

高性能シングルチップマイクロコンピュータ H8シリーズ

High Performance Single-chip Microcomputer H8 Series

シングルチップマイクロコンピュータは、幅広い分野の機器に応用されており、性能機能向上の要求やニーズの多様化が著しい。H8シリーズは、高速動作、C言語の効率的実行、ZTAT[※])による使いやすさ、ASIC展開の容易さなどを目標に開発された、新世代シングルチップマイクロコンピュータシリーズである。

H8シリーズの第一弾H8/532は、命令のコード化順序を従来と逆転させるアーキテクチャ上の工夫などによって、従来製品のHD6301に比べてC言語使用時で3倍のCPU性能を実現した。CPUの高性能化に加えて、大容量PROM(32 kバイト)、高速高精度A-D変換器などを内蔵しており、応用機器の性能向上に役立つとともに、従来、マルチチップにしないわけにはいかなかったシステムのシングルチップ化が可能となった。

馬場志朗* *Shirō Baba*
 松原 清* *Kiyoshi Matsubara*
 野口孝樹** *Kouki Noguchi*
 駒川 融*** *Tōru Komagawa*
 藪下正治**** *Masaharu Yabushita*

1 シングルチップマイクロコンピュータの市場動向

マイクロプロセッサの開発では、32ビットマイクロコンピュータの開発競争に話題が集中しがちであるが、実際の市場では、民生、産業、OA(Office Automation)などの機器組み込み用として、いわゆるシングルチップマイクロコンピュータが、広く、かつ多量に使われており、マイクロエレクトロニクス革命を推進する原動力として、その影響力は32ビットマイクロコンピュータに負けず劣らず大きい。

シングルチップマイクロコンピュータの分野でも、プロセス技術の進歩を背景に、性能、機能の目覚ましい向上が実現されている。また、応用分野が拡大し、要求される仕様も機能性能重視形からコスト優先の分野まで、ニーズの多様化が著しい。一方、フィールドプログラマビリティの向上など、使い勝手の改善を図る動きや、はん(汎)用品ではなく、特定応用分野向け専用マイクロコンピュータを求める、いわゆるASIC(Application Specific Integrated Circuits)化といった新しい流れもある。

また、シングルチップマイクロコンピュータの世界でも、応用機器の多様化や機能の複雑化に伴い、プログラムサイズが飛躍的に増大し、いわゆる「ソフトウェア危機」が表面化してきている。

日立製作所では、このようなニーズの多様化、新しいニーズの流れにこたえるため、新世代のシングルチップマイクロコンピュータシリーズとして、H8シリーズを開発した。

2 H8ファミリーとその開発コンセプト

2.1 H8ファミリーのコンセプトと開発のねらい

H8の開発に当たっては、以下を目標とした。

(1) 高速動作CPU

応用機器の性能向上には、なによりもまずCPU(Central Processing Unit)の高速化が不可欠である。H8ファミリーは16ビットの加減算をはじめ基本的なレジスタ間演算が0.2 μs(10 MHz動作時)という高速動作を達成している。

(2) 高級言語の効率的実行に適したCPUアーキテクチャ

シングルチップマイクロコンピュータの分野でも、ソフトウェアの増大が著しく、その開発効率向上は必ず(須)の課題となりつつある。そのためには、コード効率を重視するシングルチップマイクロコンピュータといえども、高級言語の使用が現実的に必要となってきた。H8ファミリーでは、はん用レジスタ構成をとり、かつ各種の工夫を行い、C言語などの高級言語を効率的に実行できるアーキテクチャを実現している。

(3) ZTATコンセプトの徹底追求

マイクロコンピュータ応用システムの開発期間を著しく短縮できるのが、日立オリジナルのZTAT(Zero Turn Around Time)コンセプトである。H8ファミリーではこれを徹底し、ZTAT版〔One Time PROM(Programmable Read Only Memory)内蔵版〕とマスクROM版の同時開発と同時提供を実現した。ZTATは、開発段階ばかりでなく、多品種少量生産や、量産の立上げ時などに広く使うことができる。

