

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号  
特許第7379491号  
(P7379491)

(45)発行日 令和5年11月14日(2023.11.14)

(24)登録日 令和5年11月6日(2023.11.6)

(51)国際特許分類

F I

G 0 6 F 13/10 (2006.01)

G 0 6 F 13/10 3 1 0 B

G 0 6 F 13/14 (2006.01)

G 0 6 F 13/10 3 3 0 Z

G 0 6 F 13/14 3 1 0 J

請求項の数 25 (全27頁)

(21)出願番号	特願2021-534351(P2021-534351)	(73)特許権者	390009531
(86)(22)出願日	令和2年1月14日(2020.1.14)		インターナショナル・ビジネス・マシー
(65)公表番号	特表2022-518349(P2022-518349		ンズ・コーポレーション
	A)		INTERNATIONAL BUSI
(43)公表日	令和4年3月15日(2022.3.15)		NESS MACHINES CORPO
(86)国際出願番号	PCT/EP2020/050757		RATION
(87)国際公開番号	WO2020/156797		アメリカ合衆国10504 ニューヨー
(87)国際公開日	令和2年8月6日(2020.8.6)		ク州 アーモンク ニュー オーチャード
審査請求日	令和4年6月22日(2022.6.22)		ロード
(31)優先権主張番号	19154735.5		New Orchard Road, A
(32)優先日	平成31年1月31日(2019.1.31)		rmonk, New York 105
(33)優先権主張国・地域又は機関	欧州特許庁(EP)		04, United States of
			America
		(74)代理人	100112690
			弁理士 太佐 種一

最終頁に続く

(54)【発明の名称】 入出力ストア命令をハンドリングする方法、システム、およびプログラム

(57)【特許請求の範囲】

【請求項1】

入出力ストア命令(30)をハンドリングするためのデータ処理システム(210)であって、入出力バス・コントローラ(20)によって少なくとも1つの入出力バス(22)に通信連結されたシステム・ネスト(18)を備え、

コア(12)、システム・ファームウェア(10)、および非同期コア・ネスト・インターフェース(14)を備えるデータ処理ユニット(216)を少なくともさらに備え、

前記データ処理ユニット(216)が、集約バッファ(16)を介して前記システム・ネスト(18)に通信連結され、

前記システム・ネスト(18)が、前記入出力バス(22)に通信連結された少なくとも1つの外部デバイス(214)からデータを非同期的にロードすること、または前記少なくとも1つの外部デバイス(214)にデータを非同期的にストアすること、あるいはその両方を行うように構成され、

前記非同期コア・ネスト・インターフェース(14)が、複数の入出力ステータス・バッファ(24)を伴う入出力ステータス・アレイ(44)、アレイ管理兼アクセス・ロジック(46)を備え、

(i)前記データ処理システム(210)上で動くオペレーティング・システムが、アドレスを通じたオフセットを伴う入出力機能、移送されることになるデータ、または移送されることになるデータへのポインタ、あるいはその両方、およびデータの長さを少なくとも指定する前記入出力ストア命令(30)を発行すること、

10

20

( i i ) 前記データ処理ユニット ( 2 1 6 ) が、前記入出力ストア命令 ( 3 0 ) で指定された前記アドレスによって前記入出力機能を識別するように構成されること、

( i i i ) 前記データ処理ユニット ( 2 1 6 ) が、アドレス空間およびゲスト・インスタンス・レベルで前記入出力機能へのアクセスが許可されるかどうかを検証するように構成され、前記ゲストが前記データ処理システム ( 2 1 0 ) 上で動くこと、

( i v ) 前記データ処理ユニット ( 2 1 6 ) が、前記システム・ネスト ( 1 8 ) における前記入出力ストア命令 ( 3 0 ) の実行が完了する前に、前記入出力ストア命令 ( 3 0 ) を完了させるように構成されること、

( v ) 前記システム・ファームウェア ( 1 0 ) が、前記入出力ストア命令 ( 3 0 ) の非同期実行中に前記データ処理ユニット ( 2 1 6 ) によってエラーが検出された場合、失敗した前記非同期実行の前記データを伝送することを、割込みを通じて前記オペレーティング・システムに通知するように構成されること、

( v i ) 前記アレイ管理兼アクセス・ロジック ( 4 6 ) が、前記入出力ストア命令 ( 3 0 ) の完了を収集し、受信した完了メッセージに基づいて前記入出力ステータス・バッファ ( 2 4 ) を更新すること、

( v i i ) 前記データ処理ユニット ( 2 1 6 ) が、待ち状態の入出力ストア命令 ( 3 0 ) の完了についての情報をストアするために入出力ステータス・バッファ ( 2 4 ) が利用可能になるまで、前記入出力ストア命令 ( 3 0 ) の実行を遅らせることを含む、データ処理システム ( 2 1 0 ) 。

#### 【請求項 2】

前記データ処理ユニット ( 2 1 6 ) が、待ち状態の入出力ストア命令 ( 3 0 ) の完了についての情報をストアするために入出力ステータス・バッファ ( 2 4 ) が利用可能になるまで、非同期ストア命令 ( 3 0 ) または同期ストア命令 ( 3 0 ) あるいはその両方の実行を遅らせる、請求項 1 に記載のデータ処理システム。

#### 【請求項 3】

さらに、前記入出力ステータス・バッファ ( 2 4 ) が、前記システム・ネスト ( 1 8 ) から、または前記入出力バス・コントローラ ( 2 0 ) から、あるいはその両方からメッセージ状態を、具体的には、前記システム・ネスト ( 1 8 ) から完了ステータスを収集する、請求項 1 または 2 に記載のデータ処理システム。

#### 【請求項 4】

前記メッセージ状態または前記完了ステータスあるいはその両方が、入出力ステータス・バッファ・インデックスで番号を付けられる、請求項 3 に記載のデータ処理システム。

#### 【請求項 5】

前記集約バッファ ( 1 6 ) が、前記非同期コア・ネスト・インターフェース ( 1 4 ) に非同期バス ( 3 8 ) を介して通信連結される、請求項 1 ないし 4 のいずれか一項に記載のデータ処理システム。

#### 【請求項 6】

前記データが、前記データの前記長さが 8 バイトを超える場合、非同期伝送メカニズムを通じて前記入出力ストア命令 ( 3 0 ) によって、早期完了メッセージとともに複数のデータ・パケットで前記集約バッファ ( 1 6 ) に移送され、そうでなければ、前記データが、1 つのデータ・パケットで移送される、請求項 1 ないし 5 のいずれか一項に記載のデータ処理システム。

#### 【請求項 7】

前記システム・ファームウェア ( 1 0 ) が、前記入出力ストア命令 ( 3 0 ) をハンドリングするための非同期入出力ドライバ・コード ( 3 2 ) を含む、請求項 1 ないし 6 のいずれか一項に記載のデータ処理システム。

#### 【請求項 8】

前記コア ( 1 2 ) が、前記非同期入出力ドライバ・コード ( 3 2 ) のステータス情報のためのメモリ要件をハンドリングするための非同期セットアップ・コード ( 3 4 ) を備える、請求項 7 に記載のデータ処理システム。

10

20

30

40

50

## 【請求項 9】

前記非同期コア・ネスト・インターフェース(14)が、ローカル完了に伴い前記データを転送するための非同期コア・ネスト・インターフェース転送構成要素(36)を備える、請求項1ないし8のいずれか一項に記載のデータ処理システム。

## 【請求項 10】

前記集約バッファ(16)が、リクエストを送信した後、再使用のための空きがあるというメッセージを配信するための早期完了ロジック(26)を備える、請求項1ないし9のいずれか一項に記載のデータ処理システム。

## 【請求項 11】

前記システム・ファームウェア(10)が、前記入出力ステータス・アレイ(44)内の入出力ステータス・バッファ(24)をアロケート/デアロケートすること、または新しいストア命令(30)の開始を始めること、あるいはその両方を行うアレイ管理ロジック(42)を備える、請求項1ないし10のいずれか一項に記載のデータ処理システム。

## 【請求項 12】

システム・メッセージが、

- 階層型物理ターゲット・アドレス、
- SMTスレッドまたは集約バッファ識別子の発生源を確認すること、
- データの長さ、
- 入出力バス・アドレス、
- 入出力ステータス・バッファ・インデックス

のうちの1つを含む、請求項1ないし11のいずれか一項に記載のデータ処理システム。

## 【請求項 13】

データ処理システム(210)の少なくとも1つの外部デバイス(214)への入出力ストア命令(30)をハンドリングするための方法であって、前記データ処理システム(210)が、

入出力バス・コントローラ(20)によって少なくとも1つの入出力バス(22)に通信連結されたシステム・ネスト(18)を備え、

コア(12)、システム・ファームウェア(10)、および非同期コア・ネスト・インターフェース(14)を備えるデータ処理ユニット(216)を少なくともさらに備え、

前記データ処理ユニット(216)が、集約バッファ(16)を介して前記システム・ネスト(18)に通信連結され、

前記外部デバイス(214)が、前記入出力バス(22)に通信連結され、

前記非同期コア・ネスト・インターフェース(14)が、複数の入出力ステータス・バッファ(24)を伴う入出力ステータス・アレイ(44)、アレイ管理兼アクセス・ロジック(46)を備え、

前記方法が、

(i) 前記データ処理システム(210)上で動くオペレーティング・システムが、アドレスを通じたオフセットを伴う入出力機能、移送されることになるデータ、または移送されることになるデータへのポインタ、あるいはその両方、およびデータの長さを少なくとも指定する前記入出力ストア命令(30)を発行すること、

(ii) 前記データ処理ユニット(216)が、前記入出力ストア命令(30)で指定された前記アドレスによって前記入出力機能を識別するように構成されること、

(iii) 前記データ処理ユニット(216)が、アドレス空間およびゲスト・インスタンス・レベルで前記入出力機能へのアクセスが許可されるかどうかを検証するように構成され、前記ゲストが前記データ処理システム(210)上で動くこと、

(iv) 前記データ処理ユニット(216)が、前記システム・ネスト(18)における前記入出力ストア命令(30)の実行が完了する前に、前記入出力ストア命令(30)を完了させるように構成されること、

(v) 前記システム・ファームウェア(10)が、前記入出力ストア命令(30)の非同期実行中に前記データ処理ユニット(216)によってエラーが検出された場合、失敗

10

20

30

40

50

した前記非同期実行の前記データを伝送することを、割込みを通じて前記オペレーティング・システムに通知するように構成されること、

(v i) 前記アレイ管理兼アクセス・ロジック(46)が、前記入出力ストア命令(30)の完了メッセージを収集し、受信した完了メッセージに基づいて前記入出力ステータス・バッファ(24)を更新すること、

