

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号

特許第6475256号
(P6475256)

(45) 発行日 平成31年2月27日 (2019. 2. 27)

(24) 登録日 平成31年2月8日 (2019. 2. 8)

(51) Int. Cl.		F I			
G06F 13/10	(2006.01)	G06F 13/10	310D		
G06F 9/50	(2006.01)	G06F 9/50	120A		
		G06F 13/10	330D		

請求項の数 16 (全 26 頁)

(21) 出願番号	特願2016-553382 (P2016-553382)	(73) 特許権者	503433420
(86) (22) 出願日	平成27年2月10日 (2015. 2. 10)		華為技術有限公司
(65) 公表番号	特表2017-513096 (P2017-513096A)		HUAWEI TECHNOLOGIES
(43) 公表日	平成29年5月25日 (2017. 5. 25)		CO., LTD.
(86) 国際出願番号	PCT/CN2015/072672		中華人民共和国 518129 広東省深
(87) 国際公開番号	W02015/165298		▲チェン▼市龍崗区坂田 華為総部▲ベン
(87) 国際公開日	平成27年11月5日 (2015. 11. 5)		▼公楼
審査請求日	平成28年8月19日 (2016. 8. 19)		Huawei Administrati
(31) 優先権主張番号	201410182148.1		on Building, Bantia
(32) 優先日	平成26年4月30日 (2014. 4. 30)		n, Longgang Distric
(33) 優先権主張国	中国 (CN)		t, Shenzhen, Guangd
(31) 優先権主張番号	201410682375.0		ong 518129, P. R. Ch
(32) 優先日	平成26年11月24日 (2014. 11. 24)		ina
(33) 優先権主張国	中国 (CN)	(74) 代理人	100107766
前置審査			弁理士 伊東 忠重
			最終頁に続く

(54) 【発明の名称】 コンピュータ、制御デバイス及びデータ処理方法

(57) 【特許請求の範囲】

【請求項 1】

処理ユニットと制御デバイスとを有するコンピュータであり、前記制御デバイスは、前記コンピュータのコンポーネントに結合され、前記コンポーネントは、チップ上高速相互接続ネットワークと、キャッシュと、メモリと、入出力I/O相互接続ネットワークと、ネットワークアダプタと、ディスプレイとのうちいずれか1つを有する、コンピュータであって、

前記処理ユニットは、アプリケーション要求にタグを添付し、前記添付されたアプリケーション要求を前記制御デバイスに送信するように構成され、

前記制御デバイスは、前記添付されたアプリケーション要求を受信し、前記タグ及び予め記憶されたリソース割り当てポリシーに従って、前記アプリケーション要求のために割り当てられるリソースの量を決定するように構成され、前記リソース割り当てポリシーは、前記タグと前記アプリケーション要求のために割り当てられる前記リソースの量との間の対応関係を有し、前記コンポーネントに対して、前記アプリケーション要求のために割り当てられる前記リソースの量に従って前記アプリケーション要求を処理するように命令するように構成されるコンピュータ。

【請求項 2】

前記制御デバイスはバッファを有し、前記バッファは少なくとも2つのキューを有し、各キューはリソースの量の範囲に対応し、各キューの優先度は異なり、

前記制御デバイスは、前記アプリケーション要求のために割り当てられる前記リソース

の量に従って、前記少なくとも2つのキューから前記アプリケーション要求に対応するキューを選択し、前記アプリケーション要求に対応する前記キューに前記アプリケーション要求を記憶するように構成され、

前記コンポーネントは、各キューの前記優先度に従って前記バッファ内の前記キューを別々に処理するように構成される、請求項1に記載のコンピュータ。

【請求項3】

前記制御デバイスはプロセッサとキャッシュとを有し、前記キャッシュは前記リソース割り当てポリシーを記憶し、

前記プロセッサは、前記キャッシュから前記リソース割り当てポリシーを獲得するように構成され、前記プロセッサは、前記添付されたアプリケーション要求を受信し、前記タグ及び前記予め記憶されたリソース割り当てポリシーに従って、前記アプリケーション要求のために割り当てられる前記リソースの量を決定するように構成され、前記リソース割り当てポリシーは、前記タグと前記アプリケーション要求のために割り当てられる前記リソースの量との間の前記対応関係を有し、前記コンポーネントに対して、前記アプリケーション要求のために割り当てられる前記リソースの量に従って前記アプリケーション要求を処理するように命令するように構成される、請求項1に記載のコンピュータ。

【請求項4】

前記リソース割り当てポリシーは制御テーブルを有し、前記制御テーブルは複数のエントリを有し、前記複数のエントリのうち1つのエントリは、前記タグと前記アプリケーション要求のために割り当てられる前記リソースの量との間の前記対応関係を有し、

前記プロセッサは、クエリ命令を前記キャッシュに送信するように構成され、前記クエリ命令は前記タグを有し、

前記キャッシュは、前記クエリ命令に従って、前記タグに対応するエントリを取得し、前記タグに対応する前記エントリを前記制御デバイスの前記プロセッサに送信するように構成される、請求項3に記載のコンピュータ。

【請求項5】

前記コンピュータはメモリを更に有し、前記メモリはノード管理ソフトウェアを記憶し、

前記処理ユニットは、前記ノード管理ソフトウェアを使用することにより前記リソース割り当てポリシーを定義するように更に構成され、

前記プロセッサは、前記ノード管理ソフトウェアから前記リソース割り当てポリシーを獲得し、前記リソース割り当てポリシーを前記キャッシュに書き込むように更に構成される、請求項3又は4に記載のコンピュータ。

【請求項6】

前記処理ユニットはタグレジスタを更に有し、

前記処理ユニットは、前記ノード管理ソフトウェアを使用することにより前記タグを定義し、前記ノード管理ソフトウェアを使用することにより前記タグを前記タグレジスタに書き込むように更に構成され、

前記処理ユニットは、前記タグレジスタから前記タグを読み取るように更に構成される、請求項5に記載のコンピュータ。

【請求項7】

コンピュータのコンポーネントに結合された制御デバイスであり、前記コンポーネントは、チップ上高速相互接続ネットワークと、キャッシュと、メモリと、入出力I/O相互接続ネットワークと、ネットワークアダプタと、ディスプレイとのうちいずれか1つを有し、前記制御デバイスは、プロセッサを有し、前記コンピュータは、処理ユニットと、前記制御デバイスとを有する、制御デバイスであって、

前記プロセッサは、前記処理ユニットによりタグが添付されたアプリケーション要求を受信し、前記タグ及び予め記憶されたリソース割り当てポリシーに従って、前記アプリケーション要求のために割り当てられるリソースの量を決定するように構成され、前記リソース割り当てポリシーは、前記タグと前記アプリケーション要求のために割り当てられる

10

20

30

40

50

前記リソースの量との間の対応関係を有し、前記コンポーネントに対して、前記アプリケーション要求のために割り当てられる前記リソースの量に従って前記アプリケーション要求を処理するように命令するように構成される制御デバイス。

【請求項 8】

前記制御デバイスはバッファを更に有し、前記バッファは少なくとも2つのキューを有し、各キューはリソースの量の範囲に対応し、各キューの優先度は異なり、

前記プロセッサは、前記アプリケーション要求のために割り当てられる前記リソースの量に従って、前記少なくとも2つのキューから前記アプリケーション要求に対応するキューを選択し、前記アプリケーション要求に対応する前記キューに前記アプリケーション要求を記憶するように構成され、

前記コンポーネントは、各キューの前記優先度に従って前記バッファ内の前記キューを別々に処理するように構成される、請求項7に記載の制御デバイス。

【請求項 9】

前記制御デバイスはキャッシュを更に有し、前記キャッシュは前記リソース割り当てポリシーを記憶し、

前記プロセッサは、前記キャッシュから前記リソース割り当てポリシーを獲得するように更に構成される、請求項7に記載の制御デバイス。

【請求項 10】

前記リソース割り当てポリシーは制御テーブルを有し、前記制御テーブルは複数のエントリを有し、前記複数のエントリのうち1つのエントリは、前記タグと前記アプリケーション要求のために割り当てられる前記リソースの量との間の前記対応関係を有し、

前記プロセッサは、クエリ命令を前記キャッシュに送信するように構成され、前記クエリ命令は前記タグを有し、

前記キャッシュは、前記クエリ命令に従って、前記タグに対応するエントリを取得し、前記タグに対応する前記エントリを前記プロセッサに送信するように構成される、請求項9に記載の制御デバイス。

【請求項 11】

前記リソース割り当てポリシーは、ノード管理ソフトウェアを使用することにより前記コンピュータにより定義され、前記ノード管理ソフトウェアは前記コンピュータのメモリに記憶され、

前記プロセッサは、前記ノード管理ソフトウェアから前記リソース割り当てポリシーを獲得するように更に構成される、請求項9又は10に記載の制御デバイス。

【請求項 12】

制御デバイスにおいて使用されるデータ処理方法であり、前記制御デバイスは、コンピュータのコンポーネントに結合され、前記コンポーネントは、チップ上高速相互接続ネットワークと、キャッシュと、メモリと、入出力I/O相互接続ネットワークと、ネットワークアダプタと、ディスプレイとのうちいずれか1つを有し、前記コンピュータは、処理ユニットと、前記制御デバイスとを有する、方法であって、

前記制御デバイスにより、前記処理ユニットによりタグを添付したアプリケーション要求を受信するステップと、

前記制御デバイスにより、前記タグ及び予め記憶されたリソース割り当てポリシーに従って、前記アプリケーション要求のために割り当てられるリソースの量を決定するステップであり、前記リソース割り当てポリシーは、前記タグと前記アプリケーション要求のために割り当てられる前記リソースの量との間の対応関係を有するステップと、

前記制御デバイスにより、前記コンポーネントに対して、前記アプリケーション要求のために割り当てられる前記リソースの量に従って前記アプリケーション要求を処理するように命令するステップと

