(19)日本国特許庁(JP)

# (12)特許公報(B2)

(11)特許番号
特許第7201775号
(P7201775)

(45)発行日　令和5年1月10日(2023.1.10)

(24)登録日　令和4年12月26日(2022.12.26)

(51)国際特許分類

| | | FI | | |
|---|---|---|---|---|
| *G06F 3/06 (2006.01)* | | G06F | 3/06 | 304B |
| *G06F 13/10 (2006.01)* | | G06F | 3/06 | 302A |
| *G06F 11/20 (2006.01)* | | G06F | 13/10 | 340B |
| *G06F 12/0866(2016.01)* | | G06F | 11/20 | 689 |
| *G06F 12/0868(2016.01)* | | G06F | 12/0866 | 100 |

請求項の数　6　（全23頁）　最終頁に続く

(21)出願番号　特願2021-182809(P2021-182809)
(22)出願日　令和3年11月9日(2021.11.9)
(62)分割の表示　特願2018-210691(P2018-210691)の分割
原出願日　平成30年11月8日(2018.11.8)
(65)公開番号　特開2022-10181(P2022-10181A)
(43)公開日　令和4年1月14日(2022.1.14)
審査請求日　令和3年11月9日(2021.11.9)

(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　110000279
弁理士法人ウィルフォート国際特許事務所
(72)発明者　鴨生 悠冬
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　達見 良介
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　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

最終頁に続く

(54)【発明の名称】　ストレージシステム、データ管理方法、及びデータ管理プログラム

(57)【特許請求の範囲】

【請求項１】

複数のコントローラと、データを格納可能な記憶デバイスユニットとを有するストレージシステムであって、

前記コントローラは、

プロセッサ部と、

メモリと、を有し、

ライト要求にかかる新データが第１の前記コントローラの第１の前記メモリに格納された場合に、前記ライト要求にかかる新データにかかる第２のコントローラの第２のメモリ及び第３のコントローラの第３のメモリの旧データの状態がダーティである場合に、前記第１のコントローラは、ライト要求に対応する新データが格納された第１のメモリから、前記新データを第２のコントローラの第２のメモリに対して転送し、前記第２のメモリへの転送が完了した後に、前記新データを第３のコントローラの第３のメモリに対して転送することにより、前記第２のメモリと前記第３のメモリに別々に転送を行って前記旧データを上書きし、

前記ライト要求にかかる新データにかかる前記第２のメモリ及び前記第３のメモリの旧データの状態がダーティではない場合に、前記新データを前記第２のメモリと前記第３のメモリに並行して転送を行って前記旧データを上書きする

ストレージシステム。

【請求項２】

前記別々に転送を行う場合には、前記第２のメモリへの転送の成功を確認してから前記第３のメモリへの転送を行い、
前記転送時に障害が発生した場合には、前記障害により損傷が発生していない前記新データまたは旧データを用いて処理を行う
請求項１に記載のストレージシステム。
【請求項３】
前記転送時に障害が発生した場合に、
前記第２のメモリまたは前記第３のメモリのいずれかの転送が正常に行われた場合には、正常に転送された新データを前記記憶デバイスユニットにデステージし、
前記第２のメモリまたは前記第３のメモリのいずれかにも転送が正常に行われていない場合には、前記第２のメモリまたは前記第３のメモリに格納済みの旧データを前記記憶デバイスユニットにデステージして、前記第２のメモリまたは前記第３のメモリのデータをダーティでなくする
請求項２に記載のストレージシステム。
【請求項４】
前記第１のコントローラは、前記第２のコントローラの第２のメモリ及び前記第３のコントローラの第３のメモリに直接アクセス可能である
請求項１に記載のストレージシステム。
【請求項５】
前記転送は、前記第１のコントローラのＤＭＡ（Ｄｉｒｅｃｔ　Ｍｅｍｏｒｙ　Ａｃｃｅｓｓ）部が行う
請求項４に記載のストレージシステム。
【請求項６】
複数のコントローラと、データを格納可能な記憶デバイスユニットとを有するストレージシステムによるデータ管理方法であって、
前記コントローラは、
プロセッサ部と、
メモリと、を有し、
ライト要求にかかる新データが第１の前記コントローラの第１の前記メモリに格納された場合に、前記ライト要求にかかる新データにかかる第２のコントローラの第２のメモリ及び第３のコントローラの第３のメモリの旧データの状態がダーティである場合に、前記第１のコントローラは、ライト要求に対応する新データが格納された第１のメモリから、前記新データを第２のコントローラの第２のメモリに対して転送し、前記第２のメモリへの転送が完了した後に、前記新データを第３のコントローラの第３のメモリに対して転送することにより、前記第２のメモリと前記第３のメモリに別々に転送を行って前記旧データを上書きし、
前記ライト要求にかかる新データにかかる前記第２のメモリ及び前記第３のメモリの旧データの状態がダーティではない場合に、前記新データを前記第２のメモリと前記第３のメモリに並行して転送を行って前記旧データを上書きする
データ管理方法。
【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おけるデータを管理する技術に関する。
【背景技術】
【０００２】
ストレージシステムでは、電源喪失などの障害からデータを保護するために、複数のストレージコントローラ間でデータを多重化（一般には二重化）している。また、ストレージシステムでは、複数の専用回路を用い、ライトデータをキャッシュ領域に同時に多重化することも行われている。

【０００３】
例えば、特許文献１には、第一キャッシュとＦＩＦＯバッファとにデータを格納した時点でホストにライト完了を送信し、その後、ＦＩＦＯバッファから第二キャッシュにデータを送付することで、キャッシュを二重化するストレージシステムでの書込みを高速化する技術が開示されている。
【０００４】
一方、専用回路の開発コスト削減を目的として、特許文献２には、専用回路の処理を汎用コントローラでエミュレーションするストレージシステムにおいて、データの一貫性を保証する技術が開示されている。この技術では、コントローラ外部から受領するデータをバッファ領域に格納し、このコントローラがバッファ領域からキャッシュ領域に転送することで、Ｉ／Ｏ処理中に障害が発生しても、データの破壊を防ぐことができるようにしている。
【先行技術文献】
【特許文献】
【０００５】
【文献】特開２００５－４４０１０号公報
国際公開第２０１５／０５２７９８号
【発明の概要】
【発明が解決しようとする課題】
【０００６】
ストレージシステムの性能向上のために、汎用コントローラを多数搭載したストレージシステムが登場している。
【０００７】
このようなストレージシステムにおいて、メモリ容量の増加を抑える観点では、データの多重化を二多重とすることが望ましい。
【０００８】
例えば、このようなストレージシステムにおいてデータを二重化する場合においては、ホストからデータを受領するコントローラと、データの二重化先のコントローラとが全て異なる場合がある。このような場合に、特許文献２に記載の技術では、二重化先のコントローラのそれぞれにおいて、バッファ領域からキャッシュ領域へデータを転送する必要があり、コントローラのプロセッサへの処理負荷が掛かり、コントローラの性能が低下してしまう虞がある。
【０００９】
本発明は、上記事情に鑑みなされたものであり、その目的は、コントローラのプロセッサへの処理負荷を抑制しつつ、適切にデータの一貫性を確保することのできる技術を提供することにある。
【課題を解決するための手段】
【００１０】
上記目的を達成するため、一観点に係るストレージシステムは、複数のコントローラと、データを格納可能な記憶デバイスユニットとを有するストレージシステムであって、メモリに直接アクセス可能であるとともに、他のコントローラとの通信が可能なＤＭＡ（Ｄｉｒｅｃｔ　Ｍｅｍｏｒｙ　Ａｃｃｅｓｓ）部を有し、コントローラは、プロセッサ部と、データを一時的に格納するバッファ領域と、データをキャッシュするキャッシュ領域とを有するメモリと、を有し、ライト要求にかかる新データがバッファ領域に格納された場合に、コントローラのプロセッサ部は、ＤＭＡ部を用いて、ライト要求に対応する新データが格納されたバッファ領域から、複数のコントローラのキャッシュ領域に対して、他のバッファ領域を介さずに順次転送させる。
【発明の効果】
【００１１】
本発明によれば、コントローラのプロセッサへの処理負荷を抑制しつつ、適切にデータ