(4) ASICへの対応

H8シリーズでは、日立製作所のチップ内標準バスであるSBP

※) ZTATは、日立製作所の登録商標である。

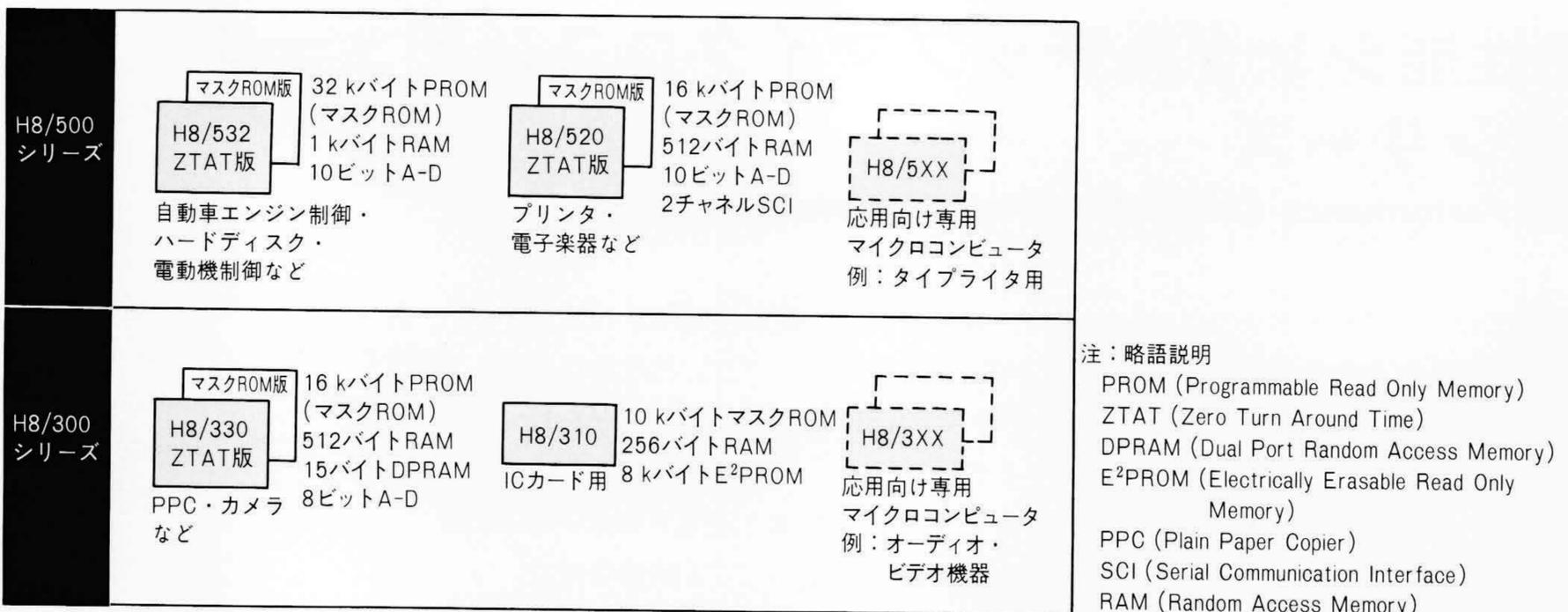


図1 H8ファミリーの製品展開 H8シリーズは、上位機種H8/500シリーズ、下位機種H8/300シリーズの二つのCPUコアを持つ。

(Silicon Back Plane)バスを採用し、H8、H16の各ファミリー間で内蔵周辺機能モジュールを共通化している。これにより、各種内蔵周辺モジュールと各種CPUの組み合わせにより種々の応用向け専用IC(ASIC)の展開が可能である。

2.2 H8ファミリーの製品展開と応用分野

図1にH8ファミリーの製品展開と応用分野を示す。シングルチップマイクロコンピュータの応用分野が広範囲にわたっていることを考慮して、二つのCPUコアを準備した。H8/500シリーズは、内部16ビット構成の高性能高機能のCPUコアを持ち、性能重視形のマーケット向けである。一方、H8/300シリーズは、コストパフォーマンスを重視する分野を主なターゲットとするCPUコアを持つシリーズである。

具体的な応用分野としては、H8/500シリーズでは自動車エンジン制御、ハードディスク、電動機制御、電子楽器など、16ビットのデータを高速処理する必要のある分野や、64kバイト以上のアドレス空間が必要な分野などがある。一方、H8/300シリーズはオーディオ・ビデオ機器、コピー機、カメラ、ICカードなど、高速動作と高いコストパフォーマンスを生かして幅広い制御機器への応用が可能である。

3 H8/532の機能

次に、H8シリーズの第一弾として開発したH8/532の詳細について説明する^{1),2)}。表1に仕様の一覧を、図2にブロック図を、図3にチップ写真を示す。内部16ビット構成のH8/500CPUを中心に、大容量のメモリ〔32kバイトEPROM(Erasable and Programmable Read Only Memory)、1kバイトRAM(Random Access Memory)〕、A-D変換器、タイマ、SCI(Serial Communication Interface)などを内蔵しており、世界でもトップクラスの高性能シングルチップマイクロコンピュータである。使用プロセスは1.3μ CMOS(Complementary Metal Oxide Semiconductor)技術であり、9.8×9.9mm²に42万個のトランジスタを集積化している。内蔵機能の主な特徴は次の点である。

(1) 大容量ZTATマイクロコンピュータ

表1 H8/532の仕様概要 H8/532は、16ビット構成のCPUに加えて、大容量メモリなどシステムの構成に必要な豊富な機能を内蔵している。

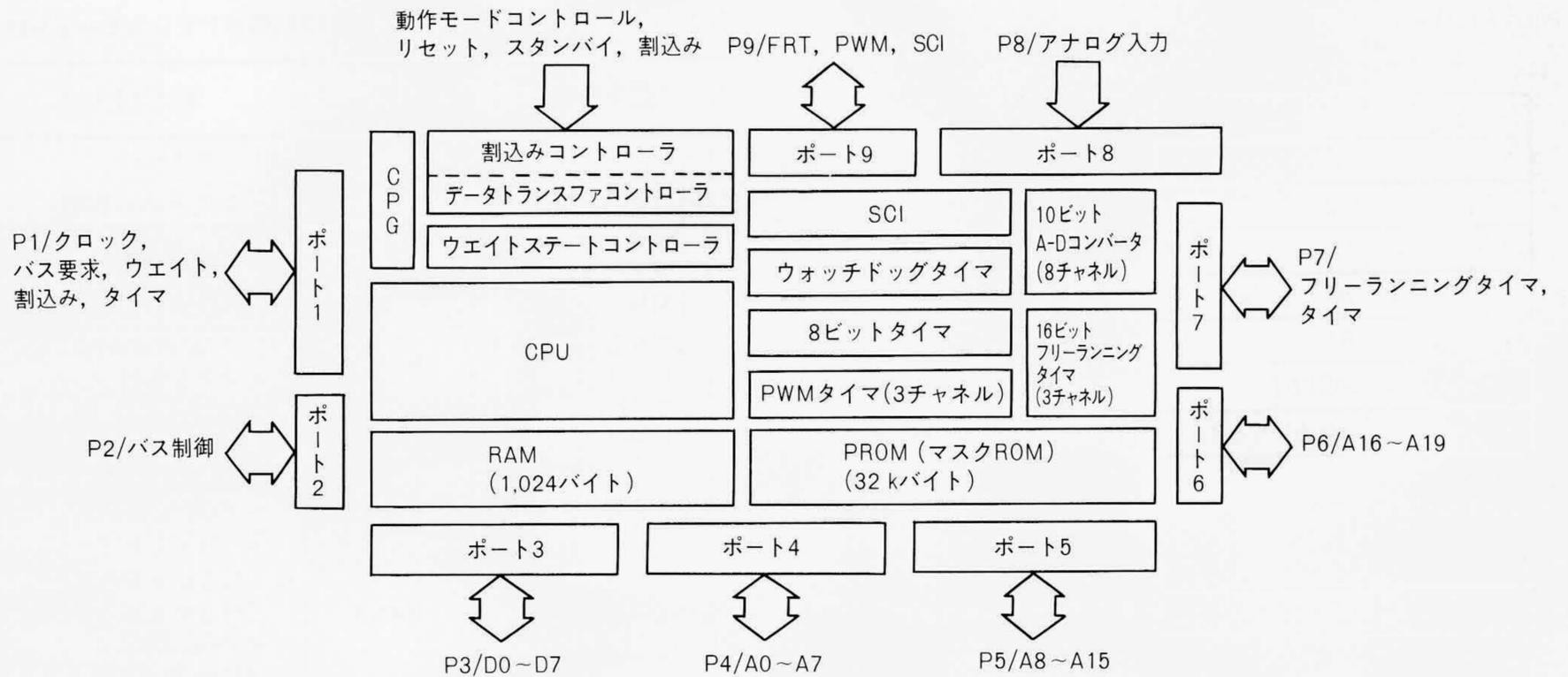
項目	内容
CPU	H8/500CPU(内部16ビット構成)
ROM	32kバイト(PROM/マスクROM)
RAM	1,024バイト
タイマ	16ビットフリーランニングタイマ：3チャンネル (インプットキャプチャ3本, アウトプットコンペヤ6本) 8ビットタイマ：1チャンネル(コンペヤ2本) PWMタイマ：3チャンネル ウォッチドッグタイマ：1チャンネル
SCI	1チャンネル(調歩同期式, クロック同期式)
A-D	10ビット, 8チャンネル(単一モード, スキャンモード)
INTC	外部割込み：3本 内部割込み：19 優先順位：8レベル
DTC	データトランスファコントローラ内蔵
WSC	ウェイトステートコントローラ内蔵
I/Oポート	入出力共通端子：57本 入力専用端子：8本
パッケージ	84ピンPLCC 84ピン窓付きLCC 80ピンQFP
プロセス	CMOS 1.3μm

