

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第4046877号  
(P4046877)

(45) 発行日 平成20年2月13日(2008.2.13)

(24) 登録日 平成19年11月30日(2007.11.30)

(51) Int.Cl.

F I

G 1 1 C 16/02 (2006.01)  
 G 0 6 F 12/02 (2006.01)  
 G 0 6 F 12/16 (2006.01)  
 G 0 6 F 15/78 (2006.01)

G 1 1 C 17/00 6 0 1 C  
 G 0 6 F 12/02 5 1 0 A  
 G 0 6 F 12/16 3 1 0 R  
 G 0 6 F 15/78 5 1 0 F

請求項の数 6 (全 13 頁)

(21) 出願番号 特願平10-355035  
 (22) 出願日 平成10年12月14日(1998.12.14)  
 (65) 公開番号 特開2000-182381(P2000-182381A)  
 (43) 公開日 平成12年6月30日(2000.6.30)  
 審査請求日 平成17年4月6日(2005.4.6)

(73) 特許権者 503121103  
 株式会社ルネサステクノロジ  
 東京都千代田区大手町二丁目6番2号  
 (73) 特許権者 501203458  
 株式会社ルネサスソリューションズ  
 東京都千代田区大手町二丁目6番2号  
 (74) 代理人 100089118  
 弁理士 酒井 宏明  
 (72) 発明者 永吉 弘尚  
 東京都港区浜松町二丁目4番1号 三菱電  
 機セミコンダクタシステム株式会社内  
 (72) 発明者 石本 真一  
 東京都港区浜松町二丁目4番1号 三菱電  
 機セミコンダクタシステム株式会社内

最終頁に続く

(54) 【発明の名称】一括消去型不揮発性メモリおよび携帯電話

(57) 【特許請求の範囲】

【請求項 1】

複数のセクタから構成されるクラスタにおける未使用領域のアドレス情報を格納するアドレス情報格納手段と、

データの書込要求を受けると、その未使用領域のアドレス情報を参照して、その未使用領域に当該データの書込処理を実行するデータ書込手段と、

上記データ書込手段がデータの書込処理を実行すると、その未使用領域のアドレス情報を更新するアドレス情報更新手段と、

を備え、

前記データ書込手段は、クラスタの未使用領域又は未使用クラスタに対するデータの書込エラーが発生すると、他の未使用クラスタにデータの書込処理を実行するとともに、データの書込エラーが発生したクラスタを一括消去することを特徴とする一括消去型不揮発性メモリ。

【請求項 2】

アプリケーションが使用するクラスタの他に、少なくとも2以上の未使用クラスタが用意されていることを特徴とする請求項1記載の一括消去型不揮発性メモリ。

【請求項 3】

アプリケーションが使用するクラスタにはユニークなIDを登録する一方、未使用クラスタには書込可能であることを示すIDを登録するID管理手段を設け、ID管理手段は、データの書込エラーが発生すると、データの書込エラーが発生したクラスタには書込不

10

20

可能であることを示すIDを登録することを特徴とする請求項1記載の一括消去型不揮発性メモリ。

【請求項4】

一括消去型不揮発性メモリを備える携帯電話であって、  
前記一括消去型不揮発性メモリは、  
複数のセクタから構成されるクラスタにおける未使用領域のアドレス情報を格納するアドレス情報格納手段と、

データの書込要求を受けると、その未使用領域のアドレス情報を参照して、その未使用領域に当該データの書込処理を実行するデータ書込手段と、

上記データ書込手段がデータの書込処理を実行すると、その未使用領域のアドレス情報を更新するアドレス情報更新手段と、

を備え、

前記データ書込手段は、クラスタの未使用領域又は未使用クラスタに対するデータの書込エラーが発生すると、他の未使用クラスタにデータの書込処理を実行するとともに、データの書込エラーが発生したクラスタを一括消去することを特徴とする携帯電話。

【請求項5】

アプリケーションが使用するクラスタの他に、少なくとも2以上の未使用クラスタが用意されていることを特徴とする請求項4記載の携帯電話。

【請求項6】

アプリケーションが使用するクラスタにはユニークなIDを登録する一方、未使用クラスタには書込可能であることを示すIDを登録するID管理手段を設け、ID管理手段は、データの書込エラーが発生すると、データの書込エラーが発生したクラスタには書込不可能であることを示すIDを登録することを特徴とする請求項4記載の携帯電話。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

この発明は、不揮発データを記録するフラッシュEEPROM（以下、フラッシュメモリという）などの一括消去型不揮発性メモリおよび携帯電話に関するものである。

【0002】

【従来の技術】

現在の携帯電話などの組込型システムにおいては、システムのプログラムを格納する場合、ランダムアクセスが可能なフラッシュメモリを用いる一方、書き換え頻度が多いデータや比較的データ量が小さい不揮発データを記録する場合にはEEPROMを用い、また、大規模な不揮発データを記録する場合には、シリアル書き込みによるフラッシュメモリまたはバックアップ電源付きSRAMを用いるようにしている。

