

## ■6群(コンピューター基礎理論とハードウェア) - 1編(コンピュータの歴史)

### 3章 歴史的意義が大きいコンピュータ

(執筆者：児玉祐悦) [2009年4月 受領]

#### ■概要■

これからコンピュータ技術を学ぶ人がどのように重要技術が開発されたかを知るための知識を与えることを目的として、その後のコンピュータに大きな影響を与えた技術を開発した歴史的コンピュータを中心に取上げるとともに、社会的影響の大きかったコンピュータについても考慮して12のコンピュータを取り上げた。

#### 【本章の構成】

本章では、高級言語指向マシンとして実用化された特徴的なコンピュータとして Burroughs stack machine B5000 (3-1 節)、最初の商用スーパーコンピュータとして CDC 6600 (3-2 節)、第3世代を創ったファミリマシンとして IBM System/360 (3-3 節)、TSSの実用化として Multics (3-4 節)、ミニコンピュータの代表として DEC PDP-8, PDP-11, VAX (3-5 節)、最初の商用ワンチップマイクロコンピュータとして Intel MCS-4 (3-6 節)、初めてのパーソナルコンピュータとして Xerox Alto (3-7 節)、商業的に成功した最初のベクトル型スーパーコンピュータとして Cray-1 (3-8 節)、RISCアーキテクチャの祖として IBM 801 (3-9 節)、パソコンの発展の基礎を築いたマシンとして IBM PC (3-10 節)、和製スーパーコンピュータとして地球シミュレータ (3-11 節)、組込みシステムの原型としてアポロ誘導コンピュータ (3-12 節)を取り上げて、その特徴やその後のコンピュータに与えた影響などについて述べた。

## ■6群 - 1編 - 3章

### 3-1 Burroughs stack machine B5000

—高級言語指向マシンとして実用化された特徴的なマシン—

(執筆者：児玉祐悦) [2009年4月 受領]

B5000 は 1961 年バロース (Burroughs) 社により発表され、1963 年に稼動したコンピュータである。チーフアーキテクトは Robert S. Barton (1979 年エッカーートモークリ賞) である。1960 年発表の高級言語 Algol60 の構造をハードウェアレベルで色濃く反映しており、高級言語コンピュータの源流となっている。初めて仮想記憶やデュアルプロセッサを実現した点でも有名である。また、命令形式としてスタック機構に基づく 0 アドレス方式を採用していた。その後 B5500, B6500 とシリーズ化されており、B7700 は最大 8 プロセッサでありマルチプロセッサとしても先駆的であった。バロース社はその後スペリー (Sperry) 社と合併し、ユニシス (Unisys) 社となっている。

Algol60 は、変数や関数のスコープがブロック構造をしている。例えば図 3・1 に示すプログラムでは、ブロック 1 ではブロック 1 と 0 で宣言した変数が参照できるが、同じ変数名の場合は近いブロック (ここではブロック 1) が参照される。そのため最初の print 文では a は 12 となる。一方、ブロック 0 ではブロック 0 で宣言した変数のみが参照できるため、次の print 文では a は 1 となる。B5000 ではこのブロック構造を実行環境としてハードウェアで実現している。ハードウェアは IP (命令ポインタ) と EP (環境ポインタ) をもっている。プログラムはブロックごとに分割されてセグメント辞書からリンクされて管理される。IP は実行中のコードセグメントを保持するセグメント辞書のインデックスとブロック内変位により実行する命令を参照する。EP は参照する変数などを保持する各ブロックのスタックセグメントの先頭へのポインタであり、最大 32 段までのポインタを保持できる。変数の参照はポインタの番号とスタックセグメント内変位で指定する。

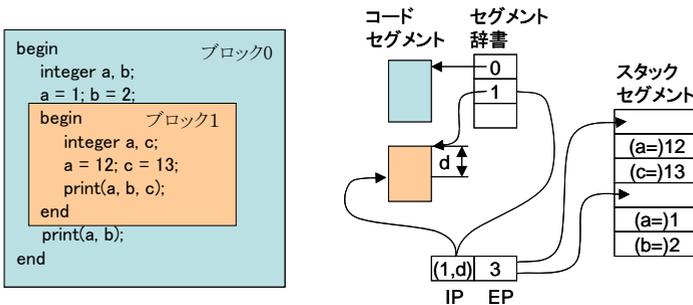


図 3・1 Algol プログラムのブロック構造と B5000 の実行環境

B5000 の機械命令形式は 0 アドレス方式となっており、スタック上のデータに対して演算が行われる。スタックは先入れ後出しを行うデータバッファである。PUSH A とするとデータ A がスタックトップに格納され、スタックにもともと存在してデータはスタックの中に押し込まれる。また、POP を実行すると、スタックトップのデータが取り除かれ、押し込まれていたほかのデータが浮かび上がる。OP (例えば加算) を実行すると、スタックトップとそ

の下のデータが POP され、演算が実行され、その結果がスタックトップに PUSH される。算術式  $(A+B) \times (C+D)$  は、まずコンパイラで逆ポーランド記法  $AB+CD+$  に変換され、ハードウェアでの実行は逆ポーランド式の左から操作をして、変数（値）であれば PUSH を、演算であれば演算を実行することでなされる。図 3・2 に  $(5+4) \times (2+3)$  の計算の例を示す。B5000 ではスタックトップの二つがレジスタにアサインされ、それ以降は現在のスタックセグメント上に格納され、そのポインタがレジスタで管理される。

通常の算術式:  $(5+4) \times (2+3)$   
逆ポーランド記法:  $5\ 4\ +\ 2\ 3\ +\ \times$

PUSH 5	PUSH 4	ADD	PUSH 2	PUSH 3	ADD	MUL
5	4	9	2	3	5	45
-	5	-	9	2	9	-
-	-	-	-	9	-	-

図 3・2 スタックによる式の計算例

B5000 は 1 語 51 ビットであり、3 ビットがタグフィールドで、残りの 48 ビットが実際の情報である。タグは、その語がデータであるか、命令であるか、制御データであるかを区別する。データ型は単精度データ（指数部 6 ビット、仮数部 39 ビット）と倍精度データのみで、整数データは指数部を 0 として扱う。タグフィールドによりデータ型を区別するため、例えば、加算命令は 1 個のみである。0 アドレス方式のため各命令は 8 ビットと非常にコンパクトであった。また、命令などデータ以外の語はユーザからの書き込みが禁止されていた。

B5000 にはセグメンテーション方式の仮想記憶が採用されている。図 3・1 でセグメント辞書には P（プレゼンス）フラグが付加されており、これが 0 のときにはメモリ上にコードセグメントがないので割込みが発生し、外部記憶からメモリにコードセグメントをロードしてから命令を再開する。

B5000 は上で述べた以外にも、OS がアセンブラではなく Algol60 を拡張した高級言語で記述されていたり、ハードウェア設計とソフトウェア設計が同時に行われたり、ハードウェア割込みやマルチプロセッシングなどがスタック構造の手続き読み出しと統一的手法で実現されていたりと、先駆的な試みがなされており、非常に特徴的なコンピュータである。

より詳しい情報は文献 1), 2), 3) を参照。

## ■6 群 - 1 編 - 3 章

### 3-2 CDC 6600

#### —最初の商用スーパーコンピュータ—

(執筆者：北村俊明) [2009年4月 受領]

CDC 6600 はスーパーコンピュータの生みの親といわれる Seymour Cray により開発され 1964 年から製造された最初のスーパーコンピュータである。図 3・3 に CDC 6600 のコンソールと筐体の写真を示す。Cray 自身は、CDC 6600 の前に、Electronic Research Associates (ERA) 社で ERA 1103 の制御ユニットを開発したのを皮切りに、Sperry Rand (ERA は、Remington-Rand に買収され、更に Sperry に買収されてその Univac 事業部門の一部となっていた) で Navy Tactical Data System (NTDS) を開発、その後、CDC 社に移って CDC 1604 とその I/O プロセッサの 160 などを開発していた<sup>4),3)</sup>。



図 3・3 CDC 6600 のコンソールと筐体  
(Computer History Museum の許可を得て掲載)

CDC 6600 の開発に当たって、世界最高速を目指し、回路方式やパッケージング、冷却に至るまで最高の技術を使用し、設計制約にコストは考えないという言葉ばスーパーコンピュータの定義を打ち立てた。基本の動作周波数は 10 MHz であるが 4 位相を使うことで当時としては驚異的な 40 MHz 相当で動作させている。このため 60 ビットの浮動小数点加算を 4 サイクル (400 ns) で実現している。

