



Title	TSSインテリジェント ターミナル ソフトウェア 操作説明書
Author(s)	濱村, 秀彦
Citation	大阪大学大型計算機センターニュース. 1986, 63, p. 35-105
Version Type	VoR
URL	<a href="https://hdl.handle.net/11094/65713">https://hdl.handle.net/11094/65713</a>
rights	
Note	

*The University of Osaka Institutional Knowledge Archive : OUKA*

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

The University of Osaka

TSS インテリジェント ターミナル ソフトウェア

操作 説明 書

A S T E R

(Advance Software for interigent TERminal)

1986年11月

大阪大学工学部精密工学科

牧之内研究室 濱村秀彦

## 序にかえて

### 精密工学科におけるインテリジェント端末の開発

大阪大学工学部精密工学教室 山縣 敬一

本書は、PC9801（NEC）またはa i - M 1 6（アイ電子測器）を大阪大学大型計算機センターのACOS-6/MVXと接続して、タイムシェアリング・システムを利用するための端末ソフトウェアの概念、機能、利用方法について説明したものである。インテリジェント端末の開発は作成者の個性と研究室の背景が反映されるものであり、いろいろな特徴をもつものがこれまでにも作成されており、これからも作成されるであろうと思われる。その意味で精密工学科牧之内研究室における開発の背景とソフトウェアの特徴と要点をここに述べる。

牧之内三郎教授の指導のもとに、ミニコンピュータを使ったインテリジェント端末の開発を最初に手がけたのは昭和46年のことであった。これは数値制御フライス盤に接続されてオンライン DNCシステムを構成し、端末からAPTプログラムを投入すると大型計算機のファイルにNC指令が作成され、これを随時端末に転送して加工が行えるようになっていた。昭和48年のセンター・ニュースに“端末計算機における回線制御について”と題する記事が掲載されたが、インテリジェント端末にまず必要であったのは伝送制御手順を実行するプログラムであった。このプロトコルはメーカーによって定められており、研究室での完全な手作りはむずかしく、基本ソフトウェアについてメーカーからの提供を受けた後、研究室の目的に応じて修正や改良を加え、手作りのプログラムを追加することによって、オンライン制御、マンマシン・インターフェースの向上、実時間適応制御システムなどの研究が行われた。

その後、センターのTSSそのものの増強に伴い、いろいろな種類のデータステーションや端末が利用可能になる。ただ、グラフィック端末についてはメモリの低価格化によるラスター・スキャン型CRTが普及するまでは、ストレージ管が比較的手軽に使える唯一のものであった。しかし、グラフィック端末の重要性は次第に認識されてくる。牧之内研究室内においても、田村坦之助教授の指導によるシステム制御技術の研究において、いろいろな情報の視覚化、制御系の評価、システム科学における諸々の概念の把握等に图形処理がしばしば使われ、より広い意味でグラフィック端末の必要性は大きくなり、グラフィック・ライブラリの開発も行われた。また精密工学科全体を通じてコンピュータ利用の要求が次第に大きくなり、当研究室に限らず、共通利用のN6300データステーションが設置され、計測データをセンターへ直接転送するマイコン利用の端末が開発されて利用されるようになった。

一方においてメーカとユーザの立場の違いから、スクリーン・エディタはメーカ指定の特定の端末で伝送制御手順の組み込まれたものでなければ使えず、また、N6300データステーションではグラフィック端末が使えないといった問題も出て来た。おそらく、大学のユーザの要求があまりにも多様であるため、メーカがネットワーク構成によってサービスを提供することもむずかしく、結局ユーザが独自で無手順端末を接続するような状況になっているのではないかと思われる。しかし、パーソナル・コンピュータの普及に伴ってユーザの要望を盛り込んだ端末が構成できるようになった。そうはいっても実際の作成は容易ではなく、当研究室でも要望が蓄積されるだけであったのだが、大学院生濱村秀彦君の手によってここに紹介するインテリジェント端末ができ上った。とくに面白い特徴を以下に示しておく。

- (1) 端末CRT上でスクリーン・エディタを使うことによってカレント・ファイルの編集ができる。
- (2) いろいろなドットサイズに対応できるグラフィック表示とクロスヘア・カーソルの位置の入力が可能。
- (3) CRT上ではグラフィック画面の分解能が640×400でも、より高分解能のハードコピーが取れる。
- (4) 大型計算機と端末の間でファイル転送が可能であり、計測データや制御用データの転送も可能。
- (5) グラフィック・データをプロッタの制御コードでファイルに保存できる。
- (6) 実行環境や周辺機器のパラメータ設定が容易に行える。

研究室ではこれを単なるパソコンのプログラムとは考えず、ai-M16を用いた柔軟な計算機制御向けのコントローラにおいて、マンマシン・インターフェースの部分にこれを使う予定である。

初心者の方でも第1章から第5章までを読めば一通りの使い方ができる。しかし、第6章から第9章までが作成者濱村の面目躍如たる所でTSSに詳しい方には是非読まれることをおすすめしたい。そして第9章を利用して更に個性的な端末を構成して頂きたい。作成者自身もそれを期待している。作成者の苦労についてはここでは述べない。研究室のメンバーはひたすらクレームをつけることで貢献した。また、大型計算機センターのスタッフの方々からはいろいろなご教示を賜った。厚く御礼申し上げる。

#### (備考)

- (1) 本ソフトウェアは大型計算機センターにおいて無償で公開しています。ソース・プログラム、オブジェクト・プログラム、マニュアルが用意されていますので利用者窓口でお問い合わせ下さい。
- (2) 本書の内容に関しては将来予告なしに変更することがあります。
- (3) 運用した結果の影響については責任を負いかねますのでご了承下さい。

# 目 次

第 1 章	端末ソフトウェアの概説	
1. 1	本プログラムの特徴	40
1. 2	本プログラムの構成	41
1. 3	本プログラムにおける特殊キー	42
1. 4	本プログラムの実行環境	42
第 2 章	センターとの接続	
2. 1	プログラムの起動方法	44
2. 2	通信回線の接続方法	44
2. 3	システムとの接続手順（ログオン手順）	45
第 3 章	センターとの会話	
3. 1	標準の会話	46
3. 2	使用できるコントロールキー	47
3. 3	使用できないコントロールキー	47
第 4 章	画面編集機能の操作方法	
4. 1	基本的なテキストの修正	48
4. 2	テキスト画面の移動	49
4. 3	カーソルの移動	50
4. 4	文字の挿入と削除	50
4. 5	プリンターの制御	51
4. 6	ヘルプメッセージの表示	51
4. 7	画面制御キーの一覧	52
第 5 章	グラフィックスの使い方	
5. 1	グラフィックスの概説	53
5. 2	図形処理ライブラリの利用	54
5. 3	図形出力例	55
5. 4	グラフィック画面のハードコピー	59
5. 5	グラフィック画面の消去	59
5. 6	グラフィックデータのファイルへの記録	59
5. 7	グラフィックス使用上の注意	60

<b>第 6 章</b>	<b>実行環境の設定</b>	
6. 1	パラメータを変更するときの注意	62
6. 2	環境設定メニューの一覧	63
6. 3	通信回線の設定	64
6. 4	会話環境の設定	66
6. 5	ファンクションキーの設定	67
6. 6	グラフィックスの設定	68
6. 7	グラフィック画面のコピーの設定	70
6. 8	ファイル転送の設定	71
6. 9	基本入出力制御コードの設定	73
6. 10	文字列を入力する上での注意	74
<b>第 7 章</b>	<b>端末用コマンド</b>	
7. 1	端末用コマンドの概説	76
7. 2	コマンドの使い方	76
7. 3	コマンドの一覧	77
7. 4	グラフィックスの操作	78
7. 5	ファイルの転送	78
7. 6	ヒストリー機能	84
7. 7	文字単位の会話	86
7. 8	その他の機能	86
<b>第 8 章</b>	<b>より高度な利用のために</b>	
8. 1	PLOTプログラム	87
8. 2	INSTALLプログラム	87
8. 3	プリンターの遠隔操作	88
8. 4	センター以外の端末として利用する	89
<b>第 9 章</b>	<b>詳細な仕様とその実現方法</b>	
9. 1	制御コードのうち解釈しているもの	89
9. 2	スクリーンエディタとその実現について	90
9. 3	グラフィックコードの解釈について	94
9. 3. 1	アルファモード	94
9. 3. 2	グラフィックモード	94
9. 3. 3	グラフィックインプットモード	96
9. 3. 4	ポイントプロットモード	96
9. 3. 5	インクリメンタルプロットモード	96
9. 4	オペレーティング・システムの利用について	97
9. 5	周辺装置との入出力について	101
<b>参考文献</b>		102

# 第1章 端末ソフトウェアの概説

## 1.1 本プログラムの特徴

本プログラムは、センターと接続するにあたり、300 ポーから 9600 ポーまでの通信速度で通信手順が無手順であることを前提としています。さらに、端末としての基本的な機能を備えていることはいうまでもなく、次のような特徴を備えています。

### (1) 画面編集可能

”②を押すとその行を送信する”という方法を採用しています。任意のプロンプトに対して入力が可能になっています。標準800行記憶していて、ページ単位で画面を切り換えることもできます。

### (2) 漢字の送受信可能

センター側は JIS 7 単位符号系、端末側はシフト JIS を採用しています。

### (3) プリンターエコー機能

送受信の文字をそのままプリンターへ出力することができます。

### (4) カラーグラフィック可能

グラフィックライブラリ (GPLTV, CPLTV, AVLlib/N6922V, AVLlib/N6922V または当研究室開発の CPLOT.C ) を用いることにより、通常の描画およびグラフィックカーソルによる入力ができます。

### (5) グラフィック画面のハードコピー機能

表示中のグラフィックを 640×400, 640×512, 1024×780, 1536×1170, 2048×1560 の中から指定された精度で出力することができます。

### (6) ファイルの送受信機能

端末OS上のファイルをセンターのカレントファイルへ転送することができます。データファイル・プログラムファイルとも送受信可能です。漢字については、JIS 7ビット単位符号系とシフト JIS の自動変換を行っています。転送は、9600 ポーでもほぼ使用可能です。

### (7) ヒストリー(交信記録)機能

センターとの間で送受信している文字またはグラフィックデータをそれぞれ分離して、端末OS上のファイルに記録することができます。グラフィックデータは、プロッターの制御コードでファイルに記録するため、研究室のプロッターをオフライン動作させることができます。

