

(19)日本国特許庁(JP)

(12)公開特許公報(A)

(11)公開番号

特開2023-24008
(P2023-24008A)

(43)公開日 令和5年2月16日(2023.2.16)

(51)国際特許分類	F I	テーマコード(参考)
G 0 6 F 12/1027(2016.01)	G 0 6 F 12/1027	5 B 1 6 0
G 0 6 F 12/00 (2006.01)	G 0 6 F 12/00 5 9 7 U	5 B 2 0 5
G 0 6 F 12/02 (2006.01)	G 0 6 F 12/02 5 7 0 A	
G 0 6 F 12/0866(2016.01)	G 0 6 F 12/0866 1 0 0	
G 0 6 F 11/10 (2006.01)	G 0 6 F 11/10 6 6 4	

審査請求 未請求 請求項の数 7 O L (全17頁)

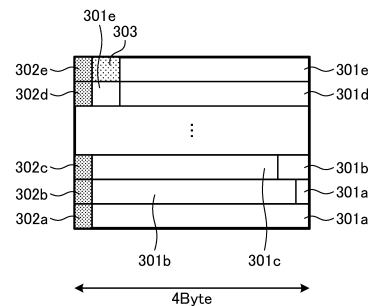
(21)出願番号	特願2021-130038(P2021-130038)	(71)出願人	318010018 キオクシア株式会社 東京都港区芝浦三丁目1番21号
(22)出願日	令和3年8月6日(2021.8.6)	(74)代理人	110002147 弁理士法人酒井国際特許事務所
		(72)発明者	市場 冬樹 東京都港区芝浦三丁目1番21号 キオクシア株式会社内
		(72)発明者	藤田 隆史 東京都港区芝浦三丁目1番21号 キオクシア株式会社内
		Fターム(参考)	5B160 AB26 MM02 5B205 MM04 MM11 RR02

(54)【発明の名称】 メモリシステム

(57)【要約】

【課題】アドレス変換情報を効率よくキャッシュする。
 【解決手段】メモリシステムはホストに接続可能である。メモリシステムは、不揮発性の第1メモリと、第2メモリと、第2メモリに格納される複数のキャッシュデータのそれぞれを、複数のキャッシュデータを含むセグメントの単位で管理するコントローラと、を備える。複数のキャッシュデータのそれぞれは、メモリシステムがホストに提供する論理アドレス空間内の位置を示す論理アドレス値と第1メモリ内の位置とを対応付ける対応情報を含む。セグメントには、少なくとも2つのキャッシュデータが、間にスペースを空けずに配置される。

【選択図】図4



【特許請求の範囲】**【請求項 1】**

ホストに接続可能なメモリシステムであって、
不揮発性の第 1 メモリと、
第 2 メモリと、

前記第 2 メモリに格納される複数のキャッシュデータのそれぞれを、前記複数のキャッシュデータを含むセグメントの単位で管理するコントローラと、を備え、

前記複数のキャッシュデータのそれぞれは、前記メモリシステムが前記ホストに提供する論理アドレス空間内の位置を示す論理アドレス値と前記第 1 メモリ内の位置とを対応付ける対応情報を含み、前記セグメントには、少なくとも 2 つの前記キャッシュデータが、
間にスペースを空けずに配置される
メモリシステム。

10

【請求項 2】

前記コントローラは、前記第 2 メモリとの間で、前記セグメントを第 1 サイズの単位で転送し、前記セグメントの前記第 1 サイズの部分には、2 つの前記キャッシュデータそれぞれの少なくとも一部が間にスペースを空けずに含まれる、

請求項 1 に記載のメモリシステム。

【請求項 3】

前記コントローラは、さらに、前記複数のキャッシュデータそれぞれの位置を示す情報を前記第 2 メモリに格納する、

請求項 1 に記載のメモリシステム。

20

【請求項 4】

第 3 メモリをさらに備え、

前記コントローラは、さらに、前記第 2 メモリに格納された前記キャッシュデータを前記第 3 メモリに格納し、前記第 3 メモリに格納した前記キャッシュデータに含まれる前記対応情報を更新して、前記更新された対応情報を含む前記キャッシュデータを前記第 2 メモリに格納する、

請求項 1 に記載のメモリシステム。

【請求項 5】

前記コントローラは、さらに、前記第 2 メモリに格納された前記複数のキャッシュデータを、前記セグメントの単位で前記第 1 メモリに格納する、

請求項 1 に記載のメモリシステム。

30

【請求項 6】

前記コントローラは、さらに、

それぞれが前記複数のキャッシュデータを含む複数の誤り訂正フレームを前記第 2 メモリに格納し、

前記複数の誤り訂正フレームのそれぞれに含まれるパリティを用いて、前記複数の誤り訂正フレームのそれぞれに対して誤り訂正を行い、

前記複数のキャッシュデータのうちの第 1 のキャッシュデータを、前記複数の誤り訂正フレームのうちの第 1 の誤り訂正フレームと第 2 の誤り訂正フレームとの 2 つに跨って前記第 2 メモリに格納する、

請求項 1 に記載のメモリシステム。

40

【請求項 7】

前記コントローラは、さらに、

前記第 1 の誤り訂正フレームと前記第 2 の誤り訂正フレームとを前記第 2 メモリから読み出し、前記第 1 の誤り訂正フレームが前記第 1 のキャッシュデータを含み前記第 2 の誤り訂正フレームが前記第 1 のキャッシュデータを含まないように、前記第 1 の誤り訂正フレームおよび前記第 2 の誤り訂正フレームを再構成し、前記第 1 のキャッシュデータを含むように再構成された前記第 1 の誤り訂正フレームおよび前記第 1 のキャッシュデータを含まないように再構成された前記第 2 の誤り訂正フレームを前記第 1 メモリに格納する、

50

請求項 6 に記載のメモリシステム。

【発明の詳細な説明】

【技術分野】

【0001】

本実施形態は、メモリシステムに関する。

【背景技術】

【0002】

不揮発性メモリを備えるメモリシステムが知られている。不揮発性メモリは、一例では、NAND型のフラッシュメモリである。

【0003】

メモリシステムは、複数のアドレスに関するアドレス変換情報 (Address Translation Information) を LUT (Lookup Table) として不揮発性メモリ内に保持する。アドレス変換情報は、論理アドレス空間内の位置を示す論理アドレスを不揮発性メモリ内の位置を示す物理アドレスに対応付ける情報である。