まず 1 番の特徴は、プロセッサ内の並列処理である<sup>5)</sup>。いまでこそ、複数の演算器で同時に処理を行うのは普通になっているが、ハードウェアによる制御ユニットが、独立した演算ユニットを利用して実行可能となった命令を次々に実行する (逆に依存関係で実行できないものは一時的に待たせておく) というデータ駆動による実行制御を行っている。この制御の鍵になるのが「スコアボード (Scoreboard)」として知られている方式である。これは命令ごとにオペランドのレジスタ番号や結果格納のレジスタ番号、オペランドが利用可能かどうか

かのフラグやそのオペランドを生成する演算器の識別子などを記憶し、レジスタと演算器間のデータの移動をすべて制御している。命令は15ビット長の3オペランド形式のものと30ビット長で18ビットの即値をもつものの2種類が用意されている。演算器は、論理演算、シフト、固定小数点加算、浮動小数点加算、浮動小数点乗算×2、浮動小数点除算、インクリメント×2、分岐のユニットが装備されており、レジスタとしては、60ビット浮動小数点レジスタ(X)が8本、18ビットのインデックスレジスタ(B)が8本、18ビットのアドレスレジスタ(A)が8本用意されている。

第2の特徴は、システムとしての並列処理である。CDC 1604でも採用されている入出力専用の小型プロセッサ(PPU: Peripheral and Control Processors)を10台備え独立して動作可能である。PPUは12本の入出力チャンネルと任意の組合せで使用でき、主記憶と直接接続されている。また、主記憶は60ビット幅で32バンクにインターリーブされ、Stunt Boxがバンク間の並列処理を制御している。また、記憶階層として拡張記憶を主記憶の外側に設け主記憶と高速転送を可能としている。

最後の特徴は、その実装方式である。極限まで動作周波数を高めるため、プリント基板に平面的に部品を配置するのではなく、並行した3インチ四方の2枚のプリント基板間に部品を挟み込むように実装してcordwoodモジュールを形成している。このモジュールをラックに差し込んで実装するが、モジュールを挟み込むようにコールドプレートを配置し、これを各ロッカーに設置されているフレオン冷却装置で冷やして、モジュールからの発熱を吸収している。ラックは4列を一つのロッカーに収容し、ロッカー間の配線長を最小にするために、四つのロッカーは十字の形に配置されている。その後Crayが開発したスーパーコンピュータの代名詞のようにいわれるCray 1では放射状に基板を配置しており、このデザインの原型を見ることができる。

このマシンは、スーパーコンピュータとの関連を一番に取り上げられるが、このマシンの設計思想は、単にスーパーコンピュータに対する影響だけでなく、RISC命令形式やスーパーカラ方式の先駆的な試みとして大きな影響を与えた。

## ■6群 - 1編 - 3章

### 3-3 IBM System/360

#### ー第3世代を創ったファミリマシンー

(執筆者：飯塚 肇) [2009年4月 受領]

1964年4月7日にIBM社が発表したマシン群で、その開発に5億ドルの投資をしたが、大きな成功を収め、メインフレームマシンを確立し、いわゆる第3世代を切り開いた。最大の目標は、従来4種あったコンピュータ系列を統一して、1個のソフトウェア互換性があるマシンファミリを構築することであった。名称の由来は、“データ処理の360度に高性能”ということから名付けられた。このファミリは大きな成功を収めて、その後各種のモデルが追加された。1970年にはSystem/370に進化し、現在のSystem zへと続いている。また、他社からも多数のソフトウェア互換マシンが発売された。ここでは、最初の発表、及び、後継機の中でもその後のマシンに大きな影響を与えたモデル91と85について述べる。

アーキテクト： アムダール (Gene M. Amdahl, 1987年エッカー特・モークリ賞)、ブラーウ (Gerrit A. Blaauw), ブルックス (Frederick P. Brooks, Jr., 1999年チューリング賞, 2004年エッカー特・モークリ賞)

特徴： その後のコンピュータに大きな影響を与えた以下のような技術を確立した。

- (1) 同じ命令セットアーキテクチャをもつ“性能/コスト”の異なるマシンファミリ (当初は、モデル30から70まで6機種で、最高性能と最小コストモデルの性能比は50倍であった) を実現し、命令セットアーキテクチャと実装を分離して、ファミリマシンの概念を確立した。
- (2) (1)の実現のためにマイクロプログラム制御方式を確立した。
- (3) バイトアドレッシング、及び、従来の専用レジスタ方式に代えたレジスタ・メモリ型汎用レジスタ (16個) アーキテクチャを採用して、可搬性のあるハードウェア/ソフトウェアインタフェースを初めて実現し、その後の命令セットアーキテクチャの先駆となった。

より詳細な情報は文献6)を参照。

#### モデル91

1967年に追加された高性能モデルで、レジスタリネイミング機構を導入して命令のアウト・オブ・オーダー (out-of-order) 実行を行った。この機構は、論文の著者の名前を用いてトマスロ (Tomasulo) アルゴリズムと呼ばれる。その後1980年代までは注目されることはなかったが、1990年代以降、高性能マイクロプロセッサのアウトオブオーダー実行の基本技術として広く利用されるようになった。

アーキテクト： フリン (Michael J. Flynn, 1992年エッカー特・モークリ賞)、トマスロ (Robert M. Tomasulo, 1997年エッカー特・モークリ賞)

特徴： このモデルの最も大きな貢献であるトマスロアルゴリズムについて述べる。

システム360の命令セットでは、浮動小数点レジスタが4個と少なく、レジスタの競合が発生して多数の命令を並列実行することができない。そこで、命令発行時に演算器に付属する予約ステーションと呼ばれる物理レジスタを割り当てて (予約ステーションは各演

演算器に複数個用意され、実行待ちの間のデータのバッファリングの役割も果たす)、以後のそのデータの参照をその予約ステーション名に変換する。演算に必要なほかのオペランドは、それを生成する演算器が結果を共通バス (CDB) に出力したとき、それを必要とする予約ステーションが取得し、その結果オペランドがそろって、演算器が利用可能なら実行が開始され、結果が得られると CDB に出力してそれを必要な予約ステーションが利用する、という手順で実行が進む。このようにレジスタが命令発行時に予約ステーション名にリネームされる結果、命令セットアーキテクチャで利用できるレジスタ数に制限されることもない並列実行が可能となる。

より詳細な情報としては、全体的考え方は文献 7) を、Tomasulo アルゴリズムについては文献 8) を参照。そのほか、モデル 91 関連論文が同じ号に掲載されている。

## モデル 85

1968 年に追加されたモデルで、キャッシュを備えた最初の商用コンピュータである。キャッシュの概念は、1965 年にウイルクス (Maurice V. Wilkes, 1967 年チューリング賞, 1980 年エッカート・モークリ賞) によって提案されたスレーブメモリに始まるが、ここで導入されたこの用語が以後広く使用されるようになった。メモリのレイテンシを小さくできるキャッシュは高速化に非常に有効で、モデル 91 のアウトオブオーダー実行に比べて、はるかに簡単かつ広範囲のプログラムに有効なので、このモデル以後広く利用されている。論文は、「将来のほとんどの大型システムに組み込まれる重要な前進 (fundamental step forward) である」と誇っているが、まさにそのようになったわけである。

**特徴:** モデル 85 のキャッシュは、現代マシンのものとは構成がやや異なるセクタ方式を採用している。容量 16 KB は 16 個の各 1 KB のセクタ (現代用語ではブロック) に分割され、各セクタは更に 16 個のブロック (現代用語ではサブブロック) に分割されている。主記憶とのマッピングはセクタ単位に完全連想方式で行われるが、転送はブロック単位で行われる。また、入換えアルゴリズムは LRU、書き込みはライトスルー、書き込み時のブロックの割当ては行わず、メモリサイクル時間は、主記憶が 1040 ns、キャッシュが 80 ns と 13 倍の差があった。

より詳細な情報は文献 9), 10) を参照。

## ■6群 - 1編 - 3章

### 3-4 Multics

#### —TSSの実用化—

(執筆著者：児玉祐悦) [2009年4月 受領]

