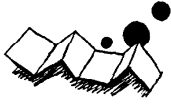


解説



モジュール型複合計算機 (ACE) の試み†

飯塚 肇** 藤井 狷介** 古谷 立美**

1. 設計思想

ACE (Adaptive Computer Experiment) はモジュール構造の複合計算機であって、名前の示す通り応用適応性を持つ計算機アーキテクチャを研究するための実験システムである。そのために ACE システムは次のような特徴を有している^{1),2)}。

(1) プロセッサ～メモリ～スイッチ (PMS) レベルでのモジュール構成を採用し、各モジュール間の通信には柔軟性、一般性の極めて高い特殊なバス (C-バス) を採用した。この C-バスはバス上で用いる情報のアドレス (グローバルアドレス) を各モジュールの内部アドレスと独立に取り、かつ、送信側モジュールに受信側モジュール番号を指定させるのではなく、受信側のモジュールがグローバルアドレスを認識する方式を採用している。この論理的に極めて自由度が高く応用に適した構造にモジュールを結合することができる。また、本質的に記憶モジュールと処理モジュールの区別がないので、完全分散型記憶構造* を簡単に実現することができる。

(2) プロセッサモジュールにダイナミックマイクロプログラミング方式を採用し、汎用エミュレーション指向のアーキテクチャとした。また、その中心要素としては LSI** 汎用計算素子 PULCE^{3),4)} を使用している。

(1) のモジュール構造は応用側の並列処理構造とハードウェアのそれを合せるには都合よいが、これは静的なもので、動的な適応性のためにはダイナミックマイクロプログラミングが最も有効である。そのため

にマイクロキャッシュを用いた強力なダイナミックマイクロプログラム機構が用意されている。

(3) ディスクリプタ方式をマイクロプログラムレベルで採用している。

プロセッサモジュールには 16 個のセグメントレジスタがあり、セグメント位置、長さ、データ長 (1, 2, 4, 8, 16, 32 ビットのいずれか、ハードウェアのデータアライメントで自動調整される。)、キャッシュバリティ、情報の種類等を指定できる。

(4) マイクロプログラム及びデータ用に独立したキャッシュメモリを用意し、必要な情報が自動的に高速記憶に格納される。

特に、データキャッシュにおいては 4 レベルのキャッシュバリティ制御* を行っている。

以上のように ACE では応用に適合した密結合分散処理ができるように工夫されているが、あくまでも実験システムであるため、多少機能が盛り沢山になっており、現実のシステムとして性能/コスト比は必ずしも最適とはいえない面もある。

2. システム構成

2.1 ハードウェア構成

現在の ACE システムは図-1 に示すとおり、3 台のプロセッサモジュール (PR)、2 台のメモリモジュール (MM)、同期モジュール (SM)⁵⁾、及び入出力用ミニコンピュータ (IOP) が、2 本の C-バスに結合されている。

プロセッサモジュールは、32 ビット幅の強力なマイクロ命令を持つ処理ユニットを中心に 256 語 (32 ビット/語) のマイクロプログラムキャッシュ、2k バイトの一般データ用キャッシュなどから成り立っている。プロセッサのデータ処理幅は 16 ビットで、内部

† An Experiment of the Modular Multiprocessor (ACE) and Its Preliminary Result by Hajime IZUKA, Kensuke FUJII and Tatsumi FURUYA (Computer Division, Electrotechnical Laboratory).

** 電子技術総合研究所電子計算機部

* 一般のマルチプロセッサでは主記憶が共用されるので記憶の観点から見ると集中型である。

** LSI の PULCE も完成しているが、現在の ACE システムのプロセッサモジュールは製作時期の関係で TTL で実現されている。

* キャッシュバリティ 0 は一際キャッシュを用いない場合、1 はストア時に主記憶とキャッシュの両方を更新する。また、2 のときは通常はキャッシュのみを更新し、主記憶の更新はキャッシュロックの割当変更時に行う。更に 3 の場合はキャッシュのみしか変更せず、主記憶の更新を行わない。

