(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号

特許第5635200号
(P5635200)

(45)発行日　平成26年12月3日(2014.12.3)　(24)登録日　平成26年10月24日(2014.10.24)

(51)Int.Cl.		F I		
G06F 13/16	(2006.01)	G06F	13/16	510D
G06F 12/00	(2006.01)	G06F	12/00	597U
G06F 12/06	(2006.01)	G06F	12/06	515H
G06F 12/16	(2006.01)	G06F	12/16	320L
G06F 3/06	(2006.01)	G06F	3/06	301A

請求項の数 9　(全 26 頁)　最終頁に続く

(21)出願番号　特願2013-535800 (P2013-535800)
(86)(22)出願日　平成23年9月30日 (2011.9.30)
(86)国際出願番号　PCT/JP2011/072653
(87)国際公開番号　WO2013/046463
(87)国際公開日　平成25年4月4日 (2013.4.4)
審査請求日　平成25年10月16日 (2013.10.16)

(73)特許権者　000005108
株式会社日立製作所
東京都千代田区丸の内一丁目６番６号
(74)代理人　110000062
特許業務法人第一国際特許事務所
(72)発明者　石川　篤
日本国東京都千代田区丸の内一丁目６番６号　株式会社日立製作所内
(72)発明者　園田　浩二
日本国東京都千代田区丸の内一丁目６番６号　株式会社日立製作所内
(72)発明者　上原　剛
日本国東京都千代田区丸の内一丁目６番６号　株式会社日立製作所内

最終頁に続く

(54)【発明の名称】不揮発半導体記憶システム

(57)【特許請求の範囲】
【請求項１】
フラッシュメモリコントローラと、
前記フラッシュメモリコントローラに接続された、第１データバスと、
前記フラッシュメモリコントローラに接続された、第２データバスと、
複数の第３データバスと、
複数の第４データバスと、
それぞれが前記複数の第３データバスの１つに接続された、複数の第１フラッシュメモリチップと、
それぞれが前記複数の第４データバスの１つに接続された、複数の第２フラッシュメモリチップと、
前記第１データバスに接続され、前記複数の第３データバスのいずれか１つに選択的に接続される、第１スイッチと、
前記第２データバスに接続され、前記複数の第４データバスのいずれか１つに選択的に接続される、第２スイッチと、
前記フラッシュメモリコントローラと前記第１スイッチとに接続された切替信号線と、を備え、
前記第１スイッチは、前記切替信号線を用いて切替信号を送信するため、前記第２スイッチに直接接続されるよう構成され、これにより前記フラッシュメモリコントローラは、前記切替信号線を用いて前記第１スイッチ及び前記第２スイッチの両方に１つの切替信号

を供給することによって、前記複数の第１フラッシュメモリチップの１つ及び前記複数の第２フラッシュメモリチップの１つに並行してアクセスする、
フラッシュメモリパッケージ。
【請求項２】
前記フラッシュメモリコントローラは、前記第１データバスを前記複数の第３データバスの１つに接続するための前記第１スイッチと、前記第２データバスを前記複数の第４データバスの１つに接続するための前記第２スイッチとを制御するよう構成される、
請求項１記載のフラッシュメモリパッケージ。
【請求項３】
前記フラッシュメモリコントローラは、
ライトデータを複数のデータ要素に分割し、
前記複数のデータ要素の第１データ要素を前記複数の第１フラッシュメモリチップの１つへ、及び前記複数のデータ要素の第２データ要素を前記複数の第２フラッシュメモリチップの１つへ、それぞれ並行して送信するよう構成される、
請求項２記載のフラッシュメモリパッケージ。
【請求項４】
前記フラッシュメモリコントローラに接続された複数のチップイネーブル信号線を更に含み、
前記複数のチップイネーブル信号線のそれぞれは、複数の第１フラッシュメモリチップの１つと複数の第２フラッシュメモリチップの１つとに接続される、
請求項３記載のフラッシュメモリパッケージ。
【請求項５】
前記複数のチップイネーブル信号線は、第１及び第２のチップイネーブル信号線を含み、
（Ａ）前記フラッシュメモリコントローラは、前記第１のチップイネーブル信号線を介してチップイネーブル信号を送信し、且つ、前記異なるスイッチのそれぞれを、前記第１のチップイネーブル信号線を介してチップイネーブル信号を受ける前記フラッシュメモリチップが接続先となるよう制御し、
（Ｂ）前記フラッシュメモリコントローラは、前記複数のデータ要素のうちの連続する２以上のデータ要素を、前記第１のチップイネーブル信号線を共通とする２以上のフラッシュメモリチップに並行して送信し、
（Ｃ）前記フラッシュメモリコントローラは、前記（Ｂ）でデータ要素を受けた前記フラッシュメモリチップに接続されているスイッチの接続先を、前記第２のチップイネーブル信号線を介してチップイネーブル信号を受けることになる前記フラッシュメモリチップに切り替え、
（Ｄ）前記フラッシュメモリコントローラは、前記第２のチップイネーブル信号線を介して前記チップイネーブル信号を送信し、
（Ｅ）前記フラッシュメモリコントローラは、前記複数のデータ要素のうちの残りのうちの連続する２以上のデータ要素を、前記第２のチップイネーブル信号線を共通とする２以上の前記フラッシュメモリチップに並行して送信する、
請求項４記載のフラッシュメモリパッケージ。
【請求項６】
前記フラッシュメモリコントローラが、前記チップイネーブル信号線毎に、ブロックに対して行われた消去処理の回数の合計を管理し、
前記複数のデータ要素の転送先として選択される前記フラッシュメモリチップは、前記チップイネーブル信号を共通にする複数の前記フラッシュメモリチップのうち、前記消去処理の回数が最も少ないフラッシュメモリチップであり、
前記フラッシュメモリチップにおいてデータ要素の書込み先となる前記ブロックは、消去回数が最も少ないブロックである、
請求項５記載のフラッシュメモリパッケージ。

【請求項７】
前記各フラッシュメモリチップは、複数のブロックを有しており、
前記フラッシュメモリコントローラは、リクラメーション処理を行うように構成されており、
前記リクラメーション処理において、前記フラッシュメモリコントローラは、
（ｒ１）前記複数のブロックから移動元のブロックを選択し、
（ｒ２）前記複数のブロックから移動先のブロックを決定し、
（ｒ３）前記移動先ブロックに、前記移動元ブロックのページ内の有効データを移動し、
（ｒ４）前記移動元ブロックからデータを消去する消去処理を行い、
前記（ｒ２）において、前記移動先ブロックは、前記移動元ブロックを有するフラッシュメモリチップとデータ通信可能な媒体インタフェースとデータ通信可能な前記フラッシュメモリチップの中から決定される、
請求項６記載のフラッシュメモリパッケージ。
【請求項８】
前記媒体インタフェースには、２以上のデータバスを介して２以上の前記スイッチが接続され、
前記移動元ブロックと前記移動先ブロックは、同一の前記データバスを介してデータ通信可能な１以上の前記フラッシュメモリチップに存在する、
請求項７記載のフラッシュメモリパッケージ。
【請求項９】
前記（ｒ２）で、前記フラッシュメモリコントローラは、前記移動元ブロックを有するフラッシュメモリチップとデータ通信するためのデータバスと同じデータバスを介してデータ通信可能な１以上のフラッシュメモリチップに空きのブロックが所定数以上あるか否か判断し、
その判断の結果が肯定的であれば、前記１以上のフラッシュメモリチップからいずれかの空きのブロックを移動先ブロックとして決定し、
その判断の結果が否定的であれば、前記移動元ブロックを有するフラッシュメモリチップとデータ通信可能な前記媒体インタフェースと異なる媒体インタフェースとデータ通信可能な２以上の前記フラッシュメモリチップの中から空きのブロックを移動先ブロックとして決定する、
請求項７記載のフラッシュメモリパッケージ。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、複数の不揮発半導体記憶媒体を有する記憶システムに関する。
【背景技術】
【０００２】
ストレージシステムは、一般に、複数の記憶デバイスで構成されたＲＡＩＤ（Ｒｅｄｕｎｄａｎｔ Ａｒｒａｙ ｏｆ Ｉｎｄｅｐｅｎｄｅｎｔ Ｄｉｓｋｓ）グループに基づいて作成された論理ボリュームを、上位装置（例えばホストコンピュータ）へ提供する。近年、記憶デバイスとしては、ＨＤＤ（Ｈａｒｄ Ｄｉｓｋ Ｄｒｉｖｅ）に加えて又は代えて、複数の不揮発チップを有する不揮発半導体記憶装置が採用されている。不揮発半導体記憶デバイスとして、例えば、複数のフラッシュメモリチップ（以下、ＦＭチップ）を有するフラッシュメモリが採用される（例えば特許文献１）。
【先行技術文献】
【特許文献】
【０００３】
【特許文献１】特開２０１０－３１６１号公報
【発明の概要】
【発明が解決しようとする課題】

【０００４】
不揮発半導体記憶デバイスにおいても、記憶容量の増加が要請されている。記憶容量を増加させるためには、搭載する不揮発チップを増加させることが必要であるが、より多くの不揮発チップを搭載するようにすると、不揮発チップに接続されるインタフェースデバイスを含んだ回路（以下、制御回路、例えばＡＳＩＣ（Ａｐｐｌｉｃａｔｉｏｎ Ｓｐｅｃｉｆｉｃ Ｉｎｔｅｇｒａｔｅｄ Ｃｉｒｃｕｉｔ））には、複数の不揮発チップに各種信号を送信するためのバスを接続するためのピンを用意しておく必要がある。
【０００５】
このため、不揮発半導体記憶デバイスの記憶容量を増加させようとすると、制御回路のサイズが大きくなってしまう。一方、制御回路のサイズを小さくすると、接続可能な不揮発チップの数が少なくなり、記憶容量を増加させることが困難である。また、多数の不揮発チップを接続すると効率よくデータ転送をすることが困難である。
【０００６】
本発明の目的は、制御回路のサイズを抑えつつ、搭載できる不揮発チップの数を増加させ、効率よくデータ転送することのできる技術を提供することにある。
【課題を解決するための手段】
【０００７】
不揮発半導体記憶システムが、（ａ）複数の不揮発半導体記憶媒体と、（ｂ）複数の不揮発半導体記憶媒体に接続された媒体インタフェース群（１以上のインタフェースデバイス）を有する制御回路と、（ｃ）複数のスイッチとを有する。媒体インタフェース群と複数のスイッチとがデータバスを介して接続され、各スイッチと、各２以上の不揮発チップとが、データバスを介して接続される。スイッチは、媒体インタフェース群に接続されているデータバスとそのスイッチに接続されている複数の不揮発チップのいずれかに接続されているデータバスとの接続を切り替えるよう構成されている。制御回路は、ライト対象のデータを複数のデータ要素に分割し、複数のスイッチを制御することにより接続を切り替えて、複数のデータ要素を複数の不揮発チップに分散して送信する。
【０００８】
不揮発半導体記憶システムは、上記（ａ）、（ｂ）及び（ｃ）の要素を有する記憶媒体グループ（例えば、後述のフラッシュメモリＰＫＧ１０）であっても良いし、そのような記憶媒体グループを複数個有する記憶装置（例えば後述のフラッシュメモリデバイス４００）であっても良いし、そのような記憶装置を複数個とそれらの記憶装置に接続されたコントローラとを有するシステム（例えば、後述のストレージシステム１）であっても良い。
【０００９】
媒体インタフェース群は、複数の媒体インタフェースを有し、媒体インタフェースが、Ｎ個の不揮発半導体記憶媒体（Ｎは１以上の整数）毎に存在しても良い。１つの不揮発半導体記憶媒体を構成する複数の不揮発チップが、例えば１つのＤＩＭＭに存在して良い。ＤＩＭＭが、それら複数の不揮発チップに接続される１以上のスイッチを有して良い。
【図面の簡単な説明】
【００１０】
【図１】図１は、実施形態に係る計算機システムの構成例を示す。
【図２】図２は、実施形態に係るフラッシュメモリデバイスの構成例を示す。
【図３】図３は、実施形態に係るフラッシュメモリパッケージの構成例を示す。
【図４】図４は、実施形態に係るフラッシュメモリパッケージの一部の詳細な構成例を示す。
【図５】図５は、実施形態に係る書込み処理の第１の例を説明する図である。
【図６】図６は、実施形態に係る書込み処理の第２の例を説明する図である。
【図７】図７は、実施形態に係るフラッシュメモリデバイスの正面上方からの斜視図の一例である。
【図８】図８は、実施形態に係るフラッシュメモリＰＫＧの上面側からの斜視図の一例で