Multics (MULTiplexed Information and Computing Service) は MIT の調査プロジェクトとしてコルバト (F. J. Corbato, 1990年チューリング賞) らを中心に 1964年に開発を開始した大型の TSS (Time Sharing System, 時分割多重) システムである。1965年秋には MIT とゼネラル・エレクトリック (GE) 社, AT&T ベル研究所との共同プロジェクトに発展した。Multics のために GE 社の GE 635 を改良して GE645 が開発された。1969年にはベル研が脱退したが、1969年10月には MIT において一般の利用者に使用されるとともに、1970年に GE のコンピュータ部門を買収したハネウェル (Honeywell) 社により商品化されている。Multics は TSS や階層化ファイルシステムなど、現在に至るまでの OS の設計に与えた影響は大きい。2007年に MIT より FreeBSD ライセンスのもとでソースコードが公開された。

TSS とは中央に汎用大型コンピュータを設置し、当時は電話回線によって多数の端末を結合したシステムであった。時分割多重方式によってユーザプロセスが一定の時間 (タイムクアンタムという。例えば 50 ms) ごとに切り替わるようになっており、多数のユーザが同時にシステムを利用でき、しかもシステムを占有しているかのように対話的に操作できた。当時はバッチ処理が一般的であり、プログラム投入から結果が出るまでのターンアラウンド時間が長かった。MIT では 1961年に最初の TSS システムのひとつである CTSS (Compatible Time-Sharing System) をマッカーシ (John McCarthy, 1971年 ACM Turing 賞) らにより開発していた。この Compatible とは、バッチ処理 OS との互換性を意味し、バッチ処理のバックグラウンドでユーザプロセスを動かすものであった。ただし、フォアグラウンドのプロセッサソースにはアクセスできないなど完全な TSS とはいえなかった。Multics では、それらの点を改良し、プログラムの容易さや、電話や電力サービスのようなユーティリティとしての新しいコンピュータの応用を目指して、対話的な利用環境を実現した。

Multics は汎用のシステムを目指し、システムソフトウェアは PL/I 言語によりプログラミングされた。これはマシンに依存しない言語を用いることにより、ハードウェアの進歩に簡単に追従できることを目指したものであった。高級言語を用いて OS を記述した例は、パロースの B5000 以外にはほとんどなかった。ソフトウェアがモジュール化されるとともに、ハードウェアもモジュール化され、プロセッサやメモリ、入出力装置などモジュール単位に拡張可能な構成となっていた。

Multics は当初 GE645 上で開発された。GE645 はセグメントとページングを組み合わせた命令アドレスをサポートしており、これによりユーザはそれぞれが大規模なメモリが使えるものとしてプログラムを作成できた。また、ダイナミックリンクが実現されており、ライブラリはプログラム実行時に必要に応じて自動的にロードされ、実行することが可能であった。これによりコードサイズの削減やライブラリの共有によるメモリ使用量の削減などが実現でき、TSS の応答性能が向上した。また、セグメントやページを指示する記述子内のフラグによりリードオンリーや実行オンリーの属性を指定できた。実行オンリーではユーザプログラムからは読み書きできない領域を実現でき、商用サービスなどでのセキュリティを実現できた。

Multics では、外部記憶装置を使いやすくするために、データを「ファイル」という単位で管理し、多数のファイルの管理を容易にするためにディレクトリという概念を入れた「階層化ファイルシステム」をはじめて実装した。マルチユーザシステムでのセキュリティを実現するためのファイルへのアクセス制御を可能とし、アクセスを自分のみに限るか、他人も許すか、許す場合に読み出しのみ許可するか、書き込みも許可するか、といった制御を指定できた。また、実際の外部記憶へのアクセスはディレクトリを含めたファイル名を与えるだけで OS が自動的にを行い、ユーザはファイルがマップされたメモリへアクセスするだけでデータにアクセスすることができた。

ベル研で誕生した UNIX は、Multics プロジェクトに参加していたトンプソン (Ken Thompson) を中心に設計されたこともあり、TSS や階層化ファイルシステムなど多くの機能が Multics 由来だといわれている。一方で、UNIX という名が Multics の Multiplexed (多重化) の反対語である Uniplexed (単一化) からきているとおり、Multics の複雑すぎた部分を単純化し、性能向上を目指したといわれている。Multics の失敗から UNIX が生まれたという話はよくいわれるが、ハネウェルから商品化され複数のシステムが使われたことから Multics が失敗だったわけではない。

より詳しい情報は文献 11), 12), 13) を参照。

## ■6 群 - 1 編 - 3 章

### 3-5 DEC PDP-8, PDP-11, VAX

— ミニコンの代表 —

(執筆者：児玉祐悦) [2009年4月 受領]

1965年、DEC (Digital Equipment Corporation) 社は最初の商用ミニコンピュータ PDP-8 を出荷した。図 3・4 に初期型 PDP-8 の筐体の写真を示す。最初の PDP-8 は小型冷蔵庫ほどの大きさであったが、当時のメインフレームなどと比べると格段に小さいためミニコンピュータ (略してミニコン) と呼ばれた。PDP は Programmed Data Processor の略である。価格も \$18,000 と安く、その後のシリーズを含めて 5 万台以上が市販された。最初の PDP-8 はトランジスタで構成されていたが、その後、TTL 集積回路を用いた PDP-8/I などシリーズ化され小型化・低価格化されていった。



図 3・4 初期型 PDP-8 の筐体

From Smithsonian Image 90-05950 NMAH Object 1989.0521.02

Academic and non-commercial use permitted

当時のコンピュータは巨大なシステムであり会社全体で管理されるのに対して、ミニコンピュータは実際にコンピュータを操作する人たちが管理するものであった。また、この時期、入出力装置にも変化があり、紙テープやパンチカードから、よりインタラクティブな端末が使われるようになった。PDP-8 はシンプルな割込み機構が備わっており、このような入出力装置の広まりと相まって PDP-8 が広まっていった。また、PDP-8 は他の装置に組み込まれて OEM としての出荷も行われた。

処理単位となる語長は 12 ビットであり、命令語長も 12 ビットであった。命令は 8 種類と少なく、アキュムレータを用いた 1 アドレス方式を用いていた。命令数が非常に少ないため、一つの命令で複数の処理を行う命令となっている。例えば、単にメモリの内容をアキュムレータに読み込んだり書き込んだりする命令はなく、アキュムレータの内容をメモリに書き込んで、アキュムレータをクリアする DCA 命令や、メモリの内容をアキュムレータに加算する TAD 命令となっている。

初期の PDP-8 にはオペレーティングシステム (OS) はなかったが、後期にはリアルタイム OS やマルチユーザマルチタスクシステムが利用可能となった。ユーザの作成したプログラムが DECUS (DEC User Society) により収集・配布が行われ、その多くがソース付きのプログラムであり、現在のフリーソフトウェアのはしりといえる。また、DEC の製品にはハードウェアやソフトウェアの情報を詳細に記述した解説書が付いており、ユーザがそれらについて学習しやすく、独自のシステムを構築しやすかったため、大学などでも広く用いられた。

1970 年、DEC 社は 16 ビットミニコン PDP-11 を製品化した。PDP-11 の命令セットは、メモリアドレスリングの直交性が高く、命令によらずに自由にアドレスリングを指定できたので、プログラマからの支持が高かった。プログラムカウンタをアドレスリングのベースレジスタとして利用することができ、プログラムカウンタからの相対アドレスを利用できた。PDP-11 は Unibus と呼ぶ汎用バスを採用した。仕様をオープンにしたこと、I/O デバイス用制御レジスタをメモリアドレス上にマップし非常に扱いやすかったことなどから、他のメーカーからの周辺機器への参入が容易となっていた。1975 年には、プロセッサを LSI により実装した LSI-11 を製品化した。PDP-11 には DEC 社からも複数のオペレーティングシステムが提供されたが、ATT ベル研究所で開発された UNIX が教育機関には無料で提供されるようになり、当時の情報処理技術者の研究プラットフォームとして広まった。

1978 年、DEC 社は 32 ビットスーパーミニコンピュータ VAX-11/780 を製品化した。開発責任者はゴードンベル (C. Gordon Bell) であった。VAX は Virtual Address Extension の略で、PDP-11 を 32 ビットに拡張するとともに、その巨大なアドレス空間を仮想記憶として使用した。VAX は 1980 年代には非常に一般的なコンピュータとなった。それを示す例として VAX MIPS という用語がある。あるコンピュータの性能が 10 VAX MIPS であるとは、そのコンピュータの性能が VAX-11/780 の 10 倍の性能であることを表し、VAX-11/780 の性能が基準として用いられていた。

