

メディアプロセッサ (MAP) のビジョンとアーキテクチャ

1C-1

小島啓二 西岡清和 野尻徹

日立製作所システム開発研究所

宮崎健司

日立製作所デバイス開発センタ

1. はじめに

実時間メディア処理応用においては数十 GOPS を超える処理性能が要求されるため、現在の性能を大幅に越えるプロセッサの登場が待望されている[1]。デジタル TV、高品質ビデオ会議、マシンビジョン、ラスト画像処理といった応用はその好例である。このニーズに対応するためアーキテクチャ面から種々のアプローチが試みられてきた。従来から使われてきた機能固定の ASIC を用いる手法では、標準規格の多様化、新規規格の登場、ユーザ要求の複雑化等の流れに対応しきれなくなっている。もうひとつのアプローチである複数の DSP や RISC プロセッサをマルチプロセッサとして使って要求される性能を実現する方式は、高いコストとプログラミングの難しさがコンシューマ製品への適用を難しくしている。

最近新たな解として実時間メディア処理に必要な性能を 1 チップで実現するプログラマブルプロセッサ、いわゆるメディアプロセッサが提案された。メディアプロセッサは対象とする応用に対して必要な性能を、ある程度専用化された最小限のハードウェアで実現することを狙って開発されている。メディア処理応用開発に携わる技術者に、これで独自のアルゴリズムを使った特徴的な製品をいち早く世に出せるとの多くの期待を抱かせたメディアプロセッサであったが、そのプログラミング環境は大きな問題を抱えていた[2]。すなわちこれらのプロセッサのプログラミングにはその性能を発揮させるためにアセンブラ言語の使用が強いられている。アセンブラ言語によるソフトウェア開発は C 言語等の高級言語に比べて約 10 分の 1 といわれる生産性の低さとともに、変更の困難という問題を持つ。わずか 1 命令の変更も注意深く並列性を発揮するように設計されたプログラムの動作を大きく変えかねず、開発者自身以外による変更はきわめて難しい場合が多い。結果として、メディアプロセッサの存在意義であったはずの柔軟性を大きく損なう結果となり、ユーザに ASIC 方式に対する明確な優位性を納得させるのは困難である。

メディアプロセッサのプログラミング効率の問題を解決すべく、いくつかのメディアプロセッサ開発ベンダは最適化された C コンパイラの開発を開始した。このような C コンパイラの開発には多大の時間と、リソースとスキルが必要となる。同時にメディアプロセッサのアーキテクチャもコンパイラによって十分な性能が発揮できるように設計されている必要がある。残念なことにこの最適化コンパイラ開発力とアーキテクチャ開発力の両立は多くの半導体設計会社にとって容易でなく、開発結果は依然ユーザを満足させるに至っていないといえる。本稿ではこれらの問題に対する解として、米国 Equator 社と共同開発した日立メディアプロセッサ (HDL2 MAP シリーズ、以下 MAP と略) について報告する。

2. ソフトウェア主導のメディアプロセッサアーキテクチャ設計

A Vision and Architecture of Media-processor (MAP)

Keiji Kojima, Kiyokazu Nishioka, Toru Nojiri

Systems Development Laboratory, Hitachi, Ltd. 292 Yoshida-cho, Totsuka-ku, Yokohama, 244 Japan

Takeshi Miyazaki

Device Development Center, Hitachi, Ltd. 6-16-3 Shinmachi Ome-shi, Tokyo 198-8512, Japan

MAPのアーキテクチャはVLIWによる命令レベル並列性と最適化Cコンパイラをベースに色々なメディア処理アプリケーションのアルゴリズムの分析、ならびにシステムレベルの動作解析結果によって定義された。具体的にはコンパイラとアーキテクチャシミュレータとを並行して開発し、この両者を4年近くをかけて各種メディア処理応用に対して繰り返し最適化した。ここでのゴールはいうまでもなく、対象とした応用群に対しC言語からのコンパイルだけで十分な性能を引き出し得る単一のアーキテクチャを設計することである。応用としては、MPEG-2 MP@ML ビデオデコード/エンコード、MPEG-4、HDTV ATSC AFD (All Format Decode)、各種ビデオ後処理、DolbyAC-3 オーディオデコード、JPEG、V.34 モデム、3DCG 等である。これらの応用が要求する計算能力ならびにデータ転送能力を満たすアーキテクチャ設計を通じて、我々はMAPのアーキテクチャが類似の性格を有する幅広いメディア処理に広く適用できるとの自信を得ることができた。MAPのアーキテクチャを定義するプロセスは、以下の基本方針のもとに行われた。

