(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第3653315号
(P3653315)
(45)発行日　平成17年5月25日(2005.5.25)　(24)登録日　平成17年3月4日(2005.3.4)

(51)Int.Cl.[7]　F I
H03M 13/17　H03M 13/17

請求項の数 6　(全 23 頁)

(21)出願番号　特願平7-265617
(22)出願日　平成7年10月13日(1995.10.13)
(65)公開番号　特開平9-107294
(43)公開日　平成9年4月22日(1997.4.22)
審査請求日　平成13年3月30日(2001.3.30)
審判番号　不服2002-8174(P2002-8174/J1)
審判請求日　平成14年5月9日(2002.5.9)

(73)特許権者　000002185
ソニー株式会社
東京都品川区北品川６丁目７番３５号
(74)代理人　100122884
弁理士　角田　芳末
(74)代理人　100113516
弁理士　磯山　弘信
(74)代理人　100080883
弁理士　松隈　秀盛
(72)発明者　荒瀧　裕司
東京都品川区北品川６丁目７番３５号　ソニー株式会社内
(72)発明者　小川　雅人
東京都品川区北品川６丁目７番３５号　ソニー株式会社内

最終頁に続く

(54)【発明の名称】エラー訂正方法及びエラー訂正装置

(57)【特許請求の範囲】
【請求項１】
入力データ列が所定のワード数から成るシンボルに分割され、上記各シンボルに対して記録媒体からの読出方向に対して傾斜した方向に配列された第１のデータ系列に対する第１のチェックワードを生成し、上記各シンボルと上記第１のチェックワードに対して記録媒体からの読出方向と同一方向に配列された第２のデータ系列に対する第２のチェックワードを生成し、上記各シンボルと上記第１のチェックワードと上記第２のチェックワードから構成される入力データ列を復号化するエラー訂正方法は、
上記各ワードと上記第１のチェックワードに対して各々異なる時間遅延を施し、記録媒体からのデータの読出し方向に対して傾斜した方向にデ・スクランブル処理を施す第１のデ・スクランブル処理工程と、
上記第１のデ・スクランブル処理が施された上記各ワードに対して上記第１のチェックワードを用いてロケーション方程式を演算する第１の演算工程と、
上記第１の演算工程にて演算した演算結果に応じてエラーの有無及びエラーシンボルの数を判別する第１の判別工程と、
上記第１の判別工程の判別結果に応じてエラー訂正及び／又はポインタ設定を行う第１の制御工程と、
上記第１の制御工程の後の上記第１のチェックワードと各ワードと第２のチェックワードに対して時間遅延を施し、記録媒体からのデータの読出し方向に対して同一方向にデ・スクランブル処理を施す第２のデ・スクランブル処理工程と、

上記第２のデ・スクランブル処理が施された上記各ワードに対して上記第２のチェックワードを用いたロケーション方程式を演算する第２の演算工程と、
上記第２の演算工程にて演算した演算結果に応じてエラーシンボルの数及びエラーロケーションを判別する第２の判別工程と、
上記第２の判別工程の判別結果に応じてエラー訂正又はポインタ設定を行う第２の制御工程とから成ることを特徴とするエラー訂正方法。
【請求項２】
入力データ列が所定のワード数から成るシンボルに分割され、上記各シンボルに対して記録媒体からの読出方向に対して傾斜した方向に配列された第１のデータ系列に対する第１のチェックワードを生成し、上記各シンボルと上記第１のチェックワードに対して記録媒体からの読出方向と同一方向に配列された第２のデータ系列に対する第２のチェックワードを生成し、上記各シンボルと上記第１のチェックワードと上記第２のチェックワードから構成される入力データ列を復号化するエラー訂正方法は、
上記各ワードと上記第１のチェックワードに対して上記第２のチェックワードを用いてロケーション方程式を演算する第１の演算工程と、
上記第１の演算工程にて演算した演算結果に応じてエラーの有無及びエラーシンボルの数を判別する第１の判別工程と、
上記第１の判別工程の判別結果に応じてエラー訂正及び／又はポインタ設定を行う第１の制御工程と、
上記第１の制御工程後の各ワード及び第１のチェックワードと上記第２のチェックワードに対して時間遅延を施し、記録媒体からのデータの読出し方向に対して傾斜した方向にデ・スクランブル処理を施す第１のデ・スクランブル処理工程と、
上記第１のデ・スクランブル処理が施された上記各ワードに対して上記第１のチェックワードを用いてロケーション方程式を演算する第２の演算工程と、
上記第２の演算工程にて演算した演算結果に応じてエラーの有無及びエラーシンボルの数を判別する第２の判別工程と、
上記第２の判別工程の判別結果に応じてエラー訂正及び／又はポインタ設定を行う第２の制御工程と、
上記第２の制御工程後の上記第１のチェックワードと各ワードに対して各々異なる時間遅延を施し、記録媒体からのデータの読出し方向に対して同一方向にデ・スクランブル処理を施す第２のデ・スクランブル処理工程と、
上記第２のデ・スクランブル処理が施された上記各ワードに対して上記第２のチェックワードを用いたロケーション方程式を演算する第３の演算工程と、
上記第３の演算工程にて演算した演算結果に応じてエラーシンボルの数及びエラーロケーションを判別する第３の判別工程と、
上記第３の判別工程の判別結果に応じてエラー訂正又はポインタ設定を行う第３の制御工程とから成ることを特徴とするエラー訂正方法。
【請求項３】
上記第３の制御工程は更に消失訂正を行うことを特徴とする請求項２記載のエラー訂正方法。
【請求項４】
入力データ列が所定のワード数から成るシンボルに分割され、上記各シンボルに対して記録媒体からの読出方向に対して傾斜した方向に配列された第１のデータ系列に対する第１のチェックワードを生成し、上記各シンボルと上記第１のチェックワードに対して記録媒体からの読出方向と同一方向に配列された第２のデータ系列に対する第２のチェックワードを生成し、上記各シンボルと上記第１のチェックワードと上記第２のチェックワードから構成される入力データ列を復号化するエラー訂正装置は、
上記各ワードと上記第１のチェックワードに対して各々異なる時間遅延を施し、記録媒体からのデータの読出し方向に対して傾斜した方向にデ・スクランブル処理を施す第１のデ・スクランブル処理手段と、

上記第１のデ・スクランブル処理が施された上記各ワードに対して上記第１のチェックワードを用いてロケーション方程式を演算する第１の演算手段と、
上記第１の演算手段にて演算した演算結果に応じてエラーの有無及びエラーシンボルの数を判別する第１の判別手段と、
上記第１の判別手段の判別結果に応じてエラー訂正及び／又はポインタ設定を行う第１の制御手段と、
上記第１の制御手段から出力される上記第１のチェックワードと各ワードと、上記第２のチェックワードの各々に対して時間遅延を施し、記録媒体からのデータの読出し方向に対して同一方向にデ・スクランブル処理を施す第２のデ・スクランブル手段と、
上記第２のデ・スクランブル処理が施された上記各ワードに対して上記第２のチェックワードを用いたロケーション方程式を演算する第２の演算手段と、
上記第２の演算手段にて演算した演算結果に応じてエラーシンボルの数及びエラーロケーションを判別する第２の判別手段と、
上記第２の判別手段の判別結果に応じてエラー訂正又はポインタ設定を行う第２の制御手段とから成ることを特徴とするエラー訂正装置。

【請求項５】

入力データ列が所定のワード数から成るシンボルに分割され、上記各シンボルに対して記録媒体からの読出方向に対して傾斜した方向に配列された第１のデータ系列に対する第１のチェックワードを生成し、上記各シンボルと上記第１のチェックワードに対して記録媒体からの読出方向と同一方向に配列された第２のデータ系列に対する第２のチェックワードを生成し、上記各シンボルと上記第１のチェックワードと上記第２のチェックワードから構成される入力データ列を復号化するエラー訂正装置は、
上記各ワードと上記第１のチェックワードに対して上記第２のチェックワードを用いてロケーション方程式を演算する第１の演算手段と、
上記第１の演算手段にて演算した演算結果に応じてエラーの有無及びエラーシンボルの数を判別する第１の判別手段と、
上記第１の判別手段の判別結果に応じてエラー訂正及び／又はポインタ設定を行う第１の制御手段と、
上記第１の制御手段から出力される各ワード及び第１のチェックワードと、上記第２のチェックワードの各々に対して時間遅延を施し、記録媒体からのデータの読出し方向に対して傾斜した方向にデ・スクランブル処理を施す第１のデ・スクランブル処理手段と、
上記第１のデ・スクランブル処理が施された上記各ワードに対して上記第１のチェックワードを用いてロケーション方程式を演算する第２の演算手段と、
上記第２の演算手段にて演算した演算結果に応じてエラーの有無及びエラーシンボルの数を判別する第２の判別手段と、
上記第２の判別手段の判別結果に応じてエラー訂正及び／又はポインタ設定を行う第２の制御手段と、
上記第２の制御手段から出力される上記第１のチェックワードと各ワードに対して各々異なる時間遅延を施し、記録媒体からのデータの読出し方向に対して同一方向にデ・スクランブル処理を施す第２のデ・スクランブル処理手段と、
上記第２のデ・スクランブル処理が施された上記各ワードに対して上記第２のチェックワードを用いたロケーション方程式を演算する第３の演算手段と、
上記第３の演算手段にて演算した演算結果に応じてエラーシンボルの数及びエラーロケーションを判別する第３の判別手段と、
上記第３の判別手段の判別結果に応じてエラー訂正又はポインタ設定を行う第３の制御手段とから成ることを特徴とするエラー訂正装置。

