

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4910360号
(P4910360)

(45) 発行日 平成24年4月4日(2012.4.4)

(24) 登録日 平成24年1月27日(2012.1.27)

(51) Int.Cl.	F 1	
G 0 6 F 12/08 (2006.01)	G 0 6 F	12/08 5 0 1 E
G 1 1 C 16/02 (2006.01)	G 0 6 F	12/08 5 1 7 Z
	G 1 1 C	17/00 6 0 1 A
	G 1 1 C	17/00 6 0 1 T
	G 0 6 F	12/08 5 5 1 B
請求項の数 10 (全 29 頁) 最終頁に続く		

(21) 出願番号	特願2005-305540 (P2005-305540)	(73) 特許権者	000002185
(22) 出願日	平成17年10月20日(2005.10.20)		ソニー株式会社
(65) 公開番号	特開2007-115004 (P2007-115004A)		東京都港区港南1丁目7番1号
(43) 公開日	平成19年5月10日(2007.5.10)	(74) 代理人	100094053
審査請求日	平成20年10月9日(2008.10.9)		弁理士 佐藤 隆久
		(72) 発明者	西原 利幸
			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	吉岡 大助
			東京都品川区北品川6丁目7番35号 ソニー株式会社内
		審査官	原 秀人
最終頁に続く			

(54) 【発明の名称】 記憶装置、コンピュータシステム、およびデータ書き込み方法

(57) 【特許請求の範囲】

【請求項1】

複数のセクタにより形成されるページ単位で書き込みが行われる主記憶媒体であるフラッシュメモリデバイスと、

エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリと、

外部から入力された書き込みコマンドとアドレス情報をもとに、入力データの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに、適時選択する制御回路と、を有し、

上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、

上記単位区画は、ひとつまたは複数の書き込みページを含み、上記フラッシュメモリデバイスの一括書き込み単位であり、

上記制御回路は、

上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割し、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する機能を有し、

各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所

10

20

属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する

記憶装置。

【請求項 2】

上記制御回路は、

書き込み先が上記フラッシュメモリデバイスに選択されたサブブロックのデータのうち、上記キャッシュメモリにヒットしたデータがある場合、対応するキャッシュメモリ上のエントリを解放する

請求項 1 記載の記憶装置。

【請求項 3】

上記制御回路は、

書き込み先がフラッシュメモリデバイスに選択されたサブブロックのデータのうち、キャッシュメモリにヒットしたデータがある場合、対応するキャッシュメモリ上のエントリを書き込みデータで更新する

請求項 1 または 2 記載の記憶装置。

【請求項 4】

上記制御回路は、

書き込み先が上記フラッシュメモリデバイスに選択されたサブブロックと同じ区画内の領域に対応するエントリが上記キャッシュメモリに存在する場合、対応するエントリを当該サブブロックのデータと合成して同時にフラッシュメモリに書き戻す

請求項 1 から 3 のいずれかーに記載の記憶装置。

【請求項 5】

複数のセクタにより形成されるページ単位で書き込みが行われる主記憶媒体であるフラッシュメモリデバイスと、

エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリと、

外部から入力された書き込みコマンドとアドレス情報をもとに、入力データの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに、適時選択する制御回路と、を有し、

上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、

上記単位区画は、ひとつまたは複数の消去ブロックを含み、上記フラッシュメモリデバイスの一括消去単位であり、

上記制御回路は、

上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて 1 または複数のセクタを含むサブブロックに分割し、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する機能を有し、

各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する

記憶装置。

【請求項 6】

上記制御回路は、

各サブブロックに対して、対応区画におけるフラッシュメモリ上の空き状態を検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する

請求項 5 記載の記憶装置。

【請求項 7】

アプリケーションソフトウェアを実行する処理ユニットと、

10

20

30

40

50

複数のセクタにより形成されるページ単位で書き込みが行われるファイルストレージであるフラッシュメモリデバイスと、

エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリ領域と、を有し、

上記処理ユニットは、

ドライバソフトの制御のもとに、アプリケーションソフトウェアからの上記ファイルストレージへの書き込み要求を受けて、当該アドレス情報をもとに、入力データの書き込み先を上記フラッシュメモリデバイスまたは上記キャッシュメモリの何れかに、適時選択する制御回路を含み、

上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、

上記単位区画は、ひとつまたは複数の書き込みページを含み、上記フラッシュメモリデバイスの一括書き込み単位であり、

上記制御回路は、

上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割し、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する機能を有し、

各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する

コンピュータシステム。

【請求項8】

アプリケーションソフトウェアを実行する処理ユニットと、

複数のセクタにより形成されるページ単位で書き込みが行われるファイルストレージであるフラッシュメモリデバイスと、

エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリ領域と、を有し、

上記処理ユニットは、

ドライバソフトの制御のもとに、アプリケーションソフトウェアからの上記ファイルストレージへの書き込み要求を受けて、当該アドレス情報をもとに、入力データの書き込み先を上記フラッシュメモリデバイスまたは上記キャッシュメモリの何れかに、適時選択する制御回路を含み、

上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、

上記単位区画は、ひとつまたは複数の消去ブロックを含み、上記フラッシュメモリデバイスの一括消去単位であり、

上記制御回路は、

上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割し、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する機能を有し、

各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する

コンピュータシステム。

【請求項9】

複数のセクタにより形成されるページ単位で書き込みが行われるフラッシュメモリデバ

10

20

30

40

50

イスと、エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリ領域を有する記憶装置へのデータ書き込み方法であって、

上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、

上記単位区画は一つまたは複数の書き込みページより形成されるフラッシュメモリデバイスへの一括書き込み単位であり、

上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、

上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割する分割ステップと、

各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する選択ステップと、を有し、

上記選択ステップでは、

各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する

データ書き込み方法。

【請求項10】

複数のセクタにより形成されるページ単位で書き込みが行われるフラッシュメモリデバイスと、エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリ領域を有する記憶装置へのデータ書き込み方法であって、

上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、

上記単位区画は、ひとつまたは複数の消去ブロックを含み、上記フラッシュメモリデバイスの一括消去単位であり、

上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、

上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割する分割ステップと、

各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する選択ステップと、を有し、

上記選択ステップでは、

各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する

データ書き込み方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、フラッシュメモリを有する記憶装置、この記憶装置を用いたコンピュータシステム、およびデータ書き込み方法に関するものであり、特に記憶装置におけるデータ転送の高速化に関するものである。

【背景技術】

【0002】

近年、デジタルスチルカメラやモバイルコンピュータ機器の記憶媒体として、フラッシュメモリが注目されている。

【0003】

フラッシュメモリは、トンネリングやホットエレクトロン加速を用いて、電子にゲート絶縁膜を通過させ、それらを浮遊ゲートやトラップ層に注入し、セルトランジスタの閾値を変化させることでデータを記憶させる半導体メモリである。積層ゲート構造やMNOS

10

20

30

40

50

構造等を用いたトランジスタ1つのみでメモリセルを構成できるため、安価かつ大容量のメモリを実現できる。

その代表例として、NAND型フラッシュメモリが挙げられる。

【0004】

図1は、NAND型フラッシュメモリの内部構成例を示す図である。

図1のNAND型フラッシュメモリは、ビット線BL1~BLnに接続された複数のメモリユニット1-1~1-nがアレイ状に(縦横)に配列されている。

たとえば、選択用トランジスタ2のゲートが選択ゲート線SL1に接続され、選択用トランジスタ3のゲートが選択ゲート線SL2に接続されている。また、各メモリセルN0~N15のゲートがワード線WL0~WL15に接続されている。

10

【0005】

各メモリセルN0~N15は積層ゲート構造を持ち、浮遊ゲートへの電荷蓄積量に従ってデータを記憶する。すなわち、浮遊ゲートに多くの電子が蓄積されていると、トランジスタの閾値が上昇するので、チャージされたビット線BL1~BLnからのメモリユニット1(-1~-n)への電流貫通の有無を、センスアンプ等を含むアクセス回路4で検出してデータ判定を行う。

【0006】

このようなNAND型フラッシュメモリは、メモリセル毎にビット線へのコンタクト領域を設ける必要もないので、特に大容量で安価な記憶装置の媒体に適している。

【0007】

20

ところで、一般にフラッシュメモリのプログラム速度は非常に遅く、セルあたり数百μ秒を必要とする。またデータの上書きはできないので、プログラムに先立って消去を行う必要があり、これには数m秒もの時間がかかる。このような問題に対しては、多くのメモリセルを並列処理することで対処している。

【0008】

すなわち、たとえば同一ワード線WL0に接続され、ページ単位を成すメモリセル群5を、同時一括に書き込み、さらに互いにメモリユニットを共有するページ群で構成されるセルブロック6を全て一括で消去することにより、プログラムの転送速度を向上させている。

【0009】

30

具体的には、たとえば非特許文献1には1GbのNAND型フラッシュメモリが掲載されており、ページサイズを2kバイト、消去ブロックサイズを128kBとしている。すなわち、一つのメモリアレイ内で128kバイトのメモリセル群を並列消去し、そこにメモリセルを2kバイト毎に並列でプログラムしていくことによって、10MB/sのプログラム転送速度を実現している。

【非特許文献1】ISSCC2002予稿集のp106、セッション6.4

【特許文献1】特開平5-216775号公報

【特許文献2】特開平6-349286号公報

【発明の開示】

【発明が解決しようとする課題】

40

【0010】

近年、ハードディスクの消費電力の大きさや、シーク時間の長さ、耐衝撃性や携帯性等の問題を解消すべく、フラッシュメモリにその代替が期待されている。

しかし上述の如く、フラッシュメモリにはアクセス単位を大きくしないと高速化できないという欠点がある。また、データの上書きができないので、書き換えには必ず消去が必要であり、その際の消去ブロックはさらに大きい。

このようにアクセス単位に対して消去単位が数十倍大きいのは、消去時間が長く、かつ書き込み時に非選択セルにディスタ urbが生じるフラッシュメモリには一般的な仕様である。しかしこれによりその書き込み効率は著しく低下する。

【0011】

50

たとえば上記2 kB単位のページ書き込み、128 kB単位のブロック消去による、転送速度10 MB/sフラッシュメモリを用いて、ハードディスク代替用の記憶装置を構成するとする。

