

## TSS PFDの使用について

南, 俊朗  
九州大学理学部数学教室 : プログラム相談員

武富, 敬  
九州大学大型計算機センター研究開発部

遠矢, 真知子  
九州大学大型計算機センター研究開発部

景川, 耕宇  
九州大学大型計算機センター研究開発部

<https://doi.org/10.15017/1474926>

---

出版情報 : 九州大学大型計算機センター広報. 15 (1), pp.21-80, 1982-03-10. 九州大学大型計算機センター

バージョン :

権利関係 :



## TSS PFDの使用について

南 俊朗\*, 武富 敬\*\*, 遠矢 真知子\*\*, 景川 耕宇\*\*

## 目 次

1. はじめに .....	23
2. PFD概説 .....	23
2.1 PFDの特徴 .....	23
2.2 PFDの機能概要 .....	24
3. PFD-EDITを主とした使用例 .....	25
3.1 PFDの開始 .....	25
3.2 EDITの開始 .....	26
3.3 EDITメンバメニュー .....	27
3.4 EDITモードの設定 .....	29
3.5 画面移動 .....	29
3.6 文字列の検索/置換 .....	30
3.7 文字単位の編集 .....	31
3.8 行単位の編集 .....	32
3.9 行の分割/重ね合わせ .....	33
3.10 画面分割 .....	34
3.11 HELP機能 .....	37
3.12 EDITの終了 .....	40
3.13 PFDの終了 .....	41
4. ATTRIBUTES オプション .....	42
4.1 TERMINAL サブオプション .....	43
4.2 LOG/LIST サブオプション .....	44
4.3 PF KEYS サブオプション .....	45
5. EDITオプション .....	48
5.0 開始, 終了 .....	48
5.1 モード .....	49
5.1.0 PROFILE コマンド .....	49
5.1.1 NUMBER モード .....	50
5.1.2 AUTONUM モード .....	51

\* 九州大学理学部数学教室(プログラム相談員)

\*\* 九州大学大型計算機センター研究開発部

5.1.3	STATS モード	51
5.1.4	PRINT モード	51
5.1.5	CAPS モード	51
5.1.6	NULLS モード	52
5.1.7	TABS モード	52
5.1.8	HEX モード	53
5.1.9	RECOVERY モード	54
5.2	画面移動	54
5.2.1	SCROLL キーによる画面移動	54
5.2.2	コマンドによる画面移動	55
5.3	行機能	58
5.4	編集機能	59
5.4.1	行の編集	59
5.4.2	行単位の編集	60
5.4.3	行の重ね合わせ	63
5.5	内容の転送	64
5.6	行の表示	65
5.7	テキスト	66
5.8	その他	67
6.	BROWSE オプション	68
6.1	開始, 終了	68
6.2	画面移動	69
6.2.1	LOCATE コマンド	69
6.2.2	FIND コマンド	70
6.3	16 進数表示	70
6.4	カラム目盛りの表示	70
7.	UTILITY オプション	71
付録 1.	PFD のオプション一覧	74
付録 2.	PFD のコマンド一覧	76
付録 3.	PF キーの標準定義	79
付録 4.	ディスプレイ装置のキーの機能	80

## 1. はじめに

昨年6月に、RACF(Resource Access Control Facility)の導入によりデータセット保護が可能となったため、PFD(Programming Facility for Display user)を公開した\*[1, 8] PFDは、他のTSSの機能と比較しても、操作性のよい幾つかの優れた特徴を持つにもかかわらず、現在までそれほど使われていないようである。しかし、利用者からの要望もあり、また、他のシステムと比較した長所・短所の紹介の意味も含めて、今回ここで解説することにする。

PFDは、フルスクリーン機能を持つ端末を用いるプログラム開発支援システムである。入力はメニュー方式で行われるため、単にメニュー画面の必要な個所にパラメータを入力するだけでよく、エンドユーザ向きである。また、HELP機能も充実しており、必要な時点でHELP画面を表示することにより情報を得ることができ、マニュアルなしで仕事を進めることができる。PFDのEDIT機能にも、他には見られない幾つかの特徴がある。会話型汎用日本語エディタ右筆[2]や、プログラム開発支援システムGEM(Generalized program Editing and Management facilities)[3, 4]は、PFDの1つのオプションであるため、それらを開始するためには、まずPFDに入る必要がある。

センターには、これまで、フルスクリーン機能を活用することにより操作性の向上をはかったシステムとして、画面編集を可能にしたTSS EDITのFSサブコマンド(Full Screen Option, 以後、FSOと呼ぶ)[5]や、本センター末永氏開発のTSSコマンド入力支援ツール(BROWSE, LIST Mコマンド)[6]などがあり、その操作性の良さのために多くの利用者を引きつけてきた。これらとPFDを比較してみると、各々特徴を持っているため、利用者の判断や好みに応じて各々使い分けをすればいいように思われる。

なお、本稿では、PFDのすべての機能について詳述する訳ではないので、詳細はマニュアル[7]を参照されたい。特に、ここでは右筆、GEMについてはふれない。

2では、PFD概説として、PFDの特徴および機能の概要を述べる。3では、実際にPFD-EDITの使用例を通して、PFDの開始やFSOにない編集機能、モードの設定、画面分割、HELP機能、データセットの更新履歴の採取などPFD特有の機能およびPFDの終了について述べる。4では、AT TRIBUTESオプションについて述べる。5では、EDITオプションについて概説する。6と7では、それぞれBROWSEオプションとUTILITYオプションについて概説する。

## 2. PFD概説

### 2.1 PFDの特徴

PFDの特徴を以下に列挙する。

- 1) PFDのEDIT機能は、FSOに比べ編集機能が拡張されている部分がある。それは、i) テキストの1行を2行に分割したり、行同志の重ね合せ(overlay)ができる、ii) 行挿入などの編集後、行番号が自動的に付け変えられる、iii) FIND, CHANGEコマンドの機能が、標準的に

---

\* PFDにおいては、他人の区分データセットへの書込みが可能であったため、RACFが導入されるまで公開を保留していた。

ファンクションキー (PF5, PF6) にも割り当てられているので、文字修正などがワンタッチで行える、等である。また、区分データセットのメンバの更新記録をメンバ単位に保存できる。この時、行番号部分を利用して、行ごとの更新回数を残すことも可能である。これらの機能は、データセットの保守管理などに有益であろう。

- 2) SPLITキー (PF2) によって、画面を上下の2つに分割できる。それぞれの画面は独立に PFDの機能を実行するので、ある作業を中断して別の作業を行うとか、2つのデータセットの内容を同時に表示させて比較するとかが可能である。
- 3) 入力、すべてメニュー形式で行われるので、画面を見ながら必要な個所に、データセットの指定、種々の属性の割り当てなどに必要なパラメータを入力すればよい。
- 4) HELPキー (PF1) を用いると、任意の時点で現在使用中の機能に関するHELP画面を表示できる。また、簡単なコマンドによりHELP画面を切り換えながら、必要な情報を得ることができる。
- 5) ファンクションキーを、利用者ごとに任意のコマンドに対応させることができ、操作性を向上させることができる。
- 6) しかし、現在のPFDのEDITは、TSS EDITのRUNサブコマンドに対応する機能を持たないので、編集中的数据を直ちに実行させてみるということとはできない。ただし、SUBMITサブコマンドの機能は持っている。

## 2.2 PFDの機能概要

PFDは、現在次の10種類の機能を備えている。これらには、0から9までの番号(オプション番号と呼ぶ)が対応づけられており、PFDでは、この番号を指定することにより機能を選択することになる。

### 1) オプション0 (ATTRIBUTES)

PFDの利用者ごとの属性を定義するもので、端末特性の定義、PFDの出力するログ/リストデータセットの処置に関するパラメータの設定、ファンクションキー(PFキー)の定義を行う。PFキーには、PFDのEDIT, BROWSEのコマンド、EDITの行コマンドを対応させることができる。

### 2) オプション1 (BROWSE)

データセットの内容を画面に表示する。画面の移動は、PFキーによる他、コマンドにより、相対行番号、文字列ピクチャ、行のラベルの指定によっても行える。データセット内容の16進表示もできる。

### 3) オプション2 (EDIT)

データセットの画面編集機能を持つ。メンバの更新履歴の採否、リカバリ機能の採否、画面の表示状態などに関する種々のモードを選ぶことができ、それによりメンバの更新履歴を保存したり、リカバリ機能を動かせたりすることができる。また、FSOよりもきめ細かい編集機能を持っている。

### 4) オプション3 (UTILITY)

データセットの割り当て、削除、属性の表示、複写、圧縮、内容のプリントなどの各種ユーティリティの機能を持つ。

#### 5) オプション4 (FOREGROUND)

コンパイル、リンクエディット、インタラクティブデバッグなどを行うTSSコマンドを、メニュー形式の入力を与えて起動する。

#### 6) オプション5 (BACKGROUND)

コンパイル、リンクエディットを行うFIBジョブ起動を、メニュー形式の入力により可能とする。

#### 7) オプション6 (TSS)

通常のTSSコマンド、コマンドプロシジャの呼び出しを行う。

#### 8) オプション7 (HELP)

PFDの使用法を表示する。HELPの画面は階層構造になっており、PFキーやコマンドで画面を移動させながら情報を得ることができる。INDEXコマンドを使うと、HELP画面の目次を表示できる。HELPの終了には、ENDキー(PF3)を用いる。

#### 9) オプション8 (YUUHITSU)

日本語処理のためのエディタであり、EDITの編集機能に日本語処理の機能を付加したものと考えてよい。右筆では、作業行を用いてその上でカナ漢字変換を行うとか、ある行からある行までを一括してカナ漢字変換を行うなどの働きをもつコマンドがある。

#### 10) オプション9 (GEM)

プログラムの開発・保守を支援することを目的とするGEMの中で、PFD-GEMは、画面編集機能をもつエディタとして動作し、モジュールの登録および更新を行う。

### 3. PFD-EDITを主とした使用例

例を示しながら、PFD-EDITの基本的な使い方を説明する。PFD-EDITによる編集作業はFSOの場合とよく似ているため、FSOを使った経験のあるユーザはPFD-EDITも楽に使いこなせるであろう。

#### 3.1 PFDの開始

PFDを開始させるためには、READYのあとにPFDと入力すればよい。

READY

PFD

PFDは10種類のオプションの集まりであるため、PFDが表示する最初の画面は、オプション選択のためのメニュー画面(プライマリオプションメニュー)である。カーソルは画面上の2行目のOPTION===>と表示された入力フィールドにある。ここにオプション番号を入力して、PFDの持っているどの機能を使うのかを選択する。EDITを選んだ例が図3.1である。

```

-----< PFD PRIMARY OPTION MENU >-----
OPTION ==> 2
_
USERID - F0052
-----
TIME - 15:50
-----
0 ATTRIBUTES - DEFINE PFD AND TERMINAL ATTRIBUTES   TERMINAL - F9526
1 BROWSE      - READ SOURCE DATA OR OUTPUT LISTINGS  PF KEYS  - 24
2 EDIT        - CREATE OR CHANGE SOURCE DATA        KANA FEATURE - YES
3 UTILITY     - PERFORM PFD UTILITY FUNCTIONS
4 FOREGROUND  - COMPILE, ASSEMBLE, LINK EDIT, OR DEBUG IN FOREGROUND
5 BACKGROUND  - SUBMIT JOB TO COMPILE, ASSEMBLE, OR LINK EDIT
6 TSS         - EXECUTE TSS COMMAND OR CLIST UNDER PFD
7 HELP        - GET INFORMATION ABOUT PFD
8 YUUHITSU*   - CREATE OR UPDATE NIHONGO DATA
9 GEM         - CREATE OR UPDATE GEM MODULE
X EXIT        - TERMINATE PFD USING LOG/LIST DEFAULTS
-----
PRESS END KEY TO TERMINATE PFD THROUGH PFD TERMINATION MENU

```

図 3.1 プライマリオプションメニュー（EDITオプション選択）

### 3.2 EDITの開始

EDITオプションの最初の画面はEDITデータセットメニュー（図3.2）である。この画面は編集するデータセットを指定するためのものであり、2通りの指定方法がある。すなわちPFDライブラリデータセットとして指定する方法および一般のデータセットとして指定する方法である。前者はPROJECT, LIBRARY, TYPE, MEMBERの4つの入力フィールドを用いて指定する。MEMBERフィールドを空白とした場合は▼プロジェクト名、ライブラリ名、タイプ名▼<sup>\*\*</sup>というデータセットを指定したものとみなされ、そうでない場合は▼プロジェクト名、ライブラリ名、タイプ名（メンバ名）▼というメンバを指定したものとみなされる。図3.2の場合は▼F0052. ALIB. ASM▼を指定している。後者は、DATASET NAMEの入力フィールドにTSSで普通に用いる書き方でデータセット名を入力する。たとえば▼F0052. ALIB. ASM▼もしくはALIB. ASMと入力して▼F0052. ALIB. ASM▼というデータセットを指定する。両者を同時に指定した時は後者の指定が優先される。

なお、PFDライブラリデータセットのライブラリ連結機能についてはPFDのマニュアル[7]を参照されたい。

\* 8番目のオプション右筆は、F6650型ディスプレイ端末（日本語ディスプレイ）を使用する時だけ、このメニュー上に表示され、それ以外には表示されない。  
 \*\* プロジェクト名は課題名、ライブラリ名はユーザ指定名、タイプ名は内容識別修飾子に対応する。

```

-----< EDIT - DATASET MENU >-----
SPECIFY DATASET TO BE EDITED:

PF D LIBRARY DATASET:
  PROJECT ==> F0052
  LIBRARY ==> ALIB      ==>          ==>          ==>
  TYPE    ==> ASM
  MEMBER  ==>          (BLANK TO DISPLAY MEMBER LIST)
  _

OTHER PARTITIONED OR SEQUENTIAL DATASET:
  DATASET NAME ==>
  VOLUME SERIAL ==>          (IF NOT IN CATALOG)

DATASET PASSWORD ==>          (IF PASSWORD PROTECTED)

PROFILE NAME    ==>          (IF BLANK, ASSUMES DATASET TYPE)

```

図3.2 EDITデータセットメニュー(▼F0052.ALIB.ASM▼指定)

### 3.3 EDITメンバメニュー

EDITデータセットメニューの指定が順データセットもしくは区分データセットのメンバ指定の場合は、直ちに編集画面が表示されるが、区分データセットでメンバを指定しない場合は、EDITメンバメニュー(図3.3)が表示される。この画面は編集するメンバを指定するためのものであり、画面移動とメンバ選択の2つの機能を持つ。

図3.3の場合と異なり、メンバ数が多いときは画面移動を行い目的のメンバ名を表示させる。移動はUP(PF7)/DOWN(PF8)キーもしくはLOCATEコマンドによる。

$\left. \begin{array}{l} \text{LOCATE} \\ \text{LOC} \\ \text{L} \end{array} \right\}$	メンバ名
--	------

指定されたメンバが存在しないときは、直前のメンバが指定されたものとみなされる。たとえばNで始まるメンバ名をみつけないときはLOC Nと入力後、もし必要ならばDOWNキー(PF8)を用いて順に探していくことができる。

メンバ選択にはメンバ名の前の入力フィールドにSと入力する方法とSELECTコマンドを用いる方法がある。図3.3は前者の方法でNXTCODEというメンバを選択している例である。

```

EDIT - MEMBER LIST - FO052.ALIB.ASM -----
COMMAND ==>                                     SCROLL ==> PAGE
NAME          VER.MOD  CREATED   LAST MODIFIED  SIZE  INIT  MOD   ID
FS10
GETC
S NXTCODE
-
OB
**END**
    
```

図 3.3 EDIT メンバメニュー (メンバNXTCODEを選択)

もう一方のSELECTコマンドは次の形式でメンバを選択する。

$\left\{ \begin{array}{l} \text{SELECT} \\ \text{SEL} \\ \text{S} \end{array} \right\}$	メンバ名
---	------

新しいメンバを作成する場合はSELECTコマンドによらざるを得ない。ただし、編集作業後に表示されるメンバメニューには新規作成のメンバは表示されない。そのため、新規作成メンバをも表示させるためには、ENDキー(PF3)を押して、一度データセットメニューまでもどり、その後メンバ名をつけないデータセット名のみを指定を用いて、再びメンバメニューを表示させると今回のメニューでは、新規メンバが加わって表示されることになる。

