

解説

応用指向メモリ†



古谷 立美††

1. はじめに

典型的なフォンノイマン型計算機は1台のCPUと1台のメモリから成る。そこでのメモリは、単に情報を記憶するだけのメモリセルに一次元のアドレスを付け、アドレスによってメモリ内のデータにアクセスするものであった。このようなアーキテクチャでは、メモリへのアクセスが、高速処理を妨げるボトルネックと考えられている。近年フォンノイマン型アーキテクチャの限界を破るためにいろいろな提案がなされているが、その一つにメモリを高機能にするというものがある。すなわち、メモリの各ビットあるいはその集合になんらかの論理回路を付加して、メモリ上で論理操作を行い計算処理を高速化しようという考え方である。この種のメモリに相当するものとしては連想メモリ、機能メモリ (functional memory)、論理付きメモリ (logic in memory)、論理メモリ¹⁾、知的メモリ等の言葉があげられる。しかし、ここにあげたようなメモリの名称はそれぞれ正式に定義されたものではなく、提案や製作した人が勝手に名前を付けているというのが現状である。本特集ではそのようなメモリを総称して機能メモリと呼んでおり、本解説でもそれに従うことにする。かつて機能メモリの提案はいろいろあったがその多くのはペーパーマシンにおわっていた。しかし近年VLSI技術が進歩し、さまざまな機能メモリがVLSIで作られるようになってきている。

機能メモリの中で最も歴史があり、機能も整理されているのが連想メモリである。連想メモリについては本特集の中でも別の解説としてまとめるので、本稿では連想メモリ以外の機能メモリを、最近の技術を中心に概観する。

機能メモリを上記のように定義すると、さまざまな形態のメモリが機能メモリの範ちゅうに入ってくるため種々の分類が考えられる。

以下第2章ではさまざまな機能メモリをその発展過程や考え方から整理する。第3章以下は機能メモリを応用分野から分類する。第3, 4, 5章はそれぞれ人工知能用メモリ、図形画像用メモリ、データベース用メモリについて実例を紹介する。

2. 機能メモリ

連想メモリ以外の機能メモリを、その起源から分類すると次の3種類になる。

- (1) セルラロジックからの発展型
- (2) VLSI アーキテクチャ
- (3) ソフトウェアに由来するもの

第1のセルラロジックの発展型の代表は1960年代末にSRIを中心に研究されたlogic in memoryである。基本的にはメモリの各記憶ビットに簡単な論理回路を付加して、記憶以外の論理機能を実現しようというものであった。この中で良く知られているものにSRIのACAM²⁾やMckeeverの機能メモリ³⁾がある。これらの機能メモリは提案された時期が古く、ペーパーマシンに終わったものが多いが、次のVLSIアーキテクチャへの導入と見ることができよう。

第2のVLSIアーキテクチャはVLSIの進歩によって応用に適したVLSIが容易に実現できるようになったことにより出現してきたものである。VLSIアーキテクチャは、簡単な論理セルやプロセッシングエレメント(PE)の繰り返し構造のものが多く、これらの中には機能メモリという観点から設計されたものでもなく、結果的に機能メモリと考えられるものが少なくない。もちろんこれらの中には機能メモリとPEアレイの中間のようなものもあるが、以下では機能メモリを広義にとらえユニークなものを中心に紹介を行う。第2のタイプは第1のタイプに比べると応用への指向が強いと考えられる。

第3のタイプは、ソフトウェアに由来するものである。プログラミング言語で用いられるデータ構造をサポートする機能メモリや、OS等のプロテクションをサポートする機能メモリ等がよく知られている。プロ

† Application Oriented Memories by Tatsumi FURUYA (Electrotechnical Laboratory, Computer Science Division).