(1) ソフトウェア生産性

既に繰り返し述べているようにMAP設計の最大の目的はC言語とコンパイラのみでの実時間メディア処理応用ソフトウェア開発を可能とすることである。この要件を満足するために、命令セットアーキテクチャはコンパイラが高い性能を引き出し易いものであることが要求される。またプロセッサがソフトウェアのデバッグを容易化するための諸機能を的確にサポートすることも重要である。

(2) スケーラビリティ

今後予測されるさらに高い処理性能を要求する応用の登場に遅滞なく対応するために、アーキテクチャは容易に性能向上がはかれるようスケーラブルなものであることが重要である。言いかえるとさらに高い並列性を得ようとすると急に設計が複雑化するようなアーキテクチャではソフトウェア資産をその上に築いていく価値は半減する。

(3) 拡張性

MAPはVLIWのCPUのような計算要素のみでなく、IOや特殊なアクセラレータなどの周辺処理要素を統合したいわゆるシステムLSIであり、付加LSIなしにシステムを構成できるよう設計されている。システム応用面からの要求でこれらの周辺処理要素を部品化して、追加や削除を容易にすることも重要な課題となった。

(4) 実時間マルチタスクサポート

メディアプロセッサはビデオ、オーディオ等複数の実時間応用をマルチタスクで走らせるため、リアルタイムOS(RTOS)のサポートは必須条件であり、MAPのアーキテクチャ定義はRTOSの検討とも並行に進められた。プロセッサ目標処理性能も特に次に示すような複数のアプリケーションを同時に走らせるに足るものとするのを念頭に設定した。

- ・ 2本以上のビデオ、オーディオストリームの同時デコード
- ・ ビデオ会議応用に向けたビデオ圧縮と伸張の同時処理
- ・ ビデオデコードと画質向上向け後処理の同時処理

