

日本語 FlexOS

FlexOS の概要説明と日本語化について
村上好彦

(株) デジタル・リサーチ・ジャパン 技術部

米国デジタル・リサーチ社で開発された、主にインテル80286/386M
PU用の汎用OSであるFlexOSについて、機能概要と日本語化方法につ
いて説明する。更にデジタル・リサーチ・ジャパンによって過去CP/Mの時代
より米国DRIオリジナルOSに対応して拡張/付加してきた日本語処理システム
(FSX: Foreign Language System extension) について総合的に説明する。

"Japanize of FlexOS 286/386" (in Japanese)

Give the outline of FlexOS and Japanize

Yoshihiko Murakami

Digital Research (Japan) Inc. Engineering Department

NCR Shibuya Building 14 Floor 16-16, Nanpeidai-cho, Shibuya-ku Tokyo 150, Japan

FlexOS is the general purpose O/S based on the Intel 80286/386 microprocessor.

FSX (Foreign Language System extension) is expanded and appended the original O/S (created by DRI) since we created CP/M.

Here it is described the function outline, how we created Japanese version of FlexOS, and then the generality of FSX.

1. はじめに

弊社 (DRJ: デジタル・リサーチ・ジャパン) は、創立当初より米国 DRI からリリースされる各種 OS (オペレーティング・システム) の国内向け販売と、その日本化作業に携わってきた。

これまでに DRI よりリリースされた OS としては、インテル MPU 系列では、CP/M-80 を初めとして CP/M+, CP/M-86, CDOS-86 (旧称: CCP/M-86), CDOS-386, 及び本論文で紹介する FlexOS シリーズがある。またモトローラ MPU 系列では、CP/M-68K, CDOS-68K (外部仕様は 68K 用 FlexOS と考えてよい), GEMDOS が挙げられる。これらプロダクトは過去 10 年以内にリリースされており、ここ数年の MPU の飛躍的な進歩と同期して商業化されたものである。

また“OS の日本語化”の意味も、一昔前の単なるメッセージの日本語表示出力と日本語のコード入力といった低レベルのものから、最近では高度の日本語文法解析ロジックをベースにした日本語入力や、ユーザ・プログラムから容易に利用できる各種の日本語独自の処理エントリィの提供へと、広範囲に解釈されるようになってきた。弊社ではこの日本語処理部分を FSX (Foreign language System extension) と総称している。FSX は前述した各種 OS に対して同一のコンセプトの基に日本語処理を提供している。

本論文では、FlexOS をベースにその日本語処理について説明する。

2. FlexOS の概要

FlexOS とはフレキシブル (柔軟性のある) なオペレーティング・システムとイメージしていただきたい。この OS は、95 パーセント以上が“C”言語によって記述されている事と、OS 機能の合理的なモジュール分散構造が、ターゲット MPU に対してもフレキシブルな環境を提供している。又、OEM によって付加される各種ペリフェラル装置についても、その特性を活かしたリソース・マネジャの管理下で、インストーラブルなデバイス・ドライバとしてフレキシブルに移殖する事が可能である。

また、これまでに弊社よりリリースした OS を大別すると、CP/M 系と Flex 系に分けることが出来る。CP/M 系が過去の資産の継承性と汎用パソコン OS をターゲットにおくとすれば、FlexOS は多様化する FA/SA (ファクトリィ/ストアード・オートメイション) 分野のシステム構築に、迅速かつ最適な OS 環境を提供するであろう。

本論文の紙面数をもって FlexOS の全容を論ずる事は困難である為、以下に FlexOS の特徴についてキーワードを列挙した。

FlexOS シリーズ

プロダクト名	対象 MPU
FlexOS 186	INTEL-80186, NEC-V20/V25
FlexOS 286	INTEL-80286
FlexOS 386	INTEL-80386
FlexOS V60	NEC-V60/V70
CDOS-68K	M68000

FlexOS の構成

- OS 機能を合理的に分散したモジュール構成 (SUP, RM, DRV, AE)
- リソース・マネジャによるペリフェラル装置の統括的な管理
- インストーラブルなデバイス・ドライバ

