

文字表示装置を使用する簡単な ディスク・エディタの作成

渡部 哲夫*・石原 好宏*

Construction of a Simplified Disk-editor manipulated
through a Character Display Unit

Tetsuo WATANABE* and Yoshihiro ISHIHARA*

Abstract

This paper gives a description of a simplified disk-editor which is to be manipulated through a character display unit. This disk-editor is designed for the editing of files on the disk attached to the computer system MELCOM 9100/30F which is installed in our college. The editing operation is intended to be performed on the conversational basis. The editor is written in ASSEMBLY language and FORTRAN language complementarily, and it is operated in two different modes, READY mode and EDIT mode. Each mode includes ten commands. Some examples of usage of the editor using these commands are also presented.

1. ま え が き

本学には、電子計算機システム MELCOM 9100/30 F が設置され、これに文字表示装置 M 345 が接続されている。しかし、この文字表示装置は、MELCOM 9100/30 F のオペレーティング・システムに強くは管理されておらず、極く基本的な回線制御プログラムしか備えられていない¹⁾。そのため、文字表示装置の使用目的に応じた支援ソフトウェアが必要となる。我々は、以前、文字表示装置の持つ機能のうち、会話型処理、データの入出力、ファイルの編集等を行うための支援ソフトウェアを作成した²⁾。一方、我々は、記号処理言語 LISP を処理する LISP システムを本学の MELCOM 9100/30 F 上に作成したいと考えている。LISP システムを作成する場合、会話型のディスク・エディタを持った電子計算機システムの環境が望まれる。そのためには以前に作成した支援ソフトウェアの“ファイルの編集”を会話的に行えるように改良することが一つの方法であろう。このような考え方で我々は今度、簡単なディスク・エディタを作成した。ここにその概要を報告する。

2. ディスク・エディタの特徴

2.1 作成の目的

本エディタは、主として、記号処理言語 LISP を MELCOM 9100/30 F 上で会話的に処理するための一つの道具として作成した。すなわち LISP システムの下で評価される EXPR 関数や FEXPR 関数の定義および評価したいダブレットをディスクを介して LISP システムに与えるための準備をする道具である。しかし、同時に本エディタは、FORTRAN 言語あるいは ASSEMBLY 言語のソースプログラムを登録するファイルやデータファイルをカードレス方式で会話的に編集、デバックすることもできる。又、このエディタで編集されたデータファイルは、FORTRAN 言語等の READ 文によって簡単に読み取ることができる。

2.2 編集機能の特徴

以前作成した支援ソフトウェアでは¹⁾、1 ボリューム=1 順編成ファイルとし 6000 記録 (1 記録当たり 80 バイト) までの編集作業が行える。又、編集作業命令とそれに付随するデータを一括して入力し処理する方式である。これは、大量の純粋なデータファイルを編集する場合は十分であると考えられる。しかし、今

* 工業短期大学部 情報処理工学科

度のように LISP や FORTRAN などのソースプログラム・ファイルを編集する場合には、複数のファイルを対象とした高速で、会話型の編集ができることが望まれる。そこで、本エディタでは、1 ボリュームに複数の直接編成ファイルが格納でき、会話的に入力される編集作業命令とそれに付随するデータによって、その都度編集処理できるようにした。このことによって、マスターボリュームにソースプログラムファイルやデータファイルを登録しておくこと、また、使用者の要求に応じてその中の任意のファイルを編集作業用ボリュームに転送してその内容を編集したり他のプログラムシステムから参照すること、などができるようになった。更に、文字表示装置のそばに設置してあるコンソールタイプライタを文字表示装置から見たハードコピー装置とみなし、編集作業の一連の記録を用紙上にも残せるように工夫した。

