

Title	マイコンを用いたTSSインテリジェンス端末
Author(s)	多田, 元英
Citation	大阪大学大型計算機センターニュース. 42 P.87-P.93
Issue Date	1981-08
Text Version	publisher
URL	<a href="http://hdl.handle.net/11094/65491">http://hdl.handle.net/11094/65491</a>
DOI	
rights	

*Osaka University Knowledge Archive : OUKA*

<https://ir.library.osaka-u.ac.jp/repo/ouka/all/>

## マイコンを用いたTSSインテリジェンス端末

大阪大学工学部 多田元英

## 1. はじめに

最近、マイコンは高性能化、低価格化により広く一般に普及している。マイコンの最大の特徴は、随時利用できること、及び各種の数値計算、実験データの整理、各種データの保守・保存など、その利用方法の汎用性にあるといえる。しかし、かえってその広範な汎用性のために、ソフトのサポートは極めて遅れているといえ、マイコンの機能の大半を眠らせたまま使用しているユーザーが筆者を含めて少なくなろう。本来、計算機はあくまでも手段であって目的ではないという観点に立てば、ソフトのサポートの充実をメーカー側が図ってくれば幸いである。

NECのマイコンシステム(PC-8001)の拡張ユニット(PC-8011)には、RS232Cインターフェイスが装備されており、ハードウェアの面ではそのままTSS端末として使用できる。しかし、マニュアル<sup>(1)</sup>のプログラム例は極めて基本的なものであり、このままではマイコンの機能のごくわずかししか利用できていないといえる。

ここに紹介するプログラムは、上記のプログラムをベースにして、単なるTSS端末として利用できるだけでなく、大型計算機から出力されるデータをミニフロッピーディスクに収納するための機能も兼ね備えている。筆者の研究室では、XYプロッターを用いて、ディスクに収納された大型計算機からのデータをグラフ化するなど、データのマイコンによる二次処理に調法している。

## 2. マイコンシステムの構成

筆者の研究室のマイコンシステムを写真1及び表1に示す。なお大型計算機とマイコン端末との転送レートは300BPSである。



写真1. マイコンシステムの全景

表1. マイコンシステムの構成

品名	型番(メーカー)
本 体	PC-8001(NEC)
ディスプレイ	JC-1202DH(NEC)
プリンター	MP-80(EPSON)
ミニフロッピー ディスク	PC-8031(NEC)
拡張ユニット	PC-8011(NEC)
音響カプラー	DPC-311(SANYO)
XYプロッター	WX-4671(渡辺測器)

(1) PC-8011ユーザーズマニュアル NEC発行

### 3. プログラムの概要

#### 操 作 法

制御キーの機能を以下に示す。

**GRAPH** + Q —— プレーク

**GRAPH** + Z —— プリンターのスイッチをON

**GRAPH** + X —— プリンターのスイッチをOFF

**GRAPH** + C —— プリンターのカラム数を132に設定

**GRAPH** + V —— プリンターのカラム数を80に設定

**GRAPH** + A —— シフトロック(アルファベットのみ)

**GRAPH** + S —— シフトロック解除

**GRAPH** + D —— ディスクOUTPUTルーチンのスイッチ(ON及びOFF)

**GRAPH** + B —— ハードコピーの準備

**GRAPH** + N —— ハードコピーの準備を解除

**GRAPH** + F —— ファンクションキー設定サブルーチンの呼びだし

画面のハードコピーをとりたいたときは、**GRAPH** + Bを入力した後、**f·9**キーを入力する。この際、**f·5**～**f·10**キーは効かなくなるが、これらを使用したければ、**GRAPH** + Nを入力すればよい。また、ファンクションキーを新たに定義したいときは、**GRAPH** + Fを入力してから、定義したいファンクションキーの番号、定義したい文字を入力し、さらに、RUN **CR**のように最後にコントロール信号を加えたい場合は、コントロールキーのキャラクターコードを入力する。( **CR** の場合は13又は&HD) コントロール信号が要らないときは **CR** のみを入力すればよい。

