

(12) 特許協力条約に基づいて公開された国際出願

(19) 世界知的所有権機関
国際事務局

(43) 国際公開日
2015年12月3日(03.12.2015)



(10) 国際公開番号
WO 2015/181902 A1

- (51) 国際特許分類:
G06F 3/06 (2006.01) **G06F 13/28** (2006.01)
G06F 12/00 (2006.01)
- (21) 国際出願番号: PCT/JP2014/064076
- (22) 国際出願日: 2014年5月28日(28.05.2014)
- (25) 国際出願の言語: 日本語
- (26) 国際公開の言語: 日本語
- (71) 出願人: 株式会社日立製作所 (HITACHI, LTD.) [JP/JP]; 〒1008280 東京都千代田区丸の内一丁目6番6号 Tokyo (JP).
- (72) 発明者: 横井 伸浩(YOKOI, Nobuhiro); 〒1008280 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 Tokyo (JP). 細谷 睦(HOSOYA, Mutsumi); 〒1008280 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 Tokyo (JP). 水島 永雅(MIZUSHIMA, Nagamasa); 〒1008280 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 Tokyo (JP). 吉井 義裕(YOSHII, Yoshihiro); 〒1008280 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 Tokyo (JP). 柴田 正文(SHIBATA, Masabumi); 〒1008280 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 Tokyo (JP).

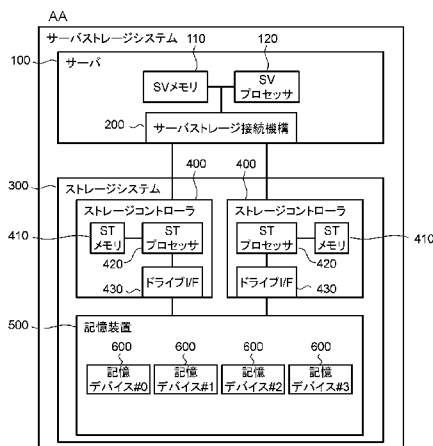
- (74) 代理人: 特許業務法人ウィルフォート国際特許事務所 (WILLFORT INTERNATIONAL PATENT FIRM); 〒1030016 東京都中央区日本橋小網町1-9-7 日本橋TCビル 1階 Tokyo (JP).
- (81) 指定国 (表示のない限り、全ての種類の国内保護が可能): AE, AG, AL, AM, AO, AT, AU, AZ, BA, BB, BG, BH, BN, BR, BW, BY, BZ, CA, CH, CL, CN, CO, CR, CU, CZ, DE, DK, DM, DO, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, GT, HN, HR, HU, ID, IL, IN, IR, IS, JP, KE, KG, KN, KP, KR, KZ, LA, LC, LK, LR, LS, LT, LU, LY, MA, MD, ME, MG, MK, MN, MW, MX, MY, MZ, NA, NG, NI, NO, NZ, OM, PA, PE, PG, PH, PL, PT, QA, RO, RS, RU, RW, SA, SC, SD, SE, SG, SK, SL, SM, ST, SV, SY, TH, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, ZA, ZM, ZW.
- (84) 指定国 (表示のない限り、全ての種類の広域保護が可能): ARIPO (BW, GH, GM, KE, LR, LS, MW, MZ, NA, RW, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, RU, TJ, TM), ヨーロッパ (AL, AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR,

[続葉有]

(54) Title: COMPUTER SYSTEM AND COMPUTER SYSTEM CONTROL METHOD

(54) 発明の名称: 計算機システムおよび計算機システム制御方法

Fig. 1



- 100 Server
110 SV memory
120 SV processor
200 Server storage connection mechanism
300 Storage system
400 Storage controller
410 ST memory
420 ST processor
430 Drive I/F
500 Storage device
600 Storage device
AA Server storage system

(57) Abstract: The present invention makes it possible to effectively reduce the amount of transfer data between server storages and implement an increase in the bandwidth of an effective band between server storages. When an interface device, which is located inside a server module, receives a read request issued by a server processor, the interface device transmits a read command based on the read request to a storage processor, and when an inverse transform instruction, which instructs the interface device to perform inverse transformation of transformed target data which is obtained by transforming target data in the read request, is received from the storage processor, the interface device performs DMA whereby transformed target data stored in a transfer source address in a storage memory is subjected to inverse transformation while being transferred to a transfer destination address in a server memory.

(57) 要約: 実効的にサーバストレージ間の転送データ量を削減し、サーバストレージ間の実効帯域の広帯域化を実現する。インタフェースデバイスは、サーバモジュール内に位置し、サーバプロセッサにより発行されたリードリクエストを受信すると、リードリクエストに基づくリードコマンドを、ストレージプロセッサへ送信し、ストレージプロセッサから、リードリクエストの対象データを変換して得られた変換後対象データの逆変換をインタフェースデバイスに行わせるとの逆変換指示を受けた場合に、ストレージメモリ上の転送元アドレスに格納された変換後対象データを逆変換しながらサーバメモリ上の転送先アドレスに転送するDMAを行う。



GB, GR, HR, HU, IE, IS, IT, LT, LU, LV, MC, MK, MT,
NL, NO, PL, PT, RO, RS, SE, SI, SK, SM, TR), OAPI
(BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, KM, ML,
MR, NE, SN, TD, TG).

添付公開書類:

— 国際調査報告 (条約第 21 条(3))

明 細 書

発明の名称： 計算機システムおよび計算機システム制御方法

技術分野

[0001] 本発明は、計算機システムおよび計算機システム制御方法に関する。

背景技術

[0002] 計算機システムで取り扱われるデータ量は近年ますます増加してきている。例えば、サーバからデータを日々収集してストレージ上でデータベースを構築し、所定のタイミングで、分析用サーバに蓄積したデータを読み出し、分析処理するような大規模データ分析システム技術が知られている。こうした計算機システムでは膨大な処理要求に対しての高速なレスポンスが求められる。

[0003] 一方で、サーバ及びストレージそれぞれの性能向上及び、フラッシュメモリ記憶装置のストレージ利用によるバックエンドの広帯域化に合わせて、サーバとストレージ間の接続帯域も、より広帯域化が必要となってきた。

[0004] 例えば、サーバ計算機とストレージ装置がS A N (Storage Area Network) を介して接続される計算機システムにおいて、例えばサーバ計算機に圧縮伸張機構を設けることで圧縮されたデータをS A Nの線路を介して転送する技術が知られている（例えば、特許文献1）。

先行技術文献

特許文献

[0005] 特許文献1：米国特許出願公開第2013／0332652号明細書

発明の概要

発明が解決しようとする課題

[0006] 特許文献1のシステムは、サーバ計算機とストレージ装置間をF C (Fibre Channel) で接続している。本構成でサーバ計算機とストレージ装置間の帯域向上を図るには、F Cのポート数増加や次世代品の利用などがあるが、高コストとなってしまう。

[0007] また、特許文献1のシステムはサーバ側にデータキャッシュ用のフラッシュメモリドライブを搭載し、リードデータのヒットミス判定を実施することでストレージアクセスの低減を可能とする。このため、フラッシュをサーバ側に準備し、サーバによるキャッシュ領域の管理が必要となる。更に、サーバ側での圧縮データを伸張せずに読み出すべきかについての判定を行うため、予めオブジェクト毎にポリシーを設定し、データが格納されている領域が圧縮されているかの情報をサーバ側で管理する必要がある。更に、キャッシュミスの場合にストレージデータをサーバ側メモリへの転送前に一旦フラッシュメモリに格納する処理、又、圧縮データの伸張処理等によるサーバ処理コストが大きく、結果としてバス帯域以外の伸張処理負荷が原因でシステム性能が低下する可能性がある。

[0008] よって、本願ではアプリケーション側での事前設計を不要とし、I/Oの特性に連動して必要に応じた伸張りード、非伸張りードの使い分けを実現し、これにより、実効的にサーバストレージ間の転送データ量を削減し、サーバストレージ間の実効帯域の広帯域化を実現することを目的とする。

課題を解決するための手段

[0009] 上記課題を解決するために、本発明の一態様である計算機システムは、ストレージプロセッサとストレージメモリと記憶デバイスとを有するストレージモジュールと、サーバプロセッサとサーバメモリとを有するサーバモジュールと、サーバプロセッサとストレージプロセッサとに接続されるインタフェースデバイスとを有する。インタフェースデバイスは、サーバモジュール内に位置し、サーバプロセッサにより発行されたリードリクエストを受信すると、リードリクエストに基づくリードコマンドを、ストレージプロセッサへ送信し、ストレージプロセッサから、リードリクエストの対象データを変換して得られた変換後対象データの逆変換をインタフェースデバイスに行わせるとの逆変換指示を受けた場合に、ストレージメモリ上の転送元アドレスに格納された変換後対象データを逆変換しながらサーバメモリ上の転送先アドレスに転送するDMA (Direct Memory Access) を行う。

図面の簡単な説明

- [0010] [図1]実施例のサーバストレージシステムの構成を示す。
- [図2]第一具体例のサーバストレージシステムの構成を示す。
- [図3]第二具体例のサーバストレージシステムの構成を示す。
- [図4]第三具体例のサーバストレージシステムの構成を示す。
- [図5]第四具体例のサーバストレージシステムの構成を示す。
- [図6]仮想サーバストレージシステムの構成を示す。
- [図7]変形例のサーバストレージシステムの構成を示す。
- [図8]S S接続機構の構成を示す。
- [図9]DMAコントローラの構成を示す。
- [図10]圧縮記憶デバイスの構成を示す。
- [図11]S Tメモリ内の記憶領域の構成を示す。
- [図12]物理デバイス内の記憶領域の構成を示す。
- [図13]ストレージシステム内の記憶領域の構成を示す。
- [図14]ストレージシステム内のページの構成を示す。
- [図15]仮想ボリューム管理テーブルを示す。
- [図16]ストレージプール管理テーブルを示す。
- [図17]RAIDグループ管理テーブルを示す。
- [図18]物理デバイス管理テーブルを示す。
- [図19]仮想ページ管理テーブルを示す。
- [図20]実ページ管理テーブルを示す。
- [図21]空き実ページ管理テーブルを示す。
- [図22]物理デバイス内の記憶領域の構成を示す。
- [図23]圧縮管理テーブルを示す。
- [図24]S Tコントローラが圧縮記憶デバイスから圧縮情報を取得する動作を示す。
- [図25]S Tコントローラが圧縮記憶デバイスから圧縮データを取得する第一の動作を示す。

[図26] S Tコントローラが圧縮記憶デバイスから圧縮データを取得する第二の動作を示す。

[図27] S Tコントローラが圧縮記憶デバイスから圧縮データを取得する第三の動作を示す。

[図28] 伸張DMA処理におけるS TコントローラとS S接続機構の間の動作を示す。

[図29] 圧縮情報SGLを示す。

[図30] データフォーマットを示す。

[図31] S Sシステムのリード処理の中の第一処理を示す。

[図32] S Sシステムのリード処理の中の第一処理に続く第二処理を示す。

[図33] S S接続機構による伸張DMA処理の中の第一処理を示す。

[図34] S S接続機構による伸張DMA処理の中の第一処理に続く第二処理を示す。

[図35] 伸張DMA処理中の保障コードチェック処理の中の第一処理を示す。

[図36] 伸張DMA処理中の保障コードチェック処理の中の第一処理に続く第二処理を示す。

[図37] 伸張DMA処理中の保障コードチェック処理の中の第二処理に続く第三処理を示す。

[図38] S Sシステムのライト処理の中の第一処理を示す。

[図39] S Sシステムのライト処理の中の第一処理に続く第二処理を示す。

[図40] S Sシステムのライト処理の中の第二処理に続く第三処理を示す。

[図41] S S接続機構による圧縮DMA処理の中の第一処理を示す。

[図42] S S接続機構による圧縮DMA処理の中の第一処理に続く第二の処理を示す。

[図43] 圧縮オフロードライト処理を示す。

[図44] サーバによる復号とS S接続機構による伸張を行う場合のリード処理の中の第一処理を示す。

[図45] サーバによる復号とS S接続機構による伸張を行う場合のリード処理

の中の第一処理の後の第二処理を示す。

[図46]サーバ・サーバ間接続の構成を示す。

[図47]ストレージ・ストレージ間接続の構成を示す。

発明を実施するための形態

- [0011] 以後の説明では「aaaテーブル」、「aaaリスト」、「aaaDB」、「aaaキュー」等の表現にて本発明の情報を説明するが、情報はテーブル、リスト、DB、キュー、等のデータ構造以外で表現されていてもよい。そのため、情報がデータ構造に依存しないことを示すために「aaaテーブル」、「aaaリスト」、「aaaDB」、「aaaキュー」等について「aaa情報」と呼ぶことがある。
- [0012] 更に、各情報の内容を説明する際に、「識別情報」、「識別子」、「名」、「名前」、「ID」という表現を用いるが、これらについてはお互いに置換が可能である。
- [0013] 以後の説明では「プログラム」を主語として説明を行う場合があるが、プログラムはプロセッサによって実行されることで定められた処理をメモリ及び通信ポート（通信制御デバイス）を用いながら行うため、プロセッサを主語とした説明としてもよい。また、プログラムを主語として開示された処理はストレージコントローラ等の計算機、情報処理装置が行う処理としてもよい。また、プログラムの一部又は全ては専用ハードウェアによって実現されてもよい。
- [0014] また、各種プログラムはプログラム配布サーバや、計算機が読み取り可能な記憶メディア（computer-readable memory media）によって各計算機にインストールされてもよい。この場合、プログラム配布サーバはプロセッサ（例えばCPU：Central Processing Unit）と記憶資源を含み、記憶資源は更に配布プログラムと配布対象であるプログラムを記憶している。そして、配布プログラムをCPUが実行することで、プログラム配布サーバのCPUは配布対象のプログラムを他の計算機に配布する。
- [0015] 以下、本発明の実施例について図面を用いて説明する。

[0016] 以下、本発明の実施例のサーバストレージ（ＳＳ）システムについて説明する。

[0017] 図１は、実施例のサーバストレージシステムの構成を示す。

[0018] このＳＳシステムは、サーバ１００と、ストレージシステム３００とを含む。サーバ１００とストレージシステム３００は、一つの筐体に格納されていても良い。ＳＳシステムを計算機システムと呼ぶことがある。

[0019] サーバ１００は、サーバ（ＳＶ）メモリ１１０と、サーバ（ＳＶ）プロセッサ１２０と、サーバストレージ（ＳＳ）接続機構２００とを含む。ＳＶメモリ１１０とＳＶプロセッサ１２０は、内部バスを介して互いに接続されている。ＳＶメモリ１１０は、メモリであり、アプリケーション、ＯＳ（Operating System）、ドライバ等、サーバ１００の制御のためのプログラム及びデータを格納する。ＳＶプロセッサ１２０は、マイクロプロセッサであり、ＳＶメモリ１１０内のプログラムに従って処理を実行する。ＳＳ接続機構２００は、サーバ側バスを介してＳＶプロセッサ１２０に接続され、ストレージ側バスを介してストレージシステム３００に接続され、ＳＶプロセッサ１２０とストレージシステム３００の間の通信を制御する。

[0020] ＳＶプロセッサ１２０とＳＳ接続機構２００の間は、サーバ側バスで接続されている。ＳＳ接続機構２００とＳＴプロセッサ４２０の間は、サーバ側バスと同一形式のバスであるストレージ側バスで接続されている。本実施例は、サーバ側バス及びストレージ側バスとして、ＰＣＩ Express（PCIe）を用いる。サーバ側バス及びストレージ側バスの夫々は、バックプレーン等の基板であっても良いし、ケーブルであっても良い。本システムでは、サーバ１００の内部バスと同じバスでサーバ１００からストレージシステム３００までを接続することでサーバ１００とストレージシステム３００の記憶領域間のＤＭＡ転送を可能とし、ＳＶプロセッサ１２０から発行した要求に対して、ストレージシステム３００内のストレージ（ＳＴ）プロセッサがＳＳ接続機構２００に搭載したＤＭＡエンジンを起動し、ストレージシステム３００内のＳＴメモリ上のデータをＳＶメモリ１１０上へ、または

、SVメモリ110上のデータをストレージシステム300内のSTメモリ上へ転送する。この構成により、サーバ100とストレージシステム300をいわゆるSAN等のネットワークスイッチのフォワード機能に頼らずに広帯域なバスで接続した一体型システムを構築し、帯域拡張とレイテンシ向上を実現する。さらに本システムにおいてDMA転送を実現することでI/O性能の向上を図ることができる。

[0021] なお、サーバ100は、圧縮伸張機能を有するフラッシュメモリ(FM)である圧縮伸張機能付FMを含んでも良い。サーバ100は、内部バスを介してSVプロセッサ120に接続され且つサーバ側バスを介してSS接続機構200に接続されたRoot Complex(RC)を含んでも良い。

[0022] SS接続機構200は、サーバ100からのライトデータを圧縮してストレージシステム300へ転送し、ストレージシステム300からのリードデータを伸張してサーバ100へ転送する。

[0023] ストレージシステム300は、ストレージ(ST)コントローラ400と、記憶装置500とを含む。記憶装置500は、サーバ100からアクセスされるデータを格納する。STコントローラ400は、記憶装置500とサーバ100に接続され、記憶装置500へのアクセスを制御する。STコントローラ400をコントローラと呼ぶことがある。

[0024] STコントローラ400は、ストレージ(ST)メモリ410と、ストレージ(ST)プロセッサ420と、ドライバインタフェース(I/F)430とを含む。STメモリ410とSTプロセッサ420とドライバI/F430は、内部バスを介して互いに接続されている。STメモリ410は、メモリであり、ストレージシステム300の制御のためのプログラム及びデータを格納する。STプロセッサ420は、マイクロプロセッサであり、ストレージ側バスを介してSS接続機構200に接続され、STメモリ410に格納されたプログラムに従って処理を実行する。ドライバI/F430は、記憶装置側バスを介して記憶装置500に接続されており、STプロセッサ420と記憶装置500の間の通信を制御する。

[0025] なお、S Tコントローラ400は、圧縮伸張機能付FMを含んでも良い。この場合、S Tプロセッサ420は、圧縮伸張機能付FMをキャッシュ領域として用い、キャッシュされたデータの圧縮及び伸張を行わせることができる。S Tコントローラ400は、内部バスを介してS Tプロセッサ420に接続され且つストレージ側バスを介してS S接続機構200に接続されたRCを含んでも良い。

[0026] 記憶装置500は、複数の記憶デバイス600（#0、#1、#2、#3）を含む。物理的な記憶デバイス600を物理デバイスと呼び、仮想的な記憶デバイスを仮想デバイスと呼ぶことがある。データの圧縮伸張機能を含む物理デバイスを圧縮記憶デバイス600aと呼び、データの圧縮伸張機能を含まない物理デバイスを非圧縮記憶デバイス600bと呼ぶことがある。

[0027] なお、データの圧縮及び復号は、圧縮記憶デバイス600aで行われても良いし、S Tコントローラ400により行われても良いし、サーバ100で行われても良い。

[0028] また、S Sシステムにおける圧縮を、重複排除（deduplication）など、他の変換に置き換えることができる。この変換により元データから変換結果データが生成される。変換結果データのサイズは、元データのサイズより小さくなる。また逆変換により変換結果データから元データが生成される。圧縮に対する逆変換は、伸張である。重複排除に対する逆変換は元データの復元である。

[0029] ここでは、サーバストレージシステムの構成の幾つかの具体例について説明する。

[0030] 図2は、第一具体例のサーバストレージシステムの構成を示す。

[0031] 各S S接続機構200は、複数のS Tコントローラ400に直接接続されていても良い。第一具体例のS Sシステムは、複数のサーバ100と、一つのストレージシステム300を含む。ストレージシステム300は、二つのS Tコントローラ400を含む。各S Tコントローラ400は、二つのS Tプロセッサ420と、各S Tプロセッサ420に接続されている複数のS T

メモリ 410 と、各 ST プロセッサ 420 に接続されている複数のドライブル／F 430 とを含む。一方の ST コントローラ 400 内の二つの ST プロセッサ 420 は、他方の ST コントローラ 400 内の二つの ST プロセッサ 420 に ST コントローラ 400 間バスを介して夫々接続されている。なお、ST プロセッサ 420 は複数でもよい。複数の場合も夫々他方の ST コントローラ 400 内の複数の ST プロセッサ 420 に ST コントローラ間バスを介して夫々接続されている。

[0032] 図 3 は、第二具体例のサーバストレージシステムの構成を示す。

[0033] 各 SS 接続機構 200 は、インタフェース拡張装置 700 を介して複数の ST コントローラ 400 に接続されていても良い。ここでは第一具体例に対する第二具体例の相違点について説明する。第二具体例の SS システムは、複数のサーバ 100 と、各 ST コントローラ 400 とに接続されたインタフェース拡張装置 700 を含む。インタフェース拡張装置 700 は、サーバ 100 と ST コントローラ 400 の間の接続パス数を増やす制御を行う。インタフェース拡張装置 700 により、より多くのサーバ 100 が一台のストレージに直結できる。インタフェース拡張装置 700 は例えば、PCIe スイッチである。

[0034] 図 4 は、第三具体例のサーバストレージシステムの構成を示す。

[0035] 各サーバ 100 は、複数の SS 接続機構 200 を含んでいても良い。ここでは第二具体例に対する第三具体例の相違点について説明する。第三具体例において、各サーバ 100 は、二つの SS 接続機構 200 を含む。二つの SS 接続機構 200 は、インタフェース拡張装置 700 を介して二つの ST コントローラ 400 に夫々接続される。

[0036] この構成において、ストレージ側バスを 2 本の PCIe x 4、サーバ側バスを 1 本の PCIe x 8 とすることにより、ストレージ側バスとサーバ側バスの帯域のバランスが取れるとする。ここで、ストレージ側バスを 1 本の PCIe x 4、サーバ側バスを 1 本の PCIe x 8 に変更することにより、ストレージ側バスの帯域だけを 1/2 に変更する。この場合、ストレージ側バ

スからSS接続機構200へ転送された圧縮データがSS接続機構200で伸張されてサーバ側バスへ転送される際に、ストレージ側バスにおいて1/2に圧縮されていたデータが、サーバ側バスにおいても元のサイズのデータに戻っても、ストレージ側バス及びサーバ側バスは、それぞれの帯域を確保できる。また、各サーバ100が2つのSS接続機構200を含むことでハードウェアの冗長性を確保できる。なお、各サーバ100が1つのSS接続機構200を含み、圧縮率を考慮してサーバ側バスの帯域をストレージ側バスより広帯域にしても良い。例えば、ストレージ側バスを2本のPCIe x 4とし、サーバ側バスを1本のPCIe x 16としても良い。更に、ストレージ側バスを2本のPCIe x 4とし、サーバ側バスを2本のPCIe x 8としても、同等の効果を得ることができる。このようにサーバ側バス及びストレージ側バスが、PCIeに従うことにより、圧縮に応じて必要な帯域を割り当てられることができる。

[0037] 図5は、第四具体例のサーバストレージシステムの構成を示す。

[0038] SSシステムは、複数のストレージシステム300を含んでも良い。ここでは第三具体例に対する第四具体例の相違点について説明する。第四具体例において、SSシステムは、二つのストレージシステム300を含む。一つのサーバ100内の二つのSS接続機構200は、インタフェース拡張装置700を介して二つのストレージシステム300に夫々接続される。一つのSS接続機構200は、インタフェース拡張装置700を介して一つのストレージシステム300内の二つのSTコントローラ400に接続される。この構成において、サーバ100に二つのSS接続機構200を搭載することで、より多くのストレージシステム300をサーバ100に接続することができ、サーバ100に対して広帯域、高性能なストレージリソースを提供できる。

[0039] 図6は、仮想サーバストレージシステムの構成を示す。

[0040] 物理的なSSシステム2000は、仮想化機構3000を実行し、仮想化機構3000上で仮想サーバストレージ(SS)システム1000を実行し

ても良い。仮想SSシステム1000は、仮想サーバ1100と、仮想ストレージシステム1300とを含む。この場合、SSシステム2000は少なくともプロセッサ2020、メモリ2010、記憶装置2500、SS接続機構2200を物理的に備える。仮想サーバ1100は、サーバ100と同様の機能を提供する。より具体的には、仮想サーバ1100は、SVメモリ110と同様の機能を提供する仮想SVメモリ2110と、SS接続機構200と同様の機能を提供する仮想SS接続機構1200とを含む。仮想ストレージシステム1300は、STコントローラ400と同様の機能を提供する仮想STコントローラ1400と、記憶装置500と同様の機能を提供する仮想記憶装置1500とを含む。仮想STコントローラ1400は、STメモリ410と同様の機能を提供する仮想STメモリ1410を含む。

[0041] 仮想化機構3000は、メモリ2010内の記憶領域を仮想SVメモリ1110に割り当て、SS接続機構2200の処理能力を仮想SS接続機構1200に割り当て、メモリ2010内の記憶領域を仮想STメモリ1410に割り当て、記憶装置2500内の記憶領域を仮想記憶装置1500に割り当てる。

[0042] 仮想SSシステム1000は、単独のサーバや、サーバとストレージシステム等、他の構成の計算機システム上で実行されても良い。また、計算機システムは、SS接続機構に代えて圧縮伸張回路を含んでも良い。

[0043] 図7は、変形例のサーバストレージシステムの構成を示す。

[0044] ここでは実施例のSSシステムに対する変形例のSSシステムの相違点について説明する。この変形例においてSSシステムは、ストレージシステム300の代わりにストレージシステム300aを有し、STコントローラ400の代わりにSTコントローラ400aを有する。サーバ100内のSS接続機構200に代えて、STコントローラ400a内にSS接続機構200が設けられる。SS接続機構200は、サーバ100内のSVプロセッサ120に接続され、STコントローラ400a内のSTプロセッサ420に接続される。サーバ100は更にFM140を含んでも良い。このFM

140が圧縮伸張機能を有し、SVプロセッサ120からの指示に応じて圧縮伸張を行っても良い。STコントローラ400aは更に、FM440を含んでも良い。このFM440が圧縮伸張機能を有し、STプロセッサ420からの指示に応じて、SS接続機構200の代わりに圧縮又は伸張を行っても良い。本構成のとき、ライトの動作は図38～40に開示したフローにおいて、SS接続機構200への送出前にFM140でデータが圧縮される。SS接続機構200はこの圧縮データを読み出して伸張し、更に、本実施例のストレージシステム300の圧縮フォーマットに圧縮しなおしながらSTメモリ410を介して記憶デバイス600に書き込む。なお、SS接続機構200は圧縮データを読み出したときに伸張せずに、更に、本実施例のストレージシステム300の圧縮フォーマットに圧縮しても良い。本構成によりライト帯域を広帯域化することができる。又、リード動作は図31, 32に開示したフローにおいてSTコントローラ400a内のFM440で圧縮されたデータまたは、記憶デバイス600に圧縮格納されたデータをSS接続機構200が読み出す場合に、データを伸張した上でサーバ100の圧縮フォーマットに圧縮しなおしながらサーバ100側のFM140またはSVメモリ110との間でDMA転送を行う。以上の構成により、SS接続機構200は、圧縮伸張機能を有したFM440と連携してシステムの広帯域化を実現する。なお、前述のようにSS接続機構200が伸張しないで圧縮データを読み出した場合、サーバ100の圧縮フォーマットに圧縮しなおさなくても良い。即ち、SS接続機構200は、サーバ100内に設けられても良いし、STコントローラ400内に設けられても良いし、サーバ100とSTコントローラ400の間に設けられても良い。

[0045] 図8は、SS接続機構200の構成を示す。

[0046] SS接続機構200は、例えば、ASIC (application specific integrated circuit) として実装される。SS接続機構200は、IF (Interface) プロセッサ210と、DMA (Direct Memory Access) コントローラ220と、内部メモリ230と、レジスタ240と、インタフェース250

と、二つのインタフェース260と、内部スイッチ270とを含む。DMAコントローラ220をDMAと呼ぶことがある。IFプロセッサ210と、DMAコントローラ220と、内部メモリ230と、レジスタ240と、インタフェース250と、二つのインタフェース260とは、内部スイッチ270を介して互いに接続されている。一方のインタフェース250は、サーバ側バスを介してSVプロセッサ120に接続されている。他方の二つのインタフェース260は、ストレージ側バスを介して二つのSTプロセッサ420に夫々接続されている。内部メモリ230は、メモリであり、プログラム及びデータを格納する。IFプロセッサ210は、マイクロプロセッサであり、内部メモリ230内のプログラムに従って処理を実行する。DMAコントローラ220は、IFプロセッサ210からの指示に従ってデータ転送（メモリ間転送）を実行する。レジスタ240は、IFプロセッサ210の設定値を格納する。内部スイッチ270は、IFプロセッサ210からの指示に従って各部の間の通信を制御する。なお、SV側インタフェース及びST側インタフェースは夫々複数あって、各バス及びインタフェース拡張を介して、SVプロセッサ120とSTプロセッサ420に接続されていても良い。

[0047] 図9は、DMAコントローラ220の構成を示す。

[0048] DMAコントローラ220は、圧縮デバイス221と、伸張デバイス222と、保障コード処理デバイス(0)223と、保障コード処理デバイス(1)224と、データ転送制御デバイス225と、二つのインタフェース226、227とを含む。二つのインタフェースは、内部スイッチに接続されている。圧縮デバイス221は、非圧縮データを圧縮して圧縮データに変換する。伸張デバイス222は、圧縮データを伸張して非圧縮データに変換する。保障コード処理デバイス(0)223は、非圧縮データの保障コードを算出し、非圧縮データに保障コードを付加する。更に、保障コード処理デバイス(0)223は、非圧縮データと保障コードに基づいて非圧縮データの誤りを検出する。保障コード処理デバイス(1)は、圧縮データの保障コー

ドを算出し、圧縮データに保障コードを付加する。更に、保障コード処理デバイス（１）２２４は、圧縮データと保障コードに基づいて圧縮データの誤りを検出する。また、夫々の保障コード処理デバイスは、保障コードの削除も行ふ。データ転送制御デバイス２２５は、データの転送を制御する。

[0049] ＳＳ接続機構２００がサーバ１００とＳＴコントローラ４００にＰＣＩｅで接続されることにより、サーバ１００内のＳＶメモリ１１０と、ＳＴコントローラ４００のＳＴメモリ４１０の間でＤＭＡ転送を行うことができ、ＤＭＡ転送時にデータの圧縮又は伸張を行うことができる。

[0050] 以下、圧縮記憶デバイス６００ａについて説明する。

[0051] 図１０は、圧縮記憶デバイス６００ａの構成を示す。

[0052] 圧縮記憶デバイス６００ａは、記憶デバイス（ＳＤ）コントローラ６１０と、ＳＤコントローラ６１０に接続されている複数のフラッシュメモリ（ＦＭ）６２０とを含む。ＳＤコントローラ６１０は、ＲＡＭ（Random Access Memory）６１１と、物理デバイス（ＳＤ）プロセッサ６１２と、保障コード処理デバイス６１３と、圧縮・伸張デバイス６１４と、Ｉ／Ｏインタフェース６１５と、ＦＭインタフェース６１６とを含む。ＳＤコントローラ６１０内の各部は内部バスを介して互いに接続されている。Ｉ／Ｏインタフェース６１５は、記憶装置側バスを介してＳＴコントローラ４００のドライブル／Ｆ４３０に接続されている。ＲＡＭ６１１は、例えばＤＲＡＭ（Dynamic RAM）であり、プログラム及びデータを格納する。ＳＤプロセッサ６１２は、マイクロプロセッサであり、ＲＡＭ６１１に格納されたプログラムに従って処理を実行する。保障コード処理デバイス６１３は、データに基づいて保障コードを生成し、データに保障コードを付加し、データと保障コードに基づいてデータのチェックをし、誤りを検出する。また、保障コードの削除も行ふ。圧縮・伸張デバイス６１４は、非圧縮データを圧縮して圧縮データに変換し、圧縮データを伸張して非圧縮データに変換する。ＦＭインタフェース６１６は、複数のＦＭ６２０に接続されている。この圧縮記憶デバイス６００ａが、ＳＴプロセッサ４２０からの指示に応じて、ＳＳ接続機構２００の

代わりに圧縮又は伸張を行っても良い。また、圧縮記憶デバイス600aは、データ入力に対して内部判断でデータを圧縮しても良く、STプロセッサ420から特に指示がない場合は内部で圧縮されているデータを伸張して出力しても良い。