最初の VAX は TTL で実装され、その後 ECL ゲートアレイを複数チップ用いた VAX 8800 が開発されたり、シングルチップで実装した MicroVAX が開発されたりした。一つのアーキテクチャでメインフレームからワークステーションまでカバーしていたのは VAX だけであった。

その後、RISC アーキテクチャが台頭し、DEC 社も Alpha プロセッサを開発していたが、DEC 社が 1998 年コンパック社に売却され、コンパック社も 2002 年にヒューレドパッカー (HP) 社に合併された。

より詳しい情報は文献 14), 15), 16), 17) を参照。

## ■6群 - 1編 - 3章

### 3-6 Intel MCS-4

#### —最初の商用ワンチップマイクロコンピュータ—

(執筆者：安藤壽茂) [2009年4月受領]

Intel MCS-4 は、1971年11月に発売された世界初の商用マイクロコンピュータシステムである。日本のBUSICOM（ビジコム）という会社からの電卓用の専用LSIの開発依頼に対して、Intel社のMarcian Edward Hoff, Jrは、より汎用のマイクロプロセッサの開発を提案した。実際のチップの設計は、このプロジェクトのために新規に採用されたFederico Faggin（後にZilog社を創設し、Z80などを開発）が中心となり、Stanley Maserやビジコム社から派遣された嶋正利らの協力を得て作り上げられた。

Hoffのコンセプトは、CPUアーキテクチャは汎用的で簡素なものとし、Intelの得意なRAM、ROMを使ってプログラムで電卓機能を実現するというものであり、4004 CPUに加えて、プログラムを格納する4001 ROM、データを格納する4002 RAM、そして時分割でI/Oポートを拡張する4003チップの計4種のチップセットが開発され、これらをまとめてMCS-4 Micro Computer Setと呼んだ。

4004 CPUは4ビット単位でデータを処理する4ビットプロセッサであり、各種演算命令やロード、ストア命令、I/O命令などを含む45種の命令をサポートしている。

製造プロセスは最小寸法が10 $\mu\text{m}$ のシリコンゲートPMOSプロセスが使われており、チップサイズは約2.8 mm $\times$ 3.8 mm、トランジスタ数は約2300である。そして、-15 Vの電源で30 mA (type)の電流で動作した。図3・5にMCS-4のデータシート<sup>18)</sup>の表紙を示す。

クロック周波数は約740 kHzであるが、1命令の処理には8サイクルを必要とする構成であり、1命令の処理時間は10.8 $\mu\text{s}$ であった。また、処理性能の例として、MCS-4マニュアル<sup>19)</sup>には8桁の10進数の加算の計算時間は850 $\mu\text{s}$ であると書かれている。

MCS-4の各チップは16ピンという端子の少ないパッケージを使用しており、ピンの制約から4ビット幅のアドレス、データ共通バスを介して相互接続され、4004 CPUに最大16個の4001 ROM（合計32768ビット）と最大16個の4002 RAM（合計5120ビット）を接続することができる。また、これらのROM、RAMチップはI/Oポートを内蔵しているが、これらのI/Oポートだけでは不足する場合は、4003チップを使ってI/Oポート数を拡張することができるようになっている。

この4001~4004チップセットはビジコム社の費用で開発されたものであり、当初は、ビジコムの専用品であったが、その後、電卓の価格競争の激化に伴うビジコムからの値下げ要求と引き換えにIntel社は外部への販売権を獲得し、MCS-4として発売されることになった。

4004 CPUは簡単なCPUであったが、続いて8ビット単位で処理を行う8008、8080、Z80、6800プロセッサなどが開発され、今日の64ビットアーキテクチャのプロセッサに連なるマイクロプロセッサの発展の基礎となったという点で歴史的に重要なコンピュータである。

より詳しい情報は文献20)、21)を参照。



# MCS-4 MICRO COMPUTER SET

MCS-4 MICRO COMPUTER SET

NOVEMBER 1971

- Microprogrammable General Purpose Computer Set
- 4-Bit Parallel CPU With 45 Instructions
- Instruction Set Includes Conditional Branching, Jump to Subroutine and Indirect Fetching
- Binary and Decimal Arithmetic Modes
- Addition of Two 8-Digit Numbers in 850 Microseconds
- 2-Phase Dynamic Operation

- 10.8 Microsecond Instruction Cycle
- Easy Expansion—One CPU can Directly Drive up to 32,768 Bits of ROM and up to 5120 Bits of RAM
- Unlimited Number of Output Lines
- Single Power Supply Operation ( $V_{DD} = -15$  Volts)
- Packaged in 16-Pin Dual In-Line Configuration

The MCS-4 is a microprogrammable computer set designed for applications such as test systems, peripherals, terminals, billing machines, measuring systems, numeric and process control. The 4004 CPU, 4003 SR, and 4002 RAM are standard building blocks. The 4001 ROM contains the custom microprogram and is implemented using a metal mask according to customer specifications.

MCS-4 systems interface easily with switches, keyboards, displays, teletypewriters, printers, readers, A-D converters and other popular peripherals.

A system built with the MCS-4 micro computer set can have up to 4K x 8 bit ROM words, 1280 x 4 bit RAM characters and 128 I/O lines without requiring any interface logic. By adding a few simple gates the MCS-4 can have up to 48 RAM and ROM packages in any combination, and 128 I/O lines. The minimum system configuration consists of one CPU and one 256 x 8 bit ROM.

The MCS-4 has a very powerful instruction set that allows both binary and decimal arithmetic. It includes conditional branching, jump to subroutine, and provides for the efficient use of ROM look-up tables by indirect fetching.

The Intel MCS-4 micro computer set (4001/2/3/4) is fabricated with Silicon Gate Technology. This low threshold technology allows the design and production of higher performance MOS circuits and provides a higher functional density on a monolithic chip than conventional MOS technologies.

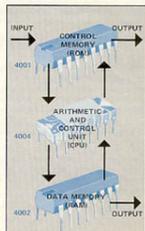


図 3・5 MCS-4 データシート<sup>18)</sup>の表紙

(© 1998-2008 Smithsonian Institution  
Academic and non-commercial use permitted)

## ■6群 - 1編 - 3章

## 3-7 Xerox Alto

ー初めてのパーソナルコンピューター

(執筆者：児玉祐悦) [2009年4月 受領]

1973年, Xerox パロアルト研究所でワークステーション Alto が開発された. Alto はアラン・ケイ (Alan Kay, 2003年チューリング賞) の提唱したパーソナルコンピュータ「ダイナブック」という概念に基づくものであった. ケイの他にもサッカー (C. P. Thacker), マックライト (E. M. McCreight), ランプソン (B. W. Lampson, 1992年チューリング賞) らが開発に携わった.

当時コンピュータは多くのユーザが共同利用することを前提としたものだったが, Alto はユーザが1台のマシンを占有する「パーソナルコンピュータ」の理想を具現化した最初のコンピュータであった. Alto は市販されなかったが, 1500台以上が製造され Xerox 社内で研究用に使用されるとともに, 一部は大学などに寄贈され, その後のパーソナルコンピュータやワークステーションに多大な影響を与えた.

