

TSS 端末としての MBC 200 の利用 (1)

井 上 徹

近年、コンピュータを始めとする、マイクロ・エレクトロニクスの発達はめざましいものがある。この影響は、自然科学のみならず、社会科学の分野にまで及んでいる。特に実証的な研究方法をとる研究者にとって、コンピュータ等を使って、多量のデータや変数を分析することは、不可避な手法の一つとなっている(澤田他、1982)。また、コンピュータについていえば、単に演算処理に利用するだけでなく、情報データベースとしての利用が進んでいる。例えば、京都大学の大型計算機センターのコンピュータには、教育・教育心理関係のデータベースERICが入っており、筑波大学のセンターには、P A、R I E、C I J Eといったデータベースが入っている。こうしたデータベースを、文献検索において利用することができるのである(宇阪、1982)。

我が国では、全国7カ所の大学に、共同利用の大型計算機センターがあり、大学等の研究機関に所属する研究者のために、その便宜をはかっている。利用にあたっては、利用のための申請が必要であるが、登録さえすればいずれのセンターも利用が可能である。

さて、各センターにおかれているような、超大型コンピュータは、普通2つの処理方式をもっている。バッチ(batch)処理方式と、TSS(Time Sharing System)方式である。バッチ処理は、センター内から、研究者が自分で、プログラムやデータをカード等で入力し、処理結果を自分でプリンターに出力する方式である。通常、研究者自身がセンターに向いて、データや結果の出入れを行なう形式をとっている。高速のセンタープリンターによって、短時間に大量の結果が得られる点、処理の過程を実感できる点などの利点はあるが、遠隔

地からセンターに出向く時間的なロス等を考えると、必ずしも最善の方法とはいえない。

最近よく利用されているのは、センター外から、通信回線を通して利用する、TSS処理方式である。TSS方式の充実によって、今後大型センターは、「データ通信センター」と化し、利用者の姿は見られないといった情景になるのではないかとさえ言われている（芋阪、1983）。さて、芋阪（1983）も述べているように、TSSの特長は、超大型コンピュータを、居ながらにして利用できることにある。パソコン端末と電話回線によって、遠隔地からでも、超大型コンピュータを動かすことができるのである。

従来、我々の研究室では、主にバッチ処理によって、大阪大学大型計算機センターのコンピュータACOSを利用してきた。幸い本学には、何台かのパーソナルコンピュータが入っており、大型コンピュータをTSSで利用するための、インテリジェント端末として、これを活用することが可能である。今回はパソコンのなかのMBC 200を用いて、大阪大学の大型コンピュータACOSとの通信を考えてみた。

本論は、4つの部分からなっている。

1. TSS端末のための基本プログラム。
2. データファイル作成のための、ディスク入力プログラム。
3. データファイル送受信プログラム。
4. TSSを活用した、SPSS利用の実際。

今回は、このなかの1、2を報告する。

1. TSS端末のための基本プログラム

手元にあるパソコンあるいはマイコンと、大型計算機センターのコンピュータの接続には、電話回線を利用する統一規格のインターフェイスRS 232Cを用いる。このインターフェイスは、1ビットずつシリアルに信号を受け渡しする構造をもっており、これに、電気（デジタル）信号を音（アナログ）信号に変換する音響カプラーを接続して、電話回線への情報の受け渡しを行なう。

パソコンをTSS 端末として利用するためには、最低限のTSSソフトウェアが必要である。すなわち、パソコンをTSS 端末として働かせるためのプログラムを作成しなければならない。ここでは、まず基本的な送受信のソフトウェアを、BASICを用いてプログラミングしてみる。なお、以下のプログラムは、藤井(1982)が、PC8801用に開発したものの基本的な部分を、MBC200用に改良したものである。以下順を追って、プログラムの改良点を述べていこう。

まず、藤井(1982)に基づいて、次のようなプログラムを作成した。

プログラム 1

```

10 REM TERMINAL MODE EXERCISE (TERMO)
20 CLS
30 PRINT "TSS-TERMINAL MODE"
40 DEL$=CHR$(&H7F)
50 INIT%1, &H7A, &H37: GOTO 90
60 '
70 END
80 '
90 K$=INKEY$
100 IF K$="" THEN 140
110 IF K$=CHR$(&H1B) THEN GOTO 70
120 PRINT K$;: PRINT%1, K$;
130 '
140 R$=INPUT(1, %1)
150 IF R$="" THEN 90
160 IF R$=DEL$ THEN 140
170 PRINT R$;
180 GOTO 90

