

(19) 日本国特許庁(JP)

(12) 特 許 公 報(B2)

(11) 特許番号  
特許第6537265号  
(P6537265)

(45) 発行日 令和1年7月3日(2019.7.3)

(24) 登録日 令和1年6月14日(2019.6.14)

(51) Int.Cl.			F I		
B 4 1 J	5/30	(2006.01)	B 4 1 J	5/30	Z
G O 6 T	1/20	(2006.01)	G O 6 T	1/20	C
H O 4 N	1/00	(2006.01)	H O 4 N	1/00	C
H O 4 N	1/21	(2006.01)	H O 4 N	1/21	

請求項の数 10 (全 15 頁)

(21) 出願番号	特願2014-251339 (P2014-251339)	(73) 特許権者	000001007 キヤノン株式会社 東京都大田区下丸子3丁目30番2号
(22) 出願日	平成26年12月11日(2014.12.11)	(74) 代理人	100126240 弁理士 阿部 琢磨
(65) 公開番号	特開2016-112724 (P2016-112724A)	(74) 代理人	100124442 弁理士 黒岩 創吾
(43) 公開日	平成28年6月23日(2016.6.23)	(72) 発明者	駒野 雄亮 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
審査請求日	平成29年12月5日(2017.12.5)	審査官	大浜 登世子
		最終頁に続く	

(54) 【発明の名称】 画像処理装置、画像処理装置の制御方法

(57) 【特許請求の範囲】

【請求項 1】

画像処理を実行する第1画像処理回路と、前記第1画像処理回路の後段の第2画像処理回路とを含み、各画像処理回路が並行に画像処理可能な画像処理回路群と、

前記各画像処理回路の処理の進行状況を管理するシーケンサと、

メモリに記憶されたプログラムを実行するCPUと、

を有し、

前記CPUは、画像処理モードに基づいて、前記画像処理回路毎に処理単位として色数及び色毎の画素数を設定し、

所定の画像処理モードでは、前記第1画像処理回路と前記第2画像処理回路とは、前記設定された処理単位が異なり、

前記シーケンサは、前記第1画像処理回路の処理の進行状況に基づいて、前記CPUによる処理の介在なしに、前記第2画像処理回路に前記設定された処理単位の画像処理を開始させることを特徴とする画像処理装置。

【請求項 2】

前記シーケンサは、前記第1画像処理回路において、前記第2画像処理回路の処理単位分の処理が完了した場合、前記第2画像処理回路に画像処理の開始を指示することを特徴とする請求項1に記載の画像処理装置。

【請求項 3】

前記シーケンサは、前記第2画像処理回路の処理が完了した場合、前記第1画像処理回

路に処理済みのデータの削除を指示することを特徴とする請求項 2 に記載の画像処理装置。

【請求項 4】

前記シーケンサは、前記画像処理回路毎の処理済み色数の情報と処理済みの回数の情報を管理することを特徴とする請求項 1 ～ 3 のいずれか 1 項に記載の画像処理装置。

【請求項 5】

前記 CPU は、画像処理モード毎に各画像処理回路に設定する処理単位を切り替えることを特徴とする請求項 1 ～ 4 のいずれか 1 項に記載の画像処理装置。

【請求項 6】

前記 CPU は、各画像処理回路から出力される出力結果が使用するメモリの使用量の上限を決定することを特徴とする請求項 1 ～ 5 のいずれか 1 項に記載の画像処理装置。

10

【請求項 7】

前記シーケンサは、前記 CPU により決定された前記メモリの使用量の上限に基づいて、前記各画像処理回路の出力データをメモリに保持すべき状態であるアクティブ状態のキューが一定数を超えないよう管理することを特徴とする請求項 6 に記載の画像処理装置。

【請求項 8】

前記第 1 画像処理回路と前記第 2 画像処理回路とは異なる画像処理を実行することを特徴とする請求項 1 ～ 7 のいずれか 1 項に記載の画像処理装置。

【請求項 9】

前記画像処理装置は、印刷手段を備える印刷装置であることを特徴とする請求項 1 ～ 8 のいずれか 1 項に記載の画像処理装置。

20

【請求項 10】

第 1 画像処理回路と、前記第 1 画像処理回路の後段の第 2 画像処理回路とを含み各画像処理回路が並行に画像処理可能な画像処理回路群と、シーケンサと、メモリに記憶されたプログラムを実行する CPU と、を有する画像処理装置の制御方法であって、

前記シーケンサにより、前記各画像処理回路の処理の進行状況を管理する管理工程と、

前記 CPU により、画像処理モードに基づいて、前記画像処理回路毎に処理単位として色数及び色毎の画素数を設定する設定工程と、  
を備え、

所定の画像処理モードでは、前記第 1 画像処理回路と前記第 2 画像処理回路とは、前記設定された処理単位が異なり、

30

前記シーケンサからの指示に基づき、前記第 1 画像処理回路の処理の進行状況に基づいて、前記 CPU による処理の介入なしに、前記第 2 画像処理回路に前記設定された処理単位の画像処理が開始されることを特徴とする制御方法。

【発明の詳細な説明】

【技術分野】

【0001】

本発明は、画像データを処理する画像処理装置、画像処理装置の制御方法、及びプログラムに関する。

【背景技術】

40

【0002】

記録装置では、高速で高品位な画像形成が求められており、それには画像を形成するインク滴の色やドット径などの記録装置の特性に適した画像処理を高速に実施する必要がある。特許文献 1 では、複数の画像処理モジュールを用いて、画像処理を行う方法が開示されている。具体的には、画像処理装置は、複数の画像処理モジュールと、バッファメモリと、FIFOメモリとを有し、ある画像処理モジュールの処理結果である出力データを所定データ量の単位毎に順次バッファメモリを介して次段の画像処理モジュールに渡す。このとき、バッファメモリ内の出力データの格納位置を示すポインタは、FIFOメモリを介して次段の画像処理モジュールに渡される。

【先行技術文献】

50

## 【特許文献】

【0003】

【特許文献1】特開2001-157049

## 【発明の概要】

【発明が解決しようとする課題】

【0004】

ところで、高品位な画像形成をするために、単色毎や複数色間での画像処理、 $4 \times 4$ 画素格子単位での画像処理や $256 \times 256$ 画素格子単位での画像処理など種々の画像処理に対応することが求められている。しかしながら、特許文献1のように、データの格納位置をFIFOメモリにより指定して画像処理モジュール間でデータの受け渡しをする場合、それぞれの画像処理モジュールの処理単位が統一されていなければならない。例えば、単色 $4 \times 4$ 画素格子単位の処理で十分な画像処理モジュールでも、一連の画像処理のうちいずれかで複数色 $256 \times 256$ 画素格子単位の処理が必要であれば、複数色 $256 \times 256$ 画素格子単位の処理として回路を形成する必要がある、回路規模が大きくなってしまふという問題があった。また、画像処理モジュール間でのデータの受け渡し単位を本来の最適な単位より大きい単位で受け渡すことになり、処理速度が低下してしまうという問題や使用するメモリ量が増加してしまうという問題があった。

