(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
2005 年12 月29 日 (29.12.2005)

PCT

(10) 国際公開番号
WO 2005/124548 A1

(51) 国際特許分類[7]: G06F 9/50

(21) 国際出願番号: PCT/JP2005/010986

(22) 国際出願日: 2005 年6 月9 日 (09.06.2005)

(25) 国際出願の言語: 日本語

(26) 国際公開の言語: 日本語

(30) 優先権データ:
特願2004-177460 2004 年6 月15 日 (15.06.2004) JP

(71) 出願人 (米国を除く全ての指定国について): 株式会社ソニー・コンピュータエンタテインメント (SONY COMPUTER ENTERTAINMENT INC.) [JP/JP]; 〒1070062 東京都港区南青山二丁目６番２１号 Tokyo (JP).

(72) 発明者; および

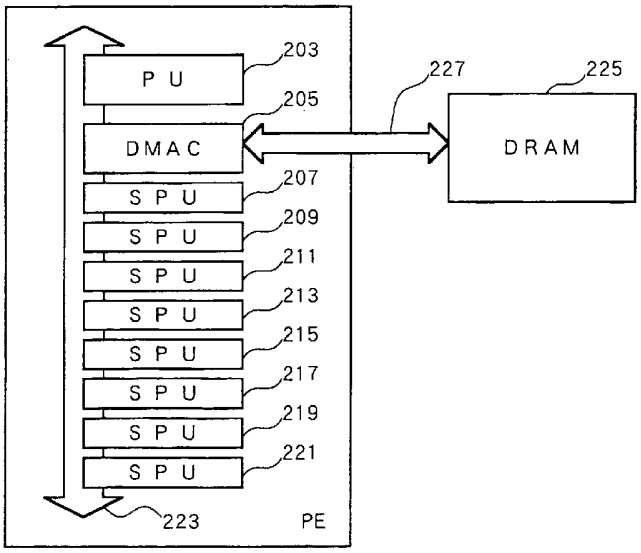
(75) 発明者/出願人 (米国についてのみ): 加藤 裕樹 (KATO, Hiroki) [JP/JP]; 〒1070062 東京都港区南青山二丁目６番２１号 株式会社ソニー・コンピュータエンタテインメント内 Tokyo (JP). 前川 博俊 (MAEGAWA, Hirotoshi) [JP/JP]; 〒1070062 東京都港区南青山二丁目６番２１号 株式会社ソニー・コンピュータエンタテインメント内 Tokyo (JP). 石田 隆行 (ISHIDA, Takayuki) [JP/JP]; 〒1070062 東京都港区南青山二丁目６番２１号 株式会社ソニー・コンピュータエンタテインメント内 Tokyo (JP).

(74) 代理人: 鈴木 正剛 (SUZUKI, Seigoh); 〒1050014 東京都港区芝三丁目２２番７号 芝ＮＫビル４階 Tokyo (JP).

[続葉有]

(54) Title: PROCESSING MANAGEMENT DEVICE, COMPUTER SYSTEM, DISTRIBUTED PROCESSING METHOD, AND COMPUTER PROGRAM

(54) 発明の名称: 処理管理装置、コンピュータ・システム、分散処理方法及びコンピュータプログラム



(57) Abstract: When performing distributed processing in processing devices connected to a network and a processing managing device for managing the processing devices, it is possible to eliminate the overhead of the processing management device. The processing management device (PU) (203) managing processing devices (SPU) (207) under its control lists up the network address of the SPU (207) and other SPU connected to the network and resource information indicating the current task execution ability of the SPU in a resource list. When one of the SPU transmits a task request to the PU (203), the PU (203) specifies one or more SPU capable of performing the task request in the resource list and requests the specified SPU to execute a task including the execution result specification destination, thereby enabling execution result transmission/reception not using the PU (203).

(57) 要約: ネットワークに接続されている複数の処理装置とこれらの処理装置を管理する処理管理装置とで分散処理を行う際の処理管理装置のオーバー

[続葉有]

WO 2005/124548 A1

(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, AT, AU, AZ, BA, BB, BG, BR, BW, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EC, EE, EG, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, KE, KG, KM, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NA, NG, NI, NO, NZ, OM, PG, PH, PL, PT, RO, RU, SC, SD, SE, SG, SK, SL, SM, SY, TJ, TM, TN, TR, TT, TZ, UA, UG, US, UZ, VC, VN, YU, 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, LS, MW, MZ, NA, SD, SL, SZ, TZ, UG, ZM, ZW), ユーラシア (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), ヨーロッパ (AT, BE, BG, CH, CY, CZ, DE, DK, EE, ES, FI, FR, GB, GR, HU, IE, IS, IT, LT, LU, MC, NL, PL, PT, RO, SE, SI, SK, TR), OAPI (BF, BJ, CF, CG, CI, CM, GA, GN, GQ, GW, ML, MR, NE, SN, TD, TG).

添付公開書類:
— 国際調査報告書

２文字コード及び他の略語については、定期発行される各*PCT*ガゼットの巻頭に掲載されている「コードと略語のガイダンスノート」を参照。

ヘッドを回避する。　傘下の複数の処理装置（ＳＰＵ）２０７等を管理する処理管理装置（ＰＵ）２０３が、ＳＰＵ２０７等と、ネットワークに接続されている他のＳＰＵに対するネットワークアドレス及び当該ＳＰＵの現在のタスク実行能力を表すリソース情報をリソースリストにリストアップしておく。いずれかのＳＰＵがＰＵ２０３に対してタスク要求を送信すると、ＰＵ２０３は、タスク要求を遂行可能な１又は複数のＳＰＵをリソースリストより特定し、特定したＳＰＵに、その実行結果の指定先を含むタスクの実行を依頼することにより、ＰＵ２０３を介在させない実行結果の受け渡しを可能にする。

明 細 書

処理管理装置、コンピュータ・システム、分散処理方法及びコンピュータプログラム

技術分野

本発明は、広帯域環境のネットワークに接続される複数のコンピュータによって効率的なタスクの実行を可能にするためのコンピュータ・システム、分散処理方法及びこのような分散処理を可能にするためのコンピュータ装置のアーキテクチャに関する。

発明の背景

今日、ネットワークに接続された複数のコンピュータ装置が協働して一つのタスクを分散処理することが一般的になっている。タスクを分散処理する場合、従来は、どのコンピュータ装置にどのタスクを割り当てるかを決めるために、予めネットワークに接続可能なすべてのコンピュータ装置の処理能力を把握しているサーバ装置の存在が不可欠となる。サーバ装置は、タスクの負荷の大きさと、分散処理を行おうとするときにネットワークに接続されている各コンピュータ装置の余剰処理能力（計算資源）とを特定し、負荷に応じた余剰処理能力を有するコンピュータ装置を逐次割り当てていき、その割り当てたコンピュータ装置からタスクの実行結果を受け取る。

サーバ装置を必要とする従来の分散処理方法では、任意の時点でネットワークに接続されたり、非接続になったりするコンピュータ装置の余剰処理能力を、サーバ装置において、迅速に把握することは、非常に困難である。また、サーバ装置が、タスクの分散処理を依頼したコンピュータ装置からその実行結果を受け取って、タスクの依頼元に転送しなければならないため、サーバ装置のオーバーヘッドが大きくなる。そのため、タスクの実行に必要な時間と、ネットワークを介するデータ伝送に必要な時間とが実質的に増加してしまうという問題がしばしば生じていた。

本発明は、このような従来の問題点を解決する分散処理の仕組みを提供することを、主たる課題とする。

発明の開示

本発明は、特徴的な処理管理装置、コンピュータ・システム、分散処理管理用のデバイス及びコンピュータプログラムによって上記の課題を解決する。

本発明の処理管理装置は、それぞれ依頼されたタスクを実行し、その実行結果を依頼元の指定先へ送信する機能を備えた複数の処理装置が接続可能なネットワークに接続される処理管理装置であって；前記ネットワークに接続されている処理装置のタスク実行能力を表すリソース情報と当該処理装置とのネットワーク通信を可能にするための通信設定情報とがリストアップされている所定のメモリへのアクセスを可能にする第１管理手段と；ネットワーク通信によりいずれかの処理装置からタスクの実行要求を受け付けたときに当該タスクを実行可能な処理装置を前記メモリにリストアップされているリソース情報より特定し、特定した処理装置についての前記通信設定情報を前記メモリより取得するとともに、前記特定した処理装置と前記タスク要求を行った処理装置の少なくとも一方の処理装置に他方の処理装置の前記通信設定情報を伝達することにより、当該処理装置間のネットワーク通信によるダイレクトな前記実行結果の受け渡しを可能にする第２管理手段と；を有するものである。

上記のメモリと、このメモリへの前記リソース情報および通信設定情報の記録を行うリソース情報等保持手段を備えて処理管理装置を構成することができる。

また、タスクは、当該タスクの実行に続く後続タスクの実行依頼とその後続タスクの実行結果の指定先の情報を含むものとすることができる。

前記第２管理手段は、例えば、前記特定した処理装置に対して前記メモリに記録されている当該処理装置の通信設定情報をもとにその実行結果の指定先を含む前記タスクの実行を依頼するように動作する。あるいは、前記第２管理手段は、例えば、前記タスク要求を行った処理装置に、前記特定した処理装置へのダイレクトなタスクの実行を依頼させるように動作する。

本発明の処理管理装置は、複数の処理装置が随時参加することができ且つ脱退

できる共有空間を前記ネットワーク上に形成する共有空間モジュールを有するものとすることができる。この場合、前記第２管理手段は、前記共有空間モジュールにより形成される共有空間に参加している処理装置の通信設定情報及び現在のリソース情報を当該処理装置より取得し、取得したこれらの情報を前記メモリにリストアップして使用可能状態にするとともに、前記共有空間から脱退した処理装置について前記メモリにリストアップされている情報を使用不能状態に変更するように動作する。使用不能状態にすることの最も単純な例は、削除することであるが、使用不能のフラグをたてるようにすることもできる。

本発明のコンピュータ・システムは、それぞれ依頼されたタスクを実行し、その実行結果を依頼元の指定先へ送信する機能を備えた処理装置と、この処理装置と内部バスを介して接続された処理管理装置とを含むものである。前記処理装置及び前記処理管理装置は、それぞれ前記内部バスを介してネットワークに接続されている。

前記処理管理装置は；前記ネットワークに接続されている処理装置のタスク実行能力を表すリソース情報と当該処理装置とのネットワーク通信を可能にするための通信設定情報とがリストアップされている所定のメモリへのアクセスを可能にする第１管理手段と；ネットワーク通信によりいずれかの処理装置からタスクの実行要求を受け付けたときに当該タスクを実行可能な処理装置を前記メモリにリストアップされているリソース情報より特定し、特定した処理装置についての前記通信設定情報を前記メモリより取得するとともに、前記特定した処理装置と前記タスク要求を行った処理装置の少なくとも一方の処理装置に他方の処理装置の前記通信設定情報を伝達することにより、当該処理装置間のネットワーク通信によるダイレクトな前記実行結果の受け渡しを可能にする第２管理手段と；を有するものである。

本発明の分散処理管理用のデバイスは、それぞれ依頼されたタスクを実行し、その実行結果を依頼元の指定先へ送信する機能を備えた複数の処理装置が接続可能なネットワークに接続されるコンピュータ・システムに搭載されるデバイスであって；所定のコンピュータプログラムを実行することにより、前記コンピュータ・システムを；前記ネットワークに接続されている処理装置のタスク実行能力

を表すリソース情報と当該処理装置とのネットワーク通信を可能にするための通信設定情報とがリストアップされている所定のメモリへのアクセスを可能にする第１管理手段；ネットワーク通信によりいずれかの処理装置からタスクの実行要求を受け付けたときに当該タスクを実行可能な処理装置を前記メモリにリストアップされているリソース情報より特定し、特定した処理装置についての前記通信設定情報を前記メモリより取得するとともに、前記特定した処理装置と前記タスク要求を行った処理装置の少なくとも一方の処理装置に他方の処理装置の前記通信設定情報を伝達することにより、当該処理装置間のネットワーク通信によるダイレクトな前記実行結果の受け渡しを可能にする第２管理手段；として動作させるものである。

本発明の分散処理方法は、それぞれ依頼されたタスクを実行し、実行結果を依頼元の指定先へ送信する機能を備えた複数の処理装置と、各処理装置との間でネットワークを通じて通信を行う処理管理装置との協働で行う分散処理方法であって、前記処理管理装置が、前記ネットワークに接続されている処理装置のタスク実行能力を表すリソース情報および当該処理装置へのアクセスを可能にするための通信設定情報を当該処理装置より取得し、取得したこれらの情報を所定のメモリにリストアップする段階と、いずれかの前記処理装置が前記処理管理装置に対してタスク要求を送信する段階と、前記タスク要求を受信した前記処理管理装置が、受信したタスク要求を遂行可能な１又は複数の前記処理装置を前記メモリの記録情報より特定し、特定した処理装置に、その実行結果の指定先を含むタスクの実行を依頼する段階とを有し、前記処理管理装置を介在させない前記実行結果の受け渡しを可能にする方法である。

本発明が提供するコンピュータプログラムは、コンピュータを、それぞれ依頼されたタスクを実行し、その実行結果を依頼元の指定先へ送信する機能を備えた複数の処理装置が接続可能なネットワークに接続される処理管理装置として動作させるためのプログラムであって；前記コンピュータを；前記ネットワークに接続されている処理装置のタスク実行能力を表すリソース情報と当該処理装置とのネットワーク通信を可能にするための通信設定情報とがリストアップされている所定のメモリへのアクセスを可能にする第１管理手段；ネットワーク通信により

いずれかの処理装置からタスクの実行要求を受け付けたときに当該タスクを実行可能な処理装置を前記メモリにリストアップされているリソース情報より特定し、特定した処理装置についての前記通信設定情報を前記メモリより取得するとともに、前記特定した処理装置と前記タスク要求を行った処理装置の少なくとも一方の処理装置に他方の処理装置の前記通信設定情報を伝達することにより、当該処理装置間のネットワーク通信によるダイレクトな前記実行結果の受け渡しを可能にする第２管理手段；として機能させるためのコンピュータプログラムである。

図面の簡単な説明

図１は、本発明が適用されるコンピュータ・システムの全体図である。

図２は、すべてのコンピュータ装置に共通となるＰＥの構造説明図である。

図３は、複数のＰＥを有するＢＥの構造説明図である。

図４は、ＳＰＵの構造説明図である。

図５は、ネットワークを伝送するパケットの構造説明図である。

図６は、パケットに含まれるＤＭＡコマンドの例を示した図であり、（ａ）はＳＰＵのローカル・メモリへのプログラム等のロードを指示するためのコマンドを含むＤＭＡコマンド、（ｂ）はプログラムをキックさせるためのキックコマンドを含むＤＭＡコマンドの例を示す。

