

日本語入力システムJEFの使用方法

武富, 敬
九州大学大型計算機センター研究開発部

高木, 利久
九州大学大型計算機センター研究開発部

川崎, 正子
九州大学大型計算機センター業務掛

富山, 実
九州大学大型計算機センター業務掛

他

<https://doi.org/10.15017/1474891>

出版情報 : 九州大学大型計算機センター広報. 13 (4), pp. 406-468, 1980-12-15. 九州大学大型計算機センター
バージョン :
権利関係 :

日本語情報システム J E F の使用法

武富 敬^{*}, 高木 利久^{*}, 川崎 正子^{**}, 富山 実^{**}, 柳池 定^{**},
原田 盛一^{***}, 関 武治^{***}, 末永 邦夫^{****}, 清水 和明^{****}

目 次

1. はじめに	407
2. J E F 概説	407
2.1 概 要	407
2.2 J E F 漢字コード	408
2.3 J E F のハードウェア	409
2.4 J E F のソフトウェア	412
3. F D M S (和文エディタ) の使用法	416
3.1 概 要	416
3.2 作成例	418
3.3 カナ漢字変換	421
3.4 テキスト入力法	422
3.5 テキスト作成法	424
3.6 種々のテキスト編集法	426
3.7 テキストマクロの使用法	429
3.8 私的カナ漢字変換辞書の使用法	430
4. K I N G (文章処理) の使用法	430
4.1 概 要	430
4.2 使 用 法	431
4.3 備 考	432
5. P L / I , C O B O L における使用法	432
5.1 P L / I における日本語処理	432
5.2 C O B O L における日本語処理	438
6. K I N G (グラフ・図形出力) の P S P C N V の使用法	444
6.1 図形制御カード	445
6.2 ジョブ制御文	448
6.3 作 図 例	449
付表 1. J I S 非漢字と J I S 第 1 水準漢字 (J E F 漢字コード系による)	452
付表 2. J I S 第 2 水準漢字 (J E F 漢字コード系による)	455
付表 3. F D M S (和文エディタ) のコマンド入力形式一覧	458
付表 4. 制御コード入力形式一覧	464
付表 5. テキスト入力における変換指定一覧	466
付表 6. カタカナ・ローマ字対応表	467
付録 1. 図形制御カードの一般規約	468
付録 2. 図形制御カードのエラーコード	468

1 はじめに

計算機が誕生して以来、ほぼ30年が経過した。この間、その利用される分野も、当初の数値のみを取り扱う計算処理から、図形処理、テキスト処理、情報検索処理など広汎な情報処理を含むところまで拡大されてきた。このような情勢のもとで、従来の英数字・カナを用いた情報処理に対し、漢字かな混り文による日本語情報処理の要求も、さまざまな分野で強かったものと思われる。しかしながら、取り扱う文字種の多様さ、同音異義語の選択、漢字入力 of 困難性などの日本語の持つ特異性のために、その汎用処理システムの実現は、それほど容易ではなかったようである。ところが、最近のハードウェア・ソフトウェアを含む計算機技術の急速な進展は、この事を可能にしつつある。また、漢字を中心とするコード系の標準化の要求の高まりの中で、昭和53年1月にはJIS漢字コードも制定され、日本語情報処理に対する共通の基盤がしかれることとなった。

今回、本センターでも、富士通により開発された汎用な日本語情報システム(Japanese processing Extended Feature, JEFと略称される)を導入することにより、そのサポートの態勢が整った。これによって、大学においても大量に生産される文書、特に、和文の論文、学会予稿などの原稿、国内での研究会・シンポジウム等の報告文書、会議・連絡会等の連絡文書などの作成・管理に大いに役立つものと思われる。以下、このJEFシステムの概要について解説する。

2 では、JEFシステムについて概観する。まずJEFで用いられる漢字コード、文字の種類などについて述べ、次にJEFのハードウェア、最後にJEFのソフトウェアについて述べる。3 では、日本語テキストの作成・編集のために用いられるFDMS(和文エディタ)の使用法について述べる。4 では、FDMS(和文エディタ)で作成した日本語テキストを、フォーマットして日本語ラインプリンタ装置に出力するためのKING(文章処理)の使用法について述べる。5 では、言語処理プログラムPL/I, COBOLにおける日本語処理について述べる。6 では、日本語ラインプリンタ装置を用いる1つのアプリケーションとして、PSPで作成した図形を出力するKING(グラフ・図形出力)のPSPCNVの使用法について述べる。なお、本稿では、上記のいくつかのソフトウェアの使用法を解説することにし、それ以外のもの、たとえば、会話型汎用日本語エディタ右筆(ユーヒツと読む)、KING(グラフ・図形出力)のPSPCNV以外の部分、KING(帳票出力)等については、次の機会に譲ることとする。

2 JEF概説

2.1 概 要

JEFは、従来の英数字・カナ処理システムを基盤として、その拡張形として漢字かな混り文の処理を可能にする汎用日本語情報システムである。図2.1に、JEFのハードウェア、ソフトウェアコンポーネントの関連図を示す。ハードウェアに関しては、本センターでは、日本語テキストの入力・編集のために日本語ディスプレイ装置を使用し、日本語文章あるいは図形の出力のために日本語ライ

-
- * 九州大学大型計算機センター 研究開発部
 - ** " 業 務 掛
 - *** " 共同利用掛
 - **** 富士通 SE

ンプリンタ装置を使用する。ソフトウェアに関しては、日本語処理のために新規にサポートされるものとして、サービスプログラムのADJUST, アプリケーションプログラムのKING, FDMS, KUINなどがある。また、日本語処理に特有な、多種多様の文字パターンを納めた辞書, 文字属性辞書, およびカナ漢字変換のための辞書などが、テーブルウェアと総称して提供されている。

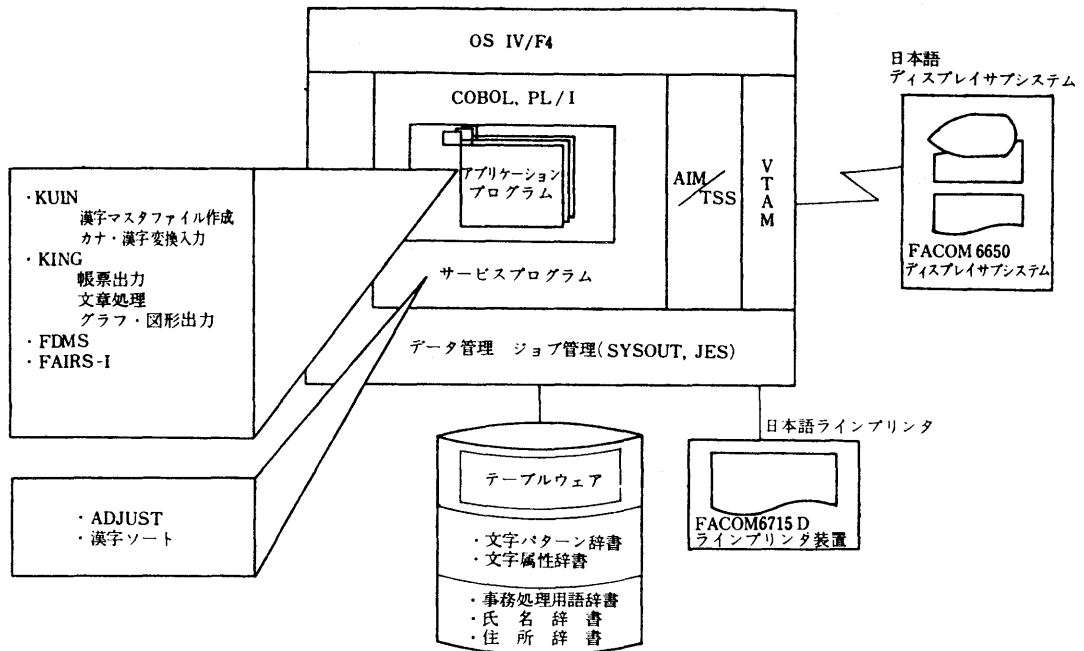
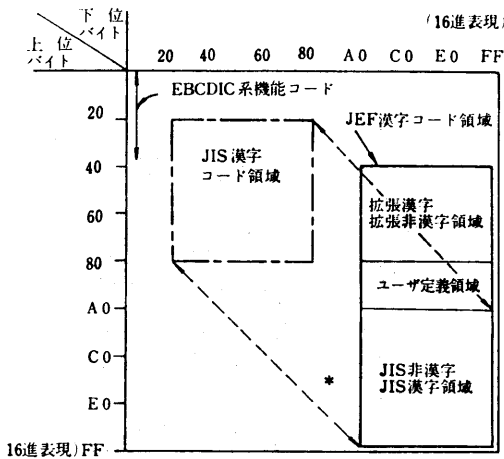


図 2.1 J E F のハードウェア, ソフトウェアコンポーネントの関連図

2.2 J E F 漢字コード

従来の英数字・カナを表現するEBCDICは、1バイトコード系で、それでは高々256文字種しか表せない。漢字は、当用漢字に限っても1850字あり、文字種が豊富なため、2バイトコード系で表現される。JEFでは、図2.2に示すように、JIS漢字コードに準拠し、それよりも大きい漢字集合を表わす「JEF漢字コード系」を採用している。表2.1に示されるように、JIS文字セットでは、JIS非漢字、JIS第1水準漢字、JIS第2水準漢字合わせて6802文字種あるのに対して、JEF文字セットでは、さらに独自の拡張非漢字、拡張漢字、ユーザ定義文字を含め、13996文字種が取り扱える。付表1に、JEF漢字コード系でのJIS非漢字とJIS第1水準漢字、付表2には、JIS第2水準漢字のコード表を掲げた。実用上は、JIS第1水準漢字の3000字程度で十分と思われる。



* JIS漢字コード/JEF漢字コード変換
 JISコード(1文字)+16進(X'8080')=JEF漢字コード
 例:「君」JISコード(X'372F')+(X'8080')=JEFコード(X'B7AF')

上図のようにシフトしている理由は、JIS漢字コード領域の一部がEBCDIC系機能コードと重複して区別がつかないためである。

図 2.2 J E F 漢字コード系

2.3 J E F のハードウェア

2.3.1 日本語ラインプリンタ装置

日本語ラインプリンタ装置 (NLP と略称される) は、ノン・インパクト型のレーザ・プリンタである。印字方式は、レーザビーム書き込みにより感光ドラムに潜像を作成しておき、トナーで現像し、普通紙に転写・定着させる乾式電子写真方式を採用している。また、NLP は日本語処理に対応できるように、フロッピディスクに多様な文字パターンを收容しておくための文字收容機構、実際に印字できる形に、文字をドットパターンで記憶しておくためのメモリを持つ文字発生機構、固定的なフォーマット部分を、可変な文字列と重ね合わせて印刷できるようにするフォームオーバレイ機構を備えている。

図 2.3 に、文字收容機構と文字発生機構の関連図を示す。NLP 電源投入時に、ハードウェアの機能により、JIS 非漢字、JIS 第1水準漢字合わせて 3418 文字、および書式文字 (罫線出力用としてコード化されている文字) の 11 文字の文字パターンは、自動的に文字收容機構から文字発生機構にローディングされ印字できる態勢になる (イニシャルローディング)。現在、増設部

表 2.1 J E F 漢字コードの文字種

区 分	内 容 概 要	文 字 種 数	
非漢字	JIS 非漢字 (英数字など)	453	808
	拡張非漢字 (変体仮名、書式文字、特殊文字、グラフ・図形記号)	355	
JIS 漢字	JIS 第 1 水準 漢字	2,965	6,349
	JIS 第 2 水準 漢字	3,384	
拡張漢字	富士通選定漢字 (JISに含まれず、従来漢字) (処理で使用されている文字)	3,737	3,737
ユーザ定義文字	ユーザ固有の文字、パターンなど	3,102	3,102
合 計			13,996

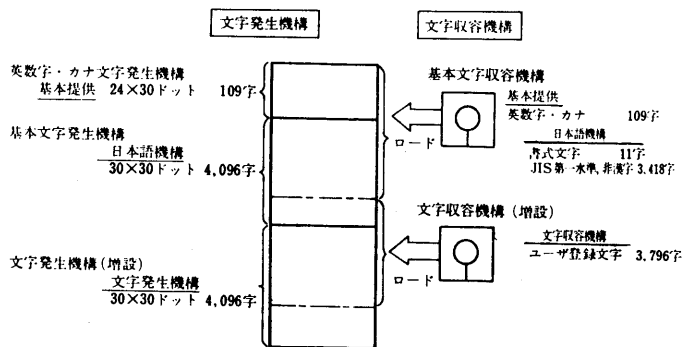
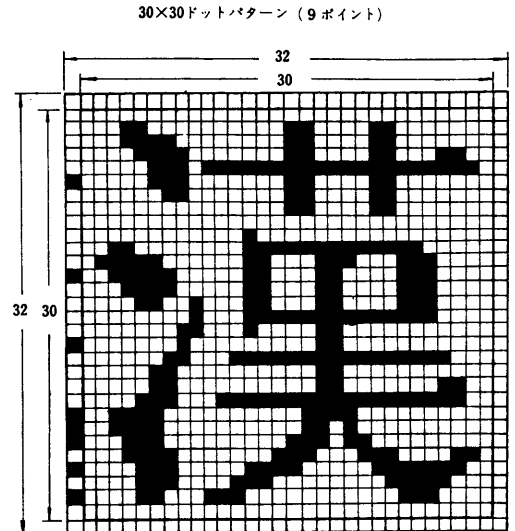


図 2.3 文字收容機構と文字発生機構の関連図

分の文字収容機構のフロッピーは空で、イニシャルローディングに関係しないため、文字発生機構において残された約4800字分のメモリは、本センターで頻繁に使用されると思われる図形出力のための図形パターンを格納する場としてとってある。

NLPで出力される漢字は、明朝体の書体を用い、**図2.4**に示すように30×30のドットパターンで表現されており、高品質の印字文字が得られる。文字サイズは、JEF漢字コードで表される漢字・ひらがななどは、文字間隔が8字/インチの大きさ(9ポイント相当の文字、以下9ポの文字と呼ぶ)と、5字/インチの大きさ(12ポイント相当の文字、以下12ポの文字と呼ぶ)の2種類が使用できる。ただし、文字サイズごとに文字パターンを持つことからメモリ増の欠点をなくすため、40×40ドット相当の12ポの文字は、30×30ドットの文字パターンから、文字拡大アルゴリズムにより作成するという方法が採られている。これにより、文字発生機構には30×30ドットのパターンのみが格納されている。EBCDICの英数字・カナの文字間隔は、従来と同じく10字/インチの大きさで、24×30ドットで表現されている。

行間隔は、6行/インチ(6LPIと呼ぶ、標準値は6LPIである。なお従来のラインプリンタ出力も6LPI)と、8行/インチ(8LPIと呼ぶ)の両方が可能である。英数字・カナおよび9ポの文字は、6LPIおよび8LPIの行送りで印字可能であるが、12ポの文字は、6LPIでのみ印字可能である。**図2.5**に、文字サイズと印字位置について示す。**表2.2**には、文字枠の大きさと行内印字文字数について示す。

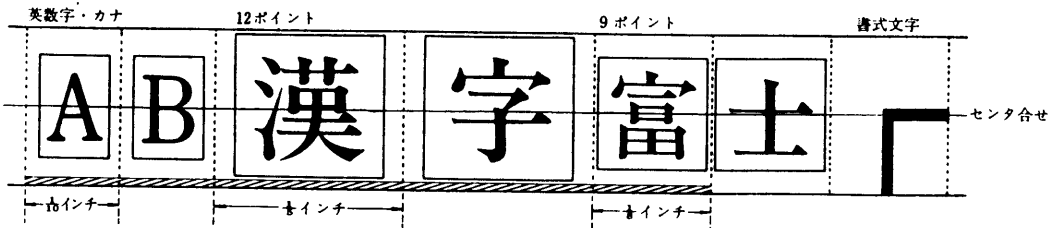


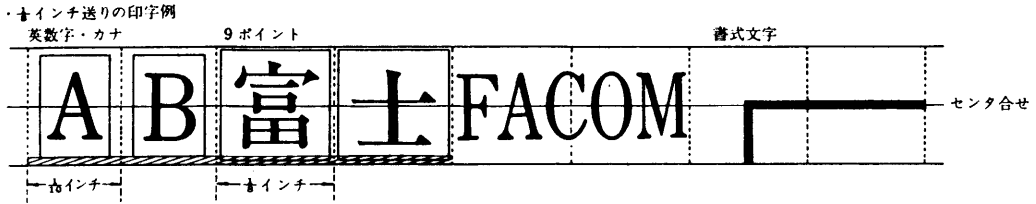
拡張ビット
(30ドットから40ドットの拡大では、拡張ビットは全部で8ヶ所拡張を指示する)

- 注1 9ポイントの文字は、印字されるのは30×30ドットのパターンであるが、文字発生機構では拡張ビットを含む32×32ビットの構成で記憶されている。
- 注2 この文字のパターンは、実際に提供される文字のパターンとは異なる。

図2.4 JEFにおける漢字のドットパターンの例

・1/4インチ送りの印字例





(注1) 1/10インチ送りでは、12ポイントの文字は印字できない。

- 重ね打ちによるアンダラインの印字領域
- レターフェイス(ドットパターンで表現する文字の大きさの領域)
- 文字枠の大きさ(印字幅)

図 2.5 NLPにおける文字サイズと印字位置

表 2.2 文字枠の大きさと行内印字文字数

出力文字		行送り		行内印字文字数		
		幅	高さ	幅	高さ	
英数字・カナ		1/10インチ	1/8インチ	1/10インチ	1/8インチ	136文字
12ポイント	書式文字	×		×		—
	書式文字以外	1/5インチ	1/6インチ	×		68文字
9ポイント	書式文字	1/8インチ	1/6インチ	1/8インチ	1/8インチ	109文字
	書式文字以外	1/8インチ	1/8インチ	1/8インチ	1/8インチ	109文字

(注) ×印の組み合わせを指定すると、印字結果が保障されない。

用紙の紙送り制御は、FCB (Forms Control Buffer) により、ページ単位に制御できるので、同一ページ内に6LPIと8LPIの混在が可能である。しかしながら、利用者がどのようなFCBイメージを欲しているか予測はできないので、現在は標準FCBイメージを採用しており、混在はできない。また、図形出力データセット中の図形パターンを、一種の文字パターンとみて文字発生機構に登録し、文字・図形の出力を可能にするユーザライターであるKINGライターは、標準値以外のページ行数などを指定すると、FCBイメージを書き替えてしまうので、それ以後、セパレータ出力がずれるなどの障害を起こしてしまう。まもなく修正される予定なので、それまでは注意していただきたい。さらに、NLPでは、従来のプリンタと違って、用紙のミシン目前後各々2行は、感光ドラムから用紙への転写時、用紙がドラムに密着しないので、印字禁止領域となっていることに注意されたい。

最後に、印刷データの形式についてふれる。図 2.6 にその一例を示す。データ中には、英数字・カナのEBCDIC(1バイトコード)と、漢字・

英数字・カナ (EBCDIC)	K シフト	日本語データ (JEF漢字コード)	A シフト	英数字・カナ (EBCDIC)
--------------------	----------	----------------------	----------	--------------------

