

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4985895号
(P4985895)

(45) 発行日 平成24年7月25日 (2012. 7. 25)

(24) 登録日 平成24年5月11日 (2012. 5. 11)

(51) Int. Cl.	F I
H O 4 N 13/04 (2006. 01)	H O 4 N 13/04
H O 4 N 5/92 (2006. 01)	H O 4 N 5/92 C
G 1 1 B 20/12 (2006. 01)	G 1 1 B 20/12 1 O 3

請求項の数 4 (全 48 頁)

(21) 出願番号	特願2012-29502 (P2012-29502)	(73) 特許権者	000002185
(22) 出願日	平成24年2月14日 (2012. 2. 14)		ソニー株式会社
(62) 分割の表示	特願2010-65109 (P2010-65109)		東京都港区港南1丁目7番1号
原出願日	平成22年3月19日 (2010. 3. 19)	(74) 代理人	100082131
(65) 公開番号	特開2012-104222 (P2012-104222A)		弁理士 稲本 義雄
(43) 公開日	平成24年5月31日 (2012. 5. 31)	(74) 代理人	100121131
審査請求日	平成24年2月14日 (2012. 2. 14)		弁理士 西川 孝
(31) 優先権主張番号	特願2009-93159 (P2009-93159)	(72) 発明者	加藤 元樹
(32) 優先日	平成21年4月7日 (2009. 4. 7)		東京都港区港南1丁目7番1号 ソニー株
(33) 優先権主張国	日本国 (JP)		式会社内
早期審査対象出願		審査官	伊東 和重

最終頁に続く

(54) 【発明の名称】 再生装置、再生方法、および記録方法

(57) 【特許請求の範囲】

【請求項 1】

ビデオストリームを構成する所定の単位の水データであるエクステンツとして、複数視点から撮影されたビデオストリームをH.264 AVC/MVCによって符号化して得られた基本ストリームの前記エクステンツと拡張ストリームの前記エクステンツが交互に配置して記録された記録媒体から、前記基本ストリームと前記拡張ストリームの再生を制御する情報であるPlayListファイルを読み出し、前記PlayListファイルに含まれる情報からファイル名が特定される、前記基本ストリームのファイルおよび前記拡張ストリームのファイルの拡張子であるm2tsとは異なる拡張子のファイルであるインタリーブファイルに基づいて、前記基本ストリームの前記エクステンツと前記拡張ストリームの前記エクステンツを読み出す読み出し部と、

読み出された前記基本ストリームの前記エクステンツと前記拡張ストリームの前記エクステンツを分離する分離部と、

分離されたエクステンツから構成される前記基本ストリームと前記拡張ストリームを復号する復号部と

を備える再生装置。

【請求項 2】

前記読み出し部は、さらに、前記基本ストリームの前記エクステンツに関する情報を含む第1のクリップ情報と、前記拡張ストリームの前記エクステンツに関する情報を含む第2のクリップ情報とを前記記録媒体から読み出し、

10

20

前記分離部は、前記第1のクリップ情報に含まれる前記エクステントに関する情報と前記第2のクリップ情報に含まれる前記エクステントに関する情報を用いて、前記基本ストリームの前記エクステントと前記拡張ストリームの前記エクステントを分離する

請求項1に記載の再生装置。

【請求項3】

ビデオストリームを構成する所定の単位の水データであるエクステントとして、複数視点から撮影されたビデオストリームをH.264 AVC/MVCによって符号化して得られた基本ストリームの前記エクステントと拡張ストリームの前記エクステントが交互に配置して記録された記録媒体から、前記基本ストリームと前記拡張ストリームの再生を制御する情報であるPlayListファイルを読み出し、前記PlayListファイルに含まれる情報からファイル名が特定される、前記基本ストリームのファイルおよび前記拡張ストリームのファイルの拡張子であるm2tsとは異なる拡張子のファイルであるインタリーブファイルに基づいて、前記基本ストリームの前記エクステントと前記拡張ストリームの前記エクステントを読み出し、

10

読み出した前記基本ストリームの前記エクステントと前記拡張ストリームの前記エクステントを分離し、

分離したエクステントから構成される前記基本ストリームと前記拡張ストリームを復号する

ステップを含む再生方法。

【請求項4】

20

複数視点から撮影されたビデオストリームをH.264 AVC/MVCによって符号化し、

ビデオストリームを構成する所定の単位の水データであるエクステントとして、符号化して得られた基本ストリームの前記エクステントと拡張ストリームの前記エクステントを交互に配置して記録し、

前記基本ストリームのファイルと、

前記拡張ストリームのファイルと、

前記基本ストリームのファイルおよび前記拡張ストリームのファイルの拡張子であるm2tsとは異なる拡張子のファイルであるインタリーブファイルと、

前記基本ストリームと前記拡張ストリームの再生を制御する情報であり、前記インタリーブファイルのファイル名の特定に再生装置によって用いられる情報を含むPlayListと

30

を記録する

ステップを含む記録方法。

【発明の詳細な説明】

【技術分野】

【0001】

本技術は、再生装置、再生方法、および記録方法に関し、特に、記録媒体から、3D(Dimension)画像のコンテンツを適切に再生することができるようにした再生装置、再生方法、および記録方法に関する。

【背景技術】

【0002】

40

例えば、映画等のコンテンツとしては、2次元(2D)画像のコンテンツが主流であるが、最近では、立体視が可能な3次元(3D)画像(グラフィック)のコンテンツが注目を集めている。

【0003】

3D画像(以下、ステレオ画像ともいう)の表示の方式としては、種々の方式があるが、いずれの方式を採用する場合であっても、3D画像の水データ量は、2D画像の水データ量よりも大になる。

【0004】

また、映画等の、高解像度の画像のコンテンツは、大容量であることもあり、そのような大容量の画像のコンテンツを、水データの多い3D画像として記録するには、大容量の記

50

録媒体が必要である。

【0005】

そのような大容量の記録媒体としては、例えば、BD(Blu-Ray (登録商標))-ROM(Read Only Memory)等のBlu-Ray (登録商標) Disc (以下、BDともいう)がある。

【0006】

ここで、ファイル化手段により信号の記録媒体における記録位置をセクタよりも小さい所定単位で指定してファイル化するようにしたことにより、ファイル化した信号を任意の位置で分割又は結合するような編集処理でも、当該信号の編集対象部分のみをファイル化するように記録媒体に記録しなおす必要がなく、ファイル管理情報上のみで編集処理することができ、かくして編集処理を格段的に簡易化させ得るファイル管理装置が提案されている(特許文献1)。

10

【先行技術文献】

【特許文献】

【0007】

【特許文献1】特開平11-195287号公報

【発明の概要】

【発明が解決しようとする課題】

【0008】

ところで、現行のBDの規格では、3D画像のコンテンツを、BDに、どのように記録し、また、再生するかは、規定されていない。

20

【0009】

しかしながら、3D画像のコンテンツの記録や再生の仕方を、3D画像のコンテンツのオーサリングを行うオーサ(author)に委ねてしまうと、3D画像のコンテンツを適切に再生することができなくなるおそれがある。

【0010】

本技術は、このような状況に鑑みてなされたものであり、BD等の記録媒体から、ステレオ画像(3D画像)のコンテンツを適切に再生することができるようにするものである。

【課題を解決するための手段】

【0011】

本技術の一側面の再生装置は、ビデオストリームを構成する所定の単位の水データであるエクステンツとして、複数視点から撮影されたビデオストリームをH.264 AVC/MVCによって符号化して得られた基本ストリームの前記エクステンツと拡張ストリームの前記エクステンツが交互に配置して記録された記録媒体から、前記基本ストリームのファイルおよび前記拡張ストリームのファイルの拡張子であるm2tsとは異なる拡張子のファイルであるインタリーブファイルに基づいて、前記基本ストリームの前記エクステンツと前記拡張ストリームの前記エクステンツを読み出す読み出し部と、読み出された前記基本ストリームの前記エクステンツと前記拡張ストリームの前記エクステンツを分離する分離部と、分離されたエクステンツから構成される前記基本ストリームと前記拡張ストリームを復号する復号部とを備える。

30

【0012】

前記読み出し部には、前記基本ストリームと前記拡張ストリームの再生を制御する情報であるPlayListファイルを前記記録媒体から読み出させ、前記PlayListファイルに含まれる情報から特定される前記インタリーブファイルに基づいて、前記基本ストリームの前記エクステンツと前記拡張ストリームの前記エクステンツを読み出させることができる。

40

【発明の効果】

【0013】

本技術によれば、ステレオ画像(3D画像)のコンテンツを適切に再生することができる。

【図面の簡単な説明】

【0014】

50

【図 1】 本技術を適用した再生装置を含む再生システムの構成例を示す図である。

【図 2】 撮影の例を示す図である。

【図 3】 MVCエンコーダの構成例を示すブロック図である。

【図 4】 参照画像の例を示す図である。

【図 5】 TSの構成例を示す図である。

【図 6】 TSの他の構成例を示す図である。

【図 7】 TSのさらに他の構成例を示す図である。

【図 8】 AVストリームの管理の例を示す図である。

【図 9】 Main PathとSub Pathの構造を示す図である。

【図 10】 光ディスクに記録されるファイルの管理構造の例を示す図である。

10

【図 11】 Playlistファイルのシンタクスを示す図である。

【図 12】 図 11 のPlaylist()のシンタクスを示す図である。

【図 13】 図 12 のSubPath()のシンタクスを示す図である。

【図 14】 図 13 のSubPlayItem(i)のシンタクスを示す図である。

【図 15】 図 12 のPlayItem()のシンタクスを示す図である。

【図 16】 図 15 のSTN_table()のシンタクスを示す図である。

【図 17】 3D_PlayListの具体例を示す図である。

【図 18】 typeの意味を示す図である。

【図 19】 SubPath_typeの意味を示す図である。

【図 20】 再生装置の構成例を示すブロック図である。

20

【図 21】 図 20 のデコーダ部の構成例を示す図である。

【図 22】 3D_PlayListの例を示す図である。

【図 23】 clpiファイルのシンタクスを示す図である。

【図 24】 図 22、図 23 のデータを用いて行われるファイル管理の概念を示す図である。

。

【図 25】 図 22 の3D_PlayListファイルに従って行われる再生処理について説明するフローチャートである。

【図 26】 chunk_map()のシンタクスの例を示す図である。

【図 27】 chunk_map()の具体例を示す図である。

【図 28】 チャンクの分離について示す図である。

30

【図 29】 3D_PlayListの他の例を示す図である。

【図 30】 clpiファイルのシンタクスを示す図である。

【図 31】 図 29、図 30 のデータを用いて行われるファイル管理の概念を示す図である。

。

【図 32】 図 29 の3D_PlayListファイルに従って行われる再生処理について説明するフローチャートである。

【図 33】 3D_PlayListのさらに他の例を示す図である。

【図 34】 clpiファイルのシンタクスを示す図である。

【図 35】 図 33、図 34 のデータを用いて行われるファイル管理の概念を示す図である。

40

。

【図 36】 図 33 の3D_PlayListファイルに従って行われる再生処理について説明するフローチャートである。

【図 37】 図 27 のchunk_map()の内容をまとめて示す図である。

【図 38】 EP_map()のシンタクスを示す図である。

【図 39】 再生装置のランダムアクセス処理について説明するフローチャートである。

【図 40】 ステップ S 44、S 45 の処理により特定される位置の例を示す図である。

【図 41】 ステップ S 46 の処理により特定されるSPN_chunk_start[k]を示す図である。

【図 42】 光ディスク上に記録されたAVストリームの構造を示す図である。

【図 43】 Clip AVストリームの例を示す図である。

【図 44】 EP_mapの例を示す図である。

50

【図４５】SPN_EP_startが指すソースパケットのデータ構造の例を示す図である。

【図４６】ソフト製作処理部の構成例を示すブロック図である。

【図４７】ソフト製作処理部を含む構成の例を示す図である。

【発明を実施するための形態】

【００１５】

[再生システムの構成例]

図１は、本技術を適用した再生装置２０１を含む再生システムの構成例を示す図である。

【００１６】

図１に示すように、この再生システムは、再生装置２０１と表示装置２０３がHDMI(High Definition Multimedia Interface)ケーブルなどで接続されることによって構成される。再生装置２０１には、BDなどの光ディスク２０２が装着される。

10

【００１７】

光ディスク２０２には、視点の数が２つのステレオ画像（３Ｄ画像）を表示するために必要なストリームが記録されている。

【００１８】

各ストリームのデータは、エクステンツ単位で、インターリーブされた状態で光ディスク２０２に記録されている。

【００１９】

再生装置２０１は、光ディスク２０２に記録されているストリームの３Ｄ再生に対応したプレーヤである。再生装置２０１は、光ディスク２０２に記録されているストリームを再生し、再生して得られた３Ｄ画像をテレビジョン受像機などよりなる表示装置２０３に表示させる。音声についても同様に再生装置２０１により再生され、表示装置２０３に設けられるスピーカなどから出力される。

20

【００２０】

光ディスク２０２には、３Ｄ画像を表示することができるようなストリームが記録されている。ストリームを光ディスク２０２に記録するための符号化の方式として、例えば、H.264 AVC(Advanced Video Coding)/MVC(Multi-view Video coding)が採用される。

【００２１】

[H.264 AVC/MVC Profile]

30

H.264 AVC/MVCでは、Base view videoと呼ばれる画像ストリームと、Dependent view videoと呼ばれる画像ストリームとが定義されている。以下、適宜、H.264 AVC/MVCを単にMVCという。

【００２２】

図２は、撮影の例を示す図である。

【００２３】

図２に示すように、同じ被写体を対象として、Ｌ画像用のカメラとＲ画像用のカメラによって撮影が行われる。Ｌ画像用のカメラとＲ画像用のカメラによって撮影された映像のエレメンタリストリームがMVCエンコーダに入力される。

【００２４】

40

図３は、MVCエンコーダの構成例を示すブロック図である。

【００２５】

図３に示すように、MVCエンコーダ２１１は、H.264/AVCエンコーダ２２１、H.264/AVCデコーダ２２２、Depth算出部２２３、Dependent view videoエンコーダ２２４、およびマルチプレкса２２５から構成される。

【００２６】

Ｌ画像用のカメラにより撮影された映像＃１のストリームはH.264/AVCエンコーダ２２１とDepth算出部２２３に入力される。また、Ｒ画像用のカメラにより撮影された映像＃２のストリームはDepth算出部２２３とDependent view videoエンコーダ２２４に入力される。映像＃２のストリームがH.264/AVCエンコーダ２２１とDepth算出部２２３に入力され、

50

映像#1のストリームがDepth算出部223とDependent view videoエンコーダ224に
入力されるようにしてもよい。

【0027】

H.264/AVCエンコーダ221は、映像#1のストリームを、例えばH.264 AVC/High Prof
ileビデオストリームとして符号化する。H.264/AVCエンコーダ221は、符号化して得ら
れたAVCビデオストリームを、Base view videoストリームとしてH.264/AVCデコーダ22
2とマルチプレクサ225に出力する。

【0028】

H.264/AVCデコーダ222は、H.264/AVCエンコーダ221から供給されたAVCビデオス
トリームをデコードし、デコードして得られた映像#1のストリームをDependent view v
ideoエンコーダ224に出力する。 10

【0029】

Depth算出部223は、映像#1のストリームと映像#2のストリームに基づいてDepth
(視差)を算出し、算出したDepthのデータをマルチプレクサ225に出力する。

【0030】

Dependent view videoエンコーダ224は、H.264/AVCデコーダ222から供給された
映像#1のストリームと、外部から入力された映像#2のストリームをエンコードし、De
pendent view videoストリームを出力する。

【0031】

Base view videoには、他のストリームを参照画像とする予測符号化が許されていない
が、図4に示すように、Dependent view videoには、Base view videoを参照画像とする
予測符号化が許されている。例えばL画像をBase view videoとするとともにR画像をDepen
dent view videoとして符号化を行った場合、その結果得られるDependent view videoス
トリームのデータ量は、Base view videoストリームのデータ量に比較して少なくなる。 20

【0032】

なお、H.264/AVCでの符号化であるから、Base view videoについて時間方向の予測は行
われている。また、Dependent view videoについても、view間の予測とともに、時間方向
の予測が行われている。Dependent view videoをデコードするには、エンコード時に参照
先とした、対応するBase view videoのデコードが先に終了している必要がある。

【0033】

Dependent view videoエンコーダ224は、このようなview間の予測も用いて符号化し
て得られたDependent view videoストリームをマルチプレクサ225に出力する。 30

【0034】

マルチプレクサ225は、H.264/AVCエンコーダ221から供給されたBase view video
ストリームと、Depth算出部223から供給されたDependent view videoストリーム (Dep
thのデータ) と、Dependent view videoエンコーダ224から供給されたDependent view
videoストリームとを、例えばMPEG2 TSとして多重化する。Base view videoストリーム
とDependent view videoストリームは1本のMPEG2 TSに多重化されることもあるし、別々
のMPEG2 TSに含まれることもある。

【0035】

マルチプレクサ225は、生成したTS (MPEG2 TS) を出力する。マルチプレクサ225
から出力されたTSは、他の管理データとともに記録装置において光ディスク202に記録
され、光ディスク202に記録された形で再生装置201に提供される。 40

