

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号  
特許第4606589号  
(P4606589)

(45) 発行日 平成23年1月5日 (2011.1.5)

(24) 登録日 平成22年10月15日 (2010.10.15)

(51) Int.Cl.

G O 6 F 13/12 (2006.01)

F I

G O 6 F 13/12 3 4 O A

請求項の数 9 (全 13 頁)

(21) 出願番号	特願2000-592732 (P2000-592732)	(73) 特許権者	596092698
(86) (22) 出願日	平成12年1月5日 (2000.1.5)		アルカテルルーセント ユーエスエー
(65) 公表番号	特表2003-532165 (P2003-532165A)		インコーポレーテッド
(43) 公表日	平成15年10月28日 (2003.10.28)		アメリカ合衆国 07974 ニュージャ
(86) 国際出願番号	PCT/US2000/000177		ーシー, マレイ ヒル, マウンテン アヴ
(87) 国際公開番号	W02000/041074		ェニュー 600-700
(87) 国際公開日	平成12年7月13日 (2000.7.13)	(74) 代理人	100094112
審査請求日	平成18年12月22日 (2006.12.22)		弁理士 岡部 譲
(31) 優先権主張番号	60/114, 771	(74) 代理人	100064447
(32) 優先日	平成11年1月5日 (1999.1.5)		弁理士 岡部 正夫
(33) 優先権主張国	米国 (US)	(74) 代理人	100085176
(31) 優先権主張番号	60/114, 772		弁理士 加藤 伸晃
(32) 優先日	平成11年1月5日 (1999.1.5)	(74) 代理人	100106703
(33) 優先権主張国	米国 (US)		弁理士 産形 和央

最終頁に続く

(54) 【発明の名称】 入出力デバイスを動的に再構成する装置及び方法

(57) 【特許請求の範囲】

【請求項 1】

ＩＯコントローラを介して、１つ又は２つ以上のポート・インタフェースからコンピュータ・システムの内部回路へ流れるデータ・フローを制御する方法であって、

前記ＩＯコントローラの所要構成を決定するステップと、

１つ又は２つ以上の記述子リストを処理できるように前記ＩＯコントローラを再プログラムするステップと、

１つ又は２つ以上の仮想コントローラの付加又は抹消を反映するように前記ＩＯコントローラの構成を変更するステップであって、前記変更は残りの１つ又は２つ以上の仮想コントローラの各々を介してデータ・フローを再配分することを特徴とする、ステップと、

前記ＩＯコントローラを再枚挙するステップと、

前記ＩＯコントローラ及び前記１つ又は２つ以上の仮想コントローラのそれぞれに対応する記述子を処理するステップとを含み、

ポート・インターフェースは、最大帯域幅を支持し、

前記１つ又は２つ以上の仮想コントローラは、前記再枚挙の過程において、発見され、起動され、前記ポート・インターフェースによって支持される前記最大帯域幅を提供することを特徴とする前記方法。

【請求項 2】

前記ＩＯコントローラの構成を変更するステップとして、前記１つ又は２つ以上の仮想コントローラの存在を示唆する１つ又は２つ以上の新しい構成記述子を作成することを特

徴とする請求項 1 に記載の方法。

【請求項 3】

前記 I O コントローラへの結線を、所要のルーティング構成となるように再ルートするステップをも含むことを特徴とする請求項 1 に記載の方法。

【請求項 4】

前記 I O コントローラの所要構成を決定し、前記コンピュータ・システム中のドライバによる制御下に、前記 I O コントローラの構成を変更することを特徴とする請求項 1 に記載の方法。

【請求項 5】

前記 I O コントローラの所要構成を決定し、前記 I O コントローラに組込まれたファームウェアによる制御下に、前記 I O コントローラの構成を変更することを特徴とする請求項 1 に記載の方法。

【請求項 6】

前記 I O コントローラの所要構成を決定するステップが、前記 I O コントローラの既存構成を、既存の条件に基づいて変更すべきかどうかを判断し、前記既存の条件に基づいて前記 I O コントローラの最適化構成を求めるステップを含むことを特徴とする請求項 1 に記載の方法。

【請求項 7】

コンピュータ・システムにおいて、I O コントローラを流れるデータ・フローを制御する方法であって、

1 つ又は 2 つ以上の所定条件が存在すると、再構成モードに入るステップと、

前記 I O コントローラの所要構成を決定するステップと、

1 つ又は 2 つ以上の記述子リストを処理できるように前記 I O コントローラを再プログラムするステップと、

1 つ又は 2 つ以上の仮想コントローラの付加又は抹消を反映するように前記 I O コントローラの構成を変更するステップであって、前記変更は残りの 1 つ又は 2 つ以上の仮想コントローラの各々を介してデータ・フローを再配分することを特徴とする、ステップと、