の一貫性を確保することができる。
【図面の簡単な説明】
【００１２】
【図１】図１は、実施例１に係るストレージシステムのライト処理の概要を説明する図である。
【図２】図２は、実施例１に係るストレージシステムの障害発生時のライト処理の概要を説明する図である。
【図３】図３は、実施例１に係る計算機システムの構成図である。
【図４】図４は、実施例１に係るコントローラ状態管理情報のデータ構造の一例を示す図である。
【図５】図５は、実施例１に係るキャッシュ状態管理情報のデータ構造の一例を示す図である。
【図６】図６は、実施例１に係る転送管理情報のデータ構造の一例を示す図である。
【図７】図７は、実施例１に係る転送状態管理情報のデータ構造の一例を示す図である。
【図８】図８は、実施例１に係る逐次転送依頼処理のフローチャートである。
【図９】図９は、実施例１に係る逐次転送完了待ち処理のフローチャートである。
【図１０】図１０は、実施例１に係る逐次転送処理のフローチャートである。
【図１１】図１１は、実施例１に係る障害対応処理のフローチャートである。
【図１２】図１２は、実施例２に係る逐次転送完了待ち処理のフローチャートである。
【図１３】図１３は、実施例２に係る障害対応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】
【００１３】
以下、本発明の実施例を、図面を用いて説明する。ただし、本発明は以下に示す実施の形態の記載内容に限定して解釈されるものではない。本発明の思想ないし趣旨から逸脱しない範囲で、その具体的構成を変更し得ることは当業者であれば容易に理解される。
【００１４】
以下に説明する発明の構成において、同一又は類似する構成又は機能には同一の符号を付し、重複する説明は省略する。
【００１５】
本明細書等における「第１」、「第２」、「第３」等の表記は、構成要素を識別するために付するものであり、必ずしも、数又は順序を限定するものではない。
【００１６】
図面等において示す各構成の位置、大きさ、形状、及び範囲等は、発明の理解を容易にするため、実際の位置、大きさ、形状、及び範囲等を表していない場合がある。したがって、本発明では、図面等に開示された位置、大きさ、形状、及び範囲等に限定されない。
【００１７】
また、以下の説明における用語の意味は、下記の通りである。
（＊）「ＰＤＥＶ」は、不揮発性の物理的な記憶デバイスの略である。複数のＰＤＥＶで複数のＲＡＩＤグループが構成されてよい。「ＲＡＩＤ」は、Redundant Array of Independent (or Inexpensive) Disksの略である。ＲＡＩＤグループはパリティグループと呼ばれてもよい。
（＊）ＨＣＡ（Host Channel Adaptor）は、ＣＰＵに指示され、コントローラ間の通信を行うデバイスである。ＨＣＡは、例えば、ＤＭＡ（Direct Memory Access）部の一例であり、メモリに直接アクセスすることができる。
（＊）プロセッサ部は、１以上のプロセッサを含む。少なくとも１つのプロセッサは、典型的には、ＣＰＵ（Central Processing Unit）のようなマイクロプロセッサである。１以上のプロセッサの各々は、シングルコアでもよいしマルチコアでもよい。プロセッサは、処理の一部または全部を行うハードウェア回路を含んでもよい。
【実施例１】
【００１８】

まず、実施例１に係る計算機システムについて説明する。

【００１９】

図１は、実施例１に係るストレージシステムのライト処理の概要を説明する図である。図１は、二重化先のコントローラ２２（＃１、＃２）それぞれのキャッシュ領域２４３（＃１、＃２）に対して逐次にデータを転送するライト処理の流れを示している。

【００２０】

本実施例に係る計算機システム１００のストレージシステム２は、複数のコントローラ２２（コントローラ＃０，＃１，＃２）を備えている。複数のコントローラ２２は、相互に接続されている。コントローラ２２は、例えば、ストレージシステム専用のコントローラではなくて、汎用のコントローラである。コントローラ２２は、ＦＥ－Ｉ／Ｆ２１０と、プロセッサ部の一例としてのＣＰＵ２３０と、メモリ２４０とを有する。メモリ２４０は、バッファ領域２４２及びキャッシュ領域２４３を有するとともに、転送状態管理情報２４７を格納している。

【００２１】

ストレージシステム２においては、各コントローラ２２がホスト計算機（ホストともいう）１からのＩ／Ｏ要求を並列に処理できるよう、Ｉ／Ｏ処理対象の空間（例えば、論理ユニット：ＬＵ）ごとに処理担当（この処理担当である権利を、オーナ権という）のコントローラ２２を定めている。例えば、コントローラ＃１が、ＬＵＮ＃０のＬＵに対するオーナ権を持っているとき、ＬＵＮ＃０のＬＵに対するＩ／Ｏ要求は、このコントローラ＃１の制御により処理される。

【００２２】

ホスト計算機１（ホスト計算機＃０、＃１、＃２）は、通信ネットワーク１１を介して、ストレージシステム２と接続されている。ホスト計算機１は、例えば、それぞれ１つのコントローラ２２と接続する。

【００２３】

コントローラ２２は、ホスト計算機１からのライト要求に従うライトデータを、記憶デバイスユニット２０に書き込まず、複数のコントローラ２２内のキャッシュ領域２４３に二重化して格納した後に、ホスト計算機１に対してライト処理の完了を通知する。これにより、高速なライト処理を実現することができる。

【００２４】

コントローラ２２は、ライト要求とは非同期にキャッシュ領域２４３内のライトデータを記憶デバイスユニット２０に書き込む。既にキャッシュ領域２４３に二重化されたライトデータが格納されていて、まだ記憶デバイスユニット２０に書き込まれていない状態（ダーティ状態といい、このライトデータをダーティデータという）において、ホスト計算機１から同じ書き込み先への新たなライトデータを受領するときには、コントローラ２２は、キャッシュ領域２４３内のライトデータの破壊を避けるために、新たなライトデータをバッファ領域２４２に格納する。その後、コントローラ２２は、バッファ領域２４２内に格納した新たなライトデータを複数（二重化の場合には２つ）のコントローラ内のそれぞれのキャッシュ領域２４３に逐次に転送することでライトデータの一貫性を保持する。転送状態管理情報２４７は、二重化先のコントローラ２２のそれぞれのキャッシュ領域２４３に対してライトデータを逐次に転送する際の進捗状況（転送状態）を管理する情報である。

【００２５】

ここで、ライト要求を受領したコントローラ２２以外のコントローラ２２のバッファ領域２４２を介さずに、一のコントローラ２２のキャッシュ領域２４３にライトデータを転送し、転送が完了した後、他のコントローラ２２のキャッシュ領域２４３にライトデータを転送することを「逐次転送」という。

【００２６】

ここで、コントローラ＃０が、コントローラ＃１がオーナ権を有するＬＵに対するライト要求をホスト計算機＃０から受け取った場合のライト処理について説明する。

【００２７】
コントローラ＃０は、ホスト計算機＃０からライト要求を受信した場合、コントローラ＃０のＣＰＵ＃０は、ライト要求の対象となるＬＵのオーナ権を有するコントローラ＃１のＣＰＵ＃１にライト要求を転送する。
【００２８】
ＣＰＵ＃１は、バッファ領域＃０上にライトデータを格納する領域を確保させ、ライトデータに対応するキャッシュ領域２４３（本例では、キャッシュ領域＃１、＃２）上に格納されているデータの状態を確認する。本実施形態では、キャッシュ領域２４３のデータは、ダーティ状態であることとする。キャッシュ領域２４３のデータがダーティ状態（ダーティデータ）であるので、ＣＰＵ＃１は、逐次転送が必要と判断する。
【００２９】
その後、コントローラ＃０のＣＰＵ＃０は、ＦＥ－ＩＦ＃０を介してバッファ領域＃０に確保された領域にライトデータを格納する（ステップＳ１）。
【００３０】
次いで、ＣＰＵ＃１は、ＨＣＡ＃０に、バッファ領域＃０からキャッシュ領域＃２へライトデータをコピー（転送）し、その後、キャッシュ領域＃１にライトデータをコピー（転送）すること（逐次転送）を依頼する（ステップＳ２）。
【００３１】
ＨＣＡ＃０は、ＨＣＡ＃２を介して、バッファ領域＃０からキャッシュ領域＃２へライトデータをコピーする（以降、第一転送という）（ステップＳ３）。この際、ＨＣＡ＃０は、データコピー時にデータに付与された保証コードを確認する。保証コードは、データの格納位置を示す情報（ＶＯＬ番号やＶＯＬのアドレス等）やデータの一貫性を確認する情報（ＣＲＣ（Cyclic Redundancy Check）等）から構成されてもよい。
【００３２】
次に、ＨＣＡ＃０は、ＨＣＡ＃２を介して、転送状態管理情報＃２に、キャッシュ領域＃２へのライトデータの転送受領を格納させ、ＨＣＡ＃２に、転送状態管理情報＃１にキャッシュ領域＃２へのライトデータの転送完了を格納させることを依頼する（ステップＳ４）。依頼を受けたＨＣＡ＃２は、ＨＣＡ＃１を介して、転送状態管理情報＃１にキャッシュ領域＃２へのライトデータの転送完了を格納させる（ステップＳ５）。
【００３３】
次に、ＨＣＡ＃０は、ＨＣＡ＃１を介して、バッファ領域＃０からキャッシュ領域＃１へライトデータをコピーする（以降、第二転送という）（ステップＳ６）。
【００３４】
次に、ＨＣＡ＃０は、ＨＣＡ＃１を介して、転送状態管理情報＃１にキャッシュ領域＃１へのライトデータの転送完了を格納させる（ステップＳ７）。
【００３５】
ＣＰＵ＃１は、転送状態管理情報＃１を参照し、ライトデータの二重化完了を確認する（ステップＳ８）。次に、ＣＰＵ＃１は、ＣＰＵ＃０及びＦＥ－Ｉ／Ｆ＃０を介して、ホスト計算機＃０にライト要求完了を報告する（ステップＳ９）。これにより、ホスト計算機１からのライトデータは、キャッシュ領域＃１と、キャッシュ領域＃２とに二重化して格納される。
【００３６】
なお、上記例では、キャッシュ領域＃２、＃１の順でライトデータを順次転送させていたが、キャッシュ領域＃１、＃２の順としてもよい。
【００３７】
図２は、実施例１に係るストレージシステムの障害発生時のライト処理の概要を説明する図である。図２は、図１に示すライト処理の途中に障害が発生した時のライト処理の概要を示している。
【００３８】
ＨＣＡ＃０が、ＣＰＵ＃１から逐次転送の依頼を受け、逐次転送を実施中にＨＣＡ２５