10

【0005】

そこで、本発明では、画像処理モジュール群が有する各画像処理モジュールの回路規模を大きくすることなく、高速で高品位な画像処理を実現することを目的とする。

20

【課題を解決するための手段】

【0006】

上記課題を解決するために、本発明の画像処理装置は、画像処理を実行する第1画像処理回路と、前記第1画像処理回路の後段の第2画像処理回路とを含み、各画像処理回路が並行に画像処理可能な画像処理回路群と、前記各画像処理回路の処理の進行状況を管理するシーケンサと、メモリに記憶されたプログラムを実行するCPUと、を有し、前記CPUは、画像処理モードに基づいて、前記画像処理回路毎に処理単位として色数及び色毎の画素数を設定し、所定の画像処理モードでは、前記第1画像処理回路と前記第2画像処理回路とは、前記設定された処理単位が異なり、前記シーケンサは、前記第1画像処理回路の処理の進行状況に基づいて、前記CPUによる処理の介在なしに、前記第2画像処理回路に前記設定された処理単位の画像処理を開始させる。

30

【発明の効果】

【0007】

本発明によれば、画像処理モジュール群が有する各画像処理モジュールの回路規模を大きくすることなく、高速で高品位な画像処理を実現することができる。

【図面の簡単な説明】

【0008】

【図1】実施形態1に係る画像処理装置の概略図である。

【図2】実施形態1に係る画像処理モードAでのシーケンサ処理の概念図である。

【図3】実施形態1に係る画像処理モードBでのシーケンサ処理の概念図である。

40

【図4】実施形態1に係る画像処理モードAと画像処理モードBの中間バッファ量を示す図である。

【図5】実施形態1に係るシーケンサ構成の概念図である。

【図6】実施形態1に係るシーケンサから画像処理モジュールへの処理開始通知時のフロー図である。

【発明を実施するための形態】

【0009】

(実施形態1)

以下、本発明の実施形態について図面を用いて説明する。なお、この実施の形態で用いる装置の各構成要素の相対配置、装置形状等は、あくまで例示であり、これらのみに限定

50

するものではない。

【 0 0 1 0 】

図 1 は、本発明の一実施形態に係る画像処理装置の概略図である。なお、図 1 では、印刷機能を有するプリンタを例に挙げて説明するが、画像処理装置は、これに限定されず、例えば、印刷機能を有していなくてもよいし、印刷機能及び読取機能を有する複合プリンタであってもよい。

【 0 0 1 1 】

図 1 に示すように、画像処理装置は、入力 I / F 1 0 0 と、CPU 1 0 1 と、ROM 1 0 2 と、複数の画像処理モジュール ( 1 0 3 ~ 1 0 7 ) と、メモリ 1 0 8 と、シーケンサ 1 0 9 と、プリントヘッド制御部 1 1 5 と、プリントヘッド 1 1 6 とを備える。

10

【 0 0 1 2 】

入力 I / F 1 0 0 は、パーソナルコンピュータ、デジタルカメラ等の外部装置から印刷画像データ等の各種入力データを受信するための I / F である。

【 0 0 1 3 】

CPU 1 0 1 は、中央演算処理部であり、プログラムの実行やハードウェアの起動により画像処理装置全体の動作を制御する。

【 0 0 1 4 】

ROM 1 0 2 は、CPU 1 0 1 が実行する各種制御プログラムや画像処理装置の各種動作に必要な固定データを格納する。例えば、画像処理装置の記録 ( 印刷 ) 処理を実行するプログラムを記憶する。

20

【 0 0 1 5 】

メモリ 1 0 8 は、CPU 1 0 1 等を動作させるために必要であり、CPU 1 0 1 のワークエリアとして用いられ、種々の受信データの一時格納領域として用いられ、各種設定データを記憶させたりする。本実施形態では、後述するように、メモリ 1 0 8 は、画像処理モジュールの出力結果を一時的に格納する。なお、図 1 では、メモリ 1 0 8 を一つとしたが、これに限定されるものではない。例えば、DRAM と SRAM のように処理速度の異なる複数のメモリからなるものとしてもよい。本実施形態では、メモリ 1 0 8 は、DRAM 及び SRAM を有するものとする。DRAM は、SRAM より低コストであり、SRAM は、DRAM よりもデータの読み出し及び書き出しを高速に実行することができる。ここでは、DRAM は、画像処理モジュール群へ入力するデータ及び画像処理モジュール群から出力するデータ、すなわち、画像処理モジュール A 1 0 3 に入力するデータと、画像処理モジュール E が出力するデータと、を格納するものとする。入力 I / F が受信した入力データ ( 画像データ ) は、画像処理モジュール群へ入力され、画像処理モジュール群から出力するデータは、プリントヘッド制御部 1 1 5 へ出力される。また、SRAM は、画像処理モジュール群の内部で入出力するデータ ( 画像処理モジュール A 1 0 3 から出力するデータから画像処理モジュール E 1 0 7 へ入力するデータまで ) を格納する中間バッファとして用いる。

30

【 0 0 1 6 】

シーケンサ 1 0 9 は、複数のキュー管理部 ( 1 1 0 ~ 1 1 4 ) を有し、各画像処理モジュールに対して処理の開始を通知する。より具体的には、シーケンサ 1 0 9 は、各画像処理モジュールの処理状況を複数のキュー管理部により管理し、各画像処理モジュールの処理の実行 ( 開始 ) を制御する。本実施形態では、シーケンサ 1 0 9 は、キュー管理部 A 1 1 0、キュー管理部 B 1 1 1、キュー管理部 C 1 1 2、キュー管理部 D 1 1 3、キュー管理部 E の 5 つのキュー管理部を有し、各キュー管理部が各画像処理モジュールに対して処理開始を通知する。なお、シーケンサ 1 0 9 が備えるキュー管理部の数はこれに限定されず、5 未満であってもよいし、6 以上であってもよい。

40

【 0 0 1 7 】

画像処理モジュール A 1 0 3 ~ 画像処理モジュール E 1 0 7 は、詳細については後述するが、それぞれ異なる画像処理を行う。このように、画像処理装置は、画像処理を実行する画像処理モジュールを複数含む。本実施形態では、画像処理装置が備える複数の画像処

50

理モジュールをまとめて、画像処理モジュール群という。

【 0 0 1 8 】

プリントヘッド 1 1 6 は、画像処理モジュール群で生成された印刷データに基づき被記録媒体に印刷を行う。プリントヘッド 1 1 6 は、各種印刷動作を実行するプリント部に含まれる。本実施形態では、プリントヘッド 1 1 6 は、インクを吐出することにより印刷動作を行う。また、プリントヘッド制御部 1 1 5 は、図示しないモータドライバ等を駆動させることにより、プリントヘッド 1 1 6 を印刷位置まで移動させたりして、印刷動作を制御する。

