

# 大形プロセッサの技術動向

## Technological Trends of Large Scale General Purpose Processors

大形プロセッサ、いわゆるメインフレームはシステムの分散化、統合化の潮流の中で大規模データベースを扱うサーバとしての役割が重要になってきている。このような背景から、大容量のデータベースを高速に処理するためのシステム性能が要求されている。本論文では、このシステム性能を向上させるための基本的な課題である命令処理性能向上と記憶装置の大容量化について概観する。さらに、将来の高性能化技術としては、プロセッサを大幅に増加させた多数台マルチプロセッサ方式が中心になると予想される。

和田健一\* *Ken'ichi Wada*  
 坂東忠秋\*\* *Tadaaki Bandō*  
 若井勝郎\*\*\* *Katsurō Wakai*

### 1 緒言

大形プロセッサ、いわゆるメインフレーム市場もほぼ $\frac{1}{4}$ 世紀を経て、変化の兆しがみられている。すなわち、高速・高機能のネットワーク技術の進歩、ULSI化の急速な進展に支えられたパーソナルコンピュータ、ワークステーションの普及と高性能化がもたらしたシステムの分散化、統合化である。1990年代の情報システムはこれらの潮流がいつそう進み、また企業間のシステム融合が活発となり、異機種システム間の結合も多くなると予想される。

このような時代の大型プロセッサの役割は以下に集約されつつある。

- (1) 業務システムの高速度演算処理(コンピューティング)
- (2) 大規模データベースの高速度アクセス(データベースサーバ)
- (3) 全システム構成要素の運用管理(マネジメント)

大形プロセッサはこのような時代の情報システムの中核として期待され、これに対応した高性能・高信頼のプロセッサ技術が必要とされている。

本論文では、プロセッサの高性能化技術に焦点を絞り、この高速性がどのように実現されているかについて述べるとともに、主としてハードウェアの視点からこれまでのプロセッサの発展過程を分析し、今後のプロセッサの発展方向を展望する。

### 2 大形プロセッサの高速化の課題

プロセッサの性能を示す尺度として広く用いられているMIPS (Million Instructions Per Second)は、1秒間に実行される命令数を100万命令単位に示したものであり、下記で示される。

$$\text{MIPS (M命令/秒)} = 1 / (T_{MC} \times N_{IE})$$

ここに  $T_{MC}$ : マシンサイクル時間( $\mu\text{s}$ /サイクル)

$N_{IE}$ : 平均命令実行サイクル数(サイクル/命令)

プロセッサの処理能力を決める第一の要因はMIPSである。MIPSを向上させるには以下の課題がある(表1参照)。

- (1) マシンサイクル時間 $T_{MC}$ の短縮
  - (a) 高速論理LSIによる論理1段当たりの時間の短縮
  - (b) 高速記憶LSIによる記憶装置アクセス時間の短縮
  - (c) 高密度実装技術による信号伝搬の遅延時間の短縮
- (2) 平均命令実行サイクル数 $N_{IE}$ の低減
  - (a) 各ステップの処理を時間的に重複させるパイプライン制御方式による実効 $N_{IE}$ の低減
  - (b) バッファ記憶制御方式による記憶装置参照サイクル数

表1 MIPSとMバイト向上に重要な役割を演ずる技術 計算能力を決める主な要因は、命令実行速度および記憶装置容量であり、それらに必要な技術を表に示す。

計算能力を決める要因		必要技術
命令実行速度 MIPS $\text{MIPS} = \frac{1}{T_{MC} \times N_{IE}}$	マシンサイクル時間 $T_{MC}$ の短縮 素子実装技術	高速論理素子技術 高速記憶素子技術 高密度実装技術
	命令実行サイクル数 $N_{IE}$ の削減 方式論理技術	パイプライン制御方式 バッファ記憶制御方式 ベクトル演算方式
記憶装置容量 (Mバイト)	記憶装置技術	大容量半導体記憶素子 高密度磁気ディスク技術

注: 略語説明 MIPS (Million Instructions Per Second)

\* 日立製作所 中央研究所 \*\* 日立製作所 中央研究所 工学博士 \*\*\* 日立製作所 神奈川工場

の低減

(c) ベクトル演算方式による $N_{IE}$ の低減

命令処理装置の性能が向上するにつれ、プロセッサシステムに入力されるジョブ(プログラム)の規模が大きくなる。プログラムの複雑化による命令語の増加だけでなく、データも巨大化していく。また、TSSなどの普及によって同時に使用するユーザーの数が著しく増加し、並列に動作するプログラムの本数が著しく増加したので、記憶装置への容量の要求が強まっている。記憶装置容量が不足してくると、命令処理装置の性能に見合った十分な量の仕事を記憶装置上に準備できない。したがって、記憶装置容量が処理能力を決める第二の要因であり、この課題に対しては以下の技術が必要である(表1参照)。

(3) 記憶装置容量の増大

- (a) 大容量半導体記憶素子による主記憶装置容量の拡大
- (b) 磁気ディスクの速度と容量の拡大

### 3 大形プロセッサの発展過程

これまでの大形プロセッサの発展過程を、2章に示した高速化の課題の観点から分析する。

#### 3.1 マシンサイクル時間の短縮<sup>2)</sup>

(1) 高速論理LSIの性能推移(図1参照)

高速性と集積度向上の両方を追求したLSIが開発されている。特に集積度の進歩が顕著であり、これより、LSI間を渡る信号の回数が減ることによって信号遅延時間を短縮している。

(2) 高速記憶LSIの性能推移(図2参照)

