

解説



リアルタイム映像信号処理装置とその応用†

八木伸行††

1. はじめに

世は、マルチメディアということ、パソコンやワークステーション上に動画を表示し編集まで行うことが、一般的になりつつある。それは、ひとえにマイクロプロセッサが速くなってきたからできたことである。ついこの間までは、わずか8色で静止画しか表示できなかったのであるが、それと比べると隔世の感がある。と言っても、パソコン、ワークステーションで表示できる動画は、特殊なハードウェアがなければ、解像度が足らなかったり、コマ数が少なかったり、日頃 TV 放送で見ているものに比べ、明らかに低品質である。

では、高品質の TV 放送ではどんな装置を使って処理を行っているのだろうか。コンピュータのような汎用装置ではなく、用途ごとの専用装置を使って行うのが通例である。それは、本稿のタイトルにあるリアルタイムというキーワードに関係する。日頃、映像信号処理をされている方には、当然のことかもしれないが、ここで、リアルタイム処理の意味について確認しておく。本稿でいうリアルタイム処理とは、コマンドに対して待ち時間なく処理結果が表示されることや、単に1枚の画像が高速に処理されることとは違う。TV 信号は、ご存じのとおり、毎秒約30コマの速度で連続して画像を表示することで動く画像が作られるわけであるが、この信号を処理のための遅れをとまなうが、毎秒約30コマのレートで処理結果を途切れることなく、出力し続けることをリアルタイム処理と呼ぶ。ビデオレート処理と呼ぶこともある。

従来は、このリアルタイム処理を実現するのが困難であったため、処理に合わせた専用装置を開発してきた。しかしながら、近年の LSI 技術の進展とともに高速プロセッサが実現可能になり、ソフトウェアによる映像信号のリアルタイム処理も夢ではなくなりつつある。ハードウェア（ワイヤードロジック）のソフトウェア化は、時代の流れに沿うものであり、すでに DSP (Digital Signal Processor) が開発され、音声、通信、画像などと次第に応用分野を広げている。しかしながら、映像信号に適用するためには、速度、機能とも、まだ十分とは言いがたく、並列処理などのアーキテクチャの工夫が必要である。

本稿では、プログラム可能なリアルタイム映像信号処理システムを構成する上で考慮すべき点について概説した上で、システム例、放送での実用例について述べる。

2. 映像信号処理指向アーキテクチャ

2.1 実時間映像信号処理の特質

映像信号は、図-1 に示すように画像を左から右へ上から下へ、しかも1ライン（走査線）おきの実線に沿って走査し、第1フィールド画像が作られ、次に飛び越されたラインを走査して第2

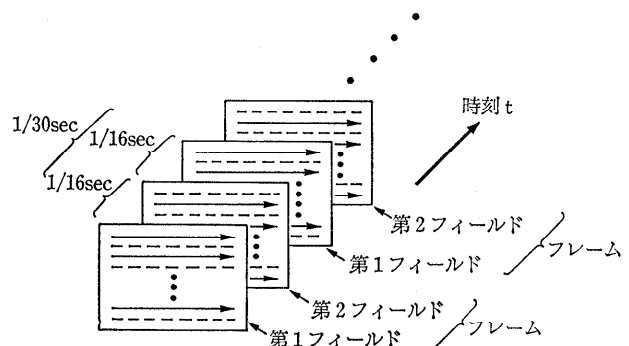


図-1 映像信号の構成

† Real-Time Video Signal Processor and Its Application by Nobuyuki YAGI (NHK Science and Technical Research Laboratories).

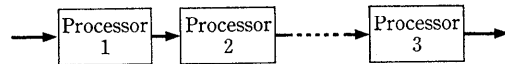
†† NHK 放送技術研究所

フィールド画像が作られている¹⁾。この2フィールドを合わせて、1枚の完全な画像(フレーム)が作られていることは、ご存知のことと思う。このシーケンスが繰り返されて、動画を送ることができるわけである。デジタル映像信号処理では、この映像信号をA/D変換し、デジタル処理回路で処理を加え、処理結果をD/A変換して、映像信号に戻す。専用機器のデジタル処理回路は、処理の系統に合わせて演算要素を空間的に並べて構成されている。映像を扱うということで、大容量メモリが必要と思われるが、ライン間の処理であれば、数ライン分のメモリで十分である。映像信号処理をいろいろな角度から分類し、必要な演算要素、メモリ容量などを示したのが、表-1である。

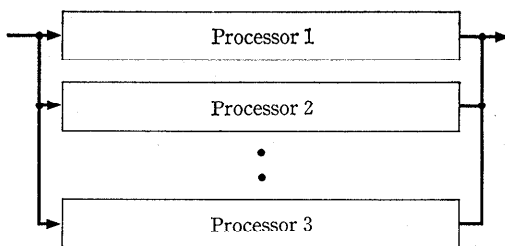
プログラム可能な汎用装置では、これらの処理を、ソフトウェアで行わなければならない。ところが、その情報量たるやフルスペックの映像信号の場合では、膨大なものである。表-2に、日本で使われている映像信号の規格を示すが、通常のTV信号の場合でも毎秒約14~40MByteものデータを処理しなければならない。ハイビジョン信号にいたっては、毎秒約150~220MByteものデータを処理しなくてはならない。各データに100ステップの処理を施すとしても、数十GOPS(Giga Operations Per Second)と汎用コンピュータをはるかにしのぐ高い処理速度が要求される。これを実現するには、デバイス技術だけでなく、並列処理技術^{2),3)}が必須となる。並列処理技術を使ったシステムは、他の分野でも多数作られている⁴⁾が、以下では、映像信号処理のための並列処理装置を構成するために考慮しなければならないファクタについて述べる。