０やＨＣＡ２５０を繋ぐネットワーク（図３のＨＣＡネットワーク２３）のパス等に障害が発生すると（ステップＳ１１）、キャッシュ領域＃１または＃２内のダーティ状態のライトデータ（ダーティデータ）を破壊してしまう虞がある（ステップＳ１２）。つまり、ダーティデータの一部分だけが新たなライトデータの一部分によって上書きされた別のデータとなってしまう虞がある。

【００３９】

そこで、ライトデータを管理するＣＰＵ＃１は、転送状態管理情報２４７を参照し、正常なダーティデータを保持するキャッシュ領域２４３を特定する（ステップＳ１３）。その後、ＣＰＵ＃１は、特定したキャッシュ領域２４３内のダーティデータをデステージ（すなわち、記憶デバイスユニット２０に転送）する（ステップＳ１４）。さらに、ＣＰＵ＃１は、デステージを完了後に、キャッシュ領域＃１及び＃２内のダーティデータを破棄する。なお、以降の説明では、特に記載しない場合には、デステージの完了後に、そのデステージしたデータに対応するデータが格納されていた複数のキャッシュ領域２４３のデータを破棄するものとする。

【００４０】

以上の処理により、正常なダーティデータを選択して記憶デバイスユニット２０に書き込むことができ、ライトデータの一貫性を保証できる。

【００４１】

次に、本実施例に係る計算機システムについて詳細に説明する。

【００４２】

図３は、実施例１に係る計算機システムの構成図である。

【００４３】

計算機システム１００は、１以上のホスト計算機１と、ストレージシステム２とを備える。ホスト計算機１と、ストレージシステム２とは、ネットワーク１１を介して接続されている。ネットワーク１１は、例えば、ＳＡＮ（Storage Area Network）である。

【００４４】

ストレージシステム２は、複数（例えば、３台以上）のコントローラ２２（コントローラ２２＃０，・・・，＃Ｎ）と、記憶デバイスユニット２０とを有する。複数のコントローラ２２は、ＨＣＡネットワーク２３を介して相互に接続されている。ストレージシステム２の可用性を向上させるため、コントローラ２２毎に専用の電源を用意し、それぞれのコントローラ２２に対して、その専用の電源を用いて給電するようにしてもよい。

【００４５】

コントローラ２２は、通信インタフェースと、記憶デバイスと、それらに接続されたプロセッサとを有する。通信インタフェースは、例えば、ＦＥ－Ｉ／Ｆ（Front End Inter/Face）２１０、ＢＥ－Ｉ／Ｆ（Back End Inter/Face）２２０、及びＨＣＡ２５０である。記憶デバイスは、例えば、メモリ２４０である。プロセッサは、例えば、ＣＰＵ（Central Processing Unit）２３０である。なお、図３においては、コントローラ２２は、１つのメモリ２４０を備えている構成としているが、メモリ２４０を複数備えてもよい。

【００４６】

ＦＥ－Ｉ／Ｆ２１０は、ホスト計算機１等のフロントエンドに存在する外部デバイスと通信するためのインタフェースデバイスである。ＢＥ－Ｉ／Ｆ２２０は、コントローラ２２が記憶デバイスユニット２０と通信するためのインタフェースデバイスである。ＨＣＡ２５０は、各コントローラ２２のメモリ２４０を操作するために他のＨＣＡ２５０と通信するためのインタフェースデバイスである。

【００４７】

メモリ２４０は、例えば、ＲＡＭ（Random Access Memory）であり、バッファ領域２４２と、キャッシュ領域２４３とを含む。また、メモリ２４０は、制御モジュール２４１、コントローラ状態管理情報２４４、キャッシュ状態管理情報２４５、転送管理情報２４６、及び転送状態管理情報２４７を記憶する。なお、メモリ２４０は、不揮発性メモリ

であっても、揮発性メモリであってもよい。
【００４８】
制御モジュール２４１は、ＣＰＵ２３０に実行されることにより、ストレージシステム２全体を制御するためのモジュール（プログラム）である。より具体的には、制御モジュール２４１は、ＣＰＵ２３０に実行されることにより、Ｉ／Ｏ処理の制御等を行う。
【００４９】
バッファ領域２４２は、ホスト計算機１から受領したライトデータを一時的に格納する領域である。
【００５０】
キャッシュ領域２４３は、ホスト計算機１から記憶デバイスユニット２０へ送信されるライトデータをキャッシュする領域である。キャッシュ領域２４３は、ダーティデータを格納することもあるので、バックアップ電源等により不揮発化されていてもよい。
【００５１】
コントローラ状態管理情報２４４は、コントローラ２２が正常状態か、故障状態かを管理するための情報である。キャッシュ状態管理情報２４５は、二重化に使用されているキャッシュ領域２４３を有するコントローラ２２とキャッシュの状態を管理するための情報である。転送管理情報２４６は、逐次転送で転送するライトデータを受信したコントローラ２２と、転送状態管理情報２４７のエントリのアドレスを管理するための情報である。転送状態管理情報２４７は、逐次転送の進捗状況（転送状態）を管理するための情報である。コントローラ状態管理情報２４４、キャッシュ状態管理情報２４５、転送管理情報２４６、及び転送状態管理情報２４７の詳細は、図４乃至図７を参照して後述する。
【００５２】
記憶デバイスユニット２０は、複数のＰＤＥＶ２００を有する。ＰＤＥＶ２００は、ＨＤＤ（Hard Disk Drive）でよいが、他種の記憶デバイス（不揮発性の記憶デバイス）、例えば、ＳＳＤ（Solid State Drive）のようなＦＭ（Flash Memory）デバイスでもよい。記憶デバイスユニット２０は、異なる種類のＰＤＥＶ２００を有してよい。また、複数の同種のＰＤＥＶ２００でＲＡＩＤグループが構成されてよい。ＲＡＩＤグループには、所定のＲＡＩＤレベルに従いデータが格納される。コントローラ２２がホスト計算機１から受信したライトデータに対しては、ＦＥ－Ｉ／Ｆ２１０によって保証コードが付与される。この保証コードが付与されたデータは、ＲＡＩＤグループに格納される。
【００５３】
ＨＣＡ２５０は、ＣＰＵ２３０から指示を受け、自コントローラ２２のメモリ２４０に対する操作や、ＨＣＡネットワーク２３を経由して、他コントローラ２２のメモリ２４０に対する操作を行う。
【００５４】
次に、コントローラ状態管理情報２４４を詳細に説明する。
【００５５】
図４は、実施例１に係るコントローラ状態管理情報のデータ構造の一例を示す図である。
【００５６】
コントローラ状態管理情報２４４は、コントローラ２２ごとのエントリを格納する。コントローラ状態管理情報２３３のエントリは、コントローラＩＤ４０１及び状態４０２のフィールドを含む。コントローラＩＤ４０１には、エントリに対応するコントローラ２２の識別子（コントローラＩＤ）が格納される。状態４０２には、エントリに対応するコントローラ２２の動作状態が格納される。動作状態としては、正常、故障等がある。
【００５７】
次に、キャッシュ状態管理情報２４５を詳細に説明する。
【００５８】
図５は、実施例１に係るキャッシュ状態管理情報のデータ構造の一例を示す図である。
【００５９】
キャッシュ状態管理情報２４５は、データアドレス毎のエントリを格納する。キャッシ

ュ状態管理情報２４５のエントリは、データアドレス５０１、第一転送先コントローラＩＤ５０２、第二転送先コントローラＩＤ５０３、及びキャッシュ状態５０４のフィールドを含む。

