

OS-9について

坂本眞一郎

はじめに

1970年代における技術革新がもたらした急速な情報化のうねりによって、コンピュータをめぐる状況はめまぐるしく変化してきている。特に近年のパーソナルコンピュータの発展と普及は目をみはるものがあり、社会全体に大きな影響を与えつつある。本稿では、このパーソナルコンピュータのオペレーティングシステムの中から、OS-9を選び、その特徴、機能、構造、BASIC09について述べ、さらにOSとしての評価と今後の動向について述べることにする。

1. OS-9の特徴

OS-9は、8ビットの6809系CPUで動作するオペレーティングシステムとして、アメリカのマイクロウェア社が開発したものであり、作成にあたっては、リアルタイムのOSであるUNIXを雛型として、8ビットのCPU上で動作するOSとして開発されてきた。現在では、OS-9は、16ビットのCPU68000でも使用できるように改良され、さらに32ビット機を意識して図1のようになっている。

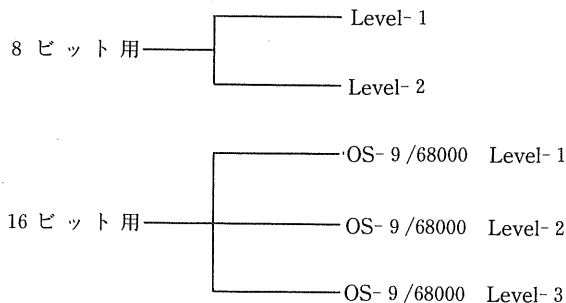


図1 OS-9の分類

このOS-9の特徴は以下の6点にまとめることができる。

(1) マルチタスク (multi task)

複数の仕事を見かけ上同時に実行することをマルチタスクと呼ぶが、これは、プログラム数に対応して、時間を見かけ上、複数のプログラムを同時に実行しているように細かく区切ることにより行う。したがって、この区切る時間を非常に短くすると、複数のプログラムが同時に実行されているようにみえることになる。具体例として、プログラムをプリンタに出力中に、別のプログラムを作成することなどがあげられる。このようなことが可能なため、マルチユーザー機能、マルチウインドウ機能、パイプライン機能が付随的に備わることになる。

(2) マルチユーザー (multi user)

上で述べたように、OS-9はマルチユーザーとして使用することも可能である。マルチユーザーとは、ひとつのCPUを複数の人間が同時に利用することができることを指し、具体例として、OS-9を動かしているパソコンを中心に、複数のパソコンを端末として接続し、複数の人間が利用することなどがあげられる。

(3) 階層ディレクトリ構造

OS-9は、ファイルを階層的に保存している。したがって、OS-9は、バイト単位の情報としてすべてのファイルを取り扱っている。このようなファイルの構造は、一般に、ツリー(tree)構造といい、ファイルを効率的に使用するうえで、きわめて有効な構造であると考えられる。

(4) モジュール構造

OS-9では、ファイルがすべてモジュール(module)構造になっている。したがって、ファイルが、一つ一つ完全に独立している。具体的には、OS-9では、メモリモジュール(memory module)という形で、メモリ上に読み込むオブジェクトを管理している。このため、プログラムをメモリのどのような場所にも自由に置くことができ、また同じモジュールを同時に複数のプログラムが共有できる構造を持つという特徴がある。

(5) マルチウインドウ

マルチウインドウとは、一つのディスプレイを分割して利用できることを指し、単独で、パソコンを利用している際に、複数のファイルの一つのディスプレイで見ることができるので、複数のファイルを同時に活用しなければならない仕事をしている場合には非常に便利である。

(6) その他

OS-9その他の特徴としては、I/Oリダイレクト機能、シェル(shell)、プログラミング可能、ROM化が可能、コマンドのパイプライン処理可能、OSがアセンブラで書かれていることがあげられる。

2. OS-9 の機能と構造

OS-9 は、以下のような機能と構造を持っている。

(1) 標準入出力 (I/O) とリダイレクト

これはプログラムを書いて出力する装置を、実行段階で指定することできる機能である。この方法には、あらかじめ出力を指定しない場合にはプリンターに出力すると決めておく、標準出力という方法と、プリンターに出力したくない場合にはその出力先を変更する装置を指定する、リダイレクトという方法がある。

