

(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2010-86497

(P2010-86497A)

(43) 公開日 平成22年4月15日(2010.4.15)

(51) Int.Cl.	F I	テーマコード (参考)
<b>G06T 1/20 (2006.01)</b>	G06T 1/20 B	5B057
<b>G06F 12/02 (2006.01)</b>	G06F 12/02 550C	5B060

審査請求 未請求 請求項の数 14 O L (全 22 頁)

(21) 出願番号	特願2008-258039 (P2008-258039)	(71) 出願人	503121103
(22) 出願日	平成20年10月3日 (2008.10.3)		株式会社ルネサステクノロジ
			東京都千代田区大手町二丁目6番2号
		(74) 代理人	100089071
			弁理士 玉村 静世
		(72) 発明者	高橋 宜孝
			茨城県日立市大みか町七丁目1番1号 株
			式会社日立製作所日立研究所内
		(72) 発明者	村松 彰二
			茨城県日立市大みか町七丁目1番1号 株
			式会社日立製作所日立研究所内
		(72) 発明者	中三川 哲明
			茨城県日立市大みか町七丁目1番1号 株
			式会社日立製作所日立研究所内

最終頁に続く

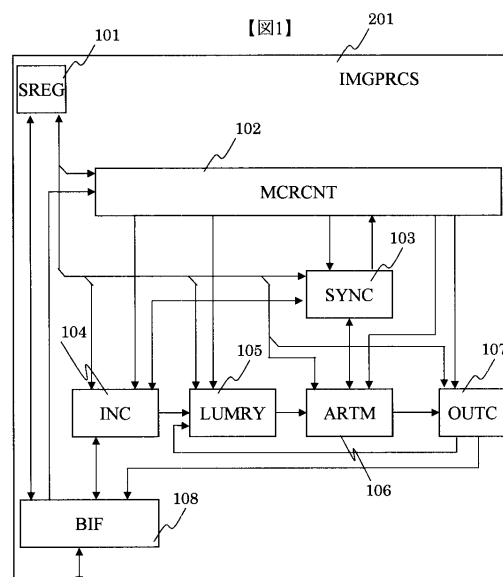
(54) 【発明の名称】 画像処理装置及びデータプロセッサ

## (57) 【要約】

【課題】ハードワイヤード方式の効率性を確保し、且つ、様々な画像処理機能を実現することが容易な画像処理装置を提供する。

【解決手段】ハードワイヤード方式で実現される画像処理のための演算機能とバッファメモリのメモリアクセス制御に制約を設けて、その制約の範囲をプログラム制御等によって可変可能とする。バッファメモリ(105)は記憶ライン単位という制約を受けて外部からデータを入力し、入力する記憶ライン数と記憶ラインの位置を制御回路(101, 102)によりプログラマブルにでき、演算回路(106)はバッファメモリから供給される単数又は複数の記憶ラインのデータ単位で演算を行うという制約を受け、そのデータ単位に対する演算処理単位の演算処理内容を制御回路(101, 102)によりプログラマブルに指定することができる。

【選択図】図1



**【特許請求の範囲】****【請求項 1】**

演算対象とするデータを外部から読み出して入力するための入力回路と、入力回路によって入力したデータを一時的に保持するバッファメモリと、前記バッファメモリから出力されたデータの演算処理を行なう演算回路と、前記演算回路による演算結果を外部又は前記バッファメモリに書き戻すための出力回路と、制御回路と、を有し、

前記バッファメモリは記憶領域として論理上直列な記憶ラインを複数有し、前記制御回路で指定された前記記憶ラインに入力データを書き込み書き込まれたデータの読出しが可能とされ、

前記演算回路はバッファから出力された単数又は複数の前記記憶ラインのデータを前記制御回路により指定された処理内容で演算処理単位毎に繰り返し演算し、

前記制御回路は、指定した記憶ラインのデータを記憶ライン単位で前記バッファメモリから前記演算回路に出力させる、画像処理装置。

**【請求項 2】**

前記制御回路は、外部から入力されるデータを書き込む単数又は複数の前記記憶ラインを指示し、演算回路による演算結果を書き戻す前記記憶ラインを指示する、請求項 1 記載の画像処理装置。

**【請求項 3】**

画像処理装置と、前記画像処理装置の制御及びメモリのアクセス制御を行う中央処理装置とを有するデータプロセッサであって、

前記画像処理装置は、演算対象とするデータを前記メモリから読み出して入力するための入力回路と、入力回路によって入力したデータを一時的に保持するバッファメモリと、前記バッファメモリから出力されたデータの演算処理を行なう演算回路と、前記演算回路による演算結果を前記メモリ又は前記バッファメモリに書き戻すための出力回路と、制御回路と、を有し、

前記バッファメモリは記憶領域として論理上直列な記憶ラインを複数有し、指定された前記記憶ラインに入力データを書き込み書き込まれたデータの読出しが可能とされ、

前記演算回路はバッファから出力された単数又は複数の前記記憶ラインのデータを指定された処理内容で演算処理単位毎に繰り返し演算し、

前記制御回路は、前記入力回路から入力されるデータを書き込む単数又は複数の前記記憶ラインを指示し、前記演算回路による演算処理内容を指示し、演算回路による演算結果を書き戻す前記記憶ラインを指示し、バッファメモリから演算回路にデータを供給する記憶ラインを指示する、データプロセッサ。

**【請求項 4】**

前記中央処理装置は画像処理装置の演算動作中に、画像処理装置による演算結果を前記メモリから参照する、請求項 3 記載のデータプロセッサ。

**【請求項 5】**

画像処理装置とメモリとを有するデータプロセッサであって、

前記画像処理装置は、演算対象とするデータを前記メモリから読み出して入力するための入力回路と、入力回路によって入力したデータを一時的に保持するバッファメモリと、前記バッファメモリから出力されたデータの演算処理を行なう演算回路と、前記演算回路による演算結果を前記メモリ又は前記バッファメモリに書き戻すための出力回路と、制御回路と、を有し、

前記バッファメモリは記憶領域として論理上直列な記憶ラインを複数有し、前記制御回路で指定された前記記憶ラインに入力データを書き込み書き込まれたデータの読出しが可能とされ、

前記演算回路は前記バッファメモリから読み出された複数の前記記憶ラインのデータを前記制御回路で指定された処理内容に従って並列に演算可能であり、

前記制御回路は、前記バッファメモリの複数の記憶ライン分に相当する第 1 の記憶領域のデータに対して順次データ処理単位毎に前記演算回路に第 1 の演算を繰り返し実行させ

10

20

30

40

50

、繰り返し実行された第 1 の演算による演算家結果が前記バッファメモリの複数の記憶ライン分に相当する第 2 の記憶領域の記憶ラインに格納されたとき、第 1 の記憶領域で最優先にデータ記憶が行われた記憶ラインに対してデータ入換えを行ってから、再び前記第 1 の演算を繰り返し実行させる制御を行う、データプロセッサ。

【請求項 6】

前記制御部は、前記第 2 の記憶領域の記憶ラインに必要な演算結果が揃ったとき、第 2 の記憶領域のデータに対して順次データ処理単位毎に前記演算回路に第 2 の演算を繰り返し実行させ、繰り返し実行された第 2 の演算による演算家結果を前記バッファメモリの第 3 の記憶領域の記憶ラインに格納させる制御を行う、請求項 5 記載のデータプロセッサ。

【請求項 7】

前記制御部は、前記第 3 の記憶領域の記憶ラインに必要な演算結果が揃ったとき、第 3 の記憶領域のデータに対して前記演算回路に第 3 の演算を繰り返し実行させ、繰り返し実行された第 3 の演算による演算家結果を前記バッファメモリの第 4 の記憶領域の記憶ラインに格納させる制御を行う、請求項 6 記載のデータプロセッサ。

【請求項 8】

前記制御回路は、前記第 4 の記憶領域の記憶ラインに必要な演算結果が揃ったとき、前記出力回路に指示を与えて当該演算結果を前記メモリに書き込む制御を行う、請求項 7 記載のデータプロセッサ。

【請求項 9】

前記制御部は、前記第 3 の記憶領域の記憶ラインに必要な演算結果が揃ったとき、第 3 の記憶領域のデータに対して前記演算回路に第 3 の演算を繰り返し実行させ、繰り返し実行された第 3 の演算による演算家結果を前記出力回路に外部へ出力させる制御を行う、請求項 6 記載のデータプロセッサ。

【請求項 10】

前記制御回路は、マイクロコントローラ、制御レジスタ、及び同期化制御回路を有し、前記マイクロコントローラはプログラムを実行して前記制御レジスタに制御データを書き込む制御を行い、

前記同期化制御回路は前記入力回路及び前記演算回路の動作状態に従って前記前記制御レジスタの書き込み制御を行い、

前記制御レジスタは書き込まれた制御データに従って前記入力回路、前記バッファ回路、前記演算回路及び前記出力回路に制御信号を出力する、請求項 5 乃至 9 の何れか 1 項記載のデータプロセッサ。

【請求項 11】

前記制御レジスタは、入力回路からデータを取り込む記憶ラインを指定するための制御情報、出力回路からデータを取り込む記憶ラインを指定するための制御情報、データを取り込む記憶ラインの本数を指定する制御情報、データを出力する記憶ラインを指定するための制御情報、及びデータを出力する記憶ラインの本数を指定する制御情報が設定される、請求項 10 記載のデータプロセッサ。

【請求項 12】

前記第 1 の演算は、複数記憶ラインの画像データに対して  $m \times n$  画素単位のデータをデータ処理単位とする平滑化のためのコンボリューション演算である、請求項 7 記載のデータプロセッサ。

