

(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2007-122369  
(P2007-122369A)

(43) 公開日 平成19年5月17日(2007.5.17)

(51) Int. Cl.

F I

テーマコード (参考)

G O 6 F 9/52 (2006.01)

G O 6 F 9/46 4 7 5 A

5 B O 5 7

G O 6 T 1/20 (2006.01)

G O 6 T 1/20 B

審査請求 未請求 請求項の数 13 O L (全 14 頁)

(21) 出願番号	特願2005-313096 (P2005-313096)	(71) 出願人	000001007
(22) 出願日	平成17年10月27日 (2005.10.27)		キヤノン株式会社
			東京都大田区下丸子3丁目30番2号
		(74) 代理人	100090273
			弁理士 國分 孝悦
		(72) 発明者	相澤 栄治
			東京都大田区下丸子3丁目30番2号 キ
			ヤノン株式会社内
		F ターム (参考)	5B057 AA20 CA01 CA08 CA12 CA17
			CB01 CB08 CB12 CB16 CC01
			CE17 CF10 CG05

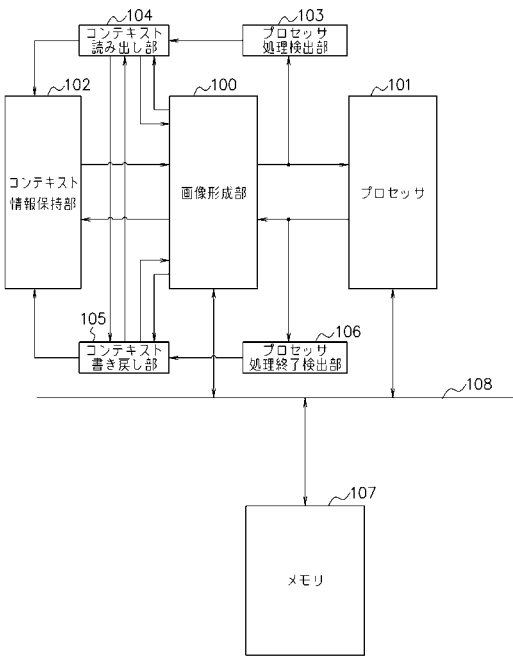
(54) 【発明の名称】 データ処理装置及びデータ処理方法

(57) 【要約】

【課題】ハード・ソフト間で同期処理が必要なタスク処理を行うデータ処理において、ソフトウェア処理に係るハードウェア処理の待ち時間を削減しスループットを改善する。

【解決手段】論理回路で構成された画像形成部100で実行中の処理にてプロセッサ101での処理が発生した場合に、その処理に係るコンテキスト情報をコンテキスト情報保持部102に退避して、プロセッサ処理終了まで他の処理を画像形成部で実行し、プロセッサ処理終了時に退避したコンテキスト情報と画像形成部で実行中のコンテキスト情報とを切り替えるようにして、画像形成部で実行中の処理にてプロセッサ処理が発生しても、プロセッサ処理と並行して画像形成部で他の処理が実行でき、画像形成部の待ち時間を削減してスループットを改善できるようにする。

【選択図】 図1



**【特許請求の範囲】****【請求項 1】**

論理回路による処理と、プロセッサでのソフトウェアによるプロセッサ処理との同期処理により複数のタスクに係る処理を行うデータ処理方法であって、

前記論理回路で実行中のタスクにてプロセッサ処理が発生した場合に、前記論理回路で実行中のタスクに係るタスク情報を退避して、前記プロセッサ処理終了までは前記タスク情報を退避したタスクとは異なる他のタスクを前記論理回路で実行し、前記プロセッサ処理終了時に退避したタスク情報と前記論理回路で実行中の前記他のタスクに係るタスク情報とを切り替えることを特徴とするデータ処理方法。

**【請求項 2】**

前記論理回路から前記プロセッサへのプロセッサ処理依頼を契機として、当該プロセッサ処理の発生時に前記論理回路で実行中のタスクに係るタスク情報を退避し、

前記プロセッサから前記論理回路へのプロセッサ処理の結果の通知を契機として、退避したタスク情報と前記論理回路で実行中の前記他のタスクに係るタスク情報とを切り替えることを特徴とする請求項 1 記載のデータ処理方法。

**【請求項 3】**

実行される複数のタスクに優先順位を付し、前記プロセッサ処理終了時に退避したタスク情報のタスクが前記論理回路で実行中の前記他のタスクより優先順位が高い場合のみタスク情報の切り替えを行うことを特徴とする請求項 1 又は 2 記載のデータ処理方法。

**【請求項 4】**

前記論理回路にて実行を開始する順に前記複数のタスクに優先順位を付すことを特徴とする請求項 3 記載のデータ処理方法。

**【請求項 5】**

論理回路及びプロセッサを有し、前記論理回路と前記プロセッサ間で同期をとりながら複数のタスクを逐次処理するデータ処理装置であって、

前記論理回路で実行中のタスクに係るタスク情報を読み出す情報読み出し手段と、

前記情報読み出し手段により読み出されたタスク情報を保持する情報保持手段と、

前記情報保持手段に保持されているタスク情報を前記論理回路に書き出す情報書き出し手段と、

前記プロセッサでのソフトウェアによるプロセッサ処理が発生したことを検出する処理検出手段と、

前記プロセッサ処理が終了したことを検出する処理終了検出手段とを備え、

前記論理回路で実行中のタスクにてプロセッサ処理が発生した場合に、前記論理回路で実行中のタスク情報を退避して、前記プロセッサ処理終了までは前記論理回路で他のタスクを実行し、前記プロセッサ処理終了時に退避したタスク情報と前記論理回路で実行中の前記他のタスクに係るタスク情報と切り替えることを特徴とするデータ処理装置。

**【請求項 6】**

前記論理回路は、パイプライン構造を有することを特徴とする請求項 5 記載のデータ処理装置。

**【請求項 7】**

前記論理回路は、画像形成処理を行う画像形成回路であることを特徴とする請求項 5 又は 6 記載のデータ処理装置。

**【請求項 8】**

前記処理検出手段は、前記論理回路と前記プロセッサを接続する信号線を介して伝送される信号を監視して前記プロセッサ処理の発生を検出することを特徴とする請求項 5 ~ 7 の何れか 1 項に記載のデータ処理装置。