プロセス及びメモリ管理

- リアル・タイム
- マルチ・タスキング
- シングル/マルチ・ユーザ

- ・ メモリ・プロテクション

ファイル・システム

- ・ シェアド・ファイル・システム
- ・ DOS 3. Xコンパチブル
- ・ 階層ディレクトリ／ファイル
- ・ ファイル・オーナーシップ

コンソール・システム

- ・ 仮想コンソール (階層構造、ウィンド管理)
- ・ 8 / 16 ビット入力モード

スーパーバイザ・コール

非同期 I / O

- ・ 1 プロセス最大 31 のイベント
- ・ イベント終了による SWI (ソフトウェア・インタラプト)
- ・ 複数のイベント待ち機能

ドライバ・サービス・ルーチン

グラフィックス

- ・ スタンダード VDI インターフェイス

ネット・ワーキング

- ・ IBMPC ネットとコンパチブル (FlexNet)

ポータビリティ

- ・ CPU 及びペリフェラルに依存しないモジュール間インターフェイス

フロント・エンド (AE: アプリケーション・エンバイアメント)

- ・ PC DOS 2. x / 3. x

開発環境

- ・ インテル / COFF ・オブジェクト・コンパティブル (* . O)
- ・ シンボリック・デバック (CSID)
- ・ アセンブラ / リンカ / C コンパイラ (CASM, CLINK, CLIB, HC)
- ・ UNIX ライク・ユーティリティ (DUP, GREP / . .)

国際性

- ・ カントリー・コード (178 ヵ国)
- ・ メッセージ・最適化 / 移植容易 (CCM)

3. FSX の概要

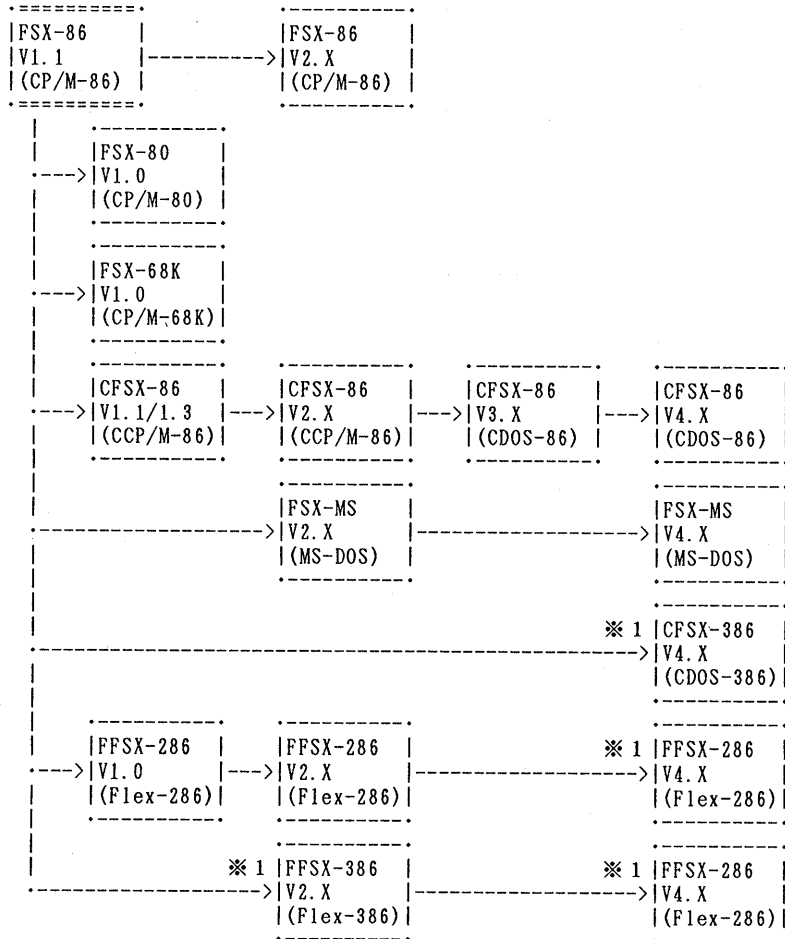
FSX は下記のコンセプトに基づいてデザインされている。

- 1) 従来、OEM の移植層に位置していた日本語入力部分をシステム・イクステンションとして統括し標準化する。
- 2) ユーザ・アプリケーションから簡単に高度の日本語処理が利用できるように、OS が標準にサポートしている SVC (スーパーバイザ・サービス・コール) の拡張として FSX のプログラム I / F を定義する。