【請求項６】

上記第３の制御手段は更に消失訂正を行うことを特徴とする請求項５記載のエラー訂正装置。

【発明の詳細な説明】

【０００１】
【発明の属する技術分野】
本発明は、例えば記録媒体より再生したり、或いは所定の伝送系により伝送されたオーディオデータなどのデジタルデータを、このデータに付与されたエラー訂正符号に基づいてエラー訂正する場合に適用して好適なエラー訂正方法及びエラー訂正装置に関する。
【０００２】
【従来の技術】
従来、光ディスク又は光磁気ディスクを記録媒体としてデジタルオーディオデータなどのデジタルデータを記録する場合に、その記録データにエラー訂正符号を付加して記録するようにしてある。そして、これらの記録媒体の記録データを再生する際には、再生データに含まれるエラー訂正符号を使用して、再生データのエラーを訂正することが行われている。
【０００３】
図７及び図８は従来のエラー訂正処理の一例を示す図で、図７はエラー訂正符号の生成処理を示し、図８はエラー訂正符号を使用した復号処理を示す。この図７及び図８に示す処理は、ＡＣＩＲＣ（Advanced Cross Interleave Reed-Solomon Code ）方式と称されるエラー訂正方式が適用される処理を示す。このＡＣＩＲＣ方式は、コンパクトディスク（ＣＤ）と称される光ディスクに適用されたＣＩＲＣ方式を改良したエラー訂正方式で、例えばミニディスク（ＭＤ）と称される光磁気ディスク又は光ディスクへのオーディオデータなどの記録に適用された方式である。
【０００４】
まず、図７はエラー訂正符号の生成処理について説明すると、例えば入力データとして左右（Ｌ，Ｒ）２チャンネルのデジタルオーディオデータが、各チャンネル６サンプル分つまり１６ビット×２×６で１９２ビット分（２４バイト）が１つの単位となって入力される。この入力データ列は、１サンプルが２ワードで、１ワードが１６ビットであるとすると、各ワードが上位シンボルＡと下位シンボルＢの２シンボル（１シンボルは８ビット）に分けられ、２４シンボルに変換される。
【０００５】
そして、この２４シンボルのデータを偶数シンボル遅延回路１１に供給し、偶数番目のシンボル（即ち下位シンボルＢ）だけを１シンボル遅延させる処理を行う。そして、この１シンボルおきに１シンボル遅延された各シンボルのデータを、スクランブル用遅延回路１２に供給する。このスクランブル用遅延回路１２では、入力データに互いに異なる遅延量を与えるスクランブルがかけられ、連続するシンボルを４フレームずつ離して遅延させる。但し、１２シンボル目のデータと１３シンボル目のデータとの間は、２０フレーム間隔を開けておく。なお、図中のＤは４を示し、最大で２７×４フレーム（即ち１０８フレーム）にわたるスクランブルが掛けられることを示す（以下、本明細書中に示すＤは全て同じ意味）。
【０００６】
そして、このスクランブルされた各シンボルのデータをＣ２パリティ生成回路１３に供給する。このＣ２パリティ生成回路１３では、誤り訂正符号Ｃ２のエンコードが行われ、ここではｍ＝８，ｎ＝２８，ｋ＝２４，ｄ＝５のリード・ソロモン符号のパリティＱを生成させる。ここで、ｍは１シンボルのビット数、ｎはパリティ生成後の全シンボル数、ｋは入力データのシンボル数、ｄは符号の最小距離である。
【０００７】
そして、このＣ２パリティ生成回路１３で生成された４シンボルのパリティＱと、このＣ２パリティ生成回路１３に供給された２４シンボルのデータとの合計２８シンボルのデータを、インターリーブ用遅延回路１４に供給する。このインターリーブ用遅延回路１４では、連続するシンボルを４フレームずつ離して遅延させる。但し、スクランブル用遅延回路１２で遅延させる間隔を離した方向とは逆方向に離し（即ちスクランブル用遅延回路１２では最初のシンボルのデータを最大の１０８フレーム遅延させたのに対し、インターリ

ーブ用遅延回路１４では最後のシンボルのデータを最大の１０８フレーム遅延させる）、スクランブル用遅延回路１２で１０８フレームにわたってスクランブルされたデータ配列を、ほぼ元に戻す処理が行われる。なお、４シンボルのパリティＱは、１２シンボル目のデータと１３シンボル目のデータとの間に配置して、この４シンボルのパリティＱについても、連続するシンボルを４フレームずつ離させる。

【０００８】

そして、このインターリーブされた各シンボルのデータをＣ１パリティ生成回路１５に供給する。このＣ１パリティ生成回路１５では、誤り訂正符号Ｃ１のエンコードが行われ、ここではｍ＝８，ｎ＝３２，ｋ＝２８，ｄ＝５のリード・ソロモン符号のパリティＰを生成させる。

【０００９】

そして、インターリーブ用遅延回路１４が出力する２８シンボルのデータと、Ｃ１パリティ生成回路１５が出力する４シンボルのパリティＰとを、奇数シンボル遅延回路１６に供給し、奇数番目のシンボルだけを１シンボル遅延させて、エラー訂正符号が付加されたデータ列を得る。

【００１０】

ここで、この奇数シンボル遅延回路１６から並列的に出力される３２シンボルのデータが、１フレームのデータとして送出される。なお、パリティＰ，Ｑのシンボルについては、インバータゲート１７により反転させて出力させる。このインバータゲート１７による反転処理は、何らかの誤動作などでデータ及びパリティが全て“０”データとなった場合でも、誤りが検出できるようにするためである。

【００１１】

そして、例えばこのデータを光磁気ディスクなどの記録媒体に記録する際には、１フレームを構成する各シンボルの８ビットデータを、ＥＦＭ（Eight to Fourteen Modulation：８・１４変換）と称される８ビットのデータを１４ビットのデータに変換する変調を行い、この変調されたデータを、記録媒体に記録させる。なお、実際には各フレームのデータに、同期データやサブコードなどを付加して記録する。

【００１２】

次に、このようにして記録媒体に記録されたデータを再生する際には、再生されたデータをＥＦＭ用復調回路に供給して、１４ビットのデータを８ビットに変換し、この８ビットずつのデータを、３２シンボル単位で図８に示す復号化回路に供給する。この復号化回路は、まず３２シンボルのデータを偶数シンボル遅延回路２１に供給し、偶数番目のシンボルだけを１シンボル遅延させる処理を行う。そして、この１シンボルおきに１シンボル遅延された３２シンボルのデータを、Ｃ１エラー訂正回路２３に供給する。但し、３２シンボルのデータの内、４シンボルのパリティＱ及び４シンボルのパリティＰについては、それぞれインバータゲート２２により反転させてから、Ｃ１エラー訂正回路２３に供給する。

【００１３】

このＣ１エラー訂正回路２３では、パリティＰを使用したロケーション方程式による演算でエラー訂正処理を行う。ここでは、２シンボル誤りまで訂正を行う。但し、２シンボル誤り訂正の場合および訂正不可能の場合には、復号後の各シンボルに誤りフラグ（以下ポインタと称する）を立てる。

【００１４】

そして、Ｃ１エラー訂正回路２３が出力する復号後の各シンボルのデータを、デ・インターリーブ用遅延回路２４に供給し、連続するシンボルを４フレームずつ離して遅延させる。ここでの遅延量を変化させる方向としては、図７に示す符号化回路内のインターリーブ用遅延回路１４での遅延で生じたインターリーブを元に戻す処理（即ちデ・インターリーブ処理）が行われる方向としてある。

【００１５】

そして、デ・インターリーブ用遅延回路２４でデ・インターリーブされた各シンボルのデ

ータを、Ｃ２エラー訂正回路２６に供給する。このＣ２エラー訂正回路２６では、パリティＱを使用したロケーション方程式による演算でエラー訂正処理を行う。ここでの訂正処理としては、各シンボルに付与されたポインタの状態（数，位置）を考慮して行う。

【００１６】

即ち、Ｃ１エラー訂正回路２３が出力するポインタのデータを、遅延回路２５を介してＣ２エラー訂正回路２６に供給し、このポインタを使用したイレージャ訂正処理を行う。ここで、遅延回路２５での遅延処理としては、Ｃ１エラー訂正回路２３が出力する１ビットのポインタのデータを、０フレーム（即ち遅延無しの状態）から４フレーム間隔で１０８フレームまで２８段階に順に遅延させて２８ビットのデータとし、この遅延されたポインタのデータを順にＣ２エラー訂正回路２６に供給する。従って、Ｃ２エラー訂正回路２６側から見ると、４フレーム間隔で１０８フレーム前までのＣ１エラー訂正回路２３でのポインタ発生状態が判ることになる。なお、このポインタの情報を考慮したＣ２エラー訂正回路２６での訂正処理については、フローチャート（図９）に基づいた説明で後述する。

【００１７】

そして、このＣ２エラー訂正回路２６で訂正処理されて復号された各シンボルのデータを、デ・スクランブル用遅延回路２７に供給し、連続するシンボルを４フレームずつ離して遅延させる。ここでの遅延量を変化させる方向としては、図７に示す符号化回路内のスクランブル用遅延回路１２での遅延で生じたスクランブルを元に戻す処理（即ちデ・スクランブル処理）が行われる方向としてある。

【００１８】

そして、デ・スクランブル用遅延回路２７でデ・スクランブルされた各シンボルのデータを、奇数シンボル遅延回路２８に供給し、奇数番目のシンボルだけを１シンボル遅延させる。そして、この奇数シンボル遅延回路２８が出力する各シンボルのデータを、復号された左右２チャンネルのデジタルオーディオデータとして出力する。

【００１９】

次に、この復号化回路でのＣ１エラー訂正回路２３及びＣ２エラー訂正回路２６での訂正処理を、図９のフローチャートに基づいて説明する。

【００２０】

まず、Ｃ１エラー訂正回路２３で訂正処理を行う（ステップ１０１）。ここでは、パリティＰを使用したロケーション方程式による演算を行う（ステップ１０２）。この演算の結果として、エラー無し（ステップ１０３），１シンボルエラー有り（ステップ１０４），２シンボルエラー有り（ステップ１０５），その他の状態（ステップ１０６）の４状態の判別を行う。

【００２１】

この４状態の判別に基づいた処理としては、エラー無しのときポインタをクリアし（ステップ１０７）、１シンボルエラー有りのとき該当する１シンボルを訂正する（ステップ１０８）と共に、その後ポインタをクリアする（ステップ１０７）。また、２シンボルエラー有りのとき該当する２シンボルを訂正する（ステップ１０９）と共に、その後ポインタをセットする（ステップ１１０）。さらに、その他の状態のときにも、ポインタをセットする（ステップ１１０）。ここまでで、Ｃ１エラー訂正回路２３での訂正処理を終了する（ステップ１１１）。

