(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7091203号
(P7091203)

(45)発行日　令和4年6月27日(2022.6.27)

(24)登録日　令和4年6月17日(2022.6.17)

(51)国際特許分類

		FI		
G06F 12/02 (2006.01)		G06F	12/02	530E
G06F 12/00 (2006.01)		G06F	12/00	597U
G06F 3/08 (2006.01)		G06F	12/00	571A
G06F 3/06 (2006.01)		G06F	3/08	H
		G06F	3/06	304Z

請求項の数　8　（全77頁）　最終頁に続く

(21)出願番号　特願2018-175148(P2018-175148)
(22)出願日　平成30年9月19日(2018.9.19)
(65)公開番号　特開2020-46963(P2020-46963A)
(43)公開日　令和2年3月26日(2020.3.26)
審査請求日　令和3年3月18日(2021.3.18)

(73)特許権者　318010018
キオクシア株式会社
東京都港区芝浦三丁目１番２１号
(74)代理人　110001737
特許業務法人スズエ国際特許事務所
(72)発明者　菅野 伸一
東京都港区芝浦一丁目１番１号 東芝メモリ株式会社内
審査官　酒井 恭信

最終頁に続く

(54)【発明の名称】　メモリシステムおよび制御方法

(57)【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、
複数のブロックを含む不揮発性メモリと、
前記不揮発性メモリに電気的に接続され、前記不揮発性メモリを制御するコントローラとを具備し、
前記コントローラは、
前記不揮発性メモリを論理的に分割することによって得られる複数の領域の中のいずれかの領域に割り当て済みのブロック毎に、各ブロックからのデータの読み出しを伴う未実行または実行中の命令の数を示すカウンタを管理し、
あるブロックを対象とするデータの読み出し処理を実行する場合、そのブロックに対応する前記カウンタの値を１つ加算し、あるブロックを対象とするデータの読み出し処理を終了した場合、そのブロックに対応する前記カウンタの値を１つ減算し、
前記複数の領域の中のいずれかの領域に割り当て済みの第１ブロックを前記複数の領域の中の任意の領域に再割り当て可能な状態に遷移させる命令を前記ホストから受信した際、前記第１ブロックに対応する前記カウンタの値が１以上である場合、前記カウンタの値が０になってから前記第１ブロックを再割り当て可能な状態に遷移させる、
ように構成されているメモリシステム。
【請求項２】
ホストに接続可能なメモリシステムであって、

複数のブロックを含む不揮発性メモリと、
前記不揮発性メモリに電気的に接続され、前記不揮発性メモリを制御するコントローラとを具備し、
前記コントローラは、
前記不揮発性メモリを論理的に分割することによって得られる複数の領域の中のいずれかの領域に割り当て済みのブロック毎に、各ブロックからのデータの読み出しを伴う未実行または実行中の命令の数を示すカウンタを管理し、
あるブロックを対象とするデータの読み出し処理を実行する場合、そのブロックに対応する前記カウンタの値を１つ加算し、あるブロックを対象とするデータの読み出し処理を終了した場合、そのブロックに対応する前記カウンタの値を１つ減算し、
前記複数の領域の中のいずれかの領域に割り当て済みの第２ブロックを前記複数の領域の中の任意の領域に再割り当て可能な状態に遷移させる命令を前記ホストから受信した際、前記第２ブロックに対応する前記カウンタの値が０でない場合、前記第２ブロックを再割り当て可能な状態に遷移させる命令に対する応答としてエラーを前記ホストに通知する、
ように構成されているメモリシステム。
【請求項３】
前記コントローラは、前記複数の領域の中のいずれかの領域に割り当て済みの第３ブロックを前記複数の領域の中の任意の領域に再割り当て可能な状態に遷移させる命令が前記ホストから受信されている状況下において、前記第３ブロックからのデータの読み出しを伴う命令が前記ホストから受信された場合、前記第３ブロックからのデータの読み出しを伴う命令に対する応答としてエラーを前記ホストに通知するように構成されている請求項１または２に記載のメモリシステム。
【請求項４】
前記複数の領域は、各々が複数のブロックを含むＱｏＳドメインである請求項１～３のいずれか１項に記載のメモリシステム。
【請求項５】
あるＱｏＳドメインに割り当て済みのブロックを任意のＱｏＳドメインに再割り当て可能な状態に遷移させる命令は、ＱｏＳドメインＩＤと、前記ＱｏＳドメインＩＤで示されるＱｏＳドメインの中のブロックを示すブロックアドレスと、を含む請求項４に記載のメモリシステム。
【請求項６】
前記コントローラは、
前記不揮発性メモリに含まれる複数のブロックの各々が１つのＱｏＳドメインのみに属するように、前記複数のブロックを複数のＱｏＳドメインに分類し、
前記Ｑｏｓドメイン毎に、未割り当てのブロックのリストと割り当て済みのブロックのリストとを管理する、
ように構成されている請求項４または５に記載のメモリシステム。
【請求項７】
複数のブロックを含む不揮発性メモリを制御するコントローラによって実行される制御方法であって、
前記不揮発性メモリを論理的に分割することによって得られる複数の領域の中のいずれかの領域に割り当て済みのブロック毎に、各ブロックからのデータの読み出しを伴う未実行または実行中の命令の数を示すカウンタを管理し、
あるブロックを対象とするデータの読み出し処理を実行する場合、そのブロックに対応する前記カウンタの値を１つ加算し、あるブロックを対象とするデータの読み出し処理を終了した場合、そのブロックに対応する前記カウンタの値を１つ減算し、
前記複数の領域の中のいずれかの領域に割り当て済みの第１ブロックを前記複数の領域の中の任意の領域に再割り当て可能な状態に遷移させる命令をホストから受信した際、前記第１ブロックに対応する前記カウンタの値が１以上である場合、前記カウンタの値が０になってから前記第１ブロックを再割り当て可能な状態に遷移させる、

制御方法。
【請求項８】
複数のブロックを含む不揮発性メモリを制御するコントローラによって実行される制御方法であって、
前記不揮発性メモリを論理的に分割することによって得られる複数の領域の中のいずれかの領域に割り当て済みのブロック毎に、各ブロックからのデータの読み出しを伴う未実行または実行中の命令の数を示すカウンタを管理し、
あるブロックを対象とするデータの読み出し処理を実行する場合、そのブロックに対応する前記カウンタの値を１つ加算し、あるブロックを対象とするデータの読み出し処理を終了した場合、そのブロックに対応する前記カウンタの値を１つ減算し、
前記複数の領域の中のいずれかの領域に割り当て済みの第２ブロックを前記複数の領域の中の任意の領域に再割り当て可能な状態に遷移させる命令をホストから受信した際、前記第２ブロックに対応する前記カウンタの値が０でない場合、前記第２ブロックを再割り当て可能な状態に遷移させる命令に対する応答としてエラーを前記ホストに通知する、
制御方法。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明の実施形態は、不揮発性メモリを制御する技術に関する。
【背景技術】
【０００２】
近年、不揮発性メモリを備えるメモリシステムが広く普及している。
このようなメモリシステムの一つとして、ＮＡＮＤフラッシュ技術ベースのソリッドステートドライブ（ＳＳＤ）が知られている。
最近では、ホストとストレージとの間の新たなインタフェースが提案され始めている。
【先行技術文献】
【非特許文献】
【０００３】
【文献】Yiying Zhang, 外, "De-indirection for flash-based SSDs with nameless writes." FAST. 2012, [online], [平成29年9月13日検索], インターネット〈URL: https://www.usenix.org/system/files/conference/fast12/zhang.pdf〉
【発明の概要】
【発明が解決しようとする課題】
【０００４】
しかし、一般に、ＮＡＮＤ型フラッシュメモリの制御は複雑であるため、Ｉ／Ｏ性能を改善するための新たなインタフェースの実現に際しては、ホストとストレージ（メモリシステム）との間の適切な役割分担を考慮することが必要とされる。
本発明が解決しようとする課題は、Ｉ／Ｏ性能の改善を図ることができるメモリシステムおよび制御方法を提供することである。
【課題を解決するための手段】
【０００５】
実施形態によれば、ホストに接続可能なメモリシステムは、複数のブロックを含む不揮発性メモリと、前記不揮発性メモリに電気的に接続され、前記不揮発性メモリを制御するコントローラとを含む。前記コントローラは、前記不揮発性メモリを論理的に分割することによって得られる複数の領域の中のいずれかの領域に割り当て済みのブロック毎に、各ブロックからのデータの読み出しを伴う未実行または実行中の命令の数を示すカウンタを管理する。前記コントローラは、あるブロックを対象とするデータの読み出し処理を実行する場合、そのブロックに対応する前記カウンタの値を１つ加算し、あるブロックを対象とするデータの読み出し処理を終了した場合、そのブロックに対応する前記カウンタの値を１つ減算する。前記コントローラは、前記複数の領域の中のいずれかの領域に割り当て済

みの第１ブロックを前記複数の領域の中の任意の領域に再割り当て可能な状態に遷移させる命令を前記ホストから受信した際、前記第１ブロックに対応する前記カウンタの値が１以上である場合、前記カウンタの値が０になってから前記第１ブロックを再割り当て可能な状態に遷移させる。

【図面の簡単な説明】

【０００６】

【図１】ホストと実施形態のメモリシステム（フラッシュストレージデバイス）との関係を示すブロック図。

【図２】従来型ＳＳＤとホストとの間の役割分担と、同実施形態のフラッシュストレージデバイスとホストとの間の役割分担とを説明するための図。

【図３】複数のホストと複数のフラッシュストレージデバイスとの間のデータ転送がネットワーク機器を介して実行される計算機システムの構成例を示すブロック図。

【図４】同実施形態のメモリシステムの構成例を示すブロック図。

【図５】同実施形態のメモリシステムに設けられたＮＡＮＤインタフェースと複数のＮＡＮＤ型フラッシュメモリダイとの関係を示すブロック図。

【図６】複数のブロックの集合によって構築されるスーパーブロックの構成例を示す図。

【図７】ホストが論理アドレスとブロック番号とを指定し且つ同実施形態のメモリシステムがブロック内物理アドレス（ブロック内オフセット）を決定するデータ書き込み動作と、ホストがブロック番号とブロック内物理アドレス（ブロック内オフセット）とを指定するデータ読み出し動作とを説明するための図。

【図８】同実施形態のメモリシステムに適用されるライトコマンドを説明するための図。

【図９】図８のライトコマンドに対するレスポンスを説明するための図。

【図１０】同実施形態のメモリシステムに適用されるＴｒｉｍコマンドを説明するための図。

【図１１】物理アドレスを表す、ブロック番号およびオフセットを説明するための図。

【図１２】ライトコマンドに応じて実行される書き込み動作を説明するための図。

【図１３】不良ページをスキップする書き込み動作を説明するための図。

【図１４】不良ページをスキップする書き込み動作の別の例を説明するための図。

【図１５】論理アドレスとデータのペアをブロック内のページに書き込む動作を説明するための図。

【図１６】データをブロック内のページのユーザデータ領域に書き込み、このデータの論理アドレスをこのページの冗長領域に書き込む動作を説明するための図。

【図１７】スーパーブロックが使用される場合におけるブロック番号とオフセットとの関係を説明するための図。

【図１８】同実施形態のメモリシステムに適用される最大ブロック番号ゲットコマンドを説明するための図。

【図１９】最大ブロック番号ゲットコマンドに対するレスポンスを説明するための図。

【図２０】同実施形態のメモリシステムに適用されるブロックサイズゲットコマンドを説明するための図。

【０００７】

【図２１】ブロックサイズゲットコマンドに対するレスポンスを説明するための図。

【図２２】同実施形態のメモリシステムに適用されるブロックアロケートコマンド（ブロック割り当て要求）を説明するための図。

【図２３】ブロックアロケートコマンドに対するレスポンスを説明するための図。

【図２４】ホストと同実施形態のメモリシステムとによって実行されるブロック情報取得処理を示すシーケンスチャート。

【図２５】ホストと同実施形態のメモリシステムとによって実行される書き込み処理のシーケンスを示すシーケンスチャート。

【図２６】すでに書き込まれているデータに対する更新データを書き込むデータ更新動作を示す図。

【図２７】同実施形態のメモリシステムによって管理されるブロック管理テーブルを更新する動作を説明するための図。
【図２８】ホストによって管理されるルックアップテーブル（論理物理アドレス変換テーブル）を更新する動作を説明するための図。
【図２９】無効化すべきデータに対応するブロック番号および物理アドレスを示すホストからの通知に応じてブロック管理テーブルを更新する動作を説明するための図。
【図３０】同実施形態のメモリシステムに適用されるリードコマンドを説明するための図。
【図３１】同実施形態のメモリシステムによって実行されるリード動作を説明するための図。
【図３２】ホストからのリードコマンドに応じて、異なる物理記憶位置にそれぞれ格納されているデータ部をリードする動作を説明するための図。
【図３３】ホストと同実施形態のメモリシステムとによって実行されるリード処理のシーケンスを示すシーケンスチャート。
【図３４】同実施形態のメモリシステムに適用されるガベージコレクション（ＧＣ）制御コマンドを説明するための図。
【図３５】同実施形態のメモリシステムに適用されるＧＣ用コールバックコマンドを説明するための図。
【図３６】ホストと同実施形態のメモリシステムとによって実行されるガベージコレクション（ＧＣ）動作の手順を示すシーケンスチャート。
【図３７】ガベージコレクション（ＧＣ）のために実行されるデータコピー動作の例を説明するための図。
【図３８】図３７のデータコピー動作の結果に基づいて更新されるホストのルックアップテーブルの内容を説明するための図。
【図３９】ホストと同実施形態のメモリシステムとのシステムアーキテクチャを示す図。
【図４０】同実施形態のメモリシステム上における仮想ストレージデバイスの定義例を示す図。
【０００８】
【図４１】同実施形態のメモリシステム上において仮想ストレージデバイス毎にＱｏＳドメインが管理される例を示す図。
【図４２】同実施形態のメモリシステムのブロックリユースコマンド受信時におけるフラッシュストレージデバイス３の動作手順（第１ケース）を示すフローチャート。
【図４３】同実施形態のメモリシステムのブロックリユースコマンド受信時におけるフラッシュストレージデバイス３の動作手順（第２ケース）を示すフローチャート。
【図４４】同実施形態のメモリシステムによって実行されるＩ／Ｏコマンド処理を説明するためのブロック図。
【図４５】同実施形態のメモリシステムによって実行される複数段階の書き込み動作を説明するための図。
【図４６】同実施形態のメモリシステム内のある書き込み先ブロックへのデータの書き込み順序を説明するための図。
【図４７】不揮発性メモリのデータ書き込み単位と同じサイズの単位でライトデータをホストから同実施形態のメモリシステムに転送する動作を説明するための図。
【図４８】同実施形態のメモリシステムによって実行されるデータ書き込み処理の手順を示すフローチャート。
【図４９】同実施形態のメモリシステムによって実行されるデータ書き込み処理の別の手順を示すフローチャート。
【図５０】同実施形態のメモリシステムによって実行されるホストへの解放可能通知の送信処理の手順を示すフローチャート。
【図５１】ホストによって実行されるライトデータ破棄処理の手順を示すフローチャート。
【図５２】最後のライトコマンドが受信されてから閾期間、次のライトコマンドが受信されない場合に、同実施形態のメモリシステムによって実行されるダミーデータ書き込み処

理を説明するための図。
【図５３】同実施形態のメモリシステムによって実行されるダミーデータ書き込み処理の手順を示すフローチャート。
【図５４】内部バッファを使用して同実施形態のメモリシステムによって実行されるデータ転送動作を示すブロック図。
【図５５】内部バッファを使用して同実施形態のメモリシステムによって実行されるデータ書き込み処理の手順を示すフローチャート。
【図５６】同実施形態のメモリシステムによって実行されるデータ読み出し処理の手順を示すフローチャート。
【図５７】同実施形態のメモリシステムに適用されるブロックリユースコマンドコマンドを説明するための図。
【図５８】同実施形態のメモリシステムに適用されるライトコマンドの別の例を説明するための図。
【図５９】図５８のライトコマンドに対するレスポンスを説明するための図。
【図６０】ホストと同実施形態のメモリシステムとによって実行される書き込み動作処理のシーケンスの別の例を示すシーケンスチャート。
【０００９】
【図６１】同実施形態のメモリシステムに適用されるガベージコレクション（ＧＣ）制御コマンドの別の例を説明するための図。
【図６２】同実施形態のメモリシステムによって実行されるガベージコレクション（ＧＣ）動作の手順の別の例を示すシーケンスチャート。
【図６３】同実施形態のメモリシステムにおける書き込み先ブロックの割り当ての手順を示すフローチャート。
【図６４】同実施形態のメモリシステムにおけるＧＣデスティネーションブロックの割り当ての手順を示すフローチャート。
【発明を実施するための形態】
【００１０】
以下、図面を参照して、実施形態を説明する。
まず、図１を参照して、一実施形態に係るメモリシステムを含む計算機システムの構成を説明する。
このメモリシステムは、不揮発性メモリにデータを書き込み、不揮発性メモリからデータを読み出すように構成された半導体ストレージデバイスである。このメモリシステムは、ＮＡＮＤフラッシュ技術ベースのフラッシュストレージデバイス３として実現されている。
【００１１】
この計算機システムは、ホスト（ホストデバイス）２と、複数のフラッシュストレージデバイス３とを含んでいてもよい。ホスト２は、複数のフラッシュストレージデバイス３によって構成されるフラッシュアレイをストレージとして使用するように構成されたサーバであってもよい。ホスト（サーバ）２と複数のフラッシュストレージデバイス３は、インタフェース５０を介して相互接続される（内部相互接続）。この内部相互接続のためのインタフェース５０としては、これに限定されないが、ＰＣＩ　Ｅｘｐｒｅｓｓ（ＰＣＩｅ）（登録商標）、ＮＶＭ　Ｅｘｐｒｅｓｓ（ＮＶＭｅ）（登録商標）、Ｅｔｈｅｒｎｅｔ（登録商標）、ＮＶＭｅ　ｏｖｅｒ　Ｆａｂｒｉｃｓ（ＮＶＭｅＯＦ）等を使用し得る。
【００１２】
ホスト２として機能するサーバの典型例としては、データセンター内のサーバが挙げられる。
ホスト２がデータセンター内のサーバによって実現されるケースにおいては、このホスト（サーバ）２は、ネットワーク５１を介して複数のエンドユーザ端末（クライアント）６１に接続されてもよい。ホスト２は、これらエンドユーザ端末６１に対して様々なサービスを提供することができる。
【００１３】

ホスト（サーバ）２によって提供可能なサービスの例には、（１）システム稼働プラットフォームを各クライアント（各エンドユーザ端末６１）に提供するプラットホーム・アズ・ア・サービス（ＰａａＳ）、（２）仮想サーバのようなインフラストラクチャを各クライアント（各エンドユーザ端末６１）に提供するインフラストラクチャ・アズ・ア・サービス（ＩａａＳ）、等がある。

【００１４】

複数の仮想マシンが、このホスト（サーバ）２として機能する物理サーバ上で実行されてもよい。ホスト（サーバ）２上で走るこれら仮想マシンの各々は、対応する幾つかのクライアント（エンドユーザ端末６１）に各種サービスを提供するように構成された仮想サーバとして機能することができる。

【００１５】

ホスト（サーバ）２は、フラッシュアレイを構成する複数のフラッシュストレージデバイス３を管理するストレージ管理機能と、エンドユーザ端末６１それぞれに対してストレージアクセスを含む様々なサービスを提供するフロントエンド機能とを含む。

従来型ＳＳＤにおいては、ＮＡＮＤ型フラッシュメモリのブロック／ページの階層構造はＳＳＤ内のフラッシュトランスレーション層（ＦＴＬ）によって隠蔽されている。つまり、従来型ＳＳＤのＦＴＬは、（１）論理物理アドレス変換テーブルとして機能するルックアップテーブルを使用して、論理アドレスそれぞれとＮＡＮＤ型フラッシュメモリの物理アドレスそれぞれとの間のマッピングを管理する機能、（２）ページ単位のリード／ライトとブロック単位の消去動作とを隠蔽するための機能、（３）ＮＡＮＤ型フラッシュメモリのガベージコレクション（ＧＣ）を実行する機能、等を有している。論理アドレスそれぞれとＮＡＮＤ型フラッシュメモリの物理アドレスとの間のマッピングは、ホストからは見えない。ＮＡＮＤ型フラッシュメモリのブロック／ページ構造もホストからは見えない。

【００１６】

一方、ホストにおいても、一種のアドレス変換（アプリケーションレベルアドレス変換）が実行されることがある。このアドレス変換は、アプリケーションレベルアドレス変換テーブルを使用して、アプリケーションレベルの論理アドレスそれぞれとＳＳＤ用の論理アドレスそれぞれとの間のマッピングを管理する。また、ホストにおいても、ＳＳＤ用の論理アドレス空間上に生じるフラグメントの解消のために、この論理アドレス空間上のデータ配置を変更する一種のＧＣ（アプリケーションレベルＧＣ）が実行される。

【００１７】

しかし、ホストおよびＳＳＤがそれぞれアドレス変換テーブルを有するという冗長な構成（ＳＳＤは論理物理アドレス変換テーブルとして機能するルックアップテーブルを有し、ホストはアプリケーションレベルアドレス変換テーブルを有する）においては、これらアドレス変換テーブルを保持するために膨大なメモリリソースが消費される。さらに、ホスト側のアドレス変換とＳＳＤ側のアドレス変換とを含む２重のアドレス変換は、Ｉ／Ｏ性能を低下させる要因にもなる。

【００１８】

さらに、ホスト側のアプリケーションレベルＧＣは、ＳＳＤへのデータ書き込み量を実際のユーザデータ量の数倍（例えば２倍）程度に増やす要因となる。このようなデータ書き込み量の増加は、ＳＳＤのライトアンプリフィケーションとあいまってシステム全体のストレージ性能を低下させ、またＳＳＤの寿命も短くする。

【００１９】

このような問題点を解消するために、従来型ＳＳＤのＦＴＬの機能の全てをホストに移すという対策も考えられる。

しかし、この対策を実装するためには、ＮＡＮＤ型フラッシュメモリのブロックおよびページをホストが直接的にハンドリングすることが必要となる。ＮＡＮＤ型フラッシュメモリにおいては、ページ書き込み順序制約があるため、ホストがページを直接ハンドリングすることは困難である。また、ＮＡＮＤ型フラッシュメモリにおいては、ブロックが不良ページ（バッドページ）を含む場合がある。バッドページをハンドリングすることはホス

トにとってはなおさら困難である。
【００２０】
そこで、本実施形態では、ＦＴＬの役割はホスト２とフラッシュストレージデバイス３との間で分担される。概していえば、ホスト２は論理物理アドレス変換テーブルとして機能するルックアップテーブルを管理するが、ホスト２はデータが書き込まれるべきブロックのブロック番号とこのデータに対応する論理アドレスだけを指定し、このデータが書き込まれるべきこのブロック内の位置（書き込み先位置）はフラッシュストレージデバイス３によって決定される。決定されたこのブロック内の位置（書き込み先位置）を示すブロック内物理アドレスは、フラッシュストレージデバイス３からホスト２に通知される。
【００２１】
このように、ホスト２はブロックのみをハンドリングし、ブロック内の位置（例えば、ページ、ページ内の位置）はフラッシュストレージデバイス３によってハンドリングされる。フラッシュストレージデバイス３にデータを書き込む必要がある時、ホスト２は、ブロック番号を選択（またはフラッシュストレージデバイス３にフリーブロックを割り当てるように要求）し、論理アドレスと、選択したブロックのブロック番号（またはフラッシュストレージデバイス３によって通知される割り当てられたブロックのブロック番号）とを指定するライト要求（ライトコマンド）をフラッシュストレージデバイス３に送信する。フラッシュストレージデバイス３は、指定されたブロック番号を有するブロックにホスト２からのデータを書き込む。この場合、フラッシュストレージデバイス３は、このブロック内の位置（書き込み先位置）を決定し、ホスト２からのデータをこのブロック内の位置（書き込み先位置）に書き込む。そして、フラッシュストレージデバイス３は、このブロック内の位置（書き込み先位置）を示すブロック内物理アドレスを、ライト要求に対するレスポンス（返り値）としてホスト２に通知する。以下では、ホスト２に移されたＦＴＬ機能をグローバルＦＴＬと称する。
【００２２】
ホスト２のグローバルＦＴＬは、ストレージサービスを実行する機能、ウェアー制御機能、高可用性を実現するための機能、同じ内容を有する複数の重複データ部がストレージに格納されることを防止する重複排除（Ｄｅ－ｄｕｐｌｉｃａｔｉｏｎ）機能、ガベージコレクション（ＧＣ）ブロック選択機能、ＱｏＳ制御機能等を有する。ＱｏＳ制御機能には、ＱｏＳドメイン毎（またはブロック毎）にアクセス単位を決める機能が含まれる。アクセス単位は、ホスト２がライト／リードすることが可能な最小データサイズ（Ｇｒａｉｎ）を示す。フラッシュストレージデバイス３は単一、あるいは複数のアクセス単位（Ｇｒａｉｎ）をサポートしており、ホスト２は、フラッシュストレージデバイス３が複数のアクセス単位をサポートしている場合にはＱｏＳドメイン毎（またはブロック毎）に、使用すべきアクセス単位をフラッシュストレージデバイス３に指示することができる。
【００２３】
また、ＱｏＳ制御機能には、ＱｏＳドメイン間の性能干渉をできるだけ防ぐための機能が含まれている。この機能は、基本的には、安定したレイテンシを保つための機能である。一方、フラッシュストレージデバイス３は、ローレベルアブストラクション（ＬＬＡ）を実行することができる。ＬＬＡはＮＡＮＤ型フラッシュメモリのアブストラクションのための機能である。ＬＬＡは、不良ページ（バッドページ）を隠蔽する機能、ページ書き込み順序制約を守る機能を含む。ＬＬＡは、ＧＣ実行機能も含む。ＧＣ実行機能は、ホスト２によって指定されたコピー元ブロック（ＧＣソースブロック）内の有効データを、ホスト２によって指定されたコピー先ブロック（ＧＣデスティネーションブロック）にコピーする。フラッシュストレージデバイス３のＧＣ実行機能は、有効データを書き込むべきＧＣデスティネーションブロック内の位置（コピー先位置）を決定し、ＧＣソースブロック内の有効データを、ＧＣデスティネーションブロック内のコピー先位置にコピーする。
【００２４】
図２は、従来型ＳＳＤとホストとの間の役割分担と、本実施形態のフラッシュストレージデバイス３とホスト２との間の役割分担とを示す。

図２の左部は、従来型ＳＳＤと仮想ディスクサービスを実行するホストとを含む計算機システム全体の階層構造を表している。

【００２５】

ホスト（サーバ）においては、複数のエンドユーザに複数の仮想マシンを提供するための仮想マシンサービス１０１が実行される。仮想マシンサービス１０１上の各仮想マシンにおいては、対応するエンドユーザによって使用されるオペレーティングシステムおよびユーザアプリケーション１０２が実行される。

【００２６】

また、ホスト（サーバ）においては、複数のユーザアプリケーション１０２に対応する複数の仮想ディスクサービス１０３が実行される。各仮想ディスクサービス１０３は、従来型ＳＳＤ内のストレージリソースの容量の一部を、対応するユーザアプリケーション１０２用のストレージリソース（仮想ディスク）として割り当てる。各仮想ディスクサービス１０３においては、アプリケーションレベルアドレス変換テーブルを使用して、アプリケーションレベルの論理アドレスをＳＳＤ用の論理アドレスに変換するアプリケーションレベルアドレス変換も実行される。さらに、ホストにおいては、アプリケーションレベルＧＣ１０４も実行される。

【００２７】

ホスト（サーバ）から従来型ＳＳＤへのコマンドの送信および従来型ＳＳＤからホスト（サーバ）へのコマンド完了のレスポンスの返送は、ホスト（サーバ）および従来型ＳＳＤの各々に存在するＩ／Ｏキュー２００を介して実行される。

従来型ＳＳＤは、ライトバッファ（ＷＢ）３０１、ルックアップテーブル（ＬＵＴ）３０２、ガベージコレクション機能３０３、ＮＡＮＤ型フラッシュメモリ（ＮＡＮＤフラッシュアレイ）３０４を含む。従来型ＳＳＤは、一つのルックアップテーブル（ＬＵＴ）３０２のみを管理しており、ＮＡＮＤ型フラッシュメモリ（ＮＡＮＤフラッシュアレイ）３０４のリソースは複数の仮想ディスクサービス１０３によって共有される。

【００２８】

この構成においては、仮想ディスクサービス１０３下のアプリケーションレベルＧＣ１０４と従来型ＳＳＤ内のガベージコレクション機能３０３（ＬＵＴレベルＧＣ）とを含む重複したＧＣにより、ライトアンプリフィケーションが大きくなる。また、従来型ＳＳＤにおいては、あるエンドユーザまたはある仮想ディスクサービス１０３からのデータ書き込み量の増加によってＧＣの頻度が増加し、これによって他のエンドユーザまたは他の仮想ディスクサービス１０３に対するＩ／Ｏ性能が劣化するというノイジーネイバー問題が生じうる。

【００２９】

また、各仮想ディスクサービス内のアプリケーションレベルアドレス変換テーブルと従来型ＳＳＤ内のＬＵＴ３０２とを含む重複したリソースの存在により、多くのメモリリソースが消費される。

図２の右部は、本実施形態のフラッシュストレージデバイス３とホスト２とを含む計算機システム全体の階層構造を表している。

【００３０】

ホスト（サーバ）２においては、複数のエンドユーザに複数の仮想マシンを提供するための仮想マシンサービス４０１が実行される。仮想マシンサービス４０１上の各仮想マシンにおいては、対応するエンドユーザによって使用されるオペレーティングシステムおよびユーザアプリケーション４０２が実行される。

【００３１】

また、ホスト（サーバ）２においては、複数のユーザアプリケーション４０２に対応する複数のＩ／Ｏサービス４０３が実行される。これらＩ／Ｏサービス４０３には、ＬＢＡベースのブロックＩ／Ｏサービス、キー・バリュー・ストアサービスなどが含まれてもよい。各Ｉ／Ｏサービス４０３は、論理アドレスそれぞれとフラッシュストレージデバイス３の物理アドレスそれぞれとの間のマッピングを管理するルックアップテーブル（ＬＵＴ）

４１１を含む。ここで、論理アドレスとは、アクセス対象のデータを識別可能な識別子を意味する。この論理アドレスは、論理アドレス空間上の位置を指定する論理ブロックアドレス（ＬＢＡ）であってもよいし、あるいは、キー・バリュー・ストアのキー（タグ）であってもよいし、キーのハッシュ値であってもよい。

【００３２】

ＬＢＡベースのブロックＩ／Ｏサービスにおいては、論理アドレス（ＬＢＡ）それぞれとフラッシュストレージデバイス３の物理アドレスそれぞれとの間のマッピングを管理するＬＵＴ４１１が使用されてもよい。

キー・バリュー・ストアサービスにおいては、論理アドレス（つまり、キーのようなタグ）それぞれとこれら論理アドレス（つまり、キーのようなタグ）に対応するデータが格納されているフラッシュストレージデバイス３内の物理記憶位置を示す物理アドレスそれぞれとの間のマッピングを管理するＬＵＴ４１１が使用されてもよい。このＬＵＴ４１１においては、タグと、このタグによって識別されるデータが格納されている物理アドレスと、このデータのデータ長との対応関係が管理されてもよい。

【００３３】

各エンドユーザは、使用すべきアドレッシング方法（ＬＢＡ、キー・バリュー・ストアのキー、等）を選択することができる。

これら各ＬＵＴ４１１は、ユーザアプリケーション４０２からの論理アドレスそれぞれをフラッシュストレージデバイス３用の論理アドレスそれぞれに変換するのではなく、ユーザアプリケーション４０２からの論理アドレスそれぞれをフラッシュストレージデバイス３の物理アドレスそれぞれに変換する。つまり、これら各ＬＵＴ４１１は、フラッシュストレージデバイス３用の論理アドレスを物理アドレスに変換するテーブルとアプリケーションレベルアドレス変換テーブルとが統合（マージ）されたテーブルである。

【００３４】

また、各Ｉ／Ｏサービス４０３は、ＧＣブロック選択機能を含む。ＧＣブロック選択機能は、対応するＬＵＴを使用して各ブロックの有効データ量を管理することができ、これによってＧＣソースブロックを選択することができる。

ホスト（サーバ）２においては、上述のＱｏＳドメイン毎にＩ／Ｏサービス４０３が存在してもよい。あるＱｏＳドメインに属するＩ／Ｏサービス４０３は、対応するＱｏＳドメイン内のユーザアプリケーション４０２によって使用される論理アドレスそれぞれと対応するＱｏＳドメインに割り当てられたリソースグループに属するブロック群のブロック番号それぞれとの間のマッピングを管理してもよい。

【００３５】

ホスト（サーバ）２からフラッシュストレージデバイス３へのコマンドの送信およびフラッシュストレージデバイス３からホスト（サーバ）２へのコマンド完了のレスポンス等の返送は、ホスト（サーバ）２およびフラッシュストレージデバイス３の各々に存在するＩ／Ｏキュー５００を介して実行される。これらＩ／Ｏキュー５００も、複数のＱｏＳドメインに対応する複数のキューグループに分類されていてもよい。

【００３６】

フラッシュストレージデバイス３は、複数のＱｏＳドメインに対応する複数のライトバッファ（ＷＢ）６０１、複数のＱｏＳドメインに対応する複数のガベージコレクション（ＧＣ）機能６０２、ＮＡＮＤ型フラッシュメモリ（ＮＡＮＤフラッシュアレイ）６０３を含む。

【００３７】

この図２の右部に示す構成においては、上位階層（ホスト２）はブロック境界を認識することができるので、ブロック境界／ブロックサイズを考慮してユーザデータを各ブロックに書き込むことができる。つまり、ホスト２はＮＡＮＤ型フラッシュメモリ（ＮＡＮＤフラッシュアレイ）６０３の個々のブロックを認識することができ、これにより、例えば、一つのブロック全体に一斉にデータを書き込む、一つのブロック内のデータ全体を削除または更新によって無効化する、といった制御を行うことが可能となる。この結果、一つの