3) システム及びトランジェント・ユーティリティの日本語化（主にメッセージの日本語化）を行なう。

4) 標準の日本語辞書を提供する。またこの辞書に対する、追加／削除／修正／再編成などの処理をオペレータ・プログラマ・レベルまで開放する。

上記の事柄を、極力標準のOSを変更する事なく、かつOEM機器の依存部分を少なく実現する事で、FSXの短期移植と新規OSへの従属性を図っている。実際にその成果として、FSXは（図3-1）のようにファミリーを形成している。



※1：作成中（昭和63年1月現在）

※注：V3.Xまでは単文節、V4.X以降は連文節変換をサポート

（図3-1：FSXファミリー）

またFSXシステムは(図3-2)のように機能別にモジュール化され分担当が明確になっている。

```

=====
| FSX・システム|
=====
|
|-->| 0.Sメッセージ |
|     | 日本語化       | | | | |
|---|---|---|---|---|---|
|-->| ローダー &      | | FSX | | FSX |
|     | インストーラ   | | .286 | | .INS |
|-----|
|-->| ローカル・      | | FSXMIF | | FSXLC |
|     | コンバージョン | | .L86 | | .286 |
|-----|
|-->| プログラム・    | | FSXPIF | | FSXCVT |
|     | インタフェース | | .286 | | .286 |
|-----|
|-->| 標準辞書        | | FSX |
|     |                | | .DIC | | |
|---|---|---|---|---|---|
|-->| 辞書メンテナンス | | FSXDUT | | FSXDMK |
|     | ユーティリティズ | | .286 | | .286 | | |
|---|---|---|---|---|---|---|---|
|-->| テスト・        | | FSXDDT | | FSXDET | | FSXIFT |
|     | ユーティリティズ | | .286 | | .286 | | .286 |
|-----|
|-->| 拡張辞書メンテ.  | | FSXDMT | | FSXMRG |
|     | ユーティリティ  | | .286 | | .286 |
|-----|

```

(図3-2: FSXの構成)

さらに(図3-2)の各モジュールの概要は以下のとおりである。

- FSX. 286 ; ローダ・モジュール

FSX. 286は、標準OS上に任意のタイミングでFSXをローディングする。また、この時点でFSX. INSファイルを参照する事により、柔軟にFSXの動作条件を選択する事ができる。(動作条件: 辞書名、コンソール画面制御コード、学習など...)
- FSXMIF. L86 ; マン・マシンI/Fモジュール

FlexOS下で動作する任意のアプリケーションから発せられるコンソール入出力時に、ローカル・コンバージョン機能を実現する。FSXの基本モジュールの中で唯一、FlexOS生成時に組み込まれる部分である。
- FSXPIF. 286 ; プログラムI/Fモジュール

