(19)日本国特許庁(JP)　(12)特　許　公　報(B2)　(11)特許番号
特許第4394624号
(P4394624)
(45)発行日　平成22年1月6日(2010.1.6)　(24)登録日　平成21年10月23日(2009.10.23)

(51)Int.Cl.　F I
G06F 9/46 (2006.01)　G06F 9/46 350

請求項の数 10 (全 26 頁)

(21)出願番号　特願2005-273400(P2005-273400)
(22)出願日　平成17年9月21日(2005.9.21)
(65)公開番号　特開2007-86963(P2007-86963A)
(43)公開日　平成19年4月5日(2007.4.5)
審査請求日　平成20年1月28日(2008.1.28)

(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　100075513
弁理士　後藤　政喜
(74)代理人　100084537
弁理士　松田　嘉夫
(74)代理人　100114236
弁理士　藤井　正弘
(72)発明者　垂井　俊明
東京都国分寺市東恋ヶ窪一丁目２８０番地
株式会社日立製作所　中央研究所内
(72)発明者　保田　淑子
東京都国分寺市東恋ヶ窪一丁目２８０番地
株式会社日立製作所　中央研究所内
最終頁に続く

(54)【発明の名称】計算機システム及びＩ／Ｏブリッジ

(57)【特許請求の範囲】
【請求項１】
複数のＣＰＵコアと、前記ＣＰＵコアに接続されたＩ／Ｏブリッジと、前記ＣＰＵコアまたはＩ／Ｏブリッジからアクセス可能な主記憶と、を備えたＣＰＵモジュールと、
前記ＣＰＵモジュールのＩ／Ｏブリッジと Ｉ／Ｏモジュールとを接続するＩ／Ｏスイッチと、を備えた計算機システムにおいて、
前記ＣＰＵモジュールは、
前記複数のＣＰＵコアと主記憶とを複数の論理区画に分割するファームウェアを備え、
前記Ｉ／Ｏブリッジは、
前記論理区画とＩ／Ｏモジュールとの間で送受信されるＩ／Ｏアクセス情報を中継する際に、前記論理区画毎に設定された仮想経路情報と、当該Ｉ／ＯブリッジからＩ／Ｏモジュールまでの経路情報とを前記Ｉ／Ｏアクセス情報の経路情報に付加して、論理区画毎に前記Ｉ／Ｏモジュールとの間のＩ／Ｏアクセス情報を切り換える仮想スイッチを備えたことを特徴とする計算機システム。
【請求項２】
請求項１に記載の計算機システムにおいて、
前記仮想スイッチは、
前記論理区画からＩ／Ｏモジュールへ送信するＩ／Ｏアクセス情報の経路情報に、前記論理区画毎の仮想経路情報を前記仮想スイッチの経路情報として付加し、前記Ｉ／Ｏスイッチへ送信する経路情報設定部と、

前記Ｉ／Ｏモジュールから受信したＩ／Ｏアクセス情報に含まれる経路情報から前記仮想スイッチの仮想経路情報を抽出する宛先抽出部と、
前記抽出した仮想経路情報に対応する論理区画を特定して前記Ｉ／Ｏアクセス情報を当該論理区画へ転送する宛先論理区画特定部と、
を備えたことを特徴とする計算機システム。
【請求項３】
請求項１に記載の計算機システムにおいて、
前記仮想スイッチは、
前記論理区画が割り当てられた主記憶上の領域のベースアドレスとサイズを、論理区画に対応付けて格納するアドレス管理部と、
前記論理区画毎にマルチキャストによる書き込みを制御するための宛先論理区画設定部と、
前記Ｉ／Ｏモジュールから受信したＩ／Ｏアクセス情報に含まれるコマンドを抽出するコマンド抽出部と、
前記抽出したコマンドがマルチキャストＤＭＡのときには、前記論理区画に対応するベースアドレスとサイズを前記アドレス管理部から取得し、前記マルチキャストＤＭＡコマンドに含まれるＤＭＡアドレスに前記ベースアドレスを論理区画毎に加算するアドレス変換部と、
前記加算結果が前記サイズで示される前記論理区画に割り当てられた主記憶上の領域を超えていなければ、前記加算結果が示す論理区画に対応する主記憶のアドレスにそれぞれＤＭＡアクセスを行うＤＭＡ処理部と、
マルチキャストの宛先である全ての論理区画に対して上記書き込みを行う、複数回書き込み処理部と、
を備えたことを特徴とする計算機システム。
【請求項４】
請求項１に記載の計算機システムにおいて、
前記仮想スイッチは、
前記論理区画毎にマルチキャストによるイベントを制御するための宛先論理区画設定部と、
前記Ｉ／Ｏモジュールから受信したＩ／Ｏアクセス情報に含まれるコマンドを抽出するコマンド抽出部と、
前記抽出したコマンドがマルチキャストイベントのときには、宛先論理区画番号を前記宛先論理区画設定部に設定し、前記ファームウェアにマルチキャストイベントに含まれるイベントを通知するマルチキャストイベント通知部と、を有し、
前記ファームウェアは、前記通知を受けたイベントを前記宛先論理区画設定部の設定に基づく複数の論理区画に通知することを特徴とする計算機システム。
【請求項５】
請求項１に記載の計算機システムにおいて、
前記仮想スイッチは、
前記論理区画毎に設定する仮想ポートを複数備え、
前記論理区画と仮想ポートとの対応関係と、論理区画の状態を管理する論理区画管理部を有し、
前記ファームウェアは、前記論理区画の生成時または削除時に前記論理区画管理部の前記状態を更新することを特徴とする計算機システム。
【請求項６】
請求項１に記載の計算機システムにおいて、
前記Ｉ／Ｏスイッチは、複数のＣＰＵブレードと複数のＩ／Ｏモジュールとを接続し、
前記ＣＰＵモジュールに設定された前記複数の論理区画に対する前記Ｉ／Ｏモジュールの割り当て状態を管理するファブリック管理装置を備え、
前記ファブリック管理装置は、ひとつのＩ／Ｏモジュールを複数の論理区画またはＣＰ

Ｕモジュールに割り当てることを特徴とする計算機システム。
【請求項７】
複数のＣＰＵコアと主記憶とをファームウェアによって分割した複数の論理区画をＩ／Ｏスイッチに接続し、前記Ｉ／Ｏスイッチに接続されたＩ／Ｏモジュールと論理区画との間でＩ／Ｏアクセスを行うＩ／Ｏブリッジであって、
前記Ｉ／Ｏブリッジは、
前記論理区画とＩ／Ｏモジュールとの間で送受信されるＩ／Ｏアクセス情報を中継する際に、前記論理区画毎に設定された仮想経路情報を前記Ｉ／Ｏアクセス情報の経路情報に付加して、論理区画毎に前記Ｉ／Ｏモジュールとの間のＩ／Ｏアクセス情報を切り換える仮想スイッチを備えたことを特徴とするＩ／Ｏブリッジ。
【請求項８】
請求項７に記載のＩ／Ｏブリッジにおいて、
前記仮想スイッチは、
前記論理区画からＩ／Ｏアクセス情報を受信すると、このＩ／Ｏアクセス情報を発行した論理区画を特定する発信元特定部と、
論理区画に対応する前記仮想スイッチの仮想経路情報を前記Ｉ／Ｏアクセス情報に付加し、前記Ｉ／Ｏスイッチへ送信する経路情報設定部と、
を有することを特徴とするＩ／Ｏブリッジ。
【請求項９】
請求項７に記載のＩ／Ｏブリッジにおいて、
前記仮想スイッチは、
前記Ｉ／Ｏモジュールから受信したＩ／Ｏアクセス情報に含まれる経路情報から前記仮想スイッチの仮想経路情報を抽出する宛先抽出部と、
前記抽出した仮想経路情報に対応する論理区画を特定して前記Ｉ／Ｏアクセス情報を当該論理区画へ転送する宛先論理区画特定部と、
を備えたことを特徴とするＩ／Ｏブリッジ。
【請求項１０】
請求項７に記載のＩ／Ｏブリッジにおいて、
前記仮想スイッチは、
前記論理区画が割り当てられた主記憶上の領域のベースアドレスとサイズを、論理区画に対応付けて格納するアドレス管理部と、
前記論理区画毎にマルチキャストによる書き込みを制御するための宛先論理区画設定部と、
前記Ｉ／Ｏモジュールから受信したＩ／Ｏアクセス情報に含まれるコマンドを抽出するコマンド抽出部と、
前記抽出したコマンドがマルチキャストＤＭＡのときには、前記論理区画に対応するベースアドレスとサイズを前記アドレス管理部から取得し、前記マルチキャストＤＭＡコマンドに含まれるＤＭＡアドレスに前記ベースアドレスを論理区画毎に加算するアドレス変換部と、
前記加算結果が前記サイズで示される前記論理区画に割り当てられた主記憶上の領域を超えていなければ、前記加算結果が示す論理区画に対応する主記憶のアドレスにそれぞれＤＭＡアクセスを行うＤＭＡ処理部と、
マルチキャストの宛先である全ての論理区画に対して上記書き込みを行う、複数回書き込み処理部と、
を備えたことを特徴とするＩ／Ｏブリッジ。
【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、仮想計算機システムに関し、複数の論理区画とＩ／Ｏデバイスの割当をＩ／Ｏスイッチで実現する技術に関する。

【背景技術】
【０００２】
近年、サーバ台数の増加と共に運用に関する複雑さが増加し、運用コストの増大が問題化している。この運用コストを低減する技術として複数サーバを１台にまとめるサーバコンソリデーション（サーバ統合）が注目を集めている。
【０００３】
サーバコンソリデーションを実現する技術のひとつとして、一つの計算機を任意の割合で論理的に分割する仮想計算機が知られており、ハイパバイザなどのファームウェア（またはミドルウェア）により、物理計算機を複数の論理区画（ＬＰＡＲ：Logical PARtition）に分割し、各ＬＰＡＲに対して計算機資源（ＣＰＵ、主記憶、Ｉ／Ｏ）を割当て、各ＬＰＡＲ上でそれぞれＯＳを動作させるものが知られている（例えば、特許文献１）。
【０００４】
あるいは、ＶＭｗａｒｅ（登録商標）のように、ホストＯＳ上で複数のゲストＯＳを稼動させ、各ゲストＯＳを論理区画として提供する技術も知られている。
【０００５】
また、近年、基板単位で計算機を変更可能なブレードサーバにより、ＩＴシステムの構成をより柔軟に変更する技術が知られており、多数のサーバをブレードサーバに統合するサーバコンソリデーション（サーバ統合）が行われている。
【０００６】
さらに近年では、ＡＳ（Advanced Switching）等に代表される、Ｉ／Ｏスイッチが提案されている（例えば、非特許文献１）。この種のＩ／Ｏスイッチは、複数のＣＰＵ（物理的なＣＰＵ）からＩ／Ｏカード（またはＩ／Ｏデバイス）の共有を実現しようとするものである。ＡＳでは、現在普及しつつある、ＰＣＩ－ＥＸＰＲＥＳＳ規格のＩ／Ｏデバイス（またはＩ／Ｏカード）を利用できるため、汎用性が高い。
【特許文献１】特開２００２－３０４３６４号
【非特許文献１】「Advanced Switching 技術概要」、[online]、「平成１７年８月１６日検索」、インターネット＜http://www.asi-sig.org/education/ASI_AdvSwitch_TB_JP.pdf＞
【発明の開示】
【発明が解決しようとする課題】
【０００７】
上記ＶＭｗａｒｅによる論理区画間のＩ／Ｏ共有では、Ｉ／Ｏデバイス側で論理区画を識別することができないため、各論理区画のＩ／Ｏ要求を取りまとめて物理的なＩ／Ｏデバイスにリクエストを出す必要がある。このため、ホストＯＳがＩ／Ｏ操作を集中管理することで、ゲストＯＳで構成される各論理区画間でＩ／Ｏデバイスの共有を実現することになる。この結果、各論理区画からのＩ／Ｏ要求は、一旦、ホストＯＳの処理を待ってから物理的なＩ／Ｏデバイスに対して操作が行われるため、ホストＯＳの処理がオーバーヘッドとなって、Ｉ／Ｏ操作のレスポンスが低下するという問題がある。
【０００８】
上記ハイパバイザでは、論理区画を意識することができる特別なＩ／Ｏ装置（チャネル等）を用いることで、仮想計算機間でＩ／Ｏの共有を行っている。Ｉ／Ｏ装置がパーティションを意識してＩ／Ｏ操作を行い、各論理区画からＩ／Ｏデバイスを直接アクセスできる。しかし、特別なＩ／Ｏ装置を必要とするため製造コストが高く、また、汎用性も低いという問題がある。
【０００９】
一方、上記従来例のハイパバイザにAdvanced Switching等のＩ／Ｏスイッチを用いて論理区画間でＩ／Ｏデバイスの共有を行う場合には、次のような問題が生じる。
【００１０】
上述のAdvanced Switching（以下、単にＡＳとする）では、物理的なノード（またはデバイス）しか識別できないため、ＡＳではハイパバイザが提供する論理区画を識別できず

