

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第4092804号
(P4092804)

(45) 発行日 平成20年5月28日 (2008.5.28)

(24) 登録日 平成20年3月14日 (2008.3.14)

(51) Int.Cl.

F I

G 1 1 B 27/00 (2006.01)

G 1 1 B 27/00

D

G 1 1 B 20/12 (2006.01)

G 1 1 B 20/12

請求項の数 14 (全 38 頁)

(21) 出願番号 特願平11-7946
 (22) 出願日 平成11年1月14日 (1999.1.14)
 (65) 公開番号 特開2000-113644 (P2000-113644A)
 (43) 公開日 平成12年4月21日 (2000.4.21)
 審査請求日 平成18年1月12日 (2006.1.12)
 (31) 優先権主張番号 特願平10-223388
 (32) 優先日 平成10年8月6日 (1998.8.6)
 (33) 優先権主張国 日本国 (JP)

(73) 特許権者 000002185
 ソニー株式会社
 東京都港区港南1丁目7番1号
 (74) 代理人 100086841
 弁理士 脇 篤夫
 (74) 代理人 100102635
 弁理士 浅見 保男
 (72) 発明者 穴戸 由紀夫
 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 宮下 誠

(56) 参考文献 特開平09-091880 (JP, A)

最終頁に続く

(54) 【発明の名称】 再生装置

(57) 【特許請求の範囲】

【請求項1】

主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、
 前記再生手段によって、前記記録媒体から読み出される管理情報を検出することができる管理情報検出手段と、
 前記管理情報検出手段によって検出された、前記管理情報を順次格納することができるようにされている管理情報記憶手段と、
 前記管理情報記憶手段に順次格納される前記管理情報に含まれる前記文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、
 前記ヘッダ情報検出手段によって検出される第一のヘッダ情報を起点とした順方向における所定の位置のデータ内容を検証する第一の検証手段と、
 前記第一のヘッダ情報に続いて検出される第二のヘッダ情報を起点とした逆方向における所定の位置のデータ内容を検証する第二の検証手段と、
 前記第一、第二の検証手段の検証結果に基づいて、前記第一、第二のヘッダ情報の間にある前記文字情報が有効であるか否かを判断することができる判断手段と、
 を備えたことを特徴とする再生装置。

【請求項2】

前記管理情報記憶手段は、前記管理情報を格納することができる管理情報格納領域と、
 前記文字情報を格納することができる文字情報格納領域によって形成され、

前記文字情報格納領域には、前記管理情報格納領域に格納された前記管理情報のうち前記文字情報のみが格納されるようにしたことを特徴とする請求項 1 に記載の再生装置。

【請求項 3】

前記記録媒体上の前記管理情報はインターリーブされたデータ形態とされているとともに、前記管理情報検出手段はデインターリーブされる前の管理情報を検出するようにされていることを特徴とする請求項 1 に記載の再生装置。

【請求項 4】

主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、

前記再生手段によって、前記記録媒体から読み出される管理情報を検出することができる管理情報検出手段と、

前記管理情報検出手段によって検出された前記管理情報を順次格納することができるようにされている管理情報記憶手段と、

前記管理情報記憶手段に順次格納される前記管理情報に含まれる文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、

前記ヘッダ情報検出手段によって検出される第一のヘッダ情報と、該第一のヘッダ情報に続いて検出される第二のヘッダ情報の間の所定の位置にあるデータ内容が、前記文字情報において規定されている所定のデータ内容に一致しているか否かを検証する検証手段と、前記検証手段の検証結果に基づいて、前記第一、第二のヘッダ情報の間にある前記文字情報が有効であると判断することができる判断手段と、

を備えたことを特徴とする再生装置。

【請求項 5】

前記管理情報記憶手段は、前記管理情報を格納することができる管理情報格納領域と、前記文字情報を格納することができる文字情報格納領域によって形成され、

前記文字情報格納領域には、前記管理情報格納領域に格納された前記管理情報のうち前記文字情報のみが格納されるようにしたことを特徴とする請求項 4 に記載の再生装置。

【請求項 6】

前記記録媒体上の前記管理情報はインターリーブされたデータ形態とされているとともに、前記管理情報検出手段はデインターリーブされる前の管理情報を検出するようにされていることを特徴とする請求項 4 に記載の再生装置。

【請求項 7】

主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、

前記再生手段によって、前記記録媒体から読み出される管理情報を検出することができる管理情報検出手段と、

前記管理情報検出手段によって検出された前記管理情報を順次格納することができるようにされている管理情報記憶手段と、

前記管理情報記憶手段に順次格納される前記管理情報に含まれる前記文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、

ヘッダ情報検出手段によって検出される前記ヘッダ情報の間にあるデータを一つの単位として複数単位のデータの内容を比較を行ない、比較したデータの内容が同等であった場合に、当該データが前記文字情報として有効であると判断することができる判断手段と、を備えたことを特徴とする再生装置。

【請求項 8】

前記管理情報記憶手段は、前記管理情報を格納することができる管理情報格納領域と、前記文字情報を格納することができる文字情報格納領域によって形成され、

前記文字情報格納領域には、前記管理情報格納領域に格納される前記管理情報のうち前記文字情報のみが格納されるようにされていることを特徴とする請求項 7 に記載の再生装置。

【請求項 9】

前記記録媒体上の前記管理情報はインターリーブされたデータ形態とされているとともに、前記管理情報検出手段はデインターリーブされる前の管理情報を検出するようにされていることを特徴とする請求項 7 に記載の再生装置。

【請求項 10】

主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、
前記再生手段によって、前記記録媒体から読み出される管理情報を検出することができる管理情報検出手段と、
前記管理情報検出手段によって検出された前記管理情報を順次格納することができるようにされている管理情報記憶手段と、
前記管理情報記憶手段に順次格納される前記管理情報に含まれる前記文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、
前記ヘッダ情報検出手段によって検出される所要のヘッダ情報に基づいて文字情報の検証処理を行なう検証手段と、
を備え、前記ヘッダ情報検出手段によって所要のヘッダ情報が 2 回検出された場合、前記再生手段による前記管理情報の読み出しを終了するとともに、前記管理情報検出手段による前記管理情報の検出を終了したうえで、前記検証手段は前記ヘッダ情報に基づいた前記文字情報の検証処理を行なうようにしたことを特徴とする再生装置。

10

【請求項 11】

前記管理情報記憶手段は、前記管理情報を格納することができる管理情報格納領域と、
前記文字情報を格納することができる文字情報格納領域によって形成され、
前記文字情報格納領域には、前記管理情報格納領域に格納される前記管理情報のうち前記文字情報のみが格納されるようにされていることを特徴とする請求項 10 に記載の再生装置。

20

【請求項 12】

前記記録媒体上の前記管理情報はインターリーブされたデータ形態とされているとともに、前記管理情報検出手段はデインターリーブされる前の管理情報を検出するようにされていることを特徴とする請求項 10 に記載の再生装置。

【請求項 13】

主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、
前記再生手段によって、前記記録媒体から読み出される管理情報を検出することができる管理情報検出手段と、
前記管理情報検出手段によって検出された前記管理情報を所定の単位毎に順次格納することができるようにされている管理情報記憶手段と、
前記文字情報に対する誤り検出符号に基づいて、前記管理情報記憶手段に格納される前記管理情報に含まれる前記文字情報の誤り検出を行う誤り検出手段と、前記管理情報記憶手段に前記所定の単位とされる前記管理情報が格納された場合に、格納された前記管理情報に含まれる前記文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、
前記誤り検出手段によって前記管理情報記憶手段に格納されている前記文字情報に誤りが検出された場合に、前記ヘッダ情報検出手段によって前記ヘッダ情報が検出されないまま、それまでに前記誤り検出手段によって検出された前記文字情報の誤り回数が前記管理情報の所定の単位数以上となったら、前記記録媒体に前記文字情報が記憶されていないと判別するようにした文字情報有無識別手段と、
を備えたことを特徴とする再生装置。

30

40

【請求項 14】

前記文字情報有無識別手段によって、前記記憶媒体に前記文字情報が記録されていないと判別した場合は、前記再生手段は前記記録媒体からの前記管理情報の読み出しを終了するようにしたことを特徴とする請求項 13 に記載の再生装置。

【発明の詳細な説明】

50

【 0 0 0 1 】

【 発明の属する技術分野 】

本発明は、C D - T E X T に記憶されている文字情報を再生することができる再生装置に関するものである。

【 0 0 0 2 】

【 従来の技術 】

例えばオーディオ用のC D (COMPACT DISC) のフォーマットに準拠した上で、そのサブコードデータとして所要の文字情報を記憶したC D - T E X T が知られている。記憶される文字情報としては、例えばディスクのタイトルやアーティスト名、楽曲名などの情報とされ、これらの情報を読み出して表示することによって、そのオーディオC D に収録されて

10

【 0 0 0 3 】

【 発明が解決しようとする課題 】

しかし、C D - T E X T は、E C C などのエラー訂正手段を有していないことが例えばC D - R O M などと異なっている。したがって、再生状況によっては、正規のデータが検出されず、これにより文字情報の始まりと終わりが正しく検出することができないという不具合が生じることがあった。

また、文字情報が前記リードイン領域に記憶されている場合は、ディスクのアドレスに基づいたリトライ処理を有効に行なうことができない。

即ち、従来では読み出されたデータにおける文字情報が確かなものか否かを判断することができなかった。

20

【 0 0 0 4 】

【 課題を解決するための手段 】

本発明はこのような問題点を解決するために、主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、前記再生手段によって、前記記録媒体から読み出される管理情報を検出することができる管理情報検出手段と、前記管理情報検出手段によって検出された、前記管理情報を順次格納することができるようにされている管理情報記憶手段と、前記管理情報記憶手段に順次格納される前記管理情報に含まれる前記文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、前記ヘッダ情報検出手段によって検出される第一のヘッダ情報を起点とした順方向における所定の位置のデータ内容を検証する第一の検証手段と、前記第一のヘッダ情報に続いて検出される第二のヘッダ情報を起点とした逆方向における所定の位置のデータ内容を検証する第二の検証手段と、前記第一、第二の検証手段の検証結果に基づいて、前記第一、第二のヘッダ情報の間にある前記文字情報が有効であるか否かを判断することができる判断手段を備えて再生装置を構成する。

30

【 0 0 0 5 】

また、主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、前記再生手段によって、前記記録媒体から読み出される管理情報を検出することができる管理情報検出手段と、前記管理情報検出手段によって検出された前記管理情報を順次格納することができるようにされている管理情報記憶手段と、前記管理情報記憶手段に順次格納される前記管理情報に含まれる文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、前記ヘッダ情報検出手段によって検出される第一のヘッダ情報と、該第一のヘッダ情報に続いて検出される第二のヘッダ情報の間の所定の位置にあるデータ内容が、前記文字情報において規定されている所定のデータ内容に一致しているか否かを検証する検証手段と、前記検証手段の検証結果に基づいて、前記第一、第二のヘッダ情報の間にある前記文字情報が有効であると判断することができる判断手段を備えて再生装置を構成する。

40

【 0 0 0 6 】

また、主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、前記再生手段によって、前記記録媒体から読み出さ

50

れる管理情報を検出することができる管理情報検出手段と、前記管理情報検出手段によって検出された前記管理情報を順次格納することができるようにされている管理情報記憶手段と、前記管理情報記憶手段に順次格納される前記管理情報に含まれる前記文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、ヘッダ情報検出手段によって検出される前記ヘッダ情報の間にあるデータを一つの単位として複数単位のデータの内容を比較を行ない、比較したデータの内容が同等であった場合に、当該データが前記文字情報として有効であると判断することができる判断手段を備えて再生装置を構成する。

【0007】

さらに、主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、前記再生手段によって、前記記録媒体から読み出される管理情報を検出することができる管理情報検出手段と、前記管理情報検出手段によって検出された前記管理情報を順次格納することができるようにされている管理情報記憶手段と、前記管理情報記憶手段に順次格納される前記管理情報に含まれる前記文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、前記ヘッダ情報検出手段によって検出される所要のヘッダ情報に基づいて文字情報の検証処理を行なう検証手段を備え、前記ヘッダ情報検出手段によって所要のヘッダ情報が2回検出された場合、前記再生手段による前記管理情報の読み出しを終了するとともに、前記管理情報検出手段による前記管理情報の検出を終了したうえで、前記検証手段は前記ヘッダ情報に基づいた前記文字情報の検証処理を行なうように再生装置を構成する。

【0008】

また、主データの再生管理情報と文字情報を含む管理情報が記録された記録媒体から情報を読み出すことができる再生手段と、前記再生手段によって、前記記録媒体から読み出される管理情報を検出することができる管理情報検出手段と、前記管理情報検出手段によって検出された前記管理情報を所定の単位毎に順次格納することができるようにされている管理情報記憶手段と、前記文字情報に対する誤り検出符号に基づいて、前記管理情報記憶手段に格納される前記管理情報に含まれる前記文字情報の誤り検出を行う誤り検出手段と、前記管理情報記憶手段に前記所定の単位とされる前記管理情報が格納された場合に、格納された前記管理情報に含まれる前記文字情報の所要のヘッダ情報を検出するヘッダ情報検出手段と、前記誤り検出手段によって前記管理情報記憶手段に格納されている前記文字情報に誤りが検出された場合に、前記ヘッダ情報検出手段によって前記ヘッダ情報が検出されないまま、それまでに前記誤り検出手段によって検出された前記文字情報の誤り回数が前記管理情報の所定の単位数以上となったら、前記記録媒体に前記文字情報が記憶されていないと判別するようにした文字情報有無識別手段を備えて再生装置を構成する。

【0009】

本発明によれば、記録媒体から読み込んだ管理情報から、この管理情報に含まれる文字情報を示すヘッダ情報に対応した各種データ内容を検証し、所定のデータ内容が検出され場合に、当該ヘッダ情報に対応した文字情報の有効性を確認することができるようにしている。

これにより、文字情報の誤検出を抑制し検出能力を向上させることができ、より正確な文字情報を得ることができるようになる。

【0010】

また、所定数のヘッダ情報が2回検出された時点で、管理情報の検出を終了したうえで文字情報の検証処理を行なうようにすることにより、それ以降行なう検証処理の処理負担を軽減することができるようになる。

【0011】

さらに、前記文字情報に対する誤り検出符号に基づいて、当該文字情報に誤りが検出されたときに、ヘッダ情報が検出されないまま、それまでに前記文字情報の誤りが所定回数以上検出されていた場合には、記録媒体に文字情報が記録されていないと判断するようにしているので、必要以上の再生動作を抑制したうえで、記録媒体の種別を識別することができるようになる。

【 0 0 1 2 】

【 発明の実施の形態 】

以下、本発明の実施の形態として、C D - T E X Tとしてのディスクに対応して、オーディオデータ及びサブコードとしての文字情報を再生することができる再生装置を例に挙げて次に示す順序で説明する。なお、サブコードとして記録される文字情報を「テキストデータ」ともいうこととする。

- < 1 . 再生装置の構成 >
- < 2 . T O C 及びサブコード >
- < 3 . テキストデータ >
- < 4 . 再生処理 >
- < 5 . パファリング処理 >
- < 6 . 8 ビット / 6 ビット変換 >
- < 7 . テキストデータ有効性検証処理 >
- < 8 . データ例 >
- < 9 . 変形例 >

10

【 0 0 1 3 】

1 . 再生装置の構成

本実施の形態の再生装置に装填される光ディスクは、例えばC D - T E X TなどのC D 方式のディスクや、D V Dなどが考えられる。もちろん他の種類の光ディスクに対応する再生装置でも本発明は適用できるものである。

20

【 0 0 1 4 】

図 1 は本例の再生装置 7 0 の要部のブロック図である。

ディスク 9 0 は、ターンテーブル 7 に積載され、再生動作時においてスピンドルモータ 6 によって一定線速度 (C L V) もしくは一定角速度 (C A V) で回転駆動される。そしてピックアップ 1 によってディスク 9 0 にエンボスピット形態や相変化ピット形態などで記録されているデータの読み出しが行なわれることになる。なお本例ではC L V 方式として説明を続ける。