ブロックに有効データと無効データが混在されるという状況を起こりにくくすることが可能となる。したがって、ＧＣを実行することが必要となる頻度を低減することができる。ＧＣの頻度を低減することにより、ライトアンプリフィケーションが低下され、フラッシュストレージデバイス３の性能の向上、フラッシュストレージデバイス３の寿命の最大化を実現できる。このように、上位階層（ホスト２）がブロック番号を認識可能な構成は有用である。

【００３８】

一方、データが書き込まれるべきブロック内の位置は、上位階層（ホスト２）ではなく、フラッシュストレージデバイス３によって決定される。したがって、不良ページ（バッドページ）を隠蔽することができ、またページ書き込み順序制約を守ることができる。

【００３９】

図３は、図１のシステム構成の変形例を示す。

図３においては、複数のホスト２Ａと複数のフラッシュストレージデバイス３との間のデータ転送がネットワーク機器（ここでは、ネットワークスイッチ１）を介して実行される。すなわち、図３の計算機システムにおいては、図１のホスト（サーバ）２のストレージ管理機能がマネージャ２Ｂに移され、且つホスト（サーバ）２のフロントエンド機能が複数のホスト（エンドユーザサービス用ホスト）２Ａに移されている。

【００４０】

マネージャ２Ｂは、複数のフラッシュストレージデバイス３を管理し、各ホスト（エンドユーザサービス用ホスト）２Ａからの要求に応じて、これらフラッシュストレージデバイス３のストレージリソースを各ホスト（エンドユーザサービス用ホスト）２Ａに割り当てる。

【００４１】

各ホスト（エンドユーザサービス用ホスト）２Ａは、ネットワークを介して一つ以上のエンドユーザ端末６１に接続される。各ホスト（エンドユーザサービス用ホスト）２Ａは、上述の統合（マージ）された論理物理アドレス変換テーブルであるルックアップテーブル（ＬＵＴ）を管理する。各ホスト（エンドユーザサービス用ホスト）２Ａは、自身のＬＵＴを使用して、対応するエンドユーザによって使用される論理アドレスそれぞれと自身に割り当てられたリソースの物理アドレスそれぞれとの間のマッピングのみを管理する。したがって、この構成は、システムを容易にスケールアウトすることを可能にする。

【００４２】

各ホスト２ＡのグローバルＦＴＬは、ルックアップテーブル（ＬＵＴ）を管理する機能、高可用性を実現するための機能、ＱｏＳ制御機能、ＧＣブロック選択機能等を有する。

マネージャ２Ｂは、複数のフラッシュストレージデバイス３を管理するための専用のデバイス（計算機）である。マネージャ２Ｂは、各ホスト２Ａから要求された容量分のストレージリソースを予約するグローバルリソース予約機能を有する。さらに、マネージャ２Ｂは、各フラッシュストレージデバイス３の消耗度を監視するためのウェアー監視機能、予約されたストレージリソース（ＮＡＮＤリソース）を各ホスト２Ａに割り当てるＮＡＮＤリソース割り当て機能、ＱｏＳ制御機能、グローバルクロック管理機能、等を有する。

【００４３】

各フラッシュストレージデバイス３のローレベルアブストラクション（ＬＬＡ）は、不良ページ（バッドページ）を隠蔽する機能、ページ書き込み順序制約を守る機能、ライトバッファを管理する機能、ＧＣ実行機能等を有する。

図３のシステム構成によれば、各フラッシュストレージデバイス３の管理はマネージャ２Ｂによって実行されるので、各ホスト２Ａは、自身に割り当てられた一つ以上のフラッシュストレージデバイス３にＩ／Ｏ要求を送信する動作と、フラッシュストレージデバイス３からのレスポンスを受信するという動作とのみを実行すればよい。つまり、複数のホスト２Ａと複数のフラッシュストレージデバイス３との間のデータ転送はネットワークスイッチ１のみを介して実行され、マネージャ２Ｂはこのデータ転送には関与しない。また、上述したように、ホスト２Ａそれぞれによって管理されるＬＵＴの内容は互いに独立して

いる。よって、容易にホスト２Ａの数を増やすことができるので、スケールアウト型のシステム構成を実現することができる。

【００４４】

図４は、フラッシュストレージデバイス３の構成例を示す。

フラッシュストレージデバイス３は、コントローラ４および不揮発性メモリ（ＮＡＮＤ型フラッシュメモリ）５を備える。フラッシュストレージデバイス３は、ランダムアクセスメモリ、例えば、ＤＲＡＭ６も備えていてもよい。

【００４５】

ＮＡＮＤ型フラッシュメモリ５は、マトリクス状に配置された複数のメモリセルを含むメモリセルアレイを含む。ＮＡＮＤ型フラッシュメモリ５は、２次元構造のＮＡＮＤ型フラッシュメモリであってもよいし、３次元構造のＮＡＮＤ型フラッシュメモリであってもよい。

【００４６】

ＮＡＮＤ型フラッシュメモリ５のメモリセルアレイは、複数のブロックＢＬＫ０～ＢＬＫｍ－１を含む。ブロックＢＬＫ０～ＢＬＫｍ－１の各々は多数のページ（ここではページＰ０～Ｐｎ－１）によって編成される。ブロックＢＬＫ０～ＢＬＫｍ－１は、消去単位として機能する。ブロックは、「消去ブロック」、「物理ブロック」、または「物理消去ブロック」と称されることもある。ページＰ０～Ｐｎ－１の各々は、同一ワード線に接続された複数のメモリセルを含む。ページＰ０～Ｐｎ－１は、データ書き込み動作およびデータ読み込み動作の単位である。

コントローラ４は、Ｔｏｇｇｌｅ、オープンＮＡＮＤフラッシュインタフェース（ＯＮＦＩ）のようなＮＡＮＤインタフェース１３を介して、不揮発性メモリであるＮＡＮＤ型フラッシュメモリ５に電気的に接続されている。コントローラ４は、ＮＡＮＤ型フラッシュメモリ５を制御するように構成されたメモリコントローラ（制御回路）である。

【００４７】

ＮＡＮＤ型フラッシュメモリ５は、図５に示すように、複数のＮＡＮＤ型フラッシュメモリダイを含む。各ＮＡＮＤ型フラッシュメモリダイは、複数のブロックＢＬＫを含むメモリセルアレイとこのメモリセルアレイを制御する周辺回路とを含む不揮発性メモリダイである。個々のＮＡＮＤ型フラッシュメモリダイは独立して動作可能である。このため、ＮＡＮＤ型フラッシュメモリダイは、並列動作単位として機能する。ＮＡＮＤ型フラッシュメモリダイは、「ＮＡＮＤ型フラッシュメモリチップ」または「不揮発性メモリチップ」とも称される。図５においては、ＮＡＮＤインタフェース１３に１６個のチャンネルＣｈ１、Ｃｈ２、…Ｃｈ１６が接続されており、これらチャンネルＣｈ１、Ｃｈ２、…Ｃｈ１６の各々に、同数（例えばチャンネル当たり２個のダイ）のＮＡＮＤ型フラッシュメモリダイそれぞれが接続されている場合が例示されている。各チャンネルは、対応するＮＡＮＤ型フラッシュメモリダイと通信するための通信線（メモリバス）を含む。

【００４８】

コントローラ４は、チャンネルＣｈ１、Ｃｈ２、…Ｃｈ１６を介してＮＡＮＤ型フラッシュメモリダイ＃１～＃３２を制御する。コントローラ４は、チャンネルＣｈ１、Ｃｈ２、…Ｃｈ１６を同時に駆動することができる。

チャンネルＣｈ１～Ｃｈ１６に接続された１６個のＮＡＮＤ型フラッシュメモリダイ＃１～＃１６は第１のバンクとして編成されてもよく、またチャンネルＣｈ１～Ｃｈ１６に接続された残りの１６個のＮＡＮＤ型フラッシュメモリダイ＃１７～＃３２は第２のバンクとして編成されてもよい。バンクは、複数のメモリモジュールをバンクインタリーブによって並列動作させるための単位として機能する。図５の構成例においては、１６チャンネルと、２つのバンクを使用したバンクインタリーブとによって、最大３２個のＮＡＮＤ型フラッシュメモリダイを並列動作させることができる。

【００４９】

本実施形態では、コントローラ４は、各々が複数のブロックＢＬＫから構成される複数のブロック（以下、スーパーブロックと称する）を管理してもよく、スーパーブロックの単

位で消去動作を実行してもよい。
スーパーブロックは、これに限定されないが、ＮＡＮＤ型フラッシュメモリダイ＃１～＃３２から一つずつ選択される計３２個のブロックＢＬＫを含んでいてもよい。なお、ＮＡＮＤ型フラッシュメモリダイ＃１～＃３２の各々はマルチプレーン構成を有していてもよい。例えば、ＮＡＮＤ型フラッシュメモリダイ＃１～＃３２の各々が、２つのプレーンを含むマルチプレーン構成を有する場合には、一つのスーパーブロックは、ＮＡＮＤ型フラッシュメモリダイ＃１～＃３２に対応する６４個のプレーンから一つずつ選択される計６４個のブロックＢＬＫを含んでいてもよい。図６には、一つのスーパーブロックＳＢが、ＮＡＮＤ型フラッシュメモリダイ＃１～＃３２から一つずつ選択される計３２個のブロックＢＬＫ（図５においては太枠で囲まれているブロックＢＬＫ）から構成される場合が例示されている。

【００５０】
図４に示されているように、コントローラ４は、ホストインタフェース１１、ＣＰＵ１２、ＮＡＮＤインタフェース１３、およびＤＲＡＭインタフェース１４等を含む。これらＣＰＵ１２、ＮＡＮＤインタフェース１３、ＤＲＡＭインタフェース１４は、バス１０を介して相互接続される。

【００５１】
このホストインタフェース１１は、ホスト２との通信を実行するように構成されたホストインタフェース回路である。このホストインタフェース１１は、例えば、ＰＣＩｅコントローラ（ＮＶＭｅコントローラ）であってもよい。ホストインタフェース１１は、ホスト２から様々な要求（コマンド）を受信する。これら要求（コマンド）には、ライト要求（ライトコマンド）、リード要求（リードコマンド）、他の様々な要求（コマンド）が含まれる。

【００５２】
ＣＰＵ１２は、ホストインタフェース１１、ＮＡＮＤインタフェース１３、ＤＲＡＭインタフェース１４を制御するように構成されたプロセッサである。ＣＰＵ１２は、フラッシュストレージデバイス３の電源オンに応答してＮＡＮＤ型フラッシュメモリ５または図示しないＲＯＭから制御プログラム（ファームウェア）をＤＲＡＭ６にロードし、そしてこのファームウェアを実行することによって様々な処理を行う。なお、ファームウェアはコントローラ４内の図示しないＳＲＡＭ上にロードされてもよい。このＣＰＵ１２は、ホスト２からの様々なコマンドを処理するためのコマンド処理等を実行することができる。ＣＰＵ１２の動作は、ＣＰＵ１２によって実行される上述のファームウェアによって制御される。なお、コマンド処理の一部または全部は、コントローラ４内の専用ハードウェアによって実行してもよい。

【００５３】
ＣＰＵ１２は、ライト動作制御部２１、リード動作制御部２２、およびＧＣ動作制御部２３として機能することができる。これらライト動作制御部２１、リード動作制御部２２、およびＧＣ動作制御部２３においては、図２の右部に示すシステム構成を実現するためのアプリケーションプログラムインタフェース（ＡＰＩ）が実装されている。

【００５４】
ライト動作制御部２１は、ブロック番号と論理アドレスを指定するライト要求（ライトコマンド）をホスト２から受信する。論理アドレスは、書き込むべきデータ（ユーザデータ）を識別可能な識別子であり、例えば、ＬＢＡであってもよいし、あるいはキー・バリュー・ストアのキーのようなタグであってもよいし、キーのハッシュ値であってもよい。ブロック番号は、このデータが書き込まれるべきブロックを指定する識別子である。ブロック番号としては、複数のブロック内の任意の一つを一意に識別可能な様々な値を使用し得る。ブロック番号によって指定されるブロックは、物理ブロックであってもよいし、上述のスーパーブロックであってもよい。ライトコマンドを受信した場合、ライト動作制御部２１は、まず、ホスト２からのデータを書き込むべき、この指定されたブロック番号を有するブロック（書き込み先ブロック）内の位置（書き込み先位置）を決定する。次いで、

ライト動作制御部２１は、ホスト２からのデータ（ライトデータ）を、この書き込み先ブロックの書き込み先位置に書き込む。この場合、ライト動作制御部２１は、ホスト２からのデータのみならず、このデータとこのデータの論理アドレスの双方を書き込み先ブロックに書き込むことができる。

【００５５】

そして、ライト動作制御部２１は、この書き込み先ブロックの上述の書き込み先位置を示すブロック内物理アドレスをホスト２に通知する。このブロック内物理アドレスは、この書き込み先ブロック内の書き込み先位置を示すブロック内オフセットによって表される。

【００５６】

この場合、このブロック内オフセットは、書き込み先ブロックの先頭から書き込み先位置までのオフセット、つまり書き込み先ブロックの先頭に対する書き込み先位置のオフセットを示す。書き込み先ブロックの先頭から書き込み先位置までのオフセットのサイズは、ページサイズとは異なるサイズを有する粒度（Ｇｒａｉｎ）の倍数で示される。粒度（Ｇｒａｉｎ）は、上述のアクセス単位である。粒度（Ｇｒａｉｎ）のサイズの最大値は、ブロックサイズまでに制限される。換言すれば、ブロック内オフセットは、書き込み先ブロックの先頭から書き込み先位置までのオフセットをページサイズとは異なるサイズを有する粒度の倍数で示す。

【００５７】

粒度（Ｇｒａｉｎ）は、ページサイズよりも小さいサイズを有していてもよい。例えば、ページサイズが１６Ｋバイトである場合、粒度（Ｇｒａｉｎ）は、そのサイズが４Ｋバイトであってもよい。この場合、ある一つのブロックにおいては、各々サイズが４Ｋバイトである複数のオフセット位置が規定される。ブロック内の最初のオフセット位置に対応するブロック内オフセットは、例えば０であり、ブロック内の次のオフセット位置に対応するブロック内オフセットは、例えば１である、ブロック内のさらに次のオフセット位置に対応するブロック内オフセットは、例えば２である。

【００５８】

あるいは、粒度（Ｇｒａｉｎ）は、ページサイズよりも大きなサイズを有していてもよい。例えば、粒度（Ｇｒａｉｎ）は、ページサイズの数倍のサイズであってもよい。ページサイズが１６Ｋバイトである場合、粒度は、３２Ｋバイトのサイズであってもよい。

【００５９】

このように、ライト動作制御部２１は、ホスト２からのブロック番号を有するブロック内の書き込み先位置を自身で決定し、そしてホスト２からのライトデータをこのブロック内のこの書き込み先位置に書き込む。そして、ライト動作制御部２１は、この書き込み先位置を示すブロック内物理アドレス（ブロック内オフセット）をライト要求に対応するレスポンス（返り値）としてホスト２に通知する。あるいは、ライト動作制御部２１は、ブロック内物理アドレス（ブロック内オフセット）のみをホスト２に通知するのではなく、論理アドレスとブロック番号とブロック内物理アドレス（ブロック内オフセット）との組をホスト２に通知してもよい。

【００６０】

したがって、フラッシュストレージデバイス３は、ブロック番号をホスト２にハンドリングさせつつ、ページ書き込み順序制約、バッドページ、ページサイズ等を隠蔽することができる。

この結果、ホスト２は、ブロック境界は認識できるが、ページ書き込み順序制約、バッドページ、ページサイズについては意識することなく、どのユーザデータがどのブロック番号に存在するかを管理することができる。

【００６１】

リード動作制御部２２は、物理アドレス（すなわち、ブロック番号およびブロック内オフセット）を指定するリード要求（リードコマンド）をホスト２から受信した場合、これらブロック番号およびブロック内オフセットに基づいて、リード対象のブロック内のリード対象の物理記憶位置からデータをリードする。リード対象のブロックは、ブロック番号に

よって特定される。このブロック内のリード対象の物理記憶位置は、ブロック内オフセットによって特定される。このブロック内オフセットを使用することにより、ホスト２は、ＮＡＮＤ型フラッシュメモリの世代毎の異なるページサイズをハンドリングする必要がない。

【００６２】

リード対象の物理記憶位置を得るために、リード動作制御部２２は、まず、このブロック内オフセットを、ページサイズを表す粒度の数（ページサイズが１６Ｋバイトで粒度（Ｇｒａｉｎ）が４Ｋバイトである場合には、ページサイズを表す粒度の数は４）で除算し、そしてこの除算によって得られる商および余りを、リード対象のページ番号およびリード対象のページ内オフセットとしてそれぞれ決定してもよい。

【００６３】

ＧＣ動作制御部２３は、ＮＡＮＤ型フラッシュメモリ５のガベージコレクションのためのコピー元ブロック番号（ＧＣソースブロック番号）およびコピー先ブロック番号（ＧＣデスティネーションブロック番号）を指定するＧＣ制御コマンドをホスト２から受信した場合、ＮＡＮＤ型フラッシュメモリ５の複数のブロックから、指定されたコピー元ブロック番号を有するブロックと指定されたコピー先ブロック番号を有するブロックとをコピー元ブロック（ＧＣソースブロック）およびコピー先ブロック（ＧＣデスティネーションブロック）として選択する。ＧＣ動作制御部２３は、選択されたＧＣソースブロックに格納されている有効データを書き込むべきＧＣデスティネーションブロック内のコピー先位置を決定し、有効データをＧＣデスティネーションブロック内のコピー先位置にコピーする。

【００６４】

そして、ＧＣ動作制御部２３は、有効データの論理アドレスと、コピー先ブロック番号と、ＧＣデスティネーションブロック内のコピー先位置を示すブロック内物理アドレス（ブロック内オフセット）とを、ホスト２に通知する。

有効データ／無効データの管理は、ブロック管理テーブル３２を使用して実行されてもよい。このブロック管理テーブル３２は、例えば、ブロック毎に存在してもよい。あるブロックに対応するブロック管理テーブル３２においては、このブロック内のデータそれぞれの有効／無効を示すビットマップフラグが格納されている。ここで、有効データとは、論理アドレスから最新のデータとして紐付けられているデータであって、後にホスト２からリードされる可能性があるデータを意味する。無効データとは、もはやホスト２からリードされる可能性が無いデータを意味する。例えば、ある論理アドレスに関連付けられているデータは有効データであり、どの論理アドレスにも関連付けられていないデータは無効データである。

【００６５】

上述したように、ＧＣ動作制御部２３は、コピー元ブロック（ＧＣソースブロック）内に格納されている有効データを書き込むべきコピー先ブロック（ＧＣデスティネーションブロック）内の位置（コピー先位置）を決定し、有効データをコピー先ブロック（ＧＣデスティネーションブロック）のこの決定された位置（コピー先位置）にコピーする。この場合、ＧＣ動作制御部２３は、有効データとこの有効データの論理アドレスの双方を、コピー先ブロック（ＧＣデスティネーションブロック）にコピーしてもよい。

【００６６】

本実施形態では、上述したように、ライト動作制御部２１は、ホスト２からのデータ（ライトデータ）とホスト２からの論理アドレスの双方を書き込み先ブロックに書き込むことができる。このため、ＧＣ動作制御部２３は、コピー元ブロック（ＧＣソースブロック）内の各データの論理アドレスをこのコピー元ブロック（ＧＣソースブロック）から容易に取得することができるので、コピーされた有効データの論理アドレスをホスト２に容易に通知することができる。

【００６７】

ＮＡＮＤインタフェース１３は、ＣＰＵ１２の制御の下、ＮＡＮＤ型フラッシュメモリ５を制御するように構成されたメモリ制御回路である。ＤＲＡＭインタフェース１４は、Ｃ

ＰＵ１２の制御の下、ＤＲＡＭ６を制御するように構成されたＤＲＡＭ制御回路である。ＤＲＡＭ６の記憶領域の一部は、内部バッファ（共有キャッシュ）３１の格納のために使用される。また、ＤＲＡＭ６の記憶領域の他の一部は、ブロック管理テーブル３２の格納のために使用される。なお、これら内部バッファ（共有キャッシュ）３１、およびブロック管理テーブル３２は、コントローラ４内の図示しないＳＲＡＭに格納されてもよい。

【００６８】

図７は、ホスト２が論理アドレスとブロック番号とを指定し且つフラッシュストレージデバイス３がブロック内物理アドレス（ブロック内オフセット）を決定するデータ書き込み動作と、ホスト２がブロック番号とブロック内物理アドレス（ブロック内オフセット）とを指定するデータ読み出し動作とを示す。

【００６９】

データ書き込み動作は以下の手順で実行される。

（１）ホスト２のライト処理部４１２がフラッシュストレージデバイス３にデータ（ライトデータ）を書き込むことが必要な時、ライト処理部４１２は、フリーブロックを割り当てるようにフラッシュストレージデバイス３に要求してもよい。フラッシュストレージデバイス３のコントローラ４は、ＮＡＮＤ型フラッシュメモリ５のフリーブロック群を管理するブロック割り当て部７０１を含む。ブロック割り当て部７０１がライト処理部４１２からこの要求（ブロック割り当て要求）を受信した時、ブロック割り当て部７０１は、フリーブロック群の一つのフリーブロックをホスト２に割り当て、割り当てられたブロックのブロック番号（ＢＬＫ＃）をホスト２に通知する。

【００７０】

あるいは、ライト処理部４１２がフリーブロック群を管理する構成においては、ライト処理部４１２が自身で書き込み先ブロックを選択してもよい。

（２）ライト処理部４１２は、ライトデータに対応する論理アドレス（例えばＬＢＡ）と書き込み先ブロックのブロック番号（ＢＬＫ＃）とを指定するライト要求をフラッシュストレージデバイス３に送信する。

【００７１】

（３）フラッシュストレージデバイス３のコントローラ４は、データ書き込み用のページを割り当てるページ割り当て部７０２を含む。ページ割り当て部７０２がライト要求を受信した時、ページ割り当て部７０２は、ライト要求によって指定されたブロック番号を有するブロック（書き込み先ブロック）内の書き込み先位置を示すブロック内物理アドレス（ブロック内ＰＢＡ）を決定する。ブロック内物理アドレス（ブロック内ＰＢＡ）は、上述のブロック内オフセット（単にオフセットとしても参照される）によって表すことができる。コントローラ４は、ライト要求によって指定されたブロック番号と、ブロック内物理アドレス（ブロック内ＰＢＡ）とに基づいて、ホスト２からのライトデータを、書き込み先ブロック内の書き込み先位置に書き込む。

【００７２】

（４）コントローラ４は、書き込み先位置を示すブロック内物理アドレス（ブロック内ＰＢＡ）をライト要求に対するレスポンスとしてホスト２に通知する。あるいは、コントローラ４は、ライトデータに対応する論理アドレス（ＬＢＡ）と、書き込み先ブロックのブロック番号（ＢＬＫ＃）と、書き込み先位置を示すブロック内ＰＢＡ（オフセット）との組を、ライト要求に対するレスポンスとしてホスト２に通知してもよい。換言すれば、コントローラは、ブロック内物理アドレス、または論理アドレスとブロック番号とブロック内物理アドレスとの組のいずれかを、ホスト２に通知する。ホスト２においては、ライトデータが書き込まれた物理記憶位置を示す物理アドレス（ブロック番号、ブロック内物理アドレス（ブロック内オフセット））が、このライトデータの論理アドレスにマッピングされるように、ＬＵＴ４１１が更新される。

【００７３】

データリード動作は以下の手順で実行される。

（１）’ホスト２がフラッシュストレージデバイス３からデータをリードすることが必要な

時、ホスト２は、ＬＵＴ４１１を参照して、リードすべきデータの論理アドレスに対応する物理アドレス（ブロック番号、ブロック内物理アドレス（ブロック内オフセット））をＬＵＴ４１１から取得する。

【００７４】

（２）’ホスト２は、取得されたブロック番号およびブロック内物理アドレス（ブロック内オフセット）を指定するリード要求をフラッシュストレージデバイス３に送出する。フラッシュストレージデバイス３のコントローラ４がこのリード要求をホスト２から受信した時、コントローラ４は、ブロック番号およびブロック内物理アドレスに基づいて、リード対象のブロックおよびリード対象の物理記憶位置を特定し、このリード対象のブロック内のリード対象の物理記憶位置からデータをリードする。

【００７５】

図８は、フラッシュストレージデバイス３に適用されるライトコマンドを示す。

ライトコマンドは、フラッシュストレージデバイス３にデータの書き込みを要求するコマンドである。このライトコマンドは、コマンドＩＤ、ブロック番号ＢＬＫ＃、論理アドレス、長さ、等を含んでもよい。

【００７６】

コマンドＩＤはこのコマンドがライトコマンドであることを示すＩＤ（コマンドコード）であり、ライトコマンドにはライトコマンド用のコマンドＩＤが含まれる。

ブロック番号ＢＬＫ＃は、データが書き込まれるべきブロックを一意に識別可能な識別子（ブロックアドレス）である。

【００７７】

論理アドレスは、書き込まれるべきライトデータを識別するための識別子である。この論理アドレスは、上述したように、ＬＢＡであってもよいし、キー・バリュー・ストアのキーであってもよいし、キーのハッシュ値であってもよい。論理アドレスがＬＢＡである場合には、このライトコマンドに含まれる論理アドレス（開始ＬＢＡ）は、ライトデータが書き込まれるべき論理位置（最初の論理位置）を示す。

【００７８】

長さは、書き込まれるべきライトデータの長さを示す。この長さ（データ長）は、粒度（Ｇｒａｉｎ）の数によって指定されてもよいし、ＬＢＡの数によって指定されてもよいし、あるいはそのサイズがバイトによって指定されてもよい。

ホスト２からライトコマンドを受信した時、コントローラ４は、ライトコマンドによって指定されたブロック番号を有するブロック内の書き込み先位置を決定する。この書き込み先位置は、ページ書き込み順序の制約およびバッドページ等を考慮して決定される。そして、コントローラ４は、ホスト２からのデータを、ライトコマンドによって指定されたブロック番号を有するこのブロック内のこの書き込み先位置に書き込む。

【００７９】

図９は、図８のライトコマンドに対するレスポンスを示す。

このレスポンスは、ブロック内物理アドレス、長さを含む。ブロック内物理アドレスは、データが書き込まれたブロック内の位置（物理記憶位置）を示す。ブロック内物理アドレスは、上述したように、ブロック内オフセットによって指定可能である。長さは、書き込まれたデータの長さを示す。この長さ（データ長）は、粒度（Ｇｒａｉｎ）の数によって指定されてもよいし、ＬＢＡの数によって指定されてもよいし、あるいはそのサイズがバイトによって指定されてもよい。

【００８０】

あるいは、このレスポンスは、ブロック内物理アドレスおよび長さだけでなく、論理アドレスおよびブロック番号をさらに含んでいてもよい。論理アドレスは、図８のライトコマンドに含まれていた論理アドレスである。ブロック番号は、図８のライトコマンドに含まれていた論理アドレスである。

【００８１】

図１０は、フラッシュストレージデバイス３に適用されるＴｒｉｍコマンドを示す。

このＴｒｉｍコマンドは、無効にすべきデータが格納されている物理記憶位置を示すブロック番号およびブロック内物理アドレス（ブロック内オフセット）を含むコマンドである。つまり、このＴｒｉｍコマンドは、ＬＢＡのような論理アドレスではなく、物理アドレスを指定可能である。このＴｒｉｍコマンドは、コマンドＩＤ、物理アドレス、長さを含む。

【００８２】

コマンドＩＤはこのコマンドがＴｒｉｍコマンドであることを示すＩＤ（コマンドコード）であり、ＴｒｉｍコマンドにはＴｒｉｍコマンド用のコマンドＩＤが含まれる。

物理アドレスは、無効化すべきデータが格納されている最初の物理記憶位置を示す。本実施形態では、この物理アドレスは、ブロック番号とオフセット（ブロック内オフセット）との組み合わせによって指定される。

【００８３】

長さは、無効化すべきデータの長さを示す。この長さ（データ長）は、粒度（Ｇｒａｉｎ）の数によって指定されてもよいし、バイトによって指定されてもよい。

コントローラ４は、複数のブロックの各々に含まれるデータそれぞれの有効／無効を示すフラグ（ビットマップフラグ）をブロック管理テーブル３２を使用して管理する。無効にすべきデータが格納されている物理記憶位置を示すブロック番号およびオフセット（ブロック内オフセット）を含むＴｒｉｍコマンドをホスト２から受信した場合、コントローラ４は、ブロック管理テーブル３２を更新して、Ｔｒｉｍコマンドに含まれるブロック番号およびブロック内オフセットに対応する物理記憶位置のデータに対応するフラグ（ビットマップフラグ）を無効を示す値に変更する。

【００８４】

図１１は、ブロック内物理アドレスを規定するブロック内オフセットを示す。

ブロック番号はある一つのブロックＢＬＫを指定する。各ブロックＢＬＫは、図１１に示されているように、複数のページ（ここでは、ページ０～ページｎ）を含む。

ページサイズ（各ページのユーザデータ格納領域）が１６Ｋバイトであり、粒度（Ｇｒａｉｎ）が４ＫＢのサイズであるケースにおいては、このブロックＢＬＫは、４×（ｎ＋１）個の領域に論理的に分割される。

【００８５】

オフセット＋０はページ０の最初の４ＫＢ領域を示し、オフセット＋１はページ０の２番目の４ＫＢ領域を示し、オフセット＋２はページ０の３番目の４ＫＢ領域を示し、オフセット＋３はページ０の４番目の４ＫＢ領域を示す。

オフセット＋４はページ１の最初の４ＫＢ領域を示し、オフセット＋５はページ１の２番目の４ＫＢ領域を示し、オフセット＋６はページ１の３番目の４ＫＢ領域を示し、オフセット＋７はページ１の４番目の４ＫＢ領域を示す。

【００８６】

図１２は、ライトコマンドに応じて実行される書き込み動作を示す。

いま、ブロックＢＬＫ＃１が書き込み先ブロックとして割り当てられている場合を想定する。コントローラ４は、ページ０、ページ１、ページ２、…ページｎという順序で、データをページ単位でブロックＢＬＫ＃１に書き込む。

【００８７】

図１２においては、ブロックＢＬＫ＃１のページ０に１６Ｋバイト分のデータがすでに書き込まれている状態で、ブロック番号（＝ＢＬＫ＃１）、論理アドレス（ＬＢＡｘ）および長さ（＝４）を指定するライトコマンドがホスト２から受信された場合が想定されている。コントローラ４は、ブロックＢＬＫ＃１のページ１を書き込み先位置として決定し、ホスト２から受信される１６Ｋバイト分のライトデータをブロックＢＬＫ＃１のページ１に書き込む。そして、コントローラ４は、このライトコマンドに対するレスポンスとして、オフセット（ブロック内オフセット）、長さをホスト２に返す。このケースにおいては、オフセット（ブロック内オフセット）は＋５であり、長さは４である。あるいは、コントローラ４は、このライトコマンドに対するレスポンスとして、論理アドレス、ブロック

番号、オフセット（ブロック内オフセット）、長さをホスト２に返してもよい。このケースにおいては、論理アドレスはＬＢＡｘであり、ブロック番号はＢＬＫ＃１であり、オフセット（ブロック内オフセット）は＋５であり、長さは４である。

【００８８】

図１３は、不良ページ（バッドページ）をスキップする書き込み動作を示す。

図１３においては、ブロックＢＬＫ＃１のページ０、ページ１にデータがすでに書き込まれている状態で、ブロック番号（＝ＢＬＫ＃１）、論理アドレス（ＬＢＡｘ＋１）および長さ（＝４）を指定するライトコマンドがホスト２から受信された場合が想定されている。もしブロックＢＬＫ＃１のページ２が不良ページであるならば、コントローラ４は、ブロックＢＬＫ＃１のページ３を書き込み先位置として決定し、ホスト２から受信される１６Ｋバイト分のライトデータをブロックＢＬＫ＃１のページ３に書き込む。そして、コントローラ４は、このライトコマンドに対するレスポンスとして、オフセット（ブロック内オフセット）、長さをホスト２に返す。このケースにおいては、オフセット（ブロック内オフセット）は＋１２であり、長さは４である。あるいは、コントローラ４は、このライトコマンドに対するレスポンスとして、論理アドレス、ブロック番号、オフセット（ブロック内オフセット）、長さをホスト２に返してもよい。このケースにおいては、論理アドレスはＬＢＡｘ＋１であり、ブロック番号はＢＬＫ＃１であり、オフセット（ブロック内オフセット）は＋１２であり、長さは４である。

【００８９】

図１４は、不良ページをスキップする書き込み動作の別の例を示す。

図１４においては、不良ページを挟む２つのページに跨がってデータが書き込まれる場合が想定されている。いま、ブロックＢＬＫ＃２のページ０、ページ１にデータがすでに書き込まれており、且つ内部バッファ（共有キャッシュ）３１に未書き込みの８Ｋバイト分のライトデータが残っている場合を想定する。この状態で、ブロック番号（＝ＢＬＫ＃２）、論理アドレス（ＬＢＡｙ）および長さ（＝６）を指定するライトコマンドが受信されたならば、コントローラ４は、未書き込みの８Ｋバイトライトデータと、ホスト２から新たに受信される２４Ｋバイトライトデータ内の最初の８Ｋバイトライトデータとを使用して、ページサイズに対応する１６Ｋバイトライトデータを準備する。そして、コントローラ４は、この準備した１６ＫバイトライトデータをブロックＢＬＫ＃２のページ２に書き込む。

【００９０】

もしブロックＢＬＫ＃２の次のページ３が不良ページであるならば、コントローラ４は、ブロックＢＬＫ＃２のページ４を次の書き込み先位置として決定し、ホスト２から受信された２４Ｋバイトライトデータ内の残りの１６Ｋバイトライトデータを、ブロックＢＬＫ＃２のページ４に書き込む。

【００９１】

そして、コントローラ４は、このライトコマンドに対するレスポンスとして、２つのオフセット（ブロック内オフセット）と、２つの長さをホスト２に返す。このケースにおいては、このレスポンスは、オフセット（＝＋１０）、長さ（＝２）、オフセット（＝＋１６）、長さ（＝４）を含んでもよい。あるいは、コントローラ４は、このライトコマンドに対するレスポンスとして、ＬＢＡｙ、ブロック番号（＝ＢＬＫ＃２）、オフセット（＝＋１０）、長さ（＝２）、ブロック番号（＝ＢＬＫ＃２）、オフセット（＝＋１６）、長さ（＝４）をホスト２に返してもよい。

