(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4960705号
(P4960705)
(45)発行日　平成24年6月27日(2012.6.27)　(24)登録日　平成24年3月30日(2012.3.30)

(51)Int.Cl.　　F I
G11C 29/00 (2006.01)　G11C 29/00 603Z
G11C 16/06 (2006.01)　G11C 29/00 603D
G11C 16/02 (2006.01)　G11C 17/00 639B
G06F 12/16 (2006.01)　G11C 17/00 601U
　　G06F 12/16 310R

請求項の数 17　(全 25 頁)

(21)出願番号　特願2006-547154 (P2006-547154)
(86)(22)出願日　平成16年12月15日 (2004.12.15)
(65)公表番号　特表2007-520842 (P2007-520842A)
(43)公表日　平成19年7月26日 (2007.7.26)
(86)国際出願番号　PCT/US2004/042400
(87)国際公開番号　WO2005/066974
(87)国際公開日　平成17年7月21日 (2005.7.21)
審査請求日　平成19年12月7日 (2007.12.7)
(31)優先権主張番号　10/750,157
(32)優先日　平成15年12月30日 (2003.12.30)
(33)優先権主張国　米国(US)

(73)特許権者　506197901
サンディスク　コーポレイション
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
(74)代理人　100075144
弁理士　井ノ口　壽
(72)発明者　ゴンザレス，カルロス　ジェイ.
アメリカ合衆国、９５０３２、カリフォルニア州、ロス　ガトス、マッツソン　アベニュー　２４９
(72)発明者　ブルース，アラン　ディー.
オーストラリア、６１６９、ウェスタンオーストラリア、ウォーンブロ、ブライトン　クローズ　６

最終頁に続く

(54)【発明の名称】マルチブロック単位へのブロックの適応決定論的グループ化

(57)【特許請求の範囲】
【請求項１】
少なくとも２つのプレーンに編成されたメモリ記憶素子のアレイを有する不揮発性メモリ（１１０３）システムを操作する方法であって、個々のプレーンは複数の重ならない記憶素子のブロックに分割され、ブロックは一緒に消去可能なメモリ記憶素子の最小グループを含み、個々のブロックは記憶素子の複数のページに分割され、ページは一緒にプログラム可能なメモリ記憶素子の最小グループである方法において、
メタブロック構成要素が１つの単位として一緒に消去されるメタブロックの当初のリンキングを形成するステップと、
前記当初のリンキングを形成するステップの後で、同じプレーン中に利用可能な予備物理ブロックがあれば、欠陥物理ブロックに関連付けられたメタブロック構成要素ブロックを、同じプレーン中の予備物理ブロックへリダイレクトするステップと、
同じプレーン中に利用可能な予備物理ブロックがなければ、欠陥物理ブロックに関連付けられていないメタブロック構成要素を、予備ブロック領域に割り当てるステップと、
メタブロックに対応するエントリを含む欠陥マップ構造（１１０１）を更新するステップと、
を含む方法。
【請求項２】
請求項１記載の方法において、
メタブロック・リンキングの記録を不揮発性メモリ（１１０３）に格納するステップを

さらに含む方法。
【請求項３】
請求項２記載の方法において、
各々が複数のブロックで構成され、それによってコントローラが不揮発性メモリにアクセスする、メタブロック・リンキングのセットを確立するステップをさらに含み、
前記記録は、ブロックの観点からのメタブロック・リンキングのセットの完全な規格である方法。
【請求項４】
請求項２記載の方法において、
各々が複数のブロックで構成され、それによってコントローラが不揮発性メモリにアクセスする、メタブロック・リンキングのセットを確立するステップをさらに含み、
前記メタブロック・リンキングのセットは規則またはアルゴリズムに従って形成され、前記記録はその規則に対する例外であるリンキングから成る方法。
【請求項５】
請求項３または４記載の方法において、
前記予備物理ブロックは、リンクされていないブロックのリストから選ばれる方法。
【請求項６】
請求項５記載の方法において、
前記リンクされていないブロックのリストは、不揮発性メモリ中に保持される方法。
【請求項７】
請求項６記載の方法において、
欠陥ブロックを前記ブロックのうちの別の１つと置き換えるステップに続いて、前記リンクされていないブロックのリストを更新するステップをさらに含む方法。
【請求項８】
請求項３または４記載の方法において、
前記予備物理ブロックは、以前別のリンキングに属していたブロックから選ばれる方法。
【請求項９】
請求項３または４記載の方法において、
リンクされていないブロックのリストを保持するステップと、
前記メタブロック・リンキングの第１のメタブロック中の１つ以上のブロックが欠陥であることを決定するステップと、
第１のメタブロック中の欠陥ではないブロックを、リンクされていないブロックのリストに加えるステップと、
をさらに含む方法。
【請求項１０】
請求項３または４記載の方法において、
前記メタブロック・リンキングの第１のメタブロック中の１つのブロックが欠陥であることを決定するステップと、
その欠陥ブロックについて代替のブロックが利用可能であるかどうかを決定するステップと、
代替のブロックが利用可能でないことを決定したことに応じて、第１のメタブロックをメタブロック・リンキングのセットから取り除くステップと、
をさらに含む方法。
【請求項１１】
請求項３または４記載の方法において、
前記不揮発性メモリは複数の準独立アレイを含み、前記メタブロック・リンキングの任意の１つの中にある複数のブロックのうちの各ブロックは、前記準独立アレイのうちの異なるアレイからのものである方法。
【請求項１２】

請求項３または４記載の方法において、
前記不揮発性メモリは複数の準独立アレイを含み、前記メタブロック・リンキングの任意の１つの中にある複数のブロックは同じ準独立アレイからの対のブロックで構成され、各々の対は前記準独立アレイのうちの異なるアレイからのものである方法。
【請求項１３】
請求項１１または１２記載の方法において、
前記準独立アレイは、別個のチップ上にある方法。
【請求項１４】
請求項２記載の方法において、
前記メタブロック・リンキングの記録は、不揮発性メモリ（１１０３）の、ユーザデータ用に割り当てられた部分以外の部分に格納される方法。
【請求項１５】
請求項２記載の方法において、
各ブロックは複数のセクタで構成され、各セクタはデータ領域およびオーバーヘッド領域を含み、データを含むブロックについての記録情報はそれらのオーバーヘッド領域中に保持される方法。
【請求項１６】
請求項１５記載の方法において、
データを含まないブロックについての記録情報は、不揮発性メモリ（１１０３）の、ユーザデータ用に割り当てられた部分以外の部分に格納される方法。
【請求項１７】
請求項３または４記載の方法において、
前記メタブロック・リンキングのセットを確立するステップは、アルゴリズムに従ってメタブロック・リンキングを形成することを含む方法。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、一般に、半導体不揮発性データ格納システムに関し、より具体的には、物理ブロックを、不揮発性データ格納システム中の欠陥に対応するより大きな論理構造に形成するためのシステムおよび方法に関する。
【背景技術】
【０００２】
フラッシュメモリのような不揮発性メモリデバイスは一般に、大量データ格納サブシステムとして用いられている。このような不揮発性メモリデバイスは典型的に、ホストシステムに取り外し可能に接続される密封カード中にパッケージされ、ホストシステム内の取り外しできない組み込み形記憶装置としてパッケージされることもある。典型的な実装例において、サブシステムは、１つ以上のフラッシュデバイスと、しばしば１つのサブシステムコントローラとを含む。
【０００３】
現行の市販されているメモリカードのフォーマットとして、パーソナルコンピュータメモリカード国際協会（ＰＣＭＣＩＡ）、コンパクトフラッシュ（ＣＦ）、マルチメディアカード（ＭＭＣ）、セキュアデジタル（ＳＤ）、メモリスティック、およびメモリスティックプロがある。これらのカードの１つの供給業者が、本願の譲受人であるサンディスクコーポレーションである。このようなカードが共に用いられるホストシステムとして、デジタルカメラ、携帯電話、パーソナルコンピュータ、ノートブックコンピュータ、ハンドヘルド計算装置、音声再生装置などがある。
【０００４】
不揮発性メモリデバイス自体は、不揮発性記憶素子の１つ以上のアレイで構成されている。各記憶素子は、１ビット以上のデータを格納することができる。不揮発性メモリアレイの１つの重要な特徴は、電源がもはやメモリアレイに加えられない場合でさえも、不揮

発性メモリアレイがその中でプログラムされたデータを保持することである。対照的に、揮発性メモリデバイスは、揮発性メモリアレイ中に含まれているデータを保存するために、アレイへの電源が周期的にリフレッシュされることを必要とする。不揮発性メモリの別の特徴は、不揮発性メモリアレイ内に含まれるセルがひとたびプログラムされると、そのセルは、新しいデータ値で再プログラムされる前に消去されなければならないことである。

【０００５】

メモリセル中に電荷を格納するための物理的手段は、電気的に消去可能でプログラム可能な読み出し専用メモリ（ＥＥＰＲＯＭ）のようなフローティングゲートトランジスタを用いて実施することができる。ＥＥＰＲＯＭなどのフローティングゲートデバイスに伴う１つの知られている問題は、フローティングゲートが最終的に損耗し、非常に多数回の書き込み、プログラミングおよび消去サイクルの後に故障することである。これが生じると、そのセルはもはや使用することができなくなり、アレイ中の利用可能なメモリセルのリストから取り除かれなければならない。この種類の欠陥は、「成長（ｇｒｏｗｎ）」欠陥と呼ばれる。１つの商業的に利用可能な実装例において、これらの欠陥は、欠陥セルをマップアウトし、新しく検出された欠陥メモリセルを健全なメモリセルの物理アドレスで置き換えることにより対処される。欠陥セルまたはセクタがマップアウトされて置き換えられる実装例は、１９９７年８月１９日登録のメロートラらによる米国特許第５，６５９，５５０号（特許文献１）、１９９７年９月２３日登録のハラリらによる米国特許第５，６７１，２２９号（特許文献２）、および１９９９年１月１９日登録のハラリらによる米国特許第５，８６２，０８０号（特許文献３）に記載され、これらの特許は、その全体が本願明細書において参照により明確に援用されている。

【０００６】

不揮発性メモリセルのアレイは典型的に、読み出し機能、プログラミング機能、および消去機能の効率的な実施を提供するために、複数のグループに区分される。例えば、多くの不揮発性メモリの構成において、メモリセルは、消去単位と呼ばれるより大きなグループに配列される。この消去単位は、１度に消去可能な最小数のメモリセルである。