3. ディスク・エディタの概要

3.1 編集機能の概要

本エディタは2段階構造をもっている。第1段階は処理の対象がマスターボリュームであり、そこに登録された個々のファイルを対象として編集作業を行う命

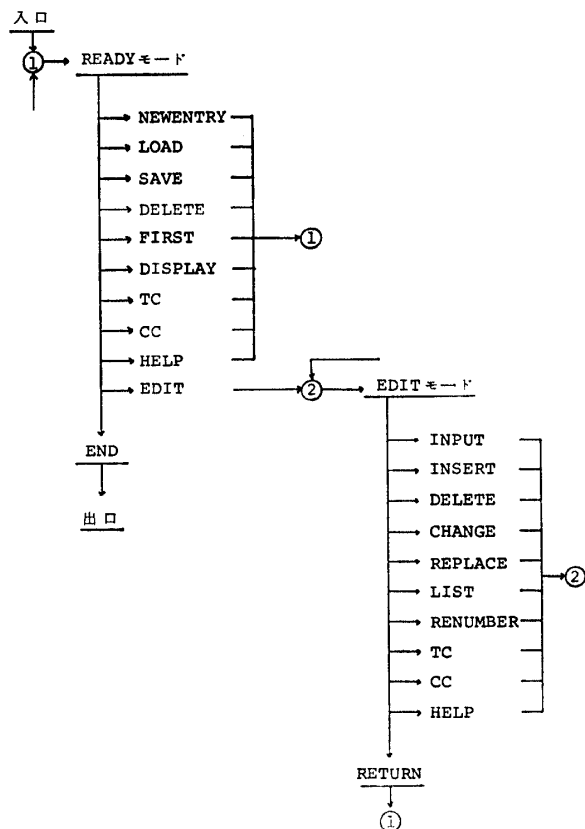


Fig. 1 Editing commands which constitute the editor and the interrelation among these commands.

令で構成されている。これを READY モードと呼ぶ。一方、第2段階は処理の対象が編集作業用ボリュームであり、そこに登録された被編集ファイルの記録を対象として編集作業を行う命令で構成されている。これを EDIT モードと呼ぶ。

各モードの内部では、オペレータは任意の編集作業を行うことができる。又、READY モードと EDIT モードの間も必要に応じて任意に往復することができる。こうしてファイルの編集作業は効率よく進められる。Fig. 1 に本エディタを構成する編集作業命令とそれらの相互関係を示す。次節以降で各モードにおける編集作業命令の書式と機能について説明する。

3.1.1 READY モード

READY モードの編集作業命令には、ファイルを対象として、ファイル名宣言、転送、登録、削除を行う命令、及び幾つかの補助命令がある。なお、補助命令はファイルの内容の変更は伴わない。

(1) ファイル名宣言

▶ *1 NEWENTRY *2, ファイル名*3.

下線部省略形 [NE]

機能：編集作業用ボリュームに新しいファイルを作成することを宣言し、記録制御表の内容を初期状態にする。

(2) 転送

▶ LOAD, ファイル名.

下線部省略形 [LO]

機能：マスターボリュームに登録済のファイルを編集作業用ボリュームに転送する。

(3) 登録

▶ SAVE, ファイル名.

下線部省略形 [SA]

機能：編集作業用ボリュームのファイルをマスターボリュームに登録する。

(4) 削除

▶ DELETE, ファイル名.

下線部省略形 [DE]

機能：マスターボリュームから指定したファイルを削除する。

*1 ▶は SOE マークであって、これはオペレータに対する入力要求を意味する。

*2 下線を施した部分は、下線部省略形に示した内容で置き換えることができる。

*3 6文字以内の英数字。但し先頭文字は英字。

(5) 補助命令

(5-1) 初期設定

▶ FIRST. 下線部省略形 [FI]
機能：マスターボリュームを初期状態にもどす。

(5-2) 表示

▶ DISPLAY. 下線部省略形 [DI]
機能：マスターボリュームに登録済のファイル名をすべて表示する。

(5-3) 表示装置の選択

▶ TC. 又は ▶ CC.
機能：TC の場合タイプライタと文字表示装置の両方に表示する。
CC の場合文字表示装置だけに表示する。
(ハードコピーの要・不要を宣言する)