【0036】

以下、適宜、図3に示すように、2つのDependent view videoのうち、Depth情報から
なるDependent view videoをD1 view videoといい、R画像からなるDependent view video
をD2 view videoという。なお、R画像からなるDependent view videoをD1 view videoと
して処理し、Depth情報からなるDependent view videoをD2 view videoとして処理するよ
うにすることも可能である。

【0037】

また、Base view videoとD1 view videoを用いて行われる3D再生をB-D1再生という。Base view videoとD2 view videoを用いて行われる3D再生をB-D2再生という。

【0038】

再生装置201は、ユーザによる指示などに応じてB-D1再生を行う場合、Base view videoストリームとD1 view videoストリームを光ディスク202から読み出して再生する。

【0039】

また、再生装置201は、B-D2再生を行う場合、Base view videoストリームとD2 view videoストリームを光ディスク202から読み出して再生する。

【0040】

さらに、再生装置201は、通常の2D画像の再生を行う場合、Base view videoストリームだけを光ディスク202から読み出して再生する。 10

【0041】

Base view videoストリームはH.264/AVCで符号化されているAVCビデオストリームであるから、BDのフォーマットに対応したプレーヤであれば、そのBase view videoストリームを再生し、2D画像を表示させることが可能になる。

【0042】

[TSの構成例]

図5は、TSの構成例を示す図である。

【0043】

図5のMain TSにはBase view video、Dependent view video、Primary audio、Base PG、Dependent PG、Base IG、Dependent IGのそれぞれのストリームが多重化されている。このように、Dependent view videoストリームが、Base view videoストリームとともにMain TSに含まれていることもある。 20

【0044】

光ディスク202には、Main TSとSub TSが記録されている。Main TSは、少なくともBase view videoストリームを含むTSである。Sub TSは、Base view videoストリーム以外のストリームを含み、Main TSとともに用いられるTSである。

【0045】

ビデオと同様に3Dでの表示が可能になるように、PG、IGについてもBase viewとDependent viewのそれぞれのストリームが用意されている。 30

【0046】

図6は、TSの他の構成例を示す図である。

【0047】

図6のMain TSにはBase view video、Dependent view videoのそれぞれのストリームが多重化されている。

【0048】

一方、Sub TSにはPrimary audio、Base PG、Dependent PG、Base IG、Dependent IGのそれぞれのストリームが多重化されている。

【0049】

このように、ビデオストリームがMain TSに多重化され、PG、IGのストリーム等がSub TSに多重化されていることもある。 40

【0050】

図7は、TSのさらに他の構成例を示す図である。

【0051】

図7のMain TSにはBase view video、Primary audio、Base PG、Dependent PG、Base IG、Dependent IGのそれぞれのストリームが多重化されている。

【0052】

一方、Sub TSにはDependent view videoストリームが含まれている。

【0053】

このように、Dependent view videoストリームがBase view videoストリームとは別のT 50

Sに含まれていることもある。

【0054】

[アプリケーションフォーマット]

図8は、再生装置201によるAVストリームの管理の例を示す図である。

【0055】

AVストリームの管理は、図8に示すようにPlayListとClipの2つのレイヤを用いて行われる。AVストリームは、光ディスク202だけでなく、再生装置201のローカルストレージに記録されていることもある。

【0056】

ここでは、1つのAVストリームとそれに付随する情報であるClip Informationのペアを1つのオブジェクトと考え、それらをまとめてClipという。以下、AVストリームを格納したファイルをAVストリームファイルという。また、Clip Informationを格納したファイルをClip Informationファイルともいう。

10

【0057】

AVストリームは時間軸上に展開され、各Clipのアクセスポイントは、主に、タイムスタンプでPlayListにおいて指定される。Clip Informationファイルは、AVストリーム中のデコードを開始すべきアドレスを見つけるためなどに使用される。

【0058】

PlayListはAVストリームの再生区間の集まりである。AVストリーム中の1つの再生区間はPlayItemと呼ばれる。PlayItemは、時間軸上の再生区間のIN点とOUT点のペアで表される。図8に示すように、PlayListは1つまたは複数のPlayItemにより構成される。

20

【0059】

図8の左から1番目のPlayListは2つのPlayItemから構成され、その2つのPlayItemにより、左側のClipに含まれるAVストリームの前半部分と後半部分がそれぞれ参照されている。

【0060】

左から2番目のPlayListは1つのPlayItemから構成され、それにより、右側のClipに含まれるAVストリーム全体が参照されている。

【0061】

左から3番目のPlayListは2つのPlayItemから構成され、その2つのPlayItemにより、左側のClipに含まれるAVストリームのある部分と、右側のClipに含まれるAVストリームのある部分がそれぞれ参照されている。

30

【0062】

例えば、左から1番目のPlayListに含まれる左側のPlayItemが再生対象としてディスクナビゲーションプログラムにより指定された場合、そのPlayItemが参照する、左側のClipに含まれるAVストリームの前半部分の再生が行われる。このように、PlayListは、AVストリームの再生を管理するための再生管理情報として用いられる。

【0063】

PlayListの中で、1つ以上のPlayItemの並びによって作られる再生パスをメインパス(Main Path)という。

40

【0064】

また、PlayListの中で、Main Pathに並行して、1つ以上のSubPlayItemの並びによって作られる再生パスをサブパス(Sub Path)という。

【0065】

図9は、Main PathとSub Pathの構造を示す図である。

【0066】

PlayListは、1つのMain Pathと1つ以上のSub Pathを持つことができる。

【0067】

上述したBase view videoストリームは、Main Pathを構成するPlayItemが参照するストリームとして管理される。また、Dependent view videoストリームは、Sub Pathを構成す

50

るSubPlayItemが参照するストリームとして管理される。

【0068】

図9のPlayListは、3つのPlayItemの並びにより作られる1つのMain Pathと、3つのSub Pathを有している。

【0069】

Main Pathを構成するPlayItemには、先頭から順番にそれぞれIDが設定される。Sub Pathにも、先頭から順番にSubpath_id=0、Subpath_id=1、およびSubpath_id=2のIDが設定される。

【0070】

図9の例においては、Subpath_id=0のSub Pathには1つのSubPlayItemが含まれ、Subpath_id=1のSub Pathには2つのSubPlayItemが含まれる。また、Subpath_id=2のSub Pathには1つのSubPlayItemが含まれる。

10

【0071】

1つのPlayItemが参照するClip AVストリームには、少なくともビデオストリーム（メイン画像データ）が含まれる。

【0072】

また、Clip AVストリームには、Clip AVストリームに含まれるビデオストリームと同じタイミングで（同期して）再生されるオーディオストリームが1つ以上含まれてもよいし、含まなくてもよい。

【0073】

20

Clip AVストリームには、Clip AVストリームに含まれるビデオストリームと同期して再生されるビットマップの字幕データ（PG(Presentation Graphic)）のストリームが1つ以上含まれてもよいし、含まなくてもよい。

【0074】

Clip AVストリームには、Clip AVストリームファイルに含まれるビデオストリームと同期して再生されるIG(Interactive Graphic)のストリームが1つ以上含まれてもよいし、含まなくてもよい。IGのストリームは、ユーザにより操作されるボタンなどのグラフィックを表示させるために用いられる。

【0075】

1つのPlayItemが参照するClip AVストリームには、ビデオストリームと、それと同期して再生される0個以上のオーディオストリーム、0個以上のPGストリーム、および、0個以上のIGストリームが多重化されている。

30

【0076】

また、1つのSubPlayItemは、PlayItemが参照するClip AVストリームとは異なるストリーム（別ストリーム）の、ビデオストリーム、オーディオストリーム、または、PGストリームなどを参照する。

【0077】

このようなPlayList、PlayItem、SubPlayItemを使ったAVストリームの管理については、例えば、特開2008-252740号公報、特開2005-348314号公報に記載されている。

40

【0078】

[ディレクトリ構造]

図10は、光ディスク202に記録されるファイルの管理構造の例を示す図である。

【0079】

図10に示すように、ファイルはディレクトリ構造により階層的に管理される。光ディスク202上には1つのrootディレクトリが作成される。rootディレクトリの下が、1つの記録再生システムで管理される範囲となる。

【0080】

rootディレクトリの下にはBDMVディレクトリが置かれる。

【0081】

50

BDMVディレクトリの直下に、「Index.bdmv」の名前が設定されたファイルであるIndexファイルと、「MovieObject.bdmv」の名前が設定されたファイルであるMovieObjectファイルが格納される。

【0082】

BDMVディレクトリの下には、BACKUPディレクトリ、PLAYLISTディレクトリ、CLIPINFディレクトリ、STREAMディレクトリ等が設けられる。

【0083】

PLAYLISTディレクトリには、PlayListを記述したPlayListファイルが格納される。各PlayListファイルには、5桁の数字と拡張子「.mpls」を組み合わせた名前が設定される。

図10に示す1つのPlayListファイルには「00000.mpls」のファイル名が設定されている。

10

【0084】

CLIPINFディレクトリにはClip Informationファイルが格納される。各Clip Informationファイルには、5桁の数字と拡張子「.clpi」を組み合わせた名前が設定される。

【0085】

図10の3つのClip Informationファイルには、それぞれ、「00001.clpi」、「00002.clpi」、「00003.clpi」のファイル名が設定されている。以下、適宜、Clip Informationファイルをclpiファイルという。

【0086】

例えば、「00001.clpi」のclpiファイルは、Base view videoのClipに関する情報が記述されたファイルである。

20

【0087】

「00002.clpi」のclpiファイルは、D2 view videoのClipに関する情報が記述されたファイルである。

【0088】

「00003.clpi」のclpiファイルは、D1 view videoのClipに関する情報が記述されたファイルである。

【0089】

STREAMディレクトリにはストリームファイルが格納される。各ストリームファイルには、5桁の数字と拡張子「.m2ts」を組み合わせた名前、もしくは、5桁の数字と拡張子「.ilvt」を組み合わせた名前が設定される。以下、適宜、拡張子「.m2ts」が設定されたファイルをm2tsファイルという。また、拡張子「.ilvt」が設定されたファイルをilvtファイルという。

30

【0090】

「00001.m2ts」のm2tsファイルは2D再生用のファイルであり、このファイルを指定することによってBase view videoストリームの読み出しが行われる。

【0091】

「00002.m2ts」のm2tsファイルはD2 view videoストリームのファイルであり、「00003.m2ts」のm2tsファイルはD1 view videoストリームのファイルである。

【0092】

40

「10000.ilvt」のilvtファイルはB-D1再生用のファイルであり、このファイルを指定することによってBase view videoストリームとD1 view videoストリームの読み出しが行われる。

【0093】

「20000.ilvt」のilvtファイルはB-D2再生用のファイルであり、このファイルを指定することによってBase view videoストリームとD2 view videoストリームの読み出しが行われる。

【0094】

図10に示すものの他に、BDMVディレクトリの下には、オーディオストリームのファイルを格納するディレクトリなども設けられる。

50

【0095】

[各データのシンタクス]

図11は、PlayListファイルのシンタクスを示す図である。

【0096】

PlayListファイルは、図10のPLAYLISTディレクトリに格納される、拡張子「.mpls」が設定されるファイルである。

【0097】

図11のtype_indicatorは、「xxxx.mpls」のファイルの種類を表す。

【0098】

version_numberは、「xxxx.mpls」のバージョンナンバーを表す。version_numberは4桁の数字からなる。例えば、3D再生用のPlayListファイルには、「3D Spec version」であることを表す“0240”が設定される。“0240”が設定されたPlayListファイルに記述されるPlayListが、後述する3D_PlayListとなる。

【0099】

PlayList_start_addressは、PlayListファイルの先頭のバイトからの相対バイト数を単位として、PlayList()の先頭アドレスを表す。

【0100】

PlayListMark_start_addressは、PlayListファイルの先頭のバイトからの相対バイト数を単位として、PlayListMark()の先頭アドレスを表す。

【0101】

ExtensionData_start_addressは、PlayListファイルの先頭のバイトからの相対バイト数を単位として、ExtensionData()の先頭アドレスを表す。

【0102】

ExtensionData_start_addressの後には、160bitのreserved_for_future_useが含まれる。

【0103】

AppInfoPlayList()には、再生制限などの、PlayListの再生コントロールに関するパラメータが格納される。

【0104】

PlayList()には、Main PathやSub Pathなどに関するパラメータが格納される。PlayList()の内容については後述する。

【0105】

PlayListMark()には、PlayListのマーク情報、すなわち、チャプタジャンプなどを指令するユーザオペレーションまたはコマンドなどにおけるジャンプ先（ジャンプポイント）であるマークに関する情報が格納される。

【0106】

ExtensionData()には、プライベートデータが挿入できるようになっている。

【0107】

図12は、図11のPlayList()のシンタクスを示す図である。

【0108】

lengthは、このlengthフィールドの直後からPlayList()の最後までバイト数を示す32ビットの符号なし整数である。すなわち、lengthは、reserved_for_future_useからPlayListの最後までバイト数を表す。

【0109】

lengthの後には、16ビットのreserved_for_future_useが用意される。

【0110】

number_of_PlayItemsは、PlayListの中にあるPlayItemの数を示す16ビットのフィールドである。図9の例の場合、PlayItemの数は3である。PlayItem_idの値は、PlayListの中でPlayItem()が現れる順番に0から割り振られる。例えば、図9のPlayItem_id=0, 1, 2が割り振られる。

10

20

30

40

50

【0111】

number_of_SubPathsは、PlayListの中にあるSub Pathの数を示す16ビットのフィールドである。図9の例の場合、Sub Pathの数は3である。SubPath_idの値は、PlayListの中でSubPath()が現れる順番に0から割り振られる。例えば、図9のSubpath_id=0, 1, 2が割り振られる。その後のfor文では、PlayItemの数だけPlayItem()が参照され、Sub Pathの数だけSubPath()が参照される。

【0112】

図13は、図12のSubPath()のシンタクスを示す図である。

【0113】

lengthは、このlengthフィールドの直後からSub Path()の最後までバイト数を示す32ビットの符号なし整数である。すなわち、lengthは、reserved_for_future_useからPlayListの最後までバイト数を表す。

10

【0114】

lengthの後には、16ビットのreserved_for_future_useが用意される。

【0115】

SubPath_typeは、Sub Pathのアプリケーションの種類を示す8ビットのフィールドである。SubPath_typeは、例えば、Sub Pathがオーディオであるか、ビットマップ字幕であるか、テキスト字幕であるかなどの種類を示す場合に利用される。

【0116】

SubPath_typeの後には、15ビットのreserved_for_future_useが用意される。

20

【0117】

is_repeat_SubPathは、Sub Pathの再生方法を指定する1ビットのフィールドであり、Main Pathの再生の間にSub Pathの再生を繰り返し行うか、またはSub Pathの再生を1回だけ行うかを示す。例えば、Main Pathが参照するClipとSub Pathが参照するClipの再生タイミングが異なる場合（Main Pathを静止画のスライドショーのパスとし、Sub PathをBGMとするオーディオのパスとして使う場合など）に利用される。

【0118】

Is_repeat_SubPathの後には、8ビットのreserved_for_future_useが用意される。

【0119】

number_of_SubPlayItemsは、1つのSub Pathの中にあるSubPlayItemの数（エントリー数）を示す8ビットのフィールドである。例えば、図9のSubPath_id=0のSubPlayItemのnumber_of_SubPlayItemsは1であり、SubPath_id=1のSubPlayItemのnumber_of_SubPlayItemsは2である。その後のfor文では、SubPlayItemの数だけ、SubPlayItem()が参照される。

30

【0120】

図14は、図13のSubPlayItem(i)のシンタクスを示す図である。

【0121】

lengthは、このlengthフィールドの直後からSub playItem()の最後までバイト数を示す16ビットの符号なし整数である。

【0122】

図14のSubPlayItem(i)は、SubPlayItemが1つのClipを参照する場合と、複数のClipを参照する場合に分けて記述されている。

40

【0123】

SubPlayItemが1つのClipを参照する場合について説明する。

【0124】

Clip_Information_file_name[0]は、SubPlayItemが参照するClipのClip Informationファイルの名前を表す。

【0125】

Clip_codec_identifier [0] はClipのコーデック方式を表す。Clip_codec_identifier [0] の後にはreserved_for_future_useが含まれる。

50

【0126】

is_multi_Clip_entriesはマルチClipの登録の有無を示すフラグである。is_multi_Clip_entriesのフラグが立っている場合、SubPlayItemが複数のClipを参照する場合のシンタクスが参照される。

【0127】

ref_to_STC_id [0] はSTC不連続点（システムタイムベースの不連続点）に関する情報である。

【0128】

SubPlayItem_IN_timeはSub Pathの再生区間の開始位置を表し、SubPlayItem_OUT_timeは終了位置を表す。

10

【0129】

sync_PlayItem_idとsync_start_PTS_of_PlayItemは、Main Pathの時間軸上でSub Pathが再生を開始する時刻を表す。

【0130】

SubPlayItem_IN_time、SubPlayItem_OUT_time、sync_PlayItem_id、sync_start_PTS_of_PlayItemは、SubPlayItemが参照するClipにおいて共通に使用される。

【0131】

「if (is_multi_Clip_entries == 1 b)」であり、SubPlayItemが複数のClipを参照する場合について説明する。

【0132】

num_of_Clip_entriesは参照するClipの数を表す。Clip_Information_file_name[SubClip_entry_id]の数が、Clip_Information_file_name[0]を除くClipの数を指定する。

