

(19) 日本国特許庁(JP)

## (12) 特許公報(B2)

(11) 特許番号

特許第5529748号  
(P5529748)

(45) 発行日 平成26年6月25日(2014.6.25)

(24) 登録日 平成26年4月25日(2014.4.25)

(51) Int.Cl.

F 1

|      |       |           |      |      |      |
|------|-------|-----------|------|------|------|
| GO6F | 3/14  | (2006.01) | GO6F | 3/14 | 310A |
| GO9G | 5/00  | (2006.01) | GO9G | 5/00 | 555D |
| GO6F | 13/38 | (2006.01) | GO9G | 5/00 | 510X |
| GO6F | 3/153 | (2006.01) | GO9G | 5/00 | 550M |
|      |       |           | GO9G | 5/00 | 510V |

請求項の数 7 (全 31 頁) 最終頁に続く

(21) 出願番号 特願2010-536221 (P2010-536221)  
 (86) (22) 出願日 平成20年12月1日 (2008.12.1)  
 (65) 公表番号 特表2011-509445 (P2011-509445A)  
 (43) 公表日 平成23年3月24日 (2011.3.24)  
 (86) 國際出願番号 PCT/US2008/085160  
 (87) 國際公開番号 WO2009/073617  
 (87) 國際公開日 平成21年6月11日 (2009.6.11)  
 審査請求日 平成23年11月30日 (2011.11.30)  
 (31) 優先権主張番号 11/987,559  
 (32) 優先日 平成19年11月30日 (2007.11.30)  
 (33) 優先権主張国 米国(US)

(73) 特許権者 508301087  
 エーティーアイ・テクノロジーズ・ユーワ  
 ルシー  
 A T I T E C H N O L O G I E S U L  
 C  
 カナダ、オンタリオ エル3ティー 7  
 エックス6、マーカム、コマース バリー  
 ドライブ イースト 1  
 One Commerce Valley  
 Drive East, Markha  
 m, Ontario, L3T 7X6  
 Canada  
 (74) 代理人 100108833  
 弁理士 早川 裕司

最終頁に続く

(54) 【発明の名称】高速周辺相互接続バスにおけるビデオレンダリング

## (57) 【特許請求の範囲】

## 【請求項 1】

プロセッサと、メモリと、ディスプレイにビデオフレームを提示するためのディスプレイインターフェースおよび第1のグラフィックプロセッサを備える第1のグラフィックサブシステムと、前記プロセッサ、前記メモリおよび前記第1のグラフィックサブシステムと相互接続するインターフェース回路と、第2のグラフィックプロセッサを有するが専用のフレームバッファメモリを有しない第2のグラフィックサブシステムと、周辺相互接続バスとを有するコンピューティングデバイスであって、前記周辺相互接続バスは、前記第2のグラフィックサブシステムへ提供され若しくは前記第2のグラフィックサブシステムから提供される全てのデータが前記インターフェース回路および前記周辺相互接続バス経由で提供されるように、前記第2のグラフィックサブシステムと前記インターフェース回路とを相互接続している、コンピューティングデバイスを動作させる方法であって、

前記ディスプレイインターフェースと相互接続している前記メモリ内にフレームバッファを確立するステップと、

ビデオフレームを、前記周辺相互接続バス経由で、プリミティブごとに、前記フレームバッファ内にレンダリングするように、前記第2のグラフィックプロセッサに対して、前記周辺相互接続バス経由で指示を提供するステップであって、前記ディスプレイに提示される全てのフレームは、前記第2のグラフィックプロセッサによってレンダリングされているステップと、

前記フレームバッファからの前記フレームを、前記ディスプレイインターフェース経由で

10

20

ディスプレイに提示するステップとを含む、方法。

【請求項 2】

前記フレームバッファ内の前記フレームを前記ディスプレイに提示するように、前記ディスプレイインターフェースをプログラムするステップを更に含む、請求項 1 に記載の方法。

【請求項 3】

コンピューティングデバイスであって、

中央処理装置と、

前記プロセッサと通信するメモリと、

グラフィックメモリ内のフレームバッファと、

前記グラフィックメモリと相互接続されたディスプレイインターフェースであって、ディスプレイにビデオフレームを提示するためのディスプレイインターフェースを備える第 1 のグラフィックサブシステムと、

前記プロセッサ、前記メモリおよび前記第 1 のグラフィックサブシステムと相互接続するインターフェース回路と、

周辺拡張カードに形成された第 2 のグラフィックサブシステムであって、ローカルなフレームバッファを定義するメモリを有しない第 2 のグラフィックサブシステムと、

前記第 2 のグラフィックサブシステムと前記インターフェース回路とを相互接続する周辺相互接続バスであって、前記第 2 のグラフィックサブシステムへ提供され若しくは前記第 2 のグラフィックサブシステムから提供される全てのデータが、前記周辺相互接続バス経由で提供される、周辺相互接続バスとを備え、

前記第 2 のグラフィックサブシステムは、レンダリングの指示を受信し、前記周辺相互接続バス経由で画像をレンダリングするように動作し、

前記メモリは、前記ディスプレイに提示される全てのビデオフレームを、前記周辺相互接続バス経由で、プリミティブごとに、前記フレームバッファ内にレンダリングするよう前記第 2 のグラフィックサブシステムを動作させるコンピュータ実行可能命令を記憶している、コンピューティングデバイス。

【請求項 4】

前記ディスプレイインターフェースは周辺拡張インターフェースの一部を構成している、請求項 3 に記載のコンピューティングデバイス。

【請求項 5】

前記ディスプレイインターフェースは、前記中央処理装置を前記周辺相互接続バスと相互接続している周辺拡張インターフェース上の前記第 1 のグラフィックサブシステムの一部を構成している、請求項 3 に記載のコンピューティングデバイス。

【請求項 6】

グラフィックサブシステムであって、

グラフィックプロセッサと、周辺相互接続バスインターフェースとを備え、前記周辺相互接続バスインターフェースは、前記グラフィックサブシステムへ提供され若しくは前記グラフィックサブシステムから提供される全てのデータが、前記周辺相互接続バスインターフェース経由で送られるように、前記グラフィックサブシステムをホストコンピューティングデバイスの周辺相互接続バスと相互接続するためのものであり、前記ホストコンピューティングデバイスは、前記グラフィックメモリと相互接続されたディスプレイインターフェースであって、ディスプレイにビデオフレームを提示するためのディスプレイインターフェースを備え、前記グラフィックサブシステムは前記グラフィックサブシステムにローカルなフレームバッファを定義するメモリを持たず、

前記グラフィックサブシステムは、レンダリングの指示を受信し、前記ビデオフレームを、前記周辺相互接続バスインターフェース経由で、プリミティブごとに、前記ホストコンピューティングデバイスの前記グラフィックメモリのフレームバッファ内にレンダリングするように動作可能であり、

前記グラフィックプロセッサは、前記ディスプレイに提示される全てのフレームをレン

10

20

30

40

50

ダーリングし、全てのグラフィックコマンドは、前記周辺相互接続バスインターフェースを介して前記ホストコンピューティングデバイスから受信される、グラフィックサブシステム。

【請求項 7】

コンピューティングデバイスを請求項 1 に記載の方法を実行するように動作させるコンピューティング実行可能命令を記憶しているコンピュータ可読媒体。

【発明の詳細な説明】

【技術分野】

【0001】

【関連出願への相互参照】

本願は、2005年5月24日に出願の米国特許出願第11/136,483号、および2006年5月30日に出願の米国特許出願第11/421,005号の米国一部継続出願であり、そのいずれも参照によりここに援用する。

【0002】

本発明は、一般に、コンピューティングデバイスに関し、より詳細には、高速周辺バスによって相互接続されているグラフィックプロセッサを有するコンピューティングデバイスに関する。

【背景技術】

【0003】

近年、コンピュータハードウェアが次第に一体化されている。コンピュータの周辺機器は、中央処理装置と協動する一体化チップセットの一部として形成されることが多い。多くの場合、コンピュータの音声、グラフィックアダプタおよびインターフェースは、一体化チップセットの一部を構成している。このような一体化により、コンピューティングデバイスのコスト低減と小型化が実現される。

【発明の概要】

【発明が解決しようとする課題】

【0004】

同時に、グラフィックプロセッサは、改善された性能と特徴を提供するために常に進歩している。最先端のグラフィックプロセッサは、中央処理装置と同程度に複雑化している。このため、統合の傾向にもかかわらず、高性能のグラフィックプロセッサは、未だに周辺機器カードに形成されることが一般的である。比較すると、集積されたグラフィックプロセッサは比較的単純であり、高性能の周辺機器カードに形成されるグラフィックプロセッサの特徴または性能を提供しない。一体型グラフィック部品は、アドオンの周辺機器カードが存在している場合には、通常はあっても冗長であり、このため無効にされる。

【0005】

同様に、集積化、外部を問わず、グラフィックアダプタをアップグレードすると、多くの場合、非力なアダプタは役に立たなくなり、廃棄されるか無効にされる。

【0006】

したがって、部品の冗長性を低減する必要が依然として存在する。

【課題を解決するための手段】

【0007】

本発明の一態様によれば、1つのグラフィックプロセッサによって生成されるグラフィックが、高速拡張／相互接続バスを介してフレームバッファに転送される。フレームバッファからのレンダリングされたフレームが、フレームバッファと通信しているディスプレイインターフェース経由でディスプレイで提示される。

【0008】

別の既存の（例えば一体化）グラフィックアダプタ／サブシステムのディスプレイインターフェースは、相互接続されているディスプレイにレンダリングされたフレームを提示するため使用することができ、利便性が高い。

【0009】

10

20

30

40

50

このようにして、グラフィックプロセッサは、ディスプレイインターフェースを有さず、場合によってはフレームバッファさえも備えている必要がないグラフィックアダプタの一部を構成しうる。同様に、グラフィックプロセッサを、最新のグラフィックプロセッサで置き換えることができる。

【0010】

グラフィックプロセッサは、周辺拡張カードに包含されるグラフィックアダプタの一部を構成しうる。

【0011】

一実施形態では、コンピューティングデバイスは、第1のグラフィックアダプタおよび第2のグラフィックアダプタを有しうる。ディスプレイに表示させる画像が、前記第2のグラフィックアダプタによってアクセス可能なメモリから転送される。前記第2のグラフィックアダプタの前記ディスプレイインターフェースが、前記第2のグラフィックアダプタの前記メモリ内の画像を、前記複数のディスプレイの少なくとも1つで提示するために使用される。このようにして、前記第2のアダプタのポートと共に、前記ディスプレイインターフェースを形成しているデバイスの電子回路が使用されうる。レンダリングされたグラフィックは、フレーム単位で転送されるか、グラフィックプリミティブ(graphics primitive)単位で転送されうる。後者の場合、グラフィックが、前記第2のグラフィックアダプタによってアクセス可能なメモリ内に、効率的に直接レンダリングされる。このメモリは、システムメモリでもアダプタメモリでもよい。

【0012】

別の実施形態によれば、前記第1のグラフィックプロセッサが、前記高速相互接続バスを介して転送されるフレームをレンダリングしている間、前記第2のアダプタのグラフィックプロセッサが無効にされるか、低電力モードに設定される。

【0013】

本発明の別の態様によれば、第1のグラフィックアダプタの一部を構成しているグラフィックプロセッサが、第1の画像を第1のディスプレイに、第2の画像を第2のディスプレイに、それぞれ描画するように動作される。前記第1のディスプレイは、前記第1のグラフィックアダプタの第1のディスプレイインターフェースと相互接続されうる。前記第2のディスプレイは、第2のグラフィックアダプタの第2のディスプレイインターフェースと相互接続されうる。前記方法は、前記第1のグラフィックアダプタのメモリ内に前記第1の画像と前記第2の画像とをレンダリングするステップと、前記第2の画像を前記第1のアダプタの前記メモリから前記第2のグラフィックアダプタのフレームバッファに転送するステップと、前記第2のグラフィックアダプタの前記第2のディスプレイインターフェースを使用して、前記第2のグラフィックアダプタの前記フレームバッファから前記第2のディスプレイに前記第2の画像を表示するステップと、を有する。

【0014】

本発明の別の態様によれば、コンピューティングデバイスは、プロセッサと、前記プロセッサと通信しているコンピュータ可読メモリと、第1のグラフィックアダプタと、前記第1のグラフィックアダプタと通信している第1のアダプタメモリと、第1のディスプレイで画像を提示するための第1のディスプレイインターフェースと、第2のグラフィックアダプタと、第2のディスプレイで画像を提示するための第2のディスプレイインターフェースと、前記第1のグラフィックアダプタと前記第2のグラフィックアダプタとを相互接続している高速バスと、を有する。プログラムコードが、前記コンピュータ可読メモリに記憶され、前記プロセッサによって実行可能である。前記プログラムコードは、実行されると、前記第1のグラフィックアダプタに、前記第1のアダプタメモリ内に第1の画像および第2の画像をレンダリングさせ、前記第2の画像を前記第1のアダプタメモリから前記第2のグラフィックアダプタのフレームバッファに転送させ、前記第2のグラフィックアダプタの前記フレームバッファから前記第2のディスプレイに、前記第2の画像を表示するように、前記第2のグラフィックアダプタの前記第2のディスプレイインターフェースをプログラムさせる。

