

(19)日本国特許庁(JP)

(12)特許公報(B2)

(11)特許番号
特許第7682148号
(P7682148)

(45)発行日 令和7年5月23日(2025.5.23)

(24)登録日 令和7年5月15日(2025.5.15)

(51)国際特許分類	F I			
G 0 6 F 12/109(2016.01)	G 0 6 F 12/109	1 0 0		
G 0 6 F 12/0842(2016.01)	G 0 6 F 12/0842			
G 0 6 F 15/177(2006.01)	G 0 6 F 15/177		Z	

請求項の数 17 (全12頁)

(21)出願番号	特願2022-503804(P2022-503804)	(73)特許権者	508301087
(86)(22)出願日	令和2年7月22日(2020.7.22)		エーティーアイ・テクノロジーズ・ユー
(65)公表番号	特表2022-541059(P2022-541059		エルシー
	A)		ATI TECHNOLOGIES ULC
(43)公表日	令和4年9月21日(2022.9.21)		カナダ、オンタリオ エル3ティー 7
(86)国際出願番号	PCT/IB2020/056906		エックス6、マーカム、コマース パリー
(87)国際公開番号	WO2021/014382		ドライブ イースト 1
(87)国際公開日	令和3年1月28日(2021.1.28)		One Commerce Valley
審査請求日	令和5年7月7日(2023.7.7)		Drive East, Markham
(31)優先権主張番号	16/519,311		, Ontario, L3T 7X6 C
(32)優先日	令和1年7月23日(2019.7.23)		anada
(33)優先権主張国・地域又は機関	米国(US)	(74)代理人	100108833
			弁理士 早川 裕司
		(74)代理人	100111615
			弁理士 佐野 良太

最終頁に続く

(54)【発明の名称】 ヘテロジニアスコンピューティングのための統合カーネル仮想アドレス空間

(57)【特許請求の範囲】

【請求項 1】

第 1 のサブシステムが、第 1 のカーネルアドレス空間の第 1 の領域を第 1 の物理アドレス空間の第 1 の領域にマッピングすることと、
前記第 1 のサブシステムとは異なる第 2 のサブシステムが、第 2 のカーネルアドレス空間の第 1 の領域を前記第 1 の物理アドレス空間の第 1 の領域及びデバイスメモリ空間にマッピングすることと、

前記第 1 のカーネルアドレス空間及び前記第 2 のカーネルアドレス空間の両方において共通の共有領域を使用可能にすることと、を含む、
方法。

【請求項 2】

前記共通の共有領域は、第 1 のサブシステム及び前記第 1 のサブシステムとは異なる第 2 のサブシステムの両方による共有が可能である、
請求項 1 の方法。

【請求項 3】

前記第 1 のカーネルアドレス空間の第 1 の領域内のアドレスは、前記第 1 の物理アドレス空間において、前記第 2 のカーネルアドレス空間の第 1 の領域内のアドレスと同じ位置にマッピングされる、
請求項 1 の方法。

【請求項 4】

前記第 1 のカーネルアドレス空間の第 1 の領域を前記第 1 の物理アドレス空間の第 1 の領域にマッピングすることは、第 1 のサブシステムによって実行される、
請求項 1 の方法。

【請求項 5】

前記第 2 のカーネルアドレス空間の第 1 の領域を前記第 1 の物理アドレス空間の第 1 の領域にマッピングすることは、前記第 1 のサブシステムとは異なる第 2 のサブシステムによって実行される、
請求項 4 の方法。

【請求項 6】

前記第 2 のサブシステムは、入力 / 出力メモリ管理ユニットである、
請求項 5 の方法。

10

【請求項 7】

システムであって、
第 1 のカーネルアドレス空間の第 1 の領域を第 1 の物理アドレス空間の第 1 の領域にマッピングするように構成された第 1 のサブシステムと、
第 2 のカーネルアドレス空間の第 1 の領域を前記第 1 の物理アドレス空間の第 1 の領域及びデバイスメモリ空間にマッピングするように構成された、前記第 1 のサブシステムとは異なる第 2 のサブシステムと、を備え、

前記システムは、前記第 1 のカーネルアドレス空間及び前記第 2 のカーネルアドレス空間の両方において共通の共有領域を使用可能にするように構成されている、
システム。

20

【請求項 8】

前記第 1 のカーネルアドレス空間の第 1 の領域内のアドレスは、前記第 1 の物理アドレス空間において、前記第 2 のカーネルアドレス空間の第 1 の領域内のアドレスと同じ位置にマッピングされる、
請求項 7 のシステム。

【請求項 9】

前記第 2 のサブシステムは、入力 / 出力メモリ管理ユニットである、
請求項 7 のシステム。

【請求項 10】

前記第 1 のサブシステムは、前記第 1 の領域を前記第 1 の物理アドレス空間に割り当てるように構成されている、
請求項 7 のシステム。

30

【請求項 11】

前記第 1 のサブシステムは、前記第 1 の物理アドレス空間の第 1 の領域内の位置に対するポインタを、前記共通の共有領域を介して前記第 2 のサブシステムに伝達するように構成されている、
請求項 8 のシステム。