【請求項 13】

前記第 2 の演算は、前記コンボリューション演算された複数記憶ラインの画像データに対して  $i \times j$  画素単位もデータをデータ処理単位とする輪郭強調のためのフィルタ演算である、請求項 12 記載のデータプロセッサ。

【請求項 14】

前記第 3 の演算は、前記前記フィルタ演算された画像データを 2 値化する演算である、請求項 13 記載のデータプロセッサ。

【発明の詳細な説明】

10

20

30

40

50

## 【技術分野】

## 【0001】

本発明は、画像処理のためのデータ処理技術に関し、例えばラインメモリのようなバッファメモリと演算回路を用いて画像処理を行なう画像処理装置、更にはデータプロセッサに適用して有効な技術に関する。

## 【背景技術】

## 【0002】

各種の画像処理アルゴリズムについて研究が進展しており、新しいアルゴリズムの発表や従来アルゴリズムの改良等が発表され続けている。これら新アルゴリズムに対応するためには、画像処理演算のプログラムを改変することにより対応することが考えられる。これはDSP（デジタル・シグナル・プロセッサ）や一部の多並列演算器を備えた画像処理アーキテクチャによって実現されており、プログラムを組むことにより新アルゴリズムにも一定の対応が可能なものとなっている。プログラムによって新アルゴリズムに対応できるようなアーキテクチャは、一般に消費電力や面積性能比（コストパフォーマンス）の上でハードワイヤード方式に劣る。組込みコントローラ向けには、消費電力や要求される高い面積性能比（コストパフォーマンス）から、ハードワイヤード方式の方が有利である。しかしながら、ハードワイヤード方式はハード設計後に登場する新アーキテクチャへの対応が困難であるという課題があった。ハードワイヤードをベースとして、演算データをラインメモリに書き戻す等、演算の自由度を増やすことを狙った発明として、特許文献1

10

20

## 【0003】

【特許文献1】特開平10-340340号公報

## 【発明の開示】

## 【発明が解決しようとする課題】

## 【0004】

しかしながら、演算の自由度を増やすことを狙った特許文献1に開示の技術では、ハードワイヤード方式とプログラム方式との双方の利点を生かすことが難しいことが本発明者によって見出された。即ち、画像処理機能をハードワイヤード方式で実現される専用ハードウェアで構成した場合、必要最小限のハードウェアで、高い性能を実現することができる反面、メモリのアクセス制御や画像処理の演算処理をハードワイヤード方式にした場合には制御や機能が回路として実装されているため、設計時に予定されている演算アルゴリズム以外の処理をすることは事実上できない。また、汎用プロセッサのようにメモリのアクセス制御や演算処理をプログラムで記述する方式の場合、様々な演算アルゴリズムを可能とするため、メモリへのアクセスに自由度を持たせ、回路が複雑かつ回路規模も大きくなり、その結果、自由度の大きなプログラム記述方式では、ハードワイヤード方式に比べると、同じ性能を実現するためには大きな回路規模で実現しなければならない。

30

## 【0005】

本発明の別の目的は、回路規模が小さく、処理性能に優れた画像処理装置を提供することにある。

## 【0006】

本発明の別の目的は、ハードワイヤード方式の効率性を確保し、且つ、様々な画像処理機能を実現することが容易な画像処理装置、更にはデータプロセッサを提供することにある。

40

## 【0007】

本発明の前記並びにその他の目的と新規な特徴は本明細書の記述及び添付図面から明らかになるであろう。

## 【課題を解決するための手段】

## 【0008】

本願において開示される発明のうち代表的なものの概要を簡単に説明すれば下記の通りである。

50

## 【 0 0 0 9 】

すなわち、ハードワイヤード方式で実現される画像処理のための演算機能とバッファメモリのメモリアクセス制御に制約を設けて、その制約の範囲をプログラム制御等によって可変可能とする。これにより、ハードワイヤード方式の効率性が確保され、様々な画像処理機能を実現できるようにそれらのハードワイヤード回路を制御できるようになる。

## 【 発明の効果 】

## 【 0 0 1 0 】

本願において開示される発明のうち代表的なものによって得られる効果を簡単に説明すれば下記のとおりである。

## 【 0 0 1 1 】

すなわち、ハードワイヤード方式で高効率に実装された演算アルゴリズムを可変可能に用いてより複雑な演算アルゴリズムを実現することができ、回路規模が小さく、処理性能に優れた画像処理装置を提供することができる。

## 【 発明を実施するための最良の形態 】

## 【 0 0 1 2 】

## 1. 実施の形態の概要

先ず、本願において開示される発明の代表的な実施の形態について概要を説明する。代表的な実施の形態についての概要説明で括弧を付して参照する図面中の参照符号はそれが付された構成要素の概念に含まれるものを例示するに過ぎない。

## 【 0 0 1 3 】

〔 1 〕本発明に係る画像処理装置（ 2 0 1 ）は、演算対象とするデータを外部から読み出して入力するための入力回路（ 1 0 4 ）と、入力回路によって入力したデータを一時的に保持するバッファメモリ（ 1 0 5 ）と、前記バッファメモリから出力されたデータの演算処理を行なう演算回路（ 1 0 6 ）と、前記演算回路による演算結果を外部又は前記バッファメモリに書き戻すための出力回路（ 1 0 7 ）と、制御回路（ 1 0 1 , 1 0 2 ）と、を有する。前記バッファメモリは記憶領域として論理上直列な記憶ライン（ M L i ）を複数有し、前記制御回路で指定された前記記憶ラインに入力データを書き込み書き込まれたデータの読出しが可能とされる。前記演算回路はバッファから出力された単数又は複数の前記記憶ラインのデータを前記制御回路により指定された処理内容で演算処理単位毎に繰り返し演算する。前記制御回路は、指定した記憶ラインのデータを記憶ライン単位で前記バッファメモリから前記演算回路に出力させる。

## 【 0 0 1 4 】

上記によれば、バッファメモリは記憶ライン単位という制約を受けて外部からデータを入力し、入力する記憶ライン数と記憶ラインの位置を制御回路によりプログラマブルにでき、演算回路はバッファメモリから供給される単数又は複数の記憶ラインのデータ単位で演算を行うという制約を受け、そのデータ単位に対する演算処理単位の演算処理内容を制御回路によりプログラマブルに指定することができる。したがって、ハードワイヤード方式の効率性を確保し、様々な画像処理機能を実現できるようにそれらのハードワイヤード回路を制御回路で制御することができる。

## 【 0 0 1 5 】

〔 2 〕項 1 の画像処理装置において、前記制御回路は、外部から入力されるデータを書き込む単数又は複数の前記記憶ラインを指示し、演算回路による演算結果を書き戻す前記記憶ラインを指示する。

## 【 0 0 1 6 】

〔 3 〕本発明に係るデータプロセッサは、画像処理装置（ 2 0 1 ）と、前記画像処理装置の制御及びメモリのアクセス制御を行う中央処理装置（ 2 0 8 ）とを有する。前記画像処理装置は、演算対象とするデータを前記メモリから読み出して入力するための入力回路と、入力回路によって入力したデータを一時的に保持するバッファメモリと、前記バッファメモリから出力されたデータの演算処理を行なう演算回路と、前記演算回路による演算結果を前記メモリ又は前記バッファメモリに書き戻すための出力回路と、制御回路と、を

10

20

30

40

50

有する。前記バッファメモリは記憶領域として論理上直列な記憶ラインを複数有し、指定された前記記憶ラインに入力データを書き込み書き込まれたデータの読出しが可能とされる。前記演算回路はバッファから出力された単数又は複数の前記記憶ラインのデータを指定された処理内容で演算処理単位毎に繰り返し演算する。前記制御回路は、前記入力回路から入力されるデータを書き込む単数又は複数の前記記憶ラインを指示し、前記演算回路による演算処理内容を指示し、演算回路による演算結果を書き戻す前記記憶ラインを指示し、バッファメモリから演算回路にデータを供給する記憶ラインを指示する。

【0017】

上記同様に、ハードワイヤード方式の効率性を確保し、様々な画像処理機能を実現できるようにそれらのハードワイヤード回路を制御回路で制御することができる。

【0018】

〔4〕項3のデータプロセッサにおいて、前記中央処理装置は画像処理装置の演算動作中に、画像処理装置による演算結果を前記メモリから参照する。

【0019】

〔5〕本発明の別の観点によるデータプロセッサは、画像処理装置と、前記画像処理装置の制御及びメモリのアクセス制御を行う中央処理装置とを有する。前記画像処理装置は、演算対象とするデータを前記メモリから読み出して入力するための入力回路と、入力回路によって入力したデータを一時的に保持するバッファメモリと、前記バッファメモリから出力されたデータの演算処理を行なう演算回路と、前記演算回路による演算結果を前記メモリ又は前記バッファメモリに書き戻すための出力回路と、制御回路と、を有する。前記バッファメモリは記憶領域として論理上直列な記憶ライン（ $ML_i$ ）を複数有し、前記制御回路で指定された前記記憶ラインに入力データを書き込み書き込まれたデータの読出しが可能とされる。前記演算回路は前記バッファメモリから読み出された複数の前記記憶ラインのデータを前記制御回路で指定された処理内容に従って並列に演算可能である。前記制御回路は、前記バッファメモリの複数の記憶ライン分に相当する第1の記憶領域（ $ML_i \sim ML_{i+4}$ ）のデータに対して順次データ処理単位毎に前記演算回路に第1の演算を繰り返し実行させ、繰り返し実行された第1の演算による演算結果が前記バッファメモリの複数の記憶ライン分に相当する第2の記憶領域（ $ML_j \sim ML_{j+2}$ ）の記憶ラインに格納されたとき、第1の記憶領域で最も先にデータ記憶が行われた記憶ラインに対してデータ入換えを行ってから、再び前記第1の演算を繰り返し実行させる制御を行う。