10

20

30

40

50

**【 0 0 1 5 】**

更に別の態様によれば、前記第1のアダプタは、前記第2のアダプタによってアクセス可能な前記システムメモリに第1の画像を直接レンダリングする。前記第2のアダプタは、前記第2の画像を同じメモリにレンダリングする。前記第2のアダプタは、前記システムメモリから両方の画像を表示する。

**【 0 0 1 6 】**

本発明の別の態様によれば、コンピューティングデバイスを動作させる方法が提供される。前記コンピューティングデバイスは、プロセッサ、前記プロセッサと通信しているメモリ、前記プロセッサをグラフィックプロセッサと相互接続している周辺相互接続バス、およびディスプレイインターフェースを有する。前記方法は、前記メモリ内にフレームバッファを確立するステップと、ビデオフレームをレンダリングするように前記グラフィックプロセッサに指示するステップと、前記ビデオフレームを前記周辺相互接続バス経由で前記フレームバッファに転送するステップと、前記フレームバッファからの前記フレームを、前記ディスプレイインターフェース経由でディスプレイで提示するステップと、を有する。  
10

**【 0 0 1 7 】**

本発明の更に別の態様によれば、コンピューティングデバイスが提供される。前記コンピューティングデバイスは、中央処理装置と、前記中央処理装置と通信しているメモリと、前記中央処理装置をグラフィックプロセッサと相互接続している周辺相互接続バスと、ディスプレイインターフェースと、前記メモリに記憶されたコンピュータ実行可能命令と、を有し、前記命令は、コンピューティングデバイスを、前記グラフィックプロセッサに、ビデオフレームをレンダリングさせ、前記ビデオフレームを前記周辺相互接続バス経由で前記フレームバッファに転送し、前記フレームバッファからのレンダリングされた前記フレームを、前記ディスプレイインターフェース経由でディスプレイで提示するように適合させる。  
20

**【 0 0 1 8 】**

本発明の別の態様によれば、コンピューティングデバイスが提供される。前記コンピューティングデバイスは、中央処理装置と、前記プロセッサと通信しているメモリと、第1のフレームバッファと、前記プロセッサをグラフィックサブシステムと相互接続している周辺相互接続バスであって、前記グラフィックサブシステムは、前記グラフィックサブシステムにローカルなフレームバッファを定義するメモリを有さない周辺相互接続バスと、前記メモリと通信しているディスプレイインターフェースと、を有する。前記メモリは、前記グラフィックプロセッサを、ビデオフレームをレンダリングし、前記ビデオフレームを前記周辺相互接続バス経由で前記第1のフレームバッファに転送するように適合させるコンピュータ実行可能命令を記憶している。  
30

**【 0 0 1 9 】**

本発明の別の態様によれば、グラフィックサブシステムが提供され、前記グラフィックサブシステムは、グラフィックプロセッサと、前記グラフィックサブシステムをホストプロセッサと相互接続するための周辺相互接続バスインターフェースと、を有し、前記グラフィックサブシステムは、前記グラフィックサブシステムにローカルなフレームバッファを定義するメモリを有さず、前記グラフィックサブシステムは、ビデオフレームをレンダリングし、前記ビデオフレームを前記周辺相互接続バス経由でフレームバッファに転送するように動作可能である。  
40

**【 0 0 2 0 】**

本発明の別の態様によれば、第1の画像を第1のディスプレイに、第2の画像を第2のディスプレイに、それぞれ描画するように、第1のグラフィックアダプタを動作させる方法が提供される。前記第2のディスプレイは、第2のグラフィックアダプタのディスプレイインターフェースと相互接続されている。前記方法は、前記第1のグラフィックプロセッサと通信しているメモリ内に前記第1の画像と前記第2の画像とをレンダリングするステップと、前記第2の画像を前記第2のグラフィックアダプタのフレームバッファに転送す  
50

るステップと、前記第2のグラフィックアダプタの前記フレームバッファから前記第2のディスプレイに、前記第2の画像を表示するように、前記第2のグラフィックアダプタの前記第2のディスプレイインターフェースをプログラムするステップと、を有する。

【図面の簡単な説明】

【0021】

【図1】本発明の実施形態の例示的なコンピューティングデバイスの簡略ブロック図。

【図2】図1のコンピューティングデバイスのソフトウェアの簡略論理ブロック図。

【図3】図1のコンピューティングデバイスの一部の簡略ブロック図。

【図4】図1のコンピューティングデバイスの一部の別の模式ブロック図。

【図5A】本発明の実施形態の例示的な、図1のコンピューティングデバイスで実行されるステップを示すフローチャート。 10

【図5B】図5Aのステップの、図1のデバイスの部品への効果を示す図。

【図6A】本発明の実施形態の例示的な、図1のコンピューティングデバイスで実行されるステップを示すフローチャート。

【図6B】図6Aのステップの、図1のデバイスの部品への効果を示す図。

【図6C】図6Aのステップの、図1のデバイスの部品への効果を示す図。

【図7】本発明の別の実施形態の例示的な、図1のコンピューティングデバイスで実行されるステップを示すフローチャート。

【図8】本発明の別の実施形態の例示的な、図1のコンピューティングデバイスで実行されるステップを示すフローチャート。 20

【図9】本発明の別の実施形態の例示的なコンピューティングデバイスの一部の別の部分簡略ブロック図。

【図10】本発明の実施形態の例示的な、図9のデバイスのソフトウェアによって実行されるステップを示すフローチャート。

【図11A】図9のデバイスの動作を示す簡略ブロック図。

【図11B】図9のデバイスの動作を示す簡略ブロック図。

【図12】本発明の別の実施形態の例示的なコンピューティングデバイスの一部の別の簡略模式ブロック図。

【発明を実施するための形態】

【0022】

本発明のほかの態様および特徴は、添付の図面を参照して、以下の本発明の特定の実施形態の説明を検討すれば、当業者にとって明らかとなるであろう。

【0023】

図面において、本発明の実施形態を、例示のみを目的として図示する。

【0024】

図1は、本発明の実施形態の例示的なコンピューティングデバイスの簡略模式ブロック図である。コンピューティングデバイス10は、従来のインテル×86アーキテクチャをベースとしている。しかし、明らかなように、本発明は、PowerPC、AMD×86または他のアーキテクチャを有するコンピューティングデバイスにおいても容易に実施することができる。 40

【0025】

コンピューティングデバイス10は、プロセッサ12を有し、プロセッサ12は一体型インターフェース回路14, 20を介してシステムメモリ16および周辺機器と相互接続されている。

【0026】

例示的なコンピューティングデバイス10のプロセッサ12は、従来の中央処理ユニットであり、例えばインテル(登録商標)×86ファミリのマイクロプロセッサと互換のマイクロプロセッサなどである。

【0027】

一体型インターフェース回路14と20は、それぞれ、ノースブリッジ、サウスブリッジ

50

と呼ばれることが多く、プロセッサ 1 2 に対して、周辺機器およびメモリにインタフェースを行う。図に示すように、ノースブリッジ 1 4 は、プロセッサ 1 2 、メモリ 1 6 、高速相互接続バス 2 2 経由で複数の拡張スロット 1 8 、およびサウスブリッジ 2 0 を相互接続している。サウスブリッジ 2 0 は、低速の周辺機器およびインターフェースを相互接続している。これには、例えば、統合 I D E / S A T A ポート 2 4 経由のディスクドライブ、統合ユニバーサルシリアルバス ( U S B ) ポート 2 6 経由のユニバーサルシリアルバスデバイスや、例えば、公知の P C I または I S A 規格に準拠した、低速相互接続バス 2 8 経由の他の周辺機器がある。サウスブリッジ 2 0 は、一体化音声部品 3 0 、ネットワークインターフェース ( 図示せず ) などを含んでもよい。

## 【 0 0 2 8 】

10

例示的な相互接続バス 2 2 は、ギガバイト / 秒の範囲の帯域幅を有する P C I エクスプレス ( P C I e ) バスであり、この帯域幅でのデータ転送の読み出しおよび書き込みに対応している。例えば、現在の P C I e バスの速度により、最大 2 6 6 M H z または 5 3 3 M H z の速度での双方向データ転送が可能となり、最大 2 . 1 ギガバイト / 秒または 4 . 2 ギガバイト / 秒のデータ速度が可能となる。

## 【 0 0 2 9 】

図の実施形態では、ノースブリッジ 1 4 は、コンピューティングデバイス 1 0 を、モニタ、 L C D パネル、テレビなどの形態の第 1 のディスプレイ 3 2 と相互接続するために適した一体型グラフィックアダプタ 3 4 を更に備える。

## 【 0 0 3 0 】

20

下記で詳細に説明するように、相互接続バス 2 2 経由で相互接続される追加のグラフィックアダプタ 5 2 が、例えば、相互接続バス 2 2 上の拡張スロット 1 8 内の周辺拡張カード 5 0 の一部を構成しており、更にコンピューティングデバイス 1 0 の一部も構成している。

## 【 0 0 3 1 】

図に示した実施形態では、コンピューティングデバイス 1 0 は、メモリ 1 6 に記憶されているソフトウェアを実行する。図 2 に示すように、例示的なソフトウェア 1 0 0 には、オペレーションシステム 1 0 2 、グラフィックライブラリ 1 0 4 およびアプリケーションソフトウェア 1 0 6 が含まれ、メモリ 1 6 に記憶されている。代表的なオペレーティングシステムとしては、 W i n d o w s V i s t a 、 W i n d o w s X P 、 W i n d o w s N T 4 . 0 、 W i n d o w s M E 、 W i n d o w s 9 8 、 W i n d o w s 2 0 0 0 、 W i n d o w s 9 5 または L i n u x オペレーティングシステムが挙げられる。代表的なグラフィックライブラリとしては、 M i c r o s o f t D i r e c t X ライブラリや O p e n G L ライブラリが挙げられる。コンピュータメモリ 1 6 ( 図 1 ) と、相互接続されたディスクドライブ ( 図示せず ) とは、ランダムアクセスメモリ、リードオンリーメモリおよびディスクストレージメモリの適した組み合わせを有し、これは、本発明の実施形態の例示的な方法によりデバイス 1 0 を適合させる、オペレーティングシステムとグラフィックアダプタドライバプログラムを記憶および実行するためにデバイス 1 0 によって使用される。例示的なソフトウェア 1 0 0 ( 図 2 ) は、例えば、リードオンリーメモリに記憶されるか、または I D E / S A T A ポート 2 4 と通信しているディスクドライブ等の外部周辺機器からロードされうる。コンピュータ可読媒体 2 7 は、光学記憶媒体、磁気ディスク、テープ、 R O M カートリッジなどであってもよい。

30

## 【 0 0 3 2 】

40

グラフィックライブラリ 1 0 4 またはオペレーティングシステム 1 0 2 には、グラフィックアダプタ 3 4 , 5 2 との下位レベルの通信に使用されるグラフィックドライバ 1 0 8 が更に含まれる。ソフトウェアは階層構造を有しており、上位レベルのレイヤが、特定の機能を提供するために下位レイヤを使用する。アプリケーションは、二次元または三次元のグラフィックをレンダリングするために、オペレーティングシステム 1 0 2 とグラフィックライブラリ 1 0 4 を使用しうる。この文脈において、「レンダリング」とは、描画またはその他の表現のためにグラフィック画像を生成することを意味し、例えば、ポリゴン

50

のレンダリング、レイトレイシング、画像のデコーディング、ライン描画などが含まれる。明らかなように、本発明の実施形態の例示的なソフトウェアは、グラフィックライブラリ 104 および / またはドライバ 108 の一部を構成しうる。ここに示す実施形態では、例示的なソフトウェアは、Open GL ライブラリおよび関連するドライバの一部を構成しうる。Open GL については、「The Official Guide to Learning Open GL、第 1.1 版」に詳しく記載されており、その内容をここに参照により援用する。しかし、当業者は、本発明の実施形態の例示的な方法は、Microsoft Direct 3D ライブラリ、カスタムライブラリまたはルーチンを使用しないアプリケーション（ゲームなど）、あるいは他の任意の三次元アプリケーションまたはライブラリの一部を構成しうることを認めるであろう。

10

### 【0033】

一体型グラフィックアダプタ 34 を示す例示的なコンピューティングデバイス 10（および特にノースブリッジ 14）の一部の更に詳しい簡略ブロック図が、図 3 に示される。図に示すように、ノースブリッジ 14 は、高速バスインターフェース 36 およびメモリコントローラ 38 を有し、ブリッジ 14 を、相互接続バス 22 とシステムメモリ 16 とを相互接続している。ノースブリッジ 14 は、一体型グラフィックアダプタ 34 を更に有し、一体型グラフィックアダプタ 34 は、グラフィックプロセッサ 40、コマンドプロセッサ 42 およびディスプレイインターフェース 44 を有する。

### 【0034】

グラフィックプロセッサ 40 は、システムメモリ 16 内に割り当てられるフレームバッファ 46 に、三次元画像をレンダリング可能な三次元（3D）グラフィックプロセッサの形態をとってもよく、グラフィックの高速レンダリングのために複数のパイプラインとパイプラインステージを有しうる。