【００９２】

図１５、図１６は、論理アドレスとデータのペアをブロック内のページに書き込む動作を示す。

各ブロックにおいて、各ページは、ユーザデータを格納するためのユーザデータ領域と管理データを格納するための冗長領域とを含んでもよい。ページサイズは１６ＫＢ＋アルファである。

【００９３】

コントローラ４は、４ＫＢユーザデータとこの４ＫＢユーザデータに対応する論理アドレス（例えばＬＢＡ）との双方を書き込み先ブロックＢＬＫに書き込む。この場合、図１５に示すように、各々がＬＢＡと４ＫＢユーザデータとを含む４つのデータセットが同じページに書き込まれてもよい。ブロック内オフセットは、セット境界を示してもよい。

【００９４】

あるいは、図１６に示されているように、４つの４ＫＢユーザデータがページ内のユーザデータ領域に書き込まれ、これら４つの４ＫＢユーザデータに対応する４つのＬＢＡがこのページ内の冗長領域に書き込まれてもよい。

図１７は、スーパーブロックが使用されるケースにおけるブロック番号とオフセット（ブロック内オフセット）との関係を示す。以下では、ブロック内オフセットは単にオフセットとしても参照される。

【００９５】

ここでは、図示を簡単化するために、ある一つのスーパーブロックＳＢ＃１が４つのブロックＢＬＫ＃１１、ＢＬＫ＃２１、ＢＬＫ＃３１、ＢＬＫ＃４１から構成されている場合が想定されている。コントローラ４は、ブロックＢＬＫ＃１１のページ０、ブロックＢＬＫ＃２１のページ０、ブロックＢＬＫ＃３１のページ０、ブロックＢＬＫ＃４１のページ０、ブロックＢＬＫ＃１１のページ１、ブロックＢＬＫ＃２１のページ１、ブロックＢＬＫ＃３１のページ１、ブロックＢＬＫ＃４１のページ１、…という順序でデータを書き込む。

【００９６】

オフセット＋０はブロックＢＬＫ＃１１のページ０の最初の４ＫＢ領域を示し、オフセット＋１はブロックＢＬＫ＃１１のページ０の２番目の４ＫＢ領域を示し、オフセット＋２はブロックＢＬＫ＃１１のページ０の３番目の４ＫＢ領域を示し、オフセット＋３はブロックＢＬＫ＃１１のページ０の４番目の４ＫＢ領域を示す。

【００９７】

オフセット＋４はブロックＢＬＫ＃２１のページ０の最初の４ＫＢ領域を示し、オフセット＋５はブロックＢＬＫ＃２１のページ０の２番目の４ＫＢ領域を示し、オフセット＋６はブロックＢＬＫ＃２１のページ０の３番目の４ＫＢ領域を示し、オフセット＋７はブロックＢＬＫ＃２１のページ０の４番目の４ＫＢ領域を示す。

【００９８】

同様に、オフセット＋１２はブロックＢＬＫ＃４１のページ０の最初の４ＫＢ領域を示し、オフセット＋１３はブロックＢＬＫ＃４１のページ０の２番目の４ＫＢ領域を示し、オフセット＋１４はブロックＢＬＫ＃４１のページ０の３番目の４ＫＢ領域を示し、オフセット＋１５はブロックＢＬＫ＃４１のページ０の４番目の４ＫＢ領域を示す。

【００９９】

オフセット＋１６はブロックＢＬＫ＃１１のページ１の最初の４ＫＢ領域を示し、オフセット＋１７はブロックＢＬＫ＃１１のページ１の２番目の４ＫＢ領域を示し、オフセット＋１８はブロックＢＬＫ＃１１のページ１の３番目の４ＫＢ領域を示し、オフセット＋１９はブロックＢＬＫ＃１１のページ１の４番目の４ＫＢ領域を示す。

【０１００】

オフセット＋２０はブロックＢＬＫ＃２１のページ１の最初の４ＫＢ領域を示し、オフセット＋２１はブロックＢＬＫ＃２１のページ１の２番目の４ＫＢ領域を示し、オフセット＋２２はブロックＢＬＫ＃２１のページ１の３番目の４ＫＢ領域を示し、オフセット＋２３はブロックＢＬＫ＃２１のページ１の４番目の４ＫＢ領域を示す。

【０１０１】

同様に、オフセット＋２８はブロックＢＬＫ＃４１のページ１の最初の４ＫＢ領域を示し、オフセット＋２９はブロックＢＬＫ＃４１のページ１の２番目の４ＫＢ領域を示し、オフセット＋３０はブロックＢＬＫ＃４１のページ１の３番目の４ＫＢ領域を示し、オフセット＋３１はブロックＢＬＫ＃４１のページ１の４番目の４ＫＢ領域を示す。

【０１０２】

図１８は、フラッシュストレージデバイス３に適用される最大ブロック番号ゲットコマンドを示す。
最大ブロック番号ゲットコマンドは、フラッシュストレージデバイス３から最大ブロック番号を取得するためのコマンドである。ホスト２は、フラッシュストレージデバイス３に最大ブロック番号ゲットコマンドに送信することにより、フラッシュストレージデバイス３に含まれるブロックの数を示す最大ブロック番号を認識することができる。最大ブロック番号ゲットコマンドは、最大ブロック番号ゲットコマンド用のコマンドＩＤを含み、パラメータは含まない。
【０１０３】
図１９は、最大ブロック番号ゲットコマンドに対するレスポンスを示す。
最大ブロック番号ゲットコマンドをホスト２から受信した時、フラッシュストレージデバイス３は、図１９に示すレスポンスをホスト２に返す。このレスポンスは、最大ブロック番号（つまり、フラッシュストレージデバイス３に含まれる利用可能なブロックの総数）を示すパラメータを含む。
【０１０４】
図２０は、フラッシュストレージデバイス３に適用されるブロックサイズゲットコマンドを示す。
ブロックサイズゲットコマンドは、フラッシュストレージデバイス３からブロックサイズを取得するためのコマンドである。ホスト２は、フラッシュストレージデバイス３にブロックサイズゲットコマンドに送信することにより、フラッシュストレージデバイス３に含まれるＮＡＮＤ型フラッシュメモリ５のブロックサイズを認識することができる。
【０１０５】
なお、別の実施形態では、ブロックサイズゲットコマンドは、ブロック番号を指定するパラメータを含んでいてもよい。あるブロック番号を指定するブロックサイズゲットコマンドをホスト２から受信した場合、フラッシュストレージデバイス３は、このブロック番号を有するブロックのブロックサイズをホスト２に返す。これにより、たとえＮＡＮＤ型フラッシュメモリ５に含まれるブロックそれぞれのブロックサイズが不均一である場合であっても、ホスト２は、個々のブロックそれぞれのブロックサイズを認識することができる。
【０１０６】
図２１は、ブロックサイズゲットコマンドに対するレスポンスを示す。
ブロックサイズゲットコマンドをホスト２から受信した時、フラッシュストレージデバイス３は、ブロックサイズ（ＮＡＮＤ型フラッシュメモリ５に含まれるブロックそれぞれの共通のブロックサイズ）をホスト２に返す。この場合、もしブロック番号がブロックサイズゲットコマンドによって指定されていたならば、フラッシュストレージデバイス３は、上述したように、このブロック番号を有するブロックのブロックサイズをホスト２に返す。
【０１０７】
図２２は、フラッシュストレージデバイス３に適用されるブロックアロケートコマンドを示す。
ブロックアロケートコマンドは、フラッシュストレージデバイス３にブロック（フリーブロック）の割り当てを要求するコマンド（ブロック割り当て要求）である。ホスト２は、ブロックアロケートコマンドをフラッシュストレージデバイス３に送信することによって、フリーブロックを割り当てるようにフラッシュストレージデバイス３に要求し、これによってブロック番号（割り当てられたフリーブロックのブロック番号）を取得することができる。
【０１０８】
フラッシュストレージデバイス３がフリーブロック群をフリーブロックリストによって管理し、ホスト２はフリーブロック群を管理しないケースにおいては、ホスト２は、フリーブロックを割り当てるようにフラッシュストレージデバイス３に要求し、これによってブロック番号を取得する。一方、ホスト２がフリーブロック群を管理するケースにおいては、ホスト２は、フリーブロック群の一つを自身で選択することができるので、ブロックア

ロケートコマンドをフラッシュストレージデバイス３に送信する必要は無い。
【０１０９】
図２３は、ブロックアロケートコマンドに対するレスポンスを示す。
ブロックアロケートコマンドをホスト２から受信した時、フラッシュストレージデバイス３は、フリーブロックリストから、ホスト２に割り当てるべきフリーブロックを選択し、選択したフリーブロックのブロック番号を含むレスポンスをホスト２に返す。
【０１１０】
図２４は、ホスト２とフラッシュストレージデバイス３とによって実行されるブロック情報取得処理を示す。
ホスト２がフラッシュストレージデバイス３の使用を開始する時、ホスト２は、まず、最大ブロック番号ゲットコマンドをフラッシュストレージデバイス３に送信する。フラッシュストレージデバイス３のコントローラは、最大ブロック番号をホスト２に返す。最大ブロック番号は、利用可能なブロックの総数を示す。なお、上述のスーパーブロックが使用されるケースにおいては、最大ブロック番号は、利用可能なスーパーブロックの総数を示してもよい。
【０１１１】
次いで、ホスト２は、ブロックサイズゲットコマンドをフラッシュストレージデバイス３に送信して、ブロックサイズを取得する。この場合、ホスト２は、ブロック番号１を指定するブロックサイズゲットコマンド、ブロック番号２を指定するブロックサイズゲットコマンド、ブロック番号３を指定するブロックサイズゲットコマンド、…をフラッシュストレージデバイス３にそれぞれ送信して、全てのブロックそれぞれのブロックサイズを個別に取得してもよい。
【０１１２】
このブロック情報取得処理により、ホスト２は、利用可能ブロック数、個々のブロックのブロックサイズを認識することができる。
図２５は、ホスト２とフラッシュストレージデバイス３とによって実行される書き込み処理のシーケンスを示す。
ホスト２は、まず、書き込みのために使用すべきブロック（フリーブロック）を自身で選択するか、またはブロックアロケートコマンドをフラッシュストレージデバイス３に送信することによってフリーブロックを割り当てるようにフラッシュストレージデバイス３に要求する。そして、ホスト２は、自身で選択したブロックのブロック番号ＢＬＫ＃（またはフラッシュストレージデバイス３によって割り当てられたフリーブロックのブロック番号ＢＬＫ＃）と、論理アドレス（ＬＢＡ）と、長さとを含むライトコマンドをフラッシュストレージデバイス３に送信する（ステップＳ２０）。
【０１１３】
フラッシュストレージデバイス３のコントローラ４がこのライトコマンドを受信した時、コントローラ４は、ホスト２からのライトデータを書き込むべき、このブロック番号ＢＬＫ＃を有するブロック（書き込み先ブロックＢＬＫ＃）内の書き込み先位置を決定し、この書き込み先ブロックＢＬＫ＃の書き込み先位置にライトデータを書き込む（ステップＳ１１）。ステップＳ１１では、コントローラ４は、論理アドレス（ここではＬＢＡ）とライトデータの双方を書き込み先ブロックに書き込んでもよい。
【０１１４】
コントローラ４は、書き込み先ブロックＢＬＫ＃に対応するブロック管理テーブル３２を更新して、書き込まれたデータに対応するビットマップフラグ（つまり、このデータが書き込まれたオフセット（ブロック内オフセット）に対応するビットマップフラグ）を０から１に変更する（ステップＳ１２）。
【０１１５】
例えば、図２６に示されているように、開始ＬＢＡがＬＢＡｘである１６Ｋバイト更新データがブロックＢＬＫ＃１のオフセット＋４～＋７に対応する物理記憶位置に書き込まれた場合を想定する。この場合、図２７に示されているように、ブロックＢＬＫ＃１用のブ

ロック管理テーブルにおいては、オフセット＋４～＋７に対応するビットマップフラグそれぞれが０から１に変更される。

【０１１６】

そして、図２５に示すように、コントローラ４は、このライトコマンドに対するレスポンスをホスト２に返す（ステップＳ１３）。このレスポンスは、このデータが書き込まれたオフセット（ブロック内オフセット）を少なくとも含む。

ホスト２がこのレスポンスを受信した時、ホスト２は、ホスト２によって管理されているＬＵＴ４１１を更新して、書き込まれたライトデータに対応する論理アドレスそれぞれに物理アドレスをマッピングする。図２８に示されているように、ＬＵＴ４１１は、複数の論理アドレス（例えばＬＢＡ）それぞれに対応する複数のエントリを含む。ある論理アドレス（例えばあるＬＢＡ）に対応するエントリには、このＬＢＡに対応するデータが格納されているＮＡＮＤ型フラッシュメモリ５内の位置（物理記憶位置）を示す物理アドレスＰＢＡ、つまりブロック番号、オフセット（ブロック内オフセット）が格納される。図２６に示されているように、開始ＬＢＡがＬＢＡｘである１６Ｋバイト更新データがブロックＢＬＫ＃１のオフセット＋４～＋７に対応する物理記憶位置に書き込まれたならば、図２８に示されているように、ＬＵＴ４１１が更新されて、ＬＢＡｘに対応するエントリにＢＬＫ＃１、オフセット＋４が格納され、ＬＢＡｘ＋１に対応するエントリにＢＬＫ＃１、オフセット＋５が格納され、ＬＢＡｘ＋２に対応するエントリにＢＬＫ＃１、オフセット＋６が格納され、ＬＢＡｘ＋３に対応するエントリにＢＬＫ＃１、オフセット＋７が格納される。

【０１１７】

図２５に示すように、この後、ホスト２は、上述の更新データの書き込みによって不要になった以前のデータを無効化するためのＴｒｉｍコマンドをフラッシュストレージデバイス３に送信する。図２６に示されているように、以前のデータがブロックＢＬＫ＃０のオフセット＋０、オフセット＋１、オフセット＋２、オフセット＋３に対応する位置に格納されている場合には、図２９に示すように、ブロック番号（＝ＢＬＫ＃０）、オフセット（＝＋０）、長さ（＝４）を指定するＴｒｉｍコマンドがホスト２からフラッシュストレージデバイス３に送信される。フラッシュストレージデバイス３のコントローラ４は、このＴｒｉｍコマンドに応じて、ブロック管理テーブル３２を更新する（図２５、ステップＳ１４）。ステップＳ１５においては、図２９に示すように、ブロックＢＬＫ＃０用のブロック管理テーブルにおいて、オフセット＋０～＋３に対応するビットマップフラグそれぞれが１から０に変更される。

【０１１８】

図３０は、フラッシュストレージデバイス３に適用されるリードコマンドを示す。

リードコマンドは、フラッシュストレージデバイス３にデータの読み出しを要求するコマンドである。このリードコマンドは、コマンドＩＤ、物理アドレスＰＢＡ、長さ、転送先ポインタを含む。

【０１１９】

コマンドＩＤはこのコマンドがリードコマンドであることを示すＩＤ（コマンドコード）であり、リードコマンドにはリードコマンド用のコマンドＩＤが含まれる。

物理アドレスＰＢＡは、データが読み出されるべき最初の物理記憶位置を示す。物理アドレスＰＢＡは、ブロック番号、オフセット（ブロック内オフセット）によって指定される。

【０１２０】

長さは、リードすべきデータの長さを示す。このデータ長は、Ｇｒａｉｎの数によって指定可能である。

転送先ポインタは、読み出されたデータが転送されるべきホスト２内のメモリ上の位置を示す。

一つのリードコマンドは、物理アドレスＰＢＡ（ブロック番号、オフセット）と長さの組を複数指定することができる。

【０１２１】

図３１は、リード動作を示す。
ここでは、ブロック番号（＝ＢＬＫ＃２）、オフセット（＝＋５）、長さ（＝３）を指定するリードコマンドがホスト２から受信された場合が想定されている。フラッシュストレージデバイス３のコントローラ４は、ブロック番号（＝ＢＬＫ＃２）、オフセット（＝＋５）、長さ（＝３）に基づいて、ＢＬＫ＃２からデータｄ１～ｄ３をリードする。この場合、コントローラ４は、ＢＬＫ＃２のページ１から１ページサイズ分のデータをリードし、このリードデータからデータｄ１～データｄ３を抽出する。次いで、コントローラ４は、データｄ１～データｄ３を、転送先ポインタによって指定されるホストメモリ上に転送する。

【０１２２】

図３２は、ホスト２からのリードコマンドに応じて、異なる物理記憶位置にそれぞれ格納されているデータ部をリードする動作を示す。
ここでは、ブロック番号（＝ＢＬＫ＃２）、オフセット（＝＋１０）、長さ（＝２）、ブロック番号（＝ＢＬＫ＃２）、オフセット（＝＋１６）、長さ（＝４）を指定するリードコマンドがホスト２から受信された場合が想定されている。フラッシュストレージデバイス３のコントローラ４は、ブロック番号（＝ＢＬＫ＃２）、オフセット（＝＋１０）、長さ（＝２）に基づいて、ＢＬＫ＃２のページ２から１ページサイズ分のデータをリードし、このリードデータからデータｄ１～データｄ２を抽出する。次いで、コントローラ４は、ブロック番号（＝ＢＬＫ＃２）、オフセット（＝＋１６）、長さ（＝４）に基づいて、ＢＬＫ＃２のページ４から１ページサイズ分のデータ（データｄ３～データｄ６）をリードする。そして、コントローラ４は、データｄ１～データｄ２とデータｄ３～データｄ６とを結合することによって得られる長さ（＝６）のリードデータを、リードコマンド内の転送先ポインタによって指定されるホストメモリ上に転送する。

【０１２３】

これにより、たとえブロック内に不良ページが存在する場合であっても、リードエラーを引き起こすことなく、別個の物理記憶位置からデータ部をリードすることができる。また、たとえデータが２つのブロックに跨がって書き込まれている場合であっても、このデータを一つのリードコマンドの発行によってリードすることができる。

【０１２４】

図３３は、ホスト２とフラッシュストレージデバイス３とによって実行されるリード処理のシーケンスを示す。
ホスト２は、ホスト２によって管理されているＬＵＴ４１１を参照して、ユーザアプリケーションからのリード要求に含まれる論理アドレスをブロック番号、オフセットに変換する。そして、ホスト２は、このブロック番号、オフセット、長さを指定するリードコマンドをフラッシュストレージデバイス３に送信する。

【０１２５】

フラッシュストレージデバイス３のコントローラ４がリードコマンドをホスト２から受信した時、コントローラ４は、このリードコマンドによって指定されたブロック番号に対応するブロックをリード対象のブロックとして決定するとともに、このリードコマンドによって指定されたオフセットに基づいてリード対象のページを決定する（ステップＳ３１）。ステップＳ３１では、コントローラ４は、まず、リードコマンドによって指定されたオフセットを、ページサイズを表す粒度の数（ここでは、４）で除算してもよい。そして、コントローラ４は、この除算によって得られる商および余りを、リード対象のページ番号およびリード対象のページ内オフセット位置としてそれぞれ決定してもよい。

【０１２６】

コントローラ４は、ブロック番号、オフセット、長さによって規定されるデータをＮＡＮＤ型フラッシュメモリ５からリードし（ステップＳ３２）、このリードデータをホスト２に送信する。
図３４は、フラッシュストレージデバイス３に適用されるＧＣ制御コマンドを示す。

【０１２７】

ＧＣ制御コマンドは、ＧＣソースブロック番号およびＧＣデスティネーションブロック番号をフラッシュストレージデバイス３に通知するために使用される。ホスト２は、各ブロックの有効データ量／無効データ量を管理しており、有効データ量がより少ない幾つかのブロックをＧＣソースブロックとして選択することができる。また、ホスト２は、フリーブロックリストを管理しており、幾つかのフリーブロックをＧＣデスティネーションブロックとして選択することができる。このＧＣ制御コマンドは、コマンドＩＤ、ＧＣソースブロック番号、ＧＣデスティネーションブロック番号、等を含んでもよい。

【０１２８】

コマンドＩＤはこのコマンドがＧＣ制御コマンドであることを示すＩＤ（コマンドコード）であり、ＧＣ制御コマンドにはＧＣ制御コマンド用のコマンドＩＤが含まれる。

ＧＣソースブロック番号は、ＧＣソースブロックを示すブロック番号である。ホスト２は、どのブロックをＧＣソースブロックとすべきかを指定することができる。ホスト２は、複数のＧＣソースブロック番号を一つのＧＣ制御コマンドに設定してもよい。

【０１２９】

ＧＣデスティネーションブロック番号は、ＧＣデスティネーションブロックを示すブロック番号である。ホスト２は、どのブロックをＧＣデスティネーションブロックとすべきかを指定することができる。ホスト２は、複数のＧＣデスティネーションブロック番号を一つのＧＣ制御コマンドに設定してもよい。

【０１３０】

図３５は、ＧＣ用コールバックコマンドを示す。

ＧＣ用コールバックコマンドは、ＧＣによってコピーされた有効データの論理アドレスとこの有効データのコピー先位置を示すブロック番号およびオフセットとをホスト２に通知するために使用される。

ＧＣ用コールバックコマンドは、コマンドＩＤ、論理アドレス、長さ、デスティネーション物理アドレスを含んでよい。

【０１３１】

コマンドＩＤはこのコマンドがＧＣ用コールバックコマンドであることを示すＩＤ（コマンドコード）であり、ＧＣ用コールバックコマンドにはＧＣ用コールバックコマンド用のコマンドＩＤが含まれる。

論理アドレスは、ＧＣによってＧＣソースブロックからＧＣデスティネーションブロックにコピーされた有効データの論理アドレスを示す。

【０１３２】

長さは、このコピーされたデータの長さを示す。このデータ長は、粒度（Ｇｒａｉｎ）の数によって指定されてもよい。

デスティネーション物理アドレスは、有効データがコピーされたＧＣデスティネーションブロック内の位置を示す。デスティネーション物理アドレスは、ブロック番号、オフセット（ブロック内オフセット）によって指定される。

【０１３３】

図３６は、ガベージコレクション（ＧＣ）動作の手順を示す。

例えば、ホスト２は、ホスト２によって管理されているフリーブロックリストに含まれている残りフリーブロックの数が閾値以下に低下した場合、ＧＣソースブロックおよびＧＣデスティネーションブロックを選択し、選択されたＧＣソースブロックおよび選択されたＧＣデスティネーションブロックを指定するＧＣ制御コマンドをフラッシュストレージデバイス３に送信する（ステップＳ４１）。あるいは、ライト処理部４１２がフリーブロック群を管理する構成においては、残りフリーブロックの数が閾値以下に低下した際にライト処理部４１２がホスト２にその旨通知を行ない、通知を受信したホスト２がブロック選択およびＧＣ制御コマンドの送信を行なってもよい。

【０１３４】

このＧＣ制御コマンドを受信すると、フラッシュストレージデバイス３のコントローラ４は、ＧＣソースブロック内の有効データを書き込むべきＧＣデスティネーションブロック

内の位置（コピー先位置）を決定する動作と、ＧＣソースブロック内の有効データをＧＣデスティネーションブロック内のコピー先位置にコピーする動作とを含むデータコピー動作を実行する（ステップＳ５１）。ステップＳ５１では、コントローラ４は、ＧＣソースブロック（コピー元ブロック）内の有効データのみならず、この有効データとこの有効データに対応する論理アドレスの双方を、ＧＣソースブロック（コピー元ブロック）からＧＣデスティネーションブロック（コピー先ブロック）にコピーする。これにより、ＧＣデスティネーションブロック（コピー先ブロック）内にデータと論理アドレスとのペアを保持することができる。

【０１３５】

また、ステップＳ５１では、ＧＣソースブロック内の全ての有効データのコピーが完了するまでデータコピー動作が繰り返し実行される。複数のＧＣソースブロックがＧＣ制御コマンドによって指定された場合には、全てのＧＣソースブロック内の全ての有効データのコピーが完了するまでデータコピー動作が繰り返し実行される。

【０１３６】

そして、コントローラ４は、コピーされた有効データ毎に、その有効データの論理アドレス（ＬＢＡ）と、その有効データのコピー先位置を示すデスティネーション物理アドレス等を、ＧＣ用コールバックコマンドを使用してホスト２に通知する（ステップＳ５２）。ある有効データに対応するデスティネーション物理アドレスは、この有効データがコピーされたコピー先ブロック（ＧＣデスティネーションブロック）のブロック番号と、この有効データがコピーされたコピー先ブロック内の物理記憶位置を示すブロック内物理アドレス（ブロック内オフセット）とによって表される。

【０１３７】

ホスト２がこのＧＣ用コールバックコマンドを受信した時、ホスト２は、ホスト２によって管理されているＬＵＴ４１１を更新して、コピーされた各有効データに対応する論理アドレスにデスティネーション物理アドレス（ブロック番号、ブロック内オフセット）をマッピングする（ステップＳ４２）。

【０１３８】

図３７は、ＧＣのために実行されるデータコピー動作の例を示す。

図３７では、ＧＣソースブロック（ここではブロックＢＬＫ＃５０）のオフセット＋４に対応する位置に格納されている有効データ（ＬＢＡ＝１０）が、ＧＣデスティネーションブロック（ここではブロックＢＬＫ＃１００）のオフセット＋０に対応する位置にコピーされ、ＧＣソースブロック（ここではブロックＢＬＫ＃５０）のオフセット＋１０に対応する位置に格納されている有効データ（ＬＢＡ＝２０）が、ＧＣデスティネーションブロック（ここではブロックＢＬＫ＃１００）のオフセット＋１に対応する位置にコピーされた場合が想定されている。この場合、コントローラ４は、｛ＬＢＡ１０、ＢＬＫ＃１００、オフセット（＝＋０）、ＬＢＡ２０、ＢＬＫ＃１００、オフセット（＝＋１）｝をホストに通知する（ＧＣ用コールバック処理）。

【０１３９】

図３８は、図３７のデータコピー動作の結果に基づいて更新されるホスト２のＬＵＴ４１１の内容を示す。

このＬＵＴ４１１においては、ＬＢＡ１０に対応するブロック番号およびオフセットは、ＢＬＫ＃５０、オフセット（＝＋４）から、ＢＬＫ＃１００、オフセット（＝＋０）に更新される。同様に、ＬＢＡ２０に対応するブロック番号およびオフセットは、ＢＬＫ＃５０、オフセット（＝＋１０）から、ＢＬＫ＃１００、オフセット（＝＋１）に更新される。

【０１４０】

ＬＵＴ４１１が更新された後、ホスト２は、ＢＬＫ＃５０およびオフセット（＝＋４）を指定するＴｒｉｍコマンドをフラッシュストレージデバイス３に送信して、ＢＬＫ＃５０のオフセット（＝＋４）に対応する位置に格納されているデータを無効化してもよい。さらに、ホスト２は、ＢＬＫ＃５０およびオフセット（＝＋１０）を指定するＴｒｉｍコマンドをフラッシュストレージデバイス３に送信して、ＢＬＫ＃５０のオフセット（＝＋１

０）に対応する位置に格納されているデータを無効化してもよい。
あるいは、ホスト２からＴｒｉｍコマンドを送信せず、ＧＣ処理の一環としてコントローラ４がブロック管理テーブル３２を更新してこれらのデータを無効化してもよい。
【０１４１】
以上説明したように、本実施形態によれば、第１の論理アドレスと第１のブロック番号とを指定するライト要求をホスト２から受信した場合、フラッシュストレージデバイス３のコントローラ４は、ホスト２からのデータを書き込むべき、第１のブロック番号を有するブロック（書き込み先ブロック）内の位置（書き込み先位置）を決定し、ホスト２からのデータを書き込み先ブロックの書き込み先位置に書き込み、第１の位置を示す第１のブロック内物理アドレス、または第１の論理アドレスと第１のブロック番号と第１のブロック内物理アドレスとの組のいずれかを、ホスト２に通知する。
【０１４２】
したがって、ホスト２がブロック番号をハンドリングし、フラッシュストレージデバイス３がページ書き込み順序制約／バッドページ等を考慮して、ホスト２によって指定されるブロック番号を有するブロック内の書き込み先位置（ブロック内オフセット）を決定するという構成を実現できる。ホスト２がブロック番号をハンドリングすることにより、上位階層（ホスト２）のアプリケーションレベルアドレス変換テーブルと従来型ＳＳＤのＬＵＴレベルアドレス変換テーブルとのマージを実現できる。また、フラッシュストレージデバイス３は、ＮＡＮＤ型フラッシュメモリ５の特徴／制約を考慮してＮＡＮＤ型フラッシュメモリ５を制御することができる。さらに、ホスト２はブロック境界を認識することができるので、ブロック境界／ブロックサイズを考慮してユーザデータを各ブロックに書き込むことができる。これにより、ホスト２が同一ブロック内のデータをデータ更新等によって一斉に無効化する等の制御を行うことが可能となるので、ＧＣが実行される頻度を下げることが可能となる。この結果、ライトアンプリフィケーションが低下され、フラッシュストレージデバイス３の性能の向上、フラッシュストレージデバイス３の寿命の最大化を実現できる。
【０１４３】
したがって、ホスト２とフラッシュストレージデバイス３との間の適切な役割分担を実現でき、これによってホスト２とフラッシュストレージデバイス３とを含むシステム全体のＩ／Ｏ性能の向上を図ることができる。
また、ガベージコレクションのためのコピー元ブロック番号およびコピー先ブロック番号を指定する制御コマンドをホスト２から受信した場合、フラッシュストレージデバイス３のコントローラ４は、複数のブロックから、コピー元ブロック番号を有する第２のブロックとコピー先ブロック番号を有する第３のブロックとを選択し、第２のブロックに格納されている有効データを書き込むべき第３のブロック内のコピー先位置を決定し、有効データを第３のブロックのコピー先位置にコピーする。そして、コントローラは、有効データの論理アドレスと、コピー先ブロック番号と、第３のブロック内のコピー先位置を示す第２のブロック内物理アドレスとを、ホスト２に通知する。これにより、ＧＣにおいても、ホスト２がブロック番号（コピー元ブロック番号、コピー先ブロック番号）のみをハンドリングし、フラッシュストレージデバイス３がコピー先ブロック内のコピー先位置を決定する、という構成を実現できる。
【０１４４】
なお、フラッシュストレージデバイス３は、ストレージアレイ内に設けられる複数のフラッシュストレージデバイス３の一つとして利用されてもよい。ストレージアレイは、サーバ計算機のような情報処理装置にケーブルまたはネットワークを介して接続されてもよい。ストレージアレイは、このストレージアレイ内の複数のフラッシュストレージデバイス３を制御するコントローラを含む。フラッシュストレージデバイス３がストレージアレイに適用された場合には、このストレージアレイのコントローラが、フラッシュストレージデバイス３のホスト２として機能してもよい。
【０１４５】

また、本実施形態では、不揮発性メモリとしてＮＡＮＤ型フラッシュメモリを例示した。しかし、本実施形態の機能は、例えば、ＭＲＡＭ（Ｍａｇｎｅｔｏｒｅｓｉｓｔｉｖｅ Ｒａｎｄｏｍ Ａｃｃｅｓｓ Ｍｅｍｏｒｙ）、ＰＲＡＭ（Ｐｈａｓｅ ｃｈａｎｇｅ Ｒａｎｄｏｍ Ａｃｃｅｓｓ Ｍｅｍｏｒｙ）、ＲｅＲＡＭ（Ｒｅｓｉｓｔｉｖｅ Ｒａｎｄｏｍ Ａｃｃｅｓｓ Ｍｅｍｏｒｙ）、又は、ＦｅＲＡＭ（Ｆｅｒｒｏｅｌｅｃｔｒｉｃ Ｒａｎｄｏｍ Ａｃｃｅｓｓ Ｍｅｍｏｒｙ）のような他の様々な不揮発性メモリにも適用できる。

【０１４６】

図３９は、ホスト２とフラッシュストレージデバイス３とのシステムアーキテクチャを示す。具体的には、図３９は、ホスト２に含まれるライトデータバッファ５１およびフラッシュトランスレーション部５２と、フラッシュストレージデバイス３に含まれるライト動作制御部２１、リード動作制御部２２および（ＧＣ動作制御部２３を含む）最適化処理部５３との関係を示す。

【０１４７】

ホスト２は、ライトデータをホストメモリ上のライトデータバッファ５１に格納し、そしてライトコマンドをフラッシュストレージデバイス３に発行する。このライトコマンドは、このライトデータが存在するライトデータバッファ５１上の位置を示すデータポインタと、このライトデータを識別するタグ（例えばＬＢＡ）と、このライトデータの長さと、ライトデータが書き込まれるべきブロックを示す識別子（ブロックアドレス、またはストリームＩＤ）とを含んでいてもよい。

【０１４８】

フラッシュストレージデバイス３は、以下のタイプ＃１－ストレージデバイス、タイプ＃２－ストレージデバイス、タイプ＃３－ストレージデバイスのうちの任意のストレージデバイスとして実現されうる。

タイプ＃１－ストレージデバイスは、ホスト２が、データが書き込まれるべきブロックとこのデータが書き込まれるべきページアドレスの双方を指定するタイプのストレージデバイスである。タイプ＃１－ストレージデバイスに適用されるライトコマンドは、ブロックアドレス、ページアドレス、データポインタ、長さを含む。ブロックアドレスは、ホスト２から受信されるライトデータが書き込まれるべきブロックを指定する。ページアドレスは、このライトデータが書き込まれるべきこのブロック内のページを指定する。データポインタは、このライトデータが存在するホスト２内のメモリ上の位置を示す。長さは、このライトデータの長さを示す。

【０１４９】

タイプ＃２－ストレージデバイスは、ホスト２がデータが書き込まれるべきブロックを指定し、ストレージデバイスがこのデータが書き込まれるべきこのブロック内の位置（ページ）を指定するタイプのストレージデバイスである。タイプ＃２－ストレージデバイスに適用されるライトコマンドは、書き込まれるべきライトデータを識別するためのタグ（例えば、ＬＢＡ、キー）、ブロックアドレス、データポインタ、長さを含む。さらに、ライトコマンドは、ＱｏＳドメインＩＤを含んでもよい。ＱｏＳドメインＩＤは、ＮＡＮＤ型フラッシュメモリを論理的に分割することによって得られる複数の領域の一つを指定する。複数の領域の各々は、複数のブロックを含む。タイプ＃２－ストレージデバイスは、不良ページ、ページ書き込み順序の制約を考慮して、データが書き込まれるべきページを決定することができる。