【0020】

上記同様に、ハードワイヤード方式の効率性を確保し、様々な画像処理機能を実現できるようにそれらのハードワイヤード回路を制御回路で制御することができる。更に、バッファメモリに格納したデータを用いた第1の演算の結果をバッファメモリに書き戻すと共に、第1の演算の用いるデータを記憶ライン単位で更新しながら第1の演算を継続する演算アルゴリズムを実現することができる。

【0021】

〔6〕項5のデータプロセッサにおいて、前記制御部は、前記第2の記憶領域の記憶ラインに必要な演算結果が揃ったとき、第2の記憶領域のデータに対して順次データ処理単位毎に前記演算回路に第2の演算を繰り返し実行させ、繰り返し実行された第2の演算による演算結果を前記バッファメモリの第3の記憶領域（ $ML_k$ ）の記憶ラインに格納させる制御を行う。

【0022】

これにより、バッファメモリに書き戻した第1の演算結果を更に用いて第2の演算を行い、その結果を更にバッファメモリに書き戻して次の演算に備えるという演算アルゴリズムを実現することができる。

【0023】

〔7〕項6のデータプロセッサにおいて、前記制御部は、前記第3の記憶領域の記憶ラインに必要な演算結果が揃ったとき、第3の記憶領域のデータに対して前記演算回路に第3の演算を繰り返し実行させ、繰り返し実行された第3の演算による演算結果を前記バ

ッファメモリの第 4 の記憶領域 (MLm) の記憶ラインに格納させる制御を行う。

【 0 0 2 4 】

これにより、バッファメモリに書き戻した第 2 の演算結果を更に用いて第 3 の演算を行い、その結果を更にバッファメモリに書き戻して次の処理に備えるという演算アルゴリズムを実現することができる。

【 0 0 2 5 】

〔 8 〕項 7 のデータプロセッサにおいて、前記制御回路は、前記第 4 の記憶領域の記憶ラインに必要な演算結果が揃ったとき、前記出力回路に指示を与えて当該演算結果を前記メモリに書き込む制御を行う。これにより、画像処理装置とメモリとの間のデータ転送に伴うオーバーヘッドを抑えることができる。

【 0 0 2 6 】

〔 9 〕項 6 のデータプロセッサにおいて、前記制御部は、前記第 3 の記憶領域の記憶ラインに必要な演算結果が揃ったとき、第 3 の記憶領域のデータに対して前記演算回路に第 3 の演算を繰り返し実行させ、繰り返し実行された第 3 の演算による演算家結果を前記出力回路に外部へ出力させる制御を行う。

【 0 0 2 7 】

〔 1 0 〕項 5 乃至 9 の何れかのデータプロセッサにおいて、前記制御回路は、マイクロコントローラ、制御レジスタ、及び同期化制御回路を有する。前記マイクロコントローラはプログラムを実行して前記制御レジスタに制御データを書き込む制御を行う。前記同期化制御回路は前記入力回路及び前記演算回路の動作状態に従って前記前記制御レジスタの書き込み制御を行う。前記制御レジスタは書き込まれた制御データに従って前記入力回路、前記バッファ回路、前記演算回路及び前記出力回路に制御信号を出力する。

【 0 0 2 8 】

〔 1 1 〕項 1 0 のデータプロセッサにおいて、前記制御レジスタは、入力回路からデータを取り込む記憶ラインを指定するための制御情報、出力回路からデータを取り込む記憶ラインを指定するための制御情報、データを取り込む記憶ラインの本数を指定する制御情報、データを出力する記憶ラインを指定するための制御情報、及びデータを出力する記憶ラインの本数を指定する制御情報が設定される。

【 0 0 2 9 】

〔 1 2 〕項 7 のデータプロセッサにおいて、前記第 1 の演算は、複数記憶ラインの画像データに対して  $m \times n$  画素単位 of データをデータ処理単位とする平滑化のためのコンボリューション演算である。

【 0 0 3 0 】

〔 1 3 〕項 1 2 のデータプロセッサにおいて、前記第 2 の演算は、前記コンボリューション演算された複数記憶ラインの画像データに対して  $i \times j$  画素単位もデータをデータ処理単位とする輪郭強調のためのフィルタ演算である。

【 0 0 3 1 】

〔 1 4 〕項 1 3 のデータプロセッサにおいて、前記第 3 の演算は、前記前記フィルタ演算された画像データを 2 値化する演算である。

【 0 0 3 2 】

## 2. 実施の形態の詳細

実施の形態について更に詳述する。以下、本発明を実施するための形態を図面に基づいて詳細に説明する。なお、発明を実施するための形態を説明するための全図において、同一の機能を有する要素には同一の符号を付して、その繰り返しの説明を省略する。

【 0 0 3 3 】

図 2 には本発明に係るデータプロセッサが例示される。同図に示されるデータプロセッサ (MCU) 1 は、特に制限されないが、相補型 MOS 集積回路製造技術等により単結晶シリコン等の 1 個の半導体基板に形成され、例えばシステム・オン・チップの LSI として構成される。。

【 0 0 3 4 】

10

20

30

40

50

図 2 において、201 は画像処理装置 (IMGPRCS)、202 はチップ内バス、203 は周辺インタフェース (I/O)、204 はリードオンリーメモリ (ROM)、205 は表示出力回路 (DCNT)、206 は主記憶インタフェース (MCNT)、207 は映像入力回路 (VIN)、208 は中央演算装置 (CPU)、209 は主記憶 (RAM) である。主記憶 209 はデータプロセッサ 1 にオンチップされても良い。

【0035】

画像処理装置 201 は、画像処理を高速に処理するハードウェアであり、その詳細は後述する。

【0036】

チップ内バス 202 は、便宜的に 1 階層で例示した内部バスであり、夫々の内部回路モジュール間でデータやアドレス等を伝達するために使用され、例えば、スプリットトランザクションバス等によって構成することが可能である。

【0037】

周辺インタフェース 203 は、データプロセッサ 1 を用いる組み込みシステムにおける信号のインプットとアウトプットをつかさどる回路である。

【0038】

リードオンリーメモリ 204 は、読み出し専用のメモリで、本システムのブートプログラムやシステムに必要な設定等が格納されている。

【0039】

表示出力回路 205 は、液晶ディスプレイ等の表示装置に接続するための回路である。

【0040】

主記憶インタフェース 206 は、シンクロナス DRAM 等から成る主記憶を制御するメモリコントローラである。

【0041】

映像入力回路 207 は、画像入力カメラ等からの画像データを受け取り、その信号をチップ内バス 202 経由で画像処理装置 201 に伝える。

【0042】

CPU 208 はデータプロセッサ 1 の全体に対する制御を司る回路であり、画像処理装置 201 に対する各種設定も CPU 208 が行う。特に図示はしないが、ROM 204 を命令をフェッチし、フェッチした命令を解釈し、その解釈結果に従って命令を実行するための演算処理やアクセス処理等を行なう。

【0043】

画像処理に使用する画像データは、例えば主記憶 209 に格納されている。この画像データは、カメラから入力される場合、映像入力回路 207 を介して主記憶 209 に格納される。また、カメラ以外から入力される場合、周辺インタフェース 203 などの外部インタフェースから入力されて主記憶 209 に格納される。格納された画像データは、例えば図 13 に示したメモリ空間にあるように、一塊のデータとして格納されている。この一塊とは、物理的なアドレスを意味しているわけではなく、論理的に一塊のデータとして扱う状況を意味し、連続した画像データが一続きのアドレスに格納されている。カメラから入力された画像データは、映像入力回路 207 によって主記憶 209 の指定されたアドレスに書き込まれる。書き込むアドレスは、CPU 208 がプログラムによって決定し、予め映像入力回路 207 に設定されている。

【0044】

画像処理を行う場合、CPU 208 が処理する場合と、画像処理装置 201 が処理する場合の、二通りの処理形態が可能である。CPU 208 で処理する場合は、その命令セットに含まれる演算命令や転送命令等を用いて行うものであり、通常の汎用 CPU の動作と差が無いので、ここではその詳細な説明を省略する。画像処理装置 201 で処理を行なう場合は、主記憶 209 の画像データを主記憶インタフェース 206 経由で画像処理装置 201 に転送する。画像処理の結果は、CPU 208 や表示出力回路 205 がアクセスできるように、主記憶 209 に書き戻される。これらの設定も CPU 208 の動作プログラム

10

20

30

40

50



に従って制御される。すなわち、主記憶 209 のどのアドレスの画像データに関して、画像処理装置 201 がどのような処理をし、主記憶 209 のどのアドレスに書き戻すかについては、CPU 208 がその動作プログラムにしたがって処理を行う。

#### 【0045】

図 1 には画像処理装置 201 の一例が示される。画像処理装置 201 は、動作を規定するシステムレジスタ (SREG) 101、画像処理装置 201 の動作をプログラムによって制御するマイクロコントローラ (MCRCNT) 102、画像処理に必要なデータ取得のタイミングや画像処理機能の切り替えのタイミングを制御する同期化回路 (SYNC) 103、主記憶 209 から画像データを読み出す入力回路 (INC) 104 および処理結果を主記憶 209 に書き戻す出力回路 (OUTC) 107、処理に必要な画像データを一時記録しておくバッファメモリとしてのラインメモリ (LNMRY) 105、主にハードワイヤードロジック方式で構成され高速に画像処理が可能である演算回路 (ARTM) 106、及びバスインタフェース (BIF) 108 から構成されている。バスインタフェース 108 は、画像処理装置 201 をバス 202 に接続するためのインタフェースである。