†† 電子技術総合研究所

グラミング言語の進歩によりスタック、リスト、多次元データ等さまざまなデータ構造が用いられるようになった。データ構造を従来のメモリで実現しようとすると、ソフトウェアでデータ構造をシミュレートすることとなり効率が悪い。そこでデータ構造をハードウェア的にサポートしようとする考え方が生まれた。この中にはパローズのスタックマシン、SYMBOL等のようにデータ構造を直接ハードウェア化したものもあるが、一般には汎用性を重視し従来のメモリに外部回路やマイクロプログラムプロセッサをつけてプロセッサとのインタフェースをデータ構造メモリに見せているものが多い。

以上、機能メモリを、その起源から分類してきた。この他、メモリセルに対する論理回路のサイズ等による分類も考えられよう。

以下では、今までに提案された興味深い機能メモリを応用の面から分類して検討する。

3. 人工知能メモリ

人工知能用言語として最もよく使われてきたのはLISPであるが、最近では論理型言語やオブジェクト指向言語等も用いられるようになってきた。また、知識の表現法では、意味ネットワーク、フレーム、プロダクションシステム等が広く用いられている。

このような人工知能の処理に最も共通的なものはリスト処理と連想機能である。リストは汎用性が高いため完全なハードウェア化は難しいが、普通のメモリに外部回路やマイクロプロセッサを付けてリスト処理を助けるものや、ハッシュメモリでリストをサポートするマシン⁶⁾などがある。

連想機能は論理プログラムのマッチング等に有効で人工知能メモリの中心的役割をはたすことになる。本章では、VLSI化に適するとして最近注目されている意味ネットワーク用メモリと、機能メモリを多用したAIマシンFAIM-1を報告する。

3.1 意味ネットワーク用メモリ

意味ネットワークとは知識をノード(概念)とリンク(概念間の関係)で表す方法である。もし各ノードを1PEに割り当て、リンクをPE間結線で実現するようなマシンがあれば、高度並列を利用した高速処理が可能である。

意味ネットワークをハードウェア化する提案が1980年Fahlmanにより行われた⁵⁾。これはNETLと呼ばれるもので、意味ネットワークのノードをPE

に割り当て、ノード間の関係をハッシュネットというネットワークで結合した。NETLについてはハッシュネットの方が有名くらいであるが、情報とは単にデータを記憶するだけでなく、データ間の関係も記憶すべきであることを指摘している。

意味ネットワーク用メモリとして最近注目されているのはconnection machine⁶⁾である。これは1981年MITのHillisにより提案され、現在thinking machines社で製作中である。マシンの基本思想はシンプルなPEを2次元アレイ状に4近傍結合し、各PEに意味ネットワークのノードを割り当て、意味ネットワークのノード間の関係はPE間のメッセージ交換で実現しようというものである。図-1はPEの構成を示すものである。現在最初の版を製作中であり、そこでは4×4PEを1チップにおさめ、そのチップ間をキューブ結合するという方式をとっている。意味ネットワークマシンとしてはこの他南加大のSNAP⁷⁾や電総研のIXM⁸⁾がある。

3.2 FAIM-1

FAIM-1はSchlumbergerの研究所で開発中のAIマシンである⁹⁾。図-2はFAIM-1の構成でありHectagonと呼ばれるPEを六方結合したものである。Hectagonの中にはさらに6つのサブシステムに分かれており、内部で並列動作する。このうち次の三つのサブシステムは機能メモリである。

① Context Addressable Memory (CxAM): 普通の連想メモリが、ビットやワードに対して連想処理を行うのに対しCxAMはLispのS式等構造化データに対して連想処理を行う。CxAM等はFIND match, GIVE match, Delete Structure Add structureという命令で動作を行う。CxAMメモリは何台かのサーチエンジンを持ち、各サーチエンジンはメモリのサブエリアに対して並列に構造サーチを行う。

② Instruction Stream Memory (ISM): このメモリはHectagonのプロセッサに命令を供給するメモリである。ISMはブランチ予測を行い命令のプリフェッチ等を実現する。命令はジャンプやコールを起さずに実行できる命令列ごとに固定長トラックに格納される。(トラック末命令がジャンプやコールのことはある。)あるトラック実行中には次に実行される可能性のあるトラックを、バッファに用意し、時間遅れなく命令をプロセッサへ供給する。

