(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7502338号
(P7502338)

(45)発行日　令和6年6月18日(2024.6.18)

(24)登録日　令和6年6月10日(2024.6.10)

(51)国際特許分類

		FI		
Ｇ０６Ｆ	*9/50 (2006.01)*	Ｇ０６Ｆ	9/50	１５０Ｃ
Ｇ０６Ｆ	*9/455(2018.01)*	Ｇ０６Ｆ	9/455	１５０

請求項の数　20　（全14頁）

(21)出願番号　特願2021-569031(P2021-569031)
(86)(22)出願日　令和2年4月16日(2020.4.16)
(65)公表番号　特表2022-533997(P2022-533997A)
(43)公表日　令和4年7月27日(2022.7.27)
(86)国際出願番号　PCT/US2020/028603
(87)国際公開番号　WO2020/236363
(87)国際公開日　令和2年11月26日(2020.11.26)
審査請求日　令和5年4月14日(2023.4.14)
(31)優先権主張番号　62/850,421
(32)優先日　令和1年5月20日(2019.5.20)
(33)優先権主張国・地域又は機関
米国(US)
(31)優先権主張番号　16/808,286
(32)優先日　令和2年3月3日(2020.3.3)
最終頁に続く

(73)特許権者　314015767
マイクロソフト テクノロジー ライセンシング，エルエルシー
アメリカ合衆国 ワシントン州 ９８０５２ レッドモンド ワン マイクロソフト ウェイ
(74)代理人　100118902
弁理士　山本 修
(74)代理人　100106208
弁理士　宮前 徹
(74)代理人　100196508
弁理士　松尾 淳一
(72)発明者　チオウ，デレク
アメリカ合衆国 ワシントン州 ９８０５２－６３９９ レッドモンド ワン マイ
最終頁に続く

(54)【発明の名称】　ＳＯＣおよびＦＰＧＡを有するサーバオフロードカード

(57)【特許請求の範囲】
【請求項１】
サーバであって、
ＣＰＵ（中央処理装置）コンプレックスと、
オフロードカードであって、
ＳｏＣ（システムオンチップ）と、
前記ＳｏＣの外部にあり、且つ、前記ＳｏＣに連結されたＦＰＧＡ（フィールドプログラマブルゲートアレイ）と
を含むオフロードカードと、
を備え、
前記ＣＰＵコンプレックスが、１つまたは複数の仮想マシン（ＶＭ）を実行するように構成され、
前記ＳｏＣが、前記１つまたは複数のＶＭに関連付けられたハイパーバイザの１つまたは複数の第１の機能をソフトウェアで実行するように構成され、
前記ＦＰＧＡが、前記１つまたは複数のＶＭに関連付けられた前記ハイパーバイザの１つまたは複数の第２の機能をハードウェアで実行するように構成される、
サーバ。
【請求項２】
請求項１に記載のサーバであって、前記ＳｏＣおよび前記ＦＰＧＡが、前記オフロードカードの内部にあるＰＣＩｅ（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

）インターフェースを介して、および、前記オフロードカードの内部にあるイーサネットインターフェースを介して、互いに通信可能に連結される、サーバ。
【請求項３】
請求項２に記載のサーバであって、前記ＳｏＣおよび前記ＦＰＧＡが、前記オフロードカードの内部にあるＪＴＡＧ（ジョイントテストアクショングループ）インターフェースを介して、互いに通信可能に連結される、サーバ。
【請求項４】
請求項１に記載のサーバであって、前記オフロードカードが、前記サーバのメインボードにＰＣＩｅエッジコネクタインターフェースを介して挿入される、サーバ。
【請求項５】
請求項４に記載のサーバであって、前記ＳｏＣが、前記サーバのベースボードマネージメントコントローラ（ＢＭＣ）に、前記ＰＣＩｅエッジコネクタインターフェースを通じて通信可能に連結される、サーバ。
【請求項６】
請求項４に記載のサーバであって、前記ＦＰＧＡが、前記ＣＰＵコンプレックスと前記ＰＣＩｅエッジコネクタインターフェースを通じて通信可能に連結される、サーバ。
【請求項７】
請求項１に記載のサーバであって、前記ＳｏＣが、前記オフロードカード上にある１つまたは複数の揮発性メモリモジュールと通信可能に連結され、前記１つまたは複数の揮発性メモリモジュールが、前記ＳｏＣが前記１つまたは複数の第１の機能をそこから実行できる作業メモリとして機能する、サーバ。
【請求項８】
請求項１に記載のサーバであって、前記ＳｏＣが、前記オフロードカード上にあるフラッシュメモリモジュールと通信可能に連結され、前記フラッシュメモリモジュールが、前記１つまたは複数の第１の機能のためのプログラムコードを格納する、サーバ。
【請求項９】
請求項１に記載のサーバであって、前記ＦＰＧＡが、前記オフロードカード上にある１つまたは複数の揮発性メモリモジュールと通信可能に連結され、前記１つまたは複数の揮発性メモリモジュールが、前記１つまたは複数の第２の機能を実行する時に前記ＦＰＧＡのための作業メモリとして機能する、サーバ。
【請求項１０】
請求項１に記載のサーバであって、前記ＦＰＧＡが、前記オフロードカード上にあるフラッシュメモリモジュールと通信可能に連結され、前記フラッシュメモリモジュールが、１つまたは複数の第２の機能を実行するように前記ＦＰＧＡを構成するための少なくとも１つの構成イメージを格納する、サーバ。
【請求項１１】
請求項１０に記載のサーバであって、前記フラッシュメモリモジュールが、前記ＦＰＧＡのための正常動作構成に対応する第１の構成イメージ、および、前記ＦＰＧＡのためのフェールセーフ動作構成に対応する第２の構成イメージを格納する、サーバ。
【請求項１２】
請求項１１に記載のサーバであって、前記第１の構成イメージが、前記オフロードカードの電源投入時にデフォルトで前記ＦＰＧＡに適用される、サーバ。
【請求項１３】
請求項１２に記載のサーバであって、前記第１の構成イメージを適用している間にエラーが発生した場合、前記第２の構成イメージが、前記ＦＰＧＡに適用される、サーバ。
【請求項１４】
請求項１に記載のサーバであって、前記ＦＰＧＡが、ＴＯＲ（トップオブラック）ネットワークスイッチと通信可能に連結された第１の外部ネットワークインターフェース、および、前記サーバのＮＩＣ（ネットワークインターフェースカード）と通信可能に連結された第２の外部ネットワークインターフェースを含む、サーバ。