【００２２】

次に、Ｃ２エラー訂正回路２６で訂正処理を行う（ステップ１２１）。ここでは、パリティＱを使用したロケーション方程式による演算を行う（ステップ１２２）。なお、以下の説明でｉ，ｊはそれぞれエラーロケーションを示し、ｉ＜ｊの関係にある。また、Ｎｐは立てられたポインタの数を示し、Ｐｉ，Ｐｊはそれぞれポインタｉ，ｊを示す。

【００２３】

そしてステップ１２２での演算の結果として、エラー無し（ステップ１２３），１シンボルエラー有り（ステップ１２４），２シンボルエラー有り（ステップ１２５），３シンボル以上のエラー有り（ステップ１２６）の４状態の判別を行う。

【００２４】
そして、ステップ１２３でエラー無しと判別したときには、全ポインタをクリアする（ステップ１３１）。また、ステップ１２４で１シンボルエラー有りを判別したときには、このエラー有りのシンボルＳｉのエラーロケーションｉの値を判断し、ｉが２７を越えるとき（ステップ１３２），ｉが２７以下のとき（ステップ１３３）の２状態の判別を行う。
【００２５】
そして、ステップ１３２でｉが２７を越えると判断したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが５以上のとき（ステップ１３４），Ｎｐが４のとき（ステップ１３５），Ｎｐが０～３のとき（ステップ１３６）の３状態の判別を行う。
【００２６】
そして、ステップ１３４でポインタの数Ｎｐが５以上であると判断したとき、ポインタをコピーする（ステップ１５７）。また、ステップ１３５でポインタの数Ｎｐが４であると判断したとき、４シンボルのイレージャ訂正を行う（ステップ１６０）。また、ステップ１３６でポインタの数Ｎｐが０～３であると判断したとき、全ポインタをセットする（ステップ１５９）。
【００２７】
そして、ステップ１３３でｉが２７以下であると判断したときには、１シンボル訂正を行う（ステップ１３７）。
【００２８】
そして、ステップ１２５で２シンボルエラー有りを判別したときには、このエラー有りのシンボルＳｉ，Ｓｊの内のシンボルＳｊのエラーロケーションｊの値を判断し、ｊが２７を越えるとき（ステップ１３８），ｊが２７以下のとき（ステップ１３９）の２状態の判別を行う。
【００２９】
そして、ステップ１３８でｊが２７を越えると判断したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが０～２のとき（ステップ１４０），Ｎｐが５以上のとき（ステップ１４１），Ｎｐが３のとき（ステップ１４２），Ｎｐが４のとき（ステップ１４３）の４状態の判別を行う。
【００３０】
そして、ステップ１４０でポインタの数Ｎｐが０～２を判別したときには、全ポインタをセットする（ステップ１５９）。また、ステップ１４１でポインタの数Ｎｐが５以上を判別したときには、ポインタをコピーする（ステップ１５７）。また、ステップ１４２でポインタの数Ｎｐが３であると判別したときには、３シンボルのイレージャ訂正を行う（ステップ１４６）。さらに、ステップ１４３でポインタの数Ｎｐが４であると判別したときには、４シンボルのイレージャ訂正を行う（ステップ１６０）。
【００３１】
また、ステップ１３９でエラーロケーションｊが２７以下であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが５以下のとき（ステップ１４４），Ｎｐが５を越えるとき（ステップ１４５）の２状態の判別を行う。
【００３２】
そして、ステップ１４４でポインタの数Ｎｐが５以下であると判別したときには、エラー有りのシンボルＳｉ，Ｓｊのポインタｐｉ，ｐｊの値を判別し、ｐｉ＝ｐｊ＝０のとき（ステップ１４７），ｐｉ＝ｐｊ＝１のとき（ステップ１４８），ｐｉ＋ｐｊ＝１のとき（ステップ１４９）の３状態の判別を行う。
【００３３】
そして、ステップ１４７でｐｉ＝ｐｊ＝０であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが１～５のとき（ステップ１５０），Ｎｐが０のとき（ステップ１５１）の２状態の判別を行う。
【００３４】
そして、ステップ１５０でＮｐが１～５であると判別したときには、そのときの立てられ

たポインタの数Ｎｐについて、Ｎｐが０～２のとき（ステップ１４０），Ｎｐが５以上のとき（ステップ１４１），Ｎｐが３のとき（ステップ１４２），Ｎｐが４のとき（ステップ１４３）の４状態の判別を行い、その判別した状態に基づいた上述した処理（ステップ１５９の全ポインタセット，ステップ１５７のポインタコピー，ステップ１４６の３シンボルイレージャ訂正又はステップ１６０の４シンボルイレージャ訂正）を行う。また、ステップ１５１でＮｐが０のときには、２シンボル訂正を行う（ステップ１５６）。

【００３５】

そして、ステップ１４８でＰｉ＝Ｐｊ＝１であると判別したときには、２シンボル訂正を行う（ステップ１５６）。

【００３６】

そして、ステップ１４９でＰｉ＋Ｐｊ＝１であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが１のとき（ステップ１５２），Ｎｐが２又は３のとき（ステップ１５３），Ｎｐが４のとき（ステップ１５４），Ｎｐが５のとき（ステップ１５５）の４状態の判別を行う。

【００３７】

そして、ステップ１５２でポインタの数Ｎｐが１であると判別したときには、２シンボル訂正を行う（ステップ１５６）。また、ステップ１５３でポインタの数Ｎｐが２又は３であると判別したときには、全ポインタをセットする（ステップ１５９）。また、ステップ１５４でポインタの数Ｎｐが４であると判別したときには、４シンボルイレージャ訂正を行う（ステップ１６０）。さらに、ステップ１５５でポインタの数Ｎｐが５であると判別したときには、ポインタコピーを行う（ステップ１５７）。

【００３８】

ここまでの処理を行うことで、Ｃ２エラー訂正が終了し、Ｃ１エラー訂正とＣ２エラー訂正とが行われたことになる（ステップ１６１）。このようにしてＣ１エラー訂正が行われた後に、そのポインタ情報を利用してＣ２エラー訂正を行うことで、効率良くエラー訂正が行われ、高いエラー訂正能力が得られる。

【００３９】

【発明が解決しようとする課題】

ところで、このＡＣＩＲＣ方式の場合には、多くのフレーム数にわたってスクランブルされているので、エラーの発生状態によっては、複数のフレームにわたる長い区間で最終的にエラー訂正不能となってしまう場合がある。

【００４０】

即ち、例えば図１０に示すように、或る区間ｘでバーストエラー（即ち区間ｘ内でエラーが連続する状態）が発生すると、Ｃ１エラー訂正回路２３では、図１０に実線の矢印でデータ処理を行う方向を示すように、記録媒体からのデータの読出し方向と一致した方向に対してエラー処理が行われる。従って、Ｃ１エラー訂正回路２３でのエラー訂正が不可能な範囲は、図１０にＣ１として示す波形の立ち上がった範囲となり、バーストエラーの範囲ｘと一致し、バーストエラーがあってもエラー訂正が不可能な範囲が広がることはない。

【００４１】

これに対し、Ｃ２エラー訂正回路２６では、図１０に破線の矢印でデータ処理を行う方向を示すように、１０８フレームにわたってインターリーブされたデータに対して処理される。このため、Ｃ２エラー訂正回路２６でのエラー訂正が不可能な範囲は、図１０にＣ２として示す波形の立ち上がった範囲となり、バーストエラーの範囲ｘの前後に大きく広がって、最大でインターリーブされた期間である１０８フレームにわたってエラー訂正が不可能になるおそれがある。

更にエラー訂正後のデータを読み出した時点ではエラー訂正が不可能なデータが１０８×２フレームに渡ってばらまかれる。

【００４２】

このような状態になると、元のデータのダメージよりずっと広い範囲に対して訂正不能と

なってしまう。このようにエラー訂正できない範囲が広がるのは、好ましくない。

【００４３】

本発明はかかる点に鑑み、エラー訂正能力を高めることができるエラー訂正方法及びエラー訂正装置を提供することを目的とする。

【００４４】

【課題を解決するための手段】

この問題点を解決するために本発明は、第１のチェックワード，第２のチェックワードの順で訂正符号が生成されて付与されたデータのエラー訂正方法において、最終段の訂正処理として、第２のチェックワードによる第２のデータ系列に対するエラー訂正を行うようにしたものである。

【００４５】

かかる方法によると、最終的なエラー訂正の方向を、第２のデータ系列の配列の方向とすることができる。

【００４６】

【発明の実施の形態】

以下、本発明の第１の実施例を図１～図３を参照して説明する。

【００４７】

本例においては、ＡＣＩＲＣ方式と称されるエラー訂正方式の復号化回路に適用したもので、従来例として図７に示した符号化回路にてエラー訂正符号が符号化されたデータの復号を行うものである。

【００４８】

まず、記録媒体から再生された（又は何らかの伝送路により伝送された）デジタルオーディオデータを、ＥＦＭ用復調回路などの処理の復調回路に供給して、８ビットずつのデータとし、この８ビットずつのデータを、３２シンボル単位で図１に示す復号化回路に供給する。この場合、１サンプルのデジタルオーディオデータは１６ビットで構成され、この１サンプル１６ビットのデータは、上位８ビットのシンボルＡと、下位８ビットのシンボルＢとで構成され、３２シンボルの内の２４シンボルがオーディオデータで、残りの８シンボルがＣ１パリティ生成回路で生成されたパリティＰ（４シンボル）とＣ２パリティ成分回路で生成されたパリティＱ（４シンボル）である。

【００４９】

図１の復号化回路の構成について説明すると、まず３２シンボルのデータを偶数シンボル遅延回路３１に供給し、偶数番目のシンボルだけを１シンボル遅延させる処理を行う。そして、この１シンボルおきに１シンボル遅延された３２シンボルのデータの内、４シンボルのパリティＱ及び４シンボルのパリティＰを、それぞれインバータゲート３２により反転させる。

