한 엘레멘터리 스트림을 선택할 수 있다.

[0272] 예를 들면, 재생장치가 비트맵 자막 엘레멘터리 스트림의 재생기능만을 가지고, 텍스트 자막 엘레멘터리 스트림의 재생기능을 가지지 않은 경우를 상정한다. 이 재생장치에 있어서, 사용자가 언어전환을 지시한 경우, 재생장치는 자막 스트림 ID(PG_txtST_stream_id)의 for루프 내에서, 비트맵 자막 엘레멘터리 스트림만을 차례로 선택하고, 재생한다.

[0273] 또, 예를 들면, 재생장치의 언어설정의 초기정보가 일본어인 경우를 상정한다. 이 재생장치에 있어서, 사용자가 음성전환을 지시한 경우, 재생장치는 오디오 스트림 ID(Audio stream id)의 for루프 내에서, 언어코드가 일본어인 오디오 엘레멘터리 스트림만을 차례로 선택하고, 재생한다.

[0274] 또한, 예를 들면, 메인 패스에 의해 참조되는 비디오 스트림과 오디오 스트림으로 이루어지는 AV 스트림(영화)을 재생하는 경우, 재생장치에 대하여 사용자가 음성의 전환을 지령하고, 오디오 스트림 #1(통상의 영화로 출력되는 음성)과 오디오 스트림 #2(감독이나 출연자에 의한 코멘트)를 재생하는 음성으로서 지정(선택)한 경우, 재생장치는 오디오 스트림 #1과 오디오 스트림 #2를 믹싱(중첩)하고, 비디오 스트림과 함께 재생한다.

[0275] 또한, 오디오 스트림 #1과 오디오 스트림 #2는 도 25 및 도 26의 STN_table()를 참조하여도 알 수 있는 바와 같이, 양쪽 모두 메인 패스에 의해 참조되는 클립에 포함되는 오디오 스트림으로 해도 좋다. 또, 오디오 스트림 #1과 오디오 스트림 #2 중, 한쪽을 메인 패스에 의해 참조되는 클립에 포함되는 오디오 스트림으로 하고, 다른 쪽을 서브 패스에 의해 참조되는 클립에 포함되는 오디오 스트림으로 해도 좋다. 이와 같이, 메인 패스에 의해 참조되는 메인 AV 스트림에 중첩된 복수의 오디오 스트림을 2개 선택하고, 믹싱하여 재생하는 것도 가능하다.

[0276] 이와 같이 하여, PlayItem() 안의 STN_table()는 이 PlayItem과 그것에 관련지어서 재생되는 1개 이상의 서브 패스가 준비되어 있는 경우에, 사용자에 의한 음성전환이나 자막전환이라는 조작이, 이 PlayItem이 참조하는 클립과 1개 이상의 서브 패스가 참조하는 클립 중에서 선택할 수 있는 구조를 제공하도록 했으므로, 메인 AV 스트림이 기록되어 있는, 재생하는 AV 스트림과는 다른 스트림이나 데이터 파일에 대하여도, 인터랙티브한 조작을 행할 수 있다.

[0277] 또, 1개의 PlayList 안에 서브 패스를 복수 사용하고, 각각의 서브 패스가 각각 서브 플레이 아이템(SubPlayItem)을 참조하는 구성으로 했으므로, 확장성이 높고, 또, 자유도가 높은 AV 스트림을 실현할 수 있다.

즉, 나중에 SubPlayItem을 추가할 수 있는 구성으로 할 수 있다. 예를 들면, 메인 패스가 참조하는 클립 AV 스트림 파일과 이것에 대응시킬 수 있는 PlayList가 있고, 이 PlayList를 새로운 서브 패스를 추가한 PlayList로 고쳐 쓸 수 있는 경우, 새로운 PlayList에 근거하여, 메인 패스가 참조하는 클립 AV 스트림 파일과 함께, 메인 패스가 참조하는 클립 AV 스트림 파일과는 다른 클립 AV 스트림 파일을 참조하여, 재생을 행할 수 있다. 이와 같이, 확장성을 가지는 구성으로 할 수 있다.

[0278] 또한, PlayItem() 안의 STN_table()은 후술하는 도 41의 재생장치(20-2)의 제 1 오디오 디코더(75-1)로 복호화되는 오디오 스트림 #1과, 제 2 오디오 디코더(75-2)로 복호화되는 오디오 스트림 #2를 믹싱(합성)하여 재생할

수 있는 구조를 제공한다. 예를 들면, PlayItem()과 그것에 관련지어 재생되는 1개 이상의 서브 패스가 준비되어 있는 경우에, PlayItem이 참조하는 클립의 오디오 스트림을 오디오 스트림#1으로 하고, 서브 패스가 참조하는 클립의 오디오 스트림을 오디오 스트림#2으로 하고, 이들을 믹싱하여 재생할 수 있는 구조를 제공하도록 했다. 또, 예를 들면, PlayItem이 참조하는 클립(메인 Clip)에 포함되는 2개의 오디오 스트림을, 각각 오디오 스트림#1과 오디오 스트림#2로 하고, 이들을 믹싱하여 재생할 수 있는 구조를 제공하도록 했다. 이것이에 의해, 메인 AV 스트림이 기록되어 있는, 재생하는 메인 오디오 스트림과는 다른 오디오 스트림(예를 들면, 감독의 코멘트의 스트림)의 중첩재생을 행할 수 있다. 또, 메인 AV 스트림에 중첩되어 있는 2개의 오디오 스트림 #1과 오디오 스트림 #2를 중첩(믹싱)하여 재생할 수 있다.

[0279] 구체적인 예를, 도 35를 참조해 설명한다. 도 35는 사용자에게 제공하는 음성 신호와 자막 신호의 관계를 나타내는 스트림 넘버 테이블의 예를 나타내는 도면이다.

[0280] 도 35에 있어서는, 음성 번호를 A_SN(Audio Stream Number), A_SN2라고 칭하고, 자막 번호를 S_SN(SubPicture Stream Number)로 칭한다. 도 35에 있어서는, 플레이 리스트의 메인 패스를 구성하는 플레이 아이템의 STN_table()로 엔트리 되는 오디오 스트림#1(audio_stream_id로 엔트리 되는 오디오 스트림)의 각각에 A_SN이 주어지고, 플레이 리스트의 메인 패스를 구성하는 플레이 아이템의 STN_table()로 엔트리 되는 오디오 스트림#2(audio_stream_id2로 엔트리 되는 오디오 스트림)의 각각에 A_SN2가 주어진다.

[0281] 구체적으로는, A_SN = 1에 오디오(2)가 주어지고, A_SN = 2에 오디오(1)가 주어지고, A_SN = 3에 오디오(3)가 주어져 있다. 또, A_SN2 = 1에 오디오(4)가 주어지고, A_SN2 = 2에 오디오(5)가 주어져 있다. 사용자는 A_SN이 주어져 있는 오디오 스트림 중에서, 재생하는 오디오 스트림#1을 선택하고, 선택한 오디오 스트림#1과 믹싱하는 오디오 스트림#2를 A_SN2가 주어져 있는 오디오 스트림 중에서 선택한다. 예를 들면, 사용자는 A_SN = 2의 오디오(1)와 A_SN2 = 2의 오디오(5)를 재생하는 오디오 스트림으로서 선택한다.

[0282] 구체적인 예로서는, A_SN = 1의 오디오(2)가 선택되어 있는 경우에, 사용자가 오디오의 전환을 지령한 경우, 오디오는 A_SN = 2의 오디오(1)로 전환할 수 있고, 사용자가 또 오디오의 전환을 지령한 경우, 오디오는 A_SN = 3의 오디오(3)로 전환할 수 있다. 또, 사용자가 또 오디오의 전환을 지령한 경우, 오디오는 A_SN = 1의 오디오(2)로 전환할 수 있다. 또, 예를 들면, A_SN2 = 1의 오디오(4)가 선택되고 있는 경우에, 사용자가 오디오의 전환을 지령한 경우, 오디오는 A_SN2 = 2의 오디오(5)로 전환할 수 있고, 사용자가 또 오디오의 전환을 지령한 경우, 오디오는 A_SN2 = 1의 오디오(4)로 전환할 수 있다. 이와 같이, 오디오 스트림#1을 선택하기 위한 A_SN와 오디오 스트림#2을 선택하기 위한 A_SN2는 음성의 전환을 독립시키고 있다. 즉, 사용자는 A_SN = 1 내지 A_SN = 3으로부터 1개의 오디오 스트림을 선택하고, A_SN2 = 1 및 A_SN2 = 2에서 1개의 오디오 스트림을 선택하게 된다.

[0283] 또한, 여기에서는, A_SN, A_SN2의 번호가 작을수록, 사용자에게 제공되는 음성신호의 우선도가 높다. 또, A_SN로 주어지는 스트림 쪽이 A_SN2로 주어지는 스트림보다 우선도가 높다. 즉, A_SN = 1은 디폴트로 재생되는 오디오 스트림이다.

[0284] 구체적으로는, 재생장치의 언어설정의 초기정보에 근거하여 재생되는 음성은 A_SN = 1인 오디오(2)(도 35)에 대응하고 있고, 음성이 전환된 후, 재생되는 음성은 A_SN = 2인 오디오(1)(도 35)에 대응하고 있다.

[0285] 이러한 스트림 넘버 테이블을 제공하기 위해, PlayList()로 참조되는 PlayItem() 안의 STN_table()(도 25)에 있어서, 최초로, 오디오 스트림#1로 엔트리하기 위한 audio_stream_id = 0(A_SN = 1)으로 오디오(2)를 주고, audio_stream_id = 1(A_SN = 2)로 오디오(1)를 주고, audio_stream_id = 2(A_SN = 3)로 오디오(3)를 준다.

다음에, STN_table()(도 25)에 있어서, 오디오 스트림#2으로 엔트리하기 위한 audio_stream_id2 = 0(A_SN2 = 1)에서 오디오(4)를 주고, audio_stream_id2 = 1(A_SN2 = 2)에서 오디오(5)를 준다.

[0286] 즉, 재생하는 2개의 오디오 스트림(오디오 스트림#1과 오디오 스트림#2)을 별개로 정의하는 것으로, 사용자는 재생하는 2개의 오디오 스트림을 정의된 것 중에서 임의로 선택할 수 있다. 환연하면, 사용자는 재생하는 2개의 오디오 스트림의 선택을 자유롭게(오디오 스트림#1과 오디오 스트림#2로 정의되어 있는 것 중에서 자유롭게) 실시할 수 있으므로, 조합의 자유도가 높은 선택을 행할 수 있다. 예를 들면, 사용자는, 오디오(2)+오디오(4)의 조합(A_SN = 1과 A_SN2 = 1의 조합)이나, 오디오(2)+오디오(5)의 조합(A_SN = 1과 A_SN2 = 2의 조합)을 선택할 수 있다.

[0287] 이와 같이, PlayItem() 안의 STN_table()(도 25)의 stream_entry()(도 26)에서, 2개의 오디오 스트림을 엔트리하는 것이 가능한 구성으로 했으므로, 2개의 오디오 스트림을 믹싱하여 재생할 수 있다. 즉, 복수의 종류로

이루어지는 스트림으로부터, 같은 종류(이 예의 경우, 오디오 스트림)의 2개의 스트림을 믹싱(중첩, 또는 합성)하고, 동시에 재생할 수 있다. 또, 사용자는 소망하는 2개의 같은 종류의 스트림의 믹싱 재생을 지령할 수 있다.

[0288] 또한, 이상의 예에서는, 도 41의 재생장치(20-2)의 제 1 오디오 디코더(75-1)로 복호화되는 오디오 스트림 #1과 제 2 오디오 디코더(75-2)로 복호화되는 오디오 스트림 #2를 사용자가 별개로 선택 가능한 구성으로 했으나, 오디오 스트림 #1과 오디오 스트림 #2의 조합을 정의하고 믹싱 재생하는 경우에는, 이 조합 중에서 사용자에게 선택시키는 구성으로 해도 좋다.

[0289] 다음에, 도 36을 참조하여, AUXDATA 디렉토리의 sound.bdmv의 구문에 대하여 설명한다.

[0290] sound.bdmv는 인터랙티브 그래픽스를 이용한 스트림에 대한 1개 또는 복수의 효과음을 포함한 파일이다.

[0291] SoundData_start_address는 SoundData()의 선두주소를 나타내는 32비트의 필드이며, SoundIndex()는 실제의 효과음의 데이터인 SoundData()의 효과음의 속성(예를 들면, 채널수·주파수등)을 나타내는 정보이다.

[0292] 실제의 효과음의 데이터인 SoundData()는 비압축의 오디오 데이터이며, 예를 들면, 사운드 스트림과는 별도로 효과음이 준비되고, 소정의 위치에서 출력되도록 되어 있는 컨텐츠나, 사용자의 조작 입력을 기본으로 재생하는 데이터를 변경할 수 있도록 되어 있는 등, 사용자의 조작 입력을 받는 것을 전제로 하고 있는 컨텐츠나, 인터랙티브인 컨텐츠 등에 들 수 있는 클릭음 등으로 하여 준비되는 데이터이다. 환연하면, 컨텐츠, 또는, 플레이리스트 또는 플레이 아이템으로 지정되는 컨텐츠의 일부의 사양에 따라서는 SoundData()를 가지지 않은 것도 있다. SoundData()는 예를 들면, 효과음의 재생이 커맨드에 의해서 지령된 경우나 사용자의 조작 입력을 받은 경우, 오디오 스트림으로 합성되어 출력되도록 되어 있다.

[0293] 다음에, 상술한 구성을 가지는 데이터를 재생 출력하는 재생장치(20)의 구성예와 처리에 대해 설명한다.

[0294] 도 37은 본 발명을 적용한 재생장치의 제 1예인 재생장치(20-1)의 구성예를 나타내는 블럭도이다. 이 재생장치(20-1)는 상술한 메인 패스와 서브 패스를 가지는 플레이 리스트를 재생할 수 있다. 재생장치(20-1)는 audio_stream_id로 주어지는 오디오 스트림 #1만을 재생하는 것이 가능하고, audio_stream_id2로 주어지는 오디오 스트림 #2이 존재하는지 하지 않는지에 관계없이 재생할 수 없다. 그리고, 재생장치(20-1)는 오디오 스트림 #1에 대하여, 사운드 데이터를 합성하여 재생할 수 있다.

[0295] 재생장치(20-1)에는 재생 데이터 취득부(31), 스위치(32), AV 디코더부(33-1), 컨트롤러(34-1), 오디오 엔코더(41), 비디오 엔코더(42), D/A 변환부(43), D/A 변환부(44), 비압축 오디오신호 인터페이스(51), 압축 오디오신호 인터페이스(52), 비압축 비디오신호 인터페이스(53), 압축 비디오신호 인터페이스(54), 아날로그 오디오신호 인터페이스(55) 및 아날로그 비디오신호 인터페이스(56)가 설치되어 있다.

[0296] 도 37의 예의 경우, 최초로, 컨트롤러(34-1)가, 예를 들면, 장착된 기록매체(21)로부터 데이터를 읽어들이는 스토리지 드라이브, 자기 자신의 내부의 기록 매체로부터 데이터를 읽어들이는 데이터 드라이브, 또는, 네트워크(22)를 통하여 데이터를 취득하는 네트워크 인터페이스 등의 재생 데이터 취득부(31)를 통하여 인덱스 파일을 독출하고, 발생되는 커맨드에 근거하여 플레이 리스트 파일을 독출하고, 플레이 리스트 파일의 정보에 근거하여 플레이 아이템을 독출하고, 이것에 대응하는 클립을 검출하고, ClipInfo를 근거로, 대응하는 AV 스트림이나 AV 데이터 등을 독출한다. 사용자는 상술한 리모트 커맨드(23) 등의 사용자 인터페이스를 이용하고, 컨트롤러(34-1)에 대하여, 음성이나 자막 등의 전환의 지령을 행할 수 있다. 또, 컨트롤러(34-1)에는 재생장치(20-1)의 언어설정의 초기정보가 도시하지 않은 기억부 등으로부터 공급된다.

[0297] 또, 컨트롤러(34-1)는 인덱스 파일, 플레이 리스트 파일(AppInfoPlayList(), PlayList(), 또는, PlayItem) 중 어느 쪽인가에 기재되는, 오디오 스트림에 합성 가능한 사운드 데이터의 유무를 나타내는 플래그인 is_MixApp 또는 is_MixApp_2의 값을 근거로, 스위치(61)를 제어한다.

[0298] 플레이 리스트 파일에는, 메인 패스, 서브 패스의 정보 외, STN_table()가 포함되어 있다. 컨트롤러(34-1)는 플레이 리스트 파일에 포함되는 플레이 아이템이 참조하는 메인 클립에 포함되는 메인 클립 AV 스트림 파일, 서브 클립에 포함되는 서브 플레이 아이템이 참조하는 서브 클립 AV 스트림 파일 및 서브 플레이 아이템이 참조하는 텍스트 서브 데이터를, 재생 데이터 취득부(31)를 통하여 기록매체(21) 등으로부터 독출한다. 여기서, 플레이 아이템이 참조하는 메인 클립과 서브 플레이 아이템이 참조하는 서브 클립이, 다른 기록 매체에 기록되어 있어도 좋다. 예를 들면, 메인 클립이 기록매체(21)에 기록되어 있고, 대응하는 서브 클립은 예를 들면, 네트워크(22)를 통하여 공급되어 재생장치(20-1)에 내장되어 있는 도시하지 않은 HDD에 기억된 것이어도 좋다.

또, 컨트롤러(34-1)는 자기 자신(재생장치(20-1))의 재생 기능에 대응하는 엘레멘터리 스트림을 선택하여 재생하도록 제어하거나 재생장치(20-1)의 언어설정의 초기정보에 대응하는 엘레멘터리 스트림만을 선택하여 재생하도록 제어한다.

[0299] AV 디코더부(33-1)에는, 베퍼(51 내지 54), PID 필터(55), PID 필터(56), 스위치(57 내지 59), 스위치(61), 백그라운드 디코더(71), 비디오 디코더(72), 프리젠테이션 그래픽스 디코더(73), 인터랙티브 그래픽스 디코더(74), 오디오 디코더(75), Text-ST 콤포지션(76), 스위치(77), 백그라운드 플레이인 생성부(91), 비디오 플레이인 생성부(92), 프레젠테이션 그래픽스 플레이인 생성부(93), 인터랙티브 그래픽스 플레이인 생성부(94), 베퍼(95), 비디오 데이터 처리부(96) 및 막상 처리부(97)가 설치되어 있다. 여기에서는, 오디오 디코더로서 오디오 디코더(75)만이 설치되어 있으므로, 오디오 스트림#1을 복호화할 수 있으나, 오디오 스트림#2를 복호화할 수 없다. 구체적으로는, 재생장치(20-1)에 있어서는, 도 25의 STN_table()에 있어서, audio_stream_id로 주어지는 오디오 스트림을 복호화하는 것은 가능하지만, audio_stream_id2로 주어지는 오디오 스트림을 복호화할 수 없다. 또, 비디오 디코더(72)로서는, 재생되는 스트림에 따라서, 예를 들면, MPEG2, MPEG4, H.264/AVC 등의 여러 가지의 디코더를 채용할 수 있다.

[0300] 컨트롤러(34-1)에 의해 독출된 파일 데이터는, 도시하지 않은 복조, ECC 복호부에 의해 복조되고, 복조된 다중화 스트림으로 오류정정이 실시된다. 스위치(32)는 복조되고 오류정정이 실시된 데이터를 컨트롤러(34-1)로부터의 제어에 근거하여 스트림의 종류마다 선택하고, 대응하는 베퍼(51 내지 54)에 공급한다. 구체적으로는, 스위치(32)는 컨트롤러(34-1)로부터의 제어에 근거하여 백그라운드 이미지 데이터를 베퍼(51)에 공급하고, 메인 클립 AV 스트림의 데이터를 베퍼(52)에 공급하고, 서브 클립 AV 스트림의 데이터를 베퍼(53)에 공급하고, Text-ST의 데이터를 베퍼(54)에 공급하도록 스위치(32)를 전환한다. 베퍼(51)는 백그라운드 이미지 데이터를 베퍼링하고, 베퍼(52)는 메인 클립 AV 스트림의 데이터를 베퍼링하고, 베퍼(53)는 서브 클립 AV 스트림의 데이터를 베퍼링하고, 베퍼(54)는 Text-ST 데이터를 베퍼링한다.

[0301] 메인 클립 AV 스트림은 비디오와 오디오와 비트맵 자막(Presentation Graphics stream)과 인터랙티브 그래픽스 중, 비디오에 더하여 1개 이상의 스트림을 다중화한 스트림(예를 들면 트랜스포트 스트림(transport stream))이다. 서브 클립은 오디오와 비트맵 자막(Presentation Graphics stream)과 인터랙티브 그래픽스와 오디오 중, 1개 이상의 스트림을 다중화한 스트림이다. 또한, 텍스트 서브 타이틀 데이터 파일(Text-ST)의 데이터는, 트랜스포트 스트림과 같은 다중화 스트림의 형식이어도, 그렇지 않아도 좋다.

[0302] 또, 재생 데이터 취득부(31)는 메인 클립 AV 스트림과 서브 클립 AV 스트림 및 텍스트 서브 타이틀 데이터의 각각의 파일을 시분할하여 교대로 독출하여도 좋고, 또는, 메인 클립 AV 스트림을 독출하기 전에, 서브 클립 AV 스트림이나 텍스트 서브 타이틀 데이터를 미리 독출하여 두고, 모두 베퍼(베퍼(53) 또는 베퍼(54))에 프리로드하여 두도록 해도 좋다.

[0303] 그리고, 메인 클립 AV 스트림용 리드 베퍼인 베퍼(52)로부터 독출된 스트림 데이터는, 소정의 타이밍으로, 후단의 PID(패킷 ID) 필터(55)에 출력된다. 이 PID 필터(55)는 입력된 메인 클립 AV 스트림을 PID(패킷 ID)에 따라서, 후단의 각 엘레멘터리 스트림의 디코더에 배분하여 출력한다. 즉, PID 필터(55)는 비디오 스트림을 비디오 디코더(72)에 공급하고, 프레젠테이션 그래픽스 스트림을 프레젠테이션 그래픽스 디코더(73)에의 공급원이 되는 스위치(57)에 공급하고, 인터랙티브 그래픽스 스트림을 인터랙티브 그래픽스 디코더(74)에의 공급원이 되는 스위치(58)에 공급하고, 오디오 스트림을 오디오 디코더(75)에의 공급원이 되는 스위치(59)에 공급한다.

[0304] 프레젠테이션 그래픽스 스트림은 예를 들면, 비트맵의 자막 데이터이며, 텍스트 서브 타이틀 데이터는 예를 들면, 텍스트 자막 데이터이다.

[0305] 서브 클립 AV 스트림용 리드 베퍼인 베퍼(53)로부터 독출된 스트림 데이터는, 소정의 타이밍으로 후단의 PID(패킷 ID) 필터(56)에 출력된다. 이 PID 필터(56)는 입력된 서브 클립 AV 스트림을 PID(패킷 ID)에 따라 후단의 각 엘레멘터리 스트림의 디코더에 배분하여 출력한다. 즉, PID 필터(56)는 프레젠테이션 그래픽스 스트림을 프레젠테이션 그래픽스 디코더(73)에의 공급원이 되는 스위치(57)에 공급하고, 인터랙티브 그래픽스 스트림을 인터랙티브 그래픽스 디코더(74)에의 공급원이 되는 스위치(58)에 공급하고, 오디오 스트림을 스위치(61)에의 공급원이 되는 스위치(59)에 공급한다.

[0306] 백그라운드 이미지 데이터를 베퍼링하는 베퍼(51)로부터 독출된 데이터는 소정의 타이밍으로 백그라운드 디코더(71)에 공급된다. 백그라운드 디코더(71)는 백그라운드 이미지 데이터를 복호화하고, 복호화한 백그라운드 이미지 데이터를 백그라운드 플레이인 생성부(91)에 공급한다.