【０００７】

消去単位のサイズは、実施されているメモリ構成に依存する。以前の不揮発性メモリにおいて、消去単位は、標準の５１２バイトのディスクドライブセクタと同サイズのブロックであった。１つの商業的形態において、各ブロックは、ユーザデータの１つのセクタに加えてそのユーザデータおよび／またはそのユーザデータが格納されたブロックに関連したいくらかのオーバーヘッドデータを格納するために十分なセルを含んでいた。セルのブロックが個々に消去可能であることを保証するために、ブロックは互いに十分に隔離される必要があった。

【０００８】

この隔離が集積回路チップ上の貴重な空間を占有したため、消去単位がかなり大きくされ、このような隔離のために必要な空間がより少なくなる別のメモリ構成が開発された。この大きなブロックシステム構成の例は、２００３年６月１７日登録のコンレーらによる米国特許第６，５８０，６３８号（特許文献４）に記載され、この特許は、２００２年７月３０日登録の米国特許第６，４２６，８９３号（特許文献５）の継続出願である。これらの特許は両方とも、その全体が本願明細書において参照により明確に援用されている。大きなブロックシステムにおいて、消去単位は、ユーザデータの読み出しおよびプログラミングを行うための基本単位（プログラミングおよび／または読み出しの単位）である個別にアドレス可能なページにさらに区分されることがよくある。１つの商業的な実施において、消去単位はメタブロックである。メタブロックは、複数の物理的消去単位で構成される仮想消去単位である。これらの複数の物理的消去単位は、並行したプログラミング操作および消去操作において用いることができるが、単一の論理ブロックとしてアドレス指定される。

【０００９】

メタブロック、または「スーパー」ブロックを形成する１つの方法が、米国特許第６，０３４，８９７号（特許文献６）に記載され、この米国特許は、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。その中に記載されているように、いくつかのデバイスを有するメモリについて、デバイスの各々の中の同じ物理ブロックは、メタブロックにグループ化される。これは、メタブロックおよび同じアドレス、すなわち複数のデバイスのうちの最初のデバイスの中のブロックのアドレスによりアドレス指定される任意のメタブロック中のすべてのブロックの形成を考慮に入れているが、いくつかの制限がある。例えば、メタブロックへのブロックのリンキングがこの固定構成においてあらかじめ決定されているので、あるブロックがだめになると、そのブロックが属するメタブロックは、その中の他のブロックが依然として機能するにもかかわらず、だめになる。従って、必要になるたびに新しいリンキングを確立する操作上のオーバーヘッドのない適応することができるメタブロック配置のためのシステムおよび方法を提供することが望ましい目標である。

【特許文献１】米国特許第５，６５９，５５０号
【特許文献２】米国特許第５，６７１，２２９号
【特許文献３】米国特許第５，８６２，０８０号
【特許文献４】米国特許第６，５８０，６３８号
【特許文献５】米国特許第６，４２６，８９３号
【特許文献６】米国特許第６，０３４，８９７号
【特許文献７】米国公開特許出願第２００３／００６５８９９号

【発明の開示】

【００１０】

本発明の様々な態様は、複合論理構造への物理メモリ構造のリンキングのための先行技術に見られるこれらの制限およびその他の制限を克服する。本発明の最初の実施形態には、フラッシュメモリにおけるメタブロック構造の形成のための手法が提示される。本発明の第１の態様によれば、メタブロックへのブロックのリンキングは、メモリの欠陥部分に対応するために更新される。欠陥マップを維持するためのシステムは、メタブロックに編成される不揮発性メモリシステム中の工場欠陥および成長欠陥の存在を追跡するための欠陥マップ構造を含む。欠陥マップを維持するための方法は、欠陥マップ構造およびメタブロックの特性を利用する。１つの実施形態において、欠陥マップ構造のサイズは、不揮発性メモリシステムの並列性と等しい。別の実施形態において、メタブロックに割り当てられない残りの消去単位は、後に使用するための予備消去単位のリスト中に保存される。

【００１１】

本発明の別の態様によれば、メタブロックへのブロックのリンキングの記録が、不揮発性メモリ中に保持される。次に、コントローラは、必要に応じてその揮発性メモリへのリンキングを読み出すことができる。記録は、欠陥に起因する再リンキングに応じて更新される。完全なリンキングの記録を不揮発性メモリ中、例えば、指定された位置に保持することができ、或いは部分的なリンキングの記録のみを保持することもできる。本発明の別の態様において、メタブロックへのブロックの初期リンキングは、例えば、ファームウェアに基づく実施を用いるアルゴリズムによって決定論的に形成される。この場合には、不揮発性メモリ中に格納されたリンキングの記録は、標準アルゴリズムからの逸脱を含みさえすればよい。

【００１２】

本発明の付加的な態様、特徴、および利点は、添付図面と共に読まれるべきである代表的な実施形態の以下の説明に含まれる。

【発明を実施するための最良の形態】

【００１３】

一般に、本発明のメモリシステムは、その主不揮発性メモリ中に、マルチブロック・メタブロック構造へのブロックのリンキングの記録を保持する。この記録は、要求に応じて不揮発性メモリに読み込むことができるマップテーブルに工場欠陥および成長欠陥の欠陥マップを含んでいる。このマップは、成長欠陥が発生してブロックが新しい位置に割り当

てられる際に更新される。メタブロックは好ましくは、いくつかのサブアレイ、すなわちプレーンの各々における１つのブロックにより形成され、この場合、グループ化は好ましくは、不良とマークされてマップ中に配置されるブロックを除いて、各プレーンにおける同じ番号のブロックで行われる。１つの実施形態において、ブロックは、ブロックを交互にするように直接的にマッピングされる。次に、欠陥ブロックを含みうるメタブロックは、対応するプレーン中の交互のブロックで再形成される。不揮発性メモリ内のメタブロックへのブロックのリンキングの完全な記録を保持するのではなく、「標準」リンキングが、例えば、システムのファームウェア中に埋め込まれたアルゴリズムに基づくことができ、欠陥によるこのアルゴリズムからの逸脱のみが格納される必要がある。欠陥がまったく存在しない標準メタブロックは、使用可能な標準メタブロックと呼ばれる。残りのメタブロックは、使用不能な標準メタブロックと呼ばれ、構成物理ブロックは予備ブロックと呼ばれる。

【００１４】

図１～図３は、ホスト１０２、２０２、３０２、コントローラ１０４、２０４、３０４、および１つの不揮発性メモリアレイ、または複数のアレイ１０３、２０３、３０３を含む代表的な現行の不揮発性メモリシステムのブロック図である。図１において、不揮発性メモリアレイ１０３およびメモリコントローラ１０４は、ホストシステム１０２に取り外し可能に接続される密封カード中にパッケージされているものとして示されている。図２において、コントローラ２０４は、ホストシステム２０２の一部として示され、不揮発性メモリアレイ２０３は、ホストシステムに取り外し可能に接続される密封カード中にパッケージされているものとして示されている。図３において、不揮発性メモリアレイ３０３は、ホストシステム３０２内の取り外しできない組み込み形記憶装置としてパッケージされているものとして示されている。

【００１５】

一般的に、不揮発性メモリアレイは、記憶素子として１つ以上の導電性フローティングゲートを含む複数のメモリセルを含む。メモリアレイは、代わりに他の長期の電子電荷記憶素子、例えば電荷トラッピング誘電体を含みうる。不揮発性多状態記憶素子、フラッシュ／ＥＥＰＲＯＭ／フローティングゲートである記憶素子、ＮＲＯＭ、ＭＮＯＳ、ＭＲＡＭ／ＦＲＡＭ、ＮＡＮＤ、ＮＯＲなどの不揮発性記憶素子、および複数の消去単位に編成される不揮発性記憶素子を含む様々な不揮発性記憶素子を用いることができる。デジタル情報、アナログ情報、または両者の組み合わせを格納する不揮発性記憶素子を用いることができる。

【００１６】

メモリセルアレイは、１ビットのデータを各素子で格納するために各記憶素子について定義された２つの電荷レベルで操作されうる。代わりに、各記憶素子について２つを超える格納状態を定義でき、その場合、１ビットを超えるデータが各記憶素子中に格納される。

【００１７】

図４ａは、１つのメモリチップが、消去単位のグループ（またはブロック）にどのようにフォーマットされうるかの例を例示し、これらのブロックは、物理プレーンおよび論理ゾーンに従って配置されている。チップをこのようにして分割することについてのいくつかの理由として、各グループ内の健全な消去単位の数を均等化することと、メモリチップの特殊目的のセクションを取っておくこととが含まれる。例えば、特別な「予約された」セクションを、システムパラメータおよびデバイスパラメータのような情報と、ゾーン整列についての情報を格納するために取っておくことができる。１つのセクションを、予備としても知られる予備消去単位、または予備ブロックを格納するために取っておくこともできる。予備に関連する情報、例えば、それらの物理アドレスは、必要時の検索用に予備リスト中に保管しておくことができる。例えば、Ｎ個のゾーンの１つに含まれる消去単位の１つに欠陥が検出された場合、利用可能な予備消去単位の１つに欠陥消去単位を再マップすることにより予備を見出すことができる。

【００１８】
ホストからのデータは一般に、論理セクタとして識別されるメモリシステムに単に提供される。例えば、ホストは、開始論理セクタおよびセクタ総数に関して記述されたデータを送信する。ホスト内部で、これらの論理セクタはより大きい単位に構造化されうるが、この情報は一般にメモリのコントローラには渡されない。（もっとも、これらのホスト構造の知識は、メモリまたはコントローラ構成体中に物理構造を持つために用いることができ、これらの構成体は、ホストがデータをどのように転送するかをこれらのホスト構造が反映するので、これらのホスト構造を反映する。例えば、メモリは一般に、論理セクタと同サイズの物理セクタに基づいてユーザデータを格納するために構造化される。）コントローラは、ホスト構造を、物理メモリ中のそれらのより効率的な格納のための物理構造を模倣する論理構成体に編成する。例えば、一般的な配置において、メモリシステムコントローラは、（ホストにより定義されるような）論理セクタを、フラッシュ形メモリのための物理的消去単位であるブロックの物理構造にサイズにおいて対応する論理ブロックにグループ化する。次に、コントローラは、論理構成体に対する物理構造の関係を、例えば、セクタアドレステーブル（ＳＡＴ）の形で保持することができ、論理－物理の関係が変化する際にこの対応を更新する。