20

【0133】

Clip_codec_identifier[SubClip_entry_id]はClipのコーデック方式を表す。

【0134】

ref_to_STC_id[SubClip_entry_id]はSTC不連続点（システムタイムベースの不連続点）に関する情報である。ref_to_STC_id[SubClip_entry_id]の後にはreserved_for_future_useが含まれる。

【0135】

図15は、図12のPlayItem()のシンタクスを示す図である。

30

【0136】

lengthは、このlengthフィールドの直後からPlayItem()の最後までバイト数を示す16ビットの符号なし整数である。

【0137】

Clip_Information_file_name[0]は、PlayItemが参照するClipのClip Informationファイルの名前を表す。なお、Clipを含むm2tsファイルのファイル名と、それに対応するClip Informationファイルのファイル名には同じ5桁の数字が含まれる。

【0138】

Clip_codec_identifier [0] はClipのコーデック方式を表す。Clip_codec_identifier [0] の後にはreserved_for_future_useが含まれる。reserved_for_future_useの後にはis_multi_angle、connection_conditionが含まれる。

40

【0139】

ref_to_STC_id [0] はSTC不連続点（システムタイムベースの不連続点）に関する情報である。

【0140】

IN_timeはPlayItemの再生区間の開始位置を表し、OUT_timeは終了位置を表す。

【0141】

OUT_timeの後にはUO_mask_table()、PlayItem_random_access_mode、still_modeが含まれる。

【0142】

50

STN_table()には、対象のPlayItemが参照するAVストリームの情報が含まれる。また、対象のPlayItemと関連付けて再生されるSub Pathがある場合、そのSub Pathを構成するSubPlayItemが参照するAVストリームの情報も含まれる。

【0143】

図16は、図15のSTN_table()のシンタクスを示す図である。

【0144】

STN_table()は、PlayItemの属性として設定されている。

【0145】

lengthは、このlengthフィールドの直後からSTN_table()の最後までバイト数を示す16ビットの符号なし整数である。lengthの後には、16ビットのreserved_for_future_useが用意される。

10

【0146】

number_of_video_stream_entriesは、STN_table()の中でエントリーされる（登録される）、video_stream_idが与えられるストリームの数を表す。

【0147】

video_stream_idは、ビデオストリームを識別するための情報である。例えば、Base view videoストリームがこのvideo_stream_idにより特定される。

【0148】

Dependent view videoストリームのIDについては、STN_table()内で定義されるようにしてもよいし、Base view videoストリームのIDに所定の値を加算するなどして計算により求められるようにしてもよい。

20

【0149】

video_stream_numberは、ビデオ切り替えに使われる、ユーザから見えるビデオストリーム番号である。

【0150】

number_of_audio_stream_entriesは、STN_table()の中でエントリーされる、audio_stream_idが与えられる1番目のオーディオストリームのストリームの数を表す。audio_stream_idは、オーディオストリームを識別するための情報であり、audio_stream_numberは、音声切り替えに使われるユーザから見えるオーディオストリーム番号である。

【0151】

30

number_of_audio_stream2_entriesは、STN_table()の中でエントリーされる、audio_stream_id2が与えられる2番目のオーディオストリームのストリームの数を表す。audio_stream_id2は、オーディオストリームを識別するための情報であり、audio_stream_numberは、音声切り替えに使われるユーザから見えるオーディオストリーム番号である。この例においては、再生する音声を切り替えることができるようになっている。

【0152】

number_of_PG_txtST_stream_entriesは、STN_table()の中でエントリーされる、PG_txtST_stream_idが与えられるストリームの数を表す。この中では、ビットマップ字幕をランレングス符号化したPGストリームとテキスト字幕ファイル(txtST)がエントリーされる。PG_txtST_stream_idは、字幕ストリームを識別するための情報であり、PG_txtST_stream_numberは、字幕切り替えに使われるユーザから見える字幕ストリーム番号である。

40

【0153】

number_of_IG_stream_entriesは、STN_table()の中でエントリーされる、IG_stream_idが与えられるストリームの数を表す。この中ではIGストリームがエントリーされる。IG_stream_idは、IGストリームを識別するための情報であり、IG_stream_numberは、グラフィックス切り替えに使われるユーザから見えるグラフィックスストリーム番号である。

【0154】

Main TS、Sub TSのIDもSTN_table()に登録される。そのIDがエレメンタリストリームではなくTSのIDであることは、stream_attribute()に記述される。

【0155】

50

[Playlistの具体例]

図17は、3D再生用のPlaylistである3D_PlayListの具体例を示す図である。

【0156】

説明の便宜上、図17の左側には行数を表す数字と「:」を示している。この数字と「:」は3D_PlayListを構成するものではない。

【0157】

2行目のnumber_of_PlayItemsは図12のnumber_of_PlayItemsに対応し、3D_PlayList中のPlayItemの数を表す。2行目から8行目がPlayItemに関する記述になる。すなわち、3行目から8行目が、図12のfor文を用いたPlayItemに記述に対応する。

【0158】

5行目のref_to_B_clpi_file_nameは図15のClip_Information_file_name[0]に対応し、Base view videoストリームを格納したm2tsファイルのファイル名のうちの、「.m2ts」の拡張子を除いた5桁の数字を表す。この記述により、参照するm2tsファイルと、Base view videoのClipのclpiファイルが特定される。

【0159】

6行目のtypeは、Base view videoと、それに関連付けられたD1/D2 view videoのデータの光ディスク202上での配置のタイプを表す。typeは、例えば図15のClip_codec_identifier [0] に続くreserved_for_future_useを用いて設定される。

【0160】

図18は、typeの意味を示す図である。

【0161】

typeの値が0であることは、Base view video、D1 view video、D2 view videoがインターリーブされていないことを表す。

【0162】

この場合、存在するD1/D2 view videoの両方、またはいずれかのパッケージが、Base view videoのパッケージとともに1本のMPEG2-TSに多重化される。

【0163】

typeの値が1であることは、Base view video、D1 view video、D2 view videoが全てインターリーブされていることを表す。

【0164】

この場合、B view videoを含む第1のTSと、D1 view videoを含む第2のTSと、D2 view videoを含む第3のTSの3本のTSが、エクステント単位で光ディスク202上にインターリーブされている。

【0165】

typeの値が2であることは、Base view videoとD1 view videoがインターリーブされていることを表す。

【0166】

この場合、B view videoのパッケージを含む第1のTSと、D1 view videoのパッケージを含む第2のTSの2本のTSが、エクステント単位で光ディスク202上にインターリーブされている。第1のTSにはD2 view videoのパッケージが多重化されていても良い。また、第2のTSには、D2 view videoのパッケージが多重化されていても良い。

【0167】

typeの値が3であることは、Base view videoとD2 view videoがインターリーブされていることを表す。

【0168】

この場合、B view videoのパッケージを含む第1のTSと、D2 view videoのパッケージを含む第2のTSの2本のTSが、エクステント単位で光ディスク202上にインターリーブされている。第1のTSには、D1 view videoのパッケージが多重化されていても良い。また、第2のTSには、D1 view videoのパッケージが含まれてはいけない。

【0169】

10

20

30

40

50

図 1 7 の説明に戻り、7 行目のSTN_tableは図 1 5 のSTN_table()に対応する。図 1 6 を参照して説明したように、STN_tableには3D_PlayListにおいて参照する各ストリームのIDが記述される。

【0 1 7 0】

9 行目のnumber_of_SubPathsは図 1 2 のnumber_of_SubPathsに対応し、3D_PlayList中のSubPathの数を表す。9 行目から 1 4 行目がSubPathに関する記述になる。すなわち、1 0 行目から 1 4 行目が、図 1 2 のfor文を用いたSubPathに記述に対応する。

【0 1 7 1】

1 2 行目のSubPath_typeは図 1 3 のSubPath_typeに対応し、SubPathの種類を表す。

【0 1 7 2】

10

図 1 9 は、SubPath_typeの意味を示す図である。

【0 1 7 3】

図 1 9 に示す各値のうちの主なものについて説明すると、SubPath_typeの値が8であることは、D1 view videoを再生するサブパスであることを表す。

【0 1 7 4】

また、SubPath_typeの値が9であることは、D2 view videoを再生するサブパスであることを表す。

【0 1 7 5】

図 1 7 の 1 3 行目のref_to_clpi_file_nameは図 1 4 のClip_Information_file_name[0]に対応する。

20

【0 1 7 6】

このSubPathがD1 view videoを再生するものである場合には、ref_to_clpi_file_nameは、D1 view videを格納したm2tsファイルのファイル名のうちの、「.m2ts」の拡張子を除いた 5 桁の数字を表す。この記述により、参照するclpiファイルが特定される。

【0 1 7 7】

一方、1 3 行目のref_to_clpi_file_nameは、このSubPathがD2 view videoを再生するものである場合には、D2 view videを格納したm2tsファイルのファイル名のうちの、「.m2ts」の拡張子を除いた 5 桁の数字を表す。

【0 1 7 8】

1 6 行目から 3 0 行目は、interleaved_file_info()、すなわちilvtファイルに関する記述である。例えば、PlayItem()、SubPath()中のreserved_for_future_useを用いてilvtファイルに関する記述が用意される。

30

【0 1 7 9】

1 7 行目から 2 2 行目は、6 行目のtypeの値が1であり、Base view video、D1 view video、D2 view videoが全てインターリーブされている場合に参照する記述である。

【0 1 8 0】

1 8 行目のref_to_D1-B_interleaved_file_nameは、Base view videoとD1 view videoの再生用のilvtファイルのファイル名のうちの、「.ilvt」の拡張子を除いた 5 桁の数字を表す。

【0 1 8 1】

40

1 9 行目のref_to_D2-B_interleaved_file_nameは、Base view videoとD2 view videoの再生用のilvtファイルのファイル名のうちの、「.ilvt」の拡張子を除いた 5 桁の数字を表す。

【0 1 8 2】

2 0 行目のref_to_D1_clpi_file_nameは、D1 view videを格納したm2tsファイルのファイル名のうちの、「.m2ts」の拡張子を除いた 5 桁の数字を表す。この記述により、D1 view videのm2tsファイルの再生時に参照するclpiファイルが特定される。

【0 1 8 3】

2 1 行目のref_to_D2_clpi_file_nameは、D2 view videを格納したm2tsファイルのファイル名のうちの、「.m2ts」の拡張子を除いた 5 桁の数字を表す。この記述により、D2 vi

50

ew videのm2tsファイルの再生時に参照するclpiファイルが特定される。

【0184】

23行目から26行目は、6行目のtypeの値が2であり、Base view videoとD1 view videoがインターリーブされている場合に参照する記述である。

【0185】

24行目のref_to_D1-B_interleaved_file_nameは、Base view videoとD1 view videoの再生用のilvtファイルのファイル名のうちの、「.ilvt」の拡張子を除いた5桁の数字を表す。

【0186】

25行目のref_to_D1_clpi_file_nameは、D1 view videを格納したm2tsファイルのファイル名のうちの、「.m2ts」の拡張子を除いた5桁の数字を表す。この記述により、D1 view videのm2tsファイルの再生時に参照するclpiファイルが特定される。

【0187】

27行目から30行目は、6行目のtypeの値が3であり、Base view videoとD2 view videoがインターリーブされている場合に参照する記述になる。

【0188】

28行目のref_to_D2-B_interleaved_file_nameは、Base view videoとD2 view videoの再生用のilvtファイルのファイル名のうちの、「.ilvt」の拡張子を除いた5桁の数字を表す。

【0189】

29行目のref_to_D2_clpi_file_nameは、D2 view videを格納したm2tsファイルのファイル名のうちの、「.m2ts」の拡張子を除いた5桁の数字を表す。この記述により、D2 view videのm2tsファイルの再生時に参照するclpiファイルが特定される。

【0190】

このように、3D_PlayListには、データが光ディスク202上でインターリーブされている場合、D1 view video、D2 view videoについても、それらのClip AVストリームに対応するclpiファイルのファイル名を特定可能な情報が記述される。

【0191】

[再生装置201の構成例]

図20は、再生装置201の構成例を示すブロック図である。

【0192】

コントローラ251は、予め用意されている制御プログラムを実行し、再生装置201の全体の動作を制御する。

【0193】

例えば、コントローラ251は、ディスクドライブ252を制御し、3D再生用のPlayListファイルを読み出す。また、コントローラ251は、STN_tableに登録されているIDに基づいて、Main TSとSubTSを読み出させ、デコーダ部256に供給させる。

【0194】

ディスクドライブ252は、コントローラ251による制御に従って光ディスク202からデータを読み出し、読み出したデータを、コントローラ251、メモリ253、またはデコーダ部256に出力する。

【0195】

メモリ253は、コントローラ251が各種の処理を実行する上において必要なデータなどを適宜記憶する。

【0196】

ローカルストレージ254は例えばHDD(Hard Disk Drive)により構成される。ローカルストレージ254には、サーバ272からダウンロードされたD1/D2 view videoストリームなどが記録される。ローカルストレージ254に記録されているストリームもデコーダ部256に適宜供給される。

【0197】

10

20

30

40

50

インターネットインタフェース 255 は、コントローラ 251 からの制御に従ってネットワーク 271 を介してサーバ 272 と通信を行い、サーバ 272 からダウンロードしたデータをローカルストレージ 254 に供給する。

【0198】

サーバ 272 からは、光ディスク 202 に記録されているデータをアップデートさせるデータがダウンロードされる。ダウンロードした D1/D2 view video ストリームを光ディスク 202 に記録されている Base view video ストリームと併せて用いることができるようにすることにより、光ディスク 202 の内容とは異なる内容の 3D 再生を実現することが可能になる。D1/D2 view video ストリームがダウンロードされたとき、PlayList の内容も適宜更新される。

10

【0199】

デコーダ部 256 は、ディスクドライブ 252、またはローカルストレージ 254 から供給されたストリームをデコードし、得られたビデオ信号を表示装置 203 に出力する。オーディオ信号も所定の経路を介して表示装置 203 に出力される。

【0200】

操作入力部 257 は、ボタン、キー、タッチパネル、ジョグダイヤル、マウスなどの入力デバイスや、所定のリモートコマンドから送信される赤外線などの信号を受信する受信部により構成される。操作入力部 257 はユーザの操作を検出し、検出した操作の内容を表す信号をコントローラ 251 に供給する。

【0201】

図 21 は、デコーダ部 256 の構成例を示す図である。

20

【0202】

分離部 281 は、コントローラ 251 による制御に従って、ディスクドライブ 252 から供給されたデータを、Main TS を構成するパケットのデータと Sub TS を構成するパケットのデータに分離する。例えば、3D PlayList ファイルの STN_table() に記述されているストリーム ID に基づいて光ディスク 202 から読み出された TS が分離部 281 に供給される。

【0203】

分離部 281 は、分離した Main TS を構成するパケットをリードバッファ 282 に出力して記憶させ、Sub TS を構成するパケットをリードバッファ 285 に出力して記憶させる。

30

【0204】

また、分離部 281 は、ローカルストレージ 254 から供給された Sub TS を構成するパケットをリードバッファ 285 に出力して記憶させる。

【0205】

上述したように、サーバ 272 からダウンロードされた D1/D2 view video がローカルストレージ 254 に記憶されている場合がある。光ディスク 202 に記録されている Base view video とともに再生することが指示されたとき、Sub TS としての D1/D2 view video ストリームがローカルストレージ 254 から読み出され、分離部 281 に供給される。

【0206】

PID フィルタ 283 は、リードバッファ 282 に記憶されている Main TS を構成するパケットを、各パケットに設定されている PID に基づいて振り分ける。コントローラ 251 からは、Base view video を構成するパケットの PID、D1 view video を構成するパケットの PID、D2 view video を構成するパケットの PID がそれぞれ指定される。

40

【0207】

PID フィルタ 283 は、Main TS に含まれる Base view video のパケットをリードバッファ 282 から読み出し、ES バッファ 284 に出力して記憶させる。ES バッファ 284 には、Base view video のパケットからなる ES (Elementary Stream) が記憶される。

【0208】

また、PID フィルタ 283 は、Main TS に D1/D2 view video のパケットが多重化されてい

50

る場合、それらのパケットをPIDに基づいて抽出し、スイッチ 287 に出力する。

【0209】

PIDフィルタ 286 は、Sub TSに含まれるD1/D2 view videoのパケットをリードバッファ 285 から読み出し、スイッチ 287 に出力する。

【0210】

なお、ここではBase view video、D1/D2 view videoのビデオストリームの処理についてだけ説明しているが、図 5 を参照して説明したように、Main TSにはPGやIGなどのグラフィックスのデータが多重化されていることがある。同様に、図 6 を参照して説明したように、Sub TSには、D1/D2 view videoの他にも、字幕のデータやグラフィックのデータが多重化されていることがある。

10

【0211】

PIDフィルタ 283 とPIDフィルタ 286 は、適宜、それらのデータをもPIDに基づいて振り分け、所定の出力先に出力する。図 21 のPIDフィルタ 283 とPIDフィルタ 286 のブロック内に示す出力先の端子（丸）には、グラフィックのデータをデコードするデコーダなどが接続される。