【００６０】

データアドレス５０１には、エントリに対応するストレージシステム２内のユーザデータの格納位置を示す値（データアドレス）が格納される。

【００６１】

第一転送先コントローラＩＤ５０２には、エントリに対応するデータアドレスのデータが二重化されてキャッシュされている、第一転送の転送先のキャッシュ領域２４３を有するコントローラ２２（転送先コントローラの一例）の識別子（コントローラＩＤ：第一転送先コントローラＩＤ）が格納される。

【００６２】

第二転送先コントローラＩＤ５０３には、エントリに対応するデータアドレスのデータが二重化されてキャッシュされている、第二転送の転送先のキャッシュ領域２４３を有するコントローラ２２（担当コントローラの一例）の識別子（コントローラＩＤ：第二転送先コントローラＩＤ）が格納される。本実施形態では、第二転送先コントローラＩＤ５０３には、エントリに対応するデータアドレスのデータが属する論理ユニットのオーナ権を有するコントローラ（オーナコントローラ）２２のコントローラＩＤが格納される。

【００６３】

キャッシュ状態５０４には、エントリに対するデータアドレスのデータのキャッシュの状態を示す情報が格納される。キャッシュの状態としては、記憶デバイスユニット２０にデステージされていないことを示すダーティと、デステージされていることを示すクリーンとがある。

【００６４】

次に、転送管理情報２４６を詳細に説明する。

【００６５】

図６は、実施例１に係る転送管理情報のデータ構造の一例を示す図である。

【００６６】

転送管理情報２４６は、データアドレス毎のエントリを格納する。転送管理情報２４６のエントリは、データアドレス６０１、コントローラＩＤ６０２、及び転送状態管理情報アドレス６０３のフィールドを含む。データアドレス６０１には、エントリに対応するストレージシステム２内のユーザデータの格納位置（記憶空間）を示す値（データアドレス）が格納される。コントローラＩＤ６０２には、エントリに対応するデータアドレスのライトデータをホスト１から受信したコントローラ（受信コントローラ）２２の識別情報（コントローラＩＤ）が格納される。転送状態管理情報アドレス６０３には、エントリに対応するデータアドレスの転送状態管理情報２４７における対応するエントリの格納場所を示す値（アドレス）が格納される。

【００６７】

次に、転送状態管理情報２４７を詳細に説明する。

【００６８】

図７は、実施例１に係る転送状態管理情報のデータ構造の一例を示す図である。

【００６９】

転送状態管理情報２４７は、データアドレス毎のエントリを格納する。転送状態管理情報２４７のエントリは、データアドレス７０１、第一転送データ受領済フラグ７０２、第一転送完了フラグ７０３、及び第二転送完了フラグ７０４のフィールドを含む。

【００７０】

データアドレス７０１は、エントリに対応するストレージシステム２内のユーザデータの格納位置を示す値（データアドレス）が格納される。第一転送データ受領済フラグ７０２には、ＨＣＡ２５０によって、データアドレス７０１のデータアドレスに対応するライトデータの第一転送のデータが受領されたか否かを示す値（受領済みフラグ）が格納され

る。受領済みフラグは、受領された場合には、「１」が設定され、受領されていない場合には、「０」が設定される。第一転送完了フラグ７０３には、ＨＣＡ２５０によって、データアドレス７０１に対応するデータアドレスのライトデータの第一転送が完了したか否かを示す値（第一転送完了フラグ）が格納される。第一転送完了フラグは、第一転送が完了された場合には、「１」が設定され、第一転送が完了されていない場合には、「０」が設定される。第二転送完了フラグ７０４には、ＨＣＡ２５０によって、データアドレス７０１に対応するデータアドレスのライトデータの第二転送が完了したか否かを示す値（第二転送完了フラグ）が格納される。第二転送完了フラグは、第二転送が完了された場合には、「１」が設定され、第二転送が完了されていない場合には、「０」が設定される。

【００７１】

次に、実施例１に係る計算機システムによる処理動作について説明する。

【００７２】

まず、逐次転送依頼処理について説明する。

【００７３】

図８は、実施例１に係る逐次転送依頼処理のフローチャートである。

【００７４】

逐次転送依頼処理は、ライト要求に対応するライトデータ（新データ）が対象とする記憶デバイスユニット２０における論理ユニット（記憶空間）のオーナ権を有するコントローラ２２（オーナコントローラ２２という。：担当コントローラの一例）がライト要求を受信した場合に実行される。ここで、オーナコントローラ２２にライト要求が送信される場合としては、ホスト計算機１から直接オーナコントローラ２２に送られる場合と、ライト要求に対応するライトデータに対応するキャッシュ領域２４３を有さず、ＦＥ－Ｉ／Ｆ２１０を介してホスト計算機１からライトデータを受領したコントローラ２２（ＦＥコントローラ２２という。受信コントローラの一例）からオーナコントローラ２２に転送される場合と、がある。

【００７５】

本例では、ＦＥコントローラ２２からオーナコントローラ２２にライト要求が転送された場合を例に説明する。

【００７６】

オーナコントローラ２２は、ライト要求を受信する（ステップＳ１０１）。次いで、オーナコントローラ２２は、キャッシュ状態管理情報２４５を参照し、ライト要求のデータアドレスに対応するエントリのキャッシュ状態５０４からキャッシュ状態を取得し（ステップＳ１０２）、キャッシュ状態がダーティであるか否かを判定する（ステップＳ１０３）。

【００７７】

この結果、ダーティでないと判定された場合（ステップＳ１０３：ＮＯ）には、キャッシュ領域２４３のデータ（旧データ）が既に記憶デバイスユニット２０に格納されていることを示すので、オーナコントローラ２２は、ライトデータを２つのコントローラ２２のキャッシュ領域２４３に同時に（並行して）転送し、処理を終了する（Ｓ１０６）。

【００７８】

一方、ダーティであると判定された場合（ステップＳ１０３：ＹＥＳ）には、オーナコントローラ２２は、ライト要求のデータアドレスに対応する転送状態管理情報２４７のエントリの格納先を示す値（転送状態管理情報アドレス）を取得し、転送管理情報２４６にエントリを追加する。オーナコントローラ２２は、追加したエントリのデータアドレス６０１、コントローラＩＤ６０２、及び転送状態管理情報アドレス６０３に、それぞれ、ライトデータのデータアドレス、ＦＥコントローラ２２のコントローラＩＤ、及び転送状態管理情報２４７のエントリの転送状態管理情報アドレスを設定する（ステップＳ１０４）。

【００７９】

次に、オーナコントローラ２２は、ライトデータの逐次転送をＦＥコントローラ２２内のＨＣＡ２５０に依頼し（ステップＳ１０５）、次の処理（図９の逐次転送完了待ち処理

）を実行する（Ｌ０）。なお、ＦＥコントローラ２２のＨＣＡ２５０への依頼は、自コントローラ２２のＨＣＡ２５０を経由して通知してもよい。

【００８０】

次に、逐次転送完了待ち処理について説明する。

【００８１】

図９は、実施例１に係る逐次転送完了待ち処理のフローチャートである。

【００８２】

オーナコントローラ２２は、逐次転送が完了しているか否かを判定する（ステップＳ２０１）。すなわち、オーナコントローラ２２は、転送状態管理情報２４７を参照し、ライトデータのデータアドレスに対応するエントリ、すなわち、データアドレス７０１の値がライトデータのデータアドレスであるエントリにおける第一転送完了フラグ７０３及び第二転送完了フラグ７０４のフラグが立っているか否か、すなわち、フラグの値が“１”であるか否かを判定する。なお、本ステップの処理は、一定の周期で行ってもよい。

【００８３】

この結果、逐次転送が完了していると判定した場合（ステップＳ２０１：ＹＥＳ）には、オーナコントローラ２２は、ＦＥコントローラ２２を経由してホスト１にライト処理が終了したことを意味するＧｏｏｄ応答を送信し（ステップＳ２０２）、処理を終了する。一方、逐次転送が完了していないと判定した場合（ステップＳ２０１：ＮＯ）には、オーナコントローラ２２は、コントローラ状態管理情報２４４から他のコントローラ２２の状態を取得し、状態が故障であるコントローラＩＤ（故障コントローラＩＤ）を特定する（ステップＳ２０３）。

【００８４】

次いで、オーナコントローラ２２は、第一転送の転送先コントローラ２２（第一転送先コントローラ２２）が故障しているか否かを判定する（ステップＳ２０４）。具体的には、オーナコントローラ２２は、ステップＳ２０３で特定した故障コントローラＩＤに、データアドレスに対応するキャッシュ状態管理情報２４５のエントリにおけるデータアドレス５０１に格納された第一転送先コントローラＩＤ５０２の第一転送先コントローラＩＤと一致するものが存在するか否かにより、第一転送先コントローラ２２が故障しているか否かを判定する。

