

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4816262号
(P4816262)

(45) 発行日 平成23年11月16日 (2011.11.16)

(24) 登録日 平成23年9月9日 (2011.9.9)

(51) Int.Cl.		F I	
H O 4 N	5/92	(2006.01)	H O 4 N 5/92 C
H O 4 N	5/93	(2006.01)	H O 4 N 5/93 A
G 1 1 B	20/10	(2006.01)	G 1 1 B 20/10 3 2 1 Z

請求項の数 14 (全 47 頁)

(21) 出願番号	特願2006-156905 (P2006-156905)	(73) 特許権者	000002185
(22) 出願日	平成18年6月6日 (2006.6.6)		ソニー株式会社
(65) 公開番号	特開2007-329529 (P2007-329529A)		東京都港区港南1丁目7番1号
(43) 公開日	平成19年12月20日 (2007.12.20)	(74) 代理人	100082762
審査請求日	平成21年5月25日 (2009.5.25)		弁理士 杉浦 正知
		(72) 発明者	内村 幸一
			東京都品川区北品川6丁目7番35号 ソニー株式会社内
		審査官	小田 浩

最終頁に続く

(54) 【発明の名称】 再生装置、再生方法および再生プログラム

(57) 【特許請求の範囲】

【請求項 1】

コンテンツデータと、該コンテンツデータに付随する情報を表示するための複数の第1の単位から構成される付随データとが入力される入力部と、

上記入力部から入力された上記付随データのうちデコード時刻を示す情報が定義された上記第1の単位を記憶する第1の記憶部と、

上記デコード時刻を示す情報を保持する時刻情報保持部と、

上記時刻情報保持部に保持された上記デコード時刻を示す情報に従い上記第1の記憶部から上記第1の単位を取り出してデコードするデコード部と、

上記デコード部でデコードされた上記第1の単位を記憶する第2の記憶部とを備え、

上記時刻情報保持部は、上記第1の記憶部に記憶される上記第1の単位数の上限に基づく容量を少なくとも有し、

上記第2の記憶部は、

上記付随データにより表示される画像データを形成するための情報を保持する第3の記憶部と、

上記付随データにより表示される画像データの属性情報を保持する第4の記憶部とからなり、

上記第1の記憶部に格納される付随データ数の上限は、上記第3の記憶部に記憶される上記画像データを形成するための情報数の上限と、上記第4の記憶部に記憶される上記画

10

20

像データの属性情報数の上限とに基づき決定される再生装置。

【請求項 2】

請求項 1 に記載の再生装置において、
上記付随データは、

ユーザに対して操作を促す操作画面に用いるボタンを表示するための、アニメーション表示が可能なボタン画像と、該ボタン画像に対する表示制御情報とからなることを特徴とする再生装置。

【請求項 3】

請求項 1 に記載の再生装置において、
上記第 1 の単位は、上記付随データの構造を示す情報と、該付随データの属性を示す情報と、該付随データの画像情報と、区切りを示す情報とが定義され、

1 の上記構造を示す情報と 1 の上記区切りを示す情報とを含む複数の上記第 1 の単位から第 2 の単位が構成され、

1 または複数の上記付随データの属性を示す情報を含む 1 または複数の上記第 2 の単位から第 3 の単位が構成され、

上記付随データは、上記第 3 の単位の集合体である再生装置。

【請求項 4】

請求項 3 に記載の再生装置において、
上記デコード時刻を示す情報は、上記第 1 の単位に対して定義され、上記第 1 の単位をパケット化した際のパケットヘッダに格納される再生装置。

【請求項 5】

請求項 3 に記載の再生装置において、
上記デコード時刻を示す情報は、上記構造を示す情報および上記画像情報が格納される上記第 1 の単位に対して定義される再生装置。

【請求項 6】

請求項 5 に記載の再生装置において、
上記デコード時刻を示す情報は、さらに、上記属性を示す情報が格納される上記第 1 の単位に対して定義される再生装置。

【請求項 7】

請求項 3 に記載の再生装置において、
上記第 1 の記憶部に記憶される第 1 の単位数の上限は、上記第 3 の単位に格納可能とされた、上記構造を示す情報、上記属性を示す情報および上記画像情報が格納される上記第 1 の単位の数に基づく

再生装置。

【請求項 8】

請求項 3 に記載の再生装置において、
上記第 2 の記憶部は、上記第 3 の単位毎にクリアされる再生装置。

【請求項 9】

請求項 3 に記載の再生装置において、
上記付随データは、出力時刻がさらに定義され、上記時刻情報保持部は、該付随データから取り出された該出力時刻を示す情報をさらに保持し、

上記デコード部でデコードされ上記第 2 の記憶部に記憶された上記第 1 の単位を、上記時刻情報保持部に保持された上記出力時刻に従い該第 2 の記憶部から出力するようにされ

10

20

30

40

50

、
上記時刻情報保持部は、該時刻情報保持部に格納されている上記出力時刻を示す情報に基づき、上記第2の記憶部に記憶されている上記第1の単位による表示がなされた後、該第1の単位毎にクリアされる再生装置。

【請求項10】

請求項9に記載の再生装置において、
上記時刻情報保持部は、
上記デコード時刻を示す情報と上記第1の単位とを関連付ける識別情報がさらに保持され、

10

上記識別情報は、上記時刻情報保持部に格納されている上記出力時刻を示す情報に基づき、上記第2の記憶部に記憶されている上記第1の単位による表示がなされた後、該第1の単位毎にクリアされる再生装置。

【請求項11】

請求項3に記載の再生装置において、
上記時刻情報保持部に保持されている、上記第3の単位に含まれる上記第1の単位に定義される上記デコード時刻を示す情報を、該第3の単位毎にクリアする再生装置。

【請求項12】

20

請求項11に記載の再生装置において、
上記時刻情報保持部は、
上記デコード時刻を示す情報と上記第1の単位とを対応付ける識別情報がさらに保持され、

上記デコード時刻を示す情報が上記クリアされる際に、該デコード時刻を示す情報に対応する上記識別情報も共にクリアされる再生装置。

【請求項13】

コンテンツデータと、該コンテンツデータに付随する情報を表示するための複数の第1の単位から構成される付随データとが入力される入力ステップと、

30

上記入力ステップにより入力された上記付随データのうちデコード時刻を示す情報が定義された上記第1の単位を第1の記憶部に記憶するステップと、

上記デコード時刻を示す情報を時刻情報保持部に保持するステップと、

上記時刻情報保持部に保持された上記デコード時刻を示す情報に従い上記第1の記憶部から上記第1の単位を取り出してデコードするデコードのステップと、

上記デコードのステップによりデコードされた上記第1の単位を第2の記憶部に記憶するステップとを備え、

上記時刻情報保持部は、上記第2の記憶部に記憶される上記第1の単位数の上限に基づく上記デコード時刻を示す情報を少なくとも保持可能であり、

40

上記第2の記憶部は、

上記付随データにより表示される画像データを形成するための情報を保持する第3の記憶部と、

上記付随データにより表示される画像データの属性情報を保持する第4の記憶部と
からなり、

上記第1の記憶部に格納される付随データ数の上限は、上記第3の記憶部に記憶される上記画像データを形成するための情報数の上限と、上記第4の記憶部に記憶される上記画像データの属性情報数の上限とに基づき決定される

再生方法。

【請求項14】

50

コンテンツデータと、該コンテンツデータに付随する情報を表示するための複数の第 1 の単位から構成される付随データとが入力される入力ステップと、

上記入力ステップにより入力された上記付随データのうちデコード時刻を示す情報が定義された上記第 1 の単位を第 1 の記憶部に記憶するステップと、

上記デコード時刻を示す情報を時刻情報保持部に保持するステップと、

上記時刻情報保持部に保持された上記デコード時刻を示す情報に従い上記第 1 の記憶部から上記第 1 の単位を取り出してデコードするデコードのステップと、

上記デコードのステップによりデコードされた上記第 1 の単位を第 2 の記憶部に記憶するステップと

を備え、

上記時刻情報保持部は、上記第 2 の記憶部に記憶される上記第 1 の単位数の上限に基づく上記デコード時刻を示す情報を少なくとも保持可能であり、

上記第 2 の記憶部は、

上記付随データにより表示される画像データを形成するための情報を保持する第 3 の記憶部と、

上記付随データにより表示される画像データの属性情報を保持する第 4 の記憶部とからなり、

上記第 1 の記憶部に格納される付随データ数の上限は、上記第 3 の記憶部に記憶される上記画像データを形成するための情報数の上限と、上記第 4 の記憶部に記憶される上記画像データの属性情報数の上限とに基づき決定される

再生方法をコンピュータ装置に実行させる

再生プログラム。

【発明の詳細な説明】

【技術分野】

【0001】

この発明は、ブルーレイディスク(Blu-ray Disc:登録商標)といった大容量の記録媒体に記録された、ビデオデータおよびオーディオデータに対して字幕データおよびボタンデータが多重化されたストリームを再生する再生装置、再生方法および再生プログラムに関する。

【背景技術】

【0002】

近年、記録可能で記録再生装置から取り外し可能なディスク型記録媒体の規格として、Blu-ray Disc(ブルーレイディスク)規格が提案されている。Blu-ray Disc規格では、記録媒体として直径12cm、カバー層0.1mmのディスクを用い、光学系として波長405nmの青紫色レーザ、開口数0.85の対物レンズを用いて、最大で27GB(ギガバイト)の記録容量を実現している。これにより、日本のBSデジタルハイビジョン放送を、画質を劣化させることなく2時間以上記録することが可能である。

【0003】

この記録可能光ディスクに記録するAV(Audio/Video)信号のソース(供給源)としては、従来からの、例えばアナログテレビジョン放送によるアナログ信号によるものと、例えばBSデジタル放送をはじめとするデジタルテレビジョン放送によるデジタル信号によるものとが想定されている。Blu-ray Disc規格では、これらの放送によるAV信号を記録する方法を定めた規格は、既に作られている。

【0004】

一方で、現状のBlu-ray Discの派生規格として、映画や音楽などが予め記録された、再生専用の記録媒体の開発が進んでいる。映画や音楽を記録するためのディスク状記録媒体としては、既にDVD(Digital Versatile Disc)が広く普及しているが、このBlu-ray Discの規格に基づいた再生専用光ディスクは、Blu-ray Discの大容量および高速な転送速度などを活かし、ハイビジョン映像を高画質なまま

10

20

30

40

50

で2時間以上収録できる点が、既存のDVDとは大きく異なり、優位である。

【0005】

このBlu-ray Discにおける再生専用の記録媒体の規格を、以下では、BD-ROM規格(Blu-ray Disc Read-Only Format)と呼ぶ。

【0006】

再生専用の記録媒体に記録するコンテンツに不可欠の要素として、字幕がある。BD-ROM規格においては、字幕は、動画像が表示されるプレーンとは別のプレーンに表示され、字幕が表示されるプレーンと動画像が表示されるプレーンとを合成することで、字幕と動画像とが重ね合わされて表示される。例えば、字幕を表示する字幕プレーンを動画像を表示する動画像プレーンの前面に配置し、字幕プレーンにおいて字幕以外の部分に透明の属性を持たせることで、動画像上に字幕が表示された1枚の画像データが構成できる。

10

【0007】

また、ディスクに映画などのコンテンツを記録して、パッケージメディアとして販売を行う場合、コンテンツに伴う様々なプログラムの実行などを制御するためのユーザインターフェイスをコンテンツと共にディスクに記録することが行われる。代表的なユーザインターフェイスとしては、メニュー表示が挙げられる。一例として、メニュー表示においては、機能を選択するためのボタンがボタン画像として用意され、所定の入力手段によりボタンを選択し決定することで、当該ボタンに割り当てられた機能が実行されるようになっている。一般的には、プレーヤに対応したリモートコントロールコマンドの十字キーなどを用いて、画面に表示されたボタンを選択し、決定キーを押下することで、当該ボタンに割り当てられた機能が実行される。

20

【0008】

特許文献1には、記録再生規格であるBlu-ray Disc規格(Blu-ray Disc Rewritable Format Ver1.0)を元に、動画像および字幕を表示するプレーンをそれぞれ設け、ビデオデータによる動画像と字幕画像データによる字幕とを1画面上に表示する技術や、ボタン画像を用いたメニュー表示を実現するための技術が記載されている。

【特許文献1】特開2004-304767号公報

【0009】

ところで、BD-ROM規格では、ビデオデータやオーディオデータは、MPEG2(Moving Pictures Experts Group 2)システムズに規定される、トランスポートストリーム(TS)形式でディスクに記録される。すなわち、ビデオデータやオーディオデータは、分割されてPES(Packetized Elementary Stream)パケットに詰め込まれ、さらに所定サイズに分割されデータ種類を示す識別情報(PID: Packet Identification)を有するヘッダをそれぞれ付加されて、188バイトのTSパケットとされて多重化され、ディスクに記録される。

30

【0010】

また、上述の字幕画像データや、メニュー画面を表示するための画像データやコマンドも、所定の単位でPESパケットに詰め込まれ、PESパケットが分割され所定のPIDを有するヘッダ情報を付加されてTSパケットとされ、ビデオデータやオーディオデータに対してさらに多重化され、ディスクに記録される。

40

【0011】

字幕画像データの場合を例として、より詳細に説明する。字幕画像データは、例えばオーディオ側が意図する字幕の区切りに基づく単位で構成され、当該単位毎に所定にPESパケットに格納される。このとき、字幕画像データは、例えばビットマップデータからなる字幕画像データの本体、字幕画像データの表示位置や表示方法を示す情報、字幕画像データが表示可能な表示枠を示す情報など、複数のデータから構成され、これら複数のデータのそれぞれが個別のPESパケットに格納される。以下では、字幕画像データを構成する複数のデータを総称して、字幕画像データと呼ぶ。

【0012】

PESパケットに詰め込まれた字幕画像データのデコード時刻を示すDTS(Decoding

50

Time Stamp)と、表示時刻を示すPTS(Presentation Time Stamp)とが、PESパケットのヘッダであるPESヘッダに格納される。PESパケットは、さらに所定サイズに分割され、データ種類を識別するためのPIDを所定に付加され、データサイズが188バイトのTSパケットとされる。このTSパケットがビデオデータやオーディオデータのTSパケット、ならびに、後述のボタン画像データのTSパケットと多重化されて、ディスクに記録される。

【0013】

なお、メニュー画面を表示するためのデータも、字幕画像データと略同様にして、メニュー画面を表示するための複数のデータがそれぞれPESパケット化され、PESパケットが所定サイズに分割されてPIDを付されてTSパケットとされる。

10

【0014】

ディスクを再生して上述した字幕画像データを表示させる場合、ディスクから読み出されたトランスポートストリームに対して、TSパケット毎にヘッダを抽出してPIDを取得し、PIDに基づき字幕画像データが格納されたTSパケットを振り分けてバッファに一旦格納する。バッファに対してPESパケットを形成できるだけのTSパケットが溜め込まれたら、PIDに基づき同種のTSパケットのペイロードからデータが取り出されPESパケットが再構築される。再構築されたPESパケットは、PESヘッダを取り除かれ、さらに字幕画像データを構成する複数のデータが再構築される。

【0015】

再構築された字幕画像データを構成する複数のデータは、デコードバッファに一旦格納される。そして、PESヘッダに格納されるDTSの時刻にデコードバッファから読み出されてデコードされ、出力側のバッファに格納される。出力側のバッファに格納されたデータは、PESヘッダに格納されるPTSの時刻に出力される。

20

【発明の開示】

【発明が解決しようとする課題】

【0016】

このように、字幕画像データのデコードタイミングや出力タイミングは、上述したように、PESヘッダに格納されるDTSおよびPTSによって指示される。したがって、デコードバッファに溜め込まれている字幕画像データに対応するDTSおよびPTSを、例えばデコードされたデータが出力側のバッファから出力されるまで、メモリに保持しておく必要がある。

30

【0017】

ところが、既に説明したように、字幕画像データは、上述のようにオーサ側の意図する字幕の区切りに基づき構成されるため、字幕画像データ毎にサイズが区々となる。そのため、字幕画像データを構成する複数のデータがデコードバッファに何個格納されるかを見積もるのが難しく、デコードバッファの容量に対する、DTSおよびPTSを保持するためのメモリの容量を見積もることが極めて困難であるという問題点があった。この問題は、メニュー画面を表示するためのデータにおいても、同様である。

【0018】

この問題を回避するために、このため、DTSおよびPTSを保持するためのメモリの容量を十分に大きく設計する方法が考えられる。しかしながら、この方法では、装置のコストが嵩んでしまうことになるという問題点があった。

40

【0019】

このように、従来では、デコード時刻および出力時刻が構成データ毎にそれぞれ指定された、字幕画像データやメニュー画面を表示するためのデータをデコードするデコーダを適切に設計および構成することが困難であるという問題点があった。

【0020】

したがって、この発明の目的は、デコード時刻および出力時刻が構成データ毎にそれぞれ指定された、字幕画像データやメニュー画面を表示するためのデータをデコードするデコーダが適切に構成された再生装置、再生方法および再生プログラムを提供することにあ

50

る。

【課題を解決するための手段】

【0021】

この発明は、上述した課題を解決するために、コンテンツデータと、コンテンツデータに付随する情報を表示するための複数の第1の単位から構成される付随データとが入力される入力部と、入力部から入力された付随データのうちデコード時刻を示す情報が定義された第1の単位を記憶する第1の記憶部と、デコード時刻を示す情報を保持する時刻情報保持部と、時刻情報保持部に保持されたデコード時刻を示す情報に従い第1の記憶部から第1の単位を取り出してデコードするデコード部と、デコード部でデコードされた第1の単位を記憶する第2の記憶部とを備え、時刻情報保持部は、第1の記憶部に記憶される上記第1の単位数の上限に基づく容量を少なくとも有し、第2の記憶部は、付随データにより表示される画像データを形成するための情報を保持する第3の記憶部と、付随データにより表示される画像データの属性情報を保持する第4の記憶部とからなり、第1の記憶部に格納される付随データ数の上限は、第3の記憶部に記憶される画像データを形成するための情報数の上限と、第4の記憶部に記憶される画像データの属性情報数の上限とに基づき決定される再生装置である。

10

【0022】

また、この発明は、コンテンツデータと、コンテンツデータに付随する情報を表示するための複数の第1の単位から構成される付随データとが入力される入力部と、入力部により入力された付随データのうちデコード時刻を示す情報が定義された第1の単位を第1の記憶部に記憶するステップと、デコード時刻を示す情報を時刻情報保持部に保持するステップと、時刻情報保持部に保持されたデコード時刻を示す情報に従い第1の記憶部から第1の単位を取り出してデコードするデコードのステップと、デコードのステップによりデコードされた第1の単位を第2の記憶部に記憶するステップとを備え、時刻情報保持部は、第2の記憶部に記憶される第1の単位数の上限に基づくデコード時刻を示す情報を少なくとも保持可能であり、第2の記憶部は、付随データにより表示される画像データを形成するための情報を保持する第3の記憶部と、付随データにより表示される画像データの属性情報を保持する第4の記憶部とからなり、第1の記憶部に格納される付随データ数の上限は、第3の記憶部に記憶される画像データを形成するための情報数の上限と、第4の記憶部に記憶される画像データの属性情報数の上限とに基づき決定される再生方法である。

20

30

【0023】

また、この発明は、コンテンツデータと、コンテンツデータに付随する情報を表示するための複数の第1の単位から構成される付随データとが入力される入力部と、入力部により入力された付随データのうちデコード時刻を示す情報が定義された第1の単位を第1の記憶部に記憶するステップと、デコード時刻を示す情報を時刻情報保持部に保持するステップと、時刻情報保持部に保持されたデコード時刻を示す情報に従い第1の記憶部から第1の単位を取り出してデコードするデコードのステップと、デコードのステップによりデコードされた第1の単位を第2の記憶部に記憶するステップとを備え、時刻情報保持部は、第2の記憶部に記憶される第1の単位数の上限に基づくデコード時刻を示す情報を少なくとも保持可能であり、第2の記憶部は、付随データにより表示される画像データを形成するための情報を保持する第3の記憶部と、付随データにより表示される画像データの属性情報を保持する第4の記憶部とからなり、第1の記憶部に格納される付随データ数の上限は、第3の記憶部に記憶される画像データを形成するための情報数の上限と、第4の記憶部に記憶される画像データの属性情報数の上限とに基づき決定される再生方法をコンピュータ装置に実行させる再生プログラムである。