20

### 【0035】

通常、シングルモニタ動作では、コマンドが、コマンドプロセッサ 42 によって従来の方法でグラフィックプロセッサ 40 に渡される。例えば、コマンドプロセッサ 42 は、メモリ 16 内にコマンドキュー 47 を定義するために使用するレジスタを備えうる。アプリケーションソフトウェア 106 またはオペレーティングシステム 102 は、ライブラリ 104 を使用してグラフィック画像をレンダリングする。ドライバソフトウェア 108 は、グラフィックプロセッサ固有のコマンドを生成して、コマンドキュー 47 にこのコマンドをキューイングする一方、キューの終点を変更するためにコマンドプロセッサ 42 のレジスタも更新する。グラフィックプロセッサ 42 は、とりわけ、フレームバッファ 46 にレンダリング画像をレンダリングするために、キューイングされたコマンドを、非同期でパイプライン方式で処理する。グラフィックプロセッサ 40 がキュー内の次のコマンドを処理可能な場合、コマンドプロセッサ 42 に信号を供給し、これを受け、コマンドプロセッサ 42 は、キュー 47 にある次のコマンドを取り出し、コマンドプロセッサ 42 のレジスタのキュー pointer を進める。

30

### 【0036】

ディスプレイインターフェース 44 は、メモリコントローラ 38 経由でフレームバッファ 46 をサンプリングし、テレビ、モニタなどの形態のディスプレイ 32（図 1）等のビデオデバイスに 1 つ以上のビデオ画像を表示するために、VGA ポート、コンポジットビデオポート、DVI ポート、HDMI、Display Port などの形態で、1 つ以上のビデオ出力ポート 48 に画像を提示する。このようにして、フレームバッファ 46 内の、グラフィックプロセッサ 40 によってレンダリングされる画像が提示されうる。

40

### 【0037】

理解されるように、ディスプレイインターフェース 44 は、バッファ内データをディスプレイ装置に表示するために変換するのに適したインターフェースであれば、どのようなものでもよい。例えば、ディスプレイインターフェース 44 は、RAMDAC の形態をとりうる。ディスプレイインターフェース 44 は、通常、例えば複数のレジスタによってプログラム可能であり、これにより、ドライバソフトウェア 108 または同様のソフトウェアまた

50

はグラフィックプロセッサ40が、そのディスプレイ出力を提示するために、メモリ16内に先頭アドレスを設定可能となる。同様に、インターフェース44のピクセル深度（すなわちピクセル当たりのビット数）とスクリーン解像度も、通常はプログラム可能である。

#### 【0038】

理解されるように、メモリ16内に1つのフレームバッファ46を割り当てる代わりに、2つのフレームバッファが割り当てられてもよい。この2つのバッファは、レンダリングされるグラフィックをよりスムーズに表現するために交互に使用されうる。ディスプレイインターフェース44は、いつの時点でも、2つのバッファの一方の内容しか表示しない。表示中のバッファは、通常「フロントバッファ」と呼ばれる。グラフィックプロセッサ40は、ディスプレイインターフェース44によって現在表示されていないフレームバッファである、いわゆる「バックバッファ」に表示するために画像をレンダリングする。フレームのレンダリングが完了すると、フロントバッファとバックバッファが反転されて、ディスプレイインターフェース44が、バックバッファを提示するようにプログラムされる。このようにして、グラフィックプロセッサ40によってレンダリングされる逐次的なフレーム間の遷移がスムーズに表示されうる。

#### 【0039】

特に、アダプタ34によって使用されるレジスタおよびフレームバッファメモリ46は全て、グローバルなメモリ空間内のアドレスにマップされている。

#### 【0040】

しかし、本発明の例示的なコンピュータデバイス10は、図1に示すように、ノースブリッジ14と通信している高速相互接続バス22上の拡張スロット経由でデバイス10に挿入されうる周辺拡張カード50に形成されているグラフィックアダプタ52を更に有する。ノースブリッジ14に相互接続されているグラフィックアダプタ52が、図4に模式的に概略的に示されている。図に示すように、グラフィックアダプタ52は、グラフィックプロセッサ54、ビデオメモリ56、コマンドプロセッサ58、メモリコントローラ60、およびディスプレイインターフェース62を有する。1つ以上のポート64, 66により、ディスプレイ68（図1）などの1台以上のディスプレイ装置とアダプタ52を相互接続することができる。

#### 【0041】

コマンドプロセッサ58はコマンドプロセッサ42と、ディスプレイインターフェース62はディスプレイインターフェース44と、それぞれ全く同様に機能する。グラフィックプロセッサ54は、通常動作モードにおいて、グラフィックプロセッサ40と全く同様に動作する。グラフィックアダプタ52は、自身のメモリ56と、メモリ56およびシステムメモリ16へのアクセスを提供するメモリコントローラ60とを有する。ローカルメモリ56は、なかでも、グラフィックプロセッサ54によって使用され、フレームバッファとして機能しうる、グラフィックアダプタ52用の1つ以上のバッファ70を保持するために使用される。グラフィックプロセッサ54用のコマンドキュー57も、システムメモリ16内に作成される。

#### 【0042】

この場合も、ローカルメモリ56は、メモリコントローラ60を介して、グローバルシステムメモリ空間内のアドレスでアドレッシング可能である。同様に、コマンドプロセッサ54およびディスプレイインターフェース62によって使用されるレジスタは、グローバルアドレス空間内のアドレスでアドレッシング可能である。上で説明したように、アダプタ34によって使用されるレジスタおよびフレームバッファ46はグローバルメモリ空間にマップされる。このため、コマンドプロセッサ58とグラフィックプロセッサ54は、ディスプレイインターフェース44とコマンドプロセッサ42によって使用されるフレームバッファ46およびレジスタに書き込むことができる。

#### 【0043】

また、グラフィックプロセッサ54は、メモリコントローラ60と連動して、メモリアドレスにおいて、システムメモリ16およびアダプタメモリ56との間でデータを転送す

10

20

30

40

50

るために、ブロック転送操作を実行するように動作可能である。

【0044】

経済的な一体化部品を提供するために、一体型グラフィックアダプタ34は機能が限定されている。例えば、一体型グラフィックコントローラの解像度、三次元グラフィック機能などは、比較的制限されている。このため、一体型グラフィックアダプタ34は、通常、グラフィックアダプタ52よりもグラフィック処理能力が格段に低い。例えば、アダプタ34は、アダプタ52よりも動作が低速であったり、機能が限られていたりなどである。

【0045】

ここで、一体型アダプタ34またはアダプタ52は、単独で、ポート48などの複数のポートを介して、複数台の物理的ディスプレイ装置の相互接続に対応しても、対応していくなくてもよい。しかし、各アダプタは、少なくとも1台の物理的ディスプレイと相互接続されうる。このため、本発明の実施形態の例示では、メモリ16にロードされたアダプタドライバソフトウェア108が、グラフィックアダプタ52を、ディスプレイ32と68の両方で提示するグラフィックをレンダリングするマスタ（または第1の）グラフィックアダプタとして動作させ、グラフィックアダプタ34を、第2またはスレーブアダプタとして動作させる。

【0046】

明らかなように、このマスタ／スレーブ動作モードにおいては、一体型グラフィックアダプタ34が実質的に無効にされる。しかし、オンボードグラフィックアダプタ34（特にディスプレイインターフェース44）の部品を、相互接続されたモニタや他のディスプレイ装置を駆動するために使用することができる。このため、カード50が、複数のディスプレイポートおよびディスプレイドライバを有している必要はない。この代りに、ディスプレイインターフェース44とそのポート48を、カード50によって駆動されるディスプレイ68またはディスプレイに加えて、少なくとも1台のモニタまたはディスプレイ32を駆動するために使用することができる。アダプタ52が複数のポート（ポート64, 66など）を有する場合、プロセッサ52が、これらの複数のポートを介して提示するための画像を附加的にレンダリングしてもよい。当然、ドライバソフトウェア108は、上記のように、アダプタ34を、その従来の動作モードで動作させてもよい。アダプタ34（このためアダプタ52）の動作モードの選択は、アプリケーションソフトウェア106によって行われるか、またはアプリケーションソフトウェア106を介して行われる。

【0047】

詳細には、図4に模式的に示すように、本発明の実施形態の例示では、マスタ／スレーブ動作モードにおいて、メモリ16内のドライバソフトウェア108が、アダプタ52のメモリ56内に、アクティブフレームバッファ70とセカンダリアダプタバッファ72を割り当てる。アクティブフレームバッファ70は、アダプタ52用の従来のフレームバッファとして機能し、このため、アダプタ52のディスプレイ68に表示するためのフレームを記憶する。このように、グラフィックプロセッサ54は、従来の方法で、メモリ56内のアクティブフレームバッファ70に、二次元または三次元のグラフィックをレンダリングする。カード50のディスプレイインターフェース62は、相互接続されたモニタまたはディスプレイ装置に表示させるために、アクティブフレームバッファ70の内容を変換する。任意選択で、前述のように、アクティブフレームバッファ70の代りに、フロントバッファとリアバッファが使用されてもよい。

【0048】

グラフィックプロセッサ54は、更に、セカンダリアダプタフレームバッファ72に、ディスプレイ装置32に表示させるための画像をレンダリングする。セカンダリアダプタフレームバッファ72へのフレームのレンダリングが終了すると、下記で詳細に説明するように、セカンダリアダプタフレームバッファ72の内容が、グラフィックアダプタ34のディスプレイインターフェース44によって提示するために、フレームバッファ46から転送される。

10

20

30

40

50

## 【0049】

第1実施形態では、メモリ16内のグラフィックアダプタドライバソフトウェア108が、図5Aに示したステップS500を実行する。詳細には、ステップS502において、ドライバソフトウェア108は、メモリ56のセカンダリアダプタバッファ72をレンダリングするために、グラフィックプロセッサ54宛のコマンドを生成する。上で説明したように、コマンドがキュー47にキューイングされるのと全く同じ方法で、これらのコマンドがアダプタ52のコマンドキュー57にキューイングされ、グラフィックプロセッサ54によって実行される。フレームがレンダリングされると、ドライバソフトウェア108は更に、1つ以上のコマンドを生成し、この結果、ステップS504において、プロセッサ54は、メモリ56内のセカンダリアダプタフレームバッファ72の内容を、グラフィックアダプタ34用のフレームバッファ46として割り当てられたメモリ16の当該領域に、ビットロック転送(BITBLT)する。

## 【0050】

これに対し、グラフィックアダプタ34のディスプレイインターフェース44は、ポート48において相互接続されたモニタまたはディスプレイに、自身のフレームバッファ46の内容を表示するように事前にプログラムされている。このようにして、ポート48とポート64によって相互接続されたディスプレイ32と68に表示される画像が、グラフィックプロセッサ54によってレンダリングまたは描画され、有利である。バッファ72およびフレームバッファ46へのステップS504の効果が図5Bに示される。

## 【0051】

第2実施態様では、図6Aに示したステップS600が実行される。まず、フロントフレームバッファとバックフレームバッファが、アダプタ52のメモリ16内に割り当てられる。便宜上、これらを、バッファ46a, 46bと呼び、図6Bに示す。この場合も、ステップS602において、プロセッサ54は、ステップS502を参照して上で説明したように、ディスプレイインターフェース44によって提示させる任意のフレームを、メモリ56のセカンダリアダプタフレームバッファ72にレンダリングする。

## 【0052】

デバイス32に表示するためのフレームのレンダリングが終了すると、ステップS604において、ドライバソフトウェア108は、メモリ56内のセカンダリアダプタフレームバッファ72の内容を、グラフィックアダプタ34用のメモリ16内の現在のバックバッファにビットロック転送(BITBLT)するように、グラフィックプロセッサ54をプログラムする。BITBLTが終了すると、ステップS606において、ドライバは、アダプタ34のバックバッファとフロントバッファを反転させる(すなわち、バックバッファをフロントバッファとして使用する)ように、ディスプレイインターフェース44のレジスタをプログラムするように、プロセッサ54をプログラムする。このことは、ディスプレイインターフェース44がデータを提示するために使用する開始アドレスを指定するように、ディスプレイインターフェース44のレジスタを直接再プログラムすることによって行うことができる。これに対し、グラフィックアダプタ34のディスプレイインターフェース44は、ポート48において相互接続されたモニタ(すなわちディスプレイ68)に、ディスプレイインターフェース44のバッファの内容を提示する。前のフロントバッファが、今度はアダプタ34用のバックバッファとして使用され、ステップS604において、ディスプレイ32に表示させるためにプロセッサ54によってレンダリングされる次のフレームが、最初にこのバックバッファから転送される。ドライバソフトウェア108は、当然、各バッファ46a, 46bの開始位置と、フロントバッファおよびバックバッファとしての状態を保持している。バッファ46a, 46bへのステップS604, S606の効果が、図6B、図6Cに模式的に示される。

## 【0053】

同期を行わない場合には、1つのバッファ46の使用、あるいはアダプタ34のディスプレイインターフェース44の直接のプログラミングにより、目に見えるティアリングが発生しうる。すなわち、インターフェース44によるフレーム出力の途中でディスプレイパラ

10

20

30

40

50

メータが変更されるため、ディスプレイインターフェース44のレジスタが再プログラムされる垂直帰線消去期間以外の任意の時点でも、ティアリングが見えてしまう。

#### 【0054】

このため、更に別の第3実施形態では、図7に示したステップS700が実行されうる。この場合も、最初に、アダプタ34によって使用されるメモリ16内に、フロントリアバッファ46aとリアバッファ46bが割り当てられる。ステップS502, S602を参照して上で説明したように、この場合も、プロセッサ54は、ローカルメモリ56のバッファ72に、装置32に表示させるための画像をレンダリングする。