、上記ハイパバイザにＡＳをそのまま適用することはできない。このため、上述のＶＭｗａｒｅと同様にＡＳのＩ／Ｏ操作を集中管理するソフトウェアが必要になってしまい、このソフトウェアの処理がそのままＩ／Ｏアクセスのオーバーヘッドになるという問題を生じていた。

【００１１】

そこで本発明は、上記問題点に鑑みてなされたもので、ＡＳ等に代表される汎用性の高いＩ／Ｏスイッチを用いて、仮想計算機の論理区画間でＩ／Ｏの共有を実現する際のオーバーヘッドを低減することを目的とする。

【課題を解決するための手段】

【００１２】

本発明は、複数のＣＰＵコアと、前記ＣＰＵコアに接続されたＩ／Ｏブリッジと、前記ＣＰＵコアまたはＩ／Ｏブリッジからアクセス可能な主記憶と、を備えたＣＰＵモジュールと、前記ＣＰＵモジュールのＩ／Ｏブリッジとの間で送受信されるＩ／Ｏアクセス情報を中継する際に、前記論理区画毎に設定された仮想経路情報と、当該Ｉ／ＯブリッジからＩ／Ｏモジュールまでの経路情報とを前記Ｉ／Ｏアクセス情報の経路情報に付加して、論理区画毎に前記Ｉ／Ｏモジュールとの間のＩ／Ｏアクセス情報を切り換える仮想スイッチを備える。

【発明の効果】

【００１３】

したがって、本発明は、Ｉ／ＯスイッチでＩ／Ｏモジュールを複数の論理区画間で共有する際に、ＣＰＵモジュールに備えたＩ／Ｏブリッジを仮想スイッチとして機能させ、各論理区画毎に仮想経路情報を設定することで、ひとつのＣＰＵモジュールで複数の論理区画を提供する仮想計算機システムにおいてＩ／Ｏスイッチを用いたＩ／Ｏ共有を実現することができる。

【００１４】

さらに、Ｉ／Ｏブリッジのハードウェアレベルで論理区画間のＩ／Ｏ共有を行うようにすることで、論理区画を識別するためのソフトウェアは不要となってＩ／Ｏアクセスのオーバーヘッドを低減して高速化を図ることができる。

【発明を実施するための最良の形態】

【００１５】

以下、本発明の一実施形態を添付図面に基づいて説明する。

【００１６】

図１は第１の実施形態を示し、本発明を適用するブレードサーバシステム（物理計算機）のブロック図である。

【００１７】

ブレードサーバシステムは、複数のＣＰＵブレード＃０～２と、各種Ｉ／Ｏインターフェースを備えたＩ／Ｏブレード（またはＩ／Ｏカード）＃０～５と、ＣＰＵブレード＃０～２とＩ／Ｏブレード＃０～５を接続する複数のＩ／Ｏスイッチ＃０～２を備えたスイッチブレード３が接続されて、図示しない筐体に格納されている。なお、スイッチブレード３にはＩ／Ｏスイッチ＃０～２やＣＰＵブレード＃０～２に対するＩ／Ｏブレード＃０～５の割り当てを管理するファブリック管理サーバ４が接続され、ファブリック管理サーバ４には管理者に対する入出力が可能なコンソール５が接続されている。なお、ＣＰＵブレード（ＣＰＵモジュール）＃０～２、Ｉ／Ｏブレード（Ｉ／Ｏモジュール）＃０～５、スイッチブレード（スイッチモジュール）３は図示しないバックプレーンを介して接続されている。

【００１８】

本実施形態のブレードサーバシステムでは、後述するように、ＣＰＵブレード＃０～２

の計算機資源を複数の論理区画（パーティション）に分割した仮想計算機が稼動する。

【００１９】

以下に、ブレードサーバの各ブレードの概要を説明する。

【００２０】

＜ＣＰＵブレード＞

ＣＰＵブレード＃０～２はそれぞれ、複数のＣＰＵコア＃０、＃１を備えたＣＰＵ（所謂、マルチコアＣＰＵ）を複数備えている。なお、ＣＰＵブレード＃０～２は同一の構成であるので、ＣＰＵブレード＃０の構成についてのみ以下に説明する。

【００２１】

ＣＰＵ＃０は複数のＣＰＵコア＃０－０、＃０－１を有し、ＣＰＵ＃１は複数のＣＰＵコア＃１－０、＃１－１を有する。ＣＰＵ＃０、＃１は、フロントサイドバス１１を介してノースブリッジ１２に接続される。

【００２２】

ノースブリッジ１２はメモリバスを介して主記憶１３に接続され、ＣＰＵ＃０、＃１のメモリアクセス要求に応じて主記憶１３をアクセスする。また、ノースブリッジ１２はＩ／Ｏバス１４を介してＩ／ＯブリッジとしてのＡＳ（Advanced Switching）ブリッジ１５に接続され、ＣＰＵ＃０、＃１のＩ／Ｏアクセス要求に応じてＡＳブリッジ１５にアクセスを行う。ここでは、Ｉ／Ｏバス１４はＰＣＩ－ＥＸＰＲＥＳＳで構成される。

【００２３】

ＡＳブリッジ１５は、ＡＳ（Advanced Switching）の規格に準じて構成され、Ｉ／Ｏバス１４からのＰＣＩ－ＥＸＰＲＥＳＳ（以下、ＰＣＩｅとする）のパケット（Ｉ／Ｏアクセス情報）に経路情報を付加してＡＳパケットとしてから、後述するスイッチ＃０（ＳＷ０）～＃２（ＳＷ２）に送信する。また、ＡＳブリッジ１５は、後述するスイッチ＃０（ＳＷ０）～＃２（ＳＷ２）から受信したパケットを、ＰＣＩｅのパケットに変換し、ＣＰＵコア＃０－０～ＣＰＵコア＃１－１に割り当てられた論理区画を識別して配信する。このため、ＡＳブリッジ１５には、制御部１６とメモリ１７を備えて、スイッチブレード３に接続されている。

【００２４】

また、ＡＳブリッジ１５は、後述するように、ＡＳ（スイッチブレード３）側からのパケットを論理区画に割り振る仮想スイッチとして機能する。

【００２５】

さらに、ノースブリッジ１２とＡＳブリッジ１５は、ＣＰＵブレード＃０上のハードウェアを監視するＢＭＣ（Baseboard Management Controller または Blade Management Controller）１８に接続され、各ブリッジに接続されたハードウェアの監視が行われる。このＢＭＣ７は、基板上のハードウェアの電圧、温度、エラーなどを監視し、ＯＳやハイパバイザ等に通知するものである。なお、ＢＭＣ７は各ＣＰＵブレード毎に配置されて、各ＢＭＣ７はネットワークを介して接続されており、各ＢＭＣ７を管理するコンソール７０を有する。

【００２６】

本実施例では、コンソール７０とファブリック管理サーバ４は別としたが、同一のサーバとして実装しても良い。

【００２７】

＜スイッチブレード＞

スイッチブレード３には、複数のＡＳスイッチ（Ｉ／Ｏスイッチ）＃０（ＳＷ０）、＃１（ＳＷ１）、＃２（ＳＷ２）が配置される。ＡＳスイッチ＃０はＣＰＵブレード＃０～２に接続され、ＡＳスイッチ＃１、２はＩ／Ｏブレード＃０～５と接続されて、ＡＳスイッチ＃０～２が相互に接続される。また、ＡＳスイッチ＃０～２はそれぞれ相互に接続される。

【００２８】

各ＡＳスイッチ＃０～２は、複数のポートを有しており、例えば、８ポートで構成され

る。ＡＳスイッチ＃０は、ポート７をＣＰＵブレード＃０のＡＳブリッジ１５に接続し、ポート０をＣＰＵブレード＃１のＡＳブリッジ（図示省略）に接続し、ポート１をＣＰＵブレード＃２のＡＳブリッジ（図示省略）に接続し、ポート２をＡＳスイッチ＃１のポート７に接続し、ポート５をＡＳスイッチ＃２のポート１に接続している。

【００２９】

ＡＳスイッチ＃１は、ポート３～５をＩ／Ｏブレード＃３～５に接続し、ポート７をＡＳスイッチ＃０のポート２に接続し、ポート６をＡＳスイッチ＃１のポート２に接続している。

【００３０】

ＡＳスイッチ＃２は、ポート３～５をＩ／Ｏブレード＃０～２に接続し、ポート１をＡＳスイッチ＃０のポート５に接続し、ポート２をＡＳスイッチ＃１のポート６に接続している。

【００３１】

ここで、ＡＳにおけるパケット形式とルーティングについて述べる。ＡＳスイッチ＃０～２を通過する従来のAdvanced Switchingパケット（以下、ＡＳパケットとする）は、図５～６で示すように、ヘッダー部Ｐ０とデータ部Ｐ１から構成されたＰＣＩｅのパケットに、ＡＳヘッダーが付加されている。

【００３２】

ＡＳパケットには単一の宛先に送られるユニキャストパケットと、複数の宛先に送られるマルチキャストパケットがある。ユニキャスト／マルチキャストは、ＡＳヘッダー内のプロトコルインタフェース（コマンド種別を表す）フィールド（図示しない）により区別される。以下では、各々の場合のルーティング方式を説明する。

（１）ユニキャスト（マルチキャストでない）パケットの場合

ユニキャストパケットでは、経路情報として各スイッチでの切り換え数を示すターンプール値ＴＰ１～ｎ、パケットの転送方向（上り／下り）を表すビットＤＩＲ等の情報を持つ（図では、ＴＰ値、ＤＩＲ以外のヘッダー情報は略す）。ＤＩＲは０が下り、１が上りをあらわす。このターンプール値ＴＰ１～ｎは、ＡＳパケットが通過するＡＳスイッチの数に応じてＡＳブリッジ１５またはＩ／Ｏブレード＃０～５が付加する。