【００８５】

この結果、第一転送先コントローラ２２が故障していると判定した場合（ステップＳ２０４：ＹＥＳ）には、オーナコントローラ２２は、第二転送が完了しているか否かを判定する（ステップＳ２０５）。すなわち、オーナコントローラ２２は、転送状態管理情報２４７を参照し、データアドレスに対応するエントリの第二転送完了フラグ７０４のフラグが立っているか否かを判定する。

【００８６】

この結果、第二転送が完了していると判定した場合（ステップＳ２０５：ＹＥＳ）には、第二転送により、第二転送先コントローラ（オーナコントローラ２２）のキャッシュ領域２４３に対してライトデータが格納されていることを意味しているので、オーナコントローラ２２は、オーナコントローラ２２のキャッシュ領域２４３に格納されているライトデータ（保証データ）をデステージ（記憶デバイスユニット２０に転送）する（ステップＳ２０７）。次に、オーナコントローラ２２は、ＦＥコントローラ２２を経由してホスト１に失敗応答を送信し、処理を終了する（ステップＳ２１１）。

【００８７】

ここで、第二転送が完了している場合（ステップＳ２０５：ＹＥＳ）には、第二転送の転送先のコントローラ２２（第二転送先コントローラ２２、オーナコントローラ）のキャッシュ領域２４３のライトデータは壊れていないことを示しているため、キャッシュ領域２４３のライトデータをデステージすることで、データの一貫性を保証できる。

【００８８】

一方、第二転送が完了していないと判定した場合（ステップＳ２０５：ＮＯ）には、オ

ーナコントローラ２２は、第一転送が完了しているか否かを判定する（ステップＳ２０６）。すなわち、オーナコントローラ２２は、転送状態管理情報２４７を参照し、データアドレスに対応するエントリの第一転送完了フラグ７０３のフラグが立っているか否か判定する。

【００８９】

この結果、第一転送が完了していると判定した場合（ステップＳ２０６：ＹＥＳ）には、オーナコントローラ２２は、処理をステップＳ２０１に進め、第二転送の完了を待つ。

【００９０】

一方、第一転送が完了していないと判定した場合（ステップＳ２０６：ＮＯ）、オーナコントローラ２２は、処理をステップＳ２０７に進める。

【００９１】

ここで、第一転送が完了していない場合、第二転送の開始前に第一転送先コントローラ２２が故障していることを意味しているので、第二転送先コントローラであるオーナコントローラ２２のキャッシュ領域２４３のダーティデータは更新されておらず、このキャッシュ領域２４３のダーティデータ（保証データ）をデステージすることで、データの一貫性を保証できる。

【００９２】

一方、ステップＳ２０４で、第一転送先コントローラ２２が故障していないと判定した場合（Ｓ２０４：ＮＯ）には、オーナコントローラ２２は、ＦＥコントローラ２２が故障しているか否かを判定する（ステップＳ２０８）。すなわち、オーナコントローラ２２は、転送管理情報２４６を参照し、データアドレスに対応するエントリのコントローラＩＤ６０２のコントローラＩＤを取得し、このコントローラＩＤと一致するものがステップＳ２０３で特定した故障コントローラＩＤに存在するか否かにより、ＦＥコントローラ２２が故障しているか否かを判定する。

【００９３】

この結果、ＦＥコントローラ２２が故障していないと判定した場合（ステップＳ２０８：ＮＯ）には、オーナコントローラ２２は、処理をステップＳ２０１に戻し、逐次転送の完了を待つ。

【００９４】

一方、ＦＥコントローラ２２が故障していると判定した場合（ステップＳ２０８：ＹＥＳ）には、オーナコントローラ２２は、第一転送が完了しているか否かを判定する（ステップＳ２０９）。すなわち、オーナコントローラ２２は、転送状態管理情報２４７を参照し、データアドレスに対応するエントリの第一転送完了フラグ７０３のフラグが立っているか否か判定する。なお、第一転送完了フラグ７０３のフラグで判定する代わりに、第一転送データ受領済フラグ７０２のフラグが立っているか否かを判定してもよい。また、ステップＳ２０９の前に、第二転送が完了しているか否かを判定し、第二転送が完了している場合、ホスト１にＧｏｏｄ応答を送信し、処理を終了してもよい。

【００９５】

この結果、第一転送が完了していると判定した場合（ステップＳ２０９：ＹＥＳ）、オーナコントローラ２２は、第一転送先のコントローラ２２にそのコントローラ２２のキャッシュ領域２４３からのデータ（保証データ）のデステージを依頼し（ステップＳ２１０）、処理をステップ２１１に進める。なお、第一転送先のコントローラ２２は、依頼に対応して、キャッシュ領域２４３のデータをデステージすることとなる。ここで、第一転送が完了している場合、第一転送先コントローラ２２のキャッシュ領域２４３のライトデータは壊れていないため、このキャッシュ領域２４３のライトデータがデステージされることによりデータの一貫性を保証できる。

【００９６】

一方、第一転送が完了していないと判定した場合（ステップＳ２０９：ＮＯ）には、オーナコントローラ２２は、処理をステップＳ２０７に進め、自コントローラ２２のキャッシュ領域２４３のライトデータ（保証データ）をデステージする。ここで、第一転送が完

了していない場合、第二転送の開始前にＦＥコントローラ２２が故障しているため、オーナコントローラ２２のキャッシュ領域２４３のダーティデータは更新されておらず、このキャッシュ領域２４３のダーティデータをデステージすることでデータの一貫性を保証できる。

【００９７】

次に、逐次転送処理について説明する。

【００９８】

図１０は、実施例１に係る逐次転送処理のフローチャートである。

【００９９】

ＦＥコントローラ２２（具体的には、ＦＥコントローラ２２のＨＣＡ２５０）は、オーナコントローラ２２から送信された逐次転送依頼を受信し、逐次転送依頼からライトデータのデータアドレスを取得する（ステップＳ３０１）。次に、ＦＥコントローラ２２は、キャッシュ状態管理情報２４５を参照し、データアドレスに対応するエントリの第一転送先コントローラＩＤ５０２及び第二転送先コントローラＩＤ５０３から、第一転送先コントローラＩＤ及び第二転送先コントローラＩＤを取得する（ステップＳ３０２）。

【０１００】

次に、ＦＥコントローラ２２のＨＣＡ２５０は、第一転送を実行する（ステップＳ３０３）。具体的には、ＦＥコントローラ２２のＨＣＡ２５０は、ライトデータをバッファ領域２４２から取り出し、第一転送先コントローラ２２のＨＣＡ２５０を介して、ライトデータをキャッシュ領域２４３に転送する（ステップＳ３０３）。この際、ライトデータは、第一転送先コントローラ２２のバッファ領域２４０を経由することなく、また、第一転送先コントローラ２２のＣＰＵ２３０の関与なしに、キャッシュ領域２４３に転送される。

【０１０１】

次に、ＦＥコントローラ２２のＨＣＡ２５０は、ライトデータの転送が成功したか否かを判定する（ステップＳ３０４）。

【０１０２】

この結果、転送が失敗したと判定した場合（ステップＳ３０４：ＮＯ）には、ＦＥコントローラ２２のＨＣＡ２５０は、逐次転送処理を終了する。

【０１０３】

一方、転送が成功したと判定した場合（ステップＳ３０４：ＹＥＳ）には、ＦＥコントローラ２２のＨＣＡ２５０は、第一転送先コントローラ２２のメモリ２４０内に存在する転送状態管理情報２４７のデータアドレスに対応するエントリの第一転送データ受領済フラグ７０２のフラグを立てる、すなわち、フラグを１に設定する（ステップＳ３０５）。

【０１０４】

次に、ＦＥコントローラ２２のＨＣＡ２５０は、第一転送先コントローラ２２のＨＣＡ２５０に第二転送先コントローラ２２のメモリ２４０内に存在する転送状態管理情報２４７のデータアドレスに対応するエントリの第一転送完了フラグ７０３のフラグを立てることを指示する（ステップＳ３０６）。

【０１０５】

次に、ＦＥコントローラ２２のＨＣＡ２５０は、第二転送を実行する（ステップＳ３０７）。具体的には、ＦＥコントローラ２２のＨＣＡ２５０は、ライトデータをバッファ領域２４２から取り出し、第二転送先コントローラ２２のＨＣＡ２５０を介して、ライトデータをキャッシュ領域２４３に転送する（ステップＳ３０７）。

【０１０６】

次に、ＦＥコントローラ２２のＨＣＡ２５０は、ライトデータの転送が成功したか否かを判定する（ステップＳ３０８）。

【０１０７】

この結果、転送が失敗したと判定した場合（ステップＳ３０８：ＮＯ）には、ＦＥコントローラ２２は、逐次転送処理を終了する。

【０１０８】

一方、転送が成功したと判定した場合（ステップＳ３０８：ＹＥＳ）には、ＦＥコントローラ２２のＨＣＡ２５０は、第二転送先コントローラ２２のメモリ２４０内に存在する転送状態管理情報２４７のデータアドレスに対応するエントリの第二転送完了フラグ７０４のフラグを立てる、すなわち、フラグを１に設定し（ステップＳ３０９）、処理を終了する。