```

- 文番号40 DELコードの定義。
- 50 RS232Cシリアルインターフェイスの初期値設定。
- 90 キーボードからの入力。
- 100 入力がなければ、ホストからの受信へ。

- 110 **ESC** キーを用いて、プログラムを終了させる。
- 120 画面表示とホストへの送信。
- 140 RS232Cからデータを1ビット取り込む。
- 150 データがなければ、キーボード入力へ。
- 160 ホストから送られてくる、DEL\$は無視する。
- 170 ホストからのデータを画面に表示。

まず問題になるのは、文番号 50 の RS 232 C の初期値の設定である。菅阪 (1983) によると、最近のパソコンは、ROMにTSSのソフトウェアが用意されており、TERMモードを指定すれば、直ちに無手順端末として使用できるとされている。例えば、大阪大学のACOSの場合、通信パラメータの設定は、「調歩同期式、7データ・ビット、偶数パリティ、1ストップ・ビット、半2重通信」であるが、PC8801でセッティングをする時は、TERMモードを利用して、OPEN TERM *COM:E71NS*, H とすればよい。ところが、MBC200の場合には、TERMモードの機能がなく、INIT\$文で、シリアルポートの初期値を設定しなければならない。通信パラメータの設定は、次のように行なった。

モード設定

ボーレート・ファクター：16× (300ボー)

キャラクタ ：7ビット

パリティ ：イネーブル

パリティ・チェック ：偶数

ストップ・ビット ：1ビット

コマンド設定

送信イネーブル ：1

データ・ターミナル ：1

受信イネーブル ：1

ブレーク・キャラクタ ：0

エラーリセット	: 1
送信キャリア制御	: 1
内部リセット	: 0
ハントモード	: 0

これを16進数を用いて表記し、文番号50のように、&H7A、&H37 と定義した。なお、ボーレイトの設定には、MBC 200 の基板右下のジャンパーピンを変更する必要がある。

次に問題になるのは、ホストのコンピュータからの信号が、RS232C にきているかどうかの判断である。ホストから信号がきていれば、それを表示しなければならない。また信号がきていなければ、キーボードから入力するなりして、こちらから信号を送らねばならない。PC8801では、TERMモードで、ランダムファイルをオープンしているので、LOC (X) がONかOFFかでそれを判断できる。ところがMBC 200 の場合には、それができないため（フェイルエラーになる）。文番号140、150 のような形を考えてみた。ポート番号1から1バイトを取り込み、それがヌルストリングかどうかで判断させようとした。しかし、このプログラムをRUNさせてみると、文番号140でプログラムが止まってしまい、カーソルが点滅するだけで先に進まなくなった。つまり、ホストからの信号を待つ状態のまま、信号が入ってこなければ、先に進まないのである。

この点を解消するためには、ハードの側面に立ち入って考えなければならない。RS232C のシリアルポートのLSIは、8251A が使用されている。このポートは0EDHに設定されており、このコントロール・ステータスの第2ビットに、受信あり・なしのビットがある（1で受信あり、0で受信なし）。このコントロール・ステータスを判断することによって、ホストからの受信があるかないかを、判断することにした。次の1行である。

```
IF (INP(&HED) AND &HD 2)=&HD 2
  THEN (入力あり) ELSE (入力なし)
```

これは次のように2行に書くこともできる。

```
X=(INP(&HED))
```

```
IF (X AND &HD 2)=&HD 2 THEN (入力あり)
```

```
ELSE (入力なし)
```

INP (&HED) は、ポート番号EDから1バイトを取り込む命令であり、論理式ANDによって、他のビットがどうであれ、Xの第2ビットが受信あり(1)であれば、&HO 2と等しくなり、ELSEの後の文番号へとぶ。