【００５０】

そして、偶数シンボル遅延回路３１の出力（データのシンボル）及びインバータゲート３２の出力（パリティのシンボル）を、デ・スクランブル用遅延回路３３に供給する。但し、Ｐパリティについては、遅延回路３３には供給しない。この遅延回路３３は、連続するシンボルを４フレームずつ離して遅延させる。即ち、隣接するシンボル毎に４フレームずつ（図中のＤは４）遅延量を変化させて、最小で遅延無しの状態から、最大で２７×Ｄフレーム＝１０８フレーム遅延までの状態を設定させて、デ・スクランブル処理が行われる。ここでの処理としては、最初のシンボルのデータに最大の遅延量を設定し、最後のシンボルのデータを遅延無しとするように遅延量を変化させる。

【００５１】

そして、遅延回路３３でデ・スクランブルされたデータ及びＱパリティのシンボルを、Ｃ２エラー訂正回路３４に供給する。このＣ２エラー訂正回路３４では、パリティＱを使用したロケーション方程式による演算でエラー訂正処理を行う。ここでのエラー訂正処理としては、２シンボルエラーまでの訂正を行う。そして、２シンボルのエラー訂正をした場合、及び２シンボルを越えるエラーがあって訂正ができなかった場合には、該当するシン

ボルにポインタを立てる。
【００５２】
そして、Ｃ２エラー訂正回路３４が出力するデータを、デ・インターリーブ用遅延回路３５に供給する。この場合、本例においては、Ｃ２エラー訂正回路３４に供給された４シンボルのパリティＱについても、そのまま出力させてデ・インターリーブ用遅延回路３５に供給する。また、インバータゲート３２で反転された４シンボルのパリティＰも、デ・インターリーブ用遅延回路３５に供給する。
【００５３】
このデ・インターリーブ用遅延回路３５では、供給される３２シンボルの内の２４シンボルのデータと４シンボルのパリティＱに対して、それぞれ異なる遅延量を与えるデ・インターリーブ処理を行う。即ち、隣接するシンボル毎に４フレームずつ遅延量を変化させて、最小で遅延無しの状態から、最大で２７×Ｄフレーム＝１０８フレーム遅延までの状態を設定させて、デ・インターリーブ処理が行われる。ここでの処理としては、デ・スクランブル用遅延回路３５で遅延量を変化させた方向とは逆、即ち最初のシンボルのデータを遅延無しとし、最後のシンボルのデータに最大の遅延量を設定とする。また、このデ・スクランブル用遅延回路３５では、４シンボルのパリティＰに対しても、それぞれ１０８フレームずつ遅延させる。
【００５４】
そして、このデ・インターリーブ用遅延回路３５でデ・インターリーブされた各シンボルのデータを、Ｃ１エラー訂正回路３７に供給する。このＣ１エラー訂正回路３７では、パリティＰを使用したロケーション方程式による演算でエラー訂正処理を行う。ここでの訂正処理としては、各シンボルに付与されたポインタの状態（数，位置）を考慮して行う。
【００５５】
即ち、Ｃ２エラー訂正回路３４が出力するポインタのデータを、遅延回路３６を介してＣ１エラー訂正回路３７に供給し、このポインタを使用した訂正処理を行う。ここで、遅延回路３６での遅延処理としては、Ｃ２エラー訂正回路３４が出力する１ビットのポインタのデータを、０フレーム（即ち遅延無しの状態）から４フレーム間隔で１０８フレームまで２８段階に順に遅延させて２８ビットのデータとし、この遅延されたポインタのデータを順にＣ１エラー訂正回路３７に供給する。従って、Ｃ１エラー訂正回路３７側から見ると、４フレーム間隔で１０８フレーム前までのＣ２エラー訂正回路３４でのポインタ発生状態が判ることになる。なお、このポインタの情報を考慮したＣ１エラー訂正回路３７での訂正処理については、フローチャート（図２）に基づいた説明で後述する。
【００５６】
そして、このＣ１エラー訂正回路３７で訂正処理されて復号された各シンボルのデータを、奇数シンボル遅延回路３８に供給し、奇数番目のシンボルだけを１シンボル遅延させる。そして、この奇数シンボル遅延回路３８が出力する各シンボルのデータを、復号された左右２チャンネルのデジタルオーディオデータとして出力する。
【００５７】
次に、本例の復号化回路でのＣ２エラー訂正回路３４及びＣ１エラー訂正回路３７での訂正処理を、図２のフローチャートに基づいて説明する。
【００５８】
まず、Ｃ２エラー訂正回路３４で訂正処理を行う（ステップ２０１）。ここでは、パリティＱを使用したロケーション方程式による演算を行う（ステップ２０２）。この演算の結果として、エラー無し（ステップ２０３），１シンボルエラー有り（ステップ２０４），２シンボルエラー有り（ステップ２０５），その他の状態（ステップ２０６）の４状態の判別を行う。
【００５９】
この４状態の判別に基づいた処理としては、エラー無しのときポインタをクリアし（ステップ２０７）、１シンボルエラー有りのとき該当する１シンボルを訂正する（ステップ２０８）と共に、その後ポインタをクリアする（ステップ２０７）。また、２シンボルエラ

ー有りのとき該当する２シンボルを訂正する（ステップ２０９）と共に、その後ポインタをセットする（ステップ２１０）。さらに、その他の状態のときにも、ポインタをセットする（ステップ２１０）。ここまでで、Ｃ２エラー訂正回路３４での訂正処理を終了する（ステップ２１１）。

【００６０】

次に、Ｃ１エラー訂正回路３７で訂正処理を行う（ステップ２２１）。ここでは、パリティＰを使用したロケーション方程式による演算を行う（ステップ２２２）。なお、以下の説明でｉ，ｊはそれぞれエラーロケーションを示し、ｉ＜ｊの関係にある。また、Ｎｐは立てられたポインタの数を示し、Ｐｉ，Ｐｊはそれぞれポインタｉ，ｊを示す。

【００６１】

そしてステップ２２２での演算の結果として、エラー無し（ステップ２２３），１シンボルエラー有り（ステップ２２４），２シンボルエラー有り（ステップ２２５），３シンボル以上のエラー有り（ステップ２２６）の４状態の判別を行う。

【００６２】

そして、ステップ２２３でエラー無しと判別したときには、全ポインタをクリアする（ステップ２３１）。また、ステップ２２４で１シンボルエラー有りを判別したときには、このエラー有りのシンボルＳｉのエラーロケーションｉの値を判断し、ｉが３１を越えるとき（ステップ２３２），ｉが３１以下のとき（ステップ２３３）の２状態の判別を行う。

【００６３】

そして、ステップ２３２でｉが３１を越えると判断したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが４以上のとき（ステップ２３４），Ｎｐが０～３のとき（ステップ２３５）の２状態の判別を行う。

【００６４】

そして、ステップ２３４でポインタの数Ｎｐが４以上であると判断したとき、ポインタをコピーする（ステップ２５７）。また、ステップ２３５でポインタの数Ｎｐが０～３であると判断したとき、全ポインタをセットする（ステップ２５８）。

【００６５】

そして、ステップ２３３でｉが３１以下であると判断したときには、１シンボル訂正を行う（ステップ２３６）。

【００６６】

そして、ステップ２２５で２シンボルエラー有りを判別したときには、このエラー有りのシンボルＳｉ，Ｓｊの内のシンボルＳｊのエラーロケーションｊの値を判断し、ｊが３１を越えるとき（ステップ２３７），ｊが３１以下のとき（ステップ２３８）の２状態の判別を行う。

【００６７】

そして、ステップ２３７でｊが３１を越えると判断したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが０～２のとき（ステップ２３９），Ｎｐが３以上のとき（ステップ２４０）の２状態の判別を行う。

【００６８】

そして、ステップ２３９でポインタの数Ｎｐが０～２を判別したときには、全ポインタをセットする（ステップ２５８）。また、ステップ２４０でポインタの数Ｎｐが３以上を判別したときには、ポインタをコピーする（ステップ２５７）。

【００６９】

また、ステップ２３８でエラーロケーションｊが３１以下であると判別したときには、そのときに立てられたポインタの数Ｎｐについて、Ｎｐが５以下のとき（ステップ２４１），Ｎｐが５を越えるとき（ステップ２４２）の２状態の判別を行う。

【００７０】

そして、ステップ２４１でポインタの数Ｎｐが５以下であると判別したときには、シンボルＳｉ，Ｓｊのエラーロケーションｉ，ｊの値を判断し、ｉが２７を越えるとき（ステップ２４３），ｉが２７以下且つｊが２７を越えるとき（ステップ２４４），ｊが２７以下

のとき（ステップ２４５）の３状態の判別を行う。そして、ステップ２４３でｉが２７を越えると判別したときには、そのときに立てられたポインタの数Ｎｐについて、Ｎｐが１～５のとき（ステップ２５１），Ｎｐが０のとき（ステップ２５２）の２状態の判別を行い、ステップ２５１でＮｐが１～５のときには、既に述べたステップ２３９，２４０の判別処理に移る。また、ステップ２５２でＮｐが０のときには、２シンボル訂正を行う（ステップ１５６）。

【００７１】

そして、ステップ２４４でｉが２７以下且つｊが２７を越えると判別したときには、ポインタＰｉが０のとき（ステップ２４６），ポインタＰｉが１のとき（ステップ２４７）の２状態の判別を行う。そして、ステップ２４６でポインタＰｉが０であると判別したときには、既に述べたステップ２５１，２５２の判別処理に移る。また、ステップ２４７でポインタＰｉが１であると判別したときには、２シンボル訂正を行う（ステップ２５６）。

【００７２】

そして、ステップ２４５でｊが２７以下であると判別したときには、エラー有りのシンボルＳｉ，ＳｊのポインタＰｉ，Ｐｊの値を判別し、Ｐｉ＝Ｐｊ＝０のとき（ステップ２４６），Ｐｉ＝Ｐｊ＝１のとき（ステップ２４９），Ｐｉ＋Ｐｊ＝１のとき（ステップ２５０）の３状態の判別を行う。