図７は、多数のＳＰＵから構成される情報統合体の説明図である。

図８は、ＰＵに形成される機能及びテーブル等の説明図である。

図９は、クラスタテーブル及びリソースリストの利用形態の説明図である。

図１０は、リソースリスト、ＳＰＵステータステーブル及びクラスタリストの関係を示した説明図である。

図１１は、ＰＵに形成されるタスクマネージャにおける処理手順図である。

図１２は、ＰＵに形成されるリソースアロケータにおける処理手順図である。

図１３は、タスクマネージャを用いた全体的な処理概要の説明図である。

図１４は、タスクマネージャを用いた処理概要の他の説明図。

発明を実施するための最良の形態

まず、本発明が適用されるコンピュータ・システムの概要を説明する。

＜ネットワーク型コンピュータ・システムの概要＞

図１は、本発明が適用されるコンピュータ・システム１０１の全体図が示されている。このコンピュータ・システム１０１は、ネットワーク１０４を含んでいる。ネットワーク１０４の例としては、ローカル・エリア・ネットワーク（ＬＡＮ）、インターネットのようなグローバルネットワーク、あるいは他のコンピュータ・ネットワークが挙げられる。

ネットワーク１０４には、複数のコンピュータ装置の各々がそれぞれ任意のタイミングで接続して、他のコンピュータ装置と双方向の通信を行うことができる。コンピュータ装置の例としては、パーソナルコンピュータ１０６、サーバコンピュータ１０８、通信機能付のゲームコンソール１１０、ＰＤＡ１１２及びその他の有線または無線コンピュータとコンピューティング・デバイスなどが含まれる。

各コンピュータ装置は、それぞれ、共通の構造を持つプロセッサ・エレメント（以下、「ＰＥ」）を有している。これらのＰＥは、命令セット・アーキテクチャ（ＩＳＡ）がすべて同じで、同じ命令セットに従って所要の処理を実行できるものである。個々のコンピュータ装置に含まれるＰＥの数は、そのコンピュータ装置がタスクを実行する上で必要な処理能力によって決められる。

コンピュータ装置のＰＥがこのように均質な構成を有することから、コンピュータ・システム１０１におけるアダプタビリティを改善することができる。また、各コンピュータ装置が、各々のＰＥのうち１つまたはそれ以上、またはＰＥの一部を用いて、他から依頼されたタスクを実行できるようにすることにより、タスクをどのコンピュータ装置で実行するかはさほど重要ではなくなる。依頼されたタスクの実行結果を、タスクを依頼したコンピュータ装置あるいは後続タスクを実行するコンピュータ装置を指定先として伝達するだけで足りる。そのため、個々のタスクは、ネットワーク１０４に接続されている複数のコンピュータ装置の間で分散実行することが容易になる。

各コンピュータ装置が、共通の構造のＰＥと共通のＩＳＡとを有するので、コンピュータ装置間の互換性を達成するためのソフトウェアの追加層の計算上の負担が回避される。また、異質なネットワークの混在という問題の多くを防ぐこと

ができる。したがって、このシステム１０１は、広帯域処理の実現が可能となる。

＜コンピュータ装置のアーキテクチャ＞

次に、コンピュータ装置のアーキテクチャを明らかにする。まず、各コンピュータ装置が有するＰＥの構造例を図２を参照して説明する。

図２に示されるように、ＰＥ２０１は、プロセッサ・ユニット(ＰＵ)２０３、ダイレクト・メモリ・アクセス・コントローラ（ＤＭＡＣ）２０５、ＰＵとは異なる複数のプロセッサ・ユニット(ＳＰＵ)、すなわち、ＳＰＵ２０７、ＳＰＵ２０９、ＳＰＵ２１１、ＳＰＵ２１３、ＳＰＵ２１５、ＳＰＵ２１７、ＳＰＵ２１９、ＳＰＵ２２１を含んでいる。ＰＥバス２２３は、ＳＰＵ、ＤＭＡＣ２０５、ＰＵ２０３を相互に接続する。このＰＥバス２２３は、従来型のアーキテクチャなどを備えていても良いし、あるいは、パケット交換式ネットワークとして実現されても良い。パケット交換式ネットワークとして実現される場合、より多くのハードウェアが必要となり、その一方で、利用可能な帯域幅が増加する。

ＰＥ２０１は、デジタル論理回路を実現する様々な方法を用いて構成可能であるが、単一の集積回路として構成されることが望ましい。ＰＥ２０１は、高帯域メモリ接続部２２７を介して、ダイナミック・ランダム・アクセス・メモリ(ＤＲＡＭ)２２５に接続される。ＤＲＡＭ２２５は、ＰＥ２０１のメイン・メモリとして機能する。なお、メインメモリは、ＤＲＡＭ２２５であることが望ましいが、他の種類のメモリ、例えばスタティック・ランダム・アクセス・メモリ(ＳＲＡＭ)としての磁気ランダム・アクセス・メモリ(ＭＲＡＭ)、光メモリまたはホログラフィ・メモリなどをメインメモリとして用いることもできる。ＤＭＡＣ２０５は、ＰＵ２０３及びＳＰＵ２０７等によるＤＲＡＭ２２５へのダイレクト・メモリ・アクセス（ＤＭＡ）を実現する。ＤＭＡの実現手法の例としては、クロスバー・スイッチをその前段に設けることなどが挙げられる。

ＰＵ２０３は、ＤＭＡＣ２０５に対してＤＭＡコマンドを出すことによりＳＰＵ２０７等の制御を行う。その際、ＰＵは、独立したプロセッサとしてＳＰＵ２０７等を扱う。したがって、ＳＰＵ２０７等による処理を制御するために、ＰＵは、遠隔手順呼出しに類似したコマンドを使用する。これらのコマンドは“遠隔手順呼出し(ＲＰＣ)”と呼ばれる。ＰＵ２０３は、一連のＤＭＡコマンドをＤＭ

ＡＣ２０５へ出すことによりＲＰＣを実行する。ＤＭＡＣ２０５は、依頼されたタスクの実行に対してＳＰＵが必要とするプログラム（以下、「ＳＰＵプログラム」）とそれに関連するデータとをＳＰＵのローカル・ストレージへロードする。ＰＵ２０３は、その後、ＳＰＵへ最初のキック・コマンドを出し、ＳＰＵプログラムを実行させる。

ＰＵ２０３は、必ずしもネットワーク型のプロセッサである必要はなく、スタンド・アローン型の処理が可能な標準的なプロセッサを主たるエレメントとして含むものであって良い。このプロセッサがＤＲＡＭ２２５あるいは図示しない読み出し専用メモリに記録されている本発明のコンピュータプログラムを読み込んで実行することにより、通信モジュールを形成するとともに、自己が管理可能なＳＰＵ２０７，２０９，２１１，２１３，２１５，２１７，２１９，２２１、他のＰＥあるいは後述するＢＥのＰＥが管理するＳＰＵの管理を行うための各種機能を形成する。これらの機能については、後で詳しく述べる。

図２に示したＰＥ２０１のようないくつかのＰＥを結合して処理能力を向上させることもできる。例えば、図３に示されるように、１以上のチップ・パッケージなどの中に複数のＰＥを結合してパッケージ化し、これにより単一のコンピュータ装置を形成しても良い。このような構成のコンピュータ装置は、広帯域エンジン（ＢＥ）と呼ばれる。

図３の例では、ＢＥ３０１には、４つのＰＥ（ＰＥ３０３、ＰＥ３０５、ＰＥ３０７、ＰＥ３０９）が含まれる。これらのＰＥ間の通信は、ＢＥバス３１１を介して行われる。また、これらのＰＥと共用ＤＲＡＭ３１５とが、広帯域メモリ接続部３１３によって接続されている。ＢＥバス３１１の代わりに、ＢＥ３０１のＰＥ間の通信は、共用ＤＲＡＭ３１５とこの広帯域メモリ接続部３１３とを介して行うことができる。ＢＥ３０１において、１つのＰＥが有する複数のＳＰＵのすべては、それぞれ独立に、但し、一定の制限下で、共用ＤＲＡＭ３１５内のデータへアクセスすることができる。

入力／出力インターフェース（Ｉ／Ｏ）３１７と外部バス３１９とは、ＢＥ３０１と、ネットワーク１０４に接続されている他のコンピュータ装置との間の通信インタフェースとなるものである。Ｉ／Ｏ３１７は、例えばプロセッサ等の能

動素子を含んで構成され、ネットワーク１０４とＢＥ３０１内の各ＰＥ３０３，３０５，３０７，３０９との間の通信を制御するほか、ネットワーク１０４からの種々の割込をも受け付け、これを該当するＰＥに伝達する。そこで、以後の説明では、これらを総称して「ネットワークインタフェース」と称する場合がある。このようなネットワークインタフェースは、必ずしもＢＥ３０１に内蔵されていなくても良く、ネットワーク１０４上に設けられていても良い。

なお、図２及び図３では図示していないが、画像処理を行うためのピクセル・エンジン、画像用キャッシュ、ディスプレイへの表示コントローラ等を含んで、ＰＥ２０１あるいはＢＥ３０１を構成することもできる。このような付加的な機能は、後述するリソース管理において重要となる。

ＰＥ又はＢＥに含まれるＳＰＵは、単一命令、複数データ（ＳＩＭＤ）プロセッサであることが望ましい。ＰＵ２０３の制御によって、ＳＰＵは、並列的かつ独立に、ＰＥを通じて依頼されたタスクを実行し、その実行結果を依頼元が指定するＳＰＵに向けて出力する機能を実現する。図２の例では、１つのＰＥ２０１が８個のＳＰＵを有するが、ＳＰＵの数は、必要とする処理能力に応じて任意であって良い。

次に、ＳＰＵの構造を詳しく説明する。ＳＰＵは、図４に例示される構造を有する。ＳＰＵ４０２には、ＳＰＵ内で処理されるデータ及びＳＰＵ内で実行される各種スレッドなどを格納するためのローカル・ストレージとなるローカル・メモリ４０６、レジスタ４１０、複数の浮動小数点演算ユニット４１２及び複数の整数演算ユニット４１４が含まれる。浮動小数点演算ユニット４１２と整数演算ユニット４１４の数は、予め必要となる処理能力に応じて、任意に決めることができる。

ローカル・メモリ４０６は、好適にはＳＲＡＭとして構成される。ＳＰＵ４０２には、さらに、各種スレッドの実行結果、タスク依頼あるいはタスクの実行結果の受け渡しを行うためのバス４０４が含まれる。ＳＰＵ４０２にはさらに内部バス４０８、４２０、４１８が含まれる。内部バス４０８は、ローカル・メモリ４０６とレジスタ４１０とを接続する。内部バス４２０、４１８は、それぞれ、レジスタ４１０と浮動小数点演算ユニット４１２との間、及び、レジスタ４１０

と整数演算ユニット４１４と間を接続する。ある好ましい実施の態様では、レジスタ４１０から浮動小数点演算ユニット４１２または整数演算ユニット４１４へのバス４１８と４２０の幅は、浮動小数点演算ユニット４１２または整数演算ユニット４１４からレジスタ４１０へのバス４１８と４２０の幅よりも広い。レジスタ４１０から浮動小数点演算ユニットまたは整数演算ユニットへの上記バスの広い幅によって、レジスタ４１０からのより広いデータ・フローが可能になる。

なお、図示を省略したが、コンピュータ装置は、絶対タイマを使用する。絶対タイマはＳＰＵとＰＥの他のすべてのエレメントへクロック信号を出力する。このクロック信号はこれらのエレメントを駆動するクロック信号に依存せず、かつ、このクロック信号より高速である。この絶対タイマによって、ＳＰＵによるタスク・パフォーマンスのためのタイム・バジェット（割り当て時間）が決定される。このタイム・バジェットによって、これらのタスクの完了時間が設定されるが、この時間は、ＳＰＵによるタスク処理に必要な時間よりも長い時間になる。その結果、個々のタスクについて、タイム・バジェットの範囲内に、ビジーな時間とスタンバイ時間とが存在することになる。ＳＰＵプログラムは、ＳＰＵの実際の処理時間にかかわらず、このタイム・バジェットに基づいて処理を行うように作成される。

＜通信データ＞

以上のように構成されるコンピュータ装置は、ネットワーク１０４に接続されている任意のタイミングで、ネットワーク１０４上の他のコンピュータ装置に対してタスクを依頼したり、あるいは、依頼されて自己が行ったタスクの実行結果等を伝達するための通信データを生成する。この通信データには、タスクの依頼内容、タスクを実行するための処理プログラム、データ、実行結果あるいは後続タスクの内容を伝達する指定先の通信設定情報など、様々な種類の情報が含まれる。この実施形態では、これらの様々な種類の情報を所定のデータ構造のもとでパケット化し、このパケットをコンピュータ装置相互で受け渡しするようにする。

パケットの受け渡しは、通常は、ＤＲＡＭを介して行われる。例えば、コンピュータ装置が図２のようなＰＥを一つだけ有する場合はＤＲＡＭ２２５、コンピュータ装置が図３のようなＢＥを有する場合は共用ＤＲＡＭ３１５から、それぞ

れ、図４に示した各ＳＰＵのローカル・メモリ４０６に読み出され、そのＳＰＵによって直接処理される。

なお、上記の場合、ＳＰＵのローカル・メモリ４０６に、プログラム・カウンタと、スタックと、処理プログラムを実行するために必要な他のソフトウェア・エレメントとが含まれるようにしても良い。

パケットの受け渡しに際しては、常にＤＲＡＭを介さなければならないというものでもない。すなわち、コンピュータ装置が図２のようなＰＥを一つだけ有する場合はＳＰＵのローカル・メモリ４０６に直接ロードされるようにし、他方、コンピュータ装置が図３のようなＢＥを有する場合は、Ｉ／Ｏ３１７から各ＳＰＵのローカル・メモリ４０６に直接ロードされるようにして、それぞれのＳＰＵによって処理されるようにしても良い。

図５は、パケットの構造例を示している。パケット２３０２の中には、ルート選定情報セクション２３０４と本体部分２３０６とが含まれる。ルート選定情報セクション２３０４に含まれる情報は、ネットワーク１０４のプロトコルに依って決められる。ルート選定情報セクション２３０４の中には、ヘッダ２３０８、宛先ＩＤ２３１０、ソースＩＤ２３１２及び応答ＩＤ２３１４が含まれる。宛先ＩＤ２３１０には、タスクの実行結果又は後続タスクを実行させるための当該パケットの伝送先となるＳＰＵの識別子が含まれる。