を有するデータ処理方法。

【請求項 13】

前記制御デバイスはバッファを更に有し、前記バッファは少なくとも2つのキューを有

10

20

30

40

50

し、各キューはリソースの量の範囲に対応し、各キューの優先度は異なり、

前記制御デバイスにより、前記コンポーネントに対して、前記アプリケーション要求のために割り当てられる前記リソースの量に従って前記アプリケーション要求を処理するように命令するステップは、

前記コンポーネントが各キューの前記優先度に従って前記バッファ内の前記キューを別々に処理するように、前記制御デバイスにより、前記アプリケーション要求のために割り当てられる前記リソースの量に従って、前記少なくとも2つのキューから前記アプリケーション要求に対応するキューを選択し、前記アプリケーション要求に対応する前記キューに前記アプリケーション要求を記憶するステップを有する、請求項12に記載の方法。

【請求項14】

10

前記制御デバイスはプロセッサ及びキャッシュを更に有し、前記キャッシュは前記リソース割り当てポリシーを記憶し、

前記データ処理方法は、前記プロセッサにより、前記キャッシュから前記リソース割り当てポリシーを獲得するステップを更に有し、

前記制御デバイスにより、前記タグ及び予め記憶されたリソース割り当てポリシーに従って、前記アプリケーション要求のために割り当てられるリソースの量を決定するステップは、

前記プロセッサにより、前記タグ及び前記予め記憶されたリソース割り当てポリシーに従って、前記アプリケーション要求のために割り当てられる前記リソースの量を決定するステップを有し、

20

前記制御デバイスにより、前記コンポーネントに対して、前記アプリケーション要求のために割り当てられる前記リソースの量に従って前記アプリケーション要求を処理するように命令するステップは、

前記プロセッサにより、前記コンポーネントに対して、前記アプリケーション要求のために割り当てられる前記リソースの量に従って前記アプリケーション要求を処理するように命令するステップを有する、請求項12に記載の方法。

【請求項15】

前記リソース割り当てポリシーは制御テーブルを有し、前記制御テーブルは複数のエントリを有し、前記複数のエントリのうち1つのエントリは、前記タグと前記アプリケーション要求のために割り当てられる前記リソースの量との間の前記対応関係を有し、

30

前記プロセッサにより、前記キャッシュから前記リソース割り当てポリシーを獲得するステップは、

前記プロセッサにより、前記タグを含むクエリ命令を前記キャッシュに送信するステップと、

前記キャッシュにより、前記クエリ命令に従って、前記タグに対応するエントリを取得し、前記タグに対応する前記エントリを前記プロセッサに送信するステップと

を有する、請求項14に記載の方法。

【請求項16】

前記リソース割り当てポリシーは、ノード管理ソフトウェアを使用することにより前記コンピュータにより定義され、前記ノード管理ソフトウェアは前記コンピュータのメモリに記憶され、

40

前記データ処理方法は、前記プロセッサにより、前記ノード管理ソフトウェアから前記リソース割り当てポリシーを獲得するステップを更に有する、請求項14又は15に記載の方法。

【発明の詳細な説明】

【技術分野】

【0001】

[関連出願への相互参照]

この出願は、“COMPUTER, CONTROL DEVICE, AND DATA PROCESSING METHOD”とい

50

う名称で2014年4月30日に中国特許庁に出願された中国特許出願第201410182148.1号及び“COMPUTER, CONTROL DEVICE, AND DATA PROCESSING METHOD”という名称で2014年11月24日に中国特許庁に出願された中国特許出願第201410682375.0号の優先権を主張し、これらの全内容を参照により援用する。

【0002】

〔技術分野〕

本発明は、コンピュータ分野に関し、特にコンピュータ、制御デバイス及びデータ処理方法に関する。

【背景技術】

【0003】

コンピュータ又はサーバの動作効率を改善するために、複数のアプリケーションプログラムは、コンピュータ内にリソース共有を実現し得る。例えば、複数のアプリケーションプログラムは、リソースのためにメモリに同時に問い合わせてもよく、これにより、メモリリソースの利用を改善する。しかし、複数のアプリケーションプログラムは、リソースを共有する場合に相互に干渉し、従って、或る重要なアプリケーションプログラムは、優先的に処理されることができず、これにより、サービス品質に影響を与える。

【発明の概要】

【0004】

本発明の実施例は、アプリケーションプログラムのサービス品質を改善するために使用されるコンピュータ、制御デバイス及びデータ処理方法を提供する。

【0005】

本発明の実施例の第1の態様は、コンピュータを提供し、コンピュータは処理ユニットと制御デバイスとを含み、処理ユニットは、アプリケーション要求にタグを添付し、添付されたアプリケーション要求を制御デバイスに送信するように構成され、制御デバイスは、添付されたアプリケーション要求を受信し、タグ及び予め記憶されたリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定するように構成され、リソース割り当てポリシーは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含み、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令するように更に構成される。

【0006】

第1の態様を参照して、第1の可能な実現方式では、制御デバイスはバッファを更に含み、バッファは少なくとも2つのキューを含み、各キューはリソースの量の範囲に対応し、各キューの優先度は異なり、制御デバイスは、アプリケーション要求のために割り当てられるリソースの量に従って、少なくとも2つのキューからアプリケーション要求に対応するキューを選択し、アプリケーション要求に対応するキューにアプリケーション要求を記憶するように具体的に構成され、コンピュータのコンポーネントは、アプリケーション要求に対応するキューからアプリケーション要求を取得し、アプリケーション要求を実行するように構成される。

【0007】

第1の態様を参照して、第2の可能な実現方式では、制御デバイスはプロセッサとキャッシュとを更に含み、キャッシュはリソース割り当てポリシーを記憶し、プロセッサは、キャッシュからリソース割り当てポリシーを獲得するように更に構成される。

【0008】

第1の態様の第2の可能な実現方式を参照して、第3の可能な実現方式では、リソース割り当てポリシーは制御テーブルを含み、制御テーブルは複数のエントリを含み、複数のエントリのうち1つのエントリは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含み、プロセッサは、クエリ命令をキャッシュに送信するように具体的に構成され、クエリ命令はタグを含み、キャッシュは、クエリ命令に従って、タグに対応するエントリを取得し、タグに対応するエントリを制御デバイスのプロセ

10

20

30

40

50

ッサに送信するように構成される。

【0009】

第1の態様又は第1の態様の第1～第3の可能な実現方式を参照して、本発明の第4の可能な実現方式では、制御デバイスはプログラミングインタフェースを更に含み、プログラミングインタフェースは、リソース割り当てポリシーを変更するように構成される。

【0010】

第1の態様又は第1の態様の第1～第4の可能な実現方式を参照して、第5の実現方式では、コンピュータはメモリを更に含み、メモリはノード管理ソフトウェアを記憶し、処理ユニットは、ノード管理ソフトウェアを使用することによりリソース割り当てポリシーを定義するように更に構成され、制御デバイスは、ノード管理ソフトウェアからリソース
10 割り当てポリシーを獲得し、リソース割り当てポリシーをキャッシュに書き込むように更に構成される。

【0011】

第1の態様の第5の可能な実現方式を参照して、第6の可能な実現方式では、処理ユニットはタグレジスタを更に含み、処理ユニットは、ノード管理ソフトウェアを使用することによりタグを定義し、ノード管理ソフトウェアを使用することによりタグをタグレジスタに書き込むように更に構成され、処理ユニットは、タグレジスタからタグを読み取るように更に構成される。

【0012】

本発明の実施例の第2の態様は、制御デバイスを提供し、制御デバイスはコンピュータ
20 のコンポーネントに配置され、制御デバイスはプロセッサを含み、プロセッサは、タグが添付されたアプリケーション要求を受信し、タグ及び予め記憶されたりソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定するように構成され、リソース割り当てポリシーは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含み、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令するように更に構成される。

【0013】

第2の態様を参照して、第1の可能な実現方式では、制御デバイスはバッファを更に含み、バッファは少なくとも2つのキューを含み、各キューはリソースの量の範囲に対応し
30 、各キューの優先度は異なり、プロセッサは、アプリケーション要求のために割り当てられるリソースの量に従って、少なくとも2つのキューからアプリケーション要求に対応するキューを選択し、アプリケーション要求に対応するキューにアプリケーション要求を記憶するように具体的に構成され、コンピュータのコンポーネントは、アプリケーション要求に対応するキューからアプリケーション要求を取得し、アプリケーション要求を実行するように具体的に構成される。

【0014】

第2の態様を参照して、第2の可能な実現方式では、制御デバイスはキャッシュを更に含み、キャッシュはリソース割り当てポリシーを記憶し、プロセッサは、キャッシュから
40 リソース割り当てポリシーを獲得するように更に構成される。

【0015】

第2の態様の第2の可能な実現方式を参照して、第3の可能な実現方式では、リソース割り当てポリシーは制御テーブルを含み、制御テーブルは複数のエントリを含み、複数のエントリのうち1つのエントリは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含み、プロセッサは、クエリ命令をキャッシュに送信するように具体的に構成され、クエリ命令はタグを含み、キャッシュは、クエリ命令に従って、タグに対応するエントリを取得し、タグに対応するエントリをプロセッサに送信するように構成される。

【0016】

第2の態様又は第2の態様の第1～第3の可能な実現方式を参照して、第4の可能な実
50

現方式では、制御デバイスはプログラミングインタフェースを更に含み、プログラミングインタフェースは、リソース割り当てポリシーを変更するように構成される。

【0017】

第2の態様又は第2の態様の第1～第4の可能な実現方式を参照して、第5の実現方式では、リソース割り当てポリシーは、ノード管理ソフトウェアを使用することによりコンピュータにより定義され、制御デバイスに送信され、ノード管理ソフトウェアはコンピュータのメモリに記憶される。

