(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第5806776号
(P5806776)
(45)発行日　平成27年11月10日(2015.11.10)　(24)登録日　平成27年9月11日(2015.9.11)

(51)Int.Cl.　F I
G06F 3/06 (2006.01)　G06F 3/06 301Z
G06F 13/10 (2006.01)　G06F 13/10 340A

請求項の数 17　(全 57 頁)

(21)出願番号　特願2014-505435 (P2014-505435)
(86)(22)出願日　平成23年10月7日 (2011.10.7)
(65)公表番号　特表2014-527204 (P2014-527204A)
(43)公表日　平成26年10月9日 (2014.10.9)
(86)国際出願番号　PCT/JP2011/005649
(87)国際公開番号　WO2013/051069
(87)国際公開日　平成25年4月11日 (2013.4.11)
審査請求日　平成26年2月3日 (2014.2.3)

(73)特許権者　000005108
株式会社日立製作所
東京都千代田区丸の内一丁目６番６号
(74)代理人　110001678
特許業務法人藤央特許事務所
(72)発明者　吉原　朋宏
東京都千代田区丸の内一丁目６番６号　株式会社日立製作所内
(72)発明者　出口　彰
東京都千代田区丸の内一丁目６番６号　株式会社日立製作所内
(72)発明者　坏　弘明
東京都千代田区丸の内一丁目６番６号　株式会社日立製作所内

最終頁に続く

(54)【発明の名称】ストレージシステム

(57)【特許請求の範囲】
【請求項１】
それぞれが、複数の不揮発性半導体メモリを含む複数の記憶ドライブに基づいて構成される複数のボリュームと、
前記複数のボリュームのデータを一時的に格納するキャッシュメモリと、
複数のプロセッサパッケージと、
共有メモリと、を含み、
前記複数のプロセッサパッケージそれぞれは、前記複数のボリュームに含まれる１つのボリュームに対する入出力を担当するプロセッサと、前記１つのボリュームのデータキャッシング制御情報を格納するローカルメモリと、を含み、
前記共有メモリは、前記複数のプロセッサパッケージそれぞれのローカルメモリに格納されたデータキャッシング制御情報を格納し、前記プロセッサによってアクセス可能であり、
第１プロセッサパッケージに含まれ、第１ボリュームを担当する第１プロセッサは、計算機からの前記第１ボリュームへのリードコマンドを受信すると、前記第１プロセッサパッケージに含まれる第１ローカルメモリに格納されたデータキャッシング制御情報を更新し、
前記第１プロセッサは、前記第１ローカルメモリにおける前記データキャッシング制御情報の更新を前記共有メモリに反映するか否か、及び前記リードコマンドにおけるリードデータを前記キャッシュメモリに格納するか否かを決定し、

前記第１プロセッサは、前記第１ボリュームの記憶領域を提供する記憶ドライブに前記リードデータが格納されている場合に、前記リードデータを前記キャッシュメモリに格納することなく、かつ前記共有メモリに前記データキャッシング制御情報の更新を反映することなく、前記第１ボリュームの記憶領域を提供する記憶ドライブから読み出された前記リードデータを前記計算機に送信することを決定する、ストレージシステム。

【請求項２】

異なる制御情報を担当する複数のプロセッサと、

前記複数のプロセッサのそれぞれに割り当てられており、割り当てられたプロセッサが担当する制御情報を格納するローカルメモリと、

前記複数のプロセッサがアクセス可能であり、前記複数のプロセッサにおける第１プロセッサが担当する制御情報を格納する共有メモリと、を含み、

前記第１プロセッサは、割り当てられている第１ローカルメモリにおいて制御情報を更新し、

前記第１プロセッサは、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定し、

前記第１プロセッサは、前記共有メモリにおける前記制御情報に反映することを決定した前記第１ローカルメモリにおける前記制御情報の更新を、前記共有メモリにおける前記制御情報に反映する、ストレージシステムであって、

前記ストレージシステムは、複数のボリュームを提供するアクセス性能が異なる複数種別の不揮発性記憶領域と、キャッシュ領域とを含み、

前記第１ローカルメモリにおける前記制御情報及び前記共有メモリにおける前記制御情報は、それぞれ、前記複数のボリュームにおける前記第１プロセッサが担当する第１ボリュームのデータキャッシング制御情報を含み、

前記第１プロセッサは、前記第１ボリュームを提供する不揮発性記憶領域の種別に基づいて、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定する、ストレージシステム。

【請求項３】

前記第１プロセッサは、当該第１プロセッサの負荷、前記キャッシュ領域の負荷及び前記第１ボリュームのキャッシュヒット率の少なくとも一つに基づいて、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定する、請求項２に記載のストレージシステム。

【請求項４】

異なる制御情報を担当する複数のプロセッサと、

前記複数のプロセッサのそれぞれに割り当てられており、割り当てられたプロセッサが担当する制御情報を格納するローカルメモリと、

前記複数のプロセッサがアクセス可能であり、前記複数のプロセッサにおける第１プロセッサが担当する制御情報を格納する共有メモリと、を含み、

前記第１プロセッサは、割り当てられている第１ローカルメモリにおいて制御情報を更新し、

前記第１プロセッサは、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定し、

前記第１プロセッサは、前記共有メモリにおける前記制御情報に反映することを決定した前記第１ローカルメモリにおける前記制御情報の更新を、前記共有メモリにおける前記制御情報に反映する、ストレージシステムであって、

前記共有メモリにおける前記制御情報はカウントされる数値を含み、

前記第１ローカルメモリにおける前記制御情報は、前記数値の前回の更新からの変化を示す差分値を含み、

前記第１プロセッサは、前記差分値が規定数に達すると、前記第１ローカルメモリにおける前記制御情報に基づいて前記共有メモリにおける前記制御情報に含まれる前記数値を更新する、ストレージシステム。

【請求項５】
異なる制御情報を担当する複数のプロセッサと、
前記複数のプロセッサのそれぞれに割り当てられており、割り当てられたプロセッサが担当する制御情報を格納するローカルメモリと、
前記複数のプロセッサがアクセス可能であり、前記複数のプロセッサにおける第１プロセッサが担当する制御情報を格納する共有メモリと、を含み、
前記第１プロセッサは、割り当てられている第１ローカルメモリにおいて制御情報を更新し、
前記第１プロセッサは、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定し、
前記第１プロセッサは、前記共有メモリにおける前記制御情報に反映することを決定した前記第１ローカルメモリにおける前記制御情報の更新を、前記共有メモリにおける前記制御情報に反映する、ストレージシステムであって、
前記ストレージシステムはボリュームを提供する１以上の不揮発性記憶領域を含み、
前記第１ローカルメモリにおける前記制御情報及び前記共有メモリにおける前記制御情報は、それぞれ、前記ボリュームにおける記憶領域へのアクセス数の情報を含み、
前記第１プロセッサは、前記ボリュームにおける前記記憶領域へのアクセスに応答して前記第１ローカルメモリにおける前記アクセス数の情報を更新し、
前記第１プロセッサは、前記第１ローカルメモリにおけるアクセス数の情報の更新回数が規定値に達すると、前記第１ローカルメモリにおける前記アクセス数の情報の更新を、前記共有メモリにおける前記アクセス数の情報に反映する、ストレージシステム。
【請求項６】
異なる制御情報を担当する複数のプロセッサと、
前記複数のプロセッサのそれぞれに割り当てられており、割り当てられたプロセッサが担当する制御情報を格納するローカルメモリと、
前記複数のプロセッサがアクセス可能であり、前記複数のプロセッサにおける第１プロセッサが担当する制御情報を格納する共有メモリと、を含み、
前記第１プロセッサは、割り当てられている第１ローカルメモリにおいて制御情報を更新し、
前記第１プロセッサは、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定し、
前記第１プロセッサは、前記共有メモリにおける前記制御情報に反映することを決定した前記第１ローカルメモリにおける前記制御情報の更新を、前記共有メモリにおける前記制御情報に反映する、ストレージシステムであって、
前記ストレージシステムは、プライマリボリュームと、当該プライマリボリュームとコピーペアを構成するセカンダリボリュームと、前記プライマリボリュームの更新データを前記セカンダリボリュームにコピーする前に前記更新データを更新順序に従って格納するジャーナルボリュームと、前記ジャーナルボリュームにおける更新データの順序を示すシーケンス番号を含むジャーナル管理情報と、をさらに含み、
前記共有メモリにおける前記制御情報は、前記ジャーナル管理情報における先頭シーケンス番号を示す値を含み、
前記第１ローカルメモリにおける前記制御情報は、前記ジャーナル管理情報における先頭シーケンス番号を示す値と、前記第１ローカルメモリにおける前記値が示す先頭シーケンス番号と前記共有メモリにおける前記値が示す先頭シーケンス番号との差分を示す値と、を含み、
前記第１プロセッサは、前記ジャーナルボリュームへの更新データの格納に応答して、前記第１ローカルメモリにおける前記先頭シーケンス番号を示す値と前記差分を示す値とを更新し、
前記第１プロセッサは、前記差分を示す値が規定値に達すると、前記第１ローカルメモリにおける前記先頭シーケンス番号を示す値の更新を、前記共有メモリにおける前記先頭

シーケンス番号を示す値に反映する、ストレージシステム。
【請求項７】
障害発生に起因して前記第１プロセッサの担当を引きついだ第２プロセッサは、前記共有メモリにおける前記先頭シーケンス番号を示す値を取得し、前記ジャーナル管理情報において、前記取得した値が示す先頭シーケンス番号より先のシーケンス番号領域を検索して、前記ジャーナル管理情報における先頭シーケンス番号を特定する、請求項６に記載のストレージシステム。
【請求項８】
異なる制御情報を担当する複数のプロセッサと、
前記複数のプロセッサのそれぞれに割り当てられており、割り当てられたプロセッサが担当する制御情報を格納するローカルメモリと、
前記複数のプロセッサがアクセス可能であり、前記複数のプロセッサにおける第１プロセッサが担当する制御情報を格納する共有メモリと、を含み、
前記第１プロセッサは、割り当てられている第１ローカルメモリにおいて制御情報を更新し、
前記第１プロセッサは、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定し、
前記第１プロセッサは、前記共有メモリにおける前記制御情報に反映することを決定した前記第１ローカルメモリにおける前記制御情報の更新を、前記共有メモリにおける前記制御情報に反映する、ストレージシステムであって、
前記ストレージシステムは、プライマリボリュームと、前記プライマリボリュームとコピーペアを構成するセカンダリボリュームと、を含み、
前記共有メモリにおける前記制御情報は、それぞれが前記プライマリボリュームの複数の部分領域のそれぞれに対応し、前記プライマリボリュームと前記セカンダリボリュームとの間に差が存在するか否かを示す複数の差分フラグを含み、
前記第１ローカルメモリにおける制御情報は、それぞれが前記プライマリボリュームの前記複数の部分領域のそれぞれに対応し、前記プライマリボリュームと前記セカンダリボリュームとの間に差が存在するか否かを示す複数の差分フラグを含み、
前記第１ローカルメモリにおける制御情報は、それぞれが前記第１ローカルメモリにおける前記複数の差分フラグの一部の複数の差分フラグに対応し、当該一部の複数の差分フラグの更新を前記共有メモリにおける前記制御情報に反映するか否かを示す、複数の反映制御フラグを含み、
前記第１プロセッサは、前記反映制御フラグが反映を指示する差分フラグの更新を、前記共有メモリにおける前記制御情報に反映する、ストレージシステム。
【請求項９】
前記複数の反映制御フラグのそれぞれは、対応する複数の差分フラグにおいて、差が存在することを示す差分フラグの比率が規定値に達している場合に、前記対応する複数の差分フラグの更新を前記共有メモリにおける前記制御情報に反映しないことを示す、請求項８に記載のストレージシステム。
【請求項１０】
障害発生に起因して前記第１プロセッサの担当を引きついだ第２プロセッサは、
前記共有メモリにおける前記複数の反映制御フラグを参照し、
前記参照した反映制御フラグにおいて、前記共有メモリにおける前記制御情報に更新を反映しないことを示す反映制御フラグを特定し、
前記特定した反映制御フラグに対応する前記プライマリボリュームの領域における全データを、前記セカンダリボリュームにコピーする、請求項８に記載のストレージシステム。
【請求項１１】
前記ストレージシステムは、スイッチを含むパスにより接続された第１ストレージモジュールと第２ストレージモジュールとを含み、

前記第１プロセッサ及び前記第１ローカルメモリは前記第１ストレージモジュール内に実装され、
前記共有メモリは前記第２ストレージモジュール内に実装され、
前記第１プロセッサは、前記パスにおける負荷に基づいて前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定する、請求項３に記載のストレージシステム。
【請求項１２】
異なる制御情報を担当する複数のプロセッサと、
前記複数のプロセッサのそれぞれに割り当てられており、割り当てられたプロセッサが担当する制御情報を格納するローカルメモリと、
前記複数のプロセッサがアクセス可能であり、前記複数のプロセッサにおける第１プロセッサが担当する制御情報を格納する共有メモリと、を含み、
前記第１プロセッサは、割り当てられている第１ローカルメモリにおいて制御情報を更新し、
前記第１プロセッサは、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定し、
前記第１プロセッサは、前記共有メモリにおける前記制御情報に反映することを決定した前記第１ローカルメモリにおける前記制御情報の更新を、前記共有メモリにおける前記制御情報に反映する、ストレージシステムであって、
前記第１プロセッサは、前記共有メモリの負荷に基づいて、前記共有メモリにおける制御情報の少なくとも一部の情報の格納領域を、前記共有メモリから異なる種別のデバイスの記憶領域に変更することを決定し、
前記第１プロセッサは、前記他のデバイスの前記記憶領域における前記少なくとも一部の情報を、前記第１ローカルメモリにおける更新に同期して更新する、ストレージシステム。
【請求項１３】
前記少なくとも一部の情報は、前記第１プロセッサが担当する第１ボリュームのデータキャッシング制御情報を含み、
前記第１ローカルメモリにおける前記第１ボリュームのデータキャッシング制御情報の更新を前記共有メモリに反映しないことを決定する条件は、前記第１プロセッサの負荷が第１閾値以上であることを含み、
前記他のデバイスの前記記憶領域は、前記ストレージシステムにおいてボリュームを提供する不揮発性記憶装置の記憶領域であり、
前記少なくとも一部の情報の格納場所を、前記第２共有メモリに変更することを決定する条件は、前記第１プロセッサの負荷が前記第１閾値よりも小さい第２閾値よりも小さいことを含む、請求項１２に記載のストレージシステム。
【請求項１４】
異なる制御情報を担当する複数のプロセッサと、
前記複数のプロセッサのそれぞれに割り当てられており、割り当てられたプロセッサが担当する制御情報を格納するローカルメモリと、
前記複数のプロセッサがアクセス可能であり、前記複数のプロセッサにおける第１プロセッサが担当する制御情報を格納する共有メモリと、を含み、
前記第１プロセッサは、割り当てられている第１ローカルメモリにおいて制御情報を更新し、
前記第１プロセッサは、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定し、
前記第１プロセッサは、前記共有メモリにおける前記制御情報に反映することを決定した前記第１ローカルメモリにおける前記制御情報の更新を、前記共有メモリにおける前記制御情報に反映する、ストレージシステムであって、
前記第１ローカルメモリにおける前記制御情報及び前記共有メモリにおける前記制御情

報は、それぞれ、前記複数のボリュームにおける前記第１プロセッサが担当する第１ボリュームのデータキャッシング制御情報を含み、
前記第１プロセッサは、前記第１ボリュームのデータキャッシングによるアクセス性能の向上に基づき、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定する、ストレージシステム。
【請求項１５】
前記複数の不揮発性半導体メモリそれぞれは、フラッシュメモリである、請求項１に記載のストレージシステム。
【請求項１６】
それぞれが、複数の不揮発性半導体メモリを含む複数の記憶ドライブに基づいて構成される複数のボリュームと、
前記複数のボリュームのデータを一時的に格納するキャッシュメモリと、
複数のプロセッサパッケージと、
共有メモリと、を含み、
前記複数のプロセッサパッケージそれぞれは、前記複数のボリュームに含まれる１つのボリュームに対する入出力を担当するプロセッサと、前記１つのボリュームのデータキャッシング制御情報を格納するローカルメモリと、を含み、
前記共有メモリは、前記複数のプロセッサパッケージそれぞれのローカルメモリに格納されたデータキャッシング制御情報を格納し、前記プロセッサによってアクセス可能であり、
第１プロセッサパッケージに含まれ、第１ボリュームを担当する第１プロセッサは、計算機からの前記第１ボリュームへのリードコマンドを受信すると、前記第１プロセッサパッケージに含まれる第１ローカルメモリに格納されたデータキャッシング制御情報を更新し、
前記第１プロセッサは、前記第１ボリュームの記憶領域を提供する記憶ドライブに、前記リードコマンドにおけるリードデータが格納されているか否かを判定し、
前記第１プロセッサは、前記第１ボリュームの記憶領域を提供する記憶ドライブに前記リードデータが格納されている場合に、前記リードデータを前記キャッシュメモリに格納することなく、かつ前記共有メモリに前記データキャッシング制御情報の更新を反映することなく、前記第１ボリュームの記憶領域を提供する記憶ドライブから読み出された前記リードデータを前記計算機に送信することを決定する、ストレージシステム。
【請求項１７】
前記複数の不揮発性半導体メモリそれぞれは、フラッシュメモリである、請求項１６に記載のストレージシステム。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明はストレージシステムに関し、特に、ストレージシステムの制御に関する。
【背景技術】
【０００２】
国際公開第２０１０／１３１３７３号パンフレット（特許文献１）は、各ボリュームのＩ／Ｏ担当プロセッサが、共有メモリ上のデータキャッシング制御情報をローカルメモリへキャッシング（制御キャッシング）することで、ストレージシステムを高性能化する技術を開示している。
【０００３】
プロセッサは、ローカルメモリの制御情報を更新する時、共有メモリの制御情報も同期して更新する。これにより、障害が起きたプロセッサから担当を引き継いだ他のプロセッサは、共有メモリから最新のデータキャッシング制御情報を取得することができ、キャッシュヒット率低下によるストレージシステムの性能低下、を抑止することができる。
【０００４】

この他、ストレージシステムでは、不揮発性メディアからユーザデータをキャッシュメモリにキャッシングすることでストレージシステムを高性能化する、データキャッシングが広く利用されている。

【先行技術文献】

【特許文献】

【０００５】

【特許文献１】国際公開第２０１０／１３１３７３号パンフレット

【発明の概要】

【発明が解決しようとする課題】

【０００６】

しかし、性能向上が目的である共有メモリにおける制御情報の更新が、アクセス対象である共有メモリとアクセスを制御するプロセッサのオーバヘッドを増加させている。性能向上が目的であるデータキャッシングが、アクセス対象であるキャッシュメモリとアクセスを制御するプロセッサのオーバヘッドを増加させている。

【課題を解決するための手段】

【０００７】

本発明の一態様のストレージシステムは、異なる制御データを担当する複数のプロセッサと、前記複数のプロセッサのそれぞれに割り当てられており、割り当てられたプロセッサが担当する制御情報を格納するローカルメモリと、前記複数のプロセッサがアクセス可能であり、前記複数のプロセッサにおける第１プロセッサが担当する制御情報を格納する共有メモリと、を含む。前記第１プロセッサは、割り当てられている第１ローカルメモリにおいて制御情報を更新する。前記第１プロセッサは、前記第１ローカルメモリにおける前記制御情報の更新を前記共有メモリにおける前記制御情報に反映するか否かを決定する。前記第１プロセッサは、前記共有メモリにおける前記制御情報に反映することを決定した前記第１ローカルメモリにおける前記制御情報の更新を、前記共有メモリにおける前記制御情報に反映する。

【発明の効果】

【０００８】

本発明の一態様は、ストレージシステムにおけるオーバヘッドを低減しストレージシステムの性能を向上する。

【図面の簡単な説明】

【０００９】

【図１】第１実施形態において、計算機システムの全体構成を模式的に示すブロック図である。

【図２】第１実施形態において、ストレージシステムのローカルメモリに格納されている情報を示す図である。

【図３】第１実施形態において、ストレージシステムの共有メモリに格納されている情報を示す図である。

【図４】第１実施形態において、管理計算機の構成を模式的に示す図である。

【図５】第１実施形態において、性能ブースト機能有効化テーブルの一例を示す図である。

【図６】第１実施形態において、ボリューム毎性能ブースト機能有効化テーブルの一例を示す図である。

【図７】第１実施形態において、メディア種別テーブルの一例を示す図である。

【図８】第１実施形態において、ＲＡＩＤレベルテーブルの一例を示す図である。

【図９】第１実施形態において、ボリューム毎ヒット率テーブルの一例を示す図である。

【図１０】第１実施形態において、ヒット率閾値テーブルの一例を示す図である。

【図１１】第１実施形態において、ＭＰ稼働率テーブルの一例を示す図である。

【図１２】第１実施形態において、ＭＰ稼働率閾値テーブルの一例を示す図である。

【図１３】第１実施形態において、ＣＭ稼働率テーブルの一例を示す図である。

【図１４】第１実施形態において、ＣＭ稼働率閾値テーブルの一例を示す図である。
【図１５】第１実施形態におけるホストからのリードコマンドの処理のフローチャートである。
【図１６】第１実施形態におけるデータキャッシングに関するＳＭ制御情報更新判定処理のフローチャートである。
【図１７】第１実施形態におけるホストデータキャッシング判定処理のフローチャートである。
【図１８Ａ】第１実施形態において、ホストからのライトコマンドの処理のフローチャートの一部である。
【図１８Ｂ】第１実施形態において、ホストからのライトコマンドの処理のフローチャートの他の一部である。
【図１９】第１実施形態における管理計算機からの設定処理のフローチャートである。
【図２０】第１実施形態において、管理計算機における設定メニュー画面の一例を示す図である。
【図２１】第１実施形態におけるメディア種別テーブルの更新処理のフローチャートである。
【図２２】第１実施形態におけるＣＭＰＫ稼働率更新処理のフローチャートである。
【図２３】第１実施形態におけるヒット率更新処理のフローチャートである。
【図２４】第１実施形態におけるＭＰ稼働率更新処理のフローチャートである。
【図２５】第１実施形態におけるオーナ移動時ＳＭ更新処理のフローチャートである。
【図２６】第２実施形態において、ローカルメモリに格納されている情報を示す図である。
【図２７】第２実施形態において、共有メモリに格納されている情報を示す図である。
【図２８】第２実施形態において、ダイナミックマッピングテーブルの一例を示す図である。
【図２９】第２実施形態において、ページ毎モニタテーブルの一例を示す図である。
【図３０】第２実施形態において、ページ毎モニタ差分テーブルの一例を示す図である。
【図３１】第２実施形態におけるストレージ階層仮想化機能モニタ更新処理のフローチャートである。
【図３２】第３実施形態において、計算機システム全体構成を模式的に示す図である。
【図３３】第３実施形態において、非同期リモートコピーを説明する図である。
【図３４】第３実施形態において、ローカルメモリに格納されている情報を示す図である。
【図３５】第３実施形態において、共有メモリに格納されている情報を示す図である。
【図３６】第３実施形態において、ＬＭ非同期リモートコピーシーケンス番号管理テーブルの一例を示す図である。
【図３７】第３実施形態において、ＳＭ非同期リモートコピーシーケンス番号管理テーブルの一例を示す図である。
【図３８】第３実施形態における非同期リモートコピーシーケンス番号更新処理のフローチャートである。
【図３９】第３実施形態におけるＭＰＰＫ障害時非同期リモートコピーシーケンス番号回復処理のフローチャートである。
【図４０】第４実施形態において、ローカルメモリに格納されている情報を示す図である。
【図４１】第４実施形態において、共有メモリに格納されている情報を示す図である。
【図４２】第４実施形態において、ＬＭローカルコピー差分管理テーブルの一例を示す図である。
【図４３】第４実施形態において、ＳＭローカルコピー差分管理テーブルの一例を示す図である。
【図４４】第４実施形態において、ＬＭローカルコピー差分領域間引き動作管理テーブル

の一例を示す図である。
【図４５】第４実施形態において、ＳＭローカルコピー差分領域間引き動作管理テーブルの一例を示す図である。
【図４６】第４実施形態における非同期ローカルコピー差分管理情報更新処理のフローチャートである。
【図４７】第４実施形態におけるＭＰＰＫ障害時ローカルコピー差分コピー処理のフローチャートである。
【図４８】第４実施形態において、管理計算機における設定メニュー画面の一例を示す図である。
【図４９】第５実施形態において、計算機システムの全体構成を模式的に示す図である。
【図５０】第５実施形態において、ローカルメモリに格納されている情報を示す図である。
【図５１】第５実施形態において、Ｘパス稼働率テーブルの一例を示す図である。
【図５２】第５実施形態において、Ｘパス稼働率閾値テーブルの一例を示す図である。
【図５３】第５実施形態におけるＸパスを考慮したデータキャッシングに関するＳＭ制御情報更新判定処理のフローチャートである。
【図５４】第５実施形態におけるＸパス稼働率更新処理のフローチャートである。
【図５５】第６実施形態において、計算機システムの全体構成を模式的に示す図である。
【図５６】第６実施形態において、ローカルメモリに格納されている情報を示す図である。
【図５７】第６実施形態において、ＭＰ稼働率テーブルの一例を示す図である。
【図５８】第６実施形態において、ＭＰ稼働率閾値テーブルの一例を示す図である。
【図５９】第６実施形態において、共有メモリ領域管理テーブルの一例を示す図である。
【図６０Ａ】第６実施形態におけるデータキャッシングに関するＳＭ制御情報更新判定処理のフローチャートの一部である。
【図６０Ｂ】第６実施形態におけるデータキャッシングに関するＳＭ制御情報更新判定処理のフローチャートの他の一部である。
【図６１】第６実施形態におけるＭＰ稼働率更新処理のフローチャートである。
【図６２】第７実施形態において、ローカルメモリに格納されている情報を示す図である。
【図６３】第７実施形態において、レスポンステーブルの一例を示す図である。
【図６４】第７実施形態において、ＣＭ利用閾値テーブルの一例を示す図である。
【図６５】第７実施形態におけるヒット率更新処理のフローチャートである。
【発明を実施するための形態】
【００１０】
本発明は、ストレージシステムの性能を向上するための技術に関する。以下、添付図面を参照して本発明の実施形態を説明する。説明の明確化のため、以下の記載及び図面の詳細は、適宜、省略及び簡略化がなされており、必要に応じて重複説明は省略されている。本実施形態は本発明を実現するための一例に過ぎず、本発明の技術的範囲を限定するものではない。
【００１１】
第１実施形態
本実施形態のストレージシステムは、それぞれが異なるボリュームの入出力（Ｉ／Ｏ）を担当するプロセッサを含む。各プロセッサは、ローカルメモリが割り当てられている。本実施形態のストレージシステムは、異なるボリュームを担当する複数のプロセッサがアクセス可能な共有メモリを有する。ローカルメモリ及び共有メモリは、典型的には揮発性半導体メモリである。
【００１２】
プロセッサが担当するボリュームのデータキャッシング制御情報は、当該プロセッサのローカルメモリに格納されている（制御データキャッシング）。さらに、共有メモリは、

