

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2004-282737

(P2004-282737A)

(43) 公開日 平成16年10月7日(2004.10.7)

(51) Int. Cl.⁷

H03M 13/35
G06F 11/08
G11B 20/18
H03M 13/09

F I

H03M 13/35
G06F 11/08 310B
G11B 20/18 512D
G11B 20/18 520E
G11B 20/18 532D

テーマコード(参考)

5B001
5J065

審査請求 未請求 請求項の数 27 O L (全 20 頁) 最終頁に続く

(21) 出願番号 特願2004-64267 (P2004-64267)
(22) 出願日 平成16年3月8日(2004.3.8)
(31) 優先権主張番号 10/390380
(32) 優先日 平成15年3月14日(2003.3.14)
(33) 優先権主張国 米国(US)

(71) 出願人 591179352
クワンタム・コーポレーション
QUANTUM CORPORATION
アメリカ合衆国、95110 カリフォル
ニア州、サン・ノゼ、テクノロジー・ドラ
イブ、1650、スイート・800

(74) 代理人 100064746
弁理士 深見 久郎
(74) 代理人 100085132
弁理士 森田 俊雄
(74) 代理人 100083703
弁理士 仲村 義平
(74) 代理人 100096781
弁理士 堀井 豊

最終頁に続く

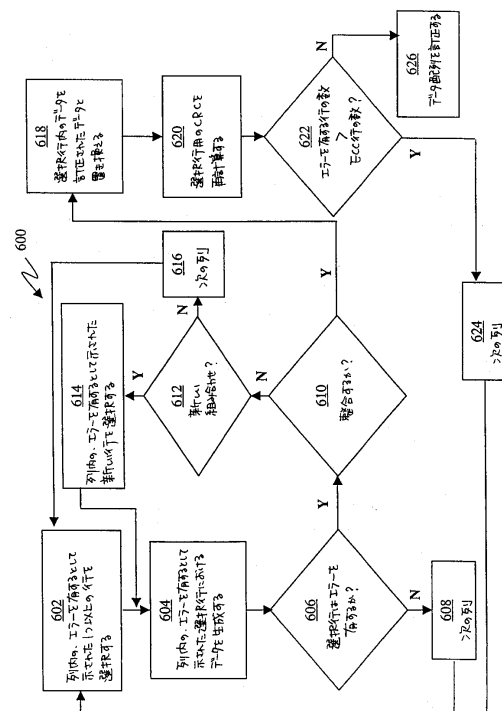
(54) 【発明の名称】 記憶媒体から検索されたデータを検証する方法およびシステム、ならびにコンピュータ可読記憶媒体

(57) 【要約】

【課題】 誤り訂正符号を用いて記憶媒体に記憶されたデータ内の誤りを検証する。

【解決手段】 記憶媒体から検索されたデータ内の誤りが、記憶媒体から複数のデータブロックを検索することにより検証される。多数のデータブロックからのデータを有するデータセットが選択される。1以上の誤り訂正符号(ECC)が記憶媒体から検索される。複数のチェックサムが記憶媒体から検索される。検索された、誤りを有するデータブロックは、データブロックに対応するチェックサムを用いて識別される。誤りを有するデータブロックの数がデータセット用のECCの数より大きい場合、この識別されたデータブロックに対応するデータセット内の第1の組の行が選択される。第1の組の行についてのデータは、データセット用のECCを用いて生成され、誤りを有するデータブロックに対応する行は、第1の組の行についての生成されたデータに基づいて検証される。

【選択図】 図6



【特許請求の範囲】

【請求項 1】

記憶媒体から検索されたデータ内の誤りを検証する方法であって、
記憶媒体から複数のデータブロックを検索するステップと、
多数のデータブロックからのデータを有するデータセットを選択するステップとを含み

、
データセットは、データブロックに対応する複数の行を含み、
1行は、その行に対応する1データブロックからのデータを有し、前記方法はさらに、
記憶媒体から1つ以上の誤り訂正符号(ECC)を検索するステップを含み、
1つ以上のECCはデータセットに対応しており、前記方法はさらに、
記憶媒体から複数のチェックサムを検索するステップを含み、
1チェックサムは1データブロックに対応しており、前記方法はさらに、
記憶媒体から検索された、誤りを有するデータブロックを、データブロックに対応する
チェックサムを用いて識別するステップを含み、
誤りを有するとして識別されたデータブロックの数が、データセット用のECCの数より
も大きい場合、

(a) 誤りを有するとして識別されたデータブロックに対応するデータセット内の第1
の組の行を選択する(第1の組の行における行の数は、データセット用のECCの数と等
しく、誤りを有するとして識別されたデータブロックの数よりも小さい)ステップと、

(b) データセット用のECCを用いて、第1の組の行についてのデータを生成するス
テップと、

(c) 第1の組の行についての生成されたデータに基づいて、誤りを有するとして識別
されたデータブロックに対応する行を検証するステップとを含む、データ内の誤りを検証
する方法。

【請求項 2】

検証するステップは、

第1の組の行についての生成されたデータを、元々第1の組の行にあるデータと比較す
るステップと、

生成されたデータと元々第1の組の行にあるデータとがすべて整合する場合、多数のデ
ータブロックからのデータを有する別のデータセットを処理するステップとを含む、請求
項1に記載の方法。

【請求項 3】

検証するステップは、

第1の組の行についての生成されたデータを、元々第1の組の行にあるデータと比較す
るステップと、

生成されたデータと元々第1の組の行にあるデータがどれも整合しない場合、

(a) 誤りを有するとして識別されたデータブロックに対応するデータセット内の第2
の組の行を選択する(第2の組の少なくとも1つの行は第1の組の行とは異なる)ステッ
プと、

(b) ECCを用いて、第2の組の行についてのデータを生成するステップと、

(c) 第2の組の行についての生成されたデータに基づいて、誤りを有するとして識別
されたデータブロックに対応する行を検証するステップとを含む、請求項1に記載の方法

【請求項 4】

検証するステップは、

第1の組の行についての生成されたデータを、元々第1の組の行にあるデータと比較す
るステップと、

生成されたデータと、元々第1の組の行の1つ以上の行にあるデータとが整合し(整合
行)、生成されたデータと、元々第1の組の行の1つ以上の行にあるデータとが整合しな
い(非整合行)場合、

10

20

30

40

50

元々非整合行にあるデータを、生成されたデータと置き換えるステップとを含む、請求項 1 に記載の方法。

【請求項 5】

非整合行に対応するデータブロック用のチェックサムを生成するステップと、生成されたチェックサムを用いて、誤りを有するとして識別されたデータブロックの数を判断するステップとをさらに含む、請求項 4 に記載の方法。

【請求項 6】

誤りを有するとして識別されたデータブロックの判断された数が、データセット用の E C C の数よりも大きい場合、多数のデータブロックからのデータを有する別のデータセットを処理するステップをさらに含む、請求項 5 に記載の方法。

10

【請求項 7】

誤りを有するとして識別されたデータブロックの判断された数が、データセット用の E C C の数よりも大きくない場合、E C C を用いてデータブロックを訂正するステップをさらに含む、請求項 5 に記載の方法。

【請求項 8】

記憶媒体から検索された複数のデータブロックは、行および列を有する第 1 の論理データアレイを形成しており、1 データブロックは第 1 の論理データアレイ内の 1 行に対応し、選択されたデータセットは第 1 の論理データアレイ内の 1 列に対応している、請求項 1 に記載の方法。

【請求項 9】

第 1 の論理データアレイから第 2 の論理データアレイを形成するステップをさらに含み、第 2 の論理データアレイは、誤りを有するとして識別された第 1 の論理データアレイからの列のみを含み、選択されたデータセットは第 2 の論理データアレイ内の 1 列に対応している、請求項 8 に記載の方法。