標準出力に対して、標準エラー出力という概念がある。標準エラー出力とは、エラーメッセージやプロンプトなどの付随的な情報を出力するためのものである。たとえば、プログラムのリストをプリンターに出力中にエラーが発生した場合、エラーメッセージがプリンターに出力されると、プログラムのリストにエラーメッセージが混入するため、エラーに対する対処が遅れてしまう。このためディスプレイ上にエラーメッセージが出力するように設定できることを指す。

(2) パイプライン機能

パイプライン機能とは、ある特定の命令の出力と別の命令の入力を結ぶ機能である。このような、パイプラインに使用できる命令のことをフィルタと呼ぶ。よく使うフィルタとしては入力ファイルの中に大文字をすべて小文字に変換する `lower`、入力ファイルを単語ごとに切り放す機能を持つ `set`、入力ファイルの行数を数える `lc` などがある。パイプラインの基本的な考え方は、小さなプログラムを組み合わせる効果的に使用するという点である。

(3) 階層ディレクトリ

ファイルを効果的に整理、分類する際に便利なように、OS-9 は、UNIX と同様に階層ディレクトリ構造になっている。したがって、OS-9 では、階層構造によりディスク空間は管理されることになる。この考え方は、UNIX と共通している。この階層ディレクトリ構造をとることにより利用者の使いやすいうように、ファイルを並列的にも階層にもできる利点が生じる。

OS-9 の階層構造は、上位からルート (root)、中位はノード (node)、下位はリーフ (leaf) のように構成されている。上位のルートは、ルートディレクトリともよばれ、`d 1`、`d 2` というデバイス名で表現されています。中位のノードは、ディレクトリファイルともよばれるリーフ上のファイルである。リーフは、ファイルそれ自体である。具体的に、住所でこの階層構造を表現してみると下図のようになる。

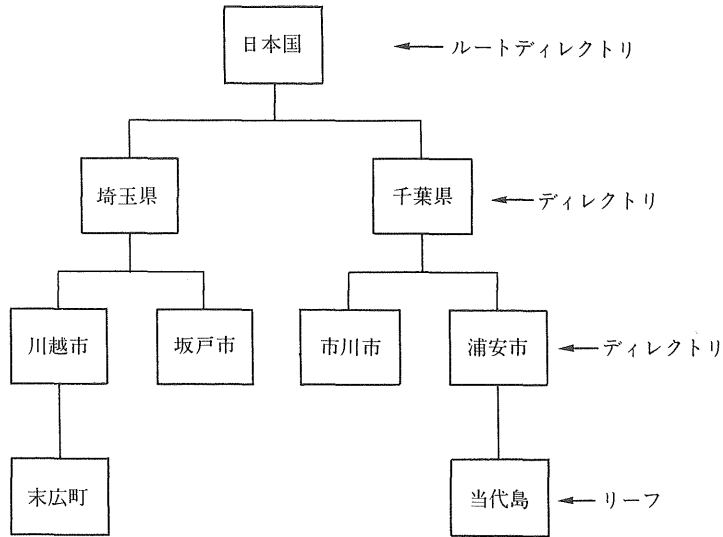


図2 階層構造

(4) シェル (shell)

これは、コマンドインタプリタといわれる命令処理のプログラムで、利用者が入力した命令をシェルが判断して、システムに伝達する働きを持っている。具体的には、まずディスプレイ上に「OS 9:」を出力し、次に標準入力から1行入力し最後に入力した内容にしたがって、プログラムを起動するようにシステムに命令することになる。このため、ディスクファイルから命令を受け取るように、シェルのリダイレクトすることも可能である。

(5) プログラムのモジュール化

さらに、OS-9の特徴としては、プログラムのモジュール化があげられる。6809のプログラムは、原則としてメモリ上のどの位置に置いてもよいという特徴を持っている。OS-9ではこの特徴を拡張し、モジュール化という形で、プログラムを扱い、自動的にプログラムをメモリのどこに置くか管理している。このため、メモリを効果的に使用することが可能になるとともに、プログラムの機能の中で共同利用できるものがある場合は、これを共同利用することにより、プログラムの開発手数を減少することができる。