- [0307] PID 필터(55)에 의해 배분된 비디오 스트림은 후단의 비디오 디코더(72)에 공급된다. 비디오 디코더(72)는 비디오 스트림을 복호화하고, 복호화한 비디오 데이터를 비디오 플레이어 생성부(92)에 출력한다.
- [0308] 스위치(57)는 PID 필터(55)로부터 공급된 메인 클립에 포함되는 프레젠테이션 그래픽스 스트림과, 서브 클립에 포함되는 프레젠테이션 그래픽스 스트림 중 어느 쪽이든 1개를 선택하고, 선택한 프레젠테이션 그래픽스 스트림을, 후단의 프레젠테이션 그래픽스 디코더(73)에 공급한다. 프레젠테이션 그래픽스 디코더(73)는 프레젠테이션 그래픽스 스트림을 복호화하고, 복호화한 프레젠테이션 그래픽스 스트림의 데이터를 프레젠테이션 그래픽스 플레이어 생성부(93)에의 공급원이 되는 스위치(77)에 공급한다.
- [0309] 또, 스위치(58)는 PID 필터(55)로부터 공급된 메인 클립에 포함되는 인터랙티브 그래픽스 스트림과, 서브 클립에 포함되는 인터랙티브 그래픽스 스트림 중 어느 쪽이든 1개를 선택하고, 선택한 인터랙티브 그래픽스 스트림을 후단의 인터랙티브 그래픽스 스트림 디코더(74)에 공급한다. 즉, 인터랙티브 그래픽스 디코더(74)에 동시에 입력되는 인터랙티브 그래픽스 스트림은 메인 클립 AV 스트림 또는 서브 클립 AV 스트림으로부터 분리된 스트림이다. 인터랙티브 그래픽스 디코더(74)는 인터랙티브 그래픽스 스트림을 복호화하고, 복호화한 인터랙티브 그래픽스 스트림의 데이터를 인터랙티브 그래픽스 플레이어 생성부(94)에 공급한다.
- [0310] 또한, 스위치(59)는 PID 필터(55)로부터 공급된 메인 클립에 포함되는 오디오 스트림과, 서브 클립에 포함되는 오디오 스트림 중 어느 쪽이든 1개를 선택하고, 선택한 오디오 스트림을 후단의 스위치(61)에 공급한다.
- [0311] 스위치(61)는 컨트롤러(34-1)의 제어에 근거하여, 공급된 오디오 데이터를 오디오 디코더(75)에 공급하거나, 또는 압축 오디오신호 인터페이스(52)에 공급한다.
- [0312] 오디오 디코더(75)는 오디오 스트림을 복호화하고, 복호화한 오디오 스트림의 데이터를 믹싱 처리부(97)에 공급한다.
- [0313] 또, 스위치(32)에 의해 선택된 사운드 데이터는 베퍼(95)에 공급되어 베퍼링된다. 베퍼(95)는 소정의 타이밍으로 사운드 데이터를 믹싱 처리부(97)에 공급한다. 사운드 데이터는, 이 경우, 메뉴 선택 등에 의한 효과음의 데이터이다. 믹싱 처리부(97)는 오디오 디코더(75)로부터 공급된 오디오 데이터와 베퍼(95)로부터 공급되어 온 사운드 데이터를 믹싱(중첩 또는 합성)하고, 음성 신호로서 출력한다.
- [0314] 텍스트 서브 타이틀용 리드 베퍼인 베퍼(54)로부터 독출된 데이터는 소정의 타이밍으로, 후단의 텍스트 서브 타이틀 콤포지션(디코더)(76)에 출력된다. 텍스트 서브 타이틀 콤포지션(76)은 Text-ST 데이터를 복호화하고, 스위치(77)에 공급한다.
- [0315] 스위치(77)는 프레젠테이션 그래픽스 디코더(73)에 의해 복호화된 프레젠테이션 그래픽스 스트림과, Text-ST(텍스트 서브 데이터) 중 어느 것을 선택하고, 선택한 데이터를 프레젠테이션 그래픽스 플레이어 생성부(93)에 공급한다. 즉, 프레젠테이션 그래픽스 플레이어 생성부(93)에 동시에 공급되는 자막 화상은 프레젠테이션 그래픽스 디코더(73) 또는 텍스트 서브 타이틀(Text-ST) 콤포지션(76) 중 몇 개의 출력이다. 또, 프레젠테이션 그래픽스 디코더(73)에 동시에 입력되는 프레젠테이션 그래픽스 스트림은 메인 클립 AV 스트림 또는 서브 클립 AV 스트림의 어느 쪽으로부터 분리된 스트림이다(스위치(57)에 의해 선택된다). 따라서, 프레젠테이션 그래픽스 플레이어 생성부(93)에 동시에 출력되는 자막 화상은 메인 클립 AV 스트림으로부터의 프레젠테이션 그래픽스 스트림, 또는 서브 클립 AV 스트림으로부터의 프레젠테이션 그래픽스 스트림, 또는 텍스트 서브 타이틀 데이터의 복호화 출력이다.
- [0316] 백그라운드 플레이어 생성부(91)는 백그라운드 디코더(71)로부터 공급된 백그라운드 이미지 데이터에 근거하여, 예를 들면, 비디오 화상을 축소 표시한 경우에 벽지(壁紙) 화상이 되는 백그라운드 플레이어 생성하고, 이것을 비디오 데이터 처리부(96)에 공급한다. 비디오 플레이어 생성부(92)는 비디오 디코더(72)로부터 공급된 비디오 데이터에 근거하여, 비디오 플레이어 생성하고, 이것을 비디오 데이터 처리부(96)에 공급한다. 프레젠테이션 그래픽스 플레이어 생성부(93)은 스위치(77)에 의해 선택되고, 공급된 데이터(프레젠테이션 그래픽스 스트림 또는 텍스트 서브 타이틀 데이터)에 근거하여, 예를 들면, 렌더링 화상인 프레젠테이션 그래픽스 플레이어 생성하고, 이것을 비디오 데이터 처리부(96)에 공급한다. 인터랙티브 그래픽스 플레이어 생성부(94)는 인터랙티브 그래픽스 디코더(74)로부터 공급된 인터랙티브 그래픽스 스트림의 데이터에 근거하여, 인터랙티브 그래픽스 플레이어 생성하고, 이것을 비디오 데이터 처리부(96)에 공급한다.
- [0317] 비디오 데이터 처리부(96)는 백그라운드 플레이어 생성부(91)로부터의 백그라운드 플레이어, 비디오 플레이어 생성부(92)로부터의 비디오 플레이어, 프레젠테이션 그래픽스 플레이어 생성부(93)로부터의 프레젠테이션 그래픽스 플레이어

및 인터랙티브 그래픽스 플레인 생성부(94)로부터의 인터랙티브 그래픽스 플레인을 합성하고, 비디오신호로서 출력한다.

- [0318] 이러한 스위치(57 내지 59) 및 스위치(77)는 사용자 인터페이스를 통하여 사용자로부터의 선택, 또는 대상이 되는 데이터가 포함되는 파일 측에 기초하여 스위치를 전환한다. 예를 들면, 서브 클립 AV 스트림 파일만으로, 오디오 스트림이 포함되지 않은 경우, 스위치(59)는 서브 측으로 스위치를 전환한다.
- [0319] 오디오 엔코더(41)는 믹싱 처리부(97)로부터 공급된 비압축의 오디오 데이터를 부호화하고, 압축 오디오신호 인터페이스(52)에 공급한다. 비디오 엔코더(42)는 비디오 데이터 처리부(96)로부터 공급된 비압축의 비디오신호를 부호화 하고, 압축 비디오신호 인터페이스(54)에 공급한다. D/A 변환부(43)는 믹싱 처리부(97)로부터 공급된 비압축의 오디오 데이터를 디지털 신호로부터 아날로그 신호로 변환하고, 아날로그 오디오신호 인터페이스(55)에 공급한다. D/A 변환부(44)는 비디오 데이터 처리부(96)로부터 공급된 비압축의 비디오신호를 디지털 신호로부터 아날로그 신호로 변환하고, 아날로그 비디오신호 인터페이스(56)에 공급한다. 비압축 오디오 신호 인터페이스(51)는 믹싱 처리부(97)로부터 공급된 비압축의 오디오 데이터를 외부에 출력한다. 압축 오디오신호 인터페이스(52)는 오디오 엔코더(41) 또는 스위치(61)로부터 공급된 압축 오디오신호를 외부에 출력한다. 비압축 비디오신호 인터페이스(53)는 비디오 데이터 처리부(96)로부터 공급된 비압축의 비디오신호를 외부에 출력한다. 압축 비디오신호 인터페이스(54)는 비디오 엔코더(42)로부터 공급된 압축 비디오신호를 외부에 출력한다. 아날로그 오디오신호 인터페이스(55)는 D/A 변환부(43)로부터 공급된 아날로그 오디오신호를 외부에 출력한다. 아날로그 비디오신호 인터페이스(56)는 D/A 변환부(44)로부터 공급된 아날로그 비디오신호를 외부에 출력한다.
- [0320] 다음으로, 도 38의 플로차트(flow chart)를 참조하여, 오디오 스트림 #1에 합성되는 다른 음성 데이터가 있는지 아닌지를 나타내는 플래그가 인덱스 파일의 Indexes()에 기재되어 있는 데이터를 재생장치(20-1)가 재생하는 경우에 실행되는 재생처리(1)에 대하여 설명한다.
- [0321] 단계(S1)에 있어서, 컨트롤러(34-1)는 재생장치(20-1)로부터 출력되는 것은 부호화 스트림인지 아닌지를 판단한다. 단계(S1)에 있어서, 재생장치(20-1)로부터 출력되는 것은 부호화 스트림이 아니라고 판단된 경우, 처리는 후술하는 단계(S24)로 진행된다.
- [0322] 단계(S1)에 있어서, 재생장치(20-1)로부터 출력되는 것은 부호화 스트림이라고 판단된 경우, 단계(S2)에 있어서, 컨트롤러(34-1)는 스위치(32)로부터 공급되는 인덱스 파일을 독출한다.
- [0323] 단계(S3)에 있어서, 컨트롤러(34-1)는 indexes()에 기재되어 있는, 사운드 데이터가 존재하는 것을 나타내는 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있는지 아닌지를 판단한다.
- [0324] audio_stream_id로 주어지는 오디오 스트림 #1에 더하여 audio_stream_id2로 주어지는 오디오 스트림 #2도 존재하고 있는 컨텐츠에는, 사운드 데이터가 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_2에 더하여 오디오 스트림 #2이 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_1이 기재되어 있다. 그러나, 재생장치(20-1)는 audio_stream_id2로 주어지는 오디오 스트림 #2를 재생하는 기능을 가지지 않는다. 따라서, 재생장치(20-1)는 단계(S3)에 있어서, is_MixApp 또는 is_MixApp_2로 나타내지는 플래그의 값만을 검출하고, 독출한 데이터 중에 is_MixApp_1로 나타내지는 플래그가 기재되어 있어도, 그 플래그는 참조하지 않는다. 즉, 재생장치(20)에 있어서, 플래그 is_MixApp_1의 값은 스위치(61)의 제어에 영향을 미치지 않는다.
- [0325] 단계(S3)에 있어서, 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있다고 판단한 경우, 처리는 후술하는 단계(S14)로 진행된다.
- [0326] 단계(S3)에 있어서, 플래그(is_MixApp 또는 is_MixApp_2)가 1은 아니라고 판단된 경우, 이 Index 파일에 관련하는 데이터에는 재생장치(20-1)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터가 포함되어 있지 않으므로, 단계(S4)에 있어서, 컨트롤러(34-1)는 이 Index 파일에 관련하는 오디오 데이터를 압축 부호화 오디오신호의 출력 단자인 압축 오디오신호 인터페이스(52)에 공급하도록 스위치(61)를 제어한다.
- [0327] 단계(S5)에 있어서, 컨트롤러(34-1)는 예를 들면, 사용자의 조작입력 등에 의해서 발생되는 커맨드에 의해, 플레이 리스트 파일(예를 들면, 도 13의 ×××××.mpls)의 독출이 지령되었는지 아닌지를 판단한다. 단계(S5)에 있어서, 플레이 리스트 파일의 독출이 지령되어 있지 않다고 판단된 경우, 플레이 리스트 파일의 독출이 지령되었다고 판단될 때까지, 단계(S5)의 처리가 반복된다.
- [0328] 단계(S5)에 있어서, 플레이 리스트 파일의 독출이 지령되었다고 판단된 경우, 단계(S6)에 있어서, 컨트롤러(34-

1)는 스위치(32)로부터 공급되는 플레이 리스트 파일을 독출한다.

[0329] 단계(S7)에 있어서, 컨트롤러(34-1)는 그 플레이 리스트 파일에 기재되어 있는 플레이 리스트(도 13의 Playlist())에 의해 지정되는 플레이 아이템을 독출하고, 대응하는 메인 클립 AV 스트림, 서브 클립 AV 스트림 및 텍스트 서브 타이틀 데이터를 독출하여 스위치(32)에 공급한다.

[0330] 단계(S8)에 있어서, 컨트롤러(34-1)는 독출한 데이터를 대응하는 버퍼에 공급하여 버퍼링한다. 구체적으로는, 컨트롤러(34-1)는 백그라운드 이미지 데이터를 버퍼(51)에 공급하고, 메인 클립 AV 스트림의 데이터를 버퍼(52)에 공급하고, 서브 클립 AV 스트림의 데이터를 버퍼(53)에 공급하고, Text-ST의 데이터를 버퍼(54)에 공급하도록 스위치(32)를 전환한다. 그리고, 버퍼(51 내지 54)는 공급된 데이터를 각각 버퍼링한다.

구체적으로는, 버퍼(51)는 백그라운드 이미지 데이터를 버퍼링하고, 버퍼(52)는 메인 클립 AV 스트림의 데이터를 버퍼링하고, 버퍼(53)는 서브 클립 AV 스트림의 데이터를 버퍼링하고, 버퍼(54)는 Text-ST 데이터를 버퍼링한다.

[0331] 단계(S9)에 있어서, 컨트롤러(34-1)는 PID 필터(55, 56) 및 스위치(67 내지 59)를 제어하고, 영상계 데이터를 소정의 디코더로 복호화하고, 비디오 데이터 처리부(96)에서 처리한다.

[0332] 단계(S10)에 있어서, 비디오 엔코더(42)는 비디오 데이터 처리부(96)로부터 공급된 처리 후의 비디오 데이터를 부호화한다.

[0333] 단계(S11)에 있어서, 압축 부호화 오디오신호의 출력 단자인 압축 오디오신호 인터페이스(52)는 스위치(61)로부터 출력되는 부호화 오디오 데이터인 오디오 스트림 #1을 외부에 출력하고, 압축 부호화 비디오신호의 출력 단자인 압축 비디오신호 인터페이스(54)는 부호화된 부호화 비디오 데이터를 외부에 출력한다.

[0334] 즉, 단계(S11)에 있어서 압축 오디오신호 인터페이스(52)로부터 출력되는 압축 부호화 오디오 데이터는 재생장치(20-1)의 AV 디코더부(33-1)에 있어서, 복호처리가 실시되지 않으므로, 음질이 열화 하지 않는다.

[0335] 단계(S12)에 있어서, 컨트롤러(34-1)는 플레이 리스트를 참조하여 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S12)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S7)로 돌아가고, 그 이후의 처리가 반복된다.

[0336] 단계(S12)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S13)에 있어서, 컨트롤러(34-1)는 재생 종료인지 아닌지를 판단한다. 단계(S13)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S5)로 돌아가고, 그 이후의 처리가 반복된다. 단계(S13)에 있어서, 재생종료라고 판단된 경우, 처리는 종료된다.

[0337] 단계(S3)에 있어서, 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있다고 판단된 경우, 이 인덱스 파일에 관련하는 데이터에는 재생장치(20-1)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터가 포함되어 있으므로, 단계(S14)에 있어서, 컨트롤러(34-1)는 이 인덱스 파일에 관련하는 오디오 스트림 #1을 오디오 디코더(75)에 공급하도록 스위치(61)를 제어한다.

[0338] 단계(S15) 내지 단계(S19)에 있어서, 단계(S5) 내지 단계(S9)와 기본적으로 같은 처리가 실행된다. 즉, 플레이 리스트의 독출이 지령되었는지 아닌지가 판단되고, 독출이 지령되었다고 판단된 경우, 플레이 리스트가 독출되어, 플레이 리스트에 의해 지정되는 플레이 아이템이 독출된다. 그리고, 독출된 플레이 아이템에 대응하는 클립의 데이터(AV Stream)가 대응하는 버퍼에 공급되어 버퍼링되고, PID 필터 및 각 스위치가 제어되고, 버퍼링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다.

[0339] 단계(S20)에 있어서, 컨트롤러(34-1)는 PID 필터 및 각 스위치를 제어하고, 버퍼(53)에 버퍼링된 오디오 스트림 #1을 스위치(61)를 통하여 오디오 디코더(75)에 공급하여 복호화하고, 믹싱 처리부(97)를 제어하여, 버퍼(95)에 보관되어 있는 사운드 데이터와의 합성처리를 적절히 실행시킨다.

[0340] 즉, 스위치(61)가 제어되고, 오디오 스트림 #1이 오디오 디코더(75)에 의해 복호화되도록 되어 있으므로, 오디오 스트림 #1으로 사운드 데이터를 합성하는 것이 가능해진다.

[0341] 단계(S21)에 있어서, 오디오 엔코더(41)는 합성처리가 실시된 비압축의 오디오 데이터를 부호화하고, 압축 오디오신호 인터페이스(52)에 공급하여 외부 출력시키고, 비디오 엔코더(42)는 비디오 데이터 처리부(96)에 의해 처리된 비압축의 비디오 데이터를 부호화하고, 압축 비디오신호 인터페이스(54)에 공급하여 외부에 출력시킨다.

[0342] 단계(S22)에 있어서, 컨트롤러(34-1)는 플레이 리스트를 참조하여, 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S22)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S17)로

돌아가고, 그 이후의 처리가 반복된다.

- [0343] 단계(S22)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S23)에 있어서, 컨트롤러(34-1)는 재생 종료인지 아닌지를 판단한다. 단계(S23)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S15)로 돌아가고, 그 이후의 처리가 반복된다. 단계(S23)에 있어서, 재생종료라고 판단된 경우, 처리는 종료된다.
- [0344] 단계(S1)에 있어서, 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 단계(S24)에 있어서, 컨트롤러(34-1)는 공급된 오디오 스트림 #1을 오디오 디코더(75)에 공급하도록 스위치(61)를 제어한다.
- [0345] 즉, 출력되는 것은 부호화 스트림이 아닌 경우, 오디오 스트림 #1에 합성되는 사운드 데이터가 있는지 아닌지에 관계없이, 오디오 스트림 #1은 복호화된다.
- [0346] 단계(S25) 내지 단계(S30)에 있어서, 단계(S15) 내지 단계(S20)와 기본적으로 같은 처리가 실행된다. 즉, 플레이 리스트의 독출이 지령되었는지 아닌지가 판단되고, 독출이 지령되었다고 판단된 경우, 플레이 리스트가 독출되어, 플레이 리스트에 의해 지정되는 플레이 아이템이 독출된다. 그리고, 독출된 플레이 아이템에 대응하는 클립의 데이터(AV Stream)가 대응하는 베퍼에 공급되어 베퍼링되고, PID 필터 및 각 스위치가 제어되고, 베퍼링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다. 그리고, PID필터 및 각 스위치가 제어되어 베퍼(53)에 베퍼링된 오디오 스트림 #1이 스위치(61)를 통하여 오디오 디코더(75)에 공급되어 복호화되고, 베퍼(95)에 사운드 데이터가 공급되어 존재하는 경우는, 막싱 처리부(97)의 처리에 의해, 적절히 합성처리가 실행된다.
- [0347] 단계(S31)에 있어서, 비압축 오디오신호 인터페이스(51)는 막싱 처리부(97)로부터 공급된 처리 후의 비압축의 비디오 데이터를 외부에 출력하고, 비압축 비디오신호 인터페이스(53)는 비디오 데이터 처리부(96)로부터 공급된 비압축의 오디오 데이터를 외부에 출력한다. 또는, D/A 변환부(43)는 비압축의 오디오 데이터를 D/A 변환하고, 아날로그 오디오신호 인터페이스(55)는 아날로그 신호를 외부에 출력하고, D/A 변환부(44)는 비압축의 비디오 데이터를 D/A 변환하고, 아날로그 비디오신호 인터페이스(56)는 아날로그 신호를 외부에 출력한다.
- [0348] 단계(S32)에 있어서, 컨트롤러(34-1)는 플레이 리스트를 참조하여 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S32)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S27)로 돌아가고, 그 이후의 처리가 반복된다.
- [0349] 단계(S32)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S33)에 있어서, 컨트롤러(34-1)는 재생 종료인지 아닌지를 판단한다. 단계(S33)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S25)로 돌아가고, 그 이후의 처리가 반복된다. 단계(S33)에 있어서, 재생종료라고 판단된 경우, 처리는 종료된다.
- [0350] 이러한 처리에 의해, 재생장치(20-1)는 Indexes()에 기재되어 있는 플래그를 참조하고, 이 Indexes()에 의해 참조되는 데이터에 사운드 데이터가 포함되지 않은 경우는 오디오 스트림 #1을 복호화하지 않고 압축 부호화된 데이터인 채 외부로 출력하므로, 필요 이상으로 음질을 열화시키는 것을 방지할 수 있다.
- [0351] 또한, 이 처리에 있어서는, 처리의 최초에 있어서, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지 판단하고, 그 후, 플래그를 참조하여, 합성처리가 실행되는지 아닌지를 판단하는 것으로 하고 있으나, 예를 들면, 우선, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하고, 그 후, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지를 판단하도록 하여도, 판단하는 처리의 순서가 다른 것뿐으로, 실질적으로 같은 처리가 실행되는 것임은 말할 필요도 없다.
- [0352] 다음으로, 도 39의 플로차트(flow chart)를 참조하여, 오디오 스트림 #1에 합성되는 다른 음성 데이터가 있는지 아닌지를 나타내는 플래그가 플레이 리스트에 기재되어 있는 데이터를 재생장치(20-1)가 재생하는 경우에 실행되는 재생처리(2)에 대하여 설명한다.
- [0353] 단계(S61)에 있어서, 컨트롤러(34-1)는 재생장치(20-1)로부터 출력되는 것은 부호화 스트림인지 아닌지를 판단한다. 단계(S61)에 있어서, 재생장치(20-1)로부터 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 처리는 후술하는 단계(S82)로 진행된다.
- [0354] 단계(S61)에 있어서, 재생장치(20-1)로부터 출력되는 것은 부호화 스트림이라고 판단된 경우, 단계(S62)에 있어서, 컨트롤러(34-1)는 스위치(32)로부터 공급되는 인덱스 파일을 독출한다.
- [0355] 단계(S63)에 있어서, 컨트롤러(34-1)는 예를 들면, 사용자의 조작입력 등에 의해 발생되는 커맨드에 의해, 플레이 리스트 파일(예를 들면, 도 13의 ××××.mpls)의 독출이 지령되었는지 아닌지를 판단한다. 단계(S63)

에 있어서, 플레이 리스트 파일의 독출이 지령되어 있지 않는다고 판단된 경우, 플레이 리스트 파일의 독출이 지령되었다고 판단될 때까지, 단계(S63)의 처리가 반복된다.

[0356] 단계(S63)에 있어서, 플레이 리스트 파일의 독출이 지령되었다고 판단된 경우, 단계(S64)에 있어서, 컨트롤러(34-1)는 스위치(32)로부터 공급되는 플레이 리스트 파일을 독출한다.

[0357] 단계(S65)에 있어서, 컨트롤러(34-1)는 플레이 리스트 파일에 기재되어 있다. 즉, AppInfoPlayList() 또는 PlayList()에 기재되어 있는, 사운드 데이터가 존재하는 것을 나타내는 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있는지 아닌지를 판단한다.

[0358] audio_stream_id로 주어지는 오디오 스트림 #1에 더하여 audio_stream_id2로 주어지는 오디오 스트림 #2도 존재하고 있는 컨텐츠에는, 사운드 데이터가 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_2에 더하여 오디오 스트림 #2가 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_1이 기재되어 있다. 그러나, 재생장치(20-1)는 audio_stream_id2로 주어지는 오디오 스트림 #2를 재생하는 기능을 가지지 않으므로, is_MixApp_2로 나타내지는 플래그의 값만을 검출하고, 독출한 데이터 중에 is_MixApp_1로 나타나는 플래그가 기재되어 있어도 그 플래그는 참조하지 않도록 되어 있다.

[0359] 단계(S65)에 있어서, 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있다고 판단된 경우, 처리는 후술하는 단계(S74)로 진행된다.

[0360] 단계(S65)에 있어서, 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있지 않다고 판단된 경우, 이 플레이 리스트 파일에 관련하는 오디오 데이터에는, 재생장치(20-1)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터가 포함되어 있지 않으므로, 단계(S66)에 있어서, 컨트롤러(34-1)는 이 플레이 리스트 파일에 관련하는 오디오 데이터를 압축 부호화 오디오신호의 출력단자인 압축 오디오신호 인터페이스(52)에 공급하도록 스위치(61)를 제어한다.

[0361] 단계(S67) 내지 단계(S71)에 있어서, 도 38의 단계(S7) 내지 단계(S11)와 기본적으로 같은 처리가 실행된다.

[0362] 즉, 플레이 리스트에 의해 지정되는 플레이 아이템이 독출되어 대응하는 클립의 데이터 (AV Stream)가 독출되고, 스위치(32)에 공급되어 독출된 데이터가 대응하는 베피에 공급되어 베피링된다. 그리고, PID 필터(55, 56) 및 스위치(67 내지 59)가 제어되고, 영상계 데이터가 소정의 디코더로 복호화되어 비디오 데이터 처리부(96)에서 처리되어 처리 후의 비디오 데이터가 부호화된다. 그리고, 스위치(61)로부터 출력되는 부호화 오디오 데이터가 압축 오디오신호 인터페이스(52)로부터 외부에 출력되고, 부호화된 부호화 비디오 데이터가 압축 비디오신호 인터페이스(54)로부터 외부에 출력된다.

[0363] 즉, 단계(S71)에 있어서 압축 오디오신호 인터페이스(52)로부터 출력되는 압축 부호화 오디오 데이터는 재생장치(20-1)의 AV 디코더부(33-1)에서 복호처리가 실시되지 않으므로, 음질이 열화하고 있지 않다.

[0364] 단계(S72)에 있어서, 컨트롤러(34-1)는 플레이 리스트를 참조하여 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S72)에 있어서, 다음의 플레이 아이템이 있다고 판단되었을 경우, 처리는 단계(S67)로 돌아가고, 그 이후의 처리가 반복된다.