当該ボリュームのデータキャッシング制御情報を格納する。

【００１３】

プロセッサは、ローカルメモリ上のキャッシング制御情報を参照及び更新して、担当ボリュームのデータキャッシング制御を行う。これにより、データキャッシング制御の処理を高速化することができる。

【００１４】

上述のように、共有メモリは、異なるボリュームを担当する複数のプロセッサがアクセスすることができる。いずれかのボリュームの担当プロセッサについて障害が発生した場合、他のプロセッサがその担当を引き継ぎ、共有メモリから対応するデータキャッシング制御情報を自身のローカルメモリにロードする。上記他のプロセッサは、共有メモリから取得したデータキャッシング制御情報を使用して、引き継いだボリュームのデータキャッシングを制御する。

【００１５】

本実施形態において、プロセッサは、予め定められている条件に従って、ローカルメモリにおけるキャッシング制御情報の更新を、共有メモリにおける制御情報に反映するか否かを決定する。ローカルメモリにおける制御情報の更新において必要な更新のみを共有メモリにおける制御情報に反映することで、プロセッサと共有メモリの通信におけるオーバヘッドを低減し、ストレージシステムの性能を向上することができる。

【００１６】

さらに、本実施形態のストレージシステムは、リードデータ及びライトデータのキャッシングの有無を、予め定められている条件に従って決定する。リードデータ及びライトデータを選択的にキャッシングすることで、キャッシュ領域を効率的に利用し、さらに、キャッシュメモリ及びデータキャッシングを行うプロセッサのオーバヘッドを低減することで、ストレージシステムの性能を向上する。

【００１７】

以下において、図１から図２５を参照して本実施形態を具体的に説明する。図１は、本実施形態のストレージシステム１０、データ処理及び演算を行うホスト計算機１８０及びストレージシステム１０を管理する管理計算機２０を含む、計算機システムの一例を示す。計算機システムは、複数のホスト計算機１８０を含むことができる。

【００１８】

ストレージシステム１０とホスト計算機１８０とは、データネットワーク１９０を介して互いに接続される。データネットワーク１９０は、例えば、ＳＡＮ（Ｓｔｏｒａｇｅ Ａｒｅａ Ｎｅｔｗｏｒｋ）である。データネットワーク１９０は、ＩＰネットワーク若しくはその他のいかなる種類のデータ通信用のネットワークであってもよい。

【００１９】

ストレージシステム１０、ホスト計算機１８０及び管理計算機２０は、管理ネットワーク（不図示）を介して互いに接続される。管理ネットワークは、例えば、ＩＰネットワークである。管理ネットワークは、ＳＡＮ若しくはその他のいかなる種類のネットワークであってもよい。データネットワーク１９０と管理ネットワークとが同一のネットワークであってもよい。

【００２０】

ストレージシステム１０は、複数の記憶ドライブ１７０を収容している。記憶ドライブ１７０は、不揮発性の磁気ディスクを有するハードディスクドライブ（ＨＤＤ）、不揮発半導体メモリ（例えばフラッシュメモリ）を搭載したＳｏｌｉｄ Ｓｔａｔｅ Ｄｒｉｖｅ（ＳＳＤ）を含む。記憶ドライブ１７０は、ホスト計算機１８０から送られたデータ（ユーザデータ）を格納する。複数の記憶ドライブ１７０がＲＡＩＤ演算によるデータの冗長化を行うことで、１つの記憶ドライブ１７０に障害が発生した場合のデータ消失を防ぐことができる。

【００２１】

ストレージシステム１０は、ホスト計算機１８０と接続するためのフロントエンドパッ

ケージ（ＦＥＰＫ）１００、記憶ドライブ１７０と接続するためのバックエンドパッケージ（ＢＥＰＫ）１４０、キャッシュメモリを搭載するキャッシュメモリパッケージ（ＣＭＰＫ）１３０、内部処理を行うマイクロプロセッサを搭載するマイクロプロセッサパッケージ（ＭＰＰＫ）１２０、及びそれらを接続する内部ネットワーク１５０を有する。図１に示すように、本例のストレージシステム１０は、複数のＦＥＰＫ１００、複数のＢＥＰＫ１４０、複数のＣＭＰＫ１３０、そして複数のＭＰＰＫ１２０を含む。

【００２２】

各ＦＥＰＫ１００は、ホスト計算機との接続を行うためのインタフェース１０１、ストレージシステム１０内のデータ転送を行うための転送回路１１２を基板上に有する。インタフェース１０１は複数のポートを含むことができ、各ポートがホスト計算機と接続することができる。インタフェース１０１は、ホスト計算機１８０とストレージシステム１０との間の通信に用いられるプロトコル、例えばＦｉｂｒｅ Ｃｈａｎｎｅｌ Ｏｖｅｒ Ｅｔｈｅｒｎｅｔ（ＦＣｏＥ）を、内部ネットワーク１５０で用いられるプロトコル、例えばＰＣＩ－Ｅｘｐｒｅｓｓに変換する。

【００２３】

各ＢＥＰＫ１４０は、ドライブ１７０と接続するためにインタフェース１４１、ストレージシステム１０内のデータ転送を行うための転送回路１４２を基板上に有する。インタフェース１４１は複数ポートを含むことができ、各ポートがドライブ１７０と接続することができる。インタフェース１４１は、記憶ドライブ１７０との通信に用いられるプロトコル、例えばＦＣを、内部ネットワーク１５０で用いられるプロトコルに変換する。

【００２４】

各ＣＭＰＫ１３０は、ホスト計算機１８０から読み書きされるユーザデータを一時的に格納するキャッシュメモリ（ＣＭ）１３１及び１又は複数のＭＰＰＫ１２０が扱う制御情報を格納する共有メモリ（ＳＭ）１３２を基板上に有する。異なるボリュームを担当する複数のＭＰＰＫ１２０（のマイクロプロセッサ）が、共有メモリ１３２にアクセスすることができる。ＭＰＰＫ１２０が扱うデータやプログラムは、ストレージシステム１０内の不揮発性メモリ（不図示）又は記憶ドライブ１７０からロードされる。関連づけられるキャッシュメモリ１３１と共有メモリ１３２とは、別の基板上（パッケージ内）に実装されていてもよい。

【００２５】

各ＭＰＰＫ１２０は、１以上のマイクロプロセッサ１２１、ローカルメモリ（ＬＭ）１２２及びそれらを接続するバス１２３を有する。本例は複数のマイクロプロセッサ１２１が実装されている。マイクロプロセッサ１２１の数は１つでもよい。複数のマイクロプロセッサ１２１を一つのプロセッサと見ることができる。ローカルメモリ１２２は、マイクロプロセッサ１２１が実行するプログラム及びマイクロプロセッサ１２１が使用する制御情報を格納する。

【００２６】

上述のように、一つの共有メモリ１３２は、ＭＰＰＫ１２０が扱う制御情報を格納する。ＭＰＰＫ１２０は、共有メモリ１３２から、自身が必要とする制御情報を自身のローカルメモリ１２２に格納する（制御キャッシング）。これにより、マイクロプロセッサ１２１による制御情報への高速アクセスが実現され、ストレージシステム１０の性能を向上することができる。

【００２７】

マイクロプロセッサ１２１は、ローカルメモリ１２２の制御情報を更新すると、その更新を、必要により、共有メモリ１３２上の制御情報に反映する。本実施形態の特徴の一つは、この更新の制御である。マイクロプロセッサ１２１は、予め定められた条件が満たされている場合に、ローカルメモリ１２２における制御情報の更新を共有メモリ１３２における制御情報に反映する。

【００２８】

本構成例において、マイクロプロセッサ１２１には、ストレージシステム１０がホスト

計算機１８０に提供するボリュームの担当が割り当てられる。マイクロプロセッサ１２１に割り当てられているローカルメモリ１２２及び共有メモリ１３２が、上記マイクロプロセッサがＩ／Ｏを担当するボリュームのデータキャッシング制御情報を格納する。

【００２９】

なお、本発明を適用できる制御情報は、共有メモリ１３２における制御情報を更新しない場合でもＭＰ障害時にホストデータロストにつながらない制御情報全般である。本実施形態におけるデータキャッシング制御情報以外の制御情報の例は、他の実施形態で説明する。本実施形態はマイクロプロセッサがボリュームを担当する例を記載しているが、担当マイクロプロセッサが割り当てられる対象はボリュームに限定されず、担当マイクロプロセッサは制御情報毎に存在してもよい。

【００３０】

図２は、ローカルメモリ１２２内に格納されている情報を示すブロック図である。ローカルメモリ１２２は、性能ブースト機能有効化テーブル２１０、ボリューム毎性能ブースト機能有効化テーブル２２０、メディア種別テーブル２３０、ＲＡＩＤレベルテーブル２４０、ボリューム毎ヒット率テーブル２５０、ヒット率閾値テーブル２６０、マイクロプロセッサ（ＭＰ）稼働率テーブル２７０を格納する。

【００３１】

ローカルメモリ１２２は、さらに、マイクロプロセッサ（ＭＰ）稼働率閾値テーブル２８０、キャッシュメモリ（ＣＭ）稼働率テーブル２９０、キャッシュメモリ（ＣＭ）稼働率閾値テーブル３００を含む。例えば、マイクロプロセッサ１２１は、記憶ドライブ１７０その他のストレージシステム１０内の不揮発性記憶領域からこれらテーブルの少なくとも一部を取得して、ローカルメモリ１２２に格納し、いくつかのテーブルを新たに作成する。これらテーブルの詳細な説明は後述する。

【００３２】

ローカルメモリ１２２は、さらに、キャッシュディレクトリ３１０を格納している。図３は、共有メモリ１３２内のキャッシュディレクトリ５１０を示すブロック図である。マイクロプロセッサ１２１は、共有メモリ１３２からキャッシュディレクトリ５１０を自身のローカルメモリ１２２にキャッシュし、ローカルメモリ１２２上のキャッシュディレクトリ３１０の更新を、必要により、共有メモリ１３２のキャッシュディレクトリ５１０に反映する。キャッシュディレクトリ５１０は、キャッシュディレクトリ３１０のバックアップデータである。

【００３３】

マイクロプロセッサ１２１は、ホスト計算機１８０からリードコマンドを受信すると、そのローカルメモリ１２２のキャッシュディレクトリ３１０を参照して、対象データがキャッシュメモリ１３１にキャッシュされているか（キャッシュヒットか）を決定する。このように、キャッシュディレクトリ３１０は、キャッシュメモリ１３１に格納されているキャッシュデータを検索するための情報を与える。

【００３４】

キャッシュディレクトリ３１０は、参照テーブルＧＲＰＰ（GRouP Pointer）、ＧＲＰＴ（GRouP Table）１、ＧＲＰＴ２、管理テーブルとしてのスロットコントロールブロック（ＳＬＣＢ）から構成されている。参照テーブルＧＲＰＰ、ＧＲＰＴ１、ＧＲＰＴ２は、キャッシュセグメントを検索するときにマイクロプロセッサ１２１により参照されるテーブルであって、ディレクトリ構造を有する。参照テーブルＧＲＰＰが最上位に位置し、参照テーブルＧＲＰＴ２が最下位に位置している。上位のテーブルは、次のテーブルのポインタを含む。ＧＲＰＴ２には、ＳＬＣＢへのポインタが含まれている。

【００３５】

ＳＬＣＢは、キャッシュ管理の最小単位であるセグメントに関する制御情報を管理するテーブルであって、キャッシュメモリ１３１上にリードコマンドの指定データが存在するか否か、キャッシュされているデータのキャッシュメモリ１３１上のアドレス情報等、が格納されている。

【００３６】
１つのスロットには、１又は複数のセグメントを関連付けることができる。１つのセグメントには、例えば、６４ＫＢのデータを格納することができる。キャッシュ管理の最小単位はセグメントであるが、キャッシュをスロット単位で管理してもよい。典型的には、ダーティデータ（物理ディスクへの書込み前の状態）と、クリーンデータ（物理ディスクへの書込み後の状態）の各状態間の遷移は、スロット単位で行われる。キャッシュ領域のリザーブ及びリリースは、スロット単位又はセグメント単位で行われる。
【００３７】
ホスト計算機１８０からリードアクセスがあった場合は、マイクロプロセッサ１２１は、それに含まれる論理ブロックアドレス（ＬＢＡ）に基づいて、各階層テーブルを順番に辿ることにより、要求されたデータがキャッシュメモリ１３１に存在するか、存在する場合にはそのアドレスを知ることができる。
【００３８】
要求されたデータがキャッシュメモリ１３１に存在する場合、マイクロプロセッサ１２１は、そのデータをホスト計算機１８０に送信する。要求されたデータがキャッシュメモリ１３１に存在しない場合、マイクロプロセッサ１２１は、ホスト計算機１８０が要求しているデータを記憶ドライブ１７０から読み出して、キャッシュ領域上の１つまたは複数のスロットに格納する。ライトデータも同様にキャッシュされる。なお、キャッシュディレクトリを使用したキャッシュデータの検索は広く知られた技術であり、ここでの詳細な説明を省略する。
【００３９】
図４は、管理計算機２０の構成を模式的に示すブロック図である。管理計算機２０は、入力インタフェース２２、入力デバイス２８、表示インタフェース２３、表示デバイス２９、ＣＰＵ２６、通信インタフェース２１、メモリ２４、ＨＤＤ２５を備える。入力デバイス２８の典型的な例は、キーボード及びポインタデバイスであるが、これらと異なるデバイスでもよい。表示デバイス２９は、典型的には、液晶表示装置である。
【００４０】
管理者（ユーザ）は、表示デバイス２９によって処理結果を視認しながら、入力デバイス２８によって必要なデータを入力する。管理者が入力する情報及び表示デバイス２９による表示例は、後述する。図１の計算機システムにおいて、管理システムは一つの管理計算機２０で構成されているが、管理システムは、管理計算機２０に加え、管理コンソールを含むことができる。管理コンソールは、入力デバイス及び表示デバイスを含み、ネットワークを介して管理計算機２０に接続する。
【００４１】
管理者は管理コンソールから管理計算機２０にアクセスし、管理計算機２０に処理を指示するとともに、管理コンソールに管理計算機２０の処理結果を取得、表示させる。管理システムは、それぞれが管理計算機２０の機能の一部又は全部を備える複数の計算機を含むこともできる。ＣＰＵ２６は、メモリ２４に格納されたプログラムを実行するプロセッサである。通信Ｉ／Ｆ２１は、管理ネットワークとのインタフェースであって、システム管理のためにホスト計算機１８０及びストレージシステム１０と、データや制御命令の送受信を行う。
【００４２】
図５は、性能ブースト機能有効化テーブル２１０の構成例を示している。性能ブースト機能有効化テーブル２１０は、性能ブースト機能有効フラグのカラム２１１を有する。性能ブースト機能有効フラグは、ストレージシステム１０全体の性能ブースト機能がアクティブであるか否かを示す。このフラグが１である場合、ストレージシステム１０全体の性能ブースト機能がアクティブである。
【００４３】
本実施形態において、性能ブースト機能は、ローカルメモリ１２２に格納された制御情報更新の共有メモリ１３２への反映（バックアップ）の制御及びデータキャッシング制御

の機能である。この機能については後述する。性能ブースト機能有効化テーブル２１０のデータは、例えば、管理者が管理計算機２０から設定する。

【００４４】

図６は、ボリューム毎性能ブースト機能有効化テーブル２２０の構成例を示している。ボリューム毎性能ブースト機能有効化テーブルテーブル２２０は、ボリューム毎の性能ブースト機能を管理する。ボリューム毎性能ブースト機能有効化テーブル２２０は、論理ボリューム番号のカラム２２１及び性能ブースト機能有効化フラグのカラム２２２を有する。論理ボリューム番号は、論理ボリュームの識別子である。

【００４５】

性能ブースト機能有効化フラグが１である場合、そのボリュームの性能ブースト機能がアクティブであることを示す。システム全体及びボリュームの性能ブースト機能有効化フラグの双方がＯＮ（１）である場合、そのボリュームの性能ブースト機能が有効化される。このように、ボリューム毎に性能ブースト機能を管理、制御することで、ボリューム特性に応じた制御が実現される。ボリューム毎性能ブースト機能有効化テーブル２２０のデータは、例えば、管理者が管理計算機２０から設定する。

【００４６】

図７は、メディア種別テーブル２３０の構成例を示している。メディア種別テーブル２３０は、ＲＡＩＤグループのメディア種別を管理する。本実施形態において、１又は複数の記憶ドライブ１７０が提供する記憶領域及びそのインタフェースを含む構成をメディアと呼ぶ。メディア種別テーブル２３０は、ＲＡＩＤグループ番号のカラム２３１及びメディア種別のカラム２３２を含む。

【００４７】

ＲＡＩＤグループ番号は、ＲＡＩＤグループを一意に識別する識別子である。なお、本明細書において、対象を識別する識別情報のために、識別子、名、ＩＤ等の表現を用いることができ、これらは置換可能である。メディア種別テーブル２３０のデータは、例えば、管理者が管理計算機２０から設定する。

【００４８】

図８は、ＲＡＩＤレベルテーブル２４０の構成例を示している。ＲＡＩＤレベルテーブル２４０は、ＲＡＩＤグループのＲＡＩＤレベルを管理する。ＲＡＩＤグループ番号のカラム２４１及びＲＡＩＤレベルのカラム２４２を有する。ＲＡＩＤレベルテーブル２４０のデータは、例えば、管理者が管理計算機２０から設定する。

【００４９】

図９は、ボリューム毎ヒット率テーブル２５０の構成例を示している。ボリューム毎ヒット率テーブル２５０は、各ボリュームのキャッシュヒット率を管理する。ボリューム毎ヒット率テーブル２５０は、論理ボリューム番号のカラム２５１、ヒット率のカラム２５２、Ｉ／Ｏ数のカラム２５３、ヒット数のカラム２５４、低ヒット率フラグのカラム２５５を有する。

【００５０】

Ｉ／Ｏ数は、論理ボリュームに対して発行されたリードコマンド数である。ヒット数は、キャッシュヒットしたリードコマンド数である。低ヒット率フラグが１である場合、そのエントリのヒット率が、規定閾値未満であることを示す。マイクロプロセッサ１２１は、ボリュームへのリードアクセス及びキャッシュヒット数をカウントし、ボリューム毎ヒット率テーブル２５０の各フィールドのデータを更新する。

【００５１】

なお、マイクロプロセッサ１２１がヒット率をモニタする単位は、論理ボリュームより小さい単位でもよい。例えば、仮想ボリューム機能や階層化機能で使用されるページを単位としてもよい。

【００５２】

ヒット率の算出は、リードキャッシュのヒット率の他にライトキャッシュのヒット率を含んでもよい。ライトキャッシュのヒット率[%]は、１００×（１－(記憶ドライブへのラ

イト処理回数／ライトコマンドＩ／Ｏ数)で表すことができる。マイクロプロセッサ１２１は、リードキャッシュのヒット率とライトキャッシュのヒット率を個別に管理してもよい。例えば、マイクロプロセッサ１２１は、後述のリードキャッシング制御及びライトキャシング制御において、それぞれのヒット率を参照する。

【００５３】

図１０は、ヒット率閾値テーブル２６０の構成例を示している。ヒット率閾値テーブル２６０は、ヒット率閾値のカラム２６１を有する。ヒット率がここに登録されている閾値以下である場合、ボリューム毎ヒット率テーブル２５０におけるそのエントリの低ヒット率フラグが１（ＯＮフラグ）に設定される。ヒット率閾値は、例えば、管理者が管理計算機２０から設定する。

【００５４】

図１１は、マイクロプロセッサ１２１の稼働率を管理するＭＰ稼働率テーブル２７０の構成例を示している。ＭＰ稼働率は、単位時間内のマイクロプロセッサ１２１の処理時間であり、マイクロプロセッサの負荷を表す。ＭＰ稼働率テーブル２７０は、マイクロプロセッサ番号のカラム２７１、稼働率のカラム２７２、過負荷判定フラグのカラム２７３、稼働時間のカラム２７４を有する。マイクロプロセッサ番号は、ストレージシステム１０内で一意にマイクロプロセッサを識別する。

【００５５】

各マイクロプロセッサ１２１は、自身の稼働状況を監視し、稼働率及び稼働時間の値を、自身のエントリの稼働率のカラム２７２及び稼働時間のカラム２７４のフィールドに格納する。稼働時間は、単位時間（本例で１秒）当たりの稼働時間である。稼働率は、単位時間で稼働時間を割った値である。マイクロプロセッサ１２１は、自己の稼働率を規定の閾値と比較し、その閾値以上である場合に、自己エントリの過負荷判定フラグのフィールの値を１（ＯＮフラグ）に設定する。

【００５６】

図１２は、上記閾値を格納するカラム２８１を有する、ＭＰ稼働率閾値テーブル２８０の構成例を示している。本例において、ＭＰ稼働率閾値は、全てのマイクロプロセッサに共通であるが、異なる閾値を使用してもよい。

【００５７】

図１３は、キャッシュメモリの稼働率を管理する、ＣＭ稼働率テーブル２９０の構成例を示している。ＣＭ稼働率は、単位時間内のキャッシュメモリ１３１へのアクセス時間である。ＣＭ稼働率テーブル２９０は、ＣＭＰＫ番号のカラム２９１、稼働率のカラム２９２、過負荷判定フラグのカラム２９３を有する。ＣＭＰＫ番号は、ストレージシステム１０内のＣＭＰＫの識別子である。

【００５８】

マイクロプロセッサ１２１は、ＣＭＰＫ１３０上のコントローラから、その稼働率の値を取得し、稼働率のカラム２９２の該当フィールドにそれを格納する。マイクロプロセッサ１２１は、取得した稼働率の値を規定閾値と比較し、稼働率の値が閾値以上である場合に、そのエントリの過負荷判定フラグのフィールドに１（ＯＮフラグ）を設定する。

【００５９】

図１４は、上記閾値を格納するＣＭ稼働率閾値テーブル３００の構成例を示している。本例において、ＣＭ稼働率閾値は、全てのＣＭＰＫに共通であるが、異なる閾値を使用してもよい。

【００６０】

図１５のフローチャートを参照して、ストレージシステム１０がホスト計算機１８０から受けたリードコマンドに対して行う処理を説明する。ホスト計算機１８０からのリードコマンドを受けた（Ｓ１０１）マイクロプロセッサ１２１は、リードコマンドが示す論理ボリューム（ＬＤＥＶとも呼ぶ）に、自身がアクセス権を有するか判定する（Ｓ１０２）。アクセス権を有していない場合（Ｓ１０２：ＮＯ）、そのマイクロプロセッサ１２１は、アクセス権を有するＭＰＰＫ１２０にリードコマンドを転送する（Ｓ１０３）。

【００６１】
マイクロプロセッサ１２１がアクセス権を有する場合（Ｓ１０２：ＹＥＳ）、そのマイクロプロセッサ１２１は、同一ＭＰＰＫ１２０上のローカルメモリ１２２内で、キャッシュディレクトリ３１０を検索する（Ｓ１０４）。リードコマンドが指定するアドレス（データ）が見つかった場合（Ｓ１０５：ＹＥＳ）、マイクロプロセッサ１２１は、キャッシュディレクトリ３１０の情報に従って、キャッシュメモリ１３１からリードデータを読み出し、ホスト計算機１８０に送信する（Ｓ１０６）。
【００６２】
リードコマンドが指定するアドレス（データ）が見つからなかった（キャッシュミス）場合（Ｓ１０５：ＮＯ）、マイクロプロセッサ１２１は、ローカルメモリ１２２の未キャッシュフラグを確認する（Ｓ１０７）。未キャッシュフラグは、共有メモリ１３２のキャッシュディレクトリ５１０の全てのデータが、ローカルメモリ１２２にキャッシュされているかを示すフラグであり、ローカルメモリ１２２内に格納されている。一部のデータが読み込まれていない場合、その値はＯＮである。例えば、障害フェイルオーバ直後で共有メモリ１３２からローカルメモリ１２２へ制御情報が読み込まれてない場合、未キャッシュフラグはＯＮである。
【００６３】
未キャッシュフラグがＯＮである場合（Ｓ１０７：ＹＥＳ）、共有メモリ１３２のキャッシュディレクトリ５１０の一部データがキャッシュされていない。マイクロプロセッサ１２１は、ＣＭＰＫ１３０のコントローラを介して、共有メモリ１３２からローカルメモリ１２２へキャッシュディレクトリ（制御情報）を転送する（Ｓ１０８）。
【００６４】
マイクロプロセッサ１２１は、ローカルメモリ１２２内で、キャッシュディレクトリ３１０を検索する（Ｓ１０９）。リードコマンドが指定するデータが見つかった場合（Ｓ１１０：ＹＥＳ）、マイクロプロセッサ１２１は、キャッシュディレクトリ３１０の情報に従って、キャッシュメモリ１３１からリードデータを読み出し、ホスト計算機１８０に送信する（Ｓ１１１）。
【００６５】
キャッシュミスの場合（Ｓ１１０：ＮＯ）又は未キャッシュフラグがＯＦＦの場合（Ｓ１０７：ＮＯ）、マイクロプロセッサ１２１は、キャッシュメモリ１３１のセグメントをリードデータのためのスロットとして確保し、さらに、ローカルメモリ１２２のキャッシュディレクトリ３１０を更新する（Ｓ１１２）。
【００６６】
マイクロプロセッサ１２１は、データキャッシングに関する制御情報であるキャッシュディレクトリ３１０の更新を、共有メモリ１３２のデータ５１０に反映するか否かを判定する（Ｓ１１３）。この判定の具体的な方法については後に詳述する。共有メモリ１３２の制御情報の更新を行うと判定した場合（Ｓ１１４：ＹＥＳ）、マイクロプロセッサ１２１は、共有メモリ１３２のキャッシュディレクトリ５１０を更新して（Ｓ１１５）、次のステップＳ１１６に進む。
【００６７】
共有メモリ１３２の制御情報の更新を行わないと判定した場合（Ｓ１１４：ＮＯ）、マイクロプロセッサ１２１は、共有メモリ１３２の制御情報を更新することなく、ステップＳ１１６に進む。ステップＳ１１６において、マイクロプロセッサ１２１は、リードデータ（ホストデータ）をキャッシングするか否かを判定する。この判定方法については後述する。
【００６８】
リードデータをキャッシュメモリ１３１に格納してからホスト計算機１８０に送信すると判定した場合（Ｓ１１７：ＹＥＳ）、マイクロプロセッサ１２１は、ＢＥＰＫ１４０及びＣＭＰＫ１３０により、記憶ドライブ１７０（永続メディア）からリードデータを読み出し、キャッシュメモリ１３１上の確保したセグメントに格納する。その後、マイクロプ

