

(19) 日本国特許庁 (JP)

(12) 特 許 公 報 (B2)

(11) 特許番号

特許第6225154号
(P6225154)

(45) 発行日 平成29年11月1日 (2017. 11. 1)

(24) 登録日 平成29年10月13日 (2017. 10. 13)

(51) Int. Cl.	F I
H04L 12/28 (2006.01)	H04L 12/28 200Z
G06F 15/173 (2006.01)	G06F 15/173 665Z
G06F 15/167 (2006.01)	G06F 15/167 610Z

請求項の数 16 外国語出願 (全 45 頁)

(21) 出願番号	特願2015-218678 (P2015-218678)	(73) 特許権者	591003943
(22) 出願日	平成27年11月6日 (2015. 11. 6)		インテル・コーポレーション
(65) 公開番号	特開2016-123076 (P2016-123076A)		アメリカ合衆国 95054 カリフォル
(43) 公開日	平成28年7月7日 (2016. 7. 7)		ニア州・サンタクララ・ミッション カレ
審査請求日	平成27年11月6日 (2015. 11. 6)		ッジ ブレーバード・2200
(31) 優先権主張番号	14/576, 125	(74) 代理人	110000877
(32) 優先日	平成26年12月18日 (2014. 12. 18)		龍華国際特許業務法人
(33) 優先権主張国	米国 (US)	(72) 発明者	ジェン、マイケル シー、
			アメリカ合衆国 95054 カリフォル
			ニア州・サンタクララ・ミッション カレ
			ッジ ブレーバード・2200 インテル
			・コーポレーション内

最終頁に続く

(54) 【発明の名称】 共有メモリリンクの低電力エントリ

(57) 【特許請求の範囲】

【請求項 1】

共有メモリに関連付けられたロード／格納タイプ動作に対応するデータを送信する送信機と、

低電力状態に入るためのリクエストを送信する電力管理ロジックと、を備え、

前記データはメモリアクセスリンクプロトコルに従うリンク経由で送信され、前記メモリアクセスリンクプロトコルは異なる別のリンクプロトコル上にオーバーレイされ、

前記別のリンクプロトコルは、汎用入出力 (I/O) 相互接続プロトコルを含み、

前記リクエストは、前記汎用 I/O 相互接続プロトコルに従い定義されるトークンのフィールド内でエンコードされたデータ値を含み、前記トークンはパケットデータの開始を示し、および前記トークンが前記別のリンクプロトコルおよび前記メモリアクセスリンクプロトコルのうちの 1 つに従うデータを含む後、後続のデータが送信されるかどうかを更に示し、

前記トークンの第 1 のフィールドが、前記後続のデータがメモリアクセスリンクプロトコルデータを含むかまたは汎用 I/O 相互接続プロトコルデータを含むかを示し、

前記第 1 のフィールドが、前記後続のデータがメモリアクセスリンクプロトコルデータを含むことを示す場合、前記データ値は前記トークンの第 2 のフィールド内でエンコードされ、

前記データ値は前記メモリアクセスリンクプロトコル下で定義されたエンコーディングを含み、前記第 2 のフィールドは前記汎用 I/O 相互接続プロトコルの物理層経由で前記

10

20

エンコーディングをトンネリングすべく使用される、装置。

【請求項 2】

前記リンク上で送信されるデータが、前記トークンの複数のインスタンスに基づいて、メモリアクセスリンクプロトコルデータと、汎用 I / O 相互接続プロトコルデータとの間で切り替わる、請求項 1 に記載の装置。

【請求項 3】

前記電力管理ロジックは、前記低電力状態に入るための前記リクエストを送信する前に、前記汎用 I / O 相互接続プロトコルのデータが送信されることを待機しているかどうかを確認する、請求項 2 に記載の装置。

【請求項 4】

前記電力管理ロジックは、前記汎用 I / O 相互接続プロトコルデータが送信されるまで、前記低電力状態に入るための前記リクエストを送信することを待機する、請求項 3 に記載の装置。

【請求項 5】

前記メモリアクセスリンクプロトコルまたは汎用 I / O 相互接続プロトコルのいずれかのリンク層が、物理層の低電力状態へのエントリをトリガできる、請求項 2 に記載の装置。

【請求項 6】

前記汎用 I / O 相互接続プロトコル内で低電力状態に入るために使用される信号が、前記メモリアクセスリンクプロトコル内で低電力状態に入るために使用される信号とは異なる、請求項 5 に記載の装置。

【請求項 7】

前記メモリアクセスリンクプロトコルは、前記メモリアクセスリンクプロトコル内の前記低電力状態に入るための信号をその中で送信する複数の周期的制御ウィンドウを定義する、請求項 6 に記載の装置。

【請求項 8】

リンク経由でデータストリームを受信する受信機と、
リクエストに基づいて低電力リンク状態へのエントリを開始するプロトコルロジックと、
を備え、

前記データストリームはトークンを含み、前記トークンは前記トークンの後続のデータが少なくとも 2 つの代替的なプロトコルのうちの 1 つに従うデータを含むことを示す第 1 のフィールドを含み、前記少なくとも 2 つの代替的なプロトコルはメモリアクセスプロトコルおよび汎用 I / O プロトコルを含み、前記リンク経由で送信されるすべてのデータは前記汎用 I / O プロトコルの物理層経由で送信され、前記トークンの第 2 のフィールドが前記メモリアクセスプロトコルのリンク層による前記低電力リンク状態に入るための前記リクエストを示すべくエンコードされる、装置。

【請求項 9】

前記リクエストに対する応答を送信する送信機を更に備え、
前記応答は受信確認および否定的受信確認のうちの 1 つを含み、前記応答は前記トークンの別のインスタンスで送信される、請求項 8 に記載の装置。

【請求項 10】

前記応答は前記トークンの前記別のインスタンス内で前記第 2 のフィールドのインスタンスが送信される、請求項 9 に記載の装置。

【請求項 11】

リンク経由でデータストリームを受信する段階と、
リクエストに対する応答を送信する段階と、
前記リクエストに基づいて低電力リンク状態へのエントリを開始する段階と、を備え、
前記データストリームはトークンの第 1 のインスタンスを含み、前記トークンの複数のインスタンスは前記トークンの後続のデータが少なくとも 2 つの代替的なプロトコルのうちの 1 つに従うデータを含むことを示す第 1 のフィールドを含み、前記少なくとも 2 つの

10

20

30

40

50

代替的なプロトコルはメモリアクセスプロトコルおよび汎用 I / O プロトコルを含み、前記リンク経由で送信されるすべてのデータは前記汎用 I / O プロトコルの物理層経由で送信され、前記トークンの前記第 1 のインスタンスの第 2 のフィールドが前記メモリアクセスプロトコルのリンク層による前記低電力リンク状態に入るための前記リクエストを示すべくエンコードされ、

前記応答は受信確認および否定的受信確認のうちの 1 つを含み、前記応答は前記トークンの別のインスタンスで送信される、方法。

【請求項 1 2】

第 1 のプロトコルのリンク層と、
異なる第 2 のプロトコルのリンク層と、
前記第 1 のプロトコルの物理層と、

10

データストリームを受信し、前記物理層を低電力リンク状態に入らせるデータを前記物理層に送信するシムと、を備え、

前記第 2 のプロトコルのデータが前記第 1 のプロトコルの前記物理層経由で送信され、

前記データストリームはトークンを含み、前記トークンは前記トークンの後続のデータが第 1 のプロトコルデータまたは第 2 のプロトコルデータのうちの 1 つを含むことを示す第 1 のフィールドを含み、前記トークンの第 2 のフィールドが前記第 2 のプロトコルの前記リンク層による前記低電力リンク状態に入るためのリクエストを示すべくエンコードされる、装置。

【請求項 1 3】

20

前記第 1 のプロトコルは汎用 I / O 相互接続プロトコルを含み、前記第 2 のプロトコルはメモリアクセスプロトコルを含む、請求項 1 2 に記載の装置。

【請求項 1 4】

前記第 1 のプロトコルの 1 または複数の層および前記第 2 のプロトコルの 1 または複数の層を更に備える、請求項 1 2 または 1 3 に記載の装置。

【請求項 1 5】

前記第 1 のプロトコルの前記リンク層は無効にされる、請求項 1 2 から 1 4 のいずれか一項に記載の装置。

【請求項 1 6】

リンク経由でデータストリームを受信する手段と、

30

リクエストに対する応答を送信する手段と、

前記リクエストに基づいて低電力リンク状態へのエントリを開始する手段と、を備え、

前記データストリームはトークンの第 1 のインスタンスを含み、前記トークンの複数のインスタンスは前記トークンの後続のデータが少なくとも 2 つの代替的なプロトコルのうちの 1 つに従うデータを含むことを示す第 1 のフィールドを含み、前記少なくとも 2 つの代替的なプロトコルはメモリアクセスプロトコルおよび汎用 I / O プロトコルを含み、前記リンク経由で送信されるすべてのデータは前記汎用 I / O プロトコルの物理層経由で送信され、前記トークンの前記第 1 のインスタンスの第 2 のフィールドが前記メモリアクセスプロトコルのリンク層による前記低電力リンク状態に入るための前記リクエストを示すべくエンコードされ、

40

前記応答は受信確認および否定的受信確認のうちの 1 つを含み、前記応答は前記トークンの別のインスタンスで送信される、システム。

【発明の詳細な説明】

【技術分野】

【0001】

本開示は、コンピューティングシステムに関し、具体的には（限定はされないが）、コンピューティングシステム内のコンポーネント間のメモリアクセスに関する。

【背景技術】

【0002】

半導体プロセッシングおよびロジック設計における進歩は、集積回路デバイスに存在し

50

得るロジック量の増加を可能にするに至った。当然の結果として、コンピュータシステム構成は、システム内の単一または複数の集積回路から、個々の集積回路上にある複数のコア、複数のハードウェアスレッド、および複数の論理プロセッサ、並びにそのようなプロセッサと統合された他のインタフェースへと進化した。プロセッサまたは集積回路は通常、単一の物理プロセッサダイを備え、当該プロセッサダイは、任意の数のコア、ハードウェアスレッド、論理プロセッサ、インタフェース、メモリ、コントローラハブ等を含み得る。

【0003】

より小さなパッケージで、より大きな処理能力に適合できる、より高い能力の結果として、より小さなコンピューティングデバイスが人気を高めてきた。スマートフォン、タブレット、極薄ノートブック、および他のユーザ機器が、飛躍的に成長してきた。しかしながら、これらのより小さなデバイスは、データストレージおよびフォームファクタを超過する複雑な処理の両方について、サーバに依存する。結果として、高性能コンピューティング市場（すなわち、サーバ空間）におけるニーズも増大した。例えば、現在のサーバにおいて、コンピューティング能力を増大させるには通常、複数のコアを持つシングルプロセッサだけでなく、複数の物理プロセッサ（複数のソケットとも称される）が存在する。しかしながら、コンピューティングシステム内のデバイス数と共に処理能力が増大するにつれ、ソケットと他のデバイス間の通信がより重要なものになっている。

【0004】

実際、相互接続は、電気通信を主に処理してきた従来型のマルチドロップバスから、高速通信を容易にする本格的な相互接続アーキテクチャへと成長した。残念ながら、はるかに使用効率の高い将来のプロセッサへのニーズがあり、対応する要求は、既存の相互接続アーキテクチャの機能に対しある。様々な図面中の同様の参照符号および記号は、同様の要素を示す。

【図面の簡単な説明】

【0005】

【図1】相互接続アーキテクチャを含む、コンピューティングシステムの一実施形態を示す。

【図2】層状スタックを含む、相互接続アーキテクチャの一実施形態を示す。

【図3】相互接続アーキテクチャ内で生成または受信されるリクエストまたはパケットの一実施形態を示す。

【図4】相互接続アーキテクチャのための送信機と受信機のペアに係る一実施形態を示す。

【図5】相互接続の層状プロトコルスタックの例を示す。

【図6】例示ノードの一実施形態に係る簡易ブロックダイアグラムを示す。

【図7】複数のノードを含む、例示システムの一実施形態の簡易ブロックダイアグラムを示す。

【図8】シムロジックを含む、ノードの例示ロジックを示す。

【図9】例示共有メモリリンクに従って送信されるデータ表現である。

【図10】データストリーム内に埋め込まれた例示制御ウィンドウの表現を示す。

【図11A】複数の埋め込み制御ウィンドウを利用する、複数の例示ハンドシェイクのフローダイアグラムを示す。

【図11B】複数の埋め込み制御ウィンドウを利用する、複数の例示ハンドシェイクのフローダイアグラムを示す。

【図12】別の例示ハンドシェイクの表現である。

【図13】データフレーミングトークンの例示的開始の表現である。

【図14】低電力リンク状態に入るための複数の例示的な技術を示すフローチャートである。

【図15】マルチコアプロセッサを含む、コンピューティングシステムのための一実施形態のブロック図を示す。

10

20

30

40

50

【図16】マルチコアプロセッサを含むコンピューティングシステムのための別の実施形態のブロック図を示す。

【発明を実施するための形態】

【0006】

以下の説明には、本発明の深い理解を与えるべく、多数の具体的な詳細が記載されている。例えば、複数の特定のタイプのプロセッサおよびシステム構成、特定のハードウェア構造、特定の設計上およびマイクロ設計上の細部、特定のレジスタ構成、特定の命令タイプ、特定のシステムコンポーネント、特定の寸法／高さ、特定のプロセッサパイプライン段階、および動作等の複数の例である。しかしながら、当業者であれば、本発明を実行するにあたりこれらの具体的な複数の詳細は採用する必要がないことは明らかであろう。他の複数の例において、複数の特定および代替的なプロセッサアーキテクチャ、記載された複数のアルゴリズム用の複数の特定のロジック回路／コード、特定のファームウェアコード、特定の相互接続動作、複数の特定のロジック構成、複数の特定の製造技術および材料、複数の特定のコンパイラ実装、コード内の複数のアルゴリズムについての特定の表現、特定のパワーダウンおよびゲーティング技術／ロジック並びにコンピュータシステムの他の特定の動作の詳細のような、複数の周知コンポーネントまたは方法は、本発明を不必要に不明瞭にするのを回避すべく詳細には記載されていない。

【0007】

以下の複数の実施形態は、複数のコンピューティングプラットフォームまたはマイクロプロセッサといった複数の特定の集積回路内のエネルギー管理およびエネルギー効率に関し記載されている場合があるが、複数の他の実施形態が、集積回路およびロジックデバイスの複数の他のタイプに適用可能である。本明細書に記載の複数の実施形態に係る同様の複数の技術および教示が、より良好なエネルギー効率およびエネルギー管理からまた利益を享受し得る回路または半導体デバイスの他の複数のタイプに適用されてよい。例えば、開示された複数の実施形態は、デスクトップコンピュータシステムまたはウルトラブック（登録商標）に限定はされない。また、例えば複数のハンドヘルドデバイス、タブレット、他の薄型ノートブック、システムオンチップ（SOC）デバイス、および埋め込みアプリケーションのような他の複数のデバイスにおいて使用されることもある。ハンドヘルドデバイスのいくつかの例は、セルラ電話、インターネットプロトコルデバイス、デジタルカメラ、パーソナルデジタルアシスタント（PDA）、およびハンドヘルドPCを含む。複数の埋め込みアプリケーションは通常、マイクロコントローラ、デジタル信号プロセッサ（DSP）、システムオンチップ、ネットワークコンピュータ（NetPC）、セットトップボックス、ネットワークハブ、広域ネットワーク（WAN）スイッチ、または以下に教示される複数の機能および動作を実行可能な任意の他のシステムを含む。さらに、本明細書に記載の複数の装置、方法、およびシステムは、複数の物理的コンピューティングデバイスに限定はされず、エネルギー管理および効率のための複数のソフトウェア最適化にも関連してよい。以下の説明で容易に明らかになるように、本明細書に記載の複数の方法、装置、およびシステムに係る複数の実施形態（ハードウェア、ファームウェア、ソフトウェア、またはそれらの組み合わせに関するかどうかに関わらず）は、性能の考慮事項と調和された「グリーンテクノロジー」の将来にとって不可欠である。

【0008】

コンピューティングシステムが進歩するにしたがい、それらのコンポーネントもより複雑化している。結果として、最適なコンポーネント動作のための帯域幅要件が満たされることを保証すべく、コンポーネント間の連結および通信を行うための相互接続アーキテクチャも複雑性が増している。更に、異なる複数の市場セグメントは市場ニーズに適合すべく、複数の相互接続アーキテクチャの異なる態様を要求する。例えば、複数のサーバがより高性能を要求する一方で、モバイルエコシステムは場合によっては、省電力化のために全体的な性能を犠牲にしてしまう可能性がある。しかし、最大限の省電力化で可能な限り高い性能を提供することが、大半のファブリックの唯一の目的である。以下に説明される多数の相互接続は、本明細書に記載される本発明の複数の態様から潜在的に利益を享受す

るであろう。

【0009】

相互接続ファブリックアーキテクチャの1つは、ペリフェラルコンポーネントインターコネクト (PCI) エクスプレス (PCIe) アーキテクチャを含む。PCIeの主目的は、複数の異なるベンダの複数のコンポーネントおよびデバイスをクライアント (デスクトップとモバイル)、サーバ (標準およびエンタープライズ)、および埋め込み通信デバイス等、複数の市場セグメントにわたるオープンアーキテクチャで相互運用できるようにすることである。PCI Expressは、様々な将来の複数のコンピューティングおよび通信プラットフォームのために定義された高性能で、汎用性のあるI/O相互接続である。その利用モデル、ロード/格納アーキテクチャ、ソフトウェアインタフェースのよう10
ないいくつかのPCI属性が、その複数の改訂版を通して維持されてきたが、これに対し、以前の複数のパラレルバス実装は、高度に拡張可能な完全シリアルインタフェースによって置き換えられてきた。PCI Expressのより最近のバージョンでは、性能と複数の特徴に係る複数の新レベルを供給すべく、ポイントツーポイント相互接続、スイッチベースの技術、およびパケット化されたプロトコルにおける複数の利点を活用している。電力管理、サービスの品質 (QoS)、ホットプラグ/ホットスワップサポート、データ整合性、およびエラー処理は、PCI Expressによってサポートされる高度な複数の特徴のうちいくつかである。

【0010】

図1を参照すると、コンポーネントのセットを相互接続するポイントツーポイントリンクから構成されるファブリックの一実施形態が示されている。システム100は、プロセッサ105およびコントローラハブ115に連結されたシステムメモリ110を含む。プロセッサ105は、マイクロプロセッサ、ホストプロセッサ、埋め込みプロセッサ、コプロセッサ、または他のプロセッサのような任意の処理要素を含む。プロセッサ105はフ10
ロントサイドバス (FSB) 106を介してコントローラハブ115に連結される。一実施形態において、FSB106は後述のようなシリアルポイントツーポイント相互接続である。別の実施形態において、リンク106は、異なる相互接続規格に準拠するシリアル差動相互接続アーキテクチャを含む。

【0011】

システムメモリ110は、ランダムアクセスメモリ (RAM)、不揮発性 (NV) メモリ、またはシステム100における複数のデバイスによってアクセス可能な他のメモリのような、任意のメモリデバイスを含む。システムメモリ110は、メモリインタフェース116を介してコントローラハブ115に連結される。メモリインタフェースの複数の例にはダブルデータレート (DDR) メモリインタフェース、デュアルチャネルDDRメモ10
リインタフェース、および動的RAM (DRAM) メモリインタフェースが含まれる。

【0012】

一実施形態において、コントローラハブ115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またはPCI-E) 相互接続階層内のルートハブ、ルートコンプレックス、またはルートコントローラである。コントローラハブ115の複数の例には、チップセット、メモリコントローラハブ (MCH)、ノースブリッジ、相互接続10
コントローラハブ (ICH)、サウスブリッジ、およびルートコントローラ/ハブが含まれる。チップセットという用語はしばしば、2つの物理的に別個のコントローラハブ、すなわち相互接続コントローラハブ (ICH) に連結されたメモリコントローラハブ (MCH) を指す。現行システムはしばしばプロセッサ105に統合されたMCHを含む一方、コントローラ115は後述と同様の態様で複数のI/Oデバイスと通信することに留意されたい。いくつかの実施形態において、ピアツーピアルーティングが、ルートコンプレックス115を介して任意的にサポートされる。