【０１５０】

つまり、フラッシュストレージデバイス３がタイプ＃２－ストレージデバイスとして実現されているケースにおいては、フラッシュストレージデバイス３は、ブロックをホスト２にハンドリングさせつつ、ページ書き込み順序制約、バッドページ、ページサイズ等を隠蔽することができる。この結果、ホスト２は、ブロック境界を認識でき、且つページ書き込み順序制約、バッドページ、ページサイズについては意識することなく、どのユーザデータがどのブロックに存在するかを管理することができる。

【０１５１】
タイプ＃３－ストレージデバイスは、ホスト２がデータを識別するタグ（例えばＬＢＡ）を指定し、ストレージデバイスがこのデータが書き込まれるべきブロックおよびページの双方を決定するタイプのストレージデバイスである。タイプ＃３－ストレージデバイスに適用されるライトコマンドは、書き込まれるべきライトデータを識別するためのタグ（例えば、ＬＢＡ、キー）、ストリームＩＤ、データポインタ、長さを含む。ストリームＩＤは、このライトデータに関連付けられたストリームの識別子である。フラッシュストレージデバイス３がタイプ＃３－ストレージデバイスとして実現されているケースにおいては、フラッシュストレージデバイス３は、ストリームＩＤそれぞれとブロックアドレスそれぞれとの間のマッピングを管理する管理テーブルを参照して、このライトデータが書き込まれるべきブロックを決定する。さらに、フラッシュストレージデバイス３は、論理物理アドレス変換テーブルと称されるアドレス変換テーブルを使用して、タグ（ＬＢＡ）それぞれとＮＡＮＤ型フラッシュメモリの物理アドレスそれぞれとの間のマッピングを管理する。
【０１５２】
フラッシュストレージデバイス３がタイプ＃１－ストレージデバイスとして実現されている場合、フラッシュストレージデバイス３においては、ライト動作制御部２１の制御の下、このブロックの識別子によって指定される書き込み先ブロックの書き込み動作の進行に合わせて、ライトデータバッファ５１から内部バッファ（共有キャッシュ）３１へのデータ転送がＤＭＡＣによって実行される。このデータ転送は、ＮＡＮＤ型フラッシュメモリ５のデータ書き込み単位と同じデータサイズの単位で実行される。ライト動作制御部２１の制御の下、書き込むべきライトデータが、内部バッファ（共有キャッシュ）３１から、この書き込み先ブロックを含むＮＡＮＤ型フラッシュメモリチップ１５に転送され、そして、ライト動作制御部２１からこのＮＡＮＤ型フラッシュメモリチップ１５に書き込み指示用のＮＡＮＤコマンドが送出される。
【０１５３】
フラッシュストレージデバイス３がタイプ＃２－ストレージデバイスとして実現されている場合には、ライト動作制御部２１は、ホスト２から受信されるブロック割り当て要求に応答して、フリーブロックの一つを書き込み先ブロックとしてホスト２に割り当てる処理も実行する。ブロック割り当て要求は、ＱｏＳドメインＩＤを含んでいてもよい。ライト制御部２１は、このＱｏＳドメインＩＤに属するフリーブロックの一つを書き込み先ブロックとして決定し、この書き込み先ブロックのブロックアドレスをホスト２に通知する。これにより、ホスト２は、このブロックアドレスと、データポインタと、タグ（例えばＬＢＡ）と、長さとを指定するライトコマンドを発行することができる。このライトデータがこの書き込み先ブロックに書き込まれた後、ライト動作制御部２１は、このライトデータが書き込まれた書き込み先ブロックを示すブロックアドレスと、このライトデータが書き込まれたこの書き込み先ブロック内のページを示すページアドレスと、このライトデータのタグ（例えばＬＢＡ）とをホスト２に通知する。ホスト２のフラッシュトランスレーション部５２は、タグ（例えばＬＢＡ）それぞれとＮＡＮＤ型フラッシュメモリ５の物理アドレス（ブロックアドレス、ページアドレス、等）それぞれとの間のマッピングを管理するアドレス変換テーブルであるＬＵＴ４１１を含む。ブロックアドレス、ページアドレス、およびタグ（例えばＬＢＡ）がフラッシュストレージデバイス３から通知された場合、フラッシュトランスレーション部５２は、ＬＵＴ４１１を更新し、通知されたタグ（例えばＬＢＡ）に、通知された物理アドレス（ブロックアドレス、ページアドレス）をマッピングする。フラッシュトランスレーション部５２は、ＬＵＴ４１１を参照することによって、リード要求に含まれるタグ（例えばＬＢＡ）を物理アドレス（ブロックアドレス、ページアドレス）に変換することができ、これによって物理アドレスを含むリードコマンドをフラッシュストレージデバイス３に発行することができる。
【０１５４】
フラッシュストレージデバイス３がタイプ＃１－ストレージデバイスまたはタイプ＃２－

ストレージデバイスとして実現されている場合、リード動作制御部２２は、リードコマンドに含まれる物理アドレスに基づき、ＮＡＮＤ型フラッシュメモリチップ１５に読み出し指示用のＮＡＮＤコマンドを送出する。フラッシュストレージデバイス３がタイプ＃３－ストレージデバイスとして実現されている場合には、リード動作制御部２２は、アドレス変換テーブルを参照して、リードコマンドに含まれるタグ（ＬＢＡ）に対応する物理アドレスを取得し、取得した物理アドレスに基づき、ＮＡＮＤ型フラッシュメモリチップ１５に読み出し指示用のＮＡＮＤコマンドを送出する。

【０１５５】

リード動作制御部２２の制御の下、ＮＡＮＤ型フラッシュメモリチップ１５から読み出されたデータは、内部バッファ（共有キャッシュ）３１に転送される。そして、リード動作制御部２２の制御の下、内部バッファ（共有キャッシュ）３１からホスト２へのデータ転送がＤＭＡＣによって実行される。また、リード動作制御部２２は、読み出すべきリードデータがホスト２のライトデータバッファ５１内に存在する場合、ライトデータバッファ５１からリードデータを取得することができる。あるいは、ホスト２に対して、リードデータをライトデータバッファ５１から取得することを指示してもよい。なお、ライトデータバッファ５１上のライトデータが格納される領域は、ライト動作制御部２１によるＮＡＮＤ型フラッシュメモリ５への書き込みが完了した場合にライト動作制御部２１からホスト２へ送信される解放可能通知により、ホスト２側において解放される。たとえば、ライト動作制御部２１によるＮＡＮＤ型フラッシュメモリ５への書き込みが失敗し、ライトデータが別の場所（異なるページやブロック）に書き込まれる場合、その書き込みに必要なデータについて、まだ解放されていない、ホスト２のライトデータバッファ５１の領域からフラッシュストレージデバイス３の内部バッファ（共有キャッシュ）３１へのデータ転送が再実行される。データの再書き込みは、エラーが検出された範囲で実行されてもよいし、ライトコマンドの範囲すべてで実行されてもよい。解放可能通知は、ライトコマンド単位でホスト２に通知されてもよいし、ホスト２のデータ使用単位でホスト２に通知されてもよい。

【０１５６】

（ＧＣ動作制御部２３を含む）最適化処理部５３は、たとえば、ホスト２から受信されるブロック解放要求に応答して、割り当て済みのブロックをフリーブロックに戻す処理などを実行する。ホスト２は、ブロック解放要求を、ブロックリユースコマンドとしてフラッシュストレージデバイス３に送信する。ブロックリユースコマンドで指定され得る割り当て済みのブロックは、フラッシュストレージデバイス３がタイプ＃１－ストレージデバイスとして実現され、かつ、ホスト２がフリーブロック群を管理しない場合、または、フラッシュストレージデバイス３がタイプ＃２－ストレージデバイスとして実現されている場合において、ブロックアロケートコマンドとしてホスト２から受信したブロック割り当て要求に応答して、フリーブロックの中から割り当てたブロックである。また、最適化処理部５３は、たとえば、ホスト２から受信されるＧＣ制御コマンドに応答して、あるブロックのデータを別のブロックにコピーする処理などを実行する。

【０１５７】

また、フラッシュストレージデバイス３がホスト２から受信する各種コマンドには、優先度が含まれ得る。つまり、フラッシュストレージデバイス３は、先にホスト２から受信したコマンドに優先して、後からホスト２から受信したコマンドを実行し得る。コマンドの実行順の制御は、たとえば、ホスト２から受信される各種コマンドが一時的に格納されるＩ／Ｏコマンドキューからコマンドを取り出す際にコマンド間の優先度を比較することによって実現できる。Ｉ／Ｏコマンドキューは、ＱｏＳドメイン毎に設けられるものであってもよいし、後述する仮想ストレージデバイス（ＶＤ：Virtual Device）毎に設けられるものであってもよいし、各フラッシュストレージデバイス３に１つずつ設けられるものであってもよい。

【０１５８】

ＮＡＮＤ型フラッシュメモリ５が複数のＮＡＮＤ型フラッシュメモリチップ１５を含むフ

ラッシュストレージデバイス３においては、１以上の仮想ストレージデバイスを定義することができる。図４０は、フラッシュストレージデバイス３上における仮想ストレージデバイスの定義例を示す。

【０１５９】

図４０中、（Ａ）は、仮想ストレージデバイス間でＮＡＮＤインタフェース１３に接続されるチャンネルが共有される、複数の仮想ストレージデバイスの定義例を示す。（Ｂ）は、仮想ストレージデバイス間でＮＡＮＤインタフェース１３に接続されるチャンネルが共有されない、複数の仮想ストレージデバイスの定義例を示す。（Ｃ）は、ＮＡＮＤ型フラッシュメモリ５が含む複数のＮＡＮＤ型フラッシュメモリチップ１５すべてを用いた、１つの仮想ストレージデバイスの定義例を示す。（Ｄ）は、ＮＡＮＤ型フラッシュメモリ５が含む複数のＮＡＮＤ型フラッシュメモリチップ１５それぞれを個別に用いた、ＮＡＮＤ型フラッシュメモリチップ１５と同数、つまり最大数の仮想ストレージデバイスの定義例を示す。

【０１６０】

このように、フラッシュストレージデバイス３上には、様々な形態で、１以上の仮想ストレージデバイスを定義することができる。仮想ストレージデバイスを定義すると、たとえばＮＡＮＤ型フラッシュメモリチップ１５の消耗度を監視するためのウェアー監視などを、この仮想ストレージデバイス毎に実行することができる。

【０１６１】

また、１以上の仮想ストレージデバイスを定義し得るフラッシュストレージデバイス３においては、仮想ストレージデバイス毎にＱｏＳドメインを管理することができる。図４１は、仮想ストレージデバイス毎にＱｏＳドメインが管理される例を示す。

フラッシュストレージデバイス３のブロックは、同一の仮想ストレージデバイス上に定義されるＱｏＳドメイン間で共有される。ブロックを取り扱う単位は、複数のブロックで構成されるスーパーブロックの単位であってもよい。つまり、スーパーブロックが、ＱｏＳドメイン間で共有されてもよい。たとえばエンドユーザ毎にＱｏＳドメインが割り当てられている場合において、ＱｏＳドメインを示すＱｏＳドメインＩＤを含むブロックアロケートコマンドがホスト２から受信されると、仮想ストレージデバイス内で共有されるフリーブロック群の中の１つのフリーブロックが、ＱｏＳドメインＩＤで示されるＱｏＳドメインに割り当てられる。

【０１６２】

一方、ＱｏＳドメインＩＤとブロックアドレスとを含むブロックリユースコマンドがホスト２から受信されると、ＱｏＳドメインＩＤで示されるＱｏＳドメインに割り当てられているブロックの中のブロックアドレスで示されるブロックが、フリーブロックとしてフリーブロック群に戻される。ＱｏＳドメインに割り当てられているブロックをフリーブロックとしてフリーブロック群に戻すことは、ブロックを解放するとも称される。解放されたブロックは、その後、たとえばホスト２からのブロックアロケートコマンドなどによって、その仮想ストレージデバイス内のいずれのＱｏＳドメインにも割り当てられ得る。

【０１６３】

ところで、あるＱｏＳドメイン内のあるブロックを対象とするブロックリユースコマンドがホスト２から受信された際、フラッシュストレージデバイス３内において、そのブロックを対象とするリードコマンドが実行中または未実行の状態であった場合、そのリードコマンドよりも先にブロックリユースコマンドが実行されてしまうと、たとえば値が不定のデータがホスト２に返却されるなどのおそれがある。前述したように、フラッシュストレージデバイス３がホスト２から受信する各種コマンドには、優先度が含まれ得るので、フラッシュストレージデバイス３においては、先にホスト２から受信したリードコマンドに優先して、後からホスト２から受信したブロックリユースコマンドが実行され得る。また、リードコマンドのほか、たとえば、ＧＣ制御コマンドに応答して、そのブロック内のデータを他のブロックにコピーする場合にも、同様の事態が発生し得る。つまり、そのブロックを対象とするデータの読み出し処理を実行中または未実行の状態にある場合、この実

行中または未実行のデータの読み出し処理において意図しないデータが読み出されてしまうおそれがある。

【０１６４】

このような事態をホスト２側の制御によって防止するためには、ホスト２において、たとえば仕掛り中のデータの読み出し処理の有無をブロック毎に管理することなどが必要となる。そこで、このフラッシュストレージデバイス３は、このような事態を防止するための仕組みを備えて、ホスト２の負担を軽減するようにしてもよい。

【０１６５】

フラッシュストレージデバイス３は、ホスト２からブロックリユースコマンドが受信された場合において、そのブロックリユースコマンドで指定されるブロックを対象とするデータの読み出し処理が実行中または未実行の状態であるならば、ホスト２に対してエラーを通知し、あるいは、実行中または未実行の処理が終了するまでブロックリユースコマンドの実行を保留し、実行中または未実行の処理が終了したら、ブロックリユースコマンドを実行する。

【０１６６】

この仕組みをフラッシュストレージデバイス３が備えることで、ホスト２は、たとえば解放しようとするブロックを対象とする仕掛り中のデータの読み出し処理の有無などを気にせずに、ブロックリユースコマンドをフラッシュストレージデバイス３に送信することができるようになる。つまり、ホスト２の負担を軽減することが実現される。

【０１６７】

この仕組みは、たとえば、最適化処理部５３が、ブロックリユースコマンドの受信時またはブロックリユースコマンドの実行時、そのブロックリユースコマンドで指定されるブロックを対象とするリードコマンドやＧＣ制御コマンドがＩ／Ｏコマンドキュー４２に格納されていないかを検索することによって実現できる。図４１においては、Ｉ／Ｏコマンドキュー４２がＱｏＳドメイン毎に設けられる例が示されており、この場合は、最適化処理部５３は、ブロックリユースコマンドに含まれるＱｏＳドメインＩＤで示されるＱｏＳドメインに対して設けられるＩ／Ｏコマンドキュー４２について、ブロックリユースコマンドに含まれるブロックアドレスで示されるブロックを対象とするリードコマンドまたはＧＣ制御コマンドが存在していないかどうかを調べる。Ｉ／Ｏコマンドキュー４２が仮想ストレージデバイス毎または各フラッシュストレージデバイスに１つずつ設けられる場合には、最適化処理部５３は、そのＩ／Ｏコマンドキュー４２について、ブロックリユースコマンドに含まれるＱｏＳドメインＩＤで示されるＱｏＳドメインの中のブロックリユースコマンドに含まれるブロックアドレスで示されるブロックを対象とするリードコマンドまたはＧＣ制御コマンドが存在していないかどうかを調べる。存在する場合、最適化処理部５３は、ホスト２に対してエラーを通知し、あるいは、Ｉ／Ｏコマンドキュー４２に存在する、ブロックリユースコマンドで指定されるブロックを対象とするリードコマンドまたはＧＣ制御コマンドが終了するまでブロックリユースコマンドの実行を保留し、それらが終了したら、ブロックリユースコマンドを実行する。

【０１６８】

あるいは、この仕組みは、たとえば、フリーブロック群から選ばれてＱｏＳドメインに割り当てられたブロックそれぞれについて、そのブロックを対象として実行中のリードコマンドの数、および、そのブロックをコピー元として実行中のＧＣ制御コマンドの数を示すカウンタをメタデータなどとして設けることによって実現できる。たとえば、リード動作制御部２１や（ＧＣ動作制御部２３を含む）最適化処理部５３は、あるブロックを対象とするデータの読み出し処理を実行する場合、そのブロックのカウンタの値を１つ加算する。また、リード動作制御部２１や最適化処理部５３は、データの読み出し処理を終了した場合、そのブロックのカウンタの値を１つ減算する。最適化処理部５３は、ブロックリユースコマンドの受信時またはブロックリユースコマンドの実行時、そのブロックリユースコマンドで指定されるブロックのカウンタの値が０でない場合、ホスト２に対してエラーを通知し、あるいは、カウンタの値が０になるまでブロックリユースコマンドの実行を保

留し、カウンタの値が０になったら、ブロックリユースコマンドを実行する。
【０１６９】
図４２は、ブロックリユースコマンド受信時におけるフラッシュストレージデバイス３の動作手順（第１ケース）を示すフローチャートである。なお、ここでは、ブロックリユースコマンド受信時を想定するが、以下に説明する動作は、ブロックリユースコマンド実行時に行われるものであってもよい。
【０１７０】
最適化処理部２３は、ブロックリユースコマンドがホスト２から受信された場合（ステップＡ１）、そのブロックリユースコマンドで指定されるブロックを対象とする実行中または実行待ちの読み出し処理が存在するか否かを判定する（ステップＡ２）。存在しない場合（ステップＡ２：ＮＯ）、最適化処理部２３は、指定されたブロックをフリーブロック化（解放）し、リユース完了を示すレスポンスをホスト２に返す（ステップＡ３）。
【０１７１】
一方、存在する場合（ステップＡ２：ＹＥＳ）、最適化処理部２３は、エラーをホスト２に通知する（ステップＡ４）。
図４３は、ブロックリユースコマンド受信時におけるフラッシュストレージデバイス３の動作手順（第２ケース）を示すフローチャートである。ここでも、ブロックリユースコマンド受信時を想定するが、以下に説明する動作は、ブロックリユースコマンド実行時に行われるものであってもよい。
【０１７２】
最適化処理部２３は、ブロックリユースコマンドがホスト２から受信された場合（ステップＡ１１）、そのブロックリユースコマンドで指定されるブロックを対象とする実行中または実行待ちの読み出し処理が存在するか否かを判定する（ステップＡ１２）。
【０１７３】
存在しない場合（ステップＡ１２：ＮＯ）、最適化処理部２３は、即時的に、指定されたブロックをフリーブロック化（解放）し、リユース完了を示すレスポンスをホスト２に返す（ステップＡ１４）。一方、存在する場合（ステップＡ１２：ＹＥＳ）、続いて、最適化処理部２３は、該当する読み出し処理がすべて完了したか否かの判定を行い（ステップＡ１３）、すべて完了したら（ステップＡ１３：ＹＥＳ）、指定されたブロックをフリーブロック化（解放）し、リユース完了を示すレスポンスをホスト２に返す（ステップＡ１４）。
【０１７４】
なお、以上では、ブロックリユースコマンドがホスト２から受信された際、そのブロックリユースコマンドで指定されるブロックを対象とする実行中または実行待ちのリードコマンドやＧＣ制御コマンドを受信済みである場合におけるブロックリユースコマンドの取り扱いを説明した。このフラッシュストレージデバイス３は、さらに、ブロックリユースコマンドがホスト２から受信された後、そのブロックリユースコマンドで指定されるブロックを対象とするリードコマンドやＧＣ制御コマンドがホスト２から受信された場合、エラーをホスト２に返却するようにしてもよい。
【０１７５】
また、図４１を参照して説明したように、フラッシュストレージデバイス３のブロックは、たとえば仮想ストレージデバイス毎に管理されるＱｏＳドメイン間で共有される。つまり、たとえば仮想ストレージデバイス毎にフリーブロック群が管理され、そのフリーブロック群の中から各ＱｏＳドメインへフリーブロックが割り当てられていく。
【０１７６】
ブロックにデータを書き込むケースは、大きく分けて、ホスト２から受信されるライトコマンドに応じて、ホスト２のライトデータバッファ５１に格納されているデータを書き込むケースと、ホスト２から受信されるＧＣ制御コマンドに応じて、フラッシュストレージデバイス３の別のブロックに格納されているデータを書き込むケースとが存在する。ホスト２のライトデータバッファ５１に格納されているデータは新しく、フラッシュストレー

ジデバイス３の別のブロックに格納されているデータは古い。したがって、これらのデータを同一のブロックに混在させると、ライトアンプリフィケーションが悪化するおそれがある。そこで、このフラッシュストレージデバイス３は、データが書き込まれるべきブロックおよびページの双方をストレージデバイスが決定するタイプ＃３－ストレージデバイスとして実現されている場合、ＱｏＳドメイン毎に、ホスト２からのデータを書き込むためのブロックと、フラッシュストレージデバイス３内のデータをコピーするためのブロックとに分離する仕組みを備えてもよい。ブロックがスーパーブロックの単位で取り扱われる場合には、スーパーブロックを、ホスト２からのデータを書き込むためのスーパーブロックと、フラッシュストレージデバイス３内のデータをコピーするためのスーパーブロックとに分離する。つまり、ＱｏＳドメイン毎に、空きページを含むブロックとして、ホスト２からのデータを書き込むためのブロックと、フラッシュストレージデバイス３内のデータをコピーするためのブロックとを各々確保する。

【０１７７】

このブロックの分離は、たとえば、フリーブロック群から選ばれてＱｏＳドメインに割り当てられるブロックそれぞれについて、その用途を示す属性情報をメタデータなどとして保持することによって実現できる。ＱｏＳドメインが利用を開始された時、ホスト２からのデータを書き込むためのブロックと、フラッシュストレージデバイス３内のデータをコピーするためのブロックとのいずれも確保されていない。なお、ブロックが確保されているとは、空きページを含むブロックが割り当てられていることである。

【０１７８】

たとえば、ライト動作制御部２１は、あるＱｏＳドメインに関してホスト２からのデータの書き込みを実行する場合、そのＱｏＳドメインにおいて属性情報がホスト２からのデータを書き込むためのブロックであることを示すブロックが確保されていなければ、フリーブロック群の中の１つのフリーブロックをそのＱｏＳドメイン用に取得して、取得したブロックにデータを書き込む。この取得時、ライト動作制御部２１は、そのブロックがホスト２からのデータを書き込むためのブロックであることを示す属性情報をメタデータとして記録する。一方、属性情報がホスト２からのデータを書き込むためのブロックであることを示す、ホスト２からのデータを書き込むためのブロックが確保されているならば、ライト動作制御部２１は、そのブロック中の最後に書き込みが行われたページに続くページからデータの書き込みを実行する。データの書き込み途中で、そのブロックの最後のページまでデータが書き込まれると、ブロック未確保の状態に戻るので、ライト動作制御部２１は、フリーブロック群の中の１つのフリーブロックをそのＱｏＳドメイン用に取得して、取得したブロックに続きのデータを書き込む。この取得時にも、ライト動作制御部２１は、そのブロックがホスト２からのデータを書き込むためのブロックであることを示す属性情報をメタデータとして記録する。

【０１７９】

また、たとえば、（ＧＣ動作制御部２３を含む）最適化処理部５３は、あるＱｏＳドメインに関してデータのコピーを実行する場合、そのＱｏＳドメインにおいて属性情報がフラッシュストレージデバイス３内のデータをコピーするためのブロックが確保されていなければ、フリーブロック群の中の１つのフリーブロックをそのＱｏＳドメイン用に取得して、取得したブロックにデータを書き込む（コピーする）。この取得時、最適化処理部５３は、そのブロックがフラッシュストレージデバイス３内のデータをコピーするためのブロックであることを示す属性情報をメタデータとして記録する。一方、属性情報がフラッシュストレージデバイス３内のデータをコピーするためのブロックであることを示す、フラッシュストレージデバイス３内のデータをコピーするためのブロックが確保されているならば、最適化処理部５３は、そのブロック中の最後に書き込みが行われたページに続くページからデータの書き込みを実行する。データの書き込み途中で、そのブロックの最後のページまでデータが書き込まれると、ブロック未確保の状態に戻るので、最適化処理部５３は、フリーブロック群の中の１つのフリーブロックをそのＱｏＳドメイン用に取得して、取得したブロックにデータを書き込む。この取得時にも、最適化処理部５３は、そのブ

ロックがフラッシュストレージデバイス３内のデータをコピーするためのブロックであることを示す属性情報をメタデータとして記録する。

【０１８０】

このように、このフラッシュストレージデバイス３は、ホスト２からの新しいデータを書き込むためのブロックと、フラッシュストレージデバイス３内の古いデータをコピーするためのブロックとを分離することによって、ライトアンプリフィケーションの悪化を防ぐことができる。

【０１８１】

また、図３９を参照して説明したように、リード動作制御部２２は、読み出すべきリードデータがホスト２のライトデータバッファ５１内に存在する場合、ライトデータバッファ５１からリードデータを取得することができる。一方、ライトデータバッファ５１上のライトデータが格納される領域は、ライト動作制御部２１からホスト２へ解放可能通知が送信されると、ホスト２側において解放される。そこで、このフラッシュストレージデバイス３は、ライトデータバッファ５１内に存在するライトデータ中のデータが対象となるリードコマンドがホスト２から受信された場合、そのリードコマンドが終了するまで、そのデータが格納される領域についての解放可能通知をホスト２へ送信しないようにする仕組みを備えてもよい。

【０１８２】

この仕組みは、たとえば、ホスト２のライトデータバッファ５１に格納されるライトデータに対し、ホスト２から受信されるライトコマンド単位またはホスト２のデータ使用単位で、データの書き込み処理数および読み出し処理数の残数を示すカウンタをメタデータなどとして設けることによって実現できる。カウンタは、たとえば、解放可能通知がホスト２に通知される単位と一致させて設けられる。解放可能通知がライトコマンド単位でホスト２に通知される場合において、カウンタがホスト２のデータ使用単位で設けられてもよい。

【０１８３】

カウンタがホスト２のデータ使用単位で設けられるものと想定すると、ライト動作制御部２１は、各カウンタの初期値として、ＮＡＮＤ型フラッシュメモリ５へのデータの書き込みに必要なデータの転送回数＋１をセットする。＋１は、エラーが検出された場合の再書き込み処理のために加算しておくものである。

【０１８４】

ライト動作制御部２１は、ＮＡＮＤ型フラッシュメモリ５へデータを転送する都度、対応するカウンタの値を１ずつ減算する。あるデータ使用単位についてデータの転送が終了したとすると、その時点で、一般的には、カウンタの値は１となる。ライト動作制御部２１は、転送したデータすべてがＮＡＮＤ型フラッシュメモリ５に書き込まれ、エラーが検出された場合の再書き込み処理が不要となったことが確定すると、対応するカウンタの値をさらに１減算する。この時点で、一般的には、カウンタの値は０となる。仮に、解放可能通知がホスト２のデータ使用単位でホスト２に通知されるものと想定すると、ライト動作制御部２１は、カウンタの値が０になったことを検知した場合、対応する領域についての解放可能通知をホスト２へ通知する。なお、エラーが検出された場合、ライト動作制御部２１は、再書き込み処理に必要なデータの転送回数をカウンタに再加算する。ＮＡＮＤ型フラッシュメモリ５にデータを転送し終えた後にエラーが検出されても、カウンタの値が０になっていないことから、解放可能通知はホスト２へ通知されておらず、再書き込み処理に必要なデータはホスト２のライトデータバッファ５１に存在する。したがって、ホスト２のライトデータバッファ５１からフラッシュストレージデバイス３の内部バッファ（共有キャッシュ）３１へのデータ転送を再実行することができる。

【０１８５】

リード動作制御部２１も、ホスト２のライトデータバッファ５１に存在するライトデータ中のデータが対象となるリードコマンドがホスト２から受信された場合、そのデータに対応するカウンタの値を１加算する。そして、その読み出し処理が終了したら、リード動作

制御部２１は、対応するカウンタの値を１減算する。
【０１８６】
ライトデータバッファ５１上に存在するライトデータ中の読み出し処理の対象となるデータについては、ＮＡＮＤ型フラッシュメモリ５への書き込みが終了しても、対応するカウンタの値は０にはならず、解放可能通知がホスト２へ通知されない。つまり、カウンタの値を１加算することで、リード動作制御部２１は、ライトデータバッファ５１上の目的の領域を解放禁止状態とする。したがって、ホスト２のライトデータバッファ５１に存在するライトデータ中のデータが対象となるリードコマンドがホスト２から受信されている状況下において、そのデータを含むライトデータバッファ５１上の領域がホスト２側において解放されてしまうことがない。
【０１８７】
なお、ホスト２のライトデータバッファ５１に存在するライトデータ中のデータが対象となるリードコマンドがホスト２から受信された場合、リード動作制御部２１は、そのデータを必ずしもライトデータバッファ５１から読み出さなくてもよく、ライトデータのＮＡＮＤ型フラッシュメモリ５への書き込みが終了し、読み出し可能な状態になっているならば、ＮＡＮＤ型フラッシュメモリ５から読み出してもよい。この場合、ライトデータバッファ５１上のライトデータは、たとえば、予備のデータなどとして活用し得る。
【０１８８】
解放可能通知がライトコマンド単位でホスト２に通知され、カウンタがホスト２のデータ使用単位で設けられる場合、ライト動作制御部２１は、ライトコマンドで書き込まれるべきライトデータに対応する複数のカウンタすべての値が０になった時点で、解放可能通知をホスト２に通知する。
【０１８９】
また、リード動作制御部２２が、ライトデータバッファ５１からリードデータを取得することができる点に着目し、このフラッシュストレージデバイス３は、タイプ＃２－ストレージデバイスとして実現されている場合、ホスト２からのライトデータを書き込み予定のページアドレスを、ＮＡＮＤ型フラッシュメモリ５への書き込みの終了を待機することなく、ホスト２に通知する仕組みを備えてもよい。この仕組みをフラッシュストレージデバイス３が備える場合、ホスト２は、たとえば、ライトコマンドで書き込んだデータがフラッシュストレージデバイス３において読み出し可能な状態となるまで待たされることなどがなく、ライトコマンドで書き込んだデータ中のデータを対象とするリードコマンドを速やかに発行することが可能となる。
【０１９０】
この仕組みは、たとえば、ライト動作制御部２１が、ライトコマンドの受信時にホスト２から通知されるライトデータバッファ５１上のライトデータに関する情報を、たとえばメタデータなどとして設ける書き込み先ブロック毎のライトバッファリストに登録し、ホスト２のデータ使用単位毎に、ライトデータを書き込み予定のページアドレスをホスト２に通知することによって実現できる。ライトバッファリストに登録されたライトデータのサイズが、書き込み先ブロックの残り書き込み領域のサイズより大きくても構わない。この場合、ライト動作制御部２１は、まず、その書き込み先ブロックに書き込み可能な分について書き込み予定のページアドレスをホスト２に通知し、その書き込み先ブロックへの書き込みが終了して、新たな書き込み先ブロックが確保された後、残りの分について書き込み予定のページアドレスをホスト２に通知する。ライトデータはすべてライトデータバッファ５１上に存在し、書き込み先ブロックの確保は極めて短時間で済むので、ライトデータの書き込みがブロックを跨ぐ場合であっても実用上の問題はない。
【０１９１】
ＮＡＮＤ型フラッシュメモリ５への書き込み時にエラーが検出された場合、ライト動作制御部２１は、新たに決定される書き込み予定のページアドレスをホスト２に改めて通知する。なお、ページアドレスのホスト２への通知は、前述したように、書き込み予定のページアドレスが決定した時点で行ってもよいし、ホスト２のデータ使用単位で書き込みが終

了する毎に行ってもよい。前者の場合、エラーが検出された場合にホスト２への通知が複数回発生し得るが、ホスト２への通知が速い。後者の場合、ホスト２への通知が前者の場合よりも遅くはなるが、ＮＡＮＤ型フラッシュメモリ５への書き込み時にエラーが検出された場合であっても、その回数に関係なく、１度の通知で済むことになる。

【０１９２】

次に、ホスト２のライトデータバッファ５１を用いるライトコマンド処理を含む、このフラッシュストレージデバイス３によって実行される各種Ｉ／Ｏコマンド処理について詳細に説明する。

図４４は、フラッシュストレージデバイス３によって実行されるＩ／Ｏコマンド処理を示す。

【０１９３】

上述したように、本実施形態では、フラッシュストレージデバイス３はタイプ＃１－ストレージデバイス、タイプ＃２－ストレージデバイス、タイプ＃３－ストレージデバイスのいずれであってもよいが、図４４では、フラッシュストレージデバイス３がタイプ＃１－ストレージデバイスである場合を例示する。

【０１９４】

ホスト２によって発行される各ライトコマンドは、ブロックアドレス、ページアドレス、データポインタ、長さを含む。発行された各ライトコマンドはＩ／Ｏコマンドキュー４２に入れられる。ホスト２によって発行される各リードコマンドも、ブロックアドレス、ページアドレス、データポインタ、長さを含む。発行された各リードコマンドもＩ／Ｏコマンドキュー４２に入れられる。

【０１９５】

ホスト２がライトデータの書き込みをフラッシュストレージデバイス３に要求することを望む場合、ホスト２は、まず、このライトデータをホストメモリ上のライトデータバッファ５１に格納し、そして、ライトコマンドをフラッシュストレージデバイス３に発行する。このライトコマンドは、このライトデータが書き込まれるべき書き込み先ブロックを示すブロックアドレスと、このライトデータが書き込まれるべきこの書き込み先ブロック内のページを示すページアドレスと、このライトデータが存在するライトデータバッファ５１内の位置を示すデータポインタと、このライトデータの長さとを含む。

【０１９６】

フラッシュストレージデバイス３は、プログラム／リードシーケンサ４１を含む。このプログラム／リードシーケンサ４１は、上述のライト制御部２１およびリード制御部２２によって実現される。プログラム／リードシーケンサ４１は、Ｉ／Ｏコマンドキュー４２に入れられたコマンドそれぞれを任意の順序で実行することができる。