【0018】

本発明の実施例の第3の態様は、データ処理方法を提供し、この方法は制御デバイスに適用され、制御デバイスはコンピュータのコンポーネントに配置され、この方法は、制御デバイスにより、タグを搬送するアプリケーション要求を受信するステップと、制御デバイスにより、タグ及び予め記憶されたリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定するステップであり、リソース割り当てポリシーは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含むステップと、制御デバイスにより、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令するステップとを含む。

10

【0019】

第3の態様を参照して、第1の可能な実現方式では、制御デバイスはバッファを更に含み、バッファは少なくとも2つのキューを含み、各キューはリソースの量の範囲に対応し、各キューの優先度は異なり、制御デバイスにより、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令するステップは、コンピュータのコンポーネントがアプリケーション要求に対応するキューからアプリケーション要求を取得し、アプリケーション要求を実行するように、制御デバイスにより、アプリケーション要求のために割り当てられるリソースの量に従って、少なくとも2つのキューからアプリケーション要求に対応するキューを選択し、アプリケーション要求に対応するキューにアプリケーション要求を記憶するステップを含む。

20

【0020】

第3の態様を参照して、第2の可能な実現方式では、制御デバイスはプロセッサ及びキャッシュを更に含み、キャッシュはリソース割り当てポリシーを記憶し、この方法は、制御デバイスのプロセッサにより、キャッシュからリソース割り当てポリシーを獲得するステップを更に含む。

30

【0021】

第3の態様の第2の可能な実現方式を参照して、第3の可能な実現方式では、リソース割り当てポリシーは制御テーブルを含み、制御テーブルは複数のエントリを含み、複数のエントリのうち1つのエントリは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含み、制御デバイスのプロセッサにより、キャッシュからリソース割り当てポリシーを獲得するステップは、制御デバイスのプロセッサにより、クエリ命令をキャッシュに送信するステップであり、クエリ命令はタグを含むステップと、キャッシュにより、クエリ命令に従って、タグに対応するエントリを取得し、タグに対応するエントリを制御デバイスのプロセッサに送信するステップとを含む。

40

【0022】

本発明の実施例は、コンピュータを提供し、コンピュータは処理ユニットと制御デバイスとを含む。処理ユニットは、アプリケーション要求にタグを添付し、添付されたアプリケーション要求を制御デバイスに送信する。制御デバイスは、タグ及び予め記憶されたリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定し、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令する。このように、アプリケーション要求を処理する場合、コンピュータのコンポーネントは

50

、アプリケーション要求のために割り当てられるリソースの量に従って処理を実行してもよく、これは、リソースの量が複数のアプリケーション要求により相互に先取りされることを或る程度回避し、これにより、サービス品質を改善する。

【図面の簡単な説明】

【0023】

本発明の実施例又は従来技術における技術的対策を更に明確に説明するために、以下に、実施例又は従来技術を説明するために必要な添付図面を簡単に紹介する。明らかに、以下の説明における添付図面は、本発明の単にいくつかの実施例を示しているに過ぎず、当業者は、創造的取り組みを行うことなく、これらの添付図面から他の図面を導き得る。

【図1】本発明の実施例によるコンピュータのシステムアーキテクチャの図

10

【図2a】本発明の実施例による他のコンピュータのシステムアーキテクチャの図

【図2b】本発明の実施例による制御デバイスの概略構成図

【図3】本発明の実施例による他の制御デバイスの概略構成図

【図4】本発明の実施例による更に他の制御デバイスの概略構成図

【図5】本発明の実施例による更に他の制御デバイスの概略構成図

【図6】本発明の実施例による制御プレーンのネットワークアーキテクチャの概略図

【図7】本発明の実施例によるノード管理ソフトウェアの概略構成図

【図8】本発明の実施例によるデータ処理方法の概略フローチャート

【図9】本発明の実施例による他のデータ処理方法の概略フローチャート

【図10】本発明の実施例による更に他のデータ処理方法の概略フローチャート

20

【発明を実施するための形態】

【0024】

以下に、本発明の実施例における添付図面を参照して、本発明の実施例における技術的対策を明確且つ完全に説明する。明らかに、説明する実施例は、本発明の実施例の全部ではなく、一部である。創造的取り組みを行うことなく本発明の実施例に基づいて当業者により得られる全ての他の実施例は、本発明の保護範囲内に入るものとする。

【0025】

本発明の実施例は、アプリケーションプログラムのサービス品質を改善するために使用されるコンピュータ、制御デバイス及びデータ処理方法を提供する。

【0026】

30

図1は、本発明の実施例によるコンピュータ10のシステムアーキテクチャの概略図である。図1に示すように、コンピュータ10は、コンピュータの複数の処理ユニット11と、複数の制御デバイス66と、複数のコンポーネント33(図1では略して部品と呼ばれる)とを含む。本発明のこの実施例におけるコンピュータのコンポーネントは、リソースが複数のアプリケーションプログラムにより占有され得るコンピュータのコンポーネントを示す。

【0027】

処理ユニット11は、中央処理装置(Central Processing Unit、CPU)上にあり、同じ機能を有するプロセッサコアの1つを示し、読み取り及び書き込みのような様々な動作コマンドを実行するように構成される。

【0028】

40

コンピュータのコンポーネント33は、チップ上高速相互接続ネットワークと、チップ上高速相互接続ネットワークに直接接続されたコンピュータのコンポーネント、例えば、キャッシュ(cacheとも呼ばれる)、メモリ、グラフィック処理ユニット(Graphic Processing Unit、GPU)、及びビデオRAMとを含み、I/O相互接続ネットワークと、I/O相互接続ネットワークに接続されたI/Oデバイス、例えば、磁気ディスク(ハードディスクとも呼ばれる)、ネットワークアダプタ、及びディスプレイとを更に含んでもよい。

【0029】

チップ上高速相互接続ネットワークは、複数の処理ユニット11を接続するコネクタである。チップ上高速相互接続ネットワークはまた、キャッシュ、メモリ、グラフィック処理ユニット、ビデオRAM等に接続される。

50

【0030】

キャッシュについて、キャッシュによりアプリケーションプログラムに割り当てられるリソースは、キャッシュ空間でもよい。メモリについて、メモリによりアプリケーションプログラムに割り当てられるリソースは、メモリ空間でもよい。グラフィック処理ユニットについて、グラフィック処理ユニットによりアプリケーションプログラムに割り当てられるリソースは、ハードウェアアクセラレーションリソースでもよい。ビデオRAMについて、ビデオRAMによりアプリケーションプログラムに割り当てられるリソースは、ビデオRAM空間でもよい。

【0031】

更に、チップ上高速相互接続ネットワークはまた、I/O相互接続ネットワークに接続されてもよい（サウスブリッジとも呼ばれる）。

10

【0032】

I/O相互接続ネットワークは、I/Oデバイスを制御するデバイスである。

【0033】

コンピュータのコンポーネント33は、I/O相互接続ネットワークに直接接続されたI/Oデバイス、例えば、磁気ディスク（ハードディスクとも呼ばれる）、ネットワークアダプタ、及びディスプレイを更に含む。

【0034】

例えば、或る期間に、コンピュータ10は、複数のアプリケーションプログラムを処理してもよく、全てのこれらのアプリケーションプログラムは、コンピュータのコンポーネント（例えば、メモリ）のリソースを占有する必要がある。しかし、メモリ内のリソースは限られているため、或る重要なアプリケーションプログラムは、適時に処理されることができず、従って、サービス品質が影響を受ける。

20

【0035】

従って、本発明のこの実施例では、制御デバイス66は、コンピュータのコンポーネントに配置され、コンポーネントのリソースは、複数のアプリケーションプログラムに申し込まれてもよく、複数のアプリケーションプログラムにより占有されてもよい。制御デバイス66は、アプリケーションプログラムを処理するために、異なる種類のアプリケーションプログラムに従って、異なる量のリソースをアプリケーションプログラムに割り当てるように構成される。ここで、リソースが複数のアプリケーションプログラムにより申し込まれてもよく、複数のアプリケーションプログラムにより占有されてもよいコンピュータのコンポーネントは、チップ上高速相互接続ネットワークと、キャッシュと、メモリと、グラフィック処理ユニットと、ビデオRAMと、I/O相互接続ネットワークとを含むが、これらに限定されない。

30

【0036】

本発明のこの実施例では、制御デバイス66は、コンピュータの複数のコンポーネントの中のコンピュータの1つのコンポーネントのみに配置されてもよく、制御デバイス66は、コンピュータの複数のコンポーネントに配置されてもよく、更に、制御デバイス66は、コンピュータの前述のコンポーネントの全てに配置されてもよい点に留意すべきである。

【0037】

制御デバイス66が異なる種類のアプリケーションプログラムを識別することを可能にするために、アプリケーション要求の種類は、アプリケーション要求（アプリケーションプログラムに対応する要求）が生成されるソースエンドで識別及びタグ付けされる必要がある。従って、アプリケーション要求がコンピュータのコンポーネント上の制御デバイス66に後に送信された場合、制御デバイス66は、タグに従って、異なる種類のアプリケーションプログラムで異なる処理を実行してもよい。本発明のこの実施例では、アプリケーションプログラム及びアプリケーション要求は同じ意味を表す点に留意すべきである。更に、本発明のこの実施例のアプリケーション要求は、コンピュータ内で生成された様々な命令及びコンピュータの外部から受信した様々な命令、例えば、ファイルアクセス要求、ビデオ再生要求、メモリアクセス要求、I/O要求、及び相互接続（Interconnect）要求を含む

40

50

。

【0038】

ここで、アプリケーション要求が生成されるソースエンドは、処理ユニット11又はI/Oデバイス（例えば、ネットワークアダプタ）でもよい。アプリケーション要求がコンピュータ10からのローカルのものである場合、アプリケーション要求が生成されるソースエンドは、処理ユニット11でもよい。アプリケーション要求がコンピュータ10の外部からのものである場合、例えば、インターネットを使用することにより送信されたアプリケーション要求が受信された場合、アプリケーション要求が生成されるソースエンドは、ネットワークアダプタ又は他の入出力デバイスでもよい。