40

【0024】

上述したように、この発明は、コンテンツデータと、コンテンツデータに付随する情報を表示するための複数の第1の単位から構成される付随データとが入力され、入力された

50

付随データのうちデコード時刻を示す情報が定義された第1の単位を第1の記憶部に記憶すると共に、デコード時刻を示す情報を時刻情報保持部に保持し、時刻情報保持部に保持されたデコード時刻を示す情報に従い第1の記憶部から第1の単位を取り出してデコードし、デコードされた第1の単位を第2の記憶部に記憶するようにされ、時刻情報保持部は、第1の記憶部に記憶される第1の単位数の上限に基づく容量を少なくとも有するようにしているため、時刻情報保持部の容量を容易に見積もることが可能で、デコーダを適切に構成することができる。

【発明の効果】

【0025】

この発明は、上述したように、コンテンツデータと、コンテンツデータに付随する情報を表示するための複数の第1の単位から構成される付随データとが入力され、入力された付随データのうちデコード時刻を示す情報が定義された第1の単位を第1の記憶部に記憶すると共に、デコード時刻を示す情報を時刻情報保持部に保持し、時刻情報保持部に保持されたデコード時刻を示す情報に従い第1の記憶部から第1の単位を取り出してデコードし、デコードされた第1の単位を第2の記憶部に記憶するようにされ、時刻情報保持部は、第1の記憶部に記憶される第1の単位数の上限に基づく容量を少なくとも有するようにしているため、時刻情報保持部の容量を容易に見積もることが可能で、デコーダを適切に構成することができる効果がある。

【発明を実施するための最良の形態】

【0026】

以下、この発明の実施の形態を、図面を参照しながら説明する。先ず、理解を容易とするために、Blu-ray Discに関し、"Blu-ray Disc Read-Only Format Ver1.0 part3 Audio Visual Specifications"で規定されている、読み出し専用タイプのBlu-ray DiscであるBD-ROMに記録されたコンテンツすなわちAV(Audio/Video)データの管理構造について説明する。以下では、このBD-ROMにおける管理構造をBDMVフォーマットと称する。

【0027】

例えばMPEG(Moving Pictures Experts Group)ビデオやMPEGオーディオなどの符号化方式で符号化され、MPEG2システムに従い多重化されたビットストリームは、クリップAVストリーム(またはAVストリーム)と称される。クリップAVストリームは、Blu-ray Discに関する規格の一つである"Blu-ray Disc Read-Only Format part2"で定義されたファイルシステムにより、ファイルとしてディスクに記録される。このファイルを、クリップAVストリームファイル(またはAVストリームファイル)と称する。

【0028】

クリップAVストリームファイルは、ファイルシステム上での管理単位であり、ユーザにとって必ずしも分かりやすい管理単位であるとは限らない。ユーザの利便性を考えた場合、複数のクリップAVストリームファイルに分割された映像コンテンツを一つにまとめて再生する仕組みや、クリップAVストリームファイルの一部だけを再生する仕組み、さらには、特殊再生や頭出し再生を滑らかに行うための情報などをデータベースとしてディスクに記録しておく必要がある。Blu-ray Discに関する規格の一つである"Blu-ray Disc Read-Only Format part3"で、このデータベースが規定される。

【0029】

図1は、BD-ROMのデータモデルを概略的に示す。BD-ROMのデータ構造は、図1に示されるように4層のレイヤよりなる。最も最下層のレイヤは、クリップAVストリームが配置されるレイヤである(便宜上、クリップレイヤと呼ぶ)。その上のレイヤは、クリップAVストリームに対する再生箇所を指定するための、ムービープレイリスト(Movie Playlist)と、プレイアイテム(PlayItem)とが配置されるレイヤである(便宜上、プレイリストレイヤと呼ぶ)。さらにその上のレイヤは、ムービープレイリストに対して再生順などを指定するコマンドからなるムービーオブジェクト(Movie Object)などが配置さ

10

20

30

40

50

れるレイヤである（便宜上、オブジェクトレイヤと呼ぶ）。最上層のレイヤは、このBD-ROMに格納されるタイトルなどを管理するインデックステーブルが配置される（便宜上、インデックスレイヤと呼ぶ）。

【0030】

クリップレイヤについて説明する。クリップAVストリームは、ビデオデータやオーディオデータがMPEG2 TS（トランスポートストリーム）の形式などに多重化されたビットストリームである。このクリップAVストリームに関する情報がクリップ情報(Clip Information)としてファイルに記録される。

【0031】

また、クリップAVストリームには、ビデオデータやオーディオデータによるコンテンツデータに対して付随して表示される字幕やメニュー表示を行うためのストリームも多重化される。字幕を表示するためのグラフィックスストリームは、プレゼンテーショングラフィックス(PG)ストリームと呼ばれる。また、メニュー表示などに用いられるデータをストリームにしたものは、インタラクティブグラフィックス(IG)ストリームと呼ばれる。

【0032】

クリップAVストリームファイルと、対応するクリップ情報が記録されたクリップ情報ファイルとをひとまとまりのオブジェクトと見なし、クリップ(Clip)と称する。すなわち、クリップは、クリップAVストリームとクリップ情報とから構成される、一つのオブジェクトである。

【0033】

ファイルは、一般的に、バイト列として扱われる。クリップAVストリームファイルのコンテンツは、時間軸上に展開され、クリップ中のエントリーポイントは、主に時間ベースで指定される。所定のクリップへのアクセスポイントのタイムスタンプが与えられた場合、クリップ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アドレス情報を見つけるために、クリップ情報ファイルを用いることができる。

【0034】

プレイリストレイヤについて説明する。ムービープレイリストは、再生するAVストリームファイルの指定と、指定されたAVストリームファイルの再生箇所を指定する再生開始点(IN点)と再生終了点(OUT点)の集まりとから構成される。この再生開始点と再生終了点の情報を一組としたものは、プレイアイテム(PlayItem)と称される。ムービープレイリストは、プレイアイテムの集合で構成される。プレイアイテムを再生するということは、そのプレイアイテムに参照されるAVストリームファイルの一部を再生することになる。すなわち、プレイアイテム中のIN点およびOUT点情報に基づき、クリップ中の対応する区間が再生される。

【0035】

オブジェクトレイヤについて説明する。ムービーオブジェクトは、HDMVナビゲーションコマンドプログラム(HDMVプログラム)と、ムービーオブジェクトとを連携するターミナルインフォメーションを含む。HDMVプログラムは、プレイリストの再生を制御するためのコマンドである。ターミナルインフォメーションは、ユーザのBD-ROMプレーヤに対するインタラクティブな操作を許可するための情報を含んでいる。このターミナルインフォメーションに基づき、メニュー画面の呼び出しや、タイトルサーチといったユーザオペレーションが制御される。

【0036】

BD-Jオブジェクトは、Javaプログラム(Javaは登録商標)によるオブジェクトからなる。BD-Jオブジェクトは、この発明と関わりが薄いので、詳細な説明を省略する。

【0037】

インデックスレイヤについて説明する。インデックスレイヤは、インデックステーブルからなる。インデックステーブルは、BD-ROMディスクのタイトルを定義する、トップレベルのテーブルである。インデックステーブルに格納されているタイトル情報に基づ

10

20

30

40

50

き、BD-ROM常駐システムソフトウェア中のモジュールマネージャによりBD-ROMディスクの再生が制御される。

【0038】

すなわち、図2に概略的に示されるように、インデックステーブル中の任意のエントリは、タイトルと称され、インデックステーブルにエントリされるファーストプレイバック(First Playback)、トップメニュー(Top Menu)およびタイトル(Title) # 1、# 2、・・・は、全てタイトルである。各タイトルは、ムービーオブジェクトあるいはBD-Jオブジェクトに対するリンクを示し、各タイトルは、HDMVタイトルあるいはBD-Jタイトルの何れかを示す。

【0039】

例えば、ファーストプレイバックは、当該BD-ROMに格納されるコンテンツが映画であれば、映画本編に先立って映出される映画会社の宣伝用映像(トレーラ)である。トップメニューは、例えばコンテンツが映画である場合、本編再生、チャプタサーチ、字幕や言語設定、特典映像再生などを選択するためのメニュー画面である。また、タイトルは、トップメニューから選択される各映像である。タイトルがさらにメニュー画面であるような構成も可能である。

【0040】

図3は、上述のようなクリップAVストリーム、クリップ情報(Stream Attributes)、クリップ、プレイアイテムおよびプレイリストの関係を示すUML(Unified Modeling Language)図である。プレイリストは、1または複数のプレイアイテムに対応付けられ、プレイアイテムは、1のクリップに対応付けられる。1のクリップに対して、それぞれ開始点および/または終了点が異なる複数のプレイアイテムを対応付けることができる。1のクリップから1のクリップAVストリームファイルが参照される。同様に、1のクリップから1のクリップ情報ファイルが参照される。また、クリップAVストリームファイルとクリップ情報ファイルとは、1対1の対応関係を有する。このような構造を定義することにより、クリップAVストリームファイルを変更することなく、任意の部分だけを再生する、非破壊の再生順序指定を行うことが可能となる。

【0041】

また、図4のように、複数のプレイリストから同一のクリップを参照することもできる。また、1のプレイリストから複数のクリップを指定することもできる。クリップは、プレイリスト中のプレイアイテムに示されるIN点およびOUT点により、参照される。図4の例では、クリップ300は、プレイリスト310のプレイアイテム320から参照されると共に、プレイリスト311を構成するプレイアイテム321および322のうちプレイアイテム321から、IN点およびOUT点で示される区間が参照される。また、クリップ301は、プレイリスト311のプレイアイテム322からIN点およびOUT点で示される区間が参照されると共に、プレイリスト312のプレイアイテム323および324のうち、プレイアイテム323のIN点およびOUT点で示される区間が参照される。

【0042】

なお、プレイリストは、図5に一例が示されるように、主として再生されるプレイアイテムに対応するメインパスに対して、サブプレイアイテムに対応するサブパスを持つことができる。サブプレイアイテムは、例えば、このプレイリストに付けられているアフレコオーディオ用のプレイアイテムである。詳細は省略するが、プレイリストは、所定の条件を満たす場合にだけ、サブプレイアイテムを持つことができる。

【0043】

次に、"Blu-ray Disc Read-Only Format part3"で規定された、BD-ROMに記録されるファイルの管理構造について、図6を用いて説明する。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。記録媒体上には、先ず、1つのディレクトリ(図6の例ではルート(root)ディレクトリ)が作成される。この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する。

10

20

30

40

50

【 0 0 4 4 】

ルートディレクトリの下に、ディレクトリ"BDMV"およびディレクトリ"CERTIFICATE"が置かれる。ディレクトリ"CERTIFICATE"は、著作権に関する情報が格納される。ディレクトリ"BDMV"に、図 1 を用いて説明したデータ構造が格納される。

【 0 0 4 5 】

ディレクトリ"BDMV"の直下には、ファイルは、ファイル"index.bdmv"およびファイル"MovieObject.bdmv"の 2 つのみを置くことができる。また、ディレクトリ"BDMV"の下に、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"AUXDATA"、ディレクトリ"META"、ディレクトリ"BDJO"、ディレクトリ"JAR"、およびディレクトリ"BACKUP"が置かれる。

10

【 0 0 4 6 】

ファイル"index.bdmv"は、ディレクトリBDMVの内容について記述される。すなわち、このファイル"index.bdmv"が上述した最上層のレイヤであるインデックスレイヤにおけるインデックステーブルに対応する。また、ファイルMovieObject.bdmvは、1 つ以上のムービーオブジェクトの情報が格納される。すなわち、このファイル"MovieObject.bdmv"が上述したオブジェクトレイヤに対応する。

【 0 0 4 7 】

ディレクトリ"PLAYLIST"は、プレイリストのデータベースが置かれるディレクトリである。すなわち、ディレクトリ"PLAYLIST"は、ムービープレイリストに関するファイルであるファイル"xxxxxx.mpls"を含む。ファイル"xxxxxx.mpls"は、ムービープレイリストのそれぞれに対して作成されるファイルである。ファイル名において、"."（ピリオド）の前の"xxxxxx"は、5 桁の数字とされ、ピリオドの後ろの"mpls"は、このタイプのファイルに固定的とされた拡張子である。

20

【 0 0 4 8 】

ディレクトリ"CLIPINF"は、クリップのデータベースが置かれるディレクトリである。すなわち、ディレクトリCLIPINFは、クリップA Vストリームファイルのそれぞれに対するクリップインフォメーションファイルであるファイル"zzzzz.clpi"を含む。ファイル名において、"."（ピリオド）の前の"zzzzz"は、5 桁の数字とされ、ピリオドの後ろの"clpi"は、このタイプのファイルに固定的とされた拡張子である。

【 0 0 4 9 】

ディレクトリ"STREAM"は、実体としてのA Vストリームファイルが置かれるディレクトリである。すなわち、ディレクトリ"STREAM"は、クリップインフォメーションファイルのそれぞれに対応するクリップA Vストリームファイルを含む。クリップA Vストリームファイルは、MPEG 2 (Moving Pictures Experts Group 2)のトランスポートストリーム（以下、MPEG 2 TSと略称する）からなり、ファイル名が"zzzzz.m2ts"とされる。ファイル名において、ピリオドの前の"zzzzz"は、対応するクリップインフォメーションファイルと同一することで、クリップインフォメーションファイルとこのクリップA Vストリームファイルとの対応関係を容易に把握することができる。

30

【 0 0 5 0 】

ディレクトリ"AUXDATA"は、メニュー表示などに用いられる、サウンドファイル、フォントファイル、フォントインデックスファイルおよびビットマップファイルなどが置かれる。ファイル"sound.bdmv"は、HDMVのインタラクティブなグラフィクスストリームのアプリケーションに関連したサウンドデータが格納される。ファイル名は、"sound.bdmv"に固定的とされる。ファイル"aaaaa.otf"は、字幕表示や上述したBD-Jアプリケーションなどで用いられるフォントデータが格納される。ファイル名において、ピリオドの前の"aaaaa"は、5 桁の数字とされ、ピリオドの後ろの"otf"は、このタイプのファイルに固定的とされた拡張子である。ファイル"bdmv.fontindex"は、フォントのインデックスファイルである。

40

【 0 0 5 1 】

ディレクトリ"META"は、メタデータファイルが格納される。ディレクトリ"BDJO"および

50

ディレクトリ"JAR"は、上述のBD-Jオブジェクトに関連するファイルが格納される。また、ディレクトリ"BACKUP"は、上述までの各ディレクトリおよびファイルのバックアップが格納される。これらディレクトリ"META"、ディレクトリ"BDJO"、ディレクトリ"JAR"およびディレクトリ"BACKUP"は、この発明の主旨と直接的な関わりがないので、詳細な説明を省略する。

【0052】

上述したようなデータ構造を有するディスクがプレーヤに装填されると、プレーヤは、ディスクから読み出されたムービーオブジェクトなどに記述されたコマンドを、プレーヤ内部のハードウェアを制御するための固有のコマンドに変換する必要がある。プレーヤは、このような変換を行うためのソフトウェアを、プレーヤに内蔵されるROM(Read Only Memory)にあらかじめ記憶している。このソフトウェアは、ディスクとプレーヤを仲介してプレーヤにBD-ROMの規格に従った動作をさせることから、BD仮想プレーヤと称される。

【0053】

図7は、このBD仮想プレーヤの動作を概略的に示す。図7Aは、ディスクのローディング時の動作の例を示す。ディスクがプレーヤに装填されディスクに対するイニシャルアクセスがなされると(ステップS30)、1のディスクにおいて共有的に用いられる共有パラメータが記憶されるレジスタが初期化される(ステップS31)。そして、次のステップS32で、プログラムがディスクから読み込まれて実行される。なお、イニシャルアクセスは、ディスク装填時のように、ディスクの再生が初めて行われることをいう。

【0054】

図7Bは、プレーヤが停止状態からユーザにより例えばプレイキーが押下され再生が指示された場合の動作の例を示す。最初の停止状態(ステップS40)に対して、ユーザにより、例えばリモートコントロールコマンドなどを用いて再生が指示される(UO: User Operation)。再生が指示されると、まず、レジスタすなわち共通パラメータが初期化される(ステップS41)、次のステップS42で、プレイリストの再生フェイズに移行する。なお、この場合にレジスタがリセットされない実装としてもよい。

【0055】

ムービーオブジェクトの実行フェイズにおけるプレイリストの再生について、図8を用いて説明する。UOなどにより、タイトル番号#1のコンテンツを再生開始する指示があった場合について考える。プレーヤは、コンテンツの再生開始指示に応じて、上述した図2に示されるインデックステーブル(Index Table)を参照し、タイトル#1のコンテンツ再生に対応するオブジェクトの番号を取得する。例えばタイトル#1のコンテンツ再生を実現するオブジェクトの番号が#1であったとすると、プレーヤは、ムービーオブジェクト#1の実行を開始する。

【0056】

この図8の例では、ムービーオブジェクト#1に記述されたプログラムは2行からなり、1行目のコマンドが"Play Playlist(1)"であるとする、プレーヤは、プレイリスト#1の再生を開始する。プレイリスト#1は、1以上のプレイアイテムから構成され、プレイアイテムが順次再生される。プレイリスト#1中のプレイアイテムの再生が終了すると、ムービーオブジェクト#1の実行に戻り、2行目のコマンドが実行される。図8の例では、2行目のコマンドが"jump TopMenu"であって、このコマンドが実行されインデックステーブルに記述されたトップメニュー(Top Menu)を実現するムービーオブジェクトの実行が開始される。

【0057】

次に、この発明の実施の形態に適用可能な画像の表示系について説明する。この発明の実施の形態では、画像の表示系について、図9に一例が示されるようなプレーン構成を取る。動画プレーン10は、最も後ろ側(ボトム)に表示され、プレイリストで指定された画像(主に動画データ)が扱われる。字幕プレーン11は、動画プレーン10の上に表示され、動画再生中に表示される字幕データが扱われる。インタラクティブグラフィクス

レーン 12 は、最も前面に表示され、メニュー画面を表示するための文字データや、ボタン画像用のビットマップデータなどのグラフィクスデータが扱われる。1つの表示画面は、これら3つのプレーンが合成されて表示される。

【0058】

なお、以下では、グラフィクスプレーン 12 をインタラクティブグラフィクスプレーン 12 と称する。

【0059】

動画プレーン 10、字幕プレーン 11 およびインタラクティブグラフィクスプレーン 12 は、それぞれ独立して表示が可能とされ、例えば、図 10 に一例が示されるような解像度および表示可能色を有する。動画プレーン 10 は、解像度が 1920 画素 × 1080 ラインで 1 画素当たり換算したデータ長が 16 ビットであって、輝度信号 Y、色差信号 Cb、Cr が 4 : 2 : 2 のシステム（以下、YCbCr (4 : 2 : 2)）とされる。なお、YCbCr (4 : 2 : 2) は、各画素当たり輝度信号 Y が 8 ビット、色差信号 Cb、Cr がそれぞれ 8 ビットで、色差信号 Cb、Cr が水平 2 画素で一つの色データを構成すると見なすカラーシステムである。インタラクティブグラフィクスプレーン 12 および字幕プレーン 11 は、1920 画素 × 1080 ラインで各画素のサンプリング深さが 8 ビットとされ、カラーシステムは、256 色のパレットを用いた 8 ビットカラーマップアドレスとされる。

【0060】

インタラクティブグラフィクスプレーン 12 および字幕プレーン 11 は、256 段階のアルファブレンディングが可能とされており、他のプレーンとの合成の際に、不透明度を 256 段階で設定することが可能とされている。不透明度の設定は、画素毎に行うことができる。以下では、不透明度 が (0 1) の範囲で表され、不透明度 = 0 で完全に透明、不透明度 = 1 で完全に不透明であるものとする。