【 0 0 1 5 】

ピックアップ 1 内には、レーザ光源となるレーザダイオード 4 や、反射光を検出するためのフォトディテクタ 5、レーザ光の出力端となる対物レンズ 2、レーザ光を対物レンズ 2 を介してディスク記録面に照射し、またその反射光をフォトディテクタ 5 に導く光学系が形成される。

30

対物レンズ 2 は二軸機構 3 によってトラッキング方向及びフォーカス方向に移動可能に保持されている。

またピックアップ 1 全体はスレッド機構 8 によりディスク半径方向に移動可能とされている。

【 0 0 1 6 】

ディスク 9 0 からの反射光情報はフォトディテクタ 5 によって検出され、受光光量に応じた電気信号とされてR F アンプ 9 に供給される。

R F アンプ 9 には、フォトディテクタ 5 としての複数の受光素子からの出力電流に対応して電流電圧変換回路、マトリクス演算 / 増幅回路等を備え、マトリクス演算処理により必要な信号を生成する。例えば再生データであるR F 信号、サーボ制御のためのフォーカスエラー信号F E、トラッキングエラー信号T Eなどを生成する。

40

R F アンプ 9 から出力される再生R F 信号は2 値化回路 1 1 へ、フォーカスエラー信号F E、トラッキングエラー信号T Eはサーボプロセッサ 1 4 へ供給される。

【 0 0 1 7 】

R F アンプ 9 で得られた再生R F 信号は2 値化回路 1 1 で2 値化されることでいわゆるE F M 信号 (8 - 1 4 変調信号 ; C D の場合) もしくはE F M + 信号 (8 - 1 6 変調信号 ; D V D の場合) とされ、デコーダ 1 2 に供給される。デコーダ 1 2 ではE F M 復調 , エラー訂正処理等を行ない、また必要に応じてC D - T E X T デコード、C D - R O M デコー

50

ド、またはMPEGデコードなどを行なってディスク90から読み取られた情報の再生を行なう。

【0018】

なおデコーダ12は、EFM復調したデータをデータバッファとしてのキャッシュメモリ20に蓄積していき、このキャッシュメモリ20上でエラー訂正処理等を行う。そしてエラー訂正され適正な再生データとされた状態で、キャッシュメモリ20へのバッファリングが完了される。また、本例ではCD-TEXTデータを検証する際のローデータ(RAW DATA・・・ディスク90から読み出されインターリーブされていないデータ)のバッファリングエリアとしても用いられる。なお、ローデータのバッファリング処理については後述する。

10

再生装置70からの再生出力としては、キャッシュメモリ20でバッファリングされているデータが読み出されて転送出力されることになる。

【0019】

インターフェース部13は、外部のホストコンピュータ80と接続され、ホストコンピュータ80との間で再生データやリードコマンド等の通信を行う。

即ちキャッシュメモリ20に格納された再生データは、インターフェース部13を介してホストコンピュータ80に転送出力される。

またホストコンピュータ80からのリードコマンドその他の信号はインターフェース部13を介してシステムコントローラ10に供給される。

【0020】

20

サーボプロセッサ14は、RFアンプ9からのフォーカスエラー信号FE、トラッキングエラー信号TEや、デコーダ12もしくはシステムコントローラ10からのスピンドルエラー信号SPE等から、フォーカス、トラッキング、スレッド、スピンドルの各種サーボドライブ信号を生成しサーボ動作を実行させる。

即ちフォーカスエラー信号FE、トラッキングエラー信号TEに応じてフォーカスドライブ信号、トラッキングドライブ信号を生成し、二軸ドライバ16に供給する。二軸ドライバ16はピックアップ1における二軸機構3のフォーカスコイル、トラッキングコイルを駆動することになる。これによってピックアップ1、RFアンプ9、サーボプロセッサ14、二軸ドライバ16、二軸機構3によるトラッキングサーボループ及びフォーカスサーボループが形成される。

30

【0021】

またサーボプロセッサ14はスピンドルモータドライバ17に対して、スピンドルエラー信号SPEに応じて生成したスピンドルドライブ信号を供給する。スピンドルモータドライバ17はスピンドルドライブ信号に応じて例えば3相駆動信号をスピンドルモータ6に印加し、スピンドルモータ6のCLV回転を実行させる。またサーボプロセッサ14はシステムコントローラ10からのスピンドルキック/ブレーキ制御信号に応じてスピンドルドライブ信号を発生させ、スピンドルモータドライバ17によるスピンドルモータ6の起動、停止、加速、減速などの動作も実行させる。

【0022】

なお、スピンドルモータ6のCLV回転としての線速度については、システムコントローラ10が各種速度に設定できる。

40

例えばデコーダ12は、デコード処理に用いるためにEFM信号に同期した再生クロックを生成するが、この再生クロックから現在の回転速度情報を得ることができる。システムコントローラ10もしくはデコーダ12は、このような現在の回転速度情報と、基準速度情報を比較することで、CLVサーボのためのスピンドルエラー信号SPEを生成する。従って、システムコントローラ11は、基準速度情報としての値を切り換えれば、CLV回転としての線速度を変化させることができる。例えばある通常の線速度を基準として4倍速、8倍速などの線速度を実現できる。

これによりデータ転送レート的高速化が可能となる。

なお、もちろんCAV方式であっても回転速度の切換は可能である。

50

【 0 0 2 3 】

サーボプロセッサ 1 4 は、例えばトラッキングエラー信号 T E の低域成分として得られるスレッドエラー信号や、システムコントローラ 1 0 からのアクセス実行制御などに基づいてスレッドドライブ信号を生成し、スレッドドライバ 1 5 に供給する。スレッドドライバ 1 5 はスレッドドライブ信号に応じてスレッド機構 8 を駆動する。スレッド機構 8 には図示しないが、ピックアップ 1 を保持するメインシャフト、スレッドモータ、伝達ギア等による機構を有し、スレッドドライバ 1 5 がスレッドドライブ信号に応じてスレッドモータ 8 を駆動することで、ピックアップ 1 の所要のスライド移動が行なわれる。

【 0 0 2 4 】

ピックアップ 1 におけるレーザダイオード 4 はレーザドライバ 1 8 によってレーザ発光駆動される。

10

システムコントローラ 1 0 はディスク 9 0 に対する再生動作を実行させる際に、レーザパワーの制御値をオートパワーコントロール回路 1 9 にセットし、オートパワーコントロール回路 1 9 はセットされたレーザパワーの値に応じてレーザ出力が行われるようにレーザドライバ 1 8 を制御する。

【 0 0 2 5 】

以上のようなサーボ及びデコード、エンコードなどの各種動作はマイクロコンピュータによって形成されたシステムコントローラ 1 0 により制御される。

そしてシステムコントローラ 1 0 は、ホストコンピュータ 8 0 からのコマンドに応じて各種処理を実行する。

20

例えばホストコンピュータ 8 0 から、ディスク 9 0 に記録されている或るデータの転送を求めるリードコマンドが供給された場合は、まず指示されたアドレスを目的としてシーク動作制御を行う。即ちサーボプロセッサ 1 4 に指令を出し、シークコマンドにより指定されたアドレスをターゲットとするピックアップ 1 のアクセス動作を実行させる。

その後、その指示されたデータ区間のデータをホストコンピュータ 8 0 に転送するために必要な動作制御を行う。即ちディスク 9 0 からのデータ読出 / デコード / バファリング等を行って、要求されたデータを転送する。

【 0 0 2 6 】

ホストコンピュータ 8 0 からのリードコマンド、即ち転送要求としては、要求するデータ区間の最初のアドレスとなる要求スタートアドレスと、その最初のアドレスからの区間長として要求データ長（データレングス）となる。

30

例えば要求スタートアドレス = N、要求データ長 = 3 という転送要求は、L B A 「 N 」 ~ L B A 「 N + 2 」 の 3 セクターのデータ転送要求を意味する。L B A とは論理ブロックアドレス（LOGICAL BLOCK ADDRESS）であり、ディスク 9 0 のデータセクターに対して与えられているアドレスである。

【 0 0 2 7 】

2 . T O C 及びサブコード

次に、ディスク 9 0 においてリードインエリアに記録される T O C、及びサブコードについて説明する。

ディスク 9 0 において記録されるデータの最小単位は 1 フレームとなる。9 8 フレームで 1 ブロック（1 サブコーディングフレーム）が構成される。

40

【 0 0 2 8 】

1 フレームの構造は図 2 のようになる。

1 フレームは 5 8 8 ビットで構成され、先頭 2 4 ビットが同期データと、これに続く 3 ビットによるマージンビットが設定され、続いて 1 4 ビットがサブコードデータエリアとされる。そして、その後に 1 2 シンボルのメインデータ及び 4 シンボルのパリティデータが配される。

【 0 0 2 9 】

この構成のフレームが 9 8 フレームで 1 ブロックが構成され、9 8 個のフレームから取り出されたサブコードデータが集められて図 3（a）のような 1 ブロックのサブコードデー

50

タが形成される。

98フレームの先頭の第1、第2のフレーム(フレーム98n+1, フレーム98n+2)からのサブコードデータは同期パターンとされている。そして、第3フレームから第98フレーム(フレーム98n+3~フレーム98n+98)までで、各96ビットのチャンネルデータ、即ちP, Q, R, S, T, U, V, Wのサブコードデータが形成される。

【0030】

このうち、アクセス等、再生に関わる各種制御には再生管理情報とされるPチャンネルとQチャンネルが用いられる。但し、Pチャンネルはトラックとトラックの間のポーズ部分を示しているのみで、より細かい制御はQチャンネル(Q1~Q96)によって行なわれる。96ビットのQチャンネルデータは図3(b)のように構成される。

10

Rチャンネル~Wチャンネルのデータは、テキストデータ群を形成するために設けられるが、これについては後述する。

【0031】

まずQ1~Q4の4ビットはコントロールデータとされ、オーディオのチャンネル数、エンファシス、CD-ROMの識別などに用いられる。

即ち、4ビットのコントロールデータは次のように定義される。

『0***』・・・・・・2チャンネルオーディオ

『1***』・・・・・・4チャンネルオーディオ

『*0**』・・・・・・CD-DA(CDデジタルオーディオ; CD-TEXTを含む)

『*1**』・・・・・・CD-ROM

20

『**0*』・・・・・・デジタルコピー不可

『**1*』・・・・・・デジタルコピー可

『***0』・・・・・・プリエンファシスなし

『***1』・・・・・・プリエンファシスあり

【0032】

次にQ5~Q8の4ビットはアドレスとされ、これはサブQデータのコントロールビットとされている。

このアドレス4ビットが『0001』である場合は、続くQ9~Q80のサブQデータはオーディオQデータであることを示し、また『0100』である場合は、続くQ9~Q80のサブQデータがビデオQデータであることを示している。

30

そしてQ9~Q80で72ビットのサブQデータとされ、残りのQ81~Q96はCRCとされる。

【0033】

リードインエリアにおいては、そこに記録されているサブQデータが即ちTOC情報となる。

つまりリードインエリアから読み込まれたQチャンネルデータにおけるQ9~Q80の72ビットのサブQデータは、図4(a)のような情報を有するものである。サブQデータは各8ビットのデータを有している。

【0034】

まずトラックナンバが記録される。リードインエリアではトラックナンバは『00』に固定される。

40

続いてPOINT(ポイント)が記され、さらにトラック内の経過時間としてMIN(分)、SEC(秒)、FRAME(フレーム番号)が示される。

さらに、PMIN, PSEC, PFRAMEが記録されるが、このPMIN, PSEC, PFRAMEは、POINTの値によって、次に述べるように意味が決定されている。

【0035】

POINTの値が『01h』~『99h』(hは16進表現であることを示す)のときは、その値はトラックナンバを意味し、この場合PMIN, PSEC, PFRAMEにおいては、そのトラックナンバのトラックのスタートポイント(絶対時間アドレス)が分(PMIN), 秒(PSEC), フレーム番号(PFRAME)として記録されている。

50

【 0 0 3 6 】

P O I N T の値が『 A 0 h 』のときは、P M I N に最初のトラックのトラックナンバが記録される。また、P S E C の値によって C D - D A , C D - I , C D - R O M (X A 仕様) が区別される。

P O I N T の値が『 A 1 h 』のときは、P M I N に最後のトラックのトラックナンバが記録される。

P O I N T の値が『 A 2 h 』のときは、P M I N , P S E C , P F R A M E にリードアウトエリアのスタートポイントが絶対時間アドレスとして示される。

【 0 0 3 7 】

例えば 6 トラックが記録されたディスクの場合、このようなサブ Q データによる T O C としては図 5 のようにデータが記録されていることになる。

図 5 に示すようにトラックナンバ T N O は全て『 0 0 h 』である。

ブロック N O . とは上記のように 9 8 フレームによるブロックデータとして読み込まれた 1 単位のサブ Q データのナンバを示している。

各 T O C データはそれぞれ 3 ブロックにわたって同一内容が書かれている。

図示するように P O I N T が『 0 1 h 』～『 0 6 h 』の場合、P M I N , P S E C , P F R A M E としてトラック # 1 ～トラック # 6 のスタートポイントが示されている。

【 0 0 3 8 】

そして P O I N T が『 A 0 h 』の場合、P M I N に最初のトラックナンバとして『 0 1 』が示される。また P S E C の値によってディスクが識別され、このディスクが C D - D A の場合は、図示するように P S E C = 『 0 0 h 』とされる。なお、C D - R O M (X A 仕様) の場合は、P S E C = 『 2 0 h 』、C D - I の場合は『 1 0 h 』となる。

【 0 0 3 9 】

そして P O I N T の値として『 A 1 h 』の位置に P M I N に最後のトラックのトラックナンバが記録され、さらに P O I N T の値として『 A 2 h 』の位置には、P M I N , P S E C , P F R A M E にそれぞれリードアウトエリアのスタートポイントが示される。

ブロック n + 2 7 以降は、ブロック n ~ n + 2 6 の内容が再び繰り返して記録されている。

【 0 0 4 0 】

ディスク 9 0 上で実際に音楽等のデータが記録されるトラック # 1 ~ # n 、及びリードアウトエリアにおいては、そこに記録されているサブ Q データは図 4 (b) に示されている情報を有する。

まずトラックナンバが記録される。即ち各トラック # 1 ~ # n では『 0 1 h 』～『 9 9 h 』のいずれかの値となる。またリードアウトエリアではトラックナンバは『 A A h 』とされる。

続いてインデックスとして各トラックをさらに細分化することができる情報が記録される。

【 0 0 4 1 】

そして、トラック内の経過時間として M I N (分) 、 S E C (秒) 、 F R A M E (フレーム番号) が示される。

さらに、A M I N , A S E C , A F R A M E として、絶対時間アドレスが分 (A M I N) , 秒 (A S E C) , フレーム番号 (A F R A M E) として記録されている。

【 0 0 4 2 】

このように T O C 及びサブコードが形成されているわけであるが、ディスク上のアドレス、即ち A M I N , A S E C , A F R A M E は、9 8 フレーム単位で記録されることが理解される。

この 9 8 フレーム (1 ブロック) は 1 サブコーディングフレームと呼ばれ、音声データとしての 1 秒間には 7 5 サブコーディングフレームが含まれることになる。つまり、アドレスとしての『 A F R A M E 』がとりうる値は『 0 』～『 7 4 』となる。なお、後述するフレームチェック処理でデータの連続性がチェックされるのは、このサブコーディングフレ

10

20

30

40

50

ーム単位となる。

【 0 0 4 3 】

3 . テキストデータ

以降、図 2 及び図 3 に示す構造のサブコードに含まれるテキストデータについて説明を行うこととし、まず、図 6 によりテキストデータの包括的な構造について説明する。

サブコード内に含まれるテキストデータのみを抽出して構造的に見た場合、テキストデータは図 6 に示すようなものとなる。テキストデータとしての最も大きなデータ単位は、図 6 (a) に示す『テキストグループ』とされる。図 6 (a) においては複数のテキストグループが示されているが、各テキストグループのデータは同一内容とされており、従って、サブコード内においては、同一データ内容の所定数の複数のテキストグループが繰り返し記録されていることになる。