【請求項１５】
請求項１に記載のサーバであって、前記ＳｏＣが、前記オフロードカード上にあるＢＩＯＳ（ベーシックインプット／アウトプット）フラッシュ構成要素に、セキュリティチップを介して通信可能に連結され、前記セキュリティチップが、前記ＢＩＯＳフラッシュ構成要素に格納されたファームウェアの完全性を検証するように構成される、サーバ。
【請求項１６】
請求項１に記載のサーバであって、前記１つまたは複数の第１の機能が、ネットワーク制御プレーン機能またはストレージ制御プレーン機能を含む、サーバ。
【請求項１７】
請求項１に記載のサーバであって、前記１つまたは複数の第２の機能が、ネットワークデータプレーン機能またはストレージデータプレーン機能を含む、サーバ。
【請求項１８】
サーバであって、
１つまたは複数の仮想マシン（ＶＭ）を実行するように構成されたＣＰＵ（中央処理装置）コンプレックスと、
オフロードカードであって、
前記１つまたは複数のＶＭに関連付けられたハイパーバイザの１つまたは複数の第１の機能をソフトウェアで実行するための手段と、
前記１つまたは複数のＶＭに関連付けられた前記ハイパーバイザの１つまたは複数の第２の機能をハードウェアで実行するための手段と、
を含むオフロードカードと
を備えるサーバ。
【請求項１９】
方法であって、
サーバのオフロードカード上にあるＦＰＧＡ（フィールドプログラマブルゲートアレイ）が、前記サーバのＮＩＣ（ネットワークインターフェースカード）からネットワークパケットを受け取るステップであって、前記ネットワークパケットが、前記ＦＰＧＡと前記ＮＩＣを相互接続するイーサネットインターフェースを介して受け取られる、ステップと、
前記ＦＰＧＡが、前記ネットワークパケットのヘッダに基づいてフローテーブルへのルックアップをハードウェアで実施するステップと、
マッチするエントリが前記ヘッダのための前記フローテーブル内で見つからなかったと判定すると、前記ＦＰＧＡが、前記オフロードカード上にあるＳｏＣ（システムオンチップ）に前記ネットワークパケットを転送するステップであって、前記ネットワークパケットが、前記ＦＰＧＡと前記ＳｏＣを相互接続するイーサネットインターフェースを介して転送される、ステップと、
前記ＳｏＣが、前記ネットワークパケットのためのネクストホップの宛先をソフトウェアで計算するステップと、
前記ＳｏＣが、前記ネクストホップの宛先を含む新しいフローエントリで前記フローテーブルをソフトウェアで更新するステップと
を含む方法。
【請求項２０】
請求項１９に記載の方法であって、マッチするエントリが前記フローテーブル内で見つかったと判定すると、
前記ＦＰＧＡが、前記マッチするエントリに基づいて前記ネットワークパケットを更新するステップと、
前記ＦＰＧＡが、前記ＦＰＧＡの外部ネットワークインターフェースを介して外部ネットワークに前記ネットワークパケットを伝送するステップと、
をさらに含む方法。
【発明の詳細な説明】
【背景技術】

【０００１】
[0001]Ｍｉｃｒｏｓｏｆｔ　ＡｚｕｒｅおよびＡｍａｚｏｎ　ＡＷＳなどのクラウドプラットフォームは、地理的に拡散されたデータセンタの全体にわたって分散された物理サーバの大きい集団（本明細書ではクラウドサーバと呼ばれる）の上で動く。これらのクラウドサーバのかなりの部分が、仮想マシン（ＶＭ）のホスティングを可能にするハイパーバイザとして知られる仮想化ソフトウェアレイヤを実装する。数ある中でも、仮想化ソフトウェアレイヤは、クラウドプラットフォームの顧客が購入し、ＶＭを使用して顧客のアプリケーションワークロードを実行するＩａａＳ（サービスとしてのインフラストラクチャ）シナリオを可能にする。
【発明の概要】
【発明が解決しようとする課題】
【０００２】
[0002]伝統的に、ハイパーバイザを実装する各クラウドサーバでは、クラウドサーバのＣＰＵ（中央処理装置）コアの一定の割合が、ハイパーバイザ用に確保される。この確保は、ハイパーバイザがその機能を実行するのに十分な計算リソースを有することを保証するが、例えば顧客ＶＭによる使用のために利用可能なＣＰＵコアの数も低減させる。規模の点で、これは、顧客とじかに接するクラウドプラットフォームの全般的な計算容量の大幅な低下になる恐れがある。
【課題を解決するための手段】
【０００３】
[0003]ＳｏＣ（システムオンチップ（ｓｙｓｔｅｍ－ｏｎ－ｃｈｉｐ））およびＦＰＧＡ（フィールドプログラマブルゲートアレイ（ｆｉｅｌｄ　ｐｒｏｇｒａｍｍａｂｌｅ　ｇａｔｅ　ａｒｒａｙ））を含むオフロードシステムを有する物理サーバが開示される。オフロードシステムの１つの可能な実施形態はカード上にある。実施形態の１つのセットによれば、ＳｏＣは、ソフトウェアでの実行に適したサーバのＣＰＵコンプレックスから１つまたは複数のハイパーバイザ機能をオフロードするように構成することができ、ＦＰＧＡは、ハードウェアでの実行に適したＣＰＵコンプレックスから１つまたは複数のハイパーバイザ機能をオフロードするように構成することができる。
【図面の簡単な説明】
【０００４】
【図１】[0004]いくつかの実施形態による、ＳｏＣおよびＦＰＧＡを有するオフロードカードを含む物理サーバトポロジを描写する図である。
【図２】[0005]いくつかの実施形態による、図１のオフロードカードのためのアーキテクチャを描写する図である。
【図３】[0006]いくつかの実施形態による、ＪＴＡＧ（ジョイントテストアクショングループ（Ｊｏｉｎｔ　Ｔｅｓｔ　Ａｃｔｉｏｎ　Ｇｒｏｕｐ））多重化装置の実装形態を描写する図である。
【図４】[0007]いくつかの実施形態による、実例のネットワーク処理フロー図である。
【発明を実施するための形態】
【０００５】
[0008]以下の説明では、説明のために、非常に多くの例および詳細が、様々な実施形態を理解するために示されている。それでも、いくつかの実施形態を、これらの詳細の一部がなくても実践できること、または、変更形態もしくはその同等物で実践できることが当業者には明らかであろう。
【０００６】
１．全体像
[0009]本開示の実施形態は、ＳｏＣ（システムオンチップ）およびＦＰＧＡ（フィールドプログラマブルゲートアレイ）を備えるオフロードカードを用いる物理サーバ設計を対象とする。様々な実施形態では、ＳｏＣおよびＦＰＧＡは、サーバのＣＰＵコンプレックスによって伝統的に実行されるハイパーバイザ機能を動かすことができ、これにより、こ