ある。
【図９】図９は、実施形態に係るフラッシュメモリＰＫＧの下面側からの斜視図の一例である。
【図１０】図１０は、実施形態に係るＤＩＭＭの概略構成の一例を示す。
【図１１】図１１は、実施形態に係るチップリード処理のフローチャートの一例である。
【図１２】図１２は、実施形態に係るチップライト処理のフローチャートの一例である。
【図１３】図１３は、実施形態に係るチップ多重ライト処理のフローチャートの一例である。
【図１４】図１４は、実施形態の一変形例に係るフラッシュメモリパッケージの一部の詳細な構成例を示す。
【図１５】図１５は、実施形態に係る論理アドレス層と物理層との関係の一例を示す。
【図１６】図１６は、実施形態に係るユーザアドレス空間とフラッシュストレージ論理空間との関係の一例を示す。
【図１７】図１７は、実施形態に係る論理物理変換情報の構成例を示す。
【図１８】図１８は、実施形態に係るリクラメーション処理のフローチャートの一例を示す。
【図１９】図１９は、実施形態に係る消去管理情報の構成例を示す。
【発明を実施するための形態】
【００１１】
以下、一実施形態を、図面を参照して説明する。
【００１２】
なお、以下の説明では、要素（例えば、ページ、フラッシュメモリチップ（ＦＭチップ）、スイッチ（ＳＷ））を特定するために番号を含む識別情報が使用されるが、識別情報として、番号を含まない情報が使用されても良い。
【００１３】
また、以下の説明では、同種の要素を区別して説明する場合、要素名と参照符号との組合せに代えて、要素名と識別情報との組合せが使用されることがある。例えば、識別情報（識別番号）「０」のスイッチを、「スイッチ＃０」と表記することがある。
【００１４】
また、以下の説明では、インタフェースデバイスを「Ｉ／Ｆ」と略記することがある。
【００１５】
また、以下の説明では、不揮発半導体記憶媒体は、フラッシュメモリ（ＦＭ）であるとする。そのフラッシュメモリは、ブロック単位で消去が行われ、ページ単位でアクセスが行われる種類のフラッシュメモリ、典型的にはＮＡＮＤ型のフラッシュメモリであるとする。しかし、フラッシュメモリは、ＮＡＮＤ型に代えて他種のフラッシュメモリ（例えばＮＯＲ型）でも良い。また、フラッシュメモリに代えて、他種の不揮発半導体記憶媒体、例えば相変化メモリが採用されても良い。
【００１６】
また、以下の説明では、不揮発半導体記憶媒体は、前述したように、ＮＡＮＤ型フラッシュメモリである。このため、ページとブロックという用語が使用される。また、或る論理領域（この段落において「対象論理領域」と言う）がライト先であり、且つ、対象論理領域に既にページ（この段落において「第１のページ」と言う）が割り当てられていて第１のページにデータが格納されている場合、対象論理領域には、第１のページに代えて、空きのページ（この段落において「第２のページ」と言う）が割り当てられ、第２のページにデータが書き込まれることになる。第２のページに書き込まれたデータが、対象論理領域にとって最新のデータであり、第１のページに格納されているデータは、対象論理領域にとって旧いデータとなる。以下、各論理領域について、最新のデータを「有効データ」と言い、旧いデータを「無効データ」と言うことがある。また、有効データを格納しているページを「有効ページ」と言い、無効データを格納しているページを「無効ページ」と言うことがある。

【００１７】
さて、まず、本実施形態の概要を説明する。
【００１８】
フラッシュメモリデバイス４００は、図２に示すように、例えば、１以上のフラッシュメモリパッケージ（フラッシュメモリＰＫＧ）１０を備える。フラッシュメモリＰＫＧ１０は、図３に示すように、複数のフラッシュメモリチップ（ＦＭチップ）３２を有する。
【００１９】
フラッシュメモリＰＫＧ１０の媒体インタフェースの一例であるＦＭ　Ｉ／Ｆ制御部２４は、ＦＭチップ３２に対して、チップイネーブル信号（ＣＥ信号）、及び、そのＦＭチップ３２に書き込むデータ、そのデータのライト先とするアドレスを出力する。本実施形態においては、図４に示すように、ＦＭ　Ｉ／Ｆ制御部２４から出力されるＣＥ信号用の信号線２７は、複数のＦＭチップ３２に接続されるように配されている。ＦＭ　Ｉ／Ｆ制御部２４では、一つのＣＥ信号線２７に対して、一つの出力端子（ピン）があればよいので、ＦＭチップ３２の個数よりも少ない数のピンがあればよい。このため、ＦＭ　Ｉ／Ｆ制御部２４を含むＡＳＩＣ等の回路におけるピンの配置に必要な領域を低減することができる。
【００２０】
また、本実施形態においては、図４に示すように、ＦＭ　Ｉ／Ｆ制御部２４から出力されるデータ、アドレス等の信号（ＣＥ信号以外の信号）が流れるバス（バス：制御線が含まれてないがバスということとする）２５は、それぞれスイッチ３１に接続されている。また、スイッチ３１には、Ｍ個（Ｍは２以上の整数、例えば、Ｍ＝４）のＦＭチップ３２がバス２８を介して接続されている。スイッチ３１は、バス２５と、いずれかのバス２８との接続を切り替えることができるようになっている。複数のＦＭチップ３２に対するリードとライトにおいては、バス２５を介してデータ、アドレス等がやり取りされるので、ＦＭ　Ｉ／Ｆ制御部２４では、バス２５が接続されるピンが確保されていればよい。したがって、ＦＭ　Ｉ／Ｆ制御部２４を含むＡＳＩＣ等の回路におけるピンの配置に必要な領域を低減することができる。また、バス２５と、複数のバス２８のいずれかをスイッチ３１により接続させるので、複数のバス２８が電気的に接続されている状態とはならない。このため、バス２５を介してＦＭチップ３２に接続される全体の配線における負荷容量を抑えることができ、ＦＭチップ３２との間のデータ交換の品質を比較的高くすることができる。
【００２１】
次に、本実施形態を詳細に説明する。
【００２２】
図１は、本実施形態に係る計算機システムの構成例を示す。
【００２３】
計算機システムは、ストレージシステム１と、ホストコンピュータ（ホストともいう）２００とを有する。ストレージシステム１、ホスト２００の数は、それぞれ、１以上とすることができる。ストレージシステム１と、ホスト２００とは、通信ネットワーク（例えば、ＳＡＮ（Storage　Area　Network））を介して相互に接続されている。ストレージシステム１は、ホスト２００で利用されるデータを記憶する。ホスト２００は、各種処理を実行し、ストレージシステム１からデータを読み出したり、ストレージシステム１へデータを書き込んだりする。
【００２４】
ストレージシステム１は、複数の記憶デバイスと、それら複数の記憶デバイスに接続されたＲＡＩＤ（Ｒｅｄｕｎｄａｎｔ　Ａｒｒａｙ　ｏｆ　Ｉｎｄｅｐｅｎｄｅｎｔ　（ｏｒ　Ｉｎｅｘｐｅｎｓｉｖｅ）　Ｄｉｓｋｓの略））コントローラデバイス３００とを有する。
【００２５】
複数の記憶デバイスは、複数種類の記憶デバイスを含む。少なくとも１種類の記憶デバ

イスは、１以上存在して良い。記憶デバイスとして、例えば、フラッシュメモリデバイス４００、ＳＳＤ（Solid State Drive）デバイス５００、ＨＤＤ（Hard Disk Drive）デバイス（ＳＡＳ：Serial Attached SCSI）６００及びＨＤＤデバイス（ＳＡＴＡ：Serial ATA）７００がある。

【００２６】

ＲＡＩＤコントローラデバイス３００は、複数のＲＡＩＤコントローラ３０１を有する。各ＲＡＩＤコントローラ３０１は、フラッシュメモリデバイス４００、ＳＳＤデバイス５００、ＨＤＤデバイス（ＳＡＳ）６００及びＨＤＤデバイス（ＳＡＴＡ）７００と内部バスを介して接続されている。

【００２７】

なお、ＲＡＩＤコントローラ３０１は、フラッシュメモリデバイス４００、ＳＳＤデバイス５００、ＨＤＤデバイス（ＳＡＳ）６００及びＨＤＤデバイス（ＳＡＴＡ）７００にとっての上位装置の一例である。ＲＡＩＤコントローラ３０１は、ＲＡＩＤコントローラ３０１にとっての上位装置（例えば、ホスト２００）からＩ／Ｏコマンドを受け、そのＩ／Ｏコマンドに従い、フラッシュメモリデバイス４００、ＳＳＤデバイス５００、ＨＤＤデバイス（ＳＡＳ）６００又はＨＤＤデバイス（ＳＡＴＡ）７００へのアクセス制御を行う。ＲＡＩＤコントローラ３０１は、フラッシュメモリデバイス４００、ＳＳＤデバイス５００、ＨＤＤデバイス（ＳＡＳ）６００、ＨＤＤデバイス（ＳＡＴＡ）７００のそれぞれの記憶領域をそれぞれ異なる記憶階層として管理し、データのライト先の論理領域に対して、いずれかの記憶階層の記憶領域を割当てる処理を行うようにしてもよい。

【００２８】

ここで、ＳＳＤデバイス５００の方がフラッシュメモリデバイス４００よりもフラッシュメモリの書き込み可能回数が多い、一方、読み出し速度及びコスト面では、フラッシュメモリデバイス４００の方が優れているという特徴があるとする。このため、ＲＡＩＤコントローラ３０１は、リードが比較的頻繁に行われるデータを、フラッシュメモリデバイス４００に格納し、ライトが比較的頻繁に行われるデータを、ＳＳＤデバイス５００に格納してもよい。

【００２９】

図２は、本実施形態に係るフラッシュメモリデバイスの構成例を示す。

【００３０】

フラッシュメモリデバイス４００は、１以上の上位Ｉ／Ｆスイッチ（上位Ｉ／Ｆ　Ｓｗｉｔｃｈ）４０１と、１以上のフラッシュメモリパッケージ（ＰＫＧ）１０とを有する。上位Ｉ／Ｆスイッチ４０１は、ＲＡＩＤコントローラ３０１と、複数のフラッシュメモリＰＫＧ１０との間のデータの中継を行う。

【００３１】

図３は、本実施形態に係るフラッシュメモリパッケージの構成例を示す。

【００３２】

フラッシュメモリＰＫＧ１０は、主記憶メモリの一例としてＤＲＡＭ（Dynamic Random Access Memory）１１を有し、また、ＦＭコントローラ２０と、複数（又は１つ）のＤＩＭＭ（Dual Inline Memory Module）３０とを有する。ＤＲＡＭ１１は、ＦＭコントローラ２０で使用するデータ等を記憶する。ＤＲＡＭ１１は、ＦＭコントローラ２０に搭載されていても良いし、ＦＭコントローラ２０とは別の部材に搭載されていても良い。