【0212】

スイッチ 287 は、PIDフィルタ 283 から供給されたD1/D2 view videoのパケットをESバッファ 288 に出力して記憶させる。また、スイッチ 287 は、PIDフィルタ 286 から供給されたD1/D2 view videoのパケットをESバッファ 288 に出力して記憶させる。ESバッファ 288 には、D1/D2 view videoのパケットからなるESが記憶される。

20

【0213】

スイッチ 289 は、ESバッファ 284 に記憶されているBase view videoのパケットと、ESバッファ 288 に記憶されているD1/D2 view videoのパケットのうちの、デコードの対象になるパケットをデコーダ 290 に出力する。DTS(Decoding Time Stamp)などの時刻情報がBase view videoとD1/D2 view videoのPESパケットに設定されており、その時刻情報に基づいてバッファからの読み出しが行われる。

【0214】

ビデオデコーダ 290 は、スイッチ 289 から供給されたパケットをデコードし、デコードすることによって得られたBase view video、またはD1/D2 view videoのデータを出力する。

30

【0215】

[3D_PlayListの例 1]

図 22 は、3D_PlayListの例を示す図である。

【0216】

図 22 の「00000.mpls」のPlayListファイルに記述されている3D_PlayListは、Base view video、D1 view video、D2 view videoが全てインターリーブされている光ディスク 202 の再生を管理するPlayListである。すなわち、typeの値は1である。

【0217】

図 22 の例においては、PlayItem()のref_to_B_clpi_file_nameは「00001」である。この記述から、Base view videoのm2tsファイルである「00001.m2ts」を再生する場合には、図 10 の「00001.clpi」のclpiファイルを参照することが特定される。

40

【0218】

また、SubPath()[1]のSubPath_typeは「9」である。SubPath_typeが「9」であることは、この 1 つ目のSubPathがD2 view videoを再生するサブパスであることを表す。

【0219】

SubPath()[1]のref_to_clpi_file_nameは「00002」である。この記述から、D2 view videoを再生する場合には図 10 の「00002.clpi」のclpiファイルを参照することが特定される。

【0220】

SubPath()[2]のSubPath_typeは「8」である。SubPath_typeが「8」であることは、この

50

2 つ目のSubPathがD1 view videoを再生するサブパスであることを表す。

【0 2 2 1】

SubPath()[2]のref_to_clpi_file_nameは「00003」である。この記述から、D1 view videoを再生する場合には図 1 0 の「00003.clpi」のclpiファイルを参照することが特定される。

【0 2 2 2】

interleaved_file_info()のref_to_D1-B_interleaved_file_nameは「10000」である。この記述から、D1-B再生を行う場合には図 1 0 の「10000.ilvt」のilvtファイルを参照することが特定される。

【0 2 2 3】

また、ref_to_D2-B_interleaved_file_nameは「20000」である。この記述から、D2-B再生を行う場合には図 1 0 の「20000.ilvt」のilvtファイルを参照することが特定される。

【0 2 2 4】

ref_to_D1_clpi_file_nameは「00003」である。この記述から、D1 view videoを再生する場合には図 1 0 の「00003.clpi」のファイルを参照することが特定される。

【0 2 2 5】

ref_to_D2_clpi_file_nameは「00002」である。この記述から、D2 view videoを再生する場合には図 1 0 の「00002.clpi」のファイルを参照することが特定される。

【0 2 2 6】

図 2 3 は、図 2 2 の3D_PlayListとともに用いられるclpiファイルのシンタクスを示す図である。

【0 2 2 7】

図 2 3 A は「00001.clpi」のclpiファイルの例を示す図である。上述したように、「00001.clpi」のclpiファイルは、Base view videoのm2tsファイルである「00001.m2ts」を再生する場合に参照するファイルである。

【0 2 2 8】

number_of_source_packets1は、「00001.m2ts」のm2tsファイルに含まれるソースパケットの数を表す。

【0 2 2 9】

EP_mapは、「00001.m2ts」のm2tsファイルに含まれるTS中のエントリーポイント(EP)の位置情報を表す。

【0 2 3 0】

chunk_map()は、「00001.m2ts」のm2tsファイルに含まれるTSについて、先頭のチャンクから順番に、各チャンクの開始位置を表すSource Packet Number (SPN) を表す。

【0 2 3 1】

チャンクは、1 つのTSに属し、光ディスク 2 0 2 上に連続配置されるSource packetの集まりである。ここでは、1 つのチャンクが、光ディスク 2 0 2 上に配置される 1 つのエクステンツに相当するものとして説明する。

【0 2 3 2】

chunk_map()が、各チャンクの長さを表すことになる。chunk_map()の具体例については後述する。

【0 2 3 3】

図 2 3 B は「00002.clpi」のclpiファイルの例を示す図である。「00002.clpi」のclpiファイルは、D2 view videoを再生する場合に参照するファイルである。

【0 2 3 4】

図 2 3 C は「00003.clpi」のclpiファイルの例を示す図である。「00003.clpi」のclpiファイルは、D1 view videoを再生する場合に参照するファイルである。図 2 3 B、図 2 3 C の記述内容は図 2 3 A に示すものと同じである。

【0 2 3 5】

図 2 4 は、図 2 2、図 2 3 のデータを用いて行われるファイル管理の概念を示す図であ

10

20

30

40

50

る。

【0236】

図24に示すように、ファイル管理は、物理レイヤ、ファイルシステムレイヤ、アプリケーションレイヤの3層構造の形で行われる。図22の3D_PlayList、図23のclpiファイルがアプリケーションレイヤの情報になる。

【0237】

物理レイヤは、Base view video、D1 view video、D2 view videoが全てインターリーブされた状態で記録されている光ディスク202のレイヤとなる。

【0238】

図24の例においては、D1 view videoのチャンク、D2 view videoのチャンク、Base view videoのチャンクがその順番で配置されている。図24において、「B」の文字を付けているブロックはBase view videoのチャンクを表し、「D1」の文字を付けているブロックはD1 view videoのチャンクを表す。「D2」の文字を付けているブロックはD2 view videoのチャンクを表す。 10

【0239】

このように、Base view video、D1 view video、D2 view videoの各エクステント（チャンク）は、光ディスク202上においてインターリーブ配置されている。インターリーブ配置は、同一種類のストリームのエクステントが隣接しない、周期的な配置を意味する。

【0240】

ファイルシステムレイヤにおいては、アプリケーションがファイル名で指定したストリームファイル（m2tsファイル、ilvtファイル）と、光ディスク202上の各チャンクが対応付けられる。ファイルシステムは例えばUDF file systemである。 20

【0241】

図24に示すように、「00001.m2ts」のm2tsファイルは、光ディスク202上に配置されたBase view videoのチャンクから構成される。

【0242】

また、「00002.m2ts」のm2tsファイルは、光ディスク202上に配置されたD2 view videoのチャンクから構成される。

【0243】

「00003.m2ts」のm2tsファイルは、光ディスク202上に配置されたD1 view videoのチャンクから構成される。 30

【0244】

「20000.ilvt」のファイルは、光ディスク202上に配置されたD2 view videoのチャンクとBase view videoのチャンクから構成される。

【0245】

「10000.ilvt」のファイルは、光ディスク202上に配置されたD1 view videoのチャンクとBase view videoのチャンクから構成される。

【0246】

2D再生を行うために、アプリケーションから「00001.m2ts」を指定してデータの読み出しが指示された場合、ファイルシステムによる管理に従って、Base view videoのチャンクが読み出される。 40

【0247】

また、B-D1再生を行うために、アプリケーションから「10000.ilvt」を指定してデータの読み出しが指示された場合、ファイルシステムによる管理に従って、D1 view videoのチャンクとBase view videoのチャンクが読み出される。

【0248】

B-2D再生を行うために、アプリケーションから「20000.ilvt」を指定してデータの読み出しが指示された場合、ファイルシステムによる管理に従って、D2 view videoのチャンクとBase view videoのチャンクが読み出される。 50

【0249】

[動作例1]

ここで、図25のフローチャートを参照して、図22の3D_PlayListファイルに従って行われる再生処理について説明する。

【0250】

ステップS1において、コントローラ251は、typeの値から、Base view video、D1 view video、D2 view videoが全てインターリーブされていることを確認する。

【0251】

この場合、ステップS2において、コントローラ251は、interleaved_file_info()を読みに行く。

10

【0252】

ステップS3において、コントローラ251は、ユーザによる操作などに基づいて、B-D1再生が指示されたか否かを判定する。

【0253】

B-D1再生が指示されたステップS3において判定した場合、ステップS4において、コントローラ251は、interleaved_file_info()に記述された「10000.ilvt」(ref_to_D1-B_interleaved_file_name)を指定し、UDF file systemを通して、Base view videoのチャンクとD1 view videoのチャンクを光ディスク202から読み出させる。

【0254】

ディスクドライブ252により読み出されたBase view videoのチャンクとD1 view videoのチャンクはデコーダ部256の分離部281に供給される。

20

【0255】

ステップS5において、分離部281は、図23の「00001.clpi」(ref_to_B_clpi_file_name)のchunk_map()と、「00003.clpi」(ref to D1_clpi_file_name)のchunk_map()に基づいて、供給されたデータを、「00001.m2ts」のm2tsファイルのデータと、「00003.m2ts」のm2tsファイルのデータに分離する。分離部281は、「00001.m2ts」のm2tsファイルのデータをリードバッファ282に出力し、「00003.m2ts」のm2tsファイルのデータをリードバッファ285に出力する。chunk_map()を使って行われるデータの分離については後述する。

【0256】

30

リードバッファ282に記憶された「00001.m2ts」のm2tsファイルのデータは、PIDフィルタ283、ESバッファ284、スイッチ289を介してデコーダ290に供給される。リードバッファ285に記憶された「00003.m2ts」のm2tsファイルのデータは、PIDフィルタ286、スイッチ287、ESバッファ288、およびスイッチ289を介してデコーダ290に供給される。

【0257】

ステップS6において、デコーダ290は、スイッチ289から順次供給されるパケットをデコードする(再生する)。

【0258】

一方、ステップS3において、B-D1再生が指示されていない、すなわちB-D2再生が指示された場合、ステップS7において、コントローラ251は、interleaved_file_info()に記述された「20000.ilvt」(ref_to_D2-B_interleaved_file_name)を指定し、UDF file systemを通して、Base view videoのチャンクとD2 view videoのチャンクを光ディスク202から読み出させる。

40

【0259】

ステップS8において、分離部281は、「00001.clpi」(ref_to_B_clpi_file_name)のchunk_map()と、「00002.clpi」(ref to D2_clpi_file_name)のchunk_map()に基づいて、供給されたデータを、「00001.m2ts」のm2tsファイルのデータと、「00002.m2ts」のm2tsファイルのデータに分離する。分離部281は、「00001.m2ts」のm2tsファイルのデータをリードバッファ282に出力し、「00002.m2ts」のm2tsファイルのデータをリードバ

50

ッファ 2 8 5 に出力する。

【0 2 6 0】

その後、「00001.m2ts」のm2tsファイルのデータと「00002.m2ts」のm2tsファイルのデータは、B-D1再生のときと同様にデコーダ 2 9 0 に供給され、ステップ S 6 において再生される。

【0 2 6 1】

[chunk_map()を使ったデータの分離]

図 2 6 は、chunk_map()のシンタクスの例を示す図である。

【0 2 6 2】

number_of_chunksは、参照するチャンクの数を表す。number_of_chunks以降には、こ
こで指定される数だけチャンクの情報が記述される。

10

【0 2 6 3】

SPN_chunk_start[i]は、例えば先頭のチャンクの開始位置を基準として、その基準の位置から各チャンクの開始位置までのSPN（長さ）を表す。各チャンクの開始位置までのSPNが、先頭のチャンクのものから順に記述される。

【0 2 6 4】

図 2 7 は、chunk_map()の具体例を示す図である。

【0 2 6 5】

図 2 7 A は「00001.clpi」のclpiファイルに記述されるchunk_map()であり、number_of_chunksは(n+1)である。

20

【0 2 6 6】

また、SPN_chunk_start[i]は0,c1,c2,⋯,cnである。

【0 2 6 7】

1 番目の値0は、図 2 8 C に示すように、「00001.m2ts」のm2tsファイルに含まれるBase view videoの先頭のチャンクの開始位置を基準として、その基準の位置から、1 番目のチャンク (B[0]) の開始位置までのSPNが0であることを表す。

【0 2 6 8】

2 番目の値c1は、基準の位置から、2 番目のチャンク (B[1]) の開始位置までのSPNがc1であることを表す。

【0 2 6 9】

30

3 番目の値c2は、基準の位置から、3 番目のチャンク (B[2]) の開始位置までのSPNがc2であることを表す。

【0 2 7 0】

n+1番目の値cnは、基準の位置から、最後のチャンクであるn+1番目のチャンク (B[n]) の開始位置までのSPNがcnであることを表す。

【0 2 7 1】

図 2 7 B は「00002.clpi」のclpiファイルに記述されるchunk_map()であり、number_of_chunksは(n+1)である。

【0 2 7 2】

また、SPN_chunk_start[i]は0,b1,b2,⋯,bnである。

40

【0 2 7 3】

1 番目の値0は、図 2 8 B に示すように、「00002.m2ts」のm2tsファイルに含まれるD2 view videoの先頭のチャンクの開始位置を基準として、その基準の位置から、1 番目のチャンク (D2[0]) の開始位置までのSPNが0であることを表す。

【0 2 7 4】

2 番目の値b1は、基準の位置から、2 番目のチャンク (D2[1]) の開始位置までのSPNがb1であることを表す。

【0 2 7 5】

3 番目の値b2は、基準の位置から、3 番目のチャンク (D2[2]) の開始位置までのSPNがb2であることを表す。

50

【0276】

$n+1$ 番目の値 b_n は、基準の位置から、最後のチャンクである $n+1$ 番目のチャンク ($D2[n]$) の開始位置までのSPNが b_n であることを表す。

【0277】

図27Cは「00003.clpi」のclpiファイルに記述される`chunk_map()`であり、`number_of_chunks`は $(n+1)$ である。

【0278】

また、`SPN_chunk_start[i]`は $0, a_1, a_2, \dots, a_n$ である。

【0279】

1番目の値0は、図28Aに示すように、「00003.m2ts」のm2tsファイルに含まれるD1 view videoの先頭のチャンクの開始位置を基準として、その基準の位置から、1番目のチャンク ($D1[0]$) の開始位置までのSPNが0であることを表す。

10

【0280】

2番目の値 a_1 は、基準の位置から、2番目のチャンク ($D1[1]$) の開始位置までのSPNが a_1 であることを表す。

【0281】

3番目の値 a_2 は、基準の位置から、3番目のチャンク ($D1[2]$) の開始位置までのSPNが a_2 であることを表す。

【0282】

$n+1$ 番目の値 a_n は、基準の位置から、最後のチャンクである $n+1$ 番目のチャンク ($D1[n]$) の開始位置までのSPNが a_n であることを表す。

20

【0283】

$D1[i], D2[i], B[i]$ は、図28Dに示すようにその順番で、光ディスク202上に周期的に配置されている。

【0284】

分離部281は、光ディスク202から読み出されたデータがディスクドライブ252から供給された場合、図27の3つの`chunk_map()`の記述に基づいて、供給されたデータ先頭から a_1 に相当するSPN分のデータを $D1[0]$ として分離する。

【0285】

また、分離部281は、 $D1[0]$ の終わりの位置から b_1 に相当するSPN分のデータを $D2[0]$ として分離し、 $D2[0]$ の終わりの位置から c_1 に相当するSPN分のデータを $B[0]$ として分離する。

30

【0286】

分離部281は、 $B[0]$ の終わりの位置から $a_2 - a_1$ に相当するSPN分のデータを $D1[1]$ として分離する。

【0287】

分離部281は、 $D1[1]$ の終わりの位置から $b_2 - b_1$ に相当するSPN分のデータを $D2[1]$ として分離し、 $D2[1]$ の終わりの位置から $c_2 - c_1$ に相当するSPN分のデータを $B[1]$ として分離する。

【0288】

なお、分離の対象になるチャンクは、B-D1再生を行う場合には $D1[i], B[i]$ だけであり、B-D2再生を行う場合には $D2[i], B[i]$ だけである。

40

【0289】

このように、分離部281によるデータの分離は、`chunk_map()`に記述される各チャンクの長さの情報を用いて行われる。

【0290】

`chunk_map()`について補足する。

【0291】

`type=0`のとき、`ref_to_B_clpi_file_name`により参照されるclpiファイルについては、`chunk_map()`はオプションとなる(無くても良い)。そして、`chunk_map()`がある場合、プ

50

レーヤはそのchunk_map()を無視しなければならない。

【0292】

また、ローカルストレージ254上のm2tsファイルに対応するclpiファイルについては、chunk_map()はオプションとなる(無くても良い)。そして、chunk_map()がある場合、プレーヤはそのchunk_map()を無視しなければならない。

【0293】

type=1のとき、対応するBase view videoのTS、D1 view videoのTS、D2 view videoのTSの3本のTSは、それぞれ同じ数(n+1)のチャンクに分割される。すなわち、図28のD1[i], D2[i], B[i]について、添え字iの値が同じ組のチャンクは、同じ再生時間を持つように分割される。

10

【0294】

同様に、type=2のとき、対応するBase view videoのTS、D1 view videoのTSの2本のTSは、それぞれ同じ数(n+1)のチャンクに分割される。すなわち、インターリーブされるD1[i], B[i]について、添え字iの値が同じ組のチャンクは、同じ再生時間を持つように分割される。