(5-4) 救援

▶ HELP. 下線部省略形 [HE]
機能：READY モードにおける編集作業命令の入力形式を表示する。

(5-5) 編集モードの切換え

▶ EDIT. 下線部省略形 [ED]
機能：EDIT モードへの遷移を指示する。

(5-6) 終了

▶ END. 下線部省略形 [EN]
機能：本エディタの終了を指示する。

本エディタは END 命令で終了するが、EDIT モードで編集されたファイルは必ずしも物理的にも順序正しく並んでいるわけではない。これは、3.2 で述べるように、主記憶上のポインタの変更だけで作業を進めているためである。そこで、END 命令が入力されると、作業用ボリュームを利用して編集作業用ボリューム内のファイルの記録を物理的にも順序正しく並べかえる。この作業が終わった時点で、すべての編集作業が終ることになる。

3.1.2 EDIT モード

EDIT モードの編集作業命令には、被編集ファイル内の記録を対象として、追加、挿入、削除、一部修正、全体修正、及び幾つかの補助命令がある。なお、被編集ファイルは、マスターボリュームから編集作業用ボリュームに転送される場合と、新たに編集作業用ボ

リューム上に作られる場合とがある。

(1) 追加

▶ INPUT, 開始行番号*1—間隔数*1.
下線部省略形 [INPU,]
開始行番号は本エディタが自動的に決定する。間隔数は10である。
(INPU, 開始行番号)
間隔数は10である。
機能：被編集ファイルに新しい記録を追加する。既に同一番号の記録がファイル内に存在する時は、新しい記録内容で古い記録を置き換える。
注意：INPUT 処理の中止は、“▶] (XMIT) 押下”で行う。

(2) 挿入

▶ INSERT, 開始行番号—間隔数.
下線部省略形 [INSE, 開始行番号—間隔数]
(INSE, 開始行番号) 1行だけの挿入
機能：被編集ファイルの記録と記録の間に新しく記録を挿入する。
注意：開始行番号に間隔数を加えた行番号と同一の行番号をもつ記録が被編集ファイル中に存在する時、あるいは“▶] (XMIT) 押下”によって INSERT 処理は終了し、EDIT モードにもどる。

(3) 削除

▶ DELETE, 開始行番号—終了行番号.
下線部省略形 [DELE]
すべての記録を削除する。
(DELE, 行番号)
指定した行番号の記録を削除する。
(DELE, 開始行番号—終了行番号)
指定された範囲の記録を削除する。
機能：被編集ファイルの記録を一部（あるいは全部）削除する。

(4) 一部修正

▶ CHANGE, 行番号.
下線部省略形 [CHAN]
機能：指定した行番号の記録の一部分だけを変更する。

*1 5ケタ以内の数値

(5) 全体修正

▶ REPLACE, 行番号.
 下線部省略形 [REPL]
 機能: 指定した行番号の記録を新しい記録で置き換える.

(6) 補助命令

(6-1) 表示

▶ LIST, 開始行番号—最終行番号.
 下線部省略形
[LIST]
 すべての記録を表示する.
[LIST, 開始行番号]
 1 記録だけ表示する.
 機能: 被編集ファイルの内容をこの命令で指定した範囲で行番号とともに表示する.

(6-2) 行番号付けの変更

▶ RENUMBER, 開始行番号—終了行番号—間隔数.
 下線部省略形
[RENU, 開始行番号—終了行番号—間隔数]
[RENU, 開始行番号—終了行番号]
 間隔数は10になる.
[RENU, 開始行番号]
 間隔数を10, 先頭の記録を指定の開始行番号として最終記録まで順次行番号を付けかえる.
 機能: 被編集ファイル中の先頭からの各記録の行番号を指定に従って順次新しく付けかえる.

(6-3) 表示装置の選択

▶ TC. 又は ▶ CC.
 機能: READY モードの場合と同じ (3.1.1 の (5-4) 参照).

(6-4) 救援

▶ HELP.
 機能: EDIT モードにおける命令の入力形式を表示する.

(6-5) 復帰

▶ RETURN. 下線部省略形 [RETU]
 機能: READY モードへ復帰する.