【請求項 12】

前記第 2 のサブシステムは、前記第 2 のカーネルアドレス空間の第 1 の領域を前記第 1 の物理アドレス空間の第 1 の領域にマッピングするように構成された入力 / 出力メモリ管理ユニットを備える、
請求項 11 のシステム。

40

【請求項 13】

システムであって、
メモリデバイスと、
第 1 の仮想アドレス空間の第 1 の領域を前記メモリデバイス内の領域にマッピングするように構成された第 1 の処理デバイスと、
第 2 の仮想アドレス空間の第 1 の領域を前記メモリデバイス内の領域にマッピングして、前記メモリデバイス内の領域を前記第 1 の処理デバイスと第 2 の処理デバイスによって

50

共有させるように構成された第 2 の処理デバイスと、を備え、

前記第 1 の仮想アドレス空間の第 1 の領域内のアドレスは、前記メモリデバイスにおいて、
前記第 2 の仮想アドレス空間の第 1 の領域内のアドレスと同じ位置にマッピングされる、
システム。

【請求項 1 4】

前記第 2 の処理デバイスは、前記第 2 の仮想アドレス空間の第 1 の領域を前記メモリデバイス内の領域にマッピングするために、前記第 2 の仮想アドレス空間の第 1 の領域を、前記メモリデバイス内の領域にマッピングされるデバイスメモリ空間にマッピングするように構成されている、

請求項 1 3 のシステム。

10

【請求項 1 5】

前記第 2 の処理デバイスは、前記デバイスメモリ空間を前記メモリデバイス内の領域にマッピングするように構成された入力 / 出力メモリ管理ユニットを備える、

請求項 1 4 のシステム。

【請求項 1 6】

前記第 1 の処理デバイスは、前記領域を前記メモリデバイスに割り当てるように構成されている、

請求項 1 3 のシステム。

【請求項 1 7】

前記第 1 の処理デバイスは、前記メモリデバイスの領域内の位置に対するポインタを、
前記メモリデバイス内の前記位置を介して前記第 2 の処理デバイスに伝達するように構成されている、

20

請求項 1 3 のシステム。

【発明の詳細な説明】

【背景技術】

【0001】

(関連技術の説明)

一般に、コンピューティングデバイス間でメモリバッファを共有するには、デバイスがデバイスアドレス空間又は物理アドレス空間においてポインタ又はハンドルを交換する必要がある。

30

【0002】

殆どの場合、このポインタを使用する前に、計算デバイスは、メモリ管理ユニット (MMU) ページテーブルマッピングでポインタをローカル仮想アドレス空間に変換する。これにより、ポインタ自体又はポインタを含む構造の共有が非効率的で困難になる。

【0003】

添付図面と併せて以下の説明を参照することによって、本明細書に記載される方法及びメカニズムの利点をより良く理解することができる。

【図面の簡単な説明】

【0004】

【図 1】コンピューティングシステムの一実施形態のブロック図である。

40

【図 2】ヘテロジニアスコンピューティングのための統合カーネル仮想アドレス空間を生成する一実施形態の図である。

【図 3】2 つの別個のサブシステム間でバッファを共有する一実施形態の図である。

【図 4】メモリの領域を複数のカーネルアドレス空間にマッピングする一実施形態の図である。

【図 5】統合カーネル仮想アドレス空間を生成する方法の一実施形態を示す一般化されたフロー図である。

【図 6】複数のカーネルアドレス空間において共通の共有領域を有効にする方法の一実施形態を示す一般化されたフロー図である。

【図 7】2 つの異なるサブシステム間でバッファポインタ共有を可能にする方法の一実施

50

形態を示す一般化されたフロー図である。

【発明を実施するための形態】

【0005】

以下の説明では、本明細書で提示される方法及びメカニズムの十分な理解をもたらすために、多数の特定の詳細が示されている。しかし、当業者は、様々な実施形態がこれらの特定の詳細無しに実施し得ることを認識するはずである。いくつかの例では、本明細書で説明するアプローチを曖昧にしないように、周知の構造、構成要素、信号、コンピュータプログラム命令、及び、技術が詳細に示されていない。説明を簡単且つ明確にするために、図面に示される要素が必ずしも縮尺通りに描かれていないことを理解されたい。例えば、いくつかの要素の寸法は、他の要素に対して誇張されている場合がある。

