

高性能浮動小数点演算プロセッサ

High Performance Floating-point Processing Units

情報処理や制御の分野で、32ビットマイクロプロセッサの応用が広く展開されている。これらの分野では、整数演算だけでなく浮動小数点演算が必要になる。日立製作所では、それに対処するために32ビットマイクロプロセッサと同時に浮動小数点演算プロセッサを開発した。

この浮動小数点演算プロセッサの開発に当たっては、演算の高速化を重点に進めた。特にデータ転送のオーバーヘッドの削減、演算処理の並列化によって、演算時間の短縮を図った。この結果、浮動小数点演算プロセッサの性能指標となる科学技術計算のWhetstoneベンチマークで3.2~4.0 MWIPSを達成できた。

森永茂樹* Shigeki Morinaga
河崎俊平** Shumpei Kawasaki

1 緒 言

近年、32ビットマイクロプロセッサは、高性能パーソナルコンピュータ、ミニコンピュータ、ワークステーション、グラフィック、NC(数値制御)、計測制御などの分野へ本格的に普及しつつある。これらの応用分野では、高速・高機能のCPU(Central Processing Unit)のほかに、強力な科学技術計算能力及び座標変換機能をシステムに付加するFPU(Floating-point Processing Unit: 浮動小数点演算プロセッサ)が不可欠である。

このような状況により、日立製作所では上記システムの高性能化に大きく寄与する32ビットCPU(H32/200)とFPUであるHFPU(High performance Floating-point Processing Unit)を開発した。このHFPUは、浮動小数点演算の世界的標準仕様であるIEEE(米国電気電子学会)規格に準拠している。

