

(19) 日本国特許庁(JP)

(12) 特許公報(B2)

(11) 特許番号

特許第6713934号
(P6713934)

(45) 発行日 令和2年6月24日(2020.6.24)

(24) 登録日 令和2年6月8日(2020.6.8)

| | |
|----------------------|----------------|
| (51) Int.Cl. | F 1 |
| GO6F 12/04 (2006.01) | GO6F 12/04 530 |
| GO6F 3/06 (2006.01) | GO6F 3/06 301N |
| GO6F 3/08 (2006.01) | GO6F 3/08 H |

請求項の数 20 (全 31 頁)

| | | | |
|--------------------|-------------------------------|-----------|---|
| (21) 出願番号 | 特願2017-1652 (P2017-1652) | (73) 特許権者 | 390019839 三星電子株式会社 Samsung Electronics Co., Ltd. 大韓民国京畿道水原市靈通区三星路 129 129, Samsung-ro, Yeongtong-gu, Suwon-si, Gyeonggi-do, Republic of Korea |
| (22) 出願日 | 平成29年1月10日 (2017.1.10) | | |
| (65) 公開番号 | 特開2017-126334 (P2017-126334A) | | |
| (43) 公開日 | 平成29年7月20日 (2017.7.20) | | |
| 審査請求日 | 令和2年1月10日 (2020.1.10) | | |
| (31) 優先権主張番号 | 62/279, 655 | (74) 代理人 | 110000051 特許業務法人共生国際特許事務所 |
| (32) 優先日 | 平成28年1月15日 (2016.1.15) | (72) 発明者 | レザエイ アラッシュ アメリカ合衆国, ノースカロライナ州 27612, ローリー ベイマードライ ブ #207, 4504 |
| (33) 優先権主張国・地域又は機関 | 米国(US) | | |
| (31) 優先権主張番号 | 15/086, 020 | | |
| (32) 優先日 | 平成28年3月30日 (2016.3.30) | | |
| (33) 優先権主張国・地域又は機関 | 米国(US) | | |
| 早期審査対象出願 | | | 最終頁に続く |

(54) 【発明の名称】記憶装置及びその動作方法並びにシステム

(57) 【特許請求の範囲】

【請求項 1】

メモリと、

前記メモリに接続されるメモリインターフェイスを含むコントローラと、を有し、

前記コントローラは、前記メモリに格納された第1データと関連したアドレスに書き込む書込みデータと、前記メモリに記憶された第1差分圧縮 (differential compression) された値とを受信し、

前記書込みデータと前記第1データとに基づいて第2差分圧縮された値を計算し、

前記メモリに前記第2差分圧縮された値を格納し、

前記第1差分圧縮された値の代わりに前記第2差分圧縮された値を参照するために、前記アドレスの関連性を変更し、
10

前記第1差分圧縮された値は、前記第1データのハッシュ関数と圧縮関数に基づいて決定され、

前記第2差分圧縮された値は、前記書込みデータと前記第1データのハッシュ関数と圧縮関数に基づいて決定されることを特徴とする記憶装置。

【請求項 2】

前記コントローラは、前記アドレスと関連する読み出し要請を受信し、

前記第1データを読み出し、

前記第2差分圧縮された値を読み出し、

第2データを形成するために前記第1データと前記第2差分圧縮された値とを結合し、

前記第2データで読み出し要請に応答することを特徴とする請求項1に記載の記憶装置。

【請求項3】

前記コントローラは、前記アドレスと関連する読み出し要請を受信し、

前記第1データを読み出し、

前記第1データで前記読み出し要請に応答することを特徴とする請求項1に記載の記憶装置。

【請求項4】

前記コントローラは、前記アドレスと前記第1差分圧縮された値との関連性を維持し、

前記アドレスと関連する読み出し要請を受信し、

前記第1データを読み出し、

前記第1差分圧縮された値を読み出し、

第3データを形成するために前記第1データと前記第1差分圧縮された値とを結合し、

前記第3データで前記読み出し要請に応答することを特徴とする請求項1に記載の記憶装置。

【請求項5】

前記アドレスは、論理的アドレスとして参照され、

前記コントローラは、前記第1データを読み出し、

前記第2差分圧縮された値を読み出し

第2データを形成するために前記第1データと前記第2差分圧縮された値とを結合し、

前記第2データを前記メモリ内の物理的アドレスに格納し、

前記第2データを格納する前記物理的アドレスを参照するために前記論理的アドレスの関連性をアップデートすることを特徴とする請求項1に記載の記憶装置。

【請求項6】

前記コントローラは、複数のエントリを含む前記メモリ内のマッピングテーブルを維持し、

各々のエントリは、論理的アドレス、物理的アドレス、及び差分圧縮された値の表示を含むよう構成されることを特徴とする請求項1に記載の記憶装置。

【請求項7】

前記メモリは、不揮発性メモリと揮発性メモリを含み、

前記コントローラは、前記第1データを前記不揮発性メモリに格納し、前記第2差分圧縮された値を前記揮発性メモリに格納するよう構成されることを特徴とする請求項1に記載の記憶装置。

【請求項8】

前記コントローラは、前記第2差分圧縮された値を前記揮発性メモリから前記不揮発性メモリに伝送するよう構成されることを特徴とする請求項7に記載の記憶装置。

【請求項9】

前記第1データは、順次書き込み(`in-place writes`)を遂行できないメモリの少なくとも一部分に格納されることを特徴とする請求項1に記載の記憶装置。

【請求項10】

前記コントローラは、前記メモリに格納されたデータと関連しない新しいアドレスを有する書き込み要請を受信し、

前記書き込み要請のデータを前記メモリに書き込み、

前記メモリに書き込まれたデータと前記新しいアドレスとの関連性を生成することを特徴とする請求項1に記載の記憶装置。

【請求項11】

記憶装置の動作方法であって、

メモリに格納された第1データと関連するアドレスに書き込む書き込みデータと、前記メモリに格納された第1差分圧縮された値とを受信する段階と、

前記書き込みデータと前記第1データとに基づいて第2差分圧縮された値を計算する段階と、

10

20

30

40

50

前記メモリに前記第2差分圧縮された値を格納する段階と、
 前記第1差分圧縮された値の代わりに前記第2差分圧縮された値を参照するために、前記アドレスの関連性を変更する段階と、を有し、
前記第1差分圧縮された値は、前記第1データのハッシュ関数と圧縮関数に基づいて決定され、
前記第2差分圧縮された値は、前記書き込みデータと前記第1データのハッシュ関数と圧縮関数に基づいて決定されることを特徴とする記憶装置の動作方法。

【請求項12】

前記アドレスと関連する読み出し要請を受信する段階と、
 前記第1データを読み出す段階と、
 前記第2差分圧縮された値を読み出す段階と、
第2データを形成するために前記第1データと前記第2差分圧縮された値とを結合する段階と、
 前記第2データで前記読み出し要請に対応する段階と、をさらに有することを特徴とする請求項11に記載の記憶装置の動作方法。

【請求項13】

前記アドレスと関連する読み出し要請を受信する段階と、
 前記第1データを読み出す段階と、
 前記第1データで前記読み出し要請に応答する段階をさらに有することを特徴とする請求項11に記載の記憶装置の動作方法。

10

20

【請求項14】
 前記アドレスは論理的アドレスとして参照され、
 前記記憶装置の動作方法は、前記第1データを読み出す段階と、
 前記第2差分圧縮された値を読み出す段階と、
第2データを形成するために前記第1データと前記第2差分圧縮された値とを結合する段階と、
 前記第2データを前記メモリ内の物理的アドレスに格納する段階と、

前記第2データを格納する前記物理的アドレスを参照するために前記論理的アドレスの関連性をアップデートする段階をさらに有することを特徴とする請求項11に記載の記憶装置の動作方法。

30

【請求項15】
 複数のエントリを含む前記メモリ内のマッピングテーブルを維持する段階を、さらに有し、
 各々のエントリは、論理的アドレス、物理的アドレス、及び差分圧縮された値の表示を含むことを特徴とする請求項11に記載の記憶装置の動作方法。

【請求項16】
 前記第1データを不揮発性メモリに格納する段階と、
 前記第2差分圧縮された値を前記揮発性メモリに格納する段階と、をさらに有することを特徴とする請求項1に記載の記憶装置の動作方法。

40

【請求項17】
 前記第2差分圧縮された値を前記揮発性メモリから前記不揮発性メモリに伝送する段階をさらに有することを特徴とする請求項16に記載の記憶装置の動作方法。

【請求項18】
 通信インターフェイスと、
 前記通信インターフェイスを通じてメモリに接続されるプロセッサと、を有し、
 前記プロセッサは、前記プロセッサに接続される記憶装置に格納された第1データと関連するアドレスに書き込む書き込みデータと、第1差分圧縮された値とを受信し、
 前記書き込みデータと前記第1データとに基づいて第2差分圧縮された値を計算し、
 前記第1差分圧縮された値の代わりに前記第2差分圧縮された値を参照するために、前記アドレスの関連性を変更し、

50

前記第1差分圧縮された値は、前記第1データのハッシュ関数と圧縮関数に基づいて決定され、

前記第2差分圧縮された値は、前記書き込みデータと前記第1データのハッシュ関数と圧縮関数に基づいて決定されることを特徴とするシステム。

【請求項19】

前記プロセッサに接続されたメモリをさらに有し、

前記プロセッサは、前記メモリに前記第2差分圧縮された値を格納するよう構成されることを特徴とする請求項18に記載のシステム。

【請求項20】

前記プロセッサは、前記記憶装置に前記第2差分圧縮された値を格納するよう構成されることを特徴とする請求項18に記載のシステム。 10

【発明の詳細な説明】

【技術分野】

【0001】

本発明は記憶装置に関し、さらに詳細にはバージョニング記憶装置 (versioning storage devices) 及びその動作方法に関する。

【背景技術】

【0002】

記憶装置はレイテンシに影響を及ぼす様々な方法として動作することができる。

例えば、データはSSD (Solid State Drive) にページ単位に書き込まれる。 20

ブロックは多数のページからなる。

フラッシュメモリは、ブロック単位に消去動作を遂行することができる。

【0003】

仮に1つのブロックにある様々なページがそれ以上必要でなければ、そのブロックにある他の有効なページを読み出して他のブロックに書き込み、そのブロックをフリーアップ (free up) 状態にする。

その次に、その状態のブロックを消去することができる。

このようなプロセスをガーベッジコレクションと称する。

【0004】

ガーベッジコレクションは、記憶装置のレイテンシを増加させるおそれがある。

特に、SSDは、ガーベッジコレクションを遂行する間に読み出し及び/又は書き込み要請を処理できない。

その結果として、入ってくる読み出し/書き込み要請が、ガーベッジコレクションが終わるまで、遅延されてしまう。

【0005】

いくつかのハードディスクは、SMR (shingled magnetic recording) を使用する。

SMRによって、記憶媒体のトラックをオーバーラップさせることができる。

トラックに記憶されたデータが変更され、そのトラックが再び書き込まれる時、オーバーラップするトラックもまた読み出し、再び書き込まれなければならない。

この追加的な動作は、それが遂行される間にレイテンシを引き起こすという問題がある。

【発明の概要】

【発明が解決しようとする課題】

【0006】

本発明は上記従来の記憶装置における問題点に鑑みてなされたものであって、本発明の目的は、レイテンシ性能を向上させ、メモリセルの消耗を改善して寿命を増加するバージョニング記憶装置及びその動作方法並びにそれを含むシステムを提供することにある。

