

(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2017-111499

(P2017-111499A)

(43) 公開日 平成29年6月22日(2017.6.22)

(51) Int.Cl.			F I			テーマコード (参考)	
G06F	9/48	(2006.01)	G06F	9/46	455Z	5B047	
G06F	9/46	(2006.01)	G06F	9/46	35O	5C062	
G06F	9/50	(2006.01)	G06F	9/46	462Z	5C073	
H04N	1/00	(2006.01)	H04N	1/00	C		
G06T	1/60	(2006.01)	G06T	1/60	450A		
			審査請求 有 請求項の数 16 O L			(全 34 頁)	最終頁に続く

(21) 出願番号 特願2015-243357 (P2015-243357)
 (22) 出願日 平成27年12月14日 (2015.12.14)

(71) 出願人 000001270
 コニカミノルタ株式会社
 東京都千代田区丸の内二丁目7番2号
 (74) 代理人 110001195
 特許業務法人深見特許事務所
 (72) 発明者 鈴江 正
 東京都千代田区丸の内二丁目7番2号 コ
 ニカミノルタ株式会社内
 (72) 発明者 上田 滋之
 東京都千代田区丸の内二丁目7番2号 コ
 ニカミノルタ株式会社内
 (72) 発明者 細野 真央
 東京都千代田区丸の内二丁目7番2号 コ
 ニカミノルタ株式会社内

最終頁に続く

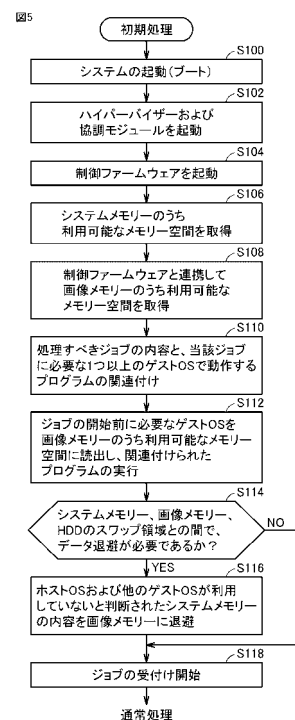
(54) 【発明の名称】 画像形成装置および画像形成装置で実行されるプログラム

(57) 【要約】 (修正有)

【課題】ジョブを優先的に処理しつつ、複数のOSを同時実行させるための画像形成装置を提供する。

【解決手段】画像形成装置は、ハイパーバイザーと、ホストOS上で実行され、画像形成装置の主たる制御を司る制御ファームウェアと、ハイパーバイザーと同一レベルで動作する協調モジュールとを含む。リソースは、制御ファームウェアにより管理される画像処理用に適合された画像メモリを含む。協調モジュールは、制御ファームウェアと連携して画像メモリのうち利用可能なメモリー空間を取得する取得手段と、処理すべきジョブの内容と、当該ジョブに必要な1つ以上のゲストOSで動作するプログラムを関連付ける関連付け手段と、ジョブの開始前に必要なゲストOSを画像メモリのうち利用可能なメモリー空間に読出し、関連付けられたプログラムを実行させる。

【選択図】図5



【特許請求の範囲】**【請求項 1】**

画像形成装置であって、

プロセッサおよびシステムメモリーを含むリソースを用いて、ホストOSおよび少なくとも1つのゲストOSを含む複数のOSの同時実行を制御するハイパーバイザーと、

前記ホストOS上で実行され、前記画像形成装置の主たる制御を司る制御ファームウェアと、

前記ハイパーバイザーと同一レベルで動作する協調モジュールと、

前記システムメモリーのうち前記ホストOSおよび前記ホストOSと同時に実行されるゲストOSが利用していない空きメモリー空間を管理する空きメモリー管理モジュールとを備え、

10

前記リソースは、前記制御ファームウェアにより管理される画像処理用に適合された画像メモリーを含み、

前記協調モジュールは、

前記制御ファームウェアと連携して前記画像メモリーのうち利用可能なメモリー空間を取得する取得手段と、

処理すべきジョブの内容と、当該ジョブに必要な1つ以上のゲストOSで動作するプログラムを関連付ける関連付け手段と、

ジョブの開始前に必要なゲストOSを前記画像メモリーのうち利用可能なメモリー空間に読み出し、関連付けられたプログラムを実行させる実行手段と、

20

前記ホストOSおよび他のゲストOSが利用していないと判断されたシステムメモリーの内容を画像メモリーに退避することで、新たに実行されたゲストOSのシステムメモリーとして利用可能にする退避手段とを含む、画像形成装置。

【請求項 2】

前記協調モジュールは、

メモリー空間の種別に区分して、単位メモリーごとに利用しているOSを特定可能なIDを含むリストを管理する管理手段と、

必要なシステムメモリーが利用可能になるまでブロック単位で並列にメモリー転送を実施し、ゲストOSに対してメモリー空間を割り当てる割り当て手段と、

メモリー転送の完了後にゲストOSへの切り替えを行なう切り替え手段とを含む、請求項1に記載の画像形成装置。

30

【請求項 3】

前記協調モジュールは、

前記ホストOSへ切り替える場合に、

先に実行されたゲストOSおよび当該ゲストOSで動作するプログラムが利用するシステムメモリーを一時凍結する処理と、

前記先に転送されたメモリーを順次入れ替える処理と、

入れ替え完了後に前記ホストOSへ切り替える処理とを、連続動作させる連続動作実行手段を含む、請求項2に記載の画像形成装置。

【請求項 4】

40

前記協調モジュールは、

読み出し済みのゲストOSがジョブ実行により破壊されたことを検知する検知手段と、

ジョブ開始前に検知された破壊されたゲストOSを再度読み出す読み出し手段とを備える、請求項1～3のいずれか1項に記載の画像形成装置。

【請求項 5】

前記退避手段は、メインメモリー内のシステムメモリーからメインメモリー内の画像メモリーに前記システムメモリーの内容を退避するとともに、メインメモリー内の画像メモリーに空きがなければ、システムメモリーから画像処理のためのサブメモリー内の画像メモリーに退避する、請求項1～4のいずれか1項に記載の画像形成装置。

【請求項 6】

50

前記協調モジュールは、画像メモリの種別に応じてジョブおよびジョブ情報に基づき優先度を設定し、当該設定した優先度に基づいてジョブの実行を制御する、請求項１～５のいずれか１項に記載の画像形成装置。

【請求項７】

前記協調モジュールは、前記ホストＯＳおよび前記ゲストＯＳについて、連続するメモリ空間を優先して前記画像メモリに退避する、請求項１～６のいずれか１項に記載の画像形成装置。

【請求項８】

前記協調モジュールは、ゲストＯＳを外部システムへマイグレーションしたときに退避したデータ内容を移動管理する機能をさらに含む、請求項１～７のいずれか１項に記載の画像形成装置。

【請求項９】

前記ゲストＯＳ上の情報を表示する表示手段をさらに備える、請求項１～８のいずれか１項に記載の画像形成装置。

【請求項１０】

前記協調モジュールは、前記ゲストＯＳ上のユーザー管理情報を必要に応じて退避前に前記制御ファームウェアと共有する共有手段をさらに含む、請求項１～９のいずれか１項に記載の画像形成装置。

【請求項１１】

前記リソースは、画像処理ＡＳＩＣをさらに含み、

前記協調モジュールは、前記画像処理ＡＳＩＣと協調動作する、請求項１～１０のいずれか１項に記載の画像形成装置。

【請求項１２】

前記画像処理用に適合された画像メモリは、メインメモリにおけるシステムメモリと異なる領域および前記画像処理ＡＳＩＣにより管理されるサブメモリが有する領域、の少なくとも一方を含む、請求項１１に記載の画像形成装置。

【請求項１３】

前記画像メモリは、ディスク媒体以外の記憶装置である、請求項１～１２のいずれか１項に記載の画像形成装置。

【請求項１４】

前記画像メモリは、ＲＡＭである、請求項１３に記載の画像形成装置。

【請求項１５】

前記リソースは、二次記憶装置をさらに含み、

前記協調モジュールは、前記ゲストＯＳへ切り替えた後に、前記ゲストＯＳから前記ホストＯＳではなく他のゲストＯＳに切り替える場合に、前記ホストＯＳに関するファイルメモリのデータを二次記憶装置へ順次並列転送する処理、先に実行されたゲストＯＳおよび当該ゲストＯＳで動作するプログラムが利用するシステムメモリを一時凍結する処理、前記二次記憶装置へ転送されたファイルメモリに対して一時凍結したシステムメモリを順次並列転送する処理と、他のゲストＯＳが必要とするシステムメモリになるまで先のゲストＯＳが転送完了したシステムメモリから順次確保する処理と、他のゲストＯＳが動作可能なシステムメモリが確保されたことで切り替えする処理とを連続動作させる連続動作手段をさらに備える、請求項２に記載の画像形成装置。

【請求項１６】

画像形成装置で実行されるプログラムであって、前記プログラムは、

プロセッサおよびシステムメモリを含むリソースを用いて、ホストＯＳおよび少なくとも１つのゲストＯＳを含む複数のＯＳの同時実行を制御するハイパーバイザーを実現するプログラムと、

前記ホストＯＳ上で実行され、前記画像形成装置の主たる制御を司る制御ファームウェアと、

前記ハイパーバイザーと同一レベルで動作する協調モジュールを実現するプログラムと

10

20

30

40

50

、

前記システムメモリーのうち前記ホストOSおよび前記ホストOSと同時に実行されるゲストOSが利用していない空きメモリー空間を管理する空きメモリー管理モジュールを実現するプログラムとを備え、

前記リソースは、前記制御ファームウェアにより管理される画像処理用に適合された画像メモリーを含み、

前記協調モジュールは、

前記制御ファームウェアと連携して前記画像メモリーのうち利用可能なメモリー空間を取得する取得機能と、

処理すべきジョブの内容と、当該ジョブに必要な1つ以上のゲストOSで動作するプログラムを関連付ける関連付け機能と、

ジョブの開始前に必要なゲストOSを前記画像メモリーのうち利用可能なメモリー空間に読み出し、関連付けられたプログラムを実行させる実行機能と、

前記ホストOSおよび他のゲストOSが利用していないと判断されたシステムメモリーの内容を画像メモリーに退避することで、新たに実行されたゲストOSのシステムメモリーとして利用可能にする退避機能とを含む、画像形成装置で実行されるプログラム。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、複数のOSを切り替え実行可能な画像形成装置およびそのような画像形成装置で実行されるプログラムに関する。

【背景技術】

【0002】

近年の情報通信技術の進歩に伴って、コンピューターの仮想化技術が様々な分野に応用されつつある。このような仮想化技術においては、例えば、ハイパーバイザーを利用して、主となるOS（Operating System）（以下では、「ホストOS」とも称する。）に加えて、1または複数の異なるOS（以下では、「ゲストOS」とも称する。）を、共通のハードウェア上で同時実行させることが可能である。

【0003】

例えば、特開2001-256066号公報（特許文献1）は、単一システム内に共存させた複数のオペレーティングシステムを高速に切り替えて使用できる環境を開示する。

【0004】

また、「IBM PowerVM Virtualization Active Memory Sharing」（非特許文献1）は、複数の論理メモリーパーティション（LPAR：Logical PARTition）を一つの共通物理メモリー空間としてまとめ、論理メモリーパーティション割り当て先としてマルチOS間で共有利用するシステムを開示する。

【0005】

一般的に、ホストOSに加えて、1または複数のゲストOSが同時実行される場合には、実行されるOS（特に、ゲストOS）の数に比例して、より大きなメモリー空間が必要となる。そのため、多くの仮想化技術においては、ゲストOSの実行によってハードウェア的に用意されているシステムメモリーの上限を超える場合に、オーバーコミットメントを許容する仕組みが備えられている。例えば、特開2014-157476号公報（特許文献2）は、仮想マシン間で共有する物理リソースの不足を検出する構成を開示する。

【0006】

オーバーコミットメントを許容する仕組みの具体例としては、ハードディスクなどの二次記憶装置にメモリー空間として利用されるメモリー領域を確保し、その確保したメモリー領域にシステムメモリーに格納されていた内容を一時的に退避する処理が含まれる。

【0007】

しかしながら、ハードディスクなどの二次記憶装置は、DRAM（Dynamic Random Access Memory）などのシステムメモリーに比較して応答性が劣るため、例えば、ホストOS

10

20

30

40

50

からゲストOSへ切り替えの実行に大きな時間を要する。このような課題は、有限のリソースにおいては解決が難しい。十分に大きなリソースを用意すれば、このような課題が発生する可能性を低減することはできる。

【0008】

画像形成装置においても、コスト制約上、利用可能なリソースをそれほど大きくすることができない。特に、最新のコンピューティング環境のように処理性能が高いハードウェアではなく、コストメリットのあるハードウェアが採用されることが一般的である。このようなコスト制約の下、画像形成装置では、ジョブをより優先して処理できるようにメモリー構成が設計される。

【0009】

例えば、特開2014-238665（特許文献3）は、ハードウェア資源を用いて画像形成に係わる処理が装置制御ファームウェアにより行なわれる第1OS、および、第1OSとは別であって第1OS上の第2OS、がそれぞれ動作可能に構成されている画像形成装置を開示する。装置制御ファームウェアは、該装置制御ファームウェアでの処理に伴って予め定められた特定の状態が検出された場合に、第2OSを起動するよう制御するとともに、当該第2OSに対するログイン処理を実行するよう制御する。この画像形成装置では、第1OSユーザー名を第2OSユーザー名として用いてログイン手順を簡略化することで、第2OSログイン手順を高速化している。

【先行技術文献】

【特許文献】

【0010】

【特許文献1】特開2001-256066号公報

【特許文献2】特開2014-157476号公報

【特許文献3】特開2014-238665号公報

【非特許文献】

【0011】

【非特許文献1】"IBM PowerVM Virtualization Active Memory Sharing", Second Edition (June 2011), IBM International Technical Support Organization, June 2011