ロセッサ１２１は、そのキャッシュデータを、ＣＭＰＫ１３０及びＦＥＰＫ１００により、ホスト計算機１８０に送信する（Ｓ１１８）。

【００６９】

リードデータをキャッシュすることなくホスト計算機１８０に送信すると判定した場合（Ｓ１１７：ＮＯ）、マイクロプロセッサ１２１は、ＢＥＰＫ１４０及びＦＥＰＫ１００により、ドライブ１７０（永続メディア）から読みだしたリードデータを、ＣＭＰＫ１３０を介することなくホスト計算機１８０に転送する（Ｓ１１９）。キャッシュメモリ１３１を経由しないデータのセグメントは、キャッシュメモリ１３１を経由するデータのセグメントと比較して、他のデータ用に再利用されやすくなるように管理するとより効率的である。例えば、ＬＲＵキュー管理をしている場合、ＬＲＵに接続することが考えられる。

【００７０】

図１６を参照して、図１５のフローチャートにおける、共有メモリ１３２内のデータキャッシング制御情報の更新についての判定（Ｓ１１３）を説明する。マイクロプロセッサ１２１は、このステップＳ１１３を開始し（Ｓ１２１）、リードコマンドの指定する論理ボリュームの性能ブースト機能がＯＮであるか否かを、性能ブースト機能有効化テーブル２１０及びボリューム毎性能ブースト機能有効化テーブル２２０を参照して判定する（Ｓ１２２）。一方のテーブルが、性能ブースト機能がＯＦＦであることを示す場合、当該ボリュームの性能ブースト機能はＯＦＦである。

【００７１】

当該論理ボリュームの性能ブースト機能がＯＮではない場合（Ｓ１２２：ＮＯ）、マイクロプロセッサ１２１は、共有メモリ１３２の制御情報（キャッシュディレクトリ）を更新することを決定する（Ｓ１２８）。当該論理ボリュームの性能ブースト機能がＯＮである場合（Ｓ１２２：ＹＥＳ）、マイクロプロセッサ１２１は、次に、指定データが格納されているＲＡＩＤグループのメディア種別がＳＳＤであるか否かを、ＲＡＩＤグループ番号をキーとしてメディア種別テーブル２３０を参照し、判定する（Ｓ１２３）。

【００７２】

マイクロプロセッサ１２１は、ローカルメモリ１２２内に、各ボリュームの構成管理情報を有しており、各ボリュームの各領域がいずれのＲＡＩＤグループに属するかをその情報を参照して知ることができる。

【００７３】

メディア種別がＳＳＤである場合（Ｓ１２３：ＹＥＳ）、マイクロプロセッサ１２１は、共有メモリ１３２の制御情報（キャッシュディレクトリ）を更新しないことを決定する（Ｓ１２７）。そのメディア種別がＳＳＤではない場合（Ｓ１２３：ＮＯ）、マイクロプロセッサ１２１は、次に、指定データが格納されている論理ボリュームの低ヒット率フラグがＯＮであるか否かを、論理ボリューム番号をキーとしてボリューム毎ヒット率テーブル２５０を参照し、判定する（Ｓ１２４）。

【００７４】

その低ヒット率フラグがＯＮである場合（Ｓ１２４：ＹＥＳ）、マイクロプロセッサ１２１は、共有メモリ１３２の制御情報（キャッシュディレクトリ）を更新しないことを決定する（Ｓ１２７）。低ヒット率フラグがＯＦＦである場合（Ｓ１２４：ＮＯ）、マイクロプロセッサ１２１は、次に、自身の過負荷フラグがＯＮであるか否かを、マイクロプロセッサ番号をキーとしてＭＰ稼働率テーブル２７０を参照し、判定する（Ｓ１２５）。

【００７５】

過負荷フラグがＯＮである場合（Ｓ１２５：ＹＥＳ）、マイクロプロセッサ１２１は、共有メモリ１３２の制御情報（キャッシュディレクトリ）を更新しないことを決定する（Ｓ１２７）。過負荷フラグがＯＦＦである場合（Ｓ１２５：ＮＯ）、マイクロプロセッサ１２１は、次に、アクセス先のＣＭＰＫ１３０の過負荷フラグがＯＮであるか否かを、ＣＭＰＫ番号をキーとしてＣＭ稼働率テーブル２９０を参照し、判定する（Ｓ１２６）。

【００７６】

過負荷フラグがＯＮである場合（Ｓ１２６：ＹＥＳ）、マイクロプロセッサ１２１は、

共有メモリ１３２の制御情報（キャッシュディレクトリ）を更新しないことを決定する（Ｓ１２７）。過負荷フラグがＯＦＦである場合（Ｓ１２６：ＮＯ）、マイクロプロセッサ１２１は、共有メモリ１３２の制御情報（キャッシュディレクトリ）を更新することを決定する（Ｓ１２８）。

【００７７】

このように、規定条件を満たす場合、マイクロプロセッサ１２１は、ローカルメモリ１２２でのキャッシュディレクトリ３１０の更新を、共有メモリ１３２のキャッシュディレクトリ５１０に反映しないことを決定する。これにより、マイクロプロセッサ１２１及びＣＭＰＫ１３０の負荷を低減し、システムのスループットを向上することができる。

【００７８】

ローカルメモリの制御情報（本例ではキャッシュディレクトリ）の更新を共有メモリ１３２に反映していないことは、その制御情報の担当ＭＰＰＫ１２０に障害が発生した場合に問題となる。通常動作において、マイクロプロセッサ１２１は、自身のローカルメモリ１２２を参照するため、更新された最新の制御情報を参照することができる。一方、担当ＭＰＰＫ１２０に障害が発生した場合、他のＭＰＰＫ１２０が担当を引き継ぐ（フェイルオーバ）。

【００７９】

障害発生したＭＰＰＫ１２０のローカルメモリ１２２上のデータは消失するため、引き継いだＭＰＰＫ１２０（のマイクロプロセッサ１２１）は、共有メモリ１３２に格納されている古い制御情報しか得ることができない。そのため、共有メモリ１３２に格納されており、更新（共有メモリ１３２へのバックアップ）を省略することができるデータは、ＭＰＰＫ１２０の障害時にユーザデータロストにつながらない制御情報である。

【００８０】

上記好ましい構成は、ＭＰＰＫ１２０で障害が発生した場合に影響が小さい共有メモリ１３２での更新を省略する。具体的には、キャッシュミスによりリードデータが読みだされる記憶ドライブ１７０がＳＳＤである場合（Ｓ１２３：ＹＥＳ）、マイクロプロセッサ１２１は、共有メモリ１３２での更新を行わないことを決定する（Ｓ１２７）。

【００８１】

ＭＰＰＫ１２０の障害により、ＳＳＤから読みだされた上記データがキャッシュされていることを示す情報が失われる。しかし、ＳＳＤは、他のメディア種別のドライブ１７０よりもアクセス性能が高く、失われた制御情報に起因するキャッシュミスの影響は小さく、ＭＰＰＫ１２０及びＣＭＰＫ１３０のオーバヘッド低減によるシステム性能向上効果の方が大きい。

【００８２】

本構成においては、共有メモリ１３２での更新を省略するメディア種別はＳＳＤであるが、この種別は、システム設計に依存する。システムに実装されるメディア（ドライブ）の種別は、ＳＳＤ及びＨＤＤに限らず、これらに加え又はこれらに代えて異なる種別のドライブを実装することができる。実装されている複数のメディア種別において、共有メモリ１３２での更新省略の条件を満たす種別は、設計に従って選択される。最もアクセス性能が高い種別を含む、１又は複数の他の種別よりもアクセス性能が高い種別が選択される。

【００８３】

本構成において、リードコマンド指定データを格納する論理ボリュームのキャッシュヒット率が低い場合（Ｓ１２４：ＹＥＳ）、マイクロプロセッサ１２１は、共有メモリ１３２での更新を行わないことを決定する（Ｓ１２７）。ヒット率が低いボリュームのデータのキャッシュ制御情報が失われても、そのボリュームのアクセス性能への影響は小さく、ＭＰＰＫ１２０及びＣＭＰＫ１３０のオーバヘッド低減によるシステム性能向上効果の方が大きい。

【００８４】

本構成は、さらに、ＭＰＰＫ１２０及びＣＭＰＫ１３０の現状負荷に基づいて、共有メ

モリ１３２での更新の有無を決定する（Ｓ１２５、Ｓ１２６）。ＭＰＰＫ１２０又はＣＭＰＫ１３０の負荷が高い場合、共有メモリ１３２での更新を省略することによる性能向上の効果が大きい。

【００８５】

このように、本構成は、対象ボリュームの性能ブースト機能がＯＮであり、上記４つの条件のいずれかが満たされる場合、共有メモリ１３２でのキャッシュ制御情報の更新を省略する。マイクロプロセッサ１２１は、これらと異なる条件に基づき共有メモリ１３２での更新の有無を決定してもよい。マイクロプロセッサ１２１は、上記４条件のうち複数の条件が満たされることを、共有メモリ１３２での制御情報更新省略の条件としてもよい。

【００８６】

図１７は、図１５のフローチャートにおける、ホストデータ（リードデータ）キャッシシングについての判定（Ｓ１１６）のフローチャートを示している。本ステップのフローチャートは、図１６に示すフローチャートと略同様である。従って、主にこれと異なる点について具体的に説明する。

【００８７】

図１７において、ステップＳ１３２からステップＳ１３６は、それぞれ、図１５のフローチャートにおけるステップ１２２からステップＳ１２６と同様である。ステップ１３７において、マイクロプロセッサ１２１は、記憶ドライブ１７０から読みだしたホストデータ（リードデータ）をキャッシュメモリ１３１に格納することなく、ホスト計算機１８０に送信することを決定する。一方、ステップＳ１３８において、マイクロプロセッサ１２１は、記憶ドライブ１７０から読みだしたホストデータをキャッシュメモリ１３１に格納する（キャッシュする）ことを決定する。

【００８８】

本例において、リードデータをキャッシュするか否かの判定条件は、キャッシュ制御情報の更新を共有メモリ１３２で行うか否かの判定条件と同一である。このように、リードデータキャッシングを制御することで、ＭＰＰＫ１２０及びＣＭＰＫ１３０のオーバヘッド低減によりシステム性能を向上することができる。キャッシュ制御の判定条件と制御情報更新制御の判定条件とは、異なっていてもよい。

【００８９】

次に、ホスト計算機１８０から受信したライトコマンドに対する処理を、図１８Ａ及び図１８Ｂに示すフローチャートを参照して説明する。マイクロプロセッサ１２１は、ホスト計算機１８０からライトコマンドを受け（Ｓ１４１）、その指定アドレスのボリューム（ＬＤＥＶ）に、アクセス権を有するか否かを判定する（Ｓ１４２）。

【００９０】

そのマイクロプロセッサ１２１がアクセス権を有しない場合（Ｓ１４２：ＮＯ）、マイクロプロセッサ１２１は、他の担当ＭＰＰＫ１２０にライトコマンドを転送する（Ｓ１４３）。そのマイクロプロセッサ１２１がアクセス権を有している場合（Ｓ１４２：ＹＥＳ）、マイクロプロセッサ１２１は、同一基板上のローカルメモリ１２２内でキャッシュディレクトリ３１０を検索する（Ｓ１４４）。

【００９１】

ライトコマンドが指定するアドレスが見つかった場合（Ｓ１４５：ＹＥＳ）、マイクロプロセッサ１２１は、キャッシュディレクトリ３１０の情報に従って、キャッシュメモリ１３１にライトデータを書き込み、ホスト計算機１８０にコマンド完了を通知する（Ｓ１４６）。

【００９２】

ライトコマンドが指定するアドレスが見つからなかった（キャッシュミス）場合（Ｓ１４５：ＮＯ）、マイクロプロセッサ１２１は、ローカルメモリ１２２への未キャッシュフラグを確認する（Ｓ１４７）。未キャッシュフラグがＯＮである場合（Ｓ１４７：ＹＥＳ）、マイクロプロセッサ１２１は、ＣＭＰＫ１３０のコントローラを介して、共有メモリ１３２からローカルメモリ１２２へキャッシュディレクトリ（制御情報）を転送する（Ｓ

１４８）。
【００９３】
マイクロプロセッサ１２１は、ローカルメモリ１２２内で、キャッシュディレクトリ３１０を検索する（Ｓ１４９）。ライトコマンドが指定するアドレスが見つかった場合（Ｓ１５０：ＹＥＳ）、マイクロプロセッサ１２１は、キャッシュディレクトリ３１０の情報に従って、キャッシュメモリ１３１にライトデータを書き込み、ホスト計算機１８０にコマンド完了を通知する（Ｓ１５１）。
【００９４】
キャッシュミスの場合（Ｓ１５０：ＮＯ）又は未キャッシュフラグがＯＦＦの場合（Ｓ１４７：ＮＯ）、マイクロプロセッサ１２１は、キャッシュメモリ１３１のセグメントをライトデータのためのスロットとして確保し、さらに、ローカルメモリ１２２のキャッシュディレクトリ３１０を更新する（Ｓ１５２）。
【００９５】
マイクロプロセッサ１２１は、データキャッシングに関する制御情報であるキャッシュディレクトリ３１０の更新を、共有メモリ１３２のデータ５１０に反映するか否かを判定する（Ｓ１５３）。この判定の具体的な方法は、図１５を参照して説明した方法と同様である。マイクロプロセッサ１２１は、さらに、ライトデータ（ホストデータ）をキャッシングするか否かを判定する（Ｓ１５４）。この判定方法は、図１６を参照して説明した方法と同様である。
【００９６】
マイクロプロセッサ１２１がライトデータをキャッシュすると判定した場合（Ｓ１５５：ＹＥＳ）、マイクロプロセッサ１２１は、キャッシュメモリ１３１に新たに確保した領域にライトデータを書き込み、ホスト計算機１８０にコマンド完了を通知する（Ｓ１５６）。マイクロプロセッサ１２１は、ステップＳ１５３での判定結果に関わらず、ローカルメモリ１２２におけるキャッシュディレクトリ３１０の更新に同期して、共有メモリ１３２におけるキャッシュディレクトリ５１０を更新する。
【００９７】
マイクロプロセッサ１２１がライトデータをキャッシュしないと判定した場合（Ｓ１５５：ＮＯ）、マイクロプロセッサ１２１は、ステップＳ１５３における判定結果に基づいて、共有メモリ１３２における制御情報の更新を行う又は省略する。マイクロプロセッサ１２１が、共有メモリ１３２におけるキャッシュ制御情報（キャッシュディレクトリ５１０）を更新すると判定した場合（Ｓ１５７：ＹＥＳ）、マイクロプロセッサ１２１は、ローカルメモリ１２２のキャッシュディレクトリ３１０の更新を、共有メモリ１３２におけるキャッシュディレクトリ５１０に反映し（Ｓ１５８）、次のステップＳ１５９に進む。
【００９８】
マイクロプロセッサ１２１が、共有メモリ１３２におけるキャッシュ制御情報を更新しないと判定した場合（Ｓ１５７：ＮＯ）、マイクロプロセッサ１２１は、書き込み先のＲＡＩＤレベルを、ＲＡＩＤレベルテーブル２４０を参照して特定する（Ｓ１５９）。そのＲＡＩＤレベルが１である場合（Ｓ１５９：ＹＥＳ）、マイクロプロセッサ１２１は、キャッシュメモリ１３１にライトデータを格納することなく、ＢＥＰＫ１４０により記憶ドライブ１７０にデータを書き込み、ホスト計算機１８０にコマンド完了を通知する（Ｓ１６０）。
【００９９】
そのＲＡＩＤレベルが１と異なる場合（Ｓ１５９：ＮＯ）、マイクロプロセッサ１２１は、パリティを生成し、キャッシュメモリ１３１にライトデータを格納することなく、ＢＥＰＫ１４０により記憶ドライブ１７０にパリティ及びライトデータを書き込む。さらに、マイクロプロセッサ１２１はホスト計算機１８０にコマンド完了を通知する（Ｓ１６１）。
【０１００】
このように、本例において、ライトコマンドのハンドリングにおいては、共有メモリ１

３２におけるキャッシュディレクトリ５１０の更新を省略するためには、キャッシュメモリ１３１へのライトデータの格納が省略されることが必要である。キャッシュされたライトデータのデステージ（ドライブ１７０への書き込み）前にそのキャッシュ制御情報が失われると、キャッシュメモリ１３１でそのライトデータを特定することができないからである。

【０１０１】

上述のように、本例において、ステップＳ１５４におけるライトデータをキャッシュするか否かの判定条件は、図１５におけるステップＳ１１６の判定条件と同一である。また、ステップＳ１５３におけるキャッシュ制御情報の更新を共有メモリ１３２で行うか否かの判定条件は、図１５におけるステップＳ１１３の判定条件と同一である。これらは異なっていてもよい。

【０１０２】

このように、ライトデータのキャッシング及びキャッシュ制御情報の更新を制御することによって、ＭＰＰＫ１２０及びＣＭＰＫ１３０のオーバヘッドを低減し、ストレージシステム１０の性能を向上することができる。

【０１０３】

次に、図１９のフローチャートを参照して、管理計算機２０からの設定処理を説明する。管理計算機２０は、その上で実行される管理プログラムに従って動作する。したがって、管理計算機２０を主語とする記載は、管理プログラム又はＣＰＵ２６を主語とすることができる。管理計算機２０は設定処理を開始し（Ｓ１７１）、設定データ入力のためのメニューを、表示デバイス２９に表示する（Ｓ１７２）。管理者は、入力デバイス２８を使用して、必要な設定データを入力する（Ｓ１７３及びＳ１７４：ＮＯ）。

【０１０４】

全ての必要なデータが入力されると（Ｓ１７４：ＹＥＳ）、管理計算機２０は、保存ボタンの選択に応答して設定データを保存する（Ｓ１７５）。設定データは、ストレージシステム１０からの要求に応じて、管理計算機２０からストレージシステム１０に送信される。管理者は、キャンセルボタンを選択することで、入力をやり直すことができる。

【０１０５】

図２０は、メニュー画面の一例２０００を示している。メニュー画面２０００は、性能ブースト機能設定エリア２００１及びボリューム毎性能ブースト機能設定エリア２００４を含む。

【０１０６】

管理者は、性能ブースト機能設定エリア２００１における“ＥＮＡＢＬＥ”又は“ＤＩＳＡＢＬＥ”の一方を入力デバイス２８で選択することで、ストレージシステム１０の性能ブースト機能（上記制御情報の更新制御及びユーザデータのキャッシング制御の機能）をイネーブル又はディセーブルすることができる。この設定が、性能ブースト機能有効化テーブル２１０に反映される。これがディセーブルされると、ストレージシステム１０の全ての性能ブースト機能が使用されない。

【０１０７】

ボリューム毎性能ブースト機能設定エリア２００４は、論理ボリューム番号のカラム２００５及び性能ブースト機能設定カラム２００６を含む。管理者は、ボリューム毎性能ブースト機能設定エリア２００４において、各論理ボリュームの性能ブースト機能のイネーブル／ディセーブルを入力デバイス２８で選択することができる。

【０１０８】

この設定が、ボリューム毎性能ブースト機能有効化テーブル２２０に反映される。システムの性能ブースト機能がイネーブルされており、かつ、ボリュームの性能ブースト機能がイネーブルされているボリュームに対して、本実施形態の性能ブースト機能が使用される。

【０１０９】

図２０は、性能ブースト機能の設定画面を例示しているが、この他、管理計算機２０は

、例えば、判定条件に含まれる閾値の設定画面を表示し、管理者によって入力された設定データをストレージシステム１０に送信する。典型的には、ストレージシステム１０は、管理者により設定可能な項目のデフォルト値を予め有しており、管理者により設定された項目のデータを、入力データにより更新する。

【０１１０】

次に、図２１から図２４を参照して、ストレージシステム１０内のテーブル更新を説明する。図２１は、メディア種別テーブル２３０の更新のフローチャートである。ＲＡＩＤグループが増減設されると（Ｓ２０１）、ＢＥＰＫ１４０が、その情報をいずれかのマイクロプロセッサ１２１に送信する。更新情報を受信したマイクロプロセッサ１２１は、ローカルメモリ１２２のメディア種別テーブル２３０及びＲＡＩＤレベルテーブル２４０を更新すると共に、不揮発性記憶領域のこれらテーブルを更新し（Ｓ２０２）、それを他のＭＰＰK１２０に通知する。管理理計算機２０がＭＰＰＫ１２０に情報を渡してもよい。

【０１１１】

図２２を参照して、ＣＭ稼働率テーブル２９０の更新を説明する。ＭＰＰＫ１２０の任意のマイクロプロセッサ１２１がこの処理を行う。典型的には、定期的（例えば１秒毎）にこの処理が行われる。マイクロプロセッサ１２１は、アクセス先のＣＭＰＫ１３０から稼働率の情報を取得する（Ｓ２１２）。具体的には、マイクロプロセッサ１２１は、ＣＭＰK１３０内のコントローラ（不図示）に、ＣＭＰK１３０の稼働率（ＣＭ稼働率）を示す値を要求し、それをＣＭＰＫ１３０内のコントローラから取得する。

【０１１２】

マイクロプロセッサ１２１は、ＣＭＰK１３０から取得した稼働率の値によって、ＣＭ稼働率テーブル２９０における対応エントリの稼働率カラム２９２のフィールドを更新する（Ｓ２１３）。さらに、マイクロプロセッサ１２１は、更新した稼働率の値が、ＣＭ稼働率閾値テーブル３００の閾値以上であるか判定する（Ｓ２１４）。

【０１１３】

稼働率が閾値以上である場合（Ｓ２１４：ＹＥＳ）、マイクロプロセッサ１２１は、ＣＭ稼働率テーブル２９０における、該当エントリの過負荷フラグを１（ＯＮ）に設定する（Ｓ２１５）。稼働率が閾値未満である場合（Ｓ２１４：ＮＯ）、マイクロプロセッサ１２１は、該当エントリの過負荷フラグを０（ＯＦＦ）に設定する（Ｓ２１６）。マイクロプロセッサ１２１は、アクセスする全てのＣＭＰＫ１３０について、ステップＳ２１２からステップＳ２１６を行う（Ｓ２１７）。

【０１１４】

図２３を参照して、ボリューム毎ヒット率テーブル２５０の更新を説明する。ＭＰＰＫ１２０の任意のマイクロプロセッサ１２１がこの処理を行う。典型的には、定期的（例えば１秒毎）にこの処理が行われる。マイクロプロセッサ１２１は、担当する一つの論理ボリュームのＩ／Ｏ数とヒット数をローカルメモリ１２２から取得する（Ｓ２２２）。当該又は他のマイクロプロセッサ１２１は、前回更新からの各担当論理ボリュームへのＩ／Ｏ数（例えばリードコマンド数）とキャッシュヒット数をカウントし、ローカルメモリ１２２内に格納し、当該マイクロプロセッサ１２１は、その値をステップＳ２２２で取得する。

【０１１５】

マイクロプロセッサ１２１は、取得した値によって、ボリューム毎ヒット率テーブル２５０における、対応エントリのヒット率を更新し、Ｉ／Ｏ数及びヒット数のフィールドを０に設定する（Ｓ２２３）。マイクロプロセッサ１２１は、さらに、ヒット率とヒット率閾値テーブル２６０の閾値とを比較する（Ｓ２２４）。

【０１１６】

ヒット率が閾値以下である場合（Ｓ２２４：ＹＥＳ）、マイクロプロセッサ１２１は、当該エントリの低ヒット率フラグを１（ＯＮ）に設定する（Ｓ２２５）。一方、ヒット率が閾値より大きい場合（Ｓ２２４：ＮＯ）、マイクロプロセッサ１２１は、当該エントリの低ヒット率フラグを０（ＯＦＦ）に設定する（Ｓ２２６）。マイクロプロセッサ１２１

は、担当する全ての論理ボリュームについて、ステップＳ２２２からステップＳ２２６を行う（Ｓ２２７）。

【０１１７】

図２４を参照して、ＭＰ稼働率テーブル２７０の更新を説明する。各マイクロプロセッサ１２１がこの処理を行う。典型的には、定期的（例えば１秒毎）にこの処理が行われる。マイクロプロセッサ１２１は、自身の単位時間（本例で１秒）当たりの稼働時間を監視し、その値をローカルメモリ１２２内に格納する。マイクロプロセッサ１２１は、ローカルメモリ１２２からその値を取得する（Ｓ２３２）。

【０１１８】

マイクロプロセッサ１２１は、取得した値を使用して、該当エントリの稼働率のフィールドを更新し、稼働時間のフィールドに０をセットする（Ｓ２３３）。さらに、マイクロプロセッサ１２１は、更新された稼働率とＭＰ稼働率閾値テーブル２８０の閾値とを比較する（Ｓ２３４）。稼働率が閾値以上である場合（Ｓ２３４：ＹＥＳ）、マイクロプロセッサ１２１は、当該エントリの過負荷フラグを１（ＯＮ）に設定する（Ｓ２３５）。稼働率が閾値未満である場合（Ｓ２３４：ＮＯ）、マイクロプロセッサ１２１は、当該エントリの過負荷フラグを０（ＯＦＦ）に設定する（Ｓ２３６）。

【０１１９】

図２５を参照して、論理ボリュームのオーナ権の現在ＭＰＰＫ１２０から他のＭＰＰＫ１２０への移動を説明する。オーナ権が移動する前に、現在ＭＰＰＫ１２０は、ローカルメモリ１２２に格納するキャッシュディレクトリ３１０における未反映部分を、共有メモリ１３２に反映する。これにより、次のＭＰＰＫ１２０が、最新のキャッシュディレクトリを使用してキャッシュ制御することができ、キャッシュヒット率を高めることができる。