ＳＰＵの識別子は、それが図２に示される一つのＰＥに所属する場合にはＰＥ及びＳＰＵのセットから成るネットワーク・アドレス、それが図３に示されるＢＥの一つのＰＥに所属する場合にはＢＥ、ＰＥ及びＳＰＵのセットから成るネットワーク・アドレスが含まれる。ＴＣＰ／ＩＰプロトコルの下でのネットワーク・アドレスはインターネット・プロトコル(ＩＰ)アドレスである。ソースＩＤ２３１２には、当該パケットの伝送元となるＳＰＵの識別子が含まれる。このソースＩＤ２３１４により識別されるＳＰＵとからパケットが生成され、ネットワーク１０４に向けて発信される。応答ＩＤ２３１４には、当該パケットに関するクエリとタスク実行結果又は後続タスクを伝送する指定先のＳＰＵの識別子が含まれる。

パケットの本体部分２３０６には、ネットワークのプロトコルとは無関係の情

報が含まれる。図５の破線で示した分解部分は、本体部分２３０６の細部を示している。本体部分２３０６のヘッダ２３１８によってパケット本体の開始領域が識別される。パケットインターフェース２３２２には、当該パケットを利用する上で必要となる各種情報が含まれる。これらの情報の中には、グローバルな一意的ＩＤ２３２４と、タスク実行のために指定されるＳＰＵ２３２６と、専有メモリサイズ２３２８と、前回処理されたＳＰＵで付されたパケットＩＤ２３３０とが含まれる。ＳＰＵは、依頼されたタスクを実行した際に、このパケットＩＤ２３３０を更新してネットワーク１０４に発信することになる。グローバルな一意的ＩＤ２３２４は、ネットワーク１０４全体を通じて当該パケット２３０２を一意的に識別するための情報である。このグローバルな一意的ＩＤ２３２４は、ソースＩＤ２３１２(ソースＩＤ２３１２内のＳＰＵの一意的識別子など)と、パケット２３０２の作成または伝送の時刻と日付とに基づいて作成される。

専有メモリサイズ２３２８によって、タスクの実行に必要なＤＲＡＭと関連する必要なＳＰＵ内に、他の処理から保護されたメモリサイズが確保される。なお、専有メモリサイズ２３２８は、必要なデータ構造としては不可欠なものではなく、予めメモリサイズが指定されている場合には、不要となる。前回のパケットＩＤ２３３０によって、シーケンシャルな実行を要求する１グループのタスクにおける前回のタスクを実行したＳＰＵの識別子を事後的に把握可能になる。

実行セクション２３３２の中には、当該パケットのコア情報が含まれる。このコア情報の中には、ＤＭＡコマンド・リスト２３３４と、タスクの実行に必要な処理プログラム２３３６と、データ２３３８とが含まれる。処理プログラム２３３６には、ＳＰＵプログラム２３６０、２３６２などの、このパケット２３０２を受け取ったＳＰＵによって実行されるプログラムが含まれ、データ２３３８にはこれらのＳＰＵプログラムにより処理される各種データが含まれる。

なお、タスク依頼先のＳＰＵにタスク実行に必要な処理プログラムが存在する場合には、処理プログラム２３３６は不要となる。

ＤＭＡコマンド・リスト２３３４には、プログラムの起動に必要な一連のＤＭＡコマンドが含まれる。これらのＤＭＡコマンドにはＤＭＡコマンド２３４０、２３５０、２３５５、２３５８が含まれる。

これらのＤＭＡコマンド２３４０、２３５０、２３５５、２３５８は、例えば、図６に示されるようなものである。

すなわち、図６（ａ）に示されるように、ＤＭＡコマンド２３４０には、ＶＩＤコマンド２３４２が含まれる。ＶＩＤコマンド２３４２は、ＤＭＡコマンドが出されたときに、ＤＲＡＭ２２５の物理ＩＤに対して対応づけられるＳＰＵのバーチャルＩＤである。ＤＭＡコマンド２３４０にはロード・コマンド２３４４とアドレス２３４６も含まれる。ロード・コマンド２３４４は、ＳＰＵにＤＲＡＭ２２５から特定の情報を読み出してローカル・メモリ４０６へ記録するように命令するためのものである。アドレス２３４６によってこの特定情報を含むＤＲＡＭ２２５内のバーチャル・アドレスが与えられる。この特定情報は、処理プログラム２３３６、データ２３３８、あるいはその他のデータなどであって良い。ＤＭＡコマンド２３４０には、また、ローカル・メモリ４０６のアドレス２３４８が含まれる。このアドレスによって、パケットに記録されているすべての情報をロードできそうなローカル・メモリ４０６のアドレスが識別される。ＤＭＡコマンド２３５０についても同様となる。

図６（ｂ）に例示されるＤＭＡコマンド２３５５は、キック・コマンドを含むものである。「キック・コマンド」とは、上述したように、ＰＵから指定先となるＳＰＵに出される当該パケット内のＳＰＵプログラムによる実行を開始させるためのコマンドである。このＤＭＡコマンド２３５５には、ＶＩＤコマンド２３５２と、上述したキック・コマンド２３５４と、プログラムカウンタ２３５６とが含まれる。ここにいうＶＩＤコマンド２３５２は、キックすべき対象ＳＰＵを識別するためのものであり、キック・コマンド２３５４は関連するキック・コマンドを与えるものであり、プログラムカウンタ２３５６は、ＳＰＵプログラムの実行用プログラムカウンタのためのアドレスを与えるものである。ＤＭＡコマンド２３５８についても同様の内容となる。

＜運用形態＞

次に、図１に示したコンピュータ・システム１０１の運用形態の一例を説明する。

上述したように、ネットワーク１０４に接続される各コンピュータ装置は、共

通構造のＰＥと共通のＩＳＡとを有する。そのため、ネットワーク１０４に接続されているコンピュータ装置間のアーキテクチャの相違は吸収され、ネットワーク１０４には、図７に示されるように、各ＰＥに含まれる多数のＳＰＵの各々があたかも情報処理の細胞（Cell）のように機能する大規模情報処理統合体ＷＯが形成される。

大規模情報処理統合体ＷＯにおける個々のＳＰＵ５１２は、物理的には自己が所属するＰＵ５０１，５０２，５０３，５０４，５０ｎ，５０ｍによって管理され、単独のＳＰＵとして動作したり、そのＰＵ内の他のＳＰＵとグループ化されて協働で動作したりするが、論理的には、ＰＵによる壁はなく、異なるＰＵのもとで管理される他のＳＰＵとの間でグループ化が可能である。このような形態でグループ化される場合、一つのタスクをグループに属する複数のＳＰＵで分散処理することにより実行することができる。

分散処理の一形態として、グループ間で共通にアクセス可能な共有空間を構築し、あるコンピュータ装置を操作するユーザが、そのコンピュータ装置のいくつかのＳＰＵのリソースを通じて、共有空間にアクセスすることができる。図７において、Ｇ０１～Ｇ０４は、それぞれ、大規模情報処理統合体ＷＯにおける複数のＳＰＵのグループを示している。

このような大規模情報処理統合体ＷＯにおける効率的な分散処理を可能にするため、ＰＵには、種々の機能が形成される。図８は、本発明のプログラムとの協働によりＰＵに形成される機能及び各種リスト並びにテーブルの説明図である。

ＰＵには、タスクマネージャ６０１と、リソースアロケータ６０３と、クラスタマネージャ６０５と、スケジューラ６０７と、共有空間モジュールの一例となる共有空間マネージャ６０９と、通信モジュール６１１とが形成される。通信モジュール６１１は、内部のＳＰＵとの間でＰＥバス２３３（図２）を介して双方向通信を行うとともに、ＰＥバス２３３及びネットワーク１０４を介して他のＰＵ（ＰＥ／ＢＥ）のＳＰＵとの間での双方向通信の手順を制御する。通信モジュール６１１は、ルーティングデータ及びプロトコルデータを記録したテーブル７１１を参照することにより、上記制御を行う。

クラスタマネージャ６０５は、通信モジュール６１１及びタスクマネージャ６

０１を介して通信可能なすべてのＳＰＵをクラスタ化するための処理を行う。この処理の内容を具体的に説明すると、まず、各ＳＰＵが処理可能なタスクの種類と現在の処理能力値（処理能力を数値化したデータ）を、例えばＳＰＵにおいて実行されるモニタリング・スレッドと、タスクマネージャ６０１により実行されるモニタリング・スレッドとを通じて取得する。そして、各ＳＰＵにおいて実行可能なタスクの種類と各々のＳＰＵの処理能力値の組み合わせを、その実行が想定されるタスクの種類又はサイズ毎にクラスタ化してクラスタリスト７０５にリストアップする。

クラスタリスト７０５の内容は、例えば図９の上段に示されたものとなる。クラスタの最低単位は、１つのＳＰＵ７０５０となる。図９の例では、それぞれ１つの特定のＳＰＵからなる複数種類のＳＰＵクラスタ７０５１と、複数のＳＰＵの組み合わせからなるＳＰＵクラスタ７０５３、７０５５、７０５７、７０５９、７０６１、７０６３がリスト化されている。それぞれ、１クラスタ毎に、ＳＰＵを識別するためのＳＰＵ＿ｉｄとそのＳＰＵが所属するＰＵの識別子（ＰＵ－ｒｅｆ）と、割当の可否を表す識別情報（valid／invalid）とが記録される。「valid」は割当可、「invalid」は割当不可を表す。

図１０の上段を参照すると、４つのＳＰＵの組み合わせからなるＳＰＵクラスタの具体例が示されている。ＳＰＵクラスタは、ＳＰＵの数（size）、タスクサイズ（length）に応じて予め定められているテーブル７０３１に基づいて作成される。図１０上段の例では、ＳＰＵが４つなので、テーブル７０３１の「SIZE」が「４」のデータに基づいてＳＰＵクラスタが生成されている。クラスタリスト７０５の記録内容は、随時、更新される。テーブル７０３１におけるＳＰＵの数（size）は、任意の数値ではなく、全体のクラスタ数に対する割合を表す関数等で表現されていても良い。

ＳＰＵステータステーブル７０１は、例えば図１０後段に示される内容のものである。すなわち、あるクラスタの「ＰＵ－ｒｅｆ」によって識別されるＰＵ３３０２におけるＳＰＵ＿ｉｄ毎にそのステータス情報（status）がＳＰＵステータステーブル７０１１に記録される。ステータス情報は、そのＳＰＵが現在クラスタ化されているかどうか、稼働状態にあるかなどを表す情報である。「busy」は

稼働状態であり「none」はそれ故にバックポインタが不要であることが示されている。

タスクマネージャ６０１は、図１１の手順で処理を行う。すなわち、ＳＰＵからタスク依頼（タスク要求）があったときに（ＴＳ１００１：Ｙｅｓ）、依頼されたタスクを実行可能なＳＰＵクラスタの割当をリソースアロケータ６０３に依頼する（ＴＳ１００２）。リソースアロケータ６０３から、割り当てられたＳＰＵクラスタに属するＳＰＵの情報が通知されると（ＴＳ１００３：Ｙｅｓ）、割り当てられたすべてのＳＰＵのＳＰＵ＿ｉｄをスケジューラ６０７に通知して、そのＳＰＵをリザーブさせる（ＴＳ１００４）。これは、ＳＰＵの動作を他の処理用と競合させないようにするためである。

タスクに記述された１個以上の処理プログラム間の呼び出し関数をつなぐデータパスと割当されたＳＰＵの位置情報から生成された関数テーブル７０７を参照して、ＳＰＵに実行させるタスクの種類に応じた実行結果の指定先となるデータパスを検索する（ＴＳ１００５）。そして、最初のタスクを実行するＳＰＵに向けて、当該タスクの依頼内容とデータパスとを送信する（ＴＳ１００６）。ここにいうタスクの記述例としては、以下のようなものがある。

〈映像処理〉

〈program=”ビデオ用プログラム” name=”video” /〉

〈program=”オーディオ用プログラム” name=”audio” /〉

〈program=”パケット化プログラム” name=”packet” /〉

〈sub from=”video.Video_OUT” to=”packet.Video_IN” /〉

〈sub from=”auido.AUdio_OUT” to=”packet.Autdio_IN” /〉

〈data file=”video_data” to=”video.Data_IN” /〉

〈data file=”audio_data” to=”audio.DATA_IN” /〉

〈data file=”packet_data” from=”packet.DATA_OUT” /〉

〈/映像処理〉

この際のタスクの依頼内容とデータパスの送信は、ＰＵ、すなわちタスクマネージャ６０１からダイレクトに行ってもよく、データパス等をタスク要求を行ったＳＰＵのローカル・メモリに記録して、そのＳＰＵにてパケットを作成し、こ

のパケットをＰＥバスを介してネットワークに送信するようにしても良い。

また、変形例として、割り当てられたＳＰＵのＳＰＵ＿ｉｄより特定されるＳＰＵに向けて、タスク要求を行ったＳＰＵのデータパスとタスク依頼の内容とを伝達し、当該特定されるＳＰＵからタスク要求を行ったＳＰＵにダイレクトに通信路を確立するようにしても良い。

タスクマネージャ６０１は、また、通信モジュール６１１を通じて、自己に属するＳＰＵについてのリソース情報、ステータス情報、スケジュール情報を他のＢＥ、ＰＥあるいはＰＥに伝達するとともに、他のＢＥ、ＰＥあるいはＰＥに所属するＳＰＵについてのこれらの情報を取得する。取得したステータス情報はリソースアロケータ６０３を介して、あるいは、タスクマネージャがダイレクトに、ＳＰＵステータステーブル７０１に記録する。リソース情報も同様にしてリソースリスト７０３に記録する。

リソース情報は、個々のＳＰＵが現在どのような種類の処理を行うことができ、且つ、その処理能力値がどの程度かを表す情報である。リソースリスト７０３には、このリソース情報が、ＳＰＵ＿ｉｄ毎に記録される。上述したように、ＳＰＵは、すべてのコンピュータ装置において共通構造であるが、ＳＰＵが所属するＰＥに付加されている処理エンジンの有無、及び、そのような処理エンジンの種類により、ＳＰＵが可能となる処理の種類が異なる場合がある。ＳＰＵプログラムさえロードすれば、どのような種類の処理あっても可能な場合、処理の種類は、図９に示されるように「all」となる。「all」以外の特定の処理の種類がある場合は、画像処理のみ、音声処理のみ、パッケージングのみ・・・のように記述される。処理能力値がすべてのＳＰＵにおいて共通で、その値が既知の場合には、処理能力値については、リストアップを省略することができる。

