

(19) 日本国特許庁(JP)

(12) 公開特許公報(A)

(11) 特許出願公開番号

特開2007-220086

(P2007-220086A)

(43) 公開日 平成19年8月30日(2007.8.30)

(51) Int.C1.	F 1	テーマコード (参考)
GO6F 13/10 (2006.01)	GO6F 13/10	310Z
GO6F 9/46 (2006.01)	GO6F 9/46	350
GO6F 13/14 (2006.01)	GO6F 13/14	310D

審査請求 未請求 請求項の数 23 O L (全 41 頁)

(21) 出願番号	特願2006-341835 (P2006-341835)	(71) 出願人	392026693
(22) 出願日	平成18年12月19日 (2006.12.19)		株式会社エヌ・ティ・ティ・ドコモ
(31) 優先権主張番号	特願2006-9190 (P2006-9190)		東京都千代田区永田町二丁目11番1号
(32) 優先日	平成18年1月17日 (2006.1.17)	(74) 代理人	100083806
(33) 優先権主張国	日本国 (JP)		弁理士 三好 秀和
(特許庁注: 以下のものは登録商標)		(74) 代理人	100100712
1. Linux			弁理士 岩▲崎▼ 幸邦
		(74) 代理人	100095500
			弁理士 伊藤 正和
		(74) 代理人	100101247
			弁理士 高橋 俊一
		(74) 代理人	100117064
			弁理士 伊藤 市太郎

最終頁に続く

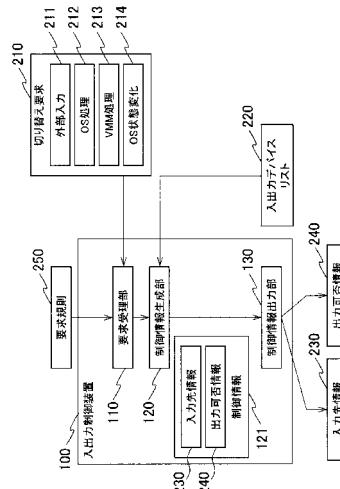
(54) 【発明の名称】入出力制御装置、入出力制御システム及び入出力制御方法

(57) 【要約】

【課題】効率性、上位プラットフォームに依存しないこと、かつ、ユーザビリティを考慮した入出力制御装置を提供する。

【解決手段】複数のOSの同時実行機能を備えるホストシステム上で、少なくとも2つ以上のOSが同時に動作する計算機に配置される入力デバイス及び出力デバイスの入出力を制御する入出力制御装置100は、入力デバイス或いは出力デバイスを占有するOSを切り替えるように要求する切り替え要求を受理する要求受理部110と、切り替え要求に応じて入力先情報230或いは出力可否情報240を含む制御情報121を生成する制御情報生成部120と、入力先情報230或いは出力可否情報240を出力する制御情報出力部130とを備える。

【選択図】図1



【特許請求の範囲】**【請求項 1】**

複数のオペレーティングシステムの同時実行機能を備えるホストシステム上で、少なくとも2つ以上のオペレーティングシステムが同時に動作する計算機に配置される入力デバイス及び出力デバイスの入出力を制御する入出力制御装置であって、

前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、

前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、

前記入力先情報或いは前記出力可否情報を出力する制御情報出力部とを備え、

前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、

前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを特徴とする入出力制御装置。

【請求項 2】

前記要求受理部は、前記計算機により予め備えられている第1の入力デバイスを制御する第1の制御ソフトウェアからの切り替え要求を受理することを特徴とする請求項1に記載の入出力制御装置。

【請求項 3】

前記要求受理部は、オペレーティングシステムからの切り替え要求を受理することを特徴とする請求項1に記載の入出力制御装置。

【請求項 4】

前記切り替え要求に対する処理を規定する要求規則が与えられた場合に、前記要求受理部は、前記要求規則に基づいて前記切り替え要求を受理或いは非受理とすることを特徴とする請求項1に記載の入出力制御装置。

【請求項 5】

前記要求規則は、前記切り替え要求の受理に秘密情報の入力を要求し、

前記要求受理部は、前記秘密情報が正しい時のみ、前記切り替え要求を受理することを特徴とする請求項4に記載の入出力制御装置。

【請求項 6】

前記要求規則は、不正な切り替え要求を定義し、

前記要求受理部は、前記切り替え要求が不正であった場合には、該切り替え要求を受理しないことを特徴とする請求項4に記載の入出力制御装置。

【請求項 7】

前記要求規則は、前記切り替え要求の受理に、ユーザからの特定の動作によるユーザの許可/不許可の判定を必要とし、

前記要求受理部は、前記特定の動作による許可を以って、前記切り替え要求を受理することを特徴とする請求項4に記載の入出力制御装置。

【請求項 8】

一定時間、前記特定の動作による許可/不許可のいずれの判定もユーザから得られなかった場合、前記要求受理部は、前記切り替え要求を受理することを特徴とする請求項7に記載の入出力制御装置。

【請求項 9】

前記要求規則は、特定の切り替え要求を特権要求とし、

前記要求受理部は、前記特権要求による制御情報の生成後、特定の条件を満たすまでは、前記特権要求により特定のオペレーティングシステムに占有された入力デバイス或いは出力デバイスに対する切り替え要求を非受理とすることを特徴とする請求項4に記載の入出力制御装置。

【請求項 10】

要求受理部は、前記ホストシステムからの切り替え要求を受理することを特徴とする請

10

20

30

40

50

求項 1 に記載の入出力制御装置。

【請求項 1 1】

前記計算機上の第 1 のオペレーティングシステム或いは前記ホストシステムが、前記計算機上で動作するオペレーティングシステムの特定の状態変化を検知する状態変化検知機能を備える場合に、前記要求受理部は、前記状態変化検知機能が、入力デバイス或いは出力デバイスを占有する第 2 のオペレーティングシステムの前記状態変化に応じて発行する切り替え要求を受理することを特徴とする請求項 3 又は 10 に記載の入出力制御装置。

【請求項 1 2】

少なくとも 1 つの入力デバイスと、少なくとも 1 つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム間における通信機能を含むホストシステムとを備える入出力制御システムであって、

前記入出力制御装置は、

前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、

前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、

前記入力先情報或いは前記出力可否情報を出力する制御情報出力部とを備え、

前記ホストシステム上で動作する前記複数のオペレーティングシステム中の特定の 1 つのオペレーティングシステムは、

前記出力された入力先情報を以って、前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定し、該入力先オペレーティングシステムへ前記入力情報を入力する入力デバイス制御部と、

前記出力された出力可否情報を以って、前記出力デバイスへ出力可能なオペレーティングシステムを特定し、該オペレーティングシステムから受け取った出力情報を前記出力デバイスへ出力する出力デバイス制御部とを備え、

前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、

前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを特徴とする入出力制御システム。

【請求項 1 3】

少なくとも 1 つの入力デバイスと、少なくとも 1 つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム間における通信機能を含むホストシステムとを備える入出力制御システムであって、

前記入出力制御装置は、

前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、

前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、

前記入力先情報及び前記出力可否情報を出力する制御情報出力部とを備え、

前記ホストシステム上で動作する前記複数のオペレーティングシステムの各々は、

前記入出力制御装置により出力された入力先情報を参照することで、前記入力デバイスからの入力情報を、前記オペレーティングシステムへ入力するか否かについて決定する入力デバイス制御部と、

前記入出力制御装置により出力された出力可否情報を以って、前記オペレーティングシステムからの出力情報を出力するか否かについて決定する出力デバイス制御部とを備えることを特徴とする入出力制御システム。

【請求項 1 4】

少なくとも 1 つの入力デバイスと、少なくとも 1 つの出力デバイスと、前記入力デバイ

10

20

30

40

50

ス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム間における通信機能を含むホストシステムとを備える入出力制御システムであって、

前記入出力制御装置は、

前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、

前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、

前記入力先情報及び前記出力可否情報を出力する制御情報出力部とを備え、

前記複数のオペレーティングシステムの各々は、

前記入力デバイスを制御可能な入力デバイス制御部と、

前記出力デバイスを制御可能な出力デバイス制御部とを備え、

前記ホストシステムは、

前記入出力制御装置により出力された入力先情報を以って、前記オペレーティングシステムの前記入力デバイス制御部を有効化或いは無効化し、

前記入出力制御装置により出力された出力可否情報を以って、前記オペレーティングシステムの前記出力制御部を有効化或いは無効化する入出力制御部を備えることを特徴とする入出力制御システム。

【請求項 1 5】

少なくとも 1 つの入力デバイスと、少なくとも 1 つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム間における通信機能を含むホストシステムとを備える入出力制御システムであって、

前記入出力制御装置は、

前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、

前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、

前記入力先情報及び前記出力可否情報を出力する制御情報出力部とを備え、

前記ホストシステムは、

前記入力デバイスに対する割り込みが発生した場合に、前記割り込みを前記オペレーティングシステムへ通知する割り込み通知部を備え、

前記割り込み通知部は、前記入出力制御装置が出力した入力先情報を以って、前記割り込みの通知先を特定することを特徴とする入出力制御システム。

【請求項 1 6】

少なくとも 1 つの入力デバイスと、少なくとも 1 つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム間における通信機能を含むホストシステムとを備える入出力制御システムであって、

前記入出力制御装置は、

前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、

前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、

前記入力先情報及び前記出力可否情報を出力する制御情報出力部とを備え、

前記ホストシステムは、

前記入出力制御装置により出力された入力先情報或いは出力可否情報に応じて、前記オペレーティングシステムに対する QoS 制御を行う QoS 制御部を備えることを特徴とする入出力制御システム。

【請求項 1 7】

10

20

30

40

50

複数のオペレーティングシステムの同時実行機能を備えるホストシステム上で、少なくとも2つ以上のオペレーティングシステムが同時に動作する計算機に配置される入力デバイス及び出力デバイスの入出力を制御する入出力制御方法であって、

前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理するステップと、

前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成するステップと、

前記入力先情報及び前記出力可否情報を出力するステップとを備え、

前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、

前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを特徴とする入出力制御方法。

【請求項18】

前記特定のオペレーティングシステムとは別の第1のオペレーティングシステムは、仮想入力デバイス制御部と、仮想出力デバイス制御部とを備え、

前記仮想入力デバイス制御部は、前記特定のオペレーティングシステムの前記入力デバイス制御部より入力された入力情報を受け取り、前記第1のオペレーティングシステム上で動作するプログラムへ入力し、

前記仮想出力デバイス制御部は、前記第1のオペレーティングシステム上で動作するプログラムからの出力情報について、前記出力可否情報を以って、前記第1のオペレーティングシステムが、前記出力デバイスに対して前記出力情報を出力可能か判断し、前記出力デバイスに対して前記出力情報を出力可能である場合に限り、前記出力情報を前記特定のオペレーティングシステムの前記出力デバイス制御部に対して出力することを特徴とする請求項12記載の入出力制御システム。

【請求項19】

少なくとも1つの入力デバイスと、少なくとも1つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能を含むホストシステムとを備える入出力制御システムであって、

前記入出力制御装置は、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、

前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、

前記入力先情報或いは前記出力可否情報を出力する制御情報出力部とを備え、

前記オペレーティングシステム上で動作するプログラムは、

前記入力先情報により、前記プログラムが動作するオペレーティングシステムが、前記入力デバイスからの入力先オペレーティングシステムである場合に限り、前記オペレーティングシステムに対して入力を要求するシステムコールを発行し、

前記出力可否情報により、前記プログラムが動作するオペレーティングシステムが、前記出力デバイスに対して出力可能である場合に限り、前記オペレーティングシステムに対して出力を要求するシステムコールを発行し、

前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、

前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを特徴とする入出力制御システム。

【請求項20】

前記オペレーティングシステムは、前記要求規則により、前記切り替え要求が許可される場合に限り、前記切り替え要求を発行することを特徴とする請求項3又は4に記載の入出力制御装置。

【請求項21】

前記ホストシステムは、前記要求規則により、前記切り替え要求が許可される場合に限

10

20

30

40

50

り、前記切り替え要求を発行することを特徴とする請求項4又は10に記載の入出力制御装置。

【請求項22】

少なくとも1つの入力デバイスと、少なくとも1つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能を含むホストシステムとを備える入出力制御システムであって、

前記入出力制御装置は、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、

前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、

前記入力先情報或いは前記出力可否情報を出力する制御情報出力部とを備え、

前記オペレーティングシステム上で動作するプログラムは、前記オペレーティングシステムに対して、前記入出力制御装置に対する切り替え要求の発行を要求し、

前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、

前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを特徴とする入出力制御システム。

【請求項23】

前記プログラムは、前記要求規則により、前記切り替え要求が許可される場合に限り、前記オペレーティングシステムに対して、前記入出力制御装置に対する切り替え要求の発行を要求することを特徴とする請求項22に記載の入出力制御システム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、複数のオペレーティングシステム（以下、OS）による入出力デバイスの共有利用を制御する入出力制御装置、入出力制御システム及び入出力制御方法に関する。

【背景技術】

【0002】

単一のOSが動作する計算機において、当該OSや当該OS上で動作するプログラムは、当該OSに適時組み込まれる入出力デバイスの制御ソフトウェア（以下、デバイスドライバ）によって、当該計算機が備える入出力デバイスにアクセスすることができる。

【0003】

ここで、当該計算機は、少なくとも、当該計算機の外部から当該OSへ情報を入力する際に利用する入力デバイス（キーボード等）と、当該OSが当該計算機外部へ情報を出力する際に利用する出力デバイス（ディスプレイ等）とを備える。

【0004】

デバイスドライバは、OSに組み込まれる際に、自らが制御するデバイスの情報を、当該OSに対して登録する。このようなデバイスの情報には、例えば、当該デバイス用の割り込み番号が挙げられる。