【0004】

メモリシステムは、ある論理アドレスを物理アドレスに変換する際にアドレス変換情報を必要とする。しかしながら、不揮発性メモリに対するアクセススピードはあまり速くないため、不揮発性メモリからアドレス変換情報を取得しようとする、アドレス変換処理に要する時間が増大する。そこで、アドレス変換情報をより高速に取得できるように、メモリシステムには、不揮発性メモリより高速にアクセス可能なキャッシュメモリが設けられる。メモリシステムの動作中、アドレス変換情報の一部が、キャッシュデータとしてキャッシュメモリに格納される。

【先行技術文献】

【特許文献】

【0005】

【特許文献 1】米国特許出願公開第 2014 / 359219 号明細書

【発明の概要】

【発明が解決しようとする課題】

【0006】

一つの実施形態は、アドレス変換情報を効率よくキャッシュできるメモリシステムを提供することを目的とする。

【課題を解決するための手段】

【0007】

一つの実施形態によれば、メモリシステムはホストに接続可能である。メモリシステムは、不揮発性の第 1 メモリと、第 2 メモリと、第 2 メモリに格納される複数のキャッシュデータのそれぞれを、複数のキャッシュデータを含むセグメントの単位で管理するコントローラと、を備える。複数のキャッシュデータのそれぞれは、メモリシステムがホストに提供する論理アドレス空間内の位置を示す論理アドレス値と第 1 メモリ内の位置とを対応付ける対応情報を含む。セグメントには、少なくとも 2 つのキャッシュデータが、間にスペースを空けずに配置される。

【図面の簡単な説明】

【0008】

【図 1】第 1 の実施形態にかかるメモリシステムの構成の一例を示す図。

【図 2】第 1 の実施形態にかかるメモリシステムに保持される各種情報を示す図。

【図 3】第 1 の実施形態にかかる LUT セグメント管理情報のデータ構造を示す図。

【図 4】第 1 の実施形態にかかる LUT セグメントの例を示す図。

【図 5】第 1 の実施形態にかかる LUT セグメントを DRAM へ書き込む処理手順を示すフローチャート。

【図 6】DRAM から LUT セグメントを読み出す処理手順を示すフローチャート。

10

20

30

40

50

【図 7】DRAM のキャッシュを更新する処理手順を示すフローチャート。

【図 8】第 2 の実施形態にかかる「packed」方式の誤り訂正フレームの構成例を示す図。

【図 9】第 2 の実施形態にかかる「aligned」方式の誤り訂正フレームの構成例を示す図。

【図 10】第 2 の実施形態にかかる LUT セグメントの制御処理手順を示すフローチャート。

【図 11 A】第 2 の実施形態にかかる「packed」方式の誤り訂正フレームの例を示す図。

【図 11 B】第 2 の実施形態にかかる「aligned」方式の誤り訂正フレームの例を示す図。 10

【図 12】第 2 の実施形態にかかる「mixed」方式の誤り訂正フレームの構成例を示す図。

【発明を実施するための形態】

【0009】

以下に添付図面を参照して、実施形態にかかるメモリシステムを詳細に説明する。なお、これらの実施形態により本発明が限定されるものではない。

【0010】

(第 1 の実施形態)

図 1 は、第 1 の実施形態によるメモリシステムの構成の一例を示す図である。このメモリシステム 1 は、例えば、SSD (Solid State Drive) である。以下では、不揮発性メモリとして NAND 型のフラッシュメモリ (以下、NAND メモリという) が用いられる場合を例に挙げる。 20

【0011】

メモリシステム 1 は、ホスト 2 と接続可能に構成されている。ホスト 2 は、例えば CPU (Central Processing Unit)、パーソナルコンピュータ、携帯用情報機器、デジタルスチルカメラ、サーバ、である。メモリシステム 1 とホスト 2 との間の通信のインタフェース規格としては、任意のインタフェース規格が採用可能である。2 以上のホスト 2 がメモリシステム 1 に同時に接続されてもよい。ホスト 2 とメモリシステム 1 とは、ネットワークを介して接続されてもよい。 30

【0012】

メモリシステム 1 は、ホスト 2 からのアクセスコマンドに応じてホスト 2 との間でデータの送受信を実行する。アクセスコマンドは、ライトコマンドおよびリードコマンドを含む。アクセスコマンドは、論理アドレスを含む。論理アドレスは、メモリシステム 1 がホスト 2 に提供する論理アドレス空間内における位置を示す位置情報である。論理アドレスとしては、例えば LBA (Logical Block Address) が採用可能である。

【0013】

メモリシステム 1 は、コントローラ 10 と、NAND メモリ 12 と、DRAM (Dynamic Random Access Memory) 18 とを備える。コントローラ 10 は、SoC (System-On-a-Chip) のような半導体チップで構成される。コントローラ 10 は、ホストバスを介して接続されたホスト 2 から受信した命令に応じて、NAND メモリ 12 にアクセスする。なお、コントローラ 10 は、複数のチップで構成されてもよい。また、DRAM 18 とコントローラ 10 とが一体に構成されてもよい。 40

【0014】

コントローラ 10 は、ホストインタフェース部 11 と、NAND コントローラ 13 と、SRAM (Static Random Access Memory) 14 と、制御部 15 と、DRAM コントローラ 17 とを備える。DRAM コントローラ 17 は、ECC (Error Checking and Correction) 回路 16 を含む。

【0015】

制御部 15 は、例えば、1 以上のプロセッサを含んで構成される。制御部 15 は、予めメモリシステム 1 内の所定の位置に格納されているプログラムを実行することによって、メモリシステム 1 の制御を実行する。プログラムの格納位置は任意に設計される。例えば、プログラムは、予め NAND メモリ 12 に格納され、メモリシステム 1 の起動時に DRAM 18 にロードされる。制御部 15 は、DRAM 18 にロードされたプログラムを実行する。メモリシステム 1 は、複数の処理によって制御される。制御部 15 が実行する各処理については後述する。

【0016】

コントローラ 10 の各部の機能は、制御部 15 がプログラムを実行することにより実現されてもよい。また、コントローラ 10 の各部の機能は、専用ハードウェア回路によって

10

【0017】

ホストインタフェース部 11 は、メモリシステム 1 がホスト 2 と通信を行うためのインタフェース装置である。例えばホストインタフェース部 11 は、制御部 15 の制御下で、ホスト 2 との間でユーザデータを含む情報の転送を実行する。

【0018】

NAND コントローラ 13 は、NAND メモリ 12 に対するアクセスのためのインタフェース装置である。NAND コントローラ 13 は、制御部 15 による制御下で、NAND メモリ 12 との間でユーザデータを含む情報の転送を実行する。NAND コントローラ 13 は、ユーザデータに関する誤り訂正処理を行うことができる。