— 行の先頭は、必ずAシフト状態である。

図 2.6 印刷データの形式

ひらがななどのJ E F漢字コード(2バイトコード)が混在可能なので、それらのコードを識別するために、その間に1バイトのシフトコードが挿入される。シフトコードには、以下の文字列がE B C D I Cであることを示すAシフトコードと、以下がJ E F漢字コードであることを示すKシフトコードとがある。Kシフトコードは、さらに、以下が9ポの文字列であることを示すK 1シフトコードと、1 2ポの文字列であることを示すK Sシフトコードとに分けられる。ただし、利用者は、ほとんどの場合、このシフトコードを意識する必要はない。

2.3.2 日本語ディスプレイ装置

本センターでは、2階に設置してある、J I S型のけん盤を持つ日本語ディスプレイ装置(NDPと略称される)を用いて、T S Sで日本語テキストの入力・編集を行う。N D P電源投入時、日本語ディスプレイ制御装置の文字収容機構から、24×24のドットパターンの文字が、N D Pの文字発生機構にローディングされ、文字の表示が可能となる。N D Pでは、J I S第2水準漢字も表示されるので、N D Pでは表示されるがN L Pには出力されないような場合も起りうる。現在、このための手段を検討中なので、センターニュース等に注意されたい。通常は、後述のF D M S(和文エディタ)により、カナ漢字変換方式でテキストの入力・編集を行うが、J E F漢字コードを16進4桁で入力する漢字直接入力も可能である。従来、日本語情報処理の最大の難点の一つに、漢字入力の問題があるとされてきたが、このN D Pとカナ漢字変換のF D M S(和文エディタ)を合わせて用いることにより、専門的なオペレータでない普通の利用者も、かなり容易に、画面編集形式でテキストの入力・編集を行うことができる。なお、N D Pの使用に便利のように、P F(プログラム・ファンクション)キー機能一覧表(F D M S(和文エディタ)により作成)をN D Pのそばに備えている。

2.4 J E Fのソフトウェア

オペレーティングシステムO S I V / F 4では、J E Fサポートのために、大幅に機能強化された

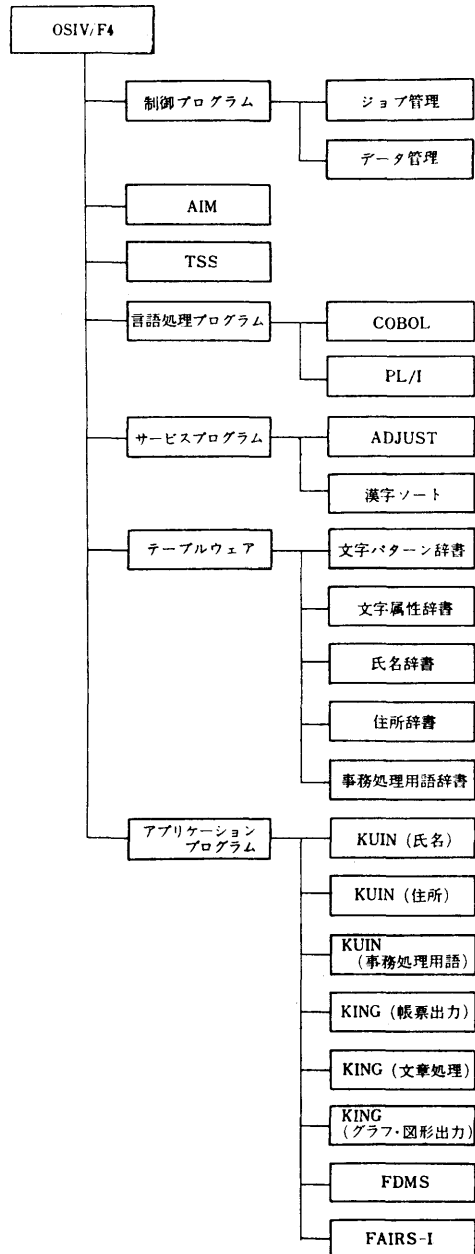


図2.7 J E Fのソフトウェア体系

部分、あるいは新規に開発された部分がある。図 2.7 に、J E F のソフトウェアの体系を示す。以下、これらについて概観する [1]。

2.4.1 制御プログラム

日本語処理装置を英数字・カナ用の装置と同じインターフェイスでサポートするために、制御プログラムの中核であるジョブ管理・データ管理に付加された機能として、文字セット管理機能、コピー機能、フォームオーバレイ機能がある。文字セット管理機能とは、日本語処理に特有な多種多様の文字パターンのローディングを管理・支援する機能で、その概念図を図 2.8 に示す。

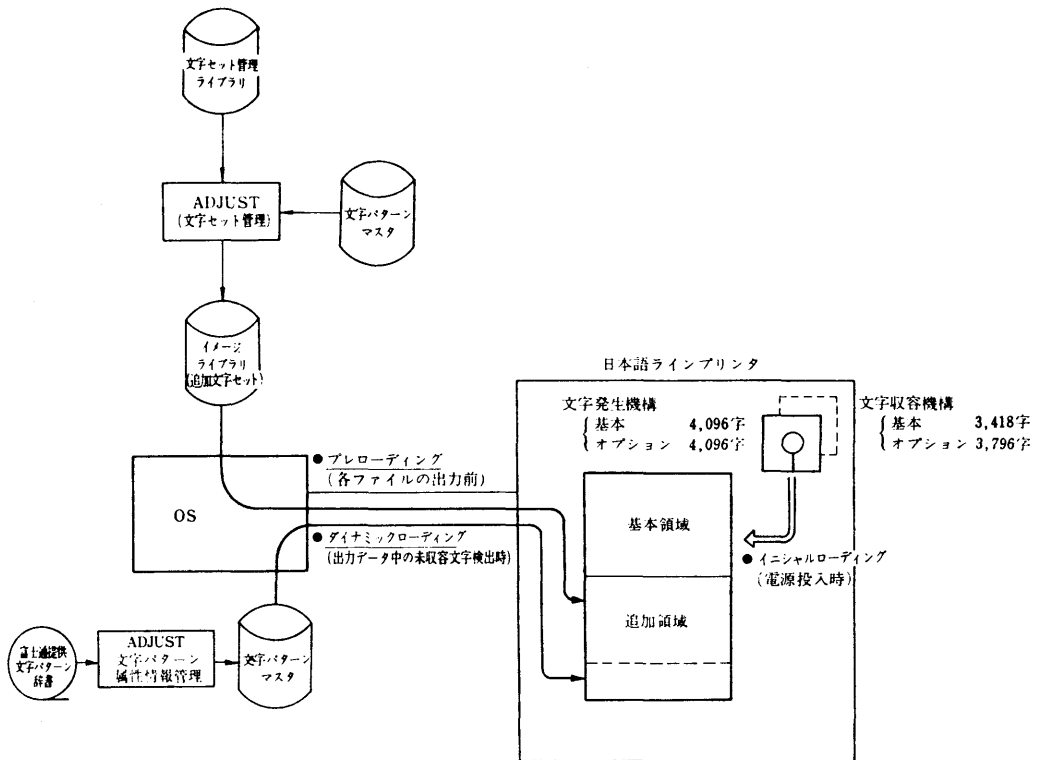


図 2.8 J E F における文字セット管理機能

プレローディングとは、各業務に固有な文字パターンを追加文字セットとして、イメージライブラリに登録しておき、DD文のCHARSパラメータでその追加文字セット名を指定することにより、出力に先立って、それらをローディングしておくものである。ダイナミックローディングとは、文字発生機構に収容されていない文字パターンを、そのコードをキーとして文字パターンマスタファイルを検索し、出力中にローディングして出力処理を続行するもので、DD文のOPTCDパラメータにより指示する。

コピー機能については、従来と同じくDD文のCOPIESパラメータで指定するが、機能が拡張されており、データセット単位およびページ単位のコピーが可能である。帳票などの作成に欠かせな

いフォームオーバレイ機能について、概念図を図 2.9 に示す。ADJUSTにより登録したフォー

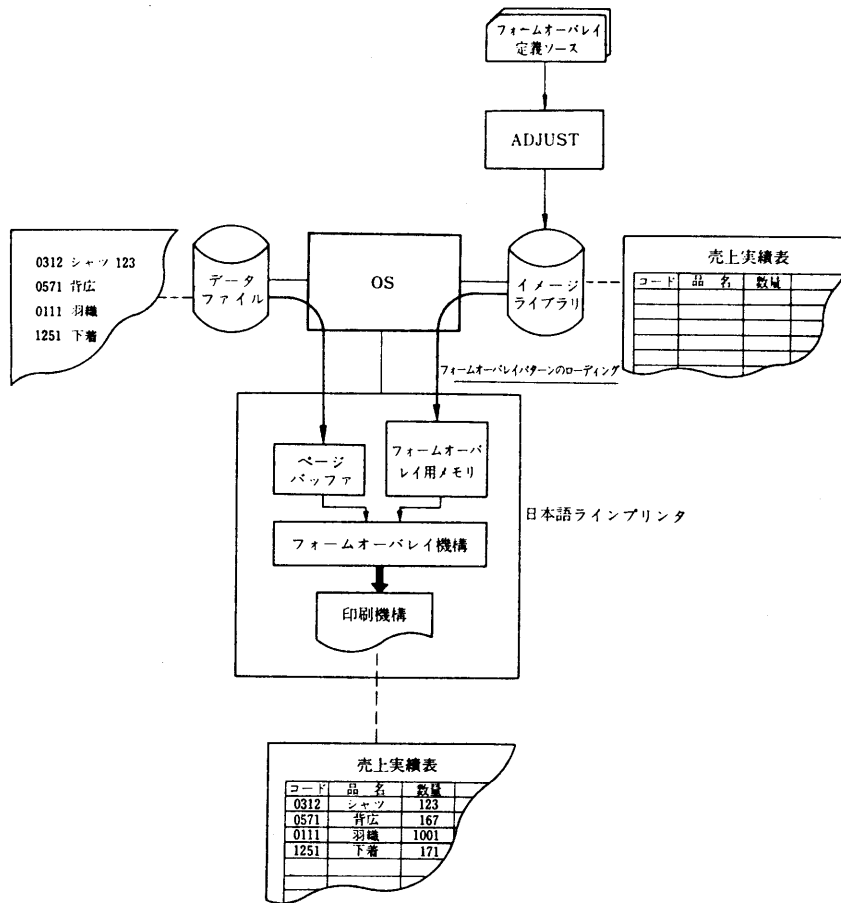


図 2.9 J E Fにおけるフォームオーバレイ機能

ムオーバレイパターンを、出力データと重ねて印刷するもので、DD文のFLASHパラメータで指示する。

2.4.2 言語処理プログラム

言語処理プログラムPL/I, COBOLにおいて、日本語処理機能がサポートされる。これについては、5で解説されるので、ここではふれない。なお、まもなくFORTRANでも日本語処理機能がサポートされる予定である。

2.4.3 サービスプログラム

JEFの導入、運用に伴う日本語処理に特有な作業を支援するために、サービスプログラムADJUST (Advanced Japanese User Support software) および漢字ソートが提供される。これによって、たとえば、センターに固有な文字パターンを登録したり、それに属性情報を与えたり、ダイナ

ミックローディング時の未定義文字を解析して、文字収容機構に登録し、システムの負荷を軽減したり、フォームオーバーレイパターンを作成する等の作業をスムーズに行うことができる。ADJUSTは、図2.10に示す各種ユーティリティプログラムおよびサブルーチンプログラムから構成される。漢字ソートは、部首、総画数、音・訓の読みなどの文字属性による、日本語特有のソーティングをサポートする。

2.4.4 テーブルウェア

テーブルウェアとは、多様な文字パターン、文字属性およびカナ漢字変換のための各種の日本語情報を格納した辞書ファイルの総称である。これには、文字パターン辞書、文字属性辞書、氏名辞書、住所辞書、事務処理用語辞書が含まれる。文字パターン辞書には、JEF漢字コード約14,000字種のうちのユーザ定義文字約3,000字種を除く、約11,000字種の文字パターンが収録されている。文字パターン辞書は、文字パターンマスタファイルとしてシステムに組み込まれ、前述のプレローディングのためのADJUSTによる追加文字セットの作成、ダイナミックローディングなどのために使用される。文字属性辞書には、同じく約11,000字種の漢字の各種属性、すなわち、音・訓の読み、総画数、部首コード、異体字コードなどが収録されている。文字属性辞書は、文字属性マスタファイルとしてシステムに組み込まれ、漢字ソートなどに使用される。氏名、住所辞書は、大量の氏名、住所データのカナ漢字変換のための辞書で、氏名辞書には約32,000件、住所辞書には約180,000件のデータが収録されている。事務処理用語辞書は、帳票等に用いられる見出し項目などの用語のカナ漢字変換のための辞書で、約4,000件のデータが収録されている。これは、PL/I、COBOL、ADJUST、KINGなどで使用される。

2.4.5 アプリケーションプログラム

日本語情報処理を行う利用者に最も密着した、アプリケーションプログラムとして、JEFでは、KING (Kanji Information Generator)、FDMS (FACOM Document Management System)、KUI (Kanji Utility for Input) が提供される。KINGは、NLP 出力支援のためのプログラムで、漢字かな混りの帳票出力をサポートするKING (帳票出力)、グラフ・図形出力をサポートし、NLPの新しい応用分野を開拓したと言われるKING (グラフ・図形出力)、日本語テキストのフォーマットを行い、種々の文書の形式に整えて出力するKING (文章処理)がある。FDMSには、NDPをTSS端末として、カナ漢字変換方式で日本語文章の入力・編集をサポート

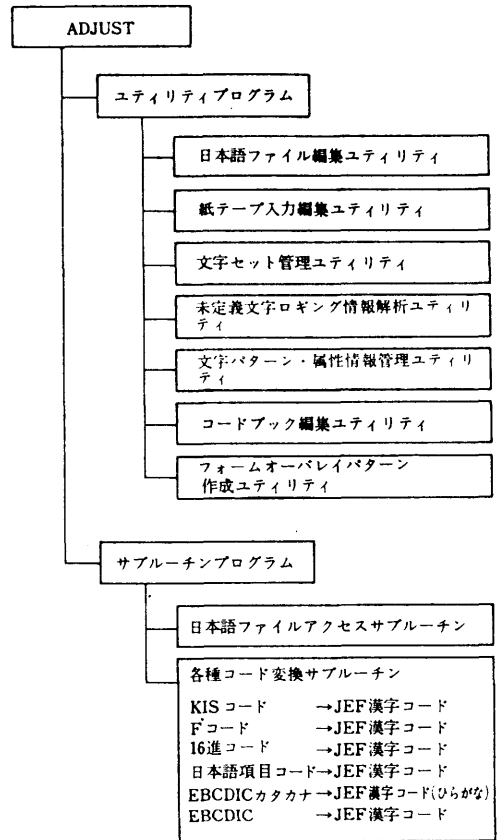


図2.10 ADJUSTのプログラム体系

するFDMS（和文エディタ）がある。KING（文章処理）は、FDMS（和文エディタ）と合わせて用いることにより、グラフ・図形などの図版も組み込んだ、広汎な日本語文書を作成可能にする。KUINは、KUIN（氏名）、KUIN（住所）、KUIN（事務処理用語）に分けられ、各々、前述の氏名、住所、事務処理用語の各辞書を参照したカナ漢字変換入力をサポートするプログラムである。

3 FDMS（和文エディタ）の使用法

3.1 概 要

FDMS（和文エディタ）——以下、和文エディタと略称——は、TSSのもとで動作し、端末にはNDPを用いる。その機能は、カナ漢字変換による日本語テキスト入力サポートと、作成されたテキストの種々の編集・管理である。これによって作成された保存テキストは、後述のKING（文章処理）により、体裁の整った日本語文書の形で出力できる。図3.1に、和文エディタとKING（文章処理）の関連図を示す。表3.1には、これらに関連するデータセットの形式と概要を掲げる。ここで、保存テキストおよびマクロライブラリのデータセットは、順・区分の両方の編成が可能である。

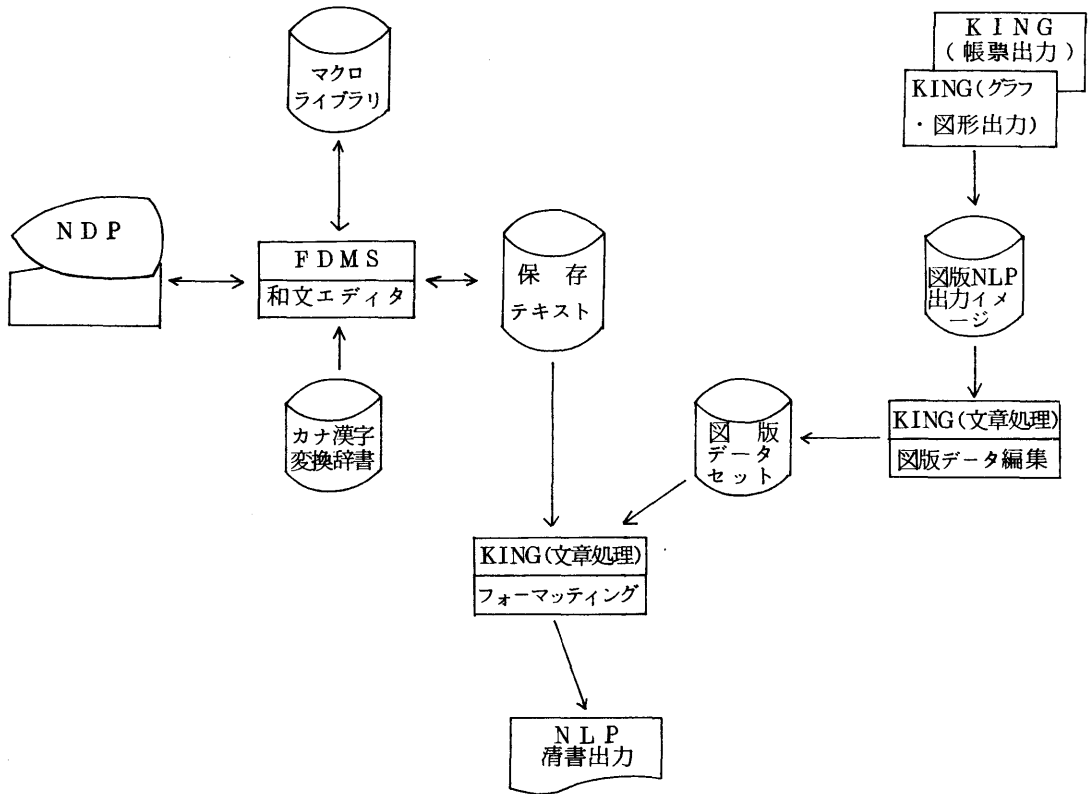


図 3.1 FDMS（和文エディタ）とKING（文章処理）の関連図

表 3.1 FDMS (和文エディタ) と KING (文章処理) で用いられるデータセットの形式と概要

名 称	編成	レコード 形 式	ブロッ ク長 (バイト)	レコー ド長 (バイト)	概 要
保 存 テ キ ス ト	順 区分	可変長	2048	最 大 260	テキストを格納する。各レコードは、検索のための参照番号やパラグラフ情報を持つ。
マ ク ロ ラ イ ブ ラ リ	順 区分	可変長	2048	最 大 260	文字列及び制御コードから成るテキストマクロを格納する。
カナ漢字変換辞書	順	固定長	2000	80	和文エディタのカナ漢字変換で使用される辞書。
図版 NLP 出力 イメージデータ	順	可変長	—	最 大 308	KING(グラフ・図形出力) / JEFなどで作成された図版を格納するNLP出力イメージのデータ。
図版データセット	区分	固定長	612	612	図版NLP出力イメージデータを、NLP出力文書中に組み込める形式に編集したデータセット。