【課題を解決するための手段】

10

20

30

40

50

【0007】

上記目的を達成するためになされた本発明による記憶装置は、メモリと、前記メモリに接続されるメモリインターフェイスを含むコントローラと、を有し、前記コントローラは、前記メモリに格納された第1データと関連したアドレスに書き込む書込みデータと、前記メモリに記憶された第1差分圧縮(*differentially compressed*)された値とを受信し、前記書込みデータと前記第1データとに基づいて第2差分圧縮された値を計算し、前記メモリに前記第2差分圧縮された値を格納し、前記第1差分圧縮された値の代わりに前記第2差分圧縮された値を参照するために、前記アドレスの関連性を変更し、前記第1差分圧縮された値は、前記第1データのハッシュ関数と圧縮関数に基づいて決定され、前記第2差分圧縮された値は、前記書込みデータと前記第1データのハッシュ関数と圧縮関数に基づいて決定されることを特徴とする。

10

【0008】

上記目的を達成するためになされた本発明による記憶装置の動作方法は、記憶装置の動作方法であって、メモリに格納された第1データと関連するアドレスに書き込む書込みデータと、前記メモリに格納された第1差分圧縮された値とを受信する段階と、前記書込みデータと前記第1データとに基づいて第2差分圧縮された値を計算する段階と、前記メモリに前記第2差分圧縮された値を格納する段階と、前記第1差分圧縮された値の代わりに前記第2差分圧縮された値を参照するために、前記アドレスの関連性を変更する段階と、を有し、前記第1差分圧縮された値は、前記第1データのハッシュ関数と圧縮関数に基づいて決定され、前記第2差分圧縮された値は、前記書込みデータと前記第1データのハッシュ関数と圧縮関数に基づいて決定されることを特徴とする。

20

【0009】

上記目的を達成するためになされた本発明によるシステムは、通信インターフェイスと、前記通信インターフェイスを通じてメモリに接続されるプロセッサと、を有し、前記プロセッサは、前記プロセッサに接続される記憶装置に格納された第1データと関連するアドレスに書き込む書込みデータと、第1差分圧縮された値とを受信し、前記書込みデータと前記第1データとに基づいて第2差分圧縮された値を計算し、前記第1差分圧縮された値の代わりに前記第2差分圧縮された値を参照するために、前記アドレスの関連性を変更し、前記第1差分圧縮された値は、前記第1データのハッシュ関数と圧縮関数に基づいて決定され、前記第2差分圧縮された値は、前記書込みデータと前記第1データのハッシュ関数と圧縮関数に基づいて決定されることを特徴とする。

30

【発明の効果】

【0010】

本発明に係る記憶装置及びその動作方法並びにシステムによれば、記憶装置のレイテンシ性能が向上されることができる。

また、記憶装置内のメモリセルの消耗を改善して寿命を増加させることができる。

【図面の簡単な説明】

【0011】

【図1】本発明の一実施形態による記憶装置を概略的に示すブロック図である。

【図2A】本発明の実施形態による記憶装置に対する書き込みを説明するための概念図である。

40

【図2B】本発明の実施形態による記憶装置に対する書き込みを説明するための概念図である。

【図2C】本発明の実施形態による記憶装置に対する書き込みを説明するための概念図である。

【図2D】本発明の実施形態による記憶装置に対する書き込みを説明するための概念図である。

【図2E】本発明の実施形態による記憶装置に対する書き込みを説明するための概念図である。

【図3A】本発明の実施形態による記憶装置から読み出す動作を説明するための概念図で

50

ある。

【図3B】本発明の実施形態による記憶装置から読み出す動作を説明するための概念図である。

【図3C】本発明の実施形態による記憶装置から読み出す動作を説明するための概念図である。

【図3D】本発明の実施形態による記憶装置から読み出す動作を説明するための概念図である。

【図3E】本発明の実施形態による記憶装置から読み出す動作を説明するための概念図である。

【図4A】本発明の他の実施形態による記憶装置を概略的に示すブロック図である。

10

【図4B】本発明のさらに他の実施形態による記憶装置を概略的に示すブロック図である。

【図5】本発明の実施形態による記憶装置内のページデータサイズを示す図である。

【図6】本発明の実施形態によるSSDを概略的に示すブロック図である。

【図7】本発明の実施形態によるバージョニング記憶システムを概略的に示すブロック図である。

【図8】本発明の実施形態によるサーバーを概略的に示すブロック図である。

【図9】本発明の実施形態による分散システムを概略的に示すブロック図である。

【図10】本発明の実施形態によるデータセンターを概略的に示すブロック図である。

【発明を実施するための形態】

20

【0012】

次に、本発明に係る記憶装置及びその動作方法並びにシステムを実施するための形態の具体例を図面を参照しながら説明する。

【0013】

本発明は、バージョニング記憶装置及びその動作方法に係る。

後述する説明は、当分野に通常的な知識を有する者（以下、当業者）が実施形態を実施できるように提供する。

実施形態及び一般的な原則及び特性に対する多様な変更可能であることは明確である。実施形態は主に具体的な具現で提供する具体的な方法及びシステムの形態として説明する。

30

【0014】

しかし、方法及びシステムは、他の具現で有効に動作することができる。“実施形態”、“一実施形態”、“他の実施形態”のような用語は多重の実施形態のみならず、同一の実施形態を参照することができる。

実施形態は特定の構成要素を具備するシステム及び／又は装置を参照して説明する。

しかし、システム及び／又は装置は図示したことより多いか、又はそれより少ない構成要素を含むことができる。

そして、本発明の技術的・思想を逸脱しない範囲内で構成要素の多様な配置及びタイプの変更が多様に実施することができる。

実施形態は特定の段階を具備する具体的な方法を参照して説明する。しかし、方法及びシステムは異なる及び／又は追加的な段階及び実施形態と一致しない異なる順序を有する段階を具備する他の方法にしたがって動作することができる。

40

したがって、実施形態は図示した具体的な実施形態に限定されなく、説明された原理及び特性と一致する最も広い範囲にしたがう。

実施形態は、特定の構成要素を具備する具体的なシステムを参照して説明する。

本発明の実施形態が他の及び／又は追加的な構成要素及び／又は他の特性を具備するシステムや装置の使用と一貫することは当業者に明確である。

また、当業者は他の方法及びシステムが他の構造と一致するを理解すべきである。また、当業者は方法及びシステムが多重要素を具備するメモリシステム構造の使用に適用されることを理解できる。

50

【0015】

ここで使用する用語、特に添付した請求の範囲に使用される用語は広い意味として解釈されることが理解される。例えば、“含む”“又は“含んでいる”の用語は“含むが、限
定されない”と解釈されなければならない。“具備した”又は“有する”の用語は“少
なくとも具備した”又は“少なくとも有する”の意味として解析されるべきである。

【0016】

図1は、本発明の実施形態による記憶装置を示すブロック図である。

実施形態として、記憶装置100はコントローラ102とメモリ104とを含む。

コントローラ102は記憶装置100の動作を管理するように構成される回路である。

そして、コントローラ102は、汎用目的プロセッサ(general purpose processor)、デジタル信号プロセッサ(DSP; digital signal processor)、ASIC(application specific integrated circuit)、マイクロコントローラ(microcontroller)、プログラム可能なロジック装置(programmable logic device)、離散回路(discrete circuits)、これらのよ
うな装置の組み合わせ等のような要素(components)を含む。
10

【0017】

コントローラ102は、レジスター、キャッシュメモリ、プロセシングコア等のよ
うな内部部分(internal portions)を含む。

そして、コントローラ102は、またアドレス及びデータバスインターフェイス、イン
タラプトインターフェイス等のよ
うな外部インターフェイスを含む。
20

例えば、ただ1つのコントローラ102が記憶装置100内に図示しているが、複数の
コントローラが存在することもできる。

また、バッファ、メモリインターフェイス回路、通信インターフェイス等のよ
うな他の
インターフェイス装置がコントローラ102を内部及び外部要素と接続する記憶装置10
0の一部分になることができる。

【0018】

他の実施形態として、コントローラ102は、記憶装置100が通信できるよ
うにする
回路を含む通信インターフェイスを含むことができる。

例えば、通信インターフェイスには、universal serial bus(USB)、small computer system interface(SCSI)、peripheral component interconnect express(PCIe)、serial attached SCSI(SAS)、para
llel ATA(PATA)、serial ATA(SATA)、NVM Express(NVMe)、universal flash storage(UFS)、F
iber channel、Ethernet、remote direct memory access(RDMA)、Infiniband、or other inter
faces等が含むことができる。
30

【0019】

このような通信インターフェイスを利用して、記憶装置100は、関連する媒体を通じ
て外部装置及びシステムと通信するよ
うに構成することができる。

他の実施形態として、コントローラ102は、通信インターフェイスを通じて読出し及
び書込み要請を受信するよ
うに構成することができる。

【0020】

メモリ104は、データを記憶(格納)できる所定の装置である。

ここで、1つのメモリ104が記憶装置100のために図示しているが、複数のメモリ
が記憶装置100内に含まれることができる。

記憶装置100は、複数の異なるメモリタイプを含むことができる。

【0021】

例として、メモリ104には、dynamic random access mem
50

ory (DRAM)、DDR、DDR2、DDR3、DDR4のような規格にしたがうdouble data rate synchronous dynamic random access memory (DDR SDRAM)、static random access memory (SRAM)、flash memory、spin-transfer torque magnetoresistive random access memory (STT-MRAM)、Phase-Change RAM、nanofloating gate memory (NFGM)、or polymer random access memory (PoRAM)、magnetic or optical mediaのようなnon-volatile memory等を含むことができる。

10

【0022】

メモリ104は、データ106、差分圧縮値 (differentially compressed values; 以下、DCVと記す) 108及びマッピングテーブル110を格納 (以下、「格納する」で記す) するよう構成される。

後述でさらに詳細に説明するが、メモリ104は複数のメモリ装置を含むことができる。

メモリ104に格納されたデータは、多様な方法でそのような装置の間に分布させることができる。

しかし、説明を簡易にするために、ここではデータは1つのメモリ104に格納されることと説明する。

20

【0023】

コントローラ102は、メモリ104に接続されるメモリインターフェイス111を含む。

コントローラ102は、メモリインターフェイス111を通じてメモリ104にアクセスするよう構成される。

メモリインターフェイス111は、命令、アドレス、及び/又はデータバスのためのインターフェイスを含む。

インターフェイスを通じて、コントローラ102とメモリ104とは通信する。

【0024】

メモリ104がコントローラ102とは別に構成されることを説明したが、他の実施形態ではキャッシュメモリやSRAM等のようなメモリ104の一部分がコントローラ102に含まれてもよい。

30

コントローラ102は内部通信バスを含む。

プロセシングコア、外部通信インターフェイス、キャッシュメモリ等のような内部要素は内部通信バスを通じて通信することができる。

【0025】

データ106は、記憶装置100に格納されたデータを示す。

後述でさらに詳細に説明するが、DCV108は、関連するデータ106と結合される時、記憶装置100に格納された現在データを示すデータを意味する。

他の実施形態として、DCV108は対応するデータ106のサイズより少ないサイズを有する値である。

40

例えば、データ106とDCV108とは各々異なるデータサイズを有するページに格納することができる。

例として、データ106のページは8Kバイトのサイズを有する。

これと対照的に、対応するDCV108のためのページのサイズは4Kバイトである。

特定データサイズを例として使用したが、他の実施形態ではデータサイズが異なってよい。

【0026】

コントローラ102は、メモリ104に格納されたデータ106と関連するアドレスに書き込む書き込みデータと書き込み要請を受信するよう構成される。

50

マッピングテーブル 110 は、論理的アドレス、物理的アドレス、DCV「アドレス又は値」等のような情報を有するエントリを含み、「アドレス又は値」又は他の情報間の関連性を作成する。