2.2 並列処理方式

並列処理の基本方式は、パイプライン方式とパラレル方式である。パイプライン方式は、図-2(a)に示すように、多数のプロセッサをパイプラインのように縦続に並べ、この中に映像データを通して処理を重ねていくものである。必要な全処理ステップを処理の順番に各プロセッサに割り当て、各プロセッサで処理を行いながら順に前から後へパイプラインのようにデータを送って処理を重ねる。自動車工場のベルトコンベアによる生産ラインを考えると分かりやすい。これに対し、



(a) Pipeline



(b) Parallel

図-2 並列処理方式

パラレル方式は、画面を領域分割して、各領域ごとにプロセッサを割り当て、おのおのの担当領域で同じ処理を行うものである。大量の仕事をみんな手分けしてやれば早く終わると同じことである。

どちらの方式も、理想的には、 n 個のプロセッサを用いると、等価的に処理時間を $1/n$ にすることができるのであるが、なかなか理想状態にならないのが通例である。パイプライン方式では、全画面に同じ処理を施す場合はよいが、画面途中で処理を切り替えたり条件分岐があったりすると、パイプが常にいっぱいにならなくなり、効率が下がる。一方、パラレル方式では、分割した領域を乗り越えたデータが必要なとき、データをオーバラップして格納しておくなどの必要があり、無駄が生じる。しかしながら、映像信号処理では、データ非依存の処理が多いので、パイプライン処理も有効な場合が多い。また、全画面に均一の処理を行う場合も多いので、パラレル処理を行っても、汎用の並列処理装置に比べ処理効率は高い。

2.3 結合方式

プロセッサ間の接続法は、図-3に示すように、バス構造、クロスバー構造、ツリー構造、メッシュ構造などさまざまな方式がある。すべてのプロセッサ間を接続するほうが通信のバンド幅からは有利であるが、プロセッサ数 n に対して nC_2 のようにあまりに多くの接続線が必要となり製作上の問題がある。一方接続を限定すると、回路規模は小さくなるが、理論的に離れたプロセッサ間の通信に時間がかかるなど効率や自由度の低下を招く。すべてのケースに最適な方式はなく、用

表-1 映像信号処理の分類

分類	処理内容	処理例	
処理域による分類	時空間	時間領域の処理（走査が行われるので空間領域も含む）	濃度値変換, 映像混合, 色補正など
	周波数	周波数領域の処理（通常, 時空間領域の畳み込み演算にして行う）	YC 分離, 輪郭補償, 帯域制限, デフォーカス, ノイズリデューサ, FFT など
	座標	座標軸を変更する処理（曲面を使う場合は, 浮動小数点演算が必要）	拡大・縮小, 回転, 透視変換, 曲面へのマッピングなどの幾何学変換
	準時間	書き込みと読出しのクロックが異なる処理（標本化周波数以下のクロック制御が必要）	時間的ゆらぎを補正する TBC, 異種同期の映像を一致させる FS など
入力数による分類	単一入力	一つの入力信号のみを使用する処理（同一画像内のライン間, フレーム間処理も含まれる）	濃度値変換, 輪郭補償, YC 分離, 幾何学変換, ノイズリデューサ, 色補正など
	複数入力	二つ以上の入力信号を同時に使用する処理	映像合成, ステレオ画像処理, カラーマトリックス (R, G, B ↔ Y, C _R , C _B) など
次元による分類	0次元	それ自身のデータしか使用しない（基本的には, メモリが不要）	濃度値変換, 映像混合, コアリング, 色補正など
	1次元	水平方向のデータを使用する（数画素のメモリが必要）	BPF による YC 分離, 水平輪郭補償, コアリングによるノイズ除去など
	2次元	垂直方向のデータを使用する（数ラインのメモリが必要）	くし型フィルタ, 垂直輪郭補償, スムージング, TBC など
	3次元	時間軸方向（フィールド/フレーム間）のデータを使用する（フィールド/フレームメモリが必要）	幾何学変換, 3次元 YC 分離, 動き検出, TV 方式変換, ノイズリデューサなど
線形性による分類	線形	線形演算要素による処理（積和演算で表現できるもの）	利得調整, 輪郭補償, コンポリューション, フィルタなど
	非線形	非線形演算要素による処理（ROM または RAM によるルック・アップ・テーブルが使える場合がある）	r 補正, コアリング, 白クリップ, メディアンフィルタなど
データ依存度による分類	非依存	濃度値, 位置に関係なく全画面で均一の処理をする	濃度値変換, 輪郭補償, コンポリューション, 拡大・縮小, デフォーカスなど
	濃度値依存	濃度値に依存して処理を変える	適応型 YC 分離, 動き補償型 TV 方式変換器, 動き補償型ノイズリデューサ, 輪郭トラッキングなど
	位置依存	画面上の位置に依存して処理を変える	シェーディング補正, レジストレーション補正, 透視変換用の折返し除去フィルタなど