[0053] 以下、ストレージシステム300内の記憶領域の構成について説明する。

[0054] 図11は、STメモリ410内の記憶領域の構成を示す。

[0055] STメモリ410内の記憶領域は、仮想ボリューム管理テーブル810と、ストレージプール管理テーブル820と、RAIDグループ管理テーブル830と、物理デバイス管理テーブル840と、仮想ページ管理テーブル850と、実ページ管理テーブル860と、空き実ページ管理テーブル870とを含む。これらのテーブルについては後述する。STメモリ410内の記憶領域は更に、ストレージ(ST)キャッシュ領域880と、ストレージ(ST)バッファ領域890とを含む。STキャッシュ領域880は、ライトデータやリードデータを一時的に保持し、ライトやリードアクセスに対する、データヒットに利用する。STバッファ領域890は、STコントローラ400により送受信されるデータや制御情報を格納する。

[0056] 図12は、圧縮記憶デバイス600a内の記憶領域の構成を示す。

[0057] 圧縮記憶デバイス600a内の記憶領域は、RAMにおける、メタデータ領域910と、キャッシュ領域930と、バッファ領域940と、FM620における、データ領域950とを含む。メタデータ領域910は、メタデータを格納し、圧縮管理テーブル920を含む。キャッシュ領域930は、一時的なデータを格納する。バッファ領域940は、圧縮記憶デバイス600aにより送受信されるデータや制御情報を格納する。データ領域950は、データを格納する。なお、SDコントローラ610内のRAM611内の情報は接続されているFM620内にあっても良い。

[0058] 図13は、ストレージシステム300内の記憶領域の構成を示す。

[0059] STコントローラ400は、物理デバイス内の記憶領域を仮想デバイスに割り当てる。物理デバイスは、圧縮記憶デバイス600a(圧縮SD)と非

圧縮記憶デバイス（非圧縮SD）の何れであっても良い。STコントローラ400は、複数の仮想デバイスを用いてパリティグループを作成する。パリティグループは、複数の圧縮記憶デバイス600aに夫々基づく複数の仮想デバイスを用いる圧縮パリティグループと、複数の非圧縮記憶デバイスに夫々基づく複数の仮想デバイスを用いる非圧縮パリティグループとの何れであっても良い。STコントローラ400は、ストレージプール管理テーブル820を用いて、パリティグループ内の記憶領域をストレージプールに割り当てる。STコントローラ400は、仮想ボリューム管理テーブル810を用いて、ストレージプール内の記憶領域を仮想ボリュームに割り当てる。ストレージプール毎に、圧縮データ格納用の圧縮ストレージプールと、圧縮されずに格納されるデータ用の非圧縮ストレージプールが存在してもよい。尚、圧縮ストレージプールは、例えば圧縮機能を有した機能デバイスの記憶領域で構成された圧縮パリティグループによって提供されていてもよいし、当該プールへの格納はSTコントローラ400やSTコントローラ400に接続される圧縮機構によって圧縮されるような構成としてもよい。また、仮想ボリュームも書き込まれたデータが圧縮されて記憶領域に格納される圧縮仮想ボリュームと書き込まれたデータが圧縮されことなく記憶領域に格納される非圧縮仮想ボリュームがあっても良く、混在していても良い。STコントローラ400は、仮想ボリュームをサーバ100へ提供する。

[0060] 図14は、ストレージシステム300内のページの構成を示す。

[0061] パリティグループは、複数の実ページを含む。仮想ボリュームは、複数の仮想ページを含む。STコントローラ400は、実ページ管理テーブル860を用いて、実ページを仮想ページに割り当てる。圧縮パリティグループを用いることにより、仮想ページに割り当てる実ページの合計サイズを物理デバイスの合計サイズより大きくすることができる。また、STコントローラ400は、圧縮パリティグループ内の実ページと非圧縮パリティグループ内の実ページとを切り替えて仮想ページに割り当てることができる。これにより、圧縮の効果の有無を切り替えることができる。また、圧縮パリティグル

ープを用いるときに、仮想ページに割り当てる実ページの合計サイズを物理デバイスの合計サイズと同じサイズにして、FM620の容量を増やさずに利用率を下げ、FM620を長寿命化することもできる。

[0062] 図15は、仮想ボリューム管理テーブル810を示す。

[0063] 仮想ボリューム管理テーブル810は、仮想ボリューム毎のエントリを含む。一つの仮想ボリュームに対応するエントリは、ボリューム番号（#）と、ボリューム容量と、割当済み容量と、プール番号（#）と、ボリューム属性とを含む。ボリューム番号は、当該仮想ボリュームを示す識別子である。ボリューム容量は、当該仮想ボリュームの容量である。割当済み容量は、当該仮想ボリューム内で、実ページが割り当てられている仮想ページの合計サイズである。プール番号は、当該仮想ボリュームに割り当てることができる実ページを含むストレージプールを示す識別子である。ボリューム属性は、当該仮想ボリュームにおける圧縮の有無を示す。

[0064] 図16は、ストレージプール管理テーブル820を示す。

[0065] ストレージプール管理テーブル820は、ストレージプール毎のエントリを含む。一つのストレージプールに対応するエントリは、プール番号（#）と、RAID（Redundant Arrays of Inexpensive Disks）グループリストと、プール属性と、空き仮想ページ数とを含む。プール番号は、当該ストレージプールを示す識別子である。RAIDグループリストは、当該ストレージプールに含まれるRAIDグループを示す識別子である。プール属性は、当該ストレージプールに用いられている物理デバイスの種類を示す。空き仮想ページ数は、当該ストレージプールの空き仮想ページの数を示す。尚、システム内に図13に示したように圧縮ストレージプールと非圧縮ストレージプールを設ける場合には、圧縮／非圧縮のプール属性を格納する欄があってもよい。

[0066] 図17は、RAIDグループ管理テーブル830を示す。

[0067] RAIDグループ管理テーブル830は、RAIDグループ毎のエントリを含む。一つのRAIDグループに対応するエントリは、RAIDグループ

番号（＃）と、ＲＡＩＤレベルと、物理デバイス番号（＃）とを含む。ＲＡＩＤグループ番号は、当該ＲＡＩＤグループを示す識別子である。ＲＡＩＤレベルは、当該ＲＡＩＤグループのＲＡＩＤレベルを示す。物理デバイス番号は、当該ＲＡＩＤグループに用いられている物理デバイスを示す識別子である。

[0068] 図１８は、物理デバイス管理テーブル８４０を示す。

[0069] 物理デバイス管理テーブル８４０は、物理デバイス毎のエントリを含む。一つの物理デバイスに対応するエントリは、仮想デバイス番号（＃）と、物理デバイス番号（＃）と、物理容量と、論理使用容量と、物理使用容量とを含む。仮想デバイス番号は、当該物理デバイスを割り当てられた仮想デバイスを示す識別子である。物理デバイス番号は、当該物理デバイスを示す識別子である。物理容量は、当該物理デバイスの容量である。論理使用容量は、当該仮想デバイスの論理記憶領域の使用量である。物理使用容量は、当該物理デバイスの物理記憶容量の使用量である。

[0070] 図１９は、仮想ページ管理テーブル８５０を示す。

[0071] 仮想ページ管理テーブル８５０は、仮想ページ毎に作成される。一つの仮想ページに対応する仮想ページ管理テーブル８５０は、仮想ボリューム番号（＃）と、実ページポインタと、ページ属性とを含む。仮想ボリューム番号は、当該仮想ページを含む仮想ボリュームを示す識別子である。実ページポインタは、当該仮想ページに割り当てられた実ページを指すポインタである。ページ属性は、当該実ページが圧縮データを格納するか非圧縮データを格納するかを示す。この図に示されている二つの仮想ページ管理テーブル８５０のうち、上の仮想ページ管理テーブル８５０は、実ページを割り当てられている仮想ページを示し、実ページポインタが実ページを指している。下の仮想ページ管理テーブル８５０は、実ページを割り当てられていない仮想ページを示し、実ページポインタがNULLを示している。

[0072] 図２０は、実ページ管理テーブル８６０を示す。

[0073] 実ページ管理テーブル８６０は、実ページ毎に作成される。一つの実ペー

ジに対応する実ページ管理テーブル 860 は、RAID グループ番号（#）と、実ページ先頭アドレスと、空きページポインタとを含む。RAID グループ番号は、当該実ページを含む RAID グループを示す識別子である。実ページ先頭アドレスは、当該 RAID グループにおける当該実ページの先頭アドレスを示す。空きページポインタは、当該実ページが空き実ページである場合に次の空き実ページの実ページ管理テーブル 860 を指す。

[0074] 図 21 は、空き実ページ管理テーブル 870 を示す。

[0075] 空き実ページ管理テーブル 870 は、空きページ管理ポインタを含む。この空きページ管理ポインタは、空きページの実ページ管理テーブル 860 を指す。連続する空きページの最後の空きページの実ページ管理テーブル 860 内の空きページポインタは、空きページ管理ポインタを指す。

[0076] 図 22 は、物理デバイス内の記憶領域の構成を示す。

[0077] 圧縮記憶デバイス 600a は、圧縮管理テーブル 920 を RAM 611 または FM 620 に格納する。圧縮記憶デバイス 600a の SD コントローラ 610 は、記憶領域のブロック単位の論理アドレスを ST コントローラ 400 へ提供し、圧縮管理テーブル 920 を用いて論理アドレスに対しブロック単位で実アドレスを割り当てる。ここで示すブロック単位はブロックストレージのデータ単位を示す。圧縮記憶デバイス 600a は、非圧縮データ換算で複数ブロックをまとめた単位（例えばブロックサイズ 0.5 kB を複数個まとめた 8 kB、以降圧縮単位）で圧縮可否を判断し、圧縮する場合にはこの単位で非圧縮データを圧縮単位毎に圧縮する。非圧縮記憶デバイスは、実アドレス管理テーブルを RAM に格納する。非圧縮記憶デバイスの SD コントローラ 610 は、ブロック毎の論理アドレスを ST コントローラ 400 へ提供し、実アドレス管理テーブルを用いて論理アドレスに対しブロックの実アドレスを割り当てる。

[0078] 図 23 は、圧縮管理テーブル 920 を示す。

[0079] 圧縮管理テーブル 920 は、記憶デバイス 600 毎に、記憶デバイス 600 内の RAM 611 または FM 620 に格納される。記憶デバイス 600 が

HDDの場合、圧縮管理テーブル920はHDDに格納される。別の方法として複数の記憶デバイス600の圧縮管理テーブル920をまとめて、ストレージコントローラのSTメモリ410、または、ストレージコントローラのフラッシュメモリに格納して管理しても良い。

[0080] 圧縮管理テーブル920は、圧縮記憶デバイス600a内の圧縮単位毎のエントリを含む。一つの圧縮単位に対応するエントリは、論理アドレスと、実アドレスと、圧縮フラグと、圧縮後ブロック数とを含む。論理アドレスは、当該圧縮データの先頭の論理アドレスを示す。実アドレスは、当該圧縮単位の先頭の実アドレスを示す。圧縮フラグは、当該圧縮単位のデータが圧縮されるか否かを示す。圧縮後ブロック数は、当該非圧縮データの圧縮後のデータについてのブロック数を示す。ここで、圧縮単位内のブロックがすべて非圧縮データだった場合の圧縮後ブロック数を最大ブロック数とし、圧縮単位が圧縮ブロックを含む場合、圧縮ブロック数は最大ブロック数より小さい。即ち、圧縮単位内のブロックがすべて非圧縮ブロックである場合、圧縮後ブロック数は一定であり、圧縮単位が圧縮ブロックを含む場合、圧縮後ブロック数は可変である。なお、圧縮後ブロック数が最大ブロック数と同じまたは多い場合、非圧縮データを圧縮データの代わりに格納するため、圧縮後ブロック数が最大ブロック数と同じである場合、非圧縮データが格納されることとし、当該圧縮単位のデータが圧縮されているか否かを示す圧縮フラグはなくても良い。図23では8KBの圧縮単位が8KBの圧縮データとなっている場合を示しており、このときは圧縮フラグで該当の8KBが圧縮データか否かを判別するようにしている。

[0081] 図24は、STコントローラ400が圧縮記憶デバイス600aから圧縮情報を取得する第一の動作を示す。

[0082] 圧縮情報は、圧縮記憶デバイス600aに格納されている対象データの圧縮に関する情報である。STコントローラ400は、対象データの圧縮情報を取得するための圧縮情報リクエストを圧縮記憶デバイス600aへ発行する。圧縮記憶デバイス600aは、圧縮情報リクエストに応じて対象データ

の圧縮情報を圧縮管理テーブル 920 から読み出し、圧縮情報を示す圧縮情報レスポンスを ST コントローラ 400 へ発行する。圧縮情報は、対象データの圧縮の状態を示す。圧縮情報リクエストは、圧縮情報を要求することを示すコマンドコードと、当該リクエストの識別子と、圧縮記憶デバイス 600 a 内での対象データの論理アドレスであるアドレスと、対象データのサイズであるリクエストサイズとを含む。圧縮情報レスポンスは、対応する圧縮情報リクエストの識別子と、対象データを含む圧縮単位群の圧縮後のサイズである圧縮後全体サイズと、当該対象データを含む圧縮単位群の先頭の圧縮単位を伸張した際に、その伸張データの先頭から対象データの先頭までのオフセットである先頭オフセット位置とを含む。更に圧縮情報レスポンスは、当該圧縮単位群内の圧縮単位毎に、当該圧縮単位の圧縮の有無を示す圧縮有無フラグと、圧縮後の当該圧縮単位内のブロック数を示す圧縮後ブロック数とを含む。なお、ST コントローラ 400 が、圧縮後ブロック数が圧縮前ブロック数と同じ場合は非圧縮と判定し、圧縮後ブロック数と圧縮前ブロック数が異なる場合は圧縮と判定する場合、圧縮有無フラグを不要としても良い。

- [0083] この動作によれば、ST コントローラ 400 は、圧縮記憶デバイス 600 a や SS 接続機構 200 等、ST コントローラ 400 以外により圧縮されたデータの圧縮情報を取得することができる。また、ST コントローラ 400 が、SS システム内の他の部位へ圧縮情報を送ることにより、圧縮を行った部位以外の部位であっても、伸張に必要な圧縮情報を取得することができる。また、ST コントローラ 400 は、複数の圧縮記憶デバイス 600 a に分散されて格納されている圧縮データを論理アドレス順に夫々読み出すと共に、夫々の圧縮記憶デバイス 600 a から圧縮情報を取得することにより、複数の圧縮記憶デバイス 600 a に分散された格納されている圧縮データを連続して ST メモリ 410 へ保存するように並びかえることができる。なお、ST コントローラ 400 は、並び替えには SGL をつかい、論理的に連続させ、物理的に連続していなくても良い。これにより、SS 接続機構 200 は

、ＳＴメモリ４１０から圧縮データをシーケンシャルに読み出すことができる。

[0084] 図２５は、ＳＴコントローラ４００が圧縮記憶デバイス６００ａから圧縮データを取得する動作を示す。

[0085] 圧縮記憶デバイス６００ａは、ＳＴコントローラ４００から受信したデータを圧縮して格納した場合、通常のリードリクエストに応じて、圧縮データを伸張して非圧縮データを生成し、非圧縮データをＳＴコントローラ４００へ送信することができる。

[0086] ここでは、ＳＴコントローラ４００は、圧縮記憶デバイス６００ａに格納されている対象データを圧縮データのままで取得するための、圧縮データリードリクエストを圧縮記憶デバイス６００ａへ発行する場合について説明する。圧縮記憶デバイス６００ａは、圧縮データリードリクエストに応じて圧縮データを読み出し、対象データの圧縮データを送信し、その読み出しの結果を示す圧縮データリードレスポンスをＳＴコントローラ４００へ発行する。圧縮データリードリクエストは、圧縮データの圧縮したままのリードを指示するコマンドコードと、当該リクエストの識別子と、圧縮記憶デバイス６００ａ内での対象データの論理アドレスであるアドレスと、対象データのサイズであるリクエストサイズとを含む。なお、対象データのサイズは前述の圧縮情報リードで取得した値を利用する。圧縮データリードレスポンスは、対応する圧縮データリードリクエストの識別子と、リードの処理に関するステータスと、ＳＴコントローラ４００へ転送される対象データのサイズを示す転送サイズとを含む。

[0087] この動作によれば、ＳＴコントローラ４００は、対象データの圧縮データを、圧縮記憶デバイス６００ａで伸張させずに取得することができる。これにより、圧縮記憶デバイス６００ａ及びＳＴコントローラ４００の間のバスの帯域と、ＳＴコントローラ４００内の帯域の使用量を削減することができる。

[0088] 圧縮情報の取得は、前述の圧縮情報リクエストを利用する方法以外に、１

回のリクエストで圧縮情報と圧縮データをＳＴコントローラ４００が取得する方法として、図２６に示された第二の動作のように、圧縮データリード時に圧縮データの先頭または末尾に圧縮情報を付与して送る方法、または、図２７に示された第三の動作のように、圧縮データリードのレスポンスに圧縮情報を付加する方法としても良い。この場合は、圧縮情報リードを実施せずに、圧縮データリードのみを実施し、圧縮データリードリクエストの対象データサイズには、圧縮前のデータサイズを設定する。このリクエストに応じた圧縮リードレスポンスの対象データのサイズは、圧縮後のデータサイズを設定する。ＳＴコントローラ４００は、圧縮リードレスポンスを受けたとき、圧縮データと一緒に読み出された圧縮情報から、圧縮後データサイズを取得し、圧縮リードレスポンスの転送済みサイズと比較することで、ＳＴコントローラ４００に転送サイズのアンダーフローエラーが発生した場合も、転送済みのデータが圧縮後データサイズ分転送されていることを判定し、正常転送と判定する。なお、この後、伸張ＤＭＡ転送を実施するが、その中の保障コードチェックにより、最終的な転送データの整合性を判断する。圧縮データと圧縮情報を同時に読み出して、圧縮リードレスポンスに圧縮情報を設定しないことで、圧縮リードレスポンスはオートレスポンスで実施できる。また、圧縮情報取得と圧縮データ取得のリクエストを１回にまとめることで、リクエスト数増加による性能オーバーヘッドを低減する。

[0089] 図２８は、伸張ＤＭＡ処理におけるＳＴコントローラ４００とＳＳ接続機構２００の間の動作を示す。

[0090] ＳＴコントローラ４００は、ＳＳ接続機構２００に対し、伸張ＤＭＡ処理を指示する伸張ＤＭＡリクエストを発行することができる。伸張ＤＭＡ処理は、指定されたメモリから対象データの圧縮データを読み出し、当該圧縮データを伸張して非圧縮データを生成し、指定されたメモリへ当該非圧縮データを書き込む処理である。例えば、伸張ＤＭＡ処理は、ＳＴコントローラ４００内のＳＴメモリ４１０から対象データの圧縮データを読み出し、当該圧縮データを伸張して非圧縮データを生成し、サーバ１００内のＳＶメモリ１

10へ当該非圧縮データを書き込む。SS接続機構200は、伸張DMAリクエストに基づいて、STメモリ410内の、圧縮データに関する情報を示すSGL (Scatter/Gather List) である圧縮情報SGLを読み出す。更にSS接続機構200は、STメモリ410内の圧縮情報SGLで示された記憶領域から圧縮データを読み出す。更にSS接続機構200は、圧縮データを伸張し、非圧縮データをSVメモリ110へ書き込んだ後、伸張DMAレスポンスをSTコントローラ400へ発行する。圧縮情報SGLは、ストレージシステム300内に分散して格納されている複数の圧縮データがSTメモリ410に読み出された場合に、STメモリ410内の複数の圧縮データのアドレスやサイズを順に示すリストである。例えば、圧縮データがRAIDグループに格納されている場合、圧縮データは、STメモリ410内に分散して格納される。SS接続機構200は、圧縮情報SGLを辿ることにより、圧縮データをシーケンシャルに読み出し、順に伸張してSVメモリ110へシーケンシャルに書き込むことができる。これにより、分散して格納された各部を伸張して転送する場合に比べて、性能を向上させることができる。

[0091] 伸張DMAリクエストは、伸張DMA処理を示すコマンドコードと、伸張DMAリクエストの基となるサーバ100からのリードリクエストを示す識別子と、対象データの圧縮等のパラメータと、対象データの非圧縮データに付加される保障コードを示す非圧縮保障コード情報と、対象データの圧縮データに付加される保障コードを示す圧縮保障コード情報と、対象データの非圧縮データの長さを示す非圧縮転送長と、対象データの非圧縮データの長さを示す非圧縮転送長と、STメモリ410内の圧縮情報SGLのアドレスであるSGLアドレスとを含む。なお、SGLアドレスの代わりにデータのアドレスを直接指定しても良いし、データのアドレスとSGLアドレスの両方を設定しても良い。データのアドレスのみ設定されている場合は、圧縮情報SGLの取得は省略される。

[0092] 保障コードは、リードされたデータと当該データが格納されているアドレ

スとからCRC (Cyclic Redundancy Check) 等の保障コード演算により算出される値の期待値である。リード時、非圧縮データに付加された保障コードは、圧縮されているため、STコントローラ400によりチェックされることができない。そこで、STコントローラ400は、保障コードのチェックに必要な保障コード情報を伸張DMAリクエスト等を含めて、SS接続機構200へ送信する。STコントローラ400により管理されているアドレスは、サーバ100により指定されるアドレスと違うことがあるため、保障コード情報に含められ、STコントローラ400からSS接続機構200へ送信される。例えば、STコントローラ400がシンプロビジョニングを行っている場合、ストレージプール内のアドレスが保障コード情報に含まれる。これにより、SS接続機構200は、圧縮データ及び非圧縮データに誤りがあるか否かと、圧縮データが指定されたアドレスから読み出されているか否かとをチェックすることができる。また、圧縮データに保障コードが付加されている場合、処理をハードウェアにオフロードし、STプロセッサ420の負荷を低減することを考え、STプロセッサ420は、SS接続機構200へ期待値を伸張DMAリクエストで渡す。

[0093] 圧縮データが記憶デバイス専用コードを含む場合、SS接続機構200は、記憶デバイス専用コードの位置を示す情報に基づいて、圧縮データから記憶デバイス専用コードを除去する。記憶デバイス専用コードの位置を示す情報は、伸張DMAリクエスト内のパラメータに含まれても良いし、SS接続機構200内のレジスタに予め格納されていても良い。これにより、圧縮記憶デバイス600aがデータを圧縮して記憶デバイス専用コードを付加した場合でも、SS接続機構200は、圧縮記憶デバイス600aから読み出された圧縮データから記憶デバイス専用コードを除去して伸張することができる。なお、記憶デバイス専用コードは、圧縮データと伸張後データ両方についている可能性があり、両方に付加されている場合は、SS接続機構200が夫々両方の除去を行う。

[0094] 伸張DMAレスポンスは、伸張DMAレスポンスであることを示すコマン

ドコードと、対応する伸張DMAリクエストの識別子と、伸張DMA処理に関するステータスと、サーバ100へ転送された非圧縮データのサイズを示す非圧縮転送サイズと、伸張DMA処理におけるエラーを示すエラーレポートとを含む。この伸張DMAレスポンスによれば、STコントローラ400は、対象データの非圧縮データを知ることができる。

[0095] 更に、STコントローラ400は、SS接続機構200に対し、圧縮DMA処理を指示する圧縮DMAリクエストを発行することができる。圧縮DMA処理は、指定されたメモリから対象データの非圧縮データを読み出し、当該非圧縮データを圧縮して圧縮データを生成し、指定されたメモリへ当該圧縮データを書き込む処理である。例えば、圧縮DMA処理は、サーバ100内のSVメモリ110から対象データの非圧縮データを読み出し、当該非圧縮データを圧縮して圧縮データを生成し、STコントローラ400内のSTメモリ410へ当該圧縮データを書き込む。

[0096] 図29は、圧縮情報SGLを示す。

[0097] STプロセッサ420は、STメモリ410内に対象データを格納する記憶領域を確保した場合、その記憶領域を示す圧縮情報SGLを生成し、STメモリ410内に保存する。STメモリ410内のデータを格納する領域は複数個所に分割されている場合もあり、圧縮情報SGLはそれら複数のアドレス毎に夫々情報を持っても良い。圧縮情報SGLは、当該圧縮情報SGLの識別子と、当該圧縮情報SGLに関する情報として、当該圧縮情報SGL内に何個の有効なアドレスがあるか、次のSGLがなく終端のSGLであるか、次のSGLがある場合は、次SGLアドレスが有効であることなどを示すSGL情報とを含む。圧縮情報SGLは更に、対象データ内の転送データ(i)毎の転送データ情報(i)を含む(iは0、1、...)。転送データ情報(i)は、STメモリ410内での当該圧縮情報(i)のアドレス(i)と、そのアドレス(i)に対応する転送データ(i)の圧縮データサイズを示す転送サイズ(i)と、当該アドレス(i)に対応する転送データ(i)を構成する圧縮単位毎の圧縮有無を示す圧縮有無フラグ(i)と、転送デー

タ（i）内の圧縮単位毎に圧縮により何ブロックになっているかを示す圧縮後ブロック数（i）とを含む。更に当該圧縮情報SGLは、前述のSGL情報で次の圧縮情報SGLが連続していると設定されている場合には、次の圧縮情報SGLのアドレスである次SGLアドレスを含む。また、圧縮情報SGLは、圧縮情報SGL内の各フィールドをチェックするためのSGLフィールドチェックコードを含む。なお、圧縮単位毎に圧縮の有無があり、圧縮できなかった場合は、圧縮単位と同じブロック数を格納し、圧縮できた場合は、圧縮単位より少ないブロック数を格納するとして、圧縮有無フラグを省略することとしても良い。また、転送サイズは圧縮後ブロック数から計算するとして、省略しても良い。さらに、ここに示した圧縮情報SGLは一例であり、前述の圧縮情報の取得により取得される先頭オフセット位置など他の圧縮に関する情報を搭載しても良い。

[0098] これにより、リード処理において、STプロセッサ420が、圧縮記憶デバイス600aからの対象データを格納するために、STメモリ410内に連続する記憶領域を確保できない場合であっても、SS接続機構200が圧縮情報SGLを辿ることにより、STメモリ410から対象データをシーケンシャルに読み出すことができる。

[0099] なお、伸張DMAリクエストが、SGLアドレスの代わりに圧縮情報SGLの内容を含んでも良い。また、伸張DMAリクエストのサイズを拡大、または連結させ、多くの情報を載せるようにし、SGLを使わずに圧縮情報SGLの内容を伸張DMAリクエストに載せても良い。また、伸張DMAリクエストとSGLの双方に分けて圧縮情報を搭載しても良い。

[0100] また、SVプロセッサ120は、SVメモリ110内に対象データを格納する記憶領域を確保した場合、その記憶領域を示すSGLであるサーバSGLを生成し、SVメモリ110内に保存する。サーバSGLは、当該サーバSGLの識別子と、当該SGL内の有効なアドレス数、次SGLがあるかどうかのフラグ、及び当該SGLが終端であるかのフラグなどで構成されるSGL情報と、サーバ100側の格納先アドレスと、転送サイズとで構成され

る。格納先アドレスと転送サイズは、圧縮情報 S G L と同様に複数格納されていても良い。また、次 S G L アドレスや S G L フィールドのチェックコードを含んでも良い。さらに、他の転送先に関する情報を含んでも良い。

[0101] ライト処理において、S V プロセッサ 1 2 0 が、対象データを格納するために、S V メモリ 1 1 0 内に連続する記憶領域を確保できない場合であっても、S S 接続機構 2 0 0 がサーバ S G L を辿ることにより、S V メモリ 1 1 0 から対象データをシーケンシャルに読み出すことができる。

[0102] 図 3 0 は、データフォーマットを示す。

[0103] この図は、圧縮データフォーマットと非圧縮データフォーマットとの関係を示す。圧縮データは、この形式で圧縮記憶デバイス 6 0 0 a に格納されている。例えば、圧縮データは、この形式のまま、圧縮記憶デバイス 6 0 0 a から S T コントローラ 4 0 0 へ読み出され、S T コントローラ 4 0 0 から S S 接続機構 2 0 0 へ読み出される。S S 接続機構 2 0 0 は、圧縮データを伸張することにより、非圧縮データを生成する。圧縮データフォーマットとして、第一圧縮データフォーマット、第二圧縮データフォーマット、第三圧縮データフォーマットがあり、非圧縮データフォーマットとして、第一非圧縮データフォーマットや、第二非圧縮データフォーマット等がある。なお、圧縮データフォーマットと非圧縮データフォーマットの組み合わせは、表記の対応関係意外の組み合わせとしても良い。また、ここに示しているフォーマットは一例で、圧縮データフォーマットと非圧縮データフォーマットとの対応関係が定められていれば、他のフォーマットとしても良い。

[0104] 第一非圧縮データフォーマットにおいて、圧縮データは、複数の圧縮データブロックを含む。圧縮データブロックは、圧縮単位である。各圧縮データブロックの後には、その圧縮データブロックの保障コードが付加されている。圧縮データブロックが伸張されると、複数のデータブロックになる。データブロックは、ブロックである。各データブロックの後には、そのデータブロックの保障コードが付加されている。

[0105] 圧縮記憶デバイス 6 0 0 a が、S T コントローラ 4 0 0 からのデータを圧

縮して格納する場合、圧縮データを複数のFM620に分散して書き込むため、圧縮記憶デバイス600aに固有のCRCやFM620内のアドレスを示す記憶デバイス専用コードを付加する場合がある。この場合、第二の非圧縮データフォーマットが用いられる。第二の非圧縮データフォーマットにおいて、各データブロックの後には、そのデータブロックの保障コードが付加されており、その保障コードの後には、記憶デバイス専用コード（専用コード）が付加されている。なお、ここで示す専用コードとは、記憶デバイス600自身が専用に利用するために付加した独自のコードであり、例えば、記憶デバイス600が生成付加した保障コードなどが該当する。SS接続機構200が圧縮データを伸張する場合、非圧縮データから記憶デバイス専用コードを除去してサーバ100へ転送する。また、第三の非圧縮データフォーマット及び第三の非圧縮データフォーマットのように、圧縮データに関しても、記憶デバイス専用コードが付加される場合があり、この場合も同様にSS接続機構200が圧縮データを伸張する前に、記憶デバイス専用コードを除去してから伸張処理を実施する。

[0106] 以下、サーバ100が圧縮記憶デバイス600a内の圧縮データを対象データとして、対象データを読み出すリード処理について説明する。本実施例では、DMA処理の中で、SS接続機構が伸張処理を担うことで、サーバストレージ間のメモリ間での直接データ転送処理の一貫でのデータの伸張処理を可能にする。

[0107] 本実施例においては、サーバ100とストレージシステム300の間にSS接続機構200を設けることで、サーバ100側とストレージシステム300側の処理の増加を低減し、圧縮データの伸張処理を実現する。この構成により、メモリ間でデータを直接やり取りする中で処理データ量を低減できる。よって、サーバストレージ間の帯域の有効利用を図ることが可能となる。

[0108] 図31は、SSシステムのリード処理の中の第一処理を示し、図32は、SSシステムのリード処理の中の第一処理に続く第二処理を示す。これらの

図は、サーバ１００内のＳＶプロセッサ１２０と、ＳＳ接続機構２００内のＩＦプロセッサ２１０と、ＳＴコントローラ４００内のＳＴプロセッサ４２０と、圧縮記憶デバイス６００ａ内のＳＤプロセッサ６１２との動作を示す。ここでは、対象データの伸張処理がＳＳ接続機構２００により行われる場合のリード処理について説明する。

[0109] Ｓ１１１０においてサーバ１００内のＳＶプロセッサ１２０は、ＳＶメモリ１１０内に、リードしたデータを格納する記憶領域として、リードサイズ分のＳＶメモリ１１０領域を準備する。ここでＳＶプロセッサ１２０は、ＳＶ領域を示すＳＧＬであるサーバＳＧＬを生成してＳＶメモリ１１０へ保存する。Ｓ１１２０においてＳＶプロセッサ１２０は、リードリクエストを生成してＳＳ接続機構２００へ発行する。リードリクエストは、対象データを格納する記憶領域を示す、仮想ボリュームと、その仮想ボリューム内の仮想ページとを示す。

[0110] Ｓ１１３０においてＳＳ接続機構２００内のＩＦプロセッサ２１０は、ＳＶプロセッサ１２０からキューイングされたリードリクエストを解析するリードリクエスト処理を行う。Ｓ１１４０においてＩＦプロセッサ２１０は、解析結果に基づくリードリクエストを生成してＳＴコントローラ４００へ発行する。なお、ＩＦプロセッサ２１０からＳＴコントローラ４００へのリードリクエストを、ＳＶプロセッサ１２０からＩＦプロセッサ２１０へリードリクエストと区別するために、リードコマンドと呼ぶことがある。発行先は、特定された対象データを格納する記憶領域のオーナー権を有するＳＴコントローラ４００及びＳＴプロセッサである。Ｓ１１５０においてＩＦプロセッサ２１０は、サーバＳＧＬを取得するためのサーバＳＧＬ処理を行う。Ｓ１１６０においてＩＦプロセッサ２１０は、サーバ１００からサーバＳＧＬを受信する。Ｓ１１７０においてＩＦプロセッサ２１０は、受信されたサーバＳＧＬを内部メモリに格納する。なお、リードリクエストにＳＧＬに搭載するアドレス情報や長さ情報を載せてもよく、リードリクエストにのりきらない場合に、残りの情報をＳＧＬの形としても良い。