EDITデータセットメニュー(図3.2)またはEDITメンバメニュー(図3.3)の後、EDIT画面(図3.4)が表示される。

```

EDIT --- FO052.ALIB.ASM(NXTCODE) ----- COLUMNS 001 072
COMMAND ==> STATS                               SCROLL ==> HALF
***** ***** TOP OF DATA *****
000010 * SUBROUTINE  NXTCODE(CODE: CHAR(2))
000020 *   CODE ORDER : 80A1 ... 80FE 81A1 ... .. A0FE 41A1 ... 79FE 80A1 ...
000030 *
000040 *   MODIFY:   81.10.22  ADD CHECK OF 79FF
000050 *
000060 NXTCODE CSECT
000070 *----- PROLOGUE -----*
000080         USING      *,15
000090         B           START
000100         DC        XL1'07',CL7'NXTCODE'
000110 START      SVG      14,12,A(SAVE)
000120         LR        8,15
    
```

```

000130          DROP      15
000140          USING    NXTCODE,8
000150 *----- MAIN -----*
000160          L          2,0(GR1)      / GET CODE ADDRESS
000170          LH         3,0(2)        / GET CODE
000180          N          3,=XL4'0000FFFF' / MASK OF UNDER 2-BYTES
000190          A          3,=F'1'       / ADD 1 TO THE CURRENT CODE
000200          C          3,=XL4'0000A0FF' / IF GR3=X'A0FF' THEN GR3=X'41A1'
000210          BE        XA0FF          / ELSE IF

```

図 3.4 EDIT画面 (STATS コマンド入力)

### 3.4 EDITモードの設定

PFD-EDITは表示や処理の形式を定めるために9つのモードを持っている。モードのON/OFFに応じて表示や処理のしかたを変えるわけである。

STATSモードは、区分データセットに対してメンバ毎の更新履歴および行単位の更新レベルの記録をとるかどうかを定めるものである。順データセットに対しては常にOFFである。

NUMBERモードは編集中のデータが行番号付きであるかどうかを定める。標準の行番号形式は、データのレコード形式が固定長 (RECFM=F) ならばレコードの後8桁、可変長 (RECFM=V) ならばレコードの前8桁を、それぞれ行番号フィールドとする形式である。さらにSTATSモードがONの場合、この8桁のうち上位6桁を行番号、下位2桁を更新レベルのフィールドとして使用する。たとえば4回目のメンバ更新の際に修正または追加された行の更新レベルフィールドは04となる。

STATSモード、NUMBERモードをONにするには、それぞれSTATSコマンド、NUMBERコマンドをコマンド入力フィールドに入力するとよい。図3.4はSTATSコマンドを入力したところである。行番号を初期値100増分値100にするため次にRENUMコマンドを入力する。RENUMコマンドによってNUMBERモードがONになり、行番号がつけ直される。

なおPFD-EDITのモードは、あるデータセットに対して一度指定すると、データセットのタイプ名、レコード形式、レコード長の3つの値の組とに1つのプロフィール情報として保存される。そして、以後同じタイプ名、レコード形式、レコード長のデータセットの編集の際には、保存されていたモードが標準となるため、ユーザはEDITを行うたびにモードの初期設定を行うわずらわしさから解放されることになる。

### 3.5 画面移動

モード設定も一応終わり、これから実際の編集作業にはいる。編集データの目的の部分が画面に現われるように画面を移動させるためには通常UP (PF7)、DOWN (PF8)、LEFT (PF10)、RIGHT (PF11) の4種類のSCROLLキーを用いる。目的行を画面上の先頭行とするためにはLOCATEコマンドを用いる。形式は次のとおりである。

$\left. \begin{array}{c} \text{LOCATE} \\ \text{LOC} \\ \text{L} \end{array} \right\}$	行番号
--	-----

指定の行番号が見つからない場合は直前の行番号が先頭行となる。

### 3.6 文字列の検索／置換

ある文字列を探す FIND コマンド、また文字列を別の文字列で置き換える CHANGE コマンドは編集作業中よく使われる。

FIND コマンドは指定された文字列を探し、見つかった場合、その文字列が表示されていないならば表示されるように画面を移動し、その後見つかった文字列の先頭にカーソルを置く。FSO の FIND コマンドと比べると PFD-EDIT の場合、文字列の指定や探すための方向等を細かく指示できる点、および FIND キー (PF5) を押すと改めてコマンドを入力しないで、同じ文字列を探することができる点が便利である。次にいくつかの例を示す。

例：

文字列指定： F ABC ..... 現在のカーソル位置から後の文字列 ▼ABC ▼ を探す。

引用符つき文字列指定： F ▼A□B ▼ ..... 上の例と同様に文字列 ▼A□B ▼ を探す。

16進文字列： F X ▼FF ▼ ..... 上の例と同様に16進コードFFの文字を探す。

ALL 指定： F XYZ ALL ..... データセットの先頭から終了まで範囲内のすべての文字列 ▼XYZ ▼ を探し、みつかった個数を表示する。

CHANGE コマンドは指定された文字列を探し、それを別の文字列で置き換える。文字列を探す機能は FIND コマンドの機能であり、文字列の指定は FIND コマンドと同じ形で指定できる。次にいくつかの例を示す。

例：

C ABC XY ... 文字列 ▼ABC ▼ を探しそれを文字列 ▼XY ▼ に置き換える。

C ▼A□B ▼ ABB ALL -- 文字列 ▼A□B ▼ をすべて文字列 ▼ABB ▼ に置き換える。

C XXX \* PREV ..... 現在のカーソル位置から前の方へ文字列 ▼XXX ▼ を探し、直前の CHANGE コマンドで指定した文字列に置き換える。

CHANGE キー (PF6) を押すと改めて CHANGE コマンドを入力しないで CHANGE を行うことができる。また FIND キー (PF5) を押すと CHANGE コマンドの機能のうち文字列を探す働きだけが行われ、見つかった文字列の先頭にカーソルが移動する。これをうまく利用すると1つ1つの文字列を確認しながら文字列を置き換えることができる。CHANGE コマンドをコマンド入力フィールドへ書き込んだ後、ENTER キーを押さず、FIND キーを押す。FIND キーの働きで見つかった文字列の先頭にカーソルが移動する。もし文字列を置き換えたいならば CHANGE キーを押し、置き換えたくないならばもう一度 FIND キーを押して次の文字列を探す。これを繰り返して行うことができる。図 3.5 に実例を示す。

```

EDIT --- F0052.ALIB.ASM(NXTCODE) ----- COLUMNS 001 072
COMMAND ==> C '*' **                               SCROLL ==> HALF
000300 *
000310 * SUBROUTINE NXTCODE(CODE: CHAR(2))
000320 * CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ...
000400 * MODIFY: 81.10.22 ADD CHECK OF 79FF
000500 *
000600 NXTCODE CSECT

```

↓ FIND キー

```

EDIT --- F0052.ALIB.ASM(NXTCODE) ----- CHARS '*' FOUND
COMMAND ==>                                         SCROLL ==> HALF
000300 *
000310 * SUBROUTINE NXTCODE(CODE: CHAR(2))
000320 * CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ...
000400 * MODIFY: 81.10.22 ADD CHECK OF 79FF
000500 *
000600 NXTCODE CSECT

```

↓ CHANGE キー

```

EDIT --- F0052.ALIB.ASM(NXTCODE) ----- CHARS '*' CHANGED
COMMAND ==>                                         SCROLL ==> HALF
000300 **
000310 * SUBROUTINE NXTCODE(CODE: CHAR(2))
000320 * CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ...
000400 * MODIFY: 81.10.22 ADD CHECK OF 79FF
000500 *
000600 NXTCODE CSECT

```

図 3.5 FIND キー (PF5) と CHANGE キー (OF6) による CHANGE の実行

### 3.7 文字単位の編集

編集作業のうちの文字列の追加、削除、置換は普通ディスプレイ装置の持つ編集機能を利用して行われる。すなわち変更したい文字の位置へカーソルを移動させ文字キーを使って新たな文字を入力していく。カーソルの移動はカーソル移動キーもしくは Tab キー、Back Tab キー、HOME キー等のフィールド指向キーで行う。文字の置換はカーソルの位置へ新しい文字を入力して行う。文字の削除は DEL キーや ERASE EOF キーで行う。削除された文字のかわりとしてフィールドの右端には NULL 文字 (16 進コード X'00') が置かれる。文字の挿入は、まず INSERT MODE キーを押してディスプレイ装置を挿入モードにした後行われる。挿入モードの解除は RESET キーを押して行う。挿入モードの際に入力される文字はカーソルの位置に置かれ、この位置から右の文字

はNULL文字を除いて1文字分ずつ右へシフトされる。1文字挿入されると、NULL文字が1個削除されて全体の文字数が一定となるわけである。したがって画面上ではNULL文字と空白は区別がつかないものの文字の挿入の際はどちらであるかが重要となる。PFD-EDITは各フィールド毎の行末の空白を空白として画面に表示するかNULLとして表示するかをNULLSモードによって選択できるようにしている。NULLSモードにすると、それぞれの行末の空白はNULL文字として表示される。したがって行末の空白数に応じた個数の文字を挿入することができる。しかしNULLSモードの場合、画面での行末の空白部分の途中に入力された文字が、その左側のNULL文字のために実際の入力位置と異なる位置に移動してしまうことが起こる。それを避けるにはNULLSモードを解除するか、最後の非空白文字のすぐ右の位置からスペースキーによって空白を埋めながら目的の位置へカーソルを移動させるかのいずれかの方法によらなければならない。

### 3.8 行単位の編集

行の複写、移動、繰り返し、削除、挿入は行番号フィールドへ入力するコマンド——行コマンド——を用いる。これらに関してもFSOの場合と同じ要領で行うことができる。複写(Copy)、移動(Move)、繰り返し(Repeat)、削除>Delete)、挿入(Insert)はそれぞれC, M, R, D, Iの行コマンドを用いる。ただし、複写と移動の両行コマンドについては複写先、移動先を指示するA (After)あるいはB (Before)のいずれかの行コマンドと組み合わせられて用いる。

上記のコマンドはいくつかの拡張形式を持っている。たとえばC行コマンドは複写元の行を指定するのにC, Cn, CC-CCと3通りの方法が使える。それぞれ指定の1行、指定の行を含めたn行、2つのCC行を含めてその間のすべての行を複写元として指定している。また複写先についても、AまたはAnで指定行の直後にそのまま複写または同じものをn回繰り返して複写することを指定できる。R行コマンドは直後への複写と解釈できるが、複写先が直後に限られるため、An, Bn型の指定に対応する指定としてRRn-RR, RR-RRn, RRn-RRnのいずれかの形で全体の複写回数を指定できる。ただし、両方にnをつける場合は、それらは一致しなければならない。

次にそれぞれの行コマンドの形式の一覧を示す。

複写 : C, Cn, CC-CCをA, An, B, Bnと組み合わせる。  
 移動 : M, Mn, MM-MMをA, An, B, Bnと組み合わせる。  
 繰り返し : R, Rn, RR-RR, RRn-RR (もしくはRR-RRn, RRn-RRn)  
 削除 : D, Dn, DD-DD  
 挿入 : I, In

図3.6に使用例を示す。

(a) I行コマンド

```

I00100 * SUBROUTINE  NXTCODE(CODE: CHAR(2))
-
000200 *   CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ...
000300 *
    
```

⇓ ENTERキー

```

000100 * SUBROUTINE  NXTCODE(CODE: CHAR(2))
-
000200 *   CODE ORDER : 80A1 ... 80FE 81A1 ... .. A0FE 41A1 ... 79FE 80A1 ...
000300 *

```

## (b) C行コマンド

```

C20100 * SUBROUTINE  NXTCODE(CODE: CHAR(2))
000200 *   CODE ORDER : 80A1 ... 80FE 81A1 ... .. A0FE 41A1 ... 79FE 80A1 ...
A00300 *
-
000400 *   MODIFY:  81.10.22  ADD CHECK OF 79FF
          ↓↓ ENTERキー
000100 * SUBROUTINE  NXTCODE(CODE: CHAR(2))
000200 *   CODE ORDER : 80A1 ... 80FE 81A1 ... .. A0FE 41A1 ... 79FE 80A1 ...
000300 *
000310 * SUBROUTINE  NXTCODE(CODE: CHAR(2))
-
000320 *   CODE ORDER : 80A1 ... 80FE 81A1 ... .. A0FE 41A1 ... 79FE 80A1 ...
000400 *   MODIFY:  81.10.22  ADD CHECK OF 79FF

```

## (c) R行コマンド

```

R20300 *
-
000310 * SUBROUTINE  NXTCODE(CODE: CHAR(2))
          ↓↓ ENTERキー
000300 *
000301 *
-
000302 *
000310 * SUBROUTINE  NXTCODE(CODE: CHAR(2))

```

図 3.6 行コマンドの使用例

## 3.9 行の分割／重ね合わせ

データセットの編集の際に1つの行を2行に分けたいとか、あるいは逆に2行に分れているものを1行に重ね合わせたいとかの機能がはしくることがある。PFD-EDITではこれらはTS (Text Split) 行コマンドおよびO (Overlay) 行コマンドとM (Move) 行コマンドの組み合わせによって実現できる。

TS行コマンドは次のように用いられる。分割したい行にTS行コマンドを入力し、分割したい位置にカーソルを合わせた後ENTERキーを押すとその行には左側部分が残る、直後に空白の挿入行が表示され、さらにその次の行に分割された右側部分からなる行が挿入される。図3.7に例を示す。

行を重ね合わせる機能はO行コマンドで重ねられる側の行を指定し、C行コマンドあるいはM行コ

マンドで重ねる側の行を指定する。C行コマンドによる場合は重ね合わせの後も重ねる側の行は残るが、M行コマンドによる場合は削除される。1行と1行の重ね合わせは次のように行われる。重ねられる側の行の中の非空白文字部分には文字を重ね合わせることができない。空白部分に対してのみ重ねる側の行の対応するカラム位置にある文字で置き換える手続きがなされる。O行コマンドはO, On, OO—OOの3とおりの方法で重ねられる行を指定する。重ねる行、重ねられる行の対応は、先頭行から順に対応させていく。もし重ねる側の行数が足りない場合は行の指定を必要だけ繰り返し用いる。逆に重ねられる側の行数が足りない場合は、重ねられる側の行がすべて対応づけられた時点で対応づけを終了する。図3.7に行の重ね合わせの例を挙げる。

(a) 行の分割

```

TS0310 * SUBROUTINE  NXTCODE(CODE: CHAR(2))
-
000320 *   CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ...
000400 *   MODIFY:   81.10.22  ADD CHECK OF 79FF

      ↓↓ ENTERキー

000310 * SUBROUTINE
.....
000311 NXTCODE(CODE: CHAR(2))
-
000320 *   CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ...
000400 *   MODIFY:   81.10.22  ADD CHECK OF 79FF
    
```

(b) 行の重ね合せ

```

000310 * SUBROUTINE
-
CC0311 *   SUBR          NXTCODE(CODE: CHAR(2))
CC0320 *   CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ...
000400 *   MODIFY:   81.10.22  ADD CHECK OF 79FF

      ↓↓ ENTERキー

000310 * SUBROUTINE          NXTCODE(CODE: CHAR(2))
-
000311 *   SUBR          NXTCODE(CODE: CHAR(2))
000320 *   CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ...
000400 *   MODIFY:   81.10.22  ADD CHECK OF 79FF
    
```

図3.7 行の分割, 重ね合わせの例

3.10 画面分割

PF2の画面は指定した任意の行を境界として上下の2画面に分割することができる。画面分割の指示はSPLITキー(PF2)を用いる。SPLITキーが押されたときにカーソルのある行で画面分割が行われ、下の画面にプライマリオプションメニューが表示される。ここで下の画面の入力フィールドに新たなオプション番号を入力してPF2を利用できる。図3.8に画面分割の例を示す。

```

EDIT --- F0052.ALIB.ASM(NXTCODE) ----- COLUMNS 001 072
COMMAND ==>                                SCROLL ==> HALF
000310 * SUBROUTINE      NXTCODE(CODE: CHAR(2))
000311 *   SUBR          NXTCODE(CODE: CHAR(2))
000320 *   CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ...
000400 *   MODIFY: 81.10.22 ADD CHECK OF 79FF
000500 *
000600 NXTCODE CSECT
000700 *----- PROLOGUE -----*
000800         USING      *,15
000900         B           START
001000         DC        XL1'07',CL7'NXTCODE'
001100 START      SVG     14,12,A(SAVE)
001200         LR        8,15
001300         DROP      15
001400         USING     NXTCODE,8
. . . . .
-----< PFD PRIMARY OPTION MENU >-----
OPTION ==> 6
_
USERID      - F0052
----- TIME      - 16:00
0  ATTRIBUTES - DEFINE PFD AND TERMINAL ATTRIBUTES   TERMINAL - F9526
1  BROWSE     - READ SOURCE DATA OR OUTPUT LISTINGS PF KEYS  - 24
2  EDIT       - CREATE OR CHANGE SOURCE DATA        KANA FEATURE - YES