【 0 0 1 9 】

上述した画像処理装置は、入力 I / F 1 0 0 を介して外部装置から入力データ（画像データ）が入力されて、メモリ 1 0 8 に一旦格納される。その後、シーケンサ 1 0 9 により起動された画像処理モジュール A 1 0 3 ~ 画像処理モジュール E 1 0 7 は、それぞれメモリ 1 0 8 からデータを読み出して画像処理を実行し、画像処理後のデータを、再度メモリ 1 0 8 に書き込む。ここで、各画像処理モジュールは、画像処理モジュール毎に存在する各キュー管理部から処理開始の通知を受けることで、処理が開始される。そして、画像処理モジュールの最後段である画像処理モジュール E 1 0 7 がメモリ 1 0 8 に出力する画像処理済みのデータは、モータ駆動タイミングに合わせて駆動するプリントヘッド制御部 1 1 5 により、吐出データとしてプリントヘッド 1 1 6 に転送される。転送された画像処理済みのデータに基づいて、プリントヘッド 1 1 6 が記録媒体に対してインク滴が吐出することで、記録媒体上に画像が形成される。

【 0 0 2 0 】

ここで、各画像処理モジュールが実行する画像処理について説明する。まず、メモリ 1 0 8 に、初期データとして、6 0 0 d p i 単位の多値（4 b i t）データが色毎に入力される。これに対し、画像処理モジュール A 1 0 3 ~ 画像処理モジュール E 1 0 7 を含む画像処理モジュール群は、以下の一連の画像処理を行う。6 0 0 d p i 単位の多値データを最大 4 色単位で、1 2 0 0 d p i 単位の 2 値データへ変換する変換処理と、一定濃度以下のデータへ変換する変換処理を行う。また、データのうち文字画素の周囲を間引く変換処理と、画像の向きを 9 0 度回転させる変換処理と、プリントヘッドの吐出順にデータを並び替える変換処理を行う。なお、画像処理モジュール群が実行する一連の画像処理は、上述したものに限定されるものではなく、他の画像処理の機能を実行させるようにしてもよい。

【 0 0 2 1 】

上述した画像処理モジュールの個々の動作について説明する。

【 0 0 2 2 】

まず、画像処理モジュール A 1 0 3 は、初期データとしてラスタデータ（6 0 0 d p i 単位の多値データ）を色毎に読み出し、D R A M からのデータ読み出し効率に適した単位（本実施形態では、1 6 × 1 6 画素格子単位）にパッキングしてメモリ 1 0 8 に出力する。

【 0 0 2 3 】

次に、画像処理モジュール B 1 0 4 は、画像処理モジュール B 1 0 4 に入力されたデータをプリントヘッド 1 1 6 の吐出解像度へ変換する変換処理を行い、メモリ 1 0 8 に出力する。具体的には、画像処理モジュール B 1 0 4 は、パッキングされた 6 0 0 d p i 単位の多値データを吐出解像度 1 2 0 0 d p i 単位の 2 値データに変換してメモリ 1 0 8 に出力する。このときの変換単位は、画像のテクスチャ現象を抑えられるサイズであることがよく、本実施形態では、6 0 0 d p i 単位の 1 6 × 1 6 画素格子を色毎に処理するものとした。

【 0 0 2 4 】

次に、画像処理モジュール C 1 0 5 は、プリントヘッド 1 1 6 のスキャン間のつなぎ部分におけるムラを抑制するため、予め決めた濃度以上（閾値以上）となっている画素の 2 値データを間引く処理を行い、メモリ 1 0 8 に出力する。本実施形態では、画素毎に全色の 2 値データの重なり関係を調べ、ムラとして視認される閾値以上の濃度になっている画

10

20

30

40

50

素の2値データは、その閾値未満となるようにデータを間引き、再度メモリ108に出力する。処理単位は、各画素の濃度を把握するのに適した単位とするのがよく、本実施形態では、1200dpi単位の32×32画素格子を全色一度に処理するものとした。

【0025】

次に、画像処理モジュールD106は、文字の視認性を上げるため、各画像データに対して文字と認識した画素（文字画素ともいう）の周囲画素のデータを間引いた後、プリントヘッド116の配列方向に並び替えて再度メモリ108に出力する。ここでは、プリントヘッド116の配列方向に並べ替える処理として、データの90度回転処理を行う。文字画素の周囲は、1200dpi単位で2画素分を間引くものとする場合、最小処理単位は5×5画素格子となるが、ハードウェア処理しやすいように2のべき乗にすると、8×8画素格子となる。しかしながら、この文字画素の周囲の間引きには処理時間を要するため、本実施形態では、横方向の処理は一度に32画素分を処理するものとして、画像処理モジュールD106では32×8画素格子を色毎に処理するものとした。

【0026】

最後に、画像処理モジュールE107は、プリントヘッド116の吐出順にデータを並び替える変換処理を実行して、メモリ108に出力する。この処理単位は、DRAMへのデータ書き出し効率に適した単位とするのがよく、本実施形態では、1200dpi単位で32×32画素格子を色毎に処理するものとした。

【0027】

上述したように、本実施形態では、画像処理モジュール毎に、処理単位とする画素数（画素の格子単位）を異なるものとする。

【0028】

図2を用いて、本実施形態に係るシーケンサ109の処理について説明する。図2は、シーケンサ109の処理の概念図であり、ある時間における画像処理モジュール群に対するキュー管理部の処理遷移を示している。図2において、1つのキューは、各キュー管理部が管理対象とする画像処理モジュールの1回分の処理状態を示している。その処理状態は、以下の5つに分類する。処理開始の条件が整っている状態を「処理待ち」、実際に処理をしている状態を「処理中」、処理完了してメモリ108に保持される出力結果を次段の画像処理モジュールで使用されることを待っている状態を「処理済み」、とする。また、次段の画像処理モジュールでの使用が完了し出力結果が不要である状態を「削除済み」、メモリ108に保持できる出力結果の許容量に達して次段の画像処理モジュールでの使用完了を待っている状態を「削除待ち」とする。

【0029】

そして、この分類の中で「処理待ち」「処理中」「削除待ち」となっている状態では、画像処理モジュールの出力結果をメモリ108に保持すべき状態となっている。そのためこれらの3つの状態をアクティブ状態と呼ぶ。保持できるメモリ量は有限であるため、各キュー管理部は、このアクティブ状態の総数が一定以上にならないよう管理する。本実施形態では、画像処理装置は、複数の画像処理モードを実行することができるものとする。なお、画像処理モジュール群が実行する画像処理モードは、画像処理モジュール群に実行させる画像処理内容に応じてCPU101が設定する。CPU101は、各画像処理モジュールに処理単位とする画素数及び処理内容を指定する。