【発明の概要】

【発明が解決しようとする課題】

【0012】

特許文献3に開示された画像形成装置において、画像処理などのジョブが発生した場合に、メモリーなどのリソースが枯渇するときには、本体制御の第1OSへリソースを明け渡す必要があり、この必要性に伴って、第2OSのリソース解放のためのログアウト処理が必要になる。このログアウト処理に要する時間によって、画像処理の実行開始が遅れる可能性がある。

【0013】

従来の画像形成装置においては、ジョブの優先処理と複数のOSを高速に切り替えて実行することとを両立させるメモリーの利用形態が考慮されていない。そのため、例えば、ゲストOS上のアプリケーションで認証が必要なジョブを実行する際に、システムメモリーにリソース不足が発生すると、システムメモリーの内容を退避する処理に要する時間が生じ、ホストOSからゲストOSに切り替えて実行する際に遅延が生じる。その結果、ジョブの処理に対するユーザビリティの劣化、および、生産性の低下といった課題が生じ得る。

【0014】

画像形成装置においては、ジョブの処理に影響を与えないゲストOSの機能を利用する作業であれば許容できるが、ジョブの処理に連携したゲストOS機能を利用する場合には、ホストOSからゲストOSへの高速な切り替えが必要である。

【0015】

本願発明者らは、ジョブの処理を優先的に実行しつつ、複数のOSを高速に切り替えて

10

20

30

40

50

実行しなければならないという新規な課題を見出した。しかしながら、このような新規な課題は、単に、従来のハイパーバイザーなどを画像形成装置に適用するだけでは、解決することができない。

【0016】

そのため、画像形成装置において、ジョブを優先的に処理しつつ、複数のOSを同時実行させるための新規な構成が要望されている。

【課題を解決するための手段】

【0017】

本発明のある局面に従う画像形成装置は、プロセッサおよびシステムメモリを含むリソースを用いて、ホストOSおよび少なくとも1つのゲストOSを含む複数のOSの同時実行を制御するハイパーバイザーと、ホストOS上で実行され、画像形成装置の主たる制御を司る制御ファームウェアと、ハイパーバイザーと同一レベルで動作する協調モジュールと、システムメモリのうちホストOSおよびホストOSと同時実行されるゲストOSが利用していない空きメモリ空間を管理する空きメモリ管理モジュールとを含む。リソースは、制御ファームウェアにより管理される画像処理用に適合された画像メモリを含む。協調モジュールは、制御ファームウェアと連携して画像メモリのうち利用可能なメモリ空間を取得する取得手段と、処理すべきジョブの内容と、当該ジョブに必要な1つ以上のゲストOSで動作するプログラムを関連付ける関連付け手段と、ジョブの開始前に必要なゲストOSを画像メモリのうち利用可能なメモリ空間に読み出し、関連付けられたプログラムを実行させる実行手段と、ホストOSおよび他のゲストOSが利用していないと判断されたシステムメモリの内容を画像メモリに退避することで、新たに実行されたゲストOSのシステムメモリとして利用可能にする退避手段とを含む。

【0018】

好ましくは、協調モジュールは、メモリ空間の種別に区分して、単位メモリごとに利用しているOSを特定可能なIDを含むリストを管理する管理手段と、必要なシステムメモリが利用可能になるまでブロック単位で並列にメモリ転送を実施し、ゲストOSに対してメモリ空間を割り当てる割り当て手段と、メモリ転送の完了後にゲストOSへの切り替えを行なう切り替え手段とを含む。

【0019】

好ましくは、協調モジュールは、ホストOSへ切り替える場合に、先に実行されたゲストOSおよび当該ゲストOSで動作するプログラムが利用するシステムメモリを一時凍結する処理と、先に転送されたメモリを順次入れ替える処理と、入れ替え完了後にホストOSへ切り替える処理とを、連続動作させる連続動作実行手段を含む。

【0020】

好ましくは、協調モジュールは、読み出し済みのゲストOSがジョブ実行により破壊されたことを検知する検知手段と、ジョブ開始前に検知された破壊されたゲストOSを再度読み出す読み出し手段とを含む。

【0021】

好ましくは、退避手段は、メインメモリ内のシステムメモリからメインメモリ内の画像メモリにシステムメモリの内容を退避するとともに、メインメモリ内の画像メモリに空きがなければ、システムメモリから画像処理のためのサブメモリ内の画像メモリに退避する。

【0022】

好ましくは、協調モジュールは、画像メモリの種別に応じてジョブおよびジョブ情報に基づき優先度を設定し、当該設定した優先度に基づいてジョブの実行を制御する。

【0023】

好ましくは、協調モジュールは、ホストOSおよびゲストOSについて、連続するメモリ空間を優先して画像メモリに退避する。

【0024】

好ましくは、協調モジュールは、ゲストOSを外部システムへマイグレーションしたと

きに退避したデータ内容を移動管理する機能をさらに含む。

【0025】

好ましくは、画像形成装置は、ゲストOS上の情報を表示する表示手段をさらに含む。

好ましくは、協調モジュールは、ゲストOS上のユーザー管理情報を必要に応じて退避前に制御ファームウェアと共有する共有手段をさらに含む。

【0026】

好ましくは、リソースは、画像処理ASICをさらに含む。協調モジュールは、画像処理ASICと協調動作する。

【0027】

好ましくは、画像処理用に適合された画像メモリーは、メインメモリーにおけるシステムメモリーと異なる領域および画像処理ASICにより管理されるサブメモリーが有する領域、の少なくとも一方を含む。

【0028】

好ましくは、画像メモリーは、ディスク媒体以外の記憶装置である。

好ましくは、画像メモリーは、RAMである。

【0029】

好ましくは、リソースは、二次記憶装置をさらに含む。協調モジュールは、ゲストOSへ切り替えた後に、ゲストOSからホストOSではなく他のゲストOSに切り替える場合に、ホストOSに関するファイルメモリーのデータを二次記憶装置へ順次並列転送する処理、先に実行されたゲストOSおよび当該ゲストOSで動作するプログラムが利用するシステムメモリーを一時凍結する処理、二次記憶装置へ転送されたファイルメモリーに対して一時凍結したシステムメモリーを順次並列転送する処理と、他のゲストOSが必要とするシステムメモリーになるまで先のゲストOSが転送完了したシステムメモリーから順次確保する処理と、他のゲストOSが動作可能なシステムメモリーが確保されたことで切り替えする処理とを連続動作させる連続動作手段をさらに含む。

【0030】

本発明のある局面に従う画像形成装置で実行されるプログラムであって、プログラムは、プロセッサおよびシステムメモリーを含むリソースを用いて、ホストOSおよび少なくとも1つのゲストOSを含む複数のOSの同時実行を制御するハイパーバイザーを実現するプログラムと、ホストOS上で実行され、画像形成装置の主たる制御を司る制御ファームウェアと、ハイパーバイザーと同一レベルで動作する協調モジュールを実現するプログラムと、システムメモリーのうちホストOSおよびホストOSと同時実行されるゲストOSが利用していない空きメモリー空間を管理する空きメモリー管理モジュールを実現するプログラムとを含む。リソースは、制御ファームウェアにより管理される画像処理用に適合された画像メモリーを含む。協調モジュールは、制御ファームウェアと連携して画像メモリーのうち利用可能なメモリー空間を取得する取得機能と、処理すべきジョブの内容と、当該ジョブに必要な1つ以上のゲストOSで動作するプログラムを関連付ける関連付け機能と、ジョブの開始前に必要なゲストOSを画像メモリーのうち利用可能なメモリー空間に読出し、関連付けられたプログラムを実行させる実行機能と、ホストOSおよび他のゲストOSが利用していないと判断されたシステムメモリーの内容を画像メモリーに退避することで、新たに実行されたゲストOSのシステムメモリーとして利用可能にする退避機能とを含む。

【発明の効果】

【0031】

本開示によれば、ジョブを優先的に処理しつつ、複数のOSを同時実行させるための新規な画像形成装置を実現できる。

【図面の簡単な説明】

【0032】

【図1】本実施の形態に従う画像形成装置の全体構成を示す概略断面図である。

【図2】本実施の形態に従う画像形成装置の制御部のハードウェア構成を示すブロック図

10

20

30

40

50

である。

【図 3】本実施の形態に従う画像形成装置の制御部でのプログラムの実行状態の一例を示す模式図である。

【図 4】本実施の形態に従う画像形成装置の制御部による処理例 1 での処理手順を示すシーケンス図である。

【図 5】本実施の形態に従う画像形成装置の制御部の初期処理の手順を示すフローチャートである。

【図 6】本実施の形態に従う画像形成装置の制御部でのプログラムの別の実行状態の一例を示す模式図である。

【図 7】本実施の形態に従う画像形成装置の制御部におけるブロック分割された画像メモリーの利用形態を説明するための模式図である。

10

【図 8】本実施の形態に従う画像形成装置の制御部におけるブロック分割された画像メモリーの管理方法を説明するための模式図である。

【図 9】本実施の形態に従う画像形成装置の制御部における複数のメモリー間でのデータ管理の一例を示す模式図である。

【図 10】本実施の形態に従う画像形成装置の制御部におけるメモリー空間の利用形態の一例を示す図である。

【図 11】本実施の形態に従う画像形成装置の制御部による処理例 2 でのメモリー確保に係る処理手順を示すフローチャートである。

【図 12】本実施の形態に従う画像形成装置の制御部でのメモリー領域のマネージメント処理の手順を示すフローチャートである。

20

【発明を実施するための形態】

【0033】

本発明の実施の形態について、図面を参照しながら詳細に説明する。なお、図中の同一または相当部分については、同一符号を付してその説明は繰り返さない。

【0034】

< A . 課題および解決手段の概要 >

まず、本発明の実施の形態に従う画像形成装置で採用する解決手段の概要について、関連技術における課題とともに説明する。

【0035】

30

上述したように、従来のハイパーバイザーなどを画像形成装置に適用した仮想化環境では、システムメモリーなどのリソースが枯渇した際には、それらがジョブの処理に必要であるか否かにかかわらず、ゲスト OS の起動もしくは実行、ならびに、ゲスト OS 上でのアプリケーションの起動もしくは実行が完了するまでに遅延時間が発生し得る。

【0036】

すなわち、上述したように、ゲスト OS への切り替え時に、システムメモリーの内容を退避する処理が発生することによる時間遅延が課題であり、このような課題は、従来の画像形成装置と従来のハイパーバイザーとを単に組み合わせただけでは解決できない。

【0037】

40

本願発明者らは、鋭意検討の結果、これらの課題を解決する手段として、画像形成装置において、システムメモリーとは別に設けられるシステムメモリーを仮想化環境の実現に利用するという新規な着想に至った。すなわち、画像形成装置固有のメモリー構成の設計として、OS を含む各種プログラムを実行するために利用されるシステムメモリーに加えて、ジョブを処理するために利用される画像メモリーが用意される。画像メモリーに対しては、比較的大きなメモリー空間がジョブの処理専用に割り当てられる。画像メモリーは、アクセス速度などを考慮して、ディスク媒体以外の記憶装置として実装されることが好ましい。典型的には、記画像メモリーは、DRAM などの RAM (Random Access Memory) が採用される。

【0038】

一方で、画像メモリーは、ホスト OS 上で実行される制御ファームウェアが操作および

50

管理するメモリー空間であり、ゲストOSおよびハイパーバイザーがこれらのメモリー空間に勝手にアクセスすると、ジョブの処理が破たんする恐れもある。一般的には、メモリー種別、例えば、システムメモリー、サブメモリー（IOアクセス）、入出力バッファなどの処理に応じて最適化されたメモリーなどに応じて、利用形態は異なる。しかしながら、従来のハイパーバイザーは、ゲストOSが利用するメモリー空間の競合管理が主たる処理であり、個々のOSが利用する専用のメモリー空間（画像メモリー）の内容を把握することはできない。また、従来のハイパーバイザーは、個々のOSが利用する専用のメモリー空間に対する詳細なメモリー制御を行なうこともできない。

【0039】

ここで、ホストOS上で実行される制御ファームウェアと連携することで、画像メモリーへのアクセス状態および画像メモリー中のデータの格納状態などを取得し、詳細なメモリー制御を実現することも考えられる。しかしながら、ハイパーバイザーが制御ファームウェアと連携するために、制御ファームウェアを稼働させるホストOSへの余分な切り替えが発生すること、および、画像メモリーについての複数の制御主体間での排他制御などの余分な負荷が発生することは好ましくない。

【0040】

本実施の形態に従う画像形成装置では、ジョブの処理に利用される画像メモリーを、ゲストOSを実行するために必要な一時退避メモリー（スワップメモリー）として利用することで、複数のOSを高速に切り替え実行することで、ホストOSでの処理が必要なジョブであっても、可能な限り遅延時間を発生させることなく、優先的な処理を可能にする。また、本実施の形態に従う画像形成装置は、このようなシステムメモリーおよび画像メモリーのそれぞれのメモリー空間を効率的に利用可能な制御方法を採用する。

【0041】

< B . 画像形成装置の全体構成 >

次に、本実施の形態に従う画像形成装置1の全体構成について説明する。

【0042】

図1は、本実施の形態に従う画像形成装置1の全体構成を示す概略断面図である。図1を参照して、画像形成装置1は、自動原稿送り装置2と、イメージスキャナー3と、プリントエンジン4と、給紙部5と、制御部100とを含む。

【0043】

自動原稿送り装置2は、原稿の連続的なスキャンを行なうためのものであり、原稿給紙台21と、送出口ローラー22と、レジストローラー23と、搬送ドラム24と、排紙台25とを含む。スキャン対象の原稿は、原稿給紙台21上にセットされ、送出口ローラー22の作動により一枚ずつ送り出される。そして、この送り出された原稿は、レジストローラー23により一旦停止されて先端が整えられた後に、搬送ドラム24へ搬送される。さらに、この原稿は、搬送ドラム24のドラム面と一体に回転し、その過程において後述するイメージスキャナー3により画像面がスキャンされる。その後、原稿は、搬送ドラム24のドラム面を略半周した位置においてドラム面から分離されて排紙台25へ排出される。