(8) ヘルプ機能

画面編集をしている時に、ヘルプメッセージを表示させることができます。

(9) プログラム実行中の環境設定

ポートレート、プリンター制御、ファンクションキーなどが容易に変更できます。

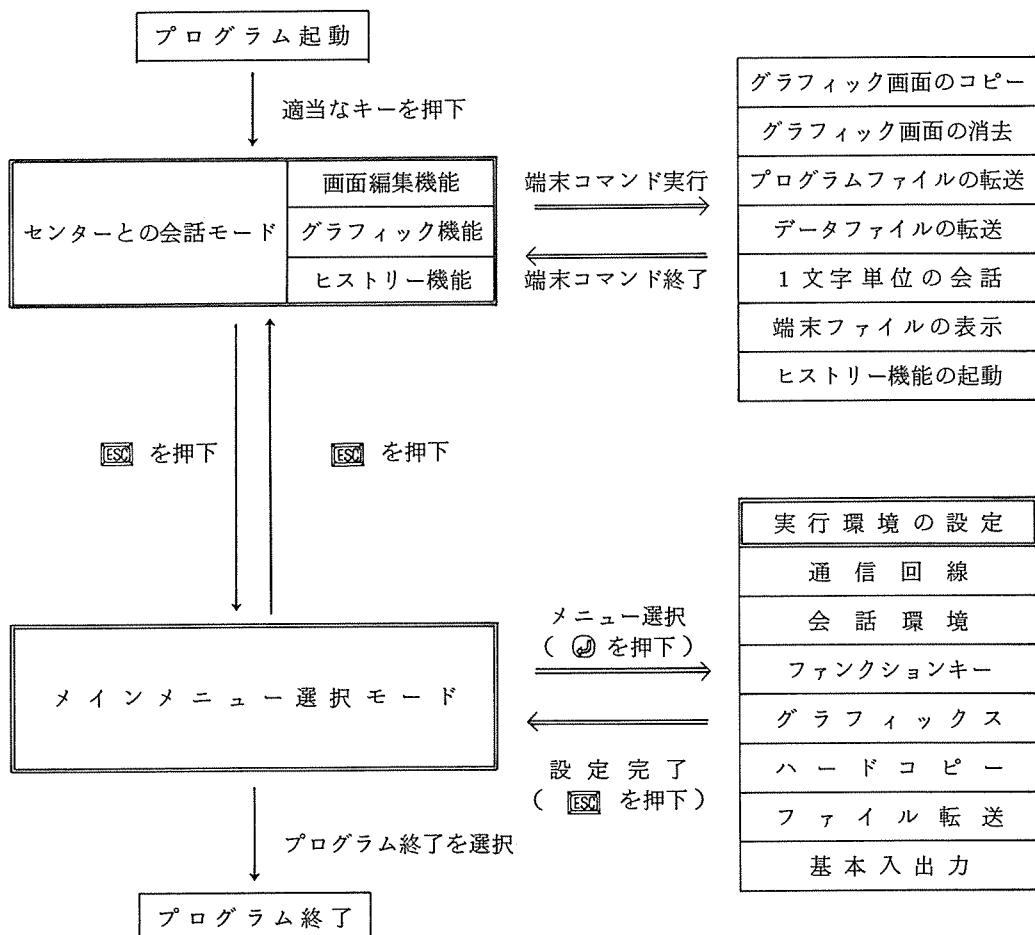
(10) 周辺機器の設定

プリンターやプロッターの機種に応じた制御コマンドを設定することができます。

(11) 初期値の変更が可能

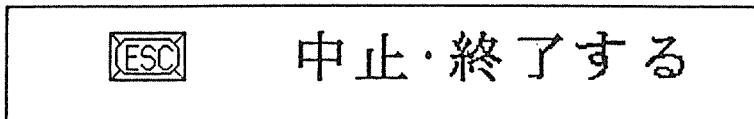
インストールプログラムを用いることにより、本プログラムを起動させた直後のパラメータを設定・変更することができます。例えば、ファンクションキーにあらかじめ、パスワードやよく使うコマンドを設定しておくこともできます。

## 1.2 本プログラムの構成



### 1.3 本プログラムにおける特殊キー

本プログラムにおいて  を押すと、いつでも現在の作業・機能を中止することができます。



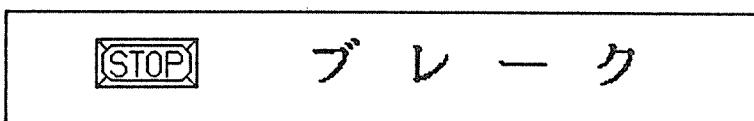
具体的には、次のように動作します。

 を押す前の状態

 を押した後の状態

センターとの会話モード	⇒	メインメニューの選択
メインメニューの選択	⇒	センターとの会話モードに戻る
サブメニューの選択	⇒	メインメニューの選択
実行環境の設定中	⇒	サブメニューの選択
ファイルの送受信中	⇒	送受信を中止して会話モードに戻る
1文字単位の会話中	⇒	センターとの会話モードに戻る
グラフィック画面のコピー中	⇒	コピーを中止して会話モードに戻る

センターに対して処理の中止を要求は、 または  +  です。



### 1.4 本プログラムの実行環境

本プログラムは、ai-M16 と PC-9801 シリーズ（XAを除く）のすべての機種で基本的に実行可能です。具体的に以下の機種でその動作を確認しています。

- ai-M16
- PC-9801 E
- PC-9801 VF
- PC-9801 VM 2
- PC-9801 VM 4

#### (1) システムの構成

PC-9801の場合、次のようなハードウェアの構成を必要とします。

- ディスプレイ 専用高解像度カラーディスプレイ ( 640×400 )
- ディスクユニット 内蔵または外付けのディスクドライブ

- プリンター                   シリアルプリンターまたは熱転写プリンター  
(ドット列印字、日本語出力が可能なものが望まれる)
- メインメモリ               256 Kバイト以上  
(全機能を活用するには640 Kバイト実装を要する)
- RS-232 C インターフェースボード  
                                内蔵されていない機種の場合必要
- 漢字 ROM ボード           JIS 第1水準漢字がオプションのものは必要  
(JIS 第2水準漢字も望まれる)
- 数値演算データプロセッサ   使用していない
- マウス・ライトペン・GP-IB・ミュージックボード  
                                サポートしていない

#### (2) ディップスイッチの設定

PC-9801の場合、正しく設定しなければならないディップスイッチがあります。機種によってスイッチ番号が若干異なりますので、以下に項目ごとに示します。

- ディスプレイの機種：                   専用高解像度ディスプレイの使用
- スーパインポーズ機能の使用：        スーパインポーズ機能を使用しない
- RS-232 C の伝送速度を決めるためのタイマ選択： 本体内蔵のタイマを使用する
- テキスト画面のキャラクタ数の指定： 80 文字 / 行、25 行 / 画面

ディスク関係のディップスイッチは、それぞれのシステム構成にあわせて適当にセットして下さい。

#### (3) メモリスイッチ

SW1, SW2

RS-232 C に関するものですが、本プログラムでは SW1 と SW2 を使用していませんので、どのように設定されても影響を受けることはありません。

SW3, SW4, SW5, SW6

システム構成にあわせて適当に設定しておいてください。ディスクとハードコピーに関するもの以外は、動作に影響を及ぼすことはありません。

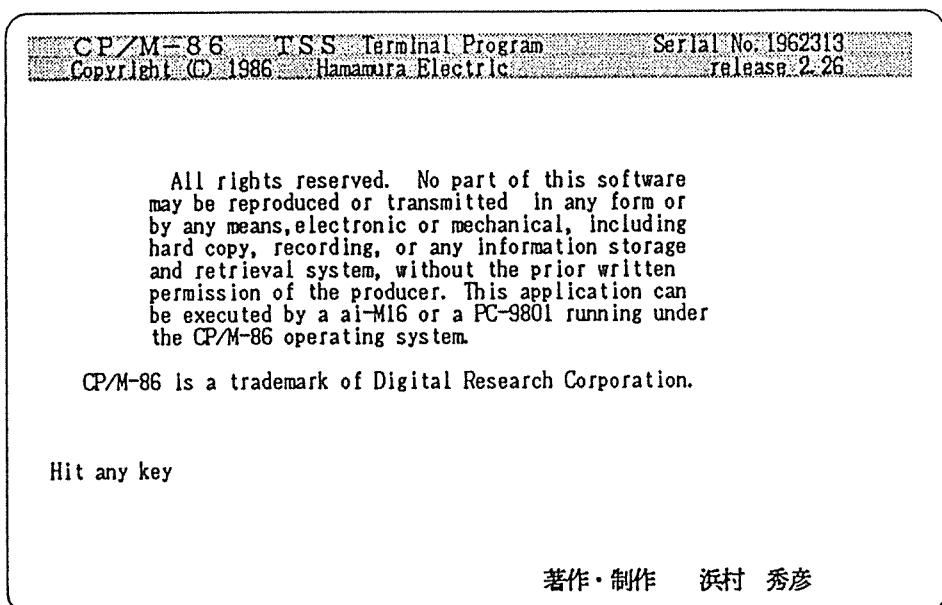
#### (4) ソフトウェア環境

PC-9801では、このプログラムを実行させるための OS として、日本語 CP/M-86 または、MS-DOS V2.11 か V3.1 が必要です。

## 第2章 センターとの接続

### 2.1 プログラムの起動方法

- (1) コンピュータ本体とディスプレイ、ディスクユニットの電源を入れます。
- (2) TSSプログラムの入ったディスクケットをセットして、CP/M-86かMS-DOSを起動します。
- (3) A>TSS②と入力して本プログラムを起動します。
- (4) しばらくして、下のような画面が出ればスタート完了です。



著作・制作 浜村 秀彦

これは、CP/M-86の場合の初期画面です。この時適当なキーを押すと、画面が消去されてセンターと会話をすることができます。

### 2.2 通信回線の接続方法

まず、以下の手順で回線を接続します。

- (1) 電話を利用している場合

モ뎀の電源を入れて、所定の電話番号を呼び出し、センターに接続したのを確認してから音響カプラに固定します。

- (2) ポートセレクター経由の専用回線を利用している場合

②を押して、「G O」のメッセージが表示されるのを確認して下さい。

- (3) 専用回線で直接接続している場合

回線は、すでに接続されていますから、なにもする必要はありません。

### 2.3 システムとの接続手順（ログオン手順）

- (1) 通信回線を接続したのち、 **[f1]** を押して下さい。すると、次のような応答が、システムからあるはずです。

```
GO  
$$CON,TSS,,KNJ@  
HANDAI TSS(R2.6) ON 03/13/86 AT 10:08:35 CHANNEL 4628  
USER ID -#
```