前記 I O コントローラを再枚挙するステップと、

前記 I O コントローラ及び前記 1 つ又は 2 つ以上の仮想コントローラのそれぞれに対応する記述子を作成するステップとを含み、

前記所定条件が、前記 I O コントローラを通過するデータ・フローの増減を含むことを特徴とする前記方法。

【請求項 8】

再構成モードに入る際に、前記コンピュータ・システムと接続しているデバイスの動作を中断させることを特徴とする請求項 7 に記載の方法。

【請求項 9】

I O コントローラとして使用される集積回路デバイスであって、

システム・バス・インタフェースと、

プログラム可能リスト・プロセッサと、

ポート・ルータと、

から成り、

前記システム・バス・インタフェース、前記プログラム可能リスト・プロセッサ及び前記ポート・ルータが機能的に接続され、

前記 I O コントローラは、1 つ又は 2 つ以上の記述子リストを処理し、

前記 I O コントローラは、1 つ又は 2 つ以上の仮想コントローラの付加又は抹消を反映し、

前記 I O コントローラは、再枚挙され、

記述子リストは、前記 I O コントローラ及び 1 つ又は 2 つ以上の仮想コントローラの各々について処理され、

1 つ又は 2 つ以上のポート・インターフェースは、最大帯域幅を支持し、

10

20

30

40

50

前記 1 つ又は 2 つ以上の仮想コントローラは、前記再枚挙の過程において、発見され、前記 1 つ又は 2 つ以上のポート・インターフェースによって支持される前記最大帯域幅を提供することを特徴とする前記集積回路デバイス。

【発明の詳細な説明】

【0001】

この出願は、その開示内容を参考のため本願明細書中に引用した、2000年1月4日付 Aguilar 外の米国特許出願第 \_\_\_\_\_ 号 "ポート・ルータ" に関連する。この出願は、その開示内容を、これも参考のため本願明細書中に引用した米国仮出願第 60 / 114 , 771 号、第 60 / 114 , 772 号及び第 60 / 114 , 767 号に関して優先権を主張する。

10

【0002】

〔発明の分野〕

この出願は半導体デバイス、特に、入／出力(I/O)コントローラに係る。

〔発明の背景〕

I/Oコントローラは、コンピュータ・オペレーティングシステムとI/Oインタフェースとを接続する機能を果たす。従来のコンピュータ・オペレーティングシステムの場合、インストラクションを形成する記述子リストをI/Oコントローラが読み取り、これに従ってその仕事を行う。最近のI/Oコントローラは、記述子中に規定されているオペレーションを読み取り、開始するハードウェア回路を含む。記述子に著しい変動があれば、新しい回路が必要になり、従って、既存回路が時代遅れのものになる。実際には、新しいオペレーティングシステムの開発と新しいハードウェアの開発が遅れる事態となる。両者の開発には、相互の整合性が要求されるからである。

20

【0003】

I/Oコントローラ・インタフェースの演算速度は固定的であるのが普通であり、コントローラはインタフェースの帯域幅に制限される。別のインタフェースを追加するには、アド・オン・ボードを追加するか、マザー・ボードを設計し直すことによって、新しいコントローラ・インタフェース・デバイスを構成しなければならない。

【0004】

I/Oデバイスの帯域幅を増大させる試みがなされているが、そのためにはデバイスのハードウェアを変更する必要がある、帯域幅を動的に加えることは不可能であった。ハードウェアの変更、又はシステムへのハードウェア追加には、再構成経費、追加ハードウェア経費の他、既存オペレーティングシステムと適合しない可能性など、数多くの問題がある。さらに、新しいハードウェアの追加はシステムをオフにして行われ、新しいハードウェアを組込む技術者が必要であり、新しいハードウェアをサポートするためにオペレーティングシステムを変更するシステム管理者も必要になるであろう。この方法は困難であり、エラーを伴い易く、コストが大きく、多大の時間を要する設計及び性能強化を必要とする。

30

【0005】

従って、記述子の変動に対応することによって、機能を追加し、性能を向上させ、又は前後両方向のコンパチビリティを可能にするデバイスの実現が望まれる。さらにまた、システムの条件に応じて、新しいハードウェアを追加することなく、I/Oコントローラの新しいインスタンスを動的に形成することによって追加の帯域幅を提供できるデバイスもの実現も望まれる。

40

【0006】

〔発明の概要〕

本発明の第1の態様においては、I/Oコントローラによってコンピュータ・システムへのデータ・フローを制御する方法が提供される。この方法は、I/Oコントローラの所要構成を求め、1つ又は2つ以上の記述子进行处理できるようにI/Oコントローラを再プログラミングし、1つ又は2つ以上の仮想コントローラの追加又は抹消を反映するようにI/Oコントローラの構成を変更し、I/Oコントローラを枚挙し直し、各I/Oコントローラ及び1つ又は2つ以上の仮想コントローラごとに、記述子リスト进行处理することを含む。枚挙し直