表-2 映像信号の規格

	NTSC 系 (通常の TV 放送)		Hi-Vision (ハイビジョン放送)
ライン数	525 本		1125 本
フィールド周波数	59.94 Hz		60 Hz
フレーム数	毎秒約 30 コマ		毎秒 30 コマ
標本化周波数	13.5 MHz	14.3 MHz	74.25 MHz
信号形式*	<ul style="list-style-type: none"> • R, G, B (4 : 4 : 4) • Y, C_R, C_B (4 : 4 : 4 または 4 : 2 : 2) 	<ul style="list-style-type: none"> • R, G, B (4 : 4 : 4) • Y, C_R, C_B (4 : 4 : 4 または 4 : 2 : 2) • コンポジット 	<ul style="list-style-type: none"> • R, G, B (4 : 4 : 4) • Y, P_R, P_B (4 : 4 : 4 または 4 : 2 : 2)
有効画素数**	720 × 483	768 × 483	1920 × 1035
量子化数	8~10 bit		

(注 1) 4 : 4 : 4 では, 全コンポーネントの標本化周波数が表記レートとなる。

4 : 2 : 2 では, 色差信号 C_R, C_B, R_R, P_B の標本化周波数が表記レートの半分となる。

(注 2) 4 : 2 : 2 では, 色差信号については, 横の画素数が半分になる。

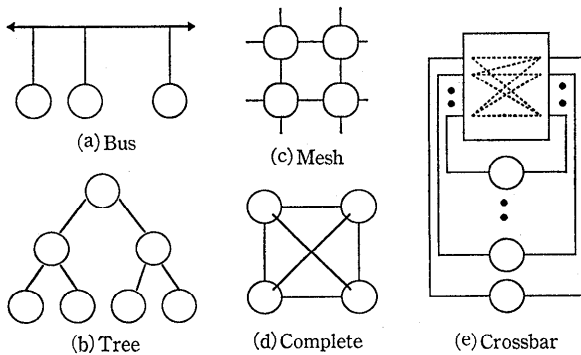


図-8 プロセッサ間接続

途、要求性能、回路規模、コストなどを考慮して構造を決めることになるのであるが、映像信号処理ではプロセッサ間の通信路で送受されるデータ列を映像信号列として、複雑なアービタ（仲裁機構）を設けない場合が多い。

2.4 制御方式

プロセッサの制御形態を示す分類法に、SIMD (Single-Instruction Multiple-Data) と MIMD (Multiple-Instruction Multiple-Data) という分け方がある。SIMD は、全プロセッサに同じ命令を与えて同じ処理を行わせるもので、単純ではあるが柔軟性に欠けると一般的に言われているが、映像信号処理では、均一の処理を全画面に施す場合が多いので、必ずしも不利ではない。一方、MIMD は、各プロセッサがそれぞれの命令で独立して処理を行うため、システムの自由度が大きい反面、プロセッサ間の同期の問題などがあり、制御は複雑になる。MIMD の変形として、SPMD (Single Program Multiple-Data) がある。これは、同じプログラムで対象とするデータ領域を変えて処理するものである。プログラミングの負担が小さくなるため、画像を領域分割して処理するタイプの並列処理によく使われる。

2.5 メモリ構成法

映像信号処理では、ライン間、フィールド間、フレーム間など大量のメモリを必要とする処理が多いため、メモリの配置の方法によって、処理能力は大きく変化する。メモリ配置の方法を大別すると、共有方式、分散方式になる。

メモリを1カ所に集中させ、これを複数のプロセッサで共有する共有方式では、複数のプロセッサが同じメモリに同時にアクセスすると、メモリ

アクセス競合が生じて処理能力が低下する。このため、メモリを分割し複数バス構成やクロスバー接続にして、同一メモリに同時にアクセスする確率を下げたり、バスの伝送容量を増やしたり、アービタを設けたりして、競合を緩和する工夫がされている。しかし、画像処理のようにメモリが大容量でかつ頻繁にアクセスする処理では、必要な接続線は膨大なものになる。

これに対して、各プロセッサにローカルメモリをもたせてデータを分散して格納する分散方式は、メモリアクセスによる競合は生じない。しかし、自分のメモリにないデータが必要になった場合、必要なデータをもつプロセッサと通信を行わなければならない。この通信が頻繁に行われると速度低下を招く。このため、同一内容のデータを複数のプロセッサで重複して格納する Replicated Memory とか Multi-read Memory と呼ばれる方式がある。この方式では、書込みは複数のプロセッサ間でデータに矛盾が生じないようにするために通信を行って書き換える必要があるが、読出しは独立して行える。フィルタ処理など近傍データを多数使用する映像信号処理では、読出し頻度のほうが書込み頻度に比べ多いので、この方式は有効である。

2.6 同期機構

リアルタイム処理を実現するためには、高速処理が要求されることは言うまでもないが、フレームやフィールドに同期した処理が求められる。すなわち、映像同期信号に同期して1フレーム当り1/30秒、1フィールド当り1/60秒のサイクルで処理結果を出力し続けなければならない。処理内容やデータに依存して処理時間が大きく変動すると、コマ落ちし不自然な動きが生ずる。処理速度は、最大瞬間風速であっては困る。最悪ケースでもビデオレートを維持できるように制御しなければならない。このため、基本的には同期信号を常にチェックし、これを基準にした制御が必要である。