10

【0006】

ヘテロジニアスコンピューティングのための統合カーネル仮想アドレス空間を実装するためのシステム、装置及び方法が本明細書に開示されている。一実施形態では、システムは、第1のカーネルを実行する第1のサブシステムと、入力/出力メモリ管理ユニット(IOMMU)と、第2のカーネルを実行する第2のサブシステムと、を少なくとも含む。一実施形態では、IOMMUは、第1のサブシステム及び第2のサブシステムがカーネルレベルでメモリバッファを共有することを可能にする統合されたカーネルアドレス空間を生成する。2つのサブシステム間でメモリバッファを共有するために、第1のサブシステムは、第1のサブシステムによって制御されるシステムメモリの一部にメモリブロックを割り当てる。第1のマッピングは、第1のサブシステムの第1のカーネルアドレス空間の第1の論理アドレスからメモリブロックに生成される。次に、IOMMUは、第2のサブシステムの第2のカーネルアドレス空間の第2の論理アドレスからそのメモリブロックの物理アドレスをマッピングするための第2のマッピングを生成する。これらのマッピングにより、第1のサブシステム及び第2のサブシステムは、メモリブロックを参照するカーネルアドレス空間内のバッファポインタを共有することができる。

20

【0007】

図1を参照すると、コンピューティングシステム100の一実施形態のブロック図が示されている。一実施形態では、コンピューティングシステム100は、第1のサブシステム110と、第2のサブシステム115と、入力/出力(I/O)インターフェース(複数可)120と、入力/出力メモリ管理ユニット(IOMMU)125と、メモリサブシステム130と、周辺機器(複数可)135と、を少なくとも含む。他の実施形態では、コンピューティングシステム100は、他の構成要素を含むことができ、及び/又は、コンピューティングシステム100は、異なるように構成することができる。

30

【0008】

一実施形態では、第1及び第2のサブシステム110, 115は、異なるカーネルアドレス空間を有するが、統合カーネルアドレス空間は、第1及び第2のサブシステム110, 115用にIOMMU125によって生成される。統合カーネルアドレス空間は、第1及び第2のサブシステム110, 115が互いにポインタを受け渡し、バッファを共有することを可能にする。第1のサブシステム110及び第2のサブシステム115の場合、各々のカーネルアドレス空間は、カーネル論理アドレス及びカーネル仮想アドレスを含む。一部のアーキテクチャでは、カーネルの論理アドレスとそれに関連する物理アドレスが、一定のオフセットで異なる。カーネル仮想アドレスは、カーネル論理アドレスを特徴付ける物理アドレスへの線形的な1対1のマッピングを必ずしも有しているわけではない。全てのカーネル論理アドレスはカーネル仮想アドレスであるが、カーネル仮想アドレスは必ずしもカーネル論理アドレスである必要はない。

40

【0009】

一実施形態では、第1のサブシステム110及び第2のサブシステム115の各々は、オペレーティングシステムを実行する1つ以上のプロセッサを含む。また、プロセッサ(複数可)は、様々な実施形態において、1つ以上のソフトウェアプログラムを実行する。第1のサブシステム110及び第2のサブシステム115のプロセッサ(複数可)は、任

50

意の数及びタイプの処理ユニット（例えば、中央処理ユニット（CPU）、グラフィック処理ユニット（GPU）、デジタル信号プロセッサ（DSP）、フィールドプログラマブルゲートアレイ（FPGA）、特定用途向け集積回路（ASIC））を含む。また、第1のサブシステム110は、メモリ管理ユニット（MMU）112を含み、第2のサブシステム115は、MMU117を含み、各MMUは、対応するサブシステム用の仮想アドレスから物理アドレスへの変換を処理する。第1及び第2のサブシステム110、115は、異なるカーネルアドレス空間を有するが、統合カーネルアドレス空間は、第1及び第2のサブシステム110、115用にIOMMU125によって生成され、第1及び第2のサブシステム110、115がポインタを受け渡し、メモリサブシステム130内でバッファを共有することを可能にする。

10

【0010】

メモリサブシステム130は、任意の数及びタイプのメモリデバイスを含む。例えば、メモリサブシステム130のメモリのタイプは、高帯域メモリ（HBM）、不揮発性メモリ（NVM）、ダイナミックランダムアクセスメモリ（DRAM）、スタティックランダムアクセスメモリ（SRAM）、NANDフラッシュメモリ、NORフラッシュメモリ、強誘電体ランダムアクセスメモリ（FeRAM）、又は、他のメモリを含むことができる。I/Oインターフェース120は、任意の数及びタイプのI/Oインターフェース（例えば、PCI（Peripheral Component Interconnect）バス、PCI-X（PCI-Extended）、PCIE（PCI Express）バス、ギガビットイーサネット（登録商標）（GBe）バス、ユニバーサルシリアルバス（USB））を表す。様々なタイプの周辺デバイス135は、I/Oインターフェース120に結合され得る。このような周辺デバイス135は、ディスプレイ、キーボード、マウス、プリンタ、スキャナ、ジョイスティック又は他のタイプのゲームコントローラ、メディア記録デバイス、外部記憶デバイス、及び、ネットワークインターフェースカード等を含むが、これらに限定されない。