20

【請求項 10】

チェックサムはサイクリック・リダンダンシー符号であり、E C C はリード - ソロモン符号である、請求項 1 に記載の方法。

【請求項 11】

選択されたデータセットは、多数のデータブロックのバイトサイズの一部に対応している、請求項 1 に記載の方法。

30

【請求項 12】

記憶媒体から検索されたデータ内の誤りを検出するシステムであって、データバッファを含み、前記データバッファは、記憶媒体から検索された複数のデータブロックと、記憶媒体から検索された 1 つ以上の誤り訂正符号 (E C C) と、記憶媒体から検索された複数のチェックサムとを含み、1 チェックサムは 1 データブロックに対応しており、前記システムはさらに、プロセッサを含み、前記プロセッサは、データバッファ内の多数のデータブロックからのデータを有するデータセットを選択するよう構成され、

40

データセットはデータブロックに対応する複数の行を含み、1 つ以上の E C C はデータセットに対応しており、前記プロセッサはさらに、記憶媒体から検索された、誤りを有するデータブロックを、データブロックに対応するチェックサムを用いて識別するよう構成され、誤りを有するとして識別されたデータブロックの数が、データセット用の E C C の数よりも大きい場合、

(a) 誤りを有するとして識別されたデータブロックに対応するデータセット内の第 1 の組の行を選択し (第 1 の組の行における行の数は、データセット用の E C C の数と等しく、誤りを有するとして識別されたデータブロックの数よりも小さい) 、

(b) データセット用の E C C を用いて、第 1 の組の行についてのデータを生成し、

50

(c) 第1の組の行についての生成されたデータに基づいて、誤りを有するとして識別されたデータブロックに対応する行を検証するよう構成されている、データ内の誤りを検出するシステム。

【請求項13】

誤りを有するとして識別されたデータブロックに対応する行は、

第1の組の行についての生成されたデータを、元々第1の組の行にあるデータと比較するステップと、

生成されたデータと元々第1の組の行にあるデータとがすべて整合する場合、多数のデータブロックからのデータを有する別のデータセットを処理するステップとによって検証される、請求項12に記載のシステム。

10

【請求項14】

誤りを有するとして識別されたデータブロックに対応する行は、

第1の組の行についての生成されたデータを、元々第1の組の行にあるデータと比較するステップと、

生成されたデータと元々第1の組の行にあるデータがどれも整合しない場合、

(a) 誤りを有するとして識別されたデータブロックに対応するデータセット内の第2の組の行を選択する(第2の組の少なくとも1つの行は第1の組の行とは異なる)ステップと、

(b) ECCを用いて、第2の組の行についてのデータを生成するステップと、

(c) 第2の組の行についての生成されたデータに基づいて、誤りを有するとして識別されたデータブロックに対応する行を検証するステップとによって検証される、請求項12に記載のシステム。

20

【請求項15】

誤りを有するとして識別されたデータブロックに対応する行は、

第1の組の行についての生成されたデータを、元々第1の組の行にあるデータと比較するステップと、

生成されたデータと、元々第1の組の行の1つ以上の行にあるデータとが整合し(整合行)、生成されたデータと、元々第1の組の行の1つ以上の行にあるデータとが整合しない(非整合行)場合、

元々非整合行にあるデータを、生成されたデータと置き換えるステップとによって検証される、請求項12に記載のシステム。

30

【請求項16】

非整合行に対応するデータブロック用のチェックサムを生成するステップと、

生成されたチェックサムを用いて、誤りを有するとして識別されたデータブロックの数を判断するステップとをさらに含む、請求項15に記載のシステム。

【請求項17】

誤りを有するとして識別されたデータブロックの判断された数が、データセット用のECCの数よりも大きい場合、多数のデータブロックからのデータを有する別のデータセットを処理するステップをさらに含む、請求項16に記載のシステム。

【請求項18】

誤りを有するとして識別されたデータブロックの判断された数が、データセット用のECCの数よりも大きくない場合、ECCを用いてデータブロックを訂正するステップをさらに含む、請求項16に記載のシステム。

40

【請求項19】

データバッファ内のデータブロックは、行および列を有する論理データアレイを形成しており、1データブロックは論理データアレイ内の1行に対応し、選択されたデータセットは論理データアレイ内の1列に対応している、請求項12に記載のシステム。

【請求項20】

選択されたデータセットは、多数のデータブロックのバイトサイズの一部に対応している、請求項19に記載のシステム。

50

【請求項 2 1】

記憶媒体から検索されたデータ内の誤りをコンピュータに検証させるためのコンピュータ実行可能命令を含むコンピュータ可読記憶媒体であって、

記憶媒体から複数のデータブロックを検索するステップのための命令と、

多数のデータブロックからのデータを有するデータセットを選択するステップのための命令とを含み、

データセットは、データブロックに対応する複数の行を含み、

1行は、その行に対応する1データブロックからのデータを有し、前記コンピュータ可読記憶媒体はさらに、

記憶媒体から1つ以上の誤り訂正符号(ECC)を検索するステップのための命令を含み、

1つ以上のECCはデータセットに対応しており、前記コンピュータ可読記憶媒体はさらに、

記憶媒体から複数のチェックサムを検索するステップのための命令を含み、

1チェックサムは1データブロックに対応しており、前記コンピュータ可読記憶媒体はさらに、

記憶媒体から検索された、誤りを有するデータブロックを、データブロックに対応するチェックサムを用いて識別するステップのための命令を含み、

誤りを有するとして識別されたデータブロックの数が、データセット用のECCの数よりも大きい場合、

(a) 誤りを有するとして識別されたデータブロックに対応するデータセット内の第1の組の行を選択する(第1の組の行における行の数は、データセット用のECCの数と等しく、誤りを有するとして識別されたデータブロックの数よりも小さい)ステップ、

(b) データセット用のECCを用いて、第1の組の行についてのデータを生成するステップ、および、

(c) 第1の組の行についての生成されたデータに基づいて、誤りを有するとして識別されたデータブロックに対応する行を検証するステップのための命令を含む、コンピュータ実行可能命令を含むコンピュータ可読記憶媒体。

【請求項 2 2】

検証するステップは、

第1の組の行についての生成されたデータを、元々第1の組の行にあるデータと比較するステップと、

生成されたデータと元々第1の組の行にあるデータとがすべて整合する場合、多数のデータブロックからのデータを有する別のデータセットを処理するステップとを含む、請求項 2 1に記載のコンピュータ可読媒体。

【請求項 2 3】

検証するステップは、

第1の組の行についての生成されたデータを、元々第1の組の行にあるデータと比較するステップと、

生成されたデータと元々第1の組の行にあるデータがどれも整合しない場合、

(a) 誤りを有するとして識別されたデータブロックに対応するデータセット内の第2の組の行を選択する(第2の組の少なくとも1つの行は第1の組の行とは異なる)ステップと、

(b) ECCを用いて、第2の組の行についてのデータを生成するステップと、

(c) 第2の組の行についての生成されたデータに基づいて、誤りを有するとして識別されたデータブロックに対応する行を検証するステップとを含む、請求項 2 1に記載のコンピュータ可読媒体。

【請求項 2 4】

検証するステップは、

第1の組の行についての生成されたデータを、元々第1の組の行にあるデータと比較す

10

20

30

40

50

るステップと、

生成されたデータと、元々第1の組の行の1つ以上の行にあるデータとが整合し(整合行)、生成されたデータと、元々第1の組の行の1つ以上の行にあるデータとが整合しない(非整合行)場合、

元々非整合行にあるデータを、生成されたデータと置き換えるステップとを含む、請求項21に記載のコンピュータ可読媒体。

【請求項25】

非整合行に対応するデータブロック用のチェックサムを生成するステップと、

生成されたチェックサムを用いて、誤りを有するとして識別されたデータブロックの数を判断するステップとをさらに含む、請求項24に記載のコンピュータ可読媒体。

10

【請求項26】

誤りを有するとして識別されたデータブロックの判断された数が、データセット用のECCの数よりも大きい場合、多数のデータブロックからのデータを有する別のデータセットを処理するステップをさらに含む、請求項25に記載のコンピュータ可読媒体。

【請求項27】

誤りを有するとして識別されたデータブロックの判断された数が、データセット用のECCの数よりも大きくない場合、ECCを用いてデータブロックを訂正するステップをさらに含む、請求項25に記載のコンピュータ可読媒体。