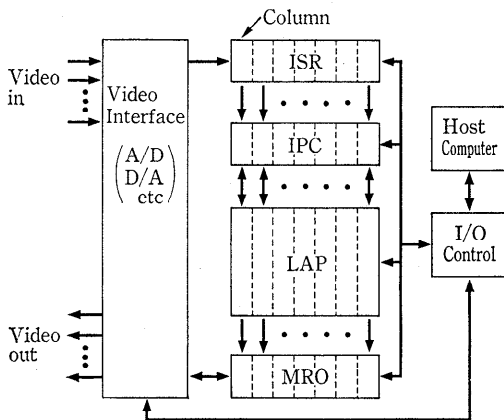
なお、通常の並列処理装置と同様、並列処理を効率よく円滑に行うために、プロセッサ間で処理の開始、終了の同期を取る必要があることは言うまでもない。

3. システム例

映像信号のリアルタイム処理を目指したシステムは、最近の研究開発が活発になってきており、各種 LSI やシステムが発表されている^{5)~17)}。以下では、筆者が直接見たことのあるシステムを取り上げ、簡単に説明する。

(1) Princeton Engine

現在、米国では HDTV の伝送方式の規格化作業を行っているが、そのはしりとなったのが米国デービッドサーノフ研究所が開発した ACTV (Advanced Compatible TV) である。この ACTV のシミュレーションに使用したのが Princeton Engine である。Princeton Engine⁵⁾では、図-4 に示すように、信号をシフトレジスタを使って縦割りにし、各コラムごとにプロセッサを割り当て、SIMD 制御でパラレル動作させ処理を行う。プロセッサ間は IPC バスで、分配、交換を高速に行うことができる。各プロセッサは、ALU、乗算器、32 フレームに相当する各コラムのデータをローカルに格納することができる。プロセッサ数は、64 から 2048 まで選択することができる。プログラムは、CAD を使って信号の流れ図を書くことにより作成できる。ライブラリ群もかなり整っており、マンマシンインタフェースのよいソフトウェア環境である。



ISR: Input Shift Register
 IPC: Inter-Processor Communication
 LAP: Linear Array of N-parallel Processor
 MRO: Multiport RAM Output

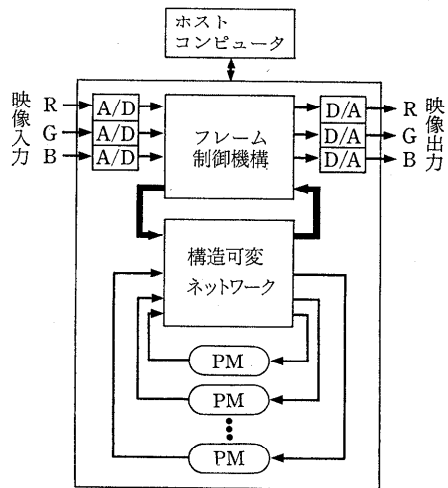
図-4 Princeton Engine の構成図
 文献 5) より

(2) 韋駄天

富士通研究所の韋駄天は、専用プロセッサ群の接続を構造可変ネットワークにより変更して、各種処理を行うシステムである。したがって、処理内容は、他のシステムと異なり、専用プロセッサ群の機能の組合せに限定される。韋駄天⁶⁾は、図-5 に示すように 48 個の処理モジュール PM とこれらを接続するネットワークで構成されている。ネットワークは、48×48 の多段スイッチングネットワークである。PM には、論理フィルタ、空間フィルタ、画像間演算器、濃度値変換器、ヒストグラム算出器のほか、ビデオレートで動作する画像メモリ、処理やネットワークによるさまざまな遅延を補正するディレイなどの専用の機能をもつユニットが用意されている。要求される処理に合わせて、必要なモジュールをネットワークを使って接続し、所望のパイプラインを構成する。PM は、1 画素当り 100 ns の速度で処理し、512×512 または 640×480 画素の画像をビデオレートで処理することができる。

(3) V S P

VSP (Video Signal Processor)⁷⁾ は、日本電気 C & C 研究所が、主に動画像の符号化アルゴリズムの研究用に開発したマルチプロセッサ型のシステムである。図-6 に示すように複数の同一仕様のプロセッサを入力ビデオバス、出力ビデオバス、フィードバックバスの 3 本のバスに接続した



PM: Processing Module

図-5 韋駄天の構成図
 文献 6) より

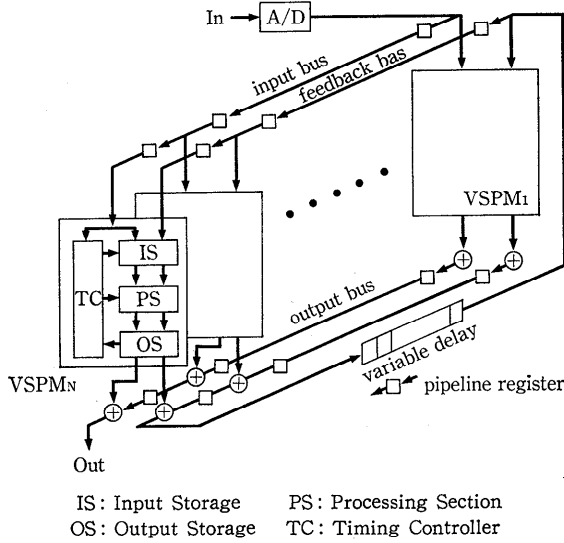


図-6 VSP の構成
文献 7) より

構成である。各プロセッサは、演算部のほかに画像メモリをもつ。1フレームの画像を複数個の小さなサブ領域に分割して、各プロセッサに割り当てることにより、処理負荷の分担を行う方式を取っている。その際出力サブ領域よりも、入力サブ領域を広くとるようにするオーバーラップセービングを使って、分割した周辺部でも処理が不連続にならないような工夫をしている。現在、36プロセッサ構成のシステム（標準TV用）と、HDTV画像を対象とした128プロセッサ構成のシステムが作られている⁹⁾。