以下、HFPUの詳細について述べる。

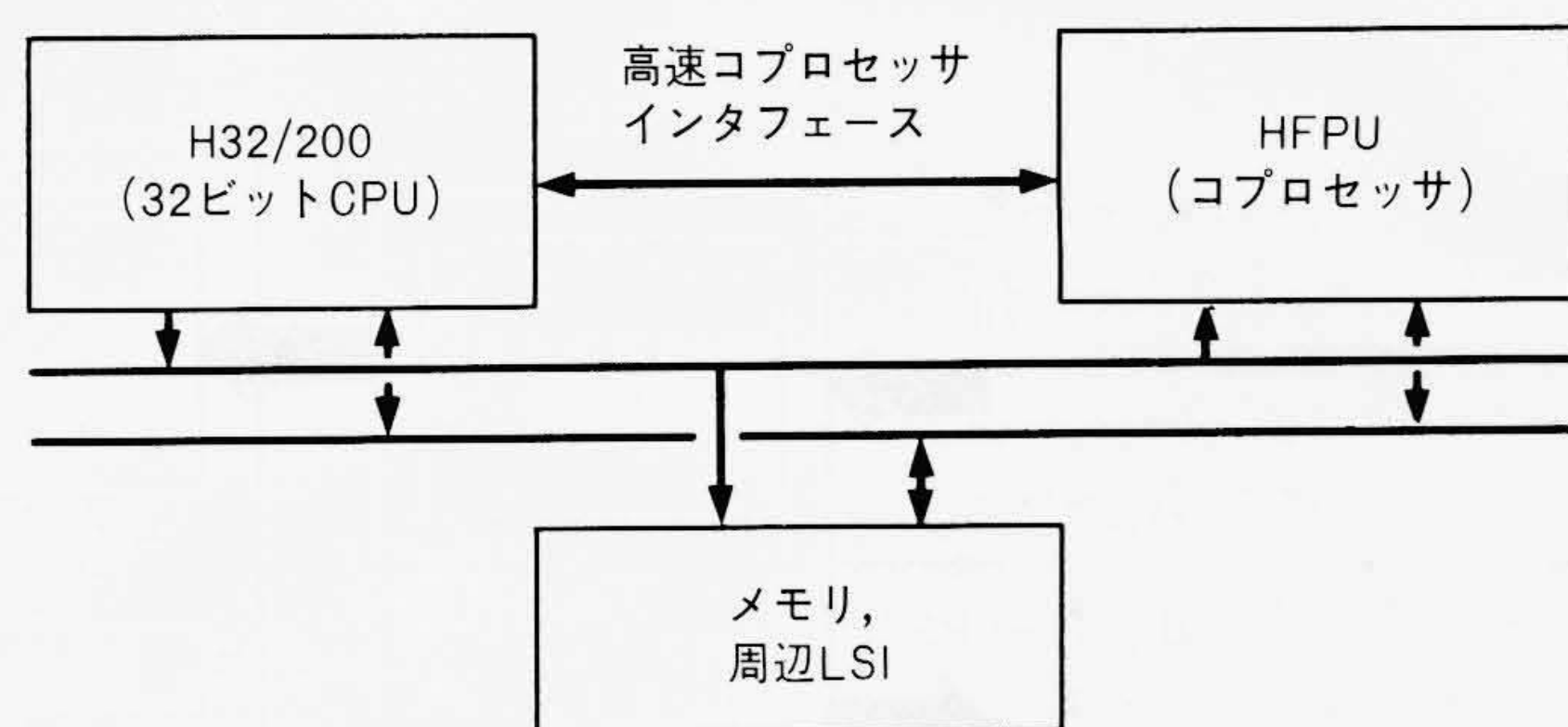
2 HFPUの概要

2.1 システム構成

HFPUを含む数値演算処理装置の構成を、単純化して図1に示す。HFPUは、ホストCPUであるH32/200と同格のプロセッサ、すなわちコプロセッサと呼ばれるものである。H32/200CPUとHFPUは一体となって、浮動小数点演算能力を持っている新しいCPUのようにふるまう。すなわち、HFPUはH32/200CPUのレジスタ及び命令を拡張する。

両プロセッサ間には、ハードウェアで制御される高速コプロセッサインタフェースが介在する。H32/200CPUは、メモリからFPU命令をフェッチすると、それを解読したのちHFPUに対するコマンドを作成し、それをFPUへ転送する。その後、H32/200CPUとHFPUは一体となって、FPU命令を実行する。

HFPUは、マイクロコンピュータシステムに強力な浮動小数点演算能力を付加することになる。なお、コプロセッサと



注: 略語説明 CPU (Central Processing Unit)
HFPU (High performance Floating-point Processing Unit)

図1 数値演算処理装置の構成 H32/200CPUとHFPUのチップセットにより、浮動小数点演算が可能な1個のCPUとして動作する。

周辺LSIを比較した場合、大きく異なる点は前者がコプロセッサ専用命令で駆動されるのに対して、後者がCPUのI/O(入出力)命令で制御されることである。またHFPUは、コプロセッサと周辺LSIのインタフェースをサポートしている。