【発明の詳細な説明】

【技術分野】

20

【0001】

1. 発明の分野

この出願は一般に、記憶媒体にデータを記憶させることに関し、より特定的には、記憶媒体に記憶されたデータにおける誤りを検証するために誤り訂正符号を使用することに関する。

【背景技術】

【0002】

2. 関連技術

データは、記憶装置内の記憶媒体へデータを書込むことによって、記憶装置に記憶される。記憶されたデータは後に、記憶媒体からデータを読み出すことによって、記憶装置から検索可能である。しかしながら、多くの理由により、記憶装置から検索されたデータに誤りが存在する、つまり、記憶されたデータが検索できないかまたは記憶媒体に元々記憶されたデータとは異なる場合がある。たとえば、記憶媒体上の記憶されたデータの一部が時を経て劣化し、記憶されたデータのその部分を後で適正に読み出すことができなくなる場合がある。

30

【0003】

そのような誤りを訂正する従来の手法では、誤り訂正符号(ECC)がデータ用に生成され、データとともに記憶媒体に記憶される。記憶されたデータが後に検索されて誤りが検出されると、ECCを用いて検索されたデータを訂正することができる。

【0004】

ECCを用いるあるアプローチでは、データがアレイとなって構成され、ECCがそのデータ用にそのアレイの2つの次元において(たとえばアレイの行および列に沿って)生成される。これらは通常、2次元ECCと呼ばれる。このアプローチの1つの欠点は、アレイのサイズが増大するにつれ、ECCを生成するため、および検索されたデータを訂正するために必要な時間の量も増大する、ということである。

40

【0005】

ECCを用いる別のアプローチでは、ECCはデータブロック用に生成され、記憶媒体に記憶される際にデータブロック内に分散される。データブロック内に分散されたECCの位置を記憶する必要があるため、このアプローチの1つの欠点は、それがECC専用のオーバーヘッドの量を増大させることであり、それは記憶装置の総記憶容量を減少させる

50

場合がある。

【0006】

加えて、ECCを用いて誤りを訂正するための従来の手法では、記憶媒体にデータを記憶させるために使用可能なデータブロックのサイズは、予想される誤りのサイズによって制限され得る。より特定的には、データブロックは通常、予想される誤りの平均サイズに対応するようサイズ変更される。このため、小さいサイズの誤りが予想される場合には小さいサイズのデータブロックが使用され、大きいサイズの誤りが予想される場合には大きいサイズのデータブロックが使用される。

【発明の開示】

【発明が解決しようとする課題】

10

【0007】

しかしながら、多くの理由により、予想される誤りの平均サイズに対応していないデータブロックサイズを有することが望ましい場合がある。たとえば、大量のデータが通常、記憶媒体に記憶され、記憶媒体から検索される場合には、大きいデータブロックが、アクセス時間およびオーバーヘッドを減らすために、ある意味では望ましい。しかしながら、誤りの平均サイズがデータブロックのサイズに対して小さい場合には、誤り訂正は比較的非効率的になる。なぜなら、小さい誤りに対してもデータブロック全体を訂正する必要があるためである。

【課題を解決するための手段】

【0008】

20

概要

記憶媒体から検索されたデータ内の誤りが、記憶媒体から複数のデータブロックを検索することによって検証される。多数のデータブロックからのデータを有するデータセットが選択され、データセットはデータブロックに対応する複数の行を含み、1行は、その行に対応する1データブロックからのデータを有する。1つ以上の誤り訂正符号(ECC)が記憶媒体から検索され、1つ以上のECCはデータセットに対応する。複数のチェックサムが記憶媒体から検索され、1チェックサムは1データブロックに対応する。記憶媒体から検索された、誤りを有するデータブロックは、データブロックに対応するチェックサムを用いて識別される。誤りを有するとして識別されたデータブロックの数がデータセット用のECCの数より大きい場合、誤りを有するとして識別されたデータブロックに対応するデータセット内の第1の組の行が選択され、第1の組の行における行の数は、データセット用のECCの数と等しく、誤りを有するとして識別されたデータブロックの数より小さい。第1の組の行についてのデータは、データセット用のECCを用いて生成され、誤りを有するとして識別されたデータブロックに対応する行は、第1の組の行についての生成されたデータに基づいて検証される。

30

【発明を実施するための最良の形態】

【0009】

詳細な説明

以下の説明は多数の特定の構成、パラメータなどを述べている。しかしながら、そのような説明はこの発明の範囲に対する限定としては意図されておらず、その代わり、例示的な実施例をより良く説明するために提供されていることが認識されるべきである。

40

【0010】

図1を参照すると、ホスト端末102が記憶装置104に接続されて図示されている。ホストコンピュータ102は、パーソナルコンピュータ、ワークステーション、サーバなどの任意の種類のコピュータであり得る。記憶装置104は、テープドライブ、ハードドライブなどの任意の種類のコピュータであり得る。ホスト端末102は任意の数の記憶装置104に接続可能であること、および、任意の数のホスト端末102が1つ以上の記憶装置104に接続可能であることが認識されるべきである。

【0011】

図1を引続き参照すると、例示的な一実施例では、記憶装置104は、記憶装置104

50

に記憶されたデータ内の誤りを検出および訂正するよう構成されている。より具体的には、記憶装置104に記憶されたデータが検索される際、記憶装置104は、サイクリック・リダンダンシー・チェック(CRC)符号などの誤り訂正符号(ECC)およびチェックサムを用いて、検索されたデータが元々記憶装置104に記憶されたデータと異なるかどうか、または記憶されたデータが検索不可能かどうかといった検索されたデータ内の誤りを検出および訂正するよう構成されている。

【0012】

図1に示す実施例では、記憶装置104は、記憶媒体106、チャネルおよび読出/書込ヘッド108、プロセッサ110、および誤り検出/訂正ユニット112を含む。記憶装置104では、データは記憶媒体106に記憶される。読出/書込ヘッド108は、記憶媒体106からデータを読出し、および/または記憶媒体106へデータを書込む。プロセッサ110は、チャネルおよび読出/書込ヘッド108の動作を含め、記憶装置104の動作を制御する。以下により詳細に説明するように、誤り検出/訂正ユニット112は、記憶媒体106に記憶されたデータ内の誤りを検出および訂正する。

10

【0013】

この例示的な実施例では、誤り検出/訂正ユニット112は、データバッファ114、ECCエンコーダ/デコーダ116、およびCRCエンコーダ/デコーダ118を含む。データが記憶媒体106に記憶される際、データはホスト端末102から受信されてデータバッファ114へ書込まれる。ECCエンコーダ/デコーダ116およびCRCエンコーダ/デコーダ118は、データバッファ114内のデータ用にECCおよびCRC符号を生成する。読出/書込ヘッド108は次に、データと生成されたECCおよびCRC符号とを、記憶媒体106へ書込む。

20

【0014】

データが記憶媒体106から読出される際、読出/書込ヘッド108は、データとECCおよびCRC符号とを、記憶媒体106からデータバッファ114へ読出す。以下により詳細に説明するように、記憶媒体106から読出されたデータ内のどの誤りも、ECCおよびCRC符号を用いて検出および訂正される。データは次にホスト端末102へ転送されてもよい。

【0015】

例示的な一実施例では、データは、ホスト端末102と記憶装置104との間を、キャッシュに記憶されるデータレコードの状態転送される。データレコードは、2キロバイト、4キロバイト、6キロバイトなどの予め定められた長さのデータブロックに分割される。しかしながら、さまざまな長さのデータブロックが使用されてもよいことが認識されるべきである。

30

【0016】

この例示的な実施例では、データブロックが記憶媒体106から検索された後で、検索されたデータブロック内の誤りを訂正するためにECCを使用する。より具体的には、データブロックを記憶媒体106に記憶させる前に、多数のデータブロックからのデータを有するデータセット用にECCが生成され、データブロックとともに記憶媒体106に記憶される。以下により詳細に説明するように、データブロックが後に検索される際、誤りを有するデータセットが、データセット内のデータおよびデータセットに対応するECCに基づいて識別可能であり、検索されたデータセット内の誤りは、検索されたデータセット内のデータが、データが元々記憶媒体106に記憶されたときのデータセット内のデータとは異なることを示す。誤りを有するとして識別されたデータセット内のデータは次に、データセットに対応するECCを用いて訂正される。