【００１９】
より高い効率のために、メモリシステムは、より大きい構造を導入することにより並列性を増大させることがよくある。例えば、書き込み、読み出し、またはその両方が並行して行える多数の半自律的なアレイを有するメモリシステムにおいて、異なるアレイからのブロックは、「メタブロック」構造にグループ化され、コントローラは、同時に操作される多重ブロックを考慮して、データの論理ブロックを対応する論理メタブロックに形成する。１つのメタブロックは、単一のメモリチップを有する複数のプレーンまたはいくつかのメモリチップを越えて分布された１つ以上のプレーンから形成することができる。このように、読み出し／プログラミング／消去が並列に行える複数の物理ブロック上に（論理的に）連続したセクタのグループを広がらせることにより、このグループは、最小回数の非並行な読み出し操作、プログラミング操作、または消去操作で、並行して読み出し／プログラミング／または消去されうる。この配置は、比較的大量のデータの論理的に隣接するセクタの連続的な読み出しおよび書き込みにとって有利である。

【００２０】
図５～図１０の構造を最初に論じ、次に各図により示される操作を個々に説明する。図５～図１０は、図１～図３のメモリアレイの例として、４つのプレーン中の６つのブロック、総計２４の消去単位の配置を示す。便宜上、ブロックは一種の消去単位なので、消去単位はブロックと呼ばれる。この６×４ブロック構造のサイズは、以下で明らかになるように、説明の便宜上選ばれている。典型的なメモリアレイは、ここに示されているよりもずっと多くのブロックを有する（ずっと多くの行を有する）し、不揮発性メモリシステムの並列性が４よりも大きければ、示されているよりも多くのプレーンを有する。

【００２１】
各プレーンは一般に、それ自体のデータレジスタおよびプログラミング回路を有している。これは、プレーンの各々のメモリセルのブロックへの同時プログラミングを考慮している。各プレーンは、他のプレーンから半独立的に操作可能である。例示の便宜上、図５～図１０に描かれたプレーンは、６つの物理ブロック０～５を有するように示されている。任意の形態におけるブロックの実際の数は、より低いことがあるが、一般にはずっと高い。

【００２２】
多数のブロックがあるシステムにおいて、プレーンは、図４ａに示されているように、複数のゾーンに論理的に分割されうる。これらのゾーンは、複数のプレーンを横断して延びる。プレーンのゾーンへの分割は、１つのプレーンをより小さくより扱いやすいセグメントに分割するために行われる。ゾーン内部の各プレーン中のブロックは、物理ブロックアドレスの同じ隣接したセットを占めうる。実際的な設計決定が、特定のメモリシステム

において実施されたゾーンの数に影響を与えることがあるが、ゾーンの数は具体的に限定されない。本発明の様々な態様は、ゾーン構造を持つか、或いは持たないメモリにおいて実施することができ、ゾーン構造を持たないメモリについては、全体としてプレーン内部で、或いは個々のゾーンと共に実施することができる。

【００２３】

図５～図１０におけるシナリオに適用されるようなメモリシステムにより用いられる操作単位は好ましくは、メタブロックである。メタブロックは、プレーンを横断する複数の物理ブロックの仮想リンキングである。メタブロックの随意の用途としては、メタブロック内部の全ブロックのパイプライン化消去、およびメタブロックの各ブロックからの１つ以上のページのパイプライン化プログラミングおよび読み出しが含まれる。

【００２４】

図５～図１０に示されている接続された矢線のセットの各々の中の矢線により示されているように、メタブロックは、複数のプレーンを横断して延び、一般にこれらのプレーンの各々からの１つのブロックを含む。図５はメタブロックの最も単純な形態を示し、すべてのメタブロックの各物理ブロックは、同じブロックオフセットを有し、４つのプレーン０～３の全てを横断して延びている。図１０は、２つのより複雑なメタブロックを例示する。これらのメタブロックの生成は、図５～図１０の説明において論じられ、メタブロックに含まれる構成ブロックは、欠陥に遭遇すると更新される。

【００２５】

メタブロックへのブロックのリンキングについてのより一般的な状況が図４ｂに示されている。システムのメモリ部分（それぞれ図１～図３の１０３、２０３、３０３）は、チップ０～Ｚから成るように取られ、各チップは、チップ０についてはプレーン００～０Ｗのような多くのプレーンを有する。ｘにより示されている、１つのプレーン中のリンクされた各物理単位は、図４ｃに示されているように単一の物理ブロックから成ることも、図４ｄに示されているように複数の物理ブロックから成ることもできる。プレーンおよびチップを横断するだけでなく深さ方向にもブロックをリンクする図４ｄに示されている二次元メタブロックのタイプは、物理ブロックが小さい場合には特に便利なことがある。しかし、便利さは、図４ｄの配置の唯一の理由または主たる理由でさえないことがある。通常の、すなわち一次元のリンキングにおいて、主たる動機は、異なるチップ／プレーン中の複数のブロックに並行して書き込むことができるデータを扱うための機構を提供することである。同様に、メモリ設計が、１つのプレーン中の１つを超える位置への並列アクセス（読み出しおよびプログラミング）を並列して可能にするのであれば、同じプレーン内の複数のブロックをリンクすることも意味をなしうる。

【００２６】

いくつかのメタブロック・リンキングを示し、ここで、各メタブロックＭＢは、リンキング中の第１のブロックの第１の行に従って番号を付けられ、例えば、ブロック０で始まるメタブロックは、MB_0 と表示され、以下同様である。図４ｂは、各プレーンが任意のリンキングにおいて１つのブロック（そして唯一のブロック）を有し、そのリンキングがいくつかのプレーンを横断していることを示しているが、これらの状況のどちらも常にそうである必要はない。例えば、図５～図１０に示されているような４つのブロックのリンキングは、単一のチップ上の４つのプレーンまたは４つのプレーンのうちの１つのプレーンで構成されうる。

【００２７】

図４ｂに示されているように、リンキングMB_0 はそのブロックのすべてが同じ行からのものであり、MB_1 はそのブロックのほとんどが同じ行からのものであり、MB_2 は各ブロックが先行ブロックの下の行からのものである。図５～図１０の説明は、すべてのブロックが同じ行からのものであるMB_0 の場合を取り上げ、説明を容易にするために、より一般的な場合を以下で論じる。これらの様々なリンキング方法は、固定されるか、或いは以下で論じられる本発明の１つの態様に従って、アルゴリズムに基づいて決定されうる。（代わりに、当初の決定は、ランダム割り当てに基づくことができる。）より一般的な

文脈において、本発明の様々な態様が、一般に物理構造がより大きな論理複合体に形成される場合に適用でき、これらの複合体がどのように管理されるかに適用することができることが理解されよう。

【００２８】

図１０に示されている第１のメタブロックは、４つのプレーン０～３の各々の中のブロック０を含んでいる。第２のメタブロックは、プレーン０中のブロック１、プレーン１中のブロック４、プレーン２中のブロック５、およびプレーン３中のブロック４を含んでいる。第３のメタブロックは、４つのプレーン０～３の各々の中のブロック２を含んでいる。第４のメタブロックは、プレーン０中のブロック３、プレーン１中のブロック３、プレーン２中のブロック３、およびプレーン３中のブロック１を含んでいる。

【００２９】

代わりに、メタブロックは、単一のゾーン内に含まれているプレーンの数のサブセットのみを含むこともできる。この例は示してないが、可変の並列度が望まれるのであれば、有用でありうる。可変の並列度を有するシステムは、異なる並列性を有する領域間でブロックが移動することができるようにしうる。例えば、４つではなく２つのブロックを含むメタブロックは、並列性が２である（２つのプレーン）領域から、（本発明者らの４プレーンの例におけるような）並列性が４である領域に動くことができる。２つの要素を各々含む２つのメタブロックがこのようなシステム中に存在すれば、それらのメタブロックは、１つの４要素メタブロックに統合することができる。このような混合並列性システムを用いることへの１つの警告は、変化する並列性の責任を欠陥マップが甘んじて受ける必要があるということである。

【００３０】

一般に、メタブロックの使用は、より多くのブロックが並列で１度に処理されるようにすることにより、メモリシステムの並列性を増大させる。この増大した並列性は、メモリシステムの性能を大きく向上させる。メモリシステムは、様々なメタブロック内部の個々のブロックの同一性を保持する。例えば、様々なメタブロック内部の個々のブロックの同一性は、リンクされたリストとしてシステム内部のどの便利な位置においても保持することができる。

【００３１】

通常、各ページのユーザデータと共に格納されるオーバーヘッドデータは、そのページがあるプレーン、ゾーン、およびブロックを識別するのに十分な論理および／または物理アドレス、並びにそのブロック内のページのオフセットを含む。次に、ページオーバーヘッドデータのこれらのアドレスフィールドを読み取るコントローラによって、コントローラのメモリ内部にアドレスマップが生成される。これは通常、メモリの一部について１度に行われ、メモリのその部分に関するプログラミング、読み出し、または消去操作に先立って行われる。

【００３２】

図５～図１０は、メモリアレイの物理ブロック中の欠陥検出にシステムがどのように対応するかを示している。これらの欠陥は、工場欠陥または成長欠陥でありうる。この一連の図は、新たな欠陥が発生するたびに何が起こるかを示すことを意図している。例示を目的として、１～５の番号を付けられた物理ブロックは欠陥を含んでいる。

【００３３】

最初の図である図５は、欠陥がまったく存在しないメモリアレイの一部分を示す。４つのプレーンすべてを横断して延びる６本の水平な矢線により複数のメタブロックが描かれている。各メタブロックは、定義された数の物理ブロックを含む。図５～図１０において、物理ブロックの定義された数は、すべてのメタブロックについて４であり、これはプレーンの数に対応する。

【００３４】

図５は、工場欠陥または成長欠陥がまったく見られないメモリアレイの一部分を示す。この図を横断して水平に延びる黒い矢線は、６つの直線的な標準メタブロックを表してい

る。直線的な標準メタブロックは、図５に示されているように、揃ったオフセットを有する物理ブロックを含むメタブロックである。当初のリンキングは、以下でさらに説明されるように、アルゴリズムに従って確立されるか、或いは利用可能な健全なブロックの消去プールから形成される。