【０１２０】

現在オーナＭＰＰＫのマイクロプロセッサ１２１は、キャッシュディレクトリ３１０において検索する対象を、オーナ権を移動する論理ボリュームの論理アドレスの０番に設定する（Ｓ２４２）。マイクロプロセッサ１２１は、そのアドレスを、キャッシュディレクトリ３１０で検索する（Ｓ２４３）。

【０１２１】

そのアドレスが、共有メモリ未反映フラグがＯＮに設定されているディレクトリに存在する場合（Ｓ２４４：ＹＥＳ）、マイクロプロセッサ１２１は、その共有メモリ１３２における当該ディレクトリを更新し（Ｓ２４５）、ステップＳ２４６に進む。共有メモリ未反映フラグは、対象ディレクトリの更新が共有メモリ１３２に反映済みであるか否かを示すフラグであり、それがＯＮである場合、対象ディレクトリの更新が共有メモリ１３２に未反映であることを示す。

【０１２２】

上記アドレスが、共有メモリ未反映フラグがＯＦＦに設定されているディレクトリに存在する場合（Ｓ２４４：ＮＯ）、マイクロプロセッサ１２１は、共有メモリ１３２上のそのディレクトリを更新することなく、ステップＳ２４６に進む。

【０１２３】

ステップＳ２４６において、マイクロプロセッサ１２１は、当該ボリュームについてのキャッシュディレクトリ３１０の探索が終了したが否かを判定する。全てのアドレスの探索を終了している場合（Ｓ２４６：ＹＥＳ）、マイクロプロセッサ１２１はこの処理を終了する。未探索のアドレスが残っている場合（Ｓ２４６：ＮＯ）、マイクロプロセッサ１２１は対象アドレスを次の論理アドレスに変更し（Ｓ２４７）、ステップＳ２４３からステップＳ２４６を繰り返す。

【０１２４】

第２実施形態

本実施形態は、ストレージ階層仮想化機能を有するストレージシステム１０を説明する。本実施形態のストレージシステム１０は、複数のプールボリューム（実ボリューム）を

含むプールを構築する。プールは、ストレージシステム１０内の性能の異なる複数のメディアを含み、アクセス性能によって複数の階層に階層化される。各階層は、１又は複数のプールボリュームで構成されている。

【０１２５】

ストレージシステム１０は、そのプールから構築した仮想ボリュームをホスト計算機１８０に提供する。ストレージシステム１０は、プールを、特定容量のページ単位で管理する。各プールボリュームは複数ページに分割され、各ページにデータが格納される。ストレージシステム１０は、仮想ボリュームに対するホスト計算機１８０からの書き込みに対して、プールから必要な容量の１又は複数ページを割り当てる。

【０１２６】

ストレージシステム１０は、ホスト計算機１８０により認識される仮想ボリュームの容量を、仮想ボリュームに割り当てられている実容量よりも大きくすることができ、ホスト計算機１８０に割り当てられる容量を実現するために必要な実容量を、それよりも小さくすることができる（シンプロビジョニング）。

【０１２７】

ストレージシステム１０は、仮想ボリュームに対するホスト計算機１８０からのＩ／Ｏ負荷を分析し、Ｉ／Ｏ負荷の高いページを、性能の高い高価なメディアで構成されたリソースから成る上位階層に、そうでないページを性能の低い安価なメディアで構成されたリソースから成る下位階層に自動配置する。これにより、仮想ボリュームへのアクセス性能を維持しつつ、システムのコストを低減することができる。

【０１２８】

以下において、第１実施形態との差異を主に説明する。図２６は、本実施形態のローカルメモリ１２２が格納している情報を示している。ローカルメモリ１２２における制御情報は、第１実施形態で説明した情報に加え、ページ毎モニタ差分テーブル３２０を含む。図２７は、本実施形態の共有メモリ１３２が格納するデータを示している。共有メモリ１３２の制御情報は、第１実施形態で説明した情報に加え、ダイナミックマッピングテーブル５２０及びページ毎モニタテーブル５３０を含む。

【０１２９】

図２８は、ダイナミックマッピングテーブル５２０の一例を示す。ダイナミックマッピングテーブル５２０は、各仮想ボリュームにおいて、アクセス数をカウントするエントリ（記憶領域のエントリ）を管理するテーブルである。例えば、１ページが、ダイナミックマッピングテーブル５２０の１エントリである。ここでは、この例を説明する。

【０１３０】

ダイナミックマッピングテーブル５２０は、プール番号のカラム５２１、仮想ボリューム番号のカラム５２２、論理アドレスのカラム５２３、プールボリューム番号のカラム５２４、論理アドレスのカラム５２５、モニタ情報インデックス番号のカラム５２６を有する。プール番号及び仮想ボリューム番号は、それぞれ、ストレージシステム１０内で、プールと仮想ボリュームを一意に識別する識別子である。モニタ情報インデックス番号は、ダイナミックマッピングテーブル５２０におけるエントリ識別子である。

【０１３１】

論理アドレスのカラム５２３は、各エントリの仮想ボリュームにおける開始論理アドレスを格納する。論理アドレスのカラム５２５は、各エントリのプールボリュームにおける開始論理アドレスを格納する。本例においてエントリの容量は一定であるが、一定でなくともよい。

【０１３２】

図２９は、ページ毎モニタテーブル５３０の一例を示す。ページ毎モニタテーブル５３０は、各ページのＩ／Ｏ数を管理する。マイクロプロセッサ１２１は、このテーブル５３０を参照して、当該ページのデータを格納する階層を決定する。

【０１３３】

ページ毎モニタテーブル５３０は、モニタ情報インデックス番号のカラム５３１、Ｉ／

Ｏカウンタ（現在）のカラム５３２、Ｉ／Ｏカウンタ（前回）のカラム５３３を有する。マイクロプロセッサ１２１は、ページへのアクセスを監視し、所定の監視期間（例えば１秒）内のＩ／Ｏ数（アクセス数）をカウントして、ページ毎モニタテーブル５３０に格納する。監視期間は連続して続く。

【０１３４】

Ｉ／Ｏカウンタ（前回）のカラム５３３は、前回監視期間におけるＩ／Ｏ数を格納する。Ｉ／Ｏカウンタ（現在）のカラム５３２は、現在監視期間におけるＩ／Ｏ数を格納する。マイクロプロセッサ１２１は、現在監視期間内において、Ｉ／Ｏカウンタ（現在）のカラム５３２の値を繰り返し更新する。

【０１３５】

本構成において、マイクロプロセッサ１２１は、ローカルメモリ１２２におけるページ毎モニタ差分テーブル３２０を使用してＩ／Ｏ数をカウントし、その値の更新を共有メモリ１３２におけるページ毎モニタテーブル５３０に反映する。この点は後述する。現在監視期間が終了すると、マイクロプロセッサ１２１は、現在監視期間におけるＩ／Ｏ数を、前回監視期間におけるＩ／Ｏ数のフィールドに移す。

【０１３６】

図３０は、ページ毎モニタ差分テーブル３２０の一例を示す。ページ毎モニタ差分テーブル３２０は、各ページへのアクセスをカウントするために使用される。ページ毎モニタ差分テーブル３２０は、モニタ情報インデックス番号のカラム３２１及びＩ／Ｏ差分カウンタのカラム３２２を有する。マイクロプロセッサ１２１は、各ページのアクセスを監視し、アクセスがあると、Ｉ／Ｏ差分カウンタのカラム３２２の該当フィールドの値をインクリメントする。

【０１３７】

Ｉ／Ｏ差分カウンタのカラム３２２のフィールドの値が規定値（本例で最大値）に達すると、マイクロプロセッサ１２１は、ページ毎モニタテーブル５３０の対応エントリのＩ／Ｏカウンタ（現在）のカラム５３２のフィールドの値にその値を加算して、当該フィールドを更新する。マイクロプロセッサ１２１は、最大値に達したＩ／Ｏ差分カウンタのカラム３２２のフィールドの値を初期値（０値）に戻す。Ｉ／Ｏ差分カウンタは、このようにページ毎モニタテーブル５３０の前回更新からの、Ｉ／Ｏ数の差分を示す。

【０１３８】

図３０及び図２９に示すように、ページ毎モニタ差分テーブル３２０のＩ／Ｏ差分カウンタのカラム３２２は８ビットデータを格納し、ページ毎モニタテーブル５３０のＩ／Ｏカウンタ（現在）のカラム５３２は、８ビットよりも大きい３２ビットのデータを格納する。データのビット数は設計に依存し、８又は３２ビットに限定されない。

【０１３９】

図３１のフローチャートを参照して、上記ストレージ階層仮想化機能モニタ更新の具体的な方法を説明する。マイクロプロセッサ１２１は、ページへのアクセスを受けると、ページ毎モニタ差分テーブル３２０におけるそのページのＩ／Ｏ差分カウンタをインクリメントする（Ｓ３０２）。ページ対するアクセスは、ページにマッピングされている記憶デバイスへのＩ／Ｏ又はページに対するホストＩ／Ｏを示し、マイクロプロセッサ１２１は、いずれかのＩ／Ｏ数をカウントする。

【０１４０】

マイクロプロセッサ１２１は、当該論理ボリューム性能ブースト機能がＯＮであるか判定する（Ｓ３０３）。このステップは、図１６におけるステップＳ１２２と同様である。ボリューム性能ブースト機能がＯＦＦである場合（Ｓ３０３：ＮＯ）、マイクロプロセッサ１２１は、ステップＳ３０７に進む。

【０１４１】

ボリューム性能ブースト機能がＯＮである場合（Ｓ３０３：ＹＥＳ）、マイクロプロセッサ１２１は、自身の過負荷フラグがＯＮであるか否かを判定する（Ｓ３０４）。このステップは、図１６におけるステップＳ１２５と同様である。

【０１４２】
過負荷フラグがＯＮである場合（Ｓ３０４：ＹＥＳ）、マイクロプロセッサ１２１は、ステップＳ３０６に進む。過負荷フラグがＯＦＦである場合（Ｓ３０４：ＮＯ）、マイクロプロセッサ１２１は、アクセス先のＣＭＰＫ１３０の過負荷フラグがＯＮであるか否かを判定する（Ｓ３０５）。このステップは、図１６におけるステップＳ１２６と同様である。
【０１４３】
ＣＭＰＫ１３０の過負荷フラグがＯＦＦである場合（Ｓ３０５：ＮＯ）、マイクロプロセッサ１２１は、ステップＳ３０７に進む。ＣＭＰＫ１３０の過負荷フラグがＯＮである場合（Ｓ３０５：ＹＥＳ）、マイクロプロセッサ１２１は、ステップＳ３０６に進む。ステップＳ３０６において、マイクロプロセッサ１２１は、ページ毎モニタ差分テーブル３２０の上記Ｉ／Ｏ差分カウンタの値が、最大値であるかを判定する。
【０１４４】
Ｉ／Ｏ差分カウンタの値が最大値未満である場合（Ｓ３０６：ＮＯ）、このフローは終了する。Ｉ／Ｏ差分カウンタの値が最大値である場合（Ｓ３０６：ＹＥＳ）、マイクロプロセッサ１２１は、ページ毎モニタテーブル５３０の対応エントリのＩ／Ｏカウンタ（現在）のカラム５３２のフィールドの値にその最大値を加算して、当該フィールドを更新する（Ｓ３０７）。マイクロプロセッサ１２１は、さらに、最大値に達したＩ／Ｏ差分カウンタのカラム３２２のフィールドの値を０値（初期値）に設定する（Ｓ３０８）。
【０１４５】
本例は、マイクロプロセッサ１２１及びＣＭＰＫ１３０の負荷が小さい場合、ローカルメモリ１２２におけるＩ／Ｏ差分カウンタの更新に同期して、共有メモリ１３２のＩ／Ｏカウンタを更新する。これらの負荷が小さいためシステム性能の低下が問題とならず、障害発生時に正確なＩ／Ｏカウント数を得ることができる。これら二つのデバイスの負荷条件は省略してもよく、双方の成立をＩ／Ｏカウンタ値の非同期更新の条件としてもよい。これらと異なる条件を使用してもよい。
【０１４６】
上述のように、マイクロプロセッサ１２１は、ローカルメモリ１２２内のカウンタでページのＩ／Ｏ数をカウントし、その値が規定値に達すると、その規定値を共有メモリ１３２のカウンタに反映する。これにより、マイクロプロセッサ１２１とＣＭＰＫ１３０との間の通信によるオーバヘッドを低減する。
【０１４７】
ページ毎モニタ差分テーブル３２０のカウンタのビット数が、ページ毎モニタテーブル５３０のカウンタのビット数より小さい。このように、ローカルメモリ上で差分をカウントすることで、Ｉ／Ｏ数カウントのためにローカルメモリ１２２で必要とされる容量を削減することができる。ＭＰＰＫ１２０の障害時には、所定期間のＩ／Ｏカウント数の情報が失われるが、ページＩ／Ｏカウント数にＩ／Ｏカウント数の差分の反映がなされないだけであるので、ページのＩ／Ｏ解析に実質的な影響を与えることはない。
【０１４８】
なお、本実施形態の性能モニタ方法は、階層仮想化機能のモニタに限らず、そのほかの性能モニタにも適用可能である。例えば、ＨＤＤやＳＳＤなどのドライブのモニタに適用できる。上記例は、カウンタ数が最大値に達するとカウンタを初期化するが、初期化においてＩ／Ｏをカウントしてもよい。マイクロプロセッサ１２１は、例えば、Ｉ／Ｏ差分カウンタの初期化と共に、その最大数に１を加えた値をページ毎モニタテーブル５３０のＩ／Ｏカウンタの値に加算する。これは、他の実施形態におけるカウント方法で同様である。
【０１４９】
第３実施形態
以下において、本発明を非同期リモートコピーに適用した例を説明する。以下においては、第１実施形態及び第２実施形態との差異を主に説明する。図３２は、本実施形態の計

算機システムの構成を模式的に示すブロック図である。本実施形態のストレージシステムは、第１ストレージシステム１０Ａ及び第２ストレージシステム１０Ｂを含む。典型的には、第１ストレージシステム１０Ａ及び第２ストレージシステム１０Ｂは異なるサイトに設置されており、データネットワーク（例えばＳＡＮ）１９０Ａ、データネットワーク（例えばＳＡＮ）１９０Ｂ及び広域ネットワークを介して通信可能に接続する。

【０１５０】

第１ストレージシステム１０Ａ及び第２ストレージシステム１０Ｂは、図１を参照して説明したハードウェア構成と同様の構成を有する。具体的には、第１ストレージシステム１０Ａは、複数のＦＥＰＫ１１０Ａ、複数のＭＰＰＫ１２０Ａ、複数のＣＭＰＫ１３０Ａ、複数のＢＥＰＫ１４０Ａを含み、これらは内部ネットワーク１５０Ａを介して接続する。第１管理計算機２０Ａは、第１ストレージシステム１０Ａを管理する。

【０１５１】

同様に、第２ストレージシステム１０Ｂは、複数のＦＥＰＫ１１０Ｂ、複数のＭＰＰＫ１２０Ｂ、複数のＣＭＰＫ１３０Ｂ、複数のＢＥＰＫ１４０Ｂを含み、これらは内部ネットワーク１５０Ｂを介して接続する。第２管理計算機２０Ｂは、第２ストレージシステム１０Ｂを管理する。

【０１５２】

第１ストレージシステム１０Ａ及び第２ストレージシステム１０Ｂは、非同期リモードコピー機能を有する。第１ストレージシステム１０Ａのプライマリボリューム（ＰＶＯＬ）１７１Ｐと、第２ストレージシステム１０Ｂのセカンダリボリューム（ＳＶＯＬ）１７１Ｓが、コピーペアを構成する。ボリュームは、典型的には、１又は複数のＲＡＩＤグループにおける１又は複数の記憶領域からなる。

【０１５３】

プライマリボリューム１７１Ｐがコピー元ボリューム、セカンダリボリューム１７１Ｓがコピー先ボリュームであり、プライマリボリューム１７１Ｐのデータが、セカンダリボリューム１７１Ｓにコピーされる。プライマリボリューム１７１Ｐへデータ書き込み順序と、セカンダリボリューム１７１Ｓへのデータコピー順序は一致する（順序保障）。

【０１５４】

同期コピーは、ホスト計算機１８０がプライマリボリューム１７１Ｐに書き込みをおこなった場合、セカンダリボリューム１７１Ｓへのコピーの完了後（典型的にはキャッシュメモリへの書き込み後）に、ホスト計算機１８０にＩ／Ｏ成功を通知する。これに対して、非同期コピーは、プライマリボリューム１７１Ｐへの書き込み完了後、セカンダリボリューム１７１Ｓへのコピー完了前に、ホスト計算機１８０にＩ／Ｏ成功を通知する。

【０１５５】

本実施形態のストレージシステムは、プライマリボリューム１７１Ｐからセカンダリボリューム１７１Ｓへのコピー用のバッファーとして、ジャーナルボリューム（ＪＶＯＬ）１７１ＪＰ、１７１ＪＳを使用する。第１ストレージシステム１０Ａにおいて、プライマリボリューム１７１Ｐとジャーナルボリューム１７１ＪＰとがグループ化されている。第２ストレージシステム１０Ｂにおいて、セカンダリボリューム１７１Ｓとジャーナルボリューム１７１ＪＳとがグループ化されている。

【０１５６】

プライマリボリューム１７１Ｐにおける更新データは、ジャーナルボリューム１７１ＪＰ、１７１ＪＳを介して、セカンダリボリューム１７１Ｓに送信される。これにより、リモートコピーのデータ転送において、性能が不安定な広域ネットワークを使用することができる。

【０１５７】

図３３を参照して、ホスト計算機１８０からのプライマリボリューム１７１Ｐへのデータ書き込み及びその更新データのセカンダリボリューム１７１Ｓへのコピーの流れを説明する。ＦＥＰＫ１１０Ａは、ホスト計算機１８０からのライトコマンド及びライトデータを受信する。ＭＰＰＫ１２０Ａ（のマイクロプロセッサ１２１）は、ライトコマンドを解

析し、ＦＥＰＫ１１０Ａ及びＢＥＰＫ１４０Ａ（不図示）に、プライマリボリューム１７１Ｐ及びジャーナルボリューム１７１ＪＰにライトデータを書き込むことを指示する。

【０１５８】

具体的には、ＭＰＰＫ１２０Ａは、ＦＥＰＫ１１０Ａ及びＢＥＰＫ１４０Ａにライトデータを指定した次の転送先に転送することを指示する。最終的な転送先はプライマリボリューム１７１Ｐ及びジャーナルボリューム１７１ＪＰであり、ライトデータは、プライマリボリューム１７１Ｐ及びジャーナルボリューム１７１ＪＰのそれぞれに書き込まれる。ジャーナルボリューム１７１ＪＰへの書き込み順序をもって、プライマリボリューム１７１Ｐへの書き込み順序とする。

【０１５９】

本図において、ライトデータのキャッシュメモリ１３１への書き込みの説明は省略されている、又はライトデータはキャッシュメモリ１３１を介することなくボリュームに格納される。ＭＰＰＫ１２０Ａは、ライトデータのキャッシュメモリ１３１への書き込み完了又はボリュームへの書き込み完了に応答して、ホスト計算機１８０に書き込み完了を通知する。

【０１６０】

ＭＰＰＫ１２０Ａは、ジャーナルボリューム１７１ＪＰの更新に従って、ジャーナルボリューム１７１ＪＰの管理データを更新する。図３３に示すように、ジャーナルボリューム１７１ＪＰは、管理領域６１１とデータ領域６１２を有し、それぞれが、ジャーナルボリューム管理データ及び更新データを格納する。ジャーナルボリューム管理データはジャーナルボリューム外に格納されていてもよい。

【０１６１】

ジャーナルボリューム管理データは、シーケンス番号６０１及びポインタ６０２のペアを含む。これらの値のペアが、各ライトデータ（更新データ）に付与される。本図の例において、シーケンス番号６０１は、１からｎの値のいずれかの値であり、データ領域に格納された順に、各ライトデータに昇順で付与される。シーケンス番号は循環的であり、ｎが付与されたライトデータの次のデータには１が付与される。ポインタ６０２は、データ領域６１２において対応するシーケンス番号が付与されているライトデータが格納されている位置（アドレス）を示す。

【０１６２】

管理領域６１１は、シーケンス番号６０１とポインタ６０２のペアが書き込まれている領域と、未使用領域６０４を含む。未使用領域６０４は初期値を格納しており、本例において初期値は０値である。マイクロプロセッサ１２１は、データ領域６１２に格納されている更新データを第２ストレージシステム１０Ｂに転送すると、そのデータのシーケンス番号６０１とポインタ６０２を格納している領域の値を初期値（無効値）に更新する。更新データの転送順序は、シーケンス番号６０１の順に一致し、シーケンス番号６０１の順をもって、更新データのジャーナルボリューム１７１ＪＰへの書き込み順とする。

【０１６３】

管理領域６１１において、シーケンス番号６０１とポインタ６０２の次の新たなペアを書き込む位置は決まっており、例えば、ペアは、管理領域６１１におけるアドレス昇順で書き込まれる。終点アドレスに書き込まれているペアの次のペアは開始アドレスに書き込まれる。

【０１６４】

シーケンス番号６０１とポインタ６０２とを格納する領域（ジャーナル領域とも呼ぶ）において、初期値を格納している領域の直前位置のシーケンス番号６０１、つまりジャーナル領域の先頭のシーケンス番号が最も新しい更新データを示す。一方、初期値を格納している領域の直後位置のシーケンス番号６０１、つまりジャーナル領域の最後尾のシーケンス番号が最も古い更新データを示す。

【０１６５】

第２ストレージステム１０ＢのＭＰＰＫ１２０Ｂが第１ストレージシステム１０Ａへ更

新データのコピーの要求を送ったとき、第１ストレージシステム１０ＡのＭＰＰＫ１２０Ａは、ジャーナルボリューム１７１ＪＰに格納されている更新データおよびシーケンス番号を、更新順（書き込み順）で、第２ストレージシステム１０Ｂに転送する。第２ストレージシステム１０ＢのＭＰＰＫ１２０Ｂは、そのＦＥＰＫ１１０Ｂが受信した更新データを、順次、ジャーナルボリューム１７１ＪＳに格納する。本図においてキャッシュメモリ１３１へのキャッシングが省略されている。ＭＰＰＫ１２０Ｂは、規定のタイミングで、ジャーナルボリューム１７１ＪＳに格納されている更新データを、更新順序でセカンダリボリューム１７１Ｓに書き込む。

【０１６６】

第２ストレージシステム１０Ｂのジャーナルボリューム１７１ＪＳは、ジャーナルボリューム１７１ＪＰと同様に、管理領域とデータ領域とを含み、それぞれが、ジャーナル管理データと更新データを格納する。

【０１６７】

ＭＰＰＫ１２０Ｂは、転送された更新データをジャーナルボリューム１７１ＪＳに格納してから、転送されたシーケンス番号及びポインタを書き込み、管理データを更新する。管理データの構成はジャーナルボリューム１７１ＪＰと同様である。ジャーナルボリューム１７１ＪＳ内の更新データがセカンダリボリューム１７１Ｓに書き込まれると、ＭＰＰＫ１２０Ｂは、対応するシーケンス番号とポインタの値を初期値（無効値）に変更する。

【０１６８】

図３４は、第１ストレージシステム１０Ａ及び第２ストレージシステム１０Ｂにおけるローカルメモリ１２２が格納している制御情報を示している。本実施形態において、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０が、ローカルメモリ１２２内に格納されている。図３５は、第１ストレージシステム１０Ａ及び第２ストレージシステム１０Ｂにおける共有メモリ１３２が格納している制御情報を示している。本実施形態において、非同期リモートコピー管理テーブル５４０及びＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０が格納されている。

【０１６９】

非同期リモートコピー管理テーブル５４０は、ペア管理のための管理情報を格納している。具体的には、プライマリボリュームとセカンダリボリュームの各ペアを管理する管理情報、リモートコピーのパスの情報、そして、プライマリボリューム及びセカンダリボリュームのそれぞれとグループ化されるジャーナルボリュームの情報を含む。マイクロプロセッサ１２１は、この管理テーブル５４０を参照して、リモートコピーの実行を制御する。

【０１７０】

図３６は、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０の一例を示す。ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０は、ローカルメモリ１２２において、各ジャーナルボリュームの最新シーケンス番号を管理する。ＭＰＰＫ１２０Ａのマイクロプロセッサ１２１は、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０を参照して、新たにジャーナルボリューム１７１ＪＳに書き込まれる更新データのシーケンス番号を決定することができる。

【０１７１】

ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０は、ＪＶＯＬ番号のカラム３３１、シーケンス番号のカラム３３２、そしてシーケンス番号差分のカラム３３３を有する。ＪＶＯＬ番号は、第１ストレージシステム１０Ａにおけるジャーナルボリュームの識別子である。シーケンス番号のカラム３３２は、ＪＶＯＬにおける先頭シーケンス番号を示すデータを格納する。シーケンス番号差分については後述する。

【０１７２】

図３７は、ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０の一例を示す。ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０は、共有メモリ１３２において、各ジャーナルボリュームのシーケンス番号を管理する。ＳＭ非同期リモートコピ

ーシーケンス番号管理テーブル５３０は、ＪＶＯＬ番号のカラム５３１及びシーケンス番号のカラム５３２を有する。

【０１７３】

シーケンス番号のカラム５３２は、ＪＶＯＬにおける先頭シーケンス番号を示すデータを格納する。１エントリにおけるシーケンス番号のカラム５３２の値は、ローカルメモリ１２２において対応するシーケンス番号のカラム３３２の値と一致する又は異なる（図３６及び図３７の例では全てのエントリの値が異なる）。それらの更新は、同期又は非同期である。

【０１７４】

図３６及び図３７に示すように、各ＪＶＯＬのエントリにおいて、シーケンス番号差分カラム３３３のフィールドの値は、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０のシーケンス番号カラム３３２の対応フィールドの値と、ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０のシーケンス番号カラム５３２の対応フィールドの値との差分である。

【０１７５】

このように、シーケンス番号差分カラム３３３のフィールドの値は、シーケンス番号カラム５３２における対応フィールドの前回更新からのＪＶＯＬにおけるシーケンス番号の更新を示し、共有メモリ１３２に格納されている前回更新時の先頭シーケンス番号と最新の先頭シーケンス番号との差分を示す。