その転送速度を、シリアルATA接続の高速ストレージで目標とされる160 MB/sにまで高めようとした場合、マルチバンクやマルチチップの構成を取りながら、たとえば16個のメモリアレイを並列動作させる必要がある。

【0012】

図2は、そのような記憶装置を構成するフラッシュメモリの概念図を示す図である。

図2において、高速転送を実現するため、16個のアレイAR0~AR15を同時に動作させる。この場合データ書き込み時は、たとえばページP0~P15を同時書き込みし、消去時はブロックB0~B15を同時消去する。このとき一括書き込み単位となる実ページ領域24は32 kB、一括消去単位となる実消去ブロック領域21は2 MBに達することになる。

10

【0013】

一方、通常のファイルシステムでは、最小の書き込み単位を成すクラスタサイズは4 kB程度であり、この単位でランダムなアクセスがなされる。

その場合、たとえばページP0とP1のみを書き換える要求が発生する。

しかし、上述のような装置でそのようなアクセスを行うと、結局実消去ブロック領域21全体を消去しなければならない。かつその中の非選択領域に有効ファイルがあれば、それを消失から保護する必要がある。その典型的な対処例は次のようになる。

20

【0014】

1. まずフラッシュメモリから、別途設けられたバッファメモリ23のメモリ領域22に、実消去ブロック領域21全体のデータを読み出す。

2. 次にメモリ領域22内で、ページP0とP1に相当するデータを更新する。

3. 次にフラッシュメモリ上のブロック領域21の消去を実行する。

4. 最後に上記消去済の領域21に、更新後のメモリ領域22のブロックデータを全て書き戻す。

【0015】

すなわち、4 kBバイトの書き込みのために、実際には消去と、2 MBのデータの読み出しおよび書き込みが必要となる。

30

具体的には、1ページ分のデータの読み出しおよび書き込みにはそれぞれ200 μsを要し、ブロックの消去には2 msを要するので、30 ms近くを必要とする。

【0016】

これに対し、予め予備の消去済みブロック領域27を用意しておいて、このブロック領域27にブロック領域21の元のデータとページP0とP1の更新後データを合成させて書き込みを行う手法もある。

この場合、仮想アドレス構成を用いて消去ブロック単位で論理アドレスと物理アドレスの対応を更新し、アクセス対象とされた論理アドレスに対応する物理ブロックは元の消去ブロック領域21からデータの移動先ブロック領域27に張り替えられる。

しかしこの場合も、有効データを消去ブロック領域21から移動先ブロック領域27に退避させる作業は必要である。またこの際、通常は元のブロック領域21を消去して、そちらを予備ブロックに変える。したがって、結局は従来とほぼ同様の読み出し、書き込み、消去が必要であり、大きなオーバーヘッドが生じることに変わりはない。

40

【0017】

実際にファイルを更新する場合、該当ファイルのみではなく、その管理領域やログの記載等、関連する複数の小領域を同時に更新するのが普通である。またファイル自体も複数の小領域に断片化して存在しているケースがある。

したがって、上述の如き制約があると、現実の転送性能は著しく低いものになってしまう。

【0018】

50

また上述の例では4 kバイトの書き換えであったが、ハードディスク等の通常のファイル記憶装置はさらに小さい512バイト単位の書き換えまでサポートしている。

しかしどのように小単位の書き換えであっても、上述の如き消去ブロック全体の書き換えが必要になる。

【0019】

このような事情から、フラッシュメモリを用いた現在の記憶装置は、特に書き込み時においては、ハードディスクをはるかに下回る転送性能しか得られていない。さらに上述のような冗長性は、単に転送性能を悪化させるのみならず、フラッシュメモリの消去回数を増加させ、その寿命まで低下させてしまう。

【0020】

このような問題に対して、たとえば特許文献1（特開平05-216775号公報）、特許文献2（特開平06-349286号公報）においてはキャッシュメモリの使用が記載されている。

ここで例示されているのは、フラッシュメモリの消去単位であるブロックデータ、あるいは一括書き込み単位であるページデータを、そのままエントリとしてキャッシュメモリに保存するものである。更新しようとするデータを含むブロックがキャッシュに格納されていれば、すなわちキャッシュがヒットすれば、キャッシュ内のデータのみが更新されるので、上述のような冗長な動作は直ちには発生しない。

【0021】

しかしこのようなキャッシュメモリの使用は、必ずしも効率の向上をもたらさない。たとえば、動画映像のように巨大な連続データよりなるファイルを記録する場合、キャッシュメモリは録画の間中ミスヒットを続け、新たなエントリを追加するために、既存のエントリの解放が必要になる。そのため、外部からキャッシュメモリに書き込んだデータと等量のデータが、常時キャッシュメモリからフラッシュメモリに書き戻される。従ってキャッシュへの書き込む工程は単なるオーバーヘッドとなってしまう、逆にプログラム能力を低下させてしまう。

【0022】

本発明の目的は、フラッシュメモリを主媒体とする記憶装置に小容量でもヒット率が高く、アクセスのオーバーヘッドも少ないキャッシュメモリを付与することが可能になり、フラッシュメモリへの書き込みを高速化し、書き換え回数も低減させることが可能な半導体記憶装置およびコンピュータシステム、並びにデータ書き込む方法を提供することにある。

【課題を解決するための手段】

【0023】

本発明の第1の観点の記憶装置は、複数のセクタにより形成されるページ単位で書き込みが行われる主記憶媒体であるフラッシュメモリデバイスと、エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリと、外部から入力された書き込みコマンドとアドレス情報をもとに、入力データの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに、適時選択する制御回路と、を有し、上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、上記単位区画は、ひとつまたは複数の書き込みページを含み、上記フラッシュメモリデバイスの一括書き込み単位であり、上記制御回路は、上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割し、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する機能を有し、各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する。

10

20

30

40

50

【 0 0 2 8 】

好適には、上記制御回路は、書き込み先が上記フラッシュメモリデバイスに選択されたサブブロックのデータのうち、上記キャッシュメモリにヒットしたデータがある場合、対応するキャッシュメモリ上のエントリを解放する。

【 0 0 2 9 】

好適には、書き込み先がフラッシュメモリデバイスに選択されたサブブロックのデータのうち、キャッシュメモリにヒットしたデータがある場合、対応するキャッシュメモリ上のエントリを上記書き込みデータで更新する。

【 0 0 3 0 】

好適には、上記制御回路は、書き込み先が上記フラッシュメモリデバイスに選択されたサブブロックと同じ区画内の領域に対応するエントリが上記キャッシュメモリに存在する場合、対応するエントリを当該サブブロックのデータと合成して同時にフラッシュメモリに書き戻す。

【 0 0 3 1 】

また、本発明の第1の観点の記憶装置は、複数のセクタにより形成されるページ単位で書き込みが行われる主記憶媒体であるフラッシュメモリデバイスと、エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリと、外部から入力された書き込みコマンドとアドレス情報をもとに、入力データの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに、適時選択する制御回路と、を有し、上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各单位区画は複数のセクタを含み、上記単位区画は、ひとつまたは複数の消去ブロックを含み、上記フラッシュメモリデバイスの一括消去単位であり、上記制御回路は、上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割し、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する機能を有し、各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する。

【 0 0 3 4 】

好適には、上記制御回路は、各サブブロックに対して、対応区画におけるフラッシュメモリ上の空き状態を検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する。

【 0 0 3 5 】

本発明の第2の観点のコンピュータシステムは、アプリケーションソフトウェアを実行する処理ユニットと、複数のセクタにより形成されるページ単位で書き込みが行われるファイルストレージであるフラッシュメモリデバイスと、エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリ領域と、を有し、上記処理ユニットは、ドライバソフトの制御のもとに、アプリケーションソフトウェアからの上記ファイルストレージへの書き込み要求を受けて、当該アドレス情報をもとに、入力データの書き込み先を上記フラッシュメモリデバイスまたは上記キャッシュメモリの何れかに、適時選択する制御回路を含み、上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各单位区画は複数のセクタを含み、上記単位区画は、ひとつまたは複数の書き込みページを含み、上記フラッシュメモリデバイスの一括書き込み単位であり、上記制御回路は、上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割し、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリ

10

20

30

40

50

の何れかに選択する機能を有し、各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する。

また、本発明の第2の観点のコンピュータシステムは、アプリケーションソフトウェアを実行する処理ユニットと、複数のセクタにより形成されるページ単位で書き込みが行われるファイルストレージであるフラッシュメモリデバイスと、エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリ領域と、を有し、上記処理ユニットは、ドライバソフトの制御のもとに、アプリケーションソフトウェアからの上記ファイルストレージへの書き込み要求を受けて、当該アドレス情報をもとに、入力データの書き込み先を上記フラッシュメモリデバイスまたは上記キャッシュメモリの何れかに、適時選択する制御回路を含み、上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、上記単位区画は、ひとつまたは複数の消去ブロックを含み、上記フラッシュメモリデバイスの一括消去単位であり、上記制御回路は、上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割し、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する機能を有し、各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する。

【0036】

本発明の第3の観点は、複数のセクタにより形成されるページ単位で書き込みが行われるフラッシュメモリデバイスと、エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリ領域を有する記憶装置へのデータ書き込み方法であって、上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、上記単位区画は一つまたは複数の書き込みページより形成されるフラッシュメモリデバイスへの一括書き込み単位であり、上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割する分割ステップと、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する選択ステップと、を有し、上記選択ステップでは、各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する。

また、本発明の第3の観点は、複数のセクタにより形成されるページ単位で書き込みが行われるフラッシュメモリデバイスと、エントリがセクタ単位で構成される、上記フラッシュメモリデバイスのためのキャッシュメモリ領域を有する記憶装置へのデータ書き込み方法であって、上記フラッシュメモリデバイスの書き込み単位または消去単位をもとに、単位区画が設定されており、各単位区画は複数のセクタを含み、上記単位区画は、ひとつまたは複数の消去ブロックを含み、上記フラッシュメモリデバイスの一括消去単位であり、上記セクタまたは連続したセクタ群へのブロックデータの書き込み命令を受けた場合、上記ブロックデータを上記単位区画に応じて1または複数のセクタを含むサブブロックに分割する分割ステップと、各サブブロックごとに、書き込むべきセクタ数または、上記キャッシュメモリの消費されるまたは解放すべきエントリ数に従って、各サブブロックの書き込み先を上記フラッシュメモリデバイスまたはキャッシュメモリの何れかに選択する選択ステップと、を有し、上記選択ステップでは、各サブブロックに対して、区画におけるデータの占有状態、あるいは同区画または所属データに対応するキャッシュメモリのエントリを検査し、当該検査結果に応じて各サブブロックの書き込み先を選択する。