3.2 データ構造の概要

本エディタはカートリッジディスク中のボリュームを3個使用している。又、編集作業を能率よく高速に行うために主記憶上に2種類の表を用意した。

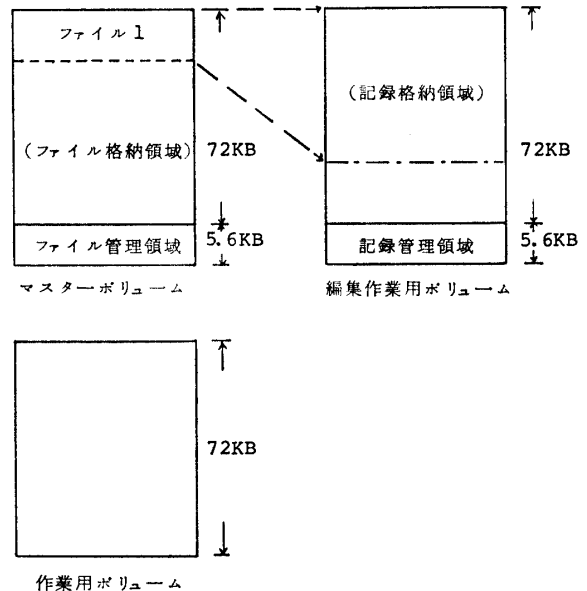


Fig. 2 Data structure of the three kinds of volumes which are used in the editor.

3.2.1 ボリュームの構造

マスターボリュームの大きさは、電子計算機システムの環境に制限されるが、現在は900記録分(1記録=80バイト)の領域をとっている。又、マスターボリュームには最高20件のファイルを登録することができる。但し、ここに登録されたファイル内の記録の総数は、900を越えてはならない。マスターボリューム内のファイル管理領域には、マスターボリュームに登録済のファイルの件数、ファイル名、各記録の行番号リスト、及びその格納番地が格納されている。これは終了命令(END)が入力された時点で、ファイル制御表(3.2.2参照)の情報を使って一部書き換えられる。

編集作業用ボリュームは、新しくファイルを編集する場合、又はマスターボリュームに登録済のファイルをここに転送して再編集する場合に使用する。現在900記録分の編集が行えるようになっている。記録管理領域には、本エディタの終了時(終了(END)命令が入力された時)に編集作業用ボリュームに登録されているファイルのファイル名およびそのファイルの大きさ等の情報が、主記憶上の記録制御表(3.2.2参照)から転送される。これは、他のプログラム・システム(たとえばLISPシステム)が編集作業用ボリューム内に残されたファイルを参照する際に必要となる。

3.2.2 表の機能と構造

主記憶上に設定される2種類の表がファイル制御表と記録制御表である。ファイル制御表の情報は、本エディタに起動がかけられた時点で、ファイル管理領域から転送される。READYモードの各編集作業命令は、このファイル制御表の情報を利用してそれぞれの処理を行う。一方、記録制御表は、ファイル名宣言命令または転送命令が実行される時点で、それぞれの実行開始時における被編集ファイルの状態を示す幾つかのデータが転送される。EDITモードの各編集作業命令は、この記録制御表の情報を利用してそれぞれの処理を行う。

以下に、2種類の表のデータ構造をそれぞれ Fig. 3 と Fig. 4 に示し、各データ領域の意味ないし役割を簡単に述べる。

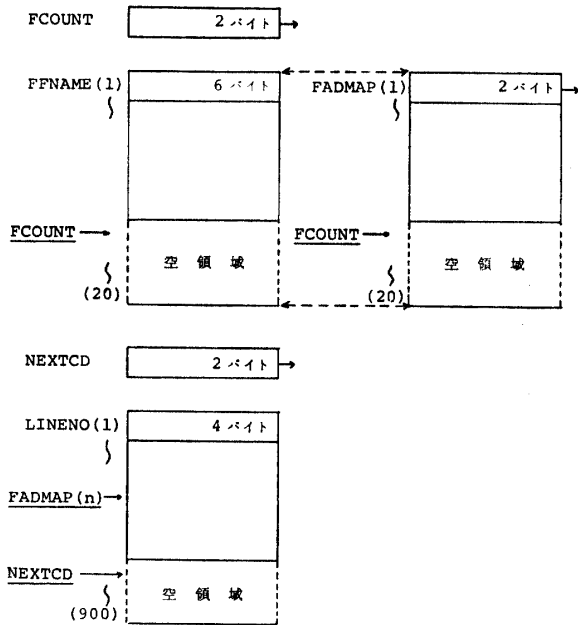


Fig. 3 File-Control table. (An underlined label with an arrow on its right implies that the location number appointed by the arrow is stored in the location indicated by the underlined label.)