【0044】

イメージスキャナー3は、撮像デバイス33と、原稿台35とを含む。撮像デバイス33は、被写体である原稿に対する相対位置を時間的に変化させて、原稿の画像を読み取って画像データを生成する。この画像データを含むデータがジョブとして処理される。撮像デバイス33は、主要な構成要素として、原稿に対して光を照射する光源と、光源から照射された光が原稿で反射して生じる画像を取得するイメージセンサーと、イメージセンサーから画像信号を出力するためのAD（Analog to Digital：アナログデジタル）変換器と、イメージセンサーの前段に配置された結像光学系とを含む。イメージセンサーとしては、典型的には、CCD（Coupled Charged Device）イメージセンサーや、密着イメージセンサー（Contact Image Sensor）が用いられる。なお、スキャン対象の原稿は、原稿台35にセットされることもできる。また、原稿の表面側および裏面側にそれぞれ撮像デバイスを配置することで、両面同時スキャンを実現してもよい。

10

20

30

40

50

【 0 0 4 5 】

プリントエンジン 4 は、一例として、電子写真方式の画像形成プロセスが実行される。具体的には、フルカラーのプリント出力が可能である。具体的には、プリントエンジン 4 は、それぞれイエロー（ Y ）、マゼンタ（ M ）、シアン（ C ）、ブラック（ K ）の各色のトナー像を生成するイメージング（作像）ユニット 4 4 Y , 4 4 M , 4 4 C , 4 4 K を含む。イメージングユニット 4 4 Y , 4 4 M , 4 4 C , 4 4 K は、プリントエンジン 4 内に張架されて駆動される転写ベルト 2 7 に沿って、その順序に配置される。

【 0 0 4 6 】

イメージングユニット 4 4 Y , 4 4 M , 4 4 C , 4 4 K は、それぞれ画像書込部 4 3 Y , 4 3 M , 4 3 C , 4 3 K と、感光体ドラム 4 1 Y , 4 1 M , 4 1 C , 4 1 K とを含む。画像書込部 4 3 Y , 4 3 M , 4 3 C , 4 3 K の各々は、対象のプリントデータに含まれる各色イメージに応じたレーザ光を発するレーザダイオードと、このレーザ光を偏向して対応の感光体ドラム 4 1 Y , 4 1 M , 4 1 C , 4 1 K の表面を主走査方向に露光させるポリゴンミラーとを含んでいる。

【 0 0 4 7 】

感光体ドラム 4 1 Y , 4 1 M , 4 1 C , 4 1 K の表面には、上述のような画像書込部 4 3 Y , 4 3 M , 4 3 C , 4 3 K による露光によって静電潜像が形成され、この静電潜像がそれぞれ対応するトナーユニット 4 4 1 Y , 4 4 1 M , 4 4 1 C , 4 4 1 K から供給されるトナー粒子によってトナー像として現像される。

【 0 0 4 8 】

感光体ドラム 4 1 Y , 4 1 M , 4 1 C , 4 1 K の表面に現像された各色のトナー像は、転写ベルト 2 7 に順次転送される。さらに、この転写ベルト 2 7 上に重ねられたトナー像は、給紙部 5 からタイミングを合わせて供給される記録紙にさらに転写される。

【 0 0 4 9 】

この記録紙上に転写されたトナー像は、下流部に配置された定着部において定着された後、トレイ 5 7 に排出される。

【 0 0 5 0 】

上述の感光体ドラム 4 1 Y , 4 1 M , 4 1 C , 4 1 K における動作と並行して、記録紙を収容する給紙部 5 の給紙カセットにそれぞれ対応する送出口ローラー 5 2 , 5 3 , 5 4 および手差給紙部 2 6 のうち、画像形成に用いられるべき記録紙に対応する部位が作動して記録紙を供給する。この供給された記録紙は、搬送ローラー 5 5 および 5 6 ならびにタイミングローラー 5 1 によって搬送され、感光体ドラム 4 1 上に形成されたトナー像に同期するように、感光体ドラム 4 1 に給紙される。

【 0 0 5 1 】

転写器 4 5 は、感光体ドラム 4 1 に反対極性の電圧を印加することで、感光体ドラム 4 1 上に形成されたトナー像を記録紙に転写する。そして、除電器 4 6 は、トナー像が転写された記録紙を除電することで、記録紙を感光体ドラム 4 1 から分離させる。その後、トナー像が転写された記録紙は定着装置 4 7 へ搬送される。なお、転写器 4 5 としては、図 1 に示すような転写ベルトを用いた転写方式に代えて、転写チャージャーまたは転写ローラーを用いた転写方式を採用してもよい。あるいは、感光体ドラム 4 1 から記録紙へトナー像を直接転写する直接転写方式に代えて、感光体ドラム 4 1 と記録紙との間に、転写ローラー、転写ベルトといった中間転写体を配置して、2 段階以上のプロセスによって転写を行なうようにしてもよい。

【 0 0 5 2 】

定着装置 4 7 は、加熱ローラー 4 7 1 と加圧ローラー 4 7 2 とを含む。加熱ローラー 4 7 1 は、記録紙を加熱することで、その上に転写されたトナーを溶融するとともに、加熱ローラー 4 7 1 と加圧ローラー 4 7 2 との間の圧縮力により、溶融したトナーが記録紙上に定着される。そして、記録紙はトレイ 5 7 に排出される。なお、定着装置 4 7 としては、図 1 に示すような定着ベルトを用いた定着方式に代えて、定着ローラー等用いた定着方式、もしくは非接触の定着方式を採用してもよい。

【 0 0 5 3 】

一方、記録紙が分離された感光体ドラム 4 1 では、その残留電位が除去された後、クリーニング部によって残留トナーが除去および清掃される。そして、次の画像形成処理が行われる。クリーニング部は、一例として、クリーニングブレード、クリーニングブラシ、クリーニングローラー、またはこれらの組み合わせにより、残留トナーを除去および清掃する。あるいは、感光体ドラム 4 1 Y , 4 1 M , 4 1 C , 4 1 K を用いて残留トナーを回収するクリーナーレス方式を採用してもよい。

【 0 0 5 4 】

I D C センサー 4 9 は、感光体ドラム 4 1 上に形成されるトナー像の濃度を検知する。この I D C センサー 4 9 は、代表的に反射型フォトセンサからなる光強度センサーであり、感光体ドラム 4 1 の表面からの反射光強度を検知する。

10

【 0 0 5 5 】

制御部 1 0 0 は、画像形成装置 1 における全体の制御を司る。本実施の形態に従う画像形成装置 1 の制御部 1 0 0 には、複数の O S の同時起動（または、並列起動）を可能にする仮想化環境が実現される。複数の O S としては、最低限実行されるホスト O S と、要求などに応じて適宜実行されるゲスト O S とがある。すなわち、ホスト O S は、退避されない。

【 0 0 5 6 】

仮想化環境を実現するための処理の詳細については、後述する。

< C . 画像形成装置 1 の制御部 1 0 0 のハードウェア構成 >

20

次に、本実施の形態に従う画像形成装置 1 の制御部 1 0 0 のハードウェア構成について説明する。図 2 は、本実施の形態に従う画像形成装置 1 の制御部 1 0 0 のハードウェア構成を示すブロック図である。

【 0 0 5 7 】

図 2 を参照して、制御部 1 0 0 は、画像形成装置 1 の全体を制御する演算処理主体であり、1 または複数の演算部を有する。具体的には、制御部 1 0 0 は、主たる構成要素として、C P U (Central Processing Unit) または M P U (Micro Processing Unit) などからなるプロセッサ 1 0 2 と、D R A M などからなるメインメモリー 1 1 0 と、画像処理 A S I C (Application Specific Integrated Circuit) 1 3 0 , 1 4 0 と、画像形成装置 1 の各コンポーネントと信号またはデータを遣り取りするインターフェイス回路 1 0 5 , 1 0 6 , 1 0 7 , 1 0 8 と、二次記憶装置の一例である H D D (Hard Disk Drive) 1 5 0 とを含む。これらの構成要素は、内部バス 1 0 4 を介して、互いにデータ通信可能に構成される。

30

【 0 0 5 8 】

メインメモリー 1 1 0 の物理メモリー空間は、システムメモリー 1 1 2 と、画像メモリー 1 1 4 とに区分されて、それぞれに応じた用途で使用される。システムメモリー 1 1 2 は、プロセッサ 1 0 2 でプログラムを実行するための各種データを格納するためのワークメモリとして用いられる。画像メモリー 1 1 4 は、画像データに対する編集・加工・表示などの処理過程で一時的に作成される画像データのバッファリングに用いられる。例えば、複数の画像データを合成する際の画像バッファ、あるいは、操作パネルなどに画像データの全部または一部をプレビュー表示する際の画像バッファなどとして用いられる。

40

【 0 0 5 9 】

メインメモリー 1 1 0 と内部バス 1 0 4 との間には、メモリー制御 I C (Integrated Circuit) 1 1 8 が配置されており、プロセッサ 1 0 2 などからの要求に应答して、メインメモリー 1 1 0 へのデータ書込み、または、メインメモリー 1 1 0 からのデータ読出しを行なう。なお、メモリー制御 I C 1 1 8 には、圧縮 / 伸張 I C 1 2 0 が接続されており、メインメモリー 1 1 0 に書込まれる画像データをブロック単位で圧縮するとともに、メインメモリー 1 1 0 から読出される画像データを伸長する。このような圧縮伸長管理によって、メモリー空間の効率的な利用および処理の高速化を実現する。

【 0 0 6 0 】

50

一例として、図 2 には、原稿の両面を一度にスキャンできる構成を例示する。すなわち、制御部 100 は、原稿の表面側に配置された撮像デバイス（表面側撮像デバイス）が原稿をスキャンすることで読み取られた画像データを処理する画像処理 ASIC 130 と、原稿の裏面側に配置された撮像デバイス（裏面側撮像デバイス）が原稿をスキャンすることで読み取られた画像データを処理する画像処理 ASIC 140 とを含む。このような構成を採用することで、スキャンニーズの高まりに対して、両面同時スキャンによる生産性改善を実現できる。

【0061】

なお、画像処理 ASIC 130 および 140 に代えて、プロセッサ 102 が画像処理を実行するようにしてもよいが、ジョブをより高速に処理するためには、ASIC などの回路ロジックを採用することが好ましい。

10

【0062】

画像処理 ASIC 130 は、表面側撮像デバイスからの画像データをサブメモリー 134 に格納する。画像処理 ASIC 130 とサブメモリー 134 との間には、メモリー制御 IC 132 が配置されており、サブメモリー 134 へのデータ書込み、および、サブメモリー 134 からのデータ読出しを仲介する。サブメモリー 134 の物理メモリー空間は、表面側撮像デバイスからの画像データを格納するための表面入力画像データバッファ 136 と、画像メモリー 138 とに区分されて、それぞれに応じた用途で利用される。画像メモリー 138 は、画像データに対する編集・加工・表示などの処理過程で一時的に作成される画像データのバッファリングに用いられる。例えば、表面側撮像デバイスからの画像データ（生データ）を指定された画像形式に変換する際の画像バッファ、FAX を送信する際の画像バッファ、または、外部ネットワークあるいは外部機器へ画像データを出力する際の画像バッファなどとして用いられる。

20

【0063】

同様に、画像処理 ASIC 140 は、裏面側撮像デバイスからの画像データをサブメモリー 144 に格納する。画像処理 ASIC 140 とサブメモリー 144 との間には、メモリー制御 IC 142 が配置されており、サブメモリー 144 へのデータ書込み、および、サブメモリー 144 からのデータ読出しを仲介する。サブメモリー 144 の物理メモリー空間は、裏面側撮像デバイスからの画像データを格納するための裏面入力画像データバッファ 146 と、画像メモリー 148 とに区分されて、それぞれに応じた用途で利用される。

30

【0064】

HDD 150 には、各種プログラムが不揮発的に格納されている。具体的には、HDD 150 は、ハイパーバイザー 152 と、ホスト OS 154 と、1 または複数のゲスト OS 156 と、1 または複数のアプリケーション 158 と、制御ファームウェア 160 とを格納している。これらのプログラムは、DVD-ROM (Digital Versatile Disk Read Only Memory) などの光学記録媒体、磁気記録媒体、半導体記録媒体、光磁気記憶媒体などの非過渡的 (non-transitory) なコンピューター読取可能な記録媒体を媒介としてインストールされてもよい。あるいは、有線通信回線または無線通信回線を介して図示しないサーバー装置などから必要なプログラムがダウンロードされて、制御部 100 にインストールされてもよい。これらの各プログラムの詳細および実行タイミングなどについては、後述する。

40

【0065】

内部バス 104 には、インターフェイス回路 105, 106, 107, 108 が接続されている。

【0066】

インターフェイス回路 105 は、FAX を送受信する FAX 通信部、および / または、ネットワークを介して他の装置との間でデータをやり取りするネットワーク通信と接続される。プロセッサ 102 からの指令などを受けて、通信部から外部装置への送信処理、および、通信部での外部装置からのデータの受信処理などを仲介する。なお、ゲスト OS

50

156 が提供する機能を外部端末で利用する機能を実装してもよい。

【0067】

インターフェイス回路106は、操作パネルに各種情報を表示する処理、および、操作パネルに対するユーザー操作を受け付ける処理などを仲介する。操作パネルは、ユーザーによるタッチ操作を受け付ける表示操作部であり、典型的には、表示部であるLCD（Liquid Crystal Display：液晶表示部）と、タッチ操作を受け付けるタッチパネルとを含む。なお、表示部には、ホストOS154およびゲストOS156の動作状況を可視化する機能を実装してもよい。例えば、表示部上に、空きメモリー空間の量およびプロセッサの利用時間などをグラフまたは数値で表示するようにしてもよい。すなわち、画像形成装置1は、ゲストOS156上の情報を表示する表示部を有していてもよい。

10

【0068】

インターフェイス回路107は、各種の認証デバイスとのデータのやり取りを仲介する。認証デバイスとしては、ICカードリーダー、指紋読み取り装置、近距離無線通信デバイスなどを含む。