```

図 3.8 画面分割 (下の画面にTSS オプションのオプション番号6を入力したところ)

分割された画面のそれぞれを論理画面と呼ぶ。分割するための境界行の位置を変更し論理画面への行数の配分を変えるときも新しい境界の行にカーソルを合わせSPLITキーを押すとよい。

画面への入力 ENTER キー、ファンクションキーを押したときのカーソルのある側の論理画面に対してなされたものと解釈される。したがって2つの論理画面を交互に使うことは可能であるが、同時に両画面への入力を行うことは不可能である。論理画面を交互に使う場合、SWAP キー (PF9) を押すとカーソルは他方の論理画面の元のカーソル位置へと移動する。また一方の論理画面の行数が4行以下の場合、SWAP キーを押すと両方の論理画面の行数も入れ換わる。SCROLL フィールドのない画面に関してはSCROLL キーによる画面移動が行えないため画面の一部が表示されない場合が生ずる。このような場合にSWAP キーによる画面行数の入れ換え機能を用いると便利である。

なお、画面分割を終了させるためには一方の論理画面に対してENDキー (PF3) を続けて押していくと、そちらの論理画面のPFDはプライマリオプションメニューを経た後終了する。その時点で終了していない側の論理画面だけの単一画面になる。

TSS-EDITと比べてPFD-EDITの劣っている点にRUN,ALLOCATE,FREEなどのコマンドが使えないことがある。FSOを利用する場合、ソースプログラムを修正したのちRUNサブコマンドで実行させ、その結果よりさらに修正を加えるという使い方ができ非常に便利である。PFD-EDITのコマンド入力フィールドにはEDIT用のコマンド以外のコマンドやRUNコマンドは入力できない。FIBジョブを起動するSUBMITコマンドがあるだけである。そのかわりとして画面分割の機能を用いると通常のコマンドの実行を編集作業を終了させずに行える。図3.8に示したようにTSSオプション(オプション番号6)を選ぶと下の論理画面ではTSSコマンドが入力できるようになる。図3.9にコマンド入力の例を示す。

```

EDIT --- F0052.ALIB.ASM(NXTCODE) ----- COLUMNS 001 072
COMMAND ==>                               SCROLL ==> HALF
000310 * SUBROUTINE      NXTCODE(CODE: CHAR(2))
000311 *   SUBR          NXTCODE(CODE: CHAR(2))
000320 *   CODE ORDER : 80A1 ... 80FE 81A1 ... .. A0FE 41A1 ... 79FE 80A1 ...
000400 *   MODIFY:  81.10.22  ADD CHECK OF 79FF
000500 *
000600 NXTCODE CSECT
000700 *----- PROLOGUE -----*
000800         USING      *,15
000900         B           START
001000         DC        XL1'07',CL7'NXTCODE'
001100 START      SVG      14,12,A(SAVE)
001200         LR        8,15
001300         DROP      15
001400         USING     NXTCODE,8
. . . . .
-----< TSS OPTION >-----
ENTER TSS COMMAND OR COMMAND PROCEDURE BELOW:

==> ALLOC F(FILE) DA(CCC.DATA) SH REU

```

図3.9 画面分割を利用したコマンド入力の例

プログラムの実行を行うためには、たとえば編集中の内容をSAVEコマンドを用いてデータセットに保存した後、TSSのRUNコマンドによって実行させることができる。現在のところ編集中の内容を実行させるにはこのような方法を使わざるを得ない。

なお、TSSオプションでは通常のコマンド、コマンドプロシジャはほとんど使えるが次に示すコマンドは利用できない。

TEST, LIBRARY, LOGON, LOGOFF, PFD, PERMIT, LISTDSD, LOG

コマンドの終了時には画面に\*\*\*が表示されブザーが鳴る。ここでENTERキーを押すとコマンドの実行は終了し画面の制御もPFDにもどる。

### 3.1.1 HELP機能

PFDのHELP機能はいろいろな場合に呼び出すことができる。HELPオプション(オプション番号7)として呼び出せるほかHELPキー(PF1)を押して呼び出せる。たとえばEDITの途中HELPキーを押すと図3.1.0に示すようなEDIT-HELP画面が表示される。HELP画面が表示されているときHELPキーを押すと図3.1.1に示すHELP-HELP画面が表示され、HELP機能そのものについての説明が表示される。また画面の1行目にエラーメッセージが表示されたときにHELPキーを押すと3行目に2次レベルのエラーメッセージが表示される。ここで再びHELPキーを押すと、そのときのエラーに関連したHELP画面が表示される。HELP画面の終了はENDキー(PF3)を押して行う。HELP画面が終了するともとの画面が再び表示される。

```
HELP -----< EDIT >----- HELP
NEXT TOPIC ==>
_
```

```
*****
*                EDIT                *
*****
```

THE FOLLOWING TOPICS ARE PRESENTED IN SEQUENCE, OR MAY BE SELECTED BY NUMBER.

0 - GENERAL INTRODUCTION	5 - EDIT AUTOMATIC RECOVERY
1 - SPECIFYING A DATA SET	6 - EDIT PROFILES
2 - SEQUENCE NUMBERING	7 - EDIT LINE COMMANDS
3 - DISPLAY MODES(CAPS/HEX/NULLS)	8 - EDIT PRIMARY COMMANDS
4 - TABBING (HARDWARE/SOFTWARE/LOGICAL)	9 - TERMINATING EDIT

図3.1.0 EDIT-HELP画面

```
HELP HELP -----< USE OF HELP >----- HELP HELP
NEXT TOPIC ==>
_
```

IN ADDITION TO SELECTING TOPICS FROM HELP PAGES, YOU MAY ENTER ONE OF THE FOLLOWING IN THE NEXT TOPIC FIELD:

BACK OR B - TO GO BACK TO THE PREVIOUS PAGE.  
 SKIP OR S - TO SKIP THE CURRENT TOPIC TO THE NEXT TOPIC.  
 UP OR U - TO DISPLAY HIGHER LEVEL PAGES.  
 TOP OR T - TO DISPLAY THE TABLE OF CONTENTS.

INDEX OR I - TO DISPLAY THE HELP INDEX.

YOU MAY USE THE FOLLOWING KEYS DURING THE HELP.

ENTER - TO DISPLAY THE NEXT SEQUENTIAL PAGE.

HELP - TO DISPLAY THIS PAGE.

END - TO EXIT FROM THE HELP.

UP - TO DISPLAY HIGHER LEVEL PAGES (INSTEAD OF TYPING UP ).

LEFT - TO DISPLAY THE PREVIOUS PAGE (INSTEAD OF TYPING BACK ).

RIGHT - TO DISPLAY THE NEXT SEQUENTIAL PAGE (INSTEAD OF PRESSING ENTER ).

(PRESS ENTER TO CONTINUE VIEWING THE HELP)

図 3.1 1 HELP-HELP 画面

HELP画面は図3.1 2に示すように階層構造になっており、コマンド、ファンクションキー、およびトピック番号によって移動させることができる。

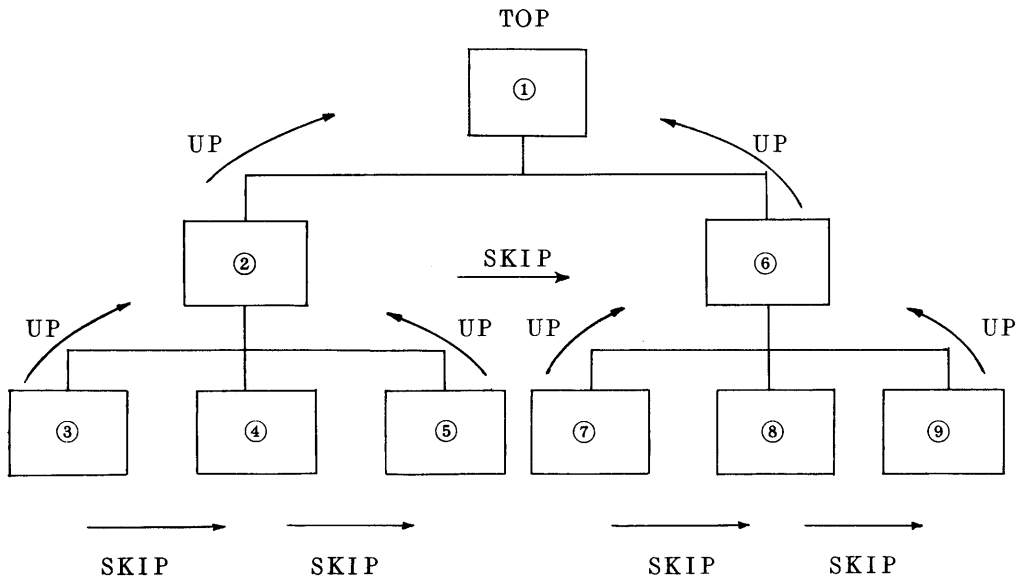


図 3.1 2 HELPの階層構造

(1) コマンドによる画面移動

(a) BACK (省略形 B)

直前のHELP画面を表示する。

(b) SKIP (省略形 S)

右隣の画面に移る。

- (c) UP (省略形U)  
階層の1つ上の画面に移る。
- (d) TOP (省略形T)  
階層の最上位の画面に移る。(図3.12の①の画面が最上位の画面である。)
- (e) INDEX (省略形I)  
HELPの目次画面(図3.13)を表示する。

(2) ファンクションキーによる画面移動

- (a) UPキー(PF7)  
UPコマンドと同じく階層が1つ上の画面に移る。
- (b) LEFTキー(PF10)  
BACKコマンドと同じく直前の画面に移る。
- (c) RIGHTキー(PF11)およびENTERキー  
次の画面へ移る。たとえば図3.12の場合①→②→③→……

(3) トピック番号による画面移動

HELP画面にトピック番号や図番号が表示されているとき、そのトピック番号や図番号を、NEXT

```
HELP -----< INDEX - 'A' TO 'C' >----- HELP
NEXT TOPIC ==> _
```

ENTER A CODE FROM A1 TO C22 TO SELECT A TOPIC:

A1 - ABEND CODES	C4 - CATALOG DATA SET NAME
A2 - AFTER 'A' LINE COMMAND	C5 - CATALOG MANAGEMENT UTILITY
A3 - ALLOCATE NEW DATA SET	C6 - 'CHANGE' PRIMARY COMMAND
A4 - ASSEMBLER (BACKGROUND)	C7 - CHANGE PF KEY (REPEAT CHANGE)
A5 - ASSEMBLER (FOREGROUND)	C8 - COBOL COMPILE (BACKGROUND)
A6 - ATTENTION (PA1) KEY	C9 - COBOL COMPILE (FOREGROUND)
A7 - 'AUTONUM' PRIMARY COMMAND	C10 - COBOL INTERACTIVE DEBUG
	C11 - 'COLS' PRIMARY COMMAND (BROWSE)
B1 - BACKGROUND OPTION	C12 - 'COLS' LINE COMMAND (EDIT)
B2 - BEFORE 'B' LINE COMMAND	C13 - COMMANDS - BROWSE COMMANDS
B3 - 'BOUNDS' LINE COMMAND	C14 - COMMANDS - EDIT LINE COMMANDS
B4 - BROWSE	C15 - COMMANDS - EDIT PRIMARY COMMANDS
B5 - BROWSE MEMBER (LIBRARY UTILITY)	C16 - COMMANDS - MEMBER LISTS
	C17 - COMPRESS DATA SET
C1 - 'CANCEL' PRIMARY COMMAND	C18 - COPY DATA SET, MEMBER, OR ALIAS
C2 - 'CAPS' PRIMARY COMMAND (BROWSE)	C19 - 'COPY' PRIMARY COMMAND
C3 - 'CAPS' PRIMARY COMMAND (EDIT)	C20 - COPY 'C' LINE COMMAND

図 3.1 3 HELP-INDEX 画面

TOPIC フィールドに入力すると、その番号に応じた画面が表示される。図 3.1 0 の EDIT-HELP 画面はトピック番号が表示されている画面の例である。

3.1 2 EDIT の終了

編集作業が終了 EDIT を終了させることにする。終了時の処理を定めるモードが 2 つある。AUTONUM モードと PRINT モードである。

AUTONUM モードは編集内容をデータセットに保存するとき行番号の付け直し——リナンバー——を行うかどうかを定める。NUMBER モードが OFF の場合は AUTONUM モードは意味をなさない。

PRINT モードは EDIT 内で何らかの修正が行われていた場合にデータセットの内容をリストするかどうかを定める。ただし PFD のリストというのはリストデータセットと呼ばれる特別なデータセットへ出力する意味である。実際に用紙に出力されるのは PFD 終了時のリストデータセットの後処理指定後である。

編集の正常終了はコマンドでなく END キー (PF3) を押して行う。このとき AUTONUM モードが ON なら行番号の付け直しが行われ、また PRINT モードが ON ならば編集データの内容がリストデータセットへ出力される。PFD の ATTRIBUTES オプションを用いて RETURN キーを定義してあるときは RETURN キーによって編集を正常終了させることもできる。END キーによる終了の場合は EDIT メンバメニューあるいは EDIT データセットメニューが表示される。RETURN キーによる終了の場合は画面はプライマリオプションメニューにまでもどる。

編集結果を保存せずに EDIT を終了させるには CANCEL コマンドを入力する。この場合は、編集作業そのものが無効になるため、行番号の付け直しや内容のリストなどは行われない。

今回の EDIT は END キーで終了させたものとしよう。画面は再びメンバメニュー (図 3.1 4) になる。

```

EDIT - MEMBER LIST -  F0052.ALIB.ASM ----- MEMBER NXTCODE SAVED
COMMAND ==>>>                                     SCROLL ==>>> PAGE
NAME -
NXTCODE          01.00  82/01/14  82/01/14 16:02   55   55   0  F0052
OB
**END**
    
```

図 3.1 4 編集後のメンバメニュー

図 3.1 4 を見ると STATS モードが ON であったため編集したメンバの更新履歴が表示されている。各項目はそれぞれ次の情報を示している。

VER. MOD : バージョン番号, 修正レベル数

CREATED :メンバを作成した日付  
 LAST MODIFIED :メンバを最後に更新した日時  
 SIZE :現在の行数  
 INIT :作成したときの行数  
 MOD :作成後追加, 修正された行数(非標準行番号のときは0)  
 ID :メンバを作成した課題番号

### 3.1.3 PFDの終了

ここでENDキー(PF3)を押すと画面はデータセットメニューになる。さらに続けてもう一度ENDキーを押すとプライマリオプションメニューが表示される。RETURNキーが定義されているならばRETURNキーを押すと直接プライマリオプションメニューに到達できる。

プライマリオプションメニューの画面が表示されたらENDキーを押す。ここで画面は終了メニュー(図3.1.5)になる。

```

-----< PFD TERMINATION MENU >-----
SPECIFY DISPOSITIONS OF LOG AND LIST DATASETS FOR THIS SESSION:
+-----+-----+-----+
| PARAMETERS      | LOG DATASET    | LIST DATASET   |
+-----+-----+-----+
| PROCESS OPTION  | ==> J         | ==> J         |
| SYSOUT CLASS   | ==> 0         | ==> 0         |
| LOCAL PRINTER ID | ==>          | ==>          |
+-----+-----+-----+
VALID PROCESS OPTIONS:
  J - SUBMIT JOB TO PRINT (AND DELETE)      K - KEEP DATASET WITHOUT PRINTING
  L - ROUTE TO LOCAL PRINTER (AND DELETE)   D - DELETE DATASET WITHOUT PRINTING
PRESS ENTER TO COMPLETE PFD TERMINATION
PRESS END KEY TO RETURN TO PRIMARY OPTION MENU
JOB STATEMENT INFORMATION TO SUBMIT JOB:
  ==> //FO052A  JOB CLASS=A,MSGCLASS=0
  ==> /**
  ==> /**   PFD LOG/LIST
  ==> /**
  