[0365] 단계(S72)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S73)에 있어서, 컨트롤러(34-1)는 재생 종료인지 아닌지를 판단한다. 단계(S73)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S63)로 돌아가고, 그 이후의 처리가 반복된다. 즉, 다음에 독출된 플레이 리스트에 기재되어 있는 플래그의 값에 의해, 스위치(61)의 제어는 변경될 가능성이 있다. 단계(S73)에 있어서, 재생종료라고 판단된 경우, 처리는 종료된다.

[0366] 단계(S65)에 있어서, 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있다고 판단된 경우, 이 플레이 리스트 파일에 관련하는 데이터에는 재생장치(20-1)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터가 포함되어 있으므로, 단계(S74)에 있어서, 컨트롤러(34-1)는 이 플레이 리스트 파일에 관련하는 오디오 데이터를 복호화하는 디코더인 오디오 디코더(75)에 공급하도록 스위치(61)를 제어한다.

[0367] 단계(S75) 내지 단계(S79)에 있어서, 도 38의 단계(S17) 내지 단계(S21)와 기본적으로 같은 처리가 실행된다.

즉, 플레이 리스트에 의해 지정되는 플레이 아이템이 독출되어, 대응하는 Clip의 데이터(AV Stream)가 독출되고, 독출된 데이터가 대응하는 베피에 공급되고, 베피링되어 PID 필터 및 각 스위치가 제어되고, 베피링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다.

- [0368] 그리고, PID 필터 및 각 스위치가 제어되고, 베피(53)에 베피링된 오디오 스트림 #1이 스위치(61)를 통하여 오디오 디코더(75)에 공급되어 복호화되고, 믹싱 처리부(97)의 처리에 의해, 베피(95)에 보관되어 있는 사운드 데이터와의 합성처리가 적절히 실행된다. 즉, 스위치(61)가 제어되고, 오디오 스트림 #1이 오디오 디코더(75)에 의해 복호화되도록 되어 있으므로, 오디오 스트림 #1과 사운드 데이터를 합성하는 것이 가능해진다.
- [0369] 그리고, 합성처리가 실시된 비압축의 오디오 데이터가 부호화되고, 압축 오디오신호 인터페이스(52)로부터 외부 출력되어 비디오 데이터 처리부(96)에 의해 처리된 비압축의 비디오 데이터가 부호화되고, 압축 비디오신호 인터페이스(54)로부터 외부에 출력된다.
- [0370] 단계(S80)에 있어서, 컨트롤러(34-1)는 플레이 리스트를 참조하여 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S80)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S75)로 돌아가고, 그 이후의 처리가 반복된다.
- [0371] 단계(S80)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S81)에 있어서, 컨트롤러(34-1)는 재생 종료인지 아닌지를 판단한다. 단계(S81)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S63)로 돌아가고, 그 이후의 처리가 반복된다. 즉, 다음에 독출된 플레이 리스트에 기재되어 있는 플래그의 값에 의해, 스위치(61)의 제어는 변경될 가능성이 있다. 단계(S81)에 있어서, 재생종료라고 판단된 경우, 처리는 종료된다.
- [0372] 단계(S61)에 있어서, 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 단계(S82) 내지 단계(S91)에 있어서, 도 38의 단계(S24) 내지 단계(S33)와 기본적으로 같은 처리가 실행된다.
- [0373] 즉, 출력되는 것은 부호화 스트림이 아닌 경우, 오디오 스트림 #1에 합성되는 사운드 데이터가 있는지 아닌지에 관계없이, 오디오 스트림 #1은 복호화되므로, 공급된 오디오 스트림 #1이 오디오 디코더(75)에 공급되도록 스위치(61)가 제어된다. 그리고, 플레이 리스트의 독출이 지령되었는지 아닌지가 판단되고, 독출이 지령되었다고 판단된 경우, 플레이 리스트가 독출되고, 플레이 리스트에 의해 지정되는 플레이 아이템이 독출되어 대응하는 클립의 데이터가 독출된다. 그리고, 독출된 데이터가 대응하는 베피에 공급되어 베피링되고, PID 필터 및 각 스위치가 제어되고, 베피링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다. 그리고, PID 필터 및 각 스위치가 제어되고, 베피(53)에 베피링된 오디오 스트림 #1이 스위치(61)를 통하여 오디오 디코더(75)에 공급되어 복호화되고, 베피(95)에 사운드 데이터가 공급되어 존재하는 경우는 믹싱 처리부(97)의 처리에 의해, 적절히 합성처리가 실행된다.
- [0374] 그리고, 비압축 오디오신호 인터페이스(51)는 믹싱 처리부(97)로부터 공급된 처리 후의 비압축의 비디오 데이터를 외부에 출력하고, 비압축 비디오신호 인터페이스(53)는 비디오 데이터 처리부(96)로부터 공급된 비압축의 오디오 데이터를 외부에 출력한다. 또는, D/A 변환부(43)는 비압축의 오디오 데이터를 D/A 변환하고, 아날로그 오디오신호 인터페이스(55)는 아날로그 신호를 외부에 출력하고, D/A 변환부(44)는 비압축의 비디오 데이터를 D/A 변환하고, 아날로그 비디오신호 인터페이스(56)는 아날로그 신호를 외부에 출력한다.
- [0375] 그리고, 재생해야 할 다음의 플레이 아이템이 있는지 아닌지가 판단되고, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S85)로 돌아가고, 그 이후의 처리가 반복된다. 다음의 플레이 아이템이 없다고 판단된 경우, 재생종료인지 아닌지가 판단되고, 재생종료는 아니라고 판단된 경우, 처리는 단계(S83)로 돌아가고, 그 이후의 처리가 반복되어, 재생종료라고 판단된 경우, 처리는 종료된다.
- [0376] 이러한 처리에 의해, 재생장치(20-1)는 플레이 리스트 파일에 기재되어 있는 플래그를 참조하여 이 플레이 리스트에 의해 참조되는 데이터에 사운드 데이터가 포함되어 있지 않은 경우는, 오디오 스트림 #1을 복호화하지 않고 압축 부호화된 데이터인 채 외부로 출력하므로, 필요 이상으로 음질을 열화시키는 것을 방지할 수 있다. 또, 플래그를 플레이 리스트 파일내의 AppInfoPlayList() 또는 PlayList()에 기재하도록 한 경우, 사운드 데이터가 포함되어 있는지 아닌지를 플레이 리스트마다 설정하는 것이 가능해지므로, 오쏘링(authoring)의 자유도가 증가한다.
- [0377] 또한, 이 처리에 있어서도, 처리의 최초에 있어서, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지 판단하고, 그 후, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하는 것으로 하고 있으나, 예를 들면, 우선, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하고, 그 후, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지를 판단하도록 하여도, 판단하는 처리의 순서가 다른 것뿐, 실질적으로 같은 처리가 실행되는 것임은 말할 필요도 없다.

- [0378] 다음으로, 도 40의 플로차트(flow chart)를 참조하여, 오디오 스트림 #1에 합성되는 다른 음성 데이터가 있는지 아닌지를 나타내는 플래그가 플레이 아이템에 기재되어 있는 데이터를 재생장치(20-1)가 재생하는 경우에 실행되는 재생처리(3)에 대하여 설명한다.
- [0379] 단계(S131)에 있어서, 컨트롤러(34-1)는 재생장치(20-1)로부터 출력되는 것은 부호화 스트림인지 아닌지를 판단한다. 단계(S131)에 있어서, 재생장치(20-1)로부터 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 처리는 후술하는 단계(S151)로 진행된다.
- [0380] 단계(S131)에 있어서, 재생장치(20-1)로부터 출력되는 것은 부호화 스트림이라고 판단된 경우, 단계(S132)에 있어서, 컨트롤러(34-1)는 스위치(32)로부터 공급되는 인덱스 파일을 독출한다.
- [0381] 단계(S133)에 있어서, 컨트롤러(34-1)는 예를 들면, 사용자의 조작 입력 등에 의해서 발생되는 커맨드에 의해, 플레이 리스트 파일(예를 들면, 도 13의 $\times \times \times \times .m3s$)의 독출이 지령되었는지 아닌지를 판단한다. 단계(S133)에 있어서, 플레이 리스트 파일의 독출이 지령되어 있지 않다고 판단된 경우, 플레이 리스트 파일의 독출이 지령되었다고 판단될 때까지, 단계(S133)의 처리가 반복된다.
- [0382] 단계(S133)에 있어서, Playlist 파일의 독출이 지령되었다고 판단되었을 경우, 단계(S134)에 있어서, 컨트롤러(34-1)는, 스위치(32)로부터 공급되는 플레이 리스트 파일을 독출한다.
- [0383] 단계(S135)에 있어서, 컨트롤러(34-1)는 그 플레이 리스트 파일에 기재되어 있는 플레이 리스트(도 13의 Playlist())에 의해 지정되는 플레이 아이템을 독출한다. 그리고, 컨트롤러(34-1)는 플레이 아이템에 근거하여, 재생 데이터 취득부(31)를 제어하고, 대응하는 메인 클립 AV 스트림, 서브 클립 AV 스트림 및 텍스트 서브 타이틀 데이터를 독출하여 스위치(32)에 공급한다.
- [0384] 단계(S136)에 있어서, 컨트롤러(34-1)는 플레이 아이템에 기재되어 있는, 사운드 데이터가 존재하는 것을 나타내는 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있는지 아닌지를 판단한다.
- [0385] audio_stream_id로 주어지는 오디오 스트림 #1에 더하여 audio_stream_id2로 주어지는 오디오 스트림 #2도 존재하고 있는 컨텐츠에는, 사운드 데이터가 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_2에 더하여 오디오 스트림 #2이 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_1이 기재되어 있다. 그러나, 재생장치(20-1)는 audio_stream_id2로 주어지는 오디오 스트림 #2를 재생하는 기능을 가지지 않으므로, 단계(S136)에 있어서, is_MixApp 또는 is_MixApp_2로 나타내지는 플래그의 값만을 검출하고, 독출한 데이터 중에 is_MixApp_1로 나타내지는 플래그가 기재되어 있어도, 그 플래그는 참조하지 않는다.
- [0386] 단계(S136)에 있어서, 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있다고 판단된 경우, 처리는 후술하는 단계(S144)로 진행된다.
- [0387] 단계(S136)에 있어서, 플래그(is_MixApp 또는 is_MixApp_2)가 1은 아니라고 판단된 경우, 이 Index 파일에 관련하는 오디오 데이터에는, 재생장치(20-1)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터가 포함되어 있지 않으므로, 단계(S137)에 있어서, 컨트롤러(34-1)는 이 플레이 아이템에 관련하는 오디오 스트림 #1을 압축 부호화 오디오신호의 출력 단자인 압축 오디오신호 인터페이스(52)에 공급하도록 스위치(61)를 제어한다.
- [0388] 단계(S137) 내지 단계(S141)에 있어서, 도 38의 단계(S8) 내지 단계(S11)와 기본적으로 같은 처리가 실행된다.
- [0389] 즉, 독출된 데이터가 대응하는 버퍼에 공급되어 버퍼링되고, PID 필터(55, 56) 및 스위치(67 내지 59)가 제어되고, 영상계 데이터가 소정의 디코더로 복호화되고, 비디오 데이터 처리부(96)에서 처리되어 처리가 끝난 비디오 데이터가 부호화된다. 그리고, 스위치(61)로부터 출력되는 부호화 오디오 데이터인 오디오 스트림 #1이 압축 오디오신호 인터페이스(52)로부터 외부에 출력되고, 부호화된 부호화 비디오 데이터가 압축 비디오신호 인터페이스(54)로부터 외부에 출력된다.
- [0390] 즉, 단계(S141)에 있어서 압축 오디오신호 인터페이스(52)로부터 출력되는 압축 부호화 오디오 데이터인 오디오 스트림 #1은 재생장치(20-1)의 AV디코더부(33-1)에 있어서, 복호처리가 실시되지 않으므로, 음질이 열화하지 않는다.
- [0391] 단계(S142)에 있어서, 컨트롤러(34-1)는 플레이 리스트를 참조하여 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S142)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S135)로 돌아가고, 그 이후의 처리가 반복된다. 즉, 다음에 독출된 플레이 아이템에 기재되어 있는 플래그의 값에

의해, 스위치(61)의 제어는 변경될 가능성이 있다.

[0392] 단계(S142)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S143)에 있어서, 컨트롤러(34-1)는 재생종료인지 아닌지를 판단한다. 단계(S143)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S133)로 돌아가고, 그 이후의 처리가 반복된다. 단계(S143)에 있어서, 재생종료라고 판단된 경우, 처리는 종료된다.

[0393] 단계(S136)에 있어서, 플래그(is_MixApp 또는 is_MixApp_2)가 1로 되어 있다고 판단된 경우, 이 플레이 아이템에 관련하는 데이터에는 재생장치(20-1)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터가 포함되어 있으므로, 단계(S144)에 있어서, 컨트롤러(34-1)는 이 플레이 아이템에 관련하는 오디오 스트림 #1을 오디오 디코더(75)에 공급하도록 스위치(61)를 제어한다.

[0394] 단계(S145) 내지 단계(S148)에 있어서, 도 38의 단계(S18) 내지 단계(S21)와 기본적으로 같은 처리가 실행된다. 즉, 독출된 데이터가 대응하는 베피에 공급되어 베피링되고, PID 필터 및 각 스위치가 제어되고, 베피링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다.

[0395] 그리고, PID 필터 및 각 스위치가 제어되고, 베피(53)에 베피링된 오디오 스트림 #1이 스위치(61)를 통하여 오디오 디코더(75)에 공급되어 복호화되고, 믹싱 처리부(97)의 처리에 의해 베피(95)에 보관되어 있는 사운드 데이터와의 합성처리가 적절히 실행된다. 즉, 스위치(61)가 제어되고, 오디오 스트림 #1이 오디오 디코더(75)에 의해 복호화되도록 되어 있으므로, 오디오 스트림 #1과 사운드 데이터를 합성하는 것이 가능해진다.

[0396] 그리고, 합성처리가 실시된 비압축의 오디오 데이터가 부호화되고, 압축 오디오신호 인터페이스(52)로부터 외부 출력되어 비디오 데이터 처리부(96)에 의해 처리된 비압축의 비디오 데이터가 부호화되고, 압축 비디오신호 인터페이스(54)로부터 외부에 출력된다.

[0397] 단계(S149)에 있어서, 컨트롤러(34-1)는 플레이 리스트를 참조하여 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S149)에 있어서, 다음의 플레이 아이템이 있다고 판단되었을 경우, 처리는 단계(S135)로 돌아가고, 그 이후의 처리가 반복된다. 즉, 다음에 독출된 플레이 아이템에 기재되어 있는 플래그의 값에 의해, 스위치(61)의 제어는 변경될 가능성이 있다.

[0398] 단계(S149)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S150)에 있어서, 컨트롤러(34-1)는 재생종료인지 아닌지를 판단한다. 단계(S150)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S133)로 돌아가고, 그 이후의 처리가 반복된다.

[0399] 단계(S150)에 있어서, 재생종료라고 판단된 경우, 처리는 종료된다.

[0400] 단계(S131)에 있어서, 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 단계(S151) 내지 단계(S160)에 있어서, 도 38의 단계(S24) 내지 단계(S33)와 기본적으로 같은 처리가 실행된다.

[0401] 즉, 출력되는 것은 부호화 스트림이 아닌 경우, 오디오 스트림 #1에 합성되는 사운드 데이터가 있는지 아닌지에 관계없이, 오디오 스트림 #1은 복호화되므로, 공급된 오디오 스트림 #1이 오디오 디코더(75)에 공급되도록 스위치(61)가 제어된다. 그리고, 플레이 리스트의 독출이 지령되었는지 아닌지가 판단되고, 독출이 지령되었다고 판단된 경우, 플레이 리스트가 독출되고, 플레이 리스트에 의해 지정되는 플레이 아이템이 독출되어 대응하는 클립의 데이터(AV Stream)가 독출된다. 그리고, 독출된 데이터가 대응하는 베피에 공급되어 베피링되고, PID 필터 및 각 스위치가 제어되고, 베피링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다.

[0402] 그리고, PID 필터 및 각 스위치가 제어되어 베피(53)에 베피링된 오디오 스트림 #1이 스위치(61)를 통하여 오디오 디코더(75)에 공급되어 복호화되고, 베피(95)에 사운드 데이터가 공급되어 존재하는 경우는, 믹싱 처리부(97)의 처리에 의해, 적절히 합성처리가 실행된다.

[0403] 그리고, 비압축 오디오신호 인터페이스(51)는, 믹싱 처리부(97)로부터 공급된 처리 후의 비압축의 비디오 데이터를 외부에 출력하고, 비압축 비디오신호 인터페이스(53)는, 비디오 데이터 처리부(96)으로부터 공급된 비압축의 오디오 데이터를 외부에 출력한다. 또는, D/A 변환부(43)는 비압축의 오디오 데이터를 D/A 변환하고, 아날로그 오디오신호 인터페이스(55)는 아날로그 신호를 외부에 출력하고, D/A 변환부(44)는 비압축의 비디오 데이터를 D/A 변환하고, 아날로그 비디오신호 인터페이스(56)는 아날로그 신호를 외부에 출력한다.

[0404] 그리고, 재생해야 할 다음의 플레이 아이템이 있는지 아닌지가 판단되어 다음의 플레이 아이템이 있다고 판단된

경우, 처리는 단계(S154)로 돌아가고, 그 이후의 처리가 반복된다. 다음의 플레이 아이템이 없다고 판단된 경우, 재생종료인지 아닌지가 판단되고, 재생종료는 아니라고 판단된 경우, 처리는 단계(S152)로 돌아가고, 그 이후의 처리가 반복되어, 재생종료라고 판단된 경우 처리는 종료된다.

[0405] 이러한 처리에 의해, 재생장치(20-1)는 플레이 아이템에 기재되어 있는 플래그를 참조하여 이 플레이 아이템에 의해 참조되는 데이터에 사운드 데이터가 포함되어 있지 않은 경우는, 오디오 스트림#1을 복호화하지 않고 압축 부호화된 데이터인 채 외부로 출력하므로, 필요 이상으로 음질을 열화시키는 것을 방지할 수 있다. 또, 플래그가 플레이 아이템에 기재되어 있으므로, 사운드 데이터가 포함되어 있는지 아닌지를 플레이 아이템마다 설정하는 것이 가능해지므로, 오쏘링의 자유도가 더욱 늘어난다.

[0406] 또한, 이 처리에 있어서도, 처리의 최초에 있어서, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지를 판단하고, 그 후, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하는 것으로 하고 있으나, 예를 들면, 우선, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하고, 그 후, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지를 판단하도록 해도, 판단하는 처리의 순서가 다른 것뿐, 실질적으로 같은 처리가 실행되는 것임은 말할 필요도 없다.

[0407] 다음으로, 본 발명을 적용한 재생장치(20)의 제 2 실시예인, 오디오 스트림#1 및 오디오 스트림#2의 2개의 음성 스트림의 재생이 가능한 재생장치(20-2)에 대하여 설명한다. 도 41은 본 발명을 적용한 재생장치(20-2)의 구성예를 나타내는 블럭도이다. 이 재생장치(20-2)는 상술한 메인 패스와 서브 패스를 가지는 플레이 리스트를 재생하는 재생장치(20-2)이다. 재생장치(20-2)는 audio_stream_id로 주어지는 오디오 스트림#1에 대하여, audio_stream_id2로 주어지는 오디오 스트림#2와 사운드 데이터를 합성하여 재생할 수 있다.

[0408] 또한, 도 37을 이용하여 설명한 재생장치(20-1)에 대응하는 부분에는 동일한 부호를 붙이고, 그 설명은 적절히 생략한다.

[0409] 즉, 재생장치(20-2)에는, 재생장치(20-1)와 동일하게 하여, 재생 데이터 취득부(31), 스위치(32), 오디오 엔코더(41), 비디오 엔코더(42), D/A 변환부(43), D/A 변환부(44), 비압축 오디오신호 인터페이스(51), 압축 오디오신호 인터페이스(52), 비압축 비디오신호 인터페이스(53), 압축 비디오신호 인터페이스(54), 아날로그 오디오신호 인터페이스(55) 및 아날로그 비디오신호 인터페이스(56)가 설치되고, 재생장치(20-1)의 AV 디코더부(33-1) 및 컨트롤러(34-1)에 대신하여, AV 디코더부(33-2) 및 컨트롤러(34-2)가 설치되어 있다.

[0410] 도 41의 예의 경우도, 도 37을 이용하여 설명한 경우와 동일하게 하여, 컨트롤러(34-2)는 재생 데이터 취득부(31)를 통하여 인덱스 파일을 독출하고, 발생되는 커맨드에 근거하여 플레이 리스트 파일을 독출하고, 플레이 리스트 파일의 정보에 근거하여 플레이 아이템을 독출하고, 대응하는 클립(AV 스트림이나 AV 데이터 등)을 독출한다. 사용자는 사용자 인터페이스를 이용하여 컨트롤러(34-2)에 대하여 음성이나 자막 등의 전환의 지령을 행할 수 있다. 또, 컨트롤러(34-2)에는, 재생장치(20)의 언어설정의 초기정보가 도시하지 않은 기억부 등으로부터 공급된다.

[0411] 또, 컨트롤러(34-2)는 인덱스 파일, 플레이 리스트 파일(AppInfoPlayList(), PlayList(), 또는, PlayItem) 중 어느 것에 기재되는 오디오 스트림#1에 합성 가능한 사운드 데이터의 유무를 나타내는 플래그인 is_MixApp 또는 is_MixApp_2의 값 및 오디오 스트림#1에 합성 가능한 오디오 스트림#2 유무를 나타내는 플래그인 is_MixApp_1을 근거로, 스위치(61)를 제어한다.

[0412] AV 디코더부(33-2)에는, AV 디코더부(33-1)와 동일하게, 버퍼(51 내지 54), PID 필터(55), PID 필터(56), 스위치(57 내지 58), 스위치(61), 스위치(102), 백그라운드 디코더(71), 비디오 디코더(72), 프레젠테이션 그래픽스 디코더(73), 인터랙티브 그래픽스 디코더(74), Text-ST 콤포지션(76), 스위치(77), 백그라운드 플레이인 생성부(91), 비디오 플레이인 생성부(92), 프레젠테이션 그래픽스 플레이인 생성부(93), 인터랙티브 그래픽스 플레이인 생성부(94), 버퍼(95), 비디오 데이터 처리부(96) 및 믹싱 처리부(97)가 설치되고, 스위치(59)에 대신하여 스위치(101)가 설치되고, 오디오 디코더(75)에 대신하여, 제 1 오디오 디코더(75-1)가 설치되고, 새롭게, 제 2 오디오 디코더(75-2) 및 믹싱 처리부(102)가 설치되어 있다.

[0413] 제 1 오디오 디코더(75-1)는 오디오 스트림#1을 복호화하기 위한 것이고, 제 2 오디오 디코더(75-2)는 오디오 스트림#2를 복호화하기 위한 것이다. 구체적으로는, 도 25의 STN_table()에 있어서, audio_stream_id로 주어지는 오디오 스트림을 복호화하기 위한 것이 제 1 오디오 디코더(75-1)이며, audio_stream_id2로 주어지는 오디오 스트림을 복호화하기 위한 것이 제 2 오디오 디코더(75-2)이다.

[0414] 이와 같이, 재생장치(20-2)는 2개의 오디오 스트림을 복호화하기 위해, 2개의 오디오 디코더(제 1 오디오 디코

더(75-1), 제 2 오디오 디코더(75-2))를 가지고 있다.

[0415] 컨트롤러(34-2)에 의해 독출된 파일 데이터는, 도시하지 않은 복조, ECC 복호부에 의해 복조되고, 복조된 다중화 스트림으로 오류정정이 실시된다. 스위치(32)는 복조되어 오류 정정이 실시된 데이터를 컨트롤러(34-2)로부터의 제어에 근거하여 스트림의 종류마다 선택하고, 대응하는 베파(51 내지 54)에 공급한다.

[0416] 또, 메인 클립 및 서브 클립에 포함되는 AV 스트림의 스트림 형식이나, 재생 데이터 취득부(31)가 데이터를 독출할 때의 방법(시분할이거나, 프리로드 하여도 좋다)은 상술한 경우와 동일한 것임은 말할 필요도 없다.

[0417] 메인 클립용 리드 베파인 베파(52)로부터 독출된 스트림 데이터는, 소정의 타이밍으로, 후단의 PID(패킷 ID) 필터(55)에 출력된다. 이 PID 필터(55)는 입력된 메인 클립을 PID(패킷 ID)에 따라서, 후단의 각 엘레멘터리 스트림의 디코더에 배분하여 출력한다. 즉, PID 필터(55)는 비디오 스트림을 비디오 디코더(72)에 공급하고, 프레젠테이션 그래픽스 스트림을 프레젠테이션 그래픽스 디코더(73)로의 공급원이 되는 스위치(57)에 공급하고, 인터랙티브 그래픽스 스트림을 인터랙티브 그래픽스 디코더(74)로의 공급원이 되는 스위치(58)에 공급하고, 오디오 스트림을 스위치(61)와 제 2 오디오 디코더(75-2)로의 공급원이 되는 스위치(101)에 공급한다.