【0069】

インターフェイス回路108は、プロセッサ102とプリントエンジン4との間のデータのやり取りを仲介する。

【0070】

図2に示す制御部100においては、メインメモリー110内の画像メモリー114と、サブメモリー134内の画像メモリー138と、サブメモリー144内の画像メモリー148との、3つの画像メモリーが用意されている。

20

【0071】

画像メモリーは、画像形成装置1の本体での画像処理において利用するメモリーである。このような画像メモリーは、画像形成装置1において必須の構成である。図2に示すように、画像メモリーは、メインメモリー110上に専用の画像処理領域として確保されてもよいし、画像処理ASIC130、140ごとに設けられるローカルメモリー（サブメモリー134、144）上に専用の画像処理領域として確保されてもよい。上述したように、表面用の画像処理ASIC130に加えて、裏面の画像処理ASIC140にもローカルメモリー（サブメモリー144）を配置し、このローカルメモリー上に画像メモリーを設けることで、高速スキャンを実現する。

30

【0072】

このように、画像形成装置1のメモリー構成の設計としては、それぞれの目的に応じて専用の領域が確保されることも多い。

【0073】

また、プリント処理（画像形成装置）において、例えば、画像形成装置1を外部のプリントサーバーと接続する場合にも、図2に示すような画像処理ASICおよびサブメモリー（画像メモリー）を追加することもある。このように、画像形成装置1の機能を拡張する場合にも、より多くの画像メモリーが追加されることもあり、これらの画像メモリーが同時に利用されることでマルチ動作を実現している。

【0074】

40

< D．画像形成装置1の制御部100での仮想化環境 >

次に、本実施の形態に従う画像形成装置1の制御部100での仮想化環境について説明する。図3は、本実施の形態に従う画像形成装置1の制御部100でのプログラムの実行状態の一例を示す模式図である。

【0075】

図3には、ホストOS154に加えて、3つのゲストOS156（ゲストOS1～OS3）が同時実行されている状態を示す。図3を参照して、これらの複数のOSは、共通のハードウェアリソース（すなわち、プロセッサ102およびシステムメモリー112）を用いて実行される。図3には、プロセッサ102がシステムメモリー112にハイパーバイザー152を展開して実行することで、複数のOSが同時実行されている状態を示

50

す。すなわち、それぞれのOSを動作させるためのデータを格納するメモリ空間は、システムメモリ112内に用意される。システムメモリ112のメモリ空間の利用制御（メモリ領域の割り当てなど）は、ハイパーバイザー152によって管理される。

【0076】

ハイパーバイザー152は、プロセッサ102およびシステムメモリ112を含むリソースを用いて、ホストOS154および少なくとも1つのゲストOS156を含む複数のOSの同時実行を制御する。ハイパーバイザー152は、各OSの実行に必要な仮想化されたプロセッサリソースの割り当て処理、割り込み処理、メモリ制御、デバイスドライバの提供などを行なう。このような処理を提供することで、複数のゲストOS156の切り替え実行を可能にする。ハイパーバイザー152は、画像形成装置1および画像形成装置1に接続されたデバイスをゲストOS156が利用するための適切な初期化を行なう機能を有している。このような機能によって、ゲストOS156に切り替えられた直後であっても、ホストOS154と同様の処理を実行することができる。

10

【0077】

なお、ハイパーバイザー152の機能として、メモリ空間のマネージメントに加えて、電力のマネージメントを行なってもよい。例えば、ハイパーバイザー152は、画像形成装置1が省電力モードのときに、ゲストOS156の動作を制限する機能を実装してもよい。

【0078】

ホストOS154は、制御部100において常時実行されているOSである。ホストOS154上では、制御ファームウェア160が実行される。ホストOS154としては、Linux（登録商標）などの仮想メモリ機構をもつOSが用いられる。但し、ホストOS154としては、汎用的なOSではなく、後述するような、協調モジュール170による介入制御を受け付けるように、画像形成装置1に独自の機能が拡張されている。

20

【0079】

制御ファームウェア160は、ホストOS154上で実行され、画像形成装置1の主たる制御を司る。より具体的には、制御ファームウェア160は、画像形成装置1の主たる処理（本体動作）の制御、画像メモリを扱う画像処理ASIC130、140の制御を主体としたジョブの処理、各種のデバイス制御などを担当する。

【0080】

ゲストOS156は、仮想メモリ機能をもつOSであれば、ホストOS154と同一種類のOSであってもよいし、異なる種類のOSであってもよい。例えば、ゲストOS156としては、LinuxおよびWindows（登録商標）などを用いることができる。

30

【0081】

ゲストOS156上では、各種のアプリケーション158が実行される。例えば、アプリケーション158としては、ユーザー認証を行なうアプリケーションを含み得る。さらに、ゲストOS156上に画像形成装置1のユーザーごとのデスクトップ環境を実現し、ジョブ内容を管理する機能を実装してもよい。なお、ゲストOS156に任意のプログラムを提供するアプリストアに類似した機構を採用してもよい。

40

【0082】

さらに、ハイパーバイザー152の機能の一部、または、ハイパーバイザー152と連携する独立した機能として、協調モジュール170および空きメモリ管理モジュール172が実行される。

【0083】

協調モジュール170は、ハイパーバイザー152と同一レベルで動作し、制御ファームウェア160とハイパーバイザー152との間を仲介し、同時実行中のゲストOS156との間でメモリ空間を協調して利用する。「同一レベル」とは、実行の優先度などが同じレベルで実行される状態を意味する。本実施の形態においては、協調モジュール170は、処理対象のジョブの動作を考慮して、画像メモリを効率的に利用することで、ゲ

50

ストOS 156の切り替えに対して先行読出し、および、メモリーの内容の退避などを実行する。なお、協調モジュール170は、いずれかのゲストOS 156上のユーザー管理情報を必要に応じて退避前に制御ファームウェア160と共有する機能を有していてもよい。

【0084】

空きメモリー管理モジュール172は、システムメモリー112のうちホストOS 154およびホストOS 154と同時に実行されるゲストOS 156が利用していない空きメモリー空間を管理する。すなわち、空きメモリー管理モジュール172は、ハイパーバイザー152が管理する物理メモリー空間で利用されていない空きメモリー空間を管理する。このとき、空きメモリー管理モジュール172は、協調モジュール170と連携動作することで、画像メモリー空間を含めた空きメモリー空間を管理する。協調モジュール170を管理するためのAPI (Application Programming Interface)を実装することで、モジュール間の連携性を高めるようにしてもよい。

10

【0085】

さらに、退避メモリー管理モジュール174をさらに実行させるようにしてもよい。退避メモリー管理モジュール174は、システムメモリーまたは画像メモリーの格納されているデータのうちゲストOS以外のデータを二次記憶装置の一例であるHDD 150内の一時退避メモリー（スワップメモリー）に退避する処理を実施する。あるいは、退避メモリー管理モジュール174は、その逆に、一時退避メモリー（スワップメモリー）に退避したデータを、システムメモリーまたは画像メモリーに戻す処理も実施する。

20

【0086】

なお、ホストOS 154およびゲストOS 156について、連続するメモリー空間を優先して画像メモリーに退避することが好ましい。

【0087】

あるゲストOS 156上のプログラムを他のゲストOS 156上で実行させるように、マイグレーション処理を適切に行なう機能を実装してもよい。この場合には、あるゲストOS 156に何らかの不具合が生じて、別のゲストOS 156がアプリケーションの処理を継続できる。すなわち、協調モジュール170は、ゲストOS 156を外部システムへマイグレーションしたときに退避したデータ内容を移動管理する機能をさらに含む。

30

【0088】

さらに、画像形成装置1を管理するサーバー型または常駐型のプログラムが、ゲストOS 156の構成を管理し、複数の画像形成装置1および画像形成装置群の全体として、最適な構成になるように実行するゲストOS 156および各ゲストOS 156上で実行するアプリケーションを選択するようにしてもよい。

【0089】

ところで、図2に示すように、画像形成装置1の制御部100には、複数の画像メモリーが用意されている場合が多い。一方で、画像形成装置1において、プリント動作（画像形成動作）のみが連続する場合、または、原稿の片面のみをスキャンする場合などのように、一度にすべての画像メモリーが利用されていない状態も多くある。

40

【0090】

また、一般的には、画像メモリーは、ブロック単位で圧縮伸長管理されている。さらに、画像メモリーを単位分割して共用利用することで、複数タスクが物理メモリーを効率よく利用できるような構成も採用されている。

【0091】

一方で、本実施の形態に従う画像形成装置1のように、複数のゲストOSを搭載して多機能化されることもあるが、ゲストOSのメモリー消費量が大きくなる場合がある。また、複数のゲストOSを稼働させるマルチOSの実現が要求されているが、一般的な仮想環境の実現の場合と同様の課題がある。そのため、画像形成装置の限られたリソース内で、複数のOSの切り替えについての応答性能に低下させず、マルチOSを実現することは容易ではない。

50

【 0 0 9 2 】

例えば、上述の特許文献 1 は、単一システム内に共存させた複数のオペレーティングシステムを高速に切り替えて使用できる環境を開示する。この特許文献 1 に記載の発明では、システムメモリー上に OS 管理外の独立したメモリー空間を事前に確保し、ゲスト OS の動作コンテキストの保存に利用する。但し、このような構成を採用したとしても、独立したメモリー空間をシステムメモリー上に確保することの制限があり、また、メモリー空間が不足した場合には二次記憶装置を併用せざるを得ず、遅延が生じるため、複数の OS を高速に切り替え実行することが難しい場合もある。

【 0 0 9 3 】

また、上述の特許文献 3 は、第 1 OS ユーザー名を第 2 OS ユーザー名として用いてロ
10 ゲイン手順を簡略化することで、第 2 OS ロゲイン手順を高速化している。しかしながら、画像処理ジョブが発生しメモリーなどリソースが枯渇する場合において、本体制御の第 1 OS へリソースを明け渡すために第 2 OS のリソース解放のためのログアウト処理時間が発生し、画像処理実行開始が遅れ得る。特に、リソース確認詳細が開示されておらず画像メモリーについて言及がないため、リソース不足時のフロー手順から第 2 OS の終了によりリソース解放すると終了手順に大きく時間を取られる。その結果、第 1 OS ジョブ実行開始への遅延が大きくなりジョブ生産性への影響が大きい。

【 0 0 9 4 】

そこで、本実施の形態に従う画像形成装置 1 は、ジョブの処理に影響を与えないように、システムメモリーに加えて、画像メモリーのメモリー空間を効率的に利用することで、
20 複数 OS を高速に切り替えて実行する環境を実現する。

【 0 0 9 5 】

< E . 画像形成装置 1 でのジョブの処理例 1 >

次に、本実施の形態に従う画像形成装置 1 でのジョブの処理例 1 について説明する。処理例 1 としては、単一のジョブとして親展 F A X を処理する際に、ゲスト OS 1 5 6 にて認証処理が必要になった場合のホスト OS 1 5 4 およびゲスト OS 1 5 6 に対するメモリー空間の割り当て処理について説明する。ここで、親展 F A X とは、登録番号またはパスワードなどによる認証を介して、特定の相手に限って F A X を送り取りする機能である。

【 0 0 9 6 】

画像形成装置 1 の本体動作の状態が、ジョブ開始前、スキャン動作、親展モードでの F
A X 受信（プリント出力時に認証が必要）、コピープリントのような順番で連続した例の
30 場合には、下表に示すように、協調モジュール 1 7 0 による 2 つのゲスト OS 1 5 6 の先行読出しが実行される。

【 0 0 9 7 】

図 4 は、本実施の形態に従う画像形成装置 1 の制御部 1 0 0 による処理例 1 での処理手順を示すシーケンス図である。図 4 には、シングルジョブの例のシーケンス図を示す。なお、図 4 には、読出されたゲスト OS 2 が破壊された場合の処理も含む。

【 0 0 9 8 】

図 4 を参照して、制御ファームウェア 1 6 0 およびハイパーバイザー 1 5 2 の起動が完了した後、本体動作を司る制御ファームウェア 1 6 0 がハイパーバイザー 1 5 2 に対して
40 、ジョブ開始前であることを通知する（シーケンス S Q 1 0 0 : 「 1 . ジョブ開始前」）。この通知に応答して、ハイパーバイザー 1 5 2 がゲスト OS 1 および OS 2 を読出す（シーケンス S Q 1 0 2 : 「 2 . ゲスト OS 読出し」）。すなわち、必要なゲスト OS の起動が指示される。

【 0 0 9 9 】

図 4 に示す例では、ゲスト OS 1 が起動されると（シーケンス S Q 1 0 4 ）、ゲスト OS 1 は、出力メモリーの F A X 送信に割り当てられている領域にあるデータを読出す（シーケンス S Q 1 0 6 : 「 3 . 読出し」）。ゲスト OS 1 では、認証アプリケーションが実行されるものとする。

【 0 1 0 0 】

10

20

30

40

50

また、ゲストOS 2 が起動されると（シーケンスSQ 1 0 8）、ゲストOS 2 は、出力メモリのスキャン出力に割り当てられている領域にあるデータを読み出す（シーケンスSQ 1 1 0：「4．読み出し」）。ゲストOS 2 では、画像変換アプリケーションが実行されるものとする。

【0 1 0 1】

制御ファームウェア 1 6 0 がジョブ開始前の認証処理を実行するにあたって、既に読み出し済みのゲストOS 1 へ切り替え、ゲストOS 1 に対して、ゲストOS 1 で実行される認証アプリケーションでの認証処理を指示する（シーケンスSQ 1 1 2：「5．認証実施」）。

【0 1 0 2】

また、制御ファームウェア 1 6 0 が画像変換を伴うスキャン動作を実行するにあたって、スキャンによって取得された画像データを入力メモリのスキャン入力に割り当てられている領域に画像データを書込む（シーケンスSQ 1 1 4：「6．スキャン」）。すると、書込まれた画像データは入力メモリの画像変換に割り当てられている領域に書込まれる（シーケンスSQ 1 1 6：「6．1．画像変換」）。