【0295】

type=3のとき、対応するBase view videoのTS、D2 view videoのTSの2本のTSは、それぞれ同じ数(n+1)のチャンクに分割される。すなわち、インターリーブされるD2[i], B[i]について、添え字iの値が同じ組のチャンクは、同じ再生時間を持つように分割される。

【0296】

20

[3D_PlayListの例2]

図29は、3D_PlayListの他の例を示す図である。

【0297】

図29の「0000.mpls」のPlayListファイルに記述されている3D_PlayListは、Base view videoとD2 view videoがインターリーブされている光ディスク202の再生を管理するPlayListである。すなわち、typeの値は3である。

【0298】

SubPathの記述がD2 view videoを参照するSubPathの記述だけになる点と、interleaved_file_info()の記述が異なる点を除いて、図29の3D_PlayListの記述は図22の記述と同様である。

30

【0299】

すなわち、SubPath()のSubPath_typeは「9」である。SubPath_typeが「9」であることは、このSubPathがD2 view videoを再生するサブパスであることを表す。

【0300】

また、ref_to_clpi_file_nameは「00002」である。

【0301】

図29のinterleaved_file_info()のref_to_D2-B_interleaved_file_nameは「20000」である。この記述から、D2-B再生を行う場合には図10の「20000.ilvt」のilvtファイルを参照することが特定される。

【0302】

40

また、ref_to_D2_clpi_file_nameは「00002」である。この記述から、D2 view videoを再生する場合には図10の「00002.clpi」のclpiファイルを参照することが特定される。

【0303】

図30は、図29の3D_PlayListとともに用いられるclpiファイルのシンタクスを示す図である。

【0304】

図30Aは、「00001.clpi」のclpiファイルの例を示す図であり、図30Bは、「00002.clpi」のclpiファイルの例を示す図である。いずれのclpiファイルにも、EP_mapと、上述したchunk_map()の記述が含まれる。

【0305】

50

図 3 1 は、図 2 9、図 3 0 のデータを用いて行われるファイル管理の概念を示す図である。

【0306】

図 3 1 に示すように、物理レイヤは、Base view video と D2 view video がインターリーブされた状態で記録されている光ディスク 2 0 2 のレイヤとなる。

【0307】

「00001.m2ts」の m2ts ファイルは、光ディスク 2 0 2 上に配置された Base view video のチャンクから構成される。

【0308】

また、「00002.m2ts」の m2ts ファイルは、光ディスク 2 0 2 上に配置された D2 view video のチャンクから構成される。 10

【0309】

「20000.ilvt」の ilvt ファイルは、光ディスク 2 0 2 上に配置された D2 view video のチャンクと Base view video のチャンクから構成される。

【0310】

2D再生を行うために、アプリケーションから「00001.m2ts」を指定してデータの読み出しが指示された場合、ファイルシステムによる管理に従って、Base view video のチャンクが読み出される。

【0311】

B-D2再生を行うために、アプリケーションから「20000.ilvt」を指定してデータの読み出しが指示された場合、ファイルシステムによる管理に従って、D2 view video のチャンクと Base view video のチャンクが読み出される。 20

【0312】

[動作例 2]

図 3 2 のフローチャートを参照して、図 2 9 の 3D_PlayList ファイルに従って行われる再生処理について説明する。

【0313】

ステップ S 1 1 において、コントローラ 2 5 1 は、type の値から、Base view video、D2 view video がインターリーブされていることを確認する。

【0314】

この場合、ステップ S 1 2 において、コントローラ 2 5 1 は、interleaved_file_info() を読みに行く。 30

【0315】

B-D2再生が指示された場合、ステップ S 1 3 において、コントローラ 2 5 1 は、interleaved_file_info() に記述された「20000.ilvt」(ref_to_D2-B_interleaved_file_name) を指定し、UDF file system を通して、Base view video のチャンクと D2 view video のチャンクを光ディスク 2 0 2 から読み出させる。

【0316】

ディスクドライブ 2 5 2 により読み出された Base view video のチャンクと D2 view video のチャンクはデコーダ部 2 5 6 の分離部 2 8 1 に供給される。 40

【0317】

ステップ S 1 4 において、分離部 2 8 1 は、「00001.clpi」(ref_to_B_clpi_file_name) の chunk_map() と、「00002.clpi」(ref_to_D2_clpi_file_name) の chunk_map() に基づいて、供給されたデータを、「00001.m2ts」の m2ts ファイルのデータと、「00002.m2ts」の m2ts ファイルのデータに分離する。分離部 2 8 1 は、「00001.m2ts」の m2ts ファイルのデータをリードバッファ 2 8 2 に出力し、「00002.m2ts」の m2ts ファイルのデータをリードバッファ 2 8 5 に出力する。

【0318】

リードバッファ 2 8 2 に記憶された「00001.m2ts」の m2ts ファイルのデータは、PID フィルタ 2 8 3、ES バッファ 2 8 4、スイッチ 2 8 9 を介してデコーダ 2 9 0 に供給される 50

。一方、リードバッファ 285 に記憶された「00002.m2ts」のm2tsファイルのデータは、PIDフィルタ 286、スイッチ 287、ESバッファ 288、およびスイッチ 289 を介してデコーダ 290 に供給される。

【0319】

ステップ S15 において、デコーダ 290 は、スイッチ 289 から順次供給されるパケットをデコードする。

【0320】

[3D_PlayListの例3]

図 33 は、3D_PlayListのさらに他の例を示す図である。

【0321】

図 33 の「00000.mpls」のPlayListファイルに記述されている3D_PlayListは、光ディスク 202 に記録されているBase view video、D2 view videoと、ローカルストレージ 254 に記録されているD1 view videoの再生を管理するPlayListである。光ディスク 202 上では、Base view videoとD2 view videoはインターリーブされている。

【0322】

光ディスク 202 上のデータの配置のタイプを表すものであるから、PlayItem()のtypeの値は3である。

【0323】

SubPath()[1]のSubPath_typeは「9」である。SubPath_typeが「9」であることは、この1つ目のSubPathがD2 view videoを再生するサブパスであることを表す。

【0324】

SubPath()[1]のref_to_clpi_file_nameは「00002」である。この記述から、D2 view videoを再生する場合には図 10 の「00002.clpi」のclpiファイルを参照することが特定される。

【0325】

SubPath()[2]のSubPath_typeは「8」である。SubPath_typeが「8」であることは、この2つ目のSubPathがD1 view videoを再生するサブパスであることを表す。

【0326】

SubPath()[2]のref_to_clpi_file_nameは「00003」である。この記述から、D1 view videoを再生する場合には、ローカルストレージ 254 に記録されている「00003.clpi」のclpiファイルを参照することが特定される。

【0327】

この2つ目のSubPathに関する記述は、例えば、D1 view videoがダウンロードされたときに追加される。

【0328】

interleaved_file_info()のref_to_D2-B_interleaved_file_nameは「20000」である。この記述から、B-D2再生を行う場合には、図 10 の「20000.ilvt」のilvtファイルを参照することが特定される。

【0329】

また、ref_to_D2_clpi_file_nameは「00002」である。この記述から、D2 view videoを再生する場合には図 10 の「00002.clpi」のファイルを参照することが特定される。

【0330】

なお、ローカルストレージ 254 上ではD1 view videoはインターリーブされていないから、D1 view videoに関するilvtファイルは不要である。

【0331】

図 34 は、図 33 の3D_PlayListとともに用いられるclpiファイルのシンタクスを示す図である。

【0332】

図 34 A は、「00001.clpi」のclpiファイルの例を示す図であり、図 34 B は、「00002.clpi」のclpiファイルの例を示す図である。いずれのclpiファイルにも、EP_mapとchun

10

20

30

40

50

k_map()の記述が含まれる。

【0333】

図35は、図33、図34のデータを用いて行われるファイル管理の概念を示す図である。

【0334】

図35に示すように、物理レイヤは、Base view videoとD2 view videoがインターリーブされた状態で記録されている光ディスク202と、2番目のSubPathが参照するD1 view videoのファイルが記録されているローカルストレージ254のレイヤとなる。

【0335】

図35の例においては、D1 view videoを格納したm2tsファイルのファイル名は「00003.m2ts」である。また、「00003.m2ts」に対応するclpiファイルのファイル名は「00003.clpi」である。 10

【0336】

「00001.m2ts」のm2tsファイルは、光ディスク202上に配置されたBase view videoのチャンクから構成される。

【0337】

また、「00002.m2ts」のm2tsファイルは、光ディスク202上に配置されたD2 view videoのチャンクから構成される。

【0338】

「20000.ilvt」のilvtファイルは、光ディスク202上に配置されたD2 view videoのチャンクとBase view videoのチャンクから構成される。 20

【0339】

2D再生を行うために、アプリケーションから「00001.m2ts」を指定してデータの読み出しが指示された場合、ファイルシステムによる管理に従って、Base view videoのチャンクが読み出される。

【0340】

B-D1再生を行うために、アプリケーションから「00001.m2ts」を指定してデータの読み出しが指示された場合、ファイルシステムによる管理に従って、Base view videoのチャンクが読み出される。また、図33の3D_PlayListの2番目のSubPathの記述に従って、「00003.m2ts」を指定して、D1 view videoのm2tsファイルがローカルストレージ254から読み出される。 30

【0341】

B-D2再生を行うために、アプリケーションから「20000.ilvt」を指定してデータの読み出しが指示された場合、ファイルシステムによる管理に従って、D2 view videoのチャンクとBase view videoのチャンクが読み出される。

【0342】

[動作例3]

図36のフローチャートを参照して、図33の3D_PlayListファイルに従って行われる再生処理について説明する。

【0343】

ステップS21において、コントローラ251は、typeの値から、Base view video、D2 view videoがインターリーブされていることを確認する。

【0344】

この場合、ステップS22において、コントローラ251は、interleaved_file_info()を読みに行く。

【0345】

ステップS23において、コントローラ251は、B-D1再生が指示されたか否かを判定する。

【0346】

B-D1再生を行う場合、光ディスク202に記録されているデータとローカルストレージ 50

254に記録されているデータが用いられる。一方、B-D2再生を行う場合、光ディスク202に記録されているデータが用いられる。

【0347】

ステップS23において、B-D1再生が指示されていない、すなわちB-D2再生が指示されたと判定した場合、ステップS24において、コントローラ251は、interleaved_file_info()に記述された「20000.ilvt」(ref_to_D2-B_interleaved_file_name)を構成するD2 view videoを含むClipの名前X「00002」(D2 view videoを含むm2tsファイルの名前のうちの拡張子を除いた部分)を取り出す。

【0348】

ステップS25において、コントローラ251は、SubPath_type=9(D2-view videoを再生するサブパス)が参照するClipの名前Y「00002」を取り出す。

10

【0349】

ステップS26において、コントローラ251は、YがXと同じであることにより、D2 view videoが「20000.ilvt」に含まれていることを認識する。ここでYがXと異なるとき、D2 view videoを含むClipはローカルストレージ254上にあることになる。

【0350】

ステップS27において、コントローラ251は、interleaved_file_info()に記述された「20000.ilvt」(ref_to_D2-B_interleaved_file_name)を指定し、UDF file systemを通して、Base view videoのチャンクとD2 view videoのチャンクを光ディスク202から読み出させる。

20

【0351】

ディスクドライブ252により読み出されたBase view videoのチャンクとD2 view videoのチャンクはデコーダ部256の分離部281に供給される。

【0352】

ステップS28において、分離部281は、「00001.clpi」(ref_to_B_clpi_file_name)のchunk_map()と、「00002.clpi」(ref to D2_clpi_file_name)のchunk_map()に基づいて、供給されたデータを、「00001.m2ts」のm2tsファイルのデータと、「00002.m2ts」のm2tsファイルのデータに分離する。分離部281は、「00001.m2ts」のm2tsファイルのデータをリードバッファ282に出力し、「00002.m2ts」のm2tsファイルのデータをリードバッファ285に出力する。

30

【0353】

リードバッファ282に記憶された「00001.m2ts」のm2tsファイルのデータは、PIDフィルタ283、ESバッファ284、スイッチ289を介してデコーダ290に供給される。一方、リードバッファ285に記憶された「00002.m2ts」のm2tsファイルのデータは、PIDフィルタ286、スイッチ287、ESバッファ288、およびスイッチ289を介してデコーダ290に供給される。

【0354】

ステップS29において、デコーダ290は、スイッチ289から順次供給されるパケットをデコードする。

【0355】

40

一方、B-D1再生が指示されたとステップS23において判定した場合、ステップS30において、コントローラ251は、interleaved_file_info()に記述された「20000.ilvt」(ref_to_D2-B_interleaved_file_name)を構成するD2 view videoを含むClipの名前X「00002」を取り出す。

【0356】

ステップS31において、コントローラ251は、SubPath_type=8(D1-view videoを再生するサブパス)が参照するClipの名前Y「00003」を取り出す。

【0357】

ステップS32において、コントローラ251は、Yが「00001.clpi」(ref_to_B_clpi_file_name)のうちの拡張子を除いた部分と異なり、かつ、YがXと異なることにより、D1 v

50

iew videoのClipがローカルストレージ254上にあることを認識する。ここでYが「00001.clpi」のうちの拡張子を除いた部分と同じであるとき、または、YがXと同じあるとき、D1 view videoは「20000.ilvt」に含まれていることになる。

【0358】

ステップS33において、コントローラ251は、「00001.clpi」(ref_to_B_clpi_file_name)のEP_mapを使って、ディスクドライブ252に「00001.m2ts」のm2tsファイルを読み出させる。「00001.clpi」のEP_mapには、「00001.m2ts」のm2tsファイルのデコード開始位置となるエントリポイントの情報が含まれている。

【0359】

ステップS34において、コントローラ251は、「00003.clpi」(SubPath()[2]で参照されるファイル)のEP_mapを使って、ローカルストレージ254から「00003.m2ts」のm2tsファイルを読み出す。「00003.clpi」のEP_mapには、「00003.m2ts」のm2tsファイルのデコード開始位置となるエントリポイントの情報が含まれている。

【0360】

読み出されたBase view videoのチャンクとD1 view videoのチャンクはデコーダ部256の分離部281に供給される。

【0361】

光ディスク202から読み出された「00001.m2ts」のm2tsファイルのデータはリードバッファ282に記憶された後、PIDフィルタ283、ESバッファ284、スイッチ289を介してデコーダ290に供給される。

【0362】

また、ローカルストレージ254から読み出された「00003.m2ts」のm2tsファイルのデータはリードバッファ285に記憶された後、PIDフィルタ286、スイッチ287、ESバッファ288、およびスイッチ289を介してデコーダ290に供給される。

【0363】

ステップS29において、デコーダ290は、スイッチ289から順次供給されるパケットをデコードする。

【0364】

[chunk_map()を利用して「10000.ilvt」をランダムアクセス再生する方法]

図37は、図27を参照して説明したchunk_map()の内容をまとめて示す図である。

【0365】

各clpiファイルのchunk_map()に記述されるSPN_chunk_start(基準の位置からのSPN(長さ))を、縦方向にiをとって並べると図37に示すようなものになる。

【0366】

図38は、chunk_map()とともに各clpiファイルに記述されるEP_map()のシンタクスを示す図である。

【0367】

EP_map()はランダムアクセスなどを行うときのデコード開始位置を特定するために参照される。

【0368】

number_of_EP_entriesはEP(エントリポイント)の数を表す。

【0369】

number_of_EP_entries以降の記述が各EPについて用意される。PTS_EP_start[i]はEPのPTSを表し、SPN_EP_start[i]はEPのSPNを表す。このように、EP_mapには、各エントリポイントについてのPTSとSPNが対応付けて登録される。

【0370】

図39のフローチャートを参照して、再生装置201の処理について説明する。

【0371】

ここでは、図22の3D_PlayListを参照してB-D1再生を行い、ランダムアクセスを行う場合について説明する。

10

20

30

40

50

【0372】

ステップS41において、コントローラ251は、typeの値から、Base view video、D1 view video、D2 view videoが全てインターリーブされていることを確認する。

【0373】

この場合、ステップS42において、コントローラ251は、interleaved_file_info()を読みに行く。

【0374】

ステップS43において、コントローラ251は、interleaved_file_info()に記述された「10000.ilvt」(ref_to_D1-B_interleaved_file_name)が読み出しファイルであると判断する。

10

【0375】

「00000.mpls」の3D_PlayListファイルの時刻xから再生を開始する場合、ステップS44において、コントローラ251は、「00001.clpi」(ref_to_B_clpi_file_name)のEP_mapを用いて、xよりも小さく最も近い値を持つPTS_EP_start[m]を見つける。

【0376】

ステップS45において、コントローラ251は、PTS_EP_start[m]に対応するSPN_EP_start[m]を取り出す。図38を参照して説明したように、EP_mapにはPTS_EP_start[i]とSPN_EP_start[i]が対応付けて登録されている。

【0377】

図40は、ステップS44、S45の処理により特定される位置の例を示す図である。

20

【0378】

図40に示すように、時間軸上の時刻xから再生を開始する場合、xよりも小さく最も近い値を持つPTS_EP_start[m]がステップS44において特定される。また、PTS_EP_start[m]に対応するSPN_EP_start[m]がステップS45において特定される。