40

【0017】

加えて、この例示的な実施例では、検索されたデータブロック内の誤りを検出するためにCRC符号を使用し、検索されたデータブロック内の誤りは、検索されたデータブロック内のデータが、データが元々記憶媒体106に記憶されたときのデータブロック内のデータとは異なることを示す。より具体的には、データブロックを記憶媒体106に記憶さ

50

せる前に、そのデータブロック用にCRC符号が生成され、データブロックとともに記憶媒体106に記憶される。データブロックが後に検索される際、検索されたデータブロック用に新しいCRC符号が生成される。検索されたデータブロックを記憶媒体106に記憶させる前に、新しいCRC符号は次に、検索されたデータブロックに対応し、検索されたデータブロック用に元々生成された、記憶媒体106から検索されたCRC符号と比較される。新しいCRC符号と検索されたCRC符号とが異なる場合には、そのデータブロックについて誤りが検出される。

【0018】

例示的な一実施例では、1組のデータブロック、1組のECC、および1組のCRC符号は、「エンティティ」と呼ばれるグループとして、ともに読出および書込可能である。図2を参照すると、16のデータブロック、4ブロックのECC、および20のCRC符号を有するエンティティの論理データアレイ202が図示されている。しかしながら、エンティティはさまざまな数のデータブロック、ECC、およびCRC符号を含み得ることが認識されるべきである。

10

【0019】

図2に示すように、データアレイ202は行204および列206を含む。より具体的には、データアレイ202は、データ行208、ECC行210、データ列212、およびCRC列214を含む。この例示的な実施例では、1データ行208は1データブロックに対応する。1ECC行210はECCの1ブロックに対応する。1データ列212は、多数のデータブロックからのデータを有する1データセットに対応する。1CRC列214は、データ行208およびECC行210用の1組のCRC符号に対応する。

20

【0020】

例示的な一実施例では、論理データアレイ202は、エンティティがデータバッファ114(図1)に記憶される形を示す。しかしながら、データアレイ202は図2および図3に示す形で物理的に存在する必要はないことが認識されるべきである。データアレイ202内のデータ、より特定的にはデータ行208およびデータ列212によって規定されるデータは、連続している必要がないことも認識されるべきである。その代わりに、例示的な一実施例では、ホスト端末102(図1)から受信されたデータはインタリーブされている。したがって、データ行208およびデータ列212によって規定されるデータは、ホスト端末102(図1)から受信される別個のファイルの一部に対応可能である。

30

【0021】

図2は、データアレイ202のデータ行208、ECC行210、データ列212、およびCRC列214間の論理的な関係を示している。図3を参照すると、データ行208、ECC行210、データ列212、およびCRC列214間の論理的な関係をより明確に示すために、データアレイ202の一部がより詳細に示されている。

【0022】

図3に示すように、CRC列214の1つのCRC符号は、データアレイ202の1つの行204に対応しており、1つの行204における誤りを検出するために使用される。たとえば、CRC符号 C_1 は、第1のデータブロックに対応しているデータアレイ202の第1のデータ行208に対応する。このため、第1のデータ行208における誤りを検出するために、記憶媒体106(図1)から第1のデータ行208を検索した後で、検索された第1のデータ行208用に新しいCRC符号 C_1 が生成される。新しいCRC符号 C_1 は次に、検索された第1のデータ行208に対応する、記憶媒体106(図1)から検索されたCRC符号(つまりCRC符号 C_1)と比較される。新しいCRC符号 C_1 と検索されたCRC符号 C_1 とが異なる場合には、第1のデータ行208について誤りが検出される。

40

【0023】

図3に同様に示すように、ECC行210内のECCは、多数のデータブロックからのデータを有するデータセットに対応するデータ列212に対応しており、データ列212における誤りを訂正するために使用されてもよい。たとえば、ECC $E_{1,1}$ 、 $E_{2,1}$ 、 E

50

$E_{3,1}$ 、および $E_{4,1}$ は第 1 のデータ列 2 1 2 に対応しており、第 1 のデータ列 2 1 2 (つまりデータ $D_{1,1}$ 、 $D_{2,1}$ 、...、 $D_{j,1}$) における任意の誤りを訂正するために使用されてもよい。 $E_{1,i}$ 、 $E_{2,i}$ 、 $E_{3,i}$ 、および $E_{4,i}$ は最後のデータ列 2 1 2 に対応しており、最後のデータ列 2 1 2 (つまりデータ $D_{1,i}$ 、 $D_{2,i}$ 、...、 $D_{j,i}$) における任意の誤りを訂正するために使用されてもよい。

【0024】

この例示的な実施例では、1 データ列 2 1 2 における 1 つのセル (つまりデータセル) は、1 データブロックの 1 バイトを表わす。このため、データアレイ 2 0 2 の 1 データ列 2 1 2 および ECC 行 2 1 0 内の 1 つのセル (つまり ECC セル) は、データ行 2 0 8 のデータブロックの 1 バイトに対応している。

10

【0025】

たとえば、第 1 のデータ列 2 1 2 および ECC $E_{1,1}$ は、データ行 2 0 8 におけるデータブロックの 1 番目のバイト (つまりデータ $D_{1,1}$ 、 $D_{2,1}$ 、...、 $D_{j,1}$) に対応する。このため、ECC $E_{1,1}$ は、第 1 のデータ列 2 1 2 (つまりデータ $D_{1,1}$ 、 $D_{2,1}$ 、...、 $D_{j,1}$) におけるデータセルを訂正するために使用されてもよい。第 2 のデータ列 2 1 2 および ECC $E_{1,2}$ は、データ行 2 0 8 におけるデータブロックの 2 番目のバイト (つまりデータ $D_{1,2}$ 、 $D_{2,2}$ 、...、 $D_{j,2}$) に対応する。このため、ECC $E_{1,2}$ は、第 2 のデータ列 2 1 2 (つまりデータ $D_{1,2}$ 、 $D_{2,2}$ 、...、 $D_{j,2}$) におけるデータセルを訂正するために使用されてもよい。

【0026】

加えて、この例示的な実施例では、1 データ列 2 1 2 における単一のデータセルを訂正するために ECC を使用してもよい。このため、1 データ列 2 1 2 につき、合計 4 つのデータセルが、そのデータ列 2 1 2 内の 4 つの ECC セルを使用して訂正可能である。たとえば、ECC $E_{1,1}$ は、第 1 のデータ列 2 1 2 における 1 データセル (つまりデータ $D_{1,1}$ 、 $D_{2,1}$ 、...、または $D_{j,1}$) を訂正するために使用されてもよい。ECC $E_{2,1}$ は、第 1 のデータ列 2 1 2 における第 2 のデータセルを訂正するために使用されてもよい。ECC $E_{3,1}$ は、第 1 のデータ列 2 1 2 における第 3 のデータセルを訂正するために使用されてもよい。ECC $E_{4,1}$ は、第 1 のデータ列 2 1 2 における第 4 のデータセルを訂正するために使用されてもよい。

20

【0027】

この例示的な実施例では、1 データ列 2 1 2 用の ECC は、そのデータ列 2 1 2 内のデータに基づいて生成される。たとえば、ECC $E_{1,1}$ 、 $E_{2,1}$ 、 $E_{3,1}$ 、および $E_{4,1}$ は、第 1 のデータ列 2 1 2 (つまりデータ $D_{1,1}$ 、 $D_{2,1}$ 、...、 $D_{j,1}$) に基づいて生成される。上述のように、図 1 を参照すると、ECC は ECC エンコーダ 1 1 6 によって生成される。また、上述のように、ECC は最初、ホスト端末 1 0 2 から受信されたデータ用に生成される。生成された ECC および受信されたデータは次に、記憶媒体 1 0 6 に記憶される。