以下に、和文エディタの使用法の概略を示す。

```

LOGON TSS  課題名/パスワード SIZE(768) ..... セッション開始。
          :                                     リージョンサイズは、
          :                                     768KB以上を指定

READY

NEDIT ユーザ指定名 [(メンバ名)] [NEW] [FORMAT] ..... 和文エディタ開始。
          [OLD] [EDIT] ..... テキストデータセット
          の指定。
    
```

```

          フルスクリーン表示、テキストの入力・編集が可能

          END [SAVE] ..... 和文エディタ終了
              [NOSAVE]

READY

LOGOFF ..... セッション終了
    
```

和文エディタには、主としてEDITモードとFORMATモードがあり、画面の表示形式が異なる。これを、図 3.2 に示す。EDITモードでは、行の先頭に、ページ番号、ユニット番号、挿入番号からなる参照番号が表示され、テキストフォーマットのための制御コードを各所に付加することにより、ソースイメージ(漢字かな混り文章と制御コードの混ったイメージ)できめ細かな編集を行うことができる。一方、FORMATモードでは、和文エディタに組み込まれている、KING(文章処理)の簡易版相当のプログラムにより、清書イメージの表示が得られる。ただし、和文エディタでは、付表4の備考欄に示すように、文字サイズ、アンダーラインなどの制御コードの指定は利かない。FORMATモードの表示により、編集結果の確認ができるほか、清書形式の変更、図版などの組み込みのための空白の確保、文章中への表の作成などができる。表示画面の2行目は入力行で、カナ漢字変換のためのカナ文字列の入力、コマンドの入力のために用いられる。3行目はメッセージ行で、カナ漢字変換結果、あるいはシステムからのメッセージが表示される。

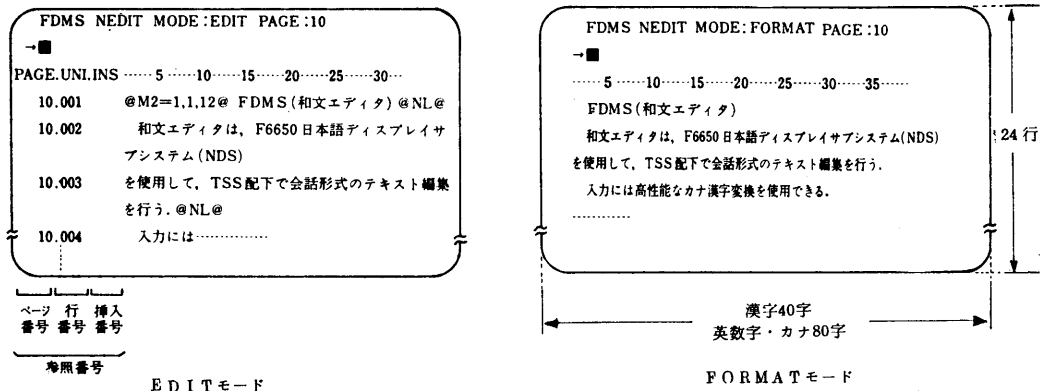


図 3.2 N D P 画面におけるテキストの表示形式

3.2 作成例

学会予稿の原稿作成を目的として、和文エディタで作成し、K I N G (文章処理)でN L P出力した例を図3.3に示す。通常、学会予稿原稿は、学会で定められた原稿用紙に、各講演者が手書きしたものを写真印刷することが多いので、その品質はあまり良くない。論文によっては、高度な内容にもかかわらず、読むことをためられる例もないことはないと思われる。このような学会予稿、その他、研究会・シンポジウム等の論文集などの高品質化、標準化のために、日本語処理システムを用いて計算機処理することは、有効な一つの手段であると思われる。そのためには、そのような雰囲気をつくるよう、学会に働きかける必要があるかも知れない。J E Fでは、漢字コードの英数字を出力すると間延びした感じになるが、これに対してピッチ詰め機能がある。しかし、図3.3に見るように、和文エディタの現在のバージョンでは、英小文字に対してはピッチ詰めが行われない。近い将来、レベルアップされることを期待する。

図3.4には、図3.3に対して、テキストフォーマットされる前の、保存テキスト(これには、ソースイメージで格納されている)の内容の出力例を示す。これから、テキストフォーマットの概要が読みとっていただけるものと思う。この出力に関しては、通常のL I S Tコマンドによる、T S S S Y S O U T出力では不可能(文字化け等が起こる)なので、センターで作成したコマンドを用いた。まもなく利用者に公開できると思われるので、センターニュース等に注意されたい。図3.5には、同じく和文エディタで作成した表の例を示す。

アミノ酸置換に伴う蛋白質の構造転移の動的挙動

九大大型計算機センター^{*}、昭和大教養、九大理・物理^{**}

○武富 敬^{*}、加納 文品^{**}、郷 信広^{***}

目的 蛋白質の立体構造の転移と揺ぎの動的過程を、統計力学的な観点から理解することは、蛋白質の機能や構造形成の機構などの解明のために不可欠である。この目的のために、格子模型として抽象したモデル蛋白質の計算機実験¹⁾を通して、その動的挙動を時間相関関数の方法で解析した。ここでは特に、アミノ酸置換に伴う平衡のおよび動的性質への影響について述べる。

方法 蛋白質の構造の秩序度 P を導入し、これに関する時間相関関数 (TCF) を次のように定義する。

$$\varphi_p(t) = \frac{\langle P(0)P(t) \rangle - \langle P \rangle^2}{\langle P^2 \rangle - \langle P \rangle^2} \rightarrow \sum_{i=1}^M \alpha_i \exp(-t/\tau_i)$$

ここで $\langle \dots \rangle$ は熱平衡平均を表わし、 τ_i は緩和時間、 α_i は緩和強度、 M はモードの数を表わす。 P としては、長距離秩序度または回転半径をとった。また、アミノ酸置換は、近距離相互作用¹⁾の局所的な不整 (連続する2つのユニットの不整) としてモデル化し、アミノ酸置換の個所が異なる2つのモデルを考察した。

結果と考察 **平衡的性質** アミノ酸置換に伴い、構造の安定性が減少する。すなわち、転移温度が下がる。その下がり方は、アミノ酸置換の個所に依存し、構造依存性を持つ。これらは、van't Hoff plot あるいは転移曲線よりみてとれる。

動的性質 1) TCF は少なくとも2つのモード (速いモードと遅いモード) から成るように見える。ここでは、長距離秩序度、回転半径のいずれに対しても2つの指数関数の和として解析した。速いモードの緩和時間を τ_{fast} 、遅いモードのそれを τ_{slow} とする。

2) τ_{slow} は秩序度によらず、誤差の範囲で一致する。すなわち、これは立体構造転移の過程に対応すると見なせる。

3) τ_{fast} は長距離秩序度で見た方が、回転半径のそれよりも一桁大きい。また τ_{fast} には、ほとんど温度依存性は見られない。

4) τ_{slow} の温度依存性を各モデルについて図1に示す。 τ_{slow} は転移点付近で peak を持つ際立った温度依存性を示すが、

peak の位置は転移温度よりわずかに低い方にずれる傾向があるようである。図1より、アミノ酸置換に伴って τ_{slow} が大きくなる。すなわち、転移が遅くなること

がわかる。また、この様子は、アミノ酸置換の個所に依存し、構造依存性を持つ。

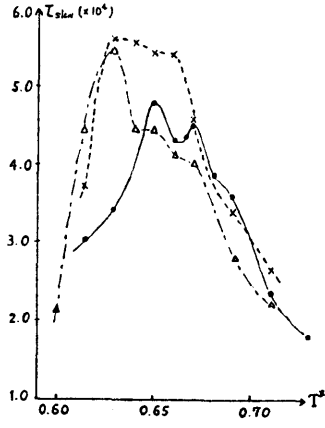


図1. τ_{slow} の温度依存性
 ●—●—●— アミノ酸置換のないモデル
 ×—×—×—×— アミノ酸置換モデル1
 △—△—△—△— アミノ酸置換モデル2

1) Go, N. & Taketomi, H. (1978) Proc. Natl. Acad. Sci., USA **75** 559-563

2) Taketomi, H., Kano, F. & Go, N., to be published

図3.3 F D M S (和文エディタ)による学会予稿原稿の作成例

PAGE UNIT INS

T E X T

0001.0001.000 @FM=C@@SZ=12@アミノ酸置換に伴う蛋白質の構造転移の動的挙動@NL@@NL@
 0001.0002.000 @FM=R@@UP=3@@SZ=9@九大大型計算機センター，昭和大教養，九大理・物理，@NL@@UP=5@
 0001.0003.000 武富 敬，加納 文晶，郷 信広@NL@@NL@
 0001.0004.000 @FM=L@@UP=0@@UL@@DN=1@目的@UE@蛋白質の立体構造の転移と揺ぎの動的過程を，統計力学的な観点から理解することは，@DN=0@蛋白質の機能や構造形成の機構などの解明のために不可欠である．この目的のために，格子模型として
 0001.0005.000 抽象したモデル蛋白質の計算機実験を通して，その動的挙動を時間相関関数
 0001.0006.000 の方法で解析した．ここでは特に，アミノ酸置換に伴う平衡のおよび動的性質への影響について述べる．@NL@
 0001.0007.000 @UL@@DN=1@方法@UE@蛋白質の構造の秩序度を導入し，これに関する時間相関関数(TCF)を次のように@DN=0@定義する．@NL@@NL@ここで<...>は熱平衡平均を表わし， τ は緩和時間， ω は緩和強度， ν はモードの数を表わす．としては，長距離
 0001.0007.001 離
 0001.0008.000 秩序度または回転半径をとった．また，アミノ酸置換は，近距離相互作用の局所的な不整(連続する2つのユニットの不整)としてモデル化し，アミノ酸置換の個所が異なる2つのモデルを考察した．
 0001.0009.000 @NL@@UL@@DN=1@結果と考察@UE@@SZ=12@平衡的性質@SZ=9@アミノ酸置換に伴い，構造の安定性が減少する，すなわち，転移温@DN=0@度が下がる．その下がり方は，アミノ酸置換の個所に依存し，構造依存性を持つ．これらは，@PT@v a n' t H o
 0001.0009.001 f f p
 0001.0009.002 l o t
 0001.0009.003 @PE@
 0001.0010.000 あるいは転移曲線よりみてとれる．@NL@@SZ=12@動的性質@SZ=9@1 TCFは少なくとも2つのモード(速いモードと遅いモード)から成るように見える．ここでは，長距離秩序度，回転半径のいずれに対しても2つの指数関数の和として解析し@SK=10,16R@た．
 0001.0011.000 速いモードの緩和時間を τ_1 ，遅いモードのそれを τ_2 とする．2は秩序度によらず，誤差の範囲で一致する．すなわち，これは立体構造転移の
 0001.0011.001 過程に対応すると見なせる．3は長距離秩序度で見た方が，回転半径のそれよりも一桁大きい．また τ_1 に
 0001.0011.002 は
 0001.0011.003 ，
 0001.0012.000 ほとんど温度依存性は見られない．
 0001.0013.000 4の温度依存性を各モデルについて図1に示す． τ_1 は転移点付近で@PT@p e a k @PE@を持つ際立った温度依存性を示す
 0001.0014.000 が，
 0001.0015.000 @PT@p e a k @PE@の位置は転移温度よりわずかに低い方にずれる傾向があるようである．5図1より，アミノ酸置換に伴って τ_1 が大きくなる，すなわち，転移が遅くなることがわかる．
 0001.0016.000 また，この様子
 0001.0017.000 は，アミノ酸置換の個所に@NL@
 0001.0017.001 依存し，構造依存性を持つ．@NL@@NL@
 0001.0018.001 @PT@l i G o , N . & T a k e t o m i , H . (1 9 7 8) P r o c . N a t l . A c a
 d . S c i . , U S A 7 5 5 5 9 - 5 6 3 @NL@ 2 T a k e t o m i , H . , K a n o ,
 F . & G o , N .
 0001.0019.001 , t o b e p u b l i s h e d @PE@

図3.4 図3.3に対するソースイメージの日本語テキスト．図3.3は，図版組み込みのため8LPIで出力したので，この図に見られる

12ボ文字の制御コードは利いていない．

物理回線群名	論理回線群名	回 線 名	端末名	学校名	設置場所	MDF	回線 番号	CS2	設 置 年月日
GROUP#50 TTY 1200BPS	LINEG#50	LINE#500	T500	九 大	情 報	19.20	38	#0	53.03.09
		501	501	九 大	電 子	17.18	39		53.09.27
		502	502	九 大	情 報	15.16	3A		53.09.26
		503	503	九 大	情 報	13.14	3B		54.02.16
		504	504	九 大	中央計数	/	3C		54.04.06
		505	505	九 大	電 子	7.8	3D		54.06.15
		506	506	九 大	情 報	9.10	3E		54.09.25
		507	507	九 大	電 子	5.6	3F		54.07.13
		508	508	九 大	情 報	1.2	40		54.11.05
		509	509	九 大	数 学	49.50	41		55.07.21
GROUP#50 TTY 1200BPS	LINEG#51	LINE#510	T510	九 大	造 船	47.48	42	#0	54.12.14
		511	511	九産大	計算機室	25.26	43		
		512	512	九 大	大学総合	38.39	44		54.12.05
		513	513	九 大	物 理	36.37	45		54.12.19
		514	514	九 大	心 理	27.28	46		55.03.14
		515	515	九 大	生 研	29.30	47		55.02.01
		516	516	九 大	応 研	69.70	48		55.02.06
		517	517	九 大	航 空	77.78	49		55.02.21
		518	518	九 大	情 報	55.56	4A		55.04.08
		519	519	九 大	生 物	93.94	4B		55.05.11

図 3.5 FDMS (和文エディタ)による表の作成例

3.3 カナ漢字変換

前述したように、日本語情報処理の技術的障害の最大のものの一つに、漢字入力の問題があるとされてきたが、和文エディタでは、普通の利用者にも最も使いやすい入力方法であろうと思われる、カナ漢字変換方式を採用している。これは、文章をカナで入力し、カナ漢字変換辞書による照合、文法接続チェックなどの計算機による処理を行って、漢字かな混り文へ変換する方式である。これに関心のある読者は、文献[2, 3]を参照されたい。この方式を実現する種々の手法のうち、JEFでは最尤評価法と二方向文法という手法を採用している[6]。また、和文エディタに組み込まれているカナ漢字変換マスタ辞書には、約10万語の単語が収録されている。しかし、日本語の特徴である多数の同音異義語のために、この方式による完全なカナ漢字変換は不可能である。たとえば、「キノウ」に対して、「機能」、「昨日」、「帰納」、「帰農」とあることから明らかであろう。ちなみに、日本語では、同音異義語の出現度は平均2.32語であると言われる。^{*}この同音異義語の選択の問題は、

最終的には文章の意味解析の研究の進展にまたねばならないであろう[4]。

このカナ漢字変換の効率、精度を向上させるために、利用者がどのような点に留意すべきか、あるいは、現和文エディタに備わっている機能をどう使うかについて以下に記す。

第1は、利用者が入力の際に分かち書きを考慮することである。たとえば、「花を渡します」と入力したい時に、現和文エディタではどのように変換されたか、その一例を以下に示そう。現和文エディタでは、カナと共に、ローマ字表記による入力も可能なので、以下、後者を用いることにする。

HANAWOWATASHIMASU	} → は名を私増す
HANAWO WATASHI MASU	
HANA WO WATA SHI MASU	→ はなを渡します
HA NA WO WA TA SHI MA SU	→ はなをわたし間好

第2は、同音異義語は避けられないとして、入力時にそれを次の候補に変換する機能(入力時再変換)、あるいは、いったん入力した後でコマンドでそれを再変換する機能を有効に使うことである。また、同音異義語が多数あると思われる場合には、後述のSHOWコマンドでそれらのメニュー表示を出しておいて選択する方法もある。しかし、いずれにせよ、入力時には少々誤って入力しても、後でそれを校正・編集するという方向で作業を進める方が能率が上がる。第3は、現和文エディタに備わっている暫定辞書の機能を有効に使うことである。暫定辞書の機能とは、たとえば、前述の再変換などにより探り当てた同音異義語の選択語を、以後は優先して処理するようにすることである。ただし、この機能は同一EDIT内でしか働かない。自分の望む選択語を後々まで保存しておくには、次の方法を用いればよい。1つは、望む選択語の入った漢字かな混り文をテキストマクロとしてマクロライブラリに登録しておき、必要時それを引用する方法である。もう1つは、カナ漢字変換マスタ辞書は書替え不可なので、個人で私的カナ漢字変換辞書を作成しておき、それに登録する方法である。大学のように、専門分野ごとに専門用語が分化している所では、これらの機能は重要であると思われる。

3.4. テキスト入力法

テキスト入力から表示までのデータの流れを、図3.6.に示す。これから分かるように、EBCDICにより入力したカナあるいはローマ字文字列などは、制御コードを除き、カナ漢字変換により、すべてJEF漢字コードの漢字かな混り文に変換される。変換されずにEBCDICのまま残るのは、制御コードだけである。こうして、EBCDIC、JEF漢字コード混在の保存テキストが作成される。ここでもシフトコードを意識する必要はない。

テキストの入力に際して、カナ漢字変換を行わず、数字・英字のまま残しておきたい場合、カタカナやひらがなで残しておきたい場合などもありうる。このための指定を変換指定といい、以下にその例を示す。i)のように何も指定しなければ、カナ漢字変換される。ii)は、ブランクの入力で、1

* これは、「松下、山崎、佐藤：漢字かな混り文変換システム、情報処理、vol. 15, No. 1, pp. 2-9, 1974」より引用した。しかし、この論文では、サンプルした単語が1万語程度とそれほど多くなく、またその出現頻度の一様性を仮定しているなど問題がないわけではないので、参考の数値と考えていただきたい。