【0030】

図2を用いて、画像処理モードAについて説明する。図2に示す画像処理モードは、色数としてカラー0～3までの4色を処理するものとし、カラー0はCyan、カラー1はMagenta、カラー2はYellow、カラー3はBlackとする。また、この画像処理モードでは、一連の画像処理機能の中で文字画素の周囲を間引くデータ変換は実施しない。画像処理モジュールA103で1色分の600dpi多値データを16×16画素格子にパッキングする処理を1回終わると、画像処理モジュールB104がその1色分のパッキングされたデータを1200dpi単位の2値データで32×32画素格子に変換する処理を1回実施する。この画像処理モジュールA103と画像処理モジュールB1

10

20

30

40

50

04の処理を各色1回ずつ、すなわち、4色の処理を終えると、画像処理モジュールC105が32×32画素格子(1200dpi)のデータを4色一度に一定濃度以下のデータへ変換する。画像処理モジュールC105がその1回の処理を終えると、画像処理モジュールD106は、32×32画素格子(1200dpi)を横方向に4分割し、32×8画素格子(1200dpi)で各色毎に回転処理、すなわち、4回の回転処理を実行する。そして画像処理モジュールD106が各色4回の処理を終えると、画像処理モジュールE107は32×32画素格子(1200dpi)単位でヘッド吐出順への変換処理を各色毎に実施するものとする。

#### 【0031】

また、各画像処理モジュールからの出力結果の総量がメモリ許容量を超えることがないように、CPU101は、各画像処理モジュールから出力される出力結果が使用するメモリ108の使用量の上限を各画像処理モジュールのそれぞれに決定する。ここでは、CPU101はキュー管理部毎にアクティブ状態のキュー総数の上限値を設定する。本実施形態では、キュー管理部A110で3個、キュー管理部B111で9個、キュー管理部C112で8個、キュー管理部Dで16個とする。キュー管理部Eは、最後段の処理であるため、「処理済み」は即座に「削除済み」となることから、アクティブ状態のキューは、「処理待ち」と「削除待ち」は存在せず、「処理中」のみとなる。

#### 【0032】

図2において、T0経過時、T1経過時、T2経過時、T3経過時、T4経過時のキューの状態について説明する。

#### 【0033】

T0経過時には、画像処理モジュールA103は、カラー0を1回終え(キュー管理部Aで「処理済み」1つ)、カラー1の処理を実行している(キュー管理部Aで「処理中」1つ)。画像処理モジュールB104は、画像処理モジュールA103においてカラー0の処理が1回終わったため、カラー0の処理を実行している(キュー管理部Bで「処理中」1つ)。

#### 【0034】

T1経過時には、画像処理モジュールB104は、カラー0～3の処理を1回ずつ終えて(キュー管理部Bで「処理済み」4つ)、カラー0の2回目の処理を開始している(キュー管理部Bで「処理中」1つ)。画像処理モジュールB104において、画像処理モジュールA103のカラー0～3の出力結果の使用が完了しているため、画像処理モジュールA103からメモリ108への出力結果のうち各色1回分の出力結果は不要となる。したがって、キュー管理部A110のキューは、その該当箇所を削除している(キュー管理部Aで「削除済み」4つ)。なお、画像処理モジュールA103は、カラー0の2回目の処理を終了し、カラー1の2回目の処理を実行している。画像処理モジュールC105は、画像処理モジュールB104でカラー0～3の処理が1回ずつ終わったため、全色同時に1度の処理を実行している(キュー管理部Cで「処理中」1つ)。

#### 【0035】

T2経過時は、T1で開始した画像処理モジュールC105による画像処理が継続している間にも、画像処理モジュールA103と画像処理モジュールB104の処理が進んでいる状態である。画像処理モジュールA103は、カラー0の3回目の処理を実行中、画像処理モジュールB104は、カラー3の2回目の処理を実行中である。

#### 【0036】

T3経過時には、T1で開始した画像処理モジュールC105による画像処理が終了したので、画像処理モジュールD106が処理を開始した状態である。画像処理モジュールB104が2回目のカラー0～カラー3の処理を終了しており、画像処理モジュールC105の2回目の処理条件を満たしていることから、画像処理モジュールC105は、2回目の処理を実行中である。また、画像処理モジュールB104からの各色1回分の出力結果が一度に不要となったため、キュー管理部B111の各色一つずつのキューを削除している(キュー管理部Bで「削除済み」4つ)。

10

20

30

40

50

## 【 0 0 3 7 】

T 4 経過時には、T 3 で開始した画像処理モジュール D 1 0 6 は、各色 4 回ずつ処理を実施するが、カラー 0 の 4 回の処理を終え、カラー 1 の 4 回目を処理中の状態である。画像処理モジュール D 1 0 6 が画像処理モジュール E 1 0 7 のカラー 0 に対する処理条件を満たしていることから、画像処理モジュール E 1 0 7 のカラー 0 での処理を開始している。

## 【 0 0 3 8 】

図 3 を用いて、画像処理モード A とは異なるモードである画像処理モード B について説明する。図 3 は、画像処理モジュール A 1 0 3 ~ 画像処理モジュール E 1 0 7 までの各経過時間における処理状態を示す。

## 【 0 0 3 9 】

画像処理モード B では、色数としてはカラー 0 ~ 2 を入力し、カラー 1 ~ 3 までの 3 色を出力する。カラー 0 は特定の色情報ではなく、文字か否かを示す文字情報である。カラー 1 ~ 3 は色情報であり、カラー 1 は G r a y、カラー 2 とカラー 3 は、カラー 2 として入力された B l a c k を 2 値化処理によりカラー 2 とカラー 3 の 2 色に分割する。同色を分割することで（ここでは B l a c k をカラー 2 とカラー 3 に分割することで）、例えば、プリントヘッド 1 1 6 からインク滴を吐出する際に時間間隔を空けることができる。また、画像処理モード B では、画像処理モード A とは異なり、一連の画像処理の中で一定濃度以下への変換処理は実行しない。しかしながら、文字画素の周囲を間引くデータ変換を実行する。文字画素の周囲を間引く方法としては、例えば、カラー 0 の文字情報を使用して文字となる画素を特定し、カラー 1 ~ 3 でその文字画素の周囲を間引く方法が挙げられる。文字画素の周囲を間引くためには、糊代となる部分を 1 画素（6 0 0 d p i）分必要とする。したがって、画像処理モジュール A 1 0 3 が 1 6 × 1 6 画素格子（6 0 0 d p i）にデータをパッキングした後、画像処理モジュール B 1 0 4 は、パッキングされたデータの左右を含め、1 8 × 1 6 画素格子（6 0 0 d p i）を 1 回分の入出力データとする。画像処理モジュール D 1 0 6 は、画像処理モジュール B 1 0 4 の出力したデータを文字画素の周囲の間引き処理に使用する。なお、画像処理モード B では、上述した通り、一定濃度以下への変換処理は実行しないため、画像処理モジュール C 1 0 5 では処理を実行しない。