20

【0019】

SRAM 14 は、第 3 メモリの一例である。SRAM 14 は、コントローラ 10 内の回路として実装され得る。あるいは、SRAM 14 は、コントローラ 10 が実装されるパッケージ内の回路として実装され得る。SRAM 14 は、DRAM 18 よりアクセススピードが速いがビット当たりの単価が高い揮発性メモリである。SRAM 14 は、DRAM 18 より記憶容量が小さく構成され得る。

【0020】

DRAM コントローラ 17 は、DRAM 18 に対するアクセスのためのインタフェース装置である。DRAM コントローラ 17 は、制御部 15 による制御下で、DRAM 18 との間でアドレス変換情報を含む情報の転送を実行する。

30

【0021】

ECC 回路 16 は、DRAM 18 に格納される情報に対して誤り訂正のための符号化を実行したり、DRAM 18 から読み出された情報に対して誤り訂正を実行したりする。なお、ECC 回路 16 は、DRAM コントローラ 17 の外に配置されてもよい。

【0022】

NAND メモリ 12 は、第 1 メモリの一例である。NAND メモリ 12 は、不揮発性の記憶媒体である。NAND メモリ 12 は、1 以上のチップによって構成され得る。

【0023】

NAND メモリ 12 を構成するチップの内部において、データの記憶領域は、複数のブロックによって構成される。各ブロックは、複数のページによって構成される。ページは、データのライトおよびリードが可能な単位である。ブロックは、一括してデータの消去が可能な最小単位である。1 ブロックは、 n (n は自然数) 個のページで構成される。

40

【0024】

1 ページより小さい単位に対して物理的な位置情報 (物理アドレス) が割り当てられる。ここでは、物理アドレスが割り当てられる単位を、クラスタと表記する。1 クラスタのサイズは、ホスト 2 からの最小のアクセス単位 (例えばセクタ) と等しくてもよいし異なってもよい。

【0025】

DRAM 18 は、第 2 メモリの一例である。DRAM 18 は、揮発性メモリパッケージとして実装され得る。DRAM 18 は、コントローラ 10 内の SRAM 14 よりアクセス

50

スピードが遅いがビット当たりの単価が安い揮発性メモリである。DRAM 18は、SRAM 14より記憶容量が大きく構成され得る。

【0026】

図2は、メモリシステム1に保持される各種情報を示す図である。NANDメモリ12には、複数のユーザデータ121が格納される。各ユーザデータ121は、ホスト2から書き込みを要求されたデータである。各ユーザデータ121は、1クラスタのサイズと同じサイズを有することとする。また、NANDメモリ12は、複数のLUTセグメント122と、LUTセグメント管理情報123とを記憶する。LUTセグメント122は、アドレス変換情報120を構成する情報である。

【0027】

アドレス変換情報120は、論理アドレスとNANDメモリ12内の位置を示す位置情報を対応付けるための情報である。NANDメモリ12内の位置を示す位置情報は、物理アドレスと称される。論理アドレスと物理アドレスとの対応関係は、クラスタ単位でアドレス変換情報120に記録される。つまり、アドレス変換情報120は、1つのクラスタに関する論理アドレス値と物理アドレス値との対応関係を示す情報（以降、対応情報と表記する）を、少なくとも論理アドレス空間内の使用中の全ての論理アドレスについて集めたものである。

【0028】

対応情報は、ユーザデータ121がNANDメモリ12に書き込まれる際に更新され、ユーザデータ121がNANDメモリ12から読み出される際に参照される。

【0029】

例えば、コントローラ10は、第1のユーザデータ121をNANDメモリ12内の第1の位置に格納する。コントローラ10は、当該第1のユーザデータ121の書き込みの際に指定された論理アドレス値（第1の論理アドレス値と表記する）と、前記第1の位置を指す物理アドレス値とを対応付けるように、第1の論理アドレスにかかる対応情報を生成する。その後、第1の論理アドレス値と同一の論理アドレス値の指定とともに第2のユーザデータ121がホスト2から書き込み要求されると、コントローラ10は、当該第2のユーザデータ121を前記第1の位置と異なる第2の位置に格納する。このとき、コントローラ10は、第1の論理アドレスにかかる対応情報の更新を行う。つまり、コントローラ10は、第1の論理アドレスにかかる対応情報を、第1の位置を指す物理アドレス値

【0030】

また、コントローラ10は、ホスト2から或る論理アドレス値（第2の論理アドレス値と表記する）の指定とともにユーザデータ121を読み出す要求を受信すると、第2の論理アドレス値にかかる対応情報を参照することによって、第2の論理アドレス値に対応付けられた物理アドレス値を特定する。そして、コントローラ10は、特定した物理アドレス値が指す位置からユーザデータ121を読み出す。

【0031】

NANDメモリ12へのアクセスは、比較的長い時間を要する。よって、コントローラ10が、ホスト2からの書き込みまたは読み出しの要求の度にNANDメモリ12に格納されたアドレス変換情報120を直接に参照したり更新したりすると、ホスト2から要求された動作に要する時間が長くなる。第1の実施形態では、アドレス変換情報120の一部または全部がDRAM18にキャッシュされ、DRAM18にキャッシュされたアドレス変換情報120が参照されたり更新されたりする。具体的には、コントローラ10は、DRAM18にキャッシュ領域180をアロケートし、当該キャッシュ領域180に、アドレス変換情報120のうちの一部または全部の対応情報を格納する。DRAM18のキャッシュ領域180は、第1領域の一例である。コントローラ10は、キャッシュ領域180への対応情報の格納およびキャッシュ領域180からNANDメモリ12への対応情報の書き戻しを、複数の対応情報を含む群の単位で実行する。各LUTセグメント122は、この群に対応する複数の対応情報を含む。すなわち、コントローラ10は、LUTセ

10

20

30

40

50

グメント 1 2 2 の単位で複数の対応情報を管理する。

【 0 0 3 2 】

なお、アドレス変換情報 1 2 0 は、ルックアップテーブル (L U T) のデータ形式で構成されていることとする。つまり、アドレス変換情報 1 2 0 は、対応情報が論理アドレスの順番で記録されたデータ形式で、論理アドレスと物理アドレスとの対応関係が記録されている。よって、各 L U T セグメント 1 2 2 は、論理アドレスが連続する複数のクラスタの対応情報を集めたデータ構造を有する。以降、キャッシュ領域 1 8 0 内の L U T セグメント 1 2 2 を、L U T セグメント 1 8 1 と表記する。