20

【0011】

一実施形態では、第1のサブシステム110と第2のサブシステム115との間で共有されるメモリ空間を生成するために、メモリブロックが、第1のサブシステム110によって管理されるシステムメモリの一部に割り当てられる。メモリの初期ブロックが割り当てられた後、適切なI/O仮想アドレス（VA）が、第2のサブシステム115に割り当てられる。一実施形態では、IOMMUマッピングは、第2のサブシステム115のカーネルアドレス空間からメモリブロックの物理アドレスに生成される。この実施形態では、IOMMU125は、第2のサブシステム115の仮想アドレスマッピングをメモリブロックに実施する。次に、追加のメモリが割り当てられると、ヒープ割り当て関数（heap allocate function）が呼び出され、アドレスが、以前に生成された同じI/O VAアドレスに基づいてマッピングされる。次に、第2のサブシステム115に統合アドレスを通知するメッセージが、第2のサブシステム115に送信される。

30

【0012】

様々な実施形態では、コンピューティングシステム100は、コンピュータ、ラップトップ、モバイルデバイス、ゲームコンソール、サーバ、ストリーミングデバイス、ウェアラブルデバイス、又は、任意の他の様々な種類のコンピューティングシステム若しくはデバイスである。コンピューティングシステム100の構成要素の数は、実施形態毎に異なることに留意されたい。他の実施形態では、図1に示す数よりも多い又は少ない構成要素が存在する。他の実施形態では、コンピューティングシステム100は、図1に示されていない他の構成要素を含むことにも留意されたい。さらに、他の実施形態では、コンピューティングシステム100は、図1に示す以外の方法で構造化される。

40

【0013】

図2を参照すると、ヘテロジニアスコンピューティングのための統合カーネル仮想アドレス空間を生成する一実施形態の図が示されている。図2の上部からの垂直の破線は、左から右に、第1のサブシステム202、第1のサブシステム202のMMU204、第1のサブシステム202上で実行されるデバイスドライバ206、メモリの共有領域208

50

、I O M M U 2 1 1、第2のサブシステム2 1 4のM M U 2 1 2、及び、第2のサブシステム2 1 4を表す。説明のために、第1のサブシステム2 0 2は第1のオペレーティングシステムを有し、第2のサブシステム2 1 4は第2のオペレーティングシステムを有し、第2のオペレーティングシステムは、第1のオペレーティングシステムと異なることが仮定される。また、説明のために、第1のサブシステム2 0 2及び第2のサブシステム2 1 4は、ヘテロジニアスコンピューティングシステムの一部であると仮定される。一実施形態では、このヘテロジニアスコンピューティングシステムの場合、第1のサブシステム2 0 2及び第2のサブシステム2 1 4の各々は、ワークロードの一部を実行し、ワークロードを実行するために、第1のサブシステム2 0 2及び第2のサブシステム2 1 4は、バッファポインタ及びバッファを互いに共有する。図2の図は、第1のサブシステム2 0 2と第2のサブシステム2 1 4との間で共有されるメモリを割り当てる一例を示している。

10

【0 0 1 4】

一実施形態では、第1のステップ2 0 5は、ヒープ(すなわち、共有メモリ領域)を生成するためにデバイスドライバ2 0 6によって実施される。本明細書で使用する場合、「ヒープ」という用語は、物理メモリプールにマッピングされる仮想メモリプールとして定義される。次に、ステップ2 1 0では、ヒープの所望のサイズが物理メモリサブシステムに割り当てられる。ステップ2 1 5では、カーネルからヒープへのマッピングが生成される。一実施形態では、`carveout`ヒープが最初に生成されると、新しいフラグが、ヒープがカーネル論理アドレス空間から取得されるかどうかを示す。たとえば、`kmallloc`関数は、カーネル論理アドレス空間のメモリを返す。一実施形態では、Linux(登録商標)オペレーティングシステムの場合、メモリマネージャーは、Linux(登録商標) `genpool`ライブラリを使用して、ヒープに割り当てられたバッファを管理する。この実施形態では、バッファが割り当てられると、そのバッファは内部プールにマークされ、物理アドレスが返される。次に、`carveout`ヒープは、この物理アドレスをバッファ及び`sg__table`記述子でラップ(wraps)する。一実施形態では、バッファがカーネルアドレス空間においてマッピングされると、`heap_map_kernel`関数は、ステップ2 2 0で`vmap`の代わりに`kmap`を使用してバッファをマッピングする。関数`kmap`は、論理マッピングに基づいて、バッファを所定の仮想アドレスにマッピングする。`kmap`又は`vmap`の選択は、`carveout`ヒープの生成中に与えられた新しいフラグによって制御される。或いは、`genpool`ライブラリ、`carveout`アプリケーションプログラミングインターフェース(API)、及び、新しいフラグを使用して、カーネルマッピングと共にヒープが生成される。この場合、カーネルマップは、事前にマッピングされたアドレスを返す。ユーザモードマッピングは、依然としてオンザフライで適用される。第2のサブシステム2 1 4の場合、第1のサブシステム2 0 2によって割り当てられた共有メモリバッファは、`carveout`ヒープ`wrapped genpool`と同様に管理される。新しいAPIにより、`carveout`ヒープに外部バッファを追加することができる。第2のサブシステム2 1 4上のローカルタスクは、同じAPIを使用してこの`carveout`ヒープから割り当てる。ステップ2 2 0の後に、共有メモリ領域の割り当てが完了する。