マッピングテーブル 110 は、ページ、ブロック、又はハイブリッドマッピング方策を使用することができる。

しかし、ここで例は説明のためにブロックマッピングを使用する。

【0027】

コントローラ 102 は、マッピングテーブル 110 に格納されたデータを使用して、物理的アドレス、DCV108、又は論理的アドレスと関連されることを識別する。

例えば、通信インターフェイスを通じて論理的アドレスと共に読み出し又は書き込み要請を受信した後に、コントローラ 102 はマッピングテーブル 110 をアクセスして論理的アドレスと関連するエントリを読み出すように構成する。

実施形態として、コントローラ 102 は、マッピングテーブル 110 を格納する内部キヤッショメモリにアクセスするよう構成される。

しかし、他の例として、コントローラ 102 は、DRAMのような外部メモリにアクセスするよう構成されてもよい。

【0028】

コントローラ 102 は、論理的アドレスと関連する物理的アドレスでメモリ 104 に格納されたデータ 106 を読み出すよう構成される。

コントローラ 102 は、書き込み要請に含まれた書き込みデータ及び物理的アドレスから読み出したデータに基づいて DCV108 を計算するよう構成される。

この計算は物理的アドレスに格納されたデータ 106 と入ってくる書き込みデータ間の差分に基づいて DCV108 を生成する。

また、この計算は物理的アドレスから読み出したデータ及び/又は書き込みデータより小さいサイズを有する DCV108 を生成する。

【0029】

DCV は、多様な方法で計算することができる。

実施形態として、コントローラ 102 は、ソフトウェア又は内部回路内で DCV を計算するよう構成することができる。

DCV 機能 (109-1) はコントローラ 102 のこのような動作を示す。

即ち、コントローラ 102 が書き込みデータ及び物理的アドレスから読み出したデータ 106 を受信した後に、コントローラ 102 は書き込みデータ及びメモリ 104 からの読み出しデータを使用して数学的な計算を遂行することによって、DCV を生成するよう構成される。

【0030】

他の例として、コントローラ 102 は、外部回路を使用して DCV を計算することができる。

例えば、コントローラ 102 は、書き込みデータ及び読み出しデータを DCV 回路 (109-2) に直接接続する。

そして、コントローラ 102 は、その応答により DCV を生成する。

DCV 回路 (109-2) は、演算ユニット (arithmetic units)、ルックアップテーブル (lookup tables)、入出力バッファ (input/output buffers) 等を含むことによって、DCV を計算し、コントローラ 102 及び/又はメモリ 104 とインターフェイスする。

【0031】

DCV 回路 (109-2) は、コントローラ 102 の DCV 機能 (109-1) に対する代替 (alternative) や結合 (conjunction) として使用することを示すように点線にて示してある。

DCV 回路 (109-2) を通じて直接又は間接的にコントローラ 102 によって遂行される多様な機能は DCV を生成するよう使用される。

10

20

30

40

50

【0032】

実施形態として、機能は簡単な引き算動作 (subtraction operation) である。

他の例として、その機能はより複雑なハッシュ関数であってもよい。

他の例として、記録はビットがフリップすることを示すように生成することができる (records can be created indicating which bits have flipped)。

実施形態として、その機能は “diff” 関数として称される。

特別な実施形態として、その機能は DCV のサイズを減らすために最適化され得る。

実施形態として、その機能は差分のサイズを減らすために圧縮関数 (compression function) を含むことができる。 10

【0033】

コントローラ 102 は、また DCV 回路 (109-2) を通じて直接又は間接的に DCV 関数の逆 (inverse) である関数を遂行するよう構成され得る。

逆 DCV 関数は、ソースデータ及び他のデータから生成されたソースデータ及び DCV を入力として使用して、他のデータを再生成する関数である。

したがって、データ 106 及び DCV 108 を維持することによって、他のデータは逆 DCV 関数を通じて利用する。

【0034】

コントローラ 102 は、メモリ 104 に計算された DCV を格納するよう構成される。 20

例えば、コントローラ 102 は、DCV 108 と共に計算された DCV をメモリ 104 に格納することができる。

しかし、後述にてさらに詳細に説明するが、他の実施形態として、DCV は DCV 108 と共に記憶される前に、メモリ 104 の 1 つ又はそれ以上の他の部分にキャッシュされることもできる。

【0035】

コントローラ 102 は、アドレスの関係性を変更させて、アドレスと関連する以前 DCV の代わりに計算された DCV を参照することができる。

後述にてさらに詳細に説明するが、アドレスと関連するデータ 106 が読み出される時、新しい DCV は以前 DCV の代わりにアクセスすることができる。 30

他の例として、単なる 1 つの DCV がアドレスと関連するデータを示すように維持することができる。

【0036】

実施形態として、メモリ 104 の少なくとも一部分は、書き込みの観点から、非対称性能 (asymmetric performance) を有することができる。

例えば、フラッシュメモリベースの記憶装置は「in-place」書き込みを許容しない。

新しいブロックは書き込みのために割り当てされなければならない。

そして、次の書き込み (future writes) を準備するために、以前ブロックは削除されなければならない。

SMR (shingled magnetic recording) を有する記憶装置で、他のトラックと重なるトラックに書き込むことは、重なるトラック (overlapping track) を再び書き込むことを含む。

後述にてさらに詳細に説明するが、DCV を使用することによって、非対称性能の影響は減らされる。

【0037】

図 2A ~ 図 2E は、本発明の実施形態による記憶装置に対する書き込みを説明するための概念図である。

図 1 の記憶装置 100 が例として使用される。

図 1 及び図 2A を参照すると、実施形態として、マッピングテーブル 210 はメモリ 1

50

0 4 に格納されたマッピングテーブル 2 1 0 に対応する。

マッピングテーブル 2 1 0 は複数のエントリ 2 1 1 を含む。

各々のエントリ 2 1 1 は、論理的ブロックアドレス (L B A) 、物理的ブロックアドレス (P B A) 、及び D C V の表示 (i n d i c a t i o n) のためのフィールドを含む。

【 0 0 3 8 】

特定フィールドを例として使用したが、他の実施形態として、他のフィールドが存在することもでき、そのフィールドが他の形態を取ってもよい。

例えば、ここで、論理的及び物理的アドレスはブロックとされるが、他の実施形態では、論理的及び物理的アドレスがブロック内のページ又はメモリ 1 0 4 の他の構造と関連することもできる。

他の例として、物理的アドレスのようなシングルアドレス (s i n g l e a d d r e s s) は現在唯一のアドレス (t h e o n l y a d d r e s s p r e s e n t) である。

【 0 0 3 9 】

マッピングテーブル 2 1 0 で、2 つのエントリ (2 1 1 - 1 、 2 1 1 - 2) は以前から存在したエントリである。

新しい論理的ブロックアドレスに書き込み要請 (2 0 2 - 1) が入力される時、新しいエントリ (2 1 1 - 3) がコントローラ 1 0 2 によって生成される。

この例では、新しいエントリ (2 1 1 - 3) は、論理的ブロックアドレス「 1 」のためのものである。コントローラ 1 0 2 は論理的ブロックアドレス 1 と物理的ブロックアドレス 2 3 とを関連させる。

【 0 0 4 0 】

しかし、物理的ブロックアドレスと関連する唯一のデータは、入力されるデータ D である。

例えば、データ D は新しいファイルと関連付けされ、 D C V は計算されない。

即ち、所定の有効データもデータメモリ (2 0 4 - 1) 内の物理的アドレス「 2 3 」に存在しない。

D C V フィールドにある ' x ' 表記は D C V が存在しないか、或いは有効でない表示を示す。

実施形態として、フラッグ (f l a g) は D C V が存在するか、或いは有効であるかを示す。

他の例として、特定アドレス / フィールド値は、 D C V が存在しないか、或いは有効でない表示として定義することができる。

その他の例として、 D C V フィールドはエントリ (2 1 1 - 3) の一部ではないこともあり得る。 D C V が存在しないか、或いは有効でない表示は他の形態を取ることもあり得る。

【 0 0 4 1 】

書き込み要請 (2 0 2 - 1) が新しいエントリ (2 1 1 - 3) と関連付けされるので、コントローラ 1 0 2 はデータ D をデータメモリ (2 0 4 - 1) の物理的ブロックアドレス「 2 3 」に格納する。

データメモリ (2 0 4 - 1) は、データ 1 0 6 が格納されるメモリ 1 0 4 の部分を示す。

したがって、新しい有効エントリ (2 1 1 - 3) は、データメモリ (2 0 4 - 1) に格納されたデータを参照して生成される。

データ D がデータメモリ (2 0 4 - 1) に書き込まれることとして説明したが、他の実施形態として、多様なバッファーリング又はキャッシング等がデータ D をデータメモリ (2 0 4 - 1) に伝送する前に書き込むパート (p a r t) として動作することもできる。

【 0 0 4 2 】

図 1 及び図 2 B を参照すると、実施形態として、新しい書き込み要請 (2 0 2 - 2) を受信する。

再び、書込み要請（202-2）が論理的ブロックアドレス「1」に直接的に向かう（directed towards）。

しかし、エントリ（211-3）が存在することによって、データは既にデータメモリ（204-1）の関連する物理的ブロックアドレスに格納されている。

即ち、図2AのデータDが格納されている。

したがって、この書込み要請（202-2）は、論理的ブロックアドレス「1」に格納されたデータを新しいデータD'でアップデートする。

【0043】

データDの元に書込みと対照的に、図2Aと関連して上記で説明したように、データD'はデータメモリ（204-1）の物理的ブロックアドレス「23」に書き込まれない。

10

コントローラ102はエントリ211がマッピングテーブル210に存在するかを決定する。

実施形態として、コントローラ102は、書込み要請の論理的ブロックアドレスと存在するエントリ211の論理的アドレスを比較する。

仮に合致（match）が発見されれば、以前書込みが発生し、データがデータメモリ（204-1）に存在する。

コントローラ102は、関連する物理的ブロックアドレスに格納されたデータDを読み出す。

ここで、コントローラ102は、物理的ブロックアドレス「23」から読み出す。

図2Aに示したように以前に書き込まれたデータDはデータメモリ（204-1）から読み出される。

20

【0044】

書込み要請（202-2）の新しいデータD'及びデータメモリ（204-1）からの存在するデータDはDCV機能（208-1）の入力として使用される。

DCV機能（208-1）は、上記で説明したように、DCVを計算するために、コントローラ102によって遂行される動作を示す。

DCV機能（208-1）は、新しいデータD'及び存在するデータDに基づいてDCV'を生成する。

ここで、DCV'はデータDと結合されてデータD'を生成することを示すためにアポストロフィを付けた。

30

コントローラ102は、DCVメモリ（204-2）内に新しいDCV'を格納する。

実施形態として、コントローラ102はDCVアドレスをマッピングテーブル210にあるエントリ（211-3）のDCVフィールドに格納する。

ここで、「0x12」の値はその値が新しいか、変更されたことを示すために、アンダーラインを付けた。

【0045】

他の例として、DCVメモリ（204-2）は使用されなく、DCV'はエントリ（211-3）のDCVフィールドに格納することができる。

例えば、DCVは相対的に小さく、それでエントリ211にDCVを記憶できるメモリの追加的な量は相対的に小さくなる。

40

それにも拘らず、エントリ（211-3）は、現在論理的ブロックアドレス、物理的ブロックアドレス、及びDCVを結合（association）する。

【0046】

図1及び図2Cを参照すると、実施形態として、他の書込み要請（202-3）がコントローラ102によって入力される。

この書込み要請（202-3）は、論理的ブロックアドレス「1」に新しいデータD"を書き込むための要請である。

再び、コントローラ102はマッピングテーブル210で合致（match）を見つけるとする。

ここで、エントリ（211-3）が存在する。

50

図2Bと同様に、コントローラ102は、エントリ(211-3)で表示された物理的ブロックアドレス「23」からデータDを読み出す。

データDは、データメモリ(204-1)から出力される。

データメモリ(204-1)から出力された、書き込み要請(202-3)の新しいデータD"と、存在するデータDはDCV機能(208-1)に対する入力として使用される。