【０１９７】

プログラム／リードシーケンサ４１が、ある同じ書き込み先ブロックを指定する１以上のライトコマンドをＩ／Ｏコマンドキュー４２から取得した後、プログラム／リードシーケンサ４１は、この書き込み先ブロックの書き込み動作の進行に合わせて、この書き込み先ブロックに書き込むべき次のライトデータ（例えば、１ページサイズ分のライトデータ）を内部バッファ（共有キャッシュ）３１またはライトデータバッファ５１から取得するための転送要求を内部バッファ（共有キャッシュ）３１に送出する。この転送要求は、データポインタと長さとを含んでいてもよい。この転送要求に含まれるデータポインタは、１つのライトコマンドに関連付けられたライトデータを分割、または同じ書き込み先ブロックを指定する２以上のライトコマンドに関連付けられた２以上のライトデータを結合する処理によって算出される。つまり、プログラム／リードシーケンサ４１は、同じ書き込み先ブロックを示す識別子を有する一つ以上のライトコマンドに関連付けられたライトデータの集合を、その先頭からＮＡＮＤ型フラッシュメモリ５のデータ書き込み単位と同じサイズを有する境界で区切り、各境界に対応するホストメモリ内の位置を特定する。これによって、プログラム／リードシーケンサ４１は、書き込み単位と同じサイズの単位でライトデータをホスト２から取得することができる。

【０１９８】
この転送要求に含まれるデータポインタは、この１ページサイズ分のライトデータが存在するライトデータバッファ５１上の位置を示す。この１ページサイズ分のライトデータは、この書き込み先ブロックを指定する複数のライトコマンドに関連付けられた複数の小さなサイズのライトデータの集合であってもよいし、この書き込み先ブロックを指定する一つのライトコマンドに関連付けられた大きなサイズのライトデータの一部分であってもよい。
【０１９９】
さらに、プログラム／リードシーケンサ４１は、この１ページサイズ分のライトデータが書き込まれるべき書き込み先ブロックのブロックアドレスと、この１ページサイズ分のライトデータが書き込まれるべきページのページアドレスとを内部バッファ（共有キャッシュ）３１に送出する。
【０２００】
フラッシュストレージデバイス３のコントローラ４は内部バッファ（共有キャッシュ）３１を制御するキャッシュコントローラを含んでいてもよい。この場合、このキャッシュコントローラは内部バッファ（共有キャッシュ）３１をあたかも制御ロジックであるかのように動作させることができる。内部バッファ（共有キャッシュ）３１と複数の書き込み先ブロック＃０、＃１、＃２、…、＃ｎとの間には、複数のフラッシュコマンドキュー４３が存在する。これらフラッシュコマンドキュー４３は、複数のＮＡＮＤ型フラッシュメモリチップにそれぞれ対応づけられている。
【０２０１】
内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、転送要求によって指定されたこの１ページサイズ分のライトデータが内部バッファ（共有キャッシュ）３１に存在するか否かを判定する。
この転送要求によって指定されたこの１ページサイズ分のライトデータが内部バッファ（共有キャッシュ）３１に存在するならば、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、この１ページサイズ分のライトデータを、このライトデータが書き込まれるべき書き込み先ブロックを含むＮＡＮＤ型フラッシュメモリチップに転送する。さらに、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、このライトデータが書き込まれるべき書き込み先ブロックを含むＮＡＮＤ型フラッシュメモリチップに、この書き込み先ブロックのブロックアドレス、このライトデータが書き込まれるべきページアドレス、書き込み指示用のＮＡＮＤコマンド（フラッシュライトコマンド）を、フラッシュコマンドキュー４３を介して送出する。フラッシュコマンドキュー４３は、ＮＡＮＤ型フラッシュメモリチップ毎に設けられている。このため、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、このライトデータが書き込まれるべき書き込み先ブロックを含むＮＡＮＤ型フラッシュメモリチップに対応するフラッシュコマンドキュー４３に、この書き込み先ブロックのブロックアドレス、このライトデータが書き込まれるべきページアドレス、書き込み指示用のＮＡＮＤコマンド（フラッシュライトコマンド）を入れる。
【０２０２】
なお、内部バッファ（共有キャッシュ）３１からＮＡＮＤ型フラッシュメモリチップへのこの１ページサイズ分のライトデータの転送が、このライトデータをＮＡＮＤ型フラッシュメモリチップに書き込むために必要な最終回のデータ転送であるならば、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、このライトデータを内部バッファ（共有キャッシュ）３１から破棄し、このライトデータが格納されていた領域を空き領域として確保する。ＮＡＮＤ型フラッシュメモリチップにデータを一回転送することを伴う書き込み動作（例えば、フル・シーケンス書き込み動作、等）によってライトデータを書き込み先ブロックに書き込むケースにおいては、ＮＡＮＤ型フラッシュメモリチップへの初回のデータ転送が最終回のデータ転送となる。一方、ＮＡＮＤ型フラッシュメモリチップにデータを複数回転送することを伴う書き込み動作（例えば、フォギー・ファイ

ン書き込み動作）によってライトデータを書き込み先ブロックに書き込むケースにおいては、最後のファイン書き込みのために必要なＮＡＮＤ型フラッシュメモリチップへのデータ転送が最終回のデータ転送となる。

【０２０３】

次に、転送要求によって指定されたこの１ページサイズ分のライトデータが内部バッファ（共有キャッシュ）３１に存在しない場合について説明する。

この転送要求によって指定されたこの１ページサイズ分のライトデータが内部バッファ（共有キャッシュ）３１に存在しないならば、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、この転送要求（データポインタ、長さ）をＤＭＡＣ１５に送出する。この転送要求（データポインタ、長さ）に基づいて、ＤＭＡＣ１５は、この１ページサイズ分のライトデータをホストメモリ上のライトデータバッファ５１から内部バッファ（共有キャッシュ）３１に転送する。このデータ転送が終了すると、ＤＭＡＣ１５は、転送完了（Ｄｏｎｅ）と、このデータポインタ、この長さとを、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラに通知する。

【０２０４】

内部バッファ（共有キャッシュ）３１に空き領域が存在するならば、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、ＤＭＡ転送によってライトデータバッファ５１から取得されたライトデータを、この空き領域に格納する。

内部バッファ（共有キャッシュ）３１に空き領域が存在しないならば、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、内部バッファ（共有キャッシュ）３１内の最も古いライトデータを内部バッファ（共有キャッシュ）３１から破棄し、最も古いライトデータが格納されていた領域を空き領域として確保する。そして、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、ＤＭＡ転送によってライトデータバッファ５１から取得されたライトデータを、この空き領域に格納する。

【０２０５】

フォギー・ファイン書き込み動作のような多段階の書き込み動作が使用されるケースにおいては、キャッシュコントローラは、フォギー書き込み動作のような第１段階の書き込み動作が終了している内部バッファ（共有キャッシュ）３１内のライトデータのうちで、最も古いライトデータを破棄する。

【０２０６】

データ書き込み量の少ない書き込み先ブロックへのデータ書き込み動作の進行速度に比べ、データ書き込み量の多い書き込み先ブロックへのデータ書き込み動作の進行速度は速くなる傾向がある。このため、データ書き込み量の多い書き込み先ブロックに書き込まれるべきライトデータは頻繁にライトデータバッファ５１から内部バッファ（共有キャッシュ）３１に転送される。この結果、この最も古いライトデータは、ホスト２から書き込まれるデータ量が比較的少ない書き込み先ブロックへのライトデータである可能性が高い。したがって、フォギー書き込み動作のような第１段階の書き込み動作が終了している内部バッファ（共有キャッシュ）３１内のライトデータのうちで最も古いライトデータを破棄するという方法を使用することにより、ホスト２とフラッシュストレージデバイス３との間のデータトラフィックを効率よく低減することが可能となる。

【０２０７】

なお、フォギー書き込み動作のような第１段階の書き込み動作が終了している内部バッファ（共有キャッシュ）３１内のライトデータの中から破棄すべきライトデータを選択するためのアルゴリズムは、最も古いデータを選択するファースト・イン・ファースト・アウトに限定されず、ＬＲＵ、ランダムのような他のアルゴリズムを使用してもよい。

【０２０８】

プログラム／リードシーケンサ４１は、各ＮＡＮＤ型フラッシュメモリチップからステータス、つまり、書き込み完了（Ｄｏｎｅ）、書き込み失敗（Ｅｒｒｏｒ）、ブロックアドレス、ページアドレス、を受信する。そして、これらステータスに基づいて、プログラム／リードシーケンサ４１は、ライトコマンド毎に、このライトコマンドに関連付けられた

ライトデータ全体に対する書き込み動作（ＮＡＮＤ型フラッシュメモリチップに同じデータを１回または複数回転送する書き込み動作）の全てが終了したか否かを判定する。あるライトコマンドに関連付けられたライトデータ全体に対する書き込み動作の全てが終了したならば、プログラム／リードシーケンサ４１は、このライトコマンドのコマンド完了を示すレスポンス（Ｄｏｎｅ）をホスト２に送信する。このコマンド完了を示すレスポンス（Ｄｏｎｅ）は、このライトコマンドを一意に識別するコマンドＩＤを含む。

【０２０９】

次に、リードコマンドの処理について説明する。

リードコマンドは、リードすべきデータが格納されているブロックを示すブロックアドレスと、このデータが格納されているページを示すページアドレスと、このデータが転送されるべきホストメモリ上のリードデータバッファ５３内の位置を示すデータポインタと、このデータの長さとを含む。

【０２１０】

プログラム／リードシーケンサ４１は、リードコマンドによって指定されたブロックアドレスおよびページアドレスを内部バッファ（共有キャッシュ）３１に送出し、リードコマンドによって指定されたデータの読み出しを内部バッファ（共有キャッシュ）３１に要求する。

【０２１１】

内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、ＮＡＮＤ型フラッシュメモリチップに、このブロックアドレスと、このページアドレスと、リード指示用のＮＡＮＤコマンド（フラッシュリードコマンド）を、フラッシュコマンドキュー４３を介して送出する。ＮＡＮＤ型フラッシュメモリチップから読み出されたデータは、ＤＭＡＣ１５によってリードデータバッファ５３に転送される。

【０２１２】

なお、リードコマンドによって指定されたデータが、書き込み動作が終了していないデータ、または書き込み動作の全てが終了しているがＮＡＮＤ型フラッシュメモリ５からまだ読み出し可能となっていないデータである場合、内部バッファ（共有キャッシュ）３１、つまりキャッシュコントローラは、内部バッファ（共有キャッシュ）３１にこのデータが存在するか否かを判定してもよい。内部バッファ（共有キャッシュ）３１にこのデータが存在するならば、このデータが内部バッファ（共有キャッシュ）３１から読み出され、そしてＤＭＡＣ１５によってリードデータバッファ５３に転送される。

【０２１３】

一方、内部バッファ（共有キャッシュ）３１にこのデータが存在しないならば、このデータは、まず、ＤＭＡＣ１５によってライトデータバッファ５１から内部バッファ（共有キャッシュ）３１に転送される。そして、このデータが内部バッファ（共有キャッシュ）３１から読み出され、そしてＤＭＡＣ１５によってリードデータバッファ５３に転送される。

【０２１４】

図４５は、フラッシュストレージデバイス３によって実行される複数段階の書き込み動作を示す。

ここでは、４つのワード線を往復する場合のフォギー・ファイン書き込み動作を例示する。また、ここでは、ＮＡＮＤ型フラッシュメモリ５が、メモリセル当たりに４ビットのデータを格納するＱＬＣ－フラッシュである場合を想定する。ＮＡＮＤ型フラッシュメモリ５内の一つの特定の書き込み先ブロック（ここでは、書き込み先ブロックＢＬＫ＃１）に対するフォギー・ファイン書き込み動作は以下のように実行される。

【０２１５】

（１）まず、４ページ（Ｐ０～Ｐ３）分のライトデータがページ単位でＮＡＮＤ型フラッシュメモリ５に転送され、この書き込み先ブロックＢＬＫ＃１内のワード線ＷＬ０に接続された複数のメモリセルに、これら４ページ（Ｐ０～Ｐ３）分のライトデータを書き込むためのフォギー書き込み動作が実行される。

【０２１６】

（２）次いで、次の４ページ（Ｐ４～Ｐ７）分のライトデータがこのＮＡＮＤ型フラッシュメモリ５にページ単位で転送され、この書き込み先ブロックＢＬＫ＃１内のワード線ＷＬ１に接続された複数のメモリセルに、これら４ページ（Ｐ４～Ｐ７）分のライトデータを書き込むためのフォギー書き込み動作が実行される。

【０２１７】

（３）次いで、次の４ページ（Ｐ８～Ｐ１１）分のライトデータがこのＮＡＮＤ型フラッシュメモリ５にページ単位で転送され、この書き込み先ブロックＢＬＫ＃１内のワード線ＷＬ２に接続された複数のメモリセルに、これら４ページ（Ｐ８～Ｐ１１）分のライトデータを書き込むためのフォギー書き込み動作が実行される。

【０２１８】

（４）次いで、次の４ページ（Ｐ１２～Ｐ１５）分のライトデータがこのＮＡＮＤ型フラッシュメモリ５にページ単位で転送され、この書き込み先ブロックＢＬＫ＃１内のワード線ＷＬ３に接続された複数のメモリセルに、これら４ページ（Ｐ１２～Ｐ１５）分のライトデータを書き込むためのフォギー書き込み動作が実行される。

【０２１９】

（５）ワード線ＷＬ３に接続された複数のメモリセルに対するフォギー書き込み動作が終了すると、書き込み対象のワード線はワード線ＷＬ０に戻り、ワード線ＷＬ０に接続された複数のメモリセルに対するファイン書き込み動作の実行が可能となる。そして、ワード線ＷＬ０に対するフォギー書き込み動作で使用された４ページ（Ｐ０～Ｐ３）分のライトデータと同じ４ページ（Ｐ０～Ｐ３）分のライトデータがページ単位でＮＡＮＤ型フラッシュメモリ５に再び転送され、この書き込み先ブロックＢＬＫ＃１内のワード線ＷＬ０に接続された複数のメモリセルに、これら４ページ（Ｐ０～Ｐ３）分のライトデータを書き込むためのファイン書き込み動作が実行される。これにより、ページＰ０～Ｐ３に対するフォギー・ファイン書き込み動作が終了する。

【０２２０】

（６）次いで、次の４ページ（Ｐ１６～Ｐ１９）分のライトデータがこのＮＡＮＤ型フラッシュメモリ５にページ単位で転送され、この書き込み先ブロックＢＬＫ＃１内のワード線ＷＬ４に接続された複数のメモリセルに、これら４ページ（Ｐ１６～Ｐ１９）分のライトデータを書き込むためのフォギー書き込み動作が実行される。

【０２２１】

（７）ワード線ＷＬ４に接続された複数のメモリセルに対するフォギー書き込み動作が終了すると、書き込み対象のワード線はワード線ＷＬ１に戻り、ワード線ＷＬ１に接続された複数のメモリセルに対するファイン書き込み動作の実行が可能となる。そして、ワード線ＷＬ１に対するフォギー書き込み動作で使用された４ページ（Ｐ４～Ｐ７）分のライトデータと同じ４ページ（Ｐ４～Ｐ７）分のライトデータがページ単位でＮＡＮＤ型フラッシュメモリ５に再び転送され、この書き込み先ブロックＢＬＫ＃１内のワード線ＷＬ１に接続された複数のメモリセルに、これら４ページ（Ｐ４～Ｐ７）分のライトデータを書き込むためのファイン書き込み動作が実行される。これにより、ページＰ４～Ｐ７に対するフォギー・ファイン書き込み動作が終了する。

【０２２２】

（８）次いで、次の４ページ（Ｐ２０～Ｐ２３）分のライトデータがこのＮＡＮＤ型フラッシュメモリ５にページ単位で転送され、この書き込み先ブロックＢＬＫ＃１内のワード線ＷＬ５に接続された複数のメモリセルに、これら４ページ（Ｐ２０～Ｐ２３）分のライトデータを書き込むためのフォギー書き込み動作が実行される。

【０２２３】

（９）ワード線ＷＬ５に接続された複数のメモリセルに対するフォギー書き込み動作が終了すると、書き込み対象のワード線はワード線ＷＬ２に戻り、ワード線ＷＬ２に接続された複数のメモリセルに対するファイン書き込み動作の実行が可能となる。そして、ワード線ＷＬ２に対するフォギー書き込み動作で使用された４ページ（Ｐ８～Ｐ１１）分のライトデータと同じ４ページ（Ｐ８～Ｐ１１）分のライトデータがページ単位でＮＡＮＤ型フ

ラッシュメモリ５に再び転送され、この書き込み先ブロックＢＬＫ＃１内のワード線ＷＬ２に接続された複数のメモリセルに、これら４ページ（Ｐ８～Ｐ１１）分のライトデータを書き込むためのファイン書き込み動作が実行される。これにより、ページＰ８～Ｐ１１に対するフォギー・ファイン書き込み動作が終了する。

【０２２４】

図４６は、書き込み先ブロックＢＬＫ＃１へのデータの書き込み順序を示す。

ここでは、図７と同様に、４つのワード線を往復する場合のフォギー・ファイン書き込み動作が実行される場合を想定する。

図４６の左部に示されるデータｄ０、データｄ１、データｄ２、データｄ３、データｄ４、データｄ５、データｄ６、データｄ７、…、データｄ２５２、データｄ２５３、データｄ２５４、データｄ２５５は、書き込み先ブロックＢＬＫ＃１を指定する複数のライトコマンドそれぞれに対応する複数のライトデータを示している。ここでは、図示の簡単化のために、全てのライトデータが同じサイズを有している場合が想定されている。

【０２２５】

図４６の右部は、書き込み先ブロックＢＬＫ＃１へのデータの書き込み順序を示している。書き込み動作は、ワード線ＷＬ０に接続された複数のメモリセルへのデータｄ０の書き込み（フォギー書き込み）、ワード線ＷＬ１に接続された複数のメモリセルへのデータｄ１の書き込み（フォギー書き込み）、ワード線ＷＬ２に接続された複数のメモリセルへのデータｄ２の書き込み（フォギー書き込み）、ワード線ＷＬ３に接続された複数のメモリセルへのデータｄ３の書き込み（フォギー書き込み）、ワード線ＷＬ０に接続された複数のメモリセルへのデータｄ０の書き込み（ファイン書き込み）、ワード線ＷＬ４に接続された複数のメモリセルへのデータｄ４の書き込み（フォギー書き込み）、ワード線ＷＬ１に接続された複数のメモリセルへのデータｄ１の書き込み（ファイン書き込み）、ワード線ＷＬ５に接続された複数のメモリセルへのデータｄ５の書き込み（フォギー書き込み）、ワード線ＷＬ２に接続された複数のメモリセルへのデータｄ２の書き込み（ファイン書き込み）、…という順序で実行される。

【０２２６】

図４７は、ＮＡＮＤ型フラッシュメモリ５のデータ書き込み単位と同じサイズの単位でライトデータをホスト２からフラッシュストレージデバイス３に転送する動作を示す。

図４７の左部に示されるデータｄ１、データｄ２、データｄ３、データｄ４、データｄ５、データｄ６、データｄ７、データｄ８、データｄ９、データｄ１０、…は、書き込み先ブロックＢＬＫ＃１を指定する１０個のライトコマンドにそれぞれ対応する１０個のライトデータを示している。ライトデータの長さ（サイズ）は、個々のライトコマンド毎に異なる。図４７では、データｄ１、データｄ２、データｄ３、データｄ４の各々が４Ｋバイトのサイズを有し、データｄ５が８Ｋバイトのサイズを有し、データｄ６が４０Ｋバイトのサイズを有し、データｄ７が１６Ｋバイトのサイズを有し、データｄ８、データｄ９の各々が８Ｋバイトのサイズを有し、データｄ１０が１Ｍバイトのサイズを有する場合が想定されている。

【０２２７】

ホスト２から受信される各ライトコマンドはデータポインタ、長さ、ブロック識別子（例えばブロックアドレス）を含むので、フラッシュストレージデバイス３のコントローラ４は、ホスト２から受信されるライトコマンドを、複数の書き込み先ブロックにそれぞれ対応する複数のグループに分類することができる。上述のデータｄ１、データｄ２、データｄ３、データｄ４、データｄ５、データｄ６、データｄ７、データｄ８、データｄ９、データｄ１０、…は、書き込み先ブロックＢＬＫ＃１に対応するグループに分類された１０個のライトコマンドにそれぞれ対応している。これら１０個のライトコマンドは、書き込み先ブロックＢＬＫ＃１を示すブロック識別子（例えばブロックアドレス）を含むライトコマンドである。

【０２２８】

フラッシュストレージデバイス３のコントローラ４は、書き込み先ブロックＢＬＫ＃１を

指定するこれらライトコマンド内のデータポインタおよび長さに基づいて、データｄ１、データｄ２、データｄ３、データｄ４、データｄ５、データｄ６、データｄ７、データｄ８、データｄ９、データｄ１０がそれぞれ存在するライトデータバッファ５１上の位置、およびデータｄ１、データｄ２、データｄ３、データｄ４、データｄ５、データｄ６、データｄ７、データｄ８、データｄ９、データｄ１０それぞれの長さを管理する。そして、コントローラ４は、一つのライトコマンドに関連付けられた大きなサイズのライトデータを複数のライトデータ（複数のデータ部）に分割、または２以上のライトコマンドにそれぞれ関連付けられた小さなサイズの２以上のライトデータを互いに結合することによって得られる、ＮＡＮＤ型フラッシュメモリ５のデータ書き込み単位と同じサイズを有するライトデータを、ホスト２から取得する。

【０２２９】

図４７では、コントローラ４は、最初に、各々が４Ｋバイトのサイズを有するデータｄ１、データｄ２、データｄ３、データｄ４を互いに結合することによって得られる１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から取得する。この場合、コントローラ４は、これに限定されないが、例えば、４回のＤＭＡ転送によって、この１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から内部バッファ３１に転送してもよい。最初のＤＭＡ転送では、データｄ１の先頭位置を指定する転送元アドレスと、データ長＝４ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ１の先頭位置を指定する転送元アドレスは、データｄ１に対応するライトコマンド内のデータポインタによって表される。２回目のＤＭＡ転送では、データｄ２の先頭位置を指定する転送元アドレスと、データ長＝４ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ２の先頭位置を指定する転送元アドレスは、データｄ２に対応するライトコマンド内のデータポインタによって表される。３回目のＤＭＡ転送では、データｄ３の先頭位置を指定する転送元アドレスと、データ長＝４ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ３の先頭位置を指定する転送元アドレスは、データｄ３に対応するライトコマンド内のデータポインタによって表される。４回目のＤＭＡ転送では、データｄ４の先頭位置を指定する転送元アドレスと、データ長＝４ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ４の先頭位置を指定する転送元アドレスは、データｄ４に対応するライトコマンド内のデータポインタによって表される。

【０２３０】

そして、コントローラ４は、ＤＭＡ転送によって取得されるこの１６Ｋバイトライトデータ（ｄ１、ｄ２、ｄ３、ｄ４）を、書き込み先ブロックＢＬＫ＃１のページＰ０に書き込まれるべきデータとしてＮＡＮＤ型フラッシュメモリ５に転送する。

コントローラ４は、書き込み先ブロックＢＬＫ＃１の次の書き込み先ページをページＰ１に変更し、８Ｋバイトのサイズを有するデータｄ３と、データｄ６内の先頭の８Ｋバイトデータｄ６－１とを互いに結合することによって得られる１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から取得する。この場合、コントローラ４は、これに限定されないが、例えば、２回のＤＭＡ転送によって、この１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から内部バッファ３１に転送してもよい。最初のＤＭＡ転送では、データｄ５の先頭位置を指定する転送元アドレスと、データ長＝８ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ５の先頭位置を指定する転送元アドレスは、データｄ５に対応するライトコマンド内のデータポインタによって表される。２回目のＤＭＡ転送では、データｄ６－１の先頭位置を指定する転送元アドレスと、データ長＝８ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ６－１の先頭位置を指定する転送元アドレスは、データｄ６に対応するライトコマンド内のデータポインタによって表される。

【０２３１】

そして、コントローラ４は、この１６Ｋバイトライトデータ（ｄ５、ｄ６－１）を、書き込み先ブロックＢＬＫ＃１のページＰ１に書き込まれるべきデータとしてＮＡＮＤ型フラッシュメモリ５に転送する。

コントローラ４は、書き込み先ブロックＢＬＫ＃１の次の書き込み先ページをページＰ２

に変更し、データｄ６の残りの３２Ｋバイトデータのうちの最初の１６Ｋバイトデータｄ６－２をホスト２のライトデータバッファ５１から取得する。この場合、コントローラ４は、これに限定されないが、例えば、１回のＤＭＡ転送によって、この１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から内部バッファ３１に転送してもよい。このＤＭＡ転送では、データｄ６－２の先頭位置を指定する転送元アドレスと、データ長＝１６ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ６－２の先頭位置を指定する転送元アドレスは、データｄ６に対応するライトコマンド内のデータポインタの値に８ＫＢ分のオフセットを加算することによって求めることができる。

【０２３２】

そして、コントローラ４は、この１６Ｋバイトライトデータ（ｄ６－２）を、書き込み先ブロックＢＬＫ＃１のページＰ２に書き込まれるべきデータとしてＮＡＮＤ型フラッシュメモリ５に転送する。

コントローラ４は、書き込み先ブロックＢＬＫ＃１の次の書き込み先ページをページＰ３に変更し、データｄ６の残りの１６Ｋバイトデータｄ６－３をホスト２のライトデータバッファ５１から取得する。この場合、コントローラ４は、これに限定されないが、例えば、１回のＤＭＡ転送によって、この１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から内部バッファ３１に転送してもよい。このＤＭＡ転送では、データｄ６－３の先頭位置を指定する転送元アドレスと、データ長＝１６ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ６－３の先頭位置を指定する転送元アドレスは、データｄ６に対応するライトコマンド内のデータポインタの値に２４ＫＢ分のオフセットを加算することによって求めることができる。

【０２３３】

そして、コントローラ４は、この１６Ｋバイトライトデータ（ｄ６－３）を、書き込み先ブロックＢＬＫ＃１のページＰ３に書き込まれるべきデータとしてＮＡＮＤ型フラッシュメモリ５に転送する。

そして、コントローラ４は、フォギー書き込み動作によって４ページ分のデータ（Ｐ０～Ｐ３）を書き込み先ブロックＢＬＫ＃１のワード線ＷＬ０に接続された複数のメモリセルに書き込む。

【０２３４】

コントローラ４は、書き込み先ブロックＢＬＫ＃１の次の書き込み先ページをページＰ４に変更し、１６Ｋバイトのサイズを有するデータｄ７をホスト２のライトデータバッファ５１から取得する。この場合、コントローラ４は、これに限定されないが、例えば、１回のＤＭＡ転送によって、この１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から内部バッファ３１に転送してもよい。このＤＭＡ転送では、データｄ７の先頭位置を指定する転送元アドレスと、データ長＝１６ＫＢがＤＭＡＣ１５にセットされてもよい。データ７の先頭位置を指定する転送元アドレスは、データｄ７に対応するライトコマンド内のデータポインタによって表される。

【０２３５】

そして、コントローラ４は、この１６Ｋバイトライトデータ（ｄ７）を、書き込み先ブロックＢＬＫ＃１のページＰ４に書き込まれるべきデータとしてＮＡＮＤ型フラッシュメモリ５に転送する。

コントローラ４は、書き込み先ブロックＢＬＫ＃１の次の書き込み先ページをページＰ５に変更し、８Ｋバイトのサイズを有するデータｄ８と、８Ｋバイトのサイズを有するデータｄ９とを互いに結合することによって得られる１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から取得する。この場合、コントローラ４は、これに限定されないが、例えば、２回のＤＭＡ転送によって、この１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から内部バッファ３１に転送してもよい。最初のＤＭＡ転送では、データｄ８の先頭位置を指定する転送元アドレスと、データ長＝８ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ８の先頭位置を指定する転送元アドレスは、データｄ８に対応するライトコマンド内のデータポインタによって表される。２回目のＤＭＡ転

送では、データｄ９の先頭位置を指定する転送元アドレスと、データ長＝８ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ９の先頭位置を指定する転送元アドレスは、データｄ９に対応するライトコマンド内のデータポインタによって表される。

【０２３６】

そして、コントローラ４は、この１６Ｋバイトライトデータ（ｄ８、ｄ９）を、書き込み先ブロックＢＬＫ＃１のページＰ５に書き込まれるべきデータとしてＮＡＮＤ型フラッシュメモリ５に転送する。

コントローラ４は、書き込み先ブロックＢＬＫ＃１の次の書き込み先ページをページＰ６に変更し、データｄ１０内の先頭の１６Ｋバイトデータｄ１０－１をホスト２のライトデータバッファ５１から取得する。この場合、コントローラ４は、これに限定されないが、例えば、１回のＤＭＡ転送によって、この１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から内部バッファ３１に転送してもよい。このＤＭＡ転送では、データｄ１０－１の先頭位置を指定する転送元アドレスと、データ長＝１６ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ１０－１の先頭位置を指定する転送元アドレスは、データｄ１０に対応するライトコマンド内のデータポインタによって表される。

【０２３７】

そして、コントローラ４は、この１６Ｋバイトライトデータ（ｄ１０－１）を、書き込み先ブロックＢＬＫ＃１のページＰ６に書き込まれるべきデータとしてＮＡＮＤ型フラッシュメモリ５に転送する。

コントローラ４は、書き込み先ブロックＢＬＫ＃１の次の書き込み先ページをページＰ７に変更し、データｄ１０内の次の１６Ｋバイトデータｄ１０－２をホスト２のライトデータバッファ５１から取得する。この場合、コントローラ４は、これに限定されないが、例えば、１回のＤＭＡ転送によって、この１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から内部バッファ３１に転送してもよい。このＤＭＡ転送では、データｄ１０－２の先頭位置を指定する転送元アドレスと、データ長＝１６ＫＢがＤＭＡＣ１５にセットされてもよい。データｄ１０－２の先頭位置を指定する転送元アドレスは、データｄ１０に対応するライトコマンド内のデータポインタの値に１６ＫＢ分のオフセットを加算することによって求めることができる。

【０２３８】

そして、コントローラ４は、この１６Ｋバイトライトデータ（ｄ１０－２）を、書き込み先ブロックＢＬＫ＃１のページＰ７に書き込まれるべきデータとしてＮＡＮＤ型フラッシュメモリ５に転送する。

そして、コントローラ４は、フォギー書き込み動作によって４ページ分のデータ（Ｐ４～Ｐ７）を書き込み先ブロックＢＬＫ＃１のワード線ＷＬ１に接続された複数のメモリセルに書き込む。

【０２３９】

このように、コントローラ４は、書き込み先ブロックＢＬＫ＃１の書き込み動作の進行に合わせて、書き込み先ブロックＢＬＫ＃１の書き込み先ページに転送すべき１６Ｋバイトデータをホスト２から取得する。

そして、ワード線ＷＬ３に接続された複数のメモリセルに対するフォギー書き込み動作が終了すると、ワード線ＷＬ０に接続された複数のメモリセルに対するファイン書き込み動作の実行が可能となる。コントローラ４は、書き込み先ブロックＢＬＫ＃１の次の書き込み先ページをページＰ１に変更し、上述と同様の手順で、ライトデータ（Ｐ０～Ｐ３）をページ単位でＮＡＮＤ型フラッシュメモリ５に再び転送し、そしてファイン書き込み動作によってこれら４ページ分のライトデータ（Ｐ０～Ｐ３）を書き込み先ブロックＢＬＫ＃１のワード線ＷＬ０に接続された複数のメモリセルに書き込む。

【０２４０】

これにより、最初の６つのライトコマンド、つまりデータｄ１に対応するライトコマンド、データｄ２に対応するライトコマンド、データｄ３に対応するライトコマンド、データｄ４に対応するライトコマンド、データｄ５に対応するライトコマンド、データｄ６に対

応するライトコマンドの各々に関しては、各ライトコマンドに関連付けせけたライトデータ全体に対するフォギー・ファイン書き込み動作の全てが終了し、且つデータｄ１～ｄ６の各々はＮＡＮＤ型フラッシュメモリ５から読み出し可能となる。このため、コントローラ４は、最初の６つのライトコマンドにそれぞれに対応する６つのコマンド完了レスポンスをホスト２に返す。

【０２４１】

なお、図４７では、書き込み先ブロックＢＬＫ＃１を指定するライトコマンドそれぞれに関連づけられたライトデータを、書き込み先ブロックＢＬＫ＃１の書き込み動作の進行に合わせて１６Ｋバイトの単位でホスト２からフラッシュストレージデバイス３に転送する動作を説明したが、他の各書き込み先ブロックＢＬＫ＃に関しても、図９で説明した動作と同様の動作が実行される。

【０２４２】

図４８のフローチャートは、フラッシュストレージデバイス３によって実行されるデータ書き込み処理の手順を示す。ここでは、フラッシュストレージデバイス３が、タイプ＃２－ストレージデバイスとして実現されており、かつ、ホスト２からのライトデータを書き込み予定のページアドレスを、ＮＡＮＤ型フラッシュメモリ５への書き込みの終了を待機することなく、ホスト２に通知する仕組みを備えていることを想定する。

【０２４３】

フラッシュストレージデバイス３のコントローラ４は、データポインタ、長さ、ブロック識別子（例えばブロックアドレス）を各々が含むライトコマンドそれぞれをホスト２から受信する（ステップＢ１）。

次いで、コントローラ４は、特定の書き込み先ブロックを指定する一つのライトコマンドに対応する大きなサイズのライトデータを２以上のデータ部に分割、またはこの特定の書き込み先ブロックを指定する２以上のライトコマンドに対応する２以上のライトデータを結合し、これによってＮＡＮＤ型フラッシュメモリ５の書き込み単位（データ転送サイズ）と同じサイズの単位でデータをホスト２からフラッシュストレージデバイス３に転送する（ステップＢ２）。ステップＢ２では、図４７で説明したように、例えば、小さなサイズを有する幾つかのライトデータ部を互いに結合することによって得られる一つの１６Ｋバイトデータ、あるいは大きなサイズを有するライトデータを分割することによって得られる幾つかの１６Ｋバイトデータの一つが、ホスト２からフラッシュストレージデバイス３に転送される。フラッシュストレージデバイス３が内部バッファ３１を含む構成であるケースにおいては、ホスト２からフラッシュストレージデバイス３に転送される各１６バイトライトデータは内部バッファ３１に格納される。また、ステップＢ２では、小さなサイズを有する幾つかのライトデータ部を互いに結合するために、コントローラ４は、ある書き込み先ブロックを指定する識別子を有する先行するライトコマンドに関連付けられたライトデータのサイズが書き込み単位（例えば１６Ｋバイト）よりも小さい場合、この書き込み先ブロックを指定する識別子を有する後続のライトコマンドの受信を待つ。