【0005】

計算機の外部からの入力デバイスを介した情報の入力が発生した場合、第1に、割り込みコントローラは、CPUに対して、割り込み信号と割り込み番号とを通知する。

【0006】

第2に、CPUは、かかる割り込み信号を受けて、現在行っている処理を中断し、当該割り込み番号に対応するデバイスドライバを呼び出す。

【0007】

第3に、デバイスドライバは、対応するデバイスへアクセスして、入力された情報を、OSへ渡す。

【0008】

また、OSは、特定情報を出力する際、かかる特定情報の内容に従い、適切な出力デバ

10

20

30

40

50

イスを特定する。その後、OSは、予め登録されている当該出力デバイスに対応するデバイスドライバを呼び出して当該特定情報を出力させる。

【0009】

上述のように、計算機上で単独のOSが動作する場合には、当該OSが、入出力デバイスの使用について管理して、当該OS上で動作するプログラムが同時に入出力デバイスへアクセスする競合状態が発生しないようにすれば良い。

【0010】

一方で、近年、計算機上で複数OSが同時に実行可能な環境が広がりつつある。このような環境は、図12に示すように、「バーチャルマシンモニタ（以下、VMM）」と呼ばれるホストシステムが、「バーチャルマシン（以下、VM）と呼ばれるゲストシステムを構成し、VM上で、各OSを動作することにより実現される（例えば、特許文献1や非特許文献1や非特許文献2参照）。

【0011】

例えば、かかる環境の利用例としては、リアルタイム処理に特化した「リアルタイムOS（以下、RTOS）」及び汎用的に利用する「汎用OS（以下、GPOS）」を1つの計算機上で同時に動作させるといった例や、1つのサーバ上で複数OSを動作させて、各OSで当該OS用に特化されたサービスをクライアントに提供するといった例が考えられる。

【0012】

このような環境では、入出力デバイスを始めとする、限られたハードウェア資源を、複数OSで共有しなければならない。

【0013】

VMMの役割の1つは、静的或いは動的に、ハードウェア資源を各OSに割り当てる事である。例えば、VMMは、各OSに対して動的にCPUの制御権を渡し、静的に主記憶を論理分割して各OSに割り当てることで、複数OSの同時実行を実現する。

【0014】

また、VMMの中には、同時に実行される複数OS間でのデータ通信を実現するものもある。代表的な実現方法は、複数OSが参照可能な共有メモリを提供する方法である。

【0015】

VMMは、共有メモリの管理、或いは、共有メモリへの書き込みや読み込みのOSへの通知といった処理を行う。

【0016】

このようなOS間通信機能は、単純なデータ通信時だけではなく、ハードウェア資源を複数OS間で共有する際にも利用される。

【0017】

例えば、ネットワークインターフェースを1つしか備えない計算機において、複数OSが、外部との通信を行う場合を考える。

【0018】

この場合、各OSが、独自にネットワークインターフェースのデバイスドライバを持つ構成は考えにくい。なぜなら、データ受信時に発生するハードウェア割り込みに対して、どのOSのドライバが呼び出されるべきかについて事前に判断するのが不可能なためである。

【0019】

したがって、あるOSが、唯一のデバイスドライバの保持者となり、データ送受信及び各OSへのデータ割り振りを行う必要がある。

【0020】

このようなハードウェア資源の共有利用時には、上述のOS間通信機能は、必須となる。

【特許文献1】U.S.2004/0205755

【非特許文献1】「Xen and the Art of Virtualization」 In Proc. of Symposium on op

10

20

30

40

50

erating systems Principles (SOSP) 2003 (<http://www.CI.cam.ac.uk/Research/SRG/networks/papers/2003-xensosp.pdf>)

【非特許文献2】「A 600MIPS 120mW 70μA Leakage Triple-CPU Mobile Application Processor Chip」 In Proc. of IEEE International Solid-State Circuits Conference (ISSCC) 2005

【発明の開示】

【発明が解決しようとする課題】

【0021】

VMM上で実現された複数OS環境において、任意の時点で特定OSによって排他的に占有されるデバイスについて考える。

10

【0022】

このようなデバイスの一例として、ヒューマンインタフェースデバイス（以下、HID）が挙げられる。例えば、HIDとして、入力デバイスであるキーボードや、出力デバイスであるディスプレイ等が挙げられる。

【0023】

これらのデバイスは、ユーザが利用しているOSによって排他的に占有される必要がある。

【0024】

例えば、ユーザが、特定プログラムを利用している際には、ユーザによりキーボードより入力された結果は、基本的には、当該プログラムに対して入力され、ディスプレイには、当該プログラムの出力結果が表示される必要があるため、キーボードからの入力結果は、当該プログラムが動作するOSに渡されなければならず、同様に、当該OSからの出力結果が、ディスプレイに表示されなければならない。

20

【0025】

ユーザが、あるOSを利用しているということと、CPUが、当該OSによって制御されているということは必ずしも一致しない。

【0026】

VMMは、ユーザが当該OSを利用している間であっても、OS間でコンテキストスイッチを行い、別のOSに対してCPUを割り当て、或いは、VMM自身の処理を行う可能性がある。

30

【0027】

VMMは、CPUの割り当てを管理するため、任意の時点で、どのOSがCPUを制御しているかについて把握することが可能であるが、任意の時点で、ユーザがどのOSを利用しているかについては把握することができない。

【0028】

したがって、例えば、VMMは、キーボードからデータが入力されて割り込みが発生した際に、かかる割り込みがどのOSに対する割り込みであるのかについて認識することができない。

【0029】

例えば、ネットワークインターフェースが共有されている際には、受信したデータに、IPアドレスやポート番号等の行き先のOSを識別するためのデータが付与されているため、一旦、あるOSが、当該データを受信した後で、当該データを割り振れば良い。

40

【0030】

しかし、キーボードから入力されたデータについて、同様の処理を行うことは困難である。

【0031】

同様に、ディスプレイへの出力時にも問題が発生する。

【0032】

全てのOSが、ディスプレイドライバを個別に保持し、勝手に出力すれば、当然当該ディスプレイへの競合が発生する。

50

【 0 0 3 3 】

あるOSに対して出力すべきデータを集約した場合でも、その時点で、どのOSが当該データを出力すべきかについて認識できなければ、ユーザに正しい出力結果を表示することができない。

【 0 0 3 4 】

以上のような問題に対して、特許文献1で述べられた複数OS環境では、特別の機能を用意していない。

【 0 0 3 5 】

特許文献1は、RTOSとGPOSとの同時実行環境を想定しており、常にGPOSが、HIDを占有するものとしているためである。

【 0 0 3 6 】

一方、非特許文献1或いは非特許文献2で述べられた複数OS環境では、図13に示すように、1つのホストOSを用意し、ホストOSのGUI(Graphical User Interface)サーバが、他の全てのゲストOSのHIDに対する入出力を処理することで、HIDの共有を実現している。

【 0 0 3 7 】

具体的には、ホストOSのみが、HIDに対するデバイスドライバ(HIDD)を保有し、かかるデバイスドライバは、ホストOSのGUIサーバからの入出力だけを処理可能とする。

【 0 0 3 8 】

ホストOSのGUIサーバは、ホストOS以外の各ゲストOS用にウィンドウ(Window)を生成し、対応するゲストOSのGUIサーバからVMMのOS間通信により受け取った出力結果を出力する。

【 0 0 3 9 】

また、当該ウィンドウが、アクティブな場合には、入力デバイスからの入力結果を、かかるOS間通信によって、対応するゲストOSのGUIサーバに渡す。

【 0 0 4 0 】

しかしながら、このようなGUIサーバによる解決方法には問題点がある。

【 0 0 4 1 】

第1の問題点は、全てのOSの全てのGUIクライアントアプリケーションが別々に入出力処理を行うため、OS間に跨るコンテキストスイッチが多発する点である。

【 0 0 4 2 】

複数OS環境におけるOS間に跨るコンテキストスイッチは、コンテキストの保存処理及び復元処理に時間がかかり、特に速度が重要なディスプレイへの出力処理の際には、大きなボトルネックとなってしまう。

【 0 0 4 3 】

また、携帯端末のような省電力処理が必須となる環境においては、電力消費の大きな要因ともなる。

【 0 0 4 4 】

第2の問題点は、各OSのプラットフォームやアプリケーションに大きく依存する点である。

【 0 0 4 5 】

このようなGUIサーバ間の入出力を実現するためには、ホストOSのGUIサーバ及びゲストOSのGUIサーバの双方が、当該機能に対応する必要がある。

【 0 0 4 6 】

また、特に、GUIサーバが動作する環境が著しく異なる場合には、GUIサーバ間で通信する際に、アプリケーションレベルでのデータの変換が必要となる。

【 0 0 4 7 】

第3の問題点は、GUIサーバに、全ての入出力処理を委ねることによるセキュリティ上の危険性である。

10

20

30

40

50

【 0 0 4 8 】

G U I サーバは、「非特権プロセス（ユーザ権限）」であるため、「特権プロセス（カーネル権限）」である O S と比較して、耐タンパ性が弱い。

【 0 0 4 9 】

G U I サーバが、改ざんされた場合、O S の入出力を適切に処理できなくなり、誤った入力先へ入力結果が渡されたり、表示されてはならない O S の出力結果が表示されたりしてしまう可能性がある。

【 0 0 5 0 】

さらに、H I D を占有する O S の切り替えにおいて、ユーザビリティを向上するためには、いくつもの切り替えパターンを考え、各切り替えパターンについて、切り替え方法の提供や、不正な切り替え及び H I D 占有の防止等を考慮しなければならない。

【 0 0 5 1 】

切り替えパターンとしては、例えば、以下のような例が考えられる。

【 0 0 5 2 】

- ・ 特定 O S 上で特定プログラムが起動した場合
- ・ キーボード等に用意された切り替えボタンが押された場合
- ・ H I D を占有している O S に異常が発生した場合
- ・ デバッグ等の目的で、外部から強制的に特定プログラムが起動された場合

また、O S の切り替え時に考慮しなければならないこととしては、以下のような例が考えられる。

【 0 0 5 3 】

- ・ 特定 O S や特定あるプログラムによる不正な H I D 占有の防止
- ・ ユーザビリティを損なうほどの、度重なる切り替えの防止
- ・ 優先度の高いプログラムを利用している場合の、切り替えの防止

そこで、本発明は、以上の点に鑑みてなされたもので、効率性や、上位プラットフォームに依存しないことや、ユーザビリティを考慮した入出力制御装置、入出力制御システム及び入出力制御方法を提供することを目的とする。

【 課題を解決するための手段】**【 0 0 5 4 】**

本発明の第 1 の特徴は、複数のオペレーティングシステムの同時実行機能を備えるホストシステム上で、少なくとも 2 つ以上のオペレーティングシステムが同時に動作する計算機に配置される入力デバイス及び出力デバイスの入出力を制御する入出力制御装置であって、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、前記入力先情報或いは前記出力可否情報を出力する制御情報出力部とを備え、前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを要旨とする。

【 0 0 5 5 】

本発明の第 1 の特徴において、前記要求受理部は、前記計算機により予め備えられている第 1 の入力デバイスを制御する第 1 の制御ソフトウェアからの切り替え要求を受理してもよい。

【 0 0 5 6 】

本発明の第 1 の特徴において、前記要求受理部は、オペレーティングシステムからの切り替え要求を受理してもよい。

【 0 0 5 7 】

本発明の第 1 の特徴において、前記切り替え要求に対する処理を規定する要求規則が与えられた場合に、前記要求受理部は、前記要求規則に基づいて前記切り替え要求を受理或いは非受理としてもよい。

10

20

30

40

50

【 0 0 5 8 】

本発明の第1の特徴において、前記要求規則は、前記切り替え要求の受理に秘密情報の入力を要求し、前記要求受理部は、前記秘密情報が正しい時のみ、前記切り替え要求を受理してもよい。

【 0 0 5 9 】

本発明の第1の特徴において、前記要求規則は、不正な切り替え要求を定義し、前記要求受理部は、前記切り替え要求が不正であった場合には、該切り替え要求を受理しなくてよい。

【 0 0 6 0 】

本発明の第1の特徴において、前記要求規則は、前記切り替え要求の受理に、ユーザからの特定の動作によるユーザの許可/不許可の判定を必要とし、前記要求受理部は、前記特定の動作による許可を以って、前記切り替え要求を受理してもよい。

【 0 0 6 1 】

本発明の第1の特徴において、一定時間、前記特定の動作による許可/不許可のいずれの判定もユーザから得られなかった場合、前記要求受理部は、前記切り替え要求を受理してもよい。

【 0 0 6 2 】

本発明の第1の特徴において、前記要求規則は、特定の切り替え要求を特権要求とし、前記要求受理部は、前記特権要求による制御情報の生成後、特定の条件を満たすまでは、前記特権要求により特定のオペレーティングシステムに占有された入力デバイス或いは出力デバイスに対する切り替え要求を非受理としてもよい。

[0 0 6 3]

本発明の第1の特徴において、要求受理部は、前記ホストシステムからの切り替え要求を受理してもよい。

【 0 0 6 4 】

本発明の第1の特徴において、前記オペレーティングシステムは、前記要求規則により前記切り替え要求が許可される場合に限り、前記切り替え要求を発行してもよい。

【 0 0 6 5 】

本発明の第1の特徴において、前記ホストシステムは、前記要求規則により、前記切り替え要求が許可される場合に限り、前記切り替え要求を発行してもよい。

[0 0 6 6]