[0418] 여기서, 스위치(61)에 공급되어 압축 오디오신호 인터페이스(52)로부터 출력되거나, 또는, 제 1 오디오 디코더(75-1)에 입력되는 오디오 스트림은 메인 Clip 또는 서브 Clip 중 어느 것으로부터 분리된 스트림이다. 또, 동일하게, 제 2 오디오 디코더(75-2)에 입력되는 오디오 스트림도, 메인 클립 또는 서브 클립으로부터 분리된 스트림이다. 예를 들면, 메인 클립에 오디오 스트림#1과 오디오 스트림#2이 포함되어 있는 경우, PID 필터(55)는 오디오 스트림의 PID에 근거하여, 오디오 스트림#1과 오디오 스트림#2을 필터링하여 스위치(101)에 공급한다.

[0419] 재생장치(20-2)에 있어서는, 비디오 데이터, 비트맵 자막 데이터, 인터랙티브 그래픽스 데이터에 관하여는, 재생장치(20-1)와 동일하게 하여 재생처리를 실시하도록 되어 있다.

[0420] 스위치(101)는, PID 필터(55)로부터 공급된 메인 클립에 포함되는 오디오 스트림과 서브 클립에 포함되는 오디오 스트림 중 어느 쪽이든 1개를 선택하고, 선택한 오디오 스트림을 후단의 스위치(61), 또는, 제 2 오디오 디코더(75-2)에 공급한다.

[0421] 스위치(101)는 예를 들면, PID 필터(55)로부터 공급된 오디오 스트림#1을 스위치(61)에 공급하도록 스위치를 선택하고, PID 필터(55)로부터 공급된 오디오 스트림#2을 제 2 오디오 디코더(75-2)에 공급하도록 스위치를 선택한다.

[0422] 스위치(61)는, 컨트롤러(34-2)의 제어에 근거하여, 공급된 오디오 데이터를 제 1 오디오 디코더(75-1) 또는 압축 오디오신호 인터페이스(52)의 어느 쪽이든 한편에 공급한다

[0423] 제 1오디오 디코더(75-1)는 오디오 스트림을 복호화하고, 복호화한 오디오 스트림의 데이터를 믹싱 처리부(102)에 공급한다. 또, 제 2 오디오 디코더(75-2)는 오디오 스트림을 복호화하고, 복호화한 오디오 스트림의 데이터를 믹싱 처리부(102)에 공급한다.

[0424] 여기서, 오디오 스트림#1과 오디오 스트림#2을 중첩하여 재생하는 경우(사용자에 의해 재생하는 오디오 스트림으로서 2개의 오디오 스트림이 선택된 경우), 제 1 오디오 디코더(75-1)에 의해 복호화된 오디오 스트림#1과 제 2 오디오 디코더(75-2)에 의해 복호화된 오디오 스트림#2이 믹싱 처리부(102)에 공급된다.

[0425] 믹싱 처리부(102)는 제 1 오디오 디코더(75-1)로부터의 오디오 데이터와 제 2 오디오 디코더(75-2)로부터의 오디오 데이터를 믹싱(중첩)하고, 후단의 믹싱 처리부(97)에 출력한다. 또한, 본 실시의 형태에 있어서는, 제 1 오디오 디코더(75-1)로부터 출력되는 오디오 데이터와 제 2 오디오 디코더(75-2)로부터 출력되는 오디오 데이터를 믹싱(중첩)하는 것을 합성한다라고도 칭한다. 즉, 합성이란 2개의 오디오 데이터를 믹싱하는 것으로도 나타내는 것으로 한다.

[0426] 또, 스위치(32)에 의해 선택된 사운드 데이터는 베파(95)에 공급되어 베파링된다. 베파(95)는 소정의 타이밍에 사운드 데이터를 믹싱 처리부(97)에 공급한다. 사운드 데이터는 이 경우, 메뉴 선택 등에 의한 효과음의 데이터 등, 스트림과는 독립하여 존재하는 음성 데이터이다. 믹싱 처리부(97)는 믹싱 처리부(102)에 의해 믹싱된 오디오 데이터(제 1 오디오 디코더(75-1)로부터 출력된 오디오 데이터와 제 2 오디오 디코더(75-2)로부터 출력된 오디오 데이터가 믹싱된 오디오 데이터)와 베파(95)로부터 공급되어 온 사운드 데이터를 믹싱(중첩), 또는 합성)하고, 음성신호로서 출력한다.

- [0427] 다음으로, 도 42의 플로차트(flow chart)를 참조하여, 오디오 스트림 #1에 합성되는 다른 음성 데이터가 있는지 아닌지를 나타내는 플래그가 인덱스 파일의 Indexes()에 기재되어 있는 데이터를 재생장치(20-2)가 재생하는 경우에 실행되는 재생처리(4)에 대하여 설명한다.
- [0428] 단계(S201)에 있어서, 컨트롤러(34-2)는 재생장치(20-2)로부터 출력되는 것은 부호화 스트림인지 아닌지를 판단한다. 단계(S201)에 있어서, 재생장치(20-2)로부터 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 처리는 후술하는 단계(S224)로 진행된다.
- [0429] 단계(S201)에 있어서, 재생장치(20-2)로부터 출력되는 것은 부호화 스트림이라고 판단된 경우, 단계(S202)에 있어서, 컨트롤러(34-2)는 스위치(32)로부터 공급되는 인덱스 파일을 독출한다.
- [0430] 단계(S203)에 있어서, 컨트롤러(34-2)는 인덱스 파일의 indexes()에 기재되어 있는, 메인의 오디오 스트림인 오디오 스트림 #1에 합성되는 데이터가 존재하는 것을 나타내는 플래그 중 어느 한쪽이 1로 되어 있는지 아닌지, 구체적으로는, 사운드 데이터가 존재하는 것을 나타내는 플래그(is_MixApp 또는 is_MixApp_2) 또는, 오디오 스트림 #2이 존재하는 것을 나타내는 플래그(is_MixApp_1) 중 어느 한쪽이 1로 되어 있는지 아닌지를 판단한다.
- [0431] audio_stream_id로 주어지는 오디오 스트림 #1에 더하여 audio_stream_id2로 주어지는 오디오 스트림 #2도 존재하고 있는 컨텐츠에는, 사운드 데이터가 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_2에 더하여 오디오 스트림 #2가 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_1이 기재되어 있다. 재생장치(20-2)는 audio_stream_id2로 주어지는 오디오 스트림 #2을 재생하는 기능을 가지고 있으므로, 단계(S203)에 있어서, is_MixApp 또는 is_MixApp_2로 나타나는 플래그의 값뿐만 아니라, is_MixApp_1로 나타나는 플래그의 값을 참조하도록 되어 있다.
- [0432] 단계(S203)에 있어서, 몇 개의 플래그가 1로 되어 있다고 판단된 경우, 처리는 후술하는 단계(S214)로 진행된다.
- [0433] 단계(S203)에 있어서, 어느 플래그도 1이 아니라고 판단된 경우, 이 인덱스 파일에 관련하는 데이터에는, 재생장치(20-2)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터 및 오디오 스트림 #2이 포함되어 있지 않으므로, 단계(S204)에 있어서, 컨트롤러(34-2)는 이 인덱스 파일에 관련하는 오디오 데이터를 압축 부호화 오디오신호의 출력단자인 압축 오디오신호 인터페이스(52)에 공급하도록 스위치(61)를 제어한다.
- [0434] 단계(S205)에 있어서, 컨트롤러(34-2)는 예를 들면, 사용자의 조작입력 등에 의해서 발생되는 커맨드에 의해, 플레이 리스트 파일(예를 들면, 도 13의 ××××.m3u)의 독출이 지령되었는지 아닌지를 판단한다. 단계(S205)에 있어서, 플레이 리스트 파일의 독출이 지령되어 있지 않다고 판단된 경우, 플레이 리스트 파일의 독출이 지령되었다고 판단될 때까지, 단계(S205)의 처리가 반복된다.
- [0435] 단계(S205)에 있어서, 플레이 리스트 파일의 독출이 지령되었다고 판단된 경우, 단계(S206)에 있어서, 컨트롤러(34-2)는 스위치(32)로부터 공급되는 플레이 리스트 파일을 독출한다.
- [0436] 단계(S207)에 있어서, 컨트롤러(34-2)는 그 플레이 리스트 파일에 기재되어 있는 플레이 리스트(도 13의 Playlist())에 의해 지정되는 플레이 아이템을 독출하고, 대응하는 메인 클립 AV 스트림, 서브 클립 AV 스트림 및 텍스트 서브 타이틀 데이터를 독출하여 스위치(32)에 공급한다.
- [0437] 단계(S208)에 있어서, 컨트롤러(34-2)는 독출한 데이터를 대응하는 베피에 공급하여 베피링한다. 구체적으로는, 컨트롤러(34-2)는 백그라운드 이미지 데이터를 베피(51)에 공급하고, 메인 클립 AV 스트림의 데이터를 베피(52)에 공급하고, 서브 클립 AV 스트림의 데이터를 베피(53)에 공급하고, Text-ST의 데이터를 베피(54)에 공급하도록 스위치(32)를 전환한다. 그리고, 베피(51 내지 54)는 공급된 데이터를 각각 베피링한다. 구체적으로는, 베피(51)는 백그라운드 이미지 데이터를 베피링하고, 베피(52)는 메인 클립 AV 스트림의 데이터를 베피링하고, 베피(53)는 서브 클립 AV 스트림의 데이터를 베피링하고, 베피(54)는 Text-ST 데이터를 베피링한다.
- [0438] 단계(S209)에 있어서, 컨트롤러(34-2)는 PID 필터(55, 56) 및 스위치(67 내지 59)를 제어하고, 영상계 데이터를 소정의 디코더로 복호화하고, 비디오 데이터 처리부(96)에서 처리한다.
- [0439] 단계(S210)에 있어서, 비디오 엔코더(42)는 비디오 데이터 처리부(96)로부터 공급된 처리 후의 비디오 데이터를 부호화한다.
- [0440] 단계(S211)에 있어서, 압축 부호화 오디오신호의 출력단자인 압축 오디오신호 인터페이스(52)는 스위치(61)로부터 출력되는 부호화 오디오 데이터인 오디오 스트림 #1을 외부에 출력하고, 압축 부호화 비디오신호의 출력단자

인 압축 비디오신호 인터페이스(54)는, 부호화된 부호화 비디오 데이터를 외부에 출력한다.

[0441] 즉, 단계(S211)에 있어서 압축 오디오신호 인터페이스(52)로부터 출력되는 압축 부호화 오디오 데이터는, 재생 장치(20-2)의 AV 디코더부(33-2)에 있어서, 복호처리가 실시되지 않으므로, 음질이 열화하고 있지 않다.

[0442] 단계(S212)에 있어서, 컨트롤러(34-2)는 플레이 리스트를 참조하여 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S212)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S207)로 돌아가고, 그 이후의 처리가 반복된다.

[0443] 단계(S212)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S213)에 있어서, 컨트롤러(34-2)는 재생종료인지 아닌지를 판단한다. 단계(S213)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S205)로 돌아가고, 그 이후의 처리가 반복된다. 단계(S213)에 있어서, 재생종료라고 판단된 경우, 처리는 종료 된다.

[0444] 단계(S203)에 있어서, 어느 플래그가 1로 되어 있다고 판단된 경우, 이 인덱스 파일에 관련하는 데이터에는, 재생장치(20-2)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터 또는 오디오 스트림 #2 중 적어도 한쪽이 포함되어 있으므로, 단계(S214)에 있어서, 컨트롤러(34-2)는 이 인덱스 파일에 관련하는 오디오 스트림 #1을 제 1 오디오 디코더(75-1)에 공급하도록 스위치(61)를 제어한다.

[0445] 단계(S215) 내지 단계(S219)에 있어서, 단계(S205) 내지 단계(S209)와 기본적으로 같은 처리가 실행된다. 즉, 플레이 리스트 파일의 독출이 지령되었는지 아닌지가 판단되고, 독출이 지령되었다고 판단된 경우, 플레이 리스트 파일이 독출되어, 그 플레이 리스트 파일에 포함되는 플레이 리스트에 의해 지정되는 플레이 아이템이 독출된다. 그리고, 독출된 플레이 아이템에 대응하는 클립의 데이터(AV Stream)가 대응하는 베퍼에 공급되어 베퍼링되고, PID 필터 및 각 스위치가 제어되고, 베퍼링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다.

[0446] 단계(S220)에 있어서, 컨트롤러(34-2)는 PID 필터 및 각 스위치를 제어하고, 베퍼(53)에 베퍼링된 오디오 스트림 #1을 스위치(101) 및 스위치(61)를 통하여 제 1 오디오 디코더(75-1)에 공급하여 복호화시키고, 베퍼(53)에 베퍼링된 오디오 스트림 #2를 스위치(101)를 통하여 제 2 오디오 디코더(75-2)에 공급하여 복호화시키고, 믹싱 처리부(102)를 제어하고, 제 1 오디오 디코더(75-1) 및 제 2 오디오 디코더(75-2)에서 복호화된 오디오 스트림의 합성처리를 적절히 실행시키고, 믹싱 처리부(97)를 제어하고, 믹싱 처리부(102)에 의해 합성된 오디오 스트림과 베퍼(95)에 보관되어 있는 사운드 데이터와의 합성처리를 적절히 실행시킨다.

[0447] 즉, 스위치(61)가 제어되고, 오디오 스트림 #1이 오디오 디코더(75)에 의해 복호화되도록 되어 있으므로, 오디오 스트림 #1과 오디오 스트림 #2나 사운드 데이터를 합성하는 것이 가능해진다.

[0448] 단계(S221)에 있어서, 오디오 엔코더(41)는 합성처리가 실시된 비압축의 오디오 데이터를 부호화하고, 압축 오디오신호 인터페이스(52)에 공급하여 외부 출력시키고, 비디오 엔코더(42)는 비디오 데이터 처리부(96)에 의해 처리된 비압축의 비디오 데이터를 부호화하고, 압축 비디오신호 인터페이스(54)에 공급하여 외부로 출력시킨다.

[0449] 단계(S222)에 있어서, 컨트롤러(34-2)는 플레이 리스트를 참조하여, 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S222)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S217)로 돌아가고, 그 이후의 처리가 반복된다.

[0450] 단계(S222)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S223)에 있어서, 컨트롤러(34-2)는 재생종료인지 아닌지를 판단한다. 단계(S223)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S215)로 돌아가고, 그 이후의 처리가 반복된다. 단계(S223)에 있어서, 재생종료라고 판단된 경우, 처리는 종료 된다.

[0451] 단계(S201)에 있어서, 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 단계(S224)에 있어서, 컨트롤러(34-2)는 공급된 오디오 스트림 #1을 오디오 디코더(75)에 공급하도록 스위치(61)를 제어한다.

[0452] 즉, 출력되는 것은 부호화 스트림이 아닌 경우, 오디오 스트림 #1에 합성되는 사운드 데이터가 있는지 아닌지에 관계없이, 오디오 스트림 #1은 복호화된다.

[0453] 단계(S225) 내지 단계(S230)에 있어서, 단계(S215) 내지 단계(S220)와 기본적으로 같은 처리가 실행된다. 즉, 플레이 리스트 파일의 독출이 지령되었는지 아닌지가 판단되고, 독출이 지령되었다고 판단된 경우, 플레이 리스트 파일이 독출되어, 그 플레이 리스트 파일에 포함되는 플레이 리스트에 의해 지정되는 플레이 아이템이

독출된다. 그리고, 독출된 플레이 아이템에 대응하는 클립의 데이터(AV Stream)가 대응하는 버퍼에 공급되어 버퍼링되고, PID 필터 및 각 스위치가 제어되고, 버퍼링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다. 그리고, PID 필터 및 각 스위치가 제어되어 버퍼(53)에 버퍼링된 오디오 스트림 #1이 스위치(101) 및 스위치(61)를 통하여 제 1 오디오 디코더(75-1)에 공급되어 복호화되고, 오디오 스트림 #2이 스위치(101)를 통하여 제 2 오디오 디코더(75-2)에 공급되어 복호화되고, 믹싱 처리부(102)를 제어하고, 제 1 오디오 디코더(75-1) 및 제 2 오디오 디코더(75-2)에서 복호화된 오디오 스트림이 적절히 합성되고, 믹싱 처리부(97)를 제어하고, 믹싱 처리부(102)에 의해 합성된 오디오 스트림과 버퍼(95)에 보관되어 있는 사운드 데이터가 적절히 합성된다.

[0454] 단계(S231)에 있어서, 비압축 오디오신호 인터페이스(51)는 믹싱 처리부(97)로부터 공급된 처리 후의 비압축의 비디오 데이터를 외부에 출력하고, 비압축 비디오신호 인터페이스(53)는 비디오 데이터 처리부(96)로부터 공급된 비압축의 오디오 데이터를 외부에 출력한다. 또는, D/A 변환부(43)는 비압축의 오디오 데이터를 D/A 변환하고, 아날로그 오디오신호 인터페이스(55)는 아날로그 신호를 외부에 출력하고, D/A 변환부(44)는 비압축의 비디오 데이터를 D/A 변환하고, 아날로그 비디오신호 인터페이스(56)는 아날로그 신호를 외부에 출력한다.

[0455] 단계(S232)에 있어서, 컨트롤러(34-2)는 플레이 리스트를 참조하여, 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S232)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S227)로 돌아가고, 그 이후의 처리가 반복된다.

[0456] 단계(S232)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S233)에 있어서, 컨트롤러(34-2)는 재생종료인지 아닌지를 판단한다. 단계(S233)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S225)로 돌아가고, 그 이후의 처리가 반복된다. 단계(S233)에 있어서, 재생종료라고 판단된 경우, 처리는 종료된다.

[0457] 이러한 처리에 의해, 재생장치(20-2)는 인덱스 파일의 Indexes()에 기재되어 있는 플래그를 참조하고, 이 Indexes()에 의해 참조되는 데이터에 사운드 데이터 또는 오디오 스트림 #2이 어느 것도 포함되어 있지 않은 경우에는, 오디오 스트림 #1을 복호화하지 않고 압축 부호화된 데이터인 채 외부로 출력하므로, 필요 이상으로 음질을 열화시키는 것을 방지할 수 있다.

[0458] 또한, 이 처리에 있어서도, 처리의 최초에 있어서, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지를 판단하고, 그 후, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하는 것으로 하고 있으나, 예를 들면, 우선, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하고, 그 후, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지를 판단하도록 해도, 판단하는 처리의 순서가 다른 것뿐, 실질적으로 같은 처리가 실행되는 것임은 말할 필요도 없다.

[0459] 다음으로, 도 43의 플로차트(flow chart)를 참조하여, 재생장치(20-2)에 있어서, 오디오 스트림 #1에 합성되는 다른 음성 데이터가 있는지 아닌지를 나타내는 플래그가 플레이 리스트 파일에 기재되어 있는 데이터가 재생되는 경우에 실행되는 재생처리(5)에 대하여 설명한다.

[0460] 단계(S261)에 있어서, 컨트롤러(34-2)는 재생장치(20-2)로부터 출력되는 것은 부호화 스트림인지 아닌지를 판단한다. 단계(S261)에 있어서, 재생장치(20-2)로부터 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 처리는 후술하는 단계(S282)로 진행된다.

[0461] 단계(S261)에 있어서, 재생장치(20-2)로부터 출력되는 것은 부호화 스트림이라 판단된 경우, 단계(S262)에 있어서, 컨트롤러(34-2)는 스위치(32)로부터 공급되는 인덱스 파일을 독출한다.

[0462] 단계(S263)에 있어서, 컨트롤러(34-2)는 예를 들면, 사용자의 조작입력 등에 의해서 발생되는 커맨드에 의해, 플레이 리스트 파일(예를 들면, 도 13의 ×××××.mpls)의 독출이 지령되었는지 아닌지를 판단한다. 단계(S263)에 있어서, 플레이 리스트 파일의 독출이 지령되어 있지 않다고 판단된 경우, 플레이 리스트 파일의 독출이 지령되었다고 판단될 때까지, 단계(S263)의 처리가 반복된다.

[0463] 단계(S263)에 있어서, 플레이 리스트 파일의 독출이 지령되었다고 판단된 경우, 단계(S264)에 있어서, 컨트롤러(34-2)는 스위치(32)로부터 공급되는 플레이 리스트 파일을 독출한다.

[0464] 단계(S265)에 있어서, 컨트롤러(34-2)는 플레이 리스트 파일에 기재되어 있다. 즉, AppInfoPlayList() 또는 PlayList()에 기재되어 있는 메인의 오디오 스트림인 오디오 스트림 #1에 합성되는 데이터가 존재하는 것을 나타내는 플래그 중 어느 한쪽이 1로 되어 있는지 아닌지, 구체적으로는, 사운드 데이터(혹은, 오디오 스트림 #

2)가 존재하는 것을 나타내는 플래그(is_MixApp), 또는, 사운드 데이터가 존재하는 것을 나타내는 플래그(is_MixApp_2), 또는, 오디오 스트림#2가 존재하는 것을 나타내는 플래그(is_MixApp_1) 중 어느 한쪽이 1로 되어 있는지 아닌지를 판단한다.

[0465] audio_stream_id로 주어지는 오디오 스트림#1에 더하여 audio_stream_id2로 주어지는 오디오 스트림#2도 존재하고 있는 컨텐츠에는, 사운드 데이터가 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_2에 더하여 오디오 스트림#2가 존재하는지 아닌지를 나타내는 플래그인 is_MixApp_1이 기재되어 있다. 재생장치(20-2)는 audio_stream_id2로 주어지는 오디오 스트림#2을 재생하는 기능을 가지고 있으므로, 단계(S265)에 있어서, is_MixApp_2로 나타나는 플래그의 값뿐만 아니라, is_MixApp_1로 나타나는 플래그의 값을 참조하도록 되어 있다.

[0466] 단계(S265)에 있어서, 어느 플래그가 1로 되어 있다고 판단된 경우, 처리는 후술하는 단계(S274)로 진행된다.

[0467] 단계(S265)에 있어서, 어느 플래그도 1이 아니라고 판단된 경우, 이 플레이 리스트 파일에 관련하는 데이터에는, 재생장치(20-2)에 있어서 오디오 스트림#1에 합성하는 것이 가능한 사운드 데이터 및 오디오 스트림#2이 포함되어 있지 않으므로, 단계(S266)에 있어서, 컨트롤러(34-2)는 이 플레이 리스트 파일에 관련하는 오디오 데이터를 압축 부호화 오디오신호의 출력단자인 압축 오디오신호 인터페이스(52)에 공급하도록 스위치(61)를 제어한다.

[0468] 단계(S267) 내지 단계(S271)에 있어서, 도 42의 단계(S207) 내지 단계(S211)와 같은 처리가 실행된다. 즉, 플레이 리스트에 의해 지정되는 플레이 아이템이 독출되어, 대응하는 메인 클립 또는 서브 클립에 포함되는 스트림 데이터 및 텍스트 서브 타이틀 데이터가 독출되어 스위치(32)에 공급된다. 독출된 데이터는 대응하는 버퍼에 공급되어 버퍼링되고, PID 필터(55, 56) 및 스위치(67 내지 59)가 제어되고, 영상계 데이터가 소정의 디코더로 복호화되고, 비디오 데이터 처리부(96)에서 처리되고, 처리 후의 비디오 데이터가 부호화된다.

[0469] 그리고, 압축 부호화 오디오신호의 출력단자인 압축 오디오신호 인터페이스(52)로부터 스위치(61)로부터 출력되는 부호화 오디오 데이터인 오디오 스트림#1이 외부로 출력되고, 압축 부호화 비디오신호의 출력단자인 압축 비디오신호 인터페이스(54)로부터 부호화된 부호화 비디오 데이터가 외부로 출력된다.

[0470] 즉, 단계(S271)에 있어서 압축 오디오신호 인터페이스(52)로부터 출력되는 압축 부호화 오디오 데이터는 재생장치(20-2)의 AV 디코더부(33-2)에서 복호 처리가 실시되지 않으므로, 음질이 열화하고 있지 않다.

[0471] 단계(S272)에 있어서, 컨트롤러(34-2)는 플레이 리스트를 참조하여 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S272)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S267)로 돌아가고, 그 이후의 처리가 반복된다.

[0472] 단계(S272)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S273)에 있어서, 컨트롤러(34-2)는 재생종료인지 아닌지를 판단한다. 단계(S273)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S263)로 돌아가고, 그 이후의 처리가 반복된다. 즉, 다음에 독출되는 플레이 리스트 파일에 기재되어 있는 플레이그에 의해서, 스위치(61)의 제어가 변경될 가능성이 있다. 단계(S273)에 있어서, 재생종료라고 판단된 경우, 처리는 종료된다.

[0473] 단계(S265)에 있어서, 어느 플래그가 1로 되어 있다고 판단된 경우, 이 플레이 리스트 파일에 관련하는 데이터에는, 재생장치(20-2)에 있어서 오디오 스트림#1에 합성하는 것이 가능한 사운드 데이터 또는 오디오 스트림#2 중 적어도 한쪽이 포함되어 있으므로, 단계(S274)에 있어서, 컨트롤러(34-2)는 이 플레이 리스트 파일에 관련하는 오디오 스트림#1을 제 1 오디오 디코더(75-1)에 공급하도록 스위치(61)를 제어한다.

[0474] 단계(S275) 내지 단계(S279)에 있어서, 도 42의 단계(S217) 내지 단계(S221)와 기본적으로 같은 처리가 실행된다. 즉, 플레이 리스트에 의해 지정되는 플레이 아이템이 독출되고, 독출된 플레이 아이템을 근거로 소정의 클립에 포함되는 데이터가 대응하는 버퍼에 공급되어 버퍼링되고, PID 필터 및 각 스위치가 제어되고, 버퍼링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다.

