

(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 1

G 11 C 16/02	(2006.01)	G 11 C 17/00	601C
G 06 F 12/02	(2006.01)	G 06 F 12/02	510A
G 06 F 12/16	(2006.01)	G 06 F 12/16	310R
G 06 F 15/78	(2006.01)	G 06 F 15/78	510F

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、

データの書き要求を受けると、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き処理を実行するデータ書き手段と、

上記データ書き手段がデータの書き処理を実行すると、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、

前記データ書き手段は、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きエラーが発生すると、他の未使用クラスタにデータの書き処理を実行するとともに、データの書き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き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、データの書きエラーが発生すると、データの書きエラーが発生したクラスタには書き不

可能であることを示す I D 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、
データの書き要求を受けると、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き処理を実行するデータ書き手段と、
上記データ書き手段がデータの書き処理を実行すると、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、
前記データ書き手段は、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
きエラーが発生すると、他の未使用クラスタにデータの書き処理を実行するとともに、デ
ータの書き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 I D を登録する一方、未使用クラ
スタには書き可能であることを示す I D を登録する I D 管理手段を設け、I D 管理手段は
、データの書きエラーが発生すると、データの書きエラーが発生したクラスタには書き不
可能であることを示す I D 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き込み処理の実
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、データの書き込み処理の実
行中に、他の書き込み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、RAM2、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、データの書き込み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、書き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、書き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、データの書き込み要求を受けると、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き込み処理を実行するデータ書き込み手段と、上記データ書き込み手段がデータの書き込み処理を実行すると、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、前記データ書き込み手段は、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き込みエラーが発生すると、他の未使用クラスタにデータの書き込み処理を実行するとともに、データの書き込み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、データの書き込み要求を受けると、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き込み処理を実行するデータ書き込み手段と、上記データ書き込み手段がデータの書き込み処理を実行すると、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、前記データ書き込み手段は、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き込みエラーが発生すると、他の未使用クラスタにデータの書き込み処理を実行するとともに、データの書き込み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BG0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BG0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、BG0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は書き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は書き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BG0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】

BG0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に行なうことは、コストの面、BG0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BG0フラッシュメモリ11上に行なうが、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、BG0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、BG0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

ラスタＩＤを参照して、プランクエリアのアドレスを取得する（ステップＳＴ４）。ただし、プ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プランクエリアを作成し（ステップＳＴ３）、そのプ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、プ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）、プランクエリア（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、プランクエリア（クラスタD）に自分のアプリケーションIDを登録すると、必要データをプ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（書きエラー）が発生すると、再度、データの書き込み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、プランクエリアを2エリア以上用意しておくようにする。

また、データの書き込みエラーが発生すると、書き込み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、新しいプ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变である。

【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】

10

20

30

40

50

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ごとに、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 はデータの書き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 ST 11）。

【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、書き要求に係るデータの書き出し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 ST 12）。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 ST 13）。

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 ST 14）。これは、次のページにおける過去のデータ書き出し時ににおいて、電源遮断等の障害が発生したことがあると、書き途中のデータが残っていることがあるからである（書き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）。

40

【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、書き要求に係るデータの書き出し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 ST 15）。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 ST 16）。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。

【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、書き要求に係るデータの書き出し処理を実行する（ステップ ST 17）。

【0045】

50

以上で明らかなように、この実施の形態 1 によれば、データの書き込み要求を受けると、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き込み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 11 の書き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、既に書き込まれているデータの後ろに詰めて最新のデータを書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き込み処理を実行するようにしてよい。

【 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、複数個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 バイトずつおき、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 10 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 10 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、00h は使用済データ ID とし、FFh はブランクデータ ID とする。その他、254 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 RAM 13 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 RAM 13 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 11 を参照）を参照してデータの読み込み処理を実行する。

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 12 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、データの読み込み時と同様に、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。

【0053】

図13はデータの書き込み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）内にデータの書き込み処理を実行する（ステップ S T 3 1, S T 3 2）。

【0054】

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の書き込みも同時に実行する（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）。

【0055】

以上で明らかのように、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き込み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254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。

【0056】

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 GOフラッシュメモリ11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、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。

【0057】

【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き込みエラーが発生すると、他の未使用クラスタにデータの書き込み処理を実行するとともに、データの書き込み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、データの書き込みエラーが発生しても、データの書き込み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。

【図2】 B GO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造図である。

【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】 データの書き込み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造図である。