【００３３】

そして、本発明のＡＳブリッジ１５は、図５、図６で示すように、ターンプール値ＴＰ１～ｎに加えて、論理区画を識別するための仮想スイッチターンプール値ＴＰ０を設定する。

【００３４】

ＡＳスイッチ＃０～２は、図５に示す下りのＡＳパケットと図６に示す上りのＡＳパケットのターンプール値に基づいて転送先を決定する。なお、本実施形態では、ＣＰＵブレードからＩ／Ｏブレードへ向かうパケットを下りのＡＳパケットとし、逆に、Ｉ／ＯブレードからＣＰＵブレードへ向かうパケットを上りのＡＳパケットとする。

【００３５】

ＡＳスイッチ＃０～２は、下りＡＳパケットの場合は、ＡＳパケットを受信した隣のポートから時計回りに０からターンプール値まで数えたポートを転送先として決定し、ＡＳパケットを転送する。下りのパケット（ＤＩＲ＝０）の場合、ＡＳスイッチを一段渡る毎に、ターンプールを（ＴＰ１～ＴＰｎの順番に）左から右に使用する。

【００３６】

例えば、図４において、ＡＳスイッチ＃０がポート７に接続されたＣＰＵブレード＃０から受信した下りＡＳパケットのターンプール値が２であれば、時計回りに０－１－２とターンプール値の数だけ０からカウントしてポート２をＡＳパケットの転送先とする。

【００３７】

それに対して上りパケット（ＤＩＲ＝１）の場合、ターンプールの値は、反時計回りに数えられ、ターンプールは（ＴＰｎ～ＴＰ１の順番に）右から左に使用される。

【００３８】

ＡＳスイッチ＃０が、ポート２に接続されたＡＳスイッチ＃１から受信した上りＡＳパケットのターンプール値が２であれば、反時計回りに１－０－７とターンプール値の数だけ０からカウントしてポート７をＡＳパケットの転送先とする。

【００３９】

以上により、要求（下り）パケットとそれに対する応答（上り）パケットのターンプール値は同一とすることができ、ＤＩＲビットを（０から１に）反転するだけで応答パケットのルーティング情報を作成することができる。これにより、ブリッジにおける応答パケットの生成を容易に行うことができる。

【００４０】

なお、ターンプール値ＴＰ０～ｎの詳細については、後述する。

【００４１】

なお、ＰＣＩｅパケットのデータ部Ｐ１には、データやコマンド（Ｉ／Ｏアクセスコマンド）等が格納される。

（２）マルチキャストパケットの場合

ＡＳでは複数の宛先に同一のデータやイベントを送付するマルチキャストを行うことができる。マルチキャストの宛先はＭＧＩＤ（マルチキャストグループＩＤ）と呼ばれるシステム内で一意な番号により管理される。同一のＭＧＩＤを持つパケットは、常に同一の宛先にマルチキャストされる。それを実現するために、あらかじめ各ＡＳスイッチに、マルチキャストグループ毎に、どのポートにデータを出力する必要があるかがテーブルで設定される。図１７にマルチキャストパケットの形式、図１８にマルチキャストを行うＡＳスイッチを示す。

【００４２】

図１７に示すように、マルチキャストパケットでは、パケットのＡＳヘッダーでは、マルチキャストグループＩＤ（ＭＧＩＤ）をルーティング情報として持つ（ＭＧＩＤ以外のヘッダー情報は省略する）。ＡＳスイッチでは、各ＭＧＩＤにおいて、ＡＳスイッチのどのポートにパケットを送付すれば良いかを示すマルチキャストテーブルを持つ。図１８にマルチキャストテーブル１８０の構成とＡＳスイッチの動作を示す。マルチキャストテーブル１８０では、各ＭＧＩＤにおいて、ＡＳスイッチの各物理ポートにパケットへパケットを出力する／しないを、ビットマップで表す。例えば図１８では、ＭＧＩＤが２５のパケットはマルチキャストテーブル１８０の該当するエントリが１であるポート３、４、５に出力される。

【００４３】

本発明では、各ＡＳスイッチに加え、ブリッジにおいても、どの論理区画にパケットをマルチキャストするかを示す仮想スイッチのマルチキャストテーブルを持つことにより、論理区画に向けたマルチキャストを可能にする。仮想スイッチのマルチキャストテーブルテーブルの詳細は後述する。

【００４４】

マルチキャストテーブル１８０はＡＳスイッチ＃０～２毎におかれ、システム初期化時や構成変更時にファブリック管理サーバ４により初期化される。

【００４５】

なお、Ｉ／Ｏブレード側からのマルチキャストは、例えば、ネットワークインターフェース（図１のＥｔｈｅｒ　Ｉ／Ｆを備えたＩ／Ｏブレード＃０、＃５等）からの受信データやＩ／Ｏブレード側からの割り込みであり、Ｉ／Ｏブレード側で宛先の論理区画を判断できない場合に使用される。

【００４６】

＜Ｉ／Ｏブレード＞

ＡＳスイッチ＃０～２を介してＣＰＵブレード＃０～２からアクセスされるＩ／Ｏブレード＃０～５には、各種Ｉ／Ｏインターフェースと、ＡＳスイッチ＃０～２と各Ｉ／Ｏインターフェースを接続するためのＬＥＡＦブリッジが設けられる。

【００４７】

Ｉ／Ｏブレード＃０～５のＬＥＡＦブリッジは、上記図４で示したように、Ｉ／Ｏブレードが受信した下りＡＳパケットのターンプール値ＴＰ０～ｎを一旦記憶し、この下りＡＳパケットの返信となる上りＡＳパケットに、受信したＡＳパケットのターンプール値ＴＰ０～ｎを設定するとともに、上述したようにＤＩＲビットを反転する。

【００４８】

そして、各Ｉ／Ｏブレード＃０～５の各ＬＥＡＦブリッジは、ＡＳパケットからＡＳヘッダを削除したＰＣＩｅパケットを各Ｉ／Ｏインターフェースに渡し、各Ｉ／ＯインターフェースはＰＣＩｅのデバイスとして機能する。Ｉ／Ｏインターフェースとしては、図１に示すように、ＦＣ（Fibre Channel）インターフェース、ＳＡＳ（Serial Attached SCSI）インターフェース、イーサネット（登録商標）インターフェース、ＳＡＴＡ（Serial ＡＴＡ）インターフェース、ＳＣＳＩインターフェースなどで構成される。

【００４９】

＜ファブリック管理サーバ＞

スイッチブレード３に接続されたファブリック管理サーバ４は、ＣＰＵ、メモリ、インターフェース（図示省略）を備え、各ＣＰＵブレード＃０～２上で稼動する論理区画が使用するＩ／Ｏブレード＃０～５の割り当ての管理、マルチキャストの宛先の管理、ＡＳスイッチ＃０～２やＡＳブリッジの設定を行う。

【００５０】

ファブリック管理サーバは、ＡＳファブリック経由で管理を行うＩｎ－ｂｏｕｎｄ管理方式、ＡＳファブリック以外の管理用ネットワーク（イーサネット（登録商標）等）（図示しない）経由で管理を行うＯｕｔ－ｏｆ－ｂｏｕｎｄ管理の２方式がある。

【００５１】

＜仮想計算機システム＞

次に、物理計算機を構成するＣＰＵブレード＃０～２上で稼動するソフトウェアについて、図２を参照しながら詳述する。

【００５２】

各ＣＰＵブレード＃０～２上ではハイパバイザ２００がファームウェアとして稼動しており、ハイパバイザ２００は物理計算機１００を２つ以上の論理区画（ＬＰＡＲ：Ｌｏｇｉｃａｌ　ＰＡＲｔｉｔｉｏｎ）ＬＰＡＲ０（２１０）～ＬＰＡＲｍ（２１ｍ）に分割し、計算機資源の割り当てを管理する。

【００５３】

論理区画ＬＰＡＲ０～ＬＰＡＲｍのそれぞれでＯＳ０（２２０）～ＯＳｍ（２２ｍ）を動作させ、各ＯＳ上でそれぞれアプリケーション０（２３０）～アプリケーションｍ（２３ｍ）を動作させる。

【００５４】

ハイパバイザは、各論理区画ＬＰＡＲ（２１０～２１ｍ）に対して各ＣＰＵブレード＃０～２のＣＰＵコア、主記憶１３、ＡＳブリッジ１５に接続されたＩ／Ｏブレード＃０～５の割り当てのリソース（計算機資源）を割当てる。

【００５５】

なお、ＣＰＵ＃０、＃１は複数のＣＰＵコアを持つ場合を示したが、ひとつのＣＰＵがひとつのＣＰＵコアを備えるものであっても良い。また複数のＣＰＵがＳＭＰを構成する場合を示したが、単一ＣＰＵでも良い。各論理区画のＣＰＵへの割り当ては、必ずしもコア毎ではなくて良く、単一のコアを複数の論理区画が時分割で共有しても良い。

【００５６】

＜本願発明の概要＞

次に、本発明の要部となるＡＳブリッジとＡＳパケットの概要について、以下に説明する。

【００５７】

各ＣＰＵブレード＃０～２に配置されたＡＳブリッジ１５は、図７で示すように、仮想のＡＳスイッチとして機能する。物理的には、図１、図４で示したように、ＡＳブリッジ

１５とＡＳスイッチ＃０がＡＳにより接続され、ＡＳブリッジ１５とノースブリッジ１２がＰＣＩｅにより接続されている。

【００５８】

図４において、ターンプール値ＴＰ１～ｎに基づいてＡＳ側（ＡＳスイッチ＃０～２）からＣＰＵブレード＃０を見たとき、ターンプール値はひとつのＡＳスイッチ＃０のポートを指し示すだけであるので、ＡＳスイッチ＃０からＣＰＵブレード＃０上の複数の論理区画を識別することはできない。これが、前記従来例の課題であり、ひとつのＣＰＵブレード上で稼動する複数の論理区画（ＯＳ）からＩ／Ｏアクセスを行うと、ＡＳスイッチ側はＩ／Ｏアクセス元の論理区画に返信を行うことはできなかった。

【００５９】

そこで、図７のように、ＡＳブリッジ１５を仮想的なＡＳスイッチ（以下、単に仮想スイッチとする）ＳＷｖ１として機能させ、且つ、図５、図６のように実際のＡＳスイッチ＃０～２のターンプール値ＴＰ１～ＴＰｎに加えて、論理区画毎に仮想的に接続された仮想スイッチＳＷｖ１内のターンプール値ＴＰ０（仮想経路情報）を付加することで、ＣＰＵブレード＃０では受信したＡＳパケットがどの論理区画に向けて送信されたものであるかを識別できるのである。

【００６０】

図７において、ＣＰＵブレード＃０からＩ／Ｏブレード＃５に向かう下りのＡＳパケットは、１つの仮想スイッチＳＷｖ１と２つの物理的なＡＳスイッチ＃０、＃１を通過するので、図５で示すように、ＡＳパケットには第１、第２、第３のターンプール値ＴＰ０、１、２を設定する。なお、ターンプール値は通過するＡＳスイッチの数（ｎ）だけＡＳブリッジ１５が付加するもので、図５の例では、３つのターンプール値ＴＰ０～ＴＰ２を付加する。