【 0 0 3 3 】

L U T セグメント管理情報 1 2 3 は、各 L U T セグメント 1 2 2 の格納位置を管理するための情報である。すなわち、L U T セグメント管理情報 1 2 3 は、各 L U T セグメント 1 2 2 が、N A N D メモリ 1 2 内のどの位置に格納されているか、または、キャッシュ領域 1 8 0 内のどの位置に格納されているか、を管理するための情報である。コントローラ 1 0 は、L U T セグメント 1 2 2 と同様に、L U T セグメント管理情報 1 2 3 を D R A M 1 8 に格納し、D R A M 1 8 内の L U T セグメント管理情報 1 2 3 を参照したり更新したりする。以降、D R A M 1 8 内の L U T セグメント管理情報 1 2 3 を、L U T セグメント管理情報 1 8 2 と表記する。

10

【 0 0 3 4 】

コントローラ 1 0 は、さらに、L U T セグメント管理情報 1 8 2 を用いて、各 L U T セグメント 1 2 2 がキャッシュ領域 1 8 0 にキャッシュされている位置を管理する。

20

【 0 0 3 5 】

図 3 は、実施形態にかかる L U T セグメント管理情報 1 8 2 のデータ構造の一例を示す模式的な図である。本図に示すように、L U T セグメント管理情報 1 8 2 は、L U T セグメント番号と、インデックスと、N A N D アドレスとを含む情報を L U T セグメント 1 2 2 毎に集めたテーブルである。

【 0 0 3 6 】

L U T セグメント番号は、L U T セグメント 1 2 2 を識別するデータであり、L U T セグメント 1 2 2 毎に一意となるデータである。インデックスは、L U T セグメント番号によって示される L U T セグメント 1 2 2 がキャッシュ領域 1 8 0 に格納されている場合に、キャッシュ領域 1 8 0 における当該 L U T セグメント 1 2 2 の格納位置を示す情報である。インデックスは、例えば、キャッシュ領域 1 8 0 内の先頭を基準としたキャッシュ順によってキャッシュ領域 1 8 0 における位置を表す。なお、キャッシュ領域 1 8 0 内での位置を表す方法はこれに限定されない。キャッシュ領域 1 8 0 内での位置は、D R A M 1 8 にマッピングされたアドレス情報を用いて表されてもよい。N A N D アドレスは、L U T セグメント番号によって示される L U T セグメント 1 2 2 が N A N D メモリ 1 2 に格納されている場合に、N A N D メモリ 1 2 における当該 L U T セグメント 1 2 2 の格納位置を示す物理アドレス値である。

30

【 0 0 3 7 】

なお、L U T セグメント管理情報 1 2 3 のデータ構造と、L U T セグメント管理情報 1 8 2 のデータ構造とは、互いに相違していてもよい。例えば、L U T セグメント管理情報 1 2 3 は、キャッシュ領域 1 8 0 における L U T セグメント 1 2 2 の格納位置を表す情報 (すなわちインデックス) を含んでいなくてもよい。

40

【 0 0 3 8 】

一般に、キャッシュされている対応情報の情報量が多いほど、キャッシュヒット率が高くなるため、メモリシステムの性能が向上する。よって、できるだけ多くの対応情報をキャッシュしたいという要望がある。一方、実装面積、配線、またはコストの観点から、メモリシステムに搭載される D R A M の容量を削減したいという要望がある。

【 0 0 3 9 】

これらの相反する要望に応えるために、第 1 の実施形態では、所定サイズの各 L U T セグメント 1 2 2 に記録される対応情報の量ができるだけ多くなるよう、各 L U T セグメン

50

ト 1 2 2 内に複数の対応情報が配置される。よって、D R A M 1 8 にキャッシュされる対応情報の情報量を多くすることができる。これにより、D R A M 1 8 の容量の削減と性能向上とを両立させることが可能である。

【 0 0 4 0 】

以降、1つのL U Tセグメント1 2 2（および1つのL U Tセグメント1 8 1）に記録された1つの対応情報を、エントリ、と表記することがある。

【 0 0 4 1 】

図 4 は、第 1 の実施形態にかかる L U Tセグメント 1 2 2、1 8 1 のデータ構造の一例を示す図である。なお、本図では、キャッシュ領域 1 8 0 にキャッシュされた L U Tセグメント 1 8 1 のデータ構造について説明するが、このデータ構造は L U Tセグメント 1 2 2 と同様である。

10

【 0 0 4 2 】

図 4 の例では、1つのL U Tセグメント1 8 1のサイズは、例えば、2 5 6 バイト（4 バイト×6 4）である。図 4 の横方向一行の4 バイトは、D R A M 1 8 とコントローラ 1 0 との間のデータ転送の幅に対応する。なお、D R A M 1 8 とコントローラ 1 0 との間のデータ転送の幅はこれに限定されない。

【 0 0 4 3 】

図 4 に示すように、L U Tセグメント 1 8 1 は、エントリ 3 0 1 a ~ エントリ 3 0 1 e 等の複数のエントリと、訂正ビット 3 0 2 a ~ 訂正ビット 3 0 2 e 等の訂正ビットと、リザーブ領域 3 0 3 とを含む。リザーブ領域 3 0 3 は、訂正ビットもエントリのデータも配置されていない領域である。1 エントリのサイズは、3 2 ビットとする。また、訂正ビットのサイズは、1 ビットとする。

20

【 0 0 4 4 】

図 4 に示すように、L U Tセグメント 1 2 2 は、4 バイトの情報毎に、訂正ビットである水平パリティを含む。この水平パリティを含む 4 バイトの情報を、単位情報と表記する。つまり、図 4 に示される例では、1つのL U Tセグメントは 6 4 個の単位情報によって構成されている。

【 0 0 4 5 】

水平パリティは E C C 回路 1 6 によって生成される。E C C 回路 1 6 は、さらに、6 4 個の単位情報に基づいて垂直パリティを生成し、D R A M 1 8 に垂直パリティ情報 1 8 3 として格納する。E C C 回路 1 6 は、D R A M 1 8 から L U Tセグメント 1 8 1 が読み出された際、当該 L U Tセグメント 1 8 1 に含まれる水平パリティの群と、当該 L U Tセグメント 1 8 1 に対応する垂直パリティ情報 1 8 3 と、に基づいて当該 L U Tセグメント 1 8 1 に対する誤り訂正を実行する。

30

【 0 0 4 6 】

なお、本図の例では、単位情報のサイズと、D R A M 1 8 とコントローラ 1 0 との間のデータ転送の幅とは等しいこととしている。しかしながら、単位情報のサイズと、D R A M 1 8 とコントローラ 1 0 の間とのデータ転送の幅とは等しくなくてもよい。

【 0 0 4 7 】

実施形態では、一例として、L U Tセグメント 1 8 1 には、連続する論理アドレスにそれぞれ対応する複数の物理アドレスが、対応する論理アドレスの順番で記録されている、としている。つまり、各エントリには、物理アドレス値が記録されている。例えば、エントリ i には、或る論理アドレス値 $A d r _ i$ に対応する物理アドレス値が記録されており、エントリ $i + 1$ には、論理アドレス値 $A d r _ i$ に後続する論理アドレス値 $A d r _ i + 1$ に対応する物理アドレス値が記録されている。なお、各エントリのデータ構造はこれに限定されない。各エントリとして、論理アドレス値と物理アドレス値とが対にして記録されてもよい。