3. MAPアーキテクチャ

図1にMAPシリーズの最初の製品であるMAP1000のチップ内部構造を示す。

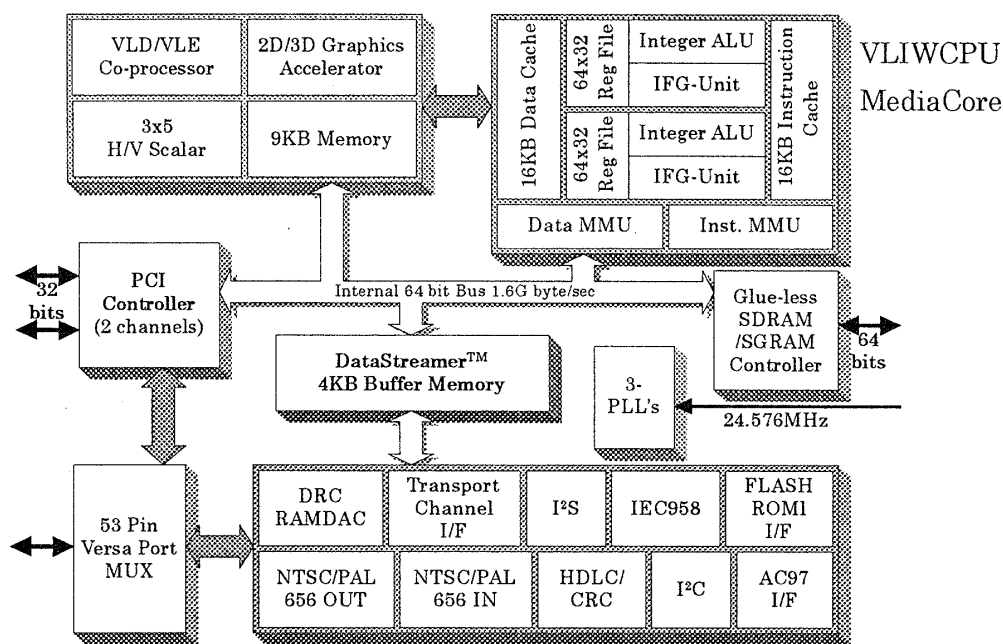


図1. MAP1000のチップレベルアーキテクチャ

3.1 VLIW CPU メディアコア

MAPの基本的なメディア処理のための並列計算能力はこのCPUコアが供給している。多くのメディア処理応用プログラムは並列処理可能な多次元行列処理のみからは構成されておらず、制御コード部分にみられるような並列性の少ない、俗にいわれるスパゲッティコードもトータル性能に大きな影響を持つことが多い。そのため、MAPのVLIW CPUコアはこのようなランダムなコードに対しても十分な効率が得られるよう設計されている。

MAP1000のCPUコア内には二つの演算用クラスタがあり、それぞれのクラスタはさらにInteger ALUとIFG-ALUの2種類のALUと64個の32ビットレジスタから成るレジスタファイル（32個の64ビットレジスタとしても使用可）を有する。二つのクラスタは完全に同一であり、将来のバージョンでクラスタを追加していけばより高い処理性能がスケラブルに得られるよう設計されている。Integer ALUは32ビットの整数演算と32/64ビットのロードストア、分岐制御等を行う演算ユニットである。一方IFG-ALUはさらに、32ビット整数演算サブユニット、グラフィック演算サブユニット、メディア演算サブユニット、浮動小数点演算サブユニット、除算サブユニットの5つの演算サブユニットから構成される。

グラフィック演算サブユニットは64ビット幅の分割加算器とデータシャッフル演算器から構成される。分割加算器は8個の8ビットデータや4個の16ビットデータといった複数のデータを同時に演算するものである。一方データシャッフル演算器は64ビットソースレジスタ中の8ビットデータ列を任意に並べ替えてデスティネーションレジスタにおけるといったシャッフルをはじめとしてメディア処理で必要となる様々なデータフォーマット変換を行う。

メディア演算サブユニットは128ビット幅の分割積和演算器と、演算結果の総和を計算するための加算木とを有する。差の絶対値の総和をとる計算や、内積計算等、ビデオの圧縮伸張において重要な演算の多くがこのメディア演算サブユニットでサポートされている。

浮動小数点演算サブユニットは2つの単精度 IEEE 互換浮動小数点積和演算器から成る。MAP1000 は二つのクラスタを有するので計4つの単精度の積和演算が同時に実行できる。

除算サブユニットは2つの 32 ビット整数または浮動小数点数の除算と平方根を計算する。浮動小数点演算サブユニットと除算サブユニットは MAP の高性能 3D グラフィックス用幾何、光源計算への適用を可能としている。

各クラスタは通常のレジスタファイルに加え、プレディケートレジスタファイルと呼ばれる 16 個の 1 ビットレジスタを持つ。プレディケートレジスタは VLIW においてオーバーヘッドが大きくなりがちな分岐によるジャンプをできる限りなくすためのもので、VLIW の性能向上にとって重要な要素の一つである。

MAP1000 のデータキャッシュは 16KB 4-way セットアソシアティブのライトバック型ノンブロッキングキャッシュである。データキャッシュは2つのクラスタとともに、ビデオ/グラフィックス用コプロセッサ、データストリーマにデータキャッシュへのアクセスを提供している。これらの複数ユニットに同時にデータを供給できるよう、内部的にはアドレスでインタリーブされた4つのバンクから構成されている。

命令キャッシュは 16KB 2-way セットアソシアティブで、VLIW 命令を圧縮した形式でキャッシングしている。データキャッシュ、命令キャッシュともに TLB ベースのアドレス変換機能をサポートしており、汎用の OS を容易に移植可能である。