20

30

【0 0 1 5】

次に、ステップ2 2 5では、入力/出力(I/O)仮想アドレスが、I O M M U 2 1 1によって共有メモリ領域に割り当てられる。次に、ステップ2 3 0では、DMAアドレス空間の連続するブロックが、共有メモリ領域に対して予約される。次に、ステップ2 3 5では、共有メモリ領域は、I O M M U 2 1 1によって、第2のサブシステム2 1 4のカーネルアドレス空間にマッピングされる。デバイスドライバ2 0 6がシャットダウンされるまで、I O M M Uマッピングが解放されるべきではないことに留意されたい。第1のサブシステム2 0 2のカーネルアドレス空間から共有メモリ領域へのマッピングは、ステップ2 4 0で無効にされる。次に、ステップ2 4 5では、メモリ解放関数を実行することによって、カーネルアドレス空間マッピングが解放される。

40

【0 0 1 6】

50

図 3 を参照すると、2 つの別個のサブシステム間でバッファを共有する一実施形態の図が示されている。図 3 の上部から下に延びる垂直の破線は、左から右に、第 1 のサブシステム 3 0 2、第 1 のサブシステム 3 0 2 の MMU 3 0 4、第 1 のサブシステム 3 0 2 上で実行されるデバイスドライバ 3 0 6、メモリの共有領域 3 0 8、I O MMU 3 1 0、第 2 のサブシステム 3 1 4 の MMU 3 1 2、及び、第 2 のサブシステム 3 1 4 を表す。説明のために、第 1 のサブシステム 3 0 2 は第 1 のオペレーティングシステムを有し、第 2 のサブシステム 3 1 4 は第 2 のオペレーティングシステムを有し、第 2 のオペレーティングシステムは、第 1 のオペレーティングシステムと異なると仮定される。

【 0 0 1 7 】

ループ交換 3 0 5 の場合、メモリブロックが割り当てられ、第 1 のサブシステム 3 0 2 のカーネルアドレス空間からメモリブロックの物理アドレスへのマッピングがデバイスドライバ 3 0 6 によって生成される。その後、マッピングは、MMU 3 0 4 によって維持される。メモリブロックは、第 2 のサブシステム 3 1 4 に排他的に割り当てられてもよいし、第 1 のサブシステム 3 0 2 と第 2 のサブシステム 3 1 4 との間で共有されてもよいし、第 1 のサブシステム 3 0 2 に排他的に割り当てられてもよい。次に、メッセージが、第 1 のサブシステム 3 0 2 から第 2 のサブシステム 3 1 4 に、メモリのブロックのアドレス及びサイズと共に送信される。一実施形態では、メッセージは、帯域外で送信されてもよい。メッセージが第 2 のサブシステム 3 1 4 によって受信された後、第 2 のサブシステム 3 1 4 のカーネルアドレス空間からメモリブロックの物理アドレスへのマッピングが MMU 3 1 2 によって生成され、維持される。ループ交換 3 1 5 の場合、データは、共有領域 3 0 8 を使用して、第 1 のサブシステム 3 0 2 と第 2 のサブシステム 3 1 4 との間で交換される。カーネル仮想アドレスが統合されているので、バッファポインタ 1st__SS__buf と 2nd__SS__buf とは同じであり、第 1 のサブシステム 3 0 2 と第 2 のサブシステム 3 1 4 との間で自由に交換することができ、genpool ライブラリを使用してさらに分割することができる。

【 0 0 1 8 】

図 4 を参照すると、メモリ領域を複数のカーネルアドレス空間にマッピングする一実施形態の図が示されている。一実施形態では、ヘテロジニアスコンピューティングシステム（例えば、図 1 のシステム 1 0 0 ）は、それ自身の独立したオペレーティングシステムを有する複数の異なるサブシステムを含む。図 4 に示す垂直方向の長方形のブロックは、ヘテロジニアスコンピューティングシステムの異なる構成要素のアドレス空間を表している。図 4 に示すアドレス空間は、左から右に、第 1 のサブシステム仮想アドレス空間 4 0 2、物理メモリ空間 4 0 4、デバイスメモリ空間 4 0 6、及び、第 2 のサブシステム仮想アドレス空間 4 0 8 である。