50

しの過程で1つ又は2つ以上の仮想コントローラが発見され、初期設定され、I/Oコントローラによってサポートされる全帯域幅を提供することができる。

【0007】

第2の態様においては、本発明は、システム・バス・インタフェース、プログラマブル・リスト・プロセッサ及びポート・ルータからなる、I/Oコントローラとして使用する集積回路デバイスである。この集積回路デバイスは、1つ又は2つ以上の仮想コントローラを追加又は抹消するための再構成に採用される。仮想コントローラは、集積回路デバイスによってサポートされる、実質的な全帯域幅を提供する。

【0008】

本発明はパーソナル・コンピュータ・システムにおいて利用できるが、例えば、情報機器、セット・トップ・ボックス、ケーブル・モデム、ゲーム・コンソール、スマート・アプライアンス、ハンドヘルド・コンピュータ、手のひらサイズ・コンピュータ、組込制御システム、ワークステーション、サーバーなどのようなタイプのコンピュート・プラットフォームにも応用できる。

【0009】

〔発明の詳細な説明〕

本発明はデータ・フロー・プロトコル及び/又は上位システム相互作用に起こる変化に対応するように、I/Oデバイスを動的に再構成する装置及び方法を提供する。本発明は、データ転送のため、オペレーティングシステムが使用できる新しいチャンネルを形成する仮想コントローラの追加(又は抹消)を可能にする。I/Oデバイスは、高帯域幅を必要とする周辺デバイスのための専用スループットを提供し、複数の低帯域幅デバイスをグルーピングすることによって、データ転送を効率化することができる。本発明は、オペレーティングシステムの変化、システム条件の変化、デバイス中の問題又はバグ、及びデータ・フロー・プロトコルに対応できる新しいオペレーティングプログラムを組込むことを可能にする。

【0010】

本発明は、追加内部回路をコンピュート・プラットフォーム(即ち、I/Oコントローラ)内に集積する、現実的且つ経済的解決を提供する。I/Oコントローラは、I/O流れ(バス、インタフェース、ネットワーク・コネクション)とホスト・コンピュータにおけるオペレーティングシステム・ドライバとの間にブリッジを形成する。本発明は、プログラム化可能性を利用することにより、ブリッジのI/O流れとオペレーティングシステム・インタフェースとの双方の側の動作を変化させる方法及び装置を提供する。

【0011】

本発明は、例えば、USB, IEEE-1394, Fibre Channel, Infiniband及びEthernetなどのようなI/Oコントローラに応用できる。本発明の装置及び方法は、周辺デバイスとコンピュート・プラットフォームとの間のデータ・フローを増大させることができる。データ・フローを増大させるのにユーザが干渉する必要はなく、コンピュート・プラットフォームに複合デバイスを組込むだけでよい。本発明はパーソナル・コンピュータ・システムに利用できるが、その他のタイプのコンピュート・プラットフォーム、例えば、情報機器、セット・トップ・ボックス、ケーブル・モデム、ゲーム・コンソール、スマート・アプライアンス、ハンドヘルド・コンピュータ、手のひらサイズ・コンピュータ、組込制御システム、ワークステーション、サーバーなどのようなタイプのコンピュート・プラットフォームにも応用できる。

【0012】

本発明のI/Oコントローラはマーケット寿命が長く、バグに対する対策を速やかに練り、実施することを可能にし、オペレーティングシステムの記述子リストのエラーにも変更にも適応し、新しいタイプの記述子演算の組込を可能にする。本発明は、広範囲にわたるシステムの諸条件を満たすのに必要な製品数を軽減する。

【0013】

添付の図面はあくまでも説明のためのものであり、本発明の可能な実施態様を制限するも

10

20

30

40

50

のではない。ポート及びＩＯコントローラの数是个々のシステム又は用途に必要な条件に応じて異なる。

図１に示す本発明の一実施例では、システム・バス・インタフェース１２０、プログラム可能なリスト・プロセッサ１１０、及びポート・ルータ１３０を有する、ＩＯコントローラとして使用するための集積回路（ＩＣ）デバイス１００を設ける。システム・バス・インタフェース１２０、プログラム可能なリスト・プロセッサ１１０及びポート・ルータ１３０は機能的に接続されている。ポート・ルータ１３０は好ましくは１つ又は２つ以上のインタフェース・ポート１４０とも接続している。

#### 【００１４】

プログラム可能なリスト・プロセッサ１１０を採用することにより、ＩＯコントローラは、"シングル・スレッド"、即ち、一度に単一の記述子リストだけを実行できる既存のシステムとは異なり、多重記述子リストを処理するために再プログラム又は再構成することができる。この機能により、本発明のＩＯコントローラは複数の別々のハードウェア・コントローラと同様に作用することができる。