[0475] 그리고, PID 필터 및 각 스위치가 제어되고, 버퍼(53)에 버퍼링된 오디오 스트림#1이 스위치(101) 및 스위치(61)를 통하여 제 1 오디오 디코더(75-1)에 공급되어 복호화되고, 버퍼(53)에 버퍼링된 오디오 스트림#2가 스위치(101)를 통하여 제 2 오디오 디코더(75-2)에 공급되어 복호화되고, 믹싱 처리부(102)가 제어되고, 제 1 오디오 디코더(75-1) 및 제 2 오디오 디코더(75-2)에서 복호화된 오디오 스트림의 합성처리가 적절히 실행되고, 믹싱 처리부(97)가 제어되고, 믹싱 처리부(102)에 의해 합성된 오디오 스트림과 버퍼(95)에 보관되어 있는 사운

드 데이터와의 합성처리가 적절히 실행된다.

[0476] 즉, 스위치(61)가 제어되고, 오디오 스트림 #1이 오디오 디코더(75)에 의해 복호화되도록 되어 있으므로, 오디오 스트림 #1과 오디오 스트림 #2나 사운드 데이터를 합성하는 것이 가능해진다.

[0477] 그리고, 오디오 엔코더(41)에 의해 합성처리가 실시된 비압축의 오디오 데이터가 부호화되고, 압축 오디오신호 인터페이스(52)로부터 외부 출력되는 동시에, 비디오 엔코더(42)에 의해 비디오 데이터 처리부(96)에 의해 처리된 비압축의 비디오 데이터가 부호화되고, 압축 비디오신호 인터페이스(54)로부터 외부에 출력된다.

[0478] 단계(S280)에 있어서, 컨트롤러(34-2)는 플레이 리스트를 참조하여 재생해야 할 다음의 플레이 아이템이 있는지 아닌지를 판단한다. 단계(S280)에 있어서, 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S275)로 돌아가고, 그 이후의 처리가 반복된다.

[0479] 단계(S280)에 있어서, 다음의 플레이 아이템이 없다고 판단된 경우, 단계(S281)에 있어서, 컨트롤러(34-2)는, 재생종료인지 아닌지를 판단한다. 단계(S281)에 있어서, 재생종료는 아니라고 판단된 경우, 처리는 단계(S263)로 돌아가고, 그 이후의 처리가 반복된다. 즉, 다음에 독출되는 플레이 리스트 파일에 기재되어 있는 플래그에 의해서, 스위치(61)의 제어가 변경될 가능성이 있다. 단계(S281)에 있어서, 재생종료이다고 판단된 경우, 처리는 종료된다.

[0480] 단계(S261)에 있어서, 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 단계(S282) 내지 단계(S291)에 있어서, 도 42의 단계(S224) 내지 단계(S233)와 기본적으로 같은 처리가 실행된다.

[0481] 즉, 출력되는 것은 부호화 스트림이 아닌 경우, 오디오 스트림 #1에 합성되는 사운드 데이터가 있는지 아닌지에 관계없이, 오디오 스트림 #1은 복호화된다.

[0482] 구체적으로는, 플레이 리스트 파일의 독출이 지령되었는지 아닌지가 판단되고, 독출이 지령되었다고 판단된 경우, 플레이 리스트 파일이 독출되어, 그 플레이 리스트 파일에 기재되어 있는 플레이 리스트에 의해 지정되는 플레이 아이템이 독출된다. 그리고, 독출된 플레이 아이템에 대응하는 클립의 데이터(AV Stream)가 대응하는 베퍼에 공급되어 베퍼링되고, PID 필터 및 각 스위치가 제어되고, 베퍼링된 영상계 데이터가 각각 소정의 디코더에서 복호화되고, 비디오 데이터 처리부(96)에서 처리된다. 그리고, PID 필터 및 각 스위치가 제어되어 베퍼(53)에 베퍼링된 오디오 스트림 #1이 스위치(101) 및 스위치(61)를 통하여 제 1 오디오 디코더(75-1)에 공급되어 복호화되고, 오디오 스트림 #2가 스위치(101)를 통하여 제 2 오디오 디코더(75-2)에 공급되어 복호화되고, 믹싱 처리부(102)를 제어하고, 제 1 오디오 디코더(75-1) 및 제 2 오디오 디코더(75-2)에서 복호화된 오디오 스트림이 적절히 합성되고, 믹싱 처리부(97)를 제어하고, 믹싱 처리부(102)에 의해 합성된 오디오 스트림과 베퍼(95)에 보관되어 있는 사운드 데이터가 적절히 합성된다. 그리고, 믹싱 처리부(97)로부터 공급된 처리 후의 비압축의 오디오 데이터가 그래로, 또는, 필요에 따라서 D/A 변환되어 외부로 출력되고, 비디오 데이터 처리부(96)로부터 공급된 비압축의 비디오 데이터가 그대로, 또는, 필요에 따라서 D/A 변환되어 외부로 출력된다.

[0483] 그리고, 플레이 리스트가 참조되고, 재생해야 할 다음의 플레이 아이템이 있는지 아닌지가 판단되어 다음의 플레이 아이템이 있다고 판단된 경우, 처리는 단계(S285)로 돌아가고, 그 이후의 처리가 반복된다. 다음의 플레이 아이템이 없다고 판단된 경우, 재생종료인지 아닌지가 판단되고, 재생종료는 아니라고 판단된 경우, 처리는 단계(S283)로 돌아가고, 그 이후의 처리가 반복된다. 재생종료라고 판단된 경우, 처리는 종료된다.

[0484] 이러한 처리에 의해, 재생장치(20-2)는 플레이 리스트 파일에 기재되어 있는 플래그를 참조하여, 이 플레이 리스트 파일에 의해 참조되는 데이터에 사운드 데이터 또는 오디오 스트림 #2가 어느 것도 포함되어 있지 않은 경우에는, 오디오 스트림 #1을 복호화하지 않고 압축 부호화된 데이터인 채 외부로 출력하므로, 필요 이상으로 음질을 열화시키는 것을 방지할 수 있다.

[0485] 또, 사운드 데이터로서 제공되는 음성 데이터 가운데, 예를 들면, 버튼의 클릭음 등은, 사용자의 조작에 따라 발생하는 효과음이며, 이 경우에는, 오디오 스트림 #1에 합성하는지, 합성을 하지 않아도 되는지는 불분명하다.

따라서, 상술한 재생처리의 흐름(flow)에서는, 사운드 데이터 또는 오디오 스트림이 존재하는지 아닌지를 나타내는 플래그로서 is_MixApp, is_MixApp_1, is_MixApp_2를 설명했으나, 플래그가 나타내는 데이터는 합성을 위해 복호화할 가능성이 없는 것을 나타내는 플래그이면 좋다. 즉, 플래그가 1의 경우여도, 반드시 합성처리가 발생한다고 한정되는 것은 아니고, 합성처리가 발생하지 않는 경우도 있다. 다만, 합성처리가 발생할 수 없는 경우에 플래그가 0으로 세트된다. 또, Indexes()는 플레이 리스트 파일을 복수 지정할 수 있으므로, 플래그를 플레이 리스트로 기재함으로써, 사운드 데이터가 포함되어 있는지 아닌지를 플레이 리스트마다 설정하는 것이 가능해지므로, Indexes()에 플래그 정보를 기재하는 것보다도 유연하게 제어의 설정이 가능해지는 동시에,

오쏘링(authoring)의 자유도가 증가한다.

[0486] 또한, 이 처리에 있어서, 처리의 최초에 있어서, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지 를 판단하고, 그 후, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하는 것으로 하고 있으나, 예를 들면, 우선, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하고, 그 후, 출력되는 신호가 압축 부호화 데이터인지 비압축의 데이터인지를 판단하도록 해도, 판단하는 처리의 순서가 다른 것뿐, 실질적으로 같은 처리가 실행되는 것임은 말할 필요도 없다.

[0487] 다음으로, 도 44의 플로차트(flow chart)를 참조하여, 오디오 스트림 #1에 합성되는 다른 음성 데이터가 있는지 아닌지를 나타내는 플래그가 플레이 아이템에 기재되어 있는 데이터를 재생장치(20-2)가 재생하는 경우에 실행되는 재생처리(6)에 대하여 설명한다.

[0488] 단계(S331)에 있어서, 컨트롤러(34-2)는 재생장치(20-2)로부터 출력되는 것은 부호화 스트림인지 아닌지를 판단 한다. 단계(S331)에 있어서, 재생장치(20-2)로부터 출력되는 것은 부호화 스트림은 아니라고 판단된 경우, 처리는 후술하는 단계(S351)로 진행된다.

[0489] 단계(S331)에 있어서, 재생장치(20-2)로부터 출력되는 것은 부호화 스트림이라고 판단된 경우, 단계(S332)에 있어서, 컨트롤러(34-2)는 스위치(32)로부터 공급되는 인덱스 파일을 독출한다.

[0490] 단계(S333)에 있어서, 컨트롤러(34-2)는 예를 들면, 사용자의 조작입력 등에 의해서 발생되는 커맨드에 의해, 플레이 리스트 파일(예를 들면, 도 13의 $\times \times \times \times \times .m3ls$)의 독출이 지령되었는지 아닌지를 판단한다. 단계(S333)에 있어서, 플레이 리스트 파일의 독출이 지령되어 있지 않다고 판단된 경우, 플레이 리스트 파일의 독출이 지령되었다고 판단될 때까지, 단계(S333)의 처리가 반복된다.

[0491] 단계(S333)에 있어서, 플레이 리스트 파일의 독출이 지령되었다고 판단된 경우, 단계(S334)에 있어서, 컨트롤러(34-2)는 스위치(32)로부터 공급되는 플레이 리스트 파일을 독출한다.

[0492] 단계(S335)에 있어서, 컨트롤러(34-2)는, 그 플레이 리스트 파일에 기재되어 있는 플레이 리스트(도 13의 `Plylist()`)에 의해 지정되는 플레이 아이템을 독출하고, 대응하는 메인 클립 AV 스트림, 서브 클립 AV 스트림 및 텍스트 서브 타이틀 데이터가 독출되어 스위치(32)에 공급한다.

[0493] 단계(S336)에 있어서, 컨트롤러(34-2)는 플레이 아이템에 기재되어 있는 메인 오디오 스트림인 오디오 스트림 #1에 합성되는 데이터가 존재하는 것을 나타내는 플래그 중 어느 한쪽이 1로 되어 있는지 아닌지, 구체적으로는, 사운드 데이터가 존재하는 것을 나타내는 플래그(`is_MixApp` 또는 `is_MixApp_2`), 또는, 오디오 스트림 #2가 존재하는 것을 나타내는 플래그(`is_MixApp_1`) 중 어느 한쪽이 1로 되어 있는지 아닌지를 판단한다.

[0494] `audio_stream_id`로 주어지는 오디오 스트림 #1에 더하여 `audio_stream_id2`로 주어지는 오디오 스트림 #2도 존재하고 있는 컨텐츠에는, 사운드 데이터가 존재하는지 아닌지를 나타내는 플래그인 `is_MixApp_2`에 더하여 오디오 스트림 #2가 존재하는지 아닌지를 나타내는 플래그인 `is_MixApp_1`이 기재되어 있다. 재생장치(20-2)는, `audio_stream_id2`로 주어지는 오디오 스트림 #2를 재생하는 기능을 가지고 있으므로, 단계(S265)에 있어서, `is_MixApp` 또는 `is_MixApp_2`로 나타나는 플래그의 값뿐만 아니라, `is_MixApp_1`로 나타나는 플래그의 값을 참조하도록 되어 있다.

[0495] 단계(S336)에 있어서, 어느 플래그가 1로 되어 있다고 판단된 경우, 처리는 후술하는 단계(S344)로 진행된다.

[0496] 단계(S336)에 있어서, 어느 플래그도 1은 아니라고 판단된 경우, 이 플레이 아이템에 관련하는 오디오 데이터에는, 재생장치(20-2)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터 및 오디오 스트림 #2가 포함되어있지 않으므로, 단계(S337)에 있어서, 컨트롤러(34-2)는 이 플레이 아이템에 관련하는 오디오 데이터를 압축 부호화 오디오신호의 출력단자인 압축 오디오신호 인터페이스(52)에 공급하도록 스위치(61)를 제어한다.

[0497] 단계(S338) 내지 단계(S341)에 있어서, 도 42의 단계(S208) 내지 단계(S211)와 같은 처리가 실행된다. 즉, 독출된 데이터가 대응하는 베퍼에 공급되어 베퍼링되고, PID 필터(55, 56) 및 스위치(67 내지 59)가 제어되고, 영상계 데이터가 소정의 디코더로 복호화되고, 비디오 데이터 처리부(96)에서 처리되어 처리 후의 비디오 데이터가 부호화된다.

[0498] 그리고, 압축 부호화 오디오신호의 출력단자인 압축 오디오신호 인터페이스(52)로부터 스위치(61)로부터 출력되는 부호화 오디오 데이터인 오디오 스트림 #1이 외부에 출력되고, 압축 부호화 비디오신호의 출력단자인 압축 비디오신호 인터페이스(54)로부터 부호화된 부호화 비디오 데이터가 외부에 출력된다.

- [0499] 즉, 단계(S341)에 있어서, 압축 오디오 신호 인터페이스(52)로부터 출력되는 압축 부호화 오디오 데이터는, 재생장치(20-2)의 AV 디코더부(33-2)에 있어서 복호처리가 실시되지 않으므로 음질이 열화되지 않는다.
- [0500] 단계(S342)에 있어서, 컨트롤러(34-2)는 PlayList를 참조하여 재생해야 할 다음의 PlayItem이 있는지 아닌지를 판단한다. 단계(S342)에 있어서, 다음의 PlayItem이 있다고 판단되었을 경우, 처리는 단계(S335)로 돌아가고, 그 이후의 처리가 반복된다. 즉, 다음에 독출되는 PlayItem에 기재되어 있는 플래그에 의해, 스위치(61)의 제어가 변경될 가능성이 있다.
- [0501] 단계(S342)에 있어서, 다음의 PlayItem이 없다고 판단되었을 경우, 단계(S343)에 있어서, 컨트롤러(34-2)는 재생종료인지 아닌지를 판단한다. 단계(S343)에 있어서, 재생종료는 아니라고 판단되었을 경우, 처리는 단계(S333)로 돌아가고, 그 이후의 처리가 반복된다. 단계(S343)에 있어서, 재생종료라고 판단되었을 경우, 처리는 종료된다.
- [0502] 단계(S336)에 있어서, 어느 플래그가 1이 되어 있다고 판단되었을 경우, 이 PlayItem에 관련하는 오디오 데이터에는, 재생장치(20-2)에 있어서 오디오 스트림 #1에 합성하는 것이 가능한 사운드 데이터 또는 오디오 스트림 #2 중 적어도 한쪽이 포함되어 있으므로, 단계(S344)에 있어서, 컨트롤러(34-2)는 이 PlayList 파일에 관련하는 오디오 스트림 #1을 제 1 오디오 디코더(75-1)에 공급하도록 스위치(61)를 제어한다.
- [0503] 단계(S345) 내지 단계(S348)에 있어서, 도 42의 단계(S218) 내지 단계(S221)과 기본적으로 같은 처리가 실행된다. 즉, PlayItem에 대응하는 Clip의 데이터가 대응하는 베퍼에 공급되어 베퍼링 되고, PID 필터 및 각 스위치가 제어되어, 베퍼링된 영상계 데이터가 각각 소정의 디코더에서 디코드되고, 비디오 데이터 처리부(96)에서 처리된다.
- [0504] 그리고, PID 필터 및 각 스위치가 제어되고, 베퍼(53)에 베퍼링 된 오디오 스트림 #1이 스위치(101) 및 스위치(61)를 통하여 제 1 오디오 디코더(75-1)에 공급되어 디코드되고, 베퍼(53)에 베퍼링된 오디오 스트림 #2가 스위치(101)를 통하여 제 2 오디오 디코더(75-2)에 공급되어 디코드되고, 믹싱 처리부(102)가 제어되고, 제 1 오디오 디코더(75-1) 및 제 2 오디오 디코더(75-2)에서 디코드된 오디오 스트림의 합성처리가 적절히 실행되고, 믹싱 처리부(97)가 제어되고, 믹싱 처리부(102)에 의해 합성된 오디오 스트림과 베퍼(95)에 보관되어 있는 사운드 데이터와의 합성처리가 적절히 실행된다.
- [0505] 즉, 스위치(61)가 제어되고, 오디오 스트림 #1이 오디오 디코더(75)에 의해 디코드되도록 되어 있으므로, 오디오 스트림 #1과 오디오 스트림 #2나 사운드 데이터를 합성하는 것이 가능해진다.
- [0506] 그리고, 오디오 엔코더(41)에 의해 합성처리가 실시된 비압축의 오디오 데이터가 부호화되고, 압축 오디오 신호 인터페이스(52)로부터 외부 출력되는 것과 동시에, 비디오 엔코더(42)에 의해 비디오 데이터 처리부(96)에 의해 처리된 비압축의 비디오 데이터가 부호화되고, 압축 비디오 신호 인터페이스(54)로부터 외부로 출력된다.
- [0507] 단계(S349)에 있어서, 컨트롤러(34-2)는 PlayList를 참조하여, 재생해야 할 다음의 PlayItem이 있는지 아닌지를 판단한다. 단계(S349)에 있어서, 다음의 PlayItem이 있다고 판단되었을 경우, 처리는 단계(S335)로 돌아가고, 그 이후의 처리가 반복된다. 즉, 다음에 독출되는 PlayItem에 기재되어 있는 플래그에 의해, 스위치(61)의 제어가 변경될 가능성이 있다.
- [0508] 단계(S349)에 있어서, 다음의 PlayItem이 없다고 판단되었을 경우, 단계(S350)에 있어서, 컨트롤러(34-2)는 재생종료인지 아닌지를 판단한다. 단계(S350)에 있어서, 재생종료는 아니라고 판단되었을 경우, 처리는 단계(S333)로 돌아가고, 그 이후의 처리가 반복된다. 단계(S350)에 있어서, 재생종료라고 판단되었을 경우, 처리는 종료된다.
- [0509] 단계(S331)에 있어서, 출력되는 것은 부호화 스트림은 아니라고 판단되었을 경우, 단계(S351) 내지 단계(S360)에 있어서, 도 42의 단계(S224) 내지 단계(S233)와 기본적으로 같은 처리가 실행된다.
- [0510] 즉, 출력되는 것이 부호화 스트림이 아닌 경우, 오디오 스트림 #1에 합성되는 사운드 데이터가 있는지 아닌지에 관계없이, 오디오 스트림 #1은 디코드된다.
- [0511] 구체적으로는, PlayList 파일의 독출이 지령되었는지 아닌지가 판단되고, 독출이 지령되었다고 판단되었을 경우, PlayList 파일이 독출되어, 그 PlayList 파일에 기재되어 있는 PlayList에 의해 지정되는 PlayItem이 독출된다. 그리고, 독출된 PlayItem에 대응하는 Clip의 데이터(AV Stream)가 대응하는 베퍼에 공급되어 베퍼링 되고, PID 필터 및 각 스위치가 제어되고, 베퍼링된 영상계 데이터가 각각 소정의 디코더에서 디코드되고, 비디오 데이터 처리부(96)에서 처리된다. 그리고, PID 필터 및 각 스위치가 제어되고, 베퍼(53)에 베퍼링 된 오

디오 스트림 #1이 스위치(101) 및 스위치(61)를 통하여 제 1 오디오 디코더(75-1)에 공급되어 디코드되고, 오디오 스트림 #2가 스위치(101)를 통하여 제 2 오디오 디코더(75-2)에 공급되어 디코드되고, 믹싱 처리부(102)를 제어하고, 제 1 오디오 디코더(75-1) 및 제 2 오디오 디코더(75-2)에서 디코드된 오디오 스트림이 적절히 합성되고, 믹싱 처리부(97)를 제어하고, 믹싱 처리부(102)에 의해 합성된 오디오 스트림과 베피(95)에 보관되어 있는 사운드 데이터가 적절히 합성된다. 그리고, 믹싱 처리부(97)로부터 공급된 처리 후의 비압축의 오디오 데이터가 그대로, 또는, 필요에 따라 D/A 변환되어 외부에 출력되고, 비디오 데이터 처리부(96)로부터 공급된 비압축의 비디오 데이터가 그대로, 또는, 필요에 따라 D/A 변환되어 외부에 출력된다.

[0512] 그리고, PlayList가 참조되어 재생해야 할 다음의 PlayItem이 있는지 아닌지가 판단되고, 다음의 PlayItem이 있다고 판단되었을 경우, 처리는 단계(S285)로 돌아가고, 그 이후의 처리가 반복된다. 다음의 PlayItem이 없다고 판단되었을 경우, 재생종료인지 아닌지가 판단되고, 재생종료는 아니라고 판단되었을 경우, 처리는 단계(S283)로 돌아가고, 그 이후의 처리가 반복된다. 재생종료라고 판단되었을 경우, 처리는 종료된다.

[0513] 이러한 처리에 의해, 재생장치(20-2)는 PlayItem에 기재되어 있는 플래그를 참조하고, 이 PlayItem에 의해 참조되는 데이터에 사운드 데이터 또는 오디오 스트림 #2가 어느 것도 포함되어 있지 않은 경우는, 오디오 스트림 #1을 디코드하지 않고 압축 부호화된 데이터인 채 외부에 출력하므로, 필요 이상으로 음질을 열화 시키는 것을 방지할 수 있다. 또, 플래그의 정보를 PlayItem에 기재함으로써, 사운드 데이터가 포함되어 있는지 아닌지를 PlayItem 마다 설정하는 것이 가능해지므로, 오쏘링(authoring)의 자유도가 더욱 늘어난다.

[0514] 또한, 이와 같이, PlayItem에 플래그 정보가 기재되어 있는 경우에는 보다 유연한 제어가 가능해지나, 한편으로는, 1개의 PlayList로 심리스 재생을 보장하면서, PlayItem으로 재부호화의 ON, OFF를 전환한다고 하면, 실제 장치의 구성상, 재생장치의 부담이 커진다. 따라서, 실제 장치의 구성상의 조건에 따라, 플래그 정보는, PlayList 파일 단위로 기록하는 것이 바람직한 경우가 있다.

[0515] 또한, 이 처리에 있어서도, 처리의 최초에 있어서, 출력되는 신호가 압축 부호화데이터인지 비압축 데이터인지 를 판단하고, 그 후, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하는 것으로 하고 있으나, 예를 들면, 우선, 플래그를 참조하여 합성처리가 실행되는지 아닌지를 판단하고, 그 후, 출력되는 신호가 압축 부호화데이터인지 비압축 데이터인지를 판단하도록 하여도, 판단하는 처리의 순서가 다른 것뿐으로, 실질적으로 동일한 처리가 실행되는 것은 말할 필요도 없다.

[0516] 이상에 있어서는, 음성 데이터의 합성에 있어서, 합성처리가 행해지는지 아닌지를 판단하고, 그것을 근거로, 주가 되는 음성 데이터의 디코드 처리를 행하는 경우와 행하지 않는 경우의 각각에 있어서 실행되는 처리에 대하여 설명했으나, 동일한 처리가, 음성 데이터의 합성에 관한 경우뿐만 아니라, 비디오 데이터의 합성에 대해서도 적용가능하다.

[0517] 즉, 예를 들면, 메인 영상 데이터, 즉, 비디오 디코더(72)에서 디코드되는 비디오 데이터에 대하여 합성되는, 백그라운드 플레이어, 프레젠테이션 그래픽 플레이어, 인터랙티브 그래픽 플레이어가 스트림 내에 존재하는지 아닌지를 나타내는 플래그를, 상술한 경우와 마찬가지로, Indexes, PlayList 또는 PlayItem의 어느 쪽에 기재한다. 도 45에 나타나는 재생장치(20-3)의 AV 디코더부(33-3)는, 또한 스위치(151)를 설치하고, 스위치(151)는, PID 필터(55)로부터 공급된 메인 영상 데이터를, 컨트롤러(34-3)의 제어에 근거하여, 비디오 디코더(72), 또는, 압축 비디오 신호 인터페이스(54)의 어느 쪽에 공급한다. 이것이에 의해, 메인 영상 데이터가 압축 부호화데이터로서 출력되고, 또, 메인 영상 데이터에 대하여, 백그라운드 플레이어, 프레젠테이션 그래픽 플레이어, 인터랙티브 그래픽 플레이어가 어느 것도 합성되지 않을 때, 메인 영상 데이터를 디코드하지 않고 출력할 수 있으므로, 불필요한 화상의 열화를 방지하는 것이 가능해진다.

[0518] 다음으로, 도 46 및 도 47을 참조하여, 재생장치(20)에서 재생가능한 데이터가 기록된 기록매체(21)의 제조방법에 대하여, 기록매체(21)가 디스크형 기록매체인 경우를 예로서 설명한다.

[0519] 즉, 도 46에 나타낸 바와 같이, 예를 들면 유리 등으로 이루어지는 원반이 준비되고, 그 위에, 예를 들면 포토레지스트(photoresist) 등으로 이루어지는 기록재료가 도포된다. 이것에 의해, 기록용 원반이 제작된다.