[0111] S 1 2 1 0においてS Tコントローラ4 0 0内のS Tプロセッサ4 2 0は、I Fプロセッサ2 1 0からキューイングされたリードリクエストを解析するリードリクエスト処理を行う。ここでS Tプロセッサ4 2 0は、仮想ボリューム管理テーブル8 1 0とストレージプール管理テーブル8 2 0とR A I Dグループ管理テーブル8 3 0と物理デバイス管理テーブル8 4 0と仮想ページ管理テーブル8 5 0と実ページ管理テーブル8 6 0とに基づいて、リードリクエストから、対象データを格納している圧縮記憶デバイス6 0 0 aと、その圧縮記憶デバイス6 0 0 a内の記憶領域の論理アドレスとを特定する。S 1 2 2 0においてS Tプロセッサ4 2 0は、伸張DMA処理を行うか否かを判定する（圧縮判定）。対象データが予め定められた判定基準を満たすと判定された場合、S 1 2 3 0においてS Tプロセッサ4 2 0は、対象データの圧縮情報を要求する圧縮情報リクエストを圧縮記憶デバイス6 0 0 aへ発行する。圧縮判定には、S Tコントローラ4 0 0内のキャッシュに対象データが記憶されているかの判定も含んでいる。キャッシュ判定については後述する。なお、圧縮判定（S 1 2 2 0）の後に処理をS 1 4 2 0へ移行させ（図中の点線の矢印）、非圧縮データのサイズでデータ格納領域確保を行い、圧縮データリードリクエストのみで圧縮情報と圧縮データの両方を取得しても良い。この場合、圧縮記憶デバイス6 0 0 aのS Dプロセッサ6 1 2は、圧縮情報取得などの処理（S 1 3 2 0～S 1 3 6 0、図中の点線の枠内）は、圧縮リードリクエストを受けてから（S 1 4 3 0の後に）実施する。なお、圧縮データリードリクエストの要求を想定して、圧縮記憶デバイス6 0 0 aが先読み処理を実施していても良い。

[0112] S 1 3 1 0において圧縮記憶デバイス6 0 0 a内のS Dプロセッサ6 1 2は、S Tプロセッサ4 2 0からキューイングされた圧縮情報リクエストを解析する圧縮情報リクエスト処理を行う。S 1 3 2 0においてS Dプロセッサ6 1 2は、R A M内の圧縮管理テーブル9 2 0から対象データの圧縮情報を取得する。S 1 3 3 0においてS Dプロセッサ6 1 2は、対象データの圧縮データのサイズである圧縮後全体サイズを計算する。S 1 3 4 0においてS

Dプロセッサ612は、対象データの圧縮情報を含む圧縮情報レスポンスをSTプロセッサ420へ発行する。S1350においてSDプロセッサ612は、FM620から対象データの圧縮データを読み出し、圧縮データをRAM内の記憶領域であるSD領域へ保存する。S1360においてSDプロセッサ612は、圧縮情報に基づいてRAM内の圧縮データを並び替える。

[0113] S1410においてSTプロセッサ420は、SDプロセッサ612からキューイングされた圧縮情報レスポンスを解析する圧縮情報レスポンス処理を行う。S1420においてSTプロセッサ420は、STメモリ410内に、圧縮情報レスポンスに基づいて圧縮データを格納するためのSTメモリ410内の記憶領域であるST領域を確保する（データ格納領域確保）。S1430においてSTプロセッサ420は、圧縮データリードリクエストをSDプロセッサ612へ発行する。S1440においてSTプロセッサ420は、ST領域を示す圧縮情報SGLを生成しSTメモリ410へ保存する。

[0114] S1510においてSDプロセッサ612は、STプロセッサ420からキューイングされた圧縮データリードリクエストに基づいてRAMから圧縮データを読み出し、処理をS1520（図中のA）へ移行させる。S1520においてSDプロセッサ612は、読み出された圧縮データを圧縮データのままSTコントローラ400のST領域へ転送する。S1530においてSDプロセッサ612は、圧縮データリードリクエストの結果を示す圧縮データリードレスポンスをSTプロセッサ420へ発行する。S1540においてSDプロセッサ612は、RAM内のSD領域を解放し、リード処理を完了する。なお、RAM内のSD領域の開放は、再送を考慮し、伸張DMAレスポンスが戻ってきてから、STプロセッサ420の指示で開放する。または、タイマをはって、所定時間経過後に開放してもよい。

[0115] もし、STプロセッサ420は、圧縮判定S1220で伸張DMAを実施しないと判断した場合は、通常のリード転送を実施することとし、S1420で非圧縮のデータサイズでデータ格納領域を確保し、S1430で通常リ

ードリクエストを発行する。その後、STプロセッサ420は、STメモリ410に読み出されたデータに対して、S1620で通常リード転送のDMAリクエストを実施する。この場合、圧縮情報リクエストS1230、圧縮情報リクエスト処理S1310、圧縮後全体サイズ計算S1330、圧縮情報レスポンスS1340、圧縮情報レスポンス処理S1410は行われない。もし、伸張DMA転送しないと判断したデータが、記憶デバイス600中で圧縮されたデータであった場合は、圧縮データリードリクエストS1430が圧縮したまま読み出す旨の指示を含まない通常のリードリクエストとなり、それを受けた記憶デバイス600が、圧縮情報取得S1320、FMから圧縮データの読み出しとDRAM格納S1350、圧縮データ並び替えS1360を記憶デバイス600内で実施し、記憶デバイス600の伸張機能（圧縮・伸張デバイス614）で伸張したデータを圧縮リード処理S1510の代わりに通常リード処理を実施し、圧縮データ格納S1520の代わりに通常データ格納し、圧縮リードレスポンスS1530の代わりに通常リードレスポンスをSTプロセッサ420に返す。

[0116] なお、記憶デバイス600が圧縮機能を持たずに、記憶デバイス600外で圧縮されたデータが記憶デバイス600に格納されている場合について説明する。例としては、圧縮機能を持たない記憶デバイス600であるHDDに圧縮データが格納されている場合などがこれに該当する。なお、ストレージシステム300は、予め接続される記憶デバイス毎に圧縮伸張機能の有無を把握している。また、把握できていない場合は、ストレージシステム300の管理装置から設定、または、STプロセッサ420が記憶デバイスへアクセスし、圧縮伸張機能の有無を確認できるものとする。STプロセッサ420は、記憶デバイス600に圧縮情報リクエスト（S1320）などを行わずに、STプロセッサ420が圧縮判定（S1220）の後に圧縮情報を格納している記憶領域から圧縮情報を読み出して、データ格納領域確保（S1420）に処理を進める（図中の点線の矢印）。圧縮情報の格納場所としてはSTメモリ410または、STコントローラ400に搭載したフラッシュ

メモリまたは、記憶デバイス600等が考えられる。STプロセッサ420は、圧縮情報を取得した後、圧縮情報SGLの生成を行い(S1440)、圧縮データをSTコントローラ400側に読み出した(S1430~S1610)後に、SS接続機構200に伸張DMA転送リクエスト(S1620)を発行する。なお、伸張DMAを行わない場合、STプロセッサ420は、記憶デバイス600から読み出した(S1430~S1610)後に、読み出したデータが圧縮データである場合は、STプロセッサ420、または、STコントローラ400に搭載した伸張ハードウェアを利用して、圧縮データを伸張してから、S1620の伸張DMAリクエストの代わりに、通常の非圧縮データのDMA転送でデータを転送する。もし、読み出したデータが非圧縮データである場合は、そのままS1620の伸張DMAリクエストの代わりに、通常の非圧縮データのDMA転送でデータを転送する。なお、読み出したデータが非圧縮データである場合は、記憶デバイス600よりSTコントローラ400へ読み出すときに、記憶デバイス600内で圧縮データを伸張している場合と、記憶デバイス600内でデータが圧縮されていない場合がある。

[0117] なお、STプロセッサ420は、伸張ハードウェアとして、SS接続機構200内の伸張DMAを使って、一旦圧縮データをSTコントローラ400からSS接続機構200に読み出して、伸張し、STコントローラ400に書き戻した後に、S1620の伸張DMAリクエストの代わりに通常DMAリクエストを発行して転送しても良い。STプロセッサ420は、通常DMAリクエストの前には、STメモリ410領域からSGLの生成を行い、非圧縮データをSTコントローラ400側に読み出した後に、SS接続機構200にDMA転送リクエストを発行する。

[0118] S1610においてSTプロセッサ420は、SDプロセッサ612からキューイングされた圧縮データリードレスポンスを解析する圧縮データリードレスポンス処理を行う。S1620においてSTプロセッサ420は、伸張DMAリクエストをIFプロセッサ210へ発行する。このとき、DMA

リクエストまたは、圧縮情報 SGL に圧縮前のデータ長と圧縮後のデータサイズを含めることで、SS 接続機構 200 が ST コントローラ 400 から圧縮後のデータサイズ分データを読み出し、サーバ 100 からの要求サイズ分、伸張 DMA 転送を完了できたかを判断できる。

[0119] S1710 において IF プロセッサ 210 は、ST プロセッサ 420 からキューイングされた伸張 DMA リクエストを解析する伸張 DMA リクエスト処理を行う。S1720 において IF プロセッサ 210 は、必要に応じて ST メモリ 410 内の圧縮情報 SGL を取得する。S1730 において IF プロセッサ 210 は、DMA コントローラ 220 の伸張 DMA 処理を起動する。S1740 において DMA コントローラ 220 は、IF プロセッサ 210 からの指示に基づいて伸張 DMA 処理を行うことにより、ST コントローラ 400 内の ST 領域に格納されている圧縮データを読み出して伸張し、得られる非圧縮データをサーバ 100 内の SV 領域へ書き込む。S1750 において IF プロセッサ 210 は、DMA コントローラ 220 から伸張 DMA 処理完了の通知を受ける。S1760 において IF プロセッサ 210 は、正常転送終了の場合はオートレスポンスを用いてリードレスポンスを SV プロセッサ 120 へ発行する。

[0120] S1810 において SV プロセッサ 120 は、IF プロセッサ 210 からキューイングされたリードレスポンスに応じてリード処理を完了する。

[0121] S1770 において IF プロセッサ 210 は、伸張 DMA リクエストに対する伸張 DMA レスポンスを ST プロセッサ 420 へ発行する。S1910 において ST プロセッサ 420 は、IF プロセッサ 210 からキューイングされた伸張 DMA レスポンスを解析する伸張 DMA レスポンス処理を行う。S1920 において ST プロセッサ 420 は、ST 領域を解放する。S1930 において ST プロセッサ 420 は、リード処理を完了する。

[0122] 以上のリード処理によれば、ST コントローラ 400 が SS 接続機構 200 に圧縮データの伸張を行わせることにより、圧縮記憶デバイス 600a から SS 接続機構 200 までの経路の帯域の使用量を削減でき、SS システム

のI/O性能を向上させることができる。

[0123] 前述したように、リード処理においてSTプロセッサ420は、IFプロセッサ210からリードリクエストを受信した時点(S1220)で伸張DMA処理を行うか否かを判定する。ストレージシステム300側で伸張DMA判定することで、サーバ100側の圧縮情報管理、伸張処理が不要となる。さらに、ストレージシステム300側で圧縮情報を管理することで、複数のサーバ100間で圧縮情報の転送を行わなくても、共有ストレージとしてサーバ100間で圧縮データを共有できる。なお、伸張DMA判定をストレージ側でI/Oに連動して行うことで、I/OサイズやSTメモリ410のヒットミス判定情報等I/Oの特性に関する情報をふまえて伸張DMA転送を実施するか否かをI/O毎に判定することができる。また、更に、伸張DMA転送を判定した後に、STプロセッサ420は、SDプロセッサ612から圧縮データリードレスポンスを受信した時点(S1610)で、再度、記憶デバイス600の応答に応じて伸張DMA処理を行うか否かを判定しても良い。伸張DMA処理を行うと判定された場合、SS接続機構200は、前述のリード処理に示したように伸張DMA処理を行う。伸張DMA処理を行わないと判定された場合、SS接続機構200は、STコントローラ400内の非圧縮データをサーバ100へ転送する通常DMA処理を行う。通常DMA処理を行う場合、圧縮記憶デバイス600aが対象データの圧縮データを伸張してSTコントローラ400へ転送し、STコントローラ400が非圧縮データを格納しても良いし、圧縮記憶デバイス600aが対象データの圧縮データをSTコントローラ400へ転送し、STコントローラ400が非圧縮データを伸張して格納しても良い。以下、STプロセッサ420が伸張DMA処理を行うと判定する圧縮判定(S1220)の条件の例であるR0～R13について説明する。なお、R0～R13の条件複数を圧縮判定に同時に用いても良い。

[0124] (R0) 対象データがSTコントローラ400においてキャッシュヒットしたときで、対象のキャッシュヒットしたデータが圧縮データの場合、STプ

ロセッサ４２０はそのデータに対し伸張DMA処理を実施する。対象のキャッシュヒットしたデータが非圧縮データの場合は通常DMA処理を実施する。対象データがキャッシュミスした場合、STプロセッサ４２０は、リードリクエストが下記の他の条件を満たすときに伸張DMA処理を実施する。このように、STプロセッサ４２０が、ヒットミス判定をすることで、リードリクエストによる転送中の圧縮データに対してライトリクエストがあった場合に、排他制御を行うことができる。例えば、リード転送後、STプロセッサ４２０は、キャッシュ領域上の圧縮データを削除し、物理デバイスに格納されている圧縮データを伸張して、そのデータに対してライトを実施する。これにより、ライトとリードの衝突を防ぎ、かつ伸張DMA処理時のライト処理に対応することができる。

[0125] (R1) 対象データがSTコントローラ４００においてキャッシュミスしたときで、且つ対象データの仮想ページもしくはプール属性もしくは仮想ボリュームが圧縮属性を有するときで、且つ伸張機能を持ったサーバ１００もしくは伸張機能を持った外接ストレージへ圧縮された対象データを転送するとき、STプロセッサ４２０は伸張DMA処理を行うと判定する。外接ストレージは、STコントローラ４００に接続されSTコントローラ４００により制御されるストレージシステム３００である。記憶デバイス６００内に格納されている圧縮データをSS接続機構２００で伸張してサーバ１００に転送することで、ストレージシステム３００内のバスと、ストレージシステム３００及びSS接続機構２００の間のバスとにおける帯域を広く見せることができ、それらのバスにおける転送時間とを短くすることができる。

[0126] (R2) データの圧縮・伸張は予め定められた大きさを有する圧縮単位毎に行われる。よって、圧縮単位以下のデータサイズのリード要求に対しては、圧縮単位にするために余分なデータを含んでいる場合があり、非圧縮データで見ると読み出す量が多くなることがある。よって、この不要なデータの読み出し分、負荷が上がりシステムの性能が上がらない要因となりうる。このため、STプロセッサ４２０は、リードリクエストで指定される非圧縮時の

データサイズが規定値以下のときは伸張転送を実施しない判断をしてもよい。非圧縮時のデータサイズが規定値以上のときは、不要なデータの読み出しが読み出しデータに対して相対的に小さくなるため、伸張DMA転送の効果が十分に出るため、STプロセッサ420は、伸張DMA転送を実施する判定を行う。なお、規定値は、圧縮後のサイズが不要なデータを含んでいたとしても、圧縮単位よりも十分に小さければ、伸張DMA転送の効果が出るが、圧縮率を取得して、判定するといったオーバーヘッド削減のために、特にこの規定値を圧縮単位サイズに設定することで、記憶デバイス600またはSTメモリ410にある圧縮情報の読み出しを待たずに転送処理できるため、転送処理を高速化できる。

[0127] (R3) STプロセッサ420がリードリクエストのコマンド履歴をSTメモリ410に保存し、コマンド履歴に基づいてリードリクエストのシーケンシャル性を判定する。リードリクエストがシーケンシャルリードと判定され、且つ対象データの記憶領域が条件R1の圧縮属性を有する場合、伸張DMA処理を判断する。シーケンシャルリードに対して伸張DMA処理を実施することで、記憶デバイス600とSTコントローラ400の間のバス、STコントローラ400内のバス、STコントローラ400とSS接続機構200の間のバス、及びSS接続機構200とサーバ100の間のバスの処理遅延を隠蔽し、各伝送路の帯域を使い切ることができる。

[0128] (R4) 対象データの圧縮データがSTコントローラ400にキャッシュされていて、且つSTプロセッサ420がそのデータを、リードリクエストの後のライトデータであると判断した場合に、シーケンシャル性ありと判断し、更に対象データの記憶領域が条件R1の圧縮属性を有する場合、STプロセッサ420は伸張DMA処理を実施する。これにより、次のデータがある程度キャッシュ領域又はバッファ領域にデステージされていれば、STプロセッサ420は、次にそのデータを利用する可能性を考慮し、シーケンシャルリードアクセスと判断し、条件R3と同様の理由で伸張DMA処理を実施する。

- [0129] (R 5) リードリクエストが条件 R 1 の圧縮属性を有する記憶領域へのアクセスであり、且つ対象データのアクセス頻度が低い場合、キャッシュの必要性が低いため、S T プロセッサ 4 2 0 は、伸張したデータを保持しておかなくても良いと判断し、伸張 DMA 処理を実施する。これにより、S T プロセッサ 4 2 0 は、アクセス頻度の低いデータを、S T コントローラ 4 0 0 のキャッシュ領域にデステージしておかずに、すぐに消してよいと判断し、伸張 DMA 処理を実施する。伸張 DMA 処理の対象データは圧縮されており、S T コントローラ 4 0 0 のキャッシュ領域でキャッシュヒットした際に、上書きコストが高いため、S T プロセッサ 4 2 0 は、なるべくアクセス頻度が低く且つシーケンシャル性のあるものを伸張 DMA 処理し、アクセス頻度が高いデータは、伸張されて S T コントローラ 4 0 0 のキャッシュ領域に格納されているほうが望ましい場合が多いため、S T プロセッサ 4 2 0 は通常 DMA 処理を実施する。
- [0130] (R 6) S T コントローラ 4 0 0 が圧縮伸張機能付きキャッシュを有し、且つ対象データがそのキャッシュ経由でリードされる場合、S T プロセッサ 4 2 0 は、伸張 DMA 処理を実施することにより、圧縮したまま S S 接続機構 2 0 0 へ転送する。この場合、圧縮データをキャッシュに保存しても、圧縮データへのライトなどがあった場合にキャッシュが圧縮データを伸張することができるので、圧縮データのまま転送を実施する。
- [0131] (R 7) 対象データの非圧縮データが S T コントローラ 4 0 0 においてキャッシュヒットし、且つそのキャッシュが圧縮伸張機能付きキャッシュである場合、S T プロセッサ 4 2 0 は、伸張 DMA 転送を行うと判断し、キャッシュが対象データを圧縮してから、S S 接続機構 2 0 0 の伸張 DMA 処理で圧縮データの転送を実施することで帯域を広帯域化できる。
- [0132] (R 8) サーバ 1 0 0 が圧縮伸張機能付きメモリを有し、且つ S T コントローラ 4 0 0 からの転送先が圧縮伸張機能付きメモリである場合、S T プロセッサ 4 2 0 は通常 DMA 処理を実施する。伸張は圧縮伸張機能付きメモリで実施したほうが、S S 接続機構 2 0 0 と圧縮伸張機能付きメモリの間のバス

を広帯域化できる。

- [0133] (R 9) 対象データがSTコントローラ400から別のSTコントローラ400へ転送される場合で、且つ転送先のSTコントローラ400が圧縮伸張機能を有していない場合、STプロセッサ420は、伸張DMA処理を実施する。転送先のSTコントローラ400が圧縮伸張機能を有する場合、圧縮データのまま転送する。転送先のSTコントローラ400が圧縮伸張機能を有していない場合、転送された圧縮データを伸張できない。
- [0134] (R 10) 対象データの圧縮データを格納する圧縮記憶デバイス600aの負荷が高いとき、STプロセッサ420は、転送時の伸張を伸張DMA処理へオフロードする。SS接続機構200が複数ある場合、STプロセッサ420はSS接続機構200のI/Oのバランスに基づいて、伸張DMA処理の負荷分散を行う。
- [0135] (R 11) STコントローラ400が圧縮伸張機能を有しており、且つそのSTコントローラ400の負荷が高いとき、STプロセッサ420は、転送時の伸張を伸張DMAへオフロードする。その他の動作は、条件R 10と同じである。
- [0136] (R 12) STプロセッサ420がサーバ100からのヒント情報に基づいて、対象データへのアクセスのシーケンシャル性を判断した場合、STプロセッサ420は伸張DMA処理を実施する。ヒント情報は、サーバ100が圧縮及び伸張を行う場合の、サーバ100による圧縮及び伸張に関する情報である。この方法は、条件R 1と同様の効果を得るために、シーケンシャル性を判断する方法の一つである。
- [0137] (R 13) STコントローラ400内のキャッシュにダーティデータがない場合、ライトが衝突しないため、STプロセッサ420は伸張DMA処理を行うと判断する。伸張DMA処理中また伸張DMA処理後に、キャッシュ領域又はバッファ領域に格納されているデータは一度伸張されないと上書きなどができないため、STプロセッサ420は、このようなデータを伸張DMA処理の対象からはずす。

[0138] ここでは、リード処理中にDMAコントローラ220により行われる伸張DMA処理（S1740）について説明する。

[0139] 図33は、SS接続機構200による伸張DMA処理の中の第一処理を示し、図34は、SS接続機構200による伸張DMA処理の中の第一処理に続く第二処理を示す。

[0140] SS接続機構200のDMAコントローラ220は、STコントローラ400から伸張DMAリクエストを受信すると、伸張DMA処理を実行する。

[0141] S2110においてDMAコントローラ220は、伸張DMAリクエストを解析し、転送元と転送先のアドレスを特定する。なお、転送元のアドレスは、伸張DMAリクエストに設定されたアドレス（データアドレス）または圧縮情報SGLから取得する。転送先のアドレスについて、SS接続機構200は、サーバ100からのリードリクエストを受けた際に、リードリクエスト内の転送先のアドレスをリードリクエストの識別子と対応させて保持し、管理していたリードリクエストに含まれていたアドレス情報またはサーバSGLに含まれているアドレス情報を利用する。S2120においてDMAコントローラ220は、必要に応じて圧縮情報SGLを取得し、圧縮情報SGLより圧縮情報を取得する。

[0142] S2130においてDMAコントローラ220は、対象データを含む圧縮単位群に対して伸張しながら転送を行い、伸張後のデータサイズから先頭オフセット位置を減じたサイズが伸張DMAリクエストで指定された転送サイズとなり、DMA転送を全て完了するまで、伸張DMA処理の対象データの中から順に対象圧縮単位を選択し、S2130～S2240を繰り返す。

[0143] S2140においてDMAコントローラ220は、対象圧縮単位の圧縮データの保障コードがあるか否かを判定する。圧縮データの保障コードがないと判定された場合（S2140：N）、DMAコントローラ220は、処理をS2160へ移行させる。圧縮データの保障コードがあると判定された場合（S2140：Y）、S2150においてDMAコントローラ220は、圧縮データの保障コードをチェックする圧縮データチェック処理を行い、保

障コードを削除し、処理をS 2 1 6 0へ移行させる。

[0144] S 2 1 6 0においてDMAコントローラ2 2 0は、対象圧縮単位が圧縮されているか否かを判定する。対象圧縮単位が圧縮されていないと判定された場合（S 2 1 6 0 : N）、DMAコントローラ2 2 0は、処理をS 2 1 8 0へ移行させる。対象圧縮単位が圧縮されていないと判定された場合（S 2 1 6 0 : N）、DMAコントローラ2 2 0は、処理をS 2 1 8 0へ移行させる。対象圧縮単位が圧縮されていると判定された場合（S 2 1 6 0 : Y）、圧縮データに記憶デバイス専用コードが付加されており、伸張時に除去してから伸張する必要がある場合、専用コードを除去する。S 2 1 7 0においてDMAコントローラ2 2 0は、対象圧縮単位の伸張処理を行うことにより、対象圧縮単位の圧縮データから非圧縮データを生成し、処理をS 2 1 8 0へ移行させる。

[0145] S 2 1 8 0においてDMAコントローラ2 2 0は、伸張処理により生成された伸張後の非圧縮データの保障コードがあるか否かを判定する。非圧縮データの保障コードがないと判定された場合（S 2 1 8 0 : N）、DMAコントローラ2 2 0は、処理をS 2 2 1 0（図中のA）へ移行させる。非圧縮データの保障コードがあると判定された場合（S 2 1 8 0 : Y）、S 2 1 9 0においてDMAコントローラ2 2 0は、非圧縮データの保障コードをチェックする非圧縮データチェック処理を行い、保障コードを削除し、処理をS 2 2 1 0（図中のA）へ移行させる。

[0146] S 2 2 1 0においてDMAコントローラ2 2 0は、当該対象圧縮単位に先頭オフセット位置が設定されているか否か、即ち転送しないデータを含むか否かを判定する。対象圧縮単位がデータ選択条件を満たさないと判定された場合（S 2 2 1 0 : N）、DMAコントローラ2 2 0は、処理をS 2 2 3 0へ移行させる。ここでデータ選択条件は、対象圧縮単位に先頭オフセット位置が設定されており、転送しないデータを含むことである。対象ブロックがデータ選択条件を満たすと判定された場合（S 2 2 1 0 : Y）、つまり、対象圧縮単位に先頭オフセット位置が設定され、転送しないデータを含む場合

、S 2 2 2 0においてDMAコントローラ2 2 0は、対象圧縮単位を伸張したデータの先頭から先頭オフセット位置だけ転送開始位置を移動させ、先頭オフセット位置分のデータは転送せずに、先頭オフセット位置から後ろのデータを転送し、処理をS 2 2 3 0へ移行させる。これにより、伸張された非圧縮データの一部が対象データである場合であっても、DMAコントローラ2 2 0は対象データだけを転送することができる。

[0147] S 2 2 3 0においてDMAコントローラ2 2 0は、転送先に対象圧縮単位のデータを格納する。S 2 2 4 0においてDMAコントローラ2 2 0は、転送されたサイズだけ、対象圧縮単位の転送開始位置のアドレスを増加させ、転送サイズ分の転送が完了したか否かを判定する。転送が完了していないと判定された場合（S 2 2 4 0 : N）、DMAコントローラ2 2 0は、処理をS 2 1 3 0（図中のB）へ移行させる。転送が完了したと判定された場合（S 2 2 4 0 : Y）、S 2 2 5 0においてDMAコントローラ2 2 0は、伸張DMA処理が正常に完了した旨の通知をSTコントローラ4 0 0へ通知し、このフローを終了する。

[0148] 圧縮データに保障コードが付いておらず、圧縮前の非圧縮データにだけ保障コードが付いている場合であっても、且つ保障コードが不一致である場合、DMAコントローラ2 2 0は、エラー発生箇所の特定のために、伸張されたデータを同パスで転送しエラーの発生箇所を特定する。

[0149] なお、S 2 2 2 0においては、伸張された非圧縮データの一部が対象データである場合に、SS接続機構2 0 0が非圧縮データから対象データを抽出したが、STコントローラ4 0 0が非圧縮データから対象データを抽出しても良い。ここで、圧縮記憶デバイス6 0 0 a内の対象データの圧縮データは、圧縮伸張機能付FMのキャッシュ領域又はSTメモリ4 1 0のキャッシュ領域へ転送される。その後、STプロセッサ4 2 0は、圧縮伸張機能付きフラッシュのハードウェア又はSTプロセッサ4 2 0のソフトウェアを用いて、圧縮データを伸張して非圧縮データを生成、非圧縮データから対象データを抽出する。その後、STプロセッサ4 2 0は、圧縮伸張機能付きフラッシュ

ユのハードウェア又はS Tプロセッサ4 2 0のソフトウェアを用いて、抽出された対象データを圧縮して圧縮データに変換してキャッシュ領域へ保存する。これにより、S Tコントローラ4 0 0のキャッシュ領域と、S Vメモリ1 1 0との間のバスの帯域の使用率を低減し、リード性能を向上させることができる。

[0150] 以上の伸張DMA処理によれば、I/O特性を把握するストレージシステム3 0 0側の判断で、状況に応じてS Tコントローラ4 0 0からサーバ1 0 0への圧縮データを転送できる。かつデータの伸張はDMAコントローラ2 2 0で実現され、サーバ1 0 0側のS Vプロセッサ1 2 0の負荷を増大させることにはならない。特にDMAコントローラ2 2 0がサーバ1 0 0側に設置されている場合には、S Tコントローラ・サーバ間の帯域以上のデータを読み出すことが可能となり、帯域の有効利用を実現することが可能となる。例えば、大規模データ分析システムでは、分析のためにデータをストレージから読み出す際に、S Tコントローラ・サーバ間のリード帯域が狭いと、分析サーバの分析処理用のプロセッサの利用効率が落ち分析処理時間が増加してしまう。本システムによれば、読み出しのデータ量を削減されるために、リード帯域を広帯域化でき、読み出し時間が短縮により分析処理を時間短縮を実現することができる。

[0151] また、本実施例の構成によれば、圧縮データ及び非圧縮データが正常である（伸張が成功である）か否かをチェックすることができる。また、S Tメモリ4 1 0内の対象データが圧縮単位であるブロックの途中から格納されている場合でも、対象データだけをS Vメモリ1 1 0へ転送することができる。

[0152] ここでは、伸張DMA処理中に、圧縮データチェック処理（S 2 1 5 0）及び非圧縮データチェック処理（S 2 1 9 0）の結果に基づいて行われる保障コードチェック処理について説明する。

[0153] 図3 5は、伸張DMA処理中の保障コードチェック処理の中の第一処理を示し、図3 6は、伸張DMA処理中の保障コードチェック処理の中の第一処

理に続く第二処理を示し、図37は、伸張DMA処理中の保障コードチェック処理の中の第二処理に続く第三処理を示す。

- [0154] 伸張DMA処理中、S2310においてSS接続機構200のIFプロセッサ210は、圧縮データチェック処理に応じて、圧縮データの保障コードにエラーがあるか否かを判定する。圧縮データの保障コードにエラーがないと判定された場合（S2310：N）、IFプロセッサ210は、処理をS2510へ移行させる。なお、圧縮データに保障コードが付加されていない場合も、圧縮データの保障コードにエラーがない場合と同じフローとする。
- [0155] 圧縮データの保障コードにエラーがあると判定された場合（S2310：Y）、S2320においてIFプロセッサ210は、圧縮データの保障コードでエラーが発生したことを示す情報を伸張DMAレスポンスに設定する。S2330においてIFプロセッサ210は、伸張DMAレスポンスをSTプロセッサ420へ発行する。
- [0156] S2340においてSTプロセッサ420は、DMAコントローラ220の故障か否かを判定する。故障の判定は、DMAコントローラ220からDMAコントローラ220内部エラー発生のお知らせがあるか否かで判定する。DMAコントローラ220の故障のお知らせがある場合（S2340：Y）、S2350においてSTプロセッサ420は、DMAコントローラ故障を判定し、このフローを終了する。なお、DMAコントローラ220の故障でも内部エラー発生のお知らせが発行されない場合は、STプロセッサ420はDMAコントローラ故障の有無を判定できないため、本判定では故障通知なしのフローに進む。また、故障通知は伸張DMAレスポンスにエラーとしてDMAコントローラ220の内部エラーとして含まれていても良い。
- [0157] DMAコントローラ220の故障のお知らせがないと判定された場合（S2350：N）、ストレージコントローラの制御ソフトウェアのエラーを疑い、S2360においてSTプロセッサ420は、予め設定されたエラーの要因と当該要因に対して定義される処理にのっとり、閉塞など、今後の動作を決定し、このフローを終了する。なお、再送などを行い、エラー回数が閾値を

