

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第4931810号
(P4931810)

(45) 発行日 平成24年5月16日 (2012.5.16)

(24) 登録日 平成24年2月24日 (2012.2.24)

(51) Int. Cl.	F I
G 0 6 F 12/02 (2006.01)	G 0 6 F 12/02 5 7 0 A
G 0 6 F 12/00 (2006.01)	G 0 6 F 12/00 5 9 7 U
	G 0 6 F 12/00 5 4 2 J

請求項の数 14 (全 29 頁)

(21) 出願番号	特願2007-522509 (P2007-522509)	(73) 特許権者	506197901
(86) (22) 出願日	平成17年6月17日 (2005.6.17)		サンディスク コーポレーション
(65) 公表番号	特表2008-507756 (P2008-507756A)		アメリカ合衆国、95035、カリフォルニア州、ミルピタス、マッカシー ブルバード 601
(43) 公表日	平成20年3月13日 (2008.3.13)	(74) 代理人	100075144
(86) 国際出願番号	PCT/US2005/021846		弁理士 井ノ口 壽
(87) 国際公開番号	W02006/019496	(72) 発明者	コンレー、ケビン エム.
(87) 国際公開日	平成18年2月23日 (2006.2.23)		アメリカ合衆国、95120、カリフォルニア州、サン ホセ、マンクーソ ストリート 6152
審査請求日	平成20年5月12日 (2008.5.12)	(72) 発明者	シンクレア、アラン ウェルシュ
(31) 優先権主張番号	10/897,049		イギリス連邦共和国、FK2 OBU、フオールカーク、マディストン、キャンディ、ブロードヘッド、ザ コテージス
(32) 優先日	平成16年7月21日 (2004.7.21)		最終頁に続く
(33) 優先権主張国	米国 (US)		
(31) 優先権主張番号	11/022,369		
(32) 優先日	平成16年12月23日 (2004.12.23)		
(33) 優先権主張国	米国 (US)		

(54) 【発明の名称】最適化されたシーケンシャルなクラスタの管理のためのFAT分析

(57) 【特許請求の範囲】

【請求項1】

コントローラと不揮発性メモリアレイとを備えたメモリシステムにデータ部分を格納する方法であって、前記コントローラが、

前記不揮発性メモリアレイのファイルアロケーションテーブルまたはディレクトリ内の、ホストによって格納された情報から前記ホストの論理アドレス範囲の属性を取得し、

前記ホストから受信するとともに前記論理アドレス範囲内に論理アドレスを有するデータ部分のための複数の格納方式のなかから、前記論理アドレス範囲の属性に従って格納方式を選択し、複数の格納方式のうちのそれぞれの格納方式には前記不揮発性メモリアレイ内の別のセットの物理アドレスへの前記ホストによって提供される論理アドレスの別のマッピングが含まれ、

前記選択された格納方式に従って前記データ部分を前記不揮発性メモリアレイに格納し、前記データ部分の物理位置を示すマッピング情報も格納するように構成し、

前記論理アドレス範囲の属性とは、前記論理アドレス範囲が論理アドレスの順にホストデータの格納を行うためのシーケンシャルな次の未使用の論理アドレス範囲である属性のことであり、前記シーケンシャルな次の未使用の論理アドレス範囲とは、データが前もって格納された最後の論理アドレス範囲の後の範囲のことである方法。

【請求項2】

請求項1記載の方法において、

前記選択された格納方式は、シーケンシャルな更新ブロックで前記データ部分を格納す

ることを含む方法。

【請求項 3】

請求項 1 記載の方法において、

前記論理アドレス範囲の属性は、前記ファイルアロケーションテーブルまたはディレクトリに格納されたデータから導き出されたデータ属性格納部から取得される方法。

【請求項 4】

請求項 1 記載の方法において、

前記データ部分を前記不揮発性メモリアレイ内の第 1 の記憶位置に前もって格納し、前記論理アドレス範囲の属性とは、前記論理アドレス範囲が前記ホストによって割り当てられる属性のことであり、前記選択された格納方式には、前記不揮発性メモリアレイ内の第 1 の記憶位置から第 2 の記憶位置へ前記データ部分をコピーすることが含まれる方法。

10

【請求項 5】

請求項 4 記載の方法において、

前記データ部分を前記不揮発性メモリアレイ内の第 1 の記憶位置に前もって格納し、前記論理アドレス範囲の属性とは、前記論理アドレス範囲が前記ホストによって割り当てを解除される属性のことであり、前記選択された格納方式は、前記不揮発性メモリアレイ内の第 1 の記憶位置から第 2 の記憶位置へ前記データ部分をコピーすることなく、第 1 の記憶位置で前記データ部分の保守管理を行う方法。

【請求項 6】

コントローラと不揮発性メモリアレイとを備えたメモリシステムにデータ部分を格納する方法であって、前記コントローラが、

20

前記不揮発性メモリアレイのファイルアロケーションテーブルまたはディレクトリ内の、ホストによって格納された情報から前記ホストの論理アドレス範囲の属性を取得し、

前記ホストから受信するとともに前記論理アドレス範囲内に論理アドレスを有するデータ部分のための複数の格納方式のなかから、前記論理アドレス範囲の属性に従って格納方式を選択し、複数の格納方式のうちのそれぞれの格納方式には前記不揮発性メモリアレイ内の別のセットの物理アドレスへの前記ホストによって提供される論理アドレスの別のマッピングが含まれ、

前記選択された格納方式に従って前記データ部分を前記不揮発性メモリアレイに格納し、前記データ部分の物理位置を示すマッピング情報も格納するように構成し、

30

前記論理アドレス範囲の属性とは、前記論理アドレス範囲が論理アドレスの順にホストデータの格納を行うためのシーケンシャルな次の未使用の論理アドレス範囲ではない属性のことである方法。

【請求項 7】

請求項 6 記載の方法において、

前記選択された格納方式は、カオス更新ブロックで前記データ部分を格納することを含む方法。

【請求項 8】

不揮発性メモリシステムであって、

その中にデータを格納する不揮発性メモリアレイと、

40

コントローラであって、

前記不揮発性メモリアレイのファイルアロケーションテーブルまたはディレクトリ内の、ホストによって格納された情報から前記ホストの論理アドレス範囲の属性を取得し、

前記ホストから受信するとともに前記論理アドレス範囲内に論理アドレスを有するデータ部分のための複数の格納方式のなかから、前記論理アドレス範囲の属性に従って格納方式を選択し、複数の格納方式のうちのそれぞれの格納方式には前記不揮発性メモリアレイ内の別のセットの物理アドレスへの前記ホストによって提供される論理アドレスの別のマッピングが含まれ、

前記選択された格納方式に従って前記データ部分を前記不揮発性メモリアレイに格納し、前記データ部分の物理位置を示すマッピング情報も格納するように構成されたコント

50

ローラと、を備え、

前記論理アドレス範囲の属性とは、前記論理アドレス範囲が論理アドレスの順にホストデータの格納を行うためのシーケンシャルな次の未使用の論理アドレス範囲である属性のことであり、前記シーケンシャルな次の未使用の論理アドレス範囲とは、データが前もって格納された最後の論理アドレス範囲の後の範囲のことであり、不揮発性メモリシステム。

【請求項 9】

請求項 8 記載の不揮発性メモリシステムにおいて、

前記選択された格納方式は、シーケンシャルな更新ブロックで前記データ部分を格納することを含む不揮発性メモリシステム。

【請求項 10】

請求項 8 記載の不揮発性メモリシステムにおいて、

前記論理アドレス範囲の属性は、前記ファイルアロケーションテーブルまたはディレクトリに格納されたデータから導き出されたデータ属性格納部から取得される不揮発性メモリシステム。

【請求項 11】

請求項 8 記載の不揮発性メモリシステムにおいて、

前記データ部分を前記不揮発性メモリアレイ内の第 1 の記憶位置に前もって格納し、前記論理アドレス範囲の属性とは、前記論理アドレス範囲が前記ホストによって割り当てられる属性のことであり、前記選択された格納方式には、前記不揮発性メモリアレイ内の第 1 の記憶位置から第 2 の記憶位置へ前記データ部分をコピーすることが含まれる不揮発性メモリシステム。

【請求項 12】

請求項 11 記載の不揮発性メモリシステムにおいて、

前記データ部分を前記不揮発性メモリアレイ内の第 1 の記憶位置に前もって格納し、前記論理アドレス範囲の属性とは、前記論理アドレス範囲が前記ホストによって割り当てを解除される属性のことであり、前記選択された格納方式は、前記不揮発性メモリアレイ内の第 1 の記憶位置から第 2 の記憶位置へ前記データ部分をコピーすることなく、第 1 の記憶位置で前記データ部分の保守管理を行う不揮発性メモリシステム。

【請求項 13】

不揮発性メモリシステムであって、

その中にデータを格納する不揮発性メモリアレイと、

コントローラであって、

前記不揮発性メモリアレイのファイルアロケーションテーブルまたはディレクトリ内の、ホストによって格納された情報から前記ホストの論理アドレス範囲の属性を取得し、

前記ホストから受信するとともに前記論理アドレス範囲内に論理アドレスを有するデータ部分のための複数の格納方式のなかから、前記論理アドレス範囲の属性に従って格納方式を選択し、複数の格納方式のうちのそれぞれの格納方式には前記不揮発性メモリアレイ内の別のセットの物理アドレスへの前記ホストによって提供される論理アドレスの別のマッピングが含まれ、

前記選択された格納方式に従って前記データ部分を前記不揮発性メモリアレイに格納し、前記データ部分の物理位置を示すマッピング情報も格納するように構成されたコントローラと、を備え、

前記論理アドレス範囲の属性とは、前記論理アドレス範囲が論理アドレスの順にホストデータの格納を行うためのシーケンシャルな次の未使用の論理アドレス範囲ではない属性のことであり、不揮発性メモリシステム。

【請求項 14】

請求項 13 記載の不揮発性メモリシステムにおいて、

前記選択された格納方式は、カオス更新ブロックで前記データ部分を格納することを含む不揮発性メモリシステム。

【発明の詳細な説明】

10

20

30

40

50

【技術分野】

【0001】

本発明は、一般に、計算システムと共に用いられるデータ格納システムに関し、特に、不揮発性メモリ記憶装置においてデータの保守管理を行うための技法に関する。

【背景技術】

【0002】

特に、小形の形状のファクタカードの形で今日使用されている商業的に成功した多くの不揮発性メモリ製品が存在する。これらの不揮発性メモリ製品には、1以上の集積回路チップ上に形成されたフラッシュEEPROMアレイ（電氣的に消去可能でプログラム可能なリードオンリメモリ）セルが使用されている。典型的には、個々のホストとのインターフェイスとして集積回路チップ上のメモリコントローラが提供される。ホストは、コントローラを使用して不揮発性メモリで（読み出し、書き込みなどの）種々の処理を行うことができる。コントローラには1以上の特別の回路が設けられ、そのような回路として、典型的には、マイクロプロセッサ、ある種の不揮発性リードオンリメモリ（ROM）、揮発性ランダムアクセスメモリ（RAM）、並びに、データがコントローラの中を通り抜けるとき、データのプログラミング中およびデータの読み出し中にデータから生じる誤り訂正符号（ECC）の計算を行う回路などがある。市販のカードのいくつかとして、コンパクトフラッシュ（登録商標）（CF）カード、マルチメディアカード（MMC）、セキュアデジタル（SD）カード、スマートメディアカード、個人用タグ（personnel tag）（P-Tag）およびメモリスティックカードがある。ホストには、パーソナルコンピュータ、ノート形コンピュータ、携帯用個人情報端末（PDA）、種々のデータ通信装置、デジタルカメラ、携帯電話、携帯用オーディオプレーヤー、自動車用サウンドシステム、および、同種の機器が含まれる。メモリカードの実施構成以外にも、不揮発性メモリは種々のタイプのホストシステムの中へ組み込むことができる。

10

20

【0003】

ほとんどすべての集積回路用アプリケーションにおける場合のように、フラッシュEEPROMメモリセルアレイの場合にも、いくつかの集積回路機能の実現に必要なシリコン基板面積の縮小に対する圧力が存在する。所定のサイズのメモリカードと、別の種類のパッケージとの記憶容量を増やすために、所定の面積のシリコン基板に格納できるデジタルデータ量を増やすようにという、容量の増加とサイズの縮小化の双方を求める要望が継続的に存在する。データの記憶密度を高める1つの方法として、1個のメモリセル当たりおよび/または記憶素子または記憶ユニット当たり2ビット以上のデータを格納する方法がある。これは、記憶素子の電荷レベルの電圧範囲のウィンドウを3以上の状態に分割することにより達成される。このような4状態を利用することにより、個々のセルに2ビットのデータを格納することが可能となり、いずれの状態も、記憶素子当たり3ビットのデータ等々が格納されることになる。フローティングゲートを利用する多状態フラッシュEEPROM構造については、米国特許第5,043,940号（特許文献1）および第5,172,338号（特許文献2）に記載され、さらに、誘電体フローティングゲートを用いる構造については、前述した米国特許出願第2003/0109093号（特許文献3）に記載されている。多状態メモリセルアレイの選択された部分は、米国特許第5,930,167号（特許文献4）および米国特許第6,456,528号（特許文献5）に記載されているように、種々の理由によって2状態（2進）での動作も可能となる。

