

# マルチメディア用エンベデッドRISCのための 新アーキテクチャの開発

4F-6

三浦 宏喜 甲村 康人 松本 健志 石川 和民

三洋電機株式会社 東京情報通信研究所

## 1 はじめに

マルチメディア機器のシステムオンチップ化が進む中で、システムの中核となるCPUとして、コスト性能比や電力性能比に優れた組み込み用のRISCマイコン（エンベデッドRISC）の開発が盛んである。このようなアーキテクチャ開発の重要課題として、(1)CPUコアの小型化、低消費電力化、高速化、(2)ソフトウェアの大規模化に伴うメモリコスト増大への対応、(3)デジタル信号処理、データ検索、通信プロトコル処理、システム制御などの幅広い応用への適応が挙げられる。

我々はこれらの課題の総合的な解決を主眼とし、新しい32ビットエンベデッドRISCアーキテクチャ（開発コード名：aleph）を開発した。本稿では、まず aleph の命令セット・アーキテクチャの概要と、独自に開発した高性能16ビット命令について述べた後、この命令セットを実装したプロセッサアーキテクチャ、およびコプロセッサ接続の新方式について報告する。

## 2 命令セット・アーキテクチャ

### ① 命令セットの概要

命令セットは、主にコードサイズの最適化を容易にするために16ビット/32ビット混在とした。また汎用レジスタを32ビット×16本に抑え、16ビット命令27種132命令、32ビット命令14種162命令の豊富な命令群を備えた。

ロード/ストア命令では、バイト、ハーフワード、ワードの各アクセス命令、符号つき、符号なしの命令を全てサポートした。また、信号処理やデータ検索等の強化のために、ロードまたはストアとアドレス更新を1命令でできる方式とした。演算命令については、一般的算術論理演算の他に、信号処理や通信プロトコル処理などの強化のために、ビットサーチ、バレルシフト、ファネル

Development of a new embedded RISC architecture for Multimedia systems. Hiroki MIURA, Yasuhito KOMURA, Kenshi MATSUMOTO and Kazuhito ISHIKAWA, SANYO Electric Co., Ltd.

シフト等の命令を備えた。さらに、後述する符号化即値命令、2.5 レジスタオペランド命令といった独自の命令を開発し、マルチメディア処理への適応を目指した。

### ② 符号化即値命令

RISCアーキテクチャでは、命令に長い即値（イミディエート）を持たせることが困難であるという問題がある。本命令はその解決のために開発したものである。

本命令は、64種類の32ビットパターンを命令の即値フィールド（6ビット）に符号化して格納し、図1に示すように、命令実行時にこれを32ビットに復号してALUに送るものである。64種類のパターンとは、いずれかのビットのみが1であるパターン、およびいずれかのビット以下が全て1であるパターンである。

この新しい命令方式によって、任意ビットのビット操作、ビットテスト、およびビットフィールドの切り出し等が16ビット1命令で処理できる。

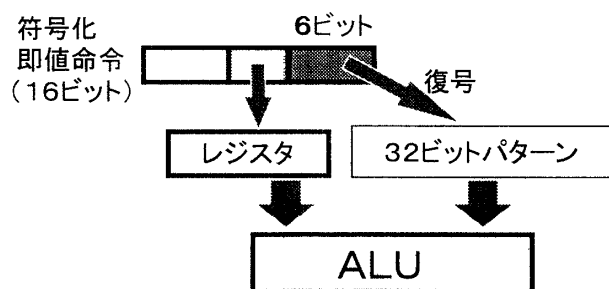


図1 符号化即値命令方式

### ③ 2.5 レジスタオペランド命令

従来のエンベデッドRISCでは、RISCの基本命令である3レジスタオペランドの演算命令を16ビット命令で実現することは極めて困難であった。

これに対し aleph では、レジスタの更新命令と更新されたレジスタを参照する命令が、時間的に近接していることに着目し、2つのソースレジスタのうち1つを最近更新されたレジスタ群の中から選ぶ方式とした。これによって、ソースレジスタのうちの1つを命令中で2ビッ

トで表現し、3レジスタオペランドの演算を、16ビット1命令で実行することを可能にした。

### 3 RISCコア・アーキテクチャ

上述の命令セットを実装した初版のRISCコア(aleph-V1)の概略構成を図2に示す。aleph-V1はI、R、A、M、Wと呼ぶ5段の命令パイプライン[1]で構成されるプロセッサである。命令実行はI（命令フェッチ）、R（命令デコード、レジスタアクセス、即値生成等）、A（演算実行）、M（メモリアクセス）、W（レジスタ書込み）を経て行なう。WステージとAステージの間には、次章で述べる新しい接続方式で、複数種類のコプロセッサが接続できる構成になっている。