【図10】 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

10

20

30

40

50

ブロックの対応図である。

【図11】 データ領域使用状況とデータ領域の対応図である。

【図12】 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。

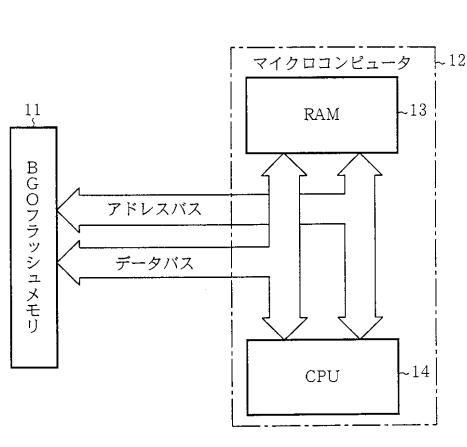
【図13】 データの書き手順を示すフローチャートである。

【図14】 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】

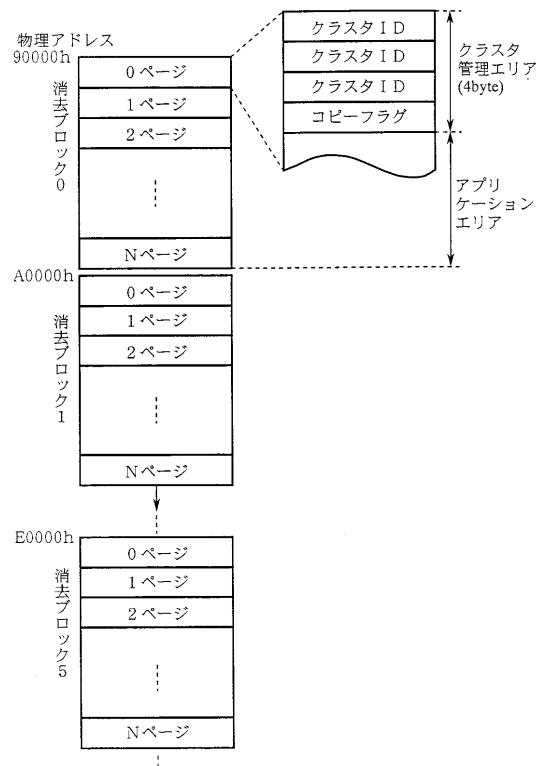
11 BGO フラッシュメモリ (アドレス情報格納手段)、14 CPU (アドレス情報格納手段、データ書き手段、アドレス情報更新手段、ID管理手段、ポインタ構築手段)
。

【図1】

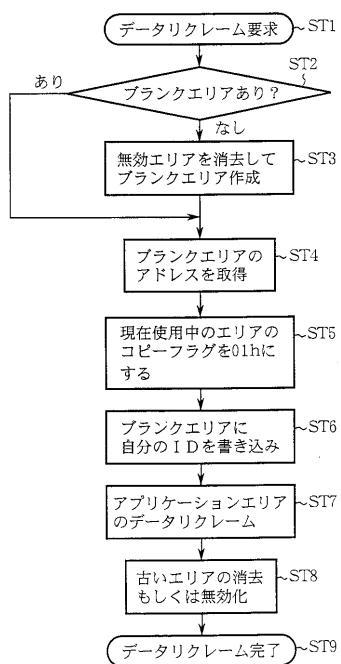


11 : BGO フラッシュメモリ (アドレス情報格納手段)
14 : CPU (アドレス情報格納手段、データ書き手段、アドレス情報更新手段、ID管理手段、ポインタ構築手段)