本発明の第1の特徴において、前記計算機上の第1のオペレーティングシステム或いは前記ホストシステムが、前記計算機上で動作するオペレーティングシステムの特定の状態変化を検知する状態変化検知機能を備える場合に、前記要求受理部は、前記状態変化検知機能が、入力デバイス或いは出力デバイスを占有する第2のオペレーティングシステムの前記状態変化に応じて発行する切り替え要求を受理してよい。

【0067】

本発明の第2の特徴は、少なくとも1つの入力デバイスと、少なくとも1つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム間ににおける通信機能を含むホストシステムとを備える入出力制御システムであって、前記入出力制御装置は、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、前記入力先情報或いは前記出力可否情報を出力する制御情報出力部とを備え、前記ホストシステム上で動作する前記複数のオペレーティングシステム中の特定の1つのオペレーティングシステムは、前記出力された入力先情報を以って、前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定し、該入力先オペレーティングシステムへ前記入力情報を入力する入力デバイス制御部と、前記出力された出力可否情報を以って、前記出力デバイスへ出力可能なオペレーティングシステムを特定し、該オペレーティングシステムへ前記出力可否情報を出力する出力デバイス制御部とを備え、前記入力デバイスと前記出力デバイスとを接続する入出力接続部とを有する。

ティングシステムから受け取った出力情報を前記出力デバイスへ出力する出力デバイス制御部とを備え、前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを要旨とする。

【0068】

本発明の第2の特徴において、前記特定のオペレーティングシステムとは別の第1のオペレーティングシステムは、仮想入力デバイス制御部と、仮想出力デバイス制御部とを備え、前記仮想入力デバイス制御部は、前記特定のオペレーティングシステムの前記入力デバイス制御部より入力された入力情報を受け取り、前記第1のオペレーティングシステム上で動作するプログラムへ入力し、前記仮想出力デバイス制御部は、前記第1のオペレーティングシステム上で動作するプログラムからの出力情報について、前記出力可否情報を以って、前記第1のオペレーティングシステムが、前記出力デバイスに対して前記出力情報を出力可能か判断し、前記出力デバイスに対して前記出力情報を出力可能である場合に限り、前記出力情報を前記特定のオペレーティングシステムの前記出力デバイス制御部に対して出力してもよい。

【0069】

本発明の第3の特徴は、少なくとも1つの入力デバイスと、少なくとも1つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム間における通信機能を含むホストシステムとを備える入出力制御システムであって、前記入出力制御装置は、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、前記入力先情報及び前記出力可否情報を出力する制御情報出力部とを備え、前記ホストシステム上で動作する前記複数のオペレーティングシステムの各々は、前記入出力制御装置により出力された入力先情報を参照することで、前記入力デバイスからの入力情報を、前記オペレーティングシステムへ入力するか否かについて決定する入力デバイス制御部と、前記入出力制御装置により出力された出力可否情報を以って、前記オペレーティングシステムからの出力情報を出力するか否かについて決定する出力デバイス制御部とを備えることを要旨とする。

【0070】

本発明の第4の特徴は、少なくとも1つの入力デバイスと、少なくとも1つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム間における通信機能を含むホストシステムとを備える入出力制御システムであって、前記入出力制御装置は、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、前記入力先情報及び前記出力可否情報を出力する制御情報出力部とを備え、前記複数のオペレーティングシステムの各々は、前記入力デバイスを制御可能な入力デバイス制御部と、前記出力デバイスを制御可能な出力デバイス制御部とを備え、前記ホストシステムは、前記入出力制御装置により出力された入力先情報を以って、前記オペレーティングシステムの前記入力デバイス制御部を有効化或いは無効化し、前記入出力制御装置により出力された出力可否情報を以って、前記オペレーティングシステムの前記出力デバイス制御部を有効化或いは無効化する入出力制御部を備えることを要旨とする。

【0071】

本発明の第5の特徴は、少なくとも1つの入力デバイスと、少なくとも1つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム

10

20

30

40

50

ム間における通信機能を含むホストシステムとを備える入出力制御システムであって、前記入出力制御装置は、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、前記入力先情報及び前記出力可否情報を出力する制御情報出力部とを備え、前記ホストシステムは、前記入力デバイスに対する割り込みが発生した場合に、前記割り込みを前記オペレーティングシステムへ通知する割り込み通知部を備え、前記割り込み通知部は、前記入出力制御装置が出力した入力先情報を以って、前記割り込みの通知先を特定することを要旨とする。

【0072】

本発明の第6の特徴は、少なくとも1つの入力デバイスと、少なくとも1つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能及び前記複数のオペレーティングシステム間における通信機能を含むホストシステムとを備える入出力制御システムであって、前記入出力制御装置は、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、前記入力先情報及び前記出力可否情報を出力する制御情報出力部とを備え、前記ホストシステムは、前記入出力制御装置により出力された入力先情報或いは出力可否情報を応じて、前記オペレーティングシステムに対するQoS制御を行うQoS制御部を備えることを要旨とする。

【0073】

本発明の第7の特徴は、複数のオペレーティングシステムの同時実行機能を備えるホストシステム上で、少なくとも2つ以上のオペレーティングシステムが同時に動作する計算機に配置される入力デバイス及び出力デバイスの入出力を制御する入出力制御方法であって、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理するステップと、前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成するステップと、前記入力先情報及び前記出力可否情報を出力するステップとを備え、前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを要旨とする。

【0074】

本発明の第8の特徴は、少なくとも1つの入力デバイスと、少なくとも1つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能を含むホストシステムとを備える入出力制御システムであって、前記入出力制御装置は、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、前記入力先情報或いは前記出力可否情報を出力する制御情報出力部とを備え、前記オペレーティングシステム上で動作するプログラムは、前記入力先情報により、前記プログラムが動作するオペレーティングシステムが、前記入力デバイスからの入力先オペレーティングシステムである場合に限り、前記オペレーティングシステムに対して入力を要求するシステムコールを発行し、前記出力可否情報により、前記プログラムが動作するオペレーティングシステムが、前記出力デバイスに対して出力可能である場合に限り、前記オペレーティングシステムに対して出力を要求するシステムコールを発行し、前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを要旨とする。

10

20

30

40

50

【0075】

本発明の第9の特徴は、少なくとも1つの入力デバイスと、少なくとも1つの出力デバイスと、前記入力デバイス及び前記出力デバイスの入出力を制御する入出力制御装置と、複数のオペレーティングシステムの同時実行機能を含むホストシステムとを備える入出力制御システムであって、前記入出力制御装置は、前記入力デバイス或いは前記出力デバイスを占有するオペレーティングシステムを切り替えるように要求する切り替え要求を受理する要求受理部と、前記切り替え要求に応じて、入力先情報或いは出力可否情報を含む制御情報を生成する制御情報生成部と、前記入力先情報或いは前記出力可否情報を出力する制御情報出力部とを備え、前記オペレーティングシステム上で動作するプログラムは、前記オペレーティングシステムに対して、前記入出力制御装置に対する切り替え要求の発行を要求し、前記入力先情報は、少なくとも前記入力デバイスからの入力情報の入力先オペレーティングシステムを特定するための情報を含み、前記出力可否情報は、少なくともオペレーティングシステムに対する前記出力デバイスへの出力可否を特定するための情報を含むことを要旨とする。

10

【0076】

本発明の第9の特徴において、前記プログラムは、前記要求規則により、前記切り替え要求が許可される場合に限り、前記オペレーティングシステムに対して、前記入出力制御装置に対する切り替え要求の発行を要求してもよい。

20

【発明の効果】

【0077】

以上説明したように、本発明によれば、効率性や、上位プラットフォームに依存しないことや、ユーザビリティを考慮した入出力制御装置、入出力制御システム及び入出力制御方法を提供することができる。

【発明を実施するための最良の形態】

【0078】

次に、図面を参照して、本発明の実施の形態を説明する。以下の図面の記載において、同一又は類似の部分には、同一又は類似の符号を付している。ただし、図面は、模式的なものであることに留意すべきである。

30

【0079】

<第1の実施の形態>

(入出力制御システム)

図1は、第1の実施形態に係る入出力制御装置100のシステム構成である。入出力制御装置100は、例えば、複数OSが、VMM上で動作しており、当該複数OSのうちの特定OSが、任意の時点において特定デバイスを占有して利用するような場合に、当該特定デバイスを占有するOSの判定や切り替え等の処理を行う。

【0080】

このような特定デバイスとしては、例えば、キーボード等の入力デバイス或いはディスプレイ等の出力デバイスに代表されるHIDが挙げられる。

【0081】

例えば、ユーザが、別のOSで動作するプログラムを利用する際に、予め計算機に用意されている切り替えボタンを押下した場合、入出力制御装置100は、その旨を検知し、当該プログラムが動作するOSが入出力デバイスを占有するための制御情報を生成する。

40

【0082】

具体的には、入出力制御装置100は、入力デバイスからの入力情報の入力先OSを特定する入力先情報230、及び、出力デバイスへ出力情報が出力可能なOSを特定する出力可否情報240を出力する。

【0083】

入出力制御装置100は、図1に示すように、要求受理部110と、制御情報生成部120と、制御情報出力部130とを備える。以下、各部の処理について述べる。

【0084】

50

要求受理部 110 は、入出力デバイスを占有する OS を切り替えることを要求する切り替え要求 210 が、特定の要求元から発行された場合に、以下の処理を行うように構成されている。

【0085】

1. 切り替え要求 210 の検知

要求受理部 110 は、特定の要求元から発行された切り替え要求 210 を検知する。

【0086】

このような切り替え要求 210 が発行される要因として、例えば、外部入力 211 や、OS 处理 212 や、VMM 处理 213 や、OS 状態変化 214 が、挙げられる。

【0087】

例えば、ユーザが、計算機に予め備えられている切り替えボタンを押した場合に、ハードウェア割り込みとして、「外部入力 211」を要因とする切り替え要求 211 が発生する。

【0088】

ハードウェア割り込みが発生した場合には、特許文献 1 の [0138] から始まる「Interrupt and Event Handling」の章等で記述されている従来の VMM の動作に従い、当該切り替えボタンを制御するデバイスドライバによる制御が開始されるため、かかる切り替え要求 210 は、当該デバイスドライバによって発行される。

【0089】

また、例えば、その時点で入出力デバイスを占有していない OS や、当該 OS 上で動作するプログラムが、入出力デバイスの占有を要求した場合に、当該 OS が、「OS 处理 212」を要因とする切り替え要求 210 を発行する。

【0090】

VMM が、OS 間のイベント等の OS より下の処理を検知することで、必要に応じて、入出力デバイスを占有する OS を切り替えるために、「VMM 处理 213」を要因とする切り替え要求 210 は発行される。

【0091】

例えば、特許文献 1 の [0159] から始まる「Debugging」の章で記述されている VMM の OS デバッグ機能実行時に、VMM が、かかる切り替え要求 210 を発行する。

入出力デバイスを占有する OS において、特定の状態変化が起こった場合に、「OS 状態変化 214」を要因とする切り替え要求 210 は発生する。

【0092】

特に、当該 OS や別の OS や VMM 等が、状態変化検知機能を備える場合において、当該状態変化検知機能が、かかる切り替え要求 210 を発行する。

【0093】

このような状態変化の一例としては、OS の再起動や OS のフリーズが考えられる。

【0094】

また、状態変化検知機能の一例は、特許文献 1 の [0122] から始まる「Hot Reboot of Secondary Operating System」の章で記述されている。

【0095】

具体的に、各プログラムや各 OS やデバイスドライバや VMM 等によって発行された切り替え要求 210 が、どのようにして要求受理部 110 へ通知されるかについては、デバイスドライバや入出力制御装置 100 等の配置方法に依存する。

【0096】

詳細については、第 2 ~ 第 4 の実施形態に記述するが、少なくとも、切り替え要求 210 は、要求元を特定可能な情報と、要求内容（例えば、どの OS が入出力デバイスを占有するための要求であるのか、或いは、どの入出力デバイスを占有するための要求であるのか等）とを含むものとする。

【0097】

2. 切り替え要求 210 の受理 / 非受理

10

20

30

40

50

要求受理部 110 は、検知された全ての切り替え要求 210 を受理することはできない。

【0098】

例えば、悪意のあるプログラムが、連続して切り替え要求 210 を発行し、要求受理部 110 が、かかる切り替え要求の全てを受理した場合、画面の切り替えが多発したり、特定 OS に入出力デバイスが長時間占有されたりすることで、ユーザビリティが著しく低下してしまう可能性があるためである。

【0099】

要求受理部 110 は、検知した切り替え要求 210 を受理すべきか否かについて判定するための要求規則 250 が与えられた場合、当該要求規則 250 に従って、かかる切り替え要求 210 に対する受理判定を行う。

【0100】

表 1 に、かかる要求規則 250 の一例を示す。

【0101】

表 1 の例では、信頼度の低い一般 OS 及び信頼度の高いセキュア OS が動作しており、一般 OS では一般プログラムだけが動作しており、セキュア OS では、一般プログラム及び特権プログラムが動作しているものとする。

【0102】

セキュア OS と一般 OS との違いは、セキュア OS は、非常に強固なセキュリティが確保されており、一般 OS は、当該 OS や当該 OS 上で動作するプログラムに、不具合や悪意が存在する可能性があるというものである。

【表 1】

要求元	プログラム	切り替え先	受理条件
VMM		Any	無条件で受理(特権要求)
セキュアOS	特権プログラム	セキュアOS	無条件で受理(特権要求)
	一般プログラム	セキュアOS	ユーザが許可した場合、またはユーザからの応答が無かった場合
一般OS	一般プログラム	一般OS	ユーザが許可した場合、かつ不当な要求でなかった場合
切り替えボタン デバイス ドライバ		セキュアOS	ユーザのパスワードが正しかった場合
		一般OS	無条件で受理