【0 1 0 3】

さらに、制御ファームウェア 1 6 0 は、既に読み出し済みのゲストOS 2 へ切り替え、ゲストOS 2 に対して、ゲストOS 2 で実行される画像変換アプリケーションでの画像変換処理を指示する（シーケンスSQ 1 1 8：「7．画像変換」）。ゲストOS 2 は、画像変換処理によって得られた画像データ（変換結果）を入力メモリのアプリ入力に割り当てられている領域に画像データを書込む（シーケンスSQ 1 2 0：「8．変換出力」）。さらに、書込まれた画像データは出力メモリのスキャン出力に割り当てられている領域に書込まれる（シーケンスSQ 1 2 2：「9．アプリ出力」）。

【0 1 0 4】

ここで、ゲストOS 2 での処理後に、出力メモリのスキャン出力に割り当てられている領域が別処理に占有されて、ゲストOS 2 が利用可能なメモリ領域を確保できなくなると、ゲストOS 2 での画像変換処理が失敗し得る。すなわち、ゲストOS 2 が破壊され得る。

【0 1 0 5】

例えば、制御ファームウェア 1 6 0 が画像変換を伴うスキャン動作を続けて実行すると、スキャンによって取得された画像データを入力メモリのスキャン入力に割り当てられている領域に画像データを書込む（シーケンスSQ 1 2 4：「10．スキャン」）。すると、書込まれた画像データは入力メモリの画像変換に割り当てられている領域に書込まれる（シーケンスSQ 1 2 6：「11．画像変換」）。さらに、制御ファームウェア 1 6 0 は、ゲストOS 2 で実行される画像変換アプリケーションでの画像変換処理を指示する（シーケンスSQ 1 2 8：「11．1．画像変換」）。ここで、例えば、連続して親展FAXを受信しているとすると、画像変換処理によって得られた画像データ（変換結果）を出力メモリのスキャン出力に割り当てられている領域に書込むことができなくなり、ゲストOS 2 での画像変換処理は失敗し得る。すなわち、ゲストOS 2 での読み出しができないため、次のジョブである画像変換を伴うスキャン動作に高速で応答することができず、失敗したと判定され得る。

【0 1 0 6】

すると、制御ファームウェア 1 6 0 は、ゲストOS 2 での失敗をハイパーバイザー 1 5 2 にエラー報告する（シーケンスSQ 1 3 0：「12．ゲストOSエラー」）。一方で、制御ファームウェア 1 6 0 は、それに続くジョブを処理する。この親展FAXのジョブ完了後に、ゲストOS 2 が読み出された時点で、先の画像変換を伴うスキャン動作を再開することも可能である。

【0 1 0 7】

図 4 に示す例では、制御ファームウェア 1 6 0 は、受信したFAXの画像データを入力メモリのFAX受信に割り当てられている領域に画像データを書込む（シーケンスSQ

10

20

30

40

50

132:「13.親展FAX」)。そして、書込まれた画像データは、スキャン印刷の処理の後、出力メモリのエンジン出力に割り当てられている領域に書込まれる(シーケンスSQ140:「16.プリント出力」)。

【0108】

さらに、制御ファームウェア160は、スキャンによって取得された画像データを入力メモリのスキャン入力に割り当てられている領域に画像データを書込む(シーケンスSQ134:「14.スキャン」)。すると、書込まれた画像データは出力メモリのエンジン出力に割り当てられている領域に書込まれる(シーケンスSQ136:「14.1.プリント出力」)。

【0109】

その後、制御ファームウェア160は、既に読み出し済みのゲストOS1へ切り替え、ゲストOS1に対して、ゲストOS1で実行される認証アプリケーションでの認証処理を指示する(シーケンスSQ138:「15.認証実施」)。

【0110】

これらの先のジョブについての処理が終了した後、制御ファームウェア160がハイパーバイザー152に対して、ジョブ開始前であることを通知する(シーケンスSQ142:「17.ジョブ開始前」)。この通知に応答して、ハイパーバイザー152がゲストOS2を読み出す(シーケンスSQ144:「18.ゲストOS読み出し」)。すなわち、必要なゲストOS2の起動が指示される。

【0111】

ゲストOS2が起動されると(シーケンスSQ146)、ゲストOS2は、出力メモリのスキャン出力に割り当てられている領域にあるデータを読み出す(シーケンスSQ148:「19.読み出し」)。

【0112】

ゲストOS2での認証処理の実行後、先のジョブである画像変換が完了し、ゲストOS2に対して、画像変換が成功したことが通知される(シーケンスSQ150:「11.2:画像変換:成功」)。

【0113】

さらに、ゲストOS2は、状態変化を示す情報を出力し(シーケンスSQ152:「20:変化出力」)、その情報は、出力メモリのスキャン出力に割り当てられている領域に書込まれる(シーケンスSQ144:「20.1:アプリ出力」)。

【0114】

図4に示すように、先に親展FAXを受信した後にさらに親展FAXを受信すると、読み出し済みのゲストOS1へ切り替えて認証アプリケーションを利用することで、当該後続の受信FAXについても滞りなくプリント出力が実施できる。そのため、全体としてのジョブ終了が早くなり、ゲストOS2の読み出しまでに要する時間が短縮され、読み出し制御のない場合に比べて、連続ジョブの生産性も向上できる。

【0115】

上述したシーケンス図に対応させた、画像形成装置1がシングルジョブを処理する際に利用されるリソースを示す対応表を示す。表中において、「○」はジョブが利用するメモリーを示す。

【0116】

10

20

30

40

【表 1】

ジョブ状態 認証有無	ジョブ	入力メモリ				出力メモリ				ゲスト OS1 (認証処理)	ゲスト OS2 (画像処理)
		スキャン 入力	アプリ 入力	FAX 受信	画像 変換	スキャン 出力	エンジン 出力	FAX 送信	画像 合成	読出状態	読出状態
ジョブ開始前	認証実施									FAX 送信	スキャン出力
認証済	スキャン	○			○	○				FAX 送信	スキャン出力
認証済	スキャン 失敗	○	○		○					FAX 送信	×
親展	FAX			○						FAX 送信	×
認証なし	コピー	○					○			FAX 送信	×
	親展 プリント			○			○			FAX 送信	×
認証済	スキャン		○		○	○				FAX 送信	FAX 受信

10

20

30

40

【 0 1 1 7 】

上表では、説明上の便宜上、処理を簡略化して画像メモリーを固定的に利用することを前提としている。但し、後述するように、画像メモリーを分割して管理することで、OSの優先度などに応じて、各画像メモリーに分散された混在のメモリー利用が可能となる。例えば、上述の非特許文献 1 によれば、一定長の固定領域確保に比べて、複数の OS 状態に依存して、より多くの空きメモリーが期待できるため、ゲスト OS 数を増加できるとい

50

う利点がある。

【 0 1 1 8 】

なお、複数の認証デバイスを用いる場合には、例えば、以下の表に示すようにそれぞれ異なるゲスト OS 1 5 6 にその処理を実行させるようにしてもよい。

【 0 1 1 9 】

【表 2】

認証デバイス	ゲスト OS
カード認証	ゲスト OS1
生体認証	ゲスト OS2
リモートパネル認証	ゲスト OS3

10

【 0 1 2 0 】

図 5 は、本実施の形態に従う画像形成装置 1 の制御部 1 0 0 の初期処理の手順を示すフローチャートである。図 5 に示す各ステップは、主として、画像形成装置 1 の制御部 1 0 0 のプロセッサ 1 0 2 が図示しない各種プログラムを指定された順序で実行することで実現される。

20

【 0 1 2 1 】

図 5 を参照して、プロセッサ 1 0 2 は、システムが起動（ブート）されると（ステップ S 1 0 0 ）、ハイパーバイザー 1 5 2 および協調モジュール 1 7 0 を起動する（ステップ S 1 0 2 ）。さらに、プロセッサ 1 0 2 は、制御ファームウェア 1 6 0 を起動する（ステップ S 1 0 4 ）。

【 0 1 2 2 】

必要なプログラムの起動が開始されると、プロセッサ 1 0 2 （協調モジュール 1 7 0 ）は、システムメモリーのうち利用可能なメモリー空間を取得する（ステップ S 1 0 6 ）とともに、制御ファームウェア 1 6 0 と連携して画像メモリーのうち利用可能なメモリー空間を取得する（ステップ S 1 0 8 ）。

30

【 0 1 2 3 】

続いて、プロセッサ 1 0 2 （協調モジュール 1 7 0 ）は、処理すべきジョブの内容と、当該ジョブに必要な 1 つ以上のゲスト OS で動作するプログラムを関連付ける（ステップ S 1 1 0 ）。例えば、上述したような親展 F A X については、ゲスト OS 上で実行される認証アプリケーションが必要であるといった情報が登録される。

【 0 1 2 4 】

続いて、プロセッサ 1 0 2 （協調モジュール 1 7 0 ）は、処理すべきジョブの開始前に、ステップ S 1 1 0 において、当該処理すべきジョブにいずれかのプログラムが必要であることが登録されていれば、ジョブの開始前に必要なゲスト OS を画像メモリーのうち利用可能なメモリー空間に読み出し、関連付けられたプログラムを実行させる（ステップ S 1 1 2 ）。

40

【 0 1 2 5 】

プロセッサ 1 0 2 （協調モジュール 1 7 0 ）は、システムメモリー 1 1 2 、画像メモリー、HDD 1 5 0 のスワップ領域との間で、データ退避が必要であるか否かを判断する（ステップ S 1 1 4 ）。データ退避が必要であると判断されると（ステップ S 1 1 4 において Y E S の場合）、プロセッサ 1 0 2 （協調モジュール 1 7 0 ）は、ホスト OS 1 5 4 および他のゲスト OS 1 5 6 が利用していないと判断されたシステムメモリー 1 1 2 の内容を画像メモリーに退避する（ステップ S 1 1 6 ）。これにより、新たに実行されたゲ

50

ストＯＳのシステムメモリーとして利用可能にする。

【０１２６】

その後、プロセッサ１０２は、ジョブの受け付けを開始する（ステップＳ１１８）。その後、ジョブが入力されると、入力されたジョブに応じた処理が実行される。なお、ホストＯＳ１５４上の要求だけでなく、ゲストＯＳ１５６上の要求を本体ジョブとして受け付けるようにしてもよい。

【０１２７】

以上のとおり、例えば、ジョブの処理と並行して、認証処理が必要な場合には、ハイパーバイザー１５２がゲストＯＳ１５６を先行読出しすることで、より高速なジョブの処理を実現できる。

【０１２８】

< F . 画像形成装置１でのジョブの処理例２ >

次に、画像メモリー特有のデータ構造を利用したメモリーマネージメントについて説明する。特に、ブロック分割された画像メモリーの利用に適した処理について説明する。

【０１２９】

図６は、本実施の形態に従う画像形成装置１の制御部１００でのプログラムの別の実行状態の一例を示す模式図である。図６を参照して、制御部１００のハードウェアを利用してハイパーバイザー１５２が実行され、ハイパーバイザー１５２の管理下で、ホストＯＳ１５４および複数のゲストＯＳ１５６が実行されている例を示す。

【０１３０】

制御部１００のハードウェアとしては、メモリー制御ＩＣ１１８および圧縮／伸張ＩＣ１２０（図２）により提供される、ＭＭＵ（メモリー管理部：Memory Management Unit）およびＩＣの機能と、プロセッサ１０２と、メインメモリー１１０とを含む。さらに、制御部１００のハードウェアとしては、サブメモリー１３４，１４４に加えて、認証デバイス１９２と、ＮＩＣ（Network Interface Card）デバイス１９４と、各種の記憶デバイス１９６を含む。

【０１３１】

ハイパーバイザー１５２は、仮想的なメモリー管理部（ＶＭＭＵ）１５２１と、圧縮／伸張ＩＣ（ＶＩＣ）１５２２と、仮想的な演算リソース（ＶＣＰＵ）１５２３とを複数のＯＳが利用可能な状態で提供する。ホストＯＳ１５４上では、プリントアプリケーション１６０１と、スキャンアプリケーション１６０２と、ＦＡＸアプリケーション１６０３と、ＢＯＸアプリケーション１６０４とが実行される。また、ゲストＯＳ１～ＯＳ３上では、認証アプリケーション１５８１，１５８２，１５８３がそれぞれ実行されるものとする。ハイパーバイザー１５２は、スタブ１５８４，１５８５，１５８６を用いてそれぞれゲストＯＳおよびアプリケーションの実行を管理する。

【０１３２】

画像メモリーは、ブロック単位で圧縮伸長管理されている。さらに、画像メモリーを単位分割して共用利用することで、複数タスクが物理メモリーを効率よく利用できるように構成が採用されているとする。

【０１３３】

ホストＯＳ１５４およびゲストＯＳ１５６についても、仮想メモリー機構を用いて一定のページ単位でメモリー空間を管理している。なお、以下の説明では、仮想メモリー機構をもたないＯＳについては考慮していない。

【０１３４】

しかしながら、画像メモリーで管理されるブロック単位と、仮想メモリー機構を用いて管理されるページ単位とは、単位サイズが同一とは限らない。また、管理対象の画像メモリーがメインメモリー１１０に設けられる場合と、サブメモリー１３４，１４４に設けられる場合とでは、その取扱いも同一にはできない。

【０１３５】

そこで、画像形成装置１での管理単位および制御を前提として、ホストＯＳ１５４およ

10

20

30

40

50

びゲストOS 156、ならびに、ハイパーバイザー 152でのメモリー共有化機構が必要になる。

【0136】

本実施の形態に従う画像形成装置 1 には、画像処理 ASIC 130, 140 が組み込まれている。但し、プリント制御に利用される画像メモリーは、プロセッサ 102 と共有するメインメモリー 100 内の画像メモリー 114、ならびに、画像処理 ASIC 130, 140 に関連付けられる専用のサブメモリー 134, 144 内の画像メモリー 138, 148 の 2 種類から構成される。このように、画像メモリーを利用するジョブに応じた利用形態となるため、メモリー領域の確保をジョブ動作に応じて上述のような単位サイズで実施する。