【０２４４】

コントローラ４は、ホスト２から転送された１６Ｋバイトデータについて、この１６Ｋバイトデータを特定の書き込み先ブロックに書き込む前に、この１６Ｋバイトデータの書き込み先に割り当てた特定の書き込み先ブロック内のアドレスをホスト２に通知する（ステップＢ３）。それから、コントローラ４は、ホスト２から転送された１６Ｋバイトデータをを ＮＡＮＤ型フラッシュメモリ５に転送し、この１６Ｋバイトデータを、この特定の書き込み先ブロック内の書き込み先に割り当てたアドレスに書き込む（ステップＢ４）。コントローラ４は、この書き込みが成功したか否かを判定し（ステップＢ５）、エラーの場合（ステップＢ５：ＮＯ）、ステップＢ３から処理を繰り返す。つまり、同一のデータについてのホスト２へのアドレスの通知は複数回発生し得る。成功の場合は（ステップＢ５：ＹＥＳ）、ステップＢ６へと進む。

【０２４５】

そして、コントローラ４は、この特定の書き込み先ブロックを指定する一つのライトコマ

ンドに関連付けられたライトデータ全体に対する書き込み動作（ＮＡＮＤ型フラッシュメモリ５に同じデータを１回または複数回転送することを伴う書き込み動作）の全てが終了したか否かを判定する（ステップＢ６）。

【０２４６】

この特定の書き込み先ブロックを指定する一つのライトコマンドに関連付けられたライトデータ全体に対する書き込み動作の全てが終了したならば、コントローラ４は、このライトコマンドのコマンド完了を示すレスポンスをホスト２に返す（ステップＢ７）。ライトデータが格納されるライトデータバッファ５１上の領域に関する解放可能通知のホスト２への送信タイミングについては後述する。

【０２４７】

図４９のフローチャートは、フラッシュストレージデバイス３によって実行されるデータ書き込み処理の別の手順を示す。ここでも、フラッシュストレージデバイス３が、タイプ＃２－ストレージデバイスとして実現されており、かつ、ホスト２からのライトデータを書き込み予定のページアドレスを、ＮＡＮＤ型フラッシュメモリ５への書き込みの終了を待機することなく、ホスト２に通知する仕組みを備えていることを想定する。

【０２４８】

フラッシュストレージデバイス３のコントローラ４は、データポインタ、長さ、ブロック識別子（例えばブロックアドレス）を各々が含むライトコマンドそれぞれをホスト２から受信する（ステップＢ１１）。

次いで、コントローラ４は、特定の書き込み先ブロックを指定する一つのライトコマンドに対応する大きなサイズのライトデータを２以上のデータ部に分割、またはこの特定の書き込み先ブロックを指定する２以上のライトコマンドに対応する２以上のライトデータを結合し、これによってＮＡＮＤ型フラッシュメモリ５の書き込み単位（データ転送サイズ）と同じサイズの単位でデータをホスト２からフラッシュストレージデバイス３に転送する（ステップＢ１２）。ステップＢ１２では、図４７で説明したように、例えば、小さなサイズを有する幾つかのライトデータ部を互いに結合することによって得られる一つの１６Ｋバイトデータ、あるいは大きなサイズを有するライトデータを分割することによって得られる幾つかの１６Ｋバイトデータの一つが、ホスト２からフラッシュストレージデバイス３に転送される。フラッシュストレージデバイス３が内部バッファ３１を含む構成であるケースにおいては、ホスト２からフラッシュストレージデバイス３に転送される各１６バイトライトデータは内部バッファ３１に格納される。また、ステップＢ２では、小さなサイズを有する幾つかのライトデータ部を互いに結合するために、コントローラ４は、ある書き込み先ブロックを指定する識別子を有する先行するライトコマンドに関連付けられたライトデータのサイズが書き込み単位（例えば１６Ｋバイト）よりも小さい場合、この書き込み先ブロックを指定する識別子を有する後続のライトコマンドの受信を待つ。

【０２４９】

コントローラ４は、ホスト２から転送された１６ＫバイトデータをＮＡＮＤ型フラッシュメモリ５に転送し、この１６Ｋバイトデータを、この特定の書き込み先ブロックに書き込む（ステップＢ１３）。

コントローラ４は、この書き込みが成功したか否かを判定し（ステップＢ１４）、エラーの場合（ステップＢ１４：ＮＯ）、ステップＢ１３から処理を繰り返す。成功の場合は（ステップＢ１４：ＹＥＳ）、ステップＢ１５へと進む。

【０２５０】

コントローラ４は、ホスト２から転送された１６Ｋバイトデータの書き込み先に割り当てた特定の書き込み先ブロック内のアドレスをホスト２に通知する（ステップＢ１５）。

そして、コントローラ４は、この特定の書き込み先ブロックを指定する一つのライトコマンドに関連付けられたライトデータ全体に対する書き込み動作（ＮＡＮＤ型フラッシュメモリ５に同じデータを１回または複数回転送することを伴う書き込み動作）の全てが終了したか否かを判定する（ステップＢ１６）。

【０２５１】

この特定の書き込み先ブロックを指定する一つのライトコマンドに関連付けられたライトデータ全体に対する書き込み動作の全てが終了したならば、コントローラ４は、このライトコマンドのコマンド完了を示すレスポンスをホスト２に返す（ステップＢ１７）。ライトデータが格納されるライトデータバッファ５１上の領域に関する解放可能通知のホスト２への送信タイミングについては後述する。

【０２５２】

図５０のフローチャートは、フラッシュストレージデバイス３によって実行されるホスト２への解放可能通知の送信処理の手順を示す。

コントローラ４は、第１に、特定の書き込み先ブロックを指定する一つのライトコマンドに関連付けられたライトデータ全体に対する書き込み動作の全てが終了したか否かを判定する（ステップＣ１）。また、コントローラ４は、第２に、このライトデータを対象とするリードコマンドが存在するか否かを判定する（ステップＣ２）。なお、ステップＣ１の処理とステップＣ２の処理とは、並列的に実行される。そして、コントローラ４は、特定の書き込み先ブロックを指定する一つのライトコマンドに関連付けられたライトデータ全体に対する書き込み動作の全てが終了し（ステップＣ１：ＹＥＳ）、かつ、このライトデータを対象とするリードコマンドが存在しない場合に（ステップＣ２：ＮＯ）、このライトデータが格納されるライトデータバッファ５１上の領域に関する解放可能通知をホスト２に送信する（ステップＣ３）。

【０２５３】

図５１のフローチャートは、ホスト２によって実行されるライトデータ破棄処理の手順を示す。

ホスト２は、ライトコマンドのコマンド完了を示すレスポンスをフラッシュストレージデバイス３から受信したか否かを判定する（ステップＤ１）。あるライトコマンドのコマンド完了を示すレスポンスをフラッシュストレージデバイス３から受信した場合（ステップＤ１：ＹＥＳ）、ホスト２は、さらに、このライトコマンドに関連付けられたライトデータに関する解放可能通知をフラッシュストレージデバイス３から受信したか否かを判定する（ステップＤ２）。このライトデータに関する解放可能通知をフラッシュストレージデバイス３から受信した場合（ステップＤ２：ＹＥＳ）、ホスト２は、このライトコマンドに関連付けられたライトデータをライトデータバッファ５１から破棄する（ステップＤ３）。

【０２５４】

図５２は、ある書き込み先ブロックを指定する最後のライトコマンドが受信されてから閾期間、この書き込み先ブロックを指定する次のライトコマンドが受信されない場合に、フラッシュストレージデバイス３によって実行されるダミーデータ書き込み処理を示す。

【０２５５】

図５２の左部に示されるデータｄ１、データｄ２、データｄ３、データｄ４は、書き込み先ブロックＢＬＫ＃１を指定する４個のライトコマンドそれぞれに対応する４個のライトデータを示している。図５２では、データｄ１、データｄ２、データｄ３、データｄ４の各々が４Ｋバイトのサイズを有する場合が想定されている。

【０２５６】

（１）コントローラ４は、データｄ１、データｄ２、データｄ３、データｄ４を互いに結合することによって得られる１６Ｋバイトライトデータをホスト２のライトデータバッファ５１から取得する。そして、コントローラ４は、この１６Ｋバイトライトデータを、書き込み先ブロックＢＬＫ＃１のページＰ０に書き込まれるべきデータとしてＮＡＮＤ型フラッシュメモリ５に転送する。書き込み先ブロックＢＬＫ＃１を指定する最後のライトコマンド、つまりデータｄ４の書き込みを要求したライトコマンド、が受信されてから閾期間、書き込み先ブロックＢＬＫ＃１を指定する後続のライトコマンドが受信されない場合、コントローラ４は、最後のライトコマンドのコマンド完了を示すレスポンスを所定時間内にホスト２に返すことを可能にするために、書き込み先ブロックＢＬＫ＃１内の１以上のページにダミーデータを書き込み、次のライトデータが書き込まれるべき書き込み先ブ

ロックＢＬＫ＃１内の書き込み先ページの位置を進める。例えば、コントローラ４は、ページＰ１～Ｐ３に対応する３ページ分のダミーデータをページ単位でＮＡＮＤ型フラッシュメモリ５に転送し、フォギー書き込み動作によって４ページ分のデータ（Ｐ０～Ｐ３）を書き込み先ブロックＢＬＫ＃１のワード線ＷＬ０に接続された複数のメモリセルに書き込む。

【０２５７】

（２）次いで、コントローラ４は、ページＰ４～Ｐ７に対応する４ページ分のダミーデータをページ単位でＮＡＮＤ型フラッシュメモリ５に転送し、フォギー書き込み動作によって４ページ分のデータ（Ｐ４～Ｐ７）を書き込み先ブロックＢＬＫ＃１のワード線ＷＬ１に接続された複数のメモリセルに書き込む。

【０２５８】

（３）次いで、コントローラ４は、ページＰ８～Ｐ１１に対応する４ページ分のダミーデータをページ単位でＮＡＮＤ型フラッシュメモリ５に転送し、フォギー書き込み動作によって４ページ分のデータ（Ｐ８～Ｐ１１）を書き込み先ブロックＢＬＫ＃１のワード線ＷＬ２に接続された複数のメモリセルに書き込む。

【０２５９】

（４）次いで、コントローラ４は、ページＰ１２～Ｐ１５に対応する４ページ分のダミーデータをページ単位でＮＡＮＤ型フラッシュメモリ５に転送し、フォギー書き込み動作によって４ページ分のデータ（Ｐ１２～Ｐ１５）を書き込み先ブロックＢＬＫ＃１のワード線ＷＬ３に接続された複数のメモリセルに書き込む。

【０２６０】

（５）次いで、コントローラ４は、データｄ１、データｄ２、データｄ３、データｄ４を互いに結合することによって得られる１６Ｋバイトライトデータをライトデータバッファ５１または内部バッファ３１からＮＡＮＤ型フラッシュメモリ５に転送し、さらに、ＷＬ０のフォギー書き込み動作で使用した３ページ分のダミーデータ（Ｐ０～Ｐ３）と同じ３ページ分のダミーデータ（Ｐ０～Ｐ３）をページ単位でＮＡＮＤ型フラッシュメモリ５に転送する。そして、コントローラ４は、ファイン書き込み動作によって４ページ分のデータ（Ｐ０～Ｐ３）を書き込み先ブロックＢＬＫ＃１のワード線ＷＬ０に接続された複数のメモリセルに書き込む。これにより、データｄ１、データｄ２、データｄ３、データｄ４の複数段階の書き込み動作が全て完了し、データｄ１、データｄ２、データｄ３、データｄ４がＮＡＮＤ型フラッシュメモリ５読み出し可能となる。コントローラ４は、データｄ１の書き込みを要求した最初のライトコマンドのコマンド完了を示すレスポンスと、データｄ２の書き込みを要求した２番目のライトコマンドのコマンド完了を示すレスポンスと、データｄ３の書き込みを要求した３番目のライトコマンドのコマンド完了を示すレスポンスと、データｄ４の書き込みを要求した４番目のライトコマンドのコマンド完了を示すレスポンスとをホスト２に返す。

【０２６１】

本実施形態では、ＮＡＮＤ型フラッシュメモリ５のデータ書き込み単位と同じデータサイズの単位でライトデータがホスト２からフラッシュストレージデバイス３に転送され、あるライトコマンドのライトデータ全体の書き込み動作の全てが終了した時点で、またはこのライトデータ全体の書き込み動作の全てが終了し且つこのライトデータ全体が読み出し可能となった時点で、このライトコマンドのコマンド完了を示すレスポンスがホスト２に返される。このため、例えば、小さなライトデータをある書き込み先ブロックに書き込むことを要求するライトコマンドがホスト２からフラッシュストレージデバイス３に発行された後にしばらくの間、この書き込み先ブロックを指定する後続のライトコマンドがホスト２から発行されない場合には、このライトコマンドのタイムアウトエラーが起こる可能性がある。本実施形態では、コントローラ４は、あるブロック識別子を有する最後のライトコマンドがホスト２から受信されてから閾期間このブロック識別子を有する次のライトコマンドが受信されない場合、ダミーデータを、このブロック識別子に対応する書き込み先ブロック内の次の１以上の未書き込みページに書き込む。したがって、必要に応じて、

この書き込み先ブロックの書き込み動作を進行させることができるので、ライトコマンドのタイムアウトエラーが起こることを防止することができる。

【０２６２】

図５３のフローチャートは、フラッシュストレージデバイス３によって実行されるダミーデータ書き込み処理の手順を示す。ここでは、フォギー・ファイン書き込み動作のような複数段階の書き込み動作によってデータが書き込み先ブロックに書き込まれる場合を想定する。

【０２６３】

フラッシュストレージデバイス３のコントローラ４は、ある書き込み先ブロックを指定する最後のライトコマンドに関連付けられたライトデータをフォギー書き込み動作のような第１段階の書き込み動作によってこの書き込み先ブロックに書き込む。この最後のライトコマンドの受信から閾期間（Ｔｈ）、この書き込み先ブロックを指定する次のライトコマンドが受信されない場合（ステップＳ３１のＹＥＳ）、コントローラ４は、最後のライトコマンドに関連付けられたライトデータが書き込まれた書き込み先ブロック内のページに後続する１以上のページにダミーデータを書き込み、これによって、次のライトデータが書き込まれるべきこの書き込み先ブロック内の書き込み先ページの位置を進める（ステップＳ３２）。この書き込み先ブロックへのダミーデータの書き込みによって書き込み先ページの位置が進み、これによって最後のライトコマンドに関連付けられたライトデータのファイン書き込み動作（第２段階の書き込み動作）が実行可能となると、コントローラ４は、最後のライトコマンドに関連付けられたライトデータをライトデータバッファ５１または内部バッファ（共有キャッシュ）３１からＮＡＮＤ型フラッシュメモリ５に再び転送し、このライトデータのファイン書き込み動作を実行する（ステップＳ３３）。

【０２６４】

最後のライトコマンドに関連付けられたライトデータのファイン書き込み動作が終了すると、つまりこのライトデータ全体の複数段階の書き込み動作の全てが終了すると、コントローラ４は、この最後のライトコマンドのコマンド完了を示すレスポンスをホスト２に返す（ステップＳ３４）。

【０２６５】

このように、複数段階の書き込み動作によってライトデータを書き込み先ブロックに書き込むケースにおいては、コントローラ４は、最後のライトコマンドに関連付けられたライトデータの第２段階の書き込み動作が実行可能になるように、ダミーデータをこの書き込み先ブロック内の１以上のページに書き込み、次のライトデータが書き込まれるべきこの書き込み先ブロック内の書き込み先ページの位置を進める。

【０２６６】

図５４は、内部バッファ（共有キャッシュ）３１を使用してコントローラ４によって実行されるデータ転送動作を示す。

内部バッファ（共有キャッシュ）３１は、複数の書き込み先ブロックＢＬＫ＃１、ＢＬＫ＃２、…、ＢＬＫ＃ｎによって共有される。フラッシュストレージデバイス３のコントローラ４は、書き込み先ブロックＢＬＫ＃１、ＢＬＫ＃２、…、ＢＬＫ＃ｎの各々について以下の処理を実行する。

【０２６７】

以下では、書き込み先ブロックＢＬＫ＃１を例示して説明する。

コントローラ４が、書き込み先ブロックＢＬＫ＃１を指定する１以上のライトコマンドを受信した後、コントローラ４は、書き込み先ブロックＢＬＫ＃１を指定する一つのライトコマンドに関連付けられたライトデータを複数のライトデータに分割、または書き込み先ブロックＢＬＫ＃１を指定する２以上のライトコマンドにそれぞれ関連付けられたライトデータを互いに結合することによって得られる、ＮＡＮＤ型フラッシュメモリ５の書き込み単位と同じサイズを有するライトデータを、ライトデータバッファ５１から取得する。そして、コントローラ４は、ライトデータバッファ５１から取得される、各々がＮＡＮＤ型フラッシュメモリ５の書き込み単位と同じサイズを有する複数のライトデータを内部バ

ッファ（共有キャッシュ）３１に格納する。
【０２６８】
ライトデータバッファ５１は、必ずしも、ホストメモリ上の連続する一つの領域から構成される必要は無く、図５４に示されているように、複数のライトデータバッファ５１－１、５１－２、…、５１－ｎによって実現されてもよい。
コントローラ４は、書き込み先ブロックＢＬＫ＃１に次に書き込むべきライトデータ（第１のライトデータ）を内部バッファ（共有キャッシュ）３１から取得し、第１のライトデータをＮＡＮＤ型フラッシュメモリ５に転送し、フォギー書き込み動作のような第１段階の書き込み動作によってこのライトデータを書き込み先ブロックＢＬＫ＃１に書き込む。
【０２６９】
内部バッファ（共有キャッシュ）３１にホスト２からのライトデータを効率よく蓄積できるようにするため、ホスト２から取得されるライトデータを格納するための空き領域が内部バッファ（共有キャッシュ）３１に無い場合には、コントローラ４は、フォギー書き込み動作のような第１段階の書き込み動作が終了している内部バッファ（共有キャッシュ）３１内のライトデータ（フォギーステートのライトデータ）を破棄して、空き領域を内部バッファ（共有キャッシュ）３１に確保する。
【０２７０】
例えば、内部バッファ（共有キャッシュ）３１に空き領域がない状態でホスト２から任意の書き込み先ブロックを指定する新たなライトコマンドを受信した場合に、コントローラ４は、フォギー書き込み動作のような第１段階の書き込み動作が終了している内部バッファ（共有キャッシュ）３１内のライトデータ（フォギーステートのライトデータ）を破棄して、新たなライトコマンドに対応するライトデータを格納可能な空き領域を内部バッファ（共有キャッシュ）３１に確保してもよい。
【０２７１】
例えば、内部バッファ（共有キャッシュ）３１全体が多数のフォギーステートのライトデータで満たされている状態でホスト２から新たなライトコマンドを受信した場合には、コントローラ４は、これらフォギーステートのライトデータの中から破棄すべき特定のライトデータを選択してもよく、この選択したライトデータを破棄してもよい。これにより、制限された容量を有する内部バッファ（共有キャッシュ）３１を、複数の書き込み先ブロック間で効率よく共有することができる。
【０２７２】
コントローラ４は、第１のライトデータのファイン書き込み動作のような第２段階の書き込み動作を実行すべき時点において第１のライトデータが内部バッファ（共有キャッシュ）３１に存在しない場合には、第１のライトデータを取得するための要求（転送要求：ＤＭＡ転送要求）をホスト２に送信することによってホスト２のライトデータバッファ５１から再び取得する。この取得された第１のライトデータは内部バッファ（共有キャッシュ）３１に格納されてもよい。そして、コントローラ４は、取得された第１のライトデータをＮＡＮＤ型フラッシュメモリ５に転送し、ファイン書き込み動作のような第２段階の書き込み動作によってこの第１のライトデータを書き込み先ブロックＢＬＫ＃１に書き込む。
【０２７３】
第１のライトデータのファイン書き込み動作のような第２段階の書き込み動作を実行すべき時点において第１のライトデータが内部バッファ（共有キャッシュ）３１に存在している場合には、コントローラ４は、この内部バッファ（共有キャッシュ）３１から第１のライトデータを取得し、取得された第１のライトデータをＮＡＮＤ型フラッシュメモリ５に転送し、ファイン書き込み動作のような第２段階の書き込み動作によってこの第１のライトデータを書き込み先ブロックＢＬＫ＃１に書き込む。
【０２７４】
ＮＡＮＤ型フラッシュメモリ５への第１のライトデータの最終回のデータ転送（ここでは、ファイン書き込み動作のためのデータ転送）を行った後、コントローラ４は、この第１のライトデータを内部バッファ（共有キャッシュ）３１から破棄することによって内部バ

ッファ（共有キャッシュ）３１に空き領域を確保する。あるいは、コントローラ４はこの第１のライトデータのファイン書き込み動作が終了した場合に、この第１のライトデータを内部バッファ（共有キャッシュ）３１から破棄してもよい。

【０２７５】

さらに、コントローラ４は、あるライトコマンドに関連付けられたライトデータ全体のファイン書き込み動作が終了した場合、あるいはこのライトデータ全体のファイン書き込み動作が終了し且つこのライトデータ全体がＮＡＮＤ型フラッシュメモリ５から読み出し可能となった場合に、このライトコマンドのコマンド完了を示すレスポンスをホスト２に返す。

【０２７６】

内部バッファ（共有キャッシュ）３１はある限られた容量を有しているが、書き込み先ブロックの数がある一定数以下であるならば、第２段階の書き込み動作を実行すべき時点において第１のライトデータが内部バッファ（共有キャッシュ）３１に存在する確率（ヒット率）は比較的高い。したがって、同じライトデータをホスト２からフラッシュストレージデバイス３に複数回転送すること無く、フォギー・ファイン書き込み動作のような複数段階の書き込み動作を実行することができる。これにより、ホスト２とフラッシュストレージデバイス３との間のデータトラフィックを削減できるので、データ書き込みの度に同じライトデータをホスト２からフラッシュストレージデバイス３に複数回転送する場合に比し、フラッシュストレージデバイス３のＩ／Ｏ性能を向上することができる。

【０２７７】

書き込み先ブロックの数は、ホスト２を利用するクライアントの数と同数であってよい。この場合、あるクライアントに対応するデータはこのクライアントに対応する書き込み先ブロックに書き込まれ、他のクライアントに対応するデータは別の書き込み先ブロックに書き込まれる。したがって、ホスト２を利用するクライアントの数が増えるにつれて、内部バッファ（共有キャッシュ）３１のヒット率は低下する。しかし、第１のライトデータが内部バッファ（共有キャッシュ）３１に存在しない場合（ミス）には、コントローラ４はこの第１のライトデータをホスト２から取得する。したがって、クライアントの数が増加しても、フォギー・ファイン書き込み動作のような複数段階の書き込み動作を正常に実行することができる。

【０２７８】

よって、フラッシュストレージデバイス３は、フラッシュストレージデバイス３を共有するクライアントの数の増加（つまり、同時に利用可能な書き込み先ブロックの数の増加）に柔軟に対応でき且つホスト２とフラッシュストレージデバイス３との間のデータトラフィックを削減することができる。

【０２７９】

ここでは、書き込み先ブロックＢＬＫ＃１にデータを書き込むための書き込み処理について説明したが、他の全ての書き込み先ブロックの各々に対しても同様の書き込み処理が実行される。

図５５のフローチャートは、内部バッファ（共有キャッシュ）３１を使用してコントローラ４によって実行されるデータ書き込み処理の手順を示す。

【０２８０】

コントローラ４は、データポインタと、ライトデータの長さと、複数の書き込み先ブロックのいずれか一つを指定する識別子（例えばブロックアドレス）とを各々が含む１つ以上のライトコマンドをホスト２から受信する（ステップＳ１０１）。同じ書き込み先ブロックを示す識別子を有する一つ以上のライトコマンドを受信した後、コントローラ４は、これらライトコマンド内の一つのライトコマンドに関連付けられたライトデータを複数のライトデータに分割、または同じ書き込み先ブロックを示す識別子を有する２以上のライトコマンドにそれぞれ関連付けられたライトデータを互いに結合することによって得られる、ＮＡＮＤ型フラッシュメモリ５の書き込み単位と同じサイズを有するライトデータを、ライトデータバッファ５１から内部バッファ（共有キャッシュ）３１に転送する（ステッ

プＳ１０２）。
【０２８１】
コントローラ４は、この書き込み先ブロックに次に書き込むべきライトデータを内部バッファ（共有キャッシュ）３１から取得し、このライトデータをＮＡＮＤ型フラッシュメモリ５に転送し、フォギー書き込み動作によってこの第１のライトデータをこの書き込み先ブロックに書き込む（ステップＳ１０３、Ｓ１０４）。ＮＡＮＤ型フラッシュメモリ５がＱＬＣ－フラッシュとして実現されている場合には、ステップＳ１０３では、４ページ分のライトデータがページ単位でＮＡＮＤ型フラッシュメモリ５に転送され、ステップＳ１０４では、４ページ分のライトデータがフォギー書き込み動作によってこの書き込み先ブロック内の書き込み対象の一つのワード線に接続された複数のメモリセルに書き込まれる。
【０２８２】
なお、ライトデータバッファ５１から内部バッファ（共有キャッシュ）３１へのライトデータの転送は、各書き込み先ブロックの書き込み動作の進行に合わせて実行される。例えば、ある書き込み先ブロックのあるページに書き込むべきライトデータをＮＡＮＤ型フラッシュメモリチップに転送する動作が終了した場合に、この書き込み先ブロックの次のページに書き込むべきライトデータがライトデータバッファ５１から内部バッファ（共有キャッシュ）３１へ転送されてもよい。あるいは、ある書き込み先ブロックのあるページに書き込むべきライトデータをこの書き込み先ブロックを含むＮＡＮＤ型フラッシュメモリチップに転送する動作が終了し、且つこのライトデータをこの書き込み先ブロックに書き込む動作が終了した場合に、この書き込み先ブロックの次のページに書き込むべきライトデータがライトデータバッファ５１から内部バッファ（共有キャッシュ）３１へ転送されてもよい。
【０２８３】
フォギー書き込み動作が行われたこのライトデータのファイン書き込み動作を開始すべき時点において、コントローラ４は、このライトデータが、内部バッファ（共有キャッシュ）３１に存在するか否かを判定する。
このライトデータが内部バッファ（共有キャッシュ）３１に存在するならば（ステップＳ１０６のＹＥＳ）、コントローラ４は、このライトデータを内部バッファ（共有キャッシュ）３１から取得し、このライトデータをＮＡＮＤ型フラッシュメモリ５に転送し、ファイン書き込み動作によってこのライトデータをこの書き込み先ブロックに書き込む（ステップＳ１０７、Ｓ１０８、Ｓ１０９）。これにより、このライトデータはＮＡＮＤ型フラッシュメモリ５から読み出し可能となる。
【０２８４】
コントローラ４は、ライトコマンド毎に、そのライトデータ全体のフォギー・ファイン書き込み動作が終了し且つそのライトデータ全体がＮＡＮＤ型フラッシュメモリ５から読み出し可能となったか否かを判定する。そして、コントローラ４は、フォギー・ファイン書き込み動作が終了し且つＮＡＮＤ型フラッシュメモリ５から読み出し可能となったライトデータに対応するライトコマンドのコマンド完了を示すレスポンスをホスト２に返す（ステップＳ１１０）。もしステップＳ１０９の処理によってあるライトコマンドに関連付けられたライトデータ全体のファイン書き込み動作が終了したならば、ステップＳ１１０では、このライトコマンドのコマンド完了を示すレスポンスがホスト２に返されてもよい。
【０２８５】
図５６のフローチャートは、コントローラ４によって実行されるデータ読み出し処理の手順を示す。ここでは、フラッシュストレージデバイス３が、ホスト２のライトデータバッファ５１内に存在するライトデータ中のデータが対象となるリードコマンドがホスト２から受信された場合、そのリードコマンドが終了するまで、そのデータが格納される領域についての解放可能通知をホスト２へ送信しないようにする仕組みを備えていることを想定する。
【０２８６】
上述したように、コントローラ４は、ホスト２から受信されるリードコマンドによって指

定されたデータが、その書き込み動作（ＮＡＮＤ型フラッシュメモリ５に同じデータを１回または複数回転送する書き込み動作）の全てが終了していないデータ、またはその書き込み動作の全てが終了しているがＮＡＮＤ型フラッシュメモリ５からまだ読み出し可能となっていないデータである場合、このデータが内部バッファ（共有キャッシュ）３１に存在するか否かを判定する。このデータが内部バッファ（共有キャッシュ）３１に存在しない場合、コントローラ４は、このデータをライトデータバッファ５１から取得し、このデータを内部バッファ（共有キャッシュ）３１に格納し、このデータを内部バッファ（共有キャッシュ）３１からホスト２に返す。

【０２８７】

具体的には、以下のデータ読み出し処理が実行される。

コントローラ４がホスト２からリードコマンドを受信した場合（ステップＥ１：ＹＥＳ）、コントローラ４は、このリードコマンドによって指定されたデータが、その書き込み動作の全てが終了し、且つＮＡＮＤ型フラッシュメモリ５から読み出し可能なデータであるか否かを判定する（ステップＥ２）。

【０２８８】

このデータがＮＡＮＤ型フラッシュメモリ５から読み出し可能であるならば（ステップＥ２：ＹＥＳ）、コントローラ４は、このデータをＮＡＮＤ型フラッシュメモリ５から読み出し、読み出されたデータをホスト２に返す（ステップＥ３）。ステップＥ３では、コントローラ４は、この読み出されたデータを、リードコマンドに含まれるデータポインタによって指定されるリードデータバッファ５３内の位置に転送する。

【０２８９】

このデータがＮＡＮＤ型フラッシュメモリ５から読み出し可能でないならば（ステップＥ２：ＮＯ）、コントローラ４は、まず、ライトデータバッファ５１上のデータが破棄されないように、解放可能通知のホスト２への送信が禁止される状態に設定する（ステップＥ５）。そして、コントローラ４は、このデータが内部バッファ（共有キャッシュ）３１に存在するか否かを判定する（ステップＥ５）。

【０２９０】

このデータが内部バッファ（共有キャッシュ）３１に存在するならば（ステップＥ５：ＹＥＳ）、コントローラ４は、このデータを内部バッファ（共有キャッシュ）３１から読み出し、この読み出したデータをホスト２に返す（ステップＥ６）。

ステップＥ６では、コントローラ４は、この読み出されたデータを、リードコマンドに含まれるデータポインタによって指定されるリードデータバッファ５３内の位置に転送する。

【０２９１】

このデータが内部バッファ（共有キャッシュ）３１に存在しないならば（ステップＥ５：ＮＯ）、コントローラ４は、このデータをライトデータバッファ５１から取得し、内部バッファ（共有キャッシュ）３１に格納する（ステップＥ７）。ステップＥ７では、ＤＭＡＣ１５によってこのデータがライトデータバッファ５１から内部バッファ（共有キャッシュ）３１の空き領域に転送される。内部バッファ（共有キャッシュ）３１の空き領域が無い場合には、内部バッファ（共有キャッシュ）３１の空き領域を確保する処理が実行される。そして、コントローラ４は、このデータを内部バッファ（共有キャッシュ）３１から読み出し、この読み出したデータをホスト２に返す（ステップＥ６）。ステップＥ６では、コントローラ４は、この読み出されたデータを、リードコマンドに含まれるデータポインタによって指定されるリードデータバッファ５３内の位置に転送する。そして、コントローラ４は、ステップＥ４で設定した、解放可能通知のホスト２への送信が禁止される状態を解除する（ステップＥ８）。

【０２９２】

図５７は、フラッシュストレージデバイス３に適用されるブロックリユースコマンドを示す。

ブロックリユースコマンドは、たとえば無効データや不要データのみが格納されているなどの理由で不要となった割り当て済みのブロックをフリーブロックに戻すことをフラッシ

ュストレージデバイス３に対して要求するコマンド（ブロック解放要求）である。ブロックリユースコマンドは、ＱｏＳドメインを指定するＱｏＳドメインＩＤと、フリーブロック化（解放）するブロックを指定するブロックアドレスとを含む。

【０２９３】

また、図５８は、フラッシュストレージデバイス３に適用されるライトコマンドの別の例を示す。具体的には、図８に示したライトコマンドが、フラッシュストレージデバイス３がタイプ＃１－ストレージデバイスとして実現されている場合において適用されるものであるのに対して、図５８に示すライトコマンドは、フラッシュストレージデバイス３がタイプ＃２－ストレージデバイスとして実現されている場合において適用されるものである。図５８中に施されているハッチングは、図８との相違点を示している。

【０２９４】

ライトコマンドは、フラッシュストレージデバイス３にデータの書き込みを要求するコマンドである。このライトコマンドは、コマンドＩＤ、ＱｏＳドメインＩＤ、論理アドレス、長さ、等を含んでもよい。

コマンドＩＤはこのコマンドがライトコマンドであることを示すＩＤ（コマンドコード）であり、ライトコマンドにはライトコマンド用のコマンドＩＤが含まれる。

【０２９５】

ＱｏＳドメインＩＤは、データが書き込まれるべきＱｏＳドメインを一意に識別可能な識別子である。あるエンドユーザからのライト要求に応じてホスト２から送信されるライトコマンドは、このエンドユーザに対応するＱｏＳドメインを指定するＱｏＳドメインＩＤを含んでもよい。ネームスペースＩＤがＱｏＳドメインＩＤとして扱われてもよい。

【０２９６】

論理アドレスは、書き込まれるべきライトデータを識別するための識別子である。この論理アドレスは、上述したように、ＬＢＡであってもよいし、キー・バリュー・ストアのキーであってもよい。論理アドレスがＬＢＡである場合には、このライトコマンドに含まれる論理アドレス（開始ＬＢＡ）は、ライトデータが書き込まれるべき論理位置（最初の論理位置）を示す。

【０２９７】

長さは、書き込まれるべきライトデータの長さを示す。この長さ（データ長）は、粒度（Ｇｒａｉｎ）の数によって指定されてもよいし、ＬＢＡの数によって指定されてもよいし、あるいはそのサイズがバイトによって指定されてもよい。

上述したように、コントローラ４は、ＮＡＮＤ型フラッシュメモリ５内の多数のブロックの各々が一つのグループのみに属するようにＮＡＮＤ型フラッシュメモリ５内の多数のブロックを複数のグループ（複数のＱｏＳドメイン）に分類することができる。そして、コントローラ４は、グループ（ＱｏＳドメイン）毎に、フリーブロックリスト（フリーブロックプール）とアクティブブロックリスト（アクティブブロックプール）とを管理することができる。