#### 【0055】

ディスプレイ32に表示させるためのフレームのレンダリングが完了すると、ステップS704において、メモリ16内のドライバソフトウェア108は、メモリ56内のセカンダリアダプタフレームバッファ72の内容を、グラフィックアダプタ34のメモリ16内の現在のバックバッファにビットブロック転送(BITBLT)するように、グラフィックプロセッサ54をプログラムする。ステップS706において、ドライバソフトウェア108は、更に、アダプタ34のコマンドキュー47に、フロントバッファとバックバッファを反転させるためのコマンドをキューイングする。しかし、コマンドプロセッサ42のレジスタが、コマンドキュー47内の未処理のコマンドを反映させるように、ドライバソフトウェア108によって更新されない。その代わりに、ステップS708において、ドライバソフトウェア108は、ステップS706で開始されたビットブロック転送の終了後に、コマンドプロセッサ42のコマンドキューを更新するためのコマンドを、グラフィックプロセッサ54に提供する。これにより、バックバッファをフロントバッファに戻させるための命令が、コマンドプロセッサ42によってプロセッサ40に提供される。これを受けて、グラフィックプロセッサ40は、(すなわち、ポート48において相互接続されたモニタまたはディスプレイに前のバックバッファの内容を提示するよう、ディスプレイインターフェース44のレジスタを再プログラムすることによって)、キューイングされた、アダプタ34のバックバッファとフロントバッファを反転させるための命令を実行する。

#### 【0056】

更に別の第4実施態様では、図8に示したステップS800が実行される。この場合も、まず、アダプタ34のメモリ16内に、フロントバッファとバックバッファが割り当てられる。この場合も、プロセッサ54は、前述のように、ローカルメモリ56のバッファ72に、ディスプレイ32用の画像をレンダリングする。

#### 【0057】

ドライバソフトウェア108は、更に、グラフィックプロセッサ40に、メモリ56内のセカンダリフレームバッファ72の内容を、グラフィックアダプタ34用の現在のバックフレームバッファとして割り当てられたシステムメモリ16の当該領域にビットブロック転送(BITBLT)させるコマンドを、アダプタ34のキュー47にキューイングする。また、ドライバソフトウェア108は、リアバッファをフロントバッファとして使用するようにディスプレイインターフェース44をプログラムすることにより、グラフィックプロセッサ40に、アダプタ34のフロントバッファとバックバッファを反転させるコマンドを、キュー47にキューイングする。しかし、フレームがバッファ72にレンダリングされるまで、キュー pointerを定義しているコマンドプロセッサ42のレジスタは更新されない。

#### 【0058】

このため、ディスプレイ32用のフレームのレンダリングの終了時に、グラフィックプロセッサ54は、コマンドプロセッサ42用のキューpointerを定義しているコマンドレジスタを更新させる命令を提供して、アダプタ34のコマンドキュー内の2つの未処理のコマンドを反映させる。これを受けて、グラフィックプロセッサ40は、バッファ72の内容をバックバッファにビットブロック転送し、バックバッファとフロントバッファを反転させるためにキューイングされた命令を(すなわち、ポート48において相互接続され

10

20

30

40

50

たモニタまたはディスプレイに前のバックバッファの内容を提示するために、ディスプレイインタフェース 44 のレジスタを再プログラムすることにより) 実行する。

#### 【0059】

ここで理解されるように、上に記載した実施形態により、マスタグラフィックプロセッサが、複数の独立したフレームバッファへの二次元または三次元のグラフィック画像の描画の責任を負うことが可能となる。スレーブグラフィックアダプタは、1つのフレームバッファに作成されたディスプレイ画像に使用されうる。追加ディスプレイを駆動するためには必要なデバイスの電子回路とポートは、スレーブグラフィックアダプタ 34 によって提供されうる。このようにして、スレーブグラフィックコントローラに設けられた電子回路が、複数台のディスプレイの駆動を可能にするために効率的に使用されうる。

10

#### 【0060】

バッファ 70, 72 を使用して1つの面を定義することができ、この面に、アプリケーションソフトウェア 106 が、グラフィックアダプタ 52, 34 と相互接続された2台以上のディスプレイ 48, 64 にまたがって表示させる1つのグラフィック画像をレンダリングすることができ、有利である。したがって、ドライバソフトウェア 108 は、利用可能なスクリーンサイズをアプリケーションソフトウェア 106 に報告し、この値はバッファ 70 とバッファ 72 のサイズと等しい。このため、バッファ 70 が  $m . s u b . 1 \times n$  の解像度を有し、バッファ 72 が  $m . s u b . 2 \times n$  の解像度を有する場合、ドライバソフトウェア 108 は、アプリケーションソフトウェア 106 のアプリケーションに、利用可能なスクリーンサイズとして  $(m . s u b . 1 + m . s u b . 2) \times n$  を報告しうる。バッファ 70, 72 が、メモリ 56 内で隣接する  $(m . s u b . 1 + m . s u b . 2) \times n$  ピクセル位置を占める場合には、ドライバソフトウェア 108 によるレンダリングが簡単になる。アプリケーションソフトウェア 106 は、ドライバソフトウェア 108 を使用して、グラフィックプロセッサ 40 に、バッファ 70, 72 によって定義される面を構成している、 $(m . s u b . 1 + m . s u b . 2) \times n$  ピクセルの解像度を有する大きな画像をレンダリングさせる。このため、バッファ 70, 72 は、画像のペアを記憶し、このペアの各画像が、アプリケーションによってレンダリングされる大きな画像の一部を構成する。大きな画像のそれぞれのレンダリングが終了すると、ドライバソフトウェア 108 は、上で説明したステップ S500, S600, S700 または S800 に従って、ディスプレイ 32 に表示させるために、バッファ 72 内の画像をフレームバッファ 46 に転送しうる。バッファ 70 内の画像は、通常の方法でディスプレイ 68 に表示されうる。エンデューザは、ディスプレイ 32 と 68 を並べて観ており、アプリケーション 106 によってレンダリングされる大きな画像を認識する。理解されるように、ゲームの形態のアプリケーション 106 は、変更を必要とせずに、マルチディスプレイの利点を享受しうる。

20

#### 【0061】

ディスプレイ 68 用にフロントバッファとバックバッファが使用される場合、ディスプレイ 32 に表示させる面部分を保持しているメモリ 56 内のバッファが、ディスプレイ 68 に表示させる面部分をレンダリング中であるバッファの隣に位置することを保証するために、バッファ 72 の代わりに、2つの異なるバッファが割り当てられて使用されてもよいしたがって、S602, S604, S702, S704 またはステップ S802, S704 は、完成直後の画像を保持しているバッファを、バックバッファ 46a または 46b にビットロック転送するように変更されてもよい。既存のアプリケーションソフトウェア 106 を変更する必要がないため、有利である。上記に代えて、アプリケーションソフトウェアが、報告された大きなスクリーンサイズを単に使用してもよい。ドライバソフトウェア 108 は、複数台のディスプレイにまたがって表面をレンダリングさせる。

30

#### 【0062】

ここで理解されるように、図に示した実施形態は、スレーブグラフィックアダプタとして一体型グラフィックコントローラを使用するが、スレーブグラフィックアダプタが、単に、グラフィックフレームをスレーブフレームバッファに転送させるのに十分な帯域幅のバス経由で、グラフィックアダプタ 52 と通信する別のグラフィックアダプタであっても

40

50

よい。このため、スレーブグラフィックプロセッサは、別のPCIe対応拡張カードとして作製されてもよい。このため、例えば、グラフィックカードがアップグレードされた場合も、古い拡張カードは、相互接続されたディスプレイを駆動可能であるため、引き続き使用することができる。同様に、相互接続バスは、PCIe相互接続バスでなくてもよく、ディスプレイインターフェース44が使用するリフレッシュレートに等しい速度で、フレームバッファ間でデータを転送させる帯域幅を有する他の任意の適切なバスであってもよい。同様に、フレームバッファ46（およびフロントバッファおよびバックバッファ46a, 46b）が、システムメモリ16内に作成されると説明したが、アダプタ34に対してメモリなローカルの一部として、容易に作成されてもよい。

## 【0063】

10

本発明の更に別の代替の実施形態では、バス22を介したレンダリングにより、高速バス22によって相互接続されたグラフィックアダプタ（またはサブシステム）が、一体型アダプタ／サブシステムの代理となることが可能となってもよい。

## 【0064】

この点において、図9は、本発明の別の実施形態の代表的なコンピューティングデバイス10'の一部の例の代表的な簡略ブロック図である。コンピューティングデバイス10'は、米国特許出願第11/421,005号明細書に詳細に記載されている。

## 【0065】

しかし簡単に説明すると、デバイス10'は2つのグラフィックサブシステム30', 40'を備える。グラフィックサブシステム30'は、グラフィックエンジン／プロセッサ32'、メモリコントローラ72'、ディスプレイインターフェース74'およびバスインターフェース78'を備える。第2のグラフィックサブシステム40'は、PCIeバスなどの高速バス22'経由でグラフィックサブシステム30'と通信している。グラフィックサブシステム40'は、自身のグラフィックエンジン／プロセッサ42'、メモリコントローラ52'、ディスプレイインターフェース54'を備える。グラフィックサブシステム40'は更にグラフィックメモリ50'と通信している。明らかなように、サブシステム40'は、バス22'を介してメモリ14'にグラフィックをレンダリングするように適合されている。

20

## 【0066】

デバイス10'はラップトップまたは小型コンピューティングデバイスの形態のポータブルコンピューティングデバイスとして好適に形成することができる。このように、1つのハウジングがDC電源、ディスプレイ26'および上述のマザーボードおよび構成要素を収容してもよい。第2のグラフィックサブシステム40'は、コンピューティングデバイスの他の部分を収容する1つのハウジングに追加されても、デバイス10'が物理的に相互接続された時にデバイス10'の一部のみを構成するドッキングステーションの一部を構成してもよい。

30

## 【0067】

デバイス10'は、少なくとも2つの電力消費モード、すなわち、高電力消費モードと低電力消費モードで動作することができる。図に示した実施形態では、デバイス10'の高電力モードとは、デバイス10'がAC（主）電源に接続された電源によって給電されている時に入るモードであり、低電力消費モードとは、デバイス10'が1つ以上のバッテリ、燃料電池などを使用するDC電源によって給電されている時に入るモードでもよい。別例では、電力消費モードが、例えばユーザの好み、実行中のソフトウェアアプリケーションのタイプ、バッテリレベルなどに基づき、ユーザによって選択されるか、ソフトウェアによって制御されるか、または別の方で選択されてもよい。

40

## 【0068】

デバイス10'のソフトウェア制御動作の構成は、米国特許出願第11/421,005号明細書に開示されており、上記のソフトウェア100と類似する点を有する。

## 【0069】

デバイス10'としてのデバイス10'の動作を制御しているソフトウェアの一部は、

50

高電力消費状態と低電力消費状態の間を遷移する。詳細には、図10は、デバイス10'のシステムメモリ内部のソフトウェアの制御下で中央処理装置12'（図1のプロセッサ12など）によって実行される本発明の実施形態の例示的なソフトウェアブロックS800'を示す。ブロックS800'はデバイス10'が状態変化を受けるたびに実行され、これに対して、サブシステム30'，40'を相応に構成する必要がある。図に示すように、ブロックS802'において、ソフトウェアは、デバイス10'が高電力消費モードまたは低電力消費モードのいずれを取るべきかを判定する。

#### 【0070】

デバイス10'が高電力消費モードを再開（またはこれに移行）すべき場合、ブロックS804'～S810'が実行される。ブロックS804において、グラフィックサブシステム40'は、まだフル動作（高電力消費）モードに設定されていなければ、高電力消費モードに設定される。このことは、グラフィックサブシステム40'を制御するドライバを介して、電力コントローラ60'に適切な信号を供給することによって実行することができる。次に、ブロックS806，S808において、グラフィックサブシステム40'が有効にされる。この場合も、このことは、ブロックS804において、グラフィックサブシステム30'に関連する、相互接続されたディスプレイを論理的に無効にし、ブロックS808において、グラフィックサブシステム40'と接続されたディスプレイを論理的に有効にすることによって実行することができる。この場合も、ブロックS806，S808は、米国特許第11/421,005号明細書に記載のEnumDisplayDevices()コールおよびChangeDisplaySettingsEX()コールなどの適切なオペレーティングシステムAPIコールによってか、またはハードウェアとの直接通信によって実行することができる。

#### 【0071】

特に言えば、グラフィックサブシステム40'に接続されている物理的ディスプレイは存在しない。グラフィックサブシステム40'の動作を制御しているドライバソフトウェアは、ステップS810'において、関連するメモリ50'内部の代わりに、グラフィックサブシステム30'のバッファ14'に画像をレンダリングするように構成される。（例えば、PCIeバスとして実施された）高速バス22'が存在している場合、このようなレンダリングが、一部には、このバスによって実現される転送速度のおかげで、バス22'を介して可能となる。