【0061】

字幕プレーン 11 では、例えば PNG (Portable Network Graphics) 形式の画像データが扱われる。また、インタラクティブグラフィクスプレーン 12 でも、例えば PNG 形式の画像データを扱うことができる。PNG 形式は、1 画素のサンプリング深さが 1 ビット ~ 16 ビットとされ、サンプリング深さが 8 ビットまたは 16 ビットの場合に、アルファチャンネル、すなわち、それぞれの画素成分の不透明度情報（アルファデータと称する）を付加することができる。サンプリング深さが 8 ビットの場合には、256 段階で不透明度を指定することができる。このアルファチャンネルによる不透明度情報を用いてアルファブレンディングが行われる。また、256 色までのパレットイメージを用いることができ、予め用意されたパレットの何番目の要素（インデックス）であるかがインデックス番号により表現される。

【0062】

なお、字幕プレーン 11 およびインタラクティブグラフィクスプレーン 12 で扱われる画像データは、PNG 形式に限定されない。JPEG 方式など他の圧縮符号化方式で圧縮符号化された画像データや、ランレングス圧縮された画像データ、圧縮符号化がなされていないビットマップデータなどを扱うようにしてもよい。

【0063】

図 11 は、上述の図 9 および図 10 に従い 3 つのプレーンを合成するグラフィクス処理部の一例の構成を示す。動画プレーン 10 の動画データが 422 / 444 変換回路 20 に供給される。動画データは、422 / 444 変換回路 20 でカラーシステムが YCbCr (4 : 2 : 2) から YCbCr (4 : 4 : 4) に変換され、乗算器 21 に入力される。

【0064】

字幕プレーン 11 の画像データがパレット 22 A に入力され、RGB (4 : 4 : 4) の画像データとして出力される。この画像データに対してアルファブレンディングによる不透明度が指定されている場合には、指定された不透明度 1 (0 1 1) がパレット 22 A から出力される。

10

20

30

40

50

【 0 0 6 5 】

図 1 2 は、パレット 2 2 A の入出力データの一例を示す。パレット 2 2 A は、例えば P N G 形式のファイルに対応したパレット情報がテーブルとして格納される。パレット 2 2 A は、入力された 8 ビットの画素データをアドレスとして、インデックス番号が参照される。このインデックス番号に基づき、それぞれ 8 ビットのデータからなる R G B (4 : 4 : 4) のデータが出力される。それと共に、パレット 2 2 A では、不透明度を表すアルファチャンネルのデータ が取り出される。

【 0 0 6 6 】

図 1 3 は、パレット 2 2 A に格納される一例のパレットテーブルを示す。2 5 6 個のカラーインデックス値 [0 x 0 0] ~ [0 x F F] ([0 x] は 1 6 進表記であることを示す) のそれぞれに対して、各々 8 ビットで表現される三原色の値 R、G および B と、不透明度 とが割り当てられる。パレット 2 2 A は、入力された P N G 形式の画像データに基づきパレットテーブルが参照され、画像データにより指定されたインデックス値に対応する、それぞれ 8 ビットのデータからなる R、G および B 各色のデータ (R G B データ) と、不透明度 とを画素毎に出力する。

【 0 0 6 7 】

パレット 2 2 A から出力された R G B データは、R G B / Y C b C r 変換回路 2 2 B に供給され、各データ長が 8 ビットの輝度信号 Y と色信号 C b、C r のデータに変換される (以下、まとめて Y C b C r データと称する)。これは、以降のプレーン間合成を共通のデータ形式で行う必要があるため、動画データのデータ形式である Y C b C r データに統一している。

【 0 0 6 8 】

R G B / Y C b C r 変換回路 2 2 B から出力された Y C b C r データと不透明度データ 1 とがそれぞれ乗算器 2 3 に入力される。乗算器 2 3 では、入力された Y C b C r データに不透明度データ 1 が乗ぜられる。乗算結果は、加算器 2 4 の一方の入力端に入力される。なお、乗算器 2 3 では、Y C b C r データにおける輝度信号 Y、色差信号 C b、C r のそれぞれについて、不透明度データ 1 との乗算が行われる。また、不透明度データ 1 の補数 (1 - 1) が乗算器 2 1 に供給される。

【 0 0 6 9 】

乗算器 2 1 では、4 2 2 / 4 4 4 変換回路 2 0 から入力された動画データに不透明度データ 1 の補数 (1 - 1) が乗ぜられる。乗算結果は、加算器 2 4 の他方の入力端に入力される。加算器 2 4 において、乗算器 2 1 および 2 3 の乗算結果が加算される。これにより、動画プレーン 1 0 と字幕プレーン 1 1 とが合成される。加算器 2 4 の加算結果が乗算器 2 5 に入力される。

【 0 0 7 0 】

インタラクティブグラフィクスプレーン 1 2 の画像データがパレット 2 6 A に入力され、R G B (4 : 4 : 4) の画像データとして出力される。この画像データに対してアルファブレンディングによる不透明度が指定されている場合には、指定された不透明度 2 (0 2 1) がパレット 2 6 A から出力される。パレット 2 6 A から出力された R G B データは、R G B / Y C b C r 変換回路 2 6 B に供給されて Y C b C r データに変換され、動画データのデータ形式である Y C b C r データに統一される。R G B / Y C b C r 変換回路 2 6 B から出力された Y C b C r データが乗算器 2 7 に入力される。

【 0 0 7 1 】

インタラクティブグラフィクスプレーン 1 2 ' で用いられる画像データが P N G 形式である場合には、画像データ中に、画素毎に不透明度データ 2 (0 2 1) を設定することができる。不透明度データ 2 は、乗算器 2 7 に供給される。乗算器 2 7 では、R G B / Y C b C r 変換回路 2 6 から入力された Y C b C r データに対し、輝度信号 Y、色差信号 C b、C r のそれぞれについて、不透明度データ 2 との乗算が行われる。乗算器 2 7 による乗算結果が加算器 2 8 の一方の入力端に入力される。また、不透明度データ 2 の補数 (1 - 2) が乗算器 2 5 に供給される。

10

20

30

40

50

【 0 0 7 2 】

乗算器 2 5 では、加算器 2 4 の加算結果に対して不透明度データ 2 の補数 (1 - 2) が乗ぜられる。乗算器 2 5 の乗算結果は、加算器 2 8 の他方の入力端に入力され、上述した乗算器 2 7 による乗算結果と加算される。これにより、動画プレーン 1 0 と字幕プレーン 1 1 との合成結果に対して、さらに、インタラクティブグラフィクスプレーン 1 2 が合成される。

【 0 0 7 3 】

字幕プレーン 1 1 およびインタラクティブグラフィクスプレーン 1 2 において、例えば、表示すべき画像の無い領域の不透明度 = 0 と設定することで、そのプレーンの下に表示されるプレーンを透過表示させることができ、例えば動画プレーン 1 0 に表示されている動画データを、字幕プレーン 1 1 やインタラクティブグラフィクスプレーン 1 2 の背景として表示することができる。

10

【 0 0 7 4 】

なお、この図 1 1 に示される構成は、ハードウェアおよびソフトウェアの何れでも実現可能なものである。

【 0 0 7 5 】

次に、図 1 4 を用いて、プレゼンテーショングラフィクス (P G) ストリームおよびインタラクティブグラフィクス (I G) ストリームについて説明する。上述したように、P G ストリームは、字幕を表示させるための字幕画像データによるストリームであって、より具体的には、例えばビットマップデータからなる字幕画像データそのものや、字幕画像データの表示方法を示すデータなどからなる。また、I G ストリームは、メニュー表示などに用いられるデータによるストリームであって、より具体的には、メニュー画面に用いられるボタン画像データ、ボタン画像データの表示方法を示すデータ、ボタンによって指示されるコマンドなどからなる。

20

【 0 0 7 6 】

P G ストリームおよび I G ストリームは、クリップ A V ストリームに含まれ、1 または複数のディスプレイセット (DisplaySet) からなるエポック (epoch) を単位として構成される。ディスプレイセットおよびエポックは、M P E G 2 (Moving Pictures Experts Group 2) で定義されるピクチャおよび G O P (Group Of Picture) にそれぞれ例えることができる。

30

【 0 0 7 7 】

すなわち、ディスプレイセットは、字幕表示上の最小単位であって、1 枚の字幕画像に対応する。1 または複数のディスプレイセットによりエポックが構成される。エポックは、一般的には、プレイアイテムに対応した単位で構成される。すなわち、あるプレイアイテムの再生中に表示される字幕を表示するためのディスプレイセットによりエポックが構成される。P G ストリームをデコードする P G デコーダは、字幕プレーン 1 1 および各種のバッファをエポックの単位でクリアする。

【 0 0 7 8 】

P G ストリームの場合、図 1 4 に例示されるように、1 のエポックに複数のディスプレイセットを含めることができる。一方、I G ストリームは、1 のエポックに含めることができるディスプレイセットは、例えば 1 個だけとされる。

40

【 0 0 7 9 】

実際には、エポックは、プレイアイテム単位に限らず、オーサリング側で任意に構成することができる。例えば、P G ストリームの場合、映画などにおいて一連の台詞に対応する複数の字幕のそれぞれを表示する複数のディスプレイセットによりエポックが構成される。また例えば、1 つの台詞を文字毎に分解し、分解された文字それぞれを字幕として表示する複数のディスプレイセットによりエポックを構成することもできる。

【 0 0 8 0 】

図 1 5 は、ディスプレイセットの構成を概略的に示す。プレゼンテーショングラフィクスのディスプレイセットは、図 1 5 A に示されるように、P C S (Presentation Composit

50

ion Segment)、W D S (Window Definision Segment)、P D S (Palette Definition Segment)、O D S (Object Definition Segment)およびE N Dセグメントの5種類のセグメントから構成される。詳細は後述するが、P C Sは、字幕の基本情報が格納される。W D Sは、ウィンドウと呼ばれる字幕を表示する表示枠を定義する。W D Sは、エポック内で変化しない。P D Sは、字幕の色情報を定義するもので、例えばインデックス化された色情報テーブルを内包する。また、O D Sは、字幕の形状情報を保持するもので、例えば字幕を表示するためのビットマップデータが、ランレングス圧縮など所定の圧縮符号化方式で圧縮符号化されて格納される。また、E N Dセグメントは、このディスプレイセットの終わりを示すセグメントである。

【0081】

インタラクティブグラフィックスのディスプレイセットは、図15Bに示されるように、I C S (Interactive Composition Segment)、P D S、O D SおよびE N Dセグメントの4種類のセグメントから構成される。4種類のセグメントのうち、I C Sは、インタラクティブグラフィックスの基本構造を保持するためのセグメントである。P D S、O D SおよびE N Dセグメントは、プレゼンテーショングラフィックスの場合と同様である。すなわち、P D Sは、ボタン画像の色情報を保持するためのセグメントである。O D Sは、ボタン画像の形状を保持するための情報である。より具体的には、O D Sは、ボタン画像そのもの、例えばボタン画像を表示するためのビットマップデータが、ランレングス圧縮など所定の圧縮符号化方式で圧縮符号化されて格納される。また、E N Dセグメントは、このディスプレイセットの終わりを示すセグメントである。

【0082】

図16は、プレゼンテーショングラフィックスのディスプレイセットに関する論理的な構造を概略的に示す。図16Aおよび図16Bに例示されるように、P C S、W D S、P D S、O D SおよびE N Dセグメント(図示しない)の4種類のセグメントが集まって、1つのディスプレイセットを構成する。P C Sは、ディスプレイセット中に必ず1だけ存在する。W D Sは、殆どの場合、ディスプレイセット中に1つだけ存在する。但し、後述するP C S中に定義されるフラグpalette_up_date_flagの値が"1"の場合は、W D Sは、オプションである。P D SおよびO D Sは、オプションであって、ディスプレイセット中に存在しない場合もあり得る。

【0083】

図16Cに例示されるように、1または複数のディスプレイセットから、エポックが構成される。ここで、詳細は後述するが、P C Sのブロックcomposition_descriptor()に含まれるフィールドcomposition_stateには、エポックスタートディスプレイセット(Epoch Start DisplaySet)を定義することができる。このエポックスタートディスプレイセットに基づき、エポックが定義される。一例として、エポックは、エポックスタートディスプレイセットがP C Sに定義されるディスプレイセットから、次のエポックスタートディスプレイセットがP C Sに定義されるディスプレイセットの1つ前のディスプレイセットまでのディスプレイセットが集まったものと定義される。エポックの定義の具体例については、後述する。図16Dに例示されるように、プレゼンテーショングラフィックスストリーム(P Gストリーム)は、1または複数のエポックが含まれる。

【0084】

図17は、インタラクティブグラフィックスのディスプレイセットに関する論理的な構造を概略的に示す。図17Aに示されるように、インタラクティブグラフィックスのディスプレイセットは、プレゼンテーショングラフィックスのディスプレイセットに対してセグメントの構成が異なる以外は、全体的な階層構造は、プレゼンテーショングラフィックスのディスプレイセットと同一である。インタラクティブグラフィックスのディスプレイセットにおいても、I C Sのブロックcomposition_descriptor()に含まれるフィールドcomposition_stateにより定義されるエポックスタートディスプレイセットに基づき、エポックが定義される。なお、インタラクティブグラフィックスにおいても、P D SおよびO D Sは、オプションであって、ディスプレイセット中に存在しない場合があり得る。インタラクティブ

10

20

30

40

50

グラフィックスのディスプレイセットについては、繁雑さを避けるために、詳細な説明を省略する。

【 0 0 8 5 】

図 1 8 は、上述したようなディスプレイセットの一例の格納形式を示す。既に説明したように、字幕を表示するためのディスプレイセットは、プレゼンテーショングラフィックスストリームとして、ボタン画像を表示するためのディスプレイセットは、インタラクティブグラフィックスストリームとして、それぞれクリップ A V ストリームに多重化される。ディスプレイセットは、上述したように、複数種類のセグメントから構成される (図 1 8 A)。

【 0 0 8 6 】

ディスプレイセットを構成する複数種類のセグメントは、図 1 8 B に一例が示されるように、それぞれヘッダ情報により区別されて P E S (Packetized Elementary Stream) パケットのペイロードに格納される。P E S パケットは、最大サイズが 6 4 k B (キロバイト) と決められているので、比較的サイズが大きな O D S や I C S は、1 セグメントのサイズがこの最大サイズを越えた場合、分割されて複数の P E S パケットのペイロードにそれぞれ詰め込まれる。一方、P C S、W D S および P D S は、サイズが 6 4 k B に満たない場合が多いため、1 つの P E S パケットに 1 プレゼンテーショングラフィックス分または 1 インタラクティブグラフィックス分のデータを格納可能である。P E S パケットのそれぞれには、ペイロードに格納されたデータが P C S、W D S、P D S および O D S のうち何れであるかの情報 (プレゼンテーショングラフィックスの場合) や、パケット毎の順番などを示す識別情報が P E S ヘッダに格納される。

【 0 0 8 7 】

また、P E S ヘッダには、この P E S パケットのデコード時刻を指定する D T S (Decoding Time Stamp) と、デコードされたデータが出力される時刻を指定する P T S (Presentation Time Stamp) とが格納される。すなわち、P E S パケットのペイロードに格納されたデータは、M P E G システムズにより規定される基準復号器内部の S T C (System Time Clock) が D T S に一致したときに、デコードが開始され、S T C が P T S に一致したときに、デコードされたデータが出力される。

【 0 0 8 8 】

なお、P T S および D T S は、オプションであって、その P E S パケットのペイロードに所定のアクセスユニットの先頭が含まれている場合に、P E S ヘッダに付加される。例えば、1 のセグメントが複数の P E S パケットに分割されて格納されている場合には、セグメントの先頭側が格納される P E S パケットのみに P T S および D T S を持たせるようにできる。

【 0 0 8 9 】

P E S パケットのそれぞれは、さらに所定に分割され、M P E G T S (トランスポートストリーム) によるトランスポートパケット (以下、T S パケット) に詰め込まれる (図 1 8 C)。T S パケット毎の順序や、T S パケットに格納されたデータを識別する識別情報などが P I D (Packet Identification) に格納される。

【 0 0 9 0 】

次に、各セグメントのうち、この発明に関わりが深いものについて、内容をより具体的に説明する。まず、プレゼンテーショングラフィックスのディスプレイセットに含まれる、P C S および W D S について説明する。図 1 9 は、P C S の一例の構造を表すシンタクスを示す。ここでは、シンタクスをコンピュータ装置などのプログラムの記述言語として用いられる C 言語の記述法に基づき示す。これは、他のシンタクスを表す図において、同様である。

【 0 0 9 1 】

ブロック segment_descriptor() は、フィールド segment_type およびフィールド segment_length からなる (図示しない)。フィールド segment_type は、このセグメントの種類を示す。フィールド segment_type の値が「 0 x 1 6 」で、このセグメントが P C S であること

10

20

30

40

50

が示される。なお、値の記述において「0x」は、16進表記であることを示す。フィールドsegment_lengthは、このフィールドsegment_length以降のこのセグメントの長さを示す。

【0092】

ブロックvideo_descriptor()は、フィールドvideo_width、フィールドvideo_heightおよびフィールドframe_rateからなる(図示しない)。フィールドvideo_widthおよびフィールドvideo_heightは、このPCSが含まれるディスプレイセットにより表示される字幕と同時に再生される動画の画枠を示す。また、フィールドframe_rateは、当該動画のフレームレートを示す。

【0093】

ブロックcomposition_descriptor()は、フィールドcomposition_numberおよびフィールドcomposition_stateからなる(図示しない)。フィールドcomposition_numberは、このPCSのエポック内での順番を示す。フィールドcomposition_stateは、このPCSが含まれるディスプレイセットの種類を示す。

【0094】

ディスプレイセットの種類について、概略的に説明する。ディスプレイセットは、エポックスタートディスプレイセット(Epoch Start DisplaySet)、エポックコンティニューディスプレイセット(Epoch Continue DisplaySet)、アキュジションポイントディスプレイセット(Acquisition Point DisplaySet)およびノーマルケースディスプレイセット(Normal Case DisplaySet)の4種類が定義される。

【0095】

エポックスタートディスプレイセットは、エポックの先頭のディスプレイセットである。エポックコンティニューディスプレイセットは、エポックがプレイアイテムを跨ぐ場合に用いられるもので、例えばエポックがプレイアイテム#1と次のプレイアイテム#2とに跨っている場合に、プレイアイテム#2の先頭にエポックコンティニューディスプレイセットが設けられる。

【0096】

アキュジションポイントディスプレイセットは、字幕のデコードに関し、安全なデコード開始点を意味する。すなわち、プレーヤにおいて、ランダムアクセスが行われた場合、アクセス先がエポックの途中になる可能性が高い。そこで、プレーヤは、ランダムアクセス後、ストリームをデコードし再生していく中で、アキュジションポイントディスプレイセットを発見すると、そこから字幕のデコードを開始し字幕表示を行う。

【0097】

ノーマルケースディスプレイセットは、字幕のデコードに関し、安全なデコード開始点ではないディスプレイセットを意味する。例えば、他のディスプレイセットにより表示中の字幕の色や形状を変更したい場合に、変更の内容を記述したディスプレイセットをノーマルケースディスプレイセットに設定する。プレーヤは、字幕のデコードおよび再生中にこのノーマルケースディスプレイセットを発見すると、その内容に従い、現在表示中の字幕の色や形状などを変更する。

【0098】

PCSのシンタクスの説明に戻り、フィールドpalette_update_flagは、パレットの更新が行われるか否かを示すフラグである。フィールドpalette_id_refは、このディスプレイセットにより表示される字幕1画面が参照するパレットのIDを示す。すなわち、このフィールドpalette_id_refにより、PGストリームにおけるPDS中の色情報が参照される。