【0003】

ここで、EEPROMは、フラッシュメモリに比べて、ビットあたりの記憶素子数が多くビットあたりの単価が増大する。また、バックアップ電源付きSRAMにおいては電源を付けているためコストが増大する。上記の別々に記憶していたデータをフラッシュメモリで記憶することでデバイスの実装面積、及びコストが減少するメリットが生まれる。

【0004】

組込型システムにおける従来のフラッシュメモリは、システムプログラムの書込処理の実行中に、他のシステムプログラムを読み込むことができず、システムプログラムの書き込み時は特殊な処理を施してプログラムを書き込んでいたが、最近、データの書込処理の実行中に、他の書込をしていないエリアからデータを読み込むことができる一括消去型不揮発性メモリ（以下、BGOフラッシュメモリと呼ぶ）の出現により、データを格納していたフラッシュメモリをプログラムが格納されているフラッシュメモリに取り込むことが可能となった。

【0005】

フラッシュメモリにデータを格納する使用例として特公平7-50558号公報に示すよ

10

20

30

40

50

うな半導体ディスクがある（図 14 を参照）。

このシステムでは、フラッシュメモリ 4 を半導体ディスクとして使用し、この半導体ディスクの制御を行うために制御装置 1、RAM 2、バス制御部 3 及び内部バス 5 が設けられているが、フラッシュメモリ 4 はEEPROMやSRAMにない制限がある。

【0006】

即ち、データの書き込みが 0 から 1、または、1 から 0 の一方通行である。このため、一度書き込みが完了した場所に再書き込みをする場合は、書き込みたい場所を含むブロックの全体を一括消去して、ブロック全体を 0 または 1 にした後、データの書き込みを行う必要がある。このため、EEPROMやSRAMのようにバイト単位で書き込みを行うことが難しい。

10

【0007】

【発明が解決しようとする課題】

従来の一括消去型不揮発性メモリは以上のように構成されているので、データの書込処理が実行された場所を含むブロックの全体を一括消去すれば、再度、同一の場所にデータを書き込むことができるが、フラッシュメモリの消去可能回数はEEPROMと比較して、10 分の 1 程度しか保証されておらず、書換頻度の高いデータの記憶には対応することが困難であるなどの課題があった。

【0008】

この発明は上記のような課題を解決するためになされたもので、書換頻度の高いデータの記憶にも対応することができる一括消去型不揮発性メモリおよび携帯電話を得ることを目的とする。

20

【0009】

【課題を解決するための手段】

この発明に係る一括消去型不揮発性メモリは、複数のセクタから構成されるクラスタにおける未使用領域のアドレス情報を格納するアドレス情報格納手段と、データの書込要求を受けると、その未使用領域のアドレス情報を参照して、その未使用領域に当該データの書込処理を実行するデータ書込手段と、上記データ書込手段がデータの書込処理を実行すると、その未使用領域のアドレス情報を更新するアドレス情報更新手段とを備え、前記データ書込手段は、クラスタの未使用領域又は未使用クラスタに対するデータの書込エラーが発生すると、他の未使用クラスタにデータの書込処理を実行するとともに、データの書込エラーが発生したクラスタを一括消去することを特徴とする。

30

【0010】

この発明に係る携帯電話は、一括消去型不揮発性メモリを備える携帯電話であって、前記一括消去型不揮発性メモリは、複数のセクタから構成されるクラスタにおける未使用領域のアドレス情報を格納するアドレス情報格納手段と、データの書込要求を受けると、その未使用領域のアドレス情報を参照して、その未使用領域に当該データの書込処理を実行するデータ書込手段と、上記データ書込手段がデータの書込処理を実行すると、その未使用領域のアドレス情報を更新するアドレス情報更新手段と、を備え、前記データ書込手段は、クラスタの未使用領域又は未使用クラスタに対するデータの書込エラーが発生すると、他の未使用クラスタにデータの書込処理を実行するとともに、データの書込エラーが発生したクラスタを一括消去することを特徴とする。

40

【0023】

【発明の実施の形態】

以下、この発明の実施の一形態を説明する。

実施の形態 1 .

図 1 はこの発明の実施の形態 1 による一括消去型不揮発性メモリを示す構成図であり、図において、11 は複数のセクタを有するクラスタから構成されるBGOフラッシュメモリ（アドレス情報格納手段）であり、BGOフラッシュメモリ 11 の先頭のセクタには未使用領域のアドレス情報が格納されている。12 はBGOフラッシュメモリ 11 を制御するマイクロコンピュータ、13 はマイクロコンピュータ 12 のRAM、14 はマイクロコン

50

ピュータ１２のＣＰＵであり、ＣＰＵ１４は未使用領域のアドレス情報をＢＧＯフラッシュメモリ１１の先頭のセクタに格納するアドレス情報格納手段と、データの書込要求を受けると、そのクラスタにおける未使用領域のアドレス情報を参照して、その未使用領域にデータの書込処理を実行するデータ書込手段と、その未使用領域のアドレス情報を更新するアドレス情報更新手段と、各クラスタにＩＤを登録するＩＤ管理手段と、ポインタ構築手段とを構成する。