10

【 0 0 4 4 】

1 つのテキストグループは、例えば最大 2 0 4 8 パック (パックの定義については後述する) により形成するものとされるが、1 テキストグループあたりのデータ読出しに要する時間等を考慮して、1 テキストグループを 5 1 2 パック以内により形成することが推奨されている。この際、1 テキストグループあたりのデータ総量としては 6 5 0 0 文字程度となる。

また 1 つのテキストグループは、図 6 (b) に示すようにブロック # 0 ~ ブロック # n により形成され、例えば最大 8 ブロック (0 ~ n - 1 となる) であると規定されている。各ブロックは、同一の内容の情報をそれぞれ異なる言語により表記するためのテキストデータが格納されているものとされる。例えば、ブロック # 0 には、当該ディスクに対応する各種情報を英語により表記するためのテキストデータが格納されており、ブロック # 1 には、ブロック # 0 と内容的には同一の情報を日本語により表記するためのテキストデータが格納されているものとされる。

20

【 0 0 4 5 】

この場合、1 テキストグループは最大 8 ブロックにより形成可能であることから、テキストデータのフォーマットとしては最大 8 言語に対応することが可能とされる。

1 つのブロックは、図 6 (c) に示すようにパック # 0 ~ パック # n のデータ単位により形成される。ここでは、1 ブロックは最大 2 5 6 パックで形成され情報量としては例えば 3 6 . 8 6 4 K B y t e とされている。なお、パック内のデータ構造等については、次の図 7、図 8 及び図 9 により説明する。

30

【 0 0 4 6 】

図 7 (a) は、図 3 に示した 1 サブコーディングフレームをデータ領域別に示すものであり、前述のように 1 サブコーディングフレームは 9 8 フレームにより形成される。

9 8 フレームの先頭の第 1、第 2 フレーム (フレーム 9 8 n + 1 , フレーム 9 8 n + 2) は、図 3 で説明したように同期パターン S 0 , S 1 の領域とされる。また、第 3 フレームから第 9 8 フレーム (フレーム 9 8 n + 3 ~ フレーム 9 8 n + 9 8) における P チャンネルはサブコード P のデータ領域とされ、Q チャンネルはサブコード Q のデータ領域とされて、前述したようにアクセス等の管理のためのデータが格納される。

【 0 0 4 7 】

そして、第 3 フレームから第 9 8 フレームにおける R チャンネル ~ W チャンネルの領域は図のようにパック 0 ~ パック 4 とされる。各パックのデータサイズは固定長とされて、図 7 (b) に示すようにシンボル 0 ~ 2 3 の 2 4 シンボルにより形成される。1 シンボルは図 7 (c) に示すように、1 フレームにおける R , S , T , U , V , W のチャンネルデータよりなる 6 ビットのデータ単位であり、この場合には R チャンネルデータが M S B、W チャンネルが L S B として定義される。

40

【 0 0 4 8 】

図 8 は、上記図 7 (a) に示す構造の 1 サブコーディングフレームから、4 つのパック (パック 0 ~ パック 4) によるデータ構造を抜き出して示している。

1 パックは、図 7 で説明したように、2 4 のシンボル (6 ビット) により形成されること

50

から、

$6 \text{ ビット} \times 24 / 8 = 18 \text{ バイト}$

で示されるように18バイトのデータサイズを有する。そして、図のように1パックは、先頭のID領域と続くテキストデータ領域により16バイトを占有し、残りの2バイトはCRC領域となる。

また、前述のように1サブコーディングフレームにおいては4つのパックが設けられるが、これら4つのパックの集合により形成されるデータ単位はパケットとして定義されている。1パックは24シンボルにより形成されることから、1パケットは、

$24 \text{ (シンボル)} \times 4 \text{ (パック)} = 96 \text{ (シンボル)}$

で示されるように96のシンボルにより形成されるものとみることができる。

10

【0049】

図9及び図10は、図8で示した1パック分のデータをシリアルに表現したものである。図9(a)から分かるように、本実施の形態におけるテキストデータのフォーマットにおいては、6ビットよりなるシンボルをシリアルに配列させたとえ、このデータ列を8ビット(1バイト)ごとに区切るようにしてデータを扱うように規定されている。

【0050】

本実施の形態のテキストデータのフォーマットでは、図9(b)及び図10に示すように、パックの先頭からはヘッダ領域としてID1、ID2、ID3、ID4の4つのIDデータが設けられる。なお、ID1はPack_Type_Indicator、ID2はTrack_Number_Indicator、ID3はSequence_Number_Indicator、ID4はBlock_Number_and_Character_Position_Indicatorとされているが、これらの内容については後述する。

20

【0051】

本実施の形態のフォーマットとして8ビット(1バイト)ごとに区切ってデータを扱うことにより、これら各IDはそれぞれ8ビット(1バイト)のデータ単位とされることになる。このため、図9(b)に示すように、ヘッダ領域(ID1~ID4)以降の残りの12バイトがテキストデータ領域として確保され、残りの2バイトがCRC領域となる。

そして、上記12バイトのテキストデータ領域も、図10に示すパックの構造図に示すように、8ビットごとのtext1~text12のデータ単位により扱われるものとされる。

30

なお、以降の説明では、単に『ヘッダ』という場合はパックの先頭にあるヘッダ領域(ID1~ID4)を示し、テキストグループ内における先頭のパックのヘッダについては『スタートヘッダ』ということとする。

【0052】

ここで、本実施の形態のテキストデータのフォーマットにおいては、図11に示すようにテキストデータ対応以外のCDのフォーマットに準じて、パックの先頭のID1の上位3ビットをモード(MODE)として扱い、続く3ビットをアイテム(ITEM)として扱うことができるようにしている。

そして、上位3ビットのモードとしては、モード4を設定してこの3ビットに対して値『100』を設定するようにしている。このモード4はCD-TEXTフォーマットが提案される前は未定義とされている。こうすることで、例えばテキストデータに対応していない再生装置等にテキストデータが格納されたCDを装填しても、モードとして認識不可能とされることでその動作を停止するだけであって、誤動作することはないようにされる。また、モード4としてはテキストデータがリードインエリアに記憶されているモードであるが、この他にも、プログラムエリア(主データが格納されているエリア)に対してテキストデータが記憶されているモード2が知られている。本例ではモード4を例に挙げて説明するが、本発明はモード2を適用している場合にも対応することが可能とされる。

40

【0053】

なお、未定義のモードとしてはモード5及びモード6が存在するので、モード4の代わり

50

にこれらのモードに対応する値を設定することも可能である。また、参考までに、使用済みのモードとしては、C D - Gに対応するモード1、C D - M I D Iに対応するモード3等が存在する。

なお、ここではアイテム (I T E M) としての値は特に設定されず、後述するように、I D 1により定義する識別内容に応じて下位3ビット以降の値は適宜異なる (実際には下位4ビットのみが変更される) もとなる。

【 0 0 5 4 】

次に、図12及び図13を参照して、本実施の形態が対応するテキストデータフォーマットにおけるヘッダ領域 (I D 1、I D 2、I D 3、及びI D 4) の各定義内容について説明する。図12 (a) ~ (d) はそれぞれI D 1 ~ I D 4を示し、図13はI D 1の定義内容を示している。

10

図12 (a) に示すI D 1 (8ビット) は、当該パックのテキストデータ領域における t e x t 1 以降に格納される文字列の内容の種類を識別するためのデータが設定されるものであり、『 8 0 h 』 ~ 『 8 F h 』の値をとるものとされる。ここで、I D 1について設定される値として上位4ビットが16進法によりすべて『 8 』が設定されているのは、図11で説明したように、I D 1の上位3ビットをモード (M O D E) として扱った場合に『 1 0 0 』の値が得られるようにして、モード4として識別できるようにするためである。

【 0 0 5 5 】

I D 1に設定される値『 8 0 h 』 ~ 『 8 F h 』に対応する定義内容は、図13に示すように規定されている。この図によると、I D 1が『 8 0 h 』の場合には、t e x t 1 以降に格納される文字列の内容が、アルバムタイトル (T i t l e O f A l b u m N a m e . . . I D 2 が『 0 0 h 』の場合) 又はトラックに記録された楽曲等の曲名 (T r a c k T i t l e . . . I D 2 が『 0 1 h ~ 6 3 h 』の場合) であることを示すことになる。また、I D 1が『 8 1 h 』の場合には演奏者、指揮者、又はオーケストラ名であることを示し、『 8 2 h 』の場合には作詞者名、『 8 3 h 』の場合には作曲者名、『 8 4 h 』の場合には編曲者名であることを示す。I D 1が『 8 5 h 』の場合には、当該C Dを供給する者 (例えばレコード会社等) や演奏者などからのメッセージであることが示される。

20

【 0 0 5 6 】

また、I D 1が『 8 6 h 』の場合には、例えばカタログナンバやレコード会社の名前等により決められるディスクIDであることが示され、『 8 7 h 』の場合にはジャンルを示すテキストデータであることが示され、『 8 8 h 』の場合にはT O Cデータであることが示される。このT O Cデータは、例えばQチャンネルのサブコードデータに準ずる内容を示すものとなる。また、『 8 9 h 』の場合には、2 n d T O Cであることが示される。I D 1として『 8 A h 』、『 8 B h 』、『 8 C h 』は予約 (R E S E R V E D) とされている。

30

【 0 0 5 7 】

『 8 D h 』は、当該C Dの製造管理に関する情報や当該パック内に記録されている内容に関するコメント等であることが示され、『 8 E h 』はアルバムのU P C / E A NコードやトラックのI S R Cコードであることが示される。

『 8 F h 』は、ブロック内のサイズインフォメーション (S i z e I n f o r m a t i o n) が示されている。

40

【 0 0 5 8 】

図12 (b) に示すI D 2は、当該パックのテキストデータ領域における t e x t 1 以降に格納される文字列がどのトラックに対応するのかをトラックナンバにより示すものとされ、I D 2を形成する8ビットにより、『 0 0 h 』 ~ 『 6 3 h 』 (10進法では0 ~ 99) の値をとるものとされる。但し、トラックナンバは「1」からインクリメントされるようにして付されていくものであるため、トラックナンバとしては『 0 1 h 』 ~ 『 6 3 h 』 (10進法では1 ~ 99) の値をとることになる。値『 0 0 h 』は、ディスク全体を代表することを意味するものとされる。

I D 2のM S Bは、拡張用フラグとされているがここでは常に『 0 』を設定するものとさ

50

れ、『1』が設定されると拡張用のフラグが立てられたことになる。

【0059】

図12(c)に示すID3は、当該パックが属するブロック内において、当該パックが何番目のパックであるのかを示すパックのブロック内連番を示すものとされ、ID3を形成する8ビットにより、『00h』～『FFh』（10進法では0～255）の値をとるものとされる。

【0060】

図12(d)に示すID4は、現パックのブロック番号（文字コードの識別情報を含む）と、1まとまりの文字列の文字位置を示すものとされる。

MSBは当該パックにおけるテキストデータが1バイトコードか2バイトコードであるかを示す2バイトコードフラグの領域とされ、値として「1」の場合には2バイトコードであることが示され、「0」の場合には1バイトコードであることが示される。

MSBに続く上位第2ビットから第4ビットまでの3ビットは、当該パックを含むブロック（図6(b)参照）のブロック番号が示され、2進法で『000』～『111』（10進法では0～7）の値を取るものとされる。図6により説明したようにブロックは最大8つ設けられて、ブロックナンバとしてはブロックナンバ0～7の値を取り得ることになるが、上記3ビットにより取り得る値はこれに対応したものとなる。

【0061】

ところで、現状として、少なくともブロック#0においては、テキストデータとしてASCIIコードを含む8859-1コードのみを使用することが規定されている。つまり、ブロック#0においては、一般的には言語として英語による表記を行うためのテキストデータが格納されることになる。なお、以降の説明においては、便宜上、ブロック0は言語として英語に対応し、文字コードとしてはASCIIコードを用いるものとする。ASCIIコード（及び8859-1コード）は1バイトコードであることから、ブロック#0に含まれる全パック内のID4においては、上位4ビットは『0000』となる。

【0062】

ID4の下位4ビットは、現パックにおける文字位置の情報が格納される。つまり、現パックのテキストデータ領域における最初のtext1に格納されている文字データが、1まとまりの文字列における何番目の文字であるのかを示すものとされ、図13(d)に示すように、2進法で『0000』～『1111』の値をとる。なお、16番目以上の文字である場合にはすべて『1111』となる。

また、ここでいう「1まとまりの文字列」とは、例えば、トラックの曲名データであれば、この1トラック分の曲名を形成する一連の文字列を意味するものである。

【0063】

図14には、トラックごとの曲名を示すテキストデータをテキストデータ領域に格納する場合の1パックの構造例が示されている。この場合、図12(a)及び図13により説明したように、ID1は『80h』とされ、ID2には、当該パックのテキストデータにより表記するタイトルのトラックに対応するトラックナンバが『01h』～『63h』（トラック1～99）により適宜示される。ID3には、ブロック内における当該パックのブロック内連番が『00h』～『FFh』により示される。ID4は当該パックを含む現ブロックのブロック番号（図6(b)参照）が第2ビット～第4ビットの3ビットにより示され、現ブロックが対応する文字コードが2バイトコードか1バイトコードであるかがMSBにより示される。例えば、現パックのテキストデータがASCIIコードに対応するものであれば、前述のようにID4の上位4ビットは『0000』となる。

【0064】

また、ID4の下位4ビットは、text1に格納されている文字データが、1まとまりの文字列における何番目の文字であるのかを示す値が適宜示されることになる。トラックごとの曲名を示すテキストデータの場合、上記「1まとまりの文字列」とは、トラックごとの曲名に対応する文字列がこれに相当するものとされ、例えば、あるトラックの曲名が『THIS IS A PEN』であるとして、このTHIS__IS__A__PENの文字

10

20

30

40

50

列のうち、2番目の『H』の文字データが当該パックのtext1に格納されているとすれば、当該パックのID4における下位4ビットは『0001(1h)』とされることになる。

【0065】

この場合、『THIS IS A PEN』のうち最初の文字データ『T』は、当該パックの直前のパックのテキストデータ領域に格納されていることになる。つまりテキストデータは、1まとまりの文字列データが連続するパックをまたがるようにしてテキストデータ領域に格納することが可能とされるフォーマットとされている。

そして、テキストデータ領域である各8ビットのtext1～text12には、本実施の形態のテキストデータフォーマットに則った規則に従って、各トラックの曲名を示す文字コードのデータが格納されることになる。

【0066】

4.再生処理

本実施の形態におけるテキストデータの再生処理の概要を説明する。まず、ここでは全体の流れを説明し、詳細な説明については順次行なっていくこととする。

【0067】

上記したように、「モード4」においてはテキストデータはディスク90のサブコードデータの一部に記録されている。したがって、テキストデータを扱う場合は、まずサブコードデータ(P～W)から不要なデータ(P、Q)を排除してテキストデータ(R～W)の抽出を行なうことになる。

この場合サブコードデータはデインターリーブを行なわないローデータとしてキャッシュメモリ20に順次格納していく。ローデータとしては96バイト(96bit×8(P～W))、即ち1サブコードフレームに相当するデータ量が1回のバッファリング単位とされる。そして、キャッシュメモリ20に格納されたサブコードデータの中から、所望するテキストデータの位置を識別する検出処理を行なう。この検出処理としてはテキストグループのスタートヘッダを検出するための処理とされる。以降、この処理をバッファリング処理という。

【0068】

バッファリング処理としては、サブコードデータにおいて図6(a)に示したテキストグループの先頭とされる位置(先頭パックのヘッダに相当する)を起点とし読み出しを行なうこと望ましいが、実際にはこれは困難な処理とされる。そこで、1回の読み込みではバッファリング処理を行なうに際して必要なデータが得られない場合を想定して、キャッシュメモリ20には少なくとも2テキストグループ分のデータ量に相当するサブコードデータを格納することができるようにして、この格納された2テキストグループ分のサブコードデータの中からスタートヘッダを検出するようにしている。