【００３５】

本発明の１つの実施形態において、メタブロックの１つの特徴は、（第１の物理ブロックアドレスに関連付けられた）第１の仮想ブロックアドレスが、その特定のメタブロックに関連付けられた残りの物理ブロックの位置を決定するために必要な唯一の情報であり、残りのリンクされたメタブロックはリンキング・アルゴリズムにより決定されるということである。これにより、メタブロックを格納するために必要とされるデータのサイズが低減される。第１の物理ブロックにおいて欠陥が発生すると、そのメタブロック全体が無効になり、アドレス空間からマップアウトされる。というのは、利用可能なプレーンの各々を横断して配置されたメタブロックの残りの要素を示すためのどのような機構ももはやないからである。この例が、以下の図９の説明において示されている。

【００３６】

図６は、第１の新しい欠陥が発生した後のアレイの状態を示す。この新しい欠陥は、プレーン２上のブロック１に位置する、MB_1 中の「Ａ」と表示された物理ブロックに位置する。メタブロック内部の物理ブロックが故障すると、そのメタブロックは、そのままではもはや使用できない。１つの単純な解決策は、そのメタブロックをアドレス空間からマップアウトし、二度と使用しないことである。システムが１つの単一の不良物理ブロックを有するのみであり、どのような予備ブロックも利用可能でなければ、これが利用可能な唯一の解決策である。しかし、複数の不良物理ブロックが発生すれば、この解決策はたちまちのうちに不経済になる。というのは、故障したメタブロックに関連付けられた残りの物理ブロックは、他の健全なメタブロックを生成するために使用することができるからである。図６において見出された（「Ａ」と表示された）欠陥の結果として、（ハッチングにより示されている）残る３つの健全な物理ブロックは予備になる。

【００３７】

図７は、第２の新しい欠陥が発生した後のアレイの状態を示す。この新しい欠陥は、プレーン３上のブロック３のMB_3 に位置する、「Ｂ」と表示された物理ブロックにある。図１１に示されているような欠陥マップを保持するための方法において、欠陥が生じたプレーン（プレーン３）は、どのような利用可能な予備ブロックについても調べられる。ここでは、プレーン３において物理ブロック１が利用可能である（図６においてハッチングにより示されている）。プレーン３においてブロック１が利用可能なので、メタブロックは、メタブロックの欠陥部分を置き換えるために、ブロック１を示すように更新される。すなわち、プレーン３のMB_3 における欠陥ブロック（「Ｂ」）は、リンキングにおいて、MB_1 のプレーン２のブロック「Ａ」の故障によって余剰にされたMB_1 の中のプレーン３の健全なブロックにより置き換えられる。物理ブロック故障が発生する場合、予備物理ブロックがメタブロックにリンクされて、故障した物理ブロックに取って代わる。

【００３８】

（使用されている命名法の下では、図６および図７の時点でメタブロックMB_1 はまったくない。というのは、メタブロック全体は、プレーン０におけるブロック番号に基づいて名付けられるからである。図８において生じるように、プレーン０のブロック１が新しいリンキングにおいて復帰させられなければ、名称MB_1 は欠番にされる。）

【００３９】

図８は、第３の新しい欠陥が発生した後のアレイの状態を示す。この新しい欠陥は、プレーン０上のブロック４に位置する「Ｃ」と表示された物理ブロックに位置する。欠陥ブロックがプレーン０上に位置しているので、このメタブロックは再リンクすることができない。プレーン０上に位置するこのブロックは、メタブロックの残りの要素への参照情報を含むブロックであり、プレーン０上のこのブロックがなければ、採用された規則の下でこのメタブロックを参照する方法は皆無である。従って、プレーン０上のブロック１にお

いて予備ブロックが利用可能であったとしても、その予備ブロックは、このメタブロックを完成させるために使用することはできない。その代わりに、プレーン０上のブロック１が仮想アドレス空間に再マッピングされ、新しいメタブロック中の第１のブロックとして使用され、以前のメタブロックMB_4のプレーン１、２、および３の中の現在は余剰になっているブロックを利用する。かくして、欠陥ブロック「Ａ」は、プレーン２のブロック４により置き換えることができる。図７においてプレーン３のブロック１が再マッピングされているので、これも置き換えられる必要がある。その結果、プレーン２およびプレーン３のブロック４の予備を用いて、再リンクされたMB_1が形成されうる。プレーン１について、ブロック１およびブロック４の両方において１つのブロックが利用可能である。再リンクされたMB_1を完成させるために、どちらも選ばれうるが、この例では、元はMB_4にあったブロックが用いられる。その結果、再リンクされたMB_1は、プレーン０上のブロック１、プレーン１上のブロック４、プレーン２上のブロック４、およびプレーン３上のブロック４を含む。この時点で、ただ１つの予備ブロックがプレーン１上のブロック１に残る。

【００４０】

図９は、第４の新しい欠陥が発生した後のアレイの状態を示す。この新しい欠陥は、プレーン０上のブロック５に位置する「Ｄ」と表示された物理ブロックに位置する。この時点でプレーン０に位置する予備ブロックは皆無なので、このメタブロックは再リンクすることができない。結果として、残りの物理ブロックは、予備物理ブロックプールに加えられる。これは、プレーン１上のブロック５、プレーン２上のブロック５、およびプレーン３上のブロック５を含む。この時点で、示されているメモリアレイにおいて、４つの予備ブロックが利用可能である。

【００４１】

図１０は、第５の新しい欠陥が発生した後のアレイの状態を示す。この新しい欠陥は、プレーン２上のブロック４に位置する「Ｅ」と表示された物理ブロックに位置する。同じプレーン上で１つの予備ブロック（プレーン２上のブロック５）が利用可能なので、メタブロックMB_1は、その予備ブロックを用いて再リンクされる。

【００４２】

図１０において、行１、３、４、および５に対応する６つの本来のメタブロックのうちの４つが不良ブロックを有しているが、再リンキングは、４つの健全なメタブロックという結果になっていることに留意されたい。米国特許第６，０３４，８９７号（特許文献６）の背景技術の欄に記載されているような固定リンキングの下では、これらの欠陥は、示されている６つのメタブロックのうちの４つが失われるという結果になった。このように、メモリが老化して欠陥が生じるにつれて、より大きいデータ容量がメモリにより維持される。

【００４３】

図１１は、論理－物理マッピングテーブル１１０１を用いてメタブロックの構成要素がどのようにマッピングされうるかを例示する。ＬＢＮ［０］と関連付けられたメモリ１１０３（またはその構造を用いる実施形態のメモリゾーン）中のメタブロックが、ハッチングでマークされたブロックにより示されている。（この例における）メモリの４つのプレーンは、１つを超えるチップを横断して再度分布させられる。ＬＢＮ［０］内に含まれるＰＢＮエントリは、ハッチングによりマークされ、メタブロックの部分であるブロックの各々の物理アドレスを参照する。プレーン０に含まれているメタブロックの第１の要素は、他のメタブロック要素にアクセスするために使用される。図５～図１０に関して説明されているように、プレーン０中のメタブロック要素が不良であることが見出されれば、そのメタブロック全体は不良としてマークされなければならない。というのは、それらはプレーン０上の要素を介してメタブロックの残りとリンクされるからである。

【００４４】

マッピングテーブル１１０１は、メタブロックを形成するためのブロックのリンキングの記録の１つの例である。メタブロック構成要素の適応性を考慮する１つの手法は、書き

込み処理用にメタブロックが要求されるたびに、利用可能な健全なブロックのプールからメタブロックを動的に形成することである。この手法はメタブロック構造のための適応性を考慮するが、この手法は、ブロックが必要とされるたびにリンキングが再確立されることを要求し、さらに、この手法は、このプールを決定するために各プレーンがアクセスされることを要求する。本発明の主要な態様の１つによれば、この記録は、不揮発性メモリ中の１つの位置に格納される。次に、この記録は、アドレス変換において使用するための要求があり次第、フラッシュメモリからコントローラ上の揮発性メモリ（または代わりの不揮発性メモリ）に読み込まれる。

【００４５】

実施形態の１つのセットにおいて、リンキング情報の記録は、ユーザデータが格納される通常の物理ブロック外部のどこかのような、特別な不揮発性メモリ位置に保持される。例えば、この記録は、図４ａのプレーン０中の予約領域の１つ、または米国公開特許出願第２００３／００６５８９９号（特許文献７）に記載されているような配置中のシステムゾーンのいずれかに保持されることができ、この米国公開特許出願は、本願明細書において参照により援用されている。実施形態の別のセットにおいて、リンキング情報は、書き込まれた、または部分的に書き込まれたブロックのためのユーザデータセクタ／制御データセクタのヘッダ領域中に保持される。消去状態にあるブロックについて、リンキング情報は、不揮発性メモリ中の特別な制御データ領域に格納される。

【００４６】

このようなテーブルの使用により、メタブロックへのブロックのリンキングは、記録を不揮発性メモリ中に保持することによって、欠陥に対応するように更新することができ、このようなテーブルは必要時に容易にアクセスでき、従って背景技術の欄で論じられた先行技術の制限を克服する。

【００４７】

本発明の欠陥マッピングの態様が図１２に例示され、この図は、プレーン１上の１つの不良ブロックを示している。この不良ブロックには「Ｘ」印が付けてある。このブロックが欠陥であることが分かる前に、点線矢線１２０１により示されているように、メタブロックＬＢＮ［０］の第２の要素がそのブロックを指摘した。予備ブロックの集まり１２０３は、ハッチングにより示されている同じプレーン上で利用可能な１つの予備を有している。（代わりに、図５～図１０の処理におけるように、エラーを発生させた別のリンキングから１つの予備がプレーン１中で利用可能であれば、その予備は、予備の集まり１２０３からのブロックの代わりに用いてもよい。予備ＰＢＮ’の物理ブロック番号（ＰＢＮ）は、故障したブロックのＰＢＮに取って代わる。結果として、この予備は、矢線１２０２により示されているように、メタブロックの一部になる。

【００４８】

図１３は、図１１におけるマップ構造と同様なマップ構造の例を示す。図１３におけるマップ構造は、プレーン２上の「Ｘ」印が付けられたブロック上に示されているように、不良ブロック、おそらくは成長欠陥を示している。プレーン２上の消去プール１３０３中の予備ブロックの集まりが予備をまったく持っていないという事実により示されているように、この不良ブロックと同じプレーン上では予備を利用することができない。従って、プレーン２上の不良ブロックの予備を示すように再指向する代わりに、３本の下方向を示している矢線により示されているように、メタブロックの残り（プレーン０、１、および３上のブロック）に関連付けられた物理ブロックが予備プールに加えられる。