(v i i) 前記データ処理ユニット(216)が、待ち状態の入出力ストア命令(30)の完了についての情報をストアするために入出力ステータス・バッファ(24)が利用可能になるまで、前記入出力ストア命令(30)の実行を遅らせることを含む、方法。

【請求項14】

前記データ処理ユニット(216)が、待ち状態の入出力ストア命令(30)の完了についての情報をストアするために入出力ステータス・バッファ(24)が利用可能になるまで、非同期ストア命令(30)または同期ストア命令(30)あるいはその両方の実行を遅らせる、請求項13に記載の方法。

【請求項15】

さらに、前記入出力ステータス・バッファ(24)が、前記システム・ネスト(18)から、または前記入出力バス・コントローラ(20)から、あるいはその両方からメッセージ状態を、具体的には、前記システム・ネスト(18)から完了ステータスを収集し、前記メッセージ状態または前記完了ステータスあるいはその両方が、入出力ステータス・バッファ・インデックスで番号を付けられる、請求項13または14に記載の方法。

【請求項16】

前記システム・ファームウェア(10)が、前記入出力ステータス・アレイ(44)内の入出力ステータス・バッファ(24)をアロケート/デアロケートすること、または新しいストア命令(30)の開始を始めること、あるいはその両方を行うアレイ管理ロジック(42)を備える、請求項15に記載の方法。

【請求項17】

(i) 前記オペレーティング・システムが、前記入出力ストア命令(30)を発行すること、

(i i) 前記システム・ファームウェア(10)が、空いている入出力ステータス・バッファ・インデックスをアロケートし、利用可能な、空いている入出力ステータス・バッファ・インデックスがない場合、空いている入出力ステータス・バッファ・インデックスを待つこと、

(i i i) 前記システム・ファームウェア(10)が、非同期送信エンジンに前記入出力ストア命令(30)を投入し、別の入出力ストア命令によってブロックされた場合、前記入出力ストア命令が完了するまで待つこと、

(i v) 前記データの前記長さに応じて、前記データの長さが8バイトを超える場合、システム・メッセージによって前記データが送信されるまで前記システム・ファームウェア(10)が待つ間、ストア・ブロックの全てのデータが前記集約バッファ(16)に転送されるまで前記集約バッファ(16)にデータ・パケットを送信するために、前記システム・ファームウェア(10)が前記システム・メッセージを繰り返し発行し、そうでなければ、

前記集約バッファ(16)に前記データを送信するために、前記システム・ファームウェア(10)がシステム・メッセージを発行すること、

(v) 前記集約バッファ(16)が完了メッセージを送信するのを待つ間、単一のネスト・メッセージとして前記データを非同期的に前記入出力バス・コントローラ(20)に転送するために、前記システム・ファームウェア(10)が、前記集約バッファ(16)へのシステム・メッセージを発行すること、

(v i) 前記集約バッファ(16)が、前記システム・ネスト(18)に前記ネスト・メッセージを投入することであって、前記集約バッファ(16)が、送信動作の直後、再使用のための空きがある、投入すること、前記システム・ファームウェア(10)にシグ

10

20

30

40

50

ナリングを返すこと、次に、前記集約バッファ（１６）が、再使用のための空きがあるというメッセージを送信すること、

（vii）前記システム・ネスト（１８）が、ターゲット位置に前記メッセージを転送すること、

（viii）前記入出力バス・コントローラ（２０）が、前記メッセージを受信し、データ・フレーム内のデータを前記入出力バスに転送すること、

（ix）前記入出力バス・コントローラ（２０）が、前記システム・ネスト（１８）に完了メッセージを送信すること、

（x）前記システム・ネスト（１８）が、発信元の前記集約バッファ（１６）に前記完了メッセージを転送すること、

（xi）前記集約バッファ（１６）が、前記非同期コア・ネスト・インターフェース（１４）に完了を転送すること、

（xii）前記非同期コア・ネスト・インターフェース（１４）が、前記入出力ステータス・バッファ・インデックスのための前記入出力ステータス・バッファ（２４）に前記完了ステータスをストアし、前記システム・ファームウェア（１０）に動作の完了をシグナリングすること、

（xiii）前記システム・ファームウェア（１０）が、前記入出力ステータス・バッファ・インデックスによって入出力ステータス・バッファのトラッキングを更新すること、

（xiv）エラーの場合、前記システム・ファームウェア（１０）が、前記オペレーティング・システムに欠陥を非同期的にシグナリングすること

をさらに含む、請求項 15 に記載の方法。

#### 【請求項 18】

前記データの前記長さが 8 バイトを超える場合、非同期伝送メカニズムを通じて前記入出力ストア命令（３０）によって、早期完了メッセージとともに複数のデータ・パケットで前記集約バッファ（１６）に前記データをさらに移送する、請求項 13 ないし 17 のいずれか一項に記載の方法。

#### 【請求項 19】

さらに、前記システム・ファームウェア（１０）が、前記入出力ストア命令（３０）をハンドリングするための非同期入出力ドライバ・コード（３２）を使用する、請求項 13 ないし 18 のいずれか一項に記載の方法。

#### 【請求項 20】

さらに、前記コア（１２）が、前記非同期入出力ドライバ・コード（３２）のステータス情報のためのメモリ要件をハンドリングするための非同期セットアップ・コード（３４）を使用する、請求項 19 に記載の方法。

#### 【請求項 21】

さらに、前記非同期コア・ネスト・インターフェース（１４）が、ローカル完了に伴い前記データを転送するための非同期コア・ネスト・インターフェース転送構成要素（３６）を使用する、請求項 13 ないし 20 のいずれか一項に記載の方法。

#### 【請求項 22】

さらに、前記集約バッファ（１６）が、リクエストを送信した後、再使用のための空きがあるというメッセージを配信するための早期完了ロジック（２６）を使用する、請求項 13 ないし 21 のいずれか一項に記載の方法。

#### 【請求項 23】

システム・メッセージが、

- 階層型物理ターゲット・アドレス、
- SMT スレッドまたは集約バッファ識別子の発生源を確認すること、
- データの長さ、
- 入出力バス・アドレス、
- 入出力ステータス・バッファ・インデックス

のうちの 1 つを含む、請求項 13 ないし 22 のいずれか一項に記載の方法。

## 【請求項 24】

請求項 13 ないし 23 のいずれか一項に記載の方法をコンピュータに実行させる、コンピュータ・プログラム。

## 【請求項 25】

請求項 24 に記載の前記コンピュータ・プログラムをコンピュータ可読ストレージ媒体に記録した、ストレージ媒体。

## 【発明の詳細な説明】

## 【技術分野】

## 【0001】

本発明は、一般に、データ処理システムに関し、詳細には、複数の外部デバイスに対する入出力ストア命令をハンドリングする方法、コンピュータ・プログラムおよびデータ処理システムに関する。

## 【背景技術】

## 【0002】

コンピューティング環境は、様々なタイプのアダプタを含む 1 つまたは複数のタイプの入出力デバイスを含むことがある。アダプタの 1 つのタイプは、ペリフェラル・コンポーネント・インターコネクト (PCI: Peripheral Component Interconnect)、またはペリフェラル・コンポーネント・インターコネクト・エクスプレス (PCIe: Peripheral Component Interconnect Express) アダプタである。このアダプタは、アダプタと、アダプタが取り付けられたシステムとの間でデータを通信するのに使用される 1 つまたは複数のアドレス空間を含む。

## 【0003】

いくつかのシステムでは、アダプタに連結された中央処理装置 (CPU: central processing unit) のアドレス空間の一部がアダプタのアドレス空間にマッピングされ、ストレージにアクセスする CPU 命令が、アダプタのアドレス空間にあるデータを直接操作するのを可能にする。

## 【0004】

PCI または PCIe アダプタなどのアダプタとの通信は、アダプタとの間でデータを通信するために特に設計され、通信のために使用される制御命令によって容易にすることができる。

## 【0005】

最先端技術では、アダプタにデータをストアするためのストア命令は、例えば、実行のための機械語命令を取得することを含み、機械語命令は、コンピュータ・アーキテクチャに応じたコンピュータ実行のために定義され、機械語命令は、例えば、アダプタへのストア命令 (a store to adapter instruction) を識別するオペコード・フィールドを含む。第 1 のフィールドは、アダプタにストアされることになるデータを含む第 1 の位置を識別する。第 2 のフィールドは、内容にアダプタを識別する機能ハンドルを含む第 2 の位置、データがストアされることになるアダプタ内のアドレス空間の指名、および、アドレス空間内のオフセットを識別する。機械語命令は実行され、実行することは、機能ハンドルを使用して、アダプタに関連付けられた機能テーブル・エントリを取得することを含む。アダプタのデータ・アドレスは、機能テーブル・エントリ内の情報、およびオフセットのうちの少なくとも 1 つを使用して取得される。データは、アドレス空間の指名によって識別されたアドレス空間における特定の位置における第 1 の位置からストアされ、特定の位置は、アダプタのデータ・アドレスによって識別される。

## 【0006】

大規模マルチプロセッサ・システムにおける既存の特徴は、ターゲット・ゾーン内の全てのプロセッサを休止させる能力である。休止機能は、プロセッサまたはプロセッサのグループの状態を、例えば、システム更新またはバックアップを実施するために、一時的に停止または変化させる働きをする。いくつかの事例では、システム・リソースのサブセットだけに、休止割込が適用可能である。このような事例では、システムは、異なるゾーン

10

20

30

40

50

に分割することができる。１つのゾーン（ターゲット・ゾーン）に適用可能な休止動作について、ターゲット・ゾーンの外側にあるプロセッサは、実行し続けることを許可されるが、新しい変換はブロックされることがある。典型的には、少なくとも１つのシステム・コントローラまたは他のメカニズムが、システム内の全ての物理的なプロセッサに休止をブロードキャストし、休止状態情報の収集をハンドリングし、いつ全てのプロセッサがスタートしたか、または、休止リクエストを無視しているか（フィルタリングしているか）をリクエストしたプロセッサに示す。

#### 【 0 0 0 7 】

休止コントローラは、マルチプロセッサ・システムにおけるプロセッサ、および、休止リクエストを受信するように構成された休止状態マシンに通信連結することができる。コンピュータ・システムは、リクエストしたプロセッサから、休止コントローラで休止リクエストを受信することを含む方法を実施するように構成され、リクエストしたプロセッサは、マルチプロセッサ・システムにおける複数のプロセッサのうちの１つであり、休止状態マシンの状態に基づいて休止リクエストを受け入れないことを決定する。また、方法は、リクエストを受け入れないことに基づいて、休止リクエストを拒絶したことを示すように構成された拒絶メッセージを生成することと、マルチプロセッサ・システムに休止コマンドがブロードキャストされるまで拒絶メッセージを保持することであって、休止コマンドが異なる休止リクエストに基づく、保持することと、休止コマンドのブロードキャストが休止コントローラによって検出されたことに基づいて、リクエストしたプロセッサに拒絶メッセージを送信することを含む。