10

20

30

40

50

【 0 0 3 7 】

本発明によれば、一般的なキャッシュメモリを搭載したストレージシステムにおける課題は、書き込み入力されたデータが、状況にかかわらず、必ず一旦キャッシュメモリに書き込まれていたことにある。

それに対して本発明の記憶装置は、外部から入力された書き込みコマンドとアドレス情報をもとに、入力されたデータの書き込み先をキャッシュメモリかフラッシュメモリの何れかに、装置内部で適時選択する。これによって、フラッシュメモリへの書き込みに負担の大きい小領域の更新や小型のファイルについてはキャッシュメモリ側に書き込み、キャッシュメモリへの書き込みが無駄になるような連続データよりなる大型ファイルは直接フラッシュメモリに書き込むようにする。

10

【発明の効果】

【 0 0 3 8 】

本発明によれば、フラッシュメモリを主媒体とする記憶装置に小容量でもヒット率が高く、アクセスのオーバーヘッドも少ないキャッシュメモリを付与することが可能になり、フラッシュメモリへの書き込みを高速化し、書き換え回数も低減させることが可能になる。

このようなシステムではフラッシュメモリを高速なハードディスクのように使用することもでき、待機時の消費電力が低く、コンパクトで高速なストレージシステムを獲得できる。

【発明を実施するための最良の形態】

20

【 0 0 3 9 】

以下、本発明の実施形態を図面に関連付けて説明する。

【 0 0 4 0 】

図3は、本発明の第1の実施形態に係るファイル記憶装置を示す構成図である。

【 0 0 4 1 】

ファイル記憶装置30は、インターフェース回路(I/F)31、ページバッファ32、メモリバス33、制御回路34、NAND型フラッシュメモリ35、36、制御回路37、キャッシュメモリ38、および内部バス39を含んで構成されている。また、40~43はページ領域を示し、制御回路34は、アドレス変換テーブル44、およびキャッシュの管理テーブル45を有する。

30

そして、たとえば、ページバッファ32、メモリバス33、NAND型フラッシュメモリ35、36、制御回路37、および内部バス39によりフラッシュメモリデバイスFLMが形成される。

【 0 0 4 2 】

ファイル記憶装置30の内部においては、32ビットのメモリバス33に、16ビットの入出力を持つ2チップのNAND型フラッシュメモリ35、36が並列に接続されている。2つのチップのNAND型フラッシュメモリ35、36は読み出しや書き込みにおいて、同時並列にアクセスされる。

各々のフラッシュメモリ35、36は書き込みや読み出しのアクセスをたとえば4kBのページ単位で行う。

40

したがって、実ページサイズとしては8kBが一括アクセスされることになる。

ページバッファ32はアクセスされたページ領域のデータを一時記憶するページバッファである。

フラッシュメモリ35、36とページバッファ32との間のデータの送受は、制御回路37で制御されている。

さらに制御回路37は、必要に応じて転送データにECC符号化によるエラー補正を施したり、フラッシュメモリ内の欠陥ブロックの管理を実施する。両フラッシュメモリ35、36はページバッファ32を介してファイル記憶装置30の内部バス39との間でデータを入出力する。

すなわち、フラッシュメモリ35、36あるいはメモリバス33上に接続されたページ

50

バッファ32と転送制御回路37を併せた回路群FLMDは、実質的に一つのフラッシュメモリデバイスを構成し、記憶装置30の内部バス39に接続されているとみなすことができる。

【0043】

さらに内部バス39には書き込み用のキャッシュメモリ38とインターフェース回路31、および制御回路34が接続されている。

インターフェース回路31はATAやPCIエクスプレス等の規格に従ってホストとの間で、データやコマンドの送受を行う。

制御回路34は、ファイル記憶装置30の内部においてページバッファ32、キャッシュメモリ38、およびインターフェース回路31の間のデータの送受を管理する。ここに内蔵されたRAMには、ページ単位の仮想アドレスを管理するアドレス変換テーブル44およびキャッシュメモリ38の管理テーブル45が構築されている。

10

【0044】

本記憶装置30は、ハードディスク等と同様に512バイト(Byte)のセクタをアクセス単位とする。装置内部では簡単のため、16進数のアドレスが次のように割り振られるとする。

たとえば外部入力アドレスが"0x5500C"であった場合、上位16ビットの"0x5500"はページアドレス、下位4ビットの"0xC"はページ領域内のセクタアドレスであり、1ページ中には16のセクタが含まれる。

本記憶装置30は1セクタ単位でランダムアクセスが可能であり、それは後述するようにページバッファ32またはキャッシュメモリ38を介して実施される。

20

【0045】

キャッシュメモリ38へのエントリは、実ページの1/4、すなわち4セクタを単位として構成する。したがって、各々のエントリは、対応する16ビットのページアドレスに加え、4ビットのセクタアドレス中上位2ビットを持って管理されることになる。

上述の例では実ページサイズが8kBなので、1エントリのサイズは2kBである。たとえば128kBのRAMをキャッシュメモリとした場合、約64個のエントリを持つことが可能である。

【0046】

図4は、キャッシュメモリのキャッシュ管理テーブル45の構成例を示す図である。

30

ここではテーブル管理の容易さとヒット率を勘案し、ページ内セクタアドレスの上位2ビットをインデックスとして、四つのセットを構成した。各セットは独立に16個ずつのエントリを保有する。すなわち4セット16ウエイのセットアソシアティブ構成が採用されている。テーブル内の各フィールドには、そこにエントリが存在する場合、各エントリに対応するページアドレスが格納されている。

【0047】

たとえば、外部からアドレス"0x5500C"のセクタへのアクセスが要求された場合、まずセクタアドレス"0xC"の上位2ビットがインデックスとして使用され、"11"に対応するセットが選択される。

さらに、このセットに登録されている最大16個のエントリが検索され、"0x5500"のページアドレスに対応するエントリの有無が判定される。

40

【0048】

さらに、本実施形態においては、ページ単位の仮想アドレス管理が採用されている。

図5は、アドレス変換テーブル44の構成例を示す図である。

メモリへの具体的アクセスは以下のようなされる。

【0049】

読み出しの際、外部から"0x5500C"のセクタアドレスが入力されると、制御回路34はまずキャッシュ管理テーブル45からヒットの有無を判定する。キャッシュにヒットしていれば、データはキャッシュメモリ(46)より読み出される。

【0050】

50

一方ミスヒットの場合、次の手順でフラッシュメモリ35, 36からページバッファ32へ該当する実ページを読み出す。

まず、ページアドレス"0x5500"を受けて内蔵RAMにアクセスし、アドレス変換テーブル44から論理ページアドレス(LA: LOGICAL ADDRESS)"0x5500"に対応する物理ページアドレス(PA: PHYSICAL ADDRESS)"0x00B0"を取得する。

【0051】

この物理ページアドレスPA中、上位の"0x00"はメモリチップ、すなわちフラッシュメモリ35, 36内の消去ブロックBL1、BL2のアドレスである。

下位"0xB0"は各消去ブロック内のページ領域40, 41のアドレスである。この物理アドレスをもとにメモリチップであるフラッシュメモリ35, 36が共にアクセスされ、各々からページアドレス"0x00B0"に格納されたページ領域40, 41のデータが読み出され、ページバッファ32に格納される。

インターフェース回路31はその中から、下位アドレス"0xC"に相当するセクタを選択し、外部に出力する。

【0052】

一方、書き込みの際には、キャッシュメモリ38に書き込む場合、1セクタ単位で高速に書き込みが実施できる。

一方、フラッシュメモリ35, 36に書き込む場合には、一旦対応する実ページを読み出し時と同様の手順でフラッシュメモリ35, 36からページバッファ32に読み出し、ページバッファ32上で対応するセクタを書き換えて、しかる後にフラッシュメモリ上に書き戻す必要がある。

1セクタのみの書き換えでも同じ手順を要し、実ページ全体(16セクタ分)のデータの読み出しと書き戻しが必要である。

さらに前述の如く、フラッシュメモリ35, 36への書き込みは時間がかかる上、構成によっては消去に伴う制約がある。したがって、そのパフォーマンスはキャッシュメモリ38への書き込みに比較して大きく劣る。

【0053】

ただし、キャッシュメモリ38への書き込みにおいても、ミスヒットし、かつメモリ内にエントリを新規に追加するための空き領域が無ければ、一旦既存のエントリを解放し、空きをつくらねばならない。解放したエントリのデータは結局フラッシュメモリに書き戻す必要があり、その作業はフラッシュメモリへの書き込みと同じである。

【0054】

ところでATA等、通常のファイルストレージにおけるインターフェース仕様では、データアクセスは幾つかの連続したセクタよりなるブロック単位での転送指定で実行される。

たとえば、コマンドは「先頭アドレスxxxからxx個のセクタ群に対して書き込みを実施」という形態で送受される。上述の如く、キャッシュメモリとフラッシュメモリとは書き込みの性能が大きく異なるので、そのようなブロックの構成に応じてアクセス手順を最適化することがパフォーマンスの向上に重要となる。

【0055】

本実施形態においては、上記のような転送ブロックを実ページごとの区画で分割し、各々の区画単位でアクセス手順を決定する。

たとえば、セクタアドレス"0x0000C"~"0x00020"の範囲のブロックデータが書き込まれるとすると、それらはページアドレスをもとに以下の如く3つのサブブロックに分割される。

【0056】

区画1(サブブロック1): "0x0000C" ~ "0x0000F"
 区画2(サブブロック2): "0x00010" ~ "0x0001F"
 区画3(サブブロック3): "0x00020"

10

20

30

40

50

【 0 0 5 7 】

このように分割した各々のサブブロックについて、最適な書き込み先を選択していく。
図 6 に従って、以下にその判定基準の例を示す。

【 0 0 5 8 】

ここでキャッシュメモリのエントリが満杯で、かつ書き込みセクタが全てミスヒットするという、最悪のケースへの対応を考えてみる。

そこに書き込みを行うには、前述の如く、書き込みに要するだけのエントリを解放し、それをフラッシュメモリに書き戻す必要が生じる。したがって、書き込むべきデータサイズが大きいほどその際の処理量が大きくなり、キャッシュメモリ使用の不利が増す。