命令処理装置内の制御記憶装置用、バッファ記憶装置用として高速記憶LSIは必要不可欠である。命令処理装置のマシンサイクル時間の短縮に見合ったアクセス時間の高速化が達成されていることがわかり、今後もこの動向は続く。

(3) 実装密度の推移(図3参照)

LSIの外での信号遅延時間を短縮するため、基板に搭載するLSIの数を増加させる技術が課題となっている。図3に示すように実装密度の向上が図られていることがわかる。

#### 3.2 平均命令実行サイクル数の低減<sup>1),3)</sup>

(1) パイプライン制御方式の発展(図4参照)

図4(a)の方式では、一つの命令の実行が終了して初めて次の命令の読出しが開始される。一方、パイプライン制御方式が発展すると、同図(b)に示すように命令実行の各ステージが重複して実行される。各ステージの実行時間が同一で、それがマシンサイクルに等しいとすると、理想的には1マシンサイクルごとに演算結果が得られる。すなわち、各ステージの実行時間が同一で、命令やデータが定常的に供給され、各ステージで遊ぶ時間がない場合にだけ最大性能を発揮する。しかし、実際には演算装置に命令やデータがそろっていないことによる演算装置の遊び時間が比較的大きく、パイプライン

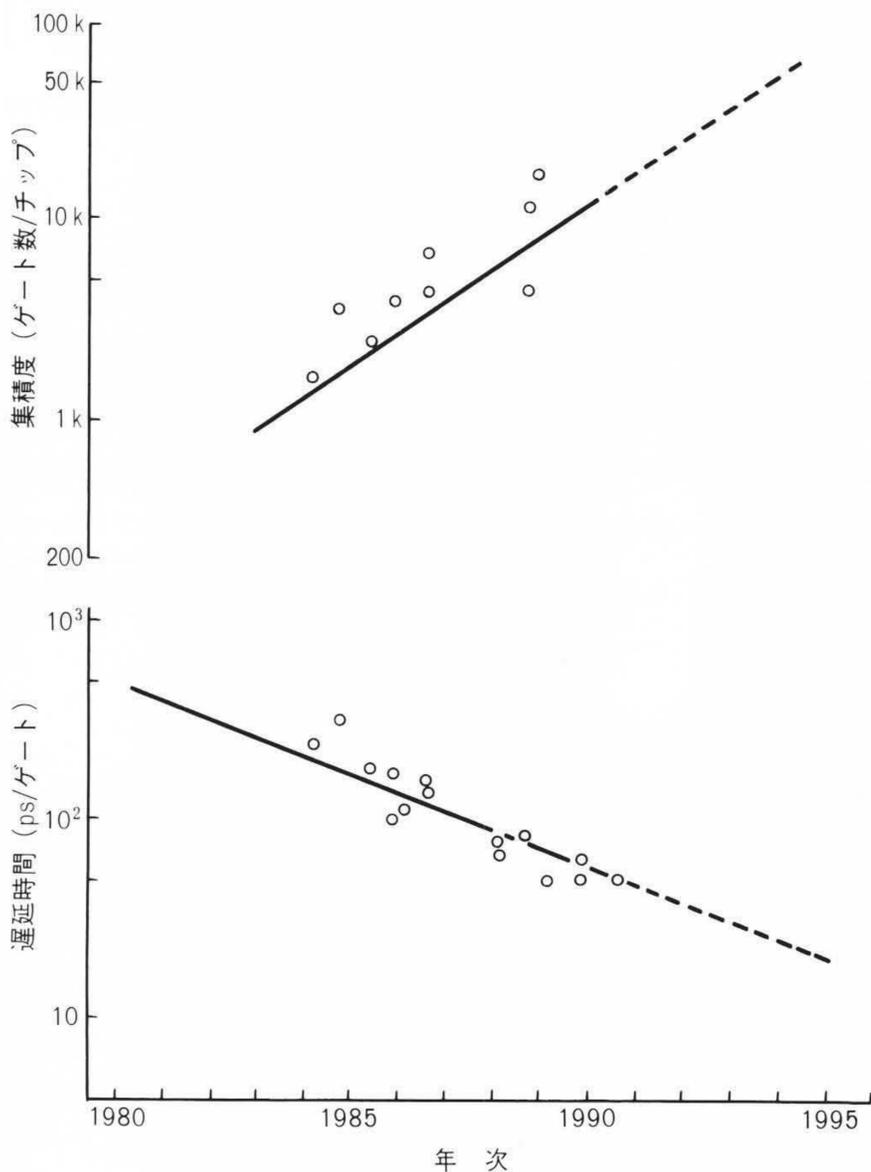


図1 高速論理LSIの性能推移 信号遅延時間と集積度の動向を図に示す。