## 【 0 0 4 0 】

画像処理モジュール A 1 0 3 は、各色で多値データ（4 b i t）の 1 6 × 1 6 画素格子（6 0 0 d p i）にパッキング処理を実施する。画像処理モジュール B 1 0 4 は、画像処理モジュール A 1 0 3 の出力結果のうち 1 のパッキングに対して同色のその前後の出力結果も参照して、1 8 × 1 6 画素格子（6 0 0 d p i）多値データから 3 6 × 3 2 画素格子（1 2 0 0 d p i）の 2 値データへの変換処理を行う。1 のパッキングに対して同色のその前後の出力結果を参照するのは、糊代が必要となるためである。画像処理モジュール B 1 0 4 が各色の処理を 1 回ずつ終わると、画像処理モジュール C 1 0 5 は処理を実行せず、画像処理モジュール D 1 0 6 は、横方向 4 分割した 3 6 × 8 画素格子単位での処理を各色 4 回ずつ実施する。ただし、カラー 0 は文字情報であるため、画像処理モジュール B 1 0 4 のカラー 0 が処理完了しただけでは、画像処理モジュール D 1 0 6 は起動しない。画像処理モジュール D 1 0 6 は、画像処理モジュール B 1 0 4 がカラー 1 以降の処理を完了すると、処理を開始し、文字画素の周囲を間引いて、糊代部分を削除した 3 2 × 8 画素格子（1 2 0 0 d p i）とした後、9 0 度回転処理を実施する。なお、文字情報は、その後の処理では使用しないため、カラー 0 の出力はしない。そして画像処理モジュール D 1 0 6 が各色 4 回ずつ処理を終えると、画像処理モジュール E 1 0 7 は、3 2 × 3 2 画素格子（1 2 0 0 d p i）でヘッド吐出順へ変換処理を各色 1 回実施する。

## 【 0 0 4 1 】

画像処理モード B では、キュー管理部でのアクティブ状態のキュー総数の上限値として、キュー管理部 A 1 1 0 で 1 2 個、キュー管理部 B 1 1 1 で 8 個、キュー管理部 D で 1 2 個、キュー管理部 E で 1 個を設定する。



## 【 0 0 4 2 】

図 3 において、T 0 経過時、T 1 経過時、T 2 経過時、T 3 経過時、T 4 経過時のキューの状態について説明する。

## 【 0 0 4 3 】

T 0 経過時には、画像処理モジュール A 1 0 3 は、カラー 0 ~ 2 までの処理を 1 回ずつ終え、カラー 0 とカラー 1 は 2 回目の処理を終えて、カラー 2 の処理を実行中である。画像処理モジュール A 1 0 3 でカラー 1 まで 2 回目の処理が完了している、すなわち、カラー 0 及びカラー 1 は、糊代を含めた前後の画素の出力が終わっている。したがって、画像処理モジュール B 1 0 4 は、カラー 0 の処理を実施し、カラー 1 の処理を実行中である。なお、T 0 は、画像処理モジュール B 1 0 4 の処理は 1 回目の処理であるため、前後処理のうち前処理は存在しない。そのため、画像処理モジュール B 1 0 4 は、画像処理モジュール A 1 0 3 の 2 回分の処理完了を受けて処理を開始する。画像処理モジュール D 1 0 6 は、カラー 0 に対する起動条件である画像処理モジュール B 1 0 4 のカラー 1 までの処理完了が整っていないため、画像処理モジュール D 1 0 6 の処理はまだ開始していない。

10

## 【 0 0 4 4 】

T 1 経過時には、画像処理モジュール B 1 0 4 は、カラー 0 ~ 2 までの処理を 1 回ずつ終え、画像処理モジュール A 1 0 3 のカラー 2 の出力結果を使用してカラー 3 の処理を実行中である。画像処理モジュール D 1 0 6 は、画像処理モジュール B 1 0 4 でカラー 0 ~ 1 の処理が終わっているため、カラー 1 の処理を実行中である。

## 【 0 0 4 5 】

20

T 2 経過時には、画像処理モジュール B 1 0 4 がカラー 0 ~ 3 までの処理を 1 回ずつ終え、カラー 2 の 2 回目の処理を終え、カラー 1 の 2 回目の処理を実行中である。画像処理モジュール B 1 0 4 がカラー 0 の 2 回目の処理を完了したので、画像処理モジュール A 1 0 3 の 1 回目の処理結果は不要となり、キュー管理部 A 1 1 0 は、カラー 0 キューを 1 つ削除している。キュー管理部 B 1 1 1 では、アクティブ状態のキュー総数が 6 個となっているが、上限値 8 個に達してはならず、キューはフル状態にはなっていない。

## 【 0 0 4 6 】

T 3 経過時には、画像処理モジュール B 1 0 4 がカラー 0 ~ 3 までの処理を 2 回ずつ終えた時点で、キュー管理部 B 1 1 1 では、アクティブ状態のキュー総数が上限値 8 個に達して、キューがフル状態になる。このとき画像処理モジュール B 1 0 4 は、キュー削除待ちとなり処理が一時停止し、それに応じて画像処理モジュール A 1 0 3 での処理も同様にキュー削除待ちで一時停止している。

30

## 【 0 0 4 7 】

T 4 経過時には、画像処理モジュール D 1 0 6 の処理が進んだことによりキュー管理部 B 1 1 1 のキューのフル状態が解除され、T 3 で一時停止した画像処理モジュール B 1 0 4 の処理が再開している。画像処理モジュール A 1 0 3 は、まだ画像処理モジュール B 1 0 4 の処理が再開直後であり、出力結果を削除済みにできないため、キュー削除待ちを継続している。

## 【 0 0 4 8 】

本実施形態では、画像処理モード A や画像処理モード B のように画像処理モード毎に画像処理モジュール間の関係が異なる場合でも、キュー管理の設定を切り替えることにより、同じハードウェア構成で対応することができる。このように、複数の画像処理モジュールを有する画像処理モジュール群において、それぞれの画像処理モジュールが前後の処理単位に依存しないことにより、画像処理モジュールの構成を画像処理モジュール毎の処理規模及び処理能力に基づき決めることができる。また、画像処理モジュール間毎に起動条件を設定することができ、画像処理モジュール間のパイプライン処理をより高速に実行することができる。このように、画像処理モジュール毎に処理単位の異なる画像処理に対応することができ、画像処理モジュール群が有する画像処理モジュールの回路規模を大きくすることなく、処理を高速化することができる。

40

## 【 0 0 4 9 】

50

また、キュー管理の設定を切り替えられるため、メモリ 108 に保持する各画像処理モジュールの出力データ量の上限を、画像処理モードによって変えることができる。

#### 【0050】

図 4 に、画像処理モード A と画像処理モード B の各画像処理モジュール間での中間バッファ (SRAM) として必要となる出力データ量を示す。中間バッファ量 (出力データ量) は、各キュー管理部におけるアクティブ状態のキュー総数の上限値と、各画像処理モジュールから一度に出力する出力データサイズとの積算により求められる。例えば、画像処理モード A の画像処理モジュール A103 の場合、一度に出力する出力データが 128 byte (16 × 16 画素格子 (600 dpi) の多値 (4 bit)) であり、アクティブ状態のキュー総数の上限値 3 個である。これらを積算することにより、画像処理モード A の画像処理モジュール A103 は、中間バッファに 384 byte 分の量を確保する必要があることがわかる。同様の計算をすると、画像処理モジュール A103 ~ 画像処理モジュール E107 の総中間バッファ量は、画像処理モード A で 3072 byte 必要となり、画像処理モード B でも 3072 byte 必要となる。そして、これらの値は、画像処理モジュール毎の中間バッファ量の最大値の加算結果 4224 byte (= 1536 + 1152 + 1024 + 512) よりも小さくなっている。これにより、キュー管理の設定を切り替えることで、使用する SRAM のメモリ量のサイズを小さくすることができることがわかる。