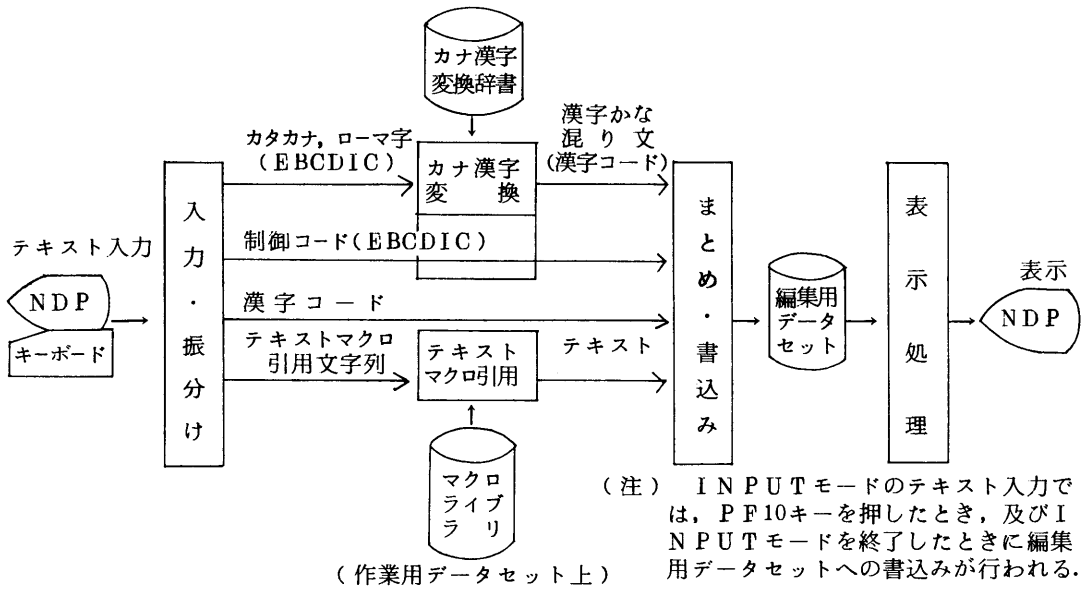


図 3.6 テキスト入力から表示までのデータの流れ

個の空白は分ち書きの区切りと見なされるため、 $n+1$ 個の空白は、 n 個の空白に変換される。iii) は、無変換指定で、EBCDICの数字・英字をそのまま、JEF漢字コードの数字・英字に変換する場合である。この際、英字については#記号で、大文字、小文字の反転の指定が可能である。iv) は、カタカナ、ひらがな指定である。v) は、SHOWコマンドで各種見出しのメニュー表示を出しておき、該当する番号を指定することにより、正しくカナ漢字変換を行うための指定である。vi) は、漢字の音・訓の読みに対する指定である。いずれにせよ、この変換指定は正しいカナ漢字変換を得るための1つの方法である。なお、参考のため、付表5に変換指定についてまとめた。

i) 変換指定をしない入力

KYOU □ HA □ YO I □ TENKI □ DESU. → 今日はいよい天気です。

ii) □ (空白) の入力

HANA □ □ □ □ GA □ □ SAKU. → 花□□□が□咲く。

iii) 無変換指定 / /

/ FACOM □ M - 200 / → FACOM □ M - 200
 / C # O M P U T E R □ # C # E N T E R / → Computer □ Center
 / FDMS // J E F / → FDMS / J E F

iv) { カタカナ指定 K / /
 { ひらがな指定 H / /

K / T A B U C H I / □ H / K U N / → タブチくん
 K / K O N P Y U - T A / → コンピュータ

v) $\left\{ \begin{array}{l} \text{単語 指定 } T / \text{見出し} [\cdot n] / \\ \text{地名 指定 } C / \text{見出し} [\cdot n] / \\ \text{人名 指定 } J / \text{見出し} [\cdot n] / \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \text{SHOW} \\ \text{SH} \end{array} \right\} \nabla \text{見出し} \left\{ \begin{array}{l} \cdot T \\ \cdot C \\ \cdot J \end{array} \right\} \nabla$
 メニュー表示を出しておく。
 (注. INPUTモードでは不可)

SH ∇ KINOU. T ∇ で

メニュー表示 1.機能 2.昨日 3.帰納 4.帰農 と出た場合、

SUUGAKU \sqcup TEKI \sqcup T / KINOU. 3 / \sqcup HOU \rightarrow 数学的帰納法

vi) $\left\{ \begin{array}{l} \text{音 指 定 } P / \text{音読み} [\cdot n] / \\ \text{訓 指 定 } R / \text{訓読み} [\cdot n] / \end{array} \right. \Rightarrow \left\{ \begin{array}{l} \text{SHOW} \\ \text{SH} \end{array} \right\} \nabla \text{見出し} \left\{ \begin{array}{l} \cdot P \\ \cdot R \end{array} \right\} \nabla$
 メニュー表示を出しておく。

SH ∇ A I. P ∇ で

メニュー表示 1.哀 2.挨 3.愛 と出た場合

P / A I. 3 / \sqcup GA \sqcup A R E \sqcup B A \rightarrow 愛があれば

SH ∇ KOTO. R ∇ で

メニュー表示 1.事 2.言 3.異 と出た場合

P / KOTO. 3 / \sqcup N A R U \sqcup M O J I \rightarrow 異なる文字

次に、ローマ字表記による入力の場合、注意しなければならない点をあげる。

1. 「ン」を表わす「N」と、次にくる母音字または「Y」とを切り離す場合は、「N」の後に「X」を入れる。

例. K I N X E N (禁煙) \leftrightarrow K I N E N (記念)

K A N X Y U U (勧誘) \leftrightarrow K A N Y U U (加入)

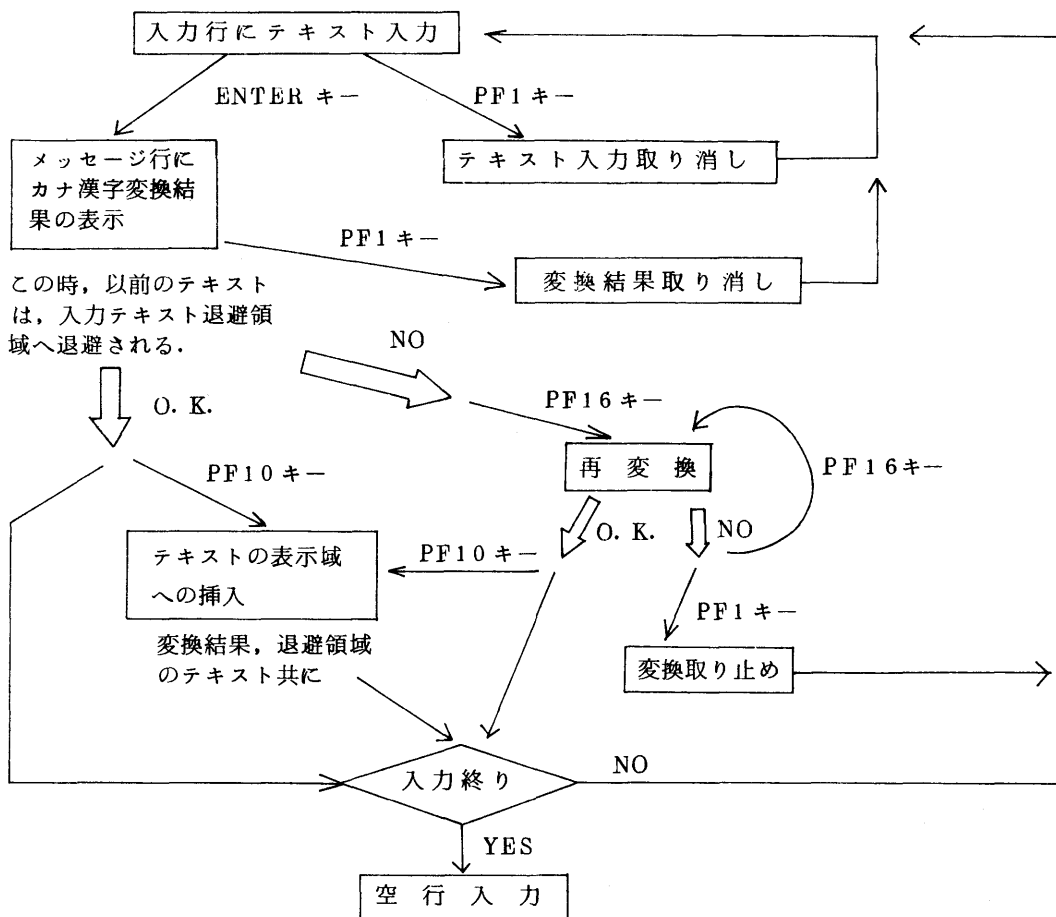
2. 促音の「ッ」は、次にくる子音字を重ねる。

例. H A P P Y O U (発表)

なお、参考のため、付表6に、カタカナ・ローマ字対応表を掲げる。

3.5. テキスト作成法

テキストを新規に作成する場合には、NEDITコマンドでNEWオペランドを指定すれば、INPUTモードになり、入力行からテキスト入力が可能となる。追加作成する場合には、FORMATモードかEDITモードで表示されている時に、入力行にINPUTコマンドを入力して、INPUTモードにすればよい。INPUTモード時の、テキスト作成の流れを以下に示す。



INPUTモード終了して{ $\frac{F}{E}$ }モードに戻る。

ここで、再変換とは、前述の入力時再変換のことで、テキスト入力時、PF16キー押下により、メッセージ行に次々に同音異義語を表示する機能である。以下に例を示す。

「校正を」と入力したい時

---	K O U S E I W O	入 力 行
	↓		
	ENTER		
	構 成 を	メ ッ セ ー ジ 行
	↓		
	PF16		
	公 正 を	メ ッ セ ー ジ 行
	↓		
	PF16		
	校 正 を	メ ッ セ ー ジ 行
	↓		
	O.K.		

次のテキストを入力する

3.6. 種々のテキスト編集法

付表3に和文エディタにおけるコマンド入力形式一覧を掲げる。以下の説明において詳しく述べない部分は、これを参照されたい。EDITモードにおいては、従来のEDITのFSOと同じように画面上で、文字の置換え、挿入（挿入キーによる）、削除（削除キーによる）が可能である。なお、以下で、1文字編集の場合にはSの付いたコマンド（XSなど）を入力する。文字列編集の場合には、文字列の最初と最後に、たとえば「X〇〇」Xのように入力する（以下では、これを「X-」Xのように表わしている）。

1) 再変換による置換え編集 (「X-」X (exchange) コマンド)

同音異義語の誤まった文字列を置換えたい場合などに用いる。

使用例：構成 → 校正と置換えたい場合

```

原文：  原稿の構成をする          . . . . . 表 示 域
        ↓ 「X-」Xコマンド入力
        原稿の「X」Xをする          . . . . . 表 示 域
        ↓ ENTER
        公 正                        . . . . . メッセージ行
        ↓ PF16
        校 正                        . . . . . メッセージ行
        ↓ ENTER
修正文： 原稿の校正をする          . . . . . 表 示 域
    
```

2) 文字列を入力テキストに置換える (「R-」R (replace) コマンド)

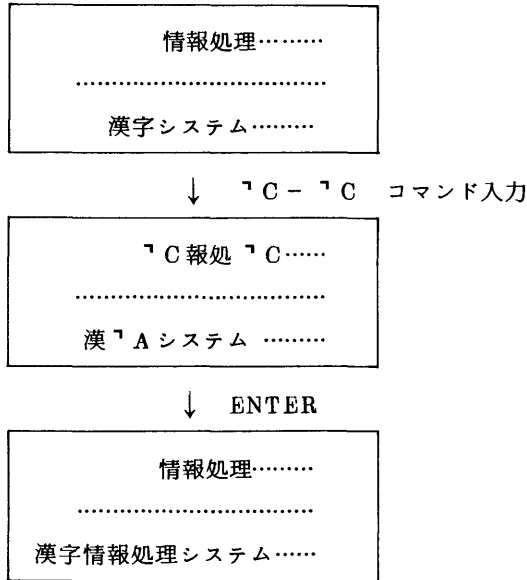
使用例：SHOWコマンドで KINOとKOUSEIのメニュー表示既知とする。

```

原文：  機能は、原稿の構成をした . . . . . 表 示 域
        ↓ 「R-」Rコマンドを文章の最初と最後に入力
        「R」能は、原稿の構成をし「R」 . . . . . 表 示 域
        ↓ ENTER
          INPUT モードになる
        T/KINO. 2 / 〇HA, GENKOU 〇NO 〇
        T/KOUSEI. 3 / 〇WO SHITA
          . . . . . 入 力 行
        ↓ ENTER
        昨日は、原稿の校正をした . . . . . メッセージ行
        ↓ ENTER
修正文： 昨日は、原稿の校正をした . . . . . 表 示 域
    
```

3) 文字列の複写 ($\uparrow C - \uparrow C$ (copy) コマンド) $\left\{ \begin{array}{l} \uparrow A (After) \\ \uparrow B (Before) \end{array} \right.$ と組み合わせる

$\uparrow C$ コマンドの例



4) 文字列の挿入 ($\uparrow I A$ / $\uparrow I B$ (insert) コマンド)

使用例: 「情報システム」の前に「漢字」を挿入する場合

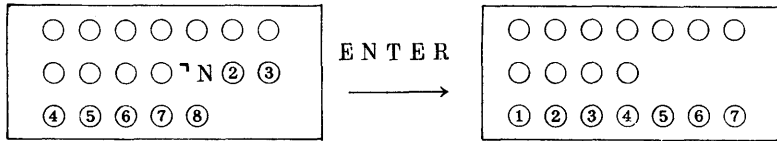
原文:です. 情報システムに表示域
	↓ $\uparrow I B$ コマンド入力	
です. $\uparrow I B$ 報システムに表示域
	↓ INPUT モード	
	KANJI入力行
	↓ ENTER	
	感 じメッセージ行
	↓ PF16	
	漢 字メッセージ行
	↓ ENTER	
修正文:です. 漢字情報システムに表示域

以上の他に、文字列の削除 $\uparrow D$ (delete) コマンド、文字列の移動 $\uparrow M$ (move) コマンド、他のテキスト内の文字列組み込み $\uparrow G$ (merge) コマンドなどがあるが説明を省く。なお、以上のコマンドは、すべて、EDITモード、FORMATモードのいずれでも使える。

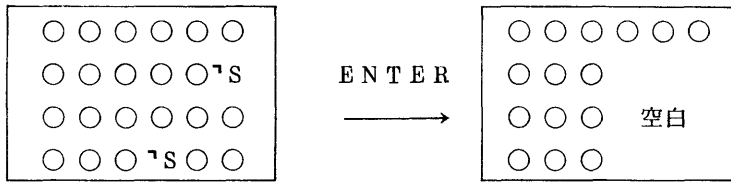
これから説明するコマンドは、書式編集のために使用され、FORMATモードでしか使えない。その理由は、この書式編集コマンドの入力により、NEEDIT時に割当てられた編集用作業データセ

ット上で、そのコマンドに対応した制御コードの付加、あるいは変更を行うからである。どのように付加、変更されたかは、EDITモードに戻してみれば分かる。いくつかのコマンドについて説明する。

5) 強制改行 (^N (new line) コマンド)

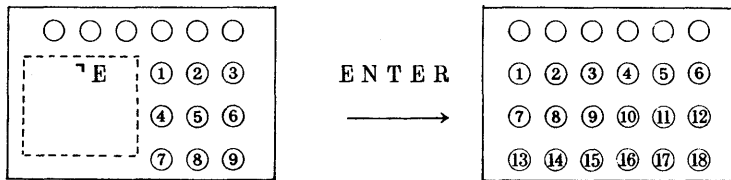


6) 空白の確保 (^S (skip) コマンド)



空白は、本文領域の左端と右端のいずれか一方、あるいはその両方に接するように指定しなければならない。

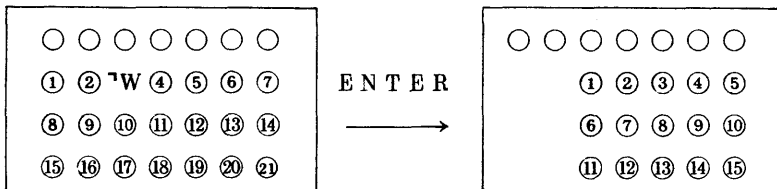
7) 空白の削除 (^E (erase) コマンド)



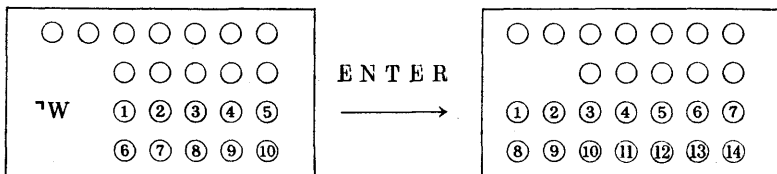
空白のどこでもよい

8) 行頭の字下げ (^W (down) コマンド)

字 下 げ



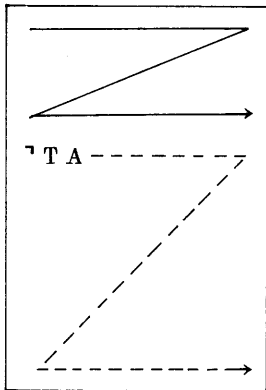
字下げの解除



次に、書式編集の一つとして、表を作成するコマンドについて概説する。表の行方向をレコード、列方向をカラムと呼ぶ。下図に示すように、表を作成したい位置に「T A」コマンドを入力し、ENTERキーを押すと、メッセージ行に表の形式問合わせが出てくるので、それを指定してやればよい。

9) 表の作成 (「T A (table add) コマンド)

・「T A」の入力



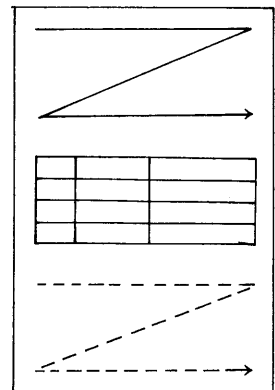
(入力行)

表の形式の指定
(オペランド)

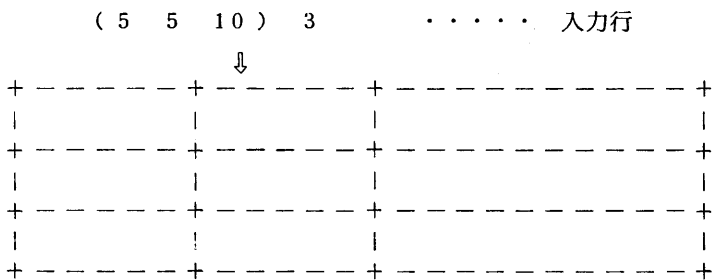
ENTERキー

ENTERキー

・表の表示(テキストなし)



最も簡単には、たとえば、次のように入力行に表の形式を指定したとすると、カラム幅がそれぞれ5, 5, 10文字、レコード数が3の、テキストなしの表枠が得られる。



あとは、今まで述べたテキストの複写、挿入などのコマンドで、表をテキストで埋めればよい。NDP表示上は、罫線が破線になっているが、KING(文章処理)で出力すれば、きれいな実線が得られる。

3.7 テキストマクロの使用法

漢字を含むある文字列に適当な名前(マクロ名)を付けて登録しておき、あとで参照、引用などができる。

テキストマクロ = マ ク ロ 名 + マ ク ロ 定 義 体

英数字4文字以内
(英字で始まる)

236 バイト以内

1) テキストマクロの登録

NEDITコマンドのMACLIBオペランドあるいはMACLIBコマンドにより、前もってマクロライブラリを割当てておく必要がある。

i) DEFMAC コマンド

DEF マクロ名 ▼マクロ定義体▼

マクロ定義体にEBCDICの文字を指定するとカナ漢字変換が行われる。

ii) 'Q (quote) コマンド

編集テキスト中のマクロ定義体として登録したい部分を'Q-'Qで指定して、ENTERキーを押し、その後入力行にマクロ名を入力する。

2) テキストマクロの引用など

入力テキスト中、文字列オペランド中に次の形式で引用できる。

Q/マクロ名/

使用例：Q/D1/ : である。 } と定義されている場合。
 Q/D2/ : です。 }

CHANGE 1.3 1.1 0 ▼Q/D1/▼ ▼Q/D2/▼ALL

↓

CHANGEコマンドは、文字列変更のためのコマンドである。この例では、参照番号1.3から1.10のテキスト中、「である。」調はすべて「です。」調に変えられる。

3.8 私的カナ漢字変換辞書の使用法

1) 私的辞書への単語の登録 ('T (tango update) コマンド)