【0013】

ここでコントローラハブ115は、シリアルリンク119を介してスイッチ/ブリッジ120に連結される。入出力モジュール117および121は、複数のインタフェース/10
50

ポート 1 1 7 および 1 2 1 と呼ばれることもあり、層状プロトコルスタックを包含／実装して、コントローラハブ 1 1 5 とスイッチ 1 2 0 との間の通信を提供する。一実施形態において、複数のデバイスがスイッチ 1 2 0 に連結可能である。

【0014】

スイッチ／ブリッジ 1 2 0 は、複数のパケット／メッセージを上流デバイス 1 2 5、すなわちルートコンプレックスに向かう 1 階層上から、下流コントローラハブ 1 1 5、すなわちルートコントローラから離れて 1 階層下へ、プロセッサ 1 0 5 またはシステムメモリ 1 1 0 からデバイス 1 2 5 へとルーティングする。一実施形態においてスイッチ 1 2 0 は、複数の仮想 P C I 対 P C I ブリッジデバイスの論理的アセンブリと呼ばれる。デバイス 1 2 5 は、任意の内部若しくは外部デバイス、または I / O デバイス、ネットワークインタフェースコントローラ (N I C)、アドインカード、オーディオプロセッサ、ネットワークプロセッサ、ハードドライブ、ストレージデバイス、C D / D V D R O M、モニタ、プリンタ、マウス、キーボード、ルータ、ポータブルストレージデバイス、ファイヤワイヤデバイス、ユニバーサルシリアルバス (U S B) デバイス、スキャナ、および他の複数の入出力デバイスのような電子システムに連結されるコンポーネントを含む。P C I e 用語ではしばしば、そのようなデバイスをエンドポイントと呼ぶ。具体的に示されていないが、デバイス 1 2 5 は、レガシーまたは他のバージョンの複数の P C I デバイスをサポートする、P C I e 対 P C I / P C I - X ブリッジを含んでよい。P C I e における複数のエンドポイントデバイスはしばしば、レガシー、P C I e、またはルートコンプレックス統合エンドポイントとして分類される。

【0015】

グラフィックアクセラレータ 1 3 0 も、シリアルリンク 1 3 2 を介してコントローラハブ 1 1 5 に連結される。一実施形態において、グラフィックアクセラレータ 1 3 0 は、I C H に連結された M C H に連結される。次にスイッチ 1 2 0、およびそれに従って I / O デバイス 1 2 5 が I C H に連結される。I / O モジュール 1 3 1 および 1 1 8 はまた、グラフィックアクセラレータ 1 3 0 およびコントローラハブ 1 1 5 間で通信する層状プロトコルスタックを実装する。上記の M C H の記載と同様に、グラフィックスコントローラまたはグラフィックアクセラレータ 1 3 0 自体がプロセッサ 1 0 5 に統合されてよい。

【0016】

図 2 を見ると、層状プロトコルスタックの一実施形態が示されている。層状プロトコルスタック 2 0 0 は、Q u i c k P a t h インターコネクト (Q P I) スタック、P C i e スタック、次世代高性能コンピューティングインターコネクトスタック、または他の層状スタックなどの任意の形態の層状通信スタックを含む。図 1 ~ 4 に関する直下の記載は P C I e スタックに関するものであるが、同一概念が複数の他の相互接続スタックに適用されてよい。一実施形態において、プロトコルスタック 2 0 0 は、トランザクション層 2 0 5 と、リンク層 2 1 0 と、物理層 2 2 0 とを含む P C I e プロトコルスタックである。図 1 中のインタフェース 1 1 7、1 1 8、1 2 1、1 2 2、1 2 6 および 1 3 1 のようなインタフェースが、通信プロトコルスタック 2 0 0 として表されてよい。通信プロトコルスタックとしての表現は、プロトコルスタックを実装／包含するモジュールまたはインタフェースとも称され得る。

【0017】

P C I E x p r e s s は、パケットを使用してコンポーネント間で情報を通信する。複数のパケットはトランザクション層 2 0 5 およびデータリンク層 2 1 0 で形成され、送信コンポーネントから受信コンポーネントへと情報を搬送する。複数の送信されるパケットが複数の他の層を流れる際に、それらの層でパケットを処理するために必要な追加の情報によってパケットが拡張される。受信側において、リバース処理が発生し、複数のパケットは、それらの物理層 2 2 0 表現からデータリンク層 2 1 0 表現へ、最終的に (複数のトランザクション層パケットのため) 受信デバイスのトランザクション層 2 0 5 によって処理され得る形態へと変換される。

【0018】

〔トランザクション層〕

一実施形態において、トランザクション層 205 は、デバイスの処理コアと、データリンク層 210 および物理層 220 のような相互接続アーキテクチャとの間のインタフェースを提供する。この点において、トランザクション層 205 の主な役割は、複数のパケット（すなわち、複数のトランザクション層パケットまたは TLP）のアセンブリおよびディスアセンブリである。トランザクション層 205 は通常、複数の TLP のためのクレジットベースのフロー制御を管理する。PCIe は、分割トランザクション、すなわちリクエストと時間によって分けられた応答を持つ複数のトランザクションを実装し、ターゲットデバイスが当該応答のためのデータを収集する間、リンクに他のトラフィックを搬送させることを可能にする。

10

【0019】

また、PCIe はクレジットベースのフロー制御を利用する。このスキームにおいて、デバイスはトランザクション層 205 内の複数の受信バッファの各々のために、クレジットの初期量を通知する。図 1 のコントローラハブ 115 のようなリンクの反対側における外部デバイスは、各 TLP によって消費されるクレジット数をカウントする。トランザクションがクレジット限界を超えなければ、当該トランザクションは送信されてよい。応答を受信する際、クレジット量が回復される。クレジットスキームの利点は、クレジット限界が発生しなければ、クレジット返却の待機時間が性能に影響しないことである。

【0020】

一実施形態において、4つのトランザクションアドレス空間は、構成アドレス空間、メモリアドレス空間、入出力アドレス空間、およびメッセージアドレス空間を含む。複数のメモリ空間トランザクションは、メモリマップ位置へ／から、データを転送する読み取りリクエストおよび書き込みリクエストのうち 1 または複数を含む。一実施形態において、複数のメモリ空間トランザクションは、2つの異なるアドレスフォーマットを使用可能である。例えば、32ビットアドレスなどの短いアドレスフォーマット、または 64ビットアドレスなどの長いアドレスフォーマットなどである。複数の構成空間トランザクションを使用して、PCIe デバイスの構成空間にアクセスする。構成空間に対する複数のトランザクションは、読み取りリクエストおよび書き込みリクエストを含む。複数のメッセージ空間トランザクション（または単に複数のメッセージ）は、複数の PCIe エージェント間の帯域内通信をサポートするために定義される。

20

30

【0021】

従って、一実施形態において、トランザクション層 205 はパケットヘッダ／ペイロード 206 を組み立てる。現行の複数のパケットヘッダ／ペイロードのためのフォーマットは、PCIe 仕様ウェブサイトの PCIe 仕様に見出され得る。

【0022】

図 3 を素早く参照するに、PCIe トランザクション記述子の一実施形態が示される。一実施形態において、トランザクション記述子 300 は、トランザクション情報を搬送するためのメカニズムである。この点において、トランザクション記述子 300 はシステム内の複数のトランザクションの識別をサポートする。他の複数の潜在的な使用は、既定のトランザクション順序の変更およびトランザクションと複数のチャネルとの関連付けの追跡を含む。

40

【0023】

トランザクション記述子 300 は、グローバル識別子フィールド 302、属性フィールド 304 およびチャネル識別子フィールド 306 を含む。図示される例では、グローバル識別子フィールド 302 は、ローカルトランザクション識別子フィールド 308 およびソース識別子フィールド 310 を含むよう示される。一実施形態において、グローバルトランザクション識別子 302 は、すべての未処理のリクエストに対し一意である。

【0024】

一実装に従うと、ローカルトランザクション識別子フィールド 308 は要求エージェントによって生成されるフィールドであり、このフィールドは、その要求エージェントに対

50

する完了を必要とするすべての未処理のリクエストに対して一意である。更に、この例において、ソース識別子 310 は P C I e 階層内のリクエストエージェントを一意に識別する。従って、ローカルトランザクション識別子 308 フィールドはソース I D 310 とともに、階層ドメイン内のトランザクションのグローバル識別を提供する。

【0025】

属性フィールド 304 は、トランザクションの複数の特性および関係を指定する。この点において、属性フィールド 304 は、複数のトランザクションの既定処理の変更を許容する追加の情報を提供するために潜在的に使用される。一実施形態において、属性フィールド 304 は、優先度フィールド 312、予約フィールド 314、順序フィールド 316、およびスヌープ無しフィールド 318 を含む。ここで、優先度サブフィールド 312 はトランザクションに優先度を割り当てるイニシエータによって変更されてよい。予約属性フィールド 314 は、将来のため、またはベンダ定義の利用のために、予約されたままとされる。優先度またはセキュリティ属性を使用する、複数の可能な利用モデルは、予約属性フィールドを使用して実装されてよい。

10

【0026】

この例において、順序属性フィールド 316 が使用され、複数の既定の順序ルールを変更し得る順序タイプを伝達する任意情報を供給する。一例示的実装によると、順序属性「0」は、複数の既定の順序ルールが適用されることを示し、順序属性「1」は緩和された順序を示し、複数の書き込みが同一方向への複数の書き込みを渡すことができ、複数の読み取り完了は、同一方向に複数の書き込みを渡すことができる。スヌープ属性フィールド 318 は、複数のトランザクションがスヌープされるかどうかを決定すべく利用される。示される通り、チャンネル I D フィールド 306 は、トランザクションが関連付けられるチャンネルを識別する。

20

【0027】

[リンク層]

リンク層 210 は、データリンク層 210 ととも称され、トランザクション層 205 と物理層 220 との間の中間段階として動作する。一実施形態において、データリンク層 210 の役割は、2つのコンポーネントリンク間で複数のトランザクション層パケット (T L P) を交換するための信頼性のあるメカニズムを提供することである。データリンク層 210 の一方側は、トランザクション層 205 によって組み立てられた複数の T L P を受け入れ、パケットシーケンス識別子 211、すなわち識別番号またはパケット番号を適用し、エラー検出コード、すなわち C R C 212 を計算および適用し、物理デバイスから外部デバイスにわたる送信のため、変更された複数の T L P を物理層 220 に送信する。

30

【0028】

[物理層]

一実施形態において、物理層 220 は、パケットを外部デバイスへ物理的に送信すべく、論理サブブロック 221 および電気サブブロック 222 を含む。ここで論理サブブロック 221 は、物理層 220 の複数の「デジタル」機能の役割を有する。この点において、論理サブブロックは、物理サブブロック 222 による送信のための発信情報を準備するための送信セクションと、受信情報をリンク層 210 に渡す前に受信情報を識別して準備する受信セクションとを含む。

40

【0029】

物理ブロック 222 は、送信機および受信機を含む。送信機には、論理サブブロック 221 によって複数のシンボルが供給され、送信機はそれらをシリアルライズし、外部デバイスへと送信する。受信機は外部デバイスからのシリアルライズされた複数のシンボルが供給され、受信された複数の信号をビットストリームに変換する。ビットストリームはデシリアルライズされ、論理サブブロック 221 に供給される。一実施形態において、8 b / 10 b 送信コードが採用され、ここでは 10 ビットの複数のシンボルが送受信される。ここでは、複数のフレーム 223 でパケットをフレーム化すべく、特別なシンボルが使用される。また、一例において、受信機は着信シリアルストリームから復元されたシンボルクロ

50

クも提供する。

【0030】

上述の通り、トランザクション層205、リンク層210、および物理層220はPC I eプロトコルスタックの具体的な実施形態に関し記載されているが、層状プロトコルスタックはこれに限定されない。実際、任意の層状プロトコルが包含／実装されてよい。一例として、層状プロトコルとして表されるポート／インタフェースは、(1)複数のパケットを組み立てる第1の層、すなわちトランザクション層と、複数のパケットを順序付ける第2の層、すなわちリンク層、および複数のパケットを送信する第3の層、すなわち物理層を含む。具体例として、共通標準インタフェース(C S I)層状プロトコルが利用される。

10

【0031】

次に図4を参照するに、PC I eシリアルポイントツーポイントファブリックの一実施形態が示される。PC I eシリアルポイントツーポイントリンクの一実施形態が示されているが、シリアルポイントツーポイントリンクはシリアルデータを送信するための任意の送信パスを含み、これに限定されない。示される実施形態において、基本のPC I eリンクは2つの低電圧の差分的に駆動される複数の信号ペアを含む。すなわち、送信ペア406／411および受信ペア412／407である。従って、デバイス405は、データをデバイス410に送信するための送信ロジック406と、デバイス410からデータを受信するための受信ロジック407とを含む。換言すると、2つの送信パス、すなわちパス416および417と、2つの受信パス、すなわちパス418および419とがPC I eリンクに含まれる。

20

【0032】

送信パスは、送信ライン、銅線、光回線、無線通信チャネル、赤外線通信リンク、他の通信パスのような、データを送信するための任意のパスを指す。例えば、デバイス405およびデバイス410など、2つのデバイス間の接続はリンク415などのリンクと呼ばれる。リンクは1レーンをサポートしてよく、各レーンは、差分信号ペアのセット(送信のための1ペア、受信のための1ペア)を表す。帯域幅をスケールリングするために、リンクはx Nで示される複数のレーンを集約してよく、ここでNは任意のサポートされるリンク幅、例えば、1、2、4、8、12、16、32、64、またはそれ以上などである。

【0033】

差分ペアとは、複数の差分信号を送信するためのライン416および417などの2つの送信パスを指す。一例として、ライン416が低電圧レベルから高電圧レベルへと切り替わる、すなわち、立ち上りエッジである場合、ライン417は、高論理レベルから低論理レベル、すなわち立ち下りエッジへと、駆動する。複数の差分信号は、複数のより良好な電気的特性、例えば、より良好な信号整合性、すなわち相互連結、電圧オーバシュート／アンダシュート、リングング等を潜在的に示す。これにより、より良好なタイミングウィンドウを可能にし、それにより、より高速な送信周波数を可能にする。

30

【0034】

PC I eを含む、既存の相互接続および通信アーキテクチャに係る複数の物理層は、システム内の共有メモリおよび複数のI/Oサービスを提供すべく活用され得る。従来、キャッシュ可能メモリは、従来のロード／格納(L D / S T)メモリセマンティクスを使用し、複数の独立したシステム間で共有できない。独立したシステム、または「ノード」は、それが単一の論理エンティティとして機能し、単一のオペレーティングシステム(および／または単一のB I O Sまたは仮想マシンモニタ(V M M))によって制御され、および／または独立したフォールトドメインを有するという意味において、独立したものであり得る。単一のノードは、1または複数のプロセッサデバイスを含み、単一のボードまたは複数のボード上で実装され、同一ノード上の複数のデバイスによってL D / S Tセマンティクスを使用してアクセス可能なキャッシュ可能メモリを含む、ローカルメモリを含み得る。ノード内で、共有メモリは、ノード内のいくつかの異なるプロセッサ(例えば、複数の中央処理装置(C P U))によってアクセスされ得るランダムアクセスメモリ(R A

40

50

M)等のメモリの1または複数のブロックを含むことができる。共有メモリはまた、複数のプロセッサのローカルメモリまたはノード内の他の複数のデバイスを含み得る。共有メモリを有するノード内の複数のデバイスは、共有メモリ内のデータの単一ビューを共有できる。共有メモリを含むI/O通信は、非常に短い待機時間であり得、複数のプロセッサによるメモリへの高速アクセスを許容する。

【0035】

従来、複数の異なるノード間のメモリ共有はロード／格納パラダイムによるメモリ共有を許容していなかった。例えば、いくつかのシステムにおいて、異なる複数のノード間のメモリ共有は、分散メモリアーキテクチャを介して容易にされる。従来のソリューションにおいて、計算タスクはローカルデータ上で動作し、別のノードのデータが所望される場合、計算タスク（例えば、別のCPUノードによって実行される）は、当該別のノードと例えばイーサネット（登録商標）、インフィニバンド、または別の層状プロトコルなどの通信プロトコルスタックを利用する通信チャネルを介して通信する。従来の複数のマルチノードシステムにおいて、異なる複数のノードに係る複数のプロセッサは、データがどこに存在するかを認識する必要がない。プロトコルスタック経由等、複数の従来のアプローチを使用してデータを共有することは、ロード／格納パラダイムを使用するノード内のメモリ共有よりも、顕著に長い待機時間を有し得る。共有メモリ内のデータに直接アドレス指定および動作するよりむしろ、あるノードはイーサネット（登録商標）（またはインフィニバンド）のような既存のプロトコルハンドシェイクを使用して別のノードからデータを要求でき、ソースノードは当該データを提供でき、その結果、他の複数の例の中でも特に、当該データは要求ノードによって格納および動作され得る。

【0036】

いくつかの実装例において、メモリが複数の独立したノード間で排他的アクセスまたは共有アクセスのために、ロード／格納（LD／ST）メモリセマンティクスを使用して共有されることを許容する共有メモリアーキテクチャが、提供され得る。一例において、I/Oセマンティクス（PCIeのような複数のプロトコルのための）と共にメモリセマンティクス（および該当する場合、ディレクトリ情報）がピンの共通セットまたはピンの別個のセットのいずれかにエクスポートされ得る。そのようなシステムにおいて、改善された共有メモリアーキテクチャは、システム内の複数のノードの各々がその独自の独立したフォールトドメイン（およびローカルメモリ）を維持することを許容する一方で、複数の当該ノードによるアクセスのための共有メモリプール、およびLD／STセマンティクスによるメモリを使用する複数のノード間で低待機時間のメッセージを渡すことを可能にする。いくつかの実装例において、そのような共有メモリプールは、動的（または静的）に異なる複数のノード間で割り当てられ得る。従って、システムの様々なノードを複数のノードからなる動的に変化する複数のグループに構成でき、様々なタスクに対し、連携して柔軟に動作させて、例えば、ニーズが生じれば共有メモリインフラストラクチャを使用することができる。

【0037】

一実施形態において、相互接続は複数のワークステーションまたはサーバのような複数の高性能コンピューティングプラットフォームにおいて提供され得、複数のプロセッサ、アクセラレータ、コンピューティングブロック、I/Oデバイス等を接続する。相互接続アーキテクチャは、複数のプロトコル層（コヒーレント、非コヒーレント、および任意で他の複数のメモリベースのプロトコル）、ルーティング層、リンク層、および物理層のような複数の定義された層を含むべく、層状にされ得る。図5は、相互接続の例示的層状プロトコルスタックの一実施形態を示す。いくつかの実装例において、図5に示される複数の層のうち少なくともいくつかは、任意であってよい。各層は、情報の粒度またはクォンタムに係るその独自のレベルを対処する（複数のパケット530を持つプロトコル層505a、b、複数のフリット535を持つリンク層510a、b、および複数のフィット540を持つ物理層505a、b）。いくつかの実施形態において、パケットは実装に基づいて、複数の部分的なフリット、単一のフリット、または複数のフリットを含んでよいこ

とに留意されたい。

【0038】

第1の例として、フィット540の幅は、リンク幅対ビット数の1対1のマッピング（例えば、20ビットリンク幅は、20ビットのフィットを含む等）を含む。複数のフリットは、184、192または200ビットのような、より大きなサイズを有してよい。フィット540が20ビット幅で、フリット535のサイズが184ビットの場合、1つのフリット535を送信するには小数値（fractional number）のフィット540が必要となる（例えば、他の複数の例の中でも特に、184ビットのフリット535の送信には20ビットで9.2フィット、192ビットのフリットの送信には20ビットで9.6となる）ことに留意されたい。物理層における基本的なリンクの複数の幅は、変化してよいこ