30

【0028】

図 3 を再度参照すると、例示的な一実施例では、ECC 行 2 1 0 内の ECC は、データ列 2 1 2 に基づいて生成されたリード - ソロモン符号である。たとえば、ECC $E_{1,1}$ は、第 1 のデータ列 2 1 2 (つまりデータ $D_{1,1}$ 、 $D_{2,1}$ 、...、 $D_{j,1}$) に基づいて生成されたリード - ソロモン符号である。リード - ソロモン符号のより詳細な説明については、ここにその全体を引用により援用する、ピーターソンおよびウェルドン (Peterson & Weldon) 著「誤り訂正符号 (Error Correcting Codes)」、第 2 版、MIT プレス、1 9 7 2 年を参照されたい。しかしながら、さまざまな種類の誤り訂正符号が使用されてもよいことが認識されるべきである。

40

【0029】

この例示的な実施例では、ECC $E_{2,1}$ 、 $E_{3,1}$ 、および $E_{4,1}$ も、第 1 のデータ列 2 1 2 (つまりデータ $D_{1,1}$ 、 $D_{2,1}$ 、...、 $D_{j,1}$) に基づいて生成されたリード - ソロモン符号である。ECC $E_{1,1}$ 、 $E_{2,1}$ 、 $E_{3,1}$ 、および $E_{4,1}$ は同じ組のデータ (つまり第 1

50

のデータ列 2 1 2) に基づいて生成されているものの、各 ECC は互いに関して固有のものである。より具体的には、1 組のデータに基づいてリード - ソロモン符号を生成する際、その 1 組のデータの要素の排他的論理和 (XOR) 演算をさまざまな組合せで行なうためにマトリックスが使用される。このため、この例示的な実施例では、ECC $E_{1,1}$ 、 $E_{2,1}$ 、 $E_{3,1}$ 、および $E_{4,1}$ は異なるマトリックスを用いて生成される。たとえば、 $E_{1,1}$ 、 $E_{2,1}$ 、 $E_{3,1}$ 、および $E_{4,1}$ はそれぞれ、第 1、第 2、第 3、および第 4 のマトリックスを用いて生成される。

【0030】

上述のように、データレイ 2 0 2 は、行 2 0 4 内の誤りを検出するために使用可能な CRC 列 2 1 4 を含む。しかしながら、CRC 列 2 1 4 は、誤りが存在する列 2 0 6 を示さない。たとえば、CRC C_2 を用いて、第 2 のデータ行 2 0 8 内のどこかにある誤りを検出することは可能であるが、その誤りは、第 2 のデータ行 2 0 8 (つまりデータ $D_{2,1}$ 、 $D_{2,2}$ 、...、 $D_{2,j}$) における任意の 1 つ以上のデータセルに存在する可能性がある。

【0031】

このため、例示的な一実施例では、ECC 行 2 1 0 内の ECC は、多数のデータブロックからのデータを有する 1 データセットに対応している 1 データ列 2 1 2 が誤りを有するかどうかを判断するために使用される。より具体的には、リード - ソロモン符号が使用される場合、排他的論理和 (XOR) 演算を含む数学的計算が 1 データ列内の 1 つの ECC およびデータに対して行なわれ、不良の列が判断される。リード - ソロモン符号の一特性に従って、ECC が計算されたときとデータ列内のデータが同じであるならば、XOR 演算の結果は 0 (ゼロ) となるはずである。このため、XOR 演算の結果を用いて、データ列が誤りを有するかどうかを判断可能である。

【0032】

たとえば、データレイ 2 0 2 が、記憶媒体 1 0 6 (図 1) から検索されたデータ、ECC、および CRC 符号を図示していると仮定する。このため、ECC 行 2 1 0 内の ECC および CRC 列 2 1 4 内の CRC 符号は、元々ホスト端末 1 0 2 (図 1) から受信されたデータに基づいて、以前に生成されたものである。上述のように、第 1 のデータ列 2 1 2 に誤りがあるかどうかを判断するために、XOR 演算が ECC $E_{1,1}$ およびデータ $D_{1,1}$ 、 $D_{2,1}$ 、...、または $D_{j,1}$ に対して行なわれる。このため、ECC $E_{1,1}$ が元々生成されたときからデータ $D_{1,1}$ 、 $D_{2,1}$ 、...、または $D_{j,1}$ が変わっていない場合には、XOR 演算はゼロ結果を生み出すはずである。このため、例示的な一実施例では、1 データ列 2 1 2 の 1 ECC およびそのデータ列 2 1 2 の XOR 演算がゼロ結果を生み出す場合、そのデータ列 2 1 2 は誤りを有しないと判断される。XOR 演算がゼロではない結果を生み出す場合、そのデータ列 2 1 2 は誤りを有すると判断される。

【0033】

別の例示的な実施例では、1 データ列 2 1 2 に基づいて新しい ECC が生成される。新しく生成された ECC は次に、検索された ECC (つまり、ホスト端末 1 0 2 (図 1) から受信されたデータから元々生成された、記憶媒体 1 0 6 (図 1) から検索された ECC) と比較される。比較の結果を用いて、そのデータ列 2 1 2 が誤りを有するかどうかを判断可能である。

【0034】

たとえば、ECC $E_{1,1}$ が第 1 のデータ列 2 1 2 (つまり $D_{1,1}$ 、 $D_{2,1}$ 、...、または $D_{j,1}$) から生成され、ECC $E_{1,1}$ および第 1 のデータ列 2 1 2 が記憶媒体 1 0 6 (図 1) に記憶されていると仮定されたい。ここで、ECC $E_{1,1}$ および第 1 のデータ列 2 1 2 が記憶媒体 1 0 6 (図 1) から検索されると仮定されたい。検索された第 1 のデータ列 2 1 2 が任意の誤りを有するかどうかを判断するために、検索された第 1 のデータ列 2 1 2 に基づいて、新しい ECC (つまり ECC $E_{1,1}$) が生成される。 $E_{1,1}$ と $E_{1,1}$ とは次に比較される。 $E_{1,1}$ と $E_{1,1}$ とが同じ場合、検索された第 1 のデータ列 2 1 2 には誤りがない。 $E_{1,1}$ と $E_{1,1}$ とが異なる場合、検索された第 1 のデータ列 2 1 2 には誤りがある。

10

20

30

40

50

【0035】

上述のように、例示的な一実施例では、 $ECC E_{2,1}$ 、 $E_{3,1}$ 、および $E_{4,1}$ も第1のデータ列212から生成される。このため、1データ列212に誤りが存在するかどうかを判断する精度を高めるために、1データ列212に対応する2つ以上の新しい ECC を、検索されたデータ列212に基づいて生成することが可能である。

【0036】

たとえば、新しい $ECC E_{2,1}$ （つまり $E_{2,1}$ ）を第1のデータ列212から生成し、検索された $ECC E_{2,1}$ と比較することが可能である。加えて、またはこれに代えて、新しい $ECC E_{3,1}$ （つまり $E_{3,1}$ ）を第1のデータ列212から生成し、検索された $ECC E_{3,1}$ と比較することが可能である。加えて、またはこれに代えて、新しい $ECC E_{4,1}$ （つまり $E_{4,1}$ ）を第1のデータ列212から生成し、検索された $ECC E_{4,1}$ と比較することが可能である。

10

【0037】

また、上述のように、 $ECC E_{1,1}$ 、 $E_{2,1}$ 、 $E_{3,1}$ 、および $E_{4,1}$ は異なるマトリックスから生成される。たとえば、 $E_{1,1}$ 、 $E_{2,1}$ 、 $E_{3,1}$ 、および $E_{4,1}$ がそれぞれ、第1、第2、第3、および第4のマトリックスから生成されると仮定されたい。このため、上述の例では、 $E_{1,1}$ 、 $E_{2,1}$ 、 $E_{3,1}$ および $E_{4,1}$ もそれぞれ、第1、第2、第3、および第4のマトリックスから生成される。

【0038】

図4を参照すると、例示的な一実施例では、減少した論理データレイ402が、論理データレイ202（図2）から形成される。この例示的な実施例では、減少した論理データレイ402は、誤りを有するとして識別されたデータレイ202（図2）内のデータ列212のみを含む。 ECC 行210が次に、誤りを有するとして CR 列214により識別されたデータ行208を訂正するために使用される。減少したデータレイ402はデータレイ202（図2）よりも小さいため、減少したデータレイ402は、データレイ202（図2）よりも短い時間で、および少ない数値計算で訂正可能である。

