

1C-8

画像処理用DMAコントローラ(IPDMAC)

山根康邦 泉正夫 野口要治 高倉正樹 賀好宣捷
(シャープ株式会社 中央研究所)

1. はじめに

LSI技術の急速な進歩により、マイクロプロセッサの高機能化が進んでおり、画像処理分野への適用も十分可能になってきた。しかし、画像データそのものを扱う信号レベルの処理(画像の前処理)では、複雑な処理は要求されない反面、扱うデータ量が膨大であるため汎用マイクロプロセッサでは高速化が難しい。[1, 2]

今回、我々は従来のDMA(Direct Memory Access)コントローラに、2次元アドレス走査機能、データ演算機能等をハードウェア・モジュールとして付加することにより、画像メモリ上の矩形領域のデータ転送や種々の画像の前処理を高速に実行する画像処理用DMAコントローラ(Image Processing DMA Controller: IPDMAC)を開発したので以下に報告する。

2. IPDMACの概要

従来、DMAコントローラはCPUによるプログラム転送では時間的に間に合わない高速なデータ転送に用いられてきた。したがって、1次元的なアドレス走査によるデータ転送しか考慮されておらず、例えば、画像メモリ上の矩形領域のデータ転送はできず、また転送データに演算処理を施すための演算機能は持っていなかった。

IPDMACは、本稿で述べる諸機能をハードウェア・モジュールとして付加することにより、画像処理において処理データ量の多い前処理を分担し、汎用マイクロプロセッサによる上位レベルの処理と合わせて全体でのスループットの向上を目指している。また、IPDMACは、図1に示すようにシステム・バスに直結できるので、既存コンピュータシステムへのインストールが非常に容易に行えるという利点を持つ。

なお、本稿では単純化のため、IPDMACは1アドレス=1画素の濃淡画像データを扱うものとして話を進めるが、勿論カラー濃淡画像の処理も可能である。

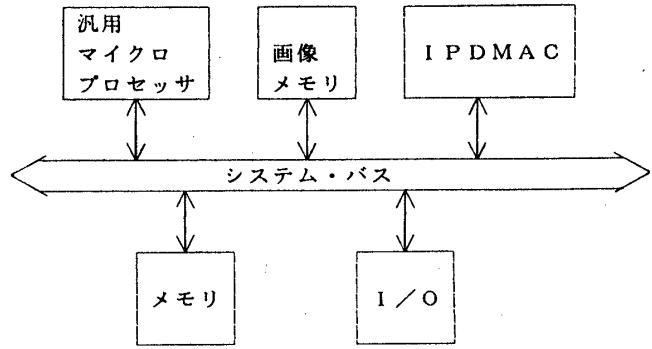


図1 IPDMACの接続例

3. IPDMACの構成

IPDMACは、図2に示すように、(1)バス・インタフェース、(2)レジスタ群、(3)2次元アドレス発生部、(4)データ演算部、(5)データ蓄積部、(6)転送条件判定部、(7)タイミング制御部などのハードウェア・モジュールにより構成される。

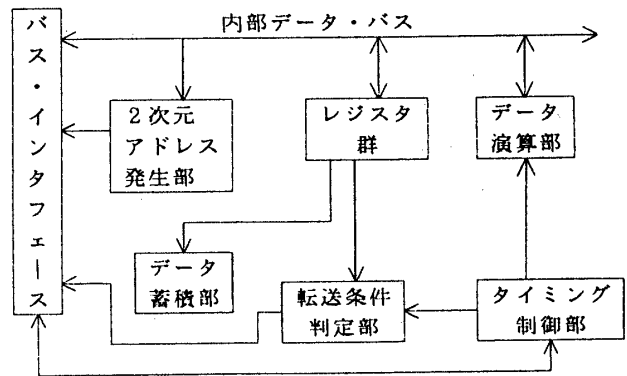


図2 IPDMACの構成

バス・インタフェースは、IPDMACをシステム・バスに接続するものである。DMA転送時はバス・マスタ、それ以外の時はバス・スレーブとなるように働く。

レジスタ群は、IPDMACにより様々な形態の画像データ転送・処理を柔軟に行わせるためのものである。転送開始アドレス、アドレス走査形式、矩形領域サイズ、画像サイズ、転送条件、演算の種類などをDMA転送実行前にホストCPUから設定する。

2次元アドレス走査部は、IPDMACによる矩形領域データ転送を可能にするものである。主走査方向と副走査方向の組み合わせにより、8通りのアドレス走査形式が可能である。

データ演算部は、濃度変換等の非線形演算に威力を発揮するルックアップテーブルRAM、2つのソース矩形領域からアクセスされたデータ間の演算処理を行うための算術論理演算回路等から成る。

データ蓄積部は、指定領域全体をアクセス後、濃度データの最大値・最小値や累積値を内部レジスタに残す機能を持つ。

転送条件判定部は、ソース領域からアクセスされたデータのビット条件や濃度範囲条件により、デスティネーション領域への転送可否を判定する機能を持つ。

4. IPDMACの動作と適用処理例

IPDMACの動作モードは、スレーブモードおよびマスタモードの2つに大別される。スレーブモードでは、バスの制御権はホストCPUに委ねられる。ホストCPUからIPDMACの各レジスタおよびルックアップテーブルRAMへのデータ設定をこの期間に行う。マスタモードは、IPDMACのコマンドが起動してから終了するまでの期間である。IPDMACはバスの制御権を得たことを確認した後、DMA転送を行う。

IPDMACのコマンドとして次の4つの基本コマンドが考えられる。

(1) リードコマンド

指定された矩形領域内のメモリ内容を読み出し、累積値等の結果を内部レジスタに残す。

[応用例：周辺分布計算、面積計算他]

(2) ライトコマンド

指定された矩形領域内のメモリに対し、レジスタに設定されたデータを書き込む。

[応用例：画面クリア]

(3) コピーコマンド

指定された矩形領域内のメモリの内容を読み出し、別の矩形領域へ書き込む。読み出しデータに対し、ルックアップテーブルRAMを用いた演算処理、データの内容に応じた転送先への書き込み禁止、ソースとデスティネーションで異なる走査形式の選択などが可能である。

[応用例：濃度変換、2値化処理、画像の反転他]

(4) 演算コマンド

2つの矩形領域の対応するメモリから読み出したデータ間の演算を行い、結果を別の指定矩形領域に書き込む。(3)と同様の操作が可能である他、一方のソース領域を別のソース領域のデータ転送条件として使用することも可能である。

[応用例：画像間演算、画像合成他]

これらの基本コマンドを適当に組み合わせ、画像メモリのアクセスを数回繰り返すことにより、画像の拡大・縮小や空間フィルタリング処理等にも適用できる。

我々は、システムバスをマルチバスとするIPDMACを設計・試作し、上述した各種処理が実現できることを確認した。また、画像メモリのアクセス速度の影響を含めても、M68000CPU(8MHz動作)に比べて最高15倍の処理速度が得られることを確認した。

5. おわりに

単なるデータ転送だけでなく画像処理機能を合わせ持ったIPDMACの構成、機能等について述べた。画像処理が普及しパソコンベースの画像処理システムも出現しているが、IPDMACは既存コンピュータシステムに容易に付加できるという特徴を持っている。

今後の課題としては、より一層の高機能化・多機能化およびLSI化の検討がある。

6. 参考文献

- [1]G.R.Nudd,"Image Understanding Architectures", Proc. of National Computer Conference, pp.377-390 (1980)
- [2]天満勉,"画像処理用のプロセッサ",情報処理,25,9, pp.909-917 (1984)