30

40

【0004】

典型的なフラッシュEEPROMアレイのメモリセルは、一括消去される個別のセル“ブロック”に分割される。個々の消去ブロックは通常1以上のデータページを格納し、この1ページがプログラミングと読み出しとを行う最小単位であるが、同時に異なるサブアレイすなわちプレーンで2以上のページのプログラミングや読み出しを行うことも可能である。個々のページは通常1以上のデータセクタを格納し、このセクタサイズはホストシステムによって定義される。一つの例示のセクタには、磁気ディスク駆動装置に関して定義されている標準規格に準拠する512バイトのユーザデータが含まれ、これに加えて、

50

ユーザデータおよび/またはユーザデータが格納されている消去ブロックに関する或るバイト数のオーバーヘッド情報も含まれている。このようなメモリは、一般に32、128またはそれ以上のページを持つ個々の消去ブロック内に構成され、個々のページには1セクタまたはわずかに数セクタのホストデータが格納される。ブロックに対してデータを再書き込みするためにブロックをまず消去することが望ましいことも留意されたい。

【0005】

ユーザデータをプログラムしてメモリアレイの中へ入れ、このメモリアレイからユーザデータを読み出す間の並行性レベルを高めるために、このアレイはプレーンと呼ばれるサブアレイに通常分割され、これらのサブアレイは、並列処理が可能となるようにサブアレイ自身のデータレジスタ並びにその他の回路を備え、これによって、数個のプレーンまたはすべてのプレーンの各々に対してデータセクタを同時にプログラムしたり、数個のプレーンまたはすべてのプレーンの各々からデータセクタを同時に読み出したりすることができるようになる。単一の集積回路のアレイを物理的に分割してプレーンに変えることも可能であるし、あるいは、別々の1以上の集積回路チップから個々のプレーンを形成することも可能である。このようなメモリの実施構成の例については、米国特許第5,798,968号(特許文献6)および米国特許第5,890,192号(特許文献7)に記載されている。

【0006】

メモリをさらに効率的に管理するために、消去ブロックを仮想ブロックすなわちメタブロックと一体に論理的にリンクすることも可能である。すなわち、個々のプレーンから得られる1つの消去ブロックを含むように個々のメタブロックは定義される。メタブロックの利用については、国際公開特許出願第02/058074号(特許文献8)に記載されている。メタブロックは、データのプログラミングと読み出しのための宛先としてホストの論理ブロックアドレスにより特定される。同様に、メタブロックからなるすべての消去ブロックは一括消去される。このような大きなブロックおよび/またはメタブロックと共に動作するメモリシステム内のコントローラは、ホストから受信した論理ブロックアドレス(LBA)と、メモリセルアレイ内の物理ブロック番号(PBN)との間での変換を含む複数の機能を実行する。このブロック内の個々のページは、典型的には、このブロックアドレス内のオフセット値によって特定される。アドレス変換は、論理ブロック番号(LBN)と論理ページの間項の利用を含む場合が多い。したがって、メモリアレイ内のブロックは単一の消去ブロックから構成されるものであってもよいし、あるいは、メタブロックを形成するために論理的にリンクされる2以上の消去ブロックから構成されるものであってもよい。

【0007】

ホストアプリケーションの視点から見れば、論理アドレス0を起点として論理アドレスNで終る論理アドレス空間にフラッシュメモリを分割することができる。図1はホストの論理アドレス空間102を示し、この論理アドレス空間102は“論理セクタ”に分割されている。ホストは(4~64などの)いくつかの論理セクタをクラスタにグループ化する。図1に示されているように、ホストは、クラスタA、B、C、D、E、F、G、Hを含むクラスタに論理アドレス空間102を分割してもよい。したがって、これらのクラスタの各々は、(クラスタ当たり4~64の論理セクタなどの)所定数のセクタを表わすことができることになる。いずれの場合にも、ホストは、“クラスタ”と呼ばれる論理データ単位でデータを割り当てることになる。例えば、クラスタA、B、Cを用いてファイル1を割り当てるのが可能であり、これに対してクラスタDを用いてファイル2が割り当てられる。ホストがクラスタの中にファイルを割り当てることに留意されたい。しかし、クラスタ内のすべての論理セクタがファイル用としてデータを含む必要があるわけではない。例えば、(唯一の論理セクタなどの)クラスタCの一部だけがデータを含むようにすることも可能である。それでも、たとえクラスタ内のすべての論理セクタが必ずしも必要となるというわけではなくても、ホストは代表的にクラスタ全体を1つのファイルに割り当てることになる。

10

20

30

40

50

【 0 0 0 8 】

ホストはファイルアロケーションテーブル (F A T) 1 0 4 を用いてファイルを追跡することができる。ファイルアロケーションテーブル (F A T) 1 0 4 は論理的ファイル構造を効果的に表わし、ファイルの各々に関連して論理的ファイル構造の中にどのようなクラスタが割り当てられているかをもまた示すものである。ホストは、ファイルアロケーションテーブル (F A T) 1 0 4 の更新によって論理的ファイル構造の保守管理を行う。例えば、新規ファイルが割り当てられると、ファイルアロケーションテーブル (F A T) 1 0 4 が更新されて、特に、どのようなクラスタが新規ファイル用として割り当てられているかが示される。また、ファイルあるいはディレクトリが削除されると、ホストはファイルアロケーションテーブル (F A T) 1 0 4 を更新する。クラスタは、更新の結果、ホストによって割当てが解除される場合もあることに留意されたい。例えば、ホストがファイル 1 を削除すると、更新されたファイルアロケーションテーブル (F A T) 1 0 6 は、ファイル 1 を格納するためにホストが割り当てていたクラスタ A、B、C を効果的に解除するという結果を生み出すことができる。

10

【 0 0 0 9 】

ホストの論理アドレス空間とは対照的に、フラッシュメモリコントローラの論理アドレス空間は“論理ブロック”に分割される。言い換えれば、コントローラ用の同じ論理アドレス空間 1 0 2 を論理アドレス空間 1 0 3 として表わすことも可能である。論理アドレス空間 1 0 2 と論理アドレス空間 1 0 3 の双方の基本単位はこの例ではセクタである。図 1 に示されているように、論理アドレス空間 1 0 3 は、論理ブロック 1 と論理ブロック 2 とを含む種々の論理ブロックに分割することも可能であり、個々の論理ブロックには、メモリアレイの単一ブロックに対応づけられる複数のデータセクタが含まれる。これらの論理ブロックは、例えば、単一の消去ブロックやメタブロックのような単位として消去することができるメモリブロックに対応づけられる。したがって、論理ブロックとメモリアレイのブロックとの間に 1 対 1 の対応関係が存在することになり、その場合、論理ブロック内のすべてのセクタは単一ブロックに格納される。例えば、論理ブロック 1 を対応づけてブロック 2 にするようにしてもよいなどである。要するに、ホストとコントローラとが異なる論理メモリ単位を使用することになる。ホストが論理アドレス空間を分割してクラスタにするのに対して、コントローラは論理アドレス空間を分割して論理ブロックにする。典型的には、アプリケーション用データおよびサブディレクトリエントリのみに対して割り当てられる論理アドレス空間にホストクラスタが適用されることにも留意されたい。さらに、別のシステムタ用として割り当てられた論理アドレス空間をホストがクラスタとして管理しなくてもよくなる。通常、第 1 のクラスタは論理アドレス空間の冒頭に位置しなくてもよいことにも留意されたい。したがって、クラスタと論理ブロックとは互いに一直線にならなくてもよくなる。

20

30

【 0 0 1 0 】

これに対して、従来方式のコントローラはホストが利用する論理構成を認知するものではない。従来方式では、コントローラは、ホストが使用する論理アドレス 1 0 2 を単に論理アドレス 1 0 3 に対応づけ、次いで、物理アドレス 1 1 0 にするにすぎない。これは、従来方式のコントローラが、ホストの論理データ構成や、ホストの処理がデータに対して及ぼしたかもしれない悪い結果を考慮していないことを意味する。例えば、新規ファイルが割り当てられると、論理的ファイルシステム 1 0 4 は更新されて、特に、どのようなクラスタが新規ファイル用として割り当てられたかが示されることになる。また、ファイルやディレクトリが削除されると、ホストは論理的ファイルシステム 1 0 4 の更新を行う。さらに、1 以上のクラスタがホストにより割当てを解除される場合もある。例えば、ホストがファイル 1 を削除すると、更新されたファイルアロケーションテーブル 1 0 6 によって、ファイル 1 を格納するために前回割り当てられたクラスタ A、B、C の割当てをホストが効果的に解除したことが示される (すなわち、ファイル 1 は更新されたファイルアロケーションテーブル 1 0 6 にはもはや表されない)。ホストによってクラスタ A、B、C が割当てを解除され、消去が可能になるとはいえ、従来方式のコントローラによって、割

40

50

当て解除済みのクラスタ A、B、C に対応する当該データの“ガーベッジコレクション”は行われなくなる。これは、データがメモリにそのまま持続して存在することを意味する。ガーベッジコレクションとは、古いデータセクタを含む原ブロックから更新ブロックへ有効なデータセクタのコピーを行う処理である。すべての有効なデータセクタがコピーされると、原ブロックは消去用としてマークすることができる。典型的には、データは 2 以上のブロックから単一ブロックへ統合され、それによってさらに効率的にデータが格納され、メモリアレイ内の空間が解放されることになる。ガーベッジコレクション中にデータの unnecessary コピーを行うことはこのような処理の効率を下げることになる。さらに、2 つのブロック内のデータが 1 つのブロックに統合される際、ガーベッジコレクション中にブロックの更新が行われると、割当て解除済みのデータが複数回コピーされる可能性がある

10

【0011】

例示によってさらに説明すると、原ブロック 120 が図 1 に描かれている。説明を簡略にするために、原ブロック 120 には、ファイル 1 と 2 を割り当てるためにホストが使用する論理クラスタ A、B、C、D に対応するデータが含まれていると想定する。また、別の想定として、ホストがファイル 1 (クラスタ A、B、C) の割当てが遅れて解除されたと想定する。この想定によって、原ブロック 120 には、すべての論理クラスタ A、B、C、D がまだ含まれていることになる。というのは、コントローラには、ファイル 1 がホストによって割当てを解除されたことを知る方法がないからである。これに対して、ホストは、原ブロック 120 のデータ部分 D の更新をコントローラに要求している。言い換えれば、ある時点において、ホストは、ファイル 2 を変更し、それによって、1 以上の論理セクタに対応するデータ用として更新すべきものであることがわかっている論理アドレスを用いて、書き込み要求の送信を望むことが可能となる。コントローラは、ホストによりアドレス指定されたこれらの論理セクタがホストによってすでに書き込まれたセクタ (すなわち、クラスタ D) に対応するものであるため、更新ブロック 122 が部分的に用いられていると判定する。というのは、原ブロック 120 内のデータの上書きは不可能であるからである。このことは、更新ブロック 122 が“閉じられる”(すなわち、原ブロック 120 内のすべてのデータがブロック 122 へコピーされる)までのしばらくの間、2 つのブロックを用いて論理ブロックを表わすことができることを意味する。例えば、ガーベッジコレクションサイクル中に原ブロック 120 と更新ブロック 122 とを組み合わせ、それによって更新ブロック 122 が“原”ブロック 122 になり、前の原ブロック 120 が消去され、利用可能なメモリブロックのプールへ戻されるようにすることも可能である。いずれの場合にも、割当てを解除されたクラスタ (A、B、C) に対応するデータが更新ブロック 122 へ遅かれ早かれコピーされ、その結果更新ブロック 122 を閉じることが可能となり、原ブロック 120 を消去できるようになることに留意されたい。コントローラは、揮発性メモリや、論理データブロックへのメモリブロックの割当てを定義する不揮発性メモリ内のマッピングテーブルの保守管理を行うことも可能である。不揮発性メモリに格納されたデータの複製処理にはかなりの量の時間とリソースとを必要とする。しかし、(割当て解除済みのクラスタ A、B、C などの)割当てを解除されたデータの複製処理を行う必要はなくなる。したがって、割当て解除済みのデータを複製しなければ、システムパフォーマンスのさらなる改善が可能となる。一般に、広く利用されているアドレス規定に従いながらできるかぎりシステム性能の改善を図ることが望ましい。

20

30

40

【0012】

したがって、代替の不揮発性メモリ管理技法は、有益なものとなる。

【特許文献 1】米国特許第 5,043,940 号

【特許文献 2】米国特許第 5,172,338 号

【特許文献 3】米国特許出願第 2003/0109093 号

【特許文献 4】米国特許第 5,930,167 号

【特許文献 5】米国特許第 6,456,528 号

【特許文献 6】米国特許第 5,798,968 号

50

【特許文献7】米国特許第5,890,192号

【特許文献8】国際公開特許出願第02/058074号

【特許文献9】米国特許出願第10/750,155号

【特許文献10】米国特許出願第10/749,831号

【発明の開示】

【0013】

概括的に言えば、本発明は、(フラッシュメモリなどの)不揮発性メモリシステムにおいてデータ管理を行うための技法に関する。本発明の1つの態様によれば、コントローラはホストのファイルシステムに関する情報を利用することが可能となり、この情報は、ホストによって不揮発性メモリに格納されて、1以上のクラスタ(または1以上のクラスタ内の1以上のセクタ)が現在割り当てられているかどうかの判定が行われることになる。理解されるように、コントローラを用いて、ホストのファイルシステムに関する情報を利用して、特に、1以上のクラスタ(または1以上のセクタ)が、例えば、ガーベッジコレクションサイクル中にメモリ内の1つの記憶位置から別の記憶位置へコピーされないようにすることを保証することが可能となる。この結果、従来行われた(データのコピーなどの)いくつかの不必要な処理の回避が可能となり、システム性能が改善されることになる。