なお、図１のシステムの周辺には、本システムを使用する携帯電話などの周辺デバイスが接続される。

#### 【００２４】

次に動作について説明する。

10

最初に、図２を参照しながらＢＧＯフラッシュメモリ１１の物理アドレス、クラスタ管理エリア及びアプリケーションエリアについて説明する。ただし、ＢＧＯフラッシュメモリ１１は、データを消去するとＦＦｈになり、データの書き換え方向は１から０であるとする。

#### 【００２５】

クラスタ管理エリアのクラスタＩＤは、１バイトで表現され、００ｈ及びＦＦｈを除いた２５４通りの表現が可能である（アプリケーションが使用するクラスタにはユニークなＩＤが登録される）。

また、ＦＦｈが初期状態のためＦＦｈが未使用クラスタＩＤ（ＦＦｈは書込可能なクラスタであることを示す）、００ｈが無効クラスタＩＤ（００ｈは書込不可能なクラスタであることを示す）となる。

20

図２の３個のクラスタＩＤには同じ値が格納され、２以上のクラスタＩＤが一致するとき、クラスタＩＤが有効であると判断する。

#### 【００２６】

３個のクラスタＩＤの次にはコピーフラグが１バイト確保され、消去ブロックのクラスタ管理エリアの４バイトを除いたエリアがアプリケーションエリアとなる。このアプリケーションエリアは、アプリケーションが自由に使用することができるエリアである。

#### 【００２７】

各アプリケーションは、自分のデータが格納されているアドレスを直接保持することではなく、自分のアプリケーションＩＤを保持している。制約事項として、このアプリケーションＩＤは、システムを使用する全てのアプリケーションにおいて唯一のものでなければならない。

30

実際にアプリケーションがＢＧＯフラッシュメモリ１１を使用する場合は、自分のアプリケーションＩＤに一致するクラスタＩＤを検索して、自分の使用している物理アドレスを検索し、そこからさらに、自分の読み書きしたいデータのアドレスを計算する。

#### 【００２８】

ＢＧＯフラッシュメモリ１１は、データの消去をブロック単位で行う必要があるため、ブロック内のあるエリアのみを消去する場合は、必要データの待避、ブロックの消去、必要データの書き込みを行う必要がある。

必要データの待避をＲＡＭ１３に行うことは、コストの面、ＢＧＯフラッシュメモリ１１を使う意味、及びデータの信頼性からも望ましいものではない。

40

このためデータの待避は必然的にＢＧＯフラッシュメモリ１１上に行うが、そのためにはデータが書き込まれていない未使用領域、即ち、ブランクエリアが必要である。上記操作をデータリクレームと呼び、ＢＧＯフラッシュメモリ１１において避けて通ることのできない処理である。

#### 【００２９】

ここで、図３及び図４を参照しながらデータリクレーム時の処理を説明する。

例えば、クラスタＡのデータリクレームが必要になると（ステップＳＴ１）、クラスタＡのアプリケーションは、ＢＧＯフラッシュメモリ１１上のブランクエリアを検索する（ステップＳＴ２）。例えば、クラスタＤがブランクエリアである場合には、クラスタＤのク

50

ラストIDを参照して、ブランクエリアのアドレスを取得する（ステップST4）。ただし、ブランクエリアが存在しない場合には、無効クラスタを一括消去してブランクエリアを作成し（ステップST3）、そのブランクエリアのアドレスを取得する（ステップST4）。

【0030】

そして、ブランクエリアのアドレスを取得すると、クラスタAのアプリケーションは、現在使用中であるクラスタAのコピーフラグを01hにして（ステップST5）、ブランクエリア（クラスタD）に自分のアプリケーションID（1hのID）を登録する（ステップST6）。（図4の中段を参照）

【0031】

クラスタAのアプリケーションは、ブランクエリア（クラスタD）に自分のアプリケーションIDを登録すると、必要データをブランクエリアにコピーし（ステップST7）、コピーが完了すると、古いエリアであるクラスタAを一括消去または無効化して、処理を終了する（ステップST8, ST9）。（図4の下段を参照）

【0032】

ただし、データリクレーム中に電源遮断などの障害が発生すると、クラスタIDが同一のクラスタが存在してしまう場合がある。このような場合は、アプリケーションに割り振られたクラスタIDは重複できないという制約事項があるため、システム起動時に取り除く必要がある。

【0033】

このため、システム起動時に、重複クラスタの存在をチェックするが、重複クラスタを検出すると、重複クラスタのうち、コピーフラグがFFhであるものを削除する。その理由は、一括消去またはデータの無効化をする前に、システムの障害が発生すると、データリクレームの処理がどこまで完了したかを見極めることが困難である。そのため、コピーフラグを見てコピー元を判断し、コピー先を削除する。コピー元を保持しておけば、データの内容が壊れて書き込みが不可能になる不具合を回避することができるからである。

【0034】

次に、データの書き込み時にデバイスエラー（書込エラー）が発生すると、再度、データの書込処理を実行する必要がある。そのような場合に対処するために、ブランクエリアを2エリア以上用意しておくようにする。

また、データの書込エラーが発生すると、書込エラーが発生したクラスタIDを無効クラスタに変更して、新しいブランククラスタを取得し、リクレーム処理を実行する。無効クラスタにおいても、一旦、一括消去を行うと、クラスタの状態が元の正常な状態に戻る場合があるため、電源投入時や、システムが暇な時などに無効エリアを一括消去する。これは、デバイスの突発的なエラーに対して有効な手段である。