スケジュール情報は、上述したように、ＳＰＵをリザーブするための情報である。スケジューラ６０７は、このスケジュール情報をＳＰＵ毎に図示しないタイムテーブル等に記録し、必要に応じてタスクマネージャ６０１に記録情報を通知する。

リソースアロケータ６０３は、図１２に示した手順で処理を行う。すなわち、タスクマネージャ６０１を通じて割当依頼があると（ＡＬ１００１：Ｙｅｓ）、リ

ソースアロケータ６０３は、タスクサイズ及びタスクの種類を解析する（ＡＬ１００２）。そして、クラスタリスト７０５にリストアップされているＳＰＵクラスタから、タスクの実行に最適なＳＰＵクラスタを検索する（ＡＬ１００３）。

最適なＳＰＵクラスタの検索基準としては、例えば、以下のようなものがある。

（１）検索の引数がタスクサイズのみとなる場合、すなわち、処理能力値がすべてのＳＰＵで共通の場合、ごく簡易な例としては、タスクサイズが一致するクラスタリストを選択し、クラスタリスト内で識別情報がvalidである先頭のクラスタを割り当てる。ネットワーク距離に基づいて最適かどうかを判定することもできる。この場合は、クラスタを構成するＳＰＵ間のすべての組み合わせについて、ネットワーク距離（相手のＳＰＵに到達するまでに経由しなければならないホストの数）を算出し、ネットワーク距離の最大値が小さい順にクラスタリストを作成し、クラスタリスト内で識別情報がvalidである先頭のクラスタを割り当てる。

（２）検索の引数がタスクサイズとタスクの種類になる場合は、例えば、タスクサイズに基づいて、クラスタリストを選択し、次いで、識別情報がvalidである各クラスタについてクラスタリストの先頭から、必要なタスク処理を備えたＳＰＵが存在するかを順にチェックする。そして、最初に必要なタスク処理を実行できるＳＰＵ群を有するクラスタを返す。

以上のようにして最適なＳＰＵクラスタを特定すると、リソースアロケータ６０３は、ＳＰＵステータステーブル７０１を参照して、特定したＳＰＵクラスタに属するＳＰＵのステータスを確認する（ＡＬ１００４）。ステータスが「busy」等のために割当可能でなかった場合は（ＡＬ１００５：Ｎｏ）、クラスタリスト７０５から他のＳＰＵクラスタを特定し（ＡＬ１００６）、ＡＬ１００４の処理に戻る。割当可能であった場合は（ＡＬ１００５：Ｙｅｓ）、割当可能なすべてのＳＰＵをタスクマネージャ６０１に通知する（ＡＬ１００７）。

共有空間マネージャ６０９は、共有空間管理データをもとに、複数のＳＰＵが任意に参加し、任意に脱退することができる共有空間を、図７に示した大規模情報処理統合体ＷＯの一部の領域に形成する。共有空間管理データは、例えば、複数のＳＰＵが同時期にアクセス可能な領域のアドレス等である。共有空間を形成したときは、上記のステータス情報、リソース情報、スケジュール情報を共有す

る必要がある。そこで、共有空間マネージャ６０９は、ＳＰＵステータステーブル７０１及びリソースリスト７０３を、参加しているＳＰＵを管理しているＰＵに伝達し、記録内容の更新時期を同期させるようにする。クラスタリスト７０５をもＰＵ間で相互に受け渡すようにしても良い。

ＰＵは、共有空間に参加しているＳＰＵの識別情報、各ＳＰＵの通信設定情報、各ＳＰＵの現在のリソース情報等を所定のメモリ領域にＳＰＵの識別情報毎にリストアップして使用可能状態（valid）とする。他方、共有空間から脱退したＳＰＵについては、リストアップされている情報を使用不能状態（invalid）に変更する。

図１３は、上記の全体的な処理の例を示した図である。この図において、網掛部分は、クラスタ化されたＳＰＵを表している。リソースアロケータ５００７は、本来、各ＰＵに備えられるものであるが、複数のＰＵに跨る共有空間での処理なので、便宜上、共通の１つのリソースアロケータとして、便宜上、ＰＵの外部において説明する。

この例では、大規模情報処理統合体ＷＯを構成する、ある１つのＳＰＵ４００１から、ＰＵ（＃Ｂ）５００３に、ビデオデータとオーディオデータとを合成してパッケージングして、自己の元に返して欲しいという内容のタスク要求が、パケットにより発信された場合の流れの例が示されている。

このタスク要求がＰＵ（＃Ｂ）５００３に渡されると（”１”）、ＰＵ（＃Ｂ）５００３は、リソースアロケータ５００７にそのタスクの実行に最適なＳＰＵクラスタの割当を依頼する（”２”）。リソースアロケータ５００７は、リソースリスト７０３のリソース情報に照らせば、ビデオデータについてはＳＰＵ４００３，オーディオデータについてはＳＰＵ４００５、パケット化プログラムはＳＰＵ４００７が適していると判定し、その旨をＰＵ（＃Ｂ）５００３に通知する（”３”）。ＰＵ（＃Ｂ）５００３は、そのＳＰＵを管理するＰＵのネットワークアドレス宛に、最初のタスクの依頼内容とタスクの実行結果の伝達すべき指定先であるＳＰＵ４００７の通信設定情報を含むパケットを各ＳＰＵ４００３，４００５に送信するための準備を行う。

すなわち、ビデオ用プログラムはＰＵ（＃Ａ）のＳＰＵ４００３、オーディオ

用プログラムはＰＵ（＃Ｃ）５００５のＳＰＵ４００５、パケット化プログラムはＰＵ＃ＡのＳＰＵ４００７に配置する。その後、データパスを、各プログラム内に設定する。

データパスは、タスク中に記述されたデータフローに基づき、各ＳＰＵに割り当てられた処理プログラムをキーとして、関数テーブル７０７の検索により取得する。例えば、ビデオ用プログラムであれば、以下のようになる。

「video.VIDEO_OUT -> packet.VIDEO_IN」

ＳＰＵの割当情報に基づきこの結果は、以下のように変換される。

「PU#A:SPU#●:VIDEO_OUT -> PU#B:SPU#○:VIDEO_IN」

その後、指定先の識別子を付加する。指定先の識別子の形態としては、例えば、ＴＣＰ、ＵＤＰなどの既存のプロトコルを利用するのであれば、ＰＵとＩＰアドレスを対応付け、さらに、ＳＰＵと通信ポート番号とを対応付ける。あるいは、各ＳＰＵ又はＰＵに個別のＩＰアドレスを割り当てる。あるいは、ＩＰアドレスによって指定先のＰＥ、ＢＥあるいはコンピュータ装置を指定し、これをローカル・ストレージに記録することで、既存のプロトコルを利用する。なお、これに限らず、ＩＰ以外のアドレス指定方法、例えばＳＰＵ毎のプロセッサ識別子でデータ送信が可能なネットワークのアドレスを指定するようにしても良い。以上の処理によって、例えばビデオデータの指定先を含むパケットのデータ部分は、以下のようなものとなる。

「192.168:0.1:10001:VIDEO_OUT -> 192.168.0.2:10002:VIDEO_IN」

このようなデータ部と、それがＳＰＵ４００３，４００５に到達するようなヘッダと利用するプロトコルに応じたパケットを生成し、これを上述した（図３）ネットワークインタフェースを通じてネットワーク１０４に送信する（”４”）。

ＳＰＵ４００３を管理するＰＵ（＃Ａ）５００１及びＳＰＵ４００５を管理するＰＵ（＃Ｃ）５００３は、それぞれ、ネットワークインタフェースからの割込を受けて、パケットを受け取り、ＳＰＵ４００３，４００５のローカル・メモリに、そのパケットの内容を書き込む。ＳＰＵ４００３，４００５は、これにより、自己に依頼されたタスクを実行する。ＳＰＵ４００３，４００５によるタスクの実行結果はパケット用プログラムと共にパケット化され、指定先であるＳＰＵ４

００７に伝達されるようにする（”５”）。ＳＰＵ４００７は、パッケージ用プログラムで、ビデオデータとオーディオデータとをパッケージ化し、その結果を各ＳＰＵ４００３、４００５が指定した指定先であるＳＰＵ４００１に向けて送信する（”６”）。これにより、ＳＰＵ４００１は、タスク要求に対応した実行結果を得ることができる。

なお、それぞれのＰＵは、パケットのヘッダ情報だけを解析することでパケットを受け取るべきＳＰＵを特定し、ＰＵは各ＳＰＵに対して必要なパケットデータをネットワークインタフェース或いはＤＲＡＭ上から取得するように依頼する実施の形態を採用することもできる。

また、上記の例では、パケット化プログラムが、ＳＰＵ４００３あるいはＳＰＵ４００５の実行結果と共にパケット化され、そのパケットがＳＰＵ４００７に送信されることでプログラムが実行される。そして、ＳＰＵ４００３，４００５のうち先に処理が終わった方のプログラムをＳＰＵ４００７で実行し、一方（オーディオまたはビデオ）からの実行結果を待つという形態例であるが、このような形態以外にも以下の形態が可能である。

すなわち、ビデオプログラム、オーディオプログラムと同様に”４”のメッセージによってパケット化プログラムをＳＰＵ４００７で実行し、ＳＰＵ４００３，ＳＰＵ４００５からの実行結果パケットの待ち受け状態にする。ＳＰＵ４００３，ＳＰＵ４００５は、実行結果をパケット化し、これをＳＰＵ４００７に送信する。このように、予めすべてのプログラムをＳＰＵ４００３，４００５，４００７に配置して、データのフローをつなぐタイプと、処理の進展に応じて先行するＳＰＵで実行結果と後続のプログラムを共にパケット化するタイプの２種類の実施が可能である。

図１４は、図１３の例において、ＳＰＵ４００５が、ＳＰＵ４００３からのパケットによって動作可能になる場合の例を示している（（”１”）～（”５”））。また、ＳＰＵ４００５からＳＰＵ４００７に送られた実行結果とＳＰＵプログラムは、ＳＰＵ４００７に送られる（”６”）。そして、ＳＰＵ４００７でパッケージ化した後、タスク要求を行ったＳＰＵ４００１ではなく、そのＳＰＵ４００１と同じグループにクラスタ化された他のＳＰＵ４００９に送信する（”７”）。

これらの例から判るように、ＰＵは、リソースアロケータ５００７に依頼して、最適なＳＰＵを特定し、特定したＳＰＵにタスクの依頼内容と指定先とを送信することにより、自己を介在させることなく、ＳＰＵだけで、以後のタスクの依頼、及びその実行結果の受け渡しが行われるので、タスク要求に際してのＰＵのオーバーヘッドが回避される。

また、予め想定されるタスクサイズ毎にクラスタ化しておき、タスク要求時に最適なＳＰＵクラスタを割り当てるようにしたので、ＳＰＵ割当に要する時間も短縮化される。このような効果は、リアルタイム性が要求される用途では、極めて意義の大きなものとなる。

なお、図１３および図１４の例において、ＰＵの機能を当該ＰＵの外部、例えば上述したネットワークインタフェースの一機能として内蔵させることで、ＰＵを介さずに直接ＳＰＵがネットワークインタフェースからの割り込みを受けて処理を行う実施形態をとることも可能である。

また、ＰＵ相当のインタフェースネットワークあるいは処理装置をネットワーク１０４上に配備し、あるＳＰＵが、目的のＳＰＵにダイレクトにアクセスするための通信設定情報等を、ＰＵを介さずに、ネットワークインタフェースあるいは上記の処理装置に、能動的に取りに行くという構成も可能である。この場合、ネットワークインタフェースないし処理装置には、ＰＵとほぼ同様のアーキテクチャ（プロセッサ等）と所定のプログラムとの協働によって、以下のような機能の全部又は一部の機能を形成するようにする。

（ａ）リソースアロケータ６０３、５００７、クラスタマネージャ６０５又はこれらと同様の機能。ＳＰＵステータステーブル７０１、リソースリスト７０３、クラスタリスト７０５については、自ら備えていても良いし、これらを管理するサーバないしＰＵにアクセスするようにしても良い。

（ｂ）リソースアロケータ６０３又は同様の機能実現体に、ネットワーク通信により、タスクの実行に最適なＳＰＵクラスタの割当を依頼し、あるいは自ら割り当てる機能。

（ｃ）クラスタマネージャ６０５又は同様の機能実現体に、ネットワーク通信により、いずれかのＳＰＵからタスク要求を受け付けたときに当該タスクのサイズ

又は種類に適合するクラスタをクラスタリストを参照することにより特定し、特定したクラスタに対して当該タスクの割当を依頼し、あるいは自ら割り当てる機能。

（ｄ）ＳＰＵ又はＳＰＵを管理するＰＵのネットワークアドレスを管理（自ら記録／更新するか、あるいは、記録されているサーバ等へのアクセスを行う）機能。

（ｅ）最初のタスクの依頼内容とタスクの実行結果の伝達すべき指定先であるＳＰＵの通信設定情報を含むパケットを、目的のＳＰＵへ送信するための準備を行う機能、あるいは、実際にパケットをＳＰＵへ送信する機能。

本発明によれば、処理管理装置が、タスク要求を受け取ったときに、そのタスク要求に対するタスクを遂行可能な処理装置を特定し、特定した処理装置とタスク要求元との間のダイレクトなタスクの実行結果の受け渡しを可能にするので、処理管理装置におけるオーバーヘッドを回避することができる。これにより、複数の処理装置による分散処理を効率的に行うことができる。

請 求 の 範 囲

1． それぞれ依頼されたタスクを実行し、その実行結果を依頼元の指定先へ送信する機能を備えた複数の処理装置が接続可能なネットワークに接続される処理管理装置であって；

前記ネットワークに接続されている処理装置のタスク実行能力を表すリソース情報と当該処理装置とのネットワーク通信を可能にするための通信設定情報とがリストアップされている所定のメモリへのアクセスを可能にする第１管理手段と；

ネットワーク通信によりいずれかの処理装置からタスクの実行要求を受け付けたときに当該タスクを実行可能な処理装置を前記メモリにリストアップされているリソース情報より特定し、特定した処理装置についての前記通信設定情報を前記メモリより取得するとともに、前記特定した処理装置と前記タスク要求を行った処理装置の少なくとも一方の処理装置に他方の処理装置の前記通信設定情報を伝達することにより、当該処理装置間のネットワーク通信によるダイレクトな前記実行結果の受け渡しを可能にする第２管理手段と；

