

### 3. 開発報告

「PC-8800/PC-8000 シリーズによるテキスト編集  
およびTSSインテリジェントターミナルのためのプログラム」

工学部 清木 泰式

薬学部 芳本 忠

#### 1. まえがき

パーソナルコンピュータ（もしくはマイコン）をTSSインテリジェントターミナルとして使用するためのプログラムは、これまでに報告されており（注1）、本学のM-180IIADシステムとの接続に関しても、その試みがすでになされている（注2）。しかし、これらのプログラムを移植するには、ホストコンピュータもしくはターミナルとなるパーソナルコンピュータの機種の違いにより、意外に労力と時間を要する。

そこで、学内でも比較的多くのユーザーをもつPC-8800/PC-8000 シリーズのパーソナルコンピュータ（以下PCと略す）を対象とした、TSSインテリジェントターミナルのためのプログラム作りを試み、満足のいく結果が得られたのでここに報告したい。これにより、専用ターミナルをもたない学内および学外のユーザーにとって、本学の大型コンピュータが身近なものとして利用可能になる他、データの送信および受信をディスクを介して行うことができるようになる。また、ホストからのデータやプログラムを変換し、PCのすぐれたスクリーンエディット機能によるテキスト編集を可能にするための変換プログラムも用意した。これにより、ソースプログラムおよびデータの作成および編集をPC側で行うことができ、ホスト利用の効率を高めることができる。

以下に、本プログラムの特徴を示す。

(1) TSSのセッション開始を自動化した。すなわち、BREAK信号送出と共に、課題番号およびパスワードを自動的に送信する。また、このとき画面への表示は行われないので、課題番号およびパスワードを他人に知られずにすむ。

(2) キーボードおよびディスクのいずれからも入力および送信が可能であり、またプリンタおよびディスクのいずれへも出力および受信が可能である。これらの機能は、ファンクションキーにより任意に選択および切り換えができるようになっている。

(3) PCの特徴の1つであるカラーグラフィックを活かして、上記の機能の使用状態を常に画面に表示させるなど、見やすさと操作性を高めるための工夫がなされている。

(4) 本プログラムは伝送速度300ボーを標準に組まれたものであるが、筆者らの実験した限り、1200ボーでの使用も可能であることがわかった。

本プログラムはPC-8000（PC-8011との組み合わせ）およびPC-8800の各シリーズについて個別に作成されているが、いずれもすべて

BASICで書かれておりしかも同等仕様であるので、ユーザーの用途にあわせた自由な変更や移植が可能である。

## 2. RS-232C 使用上の注意 (PC-8801/PC-8011)

PC-8801およびPC-8011 (拡張ユニット) のRS-232Cは、いずれも受信用のバッファをもっており、着信データは割り込み処理によりこの受信バッファに入った格納され、読み出しコマンドに応じてバッファから取り出される。PC-8801においては、BASICコマンド用の割り込み処理が用意されており、受信時の割り込み発生に応じて、BASICレベルでのRS-232C割り込みサブルーチンを働かすことも可能である。この機能により、メインルーチンとは独立の受信データ処理を行うことができ、バッファからのデータ読み出しを高速化できる。PC-8011には、BASICレベルでのRS-232C割り込み処理コマンドはないので、メインルーチンのサブルーチンとして呼び出さねばならない。以下にPC-8801およびPC-8011の各RS-232Cチャンネル使用上の相違点や特徴を述べてみる。

2.1 初期化 本学のM-180IIADに対する伝送モードは、次のように定められている。キャラクタ長 (7ビット)、ボーレート (X16)、偶数パリティ、ストップビット数 (1)、非同期モード。これをモードバイトで表わせれば&H7Aとなる。一方、コマンドバイトは、送受信 (可)、データ・ターミナル (レディ)、センド要求 (オン)、エラーリセットにより&H37となる。PC-8011のINIT%文は、モードバイトおよびコマンドバイトの2つを同時に指定できるので、RS-232Cポートの初期化はINIT%1、&H7A、&H37により行われる。%1はチャンネル1を使用することを表わす。これに対して、PC-8801の場合、RS-232Cはファイルイメージで扱われており、その初期化も上記と同じモード設定に対して、OPEN"COM: E71XS"AS#3で行われる。ここに、Eは偶パリティ、7はキャラクタ長、1はストップビット数、XはボーレートのXパラメータが有効、SはSパラメータが有効 (7bitモードでカナの送受信が可能) を表わす。また、COM:はRS-232Cのデバイス名を表わし、#3はRS-232C用のバッファとして、ファイル#3を使用することを表わしている。なお、PC-8801ではコマンドバイトの初期化は行わなくともよい。

2.2 データの送信 まず、ホストに対する割り込みのためのブレイク信号を送らなければならない。RS-232C用のポートはPC-8011の場合&HC1、PC-

8801の場合&H21である。このポートにOUT文を実行することにより、コマンドバイトを変更することができる。ブレイク信号送出のためのコマンドバイトは &H37+8により&H3Fであるから、OUT&HC1、&H3F (OUT&H21、&H3F) とすることにより、ブレイクコードが送出される。コマンドバイトをもとにもどすには、OUT&HC1、&H37 (OUT&H21、&H37) とすればよい。

キャラクタを送るためのコマンドとしては、print%1、B\$ [;] (PC-8011)、print#3、B\$ [;] (PC-8801)がある。";"の使い分けは通常のPRINT文の場合と同様である。復帰コードのみを送るときは、print%1、もしくはprint#3、とすればよい。最後の", "を付けなければ syntax errorとなる。

1200ボーでデータ送信を行う場合、print%1、D\$とする代わりに、for j=1 to len(D\$):print%1,mid\$(D\$,j,1);:next:print%1、:とする方が送信エラーが少ないようである。print#3、D\$についても同様である。

2.3 データの受信 先に述べたように、PC-8011およびPC-8801のいずれについても、RS-232Cの受信データは割り込み処理によりバッファに蓄えられ、読み出しコマンドにより取り出されるので、次々に入ってくる受信データが処理されずにバッファにたまると、その許容量をこえた場合、"communication buffer overflow (PC-8011)"もしくは"line buffer overflow (PC-8801)"の受信エラーが生じる。したがって、バッファ中のデータはできるだけ速やかに読み出して処理する必要がある。バッファからの読み出しコマンドとして INPUT\$(SS,%1) (PC-8011)もしくは INPUT\$(SS,#3) (PC-8801)がある。ここに、SSはバッファから取り出されるデータの個数を表わし、SS=PORT(1) (PC-8011)もしくはSS=LOC(3) (PC-8801)で与えられる。INPUT\$文の他に INPUT% (PC-8011)もしくはINPUT#、LINEINPUT# (PC-8801)がデータの読み出しに使用できるが、これらは区切り記号や復帰コードがくるまで入力待ちとなるので、TSSターミナル用としては適当でない。

PC-8801ではRS-232Cをファイルイメージで扱っているので、この他にEOFを使ってバッファ中のデータの有無を調べることもできる。

2.4 PC-8801では、RS-232Cのデータ受信に発生する割り込みをBASICコマンドレベルで利用することができる。ON COM GOSUB\*

## 開発報告

\*RSOUT 文により、RS-232Cにデータが入力された時、メインルーチンを離れてプログラムの流れは\*RSOUTへとぶ。したがって、メインルーチンで常に入力データの有無を調べるという必要はなくなる。その反面、\*RSOUTルーチンでの処理が完了しなければメインでのキー入力等が働かなくなる。また、必要に応じてこの割り込み処理を一時的に中断、停止、開始するための `com stop`、`com off`、`com on` を使い分けることが大切である。

### 3. TSSプログラムの解説 (TSS-88/TSS-80)

TSS-88 はPC-8800シリーズ用に作成されたものであり、一方、TSS-80はPC-8001とPC-8011 (拡張ユニット) との組み合わせを基本としたシステムのために作成されたものである。プログラムの内容や使用法はほぼ同等仕様になっているので、以下の説明ではTSS-88を取り上げ、付随的に TSS-80 との相違点を示していくことにする。

[1040] RS-232Cポートの初期化。

FM\$にはユーザの課題番号とパスワードを入れておく。

[1090-1130] TSSセッションのオートスタート。RS-232Cのポートにブレイク信号を送る。ホストに対する割り込みがかかると、ベルコード `chr$ (7)` および改行コード `chr$ (10)` が返ってくるので、改行コードを検出して "LOGON TSS" と共に課題番号およびパスワードを送る。

[1211] PC-8801ではファンクションキーに対する割り込み処理も可能である。プリンタやファイル送受信のスイッチとして、ファンクションキーを使用しているので、そのためのサブルーチンを宣言しておく。

[1213] RS-232C受信用のサブルーチンを割り込み処理で働かすための宣言。

以上の2つの割り込み処理はメインルーチン (\*INKEY) には含まれていない。一方、TSS-80 では、これらに相当する機能をすべてINKEYルーチンで用意しなければならない。(1260;1311-1315)

[1240-1340] \*INKEY 先に述べたように、TSS-88 においては各種の割り込み処理が可能であるので、メインルーチンの主な仕事はキーからの入力とディスクからのファイル送信のみですむ。

[1360-1390] \*BREAK RS-232Cポートへブレイク信号を送る。その他に、送受信用ファイルをクローズし、画面表示および各種パラメータの初期化を行う。

[1410-1510] \*OPENFR受信用ファイルをオープンする。ファイル名を

正しく入力すれば、ファンクションキーF・2 はファイル受信用スイッチFR-SWにおきかわる。

〔1530-1640〕 \*OPENFS 送信用ファイルをオープンする。ホストへ送るファイルは、PCのスクリーンエディット機能を利用して、各ユーザが作成することができるが、本プログラムでは、その編集形態を考慮して三種類のファイル形式を扱うことにしている。(その詳細は後で述べる)。その識別のための処理が含まれており、それぞれの形式に応じたフラグを立てている。\*OPENFR と同様、送信用ファイルオープン後は、F・3 はファイル送信用スイッチFS-SWにおきかわる。

〔1660-1700〕 \*FLCHK ファイル名の長さをチェックするルーチン。ファイル名が6文字をこえる場合、たとえばSAMPLE#は、本来ならばSAMPLE.#とすべきであるが、"."を付けなくともopenしてくれるので、そのままでもパスするようにしてある。アスキーセーブしたファイル名はファイルリストでは、SAMPLE # と"."の付かない表示になるので、SAMPLE.#とするよりはSAMPLE#とする方がユーザにとっても、使いやすいであろう。ただし、TEST.#をTEST#とすれば明らかにエラーとなる。また、ファイル名が9文字をこえる場合もエラーとなる。