```

図3.1.5 PFD終了メニュー(PROCESS OPTIONを両方ともJと指定)

終了メニューはログデータセットおよびリストデータセットの後処理を指定するためのものである。PFDはどのデータセットの編集を行ったとかどういふTSSコマンドを使ったとかの情報をログデータセットと呼ばれる特別のデータセットに出力している。またデータセットのリストなどのプリント機能を利用した場合、その結果はリストデータセットと呼ばれる特別のデータセットに出力してあ

る。PFDを終えるにあたってこれらのデータセットをどう処理するか指定を求めているわけである。図3.15に示したようにPROCESS OPTIONその他の値を定める必要がある。その後ENTERキーを押すと、次のように後処理メッセージの出力後READY状態にもどる。

```
KEQ56283I ***          F0052A          : (RECEIVED) ***
          *** F0052A (JOB2359) 0052 : (JOB ACCEPTED) *** FIB  CN(01)
F0052.PFDLOG4.LIST WILL BE PRINTED AND DELETED BY JOB - F0052A.
F0052.PFD5.LIST WILL BE PRINTED AND DELETED BY JOB - F0052A.
READY
```

図3.16 PFDの終了メッセージ

図3.17にPFDが起動したFIBジョブの出力例を示す。

(a) FIBジョブのJCL

```
//F0052A JOB CLASS=A,MSGCLASS=0
***
*** PFD LOG/LIST
***
//HARDCOPY EXEC PGM=JSDGENER,COND=EVEN
*****
*** PRINT THE CONTENTS OF THE DATA SET OR MEMBER *
*****
//SYSUT1 DD DSN=F0052.PFDLOG4.LIST,
//          DISP=(OLD,DELETE,KEEP)
//SYSUT2 DD SYSOUT=(0),
//          DCB=(RECFM=VA,LRECL=125,BLKSIZE=129,OPTCD=)
//SYSPRINT DD DUMMY
//SYSIN DD DUMMY
*****
//HARDCOPY EXEC PGM=JSDGENER,COND=EVEN
*****
*** PRINT THE CONTENTS OF THE DATA SET OR MEMBER *
*****
//SYSUT1 DD DSN=F0052.PFD5.LIST,
//          DISP=(OLD,DELETE,KEEP)
//SYSUT2 DD SYSOUT=(0),
//          DCB=(RECFM=FBA,LRECL=121,BLKSIZE=3146,OPTCD=)
//SYSPRINT DD DUMMY
//SYSIN DD DUMMY
*****
```

(b) PFDのログ

```
TIME          ***** PFD TRANSACTION LOG *****          USERID: F0052          DATE: 82/01/14          PAGE: 1
15:48 START OF PFD SESSION LOG #23 -----
16:01 TSS - COMMAND - ALLOC F(FILE) DA(CCC.DATA) SH REU
16:02 EDIT - SAVE - F0052.ALIB.ASM(NXTCODE) - MEMBER SAVED
16:03 END OF PFD SESSION LOG #23 -----
```

図3.17 ログ/リストデータセット出力ジョブの例

#### 4. ATTRIBUTES オプション (オプション 0)

ATTRIBUTES オプションはユーザごとにPFD属性を定めるための機能である。プライマリオプションメニューでオプション番号0を指定するとATTRIBUTESメニュー(図4.1)が表示される。

```

-----< PFD ATTRIBUTES OPTION >-----
OPTION ==>
_

1  TERMINAL  - DEFINE TERMINAL CHARACTERISTICS
2  LOG/LIST  - DEFINE PFD LOG AND LIST DEFAULT VALUES
3  PF KEYS   - DEFINE PF KEYS FOR F9526 TERMINAL WITH 24 PF KEYS

```

図4.1 ATTRIBUTESメニュー

ATTRIBUTES オプションはサブオプションとして TERMINAL, LOG/LIST, PF KEYS の 3 つをもっており、それぞれサブオプション番号 1, 2, 3 で指定する。ATTRIBUTES メニューを経由しないでプライマリオプションメニューから直接サブオプションを選択する方法がある。それにはオプション番号として # オプション番号. サブオプション番号 # の形を指定する。たとえば ATTRIBUTES オプションの中の TERMINAL サブオプションを直接指定したいときは、プライマリオプションメニュー画面でオプション 0.1 を入力するとよい。

#### 4.1 TERMINAL サブオプション (オプション 01)

TERMINAL サブオプションは現在使用中の端末に関するタイプ、PF キーの数、表示モード、カナ属性を定めるものである。端末属性メニュー (図 4.2) は 4 つの入力フィールドをもっている。属性は次のように定める。

##### (1) TERMINAL TYPE フィールド

端末の種別を定める。PFD を使える端末は画面が 24 行 80 カラムの大きさのディスプレイ端末であり F9525, F9526, F6650 の 3 つの型に分けられる。キーボードについている PF キーの数が 12 個のものが F9525 型である。PF キーの数が 24 個についている端末の中で日本語表示が可能なものが F6650 型であり、それ以外が F9526 型であるということになる。

##### (2) NUMBER OF PF KEYS フィールド

PF キーの数を指定する。F9525 型の端末の場合は 12 に限られる。

```

-----< TERMINAL CHARACTERISTICS >-----
SPECIFY PARAMETERS BELOW:

TERMINAL TYPE      ==> F9526          (F9525, F9526 OR F6650)
NUMBER OF PF KEYS ==> 24           (12 OR 24)
DISPLAY MODE       ==> DUAL        (MONO OR DUAL)
KANA FEATURE       ==> YES         (YES OR NO)

```

```

+-----+
| DUAL MEANS THAT THE TERMINAL DISPLAYS BOTH UPPER AND |
| LOWER CASE CHARACTERS. |
| MONO MEANS THAT THE TERMINAL DISPLAYS ONLY UPPER CASE |
| CHARACTERS. |
| ENTERING 'YES' IN THE KANA FEATURE FIELD MEANS THAT THE |
| TERMINAL DISPLAYS UPPER CASE CHARACTERS AND KANA |
| CHARACTERS. |
+-----+

```

図 4.2 端末属性メニューの例

(3) DISPLAY MODE フィールド

小文字を大文字として表示する端末の場合は MONO を、それ以外の場合は DUAL を指定する。

(4) KANA FEATURE フィールド

カナ付きキーボードを使用している場合は YES を、それ以外の場合は NO を指定する。

指定後 END キー (PF3) を押すと ATTRIBUTES メニューにもどる。

#### 4.2 LOG / LIST サブオプション (オプション 02)

PF D の終了時には、通常ログ/リストデータセットの後処理を指定しなければならない。後処理の指定は終了メニュー (図 3.15) によって行ってもよいが、省略値を定めておいてそれに従って後処理を行うことも可能である。後者の方法はプライマリオプションメニューの入力フィールドにオプション X を指定して行われる。しかし、この機能を使うためにはあらかじめ省略値を定めておかなければならない。省略値を定めるのが LOG / LIST サブオプションである。

ログ/リスト パラメタメニュー (図 4.3) はそれぞれのデータセットごとに 6 種類のパラメタ、また FIB ジョブのためのジョブ文を定めるための計 13 の入力フィールドをもっている。

(1) PROCESS OPTION フィールド

データセットの後処理を定める。

J : データセットの内容を出力するための FIB ジョブを起動する。出力後データセットは削除される。この指定を行う場合はジョブ文を定めておかなければならない。

L : データセットの内容をローカルプリンタに出力する。出力後データセットは削除される。

K : データセットを保存する。次回の PFD では同じデータセットに追加出力する。

D : データセットを削除する。出力は行わない。

(2) SYSOUT CLASS フィールド

PROCESS OPTION J を指定したときに必要な、出力のための SYSOUT CLASS を指定する。

(3) LOCAL PRINTER ID フィールド

PROCESS OPTION L を指定したときに、出力のためのローカルプリンタ ID を指定する。

- (4) LINES PER PAGE フィールド  
プリントするときに1ページ当たり何行出力するかを指定する。
- (5) PRIMARY PAGES フィールド  
データセットの一次割り当て量をページ数で指定する。PFDのログを取りたくない場合は0を指定すればよい。
- (6) SECONDARY PAGES フィールド  
データセットの割り当て量の増分値をページ数で指定する。
- (7) JOB STATEMENT INFORMATION フィールド  
PROCESS OPTIONがJの場合に起動されるFIBジョブのジョブ文を与える。

-----< LOG AND LIST DEFAULTS >-----  
SPECIFY PARAMETERS BELOW:

PARAMETERS	LOG DATASET	LIST DATASET
PROCESS OPTION	====> D	====> J
SYSOUT CLASS	====> 0	====> 0
LOCAL PRINTER ID	====>	====>
LINES PER PAGE	====> 60	====> 60
PRIMARY PAGES	====> 10	====> 100 **
SECONDARY PAGES	====> 10	====> 200 **

(\*\*) EFFECTIVE NEXT SESSION

VALID PROCESS OPTIONS:

J - SUBMIT JOB TO PRINT (AND DELETE)    K - KEEP DATASET WITHOUT PRINTING  
L - ROUTE TO LOCAL PRINTER (AND DELETE)    D - DELETE DATASET WITHOUT PRINTING

JOB STATEMENT INFORMATION TO SUBMIT JOB:

```
====> //F0052B    JOB CLASS=A,MSGCLASS=0
====> /**
====> /**    PFD LOG/LIST
====> /**
```

図 4.3 ログ/リスト パラメタメニューの例

### 4.3 PF KEYSサブオプション (オプション03)

PFDではファンクションキー定義メニューを用いてPFキーの動きをユーザが定めることができる。図4.4は24キーの場合の定義例である。PF1～PF12については標準値をそのまま残してあ

る。PF13～PF24についての標準値はそれぞれ12引いたPFキーと同じ値になっている。たとえばPF13の標準値はPF1と同じくHELPである。

PFキーの機能として定義可能なものは次に示す17種類である。「機能(標準定義の場合に割り当てられているPFキー)」の形式で示す。

(1) HELP(PF1,PF13)

PF1の使い方を説明するHELP画面を表示する。また1次レベルのエラーメッセージが表示されているときに2次レベルのエラーメッセージを表示させるためにも用いられる。HELP画面を終了させもとの画面にもどるときはENDキーを用いる。

(2) SPLIT(PF2,PF14)

画面分割を行う。キーを押したときのカーソルのある行を境界として上下2つの論理画面に分

```

-----< PROGRAM FUNCTION KEY DEFINITION >-----
SPECIFY PF KEY USAGE FOR F9526/F6650:

PF1 ==> HELP          PF13 ==> >PROF
PF2 ==> SPLIT         PF14 ==> >RESET
PF3 ==> END           PF15 ==> RETURN
PF4 ==> PRINT         PF16 ==> PRINT-HI
PF5 ==> FIND          PF17 ==> :TS
PF6 ==> CHANGE        PF18 ==> :COLS
PF7 ==> UP            PF19 ==> >NULLS ON
PF8 ==> DOWN          PF20 ==> >NULLS OFF
PF9 ==> SWAP          PF21 ==> :V
PF10 ==> LEFT         PF22 ==> >DUP
PF11 ==> RIGHT        PF23 ==> >TRA
PF12 ==> CURSOR       PF24 ==> CURSOR

VALID FUNCTIONS: (ENTER BLANK TO USE DEFAULT FUNCTIONS)

HELP   SPLIT   END     PRINT   FIND    CHANGE
UP     DOWN   SWAP   LEFT   RIGHT   CURSOR
NOP    RETURN  PRINT-HI

OR ENTER EDIT/BROWSE COMMANDS AS FOLLOWS:

>CMD... (PRIMARY COMMAND)   EXAMPLE: PF10 ==> >CAPS ON
:CMD... (EDIT LINE COMMAND) EXAMPLE: PF11 ==> :D
    
```

図 4.4 ファンクションキー定義メニューの例

かれる。すでに分割されているときの境界行の変更にも用いられる。一方の画面のPF1が終了すると他方だけの単一画面になり画面分割は終了する。

## (3) END (PF3, PF15)

PFDの中で各オプション、HELP等の機能を終了し、呼び出しの階層を1レベル上げる。PF1～PF15の中に少なくとも1つはこのキーが存在しなければならない。

## (4) PRINT (PF4, PF16)

現在表示中のPFD画面をリストデータセットに出力する。画面のハードコピーを得るのに用いる。

## (5) FIND (PF5, PF17)

BROWSE (オプション1) およびEDIT (オプション2) で用いられる。最近指定されたFINDコマンドまたはCHANGEコマンドで指定された文字列を探す処理を行う。

## (6) CHANGE (PF6, PF18)

EDIT (オプション2) で用いられる。最近指定されたCHANGEコマンドの処理を行う。FINDキーを用いて目的の文字列を探し出した後CHANGEキーを押して文字列の置き換えを行なうことができる。この方法によると選択的に文字列の置き換えを行うことができる。

## (7) UP (PF7, PF19)

画面を上へ移動させる。移動量は画面右上のSCROLLフィールドによって定まる。

## (8) DOWN (PF8, PF20)

画面を下へ移動させる。移動量はSCROLLフィールドによって定まる。

## (9) SWAP (PF9, PF21)

画面分割の際、カーソルを一方の論理画面から他方の論理画面の元の位置に移動させる。もし一方の論理画面が4行以下の場合、画面の行数も同時に交換されカーソルは行数の多い側の画面に移る。

## (10) LEFT (PF10, PF22)

画面を左へ移動させる。移動量はSCROLLフィールドによって定まる。

## (11) RIGHT (PF11, PF23)

画面を右へ移動させる。移動量はSCROLLフィールドによって定まる。

## (12) CURSOR (PF12, PF24)

カーソルの位置を入力フィールドに移す。すでに入力フィールドにあるときはSCROLLフィールドへ移す。

## (13) RETURN

直接プライマリオプションメニューにもどる。ENDキーを何度か押してもどることに相当する。適当な入力フィールドに“=オプション番号”を入力した後、ENTERキーでなくRETURNキーを押すとプライマリオプションメニューを経由せずに直接指定のオプション画面を表示させることができる。また=Xを入力後RETURNキーを押してPFDを直ちに終了させることもできる。

## (14) PRINT-HI

PRINTキーと異なり画面上の高輝度の部分はラインプリンタの重ね打ちを利用して濃く出力される。このキーによる出力の場合は終了メニューで出力クラスをWと指定しなければならない。

## (15) NOP

何も行わない。このPFキーは無視される。

(16) >コマンド

BROWSEおよびEDITで用いるコマンドを登録しておけば、そのPFキーを押すだけでコマンドを入力できる。たとえば>HEX ONと定義されたPFキーを押すと画面は16進数表示になる。

(17) :行コマンド

EDITで用いる行コマンドを登録しておけば、そのPFキーを押すだけで、そのときカーソルの存在する行に、登録された行コマンドが入力された場合の処理が行われる。たとえば:TSと定義されたPFキーを押すとそのときのカーソル位置で行の分割が行える。

以上がPFキーに割り当てることのできる機能である。HELP~CURSORの12個以外は各自必要に応じてPFキーに割り当てて使わなければならない。

## 5. EDIT オプション (オプション2)

EDITで用いられるコマンド、行コマンドをいくつかのテーマに分けて説明する。

### 5.0 : 開始, 終了 — CANCEL コマンド

#### 5.1 : モード — AUTONUM, CAPS, HEX, NULLS, NUMBER, PRINT, PROF, RECOVERY, RENUM, STATS, UNNUM コマンド

#### 5.2 : 画面移動 — LOCATE, FIND コマンド

#### 5.3 : 行機能 — TABS コマンド, BNDS, COLS, MASK, TABS 行コマンド

#### 5.4 : 編集機能 — CHANGE コマンド, A, B, C, D, I, M, O, R, <, >, (, ) 行コマンド

#### 5.5 : 内容の転送 — COPY, CREATE, MOVE, REPLACE, SAVE コマンド

#### 5.6 : 行の表示 — F, L, S, X 行コマンド

#### 5.7 : テキスト — TE, TF, TS 行コマンド

#### 5.8 : その他 — SUBMIT, RESET コマンド

### 5.0 開始, 終了

すでに3でも示したようにEDITオプションを開始するにはプライマリオプションメニューにオプション2を入力する。RETURNキーが定義されている場合は、それ以外の画面から直接EDITを開始できる。すなわち適当な入力フィールドに=2を入力した後ENTERキーのかわりにRETURNキーを押す方法である。RETURNキーの働きで、一度プライマリオプションメニューの状態にもどり、オプション2を選択した後画面が表示される。