【 0 0 1 9 】

一実施形態では、第 1 のサブシステム仮想アドレス空間 4 0 2 は、物理メモリ空間 4 0 4 のメモリブロック 4 2 5 及びメモリブロック 4 3 0 にマッピングされる共有領域 4 2 0 を含む。一実施形態では、共有領域 4 2 0 のメモリブロック 4 2 5 及びメモリブロック 4 3 0 へのマッピングは、第 1 のサブシステム MMU 4 1 2 によって生成及び維持される。共有領域 4 2 0 が第 2 のサブシステムと共有されるのを可能にするために、メモリブロック 4 2 5 及びメモリブロック 4 3 0 は、I O MMU 4 1 4 によってデバイスメモリ空間 4 0 6 の共有領域 4 3 5 にマッピングされる。次に、共有領域 4 3 5 は、第 2 のサブシステム MMU 4 1 6 によって、第 2 のサブシステム仮想アドレス空間 4 0 8 の共有領域 4 4 0 にマッピングされる。このマッピングスキームを介して、第 1 のサブシステム及び第 2 のサブシステムは、自身のカーネルアドレス空間においてバッファポインタを相互に共有することができる。

【 0 0 2 0 】

図 5 を参照すると、統合カーネル仮想アドレス空間を生成する方法 5 0 0 の一実施形態が示されている。説明のために、本実施形態でのステップ、及び、図 6 ~ 図 7 のステップを順番に示す。しかしながら、説明する方法の様々な実施形態では、説明する要素のうち

1つ以上は、同時に実行されてもよいし、図示した順序と異なる順序で実施されてもよいし、完全に省略されてもよいことに留意されたい。他の追加の要素も必要に応じて実施される。本明細書に記載される様々なシステム又は装置の何れも、方法500を実施するように構成されている。

【0021】

第1のサブシステムは、メモリサブシステムに対応する物理アドレス空間内の第1の物理アドレスにメモリブロックを割り当てる(ブロック505)。一実施形態では、第1のサブシステムは、第1のカーネルアドレス空間を有する第1のオペレーティングシステムを実行する。次に、第1のサブシステムは、第1のカーネルアドレス空間内の第1の論理アドレスの第1の物理アドレスへのマッピングを生成する(ブロック510)。第1のカーネルアドレス空間の第1の論理アドレス空間は、第1の物理アドレスからの第1の線形オフセットであることに留意されたい。次に、IOMMUは、第2のカーネルアドレス空間内の第2の論理アドレスの第1の物理アドレスへのIOMMUマッピングを生成する(ブロック515)。一実施形態では、第2のカーネルアドレス空間は、第2のサブシステムに関連付けられている。第2のカーネルアドレス空間の第2の論理アドレス空間は、第1の物理アドレスからの第2の線形オフセットであることに留意されたい。

10

【0022】

次に、第1のサブシステムは、バッファポインタを第2のサブシステムに伝達し、バッファポインタは、第1の論理アドレスを指す(ブロック520)。次に、第2のサブシステムは、バッファポインタを用いてアクセス要求を生成し、アクセス要求をIOMMUに伝達する(ブロック525)。次に、IOMMUは、以前に生成されたIOMMUマッピングを使用して、バッファポインタの仮想アドレスを第1の物理アドレスに変換する(ブロック530)。次に、第2のサブシステムは、第1の物理アドレスでメモリブロックにアクセスする(ブロック535)。ブロック535の後に、方法500は終了する。

20

【0023】

図6を参照すると、複数のカーネルアドレス空間内の共通の共有領域を有効にする方法600の一実施形態が示されている。第1のサブシステムは、メモリサブシステムに対応する物理アドレス空間内の第1の物理アドレスにメモリブロックを割り当てる(ブロック605)。説明のために、方法600によって説明するシステムは、第1のサブシステム及び第2のサブシステムを含むと仮定される。また、第1のサブシステムは、第1のカーネルアドレス空間を有する第1のオペレーティングシステムを実行し、第2のサブシステムは、第2のカーネルアドレス空間を有する第2のオペレーティングシステムを実行すると想定される。第1のサブシステムは、第1のカーネルアドレス空間内の第1の論理アドレスの第1の物理アドレスへのマッピングを生成する。ここで、第1のカーネルアドレス空間内の第1の論理アドレスは、第1の物理アドレスから第1のオフセットだけ離れている(ブロック610)。

30

【0024】

次に、IOMMUは、第2のカーネルアドレス空間の第1の論理アドレスから第2のオフセットだけ離れたデバイスアドレスを選択する(ブロック615)。次に、IOMMUは、デバイスアドレス空間で選択されたデバイスアドレスの第1の物理アドレスへのIOMMUマッピングを生成する(ブロック620)。IOMMUマッピングによって、第1のカーネルアドレス空間と第2のカーネルアドレス空間の両方の共通の共有領域を、第1のサブシステムと第2のサブシステムの両方で使用することが可能になる(ブロック625)。ブロック625の後に、方法600は終了する。

40

【0025】

図7を参照すると、2つの異なるサブシステム間でバッファポインタの共有を可能にする方法700の一実施形態が示されている。第1のサブシステムは、第1のカーネルアドレス空間の第1の領域を物理アドレス空間の第2の領域にマッピングする(ブロック705)。一実施形態では、物理アドレス空間は、第1のサブシステムによって制御されるシステムメモリに対応する。次に、IOMMUは、第2のカーネルアドレス空間の第3の領