〔1720-1820〕 \*FSEND ファイル送信のためのサブルーチン。このルーチンは以下に述べる\*INSKFの返す2つのフラグK, Fにより、送信のためのタイミングを判断しながらファイルのデータをRS-232Cポートへ出力している。

〔1840-1880〕 \*LNIP ソースファイルから取り出した文字列を、ファイル名に対応した方法で変換および処理をしている。

〔1930-1970〕 \*INSKF EDITモードでファイル送信を行うとき、ホストから行番号に続いて1ブランク、chr\$(0)が送られてくるので、行番号の最後の"0"とこれらの文字とを検出することにより、データ送信を行う。A\$=INPUT\$(SS,#3)により取り出されるデータの数は、プログラムの1サイクルで2, 3個であるので、そのままでは行番号の検出には用いられない。そこで、これらを検出作業用の文字変数に蓄えていき、さらにこれを改行コード以後のものにおきかえる。そして、これが所定の長さになったところで"KEQ"と"0"+chr\$(0)を検出し、それぞれに対してフラグK, Fを立てる。Fはファイル入力のルーチンへとぶためのものであり、Kは"KEQ...RECEIVED DATA ABNORMALLY-REENTER INPUT"が生じた場合の再入力のためのものである。

〔1990-2050〕 \*RSOUT〔1213〕で宣言されたRS-232C受信データの割り込み処理ルーチン。

## 開発報告

[2070-2090] \*RSERR "Line buffer overflow"のエラー (ERR=23) が生じたときの処理ルーチン。ブレークコードを送ってホストからの送信を停止させる。バッファ中の残りのデータの画面表示が完了すれば\*INKEYモードにもどる。

[2150-2180] \*TIMEファイル送信中、5秒経過してもホストから次の行番号が返ってこない場合、復帰コードを送って応答を促しているが、その時間経過を調べるためのルーチンである。PC-8801は内蔵バッテリーにより電源オフの状態でも内蔵クロックが常に働くようになっているため、これをタイマーとして利用する際、タイマリセットしたのでは不都合であるとの配慮から、このようなルーチンが用意された。PC-8801の場合はタイマリセットして `if time$="00:00:05" then……`としてもよい。

[2200-2340] f・1, f・2, f・3 用のスイッチルーチン。f・1はプリンタのためのスイッチ、f・2は\*OPENFRおよびファイル受信用スイッチ、そしてf・3は\*OPENFSおよびファイル送信用スイッチとして機能する。これらは[1211]で宣言されているように、割り込み処理ルーチンとして働く。TSS-80ではこれらはINKEYルーチンから呼び出される。

以上、主なルーチンの働きだけを説明したが、詳細はソースリストを直接解読されて確かめられたい。

### 4. TSSプログラムの使用法および諸注意 (TSS-88/TSS-80)

本節でもPC-8801のみについて、TSSプログラムの使用例を示して行くことにする。

4.1 セッションの開始 DISK起動後のファイルオープン数は3としておく。プログラムをrunさせると、セッション開始方法が2通り表示される。そのうちの1つ、オートスタートをさせるには、FM\$にユーザの課題番号とパスワードを書き込んでおく必要がある。グラフィックキーと共にxキーを押せば (これをG-xと表わす)、ブレーク信号がホストへ送られ、これが受け付けられれば"START: LOGON TSS"が表示され、画面がクリアされて図-1のような状態に変わり、ホストからのメッセージが表示されていく。G-xの代わりにRETURNキーを押せば同じく図-1の状態に変わり、キー入力が可能となる。手動でセッションを開始するには、G-xによりブレーク信号を送り、ホストからベルコード (ピーという音) と改行コード (カーソルの

下向移動) が返ってきたところで、キーにより "LOGON TSS . . ." を打ち込む。  
なおキー入力の訂正にはDELキーのみが使用できる。

4. 2 プリンタ出力 ファンクションキー f・1 はプリンタへの出力ためのスイッチである。押すたびに on/off の切り換えが行われるので、必要な箇所のみをプリントすることができる。

4. 3 ファイル受信 f・2 を押すと、受信用ファイル名を聞いてくる。ファイル名を入力すると、f・2 が FR-SW におきかわる。プリンタと同様、FR スイッチを on/off することにより、必要な箇所のみをディスクへセーブすることができる。次節で述べるファイル編集および変換の形式にもとずいて、受信用のファイル名を使い分ける必要がある。(ここで、一行番当りの桁数が 80 をこえる場合、テキストが改行コードで分離されるが、これを編集時に結合するためにも、F-形式のファイル名(後述)をつけ、LIST . . . SN で受ける方がよい)。

4. 4 ファイル送信 f・3 を押すと、送信用ファイル名を聞いてくる。ファイル名を入力すると、f・3 が FS-SW におきかわる。FS スイッチも on/off により送信を開始したり停止したりするのに用いられる。EDIT モードでファイル送信を行うには、READY モードでファイル名を入力し、例えば EDIT TEST.  
TEXT NEW と入力すれば、INPUT に続いて 00010 と行番号が返ってくる。この状態で FS-SW を on にすれば、ファイル送信を開始し、以後、行番号を検出しながら自動的に送信を続けていく。ファイルが空になったとき、\*\*\* END  
OF FILE \*\*\* を表示し、f・3 および画面表示中の送信モードパラメータをリセットし、EDIT モードにもどる。送信の途中で FS-SW を off にすれば、FSEND ルーチンから INKEY ルーチンにもどるので、キーボードからの入力が可能になる。FS-SW を on にすればファイル送信が再開される。なお、ファイル受信の場合と同様、送信ファイルを強制的にクローズするには G-x を押せばよい。

4. 5 アテンション入力 処理を途中で打ち切ったり、強制的にあるモードから抜け出すためには、G-x を押せばよい。なお、G-x により、f・1 を除く他のファンクションキーは初期設定にもどり、オープンしたファイルはすべてクローズされる。  
(RS-232C 用のファイルのみクローズされない。)

4. 6 セッションの終了 f・10 に LOGOFFC+chr\$(13) がセッ

## 開発報告

トされているので、READYモードでこのキーを押せば、TSSのセッションは終了する。メッセージがすべて打ち出されたところでG-¥ を押せば、すべてのファイル (RS-232Cも含む) がクローズされ、割り込み処理も停止し、かつファンクションキーも使用前にもどされる。ただし、f・10には run "F-CONV"+chr\$(13) がセットされるので、ただちに受信ファイル等の編集にうつることができる。

### 4. 7 諸注意

- 1) プログラムの入ったディスケットを他人にコピーさせる場合、課題番号およびパスワードを消しておくこと。
- 2) [1040] 1200 ボーでのファイル送信を行うには、Bの値を1とする。
- 3) [1140] DVの値を1とすれば、2ドライブのファイル名リストが表示される。
- 4) キー入力するコマンドやデータもプリンターへ打ち出したい時は、次の行を追加する。

```
[1335] if LP then lprint B$; これにより、PRN-SW をonにしたとき、キー入力のデータもプリント出力される。
```

- 5) ファイル送信中、紫色で表示されるテキスト部分は、実際には送信されていない。緑色の部分のみが送信されていることに注意。

## 5. ソーステキストの編集およびファイル変換 (F-CONV)

### 5. 1 ソーステキストの作成

PC-8001およびPC-8801はいずれもスクリーンエディット機能を持ち、特にPC-8801では画面のROLL UP/DOWNが可能であるほか、行番号の代わりに\*記号番地名がLISTおよびEDITで使用できるなど、PC-8001以上の強力なスクリーンエディット機能を有する。ソーステキストを作成するには、BASICのプログラムを作成する場合と同様、行番号付形式で入力していけばよいが、FORTRANや数値データなどの場合、行番号とその後続くソーステキスト中の数字との間に結合が生じる。これを避けるため、行番号のあとに'を付け、いわゆるREM文としてソーステキストを入力していく。PC-8001の場合、AUTOモード下でのプログラム作成を行い、LISTをとっても、上記のような行番号と数字との結合は起こらない。しかし、これをスクリーンエディットしたり、アスキーSAVEしたあとに再びLOADした場合には、行番号と数字が結合した新たな行が生じてしまう。しかし、PC-8001の場合、AUTOモード下で作成し、編集を行わずにそのままアスキーSAVEしたファイルの内部では、行番号と数字との結合は起こっていない。同様のことは、ホストから受信した行

番号付リストのファイルについてもいえる。従って、これらのファイルをLOADして編集可能とするためには、ソーステキストをすべてREM文に変換することが必要となる。また、ホストから受信した行番号なしリストのファイルは、そのままではLOADして編集することはできないので、同様の交換を行うことが必要である。

なお、ソーステキストを入力するには、AUTO 1000からスタートした方が、桁をそろえることができ都合がよい。1000行以上のテキストについてはAUTO 10000とする。また、すべてのソースファイルはアスキーSAVEされたものでなければならない、ということに注意する。

5. 2 ファイル変換 我々が扱うファイルは、次に示す3つの形式に限られる。

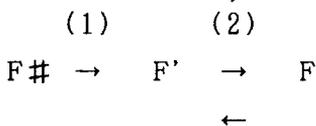
A) F#-形式 (行番号付ソースファイル、もしくは、ホストから送られてくる行番号付リストを受信するときのファイルには、#記号を含むファイル名を付ける。)

B) F'-形式 (行番号付REM文として編集用に作成されるファイルには、ファイル名に'を付ける。このファイルは主に、数値データ用ファイルや、FORTRANのようなプログラムの編集に用いられる。PL/IやPASCALプログラム等を作成するには、わざわざF'-形式としなくても、F#-形式でソースファイルを作成および編集すればよい。

C) F-形式 (行番号および'のつかない通常のソーステキストのファイル、もしくは、ホストからの行番号なしリストを受信する時のファイルには、#および'を含まないファイル名をつける。)

なお、ファイル名に#と'を混用してはならない。

これらの3つのファイル間の変換形式を、次に示す。



ここで、行番号付ソーステキストをアスキーセーブしたものを、ロードして編集に用いる場合の注意事項を述べておく。F#-形式のファイルのなかでも、ホストからの行番号付リストをREADYモードで要求すると、リストの先頭にKEQ・・・というメッセージがくることになり、これをセーブしたファイルをロードしようとしてもエラーとなる。その点、これをEDITモードで行えば、行番号付リストから始まってKEQ・・・END OF DATAで終わるので、先頭行の不要な行番号なしメッセージを除くことができる。これをセーブしたファイルをロードしてもエラーが出る場合があるが、そ