10

【0014】

コントローラは、以前に割り当てられた1以上のクラスタが割当てを解除されたかどうかの判定を行うために、ホストがファイルアロケーションテーブル(FAT)に正常に書き込んだ情報を直接利用することが可能であることも理解できる。前述したのとは別に、コントローラは、ルートディレクトリ、サブディレクトリ並びにファイルアロケーションテーブル(FAT)に書き込まれた情報を利用して、例えば、データアロケーションテーブル(DAT)内のクラスタアロケーションに関するコントローラ自身の情報の生成および/または保守管理を行うことも可能である。しかし、本発明を実現して、ファイルアロケーションテーブル(FAT)に正常に格納された情報の利用が可能となることは明らかである。したがって、コントローラがコントローラ自身の情報の保守管理を行ったり、(データアロケーションテーブルなどの)何らかの追加のデータ構造を生成したりすることは不要となる。しかし、以下説明するように、データアロケーションテーブル(DAT)の保守管理を行うことによって、ファイルアロケーションテーブル(FAT)から得られる情報に単に依拠するよりも多くの利益を得ることが可能となる。したがって、例えばデータアロケーションテーブル(DAT)として実現できるデータ属性格納部(DAS)についてさらに説明することにする。

20

30

【0015】

一般に、データ属性格納部(DAS)には、コントローラが不揮発性メモリの中で保守管理を行う1以上のデータ属性が含まれる。データ属性は、(割当て状態や割当て解除状態、サイズ、優先順位などの)データに関する有益情報を提供することができる。したがって、データ属性格納部(DAS)を用いてさらにインテリジェントにデータ管理を行うことが可能となる。本発明の別の態様は、データ属性格納部(DAS)を用いてデータの管理を行うための技法に関する。

40

【0016】

1つの実施形態では、データ属性格納部(DAS)は割当て格納部(AS)として実現される。割当て格納部(AS)は、例えばテーブル、すなわち個々のデータクラスタ(ホストの論理単位)の割当て状態を示すデータアロケーションテーブル(DAT)として実現することができる。クラスタとは、論理的にデータを構成するためにホストが用いる論理単位を表わすものである。より良好なメモリ管理を達成するために、不揮発性メモリシステムのコントローラはデータアロケーションテーブル(DAT)にアクセスを行うことができる。したがって、データアロケーションテーブル(DAT)は、ホストが利用する論理構成とコントローラとの間の橋渡しを効果的に行うものである。こうすることによって、コントローラはホストの論理構成を効果的に理解することが可能となる。さらに、コ

50

ントローラはホストの活動をモニタして、例えば、ホストがクラスタの割当てを解除されたかどうかの判定を行う。例えば、DOS対応システムのルートディレクトリ、ファイルアロケーションテーブル(FAT)およびサブディレクトリをモニタして、ホストによってデータのクラスタの割当て解除の検知を図ることが可能となる。例えば、ファイルまたはファイルの一部がホストによって削除された結果として、ルートディレクトリ、ファイルアロケーションテーブル(FAT)あるいはサブディレクトリの変更が生じる場合がある。いずれの場合にも、1以上のデータクラスタの割当て解除が検知されると、コントローラはデータアロケーションテーブル(DAT)を更新する。この結果、データアロケーションテーブル(DAT)はデータに関する更新情報を提供できることになる。さらにインテリジェントなデータ管理のためにこの情報の利用が可能となる。例えば、ガーベッジコレクション中にデータのコピーを行うべきかどうかを判定するために割当て解除状態を利用することができる。割当てが解除されたデータのコピーは行う必要がない。この結果、従来方式で行われるいくつかの不必要な処理を回避することが可能となり、システム性能を改善することが可能となる。

10

【0017】

別の実施形態では、ホストによってFATまたはDASに格納された情報を利用して、ホストが送信したデータの一部をどのように格納するのが望ましいかを決定することができる。どこか別の場所からデータをコピーすることによって、非シーケンシャルに受信したデータを非シーケンシャルに格納したり、シーケンシャルに格納したりして、何らかのギャップを受信したデータに充填するように図ることも可能である。このようなシーケンシャルな格納が利点を有する場合として、ホストが、利用可能な(未使用のあるいは割当てを解除された)(クラスタなどの)論理アドレス空間の一部へ書き込みを行う場合がある。データのこのような書き込みは、ホストが新しいデータを格納していることを示すものであり、非シーケンシャルにデータを格納する何らかの理由を示すものではない。ホストが送信するデータセクタの論理アドレスを未使用クラスタの論理アドレス範囲と比較することによって、コントローラは、ホストが次の未使用クラスタに対して書き込みを行っているかどうかを判定することが可能となる。この判定に基づいて、コントローラはセクタ用の格納方式を選択することも可能である。

20

【0018】

1つの実施形態では、ホストが次の未使用クラスタへの書き込みを行っていることがホストから受信した単複のセクタによって示されているかどうかに関して判定が行われる。ホストが次の未使用クラスタへの書き込みを行っていれば、メモリアレイ内のどこか別の場所からコピーされたデータによって、データ内のいずれのギャップも満たされた状態で、データはシーケンシャルな形でブロックに書き込まれていることになる。ホストが次の未使用クラスタへの書き込みを行っていなければ、データをシーケンシャルにあるいは非シーケンシャルに格納すべきかどうかを判定するために追加の基準を考慮してもよい。例えば、セクタ間の論理アドレス内でジャンプしてホストからセクタを受信する場合、ジャンプのサイズによって格納方式を決定することができる。ジャンプが小さなものであれば、結果として生じるギャップはコピーされたデータで満たすことも可能であり、さらに、シーケンシャルなフォーマットで更新済みデータを保持するようにしてもよい。更新ブロックがシーケンシャルで、かつ、満たされるように近接している別の例では、更新ブロックをデータで満たして、非シーケンシャルに受信したセクタに書き込みを行うのではなく、更新ブロックをシーケンシャルに保つようにすることも可能である。不揮発性メモリ内のFATのコピーから直接取得した情報に基づいて格納方式を選択したり、DAS内の、または、FATから導き出された類似の構造内の情報に基づいて格納方式を選択したりすることも可能である。

30

40

【0019】

方法、システム、デバイス、装置あるいはコンピュータ可読媒体として含まれる本発明は多くの態様で実現することが可能である。本発明の他の態様並びに利点は、本発明の原理を例示する添付図面と関連して行う以下の詳細な説明から明らかになる。

50

【発明を実施するための最良の形態】

【0020】

同じ参照番号によって類似の要素が参照される添付図面を例示として本発明を示すが、これは本発明を限定するものではない。

【0021】

本発明は、(フラッシュメモリなどの)不揮発性メモリシステムにおいてデータを管理するための技法に関する。本発明の1つの態様によれば、コントローラはホストのファイルシステムに関する情報を利用することが可能であり、この情報は、1以上のクラスタ(1以上のクラスタ内の1以上のセクタ)が現在割り当てられているかどうかを判定するために、ホストによって不揮発性メモリに格納されたものである。理解されるように、コントローラは、ホストのファイルシステムに関する情報を利用して、特に、例えば、ガーベッジコレクションサイクル中などに1以上のクラスタ(またはクラスタ内の1以上のセクタ)が、メモリ内の1つの記憶位置から別の記憶位置へコピーされないことを保証することができる。この結果、従来方式で行われる(データをコピーするなどの)いくつかの不必要な処理を回避することが可能となり、システム性能を改善することが可能となる。

【0022】

1以上の予め割り当てられたクラスタが割当てを解除されたかどうかを判定するために、ファイルアロケーションテーブル(FAT)にホストが正常に書き込んだ情報をコントローラが直接利用できることも理解できる。前述したのとは別に、コントローラは、ルートディレクトリ、サブディレクトリおよびファイルアロケーションテーブル(FAT)に書き込まれた情報を利用して、例えば、データアロケーションテーブル(DAT)の中でファイルアロケーションテーブル(FAT)自身のクラスタアロケーションに関する情報の生成および/または保守管理を行うようにすることが可能となる。しかし、本発明を実現して、ファイルアロケーションテーブル(FAT)に正常に格納された情報を利用することは明らかである。したがって、コントローラがコントローラ自身の情報の保守管理を行ったり、(データアロケーションテーブル(DAT)などの)何らかの追加のデータ構造を生成したりすることは不要となる。しかし、以下説明するように、データアロケーションテーブル(DAT)の保守管理を行うほうが、ファイルアロケーションテーブル(FAT)から得られる情報に単に依拠する場合よりも多くの利点を提供できることになる。したがって、例えばデータアロケーションテーブル(DAT)として実装できるデータ属性格納部(DAS)についてもさらに例示によって説明することにする。

【0023】

一般に、1以上のデータ属性を含むデータ属性格納部(DAS)の保守管理はコントローラによって不揮発性メモリの中で行うことが可能である。データ属性は、(割当て状態や割当て解除状態、サイズ、優先順位などの)データに関する有益情報を提供することができる。したがって、データ属性格納部(DAS)を用いてさらにインテリジェントにデータ管理を行うことが可能となる。本発明の別の態様は、データ属性格納部(DAS)を利用するデータ管理のための技法に関する。

【0024】

1つの実施形態では、データ属性格納部(DAS)は割当て格納部(AS)として実装される。割当て格納部(AS)は、例えば、テーブルとしてあるいはデータの個々のクラスタ(ホストの論理単位)の割当て状態を示すデータアロケーションテーブル(DAT)として実装することができる。クラスタとは、ホストがデータを論理的に構成するために使用する論理単位を表わすものである。より好適なメモリ管理を行うために、不揮発性メモリシステムのコントローラはデータアロケーションテーブル(DAT)にアクセスすることができる。したがって、データアロケーションテーブル(DAT)は、ホストが利用する論理構成と、コントローラとの間での橋渡しを効果的に行うことになる。こうすることによって、コントローラはホストの論理構成を効果的に理解できることになる。さらに、コントローラはホストの活動をモニタして、例えば、ホストがクラスタの割当てを解除したかどうかの判定を行う。例えば、ルートディレクトリ、ファイルアロケーションテー

10

20

30

40

50

ブル (FAT) および DOS 対応システムのサブディレクトリをモニタして、データのクラスタの割当て解除をホストが検知するように図ることも可能である。例えば、ファイルまたはファイルの一部がホストにより削除された結果として、ルートディレクトリ、ファイルアロケーションテーブル (FAT) あるいはサブディレクトリの変更が生じる場合がある。いずれの場合にも、データの 1 以上のクラスタの割当て解除が検知されると、コントローラはデータアロケーションテーブル (DAT) の更新を行うことになる。この結果、データアロケーションテーブル (DAT) はデータに関する更新された情報を提供できることになる。この情報を利用して、さらにインテリジェントなデータ管理が行われることになる。例えば、割当て解除状態を利用して、ガーベッジコレクション中にデータをコピーすべきかどうかを判定することができる。割当てを解除されたデータのコピーを行う必要はない。その結果、従来方式で行われるいくつかの不必要な処理を回避することが可能となり、システム性能を改善することが可能となる。

10

【0025】

図 2A ~ 図 8 を参照しながら以下本発明の実施形態について論ずることとする。しかし、当業者であれば容易に理解できるように、これらの図面によって本願明細書で行う以下の詳細な説明は、あくまで説明のためのものであって、本発明はこれらの限定された実施形態の範囲を越えて拡大するものである。例えば、この説明はフラッシュメモリに言及するものではあるが、別の種類の不揮発性メモリを使用することも可能である。別の種類の不揮発性メモリとしては、磁気抵抗 RAM (MRAM)、強誘電体 RAM、および (オボニクス単一化メモリ (Ovonics Unified Memory) すなわち OUM としても公知の) 相変化メモリが含まれる。