20

【0039】

また、これに代えて、例示的な一実施例では、誤りを有するとして識別されたデータレイ202（図2）内のデータ列212は、論理データレイ402を形成することなく訂正される。このため、データレイ202（図2）内の1データ列212は、それが誤りを有するかどうかを判断するために調べられる。そのデータ列212が誤りを有するとして識別される場合には、データ列212は、別のデータ列212を調べる前に訂正される。

30

【0040】

ここで図5を参照すると、記憶媒体に記憶されたデータを訂正する例示的なプロセス500が示されている。より具体的には、図5は、コンピュータが実行可能な、記憶装置の動作を指示する命令を含むコンピュータプログラムとして実現される例示的なプロセス500を示す。たとえば、図1を参照すると、図5に示すプロセスは、記憶装置104のプロセッサ110の動作を指示可能である。しかしながら、例示的なプロセス500は、特定用途向け集積回路（ASIC）といったハードウェアにおいて実現可能であることが認識されるべきである。

40

【0041】

図5を再度参照すると、この例示的なプロセスでは、記憶装置はテープドライブであり、記憶媒体は磁気テープである。例示のため、ステップ502の前に、データおよびそのデータ用の ECC がテープドライブ内のテープに記憶されたと仮定されたい。このため、図5は、テープに記憶されたデータを検索し、検索されたデータ内の任意の誤りを訂正する例示的なプロセスを示している。

【0042】

より具体的には、ステップ502で、テープから検索されたデータブロック用に新しい ECC が生成される。より具体的には、この例示的な実施例では、テープから検索された

50

データブロック（つまり“FromTapeEntity.DataBlocks”）が、ECCを生成する（つまり“GenerateECC”）ルーチンへパスされる。結果は“NewECC”として記憶される。

【0043】

上述のように、および図2に示すように、例示的な一実施例では、1ECC行210は、論理データアレイ202のデータ行208内のデータに対応する。1ECC行210内の各ECCは、1データ列212に対応している、多数のデータ行208からのデータを有する1データセットに対応する。加えて、例示的な一実施例では、論理データアレイ202は、16のデータ行208に対応する16のデータブロック用の、ECC行210に対応するECCの4ブロックを含む。このため、論理データアレイ202が例示的なプロセス500（図5）で使用されると仮定すると、ステップ502（図5）は、テープから検索されたデータ行208に基づいてECCの4つの新しいブロックを生成することに該当する。しかしながら、新しいECCは、全部一斉にというよりもむしろ、一度に1つのデータ列212について生成可能であることが認識されるべきである。

10

【0044】

図5を再度参照すると、ステップ504で、新しく生成されたECCブロック、およびテープから検索されたECCブロックに対し、論理XOR演算が行なわれる。より具体的には、この例示的な実施例では、新しく生成されたECCブロック（つまり“NewECCBlock[1]”）、およびテープから検索されたECC（つまり“FromTape.ECCBlock[1]”）に対し、XORが行なわれる。結果は“XorResult[1]”として記憶される。

【0045】

上述のように、例示的な一実施例では、16のデータブロック用に4つのECCブロックが使用される。したがって、信頼性追加のため、4つの新しいECCブロックが生成され、4つの新しく生成されたECCブロックおよび4つの検索されたECCブロックに対して、4つのXOR演算が行なわれる。4つのXOR演算の結果は、4つのXorResult変数（つまり、XorResult[0]、XorResult[1]、XorResult[2]、およびXorResult[3]）として記憶される。

20

【0046】

このため、図5を再度参照すると、ステップ506で、ステップ504での4つのXOR演算の結果に対し、論理和演算が行なわれる。結果は“EntityErrorMap”として記憶される。この例示的なプロセスでは、“EntityErrorMap”変数の各ビットは1データ列212（図2）に対応しており、0ビットは1データ列212（図2）が誤りを有していないことを示し、1ビットは1データ列212（図2）が誤りを有することを示す。1つの新しく生成されたECCブロックおよび1つの検索されたECCブロックを用いて、単一のXOR演算が実行可能であることが認識されるべきであり、その場合、ステップ506は省略可能である。

30

【0047】

ステップ508で“EntityErrorMap”変数の1ビットが調べられ、そのビットが0ビットであるかどうか判断される。上述のように、そのビットが1ビットである場合には、そのビットに対応するデータ列212（図2）は誤りを有する。このため、ステップ510で、誤りを有するとして識別されたデータ列212（図2）（つまり“FromTapeEntity[column]”）は、別のアレイにおける1列にコピーされる（つまり“PackedErrorEntity[PackedColumn]”）。このため、“PackedErrorEntity”アレイは、誤りを有するとして識別された列のみを含む。ステップ512で、変数“PackedColumn”はインクリメントされる。

40

【0048】

ステップ514で、“EntityErrorMap”変数が調べられ、調べる必要があるビットおよびデータ列がもっとあるかどうか判断される。調べられるべきビットおよびデータ列がさらにある場合には、ステップ508が繰返される。

【0049】

ステップ516で、誤りを有するとして識別されたデータ列が訂正される。より具体的

50

には、この例示的な実施例では、“PackedErrorEntity”アレイは、“PackedErrorEntity”アレイのサイズとともに、アレイを訂正する（つまり“DoGeneralCorrection”）ルーチンへパスされる。

【0050】

ステップ518、520、および522で、訂正されたアレイ内の訂正された列（つまり“PackedErrorEntity[PackedColumn]”）は、元のアレイ（つまり“FromTapeEntity[Column]”）へ逆にコピーされる。

【0051】

図2および図4を再度参照すると、上述のように、データアレイ202（図2）および減少したデータアレイ402（図4）は、行204内の誤りを検出するために使用されるチェックサムの列を含む。より特定的には、データアレイ202（図2）および402（図4）はCRC列214を含む。

【0052】

また、上述のように、データアレイ202（図2）および減少したデータアレイ402（図4）は、多数のECC行210を含み、各ECC行210は1データ列212における単一のセルを訂正するために使用可能である。このため、CRC列214によって誤りを有するとして示された、データアレイ202（図2）または402（図4）内の行204の数が、ECC行210の数よりも大きい場合には、データアレイ202（図2）または402（図4）を従来の手法でECC行210を用いて訂正することは、通常できない。

【0053】

たとえば、図2に示す例示的な実施例では、データアレイ202は4つのECC行210を含む。このため、データアレイ202内の5つ以上の行204がCRC列214によって誤りを有するとして示された場合には、データアレイ202を従来の手法でECC行210を用いて訂正することは、通常できない。

【0054】

しかしながら、CRCは行204内のセルのすべてが誤りを有するとは示さないため、どの単一の列206も、CRC列214によって示されるのと同数の誤りを持たなくてもよい。たとえば、データアレイ202内に誤りを有する行204が5つあることをCRC列214が示す場合、それらの誤りは、どの単一の列206も誤りを有するセル（つまり行）を5つ持たないように、データアレイ202中に分散されてもよい。代わりに、誤りは完全に別々の列206に存在してもよい。

【0055】

こうして、例示的な一実施例では、CRC列214によって誤りを有するとして示された、データアレイ202（図2）または402（図4）内の行204の数が、ECC行210の数よりも大きい場合、列206は、データアレイ202（図2）または402（図4）を訂正するために個々に処理される。より特定的には、列206は、1列206内の行がCRC列214によって示されるように誤りを有することを検証するために、個々に処理される。

【0056】

図6を参照すると、誤りを有するとして示された行をECC行よりも多く有するデータアレイを訂正するための例示的なプロセス600が示されている。より特定的には、k個の行が誤りを有し、データアレイがn個のECC行を有することをデータアレイのCRC列が示す場合、kはnよりも大きい。このため、データアレイを従来の手法でECC行を用いて訂正することはできない。