10

【0137】

図 7 は、本実施の形態に従う画像形成装置 1 の制御部 100 におけるブロック分割された画像メモリーの利用形態を説明するための模式図である。図 7 を参照して、物理的な画像メモリーは、画像ブロック単位で圧縮伸長管理される。そして、一定サイズの単位メモリーは、ツリー構造で管理される。より具体的には、単位メモリーの各々は、リンクトリストを用いたツリー状のデータ構造を用いて管理され、各単位メモリーの種別およびアドレスが拡張アドレス・データ構造に格納されている。

【0138】

図 7 に示すような構成において、協調モジュール 170 は、画像処理 ASIC と協調動作するように構成されることが好ましい。このような構成は、ホストOS 154 およびゲストOS 156 ならびにハイパーバイザー 152 の仮想メモリー機構との親和性は高い。一方で、このような構成は、ジョブ制御に依存した従来の画像処理を前提に構成されているため、他の目的、特にOS が扱えるメモリー空間としての利用については考慮されていない。

20

【0139】

そこで、本実施の形態に従う画像形成装置 1 では、画像形成の管理単位に対して、ホストOS 154 およびゲストOS 156 ならびにハイパーバイザー 152 でのページ単位構造を割り当て、通常のメモリー空間として管理することで、ゲストOS 156 でも画像メモリーを利用することを可能にする。

【0140】

すなわち、仮想メモリーとしては、一定サイズのページメモリーがツリー構造で管理され、いずれのゲストOS も利用可能になっている。画像メモリーは、画像メモリーの種別（すなわち、メインメモリー 110 内に設けられた場合と、サブメモリー 134, 144 に設けられた場合）により、メモリー応答性能が異なる。そのため、単純なメモリー空間として利用すると、メモリー応答性能を考慮せずアクセスされるので、全体としての性能が劣化する可能性がある。

30

【0141】

そこで、本実施の形態に従う画像形成装置 1 では、本体OS（ホストOS 154 およびゲストOS 156 とハイパーバイザー 152 との間に、協調モジュール 170 を介在させ、画像メモリーの種別に応じて、通常のメモリー空間、および、ストレージメモリー空間のいずれかとして選択的に割り当てる。

40

【0142】

例えば、ストレージメモリー空間は、ページ退避可能性が高く（Swappableに）設定され、ジョブ動作に応じてスワップされるよう、協調モジュール 170 によってメモリー管理が調停される。また、一旦ページ化した場合、再度ストレージメモリー空間へレストアされるのではなく、通常スワップと同様に扱うことで、高速なメモリー復帰が可能になる。

【0143】

図 7 には、ハイパーバイザー 152 と連携する協調モジュール 170 が、メインメモリー 110 をページ退避可能性が低いと（Residentialに）設定し、サブメモリー 134,

50

144をページ退避可能性が高いと(Swappableに)設定してもよい。また、サブメモリー134, 144上の退避メモリーはHDD150のスワップ領域にスワップアウトされてもよい。

【0144】

これらのメモリー管理には、ゲストOS156を特定するIDをメモリー空間に配置されるデータに付与することで、読出し済みのゲストOS156がジョブ実行により破壊されたことを協調モジュール170が把握する機能においても利用できるようにしてもよい。すなわち、協調モジュール170は、読出し済みのゲストOS156がジョブ実行により破壊されたことを検知し、ジョブ開始前に検知された破壊されたゲストOS156を再度読出す。

【0145】

図8は、本実施の形態に従う画像形成装置1の制御部100におけるブロック分割された画像メモリーの管理方法を説明するための模式図である。図8を参照して、あるメモリー空間内の画像ブロック単位(例えば、4kByte)の各々に対して、属性情報を含むテーブル(カラム)が関連付けられる。

【0146】

このテーブルには、複数の属性情報が格納される。図8に示す例では、ブロック番号、アドレス、メモリーID、OSID、JOB情報、C、Uなどである。

【0147】

ブロック番号は、メモリー空間上の領域を特定する情報である。アドレスは、メモリー空間上の対応するブロックのアドレスである。メモリーIDは、システムメモリー112上に設けられる画像メモリーであるのか、サブメモリー134, 144上に設けられる画像メモリーであるのかを示す。OSIDは、対応するOSのIDを示す。JOB情報は、ゲストOSに対応する認証JOBであるか否かを示す。Cは、対応ブロックが参照された回数を示し、Uは、対応ブロックが更新されたか否かを示す。

【0148】

このように、協調モジュール170は、メモリー空間の種別に区分して、単位メモリーごとに利用しているOSを特定可能なIDを含むリストを管理する機能を有している。このようなブロック単位ごとの情報に基づいて、スワップアウトまたは書き戻しなどが制御される。

【0149】

図9は、本実施の形態に従う画像形成装置1の制御部100における複数のメモリー間でのデータ管理の一例を示す模式図である。図9を参照して、ハイパーバイザーは、要求などに応じて、HDD150からゲストOSを、メインメモリー110内のシステムメモリー112上に読み出して起動する((1)起動)。その後、状況に応じて、メインメモリー110内の画像メモリー114に退避する((2)退避)。なお、メインメモリー110内の画像メモリー114に十分な空きメモリー空間が存在しなければ、いずれかの画像処理ASICに関連付けられるサブメモリー134, 144内の画像メモリー138, 148に読み出された内容が退避されることもある。

【0150】

その後、画像処理ASICによる処理要求などに応じて、サブメモリー134, 144に、十分な空きメモリー空間を確保するために、退避しているデータをHDD150にページアウトする。

【0151】

図9に示すように、本実施の形態に従う画像形成装置1においては、HDD150、メインメモリー110(含:システムメモリー112、画像メモリー114)、および、画像処理ASICに関連付けられたサブメモリー134, 144(画像メモリー138, 148)の三者あるいは四者の間で、メモリーの使用要求および空きメモリー空間の量などに応じて、データの退避処理、ページアウト処理などが最適化された状態で実施される。

【0152】

10

20

30

40

50

図10は、本実施の形態に従う画像形成装置1の制御部100におけるメモリー空間の利用形態の一例を示す図である。図10を参照して、例えば、システムメモリー112にホストOSならびにゲストOS1およびゲストOS2の実行に必要なデータが保存されている状態で、ゲストOS3の起動が指令されたとする((1)要求発生)。すると、協調モジュール170は、例えば、その実行が一時的に中止されるゲストOS1に割り当てていたメモリー領域をサブメモリー134に退避(スワップアウト)する。

【0153】

このとき、システムメモリー112からサブメモリー134へのデータ転送は、システムメモリーを中継して、そのバッファ容量(例えば、4kB)の単位(すなわち、ブロック単位)で平行伝送することもできる。

【0154】

その後、サブOS3の実行の必要性がなくなると、システムメモリー112の内容がHDD150にスワップアウトされ、サブメモリー134からシステムメモリー112へ書き戻される。

【0155】

このように、協調モジュール170は、必要なシステムメモリー112が利用可能になるまでブロック単位で並列にメモリー転送を実施し、ゲストOS156に対してメモリー空間を割り当てる。そして、協調モジュール170は、メモリー転送の完了後にゲストOS156への切り替えを行なう。さらにその後、協調モジュール170は、ホストOS154へ切り替える場合に、先に実行されたゲストOSおよび当該ゲストOS156で動作するプログラムが利用するシステムメモリーを一時凍結する処理、先に転送されたメモリーを順次入れ替える処理と、入れ替え完了後にホストOSへ切り替える処理とを連続動作させる。

【0156】

また、協調モジュール170は、ゲストOS156を切り替えた後に、ゲストOS156からホストOS154ではなく他のゲストOS156に切り替える場合に、ホストOS154に関するファイルメモリーのデータをHDD150(二次記憶装置)へ順次並列転送する処理、先に実行されたゲストOS156および当該ゲストOS156で動作するプログラムが利用するシステムメモリーを一時凍結する処理、HDD150へ転送されたファイルメモリーに対して一時凍結したシステムメモリーを順次並列転送する処理と、他のゲストOSが必要とするシステムメモリーになるまで先のゲストOSが転送完了したシステムメモリーから順次確保する処理と、他のゲストOSが動作可能なシステムメモリーが確保されたことで切り替えする処理とを連続動作させる。

【0157】

図11は、本実施の形態に従う画像形成装置1の制御部100による処理例2でのメモリー確保に係る処理手順を示すフローチャートである。図11に示す各ステップは、主として、画像形成装置1の制御部100のプロセッサ102が協調モジュール170のプログラムを実行することで実現される。

【0158】

図11を参照して、プロセッサ102(協調モジュール170)は、メモリー空間の種別に区分して、単位メモリーごとに利用しているOSを特定可能なIDを含むリストを更新する(ステップS200)。ホストOSからゲストOSへの切り替えが要求されたか否かを判断する(ステップS202)。

【0159】

ホストOSからゲストOSへの切り替えが要求されると(ステップS202においてYESの場合)、プロセッサ102(協調モジュール170)は、必要なシステムメモリーが利用可能になるまで、ブロック単位で並列にメモリー転送を実施する(ステップS204)。これにより、切り替えが要求されたゲストOSに対して必要なメモリー空間を割り当てることができる。メモリー転送の完了後に、プロセッサ102(協調モジュール170)は、ホストOSからゲストOSへの切り替えを行なう(ステップS206)。

【 0 1 6 0 】

その後、プロセッサ 1 0 2 (協調モジュール 1 7 0) は、ゲスト OS から別のゲスト OS への切り替えが要求されたか否かを判断する (ステップ S 2 0 8) 。

【 0 1 6 1 】

ゲスト OS から別のゲスト OS への切り替えが要求されると (ステップ S 2 0 8 において YES の場合) 、すなわち、ゲスト OS へ切り替えた後に、ゲスト OS からホスト OS ではなく他のゲスト OS に切り替える場合に、まず、プロセッサ 1 0 2 (協調モジュール 1 7 0) は、ホスト OS に関するファイルメモリのデータを HDD 1 5 0 (二次記憶装置) へ順次並列転送する (ステップ S 2 1 0) 。続いて、プロセッサ 1 0 2 (協調モジュール 1 7 0) は、先に実行されたゲスト OS および当該ゲスト OS で動作するプログラムが利用するシステムメモリーを一時凍結する (ステップ S 2 1 2) 。さらに、プロセッサ 1 0 2 (協調モジュール 1 7 0) は、HDD 1 5 0 (二次記憶装置) へ転送されたファイルメモリーに対して一時凍結したシステムメモリーを順次並列転送し (ステップ S 2 1 4) 、他のゲスト OS が必要とするシステムメモリーになるまで先のゲスト OS が転送完了したシステムメモリーから順次確保する (ステップ S 2 1 6) 。最終的に、プロセッサ 1 0 2 (協調モジュール 1 7 0) は、他のゲスト OS が動作可能なシステムメモリーが確保されたことで、ゲスト OS から指定された別のゲスト OS へ切り替える (ステップ S 2 1 8) 。そして、処理はステップ S 2 2 0 へ進む。

10

【 0 1 6 2 】

ホスト OS からゲスト OS への切り替えが要求されていなければ (ステップ S 2 0 2 において NO の場合) 、あるいは、ゲスト OS から別のゲスト OS への切り替えが要求されていなければ (ステップ S 2 0 8 において NO の場合) 、あるいは、ステップ S 2 1 8 の実行後、ゲスト OS からホスト OS への切り替えが要求されたか否かを判断する (ステップ S 2 2 0) 。

20

【 0 1 6 3 】

ゲスト OS からホスト OS への切り替えが要求されると (ステップ S 2 2 0 において YES の場合) 、プロセッサ 1 0 2 (協調モジュール 1 7 0) は、先に実行されたゲスト OS および当該ゲスト OS で動作するプログラムが利用するシステムメモリーを一時凍結する (ステップ S 2 2 2) 。そして、プロセッサ 1 0 2 (協調モジュール 1 7 0) は、先に転送されたメモリーを順次入れ替える (ステップ S 2 2 4) 。メモリーの入れ替え完了後に、プロセッサ 1 0 2 (協調モジュール 1 7 0) は、ゲスト OS からホスト OS へ切り替える (ステップ S 2 2 6) 。そして、処理はステップ S 2 0 0 へ戻る。

30

【 0 1 6 4 】

ゲスト OS からホスト OS への切り替えが要求されていなければ (ステップ S 2 2 0 において NO の場合) 、ステップ S 2 0 0 以下の処理が繰り返される。

【 0 1 6 5 】

< G . 画像メモリー領域の確保 >

上述の図 4 に示すシーケンス図においては、ゲスト OS への切り替えが発生したときに、先に処理されているジョブによって、利用可能なメモリー領域を確保できなければ、ゲスト OS での処理が失敗し得る。すなわち、ゲスト OS での処理と先に入力されたジョブの処理とのうち、先に入力されたジョブの処理が優先される設計となっているが、これを逆にしてもよい。すなわち、ゲスト OS への切り替えが発生したときに、ジョブ利用によって利用可能な画像メモリーがない場合にジョブを停止させ画像メモリーを確保するようにしてもよい。

40

【 0 1 6 6 】

さらに、先に画像メモリーを利用しているジョブと、当該ゲスト OS への切り替えの起因となったジョブとの間で、優先度を評価し、先行のジョブまたはゲスト OS への切り替えのいずれを優先させるかを判断するようにしてもよい。

【 0 1 6 7 】

このように、協調モジュール 1 7 0 は、画像メモリーの種別に応じてジョブおよびジョ

50

ブ情報に基づき優先度を設定し、当該設定した優先度に基づいてジョブの実行を制御する。

【0168】

< H . メモリー領域の取得可否の判断 >

上述したように、本実施の形態に従う画像形成装置 1 は、システムメモリー 112 に加えて、画像メモリーを用いて、複数の OS (ホスト OS 154 ならびに 1 または複数のゲスト OS 156) を、共通のハードウェア上で同時実行させる。