【0099】

フィールドnumber_of_composition_objectsは、このディスプレイセットにより表示される字幕1画面上に存在するオブジェクトの数を示す。このフィールドnumber_of_composition_objectsで示される回数だけ、次のfor文によるループが繰り返され、ブロックcomposition_object()により、このディスプレイセットにより表示されるオブジェクト毎の定

10

20

30

40

50

義がなされる。

【 0 1 0 0 】

図 2 0 は、ブロックcomposition_object()の一例の構造を表すシンタクスを示す。フィールドobject_id_refは、このブロックcomposition_object()による字幕表示に用いる O D S に対する参照を示す。フィールドwindow_id_refは、このブロックcomposition_object()による字幕表示に用いるオブジェクトが表示されるウィンドウの I D を示す。

【 0 1 0 1 】

フィールドobject_cropped_flagは、このブロックcomposition_object()による字幕表示に用いるオブジェクトがクロップされるか否かを示すフラグである。すなわち、字幕表示は、O D S に保持されるサイズから所定サイズを切り出して表示することができる。フラグobject_cropped_flagがクロップすることを示していれば、後述するクロップサイズを示す情報に基づき切り取られた字幕が表示される。

【 0 1 0 2 】

フィールドforced_on_flagは、このブロックcomposition_object()による字幕表示が、ユーザオペレーションによる字幕表示 O N / O F F の指示に関わらず強制的に表示されるか否かを示す。

【 0 1 0 3 】

フィールドcomposition_object_horizontal_positionおよびフィールドcomposition_object_vertical_positionは、それぞれこのブロックcomposition_object()による字幕表示の画面上の水平位置および垂直位置を示す。この水平位置および垂直位置は、例えば後述する W D S に示されるウィンドウの位置情報に対する相対位置で示される。

【 0 1 0 4 】

上述のフィールドobject_cropped_flagに示されるフラグの値がオブジェクトがクロップされることを示す値であれば、次のif(object_cropped_flag==1b)以下の記述がなされ、字幕を切り出す位置やサイズが指定される。すなわち、ブロックcropping_rectangle()内のフィールドobject_cropping_horizontal_positionおよびフィールドobject_cropping_vertical_positionにより、切り出す原点の水平位置および垂直位置が示され、フィールドobject_cropping_widthおよびフィールドobject_cropping_heightにより切り出す幅および高さが示される。

【 0 1 0 5 】

図 2 1 は、W D S の一例の構造を表すシンタクスを示す。ブロックsegment_descriptor()は、フィールドsegment_typeとフィールドsegment_lenghtとからなる(図示しない)。フィールドsegment_typeは、このセグメントの種類を示す。フィールドsegment_typeの値が「 0 x 1 7 」で、このセグメントが W D S であることが示される。フィールドsegment_lengthは、このフィールドsegment_length以降のこのセグメントの長さを示す。

【 0 1 0 6 】

フィールドnumber_of_windowsは、この W D S が含まれるディスプレイセットにより表示される字幕 1 画面に存在するウィンドウの数を示す。このフィールドnumber_of_windowsで示される数だけ、次のfor文によるループが繰り返され、ブロックwindow()によりウィンドウ毎の定義がなされる。

【 0 1 0 7 】

図 2 2 は、ブロックwindow()の一例の構造を表すシンタクスを示す。フィールドwindow_idは、このウィンドウの I D を示す。フィールドwindow_horizontal_positionおよびフィールドwindow_vertical_positionは、このウィンドウの画面の原点に対する水平位置および垂直位置を示す。フィールドwindow_widthおよびフィールドwindow_heightは、このウィンドウの幅および高さを示す。

【 0 1 0 8 】

次にインタラクティブグラフィックスのディスプレイセットに含まれる、I C S について説明する。I C S の説明に先んじて、図 2 3 を用いてメニュー画面およびボタンの構成を概略的に示す。図 2 3 A に一例が示されるような、複数のボタン 3 0 0、3 0 0、・・・

10

20

30

40

50

が配置されたメニュー画面 11 を考える。

【0109】

このメニュー画面 301 は、図 23B に一例が示されるように、複数枚のメニュー画面によって階層的に構成することができる。例えば、最前面に表示されたメニュー画面のあるボタン 300 を、所定の入力手段によって選択状態から実行状態に遷移させると、そのメニュー画面の 1 枚後ろに位置するメニュー画面が最前面のメニュー画面となるような構成が考えられる。なお、以下では、「所定の入力手段によりボタンの状態を変化させる」ことを、煩雑さを避けるために、「ボタンを操作する」などと、適宜、表記する。

【0110】

メニュー画面 301 に表示される 1 つのボタン 300 は、複数のボタン 302A、302B、・・・が階層化された構成とすることができる（図 23C および図 23D 参照）。これは、換言すれば、1 つのボタン表示位置に複数のボタンを選択的に表示することができることを意味する。例えば、複数のボタンのうち所定のボタンを操作した際に、同時に表示されている他の幾つかのボタンの機能および表示が変更されるような場合に、メニュー画面自体を書き換える必要が無いので、用いて好適である。このような、1 つのボタン位置に選択的に表示される複数のボタンからなる組を、B O G s (Button Overlap Group) と呼ぶ。

【0111】

B O G s を構成するボタンのそれぞれは、通常状態、選択状態および実行状態の 3 状態を有することができる。すなわち、図 23E に一例が示されるように、B O G s を構成するボタンのそれぞれに対して、通常状態、選択状態および実行状態をそれぞれ表すボタン 303A、303B および 303C を用意することができる。さらに、これら 3 状態を表すボタン 303A、303B および 303C のそれぞれに対して、図 23F に一例が示されるように、アニメーション表示を設定することができる。この場合には、アニメーション表示が設定されたボタンは、アニメーション表示に用いるだけの枚数のボタン画像からなることになる。

【0112】

なお、以下では、ボタンのアニメーションを構成するための複数枚のボタン画像のそれぞれを適宜、アニメーションフレームと呼ぶことにする。

【0113】

図 24 は、I C S のヘッダ情報の一例の構造を表すシンタクスを示す。I C S のヘッダは、ブロック segment_descriptor()、ブロック video_descriptor()、ブロック composition_descriptor()、ブロック sequence_descriptor() および ブロック interactive_composition_data_framemnt() からなる。ブロック segment_descriptor() は、このセグメントが I C S であることを示す。ブロック video_descriptor() は、このメニューと同時に表示されるビデオのフレームレートや画枠サイズを示す。ブロック composition_descriptor() は、フィールド composition_state を含み（図示しない）、この I C S のステータスを示す。ブロック sequence_descriptor() は、この I C S が複数の P E S パケットに跨るか否かを示す。

【0114】

より具体的には、このブロック sequence_descriptor() は、現在の P E S パケットに含まれる I C S が 1 つの I G ストリームの先頭または最後尾の何れの I C S であるか否かを示す。

【0115】

すなわち、上述したように、データサイズが 64 K B と固定的とされた P E S パケットに対して、I C S のデータサイズが大きい場合には、I C S が所定に分割されて P E S パケットにそれぞれ詰め込まれる。このとき、この図 24 に示されるヘッダ部分は、I C S が分割して詰め込まれた P E S パケットのうち先頭および最後尾の P E S パケットにだけあればよく、途中の P E S パケットにおいては省略することができる。このブロック sequence_descriptor() が先頭および最後尾を示していれば、I C S が 1 つの P E S パケット

10

20

30

40

50

に収まっていることが分かる。

【 0 1 1 6 】

図 2 5 は、ブロック `interactive_composition_data_fragemnt()` の一例の構造を表すシンタクスを示す。なお、図 2 5 では、自身のブロックがブロック `interactive_composition()` として示されている。フィールド `intaractive_composition_length` は、24 ビットのデータ長を有し、ブロック `interactive_composition()` のこのフィールド `intaractive_composition_length` 以降の長さを示す。フィールド `stream_model` は、1 ビットのデータ長を有し、このストリームが多重化されているか否かを示す。すなわち、インタラクティブグラフィクスストリームは、AV ストリームに対して多重化することもできるし、単独でクリップ AV ストリームを構成することもできる。

10

【 0 1 1 7 】

フィールド `user_interface_model` は、1 ビットのデータ長を有し、このストリームが表示するメニューがポップアップメニューか、常時表示メニューの何れであることを示す。ポップアップメニューは、例えばリモートコントロールコマンドのボタンの ON / OFF といった、所定の入力手段で表示の有無を制御できるメニューである。一方、常時表示メニューは、ユーザ操作により表示の有無を制御できない。フィールド `user_interface_model` の値が "0" で、ポップアップメニューを示し、"1" で常時表示メニューを示す。

【 0 1 1 8 】

フィールド `user_interface_model` の値が "0" であれば、If 文 `If(stream_model=='00')` 以下のフィールド `composition_time_out_pts` およびフィールド `selection_time_out_pts` が有効になる。フィールド `composition_time_out_pts` は、33 ビットのデータ長を有し、このメニュー表示が消える時刻を示す。また、フィールド `selection_time_out_pts` は、33 ビットのデータ長を有し、このメニュー表示における選択操作が可能でなくなる時刻を示す。時刻は、それぞれ MPEG 2 に規定される PTS (Presentation Time Stamp) で記述される。

20

【 0 1 1 9 】

フィールド `user_time_out_duration` は、24 ビットのデータ長を有し、このメニュー表示の自動初期化時間を示す。次のフィールド `number_of_page` は、8 ビットのデータ長を有し、このメニューが有するページ数を、初期値を "0" として示す。すなわち、このメニュー表示が図 2 3 B を用いて説明した階層構造を有し、複数のページを有するときは、フィールド `number_of_page` の値は、"1" 以上を示す。このフィールド `number_of_page` で示された回数だけ、次の for 文からのループが繰り返され、メニュー表示における各ページがそれぞれ定義される。

30

【 0 1 2 0 】

図 2 6 は、ブロック `page()` の一例の構造を表すシンタクスを示す。フィールド `page_id` は、8 ビットのデータ長を有し、このページを識別するための ID を示し、フィールド `page_version_number` は、8 ビットのデータ長を有し、このページのバージョン番号を示す。次のブロック `UO_mask_table()` は、このページの表示中に禁止される、ユーザの入力手段に対する操作 (UO : User Operation) が記述されるテーブルを示す。

【 0 1 2 1 】

ブロック `in_effect()` は、このページが表示される際に表示されるアニメーションブロックを示す。括弧 {} 内のブロック `effect_sequence()` に、アニメーションのシーケンスが記述される。また、ブロック `out_effect()` は、このページが終了する際に表示されるアニメーションブロックを示す。括弧 {} 内のブロック `effect_sequence()` に、アニメーションのシーケンスが記述される。これらブロック `in_effect()` およびブロック `out_effect()` は、それぞれページを移動した際にこの ICS が発見された場合に実行されるアニメーションである。

40

【 0 1 2 2 】

次のフィールド `animation_frame_rate_code` は、8 ビットのデータ長を有し、このページのボタン画像がアニメーションする場合の、アニメーションフレームレートの設定パラ

50

メータを示す。例えば、このICSが対応するクリップAVストリームファイルにおけるビデオデータのフレームレートを V_{frm} 、当該アニメーションフレームレートを A_{frm} とした場合、フィールドanimation_frame_rate_codeの値を、 V_{frm} / A_{frm} のように、これらの比で表すことができる。

【0123】

フィールドdefault_selected_button_id_refは、16ビットのデータ長を有し、このページが表示される際に最初に選択状態とされるボタンを指定するためのIDを示す。また、次のフィールドdefault_activated_button_id_refは、16ビットのデータ長を有し、図25を用いて説明したフィールドselection_time_out_ptsで示される時間に達した際に、自動的に実行状態となるボタンを指定するためのIDを示す。

10

【0124】

フィールドpalette_id_refは、8ビットのデータ長を有し、このページが参照するパレットのIDを示す。すなわち、このフィールドpalette_id_refにより、IGストリームにおけるPDS中の色情報が指定される。

【0125】

次のフィールドnumber_of_BOGsは、8ビットのデータ長を有し、このページで用いられるBOGsの数が示される。このフィールドnumber_of_BOGsで示される回数だけ、次のfor文からのループが繰り返され、ブロックbutton_overlap_group()により、BOGs毎の定義がなされる。

【0126】

20

図27は、ブロックbutton_overlap_group()の一例の構造を表すシンタクスを示す。フィールドdefault_valid_button_id_refは、16ビットのデータ長を有し、このブロックbutton_overlap_group()で定義されるBOGsにおいて最初に表示されるボタンのIDを示す。次のフィールドnumber_of_buttonsは、8ビットのデータ長を有し、このBOGsで用いられるボタンの数を示す。そして、このフィールドnumber_of_buttonsで示される回数だけ、次のfor文からのループが繰り返され、ブロックbutton()によりボタンそれぞれの定義がなされる。

【0127】

すなわち、既に説明したように、BOGsは、複数のボタンを持つことができ、BOGsが持つ複数のボタンのそれぞれの構造がブロックbutton()で定義される。このブロックbutton()で定義されるボタン構造が、実際に表示されるボタンである。

30

【0128】

図28は、ブロックbutton()の一例の構造を表すシンタクスを示す。フィールドbutton_idは、16ビットのデータ長を有し、このボタンを識別するIDを示す。フィールドbutton_numeric_select_valueは、16ビットのデータ長を有し、このボタンがリモートコントロールコマンド上の数字キーの何番に割り当てられているかを示す。フラグauto_action_flagは、1ビットのデータ長を有するフラグであって、このボタンが選択状態になったときに、当該ボタンに割り当てられた機能が自動的に実行されるか否かを示す。

【0129】

次のフィールドbutton_horizontal_positionおよびフィールドbutton_vertical_positionは、それぞれ16ビットのデータ長を有し、このボタンを表示する画面上の水平方向の位置および垂直方向の位置（高さ）を示す。

40

【0130】

ブロックneighbor_info()は、このボタンの周辺情報を示す。すなわち、このブロックneighbor_info()内の値により、このボタンが例えば選択状態となっている状態からリモートコントロールコマンドにおける上下左右の方向を指示できる方向キーが操作された際に、どのボタンが選択状態に遷移するかを示す。ブロックneighbor_info()内のフィールドにおいて、それぞれ16ビットのデータ長を有するフィールドupper_button_id_ref、フィールドlower_button_id_ref、フィールドleft_button_id_refおよびフィールドright_button_id_refは、上方向、下方向、左方向および右方向を指示する操作がなされた場合に

50

選択状態に遷移されるボタンのIDをそれぞれ示す。

【0131】

次からのブロックnormal_state_info()、ブロックselected_state_info()およびブロックactivated_state_info()は、それぞれ、通常状態、選択状態および実行状態時のボタンの情報を示す。

【0132】

まず、ブロックnormal_state_info()について説明する。それぞれ16ビットのデータ長を有するフィールドnormal_start_object_id_refおよびフィールドnormal_end_object_id_refは、この通常状態のボタンのアニメーションの先頭および末尾のオブジェクトを指定するIDがそれぞれ示される。すなわち、これらフィールドnormal_start_object_id_refおよびフィールドnormal_end_object_id_refにより、ボタンのアニメーション表示に用いられるボタン画像(すなわちアニメーションフレーム)が、対応するODSに対して指定される。

10

【0133】

次のフラグnormal_repeat_flagは、1ビットのデータ長を有するフラグであって、このボタンのアニメーションがリピートするか否かが示される。例えば、フラグnormal_repeat_flagの値が"0"でリピートしないことを示し、値が"1"でリピートすることを示す。

【0134】

次のフラグnormal_complete_flagは、1ビットのデータ長を有するフラグであって、このフラグがこの発明により定義される。このフラグnormal_complete_flagは、このボタンが通常状態から選択状態に遷移する際のアニメーション動作を制御するフラグである。

20

【0135】

すなわち、フラグnormal_complete_flagの値が"1"であれば、このボタンが通常状態から選択状態へと遷移する際に、通常状態に定義されているアニメーションを、全て表示する。より具体的には、フラグnormal_complete_flagの値が"1"であったなら、当該ボタンの通常状態のアニメーション表示中に当該ボタンを通常状態から選択状態に遷移させるような入力となされた場合、その時点で表示されているアニメーションフレームから、上述したフィールドnormal_end_object_id_refで示されるアニメーションフレームまで、アニメーション表示を行う。

【0136】

30

また、フラグnormal_complete_flagの値が"1"であって、且つ、フラグnormal_repeat_flagがリピートすることを示している場合(例えば値"1")も、その時点で表示されているアニメーションフレームから、上述したフィールドnormal_end_object_id_refで示されるアニメーションフレームまで、アニメーション表示を行う。

【0137】

この場合、例えばボタンが選択できない状態にされた場合や、ボタンの表示そのものが消されてしまうような場合でも、これらの状態に遷移する時点がアニメーションの表示中であれば、フィールドnormal_end_object_id_refで示されるアニメーションフレームまでアニメーション表示を行い、その後、ボタンの状態を遷移させる。

【0138】

40

ボタンが選択できない状態は、例えば上述したフィールドselection_time_out_ptsの指定により、ボタンが選択できないような状態になった場合や、フィールドuser_time_out_durationの指定によりメニューが自動的に初期化されるような場合が考えられる。

【0139】

一方、フラグnormal_complete_flagの値が"0"であれば、このボタンが通常状態から選択状態へと遷移する際に、通常状態のボタンに定義されたアニメーションをフィールドnormal_end_object_id_refで示されるアニメーションフレームまで表示せずに、状態の遷移が指示された時点でアニメーション表示を停止し、選択状態のボタンを表示する。

【0140】

次に、ブロックselected_state_info()について説明する。このブロックselected_stat

50

e_info()は、上述のブロックnormal_state_info()に対して、サウンドを指示するためのフィールドselected_state_sound_id_refが付加されている。フィールドselected_state_sound_id_refは、8ビットのデータ長を有し、この選択状態のボタンに対して再生されるサウンドファイルを示す。例えば、サウンドファイルは、ボタンが通常状態から選択状態に遷移した際の効果音として用いられる。

【0141】

それぞれ16ビットのデータ長を有するフィールドselected_start_object_id_refおよびフィールドselected_end_object_id_refは、この選択状態のボタンのアニメーションの先頭および末尾のオブジェクトを指定するIDがそれぞれ示される。また、次の1ビットのデータ長を有するフラグであるフラグselected_repeat_flagは、このボタンのアニメーションがリピートするか否かが示される。例えば値が"0"でリピートしないことを示し、値が"1"でリピートすることを示す。

10

【0142】

次のフラグselected_complete_flagは、1ビットのデータ長を有するフラグであって、上述のフラグnormal_complete_flagと共に、この発明により定義されるフラグである。このフラグselected_complete_flagは、このボタンが選択状態から他の状態に遷移する際のアニメーション動作を制御するフラグである。すなわち、フラグselected_complete_flagは、ボタンが選択状態から実行状態に遷移する場合と、選択状態から通常状態に遷移する場合とに、用いることができる。

20

【0143】

上述と同様、フラグselected_complete_flagの値が"1"であれば、このボタンが選択状態から他の状態へと遷移する際に、選択状態に定義されているアニメーションを、全て表示する。より具体的には、フラグselected_complete_flagの値が"1"であったなら、当該ボタンの選択状態のアニメーション表示中に当該ボタンを選択状態から他の状態に遷移させるような入力がなされた場合、その時点で表示されているアニメーションフレームから、上述したフィールドselected_end_object_id_refで示されるアニメーションフレームまで、アニメーション表示を行う。

【0144】

また、フラグselected_complete_flagの値が"1"であって、且つ、フラグselected_repeat_flagがリピートすることを示している場合（例えば値"1"）も、その時点で表示されているアニメーションフレームから、上述したフィールドselected_end_object_id_refで示されるアニメーションフレームまで、アニメーション表示を行う。

30

【0145】

この場合、例えばボタンが選択できない状態にされた場合や、ボタンの表示そのものが消されてしまうような場合でも、これらの状態に遷移する時点がアニメーションの表示中であれば、フィールドselected_end_object_id_refで示されるアニメーションフレームまでアニメーション表示を行い、その後、ボタンの状態を遷移させる。