本プログラムでは、大型計算機からマイコンに出力されるデータを **CR** から **CR** までを1つの文字変数としてミニフロッピーディスクに出力し、データファイルを作ることができる。但し、このファイル内のデータはすべて文字変数として収納されるため、これを数値としてマイコンで利用したいときには、次章に示すようなプログラムを用いてデータを数値化する必要がある。以下にディスクへ文字変数としてのデータを出力させるときの操作法を示す。

- ① 出力用のディスクをドライブ2に入れる。(MOUNTの必要はない。)
- ② **GRAPH** + Dを入力し、プログラムをディスクOUTPUTルーチンに入れる。
- ③ \*\*Data file name? の問に対して、作成するデータファイルの名前を入力する。ここでディスクへの出力を取りやめたいときは **CR** を入力する。
- ④ LISTなどのキー操作により、大型計算機からマイコンにデータを出力させる。このデータが前記の仕様に従ってディスクに出力される。

- ⑤ **GRAPH** +Q (ブレークキー)により途中でデータの出力を中断すると、それまでに出力されたデータによりデータファイルが作成され、プログラムはディスクOUTPUTルーチンから抜ける。
- ⑥ データファイルの作成が終了すれば、**GRAPH** +Dを入力し、プログラムをディスクOUTPUTルーチンから脱出させる。この操作を忘れると、それ以後の大型計算機からマイコンへの出力がすべてデータファイルに収納されてしまうので注意を要する。

#### TSS端末用のプログラムリスト

```

10 '#####
20 ' TSS RUNNING PROGRAM (TSS)
30 '
40 WIDTH 80,20 : CONSOLE 0,25,1,0 : COLOR 0,32,0 : PRINT CHR$(16C)
50 INIT Z1,&H7A,&H37 ----- RS 232Cチャンネルの初期設定
60 IF PORT (1) THEN KK=PORT(1) : A#=INPUT$(KK,Z1) ELSE 110
70 I=1
80 I=INSTR(I,A$,CHR$(127))
90 IF I THEN A#=LEFT$(A$,I-1)+MID$(A$,I+1) : GOTO 80
100 PRINT A$; : IF COPY THEN LPRINT A$;
110 B#=INKEY$: IF B#="" THEN 60 ELSE IF ASC(B#)<&H80 THEN 230
120 IF B#=CHR$(16C) THEN OUT &HC1,&H3F : BEEP : OUT &HC1,&H37 : GOTO 60 --- ブレーク信号の送信
130 IF B#=CHR$(16) THEN COPY=1 : GOTO 60 ----- プリンターのスイッチON
140 IF B#=CHR$(16) THEN COPY=0 : GOTO 60 ----- プリンターのスイッチOFF
150 IF B#=CHR$(16) THEN LPRINT CHR$(16F); : GOTO 60 ----- プリンターのカラム数を132に設定
160 IF B#=CHR$(16) THEN LPRINT CHR$(16); : GOTO 60 ----- プリンターのカラム数を80に設定
170 IF B#=CHR$(16) THEN 270 ----- ディスクOUTPUTルーチンへの切り換え
180 IF B#=CHR$(16) THEN GOSUB 1000 : GOTO 60 ----- ファンクションキー設定サブルーチンの呼び出し
190 IF B#=CHR$(16) THEN LOWER=1 : GOTO 60 ----- シフトロック
200 IF B#=CHR$(16) THEN LOWER=0 : GOTO 60 ----- シフトロック解除
210 IF B#=CHR$(16) THEN POKE &HEA58,1 : GOTO 60 ----- ハードコピーの準備
220 IF B#=CHR$(16) THEN POKE &HEA58,0 : GOTO 60 ----- ハードコピーの準備を解除
230 IF B#=CHR$(16) THEN PRINT Z1,B$; : PRINT CHR$(16); " ";CHR$(16);
      : IF COPY THEN LPRINT B$; : GOTO 60 ELSE GOTO 60
      ] デリート信号の送信及び画面、プリンターへの出力
240 IF LOWER THEN IF B#="a" AND B#<="z" THEN B#=CHR$(ASC(B#)-32)
250 PRINT Z1,B$; : PRINT B$; : IF COPY THEN LPRINT B$;
260 GOTO 60