#### 【0051】

図 5 は、このような種々の画像処理モードでの動作を実現するキュー管理部の構成を示す図である。図 5 に示すように、シーケンサ 109 は、制御対象キュー管理部 501 と、その前段キュー管理部 500 と、次段キュー管理部 502 とを含む。これは、図 1 の画像処理回路 (画像処理モジュール A103 ~ 画像処理モジュール E107) の一部を模式的に表したものである。キュー管理部 B111 を制御対象とする場合は、キュー管理部 B111 が制御対象キュー管理部 501 となり、キュー管理部 A110 が前段キュー管理部 500、キュー管理部 C112 が次段キュー管理部 502 となる。同様に、キュー管理部 C112 を制御対象とする場合、キュー管理部 C112 が制御対象キュー管理部 501 となり、キュー管理部 B111 が前段キュー管理部 500、キュー管理部 D113 が次段キュー管理部 502 に相当する。また、キュー管理部 D113 を制御対象とする場合、キュー管理部 C112 が前段キュー管理部 500 となり、キュー管理部 E114 が次段キュー管理部 502 に相当する。

#### 【0052】

図 5 に示す prv\_ro 及び prv\_cl は、制御対象キュー管理部 501 が処理を実行する上で、前段の画像処理モジュールが完了していなければならない処理条件を示す。prv\_ro は、前段の画像処理モジュールが完了していなければならない処理の回数、prv\_cl は、前段の画像処理モジュールが完了していなければならない色数を示す。この図では、prv\_ro = 1, prv\_cl = 2 とする。また、図 5 において、格子内の数字 (番号) は、各キュー管理部の処理の実行順を示す。この例では、前段キュー管理部 500 でキューが 1 番までの処理を完了すると、制御対象キュー管理部 501 の 0 番の起動条件 (カラー 0 の 1 回目) を満たす。前段キュー管理部 500 においてキューがどこまで完了したかは、前段キュー管理部 500 から制御対象キュー管理部 501 に対して通知されているキュー状態情報により把握できる。すなわち、前段のキュー管理部のキューの進行状況は、キュー状態情報として次段のキュー管理部へ通知される。なお、通知は、キューの進行状況に変化があるたびに通知してもよいし、所定間隔で常に通知するようにしてもよい。さらに、前段キュー管理部 500 が 2 番までの処理を完了すると、制御対象キュー管理部 501 の 1 番の起動条件 (カラー 1 の 1 回目) が整う。なお、前段キュー管理部 500 が対応可能な最大色数 (ここでは、カラー 3) の処理を実行しないこともありえる。例えば、図 5 に示すように、前段キュー管理部 500 がカラー 2 までの処理しか実行しない場合は、前段キュー管理部 500 が 2 番まで処理を終了すると、制御対象キュー管理部 501 は、カラー 2 ~ 3 の起動条件が整った (全色の処理完了) と判定する。これ

により、制御対象キュー管理部 5 0 1 は、3 番までの処理を実行できる。

【 0 0 5 3 】

また、図 5 に示す `uni__cl` 及び `uni__ro` は、処理単位を示すものであり、`uni__cl` は、画像処理モジュールで 1 度の処理で扱う色数を示し、`uni__ro` は、画像処理モジュールで 1 度の処理で扱うキューの個数を示す。すなわち、`uni__ro` は、画像処理モジュールが 1 度の処理で必要とする処理の回数を示す。`del__ro` は不要となる処理済みのキューの個数を示している。図 5 は、制御対象キュー管理部 5 0 1 が管理する画像処理モジュールについて、`uni__cl` = 4 , `uni__ro` = 3 , `del__ro` = 2 とする。

【 0 0 5 4 】

制御対象キュー管理部 5 0 1 において処理中の色もしくは全色 (`uni__cl` 個) が `uni__ro` 回の処理が完了したら、制御対象キュー管理部 5 0 1 から前段キュー管理部 5 0 0 へキュー削除通知を発行する。前段キュー管理部 5 0 0 が削除通知を受けると、前段キュー管理部 5 0 0 に対して設定された `del__ro` 個のキューがクリアされる。すなわち、キュー管理部 5 0 0 が管理する画像処理モジュールの出力結果をメモリ 1 0 8 から削除する。図 5 に示す例では、制御対象キュー管理部 5 0 1 が次段キュー管理部 5 0 2 からキュー削除通知を受け取った場合には、制御対象キュー管理部 5 0 1 の 0 ~ 7 番までの処理が不要となりキュー削除が実施される。すなわち、キュー管理部 5 0 0 が管理する画像処理モジュールの対象出力結果はメモリ 1 0 8 上で不要データとなる。キュー削除通知の発行タイミングは、色毎が全色のいずれで判定するかをあらかじめ決めておくことができる。全色単位で差分更新をするように設定された場合、制御対象キュー管理部 5 0 1 で 1 1 番まで終了すると、前段キュー管理部 5 0 0 のカラー 0 ~ 3 に対してキュー削除通知を発行する。各色単位で差分更新をするように設定された場合は、制御対象キュー管理部 5 0 1 で 8 番まで終了すると、カラー 0 に対してキュー削除通知を発行し、制御対象キュー管理部 5 0 1 で 9 番まで終了すると、カラー 1 に対してキュー削除通知を発行する。このように、各色毎にキュー削除通知が発行される。

【 0 0 5 5 】

また、制御対象キュー管理部 5 0 1 は、前段キュー管理部 5 0 0 の処理と制御対象キュー管理部 5 0 1 の処理との回数差分が `ful__ro` 以上あり、`uni__cl` 個の色の処理が終了している状態が継続される場合、キューがフル状態として管理する。ここで、`ful__ro` とは、制御対象画像処理モジュールと次段の画像処理モジュールとの間に存在するメモリが許容量まで達した状態を示す。図 5 では、`ful__ro` = 4 とする。この例では、制御対象キュー管理部 5 0 1 の処理が 1 5 番まで終了すると、これ以上の処理を受け付けられない状態となる。この状態が解除されるのは、次段の画像処理モジュールでの処理が進み、次段キュー管理部 5 0 2 から制御対象キュー管理部 5 0 1 にキュー削除通知が発行されたときである。

【 0 0 5 6 】

前段画像処理モジュールから全処理完了通知を受け取り、制御対象画像処理モジュールが動作受付可能状態である場合に、制御対象キュー管理部 5 0 1 は次段への全処理完了通知を発行する。