EDITの最初の画面はデータセットメニューである。区分データセットを指定し、メンバ名を省いた場合はメンバメニューを経てEDIT画面が表示される。

現在のデータセットの編集を終了し、引き続き別の編集を開始したいときはENDキー(PF3)を押す。そこで新しいデータセットを指定して再び編集を行うことができる。編集作業を終了し、次にEDIT以外のPFDのオプションを使いたいときはコマンド入力フィールドに=を入力しそのあとにオプション番号を指定しRETURNキーを押す。たとえば=0とするとATTRIBUTES画面に移る。=0.1のようにサブオプション番号をつけた形も許される。編集作業も含めPFDそのものを終

了させるには=Xを入力しRETURNキーを押す。ログ/リストデータセットの後処理の省略値が定義されているならばPFDは終了する。定義されていないならば終了メニューが表示されるので必要な指定の後ENTERキーを押すとPFDは終了する。

編集結果を保存しないでEDITを終了させるにはCANCELコマンドを用いる。CANCELコマンドはオペランドなしでCANCELまたはCANと入力する。CANCELコマンドによる終了の場合は、次の画面はメンバメニューもしくはデータセットメニューとなる。

### 5.1 モード

EDITはいくつかのモードをもっておりモードのON/OFFに応じて処理形態を変える。このモードの値はデータセットのタイプ名、レコード形式およびレコード長の3つの属性の組ごとに1つのプロフィール情報として保存され、後続のEDITで利用される。

表 5.1 モード一覧

モ ー ド	モ ー ド の 意 味
NUMBERモード	行番号付きか行番号なしか、および行番号はレコード内のどの位置か。
AUTONUMモード	編集内容をデータセットに保存するとき行番号の付け直しを行うかどうか。
STATSモード	メンバごとの更新履歴をとるかどうか。
PRINTモード	修正された内容をデータセットに保存するとき内容のリストをとるかどうか。
CAPSモード	英小文字を英大文字に変換するかどうか。
NULLSモード	行末の空白をNULL文字として表示するかどうか。
TABSモード	ハードウェアタブあるいは論理タブ機能を用いるかどうか。
HEXモード	編集データの16進表現を表示するかどうか。
RECOVERYモード	編集データのリカバリ機能を使用するかどうか。

#### 5.1.0 PROFILEコマンド

{ PROFILE PROF PRO }	[プロフィール名][行数]
----------------------------------	---------------

PROFILEコマンドはプロフィールの表示、生成、切り換えを行う。プロフィールとは種々のモードのON/OFF値のほかタブ位置その他の編集作業の環境情報の集まりである。プロフィールは名前をつけて保存され後続のEDITで使われる。

EDITの際に使われるプロフィール名はデータセットメニューのPROFILE NAMEフィールドに指定されたものである。指定がないときはデータセットのタイプ名がプロフィール名となる。図5.1はプロフィール表示の例である。プロフィールはプロフィール名、レコード形式、レコード長の3つの組に対

して1つ定められる。

```
=PROF> ....ASM (FIXED - 80)....RECOVERY OFF....NUMBER ON STD.....
=PROF> ....CAPS OFF....HEX OFF....NULLS ON STD....TABS ON STD.....
=PROF> ....AUTONUM ON....PRINT OFF....STATS ON.....
=BND5> < >
=COLS> -----1-----2-----3-----4-----5-----6-----7---
```

図 5.1 プロフィール表示の例

プロフィールが生成されるのはそのプロフィールのEDITが最初に行われたとき、またはPROFILEコマンドで新しいプロフィール名を指定したときである。プロフィールの値は前者についてはEDIT終了時、後者についてはコマンド実行時の値が採用される。プロフィールは最大25個登録できる。25個を越えたときは最も古いプロフィールと置き換える。PROFILEコマンドにより図5.1のようなプロフィール情報が表示されるが、PROFILEコマンドの行数指定によって表示する行数を0～7の範囲で変えられる。表示行を消去するのはRESETコマンドまたはD行コマンドによる。

### 5.1.1 NUMBERモード

NUMBERモードはデータセットを行番号付きとして扱うか、行番号なしとして扱うかを指定する。行番号に関連するコマンドはNUMBER、RENUM、UNNUMの3つである。

{ NUMBER }	[ ON ]	[ STD ]	{ COBOL }
{ NUM }	[ OFF ]		{ COB }

NUMBERコマンドはNUMBERモードのON/OFFおよび行番号が標準形式かどうか、COBOL形式行番号であるかどうかの指定を行う。NUMBERモードをONにするとき行番号が形式に合わないときは自動的に行番号の付け直しを行う。

STDは標準形式行番号であることを指定する。すなわち固定長レコードならば終わりの8桁、可変長レコードならば初めの8桁が行番号フィールドであることを意味する。

COBOLまたはCOBはCOBOL形式行番号であることを示す。すなわちレコード形式は固定長でありレコードの先頭6桁は行番号フィールドであるものとする。

行番号付きのとき行番号フィールドの位置の形式にSTD、COBOL、STD & COBOLの3つがある。STD & COBOLの場合は2か所に行番号フィールドが存在する。

STATSモードがONのとき8桁の標準行番号フィールドはさらに6桁の行番号部分と2桁の更新レベル部分の2つに分けて使われる。修正、追加された行の更新レベル部分には編集時のデータセットの更新レベルが設定される。

{ RENUM }	[ STD ]	{ COBOL }
{ REN }		{ COB }

RENUMコマンドは行番号の付け直しを行う。標準としては初期値100、増分値100である。

STD, COBOLの2つのオペランドはどの行番号フィールドを付け直しの対象とするかを指定するものであり普通は省略してかまわない。PFD-EDITはTSS-EDITと異なり行の挿入の際は必要に応じて行番号をずらすため挿入行のための行番号をあけるリナンバは必要ない。

{ UNNUM }	
{ UNN }	

UNNUMコマンドはレコード内の行番号フィールドをすべて空白にしNUMBERモードをOFFにする。

### 5.1.2 AUTONUMモード

AUTONUM	[ <u>ON</u> ] [ OFF ]
---------	--------------------------

AUTONUMモードはNUMBERモードがONのとき、編集内容をデータセットに保存する際リナンバするかどうかを定める。

### 5.1.3 STATSモード

STATS	[ <u>ON</u> ] [ OFF ]
-------	--------------------------

STATSモードはメンバの更新履歴をとるかどうかを定める。ONのときメンバのディレクトリに更新履歴を、そして標準形式行番号フィールドの下2桁に更新レベル数を設定する。順データセットではSTATSモードは常にOFFとなる。

### 5.1.4 PRINTモード

PRINT	[ <u>ON</u> ] [ OFF ]
-------	--------------------------

PRINTモードは修正の行われたEDITをENDキーまたはRETURNキーで終了させるとき編集内容をリストデータセットに出力するかどうかを定める。

### 5.1.5 CAPSモード

CAPS	[ <u>ON</u> ] [ OFF ]
------	--------------------------

CAPSモードは画面のデータ表示部分、FINDコマンド、CHANGEコマンドの文字列指定内の英小文字を英大文字に変換するかどうかを指定する。

### 5 1 6 NULLSモード

NULLSモードは行末の空白をNULL文字として表示するかどうかを定める。

$\left\{ \begin{array}{l} \text{NULLS} \\ \text{NULL} \end{array} \right\}$	$\left[ \begin{array}{l} \text{ON} \\ \text{OFF} \end{array} \right]$	[ALL]

ALLを指定すると行末の空白はすべてNULL文字として表示される。指定しないと行末の空白のうち先頭の空白以外がすべてNULL文字として表示される。ただしカーソルがデータフィールド内にあるとき、そのフィールドに関しては先頭からカーソル位置までの空白はすべて空白として表示される。

### 5.1.7 TABSモード

TABSモードはハードウェアタブ機能あるいは論理タブ機能を用いるかどうかを指定する。

$\left\{ \begin{array}{l} \text{TABS} \\ \text{TAB} \end{array} \right\}$	$\left[ \begin{array}{l} \text{ON} \\ \text{OFF} \end{array} \right]$	[タブ文字] [ALL]

TABSコマンドは実際には次の3つの形式で用いられる。

- (1) ハードウェアタブモードをONにするとき  
TABS [ON] [ALL]
- (2) 論理タブモードをONにするとき  
TABS [ON] タブ文字
- (3) ハードウェアタブ、論理タブの両方のモードを共にOFFにするとき  
TABS OFF

(1)でALLを指定しないとタブ位置が空白もしくはNULL文字の場合だけその位置にアトリビュート文字を設定する。指定すると常にアトリビュート文字を設定する。アトリビュート文字が設定されるとその位置の文字は表示されないが文字そのものが削除されるわけではない。

(2)で指定するタブ文字は英数字以外の1文字でなければならない。

ここでEDITで用いることのできる3つのタブ機能について説明する。

- (a) ソフトウェアタブ機能
- (b) ハードウェアタブ機能
- (c) 論理タブ機能

ソフトウェアタブ機能はENTERキーを押したときカーソルの位置をどこにするかを定める機能である。カーソルの位置を設定するコマンドや行コマンドが入力された場合はそれに従うが、それ以外の場合はソフトウェアタブ機能の指定に従う。通常は次の行へ移動する。タブ位置の設定はTABS行コマンドを用いてタブ定義行を表示させ、その行に下線かハイフンによってタブ位置を指定する。連続したタブ位置指定が1つのタブフィールドを定める。カーソルはあるタブフィールドから次のタブフィールドへと順に移ってゆく。同一行の最後のタブフィールドに達したら次は次の行の先頭のタブフィールドへとカーソルは移動する。

例：

```

=TABS>      ---      -----
000100      TOKYO
000200      ①      ②
000300      ③      ④
000400      ⑤      ⑥
              O SAKA
              ⑦      ⑧

```

ENTERキーを押していくとカーソルは①→②→③→……と移動する。

ハードウェアタブ機能はタブ位置にアトリビュート文字を置くことでTABキー、BackTABキー等を用いてカーソルをアトリビュート文字の直後に移動させることができるようにする機能である。これらのフィールド指向キーによるカーソルの移動はディスプレイ装置の機能を利用して行われるものなのでこのように呼ばれる。タブ位置はタブ定義行に\*を入力して指定する。この機能を用いるときはハードウェアタブ機能をONにしなければならない。アトリビュート文字の上には文字の書き込みができないため一時的に行のアトリビュート文字を除去する方法が2種類用意されている。第1の方法は除去したい行の行番号フィールドをすべて空白にした後ENTERキーを押す方法で、第2の方法はアトリビュート文字の上にカーソルを合わせENTERキーを押す方法である。いずれの方法でもその行のアトリビュート文字は除去され、次のENTERキーによって再び設定される。

論理タブ機能はタブ文字として指定した文字の入力をTABキーの入力とみなす機能である。この機能を用いるためには論理タブモードをONにしておかなければならない。

例：タブ文字が¥のとき

```

=TABS>      *      *      *
000100  ¥A¥B¥C
              ↓↓ ENTERキー
=TABS>      *      *      *
000100      A      B      C

```

### 5.1.8 HEXモード

HEXモードはデータの内容に16進数表現も加えて表示するかどうかを定める。

HEX	$\left[ \begin{array}{c} \text{ON} \\ \text{OFF} \end{array} \right]$	$\left[ \begin{array}{c} \text{VERT} \\ \text{DATA} \end{array} \right]$
-----	---	--

VERT/DATAは16進数表現を縦に表示するか横に表示するかを定める。

例：HEX DATA のとき

```

000100 ABCD E
          C1C2C3C440C5

```

HEX VERT のとき

```

000100 ABCD E

```

CCCC4C

123405

### 5.1.9 RECOVERYモード

RECOVERYモードはEDITのリカバリ機能を用いるかどうかを定める。

{ RECOVERY }	[ ON ]
{ REC }	[ OFF ]

リカバリ機能がONのとき編集操作のたびに編集内容がリカバリデータセットに記録される。編集途中でシステムダウンが生じても編集記録をもとにダウン時の内容が復元できる。編集の終了時にリカバリデータセットは削除される。システムダウンの場合のみリカバリデータセットは保存され、次のPFD-EDITの際リカバリメニューが表示され復元を行うかどうか指定できる。

## 5.2 画面移動

編集を行いたい文字列や行が表示されるように画面を移動することは画面編集を行うときの基本作業である。画面移動はSCROLLキーまたはコマンドによって行われる。PFD-EDITではいずれの方法についてもFSOと比べ、きめ細かい指定が可能である。

### 5.2.1 SCROLLキーによる画面移動

画面を上、下、左、右に移動させるにはそれぞれUP(PF7),DOWN(PF8),LEFT(PF10),RIGHT(PF11)のいわゆるSCROLLキーを用いる。移動の方向はキーの種類により定まるが移動量(行数/カラム数)はSCROLLフィールドもしくはコマンド入力フィールドに与える数によって定められる。

コマンド入力フィールドに数字を入力した後SCROLLキーを押すと、入力した行数またはカラム数の画面移動がなされる。

コマンド入力フィールドへの数字入力が行われていない場合、移動量はSCROLLフィールドの値によって定められる。SCROLLフィールドに与えることのできる値は次の5種類がある。

(1) 数字(1~9999)

与えられた行数/カラム数の移動を行う。

(2) PAGE

表示画面の分の行数/カラム数の移動を行う。

(3) HALF

表示画面の半分の行数/カラム数の移動を行う。

(4) MAX

移動可能な最大行数/カラム数の移動を行う。たとえばUPキーを押すとデータセットの先頭行へ移動する。これはFSOのTOPサブコマンドによる移動と同じである。この指定は1回限りであり移動後の画面ではSCROLLフィールドはもとの値にもどっている。

(5) CUR

現在カーソルのある行/カラムが画面の端に来るように移動する。すでにカーソルがその位置にあるか、もしくはデータ表示域外にある場合はPAGE単位の移動となる。たとえばDOWNキーを押すと、そのときのカーソルのある行が画面上の先頭行となる。

5.2.2 コマンドによる画面移動

画面を移動させるコマンドとしてLOCATEとFINDがある。CHANGEコマンドによっても移動が起こるが、このコマンドは編集を目的とするものなので「5.4 編集機能」で説明する。

(a) LOCATE コマンド

$\left\{ \begin{array}{l} \text{LOCATE} \\ \text{LOC} \\ \text{L} \end{array} \right\}$	行番号
---	-----

LOCATEコマンドは指定された行番号をもつ行が先頭となるように画面移動を行う。ここでいう行番号は、行番号付きで編集を行っているときは行番号フィールドの値であり行番号なしのときは相対レコード番号（先頭が1，以下順に2，3，……となる）のことである。指定された行が見つからないときはその直前の行を指定したものとみなされる。したがって正確な行番号を知らないときは、おおよその見当で画面を移動した後SCROLLキーによって目的の行を見つけるとよい。

(b) FIND コマンド

$\left\{ \begin{array}{l} \text{FIND} \\ \text{F} \end{array} \right\}$	文字列	$\left[ \begin{array}{l} \text{NEXT} \\ \text{ALL} \\ \text{FIRST} \\ \text{LAST} \\ \text{PREV} \end{array} \right]$	$\left[ \begin{array}{l} \text{CHARS} \\ \left\{ \begin{array}{l} \text{PREFIX} \\ \text{PRE} \end{array} \right\} \\ \left\{ \begin{array}{l} \text{SUFFIX} \\ \text{SUF} \end{array} \right\} \\ \text{WORD} \end{array} \right]$	$\left[ \begin{array}{l} \text{X} \\ \text{NX} \end{array} \right]$	[ カラム 1 [ カラム 2 ] ]
---	-----	---	---	---	---------------------

FINDコマンドは指定された文字列を含む行が表示されるように画面を移動し、その文字列の先頭にカーソルを置く。文字列の指定のしかた、文字列の探し方をオペランドで与えることができる。オペランドの意味と指定方法を順に説明する。

(1) 文字列

このオペランドは探すべき文字列を指定する。次の6種類の方法がある。

(i) そのままの文字列

空白、コンマを含まない文字列は以下の(ii)～(vi)と異なる形である限りそのまま指定できる。ただしコマンドのオペランドとして文字列だけの指定の場合を除き他のオペランドと一致する文字列はそのまま指定してはならない。