【0069】

バッファリング処理を行なう場合、キャッシュメモリ20は例えば図15に示されている形態でマッピングされた状態で使用する。ローサブコードエリア(RAW__SUB__CODE__AREA)は、例えば2テキストグループ分のデータ量に相当するのローサブコードデータ(RAW__SUB__CODE__DATA(P～W))を格納することができる領域とされる。

そして、後述するようにローサブコードエリアに格納されたサブコードデータに対してバッファリング処理を行ない、このバッファリング処理によって検出された、スタートヘッダに基づいてローテキストデータ(RAW__TEXT__DATA(R～W))の抽出(8ビット(P～W)から6ビット(R～W)への変換処理・・・以下、8ビット/6ビット変換処理という)を行なう。ここで、抽出されたローテキストデータは、図15に示すローテキストエリア(RAW__TEXT__AREA)に格納される段階で、データ内容の有効性が検証される。

有効性が確認されたローテキストデータは、ローテキストエリアから読み出されてホストコンピュータ80に対して転送されるようにされる。即ち、ホストコンピュータ80に対

10

20

30

40

50

しては、有効性が確認されたローテキストデータのみを転送することができるようになる。

【0070】

ところで、図示している各エリアの容量を示す数値は一例である。

前記したように、テキストグループの1ブロックの情報量は上述したように例えば36・864Kバイトとされる。したがって、ローテキストエリアに必要な容量としては少なくとも、 $36 \cdot 864 \times 2 = 73 \cdot 728$ Kバイト以上であれば良い。

また、ローサブコードエリアは、 $36 \cdot 864 \times 8 / 6 = 49 \cdot 152$ Kバイトが1フレーム分のデータ量に相当するので、2フレームに相当する容量としては $49 \cdot 152 \times 2 = 98 \cdot 304$ Kbyteとなる。したがって、ローサブコードエリアに必要な容量は、少なくとも98・304 Kbyte以上であれば良い。

10

【0071】

5．バッファリング処理

図16、図17は、バッファリング処理の流れを模式的に説明するフローチャートを示す図である。なお、図16に示されているフローチャートの結合子(1)は、図17に示されているフローチャートの結合子(1)に対応したものである。即ち、バッファリング処理としては図16乃至図17に示されている一連の処理行程によって実現される処理とされる。したがって、各図には連続したステップ番号を付している。

【0072】

バッファリング処理の開始を指示するコマンドはホストコンピュータ80から供給される(S001)。ここで、再生装置が受信するコマンドとしては、例えば「A0コマンド」、「Packetコマンド」などのリードTOCコマンドを拡張してCD-TEXTに格納されたテキストデータを読み込む処理を開始するためのコマンドとされる。

20

このようなコマンドを受信すると、「OPコードチェック」、「CDBチェック」、「メディアチェック」などの各種チェック処理が行なわれる(S002)。

【0073】

「OPコードチェック」によって、受信したコマンドのOPコードが当該装置に対応しているものの可否判別が行なわれ、「CDBチェック」によって、コマンド(12バイト)の内容が規格に準拠しているかの可否判別が行なわれる。これらの判別結果が「可」とされた場合、「メディアチェック」によって、再生装置70に装填されているディスク90が、例えばオーディオCDであるか否か(つまり、CD-ROMなどではなくCD-TEXTとしてテキストデータを有する可能性のあるディスクであるか否か)の判別が行なわれる。なお、CD-TEXTとは種別的にはオーディオCD(CD-DA)に含まれるものである。

30

ディスク90がオーディオCD(オーディオトラックが存在する)か否かは、上述したTOCにより判別できる。そして通常、TOCデータはディスク装填時に読み込んでいるため、即座に判別可能である。

【0074】

ここで、ディスク90がオーディオCDであると判別された場合は、「Allocation length = 0」か否か、即ち、要求するデータ長の確認を行ない(S003)、要求するデータがある場合(「0」でない場合)ステップS004に進む。

40

なお、各種チェック処理(S002)における判定が「否」であった場合は、その時点で所要のエラー処理が行なわれ、バッファリング処理は行なわれないことになる。

【0075】

要求するデータ長の確認(S003)が終了すると、ファームウェア内の状況の初期化を行なう(S004)。そして、例えば現在当該再生装置70によってオーディオ再生が行なわれている場合は、その再生処理を強制的に終了する(S005)。但し、現在オーディオ再生処理を行っていない場合は、この処理行程は省略される。ステップS005を経ると、さらにキャッシュパラメータの初期化を行なう(S006)。これは、キャッシュメモリ20の使用形態が、通常の状態から図15に示した状態(ローサブコードエリアとローテキストエリ

50

アがマッピングされる状態)に移行するため、現在まで格納されているデータが不要となるので、これらのデータを無効にするための処理とされる。

そして、本実施の形態のバッファリング処理を行なうために実行すべき割り込み処理の切替えを行なう(S007)。これは、例えばデインターリーブをオフにするなど、バッファリング処理を行なうための所要の割り込み処理を実行させるためとされる。

【0076】

図16のフローチャートにおいて割り込み処理の切替え処理(S007)を経ると、図17のフローチャートに示されているように、ディスク90のリードインエリアのTOCに対するアクセス要求を行なう(S008)。これによりピックアップ1はディスク90の内周側に記憶されているTOCに対してアクセスを開始する。

10

さらに、ローサブコードデータが96バイト分バッファリングされた時点で「1」ずつインクリメントされるパラメータとして設定される『N』、テキストグループの先頭を示すヘッダ情報とされるスタートヘッダが検出された回数を示すパラメータとして設定される『M』、及びCRCのNG回数を示すパラメータとして設定される『NG数』を初期値化「0」する(S009)。

アクセス開始後は、アクセス完了に対して待機状態となり(S010)、アクセスが完了した時点で、ローサブコードデータのバッファリングを開始する(S011)。即ち、デインターリーブをオフにするとともに、図15に示したローサブコードエリアに対してローサブコードの格納を開始させる。

【0077】

20

そして、ローサブコードエリアに対して所定量のローサブコードデータ(例えば1サブコードに相当するデータ量)が格納されるのを待つことになる(S012)。ここで、所定量のローサブコードデータが格納されると、その時点で割り込み処理として所要の通知処理が行なわれ、システムコントローラ10ではローサブコードエリアに対してローサブコードデータが格納されたことを認識することができる。なお、この図に示すフローチャートとしてはステップS012の判別処理を便宜上一連の処理行程の一部として示しているが、実際には、この判別処理は随時行なわれているものとされる。つまり、ローサブコードエリアに対するローサブコードデータの読み込みも、随時行なわれている。したがって、以降説明する処理行程は、ローサブコードエリアに所定量のローサブコードデータが格納される度に、随時割り込み処理として行なわれるようにされる。

30

【0078】

ローサブコードエリアに所定量のローサブコードデータが蓄えられると、これをカウントするパラメータ『N』をインクリメント($N = N + 1$)する処理を行なう(S013)。そして、蓄えられたローサブコードデータに対してCRCのチェックを行なう(S014)。このCRCチェックは、図8に示した4個のCRC領域に或る誤り検出符号に対して行われ、チェックによる判定結果としては、4個のCRC領域全てが「OK」とされた場合に「OK」とされ、一つでも「NG」があった場合には「NG」とされる。

ステップS014で、CRCが「OK」となった場合、即ちローサブコードエリアに蓄えられたローサブコードデータが有効であるとされた場合は、テキストグループの先頭を示すスタートヘッダのパターン(Start_Header_Pattern)のチェックを行なう(S016)。

40

【0079】

そして、スタートヘッダの検出の有無の判別を行ない(S017)、スタートヘッダが検出されない場合は、ステップS012に戻る。

また、スタートヘッダが検出された場合は、検出されたスタートヘッダの数をカウントするパラメータ『M』をインクリメント($M = M + 1$)する処理を行ない(S018)、続いて、パラメータ『M』の値の判別を行なう(S019)。ここで、パラメータ『M』が「1」以下であると判別した場合は、『Count1』にパラメータ『N』の値を設定して(S020)、ステップS012に戻る。なお、『Count1』は、何回目のローサブコードデータの読み込みにおいて第一のスタートヘッダが検出されたかを示すパラメータとされる。

50

【 0 0 8 0 】

一方、ステップS019においてパラメータ『M』が「2」または「2」以上であると判別した場合は、『Count2』にパラメータ『N』の値を設定する(S021)。この『Count2』は、何回目のローサブコードデータの読み込みにおいて第二のスタートヘッダが検出されたかを示すパラメータとされる。

即ち、パラメータ『M』が「2」または「2」以上となった時点で、第一のスタートヘッダ及び、第二のスタートヘッダが検出されたことになる。つまり、第一、第二のスタートヘッダ間に1個のテキストグループがバッファリングされたとみなすことができる。

【 0 0 8 1 】

以降、このようにして検出された第一、第二のスタートヘッダ間におけるテキストデータの有効性を検証する処理(2)に移行するが、これについては後で図21にしたがい詳しく説明する。また、ステップS021を経て処理(2)に移行した場合でも、図17に示したバッファリング処理は継続して行なうようにしている。

これにより、後述する処理(2)においてテキストデータの有効性が確認されなかった場合でも、その時点で引き続きキャッシュメモリ20のローサブコードエリアに格納されているローサブコードデータの検証を行えば良い。したがって、改めてTOCに対するアクセス処理などを行なう必要がないので、処理効率が向上する。

【 0 0 8 2 】

また、例えば2個のスタートヘッダが検出された時点でローサブコードデータのバッファリングを終了させたいと、テキストデータの有効性の検証処理に移行するようにしても良い。これは、2個のスタートヘッダが検出されたことによって、その時点で期待されるテキストデータが得られたと見なすようにする処理とされる。これにより、2個のスタートヘッダが検出された後は、システムコントローラ10は前記検証処理のみを実行させれば良いので、処理負担を軽減することが可能である。

【 0 0 8 3 】

ところで、ステップS014においてCRCが「NG」となった場合は、CRCのNG回数を示すパラメータとして設定される『NG数』のインクリメント($NG数 = NG数 + 1$)を行ない(S022)、さらに、『NG数』が当該再生装置70に設定されている『NG許容数』(例えば「4」回)以下であるか否かの判別を行なう(S023)。ここで、『NG数』が『NG許容数』以下であった場合は、リトライ処理としてステップS012に戻り、引き続きサブコードデータが読み込まれるのを待つ。

また、『NG数』が『NG許容数』以上であった場合は、リトライ処理としてステップS008に戻りTOCに対するアクセス要求を行ない、さらに上述したステップS009からの処理を実行していくようにする。

【 0 0 8 4 】

なお、本例では『NG許容数』は例えば「4」として、『NG数』がこれを超えた場合に再びTOCに対するアクセス要求を行なうようにしている。しかし、『NG数』が例えば256(全てのパックについて「NG」の判定)を超えるような場合、テキストデータを読むことができる見込みがないものと見なし、図16、図17に示すフローチャートの処理自体を終了させるようにしても良い。

【 0 0 8 5 】

この場合の処理行程としては、ステップS014、S022を経た後、例えば図18に示されるようになる。この図18において、図17と同一の符号を付した処理ステップについては同じ処理行程を示している。

図示されているように、ステップS022において『NG数』のインクリメント処理を経ると、パラメータ『M』、即ち、テキストグループの先頭を示すヘッダ情報とされるスタートヘッダが検出された回数が「0」であるか否かの判別を行う(S701)。そして判別結果が「0」とされスタートヘッダが検出されないと判別した場合は、ローサブコードデータが例えば96バイトを単位として何回バッファリングされたかを示すパラメータ『N』が所定値以上であるか否かの判別を行う(S702)。なお、ステップS702における所定値とは、

10

20

30

40

50

ディスク 90 にテキストデータが記憶されているか否かを判別するための閾値として設定される。即ち、ステップ S014 において CRC が「NG」になった時点で、ローサブコードデータが何回バッファリングされたかに基づいて、以降の処理が行われる。

【0086】

例えば、パラメータ『N』が所定値以下であったと判別した場合は、さらに、『NG数』が例えば「64」を超えたか否かの判別を行う（S703）。このステップ S703 における閾値とされる「64」は、先述したようにテキストグループを形成する 1 ブロックが 256 バック（256 バック = 64 サブコードフレーム）によって形成されていることに基づいている。

したがってステップ S703 において「NG数」が例えば「64」を超えていたと判別した場合は、再生装置 70 に装填されたディスク 90 にはテキストデータが格納されていないと判断する（S704）。これにより、テキストデータの読み込むことができる見込みがないものと見なし、バッファリング処理を終了させる（S705）。

また、「NG数」が例えば「64」を超えていないと判別した場合は、リトライ処理としてステップ S012 に戻り、引き続きサブコードデータが読み込まれるのを待つ。

【0087】

また、ステップ S702 においてパラメータ「N」が所定値を超えていなかった場合には、『NG数』が当該再生装置 70 に設定されている『NG許容数』（例えば「4」回）以下であるか否かの判別を行なう（S023）。ここで、『NG数』が『NG許容数』以下であった場合は、リトライ処理としてステップ S012 に戻り、引き続きサブコードデータが読み込まれるのを待つ。

また、『NG数』が『NG許容数』以上であった場合は、リトライ処理としてステップ S008 に戻り TOC に対するアクセス要求を行ない、図 17 で説明した場合と同様にステップ S009 からの処理を実行していくようにする。

【0088】

このように、ステップ S701、S702、S703 を経ることによって、スタートヘッドが検出されず、サブコードデータのバッファリング回数が所定値以下であり、かつ CRC の「NG数」が例えば「64」以上であった時点で、ディスク 90 にテキストデータが格納されていないことを判別することができるようになる。したがって、ディスク 90 がオーディオ CD としてテキストデータを有していないと判別した時点で、図 16、図 17 に示したバッファリング処理を終了させることができる。

つまり、再生装置 70 にテキストデータが記憶されていないオーディオ CD が装填されている場合は、バッファリング処理が開始されると、連続して CRC が NG とされる処理ステップを進むことになる。この場合、スタートヘッドは検出されないため、バッファリングの回数（S702）及び「NG数」（S703）に基づいて、当該ディスク 90 にテキストデータが記録されているか否かを判別することができるようになる。これにより、バッファリング処理が開始されてから、必要最小限のサブコードデータのバッファリングを行うことで、テキストデータの有無を判別することが可能になる。

【0089】

ところで、ステップ S014 において CRC が「NG」とされた場合は、図 17 において例えば破線で示されている結合子（3）に対応した、図 19 に示されているフローチャートの処理を行なうことも考えられる。

この図に示されているフローチャートの処理行程としては、ステップ S201 に示されてるようにパラメータ『M』が「0」であるか否かの判別を行ない、ここでパラメータ『M』が「0」であると判別した場合には、図 17 に示したステップ S023 に進み『NG数』の判定を行なうようにする。つまり、ステップ S201 でパラメータ『M』が「0」であった場合、まだスタートヘッドが検出されていない状態を示しており、したがって、再度スタートヘッドを検出する処理を行なうようにするため、ステップ S023 における NG 判定処理を経ることになる。

【0090】

また、パラメータ『M』が「0」ではないと判別した場合は、ステップS202に進みパラメータ『M』が「2」であるか否かの判別を行なう。ここで、パラメータ『M』が「2」ではないと判別した場合は、図17に示したステップS008に進みT O Cエリアに対するアクセス要求を行なう。一方、パラメータ『M』が「2」とであると判別した場合は、サブコードデータのバッファリング処理を終了して（S203）テキストデータの有効性を検証する処理（2）に進む。

【0091】

ステップ201及びステップS202の行程により、検出されているスタートヘッダの連続性が検出されるようになる。但し、図17に示したフローチャートのステップS019において、パラメータ『M』が例えば「2」または「2」以上とされた場合は、ステップS012を経てテキストデータの有効性を検証する処理（2）に進む行程を説明している。したがって、パラメータ『M』が「2」または「2」以上とされている場合、スタートヘッダの検出に伴って実行される処理行程として、図17のステップS019、S021を経て処理（2）に進むことになる。