【００６１】

ＣＰＵブレード＃０のＡＳブリッジ１５は、ハイパバイザ２００が生成した論理区画の数に応じて仮想スイッチＳＷｖ１のポートを生成する。図７の例では、２つの論理区画＃０、＃１を生成したのでポート１、２がＡＳブリッジ１５により生成される。

【００６２】

ＣＰＵ＃０、１（または各ＣＰＵコア）で稼動する複数の論理区画からＩ／Ｏアクセスがあると、ＡＳブリッジ１５は第１のターンプール値ＴＰ０にアクセス元となる論理区画からＡＳスイッチへ向かう仮想スイッチＳＷｖ１のターンプール値ＴＰ０を設定する。例えば、図７の論理区画＃０からのアクセスの場合にはＴＰ０＝０とし、論理区画＃１からのアクセスの場合にはＴＰ０＝１とする。このターンプール値の設定は、上述の図４と同様である。

【００６３】

そして、ＡＳブリッジ１５は、Ｉ／Ｏアクセス先を読み込んで、ＣＰＵブレード＃０からアクセス先のＩ／Ｏブレードまでのターンプール値ＴＰ１～ｎを設定する。

【００６４】

例えば、図４～図７では、ＣＰＵブレード＃０の論理区画＃０がＩ／Ｏブレード＃５にアクセスする場合を示す。

【００６５】

図７において、ＡＳブリッジ１５は、仮想のポート０～２を生成し、仮想ポート０をＡＳスイッチ＃０のポート７に接続し、仮想ポート１を論理区画＃１に接続し、仮想ポート２を論理区画＃０に接続する。

【００６６】

論理区画から見て１番目のスイッチとなる仮想スイッチＳＷｖ１では、論理区画＃０の隣のポート０に転送するので、第１のターンプール値である仮想スイッチターンプール値ＴＰ０に「０」を設定する。ＡＳ側では、ＡＳスイッチ＃０からＡＳスイッチ＃１のポート３からＩ／Ｏブレード＃５にＡＳパケットを転送することになる。

【００６７】

このため、２番目のスイッチとなる物理的なＡＳスイッチ＃０ではポート７からポート２へＡＳパケットを転送するので、ＡＳブリッジ１５は、時計回りに０からカウントした「２」を第２のターンプール値ＴＰ１に設定する。

【００６８】

次に、３番目のスイッチとなる物理的なＡＳスイッチ＃１ではポート７からポート３へＡＳパケットを転送するので、ＡＳブリッジ１５は、時計回りに０からカウントした「３」を第３のターンプール値ＴＰ１に設定する。

【００６９】

こうして、ＡＳブリッジ１５は図５で示すように、下りＡＳパケットのターンプール値ＴＰ０～ＴＰ２を「０、２、３」と設定する。さらに下りパケットであることを表すために、方向ビットＤＩＲを１に設定する。

【００７０】

ＣＰＵブレード＃０のＡＳブリッジ１５は、図１、図５で示すように、ＡＳスイッチ＃０のポート番号７に接続されている。論理区画＃０から見て２番目のスイッチとなるＡＳスイッチ＃０は、第２のターンプール値ＴＰ１を読み込んで、図４、図７の時計回りに０からターンプール値ＴＰ１の２まで数え、ポート番号２のポートを当該ＡＳパケットの転送先に設定し、ポート番号２へＡＳパケットを転送する。

【００７１】

ＡＳスイッチ＃０のポート２は、ＡＳスイッチ＃１のポート７に接続されているので、ＡＳパケットはＡＳスイッチ＃１に転送される。

【００７２】

ＡＳスイッチ＃１は、論理区画＃０から見て３番目のスイッチであるので、第３のターンプール値ＴＰ２を読み込んで、図４、図７の時計回りに０からターンプール値ＴＰ２の３まで数え、ポート番号３のポートを当該ＡＳパケットの転送先に設定し、ポート番号３へＡＳパケットを転送する。ＡＳスイッチ＃１のポート３は、Ｉ／Ｏブレード＃５が接続されており、ＡＳパケットはＣＰＵブレード＃０からＩ／Ｏブレード＃５に送られる。こうして、下りのＡＳパケットは、各ＡＳスイッチがターンプール値を順次読み込んで時計回りにカウントしたポートに順次転送する。

【００７３】

逆に、Ｉ／ＯブレードＩ／Ｏブレード＃５からＣＰＵブレード＃０へのＡＳパケットは、図６で示す上りのＡＳパケットのように、下りのＡＳパケットのターンプール値ＴＰ０～ｎを、方向ビットＤＩＲを反転（０から１）して設定する。方向ビットＤＩＲが１の上りパケットのターンプールは右から左に解釈され、さらに各ＡＳスイッチでターンプール値は反時計回りに数えられる。

【００７４】

つまり、図４、図７の場合の上りＡＳパケットのターンプール値は、図６のように、第１のターンプール値（方向ビットＤＩＲが１の上りパケットの場合一番右の）ＴＰ２が「３」、第２のターンプール値ＴＰ１が「２」、第３の値には仮想スイッチターンプール値ＴＰ０が「０」に設定される。なお、上りＡＳパケットのターンプール値ＴＰ０～ｎの設定は各Ｉ／ＯブレードのＬＥＡＦブリッジが行う。

【００７５】

そして、Ｉ／Ｏブレード＃５が送信したＡＳパケットは、Ｉ／Ｏブレード＃５から見て１番目のＡＳスイッチとなるＡＳスイッチ＃１が、第１のターンプール値ＴＰ２＝３を読み込んで、図４、図７の反時計回りに０からターンプール値ＴＰ１の３まで数え、ポート番号７のポートを当該ＡＳパケットの転送先に設定し、ポート番号７へＡＳパケットを転送する。

【００７６】

ＡＳスイッチ＃１のポート７は、ＡＳスイッチ＃０のポート２に接続されているので、ＡＳパケットはＡＳスイッチ＃０に転送される。

【００７７】

ＡＳスイッチ＃０は、Ｉ／Ｏブレード＃５から見て２番目のＡＳスイッチであるので、第２のターンプール値ＴＰ１を読み込んで、図４の反時計回りに０からターンプール値ＴＰ１の２まで数え、ポート番号７のポートを当該ＡＳパケットの転送先に設定し、ポート番号７へＡＳパケットを転送する。

【００７８】

図７で示すように、ＡＳスイッチ＃０のポート７は仮想スイッチＳＷｖ１に接続されており、仮想スイッチＳＷｖ１は、Ｉ／Ｏブレード＃５から見て３番目のＡＳスイッチであるので、第３のターンプール値ＴＰ０を読み込んで、図４の反時計回りに０からターンプール値ＴＰ０の０まで数え、ポート番号２を当該ＡＳパケットの転送先に設定し、ポート２と仮想的に接続された論理区画＃０へＡＳパケットを転送する。

【００７９】

こうして、上りのＡＳパケットは、各ＡＳスイッチがターンプール値を反時計回りにカウントしたポートに順次転送することで、最後に仮想スイッチターンプール値ＴＰ０とＡＳブリッジ１５の仮想スイッチＳＷｖ１を用いることで、ＡＳスイッチ側からＩ／Ｏアクセス元の論理区画＃０を識別してＡＳパケットを確実に返信することができるのである。

【００８０】

なお、ＡＳスイッチを管理するファブリック管理サーバ４は、ＡＳスイッチ＃０～＃２に加えて、各ＣＰＵブレードのＡＳブリッジ１５が提供する仮想スイッチＳＷｖ１も、ＡＳスイッチ群のトポロジとして管理する。

【００８１】

以上は宛先が単一のユニキャストパケットであるが、以下ではマルチキャストの場合について述べる。マルチキャストでは、各パケットが、マルチキャストの宛先を一意に表すＭＧＩＤ（マルチキャストグループＩＤ）を持つ。各スイッチは、あらかじめ決められているマルチキャストテーブル１８０を引き、ＭＧＩＤに対応する出力ポート全てにパケットを同時に出力する。

【００８２】

本発明では、各ＣＰＵブレードに置かれた、ＡＳブリッジ１５にも仮想スイッチＳＷｖ１のマルチキャストテーブルを新たに置き、同一ＣＰＵブレード上の複数の論理区画へのマルチキャストをサポートする。各ブリッジは、マルチキャストパケットを受け取ると、パケットのＭＧＩＤに対応する仮想スイッチのマルチキャストテーブルエントリを読み、どの論理区画にマルチキャストをする必要があるかを決める。その後、該当する論理区画の持つ主記憶空間へのデータの書き込み、該当する論理区画へのイベント配信が行われる。

【００８３】

なお、ＡＳスイッチを管理するファブリック管理サーバ４は、ＡＳスイッチ＃０～＃２に加えて、各ＣＰＵブレードのＡＳブリッジ１５が提供する仮想スイッチＳＷｖ１のマルチキャストテーブルも、ＡＳスイッチ群のマルチキャストテーブルとして管理する。

【００８４】

＜ＡＳブリッジの詳細＞

次に、図３を参照しながら、仮想スイッチとして機能するＡＳブリッジ１５の詳細について説明する。なお、図３はＣＰＵブレード＃０のＡＳブリッジ１５を主体にした機能ブロック図であり、他のＣＰＵブレードも同様に構成される。また、図３においてＡＳブリッジ１５とＣＰＵ＃０、＃１または主記憶１３との間で行われる通信は、図１で示したようにノースブリッジ１２を介して行うものである。

【００８５】

ＡＳブリッジ１５には、図１で示した制御部１６とメモリ１７を備え、ＡＳスイッチ＃０～２側から見て複数の論理区画を複数のノードとして見せる仮想ＡＳスイッチとして機能する仮想スイッチを提供する。このため、ＡＳブリッジ１５のメモリ１７には、ハイパバイザ２００上で稼動する論理区画と仮想スイッチの仮想ポート番号の対応関係を設定するパーティション管理表１５５と、各論理区画の主記憶１３上のアドレスを管理するアド

レス管理表１６０と、論理区画からのＩ／Ｏアクセスコマンドを受け付けるＩ／Ｏアクセスコマンドレジスタ１５３と、Ｉ／Ｏブレード側からのイベント（割り込みやエラー通知など）を受け付けるイベントレジスタ１５６と、Ｉ／Ｏブレードからのイベント転送があった場合に、転送先の論理区画を決定するための宛先ＬＰＡＲ番号レジスタ１５８がＢＭＣ１８によって生成される。宛先ＬＰＡＲ番号レジスタ１５８は、イベントのマルチキャスト転送をサポートするために、ビットマップで実装される。マルチキャストの場合には、複数のビットが１となる。さらに、マルチキャストの場合に、ＭＧＩＤと宛先論理区画の関係をあらわす、仮想スイッチマルチキャストテーブル１７０がファブリック管理サーバにより作成される（この処理は新しいマルチキャストグループが定義されるたびに行われる）。仮想スイッチマルチキャストテーブル１７０は、図１９で示すように構成され、マルチキャストグループＩＤ（ＭＧＩＤ）毎に、各論理区画（図中出力ＬＰＡＲ）に対応したビットマップで構成される。この仮想スイッチマルチキャストテーブル１７０は、各論理区画にマルチキャスト転送を行うか否かを示すビットが設定される。このビットは１のときにマルチキャスト転送を許可し、０のときにマルチキャスト転送を禁止する。図１９において、ＭＧＩＤ＝１のときには論理区画＃１へのマルチキャスト転送を許可し、論理区画＃０へのマルチキャスト転送を禁止する。