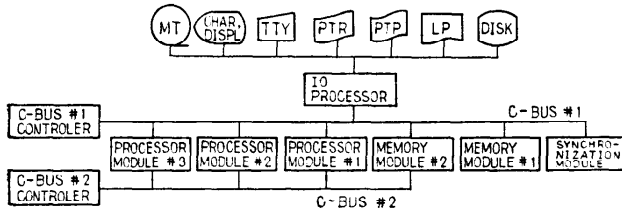


図-1 ACE システム (1978年11月)

に 16 個のレジスタと、32 語のレジスタファイルを持ちマイクロ命令を 300 ns (設計値 200 ns) で実行する。

メモリモジュールの記憶容量は 2 台共 64k バイトであるが、仕様は若干異なる。メモリモジュール 1 は 4 ウェイにインタリーブされており、高速度のブロック転送が可能である。

モジュール間通信は C-バスを通して行われ、バスの幅は 32 ビットである。目標転送速度は 200 ns であるが、現在 7 台という多数のモジュールが結合されているためバス 1 は 500 ns バス 2 は 400 ns で動作させている。バスの通信は C-バスコントローラ (CC) により集中的に制御されている。

同期モジュールは Test & Set 機能及びリクエストに対する待ち行列を管理する機能を有し、セマホアを容易にインプリメントできる。

2.2 ソフトウェア構成

ACE のソフトウェアシステムは汎用マイクロアセンブラ⁶⁾などのユーティリティプログラム群と基本オペレーティングシステム (OSK) より成っている。OSK⁷⁾では、各プロセッサモジュールへ OS のインスタンスを分散させる設計方針を取ったため、OS 自体が同時に動作できる。

OSK は表-1 に示すように、各 PR 上で動作する部分 (OSK-P) と IOP 上で動作する部分 (OSK-N) とで構成される。この中、プロセス (タスク) の制御、

表-1 OSK の構成

OSK-P	1) プリミティブ命令の受付 (入出力命令の受付も含む)。	<ul style="list-style-type: none"> ・マイクロ命令ステップ数 約 3 kW (32 bits/W) ・共通データベース 約 5 kW (32 bits/W)
	2) プロセッサ間通信。	
	3) プロセス間通信。	
	4) プロセススイッチング (スケジューリング)。	
	5) セマホアの管理。	
	6) ハードウェアスタックの管理。	
OSK-N	1) 入出力の管理 (RDOS 入出力コマンドへの変換、状態情報の転送など)。	<ul style="list-style-type: none"> ・命令ステップ数 約 5 kW (16 bits/W)
	2) マイクロプログラムのロード、初期設定、起動など。	

表-2 プリミティブ命令

CREATE, DELETE	プロセスの生成と消滅
START, STOP	プロセスの起動と停止
SEND, RECEIVE	メッセージの送信と受信
P-OPERATION V-OPERATION	セマホアへの P/V 操作
GSEM, PSEM	ユーザセマホアの get/put
TERM	プログラムの終了
NOVA	入出力要求

資源管理、スケジューリングなどを行う OSK-P の部分が実行時に各 PR へ分散され、並列に動作する。分散された各 OSK-P は、メモリモジュール上に存在する制御用の共通データベースを介してマルチ (マイクロ) プログラムの管理を行う。つまり、ACE ではマイクロプログラムを ROM に固定せず MM へ動的にロードすることが可能なので、表-2 に示すような従来のマルチプログラム用オペレーティングシステムと似た機能を持たせることが可能である。

OSK-P の主機能は C-バスによって与えられるグローバルアドレス機構、高度な PR 間通信機構、PR におけるセグメントアドレス機構などハードウェア基本機能を用いることにより、共有空間の設定、個有空間の分離、排他および同期制御用のロックセルの授受、プロセス起動停止用プロトコルの授受、入出力要求の受け付けなど見かけ上の機能拡張を達成している。その他に、プロセス間通信、プロセススイッチング、ハードウェアスタックの管理、情報授受も可能なセマホアの管理、異なったアーキテクチャを有する PR の OSK に対するサポート、論理的システム構成を分割してサブシステムを構成するためのサポートなどを行う。

