

実行時のファイル利用について(その3) : FORTRAN-D, FORTRAN-Hにおける磁気テープの利用

石田, いつ子
九州大学大型計算機センター研究開発部

<https://doi.org/10.15017/1470300>

出版情報 : 九州大学大型計算機センター広報. 9 (3), pp.187-194, 1976-06-01. 九州大学大型計算機センター
バージョン :
権利関係 :



実行時のファイル利用について(その3)

- FORTRAN-D, FORTRAN-Hにおける磁気テープの利用-

石 田 いづ子^{*1}

ファイルの媒体として磁気テープを使う場合、ファイルの入出力方法、媒体上のデータ形式は、大記憶の場合と同じである。ただし、次の機能は使えない。

1 直接編成入出力

2 UPDATE WRITE

前回の説明で、マルチファイル処理とは、本来磁気テープに作られている複数のファイルを逐次的に処理していく機能であると述べたが、実際には逐次的でなくても処理できる。

また、前回のあとがきで大記憶におけるファイルを拡張する機能は、磁気テープではファイルを追加する機能になると書いたが、実際には大記憶と同様にファイルの拡張となる。

この2点について、例題を挙げて説明する。

1 マルチファイルの処理^{*2}

(1) 磁気テープの先頭から順に、ファイルMT1, MT2, MT3と作っていく場合

```

      :
      WRITE (2) A ..... F02.001
      WRITE (2) B ..... F02.001 } ファイル MT1 に出力する
      :
      ENDFILE 2
      :
      WRITE (2) C ..... F02.002
      WRITE (2) D ..... F02.002 } ファイル MT2 に出力する
      :
      ENDFILE 2
      :
      WRITE (2) E ..... F02.003
      WRITE (2) F ..... F02.003 } ファイル MT3 に出力する
      :
      ¥ SLFILE   F02.001, MT1, QWE123, FILE=NEW
      ¥ SLFILE   F02.002, MT2, QWE123, FILE=NEW*3
      ¥ SLFILE   F02.003, MT3, QWE123, FILE=NEW*3

```

*1 九州大学大型計算機センター研究開発部

*2 FORTRAN-D では マルチファイルの処理はできない。

*3 マルチファイル処理の場合は、FLADD=YES の指定はなくても、ファイルは追加されていく。

(2) (1)で作られたファイルMT1, MT2, MT3をこの順序で読んでいく場合

```

      :
      READ(2) O ..... F02.001 } ファイル MT1 から入力する
      READ(2) P ..... F02.001
      :
      ENDFILE 2
      :
      READ(2) Q ..... F02.002 } ファイル MT2 から入力する
      READ(2) R ..... F02.002
      :
      ENDFILE 2
      :
      READ(2) S ..... F02.003 } ファイル MT3 から入力する
      READ(2) T ..... F02.003
      :
      ¥ SLFILE  F02.001, MT1, QWE123
      ¥ SLFILE  F02.002, MT2, QWE123
      ¥ SLFILE  F02.003, MT3, QWE123

```

(3) (1)で作られたファイルを物理的な位置に関係なくMT1, MT3, MT2の順序で読んでいく場合

```

      :
      READ(2) O ..... F02.001 } ファイル MT1 から入力する
      READ(2) P ..... F02.001
      :
      ENDFILE 2
      :
      READ(2) Q ..... F02.002 } ファイル MT3 から入力する
      READ(2) R ..... F02.002
      :
      ENDFILE 2
      :
      READ(2) S ..... F02.003 } ファイル MT2 から入力する
      READ(2) T ..... F02.003
      :

```

¥ SLFILE F02-001, MT1, QWE123

¥ SLFILE F02-002, MT3, QWE123

¥ SLFILE F02-003, MT2, QWE123

2 ファイルの拡張

大記憶と同様にファイルにデータを追加することが出来るが、マルチファイルになっている場合、その後にあるファイルはすべて読めなくなる。

例 磁気テープにファイルMT1, MT2, MT3, MT4がこの順序にあり、ファイルMT2にデータを追加する場合

```

      :
1    READ(2,END=10)X
      :
      GO TO 1
10   BACKSPACE 2
      WRITE(2)Y
      WRITE(2)Z
      :
¥ SLFILE F02, MT2, QWE123
  