#### 【 発明の概要 】

#### 【 0 0 0 8 】

入出力ストア命令をハンドリングするためのデータ処理システムを提案し、データ処理システムは、入出力バス・コントローラによって少なくとも１つの入出力バスに通信連結されたシステム・ネストを備える。データ処理システムは、コア、システム・ファームウェア、および非同期コア・ネスト・インターフェースを備えるデータ処理ユニットを少なくともさらに備える。データ処理ユニットは、集約バッファを介してシステム・ネストに通信連結される。システム・ネストは、入出力バスに通信連結された少なくとも１つの外部デバイスからデータを非同期的にロードすること、または少なくとも１つの外部デバイスにデータを非同期的にストアすること、あるいはその両方を行うように構成される。非同期コア・ネスト・インターフェースは、複数の入出力ステータス・バッファを伴う入出力ステータス・アレイ、および、アレイ管理兼アクセス・ロジックを備える。

#### 【 0 0 0 9 】

データ処理システムは、（ i ）データ処理システム上で動くオペレーティング・システムが、アドレスを通じたオフセットを伴う入出力機能、移送されることになるデータ、または移送されることになるデータへのポインタ、あるいはその両方、およびデータの長さを少なくとも指定する入出力ストア命令を発行すること、（ i i ）データ処理ユニットが、入出力ストア命令で指定されたアドレスによって入出力機能を識別するように構成されること、（ i i i ）データ処理ユニットが、アドレス空間およびゲスト・インスタンス・レベルで入出力機能へのアクセスが許可されるかどうかを検証するように構成され、ゲストがデータ処理システム上で動くこと、（ i v ）データ処理ユニットが、システム・ネストにおける入出力ストア命令の実行が完了する前に、入出力ストア命令を完了させるように構成されること、（ v ）システム・ファームウェアが、入出力ストア命令の非同期実行中にデータ処理ユニットによってエラーが検出された場合、失敗した非同期実行のデータを伝送することを、割込みを通じてオペレーティング・システムに通知するように構成されること、（ v i ）アレイ管理兼アクセス・ロジックが、ストア命令の完了を収集し、受信した完了メッセージに基づいて入出力ステータス・バッファを更新すること、（ v i i ）データ処理ユニットが、待ち状態のストア命令の完了についての情報をストアするために入出力ステータス・バッファが利用可能になるまで、ストア命令の実行を遅らせることを実施するように構成される。

## 【 0 0 1 0 】

好ましくは、したがって、繰り返される非同期ストア命令の命令あたりのサイクルを減らすために、複数の未処理の非同期ストア命令を同時に許可することができる。順序付けは、非同期ストア命令と同期ロード／ストア命令との間で定義される。複数の未処理の非同期ストア命令をサポートすることは、複数のステータス・メッセージの帳簿付け、およびステータス・エントリとのレスポンスの相関関係に基づく。

## 【 0 0 1 1 】

本発明の第 1 の実施形態によるデータ処理システムは、データ処理システムの少なくとも 1 つの外部デバイスとの間で入出力バスを介して命令がロードおよびストアを行うことを含む。非同期命令は、外部デバイスにデータがストアされる前に完了し、一方で、同期命令は、外部デバイスにデータがストアされた後に完了する。本明細書で説明される実施形態の中で、P C I が、他の任意の入出力技術と交換可能なように使用されることになり、したがって、本発明の実施形態を P C I に限定しない。

10

## 【 0 0 1 2 】

本発明の実施形態は、アーキテクチャ境界の上から観察できるように、厳密に順序付けられた方式で入出力ストア命令の実行を説明するが、実際の実行は、データ処理ユニット ( C P U ) のハードウェア内で順序が異なってもよい。

## 【 0 0 1 3 】

本発明の実施形態によれば、P C I ストア命令は、P C I e ストア効果の非同期実行、および非同期ステータス・ハンドリングにより実行することができる。非同期高信頼実行は、本発明のデータ処理システムのマイクロアーキテクチャにおける高信頼転送メカニズムに基づく。

20

## 【 0 0 1 4 】

既存の P C I ストアおよびストア・ブロック命令は、通常、P C I ストア・データが P C I e インターフェースに配信され、完了が処理ユニットに返されるポイントまで同期的である。

## 【 0 0 1 5 】

P C I 規格は、非同期送出によりデータを集約するプロセッサにおけるストア・キューを通じて典型的に実装される P C I 情報の非同期送信コマンドしか必要としない。

## 【 0 0 1 6 】

都合のよいことに、本発明の実施形態によれば、命令あたりのサイクルに関する改善は、入出力ストア命令の高信頼非同期送信処理に、同期 P C I 命令を置き替えることによって達成することができる。

30

## 【 0 0 1 7 】

移送されることになるデータの代替または追加として、本発明の 1 つの実施形態によるストア命令は、データを直接含むのではなく、データをフェッチするために使用されるはずのメイン・メモリへのポインタを指定することもできる。

## 【 0 0 1 8 】

ゲスト・インスタンス・レベルは、単一のゲストまたはホストがデータ処理システム上で動いていることができることを意味することもできる。

40

## 【 0 0 1 9 】

入出力機能自体のオフセットのアドレスは、仮想アドレス、物理アドレス、論理アドレスであってもよい。仮想アドレスおよび論理アドレスは、典型的には、メモリ管理ユニット ( M M U ) を通じて物理アドレスに変換され、したがって、どの機能およびオフセットを意味するかを物理アドレスが識別することを可能にする。

## 【 0 0 2 0 】

この文脈での物理アドレスは、「ゲスト／オペレーティング・システム内からアクセス可能なアドレス変換階層における最低アドレス」を意味する。

## 【 0 0 2 1 】

都合のよいことに、入出力ステータス・バッファは、システム・ネストから、または入

50



出力バス・コントローラから、あるいはその両方から返された状態を、具体的には、システム・ネストから完了メッセージを収集することができる。これらの入出力ステータス・バッファは、返された状態を収集することができ、非同期伝送処理をサポートする非同期システム・メッセージ・バッファとして機能する。都合のよいことに、入出力ステータス・バッファは、クイック・レスポンスのために、非同期コア・ネスト・インターフェース内に直接統合することができる。

#### 【 0 0 2 2 】

本発明のデータ処理システムの好ましい実施形態によれば、データ処理ユニットは、待ち状態のストア命令の完了についての情報をストアするために入出力ステータス・バッファが利用可能になるまで、非同期ストア命令または同期ストア命令あるいはその両方の実行を遅らせることができる。したがって、複数の入出力デバイスへの複数のストア命令の順序付け処理は、処理時間を節約する能力を提供する効率的な方式でハンドリングすることができる。

10

#### 【 0 0 2 3 】

本発明のデータ処理システムの好ましい実施形態によれば、入出力ステータス・バッファは、システム・ネストから、または入出力バス・コントローラから、あるいはその両方からメッセージ状態を、具体的には、システム・ネストから完了ステータスを収集することができる。このようにして、異なるストア命令の完了ステータスについての情報は、順序付けられた効率的な手法でハンドリングすることができる。

#### 【 0 0 2 4 】

本発明のデータ処理システムの好ましい実施形態によれば、メッセージ状態または完了ステータスあるいはその両方は、入出力ステータス・バッファ・インデックスで番号を付けることができる。番号付けは、他のストア命令をさらに処理するために順序付けられた効率的な方式で、メッセージを、具体的には、完了状態をハンドリングする可能性をもたらす。

20

#### 【 0 0 2 5 】

本発明のデータ処理システムの好ましい実施形態によれば、集約バッファは、非同期コア・ネスト・インターフェースに非同期バスを介して通信連結することができる。したがって、集約バッファは、外部デバイスに移送されることになる全てのデータが集約バッファにストアされるまで、非同期コア・ネスト・インターフェースによって連続して送信されたデータを直接ハンドリングすることができる。このようにして、非同期コア・ネスト・インターフェースからのデータ転送のための非同期伝送メカニズムをサポートできることが好ましい。

30

#### 【 0 0 2 6 】

本発明のデータ処理システムの好ましい実施形態によれば、データは、ソース・データの長さが 8 バイトを超える場合、非同期伝送メカニズムを通じて入出力ストア命令によって、早期完了メッセージとともに複数のデータ・パケットで集約バッファに移送することができ、そうでなければ、データは、1つのデータ・パケットで移送することができる。送信デバイスは早期の方の状態では再使用のための空きがあるので、非同期伝送メカニズムが好ましい。

40

#### 【 0 0 2 7 】

本発明のデータ処理システムの好ましい実施形態によれば、システム・ファームウェアは、入出力ストア命令をハンドリングするための非同期入出力ドライバ・コードを含むことができる。したがって、非同期伝送メカニズムは、データ処理ユニットから外部デバイスにデータを移送するために使用することができる。

#### 【 0 0 2 8 】

本発明のデータ処理システムの好ましい実施形態によれば、コアは、非同期入出力ドライバ・コードのステータス情報のためのメモリ要件をハンドリングするための非同期セットアップ・コードを備えることができる。この非同期セットアップ・コードは、システム・ネストおよび入出力バス・コントローラへの集約バッファを通じた非同期伝送メカニズ

50

ムをさらに容易にすることができる。

【 0 0 2 9 】

本発明のデータ処理システムの好ましい実施形態によれば、非同期コア・ネスト・インターフェースは、ローカル完了 (local completion) に伴いデータを転送するための非同期コア・ネスト・インターフェース転送構成要素を備えることができる。この構成要素は、非同期コア・ネスト・インターフェース内のハードウェアに実装することができる。したがって、データ・パケットでデータを集約バッファに送信するための好ましい非同期伝送モードをサポートすることができる。

【 0 0 3 0 】

本発明のデータ処理システムの好ましい実施形態によれば、集約バッファは、リクエストを送信した後、再使用のための空きがあるというメッセージ (free for reuse message) を配信するための早期完了ロジックを備えることができる。これは、システム・ネストおよび入出力バス・コントローラへの集約バッファを介したデータの伝送処理の早期継続を可能にする。

10

【 0 0 3 1 】

本発明のデータ処理システムの好ましい実施形態によれば、システム・ファームウェアは、入出力ステータス・アレイ内の入出力ステータス・バッファをアロケート/デアロケートすること、または新しいストア命令の開始を始めること、あるいはその両方を行うアレイ管理ロジックを備えることができる。したがって、非稼働のステータス・バッファは、さらなるストア命令に属させることができる。ストア命令の順序付け処理は、効率的かつ時間を節約する方式でハンドリングすることができる。

20

【 0 0 3 2 】

本発明のデータ処理システムの好ましい実施形態によれば、システム・メッセージは、  
- 階層型物理ターゲット・アドレス、  
- S M T (同時マルチスレッド: simultaneous multithreading) スレッドもしくは集約バッファ識別子の発生源を確認すること、  
- データの長さ、入出力バス・アドレス、または  
- 入出力ステータス・バッファ・インデックスのうちの1つを含むことができる。したがって、データ処理システムを通じた関係情報の有利な受け渡しを保証することができる。