20

【0026】

前述したように、コントローラはホストのファイルシステムに関する情報を利用することが可能であり、この情報はホストによって不揮発性メモリに格納され、1 以上のクラスタ (またはクラスタ内のセクタ) が現在割り当てられているかどうかの判定が本発明の 1 つの態様に従って行われる。コントローラは、ホストのファイルシステムに関する情報を利用して、特に、ガーベッジコレクションサイクル中にメモリ内の 1 つの記憶位置から別の記憶位置へ 1 以上のクラスタ (またはクラスタ内のセクタ) がコピーされないようにすることを保証することができる。例示によってさらに説明すると、本発明の 1 つの態様に基づくコピー管理方法 (250) が図 2A に描かれている。最初に、ホストは、不揮発性メモリデバイスを使用して、ファイルシステムが利用する情報を格納する (252)。次に、コントローラは、ホストが格納した情報を利用して (252)、1 以上のクラスタ (または 1 以上のクラスタ内の 1 以上のセクタ) が何らかの有効なファイルに現在割り当てられているかどうかの判定を行う (254)。当業者であれば解るように、1 以上のクラスタ (または 1 以上のクラスタ内の 1 以上のセクタ) が、何らかの有効なファイルに現在割り当てられているかどうかを判定する (254) ために、ファイルアロケーションテーブル (FAT) にホストが正常に書き込んだ情報をコントローラは直接利用することができる。これとは別に、コントローラは、ホストによってデバイスへ書き込まれた一続きの論理セクタと一緒に、ルートディレクトリ、サブディレクトリおよび FAT 内の書き込まれた情報を利用して、クラスタアロケーションに関するコントローラ自身の情報を生成し、この情報の保守管理を行うことを意図することも可能である。例えば、コントローラは、コントローラ自身のデータアロケーションテーブル (DAT) の生成および / または保守管理を行うことが可能である。いずれの場合にも、コントローラは、現在割り当てられていないと判定された (254) 任意のクラスタ (またはクラスタ内のセクタ) が、メモリ内でのガーベッジコレクション処理中に 1 つの記憶位置から別の記憶位置へコピーされないことを保証する (256)。言い換えれば、クラスタ (またはクラスタ内のセクタ) が現在割り当てられていると判定された場合には (254)、このクラスタ (またはクラスタ内のセクタ) は単にコピーされることになる。コピー管理方法 (250) は後続の処理を終了させる (256)。

30

40

【0027】

50

前述したように、データアロケーションテーブル（D A T）の保守管理を行うステップは、単にファイルアロケーションテーブル（F A T）から得られる情報に依拠する以外のさらに多くの利点を提供することができる。当業者には明らかなように、F A Tは、クラスタが割当て解除されたことを必ずしも示すというわけではない場合がある。例えば、クラスタエントリのチェーンが、ディレクトリ内のいずれのファイル参照符号ともリンクされることなくF A Tの中に存在する場合がある。

【0028】

ファイルが削除されると、ファイルのディレクトリエントリはホスト内のファイルシステムにより修正されて（ファイル名の第1の文字が0 x E 5へ変更される）、ファイルが削除されたことが示される。しかし、F A T内のクラスタエントリは、あたかもクラスタエントリがまだファイルに割り当てられているかのような状態にこのクラスタエントリの連鎖情報を保持することも可能である。クラスタを再使用するための、および、未使用を示す“0 x 0 0 0 0”へF A Tクラスタエントリを更新するためのアルゴリズムとして、典型的には、ホストにおいて特定のファイルシステムを実現する機能がある。D A Tは、ディレクトリ情報またはサブディレクトリ情報の変更を利用して、クラスタ属性が、ホストファイルシステムに依拠することなく、クラスタの現在の割当て状態をファイルに反映することを可能にすることができる。環境によっては、D A Tの実施構成のほうが、単にF A Tに依拠するよりも安全である場合さえある。クラスタの割当て状態を確かめるために単にF A Tを利用する場合、ホストがファイルの終了時にF A Tを更新するまで、メモリシステムが、ファイルへのクラスタの割当てについて認知していない場合もある。したがって、すでに書き込まれているファイルの一部をF A Tに書き込みを行う以前にホストが修正を行う場合、まだ割当てを解除された状態を有するいくつかの有効なデータがコントローラによって破棄される場合もある。D A Tの実施構成はこの問題点を効果的に処理することができる。

【0029】

さらに、本発明の原理に従って提供されるデータアロケーションテーブル（D A T）は、F A Tが通常提供していない特徴も提供することができる。例えば、F A Tはクラスタ用の単一の割当て状態を定義するものである。これに対して、例えば、1つのセクタあるいは1グループのセクタに対する割当て状態の分析をクラスタ内で可能にすることによって、マルチビット状態の保存を許可できるようにD A Tを実装することが可能である。こうすることによって、コントローラは、第1のファイルの最後のクラスタのみが部分的に使用されるファイル間で、割り当てられていないセクタのコピーを避けることが可能となる。D A Tが提供できる利点および追加の特徴を示しながら、以下説明する実施形態では、（D A Tなどの）データ属性格納部（D A S）においてコントローラ自身の情報の保守管理を行うコントローラをさらに例示する。しかし、本発明の範囲と精神から逸脱することなく、コントローラが、単にF A Tに依拠して、ホストが使用する種々のファイルと関連づけられた割当て状態を判断できることは明らかである。

【0030】

図2Bは、本発明の1つの実施形態に従う計算環境200を描く図である。この計算環境200では、フラッシュメモリコントローラ202はホストアプリケーション204の（フラッシュメモリ206などの）不揮発性メモリへのアクセスを可能にする。言い換えれば、ホストアプリケーション204は、フラッシュメモリ206へのインターフェイスとしてフラッシュメモリコントローラ202を利用する。したがって、ホストアプリケーション204は、フラッシュメモリコントローラ202が（書き込み、読み出しなどの）種々のアクセス処理をフラッシュメモリ206上で実行するように要求することができる。ホストアプリケーション204はデータのアドレス指定を行うために論理アドレス空間を利用することができる。フラッシュメモリコントローラは、ホストの論理アドレス空間を対応づけて、コントローラの論理アドレス空間の中へ入れ、ホストの論理アドレス空間はフラッシュメモリ206上の物理アドレス空間に順に対応づけられる。

【0031】

さらに、フラッシュメモリはデータ属性格納部(DAS)208の構築と保守管理とを行うことができる。DAS208には、フラッシュメモリ206に格納されたデータと関連づけられた種々の属性に関する情報が含まれる。例えば、データ属性格納部(DAS)208に格納されたデータ属性はデータの割当て状態、割り当てられたデータのサイズ、優先順位を含むものであってもよい。一般に、データ属性格納部(DAS)208は、任意の望ましいデータ属性を格納でき、また1以上のビットの情報を有する属性を表わすこともできる。フラッシュメモリコントローラ204は、ホストアプリケーション204の活動をモニタするデータ属性モニタ部208を設けるものであることに留意されたい。データ属性モニタ部208は、フラッシュメモリ206に格納されたデータに関してホストアプリケーション204の活動を解釈することができる。このことは、データ属性モニタ部208がどこでデータ属性を変更すべきかを判断し、DAS208の中でデータ属性を適宜更新できることを意味する。

10

【0032】

例示によってさらに説明すると、図3Aは本発明の1つの実施形態に従ってデータ属性格納部(DAS)208を描く図である。この実施形態では、データ属性208はデータアロケーションテーブル(DAT)308として実現される。したがって、DAT308は不揮発性メモリシステムの種々のデータ部分の割当て状態/割当て解除状態に関する情報を提供する。理解されるように、これら種々のデータ部分は論理データ構成、すなわち、ホストアプリケーション204によって利用される論理データ構成に従って構成することができる。前述したように、ホストアプリケーションはデータを構成して、クラスタと呼ぶことができるホスト論理単位に変えることも可能である。DAT308はこれらのクラスタを表わす単位に分割することができる。

20

【0033】

このことは、ホストアプリケーションが利用する論理クラスタの各々をDAT308の中でエントリとして表わすことができることを意味する。例えば、クラスタ1は、第1のエントリによって、クラスタ10は第10のエントリによって表わすことができる等々となる。さらに、DAT308の個々のエントリはクラスタの割当て状態/割当て解除状態を示すことができる。例えば、1ビットを用いて論理クラスタが現在割り当てられているか、あるいは割当てが解除されているかをDAT308の個々のエントリが示すようにすることができる。数ビットを用いて、属性に関するさらに多くの情報および/またはいくつかの属性に関する情報を示すことができることに留意されたい。いずれの場合にも、DAT308によってデータのより良好な管理ができることになる。例えば、データ部分の割当て状態/割当て解除状態に基づいて、フラッシュメモリコントローラは、ホストアプリケーションにより割当てが解除されたデータのコピーを避けるようにすることも可能である。

30

【0034】

さらに例示によって説明すると、図3Bは、本発明の1つの実施形態に従うDAT208のエントリ302を描く図である。テーブルエントリ302は4ビットを含み、その場合、第1のビット(ビット0)は割当て状態/割当て解除状態を示すことができ、3つの他のビットはまとめて、どのくらいの量のデータが割り当てられているか(半分を示す“100”および満杯を示す“111”など)を示すことができる。同様に、さらに多くのビットを利用してクラスタの別の属性に関する情報を示すことができる。例えば、4ビットによって個々のクラスタを表わすことが可能であり、その場合1ビットは割当て解除状態を表わし、3ビットは、クラスタのどのような部分が割当てを解除されているかを示すことになる。当業者には明らかなように、多数の他の方法でクラスタ用の多ビットを定義することも可能である。例えば、極性(または順序)を尊重してもよいし、あるいは(6、8、12などの)より多くのビットを使用してもよい。前述したのとは別に、“サブクラスタ”のセクタ番号を示すために、完全に割当てを解除されたすべてのビットを割り当てることが可能であり、完全に割り当てられたクラスタを例えばマークしてすべてのビットをゼロにセットすることも可能である。

40

50

【0035】

前述したように、データ属性モニタ部210はホストアプリケーション204の活動をモニタするものである。このモニタに基づいて、データ属性モニタ部259はデータ属性格納部(DAS)208を更新することができる。図2に示されているように、ホストアプリケーション204の活動を解釈するために、データ属性モニタ部210は、ホストが理解し、使用するデータの論理構成を実現する必要がある。例示によってさらに説明すると、図3Cは、ホストの論理データ構成を描く図であり、この論理データ構成は本発明の1つの実施形態に従うコントローラによって利用することができる。図3Cに示されているように、ホストは、論理アドレス空間を分割して、システム310とユーザ空間312にすることができる。システムデータ310は、ファイルアロケーションテーブル(FAT1)と、FAT2(FAT1のコピー)と、ルートディレクトリ情報とを含むことが可能であり、これらすべてをシステムデータ310として格納してもよい。ユーザ空間はサブディレクトリに関する情報を含むことも可能である。データ属性モニタ部210は、ホストアプリケーション202が割当てを解除した(例えば、ファイルを除去したり、ファイルを短くしたりした)かどうかを判定するために、FAT1、FAT2、ルートディレクトリおよびサブディレクトリをモニタすることも可能である。データのモニタについても図6を参照しながら以下で説明する。

10

【0036】

しかし、次に図4を参照しながら、フラッシュメモリに格納されたデータに関する情報をフラッシュメモリコントローラに提供する方法400が本発明の1つの実施形態に従って描かれている。データに関する情報は、フラッシュメモリコントローラによって利用され、例えば、(ガーベッジコレクションなどの)データの保守管理中にデータについての決定を通知することが可能となる。方法400は、例えば、図2のフラッシュメモリコントローラ202によって利用することができる。

20

【0037】

最初に、フラッシュメモリ記憶装置のコントローラ用のデータ属性格納部(DAS)が生成される(402)。データ属性格納部(DAS)は、フラッシュメモリ記憶装置内の複数のデータ部分の各々に対して1以上の属性を提供する。データにアクセスするためにコントローラを使用するホストの活動がモニタされる(404)。次に、モニタ(404)に基づいて、フラッシュメモリ記憶装置のデータ部分に関連する属性を更新すべきかどうか判定される(406)。したがって、フラッシュメモリ記憶装置の少なくとも1つのデータ部分に対する更新を行うべきであると判定された(406)場合、少なくとも1つのデータ部分に関連する少なくとも1つの属性が更新される(408)。更新(408)後、ホストの活動はモニタされる(404)。その後、方法(400)は、データのモニタがそれ以上所望されなくなったり、必要とされなくなったりするまで、前述したのと同様に進行することができる。

30

【0038】

前述したように、データ属性格納部(DAS)を設け、フラッシュメモリコントローラによってこのデータ属性格納部を利用して、(ガーベッジコレクションなどの)保守管理処理を実施することができる。図5は、本発明の1つの実施形態に従うフラッシュメモリの保守管理方法(500)を描く図である。最初に、フラッシュメモリ記憶装置に格納されたデータの一部に対してフラッシュメモリコントローラが保守管理処理を実施する必要があるかどうかの判定が行われる(502)。コントローラが保守管理処理を実施する必要がある旨が判定された(502)場合、データ属性格納部(DAS)のどのような部分が、保守管理の対象となるデータ部分に関する情報を提供するかが判定される(504)。したがって、データ部分に関連する少なくとも1つの属性がデータ属性格納部(DAS)から読み出される(506)。最後に、少なくとも1つのデータ属性に基づいて処理が実行される(508)。少なくとも1つのデータ属性により提供された情報に基づいて、保守管理処理をさらにインテリジェントに実行することができると理解すべきである。