2.2 HFPUの機能

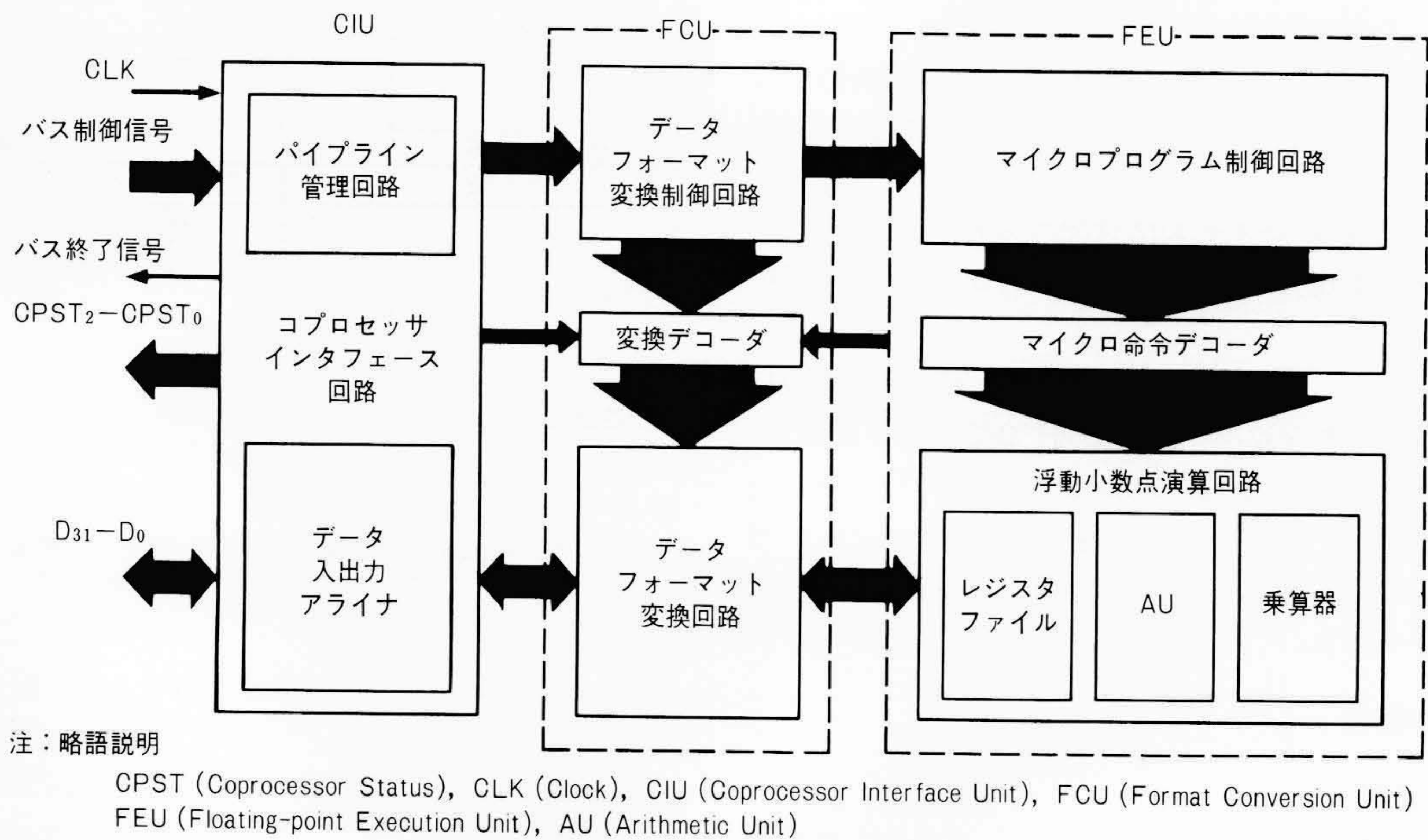
HFPUは、浮動小数点演算に関する規格であるIEEE規格(P754)をサポートしている。命令としては、表1に示すように、四則演算のほかに開平演算、三角関数、双曲線関数、指数・対数関数などの関数演算がある。また、HFPUの大きな特徴として、グラフィック処理のために、座標変換機能をサポートしている。そのため、HFPUには、積和演算、領域判定命令が用意されている。更に、プロセッサとして、一般的に必要なとする転送命令、プログラム制御命令がある。データフォーマットは、IEEE規格で定めている単精度、倍精度及び拡張倍精度をサポートすると同時に、IEEE規格で定められているその他の項目について準拠している。

* 日立製作所日立研究所 ** 日立製作所武蔵工場

表1 FPU命令の一覧 四則演算のほか、初等関数演算をもサポートし、豊富なFPU命令が用意されている。

No.	分類	ニモニック	演算処理	No.	分類	ニモニック	演算処理	No.	分類	ニモニック	演算処理
1	四則演算	FADD	加算	18	関数演算	FTAN	$\tan(x)$	35	判定	FTST	テスト
2		FSUB	減算	19		FASIN	$\sin^{-1}(x)$	36	分岐	FBCC	条件付き分岐
3		FMUL	乗算	20		FACOS	$\cos^{-1}(x)$	37	転送	FLD	二進浮動小数点
4		FDIV	除算	21		FATAN	$\tan^{-1}(x)$	38		FST	
5	他の一般演算	FREM	IEEE剰余算	22		FSINH	$\sinh(x)$	39		FLDI	符号付き整数
6		FMOD	剰余算	23		FCOSH	$\cosh(x)$	40		FSTI	
7		FSCALE	スケーリング	24		FTANH	$\tanh(x)$	41	特殊処理	FLDUI	符号なし整数
8		FABS	絶対値	25		FATANH	$\tanh^{-1}(x)$	42		FSTUI	
9		FNEG	符号反転	26		FEXP2	2^x	43		FLDC	制御レジスタ
10		FSQRT	平方根	27		FEXPE	e^x	44		FSTC	
11		FINT	整数値化	28		FEXP10	10^x	45	特殊処理	FMVR	定数ROM
12		FINTRZ	整数値化(切捨て)	29		FLOG2	$\log_2(x)$	46		FLDM	多重転送
13	関数演算	FEXTE	指数値抽出	30	座標変換演算	FLOGE	$\ln(x)$	47		FSTM	
14		FEXTM	仮数値抽出	31		FLOG10	$\log_{10}(x)$	48		FNOP	ノーオペレーション
15		FSIN	$\sin(x)$	32		FDOTPR	内積	49		FABT	中止
16		FCOS	$\cos(x)$	33	判定	FLIMIT	範囲内外判定	50		FSAVE	マシン状態退避
17		FSINCOS	$\sin(x), \cos(x)$ 同時	34		FCMP	比較	51		FREST	マシン状態復帰