このIF文を付け加えることにより、プログラムは正常に走り、キーボード入力⇄シリアルポート受信を交互に繰り返すことになった。

打ち間違った時の、修正のためのバックスペースを付け加えたのが、次のプログラムである。

プログラム 2

```
10 REM TSS-TERMINAL PROGRAM FOR MBC200 BY T.INOUE (TERM1)
20 CLS
30 PRINT "TSS-TERMINAL MODE"
40 DEL$=CHR$(&H7F)
50 INIT %1, &H7A, &H37: GOTO 90
60 '
70 END
80 '
90 K$=INKEY$
100 IF K$="" THEN 150
110 IF K$=CHR$(&H1B) THEN 70
120 IF K$=CHR$(&H08) THEN GOSUB 230: GOTO 90
130 PRINT K$;: PRINT %1, K$;
140 '
150 X=INP (&HED)
160 IF (X AND &H02)=&H02 THEN 170 ELSE 90
170 R$=INPUT$(1, %1)
180 IF R$="" THEN 90
190 IF R$=DEL$ THEN 150
200 PRINT R$;
210 GOTO 90
220 '
230 PRINT %1, CHR$(&H08);
240 LOCATE POS-1, CSRLIN: PRINT " ";: LOCATE POS-1, CSRLIN
250 RETURN
```

タイプをミスした時は、**BK** キーもしくは **→** キーを押すと、文番号 120 によってバックスペースのサブルーチンに入る。

文番号230 バックスペースのコードをホストに送信。

240 画面の1字を消す（カーソルを1つ戻し、blankを打ち、カーソルが前に進むので、blankの位置にまた戻しておく。）。

このプログラム（TERM1）を実際にRUNさせてみると、正常に阪大のACOSとの通信ができた。キーボードとキャラクターディスプレイを用いる、基本的な通信プログラムが完成したといえる。

通信の手続きは、次のように行なう。

- ① MBC200、音響カプラーをセットする。
- ② プログラムをロードし、RUNさせる。
- ③ 電話でホストのコンピュータを呼び出す（所定の電話番号にかけ、つながるとピーという連続音がする）。
- ④ 受信器カプラーにセットする。
- ⑤ カプラーの確認ランプが付いたら、\$\$\$CON、TSS **CR** とタイプする。
- ⑥ ホストからの返答を待つ。

以下、ホストからの問いかけに答えて、ID番号、パスワードを入れて、TSSの会話に入っていく。

基本的にはキーボードとキャラクターディスプレイを用いる通信で十分であるが、多量の実験・調査データを転送する場合には、キーボード入力では不十分になってくる。電話回線をあけたまま、多量のデータをキーボードから打ち込み続けるのは、時間的にも経済的にも大きなロスになる。自分の自由な時間にデータを作成しておき、一度にホストコンピュータに送り込む方が、キーボードから入力するよりも、時間的にも経済的にも効率が良い。これを可能にするのが、フロッピー・ディスクの利用である。

2. データファイル作成のための、ディスク入力プログラム

フロッピー・ディスクの利用には、ホスト・コンピュータACOSへデータを送信する場合と、ホストからいろいろなデータを受信する場合の2つがある。ここではまずホストに送信するデータを準備することを考えてみよう。

まずフロッピーディスクのファイルにデータをインプットするプログラムを作成した。

第一にファイルの種類が問題になる。ファイルには、ランダム・ファイルとシーケンシャル・ファイルの2種類がある。ランダムファイルは、どこからでも自由に読み書きができ、従って、データの修正も、修正したい部分を取り出せば、容易に修正が可能である。しかし、あらかじめ変数の数や大きさを指定しておかなければならないという不自由さがある。一方、シーケンシャル・ファイルの場合には、先頭から順に読み書きしなければならないという制約や、一度入力すると、ファイル全体を入れ換えなければ、データの修正ができないという不便さはあるが、変数の数、大きさなどに関係なく応用がきくのが特徴である横山・後藤、(1981)。ここでは、変数の定義などのいらぬ、シーケンシャル・ファイルを使用して、データを記録することにした。