れらの機能の処理負担をＣＰＵコンプレックスからオフロードする。例えば、オフロードカードのＳｏＣは、汎用プロセッサの柔軟性（例えばネットワーキングおよびストレージ制御プレーン機能）を必要とするか、汎用プロセッサの柔軟性から利益を得るハイパーバイザ機能を動かすことができるが、オフロードカードのＦＰＧＡは、ハードウェア（例えばネットワーキングおよびストレージデータプレーン機能）での実装／アクセラレーションに適したハイパーバイザ機能を動かすことができる。

【０００７】

[0010]この全体的なアーキテクチャにより、全てではないがほとんどのハイパーバイザ処理をサーバのＣＰＵコンプレックスからオフロードカードに移すことができ、オフロードカードは、都合のよいことに、ＣＰＵコンプレックスがテナント（例えば顧客）ＶＭワークロードを実行することに集中することを可能にする。ハイパーバイザがＣＰＵコンプレックスから完全に立ち退かされるケースでは、テナントコードは、ＣＰＵコンプレックス上で「ベアメタル」のように（すなわち介在するいかなるハイパーバイザ仮想化レイヤもない状態で）動かせる可能性がある。

【０００８】

[0011]なおさらに、オフロードカード上でのハイパーバイザコード／ロジックの実行は、ＣＰＵコンプレックス上のテナントコードの実行から物理的に切り離されるので、このソリューションは、テナントコードを攻撃ベクトルとして使用しようとし得るサイドチャネル攻撃からハイパーバイザを保護する。

【０００９】

[0012]さらに、ハードウェア実装に対応可能な一定のハイパーバイザ機能を加速させるためにＦＰＧＡを用いることによって、オフロードカードは、構造的柔軟性を同時に維持しつつ、サーバの効率性を改善することができる。例えば、必要ならＦＰＧＡは、１つのタイプ／クラスの機能（例えばネットワーキング）の加速から、別のタイプ／クラスの機能（例えばストレージ）の加速にプログラムし直すことができる。これは、ＡＳＩＣ（特定用途向け集積回路）などのハードロジックベースのアクセラレータでは可能ではない。

【００１０】

[0013]本開示の前述および他の態様が、以下のセクションでさらに詳細に説明される。

２．サーバトポロジ

[0014]図１は、本開示の一定の実施形態による、物理サーバ１００の高レベルトポロジを示す簡易ブロック図である。実施形態の１つのセットでは、物理サーバ１００は、クラウドプラットフォームのインフラストラクチャの一部として導入されたクラウドサーバでよい。これらの実施形態では、物理サーバ１００は、クラウドプラットフォームプロバイダによって運用されるデータセンタ内のサーバラックにマウントされてよい。他の実施形態では、物理サーバ１００は、例えばスタンドアロンサーバの形で、敷地内の企業ＩＴ環境など、他の状況でおよび／または他のフォームファクタを介して、導入されてよい。

【００１１】

[0015]背景技術セクションで述べたように、クラウドサーバは仮想化のためにハイパーバイザを実装することが多く、これによりクラウドプラットフォームが、ＩａａＳ（サービスとしてのインフラストラクチャ）などのサービスを提供することができる。それでも、ハイパーバイザ（「ホスト」としても知られる）用に、ＣＰＵコアを含むこれらのプラットフォームリソースの一部を使用することにより、従来のクラウドサーバは、これらのＣＰＵ容量の全てをＶＭに露出することができず、これによりプラットフォームの効率性を低下させる。

【００１２】

[0016]この問題および他の問題に対処するために、物理サーバ１００は、ＳｏＣ１０４およびＦＰＧＡ１０６を備える斬新なオフロードカード１０２を含む。図示の実施形態では、オフロードカード１０２は、ＰＣＩｅ（ペリフェラルコンポーネントインターフェースエクスプレス（Ｐｅｒｉｐｈｅｒａｌ Ｃｏｍｐｏｎｅｎｔ Ｉｎｔｅｒｆａｃｅ Ｅｘｐｒｅｓｓ））ベースの拡張カードとして実装され、したがって、標準ＰＣＩｅ ｘｌ６

３．０エッジコネクタインターフェース１０８を介して物理サーバ１００のメインボードとインターフェースする。他の実施形態では、オフロードカード１０２は、周辺インターフェースの他のいずれかのタイプを使用して実装されてもよい。

【００１３】

[0017]図示のように、ＳｏＣ１０４は、独自のＲＡＭ（ランダムアクセスメモリ（ｒａｎｄｏｍ　ａｃｃｅｓｓ　ｍｅｍｒｏｙ））１１０およびフラッシュメモリ１１２を有し、オフロードカード１０２の内部にある少なくともインターフェース（すなわちＰＣＩｅインターフェース１１４およびイーサネットインターフェース１１６）を介してＦＰＧＡ１０６と通信可能に連結される。さらにＳｏＣ１０４は、Ｉ２Ｃインターフェース１０８およびいくつかの他のチャネル（例えばＵＳＢおよびＣＯＭ）を通じて、物理サーバ１００のベースボードマネージメントコントローラ（ＢＭＣ：ｂａｓｅｂｏａｒｄ　ｍａｎａｇｅｍｅｎｔ　ｃｏｎｔｒｏｌｌｅｒ）１１８と通信可能に連結される。

【００１４】

[0018]ＦＰＧＡ１０６も、独自のＲＡＭ１２０およびフラッシュメモリ１２２を有し、ＰＣＩｅエッジコネクタインターフェース１０８を通じて物理サーバ１００のＣＰＵコンプレックス１２４と通信可能に連結される。このＣＰＵコンプレックスは、物理サーバ１００のメインＣＰＵコアおよび関連付けられたＲＡＭモジュールを備える。さらにＦＰＧＡ１０６は、２つの外部イーサネットインターフェースを含み、これらの一方は、（例えばＴＯＲ（トップオブラック）スイッチまたは他のいくつかのネットワークデバイスを介して）外部ネットワーク１２６に接続し、他方は、物理サーバ１００内のＮＩＣ（ネットワークインターフェースカード／コントローラ（ｎｅｔｗｏｒｋ　ｉｎｔｅｒｆａｃｅ　ｃａｒｄ／ｃｏｎｔｒｏｌｌｅｒ））１２８に接続する。