[0520] 그리고, 도 47에 나타낸 바와 같이, 소프트 제작 처리부에 있어서, 부호화장치(비디오 엔코더)로 부호화된, 재생장치(20)에서 재생가능한 형식의 비디오 데이터가 일시 베피에 기억되고, 오디오 엔코더로 부호화된 오디오 데이터가 일시 베피에 기억되는 것과 동시에, 또한, 데이터 엔코더로 부호화된 스트림 이외의 데이터(예를 들면, Indexes, Playlist, PlayItem 등)가 일시 베피에 기억된다. 각각의 베피에 기억된 비디오 데이터, 오디오 데이터 및 스트림 이외의 데이터는, 다중화기(MPX)로 동기신호와 함께 다중화되고, 오류정정코드회로(EC

C)에서 여러수정용의 코드가 부가된다. 그리고, 변조회로(MOD)에서 소정의 변조가 이루어지고, 소정의 포맷에 따라, 예를 들면 자기테이프 등에 일단 기록되어, 재생장치(20)에서 재생가능한 기록매체(21)에 기록되는 소프트웨어가 제작된다.

[0521] 이 소프트웨어를 필요에 따라 편집(프리마스터링)하여, 광디스크에 기록해야 할 포맷의 신호가 생성된다. 그리고, 도 46에 나타낸 바와 같이, 이 기록신호에 대응하여 레이저 빔이 변조되고, 이 레이저 빔이 원반상의 포토레지스트 (photoresist) 상에 조사된다. 이것에 의해, 원반상의 포토레지스트 (photoresist)가 기록신호에 대응하여 노광된다.

[0522] 그 후, 이 원반을 현상하고, 원반상에 퍼트를 출현시킨다. 이와 같이 하여 준비된 원반에, 예를 들면 전기주조 등의 처리를 실시하고, 유리 원반상의 퍼트를 전사한 금속원반을 제작한다. 이 금속 원반으로부터, 더욱이 금속 스템퍼를 제작하고, 이것을 성형용 금형으로 한다.

[0523] 이 성형용 금형에, 예를 들면 인젝션 등에 의해 PMMA(아크릴) 또는 PC(폴리카보네이트(polycarbonate))등의 재료를 주입하고, 고정화시킨다. 또는, 금속 스템퍼 상에 2P(자외선 경화수지) 등을 도포한 후, 자외선을 조사하여 경화시킨다. 이것에 의해, 금속 스템퍼상의 퍼트를 수지로 이루어지는 레플리카상에 전사할 수 있다.

[0524] 이와 같이 하여 생성된 레플리카상에, 반사막이 증착 또는 스퍼터링 등에 의해 형성된다. 또는, 생성된 레플리카상에, 반사막이 스팬코트에 의해 형성된다.

[0525] 그 후, 이 디스크에 대하여 내외지름의 가공이 실시되고, 2매의 디스크를 접착시키는 등의 필요한 처리가 실시된다. 더욱이, 라벨을 붙이고, 헤브가 장착되어, 카트리지에 삽입된다. 이와 같이 하여 재생장치(20)에 의해 재생가능한 데이터가 기록된 기록매체(21)가 완성된다.

[0526] 상술한 일련의 처리는, 하드웨어에 의해 실행시킬 수도 있고, 소프트웨어에 의해 실행시킬 수도 있다. 이 경우, 상술한 처리는, 도 48에 나타내는 퍼스널 컴퓨터(500)에 의해 실행된다.

[0527] 도 48에 있어서, CPU(Central Processing Unit)(501)는, ROM(Read Only Memory)(502)에 기억되어 있는 프로그램, 또는, 기억부(508)로부터 RAM(Random Access Memory)(503)에 로드된 프로그램에 따라 각종의 처리를 실행한다. RAM(503)에는, 또한, CPU(501)가 각종의 처리를 실행하는데 필요한 데이터 등이 적절하게 기억된다.

[0528] CPU(501), ROM(502) 및 RAM(503)은, 내부 버스(504)를 통하여 서로 접속되어 있다. 이 내부 버스(504)에는 또한, 입출력 인터페이스(505)도 접속되어 있다.

[0529] 입출력 인터페이스(505)에는, 키보드, 마우스 등으로 이루어지는 입력부(506), CRT, LCD 등으로 이루어지는 디스플레이, 스피커 등으로 이루어지는 출력부(507), 하드디스크 등으로 구성되는 기억부(508) 및 모뎀, 터미널 어댑터 등으로 구성되는 통신부(509)가 접속되어 있다. 통신부(509)는, 전화회선이나 CATV를 포함하는 각종 네트워크를 통하여 통신처리를 행한다.

[0530] 입출력 인터페이스(505)에는 또한, 필요에 따라 드라이브(510)가 접속되고, 자기디스크, 광디스크, 광자기디스크, 또는 반도체 메모리 등으로 이루어지는 리무버블 미디어(521)가 적절히 장착되고, 그것으로부터 독출된 컴퓨터 프로그램이 필요에 따라 기억부(508)에 인스톨된다.

[0531] 일련의 처리를 소프트웨어에 의해 실행시키는 경우에는, 그 소프트웨어를 구성하는 프로그램이 네트워크나 프로그램 저장매체로부터 인스톨된다.

[0532] 이 프로그램 저장매체는, 도 48에 나타낸 바와 같이, 컴퓨터와는 별도로 사용자에게 프로그램을 제공하기 위해서 배포되는, 프로그램이 기록되어 있는 리무버블 미디어(521)로 이루어지는 패키지 미디어에 의해 구성될 뿐만 아니라, 장치 본체에 미리 설치된 상태로 사용자에게 제공되는, 프로그램이 기록되어 있는 ROM(502)이나 기억부(508)가 포함되는 하드디스크 등으로 구성된다.

[0533] 또한, 본 명세서에 있어서, 컴퓨터 프로그램을 기술하는 단계는, 기재된 순서에 따라서 시계열적으로 행해지는 처리는 물론, 반드시 시계열적으로 처리되지 않아도, 병렬적 또는 개별적으로 실행되는 처리도 포함하는 것이다.

[0534] 또, 드라이브(510)는, 장착된 리무버블 미디어(521)에 기록된 데이터를 독출할 수 있을 뿐만 아니라, 장착된 리무버블 미디어(521)에 데이터를 기록할 수 있다. 그리고, 퍼스널 컴퓨터(500)가, 도 47을 이용하여 설명한 소프트 제작처리부와 같은 기능을 가지는(예를 들면, CPU(501)를 이용하여, 소프트 제작처리부와 동일한 기능을 실현하기 위한 프로그램을 실행하는) 것이 가능함은 말할 필요도 없다.

[0535] 즉, 퍼스널 컴퓨터(500)는, 도 47을 이용하여 설명한 소프트 제작처리부에 의해 생성된 데이터와 같은 데이터를 CPU(501)의 처리에 의해 생성하거나, 또는, 외부의 장치로 생성된 도 47을 이용하여 설명한 소프트 제작처리부에 의해 생성된 데이터와 같은 데이터를, 통신부(509) 또는 드라이브(510)에 장착된 리무버블 미디어(521)를 통하여 취득할 수 있다. 그리고, 퍼스널 컴퓨터(500)는, 생성 또는 취득된 도 47을 이용하여 설명한 소프트 제작처리부에 의해 생성된 데이터와 같은 데이터를, 드라이브(510)에 장착된 리무버블 미디어(521)에 기록하는 기록장치로서의 기능을 실현할 수 있다.

[0536] 또한, 본 발명의 실시의 형태는, 상술한 실시의 형태로 한정되는 것이 아니라, 본 발명의 요지를 벗어나지 않는 범위에서 여러 가지의 변경이 가능하다.

발명의 효과

[0537] 본 발명의 제 1 측면에 따르면, 스트림 데이터를 재생할 수 있고, 특히, 재생데이터에 스트림 데이터에 합성되는 합성용 데이터가 포함되어 있지 않다고 판별되고, 또, 출력되는 데이터가 부호화데이터인 경우, 스트림 데이터를 복호 및 재부호화하지 않고 출력할 수 있다.

[0538] 또, 본 발명의 제 2 측면에 따르면, 스트림 데이터의 재생순서를 관리하는 제 1 정보를 포함하고, 제 1 정보는, 스트림 데이터에 합성되는 스트림 데이터와는 다른 합성용 데이터의 유무에 근거하는 제 2 정보를 포함하는데 데이터구조를 가지는 데이터를 제공할 수 있어, 이것에 의해, 이러한 데이터구조를 가지는 데이터를 취득한 재생장치에 있어서, 재생데이터에 스트림 데이터에 합성되는 합성용 데이터가 포함되어 있지 않다고 판별되고, 또, 출력되는 데이터가 부호화데이터인 경우, 스트림 데이터를 복호 및 재부호화하지 않고 출력할 수 있다.

도면의 간단한 설명

[0001] 도 1은 종래의 음성합성을 설명하는 도면이다.

[0002] 도 2는 본 발명을 적용한 재생장치에 대하여 설명하기 위한 도면이다.

[0003] 도 3은 본 발명을 적용한 재생장치에 장착되는 기록매체 상의 애플리케이션 포맷의 예를 나타내는 도면이다.

[0004] 도 4는 인덱스 테이블과 내비게이션 오브젝트에 대하여 설명하는 도면이다.

[0005] 도 5는 메인패스(main path)와 서브패스(sub path)의 구조를 설명하는 도면이다.

[0006] 도 6은 메인패스와 서브패스의 예를 설명하는 도면이다.

[0007] 도 7은 메인패스와 서브패스의 다른 예를 설명하는 도면이다.

[0008] 도 8은 재생장치로 재생가능한 데이터의 데이터 구성예를 설명하는 도면이다.

[0009] 도 9는 index.bdmv의 구문(syntax)을 나타내는 도면이다.

[0010] 도 10은 indexes 구문을 나타내는 도면이다.

[0011] 도 11은 indexes 구문을 나타내는 도면이다.

[0012] 도 12는 indexes 구문을 나타내는 도면이다.

[0013] 도 13은 PlayList 파일의 데이터구조에 대하여 설명하기 위한 도면이다.

[0014] 도 14는 AppInfoPlayList()의 구문에 대하여 설명하기 위한 도면이다.

[0015] 도 15는 AppInfoPlayList()의 구문에 대하여 설명하기 위한 도면이다.

[0016] 도 16은 PlayList()의 구문을 나타내는 도면이다.

[0017] 도 17은 PlayList()의 구문을 나타내는 도면이다.

[0018] 도 18은 PlayList()의 구문을 나타내는 도면이다.

[0019] 도 19는 SubPath()의 구문을 나타내는 도면이다.

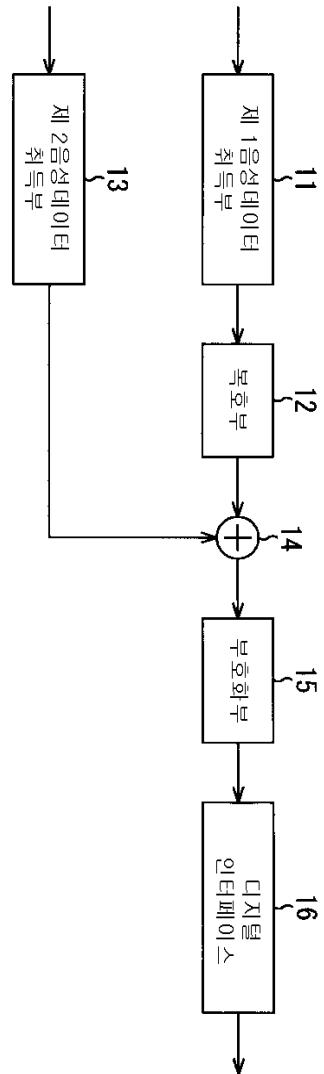
[0020] 도 20은 SubPath_type을 설명하는 도면이다.

[0021] 도 21은 SubPlayItem(i)의 구문을 나타내는 도면이다.

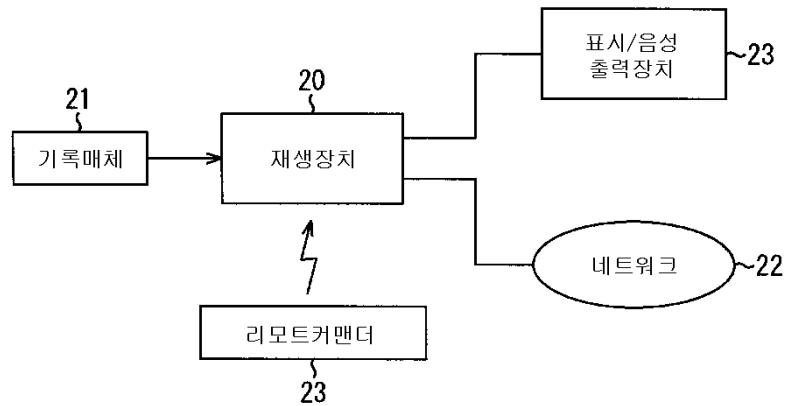
- [0022] 도 22는 PlayItem()의 구문을 나타내는 도면이다.
- [0023] 도 23은 PlayItem()의 구문을 나타내는 도면이다.
- [0024] 도 24는 PlayItem()의 구문을 나타내는 도면이다.
- [0025] 도 25는 STN_table()의 구문을 나타내는 도면이다.
- [0026] 도 26은 stream_entry()의 구문을 나타내는 도면이다.
- [0027] 도 27은 stream_attribute()의 구문을 나타내는 도면이다.
- [0028] 도 28은 stream_cording_type을 설명하는 도면이다.
- [0029] 도 29는 video_format을 설명하는 도면이다.
- [0030] 도 30은 frame_rate를 설명하는 도면이다.
- [0031] 도 31은 aspect_ratio를 설명하는 도면이다.
- [0032] 도 32는 audio_presentation_type을 설명하는 도면이다.
- [0033] 도 33은 sampling_frequency를 설명하는 도면이다.
- [0034] 도 34는 Character code를 설명하는 도면이다.
- [0035] 도 35는 사용자에게 제공하는 음성신호와 자막신호의 관계를 나타내는 스트림 넘버 테이블의 예를 설명하는 도면이다.
- [0036] 도 36은 sound.bdmv의 구문을 나타내는 도면이다.
- [0037] 도 37은 본 발명을 적용한 재생장치의 제 1 구성예를 나타내는 블럭도이다.
- [0038] 도 38은 재생처리 1을 설명하기 위한 플로차트(flow chart)이다.
- [0039] 도 39는 재생처리 2를 설명하기 위한 플로차트이다.
- [0040] 도 40은 재생처리 3을 설명하기 위한 플로차트이다.
- [0041] 도 41은 본 발명을 적용한 재생장치의 제 2 구성예를 나타내는 블럭도이다.
- [0042] 도 42는 재생처리 4를 설명하기 위한 플로차트이다.
- [0043] 도 43은 재생처리 5를 설명하기 위한 플로차트이다.
- [0044] 도 44는 재생처리 6을 설명하기 위한 플로차트이다.
- [0045] 도 45는 본 발명을 적용한 재생장치의 제 3 구성예를 나타내는 블럭도이다.
- [0046] 도 46은 재생장치로 재생가능한 데이터를 기록한 기록매체의 제조에 대하여 설명하기 위한 도면이다.
- [0047] 도 47은 재생장치로 재생가능한 데이터를 기록한 기록매체의 제조에 대하여 설명하기 위한 도면이다.
- [0048] 도 48은 퍼스널컴퓨터의 구성을 나타내는 도면이다.

도면

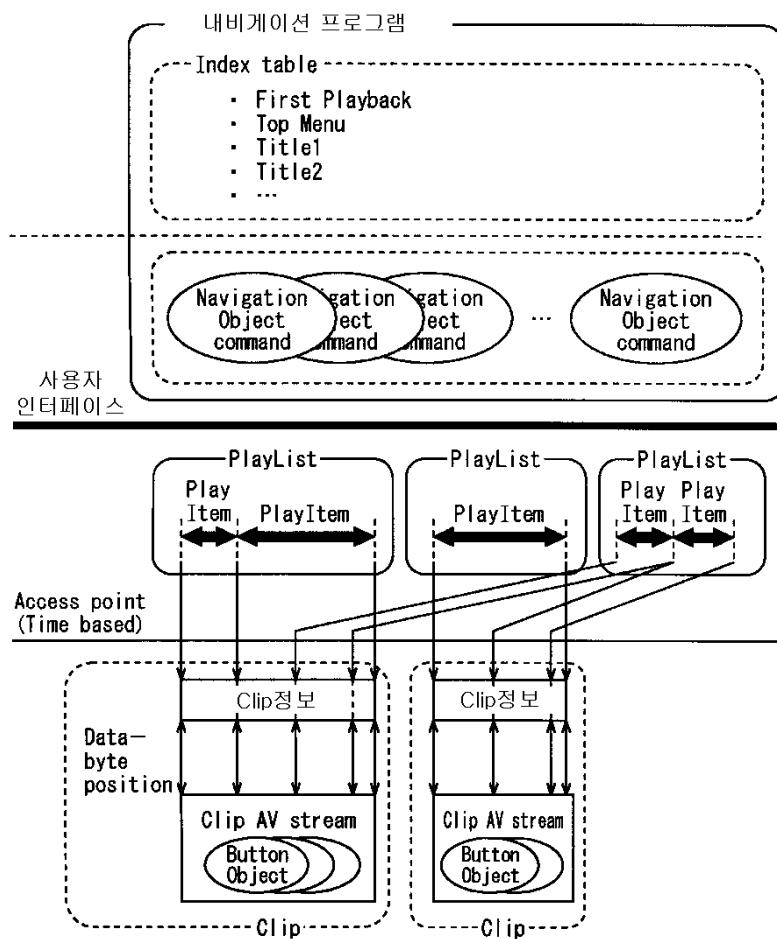
도면1



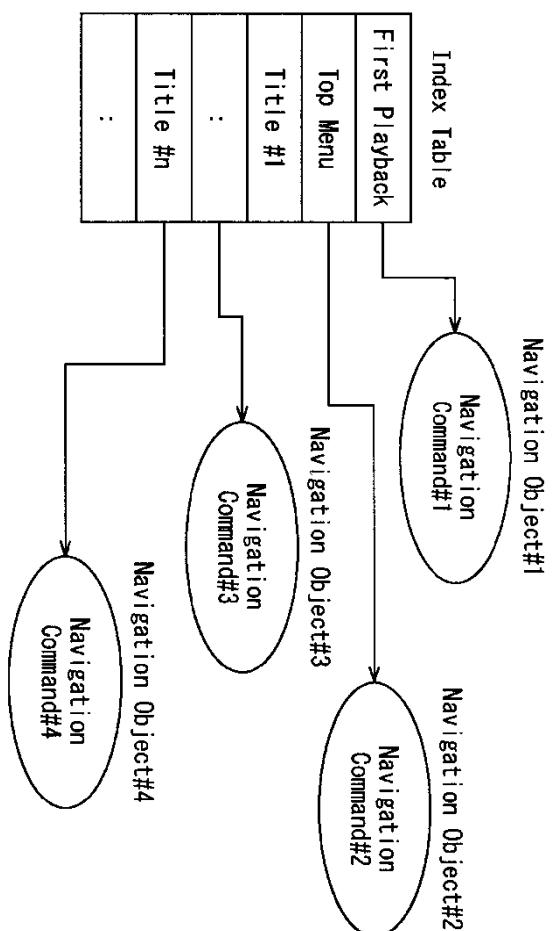
도면2



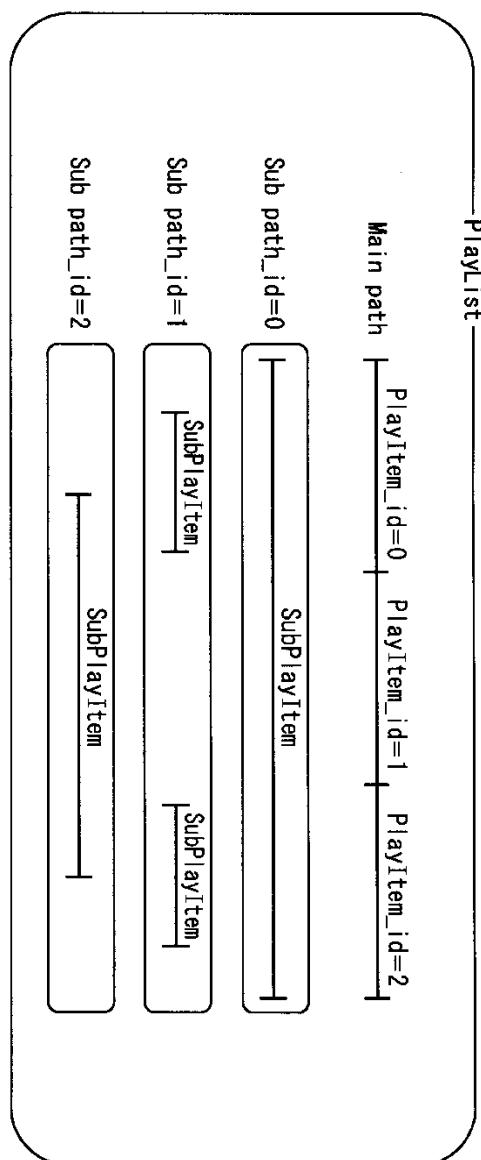
도면3



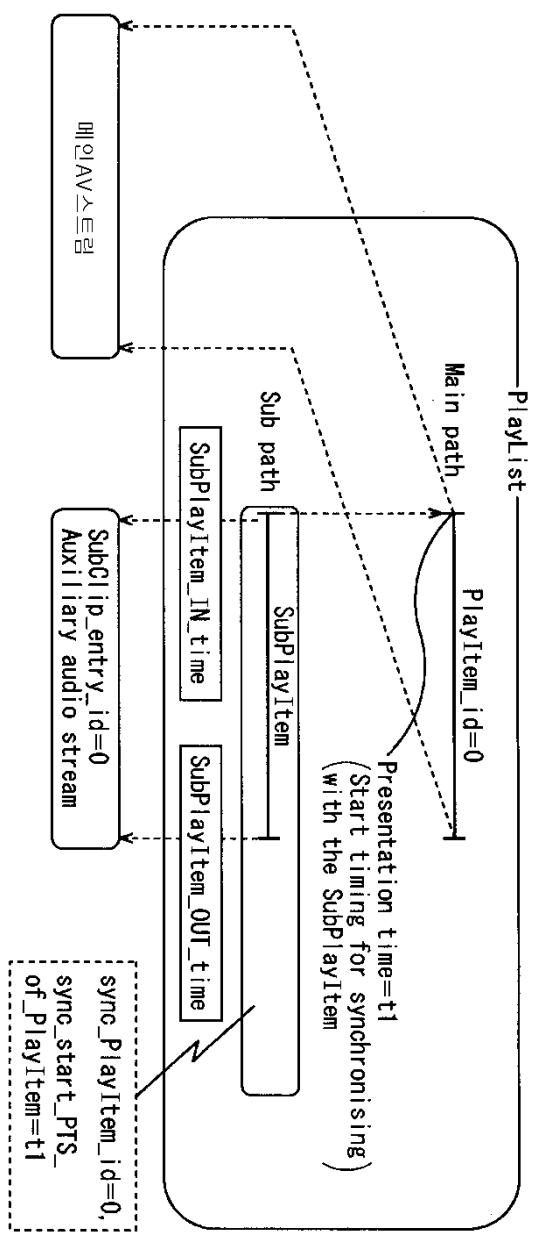
도면4



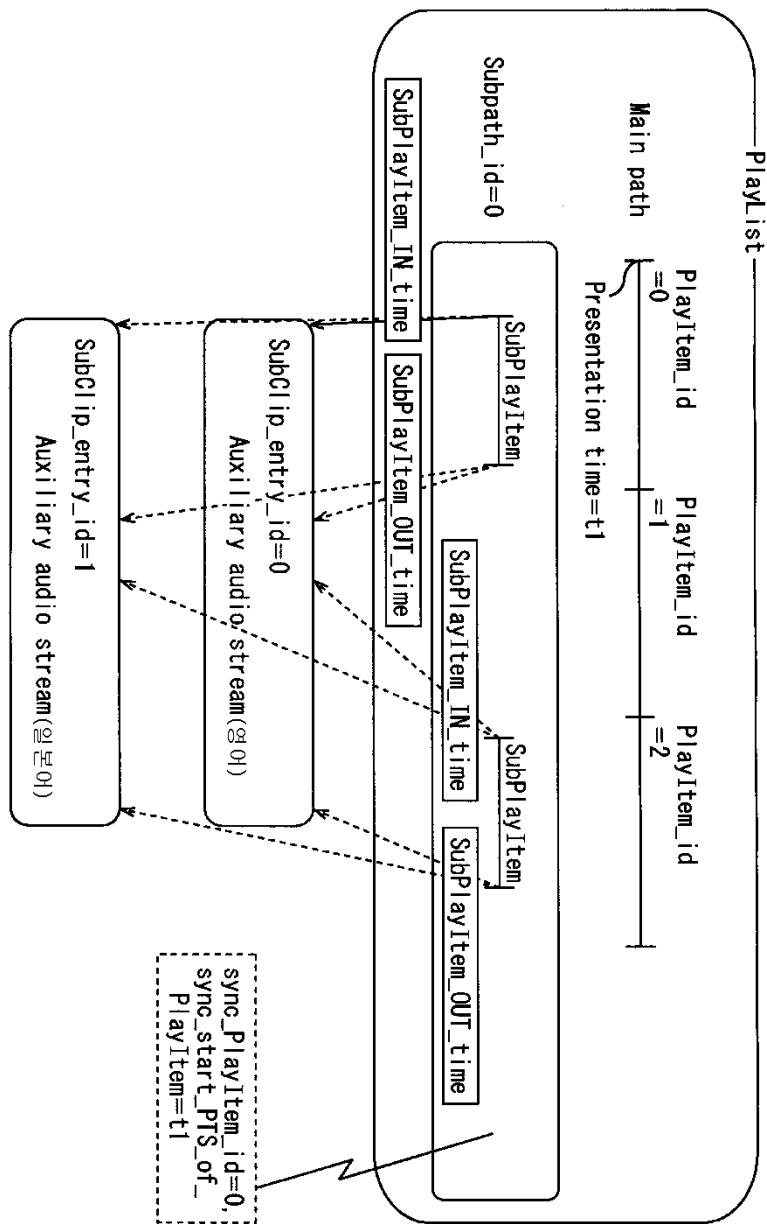
도면5

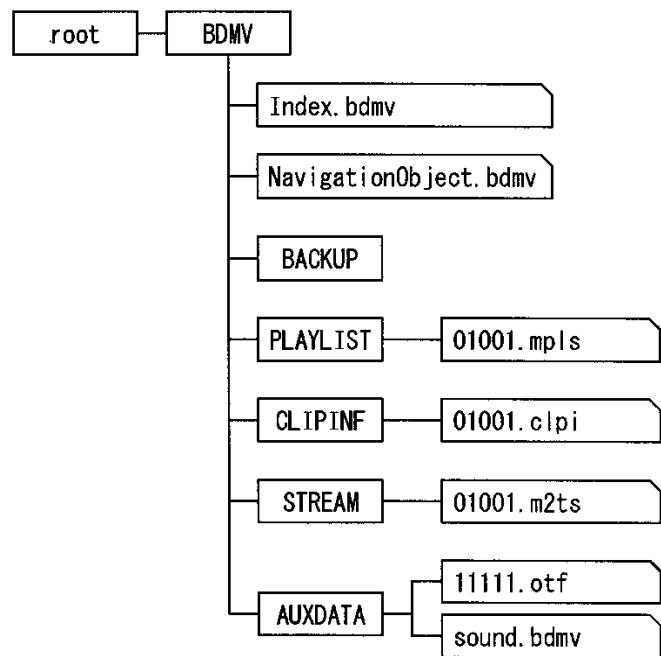


도면6



도면7



도면8

도면9

Syntax	No. of bits	Mnemonic
index bdmv [
type_indicator	8*4	bslbf
version_number	8*4	bslbf
Indexes_start_address	32	uimsbf
reserved_for_future_use	224	bslbf
AppInfoDBMV()		
for(i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
Indexes()		
for(i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
]		