【００３３】

ＦＭコントローラ２０は、例えば、１つのＡＳＩＣ（Application Specific Integrated Circuit）で構成されており、ＣＰＵ２１と、内部バス２２と、上位Ｉ／Ｆ（インタフェース）２３と、複数（又は１つ）のＦＭ　Ｉ／Ｆ制御部２４とを有する。内部バス２２は、ＣＰＵ２１と、上位Ｉ／Ｆ２３と、ＤＲＡＭ１１と、ＦＭ　Ｉ／Ｆ制御部２４とを通信可能に接続する。

【００３４】

上位Ｉ／Ｆ２３は、上位Ｉ／Ｆ　Ｓｗｉｔｃｈ４０１に接続され、上位装置との通信を

仲介する。上位Ｉ／Ｆ２３は、例えば、ＳＡＳのＩ／Ｆである。ＦＭ　Ｉ／Ｆ制御部２４は、複数のＦＭチップ３２とのデータのやり取りを仲介する。本実施形態では、ＦＭ　Ｉ／Ｆ制御部２４は、ＦＭチップ３２とのやり取りを実行するバス（データバス等）を複数組有し、複数のバスを用いて、複数のＦＭチップ３２とのデータのやり取りを仲介する。本実施形態では、ＤＩＭＭ３０毎にＦＭ　Ｉ／Ｆ制御部２４が設けられ、ＦＭ　Ｉ／Ｆ制御部２４は、その制御部２４に接続されたＤＩＭＭ３０が有する複数のＦＭチップ３２との通信を仲介する。なお、ＦＭ　Ｉ／Ｆ制御部２４が担当するＤＩＭＭ３０の枚数は、２以上であってもよい。ＣＰＵ２１は、ＤＲＡＭ１１（又は図示しない他の記憶領域）に記憶されるプログラムを実行することによって、各種処理を実行することができる。ＣＰＵ２１は複数あってもよく、複数のＣＰＵ２１が各種処理を分担してもよい。ＣＰＵ２１による具体的な処理については、後述する。

【００３５】

ＤＩＭＭ３０は、１以上のＳＷ３１と、複数のＦＭチップ３２とを有する。ＦＭチップ３２は、例えば、ＭＬＣ（Multi Level Cell）型のＮＡＮＤフラッシュメモリチップである。ＭＬＣ型のＦＭチップは、ＳＬＣ型のＦＭチップと比べて書き換え可能な回数が劣るが、１セルあたりの記憶容量が多いという特徴を有している。

【００３６】

ＳＷ３１は、ＦＭ　Ｉ／Ｆ制御部２４と、データバスを含むバス２５を介して接続されている。本実施形態では、ＳＷ３１は、ＦＭ　Ｉ／Ｆ制御部２４に接続されるデータバスを含む一組のバス２５に対して、一つ対応するように設けられている。また、ＳＷ３１は、複数のＦＭチップ３２とデータバスを含むバス２８を介して接続されている。ＳＷ３１は、ＦＭ　Ｉ／Ｆ制御部２４からのバス２５と、いずれかのＦＭチップ３２のバス２８とを選択的に切り替えて接続できるようになっている。ここで、ＤＩＭＭ３０に、ＳＷ３１と、複数のＦＭチップ３２とが設けられ、配線がされているので、これらを接続するためのコネクタを別に用意せずともよく、必要なコネクタ数を低減することが期待できる。

【００３７】

なお、図３によれば、ＦＭチップ３２が別のＦＭチップ３２を介することなくＳＷ３１に接続されているが、ＦＭチップ３２が別のＦＭチップ３２を介してＳＷ３１に接続されても良い。すなわち、ＳＷ３１に、直列になった２以上のＦＭチップ３２が接続されても良い。

【００３８】

図４は、本実施形態に係るフラッシュメモリパッケージの一部の詳細な構成例を示す。

【００３９】

ＦＭ　Ｉ／Ｆ制御部２４は、ＥＣＣ（Error Correcting Code）回路２４１と、制御レジスタ２４２と、ＦＭ／ＳＷ制御部２４３と、バッファ２４４と、ＦＭバスプロトコル制御部（図では「プロトコル１、プロトコル２」のように表記）２４６と、ＤＭＡ（Direct Memory Access）部２４７とを有する。本実施形態では、バッファ２４４、ＦＭバスプロトコル制御部２４６、及びＤＭＡ部２４７の組を、ＦＭＩ／Ｆ制御部２４が担当するデータバスの数分（例えば、２組）備えている。

【００４０】

ＤＲＡＭ１１では、後述するように、ライト対象のデータが複数のデータ要素に分割される。ＥＣＣ回路２４１は、ＤＲＡＭ１１からライト対象のデータ要素を読み出し、ライト対象のデータ要素に対応する（例えば付加される）誤り訂正符号を生成する誤り訂正処理を実行し、ライト対象のデータ要素と、そのデータ要素に対応する誤り訂正符号とをバッファ２４４に書き込む。

【００４１】

また、ＥＣＣ回路２４１は、バッファ２４４からリード対象のデータ要素とそのデータ要素に対応する誤り訂正符号とを含むデータを読み出し、リード対象データ要素に誤りが発生しているか否かをそのデータ要素に対応する誤り訂正符号を用いて判断する。その判断の結果が肯定の場合、ＥＣＣ回路２４１は、リード対象データ要素の誤りを訂正する誤

り訂正処理を実行する。ＥＣＣ回路２４１は、ＤＲＡＭ１１に、リード対象のデータを格納する。

【００４２】

本実施形態では、ＥＣＣ回路２４１は、複数のデータバスに接続される複数のＦＭチップ３２に対する誤り訂正符号生成処理、誤り訂正処理を担当する。なお、誤り訂正符号生成処理を実行する符号生成回路部、誤り訂正処理を実行する誤り訂正回路部は、一つであってもよく、複数であってもよい。符号生成回路部及び／又は誤り訂正回路部の数を抑えると、ＦＭ　Ｉ／Ｆ制御部２４の大きさを抑えることができる。なお、符号生成回路部については、回路規模が比較的小さいので、符号生成回路部は複数存在してもよい。いずれのケースでも、少なくとも誤り訂正回路部が、複数のデータバスについて共通なので、回路規模の軽減を期待することができる。

【００４３】

制御レジスタ２４２は、ＦＭチップ３２に対するアクセスを制御するために必要な情報を記憶する。アクセスを制御するために必要な情報は、例えば、ＣＰＵ２１の制御、ＦＭバスプロトコル制御部２４６により設定される。

【００４４】

ＦＭ／ＳＷ制御部２４３は、制御レジスタ２４２の設定に応じて、ＤＩＭＭ３０の複数のＳＷ２６を切り替える信号（切替信号）と、アクセス対象のＦＭチップ３２を選択するチップイネーブル信号（ＣＥ信号）とを出力する。本実施形態では、ＦＭ／ＳＷ制御部２４３には、複数のＣＥ信号線２７（２７－１、２７－２、２７－３）と、それら複数のＣＥ信号線２７に共通の、切替信号用の信号線２６（切替信号線）が接続されている。各ＣＥ信号線２７は、異なるＳＷ３１の配下にある異なるＦＭチップ３２に接続される。

【００４５】

ＦＭ／ＳＷ制御部２４３に接続された切替信号線２６は、ＤＩＭＭ３０の複数のＳＷ３１（ＳＷ１、ＳＷ２）に接続される。これにより、複数のＳＷ３１には、同じ切替信号が供給されることとなる。例えば、同一のＤＩＭＭ３０における複数のＳＷ３１の全てについて、同一番号のピンには、同一のＣＥ信号線２７に接続されたＦＭチップ３２が接続されているとする。このケースでは、それら複数のＳＷ３１が同じ切替信号を受信すれば、各ＳＷ３１の接続先を、同一のＣＥ信号線２７に接続されたＦＭチップ３２とすることができる。故に、複数のデータ要素の書込み先を、同一のＣＥ信号線２７に接続された複数のＦＭチップ３２にすることができ、以って、それら複数のデータ要素を並行して書き込むことが期待できる。

【００４６】

ＦＭ／ＳＷ制御部２４３に接続されたＣＥ信号線２７は、複数のＳＷ３１が担当する複数のＦＭチップ３２に対して接続されている。本実施形態では、ＣＥ信号線２７－１は、ＦＭ＃１－１と、ＦＭ＃２－１とに接続され、ＣＥ信号線２７－２は、ＦＭ＃１－２と、ＦＭ＃２－２とに接続され、同様に、ＣＥ信号線２７－Ｎは、ＦＭ＃１－Ｎと、ＦＭ＃２－Ｎとに接続されている。このような構成により、同一のＣＥ信号線２７に接続された複数のＦＭチップ３２には、ＣＥ信号が、ほぼ同時に供給されることとなる。このため、これらＦＭチップ３２は、略同時に並行して動作することができる。本実施形態においては、切替信号が供給されると、ＳＷ＃１及び＃２では、それぞれ同一のＣＥ信号線２７に接続されているＦＭチップ３２のデータバスを含むバスが、ＦＭ　Ｉ／Ｆ制御部２４に接続されるように切り替えられる。

【００４７】

バッファ２４４は、ＦＭチップ３２へのライト対象データ要素及びそれの誤り訂正符号を一時的に記憶する。また、バッファ２４４は、ＦＭチップ３２から読み出したリード対象データ要素及びそれの誤り訂正符号を一時的に記憶する。

【００４８】

ＤＭＡ２４７は、バッファ２４４に記憶されたライト対象データ要素及びそれの誤り訂正符号を読み出して、それらをＦＭチップ３２に書き込む。また、ＤＭＡ２４７は、ＦＭ

チップ３２からリード対象データ要素及びそれの誤り訂正符号を読み出して、それらをバッファ２４４に書き込む。

【００４９】

ＦＭバスプロトコル制御部２４６は、制御レジスタ２４２の設定に従って、ＦＭチップ３２に、コマンド（リードコマンド、プログラムコマンド）を発行（出力）する。また、ＦＭバスプロトコル制御部２４６は、コマンドに対するＦＭチップ３２の動作結果（ステータス）を確認し、動作結果を制御レジスタ２４２に設定する。

【００５０】

ＤＭＡ２４７には、データバスが接続され、ＦＭバスプロトコル制御部２４６には、コマンド用の信号線が接続され、データバス、コマンド用の信号線等を含むバス２５が、ＳＷ３１に接続される。

【００５１】

ＳＷ３１には、切替信号線２６が接続されるとともに、データバスを含むバス２５が接続される。本実施形態では、ＳＷ＃１及び＃２には、同一の切替信号線２６が接続される。また、ＳＷ３１には、複数のＦＭチップ３２に繋がるデータバスを含むバス２８が接続されている。ＳＷ３１は、切替信号線２６により供給される切替信号に基づいて、複数のバス２８のいずれか１つを選択的にバス２５に接続する。このＳＷ３１によると、１つのバス２５により複数のＦＭチップ３２に対するアクセスを実行することができようになる。また、ＳＷ３１は、複数のバス２８のいずれか１つを選択的にバス２５に接続するので、信号を送信する際におけるバスの負荷容量を抑えることができ、信号の品質を高品質に維持することができる。本実施形態においては、切替信号が供給されると、ＳＷ＃１及び＃２では、それぞれ同一のＣＥ信号線２７に接続されているＦＭチップ３２のデータバスを含むバス２８が、ＦＭ　Ｉ／Ｆ制御部２４に繋がるバス２５に接続されるように切り替えられる。