50

域をデバイスアドレス空間内の第4の領域にマッピングする(ブロック710)。説明のために、第2のカーネルアドレス空間は、第1のサブシステムと異なる第2のサブシステムに対応すると仮定される。次に、I O M M Uは、デバイスアドレス空間内の第4の領域を物理アドレス空間内の第2の領域にマッピングして、第1及び第3の領域内のアドレスが第2の領域内の一致する(すなわち、同一の)アドレスを指すようにする(ブロック715)。第1のサブシステム及び第2のサブシステムは、物理アドレス空間内の第2の領域を参照する第1又は第2のカーネルアドレス空間内のバッファポイントを共有することができる(ブロック720)。ブロック720の後に、方法700は、終了する。

【0026】

様々な実施形態では、ソフトウェアアプリケーションのプログラム命令を使用して、本明細書に説明する方法及び/又はメカニズムを実施する。例えば、汎用プロセッサ又は特殊目的プロセッサによって実行可能なプログラム命令が考慮される。様々な実施形態では、このようなプログラム命令は、高レベルプログラミング言語によって表すことができる。他の実施形態では、プログラム命令は、高レベルプログラミング言語からバイナリ、中間、又は、他の形式にコンパイルされてもよい。或いは、ハードウェアの動作又は設計を記述したプログラム命令が書き込まれてもよい。このようなプログラム命令は、C等の高レベルプログラミング言語によって表されてもよい。或いは、Verilog等のハードウェア設計言語(HDL)が使用されてもよい。様々な実施形態では、プログラム命令は、様々な非一時的なコンピュータ可読記憶媒体の何れかに記憶される。記憶媒体は、プログラム実行のためにプログラム命令をコンピューティングシステムに提供するために、使用中にコンピューティングシステムによってアクセス可能である。一般的に、このようなコンピューティングシステムは、少なくとも1つ以上のメモリと、プログラム命令を実行するように構成された1つ以上のプロセッサと、を含む。

【0027】

上記の実施形態は、実施形態の非限定的な例に過ぎないことを強調しておきたい。上記の開示が十分に理解されると、多数の変形及び修正が当業者に明らかになるであろう。以下の特許請求の範囲は、このような変形及び修正の全てを包含するものと解釈されることが意図されている。