【0035】

図5はアプリケーションにおける内部データを示す構造図である。

クラスタはnページのセクタから構成されており、0ページ目にヘッダが確保され、1ページ目以降にデータエリアが確保されている。

このうち0ページの先頭4バイトはクラスタIDとコピー用フラグとして使用され、その後ろに32バイトの格納ページ情報が格納される。書き込むデータ長はシステムにて予め決定され、そのシステム内においては不変である。

【0036】

図7はデータの読み込み手順を示すものであるが、アプリケーションは、クラスタIDを参照して格納ページ情報を読み出し、データが格納されているセクタであるページ番号を取得する。

アプリケーションは、例えば、データがxページ目に格納されていることを認識すると、xページ目の先頭位置に格納されている格納位置オフセット情報を参照し、xページの何処にデータが格納されているかを認識する。

【0037】

ここで、格納ページ情報と格納位置オフセット情報は、ビットマップフィールドで表記され、両者とも32バイトのエリアを確保し、ビットが256個存在する。各ビットの値は、セクタの使用状態が変化することにより、1から0に変更される。

【0038】

即ち、格納ページ情報については、例えば、1ページ目からxページ目までデータが格納されると、LSBからxビットまで、ビット値が1から0に変更され、0の数のトータルが指定ページに対応する。

一方、格納位置オフセット情報については、図6に示すように、データ領域の1バイトと、格納位置オフセット情報の1ビットが対応し、データ領域の先頭バイトから順次データが格納されるごとに、格納位置オフセット情報のLSBから順次1ビットが1から0に変更される。

10

【0039】

これにより、アプリケーションは、格納ページ情報における0のビットの合計からデータが格納されているページ番号を認識すると、格納位置オフセット情報のLSBから順次ビットの値をサーチして、データ領域の先頭からどこまでデータが格納されているかを認識する。

例えば、格納位置オフセット情報のLSBから24ビットまでが0の場合には、データ領域の先頭から24バイトまでデータが格納されていると認識する。

そして、アプリケーションは、データが格納されている最終位置を認識すると、そこからデータ長分前に戻った位置がデータの先頭アドレスであるので、その先頭アドレスからデータを読み出す処理を実行する。

20

【0040】

図8はデータの書込手順を示すフローチャートである。

まず、アプリケーションは、データの読み出し時と同じ手順で、データが格納されている最終位置を認識すると、そのページ内にブランクエリア（未使用領域）が存在するか否かを調査する（ステップST11）。

【0041】

ブランクエリアが存在する場合は、データが格納されている最終位置の次の位置を先頭アドレスとして、書込要求に係るデータの書込処理を実行するとともに、格納位置オフセット情報を更新する（ステップST12）。即ち、データが格納されている最終位置の次の位置に対応するビットを1から0に変更する。

30

【0042】

一方、現在のページ内にブランクエリアが存在しない場合、クラスタ内に次のページが存在するか否かを調査する（ステップST13）。

次のページが存在する場合には、次のページの格納位置オフセット情報を参照して、次のページがブランクであるか否かを調査する（ステップST14）。これは、次のページにおける過去のデータ書込時において、電源遮断等の障害が発生したことがあると、書込途中のデータが残っていることがあるからである（書込途中のデータが残っていると、新たなデータを書き込むことができない不具合が発生する）。

【0043】

アプリケーションは、次のページがブランクである場合には、データ領域の先頭位置に、書込要求に係るデータの書込処理を実行するとともに、格納位置オフセット情報を更新する（ステップST15）。また、アプリケーションは、格納ページ情報も更新する（ステップST16）。即ち、次のページに対応するビットを1から0に変更する。

40

【0044】

次のページがブランクでない場合には、更に次のページがブランクであるか否かを調査するが、次のページが存在しなくなると、上述したデータリクレームを実行して、他の未使用クラスタにおける1ページ目のデータ領域の先頭位置に、書込要求に係るデータの書込処理を実行する（ステップST17）。

【0045】

50

以上で明らかなように、この実施の形態 1 によれば、データの書込要求を受けると、未使用領域のアドレス情報を参照して、その未使用領域にデータの書込処理を実行するように構成したので、クラスタに未使用領域が存在する場合には、当該クラスタを一括消去することなく、データの書き込みが可能になり、その結果、B G O フラッシュメモリ 1 1 の書換回数が増加する効果を奏する。

#### 【 0 0 4 6 】

実施の形態 2 .