도면10

Syntax	No. of bits	Mnemonic
indexes()		
length	32	uimsbf
FirstPlayback()		
reserved_for_future_use	32	bs\bf
FirstPlayback_mobj_id_ref	16	uimsbf
}		
TopMenu()		
reserved_for_future_use	32	bs\bf
TopMenu_mobj_id_ref	16	uimsbf
}		
number_of_Titles	16	uimsbf
for(title_id=0;		
title_id<number_of_Titles;		
title_id++{		
Title[title_id]()		
Title_playback_type[title_id]	1	uimsbf
Title_access_type[title_id]	2	uimsbf
reserved_for_future_use	29	bs\bf
Title_mobj_id_ref[title_id]	16	uimsbf
}		
}		

도면11

Syntax	No. of bits	Mnemonic
indexes()		
length	32	uimsbf
FirstPlayback()		
reserved_for_future_use	32	bslbf
FirstPlayback_mobj_id_ref	16	uimsbf
}		
TopMenu()		
reserved_for_future_use	32	bslbf
TopMenu_mobj_id_ref	16	uimsbf
}		
number_of_Titles	16	uimsbf
for(title_id=0;		
title_id<number_of_Titles;		
title_id++{		
Title[title_id]()		
Title_playback_type[title_id]	1	uimsbf
Title_access_type[title_id]	2	uimsbf
is_MixApp	1	uimsbf
reserved_for_future_use	28	bslbf
Title_mobj_id_ref[title_id]	16	uimsbf
}		
}		

도면12

Syntax	No. of bits	Mnemonic
indexes()		
length	32	uimsbf
FirstPlayback()		
reserved_for_future_use	32	bs lbf
FirstPlayback_mobj_id_ref	16	uimsbf
}		
TopMenu()		
reserved_for_future_use	32	bs lbf
TopMenu_mobj_id_ref	16	uimsbf
}		
number_of_Titles	16	uimsbf
for (title_id=0;		
title_id<number_of_Titles;		
title_id++		
Title[title_id]		
Title[title_id].		
Title.playback_type[title_id]	1	uimsbf
Title.access_type[title_id]	2	uimsbf
is_MixApp_1	1	uimsbf
is_MixApp_2	1	uimsbf
reserved_for_future_use	27	bs lbf
Title_mobj_id_ref[title_id]	16	uimsbf
}		
}		
}		

도면13

Syntax	No. of bits	Mnemonic
xxxx.mpls{		
type_indicator	8*4	bslbf
version_number	8*4	bslbf
PlayList_start_address	32	uimsbf
PlayListMark_start_address	32	uimsbf
ExtensionData_start_address	32	uimsbf
reserved_for_future_use	160	bslbf
AppInfoPlayList()		
for(i=0; i<N1; i++) {		
padding_word	16	bslbf
}		
PlayList()		
for(i=0; i<N2; i++) {		
padding_word	16	bslbf
}		
PlayListMark()		
for(i=0; i<N3; i++) {		
padding_word	16	bslbf
}		
ExtensionData()		
for(i=0; i<N4; i++) {		
padding_word	16	bslbf
}		
}		

도면14

Syntax	No. of bits	Mnemonic
AppInfoPlayList()		
length	32	uimsbf
reserved_for_future_use	8	bs lbf
PlayList_playback_type	8	bs lbf
if(PlayList_playback_type==2 PlayList_playback_type==3){		
playback_count	16	uimsbf
}else{		
reserved_for_future_use	16	bs lbf
}		
U0_mask_table()		
PlayList_random_access_flag	1	bs lbf
Is_MixApp	1	bs lbf
lossless_may_bypass_mixer_flag	1	bs lbf
reserved_for_future_use	13	bs lbf
}		

도면15

Syntax	No. of bits	Mnemonic
AppInfoPlayList()		
length	32	ui <u>nsbf</u>
reserved_for_future_use	8	bs bf
PlayList_playback_type	8	bs bf
if(PlayList_playback_type==2 PlayList_playback_type==3){		
playback_count	16	ui <u>nsbf</u>
}		
} reserved_for_future_use	16	bs bf
}		
U0_mask_table()		
PlayList_random_access_flag	1	bs bf
lossless_may_bypass_mixer_flag	1	bs bf
reserved_for_future_use	13	bs bf
}		

도면16

PlayList-Syntax		No. of bits	Mnemonic
Syntax			
PlayList()			
length	32	uimsbf	
reserved_for_future_use	16	bslbf	
number_of_playItems	16	uimsbf	
number_of_SubPaths	16	uimsbf	
for (PlayItem_id=0;			
PlayItem_id<number_of_PlayItems;			
PlayItem_id++) {			
PlayItem()			
}			
for (SubPath_id=0;			
SubPath_id<number_of_SubPaths;			
SubPath_id++) {			
SubPath()			
}			
}			

도면17

PlayList-Syntax		
Syntax	No. of bits	Mnemonic
PlayList()		
length	32	uimsbf
is_MixApp	1	uimsof
reserved_for_future_use	15	bslbf
number_of_PlayItems	16	uimsof
number_of_SubPaths	16	uimsbf
for (PlayItem_id=0;		
PlayItem_id<number_of_PlayItems;		
PlayItem_id++) {		
PlayItem()		
}		
}		
for (SubPath_id= 0;		
SubPath_id<number_of_SubPaths;		
SubPath_id++) {		
SubPath()		
}		
}		

도면18

PlayList-Syntax		
Syntax	No. of bits	Mnemonic
PlayList()		
length	32	ui _{msbf}
is_MixApp_1	1	ui _{msbf}
is_MixApp_2	1	ui _{msbf}
reserved_for_future_use	14	bs\bf
number_of_PlayItems	16	ui _{msbf}
number_of_SubPaths	16	ui _{msbf}
for(PlayItem_id=0;		
PlayItem_id<number_of_PlayItems;		
PlayItem_id++) {		
PlayItem()		
}		
}		
for(SubPath_id=0;		
SubPath_id<number_of_SubPaths;		
SubPath_id++) {		
SubPath()		
}		
}		

도면19

SubPath-Syntax			
Syntax	No. of bits	Mnemonic	
SubPath()			
length	32	uimsbf	
reserved_for_future_use	8	bslbf	
SubPath_type	8	uiamsbf	
reserved_for_future_use	15	uiamsbf	
is_repeat_SubPath	1	bslbf	
reserved_for_future_use	8	bslbf	
number_of_SubPlayItems	8	uiamsbf	
for (i=0;i< number_of_SubPlayItems;i++) {			
SubPlayItem(i)			
}			
}			

도면20

SubPath_type	Meaning
0	reserved
1	reserved
2	Audio presentation path of the Browsable slideshow (The audio presentation path using the SubPath is not synchronized with the main path using PlayItems in the PlayList.)
3	Interactive graphics presentation menu (The interactive graphics menu using the SubPath is not synchronized with the main path using PlayItems in the PlayList.)
4	Text subtitle presentation path (The text subtitle presentation path using the SubPath is synchronized with the main path using PlayItems in the PlayList.)
5	2nd Audio Presentation path (The 2nd Audio Presentation path using the SubPath is synchronized with the main path using PlayItems in the PlayList.)
6	2nd Video Presentation path (The 2nd Video Presentation path using the SubPath is synchronized with the main path using PlayItems in the PlayList.)
7~255	reserved

도면21

Syntax	No. of bits	Mnemonic
SubPlayItem(i){		
length	16	uiu16bf
Clip_Information_file_name[0] //subclip_entry_id=0	8*5	bs1bf
Clip_codec_identifier[0]	8*4	bs1bf
reserved_for_future_use	31	bs1bf
is_multi_Clip_entries	1	bs1bf
ref_to_STC_id[0]	8	uiu8bf
SubPlayItem_IN_time	32	uiu16bf
SubPlayItem_OUT_time	32	uiu16bf
sync_PlayItem_id	16	uiu16bf
sync_start_PTS_of_PlayItem	32	uiu16bf
if(is_multi_Clip_entries==1b){		
reserved_for_future_use	8	bs1bf
num_of_Clip_entries	8	uiu8bf
for(subclip_entry_id=1://Note:Entries after subclip_entry_id=0 subclip_entry_id<num_of_Clip_entries;subclip_entry_id++){		
Clip_Information_file_name[subclip_entry_id]	8*5	bs1bf
Clip_codec_identifier[subclip_entry_id]	8*4	bs1bf
ref_to_STC_id[subclip_entry_id]	8	uiu8bf
reserved_for_future_use	8	bs1bf
}		
}		

도면22

PlayItem-Syntax

Syntax	No. of bits	Mnemonic
PlayItem()		
length	16	uimsbf
Clip_Information_file_name[0]	8*5	bslbf
Clip_codec_identifier[0]	8*4	bslbf
reserved_for_future_use	11	bslbf
is_multi_angle	1	bslbf
connection_condition	4	uimsbf
ref_to_STC_id[0]	8	uimsbf
IN_time	32	uimsbf
OUT_time	32	uimsbf
UO_mask_table()		
PlayItem_random_access_mode	8	uimsbf
still_mode	8	uimsbf
if(still_mode==0x1){		
still_time	16	uimsbf
}else{		
reserved	16	bslbf
}		
if(is_multi_angle==1b){		
number_of_angles	8	uimsbf
reserved_for_future_use	7	bslbf
is_seamless_angle_change	1	uimsbf
for(angle_id = 1; //Note: angles after angle_id=1		
angle_id<number_of_angles; angle_id++) {		
Clip_Information_file_name[angle_id]	8*5	bslbf
Clip_codec_identifier[angle_id]	8*4	bslbf
ref_to_STC_id[angle_id]	8	uimsbf
reserved_for_future_use	16	bslbf
}		
}		
STN_table()		
}		

도면23

PlayItem-Syntax

Syntax	No. of bits	Mnemonic
PlayItem()		
length	16	uimsbf
Clip_Information_file_name[0]	8*5	bslbf
Clip_codec_identifier[0]	8*4	bslbf
is_MixApp	1	uimsbf
reserved_for_future_use	10	bslbf
is_multi_angle	1	bslbf
connection_condition	4	uimsbf
ref_to_STC_id[0]	8	uimsbf
IN_time	32	uimsbf
OUT_time	32	uimsbf
U0_mask_table()		
PlayItem_random_access_mode	8	uimsbf
still_mode	8	uimsbf
if(still_mode==0x1){		
still_time	16	uimsbf
}else{		
reserved	16	bslbf
}		
if(is_multi_angle==1b){		
number_of_angles	8	uimsbf
reserved_for_future_use	7	bslbf
is_seamless_angle_change	1	uimsbf
for(angle_id = 1; //Note: angles after angle_id=1		
angle_id<number_of_angles; angle_id++) {		
Clip_Information_file_name[angle_id]	8*5	bslbf
Clip_codec_identifier[angle_id]	8*4	bslbf
ref_to_STC_id[angle_id]	8	uimsbf
reserved_for_future_use	16	bslbf
}		
}		
STN_table()		
}		

도면24

PlayItem-Syntax

Syntax	No. of bits	Mnemonic
PlayItem0 {		
length	16	uimsbf
Clip_Information_file_name[0]	8*5	bslbf
Clip_codec_identifier[0]	8*4	bslbf
is_MixApp_1	1	uimsbf
is_MixApp_2	1	uimsbf
reserved_for_future_use	9	bslbf
is_multi_angle	1	bslbf
connection_condition	4	uimsbf
ref_to_STC_id[0]	8	uimsbf
IN_time	32	uimsbf
OUT_time	32	uimsbf
U0_mask_table()		
PlayItem_random_access_mode	8	uimsbf
still_mode	8	uimsbf
if(still_mode==0x1){		
still_time	16	uimsbf
}else{		
reserved	16	bslbf
}		
if(is_multi_angle==1b){		
number_of_angles	8	uimsbf
reserved_for_future_use	7	bslbf
is_seamless_angle_change	1	uimsbf
for(angle_id = 1; //Note: angles after angle_id=1		
angle_id<number_of_angles; angle_id++) {		
Clip_Information_file_name[angle_id]	8*5	bslbf
Clip_codec_identifier[angle_id]	8*4	bslbf
ref_to_STC_id[angle_id]	8	uimsbf
reserved_for_future_use	16	bslbf
}		
}		
STN_table()		
}		

도면25

STN_table()

Syntax	No. of bits	Mnemonic
STN_table()		
length	16	uimsbf
reserved_for_future_use	16	bslbf
number_of_video_stream_entries	8	uimsbf
number_of_audio_stream_entries	8	uimsbf
number_of_audio_stream2_entries	8	uimsbf
number_of_PG_textST_stream_entries	8	uimsbf
number_of_IG_stream_entries	8	uimsbf
reserved_for_future_use	64	bslbf
for (video_stream_id=0; video_stream_id < number_of_video_stream_entries; video_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
for (audio_stream_id=0; audio_stream_id < number_of_audio_stream_entries; audio_stream_id++) {		
stream_entry()		
stream_attributes()		
}		
for (audio_stream_id2=0; audio_stream_id2 < number_of_audio_stream2_entries; audio_stream_id2++) {		
stream_entry()		
stream_attributes()		
}		
for (PG_textST_stream_id=0; PG_textST_stream_id < number_of_PG_textST_stream_entries; PG_textST_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
for (IG_stream_id=0; IG_stream_id < number_of_IG_stream_entries; IG_stream_id++) {		
stream_entry()		
stream_attribute()		
}		
}		

도면26

Stream_entry()	No. of bits	Mnemonic
Syntax		
stream_entry(id) {		
length	8	uimsbf
type	8	uimsbf
if(type==1) {		
ref_to_stream_PID_of_mainClip	16	uimsbf
reserved_for_future_use	48	Bs bf
} else if(type==2) {		
ref_to_SubPath_id	8	Uimsbf
ref_to_subClip_entry_id	8	Uimsbf
ref_to_stream_PID_of_subClip	16	
reserved_for_future_use	32	bs bf
}		
}		

도면27

stream_attribute()

Syntax	No. of bits	Mnemonic
stream_attribute()		
length	8	uimsbf
stream_coding_type	8	bslbf
if (stream_coding_type==0x02) {		
video_format	4	bslbf
frame_rate	4	bslbf
aspect_ratio	4	bslbf
reserved_for_future_use	4	bslbf
} else if (stream_coding_type==0x80		
stream_coding_type==0x81		
stream_coding_type==0x82) {		
audio_presentation_type	4	bslbf
sampling_frequency	4	bslbf
audio_language_code	8*2	bslbf
reserved_for_future_use	8	bslbf
} else if (stream_coding_type==0x90) {		
// Presentation graphics stream		
PG_language_code	8*2	bslbf
} else if (stream_coding_type==0x91) {		
// Interactive graphics stream		
IG_language_code	8*2	bslbf
} else if (stream_coding_type==0x92) {		
// Text subtitle stream		
character_code	8	bslbf
textST_language_code	8*2	bslbf
}		
}		

도면28

stream_coding_type

stream_coding_type	Meaning
0x02	MPEG-2 video stream
0x80	HDMV LPCM audio
0x81	Dolby AC-3 audio
0x82	dts audio
0x90	Presentation graphics stream
0x91	Interactive graphics stream
0x92	Text subtitle stream
other values	reserved

도면29**video_format**

video_format	Meaning	Video standard
0	reserved	
1	480i	ITU-R BT. 601-4
2	576i	ITU-R BT. 601-4
3	480p	SMPTE 293M
4	1080i	SMPTE 274M
5	720p	SMPTE 296M
6	1080p	SMPTE 274M
7 - 14	reserved	

도면30**frame_rate**

frame_rate	Meaning [Hz]
0	reserved
1	24 000/1001 (23.976...)
2	24
3	25
4	30 000/1001 (29.97...)
5	reserved
6	50
7	60 000/1001 (59.94...)
8-15	reserved

도면31**aspect_ratio**

aspect_ratio	Meaning
0	reserved
1	reserved
2	4:3 display aspect ratio
3	16:9 display aspect ratio
4-15	reserved

도면32

audio_presentation_type

audio_presentation_type	Meaning
0	reserved
1	single mono channel
2	dual mono channel
3	stereo (2-channel)
4	reserved
5	reserved
6	multi-channel
7-15	reserved

도면33

sampling_frequency

sampling_frequency	Meaning
0	reserved
1	48 kHz
2	reserved
3	reserved
4	96 kHz
5 -15	reserved

도면34

Character code value	Character set	Character Encoding scheme
0x00	reserved	
0x01	Unicode V1.1 (ISO 10646-1)	UTF8
0x02	Unicode V1.1 (ISO 10646-1)	UTF16 big endian
0x03	Shift JIS (Japanese)	
0x04	KSC 5601-1987 including KSC 5653 for Roman character (Korean)	
0x05	GB18030-2000 (Chinese)	
0x06	GB2312 (Chinese)	
0x07	BIG5 (Chinese)	
Others	Reserved	

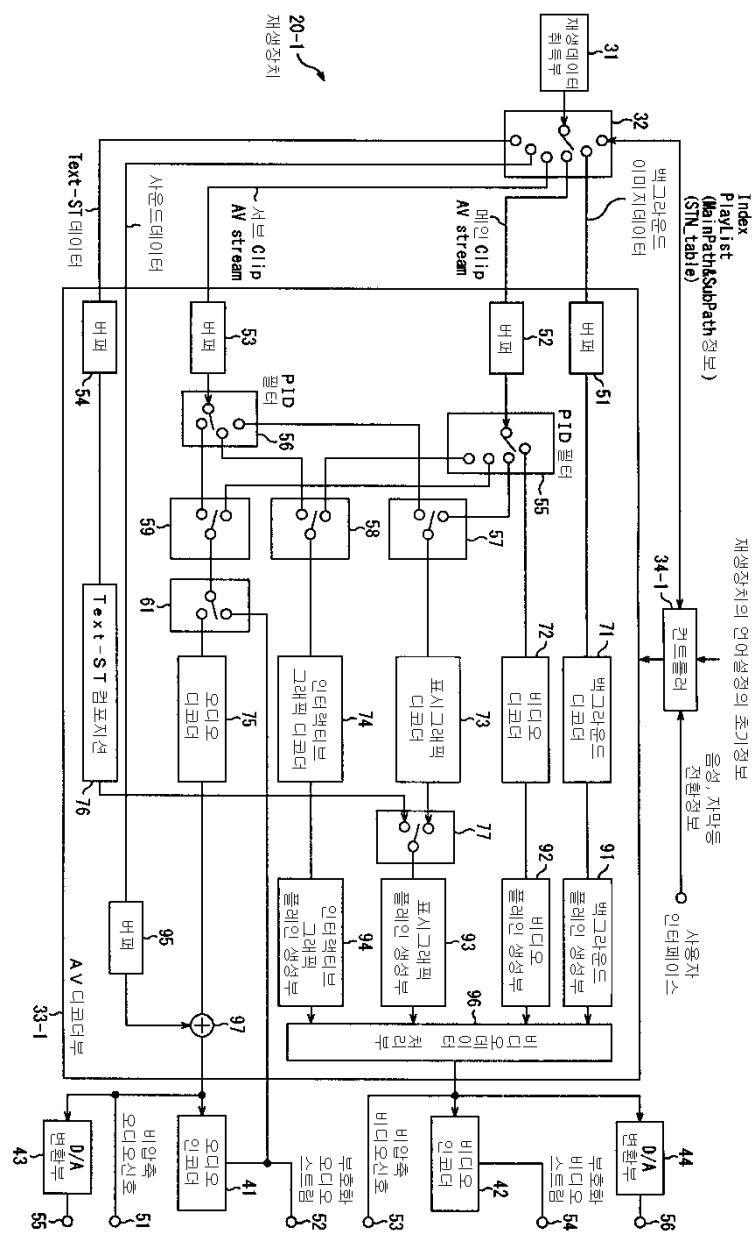
도면35

스트림번호테이블	
A_SN=1:	오디오 2
A_SN=2:	오디오 1
A_SN=3:	오디오 3
A_SN2=1:	오디오 4
A_SN2=2:	오디오 5

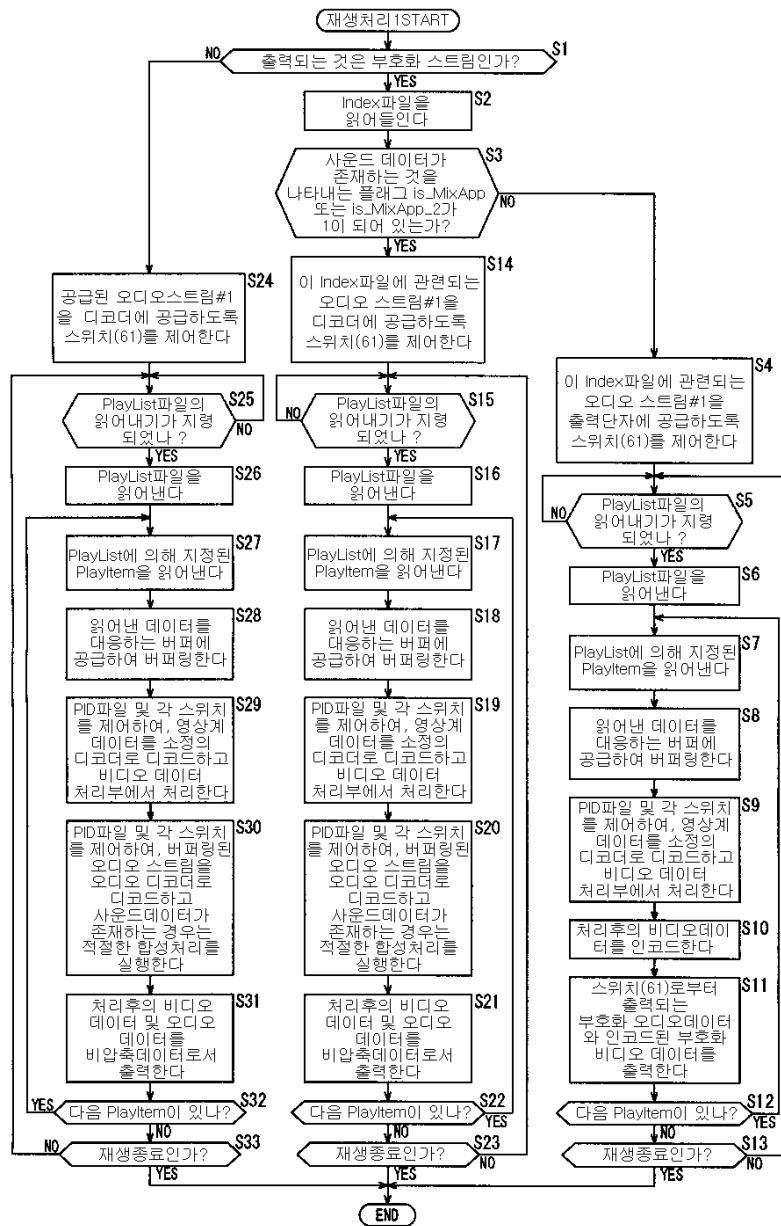
도면36

Syntax	No. of bits
sound.bdmv {	
type_indicator	8*4
version_number	8*4
SoundData_start_address	32
reserved_for_future_use	224
SoundIndex()	
for (i=0;i<L1;i++) {	
padding_word	16
}	
SoundData()	
for (i=0;i<L2;i++) {	
padding_word	16
}	
}	