FlexOS下で動作する任意のアプリケーションからFSXのプログラムI/F機能を実現する。
- FSXCVT. 286 ; かな漢字変換・モジュール

日本語のかな漢字変換を実現する。(文節単位に接頭/接尾語、連体/数詞、記号などの文法解析処理を行なう) O.E.Mによるカスタマイズが可能である。

- F S X L C . 2 8 6 ; ローカル・コンバージョン・モジュール
ローカル・コンバージョン操作を実現する。OEMによるカスタマイズが可能である。
- F S X . I N S ; インストール・セット・アップ・ファイル
F S X の動作条件を設定する。エディット可能なアスキー・ファイル。
(動作条件：辞書名、コンソール画面制御コード、学習など...)
- F S X . D I C ; 標準辞書
- F S X D U T . 2 8 6 ; 標準辞書メンテナンス・ユーティリティ・プログラム
F S X の標準辞書に対して、語句の登録／削除／表示等の操作を行なう。
- F S X D M K . 2 8 6 ; 標準辞書再編成・プログラム
F S X の標準辞書に対して、その空きエリアの報告と再編成を行なう。
- F S X D D T . 2 8 6 ; 簡易デバックング・ツール
画面上で簡単にF S X コールを発する事が出来る。
- F S X D E T . 2 8 6 ; 辞書評価・ツール
読み／変換漢字列の評価用テスト・データを自動入力して、F S X のかな漢字変換効率を評価できる。同様に自動登録／学習が行なえる。
- F S X I F T . 2 8 6 ; F S X 自動診断テスト・ツール
F S X の各種機能を自動的にテストする。
- F S X D M T . 2 8 6 ; 辞書メンテナンス・拡張・ツール
機能的にはF S X D U T / F S X D M K / F S X D E T の機能を有し、更にユーザ・登録語の追加／削除等の機能が拡張されている。
- F S X M R G . 2 8 6 ; 辞書生成/結合ユーティリティ
標準辞書に対しての大量の語句の追加／削除の操作が行なえる。また種辞書も提供される為、新規の辞書を独自に作り上げる事が出来る。

4. F l e x O S の構造とF S X

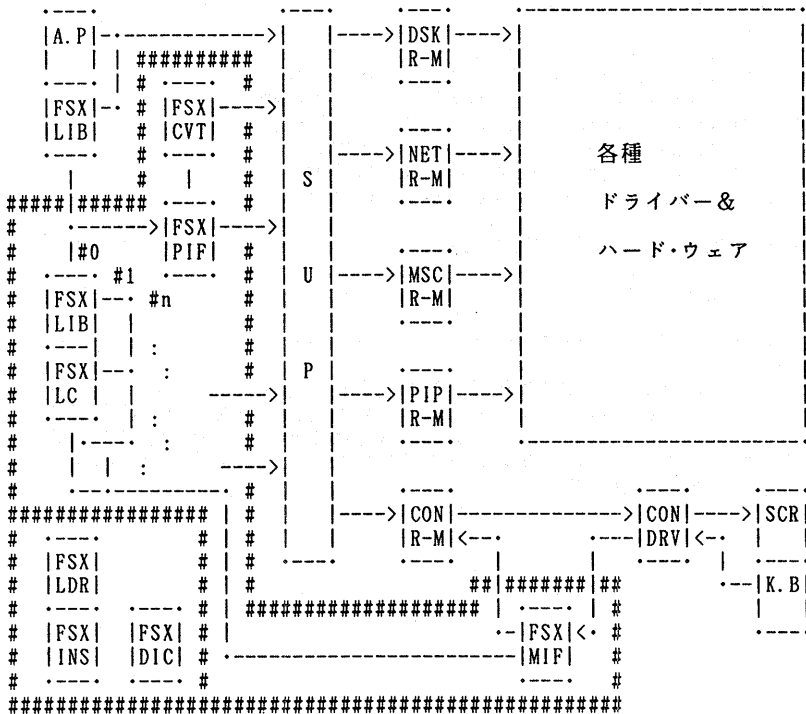
ここではまずF l e x O S の構造を説明し、その後にF S X システムとのつながりについて述べる。