- (2) ここで、「USER ID-」のあとに「利用者番号；支払いコード\$パスワード」を入力します。正しく入力しますと、以下のようにサブシステムの選択をシステムが要求します。

```
GO  
$$CON,TSS,,KNJ@  
HANDAI TSS(R2.6) ON 03/13/86 AT 10:08:35 CHANNEL 4628  
USER ID -X66001;A$PASSWORD@  
<<<< ... 37313 YEN RESOURCES USED ( AVAILABLE ... 69227 YEN ) >>>>  
<<<< .... 1234 LLINKS FILE SPACE USED >>>>  
SYSTEM ?#
```

- (3) 利用者番号・パスワード・支払いコードを別々に入れても受け付けます。

```
GO  
$$CON,TSS,,KNJ@  
HANDAI TSS(R2.6) ON 03/13/86 AT 10:08:35 CHANNEL 4628  
USER ID -X66001@  
PASSWORD—  
PASSWORDPVR  
SHIHARAI-CODE ?A@  
<<<< ... 37313 YEN RESOURCES USED ( AVAILABLE ... 69227 YEN ) >>>>  
<<<< .... 1234 LLINKS FILE SPACE USED >>>>  
SYSTEM ?#
```

- (4) 以上で、センターへの接続が完了しました。

## 第3章 センターとの会話

### 3.1 標準の会話

画面編集の機能を備えていますが、通常の端末とまったく同様に使用することができます。したがって、従来から TSS を利用されている方でも違和感なく利用していただけます。ここでは、例題を用いて実際にプログラムを作成してみます。以下の指示にしたがって、図中の下線部を入力して下さい。

- (1) システム(ACOS-6/MVX)が、サブシステムの選択を要求してきたレベルから、FORTARN 77 サブシステムを呼び出します。

まず、'FRT7' と 4 文字だけ入力して下さい。次のようになるはずです。

```
SYSTEM ?FRT7
```

このとき、'SYSTEM ?'は緑色で、この緑色の文字はふつうセンター側から送信された文字であることを示しています。一方、今入力した 'FRT7' の 4 文字は水色で、この水色の文字は、キーボードから入力して、センターへ送信していないことを表わしています。

さらに、'FRT7' に続いて ② を押して下さい。

```
SYSTEM ?FRT7②  
OLD OR NEW-
```

② を押すと、'FRT7' が黄色に変わらはずです。この黄色の文字は、その文字をセンターへ送信したこと表現しています。このように、FRT7② と入力することによって FORTARN 77 サブシステムが呼び出されました。

- (2) 次に I と J を入力するとその和を計算するプログラムを作成します。

AUTOコマンドを利用してプログラムを入力し、LISTコマンドで確認して下さい。

```

SYSTEM ?PRT7②
OLD OR NEW-NEW②
*AUTO②
*0010 PRINT *, " I と J を入力して下さい"②
*0020 READ *, I, J②
*0030 K = I+J②
*0040 PRINT *, " I+J = ", K②
*0050 STOP②
*0060 END②
*0070②
*LIST②
0010 PRINT *, " I と J を入力して下さい"
0020 READ *, I, J
0030 K = I+J
0040 PRINT *, " I+J = ", K
0050 STOP
0060 END

*

```

ここで、漢字を使っていますが、英字でも別にかまいません。漢字を入力する場合、漢字の入力方法は、端末OSのマニュアルを参考にして下さい。

(3) RUNコマンドで、このプログラムを実行すると、確かにIとJの和が求まっています。

```

*RUN②
I と J を入力して下さい
I*?2,3②
I+J = 5
*
```

### 3.2 使用できるコントロールキー

以下の四つのコントロールキーはすぐに送信されて、それぞれの機能を実行します。

<b>[CTRL] + [B]</b>	処理を中断する（ブレーク機能）
<b>[CTRL] + [S]</b>	受信を一時中断する
<b>[CTRL] + [Q]</b>	受信を再開する
<b>[ESC]</b>	リターン（一行終了）
<b>[STOP]</b>	処理を中断する（ブレーク機能）

### 3.3 使用できないコントロールキー

次のコントロールキーは、画面編集のために使用しており、センターに送信していないのでその機能を利用できません。

<b>[CTRL] + [A]</b>	簡易コネクト
<b>[CTRL] + [C]</b>	簡易ディスコネクト
<b>[CTRL] + [D]</b>	1行削除

■  
[CTRL] + [N]  
[CTRL] + [O]

1 文字削除  
シスアウトモードにする  
シスインモードにする

## 第 4 章 画面編集機能の操作方法

### 4.1 基本的なテキストの修正

画面編集において、”②を押すとその行を送信する”という方法を採用しています。②を押すまでは、送信（入力）されません。例えば、修正したい行のところへ、カーソルを移動させ、修正したのち②を押せば、修正した行が送信されます。つまり、正しい行を再び入力するかわりに、修正した行を送信することによって、テキストの修正することができます。

ここでも、3章で作成したプログラムを利用して、簡単なプログラムの修正を行なってみます。I + J を計算させていたのを、I × J を計算させることを考えます。まず、LISTコマンドで、プログラムを表示します。カーソルキー（■ ■ ← →）を使って、行番号 0030 と 0040 行目の + を \* に修正して、それぞれの行で②を押すと、次のようにになります。

\* LIST②

```
0010 PRINT *, " IとJを入力して下さい"
0020 READ *, I, J
0030 K = I * J
0040 PRINT *, " I * J = ", K
0050 STOP
0060 END
```

①      ②      ③      ④

\* 0030 K = I \* J
\* 0040 PRINT \*, " I \* J = ", K
\*

①: ■と↑を使って \* に修正する  
②: ↓を押すとその行が送信される  
③: ■と↑を使って \* に修正する  
④: ↓を押すとその行が送信される

以上で、2つの行の修正ができました。LISTコマンドで確認して、実行してみます。

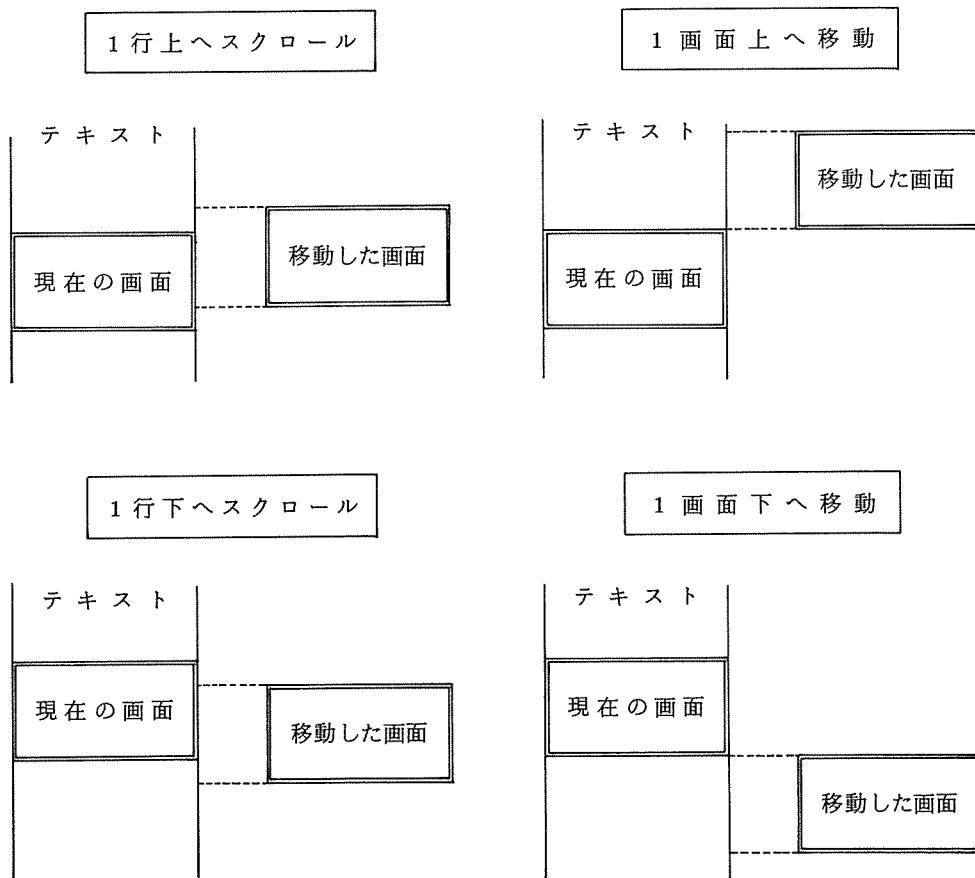
\*LIST②  
 0010 PRINT \*, "IとJを入力して下さい"  
 0020 READ \*, I,J  
 0030 K = I\*J  
 0040 PRINT \*, "I\*J =", K  
 0050 STOP  
 0060 END

\*RUN②  
 IとJを入力して下さい  
 I\*?2,3③  
 I\*J = 6  
 \*

このように、確かに修正されていて、IとJの積が計算できています。

#### 4.2 テキスト画面の移動

テキストを標準で800行記憶しているため、一度画面から消えてしまったテキストでも再び表示させることができます。そのときのテキスト上を画面が移動していく概念図を以下に図示します。



### 4.3 カーソルの移動

カーソルを移動させる操作について、以下にまとめて示します。

#### カーソルを上へ移動

[CTRL] + [R]	1画面上へ
[CTRL] + [W]	1行スクロール
[↑] or [CTRL] + [E]	1行上へ

#### カーソルを左へ移動

[←]	1文字前へ
[CTRL] + [A]	1つ左の単語へ
[CTRL] + [V]	行の先端へ

#### カーソルを右へ移動

[→] or [CTRL] + [D]	1文字後ろへ
[CTRL] + [F]	1つ右の単語へ
[CTRL] + [V]	行の最後へ

#### カーソルを下へ移動

[↓] or [CTRL] + [X]	1行下へ
[CTRL] + [Z]	1行スクロール
[CTRL] + [C]	1画面下へ

#### もう一つの移動

[CTRL] + [K]	直前に [②] を押した行へ移動
--------------	------------------

### 4.4 文字の挿入と削除

1文字挿入をすると空白が入るので、その後、挿入したい文字を入力して下さい。なお、挿入モードというのは用意されていません。

#### 1文字に関する操作

[INS] or [CTRL] + [I]	カーソルの位置へ1文字挿入
-----------------------	---------------

#### 1行に関する操作

[CTRL] + [N]	カーソルの行へ1行挿入(予定)
--------------	-----------------

文字の削除は、その操作がおよぶ範囲によって次のように、文字・行・画面に分けることができます。ここで、'消去'とは、空白でうめることであり、'削除'とは、削除してその場所をつめ

ることです。

### 1 文字に 関する 操作

<b>[ESC]</b> or <b>[CTRL]</b> + <b>H</b>	左の文字を消去
<b>[DEL]</b>	左の文字を削除(P C - 9 8 0 1)
<b>[BS]</b>	左の文字を削除(a i - M 1 6)
<b>[CTRL]</b> + <b>G</b>	カーソルの示している文字を削除