【0047】

DCV機能(208-1)は、新しいデータD"と存在するデータDに基づいて新しいDCV"を生成する。

特に、DCV"の生成は、図2Bで書き込まれたデータD'の中間状態(interm 10 ediate state)に関与しない。

データDが物理的ブロックアドレス「23」でデータメモリ(204-1)に格納されたデータであるので、関心の差(difference of interest)はデータDと書き込まれるデータD"との間の差である。

コントローラ102は、結果であるDCV"をDCVメモリ(204-2)に格納する。

コントローラ102は、エントリ(211-3)のDCVフィールドにあるDCVアドレスを使用して、予め存在するDCV'を新しいDCV"で重なって書き込む。

したがって、現在エントリ(211-3)にあるDCVフィールドは新しいDCV"を参照する(references)。

【0048】

図1及び図2Dを参照すると、その動作は図2Cで説明したものと類似する。

しかし、実施形態として、DCV"をDCVメモリ(204-2)に書き込めば、新しいDCVアドレスが生成される。

例えば、仮にDCVメモリ(204-2)がフラッシュメモリで具現されれば、新しいDCV"はDCV'を格納するページと異なる、DCVメモリ(204-2)にあるページに書き込まれる。

新しいDCVアドレスはDCVメモリ(204-2)にあるアドレスであり、DCV"はそこに格納される。エントリ(211-3)で、新しいアドレスは「0x21」で表示され、DCVフィールドで変更を示すようにアンダーラインを付けた。

【0049】

DCVメモリ(204-2)にある同一アドレスでメモリ位置(memory location)をアップデートするか、或いはエントリ(211-3)を新しいアドレスにアップデートすることが、論理的ブロックアドレスとDCVの関連性(association)を変更する方法の例として使用して新しいDCVを参照することにも拘らず、その関連性においての変更が異なる。

例えば、実施形態として、コントローラ102は、DCVフィールドのような、マッピングテーブルにDCV"を格納する。

エントリ(211-3)に格納されたDCV'はDCV"と取り替える。

【0050】

図1及び図2Eを参照すると、実施形態として、コントローラ102は、DCVフラッシュアクセス(202-4)を受信する。

ここで、DCVフラッシュアクセス(202-4)は、上記で説明した各図と同様に、論理的ブロックアドレス「1」を示す。

マッピングテーブル210は、図2Dのマッピングテーブル210と同様に、初期状態(210-1)にある。

再び、コントローラ102はエントリ(211-3)にアクセスし、データメモリ(204-1)にある関連する物理的ブロックアドレス「23」からデータDを読み出す。

しかし、コントローラ102は、DCVフィールドを使用して、DCVメモリ(204-2)に格納されたDCV"にアクセスする。

10

20

30

40

50

ここで、コントローラ 102 は、DCV アドレスを使用して DCV メモリ (204-2) にアクセスする。

【0051】

コントローラ 102 は、DCV" 及びデータ D を逆 DCV 機能 (208-2) に提供する。

逆 DCV 機能 (208-2) は、上記で説明したように、データと DCV とを組み合わせてデータアップデートされたバージョンを作るための関数を示す。

ここで、逆 DCV 機能 (208-2) は、データ D と DCV" とを使用してデータ D" を再生成する。

コントローラ 102 は、データ D の代わりにデータメモリ (204-1) にデータ D" を格納する。 10

マッピングテーブル 210 は、状態 (210-2) にアップデートされる。

ここで、エントリ (211-3) は、有効 DCV が存在しないことを示すようにアップデートされる。

したがって、論理的ブロックアドレス「1」に連続された書き込みが、図 2B で説明したように、コントローラ 102 によって取り扱われる。

【0052】

上記で説明した動作の結果として、頻繁に変更されたデータは非対称性能を有するメモリ 104 を有する記憶装置の性能において影響を与える。

例えば、持続的に変更されたデータは、全体データセットの 5% 未満を含み得る。 20

例として 200GB を使用すれば、単なる 1% 又は 2GB が持続的にアップデートされる。

DCV のより小さいサイズは書き込まれるデータの量を減少させる。

実施形態として、大部分の DCV は全体データブロックのサイズの 20% である (a majority of DCV may be on the order of 20% of a size of an entire block of data)。

【0053】

余りの大部分は相変わらず、データのサイズの 50% より小さい。

したがって、200GB の例で、DCV の 400MB ないしは 1GB が書き込まれる。 30

減られたサイズは空間を効率的に作り、浪費を減らす。

特に、2GB の新しく削除されたブロックを使用してアップデートするのに、現在では 400MB を使用することができる。

記憶装置の与えられた容量に対して、新しく削除されたブロックに対する要求を減らすことはガーベッジコレクションの遂行頻度を減らし、媒体に対する消耗度を減らす。

【0054】

図 3A ~ 図 3E は、本発明の実施形態による記憶装置から読み出す動作を説明するための概念図である。

図 1 の記憶装置 100 が例として使用される。図 2A ~ 図 2E の構成要素と類似する構成要素の説明は簡易化のために省略する。 40

図 1 及び図 3A を参照すると、マッピングテーブル 310 は、図 2A で説明したように、データが書き込まれた後の状態を示す。

即ち、データ D はデータメモリ (304-1) 内に格納され、エントリ (311-3) はマッピングテーブル 310 に追加される。

しかし、エントリ (311-3) は DCV が存在しないか、或いは有効ではない表示を含む。

【0055】

コントローラ 102 は、読み出し要請 302 を受信する。

ここで、読み出し要請 302 は、論理的ブロックアドレス「1」にアクセスするように構成される。これに応答し、コントローラ 102 は、マッピングテーブル 310 をアクセス

するか、或いは物理的ブロックアドレス「23」を読み出す。

物理的ブロックアドレスを使用して、コントローラ102はデータメモリ(304-1)からデータDを読み出す。

コントローラ102は、データDと共に読み出し要請302に応答する。

特に、エントリ(311-3)はDCVが存在しないか、或いは有効ではない表示を含むので、コントローラ102は、変更(modification)無しでデータDを応答する。

【0056】

図1及び図3Bを参照すると、実施形態として、マッピングテーブル310が図2Bで説明したようにデータが書き込まれた後の状態であるので、読み出し要請302はコントローラ102によって受信される。

即ち、データDは、初期にデータメモリ(304-1)に書き込まれ、DCVメモリ(304-2)に格納されているので、アップデートされたデータD'が書き込まれる。

したがって、コントローラ102は、読み出し要請302を再び受信し、これに応答して、物理的ブロックアドレス「23」をアクセスすることによってデータメモリ(304-1)からデータDを読み出す。

しかし、有効DCVフィールドがエントリ(311-3)に存在するので、コントローラ102はDCVメモリ(304-2)にアクセスしてDCV'を読み出す。

コントローラ102は、逆DCV機能(308-2)に対する入力としてデータDとDCV'を使用し、データDとDCV'とを組み合わせてデータD'を生成する。

コントローラ102は、データD'と共に読み出し要請302に応答する。

【0057】

増加した読み出し量がこの技術に関する間に、その増加した読み出し量は、無視できないが、少ない量を有する。

例えば、実施形態として、メモリ104の内部読み出し幅(internal read bandwidths)は、記憶装置100の外部インターフェイス幅より高い。

たとえば、読み出し性能に無視できない影響を有しても、読み出しは削除されたブロックの使用、隣接するトラックの再書き込み等をもたらしない。

このように、読み出し性能を減らす動作は、レイテンシを減らすか、或いはレイテンシの一貫性を向上させるか、あるいは、その他の関連する動作より影響が少ない。

【0058】

図1及び図3Cを参照すると、実施形態として、読み出し要請302は、マッピングテーブル310が図2Cに示したようにデータが書き込まれた後の状態にある時、コントローラ102によって受信される。

即ち、データDは、初期にデータメモリ(304-1)に書き込まれる。

アップデートされたデータD'は、DCVメモリ(304-2)に格納されてあるDCV'の結果として書き込まれる。

そして、追加してアップデートされたデータD"は、DCVメモリ(304-2)に格納されてあるDCV"の結果として書き込まれる。

【0059】

コントローラ102は、再び物理的ブロックアドレス23のためのエントリ(311-3)にアクセスし、物理的ブロックアドレス「23」を使用してデータメモリ(304-1)にアクセスしてデータDにアクセスし、DCVアドレスを使用してDCVメモリ(304-2)をアクセスする。

しかし、データD"が論理的ブロックアドレス「1」に最も最近に書き込まれたデータであるので、DCV"は利用可能であり、DCVメモリ(304-2)でアクセスされるDCVである。

コントローラ102は、DCV"とデータDを逆DCV機能(308-2)に対する入力として使用してデータD"を生成する。

コントローラ102は、データD"と読み出し要請302とに応答する。

10

20

30

40

50

したがって、初期データDがアクセスされたが、最も最近データD”は再び生成される。

特に、データD’とそれと関連するDCV’は、データD”を生成するのに使用されない。

【0060】

図3A～図3Cで説明したように、コントローラ102は、DCVフィールドがDCVが存在するか、又は有効であるかを指示するか否かに基づいて、異なった動作するように構成される。

しかし、他の実施形態として、コントローラ102は、たとえ初期データがデータメモリ(304-1)に格納されている時にも、図3B及び図3Cと同様に動作するように構成され得る。 10

特に、エントリ311のDCVフィールドは、DCVフィールドにあるか、或いはDCVメモリ(304-2)内にある識別DCV(identity DCV)を指示することで初期化される。

識別DCVは、初期データDと共に逆DCV機能(308-2)に対する入力で使用される時、データDを発生する。

結果として、実際データを比較することから得られるDCVが存在するか、或いは有効であるか否かに関係なく、実質的に同一の動作がコントローラ102によって遂行される。

【0061】

図1及び図3Dを参照すると、実施形態として、データの以前の初期バーションにアクセスすることができる。 20

特に、コントローラ102は、ソース読み出し要請(302-1)を受信する。

ここで、ソース読み出し要請(302-1)は、論理的ブロックアドレス「1」を参照する。

その応答で、コントローラ102は、図3Aで説明したアクセスと同様に、データメモリ(304-1)の物理的ブロックアドレス「23」にあるデータDにアクセスする。

しかし、エントリ(311-3)のDCVフィールドは、図3B及び図3Cと同様に有効である。

即ち、論理的ブロックアドレス「1」には、データのアップデートバージョンが存在する。 30

図3Bと図3Cとは対照的に、初期データDは、再び生成された現在データD’又はD”に復帰する(returned)。

したがって、ソース読み出し要請(302-1)又は同様の要請を使用して、記憶装置100に格納されたデータの以前のバーションを読み出すことができる。

エントリ(311-3)のDCVフィールドからのDCVアドレスを読み出す動作が例として使用される反面、他の実施形態として、DCV値はエントリ(311-3)のDCVフィールドから読み出されてもよい。