(6) マルチウインドウ

先に述べたように、マルチウインドウとは、一つのディスプレイ画面を分割して利用することを指す。OS-9は、一度に複数のプログラムを管理するために、このような機能を持っている。このディスプレイ画面を示すと図3のようになる。

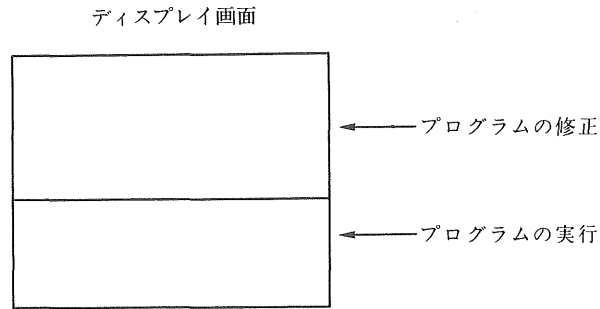


図3 マルチウインドウ

(7) OS-9の構造

OS-9 自体を構成するプログラムは、カーネルという一種の核を中心とした階層構造になっており、このカーネルは、最初のリセット後のシステムの初期化を行い、メモリの管理、マルチタスクの管理などの基本的な部分の処理を行う。これを図4に示す。

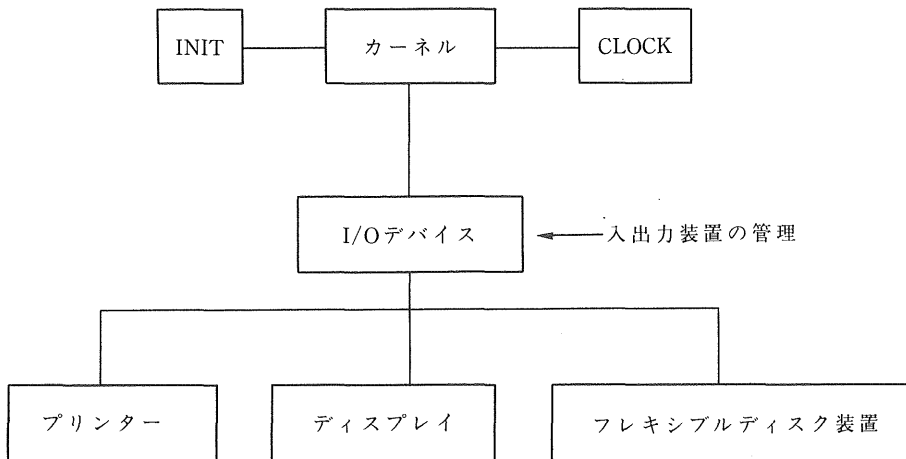


図4 OS-9の構造

INIT は、データテーブルで、カーネルが初期化を行う時に必要な、最初に起動する装置名などのシステムの立ち上げに関する情報を提供している。CLOCK は、リアルタイムクロックのハンドラとして使用しており、I/Oデバイスは、カーネルからの支持により、入出力装置の管理を行う。

(8) 日本語処理

OS-9の日本語バージョンでは、OSレベルで日本語処理機能を装備し、カナ漢字変換、ローマ字変換をサポートしている。このため、日本語処理機能は、OS-9のもとで働くBASIC09、アセンブラ、その他の全てのアプリケーションソフトで使用可能である。

日本語処理はシフトJISコードにより行われているが、JISの漢字コードに対しては、変換作業が要求されている。また、シフトJISコードを利用しているため、ANK文字と混在しての利

用が可能であり、メモリ容量の節約ができる。ただし、この場合、シフト JIS コードが利用しているコード帯にある、疑似グラフィックス文字が使用不可能となる欠点が生じることに注意が必要である。

(9) その他

OS-9 について使用するコマンドについては、大文字と小文字の区別がない。

3. BASIC09

(1) BASIC09 の特徴

BASIC09 は、OS-9 に標準装備されているプログラミング言語である。基本的には、BASIC 言語とほとんど同じであるが、次のような特徴を備えている。

① 実行速度が、ほかの BASIC に比較して早い。

これは、「I コード」と呼ばれるインタプリタを備えているためであり、BASIC のプログラムを、「I コード」という中間言語に翻訳するために生じる特徴である。したがって、同じ内容であっても他の BASIC のプログラムよりサイズが小さくメモリが少なくて済み、実行速度が早いという特徴を持っている。また、プログラムを翻訳するときに 1 行ごとに翻訳されるため、文法上の誤りがあればその時点で指摘されることになる。したがって、すべてのプログラムが翻訳が完了したあとで、プログラムの実行中に文法の誤りでエラーが生じることはない。