【００８６】

さらに、ファブリック管理サーバ４が各ＣＰＵブレードにおかれたハイパバイザとＩｎ－ｂｏｕｎｄで（ＡＳのファブリックを介して）通信する場合は、宛先ＬＰＡＲ番号レジスタ１５８には、各論理区画に対応するビットに加えて、ハイパバイザに対応するビットを持つ。この場合、ハイパバイザ宛にイベントを送る場合は、ハイパバイザに該当するビットをＯＮにする。

【００８７】

上記パーティション管理表１５５、アドレス管理表１６０、仮想スイッチマルチキャストテーブル１７０及び各レジスタを除くＡＳブリッジ１５の構成要素は、上記図１の制御部１６に相当する。

【００８８】

まず、パーティション管理表１５５は、図８で示すように、論理区画（パーティション）の番号に割り当てられた仮想スイッチの仮想ポート番号と、この論理区画が有効か無効かの状態を示すフィールドから構成されている。このパーティション管理表１５５は、ハイパバイザ２００が論理区画を生成、削除するときに更新される。

【００８９】

さらに、Ｉｎ－ｂｏｕｎｄ管理を行う場合には、ハイパバイザに対応する仮想スイッチの情報も示される。

【００９０】

次に、アドレス管理表１６０は、図９で示すように、主記憶１３上の各パーティションの位置を示すため、論理アドレス空間上の論理区画のベースアドレスと、各論理区画のサイズを論理区画番号毎に設定したものである。このアドレス管理表１６０も、論理区画の生成や削除の際にハイパバイザ２００によって更新される。なお、主記憶１３の論理アドレス空間は、例えば、図１０で示すように構成され、各論理区画（ＬＰＡＲ）のベースアドレスとサイズは、ハイパバイザ２００によって管理される。なお、各論理区画のアドレスマップ上には、ハイパバイザ２００によってＩ／Ｏレジスタがそれぞれマッピングされる。

【００９１】

Ｉ／Ｏアクセスコマンドレジスタ１５３は、各論理区画上で稼動するＯＳからのＩ／Ｏアクセスコマンドを受け付ける。Ｉ／Ｏアクセスコマンドは、例えばＩ／ＯレジスタのＰＩＯ要求（Ｉ／Ｏデバイスに持つコントロール／ステータスレジスタへの読み書き）もしくはＤＭＡ転送（ＲＥＡＤ／ＷＲＩＴＥ）要求や、ターゲットとなるＩ／Ｏブレードの情報が含まれる。

【００９２】

イベントレジスタ１５６には、ディスクアクセスの終了や、イーサネット（登録商標）インターフェース等のネットワークからの通知（データ／コマンド）に起因する、Ｉ／Ｏブレードからの割り込み、もしくはエラー通知を受け付ける。そして、ハイパバイザ２００を経由して各論理区画に転送される。イベントはユニキャスト／マルチキャストの場合がある。ユニキャストの場合は、仮想スイッチのターンプール情報より求められた宛先論理区画の番号は、宛先ＬＰＡＲ番号レジスタ１５８に設定される。マルチキャストの場合は、パケットのＭＧＩＤにより、仮想スイッチのマルチキャストテーブル１７０を引き、宛先の論理区画（ＬＰＡＲ）のビットマップが求められ、宛先ＬＰＡＲ番号レジスタ１５８に設定される。図１１に示す宛先ＬＰＡＲ番号レジスタ１５８に設定された論理区画（ＬＰＡＲ）毎のビットマップに基づいてハイパバイザ２００が転送を行う。仮想スイッチマルチキャストテーブル１７０や、図１１の宛先ＬＰＡＲ番号レジスタ１５８は、各論理区画毎にビットを有し、論理区画毎に対応するビットが１であればマルチキャストを行い、０であればマルチキャストを禁止することを示す。この仮想スイッチのマルチキャストテーブル１７０は、ファブリック管理サーバ４によって管理される。

【００９３】

次に、制御部１６を構成する各部について説明する。まず、論理区画からＩ／Ｏアクセスがあると、ＰＣＩｅのパケットがＡＳブリッジ１５に送信される。ＡＳブリッジ１５は、受信したＰＣＩｅのパケットからＩ／Ｏアクセスコマンド及びターゲットのＩ／Ｏブレードの情報を表すＩ／Ｏデバイスのアドレスを、Ｉ／Ｏアクセスコマンドレジスタ１５３に格納する。また、ＰＣＩｅのパケットに含まれるＩ／ＯレジスタアドレスやＤＭＡを行う主記憶のアドレスを抽出してアドレスデコード部１５１に入力する。これにより、Ｉ／Ｏアクセスを発行した論理区画を特定することができる。

【００９４】

アドレスデコード部１５１は、アドレス管理表１６０を参照して、抽出されたアドレスがどの論理区画のものあるかを判定する。つまり、抽出されたアドレスがアドレス管理表１６０のベースアドレス＋サイズの範囲内となる論理区画を、Ｉ／Ｏアクセスを行う論理区画として判定し、この論理区画番号を仮想スイッチターンプール情報付加部１５２へ送る。

【００９５】

また、Ｉｎ－ｂｏｕｎｄ管理を行う場合には、アドレス管理表１６０には、ハイパバイザのアドレス範囲の情報を持ち、ハイパバイザに該当する仮想スイッチターンプール情報を付加する。

【００９６】

なお、アドレスデコード部１５１で行う論理区画の判定は、Ｉ／ＯレジスタのアドレスまたはＤＭＡアドレスのアドレスとアドレス管理表１６０に設定されたアドレスのうち、予め設定した上位ビット（例えば、上位８ビット）について比較を行えばよい。

【００９７】

仮想スイッチターンプール情報付加部１５２は、受信した論理区画番号からパーティション管理表１５５を参照し、論理区画番号に対応する仮想スイッチの仮想ポート番号を決定し、これを上記図５に示した仮想スイッチターンプール値ＴＰ０としてパケット組み立て部１５４に送る。

【００９８】

パケット組み立て部１５４は、Ｉ／Ｏアクセスコマンドレジスタ１５３からターゲットのＩ／Ｏブレードの情報を読み込んで、図５に示したターンプール値ＴＰ１～ＴＰｎを経路情報として決定する。ＡＳブリッジ１５からターゲットのＩ／Ｏブレードまでの経路情報は、パケット組み立て部１５４が予めファブリック管理サーバ４から取得した経路情報に基づいて決定される。つまり、ＡＳブリッジ１５は、Ｉ／Ｏバス１４から受信したＩ／Ｏアクセス要求に含まれるアクセス先（ターゲット）のＩ／Ｏブレード（デバイス）の識別子に基づいて、ファブリック管理サーバ４からあらかじめ取得していた経路情報に基づき、ＣＰＵブレードからＩ／Ｏブレードまでの経路情報を決定する。

【００９９】
パケット組み立て部１５４は、ＡＳブリッジ１５から目的のＩ／Ｏブレードまでのターンプール値ＴＰ１～ＴＰｎに、仮想スイッチターンプール値ＴＰ０を読み込んで、図５で示すようにターンプール値ＴＰ１～ＴＰｎの先頭に付加し、ＡＳパケットのヘッダー部を生成する。
【０１００】
そして、パケット組み立て部１５４は、受信したＰＣＩｅのパケットの先頭に、仮想スイッチターンプール値ＴＰ０、ターンプール値ＴＰ１～ＴＰｎ、ＤＩＲビット０を、ＡＳパケットのヘッダー部としてＡＳパケットを生成してから、ＡＳスイッチ＃０～２へ送信する。
【０１０１】
ＡＳブリッジ１５は、上記アドレスデコード部１５１、仮想スイッチターンプール情報付加部１５２、Ｉ／ＯアクセスコマンドレジスタＩ５３、パケット組み立て部１５４により下りＡＳパケットを生成し、ＡＳブリッジ１５に接続されたＡＳスイッチ＃０にＡＳパケットを送る。
【０１０２】
次に、ＡＳブリッジ１５がＡＳスイッチ＃０からＡＳパケットを受信した場合について説明する。
【０１０３】
ＡＳブリッジ１５は、ＡＳスイッチ＃０から上りＡＳパケットを受信すると、コマンド解析部１５７で上りＡＳパケットを受け付けて、ＡＳパケット内のＰＣＩｅパケットに含まれるコマンドを解析する。コマンド解析部１５７は、コマンドの種類がＤＭＡ転送、割り込みやエラー通知等のイベント処理の何れであるかを判定する。さらにマルチキャストが必要かどうか判断する。
【０１０４】
そして、コマンド解析部１５７は、受信したＡＳパケットのコマンドがＤＭＡ転送またはマルチキャストＤＭＡであれば、ＡＳパケットから抽出したＰＣＩｅパケットをＤＭＡ処理部１６２へ送る。受信したＡＳパケットが割り込みやエラー通知等のイベント処理もしくはマルチキャストイベントの場合には、ＡＳパケットから抽出したＰＣＩｅパケットをイベントレジスタ１５６に送る。
【０１０５】
宛先ＬＰＡＲ番号抽出部１５９は、ユニキャストとマルチキャストの場合で異なる動作をする。
【０１０６】
ユニキャストの場合、上りＡＳパケットのヘッダー部から図６で示したように、最後のターンプール値に格納された仮想スイッチターンプール値ＴＰ０を抽出する。そして、宛先ＬＰＡＲ番号抽出部１５９は、パーティション管理表１５５から仮想スイッチターンプール値ＴＰ０に対応する論理区画番号を取得し、受信したＡＳパケットの宛先となる論理区画を特定する。宛先ＬＰＡＲ番号抽出部１５９は、特定した論理区画の番号を表すビットマップ（この場合は何れか１ビットのみがセットされている）をアドレス管理表１６０と宛先ＬＰＡＲ番号レジスタ１５８に送る。
【０１０７】
マルチキャストの場合、宛先ＬＰＡＲ番号抽出部１５９は、ＡＳパケットのヘッダー部からＭＧＩＤを求め、仮想スイッチのマルチキャストテーブル１７０を引き、受信したパケットの宛先となる論理区画の番号（複数）を表すビットマップを、アドレス管理表１６０と宛先ＬＰＡＲ番号レジスタ１５８に送る。ここで、上流のスイッチでのみマルチキャストが行われ、当該ＣＰＵでは宛先のＬＰＡＲが一つのみの場合があるため、仮想スイッチのマルチキャストテーブルのビットパターンで１ビットのみがセットされている場合もある。
【０１０８】

ＤＭＡ処理部１６２は、コマンドがＤＭＡ転送の場合には、宛先ＬＰＡＲ番号抽出部１５９が決定した論理区画番号からアドレス管理表１６０に基づいてベースアドレスを求め、ＰＣＩｅパケット内のＤＭＡアドレスから実際にアクセスする主記憶１３上のアドレスをアドレス変換部１６１で変換する。そして、ＡＳブリッジ１５は、アドレス変換部１６１が決定した主記憶１３上のアドレスに対してＤＭＡ転送（ＲＥＡＤまたはＷＲＩＴＥ）を実施する。