#### 【0046】

画像処理装置 201 は、システムレジスタ 101 に設定された情報に基づいて処理を行う。システムレジスタ 101 には、その他の内部回路 102 ~ 108 の動作に必要な情報が設定され、その設定値は、夫々の内部回路 102 ~ 108 に出力され、夫々の動作を制御する。また、内部レジスタ 101 は、画像処理の演算結果や動作の状況をモニタするために内部回路 102 ~ 108 から所定の情報を受け取り、保持する機能を有する。システムレジスタ 101 は、図 2 における CPU 208 からバスインタフェース 108 を介してアクセスされることにより制御データの設定が可能にされ、CPU 208 はシステムレジスタ 101 を介して画像処理装置 201 を制御する。

#### 【0047】

画像処理装置 201 における画像処理の一例を説明する。ここでは、図 13 にある画像データ A を処理して、処理結果を画像データ C として書き込むことを想定する。

#### 【0048】

CPU 208 からシステムレジスタ 101 を介して、入力回路 104 にデータの取得を設定する。この設定は、画像データ A のメモリ空間上の位置を定義している。画像処理の内容は、システムレジスタ 101 に設定され、その内容が演算回路 106 に伝達される。入力回路 104 と演算回路 106 の間にあるラインメモリ 105 は、入力回路 104 から入力されるデータを保持し、演算回路 106 に必要なデータを供給する役目を果たす。例えば、3 × 3 ウィンドウの平滑化フィルタの場合、処理対象の画素データを処理するのに前後左右一画素ずつの合計 9 個のデータを必要とする。これらの画像データを供給するためには、3 ライン分の画像データが必要となり、それらを記録するラインメモリ 105 が必要となる。ラインメモリ 105 から供給された画像データを演算回路 106 で処理する。演算回路 106 は、システムレジスタ 101 に設定された機能を実行し、ハードワイヤード方式で構成された演算回路構成を有する。ハードワイヤード方式ではあるが、システムレジスタ 101 の設定内容を変更することにより、多数の画像処理の演算アルゴリズムを実行することができる。演算回路 106 で処理された結果は出力回路 107 に出力され、バスインタフェース回路 108 経由で主記憶 209 に供給されるか、ラインメモリ 105 に書き戻さえるかの二通りから一つが選択される。この選択設定は、マイクロコントローラ 102 で制御が可能である。主記憶 209 に書き戻された処理結果は、例えば図 1 の画像データ C として管理され、パターン認識などの処理に使用される。一方、ラインメモリ 105 に書き戻された処理結果は、再度、演算回路 106 に供給され、別の処理に使用される。

#### 【0049】

本実施形態の画像処理方式では、入力回路 104 が主記憶 209 からデータを取得して、ラインメモリ 105 にライン毎に記録して、演算回路 106 で画像処理する。ここで、ラインメモリ 105 は記憶領域として論理上直列な記憶ラインを複数有し、前記システム

10

20

30

40

50

レジスタ101等で指定された前記記憶ラインに入力データを書き込み書き込まれたデータの読出しが可能とされる。更に詳しくは、図7に例示されるように、ラインメモリ105は、メモリアレイ(MARY)70、データ入力バッファ(DIB)71、データ出力バッファ(DOB)72、及びアドレスやアクセス動作形態を制御するメモリ制御回路(MCNT)73によって構成される。図7においてメモリアレイ70はML0-MLnの記憶ラインを備える。ACUNT0-ACUNT4はメモリアレイ70に対するアドレスポインタとして機能するアドレスカウンタである。アドレスカウンタACUNT0-ACUNT4の値がメモリアレイ70に対するアクセスアドレスになる。夫々のアドレスカウンタACUNT0-ACUNT4の意義については後述する。

#### 【0050】

演算回路106はラインメモリから出力された単数又は複数の前記記憶ラインのデータを前記システムレジスタにより指定された処理内容で演算処理単位毎に演算して出力する。例えば、3×3ウィンドウ、5×5ウィンドウといった処理サイズに応じて、局所的な領域のデータを用いて一画素に対応する処理結果を取得するのに、一度演算処理単位で処理をする方式が一般的であり、この場合に、ラインメモリを使用して、一ラインを単位とする処理を行なうことが一般的であるので、ここでもそれに対応する。具体的にはk×k個の演算ユニットUNT及び夫々の演算ユニットUNTによる演算結果を加減算する加算器を有し、指定されたi×i個の演算ユニットUNTを用い、i本の記憶ラインから供給されるi×i個のデータを並列処理しその処理結果を加算して出力する、というような並列演算が可能にされる。

#### 【0051】

図1において、同期化回路103は、入力回路104から画像処理に使用するデータの転送開始と終了を画面毎、およびライン毎に受け取る。また、演算回路106からの処理結果有効信号を受け取り、ライン単位の処理に必要な処理の開始と終了を判断する同期信号を生成し、マイクロコントローラ102に出力する。マイクロコントローラ102では、一ラインの処理が終了すると次のラインの処理内容、ラインメモリからの読み出し箇所、処理結果の記録箇所などの指定を、システムレジスタ101の設定更新によって行い、次の一ラインの処理を制御する。

#### 【0052】

このとき、入力回路104は、システムレジスタ101内に記録されている主記憶209上の処理対象画像データのスタートアドレスを参照し、設定されたアドレスの範囲のデータを、バスインタフェース108経由で一ライン毎に読み込む。

#### 【0053】

以上の動作を纏めると、ラインメモリ105は、入力回路104よりデータを受け取り、システムレジスタ101により設定される動作モードに従って、ラインメモリ105にその値を保持し、さらには演算回路106にデータを出力する。演算回路106は、ラインメモリ105からデータを受け取り、システムレジスタ101で設定された演算種別の演算を行う。出力回路107は、演算回路106からのデータを受け取り、必要に応じてシフト処理等を行い、システムレジスタ101の設定に従って、演算回路107による処理結果をラインメモリ105へ書き戻し、又は、バスインタフェース108経由で主記憶209へ出力する。したがって、前記制御レジスタ101及びマイクロコントローラ102は、指定した記憶ラインのデータを記憶ライン単位で前記ラインメモリ105から前記演算回路106に出力させ、演算回路106はラインメモリ105から出力された単数又は複数の前記記憶ラインのデータを前記システムレジスタ101により指定された処理内容で演算処理単位毎に繰り返し演算する。

#### 【0054】

画像処理装置の更に具体例を説明する。図3はシステムレジスタ101の構成例である。図3において、システムレジスタ101は、マイクロコントローラ設定レジスタ(MCreg)301、同期回路設定レジスタ(SYNCreg)302、入力回路設定レジスタ(INreg)303、ラインメモリ設定レジスタ(LMreg)304、演算回路設

10

20

30

40

50

定レジスタ (A R r e g) 3 0 5、出力回路設定レジスタ (O U T r e g) 3 0 6、及びバシインタフェース設定レジスタ (B I F r e g) 3 0 7を有する。3 0 8はデータバス、3 0 9はアドレスバスを意味し、夫々のレジスタに対する書き込み経路を意味し、各種レジスタに設定された制御データが制御信号として対応する内部回路に伝達される制御信号経路については図示を省略してある。

【 0 0 5 5 】

マイクロコントローラ設定レジスタ 3 0 1 は、マイクロコントローラ 1 0 2 の動作開始等の指示を行ったり、マイクロコントローラ 1 0 2 のステータス情報を知るためのレジスタである。

【 0 0 5 6 】

同期回路設定レジスタ 3 0 2 は、同期を監視すべき対象についての、動作開始指示や同期とる (waitする) ことを指示するためのレジスタである。本実施形態が、一ライン毎に処理を実施し、一回の実施毎に処理の内容と演算結果の記録先を変更することができる。そのために必要な処理の開始、終了を一ライン毎に把握する機能に対する設定が行なわれる。

【 0 0 5 7 】

入力回路設定レジスタ 3 0 3 は、入力回路 1 0 4 が主記憶 2 0 9 へ出力するアドレスを生成するために必要な値を保持するためのレジスタである。具体的には、処理対象の画像データが主記憶 2 0 9 のどこに格納されているか、処理する画像データの横方向の画素数、縦方向の画素数などが設定される。

【 0 0 5 8 】

ラインメモリ設定レジスタ 3 0 4 は、ラインメモリの構成制御を行うレジスタである。ラインメモリ 1 0 5 は、処理内容や処理方式によってラインメモリを使用しない場合もある。実施形態では、汎用の画像処理装置を想定しており、ラインメモリの使用方法と、マイクロコントローラの制御を許可するかなどについて、必要な設定が行われる。

【 0 0 5 9 】

演算回路設定レジスタ 3 0 5 は、演算の種別を設定するためのレジスタである。画像処理機能や、それぞれの画像処理で必要になるパラメータを含んでいる。また、一部の処理結果に関しては、演算回路 1 0 6 の処理結果を格納するレジスタも存在する。

【 0 0 6 0 】

出力回路設定レジスタ 3 0 6 は、演算結果を出力する際に、ラインメモリ 1 0 5 へ出力するか、あるいはバシインタフェース B I F 経由で外部に出力するかを切替えたり、あるいは、演算結果をシフトダウンするといった処理の指示を指定するためのレジスタである。バシインタフェース設定レジスタ 3 0 7 にはバシインタフェースの動作に必要な設定が行われる。