注：略語説明
 PLCC(Plastic Leaded Chip Carrier)
 QFP(Quad Flat Plastic Package)
 INTC(Interrupt Controller)
 WSC(Wait State Controller)

32kバイトの大容量PROM(世界最大)を内蔵しており、はん用のPROMライターで自由にプログラムを書き込むことのできるZTATマイクロコンピュータである。大容量メモリを内蔵しているため、従来はマルチチップにせざるを得なかったシステムのシングルチップ化が可能になり、実装面積の縮小を実現できる。

(2) 高速高精度A-D変換器

分解能10ビット、変換時間13.8μsは、オンチップA-D変換



注：略語説明 CPG (Clock Pulse Generator), CPU (Central Processing Unit)

図2 H8/532ブロック図 16ビット構成のCPUを中心に、大容量のメモリ(32 kバイトPROM, 1 kバイトRAM), A-D変換器, タイマ, シリアルなどを内蔵している。

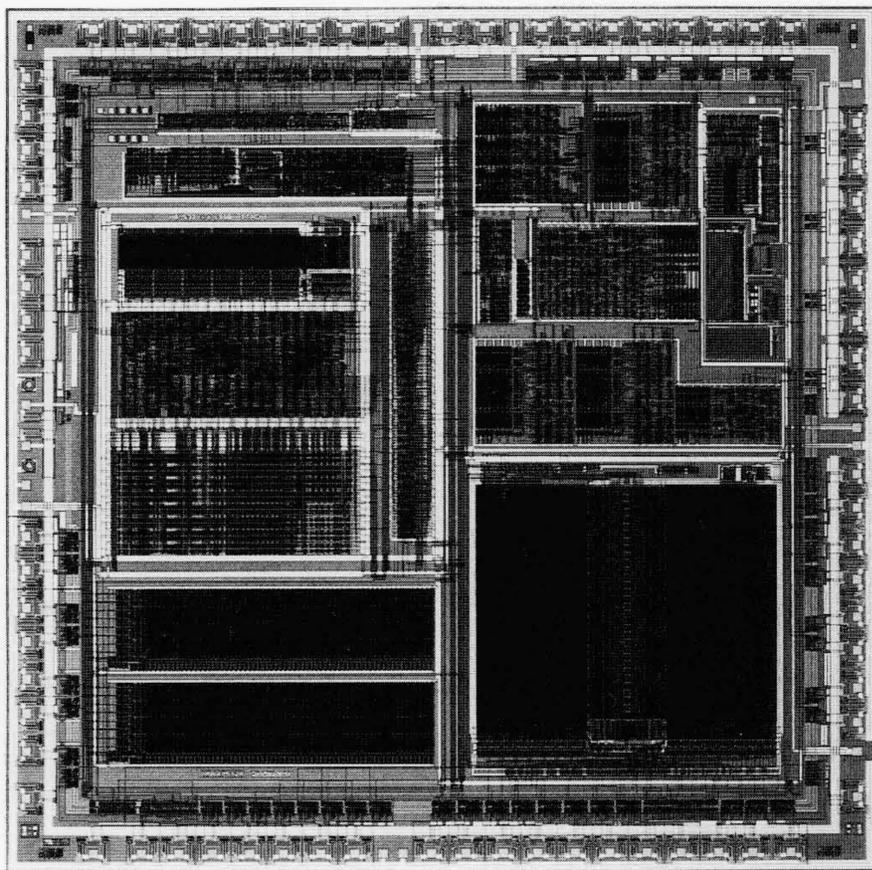


図3 H8/532のチップ写真 1.3 μm CMOS技術により、9.9×9.8mm²に42万個のトランジスタを集積化している。

器としては世界最高速であり、例えば、電動機の制御に最適である。

(3) 多機能タイマ

日立製作所の従来のシングルチップマイクロコンピュータHD6301Yの約3倍のタイマを内蔵しており、自動車のエンジン制御など、多数のパルスの入出力制御が必要な応用に適している。また内蔵のウォッチドッグタイマは、システムの信頼性向上に役立つ。

(4) DTC

DTC(データ転送コントローラ)は、CPUの介在なしに、I/O(入出力装置)とメモリ間のデータ転送を実行することができる、一種の簡易DMA(Direct Memory Access)機能であり、CPUの負担を軽減し、ソフトウェアの複雑化を防ぐのに有効である。

4 H8/500CPUアーキテクチャとその性能

4.1 CPUアーキテクチャの特長

CPUアーキテクチャの主な特長を以下に説明する。

(1) はん用レジスタ方式の16ビットCPU

図4にレジスタ構成を示す。8本の16ビット長はん用レジスタは、データ演算用又はアドレスポインタ用のいずれにも使うことができ、また、8ビットデータ、16ビットデータのいずれの演算用にも使用できる。このように、レジスタのはん用性が高いことは、プログラム作成時に変数をレジスタへ割り付ける場合の制約が少ないことを意味しており、プログラム作成が容易となる。

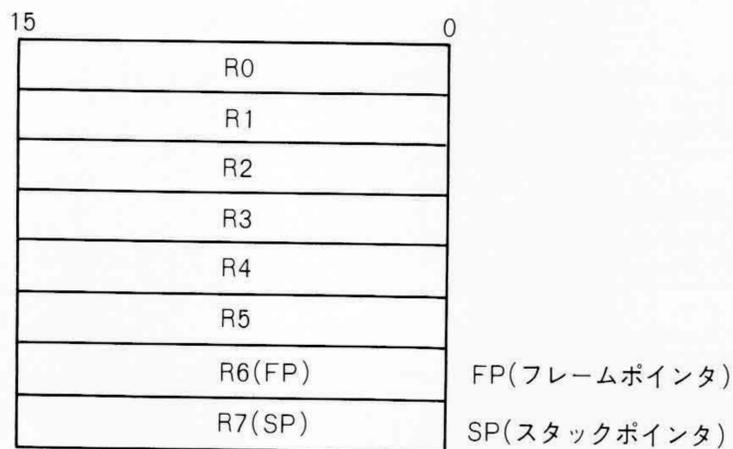
(2) 直交性の高い命令形式

命令形式としては、レジスタ~レジスタ間演算、レジスタ~メモリ間演算を基本とする、いわゆる1.5アドレス形式をとっている。一般の命令では、任意のアドレスモード(表2)と任意のデータサイズ(8ビット又は16ビット)の組み合わせが可能であり、柔軟性が高い。これもまた、プログラム作成の容易化に役立つ。