10

【0039】

一実施形態において、物理層505a、bが、物理媒体（電気または光等）上での情報の高速転送を担うことができる。物理リンクは、層505aおよび505bのような2つのリンク層エンティティ間のポイントツーポイントであり得る。リンク層510a、bは、複数の上方層から物理層505a、bを抽出でき、信頼性をもってデータ（複数のリクエストも）を転送し、2つの直接接続されたエンティティ間のフロー制御を管理する能力を提供する。リンク層はまた、物理チャネルを複数の仮想チャネルおよびメッセージクラスに仮想化する役割を有し得る。プロトコル層520a、bは、複数の物理リンクにわたる転送のために、複数のプロトコルメッセージを物理層505a、bに対し渡す前に、複数のプロトコルメッセージを複数の適切なメッセージクラスおよび仮想チャネルにマッピングすべく、リンク層510a、bに依存する。リンク層510a、bは、他の複数の例の中でも特に、リクエスト、スヌープ、応答、ライトバック、非コヒーレントデータのような複数のメッセージをサポートしてよい。

20

【0040】

物理層505a、b（またはPHY）は、電気層（すなわち、2つのコンポーネントを接続する複数の導体）の上方および図5に示されるようにリンク層510a、bの下方に実装されるよう定義され得る。物理層および対応するロジックは各エージェントに存在でき、（例えば、あるリンクのいずれかの側における複数のデバイスにおいて）互いに分離されている、2つのエージェント（AおよびB）上の複数のリンク層を接続する。ローカルおよび遠隔の複数の電気層は、物理媒体に（例えば、複数のワイヤ、導体、光等）よって接続される。一実施形態において、物理層505a、bは、2つの主要なフェーズである、初期化および動作のフェーズを有する。初期化中、接続はリンク層に対して不透明であり、シグナリングは複数の時限状態とハンドシェイクイベントとの組み合わせを含んでよい。動作中、接続はリンク層に対して透明であり、シグナリングは、すべてのレーンが単一のリンクとして共同して動作する際、特定の速度である。動作フェーズ中、物理層はエージェントAからエージェントBへ、およびエージェントBからエージェントAへ、複数のフリットをトランスポートする。接続はリンクとも称され、複数のフリットおよび現在の構成（例えば幅）の制御／状態をリンク層と交換しつつ、媒体、幅および複数のリンク層からの速度を含む、いくつかの物理的な態様を抽出する。初期化フェーズは、複数のマイナーフェーズ、例えば、ポーリングや構成を含む。動作フェーズはまた、複数のマイナーフェーズ（例えば、リンク電力管理状態）を含む。

30

40

【0041】

一実施形態において、リンク層510a、bは、2つのプロトコルまたはルーティングエンティティ間で信頼性のあるデータ転送を提供すべく実装され得る。リンク層は、プロ

50

トコル層 5 2 0 a、b から物理層 5 0 5 a、b を抽出でき、2 つのプロトコルエージェント (A、B) 間のフロー制御を担当でき、プロトコル層 (複数のメッセージクラス) およびルーティング層 (複数の仮想ネットワーク) に対し、複数の仮想チャネルサービスを提供できる。プロトコル層 5 2 0 a、b とリンク層 5 1 0 a、b との間のインタフェースは通常、パケットレベルであり得る。一実施形態において、リンク層での最小転送単位は、1 9 2 ビットまたは何らかの他の単位である指定されたビット数である、フリットと称される。リンク層 5 1 0 a、b は、物理層 5 0 5 a、b の転送単位 (フィット) を、リンク層 5 1 0 a、b の転送単位 (フリット) へフレーム化すべく、物理層 5 0 5 a、b に依存する。また、リンク層 5 1 0 a、b は、論理的に 2 つの部分、送信機と受信機とに分割されてよい。1 つのエンティティ上の送信機 / 受信機ペアは、別のエンティティ上の送信機 / 受信機ペアと接続されてよい。フロー制御はしばしば、フリットおよびパケットの両方をベースに実行される。エラー検出および訂正はまた、潜在的には、フリットレベルベースで実行される。

10

【0 0 4 2】

一実施形態において、ソースからターゲットへと複数のトランザクションをルーティングすべく、ルーティング層 5 1 5 a、b は柔軟性のある分散された方法を提供できる。複数のトポロジのための複数のルーティングアルゴリズムは、各ルータにおけるプログラマブルルーティングテーブルを介して指定され得るので、このスキームは柔軟性がある (一実施形態においてプログラミングは、ファームウェア、ソフトウェア、またはそれらの組み合わせにより実行される)。ルーティング機能は分散型であってよく、ルーティングは一連のルーティングステップを介してなされてよく、各ルーティングステップは、ソース、中間、またはターゲットのルータのいずれかにおけるテーブルの検索を介して定義される。ソースでの検索は、パケットをファブリックへと挿入するために用いられてよい。中間ルータでの検索は、入力ポートから出力ポートへパケットをルーティングするために使用されてよい。ターゲットポートでの検索は、ターゲットプロトコルエージェントをターゲットにするために使用されてよい。複数のルーティングテーブル、つまり、複数のルーティングアルゴリズムは仕様によって具体的に定義されないので、いくつかの実装例において、ルーティング層を薄くできることに留意されたい。これにより柔軟性、および柔軟なプラットフォームアーキテクチャトポロジを含む様々な利用モデルが、システム実装によって定義されることを可能にする。ルーティング層 5 1 5 a、b は、最大 3 つ (またはそれより多い) の仮想ネットワーク (VN)、一例においては各仮想ネットワーク内で定義された、いくつかのメッセージクラスを持つ VN 0 および VN 1 の 2 つのデッドロックフリーの VN の使用を提供するため、リンク層 5 1 0 a、b に依存する。共有適応仮想ネットワーク (VNA) が、リンク層で定義されてよいが、この適応ネットワークは複数のルーティング概念に直接さらされなくてよい。というのは、各メッセージクラスおよび仮想ネットワークは、他の複数の特徴および例の中でも特に、複数の専用リソースおよび保証された転送進行を有してよい。

20

30

【0 0 4 3】

一実施形態において、相互接続アーキテクチャは、メモリからの複数のデータラインをキャッシュする複数のエージェントをサポートすべく、コヒーレンシプロトコル層 5 2 0 a、b を含むことができる。メモリデータをキャッシュしたいエージェントは、コヒーレンシプロトコルを使用して、そのキャッシュにロードすべく、データラインを読み取ってよい。そのキャッシュ内のデータラインを変更したいエージェントは、コヒーレンシプロトコルを使用して、データを変更する前にラインのオーナーシップを取得してよい。ラインの変更後、エージェントはラインをメモリにライトバックするか、またはラインを外部リクエストへの応答に含むまで、ラインをそのキャッシュに保持する複数のプロトコル要件に従ってよい。最後に、そのキャッシュ内のラインを無効化すべく、エージェントは複数の外部リクエストに応じ得る。プロトコルは、すべてのキャッシュエージェントが従い得る複数のルールを指示することによって、データのコヒーレンシを保証する。また、それはキャッシュを持たない複数のエージェントのために、コヒーレントにメモリデータを読

40

50

み書きする手段を提供する。

【0044】

コヒーレンシプロトコルの例を利用する複数のトランザクションをサポートすべく、2つの条件が課され得る。第1に、プロトコルは一例として、複数のエージェントのキャッシュ内のデータにわたり、およびそれらのデータとメモリ内のデータとの間で、アドレス毎ベースでデータ一貫性を維持できる。非公式に、データ一貫性は、最新のデータ値を表わすエージェントのキャッシュ内の各有効なデータラインを指してよく、コヒーレンシプロトコルパケット内で送信されたデータは、送信された時点での最新のデータ値を表わし得る。複数のキャッシュまたは送信において、データの有効なコピーが存在しない場合、プロトコルは最新のデータ値がメモリ内に存在することを保証してよい。第2に、プロトコルは、複数のリクエストのための明確に定義された複数のコミットメントポイントを提供できる。複数の読み取りのための複数のコミットメントポイントは、データが利用可能になる時を示してよい。また、複数の書き込みのための複数のコミットメントポイントは、書き込まれたデータがグローバルに観測可能となり、複数の後続の読み取りによってロードされる時を示してよい。プロトコルは、コヒーレントメモリ空間内のキャッシュ可能およびキャッシュ不可能（UC）の両方のリクエストのためのこれらのコミットメントポイントをサポートしてよい。

10

【0045】

いくつかの実装例において、相互接続は埋め込みクロックを利用できる。クロック信号は相互接続を使用して送信されたデータ内に埋め込まれ得る。データに埋め込まれたクロック信号によって、別個の専用化された複数のクロックレーンを省略できる。例えば、これにより、デバイスのより多くのピンをデータ転送のため専用化でき、特にピンのための空間が高価であるような複数のシステムにおいて専用化できるので、有用であり得る。

20

【0046】

相互接続のいずれの側においても、2つのエージェント間でリンクを確立することが可能である。データを送信するエージェントは、ローカルエージェントであり得、データを受信するエージェントは遠隔エージェントであり得る。複数のステートマシンが両方のエージェントによって採用されてよく、リンクの様々な態様を管理する。一実施形態において、物理層データパスは、リンク層から電氣的フロントエンドに複数のフリットを送信できる。一実装では、制御パスはステートマシン（リンクトレーニングステートマシンまたはその同等物とも称される）を含む。ステートマシンの複数の動作および複数の状態からの複数の終了は、複数の内部信号、タイマ、外部信号、または他の情報に依存してよい。実際、いくつかの初期化状態のような複数の状態のうちのいくつかは、状態を終了するためのタイムアウト値を提供すべく、複数のタイマを有してよい。いくつかの実施形態において、検出とはレーンの両方の側でイベントを検出することを指すが、必ずしも同時にというわけではないことに留意されたい。

30

【0047】

ステートマシンにおいて定義される複数の状態は、他の複数のカテゴリおよびサブカテゴリの中でも特に、複数のリセット状態、複数の初期化状態、および複数の動作可能状態を含むことができる。一実施形態において、動作と終了は、複数のトレーニングシーケンスの交換に基づくことができる。一実施形態において、リンクステートマシンはローカルなエージェントのクロックドメインで実行され、ある状態から次の状態への移行は、送信機のトレーニングシーケンス境界と一致する。現在の状態を反映すべく、複数の状態レジスタが利用されてよい。複数の例、状態は例えば、次のものを含んでよい。

40

【0048】

送信リンク状態：リンク状態複数のフリットが遠隔エージェントに送信される。状態はブロッキングリンク状態から入ってよく、タイムアウトのようなイベントでブロッキングリンク状態へ戻ってよい。送信リンク状態中（T L S）、送信機は複数のフリットを送信し、受信機は複数のフリットを受信する。また、T L Sから低電力リンク状態へ終了してよい。いくつかの実装例において、T L Sは「L 0」状態として称され得る。

50

【0049】

ブロッキングリンク状態：リンク状態送信機および受信機は、統一された態様で動作している。時限状態であってよく、その間、複数のリンク層フリットが保留されるが、物理層情報は遠隔エージェントへと通信される。低電力リンク状態（または設計に基づき、他のリンク状態）へ終了してよい。一実施形態において、ブロッキングリンク状態（BLS）が周期的に発生してよい。当該期間は、BLS間隔と称され、時限的であってよく、低速度と動作速度との間で異なってよい。長さに係る物理層制御シーケンスが、送信リンク状態または部分幅送信リンク状態中等に送信され得よう、リンク層は周期的に複数のフリットを送信することをブロックされてよいことに留意されたい。いくつかの実装例において、ブロッキングリンク状態（BLS）は、L0制御または「L0c」状態と称され得る。

10

【0050】

部分幅送信リンク状態：省電力リンク状態一実施形態において、非対称部分幅とは、複数の異なる幅を有する2方向リンクの各方向を指し、これらはいくつかの設計でサポートされてよい。速度は部分幅状態では変更できない一方、当該リンクの複数のレーンの幅は変更されてよい。したがって、複数のフリットは潜在的に複数の異なる幅で送信される。部分幅状態は、特定の受信および送信された複数のメッセージに基づく低電力リンク状態、または他の複数のイベントに基づく部分幅送信リンク状態の終了またはリンクブロッキング状態のような他の複数のリンク状態へと終了してよい。一実施形態において、送信機ポートは、より良好な信号整合性（すなわち、ノイズ軽減）を提供すべく、複数のアイドルレーンを交互にオフにしてよい。複数のヌルフリットのような再試行不可能な複数のフリットが、リンク幅が変更されている期間中に利用されてよい。対応する受信機は、1または複数の構造における現在および前の複数のレーンマップを記録するのに加え、これらのヌルフリットをドロップし、複数のアイドルレーンを交互にオフにしてよい。複数の状態および関連付けられた状態レジスタは、変更されないままであってよいことに留意されたい。いくつかの実装例において、部分幅送信リンク状態は、部分的L0、またはL0p状態と称され得る。

20

【0051】

部分幅送信リンク状態の終了：部分幅状態の終了いくつかの実装例において、ブロッキングリンク状態を使用してもよいし、しなくてもよい。一実施形態において、送信機は複数のアイドルレーンをトレーニングおよびデスキューすべく、複数の部分幅終了パターンを送信することによって、終了を開始する。一例において、終了パターンはEIEOSで開始する。EIEOSはレーンがエントリを完全な送信リンク状態に開始する準備ができていることをシグナリングすべく、検出およびデバウンスされ、複数のアイドルレーン上でSDSまたは高速トレーニングシーケンス（FTS）で終了してよい。終了シーケンス（タイムアウト前に完了されないデスキューのような複数の受信機の動作）中の任意の不具合は、リンク層への複数のフリット転送を停止し、リセットをアサートする。リセットは次のブロッキングリンク状態でリンクをリセットすることによって処理される。SDSはまた、複数のレーン上のスクランブラ／デスクランブラを複数の適切な値へ初期化してよい。

30

40

【0052】

低電力リンク状態：より低電力状態一実施形態において、それは部分幅リンク状態より低電力である。というのは、本実施形態においてシグナリングはすべてのレーンおよび両方の方向で停止されるからである。複数の送信機は、低電力リンク状態を要求するためにブロッキングリンク状態を使用してよい。ここで、受信機はリクエストをデコードしてよく、ACKまたはNAKで応答してよく、さもなければ、リセットがトリガされてよい。いくつかの実装例において、低電力リンク状態はL1状態と称され得る。

【0053】

図6を参照するに、複数の独立したノード610a～610nの各々によって、ロード／格納の複数の技術を使用してアクセス可能な共有メモリ605を含む例示システムを示

50

す、簡易ブロックダイアグラム600が示される。例えば、システム上の様々なノード610a~610nに係る複数のロード／格納アクセスリクエストを受け入れ可能な共有メモリコントローラ615が提供され得る。共有メモリ605はシンクロナスダイナミックランダムアクセスメモリ（SDRAM）、複数のデュアルインラインメモリモジュール（DIMM）、および他の不揮発性メモリ（または揮発性メモリ）を利用して実装され得る。

【0054】

各ノードそれ自体で、1または複数のCPUソケットを有してよく、またシステム内の他の複数のノードによるLD／STアクセスから隔絶されたままのローカルメモリを含んでよい。ノードは、他の複数の例の中でも特に、PCIe、QPI、イーサネット（登録商標）を含む1または複数のプロトコルを使用して、システム上の複数の他のデバイス（共有メモリコントローラ615、ネットワークングコントローラ620、他の複数のノード等）と通信できる。いくつかの実装例において、共有メモリリンク（SML）プロトコルが提供され得、それを介して低待機時間LD／STメモリセマンティクスがサポートされ得る。例えば、SMLがシステムの様々なノード610a~610nによる共有メモリ605に係る複数の読み取りおよび書き込みの通信において（共有メモリコントローラ615を介して）、使用され得る。いくつかの実装例において、SMLは図5の例に関連して記載される相互接続アーキテクチャおよびプロトコルの複数の態様を採用できる。

【0055】

一例において、SMLはScalable Memory Interconnect（SMI）第3世代（SMI3）のようなメモリアクセスプロトコルに基づいてよい。複数の他のメモリアクセスプロトコルが代替的に使用されてよく、例えば他の複数の例の中でも特に、Fully Buffered DIMM（FB-DIMM）、DDR Transactional（DDR-T）のような複数のトランザクションメモリアクセスプロトコルが挙げられる。他の複数の例において、SMLは追加の複数のディレクトリ拡張を持つネイティブのPCIeメモリ読み込み／書き込みセマンティクスに基づくことができる。SMLのメモリプロトコルベースの実装は、複数のラインメモリアクセスをキャッシュするよう調整されていることにより、帯域幅効率の点で複数の利点を提供できる。PCIeのような複数の高性能デバイス間通信プロトコルが存在する一方で、そのような複数のプロトコルに係る複数の上方層（例えば、トランザクション層およびリンク層）は、共有メモリ605を必要とする複数のトランザクションを含む複数のLD／STメモリトランザクションにおける使用のために、完全なプロトコルの適用を低下させる待機時間を導入し得る。メモリリンクプロトコルはPCIeのような別のプロトコルスタックの大半を迂回できるので、SMI3のようなメモリリンクプロトコルは、複数のより短い待機時間のアクセスを供給する潜在的な追加的な利点を許容できる。従って、SMLの複数の実装例は、SMI3、またはPCIe上のSMI3のような別のプロトコルに係る論理および物理PHY上で実行されている別のメモリプロトコルを利用できる。

【0056】

記載した通り、いくつかの実装例において、システム内のノード610a~610nに係る複数のロード／格納リクエストを処理するためのロジックを含む共有メモリコントローラ（SMC）615が提供され得る。複数のロード／格納リクエストは、SMLを利用し、およびノード610a~610nをSMC615に接続することによって、複数のリンク経由でSMC615によって受信され得る。いくつかの実装例において、SMC615は、複数の共有メモリリソースのためにノード610a~610nに係る複数のアクセスリクエストをサービスするためのロジックを含む、特定用途向け集積回路（ASIC）のようなデバイスとして実装され得る。他の複数の例において、SMC615（並びに共有メモリ605）は、ノード610a~610nのうち1または複数（あるいはさらにすべて）と別のデバイス、チップ、またはボード上に存在し得る。SMC615は更に、共有メモリ605を含む様々なノードの複数のトランザクションを調整するロジックを含むことができる。さらに、SMCは、共有メモリ605内に含まれた各キャッシュラインの

ような様々なデータリソースへのディレクトリ追跡アクセスを維持できる。例えば、他の複数の潜在的な例の中でも特に、データリソースは、共有アクセス状態（例えば、ノード内のプロセッシングデバイスおよび／またはI/Oデバイスによって同時にアクセス可能（例えば、ロードされまたは読み取られ））、排他的アクセス状態（例えば、一時的でない場合、ノード内の単一のプロセッシングおよび／またはI/Oデバイスによって排他的に予約済み（例えば、格納または書き込み動作のため））、キャッシュされていない状態であり得る。更に、各ノードが共有メモリ605の1または複数の部分へのダイレクトアクセスを有してよい一方で、複数の異なるアドレス指定スキームおよび値が様々なノード（例えば、610a～610n）によって採用されてよい。これにより、第1のアドレス値による第1のノードおよび第2のアドレス値により同一データを参照する第2のノードによって、参照される同一共有メモリデータ（例えば、命令内で）をもたらす。SMC615は、複数の共有メモリリソースへの複数のノードのアドレスをマッピングする複数のデータ構造を含むロジックを含んでよく、SMC615が様々なノードの様々なアクセスリクエストを解釈することを許容する。

【0057】

さらに場合によっては、共有メモリのいくつかの部分（例えば、複数の特定のパーティション、メモリブロック、レコード、ファイル等）は、複数の特定の許可、ルール、割り当ての支配下にあるべく、結果的にノード610a～610nの一部分のみが、対応するデータにアクセスすることを許容される（例えば、SMC615によって）。実際、各共有メモリリソースは、システムのノード610a～610nのそれぞれの（および場合によっては、異なる）サブセットに割り当てられてよい。新規のまたは変更された共有メモリ605の特定部分に適用可能な複数のルール、許可、ノード割り当ておよびオーナーシップを収容すべく、これらの割り当ては動的であり得、SMC615は、そのような複数のルールおよび許可（例えば、オンデマンド、動的等）を変更できる。