【００４９】

図１４は、図１１～図１３のデータ構造を用いて欠陥マップを保持ための方法の例を示す。欠陥が検出されるか、或いは発生すれば、その欠陥の位置は、物理位置または物理ブロック番号（ＰＢＮ）を検索することによって決定される。欠陥ブロックの物理位置と同じプレーン中で予備ブロックが利用可能であれば、その欠陥に関連付けられメタブロックは、欠陥ブロックを置き換えるために予備に向けられ、その後、欠陥マップは更新される。このシナリオは図１２に示されている。さもなければ、欠陥ブロックと同じプレーン中

に予備ブロックがまったく残っていなければ、メタブロックの残りの健全なブロックは、予備プールに向けられる。このシナリオは図１３に示されている。

【００５０】

前に指摘されたように、本発明の原則的な態様の１つは、コントローラが、リンキングテーブルの記録を不揮発性メモリ中に保持することである。この記録は、図１１～図１３の論理－物理マッピングテーブル１１０１のような、任意の物理アドレスに対応する各メタブロックを形成するブロックの完全なリストである。一方、メタブロックへの物理ブロックの標準リンキングは、決定論的アルゴリズムに基づくことができ、不揮発性メモリ中に保持される記録は、リンキング規則からのいずれかの逸脱を含みさえすればよい。欠陥のためにリンキングが更新されて、その結果、標準リンキングから逸脱すると、記録はそれに対応して更新される。

【００５１】

説明された、物理ブロックをメタブロックにリンクするために用いられる機構は、不良物理ブロックのどのような分布についてもメタブロックの数を最大化し、システムの寿命の間に物理ブロックが故障した場合に、新しいメタブロックが生成されるように設計される。前述したように、メタブロック番号は、メタブロックに組み込まれた第１のプレーン中の物理ブロックのアドレスから定義される。

【００５２】

「標準」メタブロックは、物理ブロックアドレスがメタブロック番号の決定論的機能である物理ブロックを含む。例えば、これは、図４ｂにおけるメタブロックMB_0の配置とすることができ、この配置は、図５～図１０の例についても用いられ、その場合、標準メタブロックのすべてのブロックは同じ行にある。この配置は、MB_i＝（ｉ，ｉ，ｉ，ｉ）と表すことができ、ここでカッコ内のｎ番目のエントリは、ｎ番目のプレーン中のブロックが属する行を表す。

【００５３】

別の例では、図４ｂにおけるメタブロックMB_2の配置におけるように、物理ブロックアドレスは、メタブロック番号にその物理ブロックが属するプレーンの連続番号に等しいオフセットを加えたものにより与えられる。すなわち、メタブロックｉは、ブロックｉ＋（プレーン番号）で構成される。４ブロックのメタブロックについて、これは、MB_i＝（ｉ，ｉ＋１，ｉ＋２，ｉ＋３）と表すことができる。この構造は図１５に示され、アルゴリズムに基づくリンキングテーブルを論ずるために用いられる。より一般的には、アルゴリズムに基づく、決定論的リンキングは、４ブロックのメタブロックの場合について、

MB_i＝（ｉ，f_1（ｉ，プレーン番号），f_2（ｉ，プレーン番号），
f_3（ｉ，プレーン番号））

と記述することができ、他の数については自明なやり方に従って拡張される。この数式において、ｉは、行番号であり、f_1、f_2、およびf_3は、行およびプレーン番号のある種の関数である。取り決めによりメタブロック番号はプレーン０の行番号と同じなので、MB_i中の第１のエントリはｉである。図５～図１０の例については、f_1＝f_2＝f_3＝ｉであるのに対して、図１５については、f_1＝f_2＝f_3＝ｉ＋（プレーン番号）である。アルゴリズムに基づくリンキングについての再リンキング処理の議論は、図５～図１０の処理の短縮バージョンを経るが、図１５の標準リンキングに基づく。コントローラは、標準リンキングを決定するためにファームウェアに基づく実施を用いることができる。

【００５４】

物理ブロックは、標準メタブロックの考えうる数を最大化する規則に従ってメタブロックにリンクされる。メタブロックがリンクされている時に、必要とされる物理ブロックが利用可能であれば、標準ブロックが生成される。標準メタブロックの形成に必要とされるいずれかの物理ブロックが不良であれば、メタブロック番号は未使用として分類される。この場合、その結果として未リンクのまま残される他の使用可能な物理ブロックは、図１５におけるメタブロックｎ＋１について示されているように、予備として分類される。

【００５５】
本発明の別の態様は、カード上の欠陥のパターンに基づく標準リンキングパターンの最適化、または使用可能な標準メタブロックの数を別の方法で最大化することである。カードがフォーマット（または再フォーマット）されている時に、不良ブロックのパターンを走査することができ、「標準」リンキングパターンについての決定を、それらのパターンが互いにできるだけ一致するように行うことができる。多くの場合、標準リンキングのためのアルゴリズムを不良ブロックパターンに一致させることにより、必要な再リンクされたメタブロックおよびメター物理アドレス変換オーバーヘッドが低減され、それによって、（少なくとも当初の）リンキングが最適化される。この例が、図１６および図１７に示されている。

【００５６】
図１６および図１７は、プレーン０～３の各々から１つずつの４つの不良物理ブロックが対角線状に並んでいる例を検討する。不良ブロックの対角線状パターンの場合、図１６に示されているように、標準ブロックにとって最良のパターンも対角線状である。結果として、この例において、再リンクされたブロックは皆無であり、その結果として、メタブロックのうちでリンキングテーブルを介した変換を必要とするものは皆無であり、使用可能なメタブロックの数は増大する。図１７は、標準リンキングが不良ブロックパターンに一致するのではなく直線横断する場合の同等の状況を示す。標準の方法が直線横断の場合の例において、３つのメタブロック（ｎ＋２，ｎ＋３，およびｎ＋４）が再リンクされる必要がある。

【００５７】
標準ブロックパターンは、すべてのカード、すべての再フォーマット、またはその両方について一意でありうる。これにより、すべてのカードが異なるブロックを横断するデータ・スクランブリングの一意の方法を有する。その結果、標準メタブロックは同じ規則を用いて形成されるが、必ずしも同じ構成を有する必要はない。別の実施形態において、ホストは、メタブロック・リンキング規則についての規則およびコードを提供することを必要とすることがあり、従って、アルゴリズムはシステムに固有ではない。

【００５８】
初期フォーマット処理の間にメタブロックがリンクされている場合、ブロックアドレス空間の始端近く、或いはチップまたはダイ境界近くのいくつかの物理ブロックは、図２０に示されているように、標準メタブロックにリンクされない。これらの物理ブロックは、予備として分類される。予備物理ブロックは、予備物理ブロックリストに記録され、その後、図１９において説明されるように、再リンクされたメタブロックの形成において用いられる。

【００５９】
標準ブロックとして存在することができないメタブロックは、再リンクされたメタブロックとして構成される。再リンクされたメタブロックは、メモリシステムの初期フォーマット処理の間に、或いは既存のメタブロック内の物理ブロックの故障に応答して、形成される。図１９におけるメタブロックｎ＋４は、プレーン２中の必要とされる物理ブロックが不良なので、標準メタブロックとしてリンクされない。この場合、プレーン２中の予備物理ブロックで置き換えることにより再リンクされたメタブロックが生成される。不良物理ブロックの代わりに再リンクされる予備物理ブロックは好ましくは、メタブロック内でのプログラミング並列性を維持するために、不良ブロックと同じプレーン中に位置する。

【００６０】
好ましい実施形態において、別のダイまたはチップへ進む再リンキングは付加的なダイ／チップの読み出しまたはプログラミングを起動することがあるので、同じダイまたはチップ内のみのブロックを再リンクすることが好ましい。例えば、メタブロックが２つのプレーンをそれぞれ有する２つのチップにまたがる４プレーン・メタブロックの代表的な場合において、データは、４つのプレーン全部にプログラムされ（またはそれから読み取られ）、これは、２チップ読み出し／プログラミングを並列で必要とする。（これは１００

％並列ではないかもしれないが、ページの読み出し、ブロックのプログラミング、および消去が並列で行なわれてもよいようにパイプライン化されているが、データ転送が並列にならないようにデータバスが共用されている）。この場合、物理ブロックが別の第３のチップから取られるようにメタブロックが再リンクされれば、他の２つと同時にそのチップにアクセスすることができないかもしれず、操作時間が二倍になりうる。

【００６１】

プレーン０中の不良物理ブロックがプレーン０の予備ブロックにより置き換えられれば、メタブロック番号は、図２０においてメタブロックｎ＋１について示されているように、予備ブロックの番号により定義される。再リンクされたメタブロックは、その後に万一物理ブロック故障が発生したら、さらに再リンクされうる。標準メタブロックとしてリンクされない、アドレス空間の最上部に近いメタブロックは、再リンクされたメタブロックの形を取ることができる。

【００６２】

図４ｄに関して前に指摘したように、プレーンおよびチップを横断するだけでなく任意のプレーン内部の深さ方向でもブロックをリンクする二次元メタブロックを用いる構成もある。図２１は、二次元の場合における再リンキングの例を示す。図２１において、「２－Ｄメタブロック」は、２＊４メタブロックの例である。８つのブロックの全てが並列でアクセスされうる場合、性能における最大の利点が達成される。１－Ｄリンキング方法の主なリンクキングおよび再リンキング原理も当てはまるが、いくつかの拡張を伴う。図に見られるように、プレーン内部の物理ブロックのいずれかの中の不良物理ブロックは、メタブロックｎについて示されているように、使用されていないメタブロックに至る。２－Ｄメタブロックが再リンクされる場合、プレーン２中のメタブロックｎ＋２の上位ブロックについて言えば、任意のプレーン中の不良物理ブロックのみが置き換えられる必要があり、例えば、プレーン２中のメタブロックｎ＋２の上位ブロックが再リンクされるのに対して、プレーン２中のメタブロックｎ＋２の下位ブロックはそうではない。