れは、ファイルの終りに、行番号なしのテキストKEQ・・・END OF DATAがあるためであり、リストをとってみればKEQ以前までの内容はロードされていることがわかる。しかし、いずれの場合にしても、変換(1)を行えば、行番号以外の箇所は無視して、行番号の箇所のみをREM文に変換してくれる。

次にF'-形式もしくはF#-形式のファイルで編集を行う際の注意事項を述べてみる。

PC-8801で編集を行う場合、その優れたエディット機能を活かすため、ソースプログラムにコメントや参照用の番地として\*記号番地を使用することを薦める。これを使えば、目的とする編集箇所へLISTやEDITで直ちに飛ぶことができる。変換(2)ではそのような\*を先頭にもつテキストはスキップするようになっている。REM文中の小文字は通常のリストのように、小文字→大文字変換は行われないので、編集の際には注意すること。なお、F'-形式が完全なものかどうかは('をつけ忘れた行番号がないかどうか)、これをRUNさせてみればよい。'が付いた箇所はBASIC文としてはスキップするが、それ以外の箇所ではBASIC文の文法エラーが生じる。

### 5.3 ファイル変換プログラムの使用法

プログラム" F-CONV"を走らせると、その使い方が表示され、入力待ちとなる。このプログラムは、アスキーセーブされたファイルの内容を調べるためのF-DISPモードと、ファイル変換を行うためのF-CONVモードの2つのモードを選択できるようになっている。TSSプログラムと同様のファンクションキー操作を行うので、扱いやすくなっている。(図-2)。

(1) F-DISPモード FC-SWをOFFの状態ではOPEN-Sを押せば、ソースファイル名を聞いてくる。ファイル名を入力すると、f・2のFC-SWがオープンしたファイル名にかわり、f・3は、F-DISPにおき変わる。この状態でF-DISPを押せば、ファイルの内容が画面表示される。F-DISPを押すたびに表示の停止、続行が繰り返される。ファイルが空になればf・3がRESETにおき変わる。RESETを押せば最初の状態に戻る。

(2) F-CONVモード FC-SWをON(\*)の状態ではOPEN-Sを押せば、F-CONVモードとなる。(画面表示がかわり、ファイル変換形式が例示される。)変換形式(1、2、3)を入力すれば、ソースファイル名を聞いてくる。それぞれの変換形式に応じた正しいファイル名を入力しなければ、エラーメッセージが表示され再入力となる。ただし、ホストからのデータをファイル受信する際、行番号付リストの受信ファイル名に#を付けることを忘れるというケースも考えられるので、変換1の場合に限り、F#をFと入力したものについてはRENAMEによる処理を用意してある。正しいファイル名を入力すれば、OPEN-SはOPEN-Dにおき変わる。OPEN-Dを押

せば目的ファイル名を聞いてくる。ここでも、正しい目的ファイルを入力しなければ、再入力もしくは強制的に修正するようになっている。F 井が目的ファイルとなることはない。これを目的ファイル名として入力した場合にのみ、再入力を促すようにしてある。その他のファイル名のエラーは、' をつけるかつけないかだけであるので、変換形式に応じてプログラムの方で修正し、その修正した新しいファイル名を表示して確認を求めてくる。目的ファイル名が正しく入力されれば、f・2のFC-SW\*はSAVEに、一方、f・3のOPEN-SはF-CONVに置き変わる。F-CONVのみをおせばファイル変換の様子を表示し、再びSAVE(f・2)、F-CONV(f・3)の状態に戻る。SAVE\*(ON)としてF-CONVをおせば、ファイル変換の内容を目的ファイルへSAVEしていき(SAVING\*)、f・2がSAVED、f・3がRESETに置き変わる。RESETをおせばモードリセットされるが、G-xでモードリセットを行うと目的ファイルをKILLするようになっているので注意すること。プログラムの終了はG-≡にて行う。

なお、画面表示と共にプリンターへも出力するには、PRN-SWをON(\*)とすればよい。

(注) 変換(1)および(2)において、紫色で表示される部分は、\*記号番地名もしくは不要テキストであり、変換の対象にはなっていない。

## 6. あとがき

本学に大型コンピュータが導入され、ユーザの数も年々増加の傾向をたどっているが、その利用時間の大部分がソースプログラムの入力および編集作業に費やされていることを考える時、時間的な制約を受けずにPCの側でソースを作成し、ディスクからファイル送信によりホストへプログラムを入力することの意義は大きいと思われる。

また、ホストとPCの間のデータ交換がディスクを介して行われるため、大量データのための高度な処理はホスト側で行い、その結果をディスクに落として、PCの側でカラーグラフィックによるデータの多彩な処理を行わせる、といった使い方も可能である。今後は、ホストとPCの複合利用を目的とするPCのユーザが増えてくるものと思われる。本報告がそのようなユーザにとって益するところがあれば、望外の喜びである。

おわりに、本プログラム作成にあたって、種々の便宜をはかって頂いたワイズ・パーソナルコンピュータの石橋氏、に感謝の意を表す。

## 開発報告

### 参考文献

- 1) 那須・多田・赤松 「PC-8000シリーズのTSSインテリジェントターミナル化」 マイコン 10月号(1981年)
- 2) 泉・小山・山田 「マイクロコンピュータによるFACOM M-180IIADへのファイルの送信および受信について」センターレポート第1号(1980年)
- 3) PC-8011 ユーザーズマニュアル (NEC)
- 4) PC-8801 ユーザーズマニュアル (NEC)
- 5) PC TECHKNOW 8000 (システムソフト)
- 6) N88-BASIC解析マニュアル (秀和システムトレーディング)

### (追記)

本報告作成後、PC-8801用のTSSインテリジェントターミナル用のプログラムがすでに発表されていることを知った。(\*) 以下に、本報告との比較検討を行ってみる。

(1) まず、操作性に関しては、本報告と同様、ファンクションキー割り込み処理を有効に用いているが、使用するキーの数が多すぎて使いにくい。また、各種キーの使用状態および使用中のファイル名等が表示されない。本報告では、基本的には、プリンタ、ファイル受信、ファイル送信およびBREAKの4つのキーで、こと足りる。また、それらの使用状態および使用中のファイル名等をすべて画面表示することにより、操作性を高めている。なお、本報告においてはキー入力時のカーソル表示も行なっている。

(2) RS-232C受信データ処理の効率がよくない。その理由として、第1に、RS-232C用割り込み処理ルーチンを使用していないということ、第2に、受信バッファからのデータの読み出しが1文字ずつしか行われていないということの2点があげられる。従って、プリンタ出力時にLINE BUFFER OVERFLOWが発生するのも当然である。本報告では、プリンタ出力およびファイル受信を同時に行っても、受信エラーは発生しなかった。

また、1200ボーでの可能性についての検討が不十分である。本報告のプログラムはすべてBASICインタプリタで書かれているが、PC-8000シリーズおよびPC-8800シリーズのいずれについても、その可能性を確かめた。(速度の遅いプリンタの使用を除いてはトラブルは全くない) その他、ファイル送信中のエラー処理が全く行われていない。本報告では、同じデータを再度送ることのほかに、応答を促すために復帰コードを送ったりしている。

(3) PCによるテキスト編集に関しては、本報告と同様、F'形式による

FORTRANプログラムの作成について簡単に述べられているが、これをそのままホストに送り、ホストのエディットモードで'を除くなど、使用法に改良の余地がある。本報告ではTSSプログラムと一体化したファイル編集形式を用意し、ファイル編集および送信を効率的に行うための配慮がなされている。

(4) 前がきでも述べた通り、同じPCを用いたプログラムでも、対象となるホストの機種が異なれば、使用条件も変わってくる。従って、既発表のプログラムをそのまま本学のホストに適用することはできない。本報告の意義は、まさに本学のホストの特殊性を考慮してプログラム作りがなされた、ということにある。

以上、本報告の長所のみを強調するような比較検討になってしまったが、既発表のプログラムはそれなりに優れた特徴を持ち、また、PC-8801用のTSSインテリジェントターミナルプログラムの先鞭をつけられたということは評価したい。

なお、既発表のプログラムの特徴の1つとして、EDITモードでの行編集機能があるが、これを参考にして本プログラムにも組み込んでみた。ここに記して、謝意を表す。

以下にその使用法を説明する。EDITモードで修正したい目的箇所をLISTする。'^'キーを押せばカーソルが右へ6桁移動し、行編集モードとなる。カーソルを目的箇所まで移動させ、訂正、削除、挿入を行い、RETURNキーを押すと、訂正テキストがホストへ送られ、カーソルが元の位置に戻る。LISTをとって訂正部分を確認する。なお、訂正できるテキストは行番号付のものに限る。80桁を超えるテキストは改行コードにより、画面上では分離して表示されているので、編集に当たっては、分離テキストを結合しておかねばならない。

\*) 藤井「パーソナルコンピュータPC-8801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 VOL. 12. NO. 2 (1982-8)