3.2 ビデオ/グラフィックス用コプロセッサ

MAP1000 は VLD/E、ビデオスケーラ、および 2D/3D グラフィックスアクセラレータの3つのコプロセッサを有する。これらのコプロセッサはコプロセッサメモリと呼ばれるオンチップのスクラッチパッドメモリ (SRAM) リソースをシェアして処理に使っている。

VLD/E コプロセッサは汎用の 16 ビット RISC コントローラであり、VLIWCPU コアと並行して種々のビット処理を行う目的で設計されている。中心的な用途は可変長コードの圧縮伸張処理であり、これにはランレングス処理や量子化、逆量子化といった処理が含まれる。このようなビット処理に固定のハードウェアエンジンでなくプログラマブルなプロセッサを用いることによって、新アルゴリズムや新規格に対する柔軟性を大幅に改善することができた。VLD/E コプロセッサの他の用途として、ビタビ復号、リードソロモン符号処理といった応用があげられる。

MAP1000 のビデオスケーラは 4:2:2 あるいは 4:2:0 フォーマットのビデオフレームを水平 5 タップ、垂直 3 タップのコンボリューション型のフィルタを用いて任意のサイズにスケールアップまたはダウンして 4:4:4 もしくは 4:2:2 の YCrCb ピクセルとして出力するためのエンジンである。

2D グラフィックスアクセラレータは 64 ビットのラスタオペレーションを実行するユニットである。一方 3D グラフィックスアクセラレータはタイリングと呼ばれる領域ベースの 3D レンダリングアルゴリズム [3]にもとづくハードウェアエンジンである。描画する 3 角形に対する幾何、光源計算、ならびにセットアップ計算は VLIW CPU コアにより行われて結果がデータキャッシュに順次格納される。並行して 3D グラフィックスアクセラレータはこの結果をデータキャッシュから読み込み、シェーディングやテクスチャリングの処理を行って最終結果である画像を再びデータキャッシュに格納する。

3.3 データストリーマ

MAP のアーキテクチャにとって最も重要な要素の一つは 64 ビットの内部バス 2 本で結合されたチップ

内のコンポーネント間でデータを効率良く移動させる機能である。例としてビデオの IO ポートからビデオデータを読み込んでそれを外部の DRAM に格納し、それを VLIWCPU コアがデータキャッシュ経由で読み出して動き予測、DCT 等の圧縮処理を行い、結果を VLD/E コプロセッサに送ってハフマン符号化して別の IO ポートから出力するというビデオ圧縮処理の一連の流れを想定する。この各コンポーネント間のデータ移動の流れがどこで途切れても、たとえばデータキャッシュがミスヒットしてコアの処理に待ちが生じたりすればただちに全体の性能低下となって見えてくる。

データストリーマはこの複数のデータ移動を指揮するデータ転送プロセッサであり、チップ内の任意のコンポーネント間で最大 64 チャンネルの並行データ転送を行うことができる。データストリーマの使用により、データキャッシュが実際にミスする前に必要なデータをバックグラウンドでデータキャッシュに転送しておくことが可能となる。MAP1000 のデータストリーマは内部に 4KB のバッファメモリを備えており、プログラマはこのバッファを必要に応じてアロケートして使用する。転送するデータストリームは各データ要素の長さや要素数、要素間の距離を指定することで得られる 2 次元配列データの形式で定義される。

3.4 外部バスと外部メモリインタフェース

MAP1000 は外部バスインタフェースとして 2 つの 32 ビット (33/66MHz) PCI バスを備えている。2 つの PCI バスを使用するシナリオとしては、1 つはホストプロセッサとの結合に、もう 1 つは色々な PCI 用 IO デバイスを接続するのに使用することが考えられる。また、メッシュ構造のマルチプロセッサを構成するのにも用いることができる。MAP1000 は 64 ビットの SDRAM (100~133MHz) を外部メモリとして使用している。

3.5 IO 制御

MAP の IO 制御ユニットは種々の外部機器への接続インタフェースを提供している。MAP1000 はデジタルビデオ入力のための TCI、CCIR656/601、デジタルオーディオ入出力のための AC97、IIS、IEC958、通信用の ISDN、T1、汎用の USB、IIC/DDC、および RGB ディスプレイ用のインタフェース等を有する。