【0146】

ボタンが選択できない状態は、例えば上述したフィールドselection_time_out_ptsの指定により、ボタンが選択できないような状態になった場合や、フィールドuser_time_out_durationの指定によりメニューが自動的に初期化されるような場合が考えられる。

40

【0147】

一方、フラグselected_complete_flagの値が"0"であれば、このボタンが選択状態から他の状態へと遷移する際に、選択状態のボタンに定義されたアニメーションをフィールドselected_end_object_id_refで示されるアニメーションフレームまで表示せずに、状態の遷移が指示された時点でアニメーション表示を停止し、他の状態のボタンを表示する。

【0148】

次のブロックactivated_state_info()は、上述したブロックnormal_state_info()およびブロックselected_state_info()に対して、アニメーションのリピートを行うか否かを示すフラグと、ボタンが実行状態から他の状態に遷移する際のアニメーション動作を制御

50

するフラグが定義されない。ボタンが実行状態にされた後は、当該ボタンに割り当てられた機能が実行されることになり、ボタンが実行状態にある時間は非常に短いと考えられる。また、実行状態にあるボタンが他の状態に遷移するような制御も、一般的には好ましくないといえる。そのため、ブロックactivated_state_info()において、これらのフラグを省略している。勿論、ブロックactivated_state_info()においてこれらのフラグを定義することも可能である。

【 0 1 4 9 】

ブロックactivated_state_info()において、フィールドactivated_state_sound_id_refは、8ビットのデータ長を有し、この実行状態のボタンに対して再生されるサウンドファイルを示す。それぞれ16ビットのデータ長を有するフィールドactivated_start_object_id_refおよびフィールドactivated_end_object_id_refは、この実行状態のボタンのアニメーションの先頭および末尾のアニメーションフレームを指定するIDがそれぞれ示される。

10

【 0 1 5 0 】

ブロックactivated_state_info()の説明を終わり、次のフィールドnumber_of_navigation_commandsは、16ビットのデータ長を有し、このボタンに埋め込まれるコマンドの数を示す。そして、このフィールドnumber_of_navigation_commandsで示される回数だけ、次のfor文からのループが繰り返され、このボタンにより実行されるコマンドnavigation_command()が定義される。これは、換言すれば、1つのボタンから複数のコマンドを実行させることができることを意味する。

20

【 0 1 5 1 】

次に、現時点のBD-ROM規格に定義されるプレゼンテーショングラフィクスおよびインタラクティブグラフィクスのデコードモデルについて説明する。図29は、プレゼンテーショングラフィクスおよびインタラクティブグラフィクスに共通して適用可能なデコードモデルを示す。

【 0 1 5 2 】

まず、ディスクがプレーヤに装填されると、インデックスファイル"index.bdmv"およびムービーオブジェクトファイル"MovieObject.bdmv"がディスクから読み込まれ、所定にトップメニューが表示される。トップメニューの表示に基づき、再生するタイトルを指示すると、ムービーオブジェクトファイル中の対応するナビゲーションコマンドにより、指示されたタイトルを再生するためのプレイリストファイルが呼び出される。そして、当該プレイリストファイルの記述に従い、プレイリストから再生を要求されたクリップAVストリームファイル、すなわち、MPEG2のトランスポートストリームがディスクから読み出される。

30

【 0 1 5 3 】

トランスポートストリームは、TSパケットとしてPIDフィルタ100に供給されてPIDが解析される。PIDフィルタ100では、供給されTSパケットがビデオデータ、オーディオデータ、メニューデータおよびサブタイトル(字幕)データの何れを格納したパケットであるかを分類する。PIDがメニューデータすなわちインタラクティブグラフィクスを表している場合、または、サブタイトルすなわちプレゼンテーショングラフィクスを表している場合に、図29の構成が有効とされる。

40

【 0 1 5 4 】

PIDフィルタ100において、トランスポートストリームからデコードモデルが対応するデータを格納するTSパケットが選択されて、トランスポートバッファ(以下、TB: Transport Buffer)101に溜め込まれる。そして、TB101上でTSパケットのペイロードに格納されているデータが取り出される。PESパケットを構成可能なだけのデータがTB101に溜め込まれたら、PIDに基づきPESパケットが再構築される。すなわち、この段階で、TSパケットに分割されていた各セグメントがそれぞれ統合されることになる。

【 0 1 5 5 】

50

各セグメントによる P E S パケットは、P E S ヘッダが取り除かれエレメンタリストリーム形式でデコーダ 1 0 2 にそれぞれ供給され、C D B (コーデッドデータバッファ : Cod ed Data Buffer) 1 1 0 に一旦溜め込まれる。S T C に基づき、C D B 1 1 0 に溜め込まれたエレメンタリストリームのうち、対応する D T S に示される時刻に達したエレメンタリストリームがあれば、当該セグメントは、C D B 1 1 0 から読み出されてストリームグラフィックプロセッサ 1 1 1 に転送され、それぞれデコードされセグメントに展開される。

【 0 1 5 6 】

ストリームグラフィックプロセッサ 1 1 1 は、デコードが完了したセグメントを、D B (デコーデッドオブジェクトバッファ : Decoded Object Buffer) 1 1 2 または C B (コンポジションバッファ : Composition Buffer) 1 1 3 に所定に格納する。P C S、I C S、W D S および O D S のように D T S を持つタイプのセグメントであれば、対応する D T S に示されるタイミングで、D B 1 1 2 または C B 1 1 3 に格納する。また、P D S のように D T S を持たないタイプのセグメントは、即座に C B 1 1 3 に格納される。

【 0 1 5 7 】

グラフィクスコントローラ 1 1 4 は、セグメントを制御する。例えば、このデコーダモデルがプレゼンテーショングラフィクスによるデータ (以下、P G データ) に対応するモデルである場合、グラフィクスコントローラ 1 1 4 は、C B 1 1 3 から P C S を、対応する P T S に示されるタイミングで読み出すと共に、当該 P C S により参照される W D S および P D S を読み出す。また、グラフィクスコントローラ 1 1 4 は、当該 P C S から参照される O D S を D B 1 1 2 から読み出す。そして、読み出された P C S および W D S、ならびに、O D S をそれぞれデコードし、字幕を表示するためのデータを形成し、グラフィクスプレーン 1 0 3 に書き込む。

【 0 1 5 8 】

同様に、例えばこのデコーダモデルがインタラクティブグラフィクスによるデータ (以下、I G データ) に対応するモデルである場合、グラフィクスコントローラ 1 1 4 は、C B 1 1 3 から I C S を、当該 I C S に対応する P T S に示されるタイミングで読み出すと共に、当該 I C S により参照される P D S を読み出す。また、グラフィクスコントローラ 1 1 4 は、当該 I C S から参照される O D S を D B 1 1 2 から読み出す。そして、読み出された I C S および O D S をそれぞれデコードし、ボタン画像などメニュー画面を表示するためのデータを形成し、グラフィクスプレーン 1 0 3 に書き込む。

【 0 1 5 9 】

また、グラフィクスコントローラ 1 1 4 は、C B 1 1 3 から読み出された P D S をデコードして、例えば図 1 3 を用いて説明したようなカラーパレットテーブルを形成し、C L U T 1 0 4 に書き込む。

【 0 1 6 0 】

グラフィクスプレーン 1 0 3 に書き込まれた画像イメージは、所定のタイミング、例えばフレームタイミングで読み出され、C L U T 1 0 4 のカラーパレットテーブルが参照されて色情報が付加されて出力画像データが構成され、この出力画像データが出力される。プレーヤの仕様によっては、グラフィクスプレーン 1 0 3 から読み出されたデータに対して C L U T 1 0 4 のカラーパレットテーブルに基づく色情報を付加した字幕画像データを、さらにフレームメモリに書き込む場合も考えられる。

【 0 1 6 1 】

なお、エポックの更新に伴い、グラフィクスコントローラ 1 1 4 により D B 1 1 2、C B 1 1 3、グラフィクスプレーン 1 0 3 および C L U T 1 0 4 がクリアされる。これらバッファやプレーンのクリアは、C D B 1 1 0 から読み出されてデコードされた P C S または I C S におけるブロック composition_descriptor() 中のフィールド composition_state の記述が、このディスプレイセットがエポックスタートディスプレイセットであることを示している場合に、行われる。エポックの更新に伴い、C D B 1 1 0 を共にクリアすることも可能である。

10

20

30

40

50

【 0 1 6 2 】

ここで、エポックの定義について説明する。既に説明したように、プレゼンテーショングラフィクスおよびインタラクティブグラフィクスにおいて、エポックは、エポックスタートディスプレイセットに基づき定義される。

【 0 1 6 3 】

プレゼンテーショングラフィクスについては、次の項目（ 1 ）および項目（ 2 ）で示す 2 項目のうち短い方がエポックとして定義される。

【 0 1 6 4 】

項目（ 1 ）：エポックスタートディスプレイセットが P C S に定義されるディスプレイセットから、次のエポックスタートディスプレイセットが P C S に定義されるディスプレイセットの 1 つ前のディスプレイセットまで。

項目（ 2 ）：エポックスタートディスプレイセットが P C S に定義されるディスプレイセットからその P G データが関連するプレイリストの最後まで。

【 0 1 6 5 】

なお、より具体的には、項目（ 1 ）の場合、エポックスタートディスプレイセットが定義される P C S の P T S から、次のエポックスタートディスプレイセットが定義される P C S の P T S までがエポックとされる。項目（ 2 ）の場合も同様に、エポックスタートディスプレイセットが定義される P C S の P T S から、当該ディスプレイセットが参照されるプレイリストの最後までがエポックとされる。

【 0 1 6 6 】

インタラクティブグラフィクスについては、次の項目（ 3 ）、項目（ 4 ）および項目（ 5 ）のうち、最も短いものがエポックとして定義される。

【 0 1 6 7 】

項目（ 3 ）：エポックスタートディスプレイセットが I C S に定義されるディスプレイセットから、当該 I C S のフィールド composition_time_out_pts に定義される時刻まで。

項目（ 4 ）：エポックスタートディスプレイセットが I C S に定義されるディスプレイセットから、次のエポックスタートディスプレイセットが I C S に定義されるディスプレイセットの 1 つ前のディスプレイセットまで。

項目（ 5 ）：エポックスタートディスプレイセットが I C S に定義されるディスプレイセットから当該 I G データが関連するプレイリストの最後まで。

【 0 1 6 8 】

図 3 0 は、グラフィクスデコーダ 1 0 2 における、 P G データに関するディスプレイセット単位での一例の処理を示すフローチャートである。ステップ S 1 0 で、 P I D フィルタ 1 0 0 により T S パケットから P I D に基づき P G データが格納されたパケットが取得され、 T B 1 0 1 に溜め込まれる。次のステップ S 1 1 で、 T B 1 0 1 に P E S パケットを構成可能なだけの T S パケットが溜め込まれたら、 P I D に基づき T S パケットのペイロードからデータが取り出され、 P E S パケットが再構築される。 P E S パケットは、 P E S ヘッダが取り除かれてエレメンタリストリーム形式で C D B 1 1 0 に転送される。

【 0 1 6 9 】

P E S パケットから取り出されたエレメンタリストリームは、ストリームグラフィックプロセッサ 1 1 1 においてセグメントに展開される（ステップ S 1 2 ）。そして、次のステップ S 1 3 ～ステップ S 1 7 までで、展開されたセグメントが P C S 、 W D S 、 P D S および O D S の何れであるか、若しくは、 E N D セグメントであるか否かが判別され、判別結果に応じた処理が行われる。セグメントの種類は、上述の図 1 9 や図 2 1 などを用いて説明したように、シンタクスの先頭のブロック segment_description() における最初のフィールド segment_type に基づき判別できる。なお、図示しないが、 W D S および O D S も、同様の方法でセグメントの種類を判別可能である。

【 0 1 7 0 】

ステップ S 1 3 で、セグメントが P C S であるか否かが判別される。若し、セグメントが P C S であると判別されれば、処理はステップ S 1 9 に移行され、図 1 9 を用いて説明

10

20

30

40

50

したPCSのシンタクスにおけるブロックcomposition_descriptor()内のフィールドcomposition_stateに基づき、このPCSが含まれるディスプレイセットがエポックスタートディスプレイセットであるか否かが判断される。

【0171】

若し、当該ディスプレイセットがエポックスタートディスプレイセットであると判断されれば、処理はステップS20に移行され、字幕表示に関わるバッファのクリアがなされる。例えば、DB112およびCB113がステップS20でクリアされる。このとき、CDB110を共にクリアすることも可能である。そして、次のステップS21で、クリアされ空になったCB113に、DTSのタイミングでPCSが書き込まれる。CB113へのPCSの書き込みが終了すると、処理はステップS10に戻され、パケットの取得が継続される。

10

【0172】

一方、ステップS19で、PCSがエポックスタートディスプレイセットではないと判断されれば、処理はステップS22に移行され、PCSがCB113にDTSのタイミングで書き込まれる。このとき、PCSは、CB113の空き領域に所定に書き込まれる。

【0173】

上述のステップS13で、セグメントがPCSではないと判別されれば、処理はステップS14に移行され、セグメントがWDSであるか否かが判別される。若し、セグメントがWDSであると判別されれば、処理はステップS23に移行され、WDSがCB113にDTSのタイミングで書き込まれる。CB113へのWDSの書き込みが終了すると、処理はステップS10に戻され、パケットの取得が継続される。ステップS14でセグメントがWDSではないと判別されれば、処理はステップS15に移行される。

20

【0174】

ステップS15では、セグメントがPDSであるか否かが判別される。若し、セグメントがPDSであると判別されれば、処理はステップS24に移行され、PDSが即座にCB113に書き込まれる。これは、PDSは、PTSしか定義されず、DTSを持たないので、PDSであると判別された時点で、即座にCB113に書き込まれる。CB113へのWDSの書き込みが終了すると、処理はステップS10に戻され、パケットの取得が継続される。ステップS15でセグメントがPDSではないと判別されれば、処理はステップS16に移行される。

30

【0175】

ステップS16では、セグメントがODSであるか否かが判別される。若し、セグメントがODSであると判別されれば、処理はステップS25に移行され、ODSに格納された、ランレングス処理されたビットマップデータがDTSのタイミングでデコードされ、DB112に書き込まれる。DB112へのODSの書き込みが終了すると、処理はステップS10に戻され、パケットの取得が継続される。ステップS16でセグメントがODSではないと判別されれば、処理はステップS17に移行される。

【0176】

ステップS17では、セグメントがディスプレイセットの最後尾を示すENDセグメントであるか否かが判断される。若し、セグメントがENDセグメントであると判別されれば、このディスプレイセットに関する一連の処理が終了される。

40

【0177】

一方、ステップS17で、セグメントがENDセグメントではないと判別されれば、処理はステップS18に移行される。すなわち、この場合には、何らかのエラーが発生したものと判断でき、所定にエラー処理がなされる。

【0178】

なお、上述の構成では、PCSに定義されるPTSのタイミングで、字幕を表示するために用いるPCSがCB113から排出され字幕が表示されると共に、当該PCSがCB113上から破棄されるようになっている。これによれば、ユーザオペレーションにより字幕表示OFFが指示され字幕が消去されてしまうと、次に字幕表示ONを指示して字幕

50

を再び表示させようとしても、字幕表示ONの指示後の、PCSに定義されたPTSに到達するまで字幕表示を行うことができないことになる。

【0179】

そこで、現在アクティブなODSを保持するためのバッファをDB112とは別に設けると共に、現在アクティブなPCSおよびPDSを保持するためのバッファをCB113とは別に設けることができる。字幕表示ONが指示されると、これらDB112やCB113とは別に設けられたバッファに保持されたODS、PCSおよびPDSを用いて、字幕画像イメージおよびカラーパレットテーブルを形成し、グラフィクスプレーン103およびCLUT104に対してそれぞれ書き込む。

【0180】

なお、アクティブとは、PTSにより表示が指示された状態を指し、より具体的には、あるPCSにおいて、自身に定義されたPTSから次の字幕表示に用いられるPCSに定義されたPTSまでの期間がアクティブとされる。

【0181】

すなわち、グラフィクスコントローラ114は、PCS毎に定義されたPTSのタイミングでCB113から排出されるPCSを、上述のようにしてグラフィクスプレーン103およびCLUT104にそれぞれ供給すると共に、CB113とは別に設けられたバッファに対し、次のPCSに定義されたPTSのタイミングまで保持するようにしている。ODSについても同様に、DB112からPTSのタイミングで排出されるODSをグラフィクスプレーン103に供給すると共に、DB112とは別に設けられたバッファに対し、次のPCSに定義されたPTSのタイミングまで保持する。

【0182】

このように構成することで、ユーザオペレーションによる字幕表示OFFの指示に応じて字幕が消去された後に、字幕表示ONを指示した場合に、字幕表示ONのタイミングに応じて即座に、字幕の表示を再開できるようになる。

【0183】

すなわち、字幕表示OFFが指示された後に字幕表示ONが指示されると、CB113とは別に設けられたバッファからPCSを読み出すと共に、DB112とは別に設けられたバッファからODSを読み出し、これら読み出されたPCSおよびODSを用いて字幕画像イメージを形成する。形成されたこの字幕画像イメージは、グラフィクスプレーン103に書き込まれる。PDSについても同様に、字幕表示ONの指示のタイミングでCB113とは別に設けられたバッファからPDSを読み出し、読み出されたPDSを用いてカラーパレットテーブルを形成する。形成されたカラーパレットテーブルは、CLUT104に書き込まれる。

【0184】

図31は、グラフィクスデコーダ102における、IGデータに関するディスプレイセット単位での一例の処理を示すフローチャートである。この図31に一例が示される処理は、PGデータにのみ存在するWDSに関する処理を除き、略図30に示されるPGデータに関する処理と同等である。

【0185】

ステップS30で、PIDフィルタ100によりTSパケットからPIDに基づきIGデータが格納されたパケットが取得され、TB101に溜め込まれる。次のステップS31で、TB101に溜め込まれたTSパケットのペイロードからPIDに基づきデータが取り出され、PESパケットが再構築され、PESヘッダが取り除かれてエレメンタリストリーム形式でCDB111に転送される。

【0186】

PESパケットから取り出されたエレメンタリストリームは、ストリームグラフィックプロセッサ111においてセグメントに展開される(ステップS32)。そして、ステップS33以下の処理により、展開されたセグメントがICS、PDSおよびODSの何れであるか、若しくは、ENDセグメントであるか否かが判別され、判別結果に応じた処理

10

20

30

40

50

が行われる。セグメントの種類は、上述の図 2 4 を用いて説明したように、シンタクスの先頭のブロック `segment_description()` に基づき判別できる。なお、図示しないが、ODS も、同様の方法でセグメントの種類を判別可能である。

【0187】

ステップ S 3 3 で、セグメントが ICS であるか否かが判別される。若し、セグメントが ICS であると判別されれば、処理はステップ S 3 8 に移行され、図 2 4 を用いて説明した ICS のシンタクスにおけるブロック `composition_descriptor()` に基づき、この ICS が含まれるディスプレイセットがエポックスタートディスプレイセットであるか否かが判断される。

【0188】

若し、当該ディスプレイセットがエポックスタートディスプレイセットであると判断されれば、処理はステップ S 3 9 に移行され、メニュー表示に関わるバッファのクリアがなされる。例えば、DB 1 1 2 および CB 1 1 3 がステップ S 3 9 でクリアされる。このとき、CDB 1 1 0 を共にクリアすることも可能である。そして、次のステップ S 4 0 で、クリアされ空になった CB 1 1 3 に、DTS のタイミングで PCS が書き込まれる。CB 1 1 3 への PCS の書き込みが終了すると、処理はステップ S 3 0 に戻され、パケットの取得が継続される。

【0189】

一方、ステップ S 3 8 で、ICS がエポックスタートディスプレイセットではないと判断されれば、処理はステップ S 4 1 に移行され、ICS が CB 1 1 3 に DTS のタイミングで書き込まれる。このとき、ICS は、CB 1 1 3 の空き領域に所定に書き込まれる。