上記実施の形態 1 では、ページ内にブランクエリアが存在する場合、既にかき込まれているデータの後ろに詰めて最新のデータをかき込むものについて示したが、即ち、ページ内には複数のデータを格納する領域を確保して、最後に格納されたデータ（最新データ）のみを有効にするものについて示したが、図 9 に示すように、セクタ（ページ）内に複数のデータブロックを構成し、セクタのヘッダエリア内に格納されているデータブロックの使用状況を示す使用状況フラグ（ページ内使用中ブロックフラグ、ページ内使用済ブロックフラグ）を参照して、各データブロックの未使用領域にデータの書込処理を実行するようにしてもよい。

#### 【 0 0 4 7 】

図 9 はアプリケーションにおける内部データを示す構造図である。

クラスタの全てにページヘッダと、複数のデータブロックを作成する。ページヘッダにはクラスタ ID と、コピー用フラグを先頭におき、その後ろにページ内使用中ブロックフラグを 3 バイト、ページ内使用済ブロックフラグを 3 バイトおく。

なお、データブロック内には、データ NO とデータ領域使用状況を 1 バイトずつおく、その後ろ 8 バイトをデータ領域として使用する。

#### 【 0 0 4 8 】

図 1 0 はページ内使用中ブロックフラグ及びページ内使用済ブロックフラグとデータブロックの対応図である。

使用中のブロックを検索するには、ページ内使用中ブロックフラグとページ内使用済ブロックフラグの排他的論理和を求め、その結果が 1 のビットに対応するデータブロックが使用中のデータブロックになる（図 1 0 の例では、データブロック P ）。

#### 【 0 0 4 9 】

データの読み書きは、データ ID を指定することにより行う。

このうち、0 0 h は使用済データ ID とし、F F h はブランクデータ ID とする。その他、2 5 4 種類のデータ ID をとることができる。各 ID に割り振られるデータサイズは 1 バイトから 8 バイトまでの値をとることができ、プログラム中では、データサイズを変えることがないものとする。

#### 【 0 0 5 0 】

また、アプリケーションのポインタ構築手段がデータ検索用のポインタ（データを格納しているデータブロックのブロック位置を示すポインタ）を R A M 1 3 に作成する。ポインタの内部にはページ番号及びデータエリアのブロック番号を格納する。これを電源投入時など R A M 1 3 の内容が消えてしまったときに、全てのデータエリアをサーチして検索し、内容を保持する。

#### 【 0 0 5 1 】

読み出し時は、アプリケーションがデータ ID のポインタが示すデータブロックを検索し、データ領域使用状況（図 1 1 を参照）を参照してデータの読込処理を実行する。

即ち、データ領域使用状況の 8 ビットがデータ領域の 8 バイトの使用状況に対応しており、データ領域使用状況のビットが 0 の場所が使用済の場所である。

#### 【 0 0 5 2 】

図 1 2 はデータ検索用ポインタの初期ルーチンを示すフローチャートである。

電源投入時などの初期ルーチンにおいては、アプリケーションが使用クラスタのアドレスを検索し、ページ内にデータがある場合には、ポインタのデータを更新する（ステップ S T 2 1 ~ S T 2 3 ）。

10

20

30

40

50

ページ内の使用データの検索方法であるが、全てのブロックを検索すると非常にデータ数が多いため、データの読込時と同様に、ページヘッダのページ内使用中ブロックフラグとページ内使用済ブロックフラグを参照して、使用データを検索する。

そして、全てのページの検索が終了するまで、上記の処理を繰り返し（ステップS T 2 4 , S T 2 5 ）、最後のページの検索が終了した時点でポインタの作成が完了する。

【 0 0 5 3 】

図 1 3 はデータの書込手順を示すフローチャートである。

まず、アプリケーションは、データが格納されているデータブロックのブランクエリアを検索し、データを書き込むことができるブランクエリアがデータエリア内に存在する場合には、そのデータブロックのデータエリア（ブランクエリア）内にデータの書込処理を実行する（ステップS T 3 1 , S T 3 2 ）。

10

【 0 0 5 4 】

一方、データを書き込むことができるブランクエリアがデータエリア内に存在しない場合には、他の新しいデータブロックを取得できるか否かを判断する（ステップS T 3 3 ）。新しいデータブロックが取得できた場合、新しいデータブロックにデータNO、データ領域及びページ内使用中ブロックフラグを書き込む処理を実行する（ステップS T 3 4 , S T 3 5 ）。また、ページ内使用済ブロックフラグ及びデータブロックのIDの書込も同時に実行する（ステップS T 3 6 , S T 3 7 ）。なお、データブロックを新しく取得できない場合には、データリクレームを実行する（ステップS T 3 8 ）。

20

【 0 0 5 5 】

以上で明らかなように、この実施の形態 2 によれば、セクタが複数のデータブロックから構成される場合、セクタのヘッダ領域内に格納されているデータブロックの使用状況を示す使用状況フラグ（ページ内使用中ブロックフラグ、ページ内使用済ブロックフラグ）等を参照して、そのデータブロックの未使用領域にデータの書込処理を実行するように構成したので、1 バイトから 8 バイトの単位で、最大 2 5 4 種類のデータを自由に書き換えることができる効果を奏する。

【 0 0 5 6 】

実施の形態 3 .