**【請求項 9】**

前記処理検出手段は、前記プロセッサ上で実行されるソフトウェアにより実装されることを特徴とする請求項 5 ~ 7 の何れか 1 項に記載のデータ処理装置。

**【請求項 10】**

10

20

30

40

50

前記処理終了検出手段は、前記論理回路と前記プロセッサを接続する信号線を介して伝送される信号を監視することで前記プロセッサ処理の終了を検出することを特徴とする請求項 5 ～ 9 の何れか 1 項に記載のデータ処理装置。

【請求項 1 1】

前記処理終了検出手段は、前記プロセッサ上で実行されるソフトウェアにより実装されることを特徴とする請求項 5 ～ 9 の何れか 1 項に記載のデータ処理装置。

【請求項 1 2】

前記タスクは、前記論理回路により生成された一意に特定可能なタスク ID を有しており、

前記情報保持手段は、タスク情報とともにタスク ID を保持し、前記情報書き出し手段は、供給されるタスク ID に基づいて当該タスク ID に対応するタスク情報を前記論理回路に書き出すことを特徴とする請求項 5 ～ 1 1 の何れか 1 項に記載のデータ処理装置。 10

【請求項 1 3】

前記タスクは、前記論理回路を制御するソフトウェアにより生成された一意に特定可能なタスク ID を有しており、

前記情報保持手段は、タスク情報とともにタスク ID を保持し、前記情報書き出し手段は、供給されるタスク ID に基づいて当該タスク ID に対応するタスク情報を前記論理回路に書き出すことを特徴とする請求項 5 ～ 1 1 の何れか 1 項に記載のデータ処理装置。

【発明の詳細な説明】

【技術分野】

20

【0 0 0 1】

本発明は、ハードウェアとソフトウェアとで協調処理を行い複数のタスクを実行するデータ処理技術に関する。

【背景技術】

【0 0 0 2】

近年、LSI 開発の大規模化に伴い設計資産の IP (Intellectual Property) 化による再利用が進むとともに、高性能、高機能、かつ機能拡張可能な IP の必要性が高まっている。そのため、従来はハードウェアによって全体が構成されていた IP に代わり、ハードウェア IP に加えてプロセッサで実行されるソフトウェアにより構成されるソフトウェア IP を含んだハード・ソフト協調データ処理装置の開発が行われている。 30

【0 0 0 3】

一般的にハード・ソフト協調データ処理装置では、使用頻度が高い処理や高速処理が要求される処理等はハードウェア IP 部分での処理割合を多くする。それに対して、使用頻度が低い処理や下位互換性を維持するための処理等はソフトウェア IP 部分での処理割合を多くする。これにより、ハード・ソフト協調データ処理装置は、高性能、高機能、かつ高拡張化を実現している。また、特許文献 1 に記載されているようにソフトウェア IP のスレッドスケジューリングを効率化する技術が考えられている。

【0 0 0 4】

【特許文献 1】特開 2 0 0 4 - 2 5 2 9 8 3 号公報

【発明の開示】

40

【発明が解決しようとする課題】

【0 0 0 5】

ここで、ハード・ソフト協調データ処理装置において、ハード・ソフト間で同期処理が必要なタスク処理を行う場合について考える。通常、ソフトウェア処理（プロセッサ処理）は、ハードウェア処理に比べて、柔軟性及び拡張性には優れているが、処理時間が大幅に増加する。そのため、同期処理が必要なタスクをハード・ソフト協調データ処理装置でソフトウェア処理する場合、当該ソフトウェア処理が実行されると、ハードウェア処理は同期を取るためにソフトウェア処理が終了するまで待ち状態になりストールする。その結果、ハード・ソフト協調データ処理装置において、ハードウェア IP の性能が十分に発揮できずにスループットが低下し、システム要求性能を満たせないケースが発生する可能性 50

がある。

【 0 0 0 6 】

本発明は、このような事情に鑑みてなされたものであり、ハード・ソフト間で同期処理が必要なタスク処理を行うデータ処理装置にて、ソフトウェア処理に係るハードウェア処理の待ち時間を削減しスループットを改善することを目的とする。

【課題を解決するための手段】

【 0 0 0 7 】

本発明のデータ処理方法は、論理回路による処理と、プロセッサでのソフトウェアによるプロセッサ処理との同期処理により複数のタスクに係る処理を行うデータ処理方法であって、前記論理回路で実行中のタスクにてプロセッサ処理が発生した場合に、前記論理回路で実行中のタスクに係るタスク情報を退避して、前記プロセッサ処理終了までは前記タスク情報を退避したタスクとは異なる他のタスクを前記論理回路で実行し、前記プロセッサ処理終了時に退避したタスク情報と前記論理回路で実行中の前記他のタスクに係るタスク情報とを切り替えることを特徴とする。

10

本発明のデータ処理装置は、論理回路及びプロセッサを有し、前記論理回路と前記プロセッサ間で同期をとりながら複数のタスクを逐次処理するデータ処理装置であって、前記論理回路で実行中のタスクに係るタスク情報を読み出す情報読み出し手段と、前記情報読み出し手段により読み出されたタスク情報を保持する情報保持手段と、前記情報保持手段に保持されているタスク情報を前記論理回路に書き出す情報書き出し手段と、前記プロセッサでのソフトウェアによるプロセッサ処理が発生したことを検出する処理検出手段と、前記プロセッサ処理が終了したことを検出する処理終了検出手段とを備え、前記論理回路で実行中のタスクにてプロセッサ処理が発生した場合に、前記論理回路で実行中のタスク情報を退避して、前記プロセッサ処理終了までは前記論理回路で他のタスクを実行し、前記プロセッサ処理終了時に退避したタスク情報と前記論理回路で実行中の前記他のタスクに係るタスク情報と切り替えることを特徴とする。

20

【発明の効果】

【 0 0 0 8 】

本発明によれば、論理回路で実行中のタスクにてプロセッサ処理が発生した場合に、そのタスクに係るタスク情報を退避し、前記プロセッサ処理終了までは論理回路で他のタスクを実行する。そして、プロセッサ処理終了時に退避したタスク情報と論理回路で実行中の他のタスクに係るタスク情報とを切り替える。これにより、論理回路で実行中のタスクにてプロセッサ処理が発生しても、プロセッサ処理と並行して論理回路で他のタスクが実行でき、従来ストールしていた論理回路の待ち時間を削減してスループットを改善することができる。