【０１０９】

次に、障害対応処理について説明する。

【０１１０】

図１１は、実施例１に係る障害対応処理のフローチャートである。障害対応処理は、オーナコントローラ２２以外のコントローラ２２により実行される処理である。障害対応処理は、一定時間ごとに一度実行されてもよく、コントローラ２２により障害が検知された場合に実行されてもよい。

【０１１１】

コントローラ２２は、コントローラ状態管理情報２４４から他のコントローラ２２の状態を取得し、状態が故障であるコントローラＩＤ４０１（故障コントローラＩＤ）を特定する（ステップＳ４０１）。

【０１１２】

次に、コントローラ２２は、オーナコントローラ２２（第二転送先コントローラ２２）が故障しているか否かを判定する（ステップＳ４０２）。すなわち、オーナコントローラ２２は、障害コントローラＩＤに、データアドレスに対応するキャッシュ状態管理情報２４５のエントリにおける第二転送先コントローラＩＤ５０３のコントローラＩＤと一致するものが存在しているか否かにより、オーナコントローラ２２が故障しているか否かを判定する。

【０１１３】

この結果、オーナコントローラ２２が故障していないと判定した場合（ステップＳ４０２：ＮＯ）には、コントローラ２２は、処理をステップＳ４０１に戻す。一方、オーナコントローラ２２が故障していると判定した場合（ステップＳ４０２：ＹＥＳ）には、コントローラ２２は、自身（自コントローラ）が第一転送先コントローラであるか否かを判定する（ステップＳ４０３）。すなわち、コントローラ２２は、自身のコントローラＩＤ（自コントローラＩＤ）と、データアドレスに対応するキャッシュ状態管理情報２４５のエントリにおける第一転送先コントローラＩＤ５０３のコントローラＩＤとが同一であるか判定する。なお、上記処理の代わりに、自コントローラがＦＥコントローラであるか否かを判定し、すなわち、自コントローラＩＤと、転送管理情報２４６のデータアドレスに対応するエントリのコントローラＩＤ６０２のコントローラＩＤとが同一であるか否かを判定し、自コントローラがＦＥコントローラである場合に、以降の処理を行ってもよい。

【０１１４】

ステップＳ４０３の判定の結果、自コントローラが第一転送先コントローラでないと判定した場合（ステップＳ４０３：ＮＯ）には、コントローラ２２は、処理をステップＳ４０１に進める。

【０１１５】

一方、自コントローラが第一転送先コントローラであると判定した場合（ステップＳ４０３：ＹＥＳ）には、コントローラ２２は、第一転送が完了しているか否かを判定する（ステップＳ４０４）。すなわち、コントローラ２２は、転送管理情報２４６のデータアドレスに対応するエントリの転送状態管理情報アドレス６０３のアドレスを用いて、転送状態管理情報２４７のエントリを参照し、このエントリの第一転送データ受領済フラグ７０２のフラグが立っているか否かを判定する。

【０１１６】

この判定結果、第一転送が完了していないと判定した場合（ステップＳ４０４：ＮＯ）には、コントローラ２２は、処理をステップＳ４０１に進め、第一転送の完了を待つ。

【０１１７】

一方、第一転送が完了していると判定した場合（ステップＳ４０４：ＹＥＳ）には、コントローラ２２は、キャッシュ領域２４３のライトデータ（保証データ）をデステージし（ステップＳ４０５）、ＦＥコントローラ２２を経由してホスト１に失敗応答を送信し（ステップＳ４０６）、処理を終了する。ここで、第一転送が完了している場合（ステップＳ４０４：ＹＥＳ）、第一転送先コントローラ２２のキャッシュ領域２４３のライトデータは壊れていないため、キャッシュ領域２４３のライトデータをデステージすることでデータの一貫性を保証できる。

【０１１８】

以上説明したように、上記実施例に係る計算機システムでは、ライトデータの二重化の処理の進捗に合わせて、障害発生時に記憶デバイスユニット２０に書き込むキャッシュ領域２４３を使い分けることで、ライトデータの一貫性を保証できる。

【実施例２】

【０１１９】

次に、実施例２に係る計算機システムについて説明する。

【０１２０】

実施例２に係る計算機システムは、図３に示す実施例１に係る計算機システムにおいて、論理ユニットを担当するコントローラ２２を特定のコントローラ２２に限定しない、すなわち、論理ユニットのオーナ権を設定しないようにしたシステムである。この計算機システムにおいては、例えば、ホスト１からのライト要求を受信したコントローラ（受信コントローラ）が担当コントローラとなる。

【０１２１】

実施例２に係る計算機システムでは、図８に示す逐次転送依頼処理を、ホスト１からライト要求を受信したコントローラ２２（ＦＥコントローラ２２）が実行する。

【０１２２】

次に、逐次転送完了待ち処理について説明する。

【０１２３】

図１２は、実施例２に係る逐次転送完了待ち処理のフローチャートである。

【０１２４】

ＦＥコントローラ２２は、逐次転送が完了しているか否かを判定する（ステップＳ５０１）。すなわち、ＦＥコントローラ２２は、転送状態管理情報２４７を参照し、ライトデータのデータアドレスに対応するエントリ、すなわち、データアドレス７０１の値がライトデータのデータアドレスであるエントリにおける第一転送完了フラグ７０３及び第二転送完了フラグ７０４のフラグが立っているか否か、すなわち、フラグの値が“１”であるか否かを判定する。

【０１２５】

この結果、逐次転送が完了していると判定した場合（ステップＳ５０１：ＹＥＳ）には、ＦＥコントローラ２２は、ホスト１にＧｏｏｄ応答を送信し（ステップＳ５０２）、処理を終了する。一方、逐次転送が完了していないと判定した場合（ステップＳ５０１：ＮＯ）には、ＦＥコントローラ２２は、コントローラ状態管理情報２４４から他のコントローラ２２の状態を取得し、状態が故障であるコントローラＩＤ（故障コントローラＩＤ）を特定する（ステップＳ５０３）。

【０１２６】

次いで、オーナコントローラ２２は、第一転送の転送先コントローラ２２（第一転送先コントローラ２２）が故障しているか否かを判定する（ステップＳ５０４）。

【０１２７】

この結果、第一転送先コントローラ２２が故障していると判定した場合（ステップＳ５０４：ＹＥＳ）には、ＦＥコントローラ２２は、第二転送が完了しているか否かを判定する（ステップＳ５０５）。すなわち、ＦＥコントローラ２２は、転送状態管理情報２４７を参照し、データアドレスに対応するエントリの第二転送完了フラグ７０４のフラグが立っているか否かを判定する。

【０１２８】
この結果、第二転送が完了していると判定した場合（ステップＳ５０５：ＹＥＳ）には、ＦＥコントローラ２２は、第二転送先のコントローラ２２のキャッシュ領域２４３に格納されているライトデータ（保証データ）のデステージを依頼する（ステップＳ５０７）。次に、ＦＥコントローラ２２は、ホスト１に失敗応答を送信し、処理を終了する（ステップＳ５１１）。
【０１２９】
ここで、第二転送が完了している場合（ステップＳ５０５：ＹＥＳ）には、第二転送の転送先のコントローラ２２（第二転送先コントローラ２２）のキャッシュ領域２４３のライトデータは壊れていないことを示しているため、キャッシュ領域２４３のライトデータをデステージすることで、データの一貫性を保証できる。
【０１３０】
一方、第二転送が完了していないと判定した場合（ステップＳ５０５：ＮＯ）には、ＦＥコントローラ２２は、第一転送が完了しているか否かを判定する（ステップＳ５０６）。
【０１３１】
この結果、第一転送が完了していると判定した場合（ステップＳ５０６：ＹＥＳ）には、ＦＥコントローラ２２は、処理をステップＳ５０１に進め、第二転送の完了を待つ。
【０１３２】
一方、第一転送が完了していないと判定した場合（ステップＳ５０６：ＮＯ）、ＦＥコントローラ２２は、処理をステップＳ５０７に進める。
【０１３３】
ここで、第一転送が完了していない場合、第二転送の開始前に第一転送先コントローラ２２が故障していることを意味しているので、第二転送先コントローラのキャッシュ領域２４３のダーティデータは更新されておらず、このキャッシュ領域２４３のダーティデータをデステージすることで、データの一貫性を保証できる。
【０１３４】
一方、ステップＳ５０４で、第一転送先コントローラ２２が故障していないと判定した場合（Ｓ５０４：ＮＯ）には、ＦＥコントローラ２２は、第二転送先コントローラ２２が故障しているか否かを判定する（ステップＳ５０８）。すなわち、ＦＥコントローラ２２は、キャッシュ状態管理情報２４５を参照し、データアドレスに対応するエントリの第二転送先コントローラＩＤ５０３のコントローラＩＤを取得し、このコントローラＩＤと一致するものがステップＳ５０３で特定した故障コントローラＩＤに存在するか否かにより、第二転送先コントローラ２２が故障しているか否かを判定する。
【０１３５】
この結果、第二転送先コントローラ２２が故障していないと判定した場合（ステップＳ５０８：ＮＯ）には、ＦＥコントローラ２２は処理をステップＳ５０１に戻し、逐次転送の完了を待つ。
【０１３６】
一方、第二転送先コントローラ２２が故障していると判定した場合（ステップＳ５０８：ＹＥＳ）には、ＦＥコントローラ２２は、第一転送が完了しているか否かを判定する（ステップＳ５０９）。
【０１３７】
この結果、第一転送が完了していないと判定した場合（ステップＳ５０９：ＮＯ）には、ＦＥコントローラ２２は、処理をステップＳ５０１に進め、第一転送が終わるのを待つ。
【０１３８】
一方、第一転送が完了していると判定した場合（ステップＳ５０９：ＹＥＳ）、ＦＥコントローラ２２は、第一転送先のコントローラ２２にそのコントローラ２２のキャッシュ領域２４３からのデータ（保証データ）のデステージを依頼し（ステップＳ５１０）、処理をステップＳ５１１に進める。ここで、第一転送が完了している場合、第一転送先コントローラ２２のキャッシュ領域２４３のライトデータは壊れていないため、このキャッシ