【００５２】

図５は、本実施形態に係る書込み処理の第１の例を説明する図である。

【００５３】

ＦＭコントローラ２０が、ライト対象のデータ（ライトデータ）を、ＤＲＡＭ１１へ格納し、ライトデータを、複数のデータ要素（＃０～＃６等）に分割し、それら複数のデータ要素を複数のＦＭチップ３２へ転送する。ここで言う「ライトデータ」は、典型的には、ホスト２００からＲＡＩＤコントローラ３０１が受けたデータの一部又は全部である。また、データ要素のサイズは、ＦＭチップ３２のページサイズと、ＥＣＣのサイズとに基づくサイズである。データ要素が圧縮されてページに格納される場合には、データ要素のサイズはページサイズ以上であっても良い。ページには、データ要素とＥＣＣが格納される。

【００５４】

以下、書込み処理の具体例を詳細に説明する。

【００５５】

まず、ＦＭコントローラ２０（例えばＣＰＵ２１）は、ＳＷ＃１を切り替えて、ＦＭチップ＃１－１とバス２５－１とを接続し、バス２５－１を介して、ＦＭチップ＃１－１へデータ要素＃０を転送する。データ要素＃０は、ＦＭチップ＃１－１のページ＃０００に書き込まれる。また、ＦＭコントローラ２０（例えばＣＰＵ２１）は、ＳＷ＃２を切り替えて、ＦＭチップ＃２－１とバス２５－２とを接続し、バス２５－２を介してＦＭチップ＃２－１へデータ要素＃１を転送する。データ要素＃１はＦＭチップ＃２－１のページ＃１００へ書き込まれる。なお、ＦＭコントローラ２０は、ＦＭチップ＃１－１及び＃２－１に接続されたＣＥ信号線２７－１を介してＣＥ信号を送信して良い。これにより、ＦＭチップ＃１－１及び＃２－１にデータ要素＃０及び＃１を並行して書き込むことができる。

【００５６】

同様にして、ＦＭコントローラ２０は、ＳＷ＃３を切り替えて、ＦＭチップ＃３－１へ

データ要素＃２を転送し、ＳＷ＃４を切り替えて、ＦＭチップ＃４－１へデータ要素＃３を転送する。なお、ＦＭコントローラ２０は、ＦＭチップ＃３－１及び＃４－１が接続されたＣＥ信号線２７を介してＣＥ信号を送信して良い。これにより、ＦＭチップ＃３－１及び＃４－１にデータ要素＃２及び＃３を並行して書き込むことができる。

【００５７】

続いて、ＦＭコントローラ２０は、バス２５－１を介して、ＦＭチップ＃１－１へデータ要素＃４を転送する。データ要素＃４はＦＭチップ＃１－１のページ００１に書き込まれる。同様に、ＦＭコントローラ２０は、ＦＭチップ＃２－１へデータ要素＃５を転送し、ＦＭ＃３－１へデータ要素＃６を転送する。

【００５８】

ここで、ＦＭコントローラ２０がＦＭチップ＃１－１へデータ要素＃０を転送すると、バス２５－１がビジー状態となり、ビジー状態の間はバス２５－１を介してデータを転送することはできない。また、ＦＭチップ＃１－１へ転送されたデータ要素＃０は、ＦＭ＃１－１内のバッファ（図示せず）へ格納された後に、ページ＃０００へ書き込まれる。ＦＭ＃１－１は、バッファに格納されたデータ要素＃０の書き込みが完了するまで、ビジー状態となる。一般に、ライト処理の場合、ＦＭチップ３２のビジー状態の時間は、バス２５のビジー状態の時間よりも長い。このため、ＦＭコントローラ２０がデータ要素＃４をＦＭチップ＃１－１に転送する際、ＦＭチップ＃１－１がビジー状態である場合もあるが、この場合には、ＦＭコントローラ２０は、ＦＭチップ＃１－１のビジー状態が解除されてからデータ要素＃４を転送する。

【００５９】

なお、上記の流れにおいて、ＤＲＡＭ１１上のデータ要素（例えば＃０）は、そのデータ要素の転送先のＦＭチップ３２（例えば＃１－１）が接続されているＳＷ３１（例えば＃１）に接続されているバッファ２４４（例えば＃１）に格納される。ＣＰＵ２１からの指令により、プロトコル制御部２４６（例えば＃１）が、ＤＭＡ２４７（例えば＃１）に起動をかける。起動したＤＭＡ２４７が、バッファ２４４（例えば＃１）内のデータ要素（例えば＃０）を、そのデータ要素の格納先のＦＭチップ３２（例えば＃１－１）に転送する。データ要素（例えば＃０）がＦＭチップ３２（例えば＃１－１）に書きこまれれば、そのＦＭチップ３２（例えば＃１－１）から完了ステータスがプロトコル制御部２４６（例えば＃１）に送られる。プロトコル制御部２４６（例えば＃１）は、その完了ステータスを受けたことを表す情報を、制御レジスタ２４２に書き込んで良い。ＣＰＵ２１は、制御レジスタ２４２を参照することで、ＦＭチップ３２（例えば＃１－１）にデータ要素（例えば＃０）が書き込まれたことを知ることができる。

【００６０】

以上のように、ＦＭコントローラ２０は、ライトデータを複数のデータ要素に分割し、連続した２以上のデータ要素を順次異なるＦＭチップ３２に転送する。このため、バス２５及びＦＭチップ３２がビジー状態の時間を利用して、他のバス及び他のＦＭチップ３２に対してデータを転送することができるので、効率的にデータを転送することが可能となる。

【００６１】

なお、さらに別のライトデータがＤＲＡＭ１１に格納された場合には、ＦＭコントローラ２０は、直前回のライトデータの末端のデータ要素が格納されるページを含んだＦＭチップ３２（例えば＃１－１）の次のＦＭチップ３２（例えば＃２－１）から、その別のライトデータを格納していく。別のライトデータも、複数のデータ要素に分割され、それら複数のデータ要素が並行して書き込まれる。そして、ＦＭチップ＃１－１、＃２－１、＃３－１及び＃４－１（つまり１段目のＦＭチップ群）の末端ページまでデータ要素が書き込まれた場合には、ＦＭコントローラ２０は、各ＳＷ３１の接続先を、１段目のＦＭチップ群に属するＦＭチップ３２から別の段のＦＭチップ群に属するＦＭチップ（例えば、２段目のＦＭチップ群に属するＦＭチップ＃１－２、＃２－２、＃３－２、４－２）に切り替え、そのＦＭチップ３２にデータ要素を転送する。この後、格納されたライトデータに

対する上書きライト（フラッシュメモリデバイス４００が提供する論理アドレス空間の同一の論理領域をライト先としたライト）が発生した場合には、ＦＭコントローラ２０は、ＳＷ３１を切り替えて、ｎ段目のＦＭチップ群、（ｎ＋１）段目のＦＭチップ群、…にデータを分散して転送することとなる（ｎは１以上の整数）。この処理において、ＦＭコントローラ２０は、ｎ段目のＦＭチップ群のＦＭチップ３２がビジー状態であるときに、（ｎ＋１）段目のＦＭチップ群のＦＭチップ３２にデータ要素を転送して良い。

【００６２】

本実施形態では、前述したように、ＣＥ信号線２７が、複数のＦＭチップ３２（正確には、異なるＳＷ３１に接続された複数のＦＭチップ３２）で共有されている。ＦＭコントローラ２０は、ＳＷ＃１を切り替えてＦＭチップ＃１－１とバス２５－１を接続し、ＳＷ＃２を切り替えてＦＭチップ＃２－１とバス２５－２とを接続することで、ＦＭ＃１－１とＦＭ＃２－１に並行して連続した２つのデータ要素＃０及び＃１を転送することができる。さらに、ＦＭコントローラ２０は、データ要素＃０をＦＭチップ＃１－１へ、データ要素＃１をＦＭチップ＃２－１へ転送した後、ＣＥ信号線２７－１を介しＣＥ信号を送信することで、ＦＭチップ＃１－１とＦＭチップ＃２－１を同時に（並行して）起動する。これにより、ＦＭチップ＃１－１（＃２－２）は、受信したデータ要素＃０（＃１）を書き込む。

【００６３】

つまり、ＦＭコントローラ２０は、ライトデータを複数のデータ要素に分割し、ＳＷを順次に選択し、ＳＷの接続先を同一のＣＥ信号線に接続されたＦＭチップ３２とし、連続した２以上のデータ要素を、同一のＣＥ信号に接続された２以上のＦＭチップ３２にそれぞれ並行して転送し、その同一のＣＥ信号線を介してＣＥ信号を送信する。これにより、より効率良くライトデータを書き込むことができる。ＦＭコントローラ２０が複数のデータ要素を転送する際、各ＳＷを、それぞれ独立して切り替えるよう制御してもよいし、同期して切り替えるよう制御してもよい。

【００６４】

図１５は、本実施形態に係る論理アドレス層と物理層との関係の一例を示す。

【００６５】

論理アドレス層１４０１は、フラッシュメモリデバイス４００が上位装置（例えば、ＲＡＩＤコントローラ３０１又はホスト２００）に提供する論理アドレス空間である。ここで、論理アドレスとは、例えば、ＬＢＡ（Logical Block Address）で良い。論理アドレス空間１４０１は、複数の論理領域１４１１に分割されて管理される。

【００６６】

物理層１４０５は、複数のＦＭチップ３２が有する記憶空間であり、複数のブロック１４５２で構成されている。各ブロック１４５２は、複数のページ１４５３で構成されている。

【００６７】

論理領域１４１１は、物理ページ１４５３に関連付けられる。どの論理領域１４１１にどのページ１４５３が対応しているかを表す論理物理変換情報は、例えば、ＦＭコントローラ２０が有する記憶領域（例えばＤＲＡＭ１１）に記憶される。その情報は、１以上のＦＭチップ３２にバックアップされても良い。

【００６８】

例えば、図１５においては、ＬＢＡ０ｘ００から０ｘ０７までの論理領域１４１１は、ＦＭチップ＃１－１のブロック＃００のページ＃０００に割り当てられ、ＬＢＡ０ｘ０８から０ｘ０Ｆまでの論理領域１４１１は、ＦＭチップ＃２－１のブロック＃１０のページ＃１００に割り当てられている。このように、フラッシュメモリデバイス２０にとっての上位装置の１つであるＲＡＩＤコントローラ３０１から、ＬＢＡ０ｘ００から０ｘ０７までのいずれかのＬＢＡを指定したリード要求が発行された場合は、ＦＭコントローラ２０は、そのリード要求を受け、そのリード要求に従い、論理物理変換情報に基づいて、ページ＃０００からデータ要素をリードし、そのリードしたデータ要素を上位装置に返す。