#### 【００１５】

図４は本発明と併用できるポート・ルータ４００の一実施例を示すブロックダイアグラムである。ポート・ルータ４００は、コントローラ・スイッチ４２０、ポート・スイッチ４４０、及びコントローラ・スイッチ４２０とポート・スイッチ４４０の間の結線４５０から成る。結線４５０はハブ素子４０１を含む。ポート・ルータ４００はポートＡ～Ｄを単一のコントローラ（コントローラＡ）と接続する。コントローラ・スイッチ４２０、ポート・スイッチ４４０及び結線４５０はそれぞれ固有の可能入出力を有する。ポート・スイッチ４４０は固定入力４４１～４４４を有し、それぞれの固定入力はインタフェース・ポートと接続している。ポート・スイッチ４４０は任意のポートを１つ又は２つ以上のポート・スイッチ出力ヘルテイングする。ポート・スイッチ４４０はすべてのルーティング素子を支持するのに十分な出力総数を有する。図４の実施例の場合、ポート・スイッチ４４０はハブ素子４０１のための４個の出力を有する。ハブ素子は、用途に応じたＩＯインタフェースの条件を満たすものであることが好ましい。各ハブ素子はその入力を組合わせ、コントローラ・スイッチ４２０に入力される単一の出力を提供する。ＩＯインタフェースによっては、ハブ素子４０１をブリッジ、コンセントレーター又は物理的インタフェースと呼称する場合がある。ポート・スイッチ４４０をコントローラ・スイッチ４２０に接続するには、多重ハブ素子を使用すればよい。

#### 【００１６】

コントローラ・スイッチ４２０は、ハブ素子との結線ごとに１つの入力、ポート・スイッチ４４０との各直接結線（図２）ごとに１つの入力をそれぞれ有することが好ましい。コントローラ・スイッチ４２０からの出力は所与の用途に必要な、又は利用可能なコントローラの個数に合わせて設定する。図４に示すように、コントローラ・スイッチは、ハブ素子４０１と接続する１つの入力４２１、及びコントローラＡと接続する１つの出力４２７を有する。なお、ここに言う"入力"及び"出力"はいずれも、双方向データ・フローを扱うことができる結線（一般的には"入／出力"と呼称される）を意味する。

#### 【００１７】

図５は、図４のポート・ルータ４００のポートＤに新しいデバイスを付加した場合を示し、ここでは、コントローラＡの帯域幅容量が閾値に達する。システムにコントローラＢが付加され、ポートＤからハブ素子２０１への結線が除去され、直接結線５０２が付加されている。ポートＤにデバイスが挿入されると、デバイスが検知され、その容量条件がリポートされる。この容量は、コントローラＡの有効容量よりも大きくなるように計算されている。新しいコントローラＢがシステムに付加されると、ポートＡ、Ｂ及びＣがコントローラＡと接続する一方、ポートＤがコントローラＢとだけ接続するようにポート・ルータ１００が再構成される。ポートＤとハブ素子５０１とを結ぶ結線が除去され、ポート・スイッチ５４０とコントローラ・スイッチ出力５２０との間に新しい直接結線５０２が形成される。コントローラ・スイッチ出力５２８からコントローラＢへの結線が形成される。

ポートD及び新しいデバイスとの間のデータ・フローはすべて、ポート・スイッチ出力549からコントローラ・スイッチ入力522へのこの新しい結線を流れる。ポートA、B及びCからのデータ・フローはすべて、ポート・スイッチ出力545、546及び547からハブ素子501への同じルートをとる。この構成では、ポートA、B及びCからのデータ・フローが組み合わされ、コントローラ・スイッチ入力521へ供給される。

#### 【0018】

動作時には、いくつかのポート入力が必要のコントローラと接続しているデフォルト状態から、ポート・スイッチ400がスタートすることが好ましい。図4の実施例の場合、システムは4つのポート、ポートA及びBと接続する2つのデバイス、1つのハブ素子401、及び1つのコントローラAを備えたデフォルト状態にある。好ましくはポート・ルータ400中に存在する1組の(図示しない)ハードウェア・レジスタを介して、システム中で動作するソフトウェアがポート・ルータ400に問い合わせ、デフォルト・トポロジを発見することができる。システムのソフトウェアはハードウェア・レジスタの値をプログラムすることによって、ポート及びコントローラ間の信号ルートを変更することができる。動作中に事象が起こる。この事象としては、例えば、新しいデバイスの除去又はポートへの挿入が考えられる。この事象がソフトウェアをトリガーすることによって、この新しいデバイスの条件を検討する。これらの条件には、デバイスのデータ容量条件(帯域幅及び待ち時間)やそのデータ・スタイル(非同期的、等時的、バースト、流れ)などが含まれる。新しい条件が、既に実装されているデバイスから課せられる一連の条件と組み合わせられ、ポートをルーティングするための新しい、好ましくは最適化されたトポロジが計算される。好ましくは、システムのソフトウェアがIOの動作を中断させ、ポートとIOコントローラとの間を再ルーティングさせる。次いで、システムのソフトウェアがIOの動作を再開させ、最適化されたルーティングが、新しい事象が起こるまで、新しいルーティングとなる。