30

【発明を実施するための最良の形態】

【 0 0 0 9 】

以下、本発明の実施形態を図面に基づいて説明する。

図 1 は、本発明の一実施形態によるデータ処理装置の一例である画像形成装置の構成例を示すブロック図である。

【 0 0 1 0 】

図 1 に示すように、本実施形態における画像形成装置は、画像形成部 1 0 0、プロセッサ 1 0 1、及びコンテキスト情報保持部 1 0 2 を有する。また、画像形成装置は、プロセッサ処理検出部 1 0 3、コンテキスト読み出し部 1 0 4、コンテキスト書き戻し部 1 0 5、プロセッサ処理終了検出部 1 0 6、及びメモリ 1 0 7 を有する。画像形成部 1 0 0、プロセッサ 1 0 1、及びメモリ 1 0 7 は、バス 1 0 8 を介して互いに通信可能に接続されている。

40

【 0 0 1 1 】

画像形成部 1 0 0 は、論理回路（ハードウェア）で構成されている。画像形成部 1 0 0 は、メモリ 1 0 7 から表示リストを読み出すとともに、その表示リストに ID を付与して新規表示リスト ID が生成されたことをコンテキスト書き戻し部 1 0 5 に通知する。画像

50

形成部 100 は、読み出した表示リストの中に書かれた命令に従って処理を実行する。このとき、画像形成部 100 は、表示リストの中に処理できない命令を検出すると、その処理の依頼をプロセッサ 101 に発行するとともにプロセッサ 101 での処理終了時に結果を受け取る。

【0012】

また、画像形成部 100 は、コンテキスト読み出し部 104 及びコンテキスト書き戻し部 105 に自身で実行中の表示リスト ID を供給する。

画像形成部 100 は、コンテキスト読み出し部 104 から発行される処理停止信号により処理を停止し処理無効化信号により情報を無効化して、実行中の表示リストのコンテキスト情報をコンテキスト情報保持部 102 に供給する。ここで、処理無効化信号により無効化される情報は、画像形成部 100 内部に保存されている処理結果に影響を及ぼす情報である。

【0013】

画像形成部 100 は、コンテキスト書き戻し部 105 からの指示に従ってコンテキスト情報保持部 102 に保持されている表示リストのコンテキスト情報を受け取り、実行する表示リストのコンテキスト情報として内部レジスタに設定する。さらに、画像形成部 100 は、表示リスト処理終了後に表示リストに付与した表示リスト ID を削除し、削除した表示リスト ID をコンテキスト書き戻し部 105 に通知する。

【0014】

プロセッサ 101 は、画像形成部 100 からの処理依頼を受け取り、その処理内容を解析する。プロセッサ 101 は、解析結果に応じて、必要なデータをメモリ 107 から取得して処理を実行する。なお、プロセッサ 101 での処理（プロセッサ処理）は、プロセッサ 101 が当該処理に応じたプログラム（ソフトウェア）を実行することにより実現される。そして、プロセッサ 101 は、処理依頼に応じた処理が終了すると処理結果を画像形成部 100 に発行する。

【0015】

コンテキスト情報保持部 102 は、画像形成部 100 から供給される、実行中のタスクに係るタスク情報であるコンテキスト情報を保持する。コンテキスト情報保持部 102 は、コンテキスト読み出し部 104 から供給される書き込み制御信号に従って、画像形成部 100 から供給される実行中の表示リストのコンテキスト情報を内部に保存する。また、コンテキスト情報保持部 102 は、コンテキスト書き戻し部 105 から供給される読み出し制御信号に従って、内部に保存しているコンテキスト情報を画像形成部 100 に供給する。

【0016】

プロセッサ処理検出部 103 は、画像形成部 100 からプロセッサ 101 に処理の依頼が発生したことを検出する。プロセッサ処理検出部 103 は、画像形成部 100 からプロセッサ 101 に処理の転送を行うための信号線の変化を監視し、処理の転送を検出するとプロセッサ 101 への処理依頼が行われたことをコンテキスト読み出し部 104 に通知する。

【0017】

コンテキスト読み出し部 104 は、画像形成部 100 からコンテキスト情報保持部 102 に供給されるコンテキスト情報の保存場所を指示するための信号を生成する。コンテキスト読み出し部 104 は、実行中の表示リスト ID を画像形成部 100 から受け取るとともに、プロセッサ 101 への処理依頼が発行されたことを通知する信号をプロセッサ処理検出部 103 から受信する。

【0018】

また、コンテキスト読み出し部 104 は、コンテキスト読み出し指示信号をコンテキスト書き戻し部 105 から受信する。このコンテキスト読み出し指示信号を受信した場合に限り、コンテキスト読み出し部 104 は、コンテキスト情報の読み出しが終了した後でコンテキスト読み出し終了信号をコンテキスト書き戻し部 105 に発行する。

10

20

30

40

50

## 【 0 0 1 9 】

また、コンテキスト読み出し部 1 0 4 は、処理停止を指示する処理停止信号、及び処理無効化を指示する処理無効化信号を画像形成部 1 0 0 に発行する。また、コンテキスト読み出し部 1 0 4 は、画像形成部 1 0 0 からコンテキスト情報保持部 1 0 2 に供給されるコンテキスト情報の保存場所を指示する信号を画像形成部 1 0 0 に発行する。

## 【 0 0 2 0 】

コンテキスト書き戻し部 1 0 5 は、コンテキスト情報保持部 1 0 2 から画像形成部 1 0 0 へのコンテキスト情報の書き戻しを制御する。コンテキスト書き戻し部 1 0 5 は、画像形成部 1 0 0 から供給される新規表示リスト ID、終了表示リスト ID、及び現在実行中の表示リスト ID を受信し、表示リストに優先順位付けを行う。