【００７３】

そして、ステップ２４８でＰｉ＝Ｐｊ＝０であると判別したときには、既に述べたステップ２５１，２５２の判別処理に移る。また、ステップ２４９でＰｉ＝Ｐｊ＝１であると判別したときには、２シンボル訂正を行う（ステップ２５６）。さらに、ステップ２５０でＰｉ＋Ｐｊ＝１であると判別したときには、そのときに立てられたポインタの数Ｎｐについて、Ｎｐが１のとき（ステップ２５３），Ｎｐが２又は３のとき（ステップ２５４），Ｎｐが４又は５のとき（ステップ２５５）の３状態の判別を行い、ステップ２５３でＮｐが１のとき２シンボル訂正を行い（ステップ２５６）、ステップ２５４でＮｐが２又は３のとき全ポインタのセットを行い（ステップ２５８）、ステップ２５５でＮｐが４又は５のとき全ポインタのコピーを行う（ステップ２５７）。

【００７４】

ここまでの処理を行うことで、Ｃ１エラー訂正が終了し、Ｃ２エラー訂正，Ｃ１エラー訂正の順序で両エラー訂正が行われたことになる（ステップ２５９）。このようにしてＣ２エラー訂正が行われた後に、そのポインタ情報を利用してＣ１エラー訂正を行うことで、従来と同様に２種類のエラー訂正符号による効率の良いエラー訂正ができる。そして本例においては、Ｃ２エラー訂正を最初に行ってから、最後にＣ１エラー訂正を行うようにしたので、エラー訂正不能なバーストエラーが発生した場合に、その影響の広がりを抑えることができる。

【００７５】

即ち、本例の場合には、Ｃ１エラー訂正回路３７で訂正処理されたデータが、奇数シンボルの遅延だけが行われて出力され、最終的にエラー訂正されてから出力されるまでの間に、大幅にシンボル配列を変えるデ・スクランブル処理などの処理が必要なく、データの出力される方向と最終的にエラー訂正される方向とが一致し、エラー訂正不能なバーストエラーが発生した場合であっても、エラー訂正不能な範囲が広がることがない。

【００７６】

なお、本例の回路の場合には、シンボルデータの遅延用に使用する記憶素子の容量としては、従来の回路とほぼ同じで、本例の構成としたことによる回路規模の増大はない。

【００７７】

図３は、ＡＣＩＲＣ方式の符号化がなされた再生データに、所定区間のバーストエラーがあった場合のエラーフラグ（訂正不能であることを示すフラグ）の発生状態を示す図で、例えば図３のＡに区間ａのバーストエラーがあったとする。このとき、第１の実施例の復号化回路（即ちＣ２訂正の後にＣ１訂正をする回路）による処理では、図３のＢに示すように、バーストエラーの区間ａから処理に要する時間だけ遅れたタイミングにエラーフラ

グが発生するが、このエラーフラグが発生する期間の長さ自体は、バーストエラーの区間ａとほぼ同じに抑えることができる。これに対し、Ｃ１訂正後にＣ２訂正を行う従来の復号化回路による処理では、図３のＣに示すように、非常に長い期間（例えば約１０８×２＝２１６フレーム）に連続してエラーフラグが発生してしまう。

【００７８】

次に、本発明の第２の実施例を図４～図６を参照して説明する。

【００７９】

本例においても、ＡＣＩＲＣ方式と称されるエラー訂正方式の復号化回路に適用したもので、従来例として図７に示した符号化回路にてエラー訂正符号が符号化されたデータの復号を行うもので、この例では従来と同様にＣ１エラー訂正，Ｃ２エラー訂正の順序でエラー訂正をした後、再度Ｃ１エラー訂正を行うようにしたものである。

【００８０】

まず、記録媒体から再生された（又は何らかの伝送路により伝送された）デジタルオーディオデータを、ＥＦＭ用復調回路などの処理の復調回路に供給して、８ビットずつのデータとし、この８ビットずつのデータを、３２シンボル単位で図４に示す復号化回路に供給する。この場合、１サンプルのデジタルオーディオデータは１６ビットで構成され、この１サンプル１６ビットのデータは、上位８ビットのシンボルＡと、下位８ビットのシンボルＢとで構成され、３２シンボルの内の２４シンボルがオーディオデータで、残りの８シンボルがＣ１パリティ生成回路で生成されたパリティＰ（４シンボル）とＣ２パリティ成分回路で生成されたパリティＱ（４シンボル）である。

【００８１】

図４の復号化回路の構成について説明すると、まず３２シンボルのデータを偶数シンボル遅延回路４１に供給し、偶数番目のシンボルだけを１シンボル遅延させる処理を行う。そして、この１シンボルおきに１シンボル遅延された３２シンボルのデータの内、４シンボルのパリティＱ及び４シンボルのパリティＰを、それぞれインバータゲート４２により反転させる。

【００８２】

そして、偶数シンボル遅延回路４１が出力する２４のデータ及びインバータゲート４２が出力する４シンボルのパリティＱと４シンボルのパリティＰを、第１のＣ１エラー訂正回路４３に供給する。

【００８３】

この第１のＣ１エラー訂正回路４３では、パリティＰを使用したロケーション方程式による演算でエラー訂正処理を行う。ここでは、２シンボル誤りまで訂正を行う。但し、２シンボル誤り訂正の場合および訂正不可能の場合には、復号後の各シンボルに誤りフラグ（以下ポインタと称する）を立てる。

【００８４】

そして、第１のＣ１エラー訂正回路４３が出力する復号後の各シンボルのデータを、デ・インターリーブ用遅延回路４４に供給し、連続するシンボルを４フレームずつ離して遅延させる。ここでの遅延量を変化させる方向としては、符号化回路内でのインターリーブ処理を元に戻す処理（即ちデ・インターリーブ処理）が行われる方向としてある。

【００８５】

そして、デ・インターリーブ用遅延回路４４でデ・インターリーブされた各シンボルのデータを、Ｃ２エラー訂正回路４５に供給する。このＣ２エラー訂正回路４５では、パリティＱを使用したロケーション方程式による演算でエラー訂正処理を行う。ここでの訂正処理としては、各シンボルに付与されたポインタの状態（数，位置）を考慮して行う。

【００８６】

即ち、第１のＣ１エラー訂正回路４３が出力するポインタのデータを、遅延回路４６を介してＣ２エラー訂正回路４５に供給し、このポインタを使用したイレージャ訂正処理を行う。ここで、遅延回路４６での遅延処理としては、第１のＣ１エラー訂正回路４３が出力する１ビットのポインタのデータを、０フレーム（即ち遅延無しの状態）から４フレーム

間隔で１０８フレームまで２８段階に順に遅延させて２８ビットのデータとし、この遅延されたポインタのデータを順にＣ２エラー訂正回路４５に供給する。従って、Ｃ２エラー訂正回路４５側から見ると、４フレーム間隔で１０８フレーム前までの第１のＣ１エラー訂正回路４３でのポインタ発生状態が判ることになる。

【００８７】

そして、このＣ２エラー訂正回路４５で訂正処理されて復号された各シンボルのデータを、デ・スクランブル用遅延回路４７に供給し、連続するシンボルを４フレームずつ離して遅延させる。ここでの遅延量を変化させる方向としては、符号化回路内でのスクランブル処理を元に戻す処理（即ちデ・スクランブル処理）が行われる方向としてある。また本例においては、Ｃ２エラー訂正回路４５に供給された４シンボルのパリティＱと、第１のＣ１エラー訂正回路４３に供給された４シンボルのパリティＰについても、デ・スクランブル用遅延回路４７に供給して遅延処理するようにしてある。但し、パリティＱについては、１２シンボル目のデータと１３シンボル目のデータとの間に配置して、データのシンボルと連続して４フレームずつ離して遅延させる処理が行われ、パリティＰについては、４シンボル全て１０８フレーム遅延させる処理が行われる。

【００８８】

そして、デ・スクランブル用遅延回路４７でデ・スクランブルされた各シンボルのデータを、第２のＣ１エラー訂正回路４８に供給する。この第２のＣ１エラー訂正回路４８では、パリティＰを使用したロケーション方程式による演算でエラー訂正処理を行う。ここでの訂正処理としては、各シンボルに付与されたポインタの状態（数，位置）を考慮して行う。

【００８９】

即ち、第１のＣ１エラー訂正回路４３が出力するポインタのデータを、遅延回路４９を介して第２のＣ１エラー訂正回路４８に供給すると共に、Ｃ２エラー訂正回路４５が出力するポインタのデータを、遅延回路５０を介してＣ２エラー訂正回路４８に供給し、それぞれのポインタを使用したイレージャ訂正処理を行う。ここで、遅延回路４９でのＣ１訂正によるポインタの遅延処理としては、第１のＣ１エラー訂正回路４３から第２のＣ１エラー訂正回路４８までの信号のシンボルデータの遅延量に相当する１０８フレーム期間遅延させる回路で、第１のＣ１エラー訂正回路４３が出力する１ビットのポインタのデータが１０８フレーム遅延されて供給される。また、遅延回路２５でのＣ２訂正によるポインタの遅延処理としては、Ｃ２エラー訂正回路４５が出力する１ビットのポインタのデータを、０フレーム（即ち遅延無しの状態）から４フレーム間隔で１０８フレームまで２８段階に順に遅延させて２８ビットのデータとし、この遅延されたポインタのデータを順にＣ１エラー訂正回路４８に供給する。なお、このポインタの情報を考慮した第２のＣ１エラー訂正回路４８での訂正処理については、フローチャート（図５）に基づいた説明で後述する。

【００９０】

そして、この第２のＣ１エラー訂正回路４８で訂正処理されて復号された各シンボルのデータを、奇数シンボル遅延回路５１に供給し、奇数番目のシンボルだけを１シンボル遅延させる。そして、この奇数シンボル遅延回路５１が出力する各シンボルのデータを、復号された左右２チャンネルのデジタルオーディオデータとして出力する。

【００９１】

次に、本例の復号化回路内の第２のＣ１エラー訂正回路４８でのエラー訂正処理を、図５のフローチャートに基づいて説明する。なお、本例の第１のＣ１エラー訂正回路４３での訂正処理と、Ｃ２エラー訂正回路４５での訂正処理については、従来例で図９にフローチャートに示した処理と同じである。そして、この図９のフローチャートのステップ１６１の次に、図５のフローチャートのステップ３０１の処理が行われる。即ち、図９のフローチャートに示すＣ１エラー訂正及びＣ２エラー訂正が行われた後に、図５のステップ３０１に移って、第２のエラー訂正回路４８でのエラー訂正処理が開始される。このときには、まずパリティＰを使用したロケーション方程式による演算を行う（ステップ３０２）。