開発報告

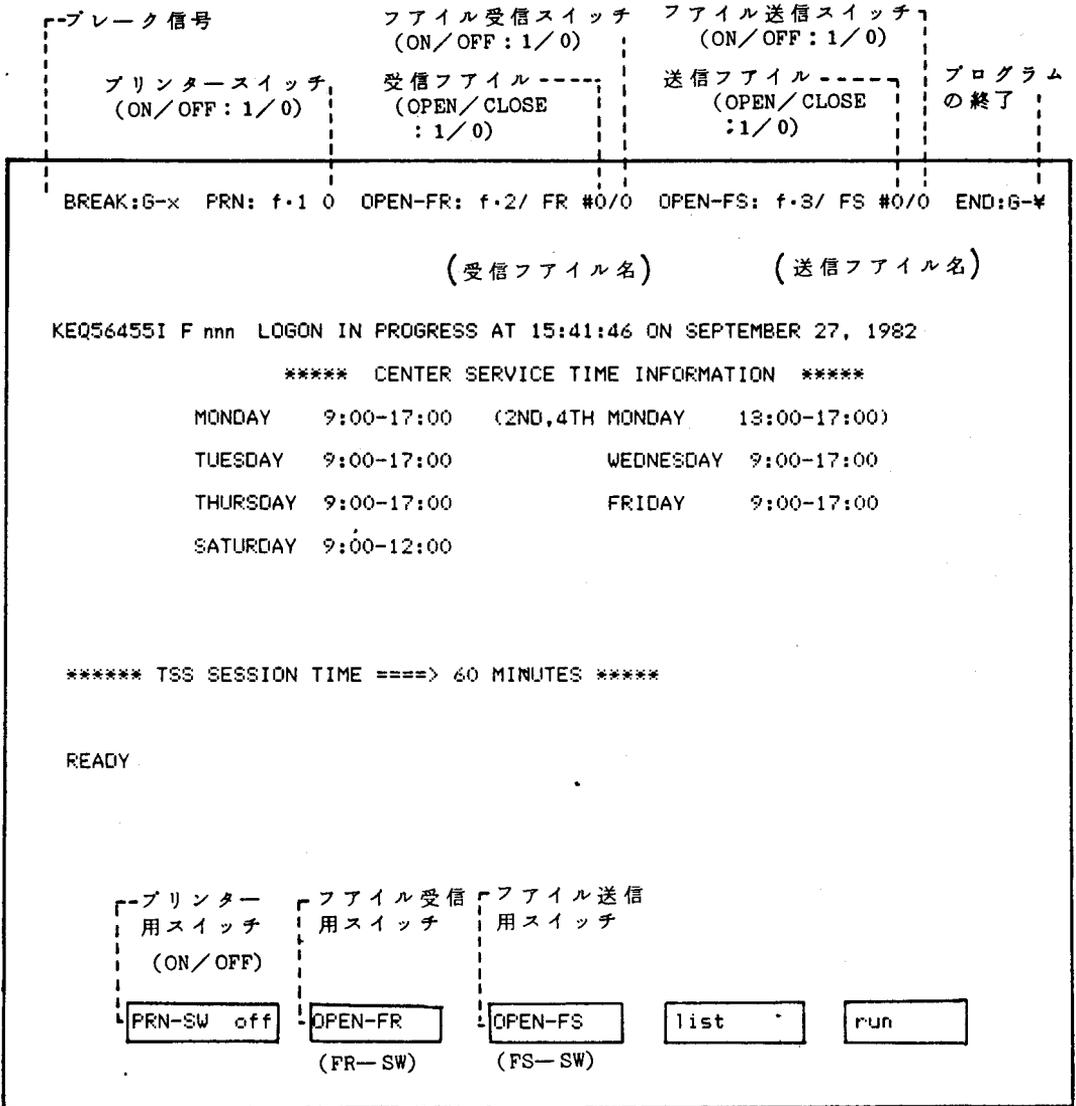


図1 TSS使用モードの画面表示

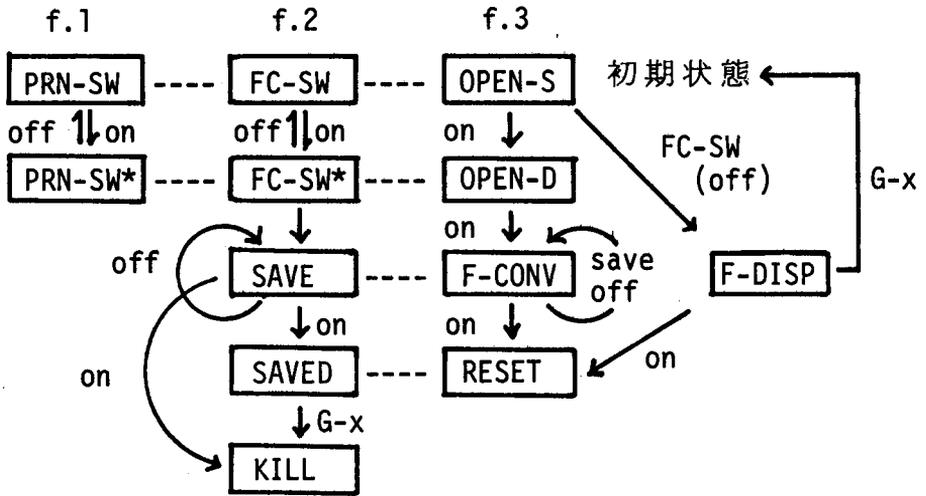


図2 ファンクションキー操作の流れ図

表I PC-8801用TSSプログラム(TSS-88)

```

1000 /
1010 / TSS Intelligent Terminal by PC-8801 ( TSS-88 )
1020 / Ver. 1.0 ( '82.8.30 ) [ Programmed by Y.SEIKI & T.YOSHIMOTO ]
1030 CLEAR 1000:WIDTH 80,20:CONSOLE 0,20,1,1:DEF USR=&H4290 :M$=""
1040 OPEN "COM:E71XS" AS #3 : FM$="Fnnnn/*****" :B=0 ( B=1 : 1200-B)
1050 COLOR 7:PRINT CHR$(12);***** PC-8801 :TSS Intelligent Terminal for FACOM
M-180 II AD *****:PRINT TAB(25);"TEL. (3606-3609)":PRINT
1060 COLOR 2:PRINT " BREAK :G-x ";:COLOR 7:PRINT " ==> ";:COLOR 3:PRINT "TSS Se
ssion Auto Start ";:COLOR 7:PRINT " ( LOGON TSS Fnnnn/PASSWORD )"
1070 COLOR 6:PRINT " Return Key ==> ";:COLOR 5:PRINT "TSS Session Manual Start
";:COLOR 7:PRINT "( BREAK---LOGON TSS Fnnnn )"
1071 PRINT:PRINT "f.1 (PRN-SW) : Printer Switch":COLOR 6:PRINT "f.2 (OPEN-FR / F
R-SW) : Open File to Receive / File Receive Switch":COLOR 4:PRINT "f.3 (OPEN-FS
/ FS-SW) : Open File to Send / File Send Switch"
1072 COLOR 7:PRINT "^ : Line Edit in EDIT-mode"
1080 /
1090 / AUTO-TSS :BREAK ==> TSS Session Auto Start
1100 B$=INKEY$:IF B$=CHR$(129) THEN POKE &HF530,ASC('*'):OUT &H21,&H3F:BEEP:POKE
&HF530,0:FOR T=0 TO 500:NEXT:OUT &H21,&H37
1110 IF B$=CHR$(13) THEN GOSUB*SBEEP:M$="PLEASE LOGON TSS!":GOTO 1140
1120 IF LOC(3) THEN SS=LOC(3):A$=INPUT$(SS,#3):LF=INSTR(A$,CHR$(10))
1130 IF LF THEN COLOR 7:PRINT:PRINT "START":;:COLOR 5:PRINT " LOGON TSS ":
BEEP:PRINT #3,"LOGON TSS "+FM$ ELSE 1100
1140 DV=0 ; : DV=0:DRIVE-1 / DV=1:DRIVE-1 & DRIVE-2
1150 COLOR 6: PRINT CHR$(12);:COLOR 2:PRINT " BREAK:G-x ";
1160 COLOR 7: PRINT "PRN: ";:COLOR 5:PRINT "f.1 ";:COLOR 7:PRINT "0 ";
1170 COLOR 6: PRINT "OPEN-FR: f.2/";:COLOR 5:PRINT " FR ";:COLOR 7:PRINT "#0/0
";
1180 COLOR 4: PRINT "OPEN-FS: f.3/";:COLOR 5:PRINT " FS ";:COLOR 7:PRINT "#0/0
";
1190 COLOR 1: PRINT "END:G-¥":LOCATE 65,18:COLOR 7:PRINT "L-EDT: ^":LOCATE 0,1
1200 CONSOLE 2,16,1,1:GOSUB *FKEY:COLOR 7:PRINT:PRINT M$,"( ;TIME$;)"
1210 /
1211 KEY (1) ON:KEY (2) ON:KEY (3) ON:ON KEY GOSUB *FK1,*FK2,*FK3
1212 /
1213 ON COM GOSUB *RSOUT:COM ON
1214 /
1220 LP=0:F1=0:F2=0:FR=0:FS=0:CL=7
1230 /
1240 *INKEY / MAIN ROUTINE
1250 ON ERROR GOTO *RSERR
1260 /
1270 IF FS THEN GOSUB *FSEND
1280 COLOR 5:B$=INKEY$:C=USR(0):IF B$="" THEN *INKEY
1290 IF B$="a" AND B$<="z" THEN B$=CHR$(ASC(B$)-32) 'small chr ==> CAP. CHR
1300 IF B$=CHR$(127) THEN PRINT #3,CHR$(8);:PRINT CHR$(29); " ";CHR$(29);:GOTO 12
80 /DEL
1305 IF B$=CHR$(94) THEN B$="":GOSUB *LEDT ' ^ :L-EDT
1310 IF B$=CHR$(129) THEN B$="":GOSUB *BREAK 'G-x
1320 IF B$=CHR$(241) THEN *ENDS 'G-¥ :end of TSS
1330 IF B$<>" THEN PRINT B$;:PRINT #3,B$;
1340 GOTO *INKEY
1350 /
1360 *BREAK
1370 COM STOP:POKE &HF3C8,ASC('*'):BEEP:CLOSE #1,2:OUT &H21,&H3F: FR$="":FS$="":
F1=0:F2=0:FR=0:FS=0:CL=7
1380 GOSUB 2270:GOSUB 2330:KEY 2,"OPEN-FR":KEY 3,"OPEN-FS":X=POS(0):Y=CSRLIN
1390 LOCATE 0,1:PRINT SPACE$(80):LOCATE 25,18:PRINT SPACE$(20);:LOCATE X,Y:OUT &
H21,&H37:POKE &HF3C8,0:COM ON:RETURN
1400 /
1410 *OPENFR (OPEN FR$ FOR OUTPUT AS #1)
1420 IF FS THEN RETURN
1430 ON ERROR GOTO 1500:X=POS(0):Y=CSRLIN:LOCATE 29,18:COLOR 7:PRINT "*":LOCATE
X,Y
1440 PRINT:COLOR 7:PRINT "DRIVE-1":COLOR 5:FILES:PRINT:IF DV THEN COLOR 7:PRINT
"DRIVE-2":COLOR 5:FILES 2:PRINT
1450 COLOR 6:FR$="":PRINT "*** File Name to Receive *** ";:COLOR 7:INPUT FR$:X$=
FR$:GOSUB*FLCHK:FR$=X$: IF BF THEN GOSUB*SBEEP:COLOR 2:PRINT "Illegal File Name!
":GOTO 1450