(4) SIPS

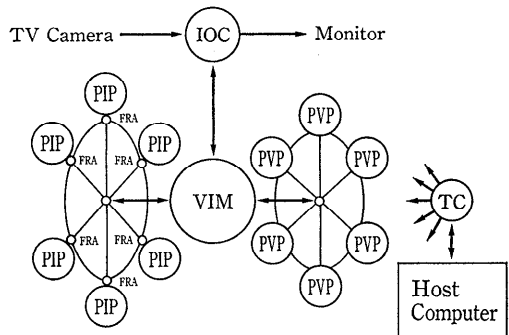
SONY 情報処理研究所の SIPS (Sight Information Processing System) は、カラー動画画像処理の研究用に開発したマルチプロセッサ型のシステムである。SIPS⁹⁾ は、図-7 に示すように画像演算部 PIP とアドレス演算部 PVP が分かれた構造となっており、おのおのが複数のプロセッサで構成されている。画像メモリ VIM は、最大 288 MByte (RGB の場合 8 秒に相当) ある。PVP でアドレスを発生させ、VIM からデータを読みだし、PIP で処理するといった機能分担を行っている。プログラムは、ホストコンピュータ上でクロスアセンブラやC言語を用いて開発する。(動画画像処理装置では、処理速度が優先されるために、処理効率を上げるために、多くの場合低レベルの言語が使われている。) プログラムの負担を軽減

するためにシミュレータなどのデバッグツールが用意されている。最近、小型で性能を向上させたシステム IMG-1000 が開発された¹⁰⁾。

4. 放送における実用例

4.1 放送における映像信号処理

放送局で番組を制作するところから、家庭のTV受像機で見るまでの過程で、さまざまな処理が行われている。特に、放送局での番組制作過程では、より良い画質で魅力的な映像を送るために、雑音除去、輪郭強調、色補正、画像合成、幾何学変換、特殊効果、補間、動き検出、圧縮などさまざまな処理が行われている¹⁸⁾。これらの処理は、現在のところビデオスイッチャや特殊効果装置など専用機器を使って行っている。それは、放送で求められる処理速度や精度が他分野に比べきわめて高いためであった。新しい映像効果や機能に対しては、要求ごとに新規に専用機器を開発してきた。しかし、このような方式では、短期間で対応することが困難であり、コスト的にも問題がある。このため、コンピュータのように同一のハードウェア上でソフトウェアを入れ替えることにより、映像の加工・合成などの放送局の広範なニーズに柔軟に対応できるシステムが求められている。



- PIP: Posmon Invariant Processor
- PVP: Posmon Variant Processor
- VIM: Video Image Memory
- FRA: Flexible Register Array
- IOC: I/O Controller
- TC: Total Controller

図-7 SIPS の構成図
文献 9) より

放送局で使用するための機器に要求される項目で一般の映像信号処理と異なる点をあげると、以下のとおりである。

(1) 多入力多出力の高画質カラー映像が扱える

放送用では、複数の映像素材を同時に使って合成処理しなければならない。場合によっては、同時に 10 種以上の映像素材を使って合成する場合もある。また、オンエアと呼ばれる通常の出力に加え、次にオンエアする映像を確認するための NEXT 出力、映像素材を確認するモニタ出力など複数の映像出力が必要である。しかも、入出力は 10 bit 程度、内部処理は 12 bit 以上の精度が要求される。

(2) 多彩な処理機能が要求される

時空間フィルタ、動ベクトル検出、輪郭抽出、幾何学変換、色処理、パターンマッチング、画像合成、テクスチャマッピングなどさまざまな処理が行われている。このうち、幾何学変換、テクスチャマッピングについては、映像表現のひとつとしてかなり重要視されている。しかしながら、その一方で、使用頻度が低かったり、飽きられて使

われなくなったりするものも多い経済効率のよくない処理でもある。最近では画像認識技術を援用して映像効果を作り出すなど放送における映像処理も幅を広げつつある。

(3) 可変で拡張可能な構造が望ましい

番組の種類によっては、使用される機器や規模も大きく変動するため、用途に合わせて機能、構造、規模が変えられるほうが望ましい。

4.2 映像コンピュータ Picot

放送現場に実際に導入されたリアルタイム映像信号処理装置として、映像コンピュータ Picot (Picture computer) を取り上げる。Picot^{(19), (20)} は、放送用途を主ターゲットにしたマルチプロセッサ型のシステムで、複数の NTSC 信号をリアルタイム処理できる。Picot は、図-8 に示すように多段に接続されたクラスタ群、A/D、D/A群、ホストコンピュータ (Laptop PC, J3100)、コントロールパネル (操作卓) から構成される。映像信号を初段のクラスタに入力し、クラスタ群で処理し、最終段のプロセッサの出力から処理結果を出力する。映像入力、映像出力とも最大 16 系統ま