【００６３】

標準リンキングからの逸脱のみが不揮発性メモリ中の記録中に保持される実施形態において、メタブロック・リンキングについての管理データ構造は、フラッシュメモリ中の１つ以上の専用ブロック・リンケージ管理ブロック中の２種類のセクタ内に含まれてもよい。この配置は、完全リンキングの節減に関して、必要とされる情報の格納がより少ないという利点を有する。メモリシステム中のすべての再リンクされたメタブロックのメタブロック番号は、リンクテーブルセクタのセット中のリンクテーブルに含まれる。標準メタブロックのためのどのようなエントリもリンクテーブル中に格納される必要はない。というのは、それらの物理ブロックアドレスは、メタブロック番号から決定論的に生成されるからである。ブロックが基準であるかどうか、およびすべてのブロックが標準のやり方でリンクされているかどうか、或いはそのブロックが再リンクされ、リンクテーブル（ＬＴ）を介した付加的なブロックアドレス変換を必要とするかどうかを示す、ＲＦ（再リンクされた）フラグを、メタブロックの属性として用いることができる。ＲＦフラグは、ＬＴを分析することにより決定でき、次にメタブロックについての属性としてＳＲＡＭ中に格納するか、或いは主不揮発性メモリ中の制御データ構造に格納することができる。代わりに、再リンクされたセクタのリストを、どのメタブロックが再リンクされているかを決定するために検索することができるリスト中に保持することができる。リンクテーブル中のエントリはメタブロック番号により配列することができ、各エントリは、メタブロック番号のためのフィールドおよび各リンクされた物理ブロックのブロックアドレスのためのフィールドを有する。

【００６４】

リンクテーブル構造の拡張において、標準リンキングを確立する基本アルゴリズムからの逸脱は、それ自体が規則のセットに基づくことができ、例えば、非標準リンキングは、単一のプレーンに限定されるか、或いはブロックがその当初の行からどれほど遠くまで再リンクすることができるかを限定する。再リンキングがアルゴリズムのすべてに対する例

外である場合にのみ、再リンキングは、不揮発性メモリ中に明示的に保持される必要がある。

【００６５】

リンクテーブルの特定の例、およびそのリンクテーブルが含む情報がどのように圧縮されるかを、図２２および表１に関して説明する。リンクテーブルセクタは、カード内で再リンクされたブロックについての情報を保持している。カードのサイズおよび各ダイ内の不良ブロックの数に応じて、１つを超えるリンクテーブルセクタが存在することが考えられ、これはブートブロックに格納されうる。前に説明されたように、メタブロックはリンキングのための標準の方法を有し、ブロックの１つが不良であれば、そのメタブロックは、同じプレーン中の予備物理ブロックと再リンクされうる。リンクテーブルセクタは、再リンキングにおいてどのブロックが用いられるかについての情報を格納する。図２２は、メタブロックの再リンキングの別の例である。

【００６６】

図２２に示されているメタブロックのうち、メタブロック０のみが、ここでは８つのプレーンを直線横断するように取られた標準リンキングを有する。この図は、以下のいくつかのブロック・リンキングの状況を示す。

Ａ．メタブロック０は、再リンキングをまったく持たない標準メタブロックであり、従ってリンクテーブルセクタには現れない。

Ｂ．メタブロック１は、プレーン０* およびプレーン１に２つの不良ブロックを有する。

Ｃ．メタブロック４は、プレーン０* およびプレーン５に２つの不良ブロックを有する。

Ｄ．メタブロック５は、プレーン１、３、および４に３つの不良ブロックを有する。

Ｅ．メタブロック７は、プレーン０* に１つの不良ブロックを有する。

Ｆ．プレーン０およびプレーン７中のブロックは本来のメタブロックからのものであるが、メタブロック９は、既存の予備物理ブロックから完全に再リンクされている。

*マークされた場合において、再リンキングは、プレーン０中の不良ブロックに起因する。この場合、物理アドレスは、本来のメタブロックが存在した位置を参照する。図２２の例における不良ブロックの数が多いため、メタブロック２、３、６、または７は皆無である。

【００６７】

図２２に示されているブロックの再リンキングのために用いられる圧縮アルゴリズムは、表１のように定義される。

【表１】

図中のメタブロック			アイテム数を一致させるための繰り返し値		エントリ中の総バイト数
	メタブロック番号	エントリ中の再リンクされたアイテム数	プレーン	物理アドレス	
	２バイト	１バイト	１バイト	２バイト	
1	0x0001	0x02	0x00	0x0002	
			0x01	0x0005	9
4	0x0004	0x02	0x00	0x0003	
			0x05	0x0001	9
5	0x0005	0x03	0x01	0x0004	
			0x03	0x0004	
			0x04	0x0007	12
7	0x0007	0X01	0x00	0x0006	6
9	0x0009	0x06	0x01	0x0008	
			0x02	0x0007	
			0x03	0x0007	
			0x04	0x0008	
			0x05	0x0008	
			0x06	0x0007	21

どのリンクテーブルのエントリのサイズも、以下のように定義される。

バイト数＝３＋（３×再リンクされたブロックの数）

例えば、メタブロック１について、最初の３バイトは、メタブロック１を識別し、そしてメタブロック１が２つの再リンキングを有することを識別し、これらの再リンキングの各々を指定するために３バイトが割り当てられる。再リンキングは、再リンクされたブロックが８つのプレーンのどれに対応するか（１バイト）およびそのブロックが再リンクされた行番号により指定される。

【００６８】

リンクテーブル中のエントリのフルセットは、複数のリンクテーブルセクタを横断して広げることができる。リンクテーブルセクタは、それをブロックのリンケージ管理ブロック中の次の利用可能なセクタ位置に書き込むことにより生成または修正される。従って、リンクテーブルセクタの多重コピーが存在することができ、最新のバージョンのみが有効である。ブロックのリンケージ管理ブロック中の有効なリンクテーブルセクタの位置のインデックスは、ブロックのリンケージ管理ブロック中の予備物理ブロックリストセクタ中のフィールドに含めることができ、これは以下で説明される。有効な各リンクテーブルセクタ中の第１のエントリに関するメタブロック番号は、どのようなメタブロック番号についても関連のリンクテーブルセクタが容易に識別することができるようにするために、コントローラＲＡＭ中のリストに保持することができる。付加的なエントリが、最大数のエントリをすでに含んでいるリンクテーブルセクタに加えられなければならないのであれば、そのセクタは、それぞれがエントリの半分を有する２つの新しいセクタに分割され、新しいエントリは、適切な新しいセクタに挿入される。

【００６９】

すべての予備物理ブロックのブロックアドレスは、予備物理ブロックリストセクタ中の未配列リスト中のエントリとして保持される。実施形態によっては、チップあたりのプレーンあたりの予備ブロックの最大数を限定することが都合がよいことがある。代表的な実施形態において、これは１６に限定され、ほとんどの場合において、この数は十分に大きいはずである。１つのプレーン中に１６を超える予備ブロックがあれば、それらは「失わ

れる」。

【００７０】

予備になるブロックは、メタブロックへのフラッシュメモリの初期構成の間に標準メタブロックを生成することができないのか、或いは標準または再リンクされたメタブロックにリンクされた物理ブロックが故障したので、新しいエントリとして予備物理ブロックリストの最後に付加される。再リンクされたメタブロックを生成するために予備ブロックが必要とされる場合、予備物理ブロックリストの冒頭に最も近い関連するプレーン中の第１のブロックが使われる。予備物理ブロックリストセクタは、有効なリンクテーブルセクタ位置へのインデックスを有するフィールドも含む。

【００７１】

メタブロック再リンク操作が行われ、リンクテーブルセクタが生成または修正される場合、予備物理ブロックリストおよびリンクテーブルセクタのインデックスフィールド双方は、予備物理ブロックリストセクタにおいて修正されうる。従って、有効な予備物理ブロックリストセクタは、ブロックのリンケージ管理ブロックに最後に書き込まれたセクタである。ブロックのリンケージ管理ブロックが満杯の場合には、このブロックは、制御書き込み操作の間に、すべての有効セクタを新しいブロック位置に再書き込みすることにより圧縮でき、満杯のブロックはその後消去される。

【００７２】

予備物理ブロックセクタの節減を簡略化し、代替のブロックを探す場合に必要な検索量を限定するために、予備物理ブロックリストは、チップ、ダイ、およびプレーンにより格納される。必要とされるセクタの実際の数は、メモリデバイスのサイズ、デバイスの数、および使用されるインタリーブな方法によって決まる。前に指摘されたように、プレーンあたり特定数の物理ブロックのみを格納することが便利なことがある。保持される数は、ブロックの予期される故障率およびプレーンあたりのブロックの数によって決まる。これは、各予備物理ブロックリストセクタが、いくつかのプレーンのための予備物理ブロックを保持することを意味する。ファームウェアは、ロードすべき正しいセクタを計算し、次に正しい値をロードするためにそのセクタを指し示す。

【００７３】

再リンキングまたは別の不良ブロックの発見に起因して、予備物理ブロックリストの数が変わる場合、新しい予備物理ブロックリストセクタが書き込まれる。これは、同じ物理領域をカバーする２つの予備物理ブロックリストセクタを意味する。ブートセクタは、有効セクタのインデックスを含む予備物理ブロックリストインデックスを保持する。１つの予備物理ブロックリストによりカバーされる領域に予備物理ブロックがまったく存在しなければ、そのセクタはブートブロックには存在しない。

【００７４】

図２２を再度参照すると、この図は、８通りにインタリーブされたカードを参照している。従って、すべての予備物理ブロックリスト情報を保持するために少なくとも８つのアレイがある必要がある。カード上のあらゆるその後のダイスについて、付加的な８つのアレイが必要とされる。図２２に対応する予備物理ブロックリストを表２に示す。

【表２】

プレーン			
0	0x08		
1			
2	0x01	0x04	0x08
3	0x01	0x08	0x09
4	0x01	0x04	0x05
5	0x04	0x07	0x09
6	0x01	0x04	0x08
7	0x01	0x04	0x07