40

【0039】

50

図6は、本発明の1つの実施形態に従うモニタ方法600を描く図である。モニタ方法(600)は、図4に例示されているモニタ処理と更新処理(すなわち、図4の処理404、406、408)中に行うことができるいくつかの処理例を描く図である。モニタ方法600はフラッシュメモリコントローラによって実行することができる。最初に、書き込み処理を求める要求が受信されたかどうか判定される(602)。次に、書き込み処理が指定アドレス空間のアドレス指定を行うかどうか判定される。例えば、この指定アドレス空間には、DOS環境の中で動作するホストがアドレス指定を行う論理アドレス空間のFAT1、FAT2、ルートディレクトリおよびサブディレクトリを含むことができる。ルートディレクトリ内のおよび別のサブディレクトリ内の情報からサブディレクトリ用の指定アドレス空間を決定することも可能である。一般に、指定アドレス空間とは、データが(例えば、ファイルのディレクトリの除去や修正、ファイルの編集などの)割当てを解除された処理であることをホスト活動が示すことができるアドレス空間を表わすものである。いずれの場合にも、書き込み処理が指定アドレス空間に対するものであると処理604時に判定された場合、指定アドレス空間に現在格納されているデータが読み出される(606)。次に、書き込み処理が実行される(608)。書き込み処理が実行された(608)後、以前に読み出された(606)古いデータは、書き込まれた(608)データと比較される(610)。したがって、データの変更が存在するかどうか判定される(612)。データの変更が存在しないと判定された場合(612)、書き込み処理要求が受信されたかどうか判定される(602)。

【0040】

前述したのとは別に、更新済みデータが古いデータから別の記憶位置に書き込まれるフラッシュメモリシステムにおいて、前回の読み出しを行う(606)ことなく、古い記憶位置と新しい記憶位置とに位置するデータ間で直接の比較(610)を行うことが可能となる。

【0041】

しかし、データに対する変更が検知されたと判定された(612)場合、この変更は解釈されて(614)、1以上のクラスタが割当てを解除されたかどうかの判定が行われる(616)ことになる。クラスタが割当てを解除されていないと判定された場合、書き込み処理要求が受信されたかどうか判定される(602)。しかし、1以上のクラスタが割当てを解除されたと判定された(616)場合には、データアロケーションテーブル(DAT)内での1以上のクラスタの状態が変更されて、“割当て解除済み”状態を示すようになることに留意されたい。したがって、指定アドレス空間内のデータの変更がデータの割当て解除を結果として生じたと知覚される(616)と、モニタ方法600によってデータアロケーションテーブル(DAT)の効果的な更新が行われる(618)。

【0042】

一方、書き込み処理が指定アドレス空間に対するものではないと判定された(604)場合、書き込み処理が実行される(620)。さらに、書き込まれたクラスタが特定され(622)、その結果、1以上の特定したクラスタの状態を“割当て済み”にセットできる。理解されるように、この処理は、無指定のアドレス空間において書き込み処理を実行したとき、特定されたクラスタを表わすDATのエントリが“割当て済み”にセットされることを保証するものである。DAT内で表わされるクラスタの状態を“割当て済み”にセットした(624)後、書き込み処理要求が受信されたかどうかの判定を行う(602)ことができる。

【0043】

前述したように、データ属性格納部(DAS)を例えば利用して、保守管理処理を実施することができる。例示によってさらに説明すると、図7は、本発明の1つの実施形態に従うガーベッジコレクション方法700の例を描く図である。理解されるように、ガーベッジコレクション方法700は、データアロケーションテーブル(DAT)を利用して、更新処理中にブロックのより良好な更新を可能にするものである。最初に、更新ブロックを閉じるべきかどうか初めに判定される(702)と、更新ブロックのすべてのセクタ

10

20

30

40

50

が書き込まれたかどうかの判定が行われる。更新ブロックのすべての論理セクタが書き込まれた場合、原ブロックが更新ブロックにより効果的に置き換えられることになる。したがって、更新ブロックが原ブロックとしてマークされ(706)、前の原ブロックは消去され(708)、その結果、この原ブロックは後で使用できることになる。

【0044】

一方、更新ブロックのすべてのセクタが書き込まれなかったと判定された(702)場合、書き込まれなかったセクタに対応する論理セクタが特定される(710)。次いで、特定された論理セクタ(710)がクラスタへ対応づけられる(712)。クラスタはデータアロケーションテーブル(DAT)の中でチェックされる(714)。したがって、特定されたクラスタのすべて(710)が割当てを解除されたことをDATが示すかがどうかが判定される(716)。すべてのクラスタの割当て解除が行われたことがDATによって示された場合、更新ブロックは原ブロックとしてマークされ(706)、古いブロックは消去される(708)。

10

【0045】

すべてのクラスタが割当てを解除されたことがDATによって示される(716)と、1以上のクラスタがコピーされなくなることに留意すべきである。1以上のクラスタがまだ割り当てられていることがDATテーブルによって示された場合にのみ、割り当てられた状態のままになっているデータは更新ブロックへコピーされる(718)。いずれの場合にも、更新ブロックをすぐに閉じたり、更新ブロックがいっぱいになったとき、後で閉じたりすることができることに留意すべきである。更新ブロックにマークをつけ(706)、前の原ブロック(古いブロック)を消去すること(708)はできるが、割り当てられていないデータの不必要なコピーは避けられることになる。ガーベッジコレクション方法(700)は消去(708)の後終了する。

20

【0046】

理解されるように、本発明の1つの実施形態に従って、フラッシュメモリを用いてデータ属性格納部(DAS)を実現することができる。例示によってさらに説明すると、図8は本発明の1つの実施形態に従うデータアロケーションテーブル(DAT)800を描く図である。メモリ部分801を利用してデータアロケーションテーブル(DAT)800が実現される。メモリ部分801は単位として消去可能なブロックであってもよい。メモリ部分801はデータアロケーションテーブル(DAT)800を表わすのに必要とされるメモリよりも大きくなる。個々のセクションは1以上のページに配置されるものであってもよく、その場合ページはプログラミングメモリの最小単位となる。

30

【0047】

図8に示されているように、データ属性格納部(DAS)800は、複数のセクション、すなわち、当初メモリ部分801の冒頭に書き込まれる原セクションDAS1(802)、DAS2(804)、DAS3(806)、DAS4(808)に分割してもよい。これら原セクションの各々は、論理的にデータを構成するためにホストが使用する複数のクラスタを表わすことができる。クラスタと関連づけられたデータ属性の更新を行う必要があるとき、データ属性格納部(DAS)800の適当な原セクション用の更新セクションに書き込みを行うことによって、データ属性格納部(DAS)800の対応するセクションを更新することができる。この更新セクションは最後の原セクションすなわちDAS4(808)の後に書き込まれる。例えば、原DAS3(806)は新たなセクション816に書き込みを行うことによって更新される。同様に、原DAS1(802)を更新する必要があるとき、新たなセクション812に書き込みを行うことができる。原DAS4(818)を表わすなどのために新たなセクション818の書き込みが行われる。

40

【0048】

さらに、インデックス方式を用いてデータアロケーションテーブル(DAT)800の保守管理を行うことができる。図8に示されているように、メモリの個々のセクションにはインデックス部分(Ii-U)が含まれる。最後のエントリ(806)のこのインデックス部分はデータ属性格納部(DAS)800の最新のセクションを参照する。このこと

50

は、インデックス820、830、840、850がそれぞれデータ属性格納部(DAS)800の第1、第2、第3および第4の最新のセクション(DAS1、DAS2、DAS3、DAS4)を指すことを意味する。例えば、メモリ部分801がほぼいっぱいになったとき、原セクション(DAS1、DAS2、DAS3、DAS4)として最新のセクションが再生されるように、メモリ部分801のガーベッジコレクションを行うことができる。

【0049】

データ属性格納部(DAS)の利用が任意の利用となるように本発明を実施することができることも留意すべきである。例えば、データ属性格納部(DAS)の利用は、ホストの実際のデータ論理構成が(DOSなどの)特定の動作環境用として予想されるデータ構成に従うことを保証するように、検証プロセスが決定するシステムコンプライアンスに依存するようにすることも可能である。前述したのとは別に、ホストによるコマンドの送信によってデータ属性格納部(DAS)の利用を可能にして、データ属性格納部(DAS)が特定の動作環境用のデータ構成に従うものであることを示すようにすることができる。割り当てられたクラスタの割り当てが解除されるとすぐに、データアロケーションテーブル(DAT)がクラスタ(またはクラスタ内のセクタ)の状態を変更する必要はないことに留意すべきである。言い換えれば、クラスタが割り当てを解除されたとき、クラスタがまだ割り当てられていると信じることは、深刻な問題をもたらすものではない。したがって、割り当て状態を割り当て解除にセットすることが可能となる。理解されるように、異なる時刻に1つずつ行うのではなく、一度にクラスタグループに対する割り当てが解除となるように割り当て状態をセットすることによって、システム性能をさらに改善することが可能となる。一方、当業者であれば理解できるように、有効な、割り当て済みのクラスタ用の状態情報は常に厳密に正確で、かつ、最新のものであることが望ましい(すなわち、割り当て済みのクラスタは割り当てられたものとして常にマークを施すことが望ましい)。割り当て済みのクラスタに正確にマークを施すことを保証するために、ホストがクラスタへデータを書き込む度に、割り当て状態クラスタが割り当て済みに自動的にセットされるようにしてもよい。データアロケーションテーブル(DAT)内の情報をマッピングテーブル内の情報と同時に更新して、論理データブロックへの物理メモリブロックの割り当てを定義し、最近書き込まれたセクタあるいはクラスタへ保存するようにしてもよいことも理解できる。

【0050】

さらに、データ属性格納部(DAS)を他の多くの保守管理処理用として利用できることが理解できる。例えば、データ属性格納部(DAS)は、ブロックに対応するすべてのクラスタが割り当てを解除されたことを示すと、ブロックを消去することができる。この消去は、例えばバックグラウンドとして実行され、性能のさらなる改善が図られることになる。

【0051】

別の実施形態では、ホストによって不揮発性メモリに格納された情報を利用して、どのようにホストから受信したデータを格納すべきであるかを決定することができる。1つの例では、ホストがFATに格納した情報とディレクトリとを利用して、シーケンシャルにあるいは非シーケンシャルに(カオス的に)データを格納すべきかどうかの判定を行うことができる。

【0052】

或る種のメモリシステムでは、データがシーケンシャルであるか非シーケンシャルであるかに応じて、ホストから受信したデータを格納するのに異なる格納方式が利用される。例えば、2003年12月30日出願の“不揮発性メモリおよびブロック管理システムに関する方法”という米国特許出願第10/750,155号(特許文献9)は、その全体が本願明細書において参照により援用されているものであるが、この特許出願には、ホストによって非シーケンシャルに送信されるデータを処理する或る技法について記載されている。特に、シーケンシャルな更新をシーケンシャルな更新ブロックに割り当て、その一方で、データを任意の順序に格納できるカオス更新ブロックに非シーケンシャルな更新を

10

20

30

40

50

割り当てることができる。カオス更新ブロックへのデータの格納によって、或る種類のデータ更新に利点が生じる一方で、別の種類のデータ更新にとっては非効率を引き起こす原因になる場合もある。この理由のために、米国特許出願第10/750,155号(特許文献9)には、所定の基準が満たされたシーケンシャルな方法で、或る非シーケンシャルに受信したデータを格納するための技法について記載されている。本発明の実施形態では、非シーケンシャルにデータを受信したとき、不揮発性メモリへデータをシーケンシャルに書き込むか、非シーケンシャルに書き込むかの判定基準は、不揮発性メモリにホストが格納した情報の分析から決定できる。2003年12月30日出願の“大きな消去ブロックを含む不揮発性メモリシステムの管理”という米国特許出願第10/749,831号(特許文献10)には、不揮発性メモリアレイに格納するためにホストが送信するデータの管理のための追加の方法について記載されている。この特許出願は、その全体が本願明細書において参照により援用されている。特に、ホストが非シーケンシャルに更新するデータの管理方法について説明する。所定の判断基準に従うデータに対して様々な指定ブロックを使用することも可能である。

10

【0053】

本発明の実施形態では、FAT、ルートディレクトリおよびサブディレクトリ(または不揮発性メモリに格納されている同様のホストデータ)と、ホストにより書き込まれている(セクタなどの)データ部分の論理アドレスとを比較したり、DASに格納された情報のような引き出された情報と比較したりして、書き込み中のデータ部分が次の未使用クラスタ内に論理アドレスを含んでいるかどうかの判定を行うようにしてもよい。データ部分が次の未使用クラスタ内にアドレスを含んでいれば、それは、ホストが次の利用可能な記憶位置に新しいデータを格納していることを示すことになる。このような格納パターンでは、シーケンシャルにデータを格納し、次いで、非シーケンシャルな順序でデータを格納するのではなく、不揮発性メモリにすでに格納されているデータをコピーし、それから、シーケンシャルに格納されている記憶位置へ後でデータをコピーすることによって、更新ブロック内のいずれかのギャップを満たすことができると好都合な場合がある。