【０１７６】

ＭＰＰＫ１２０Ａのマイクロプロセッサ１２１は、ジャーナルボリュームに更新データの書き込みがあるたびに、そのジャーナルボリュームのエントリにおいて、シーケンス番号カラム３３２及びシーケンス番号差分カラム３３３の値をインクリメントする。シーケンス番号カラム３３２の各フィールドは、対応するジャーナルボリュームの最新のシーケンス番号（最後に割り当てたシーケンス番号）を示している。シーケンス番号カラム３３２の各フィールドの値は、最大値からインクリメントされると最小値に戻る。

【０１７７】

シーケンス番号差分カラム３３３のビット数（最大値）は、シーケンス番号カラム３３２のビット数（最大値）よりも小さい。マイクロプロセッサ１２１は、シーケンス番号差分カラム３３３のフィールドの値が最大値に達すると、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０における当該エントリの更新を、ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０の対応エントリに反映する。

【０１７８】

具体的には、ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０における対応エントリのシーケンス番号を、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０の対応エントリのシーケンス番号に一致させる。ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０における更新値は、更新前の値にシーケンス番号差分カラム３３３における対応フィールドの値を加算した値である。

【０１７９】

このように、ローカルメモリ１２２においてシーケンス番号の最大数よりも小さい所定数までシーケンス番号の変化をカウントし、ローカルメモリ１２２におけるシーケンス番号の変化を共有メモリ１３２のシーケンス番号に反映することで、マイクロプロセッサ１２１によるＣＭＰＫ１３０へのアクセス回数を低減し、それらの間の通信によるマイクロプロセッサ１２１及びＣＭＰＫ１３０の負荷を低減することができる。

【０１８０】

図３８のフローチャートを参照して、本実施形態の非同期リモートコピーシーケンス番号の更新を説明する。この処理は、ジャーナルボリューム１７１ＪＰの担当ＭＰＰＫ１２０Ａのマイクロプロセッサ１２１が実行する。

【０１８１】

マイクロプロセッサ１２１は、ジャーナルボリューム１７１ＪＰへの更新データ書き込

みがあると、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０を参照して、当該ジャーナルボリューム１７１ＪＰの管理領域６１１に、新たなシーケンス番号及びポインタを追加する。さらに、マイクロプロセッサ１２１は、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０において、当該ジャーナルボリューム１７１ＪＰのエントリのシーケンス番号及びシーケンス番号差分の値を更新する（本例においてそれら値をインクリメントする）（Ｓ４１２）。

【０１８２】

マイクロプロセッサ１２１は、当該ボリュームの性能ブースト機能がＯＮであるか判定する（Ｓ４１３）。性能ブースト機能がＯＦＦである場合（Ｓ４１３：ＮＯ）、マイクロプロセッサ１２１は、ステップＳ４１７に進む。性能ブースト機能がＯＮである場合（Ｓ４１３：ＹＥＳ）、マイクロプロセッサ１２１は、自身の過負荷フラグがＯＮであるか判定する（Ｓ４１４）。

【０１８３】

過負荷フラグがＯＮである場合（Ｓ４１４：ＹＥＳ）、マイクロプロセッサ１２１は、ステップＳ４１６に進む。過負荷フラグがＯＦＦである場合（Ｓ４１４：ＮＯ）、マイクロプロセッサ１２１は、アクセス先のＣＭＰＫ１３０Ａの過負荷フラグがＯＮであるか判定する（Ｓ４１５）。

【０１８４】

ＣＭＰＫ１３０Ａの過負荷フラグがＯＦＦである場合（Ｓ４１５：ＮＯ）、マイクロプロセッサ１２１は、ステップＳ４１７に進む。ＣＭＰＫ１３０Ａの過負荷フラグがＯＮである場合（Ｓ４１５：ＹＥＳ）、マイクロプロセッサ１２１は、ステップＳ４１６に進む。ステップＳ４１３からステップＳ４１５の詳細は、第２実施形態で既に説明した通りである。マイクロプロセッサ１２１及び／又はＣＭＰＫ１３０１３０Ａの負荷に応じて制御情報の更新反映を制御することで、システム性能の低下を抑えつつ、共有メモリの更新をより適切に行うことができる。

【０１８５】

ステップＳ４１６において、マイクロプロセッサ１２１は、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０において、当該ジャーナルボリューム１７１ＪＰのシーケンス番号差分が、最大値であるか判定する。その値が最大値ではない場合（Ｓ４１６：ＮＯ）、マイクロプロセッサ１２１は、本処理を終了する。

【０１８６】

上記値が最大値である場合（Ｓ４１６：ＹＥＳ）、マイクロプロセッサ１２１は、ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０において、当該ジャーナルボリューム１７１ＪＰのシーケンス番号を更新する（Ｓ４１７）。具体的には、マイクロプロセッサ１２１は、現在のシーケンス番号の値にシーケンス番号差分の値を加算した値に更新する。ステップＳ４１８において、マイクロプロセッサ１２１は、最大値に達しているシーケンス番号差分のフィールドの値を０値に更新（初期化）する。

【０１８７】

上記シーケンス番号差分を使用した共有メモリ１３２におけるシーケンス番号の更新（性能ブースト機能）を使用しない場合、ＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０及びＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０の更新は同期する。

【０１８８】

ＭＰＰＫ１２０Ａに障害が発生した場合、ローカルメモリ１２２上のＬＭ非同期リモートコピーシーケンス番号管理テーブル３３０が失われる。上述のように、このテーブル３３０は、各ジャーナルボリュームの最新の先頭シーケンス番号を示す情報を有している。正常なリモートコピーを行うためには、ジャーナル管理データにおける最新の先頭シーケンス番号が必要である。

【０１８９】

本実施形態の第１ストレージシステム１０Ａは、障害発生したＭＰＰＫ１２０Ａと異な

るＭＰＰＫ１２０Ａが、ジャーナルボリューム１７１ＪＰの管理領域６１１を参照して、ジャーナル領域の先頭を示す最新の先頭シーケンス番号を確認する。図３９のフローチャートを参照して、ＭＰＰＫ障害発生時の非同期リモートコピーシーケンス番号回復処理を説明する。

【０１９０】

担当を引き継いだ正常なＭＰＰＫ１２０Ａのマイクロプロセッサ１２１は、共有メモリ１２３に格納されているＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０から、一つのジャーナルボリュームを選択し、そのシーケンス番号を読み出す（Ｓ４２２）。マイクロプロセッサ１２１は、そのジャーナルボリュームから、上記シーケンス番号の領域の次のシーケンス番号領域からデータを読み出す（Ｓ４２３）。

【０１９１】

マイクロプロセッサ１２１は、ステップＳ４２３で読み出したシーケンス番号が０値（無効値）であるか判定する（Ｓ４２４）。そのシーケンス番号が０値ではない場合（Ｓ４２４：ＮＯ）、マイクロプロセッサ１２１は、その読みだしたシーケンス番号をテンポラル領域（典型的にそのローカルメモリ１２２内の領域）に格納する（Ｓ４２５）。

【０１９２】

そのシーケンス番号が０値である場合（Ｓ４２４：ＹＥＳ）、その領域は未使用領域であり、マイクロプロセッサ１２１は、テンポラル領域に格納されているシーケンス番号で、ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０における対応ジャーナルボリュームのシーケンス番号を更新する。ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０のシーケンス番号が最新の先頭シーケンス番号である場合、更新は不要である。マイクロプロセッサ１２１は、障害発生したＭＰＰＫ１２０Ａが担当であった全てのジャーナルボリュームについて、上記更新を行う。

【０１９３】

上記フローにより、ＳＭ非同期リモートコピーシーケンス番号管理テーブル５３０が最新情報を含むように更新され、他のＭＰＰＫ１２０Ａが、障害が起きたＭＰＰＫ１２０Ａの担当を引き継ぎ、正常な非同期リモートコピーを続けることができる。

【０１９４】

上記シーケンス番号管理テーブル３３０、５３０が格納する値は一例であって、それらは、先頭シーケンス番号又はそれらテーブル３３０、５３０の先頭シーケンス番号間の差分を示すことができれば、どのような値を格納していてもよい。

【０１９５】

第４実施形態

以下において、本発明を非同期ローカルコピーに適用した例を説明する。以下においては、上記他の実施形態と異なる点を主に説明する。図４０は、本実施形態のローカルメモリ１２２に格納されている制御情報を示している。ローカルメモリ１２２には、ＬＭローカルコピー差分管理テーブル３４０及びＬＭローカルコピー差分領域間引き動作管理テーブル３５０が格納されている。

【０１９６】

図４１は、本実施形態の共有メモリ１３２内の制御情報を示している。ＳＭローカルコピー差分管理テーブル５６０、ＳＭローカルコピー差分領域間引き動作管理テーブル５７０、ローカルコピー管理テーブル５８０が、共有メモリ１３２における制御情報に含まれている。複数のＭＰＰＫ１２０が、共有メモリ１３２内テーブル５６０、５７０、５８０を参照可能である。特に、ＳＭローカルコピー差分管理テーブル５６０及びＳＭローカルコピー差分領域間引き動作管理テーブル５７０は、ＭＰＰＫ１２０の障害時に、他のＭＰＰＫ１２０により参照される。

【０１９７】

ローカルコピー管理テーブル５８０は、プライマリボリュームとセカンダリボリュームの各ペアを管理する管理情報を含む。例えば、ペアを構成するプライマリボリュームとセカンダリボリュームの識別情報、それらのアドレス情報及びコピーポリシの情報を含む。

マイクロプロセッサ１２１は、ローカルコピー管理テーブル５８０を参照して、ローカルコピーの実行を制御する。

【０１９８】

共有メモリ１３２内のＳＭローカルコピー差分管理テーブル５６０及びＳＭローカルコピー差分領域間引き動作管理テーブル５７０は、それぞれ、ローカルメモリ１２２内のＬＭローカルコピー差分管理テーブル３４０及びＬＭローカルコピー差分領域間引き動作管理テーブル３５０のバックアップである。マイクロプロセッサ１２１は、予め定められた規則に従って、ローカルメモリ１２２でのテーブル３４０、３５０の更新を、共有メモリ１３２のテーブル５６０、５７０に反映する。

【０１９９】

図４２は、ＬＭローカルコピー差分管理テーブル３４０の一例を示す。ＬＭローカルコピー差分管理テーブル３４０は、ボリューム番号のカラム３４１、論理アドレスのカラム３４２、差分有ビット列のカラム３４３を有する。ボリューム番号は、ストレージシステム内でのプライマリボリュームの識別子である。各エントリは、ボリューム内の所定広さの記憶領域（アドレス範囲）を示している。論理アドレスは、各エントリの記憶領域の開始論理アドレスを示す。本例において、エントリの記憶領域の広さは共通である。

【０２００】

差分有ビット列は、そのエントリの記憶領域において、プライマリボリュームとセカンダリボリュームとの間にデータの相違が存在するか否か、つまり、プライマリボリュームでの更新がセカンダリボリュームに反映されているか否かを示す。

【０２０１】

差分有ビット列の各ビット（差分有ビットとも呼ぶ）は、エントリの記憶領域における各部分領域のデータがプライマリボリュームとセカンダリボリュームとの間で異なるか否かを示す。本例では、各ビットに対応する領域の広さは共通である。本例において、差分有ビット列のビットが１である場合、その領域のデータは、プライマリボリュームとセカンダリボリュームとで異なることを示す。

【０２０２】

マイクロプロセッサ１２１は、所定のタイミングで、プライマリボリュームの更新データをセカンダリボリュームにコピーする（非同期ローカルコピー）。非同期ローカルコピーにおいて、マイクロプロセッサ１２１は、ＬＭローカルコピー差分管理テーブル３４０を参照し、プライマリボリュームにおける差分有ビットが１である領域のデータを、セカンダリボリュームにコピーする。

【０２０３】

この非同期ローカルコピーに応答して、マイクロプロセッサ１２１は、ＬＭローカルコピー差分管理テーブル３４０において、更新がセカンダリボリュームに反映された領域の差分有ビットを０値に更新する。

【０２０４】

図４３は、ＳＭローカルコピー差分管理テーブル５６０の一例を示す。ＳＭローカルコピー差分管理テーブル５６０は、ＬＭローカルコピー差分管理テーブル３４０のバックアップテーブルであり、ＬＭローカルコピー差分管理テーブル３４０と同一の構成を有する。具体的には、ボリューム番号のカラム５６１、論理アドレスのカラム５６２、差分有ビット列のカラム５６３を有する。

【０２０５】

マイクロプロセッサ１２１は、所定規則に従って、ＬＭローカルコピー差分管理テーブル３４０における更新を、ＳＭローカルコピー差分管理テーブル５６０にコピーする。本例において、プライマリボリュームからセカンダリボリュームへの非同期ローカルコピーによるＬＭローカルコピー差分管理テーブル３４０の更新とＳＭローカルコピー差分管理テーブル５６０更新は同期する。プライマリボリュームのへのデータライトによる更新に対するＳＭローカルコピー差分管理テーブル５６０の更新については後述する。

【０２０６】

図４４は、ＬＭローカルコピー差分領域間引き動作管理テーブル３５０の一例を示す。ＬＭローカルコピー差分領域間引き動作管理テーブル３５０は、ボリューム番号のカラム３５１、論理アドレスのカラム３５２、間引き中ビット列のカラム３５３を有する。各エントリは、ボリューム内の所定広さの記憶領域（アドレス範囲）を示している。

【０２０７】

論理アドレスは、各エントリの記憶領域の開始論理アドレスを示す。本例において、エントリの記憶領域の広さは共通である。ＬＭローカルコピー差分領域間引き動作管理テーブル３５０におけるエントリの記憶領域は、ＬＭローカルコピー差分管理テーブル３４０のエントリの記憶領域よりも広い。

【０２０８】

間引き中ビット列は、ＬＭローカルコピー差分管理テーブル３４０における差分有ビット列の更新を、ＳＭローカルコピー差分管理テーブル５６０の対応する差分有ビット列に反映するか否かを示す。上述のように、ＬＭローカルコピー差分領域間引き動作管理テーブル３５０において、間引き中ビット列は、論理ボリューム内の記憶領域に関連づけられている。

【０２０９】

間引き中ビット列の各ビット（間引き中ビットとも呼ぶ）は、その間引き中ビット列に関連付けられている記憶領域の部分領域に関連付けられている。間引き中ビット列の各ビットは、それが関連づけられている部分領域を介して、１又は複数の差分有ビットに関連づけられる。

【０２１０】

好ましい本例において、間引き中ビットは複数の差分有ビットに関連づけられている。また、ＬＭローカルコピー差分領域間引き動作管理テーブル３５０におけるエントリの記憶領域（アドレス範囲）は、ＬＭローカルコピー差分管理テーブル３４０におけるエントリの記憶領域（アドレス範囲）よりも広い。間引き中ビット列のビット数は、差分有ビット列のビット数と同一又は異なる（図４３、図４４の例において同一）。

【０２１１】

上述のように、ＬＭローカルコピー差分管理テーブル３４０において、各差分有ビットは、記憶領域に関連づけられている。間引き中ビットに関連づけられている記憶領域の少なくとも一部が差分有ビットの記憶領域と一致する場合、その間引き中ビットはその差分有ビットに関連づけられている。

【０２１２】

間引き中ビットが１である場合、ローカルメモリ１２２においてそれに関連づけられている差分有ビットの、プライマリボリュームの更新（データ書き込み）に応答した更新は、共有メモリ１３２における差分有ビットに反映されない。具体的には、プライマリボリュームへのライトコマンドの受信に応答して、マイクロプロセッサ１２１は、ＬＭローカルコピー差分領域間引き動作管理テーブル３５０において、ライトコマンドが指示する領域の間引き中ビットを参照する。

【０２１３】

間引き中ビットが１である場合、マイクロプロセッサ１２１は、ＬＭローカルコピー差分管理テーブル３４０において対応する差分有ビットの更新を、ＳＭローカルコピー差分管理テーブル５６０に反映しない。これにより、ＭＰＰＫ１２０とＣＭＰＫ１３０との間の通信によるＭＰＰＫ１２０とＣＭＰＫ１３０の負荷を低減する。

【０２１４】

図４５は、ＳＭローカルコピー差分領域間引き動作管理テーブル５７０の一例を示す。ＳＭローカルコピー差分領域間引き動作管理テーブル５７０は、ＬＭローカルコピー差分領域間引き動作管理テーブル３５０のバックアップテーブルであり、それと同じ構成を有する。具体的には、ボリューム番号のカラム５７１、論理アドレスのカラム５７２、間引き中ビット列のカラム５７３を有する。マイクロプロセッサ１２１は、ＬＭローカルコピー差分領域間引き動作管理テーブル３５０の更新に同期して、ＳＭローカルコピー差分領

域間引き動作管理テーブル５７０を更新する。

【０２１５】

図４６のフローチャートを参照して、非同期ローカルコピー差分管理情報の更新を説明する。プライマリボリュームにデータが書き込まれると、マイクロプロセッサ１２１は、ＬＭローカルコピー差分管理テーブル３４０を更新する（Ｓ５０２）。具体的には、プライマリボリュームにおいて更新された領域に関連づけられている差分有ビットを更新する。

【０２１６】

マイクロプロセッサ１２１は、当該ボリュームの性能ブースト機能がＯＮであるか判定する（Ｓ５０３）。性能ブースト機能がＯＦＦである場合（Ｓ５０３：ＮＯ）、マイクロプロセッサ１２１は、ステップＳ５０９に進み、ＳＭローカルコピー差分管理テーブル５６０を更新する（同期更新）。性能ブースト機能がＯＮである場合（Ｓ５０３：ＹＥＳ）、マイクロプロセッサ１２１は、自身の過負荷フラグがＯＮであるか判定する（Ｓ５０４）。

【０２１７】

過負荷フラグがＯＮである場合（Ｓ５０４：ＹＥＳ）、マイクロプロセッサ１２１は、ステップＳ５０６に進む。過負荷フラグがＯＦＦである場合（Ｓ５０４：ＮＯ）、マイクロプロセッサ１２１は、アクセス先のＣＭＰＫ１３０の過負荷フラグがＯＮであるか判定する（Ｓ５０５）。

【０２１８】

ＣＭＰＫ１３０の過負荷フラグがＯＦＦである場合（Ｓ５０５：ＮＯ）、マイクロプロセッサ１２１は、ステップＳ５０９に進み、ＳＭローカルコピー差分管理テーブル５６０を更新する。ＣＭＰＫ１３０の過負荷フラグがＯＮである場合（Ｓ５０５：ＹＥＳ）、マイクロプロセッサ１２１は、ステップＳ５０６に進む。ステップＳ５０３からステップＳ５０５の詳細は、第２実施形態で既に説明した通りであり、システム性能の低下を抑えつつ、共有メモリ１３２の制御情報を適切に更新する。

【０２１９】

ステップＳ５０６において、マイクロプロセッサ１２１は、プライマリボリュームにおいて更新された領域が間引き中であるか判定する。具体的には、マイクロプロセッサ１２１は、ＬＭローカルコピー差分領域間引き動作管理テーブル３５０を参照し、上記更新領域の各間引き中ビットを確認する。間引き中ビットが１である場合（Ｓ５０６：ＹＥＳ）、マイクロプロセッサ１２１は、ＳＭローカルコピー差分管理テーブル５６０において、その間引き中ビットに対応する差分有ビットの更新を省略する。

【０２２０】

間引き中ビットが０である場合（Ｓ５０６：ＮＯ）、マイクロプロセッサ１２１は、その間引き中ビットが関連づけられている領域の差分が閾値以上であるか判定する（Ｓ５０７）。具体的には、マイクロプロセッサ１２１は、ＬＭローカルコピー差分管理テーブル３４０を参照し、当該間引き中ビットに対応する差分有ビットにおいて、１のビット数が閾値以上であるか判定する。この判定基準については、図４７を参照して後述するＭＰＰＫ障害時の処理において説明する。

【０２２１】

差分が閾値未満である場合（Ｓ５０７：ＮＯ）、マイクロプロセッサ１２１は、ＳＭローカルコピー差分管理テーブル５６０を更新する（Ｓ５０９）。差分が閾値以上である場合（Ｓ５０７：ＹＥＳ）、マイクロプロセッサ１２１は、ＬＭローカルコピー差分領域間引き動作管理テーブル３５０及びＳＭローカルコピー差分領域間引き動作管理テーブル５６０を更新する（Ｓ５０８）。具体的には、マイクロプロセッサ１２１は、上記２つのテーブル３５０、５６０において、上記間引き中ビットを０から１に変更する。

【０２２２】

次に、図４７のフローチャートを参照して、ＭＰＰＫ１２０障害時における、ローカルコピー差分のコピーを説明する。ＭＰＰＫ１２０で障害発生した場合、他のＭＰＰＫ１２

０が、障害発生したＭＰＰＫ１２０が担当していたコピーペアにおいて、プライマリボリュームからセカンダリボリュームへ、それらの差分をコピーする。これにより、コピーペアの同一性を確保し、その後の正常な非同期ローカルコピーを実現する。

【０２２３】

上記他のＭＰＰＫ１２０におけるマイクロプロセッサ１２１は、ＳＭローカルコピー差分領域間引き動作管理テーブル５７０を参照し（Ｓ５１２）、間引き中領域が残っているか否かを判定する（Ｓ５１３）。間引き中領域は、その間引き中ビットが１である領域である。間引き領域が残っていなければ（Ｓ５１３：ＮＯ）、このフローは終了する。間引き中領域が残っている場合（Ｓ５１３：ＹＥＳ）、マイクロプロセッサ１２１は、プライマリボリュームにおけるその領域のデータを、セカンダリボリュームにコピーする（Ｓ５１４）。

【０２２４】

上述のように、共有メモリ１３２は、「１」の間引き中ビットに対応する最新の差分有ビット列を格納していない。そのため、ＭＰＰＫ１２０での障害発生時には、間引き中ビットが１（ＯＮ）である領域の全てのデータを、プライマリボリュームからセカンダリボリュームにコピーする。これにより、セカンダリボリュームのデータをプライマリボリュームのデータに正確に一致させることができる。

【０２２５】

図４６のフローチャートを参照して説明したように、本例は、間引き中ビットに対応する差分有ビットの内の「１」のビットが閾値以上である場合に、間引き中ビットをＯＮ（１）に設定する。障害時には、対応する間引き中ビットがＯＮである全てのデータをプライマリボリュームからセカンダリボリュームにコピーするため、要更新データが多い領域の更新を間引くことで、更新による負荷を低減すると共に障害時の処理を効率化することができる。

【０２２６】

本実施形態において、差分管理テーブル及び間引き動作管理テーブルの構成は一例であり、差分領域及び間引き中領域を示すことができれば、どのようなデータによりそれらを示してもよい。ＳＭローカルコピー差分領域間引き動作管理テーブル５７０を使用する代わりに、間引き領域のＬＭ／ＳＭの差分有りビットを全て１にする処理を加えることでも同様の障害時の回復を実現できる。

【０２２７】

図４８は、第２実施形態から第４実施形態で使用可能な、性能ブースト機能設定のためのメニュー画面の例４８００を示している。メニュー画面４８００は、性能ブースト機能設定エリア４８０１、ボリューム毎性能ブースト機能設定エリア４８０２及び機能毎性能ブースト機能設定エリア４８０３含む。

【０２２８】

管理者は、性能ブースト機能設定エリア４８０１における“ＥＮＡＢＬＥ”又は“ＤＩＳＡＢＬＥ”の一方を入力デバイス２８で選択することで、ストレージシステム１０の性能ブースト機能をイネーブル又はディセーブルすることができる。この設定が、性能ブースト機能有効化テーブル２１０に反映される。

【０２２９】

ボリューム毎性能ブースト機能設定エリア４８０２は、各論理ボリュームの性能ブースト機能のイネーブル／ディセーブルを可能とする。管理者は、ボリューム毎性能ブースト機能設定エリア４８０２において、各論理ボリュームの性能ブースト機能のイネーブル／ディセーブルを入力デバイス２８で選択することができる。この設定が、ボリューム毎性能ブースト機能有効化テーブル２２０に反映される。

【０２３０】

機能毎性能ブースト機能設定エリア４８０３は、各性能ブースト機能のイネーブル／ディセーブルを可能とする。管理者は、機能毎性能ブースト機能設定エリア４８０３において、各機能のイネーブル／ディセーブルを入力デバイス２８で選択することができる。こ

の設定が、ストレージシステム１０内の機能毎性能ブースト機能有効化テーブル（不図示）に反映される。システム、ボリュームそして機能のブースト機能の全てがイネーブルされている場合に、その性能ブースト機能がそのボリュームにおいて使用される。

【０２３１】

第５実施形態

本実施形態において、スイッチにより結合した複数のストレージモジュールを含むストレージシステムに本発明を適用した例を説明する。本実施形態は、主に上記他の実施形態と異なる点を説明する。図４９は、本実施形態の計算機システムの構成を模式的に示す。ストレージモジュール１０Ｃ及びストレージモジュール１０Ｄは、モジュール間パス（スイッチ）１９５（Ｘパスとも呼ぶ）により通信可能に接続されている。

【０２３２】

図４９におけるストレージモジュール１０Ｃ、１０Ｄの構成は、図１を参照して説明したストレージシステム１０の構成と同様である。本例においては、２つの結合したモジュールが一つのストレージシステムを構成するが、３以上のモジュールが一つのストレージシステムを構成してもよい。

【０２３３】

ストレージモジュール１０Ｃ及びストレージモジュール１０Ｄを結合するＸパス（スイッチ）１９５は、内部ネットワーク１５０のパスと同様のパスとして機能し、一方のモジュールの任意のパッケージは、他方のモジュールの任意のパッケージ及びメディアと、Ｘパス１９５により通信することができる。また、ホスト計算機１８０は、いずれのストレージモジュールにもアクセスすることができる。

【０２３４】

Ｘパスは、内部ネットワーク１５０よりも帯域が狭く、データ転送能力が低い。そのため、Ｘパスは、パッケージ間のデータ転送においてボトルネックとなりやすい。そのため、Ｘパスの負荷に基づいて性能ブースト機能のＯＮ／ＯＦＦを判定することで、ストレージシステムの性能の低下を小さくすることができる。

【０２３５】

本実施形態のマイクロプロセッサ１２１は、性能ブースト機能のイネーブル／ディセーブル制御において、Ｘパス１９５の稼働率を参照する。これにより、複数のモジュールからなるストレージシステムにおいて適切にシステム性能を向上することができる。

【０２３６】

図５０は、本実施形態のローカルメモリ１２２が格納している制御情報を示している。図５０において、Ｘパス稼働率テーブル３６０及びＸパス稼働率閾値テーブル３７０がローカルメモリ１２２内に格納されている。図５１は、Ｘパス稼働率テーブル３６０の一例を示す。図５２は、Ｘパス稼働率閾値テーブル３７０の一例を示す。