② プログラムの構造化がしやすい。

③ システム、エディット、実行、デバックの 4 つに区分された、プログラムの作成から実行までのモード (状態) がある。

(2) BASIC09 の起動

BASIC09 を起動させるために、

OS9: Basic09

と入力すると、画面には、

Copyright 1980 by Motorola and Microware.

Reproduced Under License

Basic09

Ready

B :

と BASIC のプロンプト文 (B :) が表示される。

(3) システムモード

システムモードとは、エディットモード、実行モード、デバッグモードを管理運用するモードであり、一つのモードから別のモードに、変更する場合にはすべてこのモードを経由して変更することになる。

(4) エディットモード

エディットモードとは、プログラムの作成、修正を行うモードである。このモードを利用するためには、システムモードのとき、次のように、エディットコマンドとプロシージャ名を入力することになる。ここで、プロシージャ (procedure) は、プログラムと同様のものである。

```
B : eddit [プロシージャ名]
E :
```

上記のエディットコマンドとプロシージャ名の入力後、システムモードからエディットモードになるため、プロンプト文が“E :”に変更される。

(5) 実行モード

実行モードとは、作成したプログラムを実行するモードであり、このモードにするためには、システムモード以外のときは、一度システムモードに戻らなければならない。たとえば、エディットモードの状態ですシステムモードに戻るためには、

```
E : q
Ready
B :
```

というように q コマンドを使用してエディットモードを終了させる必要がある。q とは、quit (終わり) の頭文字である。

システムモードに戻った後に、ワークスペース内にあるプログラムを実行する。

```
B : run [プロシージャ名]
```

ワークスペース内に、一つしかプログラムがない場合と現在作業対象としているプログラムを実行対象としている場合は、プロシージャ名は省略可能である。

(6) デバッグモード

デバックモードとは、プログラムのバグを取り除くためのモードである。したがって、RUN コマンドを実行中に、なんらかのバグが発見された場合、その時点で、自動的に「デバッガ」が起動され、デバックモードに移ることになる。

```
BREAK: [プロシージャ名]
D :
```

ここで、デバックモードからシステムモードに戻すためには、以下のように、q コマンドを入力しなければならない。

```
D : q
Ready
B :
```

(7) プログラムの呼び出しと保存

すでに、作成してディスクファイル上に保存してあるプログラムを呼び出すためには、システムモード上で、LOAD コマンドを用いる。

```
B : load [プロシージャ名]
      [プロシージャ名]
Ready
B :
```

作成したプログラムを保存するためには、システムモード上で、SAVE コマンドを用いる。

```
B: save [プロシージャ名]
```

このとき、すでに同じファイル名のプログラムがディスクファイルに存在しているときは、作成してある別のプログラムを間違えて消去しないように、プログラム作成者に確認するため、

```
Rewrite ? :
```

と聞いて来るので、書き換えても良い場合には、y と入力する。

(8) プログラムの作成

プログラムの作成は、エディットモード上で行う。つまり、

```
B : edit [プロシージャ名]
PROCEDURE [プロシージャ名]
*
E : A=100
*
E : H=50
*
E : S=A * H/2
*
E : PRINT "MENSEKI =", S
*
E : END
```

のように入力する。他の BASIC のプログラムと違い行番号は付かない。この理由は、リストコマンド I * または In で、入力されているプログラムをディスプレイ上に出力してみるとわかる。

```
B : I *
PROCEDURE [プロシージャ名]
* 0000 A=100
0015 H=50
002E S=A * H/2
0040 PRINT "MENSEKI =", S
0055 END
*
E :
```

各行の左側に付いている 16 進数の数字が行番号の代わりにしているからである。

(9) BASIC09 から OS-9 に戻す

OS-9 のひとつのコマンドとして実行されている BASIC09 を終了して OS-9 のコマンド待ちの状態に戻るには、次のように bye コマンドを使用する。

B : bye
OS9 :