【 0 0 3 3 】

さらに、データ処理システムの少なくとも1つの外部デバイスへの入出力ストア命令をハンドリングするための方法を提案し、データ処理システムは、入出力バス・コントローラによって少なくとも1つの入出力バスに通信連結されたシステム・ネストを備える。データ処理システムは、コア、システム・ファームウェア、および非同期コア・ネスト・インターフェースを備えるデータ処理ユニットを少なくともさらに備える。データ処理ユニットは、集約バッファを介してシステム・ネストに通信連結される。外部デバイスは、入出力バスに通信連結される。非同期コア・ネスト・インターフェースは、複数の入出力ステータス・バッファを伴う入出力ステータス・アレイ、および、アレイ管理兼アクセス・ロジックを備える。

30

【 0 0 3 4 】

方法は、  
( i ) データ処理システム上で動くオペレーティング・システムが、アドレスを通じたオフセットを伴う入出力機能、移送されることになるデータ、または移送されることになるデータへのポインタ、あるいはその両方、およびデータの長さを少なくとも指定する入出力ストア命令を発行すること、  
( i i ) データ処理ユニットが、入出力ストア命令で指定されたアドレスによって入出力機能を識別するように構成されること、  
( i i i ) データ処理ユニットが、アドレス空間およびゲスト・インスタンス・レベルで入出力機能へのアクセスが許可されるかどうかを検証するように構成され、ゲストがデータ処理システム上で動くこと、  
( i v ) データ処理ユニットが、システム・ネストにおける入出力ストア命令の実行が完了する前に、入出力ストア命令を完了させるように構成されること、  
( v ) システム・ファームウェアが、入出力ストア命令の非同期実行中にデータ処理ユニットによってエラーが検出された場合、失敗した非同期実行のデータを伝送すること

40

50

を、割込みを通じてオペレーティング・システムに通知するように構成されること、(v i) アレイ管理兼アクセス・ロジックが、ストア命令の完了を収集し、受信した完了メッセージに基づいて入出力ステータス・バッファを更新すること、および(v i i) データ処理ユニットが、待ち状態のストア命令の完了についての情報をストアするために入出力ステータス・バッファが利用可能になるまで、ストア命令の実行を遅らせることを含む。

【0035】

好ましくは、したがって、繰り返される非同期ストア命令の命令あたりのサイクルを減らすために、複数の未処理の非同期ストア命令を同時に許可することができる。順序付けは、非同期ストア命令と同期ロード/ストア命令との間で定義される。複数の未処理の非同期ストア命令をサポートすることは、複数のステータス・メッセージの帳簿付け、およびステータス・エントリとのレスポンスの相関関係に基づく。

10

【0036】

本発明のさらなる実施形態による方法は、データ処理システムの外部デバイスとの間で入出力バスを介して命令がロードおよびストアを行うことを含む。非同期命令は、外部デバイスにデータがストアされる前に完了し、一方で、同期命令は、外部デバイスにデータがストアされた後に完了する。本明細書で説明される実施形態の中で、P C I が、他の任意の入出力技術と交換可能なように使用されることになり、したがって、本発明の実施形態をP C I に限定しない。

【0037】

本発明の方法の実施形態は、アーキテクチャ境界の上から観察できるように、厳密に順序付けられた方式で入出力ストア命令の実行を説明するが、実際の実行は、データ処理ユニット(C P U)のハードウェア内で順序が異なってもよい。

20

【0038】

本発明の方法の実施形態によれば、P C I ストア命令は、P C I e ストア効果の非同期実行、および非同期ステータス・ハンドリングにより実行することができる。非同期高信頼実行は、本発明のデータ処理システムのマイクロアーキテクチャにおける高信頼転送メカニズムに基づく。

【0039】

既存のP C I ストアおよびストア・ブロック命令は、通常、P C I ストア・データがP C I e インターフェースに配信され、完了が処理ユニットに返されるポイントまで同期的である。

30

【0040】

P C I 規格は、非同期送出によりデータを集約するプロセッサにおけるストア・キューを通じて典型的に実装されるP C I 情報の非同期送信コマンドしか必要としない。

【0041】

都合のよいことに、本発明の方法の実施形態によれば、命令あたりのサイクルに関する改善は、入出力ストア命令の高信頼非同期送信処理に、同期P C I 命令を置き替えることによって達成することができる。

【0042】

移送されることになるデータの代替または追加として、本発明の1つの実施形態によるストア命令は、データを直接含むのではなく、データをフェッチするために使用されるはずのメイン・メモリへのポインタを指定することもできる。

40

【0043】

ゲスト・インスタンス・レベルは、単一のゲストまたはホストがデータ処理システム上で動いていることができることを意味することもできる。

【0044】

入出力機能自体のオフセットのアドレスは、仮想アドレス、物理アドレス、論理アドレスであることが可能である。仮想アドレスおよび論理アドレスは、典型的には、メモリ管理ユニット(M M U)を通じて物理アドレスに変換され、したがって、どの機能およびオフセットを意味するかを物理アドレスが識別することを可能にする。

50

## 【 0 0 4 5 】

この文脈での物理アドレスは、「ゲスト/オペレーティング・システム内からアクセス可能なアドレス変換階層における最低アドレス」を意味する。

## 【 0 0 4 6 】

本発明の方法の好ましい実施形態によれば、データ処理ユニットは、待ち状態のストア命令の完了についての情報をストアするために入出力ステータス・バッファが利用可能になるまで、非同期ストア命令または同期ストア命令あるいはその両方の実行を遅らせることができる。したがって、複数の入出力デバイスへの複数のストア命令の順序付け処理は、処理時間を節約する能力を提供する効率的な方式でハンドリングすることができる。

## 【 0 0 4 7 】

本発明の方法の好ましい実施形態によれば、入出力ステータス・バッファは、システム・ネストから、または入出力バス・コントローラから、あるいはその両方からメッセージ状態を、具体的には、システム・ネストから完了ステータスを収集することができ、メッセージ状態または完了ステータスあるいはその両方は、入出力ステータス・バッファ・インデックスで番号を付けられる。このようにして、異なるストア命令の完了ステータスについての情報は、順序付けられた効率的な手法でハンドリングすることができる。番号付けは、他のストア命令をさらに処理するために順序付けられた効率的な方式で、メッセージを、具体的には、完了状態をハンドリングする可能性をもたらす。

## 【 0 0 4 8 】

本発明の方法の好ましい実施形態によれば、システム・ファームウェアは、入出力ステータス・アレイ内の入出力ステータス・バッファをアロケート/デアロケートすること、または新しいストア命令の開始を始めること、あるいはその両方を行うアレイ管理ロジックを備えることができる。したがって、非稼働のステータス・バッファは、さらなるストア命令に属させることができる。ストア命令の順序付け処理は、効率的かつ時間を節約する方式でハンドリングすることができる。

## 【 0 0 4 9 】

好ましい実施形態によれば、方法は、( i ) オペレーティング・システムが、入出力ストア命令を発行すること、( i i ) システム・ファームウェア ( 1 0 ) が、空いている入出力ステータス・バッファ・インデックスをアロケートし、利用可能な、空いている入出力ステータス・バッファ・インデックスがない場合、空いている入出力ステータス・バッファ・インデックスを待つこと、( i i i ) システム・ファームウェアが、非同期送信エンジンにストア命令を投入し、別のストア命令によってブロックされた場合、ストア命令が完了するまで待つこと、( i v ) データの長さに応じて、データの長さが 8 バイトを超える場合、システム・メッセージによってデータが送信されるまでシステム・ファームウェアが待つ間、ストア・ブロックの全てのデータが集約バッファに転送されるまで集約バッファにデータ・パケットを送信するために、システム・ファームウェアがシステム・メッセージを繰り返し発行し、そうでなければ、データの長さにかかわらず、集約バッファにデータを送信するために、システム・ファームウェアがシステム・メッセージを発行すること、( v ) 集約バッファが完了メッセージを送信するのを待つ間、単一のネスト・メッセージとしてデータを非同期的に入出力バス・コントローラに転送するために、システム・ファームウェアが、集約バッファへのシステム・メッセージを発行すること、( v i ) 集約バッファが、システム・ネストにネスト・メッセージを投入することであって、集約バッファが、送信動作の直後、再使用のための空きがある、投入すること、システム・ファームウェアにシグナリングを返すこと、次に、集約バッファが、再使用のための空きがあるというメッセージを送信すること、( v i i ) システム・ネストが、ターゲット位置にメッセージを転送すること、( v i i i ) 入出力バス・コントローラが、メッセージを受信し、データ・フレーム内のデータを入出力バスに転送すること、( i x ) 入出力バス・コントローラが、システム・ネストに完了メッセージを送信すること、( x ) システム・ネストが、発信元の集約バッファに完了メッセージを転送すること、( x i ) 集約バッファが、非同期コア・ネスト・インターフェースに完了を転送すること、( x i i ) 非

10

20

30

40

50

同期コア・ネスト・インターフェースが、入出力ステータス・バッファ・インデックス用の入出力ステータス・バッファに完了ステータスをストアし、システム・ファームウェアに動作の完了をシグナリングすること、( x i i i ) システム・ファームウェアが、入出力ステータス・バッファ・インデックスによって入出力ステータス・バッファのトラッキングを更新すること、および( x i v ) エラーの場合、システム・ファームウェアが、オペレーティング・システムに欠陥を非同期的にシグナリングすることをさらに含むことができる。

【 0 0 5 0 】

ステップ( i i ) だけが、データの長さに依存し、データの長さが 8 バイトを超えることと、データの長さが 8 バイトを超えないことが異なる。

10

【 0 0 5 1 】

本発明の方法の実施形態によれば、データは、ストア・ブロックの全てのデータが集約バッファに転送されるまで、スライスで集約バッファに伝送され、システム・ファームウェアは、非同期コア・ネスト・インターフェースによってデータが送信されるまで待っている。

【 0 0 5 2 】

したがって、データが 8 バイトより小さい場合、データ・パケットをスライスで集約バッファに満たす処理は、スキップすることができ、外部デバイスにデータを伝送する処理は、単一のステップで完了させることができる。

【 0 0 5 3 】

20

本発明の方法の好ましい実施形態によれば、データは、データの長さが 8 バイトを超える場合、非同期伝送メカニズムを通じて入出力ストア命令によって、早期完了メッセージとともに複数のデータ・パケットで集約バッファに移送することができる。送信デバイスは早期の方の状態では再使用のための空きがあるので、非同期伝送メカニズムが好ましい。