### 1 行に 関する 操作

<b>[CTRL]</b> + <b>T</b>	カーソルより行端まで消去
<b>[CTRL]</b> + <b>Y</b>	行全体を消去

### 1 画面に 関する 操作

<b>[CLR]</b> or <b>[CTRL]</b> + <b>L</b>	画面全体を消去
--	---------

#### 4.5 プリンターの制御

**[CTRL]** + **P** を押してプリンターへのエコー機能を開始しますと、それ以後、センターとの会話がすべてプリンターに出力されます。また、プリンターへのエコー機能を停止したいときには、もう一度 **[CTRL]** + **P** を押せば、停止します。

<b>[CTRL]</b> + <b>P</b>	プリンターへの出力のオン／オフ
--------------------------	-----------------

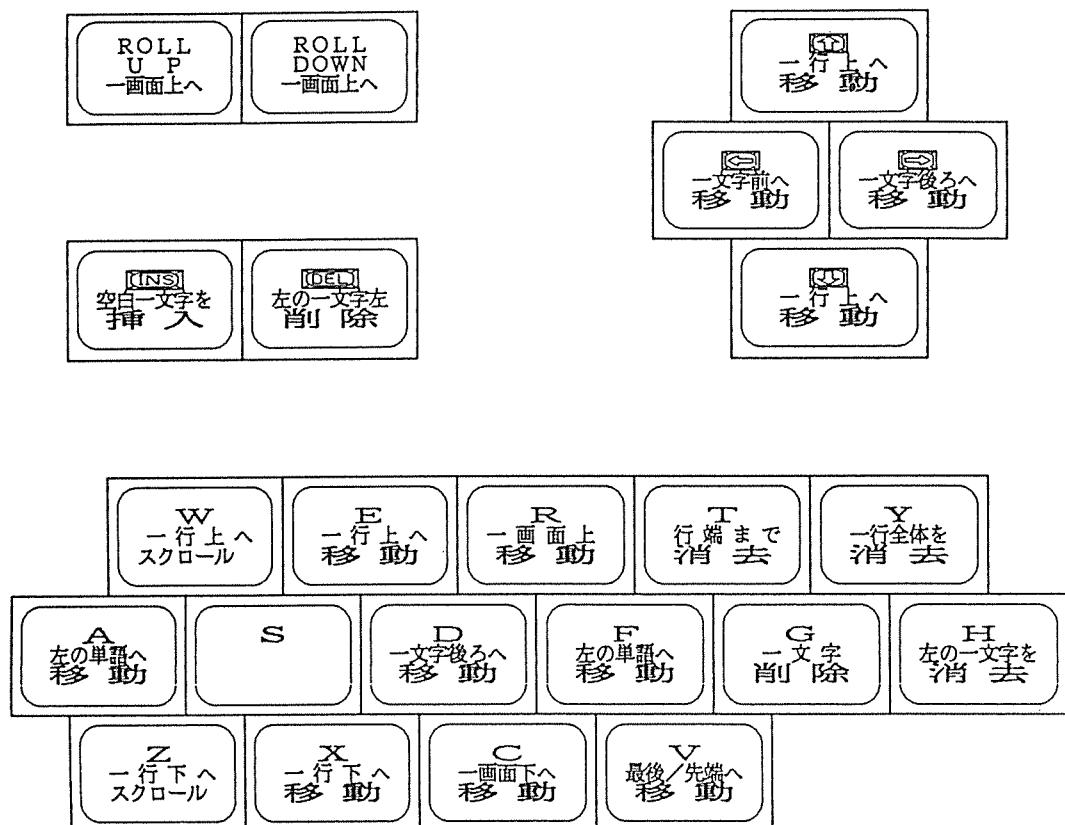
#### 4.6 ヘルプメッセージの表示

**[HELP]** または **[CTRL]** + **J** を押しますと、ヘルプメッセージを表示させることができます。表示は1ページ毎で、次のページを表示させるには再び **[CTRL]** + **J** か **[HELP]** を押してください。終了したいときは、最後まで表示させるか適当な他のキーを押して下さい。もとの画面へ戻ります。

<b>[HELP]</b> or <b>[CTRL]</b> + <b>J</b>	ヘルプメッセージの表示
---	-------------

#### 4.7 画面制御キーの一覧

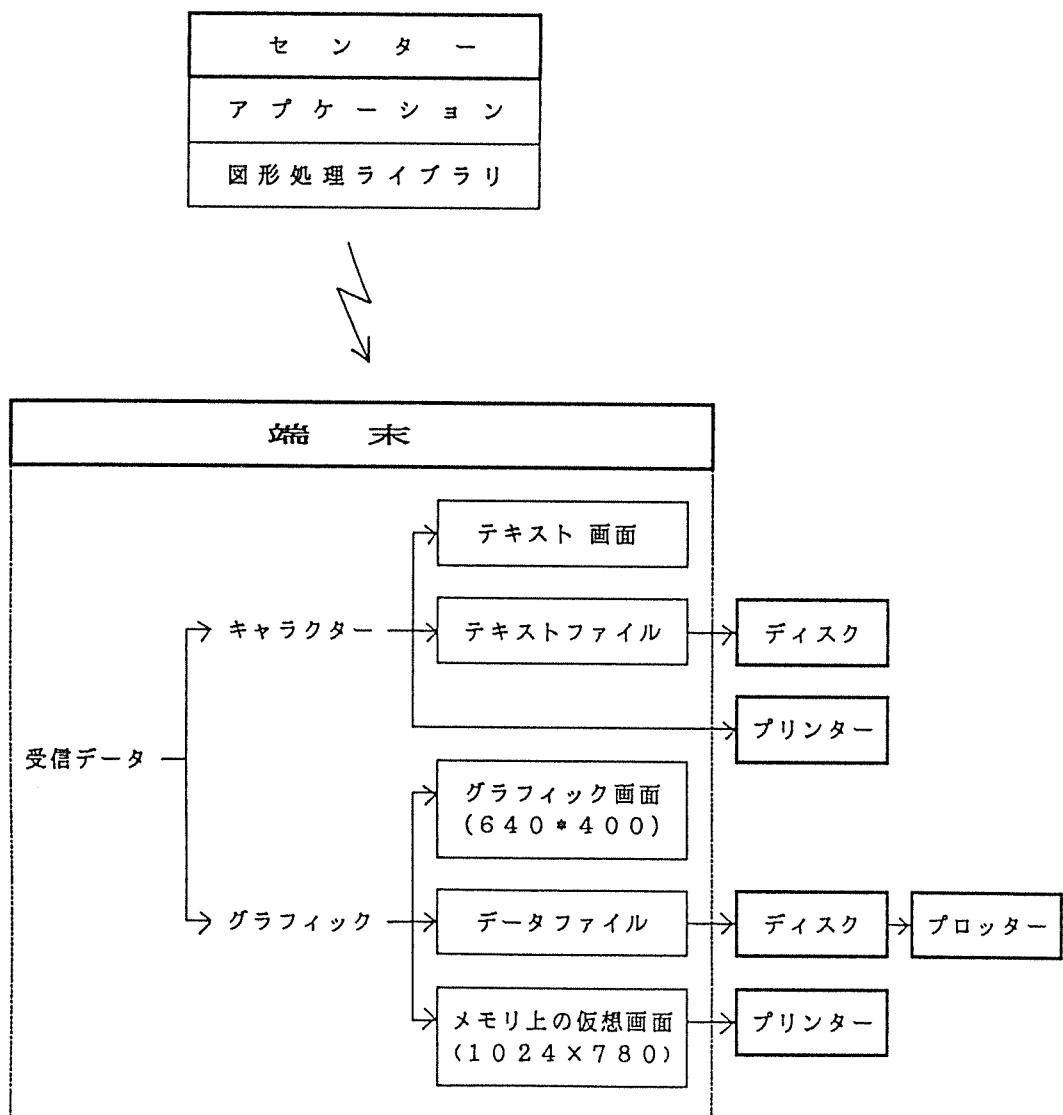
コントロールキーを鍵盤のイメージで図示しました。英文字は、**[CTRL]** を押しながら、それぞれのキーを押して使用します。



## 第5章 グラフィックスの使い方

### 5.1 グラフィックスの概説

本プログラムを使用して、図形の表示を行なう時のシステムの概要を下に示します。



#### (1) グラフィック・ディスプレイ

端末の実際の管面は  $640 \times 400$  のカラー・ディスプレイですが、グラフィックスに関するパラメータを変更することにより、より大きなドットサイズをもつ管面で作成されたプログラムでも、同じように表示させることができます。たとえば、 $1024 \times 780$  を選択して図形を表示させると  $400 / 780$  に縮小されて、実際の  $640 \times 400$  の管面に描いています。

したがって、現在利用できるグラフィック・ディスプレイのうち、下記のものとソフトウェアの互換性があります。

- $640 \times 512$  のカラー・ディスプレイ      テクトロニクス 4027型  
    日本電気 N 6940
- $1024 \times 780$  のモノクロ・ディスプレイ      テクトロニクス 4010型  
    テクトロニクス 4012型  
    日本電気 N 6921
- $4096 \times 3120$  のモノクロ・ディスプレイ      テクトロニクス 4014型  
    日本電気 N 6922

第二精工舎のD-SCAN GR-1140 や日本電気のN 6960 は、サポートしていません。

#### (2) ハードコピー

モノクロのハードコピーを、ほとんどの機種のプリンターへ出力させることができます。ハードコピーの精度は、 $640 \times 400$ ,  $1024 \times 780$ ,  $2048 \times 1560$  などの中から選択することができます。これは、図にも示したように一度メモリ上に設定された仮想画面に描画したあとプリンターへ出力するからです。カラーのハードコピーはサポートしていません。

#### (3) テキスト画面について

本システムでは、キャラクターを表示するテキスト画面は、グラフィック画面とまったく別のものであると考えています。したがって、キャラクターに関するグラフィック命令はサポートしていませんし、テキスト画面を消去してもグラフィック画面は、消去されません。たとえば、文字の大きさや輝度に関する制御をすることはできません。さらにハードコピーをとるときにもテキスト画面はプリンターへ出力されません。そのため、グラフィック画面へ文字を出力させるには、統合化ライブラリのKANJIサブルーチンのように、文字を線画にしてグラフィック画面上に描く必要があります。当研究室では、GDSP-6/PLOTを使用しているユーザーのために、ASCII文字とカタカナを描くサブルーチンを用意しています。