【０２３７】

Ｘパス稼働率テーブル３６０は、Ｘパスの稼働率を管理する。本例において、Ｘパス稼働率テーブル３６０は、Ｘパス番号のカラム３６１、稼働率のカラム３６１、そして過負荷判定フラグのカラム３６３を有する。Ｘパス番号は、システム内でＸパスを一意に識別する識別子である。図５１の例において、Ｘパス稼働率テーブル３６０は、複数のＸパスを管理している。つまり、複数のＸパスが２以上のストレージモジュールを結合している。複数のＸパスは、同一又は異なるスイッチを通過する。

【０２３８】

稼働率は、単位時間当たりのデータ転送時間である。Ｘパスの稼働率は、そのＸパスのコントローラが計算し、レジスタに格納する。マイクロプロセッサ１２１は、各Ｘパスの稼働率を、スイッチのレジスタから取得して、Ｘパス稼働率テーブル３６０に格納する。

【０２３９】

マイクロプロセッサ１２１は、Ｘパス稼働率テーブル３６０の各エントリ稼働率と、予め定められているＸパス稼働率閾値とを比較して、過負荷判定フラグの値を決定する。Ｘパス稼働率が閾値以上である場合、マイクロプロセッサ１２１は過負荷判定フラグを１に

設定する。Ｘパス稼働率閾値は、Ｘパス稼働率閾値テーブル３７０のＸパス稼働率閾値カラムに格納されている。例えば、Ｘパス稼働率閾値テーブル３７０は、ストレージシステム内の不揮発性記憶領域からロードされ、その値は管理者により設定される。

【０２４０】

図５３のフローチャートを参照して、Ｘパスの稼働率を考慮したデータキャッシングに関する制御情報の共有メモリ１３２における更新についての判定を説明する。基本的な部分は、第１実施形態と同様である。図５３のフローチャートにおいて、ステップＳ６０７以外のステップは、第１実施形態における図１６に示すフローチャートと同様であり、その説明を省略する。

【０２４１】

ステップＳ６０７において、マイクロプロセッサ１２１は、Ｘパス稼働率テーブル３６０を参照し、共有メモリ１３２へのアクセスに使用するＸパスの過負荷フラグが１（ＯＮ）であるは判定する。アクセスするＣＭＰＫ１３０と使用するＸパスとの関係を示す制御情報は、ローカルメモリ１２２内に格納されており、それにより、マイクロプロセッサ１２１は、使用するＸパスを特定することができる。

【０２４２】

過負荷フラグがＯＮである場合（Ｓ６０７：ＹＥＳ）、マイクロプロセッサ１２１は、共有メモリ１３２の制御情報を更新しないことを決定する（Ｓ６０９）。過負荷フラグがＯＦＦ（０）である場合（Ｓ６０７：ＮＯ）、マイクロプロセッサ１２１は、共有メモリ１３２の制御情報を更新することを決定する（Ｓ６０８）。本例はデータキャッシング制御情報の更新判定においてＸパスの稼働率を参照するが、他の実施形態で説明した他の判定処理も、Ｘパスの稼働率を参照することができる。

【０２４３】

次に、図５４のフローチャートを参照して、Ｘパス稼働率テーブル３６０におけるＸパス稼働率の更新を説明する。典型的には、この処理は、定期的に、例えば１秒毎に実行される。マイクロプロセッサ１２１は、一つのＸパス、一例としてＸパス１９５を選択し、そのＸパス１９５の稼働率を取得する（Ｓ６１２）。

【０２４４】

マイクロプロセッサ１２１は、取得した稼働率の値により、Ｘパス稼働率テーブル３６０の該当エントリの稼働率の値を更新する（Ｓ６１３）。マイクロプロセッサ１２１は、取得した稼働率の値が、Ｘパス稼働率閾値テーブル３７０におけるＸパス稼働率閾値以上であるか判定する（Ｓ６１４）。稼働率が閾値以上である場合（Ｓ６１４：ＹＥＳ）、マイクロプロセッサ１２１は、Ｘパス稼働率テーブル３６０における当該エントリの過負荷フラグを１（ＯＮ）に設定する（Ｓ６１５）。

【０２４５】

一方、稼働率が閾値未満である場合（Ｓ６１４：ＮＯ）、マイクロプロセッサ１２１は、Ｘパス稼働率テーブル３６０における当該エントリの過負荷フラグを０（ＯＦＦ）に設定する（Ｓ６１６）。マイクロプロセッサ１２１は、全てのＸパスの稼働率を更新したか判定し（Ｓ６１７）、全てのＸパスについて判定している場合（Ｓ６１７：ＹＥＳ）にこのフローを終了し、未判定のＸパスが残っている場合（Ｓ６１７：ＮＯ）には、残りのＸパスから一つのＸパスを選択して、このフローを繰り返す。

【０２４６】

第６実施形態

本実施形態は、ＭＰＰＫ１２０が、複数の異なる種別のデバイスに分散している複数の共有メモリ領域にアクセス可能な構成を説明する。本実施形態において、上記他の実施形態と異なる点について主に説明する。

【０２４７】

図５５は、本実施形態の計算機システムの構成を模式的に示している。ストレージシステム１０において、複数の異なるデバイスに共有メモリ（記憶領域）が存在している。具体的には、ＣＭＰＫ１３０上の共有メモリ１３２の他、ＭＰＰＫ１２０上に共有メモリ１

２４、そして記憶ドライブ１７０に共有メモリ１７８が存在している。ＭＰＰＫ１２０上の共有メモリ１２４の領域は、ローカルメモリ１２２内の記憶領域である。記憶ドライブ１７０上の共有メモリ１７８の領域は、記憶ドライブにおける不揮発性記憶媒体の記憶領域である。

【０２４８】

図５６は、本実施形態のローカルメモリ１２２が格納している制御情報を示している。図５６において、ＭＰ稼働率テーブル３８０、ＭＰ稼働率閾値テーブル３９０、ＳＭ領域管理テーブル４００がローカルメモリ１２２内に格納されている。

【０２４９】

図５７は、ＭＰ稼働率テーブル３８０の一例を示す。ＭＰ稼働率テーブル３８０は、ＭＰ番号のカラム３８１、稼働率のカラム３８２、過負荷判定フラグ１のカラム３８３、過負荷判定フラグ２のカラム３８４、稼働時間のカラム３８５を有する。過負荷判定フラグ２のカラム３８４以外のカラムは、図１１に示すＭＰ稼働率テーブル２７０と同様である。過負荷判定フラグ１のカラム３８３は、過負荷判定フラグのカラム２７３に相当する。

【０２５０】

図５８は、ＭＰ稼働率閾値テーブル３９０の一例を示す。ＭＰ稼働率閾値テーブル３９０は、ＭＰ稼働率閾値１のカラム３９１及びＭＰ稼働率閾値２のカラム３９２を有する。ＭＰ稼働率閾値１の値は、ＭＰ稼働率閾値２の値より高い。ＭＰ稼働率閾値１は、図１２に示すＭＰ稼働率閾値に相当する。

【０２５１】

図５９は、ＳＭ領域管理テーブル４００の一例を示す。ＳＭ領域管理テーブル４００は、複数のデバイスに分散している共有メモリ領域を管理する。ＳＭ領域管理テーブル４００は、種別のカラム４０１、番号のカラム４０２、先頭アドレスのカラム４０３、空き容量のカラム４０４を有する。「種別」は、共有メモリ領域が存在するデバイスの種別を示す。「番号」は、同一種別のデバイスにおける識別子である。「先頭アドレス」は、各デバイスにおける共有メモリ領域の先頭アドレスを示す。「空き容量」は、共有メモリ領域の空き容量である。

【０２５２】

種別のカラム４０１、番号のカラム４０２、先頭アドレスのカラム４０３には、予め値が設定されている。マイクロプロセッサ１２１は、各デバイスのコントローラ（ＭＰＰＫにおいてはマイクロプロセッサ１２１）から、共有メモリ領域の空き容量の値を取得し、それを空き容量のカラム４０４に格納する。

【０２５３】

図６０Ａ及び６０Ｂを参照して、データキャッシングに関する共有メモリ領域に格納された制御情報の更新についての判定を説明する。図６０ＡのフローチャートにおけるステップＳ７０２からステップＳ７０７は、図１６のフローチャートにおけるステップＳ１２２からステップＳ１２７までと同様である。ただし、ステップＳ７０６において、当該ＣＭＰＫ１３０の過負荷フラグがＯＮである場合（Ｓ７０６：ＹＥＳ）、マイクロプロセッサは図６０Ｂにおけるステップ７０９に進む。

【０２５４】

ステップＳ７０６において当該ＣＭＰＫ１３０の過負荷フラグがＯＦＦである場合（Ｓ７０６：ＮＯ）又はステップＳ７０２において当該論理ボリュームの性能ブースト機能がＯＦＦである場合（Ｓ７０２：ＮＯ）、マイクロプロセッサ１２１は、当該ＣＭＰＫ１３０の共有メモリの制御情報を更新すると決定する。

【０２５５】

図６０Ｂにおけるステップ７０９において、マイクロプロセッサ１２１は、ＳＭ領域管理テーブル４００を参照し、必要な空き共有メモリ領域を有するＭＰＰＫ１２０が存在するか判定する。いずれかのＭＰＰＫ１２０が必要な空き共有メモリ領域を有する場合（Ｓ７０９：ＹＥＳ）、マイクロプロセッサ１２１は、そのＭＰＰＫ１２０の番号を特定し、キャッシング制御情報を当該ＭＰＰＫ１２０の共有メモリ１２４に格納し、その更新を

行うことを決定する（Ｓ７１０）。このＭＰＰＫ１２０は、マイクロプロセッサ１２１が実装されたＭＰＰＫ１２０と異なるＭＰＰＫである。

【０２５６】

必要な空き共有メモリ領域を有するＭＰＰＫ１２０が存在しない場合（Ｓ７０９：ＮＯ）、マイクロプロセッサ１２１は、自身の過負荷フラグ２が１（ＯＮ）であるか判定する（Ｓ７１１）。過負荷フラグ２がＯＮである場合（Ｓ７１１：ＹＥＳ）、マイクロプロセッサ１２１は、共有メモリ領域における制御情報の更新を行わないことを決定する（Ｓ７１６）。

【０２５７】

過負荷フラグ２がＯＦＦである場合（Ｓ７１１：ＮＯ）、マイクロプロセッサ１２１は、ＳＭ領域管理テーブル４００を参照し、必要な空き共有メモリ領域を有するＳＳＤ　ＲＡＩＤグループが存在するか判定する（Ｓ７１２）。

【０２５８】

いずれかのＳＳＤ　ＲＡＩＤグループが必要な空き共有メモリ領域を有する場合（Ｓ７１２：ＹＥＳ）、マイクロプロセッサ１２１は、当該ＳＳＤ　ＲＡＩＤグループの番号を特定し、キャッシュ制御情報を当該ＳＳＤ　ＲＡＩＤグループの共有メモリ領域に格納し、その更新を行うことを決定する（Ｓ７１３）。

【０２５９】

必要な空き共有メモリ領域を有するＳＳＤ　ＲＡＩＤグループが存在しない場合（Ｓ７１２：ＮＯ）、マイクロプロセッサ１２１は、ＳＭ領域管理テーブル４００を参照し、必要な空き共有メモリ領域を有するＨＤＤ　ＲＡＩＤグループが存在するか判定する（Ｓ７１４）。必要な空き共有メモリ領域を有するＨＤＤ　ＲＡＩＤグループが存在しない場合（Ｓ７１４：ＮＯ）、マイクロプロセッサ１２１は、共有メモリ１３２における制御情報を更新しないことを決定する（Ｓ７１６）。

【０２６０】

必要な空き共有メモリ領域を有するＨＤＤ　ＲＡＩＤグループが存在する場合（Ｓ７１４：ＹＥＳ）、マイクロプロセッサ１２１は、当該ＨＤＤ　ＲＡＩＤグループの番号を特定し、キャッシュ制御情報を当該ＨＤＤ　ＲＡＩＤグループの共有メモリ領域に格納し、その更新を行うことを決定する（Ｓ７１５）。

【０２６１】

マイクロプロセッサ１２１は、共有メモリ１３２以外のいずれかの共有メモリに制御情報を格納し、その制御情報を更新することを決定すると、ローカルメモリ１２２におけるデータキャッシング制御情報を、選択した共有メモリにコピーする。

【０２６２】

このように、制御情報を現在の共有メモリ１３２の領域から他の共有メモリ領域に移動することで、共有メモリにおける制御情報の更新を、ローカルメモリにおける更新に同期させることができ、障害発生時のキャッシュヒット率を向上することができる。上記フローは、アクセス性能が高いデバイスから、空き共有メモリ領域の有無を判定する。これにより、よりアクセス性能が高い共有メモリに制御情報を格納することができ、システム性能の低下を抑えることができる。

【０２６３】

本実施形態の共有メモリ領域管理は、データキャッシング制御情報の格納及び更新管理の他、上記他の実施形態で説明した他の制御情報の格納及び更新管理に適用することができる。ＭＰＰＫ障害時には、他のＭＰＰＫ１２０は、共有メモリ領域管理テーブル４００を参照し、分散している共有メモリ領域において対応する制御情報を検索することができる。

【０２６４】

図６１のフローチャートを参照して、ＭＰ稼働率の更新を説明する。このフローは、１秒などの周期で呼び出される。マイクロプロセッサ１２１は、自身のＭＰ稼働時間を取得し（Ｓ７２２）、ＭＰ稼働率テーブル３８０の稼働率の値を更新する（Ｓ７２３）。ステ

ップＳ７２２、Ｓ７２３は、図２４におけるステップＳ２３２、Ｓ２３３と同様である。
【０２６５】
次に、ステップＳ７２４において、マイクロプロセッサ１２１は、更新した稼働率の値が、ＭＰ稼働率閾値１の値以上であるか判定する。稼働率の値がＭＰ稼働率閾値１以上である場合（Ｓ７２４：ＹＥＳ）、マイクロプロセッサ１２１は、ＭＰ稼働率テーブル３８０の過負荷フラグ１を１（ＯＮ）に設定する（Ｓ７２５）。稼働率の値がＭＰ稼働率閾値１未満である場合（Ｓ７２４：ＮＯ）、マイクロプロセッサ１２１は、ＭＰ稼働率テーブル３８０の過負荷フラグ１を０（ＯＦＦ）に設定する（Ｓ７２６）。
【０２６６】
次に、ステップＳ７２７において、マイクロプロセッサ１２１は、更新した稼働率の値が、ＭＰ稼働率閾値２以上であるか判定する。稼働率の値がＭＰ稼働率閾値２以上である場合（Ｓ７２７：ＹＥＳ）、マイクロプロセッサ１２１は、ＭＰ稼働率テーブル３８０の過負荷フラグ２を１（ＯＮ）に設定する（Ｓ７２８）。稼働率の値がＭＰ稼働率閾値２未満である場合（Ｓ７２７：ＮＯ）、マイクロプロセッサ１２１は、ＭＰ稼働率テーブル３８０の過負荷フラグ２を０（ＯＦＦ）に設定する（Ｓ７２９）。
【０２６７】
第７実施形態
本実施形態のストレージシステムは、ホストデータのキャッシングによるアクセス性能の向上に基づき、低ヒット率フラグのＯＮ／ＯＦＦを決定する。低ヒット率フラグは第１実施形態で説明した通りである。アクセス性能は、例えば、レスポンスタイムやスループットで表される。以下に説明する構成は、レスポンスタイムを使用する。
【０２６８】
データキャッシングの使用によるレスポンスタイムの向上が大きい場合に低ヒット率フラグ（第１実施形態参照）はＯＦＦに設定され、データキャッシングの使用によるレスポンスタイムの向上が小さい場合に低ヒット率フラグはＯＮに設定される。これにより、平均レスポンスタイムを向上することができる。
【０２６９】
以下において、本実施形態を具体的に説明する。主に上記他の実施形態と異なる点を説明する。図６２は、本実施形態のローカルメモリ１２２に格納されている制御情報を示している。レスポンステーブル４１０及びＣＭ利用閾値テーブル４２０がローカルメモリ１２２に格納されている。図６３はレスポンステーブル４１０の一例を示し、図６４はＣＭ利用閾値テーブル４２０の一例を示す。
【０２７０】
レスポンステーブル４１０は、メディアのレスポンスタイムを管理するテーブルである。図６３において、レスポンステーブル４１０は、メディア種別のカラム４１１及びレスポンスタイムのカラム４１２を有する。本例のレスポンステーブル４１０は、メディア種別によりレスポンスタイムを管理するが、ＲＡＩＤグループや論理ボリュームによりレスポンスタイムを管理してもよい。
【０２７１】
本例において、レスポンスタイムは、メディアからデータを読み出すために要する時間である。レスポンスタイムのカラム４１２には、予め値が格納されている、又は、マイクロプロセッサ１２１は、レスポンスタイムのカラム４１２の値を更新してもよい。マイクロプロセッサ１２１は、データ読み出しにおけるレスポンスタイムを測定し、例えば測定値の平均値をレスポンスタイムのカラム４１２に格納する。
【０２７２】
図６４において、ＣＭ利用閾値テーブル４２０は、レスポンス向上のカラム４２１において、レスポンス向上を示す値の閾値を格納している。閾値は予め設定されている。例えば、管理者により設定された値が、ストレージシステム内の不揮発性記憶領域に格納されている。
【０２７３】

後述するように、マイクロプロセッサ１２１は、メディアのレスポンスタイムとＣＭＰＫ１３０（キャッシュメモリ１３１）のレスポンスタイムとの差を使用して、レスポンス向上を表す値を算出する。この値が上記閾値より大きい場合、レスポンス向上がデータキャッシングに見合うレベルにあることを示す。

【０２７４】

図６５のフローチャートを参照して、本実施形態のレスポンス向上に基づく低ヒット率フラグ更新を含むヒット率更新処理を説明する。ＭＰＰＫ１２０は、定期的に、例えば、１秒毎にこの処理を実行する。図６５のフローチャートにおけるステップＳ８０２、Ｓ８０３、Ｓ８０５～Ｓ８０７は、それぞれ、図２３のフローチャートにおけるステップＳ２２２、Ｓ２２３、Ｓ２２５～Ｓ２２７と同様である。

【０２７５】

ステップＳ８０４において、マイクロプロセッサ１２１は、下記の式１に従って、レスポンス向上を表す値を算出する。

ヒット率×（当該メディアのレスポンスタイム－ＣＭＰＫレスポンスタイム）／１００

【０２７６】

マイクロプロセッサ１２１は、当該ボリュームのＲＡＩＤグループから、メディア種別テーブル２３０を参照して、当該メディアの種別を特定することができる。レスポンスタイムの値は、上述のように、レスポンステーブル４１０に格納されている。マイクロプロセッサ１２１は、算出した値とＣＭ利用閾値テーブル４２０のＣＭ利用閾値とを比較する。

【０２７７】

算出した値がＣＭ利用閾値以下である場合（Ｓ８０４：ＹＥＳ）、マイクロプロセッサ１２１は、当該ボリュームの低ヒット率フラグを１（ＯＮ）に設定する（Ｓ８０５）。算出した値がＣＭ利用閾値より大きい場合（Ｓ８０４：ＮＯ）、マイクロプロセッサ１２１は、当該ボリュームの低ヒット率フラグを０（ＯＦＦ）に設定する（Ｓ８０６）。

【０２７８】

以上、本発明の実施形態を説明したが、本発明が上記の実施形態に限定されるものではない。当業者であれば、上記の実施形態の各要素を、本発明の範囲において容易に変更、追加、変換することが可能である。ある実施形態の構成の一部を他の実施形態の構成に置き換えることが可能であり、ある実施形態の構成に他の実施形態の構成を加えることも可能である。各実施形態の構成の一部について、他の構成の追加・削除・置換をすることが可能である。

【０２７９】

上記の各構成、機能、処理部、処理手段等は、それらの一部又は全部を、例えば集積回路で設計されたハードウェアで実現してもよい。各機能を実現するプログラム、テーブル、ファイル等の情報は、不揮発性半導体メモリ、ハードディスクドライブ、ＳＳＤ等の記憶デバイス、または、ＩＣカード、ＳＤカード、ＤＶＤ等の計算機読み取り可能な非一時的データ記憶媒体に格納することができる。

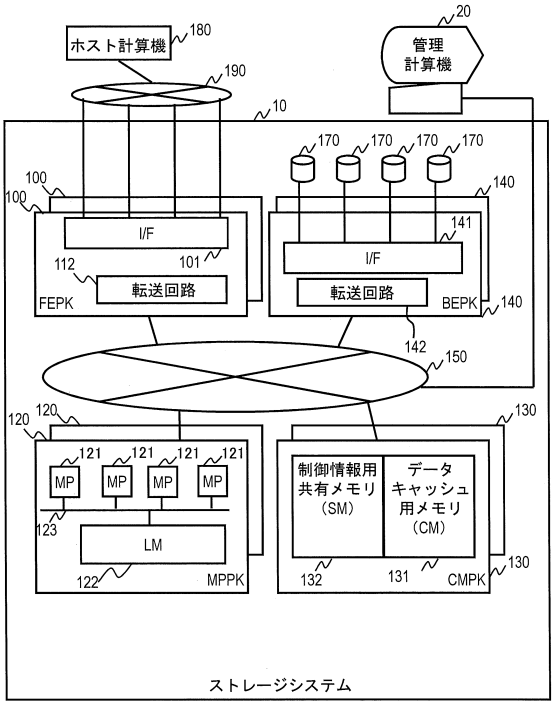
【０２８０】

上記実施形態において、制御情報は複数のテーブルにより表されているが、本発明が使用する制御情報は、データ構造に依存しない。制御情報は、テーブルの他、例えば、データベース、リスト、キュー等のデータ構造で表現することができる。上記実施形態において、識別子、名、ＩＤ等の表現は、互いに置換が可能である。

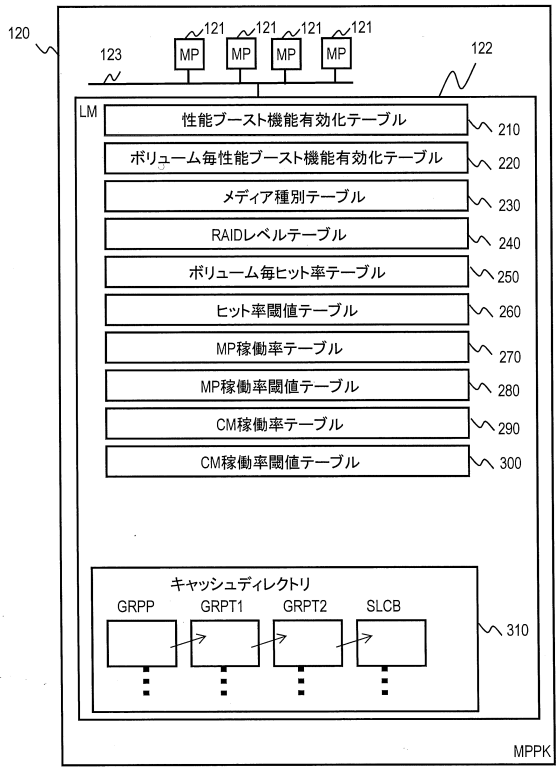
【０２８１】

プロセッサである、ＣＰＵ、マイクロプロセッサ又は複数のマイクロプロセッサのグループは、プログラムに従って動作することで、定められた処理を実行する。従って、本実施形態においてプロセッサを主語とする説明は、プログラムを主語とした説明でもよく、プロセッサが実行する処理は、そのプロセッサが実装された装置及びシステムが行う処理である。

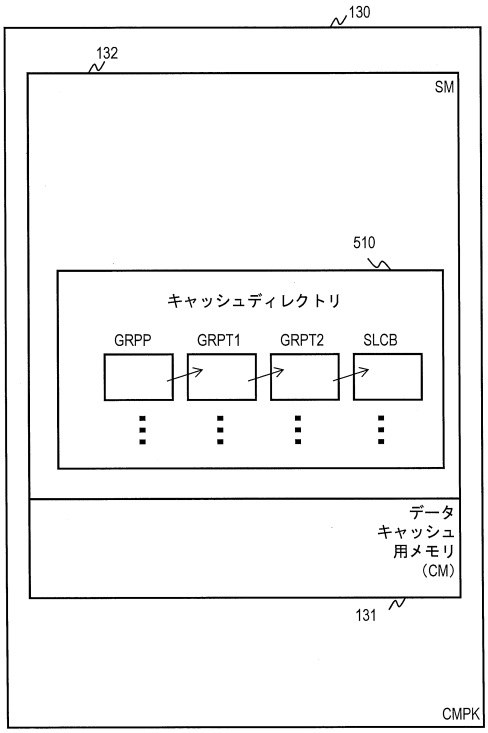
【図１】


ホスト計算機
180
管理
計算機
20
190
10
170
170
170
170
100
100
140
I/F
141
I/F
112
101
転送回路
転送回路
FEPK
BEPK
140
142
150
120
120
130
121
121
121
121
MP
MP
MP
MP
制御情報用
共有メモリ
（SM）
データ
キャッシュ
用メモリ
（CM）
123
LM
122
MPPK
132
131
CMPK
130
ストレージシステム

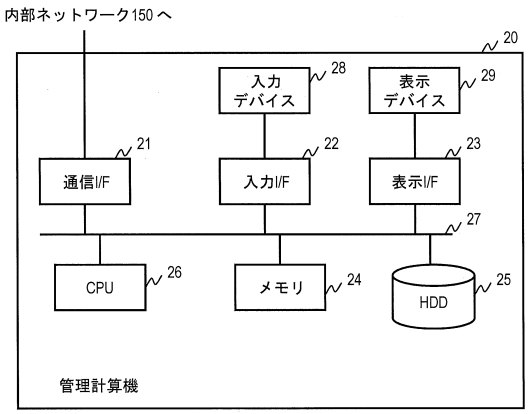
【図２】


120
121
121
121
121
MP
MP
MP
MP
123
122
LM
性能ブースト機能有効化テーブル
210
ボリューム毎性能ブースト機能有効化テーブル
220
メディア種別テーブル
230
RAIDレベルテーブル
240
ボリューム毎ヒット率テーブル
250
ヒット率閾値テーブル
260
MP稼働率テーブル
270
MP稼働率閾値テーブル
280
CM稼働率テーブル
290
CM稼働率閾値テーブル
300
キャッシュディレクトリ
GRPP
GRPT1
GRPT2
SLCB
310
MPPK

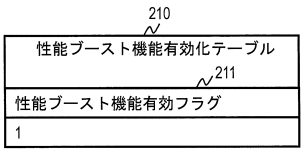
【図３】


130
132
SM
510
キャッシュディレクトリ
GRPP
GRPT1
GRPT2
SLCB
データ
キャッシュ
用メモリ
（CM）
131
CMPK

【図４】


内部ネットワーク150へ
20
入力
デバイス
28
表示
デバイス
29
21
22
23
通信I/F
入力I/F
表示I/F
27
CPU
26
メモリ
24
HDD
25
管理計算機

【図５】


210
性能ブースト機能有効化テーブル
211
性能ブースト機能有効フラグ
1

【図６】

220

ボリューム毎性能ブースト機能有効化テーブル	
221	222
論理ボリューム番号	性能ブースト機能有効化フラグ
1	1
2	0
3	0
...	...