を有する処理管理装置。

2． 前記メモリと、このメモリへの前記リソース情報および通信設定情報の記録を行うリソース情報等保持手段を有する、

請求の範囲第１項記載の処理管理装置。

3． 前記タスクは、当該タスクの実行に続く後続タスクの実行依頼とその後続タスクの実行結果の指定先の情報を含む、

請求の範囲第１項記載の処理管理装置。

4． 前記第２管理手段は、前記特定した処理装置に対して前記受け付けたタスクの実行結果の指定先を含む当該タスクの実行を依頼する、

請求の範囲第３項記載の処理管理装置。

5． 前記第２管理手段は、前記タスク要求を行った処理装置に、前記特定した処理装置へのダイレクトなタスクの実行を依頼させる、

請求の範囲第３項記載の処理管理装置。

6． 複数の処理装置が随時参加することができ且つ脱退できる共有空間を前記ネットワーク上に形成する共有空間モジュールを有し、

前記第2管理手段は、前記共有空間モジュールにより形成される共有空間に参加している処理装置の通信設定情報及び現在のリソース情報を当該処理装置より取得し、取得したこれらの情報を前記メモリにリストアップして使用可能状態にするとともに、前記共有空間から脱退した処理装置について前記メモリにリストアップされている情報を使用不能状態に変更する、

請求の範囲第1項記載の処理管理装置。

7． それぞれ依頼されたタスクを実行し、その実行結果を依頼元の指定先へ送信する機能を備えた処理装置と、この処理装置と内部バスを介して接続された処理管理装置とを含み；

前記処理装置及び前記処理管理装置は、それぞれ前記内部バスを介してネットワークに接続されており；

前記処理管理装置は；

前記ネットワークに接続されている処理装置のタスク実行能力を表すリソース情報と当該処理装置とのネットワーク通信を可能にするための通信設定情報とがリストアップされている所定のメモリへのアクセスを可能にする第1管理手段と；

ネットワーク通信によりいずれかの処理装置からタスクの実行要求を受け付けたときに当該タスクを実行可能な処理装置を前記メモリにリストアップされているリソース情報より特定し、特定した処理装置についての前記通信設定情報を前記メモリより取得するとともに、前記特定した処理装置と前記タスク要求を行った処理装置の少なくとも一方の処理装置に他方の処理装置の前記通信設定情報を伝達することにより、当該処理装置間のネットワーク通信によるダイレクトな前記実行結果の受け渡しを可能にする第2管理手段と；

を有するものである、コンピュータ・システム。

8． それぞれ依頼されたタスクを実行し、その実行結果を依頼元の指定先へ送信する機能を備えた複数の処理装置が接続可能なネットワークに接続されるコンピュータ・システムに搭載されるデバイスであって；

所定のコンピュータプログラムを実行することにより、前記コンピュータ・システムを；前記ネットワークに接続されている処理装置のタスク実行能力を表すリソース情報と当該処理装置とのネットワーク通信を可能にするための通信設定情報とがリストアップされている所定のメモリへのアクセスを可能にする第１管理手段；

ネットワーク通信によりいずれかの処理装置からタスクの実行要求を受け付けたときに当該タスクを実行可能な処理装置を前記メモリにリストアップされているリソース情報より特定し、特定した処理装置についての前記通信設定情報を前記メモリより取得するとともに、前記特定した処理装置と前記タスク要求を行った処理装置の少なくとも一方の処理装置に他方の処理装置の前記通信設定情報を伝達することにより、当該処理装置間のネットワーク通信によるダイレクトな前記実行結果の受け渡しを可能にする第２管理手段；として動作させる、

分散処理管理用のデバイス。

９．　それぞれ依頼されたタスクを実行し、実行結果を依頼元の指定先へ送信する機能を備えた複数の処理装置と、各処理装置との間でネットワークを通じて通信を行う処理管理装置との協働で行う分散処理方法であって、

前記処理管理装置が、前記ネットワークに接続されている処理装置のタスク実行能力を表すリソース情報および当該処理装置へのアクセスを可能にするための通信設定情報を当該処理装置より取得し、取得したこれらの情報を所定のメモリにリストアップする段階と、

いずれかの前記処理装置が前記処理管理装置に対してタスク要求を送信する段階と、

前記タスク要求を受信した前記処理管理装置が、受信したタスク要求を遂行可能な１又は複数の前記処理装置を前記メモリの記録情報より特定し、特定した処理装置に、その実行結果の指定先を含むタスクの実行を依頼する段階とを有し、

前記処理管理装置を介在させない前記実行結果の受け渡しを可能にする、

分散処理方法。

１０．　前記複数の処理装置のすべてが共通の構造のプロセッサ・エレメントを有する、

請求の範囲第９項記載の分散処理方法。

１１．　前記複数の処理装置が有するプロセッサエレメントは、命令セット・アーキテクチャ（ＩＳＡ）がすべて同じで、同じ命令セットに従って所要の処理を実行できるものである、

請求の範囲第１０項記載の分散処理方法。

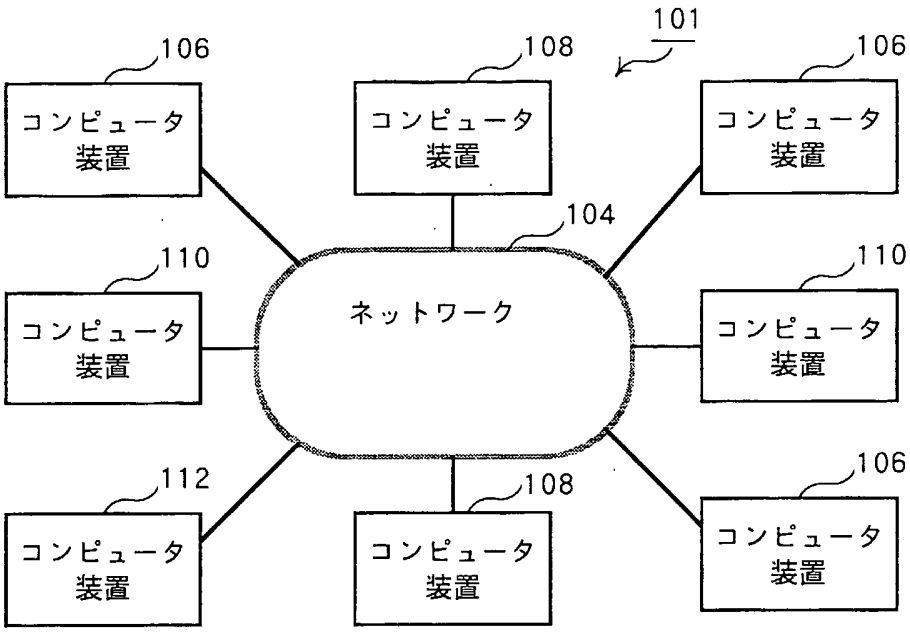
１２．　コンピュータを、それぞれ依頼されたタスクを実行し、その実行結果を依頼元の指定先へ送信する機能を備えた複数の処理装置が接続可能なネットワークに接続される処理管理装置として動作させるためのプログラムであって；

前記コンピュータを；

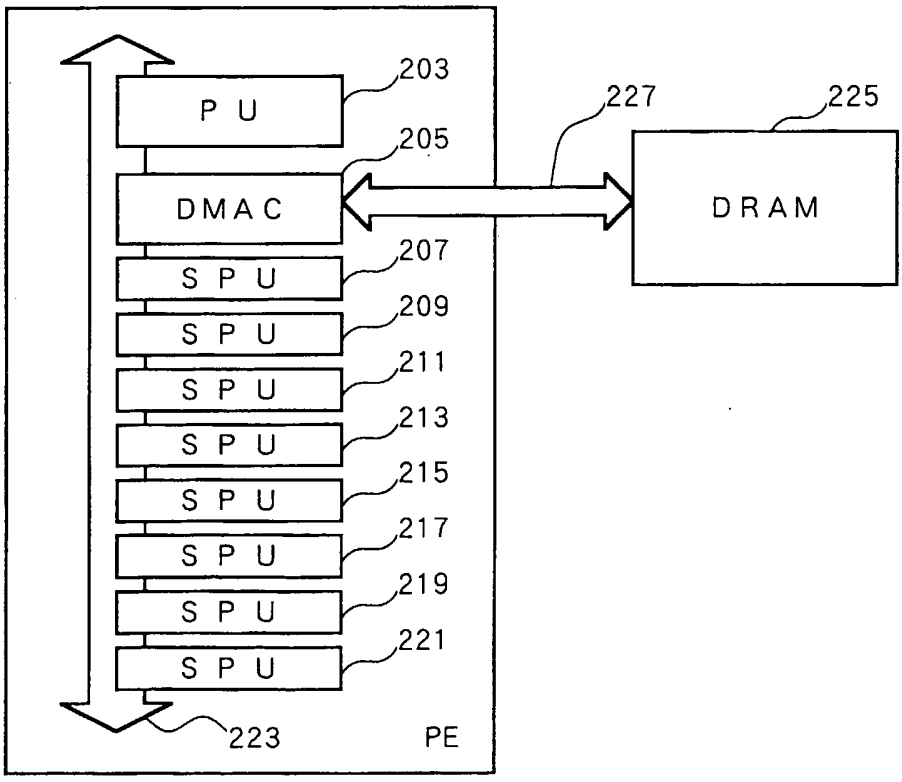
前記ネットワークに接続されている処理装置のタスク実行能力を表すリソース情報と当該処理装置とのネットワーク通信を可能にするための通信設定情報とがリストアップされている所定のメモリへのアクセスを可能にする第１管理手段；

ネットワーク通信によりいずれかの処理装置からタスクの実行要求を受け付けたときに当該タスクを実行可能な処理装置を前記メモリにリストアップされているリソース情報より特定し、特定した処理装置についての前記通信設定情報を前記メモリより取得するとともに、前記特定した処理装置と前記タスク要求を行った処理装置の少なくとも一方の処理装置に他方の処理装置の前記通信設定情報を伝達することにより、当該処理装置間のネットワーク通信によるダイレクトな前記実行結果の受け渡しを可能にする第２管理手段；

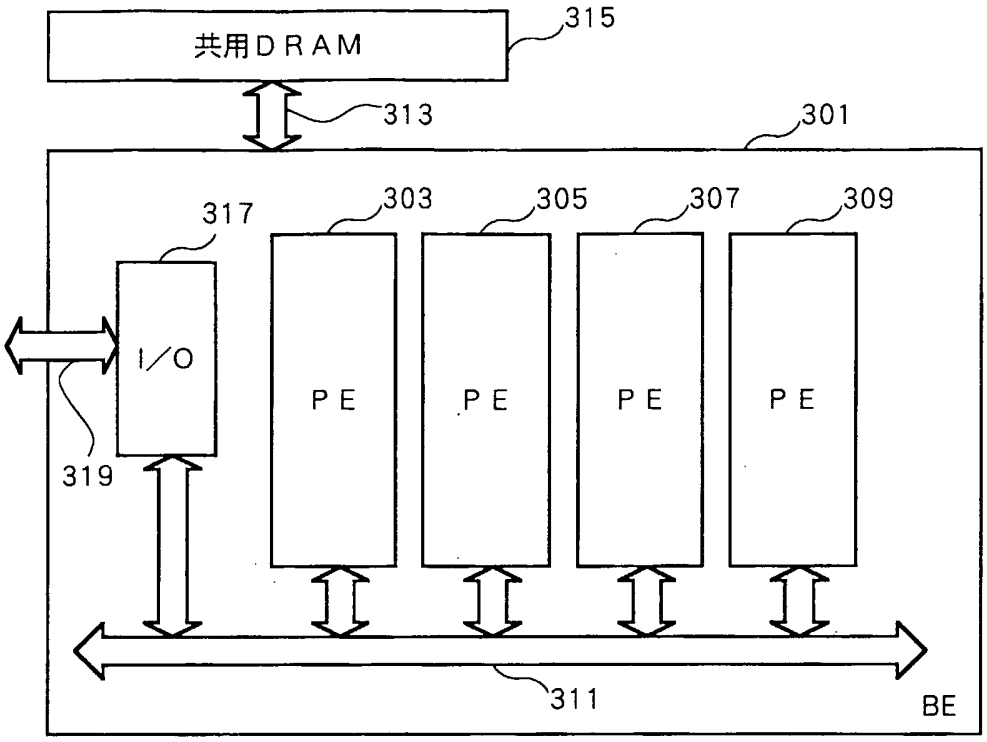
として機能させるためのコンピュータプログラム。


101
106
108
106
コンピュータ装置
コンピュータ装置
コンピュータ装置
104
110
110
ネットワーク
コンピュータ装置
コンピュータ装置
112
108
106
コンピュータ装置
コンピュータ装置
コンピュータ装置