例：F ABC ………文字列 ▼ ABC ▼ を探す。

(ii) 引用符つき文字列

引用符で囲まれた文字列。ただし文字列中に「 $\square$ および $\nabla$ 」を含んではならない。引用符として2重引用符を用いてもよい。

例：F  $\nabla$  A  $\square$  B  $\nabla$  ……文字列  $\nabla$  A  $\square$  B  $\nabla$  を探す。

(iii) 16進文字列

引用符つき文字列の前か後ろにXをつけたもの。文字列部分は指定の文字列の16進表現でなければならない。

例：F X  $\nabla$  C1C2F3  $\nabla$  ……F AB3と同じく文字列  $\nabla$  AB3  $\nabla$  を探す。

F  $\nabla$  0BFF  $\nabla$  X

(iv) テキスト文字列

引用符つき文字列の前か後ろにTをつけたもの。英大文字、英小文字の区別を無視する。

例：F T  $\nabla$  PL/I  $\nabla$  ……  $\nabla$  pl/i  $\nabla$ ,  $\nabla$  Pl/I  $\nabla$ ,  $\nabla$  pL/I  $\nabla$ ,  $\nabla$  PL/I  $\nabla$  などの文字列を探す。

ただし端末属性がカナつきと定義されているときはテキスト指定は無視され、通常の引用符つき文字列として処理される。

(v) ピクチャ文字列

引用符つき文字列の前か後ろにPをつけたもの。文字列の種別を指定するものであり、各ピクチャ文字は次の意味をもつ。

=	任意の文字
␣	空白以外の文字
•	表示不能文字
#	数字
—	数字以外の文字
@	英字（大文字または小文字）
<	英小文字またはカナ文字
>	英大文字
¥	特殊文字（英数字、空白以外の文字）
英数字	英数字そのもの
その他	指定できない

例：F P  $\nabla$  A#  $\nabla$  ……  $\nabla$  A0  $\nabla$ ,  $\nabla$  A1  $\nabla$ , …,  $\nabla$  A9  $\nabla$  を探す。

(vi) \*

以前に入力したFINDコマンドで指定した文字列と同じ文字列を指定する。

例：F XYZ

⋮

F \* ……文字列  $\nabla$  XYZ  $\nabla$  を探す。

- (2)  $\left[ \begin{array}{c} \underline{\text{NEXT}} \\ \text{ALL} \\ \text{FIRST} \\ \text{LAST} \\ \text{PREV} \end{array} \right]$

文字列を探す方向および起点を指定する。

(i) NEXT

現在のカーソル位置からデータセットの終端に向かって探す。カーソルがデータ表示行にないときは先頭のデータ表示行から探す。

(ii) ALL

データセットの先頭から終端へ向って探し、見つかった文字列の個数を表示する。カーソルは見つかった最初の文字列の位置にある。

(iii) FIRST

データセットの先頭から終端に向って探す。

(v) PREV

NEXTとは逆に、現在のカーソル位置からデータセットの先頭に向って探す。カーソルがデータ表示行にないときはデータ表示行の先頭から探す。

- (3)  $\left[ \begin{array}{c} \underline{\text{CHARS}} \\ \left\{ \begin{array}{c} \text{PREFIX} \\ \text{PRE} \end{array} \right\} \\ \left\{ \begin{array}{c} \text{SUFFIX} \\ \text{SUF} \end{array} \right\} \\ \text{WORD} \end{array} \right]$

文字列が単語（連続した英数字）内のどの位置にあるかを指定する。

(i) CHARS

単語内のどの位置にあってもよい。単なる文字列として探す。

(ii) PREFIX/PRE

単語内の先頭にある。

(iii) SUFFIX/SUF

単語内の末尾にある。

(iv) WORD

単語全体と一致する。

例：それぞれのコマンドの指定に該当する部分を下線で示す。

F IS CHARS : IS ISNT ASIS KISS IS (ISNT)

F IS PRE : IS ISNT ASIS KISS IS (ISNT)

F IS SUF : IS ISNT AS IS KISS IS (ISNT)

F IS WORD : IS ISNT ASIS KISS IS (ISNT)

(4)  $\begin{bmatrix} X \\ NX \end{bmatrix}$

5.6で述べるようにEDITではコマンド/行コマンドによって行の表示をON/OFFできる。オペランドXを指定すると表示されていない行のみを探す。NXを指定すると表示行のみを探す。両オペランドを指定しない場合はすべての行を探す。表示されていない行の中で文字列が見つかったならばその行は表示行になる。

(5) [ カラム 1 [ カラム 2 ] ]

文字列を探すカラム範囲を指定する。カラム1だけの指定のときはカラム1で始まる文字列を探す。カラム1, カラム2の両方指定のときはカラム1からカラム2の間の文字列を探す。このオペランドを指定しないときはカラム境界範囲(5.3参照)内の文字列を探す。

FINDコマンドを入力した後FINDキー(PF5)を押すと同じ文字列のFINDを繰り返して行うことができるため同じ文字列を探すために何度も同じFINDコマンドを呼ばなくてもよい。

例: F A コマンドを入力する。

```

ABCA1BAC
  ↓ FIND キー
ABCA1BAC
  ↓ FIND キー
ABC A1BAC
    
```

### 5.3 行機能

EDITの定義行には境界定義行, マスク定義行, タブ定義行があり, 表示行にはカラム表示行がある。定義行はEDITがもっていて利用しているカラムまたはカラム範囲についての情報を表示, 変更するためのものであり, 表示行はカラム位置を知るための目盛りを表示するためのものである。

これらの行を表示させるにはそれぞれの表示を指示する行コマンド, またはPROFILEコマンドによる。削除するにはD行コマンドによって行ごとに行うか, もしくはRESETコマンドによってすべての定義行, 表示行を削除する。

#### (1) 境界定義行

行コマンドBOUNDS(省略形: BOUND, BNDS, BND)はカラム境界範囲を表示, 変更するために用いる。カラム境界範囲はどのカラム範囲を編集作業の対象とするかを定めており次のようなカラム範囲として用いられる。

- (a) <, >, (, )のシフト行コマンドによるシフトの対象となるカラム範囲。
- (b) CHANGE, FINDコマンドでカラム範囲の指定が省略された場合の文字列検索のカラム範囲。
- (c) CHANGEコマンドで文字列を置き換える際の文字列シフトを行なうカラム範囲。
- (d) O行コマンドで行の重ね合わせを行うカラム範囲。
- (e) TS行コマンドで分割の対象となるデータのカラム範囲。
- (f) TE, TF行コマンドでテキストをそろえるときの対象となるカラム範囲。

境界定義行は次のようなものであり<, >の位置がそれぞれカラム境界範囲の左端, 右端を示して

いる。表示行の<, >の位置を変えることでカラム境界範囲を変更できる。

=BNDS> <

(2) マスク定義行

行コマンドMASKはマスク定義行を表示させるものである。EDITは挿入行を表示するときマスク定義行をコピーする。あらかじめ、このカラム位置にはこの文字列を入れておきたいという場合に便利である。次に例を示す。

```
=MASK>
001200          LR          8,15
.....
-
/*
*/
```

(3) タブ定義行

行コマンドTABS (省略形: TAB)はタブ位置を表示, 変更するために用いられる。タブ機能については5.1.7に説明がある。

(4) カラム表示行

行コマンドCOLS (省略形: COL)はカラム目盛りを表示させるためのものである。次に表示例を示す。

```
=COLS> ----+-----1-----+-----2-----+-----3-----+-----4-----+-----5-----+-----6-----+-----7---
```

5.4 編集機能

各行での文字単位の編集(文字の入力, 削除など)は主にディスプレイ装置のもつ文字編集機能を利用するためEDITが介入する余地は少ない。他方, 行を単位とする編集(行の移動, 削除など)はEDITがもっているさまざまな行コマンドによって行われる。

5.4.1 行の編集

ここではディスプレイ装置の文字編集機能を使わずに単一行内で編集を行う機能としてCHANGE Eコマンドおよびシフト行コマンドについて説明する。

(1) CHANGE コマンド

{ CHANGE } { CHG } { C }	文字列 1    文字列 2	[ NEXT ]	[ CHARS ]	[ X ] [ カラム 1 [ カラム 2 ] ] [ NX ]
		ALL	{ PREFIX }	
		FIRST	{ PRE }	
		LAST	{ SUFFIX }	
		[ PREV ]	{ SUF }	
			WORD	

CHANGE コマンドは指定の文字列 1 を見つけそれを文字列 2 に置き換える。この機能は 2 つの機能から成り立っている。第 1 は文字列 1 を見つける FIND 機能であり第 2 はみつかった文字列 1 を文字列 2 に置き換える CHANGE 機能である。文字列 2 を除いた他のオペランドは FIND コマンドのオペランドと同じであり FIND 機能に用いられる。オペランドの意味および指定法は 5.2 の FIND

コマンドの項を参照されたい。文字列 2 の指定法は文字列 1 と同様であるが、テキスト文字列、ピクチャ文字列は使えない。すなわち文字列、引用符つき文字列、16進文字列、\*のいずれかである。

PFキーによってCHANGEコマンドの2つの機能を分離させることができる。文字列を直ちに置き換えるのではなく、まず見つけておいて置き換えたいときにだけ置き換えを実行するわけである。それにはまずCHANGEコマンドをコマンド入力フィールドに入力したあと、ENTERキーを押さずに、かわりにFINDキー(PF5)を押す。文字列が見つかったとカーソルはその文字列の先頭へ置かれる。もし文字列を置き換えたいならCHANGEキー(PF6)を押す。もし置き換えたくないなら再びFINDキーを押して次の文字列を探す。これを繰り返すと置き換える文字列を選択しながらCHANGEコマンドを実行できる。CHANGEコマンドのALL指定では思いがけない文字列をも置き換えてしまう可能性があるが、この方法なら確認しながら置き換えるので、その心配がない。

(2) カラムシフト行コマンド

カラムシフト行コマンドはカラム境界内の文字全体を右または左に平行移動させる。境界の範囲をこえた文字は自動的に削除されるため使用する場合は十分な注意が必要である。左カラムシフトには(行コマンド、右カラムシフトには)行コマンドを用いる。これらの違いはシフトの方向だけであるので(行コマンドのみの説明を行う。

(行コマンドには拡張形として(n, (, ((nがある。(, ((はそれぞれ(2, ((2の意味である。(n, ((nのnはシフトするカラム数を示している。(nは単一行指定, ((nは行範囲指定に用いる。すなわちシフトを行いたい先頭行と最終行のそれぞれに((nを入力することでその間の行をすべてnカラム左にシフトされる。

(3) データシフト行コマンド

データシフト行コマンドは行のデータ部を左右に平行移動する。データ部は各行を次の様式によって解釈したデータ部の文字列を意味する。

ラベル部□ ..... □ データ部□□ ..... □ コメント部

空白を含まない 連続する空白を含まない

左シフトは<行コマンド、右シフトは>行コマンドによる。カラムシフトと同様に<行コマンドの拡張形として<n, <<, <<nが使える。>行コマンドについても同様である。

データシフトの場合はカラムシフトと異なり移動するデータ部が消失することはない。これ以上シフトするとデータ部とラベル部がつながってしまうといった時点でシフトは中止される。

5.4.2 行単位の編集

ここでは行の挿入、削除、移動、複写を行う行コマンドについて述べる。これらの機能については一部拡張されているもののFSOの場合とはほぼ同じである。

(1) 行の挿入

新しい行の追加はI行コマンド(拡張形: In)を用いる。I行コマンドを入力すると指定行の直後に挿入行が表示される。挿入行の行番号フィールドは6個の引用符が表示されている。挿入行

はIのときは1行、Inのときはn行表示される。ENTERキーを押した際挿入行への入力がない場合その行は削除される。またその際カーソルの位置が最後の挿入行にあったときその直後に新しい挿入行が表示される。この機能を利用すると複数行の入力を行いたい場合でもIの指定だけでよいことになる。すなわち1行の入力が終わったらENTERキーを押すと次の行に挿入行が現われカーソルもその行に移るので続けて入力を行うことができる。入力が終わったら入力しないままENTERキーをもう一度押して挿入行を消してしまえばよい。

挿入行の行番号は入力があった時点で決定される。もし以前の行番号が連続していて行番号が空いていないときは自動的に次の行の番号をずらして空きをつくるため、FSOのときのようにリナンバを行ったりの手間がいらぬ。

I行コマンドの例：

```

I01000          DC      XL1'07',CL7'NXTCODE'
-
001100 START    SVG     14,12,=A(SAVE)
001200          LR      8,15
                ↓↓
                ENTERキー

001000          DC      XL1'07',CL7'NXTCODE'
.....
001100 START    -
                SVG     14,12,=A(SAVE)
001200          LR      8,15
                ↓↓
                入力後ENTERキー

001000          DC      XL1'07',CL7'NXTCODE'
001010 *----- SAVE -----*
.....
-
001100 START    SVG     14,12,=A(SAVE)
001200          LR      8,15
    
```

(2) 行の削除

D行コマンド(拡張形：Dn, DD-DD)は行の削除を行なう。Dは単一行、Dnは指定行も含めたn行、DD-DDは2つのDD行の間のすべての行を削除する。

D行コマンドの例：

000001		000001
D00010 A	→	000020 B
000020 B	ENTER キー	000050 E
D20030 C		000090 I
000040 D		000091 J
000050 E		000092 K
DD0060 F		
000070 G		
DD0080 H		
000090 I		
000091 J		
000092 K		

(3) 行の移動

行の移動はM行コマンド（拡張形：Mn，MM—MM）で移動元を，そしてA行コマンド（拡張形：An）もしくはB行コマンド（拡張形：Bn）によって移動先を指定して行なう。A行コマンドで指定された場合は指定行の直後（After），B行コマンドで指定された場合は直前（Before）へM行コマンドで指定した単一または複数行が挿入される。移動元の行は削除される。An，Bnを指定した場合は挿入はn個のコピーを作って挿入されることになる。たとえば，M2とA3の組み合わせの指定ならば全部で6行（2行を3組コピー）が挿入される。

M行コマンドの例：

(a) MとAの組み合わせ

M00010 A	→	000020 B
000020 B	ENTER キー	000030 C
A00030 C		000031 A
000040 D		000040 D

(b) MnとBの組み合わせ

M20010 A	→	000030 C
000020 B	ENTER キー	000040 D
000030 C		000041 A
000040 D		000042 B
B00050 E		000050 E
000060 F		000060 F

(c) MM—MMとBの組み合わせ

000010 A	→	000010 A
B00020 B	ENTER キー	000011 D
000030 C		000012 E
MM0040 D		000013 F
000050 E		000020 B
MM0060 F		000030 C
000070 G		000070 G

(4) 行の複写，繰り返し

行の複写はC行コマンド（拡張形：Cn，CC）にA行コマンド（拡張形：An）もしくはB行コマ

ンド(拡張形: B<sub>n</sub>)を組み合わせる。O行コマンドで複写元を、そしてAまたはB行コマンドで複写先を指定する。行の複写は行の挿入のあと、複写元の行が削除されない点を除けば行の移動と同じ機能である。

O行コマンドの例:

C00010 A	→	000010 A
A00020 B	→	000020 B
ENTERキー		000021 A
000030 C		000030 C

行の繰り返し(Repeat)は一種の行複写である。繰り返しの場合は、複写元の直後が複写先ということになる。行の繰り返しはR行コマンド(拡張形: R<sub>n</sub>, RR-RR, RR<sub>n</sub>-RR<sub>n</sub>)で指定する。Rは指定行を、RR-RRは指定された範囲内のすべての行をRepeatする。R<sub>n</sub>もしくはRR<sub>n</sub>-RR<sub>n</sub>はn個のコピーを挿入することを指定する。これはC行コマンドによる行複写の際のA<sub>n</sub>, B<sub>n</sub>指定に相当する。なおRR<sub>n</sub>-RR<sub>n</sub>は一方のnを省略してRR<sub>n</sub>-RRまたはRR-RR<sub>n</sub>と指定してもよい。RR<sub>n</sub>-RR<sub>n</sub>の指定の場合は両方のnは同じ値でなければならない。

R行コマンドの例:

(a) R

R00010 A	→	000010 A
ENTERキー		000011 A
000020 B		000020 B

(b) R<sub>n</sub>

R20010 A	→	000010 A
ENTERキー		000011 A
		000012 A
		000020 B

(c) RR-RR

RR0010 A	→	000010 A
ENTERキー		000020 B
RR0030 C		000030 C
000040 D		000031 A
		000032 B
		000033 C
		000040 D

(d) RR<sub>n</sub>-RR<sub>n</sub>

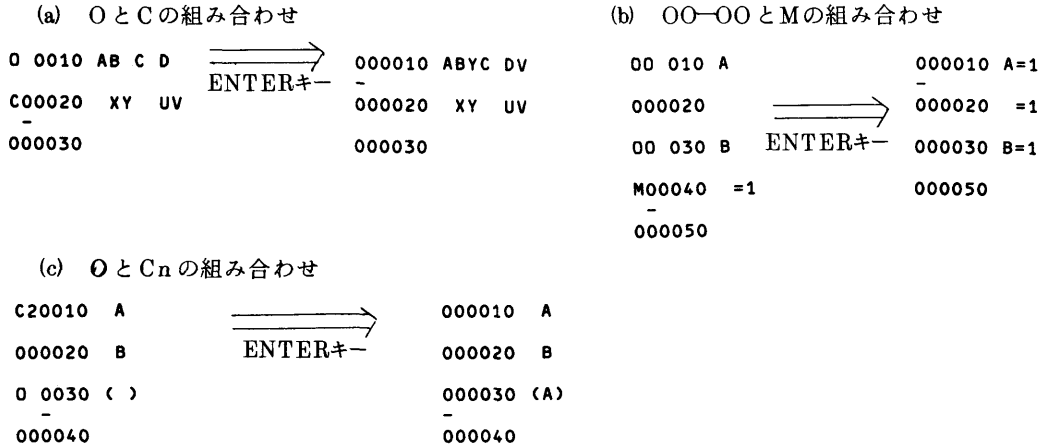
RR2010 A	→	000010 A
ENTERキー		000020 B
		000021 A
		000022 B
		000023 A
		000024 B
		000030 C

### 5.4.3 行の重ね合わせ

行の重ね合わせ(Overlay)はO行コマンド(拡張形: O<sub>n</sub>, OO-OO)と、C行コマンドもしくはM行コマンドを組み合わせる。O行コマンドで重ねられる側の行を指定し、CまたはM行コマンドで重ねる側の行を指定する。M行コマンドによる指定の場合は重ね合わせた後、重ねる側の行は削除される。重ねられる側と重ねる側の行の対応は、それぞれの先頭行から順に行われ、重ねられる側の最後の行が対応づけられた時点で対応づけは終わる。重ねる側の行数が少ないときは最後の行の対応のあと再び先頭から対応づけられる。重ねる側の行数が多いときは重ねられる側の行数分だけ

け対応がすむと残りの行は対応づけられないことになる。対応づけられた行と行の重ね合わせは次のように行われる。行と行の重ね合わせは同じカラム位置の文字どおしで行われる。重ねられる側の文字が空白以外の文字の場合は重ね合わせは行われず、それ以外の場合は重ねる行の同じカラム位置の文字が代入される。

○行コマンドの例：



### 5.5 内容の転送

EDITはデータセットの内容をすべて記憶域に読み込んで編集を行う。この編集内容と外部のデータセット間でデータの転送を行うコマンドが5つある。

- (a) 編集内容を保存するコマンド：SAVE
  - (b) 編集内容をデータセットに転送するコマンド：CREATE, REPLACE
  - (c) データセットの内容を記憶域に読み込むコマンド：COPY, MOVE
- これらのコマンドについて、以下順に説明する。

#### (1) SAVEコマンド

SAVE	
------	--

SAVEコマンドは現在記憶域にある編集内容をデータセットに保存する。そのほかEDITがENDキー(PF3)またはRETURNキーによって正常終了するときにもデータセットへの保存がなされる。

#### (2) CREATEコマンド

{ CREATE }	[メンバ名]
{ CRE }	

CREATEコマンドは区分データセットの新規メンバとして編集内容を転送する。転送される行はCまたはM行コマンドによって指定しなければならない。M行コマンドによる指定の場合はデータ転送後、指定の行は削除される。

CREATE コマンドのオペランドとしてメンバ名を指定すると現在編集中の区分データセットに新規メンバが作成される。オペランドを省略するとCREATEメニューが表示される。その場合、CまたはM行コマンドによる転送行の指定はコマンド以前に行わなければならない。CREATEメニューによって区分データセット名およびメンバ名を指定した後ENTERキーを押すと新規メンバが作成される。

### (3) REPLACE コマンド

$\left. \begin{array}{l} \text{REPLACE} \\ \text{REPL} \\ \text{REP} \end{array} \right\}$	[メンバ名]
--	--------

REPLACE コマンドは指定された行からなるデータによって区分データセットのメンバの置き換え、追加、あるいは順データセットの内容の置き換えを行う。コマンドの使い方の要領はCREATEコマンドと同様である。すでに存在するメンバを破壊しないための用心のためにCREATEコマンドを使う場合以外はCREATEコマンドの代わりにREPLACEコマンドの方が便利である。新しいメンバ名なら新規作成になり、すでにあるメンバ名ならメンバの置き換えになる。

### (4) COPY コマンド

COPY	[メンバ名]
------	--------

COPY コマンドは外部のデータセットまたはメンバの内容のすべてあるいは一部を編集中の内容へ組み込む。組み込む位置はAまたはB行コマンドで指定する。

COPY コマンドのオペランドとしてメンバ名を指定すると現在編集中の区分データセットの指定のメンバ全体が組み込まれる。オペランドを省略するとCOPYメニューが表示される。このメニューでデータセット名、メンバ名、転送される部分の開始行番号、最終行番号、行番号形式が指定できる。

### (5) MOVE コマンド

MOVE	[メンバ名]
------	--------

MOVE コマンドも外部データを編集中の内容へ組み込むものである。MOVE コマンドの場合は内容を組み込んだ後、もとのデータセットあるいはメンバが削除される点を除けばCOPYコマンドと同じである。MOVE コマンドはデータセットを削除する機能をもっているので使う場合は充分用心する必要がある。

## 5.6 行の表示

EDITは行の表示、非表示を切り換える機能をもつ。行を非表示にする行コマンドとしてX行コマンド、非表示行を表示させるための行コマンドとしてF、L、Sの3つの行コマンドがある。

X行コマンド(拡張形: X<sub>n</sub>, XX)は行を非表示にするための行コマンドである。非表示行の存在は次のように示される。

```
001600 NXTCODE CSECT
```

```
- - - - - 11 LINE(S) NOT DISPLAYED
```

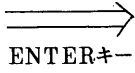
```
002800          LH          3,0(2)          /* GET CODE */
```

X行コマンドを用いると離れた2行を同一画面に表示することができる。たとえばPL/Iのプログラムの対応するDO-ENDを同じカラム位置にそろえたい場合その間の行を非表示にすると楽に作業ができる。またFINDコマンド、CHANGEコマンドのX、NXオペランドによって一部の行を文字列探索の範囲から除外する場合にX行コマンドが利用できる。

非表示の行を表示させるためにはF、L、S行コマンドを用いるとよい。RESETコマンドによる非表示の行をすべて同時に表示させることができる。

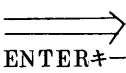
F行コマンド(拡張形:Fn)は非表示の行のうち先頭の行を表示させるものである。Fnを指定すると先頭のn行が表示される。FはF1の意味である。L行コマンド(拡張形:Ln)は逆に非表示の行のうち最終の行を表示させるものである。

F行コマンドの例:

<pre>001600 NXTCODE CSECT F - - - - - 002800          LH          3,</pre>		<pre>001600 NXTCODE CSECT 001700 *----- - - - - - 002800          LH          3,0</pre>
--	---	---

S行コマンド(拡張形:Sn)も非表示行を表示させるための行コマンドである。Snを指定すると非表示行の中でインデントーション(字下げ)が浅い順にn行表示される。もし同じ字下げ量なら行番号の小さい順に表示される。SはS1の意味である。

S行コマンドの例:

<pre>000010 DO; S4- - - - - 000300 END;</pre>		<pre>000010 DO; 000020     MODE(1)=TIME; 000030     DO P=1 TO M; - - - - - 000100     END; 000200     CALL K1(I,J); 000300 END;</pre>
---	---	---

### 5.7 テキスト

テキストとは、たとえば英語の文章のように単語が空白を区切りとして並んでいるデータをいう。テキストの入力のためにTE(Text Entry)行コマンド、テキストの並べかえのためにTF(Text Flow)行コマンド、そしてテキスト行の分割のためにTS(Text Split)行コマンドがある。

TE行コマンド(拡張形:TEn)は画面の大きさを気にせず連続してテキストデータを入力可能とするものである。TEnは画面上にn行分の、そしてTEは画面の最終行までの空白領域を確保することを指定する。領域の大きさを越えて入力を行おうとするとキーボードがロックされRESETキーで解除しないかぎり入力できない状態になるため安心してテキスト入力に専念することができる。テキスト入力後ENTERキーを押すと次の編集が行われる。

- (a) カラム境界範囲に収まるようにする。
- (b) その際2行にまたがる単語や行の右端で終わる単語は次の行に移す。ただし、次の行の先頭が3文字以上の空白で始まるときは新しい行を生成する。
- (c) 2文字以上の字下げは字下げをなくす。

TF行コマンド(拡張形: TF $n$ )は、右端が不揃いなテキストをきれいにそろえたり、テキストの右端の位置を変更するのに用いられる。処理の対象となる行はTF行コマンドが入力された行から同じ字下げ量が続く間の行である。TFを指定したときはカラム境界の右端を、TF $n$ を指定したときは $n$ カラム目を右境界とする次のようなフォーマットが行われる。

- (a) 右境界を越える単語は次の行に移す。
- (b) 右の方に空白があるときは次の行から単語をもってくる。ただし、単語を移動させた結果右境界までの間にはいりきれないときは移動を行わない。
- (c) テキストの中間の空白は除かない。
- (d) 文の終わり(コロン, ピリオド, 疑問符, 感嘆符が現われたとき)には空白を2個置く。TS行コマンド(拡張形: TS $n$ )は1行を2行に分割するために用いる。TS行コマンドは単に1行を分割する機能だけでなく、分ける部分に新しいテキストが挿入されるものとして処理している。TS行コマンドは次のように指定する。TS行コマンドを入力した行の分割したいカラム位置にカーソルを置いてENTERキーを押す。行コマンドを入力した行はもとの1行の左側部分だけになり、その次に挿入行が表示され、さらにその次の行に分割された右側部分だけの行が表示される。挿入行はTS指定のときは1行、TS $n$ 指定のときは $n$ 行表示される。

## 5.8 その他

これまでに説明しなかったコマンドとしてSUBMIT, RESETコマンドの説明を行う。

### (1) SUBMITコマンド

$\left\{ \begin{array}{l} \text{SUBMIT} \\ \text{SUB} \end{array} \right\}$	
---	--

SUBMITコマンドは編集集中の内容をFIBジョブとして起動する。

### (2) RESETコマンド

$\left\{ \begin{array}{l} \text{RESET} \\ \text{RES} \end{array} \right\}$	
--	--

RESETコマンドは次の機能をもつ。

- (a) X行コマンドで非表示にした行をすべて表示させる。
- (b) COLS, MASK, TABS, BOUNDSの各行コマンドおよびPROFILEコマンドによる表示行をすべて削除する。
- (c) 行番号フィールドの== CHG >, == ERR >のメッセージを除去する。
- (d) 誤りの行コマンド, 対の完成していない行コマンドをCANCELする。

- (e) 初期設定の際のモード変更の警告メッセージを削除する。

RESET コマンドを実行させると、通常の行番号をもった行のみが表示される状態になる。

## 6. BROWSE オプション (オプション 1)

BROWSE はデータセットの内容を画面に表示するためのオプションである。表示機能だけに限れば EDIT オプションとはほぼ同じ使い方ができる。しかし BROWSE は表示専用としての特長をいくつかもっている。

- (1) データセットの内容をそのまま表示する。(たとえば行番号を行の先頭に表示するという事はない。)
- (2) EDIT と異なりデータセットの内容をすべて記憶域に読み込むことをやらないので大きなデータセットを表示できる。
- (3) 最大 32,760 バイトのレコード長をもったデータセットの表示ができる。(EDIT では 1 行の長さの最大値は 255 バイト)
- (4) 誤ってデータセットを破壊する心配がいらぬ。(たとえば UTILITY オプションの中から暗黙に呼ばれている場合、EDIT の画面ならば知らずに内容を書き換える恐れがある。)

### 6.1 開始, 終了

BROWSE の機能と EDIT の機能は似ているため開始, 終了の方法もほぼ同じである。プライマリオプションメニューが表示されているときオプション番号 1 を入力すると BROWSE データセットメニューが表示される。EDIT の場合と同様にしてデータセットを指定する。指定のデータセットが順編成またはメンバ名つきの区分編成の場合は直ちに BROWSE データ表示画面になる。区分データセットを指定し、メンバ名を略した場合はメンバメニューが表示される。ここでメンバ名を指定すると BROWSE データ表示画面になる。

BROWSE を終了するには END キー (PF3) を押す。画面はデータセットメニューもしくはメンバメニューになる。ここでデータセット, またはメンバを指定すると再びデータ表示画面になる。プライマリオプションメニューにもどるためには END キーを押すとよい。

```

BROWSE - F0052.ALIB.ASM(NXTCODE) ----- LINE 00000 COLS 001 080
COMMAND ==> - SCROLL ==> PAGE
***** TOP OF DATA *****-CAPS OFF-***
* SUBR          NXTCODE(CODE: CHAR(2))          00000100
* CODE ORDER : 80A1 ... 80FE 81A1 ... ... A0FE 41A1 ... 79FE 80A1 ... 00000200
* MODIFY: 81.10.22 ADD CHECK OF 79FF          00000300
*
*
NXTCODE CSECT          00000500
*----- PROLOGUE -----*          00000600
      USING      *,15          00000700
      B          START        00000800
      DC         XL1'07',CL7'NXTCODE'        00000900
*----- SAVE -----*          00001000
START  SVG       14,12,=A(SAVE)          00001100
      LR        8,15          00001200
      DROP     15          00001300
      USING    NXTCODE,8          00001400
*----- MAIN -----*          00001500
      L         2,0(GR1)          /* GET CODE ADDRESS */          00001600
      LH        3,0(2)          /* GET CODE */          00001700
      N         3,=XL4'0000FFFF' /* MASK OF UNDER 2-BYTES */          00001800
      A         3,=F'1'          /* ADD 1 TO THE CURRENT CODE */          00001900
      C         3,=XL4'0000A0FF' /* IF GR3=X'A0FF' THEN GR3=X'41A1'          00002000
      BE        XA0FF           : ELSE IF          00002100

```

図 6.1 BROWSE データ表示画面

## 6.2. 画面移動

BROWSEの画面移動はSCROLLキーによる場合のほかはLOCATEコマンド、FINDコマンドで行われる。SCROLLキーによる画面移動はEDITの場合と同じであるので、使い方については「5.2.1 SCROLLによる画面移動」を参照されたい。

### 6.2.1 LOCATE コマンド

{ LOCATE }	{ 相対行番号 }
{ LOC }	{ }
{ L }	{ ラベル }

指定された相対行番号またはラベルをもつ行が先頭の表示行になるように画面を移動する。

(1) 相対行番号

相対行番号はデータセットの先頭レコードを1とする番号である。TOP OF DATAのメッセージ行は相対行番号0として扱われる。

(2) ラベル

ラベルはピリオドに続く7個以下の英数字からなる文字列である。ラベルはあらかじめ特定の行に対応づけておかなければならない。ラベルと行の対応づけはコマンド入力フィールドにラベルを入力して行う。そのときの先頭の表示行が、入力したラベルと対応づけられる。なおLOCATEコマンドとしてラベルを指定するときピリオド-英字で始まる名前ときはピリオドを略してもかまわない。

6.2.2 FINDコマンド

{ FIND } { F }	文字列	[ NEXT ]	[ CHARS ]	[ カラム 1 [ カラム 2 ] ]
		ALL FIRST LAST PREV	{ PREFIX } { PRE } { SUFFIX } { SUF } WORD	

FINDコマンドに関してもオペランドとしてX, NXがないことを除けばEDITのFINDコマンドと同じである。オペランドの意味については「5.2.2 コマンドによる画面移動」を参照されたい。