40

【 0 0 4 8 】

物理アドレス値を示すビット列の長さは、ユーザ容量に依存する。ユーザ容量が大きいほど、物理アドレス値を示すビット列の長さが長い。図 4 に示された例では、物理アドレ

50

ス値を示すビット列の長さが31ビットよりも長い32ビットであることとしている。つまり、物理アドレス値を示すビット列の長さが、31(4バイト×8-1)ビットと一致しておらず、かつ、31ビットよりも長い。したがって、1つのエントリは2つの単位情報に分けて配置される。

【0049】

LUTセグメント181内では、連続する2つの論理アドレス値にかかる2つのエントリは、互いに隣接するようにDRAM18上に配置される。つまり、1つのエントリが配置された領域の末尾と次のエントリが配置された領域の先頭との間にスペースを空けないように、複数のエントリが順次配置される。よって、64個の単位情報のうちの水平パリティを除く部分には、多くのエントリが隙間なく配置されている。そして、第1の実施形態では、多くのエントリが64個の単位情報のうちの水平パリティを除く部分に隙間無く配置された結果、末尾の単位情報に1つのエントリのサイズ未満のリザーブ領域303が生じている。

10

【0050】

ここで、比較例にかかるLUTセグメントの例を説明する。比較例にかかるLUTセグメントによれば、1バイトの整数倍の単位で単位情報が構成され、かつ、1つの単位情報には1つのエントリのみが含まれる。よって、例えば物理アドレス値を示すビット列の長さが(Nバイト×8-1)ビットと一致しない場合、各単位情報は、リザーブ領域を含むことになる。1つのLUTセグメントは、単位情報毎にリザーブ領域を有することになり、1つのLUTセグメントに含まれるリザーブ領域のトータルサイズが第1の実施形態よりも大きい。そのため、比較例にかかるLUTセグメントによれば、LUTセグメントのサイズに対する正味の対応情報の情報量が第1の実施形態よりも少ない。

20

【0051】

これに対し、第1の実施形態によれば、1つのエントリが配置された領域の末尾と次のエントリが配置された領域の先頭との間にスペースを空けずに、複数のエントリが順次配置される。そのため、物理アドレス値を示すビット列の長さが(Nバイト×8-1)ビットと一致しない場合であっても、単位情報毎のリザーブ領域は生じない。よって、比較例に比べ、LUTセグメントのサイズに対する正味の対応情報の情報量を多くすることが可能である。したがって、第1の実施形態によれば、比較例に比べ、決められたサイズのキャッシュ領域にキャッシュ可能な対応情報の量が多い。即ち、第1の実施形態によれば、アドレス変換情報120を効率よくキャッシュ領域180にキャッシュできる。

30

【0052】

次に、図5～図7を用いて、LUTセグメントの制御処理手順を説明する。

【0053】

図5は、LUTセグメント122をDRAM18へ書き込む処理手順を示すフローチャートである。コントローラ10の制御部15は、所定のタイミング(例えばキャッシュミスした場合など)に、NANDメモリ12に格納されているLUTセグメント122(キャッシュ対象LUTセグメントと表記する)をリードする(ステップS1)。

【0054】

続いて、ECC回路16は、NANDメモリ12から読み出されたキャッシュ対象LUTセグメント122から単位情報毎の水平パリティを生成する(ステップS2)。そして、DRAMコントローラ17は、単位情報毎の水平パリティとキャッシュ対象LUTセグメント122とを、DRAM18のキャッシュ領域180にキャッシュ対象LUTセグメント181として格納する(ステップS3)。

40

【0055】

ECC回路16は、キャッシュ対象LUTセグメント122の全ての単位情報に基づいて垂直パリティを生成し、生成した垂直パリティをDRAM18に垂直パリティ情報183として格納する(ステップS4)。

【0056】

制御部15は、キャッシュ領域180内のキャッシュ対象LUTセグメント181の位

50

置を表すインデックスを L U T セグメント管理情報 1 8 2 に記録する (ステップ S 5)。そして、L U T セグメントを D R A M 1 8 へ書き込む処理が終了する。

【 0 0 5 7 】

図 6 は、D R A M 1 8 から L U T セグメント 1 8 1 を読み出す処理手順を示すフローチャートである。

【 0 0 5 8 】

まず、制御部 1 5 は、D R A M コントローラ 1 7 を制御して、読み出し対象エントリを含む L U T セグメント 1 8 1 を読み出す (ステップ S 1 1)。読み出し対象エントリとは、読み出し対象のユーザデータの物理アドレス値を含むエントリである。このとき、E C C 回路 1 6 は、D R A M 1 8 から読み出された L U T セグメント 1 8 1 に対し、水平パリティおよび垂直パリティ情報 1 8 3 に基づいて誤り訂正を実行する (ステップ S 1 2)。制御部 1 5 は、誤り訂正の実行後の L U T セグメント 1 8 1 を S R A M 1 4 に格納する (ステップ S 1 3)。そして、制御部 1 5 は、S R A M 1 4 内の L U T セグメント 1 8 1 のうちの読み出し対象エントリをリードする (ステップ S 1 4)。そして、D R A M 1 8 から L U T セグメントを読み出す処理が終了する。

10

【 0 0 5 9 】

なお、読み出し対象エントリを含む L U T セグメント 1 8 1 が既に S R A M 1 4 に格納されている場合には、コントローラ 1 0 は、S 1 1 から S 1 3 までの処理を省略し、S 1 4 の処理を実行することができる。

【 0 0 6 0 】

20

図 7 は、D R A M 1 8 のキャッシュを更新する処理手順を示すフローチャートである。

【 0 0 6 1 】

コントローラ 1 0 の制御部 1 5 は、D R A M コントローラ 1 7 を制御して、更新対象エントリを含む L U T セグメント 1 8 1 を読み出す (ステップ S 2 1)。更新対象エントリとは、書き込み対象のユーザデータの物理アドレス値を含むエントリである。このとき、E C C 回路 1 6 は、L U T セグメント 1 8 1 に対し、水平パリティおよび垂直パリティ情報 1 8 3 に基づいて誤り訂正を実行する (ステップ S 2 2)。制御部 1 5 は、誤り訂正の実行後の L U T セグメント 1 8 1 を S R A M 1 4 に格納する。そして、制御部 1 5 は、S R A M 1 4 において、更新対象のエントリを含む L U T セグメント 1 8 1 である更新対象 L U T セグメント 1 8 1 を更新する (ステップ S 2 3)。