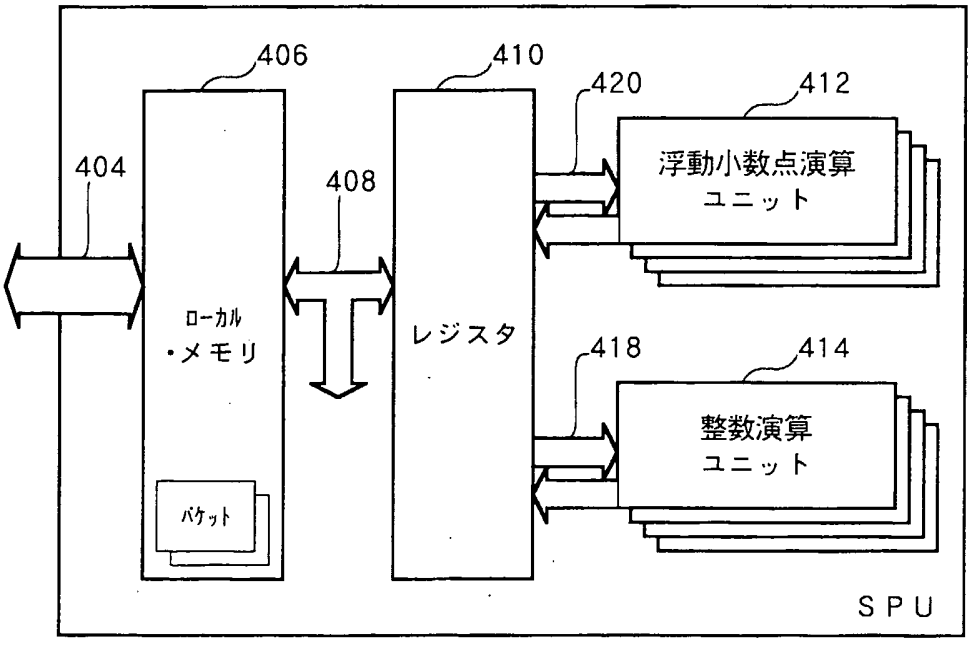
第1図


203
P U
227
225
205
DMAC
DRAM
207
S P U
209
S P U
211
S P U
213
S P U
215
S P U
217
S P U
219
S P U
221
S P U
223
PE

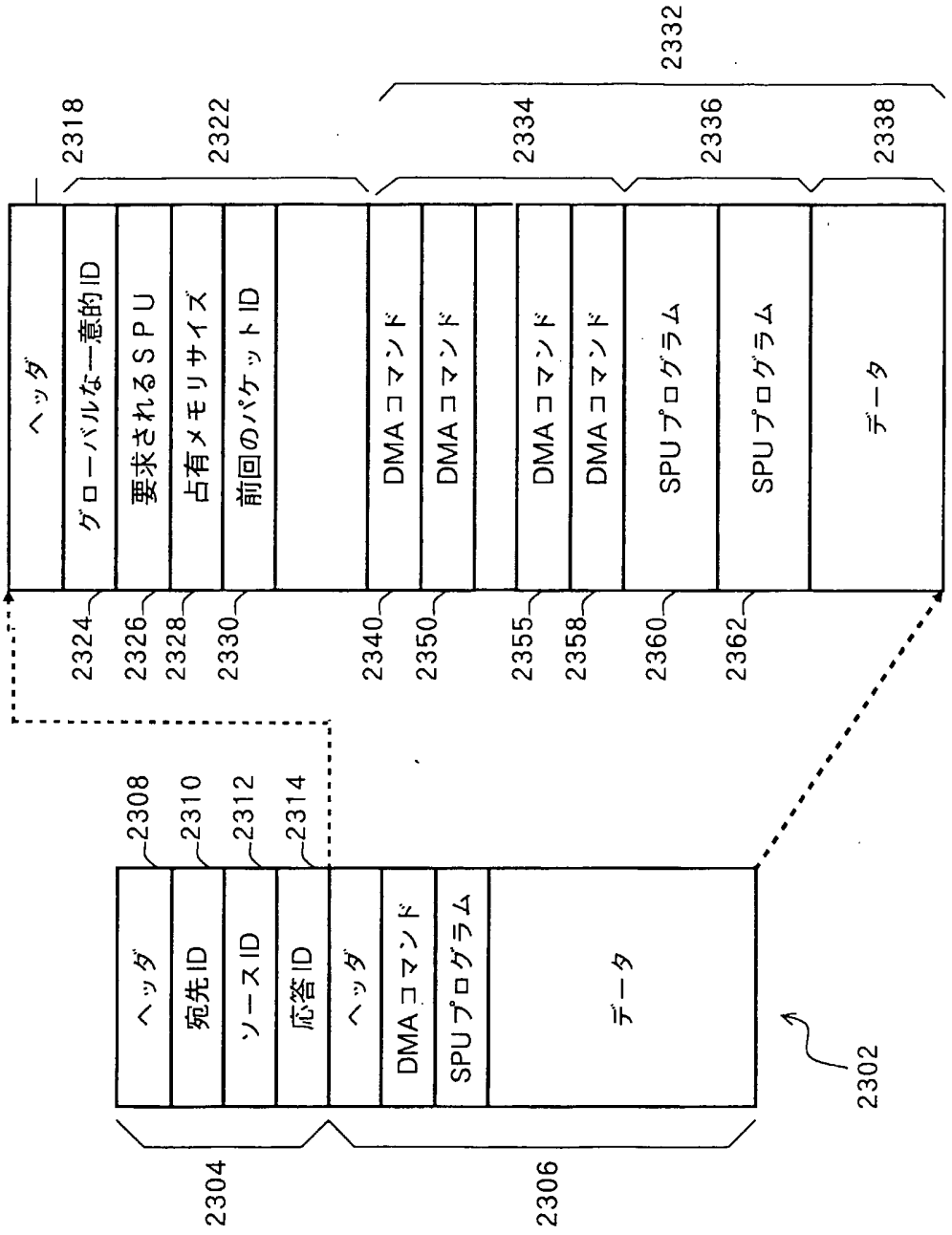
第2図


共用ＤＲＡＭ
315
313
301
317
303
305
307
309
I／O
ＰＥ
ＰＥ
ＰＥ
ＰＥ
319
311
BE

第3図

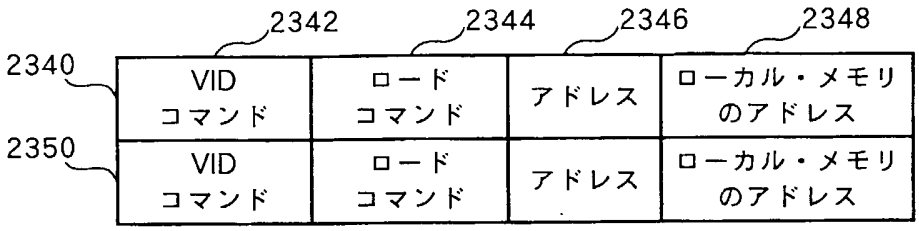

406
410
420
412
404
408
浮動小数点演算
ユニット
ローカル
・メモリ
レジスタ
418
414
整数演算
ユニット
バケット
ＳＰＵ

第4図

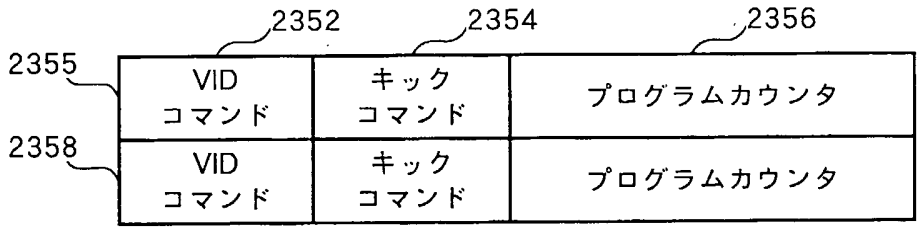

2332
2318
2322
2334
2336
2338
ヘッダ
グローバルな一意的ID
要求されるSPU
占有メモリサイズ
前回のパケットID
DMAコマンド
DMAコマンド
DMAコマンド
DMAコマンド
SPUプログラム
SPUプログラム
データ
2324
2326
2328
2330
2340
2350
2355
2358
2360
2362
2308
2310
2312
2314
ヘッダ
宛先ID
ソースID
応答ID
ヘッダ
DMAコマンド
SPUプログラム
データ
2302
2304
2306

第5図

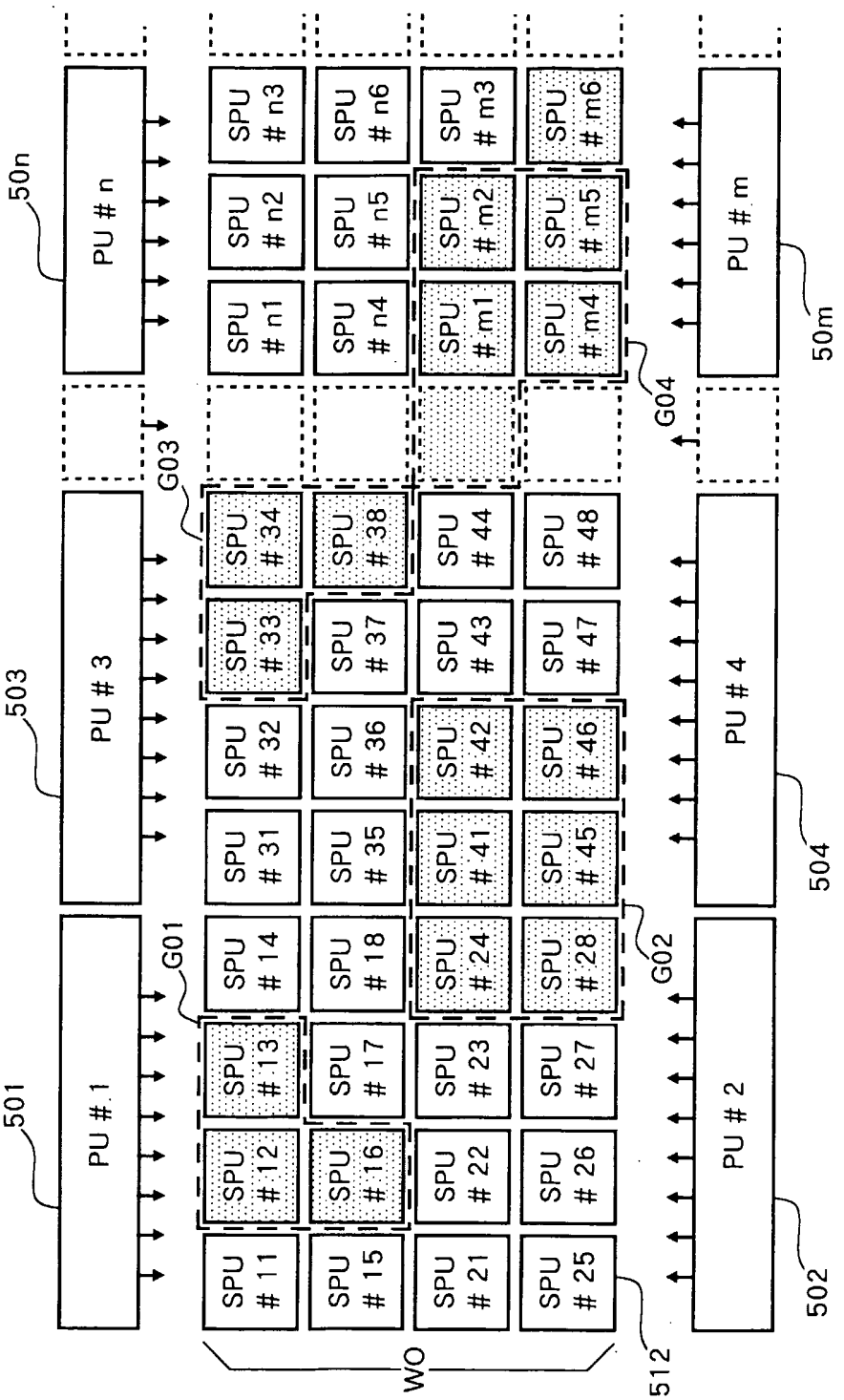
(a)


2342
2344
2346
2348
2340
VID コマンド
ロード コマンド
アドレス
ローカル・メモリ のアドレス
2350
VID コマンド
ロード コマンド
アドレス
ローカル・メモリ のアドレス

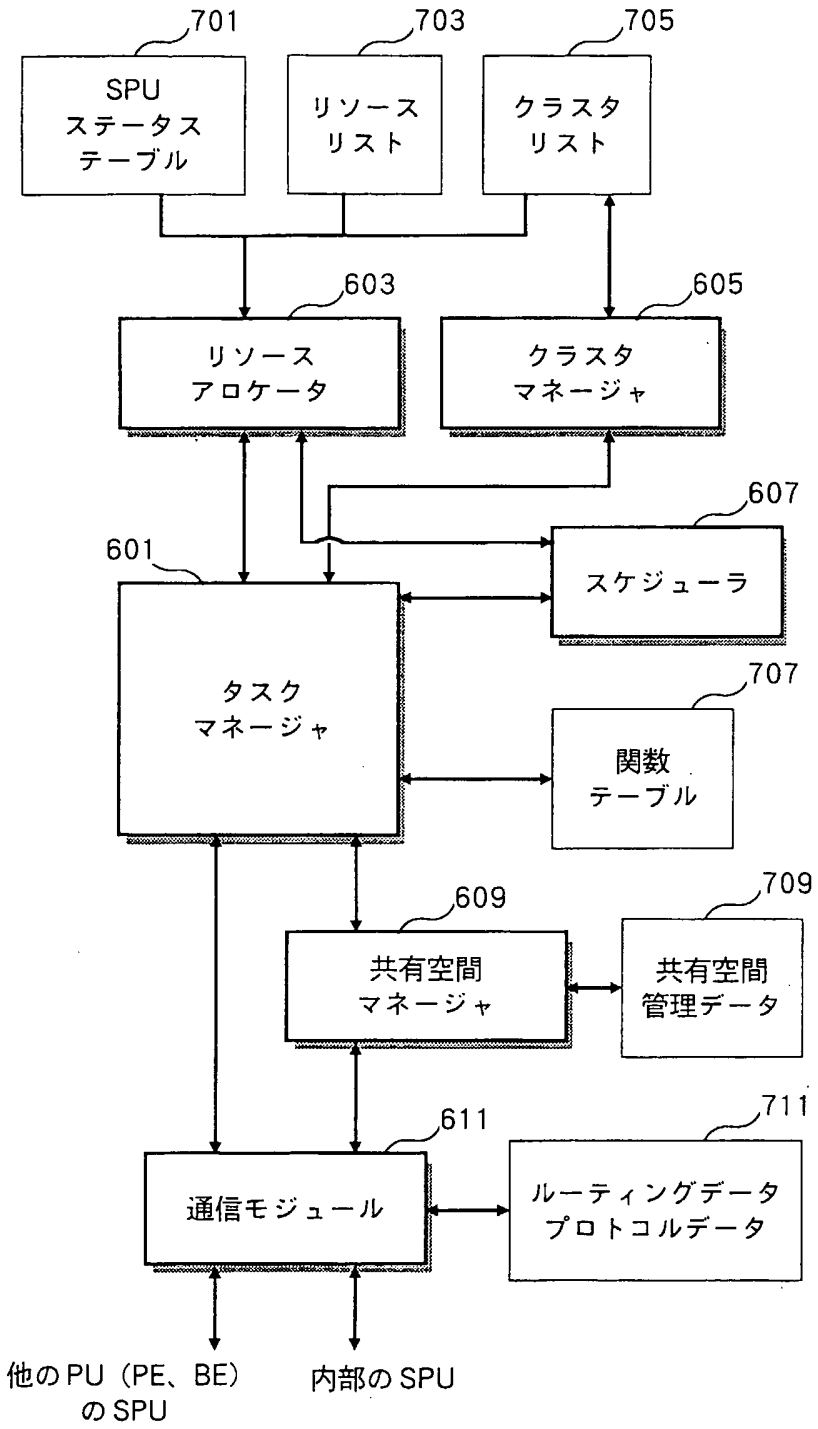
(b)