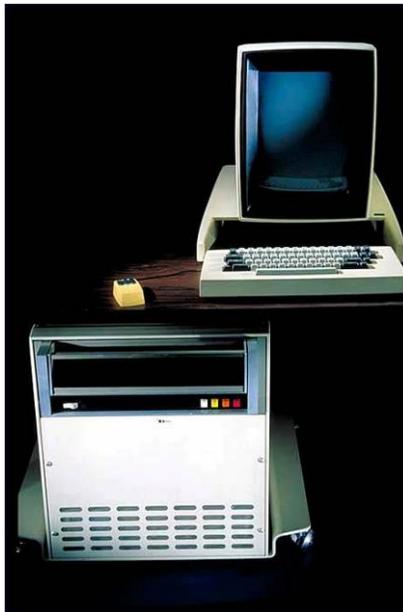


図 3・6 Xerox Alto CPU

From Smithsonian Image: 90-2234

NMAH Object 1989.0432.01.1

Academic and non-commercial use permitted

Alto はラスタスキャン型ディスプレイ (テレビと同様な方式)、3 ボタンマウス、2.5 MB のカートリッジディスク、Ethernet インタフェース、マイクロプログラム制御のプロセッサ、64 K 語 (1 語 : 16 ビット) の主記憶装置から構成されていた。図 3・6 にコンソールと筐体の写真を示す。ユーザとのインタラクションを重視した設計になっており、数値計算に必要な浮動小数点演算器はなかった。マイクロ命令は 32 ビットの垂直型であり、5.88 MHz のクロックで実行されていた。また、高級言語 BCPL、Smalltalk、Lisp、Mesa などのための機械命令セットのエミュレータがそれぞれ用意されていた。BCPL はシステム記述言語として使用された。

ディスプレイはラスタスキャン型で、606×808 の解像度がありモノクロの縦長画面であった。表示すべき情報はすべてビット単位に展開し (例えば文字 A は 5×7 の 2 次元ドットパターンなどに展開)、これを主記憶装置に格納していた。このように、画面の 1 画素がメモリ上の 1 ビットに対応させた方式をビットマップ方式と呼ぶ。当時はキャラクタ単位で表示される文字端末か、ベクタ方式の線画を描くグラフィックディスプレイが普通で、ビットマップ方式のディスプレイは初めてだった。ビット単位で自由に描画を制御できることで表現の自由度が飛躍的に向上し、「WYSIWYG」(What you see is what you get) 型のグラフィックエディタなど新しいアプリケーションが創出された。ビットマップのディスプレイへの読み出しは三つのマイクロプログラムレベルのタスクでなされていた。データの転送には平均 15 Mbits/s 程度のメモリバンド幅が必要である。また、画面のスクロールなどの操作は個々のビットマップへの操作 (スクロールはビットマップの移動) として実行された。これは BitBlt (bit boundary block transfer) あるいは RasterOp と呼ばれ、マイクロプログラムで制御されていた。フルスクリーンで 30.7 K 語 (1 語 : 16 ビット) の情報が必要であり、そのまま扱おうと 64 K 語の主記憶装置の約半分を占めることになる。そのため、スクリーン上の情報の存在する長方形領域のみをポインタ結合して、情報のない領域は実体のない空の長方形領域として扱うことにより、必要な記憶容量を減らしていた。

Alto で実装されたオブジェクト指向高級言語である Smalltalk では、開発環境としてビットマップディスプレイを利用したウィンドウシステムが開発されていた。そのほか、WYSIWYG 型のエディタなどで、メニュー操作やマウスによるポイントなどが実現されており、現在のウィンドウシステムの GUI (グラフィックユーザインタフェース) の原型を作ったコンピュータとして Alto は評価されている。

Alto のもう一つの特徴は、それらをつなぐための LAN 技術として、イーサネットが開発されたことである。イーサネット開発の中心となったのは、メトカーフ (R. M. Metcalfe) とボグズ (D. R. Boggs) であった。安価で高性能なネットワークを実現するため、ハワイ大学で研究されていた全端末に同報で信号を送る ALOHA system という無線通信のしくみを応用し、1 本の同軸ケーブルだけで多くの端末が通信する方式を考案した。ただし、Alto に使われたイーサネットは、1983 年に標準化された 10 Mbps の通信速度をもつ IEEE 802.3 の仕様と大きく異なっていたため、「実験イーサネット」(Experimental Ethernet) と呼ばれることとなった。このときの通信速度は 2.94 Mbit/s であったが、これはプロセッサのクロック (5.88 MHz) を 2 分周してそのまま利用したためであった。

より詳しい情報は文献 22), 23) を参照。

## ■6群 - 1編 - 3章

### 3-8 Cray-1

#### —商業的に成功した最初のベクトル型スーパーコンピューター—

(執筆者：飯塚 肇) [2009年4月 受領]

1975年に発表(最初の納入は1976年)され、商業的に成功した(変種も含めて約65台が売れた)最初のベクトル型スーパーコンピュータ。ベクトルレジスタを採用したそのアーキテクチャ(後述)は、その後のベクトル型スーパーコンピュータの標準となり、現在も続いている。

アーキテクト： クレイ (Seymour Cray, 1925-1996; 1989年エッカート・モークリ賞)

特徴： クレイはこれより前、CDCにおいて、極めて意欲的なスーパーコンピュータ CDC 8600を開発していたが、CDCの財政的問題でそれが中止となったために、新しい会社を作って、Cray-1の開発を開始した。そのため設計は8600よりも保守的であったが、それが商業的成功をもたらす結果になった。とはいえ、このマシンには、次のような“最初”の技術が取り入れられており、その多くはその後のコンピュータにも大きな影響を与えた。

- (1) 市販マシンに初めて、エミッタ結合論理回路 (ECL) を採用した。ECLは消費電力こそ大きい(マシン全体の消費電力は130 kW)ものの、数々の優れた特徴を有し、特にその高速性は、12.5 nsという当時のどのマシンよりもずっと小さいサイクル時間の達成(現在のCMOSがこれを越えたのは1990年代になってからである)に寄与した。
- (2) 8個の64要素ベクトルレジスタを導入し、ベクトル演算をこれらベクトルレジスタ間で行うようにした。Cray-1以前のベクトルマシンは、オペランドをメモリ内に格納していたので起動時間(最初の要素の演算結果が得られるまでの時間)が大きく、非常に長いベクトルでないとベクトル演算による高速化の効果がなかったが、これによって、比較的短いベクトルでも十分な高速化が得られようになった。
- (3) ベクトル演算器間で操作のチェイニングを行う機構を導入した。それまでのマシンでは、前のベクトル操作がすべてデスティネーションレジスタに格納されるまで次の操作を起動できなかったが、この機構によって最初の要素の結果が得られた段階で次のベクトル操作を開始できるようになり、複数の連続したベクトル操作のスループットが大きく向上した。
- (4) それまでのマシンより非常に大きいプログラマが利用できるレジスタファイルを用意した。
- (5) 当時としては、最高速のスカラ演算ユニットを用意して、ベクトル化できない操作が全体の性能を落とす効果を最小限に抑えた。このスカラとベクトル性能のバランスの良さが、Cray-1成功の大きな要因とされている。
- (6) 電氣的/機械的な実装を工夫した。すなわち、(i)フリップフロップの代わりにラッチを使用、(ii)ラックを270°の円状(C型)に配置(図3・7の写真参照)することによる配線の短縮、(iii)モジュールのエッジの液体フロンによる冷却などである。

性能： (1)～(6)の結果、当時のほかのマシンに比べて、はるかに大きい性能を達成した。

100×100のLINPAC性能は、27 MFLOPSと報告されているが、この値はこのマシンの祖先でサイクル時間が3倍のCDC7600の約8倍であるから、おおざっぱに言って、アーキテ

クチャの工夫によって2.7倍程度の性能向上を得たということになる。  
より詳しい情報はアーキテクチャについては文献 24), 実装については文献 25) を参照。



**図 3・7** Cray-1 の筐体  
Computer History Museum の許可を得て掲載

## ■6群 - 1編 - 3章

### 3-9 IBM 801

#### —RISC アーキテクチャの祖—

(執筆著：飯塚 肇) [2009年4月 受領]

1975年10月からIBMワトソン (T. J. Watson) 研究センターで開発が行われたミニコンピュータ。801は商品番号ではなく、このプロジェクトが行われたビルディングの番号から命名された。設計哲学にいわゆるRISCアーキテクチャを採用した最初のマシン (CDC 6600などの大型コンピュータを除く) とされる。1977年に完成したプロトタイプは、MECL 10KのICを利用していたが、801の設計思想は、その後開発された1チップマイクロプロセッサのRISC IやMIPS (スタンフォード大学) に大きな影響を与え、それらを通じて現代マイクロプロセッサのアーキテクチャのもとを築くとともに、IBM社のPowerアーキテクチャに引き継がれている。

アーキテクト： コック (John Cock, 1987年チューリング賞)

特徴： IBM 801の重要な貢献は次のような点である。

- (1) 命令セットを、(当時一般的であったアセンブリプログラミング指向でなく) コンパイラの必要性に合ったものとした。すなわち、全命令を32ビット固定長とし、簡単な基本的操作のみを用意し、メモリオペランドは一旦レジスタにロードしてから操作を行うロードストアアーキテクチャを採用した。
- (2) (1)の結果として、実装はハードウェアで容易に行うことができ(クロックも高速化できる)、レジスタ数を多く確保でき、パイプライン実行も簡単なハードウェアで可能になった。更に、1チッププロセッサでは、制御ロジックが減少した分を内部キャッシュの増量に用いて、メモリレイテンシを小さくすることが可能になり、全体としてそれまでのCISCアプローチよりも高速のプロセッサの実現が可能となった。