【0039】

アプリケーション要求がコンピュータ10の内部からのものである場合、タグ付けの方式は以下のようなものでもよい。

【0040】

タグレジスタ77（図2aに示す）は、処理ユニット11に配置され、タグレジスタ77は、レジスタ値を記憶する。処理ユニット11がアプリケーション要求を生成した場合、処理ユニット11は、レジスタ値を読み取ることにより、アプリケーション要求にタグを添付し、タグはレジスタ値である。

【0041】

具体的に、タグは、アプリケーション要求についてノード管理ソフトウェア（以下に詳細に説明する）により定義される。ノード管理ソフトウェアは、オペレーティングシステム内のモジュールでもよく、オペレーティングシステムとコンピュータハードウェアとの間の中間ソフトウェア層のモジュール（Hypervisor）でもよく、処理ユニット11で動作する。ノード管理ソフトウェアがアプリケーション要求のタグを定義した後に、オペレーティングシステムは、タグを、アプリケーション要求に対応するプロセスのコンテキストに書き込み、次にアプリケーション要求に対応するプロセスのコンテキストをレジスタに書き込んでよい。

【0042】

任意選択の実現方式は以下の通りである。

【0043】

処理ユニット11自体が複数のレジスタを含んでもよいため、複数のレジスタのうち1つは、タグレジスタ77として設定されてもよく、タグレジスタ77は、アプリケーションプログラムのタグを記憶するように構成される。アプリケーション要求を生成する場合、処理ユニット11は、タグレジスタ77内のレジスタ値を読み取り、レジスタ値をタグとして使用し、レジスタ値をアプリケーション要求に書き込む。タグは、アプリケーションプログラムのID、文字、数字等により表されてもよく、ここでは限定されないことが認識され得る。

【0044】

他の任意選択の実現方式は以下の通りである。

【0045】

新たなレジスタが処理ユニット11に取り付けられ、新たなレジスタは、タグレジスタ77として定義され、タグレジスタ77は、アプリケーションプログラムのタグを記憶するように構成される。以降の処理の方式は、前述の実現方式と同じであり、ここでは再び説明しない。

【0046】

ネットワークアダプタがアプリケーション要求のソースエンドとして機能する場合、実現方式では、ネットワークアダプタ自体は、アプリケーション要求にタグ付けする動作を実行しない。例えば、ネットワークアダプタがアプリケーション要求を受信した場合、アプリケーション要求は、タグを搬送するアプリケーション要求である。すなわち、アプリケーション要求の送信エンドは、アプリケーション要求を送信する前にアプリケーション要求にタグ付けしてもよい。分散システムでは、サーバ（又はコンピュータ）は、ネゴシ

10

20

30

40

50

ーションを用いてアプリケーション要求のタグを決定してもよく、タグサーバが、アプリケーション要求のタグを定義して各サーバに送信するために配置されてもよいことが認識され得る。全ての実現方式等は、本発明のこの実施例の保護範囲内に入るものとする。他の実現方式は以下の通りである。ネットワークアダプタがメッセージパケットを受信した場合、アプリケーション要求は、メッセージパケットを解析することにより取得され、次に、アプリケーション要求がタグ付けされる。この場合、ネットワークアダプタによりアプリケーション要求をタグ付けする方式は、前述の処理ユニット11によりタグ付けする方式と同様であり、ここでは再び説明しない。

【0047】

コンピュータ10内の他のコンポーネント、例えば、I/O相互接続ネットワークもまた、アプリケーション要求にタグを添付してもよく、この実現方式は、処理ユニット11又はネットワークアダプタのものと同様であり、本発明のこの実施例は、タグ付けするコンポーネントに限定を設定しない点に留意すべきである。

【0048】

以下に、処理ユニット11がアプリケーション要求にタグを添付する例を使用することにより、以降の処理手順について説明する。

【0049】

タグがアプリケーション要求に添付された後に、処理ユニット11がアプリケーション要求を制御デバイス66に送信した場合、アプリケーション要求のタグが搬送される。

【0050】

具体的に、制御デバイス66は、添付されたアプリケーション要求を受信し、タグ及び予め記憶されたりソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定するように構成され、リソース割り当てポリシーは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含み、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令するように更に構成される。

【0051】

本発明の実施例は、コンピュータを提供し、コンピュータは処理ユニットと制御デバイスとを含む。処理ユニットは、アプリケーション要求にタグを添付し、添付されたアプリケーション要求を制御デバイスに送信する。制御デバイスは、タグ及び予め記憶されたりソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定し、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令する。このように、アプリケーション要求を処理する場合、コンピュータのコンポーネントは、アプリケーション要求のために割り当てられるリソースの量に従って処理を実行してもよく、これは、リソースの量が複数のアプリケーション要求により相互に先取りされることを或る程度回避し、これにより、サービス品質を改善する。

【0052】

以下に、制御デバイス66の構成及び機能について主に説明する。

【0053】

制御デバイス66は、コンピュータのコンポーネントのうちいずれかのコンポーネントにおける、コンピュータ10内のデバイスを示す。コンピュータの或るコンポーネントがコントローラを含む場合（例えば、メモリがメモリコントローラを含む場合、又はネットワークアダプタがネットワークアダプタコントローラを含む場合）、制御デバイス66は、コントローラに組み込まれた制御デバイス、又は元のコントローラに接続された新たに取り付けられた制御デバイスでもよい。コンピュータの或るコンポーネントがコントローラを含まない場合、制御デバイス66は、コンピュータのコンポーネントに接続された新たに取り付けられたコントローラ又は制御デバイスでもよい。

【0054】

図 2 b に示すように、制御デバイス 66 はプロセッサ 600a を含む。

【 0 0 5 5 】

制御デバイス 66 は、タグが添付されたアプリケーション要求を受信し、タグ及び予め記憶されたリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定するように構成され、リソース割り当てポリシーは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含み、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令するように更に構成される。

【 0 0 5 6 】

更に、制御デバイス 66 は、バッファ (buffer と呼ばれる) 600b を更に含んでもよい。

10

【 0 0 5 7 】

例えば、プロセッサ 600a は、添付されたアプリケーション要求をバッファ 600b に記憶し、バッファ 600b からタグを読み取り、タグ及び予め記憶されたリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定するように構成され、リソース割り当てポリシーは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含み、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令するように構成される。

【 0 0 5 8 】

バッファ 600b はまた、プロセッサ 600a 内のレジスタでもよい点に留意すべきである。この場合、その処理方式は以下のようでもよい。プロセッサ 600a は、添付されたアプリケーション要求をプロセッサ 600a のレジスタに記憶し、レジスタからタグを読み取り、タグ及び予め記憶されたリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定するように構成され、リソース割り当てポリシーは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含み、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令するように構成される。

20

【 0 0 5 9 】

例えば、図 3 に示すように、任意選択の実現方式は以下の通りである。制御デバイス 66 は、プロセッサ 600i と、バッファ (buffer と呼ばれる) 600b と、キュー 600c とを含んでもよい。

30

【 0 0 6 0 】

プロセッサ 600i は、フィールドプログラマブルゲートアレイ (Field-Programmable Gate Array, FPGA) 又は他のプログラム可能デバイスでもよい。リソース割り当てポリシーは、プロセッサ 600i に内蔵され、リソース割り当てポリシーは、制御テーブル (表 1 に示す) でもよく、制御テーブルは編集可能である。

【表 1】

表 1

タグ	属性1	...	属性i	状態1	...	状態i
タグ1	値の範囲	...	値の範囲	値	...	値
...						
タグk	値の範囲	...	値の範囲	値	...	値

10

【0061】

具体的に、表1の各エントリは、1つのタグに対応する。更に、制御テーブルの各エントリは、複数の“属性”を含み、“属性”は、タグに対応するアプリケーション要求のために割り当てられるリソースの量を表す。複数の種類のリソースの量が存在してもよく、例えば、ターゲットサービス品質、IPC (Instruction per Cycle)、応答時間、最大許容度等が含まれてもよい。“属性”の値の範囲は、ユーザにより設定されてもよい。例えば、値の範囲は、30%未満ではない範囲又は80%より大きくない範囲として定義されてもよい。更に、各エントリは、複数の“状態”を更に含み、“状態”は、タグに対応するアプリケーション要求により現在消費されているリソースの量を表し、“状態”の値は、リアルタイムで監視されて更新されてもよい。

20

【0062】

更に、リソース割り当てポリシーはまた、ファームウェアコードでもよく、FPGAに内蔵される。本発明のこの実施例はリソース割り当てポリシーを記憶する方式に限定を設定しない。

30

【0063】

バッファ600bは、一時バッファである。制御デバイス66がタグを搬送するアプリケーション要求を受信した場合、アプリケーション要求は、まず、一時記憶のためにバッファ600bに配置される。

【0064】

キュー600cもまた、一時バッファであり、これは、バッファ600bと同じ一時バッファに位置してもよく、一時バッファと独立して機能するためにバッファ600bから分離されてもよい。キュー600cは、プロセッサ600iにより処理されるアプリケーション要求を記憶するように構成される。キュー600cは、複数のキューを含んでもよく、異なるキューは、バッファ600b内の異なるアドレスセグメントに対応している。異なるキューは、異なる優先度を有し、優先度は、コンピュータのコンポーネントによりキュー内のアプリケーション要求を実行するための異なる順序で表される。これは、異なる量のリソースが異なるキューのために割り当てられることを意味する。