【0103】

要求受理部 110 は、常に無条件で、VMM からの切り替え要求 210 について受理するように構成されている。

【0104】

これは、VMM からの切り替え要求 210 が、上述したようなデバッグ機能の起動等に起因して発行されるクリティカルな要求であるためである。

【0105】

また、VMM は、セキュア OS と同様に、バグや脆弱性を含んでいないものと想定されるためである。

【0106】

同様に、要求受理部 110 は、セキュア OS の特権プログラムによって発行された切り替え要求 210 についても無条件で受理するように構成されている。

【0107】

特権プログラムとしては、例えば、システムの状態変化時に起動するプログラムが想定

10

20

30

40

50

される。具体的には、特権プログラムの一例として、ハードウェアの故障をユーザに通知するプログラムや、バッテリー量の低下をユーザに通知するプログラムや、ウィルスの検知をユーザに通知するプログラム等が挙げられる。

【0108】

セキュアOS自体は、安全性が保障されており、不必要的切り替えによるユーザビリティの低下を極力避けるように設計されているため、要求受理部110は、これらの重要な情報を通知するようなプログラムからの切り替え要求210については、無条件で受理するように構成されている。

【0109】

要求受理部110は、セキュアOSの一般プログラムによって発行された切り替え要求210については、ユーザによって許可された場合に限り、受理するように構成されている。

【0110】

例えば、セキュアOS側のメーラが、メールを受信した場合に、その旨をユーザに通知するために切り替え要求210を発行する場合について考える。

【0111】

この時、ユーザは、一般OSの特定プログラムを利用しているが、OSを切り替えることにより、当該特定プログラムの利用が妨げられることを避けたいかもしれない。

【0112】

そこで、要求受理部110は、一旦、受信した切り替え要求210を保留し、一般OSに対して、ユーザに対してOSの切り替えの可否について判断させるための通知を行う。

【0113】

具体的には、要求受理部110が、ユーザへの許諾を問う当該OS上に存在するユーザ許諾プログラムを起動させ、当該ユーザ許諾プログラムが、ユーザに対する上述の判断を問う。

【0114】

このような特定OSとの通信には、特許文献1の【0149】から始まる「Inter-operating System CommunicationS-Virtual bus」に記述されているOS間通信機能等を利用する。

【0115】

もし、ユーザが、OSの切り替えを許諾した場合には、要求受理部110は、保留していた切り替え要求210を受理する。

【0116】

また、ユーザが、OSの切り替えを許諾しなかった場合には、要求受理部110は、保留していた切り替え要求210を受理しない。

【0117】

ここで、ユーザが、OSの切り替えを許諾したか否かについての反応がなかった場合を考える。

【0118】

例えば、一般OS上で動作するユーザ許諾プログラムが、不具合により正常に動作しない場合等が考えられる。

【0119】

このような場合に対応するために、要求受理部110は、一定時間、切り替え要求210が保留された場合、かかる切り替え要求210を受理するように構成されていてもよい。これにより、保留されている切り替え要求210が蓄積されたままになることを防止することができる。

【0120】

要求受理部110は、一般OSの一般プログラムにより発行された切り替え要求210については、基本的には、セキュアOSの一般プログラムと同等の判断を行う。

【0121】

10

20

30

40

50

すなわち、要求受理部 110 は、ユーザによって OS の切り替えが許諾された場合に、当該切り替え要求 210 を受理するように構成されている。

【0122】

ただし、一定時間、切り替え要求 210 が保留されている場合であっても、要求受理部 110 は、かかる切り替え要求 210 について受理しないように構成されている。

【0123】

セキュア OS 上のユーザ許諾プログラムは、不具合等によりフリーズすることはないため、ユーザが、計算機を利用している限り、基本的には、切り替え要求 210 が保留状態になることはない。

【0124】

また、一般 OS 上で動作する一般プログラムとしては、ユーザが勝手サイトからダウンロードしたようなプログラムも想定される。

【0125】

この場合、かかるプログラムが、不具合や悪意を持ち、不当な切り替え要求 210 を発行する場合も考えられる。

【0126】

例えば、かかるプログラムが、連続して切り替え要求 210 を発行したり、入出力デバイスを占有する必要がない場合であっても、切り替え要求 210 を発行したりするケースが想定される。

【0127】

ユーザは、ユーザ許諾プログラムによって、OS の切り替えを拒否することが可能であるが、毎回、OS の切り替えを許諾するか否かについて判断することは、ユーザビリティの低下を招く。

【0128】

以上のような不当な切り替え要求 210 を防止するため、ユーザ等が、予め不当な切り替え要求 210 と看做される条件について設定しておき、要求受理部 110 は、かかる条件に合致する切り替え要求 210 については自動的に非受理するように構成されていてもよい。

【0129】

要求受理部 110 は、例えば、一定時間内に一定回数以上の切り替え要求 210 を発行するようなプログラムからの切り替え要求 210 について非受理とするように構成されていてもよい。

【0130】

また、ユーザが、プログラムのインストール時に、当該プログラムにより発行される切り替え要求 210 を必ず非受理とするか否かについて設定可能とすることで、自動的に切り替え要求 210 を非受理とする条件を生成してもよい。

【0131】

この場合、プログラムのインストール中のユーザによる指定によって、一般 OS は、OS 間通信機能等を利用して、当該プログラムからの切り替え要求 210 を非受理とするように要求規則 250 を書き換えることができる。

【0132】

切り替えボタンにより発行される切り替え要求 210 は、ユーザの意思による切り替え要求 210 であるため、要求受理部 110 は、基本的には、かかる切り替え要求 210 については受理するように構成されている。

【0133】

ただし、第3者によって、セキュア OS 上のプログラムを勝手に起動されることにより、システムの根幹に係る特権プログラムを起動される可能性もある。

【0134】

したがって、セキュア OS への切り替えには、ユーザからのパスワード（秘密情報）の入力を促す。これにより、当該計算機の正規のユーザだけが、セキュア OS への切り替え

10

20

30

40

50

を行うことができる。

【0135】

なお、パスワードの入力を促すプログラムの呼び出し方法や取り扱い方法については、ユーザ許諾プログラムと同じものとする。

【0136】

また、要求規則250は、上述のように要求元によって規定される規則とは別に、要求される入出力デバイスを規定する場合もある。

【0137】

例えば、通常、標準入力であるキーボード及び標準出力であるディスプレイは、プログラムから同時に利用されるのが一般的である。

【0138】

したがって、例えば、要求規則250は、キーボードのみの占有を要求された場合には、かかる切り替え要求210について非受理としてもよい。

【0139】

これによって、特定プログラムが、キーボードを占有している状態で、ディスプレイの占有を要求し、同時に、別のプログラムが、ディスプレイを占有している状態で、キーボードの占有を要求するようなデッドロック状態を回避することが可能である。

【0140】

また、要求規則250は、VMM及びセキュアOSの特権プログラムにより発行された切り替え要求210を、特権要求として扱ってもよい。

【0141】

特権要求とは、当該特権要求に従って入出力デバイスが特定OSに占有された場合に、特定の条件を満たすまでは、当該入出力デバイスを別のOSが占有するための切り替え要求210が無条件で非受理とされるように規定されている切り替え要求である。

【0142】

例えば、セキュアOSの特権プログラムであるウィルス検知ソフトが、入出力デバイスを占有している場合、当該特権プログラムが、明示的に、当該入出力デバイスを開放するまでは、当該入出力デバイスを占有するために一般プログラムにより発行された切り替え要求210は、無条件で非受理とされる。

【0143】

なお、明示的な開放の通知は、OS間通信機能等といった切り替え要求210の通知と同様の仕組みで実現される。

【0144】

また、上述のVMMやOSやプログラムが、予め要求規則250を参照することにより、切り替え要求210を発行してもよいかどうかについて判断してもよい。この結果、無駄な切り替え要求210の発行を制限することが可能となる。

【0145】

切り替え要求210の発行は、入出力制御装置100が動作するOSやVMMへのOS間のスイッチが発生する。

【0146】

したがって、このように、切り替え要求210の発行前に、予め受理されるか否かについて判断することにより、無駄なOS間のスイッチを削減可能であり、オーバヘッドを軽減することが可能となる。

【0147】

ただし、上述の一般プログラムは、非受理になると分かっていても切り替え要求210を発行することが考えられるため、要求受理部110は、切り替え要求210の発行元に応じて、要求規則250の再チェックが必要となる場合もある。

【0148】

3. 切り替え要求210の通知

要求受理部110は、受理した切り替え要求210を、制御情報生成部120に通知す

10

20

30

40

50

るよう構成されている。

【0149】

制御情報生成部120は、要求受理部110によって受理された切り替え要求210に従って、制御情報121を生成するよう構成されている。

【0150】

ここで、制御情報121は、現在、入出力デバイスを占有するOSが、どのOSであるかについて規定する状態情報である。

【0151】

制御情報生成部120は、入力として、少なくとも入出力デバイスリスト220を有する。

10

【0152】

制御情報121は、例えば、表2に示すように、入出力デバイスリスト220により指定された全てのデバイスについて、現時点で、どのOSが占有しているかについて規定する。

【0153】

切り替え要求210は、少なくとも制御情報を更新する入出力デバイスと、当該入出力デバイスを占有するOSとを指定するため、制御情報生成部120は、当該切り替え要求210に従って、当該制御情報121を生成することが可能となる。

【0154】

ここで、マスク状態は、上述したように、特権要求の発行により、制御情報121を変更するような切り替え要求210が無条件で非受理となる状態を表している。

20

【表2】

制御情報の種類	入出力デバイス	占有OS	マスク状態
入力先情報	キーボード	セキュアOS	マスク
入力先情報	マウス	セキュアOS	マスク
出力可否情報	ディスプレイ	セキュアOS	マスク
出力可否情報	スピーカー	一般OS	

30

【0155】

制御情報出力部130は、生成された制御情報121を出力するよう構成されている。

【0156】

具体的には、制御情報出力部130は、入力先OSを特定するための入力先情報230、及び、OSに対する出力可否を決定する出力可否情報240を、適切なコンポーネントへ出力する。

【0157】

出力先は、デバイスドライバや、VMMや、ウインドウマネージャ（Window Manager）等のGUIサーバ用プログラム等が想定される。

40

【0158】

なお、制御情報出力部130の動作については、後述する。

【0159】

（入出力制御方法）

第1の実施の形態に係る入出力制御方法について、図2を用いて説明する。具体的には、複数OSの同時実行機能を備えるホストシステム上で、少なくとも2つ以上のOSが同時に動作する計算機において、当該計算機が備える入力デバイス或いは出力デバイスの入出力を制御する方法について説明する。

【0160】

50

図2に示すように、ステップS101において、入出力制御装置100は、入力デバイス或いは出力デバイスを占有するOSを切り替えるように要求する切り替え要求210を受理する。

【0161】

ここで、入出力制御装置100は、計算機により予め具備されている第1の入力デバイスを制御する第1の制御ソフトウェア(デバイスドライバ)からの切り替え要求210を受理してもよく、OSからの切り替え要求210を受理してもよく、或いは、ホストシステムからの切り替え要求210を受理してもよい。

【0162】

また、ステップS101において、切り替え要求210に対する処理を規定する要求規則250が与えられている場合には、入出力制御装置100は、かかる要求規則250に基づいて、切り替え要求210についての受理又は非受理を決定する。

【0163】

さらに、要求規則250が、切り替え要求210の受理に際して秘密情報(パスワード)の入力を要求する場合には、ステップS101において、入出力制御装置100は、秘密情報が正しい時のみ、かかる切り替え要求210を受理する。

【0164】

或いは、要求規則250が、不正な切り替え要求を定義する場合、ステップS101において、入出力制御装置100は、検知した切り替え要求210が不正であった場合には、当該切り替え要求210を受理しない。

【0165】

ここで、「不正な切り替え要求」とは、例えば、一定時間内の連続した切り替え要求210、或いは、切り替え要求210の発行が許可されないOSによって発行された切り替え要求210等が挙げられる。

【0166】

また、さらに、要求規則250が、切り替え要求210を受理する際に、ユーザからの特定の動作によるユーザの許可/不許可の判定を必要する場合、ステップS101において、入出力制御装置100は、特定の動作による許可を以って、当該切り替え要求210を受理する。

【0167】

また、一定時間、特定の動作による許可/不許可についてのいずれの判定もユーザから得られなかつた場合、ステップS101において、入出力制御装置100は、かかる切り替え要求210を受理する。

【0168】

また、要求規則250が、特定の切り替え要求210を「特権要求」とする場合、ステップS101において、入出力制御装置100は、当該特権要求による制御情報の生成後、特定の条件を満たすまでは、当該特権要求により特定OSに占有された入力デバイス或いは出力デバイスに対する切り替え要求210を非受理とする。

【0169】

ここで、「特定の条件」とは、例えば、特権要求を発行したプログラムの処理が終了した場合等が挙げられる。

【0170】

また、計算機上の第1のOS或いはホストシステムが、当該計算機上で動作するOSの特定の状態変化を検知する状態変化検知機能を備える場合に、ステップS101において、入出力制御装置100は、入力デバイス或いは出力デバイスを占有する第2のOSの状態変化に応じて当該状態変化検知機能により発行された切り替え要求210を受理してもよい。

【0171】

ここで、「状態変化」とは、上述したように、例えば、フリーズや終了等による第2のOSの正常動作が不可能な状態への変化である。

10

20

30

40

50

【0172】

ステップS102において、入出力制御装置100は、切り替え要求210に応じて、少なくとも入力デバイスからの入力先OSを特定する情報を含む入力先情報230、或いは、少なくともOSに対する出力デバイスへの出力可否を特定する情報を含む出力可否情報240を生成する。