```

(次頁に続く)

注1) 大型計算機からの **(LF)** 信号は、そのあとにキャラクターコード127の信号が付随するため、これを除去するためのフィルター。

#### (ディスクOUTPUTルーチン)

```
270 '
280 PRINT "***** Confirming the DISK mounted on drive2 *****"
290 C$="" : INPUT "Data file name ";C$
300 IF C$="" THEN PRINT "***** Setting DISK Cancelled *****" : GOTO 60
310 REMOVE2 : MOUNT 2 : OPEN "2:"+C$ FOR OUTPUT AS #1
320 IF PORT (1) THEN KK=PORT(1) : A$=INPUT$(KK,1) ELSE 370
330 I=1
340 I=INSTR(1,A$,CHR$(127))
350 IF I THEN A$=LEFT$(A$,I-1)+MID$(A$,I+1) : GOTO 340
360 PRINT A$; : PRINT #1,A$; : IF COPY THEN LPRINT A$;
370 B$=INKEY$ : IF B$="" THEN 320 ELSE IF ASC(B$)<&H90 THEN 490
380 IF B$=CHR$(&H9C) THEN OUT &HC1,&H3F : BEEP : OUT &HC1,&H37
    : CLOSE #1 : REMOVE2 : PRINT "File closed" : GOTO 60
390 IF B$=CHR$(&H90) THEN COPY=1 : GOTO 320
400 IF B$=CHR$(&H81) THEN COPY=0 : GOTO 320
410 IF B$=CHR$(&H82) THEN LPRINT CHR$(&HF); : GOTO 320
420 IF B$=CHR$(&H83) THEN LPRINT CHR$(&H12); : GOTO 320
430 IF B$=CHR$(&H86) THEN CLOSE #1 : REMOVE2 : PRINT "File closed" : GOTO 60
440 IF B$=CHR$(&H87) THEN GOSUB 1000 : GOTO 320
450 IF B$=CHR$(&H9E) THEN LOWER=1 : GOTO 320
460 IF B$=CHR$(&H9F) THEN LOWER=0 : GOTO 320
470 IF B$=CHR$(&H84) THEN POKE &HEA59,1 : GOTO 320
480 IF B$=CHR$(&H85) THEN POKE &HEA59,0 : GOTO 320
490 IF B$=CHR$(&H8) THEN PRINT #1,B$; : PRINT CHR$(&H1D);" ";CHR$(&H1D);
    : IF COPY THEN LPRINT B$; : GOTO 320 ELSE GOTO 320
500 IF LOWER THEN IF B$="a" AND B$<="z" THEN B$=CHR$(ASC(B$)-32)
510 PRINT #1,B$; : PRINT B$; : IF COPY THEN LPRINT B$;
520 GOTO 320
```

ブレーク信号の送信及びディスク  
OUTPUTルーチンからの脱出

ディスクOUTPUTルーチンからの脱出

#### (ファンクションキー設定サブルーチン)

```
1000 KY=0 : INPUT "key No. " ;KY : IF KY<1 OR KY>10 THEN RETURN
1010 KY$="" : KC=0 : INPUT "Input string " ;KY$
    : INPUT "Input CTL-key No. " ;KC
1020 KEY KY,KY$+CHR$(KC) : RETURN
```

#### 4. ディスクに収納されたデータの利用例

前章のプログラムでは、大型計算機から伝送されたデータはディスクに文字変数として収納される。我々は数値データを取り扱うことが多いため、このデータをマイコンで二次処理するために（例えばXYプロッターを用いた作図など）、文字変数から数値に変換しておくことが望ましい。以下に、前章のプログラムで作成された文字変数よりなるデータファイル（以下、文字変数型ファイルと呼ぶ）を数値に変換して、数値変数よりなる新しいファイル（以下、数値変数型ファイルと呼ぶ）に収納し直すプログラムを紹介する。

##### 操 作 法

- ① 文字変数型ファイルの名前を入力する。
- ② 画面上部に表示された文字変数型ファイルの一行分のデータ及びカラム番号を参照して、一行内に含まれる数値データの個数を入力し、さらに行内の各数値データの最初のカラム番号と最後のカラム番号を順次入力して行内のフォーマットを設定する。この際画面上部に、対応す