注：略語説明 IEEE(米国電気電子学会), ROM(Read Only Memory)



注：略語説明 CPST (Coproprocessor Status), CLK (Clock), CIU (Coproprocessor Interface Unit), FCU (Format Conversion Unit) FEU (Floating-point Execution Unit), AU (Arithmetic Unit)

図2 ブロック構成 HFPUは、三つの処理ユニットに分割したブロック構成とした。

3 HFPUの動作

3.1 高速化の方針

FPUは、演算を高速に実行することが非常に重要である。浮動小数点演算を実行するためには、HFPUとH32/200 CPUは協調動作を行う。CPUは命令を取り込み、HFPUへコマンドの形式で転送する。次に、入力データをHFPUへ転送し、演算を実行する。ここで、HFPUでは演算を高速で実行するために、CPUとHFPUの間の転送でのオーバーヘッドの減少と、浮動小数点演算時間の短縮を図った。

3.2 内部ブロック構成

HFPUは、図2に示すようなブロック構成とした。浮動小数点演算は、大きく三つのユニットによって処理することが

できる。それらのユニットは、H32/200CPUとのCIU(コプロセッサインタフェース処理ユニット)、FCU(フォーマット変換処理ユニット)及びFEU(浮動小数点演算ユニット)である。CIUは、H32/200CPUとのインタフェース処理を行う。HFPUの状態を3本の専用信号(CPST₂~CPST₀)でCPUに知らせることで、CPUとHFPUの同期に必要なハンドシェイクのオーバーヘッドをなくした。また、HFPUではCPUをバスマスタにし、データをHFPUとメモリ間で直接転送する方式とし、データ転送の高速化を図った。FCUは、外部から入力された種々の精度のデータを内部形式のデータに変換するユニットである。このように、フォーマット変換後のデータをHFPUのレジスタに格納することによって、浮動小数点演算処理ユニットでのフォーマット変換

処理を削除した。

FEUは、フォーマット変換されたデータに対して、浮動小数点演算を行うユニットである。このユニットは、浮動小数点データ特有の指数部処理と仮数部処理を並列に実行できるように、各々処理を個別の演算器で処理する構成とし、演算の高速化を図った。更に、乗算器を内蔵し、乗算の高速化も図った。

HFPUでは、スループット向上のために、CIU、FCUとFEUを3段のパイプライン制御で動作させた。

3.3 高速コプロセッサインタフェース

HFPUはH32/200との高速コプロセッサインタフェースによって接続されている。HFPUがメモリのデータとレジスタのデータの浮動小数点加算を行った場合のインタフェースの動作を図3に示す。

HFPUとH32/200CPUには、コマンド及びデータを高速に転送するために、プロトコルが規定されている。H32/200CPUからバス制御信号(BAT₂~BAT₀)をHFPUへ出力する。HFPUは、その信号からコマンドとデータの区別を知ることができる。HFPUからコプロセッサ状態信号(CPST₂~CPST₀)をH32/200CPUへ出力する。H32/200CPUは、その信号からHFPUの状態を知ることができる。

プロトコルは、バス制御信号がコマンド転送を示した時点から始まる。HFPUは、バス上にあるコマンドを取り込み、コマンドの解読を行う。次に、H32/200CPUはメモリの入力

データをHFPUへ転送し、HFPUは入力データを取り込む。そのとき、HFPUはコマンドの解読結果、そのコマンドを受け付ける場合、コマンドの正常受付けの状態をコプロセッサ状態信号によってH32/200へ通知する。最後に、H32/200CPUは、例外処理の解析に必要な命令アドレスをHFPUに転送し、HFPUは浮動小数点の加算を実行する。このようなプロトコルによって、H32/200CPUとHFPUの間のコマンドとデータの転送が正確かつむだなく高速に実行される。