例えば、図3B及び図3CのDCV’及びDCV”は各々エントリ(311-3)から読み出される。 40

【0062】

図1及び図3Eを参照すると、実施形態として、図2Cで説明したようにマッピングテーブル310がデータが書き込まれた後の状態にある時、読み出し要請(302-2)がコントローラ102によって入力される。

即ち、データDは初期にデータメモリ(304-1)に書き込まれる。

アップデートされたデータD’は、DCVメモリ(304-2)に格納されてあるDCV’の結果として書き込まれる。

そして、追加して、アップデートされたデータD”は、DCVメモリ(304-2)に格納されてあるDCV”の結果として書き込まれる。

しかし、この実施形態で、1つ又はそれ以上の中間DCVs(intermEDIATE) 50

e D C V s) が維持される。

この例で、D C V ” は現在の D C V である。

しかし、D C V ' は、また維持される。

エントリ (3 1 1 - 4) にある「 0 × 5 5 」の追加的なパラメーターは D C V ' の表示を示す (例えば、その値やアドレス) 。

【 0 0 6 3 】

この例で、読み出し要請 (3 0 2 - 2) は L B A 1 ' のための要請である。

L B A 1 ' はデータ D ' としてのデータの状態を示す。

したがって、コントローラ 1 0 2 は、D C V ' アドレス、即ち D C V ' が D C V メモリ (3 0 4 - 2) に格納された位置を示すアドレスにアクセスする。

結果として、D C V ' はアクセスされ、逆 D C V 機能 (3 0 8 - 2) にあるデータ D と結合されてデータ D ' を生成する。

たとえとして単なる 1 つの中間 D C V (例えば、D C V ') が例として使用されるが、他の実施形態としていくつかの数の中間 D C V も格納されることができる。

例えば、D C V s 、 D C V ' 、 D C V ” 、 D C V ' ' ' 、及び D C V ' ' ' ' が D C V メモリ (3 0 4 - 2) に全て格納される D C V s の各々は、データ D と結合されてデータ D ' 、 D " 、 D ' ' ' 、及び D ' ' ' ' のようなデータの以後バーションを各々生成することができる。

【 0 0 6 4 】

図 4 A 及び図 4 B は、本発明の他の実施形態による記憶装置を概略的に示すブロック図である。

図 4 A を参照すると、実施形態として、記憶装置 4 0 0 は、図 1 に示した記憶装置 1 0 0 のコントローラ 1 0 2 と同様のコントローラ 4 0 2 とを含む。

しかし、記憶装置 4 0 0 は、不揮発性メモリ (4 0 4 - 1) と揮発性メモリ (4 0 4 - 2) とを含む。

不揮発性メモリ (4 0 4 - 1) には例として、 f l a s h m e m o r y 、 S T T - M R A M 、 P h a s e - C h a n g e R A M 、 N F G M 、 o r P o R A M 、 m a g n e t i c o r o p t i c a l m e d i a 等が含まれ得る。

揮発性メモリ (4 0 4 - 2) には例として、 D R A M 、 D D R 、 D D R 2 、 D D R 3 、 D D R 4 、 S R A M 等のような多様な標準にしたがう D D R S D R A M 等が含まれ得る。

【 0 0 6 5 】

コントローラ 4 0 2 は、不揮発性メモリ (4 0 4 - 1) にデータ 4 0 6 を記憶し、揮発性メモリ (4 0 4 - 2) に D C V 4 0 8 を格納する。

コントローラ 4 0 2 は、また揮発性メモリ (4 0 4 - 2) にマッピングテーブル 4 1 0 を格納する。

実施形態として、記憶装置 4 0 0 の使用は、一貫性 (c o n s i s t e n c y) よりは一貫されたレイテンシ (c o n s i s t e n t l a t e n c y) に優先順位を置く。

したがって、データの一貫性は次順位になる (c o n s i s t e n c y o f t h e data m a y b e r e l a x e d) 。

例えば、様々なインターネット - スケールアプリケーション (s e v e r a l i n t e r n e t - s c a l e a p p l i c a t i o n s) は次順位範囲内で (w i t h i n r e l a x e d b o u n d s) 一貫性を考慮してアクセスされる。

そのようなアプリケーションは、ツイート (t w e e t s) 及びフォトタギング (p h o t o t a g g i n g) を含む。

【 0 0 6 6 】

しかし、そのようなアプリケーションのために、レイテンシスパイク (l a t e n c y s p i k e s) は受け入れない。

D C V 4 0 8 で表現されるように、このような範囲内のデータは、揮発性メモリ (4 0 4 - 2) に格納される。

コントローラ402は、DCV408のオーバープロー(overflows)を不揮発性メモリ(404-1)にフラッシュ(flush)するようとする。

実施形態として、コントローラ402はDCV408を不揮発性メモリ(404-1)のデータ406にフラッシュする。

【0067】

揮発性メモリ(404-2)はバッテリ、スーパーキャパシタ、又はNVRAMによってバックアップされる(may be backed)。

しかし、実施形態として、そのようなバックアップは、DCV408の損失を生じる失敗が相変わらず、許容可能な範囲内にあるので、必ず必要であることではない。

バックアップのために使用されるバッテリ、スーパーキャパシタ、又はNVRAMを省略すれば、記憶装置400の費用(cost)を減少させることができる。 10

【0068】

実施形態として、キャッシング412は、データを不揮発性メモリ(404-1)に伝送する前に、データをキャッシングするのに使用される。

例えば、図2A～図2Dで説明した読み出し要請から追い出されたデータは、キャッシング412に格納される。

このストレージは、図2A～図2Dで説明した多様な技術無しでも動作することができる。しかし、データがキャッシング412から追い出される時又はそれと同様な場合に、図2A～図2Dで説明した技術が使用することができる。

特別な例として、キャッシング412に格納されたデータブロックは、複数の読み出し要請によってアップデートされる。 20

このようなアップデートは、DCVの計算に関係しない。

データブロックがデータを追い出されるか、或いはコミットされる時、データの状態に基づいて図2A～図2Dで説明したようにDCVを生成する。

【0069】

図4Bを参照すると、実施形態として、記憶装置401は図4Aの記憶装置400と同様である。

しかし、記憶装置401では、コントローラ402はキャッシングされたDCV(408-1)として表現される揮発性メモリ(404-2)内DCVをキャッシングするようとする。 30

特に、コントローラ402は、頻繁にアクセスされるDCVを揮発性メモリ(404-2)に維持する。

コントローラ402は、キャッシングアルゴリズムや発見的な方法(heuristics)等を使用して、どのキャッシングされたDCV(408-1)を維持するかを決定する(determine which DCVs 408-1 to maintain in the cache)。

コントローラ402は、他のキャッシングされたDCV(408-1)を不揮発性メモリ(404-1)に格納されたDCV(408-2)に伝送し、その逆も成立する。

結果的に、実施形態として、頻繁にアクセスされるか、或いは重くアクセスされるデータはキャッシングされる。 40

さらに、キャッシングされたDCV(408-1)は、対応するデータ406よりサイズが小さいので、実際のデータがキャッシングされた場合よりも、より多くのアップデートが揮発性メモリ(404-2)で維持される。

【0070】

図5は、本発明の実施形態による記憶装置内のページデータサイズを示す図ある。

実施形態として、データページ502とDCVページ504とは同じメモリ500に格納される。

ここで、データページ(502-1、502-2)とDCVページ(504-1)～(504-4)とはメモリ500に格納されたデータページ502及びDCVページ504の例である。 50

ここで、DCVページ504はソースデータと比較して相対的にさらに小さいサイズのDCVを示すためにさらに小さく示している。

この例で、DCVページ504はデータページ502の半分サイズである。

しかし、他の実施形態では、DCVページ504のデータサイズは使用された特別なDCV機能によって異なり得る。

【0071】

実施形態として、データページ502及びDCVページ504の両方はレイテンシを増加させる順次書き込み(*in-place writes*)の不足のような、同一の制限を受け得る。

上記で説明したように、データページ502は一般的にDCVが変わる間は維持される

10

。

DCVページに対する変化は、レイテンシ影響(*latency impact*)をもたらす。

しかし、DCVページ504のサイズが段々小さくなることによって、レイテンシ影響は減少される。

例えば、DCVページ504がデータページ502と異なるブロックに格納される。

したがって、さらに多くのDCVがDCVページ504に蓄積されることによって、ガーベッジコレクションがそのブロックで遂行されて自由ブロックが回復する(*recovery*)。

DCVページ504がさらに小さいので、ガーベッジコレクションはより少ない時間で遂行されるか、及び/又はより少ない頻度で遂行される。

20

【0072】

これに加えて、データページ502はDCVページ504より長い間、有効な状態を維持する。

結果的に、データページ502を格納するブロックはDCVページ504を格納するブロックよりもガーベッジコレクションを受けにくい。

また、DCVページ504が早く無効化される可能性がより高いので、DCVページ504を格納するブロックのより多くのページが無効となり、ガーベッジコレクション操作の時間が短縮される。

【0073】

30

図6は、本発明の実施形態によるSSDを示すブロック図である。

実施形態として、SSD600は図1に図示された記憶装置100のコントローラ102と同様である。

しかし、SSD600は、フラッシュメモリ(604-1)、DRAM(604-2)、及びSRAM(604-3)を含む。

コントローラ602は、フラッシュメモリ(604-1)にデータ606とDCV(608-1)とを格納する。

例えとして、メモリの構成を特定して示しているが、他の実施形態として、SSD600はデータ606及びDCV(608-1、608-2)の構成と分布が他のメモリを含むことができ、マッピングテーブル610も異なることができる。例え、SSD600は図1、図4A、及び図4Bと同様な構成を有する。

40

【0074】

フラッシュメモリ(604-1)に格納されたデータ606とDCV(608-1)はガーベッジコレクションを頻繁に受けやすい(*susceptible to garbage collection*)。

データ606とDCV(608-1)とを格納するページが削除され、潜在的に追加的なレイテンシを招く。

しかし、先に説明したように、DCV(608-1)のサイズは対応するデータ606より小さい。

即ち、データ606を格納するために使用されるフラッシュメモリ(604-1)内の

50

ページは D C V (6 0 8 - 1) を格納するのに使用されるページよりサイズが大きい。

D C V を使用すれば、 S S D 6 0 0 内で書込み変更 (w r i t e m o d i f i c a t i o n s) 、 即ち非順次アップデート (o u t - o f - p l a c e u p d a t e s) が減少する。これはリクレーム (r e c l a i m) される無効ブロックを小さくする。

【 0 0 7 5 】

これによって、ガーベッジコレクションの回数 (f r e u e n c y) が減少する。

特に、 S S D 6 0 0 は、 ビックデータとクラウドアプリケーションとアップデート - ヘビ - I / O トランザクションを処理する。

例えば、 S S D 6 0 0 は、 アップデータとクラウドアプリケーションとを有し、ガーベッジコレクションの低い回数とそれによる高いレイテンシの低い機会、特にレイテンシスパイクを有し、アップデート - ヘビ - I / O トランザクションを処理する。