【 0 0 6 1 】

図 4 にはマイクロコントローラ 1 0 2 の構成例が示される。図 4 において、4 0 1 はマイクロプログラム保持回路 (p r o g)、4 0 2 はプログラムカウンタ (P C)、4 0 3 ~ 4 1 7 はレジスタ R 0 - R 1 4、4 1 8 は命令デコーダ (I D)、4 1 9 は実行回路 (E X E) である。このマイクロコントローラ 1 0 2 は、プログラムによってシステムレジスタ 1 0 1 にアクセスをして、ラインメモリ 1 0 5 の制御、画像処理の機能の選択、パラメータの設定、出力データの格納場所を選択することができる。このマイクロコントローラ 1 0 2 がライン毎に制御内容を決定し、さまざまな処理を組み合わせ、ハードワイヤード方式で実装された演算回路 1 0 6 の機能を大幅に拡張している。

【 0 0 6 2 】

マイクロプログラム保持回路 4 0 1 は、主記憶 2 0 9 からマイクロプログラムを読み込み、保持するための回路である。プログラムカウンタ 4 0 2 は、マイクロプログラム保持回路 4 0 1 に格納されたプログラムのうち、現在実行中のアドレスを指すポインタである。レジスタ 4 0 3 ~ 4 1 7 は、マイクロプログラムの中で参照されるレジスタであり、一部は汎用的に使われ、また一部は特定の機能を有するレジスタである。命令デコーダ 4 1 8

10

20

30

40

50

は、現在実行中の命令を解釈するための回路である。実行回路 4 1 9 は、命令解釈により、制御信号を生成し、また、レジスタの更新値を生成するための回路である。

#### 【 0 0 6 3 】

図 5 には同期化回路 1 0 3 の構成例が示される。図 5 において、5 0 1 は入力同期回路 ( I N S y n c )、5 0 2 は演算処理同期回路 ( A R s y n c ) である。5 1 0 はマイクロコントローラ 1 0 2、システムレジスタ 1 0 1 とのインタフェース信号を代表し、5 1 1 はマイクロコントローラ 1 0 2 とのインタフェース信号を代表する。5 1 2 は入力回路 1 0 4 とのインタフェース信号を代表し、5 1 3 は演算処理回路 1 0 6 とのインタフェース信号を代表する。

#### 【 0 0 6 4 】

入力同期回路 5 0 1 は、入力回路 1 0 4 について、設定された同期を監視するための回路である。入力回路 1 0 4 は、主記憶 2 0 9 からデータを 1 ライン分読み込む回路である。入力回路 1 0 4 に指示することにより、1 ライン分読み込む動作の起動を行うことができる。また、入力回路 1 0 4 によって、1 ライン読み終えたことが検出されると、それが割り込み要求等としてマイクロコントローラ 1 0 2 に伝えられる。

#### 【 0 0 6 5 】

演算処理同期回路 5 0 2 は、演算処理回路 1 0 5 について、設定された同期を監視するための回路である。演算処理回路 1 0 5 において、1 ライン単位の演算を可能にするため、演算処理同期回路 5 0 2 によって、1 ライン分の演算の起動が指示され、1 ライン分の演算の完了が検出された場合には、演算処理同期回路 5 0 2 がその旨を割り込み要求等によってマイクロコントローラ 1 0 2 に伝える。

#### 【 0 0 6 6 】

図 6 には入力回路 1 0 4 の構成例が示される。図 6 において、6 0 1、6 0 3 はカウンタ ( I N C U N T )、6 0 2、6 0 4 は入力データ保持回路 ( I D R E G )、6 0 5 は同期判別回路 ( S Y N C D E T ) である。カウンタ 6 0 1、6 0 3 は、あらかじめ設定された初期値から最終値まで、カウントアップするカウンタであり、これが主記憶 2 0 9 に対するリードアクセスのアドレスとして出力される。カウンタ 6 0 1、6 0 3 には画像処理の対象となる画面先頭 ( もしくはライン先頭 ) のアドレスが初期設定され、それを初期値として順次インクリメント動作を行って主記憶 2 0 9 上の画像データのメモリアドレスを生成する。生成されたメモリアドレスによって主記憶 2 0 9 から読出されたデータは入力データ保持回路 6 0 2、6 0 4 に順次保持される。同期判別回路 6 0 5 は 1 ラインのデータがデータ非時回路 6 0 2、6 0 4 に保持されたことをメモリアドレスから判別し、その結果を同期化回路 1 0 3 に与えてラインメモリ 1 0 5 に記憶ライン単位でデータを取り込み可能にする。ここで、入力データ保持回路 6 0 2、6 0 4、カウンタ 6 0 1、6 0 3 の夫々は、対用意されているのは、画像処理によっては、異なる画像間の演算が必要になるためである。ラインメモリが、並列動作可能なハードウェアを対備えている場合には、異なる画像を並列にラインメモリに蓄積して、演算処理を実行させることができる。6 1 2 はマイクロコントローラ 1 0 2 及びシステムレジスタ 1 0 1 に接続する信号線を代表的に示し、6 1 3 は同期化回路 1 0 3 に接続する信号線を代表的に示し、6 1 1 1 はラインメモリに接続する信号線を代表的に示し、6 1 0 はバスインタフェース 1 0 8 に接続する信号線を代表的に示す。

#### 【 0 0 6 7 】

図 7 はラインメモリ 1 0 5 の構成例を示す。記憶ライン M L 0 ~ M L n の規模は論理的に可変であり、例えば、1 0 2 4 バイト × 4 本、又は 1 2 8 バイト × 3 2 本という構成を選択できる。演算処理モードの場合、ラインメモリの外からのデータを書きこむ場所 ( どの記憶ラインか ) やラインメモリの外に出力するデータの場所 ( どの記憶ラインか ) の初期値は、マイクロコントローラ 1 0 2 で制御可能なシステムレジスタ 1 0 1 により設定される。また、あるライン上のどこに書き込まれるか等のライン上のアドレスは、ラインメモリ内にあるカウンタ A C U N T 0 ~ A C U N T 4 がその動作形態に応じて指定する。ラインメモリは、通常書き込みと読み出しを並行可能なデュアルポートメモリによって構成

10

20

30

40

50

されており、画像データを読み出しながら、処理結果を書き込むことが可能である。そのために必要なアドレス生成や動作モードの制御はアクセスコントローラ 73 が行う。アクセスコントローラ 73 は記憶ライン  $ML0 \sim MLn$  を単なる FIFO 動作させる動作形態の選択も可能になっており、その選択はシステムレジスタ 101 の設定で決まる。

#### 【0068】

図 8 には演算回路 106 の構成例が示される。演算回路 106 は、通常画像処理モードの場合には、画像処理演算のために  $3 \times 3$  のフィルタ演算や  $5 \times 5$  のフィルタ演算が可能になっている。図 8 では、一つの矩形が一つの演算機能を果たす回路（演算ユニット UNIT）であり、例えばフィルタ処理の場合、一つの演算ユニット UNIT ブロックは積和演算を実行することができる。画像処理機能の設定やパラメータ設定は、システムレジスタ 101 で制御され、様々な処理が実装可能である。また、演算処理モードの場合には、前述のフィルタ演算のための演算回路の構成を論理的に変更し、例えば  $5 \times 5$  で合計 25 個の積和回路がある場合、最大 25 並列までの積和演算が可能となる。この演算回路 106 は、ハードワイヤードロジック方式で実装されることが一般的であるが、ラインメモリの構成やフィルタのパラメータ、専用ハードウェアなどを実装することで汎用的な画像処理機能を実現することができる。その回路規模は、汎用のプロセッサを複数搭載して同等性能を実現するよりも十分に小さいハード物量で実現が可能である。

#### 【0069】

図 9 には出力回路 107 の構成例が示される。出力回路 107 はシフタ（SHIFT）901 とセクタ（SLCT）902 を有する。910 はマイクロコントローラ 102 及びシステムレジスタ 101 に接続される信号を線を代表的に示す。911 は演算回路 106 回路による演算結果を受取るための信号線を代表的に示す。912 はバッファインタフェース 108 への接続経路、913 はラインメモリ 105 への書き戻し経路である。シフタ 901 は、演算結果のシフトダウンを行うための回路である。例えば、 $8\text{bit} \times 8\text{bit}$  の演算を行った場合に、その結果は  $16\text{bit}$  となるが、それを  $8\text{bit}$  の数値としてまとめた場合等に利用する回路である。セクタ 902 は、演算結果をラインメモリ 105 に書き戻すか、あるいは、バスインタフェース 108 を通じて主記憶 209 に書き戻すか、を選択する回路である。ここでライン毎に処理結果をラインメモリ 105 に書き戻す機能を実現することによって、処理結果を演算途中の結果として利用することが可能となり、マイクロコントローラ 102 によって基本的な画像処理機能を組み合わせて、高度な画像処理を実現することができる。

#### 【0070】

図 10 にはマイクロコントローラの命令セットが抜粋される。

#### 【0071】

MV 命令は、データの転送命令である。データを第 1 オペランドのレジスタから第 2 オペランドのレジスタへ、或いは、値をレジスタへ、又は、レジスタの値をラベルが示すアドレスへ、それぞれ転送する。レジスタのデータコピー等にも利用される。

#### 【0072】

ADD 命令は、第 1 オペランドのレジスタの値と第 2 オペランドのレジスタとの和を計算し、それを第 2 オペランドのレジスタに代入する命令である。レジスタのデータ加算等にも利用される。