【 0 0 5 4 】

本発明の方法の好ましい実施形態によれば、システム・ファームウェアは、入出力ストア命令をハンドリングするための非同期入出力ドライバ・コードを使用することができる。したがって、非同期伝送メカニズムは、データ処理ユニットから外部デバイスにデータを移送するために使用することができる。

【 0 0 5 5 】

30

本発明の方法の好ましい実施形態によれば、コアは、非同期入出力ドライバ・コードのステータス情報用のメモリ要件をハンドリングするための非同期セットアップ・コードを使用することができる。この非同期セットアップ・コードは、システム・ネストおよび入出力バス・コントローラへの集約バッファを通じた非同期伝送メカニズムをさらに容易にすることができる。

【 0 0 5 6 】

都合のよいことに、非同期コア・ネスト・インターフェースは、ローカル完了に伴いデータを転送するための非同期コア・ネスト・インターフェース転送構成要素を使用することができる。したがって、データ・パケットでデータを集約バッファに送信するための好ましい非同期伝送モードをサポートすることができる。

40

【 0 0 5 7 】

都合のよいことに、集約バッファは、リクエストを送信した後、再使用のための空きがあるというメッセージを配信するための早期完了ロジックを使用することができる。これは、システム・ネストおよび入出力バス・コントローラへの集約バッファを介したデータの伝送処理の早期継続を可能にする。

【 0 0 5 8 】

都合のよいことに、入出力ステータス・バッファは、システム・ネストから、または入出力バス・コントローラから、あるいはその両方から返された状態を、具体的には、システム・ネストから完了メッセージを収集することができる。これらの入出力ステータス・バッファは、返された状態を収集することができ、非同期伝送処理をサポートする非同期

50

システム・メッセージ・バッファとして機能する。

【 0 0 5 9 】

本発明の方法の好ましい実施形態によれば、システム・メッセージは、 - 階層型物理ターゲット・アドレス、 - S M Tスレッドもしくは集約バッファ識別子の発生源を確認すること、 - データの長さ、 - 入出力バス・アドレス、または - 入出力ステータス・バッファ・インデックスのうちの1つを含むことができる。したがって、データ処理システムを通じた関係情報の有利な受け渡しを保證することができる。

【 0 0 6 0 】

さらに、データ処理システムの少なくとも1つの外部デバイスへの入出力ストア命令をハンドリングするための好都合なコンピュータ・プログラム製品を提案し、データ処理システムは、入出力バス・コントローラによって少なくとも1つの入出力バスに通信連結されたシステム・ネストを備える。データ処理システムは、コア、システム・ファームウェア、および非同期コア・ネスト・インターフェースを備えるデータ処理ユニットを少なくともさらに備える。データ処理ユニットは、集約バッファを介してシステム・ネストに通信連結される。外部デバイスは、入出力バスに通信連結される。非同期コア・ネスト・インターフェースは、複数の入出力ステータス・バッファを伴う入出力ステータス・アレイ、および、アレイ管理兼アクセス・ロジックを備える。

【 0 0 6 1 】

コンピュータ・プログラム製品は、プログラム命令を含むコンピュータ可読ストレージ媒体を備え、プログラム命令は、( i ) データ処理システム上で動くオペレーティング・システムが、アドレスを通じたオフセットを伴う入出力機能、移送されることになるデータ、または移送されることになるデータへのポインタ、あるいはその両方、およびデータの長さを少なくとも指定する入出力ストア命令を発行すること、( i i ) データ処理ユニットが、入出力ストア命令で指定されたアドレスによって入出力機能を識別するように構成されること、( i i i ) データ処理ユニットが、アドレス空間およびゲスト・インスタンス・レベルで入出力機能へのアクセスが許可されるかどうかを検証するように構成され、ゲストがデータ処理システム上で動くこと、( i v ) データ処理ユニットが、システム・ネストにおける入出力ストア命令の実行が完了する前に、入出力ストア命令を完了させるように構成されること、( v ) システム・ファームウェアが、入出力ストア命令の非同期実行中にデータ処理ユニットによってエラーが検出された場合、失敗した非同期実行のデータを伝送することを、割込みを通じてオペレーティング・システムに通知するように構成されること、( v i ) アレイ管理兼アクセス・ロジックが、ストア命令の完了を収集し、受信した完了メッセージに基づいて入出力ステータス・バッファを更新すること、( v i i ) データ処理ユニットが、待ち状態のストア命令の完了についての情報をストアするために入出力ステータス・バッファが利用可能になるまで、ストア命令の実行を遅らせることを含む方法をコンピュータ・システムに実施させるようにコンピュータ・システムによって実行可能である。

【 0 0 6 2 】

さらに、上記で説明した方法を実施するためのコンピュータ可読プログラム命令を含むデータ処理プログラムの実行のためのデータ処理システムを提案する。

【 0 0 6 3 】

上述および他の目的および長所とともに本発明は、実施形態の以下の詳細な説明から最も良く理解することができるが、実施形態に限定されない。

【図面の簡単な説明】

【 0 0 6 4 】

【図 1】本発明の1つの実施形態による、外部デバイスへの入出力ストア命令をハンドリングするためのデータ処理システムのブロック図である。

【図 2】本発明の1つの実施形態による、外部デバイスへの入出力ストア命令をハンドリングするための方法のメッセージ・シーケンス図である。

【図 3】本発明の1つの実施形態による、外部デバイスへの入出力ストア命令をハンドリ

10

20

30

40

50

ングするための流れ図の第 1 の部分である。

【図 4】本発明の 1 つの実施形態による、外部デバイスへの入出力ストア命令をハンドリングするための流れ図の第 2 の部分である。

【図 5】本発明による方法を実行するためのデータ処理システムの実施形態の例を示す図である。

【発明を実施するための形態】

【0065】

図面では、同様の要素は、等しい参照番号で参照される。図面は、概略図にすぎず、本発明の特定の範囲を表現することを意図するものではない。その上、図面は、本発明の典型的な実施形態だけを描写することを意図し、したがって、本発明の範囲を限定するものとみなされるべきではない。

【0066】

本明細書で説明される例証的实施形態は、入出力ストア命令をハンドリングするためのデータ処理システムおよび方法を提供し、入出力バス・コントローラによって少なくとも 1 つの入出力バスに通信連結されたシステム・ネストを備える。データ処理システムは、コア、システム・ファームウェア、および非同期コア・ネスト・インターフェースを備えるデータ処理ユニットを少なくともさらに備える。データ処理ユニットは、集約バッファを介してシステム・ネストに通信連結される。システム・ネストは、入出力バスに通信連結された外部デバイスからデータを非同期的にロードすること、または外部デバイスにデータを非同期的にストアすること、あるいはその両方を行うように構成される。非同期コア・ネスト・インターフェースは、複数の入出力ステータス・バッファを伴う入出力ステータス・アレイ、および、アレイ管理兼アクセス・ロジックを備える。

【0067】

例証的实施形態は、( i ) データ処理システム上で動くオペレーティング・システムが、アドレスを通じたオフセットを伴う入出力機能、移送されることになるデータ、または移送されることになるデータへのポインタ、あるいはその両方、およびデータの長さを少なくとも指定する入出力ストア命令を発行すること、( i i ) データ処理ユニットが、入出力ストア命令で指定されたアドレスによって入出力機能を識別するように構成され、( i i i ) データ処理ユニットが、アドレス空間およびゲスト・インスタンス・レベルで入出力機能へのアクセスが許可されるかどうかを検証するように構成され、ゲストがデータ処理システム上で動くこと、( i v ) データ処理ユニットが、システム・ネストにおける入出力ストア命令の実行が完了する前に、入出力ストア命令を完了させるように構成され、( v ) システム・ファームウェアが、入出力ストア命令の非同期実行中にデータ処理ユニットによってエラーが検出された場合、失敗した非同期実行のデータを伝送することを、割込みを通じてオペレーティング・システムに通知するように構成され、( v i ) アレイ管理兼アクセス・ロジックが、ストア命令の完了を収集し、受信した完了メッセージに基づいて入出力ステータス・バッファを更新すること、および( v i i ) データ処理ユニットが、待ち状態のストア命令の完了についての情報をストアするために入出力ステータス・バッファが利用可能になるまで、ストア命令の実行を遅らせることを含む方法のために使用することができる。

【0068】

移送されることになるデータの代替または追加として、本発明の 1 つの実施形態によるストア命令は、データを直接含むのではなく、データをフェッチするために使用されるはずのメイン・メモリへのポインタを指定することもできる。

【0069】

ゲスト・インスタンス・レベルは、単一のゲストまたはホストがデータ処理システム上で動いていることができることを意味することでもある。

【0070】

入出力機能自体のオフセットのアドレスは、仮想アドレス、物理アドレス、論理アドレスであることが可能である。仮想アドレスおよび論理アドレスは、典型的には、メモリ管

10

20

30

40

50

理ユニット（MMU）を通じて物理アドレスに変換され、したがって、どの機能およびオフセットを意味するかを物理アドレスが識別することを可能にする。

【0071】

この文脈での物理アドレスは、「ゲスト/オペレーティング・システム内からアクセス可能なアドレス変換階層における最低アドレス」を意味する。

【0072】

図1は、本発明の1つの実施形態による、少なくとも1つの外部デバイス214への入出力ストア命令30をハンドリングするためのデータ処理システム210のブロック図を描写する。データ処理システム210は、入出力バス・コントローラ20によって入出力バス22に通信連結されたシステム・ネスト18を備え、データ処理ユニット216は、

10

【0073】

データ処理ユニット216は、集約バッファ16を介してシステム・ネスト18に通信連結される。システム・ネスト18は、システム・ネスト18の一部としてのバッファ入出力バス・コントローラ・インターフェース28、および入出力バス・コントローラ20を介して入出力バス22に通信連結された外部デバイス214からデータを非同期的にロードすること、または、外部デバイス214にデータを非同期的にストアすること、あるいはその両方を行うように構成される。

20

【0074】

集約バッファ16は、非同期コア・ネスト・インターフェース14に通信可能に接続されている。システム・ファームウェア10は、入出力ストア命令30をハンドリングするための非同期入出力ドライバ・コード32を備える。コア12は、非同期入出力ドライバ・コード32のステータス情報用のメモリ要件をハンドリングするための非同期セットアップ・コード34を備える。非同期コア・ネスト・インターフェース14は、ローカル完了に伴いデータを転送するための非同期コア・ネスト・インターフェース転送構成要素36を備える。集約バッファ16は、リクエストを送信した後、再使用のための空きがあるというメッセージを配信するための早期完了ロジック26を備える。集約バッファ16は、非同期コア・ネスト・インターフェース14に非同期バス38を介して連結される。非同期コア・ネスト・インターフェース14は、複数の入出力ステータス・バッファ24を有する入出力ステータス・アレイ44、および、アレイ管理兼アクセス・ロジック46を備える。入出力ステータス・バッファ24は、システム・ネスト18および/または入出力バス・コントローラ20から、特にシステム・ネスト18からの完了メッセージから返された状態を収集する。入出力ステータス・バッファ24は、非同期コア・ネスト・インターフェース14に直接統合される。例えば、入出力ステータス・バッファ24のうちの1つへの完了メッセージといった、アレイ・エントリの識別を伴うメッセージ48は、システム・ネスト18によって受信することができる。