超えた場合に閉塞としても良い。

- [0158] S 2 5 1 0 において I F プロセッサ 2 1 0 は、伸張後の非圧縮データの保障コードにエラーがあるか否かを判定する。非圧縮データの保障コードにエラーがないと判定された場合 (S 2 5 1 0 : N)、I F プロセッサ 2 1 0 は、このフローを終了する。
- [0159] 非圧縮データの保障コードにエラーがあると判定された場合 (S 2 5 1 0 : Y)、S 2 5 2 0 において I F プロセッサ 2 1 0 は、非圧縮データの保障コードでエラーが発生したことを示す情報を伸張 DMA レスポンスに設定する。S 2 5 3 0 において I F プロセッサ 2 1 0 は、伸張 DMA レスポンスを S T プロセッサ 4 2 0 へ発行する。
- [0160] S 2 5 4 0 において S T プロセッサ 4 2 0 は、DMA コントローラ 2 2 0 の故障か否かを判定する。判断は前述の圧縮データの保障コードと同様の方法で判断する。DMA コントローラ 2 2 0 の故障の通知がある場合 (S 2 5 5 0 : Y)、S 2 5 5 0 において S T プロセッサ 4 2 0 は、DMA コントローラ故障を判定し、このフローを終了する。
- [0161] DMA コントローラ 2 2 0 の故障の通知がない場合 (S 2 5 5 0 : N)、S 2 5 6 0 において S T プロセッサ 4 2 0 は、圧縮データの保障コードがあるか否かを判断する。圧縮データの保障コードがある場合 (S 2 5 6 0 : Y)、S 2 5 7 0 において S T プロセッサ 4 2 0 は、例えば記憶デバイス 6 0 0 と S T コントローラ 4 0 0 間の保障コードチェック結果と、S T コントローラ 4 0 0 と S S 接続機構 2 0 0 間の保障コードチェックよりエラー発生箇所の絞込みができるため、絞込み結果に従い、予め設定されたエラーの要因と当該要因に対して定義される処理にのっとり、閉塞など、今後の動作を決定し、このフローを終了する。圧縮データの保障コードがない場合 (S 2 5 6 0 : N)、記憶デバイス 6 0 0 と S T コントローラ 4 0 0 間の保障コードチェックがないため、記憶デバイス 6 0 0 と S T コントローラ 4 0 0 間か S T コントローラ 4 0 0 と S S 接続機構 2 0 0 間のどこでエラーが発生したかの特定が難しいため、S 2 6 1 0 において S T プロセッサ 4 2 0 は、エラー

チェック強化モードに移行する（図中のB）。なお、ここでS Tプロセッサ420は、エラーチェック強化モードに移行せずに、予め設定されたエラーの要因と当該要因に対して定義される処理にのっとり、閉塞など、今後の動作を決定し、このフローを終了してもよい。S2620においてS Tプロセッサ420は、記憶デバイス600でのエラーの可能性を判定する。S2630においてS Tプロセッサ420は、対象データの非圧縮データのサイズのうち、S T領域に収まらないサイズの領域を確保してS T領域に追加し、S T領域を示すSGLを作成する。S2640においてS Tプロセッサ420は、圧縮記憶デバイス600aに格納されている対象データの圧縮データの伸張と転送を、圧縮記憶デバイス600aに依頼し、転送経路でエラーチェックする。伸張したデータを転送することによって、伸張後のデータの保障コードで、圧縮データに保障コードをつけた際と同様のチェックが可能となり、エラー発生箇所の切り分けが可能となる。S2650においてS Tプロセッサ420は、エラー発生箇所を特定する。ここでS Tプロセッサ420は、エラー発生箇所を特定できない場合も、処理を次へ移行させる。S2660においてS Tプロセッサ420は、エラーの要因によって、閉塞など、今後の動作を決定し、このフローを終了する。

[0162] なお、エラーチェック強化モードでエラーが再現しなかった場合、S Tプロセッサ420は、予め設定されたエラーの要因と当該要因に対して定義される処理にのっとり、閉塞など、今後の動作を決定し、このフローを終了するか、閾値管理を行い閾値に達するまで、エラーチェック強化モードを解除し、伸張DMA転送で再送、または、エラーチェック強化モードで転送を実施してもよい。

[0163] 以上の保障コードチェック処理によれば、S Tプロセッサ420は、圧縮データ又は非圧縮データの保障コードのエラーに基づいて、エラー発生箇所を特定し、エラー発生箇所の閉塞などの処理を行うことができる。また、S Tプロセッサ420は、保障コードがない場合も、ある程度エラー発生箇所の絞込みを実施できる。

- [0164] STプロセッサ420が、STメモリ410のキャッシュ領域にヒット無効フラグを設定しても良い。この場合、STプロセッサ420が、シーケンシャルリードにおいて、次の圧縮データを予測して先読みすることで、圧縮記憶デバイス600aとSTメモリ410のキャッシュ領域との間のレイテンシを隠蔽することができる。
- [0165] 以下、サーバ100からの対象データを圧縮して圧縮記憶デバイス600aへ書き込むライト処理について説明する。
- [0166] 図38は、SSシステムのライト処理の中の第一処理を示し、図39は、SSシステムのライト処理の中の第一処理に続く第二処理を示し、図40は、SSシステムのライト処理の中の第二処理に続く第三処理を示す。これらの図は、SVプロセッサ120と、IFプロセッサ210と、STプロセッサ420と、SDプロセッサ612との動作を示す。ここでは、対象データの圧縮処理がSS接続機構200により行われる場合のライト処理について説明する。
- [0167] S3110においてサーバ100内のSVプロセッサ120は、SVメモリ110内のSV領域に、ライト処理の対象データを保存する。ここでSVプロセッサ120は、SV領域を示すSGLであるサーバSGLを生成してSVメモリ110へ保存する。S3120においてSVプロセッサ120は、対象データのライトリクエストをSS接続機構200へ発行する。
- [0168] S3130においてSS接続機構200内のIFプロセッサ210は、SVプロセッサ120からキューイングされたライトリクエストを解析するライトリクエスト処理を行う。S3140においてIFプロセッサ210は、解析結果に基づくライトリクエストを生成してSTプロセッサ420へ発行する。なお、IFプロセッサ210からSTコントローラ400へのライトリクエストを、SVプロセッサ120からIFプロセッサ210へライトリクエストと区別するために、ライトコマンドと呼ぶことがある。S3150においてIFプロセッサ210は、サーバSGLを取得するためのサーバSGL処理を行う。S3160においてIFプロセッサ210は、サーバ100

0からサーバSGLを受信する。S3170においてIFプロセッサ210は、受信されたサーバSGLを内部メモリに格納する。なお、サーバSGLの情報をライトリクエストに搭載してもよく、ライトリクエストにのらない分をサーバSGLにのせても良い。

[0169] S3210においてSTコントローラ400内のSTプロセッサ420は、IFプロセッサ210からキューイングされたライトリクエストを解析するライトリクエスト処理を行う。S3220においてSTプロセッサ420は、圧縮DMA処理を行うか否かを判定する（圧縮判定）。対象データを圧縮すると判定された場合、S3230においてSTプロセッサ420は、解析結果に基づいて対象データの圧縮データを格納するためのST領域をSTメモリ410内に確保する。S3240においてSTプロセッサ420は、ST領域を示す圧縮情報SGLを生成してSTメモリ410へ保存する。S3250においてSTプロセッサ420は、圧縮DMAリクエストをIFプロセッサ210へ発行する。

[0170] S3310においてIFプロセッサ210は、圧縮DMAリクエストを解析する圧縮DMAリクエスト処理を行う。S3320においてIFプロセッサ210は、STメモリ410内の圧縮情報SGLを取得する。S3330においてIFプロセッサ210は、DMAコントローラ220の圧縮DMA処理を起動する。S3340においてDMAコントローラ220は、IFプロセッサ210からの指示に基づいて圧縮DMA処理を行うことにより、サーバ100内のSV領域に格納されている非圧縮データを読み出して圧縮し、得られる圧縮データをSTコントローラ400内のST領域へ書き込む。S3350においてIFプロセッサ210は、DMAコントローラ220から圧縮DMA処理完了の通知を受ける（図中のAへ移行）。S3360においてIFプロセッサ210は、圧縮DMAリクエストに対する圧縮DMAレスポンスをSTプロセッサ420へ発行する。圧縮DMAレスポンスでは、圧縮前のデータサイズ、圧縮後のデータサイズ、圧縮単位毎の圧縮データサイズおよび、圧縮の有無を示すフラグをSTプロセッサ420に通知し、S

Tコントローラ400側で圧縮情報として管理される。

[0171] S3410においてSTプロセッサ420は、IFプロセッサ210からキューイングされた圧縮DMAレスポンスを解析する圧縮DMAレスポンス処理を行う。S3420においてSTプロセッサ420は、ST領域内の圧縮データを冗長化するデータ冗長化を行う。

[0172] 例えば、STプロセッサ420は、パリティ計算を必要としないRAID1相当のデータ冗長化として、STメモリ410内の圧縮データを、STコントローラ間バスを介して他のSTコントローラ400内のSTメモリ410へ転送することにより、二つのSTコントローラ400のSTメモリ410に圧縮データを保存する。

[0173] また、例えば、STプロセッサ420は、RAID5等、パリティ計算を必要とするデータ冗長化を行っても良い。ここで、対象データの圧縮データは、STコントローラ400内のキャッシュ領域へ転送される。ここで圧縮伸張機能付FMのキャッシュ領域、又はSTメモリ410のキャッシュ領域である。その後、STプロセッサ420は、STコントローラ400内の圧縮伸張機能を用いることにより、圧縮情報に基づいて圧縮データを伸張して非圧縮データを生成する。ここでSTコントローラ400の圧縮伸張機能は、圧縮伸張機能付FMのハードウェア、又はSTプロセッサ420のソフトウェアである。その後、STプロセッサ420は、RAIDパリティ計算を行い、生成されたパリティをキャッシュ領域へ保存する。その後、STプロセッサ420は、STコントローラ400の圧縮伸張機能を用いることにより、圧縮情報に基づいてパリティを圧縮してキャッシュ領域へ保存する。これにより、SVメモリ110と、STコントローラ400のキャッシュ領域との間のバスの帯域の使用率を低減し、ライト性能を向上させることができる。

[0174] S3430においてSTプロセッサ420は、ライトレスポンスをIFプロセッサ210へ発行し、処理をS3510（図中のB）へ移行させる。

[0175] S3440においてIFプロセッサ210は、STプロセッサ420から

キューイングされたライトレスポンスを解析するライトレスポンス処理を行う。S3450においてIFプロセッサ210は、解析結果に基づくライトレスポンスをSVプロセッサ120へ発行する。S3460においてSVプロセッサ120は、ライト処理を完了する。

[0176] S3510においてSTプロセッサ420は、ST領域内の圧縮データを圧縮記憶デバイス600aへ書き込むデステージ処理を開始する。

[0177] S3520においてSTプロセッサ420は、圧縮データに関する圧縮情報を含む圧縮情報ライトリクエストを圧縮記憶デバイス600a内のSDプロセッサ612へ発行する。S3530においてSDプロセッサ612は、STプロセッサ420からの圧縮情報をRAMへ書き込む。S3540においてSDプロセッサ612は、圧縮情報ライトリクエストに対する圧縮情報ライトレスポンスをSTプロセッサ420へ発行する。

[0178] S3610においてSTプロセッサ420は、圧縮データを書き込むための圧縮データライトリクエストをSDプロセッサ612へ発行する。S3620においてSDプロセッサ612は、圧縮データリクエストに基づいて圧縮データを格納するSD領域を、RAM内に準備する。S3630においてSDプロセッサ612は、圧縮データのライトの準備の完了を示す圧縮データライトレディをSTプロセッサ420へ発行する。

[0179] S3640においてSTプロセッサ420は、SDプロセッサ612からキューイングされた圧縮データライトレディに応じて、圧縮データをSDプロセッサ612へ送信する。S3650においてSDプロセッサ612は、圧縮データをSD領域へ書き込む。S3660においてSDプロセッサ612は、書き込みの完了に応じて、圧縮データライトリクエストに対する圧縮データライトレスポンスをSTプロセッサ420へ発行する。

[0180] 以上のライト処理によれば、STコントローラ400がSS接続機構200に非圧縮データの圧縮を行わせることにより、SS接続機構200から圧縮記憶デバイス600aまでの経路の帯域の使用量を削減でき、SSシステムのIO性能を向上させることができる。

[0181] ライト処理においてS Tプロセッサ4 2 0は、I Fプロセッサ2 1 0から対象データのライトリクエストを受信した時点（S 3 2 2 0）で、圧縮DMA処理を行うか否かを判定する。また、S Tプロセッサ4 2 0は、圧縮情報S G Lを生成した時点（S 3 2 4 0）で、圧縮DMA処理を行うか否かを判定しても良い。圧縮DMA処理を行うと判定された場合、S S接続機構2 0 0は、前述のライト処理に示したように圧縮DMA処理を行う。圧縮DMA処理を行わないと判定された場合、S S接続機構2 0 0は、対象データの非圧縮データをサーバ1 0 0からS Tコントローラ4 0 0へ転送する通常DMA処理を行う。通常DMA処理を行う場合、S Tコントローラ4 0 0が対象データの非圧縮データを圧縮記憶デバイス6 0 0 aへ転送し、圧縮記憶デバイス6 0 0 aが非圧縮データを圧縮して格納しても良いし、S Tコントローラ4 0 0が対象データを圧縮して圧縮記憶デバイス6 0 0 aへ転送し、圧縮記憶デバイス6 0 0 aが格納しても良い。以下、S Tプロセッサ4 2 0が圧縮DMA処理を行うと判定する条件の例であるW 1 ~W 1 3について説明する。

[0182] （W 1）対象データのライト先の仮想ページもしくはプール属性もしくは仮想ボリュームが圧縮属性を示す時、圧縮データをまとめて管理しないと管理負荷が増加するため、S Tプロセッサ4 2 0は、圧縮DMA処理を実施する。また、圧縮データはサイズが可変長になるため、圧縮データサイズと記憶装置5 0 0の容量の管理をしなければならないため、S Tプロセッサ4 2 0は、圧縮データを格納しているプール又はボリュームを、圧縮データのための圧縮ボリュームとして扱ったほうが管理しやすい。

[0183] （W 2）ライトリクエストが条件W 1を満たし、且つR A I D 1 0などでディスク格納の際にパリティ演算が必要ないと判断されたとき、S Tプロセッサ4 2 0は、圧縮DMA処理を実施する。パリティ演算が必要な場合は、その演算をした後に圧縮したほうが管理しやすい。例えば、圧縮データからパリティグループを作っていると、上書きライトするたびにサイズが変わるため、一旦パリティグループ内の全ての圧縮データを伸張してパリティを計算

し直さなければならなくなるためである。

- [0184] (W3) STプロセッサ420がライトリクエストのコマンド履歴をSTメモリ410に保存し、コマンド履歴に基づいてライトリクエストのシーケンシャル性を判定する。ライトリクエストがシーケンシャルライトと判定され、且つライトリクエストが条件W1又はW2を満たす場合、STプロセッサ420は、圧縮DMA処理を実施する。前述の伸張DMA処理の条件W2及びW3と同様にシーケンシャルなアクセスに対して圧縮DMA処理を実施したほうが各伝送路の帯域を使い切ることができる。
- [0185] (W4) STコントローラ400が圧縮伸張機能付きキャッシュを有し、且つ対象データがそのキャッシュへライトされる場合、STプロセッサ420は、伸張DMA処理を実施する。これは、条件R6と同様の理由による。
- [0186] (W5) 対象データが圧縮記憶デバイス600aへ直接転送される、もしくは、STメモリ410を一時テンポラリとして圧縮記憶デバイス600aへ転送されると判断されたとき、STプロセッサ420は、伸張DMA処理を実施する。これは、条件R7と同様の理由による。
- [0187] (W6) 対象データが転送元のSTコントローラ400から転送先のSTコントローラ400へ転送される場合で、且つ転送先のSTコントローラ400が圧縮伸張機能を有している場合、転送元のSTコントローラ400は、対象データを圧縮して転送する。これは、条件R9と同様の理由による。
- [0188] (W7) 対象データの圧縮データを格納する圧縮記憶デバイス600aの負荷が高いとき、STプロセッサ420は、転送時の圧縮を圧縮DMA処理へオフロードする。これは、条件R10と同様の理由による。
- [0189] (W8) STコントローラ400が圧縮伸張機能を有しており、且つそのSTコントローラ400の負荷が高いとき、STプロセッサ420は、転送時の圧縮を圧縮DMA処理へオフロードする。これは、前述の条件R10と同様の理由による。
- [0190] (W9) STプロセッサ420がサーバ100からのヒント情報に基づいて、対象データへのアクセスのシーケンシャル性を判断した場合、STプロセ

ッサ４２０は圧縮DMA処理を実施する。その他の動作は、条件W３と同じである。

[0191] (W１０) ライトリクエストが新規ライトで条件W１又はW２を満たすとき、ＳＴプロセッサ４２０は圧縮DMA処理を実施する。これは、格納済みのデータを伸張して上書きする必要がないためである。

[0192] (W１１) ライトリクエストが条件W１又はW２を満たし、且つＳＴコントローラ４００内のキャッシュを経由せずに圧縮記憶デバイス６００aへ直接転送されると判断されたとき、ＳＴプロセッサ４２０は、圧縮DMA処理を実施する。これは、ＳＴプロセッサ４２０が対象データに対し、特化パスでRAIDパリティを作ったり、上書き処理をしたりしなくて良いためである。

[0193] (W１２) 対象データの書き換え頻度が低い、もしくは対象データがアーカイブ用途のデータと判断されたとき、ＳＴプロセッサ４２０は、圧縮DMA処理を実施する。これは、対象データをキャッシュする必要がないため、キャッシュ上にあるデータを伸張して上書きしたりすることがないため、圧縮して送ってしまうことで、バスの帯域を広く使うことができる。

[0194] (W１３) ＳＴコントローラ４００内のキャッシュにダーティデータがない場合、ライトが衝突しないため、ＳＴプロセッサ４２０は、圧縮DMA処理を実施する。これは、条件W１２と同じ理由による。

[0195] (W１４) 非圧縮データをまとめた圧縮単位で圧縮し管理するため、圧縮単位以下のデータサイズのライト要求に対しては、伸張、マージ、圧縮の余分なデータを読み出し、処理が必要となるため、負荷が上がり性能が上がらない場合がある。このため、データサイズが規定値以下のとき、ＳＴプロセッサ４２０は圧縮転送を実施しない判断をしてもよい。

[0196] ここでは、ライト処理中にDMAコントローラ２２０により行われる圧縮DMA処理（Ｓ３３４０）について説明する。

[0197] 図４１は、ＳＳ接続機構２００による圧縮DMA処理の中の第一処理を示し、図４２は、ＳＳ接続機構２００による圧縮DMA処理の中の第一処理に

続く第二の処理を示す。

- [0198] SS接続機構200のDMAコントローラ220は、STコントローラ400から圧縮DMAリクエストを受信すると、圧縮DMA処理を実行する。
- [0199] S4110においてDMAコントローラ220は、圧縮DMAリクエストを解析し、転送元と転送先のアドレスを特定する。S4120においてDMAコントローラ220は、転送ブロック群サイズから先頭オフセット位置を減じた転送サイズのDMA処理を完了するまで、圧縮DMA処理の対象データの中から順に対象ブロックを選択し、S4120～S4220を繰り返す。
- [0200] S4130においてDMAコントローラ220は、対象ブロックに保障コードを付加するか否かを判定する。なお、付加するか否かは圧縮DMAリクエストでSTコントローラ400から設定する。対象ブロックに保障コードを付加しないと判定された場合（S4130：N）、DMAコントローラ220は、処理をS4150へ移行させる。対象ブロックに保障コードを付加すると判定された場合（S4130：Y）、S4140においてDMAコントローラ220は、対象ブロックの圧縮前のデータの保障コードを計算し、データに保障コードを付加し、処理をS4150へ移行させる。
- [0201] S4150においてDMAコントローラ220は、対象ブロックの圧縮処理を行うことにより圧縮データを生成する。S4160においてDMAコントローラ220は、圧縮データに保障コードを付加するか否かを判定する。圧縮データに保障コードを付加しないと判定された場合（S4160：N）、DMAコントローラ220は、処理をS4210（図中のA）へ移行させる。圧縮データに保障コードを付加すると判定された場合（S4160：Y）、S4170においてDMAコントローラ220は、圧縮データの保障コードを計算し、圧縮データに保障コードを付加し、処理をS4210（図中のA）へ移行させる。
- [0202] S4210においてDMAコントローラ220は、対象ブロックの圧縮データを転送先に格納する。S4220においてDMAコントローラ220は

、転送されたサイズだけ、対象ブロックのアドレスを増加させ、転送サイズ分の転送が完了したか否かを判定する。転送が完了していないと判定された場合（S 4 2 2 0 : N）、DMAコントローラ 2 2 0 は、処理を S 4 1 2 0（図中の B）へ移行させる。転送が完了したと判定された場合（S 4 2 2 0 : Y）、S 4 2 3 0 において DMA コントローラ 2 2 0 は、圧縮 DMA 処理完了を S T コントローラ 4 0 0 へ通知し、このフローを終了する。

[0203] 以上の圧縮 DMA 処理によれば、非圧縮データを圧縮しながら、サーバ 1 0 0 から S T コントローラ 4 0 0 へ転送することができる。また、圧縮データ及び非圧縮データに保障コードを付加することができる。

[0204] ライト処理において、S S 接続機構 2 0 0 は、対象データの非圧縮データを圧縮して圧縮データを生成し、圧縮データを S T コントローラ 4 0 0 へ転送する場合に、非圧縮データと圧縮データの何れに保障コードを付加するかを決定する。例えば、S T コントローラ 4 0 0 は、非圧縮データと圧縮データの両方に保障コードを付加することが予め設定されていても良いし、記憶領域のコストや通信のコストや記憶デバイス 6 0 0 の信頼性等に基づいて、保障コードを付加する対象を決定しても良いし、圧縮記憶デバイス 6 0 0 a が読み出された保障コードを送信しない場合に、非圧縮データだけに保障コードを付加すると判定しても良い。

[0205] ここでは、ライト処理において圧縮記憶デバイス 6 0 0 a が圧縮処理を行っている場合で、且つライトリクエストが条件 W 7 を満たす場合に、ライト処理における圧縮処理を圧縮記憶デバイス 6 0 0 a から S S 接続機構 2 0 0 へオフロードする圧縮オフロードライト処理について説明する。

[0206] 図 4 3 は、圧縮オフロードライト処理を示す。

[0207] 前述のライト処理と同様にして、S 3 2 1 0 までの処理が行われる。S 3 2 2 0 においてライトリクエストが条件 W 7 を満たすと判定された場合、S 3 2 6 0 において S T プロセッサ 4 2 0 は、ライトリクエストの解析結果に基づいて対象データの圧縮データを格納するための S T 領域を S T メモリ 4 1 0 内に確保する。S 3 2 7 0 において S T プロセッサ 4 2 0 は、転送元で

あるSV領域と、転送先であるST領域とを示す圧縮情報SGLを生成してSTメモリ410へ保存する。S3280においてSTプロセッサ420は、転送元（ライトリクエストの発行元）のサーバ100に接続されているSS接続機構200の中から圧縮DMA処理を行わせるDMAコントローラ220を圧縮DMA装置として選択する。ここでSTプロセッサ420は、各DMAコントローラ220のI/O負荷を測定してSTメモリ410へ保存し、I/O負荷が最も低いDMAコントローラ220を圧縮DMA装置として選択する。S3290においてSTプロセッサ420は、圧縮DMAリクエストを圧縮DMA装置へ発行する。なお、圧縮情報SGLの情報をライトリクエストに搭載してもよく、ライトリクエストにのらない分を圧縮情報SGLにのせても良い。

[0208] 以後の処理は前述のライト処理におけるS3310～S3410と同様である。

[0209] 圧縮オフロードライト処理と同様、リード処理において圧縮記憶デバイス600aが伸張処理を行っている場合で、且つライトリクエストが条件R10を満たすと判定された場合に、リード処理における伸張処理を圧縮記憶デバイス600aからSS接続機構200へオフロードする圧縮オフロードリード処理について説明する。圧縮オフロードリード処理においてSTプロセッサ420は、リード処理のS1220～S1620の処理中に、転送先（リードリクエストの発行元）のサーバ100に接続されているSS接続機構200の中から伸張DMA処理を行わせるDMAコントローラ220を伸張DMA装置として選択し、伸張DMAリクエストを伸張DMA装置へ発行する。

[0210] このように圧縮及び伸張の負荷を圧縮記憶デバイス600aとSS接続機構200に分散させることにより、SSシステムのI/O性能を向上させることができる。

[0211] 以下、サーバ100が圧縮伸張機能付FMを含む場合について説明する。

[0212] サーバ100に搭載された圧縮伸張機能付FMが、対象データの圧縮及び

伸張を行う場合、SVプロセッサ120は、SS接続機構200へのライトリクエスト及びリードリクエストに、サーバ100による圧縮及び伸張に関するヒント情報を付加する。STプロセッサ420は、圧縮DMA処理及び伸張DMA処理の実施可否の判定に、ヒント情報を用いる。

[0213] 例えば、SS接続機構200が、サーバ100により用いられている圧縮伸張アルゴリズムと異なる圧縮伸張アルゴリズムを用いる場合、SS接続機構200による圧縮の効果があるため、ヒント情報が圧縮伸張アルゴリズムを示すことにより、STプロセッサ420は、ヒント情報に基づいて圧縮DMA処理及び伸張DMA処理を実施しても良い。SS接続機構200が、サーバ100により用いられている圧縮伸張アルゴリズムと同じ圧縮伸張アルゴリズムを用いる場合、ヒント情報が圧縮伸張アルゴリズムを示すことにより、STプロセッサ420は、ヒント情報に基づいてSS接続機構200による圧縮の効果を判定し、効果があると判定される場合に圧縮DMA処理及び伸張DMA処理を実施しても良い。

[0214] また、STプロセッサ420が、STコントローラ400に搭載された圧縮伸張機能付FM、又はSTプロセッサ420のソフトウェアを用いて、対象データの圧縮及び伸張を行っている場合、STコントローラ400により管理されている圧縮情報を、SS接続機構200を介して、サーバ100に搭載された圧縮伸張機能付FMへ通知する。これにより、サーバ100に搭載された圧縮伸張機能付FMは、圧縮情報に基づいて対象データの圧縮及び伸張を行う。

[0215] 以下、SSシステムにおける圧縮と、暗号又は重複排除との併用について説明する。

[0216] 基本的には、ライト処理時の圧縮処理と、暗号処理又は重複排除処理とを行った順の逆順に同じ主体が、リード処理時の伸張処理と、暗号処理の逆変換である復号処理、又は重複排除処理の逆変換である復元処理とを行うことで読み出す。

[0217] ライト処理時に、圧縮記憶デバイス600aが、STコントローラ400

から受信したデータの圧縮を行った後、暗号化処理又は重複排除化処理を行う場合、リード処理時に、圧縮記憶デバイス600aは、復号処理又は復元処理を行い、圧縮されたデータをSTコントローラ400へ送信する。

[0218] 復号処理は、STプロセッサ420により行われても良い。この場合、復号処理がSTプロセッサ420から、STコントローラ400内の復号機能付FMへオフロードされても良い。また、STコントローラ400内で伸張され、処理が行われたデータは、圧縮伸張機能付FMにより再圧縮されても良い。

[0219] 通信ネットワークを介してSSシステムに接続されている管理計算機、又はサーバ100がSTコントローラ400に対しポリシーを予め設定しても良い。STプロセッサ420は、ポリシーに適合するデータをSS接続機構200によるDMA伸張処理の対象とせず、STコントローラ400に搭載された圧縮伸張機能付FM、又はSTプロセッサ420のソフトウェアを用いて、伸張した後に、通常DMA処理を行っても良い。ポリシーは、例えば、暗号化され圧縮されて記憶デバイス600に格納されているデータや、重複排除され圧縮されて記憶デバイス600に格納されているデータを指す。

[0220] 以下、SSシステムが重複排除と圧縮を行う場合について説明する。

[0221] SVプロセッサ120が重複排除を行う場合、ライト処理においてSS接続機構200が、重複排除されたデータを、前述の圧縮DMA処理と同様に、圧縮してSTコントローラ400へ転送しても良い。更にSS接続機構200は、リード処理において前述の伸張DMA処理と同様に、STコントローラ400からのデータを伸張しても良い。この場合、SVプロセッサ120は、データがSS接続機構200からSVメモリ110へ転送された後、重複排除されたデータを元のデータに復元する。

[0222] STプロセッサ420が重複排除を行う場合、ライト処理において圧縮記憶デバイス600aが、重複排除されたデータを圧縮しても良い。リード処理において圧縮記憶デバイス600aは、圧縮データのままSTコントローラ400へ転送する。STプロセッサ420は、圧縮伸張機能付FM等を用

いて圧縮データを伸張り、重複排除されたデータを復元し、復元されたデータを圧縮伸張機能付FM等を用いて圧縮し、圧縮されたデータに対して伸張DMA処理を行う。

[0223] STプロセッサ420が重複排除を行う場合、ライト処理において、圧縮記憶デバイス600a又はSTコントローラ400が対象データを圧縮し、STプロセッサ420が圧縮されたデータの重複排除を実施しても良い。リード処理においてSTプロセッサ420は、記憶デバイス600からデータを読み出し、読み出されたデータの重複排除を解除し、解除されたデータに対して伸張DMA処理を行う。

[0224] この動作によれば、圧縮だけを行う場合に比べて、サーバ側バスの帯域の使用量を削減でき、他の通信の帯域の使用量を更に削減できる。

[0225] なお、SSシステムは、前述の圧縮及び伸張に代えて、重複排除及び重複排除の解除を行ってもよい。この場合も、圧縮及び伸張を行う場合と同様、サーバ側バスの帯域の使用量、ストレージシステム内の通信の帯域の使用量、記憶装置の記憶容量の使用量を削減できる。

[0226] 以下、SSシステムが暗号化と圧縮を行う場合について説明する。

[0227] 圧縮記憶デバイス600aが圧縮と暗号化を行っても良い。ライト処理において圧縮記憶デバイス600aは、記憶容量削減とデータ保護のために、対象データを圧縮し、圧縮されたデータを暗号化して格納する。リード処理において圧縮記憶デバイス600aは、格納されたデータを復号し、復号されたデータをSTコントローラ400へ転送する(S1530)。暗号化されたデータを圧縮すると、圧縮率が悪化するため、圧縮記憶デバイス600aが圧縮機能と暗号化機能に対応する場合は、圧縮してから暗号化することが望ましい。圧縮機能と暗号化機能に対応する場合に、SS接続機構200との伸張DMA転送時は、圧縮記憶デバイス600a内で暗号化されたデータを復号後、圧縮されたままデータをSTメモリ410へ転送し、SS接続機構200が伸張しながらデータをサーバメモリへ転送する。

[0228] 記憶デバイス600本体及び、記憶デバイス600からSTコントローラ

400間のパスのデータ保護の観点から、データが記憶デバイス600からSTコントローラ400まで暗号化されたまま転送されても良い。リード処理においてSTコントローラ400は、圧縮記憶デバイス600aから圧縮され暗号化されたデータを読み出し、読み出されたデータを復号し(S1610)、復号されたデータに対して伸張DMA処理を行う。STコントローラ400において、STプロセッサ420が復号を行っても良いし、STコントローラ400が暗号復号専用ハードウェアを有し、当該ハードウェアが復号を行っても良いし、STコントローラ400が暗号復号機能付FMを有し、当該FMが復号を行っても良い。

[0229] ストレージシステム300内のデータ保護の観点から、データが記憶デバイス600からサーバ100まで暗号化されたまま転送されても良い。サーバ100で復号することで、ストレージシステム300内のデータの格納場所、ストレージシステム300内のデータのパスより、データの漏洩を防ぎ、データを保護する。リード処理においてSVプロセッサ120は、STコントローラ400からSVメモリ110へ暗号化されたまま転送されたデータを復号し、復号の完了をSTプロセッサ420へ通知する。通知に応じてSTプロセッサ420が、SS接続機構200に対してSVメモリ110内の記憶領域間での伸張DMA処理を指示することにより、SS接続機構200はSVメモリ110内のデータを伸張する。なお、SS接続機構200が復号機能を有してもよく、この場合は、STメモリ410に格納された圧縮されて暗号化されたデータを伸張DMA転送で読み出し、復号して、伸張処理を行った後にSVメモリ110へ転送してもよい。

[0230] サーバ100が圧縮伸張のための圧縮伸張回路を含み、サーバ100が対象データの圧縮及び伸張の処理を担当する場合、SVプロセッサ120は、SS接続機構200及びSTコントローラ400による圧縮及び伸張が不要であることを示す識別子を、リードコマンド及びライトコマンドに付与する。この場合、STプロセッサ420は、サーバ100による圧縮及び伸張に必要な情報をSVプロセッサ120へ送信する。