30

【 0 0 6 2 】

コントローラ 1 0 は、例えば S R A M 1 4 の空きが少なくなった場合、ステップ S 2 4 以降の処理を実行する。E C C 回路 1 6 は、更新対象 L U T セグメント 1 8 1 から単位情報毎の水平パリティを生成する。E C C 回路 1 6 は、生成した水平パリティを更新対象 L U T セグメント 1 8 1 の対応する訂正ビットに上書きする (ステップ S 2 4)。D R A M コントローラ 1 7 は、上書きされた水平パリティと、更新対象 L U T セグメント 1 8 1 とを、D R A M 1 8 のキャッシュ領域 1 8 0 に格納する (ステップ S 2 5)。E C C 回路 1 6 は、更新対象 L U T セグメント 1 8 1 の全ての単位情報に基づいて垂直パリティを生成し、生成した垂直パリティを D R A M 1 8 に垂直パリティ情報 1 8 3 として格納する (ステップ S 2 6)。制御部 1 5 は、更新対象 L U T セグメント 1 8 1 の位置に関する情報を L U T セグメント管理情報 1 8 2 に記録する (ステップ S 2 7)。制御部 1 5 は、例えばメモリシステム 1 のシャットダウン時に、更新対象 L U T セグメント 1 8 1 を N A N D メモリ 1 2 へ書き込む (ステップ S 2 8)。

40

【 0 0 6 3 】

このように、第 1 の実施形態によれば、制御部 1 5 は、1 つのエントリが配置された領域の末尾と次のエントリが配置された領域の先頭との間にスペースを空けないように複数のエントリが順次配置された、L U T セグメント 1 8 1 を使用する。そのため、物理アドレス値を示すビット列の長さが (N バイト × 8 - 1) ビットと一致しない場合であっても、第 1 の実施形態によれば、単位情報毎のリザーブ領域が生じない。よって、上述の比較例に比べ、L U T セグメントのサイズに対する正味の対応情報の量を多くすることが可能

50

である。その結果、第 1 の実施形態によれば、比較例に比べ、決められたサイズのキャッシュ領域にキャッシュ可能な対応情報の量が多い。即ち、第 1 の実施形態によれば、アドレス変換情報 1 2 0 を効率よくキャッシュ領域 1 8 0 にキャッシュできる。

【 0 0 6 4 】

また、第 1 の実施形態によれば、D R A M 1 8 は、各 L U T セグメントの位置を示す L U T セグメント管理情報 1 8 2 を記憶する。上述のように、1 つの L U T セグメントに含まれる対応情報の量を多くすることができるので、1 つのアドレス変換情報 1 2 0 を構成する L U T セグメントの数を抑制することができる。その結果、D R A M 1 8 に一時格納される L U T セグメント管理情報 1 8 2 のサイズを抑制できる。よって、メモリシステム 1 に搭載される D R A M 1 8 の容量を削減でき、その結果としてメモリシステム 1 のコストを低減できる。

【 0 0 6 5 】

また、コントローラ 1 0 は、D R A M 1 8 に格納された L U T セグメント 1 8 1 を、S R A M 1 4 に格納し、L U T セグメント 1 8 1 のエントリを更新して、更新後の L U T セグメント 1 8 1 を D R A M 1 8 へ格納する。上述のように、L U T セグメント 1 8 1 により多くのエントリを格納しているので、メモリシステム 1 は、S R A M 1 4 へ格納する L U T セグメント 1 8 1 の数を低減させることができる。

【 0 0 6 6 】

また、コントローラ 1 0 は、D R A M 1 8 に格納された L U T セグメント 1 8 1 を、L U T セグメント単位で、N A N D メモリ 1 2 へ格納する。上述のように、L U T セグメント 1 8 1 により多くのエントリを格納しているので、メモリシステム 1 は、D R A M 1 8 から N A N D メモリ 1 2 へ格納する L U T セグメント 1 8 1 の数を低減させることができ、処理負荷を軽減させることができる。

【 0 0 6 7 】

(第 2 の実施形態)

第 2 実施形態では、誤り訂正フレームに基づいて、複数のエントリを格納する処理が制御される。例えば、D R A M 1 8 上で誤り訂正処理をするために、複数のエントリを含む誤り訂正フレームが生成される。誤り訂正フレームは、例えば 6 4 バイト (5 1 2 ビット) のサイズを有し、その内、1 9 ビットは E C C パリティである。本実施形態では、1 つの L U T セグメントは、2 5 6 バイトのサイズを有するものとする。この 2 5 6 バイトの L U T セグメントは、第 1 の実施形態で説明したようなリザーブ領域を含む。

【 0 0 6 8 】

1 つの誤り訂正フレームに、複数のエントリを格納することを考える。誤り訂正フレームにおける E C C パリティ以外のサイズが、エントリのサイズの整数倍でない場合、誤り訂正フレームに、リザーブ領域が生じることになる。

【 0 0 6 9 】

そこで、誤り訂正フレームにエントリをより多く格納する方法として、1 つの誤り訂正フレームが複数の L U T セグメントに含まれるエントリを格納する方法 (「 p a c k e d 」 方式) が考えられる。

【 0 0 7 0 】

例えば、図 8 に示すように、D R A M 1 8 に誤り訂正フレーム 5 0 1 a ~ 誤り訂正フレーム 5 0 1 h が格納されているものとする。誤り訂正フレーム 5 0 1 a ~ 誤り訂正フレーム 5 0 1 d はそれぞれ、L U T セグメント 1 8 1 a の一部と、E C C パリティ 5 1 1 a ~ E C C パリティ 5 1 1 d とを格納する。誤り訂正フレーム 5 0 1 e は、L U T セグメント 1 8 1 a の他の一部と、L U T セグメント 1 8 1 b の一部と、E C C パリティ 5 1 1 e とを格納する。誤り訂正フレーム 5 0 1 f ~ 誤り訂正フレーム 5 0 1 h はそれぞれ、L U T セグメント 1 8 1 b の他の一部と、E C C パリティ 5 1 1 f ~ E C C パリティ 5 1 1 h とを格納する。

【 0 0 7 1 】

一方で、L U T セグメントの先頭が、誤り訂正フレームの先頭にアラインするように、

10

20

30

40

50

1つの誤り訂正フレームに複数のエントリを格納する方法も考えられる(「aligned」方式)。この場合、LUTセグメントの先頭を、誤り訂正フレームの先頭にアラインさせるため、誤り訂正フレーム内にリザーブ領域が生じる。「aligned」方式では、1つの誤り訂正フレームが複数のLUTセグメントを格納しないので、「packed」方式に比べ、LUTセグメントの格納または読み出しの処理を、簡素化することができる。