【0057】

ステップ602で、データアレイ内の1列につき、CRC列によって誤りを有するとして示された1つ以上の行が選択される。選択される行の数はECC行の数と等しい。

【0058】

ステップ604で、選択された行についてのデータが、ECC行を用いて生成される。

非選択行（つまり、ステップ602で選択されなかった、誤りを有するとして識別された行）内のデータは正しいと仮定される。

【0059】

ステップ606で、選択行についての生成されたデータを用いて、誤りを有するとして示された行を検証する。より特定のには、選択行についての生成されたデータが、元々選択行にあったデータと比較される。これらの行のすべてについて、生成されたデータと元のデータとが同じであるならば、この列の選択行および非選択行のうちのいずれにおいても誤りはない（つまり、この列についての選択行および非選択行内のデータは正しい）。

【0060】

ステップ608で、誤りを有するとして示されたどの行も、ステップ606で実際には誤りを持っていなかった場合、処理されるべき別の列があれば別の列が処理される。たとえば、図2を参照すると、第1の列206を処理した後で、第2の列206が処理可能である。このように、データアレイ202の各列206が処理可能である。しかしながら、列206はどの順序でも処理可能であることが認識されるべきである。

【0061】

図4を参照すると、減少したデータアレイ400にプロセス600（図6）を適用する場合、ステップ606（図6）における判断は決して否定的にはならない。なぜなら、データアレイ400（図4）の各列206は、誤りを有する少なくとも1つの行を含むはずだからである。このため、データアレイ400にプロセス600（図6）を適用する場合、ステップ606および608（図6）は省略可能である。

【0062】

図6を再度参照すると、ステップ610で、選択行について生成されたデータのうちのいずれかが、元々選択行にあったデータと整合するかどうかについて、判断がなされる。ステップ610で選択行について生成されたデータと元のデータとがどれも整合しない場合には、選択行および非選択行についてのデータは、おそらく正しくない。

【0063】

このため、ステップ612で、選択行の可能な組合せのすべてが試みられたかどうかについて判断がなされる。ステップ614で、行の別の組合せが選択され、プロセスが繰返される。ステップ616で、処理すべき別の列がある場合には別の列が処理される。

【0064】

ステップ610で選択行のうちのいずれかについて生成されたデータと元のデータとが同じである場合には、残りの行（つまり、生成されたデータと元のデータとが整合しなかった選択行）について生成されたデータは、これらの行についての正しいデータであり、非選択行についてのデータも正しい。このため、ステップ618で、生成されたデータと元のデータとが整合しなかった選択行内のデータは、これらの行についての生成されたデータと置き換えられる。ステップ620で、これらの行（つまり、訂正されたデータと元のデータとが整合しなかった選択行）用のCRCが再計算される。

【0065】

ステップ622で、再計算されたCRCを含め、CRCによって誤りを有するとして示された行の数がECC行の数よりも大きいかどうかについて判断がなされる。ステップ624で、誤りを有するとして示された行の数が依然としてECC行の数よりも大きい場合、処理すべき別の列が残っている場合には別の列が処理される。ステップ626で、新しいCRCによって誤りを有するとして示された行の数がECC行の数より大きくない場合、データアレイはここでECC行を用いて訂正可能である。

【0066】

プロセス600をより明確に説明するため、図2を参照して、データアレイ202は4つのECC行210（つまり $n = 4$ ）を含み、CRC列214は、5つの行204（たとえば1番目、3番目、5番目、6番目、および16番目の行）が誤りを有する（つまり $k = 5$ ）ことを示していると仮定されたい。より特定のには、1番目、3番目、5番目、6番目、および16番目の行に対応するデータブロックを検索した後で、検索されたデータ

ブロックの各々に対して新しいCRCが生成されると仮定されたい。検索されたデータブロックを記憶媒体106(図1)に記憶させる前に、新しいCRCは、検索されたデータブロック用に元々生成されたCRCと比較され、これらの行204についての新しいCRCと元々生成されたCRCとが整合しなかった場合、それは、検索されたデータブロック内のデータが元々データブロックに記憶されたデータと同じではないことを示す。

【0067】

このため、この例では、ステップ602(図6)で、データアレイ202内の1列206につき、誤りを有するとして示された行のうちの4つが選択される。なぜなら、ECC行210が4つあるためである。例示のため、データアレイ202の1番目の列が選択され、1番目の列の3番目、5番目、6番目、および16番目の行が選択されると仮定されたい。

10

【0068】

ステップ604(図6)で、1番目の列の3番目、5番目、6番目、および16番目の行に対応する新しいデータが、4つのECC行を用いて生成される。この例では、1番目の列の3番目、5番目、6番目、および16番目の行についての元のデータをそれぞれA、B、C、およびDと仮定し、新しい生成データをA、B、C、およびDと仮定されたい。1番目の列内のデータは正しいデータを含むと仮定される。

【0069】

ステップ606で、生成データ(つまりA、B、C、およびD)と元のデータ(つまりA、B、C、およびD)とを用いて、3番目、5番目、6番目、または16番目の行が実際に誤りを有するかどうかを判断する。より特定的には、A、B、C、およびDがA、B、C、およびDと比較される。

20

【0070】

ステップ608で、A、B、C、およびDとA、B、C、およびDとの比較が全データの整合をもたらす場合、1番目の列には誤りはなかった。このため、2番目の列といった別の列が処理される。

【0071】

ステップ610で、比較されたデータのうちのいずれかが整合したかどうかについて判断がなされる。比較されたデータがどれも整合しなかった場合、次にステップ612で、選択行および非選択行の可能な組合せのすべてがなされたかどうかについて判断がなされる。この例では、誤りを有するとして示された行は5つあり、ECC行は4つあるため、選択行および非選択行の可能な組合せは合計で5つある。より特定的には、以下の表は選択行および非選択行のさまざまな組合せを記載する。

30

【0072】

【表1】

表1

組合せ	選択	非選択
1	3番目、5番目、6番目、16番目	1番目
2	1番目、5番目、6番目、16番目	3番目
3	1番目、3番目、6番目、16番目	5番目
4	1番目、3番目、5番目、16番目	6番目
5	1番目、3番目、5番目、6番目	16番目

40

【0073】

ステップ614で、選択行および非選択行の別の組合せが使用される。この例では、表1に記載された組合せ1を使用した後で、組合せ2が使用可能である。こうして、ステップ614で、1番目の列の3番目の行がここで正しいデータを含むと仮定される。ステッ

50

ブ 6 0 4 を繰返して、1 番目、5 番目、6 番目、および 1 6 番目の行についてのデータが、E C C 行を用いて生成される。

【 0 0 7 4 】

ステップ 6 1 6 で、すべての組合せが試みられていた場合、別の列が処理される。たとえば、1 番目の列を処理した後で、2 番目の列が処理可能である。

【 0 0 7 5 】

ステップ 6 1 0 で 3 番目の列だけについて生成データと元のデータとが整合した（つまり $A = A$ ）と仮定されたい。このため、ステップ 6 1 8 で、5 番目、6 番目、および 1 6 番目の行についてのデータはそれぞれ、B、C および D と置き換えられる。ステップ 6 2 0 で、5 番目、6 番目、および 1 6 番目の行用の C R C が再計算される。

10

【 0 0 7 6 】

ステップ 6 2 2 で、再計算された C R C を含め、C R C によって誤りを有するとして示された行の数が依然として E C C 行の数よりも大きいかどうかについて判断がなされる。このため、この例では、5 番目、6 番目、および 1 6 番目の行用の再生成された C R C のすべてが依然として、これらの行が誤りを有することを示す場合、ステップ 6 2 4 で、処理されるべき別の列がある場合には別の列が処理される。しかしながら、5 番目、6 番目、および 1 6 番目の行用の再生成された C R C のうちのいずれか 1 つが、これらの行のどれにももはや誤りがないことを示す場合、ステップ 6 2 6 で、データアレイは E C C 行を用いて訂正される。

【 0 0 7 7 】

上述の例は、データアレイが 4 つの E C C 行を含み、5 つの行が誤りを有するとして示されたことを仮定した。上述のように、データアレイは任意の数の E C C 行を含むことが可能であり、任意の数の行が誤りを有するとして示され得る。たとえば、データアレイが 4 つの E C C 行を含み、6 つの行が誤りを有するとして示された場合、誤りを有するとして示された行のうちの 4 つが選択され、これらの行についてのデータが 4 つの E C C 行を用いて生成される。2 つの非選択行は次に正しいデータを含むと仮定される。また、上述のように、誤りを有するとして示された行のさまざまな組合せは、列を訂正する際に選択行および非選択行として使用可能である。