【0173】

ステップS103において、入出力制御装置100は、入力先情報230及び出力可否情報240を適切なコンポーネントへ出力する。出力先は、デバイスドライバやVMMやウィンドウマネージャ等のGUIサーバプログラムである。

【0174】

（作用及び効果） 10

本実施形態によれば、様々な要因で発生する切り替え要求210に対して制御情報121を逐次生成していくことで、入力デバイスを占有するOSを逐次切り替えることができる。

【0175】

この結果、任意のタイミングで行われる入出力について、入力デバイスからの入力情報を適切なOSへ入力し、出力情報のうち適切なOSからの出力情報だけを、出力デバイスへ出力することが可能となる。

【0176】

また、OSの切り替え後は、占有OS以外からの入出力については処理しないため、OS間のコンテキストスイッチを減少させ、パフォーマンスを向上することが可能となる。 20

【0177】

さらに、入出力デバイスとOSと間のインターフェースの切り替えのみを行うため、上位アプリケーションに依存しないという利点もある。

【0178】

また、本実施形態によれば、入出力制御装置100の要求受理部110が、計算機により予め具備されている第1の入力デバイスを制御する第1の制御ソフトウェア（デバイスドライバ）からの切り替え要求210を受理するように構成されているため、当該計算機に備えられた入力デバイスからの入力により制御情報121を生成することが可能となる。 30

【0179】

この結果、最も単純な例では、特殊な切り替えボタンが用意された計算機において、ユーザが当該切り替えボタンを押すことにより、OSの切り替えが可能となる。ここで、キーボードの特定のボタンを、かかる切り替えボタンとして割り当てて使用してもよい。

【0180】

また、本実施形態によれば、入出力制御装置100の要求受理部110が、OSからの切り替え要求210を受理するように構成されているため、当該OSの特定の処理に応じて制御情報121を生成することが可能となる。

【0181】

この結果、例えば、当該OS上で、特定のGUIアプリケーションが起動した場合に、HIDの制御を、当該特定のGUIアプリケーションに移動することが可能となる。 40

【0182】

また、本実施形態によれば、入出力制御装置100の要求受理部110が、切り替え要求210に対する処理を規定する要求規則250が与えられた場合に、当該要求規則250に基づいて、当該切り替え要求210を受理するか否かについて判定するように構成されているため、様々な要因で発生する切り替え要求210について全て受理することによるユーザビリティの低下やセキュリティ上の危険性といった脅威を防止することができる。

【0183】

また、本実施形態によれば、入出力制御装置100において、要求規則250が、切り 50

替え要求 210 の受理に秘密情報（パスワード）の入力を要求し、要求受理部 110 が、当該秘密情報が正しい時のみ、当該切り替え要求を受理するように構成されているため、特定 OS への切り替えをパスワード等により制限することができる。

【0184】

例えば、計算機の基本設定が可能なセキュア OS が用意されているような場合、当該計算機のユーザにより予め設定されているパスワードにより、第 3 者による当該セキュア OS への切り替えを制限することが可能となる。

【0185】

また、本実施形態によれば、入出力制御装置 100 において、要求規則 250 が、不正な切り替え要求を定義し、要求受理部 110 が、切り替え要求 210 が不正であった場合には、当該切り替え要求 210 を受理しないように構成されているため、不正な切り替え要求 210 を制限することができる。

10 【0186】

特に、不正な OS の誤動作や悪意のある動作による不正な切り替え要求 210 を制限し、特定 OS によるデバイスの占有や、度重なる OS の切り替えによるユーザビリティの低下を防止することが可能となる。

【0187】

また、本実施形態によれば、入出力制御装置 100 において、要求規則 250 が、切り替え要求 210 の受理に、ユーザからの特定の動作によるユーザの許可/不許可の判定を必要とし、要求受理部 110 が、当該特定の動作による許可を以って、当該切り替え要求 210 を受理するように構成されているため、ユーザの許可/不許可の判定を以って、OS の切り替えを行うことができる。

20 【0188】

この結果、ユーザが、特定アプリケーションを利用中に、OS が突然切り替わることによるユーザビリティの低下を防止することが可能となる。

【0189】

ここで、ユーザから許可/不許可のいずれの判定も得られなかった場合、当該許可/不許可を問い合わせるプログラム部分に何らかの異常が発生している可能性がある。

【0190】

30 例えば、GUI アプリケーションが、フリーズしていたり、デバイス占有を目的としてユーザにわざと問い合わせなかつたりしている可能性がある。

【0191】

したがって、本実施形態によれば、入出力制御装置 100 において、一定時間、特定の動作による許可/不許可のいずれの判定もユーザから得られなかった場合、要求受理部 110 が、切り替え要求 210 を受理するように構成されているため、すなわち、当該許可/不許可の応答にタイムアウトを設けるように構成されているため、上述のような問題の発生を防止することが可能となる。

【0192】

また、本実施形態によれば、入出力制御装置 100 において、要求規則 250 が、特定の切り替え要求 210 を特権要求とし、要求受理部 110 が、当該特権要求による制御情報 121 の生成後、特定の条件を満たすまでは、当該特権要求により特定 OS に占有された入力デバイス或いは出力デバイスに対する切り替え要求 210 を非受理とするように構成されているため、特定の切り替え要求 210 を特権要求と看做し、当該特権要求に応じて最優先で入出力デバイスを割り当てることができる。

40 【0193】

このような特権要求は、例えば、端末に対するセキュリティチェックやデバッグチェック等の強制的に起動する場合が考えられるプログラムの起動時に発生する。なお、かかる特権要求は、その後に発生する一般の切り替え要求 210 よりも優先され、処理が終わるまで、入出力デバイスの占有が解かれることはない。

【0194】

50

また、本実施形態によれば、入出力制御装置 100 の要求受理部 110 が、ホストシステムからの切り替え要求 210 を受理するように構成されているため、VMM 等のホストシステムにより発行される切り替え要求 210 を受理することが可能となる。

【0195】

この結果、VMM により制御される各 OS より下のレイヤの処理に起因して、制御情報 121 を生成することが可能となる。

【0196】

また、本実施形態によれば、入出力制御装置 100 において、計算機上の第 1 の OS 或いはホストシステムが、当該計算機上で動作する OS の特定の状態変化を検知する状態変化検知機能を備える場合に、要求受理部 110 が、当該状態変化検知機能が、入力デバイス或いは出力デバイスを占有する第 2 の OS の状態変化に応じて発行される切り替え要求 210 を受理するように構成されているため、OS や VMM が、特定 OS の状態変化を検知した場合に、かかる状態変化に起因して制御情報 121 を生成することができる。

【0197】

このような状態変化としては、OS の終了や OS のフリーズが考えられ、状態変化検知機構自体は、特許文献 1 に詳しい。この結果、正常動作を行っていない OS に、入出力デバイスが占有されることを防止することが可能となる。

【0198】

また、本実施形態によれば、OS は、切り替え要求 210 を発行する前に、要求規則 250 を読み、当該切り替え要求 210 が入出力制御装置 100 の要求受理部 110 により受理される場合に限り、当該切り替え要求 210 を発行するように構成されているため、当該切り替え要求 210 に係わる OS 間のスイッチの処理コストや、入出力制御装置 100 が動作する OS や NMM 等のホストシステムの処理コストを削減することが可能となる。

【0199】

また、本実施形態によれば、ホストシステムが、切り替え要求 210 を発行する前に、要求規則 250 を読み、当該切り替え要求 210 が入出力制御装置 100 の要求受理部 110 により受理される場合に限り、当該切り替え要求 210 を発行するように構成されているため、当該切り替え要求 210 に係わる OS 間のスイッチの処理コストや、入出力制御装置 100 が動作する OS や VMM 等のホストシステムの処理コストを削減することが可能となる。

【0200】

また、本実施形態によれば、入出力デバイスを占有していない OS 上で動作するプログラムが、OS に対して切り替え要求 210 の発行を要求する前に、要求規則 250 を読み、当該切り替え要求 210 が入出力制御装置 100 の要求受理部 110 により受理される場合に限り、当該切り替え要求 210 を発行するように構成されているため、当該切り替え要求 210 に係わる当該 OS 上のプログラムと当該 OS と間でのコンテキストスイッチに係わる処理コストや、入出力制御装置 100 が動作する OS や VMM 等のホストシステムの処理コストを削減することが可能となる。

【0201】

<第 2 の実施の形態>
(入出力制御システム)

図 3 は、第 2 の実施形態に係る入出力制御装置 100 を含んだ複数 OS 環境のシステム構成である。

【0202】

第 2 の実施形態に係る入出力制御システムは、OS340 と、OS350 と、当該 OS340 上で動作するプログラム 360 と、当該 OS350 上で動作するプログラム 370 と、入力デバイス 310 と、出力デバイス 320 と、VMM330 とを備える。

【0203】

また、OS340 は、入出力制御装置 100 と、出力デバイス 320 を制御可能な出力

10

20

30

40

50

デバイス制御部 342 と、入力デバイス 310 を制御可能な入力デバイス制御部 341 を備える。

【0204】

また、OS350 は、仮想入力デバイス制御部 351 と、仮想出力デバイス制御部 352 を備える。

【0205】

ここで、OS340 は、例えば、第 1 の実施形態で述べたようなセキュア OS であり、OS350 は、一般 OS であるものとする。

【0206】

入力デバイス制御部 341 及び出力デバイス制御部 342 は、デバイスを制御するデバイスドライバとしての機能を持ち、入出力制御装置 100 と連携して動作するための拡張を施してあるものとする。 10

【0207】

また、仮想入力デバイス制御部 351 及び仮想出力デバイス制御部 352 は、OS350 やプログラム 370 に対して、あたかもデバイスドライバであるかの如く振舞う。

【0208】

実際には、仮想入力デバイス制御部 351 は、入力デバイス制御部 341 と OS350 とのインターフェースであり、入力デバイス 310 と OS350 とのデータのやり取りを中継する。

【0209】

また、仮想出力デバイス制御部 352 は、出力デバイス制御部 342 と OS350 とのインターフェースであり、出力デバイス 320 と OS350 とのデータのやり取りを中継する。 20

【0210】

すなわち、仮想入力デバイス制御部 351 は、入力デバイス制御部 341 から渡された入力デバイス 310 からの入力情報を OS350 へ入力するように構成されている。

【0211】

また、仮想出力デバイス制御部 352 は、OS350 からの出力情報を出力デバイス制御部 342 に渡すように構成されている。 30

【0212】

実際に、これらの OS 間を跨いだデータのやり取りは、VMM330 によって提供される OS 間通信機能を利用して行われる。

【0213】

第 2 の実施形態では、まず、入力デバイス制御部 341 及び出力デバイス制御部 342 が、入力先情報 230 及び出力可否情報 240 を利用して、どのように動作するかについて述べる。

【0214】

入出力制御装置 100 は、入力先情報 230 を入力デバイス制御部 341 へ出力し、出力可否情報 240 を出力デバイス制御部 342 へ出力する。

【0215】

入力デバイス制御部 341 は、外部から入力デバイス 310 へ入力情報が入力されることによって駆動する。 40

【0216】

入力についてのハードウェア割り込みが発生して、入力デバイス制御部 341 へ動作の制御が移るまでの過程は、VMM330 の動作に依存するが、一例は、特許文献 1 の [0138] から始まる「Interrupt and Event Handling」に詳しい。

【0217】

入力デバイス制御部 341 は、入力先情報 230 に基づき、どの OS が現在入力デバイス 310 を占有しているのか特定することができるため、当該 OS へ入力情報を入力する。 50

【0218】

例えば、OS340が入力デバイス310を占有している場合には、入力デバイス制御部341は、通常のデバイスドライバと同様の動作をし、OS350が入力デバイス310を占有している場合には、入力デバイス制御部341は、仮想入力デバイス制御部351へ入力情報を入力する。

【0219】

出力デバイス制御部342は、出力可否情報240により、現在、どのOSが出力デバイス320を占有しているのかについて特定することができる。

【0220】

OS340が出力デバイス320を占有している場合には、出力デバイス制御部342 10は、通常のデバイスドライバと同様の動作をする。

【0221】

また、OS350が出力デバイス320を占有している場合には、出力デバイス制御部342は、その旨をOS350の仮想出力デバイス制御部352に通知し、仮想出力デバイス制御部352からOS350の出力情報を受け取って出力デバイス320へ出力する。

【0222】

また、入出力制御装置100は、入力先情報230或いは出力可否情報240を、それぞれ仮想入力デバイス制御部351或いは仮想出力デバイス制御部352へ出力してもよい。

20

【0223】

かかる場合、OS350が、入力デバイス310を占有している場合に限り、仮想入力デバイス制御部351は、入力デバイス制御部340から入力情報を受け取り、OS350が、出力デバイス320を占有している場合に限り、仮想出力デバイス制御部352は、出力情報を出力デバイス制御部342へ出力する。

【0224】

この結果、入力デバイス制御部341と仮想入力デバイス制御部351との間で、或いは、出力デバイス制御部342と仮想出力デバイス制御部352との間で、無駄なOS間通信が発生することを防止することが可能となる。

30

【0225】

次に、第2の実施形態に係る入出力制御システム構成において、どのようにして入出力デバイスの占有OSを切り替えるように要求する切り替え要求210が発行されるかについて、図4に基づいて説明する。

【0226】

図4では、OS350が、仮想入出力制御装置351Aを新たに備えるものとしている。第1の実施形態で述べた通り、切り替え要求210は、大きく分けて、外部入力211を要因とする切り替え要求と、OS処理212を要因とする切り替え要求と、VMM処理213を要因とする切り替え要求と、OS状態変化214を要因とする切り替え要求とに分類される。