- [0231] この動作によれば、データに対して圧縮と暗号化を行うことにより、SSシステムのIO性能を向上させると共に、SSシステムのセキュリティを向上させることができる。
- [0232] ここでは、サーバ100が復号回路を含む場合で、復号回路が復号を行い、SS接続機構200が伸張を行う場合について説明する。復号回路は、復号機能付FMであっても良いし、復号機能を有する圧縮伸張機能付FMであっても良い。
- [0233] 図44は、サーバ100による復号とSS接続機構200による伸張を行う場合のリード処理の中の第一処理を示し、図45は、サーバ100による復号とSS接続機構200による伸張を行う場合のリード処理の中の第一処理の後の第二処理を示す。
- [0234] 前述のリード処理と同様にして、S1610までの処理が行われる。但し、STメモリ410には、圧縮データに代えて、圧縮後に暗号化された圧縮後暗号化データが格納されている。S5620においてSTプロセッサ420は、サーバ転送後伸張DMA処理を要求するサーバ転送後伸張DMAリクエストをIFプロセッサ210へ発行する。サーバ転送後伸張DMA処理は、通常DMA処理と、転送後伸張DMA処理とを含む。ここでの通常DMA処理は、STコントローラ400内のSTメモリ410から対象データの圧縮後暗号化データを読み出し、サーバ100内のSVメモリ110へ当該圧縮後暗号化データを書き込む。転送後伸張DMA処理は、サーバ100内のSVメモリ110から対象データの圧縮データを読み出し、当該圧縮データを伸張して非圧縮データを生成し、サーバ100内のSVメモリ110へ当該非圧縮データを書き込む。
- [0235] S5710においてIFプロセッサ210は、STプロセッサ420からキューイングされた転送後伸張DMAリクエストを解析するDMAリクエスト処理を行う。S5720においてIFプロセッサ210は、STメモリ410内の圧縮情報SGLを取得する。S5730においてIFプロセッサ210は、DMAコントローラ220の通常DMA処理を起動する。S574

0においてDMAコントローラ220は、IFプロセッサ210からの指示に基づいて通常DMA処理を行うことにより、STコントローラ400内のST領域に格納されている圧縮後暗号化データを読み出し、サーバ100内のSV領域へ書き込む。S5750においてIFプロセッサ210は、DMAコントローラ220から通常DMA処理完了の通知を受ける。S5760においてIFプロセッサ210は、オートレスポンスを用いてリードレスポンスをSVプロセッサ120へ発行する。

[0236] S5810においてSVプロセッサ120は、IFプロセッサ210からキューイングされたリードレスポンスに応じて、処理をS6110（図中のB）へ移行させる。

[0237] S6110においてSVプロセッサ120は、復号回路を用いて圧縮後暗号化データを復号する復号処理を行うことにより、圧縮後暗号化データを圧縮データに変換し、圧縮データをSVメモリ110内のSVR領域へ保存する。S6120においてSVプロセッサ120は、転送後DMA処理を要求する転送後伸張DMAリクエストをIFプロセッサ210へ発行する。

[0238] S6130においてIFプロセッサ210は、SVプロセッサ120からキューイングされた転送後伸張DMAリクエストを解析する転送後伸張DMAリクエスト処理を行う。S6720においてIFプロセッサ210は、STメモリ410内の圧縮情報SGLを取得する。S6730においてIFプロセッサ210は、DMAコントローラ220の伸張DMA処理を起動する。S6740においてDMAコントローラ220は、IFプロセッサ210からの指示に基づいて伸張DMA処理を行うことにより、サーバ100内のSVR領域に格納されている圧縮データを読み出して伸張し、得られる非圧縮データをサーバ100内のSV領域へ書き込む。S6750においてIFプロセッサ210は、DMAコントローラ220から伸張DMA処理完了の通知を受ける。S6760においてIFプロセッサ210は、転送後伸張DMAリクエストに対する転送後伸張DMAレスポンスをSVプロセッサ120へ発行する。

- [0239] S 6 8 1 0においてS Vプロセッサ1 2 0は、I Fプロセッサ2 1 0からキューイングされたリードレスポンスに応じてリード処理を完了する。
- [0240] S 6 7 7 0においてI Fプロセッサ2 1 0は、サーバ転送後伸張DMAリクエストに対するサーバ転送後伸張DMAレスポンスをS Tプロセッサ4 2 0へ発行する。S 6 9 1 0においてS Tプロセッサ4 2 0は、I Fプロセッサ2 1 0からキューイングされたサーバ転送後伸張DMAレスポンスを解析するサーバ転送後伸張DMAレスポンス処理を行う。S 6 9 2 0においてS Tプロセッサ4 2 0は、S T領域を解放する。S 6 9 3 0においてS Tプロセッサ4 2 0は、リード処理を完了する。
- [0241] 以上がサーバ1 0 0による復号とS S接続機構2 0 0による伸張を行う場合のリード処理である。
- [0242] この動作によれば、リード処理において圧縮記憶デバイス6 0 0 aからS S接続機構2 0 0までの経路においてデータが暗号化され、セキュリティを向上させることができる。また、サーバ1 0 0が復号処理を行い、S S接続機構2 0 0が伸張処理を行うことにより、負荷を分散させることができる。なお、復号処理もS S接続機構2 0 0が実施しても良い。
- [0243] 以上の各実施例によれば、サーバ1 0 0がストレージシステム3 0 0からデータをリードする場合に、サーバ1 0 0及びストレージシステム3 0 0に接続されたインタフェースデバイス（中継デバイス、S S接続機構2 0 0）がS Tコントローラ4 0 0から圧縮データを読み出すことにより、ストレージシステム3 0 0が有する物理的な帯域の使用量を低減することができる。また、サーバ1 0 0がストレージシステム3 0 0へデータをライトする場合に、インタフェースデバイスが圧縮データをS Tコントローラ4 0 0へ書き込むことにより、ストレージシステム3 0 0が有する物理的な帯域の使用量を低減することができる。これにより、サーバ1 0 0からストレージシステム3 0 0へのアクセスにおけるレスポンスタイムを低減することができる。
- [0244] 以上、サーバ1 0 0とストレージシステム3 0 0間のS S接続機構2 0 0を介するデータ転送について示したが、図4 6に示されるように、S S接続

機構 200 はサーバ 100 とサーバ 100 の間を接続し（サーバ・サーバ間接続）、データ転送に用いても良い。この場合、SVメモリ 110 またはサーバ 100 に搭載されたフラッシュメモリ 140 に格納されている圧縮データをもう一方のサーバ 100 の SVメモリ 110 またはフラッシュメモリ 140 に転送する際に、転送先の SS 接続機構 200 で前述した伸張 DMA 転送と同様の手順で伸張 DMA 転送する。これにより、サーバ・サーバ間の転送を広帯域化でき、クラスタ間転送時など、高速に実施できる。

[0245] また、図 47 に示されるように、ストレージシステム 300a とストレージシステム 300a の間を SS 接続機構 200 を介して接続し（ストレージ・ストレージ間接続）、データ転送に用いても良い。この場合ストレージシステム 300a の記憶デバイスまたは STメモリ 410 または STコントローラ 400a に搭載されたストレージ搭載フラッシュメモリ 440 に格納されている圧縮データをもう一方のストレージシステム 300a の記憶デバイス 600 または STメモリ 410 またはストレージ搭載フラッシュメモリ 440 に転送する際に、転送先の SS 接続機構 200 で前述した伸張 DMA 転送と同様の手順で伸張 DMA 転送する。これによりストレージシステム 400a とストレージシステム 300a の間の転送を広帯域化でき、外接ストレージからのデータ読み出し転送や、アーカイブストレージからのデータ読み出しを高速に実施できる。

[0246] また、前述のサーバ・サーバ間接続、ストレージ・ストレージ間接続に SS 接続機構 200 を用いる方法において、圧縮 DMA 転送を実施してもよい。

[0247] 本発明は、以上の実施例に限定されるものでなく、その趣旨から逸脱しない範囲で、他の様々な形に変更することができる。

[0248] 例えば計算機システムにおけるプロセッサとメモリの間の I/F に SS 接続機構 200 を搭載し、メモリに格納された圧縮データを同様の方法で SS 接続機構 200 が読み出しプロセッサに伸張してわたしても良い。

計算機システムは、SSシステム等を含んでも良い。計算機システムは、ストレージモジュールとして、ストレージシステム300等を含んでも良い。計算機システムは、サーバモジュールとして、サーバ100等を含んでも良い。計算機システムは、インタフェースデバイスとして、SS接続機構200や中継デバイス等を含んでも良い。計算機システムは、記憶デバイスとして、圧縮記憶デバイス600a等を含んでも良い。記憶デバイスは、コントローラとして、SDコントローラ610等を含んでも良い。コントローラは、逆変換機能として、圧縮・伸張デバイス614等を含んでも良い。記憶デバイスは、メモリとして、FM620等を含んでも良い。計算機システムは、変換機能として、圧縮デバイス221等を含んでも良い。計算機システムは、逆変換機能として、伸張デバイス222等を含んでも良い。計算機システムは、変換回路として、圧縮・伸張デバイス614、圧縮伸張機能付FM、STプロセッサ420等を含んでも良い。計算機システムは、例えば、変換として圧縮を用い逆変換として伸張を用いても良い。計算機システムは、例えば、変換として重複排除を用いて、逆変換として重複排除の復元を用いても良い。計算機システムは、変換後対象データとして圧縮データ等を用いても良い。計算機システムは、逆変換結果データとして非圧縮データ等を用いても良い。計算機システムは、逆変換指示として伸張DMAリクエスト等を用いても良い。計算機システムは、転送元アドレスとして、圧縮情報SGL等を用いても良い。インタフェースデバイスからストレージコントローラへのリードコマンドに対し、ストレージコントローラからインタフェースデバイスへの返信として、伸張DMAリクエスト及び圧縮情報SGL等が用いられても良い。計算機システムは、変換指示として圧縮DMAリクエスト等を用いても良い。計算機システムは、読み出し指示として通常DMAレスポンス等を用いても良い。計算機システムは、逆変換が成功である旨を示す応答として伸張DMAレスポンス等を用いても良い。計算機システムは、第一保障コードとして、非圧縮データの保証コード等を用い、第一保障コードの期待値として、非圧縮保証コード情報等を用いても良い。計算機システム

は、第二保障コードとして、圧縮データの保証コード等を用い、第二保障コードの期待値として、圧縮保証コード情報等を用いても良い。サーバプロセッサとインタフェースデバイスとを接続するバスとして、サーバ側バス等が用いられても良い。インタフェースデバイスとストレージプロセッサとを接続するバスとして、ストレージ側バス等が用いられても良い。

符号の説明

[0249] 1 0 0…サーバ 1 0 0 1 1 0…SVメモリ 1 2 0…SVプロセッサ
 2 0 0…SS接続機構 2 1 0…IFプロセッサ 2 2 0…DMAコントロ
 ーラ 3 0 0…ストレージシステム 4 0 0…STコントローラ 4 1 0…
 STメモリ 4 2 0…STプロセッサ 4 3 0…ディスクI/F 5 0 0…
 記憶装置 6 0 0…記憶デバイス 6 1 0…SDコントローラ 6 2 0…F
 M 7 0 0…インタフェース拡張装置

請求の範囲

- [請求項1] ストレージプロセッサとストレージメモリと記憶デバイスとを有するストレージモジュールと、
- サーバプロセッサとサーバメモリとを有するサーバモジュールと、
- 前記サーバプロセッサと前記ストレージプロセッサとに接続されるインタフェースデバイスとを有し、
- 前記インタフェースデバイスは、
- 前記サーバモジュール内に位置し、
- 前記サーバプロセッサにより発行されたリードリクエストを受信すると、前記リードリクエストに基づくリードコマンドを、前記ストレージプロセッサへ送信し、
- 前記ストレージプロセッサから、前記リードリクエストの対象データを変換して得られた変換後対象データの逆変換を前記インタフェースデバイスに行わせるとの逆変換指示を受けた場合に、前記ストレージメモリ上の転送元アドレスに格納された前記変換後対象データを逆変換しながら前記サーバメモリ上の転送先アドレスに転送するDMA (Direct Memory Access) を行う
- 計算機システム。
- [請求項2] 請求項1記載の計算機システムであって、
- 前記ストレージプロセッサは、
- 前記リードリクエストの対象データがデータ量の削減された変換後対象データとして前記記憶デバイスに格納されている場合に、前記インタフェースデバイスに前記逆変換を行わせるか判定し、
- 前記逆変換を行わせると判定した場合に、前記リードコマンドの返信として、前記ストレージメモリ上の前記転送元アドレスと、前記逆変換指示とを、前記インタフェースデバイスに送信する
- 計算機システム。

- [請求項3] 請求項1記載の計算機システムであって、
前記変換は圧縮処理であり、前記逆変換は伸張処理である
計算機システム。
- [請求項4] 請求項2記載の計算機システムであって、
前記ストレージプロセッサは、前記対象データのサイズが所定値以上の場合に、前記インタフェースデバイスに前記逆変換を行わせると
判定する
計算機システム。
- [請求項5] 請求項1記載の計算機システムであって、
前記ストレージプロセッサは、前記ストレージメモリ上の前記転送元アドレスに格納された前記変換後対象データのサイズと前記変換後対象データの
前記変換前のサイズに関する情報も、前記インタフェースデバイスに送信する
計算機システム。
- [請求項6] 請求項5記載の計算機システムであって、
前記記憶デバイスは、逆変換機能を有するコントローラと複数のメモリとを有し、
前記ストレージプロセッサは、前記インタフェースデバイスに前記逆変換を行わせると判定した場合に、前記サイズに関する情報と前記対象データの
前記変換後対象データとを、前記ストレージメモリ上の前記転送元アドレスに読み出す指示である読み出し指示を前記コントローラに行い、
前記読み出し指示は、前記逆変換を行わずに前記対象データを読み出す指示である

計算機システム。

[請求項7]

請求項5記載の計算機システムであって、

前記ストレージプロセッサは、前記インタフェースデバイスに前記逆変換を行わせないと判定した場合に、前記逆変換機能を有するコントローラに、前記リードリクエストの対象データの読み出しを指示し、

前記コントローラは記憶される前記変換後対象データを前記逆変換して前記ストレージメモリ上の前記転送元アドレスに読み出す計算機システム。

[請求項8]

請求項1記載の計算機システムであって、

前記インタフェースデバイスは、前記対象データを所定サイズのブロック毎に前記逆変換することにより逆変換結果データを生成し、前記逆変換結果データが前記リードリクエストの対象データ以外のデータを含む場合、前記逆変換結果データから前記対象データを抽出して前記サーバメモリ上の転送先アドレスに転送する計算機システム。

[請求項9]

請求項1記載の計算機システムであって、

前記逆変換指示は、前記対象データのサイズと、前記対象データに付加された第一保障コードの期待値とを含み、

前記インタフェースデバイスは、前記逆変換された逆変換結果データと前記第一保障コードの期待値とに基づいて、前記逆変換が成功であるか否かを判定し、前記逆変換が成功である旨の応答を前記ストレージプロセッサへ通知する計算機システム。

- [請求項10] 請求項9記載の計算機システムであって、
前記逆変換指示は、前記変換後対象データに付加された第二保障コードの期待値を含み、
前記インタフェースデバイスは、前記読み出された変換後対象データと前記第二保障コードの期待値とに基づいて、前記変換後対象データが正常であるか否かを判定し、前記変換後対象データが正常であると判定された場合に前記逆変換が成功であるか否かを判定する計算機システム。
- [請求項11] 請求項1記載の計算機システムであって、
前記サーバプロセッサと前記インタフェースデバイスとを接続するバスによって、前記インタフェースデバイスと前記ストレージプロセッサは接続される計算機システム。
- [請求項12] 請求項1記載の計算機システムであって、
前記インタフェースデバイスは、前記サーバプロセッサにより発行されたライトリクエストを受信し、前記ストレージプロセッサから、前記ライトリクエストのライト対象データの変換を前記インタフェースデバイスに行わせるとの変換指示をうけた場合に、
前記サーバメモリ上に格納された前記ライト対象データを前記変換しながら前記ストレージメモリ上のアドレスに転送するDMAを行う計算機システム。
- [請求項13] ストレージプロセッサとストレージメモリと記憶デバイスとを有するストレージモジュールと、
サーバプロセッサとサーバメモリとを有するサーバモジュールと、
を有する計算機システムを制御する計算機システム制御方法であって

、

前記サーバプロセッサと前記ストレージプロセッサとに接続され前記サーバモジュール内に位置するインタフェースデバイスを用いて、前記サーバプロセッサにより発行されたリードリクエストを受信すると、前記リードリクエストに基づくリードコマンドを、前記ストレージプロセッサへ送信し、

前記インタフェースデバイスを用いて、前記ストレージプロセッサから、前記リードリクエストの対象データを変換して得られた変換後対象データの逆変換を前記インタフェースデバイスに行わせるとの逆変換指示を受けた場合に、前記ストレージメモリ上の転送元アドレスに格納された前記変換データを逆変換しながら前記サーバメモリ上の転送先アドレスに転送するDMA (Direct Memory Access) を行うことを備える計算機システム制御方法。

[請求項14]

請求項13記載の計算機システム制御方法であって、

前記ストレージプロセッサを用いて、前記リードリクエストの対象データがデータ量の削減された変換後対象データとして前記記憶デバイスに格納されている場合に、前記インタフェースデバイスに前記逆変換を行わせるか判定し、

前記ストレージプロセッサを用いて、前記逆変換を行わせると判定した場合に、前記リードコマンドの送信の返信として、前記ストレージメモリ上の前記転送元アドレスと、前記逆変換指示とを、前記インタフェースデバイスに送信することを更に備える計算機システム制御方法。

[請求項15]

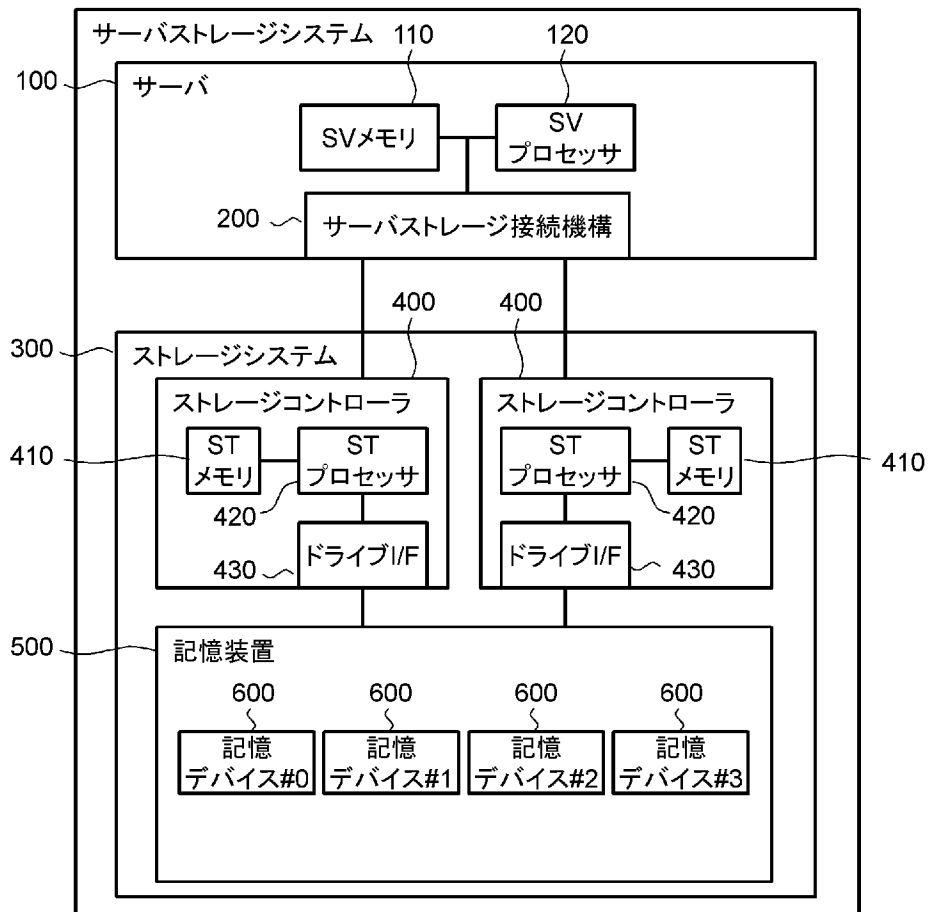
請求項13記載の計算機システム制御方法であって、

前記変換は圧縮処理であり、前記逆変換は伸張処理である計算機システム制御方法。

BLANK PAGE RECEIVED AT IB

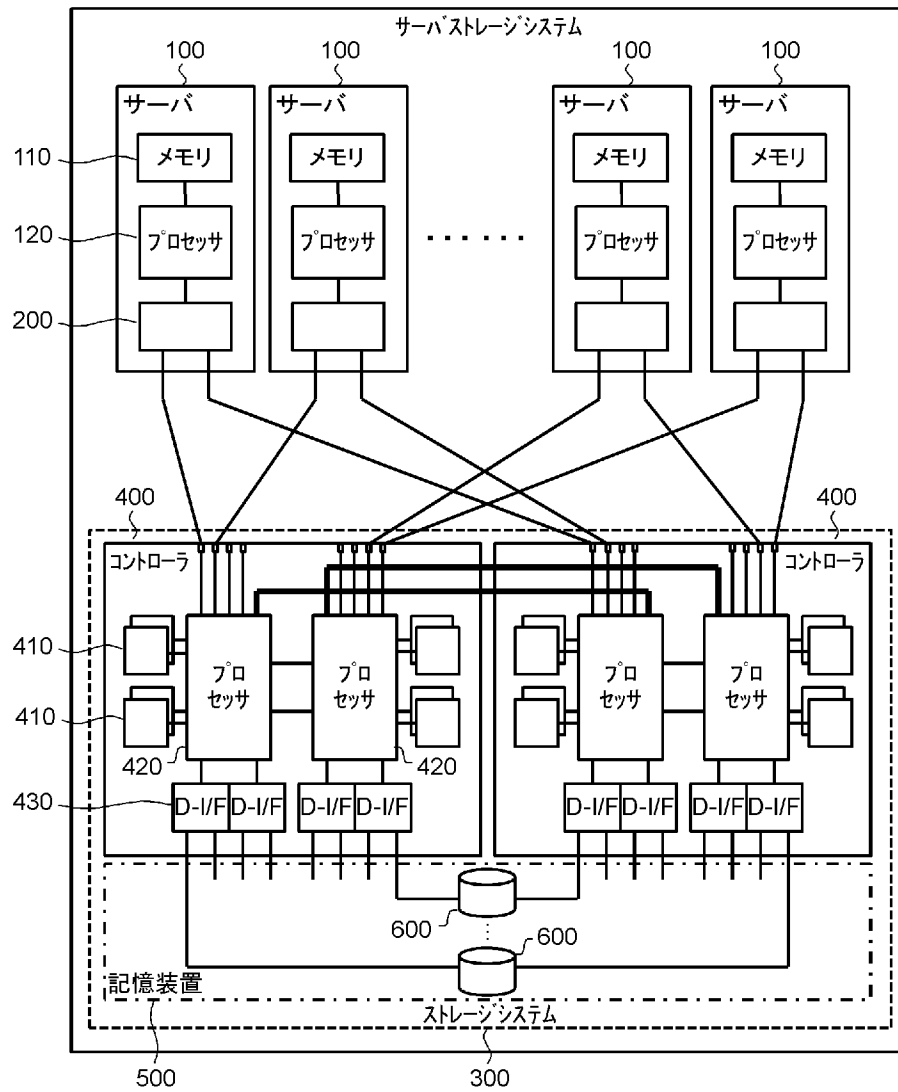
[図1]

Fig. 1



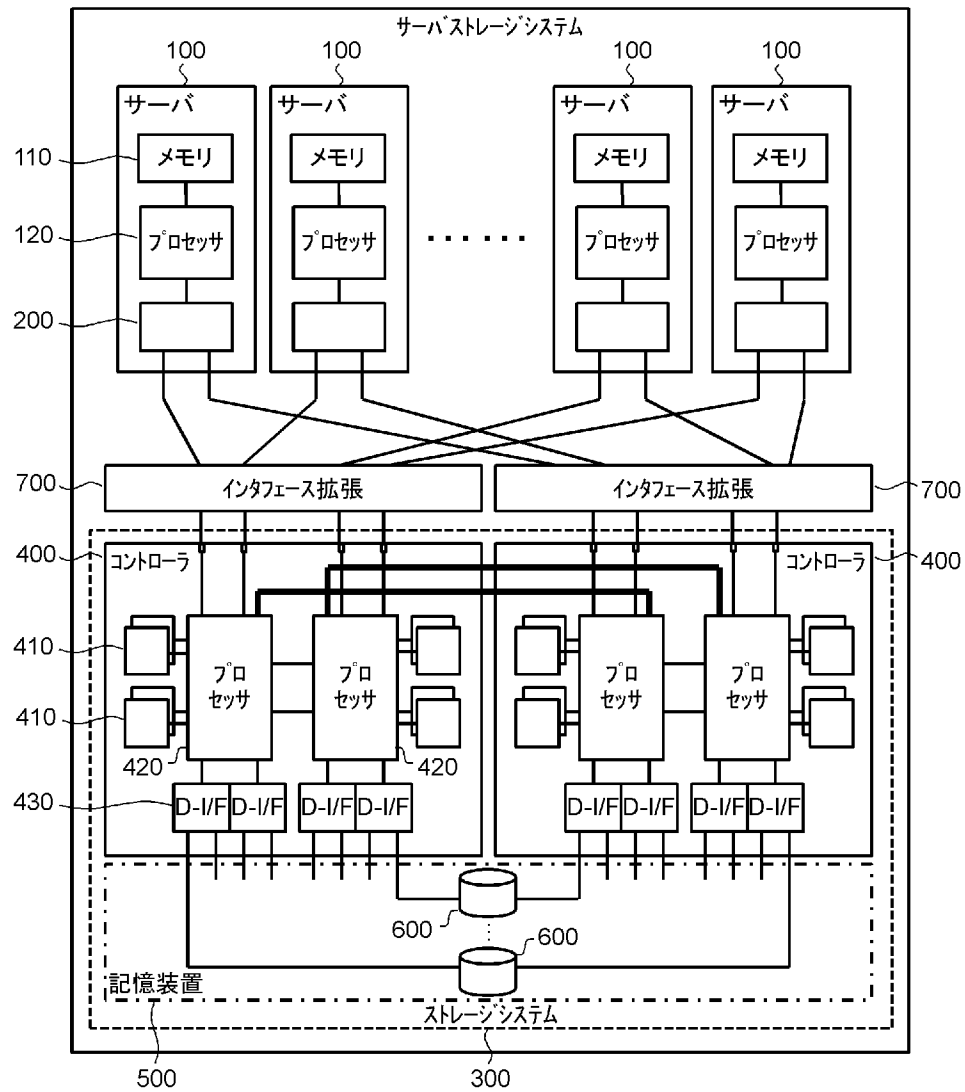
[図2]

Fig. 2



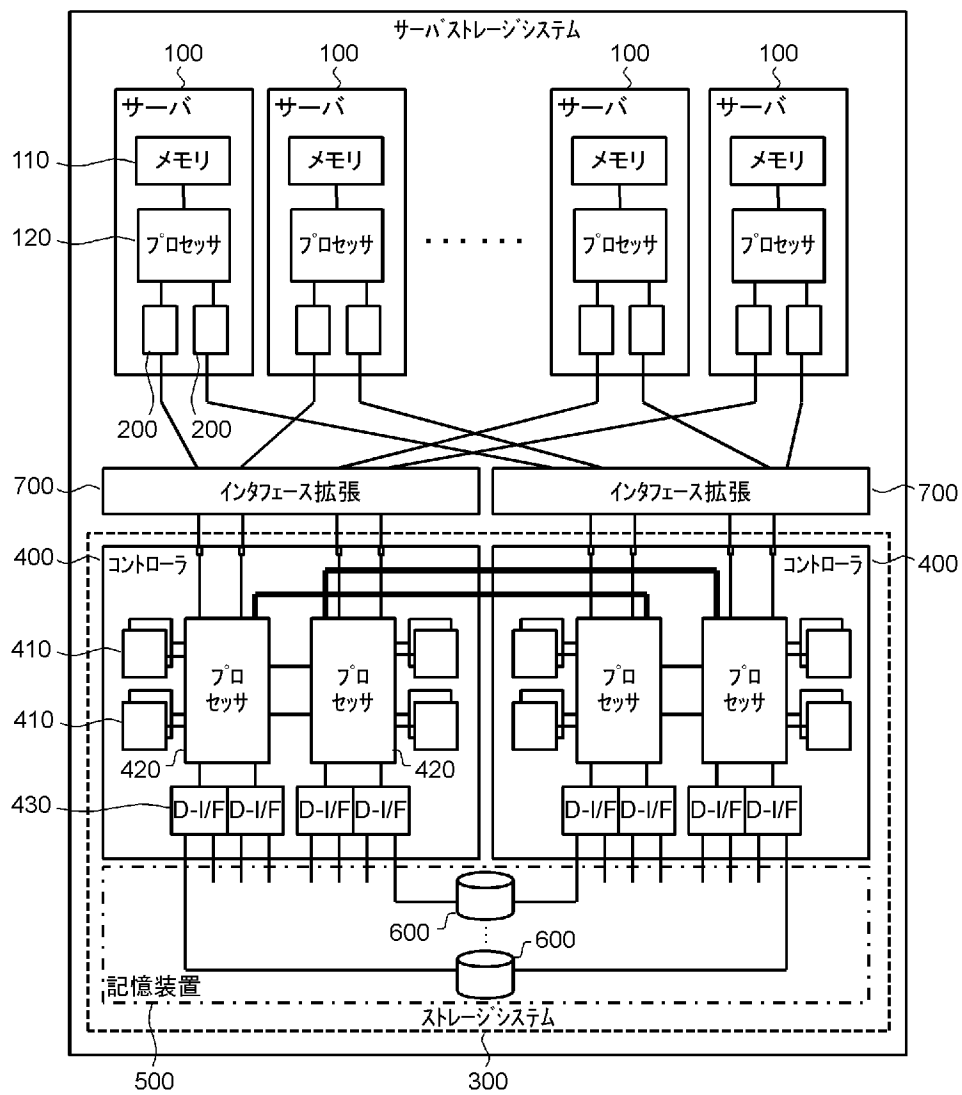
[図3]

Fig. 3



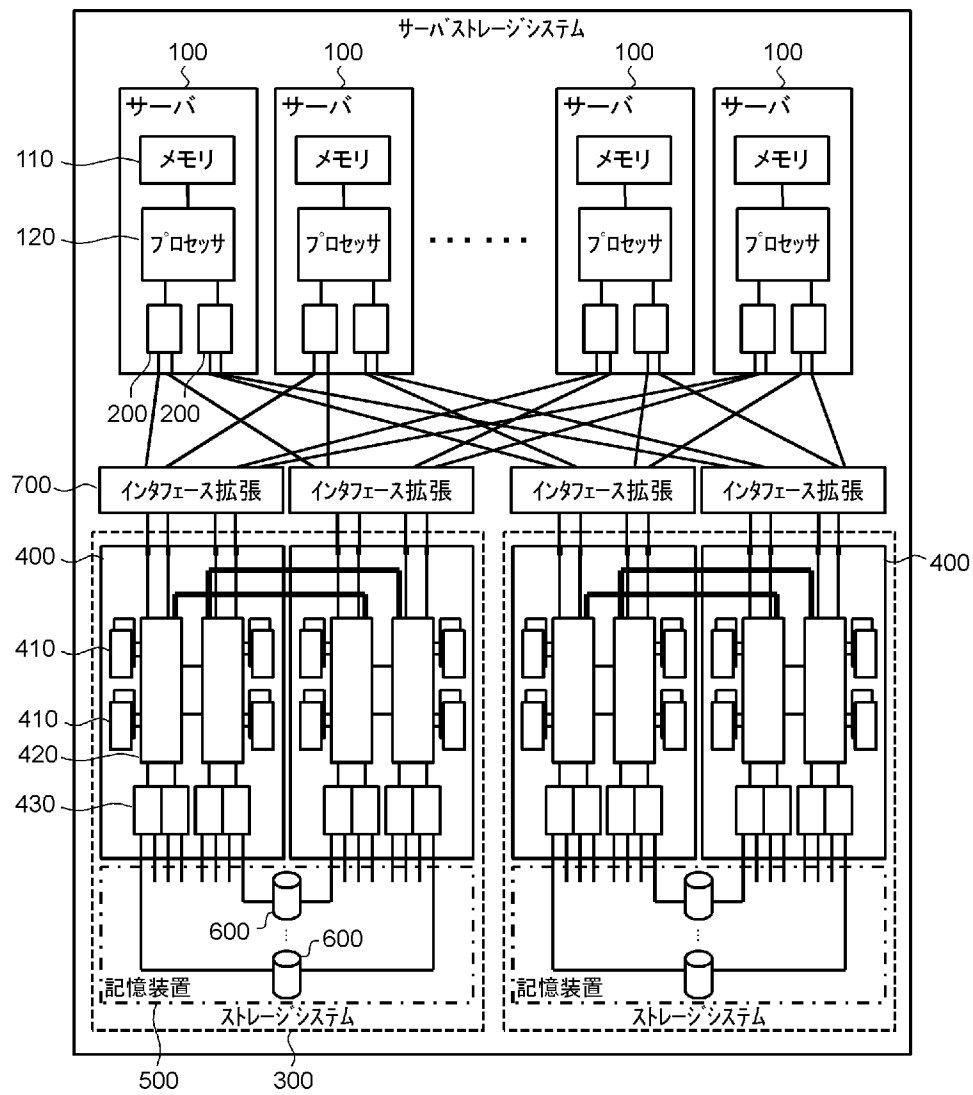
[図4]