次に、ファイルに作成するデータの内容であるが、ジョブ・コントロール用のデータ（SPSSのコントロールなど）と、調査結果などの数値データの2種類が考えられる。

ジョブ・コントロール用のデータは、量も少なく、ミスした時も、ファイルの作り直しは簡単に行なえる。しかし、数値データの場合には、大量の数字である場合がほとんどであり、直接ファイルデータを書き込む方式を取ると、パンチミスをした時、ファイルを最初から作り直さねばならないということになり、非常に面倒なことになる。従って、今回のプログラムでは、数値データは、一旦 Dimension に入れて、データを確認した後、ファイルに入力する方法を取った。図一1が、プログラムの簡単なフローチャートである。

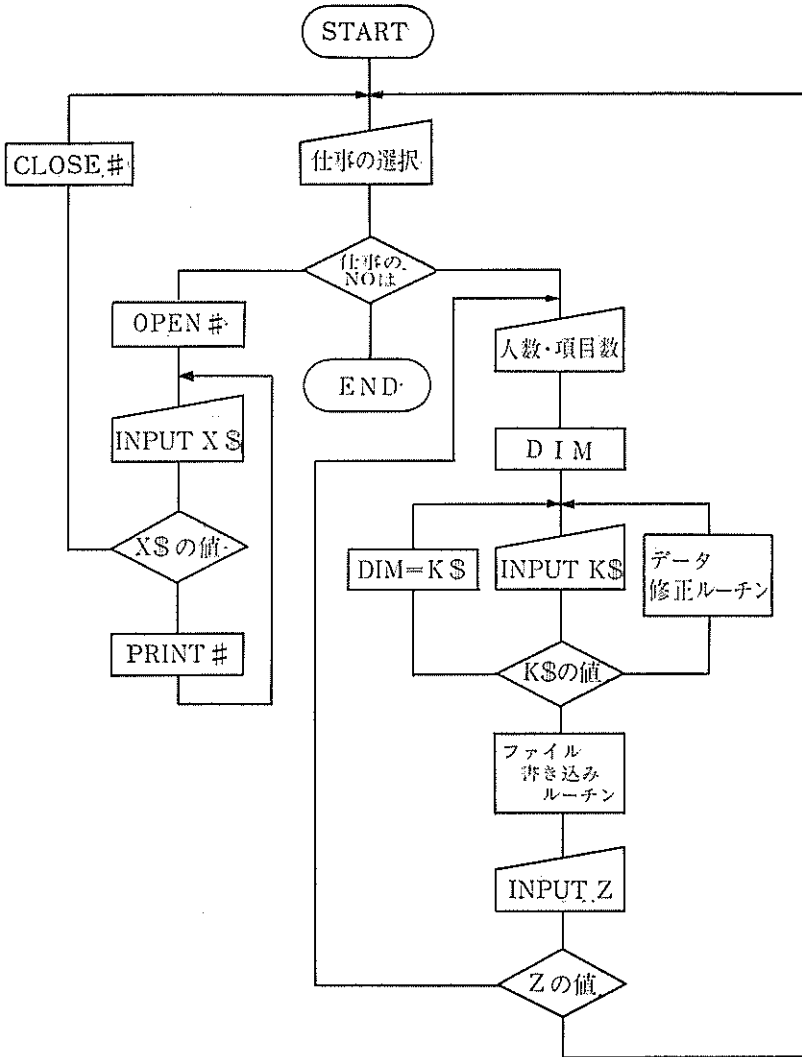


図-1 ファイル入力のためのフローチャート

図中には、ファイルの内容を確かめるためのルーチンなどは省略されているが、ルーチンは、大きく左と右の2つの部分からなっている。左部分がコントロール・データ用のルーチン、右部分が数値データ用のルーチンである。以下順を追って、プログラムを説明していこう。