【 0 0 5 9 】

たとえば、サブブロック 2 はセクタ 16 個より成り、実ページ " 0 x 0 0 0 1 " をまるごと (すべて) 含んでいる。これをキャッシュメモリ 38 に格納すると、エントリ 4 個を消費する。この場合、最悪 4 個の互いに異なるページに属するエントリを解放する必要があり、フラッシュメモリに 4 ページ分の書き込みを行う必要が生じる。

すなわち、フラッシュメモリ 35, 36 からページバッファ 32 に実ページデータを読み出し、各エントリに対応するデータ領域を更新して再度フラッシュメモリに書き戻す作業を 4 回繰り返さねばならない。

一方、このサブブロックをフラッシュメモリにそのまま書き込めば、単に 1 ページ分の書き込みで良い。したがって、このようなケースではキャッシュメモリへの書き込みは、むしろ逆効果となるリスクが高い。

【 0 0 6 0 】

一方、サブブロック 1 はセクタ 4 個、サブブロック 3 はセクタ 1 個より成り、ともにキャッシュメモリ 38 の 1 エントリ分を消費するのみである。したがって、上記最悪ケースでもキャッシュから解放し、フラッシュメモリにデータを書き戻すべきエントリは 1 個であり、フラッシュメモリへの書き込みは同じ 1 ページ分ですむ。キャッシュメモリ 38 にヒットしていたり、エントリ追加の空きがある場合は、フラッシュメモリ 38 にアクセスすることなく、高速に処理できる。したがって、このサブブロックは殆どの場合キャッシュメモリ 38 側に書き込む方が効率が良い。

【 0 0 6 1 】

ここにおいて、有効な書き込み手法の一つは、実ページ単位で区画分けされた各サブブロックごとに、そのデータの実ページ内の占有状態を検査し、それに応じて書き込み先をフラッシュメモリかキャッシュメモリの何れかに選択することである。

【 0 0 6 2 】

たとえば、各サブブロックにおいて書き込むべきセクタ数を判定基準とし、5 個以上ならばフラッシュメモリ 35, 36 に、4 個以下ならキャッシュメモリ 38 に書き込みを実施する。あるいはキャッシュメモリに書き込む際に消費されるエントリ数を基準として採用しても良い。たとえば消費されるエントリ数が 2 個以上ならフラッシュメモリに、1 個ならキャッシュメモリに書き込みを実施する。

各サブブロックにおけるセクタ数や消費されるエントリ数は、アドレス情報から瞬時に取得できる。したがって、このような判定アルゴリズムを用いてデータの書き込み先を適時選択すれば、簡便に書き込み効率及び性能を向上させることができる。

【 0 0 6 3 】

また、さらにキャッシュメモリのエントリを検査すれば、より精緻な判定が可能になる。

まず、サブブロックに属するデータのキャッシュメモリへのヒットの有無を、書き込みの事前に検査する。これはアドレス情報をもとにキャッシュ管理テーブル 45 を検索することで実施できる。これにより、このサブブロックをキャッシュメモリに書き込む場合、新規エントリを何個追加する必要があるかが明らかになる。これを判定の基準とすることで、よりの確な選択が可能になる。

【 0 0 6 4 】

仮に、サブブロック 2 が 4 個のエントリを消費する場合でも、既に 2 3 個がキャッシュメモリ 3 8 内であれば（すなわちヒットしていれば）新規の追加は 1 個で良い。たとえば 2 個以上のエントリを新規に追加する必要がある場合にはフラッシュメモリ 3 5 , 3 6 にそのまま書き込み、1 個以下のエントリしか追加の必要がない場合にはキャッシュメモリ 3 8 に書き込む。

【 0 0 6 5 】

あるいは、さらにキャッシュメモリ 3 8 の空き状態も勘案し、解放すべきエントリ数を選択の基準にしても良い。仮に、3 つのエントリを新規に追加する必要がある場合でも、そのうち 2 つの対応するセット内に空きがあれば、解放すべきエントリは一つで良い。たとえば 2 個以上のエントリをする必要がある場合にはフラッシュメモリ 3 5 , 3 6 にそのまま書き込み、1 個以下のエントリしか解放の必要がない場合にはキャッシュメモリ 3 8 に書き込む。

10

【 0 0 6 6 】

あるいは、サブブロックをフラッシュメモリ 3 5 , 3 6 へ書き込む際に、同時にキャッシュメモリ 3 8 から書き戻しが可能なデータを検査し、判定に使用方法も有効である。

たとえば、上記サブブロック 1 を、仮にフラッシュメモリにそのまま書き込もうとした場合、キャッシュテーブル 4 5 から同じ区画 1 に対応するエントリを検索する。すなわちサブブロック 1 は、論理ページアドレス " 0 x 0 0 0 0 " に属するが、同じページに対応するエントリがキャッシュメモリ 3 8 内に存在すれば、そのデータは同時にフラッシュメモリ 3 5 , 3 6 に書き戻すことができる。

20

【 0 0 6 7 】

該当したエントリはクリーンとなり、必要に応じて解放することも可能である。その場合、サブブロックの書き込みデータが 4 セクタ（エントリが 1 つ分）でも、実際には複数のエントリから多くのデータが書き込まれ、効率の良い書き込みが実施されることになる。たとえば、フラッシュメモリ 3 5 , 3 6 へ同時書き込みが可能な総セクタ数を基準とし、それが規定値より大きければフラッシュメモリにそのまま書き込み、規定値以下ならキャッシュメモリ 3 8 に書き込む。

【 0 0 6 8 】

このように、選択アルゴリズムには種々のバリエーションが存在し、さらにそれらは場合に応じて互いに組み合わせることも可能である。

30

【 0 0 6 9 】

本発明の記憶装置における基本的な書き込み手順は以下の通りである。

- 1 . 入力コマンドとアドレス情報を取得する。
- 2 . 書き込みページによる区画をもとに、ブロックデータをサブブロックに分割し、各々のブロックについて順次書き込み処理を実施する。
- 3 . その際、サブブロック内のデータの占有状態、或いはキャッシュメモリ内のエントリの検査を行い、その結果に応じて書き込み先を選択する。

【 0 0 7 0 】

また、上記 3 . の判定においては、各々のサブブロックに対して、

- ・キャッシュメモリにおけるエントリの解放も考慮した上で、実質的なフラッシュメモリへの書き込み回数を減らす。
- ・フラッシュメモリへのアクセスが必要な場合は、より多くのセクタが同時に書き込まれるようにする。

40

という視点に立ち、さらに判定に要する時間や処理の煩雑さも鑑みて、選択アルゴリズムが決定されるのが望ましい。

【 0 0 7 1 】

以下に、ブロックデータの書き込み手順について説明する。

図 7 は、本実施形態に係るブロックデータの書き込み手順を示すフローチャートである。ここで、外部からセクタ単位の先頭アドレスと書き込みセクタ数が指定され、セクタま

50

たは連続した複数セクタよりなるブロックデータが入力されたとする。セクタは一般的なファイルストレージの基本アクセス単位であり、通常512バイト(Byte)であるが、必ずしもそのサイズに限定されるものではない。

【0072】

<1> ブロックデータの区画分け

制御回路34は、まず入力されたブロックデータを、前述の如く実ページに従って区画分けを行い、各サブブロックごとに以降の処理を行う(ST1)。通常は先頭セクタを含む区画のサブブロックから順に書き込み処理を実行していく。全サブブロックの書き込みが完了すれば、処理は終了となる(ST2)。

【0073】

<2> サブブロックの書き込み処理

処理対処として選択されたサブブロックの区画内のデータの占有状態が検査され、さらに必要に応じてサブブロックの区画または所属データに対応するキャッシュメモリ38のエントリが検査される(ST3)。これらの情報をもとに、このサブブロックをフラッシュメモリ35,36にそのまま書き込むか、もしくはキャッシュメモリ38に書き込むかが判定される(ST4)。

なおここでエントリの検査が実施された場合、データがヒットしたエントリ等の情報は制御回路34内のメモリまたはレジスタに保管しておく。

【0074】

<3> フラッシュメモリへの書き込み

サブブロックの書き込み先がフラッシュメモリに選定された場合は、まず所属データに対応するキャッシュメモリのエントリが調べられ、対応するエントリがあれば、整合をとるべくその処理が行われる。既に上記<2>の処理でエントリを検査済みの場合は、改めてキャッシュテーブル45を検索する必要は無い。

しかる後に、サブブロックのフラッシュメモリへの書き込みが、たとえば以下ように行われる。

まず、サブブロックの区画に相当するページデータが、フラッシュメモリ35,36からページバッファ32に読み出され、それがサブブロックのデータで更新される。なお、サブブロックがページ全域を含む場合、この読み出し工程は省略可能である。次に、ページバッファ32の内容がフラッシュメモリ35,36に書き戻される(ST5~ST7)。

【0075】

<4> キャッシュメモリへの書き込み

サブブロックの書き込み先がキャッシュメモリ38に選定された場合も、まず所属データに対応するキャッシュメモリ38のエントリが調べられる。既に上記<2>の処理でエントリを検査済みの場合は、改めてキャッシュ管理テーブル45を検索する必要は無い。

ヒットしているセクタはそのまま対応エントリに書き込みを実施する。ミスヒットのセクタには新規にエントリを追加し、そこに書き込みを実施する。この際既存エントリの解放が必要であれば、まず適当なエントリを解放し、そのデータをフラッシュメモリに書き戻す(ST8~ST10)。

【0076】

ところで、上記<3>の処理において、サブブロックをフラッシュメモリ側に書き込む際、キャッシュメモリ38との整合をとるため、データがヒットしたエントリの処理を行った。

本発明を採用した場合、サブブロックに属するデータの一部がキャッシュメモリ38にヒットしていても、それらをフラッシュメモリ側に書き込むという選択がなされ得る。その場合キャッシュ内のデータは最新では無くなるので、対処が必要である。