【0379】

ステップS46において、コントローラ251は、「00001.clpi」のchunk_mapを用いて、SPN_EP_start[m]より小さく最も近い値をもつSPN_chunk_start[k]を見つける。ステップS46の処理により特定されるSPN_chunk_start[k]を図41に示す。

【0380】

ステップS47において、コントローラ251は、「00001.clpi」のchunk_map()のSPN_chunk_start[k]と、「00003.clpi」(ref to D1_clpi_file_name)のchunk_map()のSPN_chunk_start[k]との和を、「10000.ilvt」の読み出し開始アドレスとして決定する。

30

【0381】

ここで決定される「10000.ilvt」の読み出し開始アドレスは、「10000.ilvt」の中でのD1[k]のチャンクの開始アドレスを示す。

【0382】

ステップS48において、コントローラ251は、「10000.ilvt」(ref_to_D1-B_interleaved_file_name)を指定し、ステップS47で決定したアドレスから、UDF file systemを通して、Base view videoのチャンクとD1 view videoのチャンクを読み出させる。

【0383】

読み出されたBase view videoのチャンクとD1 view videoのチャンクはデコーダ部256の分離部281に供給される。

40

【0384】

ステップS49において、分離部281は、「00001.clpi」(ref_to_B_clpi_file_name)のchunk_map()と、「00003.clpi」(ref to D1_clpi_file_name)のchunk_map()に基づいて、供給されたデータを、「00001.m2ts」のm2tsファイルのデータと、「00003.m2ts」のm2tsファイルのデータに分離する。分離部281は、「00001.m2ts」のm2tsファイルのデータをリードバッファ282に出力し、「00003.m2ts」のm2tsファイルのデータをリードバッファ285に出力する。

【0385】

50

リードバッファ 282 に記憶された「00001.m2ts」の m2ts ファイルのデータは、PID フィルタ 283、ES バッファ 284、スイッチ 289 を介してデコーダ 290 に供給される。リードバッファ 285 に記憶された「00003.m2ts」の m2ts ファイルのデータは、PID フィルタ 286、スイッチ 287、ES バッファ 288、およびスイッチ 289 を介してデコーダ 290 に供給される。

【0386】

ステップ S50 において、デコーダ 290 は、スイッチ 289 から順次供給されるパケットをデコードする。

【0387】

ilvt ファイルのランダムアクセスは以上のようにして行われる。

10

【0388】

[EP_map について]

ここで、EP_map について説明する。

【0389】

Base view video の EP_map について説明するが、D1/D2 view video についても、同様に EP_map が設定される。例えば、Base view video のあるピクチャにエントリポイントが設定された場合、D1/D2 view video の対応するピクチャにもエントリポイントが設定される。

【0390】

それぞれのストリームのピクチャをエンコード順／デコード順、あるいは表示順に並べたときに同じ位置にある Base view video のピクチャと D1/D2 view video のピクチャが、対応するピクチャとなる。

20

【0391】

図 42 は、光ディスク 202 上に記録された AV ストリームの構造を示す図である。

【0392】

Base view video ストリームを含む TS は、6144 バイトのサイズを有する整数個のアライドユニット (Aligned Unit) から構成される。

【0393】

アライドユニットは、32 個のソースパケット (Source Packet) からなる。ソースパケットは 192 バイトを有する。1 つのソースパケットは、4 バイトのトランスポートパケットエクストラヘッダ (TP_extra header) と、188 バイトのトランスポートパケット (Transport Packet) とからなる。

30

【0394】

Base view video のデータは、MPEG2 PES パケットにパケット化されている。PES パケットのデータ部に PES パケットヘッダが付加されて PES パケットが形成される。PES パケットヘッダには、PES パケットが伝送するエレメンタリストリームの種類を特定するストリーム ID が含まれる。

【0395】

PES パケットは、さらにトランスポートパケットにパケット化される。すなわち、PES パケットがトランスポートパケットのペイロードのサイズに分割され、ペイロードにトランスポートパケットヘッダが付加されてトランスポートパケットが形成される。トランスポートパケットヘッダは、ペイロードに格納されるデータの識別情報である PID を含む。

40

【0396】

なお、ソースパケットには、Clip AV ストリームの先頭を例えば 0 として、ソースパケット毎に 1 ずつ増加するソースパケット番号が与えられる。また、アライドユニットは、ソースパケットの第 1 バイト目から始まる。

【0397】

EP_map は、Clip のアクセスポイントのタイムスタンプが与えられたときに、Clip AV ストリームファイルの中でデータの読み出しを開始すべきデータアドレスを検索するために用いられる。EP_map は、エレメンタリストリームおよびトランスポートストリームから抽

50

出されたエントリポイントのリストである。

【0398】

EP_mapは、AVストリームの中で、デコードを開始すべきエントリポイントを検索するためのアドレス情報を持つ。EP_map中の1つのEPデータは、PTSと、PTSに対応するAccess Uniteの、AVストリーム中のアドレスとの対で構成される。AVC/H.264においては、1 Access Uniteには1ピクチャ分のデータが格納される。

【0399】

図43は、Clip AVストリームの例を示す図である。

【0400】

図43のClip AVストリームは、PID=xで識別されるソースパケットからなるビデオストリーム（Base view videoストリーム）である。ビデオストリームは、ソースパケット毎に、ソースパケット内のトランスポートパケットのヘッダに含まれるPIDにより区別される。

10

【0401】

図43においては、ビデオストリームのソースパケットのうちの、IDR(Instantaneous Decoding Refresh)ピクチャの先頭バイトを含むソースパケットに色が付されている。色が付いていない四角は、ランダムアクセスポイントとならないデータが含まれるソースパケットや、他のストリームのデータが含まれているソースパケットを示す。

【0402】

IDRピクチャはIピクチャであり、IDRピクチャを含むGOP内の中で最初にデコードされる。IDRピクチャのデコード時、参照ピクチャバッファの状態や、それまで管理されていたフレーム番号やPOC(Picture Order Count)などのデコードに関する全ての情報はリセットされる。

20

【0403】

例えば、PID=xで区別されるビデオストリームのランダムアクセス可能なIDRピクチャの先頭バイトを含む、ソースパケット番号X1のソースパケットは、Clip AVストリームの時間軸上でPTS=pts(x1)の位置に配置される。

【0404】

同様に、次にランダムアクセス可能なIDRピクチャの先頭バイトを含むソースパケットはソースパケット番号X2のソースパケットとされ、PTS=pts(x2)の位置に配置される。

30

【0405】

図44は、図43のClip AVストリームに対応したEP_mapの例を概念的に示す図である。

【0406】

図44に示すように、EP_mapは、stream_PID、PTS_EP_start、およびSPN_EP_startから構成される。

【0407】

stream_PIDは、ビデオストリームを伝送するトランスポートパケットのPIDを表す。

【0408】

PTS_EP_startは、ランダムアクセス可能なIDRピクチャから始まるAccess UniteのPTSを表す。

40

【0409】

SPN_EP_startは、PTS_EP_startの値により参照されるAccess Uniteの第1バイト目を含むソースパケットのアドレスを表す。

【0410】

ビデオストリームのPIDがstream_PIDに格納され、PTS_EP_startとSPN_EP_startの対応関係を表すテーブル情報であるEP_map_for_one_stream_PID()が生成される。

【0411】

例えば、PID=xのビデオストリームのEP_map_for_one_stream_PID[0]には、PTS=pts(x1)とソースパケット番号X1、PTS=pts(x2)とソースパケット番号X2、・・・、PTS=pts(xk)と

50

ソースパケット番号Xkとがそれぞれ対応して記述される。

【0412】

このようなテーブルが、同じClip AVストリームに多重化されたそれぞれのビデオストリームについても生成される。生成されたテーブルを含むEP_mapが、当該Clip AVストリームに対応するClip Informationファイルに格納される。

【0413】

図45は、SPN_EP_startが指すソースパケットのデータ構造の例を示す図である。

【0414】

上述したように、ソースパケットは、188バイトのトランスポートパケットに4バイトのヘッダを付加した形で構成される。トランスポートパケット部分は、ヘッダ部(TP header)とペイロード部とからなる。SPN_EP_startは、IDRピクチャから始まるAccess Uniteの第1バイト目を含むソースパケットのソースパケット番号を表す。

【0415】

AVC/H.264においては、Access Uniteすなわちピクチャは、AUデリミタ(Access Unit Delimiter)から開始される。AUデリミタの後に、SRSとPPSが続く。その後に、IDRピクチャのスライスのデータの、先頭部分または全体が格納される。

【0416】

トランスポートパケットのTPヘッダにあるpayload_unit_start_indicatorの値が1であることは、新たなPESパケットがこのトランスポートパケットのペイロードから始まることを表す。このソースパケットから、Access Uniteが開始されることになる。

【0417】

このようなEP_mapが、Base view videoストリームとDependent view videoストリームについてそれぞれ用意される。

【0418】

[記録装置の構成例]

図46は、ソフト製作処理部301の構成例を示すブロック図である。

【0419】

ビデオエンコーダ311は、図3のMVCエンコーダ211と同様の構成を有している。ビデオエンコーダ311は、複数の映像データをH.264 AVC/MVCでエンコードすることによってBase view videoストリームとDependent view videoストリームを生成し、バッファ312に出力する。

【0420】

オーディオエンコーダ313は、入力されたオーディオストリームをエンコードし、得られたデータをバッファ314に出力する。オーディオエンコーダ313には、Base view video、Dependent view videoストリームとともにディスクに記録させるオーディオストリームが入力される。

【0421】

データエンコーダ315は、PlayListファイルなどの、ビデオ、オーディオ以外の上述した各種のデータをエンコードし、エンコードして得られたデータをバッファ316に出力する。

【0422】

例えば、データエンコーダ315は、Base view videoストリームのデータとD1/D2 view videoストリームのデータがエクステンツ単位でインターリーブされた状態で光ディスクに記録されているのか否かを表すtype(図18)をPlayListファイルに設定する。

【0423】

また、データエンコーダ315は、Base view videoストリームのデータとD1/D2 view videoストリームのデータがインターリーブされた状態で光ディスクに記録される場合、上述したilvtファイルをPlayListファイルに設定する。ilvtファイルは、Base view videoストリームのデータとD1/D2 view videoストリームのデータとを仮想的にまとめて管理する仮想ファイルとして機能する。

10

20

30

40

50

【0424】

さらに、データエンコーダ315は、各ClipのClip Informationファイルのファイル名をPlayListファイルに設定したり、それぞれのClip Informationファイルに、EP_mapやchunk_mapを設定したりする。

【0425】

多重化部317は、それぞれのバッファに記憶されたビデオデータ、オーディオデータ、および、ストリーム以外のデータを同期信号と共に多重化し、誤り訂正符号化部318に出力する。

【0426】

誤り訂正符号化部318は、エラー訂正用のコードを多重化部317により多重化されたデータに付加する。

10

【0427】

変調部319は、誤り訂正符号化部318から供給されたデータに対して変調を施し、出力する。変調部319の出力は、再生装置201において再生可能な光ディスク202に記録されるソフトウェアとなる。

【0428】

このような構成を有するソフト製作処理部301が記録装置に設けられる。

【0429】

図47は、ソフト製作処理部301を含む構成の例を示す図である。

【0430】

20

図47に示す構成の一部が記録装置内に設けられることもある。

【0431】

ソフト製作処理部301により生成された記録信号はプリマスタリング処理部331においてマスタリング処理が施され、光ディスク202に記録すべきフォーマットの信号が生成される。生成された信号は原盤記録部333に供給される。

【0432】

記録用原盤製作部332においては、ガラスなどよりなる原盤が用意され、その上に、フォトレジストなどよりなる記録材料が塗布される。これにより、記録用原盤が製作される。

【0433】

30

原盤記録部333において、プリマスタリング処理部331から供給された記録信号に対応してレーザビームが変調され、原盤上のフォトレジストに照射される。これにより、原盤上のフォトレジストが記録信号に対応して露光される。その後、この原盤を現像し、原盤上にピットを出現させることが行われる。

【0434】

金属原盤製作部334において、原盤に電鋳等の処理が施され、ガラス原盤上のピットを転写した金属原盤が製作される。この金属原盤から、さらに金属スタンプが製作され、これが成形用金型とされる。

【0435】

成形処理部335において、成形用金型に、インジェクションなどによりPMMA（アクリル）またはPC（ポリカーボネート）などの材料を注入し、固定化させることが行われる。あるいは、金属スタンプ上に2P（紫外線硬化樹脂）などを塗布した後、紫外線を照射して硬化させることが行われる。これにより、金属スタンプ上のピットを、樹脂よりなるレプリカ上に転写することができる。

40

【0436】

成膜処理部336において、レプリカ上に、反射膜が蒸着あるいはスパッタリングなどにより形成される。あるいはまた、レプリカ上に、反射膜がスピコートにより形成される。

【0437】

後加工処理部337において、このディスクに対して内外径の加工が施され、2枚のデ

50

ィスクを張り合わせるなどの必要な処置が施される。さらに、ラベルを貼り付けたり、ハブを取り付けたりした後、カートリッジに挿入される。このようにして再生装置 201 によって再生可能なデータが記録された光ディスク 202 が完成する。

【0438】

上述した一連の処理は、ハードウェアにより実行することもできるし、ソフトウェアにより実行することもできる。一連の処理をソフトウェアにより実行する場合には、そのソフトウェアを構成するプログラムが、専用のハードウェアに組み込まれているコンピュータ、または、汎用のパーソナルコンピュータなどにインストールされる。

【0439】

インストールされるプログラムはリムーバブル記録媒体に記録して提供される。また、ローカルエリアネットワーク、インターネット、デジタル放送といった、有線または無線の伝送媒体を介して提供されるようにしてもよい。

【0440】

本技術の実施の形態は、上述した実施の形態に限定されるものではなく、本技術の要旨を逸脱しない範囲において種々の変更が可能である。

【符号の説明】

【0441】

201 再生装置, 202 光ディスク, 203 表示装置, 211 MVCエンコーダ, 221 H.264/AVCエンコーダ, 222 H.264/AVCデコーダ, 223 Depth算出部, 224 Dependent view videoエンコーダ, 225 マルチプレкса, 251 コントローラ, 252 ディスクドライブ, 253 メモリ, 254 ローカルストレージ, 255 インターネットインタフェース, 256 デコーダ部, 257 操作入力部