【0169】

そのため、複数のメモリー空間全体を考慮して、メモリー領域のマネージメントを行なうことが好ましい。具体的には、協調モジュール 170 は、ホスト OS 154 上で実行される制御ファームウェア 160、ハイパーバイザー 152、空きメモリー管理モジュール 172 (および、退避メモリー管理モジュール 174) と情報交換し、登録された空きメモリー量および退避メモリー量を適宜更新することで、各種アプリケーション (認証アプリケーションなどを含む) の動作に必要なメモリー領域を取得できるか否かを判断するようにしてもよい。

10

【0170】

さらに、協調モジュール 170 は、何らかの処理の開始または終了を都度記録する更新履歴を保持しておき、メモリー使用量または空きメモリー量の増減傾向に基づいて、格納している内容の配置場所を都度変更してもよい。例えば、空きメモリー量 (画像メモリー) に減少傾向がみられる場合に、退避している内容を先行的にページアウト (スワップアウト)、すなわち HDD 150 に書き戻すようにしてもよい。

20

【0171】

図 12 は、本実施の形態に従う画像形成装置 1 の制御部 100 でのメモリー領域のマネージメント処理の手順を示すフローチャートである。図 12 に示す各ステップは、主として、画像形成装置 1 の制御部 100 のプロセッサ 102 が協調モジュール 170 のプログラムを実行することで実現される。

【0172】

図 12 を参照して、プロセッサ 102 は、メモリー領域の要求 (ページフォルト) が発生したか否かを判断する (ステップ S300)。メモリー領域の要求が発生していなければ (ステップ S300 において NO の場合) には、ステップ S300 の処理が繰り返される。

30

【0173】

メモリー領域の要求が発生していれば (ステップ S300 において YES の場合) には、プロセッサ 102 は、空きメモリー管理モジュール 172 へ空きメモリー量を問い合わせる (ステップ S302)。そして、プロセッサ 102 は、過去の空きメモリー量の実績と、問い合わせに対して応答された空メモリー量とから、空きメモリー量の変化量を算出する (ステップ S304)。

【0174】

さらに、プロセッサ 102 は、算出した空きメモリー量の変化量が減少傾向を示しているか否かを判断する (ステップ S306)。算出した空きメモリー量の変化量が減少傾向を示していなければ (ステップ S306 において NO の場合)、ステップ S300 以下の処理が繰り返される。

40

【0175】

算出した空きメモリー量の変化量が減少傾向を示していれば (ステップ S306 において YES の場合)、プロセッサ 102 は、退避メモリー管理モジュール 174 へ退避メモリー量を問い合わせる (ステップ S308)。そして、プロセッサ 102 は、対象となる退避メモリーを特定し、特定した退避メモリーをページアウトする (ステップ S310)。

【0176】

最後に、プロセッサ 102 は、退避メモリー管理モジュール 174 が管理する退避メ

50

メモリー量を更新し（ステップ S 3 1 2）、ステップ S 3 0 0 以下の処理を繰り返す。

【0177】

このようなトータルでのメモリー領域の管理を行なうことで、オーバーコミットメントの発生頻度などを低減できる。また、制限のあるメモリーリソースの利用効率を向上させることができる。

【0178】

< I . 付記 >

本発明は、別の局面として、さらに以下のように表現することができる。

【0179】

本発明の別の局面に従う画像形成装置は、画像形成動作に必要な制御を行なう制御ファームウェアおよびホスト OS と、複数のゲスト OS を切り替えて使用するハイパーバイザーと、制御ファームウェアを含むホスト OS とハイパーバイザーの動作に必要なシステムメモリーと、制御ファームウェアにより管理される画像処理に専用に用いる画像メモリーと、空きシステムメモリーならびに制御ファームウェアおよびホスト OS が利用するシステムメモリーについて退避可否を判断するモジュールと、制御ファームウェアおよびホスト OS が利用していないシステムメモリーについて退避可否を判断するモジュールと、制御ファームウェアおよびホスト OS で実行される画像メモリー管理と連携することで利用可能な画像メモリーを把握するモジュールとが、ハイパーバイザーと同一レベルで動作される協調モジュール内に実装されている。ここで、画像メモリー上にブロック単位でのメモリー管理を実施するファイルメモリーが実装されていることが好ましい。画像形成装置は、協調モジュールにおいてジョブ内容とゲスト OS で動作するプログラムとを関連付けて登録するモジュール（例えば、認証プログラムなどゲスト OS で動作するプログラムとジョブとの関連性が把握できる機能）と、ジョブの開始前に 1 つ以上のゲスト OS を空き画像メモリーに読出すモジュールと、ジョブの動作前にジョブ内容とゲスト OS との関連付けにて判断した協調モジュールの指示により、ハイパーバイザーにより読出したゲスト OS を稼働させるモジュールと、制御ファームウェアおよびホスト OS や他ゲスト OS が利用していないと判断されたシステムメモリー、および、稼働したゲスト OS が必要とするメモリーを入れ替えてゲスト OS のシステムメモリーとして利用可能にするモジュールと、を協調モジュールで連携動作させる。

【0180】

ジョブの開始前に 1 つ以上のゲスト OS を空き画像メモリーに読出すモジュールは、ジョブリストが追加されるごとに利用可能メモリーリストの空き画像メモリーを優先度および ID によりソートし利用可否を判断するようにしてもよい。また、読出すモジュールは、既に読み込んだゲスト OS に対して優先度が低い場合に高い優先度へ移動してもよい。また、読出すモジュールは、必要に応じて、メモリーリストに含まれる制御ファームウェアおよびホスト OS が利用するメモリーを一時凍結してもよい。

【0181】

上記画像形成装置は、各画像メモリーの種別に応じて実行されるジョブおよびジョブ情報に基づいて優先度を設定するモジュールと、利用している各ゲスト OS を特定可能な ID を設定することが可能な利用可能空きメモリーリストを管理するモジュールにより連携し、（転送に必要な最小限のメモリーを 2 セット以上有するメモリー転送機能によって）上述したメモリー量となるシステムメモリーが得られるまでファイルメモリーブロック単位に合わせてメモリー転送を順次並列実施するモジュールとにより得られるシステムメモリーに対して読出されたゲスト OS を実行するメモリー空間として割り当てメモリー転送を行なうモジュールと、メモリー転送完了後にゲスト OS へ切り替えるモジュールと、次にホスト OS へ切り替える場合にゲスト OS および動作プログラムが利用するシステムメモリーを一時凍結するモジュールと、上述のメモリー転送モジュールによって先に転送したメモリーと順次入れ替え並列転送するモジュールと、順次転送するなかで保存された元のメモリアドレスに割り当てするモジュールと、入れ替え転送完了後にホスト OS へ切り替えるモジュールとが、連続動作することを可能とするモジュールと、上述した読出し済み

のゲストOSがジョブの実行により破壊されたことを協調モジュールが把握するモジュールと、ジョブ開始前に上記読出し手段で破壊されたゲストOSを上述した手順に従って再度読出し実施すること。

【0182】

上記画像形成装置は、切り替えが発生したときに、ジョブ利用によって利用可能な画像メモリーがない場合にジョブを停止させ画像メモリーを確保する。

【0183】

上記画像形成装置は、ゲストOSへ切り替えた後に、ゲストOSからホストOSではなく他のゲストOSに切り替えるモジュールと、ファイルメモリーのホストOSに関するデータを二次記憶装置へ順次並列転送するモジュールと、ゲストOSおよび動作プログラムが利用するメモリーを一時凍結するモジュールと、二次記憶装置へ順次転送されたファイルメモリーに対して一時凍結したメモリーを順次並列転送するモジュールと、他のゲストOSが必要とするシステムメモリーになるまで先のゲストOSが転送完了したシステムメモリーから順次確保する手段と、他のゲストOSが動作可能なシステムメモリーが確保されたことで切り替えする手段とを、連続動作することを可能にするモジュールを含む。

【0184】

上記画像形成装置は、ホストOS・ゲストOSともに連続するメモリー空間を優先して画像メモリーに退避する。

【0185】

上記画像形成装置は、画像形成装置の本体以外のシステムへのゲストOSマイグレーション時に退避した画像メモリーを適切に移動管理する。

【0186】

上記画像形成装置は、画像ホストOSは退避しないことを有する。すなわち、本体動作を優先する。

【0187】

上記画像形成装置は、ゲストOS上のプログラムを他のゲストOSへマイグレーションを適切に行なう。

【0188】

上記画像形成装置は、画像形成装置および画像形成装置に接続されたデバイスをゲストOSが利用するための適切な初期化を行なう。

【0189】

上記画像形成装置は、画像形成装置を管理するサーバー型または常駐型のプログラムがゲストOS構成を管理し、複数の画像形成装置および画像形成装置群で最適な構成になるように判断する。例えば、ビデオ通信プログラムや中間サーバーを必要とする認証プログラムなどが挙げられる。

【0190】

上記画像形成装置は、ゲストOS上にユーザーごとのデスクトップ環境をもち、ジョブ内容を管理するモジュールを有していてもよい。

【0191】

上記画像形成装置は、ゲストOS上にプログラムを提供するアプリストアモジュールを有していてもよい。

【0192】

上記画像形成装置は、ゲストOS上の情報（例えば、メモリー、稼働状況、ユーザー、対応ジョブ、対応ファイル、動作している画像形成装置およびサーバーなど）を操作部に表示するモジュールを有していてもよい。

【0193】

上記画像形成装置は、ゲストOS上の要求をジョブとして受け付けるモジュールを有していてもよい。

【0194】

上記画像形成装置は、ゲストOS上のユーザー管理情報を必要に応じて退避前に画像形

10

20

30

40

50

成装置と共有するモジュールを有していてもよい。このような構成を採用することで、回避時に必要な情報が失われることを防止できる。

【0195】

上記画像形成装置は、ホストOSおよびゲストOSの動作状況を可視化する（例えば、AR（Augmented Reality））モジュールを有していてもよい。

【0196】

上記画像形成装置は、ゲストOSが提供する機能を外部端末で利用するモジュールを有していてもよい。

【0197】

上記画像形成装置は、画像処理ASICと協調動作するハードウェア協調モジュールモジュールを有していてもよい。

10

【0198】

上記画像形成装置は、協調モジュールを管理するためのAPIモジュール（例えば、メモリ管理機能、優先度、シーケンス制御パラメーターなどを外部から変更可能にするAPI）を有していてもよい。

【0199】

上記画像形成装置は、省電力時にゲストOS動作を制限するモジュールを有していてもよい。

【0200】

< J . まとめ >

20

本実施の形態に従う画像形成装置は、ハイパーバイザーに加えて、画像形成装置1での本体ジョブ動作による画像メモリーを把握する協調モジュールを追加するとともに、ジョブ開始前にゲストOSを画像メモリーに先行的に読出すことで、ゲストOSの切り替え時の時間遅延を削減できる、それによって、全体の本体ジョブ動作時間を延ばすことなくゲストOSで新たな処理を実施できる。

【0201】

今回開示された実施の形態はすべての点で例示であって制限的なものではないと考えられるべきである。本発明の範囲は上記した説明ではなくて特許請求の範囲によって示され、特許請求の範囲と均等の意味および範囲内でのすべての変更が含まれることが意図される。

30

【符号の説明】

【0202】

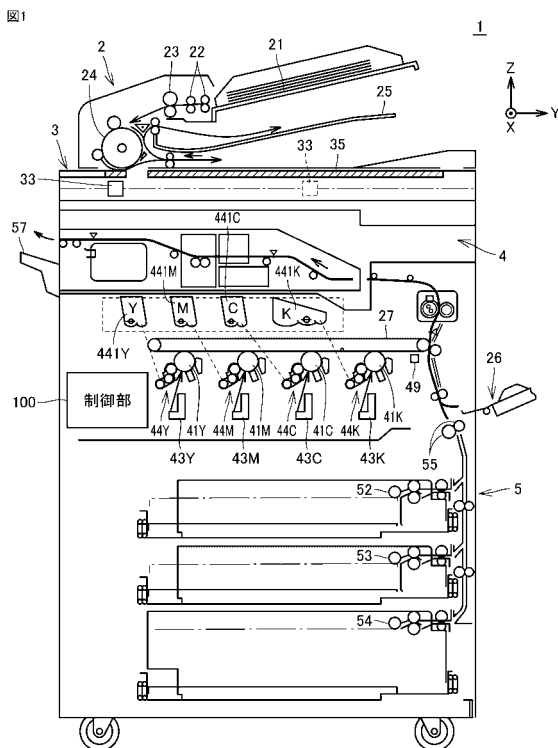
1 画像形成装置、2 自動原稿送り装置、3 イメージスキャナー、4 プリントエンジン、5 給紙部、21 原稿給紙台、22, 52, 53, 54 送出口ローラー、23 レジストローラー、24 搬送ドラム、25 排紙台、26 手差給紙部、27 転写ベルト、33 撮像デバイス、35 原稿台、41C, 41K, 41M, 41Y 感光体ドラム、43C, 43K, 43M, 43Y 画像書込部、44C, 44K, 44M, 44Y イメージングユニット、441C, 441K, 441M, 441Y トナーユニット、45 転写器、46 除電器、47 定着装置、471 加熱ローラー、472 加圧ローラー、49 IDCセンサー、51 タイミングローラー、55 搬送ローラー、57 トレイ、100 制御部、102 プロセッサー、104 内部バス、105, 106, 107, 108 インターフェイス回路、110 メインメモリー、112 システムメモリー、114, 138, 148 画像メモリー、118, 132, 142 メモリー制御IC、120 圧縮/伸張IC、130, 140 画像処理ASIC、134, 144 サブメモリー、136 表面入力画像データバッファ、146 裏面入力画像データバッファ、150 HDD、152 ハイパーバイザー、154 ホストOS、156 ゲストOS、158 アプリケーション、1581, 1582, 1583 認証アプリケーション、1584, 1585, 1586 スタブ、160 制御ファームウェア、1601 プリントアプリケーション、1602 スキャンアプリケーション、1603 FAXアプリケーション、1604 BOXアプリケーション、170 協調モジュール