【0058】

例示のSMC615は更に、1または複数の共有メモリリソースにアクセスするシステム内の複数のノード（例えば、610a～610n）を含む様々なトランザクションを追跡できる。例えば、SMC615は、他のトランザクション情報の中でも特に、トランザクションに関わるノードの識別、トランザクションの進行（例えば、それが完了されたかどうか）を含む、各共有メモリ605トランザクションのための情報を追跡できる。これは、複数の従来の分散メモリアーキテクチャに係るトランザクション指向の態様のうちいくつかは、本明細書に記載の改善されたマルチノード共有メモリアーキテクチャに適用されることを許可する。さらに、トランザクション追跡（例えば、SMCによる）は、それぞれのノードに係る複数の別個の独立したフォールトドメインを維持または強制することを補助すべく使用され得る。例えば、SMCは、メモリ内を含む、その複数の内部データ構造内の進行中の各トランザクションに対応するノードIDを維持でき、またその情報を使用して、複数のアクセス権を強制し、各ノードの複数の個々のフォールトドメインを維持できる。従って、複数のノードのうちの1つがダウンする場合（例えば、重大なエラー、トリガされた復元シーケンス、または他のフォールト若しくはイベントに起因するもの）、そのノードおよび共有メモリ605に関わるその複数のトランザクションのみが中断され（例えば、SMCによってダンプされる）、すなわち、共有メモリ605を含む複数の残りのノードに係る複数のトランザクションは、他方のノード内のフォールトから独立して継続する。

【0059】

システムは複数のノードを含むことができる。さらに、いくつかの例示システムは複数のSMCを含むことができる。場合によって、ノードは、ノードが直接取り付けられてない遠隔SMCから共有メモリにリモートアクセス可能であってよい（すなわち、ノードのローカルSMCは遠隔SMCに1または複数のSMLリンクホップを介して接続する）。遠隔SMCは、同一ボード内にあるべく、異なるボード内にあることもできる。場合によって、ノードのうちいくつかは、共有メモリ605にアクセスするものであるにも関わ

10

20

30

40

50

らず、オフシステム（例えば、オフボードまたはオフチップ）であってよい。例えば、1または複数のオフシステムノードは複数の他の例の中でも得に、SML準拠のリンクを使用してSMCに直接接続できる。さらに、それら独自のSMCおよび共有メモリを含む他の複数のシステムはまた、メモリ605の共有を、例えば、SMLリンク経由でSMCに接続される他のSMCとのインタフェースを取る別のボード上に含まれる複数のノードまで拡張すべく、SMC610に接続できる。更に、他のオフボードまたはオフチップノードへアクセスを更に拡張すべく、複数のネットワーク接続がトンネリングされ得る。例えば、SMLはイーサネット（登録商標）接続経由でトンネリングでき（例えば、ネットワークコントローラ620を介して提供される）、図6の例示システムを、1または複数の他のノードを含み、および他の複数の例の中でも特に、これらのノードがSMC615へのアクセスおよびそれにより共有メモリ605のアクセスを取得することを可能にする別のシステムに、通信可能に連結する。

10

【0060】

別の例として図7の簡易ブロックダイアグラム700に示されるように、LD/STメモリセマンティクスに従う複数の独立したノードによる共有アクセスを許可する改善された共有メモリアーキテクチャは、様々な異なるマルチノードシステム設計の提供を柔軟に許容できる。例示システムで提供される1または複数の共有メモリブロックの複数の部分を共有すべく、複数のノードの様々な組み合わせが割り当てられ得る。例えば、図7の例に示される別の例示システムは、例えば別の複数のダイ、ボード、チップ等の実装される複数のデバイス705、710、715、720を含むことができ、各デバイスは1または複数の独立したCPUノード（例えば、610a～610h）を含む。各ノードはその独自のローカルメモリを含むことができる。複数のデバイス705、710、715、720のうち1または複数の更に、システムの2または2より多いノード610a～610hによってアクセス可能な共有メモリを含むことができる。

20

【0061】

図7に示されるシステムは、本明細書に図示され記載される改善された共有メモリアーキテクチャを介して実現され得る変動性のうちいくつかを示すべく提供される例である。例えば、デバイスA705およびデバイスC715の各々は、それぞれの共有されたメモリ要素（例えば605a、605b）を含むことができる。従って、いくつかの実装例において、別個のデバイス上の各共有メモリ要素は更に、それぞれの共有メモリコントローラ（SMC）615a、615bを含んでよい。ノード610a～610hの様々な組み合わせは、各SMC（例えば、615a、615b）に通信可能に連結され得、当該複数のノードを対応する共有メモリ（例えば605a、605b）にアクセスすることを許容する。一例として、デバイスA705のSMC615aは、SMLをサポートするダイレクトデータリンクを使用してデバイスA上のノード610a、610bに接続できる。さらに、別のデバイス（例えば、デバイスC715）上の別のノード610cはまた、ノード610c（および／またはそのデバイス715）からSMC615aへ直接ハードワイヤードの接続（SMLをサポートする）によって共有メモリ605aへのアクセスを有してよい。間接ネットワークベース、または他の複数のそのような接続はまた、遠隔またはオフボードデバイス（例えば、デバイスD720）の複数のノード（例えば610f～610h）が、共有メモリ605aへのアクセスも有すべく、SMC615aとのインタフェースを取る従来のプロトコルスタックを利用することを許容すべく使用され得る。例えば、SMLトンネル725はイーサネット（登録商標）、インフィニバンド、またはデバイスAとデバイスDとを連結する他の接続経由で確立され得る。トンネルを確立および維持することが、いくつかの追加のオーバーヘッドおよび待機時間を導入する可能性がある一方、よりソフトウェアによって管理されていない複数の他の物理接続上で実行されるSMLと比較した場合、SMLリンク経由でSMCと通信する任意の他のノードが可能なのと同様、SMLトンネル725が確立された場合は、他の複数のSMLチャネルとして動作でき、ノード610f～610hが、SML経由でSMC615aとインタフェースを取ること、および共有メモリ605aにアクセスすることを許容する。例えば、複数のSM

30

40

50

Lチャネル内の複数のパケットの信頼性および順序付けは、システム内の複数のネットワークコンポーネントによって課され得るか、あるいは複数のSMC間のエンドツーエンドが課され得る。

【0062】

さらなる他の複数の例において、共有メモリ（例えば605a）の特定の部分をホストするデバイスとは異なるデバイス上の複数のノード（例えば615d、615e）は、それ自体が対応するSMC（例えば615a）に（例えば、SMLリンクを使用して）連結される別のSMC（例えば615b）に直接接続することによって、対応するSMC（例えばSMC615a）に間接的に接続できる。2または2より多いSMC（例えば615a、615b）をリンクすることは、システム上のノード610a～610hに利用可能な共有メモリ量を効果的に拡張できる。例えば、いくつかの実装例において、図7の例中のSMC615aと615bとの間のリンクにより、SMC615aを介して共有メモリ605aにアクセス可能な複数のノード（例えば、610a～610c、610f～610h）のうち任意のものはまた、SMC615aとSMC615bとの間の接続により共有可能なメモリ605bに潜在的にアクセスしてよい。同様に、いくつかの実装例において、他の複数の潜在的な例の中でも特に、SMC615bに直接アクセスする複数のノードの各々はまた、SMC615aと615bとの間の接続により、共有可能なメモリ605aにアクセスできる。

【0063】

上述したように、改善された共有メモリアーキテクチャは、SMI3のようなメモリアクセスプロトコルに基づき、また共有メモリを含む複数のロード／格納リクエストを容易にすべく提供された低待機時間リンクプロトコル（すなわち、SML）を含むことができる。従来のSMI3および複数の他のメモリアクセスプロトコルが単一ノード内のメモリ共有での使用のために構成されてよい一方で、SMLは複数のノード間のメモリ共有を許容すべく、メモリアクセスセマンティクスを複数のノードへ拡張できる。更に、SMLは潜在的に任意の物理通信リンク上で利用され得る。SMLは、別個の複数のデバイス（および複数のノード）を相互接続するよう適合された物理層（および対応する物理層論理）にオーバーレイされたLD／STメモリアクセスセマンティクスをサポートするメモリアクセスプロトコルを利用できる。さらに、SMLの物理層論理は他の複数の特徴の中でも特に、パケット低下およびエラー再試行機能を提供しないようにできる。

【0064】

いくつかの実装例において、SMLはPCIePHY上にSMI3をオーバーレイすることによって、実装され得る。SMLリンク層は、フロー制御および他の複数の特徴において先んじており、および複数の従来のCPUメモリアクセスアーキテクチャにおける特性であるような、さらに短い待機時間メモリアクセスを容易にすべく提供され得る（例えば、従来のPCIeリンク層の代わりに）。一例において、SMLリンク層論理は、複数の共有メモリトランザクションと複数の他のトランザクションとの間で多重化できる。例えば、SMLリンク層論理はSMI3およびPCIeトランザクション間で多重化できる。例えば、SMI3（または別のメモリプロトコル）は、当該リンクがSMI3とPCIeトランザクションとの間で動的に切り替え可能なように、PCIe（または別の相互接続プロトコル）の上部でオーバーレイできる。これにより、いくつかの例において、従来のPCIeトラフィックをSMLトラフィックと同じリンク上に効果的に共存することを許容する。

【0065】

図8の簡易ブロックダイアグラム800を参照するに、例示的コンピューティングノード（例えば610）の複数の特徴が示されている。コンピューティングノード610は、SMI3（または別のメモリプロトコル）をPCIe（例えばPCIe3.0）準拠の電氣的物理リンク上方にオーバーレイすることによって実装された例示的SMLを使用して、共有メモリコントローラチップ615と通信できる。ノード610は、メモリアクセスプロトコル（例えば、SMI3）のための列と、例えばPCIeのようなシリアル汎用相互

10

20

30

40

50

接続プロトコル等の別の相互接続プロトコルのための別の列を含む、マルチ列層状スタックを含むことができる。各列は、その独自の複数のリンク層（例えば805、810）並びにトランザクション、ルーティング、プロトコル、および／またはコヒーレンシ層（不図示）を含むそれぞれのリンク層805、810の上部にある他の層を有することができる。この例において、メモリアクセスプロトコルは、192ビットの複数のフリットを利用するリンク層を使用できる。そのような複数のフリットが別のプロトコルのPHY上にわたり送信され得、この場合は、ネイティブの128ビットエンコーディングスキーム（例えば、128b／130b）を利用するPCIe準拠PHYである。従って、ノード610のPCIe PHYロジックは、他の複数の例の中でも特に、PCIe論理PHY815、PCIe veneer 820（例えば、デジタルドメインとアナログドメインとの間に存在する追加の層）、およびPCIeアナログフロントエンド（AFE）825を含むことができる。

10

【0066】

変換ロジック830は、PCIe物理リンク上のキャリッジのために、メモリアクセスプロトコルに従って送信された192ビットのフリットデータを128ビットの複数のPCIeペイロードに変換できる。同様に、128ビットの複数のペイロードは192ビットのフリットデータに変換し直す（例えば、ロジック835を使用して）ことができる。この例において、PCIe列は、複数の256ビットのPCIe3.0ペイロードを128ビットデータに変換する類似の変換ロジック840、845を含むことができる。シム回路850は、複数のリンク層とPCIe物理層との間のインタフェースとして提供され得る。シム850は、どのリンク層データが物理層に送信されるかを制御できる。いくつかの実装例において、複数の列（およびリンク層805、810のうちのたった1つ）のうちの1つのみの使用を許容するよう、シム850が選択的に構成され得る（例えば、ノード610のハードウェア内のヒューズを設定することによって）。例えば、シム850は、当該ノードがネイティブPCIeポート（すなわち、PCIeリンク層805のみを利用する）またはSMLポート（すなわち、PCIe物理層上にオーバーレイされたメモリアクセスプロトコルおよびメモリアクセスプロトコルリンク層810のみを利用する）を実装するよう、設定され得る。他の複数の例において、シム850は、いずれの列からのデータも動的に多重化すべく機能し得、当該データストリームをPCIeデータモードとSMLデータモードとの間で切り替える。

20

30

【0067】

図9は、PCIeのような別の異なるシリアル相互接続プロトコル上でメモリアクセスプロトコルをオーバーレイする例を示す。図7の例において、複数の物理層フレーミングトークンが、メモリアクセスと相互接続プロトコルトラフィックとの間で複数の移行をシグナリングすべく使用され得る。フレーミングトークン（または「トークン」）は、当該トークンに関連付けられたデータストリームに含まれるシンボル数を指定または示唆する物理層データカプセル化であり得る。結果として、フレーミングトークンは、ストリームが開始されていることを識別でき、並びにストリームが終了する位置を示唆でき、したがってフレーミングトークンはまた、次のフレーミングトークンの位置を識別すべく使用され得る。データストリームのフレーミングトークンは、データストリームの第1のデータブロックの第1のレーン（例えば、レーン0）の第1のシンボル（シンボル0）に位置付けられ得る。フレーミングトークンは、トラフィックをトランスポートすべく使用される物理層のネイティブプロトコルで定義される既存のシンボルを使用して実装され得る。例えば、PCIeの例では、TLPトラフィックトークンの開始（STP）、データストリームトークンの終了（EDS）、不良トークンの終了（EDB）、DLLPトークンの開始（SDP）、および論理アイドル（IDL）トークンを含む、5つのフレーミングトークンが定義され得る。

40

【0068】

図9の例において、SMLはSMI3またはPCI上の別のデータアクセスプロトコルをオーバーレイすることによって実装され得、SMI3（TLPトラフィックの代わりに）

50

がリンクの複数のレーン上で開始することを識別する新規 S T P トークンを定義すべく、標準 P C I e S T P トークンが再使用およびエンコードされ得る。ポートが S M L で静的に機能する場合、トラフィックが S M L であることを示すべく、S T P トークンがエンコードされ得る。P C I e および S M L トラフィックがリンク上で多重化され得る場合、S T P トークンは、他の複数の例の中でも特に、P C I e トラフィックの複数のバーストと S M L トラフィックの複数のバーストとの間の複数の移行を示すことができる。

【0069】

一例において、標準 P C I e S T P トークンに係る複数の予約ビットは、S M L トラフィック S T P トークン（例えば、「S M L トークン」）の開始を定義すべく変更され得る。例えば、複数の P C I e ペイロードの複数の許容された長さのような、定義済みセットの許容パケットまたはペイロード長は、ホストプロトコルのために定義されてよい。S T P トークンは、長さフィールドを含んでよく、長さフィールドは P C I e に定義された、許容された長さのセット（または物理層の別のホストプロトコル）の範囲外にある複数のエンコーディングを許容してよい。一実装において、不許容値または許容された複数の長さの範囲外にある別の定義値を持つ長さフィールドをエンコーディングすることは、S T P トークンを S M L トークンとして識別すべく使用され得る。

【0070】

図9の例に戻ると、動的 S M L / P C I e 多重化が示されており、S M L は P C I e P H Y プロトコル経由で S M I 3 を利用している。例えば、従来の P C I e 128 b / 130 b エンコーディングのために指定されたエンコーディングに従うべく、同期ヘッダデータが定義され得る。例えば、915 a ~ 915 c において、複数のデータブロックが来ることを示す値 10 b を持つ複数の同期ヘッダが受信される。P C I e S T P（例えば920）が受信される場合、P C I e T L P ペイロードが预期され、データストリームがそれに応じて処理される。P C I e T L P ペイロードは、P C I e S T P 920 内で識別されたペイロード長と一貫性があり、割り当てられた完全なペイロード長を利用できる。別の S T P トークンは、T L P ペイロードの終了後、データブロック内で基本的にいつでも受信され得る。例えば、925 において、S M I 3 S T P が受信され得、P C I e T L P データから S M I 3 フリットデータへの移行をシグナリングする。次に、例えば、P C I e パケットデータの終了が識別されるとすぐに、S M I 3 S T P が送信され得る。

【0071】

P C I e T L P データと同様、S M I 3 S T P 925 は、後に続く S M I 3 フリットペイロードの長さを定義できる。例えば、S M I 3 データのペイロード長は、送信される S M I 3 フリット数（または対応する D W 数）に対応できる。ペイロード長に対応するウィンドウ（例えば、レーン3のシンボル15で終了する）はそれにより、S M I 3 データのみが送信される当該複数のレーン上で定義され得る。ウィンドウが終了する場合、T L P データまたは順序付けられたセットデータ等の他のデータの送信を再開すべく、別の P C I e S T P のような他のデータが送信され得る。例えば、図9の例に示されるように、S M I 3 S T P トークン925によって定義される S M I 3 データウィンドウの終了後、E D S トークンが送信される。E D S トークンはデータストリームの終了をシグナリングでき、図9の例の場合のように、順序付けられたセットブロックが次に続くことを示唆できる。順序付けられたセットブロックが送信されることを示すべくエンコード01 b された同期ヘッダ940が送信される。この場合、P C I e S K P の順序付けられたセットが送信される。そのような複数の順序付けられたセットは周期的に、または複数の設定された間隔若しくはウィンドウに従って送信され得、その結果、他の複数の例の中でも特に、ビット整列の初期化、シンボル整列の初期化、複数の P H Y パラメータの交換、2つの通信ポートのための複数の異なるビットレートの補償を含む、様々な P H Y レベルタスクおよび調整が実行され得る。場合によっては、定義されたウィンドウまたは対応する S M I 3 S T P トークンによって S M I 3 フリットデータのために指定されたデータブロックに割り込ませるべく、必須の順序付けられたセットが送信され得る。

【0072】

図9の例では明示的に図示されていないが、STPトークンはまた、リンク上のSMI3フリットデータからPCIE TLPデータへと移行すべく使用され得る。例えば、定義されたSMI3ウィンドウの終了後、PCIE STPトークン（例えば、トークン920と同様の）が、次のウィンドウは指定量のPCIE TLPデータの送信のためのものであることを示すべく送信され得る。

【0073】

いくつかの実施形態において、複数のメモリアクセスフリット（例えば、複数のSMI3フリット）はサイズが変わってよく、メモリアクセスペイロードのために、対応するSTPトークン（例えば、SMI3 STPトークン）内にどのくらいのデータが予約されているかを先験的に予測することが困難になる。一例として、図9に示されるように、SMI3 STP925は、244バイトのSMI3データのが、SMI3 STP925後に予測されことを示す長さフィールドを有し得る。しかしながら、この例において、10フリット（例えば、SMI3フリット0～9）のみが、当該ウィンドウ中に送信される準備が整っており、これら10のSMI3フリットのみが244バイトのうち240バイトを利用する。従って、4バイトの空の帯域幅が残され、これらは複数のIDLトークンで埋められる。これは、PCIE TLPデータが待ち行列に入れられ、SMI3ウィンドウが閉じるのを待機する場合、特に次善策となり得る。他の複数の場合において、複数のSMI3フリットの送信のために提供されるウィンドウは、当該レーンに準備されたSMI3データ量を送信するには不十分である可能性がある。リンク上に共存するSMI3とPCIE TLPデータとの間をどのように調整するかを決定すべく、複数の調整技術が採用され得る。更に、いくつかの実装例において、リンクのより効率的な使用を補助すべく、複数のSMI3ウィンドウの長さが動的に変更され得る。例えば、調整または他のロジックは、定義されたウィンドウ長がレーンに対し予期されるSMI3（および競合するPCIE TLPトラフィック）の量に対し、より良好に最適化され得るかどうかを決定すべく、複数の定義されたSMI3ウィンドウがどれだけ利用されているかを監視できる。従って、そのような実装例において、複数のSMI3 STPトークンに係る複数の長さフィールド値は、他の複数の例の中でも特に、SMI3フリットデータが割り当てられるべきリンク帯域幅の量に応じて（例えば、TLP、DLLP、および順序付けられたセットデータを含む他のPCIEデータに関連して）、動的に調整され得る（例えば、異なる複数の値の間で）。