③ Structured RAM: 4ビットのタグと16ビットのデータを基本エレメントとしてさまざまなデータ

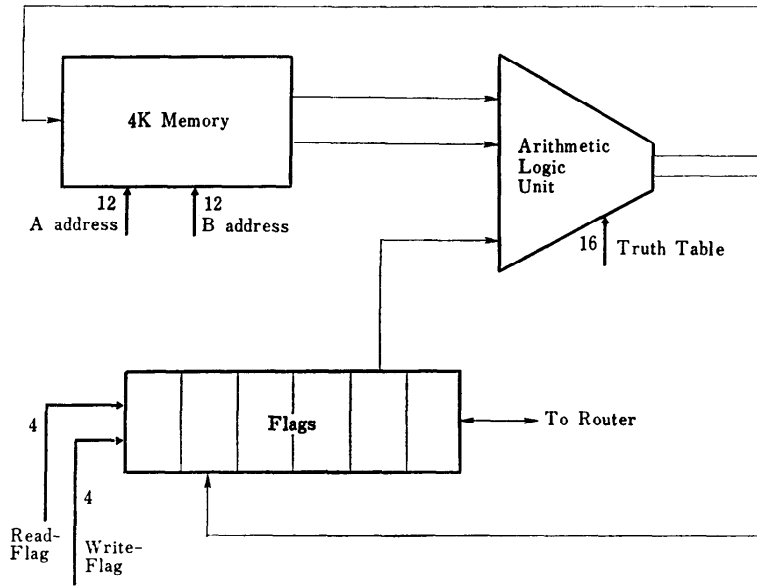


図-1 Connection Machine の PE

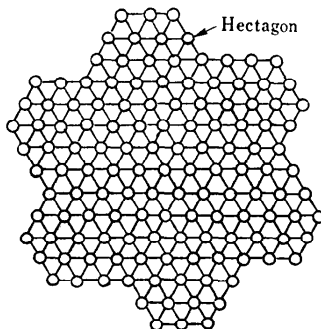


図-2 FAIM-1 の構成

構造を扱えるようになっている。

4. 図形, 画像用メモリ

画像用メモリというまず思われるのがデュアルポートメモリかもしれない。これはビットマップディスプレイのフレームバッファ用として需要が急増し、各社が力を入れているものである。この中には直並列入出力機能の他簡単な論理操作を行うものまであるが、メモリ構成は従来の RAM に外部回路を付けた形である。デュアルポートメモリにはいろいろ文献があるのでそれらを参照されたい¹⁰⁾。

図形や画像は、2次元形態をとり、近傍の画素との間の演算がよく行われるという性質があり、古くから

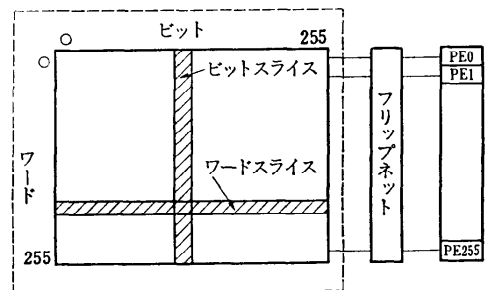


図-3 STARAN の構成

画像処理用プロセッサの提案は多い。その中には1画素を1PEに割り当てようというものも多く、それらは画像用機能メモリと考えることができる。古くはUngerマシンに始まり、最近ではPEを2次元(128×128)に並べたMPP¹¹⁾やAAP¹²⁾が知られている。

本章ではメモリセルの機能が一般のPEに比べて低く、高集積化が期待できる直交メモリ、とCADでよく用いられるパスファインダ用メモリを取り上げる。

4.1 直交メモリ

直交メモリはSTARAN¹³⁾で用いられた方法で、図-3の点線内のようにビット方向とワード方向に読み書きできるメモリである。直交メモリはそれ自体は連想メモリとは考えられないがPEでビット方向に線