#### 【0019】

他の実施例では、ポート・ルータ400が、システム・ソフトウェアの干渉を必要とすることなく、ルーティング及び再ルーティングが内部的に行われることを可能にする組込ソフトウェアを含むことが好ましい。これにより、自己モニターが可能になり、ポート・ルータ400を通過するデータ・フローに基づく動的な負荷平衡が可能になる。

#### 【0020】

図2に示す本発明の他の実施例では、システム・バス・インタフェースとしての周辺部品相互接続(PCI)バス220、プログラム可能リスト・プロセッサ210、ポート・ルータ230、データ・マックス250、及び複数のFIFO(先入れ先出し)データ・バッファ260を有する、IOコントローラとして利用される集積回路(IC)デバイス200を設ける。また、ポート・ルータ230に1つ又は2つ以上のインタフェース・ポート240を接続することが好ましい。素子のそれぞれが機能的に接続されることにより、データ信号及び制御信号が素子間で伝送されるようにすることが好ましい。

#### 【0021】

インタフェースから受信されるデータは、好ましくはポート240及びポート・ルータ230を経て、特定のFIFO260に向けられることが好ましい。データ・エレメントがFIFO260に受信されると、リスト・プロセッサ210がこれを検討し、その配置を確認することが好ましい。データ・エレメントに対する作用は、インタフェースに対するプロトコル・ルール次第であり、例えば、Ethernet、USB、IEEE-1394等では、それぞれ異なる作用となる。リスト・プロセッサ210に組込まれている制御ソフトウェアが所要の構成を決定することが好ましい。リスト・プロセッサ210は、例えば、下記のように作用することが好ましい。すなわち、データ・エレメントをシステム・メモリへ伝送し、未完のトランザクションを完了させ、エラーをシステムへ伝送させ、エラーをインタフェースへ伝送させ、インタフェースへ伝送すべきデータを作成する。PCIバス220を介してシステム・メモリとの間で伝送されるデータはすべて、好ましくはデータ・マックス250素子を通過する。システム・メモリからインタフェースヘデー

10

20

30

40

50

タ・エレメントを伝送したい場合、P C Iバス220を介して受信し、データ・マックス250を通過させ、該当のF I F O 260に入力させることが好ましい。システム・メモリからのデータ取出しを開始させたリスト・プロセッサは、インタフェースへ伝送されるように準備し、データ伝送が可能な状態にあるかどうか、インタフェースの状態をモニターすることが好ましい。もしインタフェースが応答しないか、又は作動不能なら、リスト・プロセッサ210は下記のように作用することが好ましい。すなわち、後で再伝送するためデータを保管、システムへエラーをレポート、等である。

【0022】

リスト・プロセッサ素子210は、ポートとシステム・バスの間のデータ・フローを制御する機能を有するマイクロプロセッサやマイクロコントローラのような、汎用コンピュータ・エンジンの特性の多くを有することが好ましい。リスト・プロセッサ素子210は、好ましくはR O Mに固定されているか、より好ましくはダウンロードされているソフトウェアを実行する。ソフトウェアは、好ましくはインタフェースに関するプロトコルの定義、典型的には、工業規格に従って書き込まれる。プライベート・インタフェース規格に基づくソフトウェアを使用することもできることが好ましい。例えば、プロトコルによる規定上、システムがデータ・エレメントを送信し、データが正しく受信されたとの確認を受信しなければならない場合、リスト・プロセッサ素子は、確認の受信を想定し、確認が受信されない場合には、再送信を開始するか、又はシステムへエラーを送信するソフトウェア制御構造を有する。具体的な作用内容は、プロトコル定義のルール次第である。

【0023】

システム・バス120は、例えば、P C I - X , I n f i n i b a n d , V M Eバス、H u b L i n k等のようなマルチマスター・バス構造をも含むことができる。これらオペレーティングシステムに共通の特徴は枚挙段階である。枚挙の段階において、システム・ソフトウェアが、システム・バスに存在するハードウェアを問合わせ、ハードウェアの特徴を記述するデータを作成する。本発明と併用されるオペレーティングシステムは、必要なら再枚挙できることが好ましい。

【0024】

本発明は、枚挙の段階においてレポートされるデフォルト構成を有する。好ましいデフォルト状態において、リスト・プロセッサ110は、単一のリストを処理するようにプログラムされ、オペレーティングシステムに対して単一のI Oコントローラをレポートするように構成されている。また、起動時におけるデフォルト状態が、それまでのパワー・ダウン前から維持されている状態であることが好ましい。

【0025】

動作中、リスト・プロセッサ110は好ましくは、オペレーティングシステムの指令を翻訳し、実行するように機能する。オペレーティングシステムが理解し、あたかも複数の個別I Oコントローラ・ハードウェアが存在するかのように作用できるように、仮想コントローラを形成することができる。