3.4 パイプライン制御

HFPUは、高速コプロセッサインタフェースによって、コマンドと入力データの転送時間を短縮した。更に、転送のオーバーヘッドを減少させるために、命令取込みとFPU転送を見掛け上なくすパイプライン制御を行った。パイプラインの段数は、三つの処理ユニットを効率よく動作させるために、図4に示すように3段とした。その動作は、コプロセッサインタフェース処理ユニットで、コマンド、入力データを取り込む。その入力データは、FCUによって内部の表現形式に変換される。変換終了後、FEUによって演算の実行を開始すると同時に次のコマンドを取り込む。

3.5 浮動小数点演算時間の短縮

IEEE規格の浮動小数点演算には、演算のほかにデータタイプによって例外の処理が定められている。IEEE規格の浮動小数点データタイプは、フォーマットの範囲で表現できる数である実数、零、無限大、数でない非数である。実数以外は例

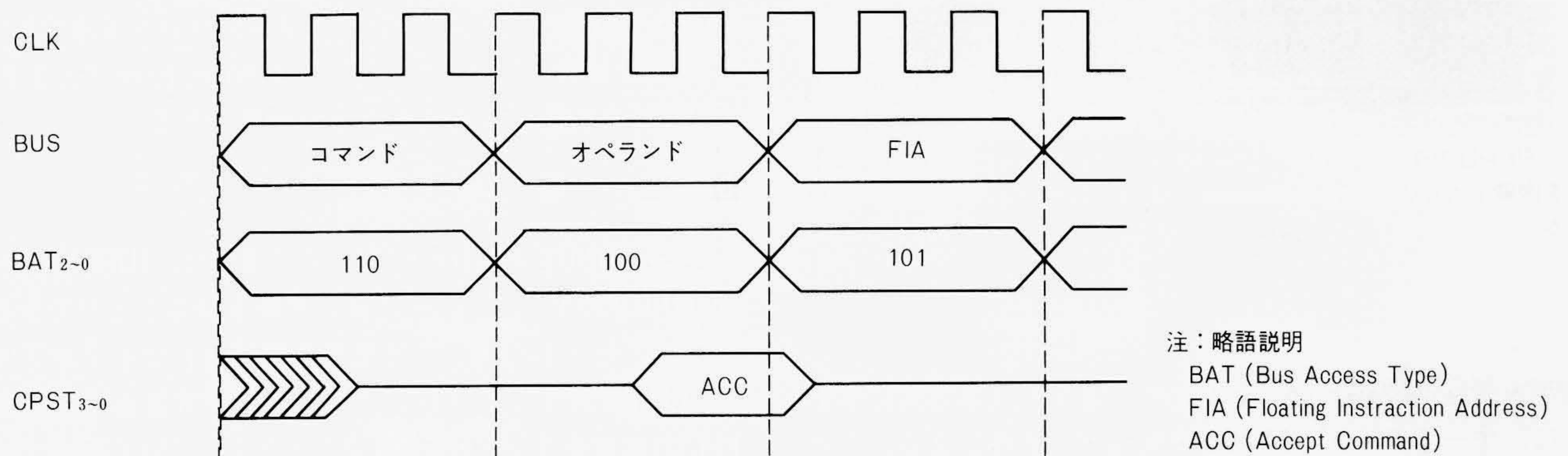


図3 バスサイクル HFPUに内蔵した高速コプロセッサインタフェースを示す。

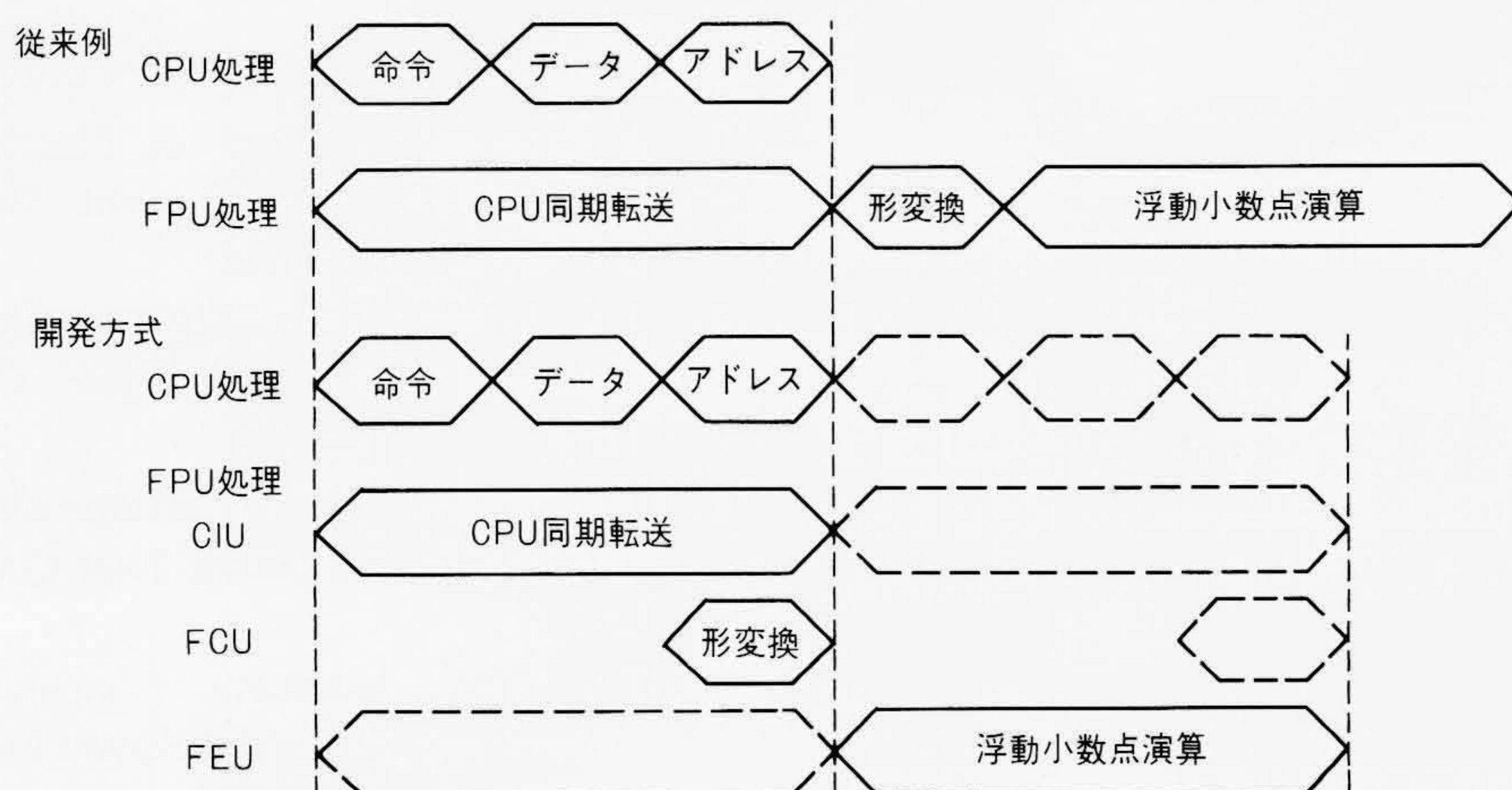