【0074】

複数のシリアル入出力（I/O）動作は、動作可能時に顕著な電力を消費し得る。従って、トラフィックがリンクの2方向で保留中でない場合、リンクを低電力またはアイドル状態（例えば、L1状態）に置くことが望ましい可能性がある。これは、PCIEまたはSMLモード動作のいずれかまたは両方に当てはまる可能性がある。更に、SMLモードで使用されるメモリアクセスプロトコルはエントリを定義し、ホストPHY層プロトコル（例えば、PCIE内）で定義されたものとは異なるように、低電力モードからの終了を定義してよい。更に、メモリアクセスプロトコルによって利用されるステートマシンは、物理層を統制する他のプロトコル内で利用されるものとは異なってよい。

【0075】

メモリアクセスプロトコルの一実装において示すべく、メモリアクセスプロトコルのリンク層は、複数の物理層が複数の制御タスクを実行する複数の機会を提供でき、およびこれらの制御タスクは送信リンク状態（TLS）から低電力またはL1状態への移行を含み得る。例えば、メモリアクセスプロトコルが外部プロトコルのPHYを使用することを必要とするSMLの文脈外で使用される場合、それは制御状態（Loc）またはブロッキングリンク状態、その中で物理層の複数のリクエストおよび応答が送信され得る複数のウィンドウを提供できる。一例として、そのような複数の制御タスクに関連付けられたメッセージングを容易にすべく、制御状態Locが提供され得る。複数の物理層制御メッセージがリンク層を使用して送信される複数のフリットのストリーム間で送信されるのを可能に

すべく、L0c状態がT L S内の周期ウィンドウとして提供され得る。例えば、図10に示される例に表されるように、L0状態が複数のL0c間隔に分割され得る。各L0c間隔は、L0c状態または、その中で複数の物理層制御コードおよび他のデータが送信され得るウィンドウ（例えば、1005）で開始し得る。L0c間隔の残り（例えば、1010）は、複数のフリットの送信専用であり得る。各間隔におけるL0c間隔およびL0c状態の長さはプログラムを使って、他の複数の例の中でも特に、例えば1または複数のデバイスのB I O S若しくは別のソフトウェアベースのコントローラによって定義され得る。L0c状態は、L0c間隔の残りよりも飛躍的に短い。例えば、一例において、L0cは8U Iであり得る一方、他の複数の例の中でも特に、L0c間隔の残りは約4K U Iである。これは、実質的にリンクデータ帯域幅を中断または消耗させることなく、比較的短い、予め定義された複数のメッセージがその中で送信され得る複数のウィンドウを可能にする。

10

【0076】

L0c状態メッセージは、物理層レベルで様々な条件を通信できる。一例において、あるデバイスは、例えば、特定の閾値を超える量の複数のビットエラーまたは他の複数のエラーに基づいて、リンクまたはレーンのリセットを開始できる。そのような複数のエラーはまた、複数のL0cウィンドウ（複数の先行するL0cウィンドウ等）内で通信され得る。L0c状態はまた、他の複数のリンク状態間の複数の移行を補助またはトリガする際使用されるシグナリングのような、他の帯域内シグナリングを実現すべく活用され得る。一例において、リンクをアクティブL0状態からスタンバイ若しくはL1状態のような低電力状態へと移行させるべく、複数のL0cメッセージが利用され得る。図11Aの簡易フロー図1100aに示されるように、L1エントリリクエスト（例えば1110）を通信すべく特定のL0c状態が使用され得る。デバイス（またはデバイス上のエージェント）がリクエスト1110の受信確認を待機する間、更なる複数のフリット（例えば1120、1130）が送信され得る。リンク上の他のデバイスが、当該受信確認（例えば、1140）を送信できる。いくつかの例において、当該受信確認はまたL0cウィンドウ内で送信され得る。いくつかの例において、L1リクエスト1110の受信／送信後、当該受信確認は次のL0cウィンドウで送信され得る。各デバイスにおいて、複数のL0c間隔を同期すべく複数のタイマが採用され得、複数の他の例の中でも特に、要求側デバイスは受信確認1140が次のL0cウィンドウにおいて送信されたという識別に基づいて、受信確認1140をリクエスト1110（例えば、独立したL1エントリリクエスト以外）の受信確認として識別できる。いくつかの例において、受信確認はL1エントリリクエスト1110で使用されるものとは別個のL0Cコードを介して通信され得る。他の複数の例において、受信確認1140は他の複数の例の中でも特に、リクエスト1110内で使用されるL1エントリリクエストコードのエコーを含むことができる。更に、図11Bに示されるような代替的な複数の例において、非受信確認信号またはN A K 1145はL0cウィンドウ内で通信され得、L1エントリリクエスト1110に関わらず、リンクをリンク送信状態に留まらせる。

20

30

【0077】

図10、11Aおよび11Bの複数の例は、S M Lの実装で使用するメモリアクセスまたは他のプロトコルに従う低電力移行ハンドシェイクの一例にすぎない。複数の他の実施形態において、アクティブ状態から低電力またはアイドル状態への移行を実現すべく、複数の他のメカニズムが採用され得る。しかしながら、そのような複数のメカニズムは、メモリアクセスプロトコルが別のプロトコルの物理層でオーバーレイされている場合ではサポートされない可能性がある。例えば、メモリアクセスプロトコル（例えばS M I 3）がP C I eのような別のシリアル相互接続プロトコル上にオーバーレイされているS M Lの複数の実装例においてである。実際、P H Yによって利用されるホストプロトコルは、アクティブ状態から低電力状態への複数の移行を異なって実装してよく、低電力状態への移行を開始するメモリアクセスプロトコルの能力を複雑化させる。これは、S M Lが静的に実装される（すなわち、他のプロトコルの物理層上の唯一のトラフィックである）複数の実

40

50

装例において特に厄介であり得る。

【0078】

そのような複数の潜在的な差異を示すべく、図10、11Aおよび11Bに示される低電力状態へのエントリのためのメカニズムを、PCIEの低電力L1状態へのエントリの表現（図12）と比較されたい。例えば、第1（上流）のコンポーネント1205は、下流コンポーネント1210へ、構成書き込みリクエスト1215を送信できる。下流コンポーネント1210は、構成書き込みリクエスト1215の受信に基づいて、送信リンク状態から低電力状態（L1）への移行処理（1220において）を開始できる。下流コンポーネント1210は、構成書き込みリクエストのための完了1225を送信でき、複数の最小クレジットを累積でき、また複数の新規トランザクション層パケット（TLP）のスケジューリングをブロックする（1230において）。下流コンポーネント1210は、直前のTLPのための受信確認（ACK）を受信するのを待機する（1235において）。下流コンポーネントのTLPのすべてが受信確認されたら、下流コンポーネントは上流コンポーネント1205からの応答を受信するまで、Enter_L1データリンク層の複数のパケット（複数のDLLP）の送信1240（L1状態へのエントリを示すべく）を開始できる。

10

【0079】

複数のEnter_L1 DLLPを受信すると、上流コンポーネント1205は、複数の新規TLPの送信のスケジューリングをブロックできる（1245において）。次に上流コンポーネント1205は、以前に送信した直前のTLPのためのリンク層受信確認を受信するまで、待機できる。下流コンポーネント1210は、送信された複数のEnter_L1 DLLPに対する複数の受信確認のDLLP（例えば、Request_ACK）を待機できる（1250において）。上流コンポーネント1205がその直前のTLPのためのACKを受信する場合（1255において）、上流コンポーネント1205は受信された複数のEnter_L1 DLLPに応答して、複数のRequest_ACK DLLPを繰り返し送信1260できる。下流コンポーネント1210が、その複数の受信レーン（上流コンポーネント1205がL1リクエストに対する移行を受信確認したことをシグナリングする）においてRequest_Ack DLLPをキャプチャすると、下流コンポーネント1210は次に、DLLP送信を無効にし（1265において）、上流方向の物理リンクを電気的アイドル状態にする。更に、上流コンポーネント1205上の複数の受信レーンが電気的アイドル状態に入ると、上流コンポーネント1205は、複数のRequest_Ack DLLPの送信を停止し、DLLP送信を無効にし（1270において）、その複数の送信レーンを電気的アイドルにし、それによりリンクのL1への移行を完了させる。L1はいくつかの例において、リンクのいずれかの側にあるコンポーネントによっても終了され得る。

20

30

【0080】

記載した通り、低電力リンク状態への移行を開始すべく使用されるホストプロトコルまたはSMLプロトコルのいずれかを提供することが望ましい可能性がある。しかしながら、いくつかの実装例において、SMLによって使用されるメモリアクセスプロトコルは、図10、11Aおよび11Bの複数の例に図示および記載されるように、低電力状態から複数のPHY層へのエントリのネゴシエーションを代理できない可能性がある。従って、SMLのいくつかの実装例において、低電力状態に入るべく使用されるメモリアクセスプロトコルの複数のネイティブエンコーディングが、SMLを容易にすべく使用される複数のトークン内にトンネリングされるのを許容するL1エントリスキームが提供され得る。例えば、一例において、SMLによって使用されるメモリアクセスプロトコルは、ブロッキングリンク状態または複数の制御ウィンドウ内で複数のリンク状態移行リクエスト（および複数の受信確認）を送信できる。これらの同一エンコーディングがトークンの予約済みフィールドまたは未使用フィールド内に含まれ得、結果的にそれらはホストプロトコル（例えばPCIE）または層スタックに影響を与えることなく、下流コンポーネントに通信される。一実装において、STPトークン（例えばSMLトークン）のフィールドは

40

50

、低電力状態に入るためのリクエスト（またはACK/NAK）を示す複数のデータ値でエンコードされ得る。

【0081】

図13を参照するに、STPトークン1305の表現が示されている。STPトークン1305は、TLPシーケンス番号フィールド1310、およびSTPの後に続くメモリアクセスプロトコル（例えばSMI3）ペイロードの長さ（フリット数の見地で）を識別する長さフィールド1315を含む、いくつかのフィールドを含むことができる。上述したように、いくつかの実装例において、後続データがホスト相互接続プロトコル（例えばPCIE）に従うか、またはメモリアクセスプロトコル（例えばSMI3）に従うかを示すべく、長さフィールドが使用され得る。例えば、1または複数の標準ペイロード長がTLPデータのために定義され得る。SMI3データはいくつかの実装例において、固定数のフリットを含むよう定義され得、あるいは他の複数の場合において、可変数のフリットを有してよく、その場合SMI3フリット数のための長さフィールドは無視可能なフィールドになる。更に、SMI3 STPのための長さフィールドは、複数の定義されたTLPのペイロード長のうちの1つ以外の長さとして定義され得る。従って、SMI3 STPは、一例としてSTP長さフィールドに存在する非TLP長さの値に基づいて識別され得る。

10

【0082】

図13の例を続けると、STPがSML STPトークンの場合、TLPシーケンス番号フィールド1310は予約（未使用）され得る。TLPシーケンス番号フィールド1310は、他の複数の可能性のあるエンコーディングの中でも、L1への移行のリクエストを示すべく、ACK若しくはNAKまたはL1に入るためのリクエストを示すべく、あるいはSML STPトークンに動作が含まれないことを示すべく、エンコードされ得る。例えば、例示的な実装において、メモリアクセスプロトコルのネイティブバージョン内の複数の制御ウィンドウでの使用のために、値のセットが定義され得、これらの同一値はSML STPトークンの予約済みフィールド内でエンコードされ得る。例えば、メモリアクセスプロトコルとしてSMI3を利用する実装においては次の通りである。

20

【0083】

【表1】

エンコーディング	意味
x333	SMI3 QNOP:動作なし
x999	SMI3 QL1:低電力L1状態に入るリクエスト
X666	SMI3 QL1n:L1に入るリクエストに対する非受信確認(NAK)
その他	予約済み

30

40

【0084】

表1の例で続行するに、SML STPトークン内にエンコーディングされたL1に入るためのリクエストが、エコー、および受信確認すべくエンコードされたSML STPトークン、L1に入るためのリクエスト（例えばQL1）、あるいは代替的には非受信確認メッセージ（例えばQL1n）のいずれかで応答され得るよう、ハンドシェイクは定義され得る。後者の場合、受信確認ハンドシェイクが実現されるまで、処理は周期的に反復され得る。L1に入るためのリクエスト（または応答）または応答を示すべくエンコードされていないSML STPトークンの複数の指定フィールドは、複数の他の例の中でも、代替的なエンコーディングまたは非動作エンコーディングでエンコードされ得る。

【0085】

50

図14は、少なくとも一例によるL1に入るためのハンドシェイク例を示すフローチャートである。この例において、PCIe経由でのメモリアクセスプロトコルの低電力状態エントリリクエストに係るトンネリングを容易にすべく、シム850が使用され得る。例えば、電力管理エージェント(PMA)1401、1402が複数のそれぞれのリンク層を低電力に非同期的にさせ得る(例えば1405、1415において)。次にそれぞれのリンク層(LL)810a、810bは、低電力状態(例えばL1)(1410、1420において)へのエントリを開始すべく、シム(それぞれ805a、805b)と通信し得る。例えば、シム805a、805bはメモリアクセスプロトコルに従ってL1エントリ信号1410、1420を受信できる。シム805a、805bは、EL1cデータがメモリアクセスプロトコルに従う、低電力状態L1に入るためのリクエストであることを識別するためのロジックを含むことができる。シム805bは、PCIeベースSTPトークンのTLPシーケンス番号フィールドのようなPCIeトークンフィールドを、メモリアクセスプロトコルで定義された低電力エントリリクエストエンコーディングでエンコードし得る。例えば、複数のそのようなコードを送信すべく、メモリアクセスプロトコルはブロッキングリンク状態(L0c)を使用でき、PCIeベーストークンフィールド(1425において)を利用してL0c低電力エントリハンドシェイクをトンネリングすべく、シム805a、bが使用され得る。従って、シム805aがエンコードされたトークンを受信する場合、当該シムはL1に入るようリンク層810aにシグナリングできる(1430において)。シム805aはまたL1エントリリクエストの受信確認を送信でき(1425において)、シム805bはACKの受信時、L1に入るようそのリンク層810bにシグナリングできる(1435において)。更に受信確認の受信の際、シム805bは更にPCIe物理層L1エントリハンドシェイクを開始するよう物理層815bにシグナリングできる(1440において)。従って、1または複数の電氣的アイドルの順序付けられたセット(EIEOS)がPHY815bによって送信され得、およびPHY815aによってエコーされ得、PCIe物理層に電氣的アイドルに入らせる1450。同一のことが信号1455、1460を介して、それぞれのシム805a、805bに対し確認され得る。

【0086】

図14の例を続けると、当該リンクのL1へのエントリ後の何らかの場合において、リンク層810b(ソフトウェアによってプロンプトされ)は、送信リンク状態(例えばL0またはL0p)に再度入るべく、低電力L1状態からの終了を開始できる。例えば、L1終了信号1465が送信され、シム805bによってデコードされ得、シム805bはPHY815bにシグナリングでき(1470において)、低電力状態を終了させ、リンクに対する送信リンク状態への構成を開始させる。例えば、リンクをデバウンス、同期化、および構成すべく、複数の順序付けられたセット(OS)およびトレーニングシーケンス(TS)が交換されてよい(1475において)。リンクトレーニング(例えば1475を介して)の完了の際、シム805a、805bは、物理層が送信状態L0であることを識別できる。次にシム805a、805bはリンク層810a、810bに、L0に入ったことを通知できる。シム805a、805bがメモリアクセスプロトコルデータの動的多重化およびPCIeデータをPCIe物理層経由で提供する場合において、L0(またはL1)へのエントリが他の複数の例の中でも特に、シム805a、805bによってメモリアクセスプロトコルおよび複数のPCIeリンク層の両方に通信され得る。

【0087】

汎用I/O(GPIO)プロトコル(例えばPCIe)とGPIOプロトコルの物理層経由のメモリアクセスプロトコル(例えば、SMI3)との間の動的切り替えをサポートする複数の実装例において、シムは両方のプロトコルスタックが、共通の物理層に電氣的アイドルに入るようシグナリングする前に、低電力リンク状態(例えばL1)に入る準備が整っていることを決定してよい。例えば、GPIOプロトコルリンク層(帯域幅のその部分中)またはメモリアクセスプロトコルのいずれかが低電力リンク状態に入るよう独立的にリクエストしてよい(そのそれぞれのプロトコルに従って)。しかしながら、ある列

10

20

30

40

50

が低電力状態に入ることを試行する場合、他の列がアクティブリンク上に送信されることを待機するデータを有してよい。複数の動的シムは、いくつかの実装例において、複数のスタック間の帯域幅要求を調整する能力を有してよい。従って、いくつかの実装例において、シムは2つの異なるプロトコルスタックからのデータを多重化する場合、低電力状態に入るための他の列のリクエストを許可する前に、帯域幅の着信部（例えば、STPトークンの後）内の複数の列のうちの1つによって何もデータが送信されないことを保証すべく、両方の列を監視できる。いくつかの実装例において、シムは、L1へのエントリを要求する列に対し、PHYがL1に入ったという確認を合成してよい一方で、そのようにすることで他の列からの送信がされるのを待機しているデータに影響を与えないであろうとシムが決定するまで、L1への実際のエントリをトリガさせるPHYに対するリクエストを保留してよい。例えば、他の第2の列がL1へのエントリを要求し、リンクが実際にL1であることを確信した後、第1の列は当該物理リンクを利用することを許可されてよい。複数のそのような例において、第1の列がデータ送信を行うとすぐに、シムは当該リンクにアイドルに入らせるよう物理層に対しシグナリングしてよい。他の複数の例において、第2の列のリンク層は当該リンクがアクティブである間、再解除されてよく、既にアクティブなリンク（すなわち、他の列のデータの送信に使用されている）を再アクティブ化することを試行してよい。複数のそのような例において、シムはリンクのアクティブ化を開始すべく、第2の列のリンク層からのデータを受信してよい。複数のそのような例において、シムはリンク層の第2の列に対し、リンクの再解除の確認を合成できる。実際には、リンクが既にトレーニングされ、動作可能であるとき、それが物理層に関連するような動作は何ら発生せず、シムは第2の列のリンク層からの信号に応答して、第2の列に対し当該リンクがアクティブであるという確認信号を単に送信してよく、当該リンクに対し低電力状態を「終了」させる。複数のそのような例において、信号物理リンクの帯域幅が、2つの別個のプロトコルのデータ間で共有され得るのみでなく、複数の異なるプロトコルに係る複数の低電力状態エントリパラダイムもサポートされてよい。

【0088】

上記原理および例の多くが場合によっては、PCIeおよびSMI3のような複数の特定のプロトコルの文脈で記載されていることに留意されたい。しかしながら、これらのプロトコルは複数の例示としてのみ列挙されており、任意のシリアルGPIOプロトコルおよびメモリアクセスプロトコルが、SMLの複数の実装で利用され得ることを理解されたい。実際、本明細書に記載の複数の原理、ソリューション、および特徴が他の複数のプロトコルおよびシステムに対し等しく適用可能である。さらに、複数の他の例の中でも特に、本明細書に記載のリンクおよびその対応するロジックに対する論理的および物理的強化に係る複数の組み合わせを含む、上記複数のソリューションに係る複数の組み合わせが複数のシステム内で適用され得る。

【0089】

上記の複数の装置、方法、およびシステムは、上記の任意の電子デバイスまたはシステムに実装されてよいことに留意されたい。複数の具体例として、以下の図面は、本明細書に記載の本発明を利用するための複数の例示的なシステムを提供する。複数の以下のシステムがより詳細に記載される通り、多数の異なる相互接続が開示され、記載され、上記説明から再検討される。さらに容易に明らかなように、上記の複数の進歩は、それらの相互接続、ファブリック、またアーキテクチャの任意のものに適用され得る。