(3) 強力な命令セット

表3に命令の一覧を示す。一般的な命令に加えて、レジスタによるビット位置の間接指定が可能なビット操作命令や、高級言語のためのLINK/UNLK命令などを持っている点に特長がある。

Rn[はん(汎)用レジスタ]



CR(コントロールレジスタ)

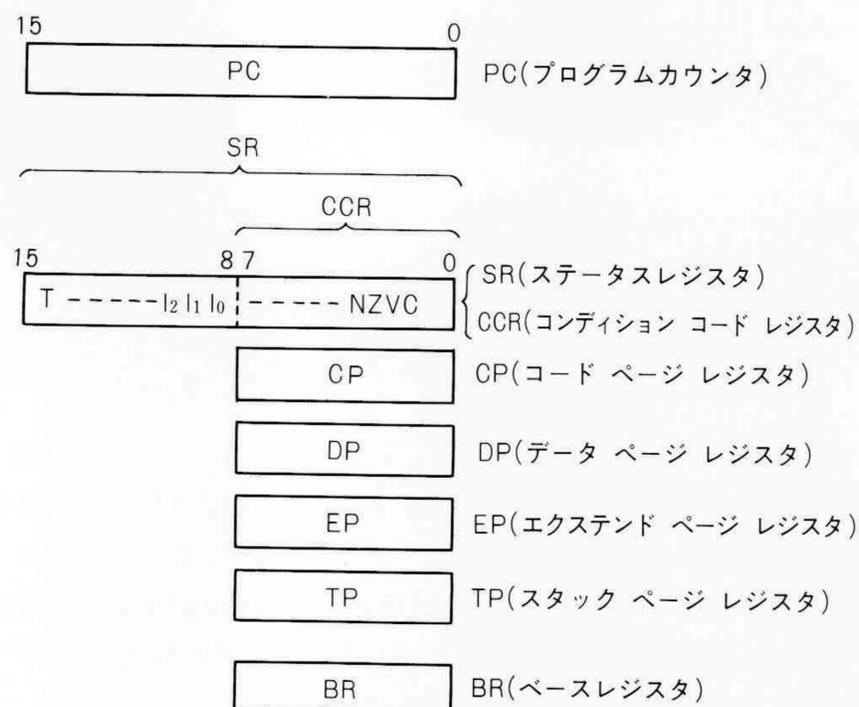


図4 H8/500CPUの内部レジスタ構成 8本の16ビット長のはん(汎)用レジスタは、データ演算用、アドレスポインタ用のいずれにも使うことができる。4本のページレジスタは、アドレスを64kバイト以上に拡張するために使われる。

表2 H8/500のアドレスモード 7種類のアドレスモードを持つ。

No.	アドレッシングモード	ニーモニック	実効アドレス
1	レジスタ直接	Rn	● レジスタRn
2	レジスタ間接	@ Rn	● レジスタRnの内容
3	ディスプレイメント付きレジスタ間接	@(d:8, Rn) @(d:16, Rn)	● レジスタRnの内容+ディスプレイメント(8ビット又は16ビット)
4	プリデクリメントレジスタ間接 ポストインクリメントレジスタ間接	@ -Rn @ Rn+	● レジスタRnの内容をデクリメントした値 ● レジスタRnの値(実行後Rnをインクリメント)
5	イミディエイト	#xx:8 #xx:16	● オペランドデータは命令コード中に含まれる8ビット又は16ビットデータ
6	絶対アドレス	@ aa:8 @ aa:16	● 上位8ビットはBRの内容、下位8ビットを命令コード中で直接指定 ● 16ビットアドレスを命令コード中で直接指定
7	PC相対	disp	● PCの内容+ディスプレイメント(8ビット又は16ビット)

注：略語説明 d, disp(ディスプレイメント)

(4) 広いアドレス空間

従来の8ビットマイクロコンピュータは64kバイトのアドレス空間をアクセスできるものが大部分であったが、最近では大容量メモリの普及に伴い、プリンタ、タイプライタ、ファクシミリなど64kバイト以上の空間が必要な応用が多くなってきている。H8/500CPUは64kバイト以内のミニマムモードに加えて、64kバイト以上の空間をアクセスできるマキシマムモードをサポートしている。マキシマムモードでは、図4の4本のページレジスタを使い、1ページ64kバイトを単位とし

表3 H8/500の命令セット概要 算術、論理、比較などの一般的命令に加えて高級言語用のLINK/UNLK命令を持っている。また、ビット操作命令はビット位置の間接指定が可能である。

命令区分	種類	代表例	命令区分	種類	代表例	
転送	転送	MOV 転送	単項演算	クリア	CLR クリア	
		STM 複数レジスタの退避		サイズ拡張	EXTS 符号拡張 EXTU ゼロ拡張	
算術演算	加算	ADD 2進加算	ビット操作	ビット操作*	BTST ビットテスト	
		DADD 10進加算			BSET ビットテスト及びセット	
	減算	DSUB 10進減算			BCLR ビットテスト及びクリア	
	乗算	MULXU 符号なし乗算		スタック操作	LINK スタックフレーム形成	
	除算	DIVXU 符号なし除算			UNLK スタックフレーム解放	
	符号反転	NEG 2の補数			ページ内分岐	BCC 条件分岐 BSR サブルーチンコール RTS サブルーチンからの復帰
論理演算	論理積	AND 論理積	分岐系	ページ間分岐	PJMP ページ間ジャンプ	
	論理和	OR 論理和			PJSR ページ間サブルーチンコール	
	排他的論理和	XOR 排他的論理和		ループ制御	PRTS ページ間サブルーチンから復帰	
	否定	NOT 否定			SCB ループ命令	
比較	比較	CMP 比較	例外処理	例外処理	TRAPA 例外処理命令	
	テスト	TST "0"との比較			RTE 例外処理ルーチンから復帰	
シフト	シフト	SHAL 算術左シフト	システム制御	システム制御	LDC コントロールレジスタへのロード	
		SHLR 論理右シフト			SLEEP 低消費電力モードへの移行	
	ローテート	ROTR 右ローテート				
		ROTXL キャリー付き左ローテート				