【００１５】

[0019]一般的に言えば、図１に示されたトポロジは、物理サーバ１００のＣＰＵコンプレックス１２４上で伝統的に動かされるハイパーバイザ機能の一部または全てをオフロードカード１０２のＳｏＣ１０４およびＦＰＧＡ１０６上で代わりに動かし、したがってオフロードカード１０２のＳｏＣ１０４およびＦＰＧＡ１０６にオフロードすることを可能にする。例えば、汎用プロセッサの柔軟性から利益を得る（またはハードウェアで実装するには単に複雑／強力すぎる）ハイパーバイザ機能をＳｏＣ１０４上で動かすことができ、ＳｏＣ１０４には１つまたは複数の汎用処理コアが組み込まれている。このような機能の例は、ＳＤＮ（ソフトウェアデファインドネットワーキング（ｓｏｆｔｗａｒｅ－ｄｅｆｉｎｅｄ　ｎｅｔｗｏｒｋｉｎｇ））制御プレーン機能を含み、ＳＤＮは複雑なルーティング計算を必要とし、新しいプロトコルおよび特徴をサポートするために比較的頻繁に更新される必要がある。

【００１６】

[0020]その一方で、ハードウェアアクセラレーションに適したハイパーバイザ機能は、ＦＰＧＡ１０６上の論理ブロックを介して実装することができる。このような機能の例は、ＳＤＮデータプレーン機能および（データ複製、重複排除などの）ストレージデータプレーン機能を含み、ＳＤＮデータプレーン機能は、制御プレーンの決定に従ってネットワークデータトラフィックを転送することを伴う。

【００１７】

[0021]このソリューションにより、従来のサーバ設計を超えるいくつかの長所が達成される。第１に、一定のホスト処理の役目をＣＰＵコンプレックス１２４から取り除くことによって、ハイパーバイザによって使用される（ＣＰＵコンプレックス１２４内のＣＰＵコアを含む）プラットフォームリソースの量を減少させることができ、（「ゲスト」としても知られる）ＶＭが利用可能なプラットフォーム容量を増加させる。これは、サーバ能力のあらゆる漸進的向上が、規模の点で重大なインパクトを与える恐れがあるパブリッククラウドプラットフォームにおいて特に有益である。いくつかの実施形態では、ハイパーバイザは、ＣＰＵコンプレックス１２４から全面的に立ち退かされ、オフロードカード１０２に移されてもよく、このケースでは、ＣＰＵコンプレックス１２４は最低限のハイパ

ーバイザを動かすことができるか（ＣＰＵコンプレックス１２４は一定のレジスタにアクセスすることなど、ＣＰＵコンプレックス自体でしか動かせない問題を処理する）、またはハイパーバイザが全くなく、ＣＰＵコンプレックス１２４の計算能力の残りをゲストワークロードに充てることができる。

【００１８】

[0022]第２に、ＳｏＣ１０４（非ハードウェアアクセラレーション機能を扱う）と、ＦＰＧＡ１０６（ハードウェアアクセラレーション機能を扱う）との両方をオフロードカード１０２上に実装し、これら２つをしっかりと連結することによって、ＳｏＣ１０４上で動くハイパーバイザコードがＦＰＧＡ１０６に実装されたロジックと相互作用しやすくなり、逆もまた同様である。オフロードカード１０２上にハードウェアアクセラレータを単に含む代替実装形態を有することができるが、これらの実装形態は、ハードウェアアクセラレータの活動をサーバのメインＣＰＵと正しく協調させるためのデータフローを必要とする。さらにこれらの代替実装形態は、「ベアメタル」プラットフォームをサポートしなくてよく、機能の大部分としてオフロードしなくてよい。

【００１９】

[0023]第３に、オフロードカード１０２上で動くホストコードは、ＣＰＵコンプレックス１２４上で動くゲストコードから物理的に切り離されるので、悪意のエンティティがＶＭを介してハイパーバイザを攻撃するのが困難になる。これは、現代のＣＰＵアーキテクチャにおける一定のサイドチャネル脆弱性の最近の発見の観点から特に適切である。これらの既知の脆弱性にはパッチを当てることができるが、他の同様の脆弱性が将来見つかる可能性がある。

【００２０】

[0024]第４に、ハードウェアアクセラレーションのためにＡＳＩＣではなくＦＰＧＡを使用することによって、ＦＰＧＡをプログラムし直すことによって改善される種々のユースケースまたは同じユースケースのためにオフロードカード１０２を簡単に再利用することができ、必要ならＦＰＧＡのロジックを更新することができる。これは、既に現場にある多くのカードを引き抜き、置き替えることが望ましくないことがある大規模な導入に有利である。

【００２１】

[0025]図１の物理サーバ１００について示された特定のトポロジは例証であり、様々な変更形態が可能であることを理解されたい。例えば、ＳｏＣ１０４およびＦＰＧＡ１０６は、周辺機器（例えばＰＣＩｅ）インターフェースを介して物理サーバのメインボードとインターフェースする拡張カード（すなわちオフロードカード１０２）上に実装されるものとして示されているが、いくつかの実施形態では、代替のオフロードアーキテクチャが使用されてもよい。特定の実施形態では、ＳｏＣ１０４および／またはＦＰＧＡ１０６の１つまたは複数は、サーバのメインボードに直接実装されてもよい。

【００２２】

[0026]別の例として、ＮＩＣ１２８がスタンドアロンの構成要素として描写されているが、いくつかの実施形態では、ＮＩＣ１２８の機能は、ＦＰＧＡ１０６内など、図１に示された１つまたは複数の他の構成要素に組み込まれてもよい。当業者は、他の変形形態、変更形態、および代替形態を認識するはずである。

３．オフロードカードアーキテクチャ

[0027]図２は、いくつかの実施形態による図１のオフロードカード１０２のアーキテクチャに関するさらなる詳細を提示する概略図２００である。次に本アーキテクチャの様々な態様が下記で論じられる。

３．１　ＳｏＣ

[0028]ＳｏＣ１０４は、１つまたは複数の汎用処理コア、（メモリ、ストレージ、および周辺機器のための）インターフェース、ならびにＮＩＣを含む、いくつかの既存のシステムオンチップ設計のいずれか１つを使用して実装することができる。特定の実施形態では、ＳｏＣ１０４は、ＡＲＭマイクロプロセッサアーキテクチャに基づく汎用処理コアを