なお、以下の説明でｉ，ｊはそれぞれエラーロケーションを示し、ｉ＜ｊの関係にある。また、Ｎｐは立てられたポインタの数を示し、Ｐｉ，Ｐｊはそれぞれポインタｉ，ｊを示す。

【００９２】

そしてステップ３０２での演算の結果として、エラー無し（ステップ３０３），１シンボルエラー有り（ステップ３０４），２シンボルエラー有り（ステップ３０５），３シンボル以上のエラー有り（ステップ３０６）の４状態の判別を行う。

【００９３】

そして、ステップ３０３でエラー無しと判別したときには、全ポインタをクリアする（ステップ３０７）。また、ステップ３０４で１シンボルエラー有りを判別したときには、このエラー有りのシンボルＳｉのエラーロケーションｉの値を判断し、ｉが３１を越えるとき（ステップ３０８），ｉが３１以下のとき（ステップ３０９）の２状態の判別を行う。

【００９４】

そして、ステップ３０８でｉが３１を越えると判断したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが５以上のとき（ステップ３１０），Ｎｐが４のとき（ステップ３１１），Ｎｐが０～３のとき（ステップ３１２）の３状態の判別を行う。

【００９５】

そして、ステップ３１０でポインタの数Ｎｐが５以上であると判断したとき、ポインタをコピーする（ステップ３３２）。また、ステップ３１１でポインタの数Ｎｐが４であると判断したとき、４シンボルのイレージャ訂正を行う（ステップ３３５）。さらに、ステップ３１２でポインタの数Ｎｐが０～３であると判断したとき、全ポインタをセットする（ステップ３３５）。

【００９６】

そして、ステップ３０９でｉが３１以下であると判断したときには、１シンボル訂正を行う（ステップ３１３）。

【００９７】

そして、ステップ３０５で２シンボルエラー有りを判別したときには、このエラー有りのシンボルＳｉ，Ｓｊの内のシンボルＳｊのエラーロケーションｊの値を判断し、ｊが３１を越えるとき（ステップ３１４），ｊが３１以下のとき（ステップ３１５）の２状態の判別を行う。

【００９８】

そして、ステップ３１４でｊが３１を越えると判断したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが０～２のとき（ステップ３１６），Ｎｐが５以上のとき（ステップ３１７），Ｎｐが３のとき（ステップ３１８），Ｎｐが４のとき（ステップ３１９）の４状態の判別を行う。

【００９９】

そして、ステップ３１６でポインタの数Ｎｐが０～２を判別したときには、全ポインタをセットする（ステップ３３４）。また、ステップ３１７でポインタの数Ｎｐが５以上を判別したときには、ポインタをコピーする（ステップ３３２）。また、ステップ３１８でポインタの数Ｎｐが３であると判別したときには、３シンボルのイレージャ訂正を行う（ステップ３３２）。さらに、ステップ３１９でポインタの数Ｎｐが４であると判別したときには、４シンボルのイレージャ訂正を行う（ステップ３３５）。

【０１００】

また、ステップ３１５でエラーロケーションｊが３１以下であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが５以下のとき（ステップ３２０），Ｎｐが５を越えるとき（ステップ３２１）の２状態の判別を行う。

【０１０１】

そして、ステップ３２０でポインタの数Ｎｐが５以下であると判別したときには、エラー有りのシンボルＳｉ，ＳｊのポインタＰｉ，Ｐｊの値を判別し、Ｐｉ＝Ｐｊ＝０のとき（ステップ３２３），Ｐｉ＝Ｐｊ＝１のとき（ステップ３２４），Ｐｉ＋Ｐｊ＝１のとき（

ステップ３２５）の３状態の判別を行う。

【０１０２】

そして、ステップ３２３でＰｉ＝Ｐｊ＝０であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが１～５のとき（ステップ３２６），Ｎｐが０のとき（ステップ３２７）の２状態の判別を行う。

【０１０３】

そして、ステップ３２６でＮｐが１～５であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが０～２のとき（ステップ３１６），Ｎｐが５以上のとき（ステップ３１７），Ｎｐが３のとき（ステップ３１８），Ｎｐが４のとき（ステップ３１９）の４状態の判別を行い、その判別した状態に基づいた上述した処理（ステップ３３４の全ポインタセット，ステップ３３２のポインタコピー，ステップ３２２の３シンボルイレージャ訂正又はステップ３３５の４シンボルイレージャ訂正）を行う。また、ステップ３２７でＮｐが０のときには、２シンボル訂正を行う（ステップ３３３）。

【０１０４】

そして、ステップ３２４でＰｉ＝Ｐｊ＝１であると判別したときには、２シンボル訂正を行う（ステップ３３３）。

【０１０５】

そして、ステップ３２５でＰｉ＋Ｐｊ＝１であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが１のとき（ステップ３２８），Ｎｐが２又は３のとき（ステップ３２９），Ｎｐが４のとき（ステップ３３０），Ｎｐが５のとき（ステップ３３１）の４状態の判別を行う。

【０１０６】

そして、ステップ３２８でポインタの数Ｎｐが１であると判別したときには、２シンボル訂正を行う（ステップ３３３）。また、ステップ３２９でポインタの数Ｎｐが２又は３であると判別したときには、全ポインタをセットする（ステップ３３４）。また、ステップ３３０でポインタの数Ｎｐが４であると判別したときには、４シンボルイレージャ訂正を行う（ステップ３３５）。さらに、ステップ３３１でポインタの数Ｎｐが５であると判別したときには、ポインタコピーを行う（ステップ３３２）。

【０１０７】

ここまでの処理を行うことで、１回目のＣ１エラー訂正，Ｃ２エラー訂正，２回目のＣ１エラー訂正の順序でエラー訂正されたことになる（ステップ３３６）。このようにしてＣ１エラー訂正とＣ２エラー訂正が行われた後に、再度Ｃ１エラー訂正を行うことで、最終的に出力されるデータが、Ｃ１エラー訂正されたデータとなり、最終的にエラー訂正される方向とデータが出力される方向とが一致し、エラー訂正不能なバーストエラーが発生した場合であっても、エラー訂正不能な範囲が広がることがなく、バーストエラーによるデータのエラー範囲を適切に反映したエラー訂正結果が得られるようになる。

【０１０８】

また本例においては、１回目のＣ１エラー訂正とＣ２エラー訂正との結果に基づいたイレージャ訂正が２回目のＣ１エラー訂正で行え、第１の実施例の場合に比べエラー訂正能力を向上させることができる。

【０１０９】

なお、この第２の実施例の場合には、エラー訂正回路としては、Ｃ１エラー訂正回路が２組必要になるが、シンボルデータの遅延用に使用する記憶素子の容量としては、従来の回路とほぼ同じである。

【０１１０】

また、この第２の実施例では、２回目のＣ１エラー訂正でイレージャ訂正を行うようにしたが、第２のＣ１エラー訂正回路４８ではイレージャ訂正をしないでエラー訂正を行うようにしても良い。図６のフローチャートは、この場合の例を示す図で、図５のフローチャートと同様に第２のＣ１エラー訂正回路４８での処理だけを示す。

【０１１１】

このイレージャ訂正をしない例を、図６のフローチャートに基づいて説明すると、図９のフローチャートに示すＣ１エラー訂正及びＣ２エラー訂正が行われた後に、図６のステップ４０１に移って、第２のエラー訂正回路４８でのエラー訂正処理が開始される。このときには、まずパリティＰを使用したロケーション方程式による演算を行う（ステップ４０２）。

【０１１２】

そしてステップ４０２での演算の結果として、エラー無し（ステップ４０３），１シンボルエラー有り（ステップ４０４），２シンボルエラー有り（ステップ４０５），３シンボル以上のエラー有り（ステップ４０６）の４状態の判別を行う。

【０１１３】

そして、ステップ４０３でエラー無しと判別したときには、全ポインタをクリアする（ステップ４０７）。また、ステップ４０４で１シンボルエラー有りを判別したときには、このエラー有りのシンボルＳｉのエラーロケーションｉの値を判断し、ｉが３１を越えるとき（ステップ４０８），ｉが３１以下のとき（ステップ４０９）の２状態の判別を行う。

【０１１４】

そして、ステップ４０８でｉが３１を越えると判断したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが４以上のとき（ステップ４１０），Ｎｐが０～３のとき（ステップ４１１）の２状態の判別を行う。

【０１１５】

そして、ステップ４１０でポインタの数Ｎｐが４以上であると判断したとき、ポインタをコピーする（ステップ４２７）。また、ステップ４１１でポインタの数Ｎｐが０～３であると判断したとき、全ポインタをセットする（ステップ４２９）。

【０１１６】

そして、ステップ４０９でｉが３１以下であると判断したときには、１シンボル訂正を行う（ステップ４１２）。

【０１１７】

そして、ステップ４０５で２シンボルエラー有りを判別したときには、このエラー有りのシンボルＳｉ，Ｓｊの内のシンボルＳｊのエラーロケーションｊの値を判断し、ｊが３１を越えるとき（ステップ４１３），ｊが３１以下のとき（ステップ４１４）の２状態の判別を行う。

【０１１８】

そして、ステップ４１３でｊが３１を越えると判断したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが０～２のとき（ステップ４１５），Ｎｐが３以上のとき（ステップ４１６）の２状態の判別を行う。

【０１１９】

そして、ステップ４１５でポインタの数Ｎｐが０～２を判別したときには、全ポインタをセットする（ステップ４２９）。また、ステップ４１６でポインタの数Ｎｐが３以上を判別したときには、ポインタをコピーする（ステップ４２７）。

【０１２０】

また、ステップ４１４でエラーロケーションｊが３１以下であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが５以下のとき（ステップ４１７），Ｎｐが５を越えるとき（ステップ４１８）の２状態の判別を行う。