図4 3段パイプライン制御 三つの処理ユニットを、3段のパイプライン化制御にして高速化を図った。

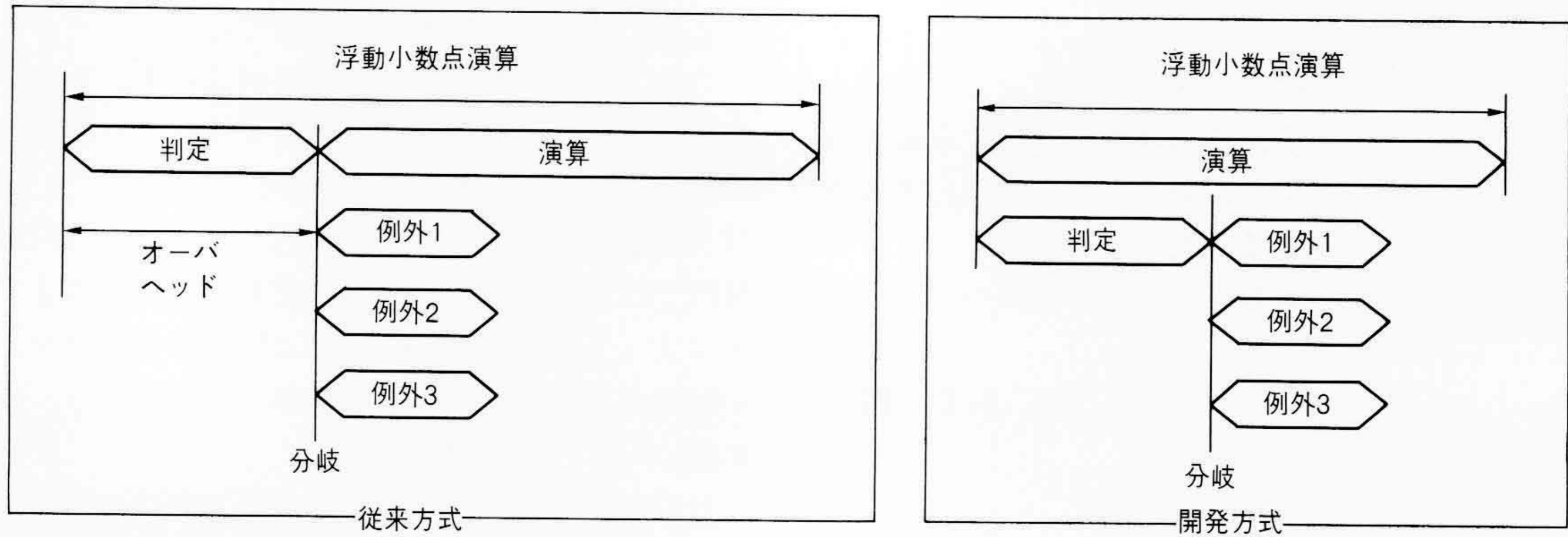


図5 高頻度データタイプ優先処理 FPUで、通常多く現れる実数データタイプの処理を優先とし、高速化を図った。

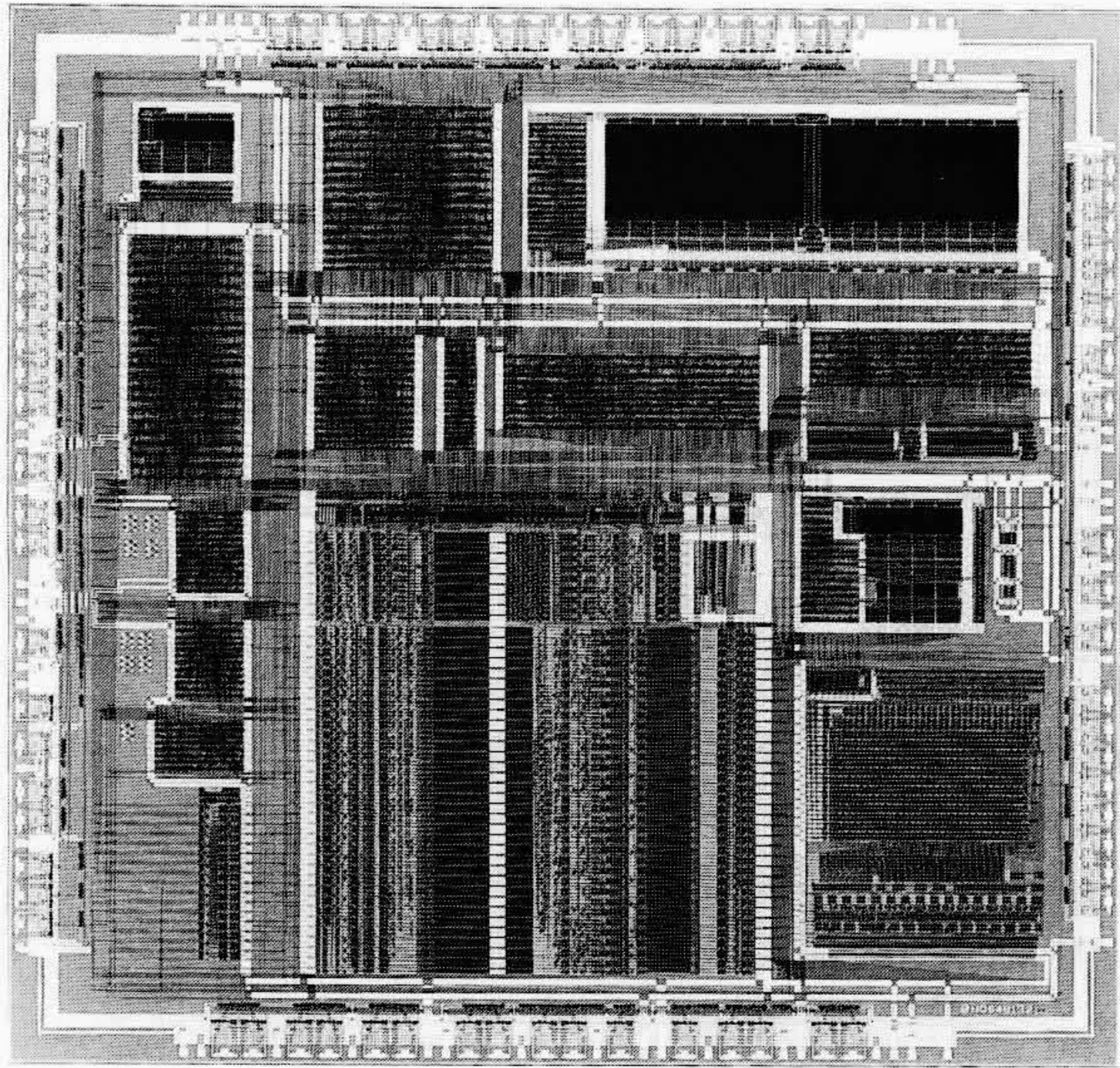


図6 HFPUのチップ写真 14.05 mm×13.37 mmに約57万トランジスタを内蔵したCMOS(Complementary Metal Oxide Semiconductor)LSIである。

外処理を行う必要がある。そのため、図5の従来方式に示すように、データタイプの判定と演算が直列処理となる。このデータタイプの判定がオーバーヘッドとして現れる。演算を高速にするためには、データタイプの判定と演算を並列処理するようにした。

除算及び開平方は、ニュートン・ラフソン法で高速に演算するハードウェアを内蔵した。更に、三角関数など初等関数の演算には、コーディック法を適用した新しいアルゴリズムを開発した。

HFPUのチップ写真を図6に示す。HFPUは、135ピンのピングリッドアレーのパッケージに収められたCMOS IC(Complementary Metal Oxide Semiconductor IC)によって構成され、その内容は総トランジスタ数約57万のLSIである。