組み込むことができる。}

[0029]図示のように、ＳｏＣ１０４は、３つの別個のインターフェースを介してＦＰＧＡ１０６と通信可能に連結され、このインターフェースは、下記のセクション３．２で論じられる。さらにＳｏＣは、（１）メモリインターフェース２０４を介して図１のＲＡＭ１１０に対応する１つまたは複数のＤＲＡＭ（ダイナミックＲＡＭ）モジュール２０２に、（２）ストレージインターフェース２０８を介して図１のフラッシュメモリ１１２に対応するｅＭＭＣ（組込型マルチメディアカード：ｅｍｂｅｄｄｅｄ　ｍｕｌｔｉｍｅｄｉａ　ｃａｒｄ）デバイス２０６に、（３）ＳＰＩ（シリアルペリフェラルインターフェース：Ｓｅｒｉａｌ　Ｐｅｒｉｐｈｅｒａｌ　Ｉｎｔｅｒｆａｃｅ）インターフェース２１２および介在するセキュリティチップ２１４を介してＢＩＯＳフラッシュメモリ構成要素２１０に、ならびに、（４）（ＦＰＧＡ１０６およびＰＣＩｅエッジコネクタインターフェース１０８にも接続する）Ｉ２Ｃバス２２２を介して（ＥＥＰＲＯＭ２１６、ホットスワップコントローラ２１８、および温度センサ２２０などの）いくつかのＩ２Ｃ（集積デバイス間）デバイスに、接続される。

【００２３】

[0030]（１）に関して、ＳｏＣ１０４は、物理サーバ１００のＣＰＵコンプレックス１２４からオフロードされたハイパーバイザコードを含むプログラムコードを動かすためのＳｏＣ１０４の作業メモリとして、ＤＲＡＭモジュール２０２を使用することができる。ＤＲＡＭモジュール２０２の固有の数および容量、ならびにメモリインターフェース２０４の仕様は、実装形態に応じて変化させることができる。特定の実施形態では、ＤＲＡＭモジュール２０２は、シングル１０２４Ｍ（メガビット）ｘ６４ビット＋ＥＣＣ（エラー訂正コード）メモリバンクとして編成された８ＧＢ（ギガバイト）のＤＤＲ４　ＤＲＡＭを備えることができ、メモリインターフェース２０４は、シングルＤＤＲ４－２４００メモリチャネルとして構成することができる。

【００２４】

[0031]（２）に関して、ＳｏＣ１０４は、ＣＰＵコンプレックス１２４からオフロードされたハイパーバイザコードを含む、ＳｏＣ上で実行されることになるプログラムコードを格納およびブートするため、ならびに、ＦＰＧＡ１０６に適用されることになるＦＰＧＡ構成イメージを格納するための非一時的ストレージ媒体として、ｅＭＭＣデバイス２０４を使用することができる。

【００２５】

[0032]（３）に関して、ＢＩＯＳフラッシュメモリ構成要素２１０は、ＳｏＣ１０４のためのシステムファームウェアを保持することができ、セキュリティチップ２１４は、数ある中でも、このシステムファームウェアが、攻撃者によって意図的または偶然に修正されることも破損させられることもないことを保証することができる。

【００２６】

[0033]（４）に関して、Ｉ２Ｃデバイス２１６、２１８、および２２０は、オフロードカード１０２に関する様々な管理情報をＢＭＣ１１８に提供することができる。これらの情報は、動作温度データ、製造情報、および電力消費量データなどの情報を含むことができる。

【００２７】

[0034]上記に加えて、ＳｏＣ１０４は、外部ヘッダ２２８、２３０、および２３２へのＵＳＢ（ユニバーサルシリアルバス）、ＣＯＭ、およびＪＴＡＧ（ジョイントテストアクショングループ）インターフェース２２４、２２５、および２２６をそれぞれ含み、これらは、デバッグまたは管理のためにＳｏＣ１０４をＢＭＣ１１８または外部デバイスと接続するために使用することができる。ＢＭＣ１１８によってＰＣＩｅエッジコネクタインターフェース１０８を通じてＳｏＣ１０４に送ることができるパワースロットル信号２３４もある。

３．２　ＳｏＣとＦＰＧＡとの間のインターフェース

[0035]既に前で述べたように、ＳｏＣ１０４は、図２の３つの内部のチップ間（ｃｈｉ

ｐ－ｔｏ－ｃｈｉｐ）インターフェース（ＰＣＩｅインターフェース２３６、イーサネットインターフェース２３８、およびＪＴＡＧインターフェース２４０）を介してＦＰＧＡ１０６と通信可能に連結される。様々な実施形態では、ＰＣＩｅインターフェース２３６は、制御能力とデータ転送／交換能力の両方を提供する。制御能力については、ＳｏＣ１０４は、ＰＣＩｅインターフェース２３６（または代替としてＪＴＡＧインターフェース）を使用して、ＦＰＧＡ１０６を管理および更新することができる。例えば、ＳｏＣ１０４は、ＲＡＭ１１０からＦＰＧＡ１０６に転送されたＦＰＧＡ構成イメージが正しいことを確認することができ、このインターフェースを使用してＦＰＧＡ上またはＦＰＧＡのフラッシュメモリ１２２内のイメージを更新することができる。データ能力については、ＰＣＩｅインターフェース２３６は、ＳｏＣ１０４上で動くプログラムコードがＦＰＧＡ１０４にデータを送ること、およびＦＰＧＡ１０４からデータを受け取ることを可能にすることができる。これは、例えばＰＣＩｅを介してデータを交換するように既に書かれているハイパーバイザコードに有用であり、なぜなら、ＳｏＣ１０４上での実行（またはＦＰＧＡ１０６での実装）のために、このようなコードを比較的少ない変更で移植することができるからである。特定の実施形態では、ＰＣＩｅインターフェース２３６は、８つのＰＣＩ３．０レーンを有することができる（すなわちＰＣＩ３．０　８ｘインターフェースに対応させることができる）。他の実施形態では、４、１２、１６など、他の任意の数のＰＣＩレーンがサポートされ得る。

【００２８】

[0036]イーサネットインターフェース２３８は、ネットワークパケットの形のデータをＳｏＣ１０４およびＦＰＧＡ１０６が交換することを可能にする。これは、例えばネットワークパケットを介してデータを交換するように既に書かれているハイパーバイザコードに有用であり、なぜなら、ＳｏＣ１０４上での実行（またはＦＰＧＡ１０６上での実装）のために、このようなコードを比較的少ない変更で移植することができるからである。例えば、ネットワークフローベースの転送がＦＰＧＡ１０６上のハードウェアで実装され、ネットワークフローのためのルートを決定するためのネットワーク制御プレーンがＳｏＣ１０４上のソフトウェアで実装されるシナリオを考察する。このケースでは、フローテーブルの例外およびルールをネットワークパケットの形で、ＦＰＧＡ１０６とＳｏＣ１０４との間で通信することができる。特定の実施形態では、イーサネットインターフェース２３８は、２５Ｇ（ギガビット）イーサネットをサポートすることができる。