プログラム 3

```

10 REM FILE INPUT PROGRAM (FINPUT01)
20 '
30 CLS
40 PRINT "シコト ノ イチランヒョウ"
50 PRINT "NO.1 CONTROL FILE サクセイ"
60 PRINT "NO.2 DATA FILE サクセイ"
70 PRINT "NO.3 FILE CONTENTS カクニン"
80 PRINT "NO.4 END"
90 PRINT
100 INPUT "ト`ノ シコト ラ シマスカ? NO.=";Z
110 ON Z GOTO 150, 410, 290, 1410
120 GOTO 90
130 '
140 REM CONTROL FILE
150 CLS: PRINT "CONTROL FILE サクセイ"
160 INPUT "FILE NAME PLEASE =";F$
170 IF F$="FILES" THEN FILES: GOTO 160
180 OPEN "O", #1, F$
190 PRINT "オウル トキ ハ (ZZZZZ) ト オス"
200 LINE INPUT X$
210 IF LEFT$(X$, 5)="ZZZZZ" THEN 240
220 PRINT #1, X$
230 GOTO 200
240 CLOSE #1
250 PRINT "FILE INPUT END": PRINT
260 GOTO 40
270 '
280 REM FILE CONTENTS
290 CLS: PRINT "FILE CONTENTS カクニン": PRINT
300 INPUT "FILE NAME PLEASE =";F$
310 IF F$="FILES" THEN FILES: GOTO 300
320 OPEN "I", #2, F$
330 LINE INPUT #2, K$
340 IF EOF(2) THEN 370
350 PRINT K$: LPRINT K$
360 GOTO 330
370 CLOSE #2
380 PRINT "DATA END & FILE CLOSE": PRINT
390 GOTO 40
400 '

```

文番号

- 40～80 仕事の内容の画面表示。
- 100, 110 仕事の選択。
- 150 コントロール用のファイル作成の表示。
- 160, 170 ファイル名の指定もし FILES と打つと、既存ファイルの一覧表が出る。
- 180 ファイルのオープン。
- 190～210 データのタイプ、終わる時は、ZZZZZ とタイプする。
- 220 ファイルへの書き込み。
- 240 ファイルをクローズ、画面に表示。
- 290 画面表示、ファイル内容の確認。
- 300, 310 ファイル名の指定。
- 320 ファイルのオープン。
- 330 ファイル中の文字を、順に一行ずつ、復改 (CR) のところまで読み取る。
- 340 ファイル・エンドの時は、ファイル・クローズへ。
- 350 一行ごとに、画面とラインプリンターに表示。
- 370, 380 ファイルをクローズ、画面に表示。

文番号

- 440, 450 項目数と被験者数を入れる。
- 460 配列の宣言。
- 480～500 オーバーフローする時は、被験者の数を少なくして入れ直す。
- 530, 540 終了時、修正時の指示を表示。
- 560 被験者番号をインプット。
- 590, 600 データ番号のカウンタと表示。
- 610 被験者番号を文字タイプで配列に入れる。
- 620～660 各データのインプットと配列への投入。

プログラム 4

```

410 REM DATA FILE
420 REM DATA INPUT (DIMENSION)
430 ON ERROR GOTO 480
440 INPUT "コウモク ノ カス". P=";P
450 INPUT "ヒケンシヤ スウ S=";S
460 DIM X$(S,P)
470 GOTO 510
480 IF ERR=7 THEN 490
490 INPUT "ヒケンシヤ スウ S=";S
500 RESUME
510 I=0
520 PRINT
530 PRINT "セ`ンフ` ウチオワツタラ (-999)"
540 PRINT "DATA INPUT ラ ミスツタラ (-111)"
550 PRINT
560 INPUT "ヒケンシヤ NO.=";N
570 IF N=-999 GOTO 850
580 IF N=-111 THEN GOSUB 730: GOTO 530
590 I=I+1
600 PRINT "DATA NO.=";I
610 X$(I,0)=STR$(N)
620 FOR J=1 TO P
630 PRINT "X(";J;")=";
640 INPUT " ";X$
650 X$(I,J)=X$
660 NEXT J
670 PRINT "DATA NO.";I;" END"
680 PRINT
690 IF I=S THEN PRINT "-----DATA ハ オワリマシタカ?":PRINT
700 GOTO 530
710 '
720 REM DATA CORRECTION
730 INPUT "テ`ータ ノ NO.=";D
740 INPUT "ヒケンシヤ NO.=";N
750 X$(D,0)=STR$(N)
760 FOR J=1 TO P
770 PRINT "X(";J;")=";
780 INPUT " ";X$
790 X$(D,J)=X$
800 NEXT J
810 PRINT "シュウテイ オワリ"
820 PRINT: RETURN
830 '