20

【0054】

図9Aは、FATテーブル900の1例を示す。個々のクラスタ0~8は、ファイル内の次のクラスタを示すFATテーブル内にエントリを有する。したがって、クラスタ0用のエントリは、このファイル用の次のクラスタがクラスタ1の中にあることを示す“1”になる。クラスタ1内のエントリは“5”であり、次のクラスタがクラスタ5であることを示している。クラスタ5内のエントリは“6”であり、次のクラスタがクラスタ6であることを示している。クラスタ6内のエントリは、“ファイルの最後”すなわちクラスタがそれ以上このファイル用のデータを含まないことを示す“EOF”である。別のファイルがクラスタ3と4とに同様に格納される。クラスタ4、7および8は、“未使用”すなわち割当てを行わないように指定する。これは、例えば、ファイルが古いものになっているホストによる割当て解除の結果である場合もある。ホストによるこの割当て解除は、不揮発性メモリの中でデータが消去されることを必ずしも意味するものではないことに留意すべきである。というのは、ファイルの格納に用いる論理アドレス範囲が新しいデータの格納用として単に利用できるようになるにすぎないからである。古いデータは、将来の或る時点まで不揮発性メモリに物理的に格納された状態のまま残る場合もある。古い制御情報を含むクラスタが割当てを解除される可能性もある。やはりホストによって不揮発性メモリの中に格納されたルートディレクトリやサブディレクトリがファイルの第1のクラスタを示す場合もある。

30

40

【0055】

図9Bは、ルートディレクトリとサブディレクトリとに含まれる情報と共にFAT900から導き出されるDAT910を示す。DAT910には、クラスタが割り当てられているか、未使用であるかを示す個々のクラスタ用のエントリが含まれる。追加データは、前述したようにDAT910に格納することも可能である。DATの保守管理を行って、DATに格納された情報が、FATおよびディレクトリ内の情報よりも最新の情報となる

50

ようにすることもできる。ホストが格納用の新しいデータを送信するとすぐに、更新されたFATまたはディレクトリ情報がホストによって書き込まれないうちにDATを更新することも可能である。同様に、DAS内の他の情報の保守管理を行って、最新のものにするように図することも可能である。

【0056】

図9Cは、FAT900の論理アドレス範囲の一部用のセクタとクラスタとの間の対応関係を示す。この例では、1つのクラスタの中に4つのデータセクタが含まれている。異なるセクタ番号を有するクラスタを使用することも可能である。典型的には、クラスタには4~64個の間のセクタが含まれる。クラスタ4にはセクタX、セクタX+1、セクタX+2およびセクタX+3が含まれている。ホストによる論理アドレス範囲の割当てが一般にクラスタ毎のベースで行われるのに対して、メモリコントローラは一般にセクタ内のデータを処理するものである。一般に、クラスタは単一の単位として割り当てられるが、これは、たとえクラスタの論理アドレス範囲全体がデータの格納用として使用されなかった場合でも、後で行われる別のデータの格納用としてクラスタの論理アドレス範囲全体を使用できないようにするためである。図9Cは、割当て済みのクラスタから得られるセクタ5と6とを示し、ホストがデータ格納用としてこれらの論理アドレスを利用できないことを示すために、これらのセクタにはシェードが施されている。

【0057】

図9Dと図9Eは、非シーケンシャルにホストから受信したデータの格納を行うための2つの代替方式を示す。双方の図には、不揮発性メモリに格納するためにホストから受信されるセクタX~X+3並びにX+12~X+15が示されている。これらのセクタは図9Cのクラスタ4と7に対応するセクタである。図9Dでは、受信したデータがカオス更新ブロック920へ書き込まれている。このブロックはカオスブロックである。というのは、セクタX+12がセクタX+3の直後に書き込まれていて、セクタX+3からセクタX+12までの論理アドレス範囲でジャンプが生じているからである。カオスブロック920は、セクタX~X+3を含むとき、シーケンシャルな更新ブロックになることを開始してもよく、さらに、ブロック920は、セクタX+12が格納されているとき、カオス更新ブロックへの変換を開始してもよい。一般に、ホストが非シーケンシャルにデータを送信する場合、カオス(非シーケンシャルな)ブロックが利用される。図9Dのカオスブロック920内の有効なデータはシーケンシャルなブロック922へ最終的に移転され、ブロック922において、別の記憶位置からコピーできるセクタX+4~X+11と共に格納される。これは一般に統合処理の一部として行われる。シーケンシャルなブロックは一般に長期の格納を行うためのものであることが望ましい。というのは、セクタが所定の方法で配列されるため、ブロック内でのセクタのインデックス作成が必要でない場合があるからである。カオスブロック920内の全ての有効なデータが別の記憶位置へコピーされた後、カオスブロック920は古いものとしてマークされ、消去が可能となる。

【0058】

図9Eは、セクタX~X+3並びにX+12~X+15の非シーケンシャルなデータを示し、このデータはシーケンシャルなブロック930の中でシーケンシャルに更新される。ブロック930がこの時点でシーケンシャルなブロックとなるようにセクタX~X+3にまず書き込みを行うことができる。セクタX+12が受信されると、たとえデータが非シーケンシャルに受信されたものであっても、このデータをシーケンシャルに格納するために判定が行われる。原ブロックから(クラスタ5および6に対応する)セクタX+4~X+11をコピーして、ホストから受信したセクタ間の論理アドレス範囲を満たすことによってシーケンシャルな格納を行うことが可能となる。セクタX+4~X+11はパディングセクタと見なすことができる。というのは、これらのセクタを利用して、ホストが送信するセクタ内のギャップの充填すなわち“パッドアウト(pad out)”が行われるからである。セクタX+4~X+11は別の記憶位置からコピーされた有効なデータセクタである。このようにして、受信されたセクタは、カオスブロック内の中間の格納段階を通ることなくシーケンシャルなブロック930に格納される。したがって、これら2つの技法間

10

20

30

40

50

の主要な相違点として、セクタ $X \sim X + 3$ 並びに $X + 12 \sim X + 15$ をカオスブロック 920へ書き込み、次いでカオスブロック 920を後で消去するステップが図 9 Eの例では不要になるという点が挙げられる。セクタ $X \sim X + 3$ 並びに $X + 12 \sim X + 15$ は一度書き込まれ、セクタ $X + 4 \sim X + 12$ はこの例ではそのままコピーされる。この処理は、8回の少ない回数でセクタ書き込みと、1回の少ない消去処理とを必要とし、その結果データ格納効率が改善されることになる。

【0059】

受信したデータをシーケンシャルに格納すべきか、非シーケンシャルに格納すべきかに関する判定は種々の基準に基づいて行うことができる。或る種のデータ、特に、すぐに再更新が行われる可能性の大きいデータの場合、非シーケンシャルな格納のほうが適している場合がある。例えば、制御データや、更新頻度の高いホストデータの一部の場合、カオスブロックに格納するほうが効率が良くなる場合がある。或る種のデータ、特に、データがすぐに再更新される可能性が小さい場合、シーケンシャルな格納のほうが適している場合がある。非シーケンシャルに受信したデータは一般に非シーケンシャルに格納されると考えられるが、場合によっては、シーケンシャルにデータを書き込むと都合がよいこともある。カオスブロックへの中間格納が回避されるため、カオスブロックのガーベッジコレクションを行ってシーケンシャルなフォーマットでデータを取得する必要はない。これによって、図 9 Dに示されている統合処理が回避されることになり、その結果、オーバーヘッドが減り、性能の改善が図られることになる。ホストが次の未使用クラスタへ書き込みを行うとき、これはホストが次の利用可能な記憶位置に単に新しいデータを書き込んでいるという表示であり、一般に、論理アドレス範囲がすぐに再更新されるという表示ではない。したがって、図 9 Eに示されているようなシーケンシャルな更新方式を選択するほうが、図 9 Dに示されているようなカオス的な更新方式を選択するよりも効率のよいものとなる可能性がある。この結果、データが次の未使用クラスタから得られたものであるか否かに従って、使用する格納方式を決定することができる。データが、図 9 Eに示されているような次の未使用クラスタからのものである場合、シーケンシャルな格納部を選択することができる。したがって、図 9 Dの格納方式はこの例ではこのデータ用としては選択されないことになる。

【0060】

図 10は、本発明の実施形態に従う格納方式の選択を示すフローチャートである。非シーケンシャルなデータがホストから受信されると、データが次の未使用クラスタ 10からのものであるか否かに関する決定がコントローラによって行われる。この決定は、FATと、ホストが不揮発性メモリに格納したディレクトリ情報とを直接見ることによって、あるいは、DASやDATに格納されているような情報のようなFATとディレクトリ情報とから導き出される情報を見ることによって行ってもよい。現在アドレス指定されているクラスタが次の未使用クラスタでなければ、データはデフォルトの方式 12に従って通常の方法で格納される。このことは、データが非シーケンシャルに格納されることを意味する場合がある。あるいは、いくつかの例では、データをシーケンシャルに書き込ませる別の基準が存在する場合もある。現在アドレス指定されているクラスタが次の未使用クラスタであれば、シーケンシャルな書き込みが選択される。この場合、データの有効なクラスタが原メタブロックから現在のシーケンシャルなメタブロック 14へコピーされて、最後に書き込まれたクラスタと現在アドレス指定されているクラスタとの間の論理アドレスのギャップのパディングが行われることになる。次いで、現在アドレス指定されているクラスタから受信したデータは現在のシーケンシャルなメタブロックへ書き込まれる。この結果、現在のメタブロック 16はシーケンシャルな状態のまま残ることになる。

【0061】

図 11は、1つの実施形態でホストから受信したセクタのための格納方式の選択を示すフローチャートである。同様の格納方式選択フローチャートが米国特許出願第 10 / 750, 155号(特許文献 9)に示されている。開かれた更新ブロックが存在する(20)場合、更新ブロックはシーケンシャルなブロックになり(22)、セクタは更新ブロック

10

20

30

40

50

の最後のセクタまでシーケンシャルになり(24)、次いで、セクタは更新ブロックへ書き込まれる(26)。セクタが更新ブロックの最後のセクタまでシーケンシャルでなければ、シーケンシャルな強制書き込みを行うべきか否かに関する決定が行われる(27)。更新ブロック内の最後のセクタから受信したセクタへのアドレスジャンプが所定量(C_B)よりも大きくなければ(28)、最後のセクタと受信したセクタとの間のギャップはコピーされたセクタによりパディングを行って(30)、前述したようにシーケンシャルなブロックとして更新ブロックの保守管理を行うようにしてもよい。更新ブロック内の空の物理セクタ数が所定量(C_B)よりも大きくなければ(32)、更新ブロックを閉じて(34)、シーケンシャルなブロックとして更新ブロックの保守管理を行うことも可能であり、さらに、前述したように新たな更新ブロックを割り当てること(36)も可能である。シーケンシャルな強制書き込みを引き起こす第3の場合として、ホストが次の利用可能なクラスタへの書き込みを行っている(40)場合がある。このような書き込みが行われているとき、シーケンシャルなブロックとして更新ブロックの保守管理を行うことができるように、不揮発性メモリ内のどこか別の場所からコピーされたセクタによって、受信したセクタと更新ブロック内の最後のセクタとの間のギャップのパディングを行う(30)。シーケンシャルな形でデータの書き込みを行うためのこれら基準のいずれも満たされなければ、更新ブロックをカオス更新ブロックへ変換すること(42)も可能である。したがって、この結果、格納方式を選択する選択基準は、ホストが不揮発性メモリに格納したデータから導き出された属性を含むことが可能となる。

10

【0062】

20

図12Aは、更新ブロック50においてシーケンシャルにデータを格納するメモリシステムによってホストから非シーケンシャルに受信したデータの1例を示す。クラスタ2のセクタAが受信され、後続してクラスタ7のセクタBが受信される。セクタAは、受信されると、更新ブロック50の第1のセクタとなることが可能であり、更新ブロック50がセクタAのみを含む場合、更新ブロック50はシーケンシャルなブロックと見なされる。これとは別に、図12Aに示されているように、更新ブロック50はシーケンシャルなブロックとすることも可能である。というのは、更新ブロック50に前回格納されたセクタと共にセクタAがシーケンシャルに格納されるからである。セクタBの受信に先行して更新ブロック50がシーケンシャルであると想定して、セクタBを受信したとき、シーケンシャルな更新ブロックとして更新ブロック50の保守管理を行うか、更新ブロック50(セクタA)へ書き込まれた最後のセクタの後にセクタBを直接格納し、それによってカオスブロックへ更新ブロック50を変換するかに関する判定が行われる。この例では、クラスタ7はセクタAに後続する次の未使用クラスタである。したがって、クラスタ7からセクタBを受信したとき、シーケンシャルな強制書き込みが行われ、セクタが更新ブロック50へコピーされ、セクタAとセクタBとの間のギャップが満たされることになる。この結果、データは、図12Aの更新ブロック50にシーケンシャルに格納されることになる。

30

【0063】

図12Bは、更新ブロック52に非シーケンシャルにデータを格納するメモリシステムによってホストから非シーケンシャルに受信したデータの1例を示す。セクタAは図12Aのようにまず受信される。セクタAはシーケンシャルに格納され、その結果、更新ブロック52は、セクタAがプログラムされるとシーケンシャルなブロックになる。次いで、セクタXが受信される。セクタXは次の未使用クラスタへは書き込まれない。セクタXは割当て済みのクラスタへ書き込まれ、したがってシーケンシャルな強制書き込みを引き起こすものではない。セクタXは非シーケンシャルに書き込みを行ってもよいし、あるいは、そのようにする何らかの別の理由(例えば、セクタXとセクタSとの間の論理的ギャップがしきい値未満であるなど)があれば、シーケンシャルに書き込みを行ってもよい。この例では、セクタXは、更新ブロック52内の次の利用可能な空間へ書き込みが行われているものとして示されている。この結果、更新ブロック52はカオスブロックとなる。図12C内の第2の例では、セクタYはセクタAの後で受信される。セクタYの受信前には