注：* ビット位置の指定は、レジスタによる間接指定が可能である。

で最大16 Mバイトまでの空間をアクセスすることができる。

4.2 CPUの性能

表4に、命令実行時間の例を示す。レジスタ間の加減算、論理演算の0.2 μs、16ビット乗除算の2.3、2.6 μs(10 MHz動作時)などは、シングルチップマイクロコンピュータとして世界最高速である。また、内部16ビット構成のCPUであるので、加減算などでは16ビットデータの演算も8ビットデータと同一の時間で実行される。

16ビットデータの高速乗除算は、例えば、サーボモータの制御性能向上に大きな効果がある。また、各種のデジタル信号処理の応用分野のうち、音声帯域以下の分野では、DSP(Digital Signal Processor)の代用として使用可能な演算性能である。

シングルチップマイクロコンピュータの代表的応用分野から選んだベンチマークプログラムについて、各種CPUと比較した結果を図5に示す。同図から分かるように、従来の6301に比べて、アセンブラでプログラムした場合で約2.5倍、C言語を使った場合で約3倍の性能向上を実現している。

表4 H8/500命令実行時間の例 シングルチップモード、10 MHz動作時の実行時間である。

演算	レジスタ～レジスタ間演算		レジスタ～メモリ*間演算	
	8ビット	16ビット	8ビット	16ビット
論理演算	0.2 μs	0.2 μs	0.6 μs	0.6 μs
加減算	0.2 μs	0.2 μs	0.6 μs	0.6 μs
乗算	1.6 μs	2.3 μs	2.0 μs	2.6 μs
除算(16÷8) (32÷16)	2.0 μs	2.6 μs	2.4 μs	3.0 μs

注：* ポストインクリメントレジスタ間接アドレッシングモード時

5 高速化実現のアーキテクチャ上の手法

前章で述べた高速性能を実現するためには、動作周波数の向上に加えてアーキテクチャ上の各種の工夫がされている。その主な手法について以下に説明する。

(1) 内部バスの構造

図6にCPUの内部構造を示す。はん用レジスタ、ALU(演算器)はすべて16ビット構成であり、それを結ぶバスも16ビット幅である。この構造により、16ビットの内部演算は1クロックで実行可能になり、命令実行の高速化に寄与している。また、CPUとROM/RAMのメモリをつなぐ内部バスも16ビット幅となっており、かつ2クロックサイクルでデータをリード・ライトしている(外部バスの場合は3クロック)。これにより、内蔵メモリからのプログラム及びデータのフェッチの高速化が実現されている。

(2) 命令のコード化手法

H8/500CPUでは、高速化を図るために、命令のコード化に従来と異なる方式を用いている。従来は、命令の最初に命令の種類や演算の種類を示すOP(Operation)コード、次にオペランドをどこからフェッチしてくるかを示すEA(Effective Addressコード=アドレスモードの指定コード)の順であったのに対し、H8/500ではこれを逆転させている(図7参照)。理由は、CPU内部の命令実行手順から考えると、EAコードを先に解釈し処理したほうが高速化できるからである。

命令の実際の実行は、まずEAコードの示すアドレスからオペランドデータをフェッチし、次にそのデータに対しOPコードで指定される演算を行うという手順となる。したがって、CPU内部の命令デコーダによるEA/OPコードの解釈とその処理実行のパイプライン動作を最適化するには、EAコード、OPコードの順に命令コードを構成したほうが効率的である。こうすると、EAコードで指定した処理の実行とOPコードの解釈を並行して実行でき、高速化が実現できる。