#### 【0073】

CMP 命令は、第 1 オペランドのレジスタの値と第 2 オペランドのレジスタの値とを比較し（差分をとり）、等しいか否かを条件フラグに反映させる命令である。この条件フラグは後述する分岐命令で利用される。

#### 【0074】

ST 命令は、レジスタのデータをストアする命令であり、画像処理装置がアクセス可能なレジスタ空間のアドレスにレジスタの値を書きこむための命令である。

#### 【0075】

LD 命令は、レジスタにデータをロードする命令であり、画像処理装置がアクセス可能

10

20

30

40

50

なレジスタ空間のアドレスに記録された値を、指定されたレジスタに読み込むための命令である。

【 0 0 7 6 】

B T 命令は、先に述べた、条件フラグが真のときに設定されたアドレスに分岐する命令である。

【 0 0 7 7 】

S N C 命令は、ライン毎に変化する信号を設定し、同期を採る命令であり、ライン毎に演算機能を切り替えるための同期をとるのに用いることができる。具体的には、オペランドに同期する処理を行なう機能ブロックを記述することにより、記述した対象の機能ブロックが処理をライン単位で処理終了する同期を監視するための命令である。その対象物のライン単位の動作が終わるまで、マイクロプログラムの実行はウェイトされる。なお、ここでいう機能ブロックとは、入力回路、演算回路を指す。例えば、画像処理を行う回路において、一ライン単位のデータに対する実行が終了されるのを待つのに利用する。

【 0 0 7 8 】

E X E 命令は、そのオペランドに機能ブロックを記述することにより、記述した対象の機能ブロックでライン単位の処理を実行するための命令である。記憶ライン単位の画像データの取り込み、又は画像処理の実行に用いられる命令であり、画像処理を行う回路において、一ライン単位の実行を指示する命令として位置付けることができる。

【 0 0 7 9 】

I N T 命令は、マイクロコントローラ 1 0 2 が必要に応じて、上位の C P U 2 0 8 に割り込みを発生するための命令である。上位の C P U 2 0 8 では、画像認識の処理を行っており、画像処理装置 2 0 1 とは独立に動作する。そのため、画像処理の要求が上位 C P U 2 0 8 で発生し、画像処理装置 2 0 1 が画像処理を行い、その終了を画像処理装置 2 0 1 が C P U 2 0 8 に通知する必要がある。この通知をマイクロコントローラ 1 0 2 がプログラム動作によって発生することにより、より効率的な並列処理が可能となる。例えば、従来はハードウェアによる終了通知だったため、予め設計されたタイミング以外では通知をすることができなかった。予めの設計段階で、全ての想定される画像処理を考えることは困難であるため、新しい機能を実現した際に対応できず無駄が発生する可能性がある。この例では、マイクロプログラムによって制御することにより、様々な演算を実装することができる。そして、条件分岐などを実行し、ある場面で上位 C P U 2 0 8 に割り込み発生をかけて、画像処理結果に対する処理や処理中の演算実行を中止するなどの、適切な処理をプログラムすることが可能となる。例えば、C P U 2 0 8 と画像処理装置 2 0 1 が独立に処理を実行しなければならないとき、画像処理装置が処理中の任意の状態を、C P U 2 0 8 に通知して、画像処理装置 2 0 1 の演算終了を待たずに、C P U 2 0 8 が処理結果に対するアクセスを開始することができる。

【 0 0 8 0 】

図 1 1 にはマイクロコントローラのプログラムで利用できるレジスタの機能が例示される。マイクロコントローラ 1 0 2 がライン毎に演算回路 1 0 6 の機能を変更できることを実現するために、記憶ライン毎のメモリアクセスの情報、画像処理の演算機能の設定、画像処理の演算に必要なデータのライン数などを記録するレジスタを保有している。

P C は、プログラムカウンタであり、現在実行中の命令の場所を示す特別なレジスタ。

R 0 ~ R 7 は、汎用レジスタであり、マイクロプログラム中での一時データの保持に使えるレジスタである。

R 8 は、ラインメモリのリードポインタのためのレジスタであり、ソースの A 側のレジスタである。

R 9 は、ラインメモリのリードポインタのためのレジスタであり、ソースの B 側のレジスタである。

R 1 0 は、ラインメモリに予め画像処理に必要なデータを記録しておく情報量を指定するレジスタである。

R 1 1 は、画像処理に必要な画像データのライン数が複数本ある場合に、マイクロプログ

10

20

30

40

50

ラムがそれを把握するために必要なデータを記録するレジスタである。例えば、必要なライン数であったり、現在読み込んでいるラインメモリのライン位置であったりする。

R 1 2 は、入力回路 1 0 4 からのデータ取り込み先の A 側用であり、R 1 3 は入力回路 1 0 4 からのデータ取り込み先の B 側用である。

R 1 4 は、出力回路 1 0 7 からのデータ取り込み先である。

R 1 5 は、スタックポインタである。

#### 【 0 0 8 1 】

図 1 2 には画像処理装置 2 0 1 を用いた処理の一例として画像認識処理の流れが例示される。図 1 2 では、原画像に対して、ノイズ除去を目的として、 $5 \times 5$  ウィンドウの平滑化処理を行い、その結果画像に対して、輪郭強調処理を行う。輪郭強調処理では、 $3 \times 3$  ウィンドウとする。最後に輪郭強調された結果画像に対して二値化処理を行い、物体の輪郭を抽出する。このような処理は画像認識では、頻繁に行われる一連の処理である。平滑化処理は  $5 \times 5$  ウィンドウの画像データのピクセル値を平均化して、中心ピクセルの値とするコンボリューション演算によって行われる。この平滑化処理においては  $5 \times 5$  ウィンドウのデータが演算処理単位のデータとなる。

#### 【 0 0 8 2 】

図 1 4 には図 1 2 の画像処理におけるラインメモリの利用状態が示される。ラインメモリ 1 0 5 の記憶ライン  $MLi \sim MLi+4$  の 5 ラインに原画像のデータを格納して  $5 \times 5$  の平滑化処理を行い、平滑化処理結果の格納にはラインメモリ 1 0 5 の記憶ライン  $MLj \sim MLj+2$  の 3 ラインを用い、3 ライン単位で輪郭強調を行い、輪郭強調された画像データの格納には記憶ライン  $MLk$  を用い、当該記憶ライン  $MLk$  単位で 2 値化を行う。2 値化された画像データはライン単位で記憶ライン  $MLm$  以下に格納され、所要のタイミングで主記憶 2 0 9 に転送される。

#### 【 0 0 8 3 】

例えば、記憶ライン  $MLi \sim MLi+4$  に画素データライン  $PXL0 \sim PXL4$  が格納されるとすると、それに対して  $5 \times 5$  の平滑化処理のためのコンボリューション演算が行われ、演算結果はその中心画素の画像データとして記憶ライン  $MLj$  に格納され、順次、 $5 \times 5$  の演算処理単位を 1 ピクセルずつ右方向にずらしながらコンボリューション演算を行ってその演算結果を記憶ライン  $MLj$  の隣の画素位置に格納していく、という演算を記憶ライン  $MLi \sim MLi+4$  の右端まで行う。この状態で  $MLj$  には画素データライン  $PXL2$  における平滑化データ  $CMB2$  が取得される。この次は、不用になった画素データライン  $PXL0$  のデータを無効にして  $MLi$  に、次の画素データライン  $PXL5$  のデータを格納し、今度は画素データライン  $PXL1 \sim PXL5$  に対して同じく平滑化処理を行なって、 $MLj+1$  に画素データライン  $PXL3$  における平滑化データ  $CMB3$  を取得する。この次も同様に、不用になった画素データライン  $PXL1$  のデータを無効にして  $MLi+1$  に、その次の画素データライン  $PXL6$  のデータを格納し、今度は画素データライン  $PXL2 \sim PXL6$  に対して同じく平滑化処理を行なって、 $MLj+2$  に画素データライン  $PXL4$  における平滑化データ  $CMB4$  を取得する。

#### 【 0 0 8 4 】

3 画素ライン分の平滑化データ  $CNB2 \sim CMB4$  が取得されたところで、今度は、それに対して  $3 \times 3$  の輪郭強調処理のための演算が行われ、演算結果はその中心画素の画像データとして記憶ライン  $MLk$  に格納され、順次、 $3 \times 3$  の演算処理単位を 1 ピクセルずつ右方向にずらしながら輪郭強調演算を行ってその演算結果を記憶ライン  $MLk$  の隣の画素位置に格納していく、という演算を記憶ライン  $MLk$  の右端まで行う。この状態で  $MLk$  には画素データライン  $PXL3$  における輪郭強調データ  $EMP3$  が取得される。

#### 【 0 0 8 5 】

輪郭強調データ  $EMP3$  が取得されたところで、今度は、それに対して 2 値化を行って記憶ライン  $MLm$  に格納する。以上の動作を順次繰り返して、記憶ライン  $MLm$  以下に 2 値化データを蓄積していく。