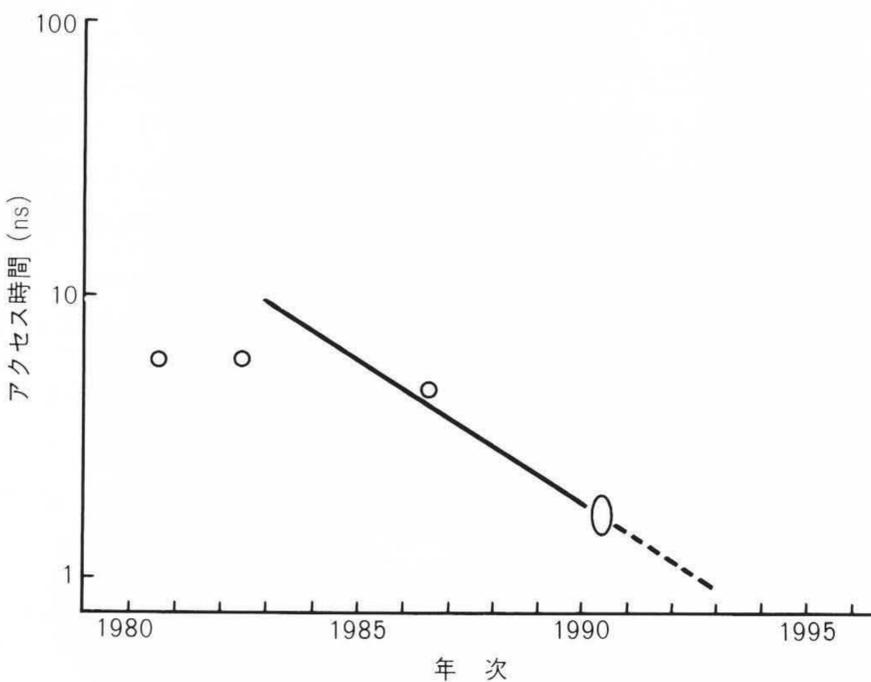


図2 高速記憶LSIの性能推移 アクセス時間の動向を図に示す。

が乱れる。この乱れを極力抑える方式の改善が課題であった。しかし、パイプライン制御方式のこれ以上の性能の改善は限界に達しつつあり、同図(c)に示すように、同一マシンサイクルに複数の命令を実行する複数命令並列制御方式が検討

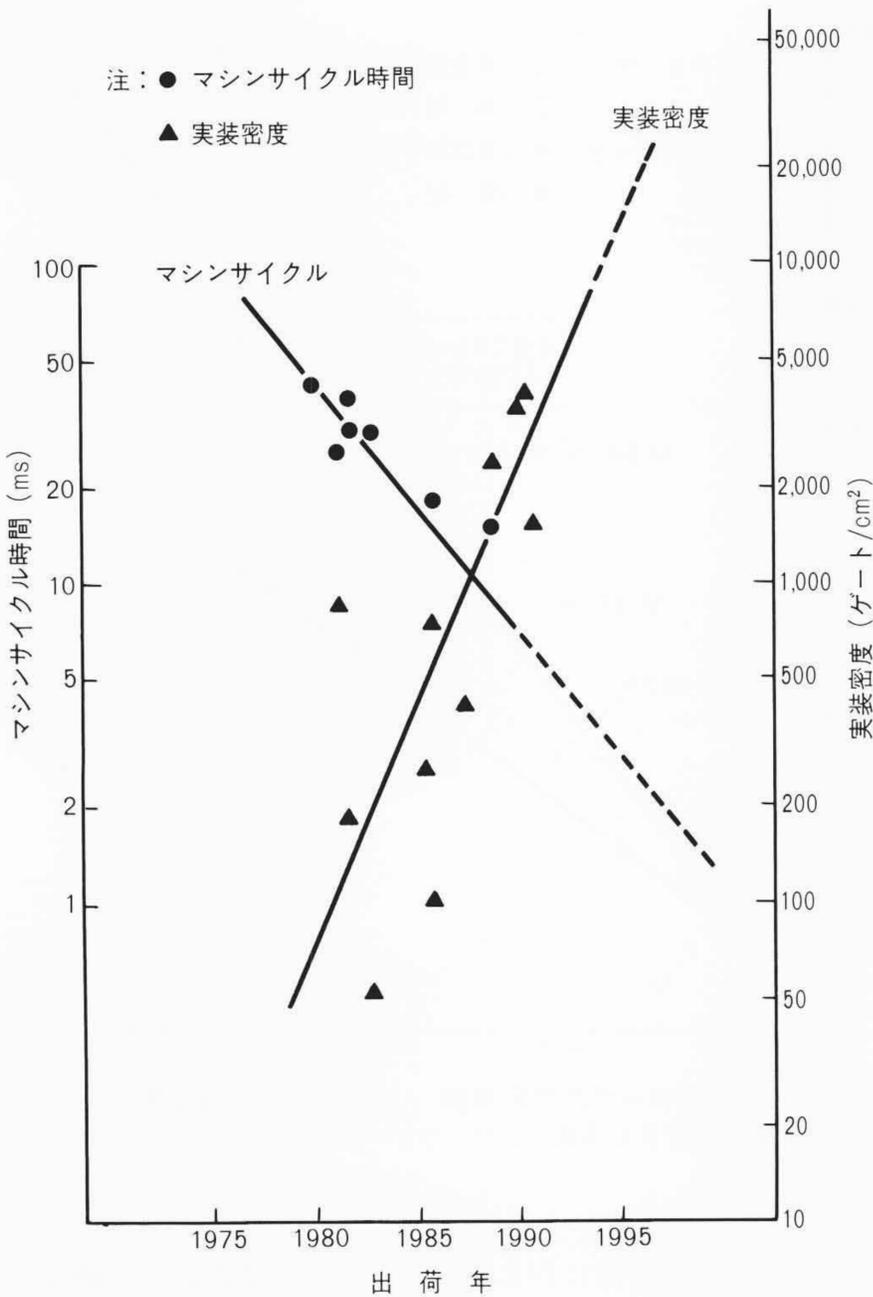


図3 実装密度の推移 プロセッサのマシンサイクル時間と、実装密度の動向を図に示す。

されている。このように複数の命令を並列して実行するには、各命令のデータ依存関係の高速解析と、その実行制御が課題である。

(2) バッファ記憶制御方式

命令処理装置は命令とデータを記憶装置から読み出す。しかし、命令処理装置の高速化に伴い、これらの情報を毎回記憶装置から読み出したのではデータの待ち時間が大きく、いくらパイプライン制御方式を高度にしても命令実行サイクル数 $N_{IE}$ を低減するのは難しいのがわかってきた。この間隙(げき)を埋めるために図5に示すように命令処理装置内に高速のバッファ記憶装置を装備し、頻繁に参照される情報をここに保存することにより、実効的に命令語およびデータのアクセス時間を短縮する目的で考案された。この方式を採用した場合の命令実行サイクル数 $N_{IE}$ は近似的に次式で表される。