【 0 0 5 7 】

なお、最前段であるキュー管理部 A 1 1 0 を制御対象キュー管理部 5 0 1 とした場合、前段キュー管理部 5 0 0 は存在せず、常に制御対象キュー管理部 5 0 1 に対する準備が整っているように振る舞う。また、最後段であるキュー管理部 E 1 1 4 を制御対象キュー管理部 5 0 1 とした場合、次段キュー管理部 5 0 2 は存在しない。

【 0 0 5 8 】

図 6 は、図 5 で示したシーケンサ 1 0 9 がキュー管理部の情報をを用いて制御対象の画像処理モジュールに処理開始通知を制御するフローを示している。図 6 のフローチャートは、CPU 1 0 1 が、ROM 1 0 2 に格納されている制御プログラムをメモリ 1 0 8 にロードし、それを実行することにより行われる処理の流れを示す。

## 【 0 0 5 9 】

まず、シーケンサ起動すると ( S 6 0 0 )、S 6 0 1 で次段キュー管理部 5 0 2 からキュー削除通知が発行されたか判定する。キュー削除通知が発行されていれば ( S 6 0 1 で Y e s )、S 6 0 2 において制御対象キュー管理部 5 0 1 のキューを d e l \_ r o 個削除し、S 6 0 3 へ進む。キュー削除通知が発行されていない場合は ( S 6 0 1 で N o )、S 6 0 3 へ進む。

## 【 0 0 6 0 】

S 6 0 3 では、処理対象の画像処理モジュールの前段の画像処理モジュールにおいて、処理対象の画像処理モジュールの処理単位分の処理が完了したか判定する。具体的には、シーケンサ 1 0 9 が以下の 2 つの条件が成立しているか判定する。キュー管理情報において、前段キュー管理部 5 0 0 で処理済みの回数と制御対象キュー管理部 5 0 1 の処理済みの回数との差分が p r v \_ r o 以上あり、前段キュー管理部 5 0 0 において、処理対象の回数差分 p r v \_ r o の位置において処理対象の p r v \_ c l 個分の色の処理が終了していることが第 1 の条件である。そして、制御対象キュー管理部 5 0 1 はキューフル状態でないことが第 2 の条件である。これら 2 つの条件が成立した場合 ( S 6 0 3 で Y e s )、S 6 0 4 へ進む。これら 2 つの条件が成り立たない場合 ( S 6 0 3 で N o )、S 6 0 1 からやり直す。

## 【 0 0 6 1 】

S 6 0 4 では、制御対象の画像処理モジュールに処理開始通知を発行、すなわち、制御対象の画像処理モジュールに画像処理の開始を指示し、S 6 0 5 で制御対象の画像処理モジュールが処理完了状態になるまで待つ。

## 【 0 0 6 2 】

制御対象の画像処理モジュールが処理完了状態になると ( S 6 0 5 で Y e s )、S 6 0 6 で前段キュー管理部 5 0 0 へのキュー削除通知の条件が整っているか判定する。キュー削除通知条件が整っている場合 ( S 6 0 6 で Y e s )、S 6 0 7 で前段キュー管理部 5 0 0 にキュー削除通知を発行、すなわち、前段キュー管理部 5 0 0 にキュー削除を指示し、S 6 0 8 へ進む。キュー削除通知の条件が整っていない場合 ( S 6 0 6 で N o )、S 6 0 8 へ進む。

## 【 0 0 6 3 】

S 6 0 8 で S 6 0 4 で処理した処理データが最終データか判定する。前段キュー管理部 5 0 0 から全処理完了通知が通知済みであり、前段キュー管理部 5 0 0 から得たキュー管理情報でアクティブなキューが存在しない場合、最終データであると判定する。

## 【 0 0 6 4 】

最終データである場合 ( S 6 0 8 で Y e s )、S 6 0 9 で次段キュー管理部 5 0 2 に全処理完了通知を発行し、制御対象の画像処理モジュールによる処理を完了する ( S 6 1 0 )。最終データでない場合 ( S 6 0 8 で N o )、再度 S 6 0 1 から S 6 0 8 までの処理を繰り返す。このように、シーケンサ 1 0 9 において、各画像処理モジュールに対する処理制御パラメータを、画像処理モジュール間でそれぞれ設定する構成とすることにより、画像処理回路が実行する一連の画像処理は、種々の処理単位に対応することができる。すなわち、例えば、単色毎や複数色間での画像処理、異なる画素格子単位での画像処理といった種々の処理単位が混在するハードウェア構成を実現することができる。

## 【 0 0 6 5 】

( 他の実施形態 )

本発明は上述した実施形態に限定されるものではない。例えば、上述した実施形態では、画像処理モジュール群は、複数の処理モードに対応可能としたが、これに限定されず、処理モードは 1 つ、すなわち、固定であってもよい。この場合は、画像処理モジュールに予め処理単位とする画素数を設定し、キュー管理部に管理するキューの総数の上限値を設定するようすればよい。

## 【 0 0 6 6 】

また、図 2 や図 3 では、4 色の処理を例に挙げて説明したが、これに限定されず、8 色

10

20

30

40

50

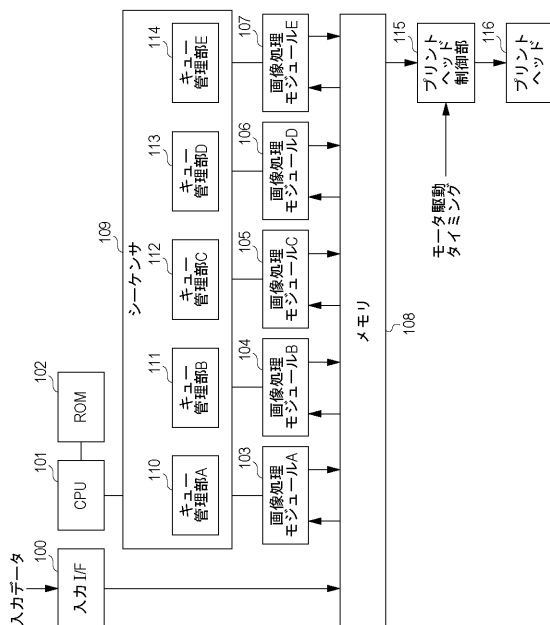
等他の処理を行う場合も上述した実施形態と同様に処理を行えばよい。また、画像処理モジュール群が実行する処理は、単色であってもよい。