より詳しい情報は文献26)を参照。

1チップRISCアーキテクチャ： IBM 801は、上記のように1チップマイクロプロセッサを指向していたわけではなく、RISCという名称の起源でもない。1チップマイクロプロセッサのためのアーキテクチャに貢献したのは、それより少し後1980年代初めに行われたRISC IやMIPSの研究である。そこで、これらについても簡単に触れておく。

- (1) RISC I：カリフォルニア大学バークレイ校、パターソン (David A. Patterson, 2008年エッカート・モークリ賞)

1980年に研究が開始されたプロジェクトで、当時の集積度で高性能な1チップマイクロプロセッサのアーキテクチャを開発することを目標とした。IBM 801と同じ固定長単純命令セットアーキテクチャを採用したほか、遅延分岐、レジスタウインドウ方式を採用して、それまでのマイクロプログラム制御方式のマシンに勝る性能が得られることを立証した。チップ名のRISC (Reduced Instruction Set Computer) は、このタイプのアーキテクチャの一般名称となり、それまでCDC6600などの高価なマシンにしか用いられていなかったパイプライン制御を高性能マシンのメインストリームとした。なお、プロジェクトの成果は、その後サンマイクロシステムズ社のSPARCアーキテクチャに引き継がれた。

初期の情報については文献27)を、汎用プロセッサの設計に対するRISCアプローチについ

ては文献 28)を参照.

- (2) MIPS : スタンフォード大学, ヘネシー (John L. Hennessy, 2001年エッカート・モークリ賞)

RISC I とほぼ同時期に同じ思想で開発されたマイクロプロセッサで, ソフトウェアの負担をより大きくする代わりに, ハードウェアを簡単にして高速実行を可能にしたことが特徴といえる. パイプラインのハードウェアによるインタロックを行わない (MIPS は Microprocessor without Interlocked Pipeline Stages を意味する. ただし, 集積度の増加に伴ってインタロックは後で追加された), TLB 内容のソフトウェアによる設定などはその特徴的例である. MIPS はその後商品化されて, SGI のワークステーションなどに使用され, 現在でも, ゲーム機などの組み込みシステムのコア用として使用されている.

初期の情報については文献 29)を参照.

## ■6群 - 1編 - 3章

### 3-10 IBM PC

— Intel 8088 を採用し、その後のパソコンの発展の基礎を築いたマシン —

(執筆者：安藤壽茂) [2009年4月 受領]

低価格のパソコンの可能性に着目し、IBM でもパソコンの開発が必要と考えた William Lowe は 12 人のチームを作り 2 週間で開発から販売までの詳細プランを作成し、経営委員会の承認を受けて、後に IBM PC となるシステムのプロトタイプの開発を開始した。開発責任者としてチームを指揮したのは、Lowe の部下であった Philip (Don) Estridge である。チームは開発期間を短縮するため、できるだけ外部の既製品を調達するという方針で開発を始め、プロトタイプ completion 後、約 1 年という IBM としては異例の短期間で IBM 5150 を開発し、発売に漕ぎつけた。図 3・8 に IBM 5150 システムの写真を示す。狭義の IBM PC は、1981 年 8 月 12 日に発売されたこの IBM 5150 を指す。



図 3・8 IBM 5150 システム

2 台のフロッピードライブとモノクロモニタ、プリンタを含むこのシステムは \$4,385

(Reprint Courtesy of International Business Machines Corporation, copyright ©

International Business Machines)

Intel 8088 は、内部的には 8086 と同じで 16 ビットアーキテクチャであるが、外部とのインタフェースは 8 ビット幅とし、それまでの 8 ビット CPU を使うシステムと同じ周辺チップを使用する安価なハードウェア構成を可能とした CPU チップである。IBM 5150 は、クロック周波数が 4.77 MHz の 8088 CPU、16 KB の RAM と BIOS などを格納する 40 KB の ROM を搭載している。また、5 個の拡張スロットを備えており、ここにメモリや I/O カードを搭載することができ、64 KB のメモリボードを 4 枚搭載することにより最大 256 KB まで RAM 領域を拡張することができた。

OS に関しては、当時、8 ビット OS の業界標準であった CP/M をもつ Digital Research 社に開発を打診したが話がまとまらず、Microsoft 社からの MS-BASIC と併せて OS も供給するという提案を受け入れることになった。しかし、当時 Microsoft は 8086/8088 用の OS をもっておらず、Seattle Computer Products 社の 86-DOS (通称、QDOS : Quick and Dirty OS) の権利を買い取り、この OS を開発した Tim Paterson を雇って作業にあたらせ、IBM 向けの PC-DOS

として完成させた。この OS は共同開発という位置づけであり、IBM、Microsoft の両社が版權を保有している。

IBM 5150 の基本モデルは ROM に格納された BASIC を使用するが、フロッピードライブをもつシステムでは、OS としてこの PC-DOS を使用した。また、CP/M-86 や UCSD p-system も使用可能であった。

カセットテーププレーヤを記憶装置とし、ディスプレイとして TV を接続して使用する基本モデルは\$1,565 であったが、これはあまり売れず、5-1/4 インチで 160 KB の容量のフロッピードライブ 1 台とモノクロディスプレイを含む\$3,005 のモデルが多く販売された。しかし、この価格は家庭用としては高過ぎ主な用途はビジネス用となったが、発売後 3 年間で 25 万台以上を販売し、ビジネス用のコンピュータとしては予想以上の大成功を収めた。

IBM は、5150 のハードウェア回路図や ROM BIOS のソースコードをテクニカルマニュアルに記載して公開した。また、Microsoft は、この OS を MS-DOS として IBM 以外の会社にも販売したため、容易にコピーを作ることが可能となり、いわゆる互換機を作るメーカーが多く出現した。

IBM PC は表計算の VisiCalc、Peachtree 社の経理ソフト、ワープロの EasyWriter などのアプリケーションを使用することができたが、アプリケーションの中には、高い操作性を実現するため OS や BIOS をバイパスして直接ビデオメモリにアクセスを行うものが多く出現した。このため、ハードウェアに非互換があるとアプリケーションが動作しないという問題が発生し、IBM PC の出現から数年後には、Apple 社の PC などの一部の例外を除いて、ほとんどの PC が IBM PC 互換になってしまった。

IBM PC のハードウェアは、CPU 以外は従来の 8 ビットシステムと大差なく、PC-DOS も機能的には CP/M のクローンであり、技術的には目新しい点はない。しかし、IBM PC は、業界標準として確立し、それまでの機種ごとに製品のカスタマイズが必要な環境とは異なり、ハード、ソフトともに一種類の製品開発でどの機種用にも販売できる大きな共通マーケットを作ったことにより、今日の PC の隆盛の源となった。

巨大な PC 市場を作った IBM であるが、競争の激化による採算の悪化に伴い、2005 年に PC ビジネスを中国の聯想集団 (Lenovo Group) に売り渡し、この業界から撤退してしまった。また、PC ビジネスを立ち上げた Don Estridge と夫人の Mary Ann が、1985 年 8 月 2 日のデルタ航空 191 便の墜落事故で死亡したのは悲劇である。

より詳しい情報は文献 30)、31) を参照。

## ■6群 - 1編 - 3章

### 3-11 地球シミュレータ

—米国を驚愕させた和製スーパーコンピュータの捲土重来—

(執筆者：谷 啓二) [2009年4月 受領]

かつて、航空宇宙技術研究所で“数値風洞”を開発した実績をもつ三好甫が発案、主導し、文部科学省傘下の3機関（宇宙航空研究開発機構，日本原子力研究開発機構，海洋研究開発機構）が共同で開発（1997～2002年）した，理論性能40 TFLOPSのベクトル型並列コンピュータシステム。受注，製作メーカはNEC。ニューヨークタイムスは，かつて米国が人工衛星開発でソ連のスプートニクに先を越されたショックを想起し，“コンピュータニク”の造語をもって，その完成を報じた。以後，LINPACK ベンチマークテストのランキング（Top 500）において，5期，2年半にわたり世界トップの座を占めた。

アーキテクチャは，8台の要素計算機（ベクトル計算機）を共有メモリで密結合して計算ノードを構成し，その計算ノード640台をネットワーク（単段クロスバースイッチ）で結合した，システム全体では分散メモリ並列コンピュータである。図3・9に全景の写真を示す。



