プログラマブルなディジタル信号処理用 LSI として、ディジタルシグナルプロセッサ (DSP) の応用が広がっています。これまでは、音声帯域の信号処理への適用 (フィルタバンク、符号化、合成、など) が中心でしたが、高性能デバイスの出現により、図形や画像処理への適用も期待されるようになっていきます。この小特集は、DSP 技術について、応用の現状、使い方の基本、画像処理への応用に向けての今後の発展の可能性に関する最新の報告をまとめたものです。

なお今回の小特集は、主として山本英雄・三橋哲雄両編集理事、古角尚之、橋本慶隆両編集幹事が企画を担当しました。
(編集委員会)

1. DSP の 技 術 動 向

西 谷 隆 夫[†]

1. ま え が き

ディジタル信号処理専用開発されたプログラマブルな高速 1 チップマイクロプロセッサのことを、ディジタルシグナルプロセッサ (以下 DSP: Digital Signal Processor と略す) もしくは簡単にシグナルプロセッサと呼ぶ。1 チップ内に高速演算回路の他、プログラムやデータを蓄えるメモリーを持ち、音声信号程度であれば適当なプログラムを載せることで信号の変復調、フィルタリング、スペクトル解析など一連の信号処理を 1 チップだけで実行できる。

DSP では算術演算、特に乗算結果を累算する積和演算を高速に実行できるように工夫した独特のアーキテクチャを採用しており、最近の DSP の例では、32 ビット浮動小数点演算回路を内蔵し、演算能力も 1 チップでありながら VAX-11/780 の 13 台分以上という高速性を誇る DSP も登場している。このため、応用分野も当初の音声信号処理に限らず、コンパクトディスク用のオーディオ信号処理や、静止画処理などにも適用されるようになって来ている。

DSP はディジタル信号処理ハードウェアをコンパクトに、かつ低価格で実現する重要な手段となって来ており、以下においてまず DSP の定義・意味を明確にし、ついで開発の経緯を振り返り、最新の DSP アーキテクチャについて述べる。さらに、現在市販され利用可能な DSP の性能比較を行う。

2. DSP と は

DSP を定義付ければ“LSI 化により実現された高速算術演算主体のプログラマブルな単一チップマイクロプロセッサ”となる。つまり DSP は内部に高速算術演算回路、データメモリー、プログラムメモリーおよび制御回路を包含した単一チップマイクロプロセッサで、一連の信号処理をこのチップだけで実現できる。

DSP はアナログ技術による既存信号処理にディジタル信号処理を導入し高機能化、高精度化を図るためのハードウェアを提供しようとして誕生した。このため、DSP に要求されたことはアナログ回路と同程度のサイズで実現でき、かつ、消費電力も十分に小さいことで、1 チップですべてを実現する 1 チップコンセプトを開発当初から採用していた。

DSP と汎用マイクロプロセッサを区別する決定的要因は、高速演算中心のアーキテクチャを採っていることであり、詳細は後述するものの、高速乗算器を内蔵させたこと、この乗算器を効率良く動作させるメモリー/バス構成を採用したこと、および、プロセッサリソースを並列にコントロールするマイクロプログラム方式を採用したことに特徴づけられる。このような特徴を有するために、たとえ命令サイクルが同じであっても、DSP をディジタル信号処理に用いた場合は汎用マイクロプロセッサと比べ 1 桁ないし 2 桁高速となる。これは、1 チップマイクロプロセッサの多くが ALU やレジスタファイルなどを中心とした既存マイクロプロセッサ (例えば 8088/8080) にメモリーおよび入出力ポートを付け加えて 1 チップ化しただけのもので、処理速度向上よりは既存マイクロプロセッサシス

[†] 日本電気株式会社 C & C システム研究所

“Recent Trends on Digital Signal Processor Development” by Taskao Nishitani (C & C Systems Research Labs., NEC Corporation, Tokyo)

テムを1チップ化する方向で設計されたのとは比べ大変な違いである。

3. 開発の経緯

ディジタル信号処理¹⁾では、アナログ信号を一度ディジタル化し、アナログ回路が実現していた微積分演算に基づく信号処理を差分方程式に基づく演算に置換してディジタル領域で実行する。このため、適当なディジタル回路さえ用意すれば、高精度システムや高安定度システムが容易に実現できる。