【 0 0 6 7 】

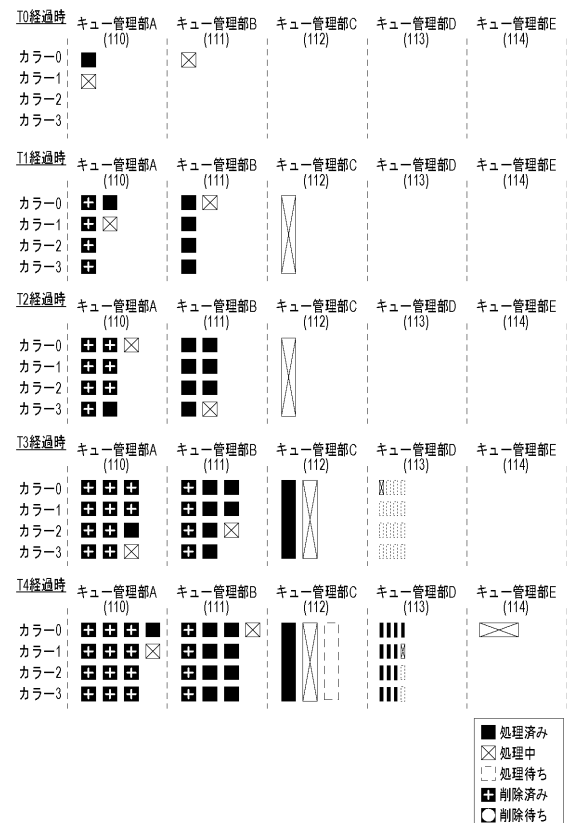
上述した実施形態は、以下の処理を実行することによっても実現される。すなわち、上述した実施形態の機能を実現するソフトウェア（プログラム）を、ネットワーク又は各種記憶媒体を介してシステム或いは装置に供給し、そのシステム或いは装置のコンピュータ（CPUやMPU等）がプログラムを読み出して実行する処理である。また、プログラムは、1つのコンピュータで実行させても、複数のコンピュータを連動させて実行させるようにしてもよい。また、上記した処理の全てをソフトウェアで実現する必要はなく、処理の一部または全部をASIC等のハードウェアで実現するようにしてもよい。また、CPUも1つのCPUで全ての処理を行うものに限らず、複数のCPUが適宜連携をしながら処理を行うものとしてもよい。

10

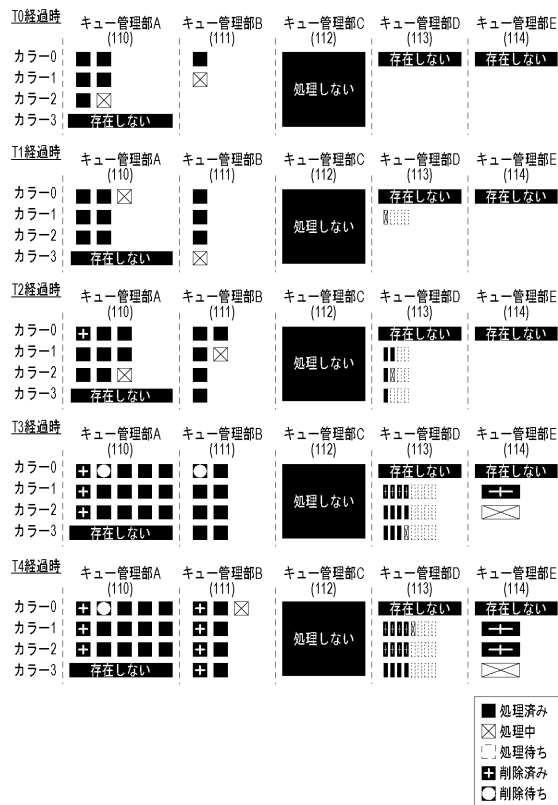
【 図 1 】



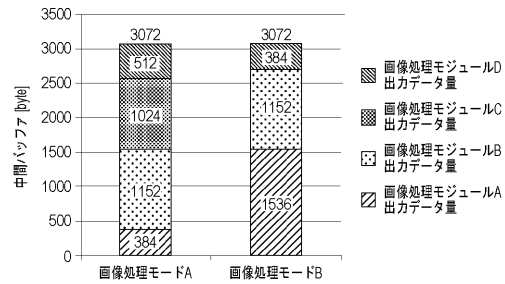
【 図 2 】



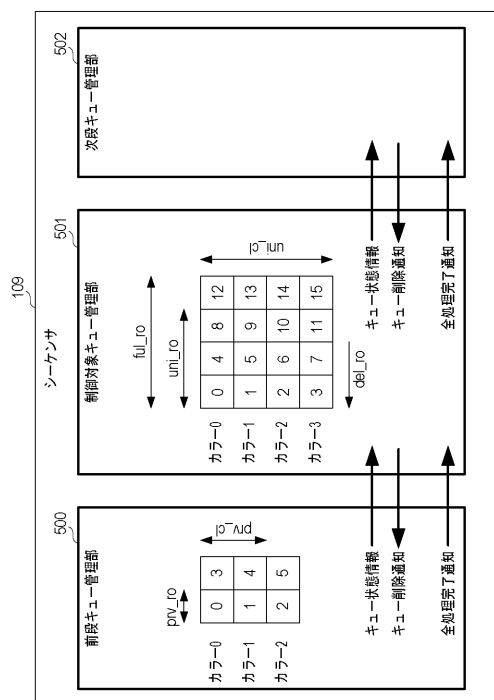
【 図 3 】



【 図 4 】

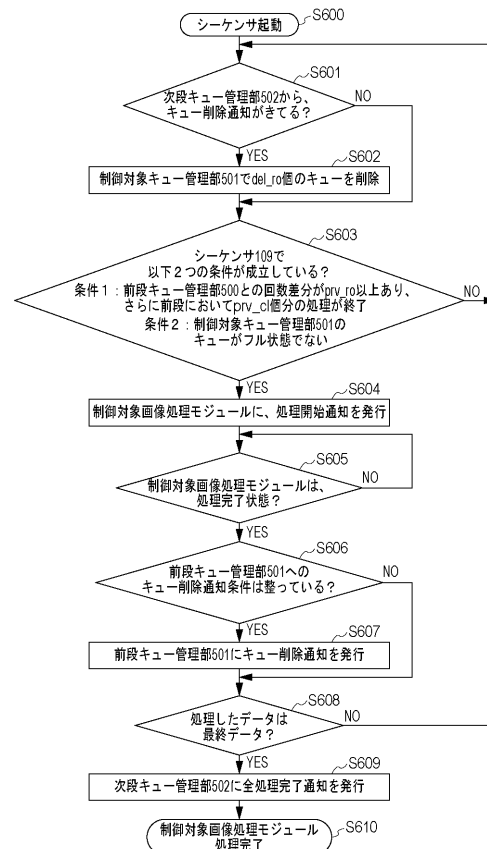


【 図 5 】



※格子内の番号は、処理順を示す

【 図 6 】



---

フロントページの続き

(56)参考文献 特開 2011-028742 (JP, A)  
特開 2005-322049 (JP, A)  
米国特許出願公開第 2005/0111728 (US, A1)  
特開 2004-222069 (JP, A)  
特開 2008-021249 (JP, A)

(58)調査した分野(Int.Cl., DB名)

B 4 1 J	5 / 3 0
G 0 6 T	1 / 2 0
H 0 4 N	1 / 0 0
H 0 4 N	1 / 2 1