#### 【0072】

このレンダリングは、バスを介してフレームバッファにプリミティブごと(primitive by primitive)に送られ、このため、バス22'を介してレンダリングされた画像が転送されてもよい。このことは、サブシステム40'によってアクセス可能なメモリ内にバッファ14'を割り当て、サブシステム40'用のドライバソフトウェアに、バッファ14'のアドレスを提供することにより容易に行うことができる。別法として、明らかに、フレームが1つのバッファでレンダリングされ、バス22'を介してダイレクトメモリーアクセスなどによって転送されてもよい。

#### 【0073】

同様に、グラフィックサブシステム30'用のドライバは更に、グラフィックサブシステム40'によってメモリ14'内のフレームバッファにレンダリングされた画像を、相互接続されたディスプレイ26'で提示するために、グラフィックサブシステム30'のディスプレイインターフェース74'に、メモリ14'のフレームバッファをサンプリングさせるように構成されている。同時に、グラフィックサブシステム30'のドライバは、グラフィックサブシステム30'のグラフィックエンジン32'に対し、実質的に活動停止状態またはアイドル状態に留まるように指示することができる。この動作モードは、図11Aに模式的に示されており、グラフィックサブシステム40'およびグラフィックサブシステム30'のアクティブなブロックのみが網掛けされている。

#### 【0074】

明らかに、図11Aの実施形態では、メモリ50'、ディスプレイインターフェー

10

20

20

30

40

50

ス 5 4 ' は使用されていない。このため、これらの機能ブロックはサブシステム 4 0 ' から削除することができ、コスト低減が可能となる。これにより得られるサブシステム 4 0 ' は、ディスプレイインターフェースを有する必要はなく、サブシステム 4 0 ' にローカルなフレームバッファを定義するメモリがないこともある。このようなグラフィックサブシステムを作成することは、サブシステム 3 0 ' によって提供される機能をサブシステム 4 0 ' が補完するように作成されうるので、有益であろう。例えば、サブシステムは、3D グラフィックまたはビデオデコーディング機能を提供するグラフィックエンジン 4 2 ' を提供してもよい。グラフィックエンジン 3 2 ' はこれらの機能を搭載していないてもよい。同時に、グラフィックエンジン 3 2 ' によって提供される 2D グラフィック機能を、サブシステム 4 0 ' に搭載する必要はない。消費者は、追加の機能が必要な場合にのみグラフィックサブシステム 3 0 ' を追加してもよい。

#### 【 0 0 7 5 】

デバイス 1 0 ' が低電力消費モードに移行するか、このモードを再開した場合、ブロック S 8 1 2 ' ~ S 8 1 8 ' が実行される。概説すると、グラフィックサブシステム 4 0 ' は、部分的または完全に無効にされて低電力消費モードに設定され、この場合も、レンダリングがグラフィックサブシステム 3 0 ' によって実行される。これを行うため、ブロック S 8 1 2 ' において、グラフィックサブシステム 3 0 ' に関連する、相互接続されたディスプレイが有効にされ、ブロック S 8 1 4 ' において、グラフィックサブシステム 4 0 ' と物理的に接続されたあらゆるディスプレイが論理的に無効にされる。次に、グラフィックサブシステム 3 0 ' の動作を制御するドライバソフトウェアが、グラフィックサブシステム 3 0 ' にメモリ 1 4 ' に画像をレンダリングせしめるように再度構成される。ディスプレイインターフェース 7 4 ' は、ポート 7 8 ' と相互接続されたディスプレイ 2 6 ' で画像を提示するために、メモリ 1 4 ' をサンプリングし続ける。同様に、ブロック S 8 1 8 ' において、プロセッサ 1 2 ' は最初に電力コントローラ 6 0 ' に適切な信号を供給し、グラフィックサブシステム 4 0 ' を低電力状態に設定する。最も単純な形態では、電力コントローラ (図示なし) がグラフィックサブシステム 4 0 ' への電力を遮断するか、またはグラフィックサブシステム 4 0 ' を低電力スリープモードに設定する。この場合も、この低電力消費モードでは、電圧が抑えられるか、グラフィックサブシステム 4 0 ' の全部または一部が電源切断されるか、グラフィックサブシステム 4 0 ' が使用する選択されたクロックが減速されるか、この組み合わせが行われる。詳細には、グラフィックサブシステム 4 0 ' のグラフィックエンジン 4 2 ' が、アイドル状態に留まるか、または実質的にアイドル状態に留まる (例えば低速化されるか、無効にされるか、または電源切断される)。この動作モードは、図 1 1 B に模式的に示されており、アダプタ 4 0 ' とグラフィックサブシステム 3 0 ' のアクティブな機能ブロックのみが網掛けされている。非作動状態 / アイドル状態の機能ブロックは、完全に無効化されるか、または低電圧または低クロック速度で動作される。

#### 【 0 0 7 6 】

任意選択で、グラフィックサブシステム 3 0 ' の一部が、グラフィックエンジン 3 2 ' の不使用時に無効にされてもよい。このことは、グラフィックサブシステム 4 0 ' が画像のレンダリングを担っている任意の時に、G P I O または類似の回路によって無効にできる 1 つ以上のボルテージアイランド (voltage island) に、グラフィックエンジン 3 2 ' および他の構成要素を配置することによって容易に行うことができる。

#### 【 0 0 7 7 】

他の変更例もまた明白であろう。例えば、図 1 1 A に示す高電力モードにおいて、グラフィックサブシステム 3 0 ' とグラフィックサブシステム 4 0 ' の両方が、メモリ 1 4 ' またはメモリ 5 0 ' にレンダリングしてもよい。このようにして、2つのグラフィックサブシステム 3 0 ' , 4 0 ' が、それぞれメモリ 1 4 ' の交番フレームにレンダリングするか、またはメモリ 1 4 ' の各フレームの交番部分 (例えばスキャニングライン) にレンダリングして、協調して動作することができる。

#### 【 0 0 7 8 】

10

20

30

40

50

更に別の実施形態では、上で説明したように、追加ディスプレイをグラフィックサブシステム 30' , 40' に接続して、高電力消費モードでの複数台のディスプレイの同時使用を可能にすることができる。このようにして、ディスプレイインターフェース 54' を、第 2 のディスプレイを駆動するために使用することができる。デバイス 10' を、低電力消費モードへの移行時に、図 11B に示すように動作するように構成してもよい。

#### 【0079】

同様に、デバイス 10' ( または 10 ) は、バス 22' ( または 22 ) に接続された複数の追加のグラフィックサブシステムを有してもよく、高電力消費モードにおいては、その全てがアクティブとなり、グラフィックサブシステム 30' のディスプレイインターフェース 74' を介してグラフィックをレンダリングしてもよい。低電力消費モードへの移行時に、このグラフィックサブシステムが無効化され、グラフィックサブシステム 30' のグラフィックエンジン 32' にレンダリングが任せられる。

10

#### 【0080】

図 12 に示す更に別の実施形態では、コンピューティングデバイス 10' はダイレクトメモリアクセス ( DMA ) コントローラ 90 を備えてもよい。DMA コントローラ 90 はデータをメモリ 50' からメモリ 14' に転送することができる。このように、デバイス 10' の高電力消費モードにおいて、グラフィックサブシステム 40' は画像をメモリ 50' にレンダリングしうる。レンダリングされたこれらの画像はその後、DMA コントローラ 90 によってメモリ 14' 内のフレームバッファに転送されうる。DMA コントローラ 90' は、( 例えはグラフィックエンジン 32' または 42' の DMA エンジンとして ) グラフィックサブシステム 30' または 40' の一部を構成するか、またはコンピューティングデバイス 10' 内に別の方法で配置されうる。データは、メモリ 50' からメモリ 14' に、バス 20' を介して転送されるか、または別 の方法で直接転送されうる。ディスプレイインターフェース 74' は、上述のように動作を続け、レンダリングされた画像をディスプレイ 26' で提示するためにメモリ 14' のフレームバッファをサンプリングする。この場合も、高電力消費モードにある図 10 のデバイス 10' のアクティブなプロックが、図 12 で網掛けして図示されている。

20

#### 【0081】

当然、上記の実施形態は、例示のみを目的としており、限定を意図するものではない。本発明を実施する記載した実施形態は、形状、構成要素の配置、操作の詳細および順序がさまざまに変更される。むしろ、本発明は、請求の範囲によって規定される範囲に、このような変更の全てを含むことを意図する。

30

【図1】

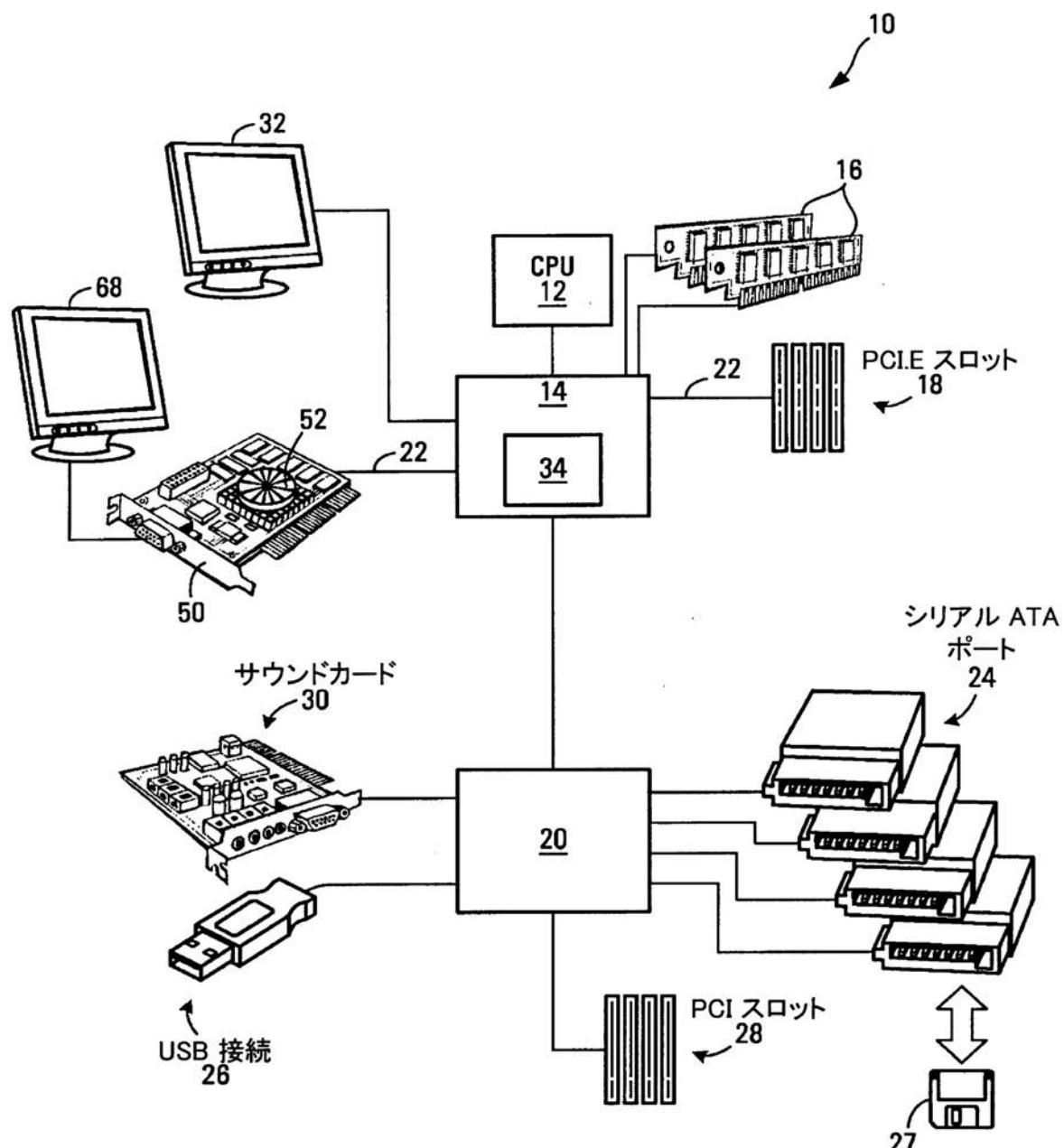
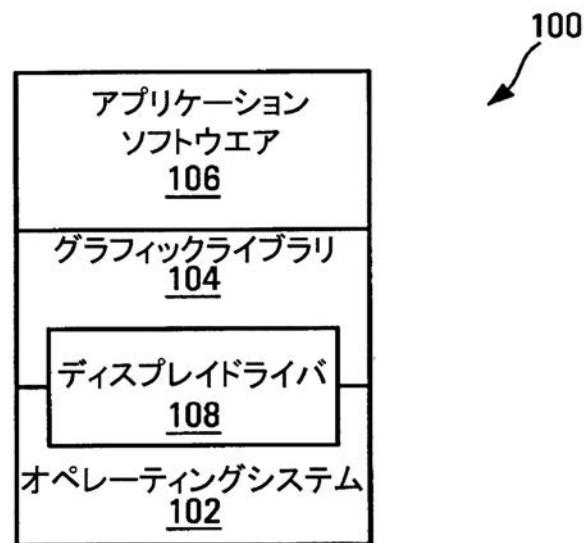