【0090】

図15を参照するに、マルチコアプロセッサを含むコンピューティングシステムのブロック図の一実施形態が示されている。プロセッサ1500はマイクロプロセッサ、組み込みプロセッサ、デジタル信号プロセッサ(DSP)、ネットワークプロセッサ、ハンドヘルドプロセッサ、アプリケーションプロセッサ、コプロセッサ、システムオンチップ(SOC)、またはコードを実行する他のデバイスのような任意のプロセッサまたは処理デバイスを含む。プロセッサ1500は、一実施形態において、少なくとも2つのコア、すなわちコア1501および1502を含み、これらは複数の非対称コアまたは対称コア(当

該図示の実施形態)を含むことができる。しかしながら、プロセッサ1500は、任意の数の対称又は非対称であり得る処理要素を含んでもよい。

【0091】

一実施形態において、処理要素は、ソフトウェアスレッドをサポートするハードウェアまたはロジックを指す。ハードウェア処理要素の複数の例は、スレッドユニット、スレッドスロット、スレッド、処理ユニット、コンテキスト、コンテキストユニット、論理プロセッサ、ハードウェアスレッド、コア、および／または実行状態またはアーキテクチャ状態のようなプロセッサのための状態を保持可能な任意の他の要素を含む。換言すると、一実施形態において、処理要素とはソフトウェアスレッド、オペレーティングシステム、アプリケーション、または他のコードのようなコードに独立して関連付け可能な、任意のハードウェアを指す。物理プロセッサ（またはプロセッサソケット）とは通常、潜在的に複数のコアまたはハードウェアスレッドのような任意の数の他の処理要素を含む集積回路を指す。

10

【0092】

コアとはしばしば、独立のアーキテクチャ状態を維持可能な集積回路上に位置するロジックを指し、独立に維持される各アーキテクチャ状態は、少なくともいくつかの専用実行リソースと関連付けられる。複数のコアと対照的に、ハードウェアスレッドとは通常、独立のアーキテクチャ状態を維持可能な集積回路上に位置する任意のロジックを指し、その場合、複数の当該独立的に維持されるアーキテクチャ状態は複数の実行リソースへのアクセスを共有する。上記からわかるように、複数の特定のリソースが共有され、他のリソースがアーキテクチャ状態専用である場合、ハードウェアスレッドとコアとの用語体系間の境界は重複する。しかししばしば、コアとハードウェアスレッドとは、オペレーティングシステムによって、複数の個々の論理プロセッサとみられており、オペレーティングシステムは、各論理プロセッサ上での複数の動作を個々にスケジューリングすることができる。

20

【0093】

図15に示されるように、物理プロセッサ1500は2つのコア、すなわちコア1501および1502を含む。ここで、コア1501および1502は複数の対称的コア、すなわち複数の同一構成、機能ユニット、および／またはロジックを有する複数のコアと考えられる。別の実施形態において、コア1501はアウトオブオーダープロセッサコアを含む一方、コア1502はインオーダープロセッサコアを含む。しかしながら、コア1501および1502は、ネイティブコア、ソフトウェア管理コア、ネイティブ命令セットアーキテクチャ（ISA）を実行すべく適応されたコア、変換された命令セットアーキテクチャ（ISA）を実行すべく適合されたコア、共同設計されたコア、または他の既知のコアといった任意のタイプのコアから個々に選択されてよい。ヘテロジニアスなコア環境（すなわち、非対称コア）では、一方または両方のコアでコードをスケジューリングあるいは実行するためにバイナリ変換などの何らかの形態の変換が利用されてよい。更なる説明を進めるために、コア1501内に示される複数の機能ユニットが詳細に後述されるが、コア1502内の複数のユニットは当該示される実施形態で同様に動作する。

30

【0094】

図示される通り、コア1501は2つのハードウェアスレッド1501aおよび1501bを含み、それらはハードウェアスレッドスロット1501aおよび1501bとも称され得る。したがって、一実施形態において、オペレーティングシステムのような複数のソフトウェアエンティティは潜在的にプロセッサ1500を4つの個別のプロセッサ、すなわち4つのソフトウェアスレッドを同時に実行可能な4つの論理プロセッサまたは処理要素としてみなす。上記されたように、第1のスレッドは複数のアーキテクチャ状態レジスタ1501aと関連付けられ、第2のスレッドは複数のアーキテクチャ状態レジスタ1501bと関連付けられ、第3のスレッドは複数のアーキテクチャ状態レジスタ1502aと関連付けられてよく、第4のスレッドは複数のアーキテクチャ状態レジスタ1502bと関連付けられてよい。ここで、アーキテクチャ状態レジスタ（1501a、1501b、1502a、および1502b）の各々は、上述の複数の処理要素、スレッドスロット

40

50

ト、またはスレッドユニットと称され得る。図示の通り、複数のアーキテクチャ状態レジスタ1501aは複数のアーキテクチャ状態レジスタ1501b内で複製されているので、複数の個々のアーキテクチャ状態／コンテキストが論理プロセッサ1501aおよび論理プロセッサ1501bのために格納可能である。コア1501において、複数の命令ポインタおよびアロケータおよびリネームブロック1530内のリネームロジックのような他の複数のより小さなリソースはまた、スレッド1501aおよび1501bのために複製されてよい。リオーダ／リタイヤユニット1535、ILTB1520内の複数のリオーダバッファ、複数のロード／格納バッファ、および複数のキューなどのいくつかのリソースは、パーティショニングを介して共有されてよい。複数の汎用内部レジスタ、ページテーブルベースレジスタ、低レベルデータキャッシュおよびデータTLB1515、実行ユニット1540、およびアウトオブオーダユニット1535の複数の部分のような他の複数のリソースが潜在的に十分共有される。

10

【0095】

プロセッサ1500はしばしば、完全に共有、パーティショニングを介して共有、または複数の処理要素により／対して専用であってよい、他の複数のリソースを含む。図15において、プロセッサに係る例示の複数の論理ユニット／リソースを持つ純粋に例示的なプロセッサの一実施形態が示されている。プロセッサは、これらの機能ユニットのうち任意のものを含むか、省略してよく、並びに図示されていない複数の任意の他の既知の機能ユニット、ロジックまたはファームウェアを含んでよいことに留意されたい。図示の通り、コア1501は、簡易な代表的アウトオブオーダ(OOO)プロセッサコアを含む。し

20

【0096】

コア1501は更に、フェッチされた複数の要素をデコードするフェッチユニット1520に連結されたデコードモジュール1525を含む。フェッチロジックは、一実施形態において、スレッドスロット1501a、1501bにそれぞれ関連付けられた複数の個々のシーケンサを含む。通常、コア1501は、プロセッサ1500上で実行可能な複数の命令を定義／指定する第1のISAと関連付けられる。しばしば、第1のISAの一部である複数のマシンコード命令は、実行される命令または動作を参照／指定する命令の一部(オペコードと称される)を含む。デコードロジック1525は、それらのオペコードからこれらの命令を認識し、第1のISAによって定義される処理のために、デコードされた複数の命令をパイプラインに渡す回路を含む。例えば、詳細に後述される通り、一実施形態において、複数のデコーダ1525はトランザクション命令のような複数の特定の命令を認識すべく、設計または適合されたロジックを含む。デコーダ1525による認識の結果として、アーキテクチャまたはコア1501は、適切な命令と関連付けられた複数のタスクを実行すべく特定の、予め定義された複数の動作を行う。本明細書に記載される複数のタスク、ブロック、動作および方法のうち任意のものが、単一または複数の命令に

30

40

【0097】

一例において、アロケータおよびリネームブロック1530は、複数の命令処理結果を格納する複数のレジスタファイルのような複数のリソースを予約するアロケータを含む。しかしながら、スレッド1501aおよび1501bは潜在的にアウトオブオーダ実行が可能であり、そこではアロケータおよびリネームブロック1530はまた、複数の命令結果を追跡する複数のリオーダバッファのような他の複数のリソースを予約する。ユニット

50

1530はまた、プロセッサ1500内部の他の複数のレジスタに対し、プログラム／命令参照レジスタをリネームするレジスタリネームを含んでよい。リオーダ／リタイヤユニット1535は、アウトオブオーダー実行および後のアウトオブオーダー実行された複数の命令のインオーダーリタイアをサポートすべく、上記の複数のリオーダバッファ、ロードバッファ、および複数の格納バッファのような複数のコンポーネントを含む。

【0098】

一実施形態において、スケジューラおよび実行ユニットブロック1540は、複数の実行ユニットに対して複数の命令／動作をスケジューリングするスケジューラユニットを含んでよい。例えば、浮動小数点命令は、利用可能な浮動小数点実行ユニットを有する実行ユニットのポートにスケジューリングされる。複数の情報命令処理結果を格納すべく、複数の実行ユニットに関連付けられた複数のレジスタファイルも含まれる。複数の例示的実行ユニットとしては、浮動小数点実行ユニット、整数実行ユニット、ジャンプ実行ユニット、ロード実行ユニット、格納実行ユニット、および他の複数の既知の実行ユニットが含まれる。

10

【0099】

より低レベルのデータキャッシュおよびデータ変換バッファ（D-TLB）1550が、実行ユニット1540に連結される。データキャッシュは、複数のメモリコヒーレンシ状態に潜在的に保持された複数のデータオペランドのような、最近使用／動作されたものを複数の要素上格納する。D-TLBは、最近の仮想／線形アドレスから物理アドレスへの変換を格納する。具体例として、プロセッサは複数の仮想ページに物理メモリを分割するページテーブル構造体を含んでよい。

20

【0100】

ここでは、コア1501および1502は、オンチップインタフェース1510に関連付けられた第2のレベルのキャッシュのような、より高レベルまたはより遠いキャッシュへのアクセスを共有する。ここで、より高レベルまたはより遠いという用語は、実行ユニットから増加するまたは更に遠ざかる複数のキャッシュレベルのことであることに留意されたい。一実施形態において、より高レベルのキャッシュは、ラストレベルデータキャッシュ、すなわちプロセッサ1500のメモリ階層内の第2または第3のレベルデータキャッシュのようなラストキャッシュである。しかしながら、より高レベルのキャッシュは命令キャッシュに関連付けられ、または命令キャッシュを含んでよいので、そのように限定されない。むしろ、命令キャッシュのタイプであるトレースキャッシュが、複数の最近デコードされたトレースを格納すべく、デコーダ1525の後に連結されてよい。ここで、命令は潜在的にマクロ命令（すなわち、複数のデコーダによって認識される一般的命令）を指し、それは多数のマクロ命令（複数のマイクロオペレーション）にデコードされてよい。

30

【0101】

示される構成では、プロセッサ1500はオンチップインタフェースモジュール1510も含む。歴史的に、より詳細に後述されるメモリコントローラが、プロセッサ1500の外部にあるコンピューティングシステムに含まれている。このシナリオにおいて、オンチップインタフェース1510は、システムメモリ1575、チップセット（しばしばメモリ1575に接続するメモリコントローラハブおよび複数の周辺デバイスと接続するI/Oコントローラハブを含む）、メモリコントローラハブ、ノースブリッジ、または他の集積回路のようなプロセッサ1500の複数の外部デバイスと通信する。このシナリオにおいて、バス1505は、マルチドロップバス、ポイントツーポイント相互接続、シリアル相互接続、パラレルバス、コヒーレント（例えば、キャッシュコヒーレント）バス、層状プロトコルアーキテクチャ、差分バス、およびGTLバスのような任意の既知の相互接続を含んでよい。

40

【0102】

メモリ1575は、プロセッサ1500に専用であってもよいし、システム内の他の複数のデバイスと共有されてもよい。メモリ1575の複数のタイプの共通の例示は、DR

50

AM、SRAM、不揮発性メモリ（NVメモリ）、および複数の他の既知のストレージデバイスを含む。デバイス1580は、グラフィックアクセラレータ、プロセッサまたはメモリコントローラハブに連結されたカード、I/Oコントローラハブに連結されたデータストレージ、無線トランシーバ、フラッシュデバイス、オーディオコントローラ、ネットワークコントローラ、または他の既知のデバイスを含んでよいことに留意されたい。

【0103】

しかしながら、最近、より多くのロジックおよびデバイスがSOCなどの単一のダイ上に統合されるにつれ、これらのデバイスの各々はプロセッサ1500上に組み込まれてよい。例えば、一実施形態において、メモリコントローラハブはプロセッサ1500と同一パッケージ上および／または同一ダイ上に存在する。ここで、コア（オンコア部分）1510の一部は、メモリ1575またはグラフィクスデバイス1580のような他の複数のデバイスとのインタフェースを取るための1または複数のコントローラを含む。相互接続およびそのような複数のデバイスとのインタフェースを取るための複数のコントローラを含む構成は、しばしばオンコア（または非コア構成）と称される。一例として、オンチップインタフェース1510は、オンチップ通信のためのリング相互接続と、オフチップ通信のための高速シリアルポイントツーポイントリンク1505とを含む。しかし、SOC環境では、ネットワークインタフェース、複数のコプロセッサ、メモリ1575、グラフィクスプロセッサ1580、および複数の任意の他の既知のコンピュータデバイス／インタフェースのようなさらに多くのデバイスが、単一のダイまたは集積回路上に統合されてよく、高機能および低電力消費性を有するスモールフォームファクタを提供する。

【0104】

一実施形態において、プロセッサ1500は、本明細書に記載の複数の装置および方法をサポートし、それらとインタフェースを取るべく、アプリケーションコード1576をコンパイル、変換、および／または最適化する、コンパイラ、最適化、および／または変換コード1577を実行可能である。コンパイラはしばしば、ソーステキスト／コードをターゲットテキスト／コードに変換するプログラムまたはプログラムセットを含む。通常、コンパイラを用いるプログラム／アプリケーションコードのコンパイルは、複数のフェーズおよびパスで行われ、高レベルプログラミング言語コードを低レベルマシン若しくはアセンブリ言語コードに変換する。しかし、複数の単一パスのコンパイラが、簡易コンパイルのために依然利用されてよい。コンパイラは複数の任意の既知のコンパイル技術を利用してよく、語彙分析、前処理、解析、セマンティクス分析、コード生成、コード変換、およびコードの最適化のような複数の任意の既知のコンパイラ動作を実行してよい。

【0105】

より大型のコンパイラはしばしば複数のフェーズを含むが、しかしながら非常に頻繁に、これらのフェーズは、次の2つの一般フェーズ内に含まれる。（1）すなわち、概して構文処理、セマンティクス処理、およびいくつかの変換／最適化が発生し得るフロントエンド、および（2）すなわち、概して分析、複数の変換、複数の最適化、およびコード生成が発生するバックエンドである。いくつかのコンパイラはミドルエンドと呼ばれ、これはコンパイラのフロントエンドとバックエンドとの間の区別が曖昧であることを示す。結果として、挿入、関連付け、生成、またはコンパイラの他の動作への参照が上記フェーズまたはパスのうち任意のものにおいて、並びにコンパイラの任意の他の複数の既知のフェーズまたはパスにおいて発生してよい。例示のように、コンパイラは潜在的に複数の動作、呼び出し、機能等をコンパイルの1または複数のフェーズ内に挿入し、例えば、コンパイルのフロントエンドフェーズ内での複数の呼び出し／動作の挿入、およびその後の、変換フェーズ中の当該複数の呼び出し／動作のより低水準コードへの変換が挙げられる。動的コンパイル中、コンパイラコードまたは動的最適化コードは、そのような複数の動作／呼び出しを挿入してよく、並びにランタイム中の実行のために当該コードを最適化してよいことに留意されたい。具体的な例示としては、バイナリコード（既にコンパイルされたコード）をランタイム中に動的に最適化してよい。ここでプログラムコードは、動的最適化コード、バイナリコード、またはこれらの組み合わせが含まれてよい。

【0106】

コンパイラと同様に、バイナリトランスレータなどのトランスレータは、コードを最適化および／または変換すべく、コードを静的または動的に変換する。したがって、コード、アプリケーションコード、プログラムコード、または他のソフトウェア環境の実行に対する言及は、（１）プログラムコードをコンパイルするため、複数のソフトウェア構造を維持するため、他の複数の動作を実行するため、コードを最適化するため、またはコードを変換するための、コンパイラプログラム、最適化コードオプティマイザ、またはトランスレータの動的または静的な実行、（２）最適化／コンパイルされたアプリケーションコードのような複数の動作／呼出しを含むメインプログラムコードの実行、（３）複数のソフトウェア構造を維持するため、他のソフトウェア関連動作を行うため、またはコードを最適化するためのメインプログラムコードに関連付けられた複数のライブラリのような他のプログラムコードの実行、または（４）それらの組み合わせを指してよい。

10

【0107】

ここで図16を参照するに、マルチコアプロセッサの一実施形態のブロック図が示される。図16の実施形態に示されるプロセッサ1600は、複数のドメインを含む。具体的には、コアドメイン1630は、複数のコア1630A～1630N、メディアエンジン1665を有する1または複数のグラフィックスエンジンを含むグラフィックスドメイン1660、およびシステムエージェントドメイン1610を含む。

【0108】

様々な実施形態において、システムエージェントドメイン1610は、複数のパワー制御イベントおよび電力管理を処理し、その結果ドメイン1630および1660（例えば、複数のコアおよび／またはグラフィックスエンジン）に係る個々の複数のユニットは、特定のユニット内で発生するアクティビティ（または非アクティビティ）の観点から、適切な電力モード／レベル（例えば、アクティブ、ターボ、スリープ、休止状態、ディープスリープ、または他のAdvanced Configuration Power Interfaceのような状態）において動的に動作するよう独立的に制御可能である。ドメイン1630および1660の各々は、異なる電圧および／または電力で動作してよく、更に当該複数のドメイン内の個々の複数のユニットはそれぞれ、独立した周波数および電圧で潜在的に動作する。3つのドメインのみが示される一方、本発明の範囲はこの点に限定されないことを理解し追加の複数のドメインが複数の他の実施形態において存在し

20

30

【0109】

示される通り、各コア1630は、様々な実行ユニットおよび複数の追加の処理要素に加え、複数の低レベルキャッシュを更に含む。ここで、様々なコアは互いに、および、ラストレベルキャッシュ（LLC）1640A～1640Nから成る複数のユニットまたはスライスで形成される共有キャッシュメモリに連結される。これらのLLCはしばしば、ストレージおよびキャッシュコントローラ機能を含み、前記コア間で並びに潜在的にはグラフィックスエンジン間でも共有される。

【0110】

見ての通り、リング相互接続1650は前記複数のコアを互いに連結し、複数のリングストップ1652A～1652Nを介して、コアドメイン1630、グラフィックスドメイン1660、およびシステムエージェント回路1610間の相互接続を提供する。当該リングストップの各々は、コアとLLCスライスとの間の連結にある。図16に見られるように、相互接続1650は、アドレス情報、データ情報、受信確認情報、およびスヌープ／無効情報を含む様々な情報を搬送するために使用される。リング相互接続が図示されているが、任意の既知のオンダイ相互接続またはファブリックが利用されてよい。図の例示によると、上記複数のファブリック（例えば、別のオンダイ相互接続、オンチップシステムファブリック（OSF）、アドバンスドマイクロコントローラバスアーキテクチャ（AMBA）相互接続、多次元メッシュファブリック、または他の既知の相互接続アーキテクチャ）のうちいくつかが同様の態様で利用されてよい。

40

50

【0111】

更に図示の通り、システムエージェントドメイン1610は関連付けられたディスプレイの制御、およびそれへのインタフェースを提供するディスプレイエンジン1612を含む。システムエージェントドメイン1610は、システムメモリ（例えば、複数のDIMMと共に実装されたDRAM）へのインタフェースを提供する統合メモリコントローラ1620、複数のメモリコヒーレンシ動作を提供するコヒーレンシロジック1622のような他の複数のユニットを含んでよい。プロセッサと他の回路との間の相互接続を可能にすべく、複数のインタフェースが存在してよい。例えば、一実施形態において、少なくとも1つのダイレクトメディアインタフェース（DMI）1616インタフェース並びに1または複数のPCIe（登録商標）インタフェース1614が提供される。ディスプレイエンジンおよびこれらのインタフェースは通常、PCIe（登録商標）ブリッジ1618を介してメモリに連結する。更に、複数の追加のプロセッサまたは他の回路のような他の複数のエージェント間の複数の通信を提供すべく、1または複数の他のインタフェースが提供されてよい。