40

【0065】

更に、制御デバイス66は、プログラミングインタフェース600dを更に含んでもよい。

【0066】

プログラミングインタフェース600dは、アドレス空間マッピング機構を実現するように構成され、プロセッサ600iに内蔵された制御テーブルをコンピュータ10の物理アドレス空間にマッピングすることができる。ノード管理ソフトウェアは、制御テーブルを編集するために、コンピュータ10の物理アドレス空間にアクセスしてもよい。例えば、プログラミ

50

ングインタフェース600dは、制御テーブルに記憶されたエントリを追加、変更又は削除するために使用される様々な機能を提供してもよい。更に、ノード管理ソフトウェアが全てのアプリケーション要求の“状態”の値で統計を収集した後にリソース割り当てポリシーを更に調整するように、プロセッサ600iは、プロセッサ600iの制御テーブルの各エントリ内の全ての“状態”の値をノード管理ソフトウェアに更に提供してもよい。

【0067】

例えば、プロセッサ600iは、バッファ600bからタグを搬送するアプリケーション要求を獲得し、アプリケーション要求の“属性”を取得するために、タグに従って表1の対応するエントリに問い合わせてもよい。アプリケーション要求の“属性”はアプリケーション要求のために割り当てられるリソースの量を示すため、プロセッサ600iは、アプリケーション要求のために割り当てられるリソースの量に従って、少なくとも2つのキューからアプリケーション要求に対応するキューを選択し、アプリケーション要求に対応するキューに配置してもよい。

10

【0068】

アプリケーション要求に対応するキューに配置した後に、プロセッサ600iは、コンピュータの対応するコンポーネントに対して、アプリケーション要求で処理を実行するように命令してもよい。ここで、“コンピュータの対応するコンポーネント”は、制御デバイスが属するコンピュータのコンポーネントを示す。例えば、制御デバイスがキャッシュ上の制御デバイスを示す場合、“コンピュータの対応するコンポーネント”は、ここではキャッシュを示す。

20

【0069】

ここで、“コンピュータの対応するコンポーネントに対して、アプリケーション要求で処理を実行するように命令すること”は、プロセッサ600iが対応するキューからアプリケーション要求を取り出し、アプリケーション要求をコンピュータの対応するコンポーネントに送信することでもよく、コンピュータの対応するコンポーネントが対応するキューからアプリケーション要求を取得することでもよい。

【0070】

更に、制御デバイス66は、データ転送器600jを更に含んでもよい。異なるアプリケーション要求を異なるキュー600cに配置した後に、プロセッサ600iは、キューからアプリケーション要求を取り出した後に、アプリケーション要求をデータ転送器600jに送信してもよい。データ転送器600jは、アプリケーション要求をコンピュータの対応するコンポーネントに転送するように構成される。すなわち、ここで、コンピュータの対応するコンポーネントは、データ転送器600jを使用することにより、対応するキューからアプリケーション要求を取得してもよい。

30

【0071】

更に、アプリケーション要求に対応するキュー600cに配置する前に、プロセッサ600iは、アプリケーション要求に圧縮及び暗号化のような或る前処理動作を実行し、次に、前処理動作を受けたアプリケーション要求に対応するキュー600cに配置してもよい。

【0072】

図4に示すように、制御デバイス66について他の任意選択の実現方式は以下の通りである。

40

【0073】

制御デバイス66は、バッファ（bufferとも呼ばれる）600bと、キュー600cと、マイクロプロセッサ600eと、キャッシュ（cacheとも呼ばれる）600fとを含む。

【0074】

バッファ（bufferとも呼ばれる）600b及びキュー600cは、図3に示すバッファ及びキューと同じであり、ここでは再び説明しない。

【0075】

マイクロプロセッサ600eは、CPU又は機能がCPUのもと同様である他のコントローラでもよい。マイクロプロセッサ600eと図3に示すプロセッサ600iとの間の差は、プロセッサ

50

600iがプログラム可能デバイスとして機能し、リソース割り当てポリシーがプロセッサ600iに内蔵され、リソース割り当てポリシーが編集可能であり、マイクロプロセッサ600eがCPUの機能を実行するが、制御テーブルがマイクロプロセッサ600eに内蔵できない点にある。従って、図4に示す制御デバイス66は、キャッシュ600fを更に含む。キャッシュ600fは、リソース割り当てポリシーを記憶し、リソース割り当てポリシーは、機能が制御テーブルのものと同様であるプログラムコードを示す。

【0076】

例えば、制御デバイス66がタグを搬送するアプリケーション要求を受信した場合、アプリケーション要求は、まずバッファ600bに配置される。マイクロプロセッサ600eは、バッファ600bに記憶されたアプリケーション要求のキューから、タグを搬送するアプリケーション要求を獲得し、リソース割り当てポリシーをキャッシュ600fからバッファ600bに読み取り、タグ及びリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定し、少なくとも2つのキューからアプリケーション要求に対応するキューを選択し、アプリケーション要求に対応するキューに配置してもよい。次に、マイクロプロセッサ600eは、対応するキューからアプリケーション要求を取り出し、アプリケーション要求をコンピュータの対応するコンポーネントに送信する。

10

【0077】

或いは、制御デバイス66は、データ転送器600jを更に含んでもよい。異なるアプリケーション要求を異なるキュー600cに配置した後に、マイクロプロセッサ600eは、キューからアプリケーション要求を取り出した後に、アプリケーション要求をデータ転送器600jに送信してもよく、データ転送器600jは、アプリケーション要求をコンピュータの対応するコンポーネントに転送するように構成される。以降の処理の方式は、図3に示す実現方式と同じであり、ここでは再び説明しない。

20

【0078】

更に、図4に示す制御デバイス66は、キャッシュ600fに記憶されたリソース割り当てポリシーを変更するために使用される制御ロジック(図4に図示せず)を更に含んでもよい。

【0079】

同様に、アプリケーション要求に対応するキュー600cに配置する前に、マイクロプロセッサ600eは、アプリケーション要求に圧縮及び暗号化のような或る前処理動作を実行し、次に、前処理動作を受けたアプリケーション要求に対応するキュー600cに配置してもよい。

30

【0080】

マイクロプロセッサ600e内にbufferが存在する場合、マイクロプロセッサ600eはまた、タグを搬送するアプリケーション要求及びキャッシュ600fに記憶されたリソース割り当てポリシーをマイクロプロセッサ600eのbufferに読み取り、マイクロプロセッサ600eのbuffer内のアプリケーション要求を処理し、処理結果に従ってアプリケーション要求に対応するキュー600cに配置してもよいことが認識され得る。

【0081】

制御デバイス66について、更に他の任意選択の実現方式は以下の通りである。

40

【0082】

図5に示すように、制御デバイス66は、バッファ600bと、比較制御ロジック600gと、キャッシュ600fと、キュー600cとを含んでもよい。

【0083】

ここで、比較制御ロジック600gは、特定用途向け集積回路(Application Specific Integrated Circuits、ASIC)又は他の集積回路でもよい。

【0084】

バッファ600bは、前述のバッファと一致する。

【0085】

キャッシュ600fは、制御テーブル(表1)を記憶する。

50

【 0 0 8 6 】

制御デバイス66がタグを搬送するアプリケーション要求を受信した場合、アプリケーション要求は、まず対応するキューに配置される。キューは、バッファ600b内のキャッシュ空間の一部又は独立したバッファでもよい。比較制御ロジック600gは、キューからバッファ600b（又は比較制御ロジック600gのbuffer）にアプリケーション要求を読み取り、タグに対応するエントリを返すようにキャッシュ600fに要求するために、アプリケーション要求のタグに従って読み取り命令をキャッシュ600fに送信する。エントリの内容は、バッファ600b（又は比較制御ロジック600gのbuffer）にロードされ、比較制御ロジック600gは、アプリケーション要求をキュー600cに配置するために、バッファ600b（又は比較制御ロジック600gのbuffer）内のエントリの内容に従って、少なくとも2つのキューから対応するキューを選択する。次に、比較制御ロジック600gは、対応するキューからアプリケーション要求を取り出し、アプリケーション要求をコンピュータの対応するコンポーネントに送信する。

10

【 0 0 8 7 】

或いは、制御デバイス66は、データ転送器600jを更に含んでもよい。異なるアプリケーション要求を異なるキュー600cに配置した後に、比較制御ロジック600gは、キューからアプリケーション要求を取り出した後に、アプリケーション要求をデータ転送器600jに送信してもよく、データ転送器600jは、アプリケーション要求をコンピュータの対応するコンポーネントに転送するように構成される。

20

【 0 0 8 8 】

同様に、比較制御ロジック600gは、バッファ600bにおいて、アプリケーション要求に圧縮及び暗号化のような或る前処理動作を更に実行してもよい。

【 0 0 8 9 】

更に、制御デバイス66は、キャッシュ600fに記憶された制御テーブルを編集するように構成されたプログラミングインタフェース600dを更に含んでもよい。プログラミングインタフェース600dの具体的な機能について、図3に示す実施例におけるプログラミングインタフェース600dの説明に参照が行われてもよい。

【 0 0 9 0 】

コンピュータ10のコンポーネントの制御デバイス66は、完全には同じでなくてもよい点に留意すべきである。具体的に、制御デバイス66により記憶されるリソース割り当てポリシーは、完全には同じでなくてもよい。例えば、同じアプリケーション要求について、アプリケーション要求がメモリにアクセスする必要がある場合、メモリによりアプリケーション要求のために割り当てられるリソースの量は80%に達してもよく、アプリケーション要求がI/Oデバイスを使用することにより出力される必要がある場合、I/O相互接続ネットワークによりアプリケーション要求のために割り当てられるリソースの量は70%のみでもよい。

30

【 0 0 9 1 】

本発明のこの実施例により提供される制御デバイスを使用することにより、アプリケーション要求のために割り当てられるリソースの量は、アプリケーション要求のタグ及び予め記憶されたりソース割り当てポリシーに従って決定されてもよく、コンピュータのコンポーネントは、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令される。このように、アプリケーション要求を処理する場合、コンピュータのコンポーネントは、アプリケーション要求のために割り当てられるリソースの量に従って処理を実行してもよく、これは、リソースの量が複数のアプリケーション要求により相互に先取りされることを或る程度回避し、これにより、サービス品質を改善する。