OSK-N は、IOP として用いているミニコンピュータ Nova の RDOS マルチタスクの管理下において、主に入出力の制御、入出力命令の変換を行う。ACE では PR 自身に入出力機器を接続しないため、入出力はすべて IOP に依頼せねばならない。OSK-N は、OSK-P を介して待行列に入った入出力要求を取り出し、RDOS の入出力コマンドへ変換し、入出力を実行し、状態情報を OSK-P へ返す。

3. 動作と応用

ACE システムは現在なお開発途上で、ソフトウェアの作成、データの収集が続けられている。本節では高級言語マシンとエミュレータに関する応用及びその 2, 3 の測定結果を示す。

3.1 Concurrent PASCAL マシン⁸⁾

ACE のダイナミックマイクロプログラミング機能とマルチプロセッサ構造を利用して Concurrent Pascal (CP) で記述された並列プロセスを複数のプロセッサに割当てて実行させる CP マシンを実現した。この CP マシンは、P.B. Hansen の CP 用仮想マシンをマルチプロセッサ用に機能拡張したものである。

CP をマルチプロセッサ上で動かす利点は、プロセスの並列実行とプロセス切換オーバーヘッドの減少であり、並列実行を妨げる要因は共有ルーチンでの待ちである。図-2 は2プロセスから成る実験プログラムを1台と2台のプロセッサで走らせた時の性能を測定したものである。効率にはプロセッサへの負荷配分によって左右されるが、これを適当に行えば大幅な性能向上が期待できることがわかる。

表-3 はシステムの動作特性を調べるために、ある Pascal プログラムの処理時間をいろいろな場合について求めたものである。列はバス速度を変えた場合、行はプログラムを1台のプロセッサで走らせた時と、プログラム、マイクロプログラム等を共有して2台のプロセッサで同じプログラムを同時に走らせた時の処理時間である。この例はマイクロプログラムキャッシュのヒット率が低く、バスの利用率が高い場合である。バス速度及びメモリ、バス共有のためのコンフリクトの影響は約 10~20% であった。

まだデータ量が不足で、確たることは言いえないが、ACE の構造と CP の性質のマッチングはよく、分散処理型マシンの応用としては向いたものと思われる。

3.2 Nova エミュレータ

ACE システムの試験、性能評価等を目的として、IOP として使用しているミニコンピュータ Nova の

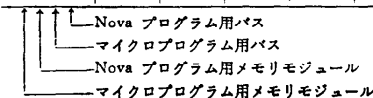
エミュレータを作成して使用している。プログラムステップ数は 308 (速度向上のための無効命令 70 を含む) とコンパクトであるが、実行速度は直接実行の場合の約 1/10 になる。これは Nova の命令自身が多フィールドでデコードに時間がかかるためであってやむをえない。ACE のプロセッサモジュールのようなユニバーサルホストプロセッサは一つの命令の機能が小さいミニコンピュータのエミュレーション等にはあまり向かないことがわかる。

さて、このエミュレータを用いて、メモリやバスの共用によるコンフリクト、キャッシュの効果等を測定中であるが、一例を次に示す。表-4 はキャッシュバリティの効果とメモリモジュールやバスの速度の関係を示したものである。MM 1 と C-バス 1 を用いたときの値を 100 として正規化した実行時間が示されているが、これから次のようなことがわかる。

- * キャッシュバリティ 1 以上はほとんど変りない。
- * キャッシュは 20~40% 速度向上にきく。
- * 2つのメモリモジュールとバスはそれぞれ #2 の方が高速で、一番速い組合せと遅い組合せではアクセス時間が3倍程度異なるが、キャッシュを用いるとそれによる実行時間の差は1%程度にしかならない。
- * 2台のプロセッサモジュールでプログラムを共用した場合、キャッシュを用いないと 0.6~3.5% の速度劣化が見られるが、キャッシュを用いると有意差はでない(従って表には省略してある)。