```

ファイルMT2が拡張された結果、ファイルMT3が壊され、ファイルMT3も含めてその後にあるファイルはすべて読めなくなる。

3 磁気テープ上のファイル形式

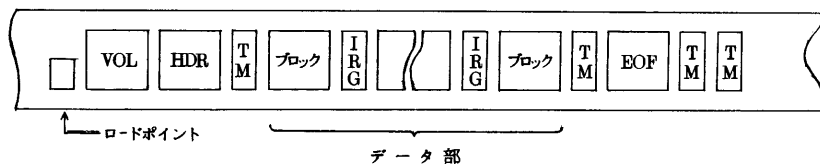


図3・1 単ファイルの場合

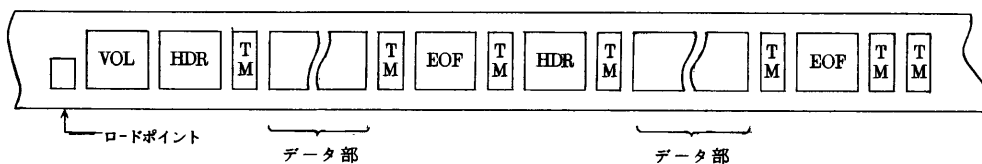


図3・2 マルチファイルの場合

VOL (ボリュームラベル) . . . ボリューム通番, 所有者名等が書込まれている. (80 バイト)
 HDR (前書ファイルラベル) . . . ファイル名, ファイルの作成年月日, レコード形式, ブロック長, レコード長等が書込まれている. (80 バイト)
 EOF (後書ファイルラベル) . . . HDR と同じ内容が書込まれている. (80 バイト)
 TM (テープマーク) ラベルとデータ部を区別するものである. (1 バイト)
 IRG (InterRecord Gap) . . . ブロック間隔

WRITE モードで OPEN すると, HDR と TM が書かれ, CLOSE すると, TM と EOF と 2 個の連続した TM が書かれる。マルチファイル処理の場合は, TM を 1 個消してファイルを作っていく。従って, 最後のファイルの後のみ TM が 2 個続いて書かれている。

ファイルを読んでいく場合, 2 個の連続した TM を検出すると, 以後にファイルはないものとみなす。

4 磁気テープに書ける情報量

磁気テープには, 次の 3 種の長さのものがある。

	磁気テープ 1 巻の長さ	有効長
フルサイズ	782 m (2400 フィート)	725 m
ハーフサイズ	366 m (1200 フィート)	359 m
クォータサイズ	183 m (600 フィート)	176 m

(1) 磁気テープ 1 巻に書ける情報量の目安

磁気テープ 1 巻に書けるバイト数は下式で概算できる。

$$1 \text{ ブロックの長さ} = \frac{\text{ブロック長} * 2}{\text{記録密度} * 3} + \text{IRG の長さ} * 4$$