ュ領域２４３のライトデータをデステージすることでデータの一貫性を保証できる。
【０１３９】
次に、逐次転送処理について説明する。
【０１４０】
実施例２に係る計算機システムの逐次転送処理は、図１０に示す逐次転送処理とは、ステップＳ３０６とステップＳ３０９における処理内容が異なる。
【０１４１】
実施例２に係る計算機システムにおいては、ステップＳ３０６では、コントローラ２２のＨＣＡ２５０は、自コントローラ２２のメモリ２４０内に存在する、データアドレスに対応する転送状態管理情報２４７のエントリの第一転送完了フラグ７０３のフラグを立てる。また、ステップＳ３０９では、コントローラ２２のＨＣＡ２５０は、自コントローラ２２のメモリ２４０内に存在する、データアドレスに対応する転送状態管理情報２４７のエントリの第二転送完了フラグ７０４のフラグを立てる。
【０１４２】
次に、障害対応処理について説明する。
【０１４３】
図１３は、実施例２に係る障害対応処理のフローチャートである。
【０１４４】
障害対応処理は、ＦＥコントローラ２２以外のコントローラ（他コントローラ）が実行する処理である。障害対応処理は、一定時間に一度実施してもよいし、他コントローラ２２の障害検知時に実施してもよい。
【０１４５】
コントローラ２２は、コントローラ状態管理情報２４４から他のコントローラ２２の状態を取得し、状態が故障であるコントローラＩＤ（故障コントローラＩＤ）を特定する（ステップＳ６０１）。次に、コントローラ２２は、ＦＥコントローラ２２が故障しているか否かを判定する（ステップＳ６０２）。
【０１４６】
この結果、ＦＥコントローラ２２が故障していないと判定した場合（ステップＳ６０２：ＮＯ）には、コントローラ２２は、処理をステップＳ６０１に戻す。一方、ＦＥコントローラ２２が故障していると判定した場合（ステップＳ６０２：ＹＥＳ）には、コントローラ２２は、自身（自コントローラ）が第一転送先コントローラであるか否かを判定する（ステップＳ６０３）。なお、上記処理の代わりに、自コントローラが第二転送先コントローラであるか否かを判定し、以降の処理を行ってもよい。
【０１４７】
ステップＳ６０３の判定の結果、自コントローラ２２が第一転送先コントローラでないと判定した場合（ステップＳ６０３：ＮＯ）には、コントローラ２２は、処理をステップＳ６０１に進める。
【０１４８】
一方、自コントローラが第一転送先コントローラであると判定した場合（Ｓ６０３：ＹＥＳ）には、コントローラ２２は、第一転送が完了しているか否かを判定する（ステップＳ６０４）。
【０１４９】
この判定結果、第一転送が完了していると判定した場合（ステップＳ６０４：ＹＥＳ）には、コントローラ２２は、キャッシュ領域２４３のライトデータ（保証データ）をデステージし（ステップＳ６０６）、ＦＥコントローラ２２を経由してホスト１に失敗応答を送信し（ステップＳ６０７）、処理を終了する。ここで、第一転送が完了している場合（ステップＳ６０４：ＹＥＳ）、第一転送先コントローラ２２のキャッシュ領域２４３のライトデータは壊れていないため、キャッシュ領域２４３のライトデータをデステージすることでデータの一貫性を保証できる。
【０１５０】

一方、第一転送が完了していないと判定した場合（ステップＳ６０４：ＮＯ）には、コントローラ２２は、第二転送先のコントローラ２２にキャッシュ領域２４３に格納されているライトデータ（保証データ）のデステージを依頼し（ステップＳ６０５）、処理をステップＳ６０７に進める。ここで、第一転送が完了していない場合（ステップＳ６０４：ＮＯ）、第二転送の開始前に第一転送先コントローラ２２が故障しているため、第二転送先コントローラ２２のキャッシュ領域２４３のダーティデータは更新されておらず、このキャッシュ領域２４３のダーティデータをデステージすることでデータの一貫性を保証できる。

【０１５１】

以上説明したように、上記実施例に係る計算機システムでは、ライトデータの二重化の処理の進捗に合わせて、障害発生時に記憶デバイスユニット２０に書き込むキャッシュ領域２４３を使い分けることで、ライトデータの一貫性を保証できる。

【０１５２】

なお、本発明は上記した実施例に限定されるものではなく、様々な変形例が含まれる。また、例えば、上記した実施例は本発明を分かりやすく説明するために構成を詳細に説明したものであり、必ずしも説明した全ての構成を備えるものに限定されるものではない。また、各実施例の構成の一部について、他の構成に追加、削除、置換することが可能である。

【０１５３】

また、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する等によりハードウェアで実現してもよい。また、本発明は、実施例の機能を実現するソフトウェア（データ管理プログラム）のプログラムコードによって実現してもよい。この場合、プログラムコードを記録した記憶媒体をコンピュータに提供し、そのコンピュータが備えるプロセッサが記憶媒体に格納されたプログラムコードを読み出す。この場合、記憶媒体から読み出されたプログラムコード自体が前述した実施例の機能を実現することになり、そのプログラムコード自体、及びそれを記憶した記憶媒体は本発明を構成することになる。このようなプログラムコードを供給するための記憶媒体としては、例えば、フレキシブルディスク、ＣＤ－ＲＯＭ、ＤＶＤ－ＲＯＭ、ハードディスク、ＳＳＤ（Ｓｏｌｉｄ　Ｓｔａｔｅ　Ｄｒｉｖｅ）、光ディスク、光磁気ディスク、ＣＤ－Ｒ、磁気テープ、不揮発性のメモリカード、ＲＯＭなどがある。

【０１５４】

また、本実施例に記載の機能を実現するプログラムコードは、例えば、アセンブラ、Ｃ／Ｃ＋＋、ｐｅｒｌ、Ｓｈｅｌｌ、ＰＨＰ、Ｊａｖａ（登録商標）等の広範囲のプログラム又はスクリプト言語で実装してもよい。

【０１５５】

さらに、実施例の機能を実現するソフトウェアのプログラムコードを、ネットワークを介して配信することによって、それをコンピュータのハードディスクやメモリ等の記憶手段又はＣＤ－ＲＷ、ＣＤ－Ｒ等の記憶媒体に格納し、コンピュータが備えるプロセッサが当該記憶手段や当該記憶媒体に格納されたプログラムコードを読み出して実行するようにしてもよい。

【０１５６】

上記実施例において、制御線や情報線は、説明上必要と考えられるものを示しており、製品上必ずしも全ての制御線や情報線を示しているとは限らない。全ての構成が相互に接続されていてもよい。

【０１５７】

また、上記実施例では、複数のコントローラ２２のキャッシュ領域２４３上での二重化ができない場合に、正常なデータを記憶デバイスユニット２０にデステージすることにより、データの一貫性を保証できるようにしていたが、本発明はこれに限られず、例えば、複数のコントローラ２２のキャッシュ領域２４３上での二重化ができない場合に、正常なデータを、正常な動作が可能なコントローラ２２のキャッシュ領域２４３にコピーして、

複数のコントローラ２２のキャッシュ領域２４３上で二重化させるようにしてもよい。

【０１５８】

また、上記実施形態では、ライトデータを複数のコントローラ２２のキャッシュ領域２４３上で二重化をさせるようにしていたが、本発明はこれに限られず、３以上の多重化をさせるようにしてもよく。