20

【 0 0 7 8 】

例示的な実施例が説明されてきたが、この発明の精神および/または範囲から逸脱することなく、さまざまな変更がなされ得る。したがって、この発明は、図面に示された、および上述された特定の形態に限定されるものとして解釈されるべきではない。

30

【 図面の簡単な説明 】

【 0 0 7 9 】

【 図 1 】 例示的な記憶装置に接続された例示的なホスト端末を示す図である。

【 図 2 】 例示的な論理データアレイを示す図である。

【 図 3 】 図 2 の例示的な論理データアレイの一部を示す図である。

【 図 4 】 図 2 の例示的な論理データアレイから派生した、減少した論理データアレイを示す図である。

【 図 5 】 記憶媒体から検索されたデータを訂正する例示的なプロセスを示す図である。

40

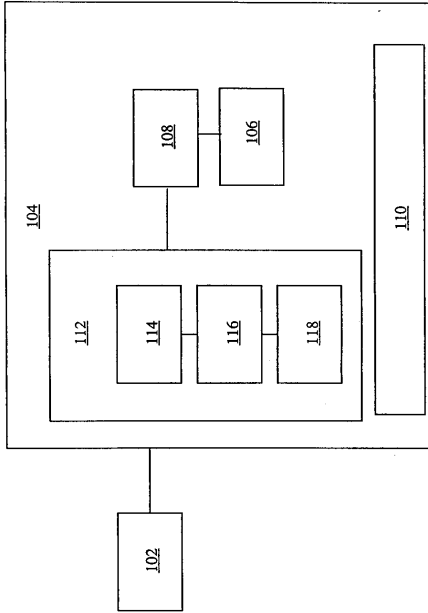
【 図 6 】 記憶媒体から検索されたデータ内の誤りを検証する例示的なプロセスを示す図である。

【 符号の説明 】

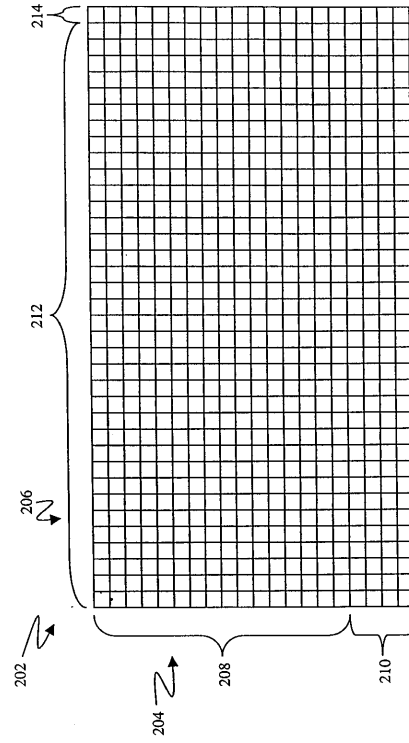
【 0 0 8 0 】

1 0 2 ホスト端末、1 0 6 記憶媒体、1 1 0 プロセッサ、1 1 4 データバッファ、2 0 4 行。

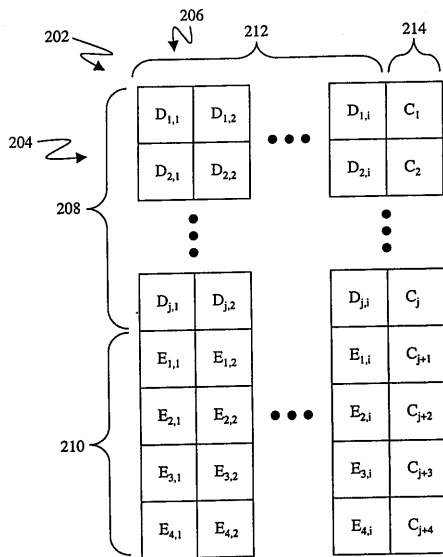
【 図 1 】



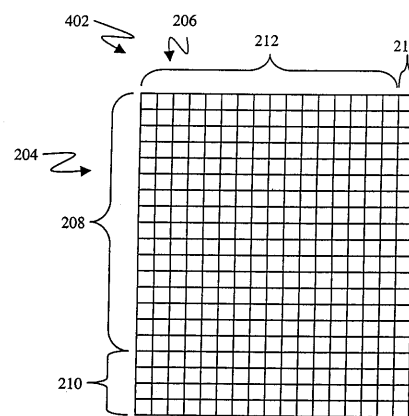
【 図 2 】



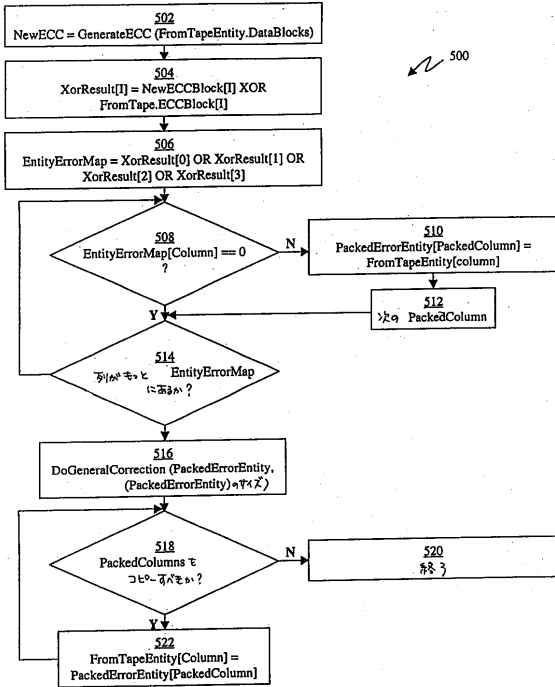
【 図 3 】



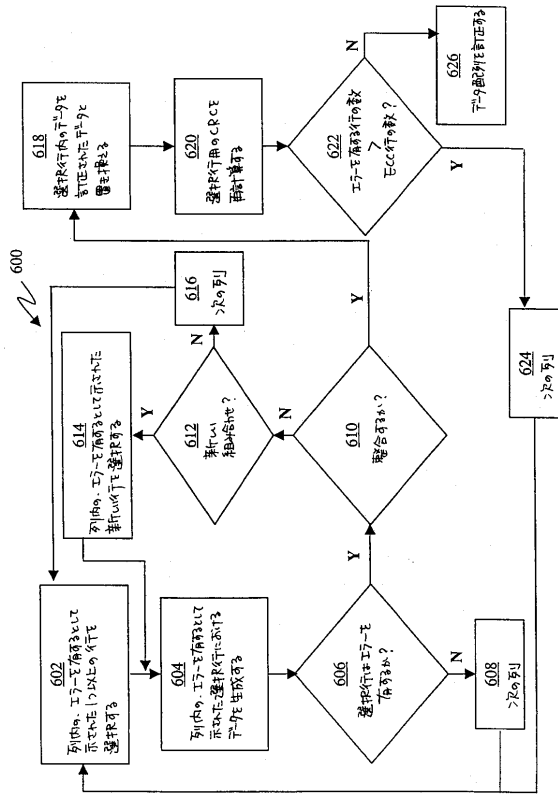
【 図 4 】



【 図 5 】



【 図 6 】



フロントページの続き

(51)Int.Cl.⁷

F I	テーマコード(参考)
G 1 1 B 20/18	5 3 2 E
G 1 1 B 20/18	5 7 2 B
G 1 1 B 20/18	5 7 2 F
G 1 1 B 20/18	5 7 2 G
H 0 3 M 13/09	

(74)代理人 100098316
弁理士 野田 久登

(74)代理人 100109162
弁理士 酒井 将行

(72)発明者 ジョージ・エイ・サリバ
アメリカ合衆国、0 1 5 3 2 マサチューセッツ州、ノースバラ、ハワード・ストリート、1 0 9

Fターム(参考) 5B001 AA03 AA04 AA14 AB01 AD04
5J065 AA01 AB01 AC03 AD11 AE06 AF03