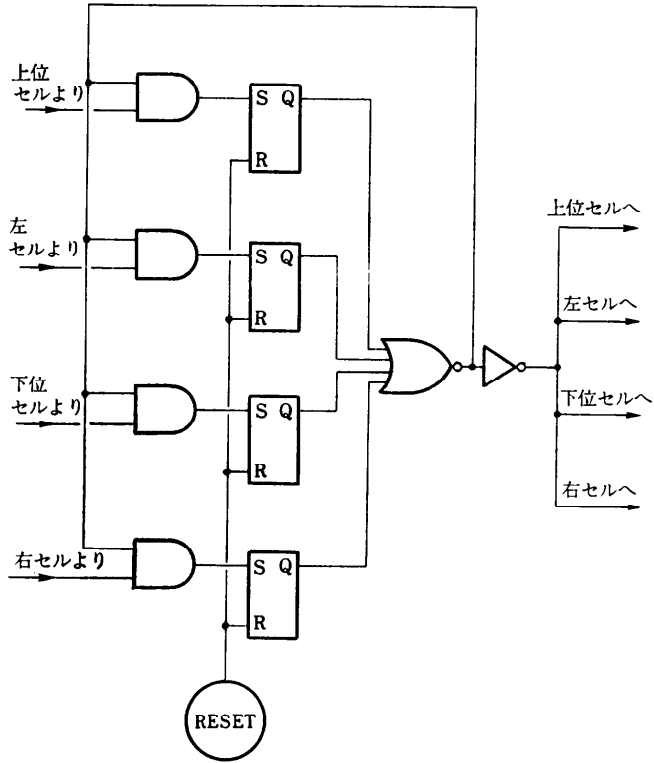
5	4	5	6	7	8	
4	3	4		B8		8
3	2					7
2	A1	2	3	4	5	6
3	2	3	4	5	6	7

Lee-Moore のアルゴリズム

→	↓	→	←	←	←	
→	↓	←		B↑		↓
→	↓					↓
→	A	←	←	←	←	←
→	↑	←	←	←	←	←

MAZER 用アルゴリズム

(a) A, B間のパスファインディング



(b) セルの構造

図-4 パスファインダ

り返し比較を行うことにより連想機能を実現でき画像処理やデータベース処理への適用が可能となる。STARAN では直交メモリに普通のメモリを用意し、ビット方向とワード方向の読み書きは直交メモリとPE間のフリップネットを利用して実現している。これに対し、実際にビット方向とワード方向を読み出せるメモリがVLSIで実現されている。電子技術総合研究所ではメモリセルにビット方向の読み出し、書き込みの線を加えて直交メモリを実現した¹⁴⁾。この直交メモリのトランジスタ数は同容量のRAMに比べて約150%であった。

4.2 パスファインダ

LSI設計でエレメント間の配線経路を探す作業は大変時間のかかる作業である。

LSIは2次元形をしているため論理付きメモリでこれを実現しようという提案がある。このうちよく知られているのは、カリフォルニア工科大学のMAZERチップである¹⁵⁾。配線経路探索アルゴリズムとしてよく用いられるのがLee-Mooreのアルゴリズムである。

Lee-Mooreのアルゴリズムとは配線する面をメッシ

ュに分割し、結線する2点の一方から4近傍に波を伝搬するように番号を付けていき目的点に到着したとき、目的点から番号を逆にトレースして道を定めるというものである。しかしLee-Mooreアルゴリズムを直接用いると、番号を覚えるレジスタサイズが大きくなる。そこでCarroll等は各セルには番号を覚えさせるかわりに波が入ってきた方向と出ていった方向を覚えさせるように変更し簡単化を図った。図-4がこの記憶セルの構造で、波の入出力方向を記憶するフリップフロップと、1度信号が入ったら次の信号をサプスする回路よりできている。

5. データベース用メモリ

データベースという大容量記憶が必要ということから、従来のデータベースマシンはデータをディスク等二次記憶へ置くことを前提に議論が進められてきた。一方、近年のVLSIの集積度の向上によりVLSIメモリを記憶媒体とするデータベースマシンも将来の有望な候補と考えられるようになってきた。これに伴いVLSIによるファイル処理やデータベース処理が