```

```

1460 IF FR$="" THEN GOSUB*SBEEP:COLOR 2:PRINT "*** Cancelled ! ***":X=POS(0):Y=C
SRLIN:LOCATE 29,18:PRINT " ";LOCATE X,Y:GOTO 1510
1470 IF FR$=FS$ THEN GOSUB*SBEEP:COLOR 3:PRINT "*** File Name ";:COLOR 4:PRINT F
R$;:COLOR 3:PRINT " already Used! ***":GOTO 1450
1480 OPEN FR$ FOR OUTPUT AS #1:F1=1:CL=6:GOSUB 2270
1490 X=POS(0):Y=CSRLIN:LOCATE 25,1:COLOR 7:PRINT "(";FR$;)"":KEY 2,"FR-SW off":
LOCATE 29,18:PRINT " ";LOCATE X,Y:GOTO 1510
1500 IF ERL=1480 THEN GOSUB*SBEEP:COLOR 2:PRINT "Abnormal File Name! ERR NO.=";
:COLOR 6:PRINT ERR:RESUME 1450
1510 ON ERROR GOTO 0:RETURN
1520 '
1530 *OPENFS (OPEN FS$ FOR INPUT AS #2)
1540 ON ERROR GOTO 1630:X=POS(0):Y=CSRLIN:LOCATE 43,18:COLOR 7:PRINT "*";:LOCATE
X,Y
1550 PRINT:COLOR 7:PRINT "DRIVE-1":COLOR 5:FILES:PRINT:IF DV THEN COLOR 7:PRINT
"DRIVE-2":COLOR 5:FILES 2:PRINT
1560 COLOR 4:FS$="":PRINT "*** File Name to Send *** ";:COLOR 7:INPUT FS$:X$=FS$
:GOSUB*FLCHK:FS$=X$: IF BF THEN GOSUB*SBEEP:COLOR 2:PRINT "Illegal File Name!":G
OTO 1560
1570 IF FS$="" THEN GOSUB*SBEEP:COLOR 2:PRINT "*** Cancelled ! ***":X=POS(0):Y=C
SRLIN:LOCATE 43,18:PRINT " ";LOCATE X,Y:GOTO 1640
1580 IF FS$=FR$ THEN GOSUB*SBEEP:COLOR 3:PRINT "*** File Name ";:COLOR 6:PRINT F
R$;:COLOR 3:PRINT " already Used! ***":GOTO 1560
1590 P=INSTR(FS$,".")
1600 Q=INSTR(FS$,"#")
1610 OPEN FS$ FOR INPUT AS #2:F2=2:CL=4:GOSUB 2330
1620 X=POS(0):Y=CSRLIN:LOCATE 48,1:COLOR 7:PRINT "(";FS$;)"":KEY 3,"FS-SW off":
LOCATE 43,18:PRINT " ";LOCATE X,Y:GOTO 1640
1630 IF ERL=1610 THEN GOSUB*SBEEP:COLOR 2:PRINT "File not found ":RESUME 1560
1640 ON ERROR GOTO 0:RETURN
1650 '
1660 *FLCHK
1670 BF=0:U=INSTR(X$,"."):V=INSTR(X$,"."):Y$=MID$(X$,U+1):VV=INSTR(Y$,"."):L0$=L
EFT$(X$,U):L=LEN(X$)-U+(V>0)
1680 IF V THEN 1690 ELSE 1700
1690 L1$=LEFT$(Y$,VV):L2$=MID$(Y$,VV+1):IF (LEN(L1$)=<7) AND (LEN(L2$)=<3) THEN
RETURN ELSE BF=1:RETURN
1700 L1$=LEFT$(Y$,6):L2$=MID$(Y$,7,3):IF L=<9 THEN X$=L0$+L1$+LEFT$(".",L2$,-4*(
L>6)):RETURN ELSE BF=1:RETURN
1710 '
1720 *FSEND
1730 IF K THEN K=0:R=1:GOSUB 2110
1740 IF F THEN 1770
1750 GOSUB*TIME:IF TIME=5 THEN GOSUB 2110
1760 RETURN
1770 IF EOF(2) THEN COLOR 2:PRINT "***** END OF FILE ";:COLOR 4:PRINT FS$;:COLO
R 2:PRINT " ***** ":GOSUB *CLOSEF:RETURN
1780 IF R THEN 1800 ELSE GOSUB*LNIP
1790 IF D$="" THEN 1770
1800 COLOR 4:PRINT D$;CHR$(13);
1810 IF B THEN FOR J=1 TO LEN(D$):PRINT #3,MID$(D$,J,1);:NEXT:PRINT #3, ELSE PRI
NT #3,D$
1820 F=0:R=0:F$="":RT$=TIME$:RETURN
1830 '
1840 *LNIP
1850 LINE INPUT#2,D$:I=INSTR(D$," "):C$=MID$(D$,I+1,1)
1860 IF P THEN IF C$<>" " THEN COLOR 3:PRINT MID$(D$,I+1):PRINT SPACE$(6);:D$=""
ELSE D$=MID$(D$,I+2)
1870 IF Q THEN IF C$="*" THEN COLOR 3:PRINT MID$(D$,I+1):PRINT SPACE$(6);:D$=""
ELSE D$=MID$(D$,I+1)
1880 RETURN
1890 '
1900 *CLOSEF
1910 COM STOP:CLOSE#2:FS$="":GOSUB 2110:F2=0:FS=0:CL=7:GOSUB 2330:X=POS(0):Y=CS
RLIN:LOCATE 48,1:PRINT SPACE$(30):LOCATE 38,18:PRINT SPACE$(10);:LOCATE X,Y: KEY
3,"OPEN-FS":COM ON:RETURN
1920 '
1930 *INSKF
1940 F$=F$+A$: LF=INSTR(F$,CHR$(10)):IF LF THEN F$=MID$(F$,LF+1)
1950 IF LEN(F$)>=3 THEN K=INSTR(F$,"KEQ")
1960 IF LEN(F$)>=7 THEN F=INSTR(F$,"0 "+CHR$(0))

```

## 開発報告

```
1970 RETURN
1980 '
1990 *RSOUT
2000 IF EOF(3) THEN RETURN
2010 SS=LOC(3):A$=INPUT$(SS,#3):IF FS THEN GOSUB *INSKF
2020 COLOR 7:PRINT A$;
2030 IF LP THEN LPRINT A$;
2040 IF FR THEN PRINT #1,A$;
2050 GOTO *RSOUT
2060 '
2070 *RSERR
2080 IF ERR=23 THEN OUT &H21,&H3F:GOSUB*BREAK:RESUME*INKEY
2090 RESUME *RSOUT
2100 '
2110 F$="":PRINT #3,:RT$=TIME$
2120 *SBEEP
2130 BEEP 1:FOR T=0 TO 50:NEXT:BEEP 0:FOR T=0 TO 30:NEXT:RETURN
2140 '
2150 *TIME
2160 RT=VAL(MID$(RT$,7,2)):ST$=TIME$:ST=VAL(MID$(ST$,7,2)):TIME=ST-RT
2170 IF TIME<0 THEN TIME=TIME+60
2180 RETURN
2190 '
2200 *FK1
2210 IF LP=0 THEN LP=1:KEY 1,"PRN-SW on" ELSE LP=0:KEY 1,"PRN-SW off"
2220 COM STOP:COLOR 7:POKE &HFC47,ASC('*')*LP:X=POS(0):Y=CSRLIN:LOCATE 20,0:PRINT LP:LOCATE X,Y:COM ON:RETURN
2230 '
2240 *FK2
2250 IF F1=0 THEN GOSUB *OPENFR:RETURN
2260 IF FR=0 THEN FR=1:KEY 2,"FR-SW on" ELSE FR=0:KEY 2,"FR-SW off"
2270 COM STOP:POKE &HFC55,ASC('*')*FR:X=POS(0):Y=CSRLIN:LOCATE 41,0
2280 COLOR CL:PRINT '#';MID$(STR$(F1),2);'/':COLOR 7:PRINT MID$(STR$(FR),2):LOCATE X,Y:COM ON:RETURN
2290 '
2300 *FK3
2310 IF F2=0 THEN GOSUB *OPENFS:RETURN
2320 IF FS=0 THEN FS=1:KEY 3,"FS-SW on":F=1:F$="":R=0 ELSE FS=0:KEY 3,"FS-SW off"
2330 COM STOP:POKE &HFC63,ASC('*')*FS:X=POS(0):Y=CSRLIN:LOCATE 64,0
2340 COLOR CL:PRINT '#';MID$(STR$(F2),2);'/':COLOR 7:PRINT MID$(STR$(FS),2):LOCATE X,Y:COM ON:RETURN
2350 '
2360 *FKEY
2370 '
2380 KEY 1,"PRN-SW off" 'f.1
2390 KEY 2,"OPEN-FR" 'f.2
2400 KEY 3,"OPEN-FS" 'f.3
2410 'key 4-9 : Free Setting by Users
2420 KEY 10,"LOGOFFC"+CHR$(13):RETURN
2430 '
2431 *LEDT
2432 IF FS OR LOC(3) THEN RETURN
2433 X=POS(0):Y=CSRLIN:GOSUB*SBEEP:LOCATE 6,Y
2434 C$="":COLOR 4:LINE INPUT C$
2435 IF Y<3 THEN Y=Y+2
2436 PRINT #3,C$:LOCATE X,Y-2:RETURN
2437 '
2440 *ENDS
2450 WIDTH 80,20:ON ERROR GOTO 0
2460 PRINT CHR$(12)"TSS SESSION ENDED!",(" ";TIME$;" )"
2470 FOR I=0 TO 159:POKE &HE6F2+I,PEEK(&H1B0+I):NEXT I ' Initialize F-keys
2480 KEY 10,"run"+CHR$(34)+F-CONV"+CHR$(34)+CHR$(13)
2490 CLOSE:COM OFF:KEY OFF
2500 GOSUB*SBEEP:GOSUB*SBEEP:CONSOLE 0,20,1,0:END
2510 'end of TSS-88. '83.1.5 'Ver. 1.3
```