【０２９８】

各ブロックの状態は、有効データを格納しているアクティブブロックと、有効データを格納していないフリーブロックとに大別される。アクティブブロックである各ブロックは、アクティブブロックリストによって管理される。一方、フリーブロックである各ブロックは、フリーブロックリストによって管理される。

【０２９９】

ホスト２からライトコマンドを受信した時、コントローラ４は、ホスト２からのデータが書き込まれるべきブロック（書き込み先ブロック）およびこの書き込み先ブロック内の位置（書き込み先位置）を決定する。コントローラ４は、ＱｏＳドメインＩＤに対応するＱｏＳドメインに属するフリーブロック群の一つを書き込み先ブロックとして決定してもよい。書き込み先位置は、ページ書き込み順序の制約およびバッドページ等を考慮して決定される。そして、コントローラ４は、ホスト２からのデータを、書き込み先ブロック内の書き込み先位置に書き込む。

【０３００】
なお、この書き込み先ブロック全体がユーザデータで満たされたならば、コントローラ４は、この書き込み先ブロックをアクティブブロックリスト（アクティブブロックプール）に移動する。そして、コントローラ４は、このＱｏＳドメインに対応するフリーブロックリストからフリーブロックを再び選択し、この選択したフリーブロックを新たな書き込み先ブロックとして割り当てる。
【０３０１】
もしフリーブロックリストによって管理されている残りフリーブロックの数が所定のポリシーによって定められる閾値以下に低下した場合あるいはホスト２からガベージコレクションを実施する指示があった場合、コントローラ４は、このＱｏＳドメインのガベージコレクションを開始してもよい。
【０３０２】
このＱｏＳドメインのガベージコレクションでは、コントローラ４は、このＱｏＳドメインに対応するアクティブブロック群からコピー元ブロック（ＧＣソースブロック）とコピー先ブロック（ＧＣデスティネーションブロック）を選択する。どのブロックをＧＣ候補（コピー元ブロック）として選択するかは、ホスト２によって指定される上述のポリシーに従って決定されてもよいし、ホスト２から指定されても良い。ポリシーも基づく場合には例えば、有効データ量が最も少ないブロックがＧＣ候補（コピー元ブロック）として選択されてもよい。
【０３０３】
図５９は、図５８のライトコマンドに対するレスポンスを示す。なお、図５９中に施されているハッチングも、図９との相違点を示している。
このレスポンスは、論理アドレス、物理アドレス、長さを含む。
論理アドレスは、図７のライトコマンドに含まれていた論理アドレスである。
【０３０４】
物理アドレスは、図７のライトコマンドに応じてデータが書き込まれたＮＡＮＤ型フラッシュメモリ５内の物理記憶位置を示す。本実施形態では、この物理アドレスは、ブロック番号とページ番号との組み合わせではなく、上述したように、ブロック番号とオフセット（ブロック内オフセット）との組み合わせによって指定される。ブロック番号は、フラッシュストレージデバイス３内の全てのブロックの任意の一つを一意に識別可能な識別子である。全てのブロックに異なるブロック番号が付与されている場合には、これらブロック番号を直接使用してもよい。あるいは、ブロック番号は、ダイ番号と、ダイ内ブロック番号との組み合わせによって表現されてもよい。長さは、書き込まれるべきライトデータの長さを示す。この長さ（データ長）は、粒度（Ｇｒａｉｎ）の数によって指定されてもよいし、ＬＢＡの数によって指定されてもよいし、あるいはそのサイズがバイトによって指定されてもよい。
【０３０５】
図６０は、ホスト２とフラッシュストレージデバイス３とによって実行される書き込み動作処理のシーケンスの別の例を示す。具体的には、図２５に示したシーケンスが、フラッシュストレージデバイス３がタイプ＃１－ストレージデバイスとして実現されている場合におけるものであるのに対して、図６０に示すシーケンスは、フラッシュストレージデバイス３がタイプ＃２－ストレージデバイスとして実現されている場合におけるものである。
【０３０６】
ホスト２は、ＱｏＳドメインＩＤ、ＬＢＡ、長さを含むライトコマンドをフラッシュストレージデバイス３に送信する。フラッシュストレージデバイス３のコントローラ４がこのライトコマンドを受信した時、コントローラ４は、ホスト２からのライトデータを書き込むべき書き込み先ブロックおよびこの書き込み先ブロック内の位置を決定する。より詳しくは、コントローラ４は、フリーブロックリストから一つのフリーブロックを選択し、選択したフリーブロックを書き込み先ブロックとして割り当てる（ステップＳ１１）。つまり、この選択されたフリーブロックおよびこの選択されたフリーブロック内の利用可能な

最初のページが、ホスト２からのライトデータを書き込むべき書き込み先ブロックおよびこの書き込み先ブロック内の位置として決定される。もし書き込み先ブロックがすでに割り当てられている場合には、このステップ１２における書き込み先ブロック割り当て処理を実行する必要は無い。すでに割り当てられている書き込み先ブロック内の利用可能な次のページが、ホスト２からのライトデータを書き込むべき書き込み先ブロック内の位置として決定される。

【０３０７】

コントローラ４は、複数のＱｏＳドメインに対応する複数のフリーブロックリストを管理してもよい。あるＱｏＳドメインに対応するフリーブロックリストにおいては、このＱｏＳドメインに対して予約されたブロック群のみが登録されてもよい。この場合、ステップＳ１２では、コントローラ４は、ライトコマンドのＱｏＳドメインＩＤによって指定されるＱｏＳドメインに対応するフリーブロックリストを選択し、この選択したフリーブロックリストから一つのフリーブロックを選択し、この選択したフリーブロックを書き込み先ブロックとして割り当ててもよい。これにより、異なるＱｏＳドメインに対応するデータが同じブロックに混在されてしまうことを防止することができる。

【０３０８】

コントローラ４は、ホスト２から受信されるライトデータを書き込み先ブロックに書き込む（ステップＳ１２）。ステップＳ１２では、コントローラ４は、論理アドレス（ここではＬＢＡ）とライトデータの双方を書き込み先ブロックに書き込む。

コントローラ４は、ブロック管理テーブル３２を更新して、書き込まれたデータに対応するビットマップフラグ（つまり、このデータが書き込まれた物理記憶位置の物理アドレスに対応するビットマップフラグ）を０から１に変更する（ステップＳ１３）。例えば、図２６に示されているように、開始ＬＢＡがＬＢＡｘである１６Ｋバイト更新データがブロックＢＬＫ＃１のオフセット＋４～＋７に対応する物理記憶位置に書き込まれた場合を想定する。この場合、図２７に示されているように、ブロックＢＬＫ＃１用のブロック管理テーブルにおいては、オフセット＋４～＋７に対応するビットマップフラグそれぞれが０から１に変更される。

【０３０９】

コントローラ４は、このライトコマンドに対するレスポンスをホスト２に返す（ステップＳ１４）。例えば、図２６に示されているように、開始ＬＢＡがＬＢＡｘである１６Ｋバイト更新データがブロックＢＬＫ＃１のオフセット＋４～＋７に対応する物理記憶位置に書き込まれたならば、ＬＢＡｘ、ブロック番号（＝ＢＬＫ１）、オフセット（＝＋４）、長さ（＝４）を含むレスポンスがコントローラ４からホスト２に送信される。

【０３１０】

ホスト２がこのレスポンスを受信した時、ホスト２は、ホスト２によって管理されているＬＵＴを更新して、書き込まれたライトデータに対応する論理アドレスそれぞれに物理アドレスをマッピングする。図２８に示されているように、ＬＵＴは、複数の論理アドレス（例えばＬＢＡ）それぞれに対応する複数のエントリを含む。ある論理アドレス（例えばあるＬＢＡ）に対応するエントリには、このＬＢＡに対応するデータが格納されているＮＡＮＤ型フラッシュメモリ５内の位置（物理記憶位置）を示す物理アドレスＰＢＡ、つまりブロック番号、オフセット（ブロック内オフセット）が格納される。図２６に示されているように、開始ＬＢＡがＬＢＡｘである１６Ｋバイト更新データがブロックＢＬＫ＃１のオフセット＋４～＋７に対応する物理記憶位置に書き込まれたならば、図２８に示されているように、ＬＵＴが更新されて、ＬＢＡｘに対応するエントリにＢＬＫ＃１、オフセット＋４が格納され、ＬＢＡｘ＋１に対応するエントリにＢＬＫ＃１、オフセット＋５が格納され、ＬＢＡｘ＋２に対応するエントリにＢＬＫ＃１、オフセット＋６が格納され、ＬＢＡｘ＋３に対応するエントリにＢＬＫ＃１、オフセット＋７が格納される。

【０３１１】

この後、ホスト２は、上述の更新データの書き込みによって不要になった以前のデータを無効化するためのＴｒｉｍコマンドをフラッシュストレージデバイス３に送信する（ステ

ップＳ２１）。図２６に示されているように、以前のデータがブロックＢＬＫ＃０のオフセット＋０、オフセット＋１、オフセット＋２、オフセット＋３に対応する位置に格納されている場合には、図２９に示すように、ブロック番号（＝ＢＬＫ＃０）、オフセット（＝＋０）、長さ（＝４）を指定するＴｒｉｍコマンドがホスト２からフラッシュストレージデバイス３に送信される。フラッシュストレージデバイス３のコントローラ４は、このＴｒｉｍコマンドに応じて、ブロック管理テーブル３２を更新する（ステップＳ１５）。ステップＳ１５においては、図２９に示すように、ブロックＢＬＫ＃０用のブロック管理テーブルにおいて、オフセット＋０～＋３に対応するビットマップフラグそれぞれが１から０に変更される。

【０３１２】

図６１は、フラッシュストレージデバイス３に適用されるＧＣ制御コマンドの別の例を示す。具体的には、図３４に示したＧＣ制御コマンドが、フラッシュストレージデバイス３がタイプ＃１－ストレージデバイスとして実現されている場合において適用されるものであるのに対して、図６１に示すＧＣ制御コマンドは、フラッシュストレージデバイス３がタイプ＃２－ストレージデバイスとして実現されている場合において適用されるものである。図６１中に施されているハッチングは、図３４との相違点を示している。

【０３１３】

ＧＣ制御コマンドは、コマンドＩＤ、ポリシー、ソースＱｏＳドメインＩＤ、デスティネーションＱｏＳドメインＩＤ、等を含んでもよい。

コマンドＩＤはこのコマンドがＧＣ制御コマンドであることを示すＩＤ（コマンドコード）であり、ＧＣ制御コマンドにはＧＣ制御コマンド用のコマンドＩＤが含まれる。

【０３１４】

ポリシーは、ＧＣ候補ブロック（ＧＣソースブロック）を選択するための条件（ＧＣポリシー）を指定するパラメータである。フラッシュストレージデバイス３のコントローラ４は、複数のＧＣポリシーをサポートしている。

コントローラ４によってサポートされているＧＣポリシーには、有効データ量が少ないブロックを優先的にＧＣ候補ブロック（ＧＣソースブロック）として選択するというポリシー（Ｇｒｅｅｄｙ）が含まれてもよい。

【０３１５】

また、コントローラ４によってサポートされているＧＣポリシーには、低い更新頻度を有するデータ（コールドデータ）が集められているブロックを、高い更新頻度を有するデータ（ホットデータ）が集められているブロックよりも優先的にＧＣ候補ブロック（ＧＣソースブロック）として選択するというポリシーが含まれていてもよい。

【０３１６】

さらに、ＧＣポリシーは、ＧＣ開始条件を指定してもよい。ＧＣ開始条件は、例えば、残りフリーブロックの個数を示してもよい。

コントローラ４は、有効データを含むブロック群をアクティブブロックリストによって管理しており、ＧＣを実行する場合には、ＧＣ制御コマンドによって指定されたＧＣポリシーに基づいて、アクティブブロックリストによって管理されているブロック群から一つ以上のＧＣ候補ブロック（ＧＣソースブロック）を選択する。

【０３１７】

ＱｏＳドメインＩＤは、ＧＣを実行すべきＱｏＳドメインを指定するパラメータである。コントローラ４は、ＱｏＳドメインＩＤによって指定されるＱｏＳドメインに属するブロック群、つまりこのＱｏＳドメインに対応するアクティブブロックリストから、一つ以上のＧＣ候補ブロック（ＧＣソースブロック）を選択する。また、コントローラ４は、ＱｏＳドメインＩＤによって指定されるＱｏＳドメインに属するフリーブロック群内の一つ以上のフリーブロックをＧＣデスティネーションブロックとして選択する。

【０３１８】

コントローラ４は、ＱｏＳドメインに対応する残りフリーブロックの数がポリシーによって指定される閾値以下になった場合に、ＧＣを開始してもよい。もしＧＣの強制実行を指

定するポリシーを含むＧＣ制御コマンドを受信したならば、コントローラ４は、ホスト２からこのＧＣ制御コマンドを受信した時にＧＣを即座に開始してもよい。

【０３１９】

図６２は、ガベージコレクション（ＧＣ）動作の手順の別の例を示す。具体的には、図３６に示した手順が、フラッシュストレージデバイス３がタイプ＃１－ストレージデバイスとして実現されている場合におけるものであるのに対して、図６２に示す手順は、フラッシュストレージデバイス３がタイプ＃２－ストレージデバイスとして実現されている場合におけるものである。図６２中に施されているハッチングは、図３６との相違点を示している。

【０３２０】

フラッシュストレージデバイス３のコントローラ４は、ホスト２によって指定されたポリシーに基づいて、ＱｏＳドメインＩＤによって指定されるＱｏＳドメインに属するブロック群から、有効データと無効データとが混在する一つ以上のＧＣソースブロック（コピー元ブロック）を選択する（ステップＳ４１）。次いで、コントローラ４は、ＱｏＳドメインＩＤによって指定されるＱｏＳドメインに属するフリーブロック群から一つ以上のフリーブロックを選択し、選択したフリーブロックをＧＣデスティネーションブロック（コピー先ブロック）として割り当てる（ステップＳ４２）。

【０３２１】

コントローラ４は、ＧＣソースブロック（コピー元ブロック）内の全ての有効データをＧＣデスティネーションブロック（コピー先ブロック）にコピーする（ステップＳ４４）。ステップＳ４４では、コントローラ４は、ＧＣソースブロック（コピー元ブロック）内の有効データのみならず、この有効データとこの有効データに対応する論理アドレスの双方を、ＧＣソースブロック（コピー元ブロック）からＧＣデスティネーションブロック（コピー先ブロック）にコピーする。これにより、ＧＣデスティネーションブロック（コピー先ブロック）内にデータと論理アドレスとのペアを保持することができる。

【０３２２】

そして、コントローラ４は、コピーされた有効データの論理アドレスと、この有効データがコピーされたＧＣデスティネーションブロック（コピー先ブロック）内の位置を示すデスティネーション物理アドレス（ブロック番号、オフセット（ブロック内オフセット））を、ＧＣ用コールバックコマンドを使用してホスト２に通知する（ステップＳ４４）。なお、ステップＳ４４では、コントローラ４は、コピーされた有効データの論理アドレスとデスティネーション物理アドレスとみならず、ソース物理アドレスもホスト２に通知してもよい。

【０３２３】

ホスト２がこのＧＣ用コールバックコマンドを受信した時、ホスト２は、ホスト２によって管理されているＬＵＴを更新して、コピーされた有効データに対応する論理アドレスそれぞれにデスティネーション物理アドレスをマッピングする（ステップＳ５１）。

【０３２４】

図６３および図６４のフローチャートは、フラッシュストレージデバイス３が、ホスト２からのデータを書き込むためのブロックと、フラッシュストレージデバイス３内のデータをコピーするためのブロックとに分離する仕組みを備えている場合における、フリーブロックの割り当ての手順を示す。

【０３２５】

図６３に示す手順は、図６０に示す書き込み動作処理のシーケンス中のステップＳ１１（書き込み先ブロックの割り当て）におけるフリーブロックの割り当ての手順である。

フラッシュストレージデバイス３のコントローラ４は、ホスト２からのライトデータを書き込むためのブロック（書き込み先ブロック）が割り当てられているか否かを判定する（ステップＦ１）。仮に、空きページを含むブロックが割り当てられていても、そのブロックが、フラッシュストレージデバイス３内のデータをコピーするためのブロックであったならば、コントローラ４は、ホスト２からのライトデータを書き込むためのブロックは割

り当てられていないと判定する。この判定は、たとえば、ブロックのメタデータなどとして保持される、そのブロックの用途を示す属性情報に基づいて実行される。

【０３２６】

ホスト２からのライトデータを書き込むためのブロックは割り当てられていないと判定した場合（ステップＦ１：ＮＯ）、コントローラ４は、たとえば同一の仮想ストレージデバイス内のＱｏＳドメイン間で共有されるフリーブロック群の中の１つのフリーブロックを、ホスト２からのライトデータを書き込むためのブロックとして割り当てる（ステップＦ２）。このとき、コントローラ４は、このブロックのメタデータなどとして、ホスト２からのライトデータを書き込むためのブロックであることを示す属性情報を記録する。

【０３２７】

一方、図６４に示す手順は、図６２に示すガベージコレクション（ＧＣ）動作の手順中のステップＳ４２（ＧＣデスティネーションブロックの割り当て）におけるフリーブロックの割り当ての手順である。

フラッシュストレージデバイス３のコントローラ４は、ＧＣソースブロック（コピー元ブロック）内の有効データをコピーするためのブロック（ＧＣデスティネーションブロック）、つまり、フラッシュストレージデバイス３内のデータをコピーするためのブロックが割り当てられているか否かを判定する（ステップＦ１１）。仮に、空きページを含むブロックが割り当てられていても、そのブロックが、ホスト２からのライトデータを書き込むためのブロックであったならば、コントローラ４は、フラッシュストレージデバイス３内のデータをコピーするためのブロックは割り当てられていないと判定する。

【０３２８】

フラッシュストレージデバイス３内のデータをコピーするためのブロックは割り当てられていないと判定した場合（ステップＦ１１：ＮＯ）、コントローラ４は、たとえば同一の仮想ストレージデバイス内のＱｏＳドメイン間で共有されるフリーブロック群の中の１つのフリーブロックを、フラッシュストレージデバイス３内のデータをコピーするためのブロックとして割り当てる（ステップＦ１２）。このとき、コントローラ４は、このブロックのメタデータなどとして、フラッシュストレージデバイス３内のデータをコピーするためのブロックであることを示す属性情報を記録する。

【０３２９】

以上のように、本実施形態のフラッシュストレージデバイス３によれば、Ｉ／Ｏ性能の改善を図ることができる。

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】

【０３３０】

２…ホスト、３…フラッシュストレージデバイス、４…コントローラ、５…ＮＡＮＤ型フラッシュメモリ、２１…ライト動作制御部、２２…リード動作制御部、２３…ＧＣ動作制御部。

【図面】

【図１】

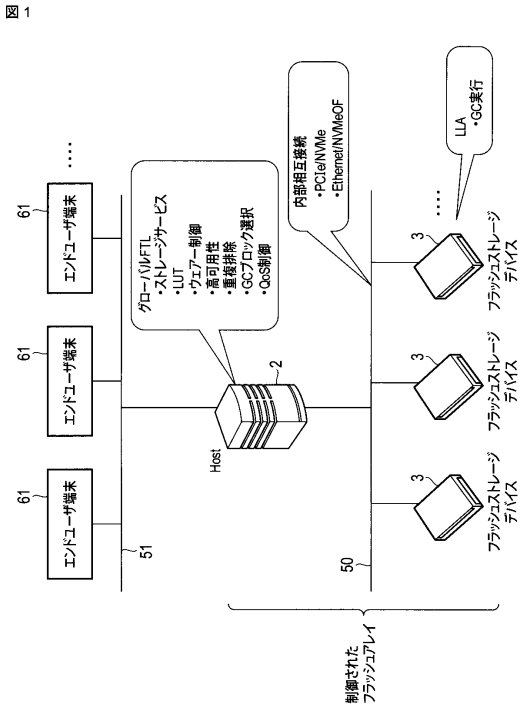


図1
エンドユーザ端末
グローバルFTL
・ストレージサービス
・LUT
・ウェアー制御
・高可用性
・重複排除
・GCブロック選択
・QoS制御
Host
内部相互接続
・PCIe/NVMe
・Ethernet/NVMeOF
LLA
・GC実行
フラッシュストレージデバイス
制御されたフラッシュアレイ

【図２】

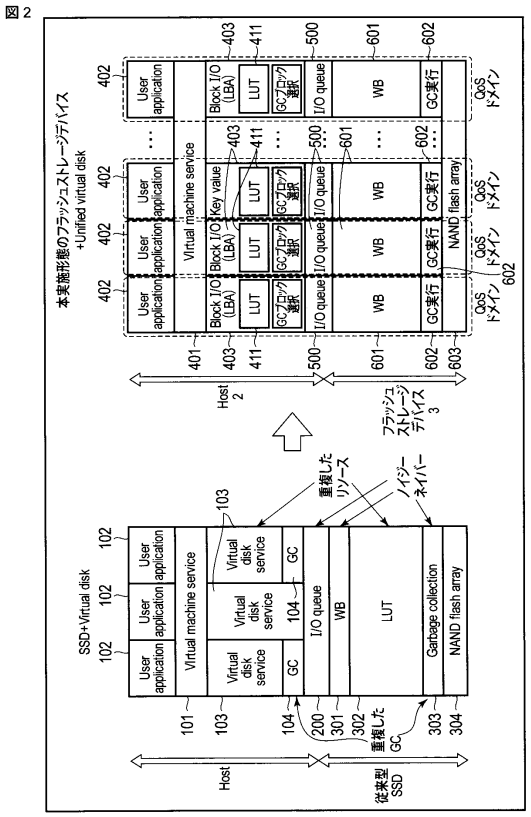


図2
本実施形態のフラッシュストレージデバイス
+Unified virtual disk
User application
Virtual machine service
Block I/O (LBA)
Key value
LUT
GCブロック選択
I/O queue
WB
GC実行
NAND flash array
QoSドメイン
Host
フラッシュストレージデバイス
SSD+Virtual disk
Virtual disk service
GC
I/O queue
WB
LUT
Garbage collection
重複したリソース
ノイジーネイバー
重複したGC
従来型SSD

【図３】

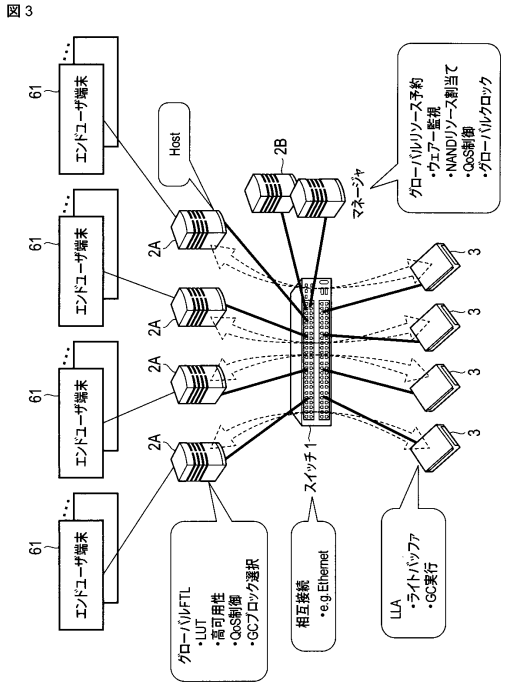


図3
エンドユーザ端末
Host
2B
マネージャ
グローバルリソース予約
・ウェアー監視
・NANDリソース割当て
・QoS制御
・グローバルクロック
スイッチ1
グローバルFTL
・LUT
・高可用性
・QoS制御
・GCブロック選択
相互接続
・e.g Ethernet
LLA
・ライトバッファ
・GC実行

【図４】

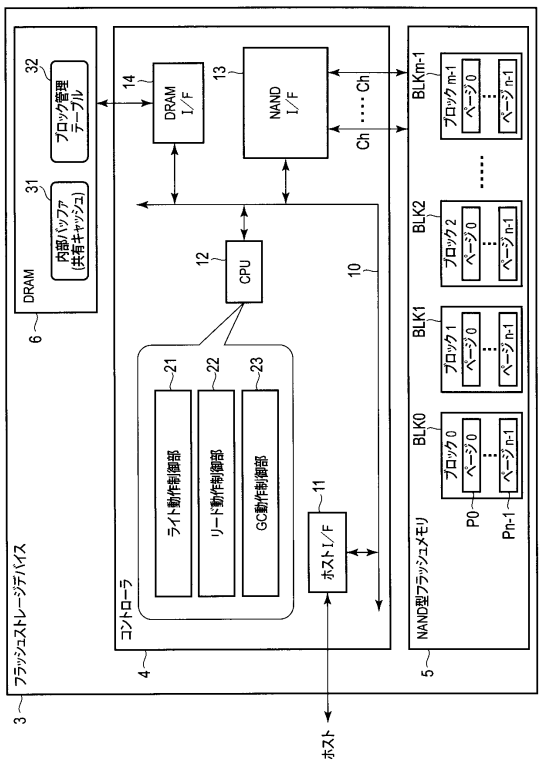
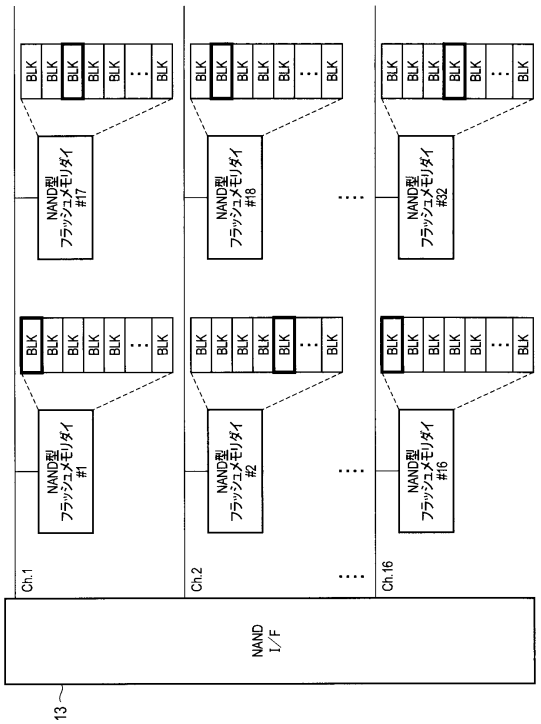


図4
フラッシュストレージデバイス
DRAM
内部バッファ(共有キャッシュ)
ブロック管理テーブル
DRAM I/F
NAND I/F
Ch ・・・・ Ch
CPU
コントローラ
ライト動作制御部
リード動作制御部
GC動作制御部
ホストI/F
NAND型フラッシュメモリ
BLK0
BLK1
BLK2
BLKm-1
ブロック0
ブロック1
ブロック2
ブロックm-1
ページ0
ページn-1
P0
Pn-1
ホスト

【図５】

図５

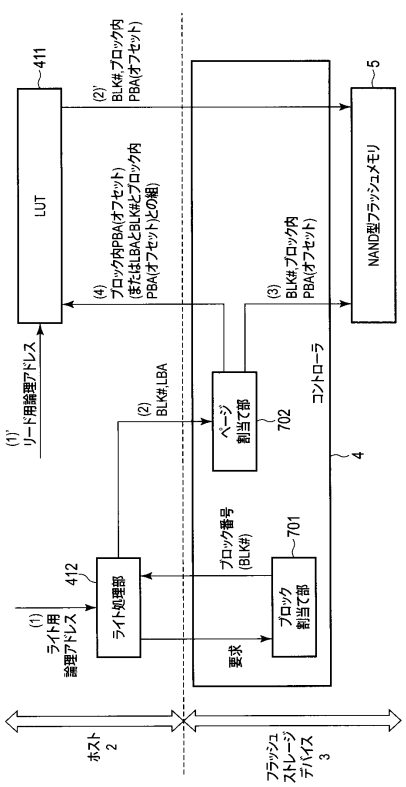


【図６】

図６

【図７】

図７



【図８】

図８

ライトコマンド

	説明
コマンドID	ライトコマンド用のコマンドID
ブロック番号 (BLK#)	データが書き込まれるべきブロックを指定する物理アドレス
論理アドレス	論理アドレスはデータが書き込まれるべき最初の論理位置を示す
長さ	書き込むべきデータのサイズ

【図９】

図 9

ライトコマンドに対するレスポンス

	説明
ブロック内 物理アドレス	ブロック内物理アドレスは、データが書き込まれたブロック内の位置を示す。 ブロック内物理アドレスは、ブロック内オフセットによって指定可能。
長さ	書き込まれたデータの長さ（データ長は、Grainの数によって指定可能）

【図１０】

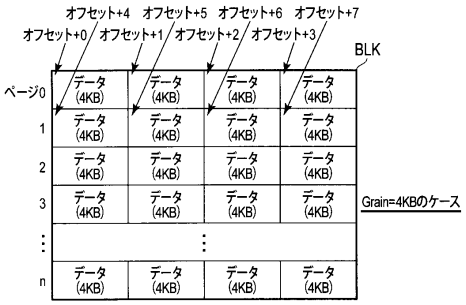
図 10

Trimコマンド（物理アドレスを指定可能）

	説明
コマンドID	Trimコマンド用のコマンドID
物理アドレス	物理アドレスは、無効化すべきデータが格納されている最初の物理記憶位置を示す。物理アドレスは、ブロック番号およびオフセットによって指定可能。
長さ	無効化すべきデータの長さ（データ長は、Grainの数によって指定可能）

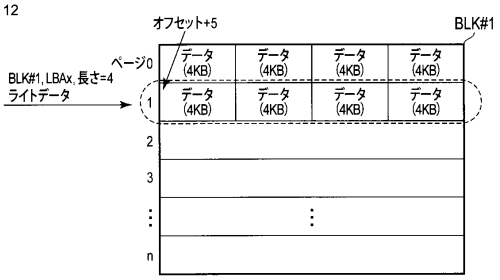
【図１１】

図 11



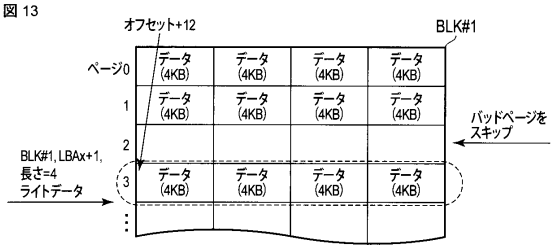
【図１２】

図 12



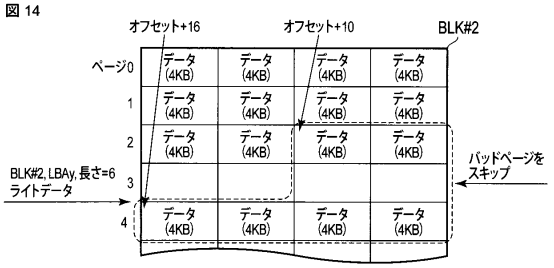
【図１３】

図 13



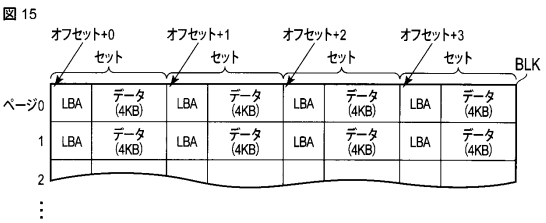
【図１４】

図 14



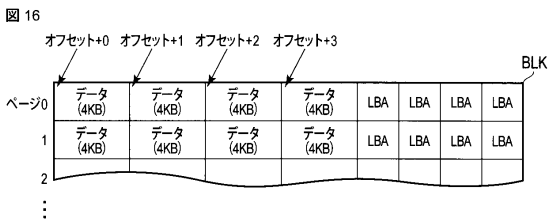
【図１５】

図 15



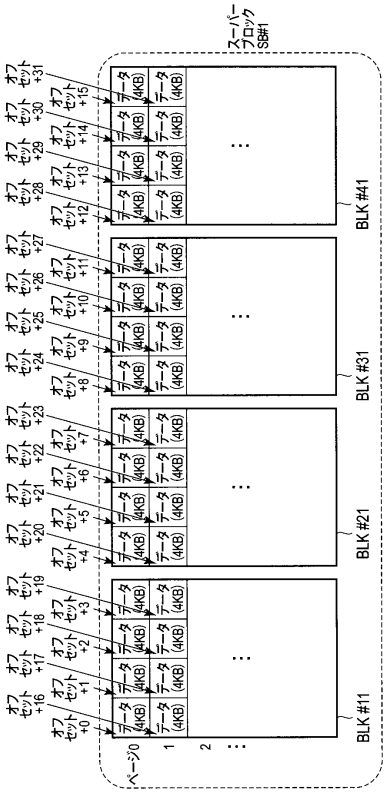
【図１６】

図 16



【図１７】

図 17



【図１８】

図 18

最大ブロック番号ゲットコマンド

	説明
コマンドID	最大ブロック番号ゲット用のコマンドID
パラメータなし	

【図１９】

図 19

最大ブロック番号ゲットコマンドに対するレスポンス

	説明
最大ブロック番号	最大ブロック番号はデバイス内ブロック番号の最大値（つまりデバイス内に存在する利用可能なブロックの数）を示す

【図２０】

図 20

ブロックサイズゲットコマンド

	説明
コマンドID	ブロックサイズゲット用のコマンドID
（オプショナル） ブロック番号	

【図２１】

図 21

ブロックサイズゲットコマンドに対するレスポンス

	説明
ブロックサイズ	ブロック番号が指定されたならば、デバイスは、指定されたブロック番号に対応するブロックのサイズをホストに返す

【図２２】

図 22

ブロックアローケートコマンド

	説明
コマンドID	ブロックアローケート用のコマンドID
	ホストは、フリーブロックを割当てるようにデバイスに要求し、これによってブロック番号をデバイスから取得することができる