【０１２１】

そして、ステップ４１７でポインタの数Ｎｐが５以下であると判別したときには、エラー有りのシンボルＳｉ，ＳｊのポインタＰｉ，Ｐｊの値を判別し、Ｐｉ＝Ｐｊ＝０のとき（ステップ４１９），Ｐｉ＝Ｐｊ＝１のとき（ステップ４２０），Ｐｉ＋Ｐｊ＝１のとき（ステップ４２１）の３状態の判別を行う。

【０１２２】

そして、ステップ４１９でＰｉ＝Ｐｊ＝０であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが１～５のとき（ステップ４２２），Ｎｐが０のと

き（ステップ４２３）の２状態の判別を行う。
【０１２３】
そして、ステップ４２２でＮｐが１～５であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが０～２のとき（ステップ４１５），Ｎｐが３以上のとき（ステップ４１６）の２状態の判別を行い、その判別した状態に基づいた上述した処理（ステップ４２９の全ポインタセット又はステップ４２７のポインタコピー）を行う。また、ステップ４２３でＮｐが０のときには、２シンボル訂正を行う（ステップ４２８）。
【０１２４】
そして、ステップ４２０でＰｉ＝Ｐｊ＝１であると判別したときには、２シンボル訂正を行う（ステップ４２８）。
【０１２５】
そして、ステップ４２１でＰｉ＋Ｐｊ＝１であると判別したときには、そのときの立てられたポインタの数Ｎｐについて、Ｎｐが１のとき（ステップ４２４），Ｎｐが２又は３のとき（ステップ４２５），Ｎｐが４又は５のとき（ステップ４２６）の４状態の判別を行う。
【０１２６】
そして、ステップ４２４でポインタの数Ｎｐが１であると判別したときには、２シンボル訂正を行う（ステップ４２８）。また、ステップ４２５でポインタの数Ｎｐが２又は３であると判別したときには、全ポインタをセットする（ステップ４２９）。また、ステップ４２６でポインタの数Ｎｐが４又は５であると判別したときには、ポインタコピーを行う（ステップ４２７）。
【０１２７】
ここまでの処理を行うことで、１回目のＣ１エラー訂正，Ｃ２エラー訂正，２回目のＣ１エラー訂正の順序でエラー訂正されたことになる（ステップ４３０）。このように２回目のＣ１エラー訂正でイレージャ訂正を行わない場合でも、１回ずつＣ１エラー訂正とＣ２エラー訂正を行う場合に比べて、エラー訂正能力を向上させることができる。
【０１２８】
なお、上述した各実施例では、ＡＣＩＲＣ方式で符号化されたデジタルオーディオデータのエラー訂正に適用した例としたが、２系列のエラー訂正符号に基づいてエラー訂正を行う他の方式のエラー訂正方法にも適用できることは勿論である。また、デジタルオーディオデータ以外の各種デジタルデータのエラー訂正にも適用できるものである。
【０１２９】
【発明の効果】
第１のチェックワード（上述した実施例ではＣ２エラー訂正用のパリティＱ），第２のチェックワード（上述した実施例ではＣ１エラー訂正用のパリティＰ）の順で訂正符号が生成されて付与されたデータをエラー訂正する場合に、本来は第２のチェックワード，第１のチェックワードの順でエラー訂正するように設定されたものを、本発明においては、最終的なエラー訂正を第２のデータ系列の配列による第２のチェックワードに基づいたエラー訂正としたので、そのデータ配列によってはバーストエラーの影響を最小限に抑えたエラー訂正ができる効果を有する。例えば、ＡＣＩＲＣ方式と称されるエラー訂正方式の場合には、最終的なエラー訂正の方向がデータの読出し方向と同一になり、バーストエラーによるデータのエラー範囲を適切に反映したエラー訂正結果が得られる効果を有する。
【０１３０】
この場合、第１のチェックワードに基づいたエラー訂正処理をした後、第２のチェックワードに基づいたエラー訂正処理をすることで、従来と比べて訂正処理をする順序を変えるだけの簡単な構成で、良好なエラー訂正が可能になる。
【０１３１】
また、第２のチェックワードに基づいたエラー訂正処理，第１のチェックワードに基づいたエラー訂正処理の順で処理した後、更に第２のチェックワードに基づいたエラー訂正処

理をすることで、エラー訂正能力を効果的に高めることができる。

【図面の簡単な説明】

【図１】本発明の第１の実施例による訂正方法を適用した復号回路の構成図である。

【図２】第１の実施例による訂正処理を示すフローチャートである。

【図３】第１の実施例によるエラーフラグの発生状態を従来と比較した波形図である。

【図４】本発明の第２の実施例による訂正方法を適用した復号回路の構成図である。

【図５】第２の実施例による訂正処理（イレージャ訂正有りの場合）を示すフローチャートである。

【図６】第２の実施例による訂正処理（イレージャ訂正無しの場合）を示すフローチャートである。

【図７】エラー訂正符号の符号化処理の一例を示す構成図である。

【図８】図７の構成で符号化されたデータを復号する従来の復号回路の構成図である。

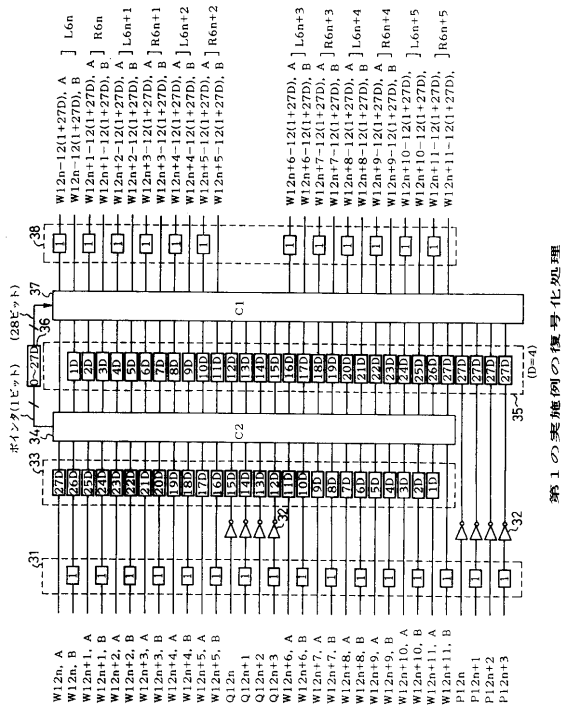
【図９】図８の例の回路による訂正処理を示すフローチャートである。

【図１０】従来のバーストエラーの発生状態の一例を示すタイミング図である。

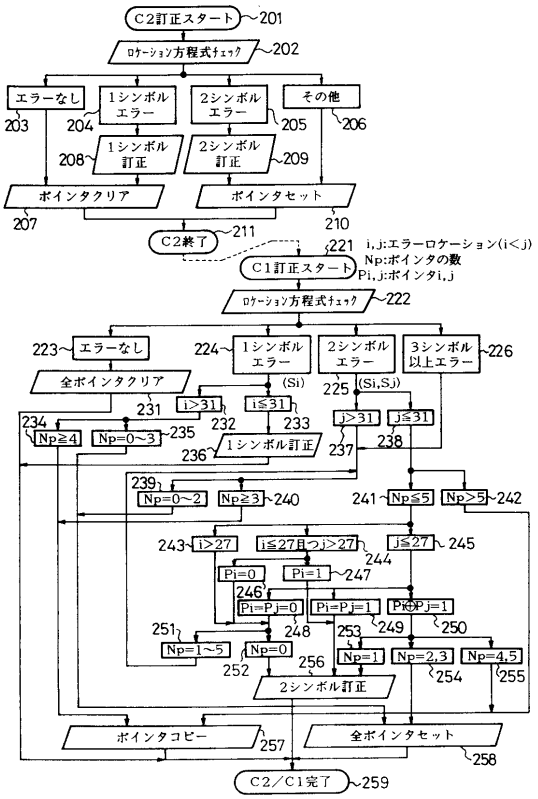
【符号の説明】

３１　偶数シンボル遅延回路
３２　インバータゲート
３３　デ・スクランブル用遅延回路
３４　Ｃ２エラー訂正回路
３５　デ・インターリーブ用遅延回路
３６　ポインタ用遅延回路
３７　Ｃ１エラー訂正回路
３８　奇数シンボル遅延回路
４１　偶数シンボル遅延回路
４２　インバータゲート
４３　第１のＣ１エラー訂正回路
４４　デ・スクランブル用遅延回路
４５　Ｃ２エラー訂正回路
４６　ポインタ用遅延回路
４７　デ・インターリーブ用遅延回路
４８　第２のＣ１エラー訂正回路
４９，５０　ポインタ用遅延回路
５１　奇数シンボル遅延回路