【００６９】
図１７は、論理物理変換情報の構成例を示す。
【００７０】
論理物理変換情報Ｔ６０１は、ページ毎に、ページを有するブロックの番号、ページの番号、ページの属性（有効ページ、無効ページ又は空きページ）、及び、ページの割当て先の論理領域の論理アドレス（例えば先頭アドレス）を含む。ＣＰＵ２１は、この情報Ｔ６０１を参照することにより、どのＦＭチップに空きブロックがあるか、どのＦＭチップ群におけるブロックが最も有効ページが少ないブロックであるか、及び、どの論理領域にどのページが割り当てられているか等を特定することができる。なお、この段落で言う「ＦＭチップ群」は、１以上のＦＭチップ３２であり、例えば、同一のＦＭ　Ｉ／Ｆ制御部２４に接続されている複数のＦＭチップ３２、同一のＳＷ３１に接続されている複数のＦＭチップ３２、或いは、特定のＦＭチップ３２である。
【００７１】
ところで、ＲＡＩＤコントローラ３０１がホスト２００から指定される論理アドレスと、ＲＡＩＤコントローラ３０１がフラッシュメモリデバイス４００に指定する論理アドレスは同一であっても良いが、本実施形態では、それらは異なる。
【００７２】
以下、ＦＭコントローラ２０によるライトデータの分散ライトと、ＲＡＩＤコントローラ３０１によるＲＡＩＤのストライピングとの違いの一例を、図１６を参照して説明する。
【００７３】
図１６は、本実施形態に係るユーザアドレス空間と論理アドレス空間との関係の一例を示す。
【００７４】
ユーザアドレス空間３００１は、ＬＵ（Ｌｏｇｉｃａｌ　Ｕｎｉｔ）番号とその論理アドレス（ＬＢＡ）によって決定される。同図においては、ＬＵ３０１１が複数あり、各ＬＵ３０１１は、複数の論理ブロック３０２１で構成されている。論理ブロック３０２１は、異なる複数のフラッシュストレージ論理空間３００２の同一論理アドレスの複数の論理ブロック３０２２に割り当てられている。１以上の論理ブロック３０２２で前述した論理領域１４１１（図１５参照）が構成されている。フラッシュストレージ論理空間３００２は、典型的には、フラッシュメモリデバイス４００が提供する論理アドレス空間１４０１である。
【００７５】
同図によれば、ユーザアドレス空間３００１に関してのストライピングは、１つの論理ブロック３０２１が、異なる複数のフラッシュストレージ論理空間３００２に跨っていることを意味する。一方、フラッシュストレージ論理空間３００２（論理アドレス空間１４０１）に関してのストライピングは、図１５によれば、アドレスの連続する２以上の論理領域１４１１が、ＣＥ信号線２７を共通にする異なる２以上のＦＭチップ３２に跨っていることを意味する。
【００７６】
なお、ＬＵ３０１１は、Ｔｈｉｎ　Ｐｒｏｖｉｏｎｉｎｇに従う仮想的なＬＵ（ＴＰ－ＬＵ）の領域に割り当てられるセグメントを含んだプールＬＵでも良い。プールＬＵは、容量プールを構成するＬＵであって、複数のセグメントに分割して管理される。ＴＰ－ＬＵの領域に対してセグメントが割り当てられる。この場合、セグメントが、１以上の論理ブロック３０２１で構成されていて良い。
【００７７】
図６は、本実施形態に係る書込み処理の第２の例を説明する図である。
【００７８】
ホスト＃Ａのデータ要素＃Ａ（例えば、ページサイズ以下のライトデータ、又は、ライトデータにおける末尾のデータ要素）を書き込む処理と、ホスト＃Ｂのデータ要素＃Ｂ（

例えば、ページサイズ以下のライトデータ、又は、ライトデータにおける先頭のデータ要素）を書き込む処理とが発生した場合には、ＦＭコントローラ２０は、ＦＭチップ３２にデータ要素＃Ａを格納させ、そのＦＭチップ３２と同一のＣＥ信号線２７に接続されている別のＦＭチップ３２にデータ要素＃Ｂを格納させるように決定し、データ要素＃Ａ及び＃Ｂをそれらの ＦＭチップ３２にほぼ同時に（並行して）書き込ませる。これにより、複数のホストからのデータを迅速に書き込むことができる。

【００７９】

図７は、フラッシュメモリデバイス４００の正面上方からの斜視図の一例である。

【００８０】

フラッシュメモリデバイス４００は、規格化された幅１９インチのラックにマウントできる形状となっているフラッシュメモリデバイス４００の高さは、例えば、２Ｕとなっている。フラッシュメモリデバイス４００は、例えば、フラッシュメモリＰＫＧ１０を１２個（３列×４段）装填できるようになっている。

【００８１】

図８は、フラッシュメモリＰＫＧ１０の上面側からの斜視図の一例であり、図９は、フラッシュメモリＰＫＧ１０の下面側からの斜視図の一例である。

【００８２】

フラッシュメモリＰＫＧ１０の上面側においては、ＡＳＩＣであるＦＭコントローラ２０がそのＰＫＧ１０の平面方向においてほぼ中央に配置され、その手前側及び奥側に、それぞれ２個ずつＤＩＭＭ３０が配置されている。また、フラッシュメモリＰＫＧ１０の下面側においては、ＦＭコントローラ２０の下面の領域に対して、その手前側及び奥側にそれぞれ２個のＤＩＭＭ３０が配置されている。従って、フラッシュメモリＰＫＧ１０には、８個のＤＩＭＭが配置されている。このように、ＦＭコントローラ２０をほぼ中央に配置しているので、ＦＭコントローラ２０から各ＤＩＭＭ３０への配線の長さをほぼ均等にすることができる。

【００８３】

図１０は、ＤＩＭＭ３０の概略構成の一例を示す。

【００８４】

ＤＩＭＭ３０は、例えば、８個のＦＭチップ３２と、２個のＳＷ３１とを備える。１個のＳＷ３１が４個のＦＭチップ３２へのバスの切替を行うようになっている。

【００８５】

なお、ＤＩＭＭ３０が有するＦＭチップ３２の数及びＳＷ３１の数は、図１０に示す数に限られない。

【００８６】

また、ＤＩＭＭ３０は、ＥＣＣ回路３４が備えてもよい。また、ＦＭ３２毎にＥＣＣ回路３５が備えられてもよい。ＤＩＭＭ３０または、ＦＭ３２がＥＣＣ回路を備える場合は、ＦＭ　ＩＦ制御部２４がＥＣＣ回路２４１を備えなくてもよい。

【００８７】

次に、フラッシュメモリＰＫＧ１０における動作を説明する。以下では、ＦＭ　ＩＦ制御部２４がＥＣＣ回路２４１を備える場合の処理を説明するが、ＤＩＭＭ３０またはＦＭ３２がＥＣＣ回路を備える場合は、ＤＩＭＭ３０またはＦＭ３２のＥＣＣ回路が誤り訂正処理を行う。

【００８８】

図１１は、チップリード処理のフローチャートの一例である。

【００８９】

チップリード処理とは、ＦＭチップ３２からデータ要素を読み出す処理である。ここで、チップリード処理の前においては、ＦＭコントローラ２０のＣＰＵ２１が、リード対象のＦＭチップ３２を特定し、その特定したＦＭチップ３２からデータ要素を読み出すための制御用の設定をＦＭ　Ｉ／Ｆ制御部２４の制御レジスタ２４２に対して行う。

【００９０】

まず、ＦＭ／ＳＷ制御部２４３が、リード元のＦＭチップ３２にバス２５が接続されるように、ＳＷ３１の切替信号を切替信号線２６により送信する。これにより、ＳＷ３１が接続を切り替えて、リード元のＦＭチップ３２がバス２５に接続されるようにする（ステップ１２０１）。

【００９１】

ＦＭ／ＳＷ制御部２４３が、リード元のＦＭチップ３２に繋がるＣＥ信号線２７を介してＣＥ信号を駆動して、リード対象のＦＭチップ３２をアクティブにする（ステップ１２０２）。次いで、ＦＭバスプロトコル制御部２４６は、バス２５を介してリードコマンドを発行する（ステップ１２０３）。これにより、リードコマンドは、バス２５、ＳＷ３１、バス２８を介してリード元のＦＭチップ３２に送信される。次いで、ＤＭＡ２４７が、リード元のＦＭチップ３２からリード対象のデータ要素を読み出して、そのデータ要素をバッファ２４４に格納する（ステップ１２０４）。

【００９２】

次いで、ＦＭバスプロトコル制御部２４６は、ＦＭチップ３２からコマンドに対するステータスを取得し、ステータスを制御レジスタ２４２に格納する。ＣＰＵ２１は、制御レジスタ２４２を参照して、リードが正常に終了していることをチェックし（ステップ１２０５）、正常終了している場合には、ＥＣＣ回路２４１により、バッファ２４４に読み出したデータ要素に対して誤り訂正処理を行わせ、ＤＲＡＭ１１に転送させる（ステップ１２０６）。これにより、ＤＲＡＭ１１には、読み出し対象のデータ要素が格納される。なお、これ以降に、ＣＰＵ２１がＤＲＡＭ１１からリード対象のデータを読み出して、上位の装置に送信することとなる。

【００９３】

図１２は、チップライト処理のフローチャートの一例である。

【００９４】

チップライト処理とは、ＦＭチップ３２にデータ要素を書き込む処理である。ここで、チップライト処理の前においては、ＦＭコントローラ２０のＣＰＵ２１が、ライト対象のＦＭチップ３２を特定し、その特定したＦＭチップ３２にデータ要素をライトするための制御用の設定を、ＦＭ　Ｉ／Ｆ制御部２４の制御レジスタ２４２に対して行う。また、ライト対象のデータ要素は、ＣＰＵ２１により、ＤＲＡＭ１１に格納されている。

【００９５】

ＣＰＵ２１は、ＤＲＡＭ１１からライト対象のデータ要素を読み出し、そのデータ要素をＥＣＣ回路２４１に渡す。ＥＣＣ回路２４１は、ライト対象のデータ要素に対応するＥＣＣを生成し、ライト対象のデータ要素とＥＣＣとを含むデータ（ここで、この処理フローにおいては、ライトデータという）をバッファ２４４へ格納する（ステップ１３０１）。

【００９６】

次いで、ＦＭ／ＳＷ制御部２４３が、ライト先のＦＭチップ３２にバス２５が接続されるように、ＳＷ３１の切替信号を切替信号線２６により送信する。これにより、ＳＷ３１が接続を切り替えて、ライト先のＦＭチップ３２がバス２５に接続されるようにする（ステップ１３０２）。

【００９７】

ＦＭ／ＳＷ制御部２４３が、ライト先のＦＭチップ３２に繋がるＣＥ信号線を介してＣＥ信号を駆動して、ライト先のＦＭチップ３２をアクティブにする（ステップ１３０３）。次いで、ＦＭ　Ｉ／Ｆサブ制御部２４６は、バス２５を介してプログラムコマンド（ライトコマンド）を発行する（ステップ１３０４）。これにより、プログラムコマンドは、バス２５、ＳＷ３１、バス２８を介してライト先のＦＭチップ３２に送信される。次いで、ＤＭＡ２４７が、バッファ２４４からライトデータを読み出して、そのデータをＦＭチップ３２に転送する（ステップ１３０５）。

【００９８】

次いで、ＦＭバスプロトコル制御部２４６は、ＦＭチップ３２からコマンドに対するス

テータスを取得し、ステータスを制御レジスタ２４２に格納する。ＣＰＵ２１は、制御レジスタ２４２を参照して、ライトが正常に終了していることをチェックし（ステップ１３０６）、正常終了している場合には、処理を終了する。

【００９９】

図１３は、チップ多重ライト処理のフローチャートの一例である。

【０１００】

チップ多重ライト処理とは、複数のＦＭチップに複数のデータ要素を並行して書き込む処理である。ここで、チップ多重ライト処理の前においては、ＦＭコントローラ２０のＣＰＵ２１が、ライト先の複数のＦＭチップ３２を特定し、それらのＦＭチップ３２にライトするための制御用の設定を、ＦＭ　Ｉ／Ｆ制御部２４の制御レジスタ２４２に対して行う。本実施形態では、同一のＣＥ信号線２７に接続されている複数のＦＭチップ３２がライト先として特定される。また、ライト対象のデータ要素は、ＣＰＵ２１により、ＤＲＡＭ１１に格納されている。