40

【 0 0 9 2 】

本発明のこの実施例では、コンピュータのコンポーネント上の制御デバイス66は、図1又は図2aにおいて点線で示すように、全ての制御デバイス66を接続するために使用され

50

る制御デバイスネットワークを形成してもよい。各制御デバイスは、1つの物理アクセスポイントを含み、物理アクセスポイントを使用することにより制御デバイスネットワークにアクセスする。制御デバイスネットワークは、ペリフェラル・コンポーネント・インターコネクト・エクスプレス(Peripheral Component Interconnect Express、PCIe)プロトコル又は他のプロトコルを使用することによりデータを伝送してもよく、プロトコルの種類は、本発明のこの実施例では限定されない。

【0093】

本発明のこの実施例では、ノード管理ソフトウェア(図1に図示せず)が更に含まれてもよく、ノード管理ソフトウェアは、オペレーティングシステム内のモジュールでもよく、オペレーティングシステムとコンピュータハードウェアとの間の中間ソフトウェア層のモジュール(Hypervisor)でもよく、1つ又は複数の処理ユニット11で動作する。ノード管理ソフトウェアは、制御デバイスネットワークを通じて全ての制御デバイス66を管理するように構成され、例えば、制御デバイス66での初期化動作を実行すること、制御デバイス66の“状態”値を収集すること、収集された“状態”値に従ってリソース割り当てポリシーを決定又は調整すること、リソース割り当てポリシーを制御デバイス66に送信すること等を行うように構成される。

【0094】

任意選択で、図6に示すように、実現方式は以下の通りである。専用ネットワークがコンピュータ10内に確立され、全ての制御デバイス66がルートルータを使用することにより接続され、全ての制御デバイス66の物理アクセスポイントは、専用ネットワークに接続される。専用ネットワークは、通信プロトコルを提供してもよく、通信プロトコルは、制御デバイス66にアクセスするためのメッセージパケットのフォーマットを定義する役目をする。例えば、メッセージパケットは、制御デバイス番号又は制御デバイスIDと、制御デバイスコマンド(例えば、リソース割り当てポリシーの追加、リソース割り当てポリシーの削除、又はリソース割り当てポリシーの変更)と、制御デバイスコマンドパラメータとを含んでもよいが、これに限定されない。更に、制御デバイス66がコンピュータ10の外部デバイス又はノード管理ソフトウェアと通信することを可能にするために、メッセージパケットはまた、PCIeプロトコル又は他のプロトコルを使用することによりカプセル化された後に伝送されてもよい。

【0095】

任意選択で、他の実現方式は以下の通りである。各制御デバイス66は、アドレス空間マッピング方式でノード管理ソフトウェアとデータ交換を実行する。具体的に、各制御デバイス66は、制御デバイス66の制御テーブル又はリソース割り当てポリシーをコンピュータ10の物理アドレス空間にマッピングする。ノード管理ソフトウェアは、制御テーブル又はリソース割り当てポリシーの編集を実現するために、アドレス空間にアクセスしてもよい。

【0096】

図7に示すように、ノード管理ソフトウェアは、制御デバイスドライバモジュール701と、監視及び管理モジュール702と、ユーザプログラミングインタフェース703とを含んでもよい。

【0097】

制御デバイスドライバモジュール701は、コンピュータのコンポーネントの新たな制御デバイス66をスキャン及び識別し、制御デバイス66を初期化し、リソース割り当てポリシーを制御デバイス66に送信するように構成される。更に、制御デバイスドライバモジュール701は、リソース割り当てポリシーを追加、変更又は削除するように更に構成される。

【0098】

監視及び管理モジュール702は、各制御デバイス66の収集された“状態”値を記憶し、各制御デバイス66の収集された“状態”値で相関分析を実行し、ユーザ要求と組み合わせるリソース割り当てポリシーを決定するように構成される。

【0099】

10

20

30

40

50

ユーザプログラミングインタフェース703は、他のソフトウェア又はアプリケーションプログラムがアプリケーションプログラミングインタフェース(Application Programming Interface、API)を使用することにより制御デバイス66のプログラミングを実現し得るように、APIを提供するように構成される。例えば、APIは少なくとも以下のインタフェース、すなわち、初期化コマンド、リソース割り当てポリシーを追加するコマンド、リソース割り当てポリシーを変更するコマンド、リソース割り当てポリシーを削除するコマンド等を含む。

【0100】

図2b、図3、図4又は図5に示す、本発明の実施例における制御デバイスは、本発明の実施例に適用可能な例に過ぎず、本発明の適用に具体的な制限を設定することを意図するものではない。例えば、制御デバイスはまた、特定用途向け集積回路でもよい。制御デバイスがどの形式にあったとしても、制御デバイスは、コンピュータ内で制御デバイスの機能を実現する。本発明の実施例に記載のコンピュータは、パーソナルコンピュータ、サーバ、移動電話、又はパームトップコンピュータでもよく、本発明は、コンピュータの具体的な実現方式に限定を設定しない。この出願の文献は、他のシステムの実施例又は適用シナリオを1つずつ説明しない。

10

【0101】

以下に、本発明の実施例におけるデータ処理を実現するために、制御デバイスがコンピュータ内に配置される手順について説明する。本発明のこの実施例におけるデータ処理方法は、図2b、図3、図4、又は図5に示す制御デバイスに実現されてもよい。図8に示すように、この方法は以下を含む。

20

【0102】

ステップS101：制御デバイスは、タグを搬送するアプリケーション要求を受信する。

【0103】

タグを搬送するアプリケーション要求は、図1又は図2aに示す処理ユニット11からのものでもよく、ネットワークアダプタからのものでもよい。タグを搬送するアプリケーション要求が処理ユニット11からのものである場合、処理ユニット11は、アプリケーション要求を受信又は生成する場合に、アプリケーション要求にタグを添付する必要がある。具体的に、処理ユニット11によりアプリケーション要求にタグを添付する方式について、図2aに示す実施例に参照が行われてもよく、詳細についてここでは再び説明しない。

30

【0104】

タグを搬送するアプリケーション要求がネットワークアダプタからのものである場合、1つの場合は、ネットワークアダプタがアプリケーション要求を受信した場合、アプリケーション要求がタグを搬送するアプリケーション要求であることである。他の場合は、ネットワークアダプタがメッセージパケットを受信した場合、アプリケーション要求は、メッセージパケットを解析することにより取得され、次にアプリケーション要求がタグ付けされることである。

【0105】

更に、ステップS101の前に、各制御デバイスが動作状態になるように、オペレーティングシステム又はHypervisorは、ノード管理ソフトウェアを使用することにより、コンピュータ内の各制御デバイス66で初期化動作を実行してもよい。制御デバイス66が初期化動作を受けた後に、ノード管理ソフトウェアは、制御デバイスネットワークを通じてリソース割り当てポリシーを各制御デバイス66に送信する。

40

【0106】

ステップS102：制御デバイスは、タグ及び予め記憶されたリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定し、リソース割り当てポリシーは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含む。

【0107】

具体的に、制御デバイスは、タグを搬送するアプリケーション要求を制御デバイスの第

50

1のバッファ(図2bに示すバッファ600b)に書き込み、第1のバッファからタグを読み取る。

【0108】

リソース割り当てポリシーは、ノード管理ソフトウェアを使用することにより制御デバイス66に送信され、処理ユニット11によりアプリケーション要求に添付されたタグもまた、ノード管理ソフトウェアを使用することにより定義される。従って、アプリケーション要求で搬送されたタグは、リソース割り当てポリシー内のタグに関連付けられてもよく、制御デバイスは、タグ及び予め記憶されたリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定してもよい。

【0109】

ここでのリソースの量は、制御デバイスが位置するコンピュータのコンポーネントによりアプリケーション要求のために割り当てられるリソースの量でもよく、パーセント値でもよく、優先度情報等を更に含んでもよい。例えば、制御デバイスが位置するコンピュータのコンポーネントがメモリである場合、リソースの量は、メモリ空間のサイズでもよく、メモリ空間のパーセント値(例えば、80%)でもよく、他の情報(表1の“属性”の説明に参照が行われてもよい)でもよい。本発明のこの実施例は、リソースの量の表現形式に限定を設定せず、コンポーネントによりアプリケーション要求を処理する優先度又は速度を表すという条件で、いずれのものも本発明のこの実施例の保護範囲内に入るものとする。

【0110】

更に、ここでのリソース割り当てポリシーは、制御デバイスのプロセッサに内蔵されてもよく(図3に示す実現方式に参照が行われる)、制御デバイスのキャッシュに記憶されてもよい(図3に示す実現方式に参照が行われる)。

【0111】

ステップS103:制御デバイスは、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令する。

【0112】

コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令することは、アプリケーション要求のために割り当てられるリソースの量及びアプリケーション要求をコンポーネントに送信することでもよく、アプリケーション要求をコンポーネントに送信し、アプリケーション要求を処理する方式をコンポーネントに通知することでもよい。

【0113】

本発明のこの実施例では、制御デバイスは、アプリケーション要求で搬送されるタグ及びタグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係に従って、アプリケーション要求のために割り当てられるリソースの量を決定し、次に、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令してもよい。従って、異なる量のリソースが異なるアプリケーション要求に割り当てられてもよく、これにより、サービス品質を改善する。

【0114】

任意選択で、ステップS103の後にアプリケーション要求の処理が完了していない場合、コンピュータの他のコンポーネントからリソースを申し込み、アプリケーション要求を処理することが依然として必要であり、処理されたアプリケーション要求は、処理のためにコンピュータの他のコンポーネント上の制御デバイスに転送されてもよく、この処理方式はステップS101~ステップS103と同様である。コンピュータの他のコンポーネント上の制御デバイスに転送されるアプリケーション要求もタグを搬送する点に留意すべきである。