$$\text{ブロック数} = \frac{\text{磁気テープ 1 巻の有効長}}{1 \text{ ブロックの長さ}}$$

$$\text{バイト数} = \text{ブロック長} \times \text{ブロック数}$$

例 600 フィートの磁気テープにブロック長 7200 バイト (1600 語) で書出す場合

$$1 \text{ ブロックの長さ} = \frac{7200}{814.96} + 1.52 = 24.88$$

$$\text{ブロック数} = \frac{17600}{24.88} = 721$$

$$\text{バイト数} = 7200 \times 721 = 5.19 \times 10^6 \text{ バイト}$$

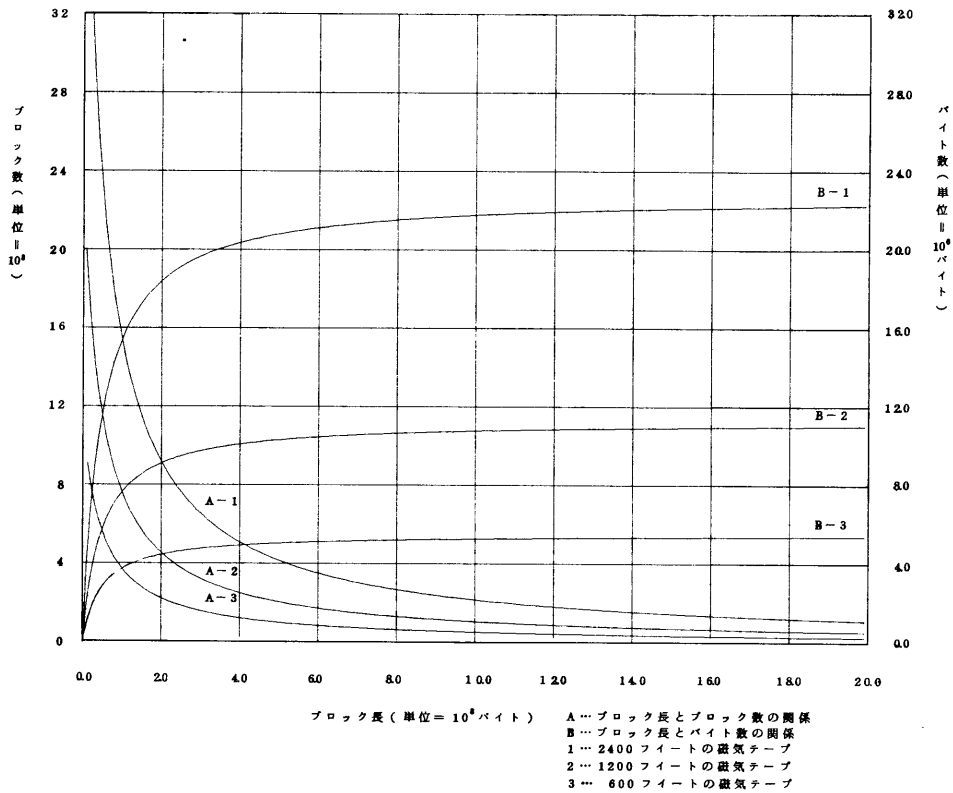
*1 磁気テープは原則としてセンター保管になっており, 利用者が専用ボリューム登録申請の手続きをすると, センターで磁気テープにボリューム通番を書込む。

*2 1 ブロックのバイト数

*3 800 バイト/インチ (= 814.96 バイト/cm)

*4 FACOM 608 F 磁気テープ装置では 1.52 cm

磁気テープにおけるブロック長とブロック数、バイト数の関係をグラフで表わすと次のようになる。



実測してみると上の値より小さい値となった。下表に実測例を示す。(磁気テープの長さは600 フィート)

ブロック長		ブロック数*1	バイト数 (単位 = 10^6 バイト)
バイト	語		
1152	256	3168	3.65
2880	640	1546	4.54
5760	1280	835	4.81
7200	1600	680	4.90
8640	1920	573	4.95
11520	2560	436	5.02
14400	3200	352	5.07

*1 ブロック数は磁気テープにより 1～50 程度の差があった。

5 コントロールカードの入れ方

磁気テープファイルを定義するためのマクロとして ¥SLFILE がある。

	マクロ名	パラメータ
¥	SLFILE	ファイル定義名, ファイル名: ボリューム通番[, FILE=NEW] [, FCBPRM=YES][, BLKSIZE=n][, RCDSIZE=n] [, TMOD=9] [, RECFM={ $\begin{smallmatrix} V \\ U \end{smallmatrix}$ }] [, FLADD=YES]

パラメータの説明

パラメータ	記入したとき	省略したとき
ファイル定義名	12文字以内の名前を指定する。 ^{*1}	省略できない。
ファイル名	44文字以内の任意の名前を指定する。 ^{*2}	省略できない。
ボリューム通番	磁気テープに与えられたボリューム通番を指定する。	省略できない。
FILE=NEW	新たに作成するファイルのとき NEW とする。	OLD となる。
FCBPRM=YES	BLKSIZE, RCDSIZE, TMOD, RECFMを指定するとき, YES とする。	標準値となる。 ^{*3}
BLKSIZE=n	ブロック長をバイト単位で指定する。	n = 1152 となる。
RCDSIZE=n	レコード長をバイト単位で指定する。	n = 1152 となる。
TMOD = 9	転送モードを9ビットモードとする。	8ビットモードとなる。
RECFM={ $\begin{smallmatrix} V \\ U \end{smallmatrix}$ }	レコード形式 ^{*4} を指定する。 可変長形式 …… Vを指定する。 不定長形式 …… Uを指定する。	固定長形式 …… Fとなる。
FLADD=YES	すでにあるファイルの後に作成するとき YES とする。	磁気テープの先頭から作成する。