リンキングテーブルと同様に、予備物理ブロックリストは、要求時に揮発性ＲＡＭにキャッシュすることができるテーブルとして不揮発性メモリに格納されてもよい。

【００７５】

本発明の様々な態様を、ファームウェアに基づく実施の観点から説明してきたが、本発明の態様は、ハードウェアに基づく実施においても実行することができる。コントローラ上、好ましくはメモリチップ自体の上のハードウェアエンジンが、再リンキングを行うことができる。再リンキング情報は、チップ自体上の特別なメモリ位置に格納されてもよい。次に、コントローラは、すべてのブロックを標準として扱うことができ、エンジンは、ブロックアドレスを真の内部アドレスに変換する。メモリは、ホスト（またはコントローラ）には、より小さいブロック構造またはどのような再リンキングの詳細も知る必要なく、標準ブロックのみを有する１つのメモリとして、或いは大きな物理ブロックを有する１つのメモリとして見える。

【００７６】

本発明の様々な態様の特定の例を説明してきたが、本発明は、添付の特許請求の範囲の全範囲内においてその権利が保護されるべきであることが理解されよう。

【図面の簡単な説明】

【００７７】

【図１】不揮発性メモリアレイおよびメモリコントローラが、ホストシステムに取り外し可能に接続される密封カード中にパッケージされている、現行の不揮発性メモリシステムのブロック図である。

【図２】不揮発性メモリアレイが、ホストシステムに取り外し可能に接続される密封カード中にパッケージされている、現行の不揮発性メモリシステムのブロック図である。

【図３】不揮発性メモリアレイが、ホストシステム内の取り外しできない組み込み形記憶装置としてパッケージされている、現行の不揮発性メモリシステムのブロック図である。

【図４ａ】プレーン、ゾーン、および予備を示すチップ形式の例を例示する。

【図４ｂ】メタブロックへの物理ブロックの様々なリンキングを例示する。

【図４ｃ】メタブロックへの物理ブロックの様々なリンキングを例示する。

【図４ｄ】メタブロックへの物理ブロックの様々なリンキングを例示する。

【図５】メモリアレイの一部分で生じる一連の成長欠陥にシステムがどのように対応することができるかを例示する。

【図６】メモリアレイの一部分で生じる一連の成長欠陥にシステムがどのように対応することができるかを例示する。

【図７】メモリアレイの一部分で生じる一連の成長欠陥にシステムがどのように対応することができるかを例示する。

【図８】メモリアレイの一部分で生じる一連の成長欠陥にシステムがどのように対応することができるかを例示する。
【図９】メモリアレイの一部分で生じる一連の成長欠陥にシステムがどのように対応することができるかを例示する。
【図１０】メモリアレイの一部分で生じる一連の成長欠陥にシステムがどのように対応することができるかを例示する。
【図１１】消去単位上の成長欠陥の検出に応答した、メタブロックの仮想構造の構成要素のマッピングの例を例示する。
【図１２】欠陥マップを保持するためのシステムの例を例示し、予備消去単位は、成長欠陥を有する消去単位と同じプレーン中では利用できない。
【図１３】欠陥マップを保持するための方法の例を例示する。
【図１４】図１１～図１３のデータ構造を用いて欠陥マップを保持するための方法の例を例示する。
【図１５】アルゴリズムに基づく物理ブロックのリンキングを例示する。
【図１６】アルゴリズムに基づく物理ブロックのリンキングを例示する。
【図１７】アルゴリズムに基づく物理ブロックのリンキングを例示する。
【図１８】アルゴリズムに基づく物理ブロックのリンキングを例示する。
【図１９】物理ブロックの再リンキングを例示する。
【図２０】物理ブロックの再リンキングを例示する。
【図２１】二次元の場合における再リンキングの例を示す。
【図２２】物理ブロックの再リンキングを例示する。