【0115】

以下に、本発明の実施例におけるデータ処理方法の他の実現方式について説明する。図

10

20

30

40

50

9 に示すように、この方法は以下を含む。

【0116】

ステップS201：ステップS101と同じである。

【0117】

ステップS202：制御デバイスは、キャッシュ（略してキャッシュと呼ばれてもよい）からリソース割り当てポリシーを取得する。

【0118】

リソース割り当てポリシーがソフトウェアコードである場合、制御デバイスは、リソース割り当てポリシーを制御デバイスのプロセッサのバッファにロードしてもよい。リソース割り当てポリシーが表1に示す制御テーブルである場合、制御デバイスは、クエリ命令をキャッシュに送信してもよく、クエリ命令は、タグを含み、キャッシュは、タグに従って対応するエントリを求めて制御テーブルを検索し、エントリを制御デバイスに戻す。制御デバイスは、エントリを制御デバイスのプロセッサのバッファにロードする。具体的に、エントリは、タグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係を含む。

10

【0119】

ステップS203：制御デバイスは、タグ及び予め記憶されたリソース割り当てポリシーに従って、アプリケーション要求のために割り当てられるリソースの量を決定する。

【0120】

ステップS204：ステップS103と同じである。

20

【0121】

本発明のこの実施例では、制御デバイスは、アプリケーション要求で搬送されるタグ及びタグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係に従って、アプリケーション要求のために割り当てられるリソースの量を決定し、次に、コンピュータのコンポーネントに対して、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求を処理するように命令してもよい。従って、異なる量のリソースが異なるアプリケーション要求に割り当てられてもよく、これにより、サービス品質を改善する。

【0122】

以下に、本発明の実施例におけるデータ処理方法の他の実現方式について説明する。図10に示すように、この方法は以下を含む。

30

【0123】

ステップS301：ステップS101と同じである。

【0124】

ステップS302：ステップS102と同じであり、ステップS202及びS203と同じである。

【0125】

ステップS303：制御デバイスは、アプリケーション要求のために割り当てられるリソースの量に従って、少なくとも2つのキューからアプリケーション要求に対応するキューを選択し、アプリケーション要求に対応するキューにアプリケーション要求を記憶する。

【0126】

具体的に、制御デバイスは、第2のバッファを含んでもよく、キューは、第2のバッファに記憶される（図3、図4又は図5に示す実現方式に参照が行われてもよい）。第2のバッファは、少なくとも2つのキューを含み、各キューは、リソースの量の範囲に対応し、各キューの優先度は異なる。

40

【0127】

ステップS304：制御デバイスのデータ転送器（例えば、図3～図5におけるデータ転送器600j）は、アプリケーション要求に対応するキューからアプリケーション要求を取得し、アプリケーション要求をコンピュータのコンポーネントに転送する。

【0128】

ステップS305：コンピュータのコンポーネントは、アプリケーション要求を取得して実

50

行する。

【0129】

ステップS304は任意選択のステップであり、コンピュータのコンポーネントはまた、対応するキューからアプリケーション要求を直接取得してもよい点に留意すべきである。

【0130】

例えば、第2のバッファは、高優先度キュー、中優先度キュー及び低優先度キューである3つのキューを含んでもよい。高優先度キューに対応するリソースの量のパーセント値は、70%~80%である。アプリケーション要求のために割り当てられるリソースの量が76%である場合、アプリケーション要求は、高優先度キューに配置される。高優先度キューは、優先処理順序又は比較的高い処理速度を意味する。コンピュータのコンポーネントは、高優先度キューからアプリケーション要求を優先的に取得し、アプリケーション要求を実行してもよい。任意選択で、制御デバイスは、どのキューにアプリケーション要求が配置されるかを検討するために、他のファクタ、例えば、アプリケーション要求により現在使用されているリソースの量を更に参照してもよい。この出願の文献では、適用シナリオを1つずつ説明しない。

10

【0131】

各キューは、複数の処理されるべきアプリケーション要求を含んでもよいことが認識され得る。各キュー内の複数の処理されるべきアプリケーション要求について、コンピュータのコンポーネントは、先入れ先出しの原則又は他の原則に従って処理を実行してもよい。

20

【0132】

更に、制御デバイスがアプリケーション要求に対応するキューに配置した後に、制御デバイスはまた、キューからアプリケーション要求を順に取り出し（キューに複数のアプリケーション要求が存在する場合）、処理のためにアプリケーション要求をコンピュータのコンポーネントに送信してもよい。

【0133】

本発明のこの実施例では、制御デバイスは、アプリケーション要求で搬送されるタグ及びタグとアプリケーション要求のために割り当てられるリソースの量との間の対応関係に従って、アプリケーション要求のために割り当てられるリソースの量を決定し、アプリケーション要求のために割り当てられるリソースの量に従ってアプリケーション要求に対応するキューに配置し、次に、アプリケーション要求に対応するキューからアプリケーション要求を取得し、アプリケーション要求をデータ転送器に送信してもよい。データ転送器は、実行のためにアプリケーション要求をコンピュータのコンポーネントに転送し、これにより、サービス品質を改善する。

30

【0134】

任意選択で、図8、図9又は図10に示すデータ処理方法の実現方式では、以下のステップが更に含まれてもよい。

【0135】

アプリケーション要求の実行が完了した後に、コンピュータのコンポーネントは、アプリケーション要求が処理されたことを説明するためのメッセージをオペレーティングシステムにフィードバックしてもよい。この場合、リソース再生(resource reclamation)メッセージが、ノード管理ソフトウェアを使用することにより制御デバイスに送信されてもよく、リソース再生メッセージは、制御デバイス内のリソース割り当てポリシーを削除するために使用される。

40

【0136】

以下に、ビデオ再生のためのアプリケーション要求が処理される例を使用することにより、図8~図10における処理手順について更に説明する。

【0137】

ステップ1: ユーザは、ローカルコンピュータ内のビデオファイルをクリックする。

【0138】

50

ステップ2：CPUは、ビデオファイルを読み取るために、メモリアクセス要求を生成する。

【0139】

ステップ3：CPUは、メモリアクセス要求にタグ付けし、タグを搬送するメモリアクセス要求をメモリ上の制御デバイスに送信する。

【0140】

ステップ4：メモリ上の制御デバイスは、タグに従って、メモリアクセス要求のために割り当てられるメモリ空間を決定し、割り当てられるメモリ空間に従ってメモリアクセス要求を対応するキューに更に配置する。

【0141】

ステップ5：メモリ上の制御デバイスは、キューからメモリアクセス要求を取り出し、メモリアクセス要求を実行するために、メモリアクセス要求をメモリに送信する。

【0142】

ステップ6：メモリは、ビデオファイルをロードした後に、応答をCPUに送信する。

【0143】

ステップ7：CPUは、メモリからビデオファイルを取得し、ビデオファイルにおいてハードウェアデコードを実行するようにGPUに要求するために、ハードウェアアクセラレーション要求をGPUに送信し、ハードウェアアクセラレーション要求は、ビデオファイル及びタグを搬送する。

【0144】

ステップ8：GPU上の制御デバイスは、タグに従って、ビデオファイルのために割り当てられるハードウェアアクセラレーションリソースの量を決定し、次に、割り当てられるハードウェアアクセラレーションリソースの量に従って、ハードウェアアクセラレーション要求を対応するキューに配置する。

【0145】

ステップ9：GPU上の制御デバイスは、キューからハードウェアアクセラレーション要求を取り出し、ハードウェアアクセラレーション要求をGPUに送信する。

【0146】

ステップ10：GPUは、ハードウェアアクセラレーション要求に従ってビデオファイルにおいてハードウェアアクセラレーション処理を実行する。

【0147】

ステップ11：処理が完了した後に、ビデオファイルにおいて他の処理を実行する必要がない場合、GPUは、出力要求をサウスブリッジ（例えば、図1又は図2aに示すI/O制御ネットワーク）に送信してもよく、出力要求は、ハードウェアアクセラレーション処理を受けたビデオファイル及びタグを含む。

【0148】

ステップ12：サウスブリッジ上の制御デバイスは、タグに従って、ビデオファイルのために割り当てられた帯域幅を決定し、割り当てられた帯域幅に従って、出力要求を対応するキューに配置する。

【0149】

ステップ13：サウスブリッジは、キューからビデオファイルを取り出し、ビデオファイルをディスプレイに送信する。

【0150】

ステップ14：ディスプレイは、ビデオファイルを表示する。

【0151】

当業者は、この明細書に開示された実施例に記載の例と組み合わせて、ユニット及びアルゴリズムのステップが電子ハードウェア、コンピュータソフトウェア、又はこれらの組み合わせにより実現されてもよいことを認識し得る。ハードウェアとソフトウェアとの間の互換性を明確に説明するために、前述では、機能に従って各例の構成及びステップを一般的に説明した。機能がハードウェアにより実行されるかソフトウェアにより実行される

10

20

30

40

50

かは、技術的対策の特定の用途及び設計上の制約条件に依存する。当業者は、特定の用途毎に記載の機能を実現するために異なる方法を使用してもよいが、この実現は、本発明の範囲を超えるものとして考えられるべきではない。

【0152】

便宜的且つ簡潔な説明の目的で、前述のシステム、装置及びユニットの詳細な動作処理について、前述の方法の実施例の対応する処理に参照が行われてもよく、詳細をここでは再び説明しないことが、当業者により明確に認識され得る。