$$N_{IE} = N'_{IE} + m \cdot N_{BF}$$

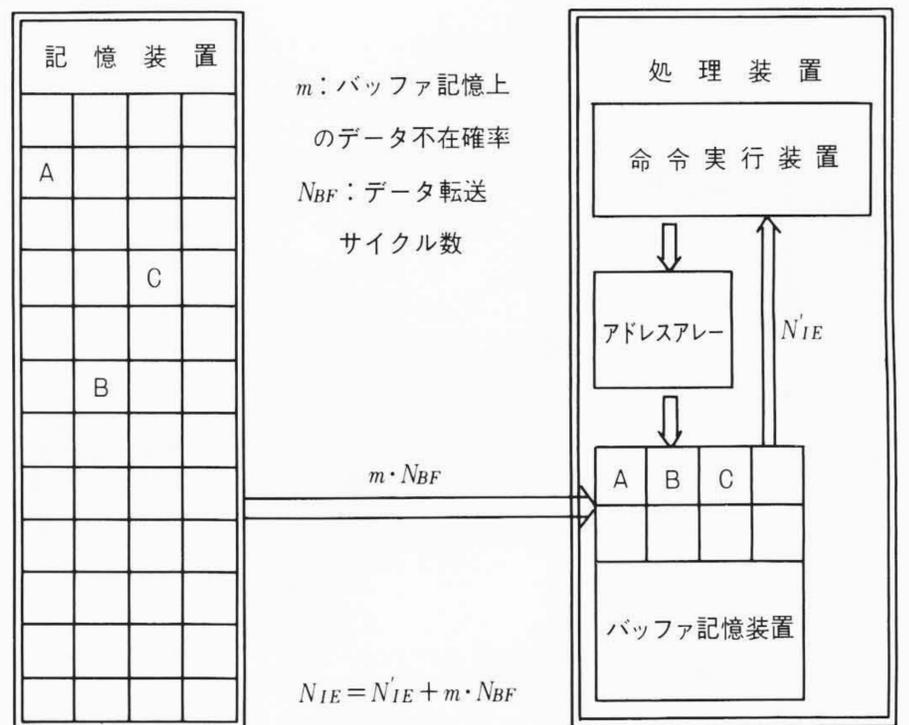
- ここに  $N_{IE}$ : すべての命令、データがバッファ記憶装置にある場合の命令実行サイクル数
- $m$ : 命令、データがバッファ記憶装置にない確率
- $N_{BF}$ : バッファ記憶装置にない場合、それを含むブロックを記憶装置からバッファ記憶装置に転送するのに要するサイクル数

バッファ記憶制御方式が十分効果をあげるには、この $m$ と $N_{BF}$ を小さく抑える必要がある。ところが、命令処理装置の高速化に伴い、記憶装置、バッファ記憶装置の2種類の記憶装置を使う2階層記憶制御方式では、十分小さな $m$ と $N_{BF}$ を実現できないことが明らかになってきた。



注: 略語説明  
 IF (命令読出し), D (命令解読), A (オペランドアドレス変換)  
 F (オペランド読出し), E (演算実行), S (結果格納)

図4 パイプライン制御方式の発展 パイプライン制御方式は、複数命令並列制御方式へと発展してきている。



$N_{IE}$ : 命令実行サイクル数  
 $N'_{IE}$ : バッファ記憶上での $N_{IE}$

図5 バッファ記憶装置の動作原理 バッファ記憶装置を採用した場合、バッファ記憶上のデータ不在確率とデータ転送サイクル数が性能上重要である。

主記憶装置に用いられているMOSメモリは高集積化の道を追求しているため、主記憶装置とバッファ記憶装置とのアクセス時間の差はすでに一けたを超えている。この差を許容するには、バッファ記憶装置を相当大容量にして、この $m$ を小さくする必要がある。ところが、バッファ記憶装置は性能第一の記憶装置であり、コスト面からも物理的大きさからも、大容量の装置は不可能である。そこで、この間に位置する新しい中速記憶階層を導入して、 $m$ の代わりに $N_{BF}$ を小さくする3階層記憶制御方式が採用された(図6参照)。

(3) ベクトル処理方式

ベクトル処理とは、科学技術計算に高い比率を占めるベクトル演算を高速に処理することにより、高速化しようとするものである。従来の機械命令に比べて機能レベルの高いベクトル命令を用意し、複合的な処理をまとめて高速化する点にその特長がある。この高速処理方式を大形プロセッサの主要な応用であるデータベース処理に適用したIDP(Integrated Data Base Processor)が実用化されている<sup>4)</sup>。IDPでは、大容量主記憶上にデータベース処理対象をベクトル形式で展開し、ベクトル演算を適用することによって高速化を図っている。

3.3 マルチプロセッサ方式

処理能力ニーズは、図7に示すシングルプロセッサの性能の伸びでは対応できないと言われている。このニーズにこたえるために、複数の命令処理装置が一つの主記憶装置を共有するマルチプロセッサ方式が一般化してきた(図7参照)。