40

50

更新ブロック54はシーケンシャルなブロックになる。セクタYは次の未使用クラスタから得られたものではない。というのは、更新ブロック(セクタA)内の最後に書き込まれたセクタとセクタYとの間に未使用クラスタ(クラスタ7)が存在するからである。したがって、セクタYはシーケンシャルな強制書き込みを引き起こすものではなく、パディングを行うことなく更新ブロック54へ書き込みを行って、更新ブロック54をカオスブロックにするものである。図12Dに示されている第3の例では、セクタAの後にセクタZが受信される。更新ブロック56はセクタZの受信に先行してシーケンシャルである。セクタZはクラスタ0から得られるものであり、したがって、次の未使用クラスタから得られるものではない。というのは、クラスタ0はクラスタ2の前に存在するからである。したがって、図示の順序で示されているように、クラスタ0は次の未使用クラスタではない。したがって、セクタZはクラスタAの直後に書き込みを行うことが可能であり、これに起因して更新ブロック56はカオスブロックとなる。

10

【0064】

本発明の利点は非常に多い。異なる実施形態または実施構成は1以上の以下の利点を生み出すことができる。以下が網羅的なリストではないこと、および本願明細書に記載されていない他の利点が存在する可能性もあることに留意されたい。本発明の1つの利点として、不揮発性メモリシステムの性能の改善が可能であるという点が挙げられる。本発明の別の利点として、広く使用されている規定を用いて本発明の実現が可能であるという点が挙げられる。本発明のさらに別の利点として、本発明はフラッシュメモリ内のテーブルとして相対的に小さな属性を実現することができるという点が挙げられる。他の実施構成ではテーブルは必要ではない。さらに別の利点として、任意の用途のために、および/または、バックグラウンドでの保守管理処理中に利用するために、本発明が実施可能であるという点が挙げられる。

20

【0065】

前述した本発明の種々の態様または特徴は、単独で、あるいは、種々に組み合わせて利用することが可能である。本発明はハードウェアあるいはハードウェアとソフトウェアとの組み合わせによって実現可能である。本発明はコンピュータ可読媒体におけるコンピュータ可読コードとして具現化することも可能である。このコンピュータ可読媒体は、コンピュータシステムによって後で読み出すことができるデータを格納することが可能な任意のデータ記憶装置であってもよい。コンピュータ可読媒体の例として、リードオンリメモリ、ランダムアクセスメモリ、CD-ROM、DVD、磁気テープ、光データ記憶装置、並びに搬送波が含まれる。これらのコンピュータ可読媒体は、ネットワーク接続されたコンピュータシステムにわたって分散するので、コンピュータ可読コードを格納し、分散して実行するように図ることが可能である。

30

【0066】

本発明の多くの特徴と利点とは前述した説明から明らかであり、したがって、本発明のこのようなすべての特徴並びに利点をカバーすることが添付の特許請求の範囲によって意図されている。さらに、多くの修正および変更が当業者の心に容易に浮かぶであろうことが考えられるため、本願明細書に例示し、記載したような構成および処理と同じ正確な構成および処理に本発明は限定されるものではない。したがって、すべての適切な修正並びに均等物は本発明の範囲内に属するものとなる。

40

【図面の簡単な説明】**【0067】**

【図1】いくつかの論理セクタに分割することができる論理アドレス空間を示す。

【図2A】本発明の1つの実施形態に従うコピー管理方法を描く。

【図2B】本発明の1つの実施形態に従う計算環境を描く。

【図3A】本発明の1つの実施形態に従うデータアロケーションテーブル(DAT)を描く。

【図3B】本発明の1つの実施形態に従うデータアロケーションテーブル(DAT)のエントリを描く。

50

【図3C】本発明の1つの実施形態に従うコントローラによって利用できるホストの論理データ構成を描く。

【図4】本発明の1つの実施形態に従う、フラッシュメモリに格納されたデータに関する情報をフラッシュメモリコントローラに提供する方法を描く。

【図5】本発明の1つの実施形態に従うフラッシュメモリの保守管理方法を示す。

【図6】本発明の1つの実施形態に従うモニタ方法を示す。

【図7】本発明の1つの実施形態に従う例示のガーベッジコレクション方法を描く。

【図8】本発明の1つの実施形態に従うデータアロケーションテーブル(DAT)を描く。

。

【図9A】ファイルアロケーションテーブル(FAT)の1例を示す。

10

【図9B】図9AのFATから導き出されるDATを示す。

【図9C】ホストデータの一部に対するクラスタとセクタ間の対応関係の1例を示す。

【図9D】カオスブロックに格納され、シーケンシャルなブロックへ遅れてコピーされる非シーケンシャルなデータを示す。

【図9E】シーケンシャルなブロックに直接格納された非シーケンシャルなデータを示す。

。

【図10】本発明の実施形態に従うホストデータの一部のための格納方式を選択するためのフローチャートを示す。

【図11】本発明の別の実施形態に従うホストデータの一部のための格納方式を選択するためのフローチャートを示す。

20

【図12A】シーケンシャルに格納された非シーケンシャルなデータの1例を示す。

【図12B】非シーケンシャルに格納された非シーケンシャルなデータの第1の例を示す。

。

【図12C】非シーケンシャルに格納された非シーケンシャルなデータの第2の例を示す。

。

【図12D】非シーケンシャルに格納された非シーケンシャルなデータの第3の例を示す。

。

【図1】

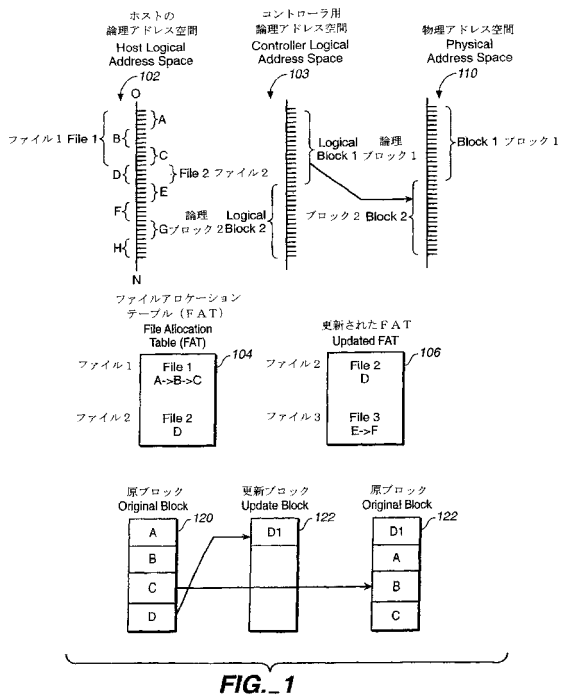
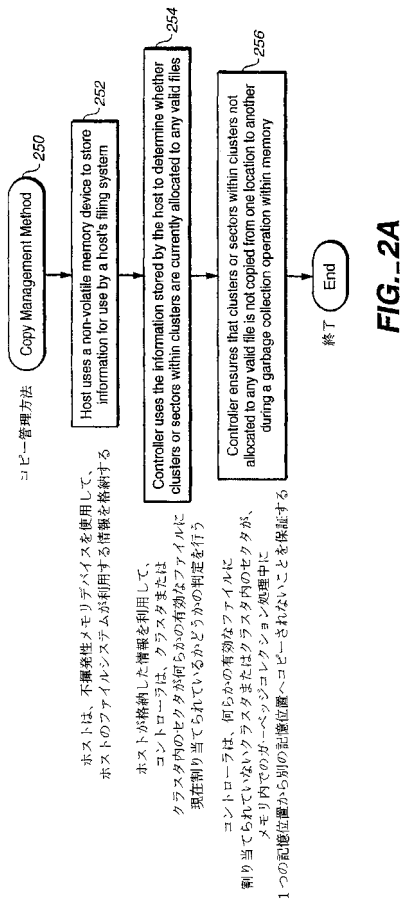
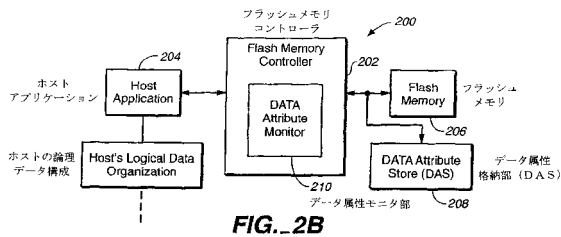


FIG._1

【図2A】



【図2B】



【図3B】

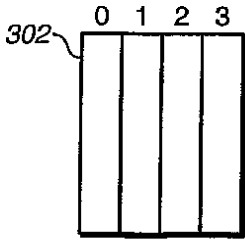


FIG._3B

【図3A】

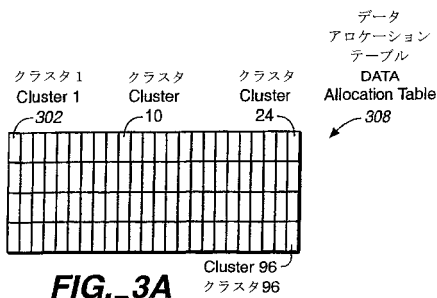


FIG._3A

【図3C】

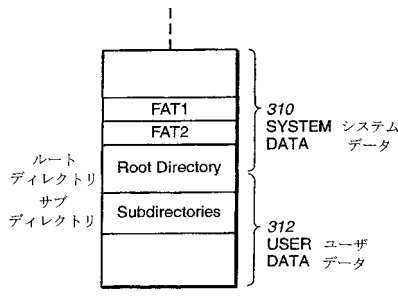
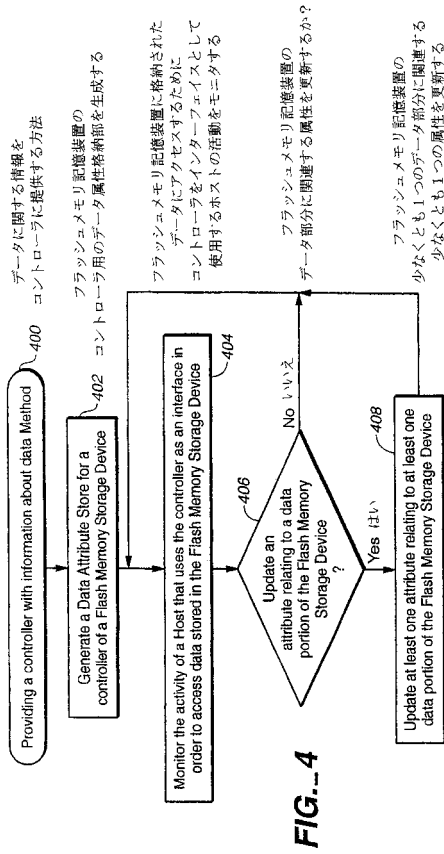
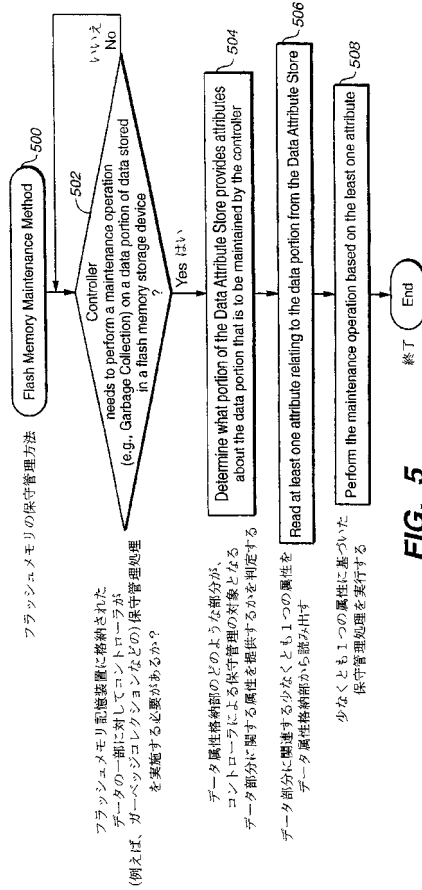