【0190】

上述のステップ S 3 3 で、セグメントが ICS ではないと判別されれば、処理はステップ S 3 4 に移行される。ステップ S 3 4 以降の処理は、上述した図 3 0 のフローチャートにより処理と同様に、セグメントが PDS であるか（ステップ S 3 4）、ODS であるか（ステップ S 3 5）、END セグメントであるか（ステップ S 3 6）の判断がなされ、それぞれの判断に応じた処理がステップ S 4 2、ステップ S 4 3 およびステップ S 3 7 において行われる。すなわち、セグメントが PDS であれば、CDB 1 1 3 に即座に書き込まれる（ステップ S 4 2）、セグメントが ODS であれば、DTS でデコードされ DB 1 1 2 に書き込まれる（ステップ S 4 3）。

【0191】

次に、この発明の実施の一形態について説明する。上述したように、BD-ROM 規格で定義されるデコーダモデルによれば、PG データおよび IG データに関し、CDB 1 1 0 にセグメントを溜め込み、セグメントのそれぞれに定義された DTS の時刻でデコードして DB 1 1 2 または CB 1 1 3 に保持し、セグメントのそれぞれに定義された PTS の時刻で出力するように規定されている。そのため、BD-ROM 規格で定義されるデコーダモデルに基づきプレーヤを設計する際には、図 3 2 に一例が示されるように、CDB 1 1 0 に溜め込まれたセグメントに対応する DTS および PTS を格納するメモリ 2 0 0（以下、PTS / DTS バッファ 2 0 0 と呼ぶ）が必須であると考えられる。

【0192】

すなわち、このデコーダモデルに基づきプレーヤの制御を行う制御部は、CDB 1 1 0 に溜め込まれたセグメントに対し、PTS / DTS バッファ 2 0 0 に格納された DTS に基づきデコードタイミングを指示し、デコードされ DB 1 1 2 または CB 1 1 3 に格納されたセグメントに対し、PTS / DTS バッファ 2 0 0 に格納された PTS に基づき出力タイミングを指示するように制御する。この PTS / DTS バッファ 2 0 0 は、現時点の BD-ROM 規格では定義されていない。

【0193】

図 3 3 は、PTS / DTS バッファ 2 0 0 に溜め込まれる一例のデータ構成を示す。PTS / DTS バッファ 2 0 0 に対して、CDB 1 1 0 に溜め込まれたセグメントに対応する DTS および PTS が格納されると共に、セグメントのそれぞれを識別するセグメント

10

20

30

40

50

ＩＤが格納される。ＤＴＳおよびＰＴＳは、規定に従いそれぞれ３３ビットのデータ長を有する。

【０１９４】

ＰＴＳ／ＤＴＳバッファ２００に対して、セグメント毎にさらに他のデータを格納してもよい。図３３の例では、セグメントＩＤで示されるセグメントのバージョンを示すバージョン情報と、当該セグメントのタイプを示す情報が格納される。セグメントのタイプを示す情報は、ＰＧデータの場合、当該セグメントがＰＣＳ、ＰＤＳ、ＷＤＳおよびＯＤＳの何れであるかを示し、ＩＧデータの場合、当該セグメントがＩＣＳ、ＰＤＳおよびＯＤＳの何れであるかを示す。

【０１９５】

一例として、図３３に示す項目のうちセグメントＩＤのデータ長を１６ビット、バージョン情報のデータ長を８ビット、セグメントタイプのデータ長を４ビットとすると、１セグメント当たりのデータサイズは、９４ビットとなり、バイト単位で切り上げると１２バイトとなる。

【０１９６】

図３２に示したデコーダモデルに基づき実際にプレーヤを設計する場合、各バッファやメモリについて最低限必要な容量をそれぞれ見積もる必要がある。ＢＤ－ＲＯＭ規格に定義されているバッファであるＴＢ１０１、ＣＤＢ１１０、ＤＢ１１２およびＣＢ１１３については、同規格において既に最低容量が規定されている。したがって、ＢＤ－ＲＯＭ規格に定義されていないＰＴＳ／ＤＴＳバッファ２００について、最低容量を見積もる必要

【０１９７】

既に説明したように、ＰＧデータおよびＩＧデータにおいて、セグメントのデータサイズが可変なので、ＣＤＢ１１０に対して何個のセグメントが溜め込まれるのかを見積もるのは困難である。したがって、ＣＤＢ１１０に溜め込まれたセグメントに対応するＰＴＳおよびＤＴＳの合計数すなわち合計サイズを見積もることも困難となる。

【０１９８】

例えば、ＣＤＢ１１０に対してデータサイズの大きなセグメントが溜め込まれる場合には、ＣＤＢ１１０に格納可能なセグメント数が少なくなるので、対応するＰＴＳおよびＤＴＳの合計サイズが小さなものとなる。例えば、ＰＧデータの場合には、ＯＤＳに大きな字幕画像データが格納される場合が考えられる。ＩＧデータの場合には、ページ数やコマンド数、使用されるボタン数が多い場合のＩＣＳや、ＯＤＳに大きな画像データが格納される場合が考えられる。

【０１９９】

一方、ＣＤＢ１１０に対してデータサイズの小さなセグメントが溜め込まれる場合には、ＣＤＢ１１０に格納可能なセグメント数が多くなるので、対応するＰＴＳおよびＤＴＳの合計サイズが大きくなる。一例として、ＩＧデータにおいて、アニメーション表示が設定されたボタン画像データが多数用いられている場合、ボタン画像データが格納される個々のＯＤＳのデータサイズは小さいが、ＯＤＳが多くなり、それに伴い対応するＰＴＳおよびＤＴＳの数が多くなる。

【０２００】

また、ＩＣＳやＯＤＳについては、セグメントのサイズがＰＥＳパケットの上限サイズの６４ｋＢよりも大きい場合があり、このような場合、既に説明したように、１のセグメントが複数のＰＥＳパケットに跨って格納される。この場合、１のセグメントに対して複数のＤＴＳおよびＰＴＳが存在することになり、ＣＤＢ１１０の容量に基づくＤＴＳおよびＰＴＳの合計サイズの見積もりを困難にする要因となる。

【０２０１】

そこで、この発明の実施の一形態では、プレゼンテーショングラフィクスおよびインタラクティブグラフィクスそれぞれの場合について、ＰＴＳ／ＤＴＳバッファ２００に格納可能なＰＴＳおよびＤＴＳの数に上限を定義し、この上限に基づきＰＴＳ／ＤＴＳバッ

10

20

30

40

50

ァ 2 0 0 の最低限必要な容量を規定することができるようにした。

【 0 2 0 2 】

P T S / D T S バッファ 2 0 0 に格納される P T S および D T S の数は、C D B 1 1 0 に格納されるセグメントの数に対応する。したがって、P T S / D T S バッファ 2 0 0 に格納可能な P T S および D T S の数の上限は、C D B 1 1 0 に格納可能なセグメント数の上限に基づき決めることができる。また、C D B 1 1 0 にエレメンタリストリームの形式で溜め込まれた P G データまたは I G データのセグメントは、図 3 0 または図 3 1 を用いて説明した処理に従い、対応する D T S でストリームグラフィクスプロセッサ 1 1 1 でデコードされて D B 1 1 2 または C B 1 1 3 に格納され、対応する P T S で出力される。そのため、D B 1 1 2 または C B 1 1 3 からセグメントが出力されるまで、当該セグメントに対応する P T S を、P T S / D T S バッファ 2 0 0 で保持しておく必要がある。したがって、C D B 1 1 0 に格納可能なセグメント数の上限は、D B 1 1 2 および C B 1 1 3 に格納可能なセグメント数に基づき決めることができる。これは、換言すれば、P T S / D T S バッファ 2 0 0 に格納可能な P T S および D T S の数の上限は、D B 1 1 2 および C B 1 1 3 に格納可能なセグメント数に基づき決めることができるということもできる。

10

【 0 2 0 3 】

B D - R O M 規格においては、P G データおよび I G データのそれぞれについて、1 エポック内のセグメント数の上限が設けられる。これにより、P G データについては、1 エポック内で使用可能な字幕画像データの数などが制限される。また、I G データについては、1 エポック内で使用可能なページ数、コマンド数、ボタン画像データ数などに制限が設けられる。この、P G データおよび I G データについて設けられる 1 エポック内のセグメント数の上限に基づき、P G データおよび I G データそれぞれに対応するデコーダモデルにおいて、D B 1 1 2 および C B 1 1 3 が格納可能とすべきセグメント数を定めることができる。

20

【 0 2 0 4 】

なお、D B 1 1 2 および C B 1 1 3 は、上述したように、図 3 0 および図 3 1 のフローチャートを用いて既に説明したように、エポックスタートディスプレイセットによりクリアされる。

【 0 2 0 5 】

P G データに関する、1 エポック内のセグメント数の上限は、図 3 4 A に一例が示されるように、P C S が 8 個、W D S が 1 個、P D S が 8 個、O D S が 6 4 個、E N D セグメントが 8 個となっている。これらのうち、E N D セグメントは、D T S を持たないと共に、D B 1 1 2 および C B 1 1 3 に格納されないため、P T S / D T S バッファ 2 0 0 の最低限必要な容量の算出対象から除外する。また、P D S は、D T S を持たないが、C B 1 1 3 に保持される実装も許されるため、P T S / D T S バッファ 2 0 0 の最低限必要な容量の算出対象とする。したがって、P C S、W D S、P D S および O D S のセグメント数の上限値に基づき、P G データに対応するデコーダモデルにおける P T S / D T S バッファ 2 0 0 の最低限必要な容量を見積もることができる。

30

【 0 2 0 6 】

一例として、上述の図 3 3 に示される全項目をセグメント毎に保持とした場合、(8 個 + 1 個 + 8 個 + 6 4 個) × 1 2 バイト = 9 7 2 バイトとなり、P G データに対応するデコーダモデルにおける P T S / D T S バッファ 2 0 0 の最低限必要な容量を、9 7 2 バイトと見積もることができる。実際には、この最低限の容量に対してある程度余裕を持たせ、例えば P G データの 1 エポック内のセグメント数の上限を 1 2 8 個程度として、P T S / D T S バッファ 2 0 0 の最低限必要な容量を 1 . 6 k B 程度とすることが考えられる。

40

【 0 2 0 7 】

I G データに関する、1 エポック内のセグメント数の上限は、図 3 4 B に一例が示されるように、I C S が 1 個、P D S が 2 5 6 個、O D S が 4 0 9 6 個、E N D セグメントが 1 個となっている。これらのうち、E N D セグメントは、D T S を持たないと共に、D B

50

1 1 2 および C B 1 1 3 に格納されないため、P T S / D T S バッファ 2 0 0 の最低限必要な容量の算出対象から除外する。また、P D S は、D T S を持たないが、C B 1 1 3 に保持される実装も許されるため、P T S / D T S バッファ 2 0 0 の最低限必要な容量の算出対象とする。したがって、I C S、P D S および O D S のセグメント数の上限値に基づき、I G データに対応するデコーダモデルにおける P T S / D T S バッファ 2 0 0 の最低限必要な容量を見積もることができる。

【 0 2 0 8 】

一例として、上述の図 3 3 に示される全項目をセグメント毎に保持するとした場合、(1 個 + 2 5 6 個 + 4 0 9 6 個) × 1 2 バイト = 5 2 2 3 6 バイトとなり、I G データに対応するデコーダモデルにおける P T S / D T S バッファ 2 0 0 の最低限必要な容量を、5 2 2 3 6 バイトと見積もることができる。実際には、この最低限の容量に対してある程度余裕を持たせ、例えば I G データの 1 エポック内のセグメント数の上限を 5 1 2 0 個程度として、P T S / D T S バッファ 2 0 0 の最低限必要な容量を 6 4 k B 程度とすることが考えられる。

【 0 2 0 9 】

なお、図 3 4 A および図 3 4 B に示した P G データおよび I G データそれぞれについての、1 エポック内のセグメント数の上限値は、一例であって、この例に限定されるものではない。

【 0 2 1 0 】

P T S / D T S バッファ 2 0 0 は、エポック毎にクリアすることができる。例えば、図 3 0 および図 3 1 のフローチャートを用いて説明した、エポックスタートディスプレイセットにより D B 1 1 2 および C B 1 1 3 がクリアされるタイミングで、P T S / D T S バッファ 2 0 0 をクリアすることができる。P T S / D T S バッファ 2 0 0 をエポック毎にクリアすることで、データをクリアするタイミングが固定的となり、モデルがシンプルになるメリットがある。

【 0 2 1 1 】

また、これに限らず、セグメント毎に P T S / D T S バッファ 2 0 0 を管理することもできる。例えば、P T S / D T S バッファ 2 0 0 に保持されている P T S や、P T S と D T S との組み合わせのうち、再生時刻が当該 P T S を過ぎたデータをセグメント単位で消去することが可能である。P T S により P T S / D T S バッファ 2 0 0 のクリアを行うことで、P T S / D T S バッファ 2 0 0 に対して不要なデータを保持しておく必要が無いというメリットがある。

【 0 2 1 2 】

次に、上述した実施の一形態に適用可能な再生装置について説明する。図 3 5 は、この発明の実施の一形態に適用可能な再生装置 1 の一例の構成を示す。再生装置 1 は、ストレージドライブ 5 0、スイッチ回路 5 1、A V デコーダ部 5 2 およびコントローラ部 5 3 を有する。ストレージドライブ 5 0 は、例えば、上述した B D - R O M を装填して再生可能とされているものとする。

【 0 2 1 3 】

コントローラ部 5 3 は、例えば C P U (Central Processing Unit) と、C P U 上で動作されるプログラムが予め格納された R O M (Read Only Memory)、C P U によりプログラムの実行時にワークメモリとして用いられる R A M (Random Access Memory) などからなり、この再生装置 1 の全体的な動作を制御する。

【 0 2 1 4 】

また、図示は省略するが、再生装置 1 には、ユーザに対して所定の制御情報を提供すると共に、ユーザオペレーションに応じた制御信号を出力するようにされたユーザインターフェイスが設けられる。例えば、赤外線通信など所定の無線通信手段を介して再生装置 1 と遠隔的に通信を行うリモートコントロールコマンドがユーザインターフェイスとして用いられる。リモートコントロールコマンド上には、上下左右方向をそれぞれ指示可能な十字キーといった方向キーや、数字キー、各種機能が予め割り当てられた機能キーなど、複

10

20

30

40

50

数の入力手段が設けられる。

【 0 2 1 5 】

リモートコントロールコマンドは、これらの入力手段に対してなされた操作に応じた制御信号を生成し、生成された制御信号を例えば赤外線信号に変調して送信する。再生装置 1 は、図示されない赤外線受信部においてこの赤外線信号を受信し、赤外線信号を電気信号に変換して復調し、元の制御信号を復元する。この制御信号がコントローラ部 5 3 に供給される。コントローラ部 5 3 は、この制御信号に応じて、プログラムに従い、再生装置 1 の動作を制御する。

【 0 2 1 6 】

ユーザインターフェイスは、リモートコントロールコマンドに限らず、例えば再生装置 1 の操作パネルに設けられたスイッチ群により構成することができる。また、再生装置 1 に、L A N (Local Area Network)などを介して通信を行う通信手段を設け、この通信手段を介して外部のコンピュータ装置などから供給された信号を、ユーザインターフェイスによる制御信号としてコントローラ部 5 3 に供給することも可能である。

【 0 2 1 7 】

また、当該再生装置 1 の言語設定の初期情報が、再生装置 1 が有する不揮発性メモリなどに記憶される。この言語設定の初期情報は、例えば再生装置 1 の電源投入時などにメモリから読み出され、コントローラ部 5 3 に供給される。

【 0 2 1 8 】

ストレージドライブ 5 0 にディスクが装填されると、コントローラ部 5 3 は、ストレージドライブ 5 0 を介してディスク上のファイル"index.bdmv"やファイル"MovieObject.bdmv"を読み出し、読み出されたファイルの記述に基づきディレクトリ"PLAYLIST"内のプレイリストファイルを読み出す。コントローラ部 5 3 は、プレイリストファイルに含まれるプレイアイテムが参照するクリップ A V ストリームを、ディスクからストレージドライブ 5 0 を介して読み出す。また、コントローラ部 5 3 は、プレイリストがサブプレイアイテムを含む場合は、サブプレイアイテムにより参照されるクリップ A V ストリームやサブタイトルデータも、ディスクからストレージドライブ 5 0 を介して読み出す。

【 0 2 1 9 】

なお、以下では、サブプレイアイテムに対応するクリップ A V ストリームを、サブクリップ A V ストリームと呼び、サブプレイアイテムに対する主たるプレイアイテムに対応するクリップ A V ストリームを、メインクリップ A V ストリームと呼ぶ。

【 0 2 2 0 】

ストレージドライブ 5 0 から出力されたデータは、図示されない復調部およびエラー訂正部により復調処理およびエラー訂正処理を所定に施され、多重化ストリームが復元される。ここでの多重化ストリームは、P I D によりデータの種類や並び順などが識別され、所定サイズに分割され時分割多重されたトランスポートストリームである。この多重化ストリームは、スイッチ回路 5 1 に供給される。コントローラ部 5 3 は、例えば P I D に基づきスイッチ回路 5 1 を所定に制御してデータを種類毎に分類し、メインクリップ A V ストリームのパケットをバッファ 6 0 に供給し、サブクリップ A V ストリームのパケットをバッファ 6 1 に供給し、サウンドデータの packets をバッファ 6 2 に供給し、テキストデータの packets をバッファ 6 3 に供給する。

【 0 2 2 1 】

バッファ 6 0 に溜め込まれたメインクリップ A V ストリームの packets は、コントローラ部 5 3 の制御に基づきバッファ 6 0 から packets 毎に読み出され、P I D フィルタ 6 4 に供給される。P I D フィルタ 6 4 は、供給された packets の P I D に基づき、packets を、ビデオストリームによる packets、プレゼンテーショングラフィクスストリーム（以下、P G ストリーム）による packets、インタラクティブグラフィクスストリーム（以下、I G ストリーム）による packets およびオーディオストリームによる packets に振り分ける。

【 0 2 2 2 】

10

20

30

40

50

一方、バッファ 61 に溜め込まれたサブクリップ AV ストリームのパケットは、コントローラ部 53 の制御に基づきバッファ 61 からパケット毎に読み出され、PID フィルタ 90 に供給される。PID フィルタ 90 は、供給されたパケットの PID に基づき、パケットを、ビデオストリームによるパケット、PG ストリームによるパケット、IG ストリームによるパケットおよびオーディオストリームによるパケットに振り分ける。

【0223】

PID フィルタ 64 により振り分けられたビデオストリームによるパケットと、PID フィルタ 90 により振り分けられたビデオストリームによるパケットは、それぞれ PID フィルタ 65 に供給され、PID に応じて振り分けられる。すなわち、PID フィルタ 65 は、PID フィルタ 64 から供給された、メインクリップ AV ストリームによるパケットを 1st ビデオデコーダ 69 に、PID フィルタ 90 から供給された、サブクリップ AV ストリームによるパケットを 2nd ビデオデコーダ 72 に、それぞれ供給するように、パケットを振り分ける。

10

【0224】

1st ビデオデコーダ 69 は、供給されたパケットのペイロードからビデオストリームを所定に取り出し、取り出されたビデオストリームの MPEG2 方式による圧縮符号を復号化する。1st ビデオデコーダ 69 の出力は、1st ビデオプレーン生成部 70 に供給され、ビデオプレーンが生成される。ビデオプレーンの生成は、例えばベースバンドのデジタルビデオデータの 1 フレームがフレームメモリに書き込まれることで生成される。1st ビデオプレーン生成部 70 で生成されたビデオプレーンは、ビデオデータ処理部 71 に供給される。

20

【0225】

2nd ビデオデコーダ 72 および 2nd ビデオプレーン生成部 73 において、上述の 1st ビデオデコーダ 69 および 1st ビデオプレーン生成部 70 と略同様の処理がなされ、ビデオストリームが復号化されビデオプレーンが生成される。2nd ビデオプレーン生成部 73 で生成されたビデオプレーンは、ビデオデータ処理部 71 に供給される。