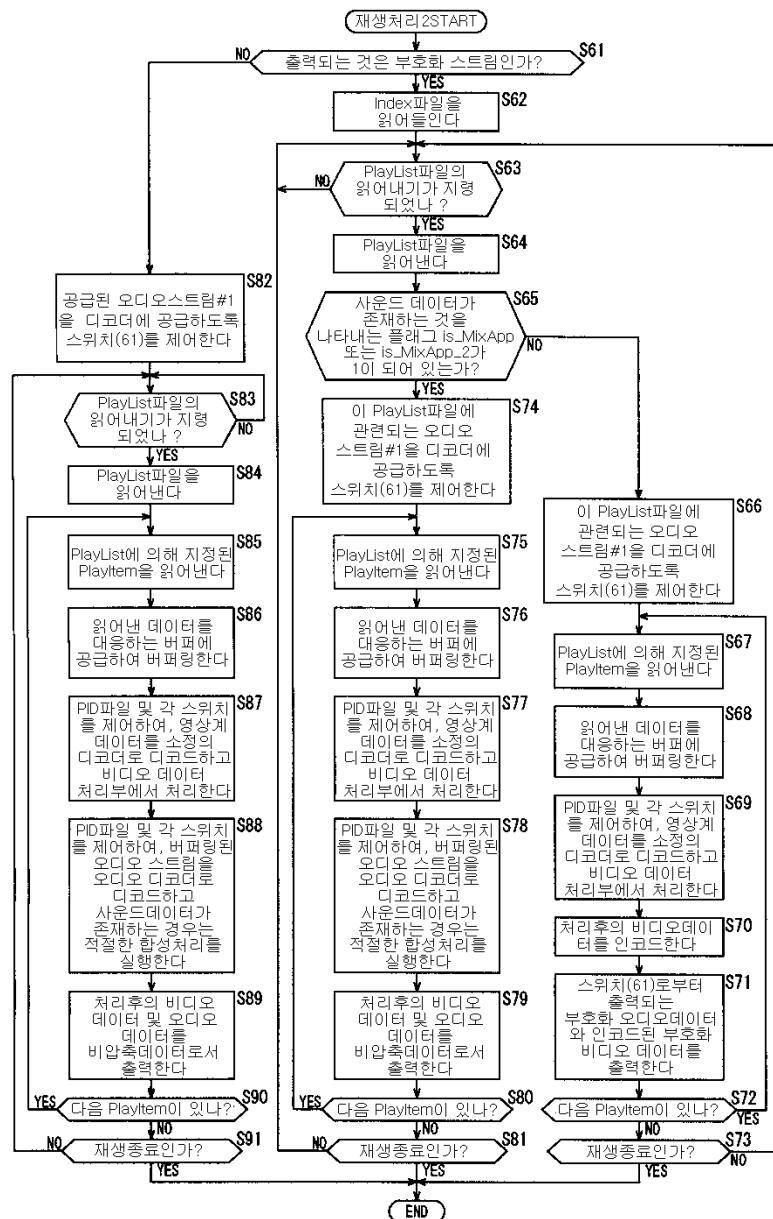
도면37



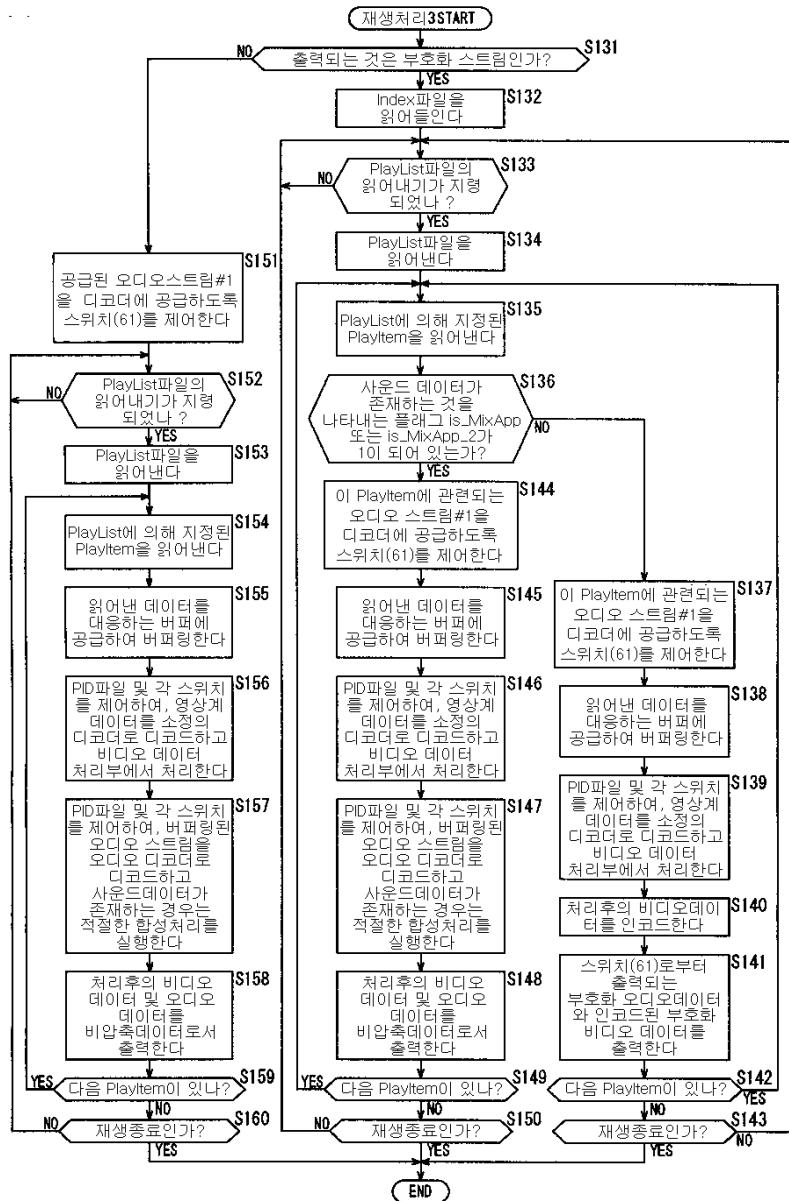
도면38



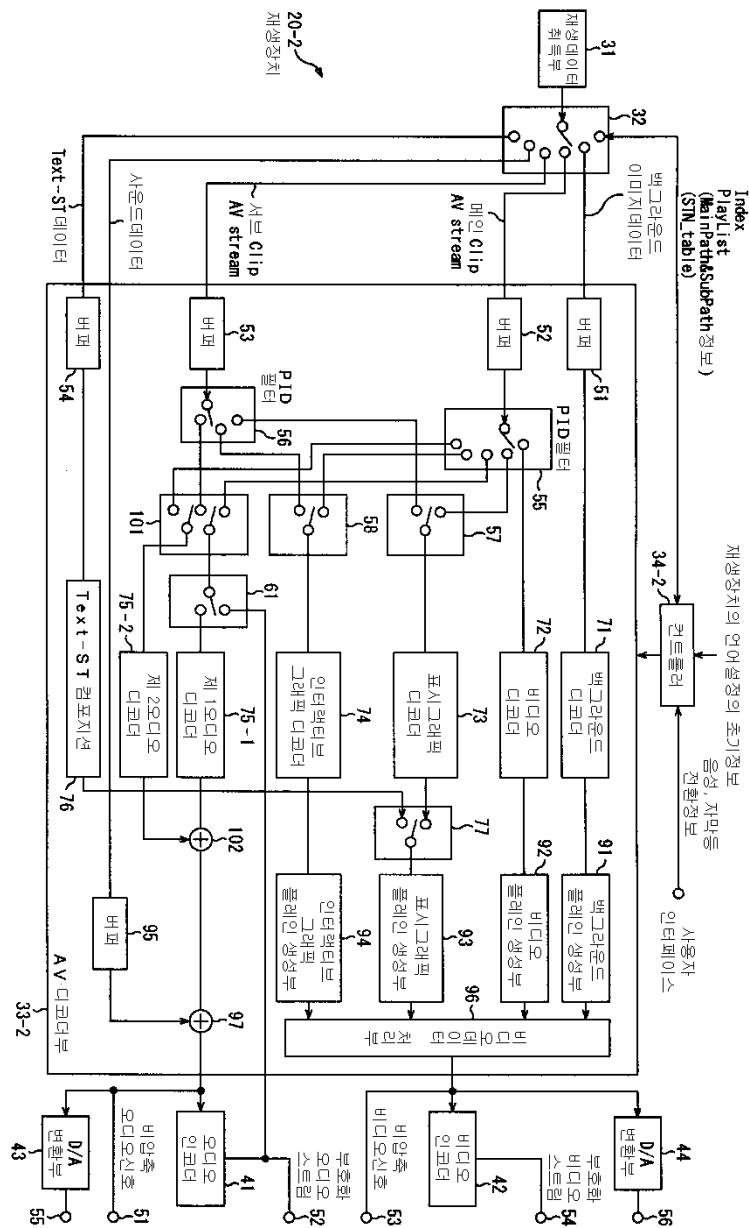
도면39



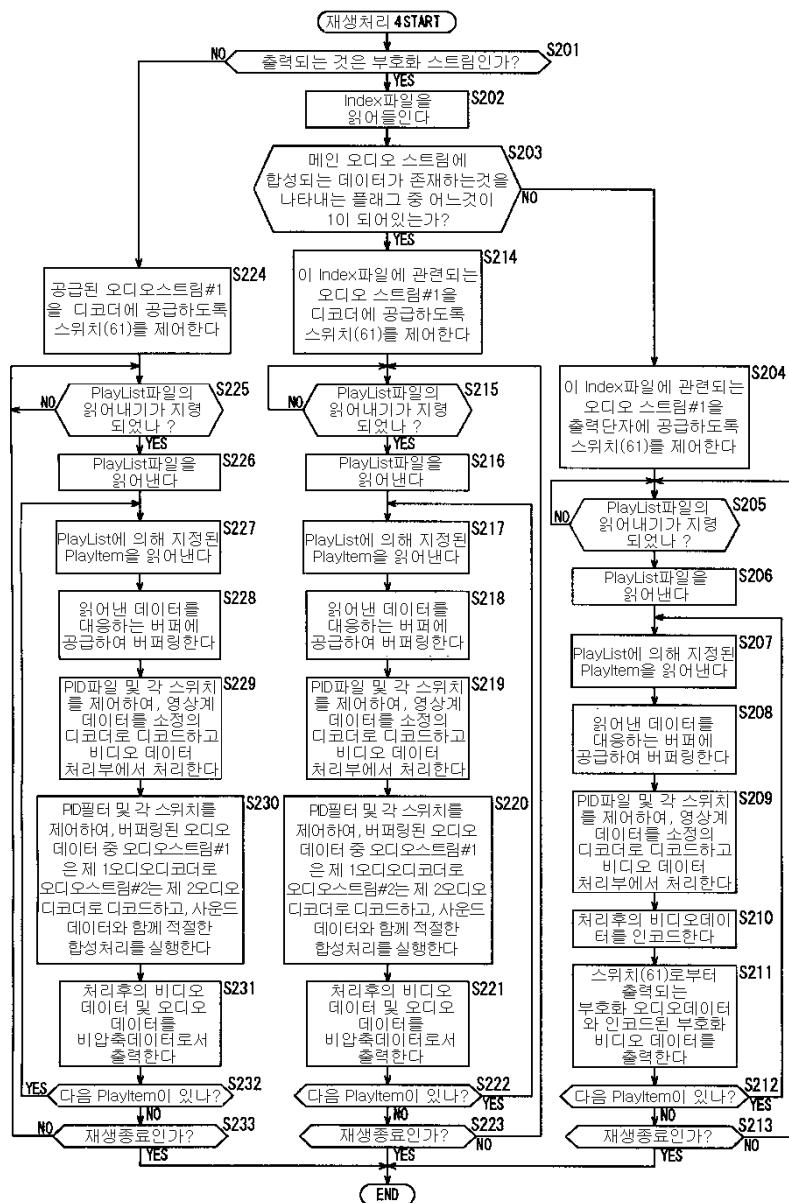
도면40



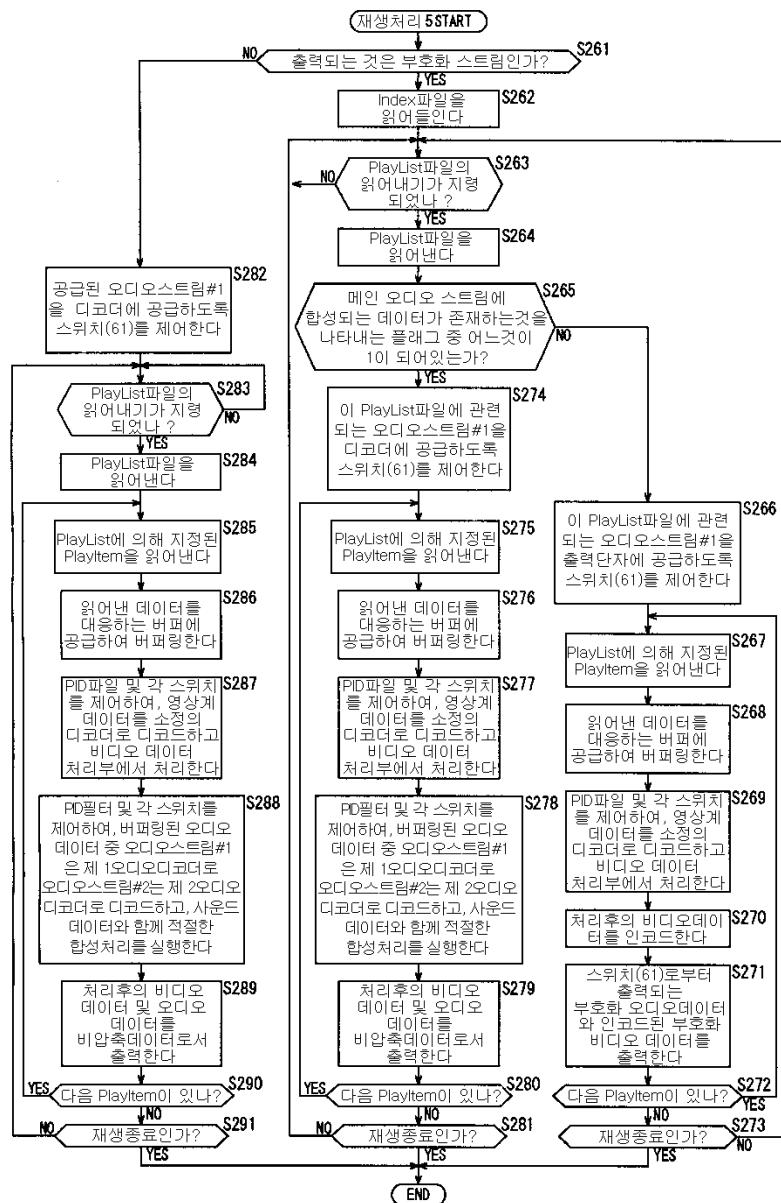
도면41



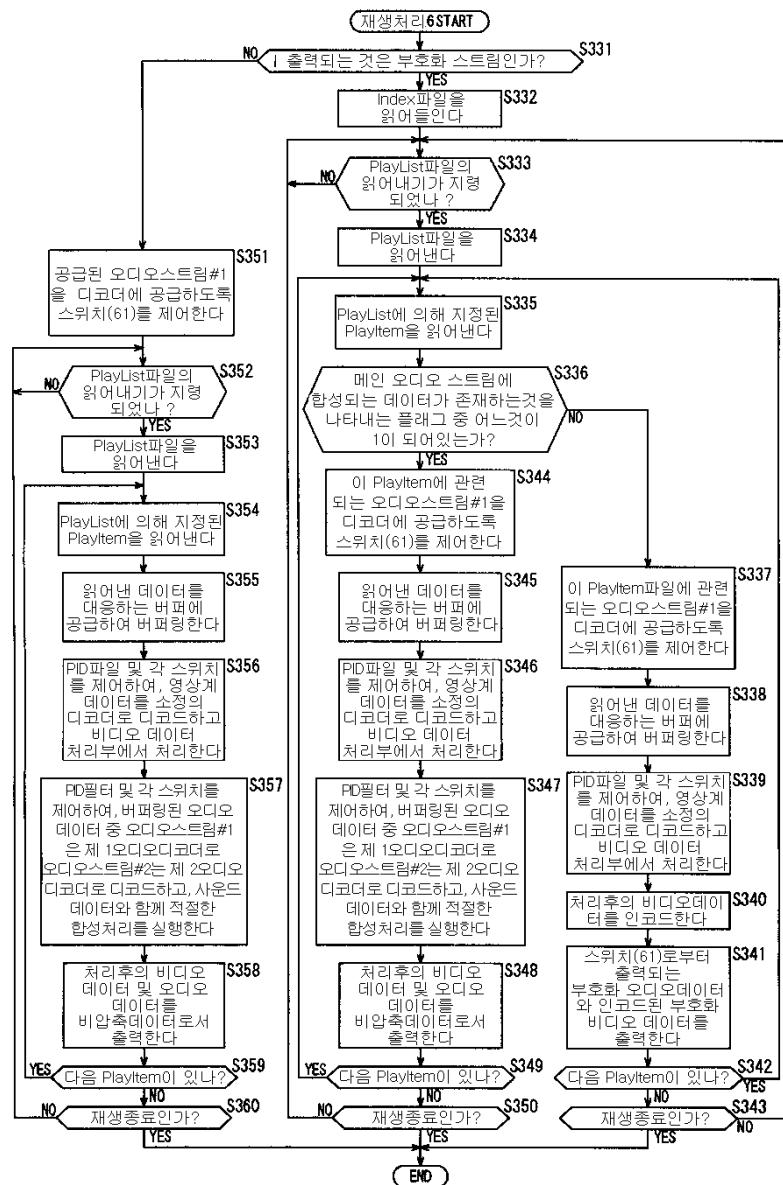
도면42



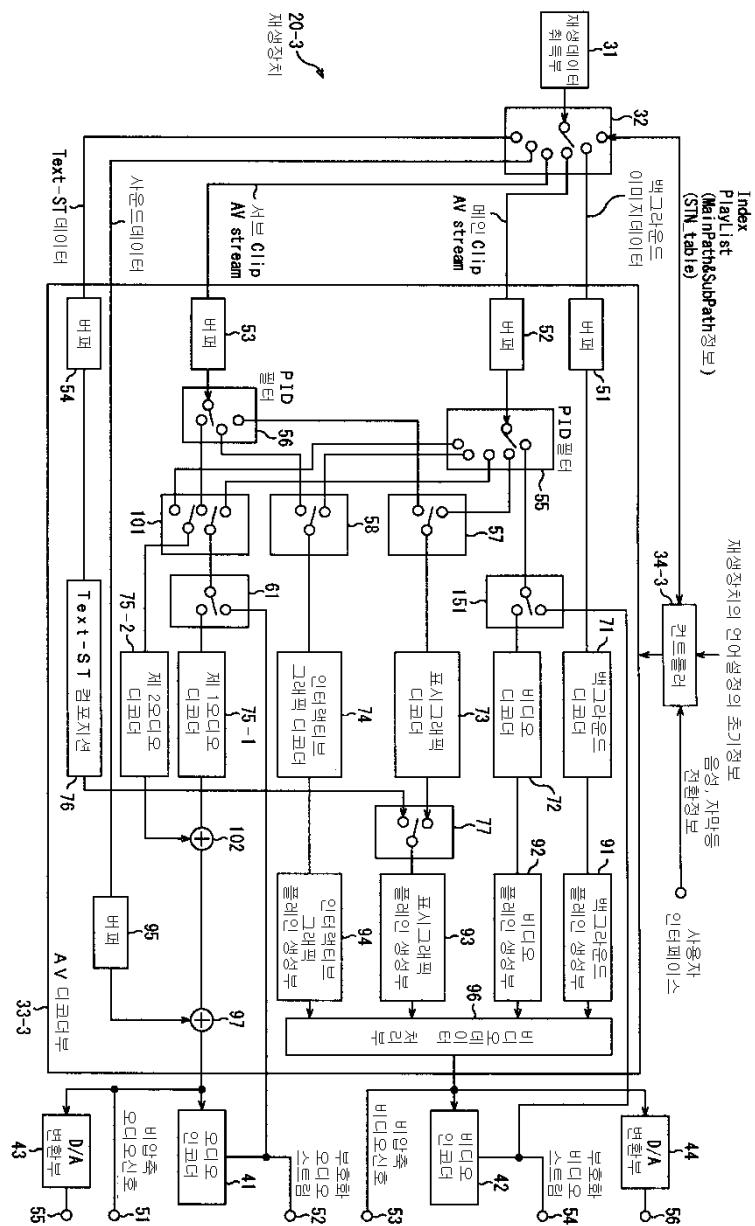
도면43



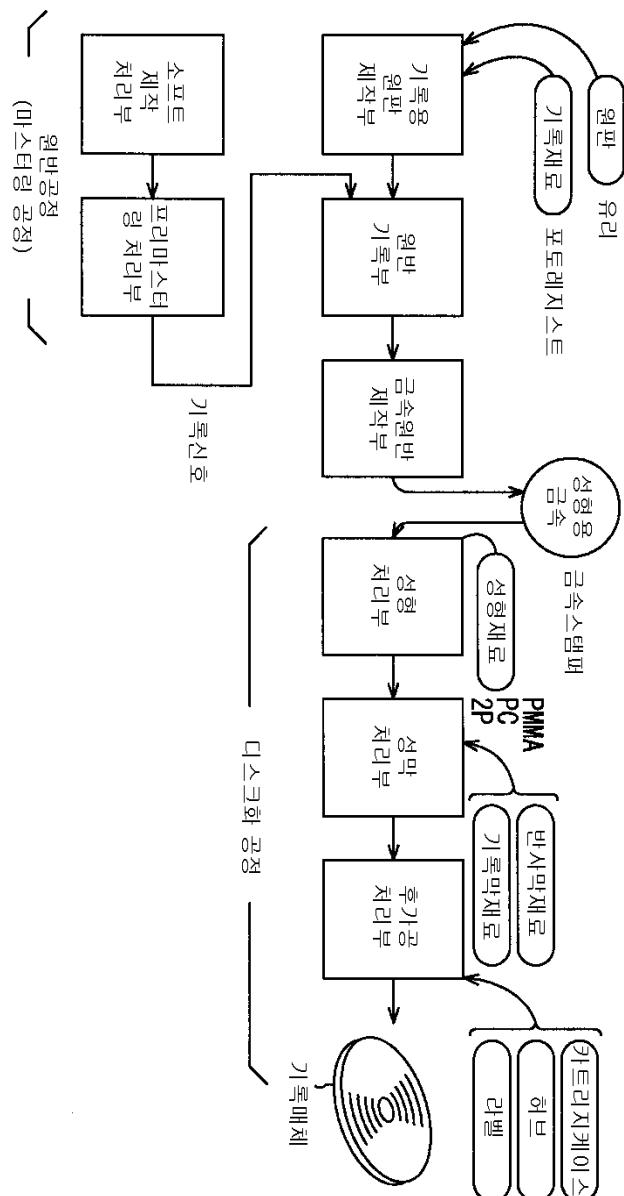
도면44



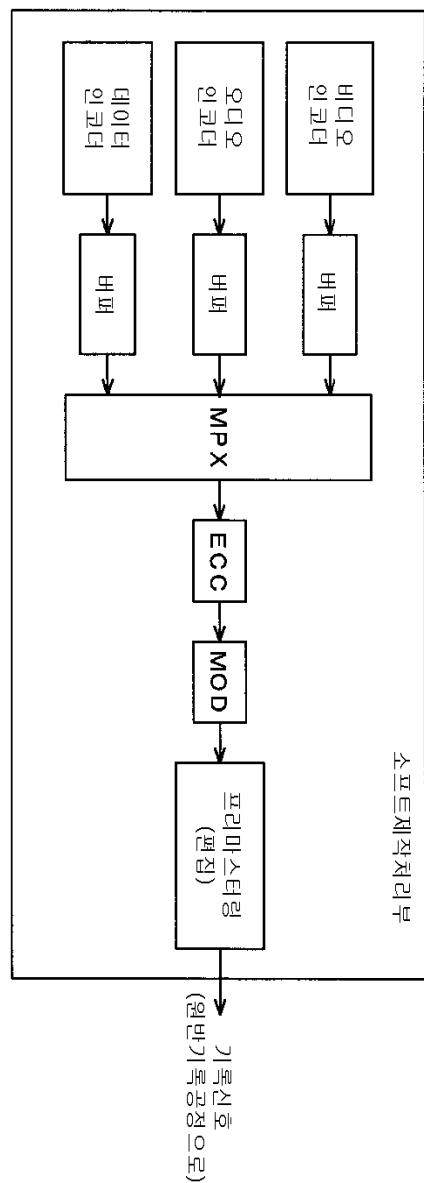
도면45



도면46



도면47



도면48