【 0 0 7 6 】

特に、データ 6 0 6 に対する書込みアップデートをする時、データ 6 0 6 の第 1 コピーはフラッシュメモリ (6 0 4 - 1) に残る。

フラッシュメモリ (6 0 4 - 1) にある D C V (6 0 8 - 1) と D R A M (6 0 4 - 2) にキャッシュされたフラッシュ D C V (6 0 8 - 2) は、書込みアップデートがヘビーな作業負荷 (h e a v y w o r k l o a d s) であっても、フラッシュメモリ (6 0 4 - 1) にあるページの寿命を増加させる。

これに加えて、先に説明したように、データ 6 0 6 の以前バージョンも利用することができる。

レイテンシを向上させることのみならず、アーキテクチャがフラッシュセル消耗 (w e a r i n g) を改善することもできる。

【 0 0 7 7 】

特に、以前ページを無効化し、アップデートデータのための新しいページを要請する代わりに、コントローラ 6 0 2 は、以前ページとアップデートとの間の差分を示す D C V と共にフラッシュメモリ (6 0 4 - 1) に有効 / アクティブとして以前ページを維持する。

続く読み出し動作の時に、コントローラ 6 0 2 は、以前ページと D C V とを全て読み出し、結合してデータページまで大部分を提供する。

上記で説明したように、 D C V をどこに記憶するかについては多数の構成がある。

例えば、フラッシュメモリ (6 0 4 - 1) 、 D R A M (6 0 4 - 2) 、 S R A M (6 0 4 - 3) 、又はそのようなメモリの組み合わせに格納する。

これに加えて、アクティブ又は“ホット”ページ D C V を D R A M (6 0 4 - 2) にキャッシュし、フラッシュメモリ (6 0 4 - 1) 上の D C V (6 0 8 - 1) に持続的な (p e r s i s t e n t) コピーを維持することによって、 D C V はキャッシュされる。

【 0 0 7 8 】

実施形態として、読み出しと書込みが非対称 (a s y m m e t r i c) でないか、同一のオーバーヘッドを有するか、或いはそのようなものである時でも、本明細書で説明したように、記憶装置は性能利点を有する。

特に、仮に書込みのサイズと D C V のデータサイズとを有する書込み時間スケール (w r i t e t i m e s s c a l e) が対応するデータブロックのデータサイズより小さければ、書込み時間は減少する。

実施形態として、 D R A M (6 0 4 - 2) は、書込み要請及び / 又は書込みデータをキャッシュするのに使用することができる。

書込み要請を処理することがデータ 6 0 6 に存在するデータを読み出すことに関する時、対応するデータ 6 0 6 はキャッシュされたデータ (6 0 6 - 1) として D R A M (6 0 4 - 2) に格納される。

コントローラ 6 0 2 は、上記で説明した D C V 関数に対する入力として、 D R A M (6 0 4 - 2) に格納されたキャッシュされたデータ (6 0 6 - 1) を使用する。

即ち、フラッシュメモリ (6 0 4 - 1) に格納されたデータ 6 0 6 からデータを読み出す代わりに、コントローラ 6 0 2 は D R A M (6 0 4 - 2) に格納されたキャッシュされ

10

20

30

40

50

たデータ(606-1)からデータを読み出す。

【0079】

実施形態として、フラッシュメモリ(604-1)はパネルとチャンネルとに区分することができる。

SSD600は、ページ単位に読み出し、書き込み、ブロック単位に消去する。ブロックは複数のページを含む。

マッピング方策(mapping strategy)は、翻訳単位(granularity of translation)を定義する。

例えば、ページレベルマッピングは、より大きい面積(larger footprint)を必要とするが、さらに高い流動性(higher degree of file 10 xibility)を提供することができる。

ブロックレベルマッピングは、より小さい面積(smaller footprint)を使用するが、配置(placement)に制限的である。

幾つのハイブリッド方策の変形がページ及びブロックに基づいたマッピングの組み合わせを利用するように提案されている。

マッピングテーブル610にあるマッピングはそのようなマッピング技術の中からいずれでも使用することができる。

【0080】

SSD600を例として説明したが、他の実施形態として、SMD(shingled magnetic drives)のような他の形態の記憶媒体も使用され得る。 20

特に、「shingled disk」上では、書き込みは以前に書き込まれた磁気トラックの一部分とオーバーラップする。

これは書き込み性能を低くし、また隣接するトラックを上書きする結果を招く。

SMD(shingled magnetic drives)はファームウェアでこれを管理することによって、このような複雑性(complexity)を隠す。

さらに低い書き込み性能(隣接するトラックに書き込みをする時)は不一致レイテンシ(inconsistent latency)をもたらし、大部分のクラウドアプリケーションで解決しなければならない問題である。

本明細書で説明したように、DCVを使用すれば、上書きが縮み、アップデート・集中作業負荷(update-intensive workloads)に対する一致したトランザクションレイテンシ(consistent transaction latency)を提供することができる。 30

【0081】

図7は、本発明の実施形態によるバージョニング記憶システムを概略的に示すブロック図である。

実施形態として、システム700は、通信リンク706を通じて記憶装置704に接続されるホスト702を含む。

ホスト702は、記憶装置のデータ記憶能力を使用するシステムとして、「general purpose processor」、「digital signal processor(DSP)」、「application specific integrated circuit」、「microcontroller, programmable logic device」、「discrete circuits」、及びこのような装置の組み合わせやこれと同様なものを含む。 40

【0082】

実施形態として、ホスト702は、コンピュータ、サーバー、ワークステーション、又は同様なものであってもよい。

ホスト702は、オペレーションシステム(Operating System: OS)及びアプリケーションのようなソフトウェアを遂行する。

ホスト702は、メモリ708に接続される。

メモリ708は、運用メモリ(operationnal memory)及び/又はホ

50

スト 702 に使用されるキャッシュメモリを含む。

キャッシュメモリの例としては、DRAMや、DDR、DDR2、DDR3、DDR4、SRAM等のような多様な標準にしたがう DDRS DRAM等が含まれる。

メモリ 708 がホスト 708 と分離して説明したが、実施形態として、メモリ 708 はホスト 702 の一部分であってもよい。

【0083】

通信リンク 706 は、ホスト 702 と記憶装置 704 とが通信するよう構成される媒体を示す。

例えば、通信リンク 706 は、USB、SCSI、PCIe、SAS、PATA、SAT、NVMe、UFS、Fiber channel、Ethernet、RDMA、Infiniband、又は他の同様なリンク、のようなリンクである。10

ホスト 702 と記憶装置 704との各々はそのような通信リンクと通信するためのインターフェイスを有するように構成される。

【0084】

実施形態として、記憶装置 704 は、先に説明した記憶装置 (100、400、401、600) と同様な記憶装置であり、図 2A～図 3D で説明した方式の通り動作するよう構成される。

ホスト 702 は、記憶装置 704 からデータを読み出すか、或いは記憶装置 704 にデータを書き込むように構成される。

記憶装置 704 は、先に説明したようにシステム 700 の動作を向上するように DCV を使用するよう構成される。20

【0085】

しかし、他の実施形態として、ホスト 702 は DCV の観点から、先に説明した方法と同様な動作を遂行するように構成される。

例として、ホスト 702 は、記憶装置ドライバー 710 を含む。

記憶装置ドライバー 710 は、記憶装置 704 と動作するホスト 702 との上で遂行されるソフトウェアを示す。

特に、記憶装置ドライバー 710 は、DCV の観点から先に説明した方法と同様な動作を遂行するように構成される。

即ち、コントローラ (102、402、602) 等によって遂行された先に説明した動作は、記憶装置ドライバー 710 によって遂行される。30

これに加えて、メモリ 708 は少なくともメモリ (104、404-2、604-2、604-3) の一部分と同様に使用される。

即ち、メモリ 708 は、例えはマッピングテーブル (110、410、610)、DCV (108、408)、及び / 又はキャッシュされた DCV (408-1、608-2) を格納するのに使用される。

【0086】

実施形態として、図 2A～図 3D で説明した論理的ブロックアドレス及び物理的ブロックアドレスの関連性はマッピングテーブルに示す必要はない。

記憶装置ドライバー 710 は、論理的アドレスと DCV の表示 (indication of DCV) の関連性を維持する。40

即ち、記憶装置ドライバー 710 は、初期データ D が格納されたところに対する表示として、物理的ブロックアドレスと同様に論理的ブロックアドレスを使用する。

実施形態として、記憶装置ドライバー 710 は、図 2A～図 3D で説明したデータメモリ (204-1、304-1) のような記憶装置 704 を使用するように構成される。

【0087】

実施形態として、論理的ブロックアドレスに対して、記憶装置ドライバー 710 は、初期データを記憶装置 704 上の論理的ブロックアドレスに書き込む。

記憶装置ドライバー 710 は、初期データをメモリ 708 にキャッシュするようにする。50

データが続いて変更される時、記憶装置ドライバー710は、キャッシュされた初期データを使用してDCVを計算し、DCVを記憶装置に書き込むようとする。

他の実施形態として、記憶装置ドライバー710は、DCVをメモリ708に維持する。

先に説明したDCV管理技術は、記憶装置ドライバー710に使用される。

【0088】

図8は、本発明の実施形態によるサーバーを概略的に示すブロック図である。

実施形態として、サーバー800は、「stand-alone server」、「rack-mounted server」、「blade server」等を含む。

サーバー800は、記憶装置802とプロセッサ804とを含む。

プロセッサ804は、記憶装置802に接続される。

例えとして、1つの記憶装置802を示しているが、複数の記憶装置802が存在することができる。

記憶装置802は、先に説明した記憶装置の中のいずれでもあってもよい。

それにより、サーバー800の性能は向上され得る。

【0089】

図9は、本発明の実施形態による分散システムを概略的に示すブロック図である。

実施形態として、サーバーシステム900は複数のサーバー(902-1)～(902-N)を含む。

サーバー(902-1)～(902-N)は各々マネージャ904に接続される。

1つ又はそれ以上のサーバー(902-1)～(902-N)は先に説明したサーバー800と同様である。

マネージャ904は、サーバー(902-1)～(902-N)及びサーバーシステム900の他の構成を管理する。

実施形態として、マネージャ904は、サーバー(902-1)～(902-N)の性能をモニターするように構成される。

例えば、サーバー(902-1)～(902-N)の各々は、先に説明した記憶装置を含む。

【0090】

図10は、本発明の実施形態によるデータセンターを概略的に示すブロック図である。

実施形態として、データセンター1000は複数のサーバーシステム(1002-1)～(1002-N)を含む。

サーバーシステム(1002-1)～(1002-N)は、図9に示したサーバーシステム900と同様である。

サーバーシステム(1002-1)～(1002-N)は、インターネットのようなネットワーク1004と結合される。

したがって、サーバーシステム(1002-1)～(1002-N)は、ネットワーク1004を通じて多様なノード(1106-1)～(1106-M)と通信する。

例えば、ノード(1106-1)～(1106-M)は、クライアントコンピュータ、他のサーバー、リモートデータセンター、記憶システム、又はこれと同様なものであってもよい。

【0091】

実施形態において、クラウド及び「large-scale latency-critical services」のように、一貫された性能が重要な要素(import ant factor)であるところで使用することができる。

そのようなサービスの例は、データ分析(data analytics)、機械学習(machine learning)、オーディオ及びビデオストリーミング(audio and video streaming)を含む。

一貫したレイテンシは高い優先順位である。

そして、幾つかの実施形態は、さらに予測可能な性能を支援する「relaxed c

10

20

30

40

50

o n s i s t e n c y」要求 (r e p u i r e m e n t s) を採用する配布されたソフトウェアスタックを使用する。

たとえば、幾つかの要素がリソース共有及び待ち行列 (r e s o u r c e s h a r i n g a n d q u e u i n g) のような、負荷されたレイテンシ (l o a d e d l a t e n c y) で不一致を発生させても、S S D 内でガーベッジコレクションは相当な利点を有することができる。