10

## 【 0 0 2 1 】

コンテキスト書き戻し部 1 0 5 は、プロセッサ処理終了検出部 1 0 6 から発行されるプロセッサ処理が終了した表示リスト ID を受信し、画像形成部 1 0 0 で実行中の表示リストとプロセッサ 1 0 1 での処理が終了した表示リストの優先順位を比較する。その結果、プロセッサ 1 0 1 での処理が終了した表示リストの優先順位が高い場合のみ、コンテキスト書き戻し部 1 0 5 は、コンテキスト読み出し指示信号をコンテキスト読み出し部 1 0 4 に発行する。コンテキスト書き戻し部 1 0 5 は、コンテキスト読み出し部 1 0 4 からコンテキスト読み出し終了信号の受信後、コンテキスト書き戻し指示信号を画像形成部 1 0 0 に発行し、読み出すコンテキスト情報を指示する信号をコンテキスト情報保持部 1 0 2 に発行する。

20

## 【 0 0 2 2 】

プロセッサ処理終了検出部 1 0 6 は、プロセッサ 1 0 1 から画像形成部 1 0 0 に処理結果が発行されたことを検出する。プロセッサ処理終了検出部 1 0 6 は、プロセッサ 1 0 1 と画像形成部 1 0 0 間の信号を監視し、処理結果の発行によりプロセッサ 1 0 1 での処理が終了したことを検出すると、処理結果に付与されていた表示リスト ID をコンテキスト書き戻し部 1 0 5 に発行する。

メモリ 1 0 7 は、処理する表示リスト、処理に必要なデータ、及び処理結果を保存する。

## 【 0 0 2 3 】

なお、上述した説明では、プロセッサ処理検出部 1 0 3 及びプロセッサ処理終了検出部 1 0 6 は、画像形成部 1 0 0 とプロセッサ 1 0 1 との間で信号線を介して授受される信号により処理依頼の発行及びその処理の終了を検出する。しかしながら、本発明はこれに限定されず、プロセッサ処理検出部 1 0 3 やプロセッサ処理終了検出部 1 0 6 をプロセッサ 1 0 1 で実行されるプログラム（ソフトウェア）により実装して処理依頼の発行及びその処理の終了を検出するようにしても良い。

30

## 【 0 0 2 4 】

ここで、画像形成部 1 0 0 は、オブジェクト・グラフィック記述から得られた表示リストを処理して画素を生成する。表示リストには、オリジナルのオブジェクト・グラフィック記述に含まれる描画オブジェクトが副走査線方向（y 座標）の昇順にソートされ記録されている。また、表示リストには、各オブジェクトの稜線情報が稜線テーブルとして、各稜線に対応づけられたレベル情報と他レベルの同一座標の画素との関係（演算方法）がレベルテーブルとして含まれる。さらに、表示リストには、オブジェクト内部の色を決定するための情報がフィルテーブルとして含まれる。なお、オブジェクトの色を決定するための情報としては、色を生成するためのデータや処理が指定されており、ビットマップの読み込みや圧縮画像を伸張し、その結果から任意の画素を取り込むことを指定している場合もある。

40

## 【 0 0 2 5 】

図 2 は、画像形成部 1 0 0 の画像形成に関する機能構成例を示すブロック図である。

図 2 において、画像形成部 1 0 0 は後述するモジュール 2 0 1 ~ 2 0 7 及び 2 1 0 ~ 2 1 3 から構成されている。画像形成部 1 0 0 は、パイプライン構造を有しており、モジュ

50

ール 201 ~ 206 を「パイプラインモジュール」とも称す。

【0026】

命令実行部 201 は、システムバスにアクセスして表示リストを読み込む。命令実行部 201 は、読み込んだ表示リスト中の命令ストリームを解釈して、後述するモジュール 202 ~ 205 からなる後段のパイプラインモジュールに対する内部コマンドを発行する。

【0027】

稜線処理部 202 は、命令実行部 201 により発行された内部コマンドに従って表示リストに含まれる稜線情報を読み取り、走査線単位で描画オブジェクトの稜線情報を抽出する。稜線処理部 202 は、抽出した稜線情報を走査線方向 (x 座標) の昇順にソートした後、パイプライン後段に位置するレベル優先度決定部 203 に該稜線情報をメッセージとして転送する。

10

【0028】

レベル優先度決定部 203 は、命令実行部 201 により発行された内部コマンドに従って表示リストに含まれるレベルテーブルと、稜線処理部 202 で生成された稜線情報とを読み込む。レベル優先度決定部 203 は、読み込んだレベルテーブル及び稜線情報を基に各走査線毎に各レベルの優先度と活性化された (描画に影響する) 画素範囲を決定する。さらに、レベル優先度決定部 203 は、各走査線の活性化された画素範囲の情報を優先度順にソートして他のレベルの画素との関係情報とともに画素範囲情報とし、パイプライン後段に位置する塗りつぶしデータ決定部 204 に該画素範囲情報を転送する。

【0029】

20

塗りつぶしデータ決定部 204 は、命令実行部 201 により発行された内部コマンドに従って表示リストに含まれるフィルテーブルと、レベル優先度決定部 203 で生成された画素範囲情報とを読み込み、それらを基にレベル毎に活性化された画素の色を決定する。塗りつぶしデータ決定部 204 は、該活性化された画素の色情報をレベル優先度決定部 203 から転送されてきた画素範囲情報とともに、パイプライン後段に位置する色合成部 205 に転送する。

【0030】

色合成部 205 は、命令実行部 201 により発行された内部コマンドに従って画素の最終的な色を生成する。色合成部 205 は、レベル優先度決定部 203 で生成された各レベルの画素範囲情報と、塗りつぶしデータ決定部 204 で決定された画素の色情報を基に画素単位に色を決定する演算を実行することにより画素の最終的な色を生成する。

30

【0031】

画素出力部 206 は、色合成部 205 により生成された最終画素を接続された外部機器に出力する。

バスアクセス調停部 207 は、稜線処理部 202、レベル優先度決定部 203、塗りつぶしデータ決定部 204、色合成部 205 によるシステムバスへのアクセスに対する調停や順序付けを行いシステムバスへアクセスを中継する。

【0032】

画像形成部 100 は、システムに要求される性能やコストに応じて、稜線処理部 202、レベル優先度決定部 203、塗りつぶしデータ決定部 204、及び色合成部 205 のそれぞれに対して拡張部を追加することが可能である。また、塗りつぶしデータ決定部 204 に対しては画像伸張部 210 を追加することも可能であり、この画像伸張部 210 にさらに拡張部を追加することが可能である。