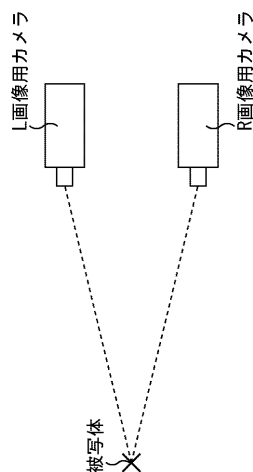
【図1】

図1



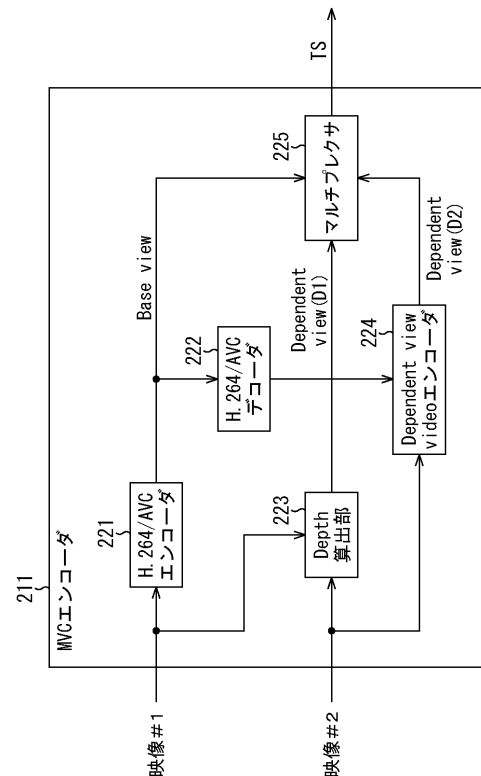
【図2】

図2



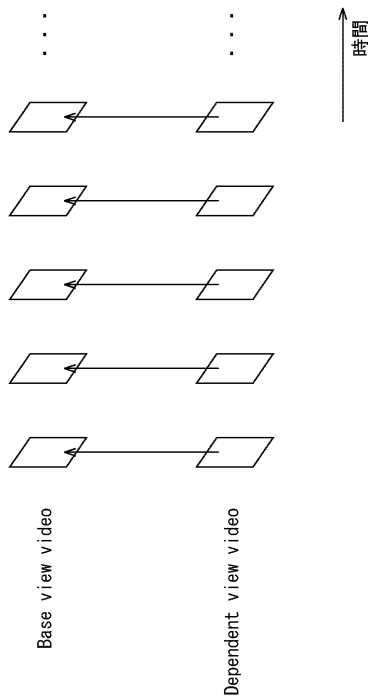
【図3】

図3



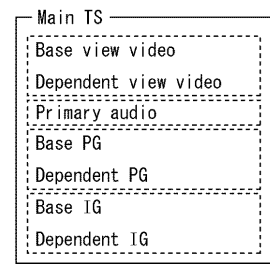
【図 4】

図4



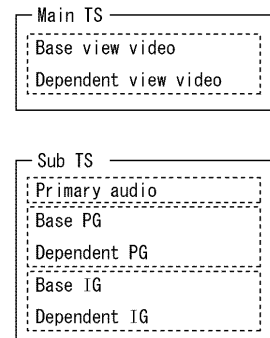
【図 5】

図5



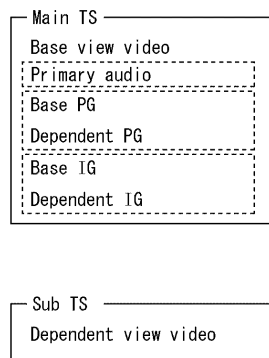
【図 6】

図6



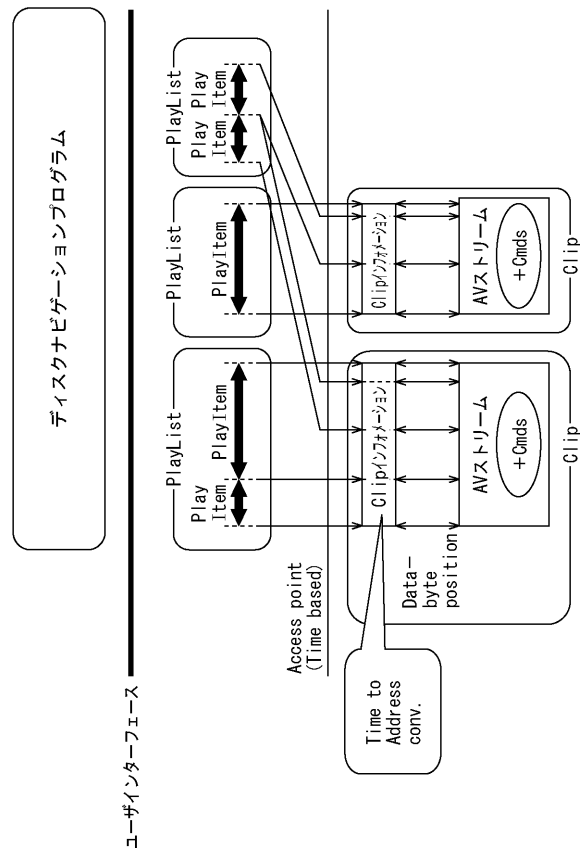
【図 7】

図7



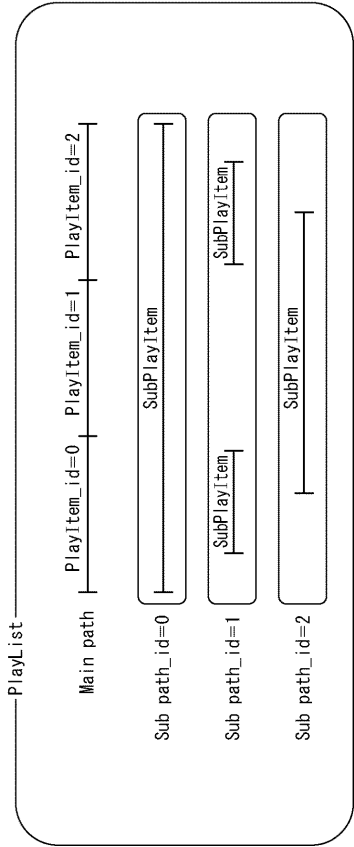
【図 8】

図8



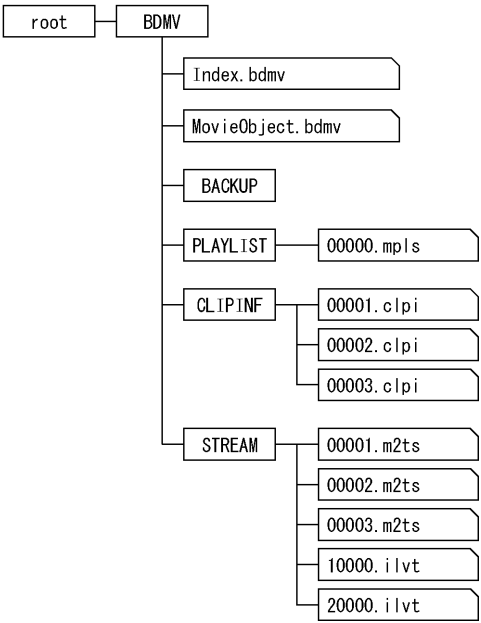
【図 9】

図9



【図 10】

図10



【図 11】

図11

Syntax	No. of bits	Mnemonic
xxxxx.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
}		
}		

【図 12】

図12

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()		
}		
}		

【図 13】

図13

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

【図 14】

図14

SubPlayItem(i)-Syntax	No. of bits	Mnemonic
Syntax		
SubPlayItem(i) {		
length	16	uimsbf
Clip_Information_file_name[0]	8*5	bslbf
Clip_codec_identifier[0]	8*4	bslbf
reserved_for_future_use	31	bslbf
is_multi_Clip_entries	1	bslbf
ref_to_STC_id[0]	8	uimsbf
SubPlayItem_IN_time	32	uimsbf
SubPlayItem_OUT_time	32	uimsbf
sync_start_PTS_of_PlayItem	16	uimsbf
if (is_multi_Clip_entries==1b) {	32	uimsbf
reserved_for_future_use	8	bslbf
num_of_Clip_entries	8	uimsbf
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	bslbf
Clip_codec_identifier[subclip_entry_id]	8*4	bslbf
ref_to_STC_id[subclip_entry_id]	8	uimsbf
reserved_for_future_use	8	bslbf
}		
}		

【図 15】

図15

PlayItem-Syntax	No. of bits	Mnemonic
Syntax		
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
UQ_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()		
}		

【図 16】

図16

STN_table()	No. of bits	Mnemonic
Syntax		
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()		
}		
}		

【図 17】

図17

```

1: 30.PlayList {
2:   number_of_PlayItems
3:   for (i=0; i<number_of_PlayItems; i++) {
4:     PlayItem() {
5:       ref_to_B_cipi_file_name // Base-view (B-view) videoを含むm2ts fileの拡張子を除いた5桁のdigit
6:       type
7:       SN_table()
8:     }
9:   }
10:   number_of_SubPaths
11:   for (i=0; i<number_of_SubPaths; i++) {
12:     SubPath() {
13:       SubPath_type
14:       ref_to_cipi_file_name
15:     }
16:   }
17:   interleaved_file_info() {
18:     if (type=="D1-D2-B are interleaved") {
19:       ref_to_D1-B_interleaved_file_name
20:       ref_to_D2-B_interleaved_file_name
21:       ref_to_D1_cipi_file_name
22:       ref_to_D2_cipi_file_name
23:     } else if (type=="D1-B are interleaved") {
24:       ref_to_D1-B_interleaved_file_name
25:       ref_to_D1_cipi_file_name
26:     } else (type=="D2-B are interleaved") {
27:       ref_to_D2-B_interleaved_file_name
28:       ref_to_D2_cipi_file_name
29:     }
30:   }
31: }
32: }

```

【図 18】

図18

type	Meaning
0	Not interleaved. A single m2ts that includes B-view and may includes D1 view and/or D2 view
1	D1-D2-B are interleaved.
2	D1-B are interleaved.
3	D2-B are interleaved.

【図 19】

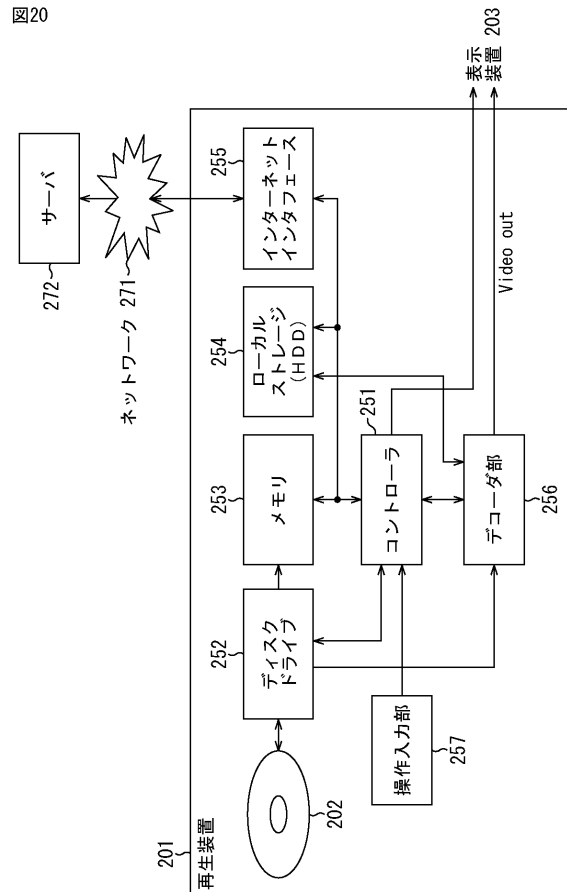
SubPath_type

図19

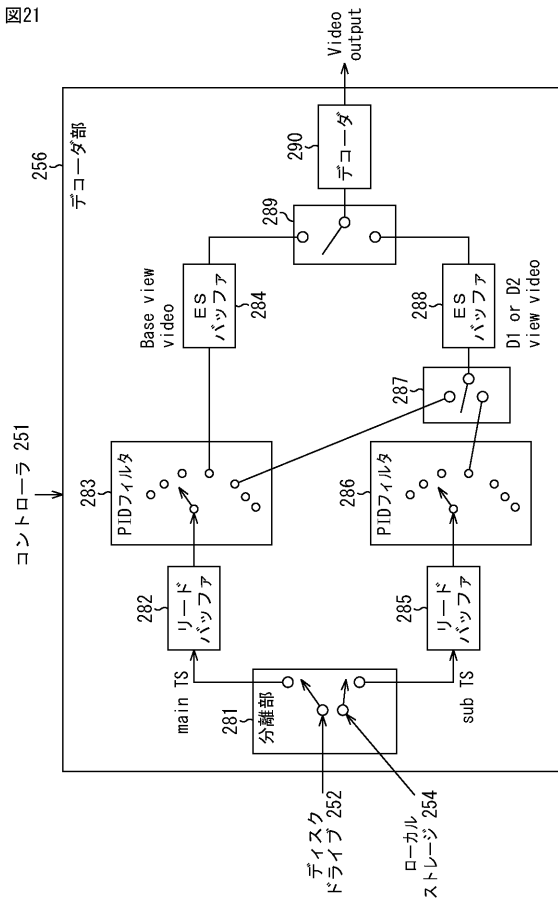
SubPath_type	Meaning
0	reserved
1	reserved
2	Out-of-mux and primary audio presentation path of the Browsible slideshow (The audio presentation path using the SubPath is not synchronized with the main path using PlayItems in the PlayList.)
3	Out-of-mux and interactive graphics menu (The interactive graphics menu using the SubPath is not synchronized with the main path using PlayItems in the PlayList.)
4	Out-of-mux and text subtitle presentation path (The text subtitle presentation path using the SubPath is synchronized with the main path using PlayItems in the PlayList.)
5	Out-of-mux and AV synchronized type of one or more elementary streams path (Primary audio/Pg/Ig/Secondary audio path.) Out-of-mux and AV synchronized type of Picture-in-Picture presentation path which contains one or more elementary stream paths. (The elementary streams used by the path are multiplexed in a separate Clip from other Clip used by PlayItem. The path using the SubPath is synchronized with the main path using PlayItems in the PlayList.)
6	Out-of-mux and AV non-synchronized type of Picture-in-Picture presentation path which contains one or more elementary stream paths. (The elementary streams used by the path are multiplexed in a separate Clip from other Clip used by PlayItem. The path using the SubPath is not synchronized with the main path using PlayItems in the PlayList.)
7	In-mux type and AV synchronized type of Picture-in-Picture presentation path which contains one or more elementary stream paths. (The elementary streams used by the path is multiplexed in the same Clip used by PlayItem. The path using the SubPath is synchronized with the main path using PlayItems in the PlayList.)
8	D1-view videoを再生するサブパス
9	D2-view videoを再生するサブパス
10-255	reserved

【図 20】

図20



【図 2 1】
図21



【図 2 2】
図22

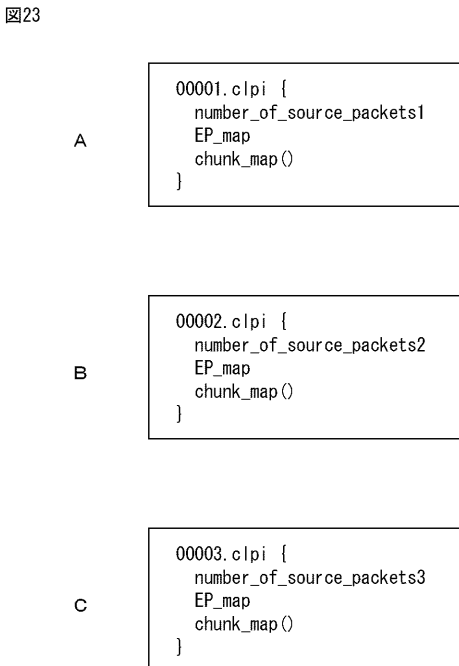
图22

```
3D_PlayList

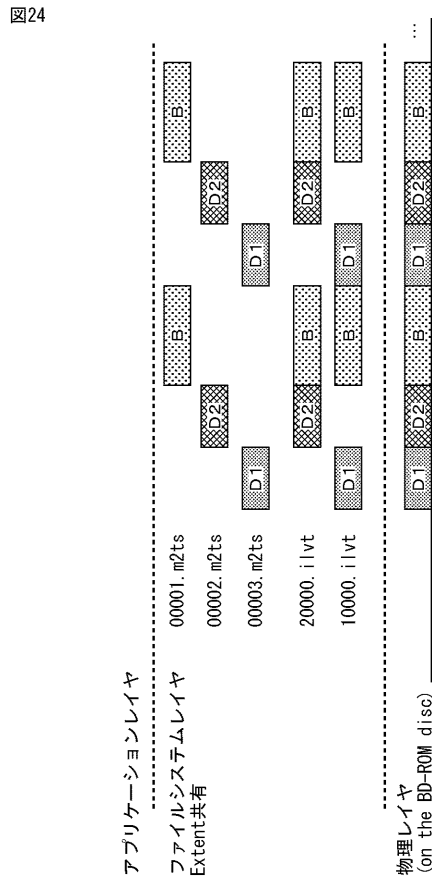
00000.mpls {
    PlayItem() {
        "00001"; // 00001.clpi (ref_to_B.clpi_file_name)
        type = "D1-D2-B are interleaved";
        STN_table()
    }
    SubPath0 [1] {
        SubPath_type=9; "00002"; // ref to 00002.clpi
    }
    SubPath0 [2] {
        SubPath_type=8; "00003"; // ref to 00003.clpi
    }
    interleaved_file_info() {
        10000; // 10000.ilvt (ref_to_D1-B_interleaved_file_name)
        20000; // 20000.ilvt (ref_to_D2-B_interleaved_file_name)
        00003; // 00003.clpi (ref_to_D1.clpi_file_name)
        00002; // 00002.clpi (ref_to_D2.clpi_file_name)
    }
}
```

【图 2 3】

图23

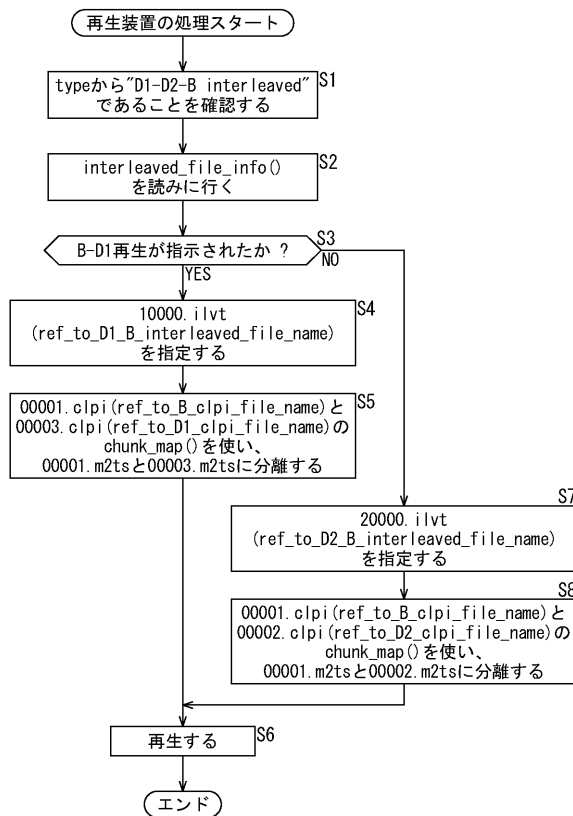


【图 2 4】
图24



【図 25】

図25



【図 26】

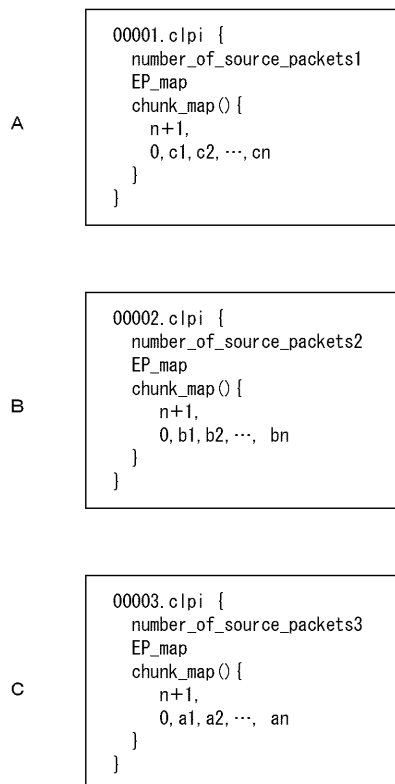
図26

```

chunk_map() {
    number_of_chunks
    for (i=0; i<number_of_chunks; i++) {
        SPN_chunk_start[i]
    }
}
  