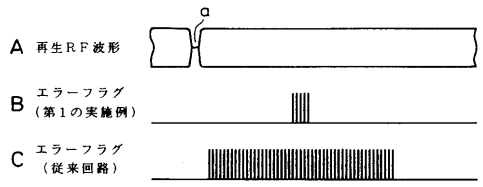
【図１】



【図２】

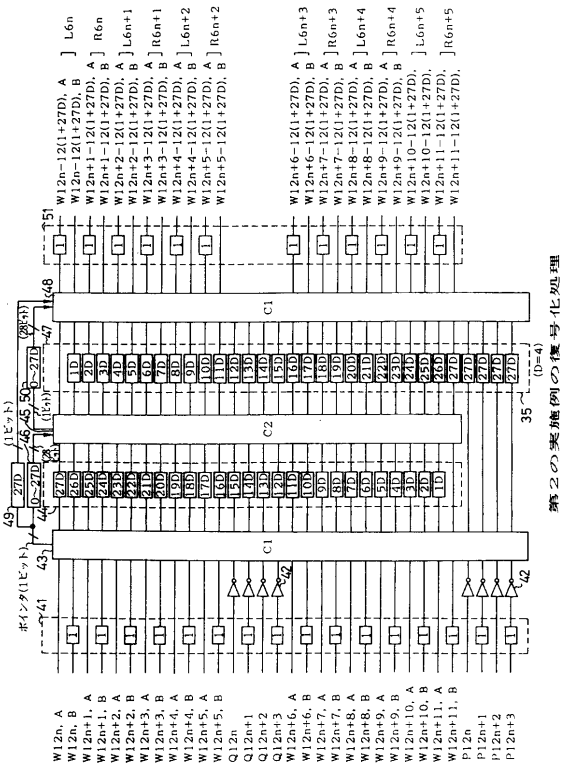


【図３】

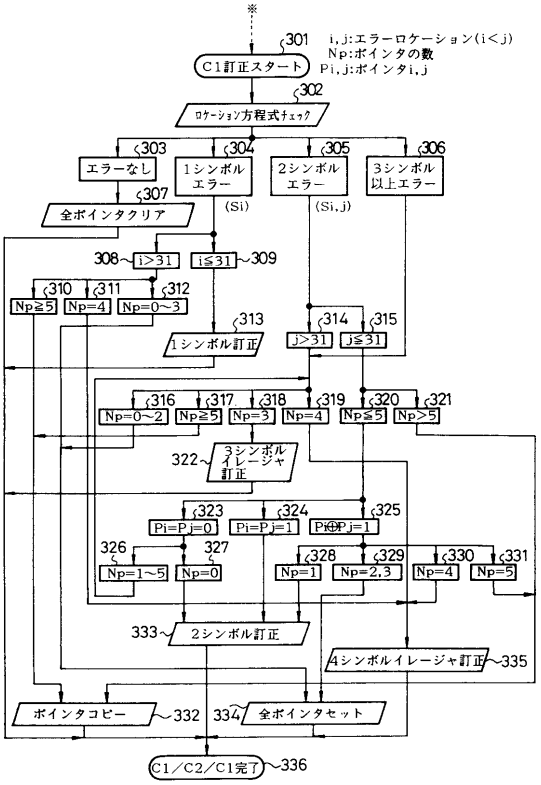


エラーフラグの発生状態

【図４】

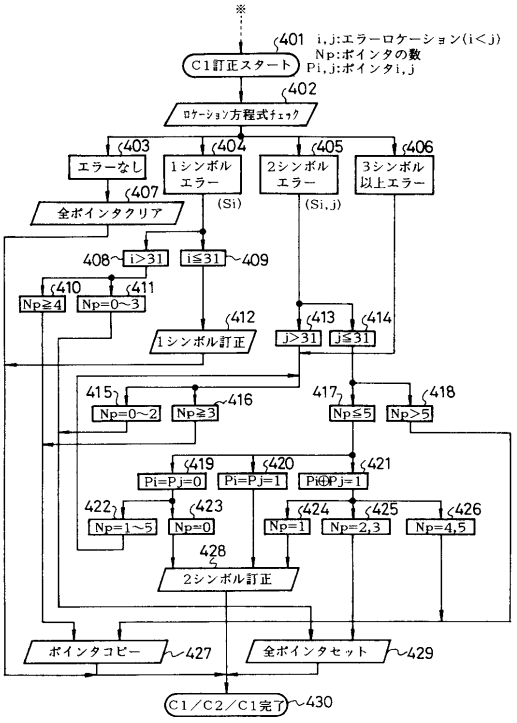


【図５】



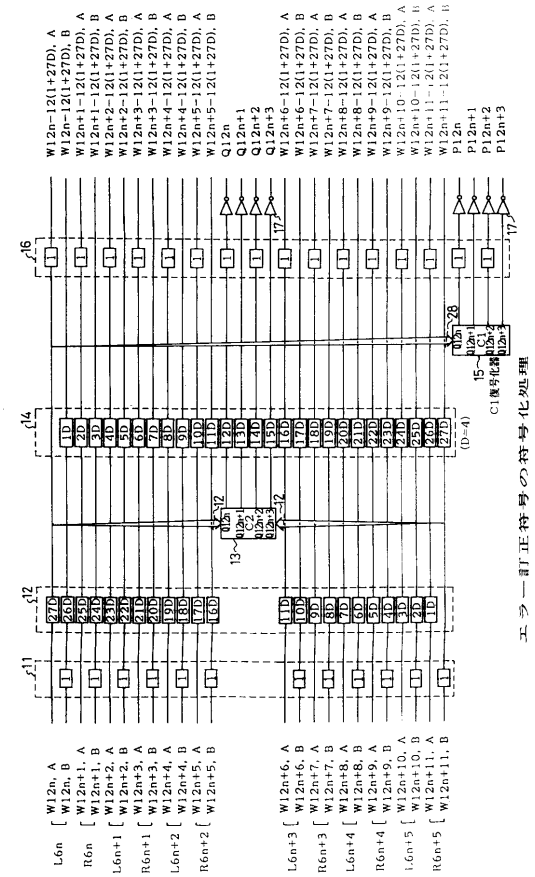
第２の実施例の処理
（イレージャ訂正有りの場合）

【図６】

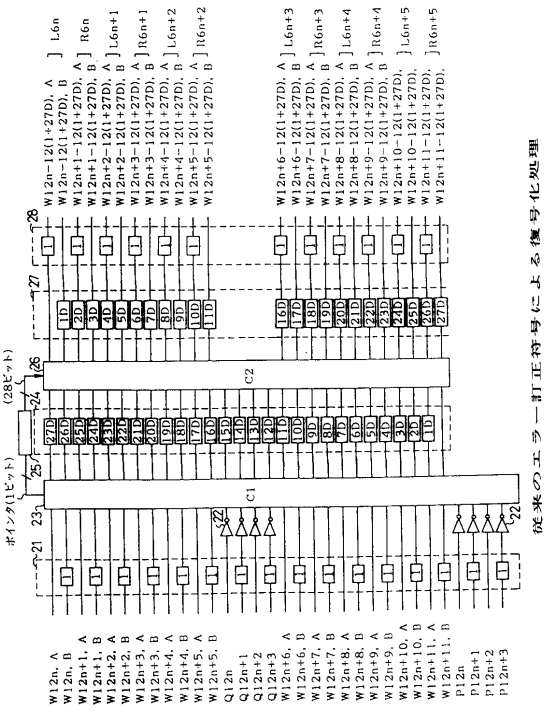


第２の実施例
（イレージャ訂正無しの場合）

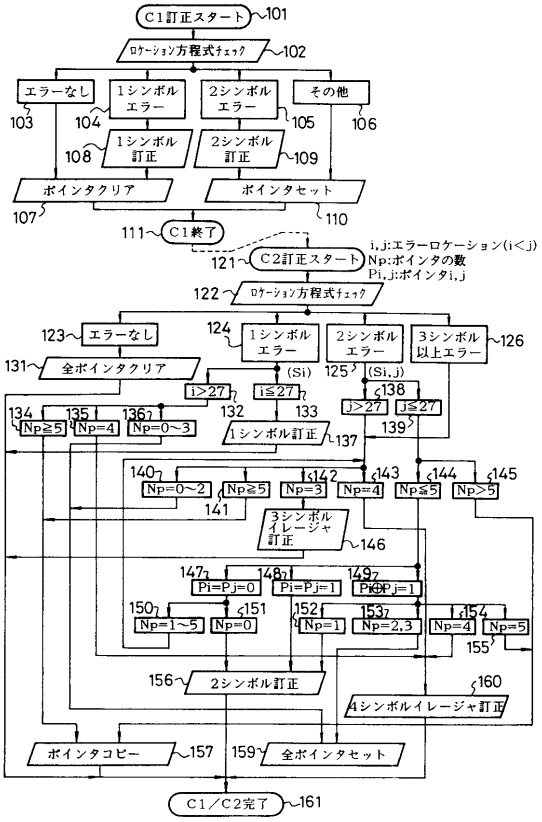
【図７】



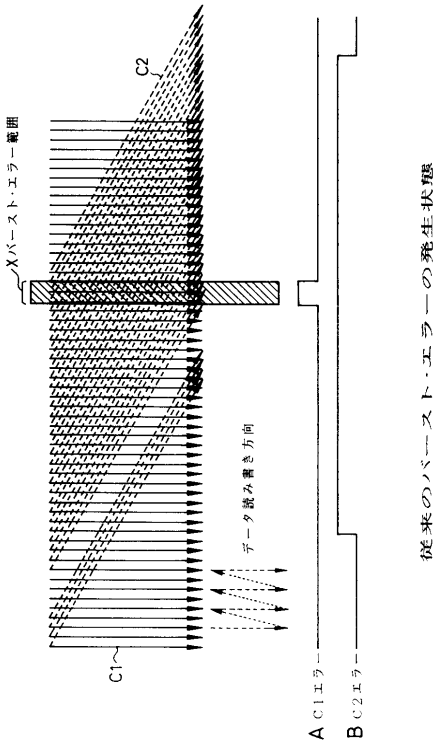
【図８】



【図９】


C1訂正スタート
101
ロケーション方程式チェック
102
エラーなし
1シンボルエラー
2シンボルエラー
その他
103
104
105
106
1シンボル訂正
2シンボル訂正
108
109
ポインタクリア
ポインタセット
107
110
111
C1終了
121
C2訂正スタート
i,j:エラーロケーション(i<j)
Np:ポインタの数
Pi,j:ポインタi,j
122
ロケーション方程式チェック
123
エラーなし
124
1シンボルエラー
2シンボルエラー
3シンボル以上エラー
126
131
全ポインタクリア
(Si)
125
(Si,j)
i>27
i≦27
138
J>27
J≦27
134
135
136
132
133
Np≧5
Np=4
Np=0～3
1シンボル訂正
137
139
140
142
143
144
145
Np=0～2
Np≧5
Np=3
Np=4
Np≦5
Np>5
141
3シンボルイレージャ訂正
146
147
148
149
Pi=Pj=0
Pi=Pj=1
150
151
152
153
154
Np=1～5
Np=0
Np=1
Np=2,3
Np=4
Np=5
155
156
2シンボル訂正
160
4シンボルイレージャ訂正
ポインタコピー
157
159
全ポインタセット
C1/C2完了
161

【図１０】


C2
Xバースト・エラー範囲
データ読み書き方向
C1
A C1エラー
B C2エラー
従来のバースト・エラーの発生状態

フロントページの続き

合議体
審判長　武井　袈裟彦
審判官　望月　章俊
審判官　衣鳩　文彦

(56)参考文献　特開昭６４－６１１２３（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.[7]，ＤＢ名)
H03M 13/00 - 13/53