30

【0075】

発明の方法の1つの実施形態によれば、データ処理システム210上で動くオペレーティング・システムは、アドレスを通じたオフセットを伴う入出力機能、移送されることになるデータ、または移送されることになるデータへのポインタ、あるいはその両方、およびデータの長さを少なくとも指定する入出力ストア命令30を発行する。データ処理ユニット216は、このようにして、入出力ストア命令30によって指定されたアドレスによって入出力機能を識別するように構成される。データ処理ユニット216は、アドレス空間およびゲスト・インスタンス・レベルで入出力機能へのアクセスが許可されるかどうかを検証するように構成され、ゲストは、データ処理システム210上で動く。データ処理ユニット216は、システム・ネスト18における入出力ストア命令30の実行が完了する前に、入出力ストア命令30を完了させるように構成される。システム・ファームウェア10は、入出力ストア命令30の非同期実行中にデータ処理ユニット216によってエ

40

50



ラーが検出された場合、失敗した非同期実行のデータを伝送することを、割込みを通じてオペレーティング・システムに通知するように構成される。

【 0 0 7 6 】

アレイ管理兼アクセス・ロジック 4 6 は、ストア命令 3 0 の完了を収集し、受信した完了メッセージに基づいて入出力ステータス・バッファ 2 4 を更新する。データ処理ユニット 2 1 6 は、待ち状態のストア命令 3 0 の完了についての情報をストアするために入出力ステータス・バッファ 2 4 が利用可能になるまで、ストア命令 3 0 の実行を遅らせる。具体的には、データ処理ユニット 2 1 6 は、待ち状態のストア命令 3 0 の完了についての情報をストアするために入出力ステータス・バッファ 2 4 が利用可能になるまで、非同期ストア命令 3 0 または同期ストア命令 3 0 あるいはその両方の実行を遅らせる。

10

【 0 0 7 7 】

入出力ステータス・バッファ 2 4 は、システム・ネスト 1 8 から、または入出力バス・コントローラ 2 0 から、あるいはその両方からメッセージ状態を、具体的には、システム・ネスト 1 8 から完了ステータスを収集する。メッセージ状態または完了ステータスあるいはその両方は、入出力ステータス・バッファ・インデックスで番号を付けることができることが好ましい。

【 0 0 7 8 】

システム・ファームウェア 1 0 は、入出力ステータス・アレイ 4 4 内の入出力ステータス・バッファ 2 4 をアロケート/デアロケートすること、または新しいストア命令 3 0 の開始を始めること、あるいはその両方を行うアレイ管理ロジック 4 2 を備える。

20

【 0 0 7 9 】

入出力ストア命令 3 0 は、システム・ハードウェア/ファームウェア 5 0 をユーザ側 4 0 と分離するアーキテクチャ境界を横切ったユーザ・インターフェース 4 0 の側のデータ処理システム 2 1 0 内にある。

【 0 0 8 0 】

したがって、データは、ソース・データの長さが 8 バイトを超える場合、非同期伝送メカニズムを通じて入出力ストア命令 3 0 によって、早期完了メッセージとともに複数のデータ・パケットで集約バッファ 1 6 に移送され、そうでなければ、データは、1 つのデータ・パケットで移送される。

【 0 0 8 1 】

30

本発明のデータ処理システムの 1 つの実施形態によるシステム・メッセージは、階層型物理ターゲット・アドレス、SMT スレッドもしくは集約バッファ識別子の発生源を確認すること、データの長さ、入出力バス・アドレス、または入出力ステータス・バッファ・インデックスのうちの 1 つを含む。

【 0 0 8 2 】

複数の外部デバイス 2 1 4 へのストア命令 3 0 をハンドリングするためのキューイングおよび順序付けの意味は、都合がよいことに、以下で説明するように実施することができる。入出力機能関係に対する個々の SMT スレッドについて、全ての旧式の入出力ロード/ストア動作は、プロセッサ・ユニット 2 1 6 の単一のスレッドに対して順序付けることができる。新しい入出力ストア命令は、互いに完全に順序付けられない。新しい入出力ストア命令は、旧式の入出力命令に対して順序付けられる。異なる入出力機能に対する全ての入出力命令は、互いに対して順序付けられない。

40

【 0 0 8 3 】

図 2 は、本発明の 1 つの実施形態による、外部デバイス 2 1 4 への入出力ストア命令 3 0 をハンドリングするための方法のメッセージ・シーケンス図を描写する。

【 0 0 8 4 】

図 2 に示したように、方法は、オペレーティング・システムが入出力ストア命令 3 0 を発行することで開始する。ステップ S 1 0 1 において、システム・ファームウェア 1 0 は、空いている入出力ステータス・バッファ・インデックスをアロケートする。利用可能な、空いている入出力ステータス・バッファ・インデックスがない場合、システム・ファーム

50

ムウェア 10 は待つ。ステップ S 103 において、システム・ファームウェア 10 は、非同期送信エンジンにストア命令を投入できるかチェックする。これが可能であれば、処理は続く。これが可能でなければ、遅延の原因となるストア命令が完了するまで、ストア命令を遅らせる。

【0085】

次に、ステップ S 100 および S 104 において示したように、システム・ファームウェア 10 は、データの長さが 8 バイトを超える場合、ストア・ブロックの全てのデータが集約バッファ 16 に転送されるまで、集約バッファ 16 にデータ・パケットを送信するためにシステム・メッセージを繰り返し発行し、一方で、システム・ファームウェア 10 は、システム・メッセージでデータが送信されるまで待つ。ステップ S 102 および S 106 において、ローカル完了メッセージが、システム・ファームウェア 10 に送り返される。

10

【0086】

次に、ステップ S 108 において、システム・ファームウェア 10 は、単一のネスト・メッセージとしてデータを非同期的に入出力バス・コントローラ 20 に転送するために、集約バッファ 16 にシステム・メッセージを発行し、一方で、集約バッファ 16 が完了メッセージを送信するのを待つ。

【0087】

次に、ステップ S 110 において、集約バッファ 16 は、システム・ネスト 18 にネスト・メッセージを投入し、送信動作の直後、集約バッファ 16 は再使用のための空きがあり、システム・ファームウェア 10 にシグナリングを返す。次に、集約バッファ 16 は、再使用のための空きがあるというメッセージを送信する。

20

【0088】

ステップ S 112 において、システム・ネスト 18 は、ターゲット位置にメッセージを転送し、その後、ステップ S 114 が続き、入出力バス・コントローラ 20 がメッセージを受信し、データ・フレーム内のデータを入出力バスに転送し、その後、ステップ S 116 において、入出力バス・コントローラ 20 が、システム・ネスト 18 に完了メッセージを送信する。

【0089】

次に、ステップ S 118 において、システム・ネスト 18 は、発信元の集約バッファ 16 に完了メッセージを転送し、その後、ステップ S 120 において、集約バッファ 16 が、非同期コア・ネスト・インターフェース 14 に完了を転送する。次に、ステップ S 122 において、非同期コア・ネスト・インターフェース 14 は、それぞれの入出力ステータス・バッファ・インデックス用の入出力バッファ 24 にステータスをストアし、システム・ファームウェア 10 に動作の完了をシグナリングする。最後に、ステップ S 123 において、システム・ファームウェア 10 は、入出力ステータス・バッファ・インデックスによって入出力ステータス・バッファ 24 のトラッキングを更新する。入出力ステータス・バッファ 24 は、現在、再び空いている。

30

【0090】

データ転送中にエラーが発生した場合、システム・ファームウェア 10 は、オペレーティング・システムに欠陥を非同期的にシグナリングする。

40

【0091】

念のため、移送されることになるデータは 8 バイトより小さく、集約バッファ 16 を繰り返し満たすことはスキップされる。

【0092】

図 3 は、本発明の 1 つの実施形態による、外部デバイス 214 への入出力ストア命令 30 をハンドリングするための流れ図の第 1 の部分を描写し、その一方で、図 4 は、流れ図の第 2 の部分を描写する。

【0093】

データ処理ユニットのシステム・ファームウェアは、ステップ S 200 において開始する。ステップ S 202 において、システム・ファームウェアは、旧式の入出力ストア・ブ

50

ロック命令を、メッセージを介して受信する。ステップ S 2 0 8 において、非同期レスポンスの有無にかかわらず未処理の新しいストア命令 3 0 があるかをチェックする。この場合、コア・ネスト・インターフェース 1 4 がこれ以上未処理の非同期レスポンスを有さなくなるまで待つ。そうでない場合、処理は、コア・ネスト・インターフェース 1 4 が旧式の入出力ストア命令 3 0 をメッセージとして送信することによってステップ S 2 1 2 をすぐに続ける。次に、システムは、ステップ S 2 1 6 における旧式の応答メッセージを受信するまで待っている（ステップ S 2 1 4）。次に、旧式のストア命令 3 0 は、ステップ S 2 1 8 においてメッセージを終了し、システム・ファームウェア 1 0 は、ステップ S 2 2 0 において処理を終える。

【 0 0 9 4 】

同時に、システム・ファームウェア 1 0 は、ステップ S 2 0 4 において、新しい入出力ストア命令 3 0 をメッセージで受信する。ステップ S 2 2 2 において、空いているステータス・スロット、すなわち、入出力ステータス・バッファ 2 4 が利用可能であるかチェックする。この場合、スロットは、ステップ S 2 2 4 において、使用済としてマークされ、ストア処理は、図 4 で描写した流れ図の第 2 の部分に描写した接続点 A を続ける。そうでない場合、ステップ S 2 2 6 において、コア・ネスト・インターフェース 1 4 が利用可能な空きスロットを有するまで待つ。

【 0 0 9 5 】

ステップ S 2 0 6 において、非同期実行完了メッセージを受信することができ、その後、ステップ S 2 2 8 において、それぞれの入出力ステータス・バッファ・インデックスによってスロットのトラッキングを更新する。次に、システム・ファームウェア 1 0 は、ステップ S 2 3 0 において処理を終える。