【0227】

例えば、ユーザが、計算機により予め備えられている切り替えボタンを押すことにより、外部入力211を要因とする切り替え要求210が発行され、占有OSの切り替えが行われる。

40

【0228】

かかる場合、切り替えボタンが押されたことを、当該切り替えボタンを制御する切り替えボタン制御ドライバ343が検知し、その旨を入出力制御装置100へ通知すればよい。

【0229】

かかる切り替えボタンと入力デバイスとが同一であった場合には、切り替えボタン制御ドライバ343は、入力された情報が、切り替え要求210に係るものであれば、当該切

50

り替え要求 210 を発行し、そうでなければ、入力デバイス制御部 341 として動作する。

【0230】

OS 处理 212 を要因とする切り替え要求 210 については、例えば、OS において、GUI を利用するプログラム 360 又は 370 によって発行される場合を考える。

【0231】

プログラム 360 が、当該切り替え要求 210 を発行する場合は、同じ OS 内であるため、入出力制御装置 100 に対してシステムコールを発行することで、当該切り替え要求 210 を発行する。

【0232】

例えば、UNIX (登録商標) 系 OS を想定した場合には、予め入出力制御装置 100 を、デバイスファイル (/dev/ioctrl 等) として、OS 340 に対して抽象化しておき、プログラム 360 が、当該デバイスファイルに対する ioctrl システムコールを発行する。

【0233】

ioctrl システムコールは、各デバイスファイルに対して独自のリクエストを送信することが可能であるため、実施形態 1 で述べたような切り替え要求 210 を生成して発行することができる。

【0234】

プログラム 370 は、入出力制御装置 100 とは別の OS に存在するため直接システムコールを発行することができない。

【0235】

そこで、OS 350 に対して切り替え要求 210 を中継する仮想入出力制御装置 351 A を設け、仮想入出力制御装置 351 A が、OS 間通信機能を利用することにより、当該切り替え要求 210 を入出力制御装置 100 へ通知する。

【0236】

プログラム 370 と仮想入出力制御装置 351 との通信については、プログラム 360 と入出力制御装置 100 と同様の手段で実現可能である。

【0237】

VMM 处理 213 を要因とする切り替え要求 210 を発行する際には、VMM 330 より signal 等の単純な通信手段を利用する。

【0238】

或いは、特許文献 1 の [0095] より始まる「Handling Virtualized Processor Exceptions」の章で記述されている VMM から OS へ発行される仮想例外を利用してよい。

【0239】

VMM 330 は、システムに 1 つしか存在しないため、このような signal 等を固定的に定義しても構わない。

【0240】

入出力制御装置 100 は、当該 signal の発行元が VMM 330 であることを確認すればよい。

【0241】

OS 状態変化を要因とする切り替え要求 210 については、状態変化検知機能が、特定 OS 或いは VMM にある場合を想定する。この場合、上述したいずれかの手段で、切り替え要求 210 の通知を実現可能である。

【0242】

(作用及び効果)

本実施形態によれば、特定 OS 340 が、入力デバイス 310 及び出力デバイス 320 を制御する手段 (入力デバイス制御部 341 及び出力デバイス制御部 342) を有することにより、全ての OS 340、350 が、入力デバイス 310 及び出力デバイス 320 を共有することができる。

10

20

30

40

50

【0243】

また、本実施形態によれば、特定のOS340のみが、安全に動作すれば、その他のOS350の誤動作や悪意のある動作等により、入力デバイス310或いは出力デバイス320が不正に占有されることを防止することができる。

【0244】

また、本実施形態によれば、各OSが備える仮想入力デバイス制御部351或いは仮想出力デバイス制御部352により、当該OSが、入力デバイス310或いは出力デバイス320を占有している場合に限り、入力デバイス制御部341或いは出力デバイス制御部342との通信によるOS間の通信が発生することとなり、無駄なOS間通信を抑えることが可能となる。

10

【0245】

また、本実施形態によれば、入出力デバイスを占有していないOS上で動作するプログラムが、当該OSが切り替え要求210を発行するように要求することが可能となる。

【0246】

例えば、あるプログラムが、ディスプレイ上に画像や文字等の出力情報を出力する場合になって初めて、OSに対して、切り替え要求210を発行するように要求する。

【0247】

かかる要求を受けて、当該OSは、入出力制御装置100に対して、当該切り替え要求210を発行する。

【0248】

この結果、当該プログラムの動作時であっても、入出力イベントが発生するぎりぎりまで、別のOSが、入出力デバイスを占有することが可能となる。

20

【0249】

(第3の実施の形態)

第3の実施形態では、各OSが入出力デバイス制御部を備える場合について述べる。

【0250】

第3の実施形態に係る入出力制御システムは、図5に示すように、OS440と、OS450と、当該OS440上で動作するプログラム460と、当該OS450上で動作するプログラム470と、入力デバイス410と、出力デバイス420と、VMM430とを備える。

30

【0251】

OS440は、入力デバイス制御部441と、出力デバイス制御部442とを備え、OS450は、入力デバイス制御部451と、出力デバイス制御部452とを備える。

【0252】

また、VMM430は、入出力制御装置100を備える。

【0253】

第3の実施形態では、全てのOS440、450が、それぞれ入出力デバイスを制御する入力デバイス制御部441或いは出力デバイス制御部442を備える。

【0254】

このような構成をとることにより、第2の実施形態で述べたシステムと比較して、入出力に係るパフォーマンスが向上する。具体的には、入出力の速度向上やリソース使用削減を見込むことができる。

40

【0255】

なぜなら、各デバイスドライバは、自分が属するOS以外の別のOSについては気にする必要がなく、直接、入出力デバイスを制御することが可能となり、OS間のスイッチに係る処理を省くことができるためである。

【0256】

このようなシステム構成においては、複数OSに属する入力デバイス制御部或いは出力デバイス制御部が、単一の入出力デバイスを同時に参照する、所謂、ハードウェアの競合状態を解決しなければならない。

50

【0257】

そのために、入出力制御装置100は、制御情報121を生成した際に、全ての入力デバイス制御部441、451或いは出力デバイス制御部442、452に対して、入力先情報230或いは出力可否情報240を出力する。

【0258】

例えば、全てのOS440、450が読み取り可能であり、VMM430が読み書き可能な主記憶装置に対して、入力先情報230或いは出力可否情報240を出力するといった方法が考えられる。

【0259】

各入力デバイス制御部441、451或いは出力デバイス制御部442、452は、入力先情報230或いは出力可否情報240により、自分が属するOS440、450が、入力デバイス410或いは出力デバイス420を占有しているか否かについて特定し、占有している場合に限り、当該入力デバイス410或いは出力デバイス420を制御する。

【0260】

入力デバイス制御部441は、第2の実施形態と同様に、入力デバイス410からのハードウェア割り込みの発生に応じて駆動される。

【0261】

特許文献1或いは非特許文献1で述べられたVMMは、本システム構成のように、1つのIRQ(Interrupt Request)ラインに、複数OS上で操作する入力デバイス制御部が対応している場合、順次、それらの入力デバイス制御部を駆動し、呼び出された入力デバイス制御部が、入力デバイス410を参照するか否かについて判断可能としている。

【0262】

この仕組みを利用し、駆動された入力デバイス制御部441は、入力先情報230を以って、OS440が、入力デバイス410を占有していれば、対応するデバイスドライバとしての処理を駆動する。なお、入力デバイス制御部451も、同様の操作を行う。

【0263】

出力デバイス制御部442は、出力可否情報240により、OS440が出力デバイス420を占有していれば、OS440の出力情報を出力デバイス420へ出力する。なお、出力デバイス制御部452も、同様の操作を行う。

【0264】

各OSが、入力デバイス制御部或いは出力デバイス制御部を備える場合の別の構成として、図6に示す構成も考えられる。図6に示す構成は、上述した図5に示す構成とほぼ同じであるが、VMM430が、新たに入出力制御部431を備える。

【0265】

入出力制御部431は、signalや仮想例外等といった各OSとの通信手段を持ち、それらの通信機能を以って、各OSの入力デバイス制御部441、451或いは出力デバイス制御部442、452を有効化或いは無効化する。

【0266】

例えば、OS440或いは450が、Linuxであった場合、Linuxは、カーネル動作中に、動的にデバイスドライバの組み込み或いは取り外しを行うことが可能である。

【0267】

そこで、例えば、入出力制御装置100により、OS440が、入力デバイス410を占有する場合に、入出力制御部431は、OS440の入力デバイス制御部441を有効化するように、OS440へ通知する。

【0268】

OS440は、かかる通知を受けて、入力デバイス制御部441をカーネルコードに組み込む。

【0269】

10

20

30

40

50

他のOS450の入力デバイス制御部451或いは出力デバイス制御部451、452に対しても同様の処理を行う。

【0270】

また、さらに、別の構成例として、図7に示す構成も考えられる。図7に示す構成では、VMM430が、新たに割り込み通知部432を備える。

【0271】

割り込み通知部432は、特許文献1或いは非特許文献1等で記述されているVMMが一般的に備える割り込みハンドラを拡張したものである。

【0272】

一般的な割り込みハンドラは、ハードウェア割り込みが発生した際、その割り込み番号を以って、どのOSに対応するデバイスドライバが存在するかについて判断し、当該OSに対して割り込みを通知する。

【0273】

IRQ番号に対応するデバイスが複数あり、当該デバイスを制御するデバイスドライバが複数OS間に分散していた場合には、当該割り込みハンドラは、順にOSに対して割り込み通知を行う。

【0274】

割り込み通知部432は、入力デバイス410からの割り込み要求の発生時に、入出力制御装置100における入力先情報230を利用することにより、割り込みを通知するOSを特定する。

【0275】

すなわち、割り込み通知部432は、割り込み発生時点で、入力デバイス410を占有するOSを特定し、当該OSに対して割り込みを通知する。

【0276】

次に、第3の実施形態に係るシステム構成において、どのようにして入出力デバイスを占有するOSを切り替えるように要求する切り替え要求210が発行されるかについて、図8に基づいて説明する。

【0277】

図8では、OS440、450が、新たに仮想入出力制御装置443、453を備え、OS440が、さらに切り替えボタン用のデバイスドライバである切り替えボタン制御ドライバ444を備える。

【0278】

プログラム460、470が、それぞれ仮想入出力制御装置443、453に対して切り替え要求210を発行する処理は、第2の実施形態において、プログラム370が、仮想入出力制御装置351に対して切り替え要求210を発行する処理と同様である。

【0279】

また、切り替えボタン制御ドライバ444が、仮想入出力制御装置443に対して切り替え要求210を発行する処理は、第2の実施形態において、切り替えボタン制御ドライバ343が、入出力制御装置100に対して切り替え要求210を発行する処理と同様である。

【0280】

さらに、VMMのデバッグ機能433及び入出力制御装置100は、VMM430によって制御される同一メモリ空間内に存在するため、切り替え要求210の発行に際して、特別な機能を必要としない。

【0281】

仮想入出力制御装置443、453が、入出力制御装置100に対して切り替え要求210を発行する際には、VMM430が、OS440に対して提供するインターフェースである「hyper call」を利用する。

【0282】

hyper callについては、同名で、非特許文献1の3.1節に説明があり、「trap call」

10

20

30

40

50

という名前で、特許文献 1 の [0 0 6 2] に説明がある。

【 0 2 8 3 】

これらの VMM-OS インタフェースは、従来技術では、システムコールと同様の使われ方をしている。すなわち、かかる VMM-OS インタフェースは、VM 上の OS の権限では処理できない特権命令を VMM に依頼する際に発行する。

【 0 2 8 4 】

本実施形態における hyper call は、これらの従来技術によって備えられるインターフェース機能を利用したものである。

【 0 2 8 5 】

hyper call で実際に通知される情報については、第 1 の実施形態で述べた通りである。 10

【 0 2 8 6 】

入出力制御装置 100 は、受け取った hyper call に対して、当該切り替え要求 210 が受理されれば、第 1 の実施形態で述べた通りの処理を行い、何らかの理由で、非受理となれば、当該 OS へエラーを返す。

【 0 2 8 7 】

次に、VMM 430 が、入出力制御装置 100 と連動して動作する QOS 制御部 434 を備える場合の動作例について、図 9 に基づいて述べる。

【 0 2 8 8 】

QOS 制御部 434 は、例えば、特許文献 1 の [0 0 9 3] より始まる scheduler の章に記述されているような CPU スケジューラを拡張したものであり、各 OS に対して割り当てるハードウェア資源を、各 OS の処理の優先度に応じて決定するものである。 20

【 0 2 8 9 】

例えば、CPU スケジューラは、各 OS の処理の優先度に応じて、CPU 制御を割り当てるために存在する。

【 0 2 9 0 】

本実施形態の QOS 制御部 434 は、入出力制御装置 100 により出力された入力先情報 230 及び出力可否情報 240 に応じて、入力デバイス 410 或いは出力デバイス 420 を占有する OS に高い優先度を設定し、優先的に CPU 等のハードウェア資源を割り当てる。

【 0 2 9 1 】

この結果、例えば、HID を占有し、実際にユーザが利用している OS の処理が優先的に処理されることとなり、ユーザビリティが向上する。

【 0 2 9 2 】

(作用及び効果)

本実施形態によれば、各 OS 440、450 が、それぞれ入出力デバイスを制御可能な手段（入力デバイス制御部 441、451 及び出力デバイス制御部 442、452）を有することにより、入出力制御装置 100 によって入力可能と判定された場合のみ、入力デバイス制御部 441、451 が、入力デバイス 410 からの入力情報を当該 OS 440、450 へ入力し、入出力制御装置 100 によって出力可能と判定された場合のみ、出力デバイス制御部 442、452 が、出力デバイス 420 へ当該 OS 440、450 からの出力情報を出力し、当該制御対象入出力デバイスの競合状態を防止することが可能となる。 40