【００２９】

[0037]ＪＴＡＧインターフェース２４０は、ＳｏＣ１０４が、低レベルテスト（例えばデバッグ）およびプログラミングのためにＦＰＧＡ１０６と通信するための通路を提供する。いくつかの実施形態では、外部ヘッダ２３２を介して接続された外部プログラマデバイスがインターフェース２４０を駆動させることを可能にするＪＴＡＧ多重化装置を、ＳｏＣ１０４とＦＰＧＡ１０６との間のＪＴＡＧ経路に挿入することができる。これらの実施形態では、外部プログラマデバイスからの「現在」の信号が、ＪＴＡＧインターフェース２４０の信号経路をＳｏＣ１０４からデバイスにスイッチする。これは、最初のビットストリームをロードするときの最初のオフロードカードの立ち上げ（ｂｒｉｎｇ－ｕｐ）に有用であり、ＳｏＣからＦＰＧＡへのＪＴＡＧ経路が用意されていないときのＦＰＧＡアプリケーション開発に有用である。図３は、いくつかの実施形態によるＪＴＡＧ多重化装置３０２を含む本アーキテクチャの実例の図３００を描写する。

３．３　ＦＰＧＡ

[0038]ＦＰＧＡ１０６は、いくつかの既存のＦＰＧＡチップのいずれか１つを使用して実装することができる。特定の実施形態では、ＦＰＧＡ１０６は、一定の最低限の数のプログラム可能論理素子（例えば１０００Ｋ個の素子）および一定の最低限のトランシーバ／ＦＰＧＡファブリックスピードグレード（例えばグレード２）をサポートする既存のＦＰＧＡチップを使用して実装することができる。図２に示されているように、ＦＰＧＡ１０６は、Ｉ２Ｃバス２２２と通信可能に連結され、上で論じたインターフェース２３６～２４０を介してＳｏＣ１０４と通信可能に連結される。さらにＦＰＧＡ１０６は、（１）

内部ＰＣＩｅインターフェース２４２を介してＰＣＩｅエッジコネクタインターフェース１０８に、（２）メモリインターフェース２４６を介して図１のＲＡＭ１２０に対応する１つまたは複数のＤＲＡＭモジュール２４４に、（３）ストレージインターフェース２４９を介して図１のフラッシュメモリ１２２に対応するＱＳＰＩ（クアッドシリアルペリフェラルインターフェース（Ｑｕａｄ　Ｓｅｒｉａｌ　Ｐｅｒｉｐｈｅｒａｌ　Ｉｎｔｅｒｆａｃｅ））フラッシュメモリモジュール２４８に、ならびに（４）イーサネットインターフェース２５４および２５６を介して２つのネットワークトランシーバモジュール２５０および２５２に、それぞれ接続される。

【００３０】

[0039]（１）に関して、内部ＰＣＩｅインターフェース２４２は、ＣＰＵコンプレックス１２４、および（例えばＮＩＣ１２８を含む）物理サーバ１００にインストールされた他のＰＣＩｅデバイスと、ＦＰＧＡ１０６が通信することを可能にする。特定の実施形態では、ＰＣＩｅインターフェース２４２は、ＰＣＩｅ３．０　ｘｌ６インターフェースでよい。

【００３１】

[0040]（２）に関して、ＦＰＧＡ１０６は、ＣＰＵコンプレックス１２４からオフロードされたハイパーバイザロジックを含む、デバイスにプログラムされたロジックを実行するときの、ＦＰＧＡ１０６の作業メモリとしてＤＲＡＭモジュール２４４を使用することができる。ＤＲＡＭモジュール２４４の固有の数および容量、ならびにメモリインターフェース２４６の仕様は、実装形態に応じて変化させることができる。特定の実施形態では、ＤＲＡＭモジュール２０２は、２つの４ＧＢバンクの５１２Ｍｘ６４ビット＋ＥＣＣとして編成された８ＧＢ（ギガバイト）のＤＤＲ４　ＤＲＡＭを備えることができ、メモリインターフェース２４６は、デュアルＤＤＲ４－２４００メモリチャネルとして構成することができる。

【００３２】

[0041]（３）に関して、ＱＳＰＩフラッシュメモリモジュール２４８は、ＦＰＧＡ１０６の指定の機能を実施するようにＦＰＧＡ１０６自体を構成するために、電源投入時にＦＰＧＡ１０６がロードできる１つまたは複数のＦＰＧＡ構成イメージを保持することができる。いくつかの実施形態では、ＱＳＰＩフラッシュメモリモジュール２４８は、下記のセクション３．４に記述された少なくとも３つの別個のイメージを保持することができる。フラッシュメモリからの構成の他に、ＦＰＧＡ１０６は、外部ＪＴＡＧプログラマデバイスを介した構成、ＪＴＡＧインターフェース２４０を介してＳｏＣ１０４によって送られたＪＴＡＧコマンド、ＰＣＩｅを介したＣｖＰ（プロトコルを介した構成（Ｃｏｎｆｉｇｕｒａｔｉｏｎ　ｖｉａ　Ｐｒｏｔｏｃｏｌ））、およびＰＣＩｅを介した部分再構成もサポートすることができる。

【００３３】

[0042]（４）に関して、ネットワークトランシーバモジュール２５０は、ＦＰＧＡ１０６が、外部ネットワーク１２６から入って来るネットワークトラフィックを受け取ること、および、外部ネットワーク１２６に出て行くネットワークトラフィックを伝送することを可能にする。さらに、ネットワークトランシーバモジュール２５２は、ＦＰＧＡ１０６がネットワークトラフィックをＮＩＣ１２８と交換することを可能にする。これは、ＦＰＧＡ１０６がモジュール２５２を介してＮＩＣ１２８から出て行くネットワークパケットを受け取り、ネットワークパケットを適切に処理／変換し、モジュール２５０を介して外部ネットワーク１２６にネットワークパケットを送出することができるので、ＦＰＧＡ１０６がネットワークプレーン機能を実装するシナリオに有用である。逆に、ＦＰＧＡ１０６は、モジュール２５０を介して外部ネットワーク１２６から入って来るネットワークパケットを受け取り、ネットワークパケットを適切に処理／変換し、モジュール２５２を介してＮＩＣ１２８にネットワークパケットを送ることができる（この時点でネットワークパケットを正しい宛先ＶＭに通信することができる）。このようにネットワークデータプレーンアクセラレーションのためにＦＰＧＡ１０６を活用する実例のネットワークデータ