上記実施の形態 1 及び実施の形態 2 では、B G O フラッシュメモリ 1 1 にデータを書き込むものについて示したが、これに限るものではなく、その他の一括消去型不揮発性メモリにおいても使用可能であることは言うまでもない。

30

【 0 0 5 7 】

【発明の効果】

以上のように、この発明によれば、クラスタの未使用領域又は未使用クラスタに対するデータの書込エラーが発生すると、他の未使用クラスタにデータの書込処理を実行するとともに、データの書込エラーが発生したクラスタを一括消去するように構成したので、データの書込エラーが発生しても、データの書込処理を実行することができる効果がある。

【図面の簡単な説明】

【図 1】 この発明の実施の形態 1 による一括消去型不揮発性メモリを示す構成図である。

40

【図 2】 B G O フラッシュメモリの内部構造及びクラスタ管理エリアを示す説明図である。

【図 3】 データリクレームの処理を示すフローチャートである。

【図 4】 データリクレーム時におけるクラスタIDの割り当てを示す説明図である。

【図 5】 アプリケーションにおける内部データを示す構成図である。

【図 6】 格納位置オフセット情報とデータの対応関係を示す説明図である。

【図 7】 データの読み込み手順を示す説明図である。

【図 8】 データの書込手順を示すフローチャートである。

【図 9】 アプリケーションにおける内部データを示す構成図である。

【図 1 0】 ページ内使用中ブロックフラグ及びページ内使用済ブロックフラグとデータ

50



ブロックの対応図である。

【図 1 1】 データ領域使用状況とデータ領域の対応図である。

【図 1 2】 データ検索用ポインタの初期ルーチンを示すフローチャートである。

【図 1 3】 データの書込手順を示すフローチャートである。

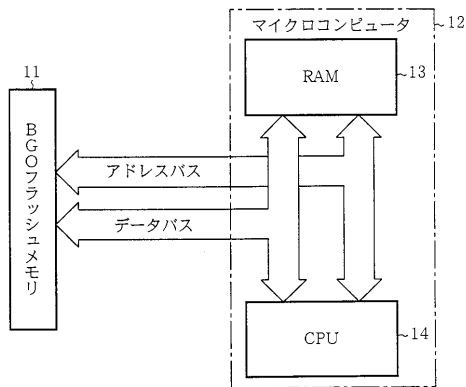
【図 1 4】 フラッシュメモリにデータを格納する半導体ディスクを示す構成図である。

【符号の説明】

1 1 BGOフラッシュメモリ（アドレス情報格納手段）、1 4 CPU（アドレス情報格納手段、データ書込手段、アドレス情報更新手段、ID管理手段、ポインタ構築手段）

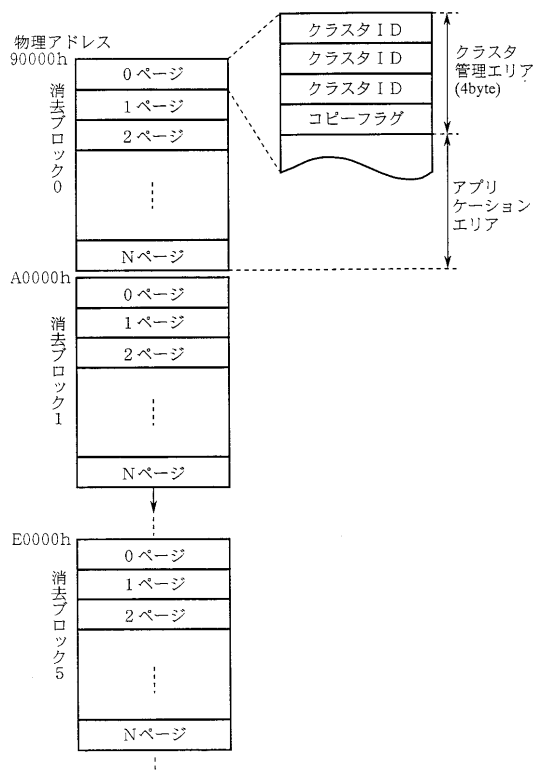
。

【図 1】

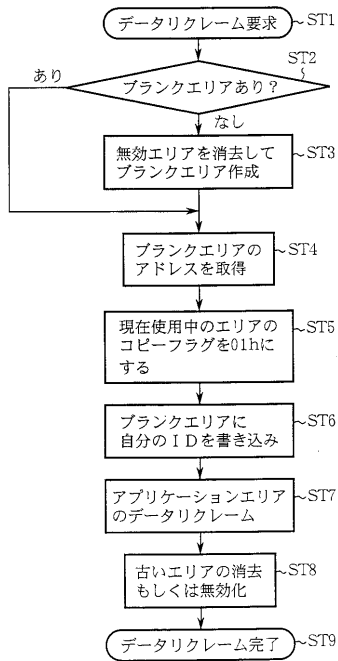


11: BGOフラッシュメモリ（アドレス情報格納手段）  
14: CPU（アドレス情報格納手段、データ書込手段、  
アドレス情報更新手段、ID管理手段、  
ポインタ構築手段）