その具体的対応手段としては、該当エントリのデータをサブブロックのデータで更新しておくか、もしくはヒットしたエントリそのものを解放する。

【0077】

10

20

30

40

50

また、サブブロックをフラッシュメモリ35, 36に書き込む際には、前述の如く、同区間(同一ページ)に属し、サブブロックには含まれないセクタについてもキャッシュテーブル45を検索し、ヒットしている(すなわち書き込み先と同一ページに対応する)エントリがあれば、そのデータは同時にフラッシュメモリに書き込むのが望ましい。書き戻したエントリはクリーンとなり、必要に応じて解放することが可能である。

【0078】

書き戻しの際は、たとえば以下のようにページバッファ32上で各種データを合成し、完全な実ページを構成してから、フラッシュメモリ35, 36に書き込みを行う。

まず、ページバッファ32上にフラッシュメモリ35, 36からの対応ページを読み出す。次に、キャッシュメモリ内の同一区画に対応したエントリの有効データで、それを上書きする。さらにサブブロックのデータをそこに上書きし、最後にそれらをフラッシュメモリ35, 36に書き戻す。

【0079】

ところで、ページバッファ32からフラッシュメモリ35, 36へのページの書き戻し作業は、以下に示すような追記型書き換えを採用すると、さらに効率が良い。

【0080】

たとえば論理アドレス"0x5500"のページを書き戻す場合、アドレス変換テーブル44における対応物理ページは"0x00B0"のアドレス位置にある領域40, 41である。しかし、このページをそのまま書き換えるには、前述のごとき消去ブロック全体での有効ページの退避処理等が必要となる。

そこで、本実施形態においては、更新後のページデータは、適当な空き領域42, 43に書き込む。このページ領域の物理ページアドレスは"0x0280"であり、すなわち消去ブロック"0x02"内の"0x80"のページに相当する。

この処理の前まで、その領域はアドレス変換テーブル44のいずれの物理アドレスフィールドにも登録されていない、未使用の空きページであった。更新後のデータがこの領域に追記の形で書き込まれるとともに、アドレス変換テーブル44の論理アドレス"0x5500"に対応する物理アドレスフィールドには、領域42, 43の物理ページアドレス"0x0280"が登録される。

【0081】

すなわち、本実施形態においては、フラッシュメモリ35, 36への実ページデータの書き戻しは、更新されたページデータを消去済みの領域に追記し、元のページ領域を無効化することによって行われている。したがって、外部で書き込み先のページアドレスをランダムに設定しても、装置内部では空き領域にシリアルに書き込む等、フラッシュメモリの構成上最も効率の良い書き込みに変換できる。

このような手法を採用すれば、書き戻しの際に元ブロックのデータ退避等を行う必要が無く、フラッシュメモリ35, 36には見かけ上消去ブロックの制約を受けず、実ページ単位で高速にランダムにアクセスできる。

本実施形態のように実ページを元にした区画による分割でサブブロックを構成した場合、消去ブロックレベルでの書き込み効率の検査は困難である。したがって、消去ブロックに伴う制約を受けにくい本機構との組み併せで、記憶システムは最大の効率を得ることができる。

【0082】

なお、ここで言う実ページとはフラッシュメモリ35, 36に一括で書き込まれる連続データの基本単位を示すが、それが複数デバイスからのページグループで構成される場合、各デバイスへのデータ転送や書き込みのタイミング設定には制約は受けない。

本実施形態においては、16ビットIOのチップ2個を32ビットバスに接続することで完全な並列動作としたが、16ビットのバスを両チップで共有すれば、転送は個々に行なわざるを得ない。その場合、ページバッファ32から片側のメモリチップにデータを転送している間に、もう片側のメモリチップ内で書き込みをスタートしても良いし、両者への転送を順次行なった後に、書き込みを同時スタートしても良い。

10

20

30

40

50

【 0 0 8 3 】

すなわち、これらの複数デバイスを実質的に一つのフラッシュメモリデバイスとみなした場合、内部の動作タイミングに関わらず、その一括書き込みの基本単位が実ページとなる。

図3の例では、フラッシュメモリ35, 36、あるいはメモリバス33上に接続されたページバッファ32と転送制御回路37を併せた回路群MFLDをひとつのフラッシュメモリデバイスとみなすことができ、その一括書き込み単位である実ページはフラッシュメモリ35, 36における各々の書き込みページを併せたページグループである。

【 0 0 8 4 】

以上、フラッシュメモリの一括書き込み単位たるページを区画として書き込みデータ群をサブブロックに分割し、それぞれに書き込み先を選択する実施形態について説明した。

【 0 0 8 5 】

ところで現在、フラッシュメモリを用いた多くのファイルストレージには、消去ブロック単位でフラッシュメモリの書き換えを管理する従来手法が採用されている。

この場合、消去ブロックの部分書き換えには、前述の如き消去ブロック全体での有効データの退避が必要になるケースが頻出する。その際のオーバーヘッドは非常に大きいので、このようなケースではその発生頻度を下げるべく、消去ブロックを元に区画を設定し、それぞれに書き込み先を最適化した方が効率上がり、性能が高くなる可能性がある。

そのようなケースに本発明を応用した第2の実施形態について、以下に説明する。

【 0 0 8 6 】

図8は、本発明の第2の実施形態に係るファイル記憶装置を示す構成図である。

【 0 0 8 7 】

ファイル記憶装置30bは、インターフェース回路(I/F)31b、ページバッファ32b、メモリバス33b、制御回路34b、NAND型フラッシュメモリ35b, 36b、制御回路37b、キャッシュメモリ38b、および内部バス39bを含んで構成されている。また、制御回路34bは、およびキャッシュの管理テーブル45b、および消去ブロックの管理テーブル46bを有する。

本第2の実施形態に係るファイル記憶装置30bは、第1の実施例とほぼ同様の構成となっているが、ページ単位のアドレス変換は行っていない。

【 0 0 8 8 】

ファイル記憶装置30bの内部においては、32ビットのメモリバス33bに、16ビットの入出力を持つ2チップのNAND型フラッシュメモリ35b, 36bが並列に接続されている。2つのチップのNAND型フラッシュメモリ35b, 36bは読み出しや書き込みにおいて、同時並列にアクセスされる。

各々のフラッシュメモリ35, 36は書き込みや読み出しのアクセスをたとえば4kBのページ単位で行う。

したがって、実ページサイズとしては8kBが一括アクセスされることになる。

ページバッファ32bはアクセスされたページ領域のデータを一時記憶するページバッファである。

フラッシュメモリ35b, 36bとページバッファ32bとの間のデータの送受は、制御回路37bで制御されている。

さらに、制御回路37bは、必要に応じて転送データにECC符号化によるエラー補正を施したり、フラッシュメモリ内の欠陥ブロックの管理を実施する。両フラッシュメモリ35b, 36bはページバッファ32bを介して記憶装置の内部バス39bとの間でデータを入出力する。

すなわち、フラッシュメモリ35b, 36bあるいはメモリバス33b上に接続されたページバッファ32bと転送制御回路37bを併せた回路群FLMDは実質的に一つのフラッシュメモリデバイスを構成し、記憶装置30bの内部バス39bに接続されているとみなすことができる。

【 0 0 8 9 】

10

20

30

40

50

さらに内部バス 39b には書き込み用のキャッシュメモリ 38b とインターフェース回路 31b、および制御回路 34b が接続されている。

インターフェース回路 31b は A T A や P C I エクスプレス等の規格に従ってホストとの間で、データやコマンドの送受を行う。

制御回路 34b は、ファイル記憶装置 30b の内部においてページバッファ 32b、キャッシュメモリ 38b、およびインターフェース回路 31b の間のデータの送受を管理する。ここに内蔵された R A M には消去ブロックの管理テーブル 46b、およびキャッシュメモリの管理テーブル 45b が構築されている。

【0090】

制御回路 34b は、消去ブロック管理テーブル 46b を使用し、転送制御回路 37b にコマンドを送ることで、フラッシュメモリデバイスである記憶装置 30b の消去ブロック単位の書き換えも制御する。すなわち、ページバッファ 32b を介して元ブロックから予備ブロックへの有効ページの退避処理を実施し、消去ブロック管理テーブル 46b を更新して、対象論理ブロックアドレスに対応する物理ブロックアドレスの張替えを行い、さらに元ブロックの消去を実施する。

10

【0091】

図 9 は、第 2 の実施形態に係るキャッシュ管理テーブル 45b の構成例を示す図である。

キャッシュメモリ 38b へのエンタリは、実ページ、すなわち 16 セクタを単位として構成する。本例では、消去ブロック内のページアドレスをインデックスとしたダイレクトビットマップ方式が採用されている。すなわち、消去ブロック内の各ページアドレスに対応して、独立して一つずつのエンタリを持っている。

20

【0092】

たとえば、書き込みのために外部から入力されたセクタのアドレスが " 0 x 0 3 0 1 A " であれば、消去ブロックの論理アドレスを上位 2 ビットの " 0 x 0 3 "、ページアドレスをその下位の " 0 x 0 1 "、セクタアドレスは最下位の " 0 x A " と定義できる。

図 9 のテーブルでは、インデックスたるページアドレス " 0 x 0 1 " のエンタリには消去ブロック " 0 x 0 3 " からのページが格納されている。すなわちこのセクタ " 0 x 0 3 0 1 A " はキャッシュにヒットしている。

【0093】

上述の例では実ページサイズが 8 kB であり、256 個のエンタリを持つので、キャッシュサイズは 2 M バイトを必要とする。無論キャッシュ構成は第 1 の実施形態と同様の形態や、それ以外にもさまざまな形態を取ることができる。それに伴ってエンタリの単位等が変わるものの、本発明の基本的なアルゴリズムはキャッシュメモリの構成に関わらず適用できる。

30

【0094】

図 10 は、本第 2 の実施形態に係る消去ブロック管理テーブルの構成例を示す図である。

【0095】

図 10 に示すように、消去ブロック管理テーブル 46b には、各論理ブロックアドレス (L B A) に対応して物理ブロックアドレス (P B A)、および書き込みポインタ (W r i t e P o i n t e r) が格納されている。