表II PC-8001用TSSプログラム(TSS-80)

```

1000 /
1010 / TSS Intelligent Terminal by PC-8001 & PC-8011 ( TSS-80 )
1020 / Ver. 1.0 ( '82.8.30 ) [ Programmed by Y.SEIKI & T.YOSHIMOTO ]
1030 CLEAR 1000:WIDTH80,20:CONSOLE0,20,1,1:DEFUSR=&HBE2 :M$=""
1040 OUT &HE4,&HFF:INIT%1,&H7A,&H37: FM$="Fnnnn/*****" :B=0 (B=1:1200-B)
1050 COLOR7:PRINT CHR$(12);***** PC-8001 :TSS Intelligent Terminal for FACOM M
-180 II AD *****:PRINT TAB(25);TEL. (3606-3609):PRINT
1060 COLOR2:PRINT " BREAK :G-x ";;COLOR7:PRINT " ==> ";;COLOR3:PRINT "TSS Sessi
on Auto Start ";;COLOR7:PRINT " ( LOGON TSS Fnnnn/PASSWORD )"
1070 COLOR6:PRINT " Return Key ==> ";;COLOR5:PRINT "TSS Session Manual Start "
;;COLOR7:PRINT " ( BREAK---LOGON TSS Fnnnn )"
1071 PRINT:PRINT 'f.1 (PRN-SW) : Printer Switch':COLOR 6:PRINT 'f.2 (OPEN-FR / F
R-SW) : Open File to Receive / File Receive Switch':COLOR 4:PRINT 'f.3 (OPEN-FS
/ FS-SW) : Open File to Send / File Send Switch'
1072 COLOR 7:PRINT '^' : Line Edit in EDIT-mode"
1080 /
1090 /AUTO-TSS :BREAK ==> TSS Session Auto Start
1100 B$=INKEY$:IF B$=CHR$(129) THEN POKE &HF468,ASC('*'):OUT &HC1,&H3F:BEEP:POKE
&HF468,0: FORT=0T0500:NEXT:OUT &HC1,&H37
1110 IF B$=CHR$(13) THEN GOSUB2120:M$="PLEASE LOGON TSS!":GOTO 1140
1120 IF PORT(1) THEN SS=PORT(1):A$=INPUT$(SS,%1):LF=INSTR(A$,CHR$(10))
1130 IF LF THEN COLOR7:PRINT:PRINT:PRINT "START:":;COLOR5:PRINT " LOGON TSS ": B
EEP: PRINT %1,"LOGON TSS "+FM$ ELSE 1100
1140 DV=0 / DV=0:DRIVE-1 / DV=1:DRIVE-1 & DRIVE-2
1150 COLOR6: PRINT CHR$(12);;COLOR2:PRINT " BREAK:G-x ";
1160 COLOR7: PRINT "PRN: ";;COLOR5:PRINT "f.1 ";;COLOR7:PRINT "0 ";
1170 COLOR6: PRINT "OPEN-FR: f.2/";;COLOR5:PRINT "FR ";;COLOR7:PRINT "#0/0 ";
1180 COLOR4: PRINT "OPEN-FS: f.3/";;COLOR5:PRINT "FS ";;COLOR7:PRINT "#0/0 ";
1190 COLOR1: PRINT "END:G-¥":LOCATE65,18:COLOR7:PRINT "L-EDT: ^";;LOCATE0,1
1200 CONSOLE2,16,1,1:GOSUB2360:COLOR7:PRINT:PRINT M$: TIME$="00:00:00"
1210 /
1220 LP=0:F1=0:F2=0:FR=0:FS=0:CL=7
1230 /
1240 /INKEY (MAIN ROUTINE)
1250 ON ERROR GOTO 2070
1260 GOSUB 1990 /RSOUT
1270 IF FS THEN GOSUB 1720
1280 COLOR5:B$=INKEY$:C=USR(0):IF B$="" THEN 1240
1290 IF B$>="a" AND B$<="z" THEN B$=CHR$(ASC(B$)-32) 'small chr ==> CAP. CHR
1300 IF B$=CHR$(8) THEN PRINT%1,B$;PRINT CHR$(29); " ;CHR$(29);:GOTO 1280 /DEL
1305 IF B$=CHR$(94) THEN B$="":GOSUB 2431 ' ^ :L-EDT
1310 IF B$=CHR$(129) THEN B$="":GOSUB 1360 'G-x
1311 IF B$=CHR$(229) THEN B$="":GOSUB 1410 'G-r
1312 IF B$=CHR$(159) THEN B$="":GOSUB 1530 'G-s
1313 IF B$=CHR$(141) THEN B$="":GOSUB 2200 'f.1
1314 IF B$=CHR$(135) THEN B$="":GOSUB 2240 'f.2
1315 IF B$=CHR$(136) THEN B$="":GOSUB 2300 'f.3
1320 IF B$=CHR$(241) THEN 2440 'G-¥ :end of TSS
1330 IF B$<>" THEN PRINT B$;PRINT %1,B$;
1340 GOTO 1240
1350 /
1360 /BREAK
1370 POKE &HF300,ASC('*'):BEEP:CLOSE:OUT &HC1,&H3F:FR$="":FS$="":F1=0:F2=0:FR=0:
FS=0:CL=7
1380 GOSUB 2270:GOSUB 2330:GOSUB 2360:X=POS(0):Y=CSRLIN
1390 LOCATE0,1:PRINT SPACE$(79):LOCATEX,Y:OUT &HC1,&H37:POKE &HF300,0:RETURN
1400 /
1410 /OPENFR (OPEN FR$ FOR OUTPUT AS #1)
1420 IF FS THEN RETURN
1430 ON ERROR GOTO 1500:X=POS(0):Y=CSRLIN:COLOR7:LOCATE29,18:PRINT "*";:LOCATEX,
Y
1440 PRINT:COLOR7:PRINT "DRIVE-1":COLOR5:FILES:PRINT:IF DV THEN COLOR7:PRINT "DR
IVE-2":COLOR5:FILES2:PRINT
1450 COLOR6:FR$="":PRINT "*** File Name to Receive *** ";:COLOR7:INPUT FR$:X$=FR
$:GOSUB1660:FR$=X$: IF BF THEN GOSUB2120:COLOR2:PRINT "Illegal File Name!":GOTO 1
450
1460 IF FR$=""THEN GOSUB2120:COLOR2:PRINT "*** Cancelled ! ***":X=POS(0):Y=CSRLI
N:LOCATE29,18:PRINT " ";:LOCATEX,Y:GOTO 1510
1470 IF FR$=FS$ THEN GOSUB2120:COLOR3:PRINT "*** File Name ";:COLOR4:PRINT FR$;:

```

## 開発報告

```
COLOR3:PRINT ' already Used! ***':GOTO 1450
1480 OPEN FR$ FOR OUTPUT AS #1:F1=1:CL=6:GOSUB2270
1490 X=POS(0):Y=CSRLIN:LOCATE25,1:COLOR7:PRINT'(';FR$;')':LOCATE21,18:PRINT 'FR-
SW      ':KEY 2,CHR$(135):LOCATE X,Y:GOTO 1510
1500 IF ERL=1480 THEN GOSUB2120:COLOR2:PRINT'Abnormal File Name!  ERR NO.=':COL
OR6:PRINT ERR:RESUME 1450
1510 ON ERROR GOTO 0:RETURN
1520
1530 'OPENFS (OPEN FS$ FOR INPUT AS #2)
1540 ON ERROR GOTO 1630:X=POS(0):Y=CSRLIN:LOCATE 43,18:COLOR7:PRINT '*':;LOCATEX
,Y
1550 PRINT:COLOR7:PRINT 'DRIVE-1':COLOR5:FILES:PRINT:IF DV THEN COLOR7:PRINT 'DR
IVE-2':COLOR5:FILES2:PRINT
1560 COLOR4:FS$='':PRINT '*** File Name to Send *** ':COLOR7:INPUT FS$:X$=FS$:G
OSUB1660:FS$=X$: IF BF THEN GOSUB2120:COLOR 2:PRINT'Illegal File Name!':GOTO 156
0
1570 IF FS$='' THEN GOSUB2120:COLOR2:PRINT '*** Cancelled ! ***':X=POS(0):Y=CSRL
IN:LOCATE43,18:PRINT ' ':;LOCATEX,Y: GOTO 1640
1580 IF FS$=FR$ THEN GOSUB2120:COLOR3:PRINT '*** File Name ':;COLOR6:PRINT FR$;:
COLOR3:PRINT ' already Used! ***':GOTO 1560
1590 P=INSTR(FS$,')')
1600 Q=INSTR(FS$,#')
1610 OPEN FS$ FOR INPUT AS #2:F2=2:CL=4:GOSUB2330
1620 X=POS(0):Y=CSRLIN:LOCATE48,1:COLOR7:PRINT'(';FS$;')':LOCATE35,18:PRINT 'FS-
SW      ':KEY 3,CHR$(136):LOCATEX,Y:GOTO 1640
1630 IF ERL=1610 THEN GOSUB2120:COLOR2:PRINT 'File not found ':RESUME 1560
1640 ON ERROR GOTO 0:RETURN
1650
1660 'FLCHK
1670 BF=0:U=INSTR(X$, ':'):V=INSTR(X$, '.'):Y$=MID$(X$,U+1):VV=INSTR(Y$, '.'):L0$=L
EFT$(X$,U):L=LEN(X$)-U+(V>0)
1680 IF V THEN 1690 ELSE 1700
1690 L1$=LEFT$(Y$,VV):L2$=MID$(Y$,VV+1):IF (LEN(L1$)=<7) AND (LEN(L2$)=<3) THEN
RETURN ELSE BF=1:RETURN
1700 L1$=LEFT$(Y$,6):L2$=MID$(Y$,7,3):IF L=<9 THEN X$=L0$+L1$+LEFT$('.','+L2$, -4*(
L>6)):RETURN ELSE BF=1:RETURN
1710
1720 'FSEND
1730 IF K THEN K=0:R=1:GOSUB2110
1740 IF F THEN 1770
1750 GOSUB2150:IF TIME=5 THEN GOSUB2110
1760 RETURN
1770 IF EOF(2) THEN COLOR2:PRINT '***** END OF FILE ':;COLOR4:PRINT FS$;:COLOR2
:PRINT ' ***** ':GOSUB1900:RETURN
1780 IF R THEN 1800 ELSE GOSUB 1840
1790 IF D$='' THEN 1770
1800 COLOR4:PRINT D$:CHR$(13);
1810 IF B THEN FOR J=1 TO LEN(D$):PRINT %1,MID$(D$,J,1);:NEXT:PRINT %1, ELSE PRI
NT %1,D$
1820 F=0:R=0:F$='':RT$=TIME$:RETURN
1830
1840 'LNIP
1850 LINEINPUT#2,D$:I=INSTR(D$, ' '):C$=MID$(D$,I+1,1)
1860 IF P THEN IF C$<>' ' THEN COLOR3:PRINT MID$(D$,I+1):PRINT SPACE$(7);:D$=' '
ELSE D$=MID$(D$,I+2)
1870 IF Q THEN D$=MID$(D$,I+1)
1880 RETURN
1890
1900 'CLOSEF
1910 CLOSE#2:FS$='':GOSUB2110:F2=0:FS=0:CL=7:GOSUB2330:X=POS(0):Y=CSRLIN:LOCATE
48,1:PRINT SPACE$(30):GOSUB2400:LOCATEX,Y:RETURN
1920
1930 'INSKF
1940 F$=F$+A$: LF=INSTR(F$,CHR$(10)):IF LF THEN F$=MID$(F$,LF+1)
1950 IF LEN(F$)>=3 THEN K=INSTR(F$, 'KEQ')
1960 IF LEN(F$)>=7 THEN F=INSTR(F$, '0 '+CHR$(0))
1970 RETURN
1980
1990 'RSOUT
2000 IF PORT(1) THEN SS=PORT(1) ELSE RETURN
2010 A$=INPUT$(SS,%1):IF FS THEN GOSUB1930
```

```

2020 COLOR7:PRINT A$;
2030 IF LP THEN LPRINT A$;
2040 IF FR THEN PRINT #1,A$;
2050 GOTO 1990
2060 '
2070 'RSERR
2080 IF ERR=27 THEN OUT &HC1,&H3F:GOSUB1360:RESUME 1240
2090 RESUME 1990
2100 '
2110 F$="":PRINT %1,:RT$=TIME$
2120 'SBEEP
2130 BEEP1:FOR T=0 TO 50:NEXT:BEEP0:FOR T=0 TO 30:NEXT:RETURN
2140 '
2150 'TIME
2160 RT=VAL(MID$(RT$,7,2)):ST$=TIME$:ST=VAL(MID$(ST$,7,2)):TIME=ST-RT
2170 IF TIME<0 THEN TIME=TIME+60
2180 RETURN
2190 '
2200 'FK1
2210 IF LP=0 THEN LP=1 ELSE LP=0
2220 COLOR7:POKE &HFB7F,ASC('*')*LP:X=POS(0):Y=CSRLIN:LOCATE 20,0:PRINT LP:LOCATE
E X,Y:RETURN
2230 '
2240 'FK2
2250 '
2260 IF FR=0 THEN FR=1 ELSE FR=0
2270 POKE &HFB8D,ASC('*')*FR:X=POS(0):Y=CSRLIN:LOCATE 41,0
2280 COLOR CL:PRINT'#';MID$(STR$(F1),2);'/':COLOR7:PRINT MID$(STR$(FR),2):LOCATE
EX,Y:RETURN
2290 '
2300 'FK3
2310 '
2320 IF FS=0 THEN FS=1:F=1:F$="":R=0 ELSE FS=0
2330 POKE &HFB9B,ASC('*')*FS:X=POS(0):Y=CSRLIN:LOCATE 64,0
2340 COLOR CL:PRINT'#';MID$(STR$(F2),2);'/':COLOR7:PRINT MID$(STR$(FS),2):LOCATE
EX,Y:RETURN
2350 '
2360 'FKEY
2370 X=POS(0):Y=CSRLIN
2380 KEY 1,CHR$(141):LOCATE6,18:COLOR7:PRINT "PRN-SW"; 'f·1
2390 KEY 2,CHR$(229):LOCATE21,18:COLOR6:PRINT "OPEN-FR"; 'f·2
2400 KEY 3,CHR$(159):LOCATE35,18:COLOR4:PRINT "OPEN-FS"; 'f·3
2410 'key 4-9 : Free Setting by Users
2420 KEY 10,"LOGOFFC"+CHR$(13):LOCATEX,Y:RETURN
2430 '
2431 'LEDT
2432 IF FS OR PORT(1) THEN RETURN
2433 X=POS(0):Y=CSRLIN:GOSUB2120:LOCATE 6,Y:C$="":COLOR4:LINEINPUT C$
2434 IF X<1 THEN X=X+1
2435 IF Y<3 THEN Y=Y+2
2436 PRINT %1,C$:LOCATEX+1,Y-2:RETURN
2437 '
2440 'ENDS
2450 WIDTH80,20:ON ERROR GOTO 0
2460 PRINT CHR$(12)"TSS SESSION ENDED!",<";TIME$;" )"
2470 FOR I=0 TO 159:POKE &HEA7C+I,PEEK(&H3A81+I):NEXT ' Initialize F-keys
2480 KEY 10,"run"+CHR$(34)+"F-CONV"+CHR$(34)+CHR$(13)
2490 CLOSE:REMOVE:MOUNT1:IF DV THEN MOUNT2
2500 GOSUB2120:GOSUB2120:CONSOLE0,20,1,0:END
2510 'end of TSS-80. '83.1.5 Ver. 1.3