FIG. 1

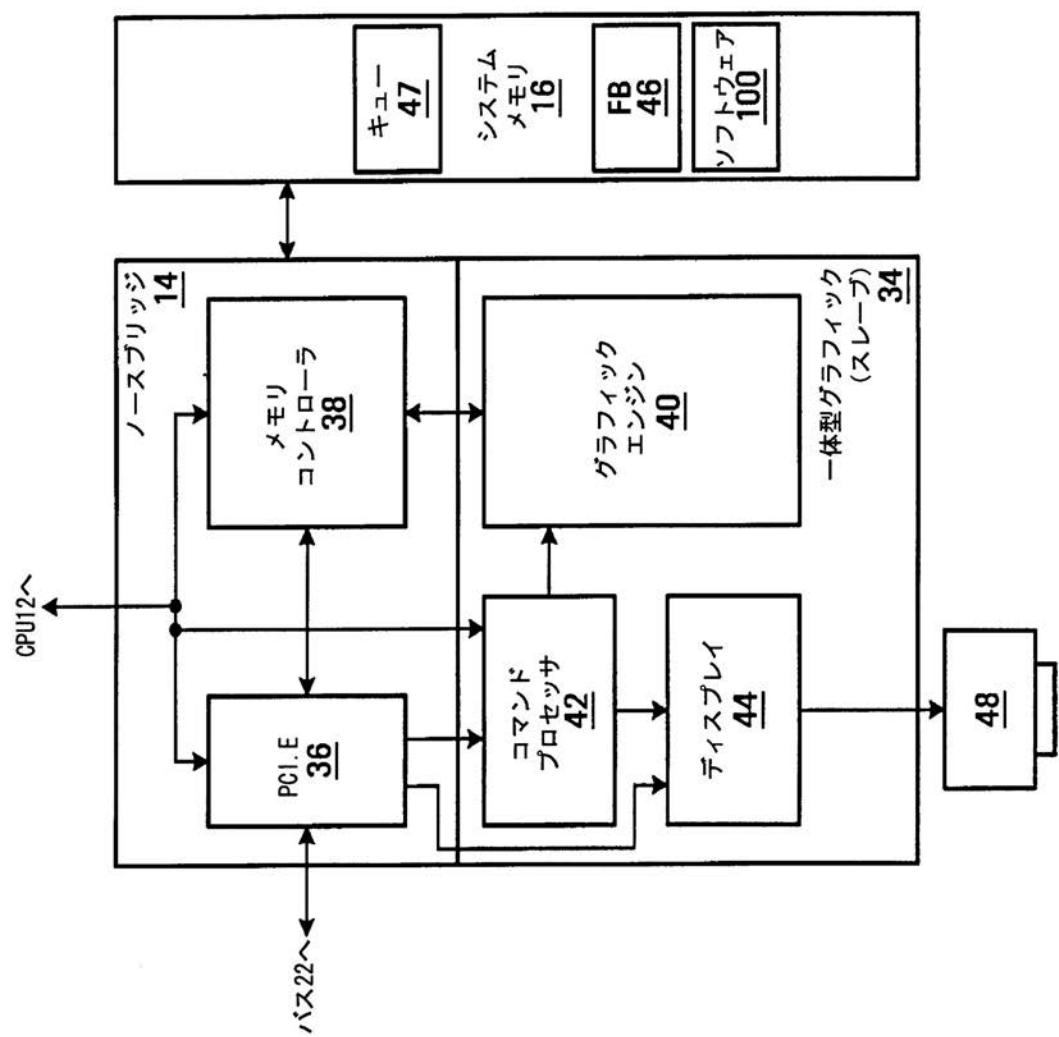
【図2】

**FIG. 2**

【図3】

FIG. 3

10



【図4】

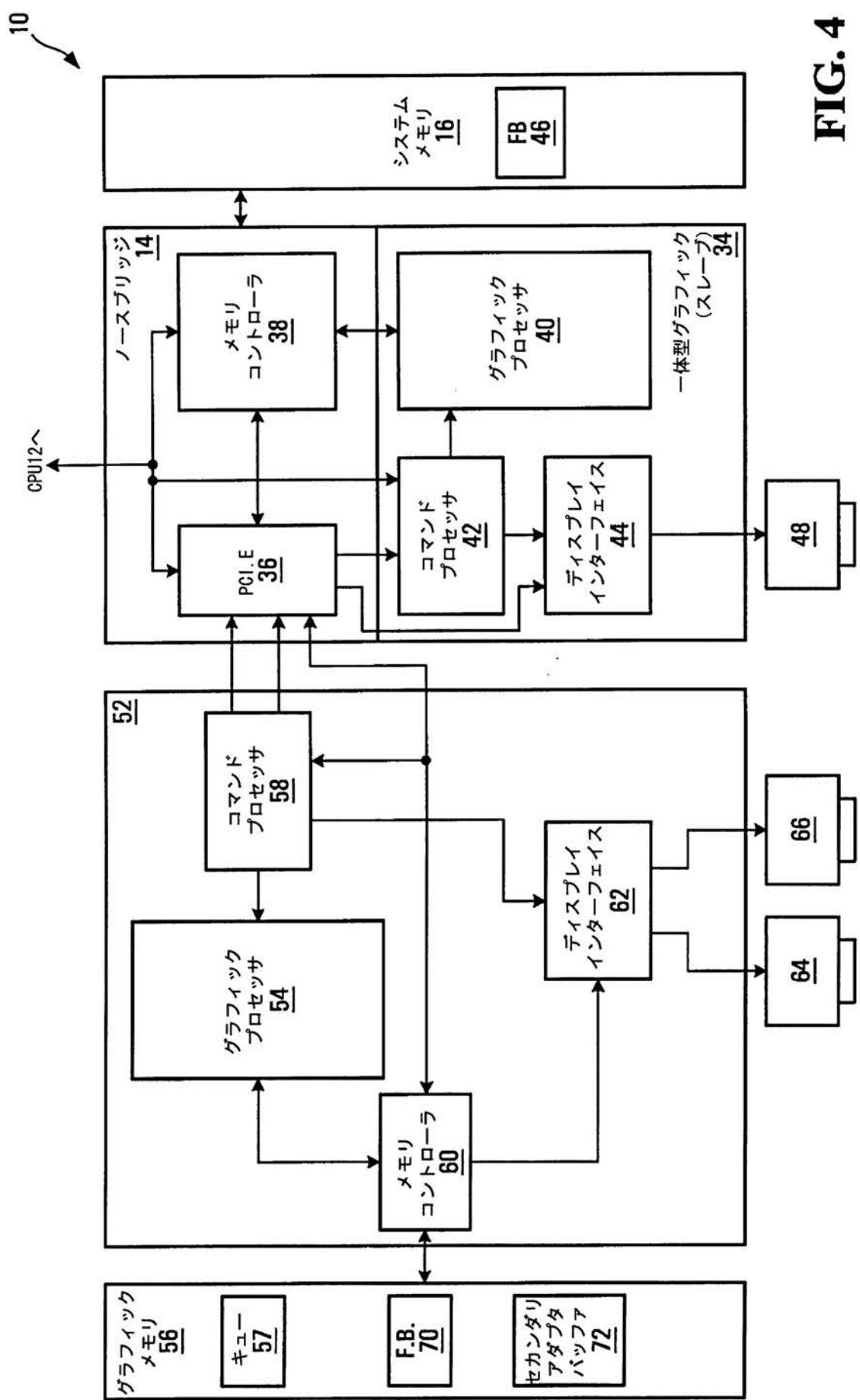
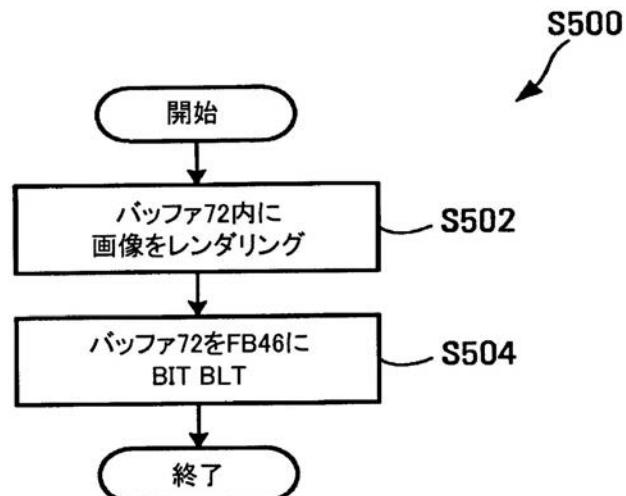
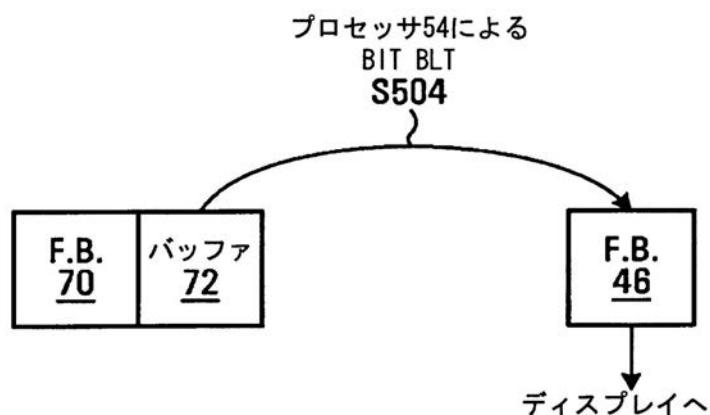