Fig. 4



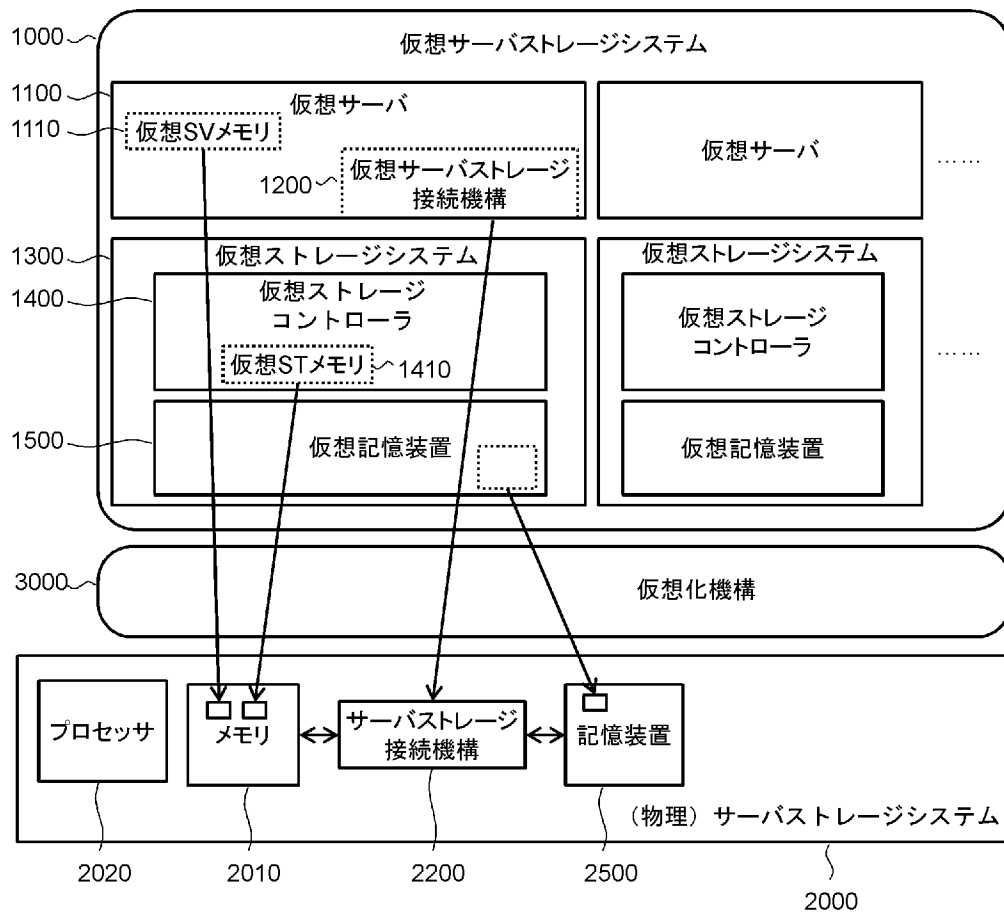
[図5]

Fig. 5



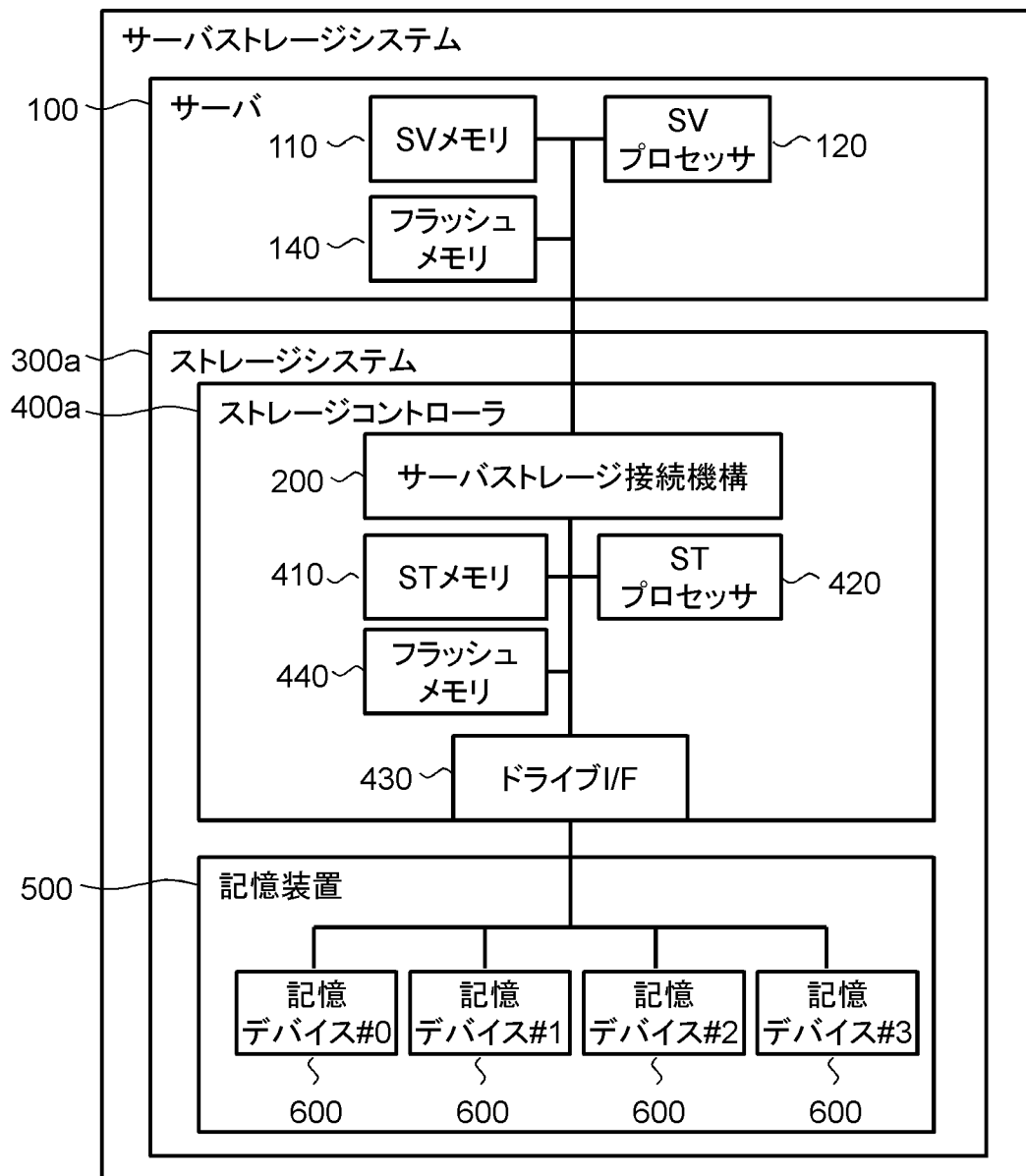
[図6]

Fig. 6



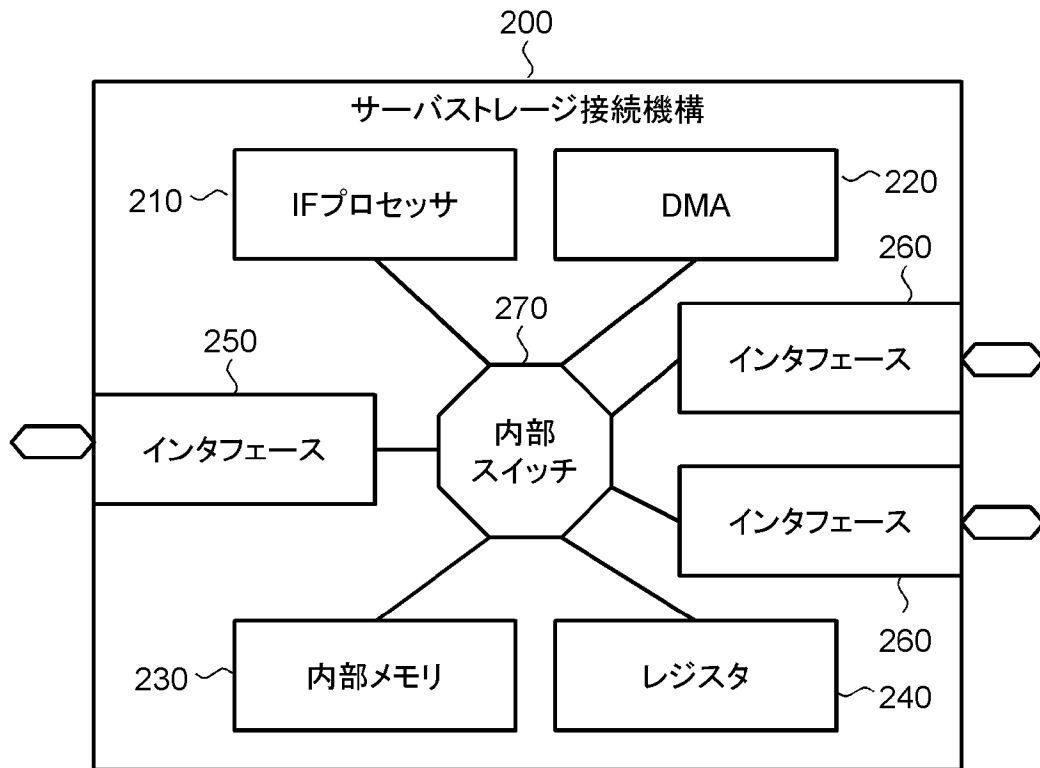
[図7]

Fig. 7



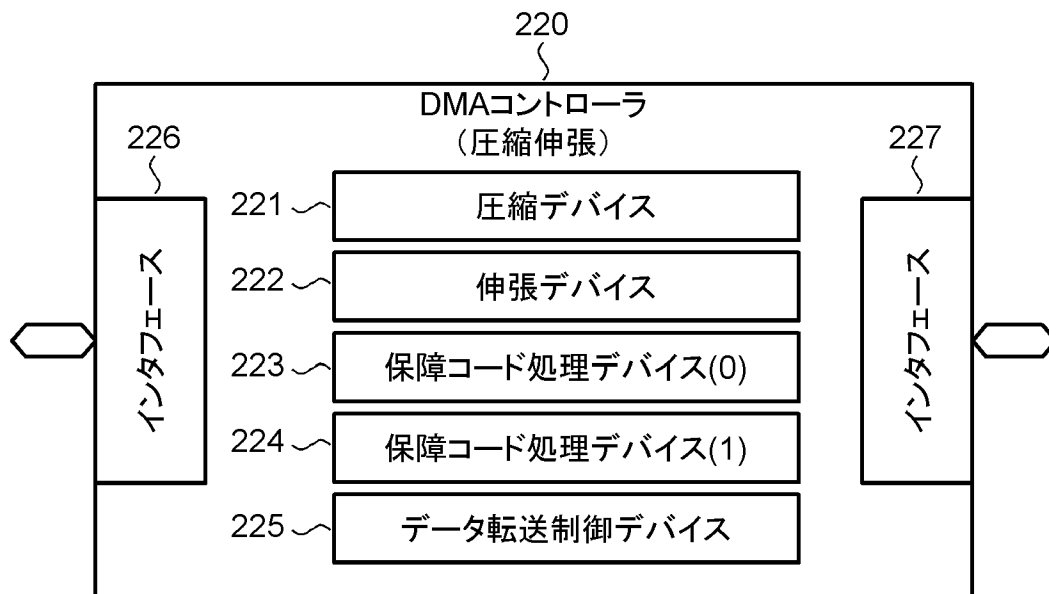
[図8]

Fig. 8



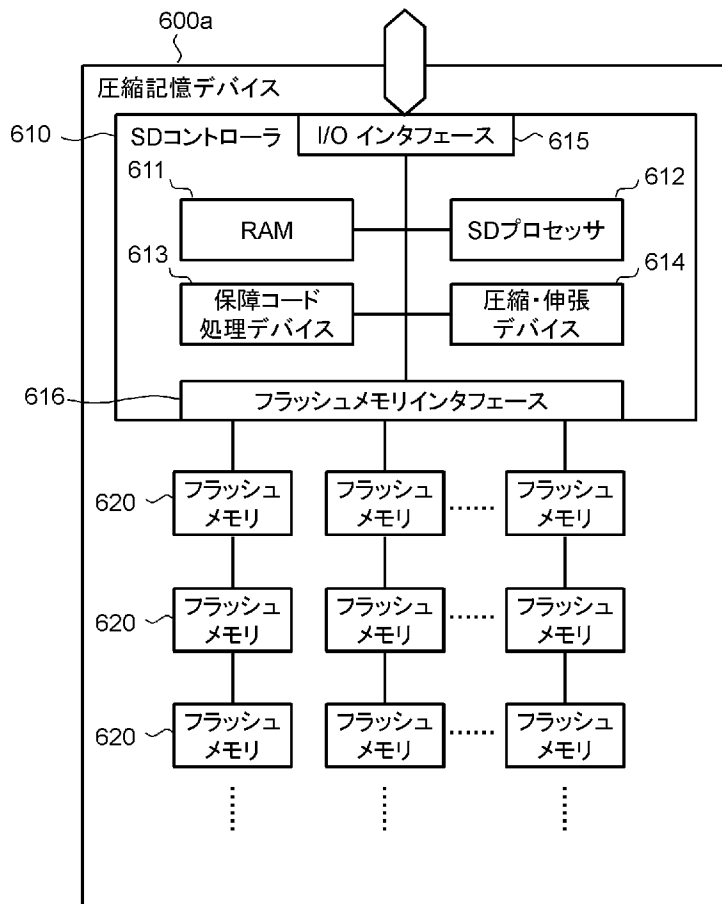
[図9]

Fig. 9



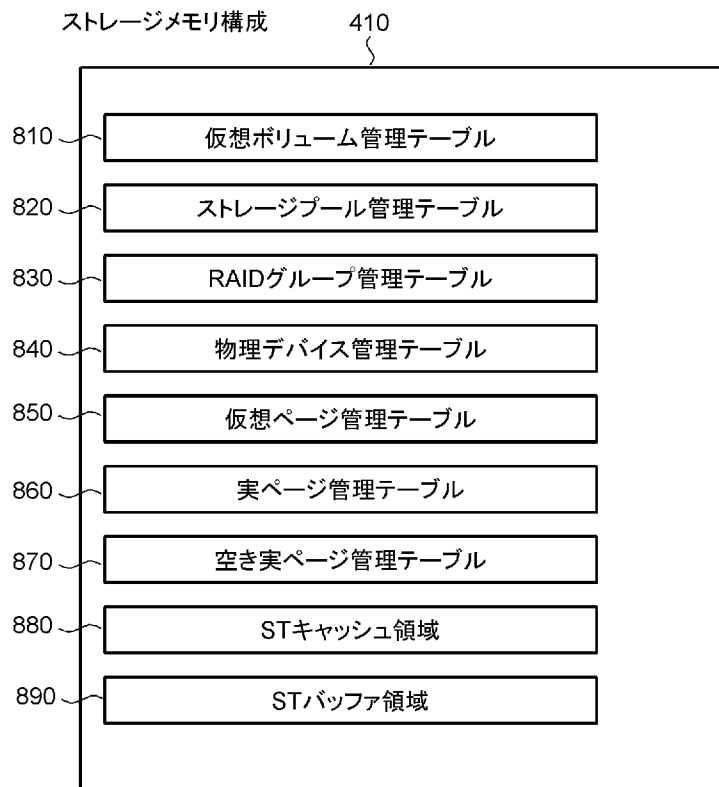
[図10]

Fig. 10



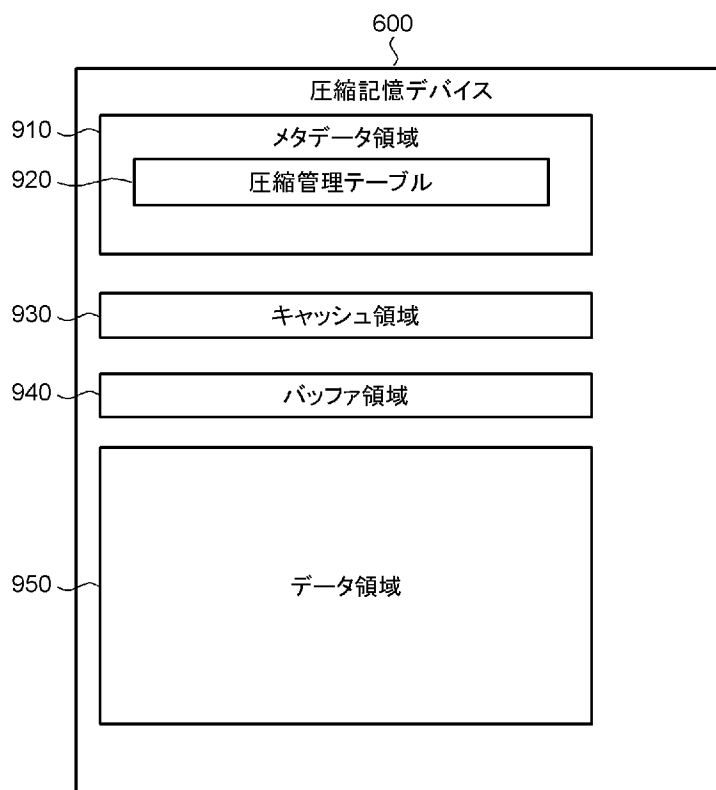
[図11]

Fig. 11



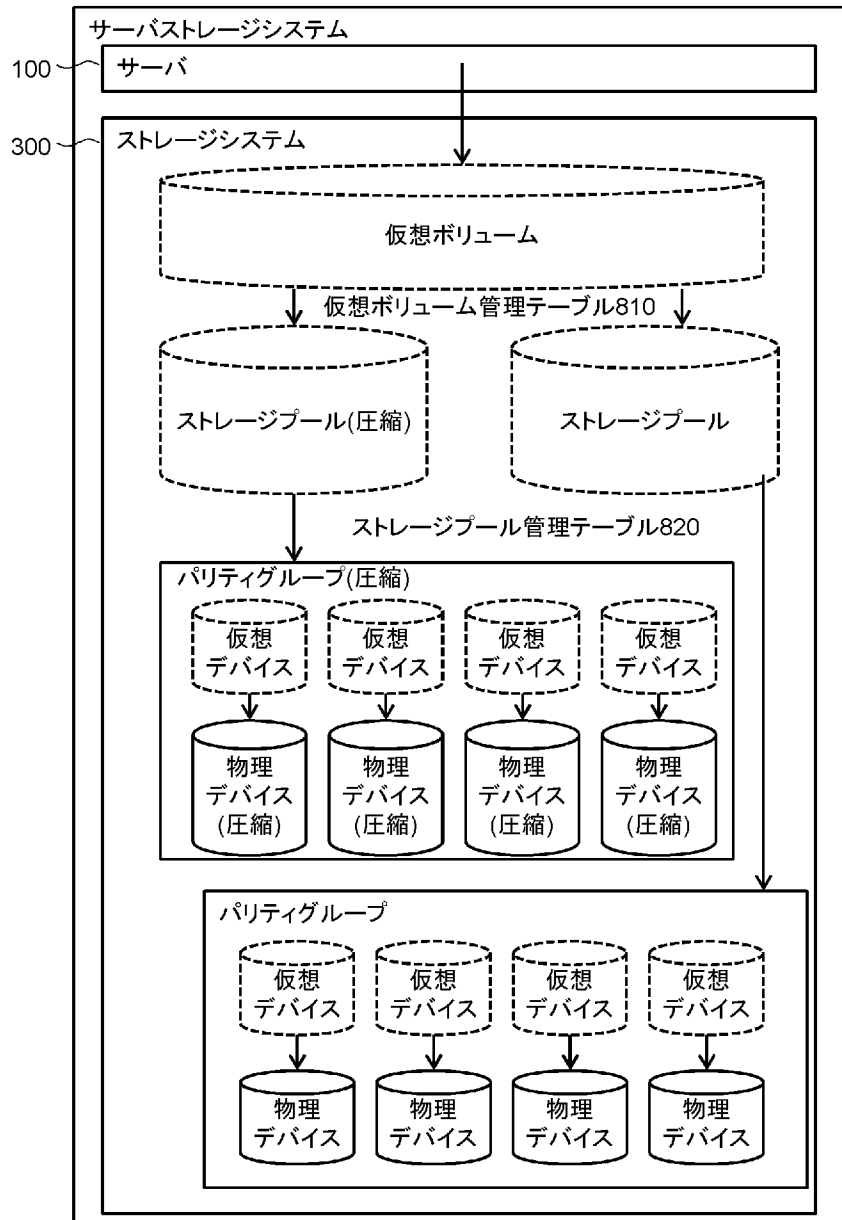
[図12]

Fig. 12



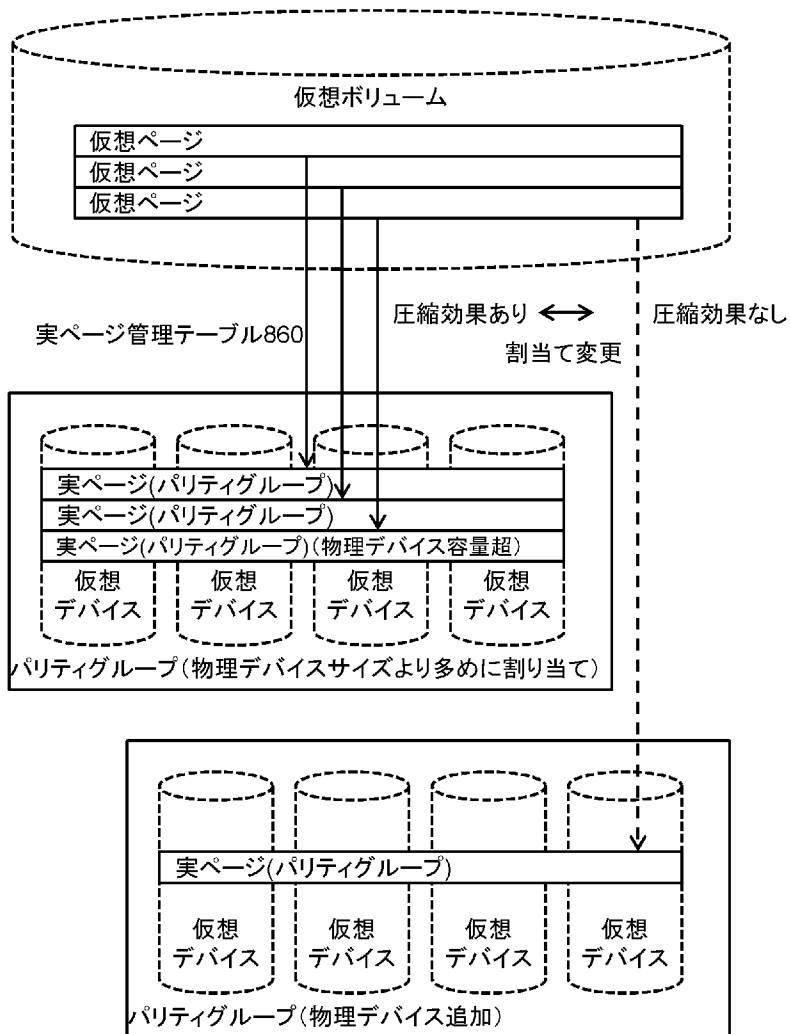
[図13]

Fig. 13



[図14]

Fig. 14



[図15]

Fig. 15

810

仮想ボリューム管理テーブル

ボリューム#	ボリューム容量	割当済み容量	プール#	ボリューム属性
0	500GB	200GB	0	Compressed
1	300GB	200GB	1	N/A
2	200GB	30GB	0	Compressed
⋮	⋮	⋮	⋮	⋮

[図16]

Fig. 16

820



ストレージプール管理テーブル

プール#	RAIDグループリスト	プール属性	空き仮想ページ数
0	1, 5, 8, 10	SSD	20000
1	2, 3, 4, 6, 7	HDD	10000
2	N/A	N/A	N/A
⋮	⋮	⋮	⋮

[図17]

Fig. 17

830



RAIDグループ管理テーブル

RAIDグループ#	RAIDレベル	物理デバイス#
0	10	0, 1, 2, 3
1	10	4, 5, 6, 7
2	5	8, 9, 10, 11
⋮	⋮	⋮

[図18]

Fig. 18

840



物理デバイス管理テーブル

仮想デバイス#	物理デバイス#	物理容量	論理使用容量	物理使用容量
0	1	500GB	50GB	10GB
1	2	500GB	50GB	20GB
2	3	500GB	100GB	30GB
⋮	⋮	⋮	⋮	⋮

[図19]

850
↓

Fig. 19

仮想ページ管理テーブル	
仮想ボリューム#	1
実ページポインタ	0x1000
ページ属性	Compressed

仮想ページ管理テーブル	
仮想ボリューム#	1
実ページポインタ	NULL
ページ属性	Not compressed

[図20]

860
↓

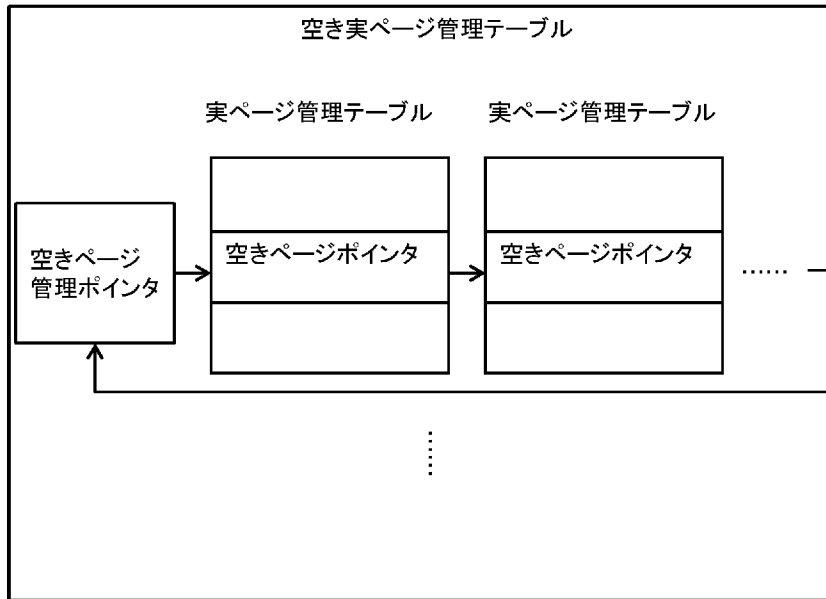
Fig. 20

実ページ管理テーブル	
RAIDグループ#	1
実ページ先頭アドレス	0x0000
空きページポインタ	NULL

[図21]

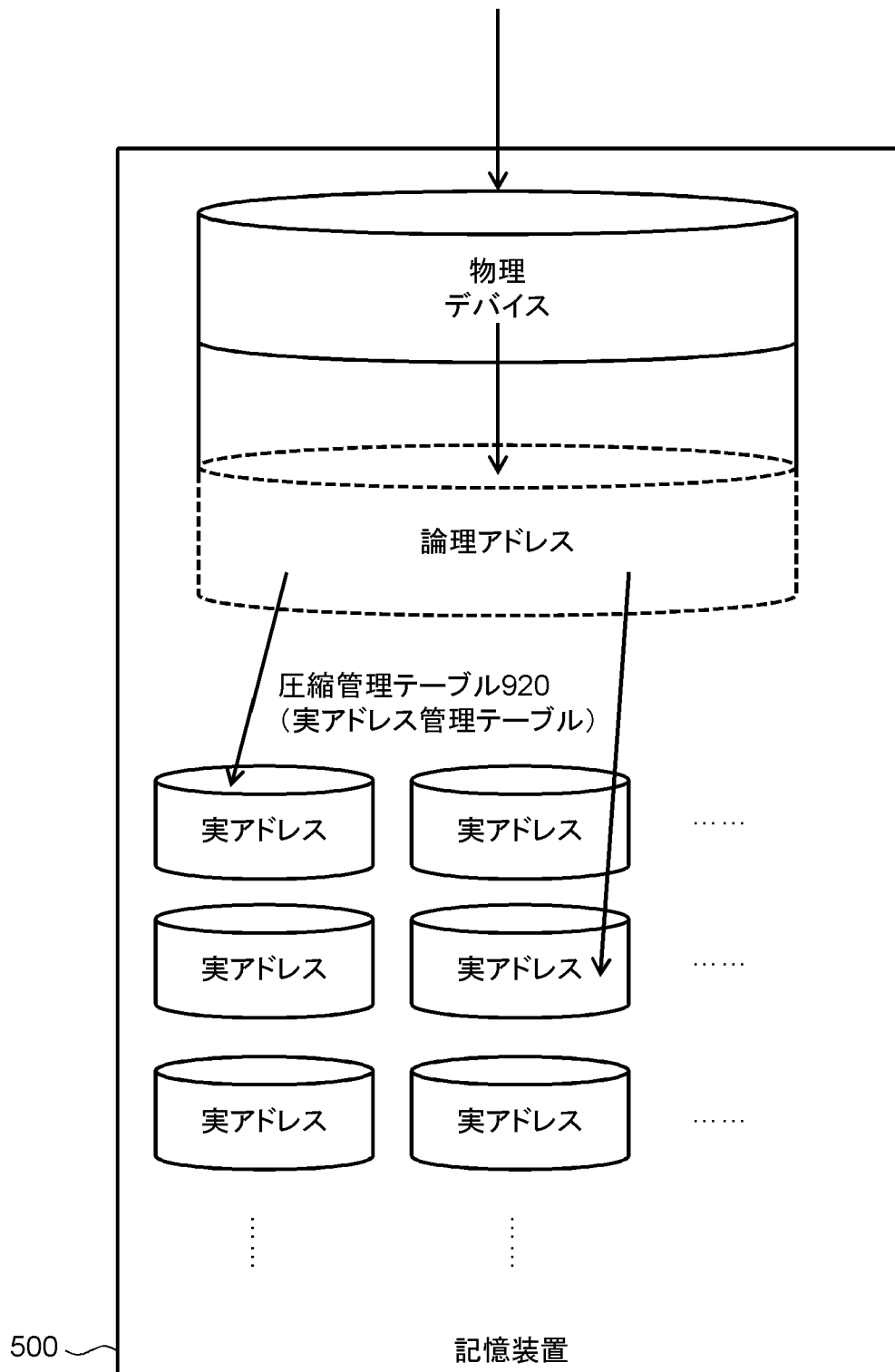
870
↓

Fig. 21



[図22]

Fig. 22



[図23]

Fig. 23

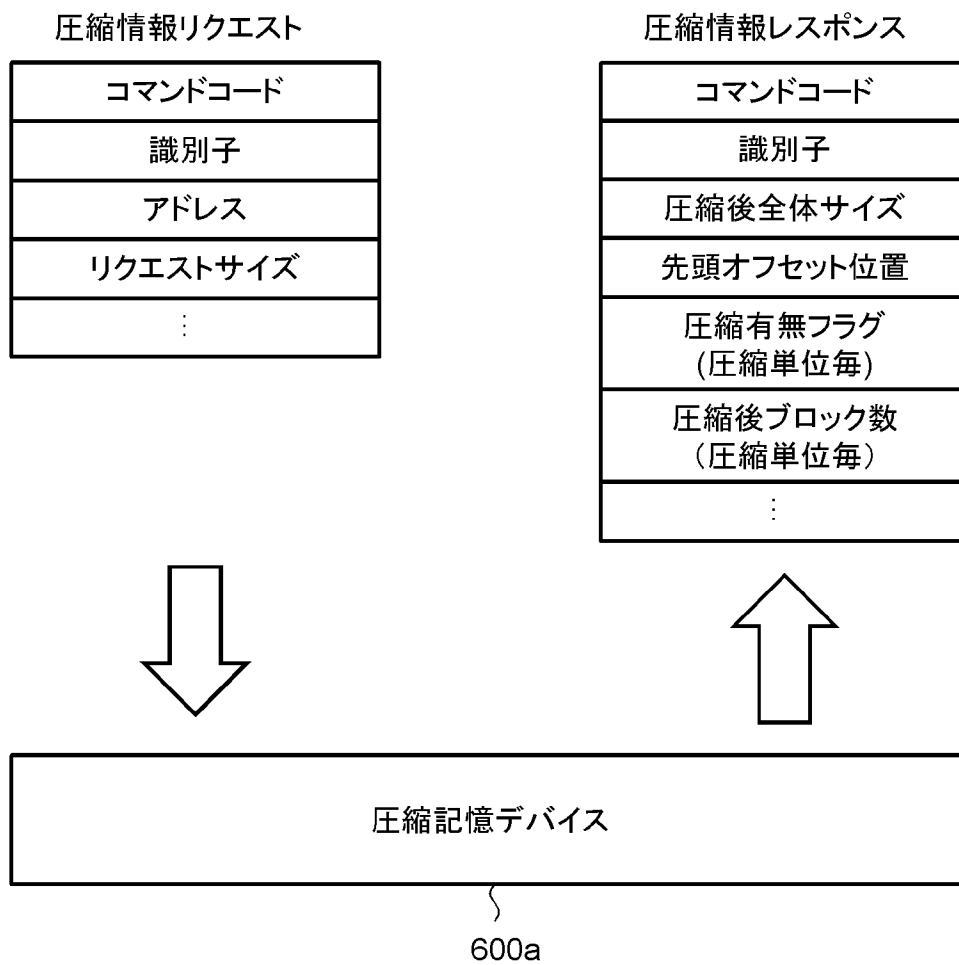
920
↓

圧縮管理テーブル

論理アドレス	実アドレス	圧縮 フラグ	圧縮後 ブロック数 (Max 16)	...
0	00100000 (フラッシュメモリ1-0)	有	14 (8KB→7KB)	...
8	00000000 (フラッシュメモリ0-0)	有	8 (8KB→4KB)	...
16	00101110 (フラッシュメモリ1-14)	無	16 (8KB→8KB)	...
24	00111110 (フラッシュメモリ1-30)	有	16 (8KB→8KB)	...
...	...	...	...	...