40

【0096】

書き込みポインタは、各々に対応するブロックのどのページまで書き込みが完了したかを示している。

前述の如く、フラッシュメモリのデータは消去ブロック内のページアドレス " 0 x 0 0 " から " 0 x F F " に向けて順方向で書き込まれる必要がある。たとえば消去ブロック " 0 x 0 0 " の書き込みポインタは " 0 x F F " であり、一杯まで書き込まれていることが判る。したがって、この中のデータを更新するには、必ず消去が必要となる。

一方、消去ブロック " 0 x 0 1 " の書き込みポインタは " 0 x 8 0 " であり、 " 0 x 8

50

1"以降のページ領域にはそのまま書き込むことが可能である。ここにページ"0xA0"が書き込まれた場合、ポインタは"0xA0"にまで一気に移動する。消去ブロック"0x7E"の書き込みポインタはどこも指していない。すなわち規定値以外の数字が入っており、この場合消去された状態から一切書き込みが行なわれていないことを示している。

【0097】

本実施形態において、読み出しについての手順は第1の実施形態とほぼ同様であるが、アドレス変換はページ単位ではなくブロック単位で実施する。

【0098】

一方、たとえば"0x0011a"というセクタの更新が成された場合、これをフラッシュメモリデバイス30b側に書き込むには、以下の処理が必要である。

【0099】

消去ブロック(LBA)"0x00"は、既に末尾まで書き込まれており上書きができない。

したがって、制御回路34bは予備ブロック(PBA)"0x7F"に元ブロック(PBA)"0x00"のページデータを順次コピーしていく。すなわち各ページ領域のデータを先頭から順にページバッファ32bに一旦読み出し、さらに予備ブロックの同じ相対位置に書き込んでいく。この際、ページバッファ32bに更新すべきセクタを含むページ"0x11"が読み出されたら、その中のセクタ"0xa"のデータをバッファ上で更新する。さらに、末尾ページまでコピーが終わると、(LBA)"0x00"に対応する物理ブロックを(PBA)"0x7F"に張り替える。さらに元ブロック(PBA)"0x00"を消去して予備ブロックとする。

【0100】

すなわちこのようなケースでは、1セクタの更新に消去ブロック全体の有効データの退避と元ブロックの消去が必要となる。このような処理が発生すると、書き込みの能力は極度に劣化する。このような更新をキャッシュメモリで処理すれば、上記オーバーヘッドを緩和することができる。

ただし、第1の実施形態と同様に、キャッシュメモリ38bへの書き込む場合でも、ミスヒットし、かつエントリ追加のための空きも無ければ、一旦既存のエントリを解放し、空きをつくらねばならない。解放したエントリのデータは結局フラッシュメモリ35b, 36bに書き戻す必要があり、その作業はフラッシュメモリ38bへの書き込みと同じである。

【0101】

本実施形態においては、転送ブロックを消去ブロックごとの区画で分割し、各々の区画単位でアクセス手順を決定する。たとえば、セクタアドレス"0x0080C"~"0x0200C"の範囲のブロックデータが書き込まれるとすると、それらは以下の如く3つのサブブロックに分割される。

【0102】

区画1(サブブロック1): "0x0080C"~"0x00FFF"
 区画2(サブブロック2): "0x01000"~"0x01FFF"
 区画3(サブブロック3): "0x02000"~"0x0200C"

【0103】

このように分割した各々のサブブロックについて、最適な書き込み先を選択していく。図11に従って、以下にその判定基準の例を示す。

【0104】

ここで、キャッシュメモリのエントリが満杯で、かつ書き込みセクタが全てミスヒットするという、最悪のケースへの対応を考えてみる。

そこに書き込みを行うには、書き込みに要するだけのエントリを解放し、それをフラッシュメモリ35b, 36bに書き戻す必要が生じる。したがって、書き込むべきデータサイズが大きいほどその際の処理量が大きくなり、キャッシュメモリ使用の不利が増す。

10

20

30

40

50

【0105】

たとえば、サブブロック2は256ページより成り、消去ブロック"0×01"をまるごと含んでいる。これをキャッシュメモリ38bに格納すると、全エントリを消費する。この場合、最悪256個の互いに異なる消去ブロックに属するエントリを解放する必要があり、フラッシュメモリの256消去ブロックに対してページ書き込みを行う必要が生じる。

一方、このサブブロックをフラッシュメモリにそのまま書き込めば、単に1ページ分の書き込みで良い。従ってこのようなケースでは、キャッシュメモリ38bへの書き込みが逆効果となるリスクが極めて高い。

【0106】

一方、サブブロック3は消去ブロック"0×02"に属する13個のセクタよりなり、キャッシュメモリ38bの1エントリ分を消費するのみである。したがって、上記最悪ケースでもキャッシュから解放し、フラッシュメモリに書き戻すべきエントリは1個であり、同じ1消去ブロックへの書き込みですむ。キャッシュメモリ38bにヒットしていたり、エントリ追加のための空きがある場合は、フラッシュメモリにアクセスすることなく高速に処理できる。したがって、このサブブロックは殆どの場合キャッシュメモリ38b側に書き込む方が効率が良い。

【0107】

ここにおいて、本第2の実施形態においても、第1の実施形態と同様のアルゴリズムで、適切な書き込み先を選択できる。

たとえば、有効な書き込み手法の一つは、消去ブロック単位で区画分けされた各サブブロックごとに、そのデータの区画内の占有状態を検査し、それに応じて書き込み先をフラッシュメモリかキャッシュメモリの何れかに選択することである。

すなわち、各サブブロックにおいて書き込むべきセクタ数、あるいはキャッシュメモリ38bに書き込む際に消費されるエントリ数を基準として、それらが既定数より多ければフラッシュメモリ35b, 36bに、既定数以下ならキャッシュメモリ38bに書き込みを実施する。

各サブブロックにおけるセクタ数や消費されるエントリ数は、アドレス情報から瞬時に取得できる。従ってこのような判定アルゴリズムを用いてデータの書き込み先を適時選択すれば、簡便に書き込み効率及び性能を向上させることができる。

【0108】

あるいは、キャッシュメモリ38bのエントリを検査して、データのヒットやエントリ追加のための空き状態を調べ、各サブブロックをキャッシュメモリ38bに書き込む際、新規に追加すべきエントリ数または解放すべきエントリ数を導出する。しかる後にそれを基準に書き込み先を選定する。すなわち既定数より多くのエントリを追加または解放する必要があればフラッシュメモリ35b, 36bに、それ以下であればキャッシュメモリ38bに書き込みを実施する。

【0109】

あるいは、フラッシュメモリ35b, 36bへの書き込み時にキャッシュメモリ38bから同一消去ブロックに書き戻し可能なデータを検査し、判定に使用する。

たとえば上記サブブロック3を、仮にフラッシュメモリにそのまま書き込もうとした場合、キャッシュテーブル45bから同じ区画3の消去ブロックに対応するエントリが検索できる。

すなわち、サブブロック3は、消去ブロック"0×02"に属するが、同じ消去ブロックからのエントリがキャッシュメモリ内に存在すれば、その消去ブロックへの書き込みで消去を必要とした場合には、該当ページの退避処理の際にそのデータをフラッシュメモリに書き戻すことができる。そのようなエントリはクリアとなり、必要に応じて解放することが可能である。すなわち、たとえばフラッシュメモリ35b, 36bへ同時書き込みが可能な総ページ数を基準とし、それが規定値以上ならフラッシュメモリにそのまま書き込む。

10

20

30

40

50

【0110】

ところで、本実施形態の如く、消去ブロックが基本的な書き換えの管理単位である場合、このブロック内の同じ場所に同じページ数を書き込む場合でも、図10の消去ブロックの管理テーブル46bに関して前述したように、書き込み先ブロックの空き状況次第で必要とされる作業量は大きく異なる。

したがって、消去ブロック管理テーブル46bから該当ブロックの空き状態を検査して、たとえば元の消去ブロックにそのまま書き込める場合は、書き込み先をフラッシュメモリ35b, 36bに選択し、それ以外の場合には前述の選択基準のいずれかに従うことで、さらに効率の良い書き込みが可能である。

【0111】

上述の如く、選択アルゴリズムには様々なバリエーションが存在し、さらにそれらは場合に応じて互いに組み合わせることも可能である。その基本的な手順は以下の通りである。

1. 入力コマンドとアドレス情報を取得する。
2. 消去ブロックによる区画をもとに、ブロックデータをサブブロックに分割し、各々のブロックについて順次書き込み処理を実施する。
3. その際、サブブロック内のデータの占有状態、さらにはキャッシュテーブルの検査、または書き込み先消去ブロックの空き領域の検査を行い、その結果に応じて書き込み先を選択する。

【0112】

上記3.の判定については、各々のサブブロックの書き込みに際して、
・フラッシュメモリデバイスにおけるブロック消去の発生回数を最小限に抑える
・ブロック消去が発生する場合には、該当ブロックの一連の処理の際に、なるべく多くの更新データを同じブロック内に書き込む
という視点に立ち、さらに判定に要する時間や処理の煩雑さも鑑みてその手順が設定されるのが望ましい。

【0113】

なお、サブブロックをフラッシュメモリ側に書き込む際は、第1の実施形態と同様に、キャッシュメモリとの整合を取るため、データがヒットしたエントリの処理を行う。その具体的対応手段としては、対応エントリ内のデータをサブブロックのデータで更新しておくか、もしくは対応エントリそのものを解放する。

【0114】

また、サブブロックをフラッシュメモリに書き込む際に少なくともブロック消去が必要であれば、前述の如く、同区間(同一消去ブロック)に属し、サブブロックに含まれないページについてもキャッシュテーブル45bを検索し、ヒットしている(すなわち書き込み先と同一の消去ブロックに属する)エントリがあれば、そのデータを同時にフラッシュメモリに書き込むのが望ましい。書き戻したエントリはクリアとなり、必要に応じて解放することが可能である。

【0115】

ところで、ここまでの説明は半導体記憶装置を対象に、記憶装置自体がその内部でデータの書き込み先をキャッシュメモリとフラッシュメモリに振り分ける形態を説明した。しかし本発明の実施形態の書き込み手順や選択アルゴリズムに関しては、それ以外の形態でも応用が可能である。

【0116】

たとえばファイルストレージとしてフラッシュメモリデバイスが組み込まれたコンピュータシステムでは、CPUがシステムメモリ内にフラッシュメモリデバイス用のキャッシュメモリやアドレス変換テーブルを構築し、ドライバソフトを用いてフラッシュメモリデバイスへの書き込み管理を実施することも多い。