NEDITコマンドのDICTIONオペランドあるいはDICTIONコマンドにより前もって私的辞書が割当てられていなければならない。

編集テキスト中の、新単語として登録したい部分を'T-'Tで指定して、ENTERキーを押し、その後入力行に見出しを入力する。

4. KING (文章処理) の使用法

4.1 概 要

KING (文章処理) の機能は、図3.1に見られるように、日本語文章の中に組み込むための図版を抽出する図版データ編集と、保存テキストの中に付加されている制御コードを解釈することにより、図版なども含んだ体裁の整った日本語文書をNLPに出力するテキストフォーマットとから成る。制御コードは、書式を整えるための種々の指示を表わし、1段組か2段組かの段組の指示あるいはページの大きさなどを指示するページ形式の指定、字下げ、ピッチ詰め、アンダーライン付加などの指示をする字詰めの指定、ページ内での空白確保および確保した空白への図版出力の指定などを可能にする。この制御コードの入力形式一覧を、付表4に掲げる。

4.2 使用法

1) テキストフォーマット

- ・ 図版出力がない場合 (テキストのフォーマットのみ)

```
// EXEC TEXTFMT
//OLDTEXT DD DSN=F9999.KANJI(EX3),DISP=SHR
//SYSIN DD *
```

<pre>/ PARA LPI=6 @PG=52,66,2,0,9@ @AR=0,0@</pre>	<pre>← PARA 文 ← ページ形式 ← 領域定義</pre>
---	------------------------------------

```
//
```

- ・ 図版出力がある場合 → KNGWTRを指定する

```
// EXEC TEXTFMT,WTR=KNGWTR
//OLDTEXT DD DSN=F9999.KINGT.DATA,DISP=SHR
//ZUHAN DD DSN=F9999.ZUHAN,DISP=SHR
//SYSIN DD *
```

<pre>PARA 文 ページ形式 領域定義</pre>

```
//
```

フォーマットの形式指示のため、SYSINで、PARA文、ページ形式、領域定義を与える。すべて標準値を用いる場合は、SYSIN DD文を省略すればよい。PARA文では、1ページに2ページ分のテキストを割りつける時以外は、行送りの6LPIか8LPIだけを指示すればよい。ページ形式、領域定義については、付表4を参照されたい。ページ形式では、段組の指示により、同じ文章を1段組でも2段組でも出力できる。なお、出力すべき保存テキストに対するDD名がOLDTEXTで、図版データセットに対するそれがZUHANである。

2) 図版データ編集

```
// EXEC ZHNEDT
//ZHNIMG DD DSN=F9999.ZHNIMG,DISP=SHR
//ZUHAN DD DSN=F9999.ZUHAN,DISP=(,CATLG),
// UNIT=PUB,SPACE=(TRK,(10,10,5))
//SYSIN DD *
```

<pre>/ SEL N= 図版名</pre>	<pre>←SEL 文</pre>
-------------------------	-------------------

```
//
```

SYSINのSEL文で、図版NLP出力イメージから図版データを抽出することを指示する。表

4.1に、SEL文の記述形式を示す。図版名は、英字で始まる英数字8文字以内を指定する。この図版名が、図版データセットのメンバ名となる。すなわち、図版データセットは区分データセットでなければならない。内部空白あるいは段頭段末空白制御コードの図版名の部分に、この名前を指示することにより、空白に図版を出力することができる。なお、図版NLP出力イメージに対するDD名がZHNIMGで、図版データセットに対するそれがZUHANである。

表 4.1 SEL文の記述形式

制御文字	名札欄	オペレーション欄	オペランド欄
/		SEL	N=図版名〔, PAGE=ページ番号〕 〔, X=(シフト文字数, 図版文字数)〕 〔, Y=(シフト行数, 図版行数)〕 〔, P={ $\frac{9}{12}$ }〕 〔, LIST={ $\frac{YES}{NO}$ }〕

使用例：

5 ページ目の3行目から4行分、6文字目から4文字分をZHN020という図版名で抽出する例

```
/ SEL N=ZHN020, PAGE=5, X=(5, 4), Y=(2, 4)
```

上記で示した図版を内部空白に出力するときの制御コードの例。

```
@SK=3, ZHN020@
```

4.3 備 考

現在、KING（文章処理）の実行は、バッチ処理あるいはTSSのFIBジョブによるしか手がないが、NLPおよびKINGライタを割り当てるコマンドをセンターで作成したので、これらすべての処理がTSSのコマンドで可能になる予定である。センターニュース等に注意されたい。

5. PL/I, COBOLにおける使用法

5.1 PL/Iにおける日本語処理

従来のPL/IコンパイラにオプションとしてJEF機能が追加された。

利用者は日本語処理を行うステートメントを含む原始プログラムを入力して、JEFオプションの指定を行いコンパイルする。ここで、日本語項目変換辞書は原始プログラムに日本語定数の指定がある場合に、カナ漢字変換用の辞書として参照する。

日本語処理のサポートで追加されたコンパイラオプションの機能と、EXEC文のパラメタの指定方法を次に示す。

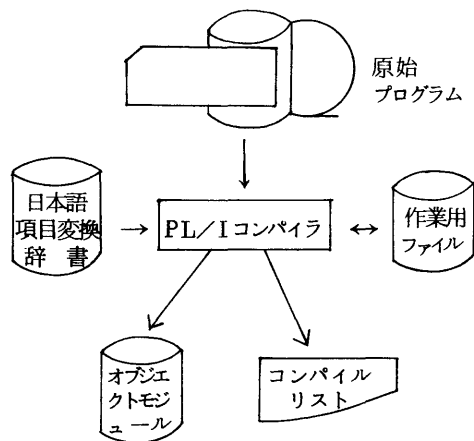


図 5.1 PL/Iコンパイラの処理形態

$$PARM = \left\{ \begin{array}{l} JEF \\ NOJEF \end{array} \right\} \left\{ \begin{array}{l} JCONST \langle (N|X) \rangle \\ NOJCONST \end{array} \right\} \left\{ \begin{array}{l} ASOURCE \\ NOASOURCE \end{array} \right\}$$

↑
日本語データの処理機能を使うかどうかを指定する。

↑
JEFオプションの指定時に有効。日本語定数をJEF漢字コードに変換した一覧表を日本語(N)又は16進数(X)で印刷するかどうかを指定する。

↑
JEFオプションの指定時に有効。入力原始プログラムを出力するかどうかを指定する。

5.1.1 日本語データ処理機能

PL/Iコンパイラでは、日本語データを扱うために以下の属性や定数表現が設けられた。

- ・日本語文字列変数であることを示すNCHARACTER属性。(省略形 NCHAR)
- ・日本語定数 **表 5.1 参照**
- ・日本語データを印刷する際の文字の大きさを示すMODE属性。
- ・日本語データ出力ファイルであることを定義するENVIRONMENT属性に指定するJEFオプション。

次に各機能の詳細について述べる。

(1) 日本語文字列変数

原始プログラムの中でNCHAR属性を指定するとその変数がJEF漢字コードを扱うことを意味する。記述形式：

変数名 NCHAR〔長さ〕

長さ属性では文字数を指定し、実際の記憶領域は文字数の2倍のバイト数分を占有する。指定できる最大値は16,383で、最小値は0、省略すると1(2バイト)とみなされる。

例) DCL MOJI1 NCHAR(10), ...①

MOJI2 NCHAR(3) INIT(ケイサンキ NN);...②

①は変数MOJI1が10文字(20バイト)の日本語文字列であることを示し、②は“ケイサンキ”が日本語項目変換辞書より“計算機”と変換されてMOJI2へ代入される。

日本語文字列変数は後述の日本語定数と共に、データの型変換が不要な次の文脈で指定できる。

- ・レコード変数
- ・INITIAL属性の項目
- ・代入文
- ・比較式
- ・アーギュメント、パラメータ
- ・ALLOCATE文、FREE文
- ・レコード転送文のキー
- ・RETURN文
- ・列処理用の組込み関数/擬変数であるPEPEATやUNSPEC

使用上の注意としてVARYING属性が指定できないこと、NCHAR属性を含む構造体のレ

ベルは14以下であることなどがあげられる。

(2) 日本語定数

日本語定数は表5.1に示すとおり6種類の指定方法があり、どの日本語定数もコンパイル時にJEF漢字コードに変換される。

表 5.1 日本語定数一覧と記述例

No.	日本語定数の種類	記述形式	変換処理	記述例	変換結果
1	日本語項目定数	'EBCDICの文字列'NN	指定された文字列をキーに日本語項目変換辞書(事務処理用語辞書等)を検索してJEF漢字コードに変換する。	'シメイ'NN	'氏名'
2	日本語英数字定数	'EBCDICの文字列'NA	指定されたEBCDICの英数字カナを、同じ文字を表わすJEF漢字コードに変換する。	'リスト'NA	'リスト'
3	日本語連想定数	'カタカナ文字列'NK	KISコードで指定されたカタカナ2文字に対応する漢字のJEF漢字コードに変換する。	'カフシキカイシャ'NK	'株式会社'
4	日本語ひらがな定数	'カタカナ文字列'NH	EBCDICのカタカナで指定された文字列をJEF漢字コードのひらがなに変換する。	'フリカ'ナ'NH	'ふりがな'
5	16進文字列定数	'16進数文字列'NX	指定された16進数をそのままコードに変換する。	'BDEAC2B0'NX	'所属'
6	日本語コード定数	'JEFコード列'NC	シフトコードを含むJEF漢字コード列からシフトコードを取り除く。(NDPから入力する場合に用いる)	'①年度②'NC	'年度'

注) 1で指定できる文字列の長さは28文字以内で、あらかじめ辞書に登録されていること。

(3) MODE属性

MODE属性は日本語データを日本語出力装置(日本語ラインプリンタ装置・日本語ディスプレイ装置)に出力する場合、EBCDIC(1バイト)とJEF漢字コード(2バイト)及び文字の大きさを区別するためのシフトコードを付加する機能である。

しかし、実際にシフトコードを付加するのは、後述の出力ファイルのENVIRONMENT属性のJEFオプションの指定による。

記述形式:

$\left\{ \begin{array}{l} \text{MODE(1)} \\ \text{MODE(2)} \end{array} \right\}$
 ……12ポイント(5文字/1インチ)の文字の大きさ K Sコード付加。
 ……9ポイント(8文字/1インチ) “ K 1コード付加。

省略時はMODE(1)がとられる。

例)

```
DCL 1 OUTREC,
      2 CTLC CHAR(1) INIT ('L'),…印刷前の紙送りの指示。1行改行
```

* KISコードとは、カナ2文字を漢字1文字に対応づけるコードのことである。このコードには、対応する漢字から容易に連想できるカナ文字が用いられており、たとえば、「イヤ」は「医」を表わし、「アホ」は「愚」を表わす。

- 2 A 1 NCHAR(10) MODE(1), 12ポで印刷
- 2 A 2 CHAR(15),
- 2 A 3 NCHAR(5) MODE(2), 9ポで印刷
- 2 A 4 NCHAR(3); 12ポで印刷

制御コード	KSコード	漢 字 コ ー ド	Aコード	EBCDIC	KIコード	漢 字 コー ド	KSコード	漢 コー ド
-------	-------	-----------	------	--------	-------	----------	-------	--------

```

DCL 1 OUTREC1 MODE(2),
      2 CTLC CHAR(1) INIT('1'), ..... 改頁
      2 A 1 NCHAR(10), ..... 9ポ
      2 A 2 CHAR(15),
      2 A 3 NCHAR(5) MODE(1), ..... 12ポ
      2 A 4 NCHAR(3); ..... 9ポ
    
```

注) MODE属性はNCHAR属性を持つ文字列のみ有効である。
 主構造体・従構造体で異なる指定をする時は下位属性が優先する。
 NDPへの出力では文字の大きさは一種類のみ。

(4) ENVIRONMENT属性のJEFオプション

NCHAR属性を含む構造体や変数を日本語出力装置に出力する場合はENV属性にJEFオプションの指定が必要である。JEFオプションはWRITE文でレコードを転送する時にMODE属性で指定されたシフトコードを付加する。

記述形式:

```

DCL NLPFILE FILE OUTPUT RECORD
      ENV(V, RECSIZE(307), CTLA, JEF, .....);
    
```

JEFオプションを持つファイルのレコード長は、NCHAR属性を持つ変数の前後にシフトコード(1バイト)を付加した長さであり、この場合、隣り合う変数が同一のMODE属性を持つ時は付加されない。

従って、RECSIZEには日本語出力装置で使える最大長(307バイト)を指定し、レコード形式はV(可変長)かU(不定長)形式とした方が良い。なお、シフトコード自身は印刷時には現れない。

```

例) DCL F1 FILE OUTPUT RECORD ENV(V, .....),
      F2 FILE OUTPUT RECORD ENV(V, JEF, .....);
      DCL 1 OUTREC,
            2 CTLC CHAR(1) INIT('1'),
            2 A 1 CHAR(20),
            2 A 2 MODE(2),
            3 B 1 NCHAR(5),
    
```

```

3 B2 NCHAR(4);
:
WRITE FILE ( F1 ) FROM(OUTREC);...シフトコードは付
加されない。
WRITE FILE ( F2 ) FROM(OUTREC);...シフトコード付加。
    
```

5.1.2 各種サブルーチンの利用

PL/Iのプログラムで日本語データの処理を行うとき、ADJUSTやKINGの汎用サブルーチンを利用できる。ADJUSTのサブルーチンは、JEF漢字コードに変換するための支援プログラム群であり、表5.2に示した機能を持っている。またKING(帳票出力)のそれは、プログラム処理の中で日本語データを出力する際の編集処理のための支援プログラム群であり、編集データの揃え(中央揃え、右詰め、均等揃え、氏名揃え)や編集データの分割または結合の機能を持つ。

なお、PL/Iの日本語処理機能およびADJUST、KINGの使用法についてはPL/Iの使用例としてあげた図5.4のマニュアルを参照のこと。

表5.2 ADJUSTサブルーチン

サブルーチンの種類	機能
日本語項目コード変換サブルーチン	日本語項目変換辞書を参照して、日本語項目コード(EB CDIC)からJEF漢字コードに変換する。
英数字カナコード変換サブルーチン	EBCDICを同じ文字を表わすJEF漢字コードに変換する。
カタカナひらがな変換サブルーチン	EBCDICのカタカナで指定された文字列をJEF漢字コードのひらがなに変換する。
KISコード変換サブルーチン	KISコードで指定された文字列をJEF漢字コードに変換する。
16進コード変換サブルーチン	16進数をJEF漢字コードに変換する。
日本語ファイルアクセスルーチン	日本語ファイルの内容をフィールド単位に読み取る。

5.1.3 使用法

(1) ジョブ制御文

PL/Iの日本語処理のためのカタログドプロシジャは従来のPLIXC, PLIXCLG等が利用でき、コンパイラオプションとして`JEF`を指定すればよい。なお、NLPへ各種リストを出力するには出力クラスに0の指定が必要となる。

```
例1) // EXEC P LIXC, PARM=JEF, SYSOUT=0 .....①
```

```
// P L I. SYSIN DD *
```

```
// ソースプログラム
```

①日本語データを含むプログラムの翻訳を示す。

例 2)

```

//A nnnn mm  ll  JOB ll Fnnnn, パスワード, MSGCLASS = 0 .....①
// EXEC PLIXCLG, PARM = ' JEF, JCONST(N) ', SYSOUT = 0 .....②
// PLI.SYSIN DD DSN = データセット名, DISP = SHR .....③
// GO. 出力ファイル名 DD SYSOUT = 0 .....④

```

- ① システムメッセージの出力を N L P に行う。
- ② 日本語データを含むプログラムを翻訳する。プログラム中に使用している定数を漢字コードに変換後、その一覧表も出力する。
- ③ ソースプログラムの入っているデータセット名を示す。
- ④ 実行時の結果を N L P に出力することを示す。

(2) 使用例

・入力プログラム

```

00000010 PRO_1:  PROCEDURE OPTIONS ( MAIN );
00000020   DCL NFILE FILE  RECORD ENV(V RECSIZE(307) CTLA JEF CONSECUTIVE);
00000030   DCL 1 OUT_A,
00000040     2 CLTA   CHAR(4),
00000050     2 A1     NCHAR(4) INIT( 'PL/I'NA ),
00000060     2 A2     NCHAR(5) INIT( 'ニチホクコウフリカ'NK ),
00000070     2 A3     NCHAR(4) INIT( 'ノタメ'NH ),
00000080     2 A4     NCHAR(2) INIT( 'サッコウ'NN ),
00000090     2 A5     NCHAR(2) INIT( 'フッコウ'NN ),
00000100     2 A6     NCHAR(7) INIT( 'A4F2BED2B2FOA4B7A4DEA4B9A1A5'NX ),
00000110   1 OUT_B,
00000120     2 CTLB   CHAR(1)  INIT( '0' ),
00000130     2 B1     CHAR(6)  INIT( '      (' ),
00000140     2 NUM     PIC'9',
00000150     2 B2     CHAR(2)  INIT( ' ) ' ),
00000160     2 MANUAL NCHAR(20) MODE(2),
00000161     2 CODE   CHAR(12);
00000170   DCL DATA1 NCHAR(6)  INIT( 'PL/I'NA ),
00000180     DATA2(2) NCHAR(5) INIT( 'ファミオカク'NK, 'シトヨケオチヒクカク'NK ),
00000190     DATA3(4) CHAR(12) INIT( '64SP-3064-1', '64SP-3074-1',
00000191                               '64SP-6620-1', '70AR-8201-1' );
00000192   DC' I FIXED BIN(15);
00000193
00000200 START:
00000210   OPEN FILE(NFILE) OUTPUT ;
00000220   NUM = 1;
00000230   WRITE FILE ( NFILE ) FROM ( OUT_A );
00000240   DO I = 1, 2, 2;
00000250     IF NUM = 3 THEN DATA1 = 'ADJUST'NA;
00000260     MANUAL = DATA1 || DATA2(I);
00000261     CODE   = DATA3(NUM);
00000270     WRITE FILE (NFILE) FROM ( OUT_B );
00000280     NUM = NUM + 1;
00000290   END;
00000300   MANUAL = 'K I N G ( メモヘヒヤリキ ) / J E F トケセホカク'NK;
00000301   CODE   = DATA3(NUM);
00000310   WRITE FILE ( NFILE ) FROM ( OUT_B );
00000320 END PRO_1;

```

図 5.2 PL/I の使用例のソースプログラム

・ JCONST(N) 指定による出力

** NCHARACTER CONSTANTS TABLE **

LINE	CONSTANT IN SOURCE	日 本 語 表 示
5	'PL/I'NA	'PL/I'
6	'ニチホンクコウフリカ'NK	'日本語処理'
7	'ノタメノ'NH	'のための'
8	'サンコウ'NN	'参考'
9	'フンケン'NN	'文献'
10	'A4F2BED2B2FOA4B7A4DEA4B9A1A5'NX	'を紹介します.'
18	'PL/I'NA	'PL/I'
19	'フミホオカク'NK	'文法書'
19	'シトヨクオチヒクカク'NK	'使用手引書'
29	'ADJUST'NA	'ADJUST'
35	'K I N G (メモヘヒヤリキ) / J E F トケ*	'KING (帳票出力) / JEF解説書'
	*セホカク'NK	

** NO SUBPROCESSOR DIAGNOSTIC MESSAGES **

図 5.3 漢字コード変換一覧表

・ 結 果

PL/I 日本語処理のための参考文献を紹介します。

- (1) PL/I 文法書 64SP-3064-1
- (2) PL/I 使用手引書 64SP-3074-1
- (3) ADJUST使用手引書 64SP-6620-1
- (4) KING (帳票出力) / JEF解説書 70AR-8201-1