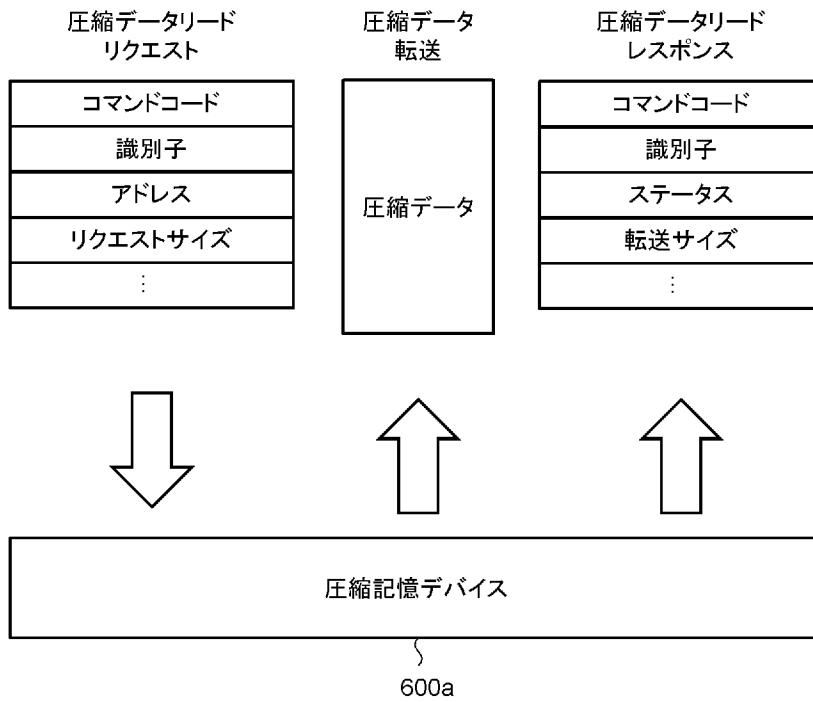
[図24]

Fig. 24



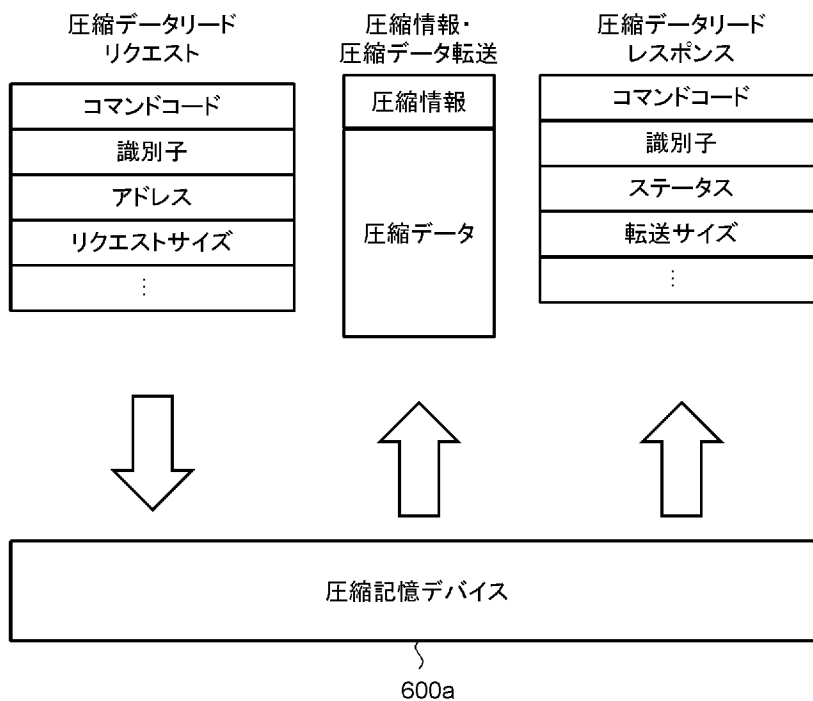
[図25]

Fig. 25



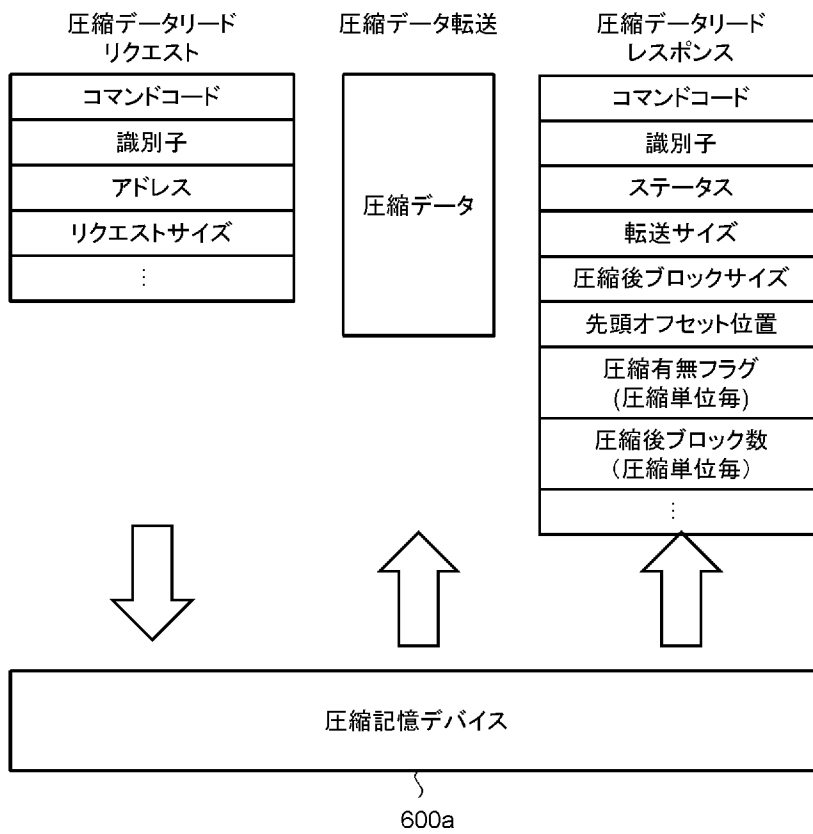
[図26]

Fig. 26



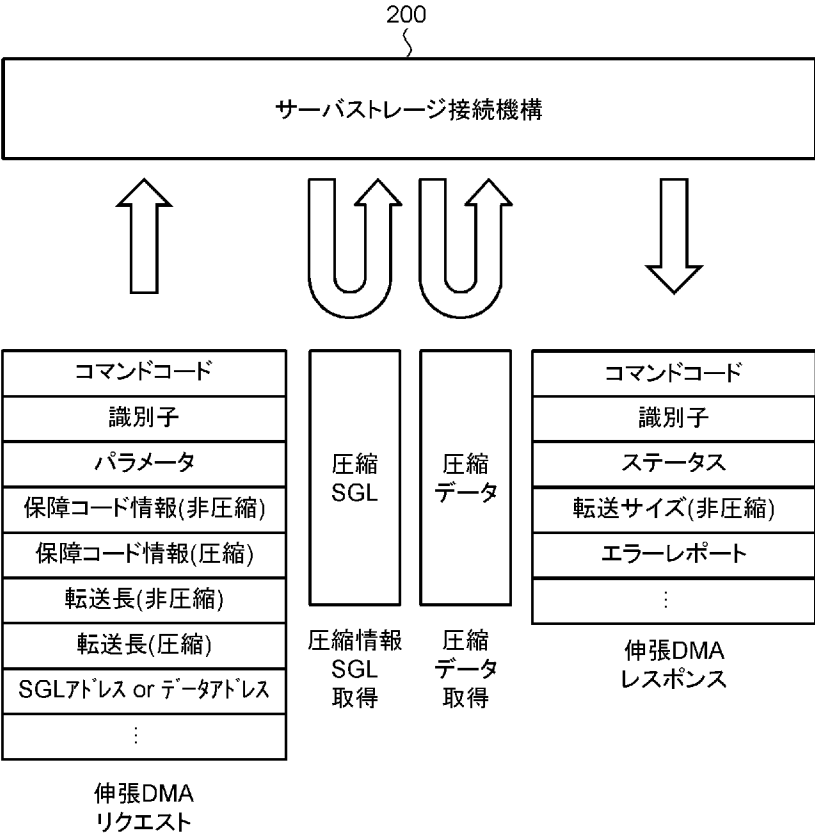
[図27]

Fig. 27



[図28]

Fig. 28



[図29]

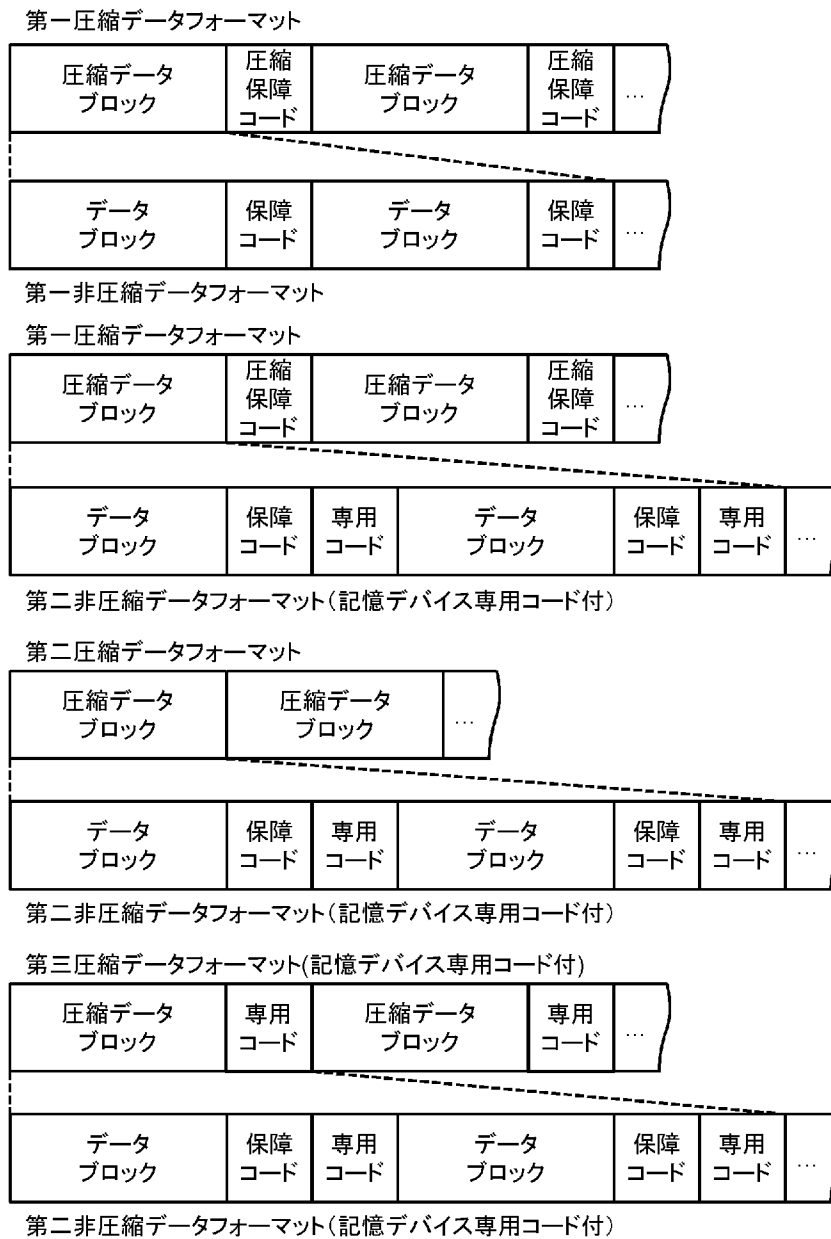
Fig. 29

圧縮情報SGL

識別子
SGL情報
アドレス (0)
転送サイズ (0)
圧縮有無フラグ(0) (圧縮単位毎)
圧縮後ブロック数(0) (圧縮単位毎)
アドレス (1)
転送サイズ (1)
圧縮有無フラグ (1) (圧縮単位毎)
圧縮後ブロック数 (1) (圧縮単位毎)
⋮
次SGLアドレス
SGLフィールド チェックコード

[図30]

Fig. 30



[図31]

Fig. 31

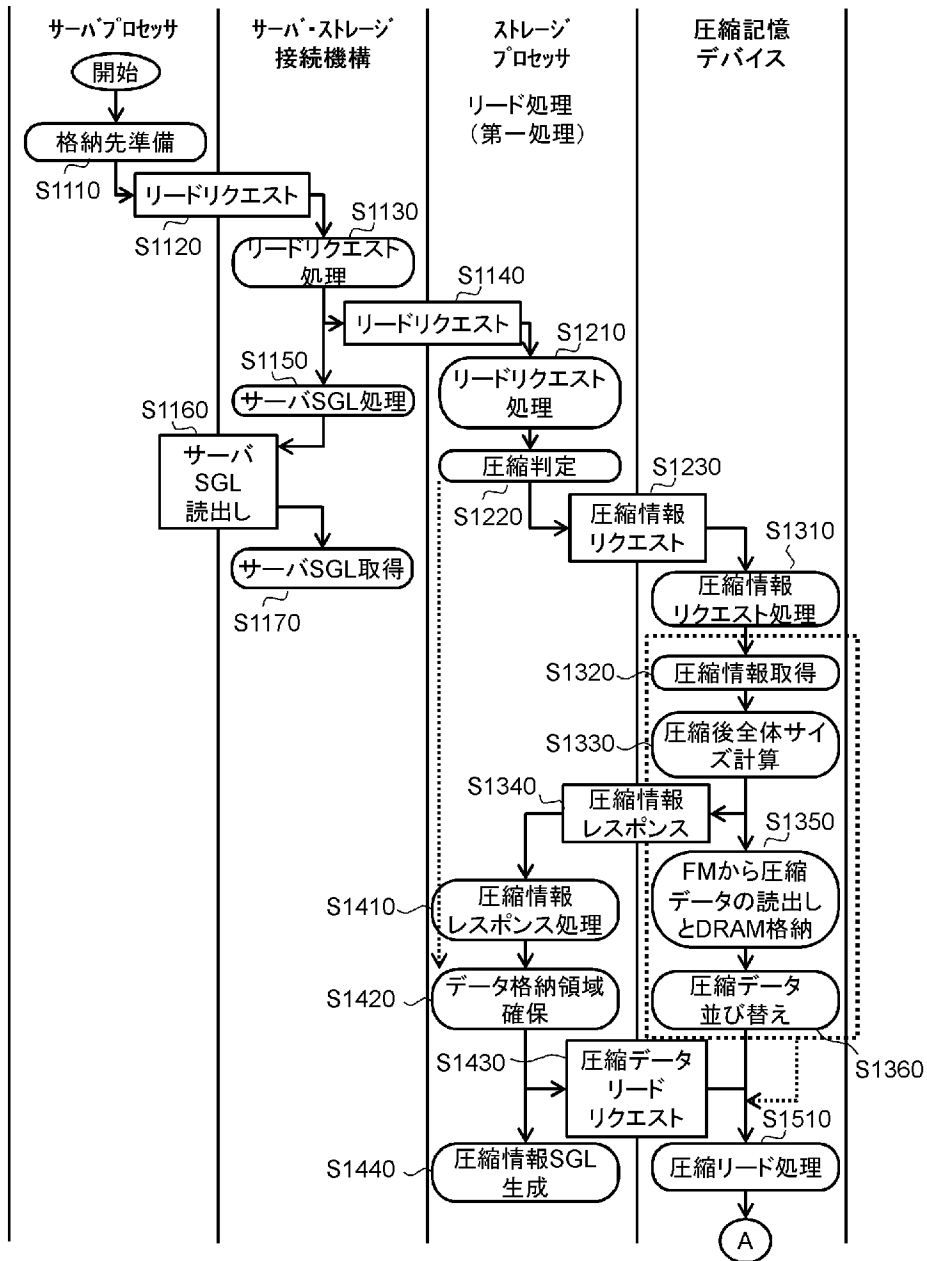
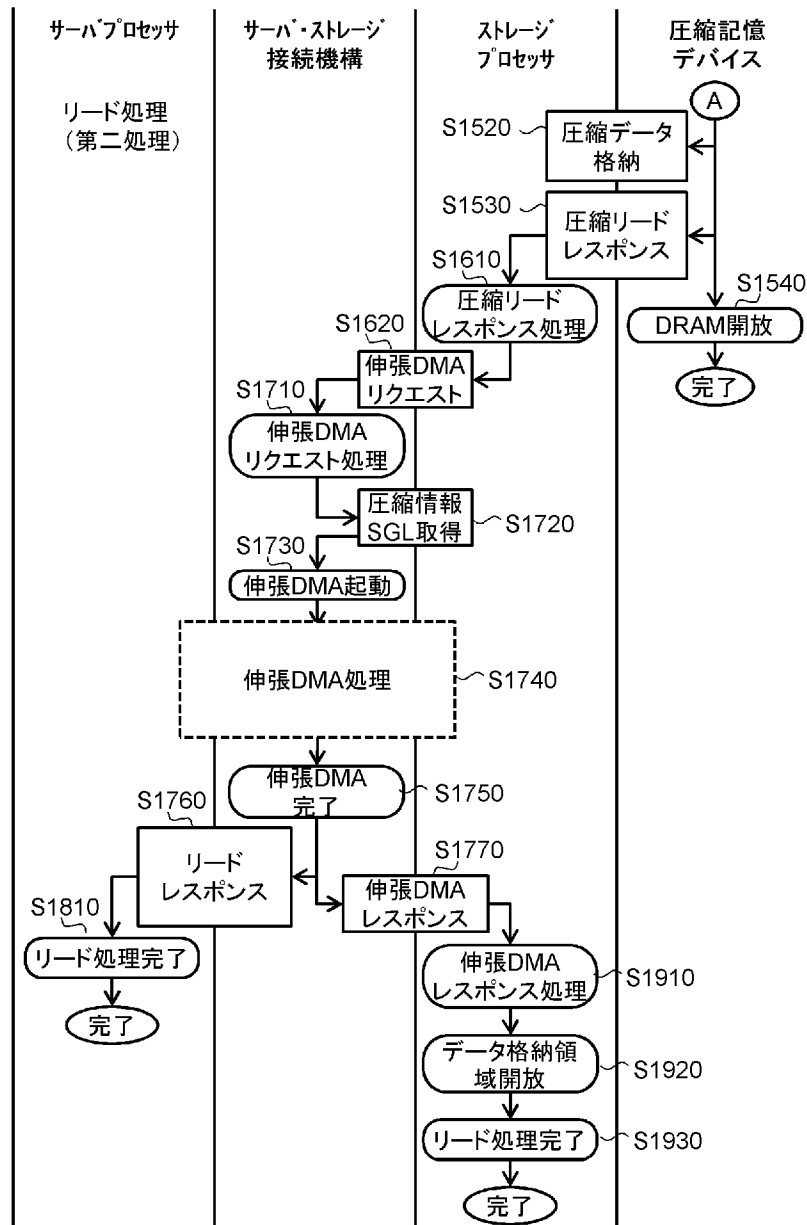
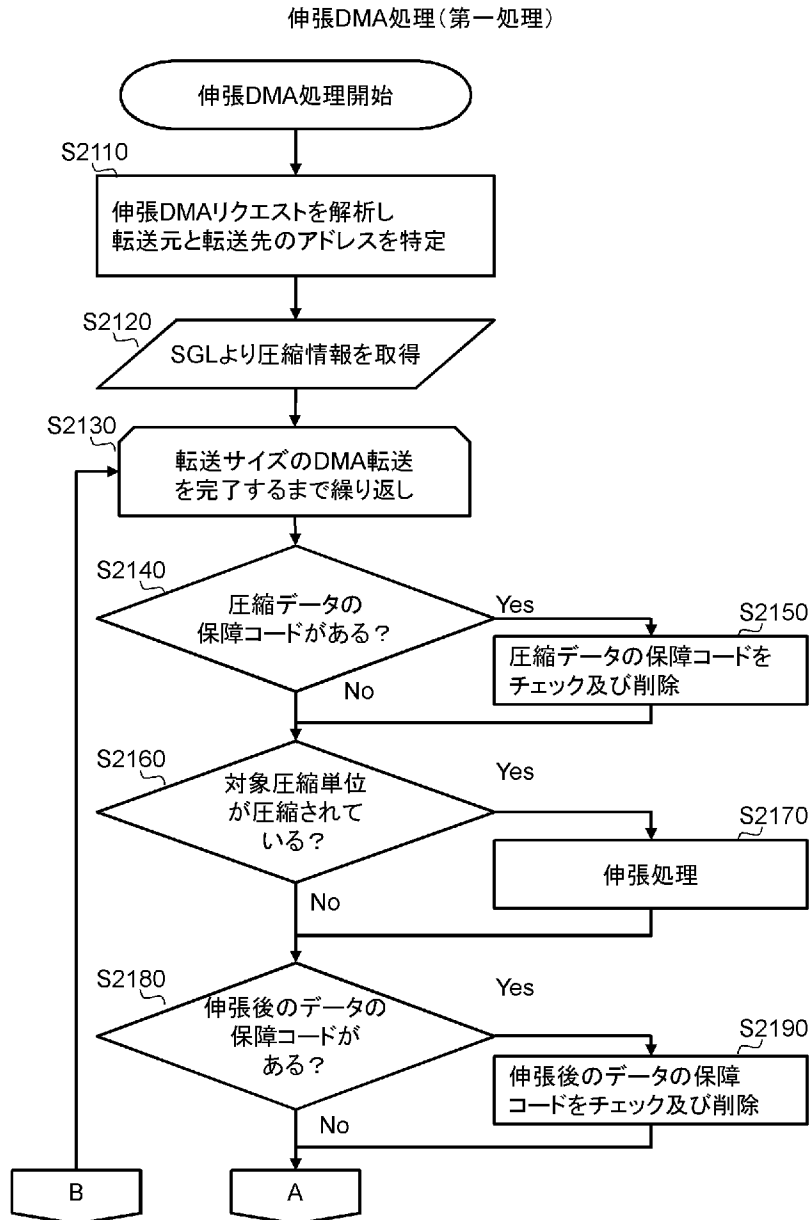


Fig. 32



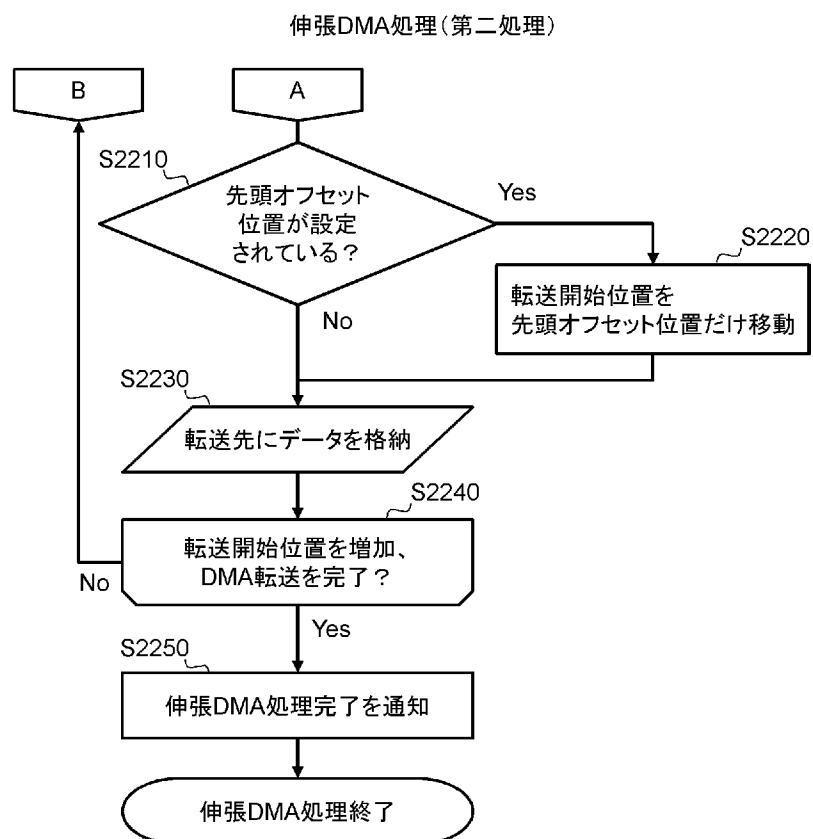
[図33]

Fig. 33



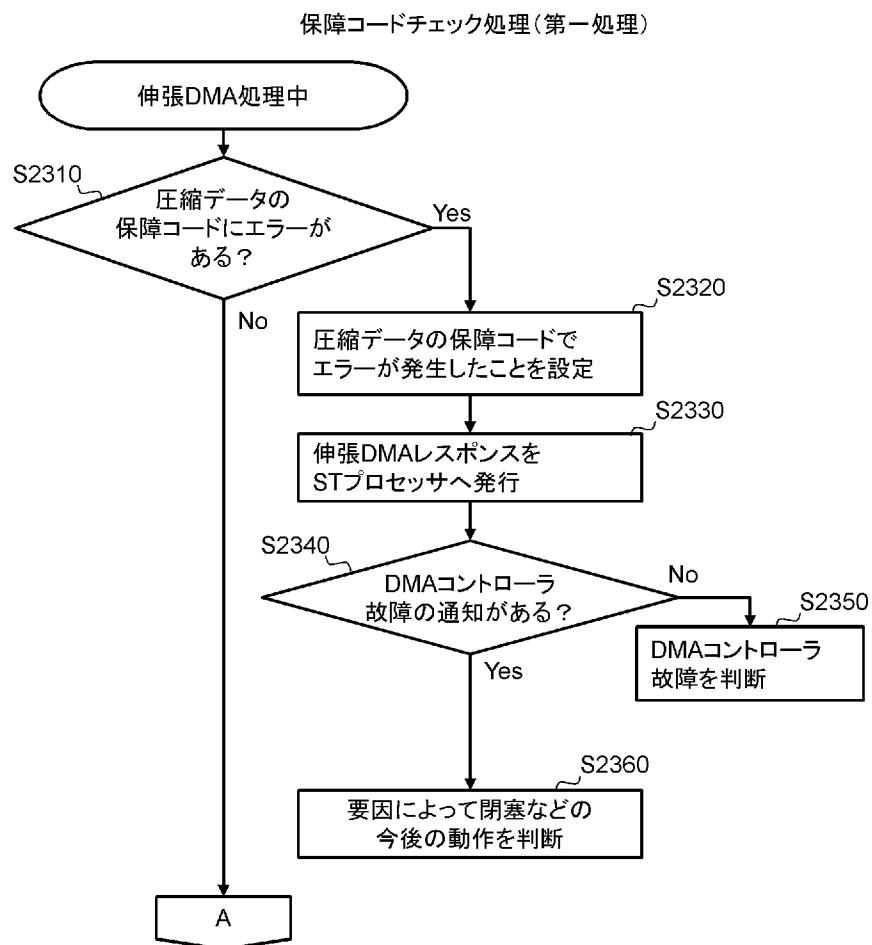
[図34]

Fig. 34



[図35]

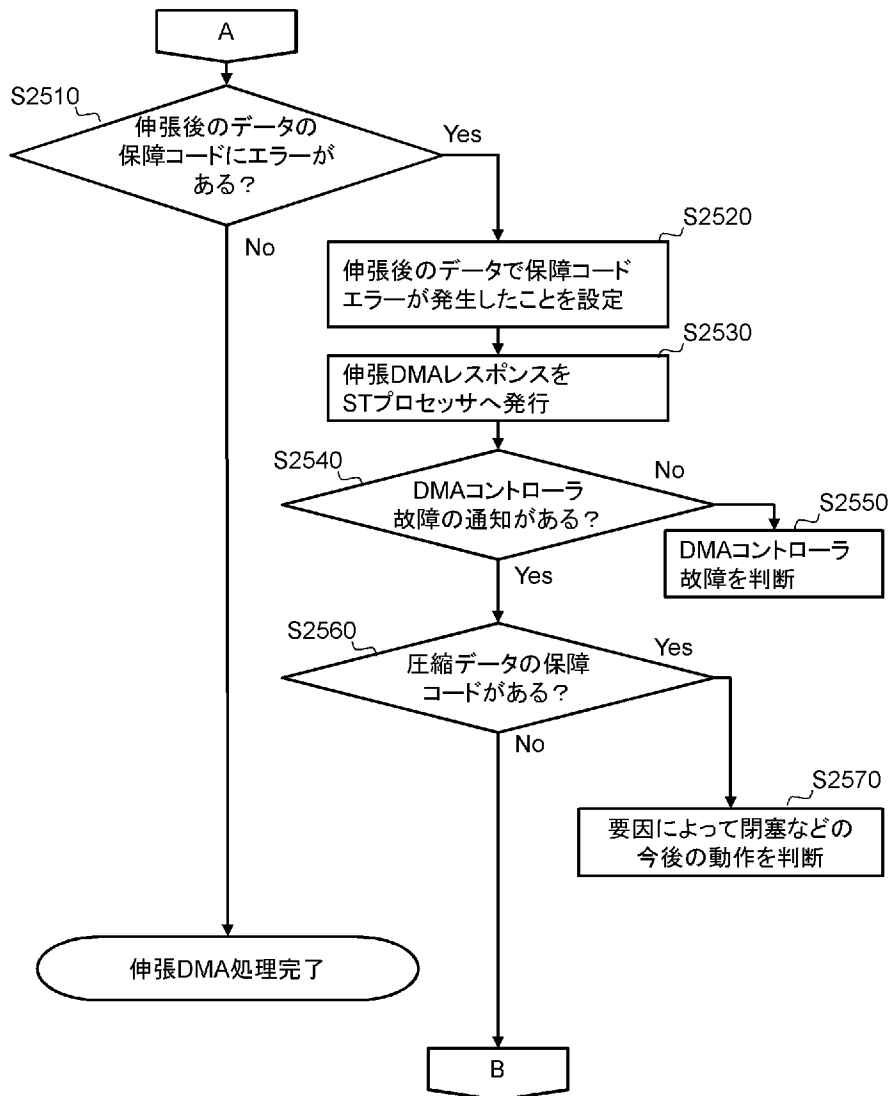
Fig. 35



[図36]