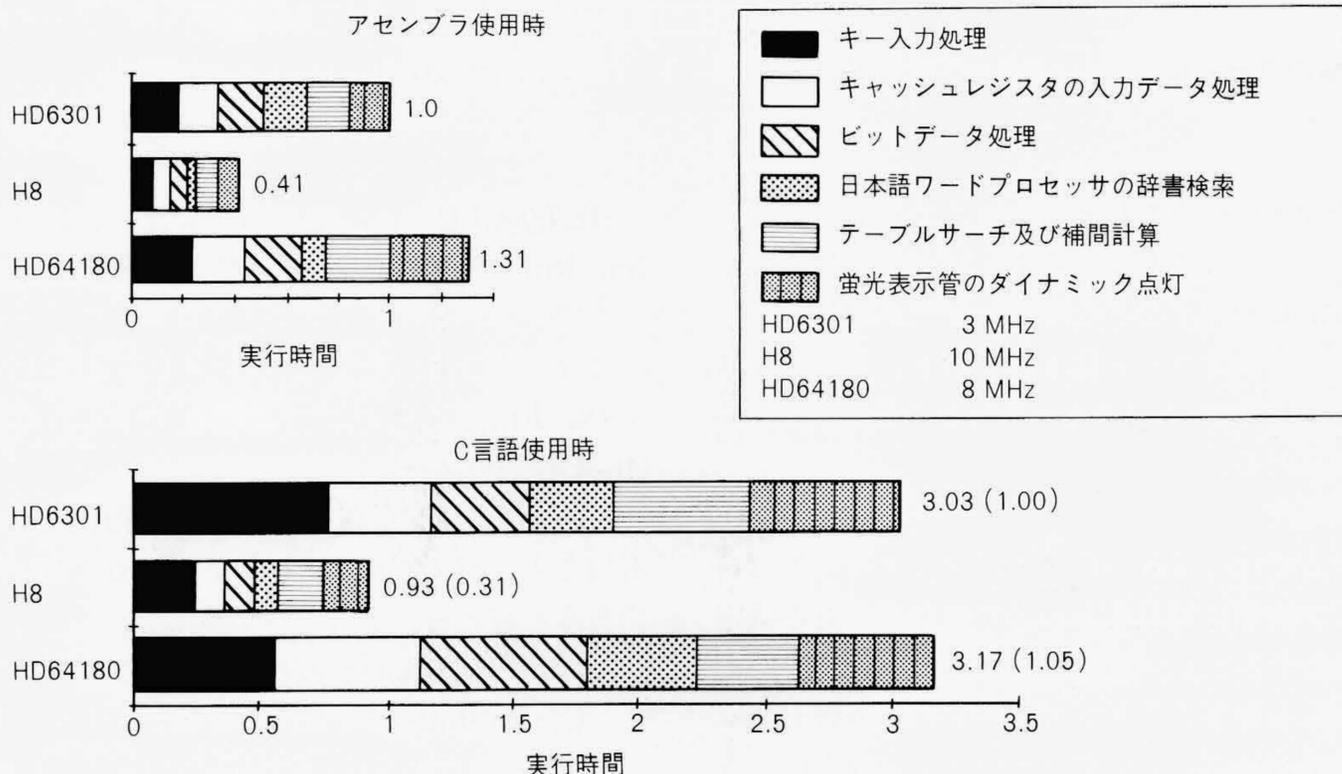
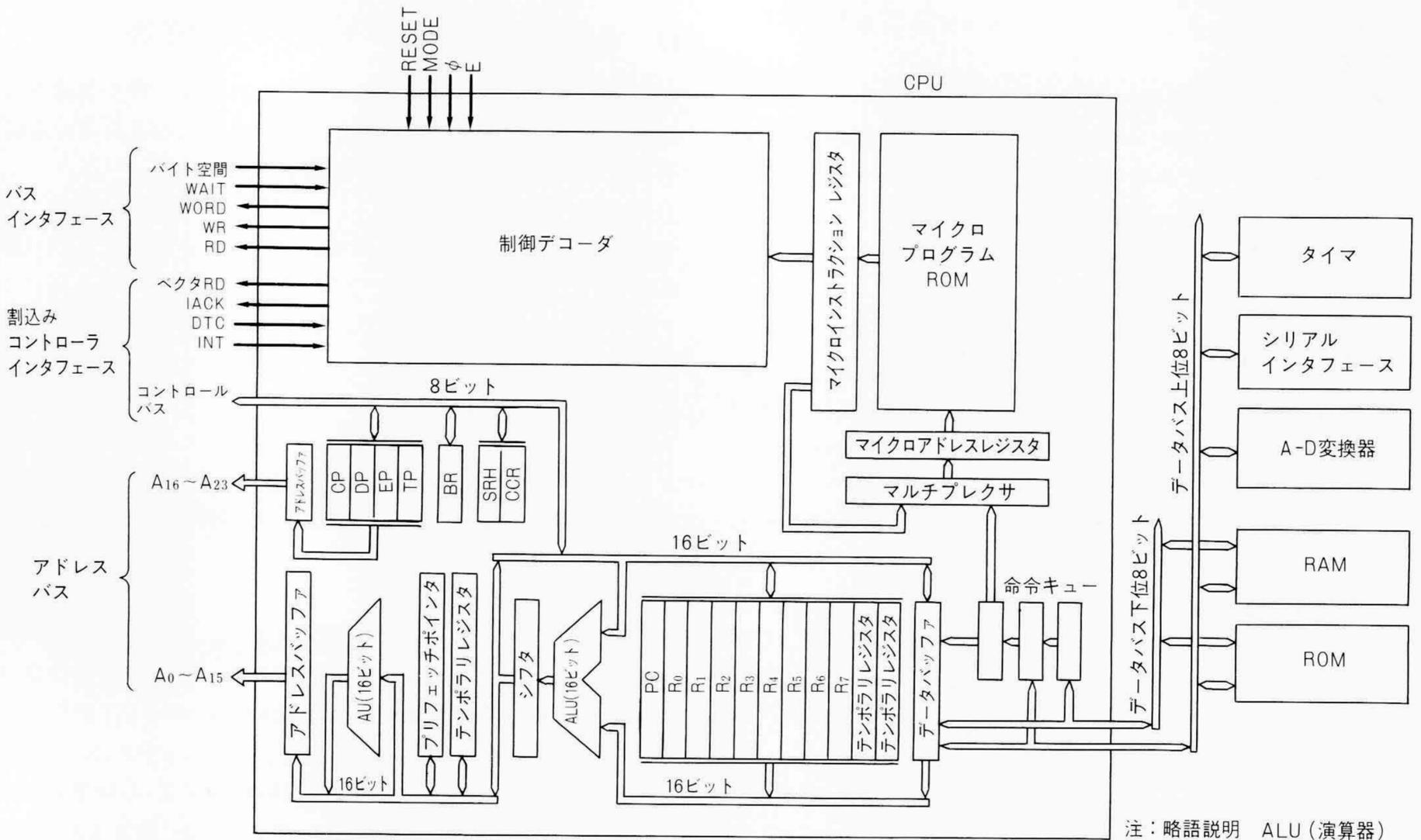
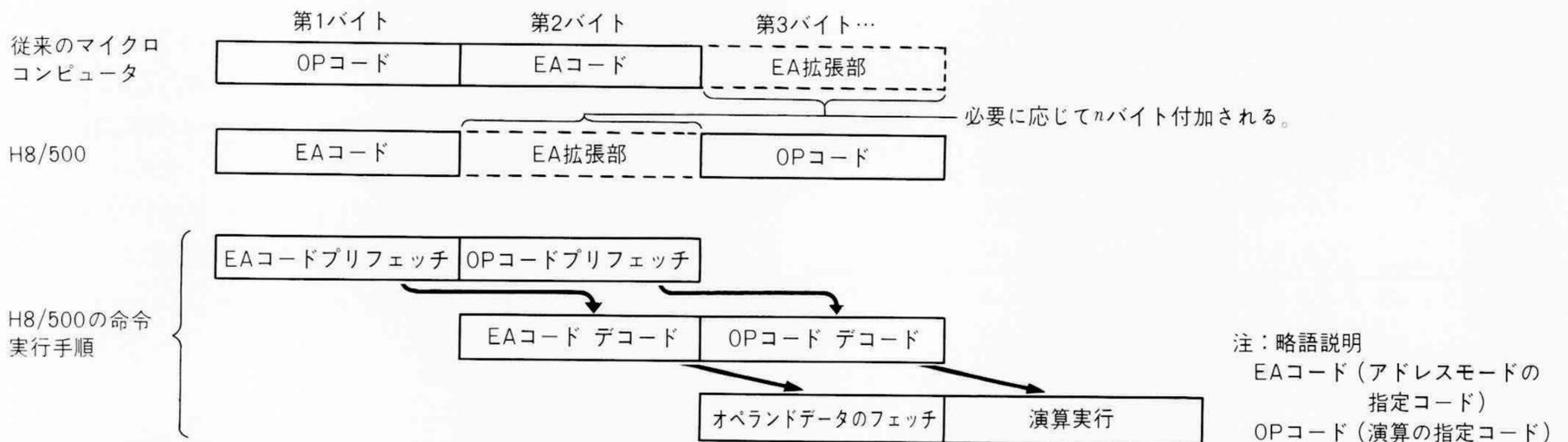


図5 ベンチマークテスト結果 シングルチップマイクロコンピュータの代表的分野から選んだベンチマークプログラムについて、各種CPUと比較する。



注：略語説明 ALU (演算器)