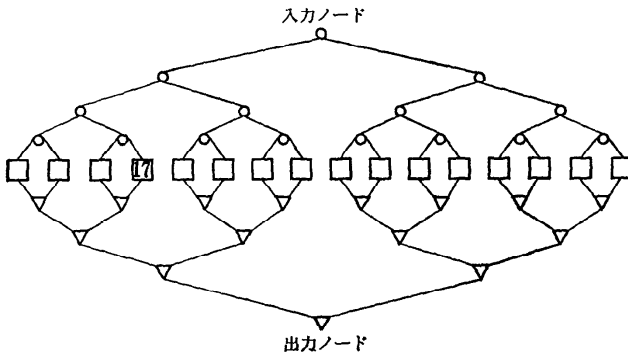


図-5 Bentley のトリーマシン

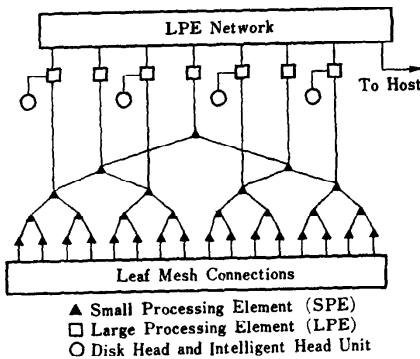


図-6 Non-Von マシン

提案されだしてきている。本章では、それらの代表的なものと考えられる、ディクショナリマシンと VLSI リレーショナルデータベースマシンを取り上げる。

5.1 ディクショナリマシン

ディクショナリマシンとは、キーとそれに対応付けられたレコードを記憶しておき、それに対してキーによるサーチ、インサート、デリートを行うマシンである。

この最初の提案は Bentley-Kung のもの¹⁶⁾である。図-5 はその構成を示している。そのマシンでは3種類のプロセッサが使われる。各キー、レコード対は四角のプロセッサに格納される。丸で示されるプロセッサは命令とデータを四角プロセッサに送るためのものである。結果は三角のプロセッサを通して集められる。たとえば“キー17を持つレコードがマシン内にあるか?”という質問が出されると、それが四角プロセッサへブロードキャストされる。各プロセッサは自分のキーを調べ、17を持つものは出力として三角プロ

セッサの方へ“1”を送る。これは最も簡単な例であるが Bentley 等はこのマシンを使ってさまざまな仕事ができることを示している。このマシンのもう一つの特徴は入力ルートプロセッサから命令をパイプライン的に送り込むことができるという点である。

その後このマシンの改良版が次々と提案されている。Bentley のマシンはデリートがパイプライン処理できない点に着目した Leiserson は隣接する四角プロセッサ間に通信回路を与え、この問題を解決した¹⁷⁾。Ottmann 等はキー、データペアを上記の三角プロセッサにも割り当てることによりサーチ時間を短縮する方法を示した¹⁸⁾。その他 Ottmann の改良型や複数キーをサポートする方法等の提案がある。

5.2 リレーショナルデータベース用 VLSI

リレーショナルをサポートする VLSI の提案としては文献 19), 20) 等がある。ここでは、Non-Von²¹⁾を取り上げて概要を示す。Non-Von は1982年ごろからコロンビア大学で研究開発されているマシンであり、知的データベース、リレーショナルデータベース、AI 応用等を対象としたマシンである。図-6 が Non-Von マシンの構成である。Non-Von グループは Non-Von をアクティブメモリと称しており、数千から百万の SPE (Simple Processing Element) を木状に結合しようとしている。

なおリーフの SPE はリーフ SPE 同士2次元メッシュ結合されている。図-6 の LPE は現在市販のマイクロプロセッサを考えており、SPE へ SIMD 形式で命令を供給する。このため、Non-Von 全体としてはマルチ SIMD で動作を行うことができる。各 SPE は 64 word×8ビット RAM, 8つの8ビットレジスタ、8つの1ビットレジスタ、1ビット ALU, バイト I/O スイッチ、1バイト算術比較ユニット等からできている。SIMD 命令としてはレジスタ間ムーブ、メモリアクセス ADD 1, SUB 1, シフト、論理演算、通信、等がある。

6. おわりに

知識処理やパターン認識の分野では、現在の計算機の数百倍の処理速度が要求されているが、そこで用いられるアルゴリズムは機能メモリという形でハードウェア化できるものが多い。一方機能メモリは簡単な回