```

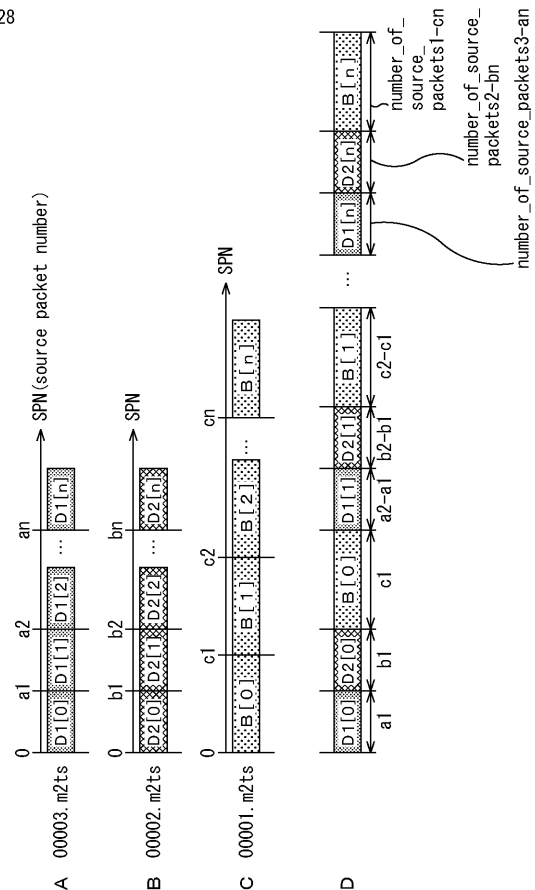
【図 27】

図27



【図 28】

図28



【図 29】

図29

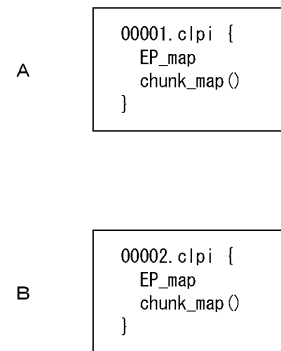
```

3D_PlayList
00000.mpls {
  PlayItem() {
    "00001"; // 00001.clpi (ref_to_B_clpi_file_name)
    type = "D2-B are interleaved";
    STN_table()
  }
  SubPath() {
    SubPath_type=9; "00002"; // ref to 00002.clpi
  }
  interleaved_file_info() {
    20000; // 20000.ilvt (ref_to_D2-B_interleaved_file_name)
    00002; // 00002.clpi (ref_to_D2_clpi_file_name)
  }
}

```

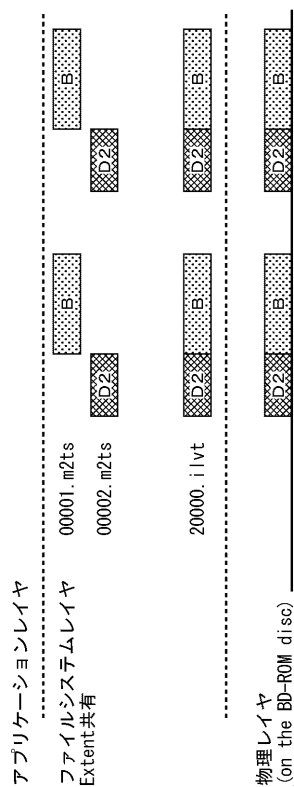
【図 30】

図30



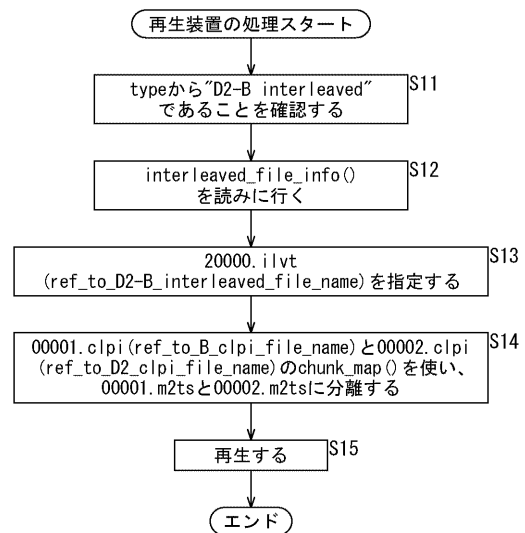
【図 31】

図31



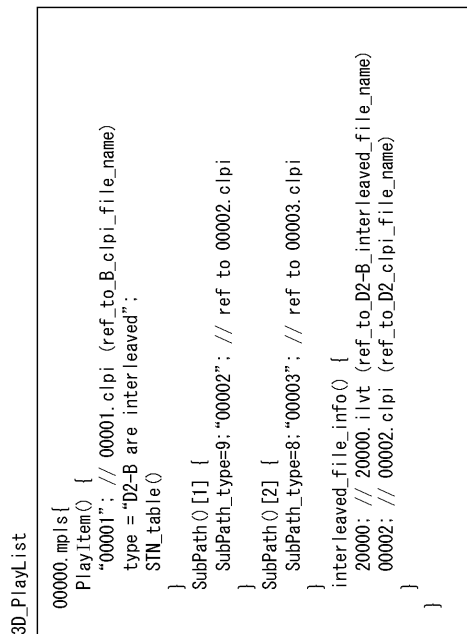
【図 32】

図32



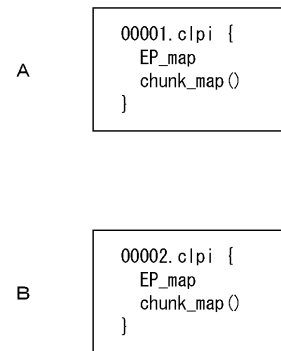
【図 33】

図33



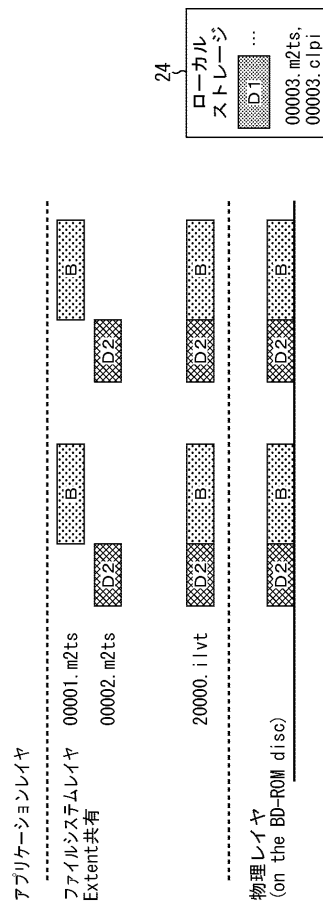
【図 34】

図34



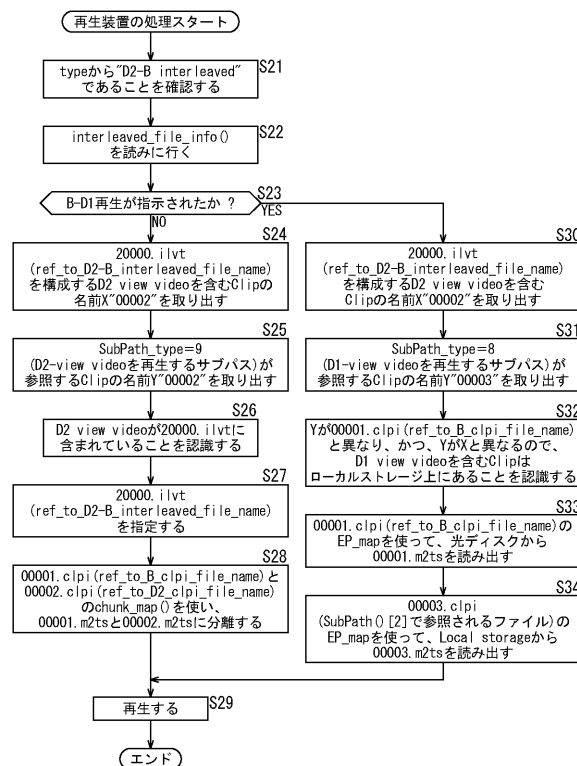
【図 35】

図35



【図 36】

図36



【図 37】

図37

i	SPN_chunk_start[i] of 00001.clpi	SPN_chunk_start[i] of 00002.clpi	SPN_chunk_start[i] of 00003.clpi
0	0	0	0
1	c1	b1	a1
2	c2	b2	a2
...	...	...	...
n	cn	bn	an

【図 38】

図38

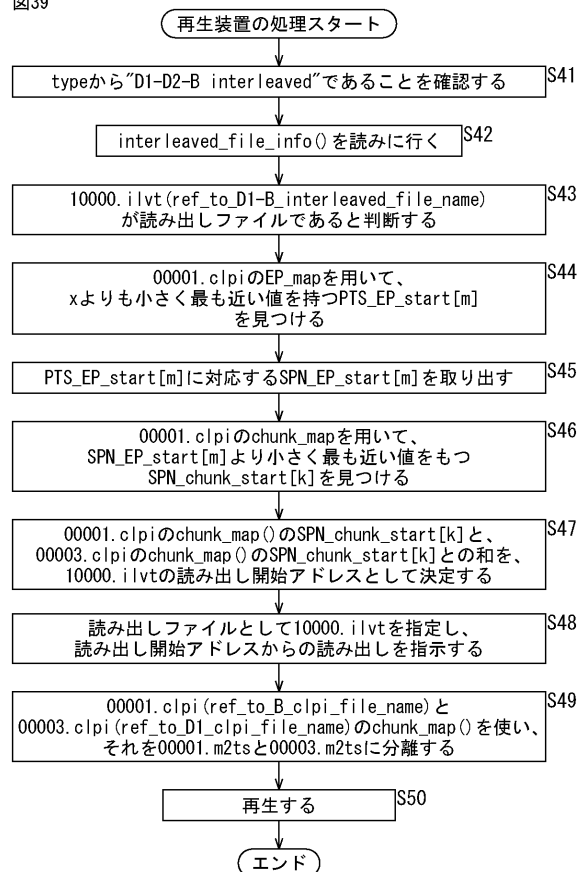
```

EP_map() {
    number_of_EP_entries
    for (i=0; i< number_of_EP_entries: i++) {
        PTS_EP_start[i]
        SPN_EP_start[i]
    }
}

```

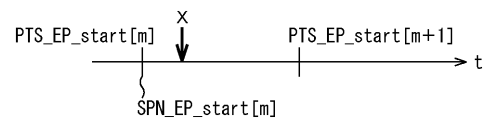
【図 39】

図39



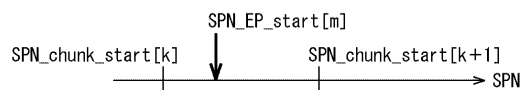
【図 40】

図40

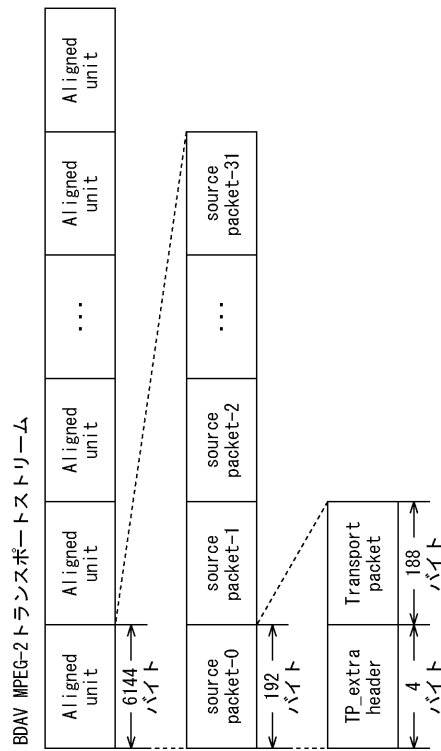


【図 41】

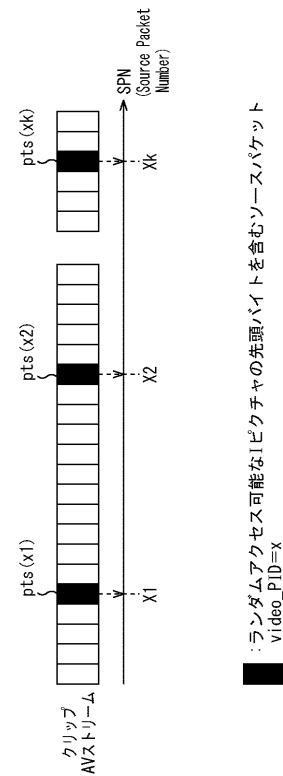
図41



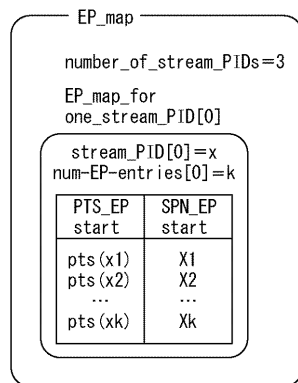
【図 4 2】
図42



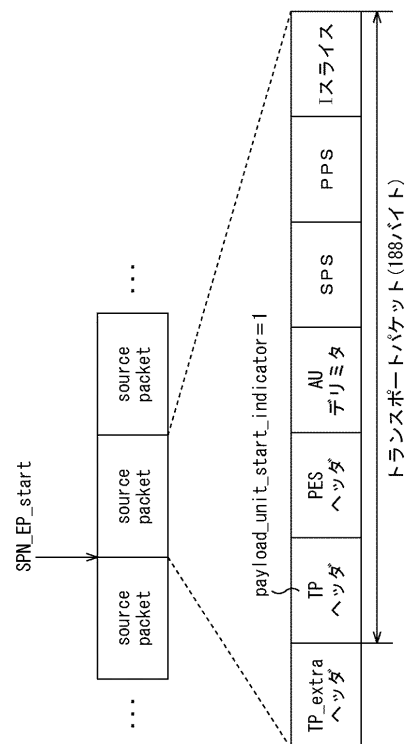
【図 4 3】
図43



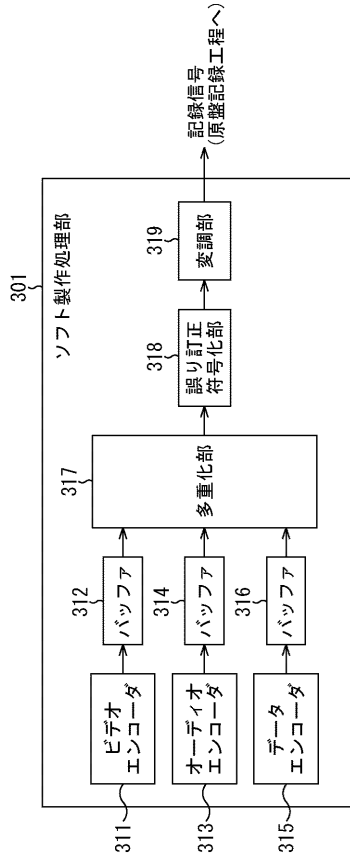
【図 4 4】
図44



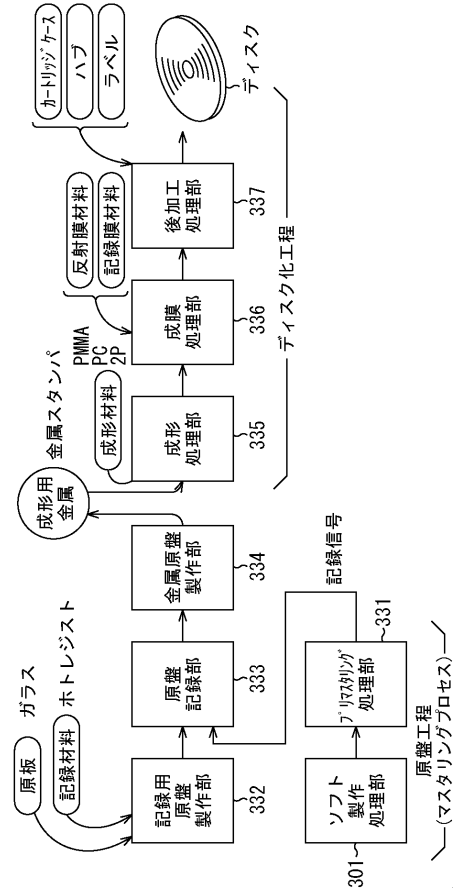
【図 4 5】
図45



【図 46】
図46



【図 47】
図47



フロントページの続き

(56)参考文献 国際公開第2010/095411 (WO, A1)
特開平11-191815 (JP, A)

(58)調査した分野(Int.Cl., DB名)
H04N 13/04
G11B 20/12
H04N 5/92