【0092】

しかしながら、前述したようにステップS012以降の処理行程は随時行なわれており、現在実行している各判別処理などに対して並行した読み込み処理が行なわれているものとされる。これは、処理中のデータに対して予め次のデータの読み込みを行なう先読み処理に相当する。したがって、ステップS014においてC R Cの判定結果が「N G」とされた場合でも、ステップ201及びステップS202の処理行程において先読みによって連続したスタートヘッダが検出された状態では、当該スタートヘッダに対応したテキストデータに対して有効性の検証を行なうことが可能とされるようにしている。

なお、図19のフローチャートにおいて、テキストデータの有効性の検証を行なうまえにステップS203においてローサブコードデータのバッファリングを終了しているのは、C R C判定が「N G」とされていることに加え、先読みによって得られたテキストデータを利用すれば良いということから、これ以上バッファリング処理を継続しなくても良いという判断を下すことが可能とされるためである。

【0093】

図20はバッファリング処理によるキャッシュメモリの内容を説明する図であり、図20（a）は、図17のフローチャートで説明した各パラメータの数値例、図20（b）は、キャッシュメモリ20のアドレスに準じたデータ内容を模式的に示す図である。

【0094】

パラメータ『N』は、1フレーム分のサブコードデータがキャッシュメモリ20のサブコードエリア格納された場合にインクリメントされ、この図に示す例では初期値とされる「0」から「517」までの数値が示されている。即ち、図17におけるステップS012以降の処理が517回行なわれたことを示している。これに対応して、図20（b）に示すキャッシュメモリ20では、C a s h e # 1 ~ C a s h e # 5 1 7 が示されているが、パラメータ『N』に対応したエリアにはバッファリングされた1フレーム分のサブコードデータが格納されている。

【0095】

パラメータ『M』は、スタートヘッダが検出された時点でインクリメントされ、この図に示す例では『N』=「7」、及び『N』=「516」となった場合に、インクリメントされている。したがって、『N』=「7」の時点で第一のスタートヘッダが、また『N』=「516」の時点で第二のスタートヘッダが検出されたことを示している。

『N G数』は『N』=「1」の時点からインクリメントされ、『N』=「4」以降は『N G数』=「4」がホールドされている。即ち、『N』=「5」以降はC R C判定が「O K」とされていることを示している。

【0096】

『C o u n t 1』はパラメータ『M』が1回目にインクリメントされた時点でのパラメータ『N』の値が設定され、本例では「7」が設定されている。同様に『C o u n t 2』は

パラメータ『M』が2回目にインクリメントされた時点で、パラメータ『N』の値が設定され「516」が設定されている。

【0097】

このようなことから、図20(b)に示されているように、キャッシュメモリ20のローサブコードエリアでは、Cash#1～Cash#4がCRC判定により「NG」とされたデータとされ、『Count1』=「7」～『Count2』=「516」に相当する、Cash#7～Cash#516までの間にテキストグループが存在すると判別することができる。

即ち、これまでの処理が、図16乃至図19で示したフローチャートの処理行程によって行なわれる。

なお、Cash#7～Cash#516については、後に8ビット/6ビット変換されキャッシュメモリ20におけるローテキストエリア内に移されることになるが、この処理については後述で詳しく説明することにする。

【0098】

図17のフローチャートのステップS016乃至ステップS017においてスタートヘッダを検出する行程を説明したが、ここではその概要を説明する。

例えば1024フレーム分のローサブコードデータをバッファリングすることによって、その中からスタートヘッダを検出することも考えられるが、CD-TEXTとして記憶されているデータ容量が比較的小さい場合などは、処理効率が良いものではない。そこで、本例では、ローサブコードデータをローサブコードエリアに格納しつつ、並行して格納されているスタートヘッダの監視を行い、検出されたスタートヘッダの数が2以上である場合、その間に1個のテキストグループがあるという判断を下している。

【0099】

スタートヘッダのパターンとしては「80h, 00h, 00h, X0h」とされる4バイトのデータとされる。このデータ配列をビットで示すと

「80h」・・・10000000b

「00h」・・・00000000b

となる。なお4バイト目の「X0h」については、「00h」または「80h」のいずれかであることを示している。つまり、このヘッダパターンは「80h, 00h」から、図13で説明したようにアルバムタイトルヘッダであることがわかる。

【0100】

ローサブコードデータがローサブコードエリアに格納された状態では図21に示されているようになる。この図には、スタートヘッダを含んだP～Wに至る8ビットのデータを示している。このデータにおいて、図7(a)で説明したパックの先頭(ヘッダ領域)をチェックし、これらのパターンが2回以上検出された場合に、1個のテキストグループが検出されたとすることができる。

【0101】

6. 8ビット/6ビット変換

ローサブコードエリアに格納されたローサブコードは図21に示したように8ビット(P～W)とされているが、テキストデータは6ビット(R～W)のデータとされる。したがって、効率の良い処理を行なうためには、8ビットのデータ(P～W)のうちP、Qデータの2バイトを除去することが必要になる。そこで、後で図23で説明するフローチャートの処理行程とされる8ビット/6ビット変換処理を行なうようにしている。

【0102】

図22(a)は、図21に示したR～Wのデータ配列を、アルファベット(a～x)に対応させて示している。この状態ではR～Wのデータは6ビット単位とされる。このように配列されている8ビットのデータからP、Qデータ(X印)を除去して再び8ビットに配列し直すと、図22(b)に示されているようになる。つまり、キャッシュメモリ20のローテキストエリアには、バイト単位で扱うことができるテキストデータが格納される。

【0103】

10

20

30

40

50

この処理について、図 20 を参照して説明する。

図 20 (b) に示されている『C o u n t 1』～『C o u n t 2』とされるテキストグループを含んだデータは、図 22 (a) に対応した 8 ビット (P ~ W) のデータとされる。つまり、ローサブコードエリアには、不要な P、Q データも格納されている。そこで、8 ビット / 6 ビット変換を行なうことによって、『C o u n t 1』～『C o u n t 2』におけるテキストグループのみをローテキストエリアに移して、該ローテキストエリアにおいて、後述する例えばトラックタイトルヘッダやサイズインフォメーションヘッダの検証処理を行なうようにしている。

【0104】

ローサブコードエリアの他にローテキストエリアを設けるようにした場合、キャッシュメモリ 20 内に必要な領域が増えることになる。しかし、検証処理などを行なう場合、R ~ W データの 6 ビット毎の検証を行なうよりも、バイト単位で検証を行なう方が、システムコントローラ 10 の処理も単純化することができ、さらには、ホストコンピュータ 80 に対する転送処理も簡単になる。

【0105】

7. テキストデータ有効性検証処理

図 23、図 24 は、図 16、図 17 で説明したバッファリング処理に続くテキストデータ有効性検証処理の流れを説明するフローチャートを示す図である。なお、図 23 に示されているフローチャートの結合子 (4) 及び結合子 (5) は、それぞれ図 24 に示されているフローチャートの結合子 (4) 及び結合子 (5) に対応したものである。即ち、テキストデータ有効性検証処理としては図 23 乃至図 24 に示されている一連の処理行程によって実現される処理とされる。

【0106】

テキストデータ有効性検証処理としては、まず図 22 で説明した 8 ビット / 6 ビット変換を行ない、テキストデータに対応した 6 ビット (R ~ W) のデータをローテキストエリアに格納していく (S301)。そして、パックのヘッダ領域の各 ID (ID1 ~ ID4) を示すためのパラメータ『U』に「0」を設定し、さらにパラメータ『M』に「0」を設定する行なう (S302)。

【0107】

ところで、ステップ S301 における 8 ビット / 6 ビット変換が開始されると、ローテキストデータがローテキストエリアに順次格納されていくことになるが、ここで順次格納されてくるデータに対してスタートヘッダ (先頭パックのヘッダ領域) のサーチを行なう。スタートヘッダのパターンは、上述したように例えば「80h, 00h, 00h, X0h」とされている。これらのパターンはそれぞれ、図 9 (b) に示したヘッダ領域の ID1 ~ ID4 に相当する。したがって、格納されてくるローテキストデータにこのパターンが含まれているか否かを判別することで、スタートヘッダを認識することができる。

【0108】

スタートヘッダのサーチ処理としては、先頭の U 番目のデータ (ID1 に相当する) が 10000000b か否か (S303)、U + 1 番目 (ID2 に相当する) のデータが 00000000b か否か (S304)、U + 2 番目 (ID3 に相当する) のデータが 00000000b か否か (S305)、さらに U + 3 番目 (ID4 に相当する) のデータが 00000000b または 10000000b か否か (S306) の判別を行なう。

【0109】

ここでの判定結果として、いずれか一つでも一致しない場合は、パラメータ『U』のインクリメントを行ない (S313) 再びステップ S303 からステップ S306 におけるスタートヘッダサーチを行なう。このようにパラメータ『U』をインクリメントすることにより、前回 1 番目のデータを ID1 に相当するものとしていたが、今回は前回の 2 番目のデータ、即ち前回 ID2 に相当するとしていたデータを、ID1 としてサーチ処理を行なう。このように、被サーチデータをシフトさせていくことで、ローサブコードデータの中から所望するスタートヘッダパターンを検出するようにしている。

10

20

30

40

50

【 0 1 1 0 】

ステップS303からステップS306において、全てのデータのパターンがスタートヘッダパターンに一致すると、パラメータ『M』をインクリメントする（S307）。

即ちパラメータ『M』は先程と同様にスタートヘッダの検出回数を示すものとして用いられている。

最初にステップ（S307）に到達した時点では『M』= 1 とされ、これは第一のスタートヘッダが検出されたことを示す。したがって、ステップS308における『M』の値の判定は「NO」とされ、スタートヘッダに続く2番目のパックのヘッダパターンの検出処理に移行する（S309）。

【 0 1 1 1 】

ここで、2番目のパックのヘッダパターンの検出処理について説明する。

本例では、スタートヘッダに続くヘッダのタイプとして、例えばトラックタイトルヘッダまたはアルバムタイトルヘッダの検証を行なうこととしている。

図25はCD-TEXTフォーマットに規定されているヘッダタイプの一例を示している。即ち、図13の一部を詳細に示した図である。

図中（a）に示されている部分がアルバムタイトルヘッダとされており、先頭のID1が「80h」、続くID2は「00h」とされる。したがって、少なくともID1、ID2がこのようなパターンとされていた場合は、当該ヘッダはアルバムタイトルヘッダとすることができる。

【 0 1 1 2 】

また、図中（b）で示されている部分がトラックタイトルヘッダとされており、先頭のID1が「80h」とされ、続くID2については曲順を示す「01h」「02h」・・・「63h」とされている。即ち、少なくともID1及びID2がこれらの値とされているパターン（例えば1曲目を示す「80h, 01h」など）はトラックタイトルヘッダと見なすことができる。

そして、後述するようにアルバムタイトルヘッダ、またはトラックタイトルヘッダはパラメータ『M』= 1 とされたスタートヘッダが検出された次（順方向）のパックの先頭、即ち18バイト後から始まるテキストグループ内の2番目のパックのヘッダパターンとされる。したがって、第一のスタートヘッダとされているテキストグループ内の先頭パックのヘッダ及びこれに続く2番目のパックのヘッダとされる、2個のヘッダの内容によって第一のスタートヘッダが認識されるようにしている。

【 0 1 1 3 】

したがって、2番目のパックのヘッダパターンを検出する場合のフローチャートは図26に示されているようになる。但し、この図に示すフローチャートは、図23のステップS309の処理内容を示している。

ステップS401において、まず次のパックの先頭とされるU+18番目のデータ（ID1に相当する）と、これに続くU+19番目のデータが「80h, 01h」に一致しているか否かを判別し、各データがそれぞれ「80h, 01h」に一致していると判別した場合は、所要のヘッダ（トラックタイトルヘッダ）が検出されたと見なす（S402）。

【 0 1 1 4 】

また、各データが「80h, 01h」に一致していない場合は、U+18番目のデータ、U+19番目のデータ、及びこれに続くU+20番目のデータ（ID3に相当する）が、「80h, 00h, 01h」に一致しているか否かを判別する（S402）。ここで判別するヘッダタイプは、ID1、ID2による「80h, 00h」からアルバムタイトルヘッダとされる。また、ID3に示されている「01h」は上述したようにシーケンスナンバーが示されている。これは後述する図29に示されているが、先頭パックが「00h」、2番目のパックが「01h」・・・というように、ブロック内における当該パックの順位を示すことができるようになっている。

【 0 1 1 5 】

したがって、先頭パックのヘッダ（スタートヘッダ）としては「80h, 00h, 00h

10

20

30

40

50

」とされる。そして、記憶されているアルバムタイトルが比較的長いものとされ、例えば先頭パックと2番目のパックにまたがって記憶されている場合は、「80h, 00h, 01h」となる。したがって、2番目のパックの先頭はID1、ID2によるトラックタイトルヘッダ「80h, 01h」、またはID1、ID2、及びID3によるアルバムタイトルヘッダ「80h, 00h, 01h」となる。

したがって、各ヘッダがそれぞれ「80h, 00h, 01h」に一致している判別した場合は、所要のヘッダ(アルバムタイトルヘッダ)が検出されたと見なし(S403)、また、一致していないと判別した場合は所要のヘッダが検出されなかったとみなす(S404)。

【0116】

なお、パックのヘッダ領域はID1～ID4によって形成されるが、これら全てのデータを検出する必要はなく、説明したように少なくとも、ヘッダの判別が可能とされるIDの検出を行なうことができれば良い。

また、この図のフローチャートでは例えば、例えばトラックタイトルヘッダ、アルバムタイトルヘッダの検出を行なう例を挙げているが、先頭パックに続く2番目のパックのヘッダであることを検出することができれば、検出項目はいずれのものであっても良い。

【0117】

図23のフローチャートに戻り説明を続ける。

ステップS309を経た後、所要のヘッダが検出されたか否かを判別する(S310)。そして、所要のヘッダが検出された場合は、システムコントローラ10はU番目のデータが格納されているローテキストエリアのアドレスをスタートアドレス(Start Address)であると認識して記憶する(S311)。そして、ステップS313によってパラメータ『U』のインクリメントを行ない、再度スタートヘッダのサーチ処理を行なう。

【0118】

また、ステップS310で所要のヘッダが検出されなかったと判別した場合は、パラメータ『M』のデクリメントを行ない(S312)、パラメータ『M』を元の値に戻す。これは、第一のスタートヘッダについて、所要のヘッダが検出されなかったという結果に基づいて、再び第一のスタートヘッダのサーチを行なうためである。したがって、第一のスタートヘッダに対して所要のヘッダが検出されるまで、ステップS308からステップS309に進むようにされる。そして第一のスタートヘッダに対して所要のヘッダが検出され、パラメータ『M』が「2」となった時点で、結合子(4)に対応した処理に進む。

【0119】

パラメータ『M』が「2」となると図24に示されているように、システムコントローラ10はU番目のデータが格納されているローテキストエリアのアドレスをエンドアドレス(End Address)であると認識して記憶する(S314)。そして、エンドアドレスの値とスタートアドレスの値の差を求め、この差、即ち各アドレス間のデータ長と、テキストデータの最小単位として規定されているパックの長さ(Pack Length・・・18バイト)と比較する(S315)。ここでアドレス間のデータ長がパックの長さよりも長いと判別した場合は、次に、前記各アドレス間のデータ長が、1ブロックが最大256パックで形成された場合の最大データ長(Max. TEXT Length・・・36,864Kバイト)以下であるか否かを判別する(S316)。そして、前記各アドレス間のデータ長が最大データ長以下であると判別した場合は、サイズインフォメーションヘッダのチェックを行なう(S319)。

つまり、ステップS315、S316を経ることによって第一のスタートヘッダに続く第二のスタートヘッダが存在する見なすことができる。そして、さらに第二のスタートヘッダの存在を確認するために、この第二のスタートヘッダから所定量遡った位置に在る、サイズインフォメーションヘッダのチェック(S319)に移行する。