2352
2354
2356
2355
VID コマンド
キック コマンド
プログラムカウンタ
2358
VID コマンド
キック コマンド
プログラムカウンタ

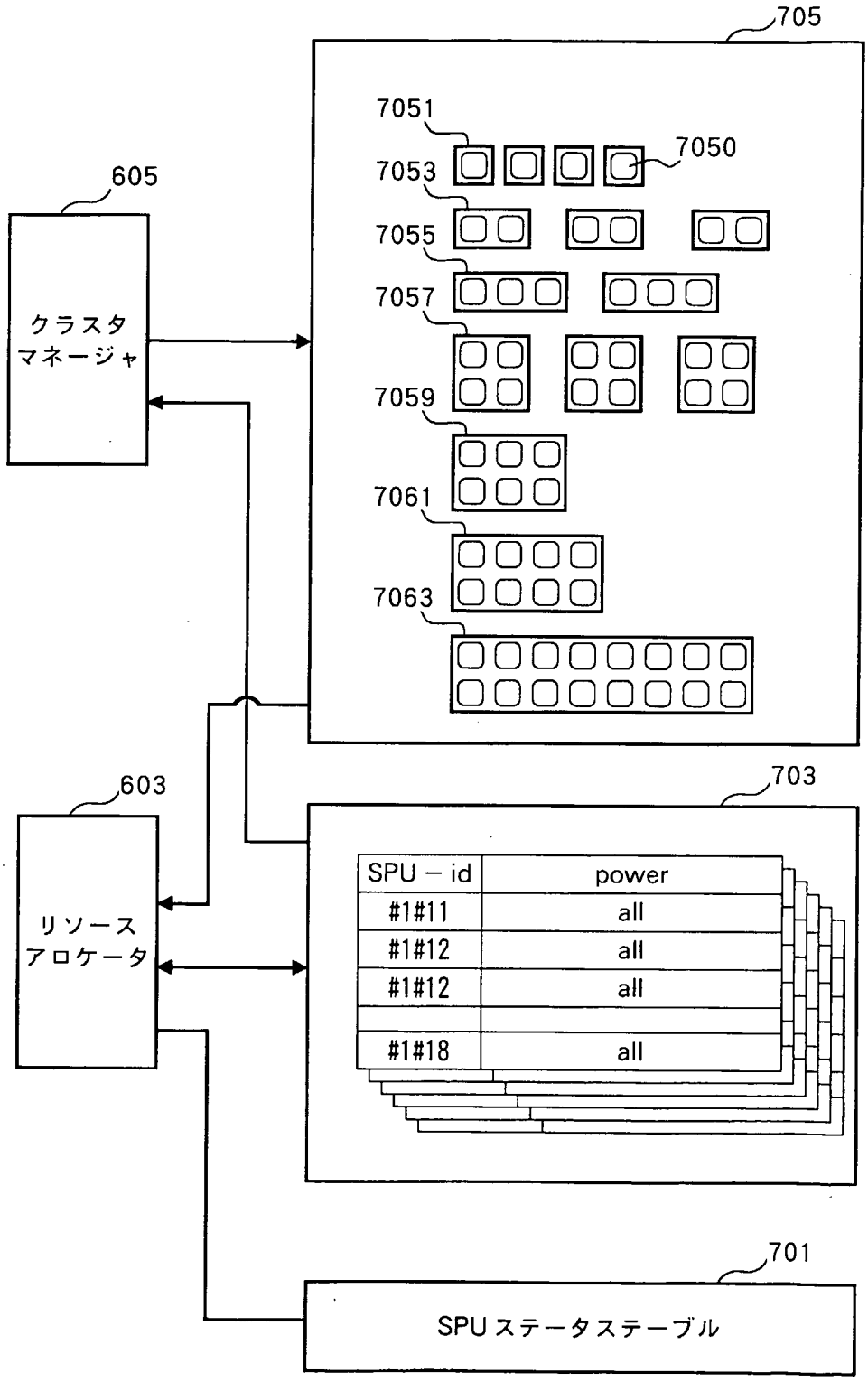
第6図


PU # 1
PU # 2
PU # 3
PU # 4
PU # n
PU # m
501
502
503
504
50n
50m
G01
G02
G03
G04
WO
512
SPU # 11
SPU # 12
SPU # 13
SPU # 14
SPU # 15
SPU # 16
SPU # 17
SPU # 18
SPU # 21
SPU # 22
SPU # 23
SPU # 24
SPU # 25
SPU # 26
SPU # 27
SPU # 28
SPU # 31
SPU # 32
SPU # 33
SPU # 34
SPU # 35
SPU # 36
SPU # 37
SPU # 38
SPU # 41
SPU # 42
SPU # 43
SPU # 44
SPU # 45
SPU # 46
SPU # 47
SPU # 48
SPU # n1
SPU # n2
SPU # n3
SPU # n4
SPU # n5
SPU # n6
SPU # m1
SPU # m2
SPU # m3
SPU # m4
SPU # m5
SPU # m6

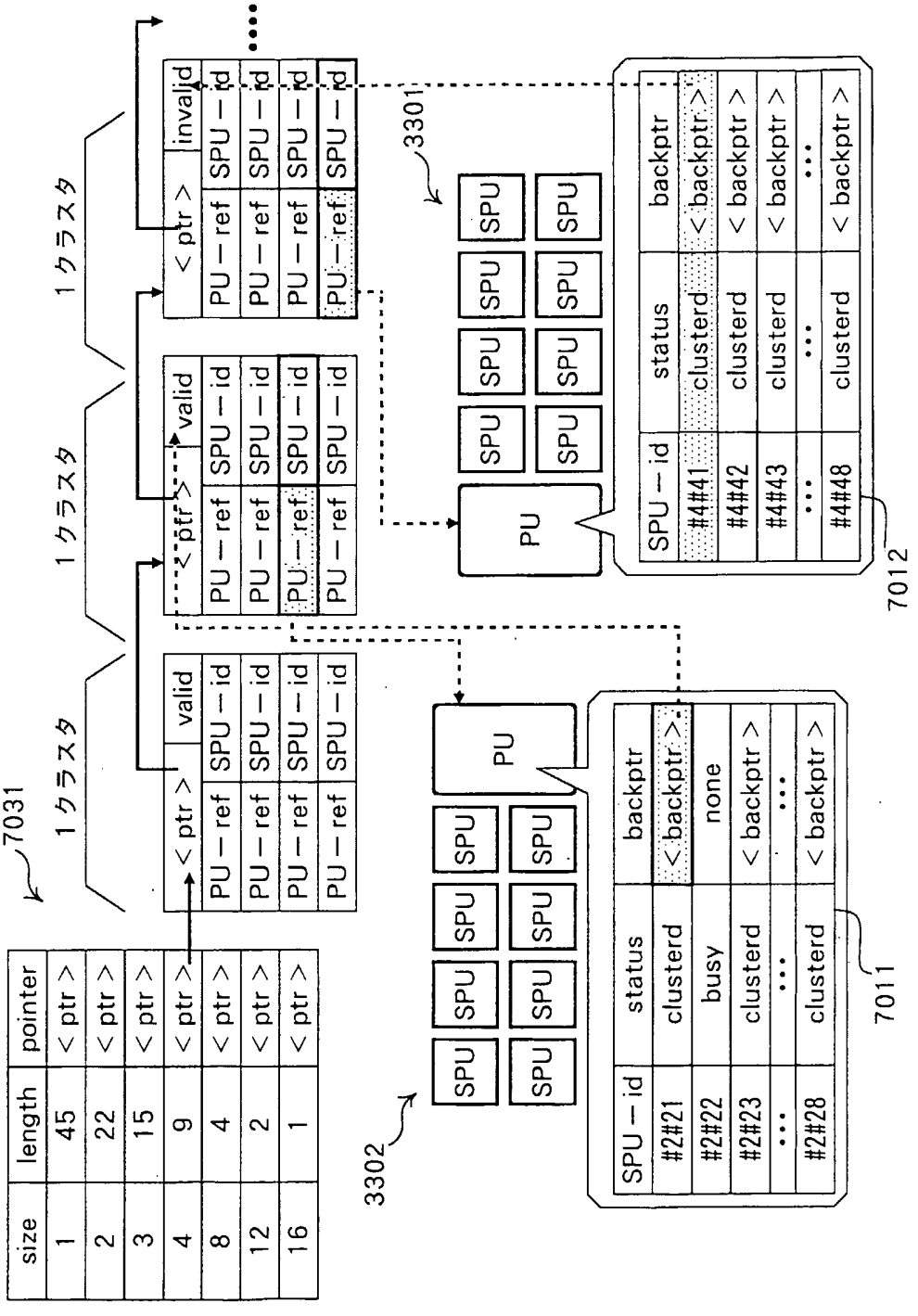
第7図


701
SPU
ステータス
テーブル
703
リソース
リスト
705
クラスタ
リスト
603
リソース
アロケータ
605
クラスタ
マネージャ
607
スケジューラ
601
タスク
マネージャ
707
関数
テーブル
609
共有空間
マネージャ
709
共有空間
管理データ
611
通信モジュール
711
ルーティングデータ
プロトコルデータ
他のPU（PE、BE）
のSPU
内部のSPU

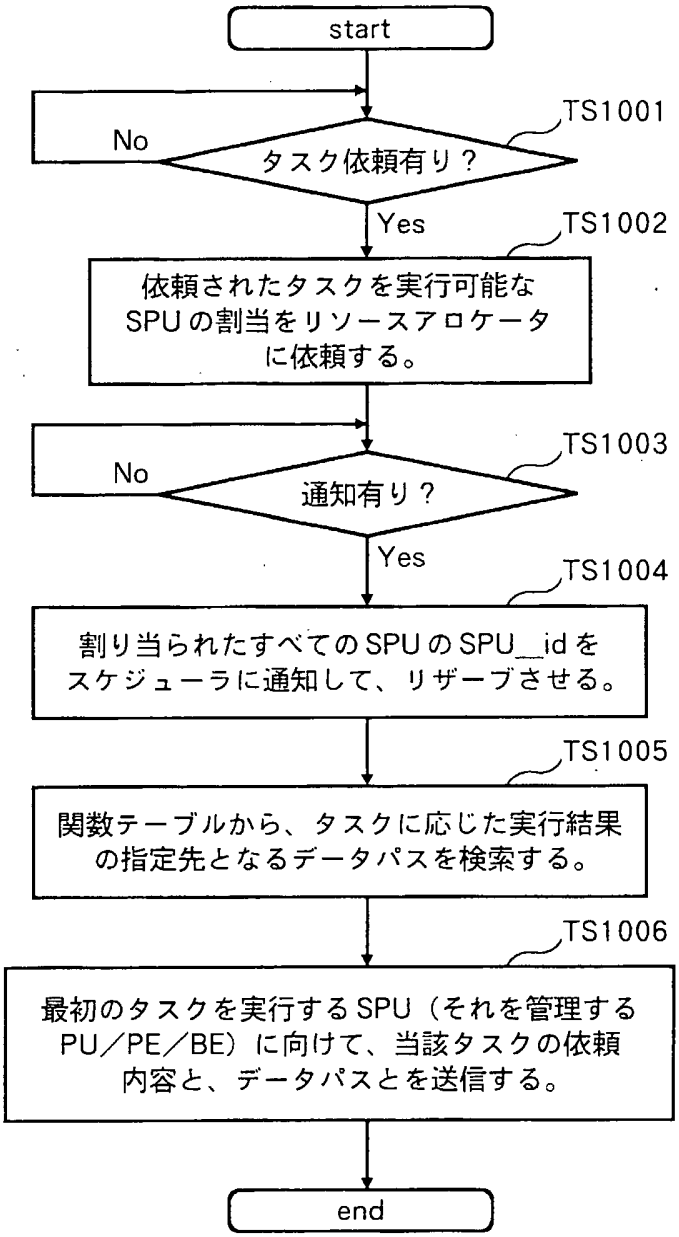
第8図


705
7051
7050
7053
7055
7057
7059
7061
7063
605
クラスタ
マネージャ
603
リソース
アロケータ
703
SPU – id
power
#1#11
all
#1#12
all
#1#12
all
#1#18
all
701
SPU ステータステーブル

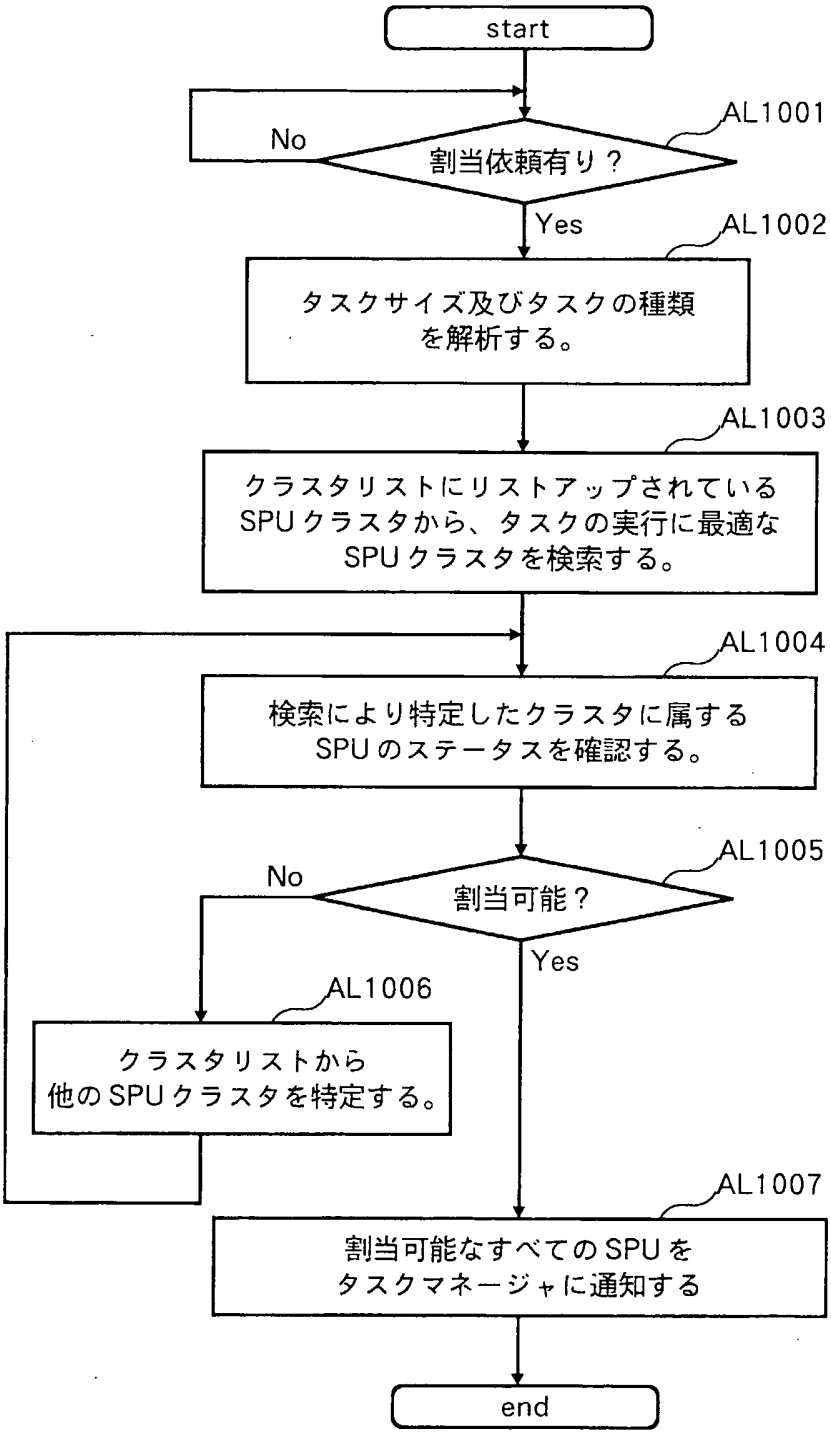
第9図


size
length
pointer
1
45
< ptr >
2
22
< ptr >
3
15
< ptr >
4
9
< ptr >
8
4
< ptr >
12
2
< ptr >
16
1
< ptr >
7031
1クラスタ
< ptr >
valid
PU − ref
SPU − id
invalid
3301
3302
PU
SPU
SPU − id
status
backptr
clusterd
busy
none
< backptr >
#2#21
#2#22
#2#23
#2#28
#4#41
#4#42
#4#43
#4#48
7011
7012