FINDコマンドに関するモードを定めるコマンドとしてCAPSコマンドがある。

CAPS	[ ON ] [ OFF ]
------	-------------------

FINDコマンドの文字列オペランドの英小文字コードを英大文字に変換するかどうかを定める。現在のモードのON/OFF値はTOP OF DATA, BOTTOM OF DATAの表示行に表示されている。

6.3 16進数表示

HEX	[ ON ] [ VERT ] [ OFF ] [ DATA ]
-----	-------------------------------------

HEXコマンドに関しても内容の修正ができないこと以外はEDITのHEXコマンドと同じである。「5.1.8 HEXモード」を参照されたい。

6.4 カラム目盛りの表示

カラム目盛りを表示させるためにCOLSコマンド(省略形: COL), 表示をやめるためにRESE

Tコマンド(省略形: RES)が用意されている。それぞれオペランドなしのコマンドである。

## 7. UTILITYオプション(オプション3)

UTILITYオプションはデータセットやメンバの割り当て、削除、内容のプリント、SYSOUTの検索などの各種のユティリティ機能から成り立っている。ここではLIBRARYユティリティのメンバプリントの機能の例を挙げるにとどめる。その他の機能については、メニュー画面に表示される説明もしくはHELPキー(PF1)を押して得られるHELP画面によって各自調べていただきたい。

プライマリオプションメニュー(図3.1)でオプション3を入力するとUTILITY選択メニュー(図7.1)が表示される。

```

-----< UTILITY OPTION >-----
OPTION ==> 1
_

      1  LIBRARY   : LIBRARY UTILITIES   *1*
      2  DATASET  : DATASET UTILITIES   *2*
      3  MOVE/COPY : MOVE OR COPY MEMBERS OR DATASETS
      4  CATALOG  : DEFINE/DELETE CATALOG ALIAS OR DISPLAY/PRINT CATALOG
      5  RESET    : RESET STATISTICS FOR MEMBERS OF PFD LIBRARY DATASET
      6  HARDCOPY : PRINT OR PUNCH DATASET
      7  VTOC     : DISPLAY OR PRINT VTOC ENTRIES FOR A DASD VOLUME
      8  OUTLIST  : DISPLAY, DELETE, OR PRINT HELD JOB OUTPUT

-----

      *1* : COMPRESS PARTITIONED DATASET
           PRINT, DISPLAY, RENAME, OR DELETE MEMBERS
           PRINT INDEX LISTING OR ENTIRE DATASET

      *2* : DISPLAY DATASET INFORMATION
           ALLOCATE, RENAME, OR DELETE DATASET
           CATALOG OR UNCATALOG DATASET
-----

```

図7.1 UTILITY選択メニュー(LIBRARYユティリティを選択)

図7.1の画面の下の方の\*1\*の部分の説明より区分データセットのメンバのプリントを行うにはLIBRARYユティリティを選択すればよいことがわかる。そこでOPTION入力フィールドにLIBRARYユティリティのサブオプション番号1を入力する。図7.1はそのときの画面である。ENTERキーを押すとLIBRARYユティリティメニュー(図7.2)が表示される。

メンバプリントを行うオプションはPであるのでOPTION入力フィールドにPを入力した後データセット名とメンバを指定する。図7.2は▼F0052.ALIB.ASM▼という区分データセットの

NXTCODEというメンバを指定している。ここでENTERキーを押すとメンバの内容がプリントさ

```

-----< LIBRARY UTILITY >-----
OPTION ==> P

      C - COMPRESS DATASET                B - BROWSE MEMBER
      L - PRINT ENTIRE DATASET            D - DELETE MEMBER
      X - PRINT INDEX LISTING            R - RENAME MEMBER
      BLANK - DISPLAY MEMBER LIST        P - PRINT MEMBER

PFD LIBRARY DATASET:
PROJECT ==> F0052
LIBRARY ==> ALIB
TYPE    ==> ASM
MEMBER  ==> NXTCODE      (IF OPTION 'P', 'R', 'D', OR 'B' SELECTED)
NEWNAME ==>              (IF OPTION 'R' SELECTED)

OTHER PARTITIONED OR SEQUENTIAL DATASET:
DATASET NAME ==>
VOLUME SERIAL ==>      (IF NOT IN CATALOG)

DATASET PASSWORD ==>      (IF PASSWORD PROTECTED)

```

図7.2 LIBRARYユーティリティメニュー（メンバプリント指定）

れる。メンバプリントに限らずPFDでプリントをとるとするのは直ちにラインプリンタか何かに出力することを意味しない。プリントをとった時点ではPFDの定めるリストデータセットへ出力するという意味である。リストデータセットに出力された内容を実際にプリンタに出力するかどうか、出力する場合どのように出力するかはリストデータセットに対する後処理として指定する。

プリントが終わったらENDキー（PF3）で再びUTILITY選択メニューにもどる。ここでENDキーを押すとプライマリオプションメニューにもどる。ATTRIBUTESオプションのPFキー定義によってRETURNキーが定められているときはRETURNキーを押すとLIBRARYユーティリティメニューから直接プライマリオプションメニューへもどることができる。プライマリオプションメニューが表示されているときENDキーを押すと終了メニュー（図3.15）が表示される。このメニューのリストデータセットに対するPROCESS OPTIONフィールドの値の選択でリストデータセットの処置が定まる。ここではJを指定しラインプリンタに出力させることにする。出力結果の例が図7.3である。



付録1. PFDのオプション一覧

- 0 ATTRIBUTES (属性定義)
  - 0.1 TERMINAL (端末属性定義)
  - 0.2 LOG/LIST (ログ/リスト データセットのパラメタ定義)
  - 0.3 PF KEYS (PFキー定義)
- 1 BROWSE (データセットの表示, 検索)
- 2 EDIT (データセットの編集)
- 3 UTILITY (各種のユティリティ)
  - 3.1 LIBRARY (区分データセットのメンバに対するユティリティ)
    - B メンバ内容の表示, 検索
    - C 区分データセットの圧縮
    - D メンバの削除
    - L データセットの内容のプリント
    - P メンバの内容のプリント
    - R メンバ名の変更
    - X データセットの属性, メンバリスト, ディレクトリ情報のプリント
    - 空白 メンバメニューの表示
  - 3.2 DATASET (データセットに対するユティリティ)
    - A 新規データセットの割り当て
    - C データセットのカタログ
    - D データセットの削除
    - R データセット名の変更
    - 空白 データセットの属性の表示
  - 3.3 MOVE/COPY (データセットの移動, 複写)
    - C データセットの複写
    - CP データセットの複写およびプリント
    - M データセットの移動
    - MP データセットの移動およびプリント
  - 3.4 使用禁止
  - 3.5 RESET (更新履歴の設定, 変更, 削除)
    - D 更新履歴の削除
    - N ディレクトリ部に記録された更新履歴の設定, 変更
    - R 更新履歴の設定, 変更
  - 3.6 HARDCOPY (データセットの内容の出力)
    - J FIBジョブによるSYSOUT出力
    - L ローカルプリンタへの出力
  - 3.7 使用禁止

- 3.8 OUTLIST (SYSOUT に対するユティリティ)
  - D SYSOUT の削除
  - L ジョブの状態表示
  - P SYSOUT の出力および削除
  - R SYSOUT クラスの変更
  - 空白 SYSOUT の検索
- 4 FOREGROUND (メニュー画面によるコマンド呼び出し)
  - 4.1 ASM コマンドの呼び出し
  - 4.2 COBOL //
  - 4.3 FORT //
  - 4.4 FORT77 //
  - 4.5 PLI //
  - 4.6 LINK //
  - 4.7 TESTCOB //
  - 4.8 TESTFORT //
  - 4.9 TESTPLI //
- 5 BACKGROUND (メニュー画面による FIB ジョブの起動)
  - 5.1 アセンブラ
  - 5.2 COBOL
  - 5.3 FORTRAN-GE
  - 5.4 FORTRAN-77
  - 5.5 PL/I
  - 5.6 リンケージエディタ
- 6 TSS (TSS コマンドの実行)
- 7 HELP (HELP 画面の表示)
- 8 YUUHITSU (右筆による日本語データセットの編集)
- 9 GEM (GEM モジュールの登録, 更新)
- X EXIT (PFD の終了)

付録 2. PFDのコマンド一覧

1. EDITのコマンド

AUTONUM	[ON] [OFF]	内容保存時のリナ バ (5.1.2) P. 51
{ CANCEL } { CAN }		編集中止 (5.0) P. 49
CAPS	[ON] [OFF]	英小文字の変換 (5.1.5) P. 51
{ CHANGE } { CHG } { C }	{ 文字列 1 } { 文字列 2 } { * } { * }	文字列の置き換え (5.4.1) P. 59
	[NEXT] [CHARS] PREV { PREFIX } FIRST { PREF } LAST { SUFFIX } ALL { SUF } WORD	
	[X] [ カラム 1 [ カラム 2 ] [NX]	
COPY	[メンバ名]	データセットからの 複写 (5.5) P. 65
{ CREATE } { CRE }	[メンバ名]	データセット新規作 成 (5.5) P. 64
{ FIND } { F }	{ 文字列 } { * }	文字列の検索 (5.2.2) P. 55
	[NEXT] [CHARS] PREV { PREFIX } [X] FIRST { PREF } [NX] LAST { SUFFIX } ALL { SUF } WORD	[カラム 1 [カラム 2 ]]
HEX	[ON] [VERT] [OFF] [DATA]	16進数表示 (5.1.8) P. 53
{ LOCATE } { LOC } { L }	行番号	表示行の移動 (5.2.2) P. 55
MOVE	[メンバ名]	データセットからの 移動 (5.5) P. 65
NULLS	[ON] [ALL] [OFF]	行末のNULL文字表 示 (5.1.6) P. 52
{ NUMBER } { NUM }	[ON] [STD] { COBOL } [OFF] { COB }	行番号の有無 (5.1.1) P. 50

PRINT	[ON] [OFF]	内容保存時のプリント (5.1.4) P. 51
{ PROFILE }	[ プロフィール名 ] [ 行数 ]	プロフィール表示 (5.1.0) P. 49
{ PROF }		
{ PRO }		
{ RECOVERY }	[ON] [OFF]	リカバリ機能 (5.1.9) P. 54
{ REC }		
{ RENUM }	[ STD ] [ COBOL ]	行番号付け直し (5.1.1) P. 50
{ REN }	[ COB ]	
{ REPLACE }	[ メンバ名 ]	データセットの置き換え (5.5) P. 65
{ RELP }		
{ REP }		
{ RESET }		表示の初期化 (5.8) P. 67
{ RES }		
SAVE		保存 (5.5) P. 64
STATS	[ON] [OFF]	更新履歴 (5.1.3) P. 51
{ SUBMIT }		FIBジョブ起動 (5.8) P. 67
{ SUB }		
{ TABS }	[ON] [OFF] [ タブ文字 ] [ ALL ]	タブ機能 (5.3) P. 58
{ TAB }		
{ UNNUM }		行番号のクリア (5.1.1) P. 51
{ UNN }		

2. EDITの行コマンド

A, An	After	複写先, 移動先指定	(5.4.2) P. 60
B, Bn	Before	"	"
BOUNDS, BOUND, BNDS, BND		カラム境界定義行表示	(5.3) P. 58
C, Cn, CC-CC	Copy	複写元指定	(5.4.2) P. 62
COLS, COL		カラム表示先の表示	(5.3) P. 59
D, Dn, DD-DD	Delete	行の削除	(5.4.2) P. 61
F, Fn	First	最初の非表示行の表示	(5.6) P. 66
I, In	Insert	行の挿入	(5.4.2) P. 60
L, Ln	Last	最後の非表示行の表示	(5.6) P. 66
M, Mn, MM-MM	Move	移動元指定	(5.4.2) P. 62
MASK		マスク定義行の表示	(5.3) P. 59
O, On, OO-OO	Overlay	重ねられる行の指定	(5.4.3) P. 63

解 説

R, Rn, RR-RR, RRn-RRn	Repeat	行の繰り返し	(5.4.2) P. 63
S, Sn	Show	非表示行の表示	(5.6) P. 66
TABS, TAB		タブ定義行の表示	(5.3) P. 59
TE, TEn	Text Entry	テキスト入力	(5.7) P. 66
TF, TFn	Text Flow	テキストの並べかえ	"
TS, TSn	Text Split	テキスト分離	"
X, Xn, XX-XX	eXclude	行を非表示にする	(5.6) P. 65
<, <n, <<-<<, <<n-<<n	data shift left	データの左シフト	(5.4.1) P. 60
>, >n, >>->>, >>n->>n	data shift right	データの右シフト	"
(, (n, ((-((, ((n-((n	column shift left	行の左シフト	"
), )n, ))-)), ))n-))n	column shift right	行の右シフト	"

3. メンバメニューのコマンド

{ LOCATE LOC L }	メンバ名	画面移動	(3) P. 27
{ SELECT SEL S }	メンバ名	メンバ選択	"

4. BROWSEのコマンド

CAPS	[ ON ] [ OFF ]	英小文字→英大文字 (6.2.2) P. 70
COLS		カラム目盛り表示 (6.4) P. 70
FIND	{ 文字列 } * [ NEXT ] PREV FIRST LAST [ ALL ]	{ CHARS } { PREFIX } PRE { SUFFIX } SUF WORD
HEX	[ ON ] [ VERT ] [ OFF ] [ DATA ]	16進数表示 (6.3) P. 70
{ LOCATE LOC L }	{ 相対行番号 } { ラベル }	画面移動 (6.2.1) P. 69
. 英数字列		ラベル定義 (6.2.1) P. 70

付録3. PFキーの標準定義





PA1 アテンション	PA2 再表示	PF1/PF13 HELP	PF2/PF14 画面分割	PF3/PF15 終了
INSERT MODE	DELETE	PF4/PF16 PRINT	PF5/PF17 FIND続行	PF6/PF18 CHANGE続行
		PF7/PF19 画面移動 (↑)	PF8/PF20 画面移動 (↓)	PF9/PF21 論理画面の 切り換え
		PF10/PF22 画面移動 (←)	PF11/PF23 画面移動 (→)	PF12/PF24 カーソル移動



付録4. ディスプレイ装置のキーの機能

ディスプレイ装置の画面のそれぞれの位置には普通の文字(コードX'00'のNULL文字を含む)またはアトリビュート文字が置かれている。行の右端の文字の次には次行の左端の文字が続いているものと考えて、画面上のすべての文字は1つにつながっているものとみなされる。あるアトリビュート文字から次のアトリビュート文字までの間をフィールドと呼ぶ。フィールドの属性として文字入力可能であるかどうか、輝度は通常輝度であるか高輝度であるか等があるが、それはフィールド前のアトリビュート文字によって規定される。ここではいくつかのキーの機能について簡単に説明する。


(1) カーソル制御キー

(a) カーソル移動キー


    カーソルを行または文字単位で指定の方向へ移動させる。


  カーソルを2文字を単位として右または左へ移動させる。

(b) フィールド指向キー

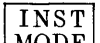
 (Tab キー) カーソルを次の入力フィールド(内の第1文字の位置)へ移動させる。

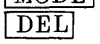
 (Back Tab キー) カーソルを直前の入力フィールドの第1文字の位置へ移動させる。

 (New Line キー) カーソルを次行以下の最初の入力フィールドへ移動させる。


 (Home キー) カーソルを画面の最初の入力フィールドへ移動させる。

(2) 編集キー

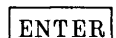
 (Insert Mode キー) キーボードを挿入モードにする。Reset キーで解除できる。



 (Delete キー) 入力フィールド内の1文字を削除する。カーソル以降のフィールド内の文字は1文字ずつ左へシフトされ、フィールドの最後にはNULL文字が挿入される。



(3) 消去キー

 (Erase to End of Field キー) 入力フィールド内の現在のカーソル位置以降のすべての文字をNULL文字にする。

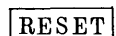
(4) プログラムアテンションキー


 (Enter キー) キーボードからのメッセージ入力の終了を示す。

 ~  (Program Function キー) プログラムで自由に機能を定義して使用するためのキー。

 ~  (Program Access キー) プログラムで機能を定めることができる。普通PA1はアテンションキーに、PA2は再表示指定キーに用いる。

(5) その他

 (Reset キー) 挿入モード、エラー状態の解除に用いる。

 (Alternate Function キー) ファンクションキーのためのシフトキー。