【0092】

現在のソフトウェアスタックは、複製 (r e p l i c a t i o n) を使用することができる。

しかし、この解決はガベージコレクションなどの問題の根本を解決することができない 10
。

これに加えて、複製は内在的にさらに多い費用が掛かり、増加されたネットワークトラフィックをもたらす。

これはさらにネットワークレイテンシに影響を及ぼす。

また、複製はソフトウェア層で「c o o r d i n a t i o n」を使用する。

大きいデータ生成率によって、さらに速いデータ分析に使用することができる。

サーチエンジンとソーシャルメディアのようなオンラインサービスにおいて重要なデザイン目標は予測可能な性能を提供することにある。

このような状況 (c o n t e x t) では、平均応答時間は性能を代表するものではない 20
。

【0093】

最も悪いケース (w o r s t c a s e) がさらに性能の関心事である。

多様な反応時間がサービスの要素においてさらに高いテールレイテンシ (h i g h e r t a i l l a t e n c y) を引き起こすことがある。

結果的に、使用者は長い応答時間を経る。

業務負荷 (w o r k l o a d) とS S D ファームウェア方策に基づいて、テールレイテンシスパイク (t a i l l a t e n c y s p i k e s) は、まれに又は頻繁に発生し得る。

しかし、大部分の場合に、現在非常に競争的な市場で使用者の経験及びサービス提供者の評判を脅かすのに十分である。 30

テールレイテンシペナルティーは、A m a z o n (登録商標) A W S やG o o g l e (登録商標) C l o u d のような共有基盤施設 (s h a r e d i n f r a s t r u c t u r e s) でさらに悪化する (e x a c e r b a t e d)。

このような問題点は、いくつかのクラウドベンダーによって共有されており、ハードウェア / ソフトウェアシステムアーキテクチャをデザインしている会社の主要挑戦の中の1つとして認識されている。

【0094】

たとえば、幾つかの要素がリソース共有及び待ち行列 (r e s o u r c e s h a r i n g a n d q u e u i n g) のような、負荷されたレイテンシ (l o a d e d l a t e n c y) で不一致を発生させても、S S D 内でガーベッジコレクションは相当な利点を有することができる。 40

順次アップデート (i n - p l a c e u p d a t e s) を許容しないフラッシュメモリの特性により、S S D ファームウェアは非順次的に (o u t o f p l a c e) にアップデートを書き込み、以前コピーを無効化させる。

リクレーム空間 (r e c l a i m s p a c e) のために、無効空間はそれが再び書き込まれる前に削除される必要がある。

しかし、削除動作 (i n m i l l i s e c o n d s) は、読み書き込み動作 (i n m i c r o s e c o n d s) に比べてさらに相当遅い。

そして、削除動作は、典型的にさらに粗い単位 (o n a c o a r s e r g r a n u l a r i t y) に行われる。 50

【0095】

このようなプロセスは、SSD内でガーベッジコレクションを生じさせ、チャンネルはガーベッジコレクションが生じる間、読み出し/書き込み要請をサービスできない。

このような理由で、ガーベッジコレクションは、臨界動作(critical operations)においてレイテンシ又は性能に悪い影響を及ぼせる。

いくつかの場合には読み出しレイテンシがガーベッジコレクションの間に100倍増加し得る。

これに加えて、ガーベッジコレクションはSSDが使用されることによってさらに頻繁に発生する可能性がある。

【0096】

10

クラウドで顕著ないくつかのアプリケーションタイプは書き込みアップデートが重い。

一例として、実世界の使用ケース(real-world use case)をエミュレートするこのような例の1つにYCSB(Yahoo(登録商標)Cloud Serving Benchmark)がある。

これはクラウドシステムを評価するためのベンチマークスイート(benchmark suite)である。

YCSBで提供される幾つかのアップデート-過作業負荷(update-heavy workloads)で、アクセス比率は50%読み出し、0%挿入、及び50%アップデートである。

【0097】

20

e-コマースアプリケーション(e-commerce applications)は、ユーザセッションにある「recording recent actions」、「e-commerce user」の典型的なアクションのような動作を含む、アプリケーションの例である。

アプリケーションのこのようなカテゴリで、SSD内のガーベッジコレクションによるテールレイテンシの影響は、使用者反応時間にさらに高いレイテンシを引き起こす可能性がある。

他の作業負荷(workload)の例として、アクセス比率は95%読み出し、0%挿入、5%アップデートである。

一例はソーシャルメディアであり、たとえ小さいアップデート率であってもガーベッジコレクションをトリガーし、サービスレベル目標を違反する可能性がある。

例えば、フォトタギングを有し、タグを加えることはアップデートであるが、大部分の動作はタグを読み出すことである。

【0098】

ビックデータとクラウドアプリケーションの大部分は、トランザクション性(アトミック性(atomicity)、一貫性、隔離性、耐久性、またはACID)などのより多くの伝統的な側面に対してスケーラビリティおよび一貫したパフォーマンスを優先させる。

強い一貫性は正しくスケールしない(strong consistency does not scale well)。

ほとんどのクラウドアプリケーションでは、一貫性のあるパフォーマンスを目標として、より緩やかなトランザクションの整合性が優先される。

このような理由で、弱い一貫性モデルは、最も一般的なクラウド規模の分散ソフトウェアスタック全体で広く使用される。

これは、高負荷の並行システムでのI/Oパフォーマンスの大幅な向上をもたらす。

【0099】

40

すべてのクラウドアプリケーションが多い量のデータを格納し、増加するデータフットプリント(footprint)にしたがってスケールされるようにデザインされているが、単なるサブセット(subset)は他のものよりもさらに頻繁にアクセスされている。このような不均等な配布はデータセンターで“ホット”データアクセスパターンを表わ

50

される。

【0100】

ここで説明した実施形態は多様なアプリケーションで使用することができ、減少され、さらに一貫されたレイテンシの利点を提供する。

先に説明したクラウドアプリケーションは、そのようなアプリケーションの単純な例に過ぎない。

たとえば、構造、装置、方法、そしてシステムを特定の実施形態にしたがって説明したが、本発明の技術分野で通常の知識を有する者は多様な変更が可能であることを容易に認識することができる。したがって、多い変形が追加された請求項の範囲及び技術的思想を逸脱しない限り、多様になされることができる。

10

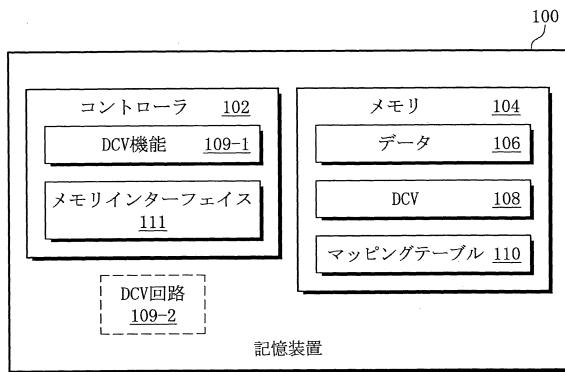
【符号の説明】

【0101】

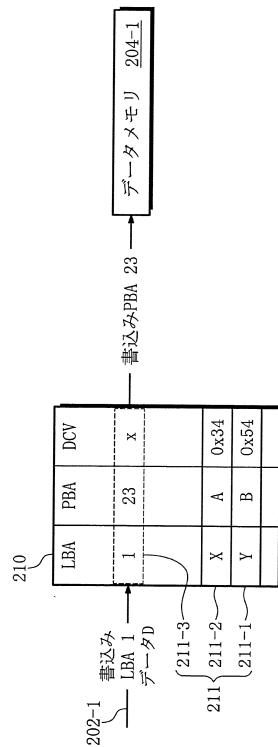
- 100 記憶装置
- 102 コントローラ
- 104 メモリ
- 106 データ
- 108 DCV
- 109-1、208-1 DCV機能
- 109-2 DCV回路
- 110、210 マッピングテーブル
- 111 メモリインターフェイス
- 204-1 データメモリ
- 204-2 DCVメモリ
- 208-2 逆DCV機能