【 0 0 9 6 】

流れ図の第 2 の部分は、接続点 A で始まり、図 4 に描写する。まず、ステップ S 3 0 4 において、9 バイト以上が移送されることになるかチェックする。この場合、コア・ネスト・インターフェースは、ステップ S 3 0 6 において、16 バイトまでのメッセージで集約バッファを満たす。システム・ファームウェアは、ステップ S 3 1 0 においてローカル完了のメッセージが送信されるまで待ち（ステップ S 3 0 8）、ステップ S 3 0 4 に戻る。ステップ S 3 0 4 のチェックで 8 バイト未満が残っている場合、フローは、コア・ネスト・インターフェースが非同期入出力メッセージを送信するステップ S 3 1 2 に続き、その後、ステップ S 3 1 4 において、ステップ S 3 1 6 におけるバッファ・レスポンスを待つ。次に、ステップ S 3 1 8 において、終了ストア・ブロック命令を実行し、フローは、ステップ S 3 2 0 において終わり、システム・ファームウェアを終える。

【 0 0 9 7 】

ステップ S 3 2 8 において、非同期コア・ネスト・インターフェース・ロジックは、アウトバウンド処理ループを開始し、その後、ステップ S 3 2 2 において、集約バッファ完了メッセージを受信し、ステップ S 3 2 4 において、集約バッファにデータ・メッセージを転送し、その後、ステップ S 3 2 6 において、システム・ファームウェアに完了メッセージを送り返す。ステップ S 3 3 0 において、非同期入出力送信メッセージを受信し、その後、集約バッファに入出力送信メッセージを転送する。

【 0 0 9 8 】

ステップ S 3 3 8 において、集約バッファ・ロジックは、アウトバウンド処理ループを開始し、その後、ステップ S 3 3 4 においてデータを受信し、ステップ S 3 3 6 において、集約バッファにデータを集約する。集約バッファは、ステップ S 3 4 0 でも入出力送信メッセージを受信し、その後、ステップ S 2 4 2 において、入出力送信メッセージとともに集約バッファからデータを転送する。次に、ステップ S 3 4 4 において、集約バッファからの応答メッセージを、コア・ネスト・インターフェースを介してシステム・ファームウェアに送信する。

【 0 0 9 9 】

次に、図 5 を参照すると、データ処理システム 2 1 0 の例の概略図を示す。データ処理

10

20

30

40

50

システム 210 は、適切なデータ処理システムの 1 つの例にすぎず、本明細書で説明される本発明の実施形態の用途または機能の範囲について何らかの限定を提案することを意図するものではない。いずれにしても、データ処理システム 210 は、本明細書で上記に示した機能のいずれかの実装または実施あるいはその両方を行うことができる。

#### 【0100】

データ処理システム 210 には、コンピュータ・システム / サーバ 212 があり、コンピュータ・システム / サーバ 212 は、非常に多くの他の汎用または専用コンピューティング・システム環境または構成で動作するものである。コンピュータ・システム / サーバ 212 とともに使用するのに適していることがある、よく知られたコンピューティング・システム、環境、または構成、あるいはその組合せの例は、パーソナル・コンピュータ・システム、サーバ・コンピュータ・システム、シン・クライアント、シック・クライアント、ハンドヘルドまたはラップトップ・デバイス、マルチプロセッサ・システム、マイクロプロセッサ・ベースのシステム、セット・トップ・ボックス、プログラム可能家庭用電化製品、ネットワーク PC、ミニコンピュータ・システム、メインフレーム・コンピュータ・システム、および、上記のシステムまたはデバイスのいずれかを含む分散型クラウド・コンピューティング環境、等を含むがこれらに限定されない。

10

#### 【0101】

コンピュータ・システム / サーバ 212 は、コンピュータ・システムによって実行されるプログラム・モジュールなどのコンピュータ・システム実行可能命令の一般的な背景で説明することができる。一般に、プログラム・モジュールは、特定のタスクを実施するか、特定の抽象データ型を実装するルーチン、プログラム、オブジェクト、コンポーネント、ロジック、データ構造などを含むことができる。コンピュータ・システム / サーバ 212 は、通信ネットワークを通じてリンクされたりリモート処理デバイスによってタスクが実施される分散型クラウド・コンピューティング環境において実践することができる。分散型クラウド・コンピューティング環境では、プログラム・モジュールは、メモリ・ストレージ・デバイスを含むローカルとリモート両方のコンピュータ・システム・ストレージ媒体に置くことができる。

20

#### 【0102】

図 5 に示したように、データ処理システム 210 におけるコンピュータ・システム / サーバ 212 を、汎用コンピューティング・デバイスの形で示す。コンピュータ・システム / サーバ 212 の構成要素は、1 つまたは複数のプロセッサまたは処理ユニット 216、システム・メモリ 228、および、システム・メモリ 228 を含む様々なシステム構成要素をプロセッサ 216 に連結するバス 218 を含むことができるがこれらに限定されない。

30

#### 【0103】

バス 218 は、メモリ・バスまたはメモリ・コントローラ、周辺機器バス、アクセラレイティッド・グラフィックス・ポート、および、様々なバス・アーキテクチャのいずれかを使用するプロセッサまたはローカル・バスを含むいくつかのタイプのバス構造のいずれかの 1 つまたは複数を表す。限定ではなく例として、このようなアーキテクチャは、インダストリ・スタンダード・アーキテクチャ (ISA) バス、マイクロ・チャンネル・アーキテクチャ (MCA) バス、エンハンスド ISA (EISA) バス、ビデオ・エレクトロニクス・スタンダーズ・アソシエーション (VESA) ローカル・バス、およびペリフェラル・コンポーネント・インターコネクト (PCI) バスを含む。

40

#### 【0104】

コンピュータ・システム / サーバ 212 は、典型的には、様々なコンピュータ・システム可読媒体を含む。このような媒体は、コンピュータ・システム / サーバ 212 によってアクセス可能な任意の利用可能な媒体であってもよく、揮発性媒体および不揮発性媒体の両方、取外し可能媒体および取外し不能媒体を含む。

#### 【0105】

システム・メモリ 228 は、ランダム・アクセス・メモリ (RAM) 230、またはキャッシュ・メモリ 232、あるいはその両方など、揮発性メモリの形のコンピュータ・シ

50

システム可読媒体を含むことができる。コンピュータ・システム/サーバ212は、他の取外し可能/取外し不能な、揮発性/不揮発性のコンピュータ・システム・ストレージ媒体をさらに含むことができる。ほんの一例として、(図示していないが、典型的には「ハード・ドライブ」と呼ばれる)取外し不能な不揮発性磁気媒体を読み書きするために、ストレージ・システム234を設置することができる。図示していないが、取外し可能な不揮発性磁気ディスク(例えば「フロッピー(R)・ディスク」)を読み書きするための磁気ディスク・ドライブ、および、CD-ROM、DVD-ROM、または他の光媒体などの取外し可能な不揮発性光ディスクを読み書きするための光ディスク・ドライブを設置することができる。このような事例では、それぞれを、1つまたは複数のデータ媒体インターフェースでバス218に接続することができる。下記でさらに描写し説明するように、メモリ228は、本発明の実施形態の機能を実行するように構成されたプログラム・モジュールのセット(例えば、少なくとも1つ)を有する少なくとも1つのプログラム製品を含むことができる。

10

#### 【0106】

プログラム/ユーティリティ240は、プログラム・モジュール242のセット(少なくとも1つ)を有し、オペレーティング・システム、1つまたは複数のアプリケーション・プログラム、他のプログラム・モジュール、およびプログラム・データと同様に、限定ではなく例として、メモリ228に格納することができる。オペレーティング・システム、1つもしくは複数のアプリケーション・プログラム、他のプログラム・モジュール、およびプログラム・データのそれぞれ、またはそのいくつかの組合せは、ネットワーキング環境の実装形態を含むことができる。プログラム・モジュール242は、一般に、本明細書で説明されるような本発明の実施形態の機能または方法あるいはその両方を実行する。

20

#### 【0107】

コンピュータ・システム/サーバ212は、キーボード、ポインティング・デバイス、ディスプレイ224等などの1つもしくは複数の外部デバイス214、コンピュータ・システム/サーバ212とユーザが対話することを可能にする1つもしくは複数のデバイス、または、1つもしくは複数の他のコンピューティング・デバイスとコンピュータ・システム/サーバ212が通信することを可能にする任意のデバイス(例えば、ネットワーク・カード、モデム、等)、あるいはその組合せとも通信することができる。このような通信は、入出力(I/O)インターフェース222を介して発生することができる。さらに、コンピュータ・システム/サーバ212は、ローカル・エリア・ネットワーク(LAN)、一般的な広域ネットワーク(WAN)、またはパブリック・ネットワーク(例えばインターネット)、あるいはその組合せなどの1つまたは複数のネットワークとネットワーク・アダプタ220を介して通信することができる。描写したように、ネットワーク・アダプタ220は、コンピュータ・システム/サーバ212の他の構成要素とバス218を介して通信する。図示していないが、他のハードウェアまたはソフトウェア構成要素あるいはその両方を、コンピュータ・システム/サーバ212とともに使用できることを理解されたい。例は、マイクロコード、デバイス・ドライバ、冗長処理ユニット、外部ディスク・ドライブ・アレイ、RAIDシステム、テープ・ドライブ、およびデータ・アーカイブ・ストレージ・システム等を含むがこれらに限定されない。

30

40

#### 【0108】

本発明は、システム、方法、またはコンピュータ・プログラム製品、あるいはその組合せであってもよい。コンピュータ・プログラム製品は、本発明の態様をプロセッサに実行させるためのコンピュータ可読プログラム命令を有するコンピュータ可読ストレージ媒体(または複数の媒体)を含むことができる。

#### 【0109】

コンピュータ可読ストレージ媒体は、命令実行デバイスによる使用のための命令を保持し格納することができる有形デバイスであってもよい。コンピュータ可読ストレージ媒体は、例えば、電子ストレージ・デバイス、磁気ストレージ・デバイス、光ストレージ・デバイス、電磁気ストレージ・デバイス、半導体ストレージ・デバイス、または前述の任意

50

の適切な組合せであってもよいがこれらに限定されない。コンピュータ可読ストレージ媒体のより具体的な例の完全に網羅されていないリストは、ポータブル・コンピュータ・ディスク、ハードディスク、ランダム・アクセス・メモリ（RAM）、リード・オンリ・メモリ（ROM）、消去可能プログラマブル・リード・オンリ・メモリ（EPROMまたはフラッシュ・メモリ）、スタティック・ランダム・アクセス・メモリ（SRAM）、ポータブル・コンパクト・ディスク・リード・オンリ・メモリ（CD-ROM）、デジタル・バーサタイル・ディスク（DVD）、メモリ・スティック、フロッピー（R）・ディスク、命令が記録されたパンチ・カードまたは溝内隆起構造などの機械的にエンコードされたデバイス、および前述の任意の適切な組合せを含む。コンピュータ可読ストレージ媒体は、本明細書で使用されるように、電波もしくは他の自由に伝搬する電磁波、導波路もしくは他の伝送媒体を通じて伝搬する電磁波（例えば、光ファイバ・ケーブルを通過する光パルス）、またはワイヤを通じて伝送される電気信号などの一過性の信号であると本質的に解釈されるべきではない。