(1) ファイル制御表

FCOUNT: マスターボリュームに登録されているファイルの件数に1を加えた値を格納している。これによってファイル名登録領域 (FFNAME) と各ファイルの先頭番地登録領域 (FADMAP) 両方の空領域先頭番地を知ることができる。

FFNAME: マスターボリュームに登録済のファイルの名前を記憶する領域で、最高20件まで登録できる。

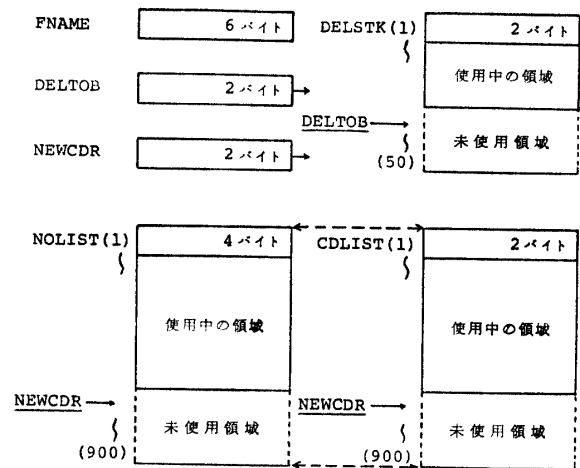


Fig. 4 Record-Control table.

FADMAP: マスターボリューム内の各ファイルの先頭番地を記憶している領域である。各ファイルの大きさは、FADMAPの内容から簡単に計算できる。

NEXTCD: マスターボリューム内の記録に便宜上付けた行番号を記憶する領域の空領域先頭番地を記憶している。

LINENO: マスターボリューム内の記録の行番号を記憶する領域である。

(2) 記録制御表

FNAME: 編集作業用ボリューム内の被編集ファイルの名前が格納される領域である。

DELSTK: 被編集ファイルから削除された記録が登録されていた場所をスタックしておく領域で、現在は50記録分に設定している。

DELTOB: DELSTK 領域を管理する領域である。

NOLIST: 各記録に便宜上付けた行番号が格納されている領域である。

CDLIST: NOLIST と対になっている。CDLISTには、カートリッジディスク上の番地が入る。

NEWCDR: NOLIST 及び CDLIST の未使用領域の先頭を示すポインターである。

3.3 作成

本エディタは、ASSEMBLY 言語と FORTRAN 言語を相補的に使ってコーディングし、MELCOM 9100/30 F 上にインプリメントしている。本エディタを構成するプログラム単位の数はサブルーチン形式にした52本である。これらを、オブジェクト・モジュール形式にしただけのものと、更に単純構造で結合編成してロード・モジュール形式にしたものの2種類としてシステムの標準ボリュームに登録している。なお、本エディタが占めるプログラム領域は、36.7 kB であ

り、データ領域は、9.3KBである。又、本エディタは文字表示装置を駆使するのであるが、そのための手続きのプログラミングにおいて先に作成した支援ソフトウェア²⁾の中からオブジェクト・モジュール形式にした8本のプログラム単位を使用している。

4. 使用方法

4.1 起 動

本エディタは、オブジェクト・モジュールとロード・モジュールの2通りの形式で利用できるようにしてある。従って使用者は、目的に応じてどちらかを使用すればよい。オブジェクト・モジュール形式を使う場合は、本エディタが使用する3個のボリュームを使用側で任意に指定できるので、データファイルの編集を行う際に有用である。これをカードリーダーから起動する場合は、次の4行の制御文を入力する。

```
//LASGN, U 9, FILE=マスターボリューム名.
```

```
//LASGN, U 10, FILE=編集作業用ボリューム名.
```

```
//LASGN, U 11, FILE=作業用ボリューム名.
```

```
//CALL, EDITSY.
```