```

プログラム 5

```

840 REM DATA PRINT OUT
850 PRINT
860 PRINT "INPUT DATAヲ PRINT OUT シマスカ?"
870 INPUT "PRINT OUT (1)YES (2)NO";Z
880 ON Z GOTO 890, .1060: GOTO 870
890 D=0
900 LPRINT
910 FOR I=1 TO S
920 D=D+1
930 LPRINT "DATA ";D;TAB(9);
940 FOR J=0 TO P
950 LPRINT X$(I,J);
960 NEXT J: LPRINT
970 NEXT I
980 PRINT
990 PRINT "ミス カ ナイカ クシカメヨウ!"
1000 PRINT
1010 INPUT "ミス (1)ナシ (2)アリ";Z
1020 PRINT
1030 ON Z GOTO 1060, 530: GOTO 1010
1040 '
1050 REM WRITING INTO THE FILE
1060 PRINT .
1070 INPUT "FILE NAME PLEASE =";F$
1080 IF F$="FILES" THEN FILES: GOTO 1060
1090 PRINT "DATAヲ ハシ"メテ イレルトキ ハ (1)ヲ, DATAヲ ツイカ スルトキ ハ
1100 INPUT "OPENING (1) OR APPENDING (2) WHICH?=";Z
1110 ON Z GOTO 1120, 1150: GOTO 1090
1120 REM OPEN "O" MODE
1130 OPEN "O", #1, F$
1140 GOTO 1170
1150 REM OPEN "A" MODE
1160 OPEN "A", #1, F$
1170 FOR I=1 TO S
1180 FOR J=0 TO P
1190 PRINT X$(I,J)
1200 PRINT #1, X$(I,J);
1210 NEXT J
1220 PRINT #1, CHR$(&HOD)
1230 NEXT I
1240 PRINT "END OF WRITING INTO THE FILE"
1250 '

```

- 670 一人の被験者のデータの終わりを表示。
- 690 全データ終了の確認を促がす表示。
- 720 コメント、データ修正。
- 730, 740 データ番号と被験者番号を入れる。
- 760~800 修正したデータを入れる。
- 810 画面表示、修正終わり。

文番号

- 860 画面表示、プリンターに出力するか。
- 870, 880 プリンター出力の選択。
- 900~970 プリンターへの出力。
- 990 ミスの確認を促がす表示。
- 1010~1030 ミスがない時はファイル入力へ、ある時は修正ルーチンへ。
- 1050 コメント、ファイルへの書き込み。
- 1070, 1080 ファイル名の指定。
- 1090~1110 ファイルオープンを選択。
- 1130 Oモードでファイルオープン。
- 1160 Aモードでファイルオープン。
- 1170~1230 ファイルへの書き込み、各被験者の区切りには、復改 (CR) コードを入れる。
- 1240 画面表示、ファイルの書き込み終了。

文番号

- 1270 画面表示、データを追加するか。
- 1280, 1290 データ追加の選択。
- 1320 データを追加する時は、すべての変数、配列をクリアしておく。
- 1350, 1360 データの最後に \$\$\$E O F を付けるかどうか。

プログラム 6

```

1260 REM DATA ADDING
1270 PRINT "DATA ヲ サラニ ツイカ シマスガ?"
1280 INPUT "DATA (1)マク アリマス OR (2)モウ アリマセン. WHICH?=";Z
1290 ON Z GOTO 1310,1350: GOTO 1280
1300 '
1310 REM CLEAR ALL VARIABLES
1320 CLEAR: PRINT
1330 GOTO 440
1340 '
1350 INPUT "FILE CLOSE -- $$$EOF ヲ ツクマスガ? (1) YES (2) NO =" ;Z
1360 ON Z GOTO 1370, 1390: GOTO 1350
1370 PRINT #1, CHR$(&H24);CHR$(&H24);CHR$(&H24);CHR$(&H45);>
                                     ↳ CHR$(&H4F);CHR$(&H46)
1380 '
1390 PRINT "FILE CLOSE": CLOSE #1: PRINT
1400 GOTO 40
1410 END

```

- 1370 \$\$\$EOFのコードを追加。
- 1390 ファイルをクローズ。
- 1410 END.