F l e x O S は大きく分けて、三つの部分から構成される。(図4-1) 1 : F l e x O S の中核として全体を制御するスーパーバイザ (S U P)、2 : システムが有する資源を統括して管理するリソース・マネジャ (R - M)、3 : 物理デバイスとのインタフェースをとるデバイス・ドライバ (D / D)、である。1 : のS U P は、アプリケーションとOSのインタフェースを提供する。アプリケーションがOSの資源にアクセスするときは、S U P を呼び出す (S V C : スーパーバイザ・コールを使用する)。するとS U P は要求に応じて、該当するR - M を (必要ならば別のS V C を発行して) 呼び出しその要求を達成する。S U P はまた、プロセス (タスク) 及びメモリを管理する。プロセス管理とメモリ管理は、S U P 中のカーネル部が担当する。S U P は、イベント発生割り込みなどを契機にプロセスの切り替えを行ない、各プロセスの状況を管理する。2 : のR - M は、各種装置用のデバイス・ドライバを一元管理する。似たような装置、例えばハード・ディスクやフロッピー・ディスク・ドライブなどを管理するD / D には、共通する部分が存在する。この部分をR - M の中にまとめあげることにより、ハードウェア依存部分 (即ち、メーカーが自分のシステムにあわせて新たに開発しなければならぬ移植部分) を出来るだけ小さくすることが可能だ。またR - M は、

対応する各D/Dとの間の標準的なインタフェースを定義している。このことによりR-MとD/D間のやりとりを明確化している。3:のD/Dは、各物理装置とのインタフェースを司るソフトウェアである。前述のように、R-Mによって制御される。またこれは実行時に動的にインストールされるため、プロセス必要に応じて新たなD/Dを必要とした時点でロードすることが出来、逆に不要になればディ・インストール(削除)することも可能である。

次にFSXとの関係について説明する。一般に、米OSにFSXを移殖しようと考えた場合、2つの事を検討する必要がある。一つは、FSXのプログラムI/Fの実現方法である。FlexOSでは標準のOSになるべく変更を加えないようにパイプを利用した。即ち、プログラムI/Fが持つ各種日本語処理機能をアプリケーションが必要とした時、その要求をパケット化してFSXプログラムI/F用の受信パイプ書き込むことによりその主旨を伝え、更に送信パイプよりその実行結果をもらう方法である。2つ目は、日本語入力の方法である。これはOSが用意しているキー入力パスの延長として実現しなければならない。つまりアプリケーションのキー入力要求は、その入力されるデータがANK(アルファベット・ニューモニック・カナ=1バイト・コード)か漢字か(=2バイト・コード)を意識して出されるわけではなく単にキー・ボードからのデータが欲しい訳である。そこでユーザがたまたま漢字入力が必要とした場合、現在使用中のANKキー・ボードが漢字も入力できるキー・ボードに変身してくれるといい訳である。これは、あるキーをその変身用スイッチとして割り当て、更にそのキーが押された時に発生するキー・コードを監視し、もし漢字要求キーでない場合はそのままアプリケーションに渡し、そうであった場合はFSXシステムを起動して漢字入力モードにはいる(画面上にガイダンス行を表示し漢字の読み入力要求状態になる)と言った一連の処理を必要とする。よって、FSXはOSが行なう通常のキー入力パスをうまく監視し、奪い取るが必要である。そしてこの作業の難易度は、対象となるOSがいかにキー入力パスを1本化しているかにより変わってくる。この点でFlexOSは、前述したようにR-M/D-D間を明確なI/Fによって構築されているため、(図4-1)に示すシンプルな移殖が可能になった。

|<=== Program ===|<=== System ===|<===== Physical =====|



(図4-1: FlexOS/FSX構成図)

5. FSXプログラムI/F

前述したようにFSXは高度の日本語処理をユーザに開放している。具体的にはFSXのプログラムI/Fを使用したアプリケーション・プログラムを作成する場合、リンク時にFSXLIBライブラリを結合する。FSXLIBは、FSXPIFが持っている入力パイプに対してその指示を与え、その結果としてFSXPIFの出力パイプに得られたデータをアプリケーションへ返還している。FSXのプログラムI/Fコールは、機能的に大きく6つのグループに分かれており、各々はまた種々のサブ・ファンクションから成っている。以下にその内訳を示す。

・プログラム・インタフェースの確定/解除を行なうグループ.....(ACC_FSX)

FSX_OPN : Program I/F Open : プログラム・インタフェースのオープン
FSX_CLS : Program I/F Close : プログラム・インタフェースのクローズ
FSX_VER : Get FSX Version : FSXのバージョン情報を得る