4. MAP ファミリーとロードマップ

最初の MAP1000 は 0.3 ミクロンの CMOS (2.5V 動作) を用いて実装された (図 2)。MAP1000 を用いて AFD をはじめとする種々の応用を C 言語のみで開発された事に成功した。さらに MAP1000 アーキテクチャをデジタル TV 等のコンシューマ応用により最適化させて 30GOPS を超える性能向上を実現するとともに論理規模を半減させたローコスト版である MAP-CA を 0.2 ミクロン CMOS テクノロジ (1.8V 動作) を用いて開発中である。図 3 に MAP のロードマップを示す。

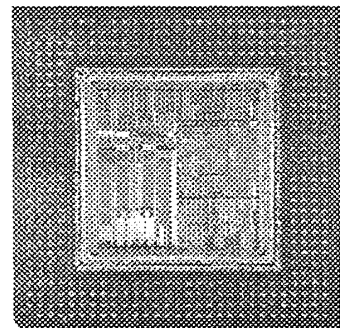


図 2 第 1 世代 MAP1000

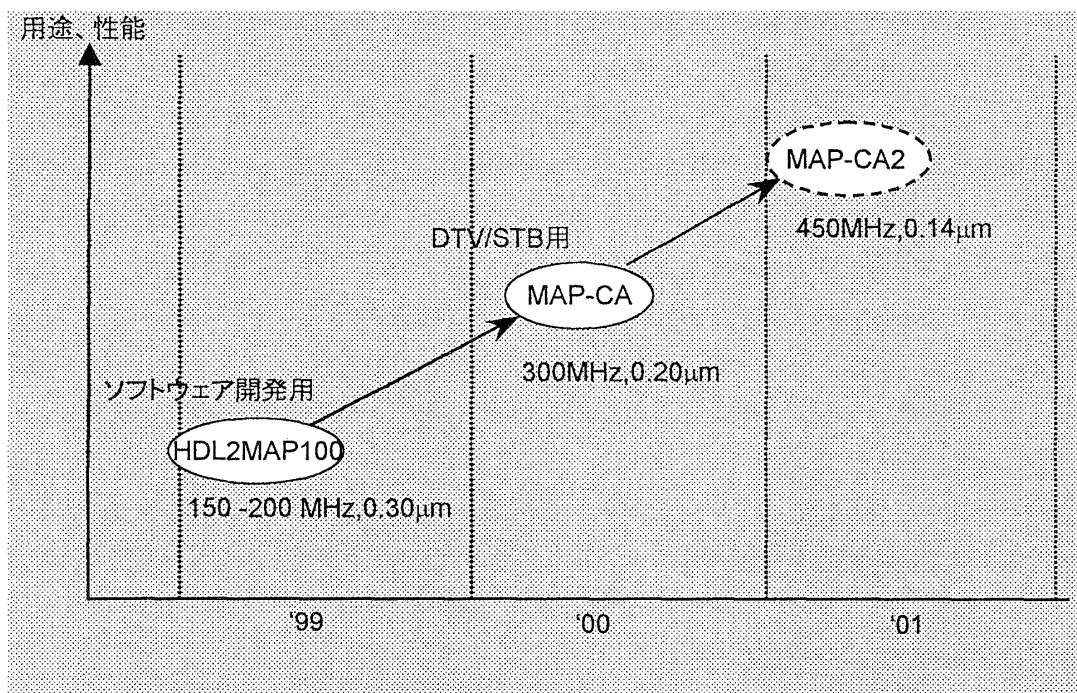


図3 MAPシリーズのロードマップ

5. 参考文献

- [1] B. Haskell et al., Digital Video: An introduction to MPEG-2, Chapman & Hall, New York, 1997.
- [2] J. Turley, Multimedia chips complicate choices, Microprocessor Report, 10 (1996), 14-19.
- [3] S.Molnar, et al., PixelFlow: High speed rendering using image composition, Proc. SIGGRAPH, 1992.