## 5.2 図形処理ライブラリの利用

図形表示をするためには、GAMEコマンドの一部のようなすでに作成されているコンプリートなプログラムを使用するか、センターにある図形処理ライブラリを利用して、ユーザがアプリケーションプログラムを作成しなければなりません。このとき、利用できるライブラリは次のようなものがあります。

- FORTARN (2進モード)

統合化ライブラリ T 4014用      LIB/ABL1B, LIB/N6922

統合化ライブラリ T4027用	LIB/ABLIB, LIB/T4027
統合化ライブラリ N6940用	LIB/ABLIB, LIB/N6940
図形処理基本システム (GDSP-6/PLOT)	LIB/GPLT
カラー表示基本システム (GDSP-6/CPLOT)	LIB/CPLOT
• FORTARN77 (Vモード)	
統合化ライブラリ T4014用	LIB/AVLIB, LIB/N6922V
統合化ライブラリ N6940用	LIB/AVLIB, LIB/N6940V
図形処理基本システム (GDSP-6/PLOT)	LIB/GPLTV
カラー表示基本システム (GDSP-6/CPLOT)	LIB/CPLTV
PLOT 10 インタラクティブ・グラフィック・ライブラリ	LIB/IGLV
• C言語用	
図形処理ルーチン	/GLIB/CPLOT.C

ライブラリは、通常コンパイル時にオプションとして指定します。ライブラリの具体的な指定方法を以下に示します。

• FORTARN (2進モード)
* <u>RUN = (ULIB, BIN) LIB/ABLIB, R; LIB/N6922, R</u>
* <u>RUN = (ULIB, BIN) LIB/GPLT</u>
• FORTARN77 (Vモード)
* <u>RUN : L=LIB/AVLIB L=LIB/N6922V</u>
* <u>RUN : L=LIB/GPLTV</u>
• C言語用
* <u>CC /GLIB/CPLOT.C -M/GLIB/CPLOT.A</u> (ライブラリのコンパイルと OU ライブラリファイルへのマージ)
* <u>CC * -L/GLIB/CPLOT.A</u> (ソースプログラムのコンパイルと OU ライブラリファイルとのリンク)

### 5.3 図形出力例

ここでは、実際に簡単なプログラムを作成して、図形を出力してみます。グラフィック出力用のライブラリの使用方法等を参考にして下さい。

#### (1) 統合化ライブラリを使用した場合

次の図のように、下線部を入力して下さい。

```

SYSTEM ?FR7①
OLD OR NEW NEW②
*AUTOX 0010③
*0010*RUN :L=LIB/AVLIB L=LIB/N6922V④
*0020C⑤
*0030 PRINT *, 'ブンカツノコスウ ?'⑥
*0040 READ(5,*) IBUN⑦
*0050C⑧
*0060 CALL DEVICE("6922",0)⑨
*0070 CALL PLOTS⑩
*0080 CALL WINDOW(-1348.0, -780.0, 700.0, 780.0)⑪
*0090 DO 1010 I = 0, IBUN-1⑫
*0100 RAD = 2.0*3.14*I/IBUN⑬
*0110 CALL PLOT(0.0, 0.0, 0, 3)⑭
*0120 CALL PLOT(700.0*SIN(RAD), 700.0*COS(RAD), 2)⑮
*0130 1010 CONTINUE⑯
*0140 CALL PLOTE⑰
*0150 END⑱
*0160⑲
*
```

このプログラムでは、管面をテクトロニクス 4014 型または、日本電気 N 6922 と仮定しています。それでは、LISTコマンドで確認して実行してみます。

```

*LIST①
0010*RUN :L=LIB/AVLIB L=LIB/N6922V
0020C
0030 PRINT *, 'ブンカツノコスウ ?'
0040 READ(5,*) IBUN
0050C
0060 CALL DEVICE("6922",0)
0070 CALL PLOTS
0080 CALL WINDOW(-1348.0, -780.0, 700.0, 780.0)
0090 DO 1010 I = 0, IBUN-1
0100 RAD = 2.0*3.14*I/IBUN
0110 CALL PLOT(0.0, 0.0, 0, 3)
0120 CALL PLOT(700.0*SIN(RAD), 700.0*COS(RAD), 2)
0130 1010 CONTINUE
0140 CALL PLOTE
0150 END

```

\*RUN<sup>⑲</sup>

ブンカツノコスウ ?  
I\*?19<sup>⑳</sup>

\*

この図のように、グラフィック画面に図形が表示されるはずです。

## (2) 図形処理基本システムを使った場合

次の図のように、AUTOコマンドを利用して下線部を入力して下さい。

```

SYSTEM ?FRIT(②)
OLD OR NEW-NEW(②)
*AUTOX 0010C(②)
*0010*RUN :L=LIB/CPLTV(②)
*0020C(②)
*0030 PRINT *, 'ブンカン ノ コスウ ?'(②)
*0040 READ(5,*) IBUN(②)
*0050C(②)
*0060 CALL INITT(960)(②)
*0070 CALL TERM(2)(②)
*0080 CALL TWINDO(0, 4095, 0, 3120)(②)
*0090 CALL DWINDO(-1348, 0, 700, 0, -780, 0, 780, 0)(②)
*0100 DO 1010 I = 0, IBUN-1(②)
*0110 RAD = 2.0*3.14*I/IBUN(②)
*0120 CALL MOVEA(0, 0, 0, 0)(②)
*0130 CALL DRAWA(700.0*SIN(RAD), 700.0*COS(RAD))(②)
*0140 1010 CONTINUE(②)
*0150 CALL FINITT(1, 1)(②)
*0160 END(②)
*0170C(②)

```

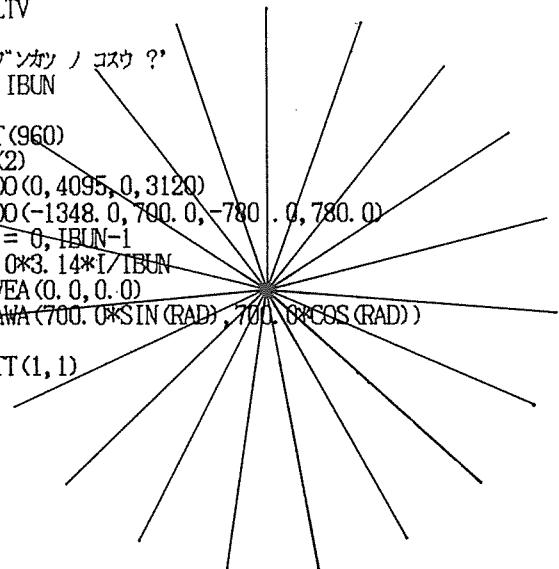
ここでも、管面をテクトロニクス 4014 型と仮定していますが、端末の設定を確認・変更したい時は、「6.6 グラフィックスの設定」のところを参考にして下さい。それでは、LIST コマンドで確認して実行してみます。

```

*LIST(②)
0010*RUN :L=LIB/CPLTV
0020C
0030 PRINT *, 'ブンカン ノ コスウ ?'
0040 READ(5,*) IBUN
0050C
0060 CALL INITT(960)
0070 CALL TERM(2)
0080 CALL TWINDO(0, 4095, 0, 3120)
0090 CALL DWINDO(-1348, 0, 700, 0, -780, 0, 780, 0)
0100 DO 1010 I = 0, IBUN-1
0110 RAD = 2.0*3.14*I/IBUN
0120 CALL MOVEA(0, 0, 0, 0)
0130 CALL DRAWA(700.0*SIN(RAD), 700.0*COS(RAD))
0140 1010 CONTINUE
0150 CALL FINITT(1, 1)
0160 END

```

\*RUN(②)  
ブンカン ノ コスウ ?  
I\*?19(②)  
\*



統合化ライブラリを使用した時と同じ图形が表示されるはずです。图形が 4 倍に拡大されたり、4 分の 1 に縮小されて表示される時は、「6.6 グラフィックスの設定」を参考にしてグラフィックスの設定を訂正して下さい。

### (3) C 言語用图形処理ルーチンの場合

まず、次のように R モードの適当なサブシステムを呼び出してから、AUTO コマンドを利

用して、プログラムを入力します。

```
SYSTEM ?FRTR②
OLD OR NEW-NEW③
*AUTOX 0010④
*0010# include <stdio.h>⑤
*0020# include <math.h>⑥
*0030main()⑦
*0040 {
*0050     int i, ibun;⑧
*0060     float rad;⑨
*0070     printf("ブンガツ ノコスウ ?\n");⑩
*0080     scanf("%d", &ibun);⑪
*0090     init(960);⑫
*0100     dwindo(-1348.0, 700.0, -780.0, 780.0);⑬
*0110     for(i = 0; i < ibun; i++) {⑭
*0120         rad = 2.0*3.14*i/ibun;⑮
*0130         movea(0.0, 0.0);⑯
*0140         drawa(700.0*sin(rad), 700.0*cos(rad));⑰
*0150     }⑱
*0160     finitt(1,1);⑲
*0170}⑳
*0180㉑
*㉒
```

このプログラムを見ただけでもわかりますが、C言語用の図形出力ルーチンは图形処理基本システムを参考にしています。したがって、関数名や引数は GDSP-6/PLOT とコンパチブルで、ファイルへの出力等の機能が拡張されています。さらに第二精工舎の D-SCAN GR-1140 もサポートしているため、日本電気の N6940 でも使用できます。それではコンパイルして、图形出力ルーチンをライブラリファイルからリンクして実行してみます。

```
*LIST㉓
0010# include <stdio.h>
0020# include <math.h>
0030main()
0040 {
0050     int i, ibun;
0060     float rad;
0070     printf("ブンガツ ノコスウ ?\n");
0080     scanf("%d", &ibun);
0090     init(960);
0100     dwindo(-1348.0, 700.0, -780.0, 780.0);
0110     for(i = 0; i < ibun; i++) {
0120         rad = 2.0*3.14*i/ibun;
0130         movea(0.0, 0.0);
0140         drawa(700.0*sin(rad), 700.0*cos(rad));
0150     }
0160     finitt(1,1);
0170}
```

```
*CC * -L/CLIB/CPLT.A㉔
```

```
*GO㉕
```

ブンガツ ノコスウ ?