I Oコントローラのハードウェア記述を修正し、オペレーティングシステムがI Oコントローラの再枚挙することをリクエストすることにより、1つ又は2つ以上の新しい仮想I Oコントローラを形成又は抹消することができる。新しいI Oコントローラの動的形成は、実際の限界（即ち、システム・バス帯域幅、I Oコントローラ数の限界等）に達するまで何度でも行うことができる。

【0026】

図3のフロー・ダイアグラムに示す本発明の方法の一実施例では、現時点のI Oコントローラ構成を変更する決定がなされる。この決定はオペレーティングシステム又はI Oコントローラ内の（図示しない）モニタリング素子によって行うことができる。現時点のI Oコントローラ構成を変更する決定は、例えば、I Oコントローラを通過するデータ・フローの増減などのような1つ又は2つ以上の所定条件の存在に基づいて行うことが好ましい。

【0027】

この決定は、例えば、下記判定基準に基づいて行うことが好ましい。周辺デバイスの総帯域幅条件、特定周辺デバイスの帯域幅条件、特定周辺デバイスの待ち時間条件、競合周辺デバイスの存在、周辺デバイスのトポロジ、周辺デバイスの付加又は除去、既存周辺デバイス帯域幅条件の増減、既存周辺デバイス待ち時間条件の増減、等。

#### 【0028】

次いで、ＩＯコントローラの動作を一時的に中断させることが好ましい。ＩＯコントローラに組込まれているデバイスの作用を中断（又は、スリープへの移行）させることにより、データ・フローを止める。次いで、リスト・プロセッサ１１０を、システムに付加された仮想コントローラのそれぞれについて、１つ又は２つ以上の記述子リストを処理できるようにプログラムすることが好ましい。それまで演算中であった記述子リストは手つかずのままであり、演算が再開されると、新しい記述子リストと一緒に処理される。ＩＯコントローラの構成は、１つ又は２つ以上の仮想コントローラの付加又は抹消を反映するように変更される。好ましくは、ＩＯコントローラと接続しているデバイスを、効率のよいデータ・パスを提供して、ＩＯコントローラを通過するデータ・フローを最適化又は改善する所要の構成となるように再ルーティングする。この再ルーティングは、典型的な方法として、データ・フローを１つ又は２つ以上のＩＯコントローラ間に分割することによって行われる。

10

#### 【0029】

再構成が完了すると、ＩＯコントローラを再枚挙するようオペレーティングシステムに対して、リクエストが行われる。仮想コントローラが検知され、どのコントローラにどのデバイスを接続すべきかをオペレーティングシステムが問合わせる。記述子リストが作成され、リスト・プロセッサ１１０が各記述子リストの処理を開始する。次いで、デバイスは動作を再開することができる。典型的には、オペレーティングシステムによってデバイスが再枚挙される。

20

#### 【0030】

リスト・プロセッサ１１０の特徴として、多重記述子リストを同時的に処理できる、いわゆるマルチ・スレッド性が挙げられる。リスト・プロセッサ１１０が同時的に実行できる代表的な動作としては、下記の動作が挙げられる。システム・メモリーからの取出し、キャッシング（又は、システム・メモリーへのアクセス回数を減らさねばならなくなる前に、記述子を取り出すこと）、記述子のバリデーション、データ伝送用ＩＯハードウェアの準備、データ・フロー・プロトコル・エレメントの構成、システム・メモリーとの間のデータ取出し又は供給、除外すべきデータ・フローのモニタリング、完成すべきデータ・フローのモニタリング、記述子の終了、エラーの処理及び一掃、及びオペレーティングシステムへの結果レポート。

30

#### 【0031】

したがって、本発明の一実施例による、コンピュータ・システムにおいてＩＯコントローラを通過するデータ・フローを制御する方法は、ＩＯコントローラに必要な構成を決定し、１つ又は２つ以上の記述子リストを処理できるようにＩＯコントローラを再プログラムし、１つ又は２つ以上の仮想コントローラの付加又は抹消を反映するようにＩＯコントローラの構成を変更し、ＩＯコントローラを再枚挙し、ＩＯコントローラ及び仮想コントローラのそれぞれについて、１つの記述子リストを処理するステップから成る。枚挙の過程において、１つ又は２つ以上の仮想コントローラを発見し、起動し、そして、ＩＯコントローラによって支持され、且つインタフェース仕様によって定められた全帯域幅を提供することが好ましい。

40

#### 【0032】

ＩＯコントローラの構成を変更する際には、１つ又は２つ以上の仮想コントローラの存在を示唆する１つ又は２つ以上の新しい構成記述子を作成することが好ましい。インタフェース・ポートからの結線を、効率的なデータ・フローを可能にする所要のルーティング構成となるように、再ルーティングすることが好ましい。ＩＯコントローラに必要な構成の決定と、ＩＯコントローラの構成変更はコンピュータ・システム中のドライバー、又はＩ