しかし、複数の命令処理装置間で共有される主記憶装置へのアクセス競合などによって、接続したプロセッサ台数に比例して性能は向上しない。マルチプロセッサ方式で性能を向上させるには次の課題がある。

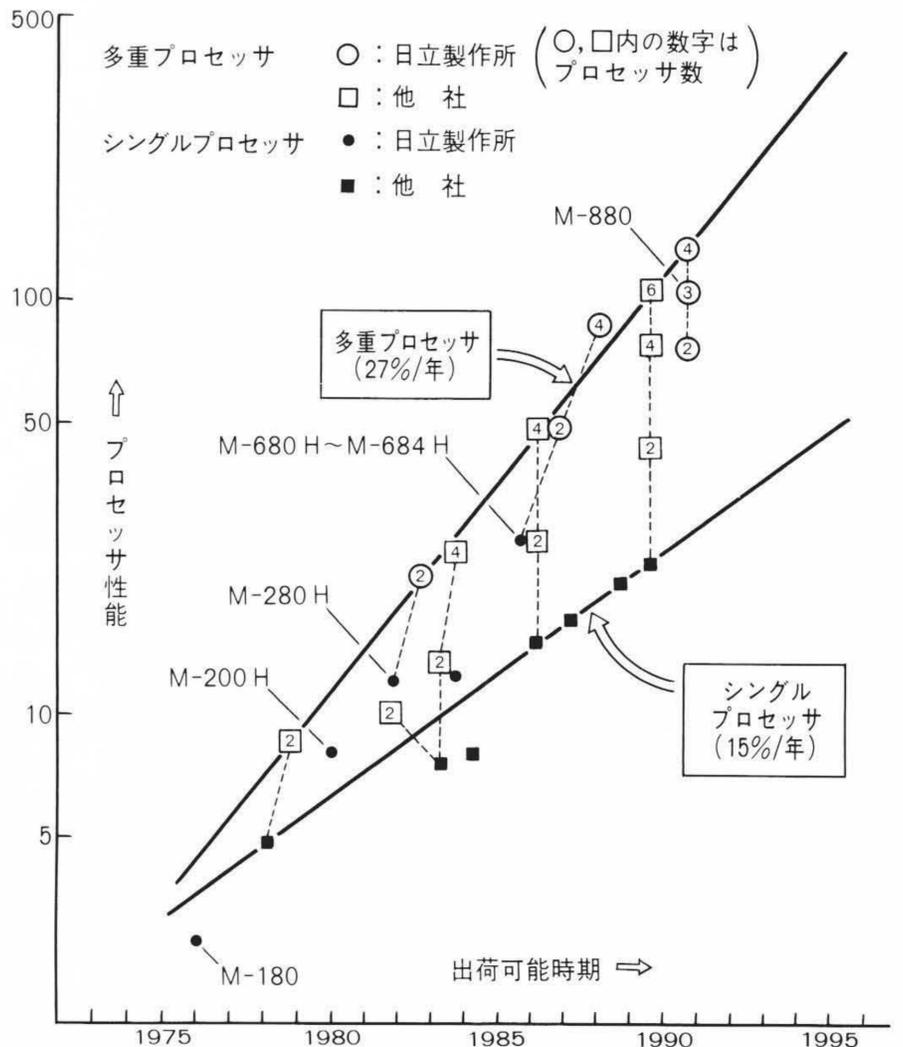


図7 大形プロセッサの性能推移 複数の命令処理装置が、一つの主記憶装置を共有する多重プロセッサが一般化してきている。

- (1) プロセッサ台数に対応した主記憶装置処理能力の確保
- (2) 逐次制御命令などマルチプロセッサ制御命令の高速化
- (3) 複数の階層記憶装置間の内容一致制御の高速化と各階層記憶装置の高効率利用方式

これらの課題に取り組んだ結果、図7に示すように、年率27%の性能向上を実現している。

今後は、より多くのプロセッサが結合でき性能を向上させるために、上記のマルチプロセッサをさらにネットワークや共有記憶で結合する種々のマルチプロセッサ方式が出現するものと期待されている。

3.4 大容量記憶装置

記憶容量の年代推移を図8に示す。半導体記憶素子の大容量化に支えられて、主記憶装置は大容量化している。むしろ現在は、利用技術がこの進歩に追いついていない状況である。

記憶容量、アクセス時間などで主記憶装置と磁気ディスク装置の間に位置する拡張記憶装置が実用化されてきた。この特長を生かして、高速な主記憶装置の実効的な容量拡張と大容量なディスク装置の実効的な高速化が期待されている。今後、拡張記憶装置は利用方式の拡充とともに一般的に普及すると思われる。