るカラム位置に“H”(データの最初)と  
 “E”(データの最後)が確認のために表  
 示される。なお、行内で同様のフォーマ  
 ットでデータが収納されている場合には、2  
 番目の数値データの最後のカラム番号から  
 後は、**CR** キーを入力していけば自動的  
 にフォーマットが設定される。

- ③ 新しく作成する数値変数型ファイルの名  
 前を入力する。
- ④ ファイルの変換作業が開始され、終了後  
 に、他のファイルの変換作業を続けて行  
 うか、同じフォーマットかどうかの間が画面  
 に表示される。

T=	0.02	A.MAX=	0.76002E+01
T=	0.04	A.MAX=	0.75717E+01
T=	0.06	A.MAX=	0.76402E+01
T=	0.08	A.MAX=	0.77452E+01
T=	0.10	A.MAX=	0.78878E+01
T=	0.12	A.MAX=	0.81121E+01
T=	0.14	A.MAX=	0.81923E+01
T=	0.16	A.MAX=	0.86927E+01
T=	0.18	A.MAX=	0.90339E+01
T=	0.20	A.MAX=	0.95421E+01
T=	0.22	A.MAX=	0.98593E+01
T=	0.24	A.MAX=	0.10485E+02
T=	0.26	A.MAX=	0.11835E+02
T=	0.28	A.MAX=	0.15740E+02
T=	0.30	A.MAX=	0.22661E+02
T=	0.32	A.MAX=	0.21169E+02
T=	0.34	A.MAX=	0.27317E+02
T=	0.36	A.MAX=	0.39905E+02
T=	0.38	A.MAX=	0.37201E+02
T=	0.40	A.MAX=	0.28974E+02

図 1(a) 文字変数型ファイル例

.02	7.6002	.04	7.5717	.06
7.6402	.08	7.7452	.1	7.8878
.12	8.1121	.14	8.1923	.16
8.6927	.18	9.0339	.2	9.5421
.22	9.8593	.24	10.485	.26
11.835	.28	15.74	.3	22.661
.32	21.169	.34	27.317	.36
39.905	.38	37.201	.4	28.974

図 1(b) 数値変数型ファイル例

図 1 に文字変数型ファイル ( a 図 ) と、このプログラムで変換された数値変数型ファイル ( b 図 )  
 の例を示す。また図 2 に入力操作時の画面の状態を示す。

```

***** Column number *****
80      90      100     110     120     130     140     150
      10      20      30      40      50      60      70
1234567890123456789012345678901234567890123456789012345678901234567890
T=  0.06      A.MAX= 0.76402E+01
  H   E           H           E

TSS made data file name ? STRING      Value type data file name ? VALUE

Number of data in 1 line ? 2

Data head column No. of 2 th data ? 21
Data end column No. of 2 th data ? 32
  