50



Ｏコントローラに組込まれたファームウェアの制御下に行われることが好ましい。ＩＯコントローラの所要構成を決定する際には、ＩＯコントローラの既存構成を、既存の条件に基づいて変更すべきかどうかを判断し、既存条件に基づいてＩＯコントローラの最適化構成を決定することが好ましい。

【００３３】

本発明の他の実施例による、コンピュータ・システムにおいてＩＯコントローラを通過するデータ・フローを制御する方法は、１つ又は２つ以上の所定条件が存在すると、再構成モードに入り、ＩＯコントローラの所要構成を決定し、１つ又は２つ以上の記述子リストを処理できるようにＩＯコントローラを再プログラムし、１つ又は２つ以上の仮想コントローラの付加又は抹消を反映するようにＩＯコントローラの構成を変更し、ＩＯコントローラを再枚挙し、各ＩＯコントローラ及び１つ又は２つ以上の仮想コントローラごとに１つの記述子リストを処理するステップから成る。再構成モードに入る際には、コンピュータ・システムと接続しているデバイスの動作を中断させることが好ましい。所定条件には、ＩＯコントローラを通過するデータ・フローの増減が含まれることが好ましい。

10

【００３４】

第２の態様においては、本発明は、システム・バス・インタフェース、プログラム可能リスト・プロセッサ及びポート・ルータから成る、ＩＯコントローラとして利用される集積回路デバイスに係る。システム・バス・インタフェース、プログラム可能リスト・プロセッサ及びポート・ルータは機能的に接続されることが好ましい。集積回路デバイスは、１つ又は２つ以上の仮想コントローラを付加又は抹消するため再構成でき、仮想コントローラが、集積回路デバイスによって支持される、実質的な全帯域幅を提供することが好ましい。

20

【００３５】

添付図面に沿って本発明の実施例を以上に説明したが、本発明はこれら実施例に制限されるものではなく、当業者なら、本発明の思想と範囲を逸脱することなく、多様な変更を加えることができる。