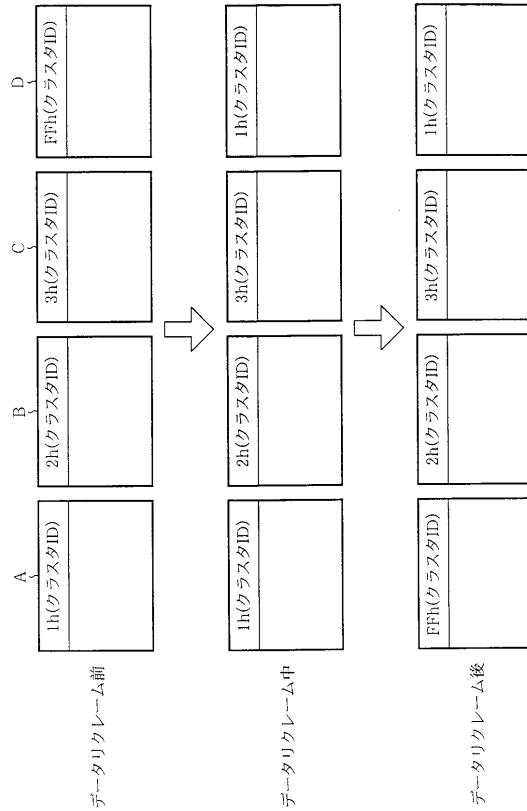
【図 2】



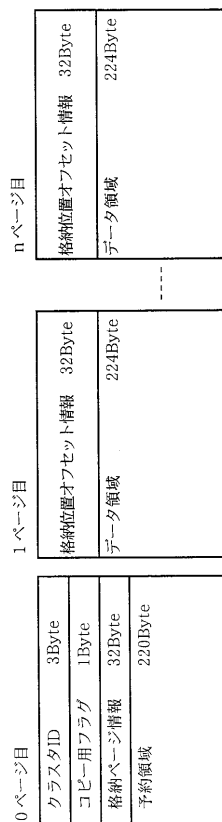
【図 3】



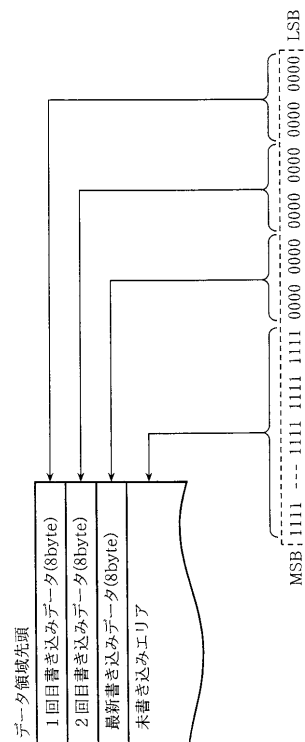
【図 4】



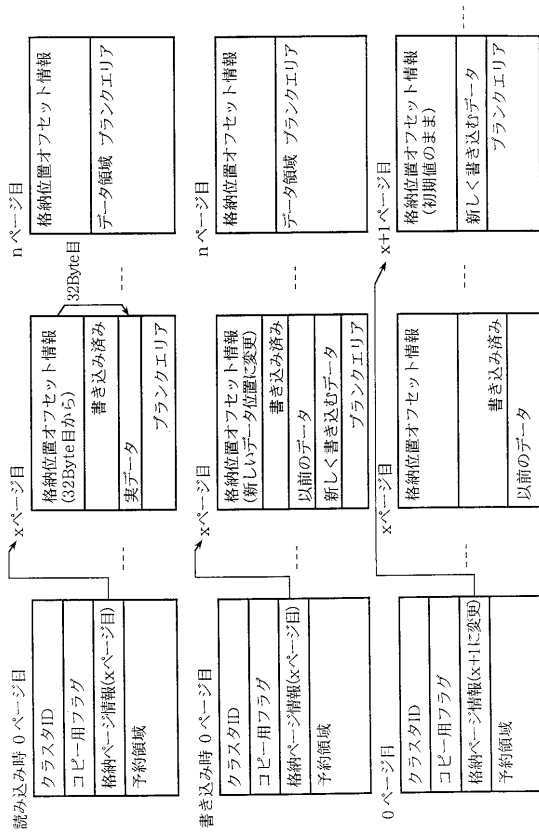
【図 5】



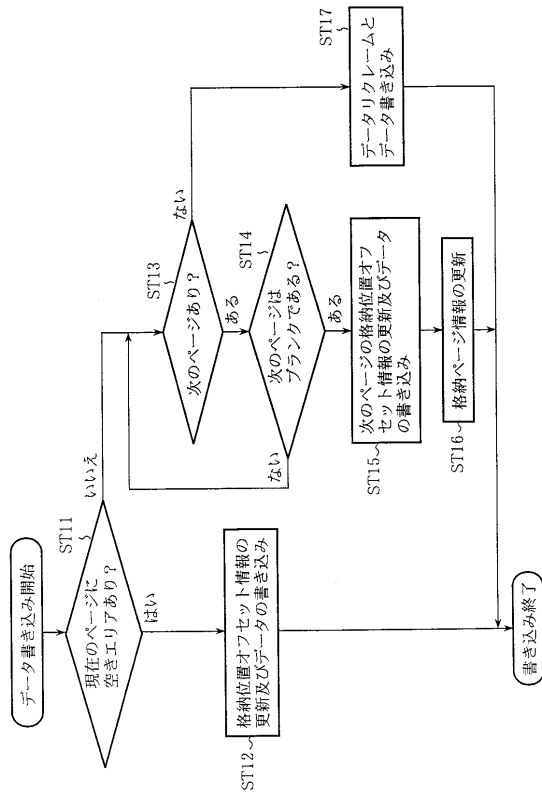
【図 6】



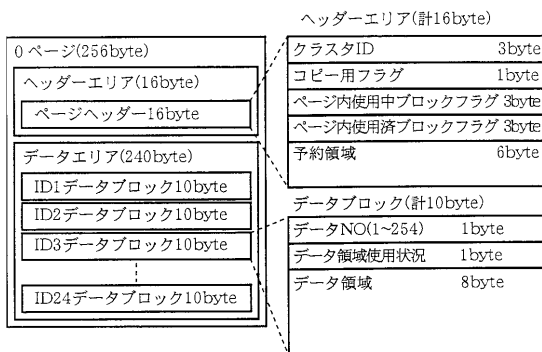
【図 7】



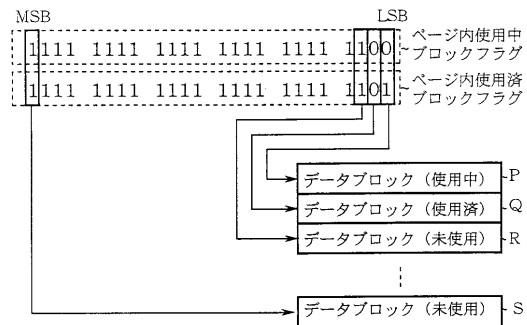
【図 8】



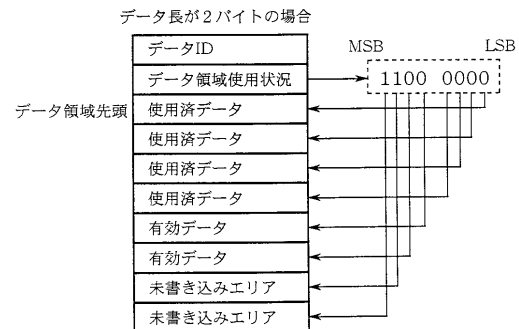
【図 9】



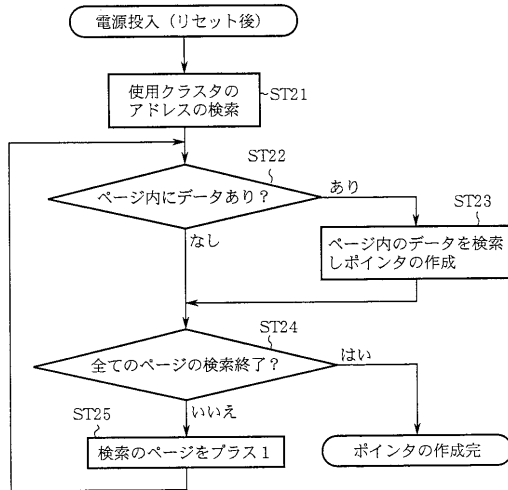
【図 10】