【 0 2 9 3 】

また、各 OS 440、450 は、直接、入出力デバイスを制御可能となり、入出力時の速度が向上する。

【 0 2 9 4 】

また、VMM 430 は、入出力制御装置 100 により出力された入力先情報 230 を以って、OS 440、450 の入力デバイス制御部 441、451 を有効化或いは無効化し、入出力制御装置 100 により出力された出力可否情報 240 を以って、OS 440、450 の出力デバイス制御部 442、452 を有効化或いは無効化するように構成されている入出力制御部 431 を備える。

【0295】

この結果、各OS440、450が、それぞれ特定の入力デバイス410を制御可能な入力デバイス制御部441、451或いは出力デバイス420を制御可能な出力デバイス制御部442、452を備える場合でも、入出力制御装置100によって入出力可能と判定されたOSのみが、入力デバイス410或いは出力デバイス420を占有することで、入力デバイス410或いは出力デバイス420の競合状態を防ぐことが可能となる。

【0296】

また、各OS440、450は、直接、入出力デバイスを制御可能となり、入出力時の速度が向上する。

【0297】

さらに、入力デバイス410或いは出力デバイス420の占有は、ホストシステムの許可によってのみ可能となるため、OSの誤動作等による、入力デバイス410或いは出力デバイス420の不正な占有を防止することが可能となる。

【0298】

また、VMM430は、入力デバイス410に対する割り込みが発生した場合に、当該割り込みをOS440、450へ通知する割り込み通知部432を備え、割り込み通知部432は、入出力制御装置100により出力された入力先情報を以って、当該割り込みの通知先を特定する。

【0299】

このため、入力デバイス410からの割り込みの発生時、ホストシステムが、入出力制御装置100を利用して、当該割り込みを通知するOSを選択することができる。

【0300】

この結果、各OS440、450は、直接、入力デバイス410を制御可能となり、入力時の速度が向上する。

【0301】

また、入力デバイス410からの入力情報は、確実に当該入力デバイス410を占有するOSへ入力するため、OSの誤作動等による入力デバイス410の不正な利用を防止することが可能となる。

【0302】

また、VMM430は、入出力制御装置100により出力された入力先情報230或いは出力可否情報240に応じて、OSに対するQoS制御を行うQoS制御部434を備える。

【0303】

このため、入出力制御装置100により出力される入力先情報230或いは出力可否情報240により、ホストシステムが、各OSに対するQoS制御を行うことができる。例えば、HIDを占有するOSの処理に、高いCPU優先度を割り当てる。

【0304】

この結果、ユーザが実際に操作しているプログラムに高いCPU優先度が割り当てられることとなり、ユーザビリティが向上する。

【0305】

(第4の実施の形態)

図10は、第4の実施形態に係る入出力制御装置100を含む複数OS環境のシステム構成である。

【0306】

第4の実施形態に係る入出力制御システムは、OS540と、OS550と、当該OS540上で動作するGUIサーバ560と、当該OS550上で動作するGUIサーバ570と、当該GUIサーバ560上で動作するGUIクライアント580と、当該GUIサーバ570上で動作するGUIクライアント590と、入力デバイス510と、出力デバイス520と、VMM530とを備える。

【0307】

10

20

30

40

50

また、O S 5 4 0 は、入出力制御装置 1 0 0 と、出力デバイス 5 2 0 を制御可能な出力デバイス制御部 5 4 2 と、入力デバイス 5 1 0 を制御可能な入力デバイス制御部 5 4 1 を備える。

【 0 3 0 8 】

ここで、第 4 の実施形態に係るシステム構成は、第 2 の実施形態で述べたシステム構成と同様であるため、両者の相違点について主として説明する。

【 0 3 0 9 】

第 4 の実施形態に係るシステムでは、各 O S 5 4 0 、 5 5 0 上に、 G U I サーバ 5 6 0 、 5 7 0 並びに G U I クライアント 5 8 0 、 5 9 0 が存在する。

【 0 3 1 0 】

G U I サーバ 5 6 0 、 5 7 0 は、 ウィンドウマネージャ相当のものであり、 当該 O S 5 4 0 、 5 5 0 上における全ての G U I プログラムの入出力を一括して処理する。

【 0 3 1 1 】

G U I クライアント 5 8 0 、 5 9 0 は、 G U I サーバ 5 6 0 、 5 7 0 に入出力を託す任意の G U I プログラムである。

【 0 3 1 2 】

G U I サーバ 5 6 0 は、 G U I クライアント 5 8 0 から受け取った入出力処理要求に従って、当該入出力を処理する。

【 0 3 1 3 】

すなわち、 G U I サーバ 5 6 0 は、 G U I クライアント 5 8 0 へ入力デバイス 5 1 0 からの入力情報を入力し、 G U I クライアント 5 8 0 からの出力情報を出力デバイス 5 2 0 へ出力するように構成されている。

【 0 3 1 4 】

さらに、 G U I サーバ 5 6 0 は、 G U I サーバ 5 7 0 を G U I クライアントとして、 G U I クライアント 5 8 0 に対する処理と同様の処理を行うように構成されている。

【 0 3 1 5 】

G U I サーバ 5 6 0 と G U I サーバ 5 7 0 と間の通信は、 V M M 5 3 0 によって備えられている O S 間通信機能を以って実現される。

【 0 3 1 6 】

G U I サーバ 5 7 0 は、 G U I クライアント 5 9 0 から受け取った入出力要求に従って、当該入出力を処理する。

【 0 3 1 7 】

すなわち、 G U I サーバ 5 7 0 は、 G U I クライアント 5 9 0 に対して、 G U I サーバ 5 6 0 を介して入力デバイス 5 1 0 から受け取った入力情報を入力し、 G U I クライアント 5 9 0 からの出力情報を、 G U I サーバ 5 6 0 を介して出力デバイス 5 2 0 へ出力するように構成されている。

【 0 3 1 8 】

G U I サーバ 5 6 0 は、 さらに、入出力制御装置 1 0 0 における入力先情報 2 3 0 或いは出力可否情報 2 4 0 を元に動作する。

【 0 3 1 9 】

すなわち、これらの情報により、 O S 5 4 0 が、入力デバイス 5 1 0 或いは出力デバイス 5 2 0 を占有する場合には、 G U I サーバ 5 6 0 は、 G U I クライアント 5 8 0 の入出力を処理する。

【 0 3 2 0 】

また、 O S 5 5 0 が、入力デバイス 5 1 0 或いは出力デバイス 5 2 0 を占有する場合には、 G U I サーバ 5 6 0 は、その旨を G U I サーバ 5 7 0 へ通知する。

【 0 3 2 1 】

当該通知により、 O S 5 5 0 が、入力デバイス 5 1 0 或いは出力デバイス 5 2 0 を占有することが判明した場合に限り、 G U I サーバ 5 7 0 は、入出力処理を O S 5 5 0 へ託す。

10

20

30

40

50

【0322】

逆に、OS550が、当該デバイスを占有しない場合には、GUIクライアント590の入出力を、OS550へ託さない。

【0323】

GUIサーバ560と入出力制御装置100との間の通信は、例えば、system callにより実現される。

【0324】

(第5の実施の形態)

図11は、第5の実施形態に係る入出力制御装置100を含んだ複数OS環境のシステム構成である。

10

【0325】

第5の実施形態に係る入出力制御システムは、第2の実施形態で示した入出力制御システムとほぼ同じ構成である。

【0326】

すなわち、OS650や、当該OS650上で動作するプログラム670は、仮想入力デバイス制御ソフトウェア651を通じて、制御対象の入力デバイス610からの入力情報を受け取り、仮想出力デバイス制御ソフトウェア652を通じて、出力デバイス制御ソフトウェア642へ出力情報を出力するように構成されている。

【0327】

本実施形態が、第2の実施形態と異なる部分は、プログラム660及びプログラム670である。以下、本実施形態に係るプログラム660及びプログラム670の動作について説明する。

20

【0328】

なお、以下、プログラム660を対象として説明をするが、プログラム660と入力デバイス制御ソフトウェア641との関係、或いは、プログラム660と出力デバイス制御ソフトウェア642との関係を、プログラム670と仮想入力デバイス制御ソフトウェア651との関係、或いは、プログラム670と仮想出力デバイス制御ソフトウェア652との関係に置き換えることで、プログラム670についても当てはまるることを明記しておく。

30

【0329】

一般的な汎用OSであるLinuxやWindows(登録商標)等は、プログラムからの要求を受け取り、OS権限でのみ可能な特権処理を実行するためのインターフェースを備える。このインターフェースに対して、プログラムは、システムコールを発行する。

【0330】

プログラム660が、入力デバイス制御ソフトウェア641や出力デバイス制御ソフトウェア642を介して、入力情報を受け取り、出力情報を出力するためには、それぞれ専用のシステムコールが必要となる。

【0331】

例えば、Linuxでは、入力情報を受け取るためのシステムコールは、readシステムコールに相当し、出力情報を出力するためのシステムコールは、writeシステムコールに相当する。

40

【0332】

これらのシステムコールの処理は、システムコールを発行したプログラムとOSとの間でのコンテキストスイッチを要するため、大きな処理コストが発生する。

【0333】

本実施形態では、プログラム660が、特別な処理を行うことにより、システムコールの発行回数を減らし、システム全体のオーバヘッドを削減することが可能となる。

【0334】

本実施形態における入出力制御装置100は、制御情報121を、プログラム660が参照可能な領域に出力する。例えば、入出力制御装置100は、制御情報121を、上述

50

したようなOS間通信時に利用される共有メモリへ出力する。

【0335】

この結果、プログラム660は、制御情報121を読むことが可能となり、制御情報121の内容に応じて、システムコールを発行するかどうかについて決定することができる。

【0336】

すなわち、制御情報121により、OS640が、制御対象である入力デバイス610或いは出力デバイス620を占有している場合に限り、プログラム660は、対応するシステムコールを発行する。

【0337】

この結果、システムコールを発行したとしても、占有権がないことから入出力デバイスに対する入出力が実行不可能な場合には、システムコールの発行を抑えることができ、オーバヘッドを削減可能となる。

【0338】

このようなシステムコールの発行の削減が可能なデバイスには、例えば、オーディオデバイスが挙げられる。

【0339】

通常は、オーディオデバイスへの出力は、プログラムからリアルタイムに発生すべきである。

【0340】

つまり、ある時点で、OSが、オーディオデバイスを占有していなかったため、音楽データを出力できなかった場合、その音楽データを蓄積し、当該OSが、当該オーディオデバイスを占有した際に出力するということは、通常利用時には考えにくい。

【0341】

したがって、音楽情報を出力するプログラム660は、OS640が、オーディオデバイスを占有していなかった場合には、出力するはずであった音楽情報を無視してしまって構わず、無駄なシステムコールを発行する必要はない。

【0342】

一方で、このようなシステムコールの発行の削減が不可能なデバイスには、ディスプレイが挙げられる。

【0343】

各OSが持つディスプレイへの出力情報は、当該OSがディスプレイを占有していようがいまいが、当該OSのフレームバッファメモリへ蓄積・更新されていかなければならない。

【0344】

こうすることで、ディスプレイの占有権が切り替わった場合でも、ユーザは、その時点でのOSの状況をディスプレイから正確に把握することが可能となる。

【0345】

そのためには、当該OSは、常に自フレームバッファメモリを更新していかなければならず、制御情報121の内容に関わらず、プログラム660から出力情報を受け取り続けなければならない。

【0346】

上述のように、制御情報121を参照して動作を変更するためには、プログラム660への改造を施せばよい。

【0347】

すなわち、プログラム660は、入出力が発生する部分のコードについて、制御情報121を参照するステップと、制御情報121の内容に応じて、システムコールを発行するか否かについて決定するステップを有すればよい。

【0348】

しかし、システムに応じて、プログラム660の内容を変更することは、プログラマに

10

20

30

40

50

とって負担が大きく、また、プログラム 660 の変更無しに動作可能とするという VMM の利点を消してしまう。

【0349】

さらに、上述したようなデバイスの違いによる改造の要不要をプログラマが判断するの 10 は非常に難しい。

【0350】

本実施形態を実装する上での現実的な手段は、プログラム 660 を直接改造することではなく、プログラム 660 が入出力の際に呼び出す API 661 を改造することである。

【0351】

一般的に、プログラマは、システムコールを直接コードに組み込むことはせず、システムが用意した API を呼び出すことで、間接的にシステムコールを発行するようにプログラムを作成する。 10

【0352】

API を利用することは、システムコールを直接呼び出すようなコードと比較して、プログラミングが容易となり、また利便性も向上する。

【0353】

また、このような API は、デバイス毎に用意されている。したがって、システムコールの発行の要不要の判断が必要なデバイスを呼び出すための API にのみ改造を施し、当該 API が、制御情報 121 を参照するステップと、参照結果に応じて、システムコールの発行を行うか否かについて決定するステップと、当該決定に応じてシステムコールを発行するステップとを備えるようにすればよい。 20

【0354】

これにより、プログラム 660 の作成者は、本実施形態に関わる如何なる変更も必要な 20 くなり、しかも、システムコールの発行回数を抑えられることで、システム全体のパフォーマンスが向上することとなる。

【0355】

(作用及び効果)

本実施形態によれば、OS 640 上で動作するプログラム 660 が、制御情報 121 を 30 参照し、その時点で、当該 OS 640 が、入力デバイス 610 或いは出力デバイス 620 を占有しているか否かについて判断し、占有している場合に限り、当該 OS に対して入出力に関わるシステムコールを発行することにより、無駄なシステムコールの発行に係る処理コストを削減することが可能となる。