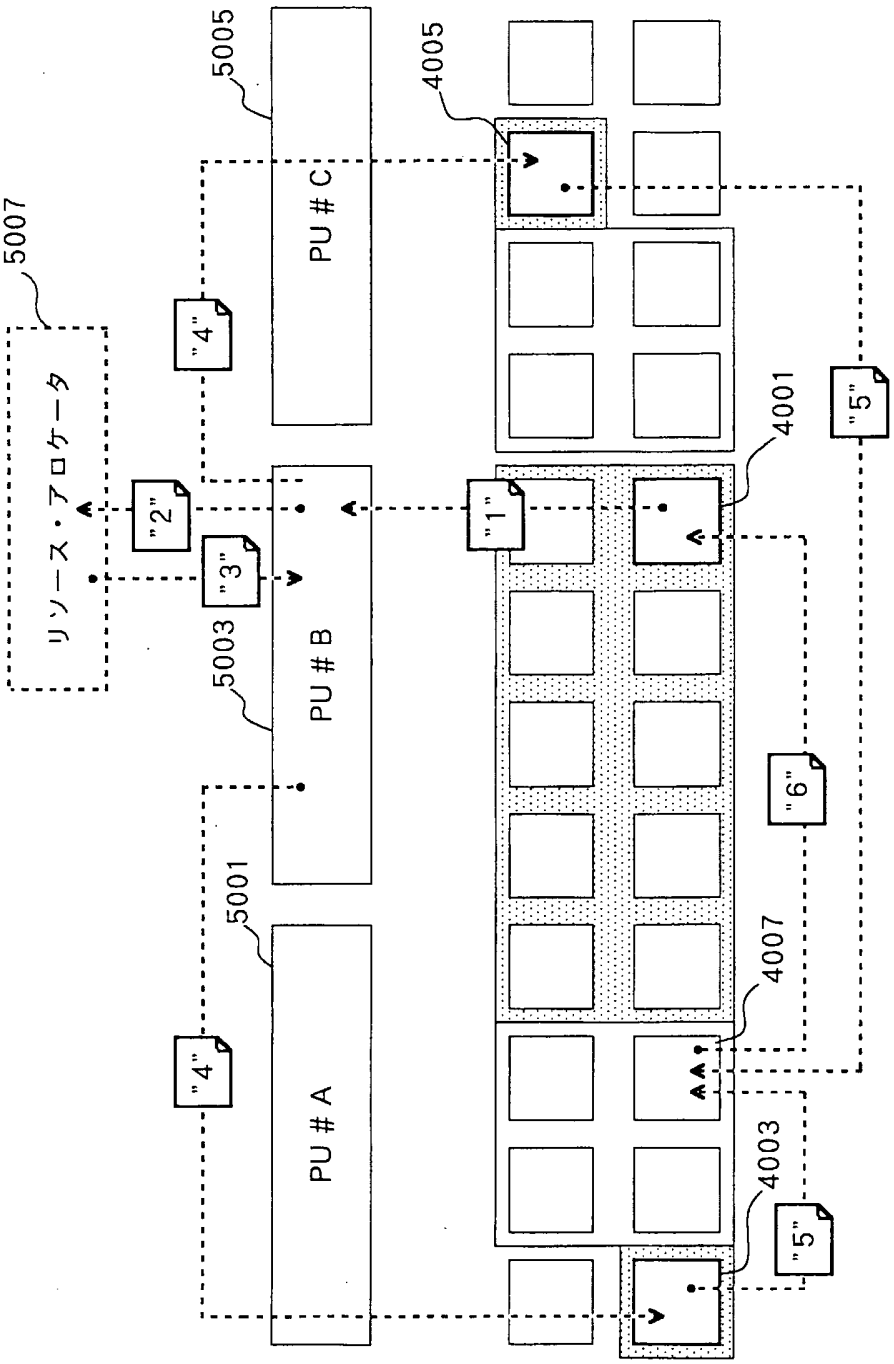
第10図


start
TS1001
No
タスク依頼有り？
Yes
TS1002
依頼されたタスクを実行可能な
SPUの割当をリソースアロケータ
に依頼する。
TS1003
No
通知有り？
Yes
TS1004
割り当られたすべてのSPUのSPU_idを
スケジューラに通知して、リザーブさせる。
TS1005
関数テーブルから、タスクに応じた実行結果
の指定先となるデータパスを検索する。
TS1006
最初のタスクを実行するSPU（それを管理する
PU／PE／BE）に向けて、当該タスクの依頼
内容と、データパスとを送信する。
end

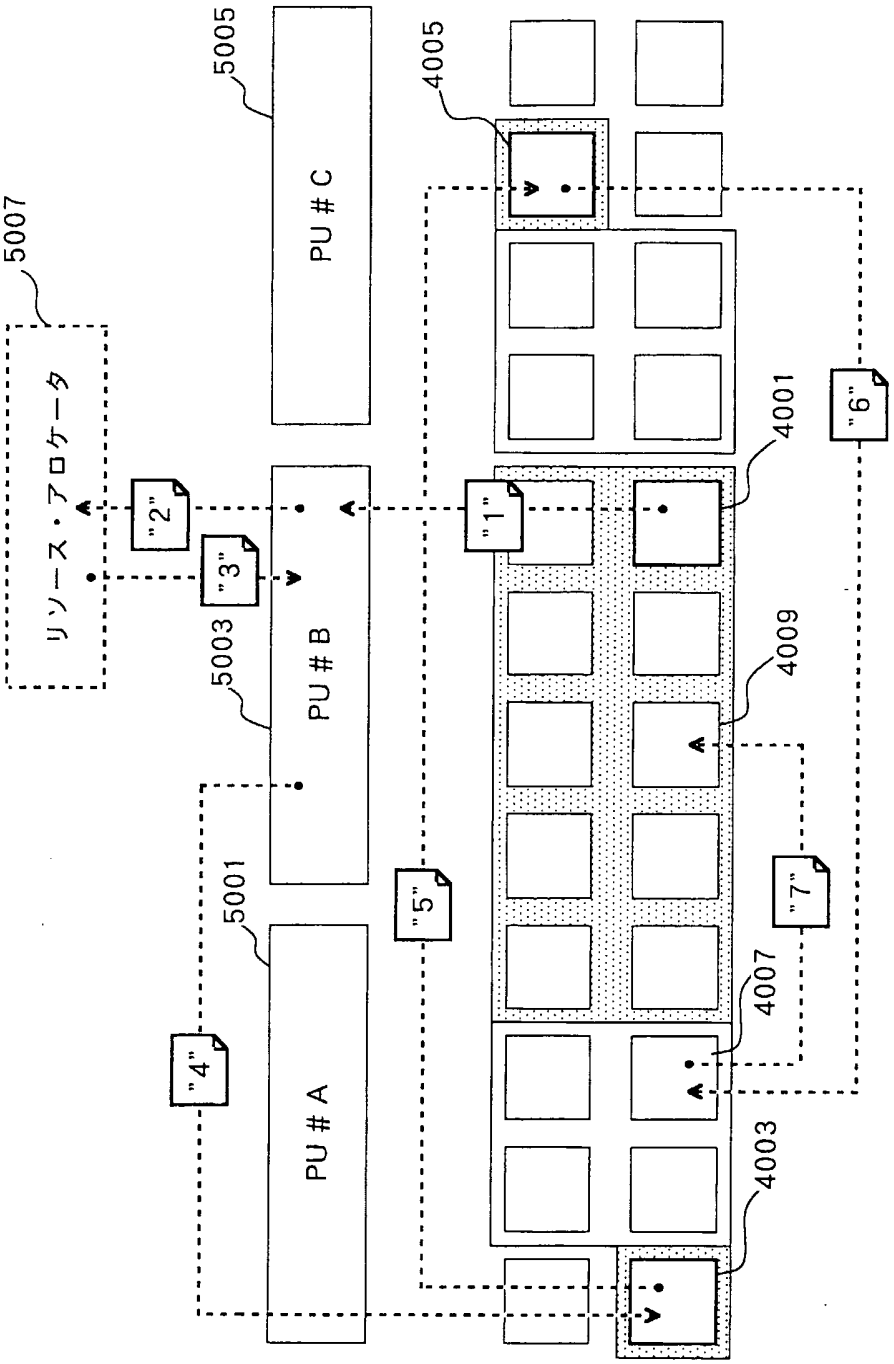
第11図


start
AL1001
No
割当依頼有り？
Yes
AL1002
タスクサイズ及びタスクの種類
を解析する。
AL1003
クラスタリストにリストアップされている
SPU クラスタから、タスクの実行に最適な
SPU クラスタを検索する。
AL1004
検索により特定したクラスタに属する
SPU のステータスを確認する。
AL1005
No
割当可能？
Yes
AL1006
クラスタリストから
他の SPU クラスタを特定する。
AL1007
割当可能なすべての SPU を
タスクマネージャに通知する
end

第12図


5005
PU # C
4005
5007
リソース・アロケータ
"4"
"2"
"3"
"1"
4001
"5"
5003
PU # B
"6"
5001
PU # A
4007
"4"
4003
"5"

第13図


5005
PU # C
4005
5007
リソース・アロケータ
"2"
"3"
"1"
4001
"6"
5003
PU # B
4009
5001
"5"
"7"
"4"
PU # A
4007
4003

第14図

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP2005/010986

A. CLASSIFICATION OF SUBJECT MATTER
Int.Cl[7] G06F9/50

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)
Int.Cl[7] G06F9/50

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 1996-2005
Kokai Jitsuyo Shinan Koho 1971-2005 Toroku Jitsuyo Shinan Koho 1994-2005

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.
P,X	JP 2004-287801 A (Sony Computer Entertainment Inc.), 14 October, 2004 (14.10.04), Par. Nos. [0009], [0031] to [0033], [0047] to [0048], [0060]	1-5,7-12
Y	JP 7-056754 A (International Business Machines Corp.), 03 March, 1995 (03.03.95), Par. Nos. [0053] to [0061]	1-12
Y	JP 7-141302 A (Director General, Agency of Industrial Science and Technology), 02 June, 1995 (02.06.95), Par. Nos. [0017] to [0019]	1-12

[X] Further documents are listed in the continuation of Box C.

[X] 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	Date of mailing of the international search report
12 September, 2005 (12.09.05)	27 September, 2005 (27.09.05)
Name and mailing address of the ISA/ Japanese Patent Office Facsimile No.	Authorized officer Telephone No.

INTERNATIONAL SEARCH REPORT

International application No.
PCT/JP2005/010986

C (Continuation). DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	JP 6-348662 A (Fuji Xerox Co., Ltd.), 22 December, 1994 (22.12.94), Par. Nos. [0013] to [0014], [0020]	1-12
Y	JP 8-083253 A (Toshiba Corp.), 26 March, 1996 (26.03.96), Par. Nos. [0021], [0027]	6

JP 2004-287801 A	2004.10.14	US 2004/205759 A1 WO 2004/084069 A2	2004.10.14 2004.09.30
JP 7-056754 A	1995.03.03	US 5574911 A1	1996.11.12
JP 7-141302 A	1995.06.02	(Family: none)	
JP 6-348662 A	1994.12.22	(Family: none)	
JP 8-083253 A	1996.03.26	US 5829041 A1	1998.10.27

国際調査報告

国際出願番号　ＰＣＴ／ＪＰ２００５／０１０９８６

Ａ．　発明の属する分野の分類（国際特許分類（ＩＰＣ））

Int.Cl.[7]　G06F9/50

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（ＩＰＣ））

Int.Cl.[7]　G06F9/50

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	１９２２－１９９６年
日本国公開実用新案公報	１９７１－２００５年
日本国実用新案登録公報	１９９６－２００５年
日本国登録実用新案公報	１９９４－２００５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	関連する請求の範囲の番号
P, X	JP 2004-287801 A（株式会社ソニー・コンピュータエンタテインメント）2004.10.14, 段落【０００９】、【００３１】－【００３３】、 【００４７】－【００４８】、【００６０】	1 - 5, 7 - 12
Y	JP 7-056754 A（インターナショナル・ビジネス・マシーンズ・コーポレイション）1995.03.03, 段落【００５３】－【００６１】	1 - 12

☑ 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 １２．０９．２００５	国際調査報告の発送日 27.9.2005
国際調査機関の名称及びあて先 日本国特許庁（ＩＳＡ／ＪＰ） 郵便番号１００－８９１５ 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）　５Ｂ　９６４６ 殿川　雅也 電話番号　０３－３５８１－１１０１　内線　３５４４

様式ＰＣＴ／ＩＳＡ／２１０（第２ページ）（２００４年１月）

Ｃ（続き）．　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	関連する請求の範囲の番号
Y	JP 7-141302 A（工業技術院長）1995.06.02, 段落【００１７】－【００１９】	1 - 12
Y	JP 6-348662 A（富士ゼロックス株式会社）1994.12.22, 段落【００１３】－【００１４】、【００２０】	1 - 12
Y	JP 8-083253 A（株式会社東芝）1996.03.26, 段落【００２１】、 【００２７】	6

国際調査報告
パテントファミリーに関する情報

国際出願番号 ＰＣＴ／ＪＰ２００５／０１０９８６

JP 2004-287801 A	2004.10.14	US 2004/205759 A1	2004.10.14
		WO 2004/084069 A2	2004.09.30
JP 7-056754 A	1995.03.03	US 5574911 A1	1996.11.12
JP 7-141302 A	1995.06.02	ファミリーなし	
JP 6-348662 A	1994.12.22	ファミリーなし	
JP 8-083253 A	1996.03.26	US 5829041 A1	1998.10.27