4. OS-9 の評価と今後の動向

プログラムをマシン語で記述する場合、どうしてもターゲットマシンの CPU に依存することになる。それは、OS もまた例外ではない。

8ビットパーソナルコンピュータの主流のCPUは、ザイログ社のZ80であった。CP/M-80は、Z80の上で動く最初のOSとして発表されたため、8ビットのパーソナルコンピュータの標準OSとしての地位を得ることができた。しかし、Z80が8ビットCPUとして、最も性能が良いわけではなく、モトローラ社の6809の方が性能では上であり、6809用のOSであるOS-9/6809は、CP/Mに比べて明らかに性能は良く、最も優れた8ビットであると考えられる。

OS-9はOSの中でもっともUNIXの影響を受けている、いわゆるUNIXライクと呼ばれるOSであるといえる。UNIXのもっている特徴の中で、OS-9が取り入れている主なものとしては、

- (1) 階層構造のディレクトリ
- (2) shellと呼ばれるコマンドインタプリタ
- (3) パイプライン処理

などがある。しかし、UNIXの備えていない特徴もいくつか持っており、その違いは、以下のとおりである。

- (1) メモリモジュールという概念
- (2) リアルタイム処理が可能
- (3) マシン語で書かれているため、UNIXに比較してかなりコンパクト。

ここでメモリモジュールという概念は、OS-9が管理する最小のオブジェクト単位を意味し、それがプログラムやサブルーチンやデータを構成している。また、メモリモジュールは各個が独立していて、ポジションインパクトであり、かつエントラントとなっている。したがって、OS-9はOS自体もまたメモリモジュールから構成されている。

これらの特徴から、OS-9はリアルタイム処理を必ず必要とする計測機器・FA等において高く評価されている。しかし、汎用ソフトとしてのOS-9を考えると、ビジネスアプリケーションソフトが少ないことに加えて、日本の16ビットパーソナルコンピュータの多くは、CPUとしてインテル8086系を使用しており（例：NEC PC-9800シリーズ）、モトローラ68000上でしか動かない（8ビットでは6809のみ）ため、OS-9は他のOSと比較して不利と考えられる。しかし、NEC

PC-9800シリーズと富士通の FM16β 上で動く 68000 ボードによりその欠点を補うことができる。
 以上述べたことから、OS-9 の評価と概要をまとめると以下ようになる。

長所	<p>リアルタイム処理機能をもっている。</p> <p>UNIX に比較してコンパクトである。</p> <p>CP/M や MS-DOS に比較してメモリ管理が強力である。</p> <p>UNIX の特徴を取り入れている。</p>
----	---

短所	<p>アプリケーションソフトが少ない。</p> <p>68000 や 6809 を CPU に持つパソコンが少ない。</p>
----	--

項目	内容
開発元	(米) Microware Systems
動作環境	<p>CPU 68000, 68010</p> <p>RAM 128 Kbyte 以上</p> <p>その他 フレキシブルディスクまたはハードディスク, リアルタイムクロック コンソール I/O, プリンタ I/O</p>
摘要機種	専用ボードにより, NE 可能 PC-9800 シリーズ FM16β シリーズに対応可能
タスク	マルチタスク
ユーザー	マルチユーザー
メモリ管理	16Mbyte

ファイル管理	階層ディレクトリ
アプリケーションソフト	少ない
ソフトの互換性	アプリケーションモジュールレベルで互換性をもつ
言語	FORTRAN, COBOL, PASCAL, C, BASIC, ASSEMBLER など
日本語処理	可能
応用分野	FA, 測定機器等

OS-9/68000 Levell

以上述べてきたように、OS-9は、非常に優れたオペレーティングシステムであり、主に産業用マイコンオペレーティングシステムとして、産業機械、制御システム、その他の工業製品に幅広く使用されている。これは、OS-9が、特有のリアルタイム・マルチタスクを採用しているからであり、この特徴を上回る6809系のオペレーティングシステムが登場してこない限り、改良を重ねることにより、今後もさらに優れたオペレーティングシステムへと変貌していくと思われる。