【0120】

なお、ステップS315において各アドレス間のデータ長がパックよりも短い場合や、またステップS316において各アドレス間のデータ長が前記最大データ長よりも長いと判別した場合、システムコントローラ10はステップS317に進み、エンドアドレスをヘッダの起点で

10

20

30

40

50

あると認識して記憶し、パラメータ『M』を初期値化「0」する処理を行なう（S318）。これは、第二のスタートヘッダが検出されなかったという結果に基づいて、再び第一のスタートヘッダのサーチを行なうためである。したがって、パラメータ『M』を「0」にした後は、結合子（5）を介して図23に示すステップS303に進む。

【0121】

この処理においては、図25（c）に示したサイズインフォメーションヘッダ#1～#3の検出を行なう。

ところで、サイズインフォメーションヘッダ（#1～#3）としては、図25（c）に示されているように先頭のID1が「8Fh」とされ、続くID2はそれぞれ「00h」「01h」「02h」とされている。即ち、少なくともID1及びID2がこれらの値とされているパターンはサイズインフォメーションヘッダと見なすことができる。また、このサイズインフォメーションは、テキストグループの最後の3個のバックのヘッダとされているので、本例では第二のスタートヘッダから遡って検出するようにしている。

【0122】

したがって、サイズインフォメーションヘッダチェック（S319）の処理としては、例えば図27のフローチャートに示されているようにされる。

まずステップS501において、最後のバックの先頭とされるU-18番目のデータ（ID1に相当する）と、これに続くU-17番目のデータ（ID2に相当する）が「8Fh, 02h」に一致しているか否かを判別し、各データがそれぞれ「8Fh, 02h」に一致していると判別した場合はステップS502に進む。

ステップS502では、最後から2番目のバックの先頭とされるU-36番目のデータ（ID1に相当する）と、これに続くU-35番目のデータ（ID2に相当する）が「8Fh, 01h」に一致しているか否かを判別し、各データがそれぞれ「8Fh, 01h」に一致していると判別した場合はステップS503に進む。

そして、ステップS503では、最後から3番目のバックの先頭とされるU-54番目のデータ（ID1に相当する）と、これに続くU-53番目のデータ（ID2に相当する）が「8Fh, 00h」に一致しているか否かを判別する。

【0123】

ステップS501からステップS503を経ることにより、各データがそれぞれ所望するデータ内容に一致していた場合はステップS504に進み、3個のサイズインフォメーションヘッダが検出されたと見なす。

また、ステップS501からステップS503において、一つでも内容が一致しないものが在れば、不一致が検出された時点でステップS505に進みサイズインフォメーションヘッダが検出されていないと見なす。

【0124】

このようにして得られたサイズインフォメーションの検出結果に基づいて、図24に示すステップS319以降の処理を行なう。ステップS319を経た後は、まずサイズインフォメーションヘッダが検出されたか否かの検出を行ない（S320）、ここで、サイズインフォメーションヘッダが検出された場合は、所望するテキストデータが得られたと見なし、図16、図17によって開始されているバッファリング処理を終了して（S321）、ステップS311で認識したスタートアドレスとステップS314で認識したエンドアドレスの間のテキストデータをホストコンピュータ80に転送する（S322）。

【0125】

再生装置70としては、上記したような処理によりテキストデータの誤検出を低減することができるので、ホストコンピュータ80に対しては精度の高いテキストデータを転送することができる。そして、ホストコンピュータ80では、ディスク90から読み出された精度の良いテキストデータを利用して所要の表示処理を行なうことができるようになる。

【0126】

8. データ例

図28はバッファリング処理を経て、キャッシュメモリ20のローサブコードエリアに格納

10

20

30

40

50

されたローサブコードデータの一例を示す図である。

この図に示すローサブコードデータは、CRCのNG数が例えば9回、パファリングフレーム数(パラメータ『N』)が例えば100回とされるデータである。したがって、パファリングされたデータの先頭から $9 \times 96 = 864$ バイトは無効とされる。即ち、アドレス0から863(dec i)までのデータは図20(b)に示したCRC__NG領域に相当する。したがって、9フレーム以降からCRCが「OK」とされている。

また、スタートヘッダとしては864番目からP、Qデータを含んだ第一のスタートヘッダパターン(20h, 00h, 00h, 00h・・・実線で囲む)が示され、これは例えば『Count 1』に対応したデータとされる。そして、1944番目から同じくP、Qデータを含んだ第二のスタートヘッダパターン(20h, 00h, 00h, 00h)が示されているが、これが『Count 2』に対応したデータとされる。したがって、この第一、第二のスタートヘッダの間に、所望するテキストデータが存在していると思われ、テキストデータ有効性検証処理を行なう。

【0127】

図29はパファリング処理、テキストデータ有効性検証処理を経て、ローテキストエリアからホストコンピュータ80に転送されるローテキストデータの一例を示す図である。

この図のデータの先頭に実線で囲んだスタートヘッダに相当するアルバムタイトルヘッダ「80h, 00h, 00h, 00h」が示され、このスタートヘッダパターンの次に、下線によりアルバムタイトルヘッダ「80h, 00h, 01h」が示されていることが解る。これは、上記したようにスタートヘッダパターンから数えて18バイト後に位置している。したがって、実線で囲んだ先頭からのデータが第一のスタートヘッダパターンとすることができる。

【0128】

なお、この図に示すデータ例では、アルバムタイトルが2つのパックにまたがっているため、1曲目のトラックタイトルヘッダ「80h, 01h」は3番目のパックのヘッダとされる。したがって、ID3のシーケンスナンバは「02h」とされている。つまり、当該ディスク90のアルバムタイトルが1パックに収まる比較的短い場合は、2番目のパックが1曲目のタイトルに対応し、ID1～ID3までの内容は、「80h, 01h, 01h」となる。

【0129】

また同じく実線で囲んだ811番目から実線で囲んだスタートヘッダパターンが示され、このスタートヘッダパターンを所定量遡った位置に、サイズインフォメーションヘッダパターン(「8Fh, 02h」、「8Fh, 01h」、「8F, 00h」)が示されていることが解る。したがって、811番目からのデータが第二のスタートヘッダパターンとすることができる。

【0130】

したがって、これら2個のスタートヘッダに囲まれたデータが、所望するテキストデータとされる。即ち、図29に示す例では、各スタートヘッダ(「80h, 00h, 00h, 00h」)の間のデータが、ローテキストエリアからホストコンピュータ80に転送されるようになる。

【0131】

9. 変形例

以下、本発明の変形例を説明する。

上記実施の形態では、例えば2個の連続したスタートヘッダを検出して、各スタートヘッダの間の所定のデータ内容の検証を行なう例としたが、例えば図30の模式図に示されているように2個以上のスタートヘッダH1、H2、H3・・・を検出して、これらのスタートヘッダがCD-TEXTのフォーマットとして規定されているパターンであった場合にテキストデータの有効性を確認するようにしても良い。

また、例えば2個以上のスタートヘッダを検出した場合、スタートヘッダH1、H2に囲まれたデータの内容(テキストデータ(1))と、スタートヘッダH2、H3に囲まれた

10

20

30

40

50

データの内容（テキストデータ（２））を比較して、双方のデータ内容が一致していることを確認した場合には、テキストデータ（（１）及び（２））の有効性を確認するようにしても良い。

【０１３２】

また、上記実施の形態ではバッファリング処理によって検出された２個のスタートヘッダの間に在る、アルバムタイトルヘッダ、トラックタイトルヘッダやサイズインフォメーションヘッダの内容が所定のものとされていれば、当該トラックタイトルヘッダやサイズインフォメーションヘッダに対応したテキストデータの有効性を確認したが、他のヘッダ内容によっても有効性の確認を行なうことができる。

【０１３３】

例えば、ヘッダ（図２９の実線で囲んだ部分に相当する）の内容チェックとして、最初の１バイト（ＩＤ１：Ｐａｃｋ＿Ｔｙｐｅ＿Ｉｎｄｉｃａｔｏｒ）をチェックする場合を例に挙げる。ここで、第一のスタートヘッダがＵ番目にバッファリングされ、第二のスタートヘッダがＵ＋Ｎ番目に在るとする。そして、検証対象とする任意のヘッダがＵ＋Ｍ番目（但し、 $M < N$ ）とした場合、処理の流れとしては図３１のフローチャートに示されているようになる。

【０１３４】

図３１のに示されているように、まず、ＩＤ１が「８０ｈ」であるか否かを判別する（Ｓ６０１）。ここで、ＩＤ１が「８０ｈ」であると判別した場合は、さらにＩＤ２が「０」以上かつ「６３」以下であるか否かを判別する（Ｓ６０２）。そして、ＩＤ２が「０」以上かつ「６３」以下であったと判別した場合は、図１３に示したように、アルバムタイトルヘッダまたはトラックタイトルヘッダであるを見なすことができ、テキストデータの有効性を確認することができる（Ｓ６１５）。

また、ＩＤ２が「０」以上かつ「６３」以下ではないと判別した場合は、即ちＣＤ－ＴＥＸＴのフォーマットで規定されている値ではない場合には、テキストデータとしての有効性を確認しないで、無効なデータであると認識する（Ｓ６１６）。

【０１３５】

以降、同様にして、ステップＳ６０３からステップＳ６１４に至るまで、各ステップ毎にＩＤ１の内容が「８１ｈ」から「８Ｆｈ」のいずれかに一致しているか否かの判別を行なう。なお、「８１ｈ」から「８Ｆｈ」についてのデータ内容（識別内容）は、図１３に示したものとされる。

このようにしてステップＳ６０３～ステップＳ６１４においての判別結果が、ＣＤ－ＴＥＸＴのフォーマットで規定されている値に一致していない場合は、ＩＤ１に対応したテキストデータは無効なデータであると認識する（Ｓ６１７）。一方、ステップＳ６０３～ステップＳ６１４において、一つでも「８１ｈ」から「８Ｆｈ」に一致しているものがあったと判別した場合には、その時点でテキストデータの有効性を確認することができる（Ｓ６１５）。

なお、この図に示すフローチャートでは、一つの任意のヘッダとして例えばＩＤ１（Ｐａｃｋ＿Ｔｙｐｅ＿Ｉｎｄｉｃａｔｏｒ）のチェックを行なう例を挙げたが、他のヘッダ（ＩＤ２、ＩＤ３、ＩＤ４）について同様の処理を行なうようにしても良い。

【０１３６】

また、ローサブコードエリアに対してローサブコードデータをバッファリングした時点で、バックの先頭のＩＤを全てスタートヘッダ（テキストグループの先頭のヘッダ）であるか否かを検出することができるハードウェアを構成することにより、システムコントローラ１０の処理負担が軽減され、バッファリングを行なうデータ量を必要最小限に低減することができる。

【０１３７】

なお、上記実施の形態は、テキストデータがディスク９０のＴＯＣ領域に記憶されている「モード４」を例に挙げて説明したが、テキストデータがプログラムエリア（主データのエリア）に記憶されている「モード２」に対しても適用することができる。

【０１３８】

【発明の効果】

以上、説明したように本発明は、例えばオーディオＣＤの一種とされるＣＤ－ＴＥＸＴに記憶されている、テキストデータ（文字情報）を構成するテキストグループの先頭にあるスタートヘッダ（先頭パックのヘッダ）を検出するようにしている。これにより、当該テキストグループの開始位置、終了位置を検出することができる。そして、前記スタートヘッダ間にある所定のデータ内容の検証や比較を行なうことにより、前記スタートヘッダ間のデータに対して、テキストデータとしての有効性を確認することができ、テキストデータの誤検出を低減することができるようになる。

【０１３９】

例えば、前記スタートヘッダが検出されるとパックが形成されていることを想定することができる。つまり、前記スタートヘッダの間に在るデータに対して、パックのヘッダに相当するデータ内容を検証し、この検証結果に基づいて前記スタートヘッダ間のデータを前記テキストデータとして認識することができるようになる。したがって、第一のスタートヘッダから順方向における所定の位置にあるパックのヘッダ（例えばアルバムタイトルヘッダ、トラックタイトルヘッダ）と、第二のスタートヘッダから逆方向における所定の位置にあるパックのヘッダ（例えばサイズインフォメーションヘッダ）のパターンを検出することにより、テキストデータの有効性を確認することができる。

つまり、前記スタートヘッダ間の所要の位置にＣＤ－ＴＥＸＴフォーマットに規定されているパックのヘッダに対応したデータパターンを検出することにより、テキストデータの有効性確認が可能となる。

【０１４０】

また、スタートヘッダ間における例えばヘッダ（ＩＤ１・・・パックインジケータ）などの所定の位置にあるデータ内容が、規定されている所定のデータ内容に一致している場合にも、テキストデータの有効性を確認することができる。つまり、読み込まれたデータと規定されているデータの内容の一致、不一致を判定することにより検証結果を得ることができるので、テキストデータの有効性確認の精度を向上することができる。

【０１４１】

また、複数の連続したスタートヘッダを検出するとともに、これらのスタートヘッダの間にある複数のテキストグループに相当するデータ内容を比較して、この比較結果が一致していた場合は、当該テキストグループに相当するテキストデータの有効性を確認することができる。つまり、テキストデータ全体を比較しているので、データの整合性に基づいた有効なテキストデータを得ることができるようになる。

【０１４２】

さらに、スタートヘッダが所定回数検出された時点で、管理情報の読み込みを終了させて、検出されているスタートヘッダに対応した文字情報の検証処理に移行することで、必要以上のデータの読み込み動作を抑制することができ、システムコントローラにかかる負担を抑制することができるようになる。

【０１４３】

また、上記したテキストデータの検証処理を行なう場合、管理情報から文字情報とされるテキストデータのみを抽出して行なうことにより、必要最小限のデータによって検証処理を行なうことができ、処理自体が簡単になるとともに、検証結果が有効であった場合についても、ホストコンピュータに対する転送処理が簡単になる。

【０１４４】

さらに、ディスクからインターリーブされた状態で読み出されたデータ形態の管理情報を、デインターリーブされる前に検出するので、デインターリーブ処理を省略することができるようになる。

【０１４５】

また、前記テキストデータに対する誤り検出符号に基づいて、当該テキストデータに誤りが検出されたときに、スタートヘッダが検出されないまま、それまでに前記テキストデータの誤りが所定回数以上検出されていた場合には、ディスクにテキストデータが記録され

10

20

30

40

50

ていないと判断するようにしている。したがって、必要以上のデータの読み込み動作を抑制したうえで、オーディオＣＤとＣＤ－ＴＥＸＴの識別を行うことができるようになる。

【０１４６】

さらに、ディスクにテキストデータが記録されていないと判断した時点で、ディスクに記録されている情報の読み出しを行うバッファリング処理を終了させることで、システムコントローラにかかる負担を抑制することができるようになるという利点がある。