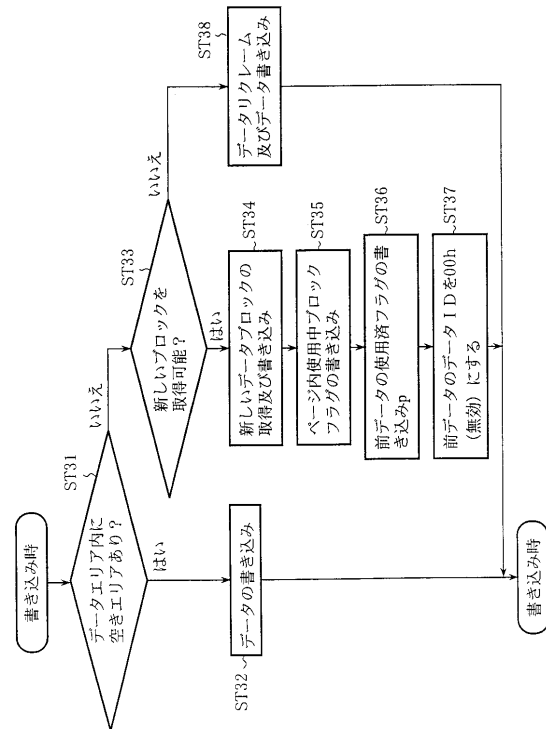
【図 11】



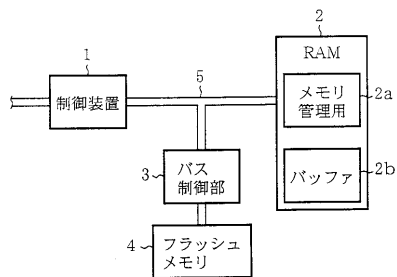
【図 12】



【図 13】



【図 14】



---

フロントページの続き

審査官 滝谷 亮一

(56)参考文献 国際公開第97/012325(WO, A1)

特開平07-105691(JP, A)

特開平06-124596(JP, A)

特開平08-137634(JP, A)

特開平06-222986(JP, A)

特開平10-320984(JP, A)

特開平09-022376(JP, A)

特開平05-027924(JP, A)

特開平09-097217(JP, A)

特開平11-073363(JP, A)

特開平07-191892(JP, A)

(58)調査した分野(Int.Cl., DB名)

G11C 16/02-16/06

G06F 12/00-12/06