フローが、下記のセクション４で論じられる。特定の実施形態では、ネットワークトランシーバモジュール２５０および２５２はＱＳＦＰ２８光モジュールであることが可能であり、イーサネットインターフェース２５４および２５６は１００Ｇイーサネットをサポートすることができる。

３．４　ＦＰＧＡフラッシュ構成の詳細

[0043]実施形態の１つのセットでは、ＱＳＰＩフラッシュメモリモジュール２４８は、ＦＰＧＡ１０６のための最低限の３つの別個の構成イメージ（ゴールデンイメージ、フェールセーフイメージ、およびユーザアプリケーションイメージ）を格納することができる。ゴールデンイメージは、最初の製造時に工場試験され、ＦＰＧＡ１０６のための正常な意図する機能を備える。フェールセーフイメージは工場でプログラムされ、製造後は決して上書きされない。様々な実施形態では、このフェールセーフイメージは、電源投入時にオフロードカード１０２によって要求される機能の最小限のセットを収め、ＦＰＧＡ１０６のネットワークインターフェースは、ＦＰＧＡによるいかなる中間処理もないインターフェース間を全てのトラフィックが直接通過する迂回モードを強制される。最後に、ユーザアプリケーションイメージは、ユーザ／顧客によって定義されたイメージである。

【００３４】

[0044]オフロードカード１０２が電源投入された時、デフォルトでゴールデンイメージは、ＱＳＰＩフラッシュメモリモジュール２４８からロードされ、ＦＰＧＡ１０６の構造を構成するためにＦＰＧＡ１０６に適用されることになる。この電源投入処理に伴う何らかのエラーがある場合（またはサーバ実行時間中に問題が見つかった場合）、ゴールデンイメージの代わりにフェールセーフイメージをロードするためにカードをリブートすることができる。

４　実例のネットワーク処理ワークフロー

[0045]前述のオフロードカードアーキテクチャを考慮に入れて、図４は、いくつかの実施形態による物理サーバ１００によって実装され得る実例のネットワーク処理ワークフローのフローチャート４００を描写する。フローチャート４００は、オフロードカード１０６のＦＰＧＡ１０６が、ＳｏＣ１０４上で動くネットワーク制御プレーンによって決定されたネットワークフローを備えるフローテーブルを維持すること、およびフローテーブルに従ってデータパケットを転送することを行うように構成されると仮定する。

【００３５】

[0046]ブロック４０２でスタートし、物理サーバ１００のＮＩＣ１２８は、ＳＲ－ＩＯＶ（シングルルートＩＯ仮想化）インターフェースを、サーバ１００上で動くＶＭに提示することができる。このＳＲ－ＩＯＶインターフェース（仮想機能と呼ばれる）は、ハイパーバイザを伴わずにＶＭがＮＩＣ１２８と直接通信することを可能にする。

【００３６】

[0047]ブロック４０４において、ＶＭは、リモートの宛先に伝送されることになるネットワークパケットのためのデータペイロードを作り出すことができ、これをＮＩＣ１２８に通知することができる。これに応答して、ＮＩＣ１２８は、ＶＭのゲストメモリ空間からデータペイロードを読み込むこと（ブロック４０６）、数ある中でも、ＶＭのＩＰアドレス、および意図される宛先のＩＰアドレスを識別するヘッダを伴う１つまたは複数のネットワークパケットにデータペイロードを組み立てること（ブロック４０８）、ならびに、ＦＰＧＡ１０６のネットワークトランシーバモジュール２５２に接続されたＮＩＣ１２８の出口ポートからネットワークパケットを出力すること（ブロック４１０）を行うことができる。

【００３７】

[0048]ブロック４１２および４１４において、ＦＰＧＡ１０６はネットワークパケットを受け取り、フローテーブルへのネットワークパケットの５－タプル（ソースＩＰアドレス、ソースポート、宛先ＩＰアドレス、宛先ポート、プロトコル）のルックアップを実施するために、ＦＰＧＡ１０６のネットワークデータプレーンロジックを適用することができる。マッチするエントリがテーブル内で見つかった場合（ブロック４１６）、ＦＰＧＡ

１０６は、エントリ内でネットワークパケットのためのネクストホップの宛先を識別すること（ブロック４１８）、パケットのヘッダを更新すること（ブロック４２０）、ネットワークトランシーバモジュール２５０から外部ネットワーク１２６にパケットを送ること（ブロック４２２）を行うことができ、これによりワークフローを終える。

【００３８】

[0049]その一方で、ブロック４１６において、マッチするエントリがテーブル内で見つからなかった場合（これがフローにおける第１のパケットであることを示す）、ＦＰＧＡ１０６は、内部イーサネットインターフェース２３８を介してネットワークパケットをＳｏＣ１０４に送ることができる（ブロック４２４）。ＳｏＣ１０４上で動くネットワーク制御プレーン構成要素は、次に、パケットのためのネクストホップの宛先を計算し、インターフェース２３８を介してパケットのネットワークフローのための新しいエントリをＦＰＧＡのフローテーブルに追加することができる（ブロック４２６）。この新しいエントリを使用して、ＦＰＧＡ１０６はブロック４２０および４２２を実行することができ、ワークフローを終えることができる。

【００３９】

[0050]上記の説明は、本開示の様々な実施形態を、これらの実施形態の態様がどのように実装され得るかについての例と共に示す。上記の例および実施形態は唯一の実施形態であると見なされるべきではなく、以下の特許請求の範囲によって定義されるような本開示の柔軟性および長所を示すために提示される。例えば、いくつかの実施形態が特定の処理フローおよびステップについて説明されてきたが、本開示の範囲が、説明されたフローおよびステップに厳密に限定されるわけではないことが当業者には明らかなはずである。連続的なものと説明されたステップは同時に実行されてもよく、ステップの順序は変化してもよく、ステップは修正、結合、追加、または省略されてもよい。別の例として、いくつかの実施形態がハードウェアとソフトウェアの特定の組合せを使用して説明されてきたが、ハードウェアとソフトウェアの他の組合せが可能であること、および、ソフトウェアで実装されるものとして説明された特定の動作をハードウェアでも実装することができ、逆もまた同様であることを認識するはずである。

【００４０】

[0051]したがって、本明細書および図面は制限的な意味ではなく、例証的なものとみなされるべきである。他の配置、実施形態、実装形態、および同等物が当業者には明らかなはずであり、以下の特許請求の範囲において示されるような本開示の趣旨および範囲から逸脱することなく採用されてもよい。