40

【0033】

図 2 においては、塗りつぶしデータ決定部 204 に画像伸張部 210 を追加するとともに、稜線処理部 202 に対して拡張部 211、212 を追加し、画像伸張部 210 に対して拡張部 213 を追加している例を一例として示している。

【0034】

画像伸張部 210 は、画素色決定の際に圧縮画像を扱うことを可能にする、例えば J P E G (Joint Photographic Experts Group) をサポートする拡張モジュールである。稜線

50

処理拡張部 2 1 1 は、ポストスクリプト (PostScript) に対応する拡張モジュールであり、稜線処理拡張部 2 1 2 は、G D I (Graphics Device Interface) に対応する拡張モジュールである。画像伸張拡張部 2 1 3 は、J B I G (Joint Bi-level Image experts Group) に対応する拡張モジュールである。

#### 【 0 0 3 5 】

画像形成部 1 0 0 は、稜線処理部 2 0 2、レベル優先度決定部 2 0 3、塗りつぶしデータ決定部 2 0 4、及び色合成部 2 0 5 に実装されていない機能を要する内部コマンドが命令実行部 2 0 1 により発行されるときには、その処理をプロセッサ 1 0 1 に依頼できる。

例えば、稜線処理部 2 0 2 が基本機能として直線セグメントの処理機能を有しているとする、稜線処理部 2 0 2 は、拡張モジュール 2 1 1 及び 2 1 2 の処理機能を併せ持つことになるが、ベジエ曲線などのスプライン曲線の処理機能を有していない。命令実行部 2 0 1 が表示リスト中にベジエ曲線の処理を必要とする命令を検出すると、命令実行部 2 0 1 は稜線処理部 2 0 2 にベジエ曲線を処理する内部コマンドを発行する。この内部コマンドを受けた稜線処理部 2 0 2 は、プロセッサ 1 0 1 にベジエ曲線の処理を依頼し、プロセッサ 1 0 1 による実行結果を受け取った後に次の稜線の処理を実行する。

なお、稜線処理部 2 0 2 を一例として説明したが、レベル優先度決定部 2 0 3、塗りつぶしデータ決定部 2 0 4、及び色合成部 2 0 5 のそれぞれについても同様であり、実装されていない機能での処理をプロセッサ 1 0 1 に依頼することが可能である。

#### 【 0 0 3 6 】

図 3 は、本実施形態における画像形成装置を用いた一連の画像形成フローを説明するための図である。

#### 【 0 0 3 7 】

図 3 に示す画像形成フローは、オブジェクト・グラフィック記述 3 1 から始まり、まずそれが実行される。このオブジェクト・グラフィック記述 3 1 は、ホスト・プロセッサによって生成されるか、又はシステムメモリによって提供され、グラフィック・オブジェクトのパラメータを記述する。例えば、オブジェクト・グラフィック記述 3 1 には、いくつかのフォーマットで稜線を有するオブジェクトを組み込むことができる。このフォーマットには、ディスプレイ上の 1 点から別の点まで横断する直線の稜線 (単純ベクトル)、又は直交する線を含む複数の稜線によって二次元オブジェクトが定義される直交稜線フォーマットを含む。これ以外に、連続曲線によってオブジェクトが定義されるフォーマットも適当である。これらには、乗算を実行する必要なしに二次曲線を単一の出力空間内で画像形成できるようにする、いくつかのパラメータによって単一の曲線を記述できる二次多項式の線分を含めることができる。三次スプライン曲線や類似物などのそれ以外のデータ・フォーマットを使用することもできる。オブジェクトには、多数の異なる稜線タイプの混合物を含めることができる。通常、すべてのフォーマットに共通するのは、それぞれの線 (直線であれ曲線であれ) の始点と終点の識別子であり、これらは、スキャン・ライン番号によって識別され、その曲線を画像形成することのできる特定の出力空間が定義される。

#### 【 0 0 3 8 】

次に、画像形成されるグラフィック・オブジェクトの記述に必要なデータを識別し、表示リスト生成 3 2 が実行される。ここで、表示リスト生成 3 2 は、ホスト・プロセッサで実行されるソフトウェア・モジュールとして実施されることが好ましい。表示リスト生成 3 2 では、周知のグラフィック記述言語、グラフィック・ライブラリ呼出し、又は他のアプリケーション固有フォーマットのうちの 1 つあるいは複数で表現されたオブジェクト・グラフィック記述を表示リストに変換する。

#### 【 0 0 3 9 】

得られた表示リストは、通常、表示リスト・ストア 3 3 に書き込まれる。表示リスト・ストア 3 3 は、一般に R A M 内で形成されるが、その代わりに画像形成装置 3 4 がローカルに有するメモリ内で形成することもできる。表示リスト・ストア 3 3 には、複数の構成要素を含めることができ、その 1 つは命令ストリームであり、もう 1 つは稜線情報であり

10

20

30

40

50



、ラスト画像画素データを含めることができる。命令ストリームには、特定の画像内で所望される特定のグラフィック・オブジェクトを画像形成するために画像形成装置 34 によって読み取られる命令として解釈可能なコードが含まれる。

【0040】

表示リスト・ストア 33 は、画像形成装置 34 によって読み取られる。画像形成装置 34 は、表示リストをラスト画素のストリームに変換し、このストリームは画素出力 35 において、例えばプリンタ、ディスプレイ、又はメモリ・ストアなどの別の装置に転送することができる。

【0041】

図 4 は、本実施形態におけるコンテキスト情報保持部 102 の構成例を示す概略図である。 10

図 4 において、400 は表示リスト ID “0” 用の入力コンテキストのセクタである。セクタ 400 は、コンテキスト読み出し部 104 から発行される制御信号 C T L A に従い、画像形成部 100 から供給されるコンテキスト情報 C I I 又はレジスタファイル 410 の情報 R I 0 を選択し出力する。レジスタファイル 410 は、セクタ 400 により選択された表示リスト ID “0” 用コンテキスト情報を保持する。

