

性能向上とコード効率向上を両立させるメカニズム

## V850ES CPU コアのアーキテクチャ

本稿では、本誌付属のV850 マイコン基板に搭載されているRISC マイコンのCPU コア「V850ES」について解説する。性能向上とコード効率向上という二つの要求を満足するため、V850ES には七つの技術が盛り込まれている。例えば、命令バスとデータ・バスが分離したハーバード・アーキテクチャを採用している。また、組み込み制御応用やC 言語などの高級言語に対応するため、CISC 風の命令をいくつか用意している。(編集部)

那須 雅樹

マイクロコントローラ(以下、マイコンと呼ぶ)は、家庭用オーディオ機器やプリンタ、情報家電など、さまざまな製品で使われています。これらの機器の性能の向上や制御の複雑化に伴って、搭載されるマイコンにも性能の向上や機能の高度化が求められています。

組み込み制御の分野では、機器をコンパクトかつ安価に構成するため、外付け部品を省けるように、1チップの中にCPU(中央演算処理ユニット)コアやフラッシュROM、RAM、周辺I/Oなどが集積されます。CPUコアとは、マイコンの中核(コア)をなす演算処理用の回路ブロックのことです。フラッシュROMやRAMから命令列を順に読み込み、解釈した結果に基づいてデータを処理します。周辺I/Oは、マイコンの外部と信号をやりとりするための回路ブロックです。

こうしたマイコンでは、チップ上に搭載されたROMやRAM(オンチップ・メモリ)を有効利用するため、効率の

良い命令コードを備えたCPUコアが求められます。筆者らが開発した「V850ES」は、性能向上とコード効率の向上という二つの要求を両立させることを目指したCPUコアです。

V850 コアが採用する七つの技術

本CPUコアは、下記のことを実現しています。

- 20MHz動作時に29MIPS(Dhrystone 1.1で測定)という高性能
- 32ビットCPUでありながら16ビットCPU並みのコード効率

これは、以下の七つの技術を採用することにより達成しました。

### 1)5段パイプラインを備えるRISC CPU

本CPUコアは、シンプルなハードウェア構成で高性能を実現できるRISC(Reduced Instruction Set Computer)命令を採用しています。RISC方式では、CPUに必要な最小