//LASGN~では、各ボリューム名を指定する。

//CALL~は、MELCOM 9100/30 F のカタログド・プロシジャ機能を利用して、ロード・モジュール化および実行指令に必要な制御文をまとめて呼出している。一方、ロード・モジュール形式を使う場合は、使用するボリュームがあらかじめ決められているので、他のプログラム単位から一定のボリュームを参照する場合（たとえば LISP システム）の編集の際に使用する。

これをカードリーダーから起動する場合には、

```
//EXEC, EDITSY.
```

又、タイプライタから起動する場合には、

```
//* EXEC, EDITSY, F 2.
```

の各制御文を一行入力すればよい。なお、ロード・モジュール形式のエディタは、MELCOM 9100/30 F の場合、Table 1 に示すシステム標準ボリュームを使用している。

Table 1 Correspondence among the Standard volumes in the system, file-reference numbers and the volumes for the disk editor

| | マスター ボリューム | 編集作業用 ボリューム | 作業用 ボリューム |
|--------------|---------------|----------------|--------------|
| ファイル参照 番号 | u 9 | u 10 | u 11 |
| ボリューム名 | DATAFA | TEMP 2 | TEMP |

オブジェクト・モジュール形式とロード・モジュール形式は、カードリーダーあるいはコンソールタイプライタのどちらからでも起動させることができる。但し、コンソールタイプライタから起動する場合の制御文の説明は一部省略した。なお、本エディタは、MELCOM 9100/30 F をノンシンビオント・モードあるいはシングルバッチ・シンビオント・モードのいずれかで運用する際に起動させることができる³⁾。

4.2 操作手順

4.1 で述べたいずれの形式の場合でも、本エディタの実行が開始されると、Fig. 5 中の ㉑ のメッセージが文字表示装置の画面上に表示される。それに続いてライトペンフィールドが現れる。これは、本エディタで使用する各ボリュームが初めて使われるか否かをオペレータに選択させるためのものである。もし初めて使用されるのであれば、それらのボリュームの管理領域がエディタによって初期化される。初めて使用されるのではない場合には、ファイル管理領域の情報がエディタによってファイル制御表に転送される。その後エディタは READY モードになる。以下、3. で述べ

```

[//*EXEC,EDITSY,F2.]
** YAMAGUCHI MELCOM 9100-30F EDITOR-SYSTEM START ** } ㉑
** YAMAGUCHI MELCOM 9100-30F EDITOR-SYSTEM START **
READY.
[>DISPLAY.]
LISP0 1 LISP02 9 LISP01 47
LISP03 62 PISP88 100 LISP88 115
[>LOAD,LISP01.]
[>EDIT.]
EDIT-START.
[>LIST.]
01000 *LISP.
01020 *
01040 *
01060 DEFINE(( ( VECPLUS ( LAMBDA ( VEC1 VEC2 )
( COND
((NULL VEC1) ( ))
( T ( COND ( PLUS ( CAR VEC1 ) ( CAR VEC2 ) ) ( VECPLUS
(CDR VEC1)(CDR VEC2) ) ) ) )
))
01120 *
01140 *
01160 *
01180 *
01200 *
01205 *
01210 *
01215 *
01220 *
01220 *WEND.
[>REPL,1120.
01120 ( T ( CONS ( PLUS ( CAR VEC1 ) ( CAR VEC2 ) )
[>REPL,1140.
01140 ( VECPLUS ( CDR VEC1 ) ( CDR VEC2 ) ) ) ) )
[>XXXXXXXXXX ]
E: ILC
[>RENU,1000-1220-100.]
[>LIST.]
01000 *LISP.
01100 *
01200 *
01300 DEFINE(( ( VECPLUS ( LAMBDA ( VEC1 VEC2 )
( COND
((NULL VEC1) ( ))
( T ( COND ( PLUS ( CAR VEC1 ) ( CAR VEC2 ) ) ( VECPLUS
(CDR VEC1) ( CDR VEC2 ) ) ) ) )
))
01900 *
02000 *
02100 *
02200 *
02300 *
02400 *WEND.
[>RETURN.]
READY.
[>SAVE,EXAMPL.]
[>DISP.]
LISP0 1 LISP02 9 LISP01 47
LISP03 62 PISP88 100 LISP88 115
EXAMPL 130
[>END.]