【0042】

401 は表示リスト ID “1” 用の入力コンテキストのセクタであり、コンテキスト読み出し部 104 からの制御信号 C T L A に従い、画像形成部 100 から供給されるコンテキスト情報 C I I 又はレジスタファイル 411 の情報 R I 1 を選択し出力する。レジスタファイル 411 は、セクタ 401 により選択された表示リスト ID “1” 用コンテキスト情報を保持する。 20

【0043】

402 は表示リスト ID “2” 用の入力コンテキストのセクタであり、コンテキスト読み出し部 104 からの制御信号 C T L A に従い、画像形成部 100 から供給されるコンテキスト情報 C I I 又はレジスタファイル 412 の情報 R I 2 を選択し出力する。レジスタファイル 412 は、セクタ 402 により選択された表示リスト ID “2” 用コンテキスト情報を保持する。

【0044】

403 は表示リスト ID “3” 用の入力コンテキストのセクタであり、コンテキスト読み出し部 104 からの制御信号 C T L A に従い、画像形成部 100 から供給されるコンテキスト情報 C I I 又はレジスタファイル 413 の情報 R I 3 を選択し出力する。レジスタファイル 413 は、セクタ 403 により選択された表示リスト ID “3” 用コンテキスト情報を保持する。 30

【0045】

420 は出力コンテキスト情報のセクタである。セクタ 420 は、コンテキスト書き戻し部 105 から発行される制御信号 C T L B に従い、レジスタファイル 410、411、412、413 のうち何れか 1 つを選択して、それに保存されたコンテキスト情報を画像形成部 100 に対して発行する。

【0046】

なお、図 4 においては、4 つの表示リスト ID に対応可能なコンテキスト情報保持部 102 を一例として示しているが、本発明はこれに限定されるものではない。付与可能な表示リスト ID の数は任意であり、その数に応じて入力コンテキストのセクタとレジスタファイルとをコンテキスト情報保持部 102 が具備するように構成すれば良い。 40

【0047】

次に、本実施形態における画像形成装置の動作について説明する。なお、以下の説明では、表示リスト ID は最大で 4 つまで付与可能な構成であるとし、説明を簡単にするために 2 つの表示リストのスイッチ（切替）動作についてのみ説明する。

【0048】

まず、画像形成部 100 は、バス 108 を介してメモリ 107 から表示リストを読み出 50

すとともに、表示リスト毎に一意に識別可能なIDとして、読み出した表示リストに表示リストID“0”を付与する。そして、画像形成部100は、付与した新規表示リストIDをコンテキスト書き戻し部105に通知し、表示リストの中に書かれた命令に従って処理を実行する。

#### 【0049】

また、画像形成部100は、内部で実行中の表示リストの表示リストID“0”をコンテキスト読み出し部104及びコンテキスト書き戻し部105に絶えず供給し、実行中の表示リストのコンテキスト情報もコンテキスト情報保持部102に絶えず供給する。画像形成部100は、実行中の表示リストの中に画像形成部100では処理できない命令を検出した場合には、プロセッサ101に表示リストID“0”が付与された処理依頼を発行すると同時にこの表示リストの処理を中断する。 10

#### 【0050】

プロセッサ処理検出部103は、画像形成部100からプロセッサ101に処理の依頼が行われたことを検出すると、コンテキスト読み出し部104にプロセッサ101への処理依頼が発生したことを通知する。コンテキスト読み出し部104は、プロセッサ処理検出部103からの通知(信号)を受け取ると、画像形成部100で実行中の表示リストID“0”に係るコンテキスト情報の保存を指示する信号をコンテキスト情報保持部102に出力する。

#### 【0051】

コンテキスト情報保持部102は、コンテキスト読み出し部104から表示リストID“0”に係るコンテキスト情報の保存が指示されると、その保存場所であるレジスタファイル410に画像形成部100から常に供給されているコンテキスト情報を保存する。 20

#### 【0052】

画像形成部100は、表示リストID“0”の表示リストに係る処理中断後に、レジスタファイル411~413が使用されていないことを確認した後、メモリ107から新規表示リストの読み出しを行う。画像形成部100は、読み出した表示リストに表示リストID“1”を付与し、付与した新規表示リストID“1”をコンテキスト書き戻し部105に通知する。画像形成部100は、表示リストID“1”が付与された表示リストの中に書かれた命令に従って処理を実行する。このとき、画像形成部100は、内部で実行中の表示リストの表示リストID“1”をコンテキスト読み出し部104及びコンテキスト書き戻し部105に絶えず供給し、表示リストのコンテキスト情報をコンテキスト情報保持部102に絶えず供給している。 30

#### 【0053】

上述のようにして画像形成部100で新規に表示リストIDが生成された際、コンテキスト書き戻し部105は、画像形成部100から新規表示リストIDの通知を受けると表示リスト毎の優先順位付けを行うために内部で表示リストIDの管理を行う。この表示リストIDの管理は、ハードウェアにより行っても良いし、ソフトウェアにより行っても良い。本実施形態では、新規に生成された表示リストIDはFIFO(First In First Out)方式で管理され、生成されてから経過時間が大きい表示リストほど優先順位が高いものとする。また、本実施形態では表示リストIDは、新規表示リストの生成時に新たに生成され、表示リストの終了時に削除される。したがって、本実施形態において、表示リストID“0”は、表示リストID“1”よりも優先順位が高い。 40

#### 【0054】

プロセッサ101は、画像形成部100から受け取ったコマンド(処理依頼)の処理終了後に、表示リストID“0”を付与した処理結果を画像形成部100に発行する。

プロセッサ処理終了検出部106は、プロセッサ101から画像形成部100に処理結果が発行されたことを検出すると、処理結果に付与されている表示リストID“0”を読み出す。プロセッサ処理終了検出部106は、読み出した表示リストIDをコンテキスト書き戻し部105に発行する。

#### 【0055】

コンテキスト書き戻し部 105 は、プロセッサ処理終了検出部 106 から表示リスト ID を受信すると、受信した表示リスト ID と画像形成部 100 で実行中の表示リストの ID とを比較し優先順位を判定する。ここでは、コンテキスト書き戻し部 105 は、受信した表示リスト ID “0” が画像形成部 100 で現在実行中の表示リスト ID “1” の表示リストより優先順位が高いため、コンテキスト読み出し部 104 に読み出し指示信号を発行する。