FIG. 4

【図 5 A】

**FIG. 5A**

【図 5 B】

**FIG. 5B**

【図 6 A】

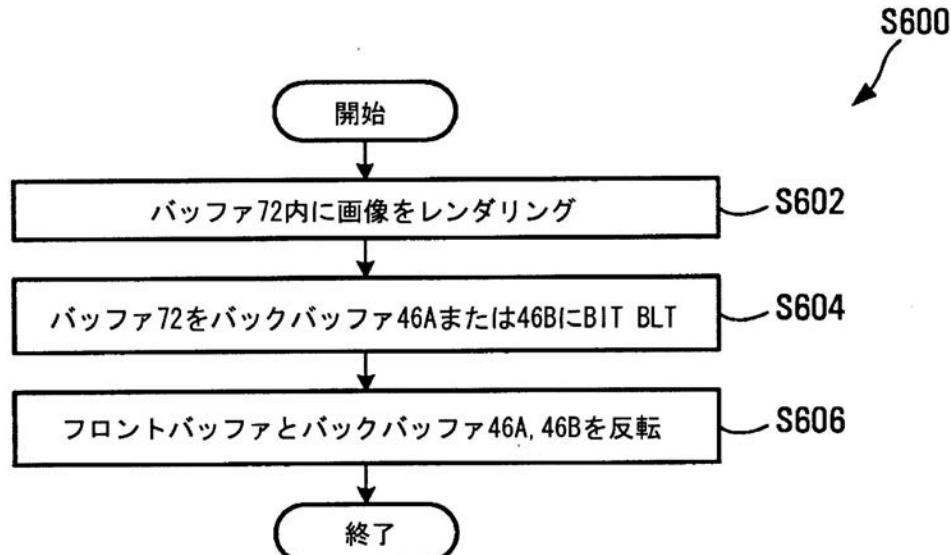


FIG. 6A

【図 6 B】

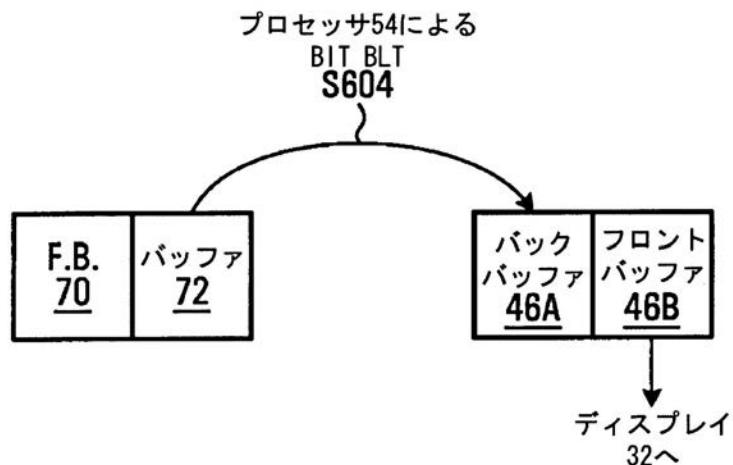


FIG. 6B

【図 6 C】

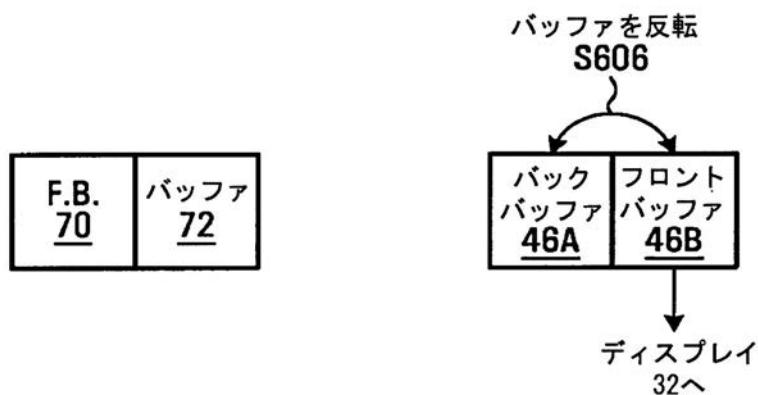


FIG. 6C

【図7】

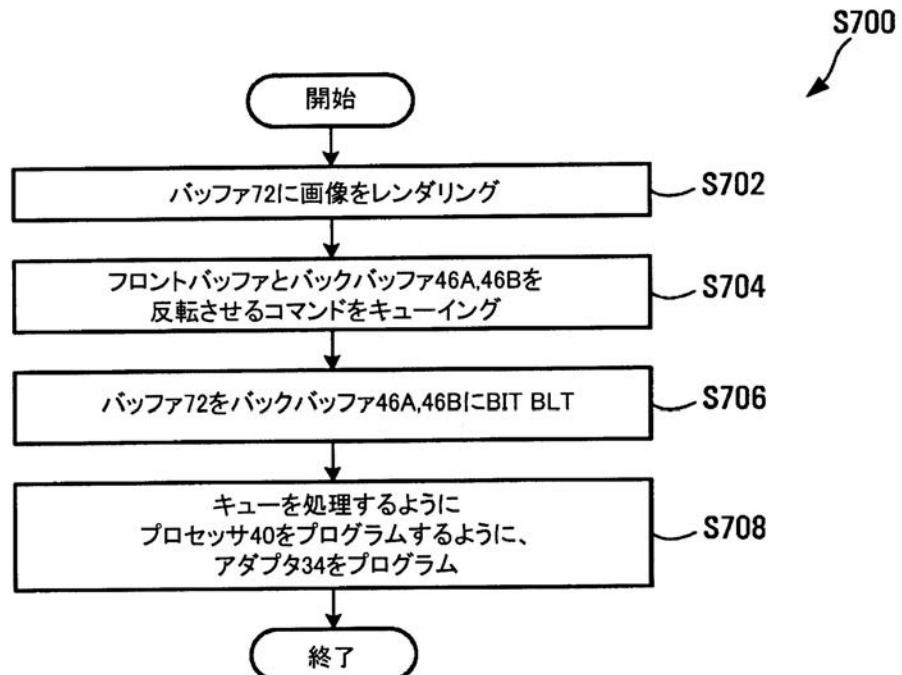


FIG. 7

【図8】

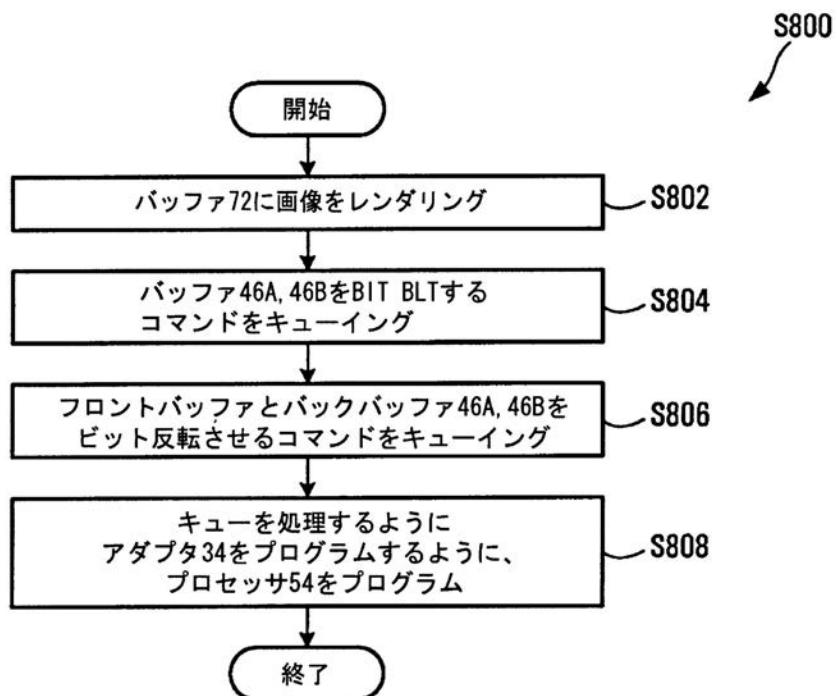


FIG. 8

【図9】

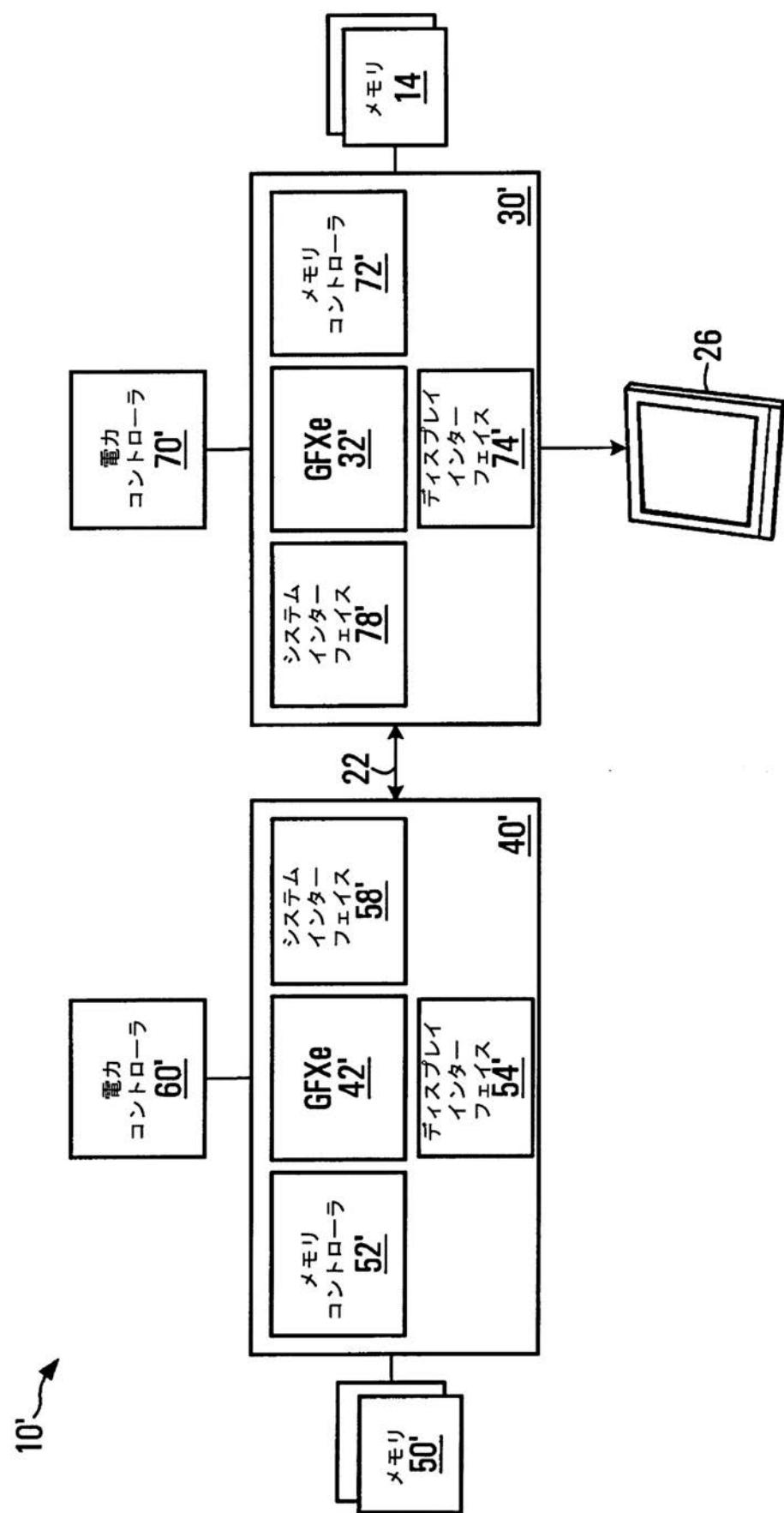


FIG. 9

【図10】

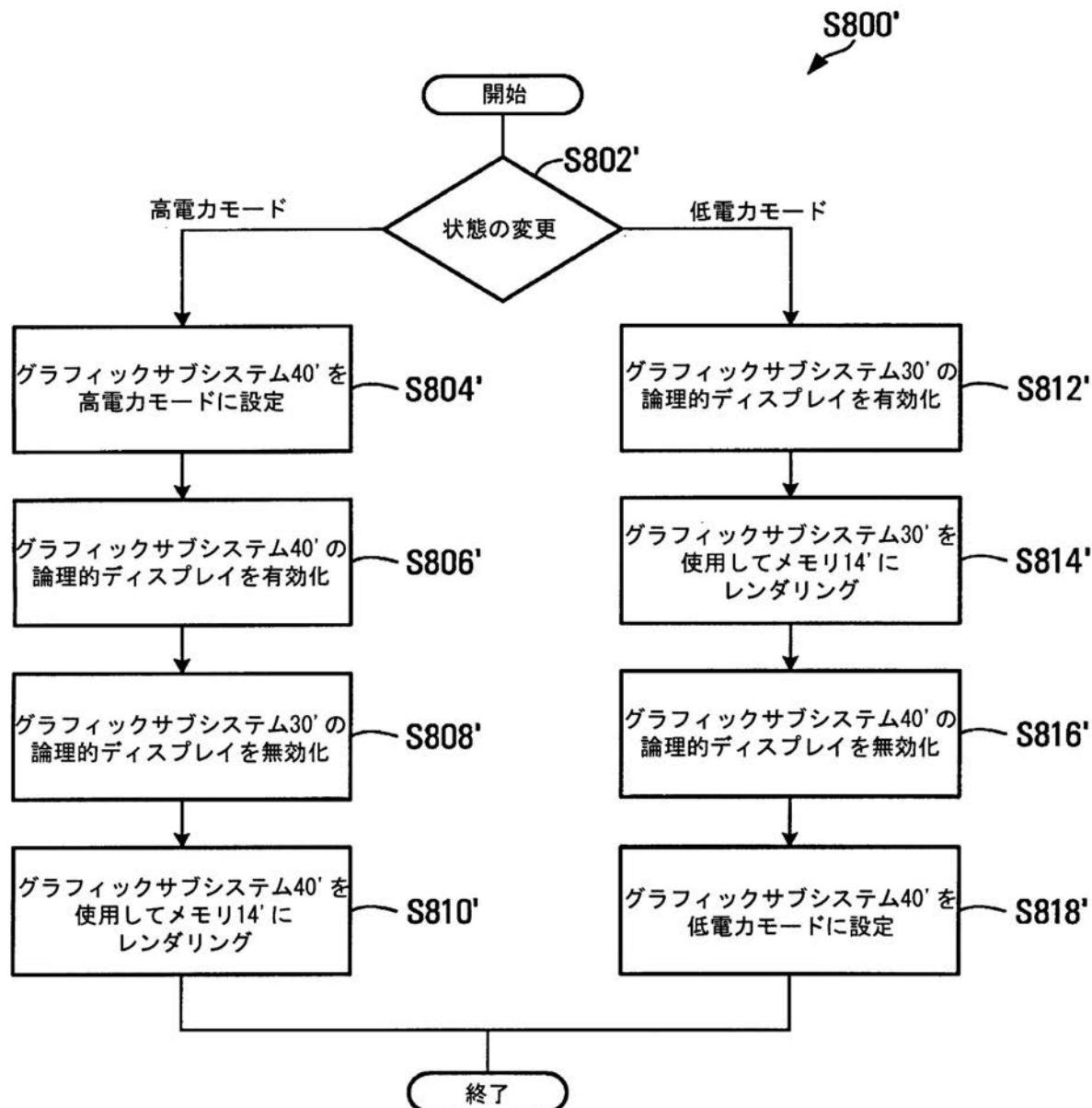


FIG. 10

【図 11A】

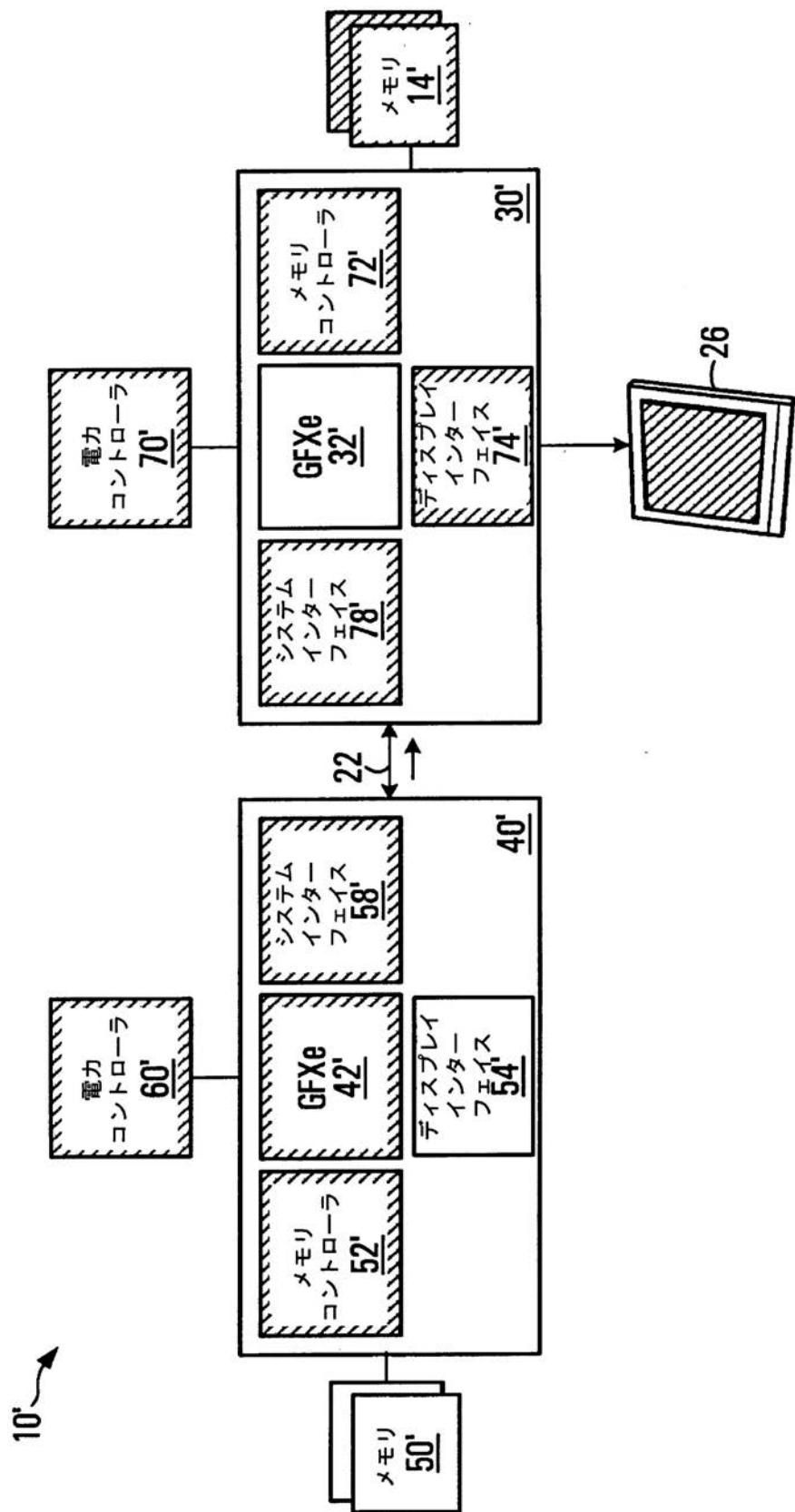


FIG. 11A

【図 11B】

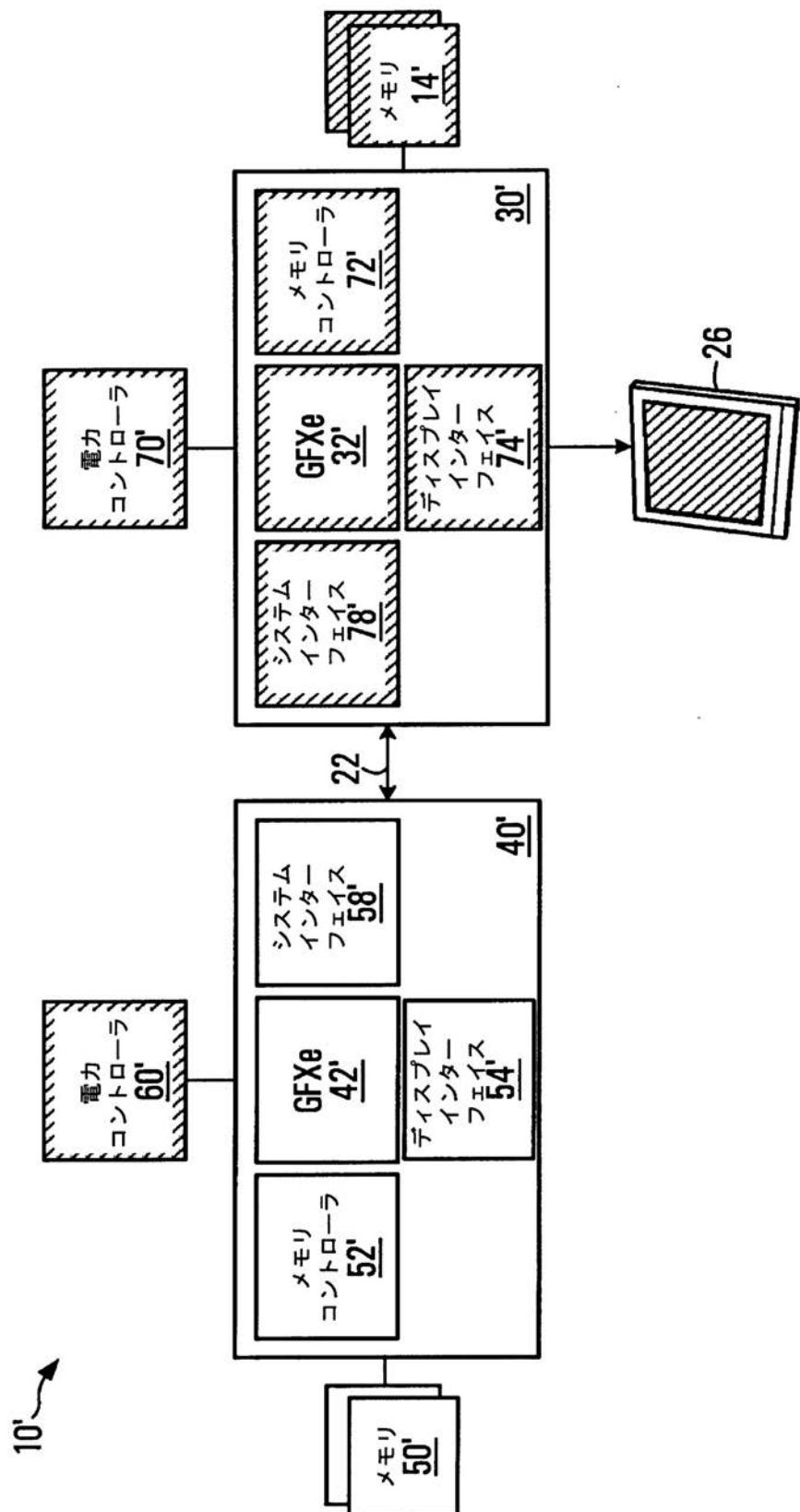


FIG. 11B

【図12】

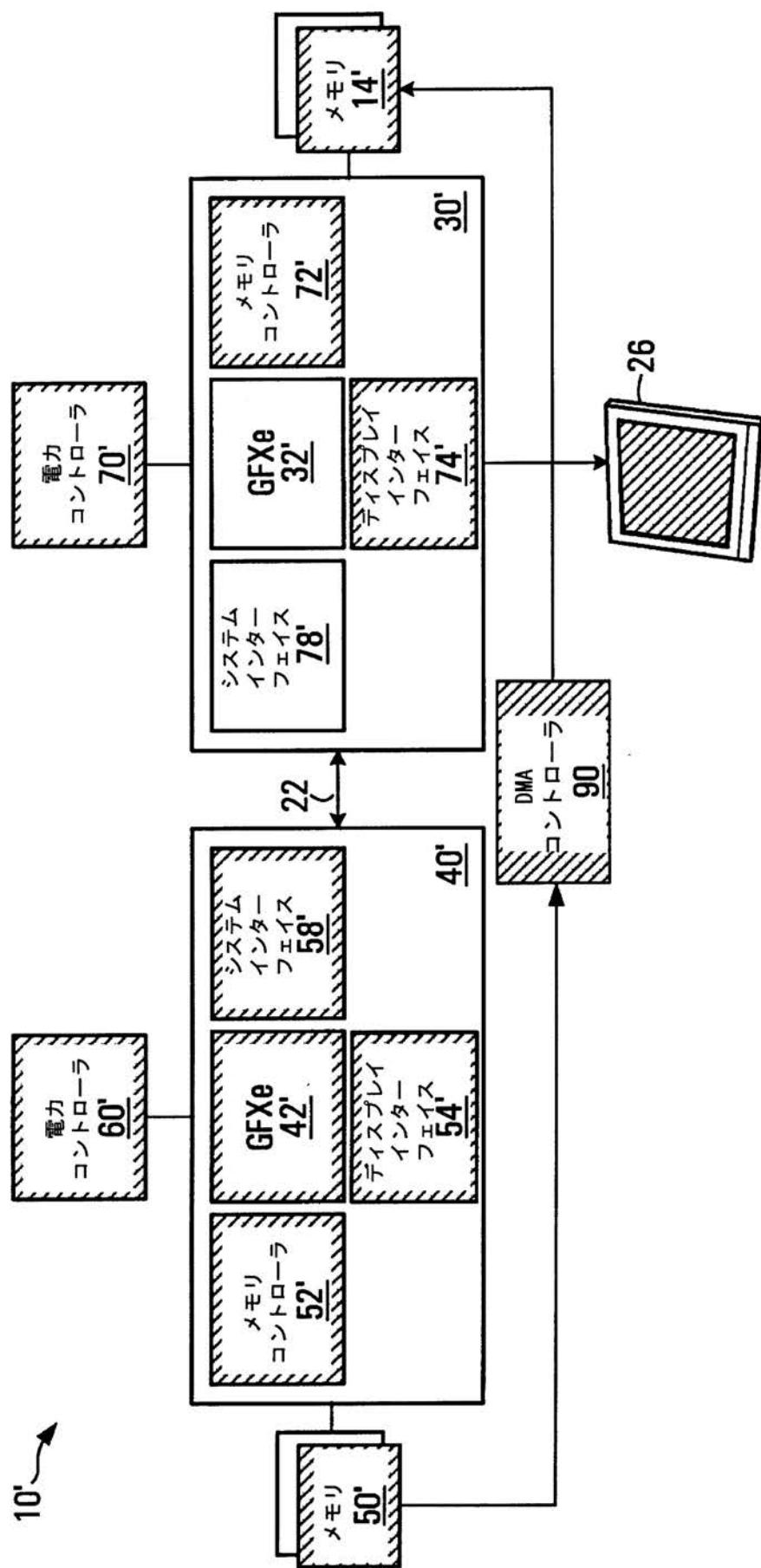


FIG. 12

## フロントページの続き

(51)Int.CI. F I  
 G 0 6 F 13/38 3 1 0 B  
 G 0 6 F 3/153 3 3 0 A

(74)代理人 100111615  
 弁理士 佐野 良太

(74)代理人 100132207  
 弁理士 太田 昌孝

(74)代理人 100162156  
 弁理士 村雨 圭介

(72)発明者 ジェフリー チェン  
 カナダ国 M 2 K 2 A 4 オンタリオ、トロント オスカーコート 8

(72)発明者 テリー ラヴィオレット  
 カナダ国 L 1 E 2 C 3 オンタリオ、コーティス オーパーンレーン 4 1

(72)発明者 ジェームス フアン  
 カナダ国 M 2 N 4 M 7 オンタリオ、ノースヨーク ホームズアベニュー 1 7 1 a