20

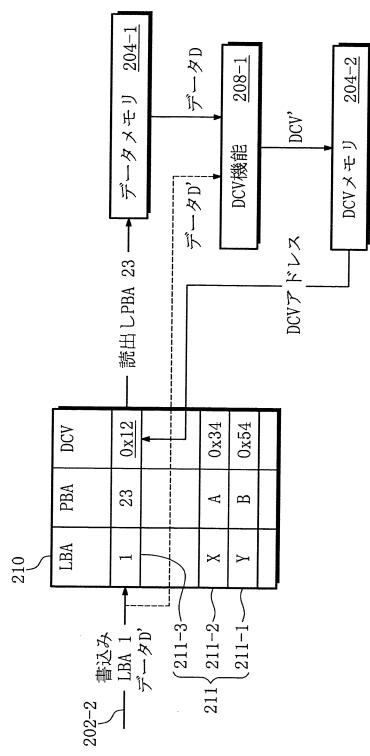
【図1】



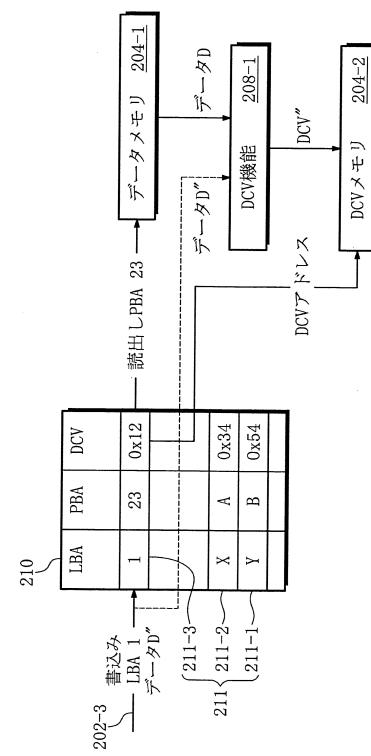
【図2A】



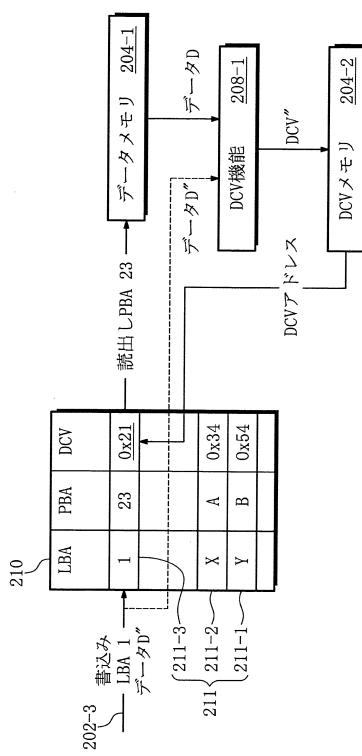
【図2B】



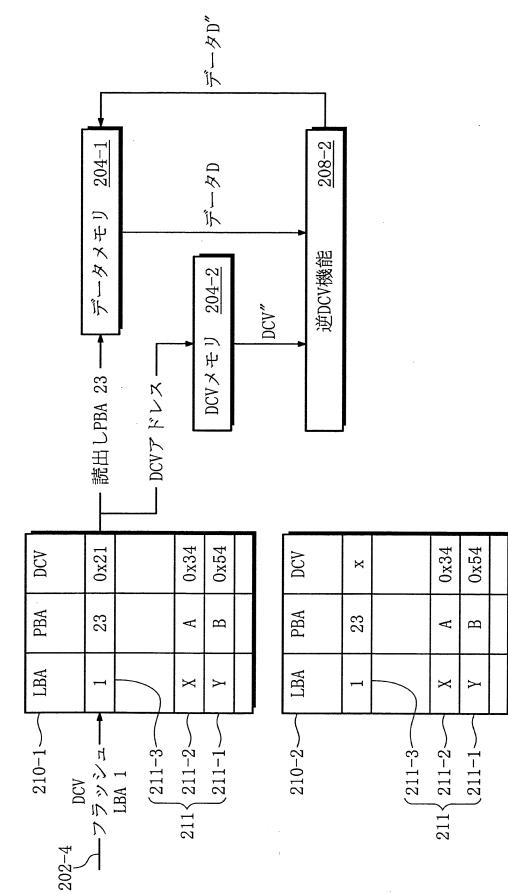
【図2C】



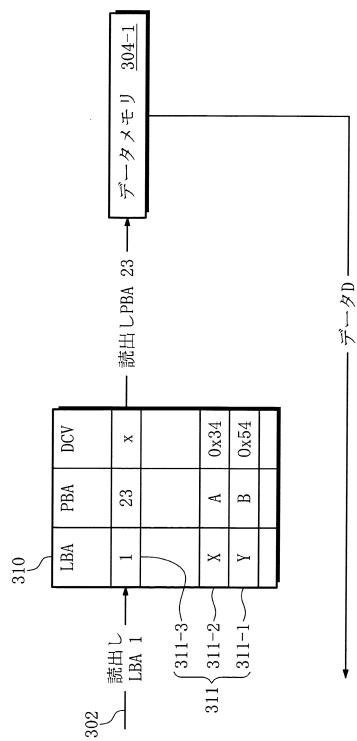
【図2D】



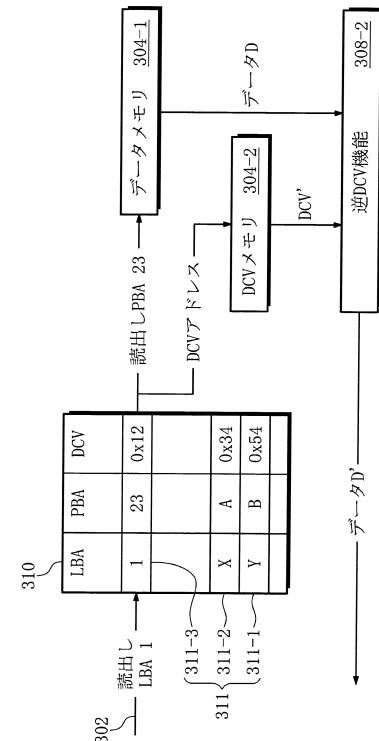
【図2E】



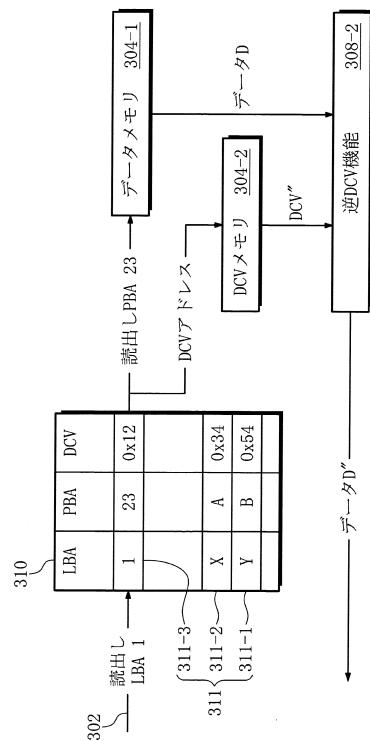
【図3A】



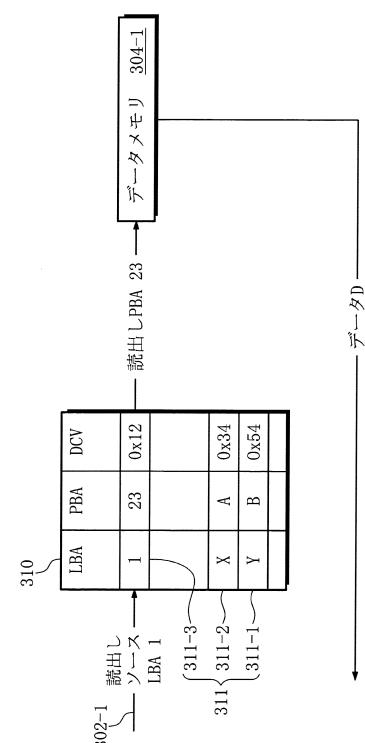
【図3B】



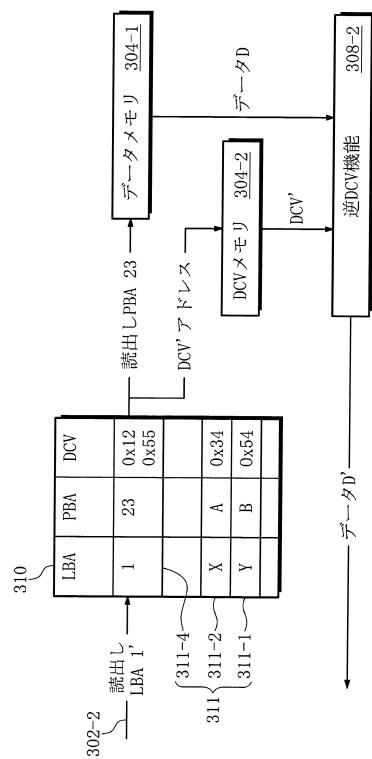
【図3C】



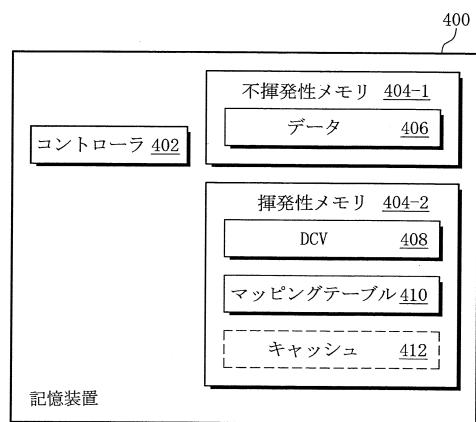
【図3D】



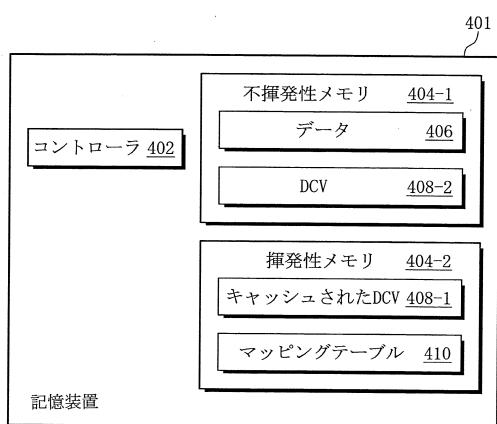
【図3E】



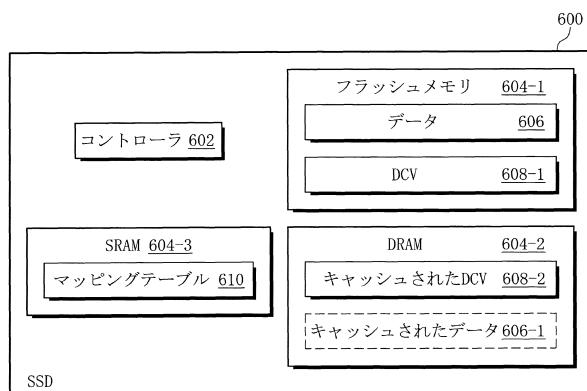
【図4A】



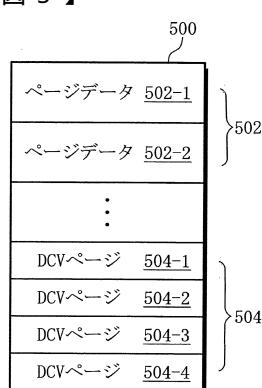
【図4B】



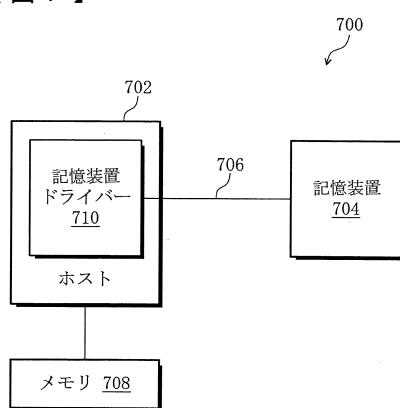
【図6】



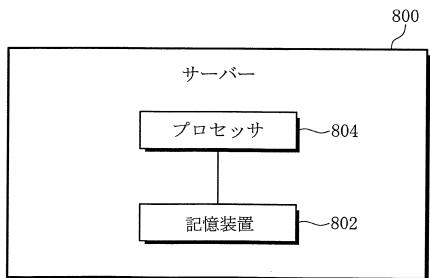
【図5】



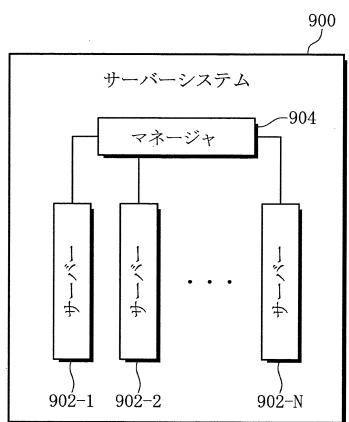
【図7】



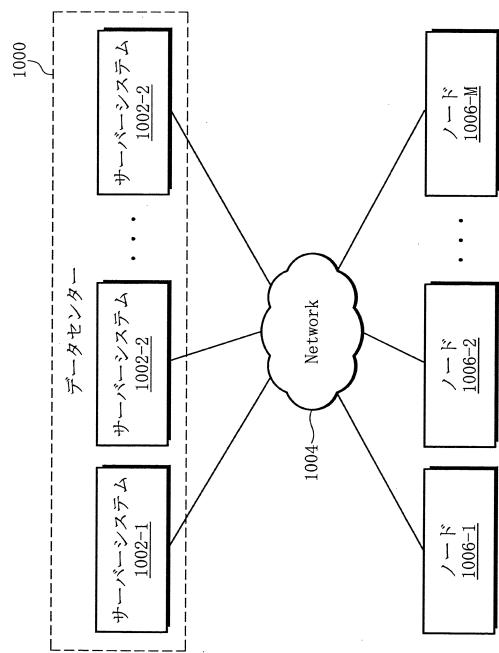
【図8】



【図9】



【図10】



フロントページの続き

(72)発明者 スーリ タメシユ

アメリカ合衆国， カリフォルニア州 95136， サンノゼ， マーブルアーチアベニュー，
361

(72)発明者 ブレナン ロバート

アメリカ合衆国， カリフォルニア州 95054， サンタクララ， ミルクリークレーン #
201， 586

審査官 後藤 彰

(56)参考文献 國際公開第2015/128955 (WO, A1)

米国特許出願公開第2016/0328154 (US, A1)

特開2010-218529 (JP, A)

(58)調査した分野(Int.Cl., DB名)

G 06 F 12/04

G 06 F 3/06

G 06 F 3/08