```

図 2. 入力操作時の画面の状態

ファイル変換用のプログラムリスト

```

10 *****
20 ' CONVERT TSS MADE STRING TYPE DATA FILE TO VALUE TYPE DATA FILE (CONV)
30 '
40 PRINT CHR#(&HC)
50 CONSOLE 0,20,1,0 : WIDTH$0,20
60 LOCATE 0,9,1 : INPUT "TSS made data file name ";A$
70 IF LEFT$(A$,1)="2" THEN REMOVE2 : MOUNT2 ELSE REMOVE1 : MOUNT1
80 OPEN A$ FOR INPUT AS #1
90 FOR I=1 TO 5 : LINEINPUT #1,Z$ : NEXT I : CLOSE #1
100 LOCATE 0,0,1 : PRINT "***** Column number *****
*****";
110 PRINT "80          90          100          110          120          130          140
150"
120 PRINT "          10          20          30          40          50          60
70"
130 PRINT "123456789012345678901234567890123456789012345678901234567890123456789
01234567890";
140 PRINT Z$ : LL=CSRLIN
150 IF C$="y" THEN 280
160 LOCATE 0,12,1 : INPUT "Number of data in 1 line ";Z3
170 DIM Z1(Z3),Z2(Z3)
180 FOR I=1 TO Z3
190 LOCATE 0,14,1 : PRINT "Data head column No. of";I;"th data "; : INPUT Z1(I)
200 IF Z1(I)=0 THEN IF I<=2 THEN 190 ELSE Z1(I)=Z1(I-1)+Z1(I-1)-Z1(I-2)
210 IF Z1(I)>80 THEN L1=LL+1 : L2=Z1(I)-81 ELSE L1=LL : L2=Z1(I)-1
220 LOCATE L2,L1,1 : PRINT "H";
230 LOCATE 0,15,1 : PRINT "Data end column No. of";I;"th data "; : INPUT Z2(I)
240 IF Z2(I)=0 THEN IF I=1 THEN 230 ELSE Z2(I)=Z1(I)+Z2(I-1)-Z1(I-1)
250 IF Z2(I)>80 THEN L1=LL+1 : L2=Z2(I)-81 ELSE L1=LL : L2=Z2(I)-1
260 LOCATE L2,L1,1 : PRINT "E";
270 NEXT I
280 II=0
290 LOCATE 0,9,1 : INPUT "Value type data file name ";B$
300 IF LEFT$(B$,1)="2" THEN REMOVE2 : MOUNT2 ELSE REMOVE1 : MOUNT1
310 OPEN A$ FOR INPUT AS #1
320 OPEN B$ FOR OUTPUT AS #2
330 CONSOLE 11,20,1,0
340 IF EOF(1) THEN 490 ELSE LINEINPUT #1,Z$
350 PRINT Z$
360 FOR Z9=1 TO Z3
370 Z1$=MID$(Z$,Z1(Z9),Z2(Z9)-Z1(Z9)+1)
380 Z2$=LEFT$(Z1$,1)
390 IF Z2$>="0" AND Z2$<="9" OR Z2$="+" OR Z2$="-" OR Z2$="." THEN 420
400 Z1$=MID$(Z1$,2) : IF Z1$="" THEN IF Z9=1 THEN 340 ELSE 460
410 GOTO 380
420 ZZ=VAL(Z1$) : PRINT ZZ, : PRINT #2,ZZ
430 NEXT Z9
440 II=II+1 : PRINT
450 GOTO 340
460 PRINT "***** String Error in line";II
470 PRINT " Error occured string is" : PRINT Z$
480 CLOSE : END
490 CLOSE : PRINT : PRINT "***** Data file converted *****"
500 INPUT "Want another work (y/n)";A$
510 IF A$="n" THEN CONSOLE 0,20,1,0 : END
520 IF A$<>"y" THEN 500
530 INPUT "Same format (y/n)";C$
540 IF C$="y" THEN 50
550 IF C$="n" THEN RUN ELSE 530

```

## 5. 結 び

本稿ではPC-8001を対象として、データの伝送が可能なTSS端末用のプログラムを紹介し、さらにこのデータをマイコンで二次処理するために、データファイルを文字変数型から数値変数型へ変換するためのプログラムを添えた。

本プログラムを用いれば、TSS端末としてだけでなく、マイコン同志のデータ伝送も可能である。一方問題点として、プリンター作動時に改ページ信号があると、改ページ速度が遅いためコミュニケーションバッファがオーバーフローしてしまうことがあるので、改ページをなるべく行わないことを奨める。

今後、追加できる機能として、以下の事項が考えられる。

- ① マイコンから大型計算機へのデータの伝送。
- ② プログラムリストの送受信。これはプログラムをアスキー形式でセーブしておけば可能であり、マイコンのスクリーンエディタ機能を用いて、容易にプログラムの作成、修正ができること、TSSのサービス終了後もマイコンでプログラムを作り、サービス時に流すことができることなどが期待できるが、伝送速度が300BPSであるため長いプログラムを扱う場合には不便であろう。

プログラムの作成にあたり那須正夫氏(大阪大学薬学部)、藤井博氏(大阪大学大型計算機センター)の助言を得た。また本稿をまとめるにあたり、井上一朗氏(大阪大学工学部建築工学科)の指導を得た。末筆ながら御礼申し上げます。