【0056】

コンテキスト読み出し部 104 は、コンテキスト書き戻し部 105 より読み出し指示信号を受け取ると、画像形成部 100 に実行中の表示リスト ID “1” に係る処理停止を指示する処理停止信号を発行する。また、コンテキスト読み出し部 104 は、表示リスト ID “1” のコンテキスト情報を保存するための制御信号をコンテキスト情報保持部 102 に発行する。

10

【0057】

そして、コンテキスト読み出し部 104 は、コンテキスト情報の保存が終了した後、実行中の表示リストの処理無効化信号を画像形成部 100 に発行すると同時にコンテキスト読み出し終了信号をコンテキスト書き戻し部 105 に発行する。また、コンテキスト読み出し部 104 は、それとともに画像形成部 100 に対する処理停止信号の発行を止める。

【0058】

コンテキスト書き戻し部 105 は、コンテキスト読み出し部 104 からコンテキスト読み出し終了信号を受け取ると、コンテキスト情報保持部 102 に表示リスト ID “0” を発行するとともに、コンテキスト書き戻し指示信号を画像形成部 100 に発行する。

20

【0059】

画像形成部 100 は、コンテキスト書き戻し部 105 からコンテキスト書き戻し指示信号を受け取ると、コンテキスト情報保持部 102 から供給されるコンテキスト情報を内部にセットして表示リスト ID “0” の処理を継続する。

その後、画像形成部 100 にて表示リスト ID “0” の処理がすべて終了すると、画像形成部 100 は表示リスト ID “0” の削除をコンテキスト書き戻し部 105 に通知する。

【0060】

このように、画像形成部 100 には実装されていない処理機能での処理を行うために、画像形成部 100 からプロセッサ 101 に処理依頼を行った場合、画像形成部 100 は、コンテキストを切り替えて別の表示リストの処理を実行する。すなわち、画像形成部 100 が待ち状態となってストールすることなく、画像形成部 100 はプロセッサ 101 での処理と並行して別の表示リストの処理を行う。そして、プロセッサ 101 での処理終了後に、画像形成部 100 は、再びプロセッサ 101 に処理依頼を行ったコンテキストに切り替えることで元の表示リストの処理を継続する。これにより、画像形成部 100 の待ち時間を削減してスループットが改善される。

30

【0061】

本実施形態における画像形成装置のスループットを図 5 及び図 6 を参照して説明する。

図 5 は画像形成部 100 にてコンテキストスイッチ（コンテキストの切り替え）を行わない場合の処理時間を示しており、図 6 は画像形成部 100 にてコンテキストスイッチを行った場合の処理時間を示している。

40

【0062】

図 5 及び図 6 に示す処理の実行条件として、表示リスト A に係る処理は画像形成部 100 での処理に 12000 サイクル（cycle）、プロセッサ 101 での処理に 3000 サイクルが必要な処理である。また、表示リスト B に係る処理は画像形成部 100 での処理に 5000 サイクルが必要な処理であり、表示リスト C に係る処理は画像形成部 100 での処理に 10000 サイクルが必要な処理であるとする。また、コンテキストの切り替えに要する時間は、上述した各処理に要する時間に比べて非常に短いので、図 6 においては、説明を簡単にするためにコンテキストの切り替えに要する時間は省略している。

50

## 【 0 0 6 3 】

図 5 において、矢印 5 0 0 は、表示リスト A で画像形成部 1 0 0 からプロセッサ 1 0 1 に処理依頼が出力されたことを示しており、矢印 5 0 2 はプロセッサ 1 0 1 から画像形成部 1 0 0 に処理が終了したことを示す信号が出力されたことを示している。また、5 0 1 は表示リスト A のプロセッサ処理中に画像形成部 1 0 0 がストールしていることを示している。図 5 に示したように画像形成部 1 0 0 にてコンテキストスイッチを行わない場合には、表示リスト A ~ C に係る処理を実行するのに要するトータルの処理時間は 3 0 0 0 0 サイクルである。

## 【 0 0 6 4 】

図 6 において、矢印 6 0 0 は、表示リスト A で画像形成部 1 0 0 からプロセッサ 1 0 1 に処理依頼が出力されたことを示しており、矢印 6 0 2 はプロセッサ 1 0 1 から画像形成部 1 0 0 に処理が終了したことを示す信号が出力されたことを示している。また、6 0 1 は表示リスト A のプロセッサ処理中に画像形成部 1 0 0 で表示リスト B の処理を実行していることを示している。また、矢印 6 1 0 は表示リスト A のコンテキストの読み出し、矢印 6 1 2 は表示リスト B のコンテキストの読み出し及び表示リスト A のコンテキストの書き戻し、矢印 6 1 3 は表示リスト B のコンテキストの書き戻しをそれぞれ示している。6 1 1 は画像形成部 1 0 0 で表示リスト B の処理をしている間、表示リスト A のコンテキストをコンテキスト情報保持部 1 0 2 に退避させ保持していることを示している。したがって、図 6 に示したように画像形成部 1 0 0 にてコンテキストスイッチを行う場合には、トータルの処理時間は、図 5 に示した場合と比較して短縮されて 2 7 0 0 0 サイクルとなり、スループットを向上させることができる。

## 【 0 0 6 5 】

なお、上記実施形態は、何れも本発明を実施するにあたっての具体化のほんの一例を示したものに過ぎず、これらによって本発明の技術的範囲が限定的に解釈されてはならないものである。すなわち、本発明はその技術思想、またはその主要な特徴から逸脱することなく、様々な形で実施することができる。