表-4 キャッシュ、メモリ、バスの評価データ例

使用モジュール番号	キャッシュバリティ				
	単一プロセッサ動作				2プロセッサ
	0	1	2	3	
1 1 1 1	146.2	[100]	99.9	99.9	151.3 (3.5%)
1 2 1 1	133.5	100.0	99.9	99.8	135.6 (1.6%)
1 2 1 2	119.2	99.9	99.9	99.9	119.9 (0.6%)
2 1 1 1	145.3	99.0	99.0	99.0	149.2 (2.7%)
2 1 2 1	145.1	98.9	98.9	98.8	148.4 (2.3%)
2 2 1 1	132.7	99.1	99.0	99.1	134.9 (1.6%)
2 2 1 2	118.2	99.2	99.0	99.0	119.2 (0.8%)
2 2 2 1	132.4	98.9	98.8	98.9	134.5 (1.6%)
2 2 2 2	118.1	98.9	98.8	98.8	119.6 (0.8%)



注) 1. 実行速度はバス、メモリともモジュール #1 を用い、キャッシュバリティを1とした時の値を100として正規化してある。
2. 2プロセッサの場合の内は1プロセッサの場合に対する増加割合を示している。

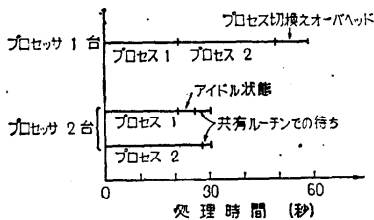


図-2 2つのプロセスより成るプログラムの処理時間

表-3 実験プログラムの処理時間

	バス速度	
	500 ns	1 μs
1つのプログラムが走った時	45 ms	53 ms
2つのプログラムが走った時	52 ms	76 ms

表-5 リソースの共用効果データ例

M M	C バス	共用モード		実行速度	速度低下度
		MM	バス		
1	1	0	0	146.2	—
		0	1	147.7	1.0%
		1	0	n.a.	n.a.
		1	1	151.3	3.5%
2	1	0	0	132.7	—
		0	1	143.7	8.3%
		1	0	133.2	1.0%
		1	1	134.9	1.7%
2	2	0	0	118.1	—
		0	1	n.a.	n.a.
		1	0	130.5	10.5%
		1	1	119.0	0.8%

- 注) ① 共用モードの0は共用なしを、1は共用を示す。
 ② 実行速度は MM、C-バスともに #1 を使用し、単一プロセッサ実行時を 100 とした相対値。
 ③ 速度低下度は同一リソース組合せの単一プロセッサ動作時に対するもの。
 ④ n.a. は当該組合せがないことを示す。

これらの事実からマルチプロセッサにおけるキャッシュの有効性がよくわかる。当然の結果ともいえるが、キャッシュビリティは2レベルで十分で、キャッシュの量を多くする方が効果的であろう。

一方、表-5 はキャッシュを用いない場合のバス及び MM の共用効果を示したものである。バス1とメモリ1の組合せはブロック転送におけるバス及びメモリ専有時間が長いので、これが他のプロセッサによって使用されると影響が大きい他は2台ぐらいではまだゆとりがある。

4. 総 括

以上、応用環境に適合した分散処理を意図して設計されたモジュール型複合計算機システム ACE について述べた。この ACE システムのハードウェアは既述のように3プロセッサモジュールのものが一応稼働しているが、ソフトウェアはなお開発中であり、測定評価データもまだ少ない。したがって結論めいたことを述べるには時期尚早であるが、最後にこれまでの経験から気をついたことを思いつくままにあげておく。

- 実験システムという性格もあって、やや盛り沢山の機能を組込んだが、それらの内、受信側認識による通信方式、キャッシュ等は特に有効であった。それに要するハードウェアはかなり多いが、柔軟性のあるモジュール間通信システムは、この種のシステムにとって必須であろう。
- 逆に、あまり効果のなかったものには、4レベルの