【０１０９】

このとき、アドレス変換部１６１では、ＤＭＡ転送を行う主記憶１３上のアドレスが、転送を行う論理区画のベースアドレスとサイズから求められるアドレス範囲内であるかを判定する。そして、判定結果が該当する論理区画内であればＤＭＡ転送を行い、ＤＭＡ転送を行う主記憶１３上のアドレスが該当する論理区画を超えていれば、ＤＭＡ転送を中止し、異なる論理区画に操作が行われるのを防いで仮想計算機の信頼性を確保する。

【０１１０】

Ｉｎ－ｂｏｕｎｄ管理を行う場合に、ファブリック管理サーバ４がハイパバイザに対して出したＤＭＡについては、上記の論理区画のアドレス範囲チェックは行わない。

【０１１１】

次に、ＡＳパケットのコマンドがマルチキャストＤＭＡの場合は、ＤＭＡ処理部１６２から複数回書込処理部１６３へＰＣＩｅパケットが転送される。複数回書込処理部１６３は、図１１に示した宛先ＬＰＡＲ番号レジスタ１５８を参照し、論理区画番号に対応するビットが１となっている論理区画ついて、アドレス変換部１６１に転送を行う論理区画のＩ／Ｏレジスタの主記憶１３上のアドレスを問い合わせ、複数の論理区画に対して複数回の書込を実施する。このマルチキャストの場合も、アドレス変換部１６１では書込の対象となる論理区画のアドレスが正当であるか否かを上記ＤＭＡ転送と同様に判定し、不正な書込が行われるのを防止する。

【０１１２】

次に、ＡＳパケットのコマンドが割り込みやエラー通知等のイベント処理（マルチキャスト含む）の場合は、コマンド解析部１５７は上りＡＳパケットから抽出したＰＣＩｅパケットをイベントレジスタ１５６に送り、イベントレジスタ１５６からハイパバイザ２００へ通知を行う。ハイパバイザは宛先ＬＰＡＲ番号レジスタ１５８を参照し、ビットマップが１である論理区画にイベントを転送する。

【０１１３】

以上のように、ＡＳブリッジ１５は、Ｉ／ＯバスＩ／Ｏ１４からＩ／Ｏアクセス要求を受け取ると、目的のＩ／Ｏブレードの識別子に基づいてターンプール値ＴＰ０～ｎと、Ｉ／Ｏアクセスを要求した論理区画に基づく仮想スイッチターンプール値ＴＰ０を生成する。そして、ＡＳブリッジ１５は、Ｉ／ＯバスＩ／Ｏ１４から受信したＩ／Ｏアクセスのパケットに仮想スイッチターンプール値ＴＰ０とターンプール値ＴＰ０～ｎを付加して下りＡＳパケットを生成することで、下りＡＳパケットの仮想スイッチとして機能する。

【０１１４】

ＡＳブリッジ１５は、接続されたＡＳスイッチ＃０から上りＡＳパケットを受信すると、ターンプール値の最後に付加された仮想スイッチターンプール値ＴＰ０を抽出して、この上りＡＳパケットの宛先となる論理区画を特定し、ＡＳパケットのヘッダー部を除去したものをＩ／ＯバスＩ／Ｏ１４に送信することで、上りＡＳパケットの仮想スイッチとして機能する。

【０１１５】

＜ＡＳブリッジの初期化・更新＞

次に、ＡＳブリッジ１５は、ＣＰＵブレードの起動時や、ハイパバイザ２００がパーティションを変更したとき、Ｉ／Ｏブレードの変更があったときなどに、ＢＭＣ７やハイパバイザ２００またはファブリック管理サーバ４によりメモリ１７上の表やレジスタの初期化や更新が行われる。

【０１１６】

ＡＳブリッジ１５が配置されたＣＰＵブレードが起動する際の初期化について図１２を参照しながら説明する。図１２は、ＣＰＵブレードの起動時にＢＭＣ７で行われる初期化処理のフローチャートである。

【０１１７】

ＢＭＣ７はＣＰＵブレードが起動すると、図１に示したコンソール７０でＣＰＵブレードに生成可能な論理区（パーティション）の数を設定する（Ｓ１）。なお、ＣＰＵブレードで生成可能な論理区画の最大数の設定は、コンソール７０からの入力の他に、予め設定したファイルをＢＭＣ７が読み込むことで行っても良い。

【０１１８】

次に、ＢＭＣ７はＡＳブリッジ１５のメモリ１７にアクセスを行い、メモリ１７の所定のアドレスにパーティション管理表１５５とアドレス管理表１６０を、上記Ｓ１で設定した論理区画の最大数に応じて作成する（Ｓ２）。

【０１１９】

上記Ｓ１、Ｓ２の処理が終了した時点では、ＡＳブリッジ１５の各表がメモリ１７に生成されただけであり、内容については未設定である。すなわち、上記図１２の処理では、ＣＰＵブレードに設定可能な論理区画の最大数に応じた各表の大きさがＡＳブリッジ１５のメモリ１７に確保される。

【０１２０】

次に、ハイパバイザ２００が論理区画の作成または変更（削除を含む）を行う際のＡＳブリッジ１５に対する処理について、図１３を参照しながら説明する。図１３は、ハイパバイザ２００が論理区画の設定を行う際に行う処理の一例を示すフローチャートである。

【０１２１】

まず、Ｓ１１ではハイパバイザ２００が作成（または変更）する論理区画の主記憶１３上のベースアドレスとサイズ及び論理区画番号を取得する。

【０１２２】

次に、Ｓ１２では変更を行う論理区画番号について、図９に示したアドレス管理表１６０に、ハイパバイザ２００がベースアドレスとサイズを書き込む。

【０１２３】

Ｓ１３では、ハイパバイザ２００が、アドレス管理表１６０に書き込みを行った論理区画番号について、図８のパーティション管理表１５５のパーティション番号を検索し、該当する論理区画番号に対応するパーティション管理表１５５のエントリ（仮想スイッチポート番号）について、有効／無効の欄を更新する。つまり、論理区画を作成するときにはパーティション管理表１５５の該当エントリを有効に設定し、論理区画を削除するときには、パーティション管理表１５５の該当エントリを無効に設定する。

【０１２４】

なお、アドレス管理表１６０に書き込みを行った論理区画番号が図８のパーティション管理表１５５のパーティション番号にない場合には、パーティション番号が空となっているパーティション管理表１５５の仮想ポート番号に論理区画番号を書き込み、有効／無効の欄を更新する。

【０１２５】

Ｓ１４では、ハイパバイザ２００が、パーティション管理表１５５に書き込みを行った仮想スイッチポート番号を、ファブリック管理サーバ４に通知する。ファブリック管理サーバ４では、ＡＳスイッチのトポロジに、ＡＳブリッジ１５が提供する仮想スイッチＳＷｖ１の情報を取得し、作成または更新された仮想スイッチ上の論理区画を管理することができる。

【０１２６】

さらに、新しいマルチキャストグループが定義された場合、既存のマルチキャストグループが削除された場合には、ファブリック管理サーバ４は、該当するマルチキャストグループがどのＬＰＡＲにパケットを送出する必要があるかを調べ、ＡＳブリッジ１５の持つ仮想スイッチマルチキャストテーブル１７０の、パケットを出力する必要がある論理区画

に該当するビットに１を、パケットを出力する必要の無い論理区画に該当するビットには０を書き込む。

【０１２７】

仮想スイッチでは、通常のＡＳスイッチのように、入力ポートにマルチキャスト結果を戻す処理（リフレクション）は行われないため、仮想スイッチマルチキャストテーブルには、ブリッジの入力を表すビットは設けられない。

【０１２８】

＜ＡＳブリッジのデータ書き込み、イベント通知処理＞

論理区画が要求したＩ／Ｏアクセスに対する応答は、図１４で示すように、ＡＳスイッチの機能により、要求を送信したＩ／Ｏブレードに対して上りＡＳパケット（返答パケット）が送られ、さらに、ＡＳブリッジ１５が提供する仮想スイッチによりＣＰＵブレード内の複数の論理区画に対してＩ／Ｏアクセス要求に対する上りＡＳパケットが送られる。

【０１２９】

ここで、ＡＳスイッチ側からＡＳブリッジ１５を見た場合、ファブリック管理サーバ４ではＡＳブリッジ１５を仮想スイッチＳＷｖ１として識別するので、各ＣＰＵブレードの論理区画（または物理的な区画）は、仮想スイッチＳＷｖ１の下位のノードとして見ることができる。

【０１３０】

上りＡＳパケットを受信したＡＳブリッジ１５は、パーティション管理表１５５から宛先の論理区画が有効であることを確認した後、図３で示すように、ＡＳパケットの内容（コマンド）に応じて各論理区画のメモリ空間へのＤＭＡによる読み書きや、ハイパバイザ２００を介した各論理区画に対するイベントの送付を行う。なお、パーティション管理表１５５の該当する論理区画が無効であれば、ＡＳパケットを破棄する。

【０１３１】

上りパケットはＩ／Ｏ発のＤＭＡ要求、イベント処理要求がある。Ｉ／Ｏ発のＤＭＡでは、指定された論理区画の主記憶にアクセスが行われる。イベント処理では、指定された論理区画に対して割り込み等のイベントが送られる。これに対してマルチキャスト（マルチキャストＤＭＡ、マルチキャストイベント）の場合は、該当するＩ／Ｏブレードから複数の論理区画に対して同一のＤＭＡまたはイベントを送ることができる。すなわち、マルチキャストの場合には、仮想スイッチマルチキャストテーブル１７０に予め設定された複数の論理区画に同一ＤＭＡデータが書き込まれ、同一イベントが送られる。このマルチキャストによる書き込みは、図３のように、マルチキャストＤＭＡの場合には、ＡＳブリッジ１５が主記憶１３上の各論理区画に対して複数回の書き込みを発行し、マルチキャストイベントの場合には、ハイパバイザ２００に対して通知が行われ、ハイパバイザ２００が宛先となる複数の論理区画にイベントを通知する。

【０１３２】

図１５は、上りＡＳパケットがマルチキャストＤＭＡのときにＡＳブリッジ１５が実行する処理の一例を示すフローチャートである。なお、この処理は図３におけるＤＭＡ処理部１６２と複数回書込部１６３の機能に相当する。

【０１３３】

まず、Ｓ２１では、上りＡＳパケットからＤＭＡの書き込み先アドレスＡと書き込みデータＤを抽出する。Ｓ２２では、宛先ＬＰＡＲ番号レジスタ１５８からＤＭＡの宛先となる論理区画（パーティション番号）の番号を表すビットマップを取得するとともに、パーティション管理表１５５の論理区画のうち、有効となっている論理区画番号を取得する。そして、Ｓ２３ではＡＳブリッジ１５は、宛先ＬＰＡＲ番号レジスタ１５８のビットマップが１であり（かつ有効な）論理区画の番号を一つ取得し、マルチキャストＤＭＡ書き込みを行う論理区画を決定する。この書き込み時には、上りＡＳパケットのＤＭＡアドレスに各論理区画のベースアドレスを加算し、この加算結果が各論理区画に割り当てられた主記憶１３上の領域（ベースアドレス～ベースアドレス＋サイズの範囲）を超えていないことを確認してからＤＭＡによって各論理区画のアドレスＡ毎にデータＤを書き込む。