40

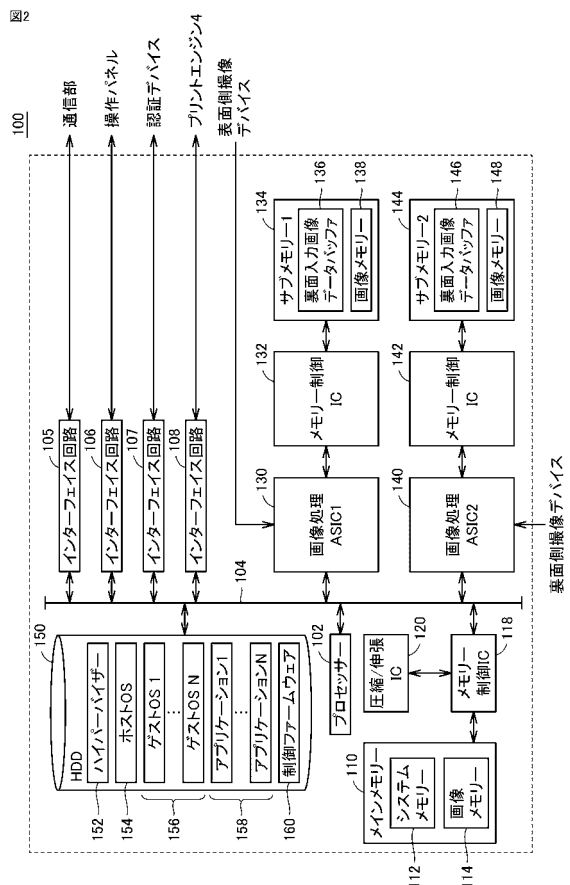
50

、 172 空きメモリー管理モジュール、 174 退避メモリー管理モジュール、 192 認証デバイス、 194 デバイス、 196 記憶デバイス。

【図 1】

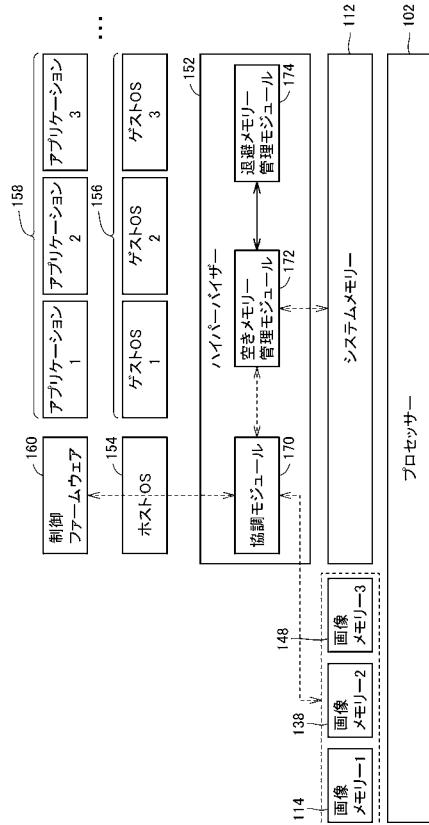


【図 2】



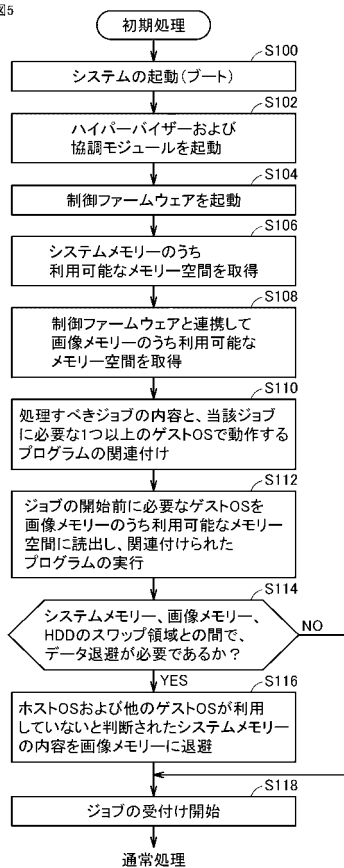
【図 3】

図3



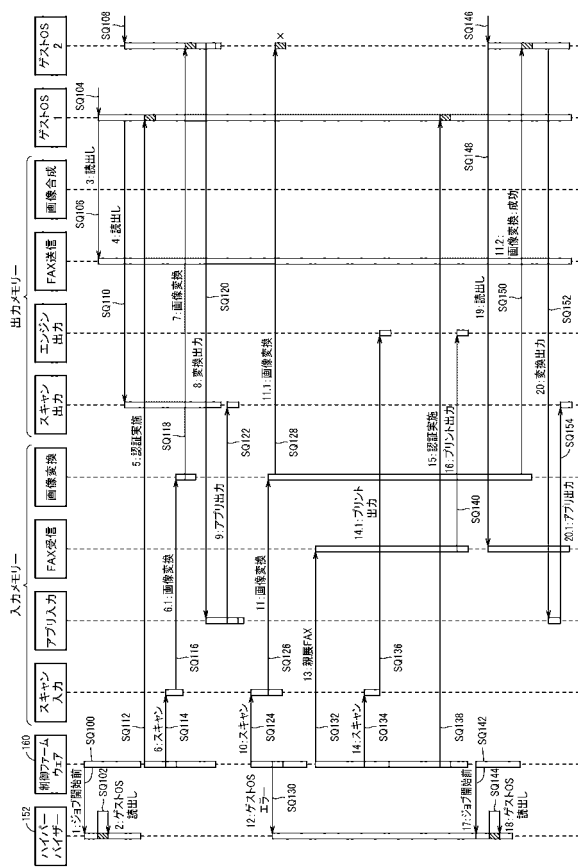
【図 5】

図5



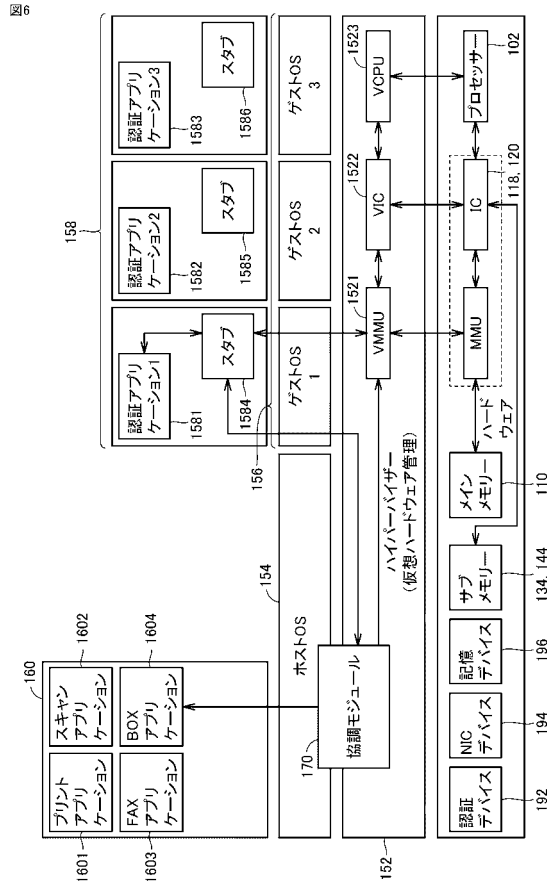
【図 4】

図4

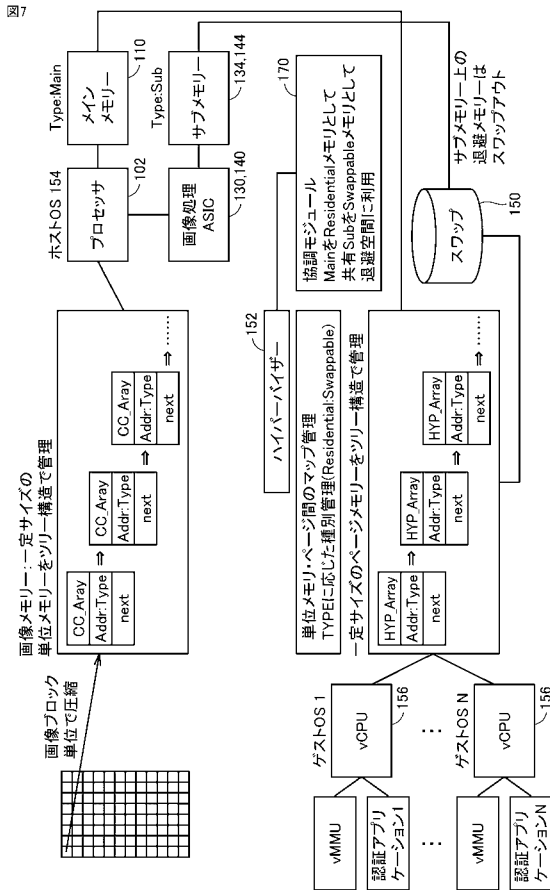


【図 6】

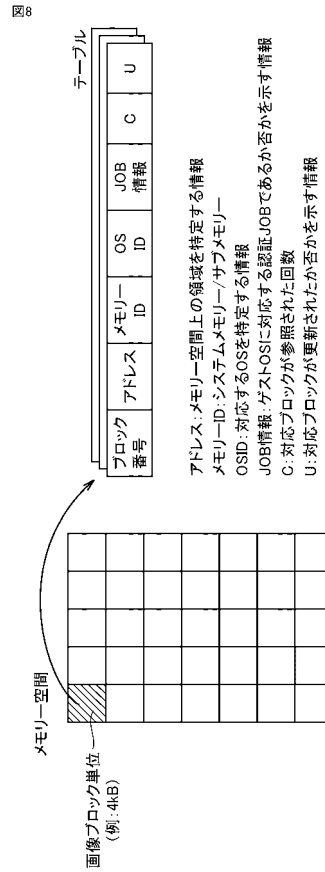
図6



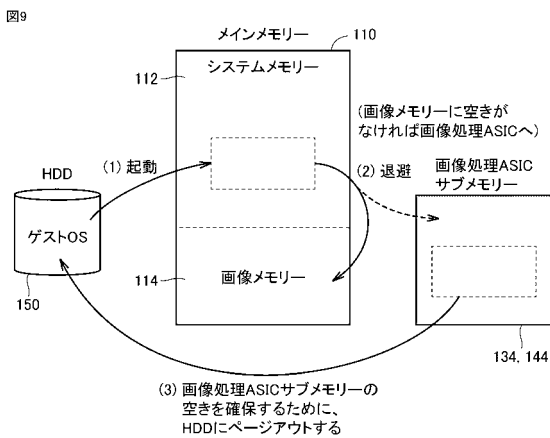
【図 7】



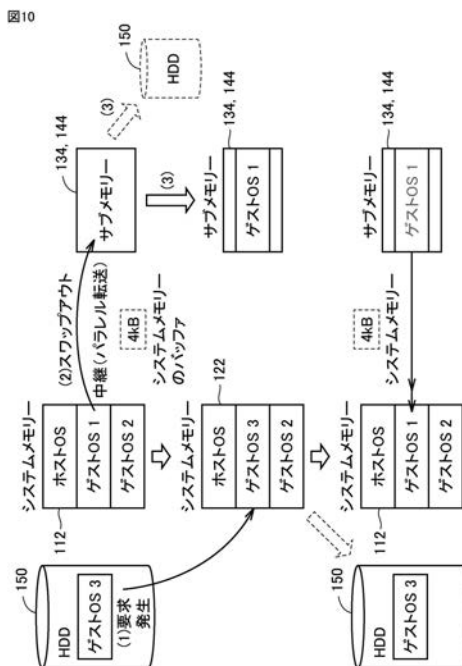
【図 8】



【図 9】

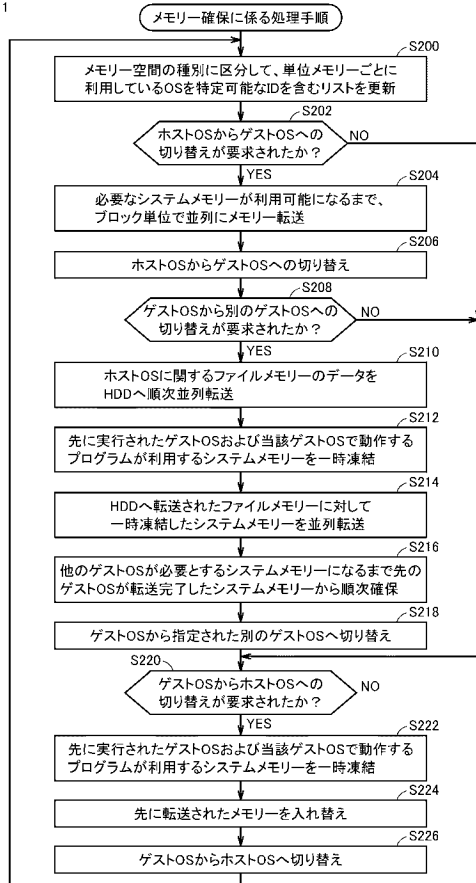


【図 10】



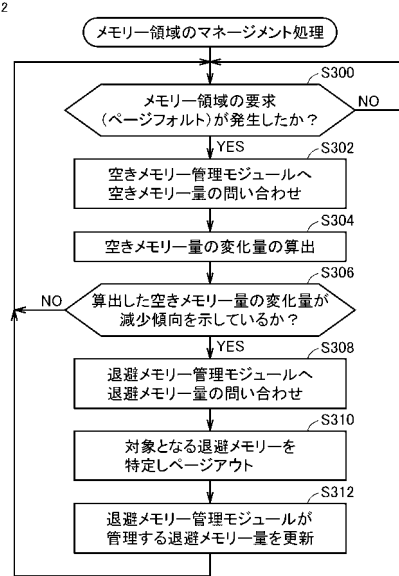
【図 1 1】

図11



【図 1 2】

図12



フロントページの続き

(51)Int.Cl. F I テーマコード(参考)
H 0 4 N 1/21 (2006.01) H 0 4 N 1/21

(72)発明者 大迫 了敏
東京都千代田区丸の内二丁目7番2号 コニカミノルタ株式会社内
Fターム(参考) 5B047 EA01 EB15 EB17 EB20
5C062 AA05 AB11 AB41 AB43 AC22 AC48
5C073 AA03 AB07 BC01 CB01