```

表Ⅲ ファイル変換プログラム (F-CONV)

```

1000 /
1010 / File Conversion & File Display Program for PC-8801/PC-8001 (F-CONV)
1020 / Ver.1.0 ( '82.8.30 ) [ Programmed by Y.SEIKI & T.YOSHIMOTO ]
1030 ON ERROR GOTO 0:CONSOLE 0,20,1,1:PRINT CHR$(12);
1040 CLEAR 1000:WIDTH 80,25:COLOR 7:PRINT "***** DISK FILE UTILITIES ( F-CONV
& F-DISP ) *****:CONSOLE 2,20,1,1:PRINT :PRINT
1050 DV=0 ;DV=0:DRIVE-1 / DV=1:DRIVE-1 & DRIVE-2
1060 COLOR 6:PRINT "FC-SW (off)":COLOR 7:PRINT " & ":COLOR 4:PRINT "OPEN-S":
COLOR 7:PRINT " ==> ":COLOR 4:PRINT "File Display Mode"
1070 COLOR 7:PRINT "F-DISP*(f-3) ..... File Display Start":PRINT
1080 COLOR 6:PRINT "FC-SW *( on)":COLOR 7:PRINT " & ":COLOR 4:PRINT "OPEN-S":
COLOR 7:PRINT " ==> ":COLOR 3:PRINT "File Conversion Mode"
1090 COLOR 6:PRINT "OPEN-D (f-3) ..... Open Destination File"
1100 COLOR 3:PRINT "F-CONV*(f-3) ..... File Conversion Start"
1110 COLOR 5:PRINT "SAVE *(f-2) ..... Save Destination File":PRINT
1120 COLOR 7:PRINT "RESET (f-3) ..... Mode Reset"
1130 COLOR 2:PRINT "G-x ..... KILL (Dest. File) & Mode Reset"
1140 COLOR 1:PRINT "G-# ..... END & F-key Setting to Edit File":PRINT
1150 COLOR 7:PRINT :PRINT "FILES ON DRIVE-1":IF DV THEN PRINT " & DRIVE-2"
1160 GOSUB 2390
1170 /
1180 LP=0:C=0:D=0:DD=0:F=0:G=0:R=0:S=0:SS=0
1190 /
1200 / INKEY
1210 B#=INKEY$:X=POS(0):Y=CSRLIN:LOCATE 0,0:PRINT " ";LOCATE X,Y
1220 IF B#=CHR$(241) THEN CLOSE:GOTO 2780 'G-#
1230 IF B#=CHR$(129) THEN GOSUB 2360:CLOSE:GOSUB 2310:GOTO 1000 'G-x
1240 IF B#="P" AND F THEN GOSUB 2460
1250 IF B#="C" AND D THEN B#="":GOSUB 2550
1260 IF B#="C" THEN GOSUB 2500
1270 IF B#="F" AND R THEN 1000
1280 IF B#="F" AND D THEN B#="":GOSUB 2610
1290 IF B#="F" AND F THEN B#="":GOSUB 1740
1300 IF B#="F" THEN GOSUB 1340
1310 IF G THEN GOSUB 1960
1320 X=POS(0):Y=CSRLIN:LOCATE 0,0:COLOR 6:PRINT "*":LOCATE X,Y:GOTO 1210
1330 /
1340 / OPENS
1350 PRINT CHR$(12):GOSUB 2390:X=POS(0):Y=CSRLIN:LOCATE 42,23:PRINT "*":IF C TH
EN LOCATE 28,23:PRINT "*"
1360 LOCATE X,Y
1370 IF C=0 THEN LOCATE 0,0:COLOR 4:PRINT "***** FILE DISPLAY *****":SPACE$(
30):PRINT : GOTO 1500 ELSE LOCATE 0,0:COLOR 3:PRINT "***** FILE CONVERSION (
):F ( ) ==> F ( ) ***** ":PRINT
1380 COLOR 4:LOCATE 0,2:PRINT "
1390 PRINT " 110 READ(5,1) | 110 READ(5,1) | READ(5,1) | ":COLOR 7:PRINT "
(2)":COLOR 4:PRINT " | READ(5,1) |
1400 PRINT " 120 1 FORMAT( | ":COLOR 7:PRINT "(1)":COLOR 4:PRINT " 120
1 FORMAT( | ==> | 1 FORMAT( |
1410 PRINT " 130 WRITE(5,10| ==> 130 WRITE(5,10| <== | WRI
TE(5,10|
1420 PRINT " 140 10 FORMAT( | 140 10 FORMAT( | ":COLOR 7:PRINT "
(3)":COLOR 4:PRINT " | 10 FORMAT( |
1430 PRINT "
1440 COLOR 6:PRINT " Source Program Source Program Or
iginal Source
1450 PRINT " with Line Number with Line Num. & Pr
ogram
1460 PRINT " ( File Name with # ) ( File Name with ' )"
1470 /
1480 PRINT
1490 M=0:COLOR 7:INPUT "Which conversion (1,2,3) ";M:IF M=0 THEN GOSUB 2360:GOT
O 1000 ELSE IF M>3 THEN GOSUB 2360:GOTO 1490
1500 COLOR 7:PRINT:PRINT "DRIVE-1":COLOR 5:FILES:PRINT:IF DV THEN COLOR 7:PRINT
"DRIVE-2":COLOR 5:FILES 2:PRINT
1510 F1#="":F2#="":IF M=1 THEN F1#="#":F2#=""
1520 IF M=2 THEN F1#=""