このように、大容量半導体記憶素子を原動力とした大容量化の傾向は、ここしばらく継続するものと思われる。

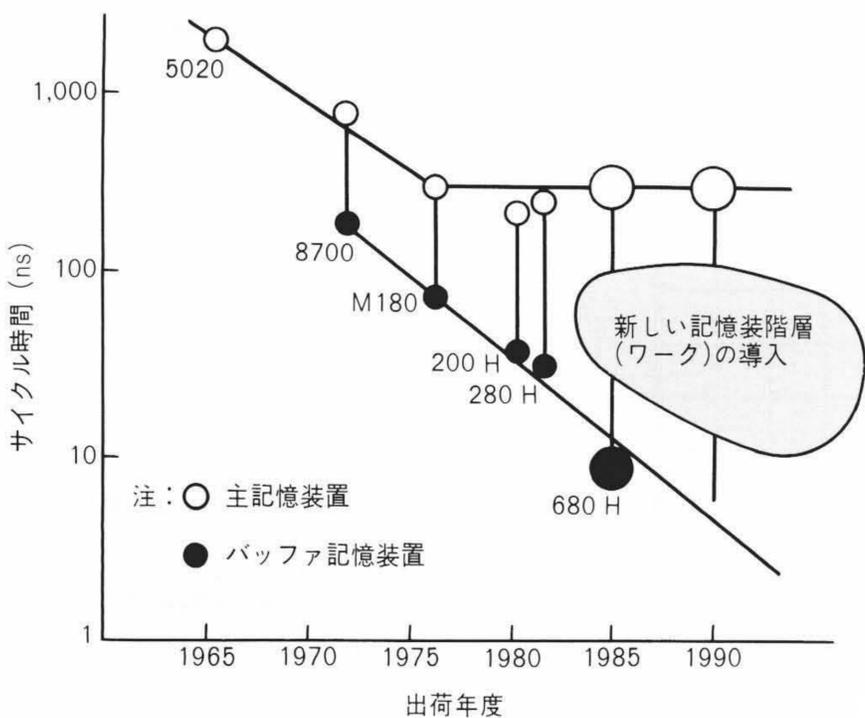


図6 2階層記憶制御方式から3階層記憶制御方式の採用 主記憶装置とバッファ記憶装置とのアクセス時間の差の拡大を解決する方式として、3階層記憶制御方式が登場している。

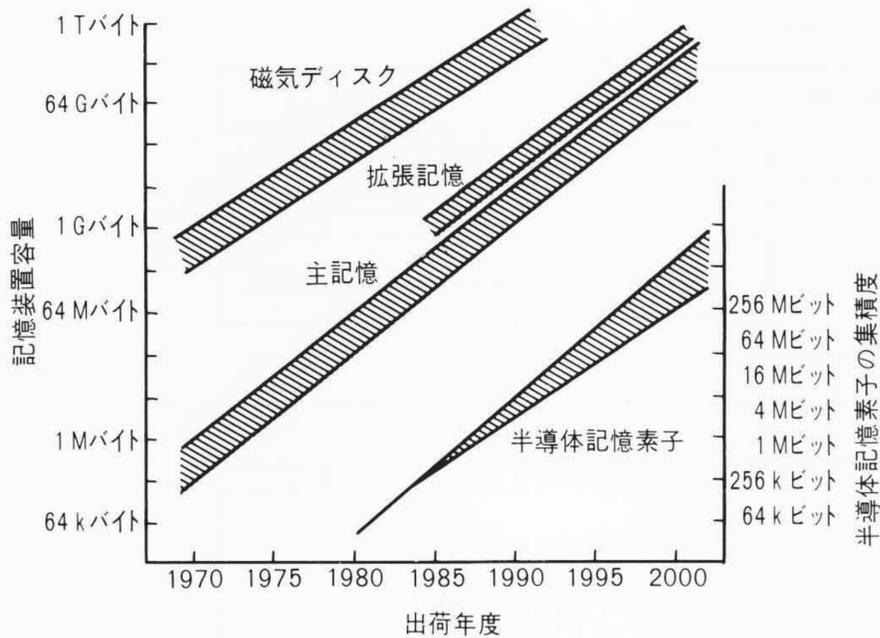


図8 記憶容量の年代推移 半導体記憶素子の大容量化に支えられて、主記憶、磁気ディスクが大容量化している。また、アクセス時間などで主記憶と磁気ディスクの中間に位置する拡張記憶が実用化されている。