10

【0112】

本発明は限定数の実施形態に関連して説明されたが、当業者であればそこから多数の変更例および修正例を理解するであろう。添付の特許請求の範囲は、本発明の真の精神および範囲内に属するそのようなすべての変更例および修正例を包含することが意図されている。

【0113】

設計は、作成からシミュレーション、製造に至る様々な段階を経るであろう。設計を表わすデータは当該設計を多数の態様で表してよい。第1に、複数のシミュレーションで有用であるように、ハードウェア記述言語（HDL）または別の機能記述言語を使用してハードウェアが表現されてよい。さらに、ロジックおよび／または複数のトランジスタゲートを備える回路レベルモデルが、設計処理のいくつかの段階で生成されてよい。更に、ある段階における大半の設計はハードウェアモデル内の様々なデバイスの物理的配置を表すデータのレベルに到達している。従来の半導体製造技術が使用される場合、ハードウェアモデルを表すデータは、集積回路を生成すべく使用されるマスクのための複数の異なるマスク層上にある様々な特徴の存在または不存在を指定するデータであってよい。

20

【0114】

いくつかの実装例において、ソフトウェアベースの複数のハードウェアモデル、およびHDLおよび複数の他の機能記述言語オブジェクトは、複数の他の例の中でも特に、複数のレジスタトランスファ言語（RTL）ファイルを含むことができる。そのような複数のオブジェクトは、設計ツールがHDLオブジェクト（またはモデル）を受け入れ、記載されたハードウェアの複数の属性のためのHDLオブジェクトを解析し、物理回路および／またはオブジェクトのオンチップレイアウトを決定できるよう、機械解析可能であり得る。設計ツールの出力は、物理デバイスを製造すべく使用され得る。例えば、設計ツールは、HDLオブジェクトから様々なハードウェアおよび／またはファームウェア要素の複数の構成を決定できる。例えば、HDLオブジェクト内でモデル化されたシステムを実現すべく実装される他の複数の属性の中でも特に、複数のバス幅、レジスタ（複数のサイズおよびタイプを含む）、メモリブロック、物理リンクパス、ファブリックポートロジが挙げられる。複数の設計ツールとは、システムオンチップ（SOC）および他のハードウェアデバイスのトポロジおよびファブリック構成を決定するための複数のツールを含むことができる。いくつかの例において、HDLオブジェクトは、記載されたハードウェアの製造機器の製造に使用され得る複数のモデルおよび設計ファイルを開発する基礎として使用され得る。実際、HDLオブジェクト自体は、記載されたハードウェアをもたらず、製造システムソフトウェアへの入力として提供され得る。

30

40

【0115】

設計に係る任意の表現において、データは任意の形態の機械可読媒体に格納されてよい。メモリ、またはディスクなどの磁気若しくは光ストレージは、そのような情報を送信す

50

べく変調あるいは生成された光波または電波を介して送信される情報を格納する機械可読媒体であってよい。コードまたは設計を示す若しくは搬送する電気搬送波が送信される場合、電気信号のコピー、バッファリング、または再送が実行される範囲において、新規コピーが作成される。従って、通信プロバイダまたはネットワークプロバイダは有形な機械可読媒体上に少なくとも一時的に、搬送波にエンコードされた情報のようなアーティクルを格納してよく、本開示の複数の実施形態に係る複数の技術を具現する。

【0116】

本明細書において使用されるモジュールは、ハードウェア、ソフトウェア、および／またはファームウェアの任意の組み合わせを指してよい。一例としてモジュールは、マイクロコントローラのようなハードウェアを含む。当該ハードウェアはマイクロコントローラによって実行されるよう適合されたコードを格納する非一時的媒体に関連付けられる。したがって、一実施形態において、モジュールへの言及は非一時的媒体に保持されるコードを認識および／または実行するよう具体的に構成されたハードウェアを指す。更に、別の実施形態において、モジュールの使用とは、予め定められた複数の動作を実行するマイクロコントローラによって実行されるよう具体的に適合させられているコードを含む非一時的媒体を指す。予期され得るように、さらなる別の実施形態において、モジュールという用語（この例において）は、マイクロコントローラおよび非一時的媒体の組み合わせを指してよい。しばしば、個別のものとして示される複数のモジュールの境界は一般に変わり、潜在的に重複する。例えば、第1のモジュールおよび第2のモジュールがハードウェア、ソフトウェア、ファームウェア、またはこれらの組み合わせを共有する一方で、いくつかの独立したハードウェア、ソフトウェア、またはファームウェアを潜在的に維持してもよい。一実施形態において、ロジックという用語の使用には、複数のトランジスタ、レジスタ、または複数のプログラマブルロジックデバイスのような他のハードウェア等のハードウェアを含む。

【0117】

一実施形態において、「構成され」という文言の使用は、指定または決定されたタスクを実行する装置、ハードウェア、ロジック、または要素を配置すること、共に配置すること、製造すること、販売に供すること、輸入および／または設計することを指す。この例において、動作していない装置または当該装置の要素が当該指定されたタスクを実行すべく設計、連結、および／または相互接続されている場合、指定されたタスクを実行するよう依然として「構成され」る。純粹に例示として、ロジックゲートは動作中0または1を提供してよい。しかしながら、イネーブル信号をクロックに提供するよう「構成され」たロジックゲートは、1または0を提供し得るあらゆる潜在的ロジックゲートを含まない。その代わり、当該ロジックゲートは、動作中に1または0出力が当該クロックを有効にするよう何らかの態様で連結されたものである。用語「構成され」の使用は、動作を必要としないがその代わり、装置、ハードウェアおよび／または要素の潜在的な状態に重点を置く。その場合、潜在的な状態において、当該装置、ハードウェア、および／または要素が動作中の場合に当該装置、ハードウェア、および／または要素が特定のタスクを実行するよう設計されることに再度留意されたい。

【0118】

更に、「可能／すべく」および／または「動作可能」という文言の使用は、一実施形態において、装置、ロジック、ハードウェア、および／または要素の特定の態様での使用を可能にするよう設計された、いくつかの装置、ロジック、ハードウェア、および／または要素を指す。すべく、可能、または動作可能という文言の使用は一実施形態において、装置、ロジック、ハードウェア、および／または要素の潜在的な状態を指し、その場合、当該装置、ロジック、ハードウェアおよび／または要素は動作していないが、特定の態様での装置の使用を可能にするよう設計されていることに上記同様留意されたい。

【0119】

本明細書において使用される値とは、数字、状態、論理状態、またはバイナリ論理状態の任意の既知の表現を含む。しばしば、複数のロジックレベル、ロジック値、または論理

10

20

30

40

50

値の使用はまた、「1の」および「0の」とも称され、それは単に複数のバイナリロジック状態を表す。例えば、1は高論理レベル、および0は低論理レベルを指す。一実施形態において、トランジスタまたはフラッシュセルのような格納セルは、単一の論理値または複数の論理値を保持可能であってよい。しかしながら、複数のコンピュータシステムにおける複数の値の他の表現が使用されている。例えば、また10進法の数の10は、バイナリ値1010および16進法の文字Aで表現されてよい。したがって、ある値は、コンピュータシステムに保持可能な情報の任意の表現を含む。

【0120】

さらに、複数の状態は、複数の値または複数の値の部分によって表現されてよい。一例として、第1の値（例えば論理1）が既定または初期状態を表してよい一方で、第2の値（例えば論理0）が、非既定状態を表してよい。また、用語リセットおよびセットは一実施形態において、既定値および更新値または状態をそれぞれ指す。例えば、既定値は潜在的に高論理値、すなわちリセットを含む一方で、更新値は潜在的に低論理値、すなわちセットを含む。任意の数の状態を表わすべく、複数の値の任意の組み合わせが利用されてよいことに留意されたい。

【0121】

上述の複数の方法、ハードウェア、ソフトウェア、ファームウェアまたはコードに係る複数の実施形態は、機械アクセス可能、機械可読、コンピュータアクセス可能、または処理要素によって実行可能なコンピュータ可読媒体上に格納された複数の命令またはコードを介して実装されてよい。非一時的機械アクセス可能／可読の媒体は、コンピュータまたは電子システムのような機械によって読み取り可能な形態で情報を提供（すなわち、格納および／または送信）する任意のメカニズムを含む。例えば、非一時的機械アクセス可能媒体は、スタティックRAM（SRAM）若しくはダイナミックRAM（DRAM）のようなランダムアクセスメモリ（RAM）、ROM、磁気若しくは光ストレージ媒体、複数のフラッシュメモリデバイス、複数の電気ストレージデバイス、複数の光ストレージデバイス、複数の音響ストレージデバイス、複数の一時的（伝搬される）信号（例えば、複数の搬送波、赤外線信号、デジタル信号）から受信される情報を保持するための複数のストレージデバイスの他の形態等を含み、これらは、そこから情報を受信できる複数の非一時的媒体とは区別されるべきである。

【0122】

本発明の複数の実施形態を実行させるようロジックをプログラムすべく使用される複数の命令は、DRAM、キャッシュ、フラッシュメモリ、または他のストレージのようなシステム内のメモリ内部に格納されてよい。更に、複数の当該命令はネットワークを介して、または他のコンピュータ可読媒体によって配信され得る。従って、機械可読媒体は、機械（例えば、コンピュータ）によって読み取り可能な形態で情報を格納または送信するための任意のメカニズムを含んでよい。そのようなものとして、限定はされないが、複数のフロッピー（登録商標）ディスク、光ディスク、コンパクトディスク、リードオンリメモリ（複数のCD-ROM）、および光磁気ディスク、リードオンリメモリ（複数のROM）、ランダムアクセスメモリ（RAM）、消去可能プログラマブルリードオンリメモリ（EPROM）、電氣的消去可能プログラマブルリードオンリメモリ（EEPROM）、磁気または光カード、フラッシュメモリ、または電気形態、光形態、音響形態若しくは他の形態の伝搬信号（例えば、複数の搬送波、赤外線信号、デジタル信号等）を介したインターネット経由での情報送信に使用される有形の機械可読ストレージが挙げられる。従って、コンピュータ可読媒体は、機械（例えば、コンピュータ）によって読み取り可能な形態で複数の電子命令または情報を格納または送信するのに適した、任意のタイプの有形の機械可読媒体を含む。

【0123】

以下の複数の例は、本明細書による複数の実施形態に関する。1または複数の実施形態は、共有メモリに関連付けられたロード／格納タイプ動作に対応するためのデータをメモリアクセスリンクプロトコルに従うリンク経由で送信し、低電力状態に入るためのリクエ

10

20

30

40

50

ストを送信するための装置と、システムと、機械可読ストレージと、機械可読媒体と、ハードウェアおよび／またはソフトウェアベースロジックと、方法とを提供してよく、上記メモリアクセスリンクプロトコルは異なる別のリンクプロトコル上でオーバーレイされ、上記リクエストはトークンのフィールド内でエンコードされるデータ値を含み、上記トークンはパケットデータの開始を示し、上記トークンは更に上記トークンが上記別のリンクプロトコルおよび上記メモリアクセスリンクプロトコルのうちの1つに従うデータを含む後、後続のデータが送信されるかどうかを示す。

【0124】

少なくとも一例において、上記別のリンクプロトコルは、汎用入出力（I／O）相互接続プロトコルを含む。

10

【0125】

少なくとも一例において、トークンは汎用 I／O 相互接続プロトコルに従い定義される。

【0126】

少なくとも一例において、トークンの第1のフィールドが、上記後続のデータがメモリアクセスリンクプロトコルデータまたは汎用 I／O 相互接続プロトコルデータを含むかどうかを示す。

【0127】

少なくとも一例において、第1のフィールドは上記後続のデータがメモリアクセスリンクプロトコルデータを含むことを示す場合、上記データ値は上記トークンの第2のフィールド内でエンコードされる。

20

【0128】

少なくとも一例において、データ値はメモリアクセスリンクプロトコル下で定義されるエンコーディングを含み、第2のフィールドは汎用 I／O 相互接続プロトコルの物理層経由でエンコーディングをトンネリングすべく使用される。

【0129】

少なくとも一例において、上記リンク上で送信されるデータが、上記トークンの複数のインスタンスに基づいて、メモリアクセスリンクプロトコルデータと、汎用 I／O 相互接続プロトコルデータとの間で切り替わる。

【0130】

30

少なくとも一例において、電力管理ロジックは、低電力状態に入るためのリクエストを送信する前に、汎用 I／O 相互接続プロトコルのデータが送信されることを待機しているかどうかを確認する。

【0131】

少なくとも一例において、電力管理ロジックは、汎用 I／O 相互接続プロトコルデータが送信されるまで、低電力状態に入るためのリクエストを送信すべく待機する。

【0132】

少なくとも一例において、メモリアクセスリンクプロトコルまたは汎用 I／O 相互接続プロトコルのいずれかのリンク層が、物理層の低電力状態へのエントリをトリガできる。

【0133】

40

少なくとも一例において、汎用 I／O 相互接続プロトコル内で低電力状態に入るために使用される信号が、メモリアクセスリンクプロトコル内で低電力状態に入るために使用される信号とは異なる。

【0134】

少なくとも一例において、メモリアクセスリンクプロトコルは、メモリアクセスリンクプロトコル内の低電力状態に入るための信号をその中で送信する複数の周期的制御ウィンドウを定義する。

【0135】

1または複数の実施形態は、リンク経由でデータストリームを受信するための装置、システム、機械可読ストレージ、機械可読媒体、ハードウェアおよび／またはソフトウェア

50

ベースのロジック、および方法を提供してよい。データストリームは、トークンの後続のデータが、少なくとも2つの代替的なプロトコルのうちの1つに従うデータを含むことを示す第1のフィールドを含む、トークンを含むことができる。少なくとも2つの代替的なプロトコルは、メモリアクセスプロトコルおよび汎用I/Oプロトコルを含むことができる。リンク経由で送信されるすべてのデータは、汎用I/Oプロトコルの物理層経由で送信され、トークンの第2のフィールドがメモリアクセスプロトコルのリンク層による低電力リンク状態に入るためのリクエストを示すべくエンコードされる。低電力リンク状態へのエントリは、リクエストに基づいて開始され得る。

【0136】

少なくとも一例において、受信確認および否定的受信確認のうちの1つを含む応答がリクエストに対し送信され、上記応答はトークンの別のインスタンスで送信される。

10

【0137】

少なくとも一例において、上記応答は、上記トークンの別のインスタンス内の第2のフィールドのインスタンスが送信される。

【0138】

1または複数の実施形態は、第1のプロトコルのリンク層と、異なる第2のプロトコルのリンク層と、少なくとも第2のプロトコルのデータを送信するための第1のプロトコルの物理層と、シムとを提供する、装置、システム、機械可読ストレージ、機械可読媒体、ハードウェアおよび/またはソフトウェアベースのロジックを提供してよい。シムは、トークンの後続のデータが第1のプロトコルデータまたは第2のプロトコルデータのうちの1つを含むことを示す第1のフィールドを含むトークンを含むデータストリームを受信でき、トークンの第2のフィールドが第2のプロトコルのリンク層による低電力リンク状態に入るためのリクエストを示すべくエンコードされる。シムは更に、物理層を低電力状態に入らせるデータを物理層に送信できる。

20

【0139】

少なくとも一例において、第1のプロトコルは汎用I/O相互接続プロトコルを含み、第2のプロトコルはメモリアクセスプロトコルを含む。

【0140】

少なくとも一例において、第1のプロトコルの1または複数の層および第2のプロトコルの1または複数の層もまた含まれる。

30

【0141】

少なくとも一例において、第1のプロトコルのリンク層は無効にされる。

【0142】

本明細書の随所における「一実施形態」または「ある実施形態」という言及は、当該実施形態に関連して記載される特定の特徵、構造、または特性が本発明の少なくとも1つの実施形態に含まれることを意味する。従って、本明細書の様々な箇所における「一実施形態において」または「ある実施形態において」という文言は必ずしもすべて同一実施形態を指すわけではない。更に、複数の特定の特徵、構造、または特性は1または複数の実施形態において、任意の好適な態様で結合されてよい。

【0143】

40

上記の明細書においては、詳細な説明が複数の特定の例示的实施形態に関し提供される。しかしながら、添付の特許請求の範囲に記載された本発明のより広い精神および範囲から逸脱することなく、そこに様々な変更および修正が加えられ得ることは明らかである。本明細書および複数の図面は従って、限定的意味ではなく、例示の意味としてみなされるものである。更に、実施形態および他の説明の文言に係る上記使用は、必ずしも同一の実施形態または同一の例を指す必要はなく、複数の異なる別個の実施形態並びに潜在的に同一の実施形態を指してよい。本明細書に記載の発明は、以下の項目に記載の形態によっても実施され得る。

[項目1]

共有メモリに関連付けられたロード/格納タイプ動作に対応するデータを送信する送信

50

機と、

低電力状態に入るためのリクエストを送信する電力管理ロジックと、を備え、

前記データはメモリアクセスリンクプロトコルに従うリンク経由で送信され、前記メモリアクセスリンクプロトコルは異なる別のリンクプロトコル上にオーバレイされ、

前記リクエストはトークンのフィールド内でエンコードされたデータ値を含み、前記トークンはパケットデータの開始を示し、および前記トークンが前記別のリンクプロトコルおよび前記メモリアクセスリンクプロトコルのうちの1つに従うデータを含む後、後続のデータが送信されるかどうかを更に示す、装置。

[項目 2]

前記別のリンクプロトコルは、汎用入出力 (I/O) 相互接続プロトコルを含む、項目 1 に記載の装置。

[項目 3]

前記トークンは、前記汎用 I/O 相互接続プロトコルに従い定義される、項目 2 に記載の装置。

[項目 4]

前記トークンの第 1 のフィールドが、前記後続のデータがメモリアクセスリンクプロトコルデータを含むかまたは汎用 I/O 相互接続プロトコルデータを含むかを示す、項目 3 に記載の装置。

[項目 5]

前記第 1 のフィールドが、前記後続のデータがメモリアクセスリンクプロトコルデータを含むことを示す場合、前記データ値は前記トークンの第 2 のフィールド内でエンコードされる、項目 4 に記載の装置。

[項目 6]

前記データ値はメモリアクセスリンクプロトコル下で定義されたエンコーディングを含み、前記第 2 のフィールドは前記汎用 I/O 相互接続プロトコルの物理層経由で前記エンコーディングをトンネリングすべく使用される、項目 5 に記載の装置。

[項目 7]

前記リンク上で送信されるデータが、前記トークンの複数のインスタンスに基づいて、メモリアクセスリンクプロトコルデータと、汎用 I/O 相互接続プロトコルデータとの間で切り替わる、項目 2 に記載の装置。

[項目 8]

前記電力管理ロジックは、前記低電力状態に入るための前記リクエストを送信する前に、前記汎用 I/O 相互接続プロトコルのデータが送信されることを待機しているかどうかを確認する、項目 7 に記載の装置。

[項目 9]

前記電力管理ロジックは、前記汎用 I/O 相互接続プロトコルデータが送信されるまで、前記低電力状態に入るための前記リクエストを送信することを待機する、項目 8 に記載の装置。

[項目 10]

前記メモリアクセスリンクプロトコルまたは汎用 I/O 相互接続プロトコルのいずれかのリンク層が、物理層の低電力状態へのエントリをトリガできる、項目 7 に記載の装置。

[項目 11]

前記汎用 I/O 相互接続プロトコル内で低電力状態に入るために使用される信号が、前記メモリアクセスリンクプロトコル内で低電力状態に入るために使用される信号とは異なる、項目 10 に記載の装置。

[項目 12]

前記メモリアクセスリンクプロトコルは、前記メモリアクセスリンクプロトコル内の前記低電力状態に入るための信号をその中で送信する複数の周期的制御ウィンドウを定義する、項目 11 に記載の装置。

[項目 13]