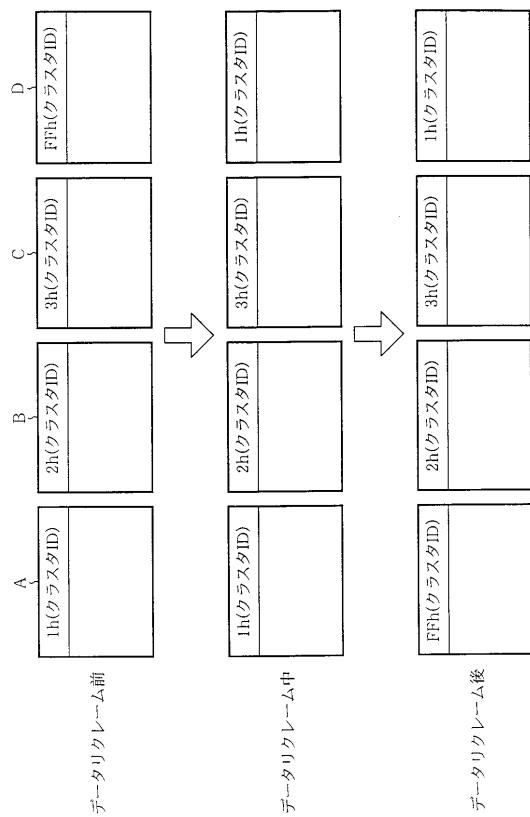
【図2】



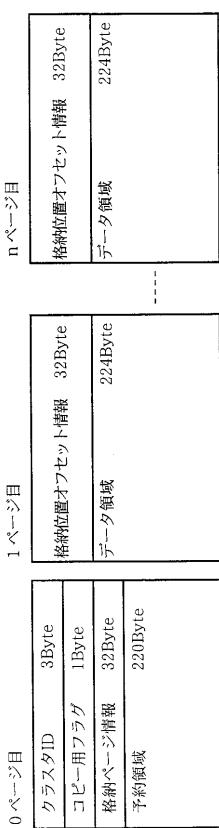
【図3】



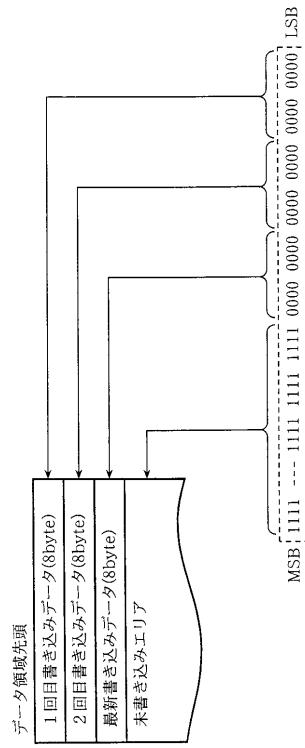
【図4】



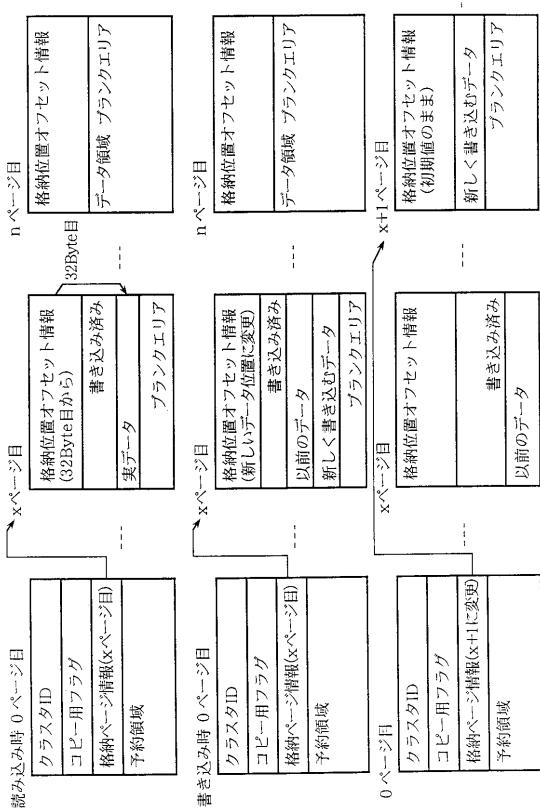
【図5】



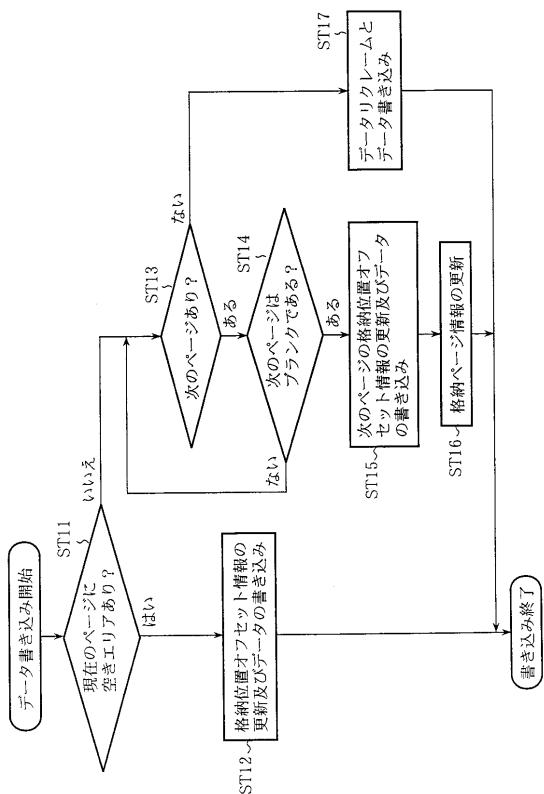
【図6】



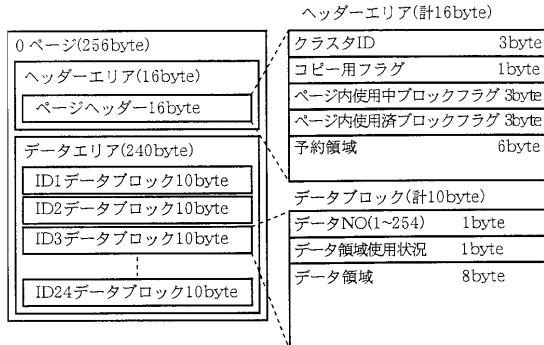
【図7】



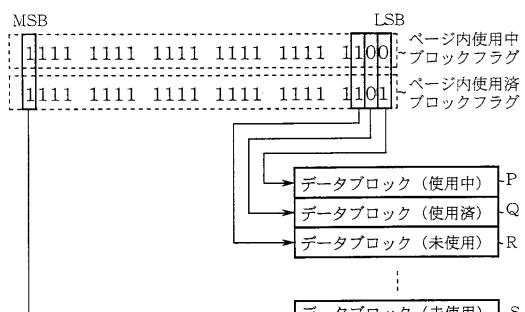