10

#### 【0110】

本明細書で説明されるコンピュータ可読プログラム命令は、コンピュータ可読ストレージ媒体からそれぞれの計算／処理デバイスに、あるいは、例えば、インターネット、ローカル・エリア・ネットワーク、広域ネットワーク、もしくはワイヤレス・ネットワーク、またはその組合せといったネットワークを介して外部コンピュータまたは外部ストレージ・デバイスに、ダウンロードすることができる。ネットワークは、銅伝送ケーブル、光伝送ファイバ、ワイヤレス伝送、ルータ、ファイアウォール、スイッチ、ゲートウェイ・コンピュータ、またはエッジ・サーバ、あるいはその組合せを備えることができる。各計算／処理デバイス内のネットワーク・アダプタ・カードまたはネットワーク・インターフェースは、コンピュータ可読プログラム命令をネットワークから受信し、それぞれの計算／処理デバイス内のコンピュータ可読ストレージ媒体に格納するためにコンピュータ可読プログラム命令を転送する。

20

#### 【0111】

本発明の動作を実行するためのコンピュータ可読プログラム命令は、アセンブラ命令、インストラクション・セット・アーキテクチャ（ISA）命令、機械語命令、機械依存命令、マイクロコード、ファームウェア命令、状態設定データ、または、Smalltalk（R）、C++、もしくは同様のものなどのオブジェクト指向プログラミング言語、および「C」プログラミング言語もしくは類似のプログラミング言語などの従来の手続き型プログラミング言語を含む1つもしくは複数のプログラミング言語の任意の組合せで書かれたソース・コードもしくはオブジェクト・コードであってもよい。コンピュータ可読プログラム命令は、スタンド・アロンのソフトウェア・パッケージとして、全面的にユーザのコンピュータ上か、部分的にユーザのコンピュータ上で実行することができ、または、部分的にユーザのコンピュータ上かつ部分的にリモート・コンピュータ上か、全面的にリモート・コンピュータもしくはサーバ上で実行することができる。後者のシナリオでは、リモート・コンピュータは、ローカル・エリア・ネットワーク（LAN）もしくは広域ネットワーク（WAN）を含む任意のタイプのネットワークを通じてユーザのコンピュータに接続することができ、または、接続は、（例えば、インターネット・サービス・プロバイダを使用してインターネットを通じて）外部コンピュータに対して行われてもよい。いくつかの実施形態では、例えば、プログラム可能ロジック回路機器、フィールド・プログラマブル・ゲート・アレイ（FPGA）、またはプログラマブル・ロジック・アレイ（PLA）を含む電子回路機器は、本発明の態様を実施するために、コンピュータ可読プログラム命令の状態情報を利用して電子回路機器を個別化することによって、コンピュータ可読プログラム命令を実行することができる。

30

40

#### 【0112】

本発明の態様は、本発明の実施形態による方法、装置（システム）、およびコンピュータ・プログラム製品の流れ図またはブロック図あるいはその両方を参照しながら本明細書で説明される。流れ図またはブロック図あるいはその両方の各ブロック、および流れ図ま

50

たはブロック図あるいはその両方におけるブロックの組合せは、コンピュータ可読プログラム命令によって実装できることが理解されよう。

【0113】

これらのコンピュータ可読プログラム命令は、コンピュータまたは他のプログラム可能データ処理装置のプロセッサによって実行する命令が、流れ図またはブロック図あるいはその両方の1つまたは複数のブロックで指定された機能／行為を実装するための手段を作り出すべく、機械を生み出すために、汎用コンピュータ、専用コンピュータ、または他のプログラム可能データ処理装置のプロセッサに提供することができる。これらのコンピュータ可読プログラム命令も、コンピュータ可読ストレージ媒体に格納することができ、流れ図またはブロック図あるいはその両方の1つまたは複数のブロックで指定された機能／行為の態様を実装する命令を含む製品を、命令を格納したコンピュータ可読ストレージ媒体が含むべく、特定の手法で機能するように、コンピュータ、プログラム可能データ処理装置、または他のデバイス、あるいはその組合せに指図することができる。

10

【0114】

コンピュータ可読プログラム命令も、コンピュータ、他のプログラム可能装置、または他のデバイス上で実行する命令が、流れ図またはブロック図あるいはその両方の1つまたは複数のブロックで指定された機能／行為を実装するべく、コンピュータ、他のプログラム可能装置、または他のデバイス上で一連の動作ステップを実施してコンピュータ実行処理を生み出すために、コンピュータ、他のプログラム可能データ処理装置、または他のデバイスにロードすることができる。

20

【0115】

図中の流れ図およびブロック図は、本発明の様々な実施形態によるシステム、方法、およびコンピュータ・プログラム製品の可能な実装形態のアーキテクチャ、機能、および動作を示す。この点に関して、流れ図またはブロック図の中の各ブロックは、指定のロジック機能を実現するための1つまたは複数の実行可能命令を含む命令のモジュール、セグメント、または一部を表すことができる。いくつかの代替実装形態では、ブロックに記された機能は、図に記された順序とは異なる順序で発生してもよい。例えば、連続して示された2つのブロックは、実際には、実質的に同時に実行されてもよく、または、ブロックは、時には、含まれる機能に応じて逆の順序で実行されてもよい。ブロック図または流れ図あるいはその両方の各ブロック、および、ブロック図または流れ図あるいはその両方におけるブロックの組合せは、指定の機能または行為を実施するか、専用ハードウェアとコンピュータ命令との組合せを実行する、専用ハードウェア・ベースのシステムによって実装できることに留意されたい。

30

【0116】

本発明の様々な実施形態の説明を例証のために提示してきたが、網羅的であること、または、開示した実施形態に限定することを意図するものではない。説明した実施形態の範囲および思想から逸脱することなく、多くの変更形態および変形形態が当業者には明らかであろう。本明細書で使用した専門用語は、実施形態の原理、実用的用途、もしくは市場で見つかる技術に対する技術的改善を最も良く説明するように、または、本明細書で開示した実施形態を当業者が理解できるように選ばれた。

40

【符号の説明】

【0117】

- 10 システム・ファームウェア (FW)
- 12 コア
- 14 非同期コア・ネスト・インターフェース (IF)
- 16 集約バッファ
- 18 システム・ネスト
- 20 I/Oバス・コントローラ
- 22 I/Oバス
- 24 I/Oステータス・バッファ

50

2 6	早期完了ロジック	
2 8	バッファ I / O バス・コントローラ・インターフェース	
3 0	I / O ストア命令	
3 2	非同期 I O ドライバ・コード	
3 4	I / O セットアップ・コード	
3 6	非同期転送	
3 8	非同期バス	
4 0	ユーザ・インターフェース	
4 2	アレイ管理ロジック	
4 4	I / O ステータス・アレイ	10
4 6	アレイ管理 & アクセス・ロジック	
4 8	アレイ・エントリ識別を伴うメッセージ	
5 0	システム・ハードウェア ( H W ) / ファームウェア ( F W )	
2 1 0	データ処理システム	
2 1 2	コンピュータ・システム / サーバ	
2 1 4	外部デバイス	
2 1 6	C P U / データ処理ユニット	
2 1 8	I / O バス	
2 2 0	ネットワーク・アダプタ	
2 2 2	I / O インターフェース	20
2 2 4	ディスプレイ	
2 2 8	メモリ	
2 3 0	R A M	
2 3 2	キャッシュ	
2 3 4	ストレージ・システム	
2 4 0	プログラム / ユーティリティ	
2 4 2	プログラム・モジュール	

30

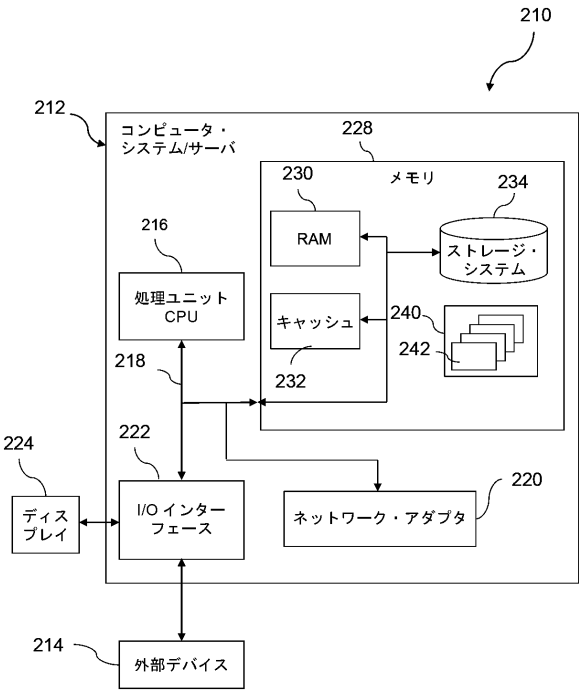
40

50





【図 5】



10

20

30

40

50

## フロントページの続き

- (72)発明者    レイッシュ、クリストフ  
              ドイツ 7 1 0 3 2   ベーブリンゲン   シェーナハイチャー・シュトラーセ 2 2 0
- (72)発明者    クレマー、マルコ  
              ドイツ 7 1 0 3 2   ベーブリンゲン   シェーナハイチャー・シュトラーセ 2 2 0
- (72)発明者    レーナルト、フランク  
              ドイツ 7 1 0 3 2   ベーブリンゲン   シェーナハイチャー・シュトラーセ 2 2 0
- (72)発明者    クライン、マティアス  
              アメリカ合衆国 1 2 6 0 1   ニューヨーク州ボキプシー   サウス・ロード 2 4 5 5
- (72)発明者    ブラッドベリー、ジョナサン  
              アメリカ合衆国 1 2 6 0 1   ニューヨーク州ボキプシー   サウス・ロード 2 4 5 5
- (72)発明者    ジャコビ、クリスチャン  
              アメリカ合衆国 1 2 6 0 1   ニューヨーク州ボキプシー   サウス・ロード 2 4 5 5
- (72)発明者    ドリーバー、ピーター  
              アメリカ合衆国 1 2 6 0 1   ニューヨーク州ボキプシー   サウス・ロード 2 4 5 5
- (72)発明者    ベルマー、ブレントン  
              アメリカ合衆国 1 2 6 0 1   ニューヨーク州ボキプシー   サウス・ロード 2 4 5 5
- 審査官    田中 啓介
- (56)参考文献    特表 2 0 1 3 - 5 3 6 4 8 7 ( J P , A )  
                 特開平 0 7 - 1 0 5 1 2 6 ( J P , A )
- (58)調査した分野 (Int.Cl., D B 名)  
                 G 0 6 F 9 / 3 0 - 9 / 3 5 5  
                 G 0 6 F 1 3 / 1 0 - 1 3 / 1 4  
                 G 0 6 F 1 3 / 3 8 - 1 3 / 4 2