=19<sup>㉖</sup>

\*

このプログラムでは、管面にテクトロニクスの 4010・4012 型を考えています。

## 5.4 グラフィック画面のハードコピー

センターからデータが送られてきていないときに、^GRCOPYと端末用コマンドを入力すると、グラフィック画面のハードコピーがプリンターへ出力されます。コマンドの操作方法や各種パラメータは下記の項目を参考にして下さい。ハードコピーに関するパラメータは必ずグラフィック画面への描画を始める前に設定・変更しておいて下さい。

ハードコピーのパラメータの設定 …… 「6.7 グラフィック画面のコピーの設定」

ハードコピーの端末用コマンド …… 「7.4 グラフィックスの操作」

## 5.5 グラフィック画面の消去

ハードコピーのときと同じく ^GRCLR と入力しますと、グラフィック画面だけが消去されます。

## 5.6 グラフィックデータのファイルへの記録

グラフィックでディスプレイに出力したものと同じものをプロッター等で出力したり、保存したりするためにグラフィックデータを端末 OS 上のファイルへ記録することができます。まず

^GRHIST filename

と入力し、記録するファイル名を指定します。これ以後センターから送られてくるグラフィックのデータが、端末 OS 上のファイルへ記録されます。この端末用コマンドはグラフィック表示を開始する前に入力しなければなりません。このとき記録されるデータの形式は、プロッターの制御コードを用いており、通常のテキストファイルと同様に編集することもできます。このグラフィック・データ・ファイルから、グラフィック画面、プリンター、プロッターのそれぞれへ出力するツールも用意しています（プログラム PLOT の操作説明書を参考にして下さい）。例えば、( 0, 0 ) – ( 1023, 779 ) の枠を表示した場合、具体的には次のように記録されます（グラフチックのプロッター用の制御コードを用いています）。

グラフィック・データ

制御コードの解説

M 0, 0 ( 0, 0 ) へ移動します（描画しない）。

D 1023, 0 ( 1023, 0 ) へ直線で描画します。

D 1023, 779 ( 1023, 779 ) へ直線で描画します。

D 0, 779 ( 0, 779 ) へ直線で描画します。

D 0, 0 ( 0, 0 ) へ直線で描画します。

ファイルへの記録を終了したいときは、次のように入力します。

^GRHIST OFF

2つの端末用コマンドの使い方とファイル上に記録されるデータの形式は、下記の項目を参考にして下さい。

データ形式の確認・変更 …… 「6.6 グラフィックスの設定」  
端末用コマンド …… 「7.4 グラフィックスの操作」

## 5.7 グラフィックス使用上の注意

以上のように、このターミナルプログラムでは、グラフィック端末用に作成されたプログラムをそのまま実行することができます。ただし、グラフィックライブラリを利用してプログラムを作成または実行する時には、次のことに注意して下さい。

### (1) テキスト画面とグラフィック画面の分離

テキスト画面とグラフィック画面はまったく別の画面であると考えています。そのため、グラフィックライブラリによるテキスト画面への書き込みで、うまく動作しないものがあります。具体的には、次のサブルーチンは、使用しないことが望されます。

ANCHOR, ANSTR, AIOUT  
AOUTST (in GDSP-6/PLOT)  
NUMBER, SYMBOL, BLINK (in AVLlib)

### (2) ハードコピーの出力

プログラムによってハードコピーをとることはできません。もし、つぎのサブルーチンを実行すると、「ハードコピーの出力要求がありました」という表示が画面の再下部にされるだけで、ハードコピーはとられません。そのときは、端末用コマンドを使ってハードコピーをとって下さい。

HDCOPY (in GDSP-6/PLOT)  
HDCERS (in AVLlib)

### (3) サポートしていないハードウェア

周辺装置のうちタブレットやライトペンによる座標入力はサポートしていません。その他ハードウェアでサポートしていないため、その機能が実現できない次のようなサブルーチンがあります。したがって、使用してはいけません。

TABINT, TABPRS, MULPNT  
ONEPNT, CHRSIZ, CZAXIS (in GDSP-6/PLOT)  
TABINT, ONEPNT, LPEN  
BLINK, LEARN, MAP, MIX  
PAT 27, PCOL 27 (in AVLlib)

## 第6章 実行環境の設定

センターとの会話モード、すなわち、画面編集機能を利用して、センターと会話をしている時に  
[ESC]を押すと、つぎのような画面が現われて、メインメニューの選択を要求します。

機能を選択して下さい

☆ TSS の会話概観

☆ 実行環境の設定

※ プログラム終了

著作・制作 浜村 秀彦

ここで、カーソルキー(↑ ↓)を使って、「実行環境の設定」を選択して、②を押します。  
すると、次のようなサブメニューが表示されます。

設定する実行環境の項目を指定して下さい

☆ 通 信 回 準

☆ 会 話 環 境

☆ ファンクションキー

☆ グラフィックス

☆ ハードコピー

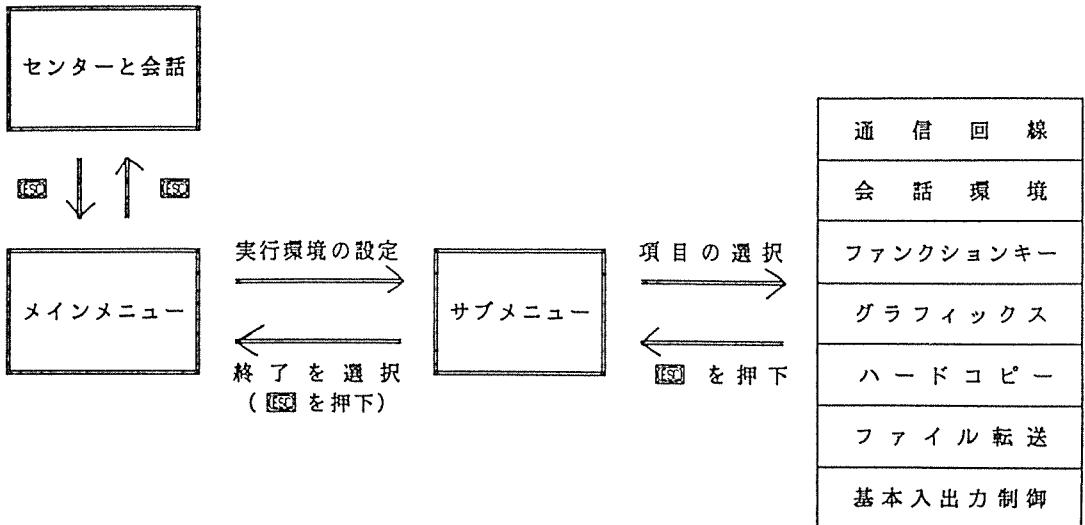
☆ ファイル転送

☆ 基本入出力制御

※ 実行環境の設定終了

著作・制作 浜村 秀彦

さらに、カーソルキー(① ②)を使って、このメニューの中から変更したい環境を選択し、③を押すと、それぞれの環境のパラメータなどが、表示されます。実行環境の設定を終了したいときは、④を押しますとメインメニューへ戻ります。



## 6.1 パラメータを変更するときの注意

実行環境に関するパラメータや制御コードを設定・変更できるようになっています。しかしその変更に際しては、下記の説明をよく読んでから行って下さい。

### (1) ★の印のついた項目の変更

この項目は、多くの場合変更する必要がないはずです。したがって、初心者は、変更しないで下さい。また、TSSや本システムを熟知している方が変更する時でも、十分に注意して行って下さい。

### (2) ☆の印のついた項目の変更

この項目は、必要あれば適当に変更して、最適な実行環境を設定して下さい。

### (3) パラメータの変更

すべてのパラメータの変更は、カーソルキー(① ②)を使って選択するだけです。③または④を押せば、隣の項目に移ります。また、このとき④でサブメニューに戻ることもできます。

### (4) 文字列の変更

文字列にはカーソルキー(① ② ③ ④)と1文字削除キー(⑤ or ⑥)以外のすべての文字を入力することができます(つまりコントロールキーや⑦も入力できます)。文

字列の編集を終了したい時には、カーソルキーのうち と を使って下さい。このとき、  
 でサブメニューに戻ることはできません。

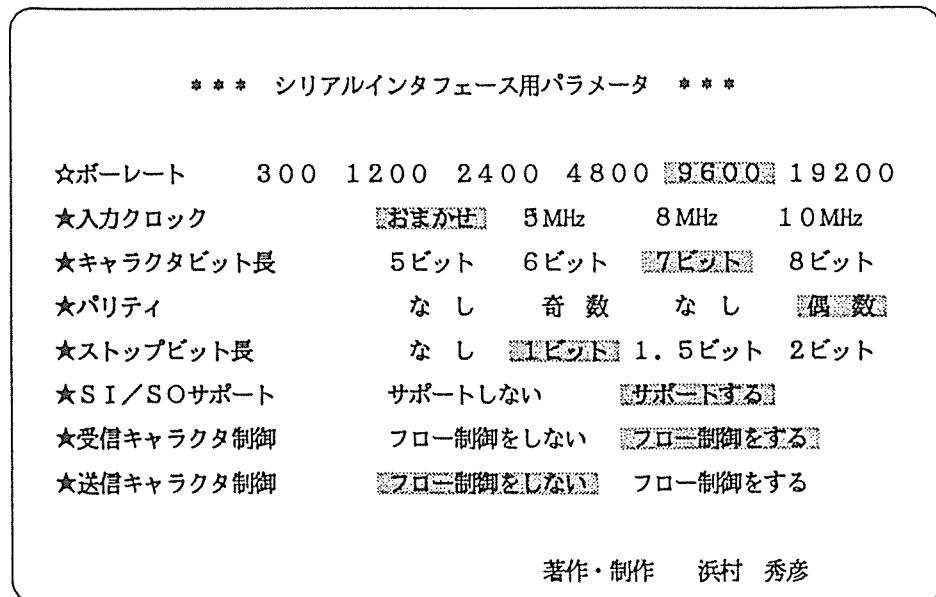
文字列の変更に際しては「6.10 文字列を入力するまでの注意」をよく読んで下さい。