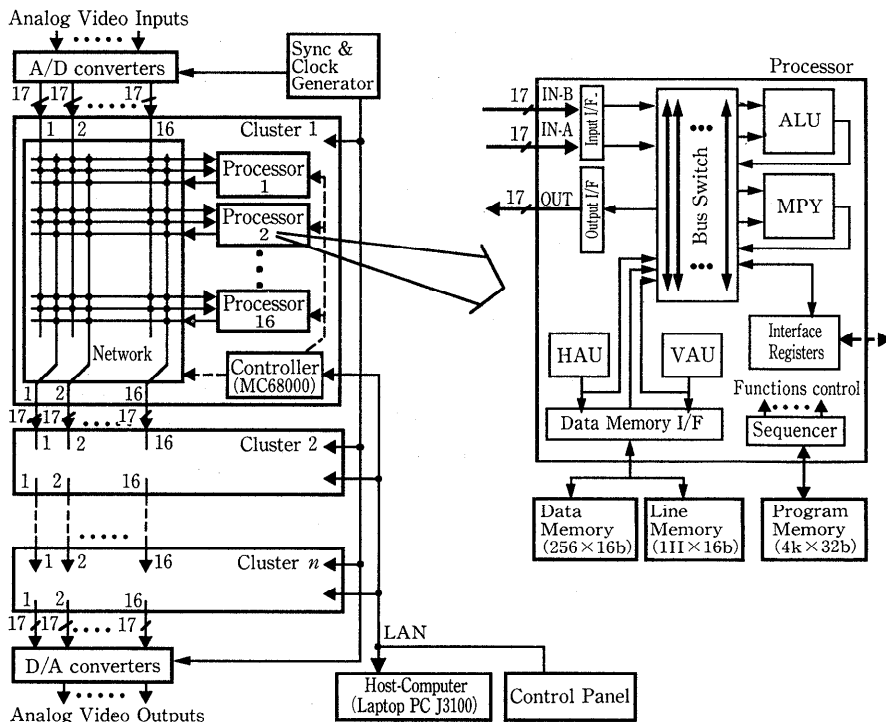


図-8 Picot の構成図

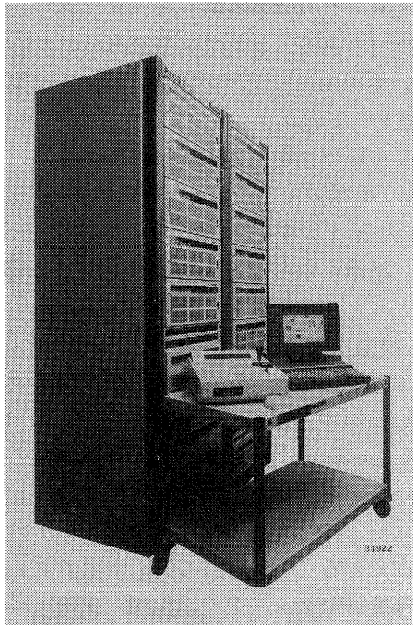


図-9 Picot の外観写真 (12クラスタ構成)

表-3 Picot の主な仕様

映像入力	NTSC 系信号 最大 16 系統 信号形式 [R, G, B], [Y, C _R , C _B], [Composite] 14.3 MHz 10 bit 標準化(注)
映像出力	NTSC 系信号 最大 16 系統 信号形式 [R, G, B], [Y, C, C _R], [Composite] 14.3 MHz 10 bit 標準化(注)
動作周波数	14.3 MHz
演算語長	16 bits
プロセッサ数	16× <i>n</i> 台 (<i>n</i> clusters), 192 台 (12 クラスタの場合)
演算速度	70 nsec/pixel, 33 msec/frame 0.7× <i>n</i> GOPS (<i>n</i> clusters), 8.4 GOPS (12 clusters)
制御	マイクロプログラム制御 (各プロセッサ 4k×32 bits)

(注) 入出力の仕様は A/D, D/A 変換器による。13.5 MHz や 8 bits の選択も可能。

で接続できる。図-9 にシステムの写真を、表-3 に主な仕様を示す。

クラスタ内には、同一構成のプロセッサが 16 台あり、プロセッサ間はネットワークにより任意の接続組合せが実現できる。また、クラスタ間も 16 本のバスで接続されており、クラスタをまたがっても並行処理やパイプライン処理を混在させて実行したり、処理の途中結果も出力できる。

各プロセッサは、二つの入力ポートから入力された映像信号を 14.3 MHz の速度で処理し出力する。プロセッサは、16 ビット精度で処理するが、入出力は同期情報を多重化した 17 ビットになっ

ている。各プロセッサには、プログラムメモリのほかに、NTSC 信号の 1 フィールド分の容量に相当するデータメモリがあり、処理に必要なとするデータは、おのおののプロセッサのデータメモリに確保しておく方式を採っている。必要なデータの配置は、処理内容ごとに前もってフレーム（またはフィールド）単位で分かっているので、処理に先立ち自分のメモリに格納しておく。複数のプロセッサに同じ内容のデータが重複して確保されることもあるが、メモリアクセス競合がなく、プロセッサ数にほぼ比例した性能向上が得られる利点がある。各プロセッサには、ALU, 乗算器 (MPY) に加え、専用のアドレス演算回路 (HAU, VAU) をもつため、メモリアクセスと処理を並行して行える。プロセッサは、絶対値、最大値、最小値演算機能、アフィン変換時の自動クリッピング機能、同期信号に同期したシステム制御、演算フラッグや同期などの 2 条件同時チェック機能など映像信号処理に適した処理機能と制御機構を有する。

Picot では、ホストコンピュータ上でプログラムを開発し、LAN を通じてクラスタ群に送り込むことにより、各種映像信号処理を行う。C 風の言語を用いてプログラムできるが、他のマルチプロセッサと同様タスク分割、同期制御などシステムに関する細かな知識が必要である。この知識をもたなくても、簡単にプログラムが作成できるツールとして、映像信号処理技術者に馴染みやすいシグナルフローグラフを主体にした PicPen (Picot Programming Environment) が試作されている²¹⁾。PicPen では、実現したい処理をホストコンピュータ上で図-10 に示すように、差分、