## 【 図面の簡単な説明 】

## 【 0 0 6 6 】

【 図 1 】 本発明の実施形態における画像形成装置の構成例を示すブロック図である。

【 図 2 】 画像形成部の画像形成に関する機能構成例を示すブロック図である。

【 図 3 】 本実施形態における画像形成装置を用いた画像形成フローを説明するための図である。

【 図 4 】 コンテキスト情報保持部の構成例を示す概略図である。

【 図 5 】 コンテキストスイッチを行わない場合の処理時間を説明するための図である。

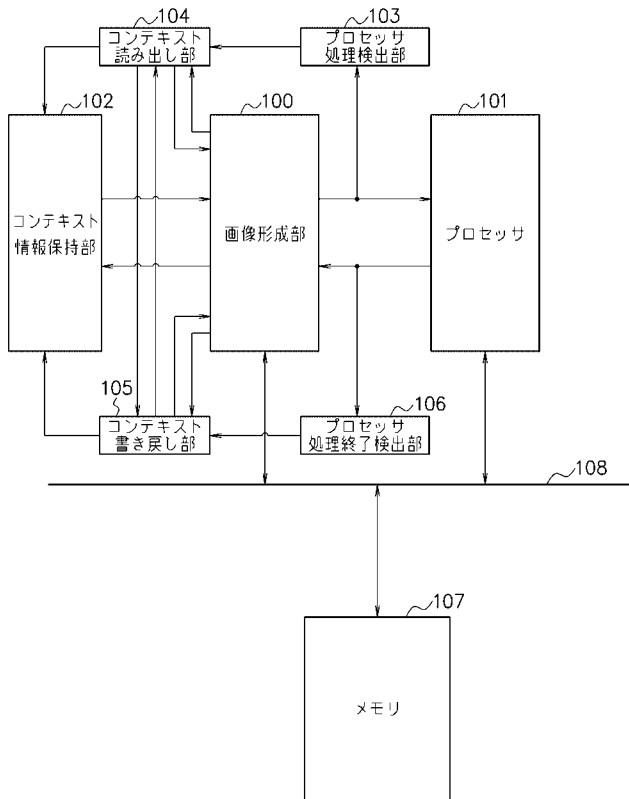
【 図 6 】 コンテキストスイッチを行った場合の処理時間を説明するための図である。

## 【 符号の説明 】

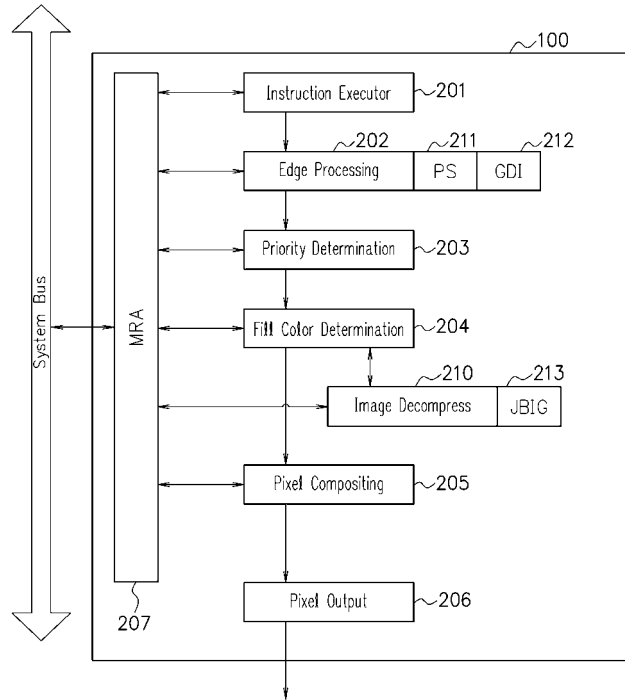
## 【 0 0 6 7 】

- 1 0 0 画像形成部
- 1 0 1 プロセッサ
- 1 0 2 コンテキスト情報保持部
- 1 0 3 プロセッサ処理検出部
- 1 0 4 コンテキスト読み出し部
- 1 0 5 コンテキスト書き戻し部
- 1 0 6 プロセッサ処理終了検出部
- 1 0 7 メモリ
- 1 0 8 バス

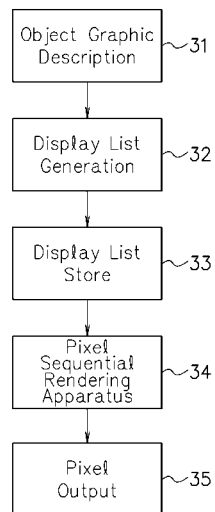
【図 1】



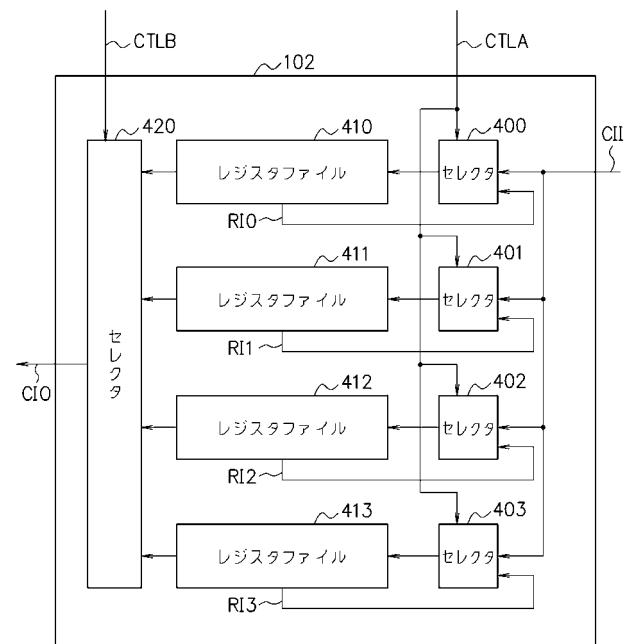
【図 2】



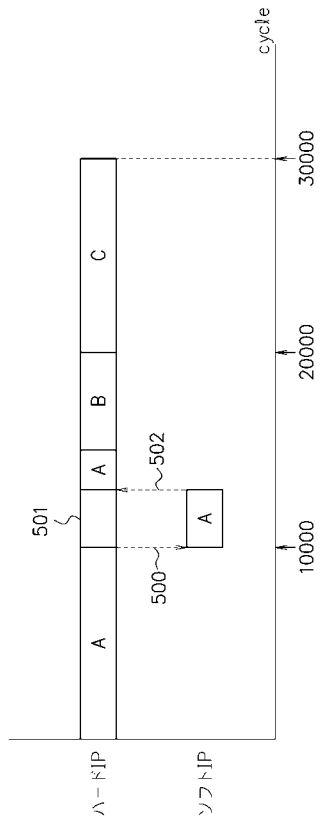
【図 3】



【図 4】



【図 5】



【図 6】