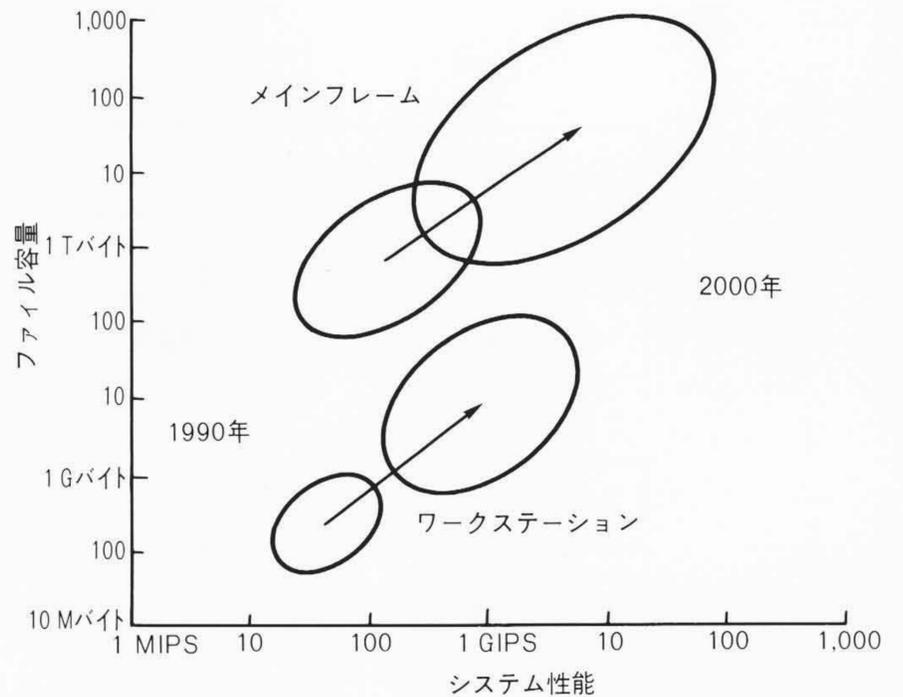


図9 システム性能とファイル容量の動向 メインフレームは、システム性能とファイル容量の観点から存在価値がなおいっそう重要となる。

#### 4 大形プロセッサシステムの今後の展望

##### 4.1 大形プロセッサの役割

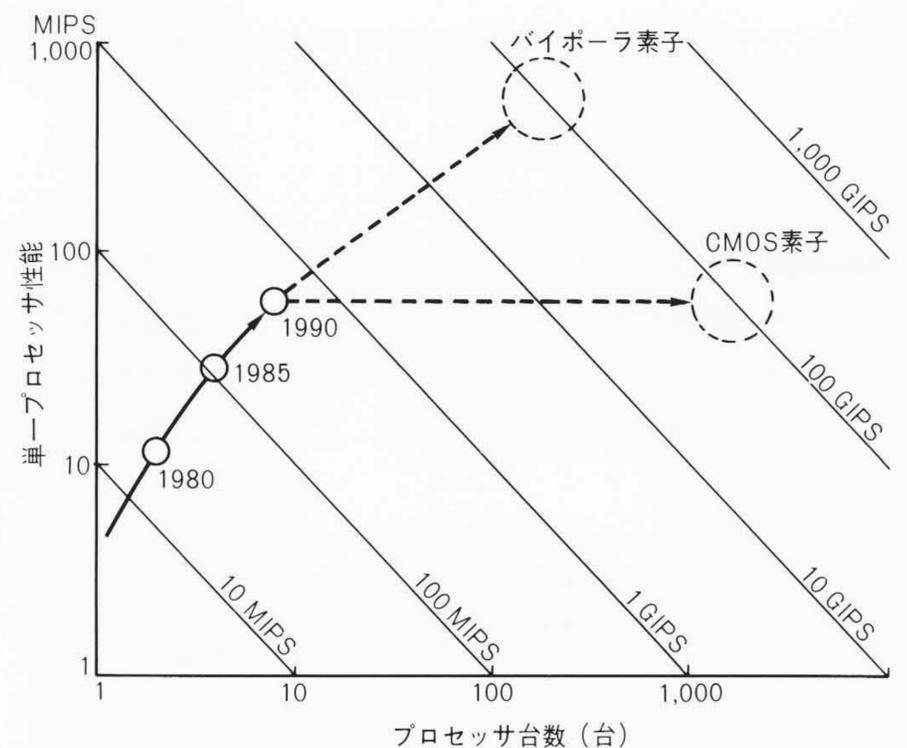
情報システムの分散化、統合化がなおいっそう進展する中で、大形システムの役割は大規模データベースサーバに向かうと思われる。システム性能とファイル容量の動向を図9に示す。同図に示すように、メインフレームはシステム性能とファイル容量の観点から、存在価値がなおいっそう重要となる。

将来のメインフレームは、情報を経営戦略に用いる戦略情報システムの中核として、ワークステーションなどと分散・協調した形態で発展するものと思われる。

##### 4.2 大形プロセッサの構成方式

処理能力のニーズにこたえるには、マルチプロセッサ方式が必須(す)である。複数の命令処理装置から成るマルチプロセッサ方式のMIPSの最大性能は  $N \times \text{MIPS}' (\propto 1 / (T_{MC} \times N_{IE}))$  で表される。ここに  $N$  はプロセッサ台数、 $\text{MIPS}'$  は単一プロセッサのMIPSである。 $T_{MC}$  の短縮は主として半導体素子技術および実装技術によって決定され、この短縮はハードウェアテクノロジーの進展に依存する。今後これを大幅に短縮するには新しい素子の開発を待つ必要がある。 $N_{IE}$  の低減にはパイプライン制御方式、バッファ記憶制御方式、ベクトル演算方式に代わる複数命令並列制御方式などのブレークスルーを待つ必要がある。したがって、単一プロセッサのMIPSに加えてマルチプロセッサ方式に期待が大きい。このような観点から、将来の大形プロセッサシステムを予測すると、図10のようになるであろう。

高速プロセッサの実現方式として、使用するLSI素子の観点



注：略語説明 GIPS(Giga Instructions Per Second)  
図10 単一プロセッサ性能とプロセッサ台数からみた大形プロセッサの予想 バイポーラ素子とCMOS素子を用いた場合のプロセッサ方式の予想を示す。

から、現在の大形プロセッサと同じバイポーラ素子を使用する方式と、小形プロセッサなどで使用されるCMOS素子を使用する方式が考えられる。

一般にバイポーラ素子は、CMOS素子と比べて数倍から一けた高速である。したがって、同一性能を実現するにはバイポーラ素子によるプロセッサ台数は、CMOS素子に比べて数分の一から十分の一で済むことになる。一方、LSIの集積度の点から比較すると、CMOS素子は低消費電力という特長を生かして、バイポーラ素子よりも一けた以上高集積化することが可能である。