【０１０１】

ＣＰＵ２１は、ＦＭチップ３２（ここでは、例えば、ＳＷ＃１に接続されているＦＭチップ＃１－１とする）にライトするライト対象のデータ要素をＤＲＡＭ１１から読み出して、そのデータ要素をＥＣＣ回路２４１に渡す。ＥＣＣ回路２４１は、そのライト対象のデータ要素に対応するＥＣＣを生成し、ライト対象のデータ要素とＥＣＣとを含むデータ（この処理フローにおいては、ライトデータという）をバッファ＃１へ格納する（ステップ１４０１）。次いで、ＦＭチップ＃１－１と同じＣＥ信号線２７－１に接続されているＦＭチップ３２（例えば、ＳＷ＃２に接続されているＦＭチップ＃２－１）にライトするライト対象のデータ要素をＤＲＡＭ１１から読み出して、そのデータ要素をＥＣＣ回路２４１に渡す。ＥＣＣ回路２４１は、そのライト対象のデータ要素に対応するＥＣＣを生成し、そのライト対象のデータ要素とＥＣＣとを含むデータ（ライトデータ）をバッファ＃２へ格納する（ステップ１４０２）。

【０１０２】

次いで、ＦＭ／ＳＷ制御部２４３が、ライト先の複数のＦＭチップ＃１－１及び＃２－１にバス２５－１及び２５－２が接続されるように、ＳＷ＃１及び＃２の切替信号を切替信号線２６により送信する。これにより、ＳＷ＃１及び＃２が、接続を切り替えて、ライト先の複数のＦＭチップ＃１－１及び＃２－１が、バス２５－１及び２５－２に接続されるようにする（ステップ１４０３）。

【０１０３】

ＦＭ／ＳＷ制御部２４３が、ライト先のＦＭチップ＃１－１及び＃２－１に繋がるＣＥ信号線２７－１を介してＣＥ信号を駆動して、そのＣＥ信号線２７－１に接続されている複数のＦＭチップ３２をアクティブにする（ステップ１４０４）。

【０１０４】

次いで、ＦＭ　Ｉ／Ｆサブ制御部＃１は、バス２５－１を介してプログラムコマンド（ライトコマンド）を発行する（ステップ１４０５）。これにより、プログラムコマンドは、バス２５－１、ＳＷ＃１、バス２８を介して、ライト先のＦＭチップ＃１－１に送信される。また、これと並行して、ＦＭ　Ｉ／Ｆサブ制御部＃２は、バス２５－２を介してプログラムコマンド（ライトコマンド）を発行する（ステップ１４０６）。これにより、プログラムコマンドは、バス２５－２、ＳＷ＃２、バス２８を介して、ライト先のＦＭチップ＃２－１に送信される。

【０１０５】

次いで、ＤＭＡ＃１が、バッファ＃１からライトデータを読み出し、そのデータをＦＭチップ＃１－１に転送するとともに、ほぼ同時に（並行して）、ＤＭＡ＃２が、バッファ＃２からライトデータを読み出し、そのデータをＦＭチップ＃２－１に転送する（ステップ１３０７）。

【０１０６】

次いで、ＦＭバスプロトコル制御部２４６は、ＦＭチップ＃１－１及び＃２－１からコ

マンドに対するステータスを取得し、ステータスを制御レジスタ２４２に格納する。ＣＰＵ２１は、制御レジスタ２４２を参照して、ライトが正常に終了していることをチェックし（ステップ１３０８）、正常終了している場合には、処理を終了する。

【０１０７】

この多重ライト処理によると、複数のＦＭチップ３２に対して、ほぼ同時に（並行して）、複数のデータ要素をライトすることができるので、ライト処理に要する時間を短縮することができる。

【０１０８】

なお、チップライト処理及びチップ多重ライト処理のいずれにおいても、消去回数を均等にするべくウェアレベリング処理を行うことができる。ウェアレベリング処理は、それらのライト処理とは非同期に行われても良い。

【０１０９】

ライト処理と非同期に行われるウェアレベリング処理によれば、例えば、任意のタイミングで、ＦＭコントローラ２０（例えばＣＰＵ２１）が、消去回数が最も多いブロックを選択し、その選択したブロック内の有効ページから、消去回数が最も少ないブロックに有効データを移動する。移動元のブロックと移動先のブロックは、同一のＦＭチップ３２にあっても良いし、異なるＦＭチップ３２にあっても良い。後者の場合、異なるＦＭチップ３２は、ＣＥ信号線２７が共通であることが望ましい。

【０１１０】

ライト処理において行われるウェアレベリング処理によれば、ＦＭコントローラ２０は、ライト先のＦＭチップ３２から消去回数の最も少ないブロックを書き込み先として選択し、そのブロックにデータ要素を書き込む。なお、この処理では、例えば、図１３或いは図１４において、ＦＭコントローラ２０は、ＣＥ信号線２７を選択する段階で、最も消去回数合計が少ないＣＥ信号線２７を選択し、そのＣＥ信号線２７に接続されているＦＭチップ３２から、そのＦＭチップ３２において最も消去回数が少ないブロックが選択して良い。ＦＭコントローラ２０の記憶領域（例えばＤＲＡＭ１１）は、図１９に例示する消去管理情報１９０１を記憶して良い。この情報１９０１は、ＣＥ信号線２７毎及びブロック毎に、ブロックの消去回数を表す。ＣＥ信号線２７の消去回数合計とは、そのＣＥ信号線２７を共通にする全てのＦＭチップ３２の全てのブロックの消去回数の合計である。ブロックに対して消去処理を行った場合、ＦＭコントローラ２０は、そのブロックに対応する消去回数と、そのブロックを有するＦＭチップ３２が接続されているＣＥ信号線２７に対応する消去回数合計とを更新して良い。この情報１９０１から、各ブロックの消去回数と、各ＣＥ信号線２７についての消去回数合計とを特定することができる。

【０１１１】

次に、リクラメーション処理、すなわち、消去処理可能なブロックを生成する処理を説明する。

【０１１２】

図１８は、リクラメーション処理のフローチャートの一例である。

【０１１３】

リクラメーション処理は、例えば、ＦＭチップ３２における利用可能な容量の枯渇をＦＭコントローラ２０が検出したことを契機として、ＦＭコントローラ２０により実行される。容量の枯渇とは、空きブロックの数が所定割合（所定数）未満になったことを意味する。容量枯渇の検出は、任意の単位でよく、或るＤＩＭＭ上の複数のＦＭ毎でもよい。リクラメーション処理は、ライト処理において空きブロックが枯渇していることが検出されたときに開始されても良いし、ライト処理とは非同期に行われても良い。

【０１１４】

ＦＭコントローラ２０は、空きブロックが枯渇しているＦＭチップ（以下、図１８において「空き枯渇チップ」と言う）３２から、移動元のブロックを選択する（ステップ１７０１）。ここで、移動元のブロックは、空き枯渇チップ３２（或いは、空き枯渇チップ３２において末端ページまでデータが書き込まれているブロック（消去候補ブロック））の

うち、有効ページが最も少ないブロックであることが望ましい。なぜなら、移動させる有効データの総量が最も少なくて済み、以って、リクラメーション処理にかかる時間及び負荷を抑えることが期待できるからである。なお、移動元ブロックは、空き枯渇チップ３２以外のＦＭチップ３２から選択されても良い。

【０１１５】

ＣＰＵ２１は、ステップ１７０１で選択した移動元ブロックを有するＦＭチップ３２とデータ通信可能なＦＭ　ＩＦ制御部２４（又は、バス２５或いはＳＷ３１）と同一の制御部２４（又は、バス２５或いはＳＷ３１）に接続されている複数のＦＭチップ３２に空きブロックが所定数以上あるか否かを判断する（ステップ１７０２）。ここで言う「所定数」は、全てのＦＭチップ３２で一律であっても良いし異なっていても良い。

【０１１６】

ステップ１７０２の判断の結果が肯定的であれば、ＣＰＵ２１は、同一の制御部２４（又は、バス２５或いはＳＷ３１）に接続されている複数のＦＭチップ３２の空きブロックを移動先ブロックとして選択する（ステップ１７０３）。同一のバス２５或いはＳＷ３１に接続されている複数のＦＭチップ３２の空きブロックが優先的に移動先ブロックとして選択されて良い。もし、同一のバス２５或いはＳＷ３１に接続されている複数のＦＭチップ３２に所定数の空きブロックが無ければ、同一の制御部２４における異なるバス２５或いはＳＷ３１に接続されている複数のＦＭチップ３２の空きブロックが移動先ブロックとして選択されても良い。その方が、ストライプ（アドレスの連続した２以上のデータ要素が同一ＣＥ信号線２７の異なるＦＭチップ３２（バス２５が異なるＦＭチップ３２）に配置されること）をより維持し易いと考えられる。例えば、図５において、リクラメーション処理により、ＦＭ＃１－１に格納されたデータ要素＃０がＦＭ＃３－１に格納されると、データ要素＃０とデータ要素＃２が同じＦＭに存在することになる。この状態で、データ要素のリード／ライトが発生すると、バス２５－３及びＦＭ＃３－１のビジー時間が重なるためリード／ライトに時間がかかる。リクラメーションの範囲を制限することで、データ要素のストライプ状態が維持され、以降のリード／ライト処理においても効率の良いデータ転送ができる。

【０１１７】

ステップ１７０３の判断の結果が否定的であれば、ＣＰＵ２１は、異なる制御部２４（又は、バス２５或いはＳＷ３１）に接続されている複数のＦＭチップ３２の空きブロックを移動先ブロックとして選択する（ステップ１７０４）。

【０１１８】

ステップ１７０３又は１７０４の後、すなわち、移動元ブロックと移動先ブロックとが決定した後、ＣＰＵ２１は、移動元ブロック内の有効データを移動先ブロックに移動する（ステップ１７０５）。すなわち、ＣＰＵ２１は、移動元ブロックから有効データを読み出してＤＲＡＭ１１に書き込み、その有効データをＤＲＡＭ１１から移動先ブロックに書き込む。この際、ＥＣＣ回路２４１が誤り訂正処理を行う。なお、ＣＰＵ２１は、移動先ブロックを決定する前に移動元ブロックから有効データを読み出しＤＲＡＭ１１に書き込んでも良い。また、ＤＩＭＭ３０がＥＣＣ回路を備える場合、ＤＩＭＭ３０において誤り訂正処理が可能であるため、移動元ブロックのデータをＤＲＡＭ１１に格納することなく移動先ブロックにデータを転送することができる。同様に、ＦＭ３２がＥＣＣ回路を備える場合、ＦＭ３２において誤り訂正処理が可能であるため、移動元ブロックのデータをＤＲＡＭ１１に格納することなく移動先ブロックにデータを転送することができる。このように、ＤＩＭＭ３０またはＦＭ３２にＥＣＣ回路がある場合は、ＦＭ　Ｉ／Ｆ制御部２４がデータ転送を実行するため、ＣＰＵ２１の処理負荷を低減することができる。

【０１１９】

移動元ブロックから読み出された有効データが移動先ブロックに書き込まれると、移動元ブロック内のデータは全て無効データとなる。ＣＰＵ２１は、移動元ブロックに対して消去処理を行う（ステップ１７０６）。これにより、移動元ブロックが空きブロックとして管理され、再びライト先として選択され得る状態となる。なお、その消去処理の際、図

１９に例示した情報１９００が更新されて良い。すなわち、移動元ブロックに対応する消去回数と、移動元ブロックを有するＦＭチップ３２が接続されているＣＥ信号線２７に対応する消去回数合計とが更新されて良い。

【０１２０】

以上が、本実施形態に係るリクラメーション処理である。

【０１２１】

なお、ステップ１７０３又は１７０４では、選択されたＦＭチップ３２のうち消去回数が最も少ない空きブロックが移動先ブロックとして選択されることが望ましい。これにより、精度の高い消去回数の平準化が可能となる。また、複数のブロックを消去回数に応じて複数のグループに分けて、消去回数が少ないグループからブロックを選択することとしてもよい。この場合、ブロックを検索する時間が短縮される。