【0226】

ビデオデータ処理部 71 は、1st ビデオプレーン生成部 70 で生成されたビデオプレーンと、2nd ビデオプレーン生成部 73 で生成されたビデオプレーンとを、1つのフレームに例えば所定に嵌め込んで 1 枚のビデオプレーンを生成することができる。1st ビデオプレーン生成部 70 で生成されたビデオプレーンと、2nd ビデオプレーン生成部 73 で生成されたビデオプレーンとを、選択的に用いてビデオプレーンを生成してもよい。このビデオプレーンは、例えば上述の図 9 に例示した動画プレーン 10 に対応する。

30

【0227】

PID フィルタ 64 により振り分けられた PG ストリームによるパケットと、PID フィルタ 90 により振り分けられた PG ストリームによるパケットは、それぞれスイッチ回路 66 に供給されて一方が所定に選択され、プレゼンテーショングラフィクスデコーダ 74 に供給される。

【0228】

プレゼンテーショングラフィクスデコーダ 74 (図 35 中では PG デコーダ 74 と表記) は、図 32 を用いて説明したデコーダモデルにおけるグラフィクスデコーダ 102 に対応する。このとき、図 32 に示される PID フィルタ 100 および TB 101 は、例えばスイッチ回路 64 および図示されないバッファに対応する。これに限らず、PG デコーダに PID フィルタ 100 および TB 101 を含んでもよい。

40

【0229】

図 35 の例では、PTS/DTS バッファ 200 は、このプレゼンテーショングラフィクスデコーダ 74 に含まれる。これに限らず、PTS/DTS バッファ 200 を、コントローラ 53 が有する RAM 上の所定領域とすることもできる。この場合、コントローラ 53 は、上述したようにして規定される、PG データについて、PTS/DTS バッファ 200 に最低限必要な容量を、RAM 上に確保する。

50

【 0 2 3 0 】

プレゼンテーショングラフィクスデコーダ 7 4 は、供給されたパケットのペイロードから P G ストリームを所定に取り出してデコードして字幕を表示するグラフィクスデータを生成し、スイッチ回路 7 5 に供給する。すなわち、プレゼンテーショングラフィクスデコーダ 7 4 は、供給されたパケットのペイロードからデータを取り出して、P I D の情報に基づき P E S パケットを再構築する。そして、再構築された P E S パケットのヘッダに D T S および / または P T S が存在すれば、それらを取り出して P T S / D T S バッファ 2 0 0 に保持すると共に、ペイロードのデータを取り出してデコードし、P C S、W D S、P D S および O D S を形成する。これら P C S、W D S、P D S および O D S は、対応する D T S に示されるタイミングで D B 1 1 2 または C B 1 1 3 に格納され、対応する P T S に示されるタイミングでプレゼンテーショングラフィクスデコーダ 7 4 から出力され、スイッチ回路 7 5 に供給される。

10

【 0 2 3 1 】

スイッチ回路 7 5 は、このグラフィクスデータと、後述するテキストデータによる字幕データとを所定に選択し、プレゼンテーショングラフィクスプレーン生成部 7 6 に供給する。プレゼンテーショングラフィクスプレーン生成部 7 6 は、供給されたデータに基づきプレゼンテーショングラフィクスプレーンを生成し、ビデオデータ処理部 7 1 に供給する。このプレゼンテーショングラフィクスプレーンは、例えば上述の図 9 に例示した字幕プレーン 1 1 に対応し、図 2 4 および図 2 8 で説明したグラフィクスプレーン 1 0 3 に格納される字幕画像イメージに対応する。

20

【 0 2 3 2 】

P I D フィルタ 6 4 により振り分けられた I G ストリームによるパケットと、P I D フィルタ 9 0 により振り分けられた I G ストリームによるパケットは、それぞれスイッチ回路 6 7 に供給されて一方が所定に選択され、インタラクティブグラフィクスデコーダ 7 7 (図 3 5 では I G デコーダ 7 7 と表記) に供給される。

【 0 2 3 3 】

インタラクティブグラフィクスデコーダ 7 7 は、図 3 2 を用いて説明したデコーダモデルにおけるグラフィクスデコーダ 1 0 2 に対応する。このとき、図 3 2 に示される P I D フィルタ 1 0 0 および T B 1 0 1 は、例えばスイッチ回路 6 7 および図示されないバッファに対応する。これに限らず、I G デコーダに P I D フィルタ 1 0 0 および T B 1 0 1 を含んでもよい。

30

【 0 2 3 4 】

図 3 5 の例では、P T S / D T S バッファ 2 0 0 は、このプレゼンテーショングラフィクスデコーダ 7 7 に含まれる。これに限らず、P T S / D T S バッファ 2 0 0 を、コントローラ 5 3 が有する R A M 上の所定領域とすることもできる。この場合、コントローラ 5 3 は、上述したようにして規定される、I G データについて、P T S / D T S バッファ 2 0 0 に最低限必要な容量を、R A M 上に確保する。

【 0 2 3 5 】

インタラクティブグラフィクスデコーダ 7 7 は、供給された I G ストリームによるパケットから I G ストリームの I C S、P D S および O D S を所定に取り出してデコードする。例えば、インタラクティブグラフィクスデコーダ 7 7 は、供給されたパケットのペイロードからデータを取り出して、P I D の情報に基づき P E S パケットを再構築する。そして、再構築された P E S パケットのヘッダに P T S および / または D T S が存在すれば、それらを取り出して P T S / D T S バッファ 2 0 0 に保持すると共に、ペイロードのデータを取り出してデコードし、I G ストリームの I C S、P D S および O D S を形成する。I C S および O D S は、P T S / D T S バッファ 2 0 0 に保持された、対応する D T S のタイミングでデコードされ、D B 1 1 2 に格納される。P D S は、即座にデコードされ、C B 1 1 3 に格納される。

40

【 0 2 3 6 】

インタラクティブグラフィクスデコーダ 7 7 において、P T S / D T S バッファ 2 0 0

50

に保持されたPTSのタイミングで、対応するセグメントがDB112またはCB113から所定に読み出され、プリロードバッファ78を介してインタラクティブグラフィクスプレーン生成部79に供給される。インタラクティブグラフィクスプレーン生成部79は、供給されたセグメントに基づき、インタラクティブグラフィクスプレーンを生成する。このインタラクティブグラフィクスプレーンは、例えば上述の図9に例示したインタラクティブグラフィクスプレーン12に対応する。

【0237】

ビデオデータ処理部71は、例えば図11を用いて説明したグラフィクス処理部を含み、供給されたビデオプレーン（図11における動画プレーン10）、プレゼンテーショングラフィクスプレーン（図11における字幕プレーン11）およびインタラクティブグラフィクスプレーン（図11におけるインタラクティブグラフィクスプレーン12）を、所定に合成して1枚の画像データとし、ビデオ信号にして出力する。

10

【0238】

例えば、図11を参照し、プレゼンテーショングラフィクスプレーン（すなわち字幕画像イメージ）に関して、パレットテーブル22A、RGB/YCbCr変換部22Bおよび乗算器23は、図32で説明したCLUT104に対応し、パレットテーブル22AおよびRGB/YCbCr変換部22Bで色情報が付加されると共に、乗算器23で透明属性を付加され、さらに、加算器24および加算器29で、ビデオプレーンおよびインタラクティブグラフィクスプレーンと合成され、1枚の画像データとされて出力される。

20

【0239】

PIDフィルタ64により振り分けられたオーディオストリームと、PIDフィルタ90により振り分けられたオーディオストリームは、それぞれスイッチ回路68に供給される。スイッチ回路68は、供給された2つのオーディオストリームのうち一方を1stオーディオデコーダ80に、他方を2ndオーディオデコーダ81に供給するように、所定に選択する。1stオーディオデコーダ80および2ndオーディオデコーダ81でそれぞれデコードされたオーディオストリームは、加算器82で合成され、さらに加算器83で、バッファ62から読み出されたサウンドデータと合成され、出力される。

【0240】

バッファ63から読み出されたテキストデータは、Text-STコンポジション部で所定に処理され、スイッチ回路75に供給される。

30

【0241】

上述では、再生装置1の各部がハードウェアで構成されるように説明したが、これはこの例に限られない。例えば、再生装置1をソフトウェア上の処理として実現することも可能である。この場合、再生装置1をコンピュータ装置上で動作させることができる。また、再生装置1をハードウェアおよびソフトウェアが混合された構成で実現することもできる。例えば、再生装置1における各デコーダ、特に、1stビデオデコーダ69、2ndビデオデコーダ72など処理の負荷が他と比べて大きな部分をハードウェアで構成し、その他をソフトウェアで構成することが考えられる。この場合、PTS/DTSバッファ200は、コンピュータ装置のメモリ上の所定領域として構成される。

【0242】

40

また、上述では、ビデオデータ、オーディオデータ、プレゼンテーショングラフィクスストリームおよびインタラクティブグラフィクスストリームを、光ディスクなどのディスク状記録媒体から読み出す構成として説明したが、これはこの例に限定されない。例えば、これらのデータの一部または全部を再生装置が有するハードディスクドライブや半導体メモリといった、光ディスク以外の記録媒体に保持し、当該記録媒体からこれらのデータを読み出すことも考えられる。

【0243】

再生装置1をソフトウェアのみ、または、ハードウェアおよびソフトウェアの混合により構成し、コンピュータ装置で実行させるためのプログラムは、例えばCD-ROM(Compact Disc-Read Only Memory)やDVD-ROM(Digital Versatile Disc Read Only Mem

50

ory)といった記録媒体に記録されて提供される。この記録媒体をコンピュータ装置のドライブに装填し、記録媒体上に記録されたプログラムを所定にコンピュータ装置にインストールすることで、上述の処理をコンピュータ装置上で実行可能な状態とすることができる。プログラムをBD-ROMに記録することも考えられる。なお、コンピュータ装置の構成は、極めて周知であるため、説明は省略する。

【図面の簡単な説明】

【0244】

【図1】BD-ROMのデータモデルを概略的に示す。

【図2】インデックステーブルを説明するための略線図である。

【図3】クリップAVストリーム、クリップ情報、クリップ、プレイアイテムおよびプレイリストの関係を示すUML図である。

10

【図4】複数のプレイリストから同一のクリップを参照する方法を説明するための略線図である。

【図5】サブパスについて説明するための略線図である。

【図6】記録媒体に記録されるファイルの管理構造を説明するための略線図である。

【図7】BD仮想プレーヤの動作を概略的に示す略線図である。

【図8】BD仮想プレーヤの動作を概略的に示す略線図である。

【図9】この発明の実施の形態で画像の表示系として用いられるプレーン構造の一例を示す略線図である。

【図10】動画プレーン、字幕プレーンおよびグラフィクスプレーンの一例の解像度および表示可能色を示す略線図である。

20

【図11】動画プレーン、字幕プレーンおよびグラフィクスプレーンを合成する一例の構成を示すブロック図である。

【図12】パレットの入出力データの一例を示す略線図である。

【図13】パレットに格納される一例のパレットテーブルを示す略線図である。

【図14】プレゼンテーショングラフィクスストリームについて説明するための略線図である。

【図15】ディスプレイセットの構成を概略的に示す略線図である。

【図16】プレゼンテーショングラフィクスのディスプレイセットに関する論理的な構成を概略的に示す略線図である。

30

【図17】インタラクティブグラフィクスのディスプレイセットに関する論理的な構成を概略的に示す略線図である。

【図18】ディスプレイセットの一例の格納形式を示す略線図である。

【図19】PCSの一例の構造を表すシンタクスを示す略線図である。

【図20】ブロックcomposition_object()の一例の構造を表すシンタクスを示す略線図である。

【図21】WDSの一例の構造を表すシンタクスを示す略線図である。

【図22】ブロックwindow()の一例の構造を表すシンタクスを示す略線図である。

【図23】メニュー画面およびボタンの構成を概略的に示す略線図である。

【図24】ICSのヘッダ情報の一例の構造を表すシンタクスを示す略線図である。

40

【図25】ブロックinteractive_composition_data_fragemnt()の一例の構造を表すシンタクスを示す略線図である。

【図26】ブロックpage()の一例の構造を表すシンタクスを示す略線図である。

【図27】ブロックbutton_overlap_group()の一例の構造を表すシンタクスを示す略線図である。

【図28】ブロックbutton()の一例の構造を表すシンタクスを示す略線図である。

【図29】現時点のBD-ROM規格に定義されるプレゼンテーショングラフィクスデータおよびインタラクティブグラフィクスデータのデコーダモデルの一例の構成を示すブロック図である。

【図30】プレゼンテーショングラフィクスデコーダにおけるディスプレイセット単位で

50

の一例の処理を示すフローチャートである。

【図 3 1】インタラクティブグラフィクスデコーダにおけるディスプレイセット単位での一例の処理を示すフローチャートである。

【図 3 2】この発明によるプレゼンテーショングラフィクスデータおよびインタラクティブグラフィクスデータのデコーダモデルの一例の構成を示すブロック図である。

【図 3 3】PTS / DTS バッファに保持されるデータの一例の構成を示す略線図である。

【図 3 4】プレゼンテーショングラフィクスおよびインタラクティブグラフィクスについて、1 エポック内の各セグメント数の上限の例をそれぞれ示す略線図である。

【図 3 5】発明の実施の一形態に適用可能な再生装置の一例の構成を示すブロック図である。

10

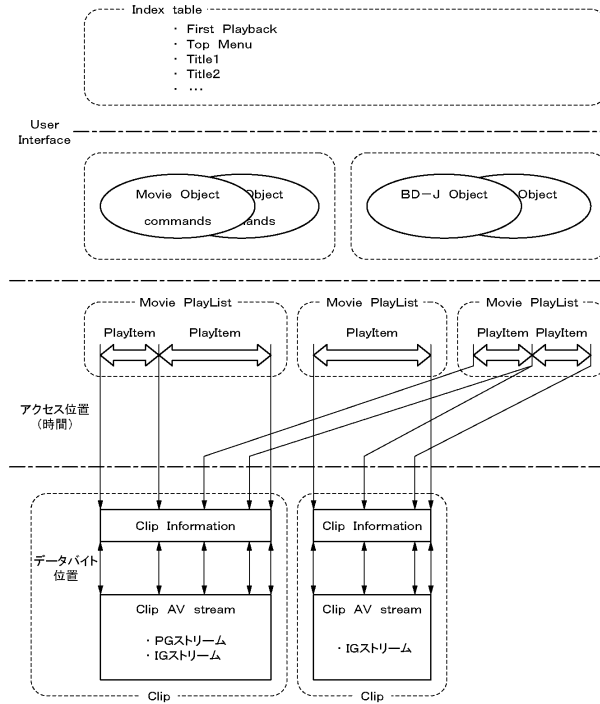
【符号の説明】

【 0 2 4 5 】

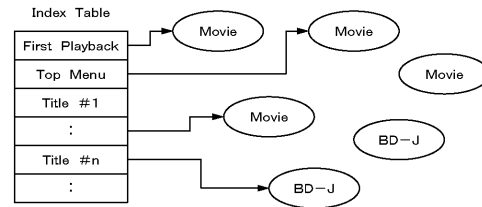
- 1 再生装置
- 5 3 コントローラ
- 7 4 プレゼンテーショングラフィクスデコーダ
- 7 6 プレゼンテーショングラフィクスプレーン生成部
- 7 7 インタラクティブグラフィクスデコーダ
- 7 9 インタラクティブグラフィクスプレーン生成部
- 1 0 0 PID フィルタ
- 1 0 1 トランスポートバッファ
- 1 0 2 PG / IG デコーダ
- 1 1 0 コーデッドデータバッファ
- 1 1 1 ストリームグラフィクスプロセッサ
- 1 1 2 デコーデッドオブジェクトバッファ
- 1 1 3 コンポジションバッファ
- 1 1 4 グラフィクスコントローラ
- 2 0 0 PTS / DTS バッファ