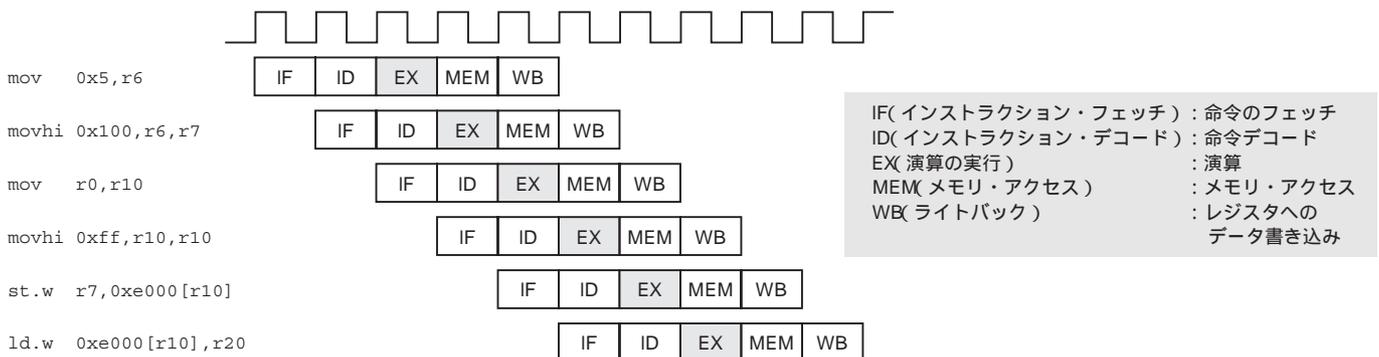


図1 パイプライン動作の例

5段のパイプライン動作で命令を実行する。ほとんどの命令を1クロックで実行できる。

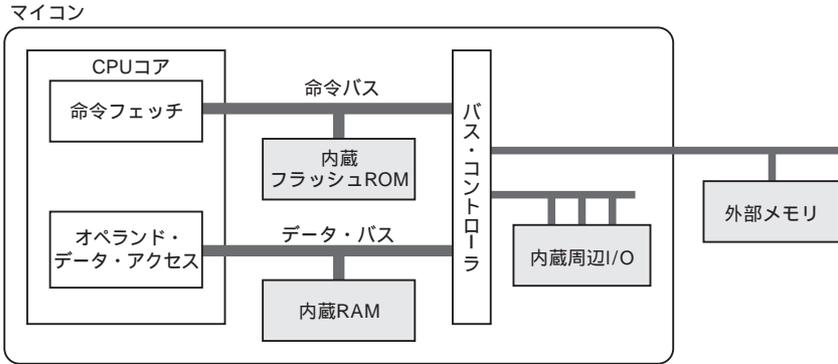


図2 ハーバード・アーキテクチャ

1サイクルで命令フェッチとデータ・アクセスを同時に実行できるため、データ・アクセス時でも命令フェッチの動作が妨げられない。

限の命令セットを持たせ、パイプライン処理<sup>注1</sup>を利用して動作速度を向上させて、全体の処理能力を引き上げます。本CPUコアの場合、例えば図1に示すように5段のパイプライン動作で命令を実行します。そのため、ほとんどの命令を1クロックごとに実行できます。

## 2) ハーバード・アーキテクチャを採用

本CPUコアはハーバード・アーキテクチャという構造を採用しています。この構造では、図2に示すように命令フェッチ用のバス(命令バス)とデータ・アクセス用のバス(データ・バス)が分離しており、それぞれ独立に動作します。1サイクルで命令フェッチ(命令をメモリから取り出す動作)とデータ・アクセス(データをメモリに書き込んだり、メモリから読み出したりする動作)を同時に実行できるた

め、データ・アクセス時でも命令フェッチの動作が妨げられずに上述のパイプライン処理を効率良く進められます。

## 3) 32本の汎用レジスタを用意

本CPUコアは32本の汎用レジスタを備えています。そのため、かなり複雑な処理であっても、メモリを使用せずに汎用レジスタのみで処理できます。これにより、処理中のメモリへのデータ読み出し/書き込みの回数を削減できます。

## 4) 命令語長は16ビットを基本とする可変長

本CPUコアは、命令セットとして16ビット長を基本とする可変長命令セットを採用しています。ロード、ストア、算術/論理演算、分岐の基本処理に16ビット長命令を適用しており、高いコード効率を実現しています。

命令コードを16ビット長に固定した場合、大きなイミディエート値(命令コード中のアドレス部や、データなどのアドレスを直接記述する値)を記載できないという問題があります。そこで、例えば16ビットのイミディエート値を扱える演算命令や、22ビットの相対分岐アドレスを用いる分岐命令なども用意しています(図3)。

このように命令コードを可変長とすることで、大きなイミディエート値でも自然に扱えるように考慮しています。

## 5) 組み込み制御向けなどのCISC風命令を用意

本CPUコアの命令セットの基本はRISCですが、組み込み制御応用やC言語などの高級言語に対応するため、CISC(Complex Instruction Set Computer)風の命令もいくつか用意しています。CISCは、RISCとは対照的に、

命令語長	命令数	命令コード・サイズ	アセンブリ・コードの記述
16ビット固定長	2命令	6バイト	やや複雑
可変命令長	1命令	4バイト	一般的な記述

(a) 命令数などの比較

16ビット固定長命令の場合 (6バイト=命令4バイト+データ・テーブル2バイト)  <pre>ld.h immデータ・テーブルのアドレス, rX add rX, rY</pre> immデータ・テーブル: .hword イミディエート・データ
可変長命令の場合(4バイト)  <pre>add imm16, rX, rY</pre>

(b) プログラムの例

## 図3 16ビット・イミディエート・データとレジスタの加算

命令長を16ビットに固定した場合、命令語長の制限で16ビット・イミディエート・データを直接命令に書くことができない。16ビット・イミディエート・データを扱うには、データ・テーブルに16ビット・データを用意し、それを参照するという処理が必要になる。

注1: CPUの高速化手法の一つ。1命令の処理を複数クロック・サイクルに分けて実行する。各クロック・サイクルの処理回路は独立に動作する。これにより、複数の命令列を“ところてん”式に処理できるようになっている。1クロック・サイクル分の処理(ロジック)がシンプルになるため、CPUの動作周波数を引き上げやすいという利点がある。