4 性能

以上の高速化によりHFPUの性能を表2に示す。基本演算である四則演算は、20 MHz動作時、加減算0.5 μs、乗算0.45 μs、除算1.6 μsと高速である。また、科学技術計算に多く使用

表2 HFPUの性能 HFPUは、高い性能を得ている。

項目	製品	開発FPU (μs)
加 算		0.5
減 算		0.5
乗 算		0.45
除 算		1.6
SIN		6.5
クロック(MHz)		20

注：単精度の場合

される三角関数の演算は、サイン演算6.5 μsと高速であり、科学技術計算のベンチマークであるWhetstoneで、3.2～4.0 MWIPSの性能を達成することが可能である。

5 結 言

以上、FPUであるHFPUについて述べた。HFPUは、H32/200CPUとのチップセットで高い性能を得ることができる。このチップセットが、今後増加する32ビットマイクロプロセッサの応用分野で、高性能化に大きく寄与すると考えている。FPUの高性能化に対して、更に努力をしていく考えである。

参考文献

1) A Proposed Standard for Binary Floating-point Arithmetic : IEEE Floating-point Subcommittee Working Document, p.754(1987)
2) J.E. Volder : The CORDIC Trigonometric Computing Technique : IEEE Trans. Elec. Comp., EC-8, 330～334 (1959)
3) J.S. Walther : A Unified Algorithm for Elementary Function : AFIP 1971 Spring Joint Comput. Conf., 379～385 (1971)
4) T.C. Chen : Automatic Computation of Exponentials, Logarithms, Ratios and Square Roots : IBM Journal Res. and Dev., 380～388(1972-7)
5) C. Huntsman : The MC68881 Floating-point Coprocessor : IEEE MICRO Vol.3, No.6, 44～54(1983-12)