(72)発明者 ロバート ザブルジッキ  
 カナダ国 L 5 H 4 A 1 オンタリオ、ミシソーガ レインツリーレーン 1 0 1 9

(72)発明者 ジェイソン ロン  
 カナダ国 L 4 S 2 J 5 オンタリオ、リッチモンドヒル ファームステッドロード 1 8 5

(72)発明者 シアンチュエン ウェン  
 カナダ国 M 4 A 2 J 9 オンタリオ、トロント ウィンドクリフクレッセント 9 1

(72)発明者 ササ マリンコヴィック  
 カナダ国 M 6 B 4 B 5 オンタリオ、トロント マーリーアベニュー 1 5 5

(72)発明者 フィル ムンマー  
 アメリカ合衆国 9 4 0 6 3 カリフォルニア、レッドウッドシティ ピー.オー. ボックス  
 5 8 2 6

(72)発明者 ミンウェイ チェン  
 カナダ国 M 2 J 5 G 5 オンタリオ、トロント ゴッドストーンロード 7 6

(72)発明者 マイケル トレシッダー  
 カナダ国 L 3 X 1 L 6 オンタリオ、ニューマーケット マリナーレーン 6 6 0

(72)発明者 ルーメン サルチェフ  
 カナダ国 L 3 R 4 C 4 オンタリオ、マークハム ブキャナンドライブ 7 3

(72)発明者 ジョージ シエ  
 カナダ国 M 2 J 1 G 3 オンタリオ、ノースヨーク ジョージヘンリーブールバード 4 1 4  
 - 1 3 0

(72)発明者 ユーリ リチマノフ  
 カナダ国 L 4 C 5 N 2 オンタリオ、リッチモンドヒル ユニット 2 0 キャッスルロック  
 ドライブ 1 4 1

審査官 萩島 豪

(56)参考文献 米国特許出願公開第2005/0285863(US, A1)  
 米国特許第05896141(US, A)  
 米国特許出願公開第2006/0267987(US, A1)  
 特表2007-535042(JP, A)  
 特開平11-109937(JP, A)  
 特開平03-233780(JP, A)

(58)調査した分野(Int.Cl., DB名)

G 0 6 F 3 / 1 4  
G 0 6 F 3 / 0 4 8  
G 0 6 F 3 / 1 5 3  
G 0 6 F 1 3 / 3 8  
G 0 9 G 5 / 0 0