【図面の簡単な説明】

【図１】 本発明の一実施形態のブロック・ダイアグラムである。

【図２】 本発明の他の実施形態のブロック・ダイアグラムである。

【図３】 本発明の方法の一実施形態を説明するフロー・ダイアグラムである。

30

【図４】 本発明と併用できるポート・ルータの一実施形態を示すブロック・ダイアグラムである。

【図５】 本発明と併用できるポート・ルータの他の実施形態を示すブロック・ダイアグラムである。

【図 1】

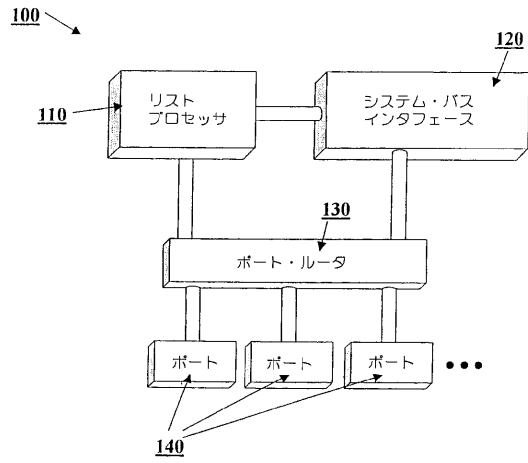


Fig 1

【図 2】

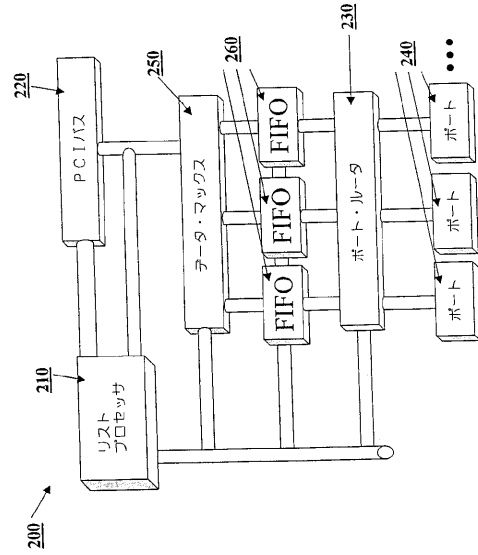


Fig 2

【図 3】

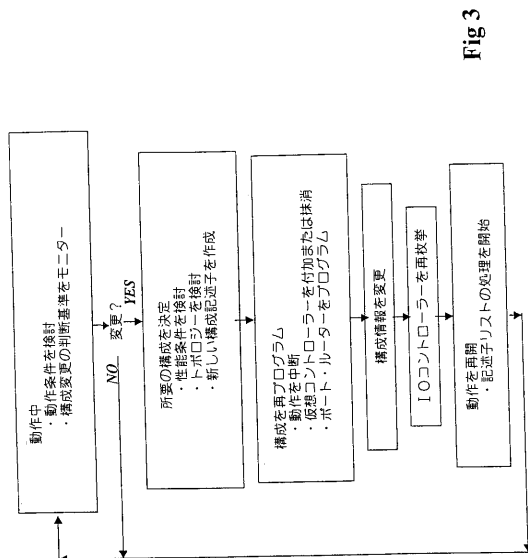


Fig 3

【図 4】

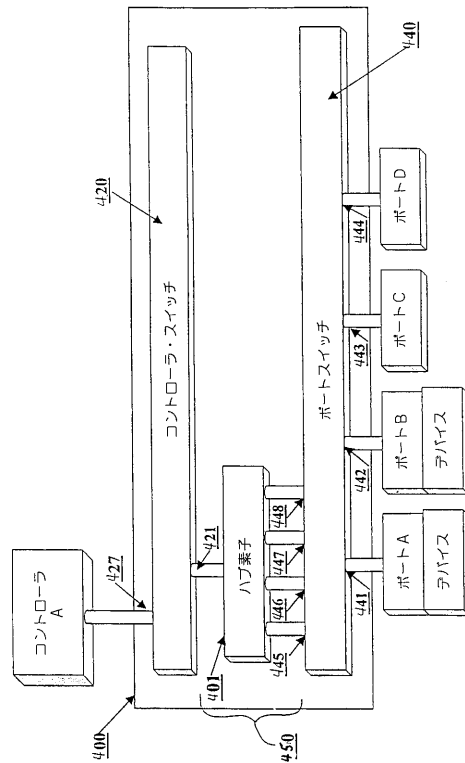


Fig 4

【図 5】

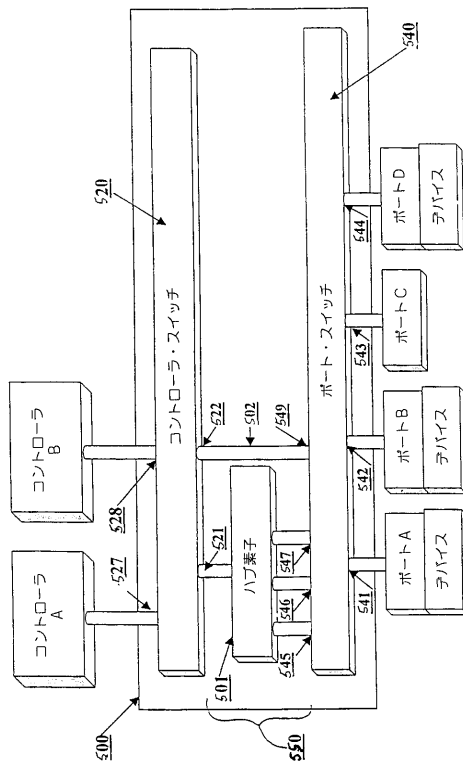


Fig 5

## フロントページの続き

(31)優先権主張番号 60/114,767

(32)優先日 平成11年1月6日(1999.1.6)

(33)優先権主張国 米国(US)

(31)優先権主張番号 09/477,591

(32)優先日 平成12年1月4日(2000.1.4)

(33)優先権主張国 米国(US)

(74)代理人 100096943

弁理士 臼井 伸一

(74)代理人 100091889

弁理士 藤野 育男

(74)代理人 100101498

弁理士 越智 隆夫

(74)代理人 100096688

弁理士 本宮 照久

(74)代理人 100102808

弁理士 高梨 憲通

(74)代理人 100104352

弁理士 朝日 伸光

(74)代理人 100107401

弁理士 高橋 誠一郎

(74)代理人 100106183

弁理士 吉澤 弘司

(72)発明者 アギラー, ラウル エー.

アメリカ合衆国, ペンシルベニア 18055, ヘラータウン, パイン マナー ドライブ 240

(72)発明者 クリー, ジェームズ ティー.

アメリカ合衆国, ペンシルベニア 18069, オーアフィールド, ローラル レーン 1708

(72)発明者 グジアック, ジェームズ イー.

アメリカ合衆国, ペンシルベニア 18059, ローリーズ ステーション, アッシュレイ ドライブ 5439

(72)発明者 リンチ, ケビン ジェイ.

アメリカ合衆国, ペンシルベニア 18080, スラティントン, シダー プレス ロード 4225

(72)発明者 ラティフ, ファールク エー.

アメリカ合衆国, ペンシルベニア 19446, ランズデール, アレパッチ ロード 2205

審査官 横山 佳弘

(56)参考文献 特開平09-160889(JP, A)

特開平04-081144(JP, A)

特開平10-112726(JP, A)

特開平04-267133(JP, A)

特開2001-167040(JP, A)

(58)調査した分野(Int.Cl., DB名)

G06F 13/12

G06F 15/16