FIG._3C

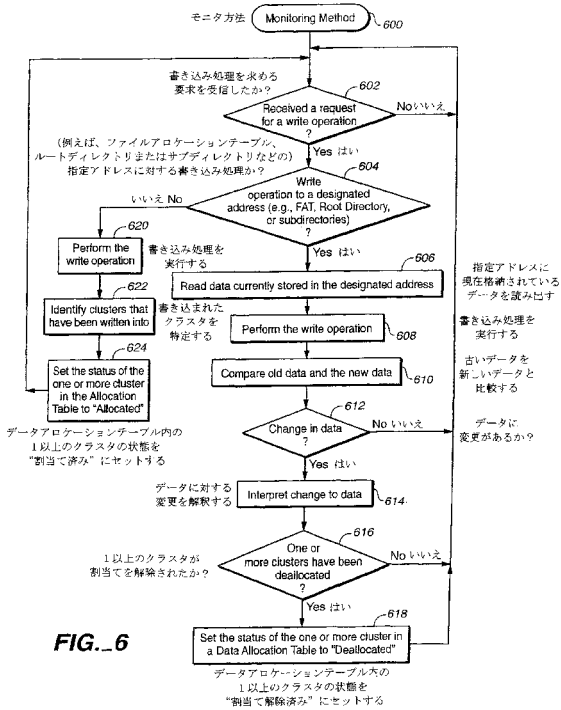
【 図 4 】



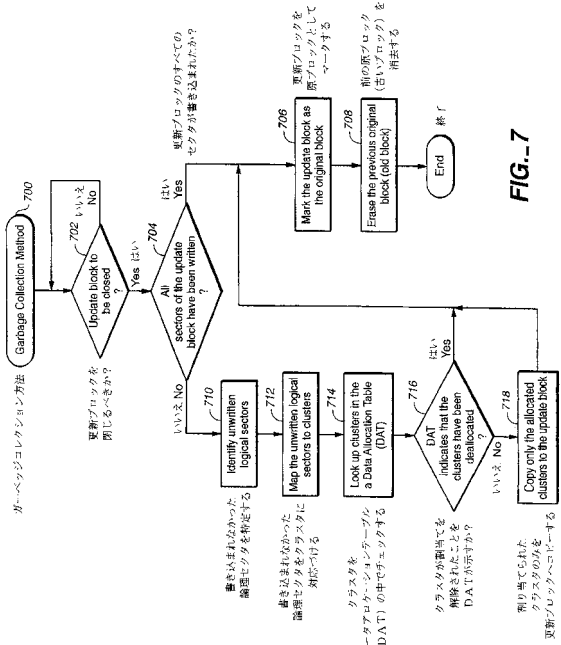
【 図 5 】



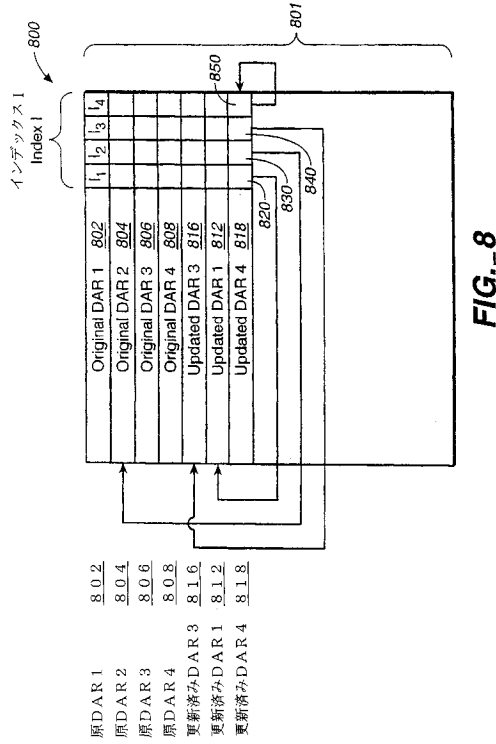
【 図 6 】



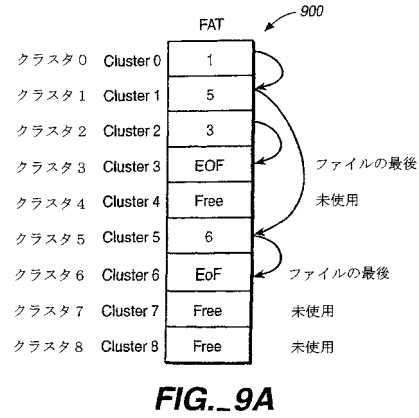
【 図 7 】



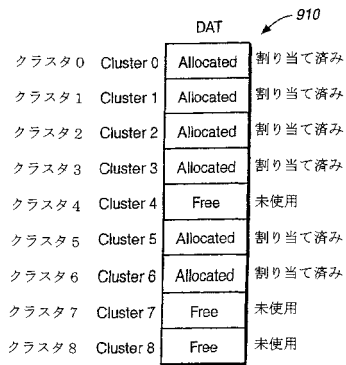
【 図 8 】



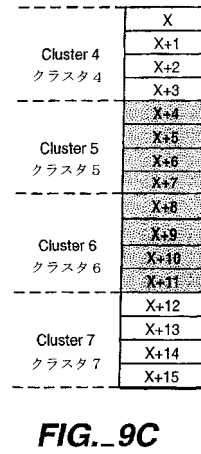
【 図 9 A 】



【 図 9 B 】



【 図 9 C 】



【図9D】

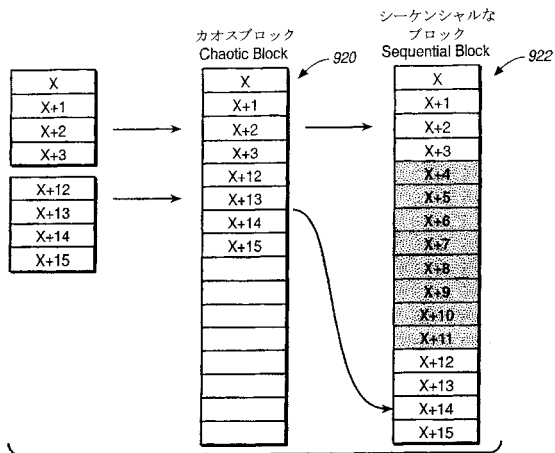


FIG. 9D

【図9E】

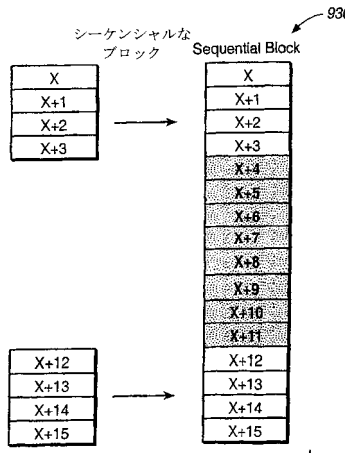


FIG. 9E

【図10】

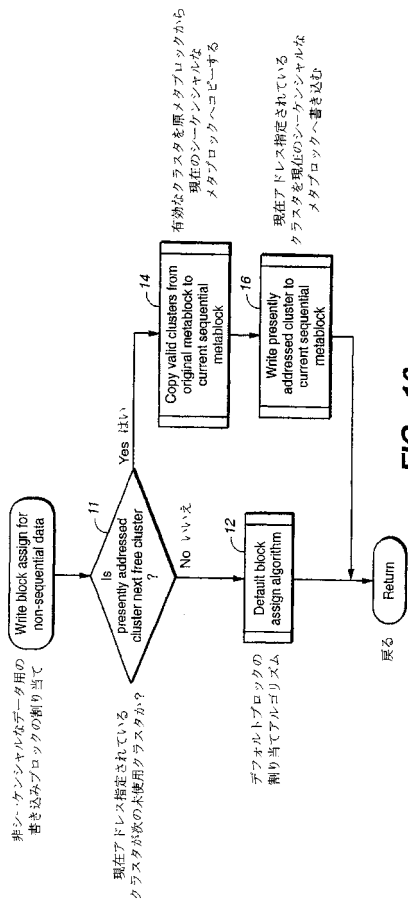


FIG. 10

【図11】

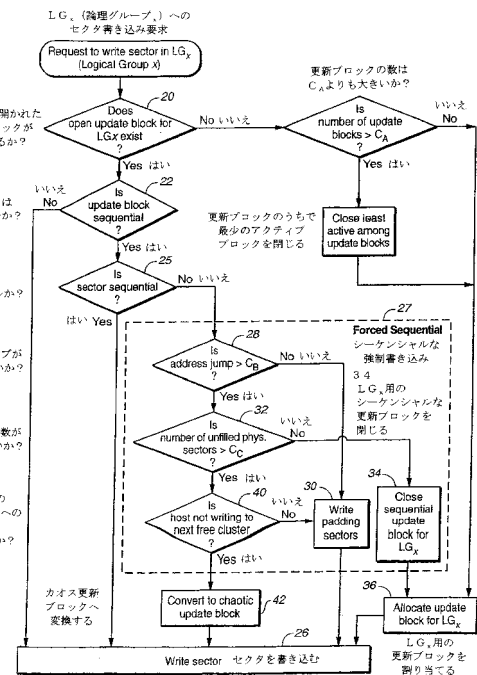


FIG. 11

【 図 1 2 A 】

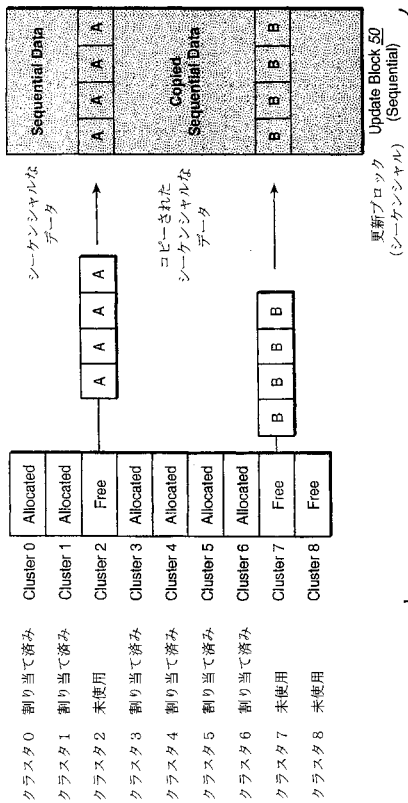


FIG. 12A

【 図 1 2 B 】

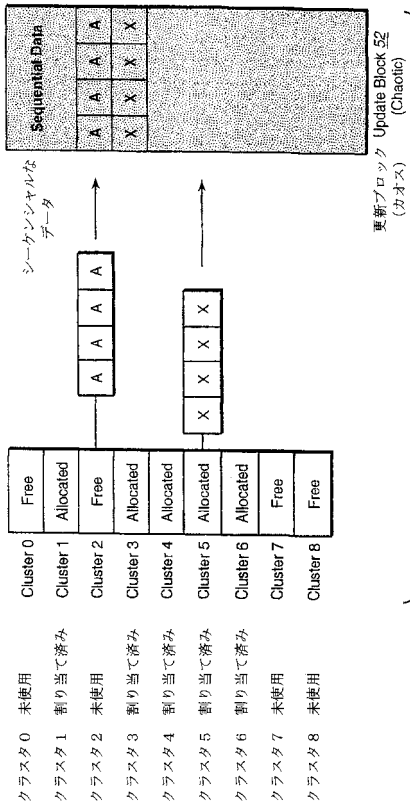


FIG. 12B

【 図 1 2 C 】

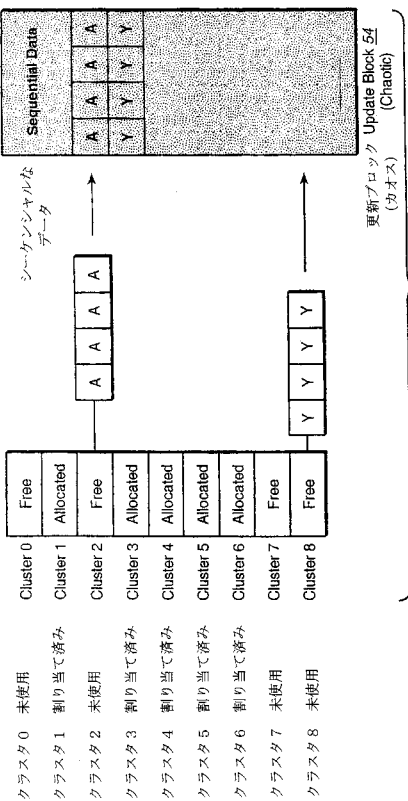


FIG. 12C

【 図 1 2 D 】

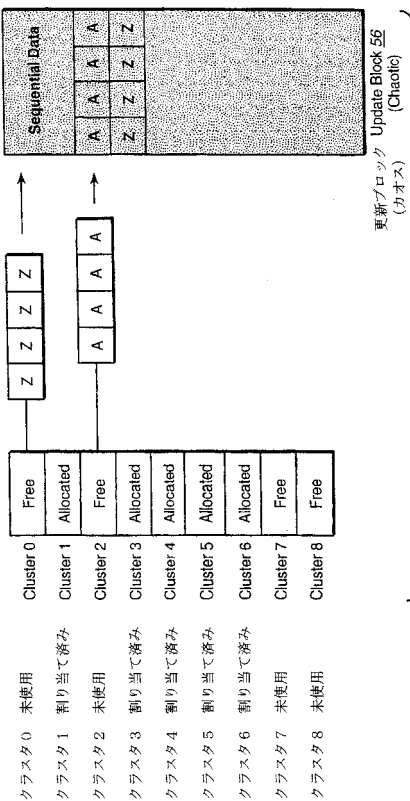


FIG. 12D

フロントページの続き

(72)発明者 スミス, ピーター ジョン
イギリス連邦共和国、EH22 3HA、ミッドロジアン スコットランド、エスクバンク、ポニ
ーリッジ ロード 21

審査官 原 秀人

(56)参考文献 特開平09-319645(JP, A)
特表2002-513484(JP, A)
特開平10-124381(JP, A)
特開平06-266587(JP, A)

(58)調査した分野(Int.Cl., DB名)
G06F 12/00-12/06