図 5.4 図 5.2 のプログラムの実行結果

5.2 COBOLにおける日本語処理

COBOLの日本語処理はCOBOLプリプロセッサ(PRECOB)で行う。

利用者は日本語処理を行うステートメントを含むプログラムを拡張COBOL原始プログラムとしてPRECOBに投入し、通常のCOBOL命令に変換する。続いてCOBOLのコンパイルを行う。PRECOBが翻訳時に参照するファイルには日本語項目変換辞書、登録集(5.2.1参照)がある。

PRECOBのコンパイラオプションの機能とEXEC文のパラメータの指定方法を次に示す。

$$PARM = \left[\begin{array}{l} \text{SOURCE} \\ \text{NOSOURCE} \end{array} \right] \left\{ \begin{array}{l} \text{JLIST} \left(\left\{ \begin{array}{l} \text{CHAR} \\ \text{HEX} \end{array} \right\} \right) \\ \text{NOJLIST} \end{array} \right\}$$

↑
拡張原始プログラムの印刷を行うかどうかを指定する。

↑
日本語直定数をJEF漢字コードに変換後の一覧表を日本語又は16進数で印刷するかどうかを指定する。

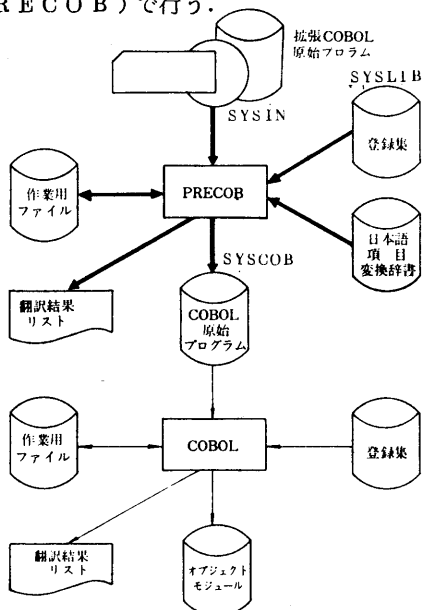
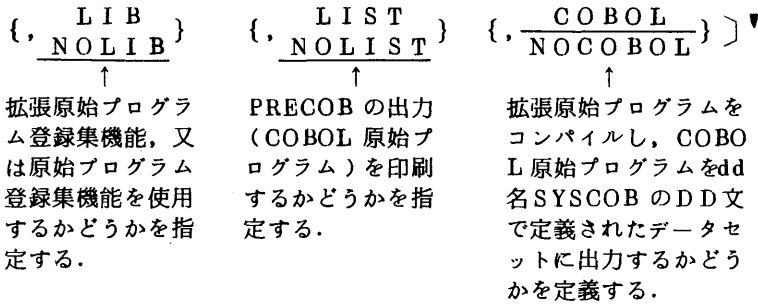


図 5.5 PRECOBの処理形態



5.2.1 日本語データ処理機能

PRECOBでは、日本語データを扱うために以下の機能をサポートする。

- ・日本語項目の定義。
- ・日本語直定数の定義。 表 5.3 参照
- ・日本語データを印刷する際の文字の大きさを示すシフトコードの付加。
- ・登録集の引用

次に各機能の詳細について述べる。

(1) 日本語項目の定義

拡張 COBOL 原始プログラム中のデータ記述項において日本語項目の定義を行う。

記述形式： 基本項目 PICTURE N (長さ)

長さは文字数を指定し、実際の記憶域は文字数の2倍のバイト数分を占有する。指定できる最大値は16,383。 PICTURE句の文字列は文字 'N' でのみ構成されていなければならない。

例)

```

0 1  OUT-REC.
      0 2  NO          PIC  X(3).
      0 2  NAME        PIC  N(8).  .....8文字(16バイト)の日本語文字列を
                                   示す。
      0 2  AGE         PIC  9(3).
      0 2  SECTION    PIC  NNN VALUE NN▽ケイサンキ▽.
    
```

.....日本語項目変換辞書より[▽] 計算機[▽]に変換される。

日本語項目は次の文脈で指定できる。

- ・比較命令 〔NOT〕=, 〔NOT〕EQUALのいずれか。
- ・MOVE命令 送り側と受け側は共に日本語項目であること。
- ・WRITE命令
- ・CALL命令, ENTRY命令及び手続き部の見出しのUSING句

(2) 日本語直定数

日本語直定数は表 5.3 に示すとおり 6 種類の指定方法があり、JEF 漢字コードに変換後の長さが 60 文字 (120 バイト) 以下でなければならない。

定数として SPACE, SPACES, ALL 日本語直定数の表意定数も指定できる。

表 5.3 日本語直定数一覧と記述例

No.	直定数の種類	記述形式	PRECOBの処理	記述例	翻訳結果
1	日本語項目直定数	NN'EBCDIC文字列'	指定されたEBCDICの文字列をキーに日本語項目変換辞書(事務処理用語辞書等)を検索して、JEF漢字コードに変換する。	NN 'テンシケイサンキ'	'電子計算機'
2	日本語英数字直定数	NA'EBCDIC文字列'	指定されたEBCDICの文字列を同じ文字を表わすJEF漢字コードに変換する。	NA 'ABC'	'ABC'
3	日本語連想直定数	NK'EBCDIC文字列'	KISコードで指定された文字列の2文字ずつをそれぞれ対応する漢字のJEF漢字コードに変換する。	NK 'ウルウエ'	'売上'
4	日本語ひらがな直定数	NH'カタカナ文字列'	EBCDICのカタカナで指定された文字列をJEF漢字コードのひらがなに変換する。	NH 'ヒラカ'ナ'	'ひらがな'
5	日本語16進数直定数	NX'16進数文字列'	指定された16進数4桁をJEF漢字コードとして内部コードに変換する。	NX 'B0A6'	'愛'
6	日本語コード直定数	NC'JEF漢字コード列'	シフトコードを含むJEF漢字コード列からシフトコードを取り除く。NDPから入力する場合に用いる)	NC 'Ⓢ漢字Ⓢ'	'漢字'

注) 1で指定できる文字列の長さは28文字以内で、あらかじめ辞書に登録されていること。
日本語直定数及び表意定数は次の文脈で指定できる。

- ・VALUE句
- ・比較命令
- ・MOVE命令の送り出し側

(3) シフトコードの付加

日本語データを日本語ラインプリンタ装置、日本語ディスプレイ装置に出力する場合、EBCDIC(1バイト)とJEF漢字コード(2バイト)及び文字の大きさを区別するためにシフトコードを付加する必要がある。利用者はデータ記述項でCHARACTER TYPE句を使用してシフトコードの有無と種類を指定する。

記述形式: CHARACTER TYPE IS { $\frac{\text{MODE-1}}{\text{MODE-2}}$ } ... 12ポ(5文字/1インチ)の文字の大きさ
... 9ポ(8文字/1インチ) //

CHARACTER TYPE句は日本語項目又はそれを含む集団項目にのみ指定でき、集団項目に書くとその中のすべての日本語項目に適用される。同時に異った指定をしてはならない。日本語データを含むレコードを日本語出力装置に出力するにはWRITE命令による。なお、シフトコードの付加はWRITE命令の直前で行われる。

```
例1)  FD LP-FILE ....
      01 LP-REC.
        02 LP-DATA PIC X(307).
      01 OUT-REC MODE-1.
        02 FILLER PIC X(4) VALUE SPACE.
        02 NO      PIC X(3).
        02 NAME   PIC N(8).
          |
      WRITE LP-REC FROM OUT-REC AFTER 2.
```

例 2)

```

01 OUT-REC
  02 FILLER PIC X(4) VALUE SPACE.
  02 NO      PIC X(3).
  02 NAME    PIC N(8) MODE-1.
  02 AGE     PIC 9(3).
  02 SECTION PIC N(3) VALUE NN▼ケイサンキ▼ MODE-2.
  ⋮
WRITE OUT-REC BEFORE 2.

```

例 3)

例 1 で下記のようにするとシフトコードは付加されない。

```

MOVE OUT-REC TO LP-DATA. ←送り側のシフトコードは受け側に
                           MOVEされない。
WRITE LP-REC AFTER 2.

```

(4) 登録集の引用

拡張原始プログラムによる登録集の引用には次の2通りがある。

- ・COBOLプログラムを作成する上で、共通に用いられるファイル記述、レコード記述や手続きなどを利用者の作成する登録集に登録しておき、INCLUDE命令によって拡張原始プログラムに組み込むことができる。(COPY命令はCOBOLコンパイル時のみ有効)
- ・拡張原始プログラム全体を登録集に登録しておき、翻訳時に呼び出して変更・追加・削除等の処理をしてPRECOBの入力プログラムとすることができる。(BASISカード、INSERTカード、DELETEカード)

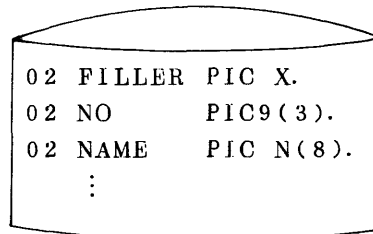
例 1)

```

01 OUT-REC.
  INCLUDE REC01.
01 MSG01.
  ⋮

```

登録集 メンバ名: REC01



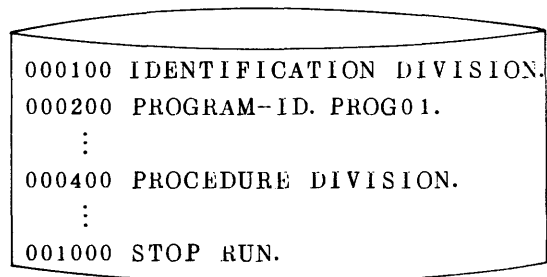
例 2)

```

//SYSIN DD *
  BASIS PROG01.
  INSERT 000500
00510 MOVE 0 TO COUNT.
  DELETE 000600
00610 ADD 1 TO COUNT.
  ⋮

```

登録集 メンバ名: PROG01



5.2.2 各種サブルーチンの利用

COBOLのプログラムで日本語データの処理を行うとき、5.1.2で述べたADJUSTや、KINGのサブルーチンが利用できる。

なお、COBOLの日本語処理機能及びADJUST、KINGの使用法については、COBOLの使用例としてあげた図5.8のマニュアルを参照のこと。

5.2.3 使用法

(1) ジョブ制御文

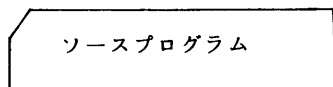
COBOLの日本語処理のためのカタログドプロシジャは入力プログラムの翻訳のみを行うCOBNと、翻訳・結合・実行を行うCOBNCLGが利用できる。なお、システムメッセージや実行結果をNLPへ出力させるには出力クラスにOの指定が必要となる。

例1)

```

//Annnmm □□ JOB□Fnnnn, パスワード, MSGCLASS=O .....①
// EXEC COBN .....②
//NCOB.SYSIN DD *

```



//

- ① システムメッセージの出力をNLPへ出力する。
- ② 日本語データを含むプログラムを翻訳し、各種リストをNLPに出力する。

例2)

```

// EXEC COBNCLG, PARM = NOJLIST .....①
//NCOB.SYSIN DD DSN=データセット名, DISP=SHR .....②
//GO. 出力ファイル名 DD SYSOUT=O .....③

```

- ① 日本語データを含むプログラムの翻訳・結合・実行処理を行う。なお、その際直定数の漢字コードへの変換一覧表は出力しない。
- ② 拡張原始プログラム(ソースプログラム)の入っているデータセット名を示す。
- ③ 実行結果をNLPに出力する。

(2) 使用例

・入力プログラム

```

000010 IDENTIFICATION DIVISION.
000020 PROGRAM-ID.        PROG2.
000030 ENVIRONMENT  DIVISION.
000040 CONFIGURATION SECTION.
000050 SOURCE-COMPUTER. FACOM-M200.
000060 OBJECT-COMPUTER. FACOM-M200.
000070 INPUT-OUTPUT SECTION.
000080 FILE-CONTROL.
000090         SELECT NFILE ASSIGN TO UR-S-SYSOUT.
000100 DATA DIVISION.
000110 FILE SECTION.
000120 FD          NFILE LABEL RECORD IS OMITTED
000130         DATA RECORD LP-REC.
000140 O1          LP-REC PIC X(307).
000150 WORKING-STORAGE SECTION.
000160 O1          OUTA MODE-1.
000170         O2 FILLER PIC X(4) VALUE SPACE.
000180         O2 A1  PIC N(5) VALUE NA'COBOL'.
000190         O2 A2  PIC N(5) VALUE NK'ニホンコクコクフリカ'.
000200         O2 A3  PIC N(4) VALUE NH'ノタメノ'.
000210         O2 A4  PIC N(2) VALUE NN'ランコウ'.
000220         O2 A5  PIC N(2) VALUE NX'A4F2BED2B2FOA4B7A4DEA4B9A1A5'.
000230         O2 A6  PIC N(7) VALUE NX'A4F2BED2B2FOA4B7A4DEA4B9A1A5'.
000240 O1          OUTB.
000250         O2 FILLER PIC X(5) VALUE SPACE.
000260         O2 B1  PIC X(1) VALUE ' '.
000270         O2 B2  PIC 9 VALUE 0.
000280         O2 B2  PIC X(2) VALUE ' ) '.

```

```

000290         O2 MANUAL PIC N(20) MODE-2.
000300         O2 MCODE PIC X(12).
000310 O1          WK-NAME.
000320         O2 DATA1.
000330         O3 FILLER PIC X(12) VALUE '64SP-3650-1 '.
000340         O3 FILLER PIC X(12) VALUE '64SP-6620-1 '.
000350         O3 FILLER PIC X(12) VALUE '70AR-8201-1 '.
000360         O2 CDATA REDEFINES DATA1.
000370         O3 NAME OCCURS 3.
000380         O4 FILLER PIC X(12).
000390
000400 PROCEDURE DIVISION.
000410 OPEN OUTPUT NFILE.
000420 WRITE LP-REC FROM OUTA AFTER PAGE.
000430 HAJIME.
000440 COMPUTE NUM = NUM + 1.
000450 MOVE NAME (NUM) TO MCODE.
000460 IF NUM = 1 THEN
000470     MOVE NK'C O B O L フリ フロ キ ツサ シトヨクオチヒクカク' TO MANUAL
000480     ELSE IF NUM = 2 THEN
000481     MOVE NK'A D J U S T シトヨクオチヒクカク' TO MANUAL
000490     ELSE IF NUM = 3 THEN
000500     MOVE NK'K I N G ( メモヒヤリキ ) / J E F トケセキカク' TO MANUAL.
000510 WRITE LP-REC FROM OUTB AFTER 2 LINES.
000520 IF NUM = 1 OR 2 GO TO HAJIME.
000530 DWARI.
000540 CLOSE NFILE.
000550 STOP RUN.

```

図 5.6 COBOL の使用例のソースプログラム

・ JLIST (CHAR) 指定による出力

*** 日本語直定数一覧表 ***

ISN	VALUE	ISN	VALUE	ISN	VALUE	ISN	VALUE
18	NA'COBOL' 'COBOL'	19	NK'ニホンコクコクフリカ' '日本語処理'	20	NH'ノタメノ' 'のための'	21	NN'ランコウ' '参考'
22	NN'ファンケン' '文献'	23	NX'A4F2BED2B2FOA4B7A4DEA4B9A1A5' 'を紹介します.'	47	NK'C O B O L フリ フロ 'COBOLプリプロセッサ使用手 引書'	49	NK'A D J U S T シトヨクオチヒクカク' 'ADJUST使用手引書'
	キ ツサ シトヨクオチヒクカク' 引書'			51	NK'K I N G (メモヒヤリ 'KING (帳票出力) / JEF解 説書'		
	キ) / J E F トケセキカク' 説書'						

図 5.7 漢字コード変換一覧表

・結果

COBOL日本語処理のための参考文献を紹介します。

- (1) COBOLプリプロセッサ使用手引書 64SP-3650-1
- (2) ADJUST使用手引書 64SP-6620-1
- (3) KING (帳票出力) / JEF解説書 70AR-8201-1

図 5.8 図 5.6 のプログラムの実行結果

6. KING (グラフ・図形出力) のPSPCNVの使用法

図形出力変換PSPCNVは、OS M/F4 PSPを使用して作成した図形をNLPに出力するプログラムである。出力に際して、図形の拡大、縮小、回転などを行えるほか、線の種類を変えたり、見出しを付けたりする機能を備えている。図形出力変換の概略図を、図6.1に示す。

以下、図形の出力を制御するための図形制御カード、PSPCNVを使うためのジョブ制御文について述べるとともに、いくつかの使用例も示すことにする。

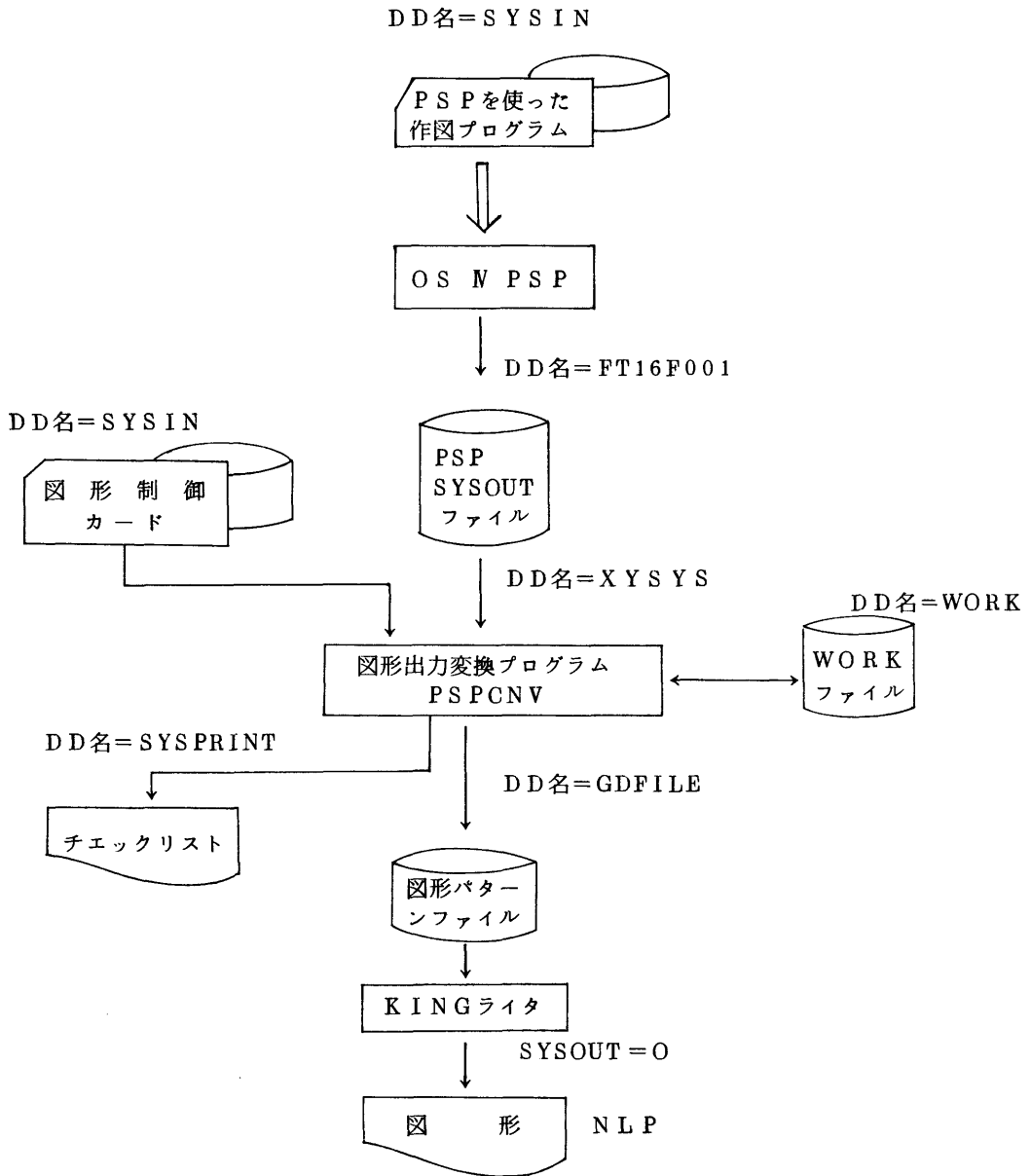


図 6.1 図形出力変換の概略図

6.1 図形制御カード

図形制御カードは、出力する図形の位置、大きさなどを与えるXYカード、図形に付加する見出しや注釈などを指示するSTカード、図形制御カードの終わりを示すENDカードの3種類から構成される。なお、図形制御カードの構文規則については付録1を参照していただきたい。