【0356】

以上、上述の実施形態を用いて本発明について詳細に説明したが、当業者にとっては、本発明が本明細書中に説明した実施形態に限定されるものではないということは明らかである。本発明は、特許請求の範囲の記載により定まる本発明の趣旨及び範囲を逸脱することなく修正及び変更態様として実施することができる。従って、本明細書の記載は、例示説明を目的とするものであり、本発明に対して何ら制限的な意味を有するものではない。

【図面の簡単な説明】

【0357】

【図 1】第 1 の実施形態に係る入出力制御システムの構成ブロック図である。 40

【図 2】第 1 の実施形態に係る入出力制御方法を示すフローチャートである。

【図 3】第 2 の実施形態に係る入出力制御システムの構成ブロック図である(その 1)。

【図 4】第 2 の実施形態に係る入出力制御システムの構成ブロック図である(その 2)。

【図 5】第 3 の実施形態に係る入出力制御システムの構成ブロック図である(その 1)。

【図 6】第 3 の実施形態に係る入出力制御システムの構成ブロック図である(その 2)。

【図 7】第 3 の実施形態に係る入出力制御システムの構成ブロック図である(その 3)。

【図 8】第 3 の実施形態に係る入出力制御システムの構成ブロック図である(その 4)。

【図 9】第 3 の実施形態に係る入出力制御システムの構成ブロック図である(その 5)。

【図 10】第 4 の実施形態に係る入出力制御システムの構成ブロック図である。 50

【図11】第5の実施形態に係る入出力制御システムの構成ブロック図である。

【図12】従来の入出力制御システムの構成ブロック図である（その1）。

【図13】従来の入出力制御システムの構成ブロック図である（その2）。

【符号の説明】

【0358】

100 ... 入出力制御装置

110 ... 要求受理部

120 ... 制御情報生成部

121 ... 制御情報

130 ... 制御情報出力部

10

210 ... 切り替え要求

211 ... 外部入力

212 ... OS処理

213 ... VMM処理

214 ... OS状態変化

220 ... 入出力デバイスリスト

230 ... 入力先情報

240 ... 出力可否情報

250 ... 要求規則

310, 410, 510 ... 入力デバイス

20

320, 420, 520 ... 出力デバイス

330, 430, 530 ... VMM

331, 433 ... デバッグ機能

340, 350, 440, 450, 540, 550 ... OS

341, 441, 451, 541 ... 入力デバイス制御部

342, 442, 452, 542 ... 出力デバイス制御部

343, 444 ... 切り替えボタン制御ドライバ

351A, 443, 453 ... 仮想入出力制御装置

351 ... 仮想入力デバイス制御部

352 ... 仮想出力デバイス制御部

30

360, 370, 460, 470 ... プログラム

431 ... 入出力制御部

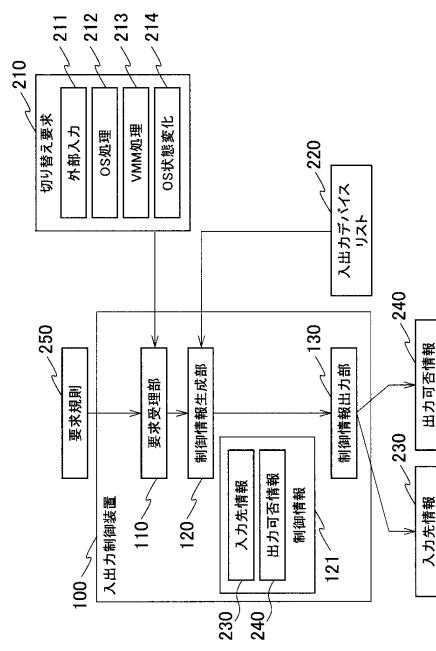
432 ... 割り込み通知部

434 ... QOS制御部

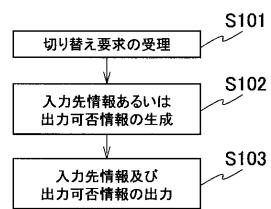
560, 570 ... GUIサーバ

580, 590 ... GUIクライアント

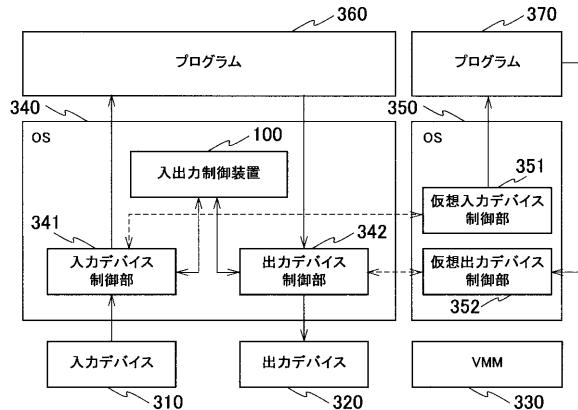
【図1】



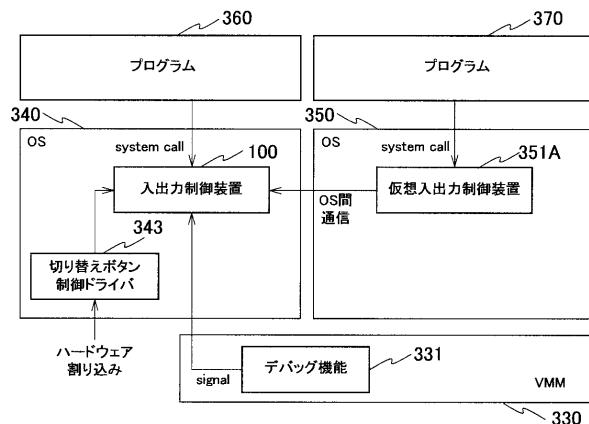
【図2】



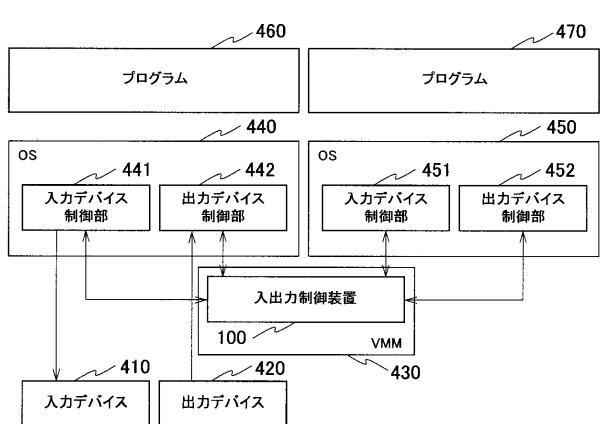
【図3】



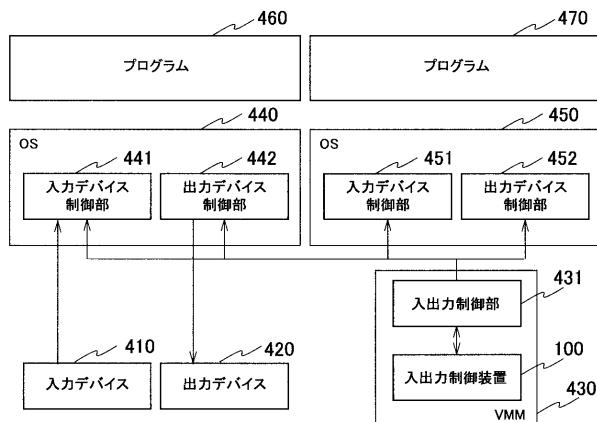
【図4】



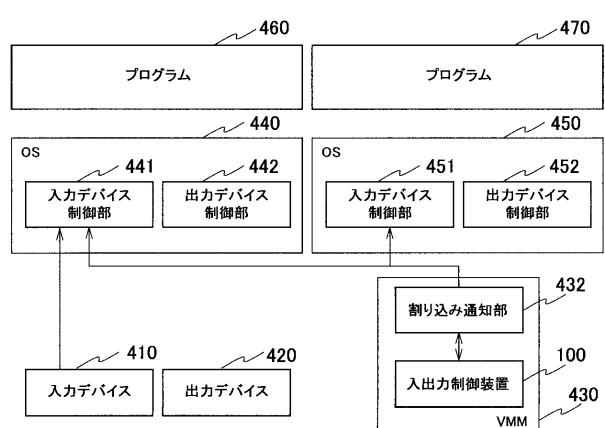
【図5】



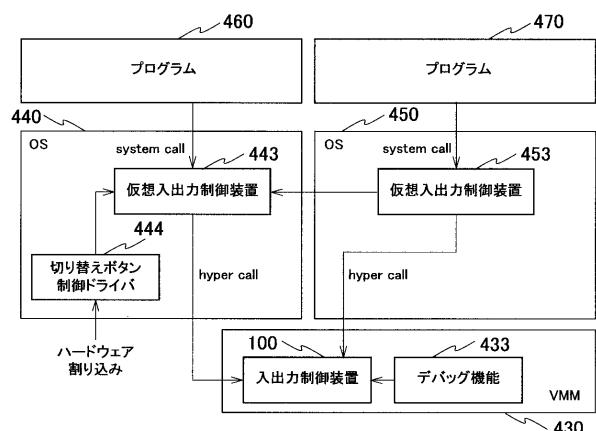
【図6】



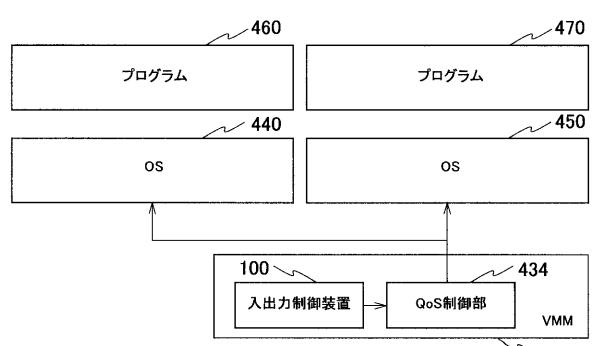
【図7】



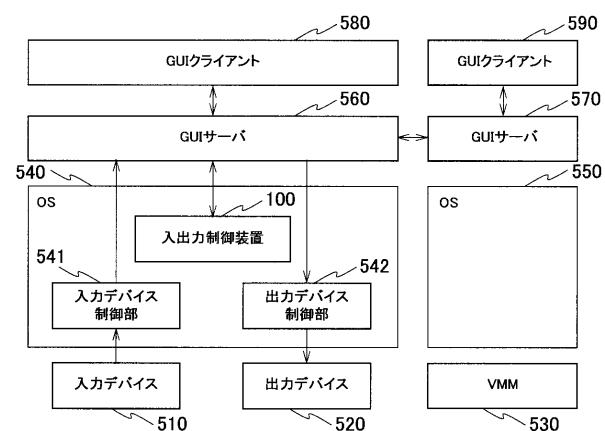
【図8】



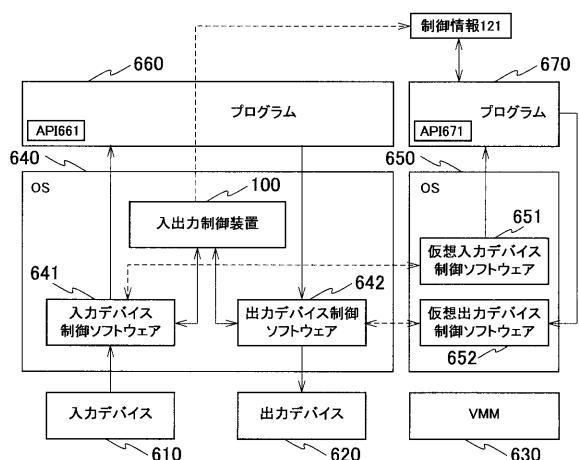
【図9】



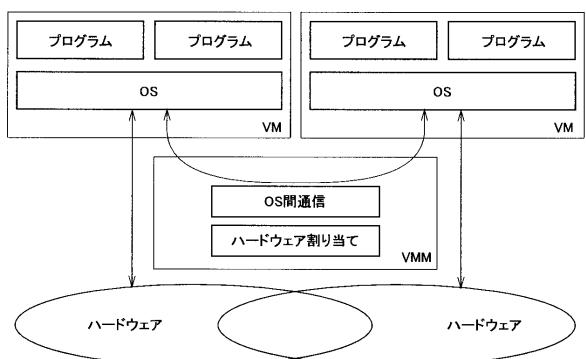
【図10】



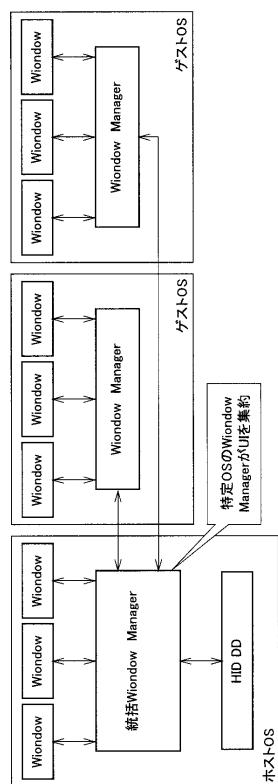
【図11】



【図12】



【図13】



フロントページの続き

(72)発明者 藤本 拓

東京都千代田区永田町二丁目11番1号 株式会社エヌ・ティ・ティ・ドコモ内

(72)発明者 鈴木 敬

東京都千代田区永田町二丁目11番1号 株式会社エヌ・ティ・ティ・ドコモ内

(72)発明者 太田 賢

東京都千代田区永田町二丁目11番1号 株式会社エヌ・ティ・ティ・ドコモ内

F ターム(参考) 5B014 FB05