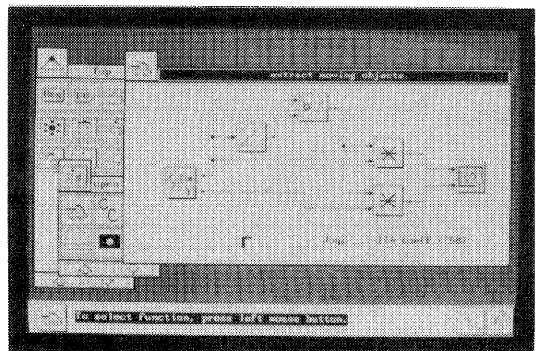


図-10 PicPen の操作画面

混合、フレームディレイなどの機能を表すアイコン化されたモジュールを使ってシグナルフローグラフを書くと、PicPEn が自動的に各種制約条件を考慮しながら、タスクをプロセッサに割り当て、プログラムを生成する。このため、プログラマはハードウェアに関する特別な知識は不要である。

Picot は、NHK スペシャル、ソウルオリンピック(NHK)¹⁹⁾大阪国際女子マラソン(関西テレビ)^{22), 23)}などの番組のほかに、2画面伝送装置²⁴⁾の開発などに使われている。Picot を使った処理例を図-11に示す。(a)は球体にテキストチャマッピングした効果、(b)は正弦波を用いてアドレスを変調した効果、(c)は乱数を用いて変調した効果、(d)は移動物体を抽出した例、(e)は顔を自動的に追跡して+印で表示した例、(f)はマラソンランナを追跡してランナのピッチ、スライドを算出し、グ

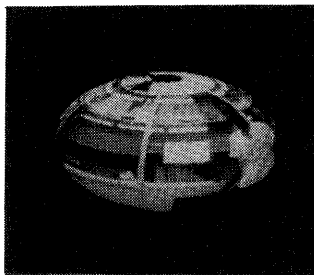
ラフィックワークステーションを使ってCG生成した例(関西テレビ「大阪国際女子マラソン 92」より)である。

現在、12クラスタ構成のシステムがNHKの番組制作現場で試験運用されている¹⁹⁾。Picot は、従来の放送機器と異なり、導入後もニーズに応じて機能が変わるのが大きな特徴で、これまでも演出サイドの要求に応じて制作現場で新機能ソフトウェアが開発されている。

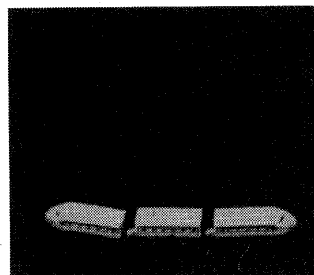
4.3 ハイビジョンへの対応～HD-Picot～

Picot は、NTSC 信号を対象にしたシステムであったが、ハイビジョン対応のシステムを現在開発中である²⁵⁾。

標本化周波数が 74.25 MHz のハイビジョン信号を、プロセッサの動作周波数が 14.3 MHz の Picot で処理しようとする、標本化周波数の比から5倍以上のプロセッサ数が必要となる。制作現場に導入した前述のシステムに相当する処理能力を得ようとする、1000台近いプロセッサが必要となり、現実的な数字ではない。そこで Picot の基本アーキテクチャを継承するが、高速高性能なプロセッサを開発した。現在の CMOS 技術レベルから考えると、プロセッサを 74.25 MHz で動作させることは困難で、30～40MHz 程度が妥当な動作周波数である。そこで、動作周波数を5倍に引き上げるのはあきらめ、プロセッサの動作周波数で2倍以上、機能で2倍以上、合計5倍の処理能力の達成を目指した。これにより、NTSC 版 Picot と同等のプロセッサ数でハイビジョンに対応できることとなる。具体的には、 P_B , P_R の標本化周波数である 37.125 MHz をプロセッサの動作周波数とした。このため、輝度信号 Y (74.25 MHz 標本化) を処理するためには、色差信号 P_B , P_R (37.125 MHz 標本化) の2倍のプロセッサを割り当て処理させる。また、プロセッサの機能に関



(a) テキスタマッピング



(d) 動物体抽出



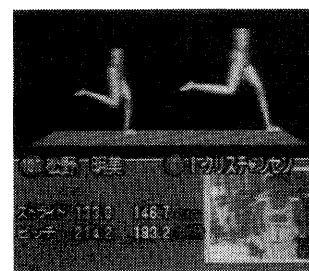
(b) ウェーブ効果



(e) 動き追跡



(c) すりガラス効果



(f) ストライドピッチ計測

図-11 Picot の処理例

しては、プロセッサ内に ALU 2 個, 乗算器 2 個, 算術アドレス演算器 2 個, 可変遅延 2 個と NTSC 版の 2 倍以上の演算器を搭載した。各演算器の演算機能も改善したので実質的にはもっと性能向上している。

5. むすび

リアルタイム映像信号処理装置を構成するうえで考慮すべき点について概説したうえで、各種システム例, 放送における実用例について述べた。現在のところ, 映像信号のプログラム処理はまだ少数波であるが, 着実にその方向に向かっている。プログラマブルシステムは, 映像信号の形式にこだわらないシステム構成が可能で, 世界中の TV 方式に対応可能な汎用システムとなるはずである。