【０１２２】

また、ステップ１７０１で、複数のブロックが移動元ブロックとして選択されても良いし、ステップ１７０３及び１７０４で、複数の空きブロックが移動先ブロックとして選択されても良い。この場合、複数の移動元ブロックは、ＣＥ信号線２７を共通にする複数のＦＭチップ３２から選択されることが望ましいが、異なるＣＥ信号線２７に接続されている複数のＦＭチップ３２から選択されても良い。なぜなら、チップリード処理はチップライト処理と比べてチップビジーとなる時間が短いから、つまり、性能への影響が小さいからである。一方、複数の移動先ブロックは、ＣＥ信号線２７を共通にする複数のＦＭチップ３２から選択されることが望ましい。そして、有効データの移動（ステップ１７０５）の際には、ＣＥ信号線２７を共通にする複数のＦＭチップ３２に複数の有効データが並行して書き込まれることが望ましい。また、ステップ１７０６において消去処理を実行する際、同一ＣＥ２７上に消去可能なブロックが存在する場合、それらも同時に（並行して）消去することができる。一般に消去処理は時間がかかるため、複数ブロックに対してまとめて消去処理を実行するのが効率的である。

【０１２３】

次に、本実施形態に係るリフレッシュ処理を説明する。

【０１２４】

リフレッシュ処理は、有効ページを有するブロックに対して定期的（例えば、そのブロックの前回のリフレッシュ処理から３０日経過した場合）に行われても良いし、リード時のＥＣＣエラーが所定ｂｉｔ数以上のブロックがある場合にそのブロックを対象に行われても良い。リフレッシュ処理は、ＦＭコントローラ２０により実行される。

【０１２５】

リフレッシュ処理では、ＦＭコントローラ２０は、リフレッシュ処理の対象のブロックを移動元のブロックとする。その後、（１）移動先のブロックの決定、（２）移動元ブロックから移動先ブロックへの有効データの移動、及び（３）移動元ブロックに対する消去処理が行われる。これら（１）～（３）については、上述したリクラメーション処理と同様である。

【０１２６】

以上、一実施形態を説明したが、本発明は、この実施形態に限定されるものでなく、その趣旨を逸脱しない範囲で種々変更可能であることはいうまでもない。

【０１２７】

例えば、上記実施形態では、不揮発半導体記憶媒体の一例として、ＮＡＮＤ型のフラッシュメモリが採用されているが、不揮発半導体記憶媒体は、これに限られない。例えば、記憶媒体は、相変換メモリでもよい。

【０１２８】

また、上記実施形態では、複数のＦＭチップ３２を搭載するメモリモジュールは、ＤＩＭＭ３０であるが、ＤＩＭＭ以外のメモリモジュールが採用されもよい。

【０１２９】

また、上記実施形態では、同一のＤＩＭＭ３０における複数のＦＭチップ３２が同一の

ＣＥ信号線２７で接続されるが、異なるＤＩＭＭ３０の複数のＦＭチップ３２が同一のＣＥ信号線２７で接続されても良い。

【０１３０】

また、上記実施形態では、フラッシュメモリデバイス４００では、並行して複数のＦＭチップ３２に複数のデータ要素を書き込むことができる。すなわち、時間当たりに書き込めるデータの量が大きい。このため、ＲＡＩＤコントローラ３０１は、フラッシュメモリデバイス４００内のデータ量（又は、フラッシュメモリデバイス４００間におけるデータ転送における転送単位のデータ量）を、フラッシュメモリデバイス４００以外の種類の記憶デバイス（例えば、ＳＳＤデバイス５００、ＨＤＤデバイス（ＳＡＳ）６００、又はＨＤＤデバイス（ＳＡＴＡ）７００）が関わるデータ転送における転送単位のデータ量よりも大きくするように制御してもよい。

【０１３１】

また、例えば、フラッシュメモリＰＫＧ１０の構成は、図１４に示す構成でも良い。図１４に示すフラッシュメモリＰＫＧは、ＣＥ信号線２７についてもＳＷ３３で切り替えることができるようになっている。ＦＭ／ＳＷ２４８には、ＣＥ信号線２７が接続されている。ＣＥ信号線２７は、バス２５とともに、ＳＷ＃１に接続されている。ＣＥ信号線２７は、ＳＷ＃２にも接続されている。ＳＷ３３（＃１、＃２）は、ＣＥ信号線及びデータバスを含むバス２９により複数のＦＭチップ３２に接続されている。ＳＷ３３は、切替信号線２６により供給される切替信号に基づいて、複数のバス２９のいずれか一つを選択的にバス２５及びＣＥ信号線２７に接続する。この構成によれば、ＦＭ　Ｉ／Ｆ制御部２４において出力するＣＥ信号線２７の本数を低減することができ、ＦＭ　Ｉ／Ｆ制御部２４のチップサイズを小型化することができる。