【符号の説明】

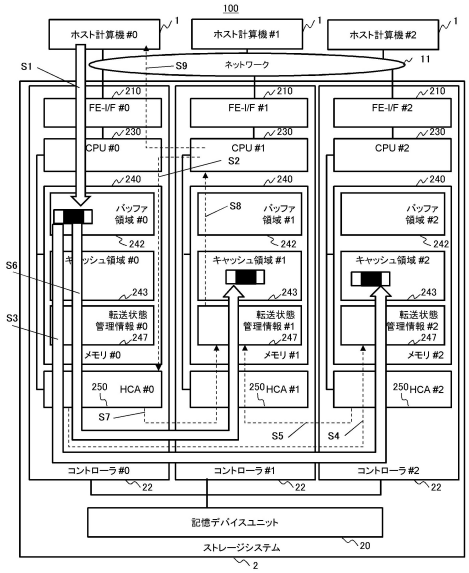
【０１５９】

１…ホスト計算機、２…ストレージシステム、１１…ネットワーク、２０…記憶デバイスユニット、２２…コントローラ、２３…ＨＣＡネットワーク、１００…計算機システム、２００…ＰＤＥＶ、２３０…ＣＰＵ、２４０…メモリ、２４３…キャッシュ領域、２４４…コントローラ状態管理情報、２４５…キャッシュ状態管理情報、２４６…転送管理情報、２４７…転送状態管理情報、２５０…ＨＣＡ

【図面】

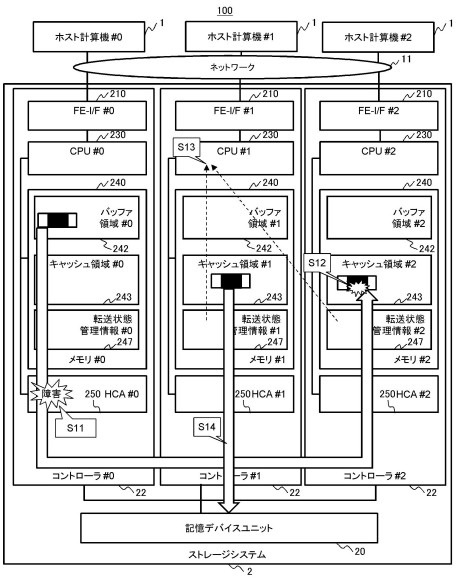
【図１】

図1



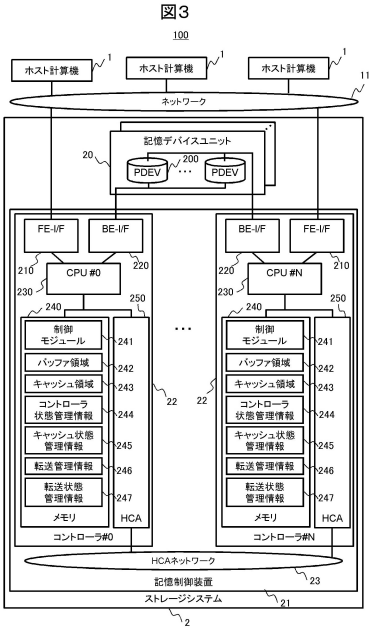

【図２】

図2




【図３】

図3




【図４】

図4

| 401 コントローラID | 402 状態 |
|---|---|
| 0 | 正常 |
| 1 | 故障 |
| 2 | 正常 |

244

【図５】

図5

| 501 データアドレス | 502 第一転送先コントローラID | 503 第二転送先コントローラID | 504 キャッシュ状態 |
|---|---|---|---|
| 0x1000 | 2 | 1 | ダーティ |
| 0x2000 | 0 | 1 | クリーン |
| 0x3000 | 2 | 0 | ダーティ |

245

【図６】

図6

| 601 データアドレス | 602 コントローラID | 603 転送状態管理情報アドレス |
|---|---|---|
| 0x1000 | 0 | 0x10000 |
| 0x2000 | 2 | 0x20000 |
| 0x3000 | 1 | 0x30000 |

246

【図７】

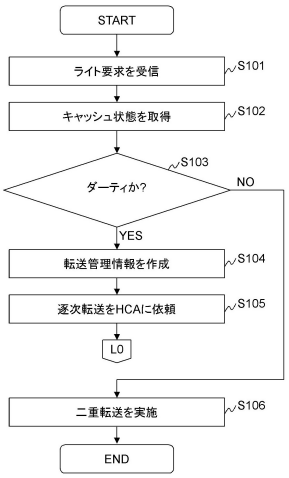
図7

| 701 データアドレス | 702 第一転送データ受領済フラグ | 703 第一転送完了フラグ | 704 第二転送完了フラグ |
|---|---|---|---|
| 0x1000 | 1 | 1 | 0 |
| 0x2000 | 1 | 1 | 1 |
| 0x3000 | 1 | 0 | 0 |

247

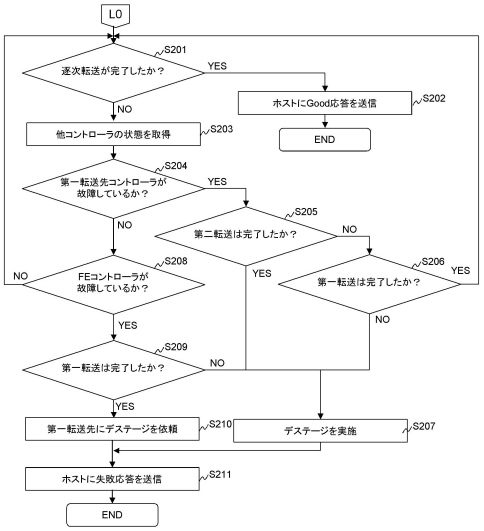
【図８】

図8



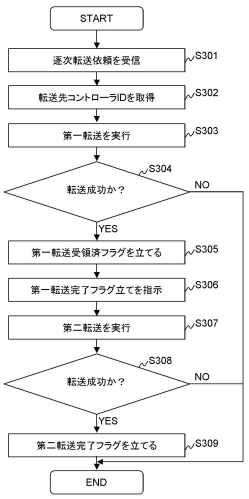

【図９】

図9



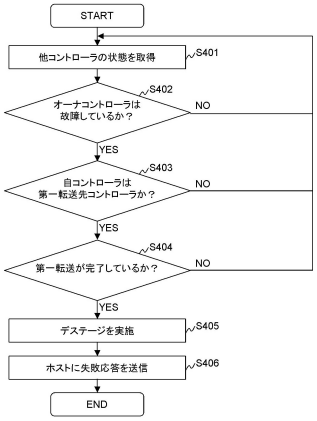

【図１０】

図10



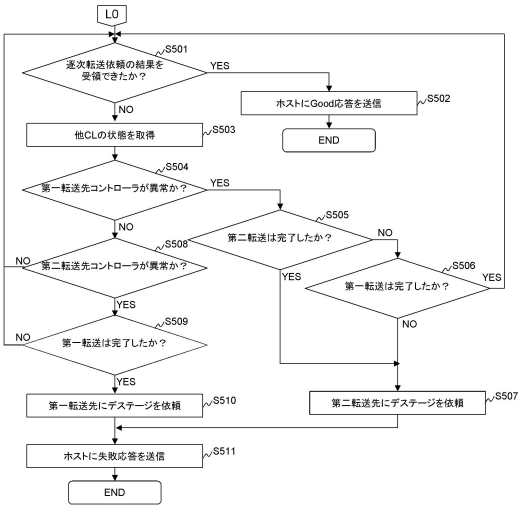
【図１１】

図11



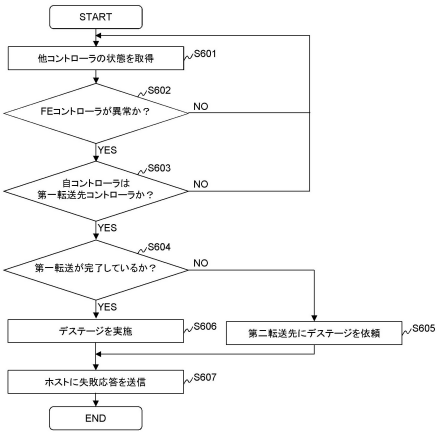

【図１２】

図12




【図１３】

図13



フロントページの続き

(51)国際特許分類　　ＦＩ
　　Ｇ０６Ｆ　　12/0868

(72)発明者　長尾 尚
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
審査官　田名網 忠雄

(56)参考文献　国際公開第２０１７／０５６２１９（WO，Ａ１）
国際公開第２０１５／０５２７９８（WO，Ａ１）

(58)調査した分野　(Int.Cl.，ＤＢ名)
Ｇ０６Ｆ　３／０６－３／０８
Ｇ０６Ｆ　１３／１０－１３／１４
Ｇ０６Ｆ　１１／２０
Ｇ０６Ｆ　１２／０８６６－１２／０８７３