```

Fig. 5 Examples of usage of the disk-editor using some of the editing commands.

(Bracketed lines in the list are typed in through the keyboard by the operator.)

Table 2 Error message list

| エラーメッセージ | 関係する命令 | 原因及び処置 |
|------------------|--|--|
| E: DISK I/O ERR. | LOAD, SAVE, INPUT, REPLACE, INSERT, CHANGE, LIST | 原因: ディスク・ファイルの形式が悪い。 処置: その命令の所属するモードになる。 |
| F: NON-FILE. | LOAD, DELETE (READY モードの) | 原因: マスターボリュームに指定したファイルが存在しない。 処置: READY モードになる。 |
| E: VOL-FULL | SAVE | 原因: マスターボリュームがいっぱいで登録できない。 処置: READY モードになる。 |
| E: DOUBLE-ENTRY | SAVE | 原因: マスターボリュームに指定したファイルと同一名のファイルが存在した。 処置: READY モードになる。 |
| E: RECORD-FULL | INPUT, INSERT | 原因: 編集作業用ボリュームのファイルの記録の総数が900を越えた。 処置: EDIT モードになる。 |
| E: ILC | すべての命令 | 原因: 異なるモードの命令あるいは誤ったつづりを含む命令を入力した。 処置: その命令が入力される前のモードにもどる。 |

た各モードの編集作業命令とそれに付随したデータとを適宜入力していけばよい。なお、Fig. 5 に編集作業命令の入力とそれに対応して変化するファイルの内容を例示する。又、本エディタが実行時に出力するエラーメッセージを Table 2 にまとめる。

5. 検 討

本エディタの作成に当たって留意した点は、高速に且つ会話型で、文字表示装置の機能を十分利用し、カートリッジディスクのファイルを効果的に編集することであった。これは、主記憶上に2種類の表を設定することで実現できたと考える。一方、本エディタで編集可能なファイルの大きさは、最大900記録分としているため、小さすぎると考えられなくもない。しかし、900記録を越えるような場合には、それを900記録以下に分割して本エディタで編集した後、それらを結合すればよい。そのためのプログラムは FORTRAN 文数行で実現できる。なお、合計して900記録を越えるような大きなファイルをマスターボリュームに登録したい場合は、3.2.1 で述べたマスターボリュームの構造では無理である。たとえマスターボリュームを単純に大きくしても、全体として空領域が占める割合が大きくなって不経済であると思われる。そこで、この

ような場合には、個々のファイルを任意のボリュームに分散型のマスターファイルとして自由に登録できるようにする。同時に、それらのファイルを管理するための管理情報ボリュームを新しく設ける。このようにすれば本エディタは更に拡充できるものと思われる。

6. む す び

本エディタは、豊富な機能を備える最近の TSS エディタから必要最少限の機能だけを抜粋して小型化したようなものとなっている。しかし、会話的に実行される LISP システム (=インタープリター) のようなものを運用するための道具としてはかなり有効に機能するものと考えられる。

参 考 文 献

- 1) 三菱電機: M 345 回線制御プログラム (LCP 345) 説明書, 技術ノート, DCSNF-NO. 1001 (1974)
- 2) 渡部哲夫, 石原好宏: 山口大学工学部研究報告, **30**, 131 (1979)
- 3) 三菱電機: MELCOM 9100 TSOS バッチオペレーティングシステム説明書, FM-SR 00-07 A <47 D0> (1974)

(昭和 56 年 4 月 3 日 受理)