【0117】

図12は、そのような本実施形態に係るコンピュータシステムの構成例を示す図である

10

20

30

40

50

。

【0118】

図12のコンピュータシステムCOMSYSは、ファイル記憶装置としてのフラッシュメモリデバイス30cおよび処理装置50を主構成要素として有している。

【0119】

フラッシュメモリデバイス30cは、ページバッファ32cを含むブリッジ回路31c52c、およびNAND型フラッシュメモリ35c、36cを含む。

また、処理装置50は、CPU(処理ユニット)51c、RAMからなるシステムメモリ、およびシステムバス53cを有する。

【0120】

CPU51cは32ビットのシステムバス53cを介してシステムメモリ52cであるRAMと接続されている。

さらに、このシステムバス53cにはフラッシュメモリデバイス30cのブリッジ回路31cが接続されている。このブリッジ回路31cに繋がる32ビットのデータバス33cには、16ビットの入出力を持つ2チップのNAND型フラッシュメモリ35c、36cが並列接続されている。

2つのチップのNAND型フラッシュメモリ35c、36cは読み出しや書き込みにおいて、同時並列にアクセスされる。

ページバッファ32cはアクセスされたページ領域を一時記憶するページバッファであり、ブリッジ回路31cに内蔵されている。

ブリッジ回路31cはCPU51cから各種コマンドを受け取り、フラッシュメモリ35c、36cとCPU51cまたはシステムメモリ52cとの間のデータのやり取りを、ページバッファ32cを用いて媒介する。

ブリッジ回路31cが受け取るコマンドは、たとえばフラッシュメモリ35c、36cの所定のページへのアクセスの他、同フラッシュメモリの所定ブロックの消去、及び所定ページの指定アドレスへのコピー、フラッシュメモリのリセット等である。すなわちCPU51cから見れば、ブリッジ回路31c以降は一つのフラッシュメモリデバイス30cとみなすことができる。

【0121】

一方、システムメモリ52c内には、上記フラッシュ記憶システム(ファイル記憶装置)を制御するためのドライバ54cが常駐している。このドライバ54cは、オペレーティングシステム(OS)やアプリケーションからの記憶装置へのアクセスを受け、同じメモリ中に構成されたアドレス変換テーブル55c(第1および第2の実施形態のアドレス変換テーブルに38cに相当する)を参照してアクセス時のページアドレスを変換する。

さらにシステムメモリ52c内にはフラッシュメモリデバイスのキャッシュメモリ領域56cおよびキャッシュテーブル57cが構築されている。

【0122】

このようなケースでは、アプリケーションから発行されたファイルストレージへの書き込み要求をドライバ54cが受けて、そのアドレス情報からデータブロックの書き込み先を、キャッシュメモリ領域56cかフラッシュメモリデバイス30cに選択すれば良い。その際には前述の実施形態と全く同様の手順で、書き込みの効率を向上させることが可能である。

すなわち、フラッシュメモリデバイス30cの書き込みページまたは消去ブロックをもとに区画を設けてデータをサブブロックに分割し、各々のサブブロックごとに既定の基準に従って書き込み先を選択する。

一括書き込み単位であるページを区画とした場合には、選択に際しては各サブブロックのデータの占有状況、さらにはキャッシュテーブル57cを検査し、その結果をもとに選択を実施すれば良い。

一括消去単位である消去ブロックを区画とした場合には、選択に際しては各サブブロックのデータの占有状況、キャッシュテーブル57c、あるいは書き込み先のフラッシュデ

10

20

30

40

50

バイス 30c の消去ブロック空き状況を検査し、その結果をもとに選択を実施すれば良い。

【0123】

なお、本発明の実施形態に係る半導体記憶装置やコンピュータシステムにおけるキャッシュメモリには S R A M、D R A M 等揮発性 R A M を使用しても良いが、たとえば半導体記憶装置がモバイルメディアの場合、急な引き抜き等の電源瞬断に耐性が必要となる。また組み込み型であっても、コンピュータ機器そのものの電源瞬断に備える必要がある。

【0124】

したがって、キャッシュメモリには強誘電体メモリ等の不揮発性 R A M を使用し、さらにはキャッシュテーブルやアドレス変換テーブルもそのような不揮発性 R A M 内に格納することが望ましい。

10

【0125】

図 13 は、キャッシュメモリに強誘電体メモリ等の不揮発性 R A M を使用した構成を採用したファイル記憶装置の構成例を示す図である。

【0126】

ファイル記憶装置 30d は、インターフェース回路 (I / F) 31d、ページバッファ 32d、メモリバス 33d、制御回路 34d、N A N D 型フラッシュメモリ 35d、36d、制御回路 37d、内部バス 39d、および強誘電体メモリ 47d を含んで構成されている。

【0127】

ファイル記憶装置 30d の内部においては、32ビットのメモリバス 33d に、16ビットの入出力を持つ 2 チップの N A N D 型フラッシュメモリ 35d、36d が並列接続されている。

20

2 つのチップの N A N D 型フラッシュメモリ 35d、36d は読み出しや書き込みにおいて、同時並列にアクセスされる。

各々のフラッシュメモリ 35d、36d は書き込みや読み出しのアクセスをたとえば 4 kB のページ単位で行う。

したがって、実ページサイズとしては 8 kB が一括アクセスされることになる。

ページバッファ 32d はアクセスされたページ領域のデータを一時記憶するページバッファである。

30

フラッシュメモリ 35d、36d とページバッファ 32d との間のデータの送受は、制御回路 37d で制御されている。

さらに制御回路 37d は、必要に応じて転送データに E C C 符号化によるエラー補正を施したり、フラッシュメモリ内の欠陥ブロックの管理を実施する。両フラッシュメモリはページバッファ 32d を介して記憶装置の内部バス 39d との間でデータを入出力する。

すなわち、フラッシュメモリ 35d、36d あるいはメモリバス 33d 上に接続されたページバッファ 32d と転送制御回路 37d を併せた回路群 F L M D は実質的に一つのフラッシュメモリデバイスを構成し、記憶装置の内部バス 39d に接続されているとみなすことができる。

【0128】

40

さらに、内部バス 39d には強誘電体メモリ 47d とインターフェース回路 31d、および制御回路 34d が接続されている。インターフェース回路 31d は A T A や P C I エクスプレス等の規格に従ってホストとの間で、データやコマンドの送受を行う。

強誘電体メモリ 47d 内には、実ページサイズの $1/N$ (N は 2 以上の整数) をエントリとしたキャッシュメモリ 38d、ページ単位の仮想アドレスを管理するアドレス変換テーブル 44d、およびキャッシュメモリの管理テーブル 45d が構築されている。

制御回路 34d は、記憶装置 30d の内部においてページバッファ 32d、キャッシュメモリ 38d、およびインターフェース回路 31d の間のデータの送受を管理する。

【0129】

または、図 13 におけるキャッシュメモリ 38d と制御回路 34d は一つの I C チップ

50

に統合して同時に製造しても良い。

さらに望ましくはインターフェース回路31d、ページバッファ32dおよび制御回路37d)も同一のICチップに統合し、ファイル記憶装置を左記ICチップとフラッシュメモリのSIP(System In Package)で構成しても良い。

【0130】

なお、不揮発性RAMとしては、強誘電体膜を用いたFeRAM(Ferroelectric RAM)、強磁性体を用いたMRAM(Magnetic RAM)や、相変化材料を用いたOUM(Ovonic Unified Memory)、RRAM等が提案されている。

FeRAMは強誘電体キャパシタの分極方向の違いでデータを記憶する半導体メモリである。たとえばUS4873664においてS. Sheffieldらがその一形態を提案している。

10

【0131】

MRAMは強磁性膜のスピン方向の違いでデータを記憶する半導体メモリである。たとえばISSCC2000の論文ダイジェストの128ページに、R. Scheuerleinらが論文を掲載している。

OUMはたとえばカルコゲナイド膜の相転移でデータを記憶する半導体メモリである。たとえばIEDM2001の論文ダイジェストの803ページに、S. Laiらが論文を掲載している。

RRAMは磁気抵抗効果材料の抵抗ヒステリシスでデータを記憶する半導体メモリである。たとえばIEDM2002の論文ダイジェストの7.5にはW. W. Zhuangらが論文を掲載している。

20

それらの不揮発性メモリは、いずれもセルレベルのアクセス速度や書き換え回数において、フラッシュメモリより数桁性能が高い。

【図面の簡単な説明】

【0132】

【図1】NAND型フラッシュメモリの内部構成例を示す図である。

【図2】記憶装置を構成するフラッシュメモリの概念図を示す図である。

【図3】本発明の第1の実施形態に係るファイル記憶装置を示す構成図である。

【図4】本発明の第1の実施形態に係るキャッシュメモリ管理テーブルの構成例を示す図である。

30

【図5】アドレス変換テーブルの構成例を示す図である。

【図6】第1の実施形態に係る区画による分割と判定基準例を示す図である。

【図7】第1の実施形態に係るブロックデータ書き込み手順を示すフローチャートである。

【図8】本発明の第2の実施形態に係るファイル記憶装置を示す構成図である。

【図9】本発明の第2の実施形態に係るキャッシュ管理テーブルの構成例を示す図である。

【図10】本発明の第2の実施形態に係る消去管理テーブルの構成例を示す図である。

【図11】第2の実施形態に係る区画による分割と判定基準例を示す図である。

40

【図12】本発明の実施形態に係るコンピュータシステムの構成例を示す図である。

【図13】上述のキャッシュメモリに強誘電体メモリ等の不揮発性RAMを使用した構成を採用したファイル記憶装置の構成例を示す図である。

【符号の説明】

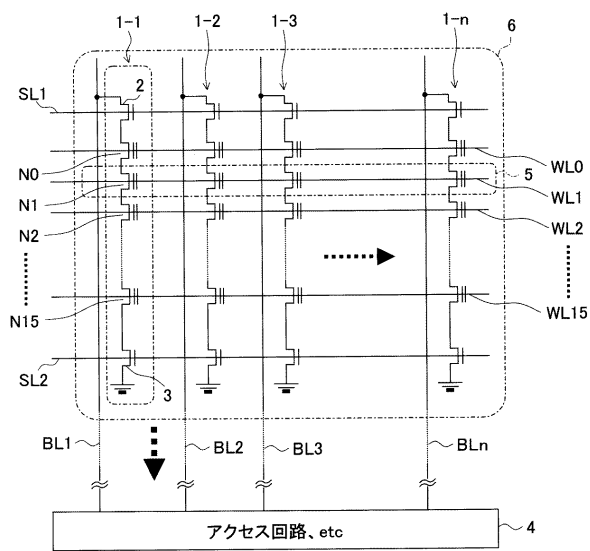
【0133】

30, 30b, 30c, 30d・・・ファイル記憶装置、31, 31b, 31d・・・インターフェース回路(I/F)、31c・・・ブリッジ回路、32, 32b, 32c, 32d・・・ページバッファ、33, 33b, 33c, 33d・・・メモリバス、34, 34b, 34d・・・制御回路、35, 36, 35b, 36b, 35c, 36c, 35d, 36d・・・NAND型フラッシュメモリ、37, 37b, 37d・・・制御回路、3