このプログラムの特徴は、数字の配列である調査データを、より簡単に入力できるように工夫したところにある。またファイル入力前に、簡単にデータの修正ができるようにした。特に文番号410以降のデータ・ファイル作成では、被験者番号の投入によって、プログラムの行き先を決めているので、被験者番号とは別に、データ番号をつけ、配列での位置を示すようにした。これによって、被験者番号が順に並んでいなくても、データの投入ができるようになっている。ただし、データの修正をする時には、データ番号を入れる必要があるが、画面やラインプリンターで番号を確認できるので、それほど煩わしくはないと思われる。

一時、配列のなかにデータを入れる点であるが、配列宣言では大きなメモリーを取ると、オーバーフローエラーが生じる。エラーが起きた時の修正ルーチンを付け、データが多い時には、被験者を区切って、ファイルに入れるようにした。実際にやってみると、150項目×50名分なら十分に一度に入力できる。

また、初回とそれ以降の場合に、ファイルのオープンの仕方が異なるので、文番号1090～1110の選択プログラムを設けている。これによって、データが一度に入り切らない場合にも、何度でも同じファイルにデータを加えることができる。

文番号1370は、データの一番終了に\$\$\$EOFを付けるプログラムである。ホスト・コンピュータのACOSが、転送データを読む時、データの終了を知るのが\$\$\$EOFであり、付いていなければ、データが終った時にキーボードから送信する必要がある。ただし、データを追加する可能性のある時は、付けてはならない。

以上で、ホスト・コンピュータにデータを転送する準備は終った。一見すると面倒なようであるが、実際にやってみると、手のあいた時間に、自由にデータの準備ができて非常に便利である。カードでバッチ処理をする場合には、ミスをするミスしたカードを破棄しなければならないが、ディスクの場合には修正がきくので、経済的な節約にもなる。最も重要なのは、心理的な安心感であろうか。慣れないうちは、機械の前に座っただけで、緊張感を覚えるものである。TSSで直接大型コンピュータと対話しているとなれば、ミスしてはならないという気持ちになり、かえってミスが増えたりうろたえたりする。その点、使い慣れているパソコンであれば、気も楽であり、余裕を持ってデータの準備ができる。

さて、いよいよホストのコンピュータにデータを転送し、計算等を行なう訳であるが、データを送信するとともに、結果を受信するプログラムも必要となる。1の基本プログラムでも受信は可能であるが、受信する量が多いと画面が流れてしまい実際の役に立たないことがある。従って、こうした点を考慮に入れたプログラムを次に考えなければならない。基本プログラムを発展させたプログラムは、具体的な計算例を含めて、稿を改めて述べることにしたい。なお、今回発表したプログラム、特にファイル投入用プログラムは、ある程度形は整っているが、まだまだ未完成である。例えば、データの修正にしても、一人の被験者全体を打ち直さなくとも、第何項目に指定して直すこともできるは

ずである。諸先輩の御批判を仰ぎながら、より良いものにしていきたくと思う。

最後に、本稿の執筆にあたって、大阪大学大型計算機センターの藤井博氏、三洋電機株式会社情報技術開発部の江田義久氏の2氏より、多大の助言をいただいた。また、本学の中嶋正夫教授には、MBC200の使用について指導をいただいた。改めて、ここに謝意を表す。

文 献

- 藤井 博, パーソナル・コンピュータPC8801を用いたインテリジェント・ターミナル, 大阪大学大型計算機センター・ニュース, 1982, Vol. 12, (2), 83-94.
- 宇阪直行, データベースによる文献検索—パソコン端末による検索—, サイコロジー, 1982, Vol. 3, (12), 53-65.
- 宇阪直行, コンピュータ・コミュニケーション, 心理学ラボラトリ・コンピュータシリーズ2, ナカニシヤ出版, 1983.
- 三洋電機株式会社, MBC-200/2251/250取扱説明書.
- 三洋電機株式会社, SBASIC III プログラミングマニュアル.
- 澤田慶輔他編, 特集=マイコン利用の最前線, サイコロジー, 1982, Vol. 3, (11), 5-85.
- 横山 淳・後藤 猛, ビジネス利用のための DISK BASIC プログラミング教本, NBCPC-8000シリーズ, 廣済堂出版, 1981.