図6 H8/500CPUの内部構造 はん用レジスタとALU間のバス、CPUと内蔵メモリの間のバスは、いずれも16ビットバスとなっている。



注：略語説明
EAコード (アドレスモードの指定コード)
OPコード (演算の指定コード)

図7 H8/500の命令のコード化方法 通常のCPUとは逆に、EAコード、OPコードの順で命令がコード化されている。

なお、演算命令などデータを扱う命令はこの方式でコード化されているが、データを扱わない命令、例えば分岐命令やNOP (Non Operation) コードなどの特殊命令では、OPコードが先頭にくる通常の方法でコード化されている。また、先頭がEAコードであるかOPコードであるかは命令の先頭バイトの上位ビットで区別される。

(3) 頻出命令の短縮化

現実のプログラムでは、多数の命令とアドレスモードの組み合わせのすべてが均等に使われるわけではなく、ある特定の組み合わせが頻出することが経験的に知られている³⁾。したがって、よく使われる命令とアドレスモードの組み合わせを特別の短縮形としておけば、プログラムのコード効率向上及び高速化を実現できる。

H8/500CPUでは、定数のレジスタへのロード命令 (MOV Im, Rn)、定数とレジスタ内容の比較命令 (CMP Im, Rn)、そして、シングルチップマイクロコンピュータでI/Oのアクセスによく使われるショート絶対アドレスMOV命令 (MOV @xx, Rn)などを短縮形とし、高速化を図っている。

(4) C言語の高速実行

これまでに述べた高速化手法はいずれもC言語の高速実行に有用であるが、これらに加えて次の点でH8/500はC言語向きであると言える。

(a) 16ビット演算が高速であること。

C言語のInteger変数は通常16ビット長で実現されるので、Cコンパイラの生成コードでは、16ビット演算が頻出する。

(b) 16ビット長のはん用レジスタ8本を持つこと。

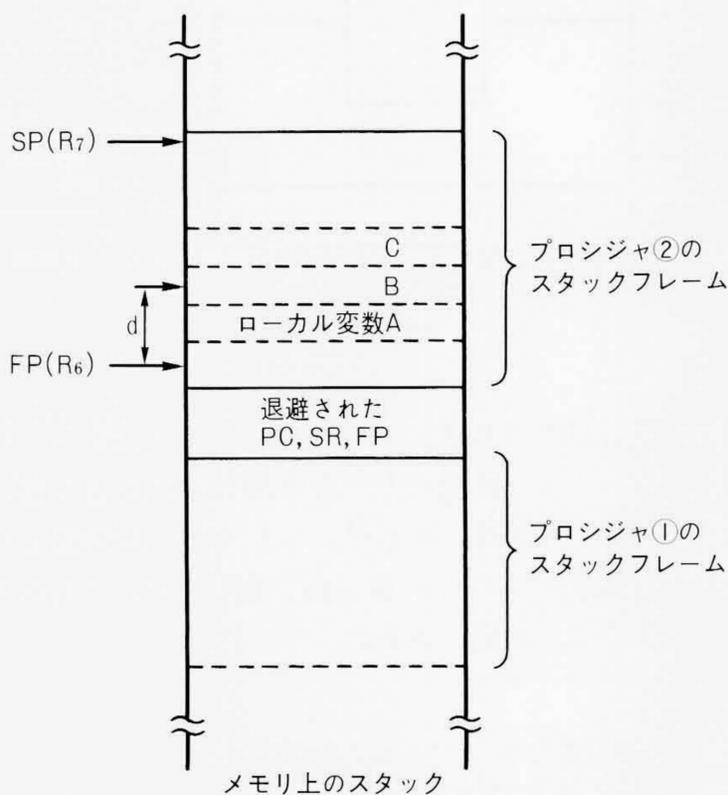
C言語ではポインタが多用され、また高速化の手法として「レジスタ変数」を使うことができる。これらを効率よく実現するには、はん用レジスタをできるだけ多数本持っていたほうがよい。

(c) LINK/UNLK命令

C言語のプロシジャコールの際のスタックフレームの生成や解消のために、これらの命令が用意されている。

(d) スタックフレーム上の変数のアクセスの高速化

C言語のプロシジャ内のローカル変数は図8に示すようにスタックフレーム上にとられ、フレームポインタからの相対アドレスでアクセスされる。したがって、ディスプレイメント付きフレームポインタ間接のアドレスモードが多用される。そこで、H8/500では、はん用レジスタのR6をフレームポインタとして使いやすいように、ショートディスプレイメント付きR6間接MOVE命令を短縮形命令として用意し、C言語のコンパイルコードの高速化を図っている。



例えば、ローカル変数Bのアクセスは、MOV@(d, R6), Rnのようにフレームポインタからの相対アドレスで行われる。

図8 C言語におけるローカル変数のアクセス方法 ローカル変数は、フレームポインタからの相対アドレスでアクセスされる。

6 開発環境

H8シリーズの開発ツールの一覧表を表5に、また、典型的な開発システムの構成例を図9に示す。ホストコンピュータとしては、一般的なパーソナルコンピュータやワークステーション、ミニコンピュータなどが接続できる。ユーザーの開発ツールへの投資を最小とするため、リアルタイムエミュレータは、Hシリーズに共通の本体(ASEステーション)と、各LSIに固有のエミュレータボックスに分割されている。したがって、Hシリーズ内の別のLSIをデバッグする場合は、エミュレータボックスだけ交換すればよい。

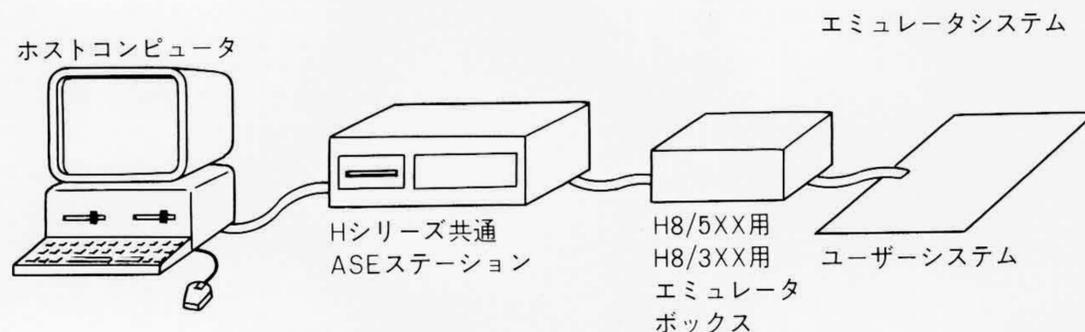
7 応用例

典型的な応用例として、自動車エンジン制御システムの構成例を図10に示す。この例では、H8/532の16ビットデータの高速度演算性能、大容量メモリ、高機能タイマ、高精度A-D変換器などの長を生かしている。このほかにも、インバータやサーボモータ制御、ハードディスクの制御などの分野は、いずれも16ビットデータの高速度演算や高速A-D変換が有効な分野である。一方、電子楽器やタイプライタ、プリンタなどの分野はCPUの高速度動作に加えて、広いアドレス空間を生かせる分野である。