【0072】

例えば、図9に示すように、DRAM18に誤り訂正フレーム601a~誤り訂正フレーム601hが格納されているものとする。誤り訂正フレーム601a~誤り訂正フレーム601dはそれぞれ、LUTセグメント181aの一部と、ECCパリティ611a~ECCパリティ611dとを格納する。誤り訂正フレーム601eは、LUTセグメント181aの他の一部と、リザーブ領域621eと、ECCパリティ611eとを格納する。誤り訂正フレーム601f~誤り訂正フレーム601hはそれぞれ、LUTセグメント181bの一部と、ECCパリティ611f~ECCパリティ611hとを格納する。LUTセグメント181bの先頭は、誤り訂正フレーム601fの先頭にアラインしている。

10

【0073】

図8に示す「packed」方式の場合、誤り訂正フレーム501eにリザーブ領域がないので、図9に示す「aligned」方式と比較してより多くのエントリを、所定の領域に格納することができる。

20

【0074】

例えば、DRAM18のキャッシュ領域180のように、領域のサイズが限られている場合、領域を効率よく使用するために「packed」方式を適用し、NANDメモリ12へLUTセグメントを格納する場合、「aligned」方式に変換して格納してもよい。

【0075】

図10および図11を用いて、「packed」方式でDRAM18に格納したLUTセグメントを、「aligned」方式に変換してNANDメモリ12に格納する処理手順を説明する。

【0076】

図10は、第2の実施形態にかかるLUTセグメントの制御処理手順を示すフローチャートである。まず、制御部15は、「aligned」方式に変換したLUTセグメントを一時的に格納するために、DRAM18上でメモリ領域を確保する(ステップS31)。そして、制御部15は、「packed」方式で格納されたLUTセグメントをリードして、LUTセグメントのデータをシフトする(ステップS32)。

30

【0077】

ここで、図11Aおよび図11Bを参照して、「packed」方式から「aligned」方式に変換する例を説明する。図11Aは、DRAM18に「packed」方式で格納されている2つの誤り訂正フレーム800a、800bを示す。図11Aに示すように、「packed」形式の場合、誤り訂正フレーム800aは、エントリ801a~エントリ801d、およびエントリ801eの一部、並びにECCパリティ811aを格納する。エントリ801a~エントリ801dは、LUTセグメント181aに属する。エントリ801eは、LUTセグメント181bに属する。エントリ801eは、LUTセグメント181bの先頭のエントリである。誤り訂正フレーム800bは、エントリ801eの他の一部、エントリ801f~エントリ801g、および、ECCパリティ811bを格納する。エントリ801f~エントリ801gは、LUTセグメント181bに属する。

40

【0078】

図11Bは、DRAM18に「aligned」方式で一時的に格納される2つの誤り訂正フレーム800a、800bを示す。制御部15は、「aligned」方式に変換

50

するため、エン트리 801e のデータをシフトさせ、誤り訂正フレーム 800b の先頭とエン트리 801e の先頭（すなわち LUT セグメント 181b の先頭）とをアラインさせる。この結果、誤り訂正フレーム 800a は、エン트리 801a ~ エン트리 801d、リザーブ領域 821a、および、ECC パリティ 811c を含む。なお、ECC パリティ 811c は、エン트리 801a ~ エン트리 801d のデータに基づいて再計算された ECC パリティである。誤り訂正フレーム 800b は、エン트리 801e ~ エン트리 801g、リザーブ領域 821b、および、ECC パリティ 811d を含む。なお、ECC パリティ 811d は、エン트리 801e ~ エン트리 801g のデータに基づいて再計算された ECC パリティである。

【0079】

10

図 10 に戻る。ECC 回路 16 は、シフト後に誤り訂正フレームに格納されるエントリのデータに基づいて ECC パリティを再計算する（ステップ S33）。制御部 15 は、シフト後のエン트리および再計算された ECC パリティに基づいて生成した「aligned」方式の誤り訂正フレームを、NAND メモリ 12 に書き込む（ステップ S34）。

【0080】

「packed」方式と「aligned」方式について上述したが、これらを合わせた「mixed」方式を適用するようにしてもよい。例えば、図 12 に示すように、誤り訂正フレーム 701a および誤り訂正フレーム 701b の間では、「packed」方式を適用してもよい。この場合、誤り訂正フレーム 701a は、LUT セグメント 181c の一部と、ECC パリティ 711a とを含む。誤り訂正フレーム 701b は、LUT セグメント 181c の他の一部と、LUT セグメント 181d の一部と、ECC パリティ 711b とを含む。また、誤り訂正フレーム 701c ~ 誤り訂正フレーム 701e の間では、「aligned」方式を適用するようにしてもよい。この場合、誤り訂正フレーム 701c は、LUT セグメント 181d の他の一部と、リザーブ領域 721c と、ECC パリティ 711c とを含む。誤り訂正フレーム 701d は、LUT セグメント 181e の一部と、ECC パリティ 711d とを含む。誤り訂正フレーム 701e は、LUT セグメント 181e の他の一部と、リザーブ領域 721e と、ECC パリティ 711e とを含む。

20

【0081】

上述の実施形態では、コントローラ 10 は、1つのエントリと、他のエントリの間にスペースを空けずに、エントリを DRAM 18 に格納する。この場合、コントローラ 10 は、より多くのエントリを DRAM 18 に格納することができる。

30

【0082】

また、コントローラ 10 は、「packed」方式の場合、誤り訂正フレームの先頭と LUT セグメントの先頭をアラインすることなく、LUT セグメントを DRAM 18 に格納する。この場合、コントローラ 10 は、より多くの LUT セグメントを DRAM 18 に格納することができる。

【0083】

また、コントローラ 10 は、「aligned」方式の場合、1つの誤り訂正フレームに複数の LUT セグメントを格納しない。従って、LUT セグメントの格納または読み出しの処理を、簡素化することができる。

40

【0084】

本発明のいくつかの実施形態を説明したが、これらの実施形態は、例として提示したものであり、発明の範囲を限定することは意図していない。これら新規な実施形態は、その他の様々な形態で実施されることが可能であり、発明の要旨を逸脱しない範囲で、種々の省略、置き換え、変更を行うことができる。これら実施形態やその変形は、発明の範囲や要旨に含まれるとともに、特許請求の範囲に記載された発明とその均等の範囲に含まれる。