```

```

1530 IF M=3 THEN F2$=""
1540 ON ERROR GOTO 2660
1550 FS$="":COLOR 4:PRINT:PRINT "Source File Name ( F';F1$;' ) is ";:COLOR 7:INP
UT FS$: X$=FS$:Z$=F1$:GOSUB 1890:FS$=X$:PRINT :IF BF THEN GOSUB 2360:COLOR 2:PRI
NT "Illegal File Name!":GOTO 1550
1560 IF FS$="" THEN GOSUB 2360:IF C THEN COLOR 2:PRINT "Cancelled!":PRINT:GOTO 1
490 ELSE 1000
1570 IF C=0 THEN 1680
1580 B1$="Bad File Name ":B2$="(without '+F1$+')":B3$="":COLOR 3
1590 P=INSTR(FS$,"'"):IF P THEN B3$="(with ')"
1600 Q=INSTR(FS$,"#"):IF Q THEN B3$="(with #)"
1610 IF (M=1)*P OR (M=2)*Q THEN GOSUB 2360:PRINT B1$;B2$;' and ';B3$;'!':GOTO 15
50
1620 IF M=3 AND (P OR Q) THEN GOSUB 2360:PRINT B1$;B3$;'!':GOTO 1550
1630 IF (M=1 AND Q=0) OR (M=2 AND P=0) THEN GOSUB 2360:PRINT B1$;B2$;'!' ELSE 16
80
1640 IF M=1 AND Q=0 THEN 1650 ELSE 1550
1650 COLOR 5:PRINT:PRINT "Rename ";:COLOR 7:PRINT FS$;:COLOR 5:PRINT ' as ';:COLO
R 4:PRINT W$;:COLOR 5:PRINT ". (Yes/No :f.2/RET) ";
1660 A$="":INPUT A$:IF A$="C" THEN 1670 ELSE 1550
1670 FR$=FS$:FS$=W$:COLOR 7:PRINT:PRINT " New Source File Name is ";:COLOR 4:PRI
NT FS$ :PRINT:NAME FR$ AS FS$ :COLOR 7
1680 OPEN FS$ FOR INPUT AS #1:F=1
1690 X=POS(0):Y=CSRLIN
1700 IF C=0 THEN 1720
1710 LOCATE 25,0:COLOR 7:PRINT MID$(STR$(M),2);:LOCATE 29,0:COLOR 4:PRINT F1$;:L
OCATE 31,0:PRINT MID$(FS$,U+1);:LOCATE 48,0:COLOR 6:PRINT F2$;:LOCATE 34,23:PRIN
T "OPEN-D ";:LOCATE X,Y:RETURN
1720 D=1:LOCATE 19,23:COLOR 4:PRINT USING "(&
&)" :MID$(FS$,U+1);:LOCATE 3
4,23:COLOR 7:PRINT "F-DISP ";:LOCATE X,Y:KEY 2,"":RETURN
1730
1740 'OPEND
1750 F2$="":IF M=1 OR M=3 THEN F2$=""
1760 N=1000:X=POS(0):Y=CSRLIN:LOCATE 42,23:PRINT "*":LOCATE X,Y
1770 ON ERROR GOTO 2730
1780 FD$="":PRINT:COLOR 6:PRINT "Destination File Name ( F';F2$;' ) is ";:COLOR
7:INPUT FD$:X$=FD$:Z$=F2$:GOSUB 1890:FD$=X$: PRINT:IF BF THEN GOSUB 2360:COLOR 2
:PRINT "Illegal File Name!":GOTO 1780
1790 IF FD$="" THEN GOSUB 2360:CLOSE:GOTO 1000
1800 P=INSTR(FD$,"'"):Q=INSTR(FD$,"#")
1810 IF Q THEN GOSUB 2360:COLOR 3:PRINT " Bad File Name (with #)!":GOTO 1780
1820 IF (M=1 OR M=3) AND P=0 THEN FD$=W$:GOSUB 2360:COLOR 5:PRINT "Destination F
ile Name should be with ' . (:F) ":GOTO 1840
1830 IF M=2 AND P THEN FD$=LEFT$(FD$,P-1)+MID$(FD$,P+1):GOSUB 2360:COLOR 5:PRINT
"Destination File name should be without ' . (:F) " ELSE 1850
1840 COLOR 7:PRINT:PRINT " New Destination File Name is ";:COLOR 6:PRINT FD$;:C
OLOR 7:PRINT ". O.K (Yes/No :RET/f.2) ";:A$="":INPUT A$:PRINT:IF A$="C" THEN 1780
1850 OPEN FD$ FOR OUTPUT AS #2:PRINT :D=1:DD=1
1860 X=POS(0):Y=CSRLIN:LOCATE 50,0:COLOR 6:PRINT MID$(FD$,U+1);
1870 LOCATE 20,23:COLOR 5:PRINT "SAVE ";:LOCATE 34,23:COLOR 3:PRINT "F-CONV
";:LOCATE X,Y:RETURN
1880
1890 'FLCHK
1900 BF=0:U=INSTR(X$,"."):V=INSTR(X$,"."):Y$=MID$(X$,U+1):VV=INSTR(Y$,"."):L0$=L
EFT$(X$,U):L=LEN(X$)-U+(V>0): IF V THEN 1910 ELSE 1930
1910 L1$=LEFT$(Y$,VV):L2$=MID$(Y$,VV+1):IF (LEN(L1$)=<7) AND (LEN(L2$)=<3) THEN
W$=L0$+L1$+LEFT$(L2$,2)+Z$ ELSE BF=1
1920 RETURN
1930 L1$=LEFT$(Y$,6):L2$=MID$(Y$,7,3):IF L=<9 THEN W$=L0$+L1$+LEFT$(".",L2$,-3*(
L)>=6))+Z$:X$=L0$+L1$+LEFT$(".",L2$,-4*(L)>=6) ELSE BF=1
1940 RETURN
1950
1960 'FDC
1970 X=POS(0):Y=CSRLIN
1980 IF S THEN LOCATE 20,23:COLOR 6:PRINT "SAVING ";:COLOR 7
1990 LOCATE X,Y
2000 IF EOF(1) THEN GOSUB 2200:RETURN
2010 SS=1:CL=7
2020 D$="": LINE INPUT#1,D$:I=INSTR(D$," ")
2030 IF C THEN ON M GOSUB 2100,2130,2170
2040 COLOR CL:PRINT D$

```

## 開発報告

```
2050 IF CL<>7 THEN 2000
2060 IF S THEN PRINT #2,D$
2070 IF LP THEN LPRINT D$
2080 N=N+10:RETURN
2090 '
2100 C$=MID$(D$,2,1):IF NOT (C$>='0' AND C$<='9') THEN CL=3:RETURN
2110 D$=MID$(STR$(N),2)+' '+MID$(D$,I+1):RETURN
2120 '
2130 C$=MID$(D$,I+1,1):IF C$<>' ' THEN CL=3:D$=MID$(D$,I+1):RETURN
2140 D$=MID$(D$,I+2)
2150 RETURN
2160 '
2170 CR=INSTR(D$,CHR$(13)):IF CR THEN D$=LEFT$(D$,CR-1)+MID$(D$,CR+1)
2180 D$=MID$(STR$(N),2)+' '+D$:RETURN
2190 '
2200 'CLOSEF
2210 CLOSE:PRINT :PRINT :COLOR 2:PRINT '***** END OF FILE '::COLOR 4:PRINT FS$:
:COLOR 2:PRINT ' *****':PRINT:COLOR 7:X=POS(0):Y=CSRLIN:IF S THEN LOCATE 20,23:
PRINT 'SAVED ' ';
2220 IF DD AND S=0 THEN GOSUB 2260:RETURN
2230 LOCATE 34,23:COLOR 7:PRINT 'RESET ' ';
2240 LOCATE X,Y:D=0:G=0:R=1:GOSUB 2360:GOSUB 2360:RETURN
2250 '
2260 'REOPEN
2270 OPEN FS$ FOR INPUT AS #1:OPEN FD$ FOR OUTPUT AS #2
2280 G=0:SS=0:N=1000
2290 LOCATE 42,23:PRINT ' ';;LOCATE X,Y:RETURN
2300 '
2310 'KILLD
2320 X=POS(0):Y=CSRLIN
2330 IF DD THEN LOCATE 20,23:COLOR 2:PRINT 'KILL ';;:KILL FD$
2340 LOCATE X,Y:COLOR 7:RETURN
2350 '
2360 'SBEEP
2370 BEEP 1:FOR T=0 TO 50:NEXT:BEEP 0:FOR T=0 TO 30:NEXT:RETURN
2380 '
2390 'FKSET
2400 X=POS(0):Y=CSRLIN
2410 KEY 1,"P":COLOR 7:LOCATE 6,23:PRINT "PRN-SW";
2420 KEY 2,"C"+CHR$(13):COLOR 6:LOCATE 20,23:PRINT "FC-SW";
2430 KEY 3,"F":COLOR 4:LOCATE 34,23:PRINT "OPEN-S";
2440 LOCATE X,Y:RETURN
2450 '
2460 'FK1
2470 X=POS(0):Y=CSRLIN:IF LP=0 THEN LP=1:LOCATE 14,23:PRINT '*'; ELSE LP=0:LOCAT
E 14,23:PRINT ' ';
2480 LOCATE X,Y:RETURN
2490 '
2500 'FK2
2510 IF F THEN RETURN
2520 X=POS(0):Y=CSRLIN:IF C=0 THEN C=1:LOCATE 28,23:PRINT '*'; ELSE C=0:LOCATE 2
8,23:PRINT ' ';
2530 LOCATE X,Y:RETURN
2540 '
2550 'SK2
2560 IF SS THEN RETURN
2570 X=POS(0):Y=CSRLIN
2580 IF S=0 THEN S=1:LOCATE 28,23:PRINT '*'; ELSE S=0:LOCATE 28,23:PRINT ' ';
2590 LOCATE X,Y:RETURN
2600 '
2610 'FK3
2620 X=POS(0):Y=CSRLIN
2630 IF G=0 THEN G=1:LOCATE 42,23:PRINT '*'; ELSE G=0:LOCATE 42,23:PRINT ' ';
2640 LOCATE X,Y:RETURN
2650 '
2660 'ERR1
2670 IF ERR=53 THEN BEEP:COLOR 2:PRINT 'File not found!':GOTO 2710
2680 IF ERL=1670 THEN BEEP:COLOR 2:PRINT 'File already exists!':GOTO 2710
2690 IF ERL=1680 THEN BEEP:COLOR 2:PRINT 'Abnormal File Name! ERR NO.';;:COLOR
4:PRINT ERR:GOTO 2710
2700 ON ERROR GOTO 0:GOTO 1550
```

```
2710 RESUME 1550
2720 '
2730 'ERR2
2740 IF ERL=1850 THEN BEEP:COLOR 2:PRINT 'Abnormal File Name' ERR NO.='':COLOR
6:PRINT ERR:GOTO 2760
2750 ON ERROR GOTO 0:GOTO 1780
2760 RESUME 1780
2770 '
2780 'ENDC
2790 WIDTH 80,20:ON ERROR GOTO 0:PRINT CHR$(12);
2800 KEY 1,'load'+CHR$(34) 'key 1,chr$(9) :(PC-8001)
2810 KEY 2,'auto '
2820 KEY 3,CHR$(13)+' '
2830 KEY 6,'save'+CHR$(34)
2840 KEY 7,CHR$(34)+'a'+CHR$(13)
2850 KEY 10,'run'+CHR$(34)+'F-CONV'+CHR$(34)+CHR$(13)
2860 'remove:mount1;if DV then mount2 '(PC-8001)
2870 GOSUB 2360:GOSUB 2360:CONSOLE 0,20,1,0:END
2880 'end of F-CONV '83.1.5 Ver. 1.3
```