将来必要なMIPSを10~100 GIPS(Giga Instructions Per

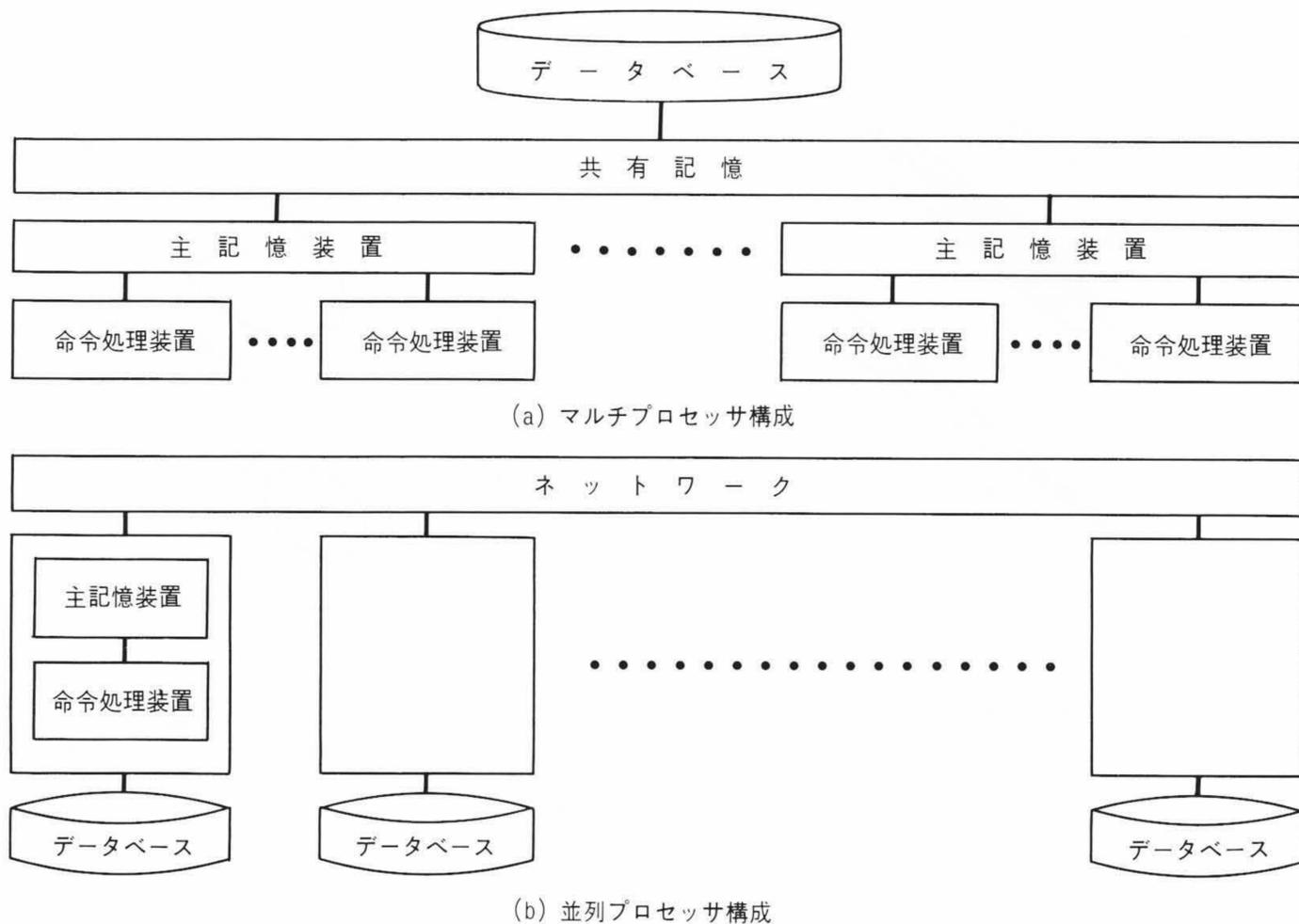


図11 将来の大形プロセッサの構成例 プロセッサを大幅に増加させた、多数台マルチプロセッサ方式が中心になると予想される。

Second) とすると、バイポーラ素子を使用する場合の必要なプロセッサ台数は20から100となる。この場合、主記憶を共有するマルチプロセッサをクラスタとして、複数のクラスタを共有記憶で結合することが可能である。このように構成した例を図11(a)に示す。

一方、CMOS素子を使用する場合の必要なプロセッサ台数は200から1,000が必要となる。このようなプロセッサの構成方式として、図11(b)に示す並列プロセッサがある。この構成では各プロセッサが主記憶を共有しないので、プロセッサ間でデータを転送するための手段が必要となる。このために、ネットワークが設けられている。また、データベースは各プロセッサに分散させることになるであろう。この方式で高速処理を実現するには、分散データベース制御方式の開発がポイントになるであろう。

## 5 結 言

以上、大形プロセッサの発展過程を述べるとともに、将来の大形プロセッサについて展望した。本論文ではほとんど触れることができなかったが、大形プロセッサの性能と使いやすさを決定するもう一つの大きな要素はソフトウェアである。

マルチプロセッサによる高速処理を実現するには、データベースの分散制御方式などが必須である。また、大形プロセッサには高信頼化技術が重要である。耐障害だけでなく、構成変更、ソフトウェアのバージョンアップに対してのノンストップが要求される。

高度情報化社会の基幹情報システムの中核として大形プロセッサは期待され、その期待にこたえる開発がいつそう切望されるであろう。

## 参考文献

- 1) 若井, 外: HITAC M-680H/M-682H処理装置, 日立評論, 67, 12, 987~992(昭60-12)
- 2) 小林, 外: HITAC M-680H/M-682Hのハードウェア技術, 日立評論, 67, 12, 993~998(昭60-12)
- 3) K. Wada, et al.: Design for a High Performance Large-Scale General Purpose Computer, The Hitachi M-680H Processor, ICCD '85, 481~484(1985-10)
- 4) S. Torii, et al.: Accelerating Non-Numerical Processing By an Extended Vector Processor, ICDE '88, 194~201(1988-2)