【図8】



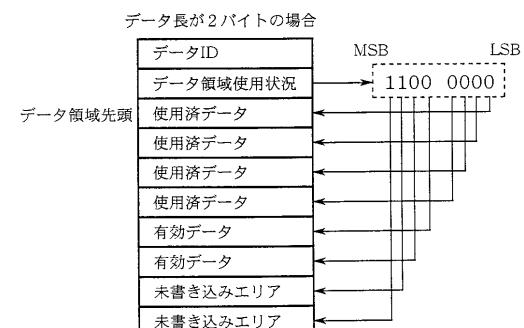
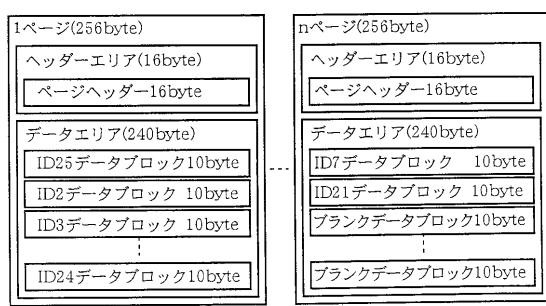
【図9】



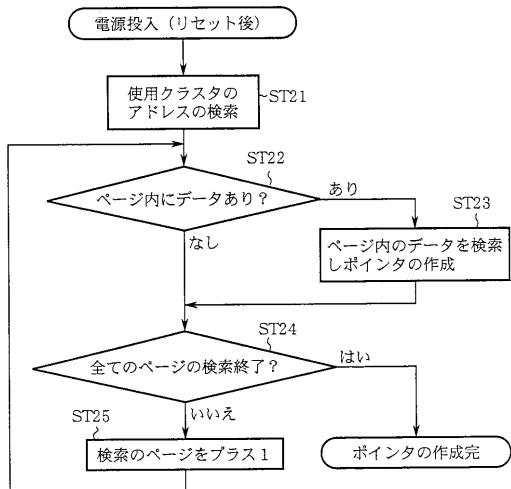
【図10】



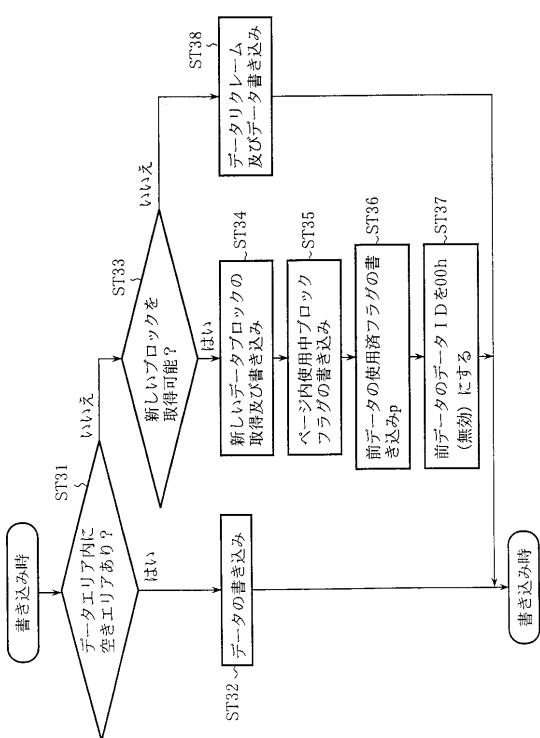
【図11】



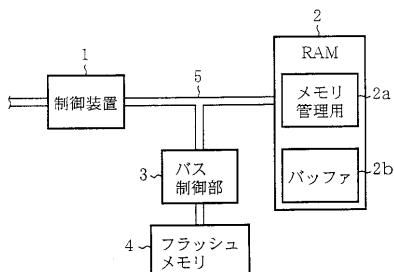
【図12】



【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