【図面の簡単な説明】

【図１】本発明の実施の形態の再生装置の構成を説明するブロック図である。

【図２】ディスク（ＣＤ）のフレーム構造の説明図である。

【図３】ディスク（ＣＤ）のサブコーディングの説明図である。

10

【図４】ディスク（ＣＤ）のサブＱデータの説明図である。

【図５】ディスク（ＣＤ）のＴＯＣデータの説明図である。

【図６】テキストデータの構造を包括的に示す説明図である。

【図７】サブコーディングフレームとテキストデータとの構造的な関係を示す説明図である。

【図８】テキストデータとしてパケットの構造を示す説明図である。

【図９】テキストデータの構造として、シンボル単位のデータからパックを形成する過程を説明するための説明図である。

【図１０】パックの構造を示す説明図である。

【図１１】ＩＤ１の構造を示す説明図である。

20

【図１２】ＩＤ１～ＩＤ４の構造をそれぞれ示す説明図である。

【図１３】ＩＤ１の定義内容を示す説明図である

【図１４】テキストデータとしてトラックの曲名を格納する場合のパックの構造を示す説明図である。

【図１５】再生処理時のキャッシュメモリのマッピング例を説明する図である。

【図１６】バッファリング処理の行程を説明するフローチャートである。

【図１７】図１７に続くバッファリング処理の行程を説明するフローチャートである。

【図１８】図１７のフローチャートに対応して、ＮＧが検出されている場合にサブコードデータのバッファリングを終了させる処理に進む流れを説明するフローチャート。

【図１９】ＮＧが検出されている場合にサブコードデータのバッファリングを終了させ、有効性を検出する処理に進む流れを説明するフローチャート。

30

【図２０】再生処理時のキャッシュメモリのローサブコードエリアの内容を模式的に説明する図である。

【図２１】キャッシュメモリのローサブコードエリアに格納されているローサブコードデータを説明する図である。

【図２２】８ビット／６ビット変換の概要を説明する図である。

【図２３】テキストデータ有効性検証処理の行程を説明するフローチャートである。

【図２４】図２２に続くテキストデータ有効性検証処理の行程を説明するフローチャートである。

【図２５】ヘッダタイプの一例を説明する図である。

40

【図２６】２番目のパックのヘッダの検証処理を説明するフローチャートである。

【図２７】サイズインフォメーションの検証処理を説明するフローチャートである。

【図２８】ローサブコードデータのデータ例を説明する図である。

【図２９】テキストデータのデータ例を説明する図である。

【図３０】本発明の変形例を説明する模式図である。

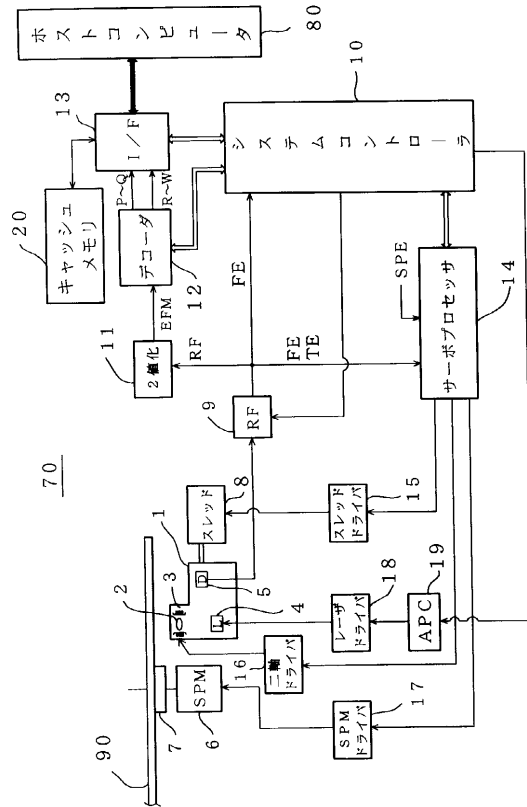
【図３１】本発明の他の変形例を説明するフローチャートである。

【符号の説明】

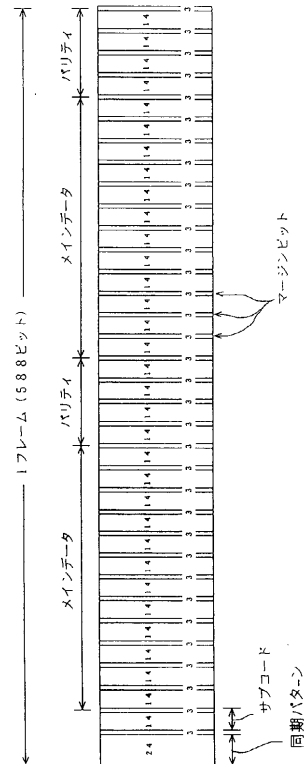
１ ピックアップ、９ ＲＦアンプ、１０ システムコントローラ、１２ デコーダ、１３ インターフェース部、１４ サーボプロセッサ、２０ キャッシュメモリ、７０ 再生装置、８０ ホストコンピュータ、９０ ディスク

50

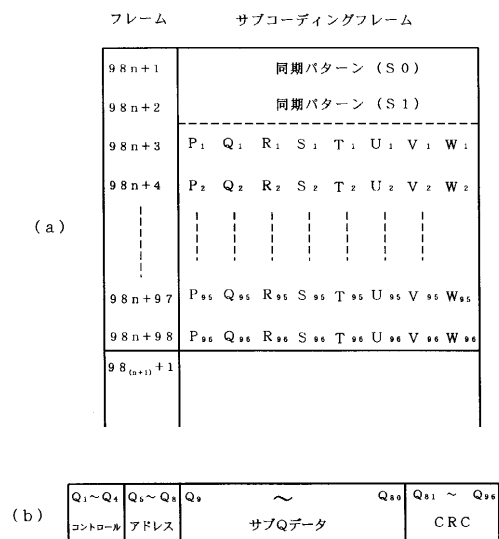
【 図 1 】



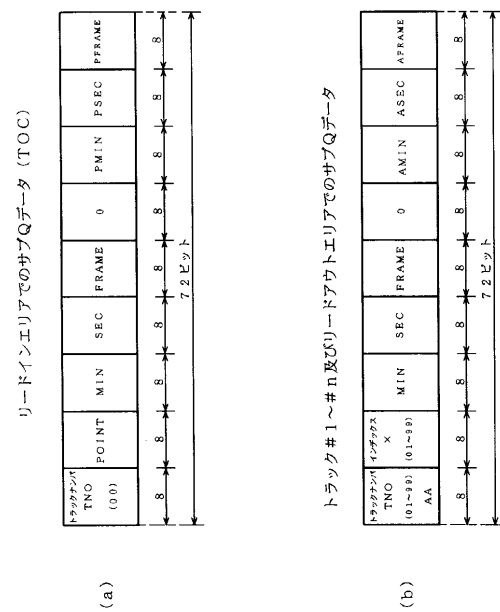
【 図 2 】



【 図 3 】



【圖 4】

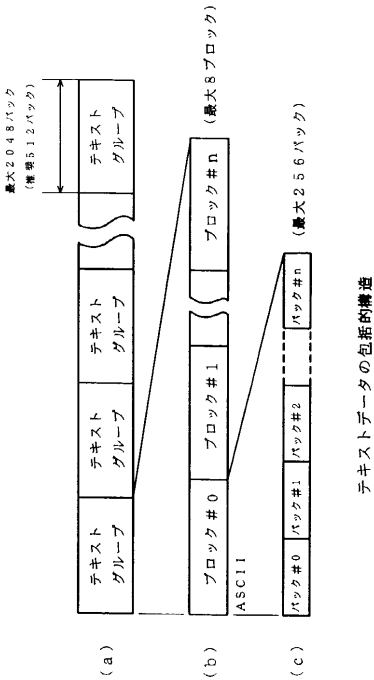


【図5】

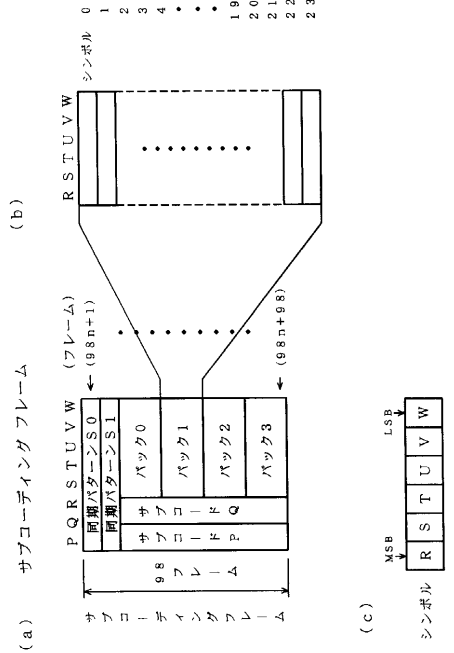
TOC構成 (6トラック入ディスクの例)

TNO	ブロックNo.	POINT	PMIN, PSEC, PFRAME	
00	n	01	00. 02. 32	トラック#1の
	n+1	01	00. 02. 32	スタートポイント
	n+2	01	00. 02. 32	
	n+3	02	10. 15. 12	トラック#2の
	n+4	02	10. 15. 12	スタートポイント
	n+5	02	10. 15. 12	
	n+6	03	16. 28. 63	トラック#3の
	n+7	03	16. 28. 63	スタートポイント
	n+8	03	16. 28. 63	
	n+9	04	.	
	n+10	04	.	
	n+11	04	.	
	n+12	05	.	
	n+13	05	.	
	n+14	05	.	
	n+15	06	49. 10. 03	トラック#6の
	n+16	06	49. 10. 03	スタートポイント
	n+17	06	49. 10. 03	
	n+18	A0	01. 00. 00	ディスクの最初のトラック
	n+19	A0	01. 00. 00	のトラックナンバ
	n+20	A0	01. 00. 00	
	n+21	A1	06. 00. 00	ディスクの最後のトラック
	n+22	A1	06. 00. 00	のトラックナンバ
	n+23	A1	06. 00. 00	
	n+24	A2	52. 48. 41	リードアウトトラックの
	n+25	A2	52. 48. 41	スタートポイント
	n+26	A2	52. 48. 41	
00	n+27	01	00. 02. 32	くり返す
00	n+28	01	00. 02. 32	
	.	.	.	
	.	.	.	

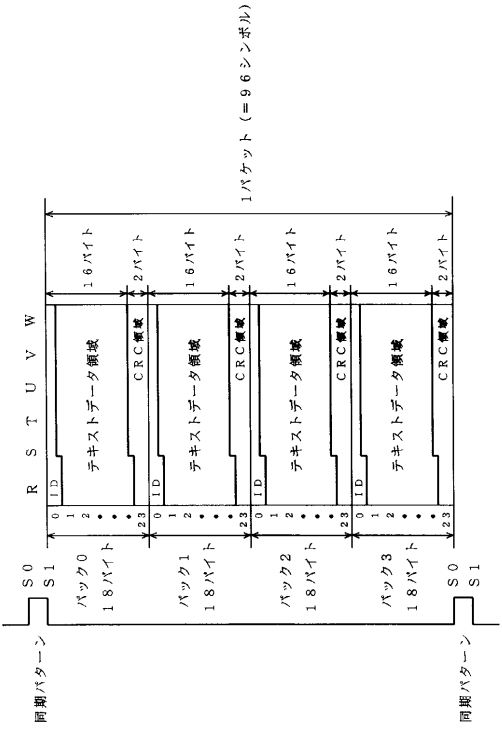
【図6】



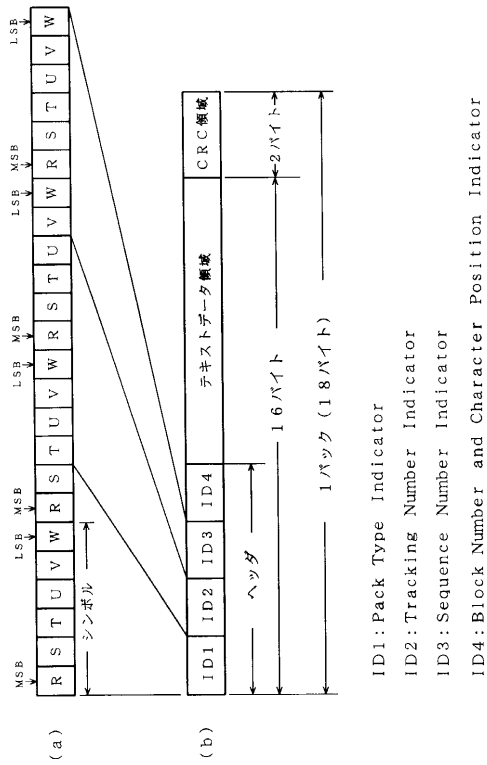
【図7】



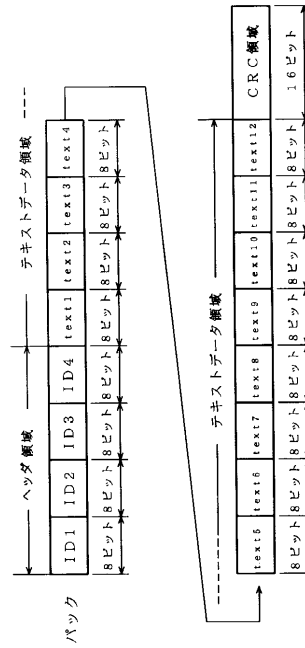
【図8】



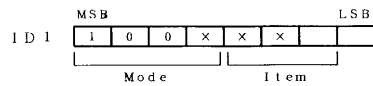
【図 9】



【図 10】



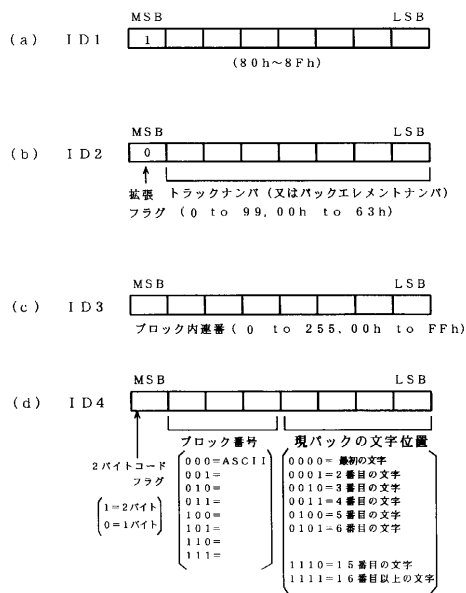
【図 11】



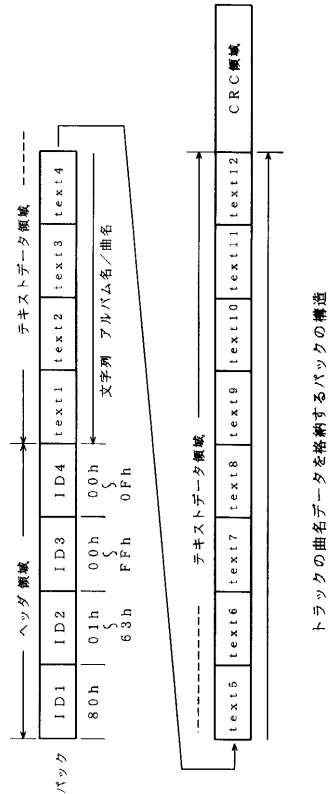
【図 13】

ID1	識別内容
80h	アルバム名 (ID2=00h) / トラックの曲名 (ID2=01h~63h)
81h	演奏者/指揮者/オーケストラ名
82h	作詞者名
83h	作曲者名
84h	編曲者名
85h	メッセージ
86h	ディスクID
87h	ジャンル
88h	TOC
89h	2nd TOC
8Ah	予約
8Bh	予約
8Ch	予約
8Dh	管理
8Eh	CPC/EAN/ISRC
8Fh	サイズインフォメーション