【図２３】

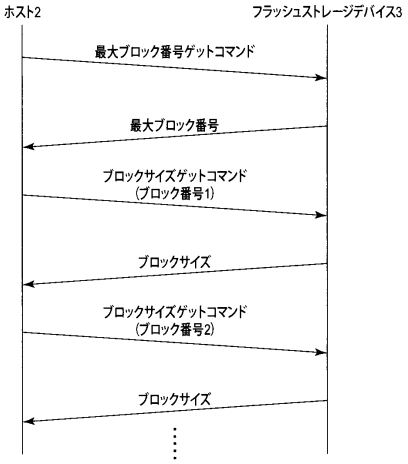
図 23

ブロックアローケートコマンドに対するレスポンス

	説明
ブロック番号	デバイスはフリーブロックリストから、ホストに割当てるべきブロックを選択し、選択されたブロックのブロック番号をホストに返す

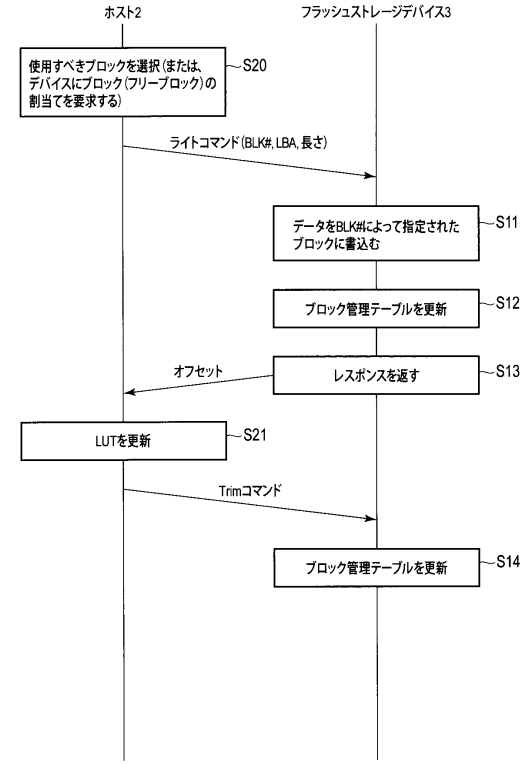
【図２４】

図 24



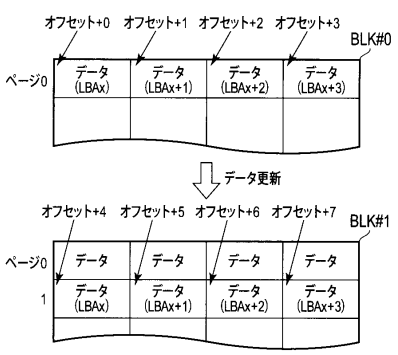
【図２５】

図 25



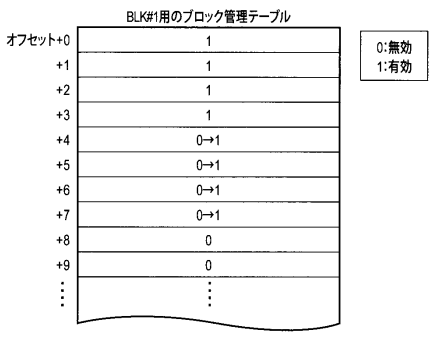
【図２６】

図 26



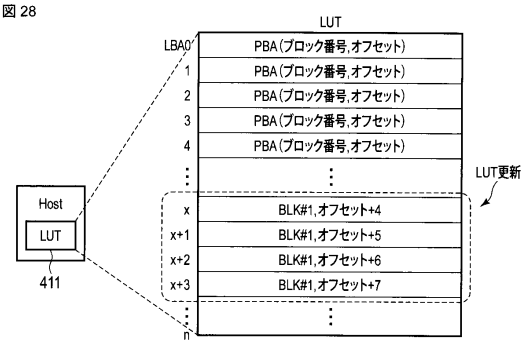
【図２７】

図 27



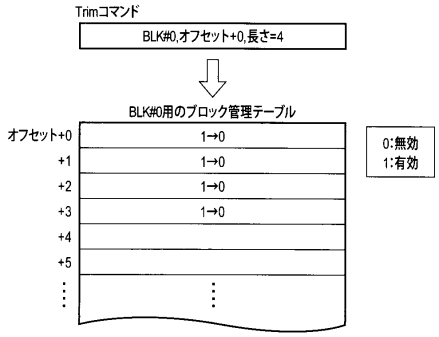
【図２８】

図 28



【図２９】

図 29



【図３０】

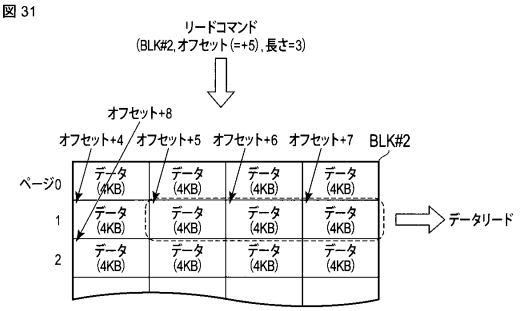
図 30

リードコマンド

	説明
コマンドID	リードコマンド用のコマンドID
物理アドレス(PBA)#1	物理アドレスはデータが読み出されるべき最初の物理記憶位置を示す。 物理アドレスはブロック番号およびオフセットによって指定可能。
長さ#1	リードすべきデータの長さ(データ長はGrainの数によって指定可能)
物理アドレス(PBA)#2	物理アドレスはデータが読み出されるべき最初の物理記憶位置を示す。 物理アドレスはブロック番号およびオフセットによって指定可能。
長さ#2	リードすべきデータの長さ(データ長はGrainの数によって指定可能)
転送先ポインタ	リードされたデータが転送されるべきホストメモリ内の位置

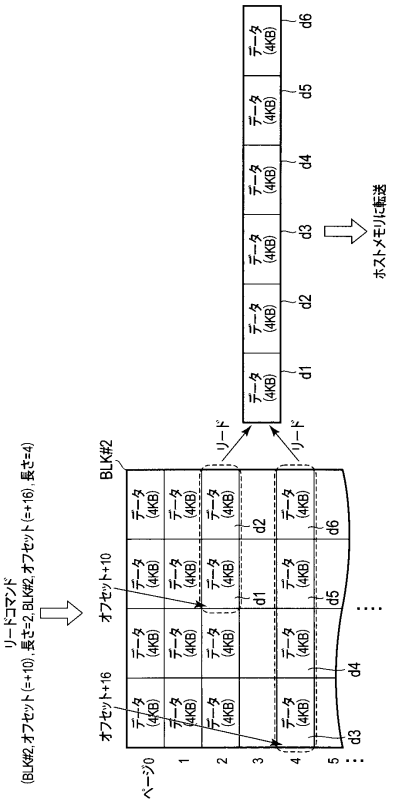
【図３１】

図 31



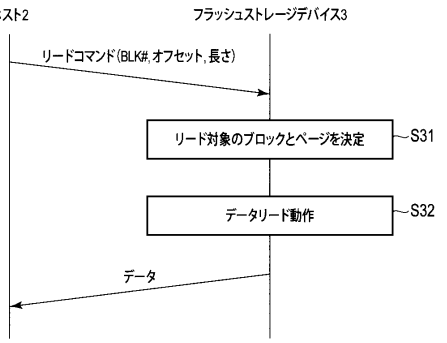
【図３２】

図 32



【図３３】

図 33



【図３４】

図 34

GC制御コマンド

	説明
コマンドID	GC制御用のコマンドID
GCソースブロック番号(BLK#)	ホストは、どのブロックをGCソースブロックとすべきかを指定することができる。ホストは、複数のGCソースブロック番号を指定することができる。
GCデスティネーションブロック番号(BLK#)	ホストは、どのブロックをGCデスティネーションブロックとすべきかを指定することができる。ホストは、複数のGCデスティネーションブロック番号を指定することができる。

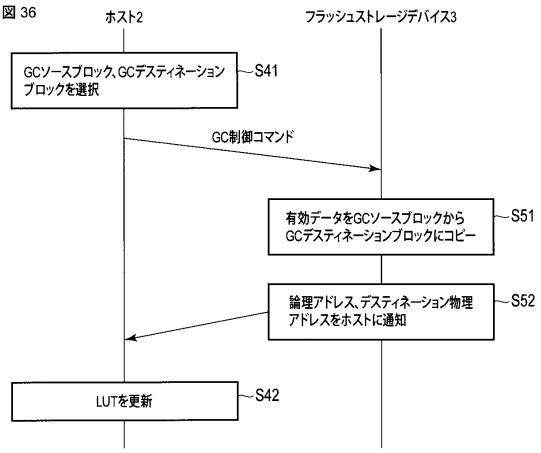
【図３５】

図 35

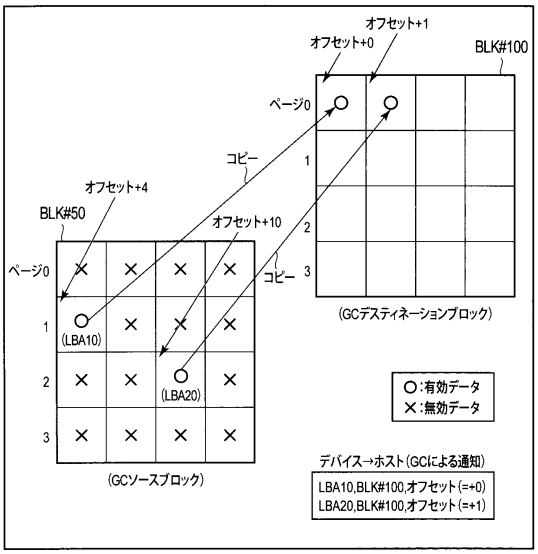
GC用コールバックコマンド

	説明
コマンドID	GC用コールバック用のコマンドID
論理アドレス	コピーされた有効データの論理アドレス
長さ	コピーされた有効データの長さ
デスティネーション物理アドレス	デスティネーション物理アドレスは、有効データがコピーされたGCデスティネーションブロック内の位置を示す。デスティネーション物理アドレスは、ブロック番号およびオフセットによって指定可能。

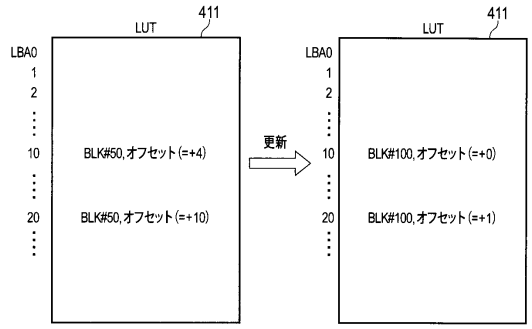
【図３６】



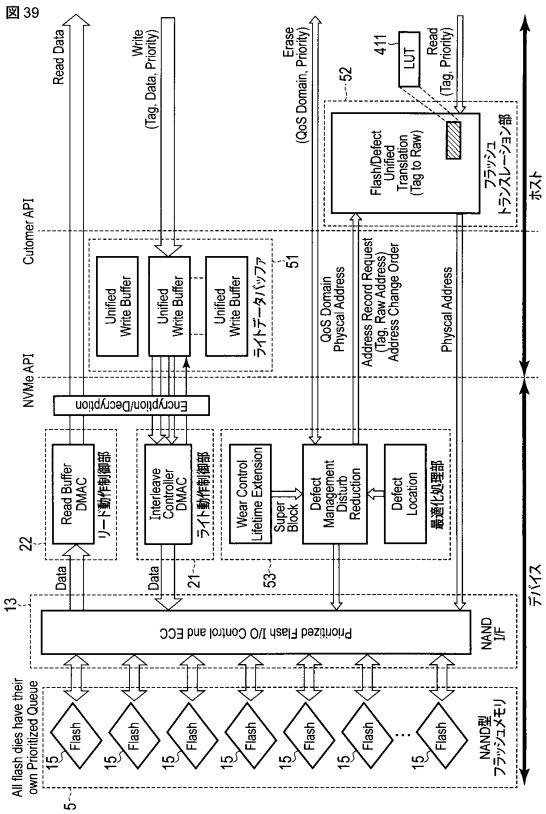
【図３７】



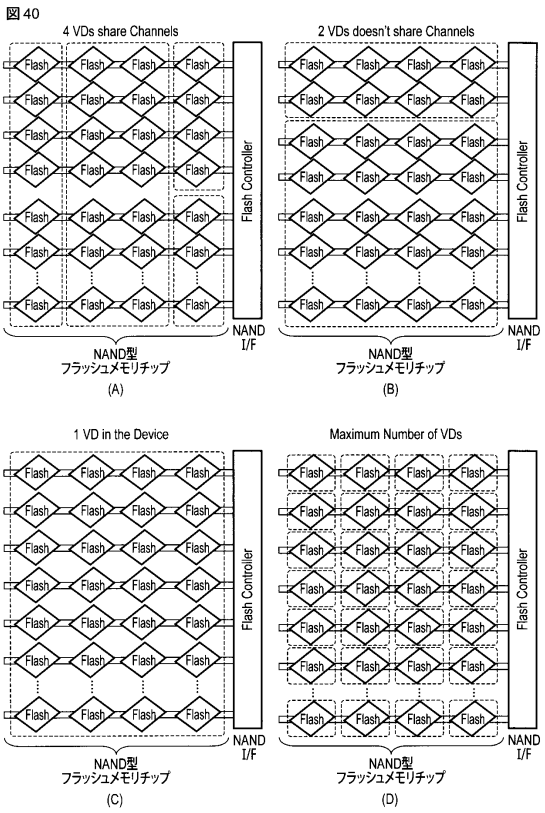
【図３８】



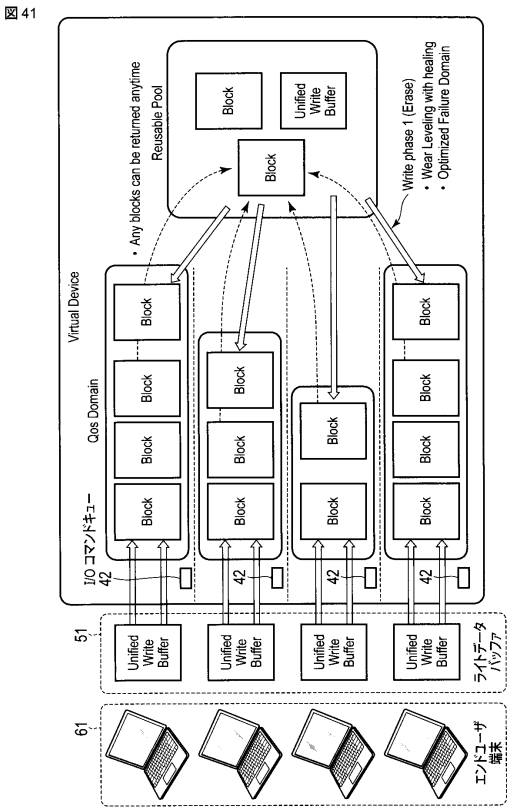
【図３９】



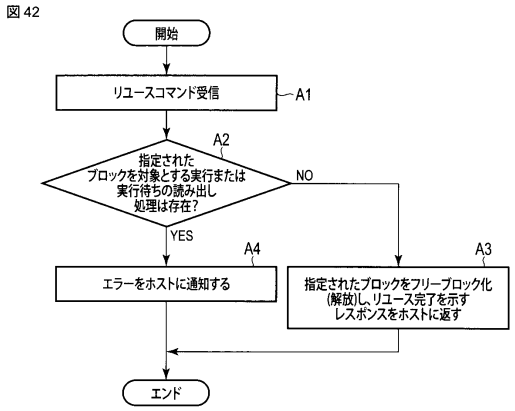
【図４０】



【図４１】

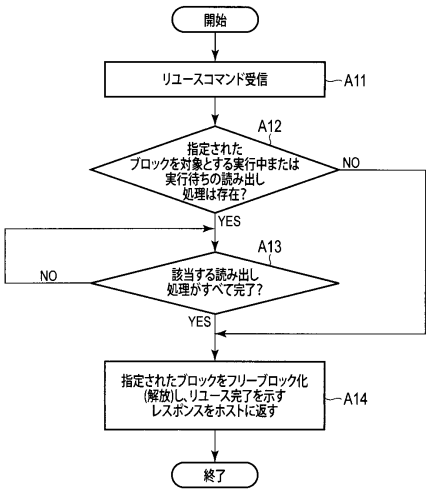


【図４２】



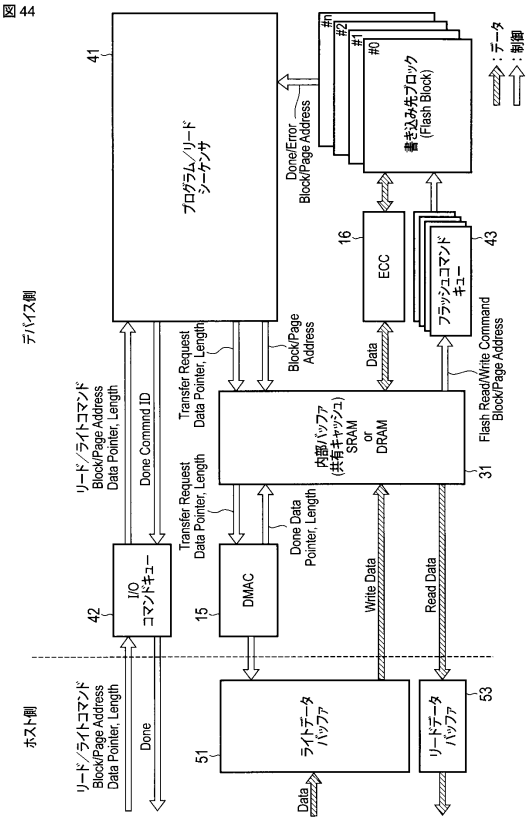
【図４３】

図43



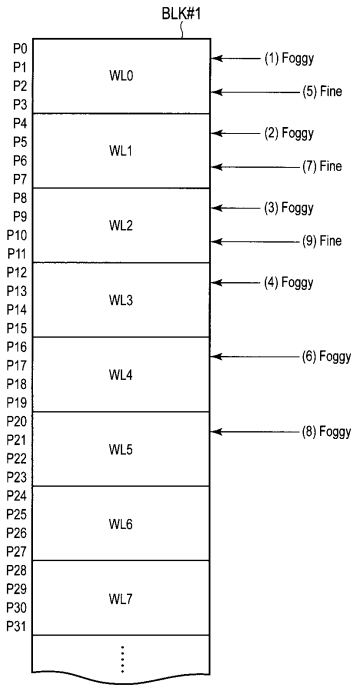
【図４４】

図44



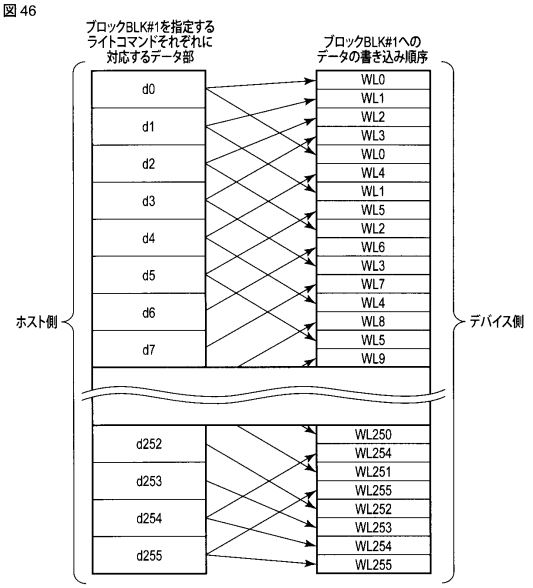
【図４５】

図45



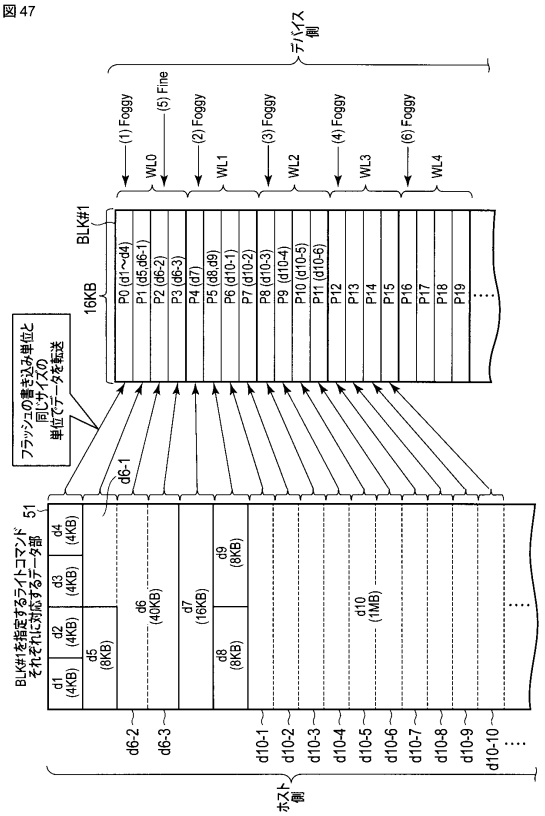
【図４６】

図46



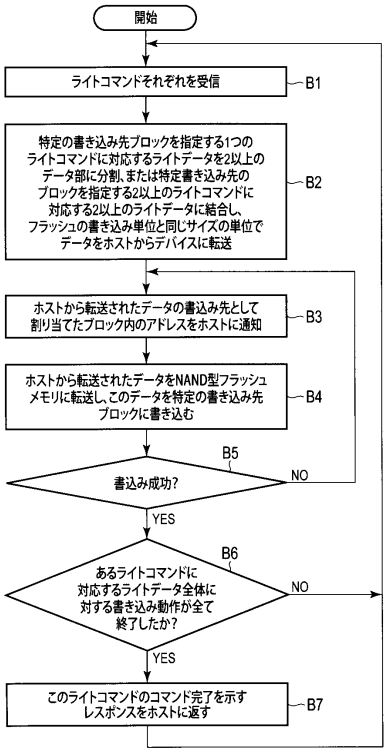
【図４７】

図47



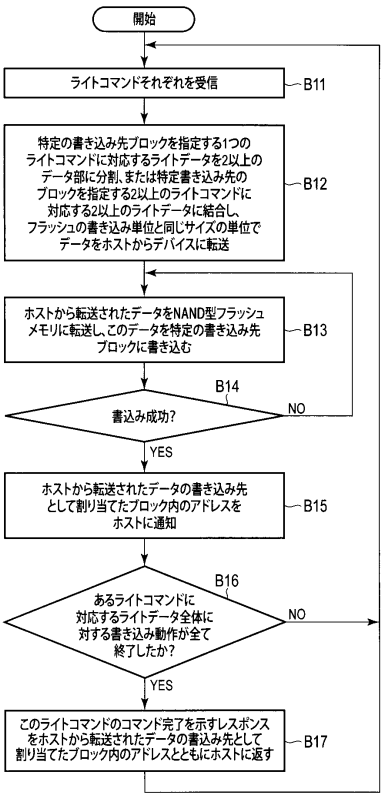
【図４８】

図48



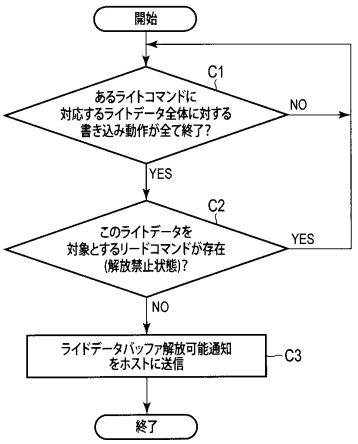
【図４９】

図49



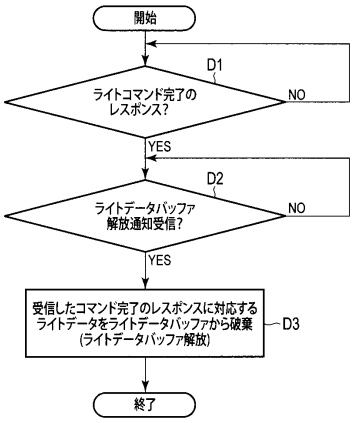
【図５０】

図50



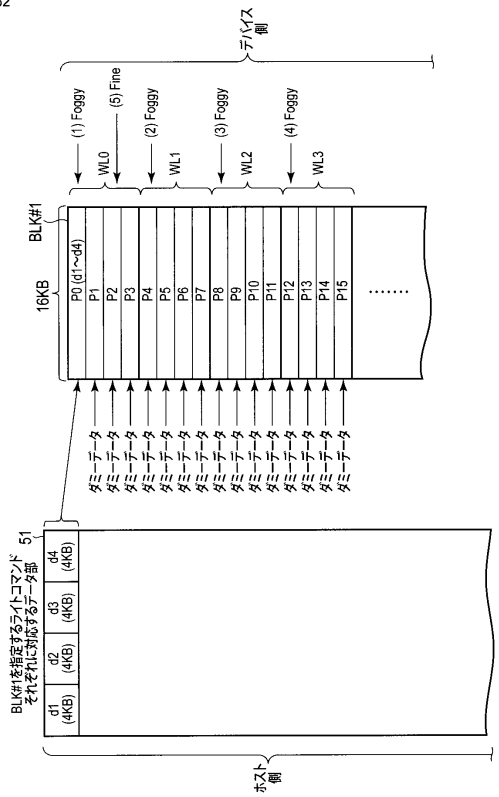
【図５１】

図 51



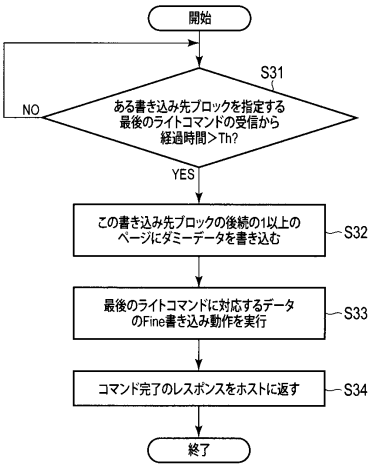
【図５２】

図 52



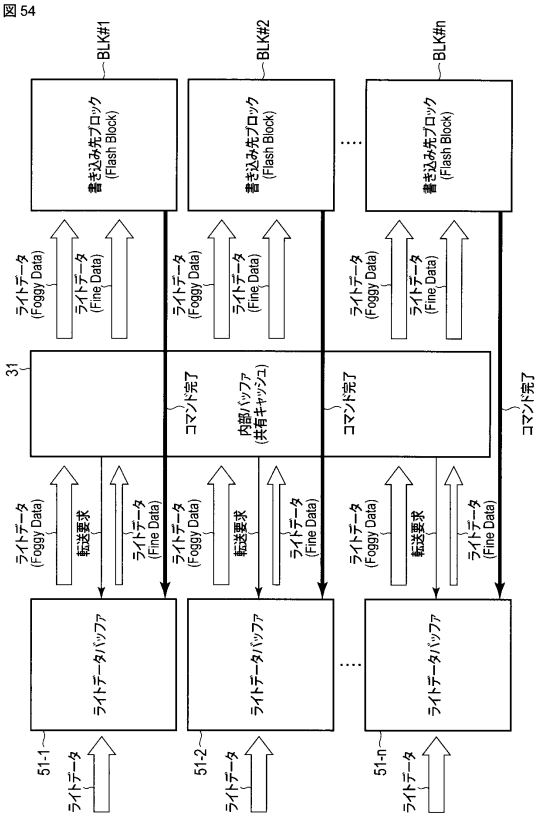
【図５３】

図 53



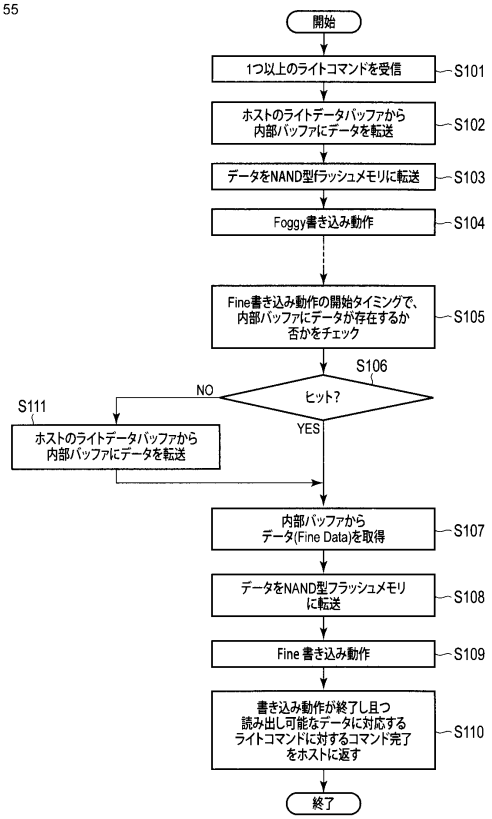
【図５４】

図 54



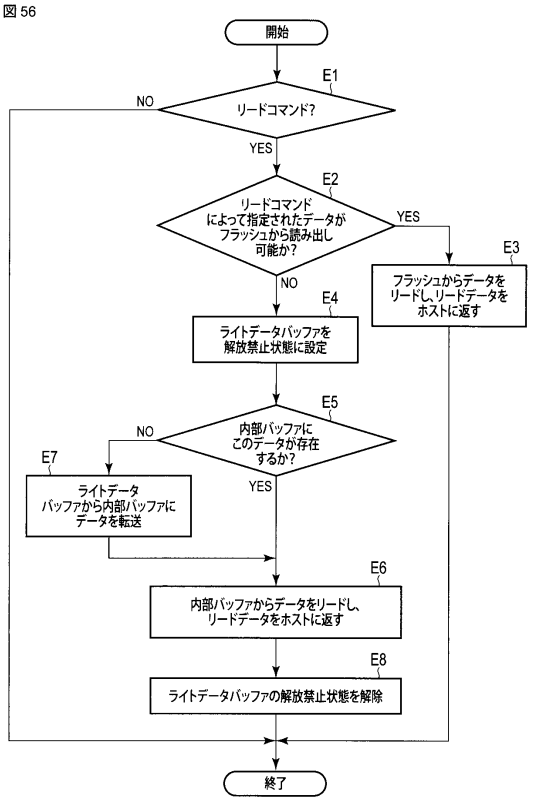
【図５５】

図 55



【図５６】

図 56



【図５７】

図 57
ブロックリユースコマンド

	説明
コマンドID	ブロックリユース用のコマンドID
QoSドメインID	QoSドメインを指定する識別子
ブロックアドレス	フリーブロック化(解放)するブロックを指定する物理アドレス

【図５８】

図 58
ライトコマンド

	説明
コマンドID	ライトコマンド用のコマンドID
QoSドメインID	データが書き込まれるべきQoSドメインを一意に識別するための識別子
論理アドレス	論理アドレスはデータが書き込まれるべき論理位置を示す
長さ	書き込むべきデータの長さ(データ長は、Grainの数によって指定可能)

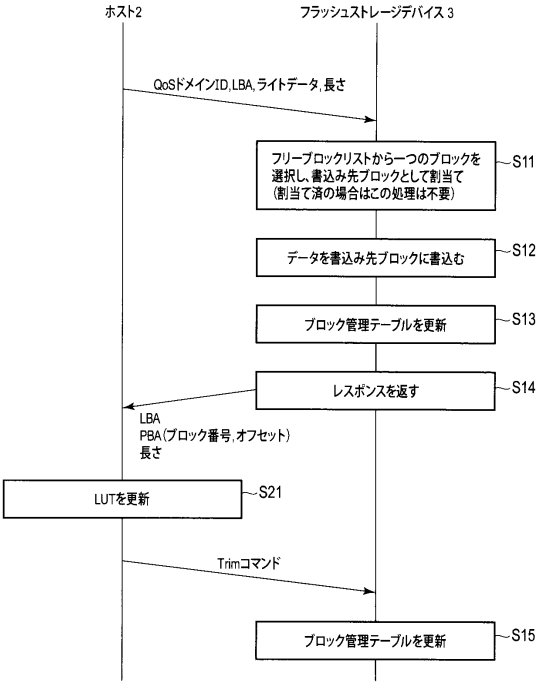
【図５９】

図 59
ライトコマンドに対するレスポンス

	説明
論理アドレス	ライトコマンドに含まれていた論理アドレス
物理アドレス	物理アドレスはデータが書き込まれた物理記憶位置を示す。物理アドレスは、ブロック番号およびオフセットによって指定される
長さ	書き込まれたデータの長さ（データ長は、Grainの数によって指定可能）

【図６０】

図 60



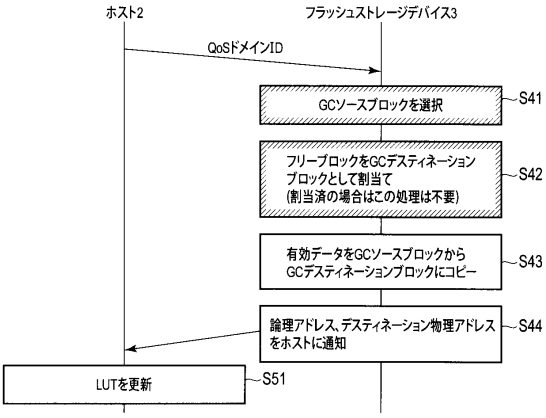
【図６１】

図 61
GC制御コマンド

	説明
コマンドID	GC制御用のコマンドID
ポリシー	ホストは、GC候補ブロックを選択するためのポリシーを指定することができる。デバイスは、複数のポリシーをサポートしており、ホストによって指定されたポリシーを使用してGC候補ブロックを選択する
QoSドメインID	GCを実行すべきQoSドメインを一意に識別するための識別子

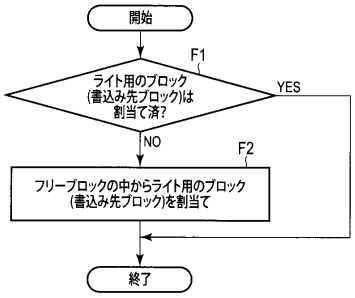
【図６２】

図 62



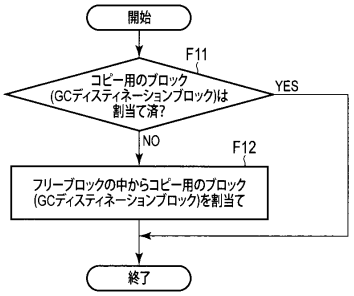
【図６３】

図63



【図６４】

図64



フロントページの続き

(51)国際特許分類　　ＦＩ
Ｇ０６Ｆ　3/06　３０２Ｅ

(56)参考文献　特開２０１７－１６２０６５（ＪＰ，Ａ）
米国特許出願公開第２０１７／０２６２１７５（ＵＳ，Ａ１）
特開２０１８－１４２２３７（ＪＰ，Ａ）
特開２０１３－１０９４１９（ＪＰ，Ａ）

(58)調査した分野　(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　１２／００ － １２／０２
Ｇ０６Ｆ　３／０８
Ｇ０６Ｆ　３／０６