【図面】

【図１】

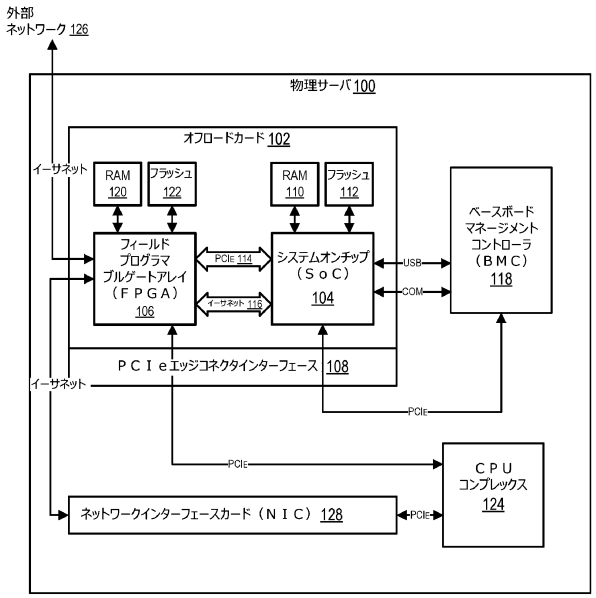

FIG. 1

【図２】

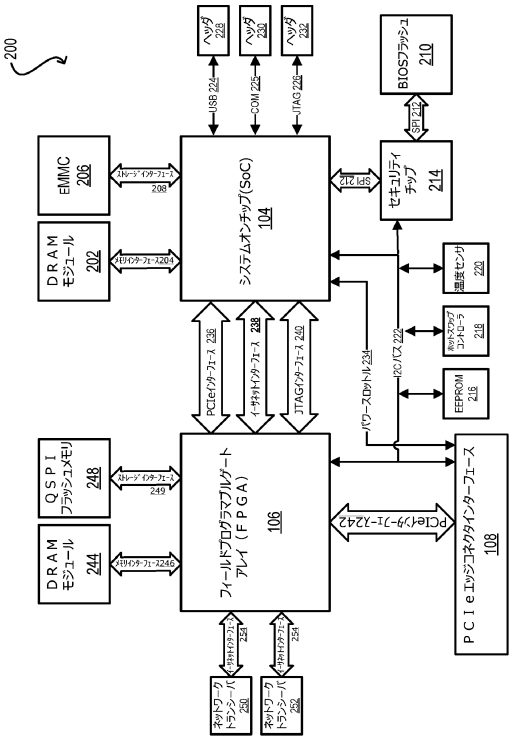

FIG. 2

【図３】

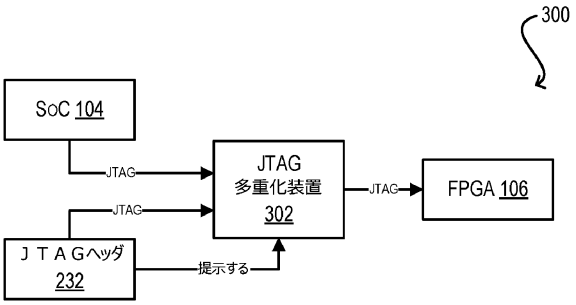

FIG. 3

【図４】

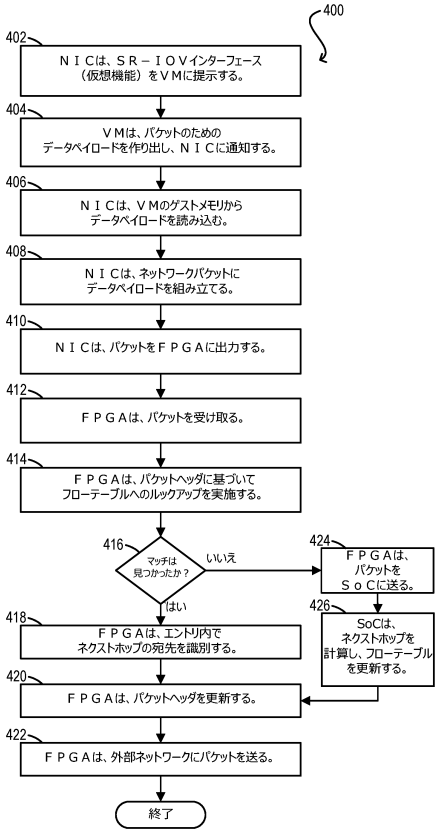

FIG. 4

フロントページの続き

(33)優先権主張国・地域又は機関
米国(US)
クロソフト ウェイ，マイクロソフト テクノロジー ライセンシング，エルエルシー
(72)発明者 パトナム，アンドリュー
アメリカ合衆国 ワシントン州 ９８０５２－６３９９ レッドモンド ワン マイクロソフト ウェイ，マイクロソフト テクノロジー ライセンシング，エルエルシー
(72)発明者 ファイアストン，ダニエル
アメリカ合衆国 ワシントン州 ９８０５２－６３９９ レッドモンド ワン マイクロソフト ウェイ，マイクロソフト テクノロジー ライセンシング，エルエルシー
(72)発明者 ラビアー，ジャック
アメリカ合衆国 ワシントン州 ９８０５２－６３９９ レッドモンド ワン マイクロソフト ウェイ，マイクロソフト テクノロジー ライセンシング，エルエルシー

審査官 坂東 博司

(56)参考文献
国際公開第２０１９／０８３９７７（WO，Ａ１）
特表２０２１－５０１４０７（ＪＰ，Ａ）
国際公開第２０１８／０６４４１５（WO，Ａ１）
特表２０１９－５３５０９２（ＪＰ，Ａ）
特開２０１２－１５６７９６（ＪＰ，Ａ）
特表２０１５－５１５７９８（ＪＰ，Ａ）
特開２０１８－１４９７５５（ＪＰ，Ａ）
特開２０１７－１７４３０１（ＪＰ，Ａ）
米国特許出願公開第２０１８／０００４５３９（ＵＳ，Ａ１）
DANIEL FIRESTONE et al，Azure Accelerated Networking:SmartNICs in Publoc Cloud，Proceedings of the 15th USENIX Symposium on Networked Systems Design and Implementation，USENIX Association，2018年04月09日，51-64，［online］，発行日2018年4月9日，［令和5年10月17日検索］，インターネット ＜URL：https://www.usenix.org/sites/default/filies/nsdi18_full_proceedings_interior.pdf＞

(58)調査した分野 (Int.Cl.，ＤＢ名)
Ｇ０６Ｆ ９／５０
Ｇ０６Ｆ ９／４５５