【0153】

この出願で提供される複数の実施例では、開示のシステム、装置及び方法は他の方式で実現されてもよいことが認識されるべきである。例えば、記載の装置の実施例は、単なる例示である。例えば、ユニットの分割は、単に論理的な機能分割であり、実際の実現では他の分割でもよい。例えば、複数のユニット又はコンポーネントは結合されてもよく、他のシステムに統合されてもよく、いくつかの機能が無視されてもよく実行されなくてもよい。更に、表示又は説明した相互結合若しくは直接結合又は通信接続は、いくつかのインタフェースを通じて実現されてもよい。装置又はユニットの間の間接結合又は通信接続は、電氣的、機械的又は他の形式で実現されてもよい。

10

【0154】

別々の部分として記載したユニットは、物理的に別々でもよく別々でなくてもよい。ユニットとして表示された部分は、物理的なユニットでもよく物理的なユニットでなくてもよく、1つの位置に存在してもよく、複数のネットワークユニットに分散されてもよい。ユニットの一部又は全部は、本発明の実施例の対策の目的を達成するために、実際のニーズに従って選択されてもよい。

20

【0155】

更に、本発明の実施例における機能ユニットは、1つの処理ユニットに統合されてもよく、ユニットのそれぞれが物理的に単独で存在してもよく、2つ以上のユニットが1つのユニットに統合されてもよい。統合されたユニットは、ハードウェアの形式で実現されてもよく、ソフトウェア機能ユニットの形式で実現されてもよい。

【0156】

統合されたユニットがソフトウェア機能ユニットの形式で実現され、独立したプロダクトとして販売又は使用される場合、統合されたユニットは、コンピュータ読み取り可能記憶媒体に記憶されてもよい。このような理解に基づいて、基本的に本発明の技術的対策若しくは従来技術に寄与する部分、又は技術的対策の全部若しくは一部は、ソフトウェアプロダクトの形式で実現されてもよい。ソフトウェアプロダクトは、記憶媒体に記憶され、コンピュータデバイス（パーソナルコンピュータ、サーバ又はネットワークデバイスでもよい）に対して本発明の実施例に記載の方法のステップの全部又は一部を実行するように命令する複数の命令を含む。前述の記憶媒体は、USBフラッシュドライブ、取り外し可能ハードディスク、読み取り専用メモリ（ROM、Read-Only Memory）、ランダムアクセスメモリ（RAM、Random Access Memory）、磁気ディスク又は光ディスクのようなプログラムコードを記憶することができるいずれかの媒体を含む。

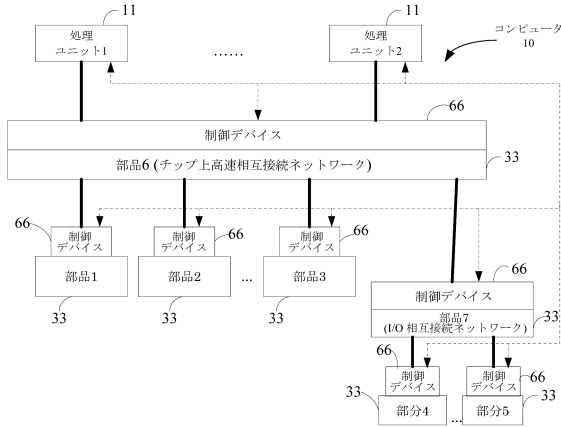
30

【0157】

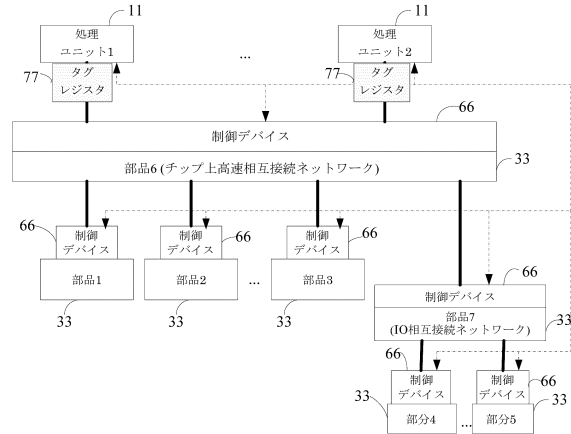
前述の説明は本発明の単に具体的な実施例に過ぎず、本発明の保護範囲を限定することを意図するものではない。本発明に開示された技術的範囲内で当業者により容易に認識される如何なる変更又は置換も、本発明の保護範囲内に入るものとする。従って、本発明の保護範囲は、特許請求の範囲の保護範囲に従うものとする。

40

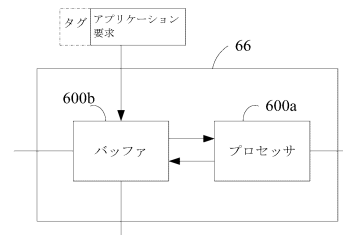
【図1】



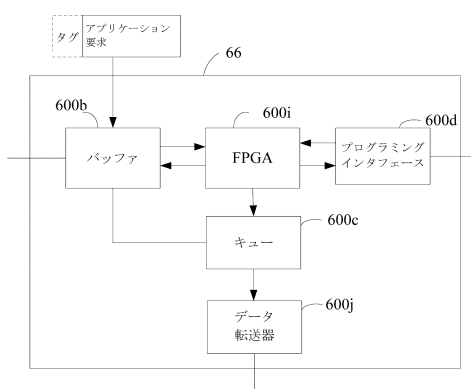
【図2a】



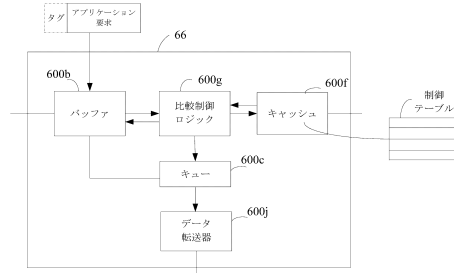
【図2b】



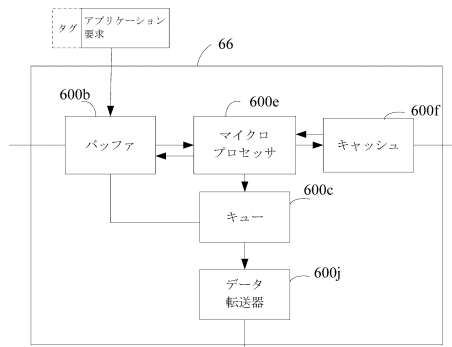
【図3】



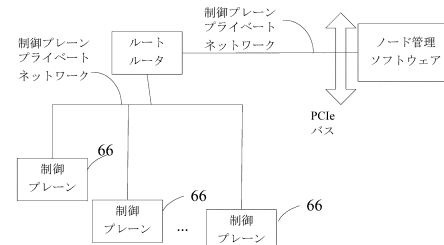
【図5】



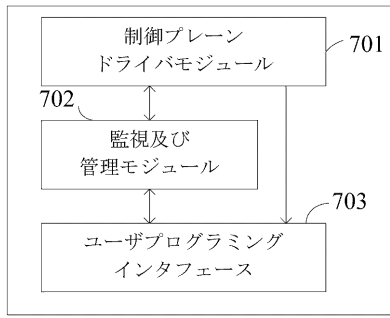
【図4】



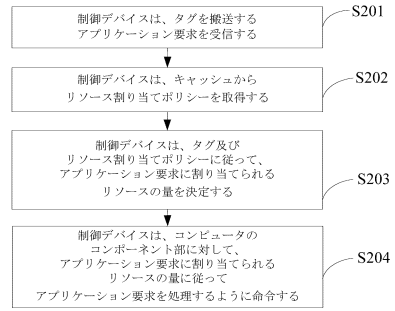
【図6】



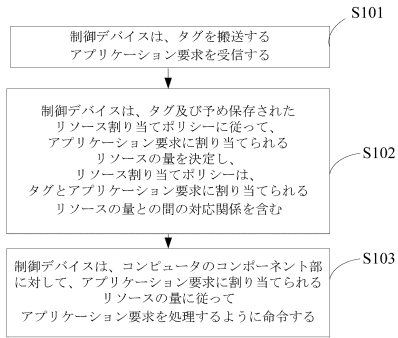
【図7】



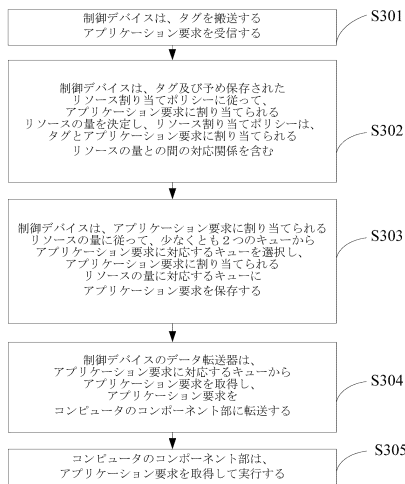
【図9】



【図8】



【図10】



フロントページの続き

- (74)代理人 100070150
弁理士 伊東 忠彦
- (74)代理人 100091214
弁理士 大貫 進介
- (72)発明者 包 雲 崗
中国100190北京市 海 淀区中 関 村科学院南路6号
- (72)発明者 馬 久 躍
中国100190北京市 海 淀区中 関 村科学院南路6号
- (72)発明者 隋 秀峰
中国100190北京市 海 淀区中 関 村科学院南路6号
- (72)発明者 任 睿
中国100190北京市 海 淀区中 関 村科学院南路6号
- (72)発明者 張 立新
中国100190北京市 海 淀区中 関 村科学院南路6号

審査官 田中 啓介

- (56)参考文献 特開2013-206233(JP,A)
特開2007-226280(JP,A)
特開2006-202244(JP,A)
特表2005-513587(JP,A)
特表2012-518825(JP,A)
米国特許出願公開第2009/0138682(US,A1)
米国特許出願公開第2007/0124446(US,A1)

- (58)調査した分野(Int.Cl., DB名)
G06F9/46-54
G06F13/10-13/14
G06F13/20-13/42