(1) XYカード

出力する図形の位置、大きさ、線の種類、回転方向などを指示する。

制御欄	オペレーション欄	オペランド欄
*	XY	$\left[\text{POS} = (\left[\text{行数} \right], \left[\text{文字数} \right]) \right] \left[, M = \text{倍率} \right] \left[, L = \text{線種} \right]$ $\left[, H \right] \left[, C = (\text{始点} X_s , \text{始点} Y_s , \text{終点} X_e , \text{終点} Y_e) \right] \left[, D = \left\{ \begin{matrix} O \\ R \\ L \\ I \end{matrix} \right\} \right]$ $\left[, \text{NOLMT} \right] \left[, \text{CODE} = (\left[\text{始コード} \right], \left[\text{終コード} \right], \left\{ \begin{matrix} \text{YES} \\ \text{NO} \end{matrix} \right\}) \right]$

・ $\left[\text{POS} = (\left[\text{行数} \right], \left[\text{文字数} \right]) \right]$

図形を出力するための画面の位置、大きさを示す。

$$0 \leq \text{行数} < 84$$

$$0 \leq \text{文字数} < 109$$

X方向の画面の大きさ = $84 - \text{行数}$

Y方向の画面の大きさ = $109 - \text{文字数}$

図6.2に行数、文字数、画面の関係を示す。

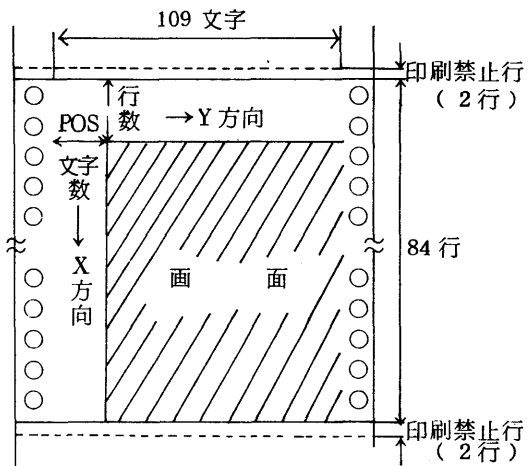


図6.2 行数、文字数と画面の関係

・ $\left[, M = \text{倍率} \right]$

図形の大きさを指定する。省略時は、1倍でXYプロッタの単位(ステップ)と画面の単位(ドット)について、1ステップ(0.1mm) = 1ドット(1/240インチ、約0.106mm)として出力する。なお、数値は、実際の倍率の10倍を整数で指定しなければならない。

指定範囲は、以下のとおりとする。

$$5 \leq \text{倍率} \leq 30 \text{ (1きざみ) (0.5倍} \sim \text{3.0倍に相当)}$$

・ $\left[, L = \text{線種} \right]$

図形を描くときに用いる線の種類と太さを2桁の数字で示す。

10の位は線の種類(実線, 破線)を表わし、1の位は線の太さ(0.1mm ~ 0.5mm)を示す。

数値の意味は、以下のとおりである。

線の種類(10の位)

	1	2
線種	実線	破線

線の太さ(1の位)

	1	2	3	4	5
太さ	0.1 mm	0.2 mm	0.3 mm	0.4 mm	0.5 mm

省略時は、 $L = 11$ (0.1mmの実線)である。

・ [, H]

画面の位置、大きさを明示するために、画面の4隅に「,」,「,」,「,」,「,」のマークを表示する。

省略時は、表示を行わない。

・ [, C = (始点 X s , 始点 Y s , 終点 X e , 終点 Y e)]

PSPを用いて作成した図形を部分的に切り出して出力する。切り出す領域は、XYプロッタの出力結果に対し終点、終点を指定する。指定する数値の単位はmmとする。

省略時は、始点(0, 0)から全データを

出力対象とする。

始点、終点、切り出し領域の関係を図6.3

3に示す。

・ [, D = { $\begin{matrix} O \\ R \\ L \\ I \end{matrix}$ }]

画面を90度きざみで回転する。パラメータの意味は以下のとおりである。

O	R	L	I
正(回転しない)	右倒し	左倒し	逆(180度回転)

省略時は、D=Oで回転しない。なお、Dオペランドは、NOLMTオペランドを省略したとき、有効となる。

・ [, NOLMT]

図形の出力範囲を示す。省略時は、日本語ラインプリンタ用紙1ページを超えた図形を出力しない。NOLMT指定時は、X軸、Y軸方向、それぞれ最大10ページとし、印刷禁止行を除いて、全図形を出力する。

・ [, CODE = ([始コード] , [終コード] ,

{ { YES } }))

PSPCNVでは、作画する図形を文字のパターンに分解する。そのために使用する文字コードの範囲をユーザ定義領域内に始コードと終コードで指定する。省略時は、始コード=80A1、終コード=A0FEである。

始コード、終コードと文字コードの範囲の関係を

図6.4に示す。

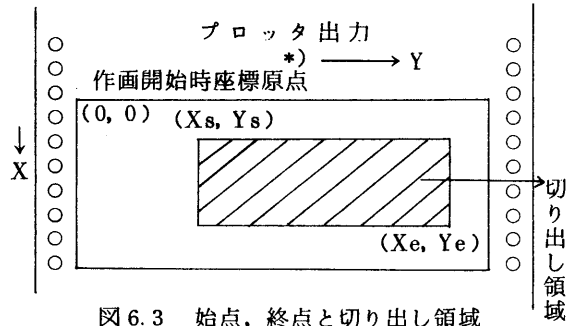


図6.3 始点、終点と切り出し領域

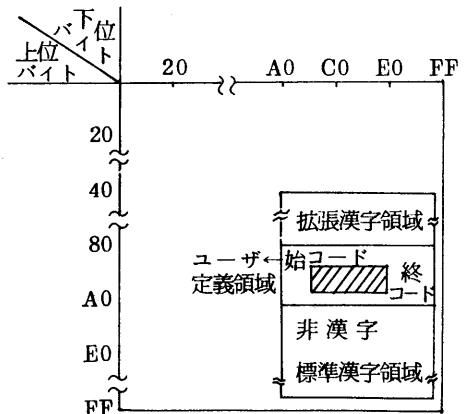


図6.4 始コード、終コードと文字コードの範囲

* 作画開始時座標原点とは、初期化サブルーチンPLOTSで設定した原点のことである。

YESを指定すると、コードが足りない場合に、拡張漢字領域内のコードも図形用の文字コードとして使用することを示す。NOの場合は、拡張しない。一般に複雑な図形を出力する場合、YESを指定しておいた方がよい。

(2) STカード

図形に付加する見出し、注釈などを指定する。

制御欄	オペレーション欄	オペランド欄
*	ST	$\left[\text{POS} = (\text{〔行数〕}, \text{〔文字数〕}, \left\{ \left\{ \frac{H}{V} \right\} \right\}) \right]$ <p>, A = (割付文字数, 割付行数)</p> $\left\{ \begin{array}{l} \text{, CA} = \text{▽ EBCDIC 群 ▽} \\ \text{, CX} = (\text{16進漢字コード}, \dots) \end{array} \right\}$

・ $\left[\text{POS} = (\text{〔行数〕}, \text{〔文字数〕}, \left\{ \left\{ \frac{H}{V} \right\} \right\}) \right]$

割り付ける文字の最初の位置を画面原点からの相対位置で表わす。

$0 \leq \text{行数} < X \text{方向の画面の大きさ}$

$0 \leq \text{文字数} < Y \text{方向の画面の大きさ}$

H指定の場合は、画面内で文字をY方向に割り付け、V指定の場合は、X方向に割り付ける。省略時は $\text{POS} = (0, 0, H)$ である。

これらの関係を図6.5に示す。

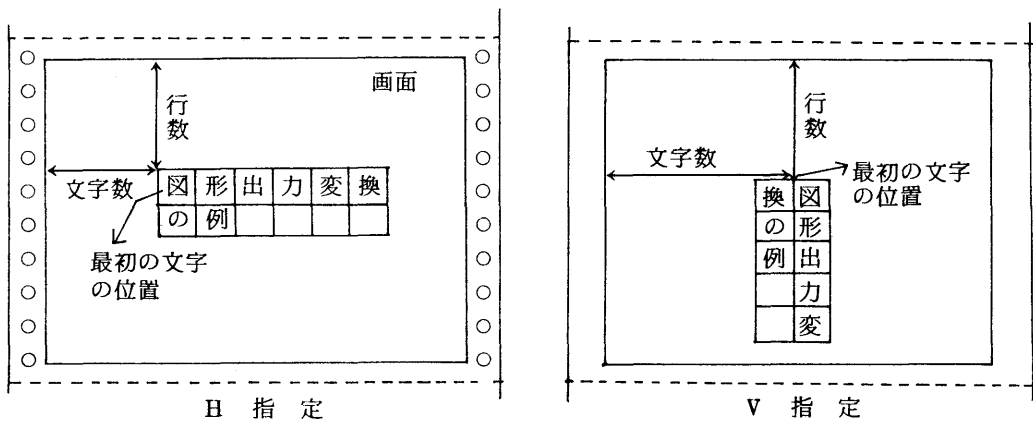


図6.5 文字の割付位置とその方向

・ A = (割付文字数, 割付行数)

割り付ける領域の大きさを指定する。

$1 \leq \text{割付文字数} \leq 50$

$1 \leq \text{割付行数} \leq 10$


```

//XYSYS          DD DSN= &&PSP, DISP=(OLD,DELETE)
//PSPCNV. SYSIN DD      *
* XY   POS=(10,10), H, CODE=(, ,YES), M=5 }
* ST   POS=(40,20), A=(10,2), CA='TEST' } 図形制御カード
* END
//

```

(ii) PSPのSYSOUTファイルが既に存在する場合

```

//A123402      JOB F1234,パスワード,REGION=1024K,MSGCLASS=O
//          EXEC  PSPCNV
//XYSYS        DD DSN=F1234.PSP, DISP=SHR ← PSPのSYSOUT
//SYSIN        DD DSN=F1234.PSPCNT.DATA, DISP=SHR ← 図形制御
//                                                    カード

```

6.3 作 図 例

以下に、いくつかの作図例とそれを出力するために用いた図形制御カードを示す。

例1)

$f(x, y) = \sin x \times \cos y$ ($-\pi \leq x \leq \pi$, $-\pi \leq y \leq \pi$) の等高線を作図したものである。出力に際して、図の大きさや位置はXYプロッタに出力した場合と同じになるように指定した。(実際には、倍率の指定のところの説明したように、約1.06倍になっている..)

0.1 mmの実線で描き、画面の四隅に「 \square 」マークを出力するようにしている。また、図形出力のための文字コードが不足すれば、拡張領域の文字コードを使うようにしている。

```

* XY POS=(0,0),M=10,L=11,D=0, CODE=(, ,YES),H
* END

```

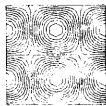


図 6.6 作図例 1

例 2)

例 1 で用いた図形を、大きさを 0.8 倍して、左に 90 度回転し、0.3 mm の破線で出力した例である。なお、必要な部分だけ図形を切り出しているため、画面と図形との相対位置が変化していることに注意していただきたい。また、ST カードを用いて P S P C N V T E S T P R O G R A M という見出しをつけている。

```
* XY POS=(0,0),M=8,L=23,D=L,CODE=(,YES),H,C=(40,20,100,100)
* ST POS=(50,0,H),A=(10,2),CA='PSPCNV TESTPROGRAM'
* END
```

PSPCNV TEST



図 6.7 作図例 2

例 3)

例 1 で示した図形を 1.5 倍して、0.1 mm の実線を用いて出力した例である。見出しは、J E F 漢字コードを用いて日本語で出力するようにしている。

```
* XY POS=(10,5),M=15,L=11,D=0,CODE=(,YES),H
* ST POS=(60,20),A=(10,2),CX=(BFDE,B7C1,BDD0,CECF,CAD1,B4B9,
* A4CE,CEE3,C2EA)
* END
```

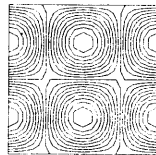


図 6.8 作図例 3

7 おわりに

以上、富士通提供の日本語情報システムJEFについて、現在使えるソフトウェアについて解説を行ってきた。各所で述べたように、現在、本センターで準備しているコマンドや、検討事項については、センターニュース等で広報していくつもりなので、注意していただきたい。また、利用者の方々のご意見、ご批判については、センターで処理できるものは処理し、そうでないものは、メーカーへの要望などの形でフィードバックしていくつもりである。

最後に、保存テキストの内容出力のためのコマンドを作成された本センターの景川耕字氏、およびNLP、KINGライター割当てのためのコマンドを作成された本センターの末永正氏に感謝する。また、広報教育委員会の委員の方達に、有益なコメントをいただき、原稿を改めることができたので、ここに感謝します。

参 考 文 献

1. 富岡 進, 西内 仁 日本語オペレーティングシステム, 電子通信学会誌, **63**, 7, 1980, 734-741.
2. 森 健一, 河田 勉 かな漢字変換, 情報処理, **20**, 10, 1979, 911-916.
3. 田中武美, 吉田 将 かな漢字変換による日本語入力システムに関する調査, 九大工学集報, **53**, 4, 1980, 365-370.
4. 長尾 真 日本語情報処理の諸問題, 電子通信学会誌, **63**, 7, 1980, 702-706.

本稿で述べたJEFのソフトウェアを利用するために読むべきマニュアルを、以下に列挙する。

5. 富士通マニュアル FACOM OSN JEF 解説書, 64SG-9000
6. 富士通マニュアル FACOM OSN FDMS (和文エディタ) / JEF 解説書, 64AR-8211
7. 富士通マニュアル FACOM OSN KING (文章処理) / JEF 解説書, 70AR-8202
8. 富士通マニュアル FACOM OSN KING (グラフ・図形出力) / JEF 解説書, 70AR-8203
9. 富士通マニュアル FACOM OSN PL/I 文法書 V05用, 64SP-3064
10. 富士通マニュアル FACOM OSN/F4 PL/I 使用手引書 V05用, 64SP-3074
11. 富士通マニュアル FACOM OSN/F4 COBOL プリプロセッサ手引書 V02用, 64SP-3650

下位 字	C0	C1	C2	C3	C4	C5	C6	C7	C8	C9	CA	CB	CC	CDE	CF	D0	D1	D2	D3	D4	D5	D6	D7	D8	D9	DA	DB	DC	DD	DE	DF
0	切	似	仟	价	伉	伏	估	佛	佢	佗	佞	佻	修	侏	侏	侏	侏	侏	侏	侏	侏	侏	侏	侏	侏	侏	侏	侏	侏	侏	
1	兩	愈	兮	冀	門	回	册	再	向	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	侗	
2	半	卮	廿	冀	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
3	唱	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
4	媽	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
5	壺	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
6	壺	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
7	壺	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
8	壺	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
9	壺	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
ABC	壺	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
BCD	壺	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	
DEF	壺	壺	咄	卉	卮	準	下	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	卮	

付表3 FDMS (和文エディタ) のコマンド入力形式一覧

名 称	機 能 概 要	入 力 形 式	EDIT モード	FORMAT モード	CALL モード	MACRO モード
NEDIT コマンド	和文エディタを起動する	{NEDIT NE } テキストデータセット名[/パスワード] [NEW OLD] [FORMAT [WIDTH(行文字数)] [DEPTH(ページ行数)] [TOP(ヘディング行数)] [BOTTOM(フッティング 行数)] [SHIFT(シフト文字数)] EDIT [DICTION(辞書データセット名[/パ スワード])])] [MACLIB(マクロライブラリ名[/パ スワード])])]				
END コマンド	和文エディタを終了する	END [SAVE NOSAVE]	○	○		
CALL コマンド	CALL モードに切り換え, 他の保 存テキストの内容を表示する	CALL [DA(テキストデータセット名)] LIST RELEASE]	○	○	○	○
EDIT コマンド	EDIT モードに切り換え, ソース 形式で表示する	{EDIT E } [表示参照番号]	○	○	○	○
FORMAT コマンド	FORMAT モードに切り換え, 清書 形式で表示する	{FORMAT FO } [表示参照番号][WIDTH(行文字数)] [DEPTH(ページ行数)] [TOP(ヘディング行数)] [BOTTOM(フッティング行数)] [SHIFT(シフト文字数)]	○	○	○	○
MACRO コマンド	MACRO モードに切り換え, テキス トマクロを表示する	{MACRO MCR } [マクロ名]	○	○	○	○

SAVE コマンド	テキストを保存する	{SAVE S} [テキストデータセット名]	○	○		
CHANGE コマンド	文字列を一括して変更する	{CHANGE C} [参照番号1 [参照番号2]] '被変更文字列' ['変更文字列'] [ALL]	○	○		
INPUT コマンド	テキストを追加する	{INPUT I} [参照番号] $\left[\frac{A}{B} \right]$	○	○		
MERGE コマンド	他のテキストの内容をユニット単位に組み込む	{MERGE M} テキストデータセット名 [参照番号1 [参照番号2]] [組み込み位置参照番号] $\left[\frac{A}{B} \right]$	○	○		
'C(copy) コマンド	文字列を複写する	{ 'CS- $\left[\frac{A}{B} \right]$ 'C- 'C- $\left[\frac{A}{B} \right]$ }	○	○		○
'D(delete) コマンド	文字列を削除する	{ 'DS 'D - 'D }	○	○		○
'G(merge) コマンド	他のテキストの内容を文字単位に組み込む	{ 'GS- $\left[\frac{A}{B} \right]$ 'G- 'G- $\left[\frac{A}{B} \right]$ }	○	○	○	
'I(insert) コマンド	テキストを挿入する	{ 'IA 'IB }	○	○		○
'M(move) コマンド	文字列を移動する	{ 'MS- $\left[\frac{A}{B} \right]$ 'M- 'M- $\left[\frac{A}{B} \right]$ }	○	○		○