【図７】

230

メディア種別テーブル	
231	232
RAIDグループ番号	メディア種別
1	SSD
2	SATA
3	SAS
...	...

【図８】

240

RAIDレベルテーブル	
241	242
RAIDグループ番号	RAIDレベル
1	5
2	6
3	1
...	...

【図９】

250

ボリューム毎ヒット率テーブル				
251	252	253	254	255
論理ボリューム番号	ヒット率[%]	I/O数	ヒット数	低ヒット率フラグ
1	1	135	2	1
2	20	355	40	0
3	80	1000	800	0
...	...			...

【図１０】

260

ヒット率閾値テーブル
261
ヒット率閾値[%]
5

【図１１】

270

MP稼働率テーブル			
271	272	273	274
マイクロプロセッサ番号	稼働率[%]	過負荷判定フラグ	稼働時間[ms]
1	65	1	650
2	20	0	200
...	...		

【図１２】

280

MP稼働率閾値テーブル
281
マイクロプロセッサ稼働率閾値[%]
80

【図１３】

290

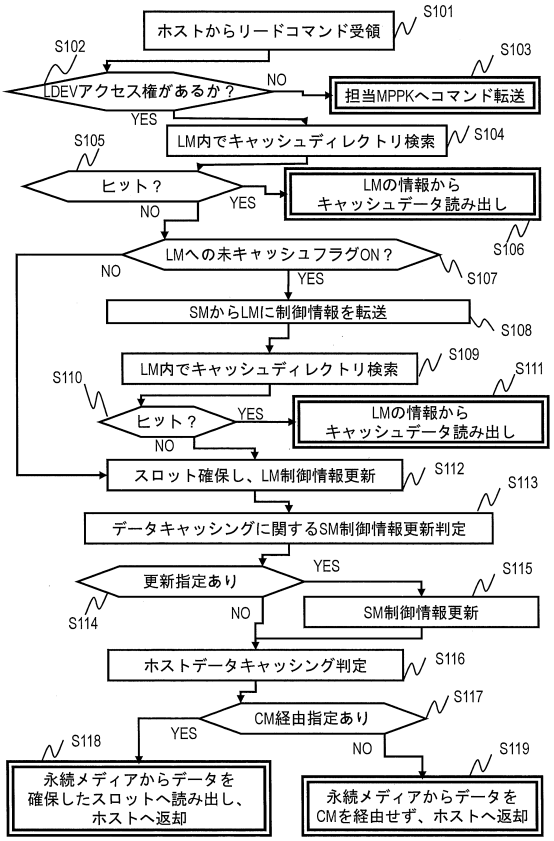
CM稼働率テーブル		
291	292	293
CMPK番号	稼働率[%]	過負荷判定フラグ
1	44	0
2	84	1
...	...	

【図１４】

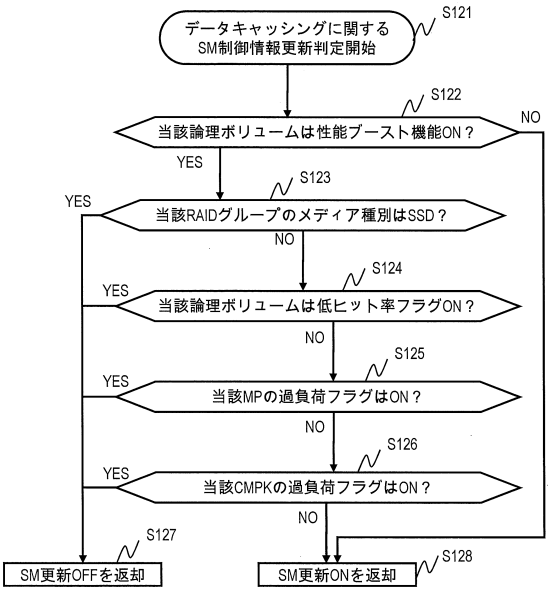
300

CM稼働率閾値テーブル
301
閾値CMPK稼働率[%]
80

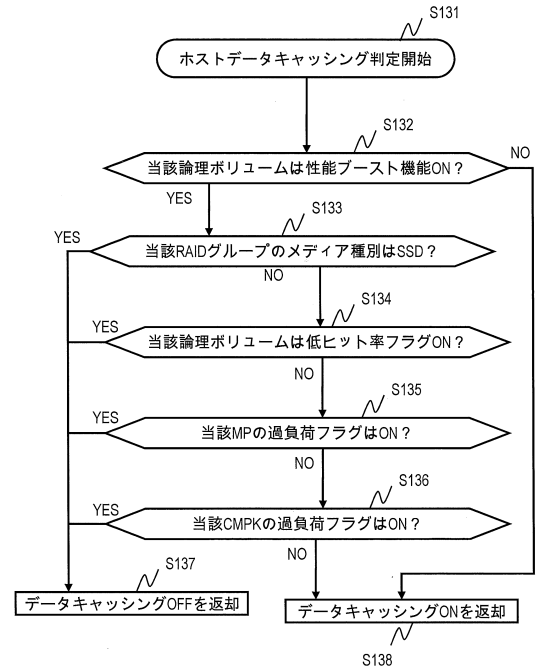
【図１５】



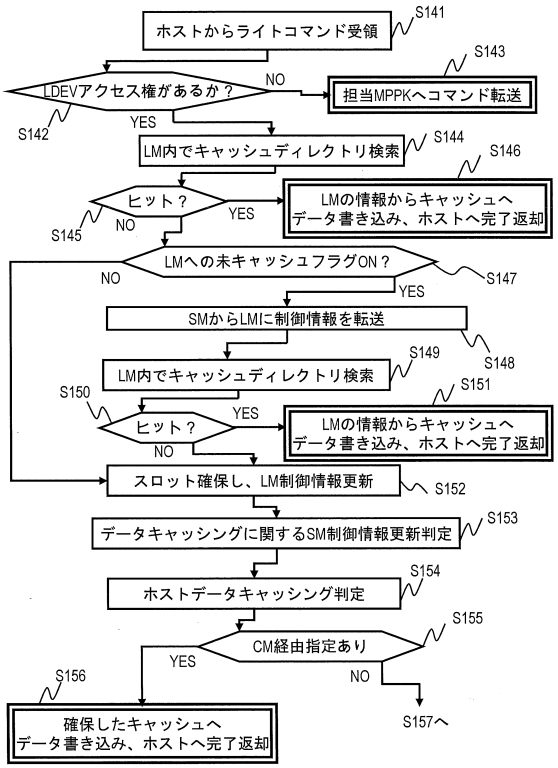
【図１６】



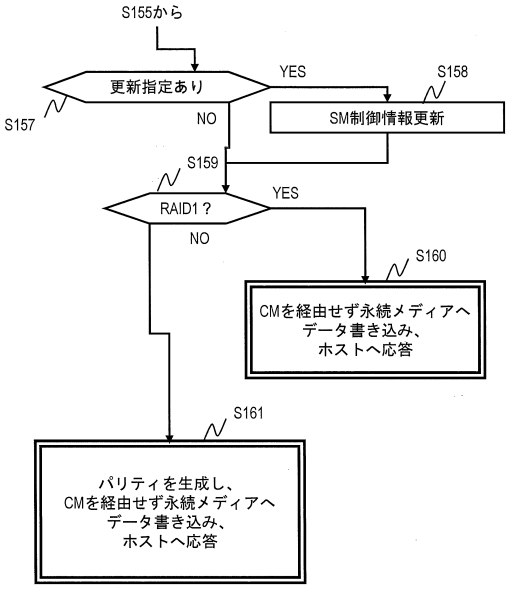
【図１７】



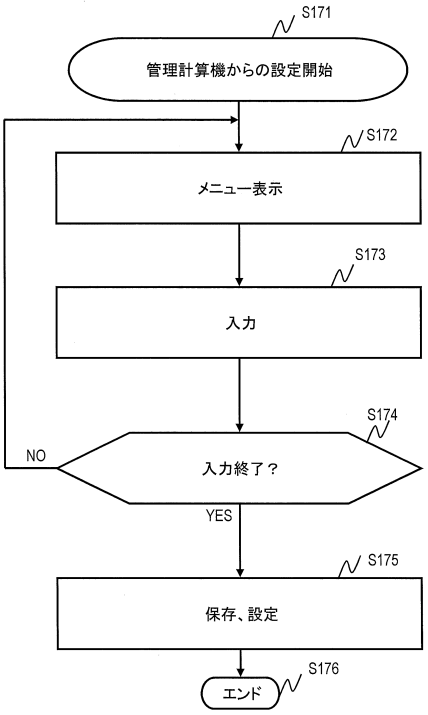
【図１８Ａ】



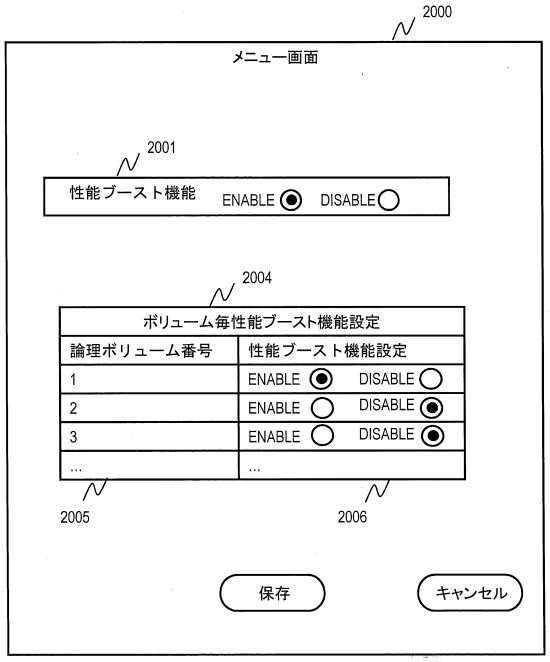
【図１８Ｂ】



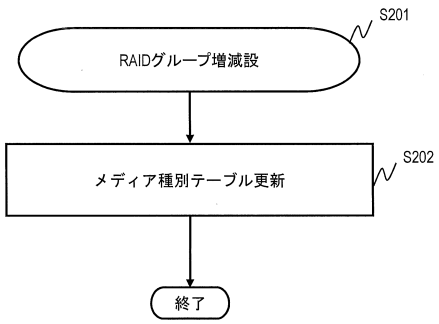
【図１９】



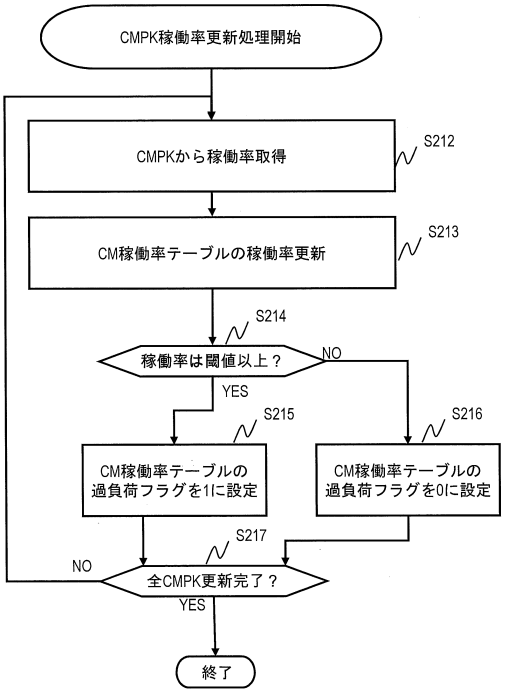
【図２０】



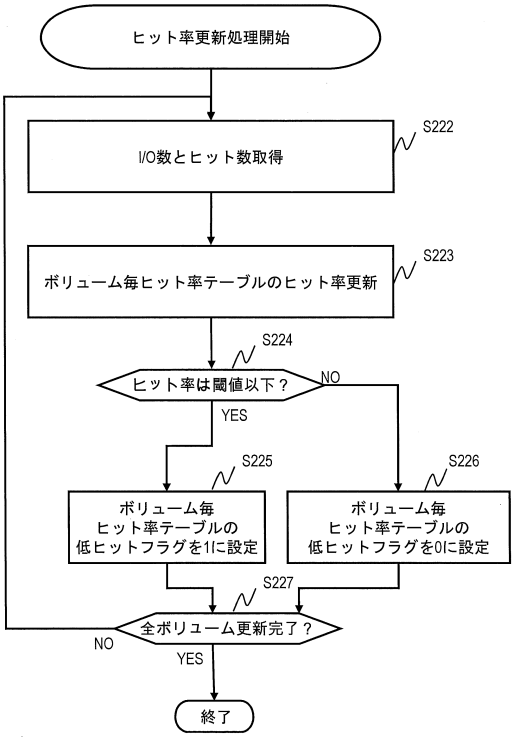
【図２１】



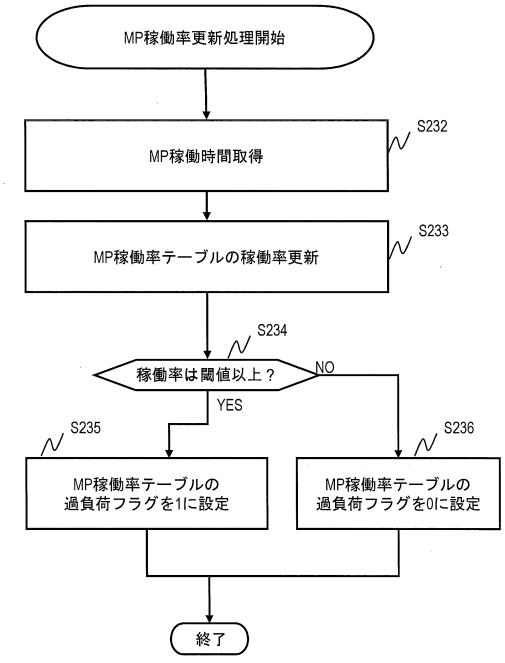
【図２２】



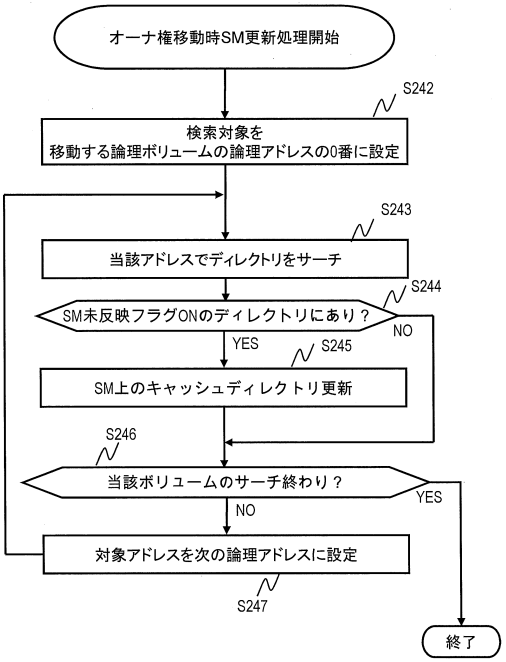
【図２３】



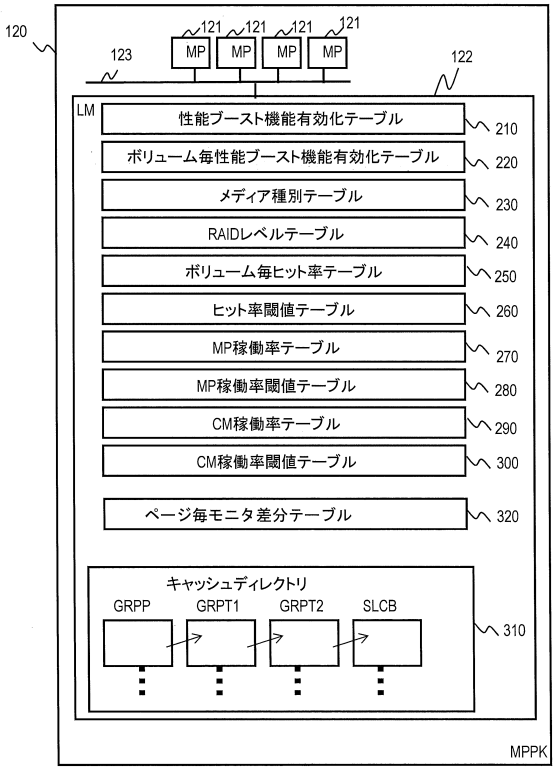
【図２４】



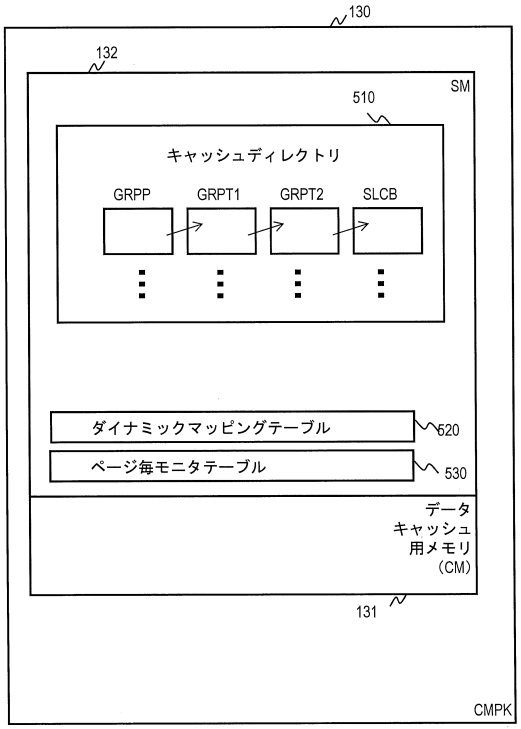
【図２５】



【図２６】



【図２７】



【図２８】

520

ダイナミックマッピングテーブル

521 プール番号	522 仮想ボリューム番号	523 論理アドレス	524 プールボリューム番号	525 論理アドレス	526 モニタ情報インデックス番号
1	1	0x0000	204	0x0040	1
1	1	0x0010	201	0x0050	2
1	2	0x0000	203	0x0020	3
1	2	0x0030	202	0x0040	4
1	2	0x0020	201	0x0010	5

【図２９】

530

ページ毎モニタテーブル

531 モニタ情報インデックス番号	532 I/Oカウンタ現在 (32bit)	533 I/Oカウンタ前回 (32bit)
1	1000	2000
2	613	12124
3	232	123
4	35	32
5	2	325