## 6.2 環境設定メニューの一覧

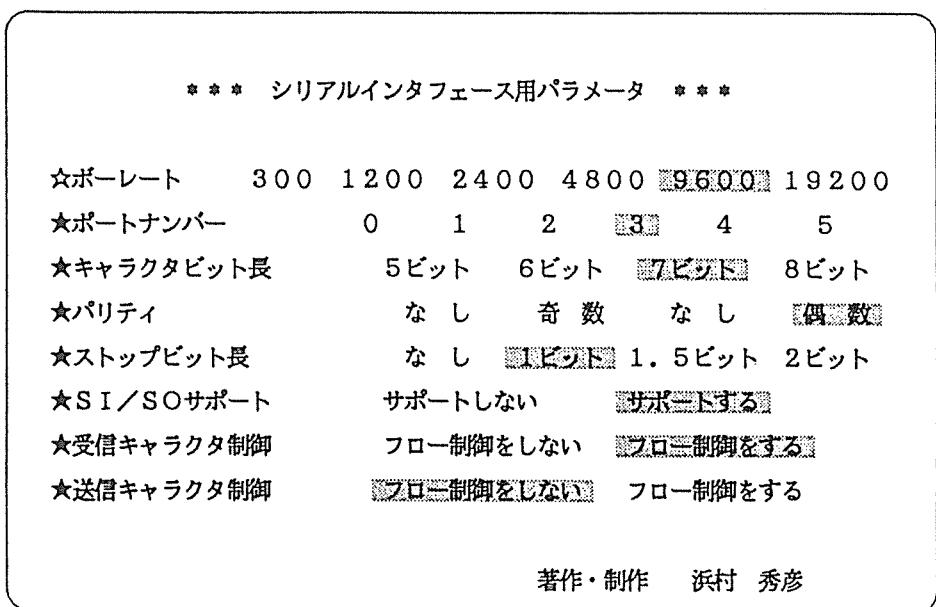
実行環境	☆通信回線 (PC 9801)	☆ボーレート -★入力クロック -★キャラクタビット長 -★パリティ -★ストップビット長 -★SI/SOサポート -★受信キャラクタ制御 -★送信キャラクタ制御	☆ボーレート -★ポートナンバー -★キャラクタビット長 -★パリティ -★ストップビット長 -★SI/SOサポート -★受信キャラクタ制御 -★送信キャラクタ制御 (シリアルインターフェース用)	☆ボーレート -★ポートナンバー -★キャラクタビット長 -★パリティ -★ストップビット長 -★SI/SOサポート -★受信キャラクタ制御 -★送信キャラクタ制御 (コンソール用)
	☆通信回線 (ai-M16)	☆ボーレート -★ポートナンバー -★キャラクタビット長 -★パリティ -★ストップビット長 -★SI/SOサポート -★受信キャラクタ制御 -★送信キャラクタ制御		
	☆会話環境	☆プリンターへの出力 -☆プリンターの遠隔制御 -★送信文字のエコー -☆受信した文字の色 -☆送信した文字の色 -☆入力した文字の色		
	☆ファンクションキー			
	☆ハードコピー	☆記憶領域上の仮想画面 -★ハードコピーのドット数 -☆ハードコピーのズーム -☆プリンターの機種 -★プリンター1の制御コード -★プリンター2の制御コード		
	☆グラフィックス	☆実画面のドット数 -☆ディスプレイの機種 -☆描画色の既定値 -☆プロッターの機種 -★プロッター1の制御コード -★プロッター2の制御コード		
	☆ファイル転送	★非表示文字の取扱い -★LOADコマンド -★DTLOADコマンド		
	☆基本入出力制御コード	☆通信回線漢字シフト -★JIS漢字制御コード -★ACOS漢字制御コード -★プリンター制御コード -☆プリンターへの制御文字の処理		

### 6.3 通信回線の設定

メインメニューのところで、「通信回線の設定」を選択すると、PC-9801を使用している場合は、以下のような画面が現われます。



一方、ai-M16を使用している場合は、次のような画面が現われます。



次に、設定できるパラメータのそれぞれの項目について説明します。

(1) ポーレート

ACOS-6 / MVX に接続する場合、基本的にはこのポーレートの項目以外を変更する必要がありません。さらに、電話回線で接続している利用者には 300, 1200, 2400 ボーしか利用できません。また、専用回線では、ポーレートがある値に固定されていることもあります。いずれにしても、モ뎀に適合したポーレートで使用して下さい。

(2) 入力クロック (PC-9801 のみ)

ここで、入力クロックとは PIT 8253 の CLK 2 端子へ入力しているクロックの周波数のことと、システムのクロックとは必ずしも、一致しません。「おまかせ」のところにセットしておくと、自動的にシステムのクロック周波数を検出し、入力クロックの周波数を適当な値に見なして、内部パラメータをセットします。自動設定で正常に動作しないときに限り、変更して正常に通信できるクロック周波数を探して下さい。

(3) ポートナンバー (ai-M16 のみ)

ai-M16 の場合、SIO ポート (RS-232C) は、6 ポート用意されています。そのうち、ポートナンバーが 0 と 1 のポート (ハードウェアでは SIO1 と SIO2) は、コンソールに割り当てられています。したがって、ポートナンバー 2 ~ 5 (SIO3 ~ 6) の中から 1 つがセンターとの接続に使用されているはずです。ここでは、このセンターと接続しているポートナンバーを指定して下さい。

(4) キャラクタビット長・パリティ・ストップビット長

ACOS-6 / MVX の場合、このパラメータを変更すれば接続できません。つまり、変更してはいけません。ここでは、これ以上の説明は省略します。詳しくは、非同期シリアルインターフェースの規格などを参考にして下さい。

(5) SI/SO サポート

キャラクター長が 7 ビットの時だけ、シフトイン / シフトアウト機能を利用する事が、できます。SI/SO 機能をサポートしないと、7 ビットのキャラクター長ではカタカナを表示することができません。センターと接続する時は、SI/SO 機能をサポートしないといけません。

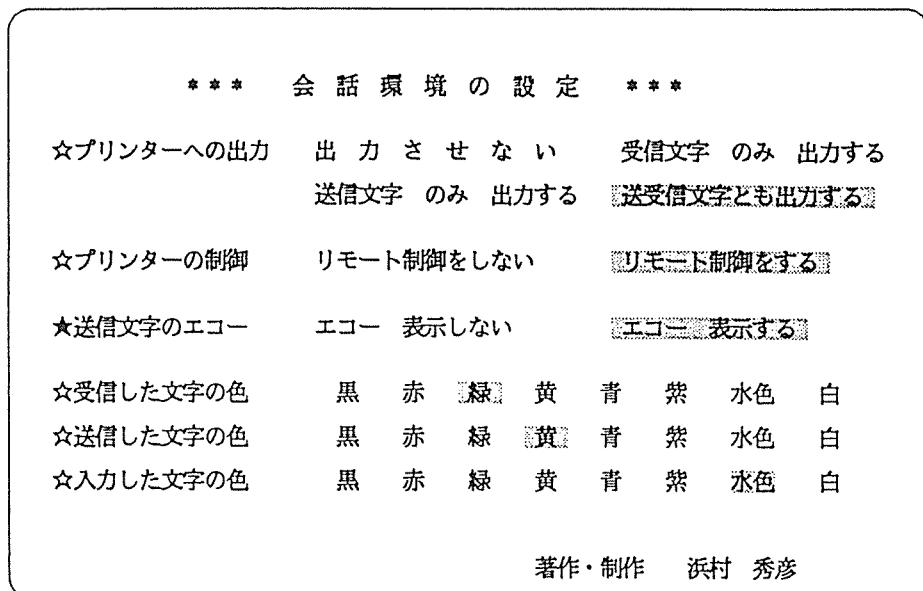
(6) 送受信キャラクターの制御

送受信の制御とは、センターからキャラクタを受信する時に端末側が、またはセンターへ送信する時にセンター側が、キャラクターの送信の停止を要求して、同期操作を行なうことを行います。同期操作には、RS, CS によるハンドシェイク、ER によるビジィ制御、DC1, DC3 による XON, XOFF 制御方式があります。このうち、XON, XOFF によるフロー制

御を使用しています。本プログラムをセンターと接続するためには、この受信キャラクターのフロー制御を行なう必要があります。このパラメータも変更する理由は、見当たりません。

#### 6.4 会話環境の設定

サブメニューのところで、「会話環境」を選択すると、次のような画面が現われます。



##### (1) プリンターへの出力

**CTRL** + **Q** によって、プリンターのエコー機能を開始した時にプリンターへ出力するソースリストを選択します。具体的に説明しますと、プリンターのエコー機能はセンターへ送信した文字とセンターから受信した文字をプリンターへ出力します。この項目では、このうちプリンターへ出力する文字を「どちらか片方を出力する」あるいは「両方とも出力する」を指定します。

##### (2) プリンターの制御

プリンターのエコー機能をセンターのアプリケーションプログラムによって ON/OFF するリモート制御機能があります。ここでは、この機能を使用するかどうかを指定します。

##### (3) 送信文字のエコー

センターと接続する時には、変更する必要はありません。

##### (4) 文字の色

文字の色は、標準では以下のようになっていて、それぞれの文字の状態が確認できます。

センターから受信した文字など	緑色
----------------	----

キーボードから入力した直後の文字 水色

センターへ送信した文字 黄色

ここでは、この色を変えることができます。趣向に合せて自由に設定して下さい。

## 6.5 ファンクションキーの設定

サブメニューのところで、「ファンクションキー」を選択すると、次のような画面が現われます。文字列の入力方法は、「6.10 文字列を入力する上での注意」を参考にして下さい。

### \* \* \* ファンクションキー の 設 定 \* \* \*

```
1> :$$$CON,TSS,,KNM  
2> :X66001;A$PASSWORD  
3> :  
4> :  
5> :  
6> :  
7> :  
8> :  
9> :  
10> :$$$CON,TSS,,ASC  
11> :^GRCOPY  
12> :^GRCLR  
13> :^LOAD  
14> :^SAVE  
15> :^DTLOAD  
16> :^DTSAVE  
17> :^ASC  
18> :^HEX  
19> :^HIST  
20>
```

著作・制作 浜村 秀彦

### (1) ファンクションキーの利用

ファンクションキーは、それぞれのキーに機能があらかじめ登録されているのではなく、文字列を登録することによって、センターのコマンドや本システムのコマンドおよび機能を自由に利用することができます。この「ファンクションキーの設定」を十分に活用して、各ユーザーにとって最適な環境を設定して下さい。

### (2) 予約しているファンクションキー

[ f · 1 ] と [ f · 11 ] は接続コマンドを入れていますので、センターと接続する時には変更しないで下さい。また、[ f · 2 ] には、パスワードを登録しておくと便利です。

### (3) 「サブメニュー」への戻り方

この「ファンクションキーの設定」から「サブメニュー」へ戻るには、 [ ESC ] や [ Q ] キーは、すべて文字として登録されますので、使うことができません。そこでまず、カーソルキー( [ ← ] [ → ] )を使って、まず [ f · 1 ] または [ f · 20 ] のところへカーソルを移動させます。