図3・9 地球シミュレータ全景

計算ノードの性能，主記憶量はそれぞれ64 GFLOPS，16 GBであり，システムの総処理性能，総主記憶容量はそれぞれ40 TFLOPS，10 TBである。システムの実効性能を左右するノード当たりのメモリスループットは256 GB/sである。

開発に5年の長期間を要したため，完成時点で予測された最先端の半導体技術を見越し，先端的な技術開発が数多く行われた。採用されたCMOSテクノロジーは150 nm。この技術によるチップ上のトランジスタの高集積化と配線の微細化により，ベクトルプロセッサにとってはエポックメイキングな開発がなされた。当時，NECではベクトルプロセッサを30～40個のLSIをボード上に配線し組み上げていたが，地球シミュレータで初めて，僅か2 cm角の1チップ上にすべてのLSI機能と配線を集積してしまう1チップ化に成功した。これにより，ベクトルプロセッサがマイクロプロセッサと半導体技術において初めて同じレベルに到達することとなった。この1チッププロセッサ上には，ベクトルユニット，スカラユニッ

ト、プロセッサネットワークユニットなどが実装されている。因みに、クロック周波数は 500 MHz であるが、ベクトルユニット部のベクトル演算器 4 セットは 1 GHz で動作することから、論理上のベクトル演算器セット数は 8 となっており、その理論性能は 8 GFLOPS である。

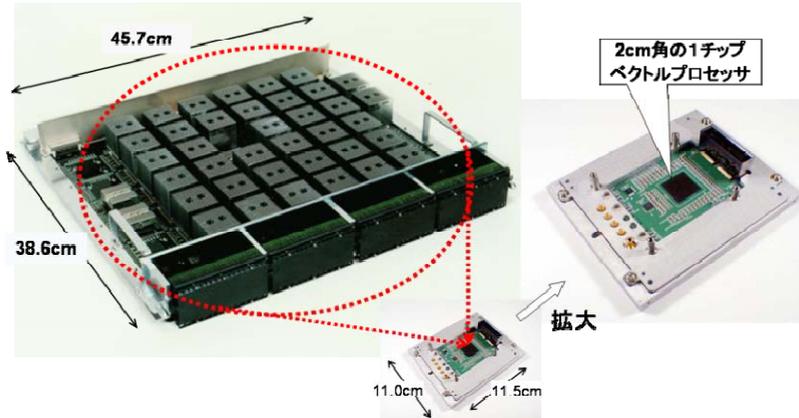


図 3・10 地球シミュレータにおけるベクトルプロセッサの 1 チップ化

図 3・10 に従来の数十個の LSI チップから構成されていたベクトルプロセッサのボード (NEC 社製 SX-4) と地球シミュレータの 1 チップ・ベクトルプロセッサの比較を示す。この 1 チップ化による省コスト化、省エネルギー化、省スペース化で、地球シミュレータの巨大システムの実現が可能となった。1 チップ化以外にも、以下のような当時としては先端的な数々の技術開発がなされた。

- ・消費電力低減のための銅配線の採用 (NEC としては初めて。それまではアルミニウム。それでも 1 チップの消費電力は 140 W 程度になった)。
- ・主記憶として超高速フルパイプラインメモリ (FPLRAM) の採用。
- ・大型ビルドアップ基板開発による高密度実装。
- ・超大型並列分散システムでのプログラム開発環境、運用管理ソフトウェアの開発。

1997 年から、文部科学省 (旧科学技術庁) は、基礎科学研究、観測、計算機シミュレーションの三位一体で、当時注目されはじめていた地球環境変動予測研究を推進するプロジェクトを立ち上げた。地球シミュレータは、その必要ツールと位置付けられ、次世代の計算地球科学に必要なコンピュータとして、「標準的な大気大循環モデルで、実効 5 TFLOPS」という開発目標が設定された。これは、それまでの気象シミュレーションの空間分解能を 1 桁改善するため、当時気象分野で標準的に用いられていた CRAY C90 の 1000 倍の実効性能が求められたことによる。更に、実効効率として 15%前後が想定されたため、システムのピーク性能として 40 TFLOPS が設定された。しかし、その後、大気大循環モデルの地球シミュレータへの最適化が進み、最終的には、ピーク性能比約 65%、実効性能 26.58 TFLOPS を実現し、2002 年の Gordon Bell 賞を獲得した。この地球シミュレータの高性能により、気象分野に留まらず、同年には、乱流直接数値シミュレーション、翌 2003 年には、米国アラスカで発生

した地震波伝播の高精度シミュレーションで、2004年には地磁気ダイナモの高速シミュレーションでと、連続して Gordon Bell 賞を獲得している。また、メーカでの地球シミュレータ開発が評価され、当時 NEC の開発責任者であった渡辺貞（現（2009年）理化学研究所次世代スーパーコンピュータ開発実施本部プロジェクトリーダー）が2006年度に Seymour Cray 賞を受賞した。

一方、開発目標には設定されなかったが、LINPACK ベンチマークテストでは、35.61 TFLOPS（ピーク性能比 87.2%）を記録し、2位の ASCI White の 7.226 TFLOPS（ピーク性能比 58.8%）に大きく水を開けた。これは、米国がスーパーコンピュータでの世界一の座の奪還に向けて、MPP 路線から IBM の BG（Blue Gene）へと、より専用性の高いシステムへの方向転換を図るきっかけとなった。

その後、2004年に米国リバモア国立研究所に設置された IBM の BG/L に LINPACK の Top 500 ランキングで世界一の座を明け渡した。しかし、プロジェクトリーダーの三好甫の「アプリで5年以上トップの座をキープできる」という生前の予測は見事に当たり、現時点（2009年）でもなお、流体系の計算において世界トップレベルの性能を有している。その応用は、地球環境分野に留まらず、航空、原子力、宇宙、バイオ、ナノテク、ものづくり（自動車）などへと拡がり、理化学研究所の次期システム開発を牽引する役割を果たした。

より詳しい情報は文献 32), 33), 34), 35), 36), 37) を参照。

## ■6群 - 1編 - 3章

## 3-12 アポロ誘導コンピュータ

---アポロ宇宙船に搭載された組み込みシステムの原型---

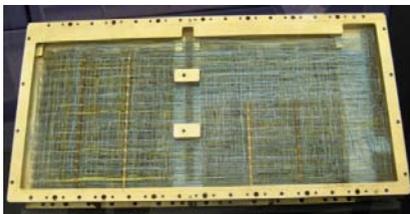
(執筆著：前島英雄) [2009年4月 受領]

アポロ誘導コンピュータは、アポロ宇宙船の司令船と月着陸船に搭載された今日の組み込みシステムの原型といえる超小型コンピュータで、AGC (Apollo Guidance Computer) と呼ばれた。AGC はアポロ宇宙船の全航行機能を自動制御するとともに、ユーザインタフェースにより宇宙飛行士がリアルタイムで飛行情報を確認・修正することができる。アポロ計画に向けて MIT Instrumentation Laboratory により、1962年に Block I の開発が開始され、1966年には Block II が開発された。

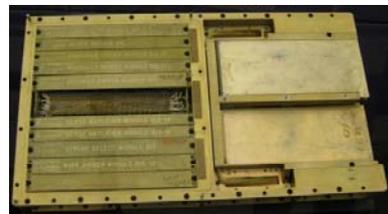
AGC Block II は1969年7月20日に月着陸を果たしたアポロ11号に搭載されたが、24×12.5×6インチ (およそ61×33×15 cm) のサイズ、65ポンド (29.5 kg) の重量で、DC電源28Vを使用し消費電力は70Wである。

また、コンピュータ本体は24個のモジュールからなる二つのトレイ A, B から構成され、トレイ内の各モジュールは72ピンのコネクタを有し、フラットパックのIC (2個の3入力 NOR ゲートを集積) 60個をマルチレイヤプリント基板に実装し、モジュール間はワイヤラップという配線方式で結線し、1.024 MHz の4相クロックで動作した。トレイ A は論理回路、インタフェース、電源から、トレイ B はメモリ、メモリドライバ、アナログアラームデバイス、クロック回路から構成され、総論理ゲート数は5600ゲートであった。図3・11にAGCの写真を示す。