【０１３４】
Ｓ２４では、宛先ＬＰＡＲ番号レジスタ１５８のビットが１となっている全ての論理区画に対してマルチキャストの書き込みが完了したか否かを判定する。完了していれば処理を終了し、未了であれば、Ｓ２３に戻って残りの論理区画へマルチキャストＤＭＡの書き込みを行う。
【０１３５】
以上のように、宛先ＬＰＡＲ番号レジスタ１５８のビットマップ、及び、パーティション管理表１５５に設定された論理区画毎の有効／無効の欄から、マルチキャストの宛先となっている論理区画の中から現在使用している論理区画を判定し、さらに、ＡＳブリッジ１５はマルチキャストＤＭＡを実施する論理区画を判定することができる。これにより、ＡＳブリッジ１５は、マルチキャストＤＭＡが必要な論理区画に対してのみ同一のデータＤを複数の論理区画に対して書き込むことが可能となる。また、ＬＰＡＲの有効／無効をチェックすることにより、ＬＰＡＲがＯＮ／ＯＦＦを繰り返す場合に、マルチキャストテーブルを変更する手間を削減することができる（特に全ビットが１のブロードキャストアドレスの場合に有効である）。
【０１３６】
ＤＭＡをアドレス範囲が複数の場合は、あるアドレスのデータを対象とする全ての論理区画に書いてから、次のアドレスのデータを書き込む。これにより、パケット全体を一旦記憶する必要が無く、バッファ領域に必要なメモリを削減できる他、全体が到着する前に書き込みを開始できるため、処理を高速化できる。
【０１３７】
なお、マルチキャストイベントの上りＡＳパケットの場合は、ＡＳブリッジ１５はイベントをハイパバイザ２００に伝達する。マルチキャストイベントの通知を受けたハイパバイザ２００は、宛先ＬＰＡＲ番号レジスタ１５８から対象となる複数の論理区画を求め、イベントを複数の論理区画に通知する。
【０１３８】
上記は、各論理区画宛のＤＭＡ、イベント送付の手順について述べた。それに加え、Ｉｎ－ｂｏｕｎｄ管理を行う際には、ハイパバイザに宛てたＤＭＡ、イベントもサポートする。その場合には、宛先ＬＰＡＲ番号レジスタ１５８、パーティション管理表１５５、アドレス管理表１６０はハイパバイザに該当するエントリを持つことにより、仮想スイッチがハイパバイザに対応するポート、ターンプール情報をサポートすることを可能にする。
【０１３９】
＜Ｉ／Ｏコンフィグレーション＞
次に、ファブリック管理サーバ４で行われるＩ／Ｏコンフィグレーションについて、図１６を参照しながら説明する。
【０１４０】
ＡＳスイッチを管理するファブリック管理サーバ４に接続されたコンソール５には、図１６に示すようなブレードサーバ管理画面が表示される。ファブリック管理サーバ４は、ＡＳスイッチに接続されたＩ／Ｏブレードと、ＣＰＵブレード上の論理区画の関係を管理する。すなわち、ファブリック管理サーバ４では、ＣＰＵブレード上の論理区の状態を把握して、論理区画に割り当てるＩ／Ｏブレードを設定することができる。
【０１４１】
図１６において、ＣＰＵブレード上＃０～２が示されており、各ＣＰＵブレード上には、図８のパーティション管理表１５５に設定されている論理区画の数に応じて複数のボックス４１が表示され、各ボックス４１の中には論理区画の状態を示す「State」欄が表示される。この「State」欄は、パーティション管理表１５５で有効となっている論理区画が「ON」として表示され、無効となっている論理区画が「off」として表示される。
【０１４２】
図１６では、ＣＰＵブレード＃０には論理区画はなく物理的な区画が有効となっている状態を示し、ＣＰＵブレード＃１には、２つの論理区画が生成されて一方が有効、他方が

初期化中となっている状態を示す。ＣＰＵブレード＃２には、３つの論理区画が生成され、２つの論理区画が有効で、ひとつのパーティションが無効となっている状態を示している。

【０１４３】

これらＣＰＵブレードは、３つのＩ／Ｏブレード＃０～２が接続されたＡＳスイッチに接続される。図１６においては、Ｉ／Ｏブレード＃０がＨＢＡ１という名称でコンソール５に表示され、Ｉ／Ｏブレード＃１がＨＢＡ２、Ｉ／Ｏブレード＃３がEther０という名称でそれぞれ表示される。

【０１４４】

各Ｉ／Ｏブレードには、Ｉ／Ｏ共有の状態を示す「State」欄が表示され、複数の論理区画または物理区画に接続されているＩ／Ｏブレードには「共有」と表示され、単一の論理区画または物理区画に接続されているＩ／Ｏブレードには「占有」と表示される。

【０１４５】

そして、ＡＳスイッチはひとつのボックスで表示され、Ｉ／Ｏブレードと論理区画（または物理区画）の接続状態を示す破線３０が表示される。破線３０は、現在割り当てられている論理区画（物理区画）とＩ／Ｏブレードを結ぶ線である。図１６において、Ｉ／Ｏブレード＃０（ＨＢＡ１）は、物理区画のみのＣＰＵブレード＃０と、ＣＰＵブレード＃１の論理区画０で共有されている状態を示している。

【０１４６】

論理区画（または物理区画）に対するＩ／Ｏブレードの割り当てを変更するには、論理区画のボックス４１または物理区画をカーソルでクリックする。このクリックにより、コンソール５の該当する論理区画（または物理区画）には階層型のメニュー４０が表示される。

【０１４７】

階層型メニュー４０には、Ｉ／Ｏブレードの設定の種類（割り当て（Attach）または割り当て解除（Detach））と、割り当て（解除）可能なＩ／Ｏブレードの名称が表示される。例えば、図示のように、現在初期化中のＣＰＵブレード＃１の論理区画１に新たなＩ／Ｏブレードを割り当てる場合は、「Attach」をクリックすると、割り当て可能なＩ／Ｏブレードの名称が表示されるので、この名称のいずれかひとつをクリックすると、このパーティション１に新たなＩ／Ｏブレードが割り当てられ、選択しＩ／Ｏブレードとこの論理区画１の間に破線３０が表示される。逆に割り当てを解除した場合には、破線３０が消去される。

【０１４８】

以上のように、ファブリック管理サーバ４はＡＳスイッチを介して接続されたＩ／Ｏブレードの論理区画（または物理区画）への割り当てを管理することができる。なお、ファブリック管理サーバ４には、図示しないＣＰＵ、記憶装置が設けられ、上記ＡＳスイッチを介して接続されたＩ／Ｏブレードの論理区画（または物理区画）への割り当てを管理するテーブルが、記憶装置に格納される。このテーブルは、例えば、Ｉ／Ｏブレード毎に、割り当てられているＣＰＵブレードの番号と、論理区画番号を格納するフィールドを備えていればよい。

【０１４９】

また、ファブリック管理サーバ４は、論理区画へのＩ／Ｏブレードの割り当てを変更すると、ハイパバイザ２００に変更した内容を通知する。この際、Ｉｎ－ｂｏｕｎｄで管理を行う場合には、ファブリック管理サーバ４は、上記Ｉ／Ｏブレードからのイベントや割り込みと同様にして、ハイパバイザ２００にＩ／Ｏコンフィグレーションの変更を、ＡＳファブリック経由で通知すればよい。Ｏｕｔ－ｏｆ－ｂｏｕｎｄな管理を行う場合には、Ｉ／Ｏコンフィグレーションの変更は、管理用ネットワーク（図示しない）により転送される。

【０１５０】

＜まとめ＞

以上のように、本発明によればＡＳスイッチなどのＩ／Ｏスイッチを用いてＩ／Ｏブレード（デバイス）を複数の論理区画間で共有する際に、ＣＰＵブレードに備えたＩ／Ｏブリッジを仮想スイッチとして機能させ、各論理区画毎に仮想経路情報（仮想スイッチターンプール情報）を設定することで、ひとつのＣＰＵブレードで複数の論理区画を提供する仮想計算機システムにおいてＩ／Ｏスイッチを用いたＩ／Ｏ共有を実現することができる。

【０１５１】

これにより、従来では仮想計算機間でＩ／Ｏの共有を行うためには、論理区画を識別するソフトウェアが必要になって、Ｉ／Ｏアクセスのオーバーヘッドが過大になっていたが、本発明のように、汎用のＩ／ＯバスであるＰＣＩ－ＥＸＰＲＥＳＳと、ＰＣＩ－ＥＸＰＲＥＳＳをスイッチングするＡＳを用いてハードウェアレベルで仮想計算機（論理区画）間のＩ／Ｏ共有を行うようにすることで、論理区画を識別するためのソフトウェアは不要となってＩ／Ｏアクセスの高速化を図ることができる。

【０１５２】

また、従来では仮想計算機間でＩ／Ｏの共有を行うために仮想計算機システムに固有の特別なＩ／Ｏ装置を必要としていたため、システムの価格が高価になっていたが、本発明のように、汎用のＩ／ＯバスであるＰＣＩ－ＥＸＰＲＥＳＳと、ＰＣＩ－ＥＸＰＲＥＳＳをスイッチングするＡＳを用いて仮想計算機間のＩ／Ｏ共有を行うようにすることで、システムの価格を大幅に低減できるのである。

【０１５３】

特に、複数サーバを１台にまとめるサーバコンソリデーション（サーバ統合）を実現する際には、Ｉ／Ｏブレードを複数の論理区画で共有できるため、従来ではＣＰＵブレード毎に設けていたＮＩＣ（Network Interface Card）やＦＣＡ（Fibre Channel Adaptor）等のＩ／Ｏデバイスを、必要な数のＩ／Ｏブレードとして導入すればよいので、Ｉ／Ｏデバイスを大幅に低減することで計算機システムの導入コストを低減できる。また、ＣＰＵブレードにはＩ／Ｏスロットが不要となるので、ＣＰＵブレードの製造コストも低減でき、サーバコンソリデーションを効果的に実現できるのである。

【０１５４】

なお、上記実施形態では、１つのスイッチブレード３に複数のＡＳスイッチを備えた例を示したが、１つのスイッチブレード３に１つのＡＳスイッチを備え、複数のスイッチブレード３でＡＳスイッチ群を構成しても良い。

【０１５５】

また、上記実施形態では、アドレスデコード部１５１で抽出した論理区画のアドレス情報のうち上位ビットで論理区画を探索する例を示したが、論理区画のＩＤを明示的に設定するレジスタをＡＳブリッジ１５のメモリ１７に設定するようにしてもよい。あるいは、このレジスタを論理区画の数に応じて複数持つようにしても良い。