DSPの登場は、その背景として音声・通信分野における高速モデム、音声認識、音声合成、高能率音声符号化、エコーキャンセラなど、ディジタル信号処理技術の導入なしでは実現が困難なシステムに対する強い要請があった。このような分野では従来の通信端末もしくは通信装置と同程度のハードウェアでなければ受入れられない。写真1はDSP登場以前に開発したADPCM 高能率符号化装置とDSPを用いた同一機能のシステムを比較したものであるが、いかに能率が上がろうともDSP登場以前のシステムを通信端末として利用する気にはなれまい。つまり、性能だけではなく装置のサイズ、消費電力、価格などでもアナログ技術によるシステムに打ち勝つ必要があり、これにはLSI化が必要不可欠であった。

しかしながら、実時間でディジタル信号処理を行うには、標本化された1信号値の処理を1標本化時間間隔以内に終わらねばならず、電話音声信号では125 μ s、またNTSCテレビ信号では125 nsがこの標本時間間隔である。この標本化時間に処理すべきことも多く、前に述べた音声信号処理に必要なとされる乗算速度だけを例にとっても1秒間に100万回から1000万回となり、容易に達成できるものではなかった²⁾。

写真1 ビットスライスマイクロプロセッサとシグナルプロセッサによるADPCM符号化器

ディジタル信号処理ハードウェアをソフトウェア制御によるプロセッサ形式で実現しようとする試みは早くからあった。これは高速な算術演算回路を効率的に使用して小型化を図れること、またソフトウェア制御の柔軟性によるシステムの機能修正・機能追加が容易であることなどの利点を活かそうとしたものである。しかしながら1970年代当初に登場した汎用マイクロプロセッサ（例えば4004や8008/8080）などでは演算語長および演算速度がまだまだ充分でなかった。

DSPに影響を与えたLSIは1970年代半ばに登場した、マイクロプログラム制御によるビットスライスマイクロプロセッサと16ビット単一乗算器LSI³⁾である。当時のミニコンピュータ用に開発されたビットスライスマイクロプロセッサは、これをワード方向に並べるだけで16ビットプロセッサとして構成でき、命令サイクルタイムを125 nsから250 nsにできた。この速度は前述の単一チップ乗算器との整合性も良かった。写真1で示したDSP以前の大きなシステムもこの組合せで実現したものである。

しかし、このようなシステムも研究用であったり、最先端システムとして開発されたもので、小型、低消費電力、低価格を満足させるものではなかった。

1980年代はDSPの年代といっても過言ではない。すでに1979年にはIntel 2920、AMI S 2811という初歩的なDSPが発表されていたが、演算能力の点から広範な応用分野を開発するには至らなかった。1チップコンセプトに基づいたDSPの方向を確立したのは1980年に登場したNEC 7720⁴⁾とBell研究所のDSP-20⁵⁾からであろう。これらDSPは前述した応用分野に適用され、大きな成功を収めた。プログラマブルな1チップコンセプトに基づくLSIを用いる利点は次のようにまとめられる。

- (1) プログラマブルだから種々のシステムに応用できLSIの量産効果により低価格化を図れる。

図1 シグナルプロセッサの進歩

- (2) プログラマブルであるためシステムの機能修正・機能追加が容易である。
- (3) 単一チップであるため複数 DSP を用いても装置の小型化が可能である。
- (4) 単一チップであるため複数 DSP を用いても装置の消費電力を低減できる。
- (5) 単一チップ DSP を用いるため信頼性が向上する。

以上の利点が明確になるに及んで LSI 製造業各社が DSP 分野に参入し始め、各社とも既存 DSP より少しでも性能の良いものを開発しようと努力したため、飛躍的に DSP の性能が向上して来ている。図 1 は DSP の発表年度と集積度の関係を示したものであるが、4 年で集積度が 3 倍となっている。この速度はメモリの集積度向上速度には達しないが、目をみはるべきものである。

4. DSP のアーキテクチャ

2 節でも触れたように、DSP を特徴付ける高速演算用のアーキテクチャは

- (1) マイクロプログラム制御方式の採用
- (2) 高速乗算器の採用
- (3) 高速乗算器の使用効率を高められるメモリ/バスアーキテクチャの採用
- (4) メモリ/乗算器/累算器からなるパイプライン化積和演算回路構成の実現