更に、DSKY (Display and Keyboard units) と呼ぶユーザインタフェースを有しており、司令船に2台、月着陸船に1台が搭載された。このDSKYは高電圧エレクトロルミネッセンスのディスプレイと電卓型のキーボードからなり、この分野のコンピュータとしては世界初のデジタル制御パネルであった。また、メモリとしては書き換え可能な2Kワードの磁気コアメモリと、36Kワードのコアロープメモリと呼ぶ固定メモリが実装され、11.7 μs のサイクル時間で動作し、加算命令は42735回/秒の性能であった。



トレイ A のワイヤラップ面



トレイ B のモジュール面

図3・11 AGC 本体 (Computer History Museum の許可を得て掲載)

AGC の1ワードは実装規模、演算速度を考慮し16ビット長で、14ビットのデータ部と各1ビットの符号ビット、パリティビット (奇数パリティ) で構成された。また、A (アキュム

レータ), Z (プログラムカウンタ), Q (除算の剰余を格納), L (乗算の下位ビットを格納) を基本として 7 個のレジスタを有しており, 命令フォーマットは, 3 ビットがオペレーションコード, 12 ビットがアドレスとなっている. TC (無条件分岐), CCS (条件分岐), INDEX (インデックス加算), XCH (A レジスタとメモリ内容の交換), CS (メモリ内容の 1 の補数を A レジスタに格納), TS (A レジスタの内容をメモリに格納), AD (加算), MASK (論理積) の 8 個を基本命令として有し, 拡張コード命令として SU (減算), MP (乗算), DV (除算), RESUME (割込みからの復帰) などの 34 命令がある.

また, 慣性飛行中で AGC を使わないときに消費電力を抑えるためのスタンバイモードが備わっている. このモードは Standby Allowed スイッチにより移行し, AGC の電源を切り, クロックだけを発生させるものである.

以上のように, AGC は今日の組込み向けコンピュータとして備えている数々の機能や仕様を既にも実現しており, 歴史的に重要である.

より詳しい情報は文献 38), 39) を参照.

#### ■参考文献

- 1) Robart. S. Barton, "A New Approach to the Functional Design of a Digital Computer," AFIP Conference Proceedings, Vol.19, pp.393-396, 1961.
- 2) Elliot I. Organick, "Computer System Organization: The B5700/B6700 Series," Academic Press, Inc., 1973.
- 3) 土井範久 (訳), "計算機システムの構造-バロース大型計算機シリーズ-," 共立出版, 1978.
- 4) G. Bell, "A Seymour Cray Perspective",  
<http://research.microsoft.com/en-us/um/people/gbell/craytalk/index.htm>
- 5) J. E. Thornton, "Design of a Computer The Control Data 6600," Scott, Foresman and Company, 1970.
- 6) G. M. Amdahl, G. A. Blaauw and F. P. Brooks, Jr, "Architecture of the IBM System/360," IBM Journal of Research and Development, vol.8, pp.87-101, Apr. 1964.
- 7) D. W. Anderson, F. J. Sparacio and R. M. Tomasulo, "Machine philosophy and instruction handling," pp.8-24, IBM Journal of Research and Development, vol.11, Jan. 1967
- 8) R. M. Tomasulo, "An efficient algorithm for exploiting multiple arithmetic units," pp.25-33, IBM Journal of Research and Development, vol.11, Jan. 1967.
- 9) C. J. Conti, D. H. Gibson and S. H. Pitkovsky, "Structural aspects of the system/360 Model 85, Part I: The cache," IBM Systems Journal, vol.7, no.1, pp.2-14, 1968.
- 10) J. S. Liptay, "Structural aspects of the system/360 Model 85, Part II: General organization," IBM Systems Journal, vol.7, no.1, pp.15-21, 1968.
- 11) F. J. Corbató, V. A. Vyssotsky, "Introduction and Overview of the Multics System," 1965 Fall Joint Computer Conference, AFIPS, 1965.
- 12) Multics History, <http://www.multicians.org/history.html>
- 13) E. I. Organick, "The Multics Systems: An Examination of Its structure," MIT Press, 1972, (菊池・佐々木訳, "MULTICS システム システムのアーキテクチャとソフトウェア," 共立出版, 1973.)
- 14) "The Digital Equipment Corporation PDP-8," 1965 - Computer History Collection from the Smithsonian
- 15) G. Bell, R. Cady, H. McFarland, B. DeLagi, J. O'Laughlin, R. Noonan, W. Wulf, "A New Architecture for Mini-Computers: The DEC PDP-11," Proc. Spring Joint Computer Conference, AFIPS, pp.657-675, 1970.
- 16) C. Gordon Bell, J. Craig Mudge, and John E. McNamara, "Computer Engineering - A DEC View of Hardware Systems Design," Digital Press, 1978.
- 17) 勝 男厚, "パソコンの礎になった DEC 社のコンピュータの変遷," トラ技コンピュータ 1995 年 7 月号, CQ 出版, <http://h50146.www5.hp.com/products/software/oe/openvms/history/digital/pdpvax/index.html>
- 18) MCS-4 データシート <http://smithsonianchips.si.edu/ice/4004.htm>
- 19) MCS-4 マニュアル <http://download.intel.com/museum/archives/pdf/msc4.pdf>

- 20) W. Aspray, "The Intel 4004 Microprocessor: What Constituted Invention?," IEEE Annals of the History of Computing, Vol. 19, No. 3, pp. 4-15, Jul-Sep. 1997
- 21) R. Noyce, M. Hoff, "A History of Microprocessor Development at Intel," IEEE Micro, Vol.1 Issue. 1, pp. 8-21, Feb. 1981
- 22) C. P. Thacker, "Personal Distributed Computing: The Alto and Ethernet Hardware," Proceedings of the ACM Conference on The history of personal workstations, pp.87-100, 1986.
- 23) C. P. Thacker, E. M. McCreight, B. W. Lampson, R. F. Sproull, D. R. Boggs, "Alto: A Personal Computer," in *Computer Structures: Principles and Examples*, D. Sieworek, C. G. Bell, A. Newell, McGraw-Hill: New York, 1982.
- 24) R. M. Russel, "The CRAY-1 computer system," CACM vol. 21, no. 1, pp.63-72, 1978.
- 25) J. S. Koledzey, "CRAY-1 computer technology," IEEE Trans. CHMT, vol. 4, no. 2, pp.181-187, June, 1981.
- 26) G. Radin: The 801 minicomputer, ASPLOS, pp.39-47, Mar. 1982.
- 27) D. A. Patterson and D. R. Ditzel, "The case for the reduced instruction set computer," ACM SIGARCH Computer Architecture News, vol.8, pp.25-33, Oct. 1980.
- 28) D. A. Patterson and C. H. Sequin, "RISC I: A reduced instruction set VLSI computer," ISCA, vol.8, pp.443-459, May 1981.
- 29) J. L. Hennessy, N. Jouppi, F. Baskett, T. Cross and J. Gill, "Hardware/software tradeoffs for increased performance," ASPLOS, pp.2-11, Mar. 1982.
- 30) "The Birth of the IBM PC", IBM アーカイブス  
[http://www-03.ibm.com/ibm/history/exhibits/pc25/pc25\\_birth.html](http://www-03.ibm.com/ibm/history/exhibits/pc25/pc25_birth.html)
- 31) P. Freiberger, M. Swaine, "Fire of the Valley: The Making of the Personal Computer (Second Edition)," McGraw-Hill, ISBN 0-07-135892-7, 2000
- 32) 谷 啓二, 横川三津夫, "地球シミュレータ計画—地球 (ガイア) との共生の指針を求めて—," 情報処理学会誌, Vol.41, No.3, pp 249-254, 2000.
- 33) 横川三津夫, 谷 啓二, "スーパーコンピュータで地球の未来を映し出す," 情報処理学会誌, Vol. 41, No.4, pp 369-374, 2000.
- 34) <http://www.jamstec.go.jp/esc/index.html>
- 35) <http://www.jamstec.go.jp/es/jp/system/arithmic.html>
- 36) <http://www.jamstec.go.jp/es/jp/system/performance.html>
- 37) 矢川元基(監修), 谷 啓二, 奥田洋司, 福井義成, 上島 豊(編著), "ペタフロップス・コンピューティング," 培風館, 2007.
- 38) Eldon C. Hall, "A Case History of the AGC Integrated Logic Circuits," E-1880, MIT Instrumentation Laboratory, 1965.
- 39) Eldon C. Hall, "Case History of the Apollo Guidance Computer," E-1970, MIT Instrumentation Laboratory, 1966.