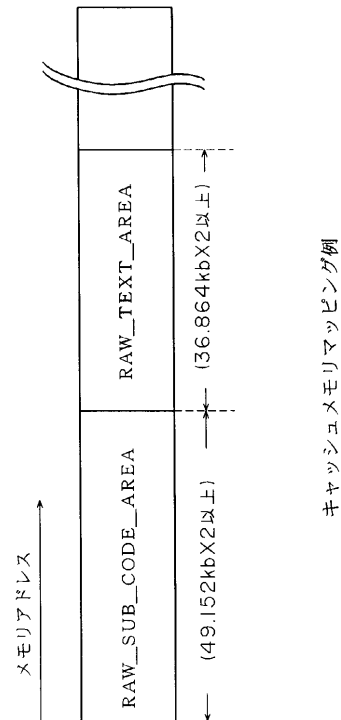
【図 12】



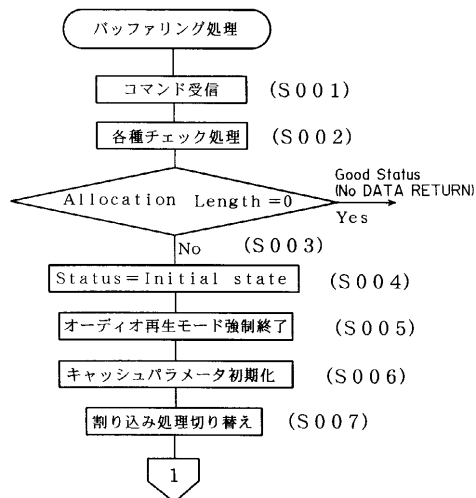
【図 14】



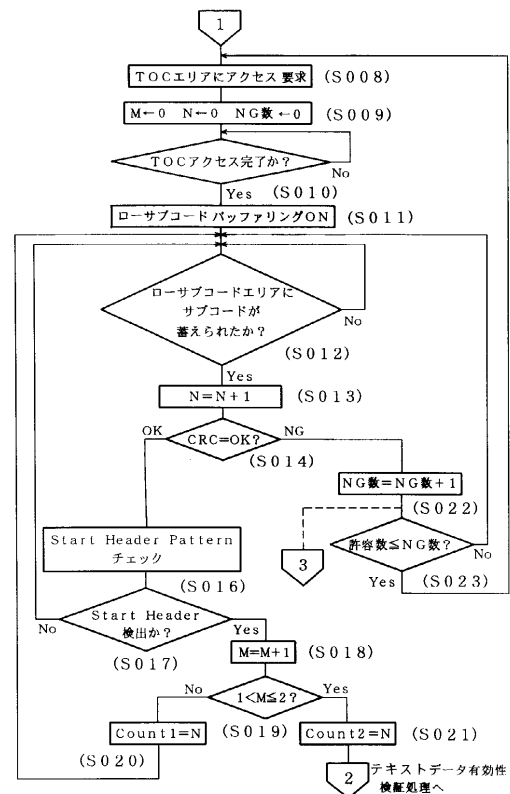
【図 15】



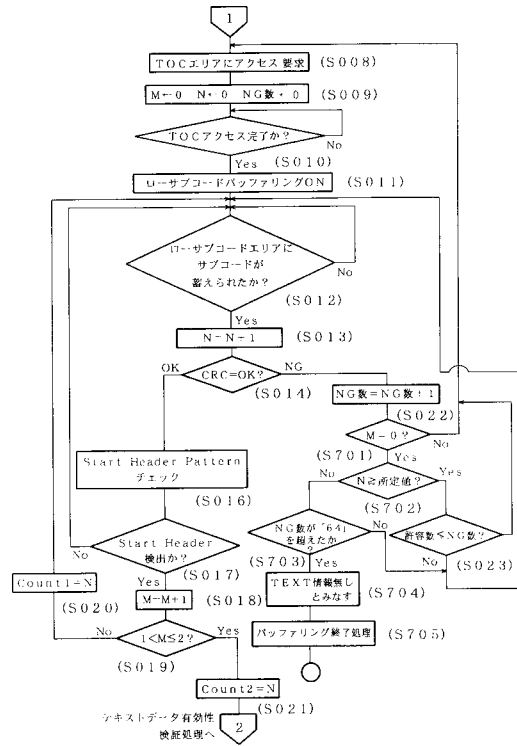
【図 16】



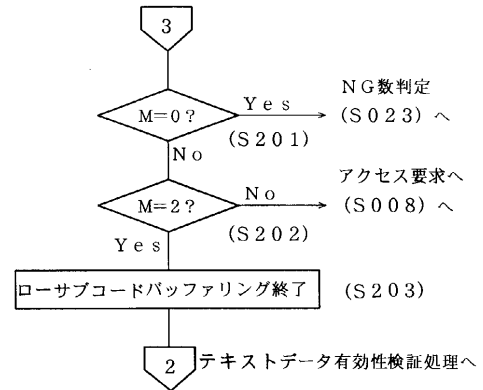
【図 17】



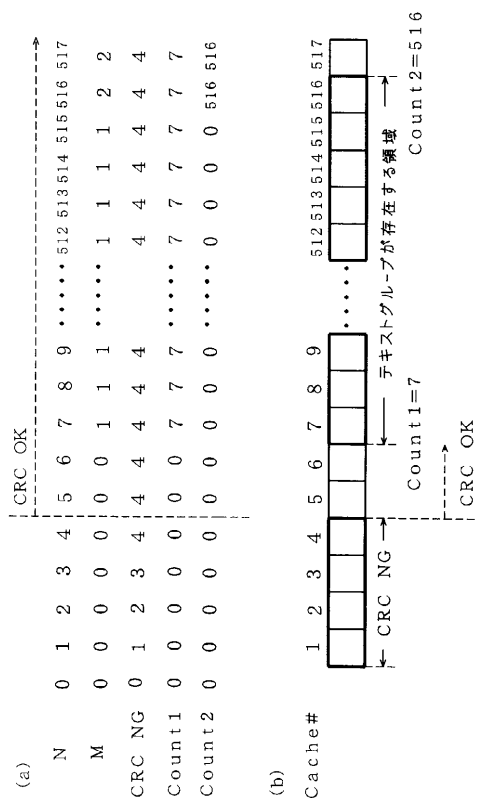
【図18】



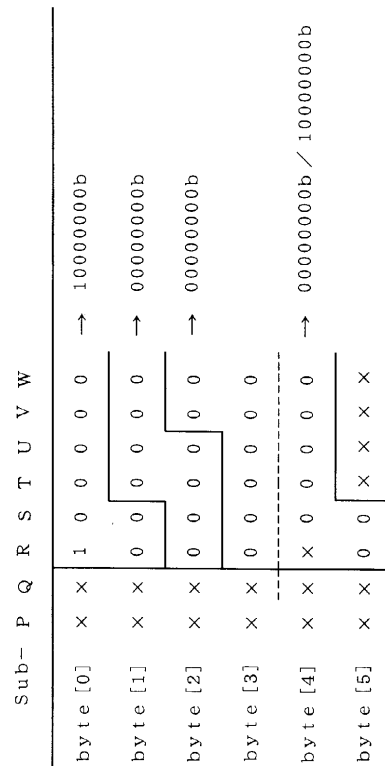
【図19】



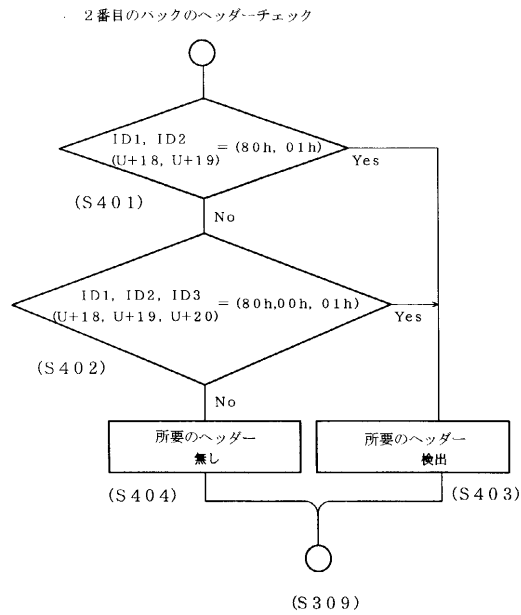
【図20】



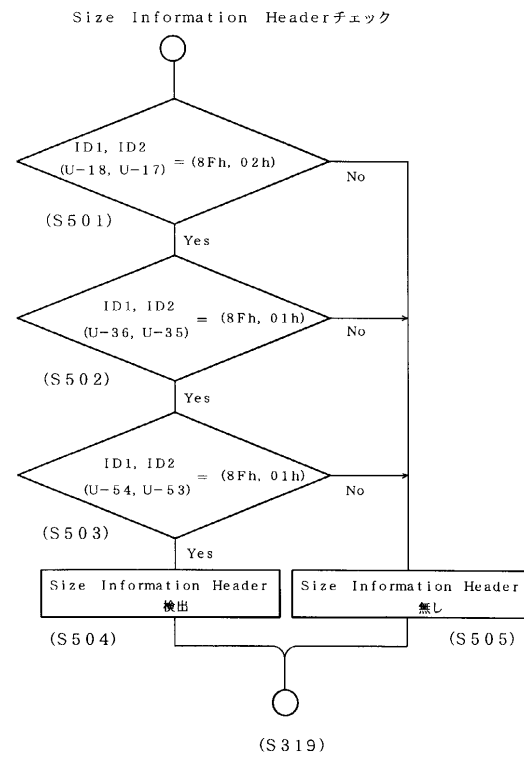
【図21】



【図 26】



【図 27】



【図 28】

Hex	dec1	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	0123456789ABCDEF
0000h	00000	20	00	18	09	00	05	00	54	10	15	11	05	08	04	30	06T.....=
0001h	00016	08	05	00	08	53	33	36	20	20	00	18	04	03	04	00	0BS36.....M
0002h	00032	00	05	01	41	14	65	51	59	08	44	45	0C	13	45	53	14A.edY.DE..ES
0003h	00048	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00	00
0036h	00064	20	00	00	00	00	03	20	70	09	37	0C	20	10	14	31	09p.7.....1
0037h	00080	15	24	14	60	0C	29	2F	73	20	00	01	03	02	40	20E
0038h	00096	08	05	09	45	51	02	00	20	00	05	11	05	13	0D	27	32E0.....'2
0039h	000112	20	00	04	02	00	34	30	20	12	44	15	12	08	04	05	4240.....B
003Ah	000128	13	35	55	14	08	4A	60	20	20	00	04	03	34	25	145U..J'.....4%
003Bh	000144	00	04	11	41	53	24	50	45	14	22	41	1A	13	7D	20	44ASSJ[E."A..]-D
003Ch	000160	20	00	08	04	02	14	39	45	00	04	21	0F	13	04	11	099E.....I
003Dh	000176	13	24	1C	60	13	3E	07	66	20	00	0C	05	02	15	55	14>.f.....U
003Eh	000192	08	04	19	4F	54	22	01	01	08	04	21	05	14	26	12	4F0T'.....I.&.0
003Fh	000208	20	00	0C	06	03	34	3C	00	12	44	05	1A	16	12	01	534<.....D.....S
0040h	000224	12	04	05	04	11	5D	46	16	20	00	10	07	02	22	01	0FE.]F....."
0073h	001840	10	10	01	43	16	5A	1E	01	20	11	04	29	00	24	79	04C.Z.....).\$y
0074h	001856	12	12	01	4C	50	15	15	10	11	15	08	40	00	05	19	09LP.....E....
0075h	001872	23	30	00	2A	00	00	04	01	04	50	0C	58	04	60	40	00	#0.*.....P.X.'e
0076h	001888	00	00	00	40	00	47	1E	2F	23	30	44	2B	00	40	00	40E.G./#0D+.e.e
0077h	001904	00	00	00	00	00	40	03	08	00	00	00	00	04	77	01	e.....e.....w	
0078h	001920	23	30	08	2C	00	00	00	40	00	00	09	00	00	00	00	#0.....e.....	
0079h	001936	00	00	00	00	40	01	05	05	20	00	00	00	00	03	60	30e.A.....'0
007Ah	001952	09	37	0C	60	50	14	31	09	15	24	14	60	0C	29	2F	73P..I..\$.>)/s
007Bh	001968	20	00	00	01	03	02	00	20	08	45	09	45	11	42	40	20E.E.D.....
007Ch	001984	00	05	11	45	13	40	27	32	20	00	44	02	00	74	30	60E.M'2.D..t0'
007Dh	002000	12	04	55	12	48	44	05	02	13	75	15	54	08	0A	60	60U.HD...u.T...m

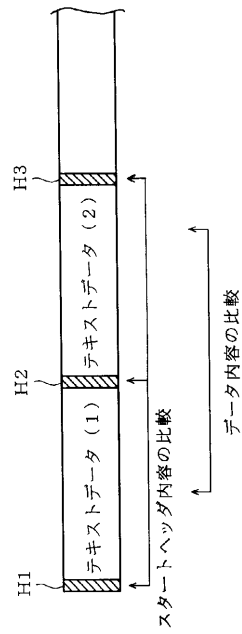
RAW SUB CODE DATA

【図 29】

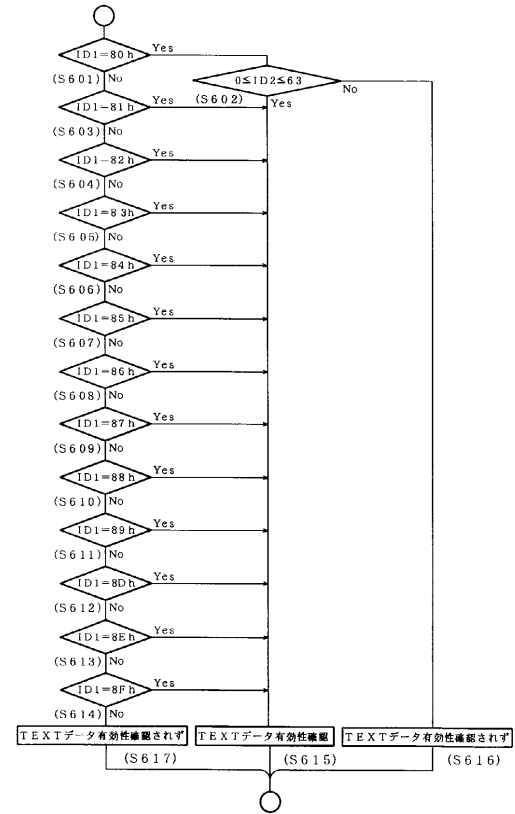
Hex	dec1	+0	+1	+2	+3	+4	+5	+6	+7	+8	+9	+A	+B	+C	+D	+E	+F	0123456789ABCDEF
0000h	00000	80	00	00	00	38	30	27	73	20	41	4C	49	56	45	20	3280's ALIVE 2
0001h	000016	9B	F3	80	00	01	0C	20	2D	20	52	45	44	20	20	00	54-RED~.T
0002h	000032	45	4C	D9	F2	80	01	02	03	4C	20	48	45	52	20	41	42L HER AB
0003h	000048	4F	55	54	20	A8	2D	80	01	03	0F	49	54	00	44	41	4EIT.DAN
0004h	000064	47	45	52	20	5A	4F	D8	44	80	02	04	09	4E	45	00	48GER ZO....NE.H
0005h	000080	4F	4C	44	49	4E	47	20	4F	E1	E6	80	03	05	09	55	54OLDING 0.....UT
0006h	000096	20	46	4F	52	20	41	20	48	45	52	64	8F	80	03	06	0FFOR A HERd....
0007h	000112	4F	00	48	41	5A	59	20	53	48	41	44	45	D1	96	80	04HAZY SHADE....
0008h	000128	07	0A	20	4F	46	20	57	49	4E	54	45	52	00	4C	7D	5DOF WINTER.U]
002Bh	000688	48	45	41	50	20	54	52	49	43	4B	00	42	05	67	81	0FHEAP TRICK.B.g..
002Ch	000704	27	01	45	52	54	49	45	20	48	49	47	47	49	4E	AB	D8ERTIE HIGGIN..
002Dh	000720	81	0F	28	00	53	00	4D	41	52	54	49	4B	41	00	43	59S.MARTIKA.CY
002Eh	000736	A7	81	11	29	02	4E	44	49	20	4C	41	55	50	45	52).NDI LAUPER	
002Fh	000752	00	00	56	49	8F	00	2A	00	01	01	11	03	18	12	00	00V.I.*.....
0030h	000768	00	00	00	00	77	AF	8F	01	2B	00	00	00	00	00	00	00w.....
0031h	000784	00	03	2C	00	00	00	4D	C1	8F	02	2C	00	00	00	00	00M.....
0032h	000800	09	00	00	00	00	00	00	00	11	45	80	00	00	00	38	30E.....80
0033h	000816	27	73	20	41	4C	49	56	45	20	32	9B	F3	80	00	01	0Cs ALIVE 2.....
0034h	000832	20	20	20	52	45	44	20	20	00	54	45	4C	D9	F2	80	01-RED~.TEL....
0035h	000848	02	03	4C	20	48	45	52	20	41	42	4F	55	54	20	A8	2DL HER ABOUT ..

CD-TEXT DATA

【図 30】



【図 31】



フロントページの続き

(58)調査した分野(Int.Cl. , D B 名)

G11B 27/00

G11B 20/10

H04N 5/76

H04N 5/91