【産業上の利用可能性】

【０１５６】

以上のように、本発明はＩ／Ｏデバイスと複数のＣＰＵの間でスイッチングを行うＩ／Ｏスイッチを仮想計算機システムに適用することができる。

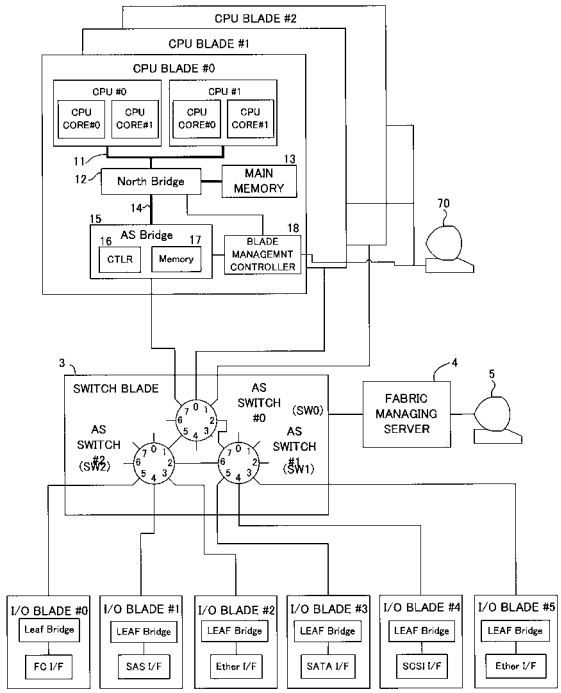
【図面の簡単な説明】

【０１５７】

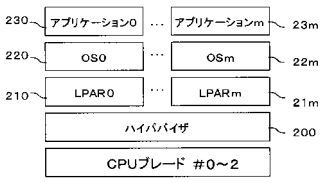
【図１】本発明を適用する計算機システムのブロック図。
【図２】本発明を適用する仮想計算機システムのソフトウェアの機能ブロック図。
【図３】ＡＳブリッジを中心とする機能ブロック図。
【図４】ＡＳスイッチの機能を説明する説明図。
【図５】下りＡＳパケットの一例を示す説明図。
【図６】上りＡＳパケットの一例を示す説明図。
【図７】ＡＳブリッジの仮想スイッチ機能を示す説明図。
【図８】パーティション管理表の一例を示す説明図。

【図９】アドレス管理表の一例を示す説明図。
【図１０】ＣＰＵブレードの主記憶の内容を示すアドレスマップ。
【図１１】宛先LPAR番号レジスタの一例を示す説明図。
【図１２】ＣＰＵブレードの初期化時のＢＭＣの制御の一例を示すフローチャート。
【図１３】論理区画作成または更新時のハイパバイザの制御の一例を示すフローチャート。
【図１４】下りＡＳパケットと上りＡＳパケットの様子を示す説明図。
【図１５】マルチキャスト時のＡＳブリッジの制御の一例を示すフローチャート。
【図１６】ファブリック管理サーバの管理画面の一例を示す説明図。
【図１７】マルチキャストパケットの一例を示す説明図。
【図１８】マルチキャストを行うスイッチの一例を示す説明図。
【図１９】仮想スイッチマルチキャストテーブルの一例を示す説明図。
【符号の説明】
【０１５８】
＃０、＃１　ＣＰＵブレード
３　スイッチブレード
４　ファブリック管理サーバ
５　コンソール
１３　主記憶
１５　ＡＳブリッジ
１８　ＢＭＣ
＃０～＃５　Ｉ／Ｏブレード

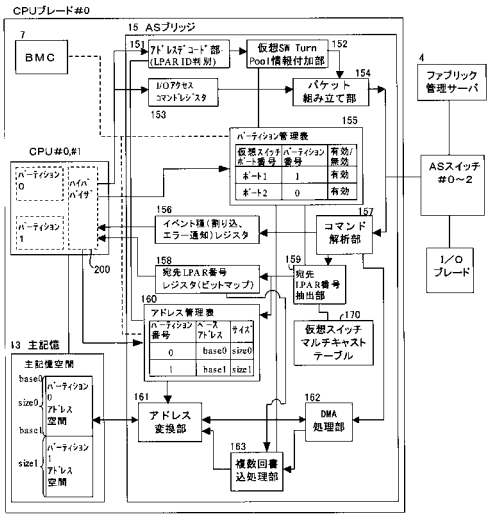
【図１】



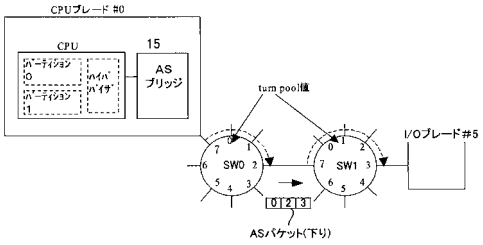
【図２】



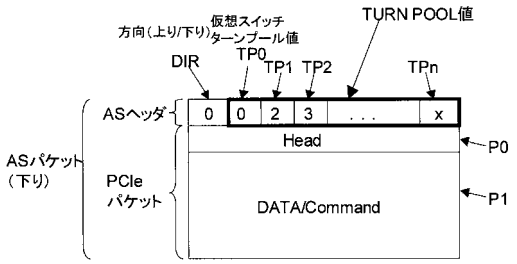
【図３】



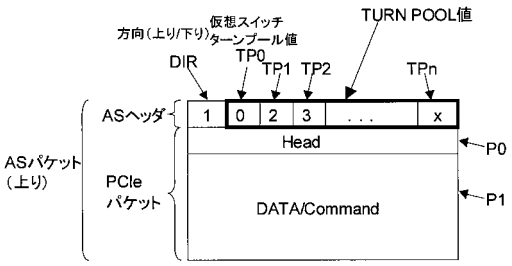
【図４】



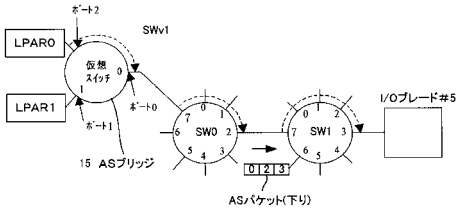
【図５】



【図６】



【図７】



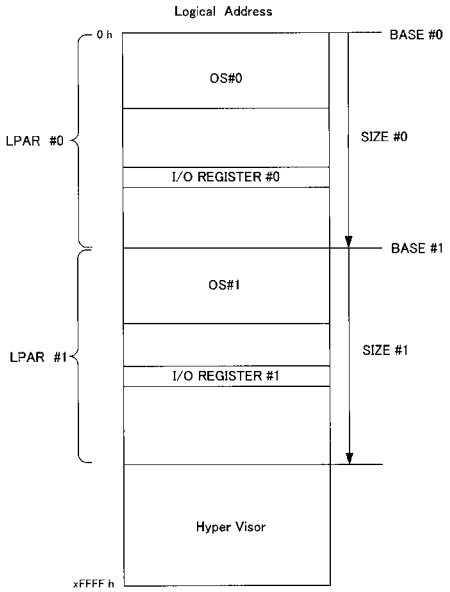
【図８】

仮想SWポートNo	パーティションNo	有効/無効
1	1	有効
2	0	有効

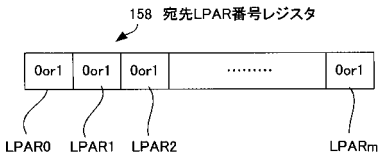
【図９】

パーティション番号	ベースアドレス	サイズ
0	Base0	Size0
1	Base1	Size1

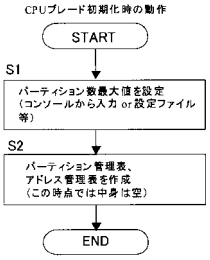
【図１０】



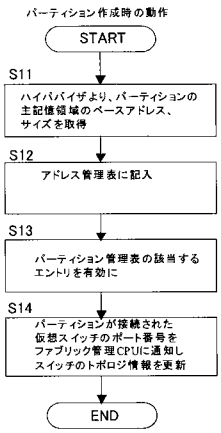
【図１１】



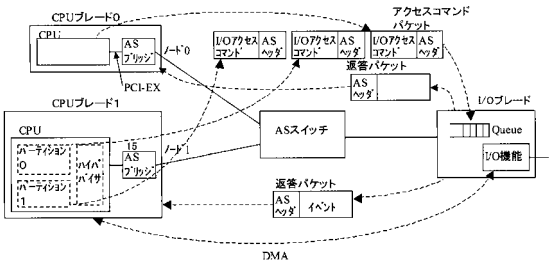
【図１２】



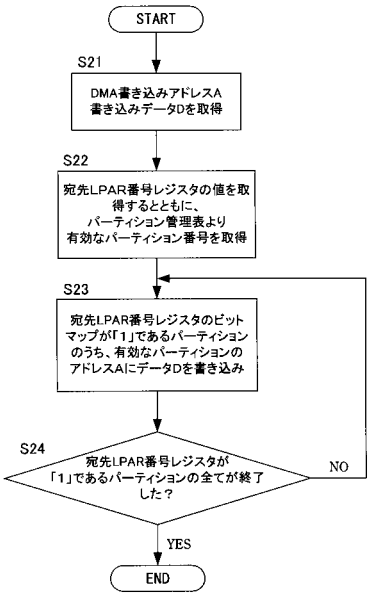
【図１３】



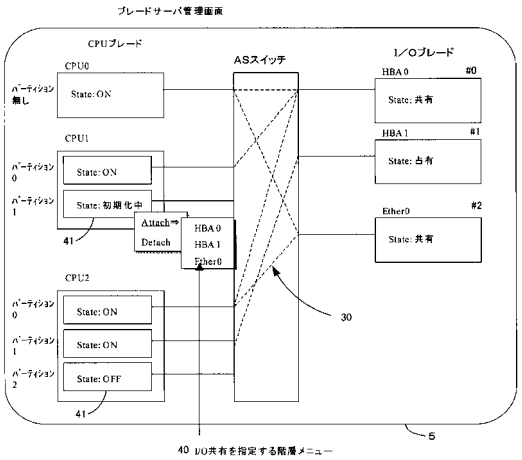
【図１４】



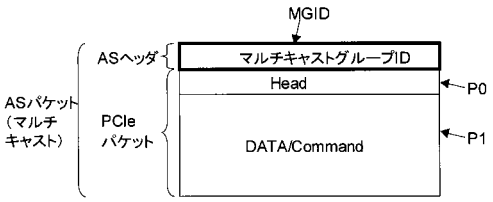
【図１５】



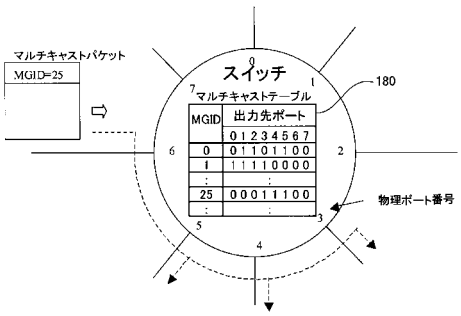
【図１６】



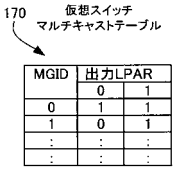
【図１７】



【図１８】



【図１９】



フロントページの続き

審査官　鈴木　修治

(56)参考文献　特開２００２－３１８７０１（ＪＰ，Ａ）
特開２００３－３１６７５２（ＪＰ，Ａ）
特開２００２－２２２１１０（ＪＰ，Ａ）
特開２００５－１２２６４０（ＪＰ，Ａ）
MAYHEW D, ＰＣＩ　Ｅｘｐｒｅｓｓ　ａｎｄ　Ａｄｖａｎｃｅｄ　Ｓｗｉｔｃｈｉｎｇ, 2003 PROCEEDINGS. 11TH SYMPOSIUM ON HIGH PERFORMANCE INTERCONNECTS, 米国, IEEE, ２００３年８月２０日, 頁21-29
WONG W, ADVANCED SWITCHING FOR PCI EXPRESS, ELECTRONIC DESIGN, 米国, PENTON MEDIA, ２００３年　6月２３日, 頁36

(58)調査した分野(Int.Cl., ＤＢ名)
Ｇ０６Ｆ　　９／４６－９／５４
Ｇ０６Ｆ　１３／１０