にある。

水平型マイクロプログラム方式を採ると、マイクロ命令サイクルはすべての動作の基本時間となり、どのマイクロ命令も同じ処理時間になるよう設計される。ほとんどの DSP ではこの基本時間である 1 命令サイクルで 1 演算を実行している。また、1 命令サイクルで、あるデータがバスを占有した場合、このデータ転送自体が基本動作であるから、他のデータはこのバスを介した転送には利用できない。このバスの競合を避けるための手段が独特なメモリ/バス構成の導入につながっている。さらに、マイクロプログラム方式では複数のプロセッサリソースが活性化されるため、高速信号処理が実現できる。しかし、この半面、プログラム開発を行う人は往々にして複数個の活性化リソースの活性化順序に混乱を起こすことも多い。

水平型マイクロプログラム方式を用いている DSP では、プログラム開発者は DSP 内部のクロック信号位相を考慮しなくとも以下のルールに従って考えれば充分である。

- (1) バス/サブバスにレジスタ/メモリからデー

図 2 32 ビット浮動小数点シグナルプロセッサ (NEC 77230)

タを載せられるのは各命令サイクルの最初である。

- (2) バス/サブバス上のデータは 1 命令サイクルの間不変である。
- (3) レジスタ/メモリへのデータを格納するのは各命令サイクルの終わりである。

このような DSP アーキテクチャを第 2 世代 DSP である NEC 77230[®] を例にとって説明しよう。図 2 に示すのは NEC 77230 のブロック図であり、大別して浮動小数点演算部、データメモリー部、プログラム制御部と入出力部からなる。

マイクロプログラム方式をとったことに起因するアーキテクチャ上の第一の特徴は、プログラム部とデータメモリー部が完全に分離されていることである。プログラムは 2048 ワードのプログラムメモリーに格納され、プログラムカウンタ (PC) により命令が読出される。データ処理部はこの部分と完全に独立しており、常に命令フェッチとデータ処理を並列に実行できる。汎用のマイクロプロセッサではプログラムとデータは同一メモリー空間内の異なった場所に格納され、プログラムの実行順序はまずプログラム部から命令を読出し、続いてオペランドであるデータを読出してから演算を実行する。

最近のマイクロプロセッサでは命令の先取り制御を行っているものもあるが、このような場合でも最悪時は従来の手順に従う。マイクロプログラム方式の他の利点としてプロセッサリソースの並列制御が挙げられ

るが、これは後述する。単にプログラム部とデータ部を分離しただけの方式は、ハーバードアーキテクチャと呼ばれている。

プログラム部にはこの他、サブルーチンジャンプ時の戻り番地を記憶するためのスタック (STACK) およびプログラムループを効率良く実現するためのループカウンタ (LP) がプログラムカウンタに接続されており、また、プログラムカウンタの内容はアドレスポート (AP) を介して外部メモリーをアクセスできる。つまり、外部メモリーにプログラムを格納しておけば、データポート (DP) を介して命令をデコードに転送できるため、外部メモリーに格納されたプログラムでも動作できる。

浮動小数点演算部は浮動小数点乗算器 (FMPY)、浮動小数点 ALU (FALU) およびワーキングレジスタファイル (WR) を中心に構成されており、乗算結果の累算、つまり、積和演算を効率的に計算できる。この DSP では浮動小数点乗算器に入力されるデータの形式は指数部 8 ビット仮数部 24 ビットの合計 32 ビットの浮動小数点表示である。乗算は指数部の加算と仮数部の積となるが、仮数部の 24 ビットデータ (符号 1 ビットと絶対値 23 ビット相当の 2 の補数表示) 間の積は 47 ビットとなるため、乗算器出力は指数部 8 ビット仮数部 47 ビットの合計 55 ビットを、そのまま浮動小数点 ALU に転送する。このため、浮動小数点 ALU およびワーキングレジスタの語長も 55 ビット分用意して、高い演算精度を維持できるようにしている。ブロック図上に二重線で示した部分はすべて 55 ビットバスである。