【符号の説明】

【0085】

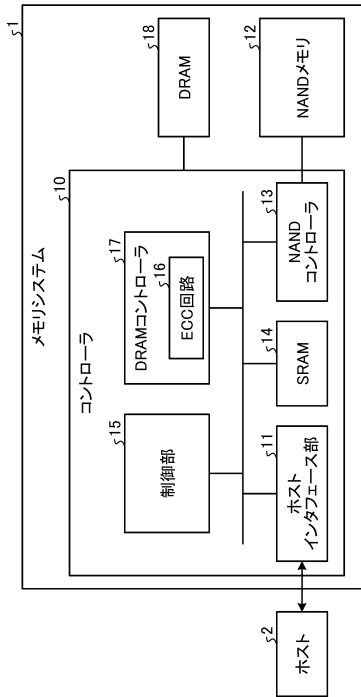
1 メモリシステム、2 ホスト、10 コントローラ、11 ホストインタフェース

50

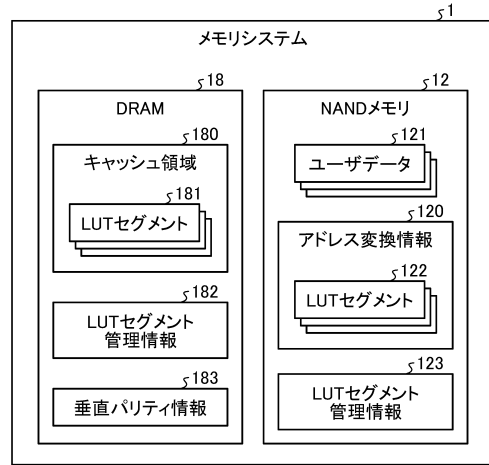
部、12 NANDメモリ、13 NANDコントローラ、14 SRAM、15 制御部、16 ECC回路、17 DRAMコントローラ、18 DRAM、121 ユーザデータ、122 LUTセグメント、123 LUTセグメント管理情報、181 LUTセグメント、182 LUTセグメント管理情報、183 垂直パリティ情報。

【図面】

【図1】



【図2】

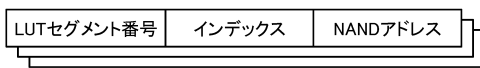


10

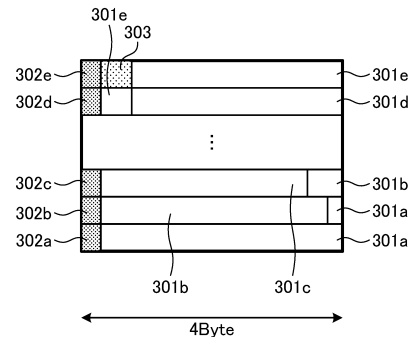
20

30

【図3】



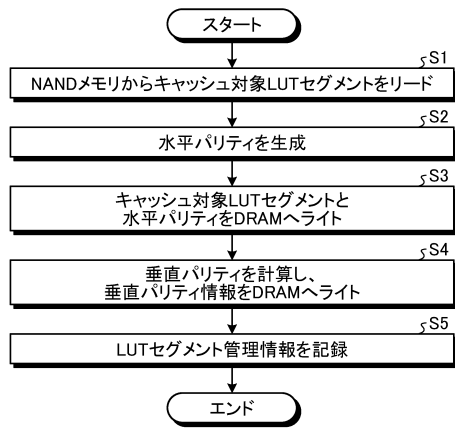
【図4】



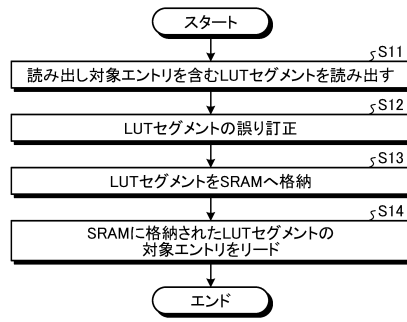
40

50

【 図 5 】



【 図 6 】



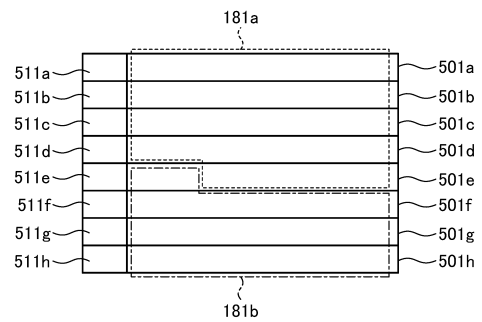
10

20

【 図 7 】



【 図 8 】

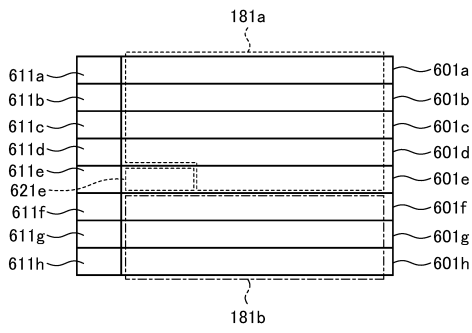


30

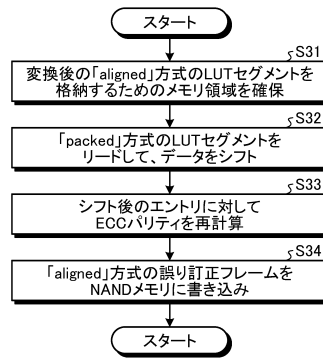
40

50

【図9】



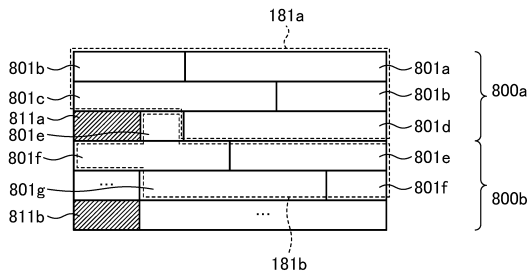
【図10】



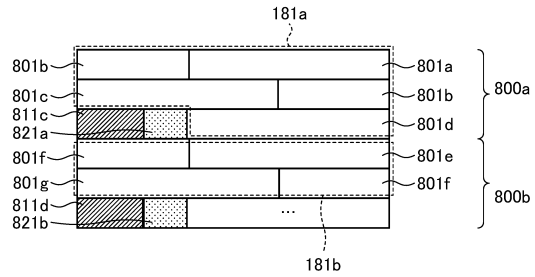
10

20

【図11A】



【図11B】

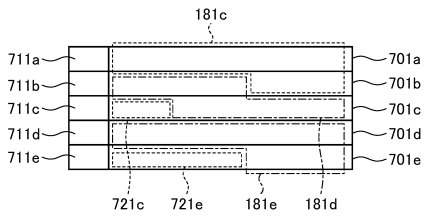


30

40

50

【 図 1 2 】



10

20

30

40

50