#### 【 0 0 8 6 】

上記処理において、図7のアドレスカウンタACUNT0, ACUNT1は記憶ラインMLi~Mi+4に順次右方向へ画素データを書き込むためのアドレスを生成するライトポイントとして機能される。ACUNT0とACUNT1の2個のカウンタを備えるのは複数ウィンドウの画像データを並列に書き込む場合を想定する。アドレスカウンタACUNT2は記憶ラインMLi~Mi+4から順次右方向に5画素の画素データを読み出すためのアドレスを生成するリードポイントとして機能される。アドレスカウンタACUNT3は記憶ラインMjに平滑化データを順次書き込む画素位置を生成するアドレスポイントとして機能される。アドレスカウンタACUNT4は記憶ラインMLj~Mj+2から順次右方向に3画素の画素データを読み出すためのアドレスを生成するリードポイントとして機能される。図示は省略するが、その他に、記憶ラインMLk, MLm等に対するライトポイントやリードポイントとして機能されるアドレスカウンタも設けられている。記憶ラインの指定はシステムレジスタ101への設定で決まる。

#### 【0087】

尚、2値化処理の演算結果を直接出力回路から主記憶209に出力してもよい。このときには、2値化演算結果を一時的に出力回路107のシフタ901に一時的に蓄積してゆき、1ライン分蓄積されたところで、主記憶209に向けて書き込み動作を行えばよい。

#### 【0088】

図15には本実施形態で説明したマイクロプログラムで制御する処理フローを示す。処理の開始前に、CPU208に制御によってマイクロプログラムをマイクロプログラム保持回路401に格納する。そして、画像処理を行う段になって、CPU208から起動の指示をレジスタ設定によって行うことで処理を開始する。マイクロプログラムに記述されている最初の命令は、5×5ウィンドウの平滑化処理を行うため、ラインメモリの5本の記憶ラインに処理対象の画像データを格納するところから始まる。これは、入力回路104で指定された処理開始のアドレスから、データを読み込み(S01)、図11で説明したR12, R13のレジスタで指定したラインメモリのアドレスに格納される。マイクロコントローラ102は同期をとるコマンドSNC(図10参照)で、5ラインの入力を待つ。このとき、1ラインずつ待つプログラムでも良いし、5ライン纏めて待つようにすることも可能である。5ラインの読み込み終了待ちの処理ステップ(S02)を終えて、ラインメモリに5×5ウィンドウの平滑化処理が可能なデータが格納されたら、平滑化処理を起動する(S03)。このとき、マイクロコントローラ102は、MVやLDといった命令を用いて、演算回路106が処理する画像処理機能を平滑化処理に設定する。マイクロコントローラは処理を起動した後、同期をとるコマンドSNCで再び、1ライン終了を待つ。平滑化処理の結果は、図11のR14に指定したラインメモリに格納する。平滑化処理を1ライン処理している間に、次の1ラインを読み込む(S04)。次の処理がデータを3ライン必要とする3×3ウィンドウの輪郭強調処理となっているため、平滑化処理を3ライン終了するまで、1ラインずつ処理を繰り返す(S05)。3ライン処理が終了した段階で、輪郭強調処理が可能となるため、マイクロコントローラ102はレジスタアクセスのコマンドを使用して、演算回路106の機能を輪郭強調処理に変更し、1ライン分輪郭強調処理を実行する(S06)。このときも平滑化処理の設定と同様にR14のレジスタに設定したラインメモリに結果を格納する。その結果に対して、マイクロコントローラ102は再度演算回路106の機能を二値化処理に変更し処理を行う(S07)。処理結果は、出力回路107を経由して、主記憶RAM209に格納される。こうして、1ライン分の一連の処理が終了する。必要ラインに対して全て処理を行っていないとき(S08のNo)、マイクロコントローラ102は、再度、主記憶RAM209から次の1ライン分のデータをラインメモリに格納する(S09)。既にこの一覧を読み込んだ際は、5×5ウィンドウの平滑化処理を実行可能なため、新しい5ライン分のデータが揃った時点で平滑化処理を開始する。平滑化処理を1ライン処理した段階で、輪郭強調処理も3ライン分のデータが既に格納されたこととなり、次の処理をする準備が整っているので、マイクロコントローラによって演算回路106の機能を切り替えて輪郭強調処理を実行する。さらに、引き続き二値化処理を実行し、新たな1ライン分の処理結果を主記憶RAM209に格納

10

20

30

40

50



する。１ライン処理が終了したので、再度同様の処理を繰り返し行い、必要ライン数処理しかたをチェックして（Ｓ０８）、処理を続ける。必要ライン処理した場合は、その後処理終了して、図１０のＩＮＴ命令によってＣＰＵ２０８に割り込みを発生し、一連の画像処理が終了したことを伝える。

【００８９】

以上のように、ラインメモリ２０１は記憶ライン単位という制約を受けて外部からデータを入力し、入力する記憶ライン数と記憶ラインの位置をシステムレジスタ１０１の設定値及びマイクロコントローラ１０２の制御回路によりプログラマブルにでき、演算回路１０６はラインメモリ１０５から供給される単数又は複数の記憶ラインのデータ単位で演算を行うという制約を受け、そのデータ単位に対する演算処理単位の演算処理内容を制御回路によりプログラマブルに指定することができる。したがって、画像処理装置２０１には、ハードワイヤードロジック方式の高効率な処理と、汎用プロセッサの柔軟性をもった処理を同時に実現することができる。

【００９０】

以上本発明者によってなされた発明を実施形態に基づいて具体的に説明したが、本発明はそれに限定されるものではなく、その要旨を逸脱しない範囲において種々変更可能であることは言うまでもない。

【００９１】

例えば、上記説明では画像認識を画像処理の一例として説明したが本発明おはそれに限定されない。また、画像認識処理として平滑化、輪郭強調、２値化を一例としたが、これについても本発明は限定されない。画像処理装置は１チップのアクセラレータとして構成されてもよい。システムＬＳＩにオンチップされる回路モジュールは図１に限定されない。

【図面の簡単な説明】

【００９２】

【図１】図１は本発明に係る画像処理装置を例示するブロック図である。

【図２】図２は本発明に係るデータプロセッサを例示するブロック図である。

【図３】図３はシステムレジスタの構成を例示するブロック図である。

【図４】図４はマイクロコントローラの構成を例示するブロック図である。

【図５】図５は同期化回路の構成を例示するブロック図である。

【図６】図６は入力回路の構成を例示するブロック図である。

【図７】図７はラインメモリの構成を例示するブロック図である。

【図８】図８は演算回路の構成を例示するブロック図である。

【図９】図９は出力回路の構成を例示するブロック図である。

【図１０】図１０はマイクロコントローラの命令セットを抜粋して示す説明図である。

【図１１】図１１はマイクロコントローラのプログラムで利用できるレジスタの機能を例示する説明図である。

【図１２】図１２は画像処理装置を用いた処理の一例として画像認識処理による大まかな処理の流れを例示する説明図である。

【図１３】図１３はデータプロセッサのメモリ空間を示すメモリマップ図である。

【図１４】図１４は図１２の画像処理におけるラインメモリの利用状態を示す説明図である。

【図１５】図１５は画像処理装置による画像認識処理の動作を例示するフローチャートである。

【符号の説明】

【００９３】

１０１ システムレジスタ

１０２ マイクロコントローラ

１０３ 同期化回路

１０４ 入力回路

10

20

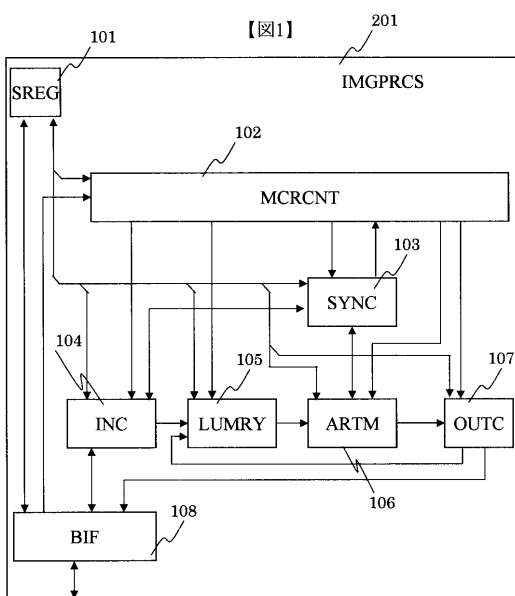
30

40

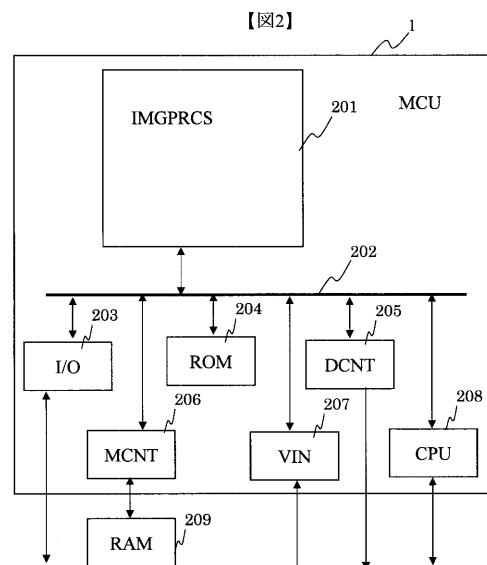
50

- 1 0 5    ラインメモリ
- 1 0 6    演算回路
- 1 0 7    出力回路
- 1 0 8    バスインタフェース
- 2 0 1    画像処理装置
- 2 0 2    チップ内バス
- 2 0 3    周辺インタフェース
- 2 0 4    リードオンリーメモリ (ROM)
- 2 0 5    表示回路
- 2 0 6    主記憶インタフェース
- 2 0 7    ビデオ入力回路
- 2 0 8    C P U