図に示したレジスタ “PIPE” は、プログラム開発者に内部クロック信号を意識させないために設けた仮想的なレジスタである。ワーキングレジスタファイル内の同一レジスタが連続して使用される場合のみ “PIPE” レジスタから FALU へのバスが開かれる。後述するが、この機能により連続的な積和演算が効率良く実行できる。

データメモリー部は 512 ワード RAM を 2 面と 1024 ワードの ROM からなり、浮動小数点演算部にデータを効率良く供給できるように工夫している。まず、乗算には乗数と被乗数という 2 個のオペランドが必要であるが、3 組のメモリー内の適当な 2 組のメモリーから 2 個のオペランドデータを同時に浮動小数点乗算器へデータ転送できる。この場合、一方のデータをメインバス経由で、他のデータを RAM から直接乗算器へ結ばれたサブバスを介して転送する。このように DSP においてはメインバスの他にプロセッサリ

ソース間のデータ転送用バスが複数あるのが通常で、これらのバスを総称してマルチバスアーキテクチャと呼ぶことも多い。

浮動小数点演算部に絶え間なく新しいデータを供給するには、メモリーのアドレスを命令サイクルごとに更新できる機能が必要となる。この機能は RAM 0 と RAM 1 の各々に独立に設けられた 2 組のポインタ (インデックスポインタ (IX) とベースポインタ (BP)) および、ROM のアドレス部に設けられた ROM ポインタ (RP) が担当する。これらのすべてのポインタはメインバスを介してデータの供給を受けることも可能であるが、各々のポインタはアップダウンカウンタ機能を持っており、現在のポインタの内容を直接マイクロ命令の指令により +1 または -1 できる。また、インデックスポインタには (IX+BP) の演算結果をロードする機能が、さらに、ROM ポインタには +2¹ 歩進する機能が設けられている。RAM 0 および RAM 1 のアクセス法は、それぞれのポインタにより IX, BP, (IX+BP) のいずれかでアクセスされる。

このようなデータメモリー部の構成は、アダプティブフィルタや高速フーリエ変換など種々の応用を調査し、汎用性とハードウェアの複雑さのトレードオフから定めたものである。例えば、複素演算を行う場合にはシントメトリックなアドレスポインタを持つ 2 組のメモリーの各々を実数部データ用と虚数部データ用に割り振ると、読出しメモリーの選択が実数/虚数の選択となり、効率的である。また、ROM は 1 組しか用意していないが、ポインタの歩進量を +1, -1, +2¹ にセットし、偶数番地と奇数番地に各々実数部と虚数部を格納しておけば、+2 歩進は複素数としてのアドレッシングに、また +1/-1 歩進機能は実数/虚数の選択に利用できる。

入出力部は A/D 変換器や D/A 変換器、また、これらのアナログインタフェースの代替としての PCM コーデックと直接接続するためのシリアル入力ポート (SI) およびシリアル出力ポート (SO) が用意されている。この他、外部アドレスレジスタ (XAR) の内容をアドレスポート (AP) より出力し、外部メモリーとデータレジスタ (DR) の間でデータをデータポート (DP) を介して互いにやり取りする 32 ビットパラレル入出力を持っている。

いま、ROM のアドレス 0 番地から n 番地に格納されている係数ベクトルと、同じく RAM 0 のアドレス 0 番地から n 番地に格納されているデータベクトルの内積演算、つまり積和演算をワーキングレジスタファイル内のレジスタ A を用いて行っている場合を考

える。以下、積和演算の第 k 番目の項を計算する命令サイクルを考えよう。この場合、命令サイクルの始めには RP および IX 0 ともにアドレス k を示しており、次の6個の独立した演算を一度に実行できる。

- (1) RAM 0 のアドレス k に蓄えられているデータの乗算器入力レジスタ K へのサブバスを介した転送。
- (2) ROM のアドレス k に蓄えられているデータの乗算器入力レジスタ L へのメインバスを介した転送。
- (3) 乗算器入力レジスタ K, L に蓄えられている RAM 0 および ROM のアドレス $k-1$ から転送されたデータ間の乗算。
- (4) 上記乗算結果の“PIPE”レジスタに蓄えられている第 $k-2$ 項までの積和演算結果への累算と、“PIPE”レジスタ第 $k-2$ 項までの積和演算結果のワーキングレジスタファイル内のレジスタ A への反映。
- (5) 次のデータ格納アドレス生成のためのインデックスレジスタ IX 0 の歩進。
- (6) 次の係数格納アドレス生成のための ROM ポインタ RP の歩進。