10

20

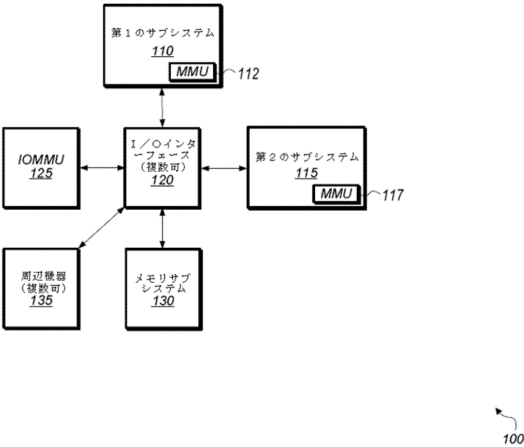
30

40

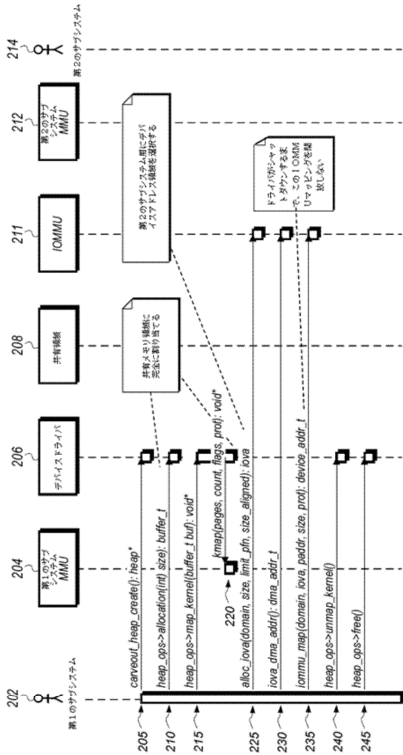
50

【図面】

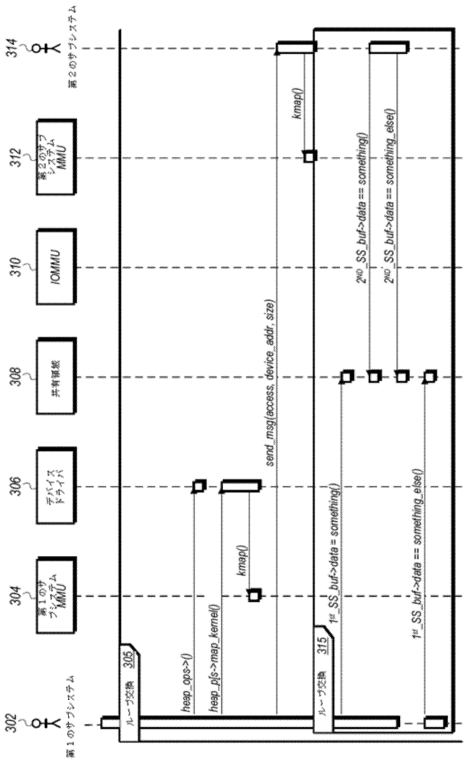
【図 1】



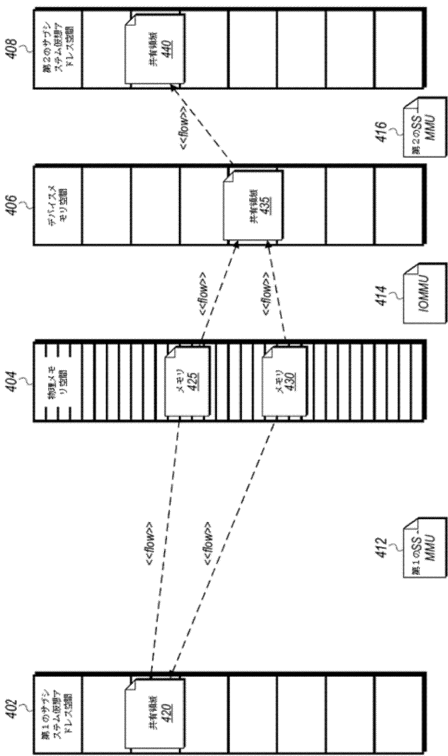
【図 2】



【図 3】



【図 4】



10

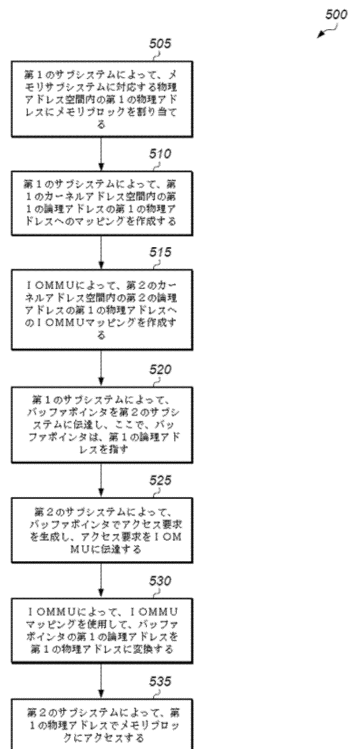
20

30

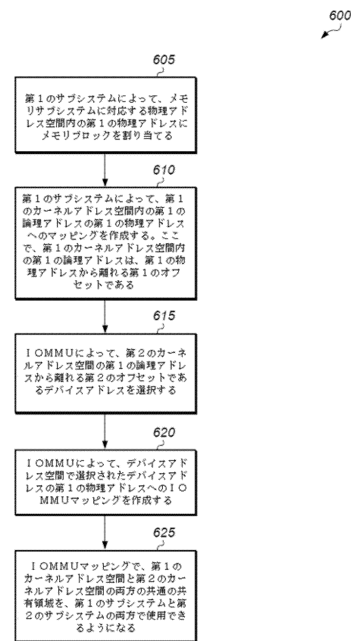
40

50

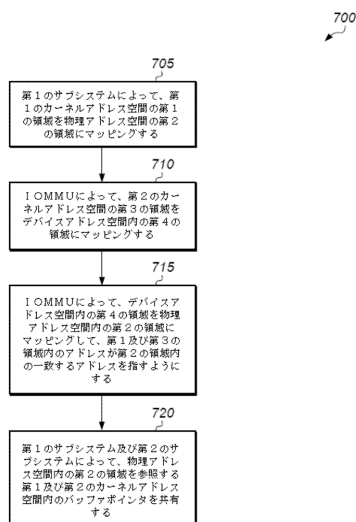
【図 5】



【図 6】



【図 7】



10

20

30

40

50

フロントページの続き

- (74)代理人 100162156
弁理士 村雨 圭介
- (72)発明者 トゥン チュアン クォン
カナダ国 L 3 T 7 X 6 オンタリオ州、マーカム、ワン コマース バリー ドライブ イースト
- (72)発明者 ベンジャミン クーン パン チャン
カナダ国 L 3 T 7 X 6 オンタリオ州、マーカム、ワン コマース バリー ドライブ イースト
- (72)発明者 ウイリアム ロイド アトキンソン
カナダ国 L 3 T 7 X 6 オンタリオ州、マーカム、ワン コマース バリー ドライブ イースト
- 審査官 太田 龍一
- (56)参考文献 米国特許出願公開第 2 0 1 9 / 0 1 1 4 1 9 3 (U S , A 1)
特表 2 0 1 5 - 5 0 0 5 2 4 (J P , A)
特表 2 0 1 4 - 5 3 1 6 7 2 (J P , A)
特表 2 0 1 6 - 5 3 2 9 5 9 (J P , A)
- (58)調査した分野 (Int.Cl. , D B 名)
G 0 6 F 1 2 / 1 0 9
G 0 6 F 1 2 / 0 8 4 2
G 0 6 F 1 5 / 1 7 7