表5 開発ツール一覧表 リアルタイムエミュレータ、評価ボードなどのハードウェアや、C言語などのソフトウェアが用意されている。

サポートソフトウェア	サポートハードウェア
●Cコンパイラ	●ASEステーション
●アセンブラ	(Hシリーズ共通)
●リンケージエディタ	●エミュレータボックス
●ライブラリアン	●評価ボード
●オブジェクトコンバータ (SYSROF→Sタイプ)	—
●シミュレータデバッガ	—
●ITRON	—

注：略語説明 SYSROF (開発ツールメーカーなどが進めている標準オブジェクト形式)
ITRON (リアルタイムオペレーティングシステム)



注：略語説明 ASE (Adaptive System Emulator)

図9 H8/500開発システムの構成例 ホストコンピュータとしては、一般的なパーソナルコンピュータやワークステーション、ミニコンピュータなどが接続できる。

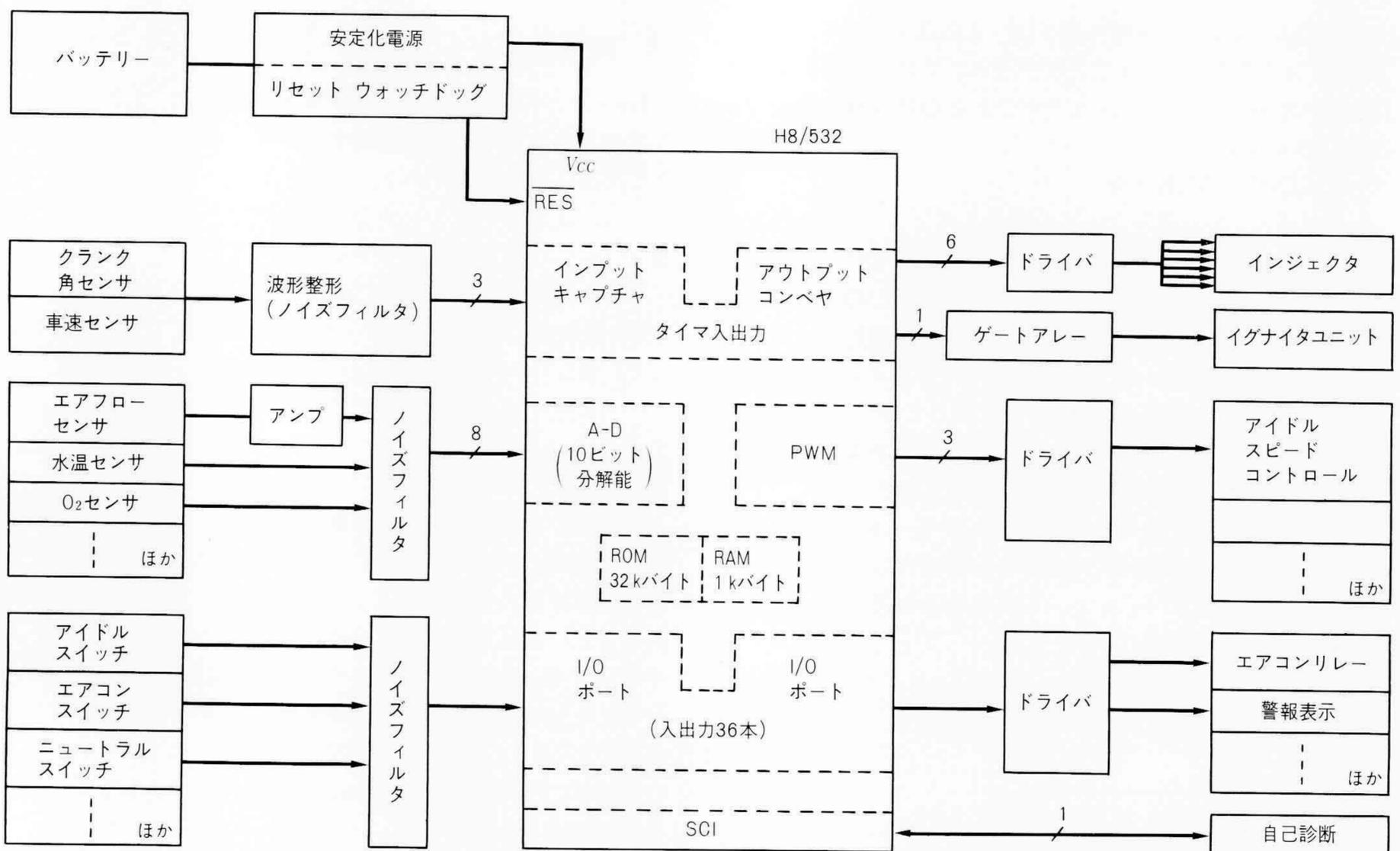


図10 H8/532を使ったエンジン制御システムの構成例(6気筒エンジン) H8/532ワンチップだけで高性能なエンジン制御システムを構成できる。

8 結 言

以上、新世代の高性能シングルチップマイクロコンピュータH8/532について、CPUアーキテクチャの特長及び高速動作実現の手法を中心に述べた。動作周波数向上及びアーキテクチャ上の工夫により、従来のHD6301に比べて、アセンブラで2.5倍、C言語で3倍のCPU性能が得られた。また、高速動作CPUに加えて、大容量PROM、高速・高精度A-D変換などの内蔵を実現しており、応用機器の性能向上はもちろん、従来、シングルチップマイクロコンピュータが適用できなかった分

野への応用拡大も可能である。

今後とも、いっそうの性能向上、各応用分野に最適な周辺機能内蔵などのニーズが強いすう勢に合わせ、今回開発したH8CPUコアを核に、ファミリー製品群を充実させ、これらのニーズにこたえてゆく予定である。

参考文献

- 1) H8/532 プログラミングマニュアル(1988年)
- 2) H8/532 ハードウェアマニュアル(1988年)
- 3) Wiecek, C.A. : A Case Study of VAX-11 Instruction Set Usage, ACM 1982, pp.177~184