よって、 k エレメントベクトル間の内積を、メモリー/浮動小数点乗算器/浮動小数点 ALU/ワーキングレジスタファイルからなるパイプライン処理により、連続的に処理できる。

図3はこのDSPを制御しているマイクロ命令形式である。マイクロ命令はALU関連命令、ブランチ命令、データロード命令に分けられる。マイクロ命令で複数のプロセッサリソースが活性化できる理由は、マイクロ命令が独立した複数のフィールドから成っているためで、各フィールドが直接対応するプロセッサリソースを制御する。上記の積和演算では6個の処理を同時に実行できたが、これを制御するのはALU命令である。まず、オペレーションフィールド(OP)で浮動小数点加算を指定し、制御フィールド(CON-TROL)でIX 0 と RP の歩進およびRAM 0, ROM

図3 マイクロ命令形式

のアドレッシングモードを指定する。ALU入力指定フィールド(ALUINPUT)では浮動小数点乗算器出力とワーキングレジスタファイル内のレジスタAを指定する。メインバス転送フィールド(SRC/DST)ではROM出力をレジスタLに転送すると同時にRAM出力をレジスタKへ転送する複合指令を指定する。制御フィールドはこの他、ループカウンタ制御やROMの直接アドレッシングなどを指定できる。

残りの2命令形式は条件フィールド(COND)で指定した条件が成立した時のみ飛び先アドレスフィールド(NEXT ADDRESS)内のアドレスへジャンプする条件ジャンプ命令と、リテラルフィールド(LITERAL)内のデータをレジスタなどに転送する命令である。

5. DSPの展開

以上、NEC 77230 を具体例として DSP の特徴を述べたが、ディジタル信号処理の潜在的な大きな需要を狙って、それぞれの特徴を生かした DSP の開発が盛んであり、そのすべてをここで比較することはできない。表1は浮動小数点演算が可能な DSP の比較一覧表である。Bell 研究所の DSP-32⁷⁾と NEC 77230 は通常の大型コンピュータなみの 32 ビット浮動小数点演算を取り入れながら高速化を狙っているが、NTT の DSSP や沖電気の MSM 6992 では浮動小数点語長を短くして命令サイクルタイムを高速化している。

1 チップコンセプトに基づいた音声・通信分野を対象としたものを中心に述べて来たが、1980年に登場した NEC 7720 および DSP-20 を原点にして DSP の進展方向を探ると図4に示す展開が進行中であることがわかる。つまり、原点に位置するのは固定小数点演算を中心とした第1世代 DSP であるが、1985年頃より顕著になって来たのが単精度 32 ビット浮動小数点

表1 浮動小数点シグナルプロセッサの比較

デバイス名	DSP-32	NEC 77230	MSM 6992	DSSP-1 /T 9511
開発機関	ベル研究所	日本電気	沖電気	日本電信電話
語長：仮数部	24 (bit)	24 (bit)	16 (bit)	12 (bit)
指数部	8 (bit)	8 (bit)	6 (bit)	6 (bit)
命令サイクル	250 ns	150 ns	100 ns	50 ns
メモリー類				
RAM	512 語×2 面	512 語×2 面	128 語×2 面	2 ポート 512 語
ROM	なし	1024 語	なし	なし
プログラム	512 語 (ROM)	2048 語 (ROM)	1024 語 (ROM)	4096 語 (ROM)
外付け	8×16 kb	32×8 kb	22×64 kb	18×4 kb
ピン数	40 (DIP)	68 (PGA)	132 (PGA)	132 (PGA)

図4 シグナルプロセッサの発展方向

演算を基礎とした第2世代DSPの方向と、静止画や動画を信号処理の対象とするより高速性なDSPへの挑戦への方向、さらに、小型化、低消費電力化を極限まで追及したシリコンコンパイラやユーザ主導型カスタムLSI (ASIC) への展開である。

高機能化DSPもまた1985年Bell研究所発表のDSP-32および1986年日電発表のNEC 77230により方向づけられ、双方ともプログラミングを容易に行えるように工夫を凝らしている。DSP-32では言語Cのサブセットを、また、NEC 77230では簡易デジタル信号処理言語DSPLをサポートしつつある。