Fig. 36

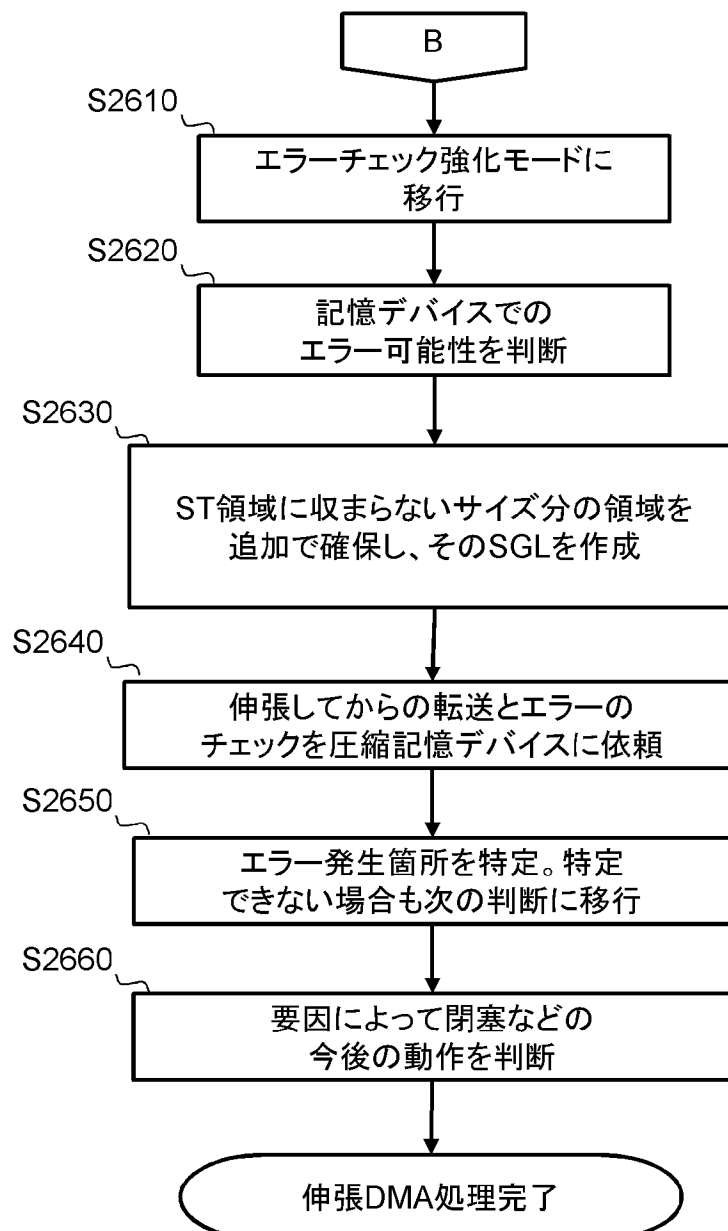
保障コードチェック処理(第二処理)



[図37]

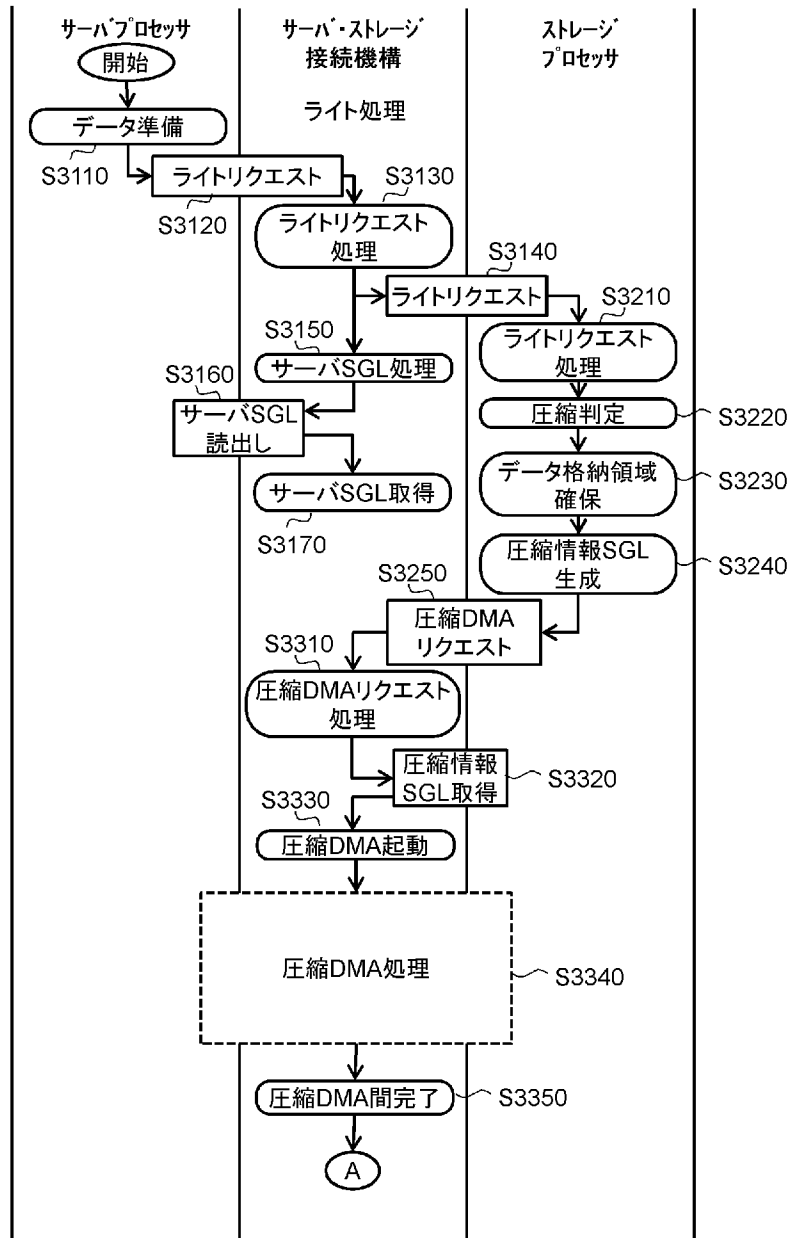
Fig. 37

保障コードチェック処理(第三処理)



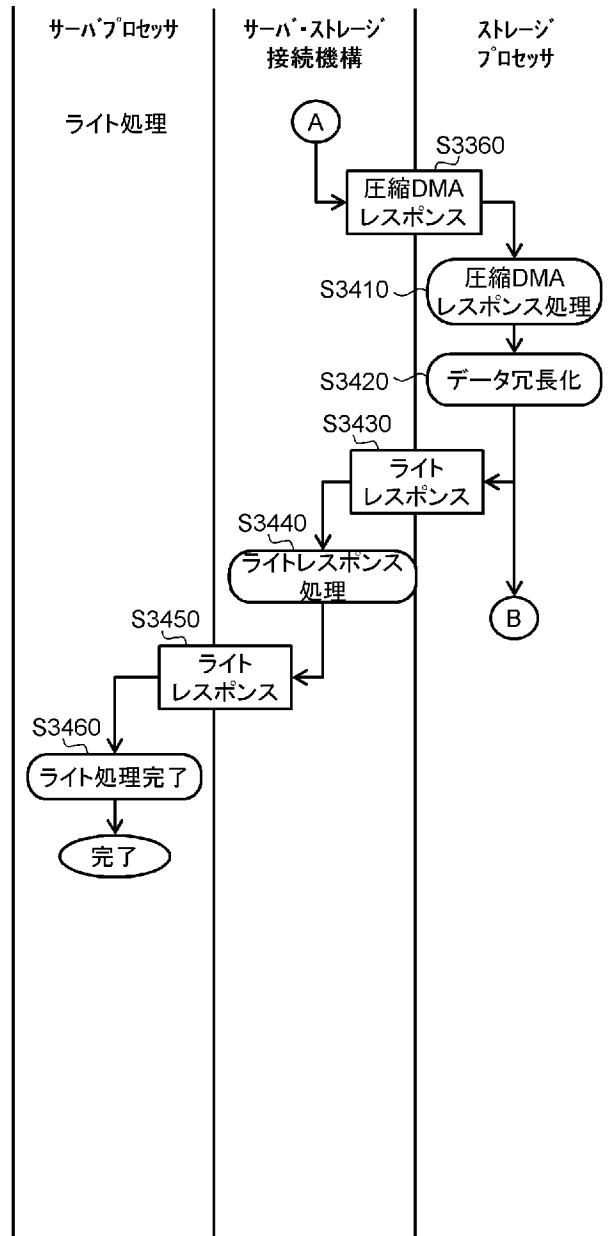
[図38]

Fig. 38



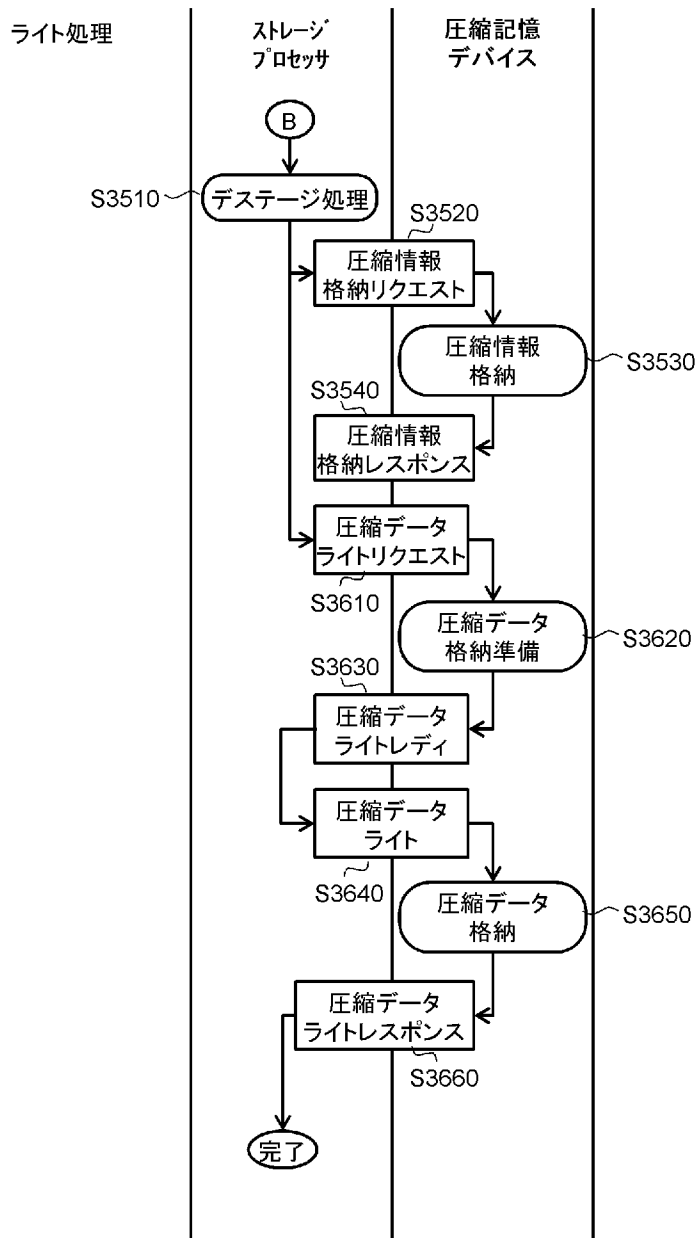
[図39]

Fig. 39



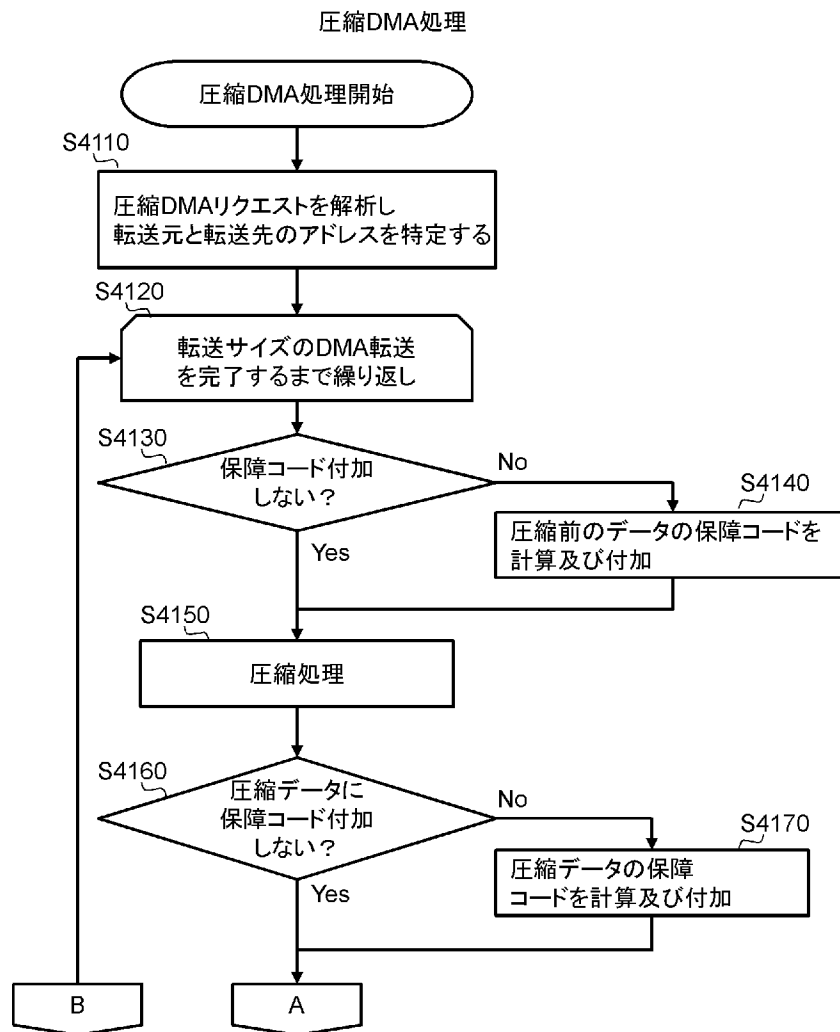
[図40]

Fig. 40



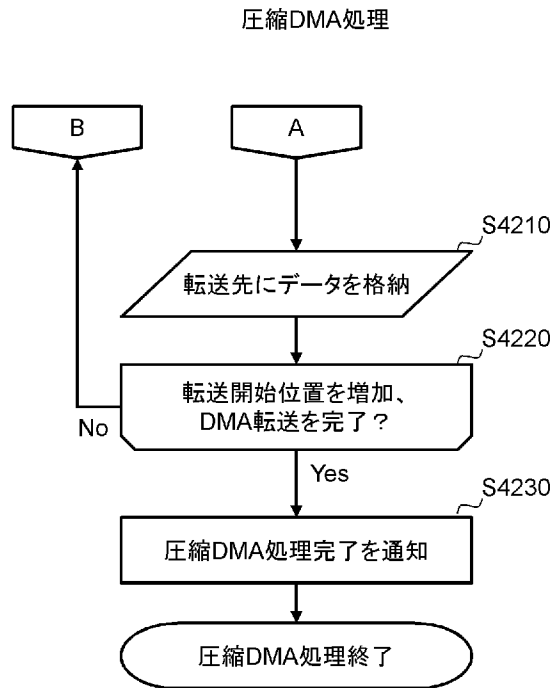
[図41]

Fig. 41



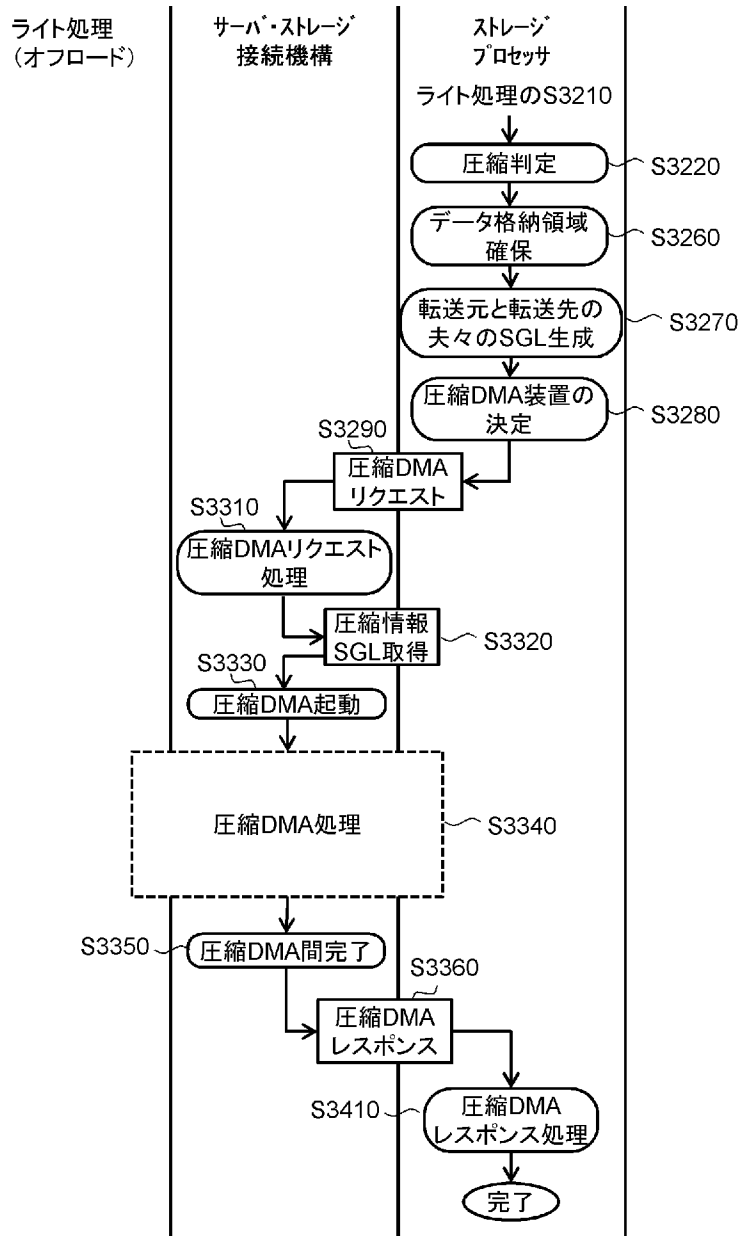
[図42]

Fig. 42



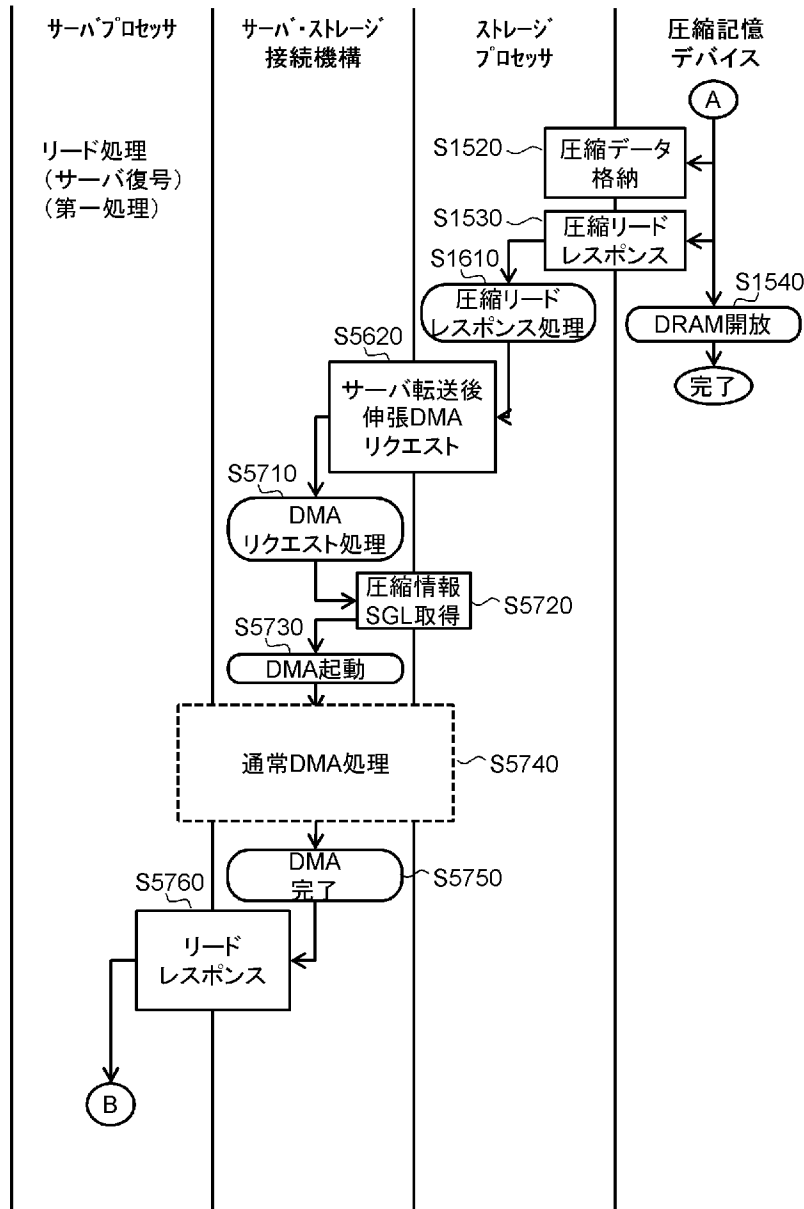
[図43]

Fig. 43



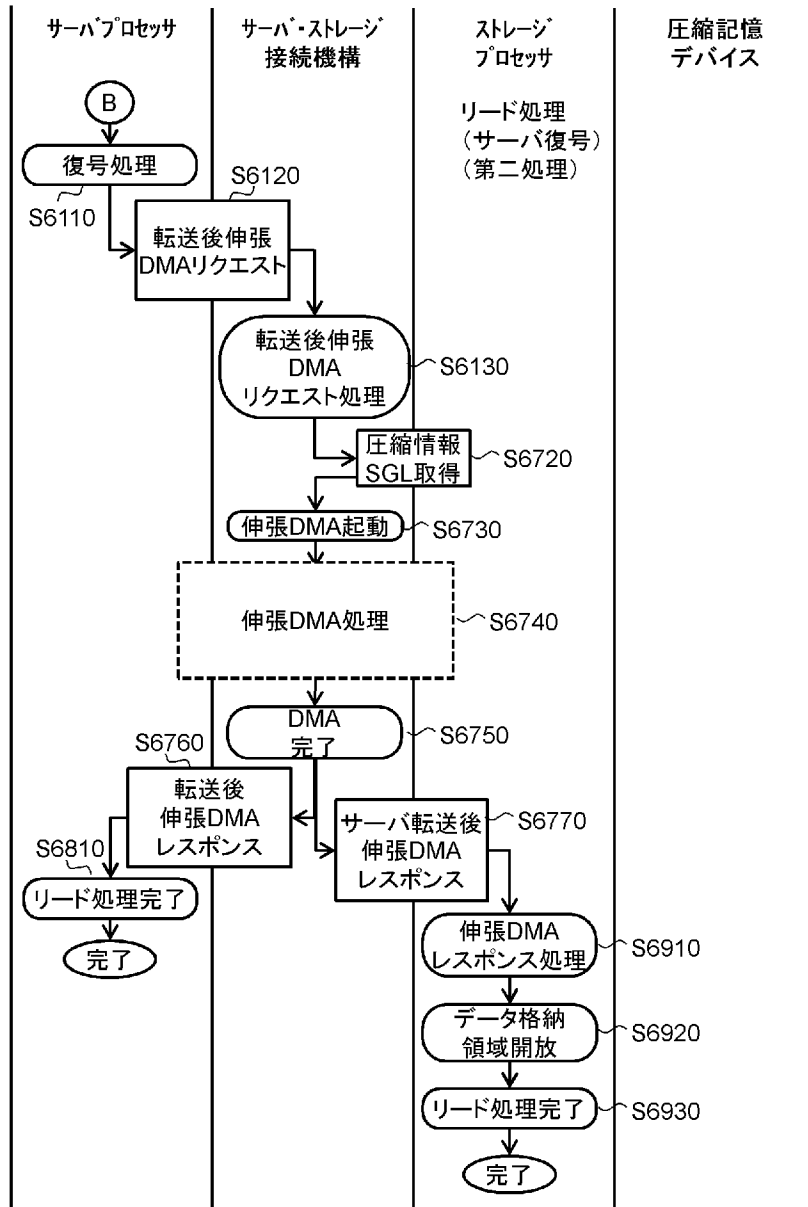
[図44]

Fig. 44



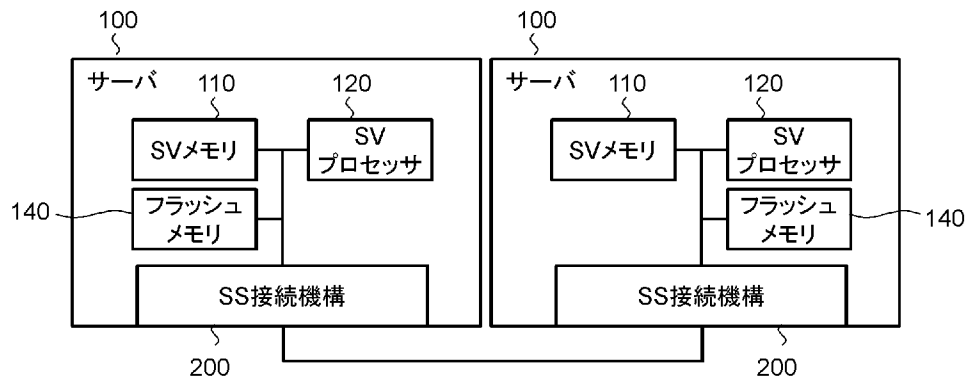
[図45]

Fig. 45



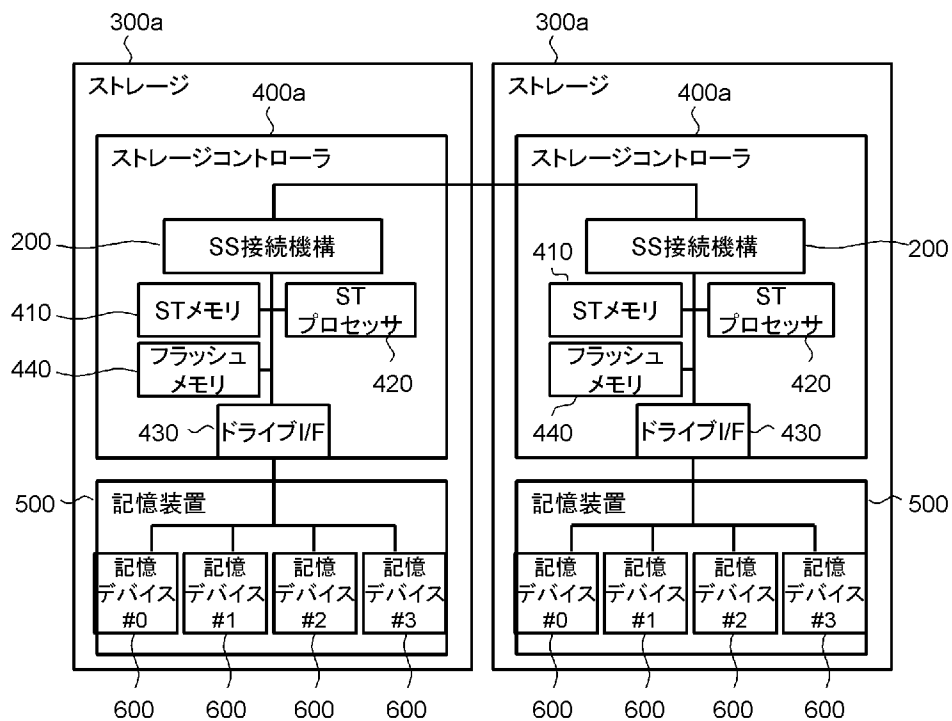
[図46]

Fig. 46



[図47]

Fig. 47



INTERNATIONAL SEARCH REPORT

International application No.

PCT/JP2014/064076

A. CLASSIFICATION OF SUBJECT MATTER

G06F3/06(2006.01)i, G06F12/00(2006.01)i, G06F13/28(2006.01)i

According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

G06F3/06, G06F12/00, G06F13/14, G06F13/38

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Jitsuyo Shinan Koho	1922-1996	Jitsuyo Shinan Toroku Koho	1996-2014
Kokai Jitsuyo Shinan Koho	1971-2014	Toroku Jitsuyo Shinan Koho	1994-2014

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
A	JP 2008-171312 A (Hitachi, Ltd.), 24 July 2008 (24.07.2008), paragraphs [0006] to [0154]; fig. 1 to 22 & US 2008/0172525 A1	1-15
A	JP 2006-251960 A (Hitachi, Ltd.), 21 September 2006 (21.09.2006), paragraphs [0004] to [0145]; fig. 1 to 24 & US 2006/0206640 A1	1-15

☐ Further documents are listed in the continuation of Box C.

☐ See patent family annex.

* Special categories of cited documents:

"A" document defining the general state of the art which is not considered to be of particular relevance

"E" earlier application or patent but published on or after the international filing date

"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)

"O" document referring to an oral disclosure, use, exhibition or other means

"P" document published prior to the international filing date but later than the priority date claimed

"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention

"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone

"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art

"&" document member of the same patent family

Date of the actual completion of the international search
06 August, 2014 (06.08.14)

Date of mailing of the international search report
19 August, 2014 (19.08.14)

Name and mailing address of the ISA/
Japanese Patent Office

Authorized officer

Facsimile No.

Telephone No.

A. 発明の属する分野の分類（国際特許分類（I P C））

Int.Cl. G06F3/06(2006.01)i, G06F12/00(2006.01)i, G06F13/28(2006.01)i

B. 調査を行った分野

調査を行った最小限資料（国際特許分類（I P C））

Int.Cl. G06F3/06, G06F12/00, G06F13/14, G06F13/38

最小限資料以外の資料で調査を行った分野に含まれるもの

日本国実用新案公報	1 9 2 2 - 1 9 9 6 年
日本国公開実用新案公報	1 9 7 1 - 2 0 1 4 年
日本国実用新案登録公報	1 9 9 6 - 2 0 1 4 年
日本国登録実用新案公報	1 9 9 4 - 2 0 1 4 年

国際調査で使用した電子データベース（データベースの名称、調査に使用した用語）

C. 関連すると認められる文献

引用文献の カテゴリー*	引用文献名 及び一部の箇所が関連するときは、その関連する箇所の表示	関連する 請求項の番号
A	J P 2 0 0 8 - 1 7 1 3 1 2 A (株式会社日立製作所) 2 0 0 8 . 0 7 . 2 4 , 段落【0 0 0 6】 - 【0 1 5 4】, 図 1 - 2 2 & U S 2 0 0 8 / 0 1 7 2 5 2 5 A 1	1 - 1 5
A	J P 2 0 0 6 - 2 5 1 9 6 0 A (株式会社日立製作所) 2 0 0 6 . 0 9 . 2 1 , 段落【0 0 0 4】 - 【0 1 4 5】, 図 1 - 2 4 & U S 2 0 0 6 / 0 2 0 6 6 4 0 A 1	1 - 1 5

☐ C 欄の続きにも文献が列挙されている。☐ パテントファミリーに関する別紙を参照。

* 引用文献のカテゴリー

「A」特に関連のある文献ではなく、一般的技術水準を示すもの

「E」国際出願日前の出願または特許であるが、国際出願日以後に公表されたもの

「L」優先権主張に疑義を提起する文献又は他の文献の発行日若しくは他の特別な理由を確立するために引用する文献（理由を付す）

「O」口頭による開示、使用、展示等に言及する文献

「P」国際出願日前で、かつ優先権の主張の基礎となる出願

の日の後に公表された文献

「T」国際出願日又は優先日後に公表された文献であって出願と矛盾するものではなく、発明の原理又は理論の理解のために引用するもの

「X」特に関連のある文献であって、当該文献のみで発明の新規性又は進歩性がないと考えられるもの

「Y」特に関連のある文献であって、当該文献と他の1以上の文献との、当業者にとって自明である組合せによって進歩性がないと考えられるもの

「&」同一パテントファミリー文献

国際調査を完了した日

0 6 . 0 8 . 2 0 1 4

国際調査報告の発送日

1 9 . 0 8 . 2 0 1 4

国際調査機関の名称及びあて先

日本国特許庁（I S A / J P）

郵便番号 1 0 0 - 8 9 1 5

東京都千代田区霞が関三丁目4番3号

特許庁審査官（権限のある職員）

古河 雅輝

電話番号 0 3 - 3 5 8 1 - 1 1 0 1 内線 3 5 6 8

5 T

3 2 4 2