路の繰返し構造である場合が多く VLSI 化に適していることから、今後さまざまな機能メモリ、応用指向メモリが開発されると思われる。

以上では、できるだけ広い範囲から応用指向メモリと言えそうなメモリを集めてみた。応用分野については機能メモリの提案者自身がさまざまな応用への適用性を示唆している場合が多く上記のように一意には決めかねる点もある。

以上で取り上げてきた例の多様性からもわかるとおり、VLSI の進歩に伴ってさまざまなメモリが登場しメモリの意味自体が明確に定義しにくくなってきている。これからさらにいろいろなタイプのメモリが登場してくるであろうが、それらが応用指向メモリという言葉で呼ばれるようになるのか、専用マシンや VLSI アーキテクチャの中に吸収されるのか歴史の評価を持つしかなさそうである。

参考文献

- 1) 飯塚：論理メモリ，情報処理，Vol. 16, No. 4, pp. 274-285 (Apr. 1975).
- 2) Kautz, W. H.: An Augmented Content Addressed Memory Array for Implementation with Large-scale Integration, JACM, Vol. 18, No. 1, pp. 19-33 (Jan. 1971).
- 3) McKeever, B. T.: The Associative Memory Proc. FJCC, pp. 371-388 (1965).
- 4) Goto, E. et al.: Flats, A Machine for Numerical, Symbolic and Associative Computing, Proc. Computer Architecture, pp. 102-110 (1980).
- 5) Fahlman, S. E.: Design Sketch for a Million-Element NETL Machine, Proc. 1st National Conf. on AI, pp. 249-252 (1980).
- 6) Hillis, W. D.: The Connection Machine, The MIT Press (1985).
- 7) Dixit, V. and Moldovan, D. I.: Discrete Relaxation on SNAP, Proc. The 1st Conf. on Artificial Intelligence Applications IEEE Computer Society (1984).
- 8) Higuchi, T. et al.: A Semantic Network Language Machine, EUROMICRO 85, pp. 95-104 (1985).
- 9) Davis, A. L. et al.: The Architecture of the FAIM-1 Symbolic Multiprocessing System, IJCAI 85, pp. 32-38 (1985).
- 10) 日経エレクトロニクス：画像用 256K デュアルポートメモリにラスタ演算や高速クリア用機能を追加，日経エレクトロニクス，No. 378, pp. 109-111 (1985).
- 11) Batcher, K.: The Massively Parallel Processor (MPP) System, Proc. of the Computer in Aerospace (Oct. 1979).
- 12) Kondo, T. et al.: An LSI Adaptive Array Processor IEEE J. Solid-State Circuit SC-18, pp. 147-156 (1983).
- 13) Batcher, K. E.: STARAN Parallel Processor System Hardware, Proc. NCC, pp. 405-410 (1974).
- 14) Kokubu, A. et al.: Orthogonal Memory-A Step toward Realization of Large Capacity Associative Memory VLSI 85, pp. 159-168 (1985).
- 15) Carroll, C. R.: A Smart Memory Array Processor for Two Layer Path Finding, Proc. 2nd Caltech Conf. VLSI '81, pp. 165-195 (1981).
- 16) Bentley, J. L. and Kung, H. T.: A Tree Machine for Searching Problems, Proc. IEEE Int. Conf. on Parallel Processing, pp. 257-266 (1979).
- 17) Leiserson, C. E.: Systolic Priority Queues, Carnegie-Mellon Univ., CMU-CS-79-115 (1978).
- 18) Ottmann, T. et al.: A Dictionary Machine for VLSI, IEEE Trans. Computer., Vol. C-32, pp. 892-896 (Oct. 1982).
- 19) Song, S. W.: A Highly Concurrent Tree Machine for Data Base Applications. Proc. IEEE 1980 Int. Conf. on Parallel Processing, pp. 256-260 (1980).
- 20) Bonuccelli, M. et al.: A VLSI Tree Machine for Relational Data Bases. Proc. 10th Annual Symp. on Computer Architecture, pp. 67-73 (June 1983).
- 21) Shaw, D. E.: NON-VON's Applicability to Three AI Task Area, Proc. of the Ninth Int. Joint Conf. on Artificial Intelligence, pp. 61-72 (Aug. 1985).

(昭和 61 年 2 月 10 日受付)