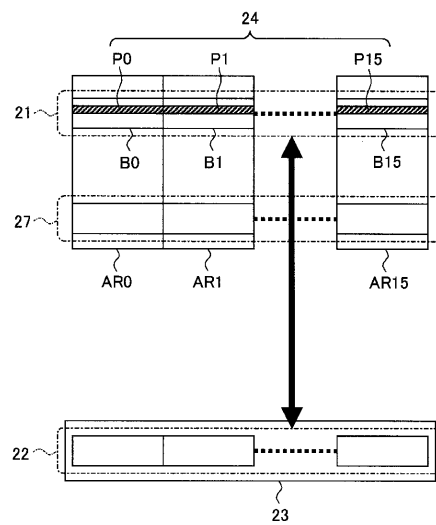
50

8, 38b, 38d, 56c...キャッシュメモリ、39, 39b...内部バス、44, 44d...アドレス変換テーブル、45, 45b, 45d...キャッシュの管理テーブル(キャッシュテーブル)、47d...強誘電体メモリ、COMSYS...コンピュータシステム、50...処理装置、51...CPU、52...システムメモリ(RAM)、53...システムバス。

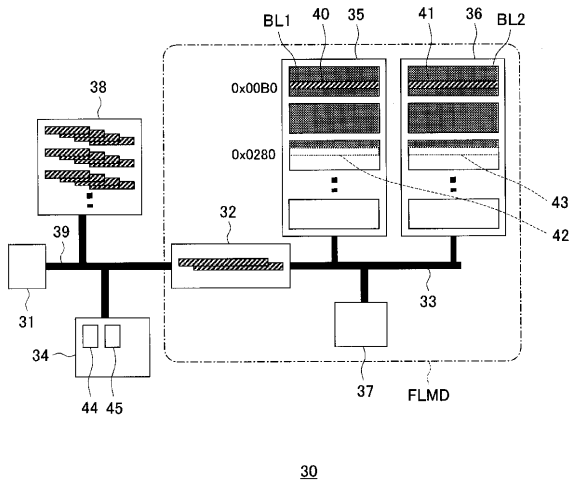
【図1】



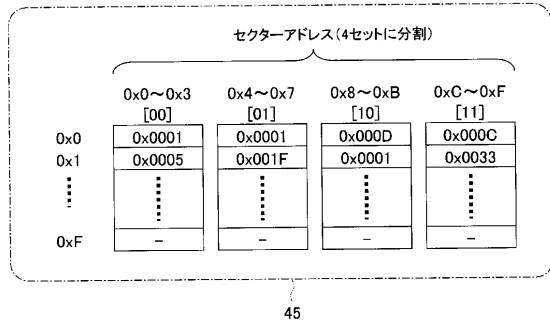
【図2】



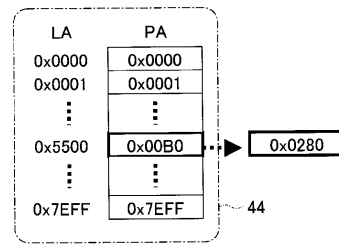
【図3】



【図4】



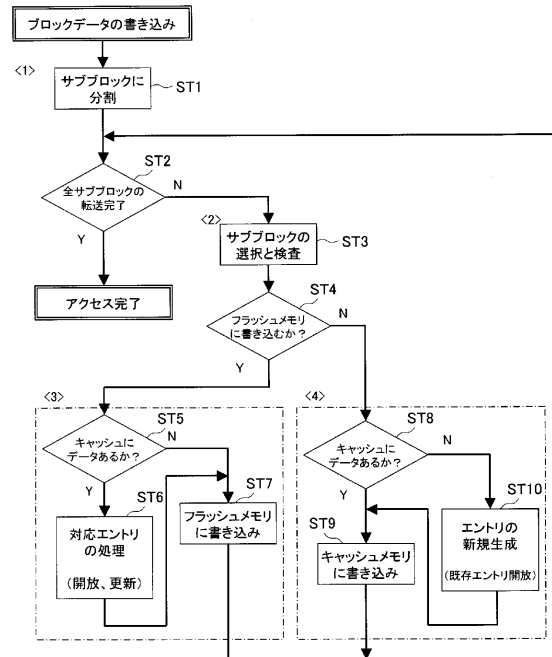
【図5】



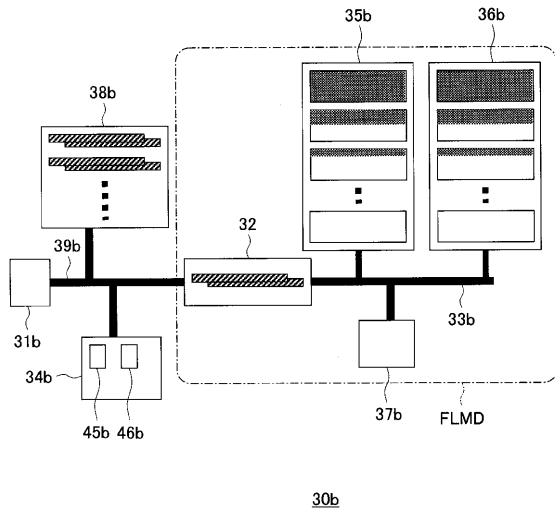
【図6】

	アドレス領域	セクター数	エントリ消費数 (a)	エントリヒット数 (b)	エントリ追加必要数 (a-b)	エントリ開放必要数
サブブロック1	0x0000C ~ 0x0000F	4	1	1	0	0
サブブロック2	0x00010 ~ 0x0001F	16	4	3	1	1
サブブロック3	0x00020	1	1	0	1	1

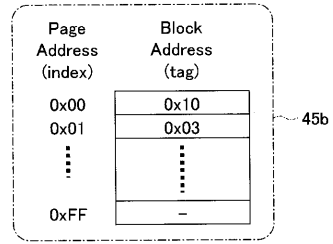
【図7】



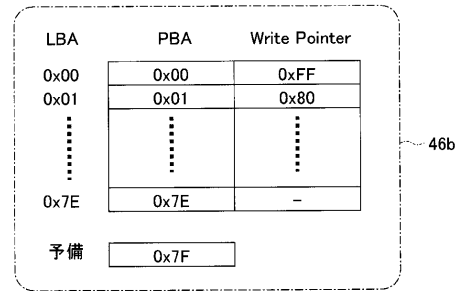
【図8】



【図9】



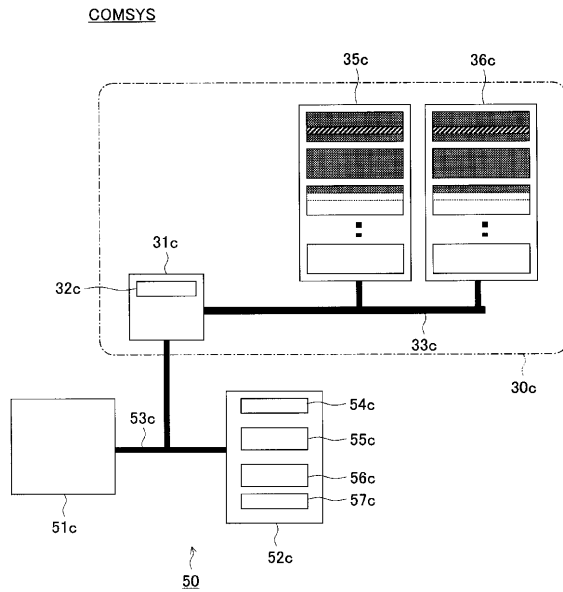
【図10】



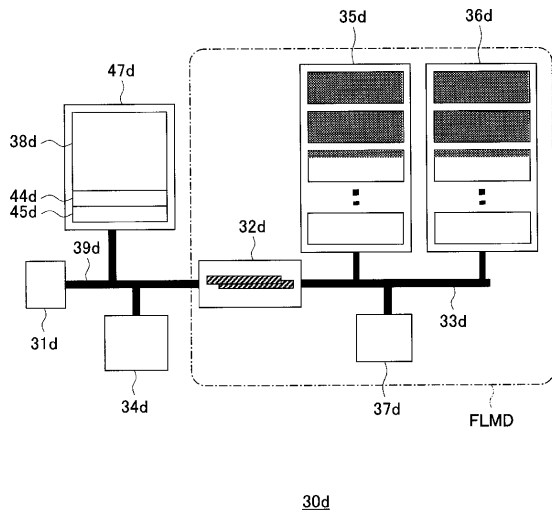
【図11】

	アドレス領域	セクタ数	エントリ消費数 (a)	エントリヒット数 (b)	エントリ追加必要数 (a-b)	エントリ開放必要数
サブブロック1	0x0080C ~ 0x00FFF	2036	128	120	8	8
サブブロック2	0x01000 ~ 0x01FFF	4096	256	0	256	256
サブブロック3	0x02000 ~ 0x0200C	13	1	0	1	1

【図12】



【 図 13 】



フロントページの続き

(51)Int.Cl.

F I

G 0 6 F 12/08 5 1 5 Z

(56)参考文献 国際公開第2005/088456(WO, A1)

特開2003-058420(JP, A)

(58)調査した分野(Int.Cl., DB名)

G 0 6 F 1 2 / 0 8

G 1 1 C 1 6 / 0 2

G 0 6 F 3 / 0 6