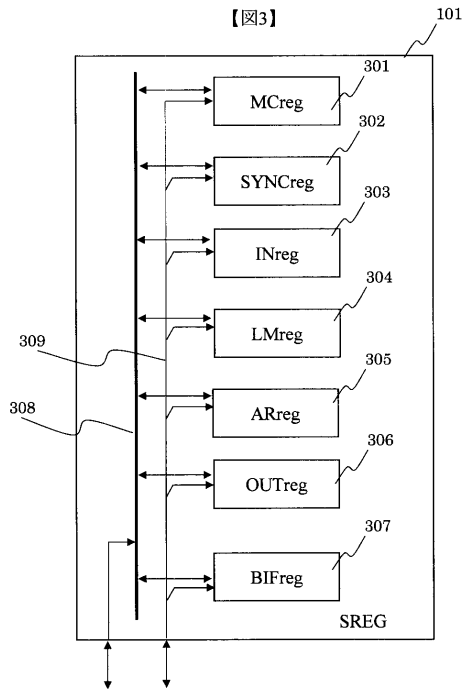
【図 1】



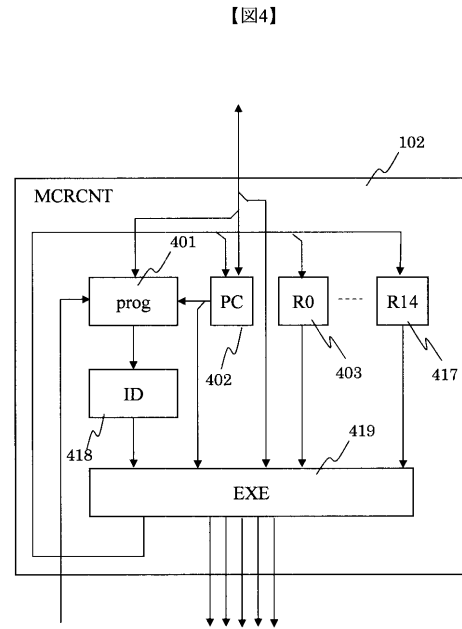
【図 2】



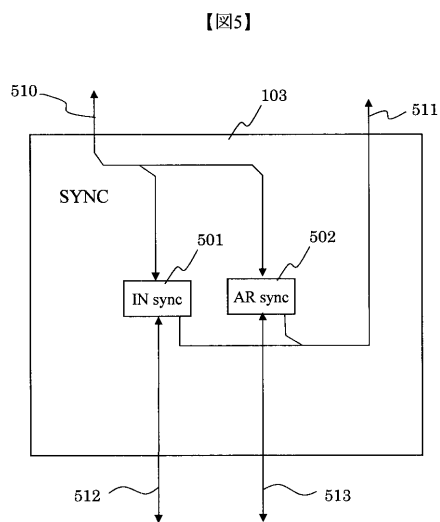
【 図 3 】



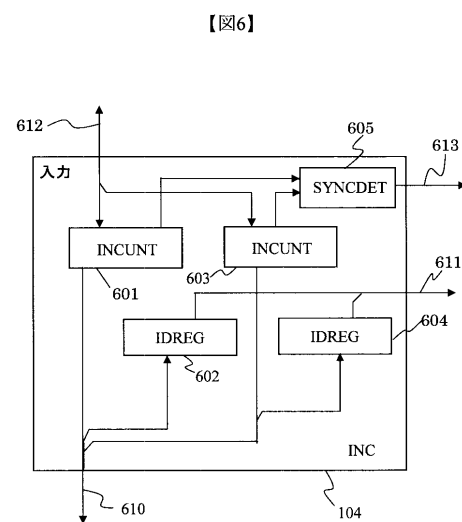
【 図 4 】



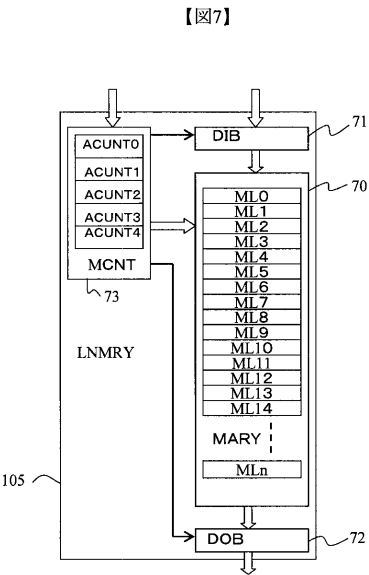
【 図 5 】



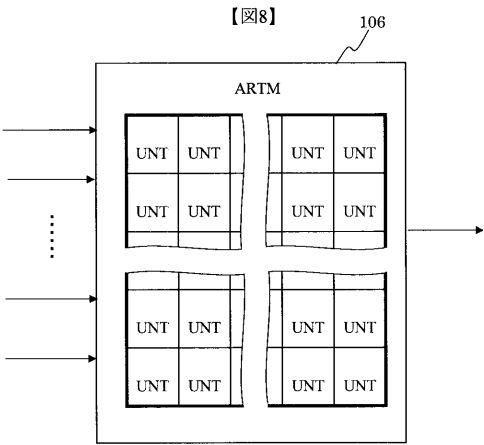
【 図 6 】



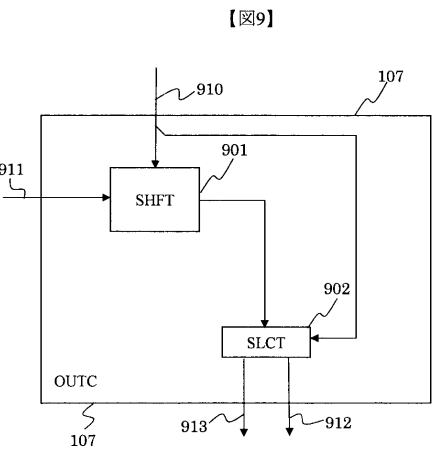
【 図 7 】



【 図 8 】



【 図 9 】



【 図 1 0 】

【図10】

MV	（レジスタのデータコピー）
ADD	（レジスタのデータを加算）
CMP	（レジスタのデータ比較）
ST	（レジスタのデータをストアする）
LD	（レジスタにデータをロードする）
BT	（分岐でプログラムカウンタにジャンプする）
SNC	（ライン毎に変化する信号を設定し、同期を取る）
EXE	（ライン毎の画像データの取込または、画像処理を実行）
INT	（上位プロセッサに割り込み発生）

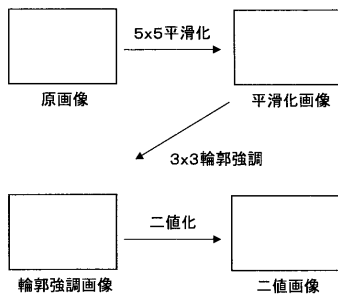
【図 1 1】

【図11】

PC	: プログラムカウンタ
R0	: 0の値を保持するレジスタ
R1	: 汎用レジスタ
R2	: (同上)
R3	: (同上)
R4	: (同上)
R5	: (同上)
R6	: (同上)
R7	: (同上)
R8	: ラインメモリのリードポインタ(A側用)
R9	: ラインメモリのリードポインタ(B側用)
R10	: 複数ラインリードするためのサイズ指定レジスタ
R11	: 複数ラインリードした際に必要なラインメモリ制御を コントロールするためのレジスタ
R12	: 入力回路からのデータ取り込み先(A側用)
R13	: 入力回路からのデータ取り込み先(B側用)
R14	: 出力回路からのデータ取り込み先
R15	: スタックポインタ

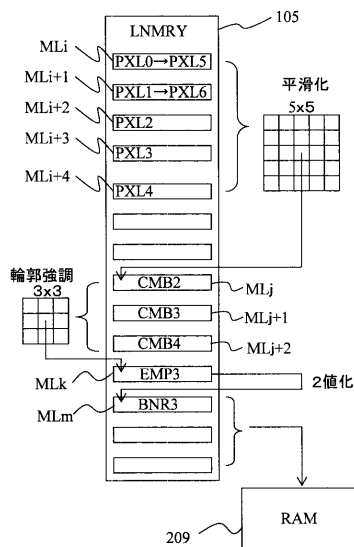
【図 1 2】

【図12】



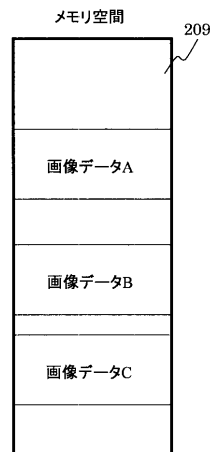
【図 1 4】

【図14】



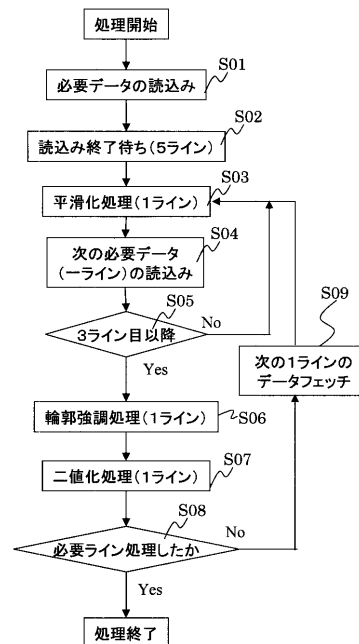
【図 1 3】

【図13】



【図 1 5】

【図15】



---

フロントページの続き

(72)発明者 浜崎 博幸

東京都千代田区大手町二丁目 6 番 2 号 株式会社ルネサステクノロジ内

(72)発明者 大塚 総

東京都千代田区大手町二丁目 6 番 2 号 株式会社ルネサステクノロジ内

F ターム(参考) 5B057 CH02 CH11

5B060 AB17 AC13