10

20

30

40

50

リンク経由でデータストリームを受信する受信機と、
リクエストに基づいて低電力リンク状態へのエントリを開始するプロトコルロジックと
、を備え、

前記データストリームはトークンを含み、前記トークンは前記トークンの後続のデータ
が少なくとも2つの代替的なプロトコルのうちの1つに従うデータを含むことを示す第1
のフィールドを含み、前記少なくとも2つの代替的なプロトコルはメモリアクセスプロト
コルおよび汎用I/Oプロトコルを含み、前記リンク経由で送信されるすべてのデータは
前記汎用I/Oプロトコルの物理層経由で送信され、前記トークンの第2のフィールドが
前記メモリアクセスプロトコルのリンク層による前記低電力リンク状態に入るための前記
リクエストを示すべくエンコードされる、装置。

10

[項目14]

前記リクエストに対する応答を送信する送信機を更に備え、
前記応答は受信確認および否定的受信確認のうちの1つを含み、前記応答は前記トーク
ンの別のインスタンスで送信される、項目13に記載の装置。

[項目15]

前記応答は前記トークンの前記別のインスタンス内で前記第2のフィールドのインスタ
ンスが送信される、項目14に記載の装置。

[項目16]

リンク経由でデータストリームを受信する段階と、
リクエストに対する応答を送信する段階と、

20

前記リクエストに基づいて低電力リンク状態へのエントリを開始する段階と、を備え、
前記データストリームはトークンの第1のインスタンスを含み、前記トークンの複数の
インスタンスは前記トークンの後続のデータが少なくとも2つの代替的なプロトコルのう
ちの1つに従うデータを含むことを示す第1のフィールドを含み、前記少なくとも2つの
代替的なプロトコルはメモリアクセスプロトコルおよび汎用I/Oプロトコルを含み、前
記リンク経由で送信されるすべてのデータは前記汎用I/Oプロトコルの物理層経由で送
信され、前記トークンの前記第1のインスタンスの第2のフィールドが前記メモリアクセ
スプロトコルのリンク層による前記低電力リンク状態に入るための前記リクエストを示す
べくエンコードされ、

前記応答は受信確認および否定的受信確認のうちの1つを含み、前記応答は前記トーク
ンの別のインスタンスで送信される、方法。

30

[項目17]

第1のプロトコルのリンク層と、
異なる第2のプロトコルのリンク層と、
前記第1のプロトコルの物理層と、
データストリームを受信し、前記物理層を低電力リンク状態に入らせるデータを前記物
理層に送信するシムと、を備え、

前記第2のプロトコルのデータが前記第1のプロトコルの前記物理層経由で送信され、
前記データストリームはトークンを含み、前記トークンは前記トークンの後続のデータ
が第1のプロトコルデータまたは第2のプロトコルデータのうちの1つを含むことを示す
第1のフィールドを含み、前記トークンの第2のフィールドが前記第2のプロトコルの前
記リンク層による前記低電力リンク状態に入るためのリクエストを示すべくエンコードさ
れる、装置。

40

[項目18]

前記第1のプロトコルは汎用I/O相互接続プロトコルを含み、前記第2のプロトコル
はメモリアクセスプロトコルを含む、項目17に記載の装置。

[項目19]

前記第1のプロトコルの1または複数の層および前記第2のプロトコルの1または複数
の層を更に備える、項目17または18に記載の装置。

[項目20]

50

前記第 1 のプロトコルの前記リンク層は無効にされる、項目 17 から 19 のいずれか一項に記載の装置。

〔項目 21〕

リンク経由でデータストリームを受信する手段と、

リクエストに対する応答を送信する手段と、

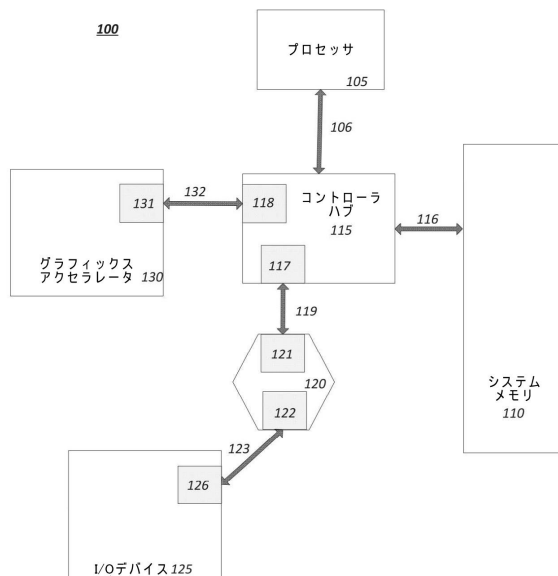
前記リクエストに基づいて低電力リンク状態へのエントリを開始する手段と、を備え、

前記データストリームはトークンの第 1 のインスタンスを含み、前記トークンの複数のインスタンスは前記トークンの後続のデータが少なくとも 2 つの代替的なプロトコルのうちの 1 つに従うデータを含むことを示す第 1 のフィールドを含み、前記少なくとも 2 つの代替的なプロトコルはメモリアクセスプロトコルおよび汎用 I/O プロトコルを含み、前記リンク経由で送信されるすべてのデータは前記汎用 I/O プロトコルの物理層経由で送信され、前記トークンの前記第 1 のインスタンスの第 2 のフィールドが前記メモリアクセスプロトコルのリンク層による前記低電力リンク状態に入るための前記リクエストを示すべくエンコードされ、

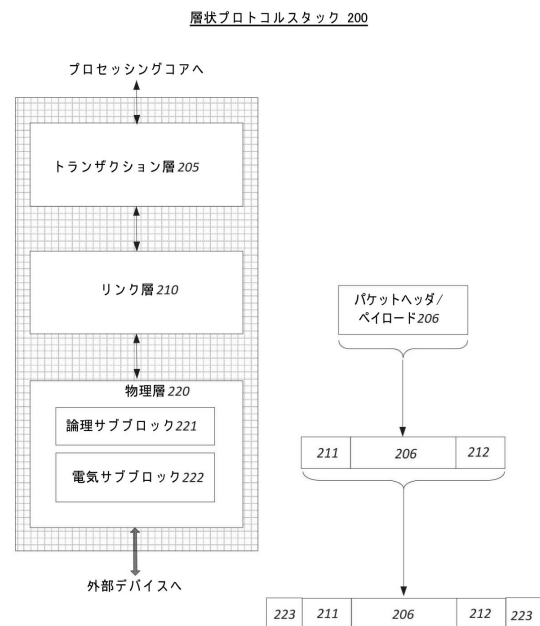
前記応答は受信確認および否定的受信確認のうちの 1 つを含み、前記応答は前記トークンの別のインスタンスで送信される、システム。

10

【図 1】



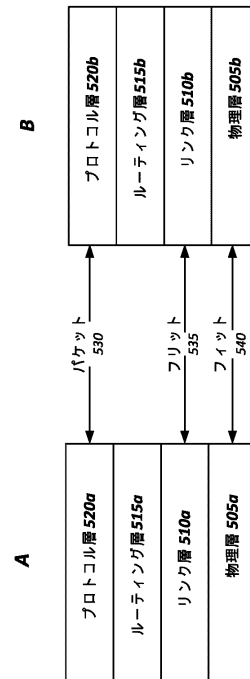
【図 2】



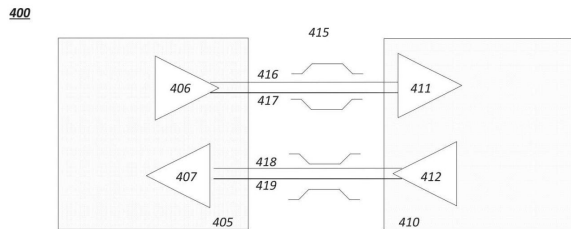
【図 3】



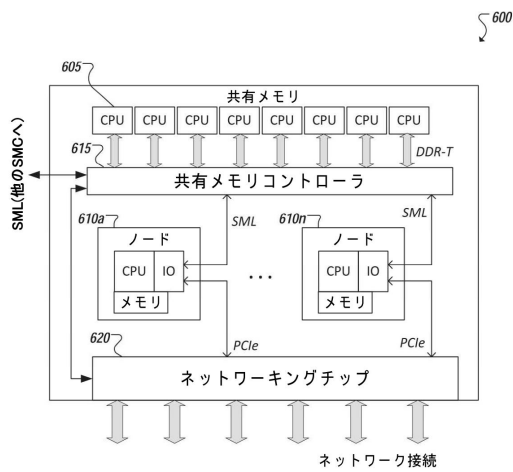
【図 5】



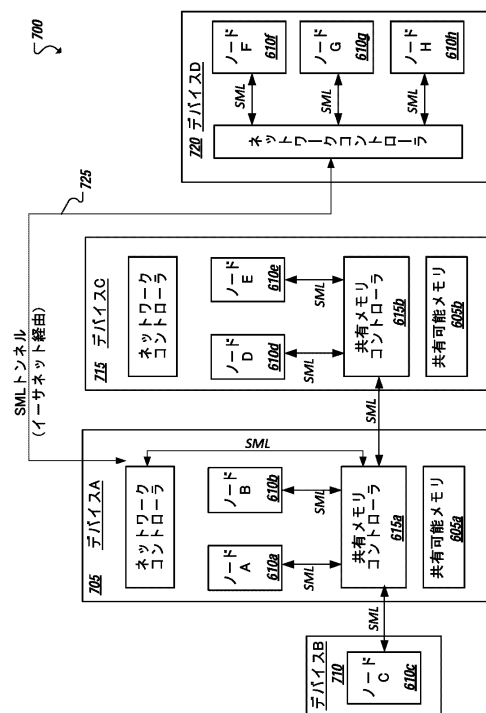
【図 4】



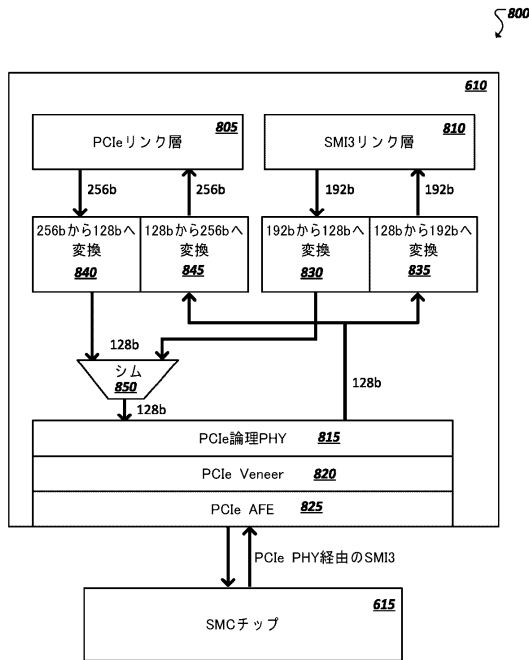
【図 6】



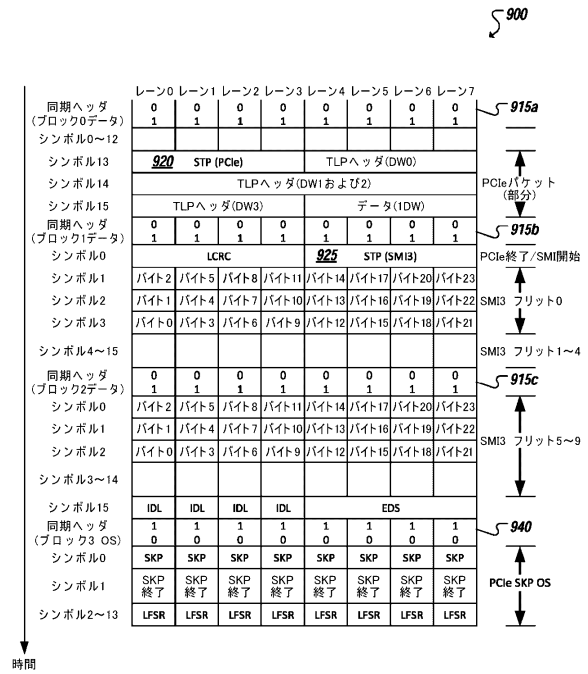
【図 7】



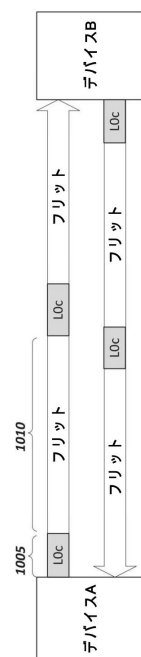
【図 8】



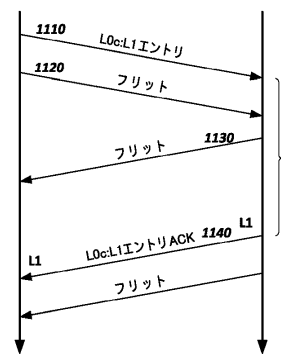
【図 9】



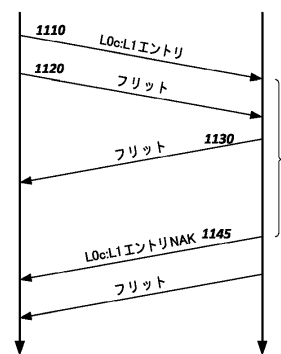
【図 10】



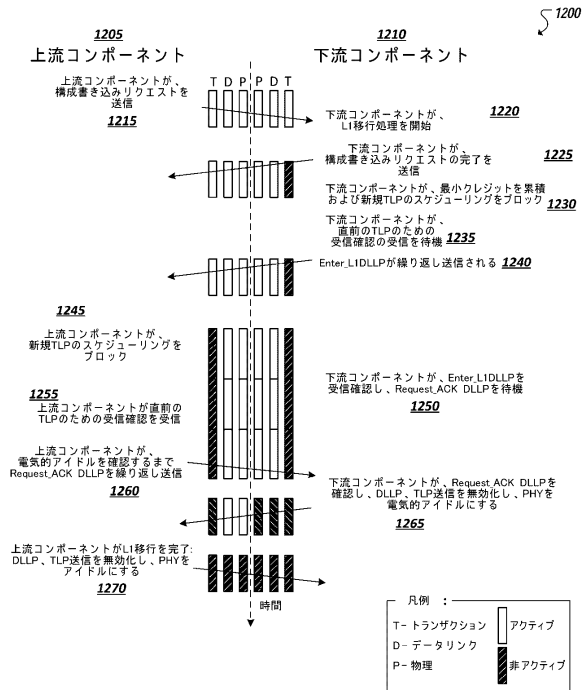
【図 11 A】



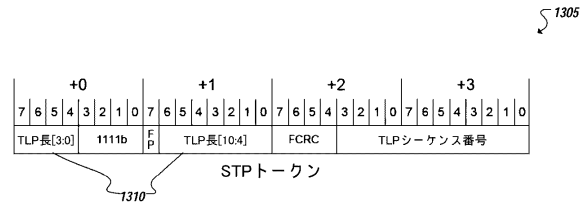
【図 11 B】



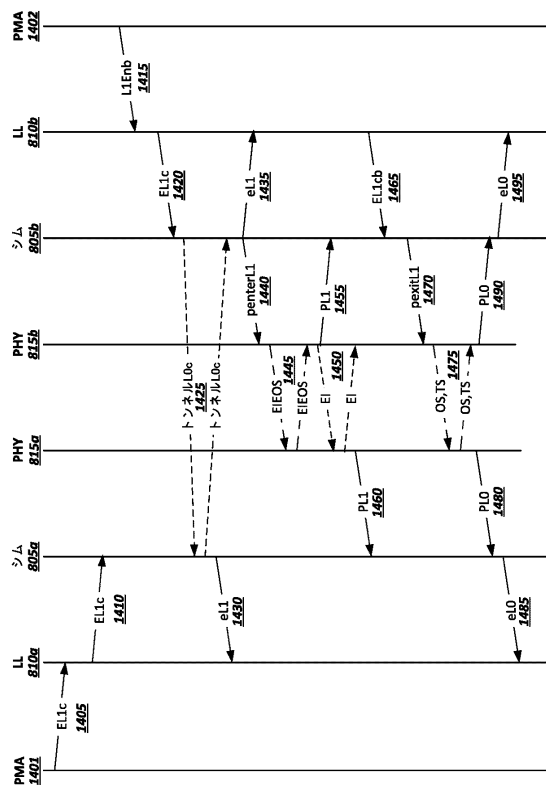
【図 12】



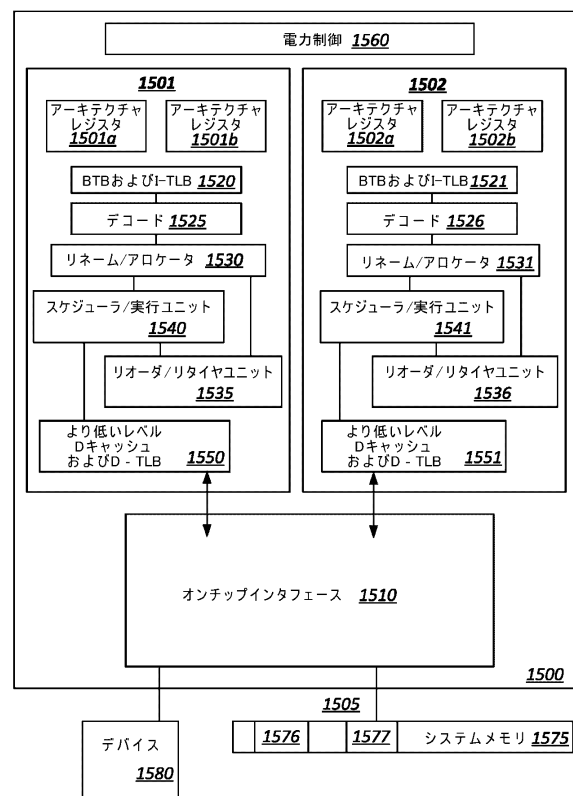
【図 13】



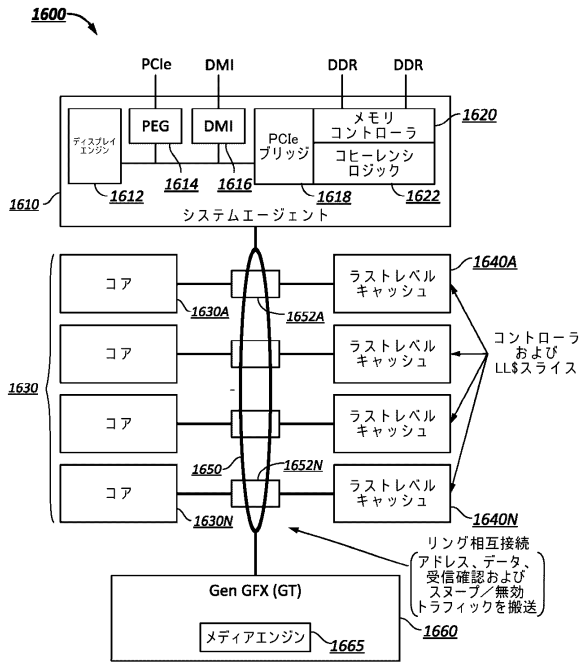
【図 14】



【図 15】



【図 16】



フロントページの続き

- (72)発明者 ダス シャーマ、デベンドラ
アメリカ合衆国 95054 カリフォルニア州・サンタクララ・ミッション カレッジ ブーレ
バード・2200 インテル・コーポレーション内
- (72)発明者 ワグ、マヘシュ
アメリカ合衆国 95054 カリフォルニア州・サンタクララ・ミッション カレッジ ブーレ
バード・2200 インテル・コーポレーション内
- (72)発明者 アイアー、ベンカトラマン
アメリカ合衆国 95054 カリフォルニア州・サンタクララ・ミッション カレッジ ブーレ
バード・2200 インテル・コーポレーション内

審査官 谷岡 佳彦

- (56)参考文献 米国特許出願公開第2014/0112339 (US, A1)
特開2011-065650 (JP, A)
特開2006-074769 (JP, A)

- (58)調査した分野(Int.Cl., DB名)
H04L 12/28
G06F 15/173
G06F 15/167