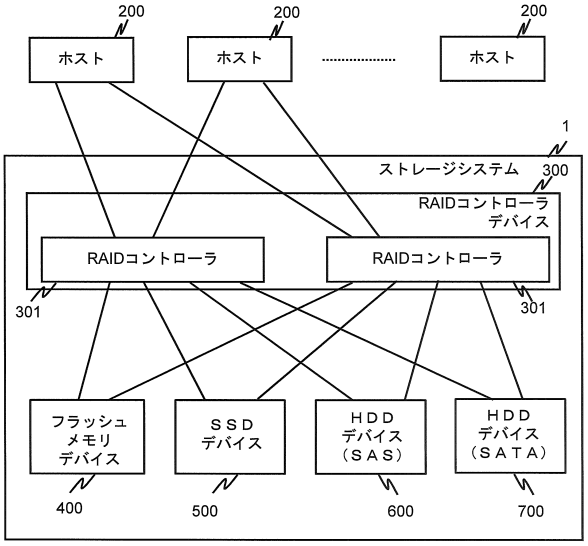
【符号の説明】

【０１３２】

１…ストレージシステム、１０…フラッシュメモリＰＫＧ、４００…フラッシュメモリデバイス

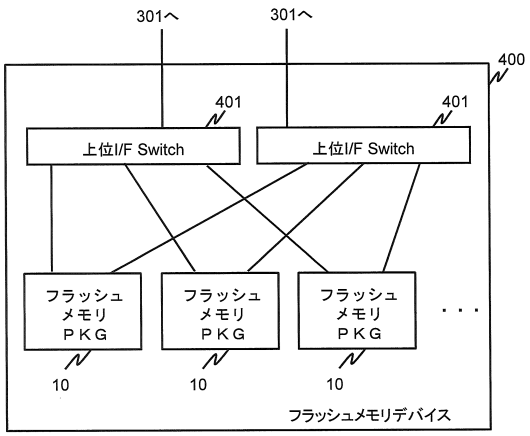
【図１】

FIG. 1



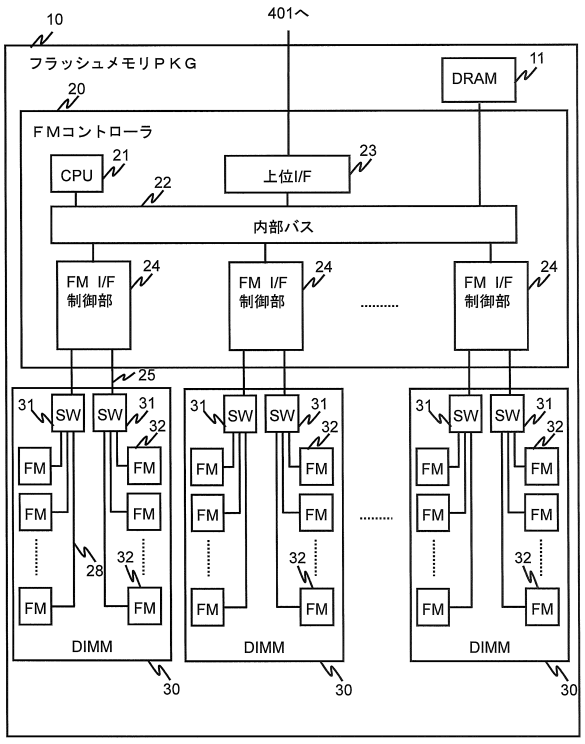
【図２】

FIG. 2



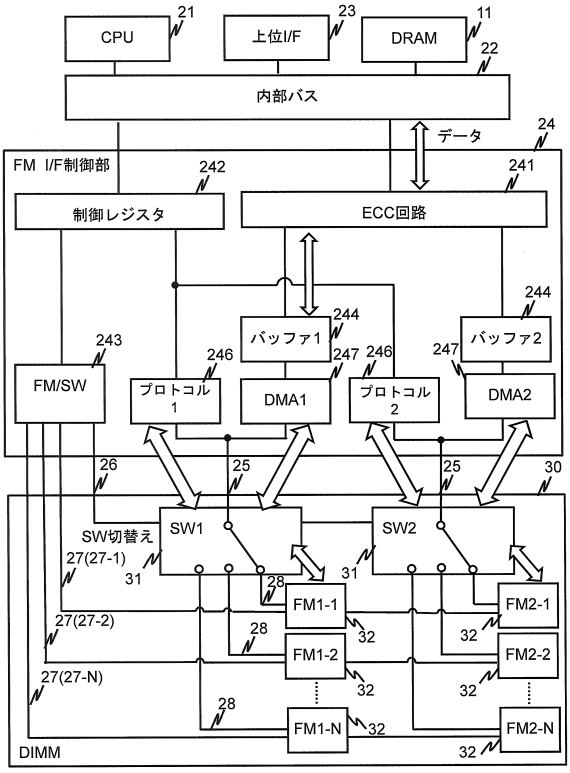
【図３】

FIG. 3



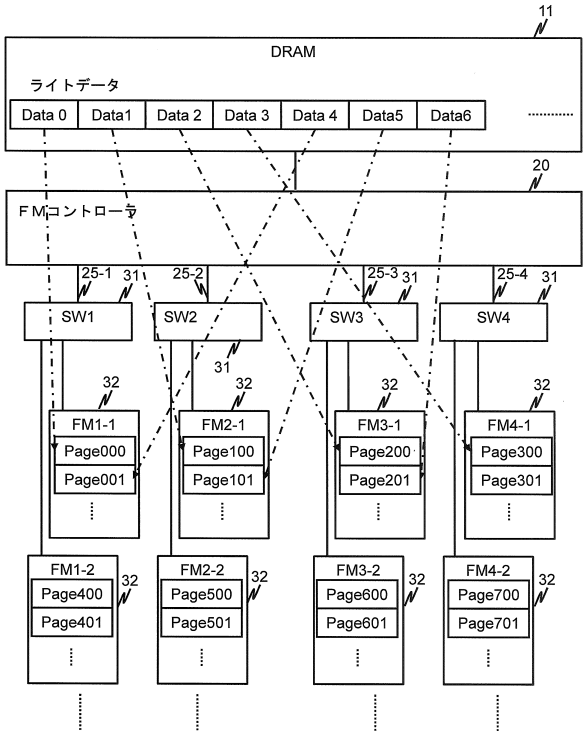
【図４】

FIG. 4



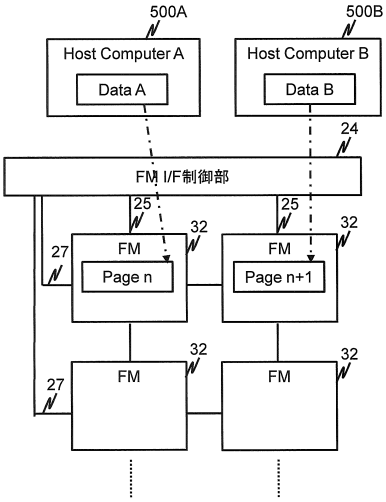
【図５】

FIG. 5



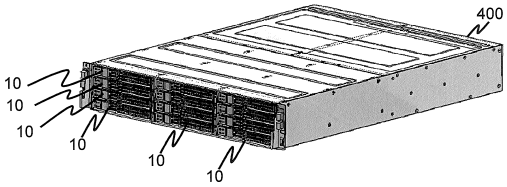
【図６】

FIG. 6



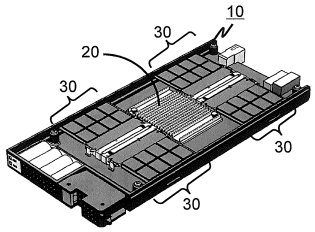
【図７】

FIG. 7



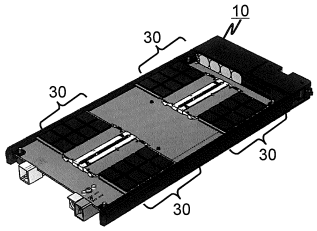
【図８】

FIG. 8



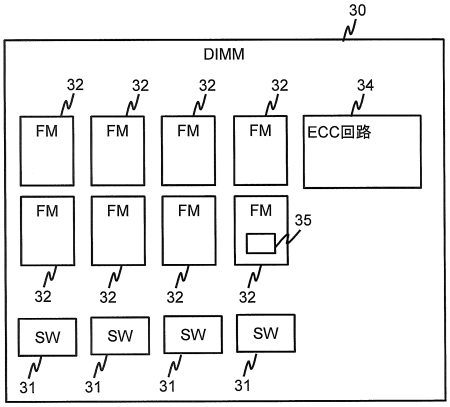
【図９】

FIG. 9



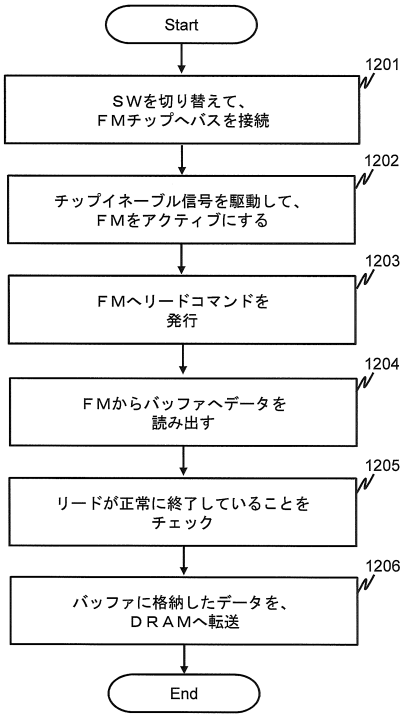
【図１０】

FIG. 10



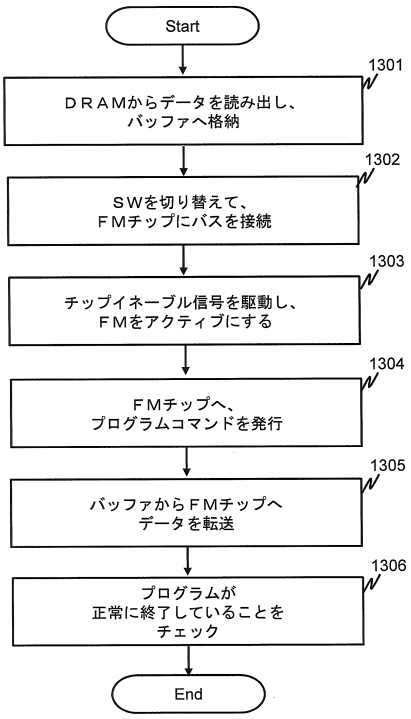
【図１１】

FIG. 11



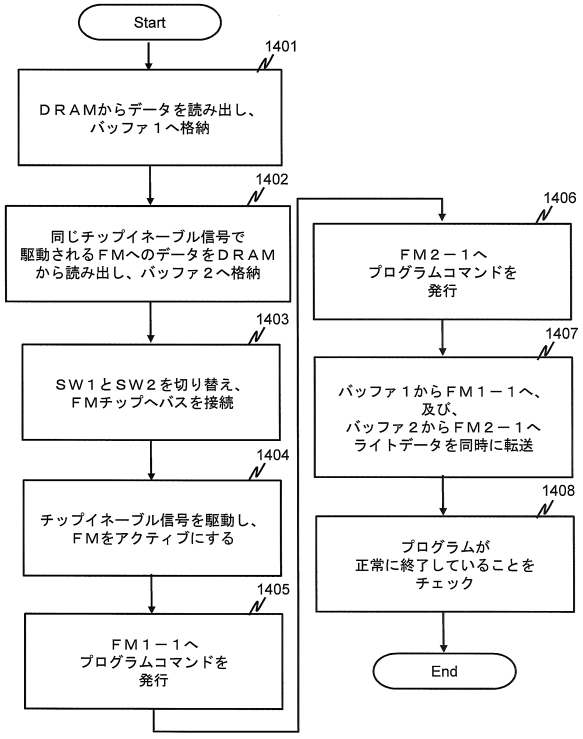
【図１２】

FIG. 12



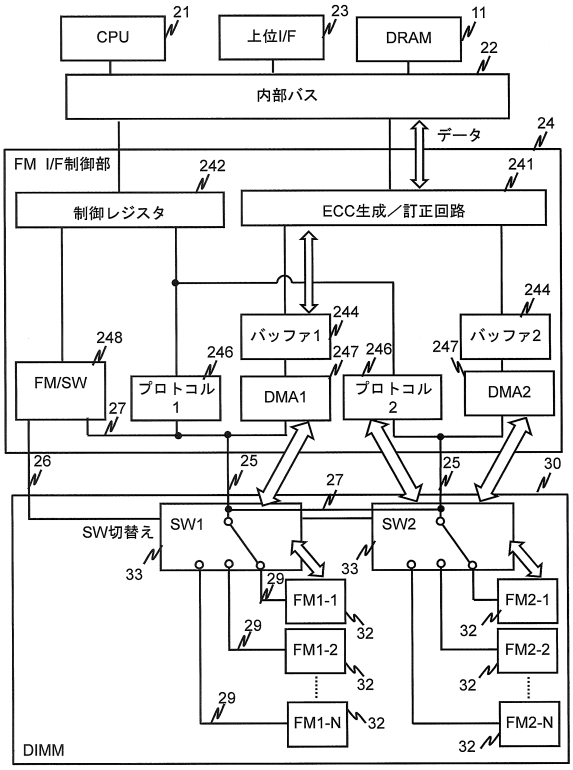
【図１３】

FIG. 13



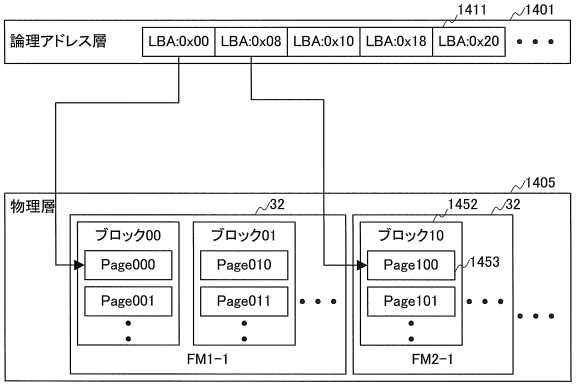
【図１４】

FIG. 14



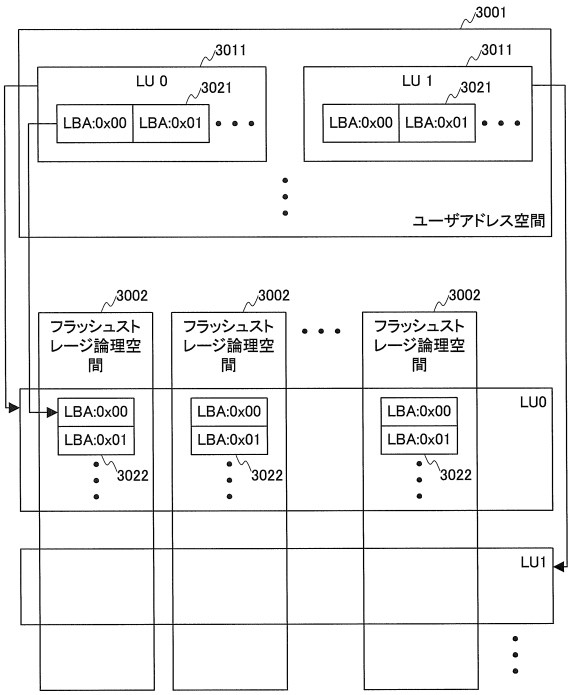
【図１５】

FIG. 15



【図１６】

FIG. 16



【図１７】

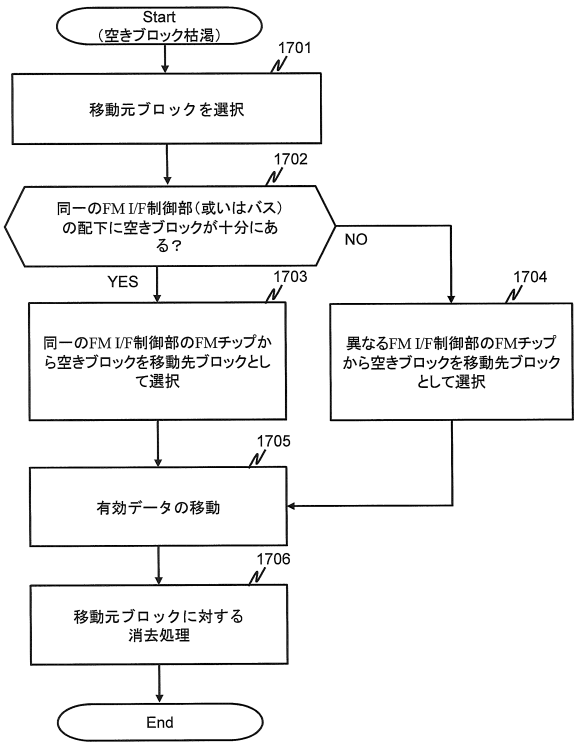
FIG. 17

T601

ブロック番号	ブロック内ページ番号	属性	アドレス
00	000	有効	0x00
	001	空き	なし
	002	空き	なし
	003	空き	なし
⋮			
10	100	有効	0x08
	101	空き	なし
	102	空き	なし
	103	空き	なし
⋮			

【図１８】

FIG. 18



【図１９】

FIG. 19

1900

CE信号線番号	FMチップ番号	ブロック番号	ブロック消去回数	合計
0	0	0	30	500
		1	40	
		・・	・・	
	1	・・	・・	
	・・	・・	・・	
1	・・	・・	・・	・・
・・	・・	・・	・・	・・

フロントページの続き

(51)Int.Cl. FI
G06F 13/14 (2006.01) G06F 13/14 310F

(72)発明者 小川 純司
日本国東京都千代田区丸の内一丁目６番６号 株式会社日立製作所内
(72)発明者 小関 英通
日本国東京都千代田区丸の内一丁目６番６号 株式会社日立製作所内

審査官 桜井 茂行

(56)参考文献 特開２０１０－０４９５８６（ＪＰ，Ａ）
特表２００８－５２３５２８（ＪＰ，Ａ）
特開２００５－２５８８７４（ＪＰ，Ａ）

(58)調査した分野(Int.Cl., ＤＢ名)
Ｇ０６Ｆ １３／１６－１３／１８
Ｇ０６Ｆ １２／００－１２／０６
Ｇ０６Ｆ ３／０６
Ｇ０６Ｆ １２／１６
Ｇ０６Ｆ １３／１４