例1 ブロック長=7200 バイト(1600 語), レコード長=720 バイト(160 語)で書き出す

場合

¥NO

¥USER

¥QJOB

¥FORTRANH

*1 FORTRAN-D, FORTRAN-H ではファイル参照番号と対応したファイル定義名が決められている。

*2 8文字以内の英字で始まり英数字からなる単純名かそれらを・(ピリオド)でつないだものでなければならない。

先頭の単純名以外は数字で始まってもよい。

*3 広報 Vol.9, No.1 p.39, p.41 広報 Vol.9, No.2 p.141 参照。

*4 広報 Vol.9, No.2 p.130, p.131 を参照。

ソースプログラム

¥ LIEDRUNH

データ

```

¥ SLFILE    F02.001,MT1,QWE123,FILE=NEW,FCBPRM=YES,/
              BLKSIZE=7200,RCDSIZE=720
¥ SLFILE    F02.002,MT2,QWE123,FILE=NEW,FCBPRM=YES,/
              BLKSIZE=7200,RCDSIZE=720

```

¥ JEND

例2 ファイルを読む場合 (BLKSIZE, RCDSIZE, TMOD, RECFM はファイルラベル情報に基づいて処理されるので、これらを指定しても無視される。)

¥ NO

¥ USER

¥ QJOB

¥ FORTRANH

ソースプログラム

¥ LIEDRUNH

データ

```

¥ SLFILE    F02,MTFL1,QSR123
¥ SLFILE    F03,MTFL2,QUT456
¥ JEND

```

6 ファイルの内容のDUMP

ファイルの内容を次のようなサブルーチンを利用して DUMP することができる。(ファイルから読み込んだデータが予想していたものと違うというような場合に利用するとよい。)

- ・ サブルーチン名 . . . FLDUMP*¹
- ・ 呼び出し方法 . . . CALL FLDUMP(FDNAME, START, NBLOCK, MODE, ILL)
- ・ パラメータの説明

FDNAME . . . 12文字でファイル定義名を指定する。(文字定数または8語の配列)
 ファイル定義名は12文字以内の英字で始まり、英数字とピリオドからなる任意の名前である。このサブルーチンは FASP 言語で書かれているので、FORTRAN-D, FORTRAN-Hで決められているファイル定義名の規制は受けない。

START . . . DUMPを開始したいブロック番号を指定する。(整数または整数変数)

NBLOCK DUMPしたいブロック数を指定する。(整数または整数)

最後まで DUMPしたいときは実際のブロック数より大きな値 -1 を指定するとよい。

MODE 0 を指定したとき, OCTAL DUMP^{*2}を行なう。

1 を指定したとき, EBCDIC コードで DUMP を行なう。

ILL 完了状態を通知する。(整数)

0 のとき 正常終了。

10000 のとき パラメータの値が誤っている。

$$\left(\begin{array}{l} \text{START} \leq 0 \\ \text{NBLOCK} = 0 \text{ または } < -1 \\ \text{MODE} < 0 \text{ または } \geq 2 \end{array} \right.$$

それ以外のとき 利用の手引 ライブラリ編 No.51 I6/QC/F/
FRDWT の ILL の値を参照のこと。

なお, DUMP するファイルの媒体は磁気テープ, 大記憶どちらでもよい。

例

¥ FORTRAND

⋮

CALL FLDUMP('MTFILE⋮⋮⋮⋮⋮^{*3}', 1, -1, 0, ILL)

IF (ILL .EQ. 0) GO TO 10

WRITE (6, 100) ILL

⋮

10 CONTINUE

⋮

¥ LIEDRUND

¥ SLFILE MTFIELD, ISHIDA, QWE123

*1 詳細は利用の手引 ライブラリ編 No.116 Q9/QC/F/FLDUMP を参照していただきたい。

*2 1語を12桁の8進数としてDUMPする。

*3 ファイル定義名が12文字にみえないときは, 必ず残りに空白をつめて指定する。