最後に, 本稿執筆にあたりご助言をいただいた NHK 放送技術研究所先端制作技術研究部二宮佑一郎長, 榎並和雅主任研究員, 東京農工大学金子俊一助教授, 写真を提供くださった東芝佐々木信之課長, 関西テレビ放送並川巖主任に感謝いたします。

参考文献

- 1) NHK 編: 放送方式, pp. 59-200, 日本放送出版協会 (1983).
- 2) 富田: 並列計算機構成論, 昭晃堂 (1986).
- 3) 村岡: 並列処理, 昭晃堂 (1986).
- 4) 特集: 並列処理マシン, 情報処理, Vol. 28, No. 1, pp. 2-105 (Jan. 1987).
- 5) Chin, D., Passe, J., Bernard, F. et al.: The Princeton Engine: A Real-Time Video System Simulator, IEEE Trans. Consu. Ele., Vol. CF-34, No. 2, pp. 285-297 (1988).
- 6) 直井, 古明地, 太田, 他: 構造可変型ビデオレートカラー画像処理システム「韋駄天」, 信学論 D, Vol. J73 DII, No. 10, pp. 751-760 (1990).
- 7) Nishitani, T., Tamitani, I., Harasaki, H. et al.: Video Signal Processor Configuration by Multi-processor Approach, Proc. ICASSP '89, pp. 797-800 (1986).
- 8) Tamitani, I., Harasaki, H. and Nishitani, T.: A Real-Time HDTV Signal Processor: HD-VSP, IEEE Trans. circuits systems for video technology, Vol. CASVT-1, No. 1, pp. 35-41 (1991).
- 9) 長谷部, 加藤, 伊藤, 菊池: マルチプロセッサ型ビデオ画像処理システム SIPS, 情報処理学会研究会資料, CV 39-5 (1985).
- 10) 伊藤, 米谷, 勝尾, 長谷部: 画像処理システム SIPS IMG-1000, TV 技報, IPCV 92-14 (1992).
- 11) van Roermund, A. H. M., Srijder, P. J., Dijkstra, H. et al.: A General-Purpose Programmable

- Video Signal Processor, IEEE Tran. on Consu. Ele., Vol. CE-35, No. 3, pp. 249-258 (1989).
- 12) Childers, J., Reinecke, P., Miyaguchi, H. et al.: SVP: Serial Video Processor, CICC '90, 17-3 (1990).
- 13) Ohta, N., Fujii, S. and Ono, S.: Performance of a Multicomputer Type DSP System NOVI for Motion Picture Coding, Proc. ISCAS '88, pp. 2305-2308 (1988).
- 14) Goto, T., Ando, K., Inoue, T. et al.: A 250 MHz 16 b 1.13 M-Transistor 0.8 μ m BiCMOS Super-High Speed Video Signal Processor, ISSCC '91 FPM 15.4 (1991).
- 15) Minami, T., Yamauchi, H., Tashiro, Y. et al.: A 300 MOPS Video Signal Processor with Parallel Architecture, ISSCC '91, FPM 15.3 (1991).
- 16) Yamazaki, T., Komuro, S., Kumata, I. et al.: A 1-GOPS CMOS Programmable Video Signal Processor, ISSCC '89, THPM 12.4 (1989).
- 17) Joanblanc, C., Senn, P. and Colaitis, M. J.: A 54-MHz CMOS Programmable Video Signal Processor for HDTV Applications, IEEE J. Solid-state Circuits, Vol. SC-25, No. 3, pp. 730-734 (1990).
- 18) 八木: 放送における画像処理, TV 学誌, Vol. 46, No. 11, pp. 1439-1442 (1992).
- 19) 八木, 福井, 榎並, 矢島: 映像コンピュータ Picot, NHK 技研 R & D, No. 16, pp. 34-53 (1991).
- 20) Yagi, N., Fukui, K., Enami, K. et al.: A Programmable Real-Time Video Signal Processing System, SMPTE J., Vol. 100, No. 11, pp. 860-868 (1991).
- 21) Ott, M., 榎並, 羽島, 他: PicPEN—実時間映像信号処理装置 Picot のためのプログラミング環境, TV 誌, Vol. 44, No. 11, pp. 1570-1578 (1990).
- 22) 並川, 宮本, 佐々木: マラソンランナーのピッチ, ストライド計測システム, TV 学技報, BFO 91-15 (1991).
- 23) 並川, 松本, 宮本, 佐々木: マラソンランナーのアニメーション表示, 1992 TV 年次大, 1-13 (1992).
- 24) 伊藤, 榎並, 二宮, 他: NTSC 2 画面伝送装置, 信学技報, IE 92-69 (1992).
- 25) 八木, 福井, 榎並, 他: HDTV 対応実時間映像信号処理システム~HD-Picot システム~, 信学技報, IE 92-94 (1992).

(平成 5 年 7 月 2 日受付)



八木 伸行 (正会員)

1956 年生。1978 年京都大学工学部電気工学科卒業。1980 年京都大学大学院工学研究科電気工学専攻修士課程修了。同年 NHK に入局。1982 年より, NHK 放送技術研究所に勤務。現在, 同所先端制作技術研究部に所属。画像処理システム, 動画像処理, 映像合成の研究に従事。工学博士。1991 年 TV 学会論文賞受賞。著書「C 言語で学ぶ実践画像処理」(共著, オーム社) など。電子情報通信学会, TV 学会各会員。