20

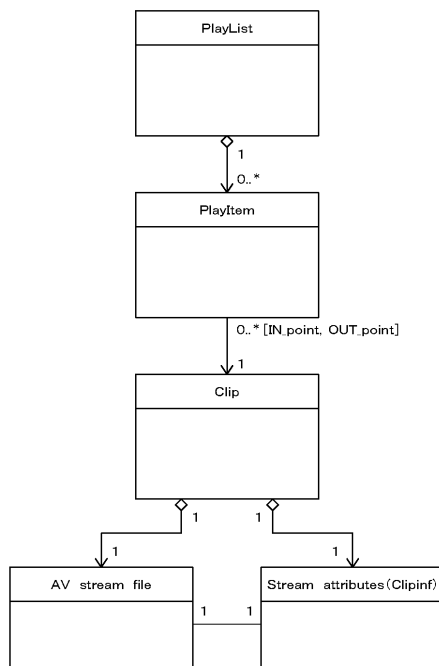
【図 1】



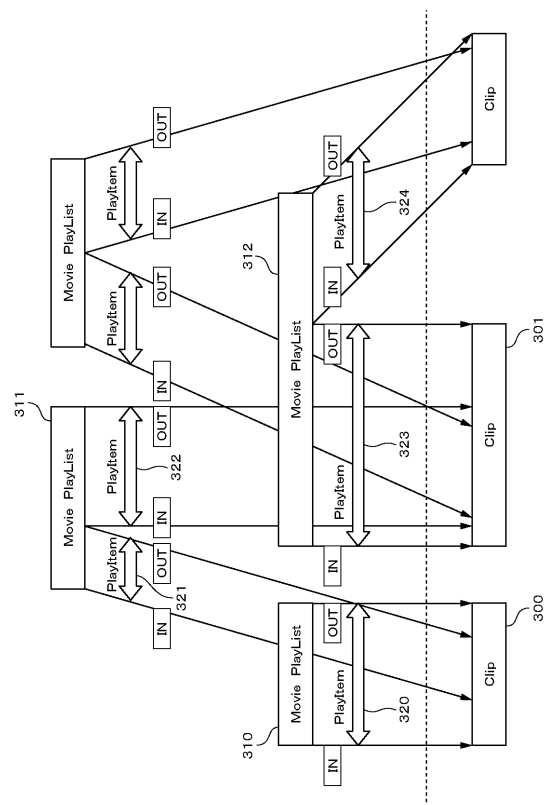
【図 2】



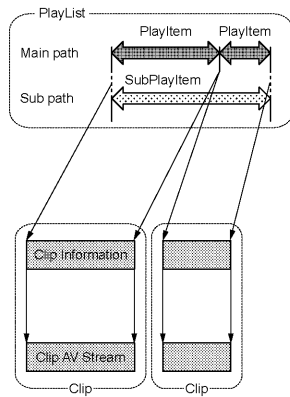
【図 3】



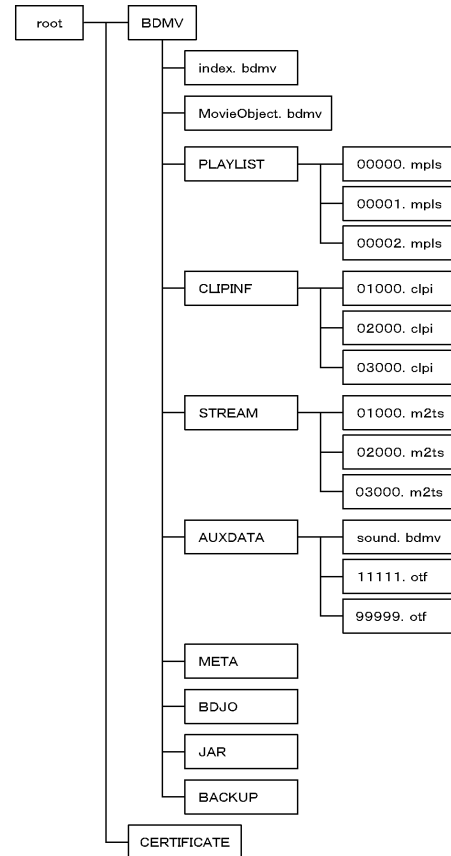
【図 4】



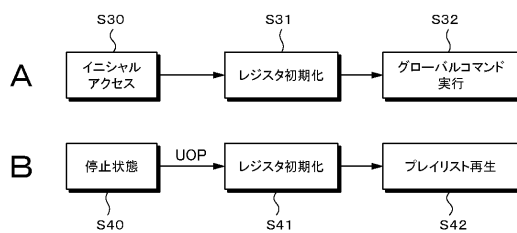
【図 5】



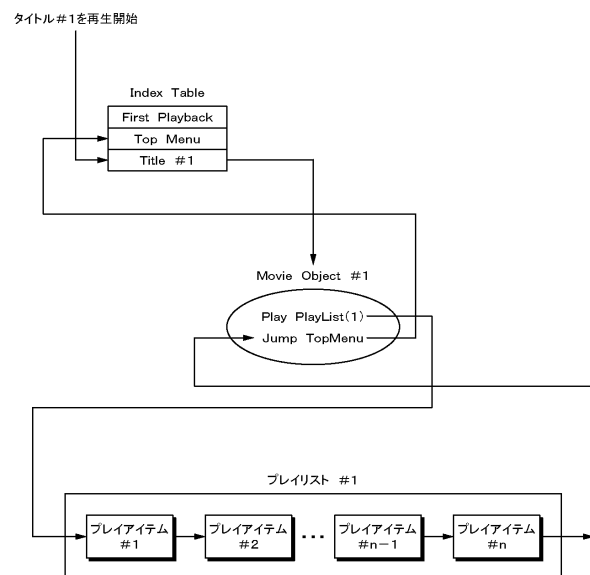
【図 6】



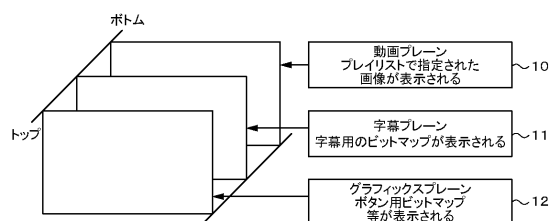
【図 7】



【図 8】



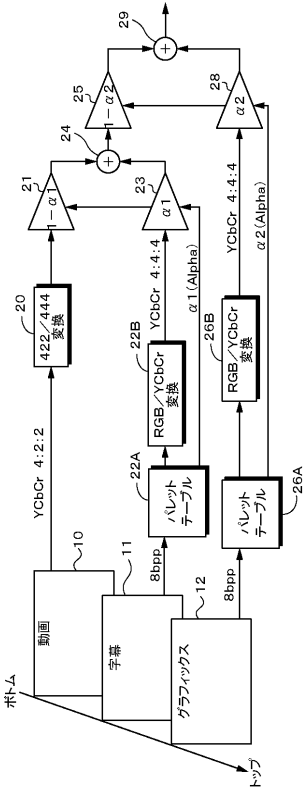
【図 9】



【図 1 0】

項目	規定内容
動画プレーン	1920×1080×16ビット,YCbCr(4:2:2),各8ビット
字幕プレーン	1920×1080×8ビット,8ビットカラーマップアドレス(ハレット)+256段階のアルファブレンディング
グラフィックスプレーン	1920×1080×8ビット, 8ビットカラーマップアドレス(ハレット)+256段階のアルファブレンディング

【図 1 1】



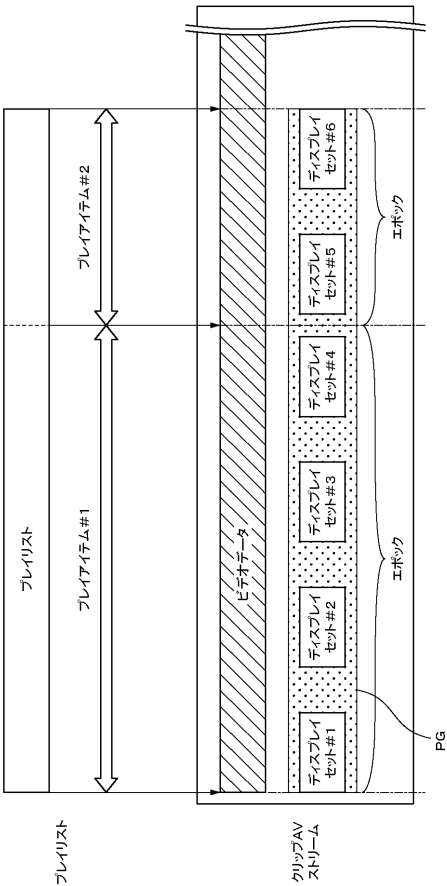
【図 1 2】

入力	入力アドレス 8ビット
出力	出力データ 8ビット×4、(R、G、B、α)出力

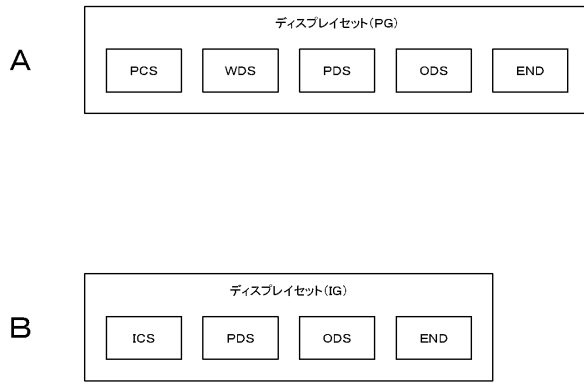
【図 1 3】

カラーインデックス値	3原色の値			不透明度
	R	G	B	α
0x00	0	0	0	0
0x01	10	100	30	0.5
⋮	⋮	⋮	⋮	
⋮	⋮	⋮	⋮	
0xFF	200	255	100	0.8

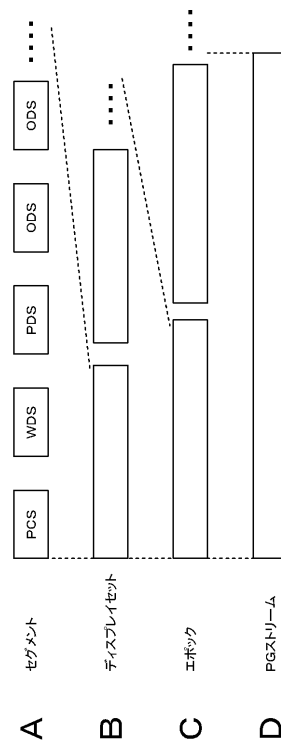
【図 1 4】



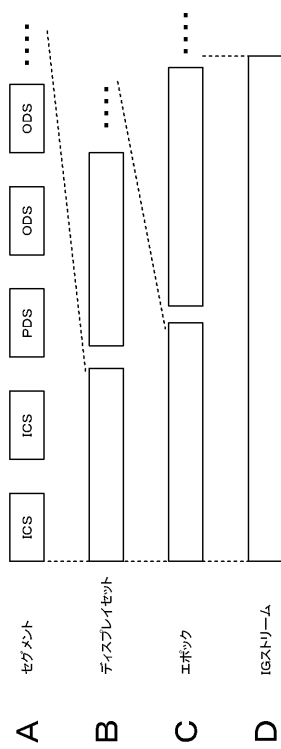
【図 15】



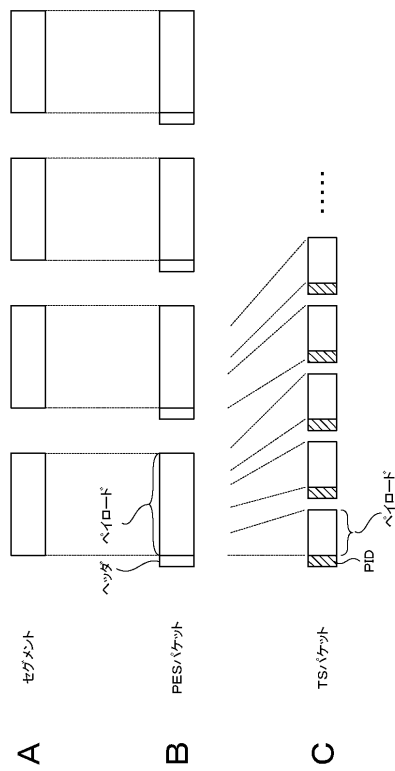
【図 16】



【図 17】



【図 18】



【図 19】

シンタクス	データ長 (ビット)	ニーモニック
presentation_composition_segment() {		
segment_descriptor()		
video_descriptor()		
composition_descriptor()		
palette_update_flag	1	bslbf
reserved	7	bslbf
palette_id_ref	8	uimbsf
number_of_composition_objects	8	uimbsf
for (i=0; i<number_of_composition_objects; i++) {		
composition_object()		
}		
}		

【図 20】

シンタクス	データ長 (ビット)	ニーモニック
composition_object() {		
object_id_ref	16	uimbsf
window_id_ref	8	uimbsf
object_cropped_flag	1	bslbf
forced_on_flag	1	bslbf
Reserved	6	bslbf
composition_object_horizontal_position	16	uimbsf
composition_object_vertical_position	16	uimbsf
if (object_cropped_flag==1) {		
cropping_rectangle() {		
object_cropping_horizontal_position	16	uimbsf
object_cropping_vertical_position	16	uimbsf
object_cropping_width	16	uimbsf
object_cropping_height	16	uimbsf
}		
}		
}		

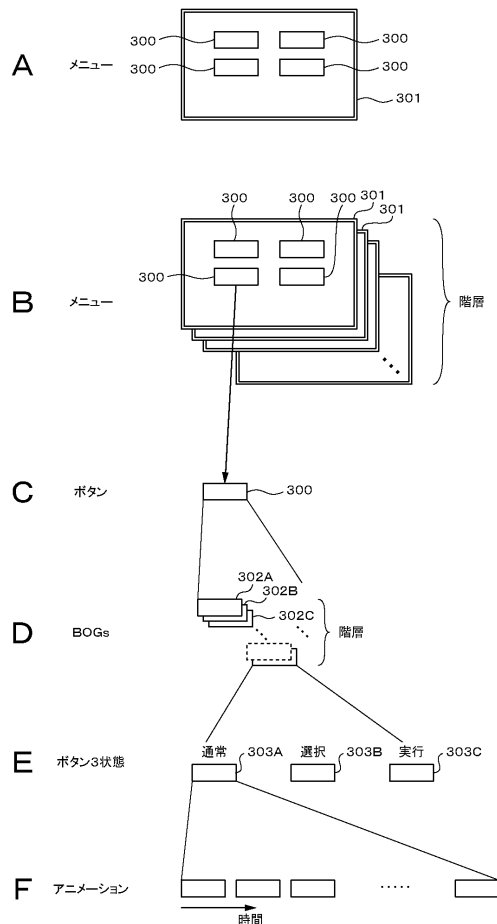
【図 21】

シンタクス	データ長 (ビット)	ニーモニック
window_definition_segment() {		
segment_descriptor()		
number_of_windows	8	uimbsf
for (i=0; i<number_of_windows; i++) {		
window()		
}		
}		

【図 22】

シンタクス	データ長 (ビット)	ニーモニック
window() {		
window_id	8	uimbsf
window_horizontal_position	16	uimbsf
window_vertical_position	16	uimbsf
window_width	16	uimbsf
window_height	16	uimbsf
}		

【図 23】



【図 24】

シンタクス	データ長 (ビット)	ニーモニック
interactive_composition_segment() {		
segment_descriptor()		
video_descriptor()		
composition_descriptor()		
sequence_descriptor()		
interactive_composition_data_fragment()		
}		

【図 25】

シンタクス	データ長 (ビット)	ニーモニック
interactive_composition() {		
interactive_composition_length	24	uimbsf
stream_model	1	bslbf
user_interface_model	1	bslbf
Reserved	6	bslbf
If (stream_model == '0b') {		
Reserved	7	bslbf
composition_time_out_pts	33	uimbsf
Reserved	7	bslbf
selection_time_out_pts	33	uimbsf
}		
user_time_out_duration	24	uimbsf
number_of_pages	8	uimbsf
for (i=0; i<number_of_pages; i++) {		
page()		
}		
}		

【図 26】

シンタクス	データ長 (ビット)	ニーモニック
page() {		
page_id	8	uimsbf
page_version_number	8	uimsbf
UO_mask_table()		
in_effects() {		
effect_sequence()		
}		
out_effects() {		
effect_sequence()		
}		
animation_frame_rate_code	8	uimsbf
default_selected_button_id_ref	16	uimsbf
default_activated_button_id_ref	16	uimsbf
palette_id_ref	8	uimsbf
number_of_BOGs	8	uimsbf
for (i=0; i<number_of_BOGs; i++) {		
button_overlap_group()		
}		
}		

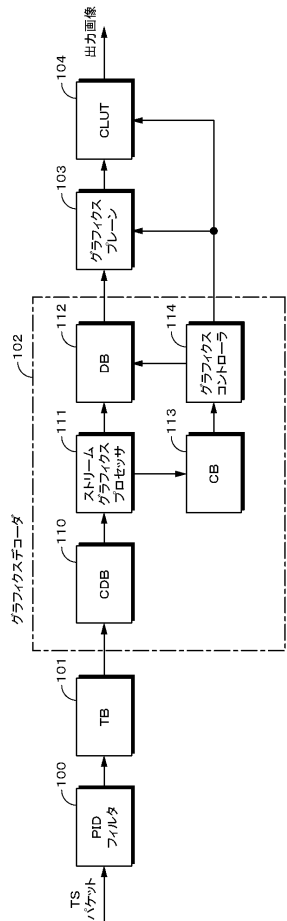
【図 27】

シンタクス	データ長 (ビット)	ニーモニック
button_overlap_group() {		
default_valid_button_id_ref	16	uimsbf
number_of_buttons	8	uimsbf
for (i=0; i<number_of_buttons; i++) {		
button()		
}		
}		

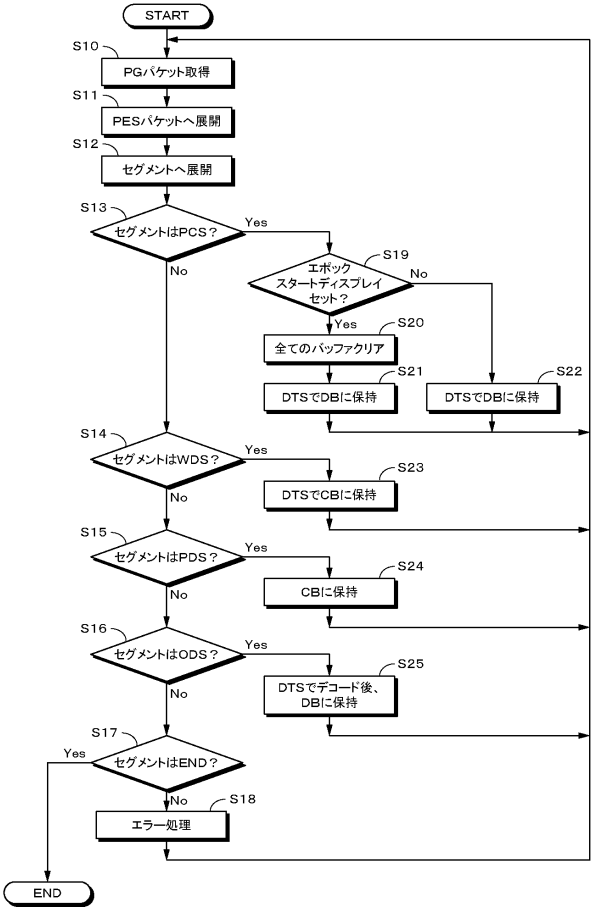
【図 28】

シンタクス	データ長 (ビット)	ニーモニック
button() {		
button_id	16	uimsbf
button_numeric_select_value	16	uimsbf
auto_action_flag	1	bslbf
reserved	7	bslbf
button_horizontal_position	16	uimsbf
button_vertical_position	16	uimsbf
neighbor_info() {		
upper_button_id_ref	16	uimsbf
lower_button_id_ref	16	uimsbf
left_button_id_ref	16	uimsbf
right_button_id_ref	16	uimsbf
}		
normal_state_info() {		
normal_start_object_id_ref	16	uimsbf
normal_end_object_id_ref	16	uimsbf
normal_repeat_flag	1	bslbf
normal_complete_flag	1	bslbf
reserved	6	bslbf
}		
selected_state_info() {		
selected_state_sound_id_ref	8	uimsbf
selected_start_object_id_ref	16	uimsbf
selected_end_object_id_ref	16	uimsbf
selected_repeat_flag	1	bslbf
selected_complete_flag	1	bslbf
reserved	6	bslbf
}		
activated_state_info() {		
activated_state_sound_id_ref	8	uimsbf
activated_start_object_id_ref	16	uimsbf
activated_end_object_id_ref	16	uimsbf
}		
number_of_navigation_commands	16	uimsbf
for (i=0; i<number_of_navigation_commands; i++) {		
navigation_command()		
}		
}		

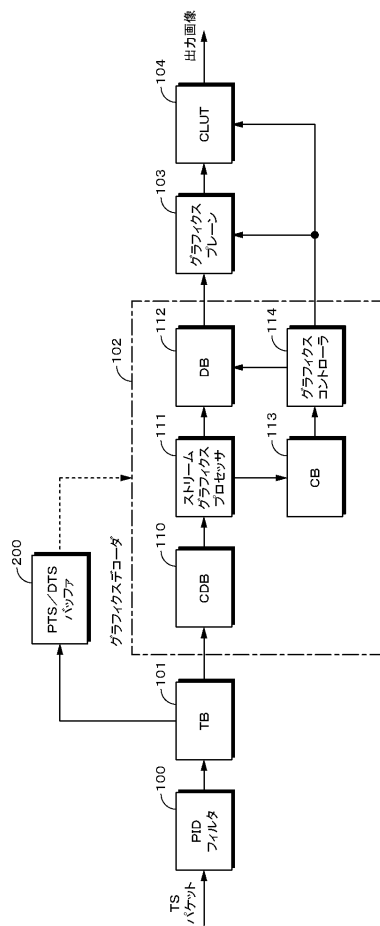
【図 29】



【図 30】

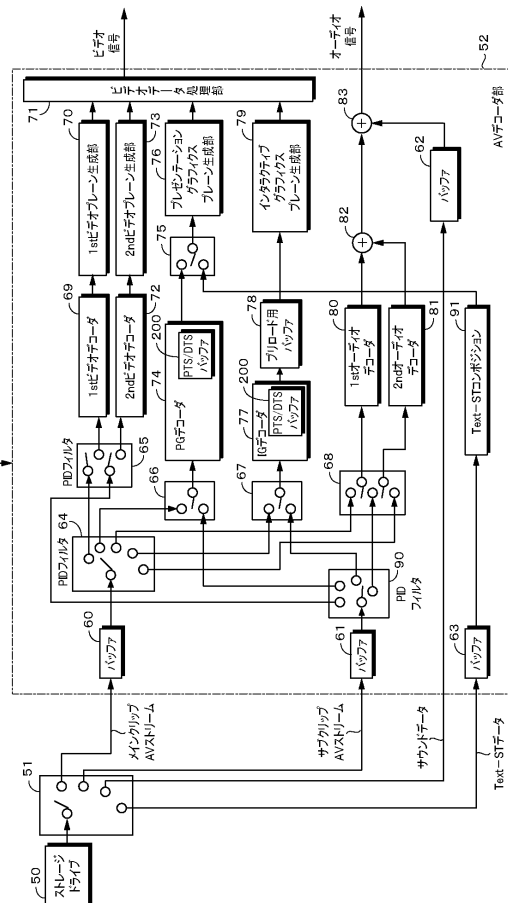


【 図 3 2 】



【 図 3 5 】

PCS, ICS, PDS, WDS, ODS



A

PG	
セグメント	上限値(個)
PCS	8
WDS	1
PDS	8
ODS	64
END	8

B

IG	
セグメント	上限値(個)
ICS	1
PDS	256
ODS	4096
END	1

フロントページの続き

(56)参考文献 特表2006-503520(JP,A)
国際公開第2005/079064(WO,A1)
特開2005-204315(JP,A)

(58)調査した分野(Int.Cl., DB名)
H04N 5/92
G11B 20/10
H04N 5/93