名 称	機 能 概 要	入 力 形 式	EDIT モード	FORMAT モード	CALL モード	MACRO モード
↑R(replace) コマンド	文字列を置き換える	{↑RS ↑R-↑R }	○	○		○
BOTTOM コマンド	画面をテキストの最後に移動する	{BOTTOM B }	○	○	○	○
DOWN コマンド	画面を下方へ移動する	{DOWN D } [整数]	○	○	○	○
FIND コマンド	文字列を検索し表示する	{FIND F } [開始参照番号] [↓ 検索文字列 ↓]	○	○	○	
LOCATE コマンド	画面を指定の位置へ移動する	{LOCATE LOC } 表示参照番号	○	○	○	
TOP コマンド	画面をテキストの先頭へ移動する	TOP	○	○	○	○
UP コマンド	画面を上方へ移動する	UP [整数]	○	○	○	○
PAGECONT コマンド	改頁を取り消す	{PAGECONT PC }		○		
↑E(erase) コマンド	空白を削除する	↑E		○		
↑H コマンド (homeposition)	この位置で改頁する	↑H		○		
↑L コマンド (linecont)	改行を取り消す	↑L		○		
↑N コマンド (newline)	この位置で改行する	↑N		○		
↑S コマンド (skip)	空白を確保する	↑S-↑S		○		

↑ U コマンド (up)	行末を字上げる	↑ U		○		
↑ W コマンド (down)	行頭を字下げする	↑ W		○		
↑ MA コマンド (midashi add)	⊠ (空白)・表に見出しを付加する	↑ MA ⊠ 表見出し文字列		○		
↑ MD コマンド (midashi delete)	⊠ (空白)・表の見出しを削除する	↑ MD		○		
↑ TA コマンド (table add)	表を作成する	↑ TA ($\begin{bmatrix} L \\ C \\ R \end{bmatrix}$ カラム文字数 $\begin{bmatrix} + \\ - \\ - \end{bmatrix}$. . .) レコード数 [(レコード番号 $\begin{bmatrix} + \\ - \\ - \\ / \end{bmatrix}$. . .)] [SHIFT (シフト文字数)] $\begin{bmatrix} KEI \\ NOKEI \\ YOKO \end{bmatrix}$ $\begin{bmatrix} HEADER \\ NOHEADER \end{bmatrix}$ $\begin{bmatrix} A \\ B \end{bmatrix}$	○	○		
↑ TD コマンド (table delete)	表を削除する	↑ TD		○		
↑ TM コマンド (table modify)	表の形式を変更する	↑ TM [SHIFT (シフト文字数)] $\begin{bmatrix} KEI \\ NOKEI \\ YOKO \end{bmatrix}$ $\begin{bmatrix} HEADER \\ NOHEADER \end{bmatrix}$		○		
↑ RA コマンド (record add)	表のレコードを追加する	↑ RA レコード数 [(レコード番号 $\begin{bmatrix} + \\ - \\ - \end{bmatrix}$. . .)] $\begin{bmatrix} A \\ B \end{bmatrix}$		○		
↑ RD コマンド (record delete)	表のレコードを削除する	↑ RD		○		
↑ RM コマンド (record modify)	表のレコードの形式を変更する	↑ RM $\begin{bmatrix} + \\ - \\ - \end{bmatrix}$		○		

名 称	機 能 概 要	入 力 形 式	EDIT モード	FORMAT モード	CALL モード	MACRO モード
↑ RX コマンド (record exchange)	表のレコードを差し換える	↑RX - ↑RX		○		
↑ CA コマンド (column add)	表のカラムを追加する	↑CA ($\begin{bmatrix} L \\ C \\ R \end{bmatrix}$ カラム文字数 $\begin{bmatrix} + \\ - \end{bmatrix}$. . .) $\begin{bmatrix} A \\ B \end{bmatrix}$		○		
↑ CD コマンド (column delete)	表のカラムを削除する	↑CD		○		
↑ CM コマンド (column modify)	表のカラムの形式を変更する	↑CM $\begin{bmatrix} L \\ C \\ R \end{bmatrix}$ [カラム文字数] [+]		○		
↑ CX コマンド (column exchange)	表のカラムを差し換える	↑CX - ↑CX		○		
DICTION コマンド	カナ漢字変換辞書を切り換える	{ DICTION } [DA (辞書データセット名)] { DI } [LIST]	○	○	○	○
SHOW コマンド	単語・漢字のメニューを表示する	{ SHOW } [見出し [. $\begin{Bmatrix} T \\ C \\ J \\ G \\ P \\ R \\ Q \end{Bmatrix}$]]	○	○	○	○
↑ X コマンド (exchange)	文字列を再変換する	{ ↑XS } { ↑X - ↑X }	○	○		○
↑ T コマンド (tango update)	辞書に単語を登録する	$\begin{Bmatrix} C \\ G \\ J \\ D-11 \\ D-K5 \\ D-KH \\ D-G5 \end{Bmatrix}$	○	○		○

		{ 'TS 'T- 'T } 見出し [.	D-S5 D-SH D-ZH D-T5 D-N5 D-B5 D-M5 D-R5 D-W5 K M				
DEFMAC コマンド	テキストマクロを登録する	{ DEFMAC DEF } マクロ名 ['マクロ定義体']		○	○	○	○
MACLIB コマンド	マクロライブラリの割当て・解放を行う	{ MACLIB MLB } [DA(マクロライブラリ名) LIST RELEASE]		○	○	○	○
MACSAVE コマンド	テキストマクロを保存する	{ MACSAVE MSV } [マクロライブラリ名]		○	○	○	○
'Q コマンド (quote)	テキストマクロを登録する	{ 'QS 'Q- 'Q } マクロ名		○	○		○
DELMAC コマンド	テキストマクロを削除する	{ DELMAC DEL } マクロ名		○	○	○	○

付表 4. 制御コード入力形式一覧

名称	入 力 形 式	機 能 概 要	備 考
ページ形式	$\textcircled{P}G = \left[\left[\begin{matrix} \text{段文字数} \\ 50 \end{matrix} \right] \left[\begin{matrix} \text{ページ行数} \\ 66 \end{matrix} \right], \left[\left[\begin{matrix} \text{段数} \\ 1 \\ 2 \end{matrix} \right] \right], \left[\begin{matrix} \text{段数文字数} \\ 5 \end{matrix} \right], \left[\left[\begin{matrix} \text{基本文字サイズ} \\ 9 \\ 12 \end{matrix} \right] \right], \left[\begin{matrix} \text{シフト文字数} \\ 4 \end{matrix} \right] \textcircled{A}$	ページの形式を指定する	和文エディタでは無効
領域定義	$\textcircled{A}R = \left[\left[\begin{matrix} \text{ヘディング領域行数} \\ 5 \end{matrix} \right], \left[\begin{matrix} \text{フッティング領域行数} \\ 5 \end{matrix} \right] \right] \textcircled{A}$	ヘディング領域、フッティング領域の行数を指定する	同上
柱定義	$\textcircled{H}S = \left[\left[\begin{matrix} \text{柱種別} \\ \left\{ \begin{matrix} H \\ F \end{matrix} \right\} \end{matrix} \right] \left[\begin{matrix} \text{柱割付けページ指定} \\ \begin{matrix} 12 \\ 13 \\ 23 \\ 24 \\ 35 \end{matrix} \end{matrix} \right] \right], \left[\begin{matrix} \text{柱行数} \\ 3 \end{matrix} \right] \textcircled{A} \left[\dots \left[\dots \right] \textcircled{T}2 \textcircled{A} \left[\dots \right] \textcircled{T}2 \textcircled{A} \right. \\ \left. \left[\dots \right] \textcircled{N}O \textcircled{A} \right] \textcircled{H}E \textcircled{A}$	ページ番号や柱の出力位置などを定義する	
T A B 制御	$\textcircled{T}B = \left[\left[\begin{matrix} \text{S字下げ文字数} \\ 50 \end{matrix} \right], \left[\begin{matrix} \text{表枠指定} \\ \left\{ \begin{matrix} K \\ Y \end{matrix} \right\} \end{matrix} \right] \right], \left[\begin{matrix} \text{F} \\ \left\{ \begin{matrix} \text{カラム揃え} \\ \left\{ \begin{matrix} L \\ C \\ R \end{matrix} \right\} \end{matrix} \right\} \right] \left[\begin{matrix} \text{カラム文字数} \\ 1 \end{matrix} \right] \left[\begin{matrix} \text{カラム間文字} \\ + \end{matrix} \right] \sim \left[\begin{matrix} \text{L, H} \end{matrix} \right] \textcircled{A} \\ \left[\begin{matrix} \text{表見出しテキスト} \\ \textcircled{M}S \textcircled{A} \dots \dots \dots \textcircled{M}E \textcircled{A} \left[\dots \left[\textcircled{C}E \textcircled{A} \right] \dots \left[\textcircled{R}E \left[\left\{ \begin{matrix} \pm \\ \mp \end{matrix} \right\} \right] \textcircled{A} \right] \sim \textcircled{T}E \textcircled{A} \right. \end{matrix} \right]$	文章を表の形式にして出力する	
見出し制御	$\left\{ \begin{matrix} \textcircled{M}1 \\ \textcircled{M}2 \\ \textcircled{M}3 \\ \textcircled{M}4 \\ \textcircled{M}5 \end{matrix} \right\} = \left[\left[\begin{matrix} \text{見出し下げ文字数} \\ 1 \end{matrix} \right], \left[\begin{matrix} \text{見出し空白行数1} \\ 4 \sim 2 \end{matrix} \right], \left[\begin{matrix} \text{見出し文字サイズ} \\ \begin{matrix} 9 \\ 12 \end{matrix} \end{matrix} \right], \left[\begin{matrix} \text{見出し空白行数2} \\ 0 \end{matrix} \right] \right] \textcircled{A} \dots \textcircled{N}L \textcircled{A}$	章、節、項などの見出しの出力形式を指定する	
パラグラフ制御	$\textcircled{P}R = \left[\left[\begin{matrix} \text{1行目字下げ文字数} \\ 1 \end{matrix} \right], \left[\begin{matrix} \text{2行目字下げ文字数} \\ 0 \end{matrix} \right] \right] \textcircled{A}$	段落の形式を指定する	
揃え制御	$\textcircled{F}M = \left[\left[\begin{matrix} L \\ C \\ R \end{matrix} \right] \right] \textcircled{A}$	行を左詰め(L)又は中央揃え(C)又は右詰め(R)にする	
両端揃え	$\dots \textcircled{T}2 \textcircled{A} \left[\begin{matrix} \text{リーダー} \\ K2 \end{matrix} \right] \textcircled{A} \left[\dots \right] \textcircled{N}L \textcircled{A}$	行内で両端揃えにする	
内部空白	$\textcircled{S}K = \left[\begin{matrix} \text{D} \end{matrix} \right] \textcircled{A} \left[\begin{matrix} \text{空白行数} \\ 0 \end{matrix} \right], \left[\begin{matrix} \text{空白文字数} \\ \left\{ \begin{matrix} L \\ R \end{matrix} \right\} \end{matrix} \right] \textcircled{A} \left[\begin{matrix} \text{図版名} \\ \textcircled{M}S \textcircled{A} \dots \dots \dots \textcircled{M}E \textcircled{A} \right] \textcircled{A}$	この位置に空白を確保する	
段頭段末空白	$\textcircled{S}K = \left[\begin{matrix} \text{D} \end{matrix} \right] \textcircled{A} \left[\begin{matrix} \text{空白行数} \\ \left\{ \begin{matrix} T \\ B \\ A \end{matrix} \right\} \end{matrix} \right], \left[\begin{matrix} \text{空白文字数} \\ \left\{ \begin{matrix} L \\ R \end{matrix} \right\} \end{matrix} \right] \textcircled{A} \left[\begin{matrix} \text{図版名} \\ \textcircled{M}S \textcircled{A} \dots \dots \dots \textcircled{M}E \textcircled{A} \right] \textcircled{A}$	ページ・段の上又は下に空白を確保する	

行間隔	@ LN [=スキップ行数] ₂] @	行間隔を変更する	
文字サイズ	@ SZ [= { 文字サイズ }] @ 9 12	文字サイズを変更する	和文エディタでは無効
改行	@ NL @	この位置で改行する	
改段	@ NB @	” 改段する	
改頁	@ NP @	” 改頁する	
字下げ	@ DN [=字下げ文字数] ₀] @	行頭を字下げする	
字上げ	@ UP [=字上げ文字数] ₀] @	行末を字上げする	
アンダライン	@ UL @ @ UE @	アンダラインを引く	和文エディタでは無効
分割禁止	@ BK @ @ BE @	文字列を行分割禁止とする	
ピッチ詰め	@ PT @ @ PE @	英数字記号をピッチ詰めする	EBCDICの文字で出力される
罫線	@ KS [= [罫線文字数] ₁ [, 罫線No.] ₁] @	罫線を引く	和文エディタでは無効
日付	@ DT [= { 日付種類 }] @ Y M D Y M Y M D M D D	日付を出力する	同上
索引	@ IX @ [@ K2 @読み方] @ IE @	索引データを抽出する	和文エディタでは無効
目次	@ MK [= { 1 } 2 3 }] @ @ NL @	目次データを抽出する	同上
印刷制御	@ NA @ @ NE @	この部分を出力しない	
ページ番号スキップ	@ PN=S [ページ数] ₁] @	ページ番号をスキップする	
ページ番号リセット	@ PN=R [ページ番号] ₁] @	ページ番号を再設定する	
オーバーレイ制御	@ OV=オーバーレイパターン名 @	オーバーレイパターンを指定する	和文エディタでは無効

付表5. テキスト入力における変換指定一覧

入 力 種 別	入 力 形 式	入 力 文 字 種	結 果	
カナ漢字変換入力	カタカナ, ローマ字, 数字, 特殊文字	漢字かな混り文	
	無変換指定	/...../	英字, 数字, 特殊文字	そのまま漢字コードへ
	ひらがな指定	H/...../	カタカナ, ローマ字, 一部の特殊文字	ひらがな
	カタカナ指定	K/...../	カタカナ, ローマ字, 一部の特殊文字	カタカナ
	単語指定	T/見出し[.n]/	カタカナ, ローマ字	指定の単語
	地名指定	C/見出し[.n]/	カタカナ, ローマ字	指定の地名
	人名指定	J/見出し[.n]/	カタカナ, ローマ字	指定の人名
	外来語指定	G/見出し[.n]/	英字, 数字, カタカナ, 特殊文字	指定の外来語(単語)
	音指定	P/音読み[.n]/	カタカナ, ローマ字	指定の漢字
	訓指定	R/訓読み[.n]/	カタカナ, ローマ字	指定の漢字
制御コード入力	@制御コード文字列@	英字, 数字, 一部の特殊文字	EBCDICのまま	
漢字直接入力	漢字コード	そのまま	
テキストマクロ引用入力	Q/マクロ名[(実パラメタ)]/	英字, 数字, 実パラメタはテキスト	マクロ定義体	

付表6. カタカナ・ローマ字対応表

母音 行	ア	イ	ウ	エ	オ	母音 行	ア	イ	ウ	エ	オ
ア行	A	I	U	E	O	チャ行	TYA	TYI	TYU	TYE	TYO
カ行	KA	KI	KU	KE	KO		CYA	CYI チイ	CYU	CYE チェ	CYO CHO
サ行	SA	SI SHI	SU	SE	SO		CHA	チイ	CHU	チェ	CHO
タ行	TA	TI CHI	TU TSU	TE	TO	ニャ行	NYA	NYI ニイ	NYU	NYE ニエ	NYO
ナ行	NA	NI	NU	NE	NO	ヒャ行	HYA	HYI ヒイ	HYU	HYE ヒエ	HYO
ハ行	HA	HI	HU FU	HE	HO	ミャ行	MYA	MYI ミイ	MYU	MYE ミエ	MYO
マ行	MA	MI	MU	ME	MO	リャ行	RYA	RYI LYI リイ	RYU LYU	RYE LYE リエ	RYO LYO
ヤ行	YA	YI イ	YU	YE イエ	YO	ファ行	FA	FI	FE	FO	
ラ行	RA	RI	RU	RE	RO	クワ行	QA	QI	QU	QE	QO
	LA	LI	LU	LE	LO		KWA	KWI クイ	KWU ク	KWE クエ	KWO クオ
ワ行	WA	WI ウイ	WU ウ	WE ウエ	WO ヲ	ギャ行	GYA	GYI ギイ	GYU	GYE ギエ	GYO
ン	N					ジャ行	ZYA	ZYI	ZYU	ZYE	ZYO
ガ行	GA	GI	GU	GE	GO	ヂャ行	JA	JU	JE	JO	
ザ行	ZA	ZI JI	ZU	ZE	ZO		JYA	JYI ジイ	JYU	JYE ジエ	JYO
ダ行	DA	DI ヂ	DU ヅ	DE	DO		DYA	DYI ヂイ	DYU	DYE ヂエ	DYO ヂョ
バ行	BA	BI	BU	BE	BO	ビャ行	BYA	BYI ビイ	BYU	BYE ビエ	BYO
パ行	PA	PI	PU	PE	PO	ピャ行	PYA	PYI ピイ	PYU	PYE ピエ	PYO
キャ行	KYA	KYI キイ	KYU	KYE キエ	KYO	ヴァ行	VA	VI ヴィ	VU	VE ヴィエ	VO ヴィオ
シャ行	SYA	SYI シイ	SYU	SYE シエ	SYO	その他	CA	CI	CU	CE	CO
	SHA	SHU	SHE シェ	SHO			カ	シ	キュ	セ	コ
							FYA	FYI ヒイ	FYU	FYE ヒエ	FYO ヒョ
							GWA	GWI グイ	GWU グ	GWE グエ	GWO グオ
							QYA	QYI クイ	QYU ク	QYE クエ	QYO クオ
							DHI デイ	THI テイ			

付録1. 図形制御カードの一般規約

	オペレーション欄	オペランド欄	識別欄	
	2		72	73 80
*	⌊識別文字⌋	パラメタ列	継続指定	

- ・ “ * ”と識別文字，識別文字とパラメタ列の間には，一つ以上の空白を置く。
- ・ オペランドは，各項目ごとに，“，”（コンマ）で区切らなければならない。
- ・ 継続するカードは，必ずコンマで終わっており，72カラムが空白以外でなければならない。
- ・ []：省略可能である。
- ・ { }：いずれか一つを選択する。
- ・ { }内の下線は，省略時のサブオペランドを表す。
- ・ 識別文字直後のオペランドのコンマは省略する。
- ・ サブオペランド一つの場合，括弧は省略できる。
- ・ (サブオペランド，……)は，サブオペランドの連続を意味する。
- ・ ([]，……)は，[]の連続を意味する。

付録2. 図形制御カードのエラーコード

	エラーコード	意味
X Y カ ー ド	X001	XYカードが無い
	X002	XYカードが2枚以上続けて指定された。
	X003	未定義のキー・ワードが現れた。
	X004	POSの誤り
	X005	M "
	X006	L "
	X007	D "
	X008	P "
	X009	NOLMT "
	X010	C "
	X011	CODE "
	X012	H "
	X013	NOLMTの指定時にDオペランドに“0”以外を指定した。
S T カ ー ド	S002	STカードが50枚をこえた
	S003	未定義のキー・ワードが現れた。
	S004	POSの誤り
	S005	A "
	S006	CA "
	S007	CX "
	S008	CN "
	S009	CA, CX又はCNオペランドが指定されていない
	A	A001
A002		{ 未定義の識別文字が現れた。 継続指定中に終了