最近の注目すべき動向はビデオ関連の信号処理用プロセッサの開発にある。静止画の分野では大量のデータを高速に処理するためにデータフローという新しい概念に基づいた制御方式を試みた日電のImPP⁸⁾、また、従来方式であるが命令サイクル20MHzを持つ東芝のプロセッサ用演算器T 9506⁹⁾などの試みがある。動画像処理への展開では高速なプロセス技術を確認することにより、DSPアプローチを実現しようという命令サイクル50MHz以上の松下のRISP¹⁰⁾など意欲的な試みもある。さらに、1プロセッサ当たりの構成をDSPなみとし、処理すべき領域をサブ画面として複数のDSPを動画処理に用いようとする試み¹¹⁾もあり、このようなアプローチはLSIの発展方向が速度よりも集積度の向上が著しい点を考慮すれば、将来、複数DSPをオンチップした単一チップ動画DSPも夢ではない。

小型化、低消費電力化を図るには適用分野の特性を100%活用した専用チップを作るのにこしたことはなく、CAD技術の発展と共にこのような環境が整って来よう。ただし、このような場合でもDSPアプローチが不要になることはなく、開発環境が整備され、実

時間シミュレーションが可能な第1世代/第2世代DSPを活用して開発期間を短縮し、これを専用LSIへ進めることが多くなろう。また、浮動小数点演算を基本にした第2世代DSPを用いた場合は、ダイナミックレンジや演算精度に起因する複雑なスケールングやリミットサイクルに悩まされることがなく、このため、いっそうの開発期間の短縮が可能である。この場合も、その後、消費電力削減、低価格化を狙って固定小数点演算のシステムに移行させることとなろうが、専用チップアプローチを採らなくとも、浮動小数点DSPの命令体系のサブセットで動作できる低消費電力固定小数点プロセッサがあれば有効であろう。このため、DSPのファミリー化も進むものと思われる。

DSPの開発が特に重要なのは、プログラマブルな1チッププロセッサでありながら音声帯域信号の実時間信号処理を可能にした点で、DSPのプログラム開発ツールは世界中の信号処理研究者に実時間シミュレーションの道を与え、デジタル信号処理の研究を加速したことであろう。この観点から見ると、プログラマブルなビデオ信号処理DSPの登場が最も期待されるところである。

(昭和61年11月5日受付)

〔参考文献〕

- 1) B. ゴールド, C. M. レイダー: 電子計算機による信号処理, 共立出版 (1980)
- 2) 西谷: デジタル信号処理 LSI プロセッサ, 情報処理学誌, 24, 7 (July, 1983)
- 3) G. W. Mcleaver, et al. "A Monolithic 16×16 Digital Multiplier", ISSCC 74 (Feb., 1974)
- 4) T. Nishitani, et al. "A Single Chip Digital Signal Processor for Telecommunication Applications", IEEE J., SC-1, 6 (Aug., 1981)
- 5) J. R. Boddie, et al.: "Digital Signal Processor: Architecture and Performance", BSTJ, 60, 7, Part 2 (Sept., 1981)
- 6) T. Nishitani: "Advanced Single-chip Signal Processor", ICASSP 86 (Apr., 1986)
- 7) W. P. Hays, et al.: "A 32-bit VLSI Digital Signal Processor", IEEE J., SC-20, 5 (Oct., 1985)
- 8) Temma, et al.: "Data Flow Processor Chip for Image Processing", IEEE Trans., ED-32, 9 (Sept., 1984)
- 9) A. Kanuma, et al.: "A 20 MHz 32 b Pipelined CMOS Image Processor", ISSCC '86 (Feb., 1986)
- 10) T. Mori, et al.: "A Micro-Programmable Realtime Image Processor", ISSCC '86 (Feb., 1986)
- 11) T. Nishitani, et al.: "Video Signal Processor Configuration by Multiprocessor Approach", ICASSP '86 (Apr., 1986)

にしに たかお
西谷 隆夫 昭和48年、大阪大学大学院修士課程終了。同年、日本電気(株)入社。C&Cシステム研究所に勤務し、現在、同所研究課長。信号の帯域圧縮およびデジタル信号処理LSIアーキテクチャの研究に従事。