Rステージの即値生成部には、符号化即値の復号回路を、命令デコード部には2.5レジスタオペランド命令のデコード機構を組み込んでいる。またAステージには、ファネルシフタおよびビットサーチのハードウェアを備えた。また、アドレス更新演算をALUで行い、レジスタファイルを2 Read 2 Writeのマルチポートで構成することで、ロード/ストア・アドレス更新命令を1サイクルで実行可能とした。これらによって、分岐命令を除く全ての命令が、1サイクルで実行可能となった。

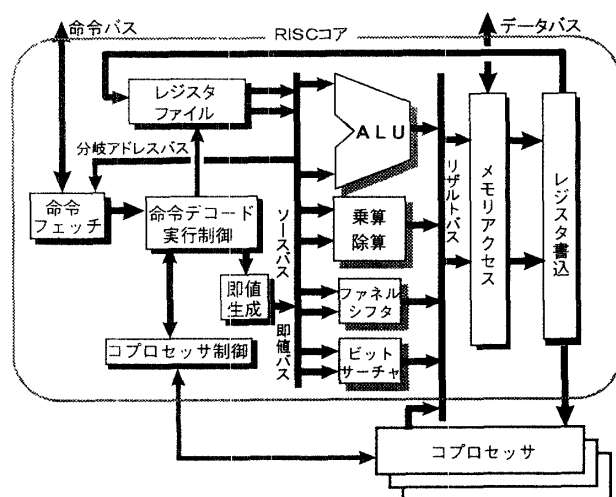


図2 aleph-V1 コア・アーキテクチャ

### 4 コプロセッサ接続方式

画像や音声のデジタル信号処理等に必要となるコプロセッサには、演算に多サイクルを要するものが多い。これをRISCコアに接続した場合、一般にコプロセッサ演算命令実行中にはRISCパイプラインが停止し、命令のスループットが大幅に低下する[2]。ここで、割

込よりも演算を優先させると演算実行中は割込禁止となり、割込への応答性も低下する。また新規コプロセッサの採用の度に、新しい演算命令を命令セットに追加することは、LSIの開発展開のオーバーヘッドともなる。

alephでは、これらの問題を総合的に解決するために、コプロセッサの新しい接続方式を開発した。すなわち、コプロセッサの入力レジスタ、出力レジスタをロード/ストア命令でアクセスできる特殊レジスタ空間に割付け、メモリから入力レジスタへのロード命令で演算を起動し、出力レジスタからメモリへのストア命令によって演算結果をRISC側に受け取る。そして、演算結果を受け取る命令は、演算終了まで命令の破棄が可能なRステージ（割込受付可能状態）で待機する方式である。

本方式は、①コプロセッサ演算命令を排除したことにより処理のスループットが向上する。②同じ理由で、新規コプロセッサの採用の際にRISCコアの変更が不要となり、LSI機種展開が容易になる。③コプロセッサにFIFO等のバッファを内蔵することにより、さらに処理のスループットを上げることができる。④コプロセッサ演算と並行して、割込処理が可能であるなど、多くの利点を有している。

### 5 おわりに

マルチメディア用に開発したエンベデッドRISCアーキテクチャalephの命令方式、プロセッサ構成、コプロセッサ接続方式について、新規性を中心に述べた。alephはマルチメディア処理を、従来よりもコンパクトなコードサイズで、かつ少ない処理サイクル数で、高速に実行できるアーキテクチャであると考えられる。

現在、既に初版RISCコアを組み込んだ1チップマイコンとASIC各1機種の試作を終えている。コアサイズは0.5  $\mu\text{m}$ プロセスで約5mm<sup>2</sup>とコンパクトである。コンパイラなどの開発ツール初版の開発も終わっており、今後はアーキテクチャの詳細評価を行う予定である。

### 参考文献

- [1]ヘネシー&パターソン（富田、村上、新實（訳））：コンピュータ・アーキテクチャ 設計・実現・評価の定量的アプローチ，日経BP社，1992年。
- [2]森河他：組み込み用途マイコンに関する高速演算器接続方式，情報処理学会研究報告，96-ARC-116-7，pp.49-54，1996年1月。