・各種モードを設定するグループ.....(ACC_MOD)

SEL_IMD : Select Input Mode : ローカル・コンパートのオン/オフ
SEL_LRN : Select Learn Mode : 学習機能オン/オフ
SEL_COD : Select Code Mode : コード体系の選択

・漢字変換処理を行なうグループ.....(ACC_CVT)

CVT_STR : Convert String : 与えられた文字列に対してカナ漢字変換
CVT_NXT : Convert Next : 次候補を要求する
CVT_RTK : Roman to Kana : ローマ字で与えられるローマ字カナ変換
CVT_HTZ : Hankaku to Zenkaku : 半角コードから全角コードへ変換する
CVT_ZTH : Zencaku to Hankaku : 全角コードから半角コードへ変換する
CVT_JTS : JIS to SHIFT JIS : JISからシフトJISコードへ変換する
CVT_STJ : SHIFT JIS to JIS : シフトJISからJISコードへ変換する

・辞書アクセスを行なうグループ.....(ACC_DIC)

DIC_OPN : Dictionary Open : 辞書ファイルをオープンする
DIC_APP : Dictionary Append : 辞書ファイルへの追加
DIC_DEL : Dictionary Delete : 辞書ファイルからの削除
DIC_LRN : Dictionary Learn : 辞書ファイルへの学習指示
DIC_SRH : Dictionary Search : 辞書ファイルの検索
DIC_NXT : Dictionary Next : 検索時の次候補要求
DIC_CLS : Dictionary Close : 辞書ファイルのクローズ

・FSXの将来の拡張の為にリザーブされているグループ.....(RSV_FSX)

・O.E.Mによって使用されるグループ.....(ACC_OEM)

・FSXの内部で使用されるグループ.....(ACC_INT)

更に、これらのエントリは“C言語”などの高級言語より簡単に利用できるようライブラリが用意されている。また、これらは主にFSXが持つ仮名漢字変換モジュール(FSXCVT)をユーザからも使用できる用にしたものであるが、昨今ではマン・マシンI/Fモジュール(FSXLC)の機能も呼び出せるように拡張I/Fを作成中である。

6. むすび

昨今、マイクロ・コンピュータ業界はインテル社80386MPUの登場を口火として16ビットから32ビットへと着実に移行が進んでいる。そしてこれら32ビットMPUが持つ高速かつ高度化された処理能力を、一般ユーザに有意義な環境として還元するために、OSの役割は重要である。例えば、大型機並のリアル・タイム処理環境を実現するための高速ディスクパッチャ、タスク管理、仮想メモリ、また32ビットにオブティマイズされた開発環境と16ビットとのクロス環境、そして従来のアプリケーションの継承性など、OSのレベルでの迅速なるサポートが求められている。そのような中で、FlexOSは文頭でも説明したようにMPUに対してもフレキシブルであるため、既にFlexOS386もリリース済みである(CompaqDESKPRO386)。更にNEC社V60/70に対応したFlexOS-V60もリリースまじかである。

また上述のようなMPUの飛躍的な発展に伴って、従来では不可能であった(即時性とメモリ・サイズの制限)マンマシン・インタフェースの実現性が論じられるようになってきた。例えば、現在でも仮名漢字変換処理は、短文節から連文節があたりまえになっている。近い将来には、知識ベースを利用した学習処理変換に、しいては他言語への翻訳処理も可能となるであろう。また大局的に見ると日本語入力処理は仮名漢字変換アルゴリズムによるデータ・ベース検索を利用したデータ・エントリィ・システムの一形態と考えることも出来るため、そのシステムが持つデータ・ベース機能との有機的な結合が考慮されるであろう。FSXはこういった観点から見ると、OS依存部、マンマシンI/F部、仮名漢字変換部、とモジュール化されているため、例えば今後登場するであろうこれらの最新機能を(制作元が自他社にかかわらず)ソフトウェア・パーツとして短期に組み込むことを由としている。

以上、今後のFlexOSとFSXの発展に期待されたい。