【図１】

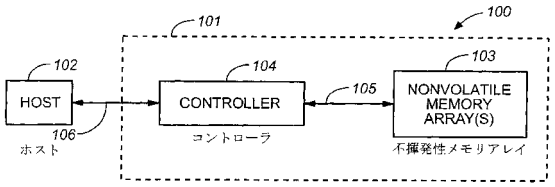

FIG. 1

【図２】

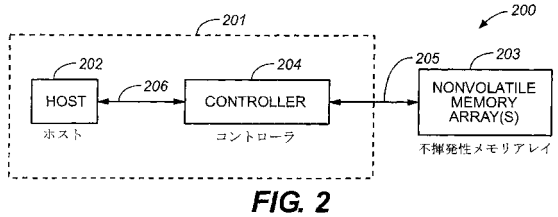

FIG. 2

【図３】

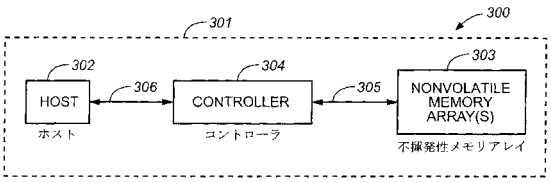

FIG. 3

【図４ａ】

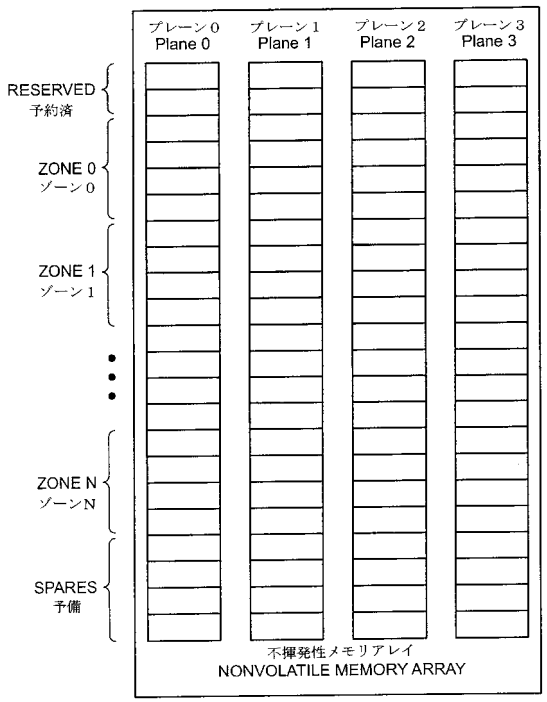

FIG. 4a

【図４ｂ】

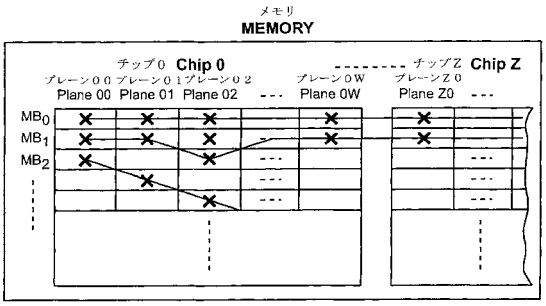

FIG. 4b

【図４ｃ】

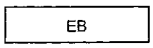

FIG. 4c

【図４ｄ】

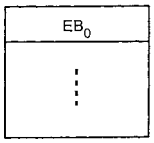

FIG. 4d

【図５】

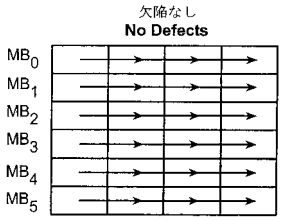

FIG. 5

【図６】

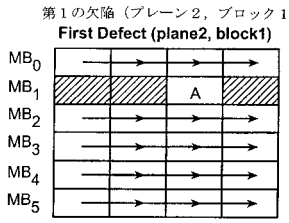

FIG. 6

【図７】

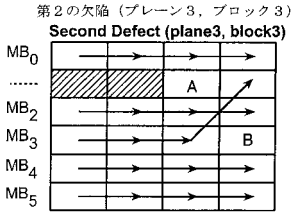

FIG. 7

【図８】

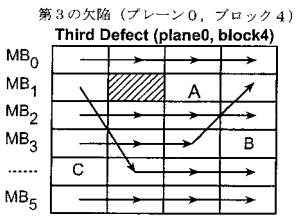

FIG. 8

【図９】

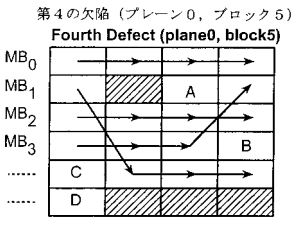

FIG. 9

【図１０】

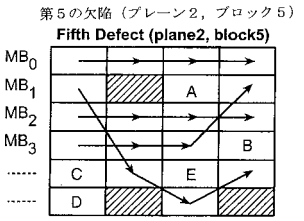

FIG. 10

【図１１】

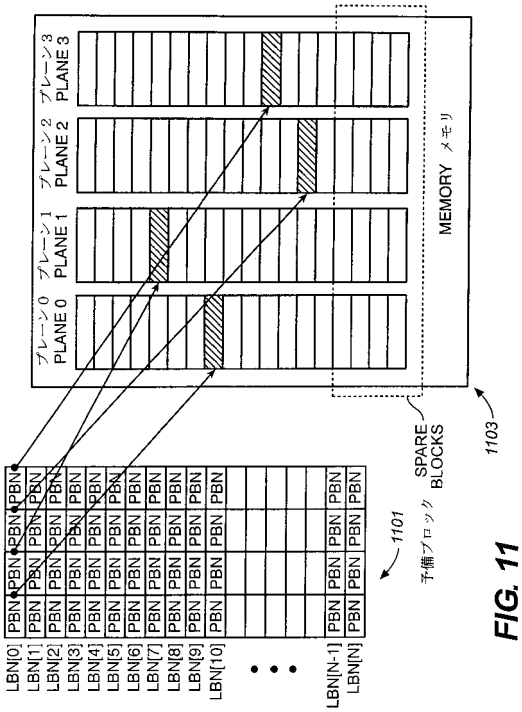

FIG. 11

【図１２】

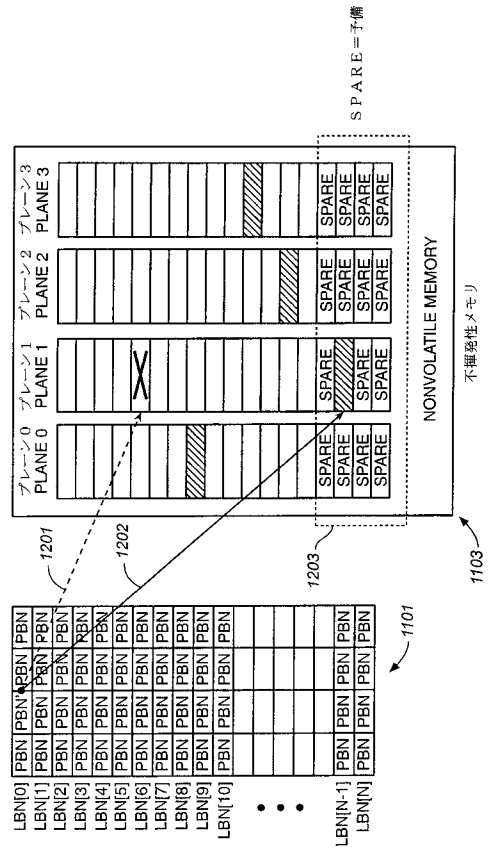

FIG. 12

【図１３】

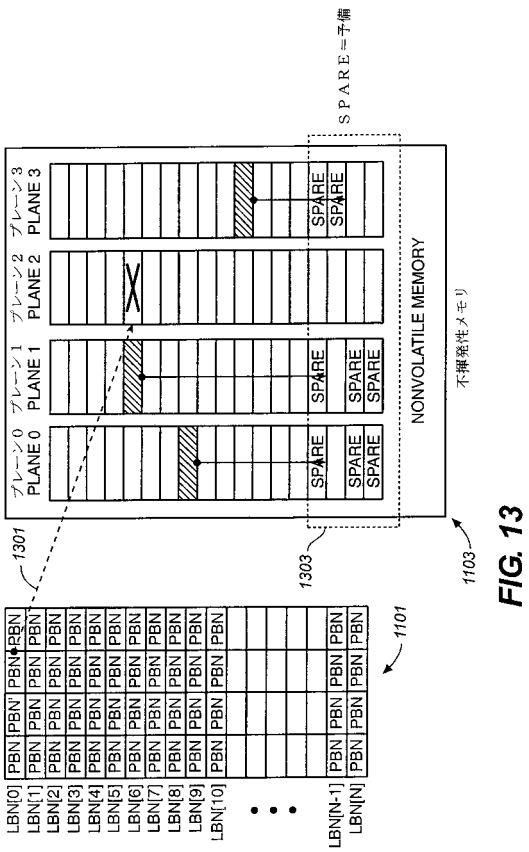

FIG. 13

【図１４】

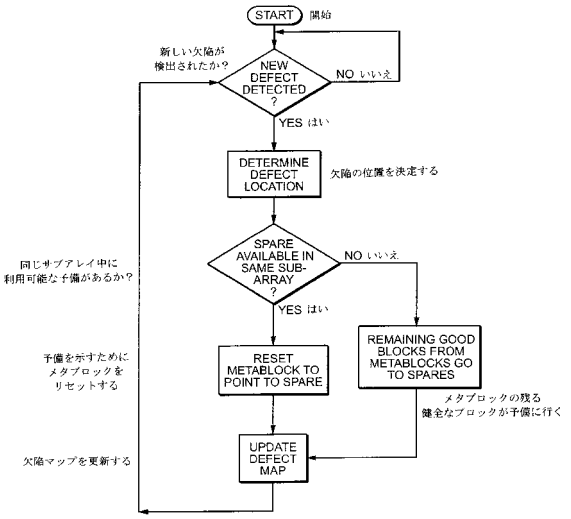

FIG. 14

【図１５】

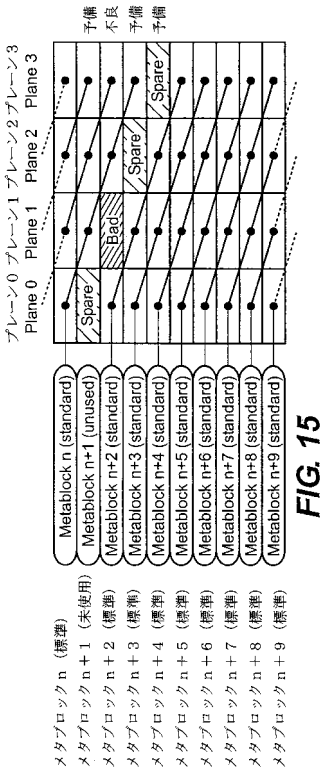

FIG. 15

【図１６】

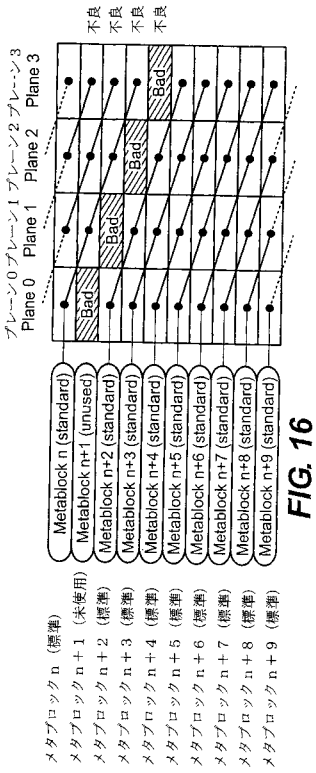

FIG. 16

【図１７】

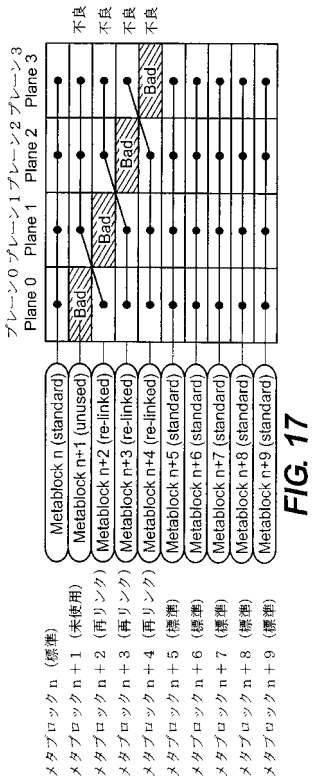

FIG. 17

【図１８】

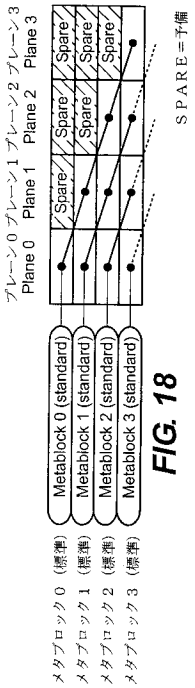

FIG. 18

【図１９】

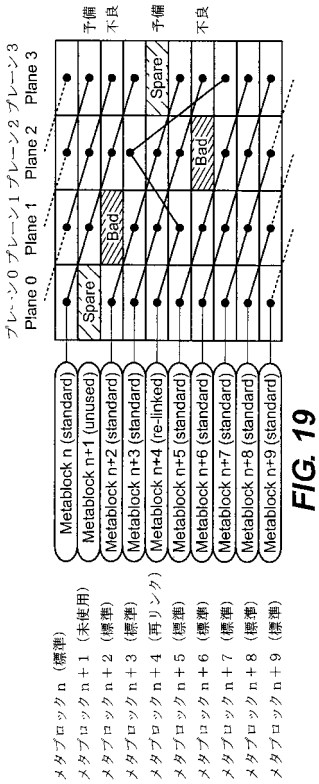

FIG. 19

【図２０】

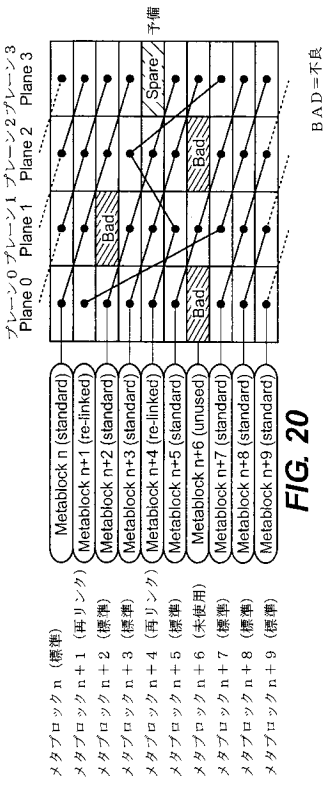

FIG. 20

【図２１】

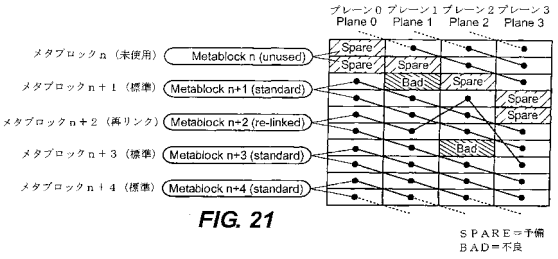

FIG. 21

【図２２】

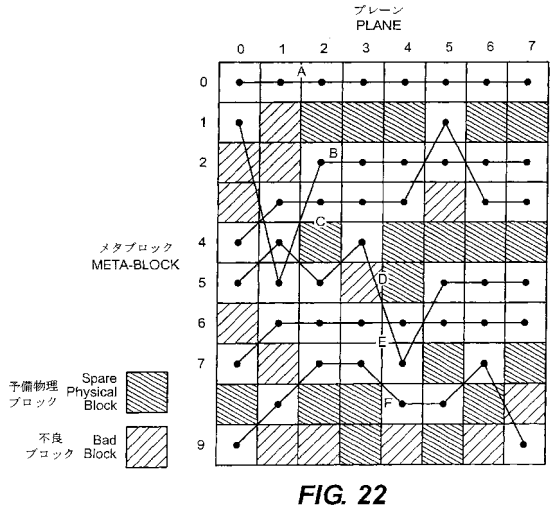

FIG. 22

フロントページの続き

(72)発明者　ゴロベッツ，セルゲイ　エイ．
イギリス連邦共和国、ＥＨ９　３ＥＳ、エディンバラ　ミッドロジアン、ブラックフォード　アベニュー　９２、１エフ１
(72)発明者　ベネット，アラン　ディー．
イギリス連邦共和国、ＥＨ１３　０ＢＱ、エディンバラ　ミッドロジアン、ソーバーン　ロード　２

審査官　小林　紀和

(56)参考文献　特開２００４－２６５１６２（ＪＰ，Ａ）
特開２００３－０１５９４６（ＪＰ，Ａ）
特開２０００－３３０８７６（ＪＰ，Ａ）
特開２００３－０５８４１７（ＪＰ，Ａ）
特開２００３－０１５９４７（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
G11C 29/04
G06F 12/16
G11C 16/02
G11C 16/06