キャッシュビリティ (もちろん2レベルはいる)、複雑なキャッシュブロック割付けアルゴリズム (但し、数値的評価はすんでいない) 等がある。

- 一方、データ長調整回路、グローバルリティ制御、比較的大容量のローカル一時記憶、マイクロプログラムレベルの強力な割込機構等は有効性はかなりありそうだが、現在のところその能力を十分使用できていない。
- ACE におけるこれまでの最大の問題点は実装にある。実験システムとしての不完全な実装のために設計値よりかなり速度を落さなければならなくなっているし、信頼性にも大きな問題を生じた。これは実験システムという性格からある程度やむをえないことでもあるが、高信頼化への対策が不十分であったのは反省点である。
- この実験には多数のプロセッサを応用環境に合わせ接続した並列プロセッサシステムと、その構成プロセッサモジュールとして強力なユニバーサルホストプロセッサを開発することの2つの大きな目標があった。しかし、結果的にこれは多少よくばりすぎであって、後者については一応の目標を達成できたが、プロセッサモジュールの機能が大きくなりすぎたために多数のプロセッサモジュールを用いた並列系の実現はコスト的に、また実装上不可能になってしまった。
- オペレーティングシステムは未完成で、まだ評価できないが、マイクロプログラムを開放するという方針と従来型設計との間に若干のいちがいがあったように感じている。新しい設計理念が必要かも知れない。

以上、ACE の実験で気づいたことを定性的に述べたが、更に、定性的にこの種の分散処理型計算機について述べさせてもらえば、ACE システムのようなユニバーサルホストプロセッサを比較的少数 (10 台以下)、柔軟性の大きい通信方式を持つ結合系で接続し、マイクロプログラムによる専用化機能を利用して、タスク (プロセス) レベルでの分散処理を行う計算機の実現性、実用性はかなり大きいものと思われる。今後そのために重要なことは、ACE の PR のような強力なプロセッサモジュールをコンパクトに実装できるような LSI プロセッサの開発、強力に柔軟性の高くあまり複雑でないモジュール間標準通信方式とそのため LSI、マルチユニバーサルホストプロセッサを効率よく管理できる OS の開発等であろう。

一方、何十ものプロセッサモジュールを結合してもっと高度の（低レベルの）並列処理を実行するシステムについては ACE 程度の考え方では実現困難であろう。そのためには、プログラムの記述方法等も一体として考えた更に高度のアーキテクチャ的ブレークスルーが必要であると思われる。

参考文献

- 1) 飯塚 肇：適応構造計算機に関する研究，電総研研究報告第 767 号（1976 年 11 月）。
- 2) Iizuka H., et al.: ACE—A New Modular Computer Architecture, Proc. 2nd US-J Comp. Conf. pp. 36-41 (1973 年 8 月)。
- 3) 飯塚, 古谷：マイクロプロセッサアーキテクチャの一設計，電子通信学会論文誌，Vol. 59 D, No. 3, pp. 188-195 (1976 年 3 月)。
- 4) Iizuka, H., et al.: Development of a High-performance Universal Computing Element—PULCE, Proc. NCC, pp. 1255-1264 (1978 年 6 月)。
- 5) 古谷, 飯塚, 大表：マルチプロセッサシステム用相互排斥モジュールの設計，情報処理，Vol. 19, No. 6, pp. 609-611 (1977 年 6 月)。
- 6) 藤井, 飯塚：機械適合性を有するマイクロセンプラ，情報処理，Vol. 18, No. 9, pp. 906-912 (1977 年 9 月)。
- 7) 藤井, 飯塚, 古谷：ACE 1.1 システムとその基本 OS，電子通信学会計算機研究会資料 EC 76-74, pp. 25-34 (1977 年 1 月)。
- 8) 古谷：マルチプロセッサシステム (ACE) における Concurrent Pascal マシン，情報処理学会計算機アーキテクチャ研究会資料 33-1, 電子通信学会電子計算機研究会資料 EC78-30, 1978/11.
(昭和 53 年 12 月 15 日受付)