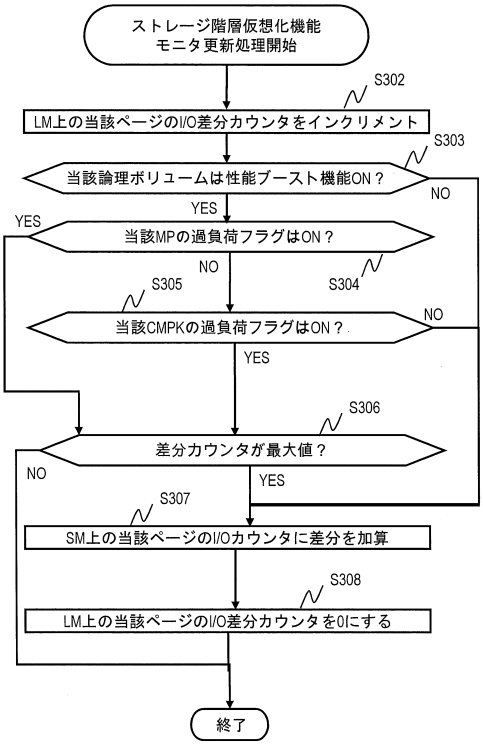
【図３０】

320

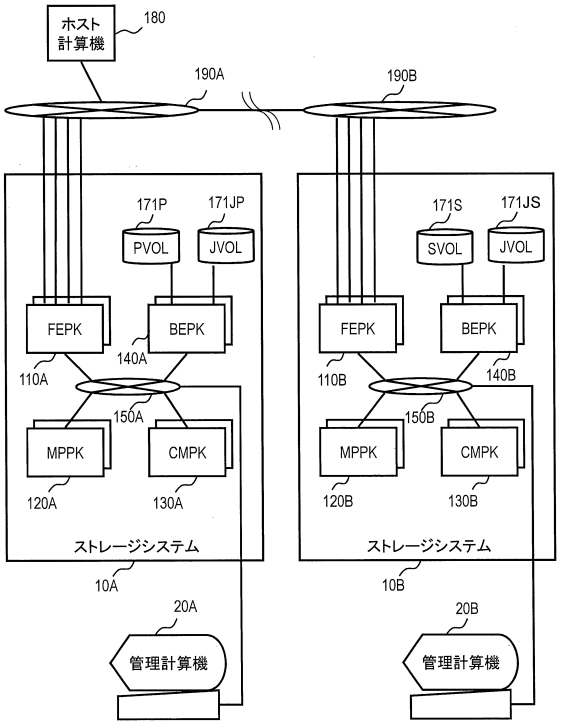
ページ毎モニタ差分テーブル

321 モニタ情報インデックス番号	322 I/O差分カウンタ (8bit)
1	56
2	61
3	23
4	35
5	2

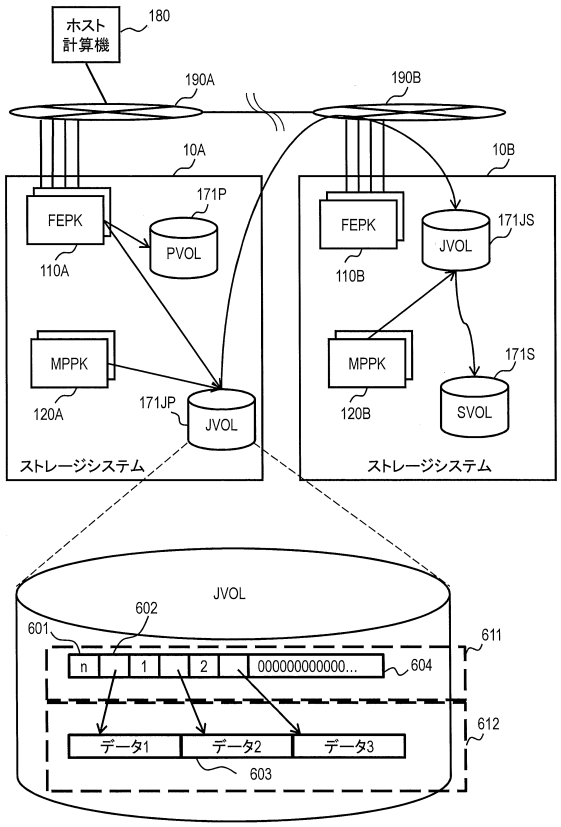
【図３１】



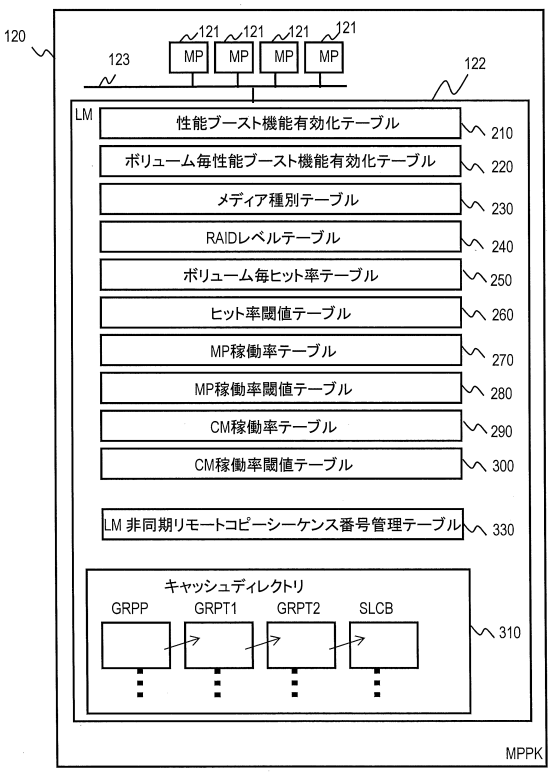
【図３２】



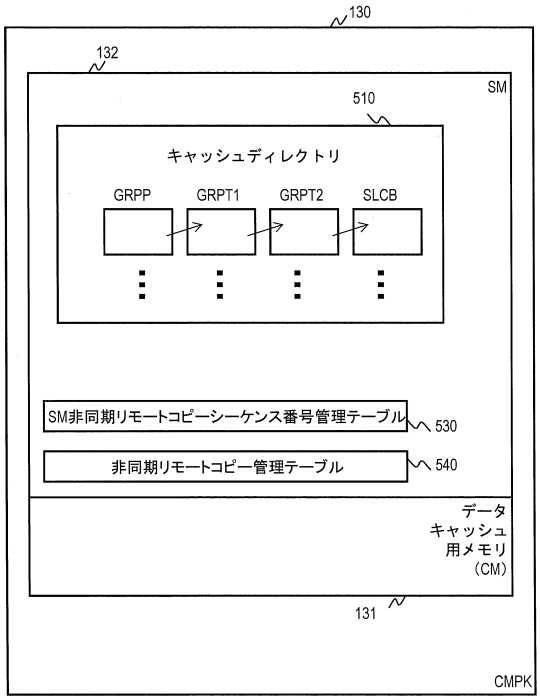
【図３３】



【図３４】



【図３５】



【図３６】

330

LM非同期リモートコピーシーケンス番号管理テーブル

331 JVOL番号	332 シーケンス番号	333 シーケンス番号差分
1	312	56
2	422	61
3	546	23
4	470	35
…	…	

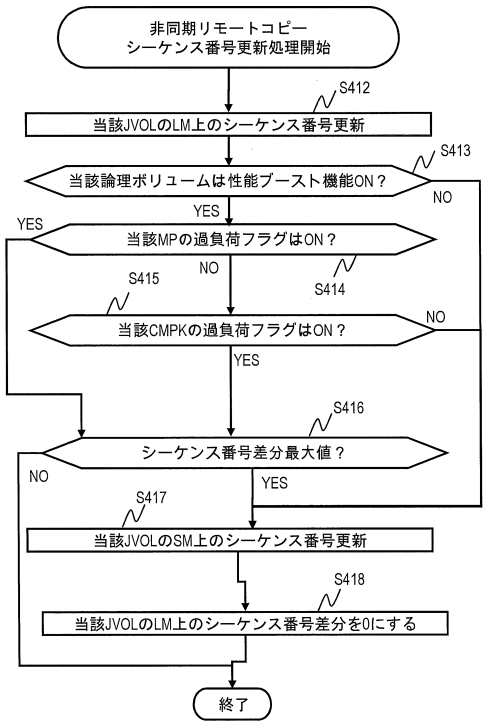
【図３７】

530

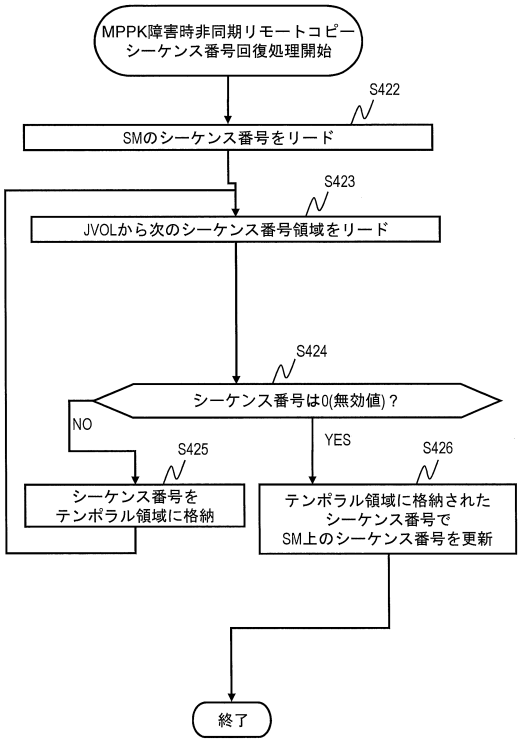
SM非同期リモートコピーシーケンス番号管理テーブル

531 JVOL番号	532 シーケンス番号
1	256
2	361
3	523
4	435
…	…

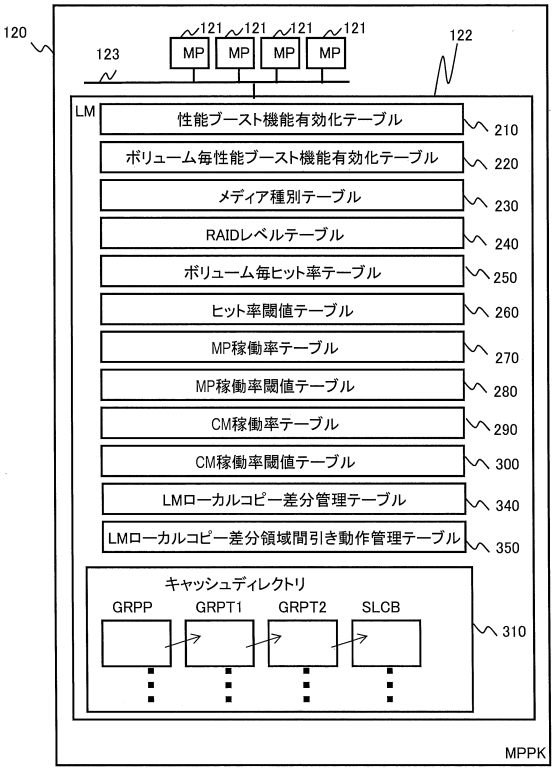
【図３８】



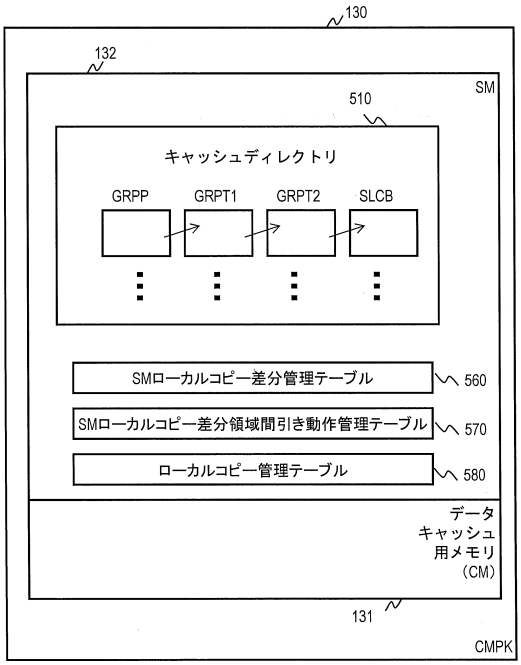
【図３９】



【図４０】



【図４１】



【図４２】

340

LMローカルコピー差分管理テーブル

341 ボリューム番号	342 論理アドレス	343 差分有ビット列
1	0x0000	01101111
1	0x0008	00000000
1	…	…
1	0xffff	00000000
2	0x0000	11110110
…	…	…

【図４３】

560

SMローカルコピー差分管理テーブル

561 ボリューム番号	562 論理アドレス	563 差分有ビット列
1	0x0000	01101111
1	0x0008	00000000
1	…	…
1	0xffff	00000000
2	0x0000	11110110
…	…	…

【図４４】

350

LMローカルコピー差分領域間引き動作管理テーブル

351 ボリューム番号	352 論理アドレス	353 間引き中ビット列
1	0x0000	01101111
1	0x0040	00000000
1	…	…
1	0xffff	00000000
2	0x0000	11110110
…	…	…

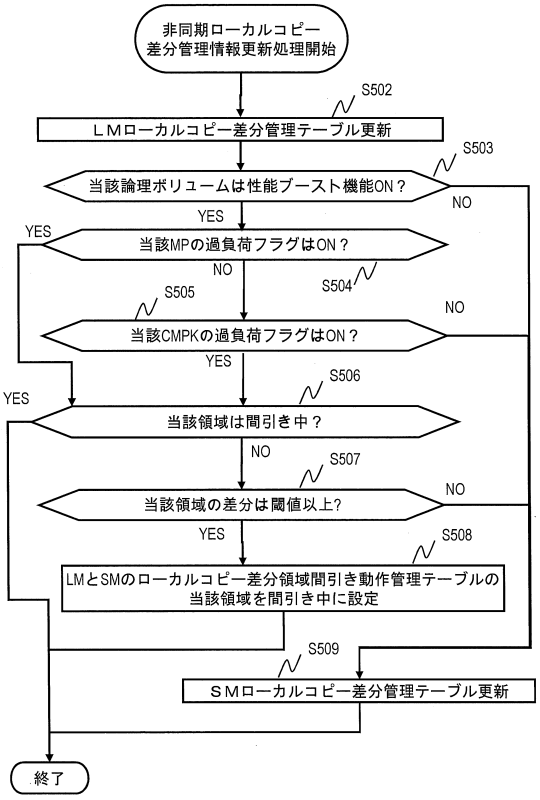
【図４５】

570

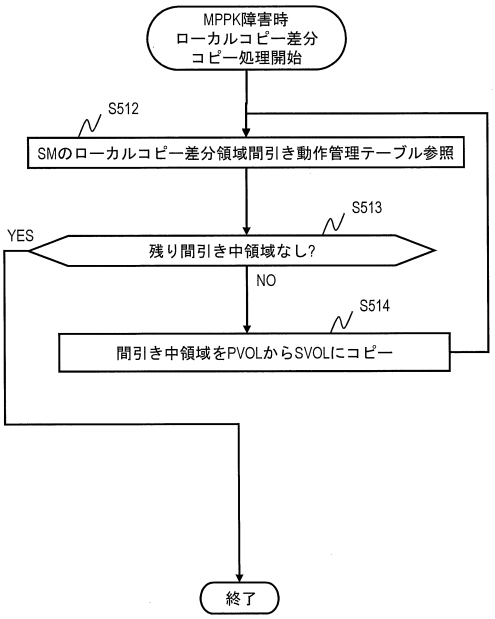
SMローカルコピー差分領域間引き動作管理テーブル

571 ボリューム番号	572 論理アドレス	573 間引き中ビット列
1	0x0000	01101111
1	0x0040	00000000
1	…	…
1	0xffff	00000000
2	0x0000	11110110
…	…	…

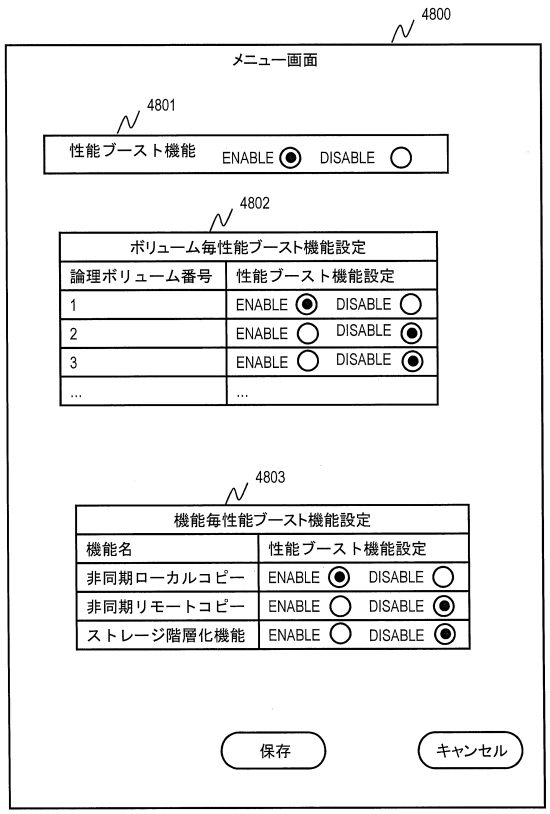
【図４６】



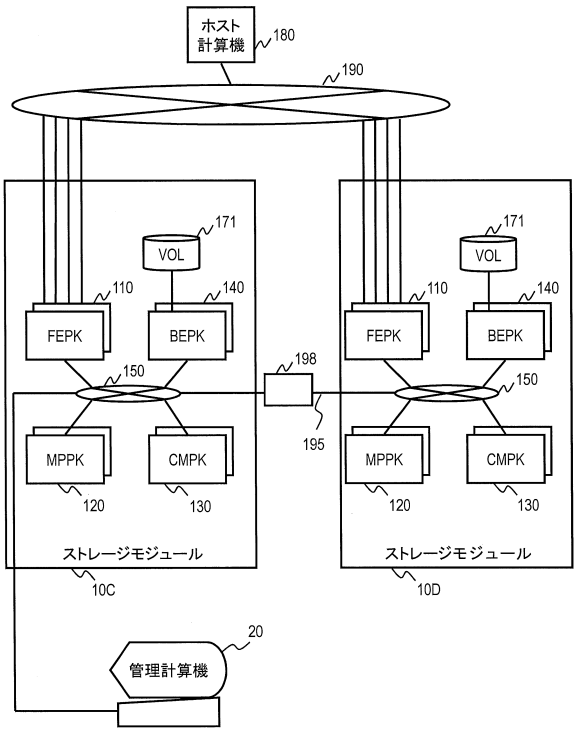
【図４７】



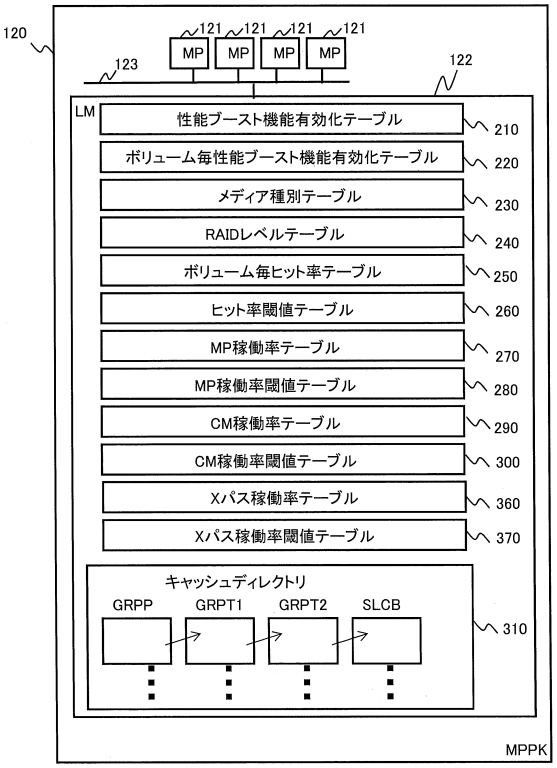
【図４８】



【図４９】



【図５０】



【図５１】

360

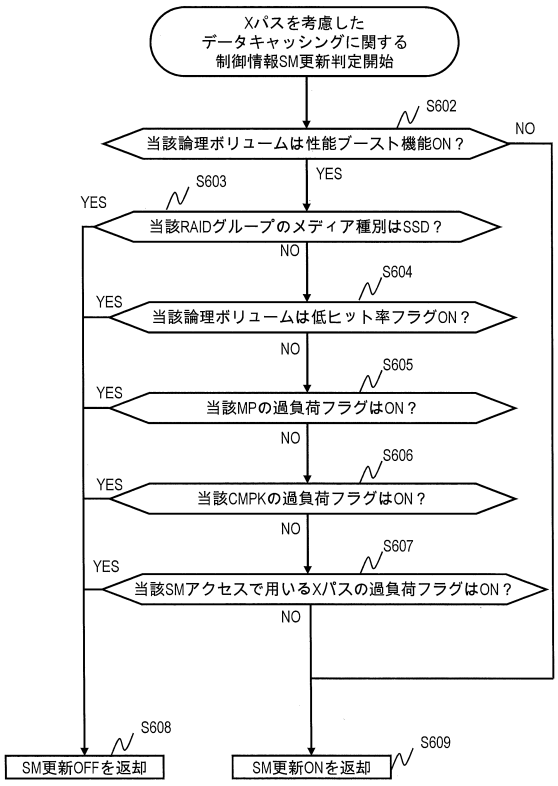
Xパス稼働率テーブル		
361 Xパス番号	362 稼働率[%]	363 過負荷判定フラグ
1	44	0
2	84	1
…	…	

【図５２】

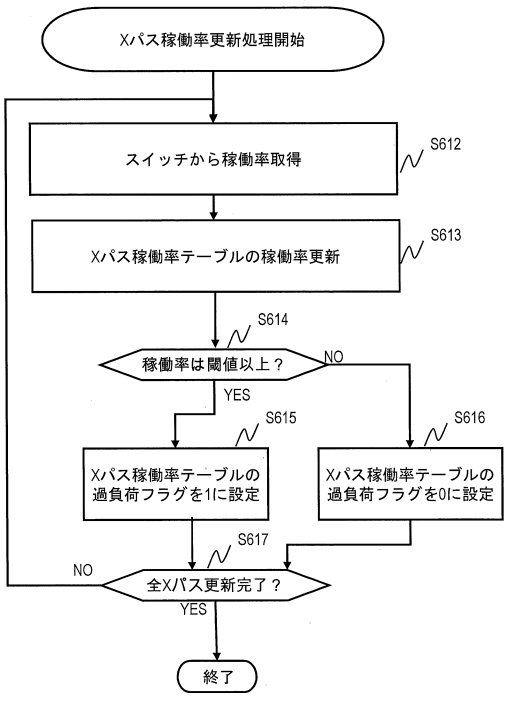
370

Xパス稼働率閾値テーブル
371 Xパス稼働率閾値[%]
80

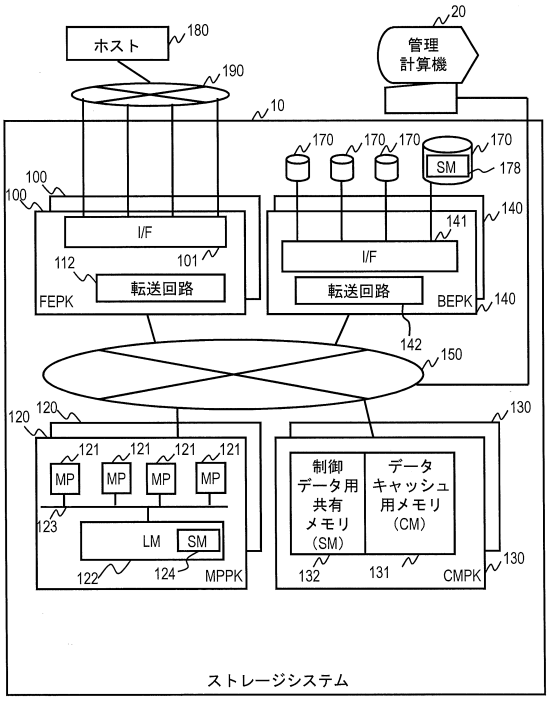
【図５３】



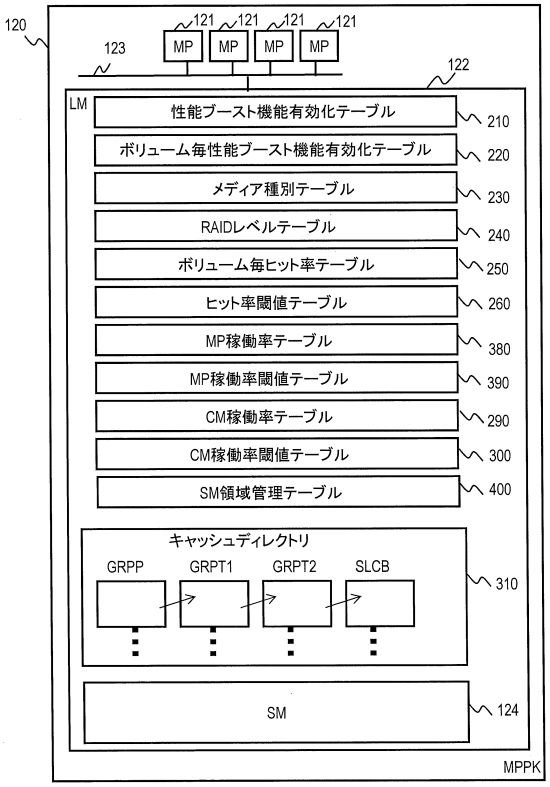
【図５４】



【図５５】



【図５６】



【図５７】

380

MP稼働率テーブル

381 MP番号	382 稼働率[%]	383 過負荷判定フラグ1	384 過負荷判定フラグ2	385 稼働時間[ms]
1	85	1	1	650
2	40	0	0	200
3	68	0	1	340
...	...			

【図５８】

390

MP稼働率閾値テーブル

391 MP稼働率閾値1[%]	392 MP稼働率閾値2[%]
80	60

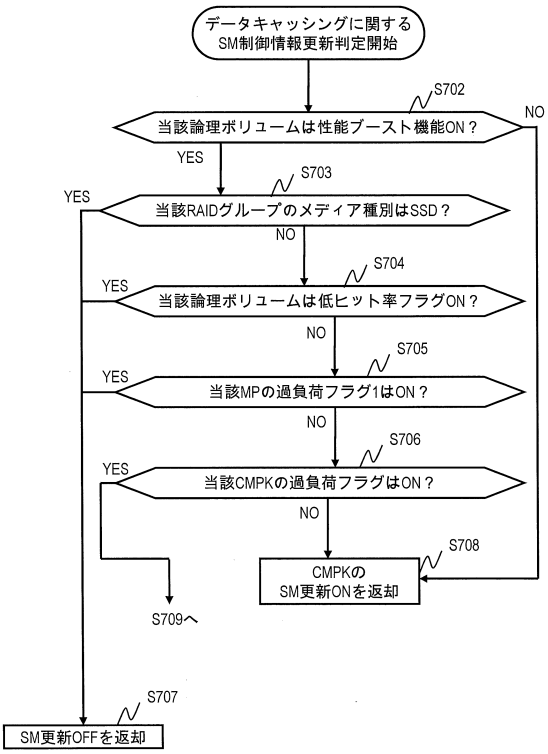
【図５９】

400

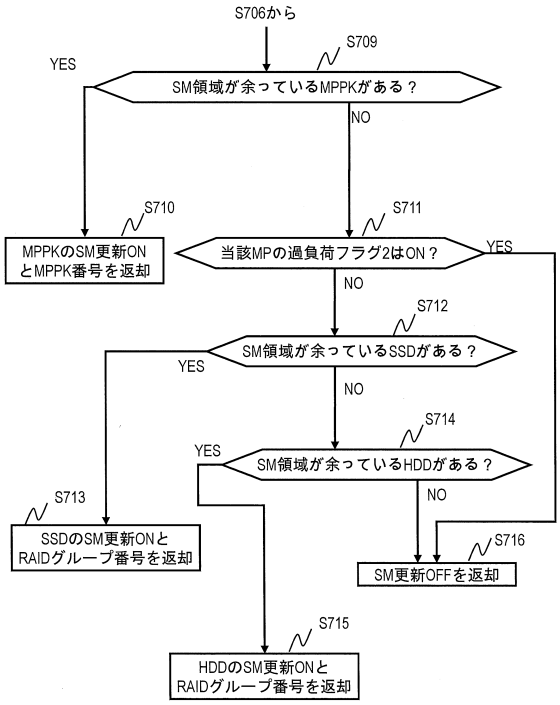
共有メモリ領域管理テーブル

401 種別	402 番号	403 先頭アドレス	404 空き容量
MPPK	0	0x40000000	1
MPPK	1	0x40000000	0
...	...		0
MPPK	16	0x40000000	3
SSD	0	0x80000000	10
SSD	1	0x80000000	10
...	...		
SSD	4	0x80000000	10
HDD	0	0x80000000	10
HDD	1	0x80000000	10
...	...		
HDD	128	0x80000000	10

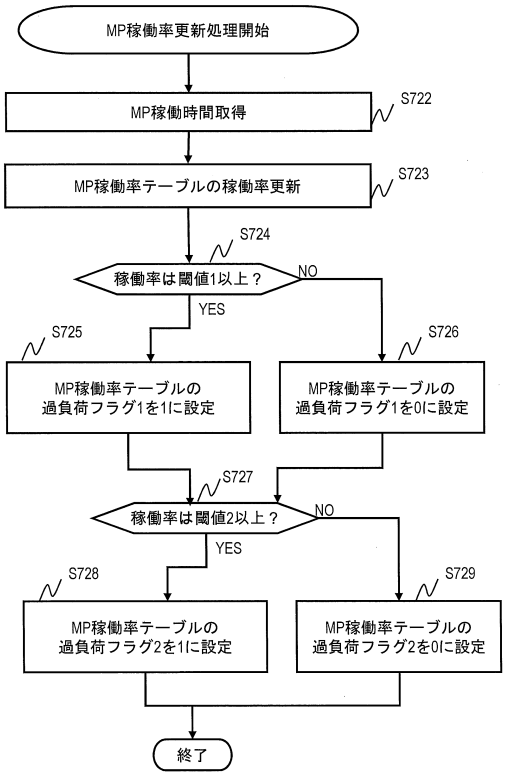
【図６０Ａ】



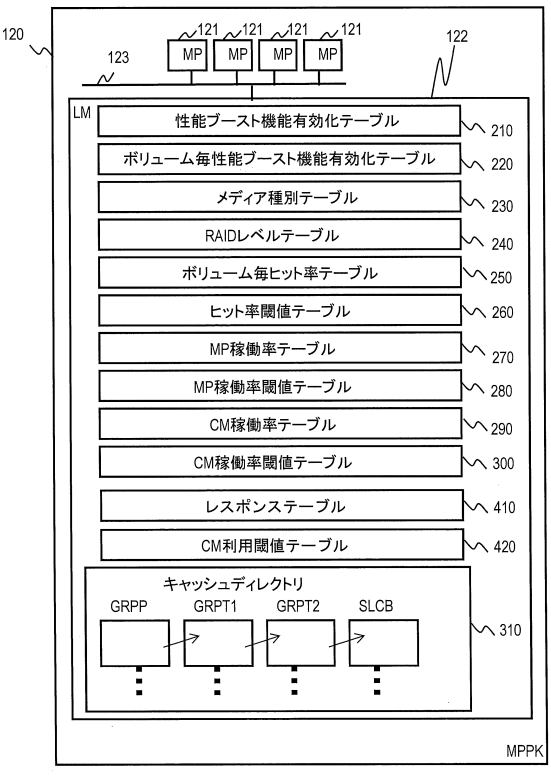
【図６０Ｂ】



【図６１】



【図６２】



【図６３】

410

レスポンステーブル

411 種別	412 レスポンスタイム
CMPK	0.1ms
SSD	1ms
SATA 7.2krpm	10ms
SAS 15krpm	5ms
...	...

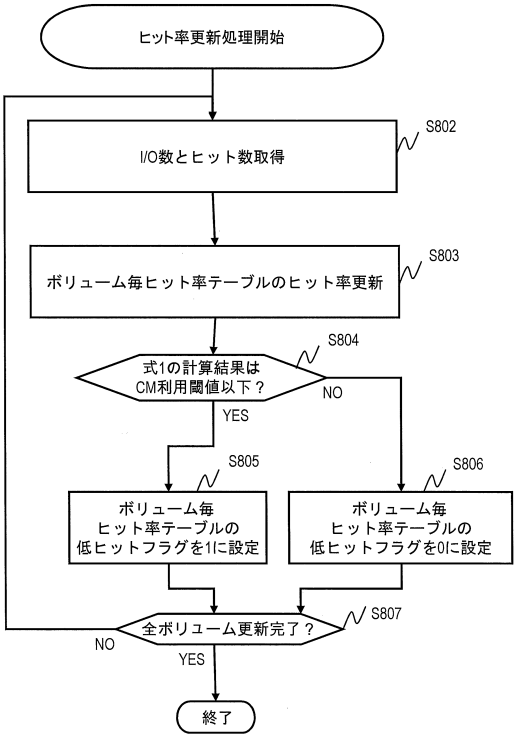
【図６４】

420

CM利用閾値テーブル

421
レスポンス向上
0.8ms

【図６５】



フロントページの続き

審査官　宮久保　博幸

(56)参考文献　特開２００３－２２８４６１（ＪＰ，Ａ）
特開２００５－１９６６７３（ＪＰ，Ａ）
特開２００７－０７９９５８（ＪＰ，Ａ）
米国特許出願公開第２０１１／０１５３９５４（ＵＳ，Ａ１）
特開２０１０－０８６２１１（ＪＰ，Ａ）
特表２０１２－５０４７９２（ＪＰ，Ａ）

(58)調査した分野(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　　３／０６
Ｇ０６Ｆ　１３／１０