そして、[ f · 1 ] からは を押せば、[ f · 20 ] からは を押せば、「サブメニュー」へ戻ることができます。

### (3) ファンクションキーの使用方法

ファンクションキーは、20個まで登録して使うことができますが、実際のキーボードには、10個あるいは16個しかありません。そこで、以下のように入力します。

ファンクション	入力方法		使用目的
	ai-M16	PC 9801	
F 1	[ F 1 ]	[ f · 1 ]	接続コマンド
F 2	[ F 2 ]	[ f · 2 ]	パスワード等
F 3	[ F 3 ]	[ f · 3 ]	
F 4	[ F 4 ]	[ f · 4 ]	
F 5	[ F 5 ]	[ f · 5 ]	
F 6	[ F 6 ]	[ f · 6 ]	
F 7	[ F 7 ]	[ f · 7 ]	
F 8	[ F 8 ]	[ f · 8 ]	
F 9	[ F 9 ]	[ f · 9 ]	
F 10	[ F 10 ]	[ f · 10 ]	
F 11	[ F 11 ]	[ SHIFT ]+[ f · 1 ]	
F 12	[ F 12 ]	[ SHIFT ]+[ f · 2 ]	
F 13	[ F 13 ]	[ SHIFT ]+[ f · 3 ]	
F 14	[ F 14 ]	[ SHIFT ]+[ f · 4 ]	
F 15	[ F 15 ]	[ SHIFT ]+[ f · 5 ]	
F 16	使用できない	[ SHIFT ]+[ f · 6 ]	
F 17	[ SHIFT ]+[ F1 ]	[ SHIFT ]+[ f · 7 ]	
F 18	[ SHIFT ]+[ F2 ]	[ SHIFT ]+[ f · 8 ]	
F 19	[ SHIFT ]+[ F3 ]	[ SHIFT ]+[ f · 9 ]	
F 20	[ SHIFT ]+[ F4 ]	[ SHIFT ]+[ f · 10 ]	

## 6.6 グラフィックスの設定

グラフィックスは、センターから送られてくるTEXTRONIXのグラフィックコードを解読して、グラフィック画面に出力しています。このグラフィックスに関するパラメータをセンターのアプリケーション・プログラムや周辺装置に応じて設定する必要があります。サブメニューで、「グラフィックス」を選択すると、次のような画面が現われます。

\* \* \* グラフィックス の 設 定 \* \* \*

☆実画面のドット数 640\*400 640\*512 1024\*780 2048\*1560 4096\*3120

☆ディスプレイの機種 テクトロ4010(N6921) テクトロ4014(N6922)

☆描画色の既定値 黒 赤 緑 黄 青 紫 水色 白

☆プロッターの機種 ヒトツメ ふたつめ

★プロッター1の制御コード

プロッター1の機種名	F P 5 3 0 1
プロッター初期化コード	:
ペンアップ移動 コード	:N%4d,%4dM
ペンダウン移動 コード	:D%4d,%4dM
ペン の 選択 コード	:J%4dM

★プロッター2の制御コード

プロッター2の機種名	X Y P L O T T E R
プロッター初期化コード	:
ペンアップ移動 コード	:N%4d,%4dM
ペンダウン移動 コード	:D%4d,%4dM
ペン の 選択 コード	:

著作・制作 浜村 秀彦

(1) 実画面のドット数・ディスプレーの機種

センターのアプリケーション・プログラムで考えている実画面の大きさとグラフィック・ディスプレイを指定します。通常は、考えているディスプレイによって以下のように指定します。

640×512 : テクトロニクスT 4027, 日本電気N 6940

1024×780 : テクトロニクスT 4010,

テクトロニクスT 4012, 日本電気N 6921

4096×3120 : テクトロニクスT 4014, 日本電気N 6922

(2) 描画色の既定値

好みに応じて選択して下さい。

(3) プロッターの制御コード

グラフィックのデータをファイルへ出力するための書式を指定します。普通はプロッター制御コードを設定しておきます。2つの制御コードの中から、使用する時は選択して下さい。変換仕様の書式と機能は、「6.10 (3) 引数のある文字列について」を参考にして下さい。

初期化コード：プロッターを初期化するコードです。

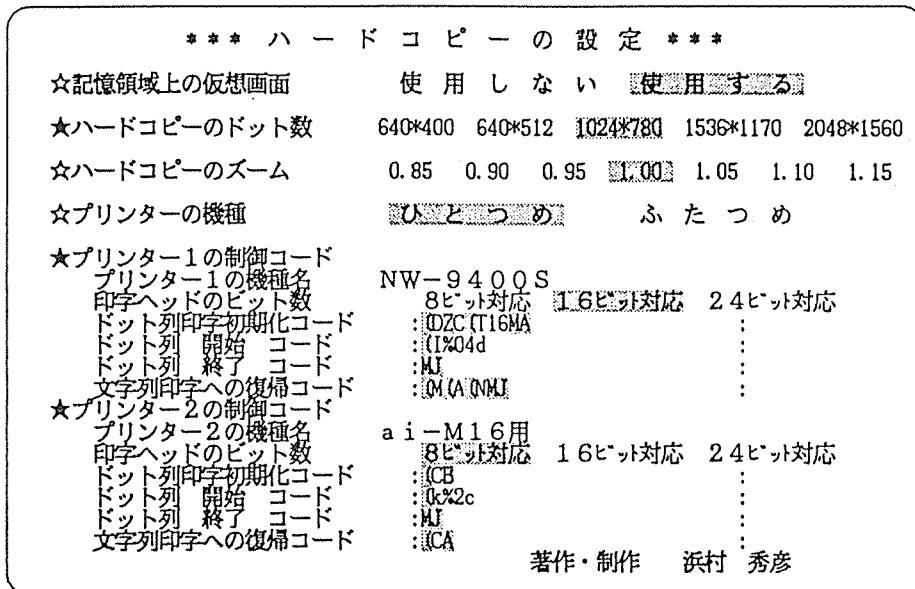
ペンアップ移動コード：プロッターのペンを上げて移動させるコードです。XとYの座標値を引数としていますので、必ず変換仕様を2つ指定して下さい。

ペンダウン移動コード：プロッターのペンを下げて線画させるコードです。XとYの座標値を引数としていますので、必ず変換仕様を2つ指定して下さい。

ペンの選択コード：グラフィックの描画色を引数として、ペンを選択するコードです。

## 6.7 グラフィック画面のコピーの設定

グラフィック画面のハードコピーは、端末の主記憶領域上に画面と同一の描画を行ない、それをプリンターへ出力しています。そこで、記憶領域に対する設定も必要になります。サブメニューで、「ハードコピー」を選択すると、次のような画面が現われます。



### (1) 記憶領域上の仮想画面

主記憶領域上に対する描画を行なうかどうかを設定します。記憶領域に対する書き込みを行なわないと、9600 ポートのとき画面への描画が速くなりますが、ハードコピーの精度が 640 × 400 に固定されます。

### (2) ハードコピーのドット数、ハードコピーのズーム

主記憶領域に設定する仮想画面のドット数を指定します。一度設定すると以後固定されままでの、変更したい時は、本プログラムを起動した後描画を行なうまでに変更するか、あらかじめ変更しておいて下さい。

### (3) プリンターの制御コード

グラフィックイメージをプリンターへ出力するための、プリンターの制御コードを 2 台まで設定することができます。この制御コードを変更することができるため、ほとんどのプリンターでハードコピーをとることができます。変換仕様の書式と機能は、「6.10 (3) 引数の文字列について」を参考にして下さい。

初期化コード： 1画面の出力を行なう前にプリンターの初期化を行なうコードです。

開始コード： 1行の出力を行なう前にプリンターへドット列印字を行なうことを指示するコードです。このコードには、1行のドット数を引数としてもっていますので、必ず変換仕様を指定して下さい。

終了コード： 1行の出力を行なったあとにその終了を指示するコードです。

復帰コード： 1画面の出力を行なった後に文字列印字へ復帰するコードです。

## 6.8 ファイル転送の設定

ファイル転送の設定項目は、2画面分あります。サブメニューで、「ファイル転送」を選択すると、次のような画面が現われます。

*** フ ァ イ ル 転 送 の 設 定 1 ***							
★非表示文字の取扱い ファイルをセンターへ送信するときに、ファイル中の非表示文字 (16進数で01～1F)をどのようにするかについて指定します							
変換しない：							
08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17
18	19	1A	1B	1C	1D	1E	1F
表示文字に変換：							
08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17
18	19	1A	1B	1C	1D	1E	1F
削除する：							
08	09	0A	0B	0C	0D	0E	0F
10	11	12	13	14	15	16	17
18	19	1A	1B	1C	1D	1E	1F
著作・制作 浜村 秀彦							

この画面でカーソルを下に向って移動させると、次のような表示に変わります。

\*\*\* ファイル転送の設定 2 \*\*\*

★LOADコマンド

	送信文字列	受信文字列
開始手続き	:NEW :AUTOX 00010,10M	:MJ* :MJ*00010
転送手続き	転送文字列	:MJ%1d%1d%1d%1d
終了手続き	:M :	:MJ*
		:

★DTLOADコマンド

	送信文字列	受信文字列
開始手続き	:NEWM :EDITM	:MJ* :MJENTERMJ*
転送手続き	転送文字列	:MJ*
終了手続き	:M :DONEM	:MJ- :MJ*
		:

著作・制作 浜村 秀彦

(1) 非表示文字の取扱い

非表示文字とは、16進数表現で $01_{16}$ から $1F_{16}$ までの文字を指します。これらの文字は、そのキャラクタが定義されていないため表示することができません。センターにファイルを送信しようとしたときに、その中に非表示文字があると送信できないことがあります。それは、非表示文字のうち以下のコードが通信の制御に割り当てられているため1キャラクタとして受信されないからです。

非表示文字		端末からの入力文字の取扱い
$01_{16}$	SON	簡易コネクト
$03_{16}$	ETX	簡易ディスコネクト
$04_{16}$	EOT	一行削除
$08_{16}$	BS	一字削除
$0D_{16}$	CR	一行の終り
$0E_{16}$	SO	シフトアウトモードにする
$0F_{16}$	SI	シフトインモードにする
$11_{16}$	DC1	送信を再開する
$13_{16}$	DC3	送信を停止する

ファイルを送信するときは、非表示文字を削除するか表示文字に変換する必要があります。ここでは、このときの非表示文字の取扱いについて指定します。指定された内容に矛盾があ

るときは、「削除する」が選択されます。

## (2) LOAD コマンド・DTLOAD コマンド

LOAD・DTLOAD コマンドでは、センターのコマンドを利用して端末のファイルをセンターへ送信します(詳しくは、「7.5 ファイルの転送」を参照して下さい)。ここでは、このファイル転送の手続きを規定します。DTLOAD コマンドを例にとって、その手続きを具体的に説明します。

開始手続き : NEWM	:	...	カレントファイルを消去します
: MJ *	:	...	'*' を受信するまで待ちます
: EDITM	:	...	EDITORを呼び出します
: MJ ENTER MJ *	:	...	'ENTER' と '*' を待ちます
転送手続き      転送文字列	...	...	ファイルを 1 行送信します
: MJ *	:	...	'*' を受信するまで待ちます
: M	:	...	リターンだけを入力します
: MJ -	:	...	'-' を受信するまで待ちます
: DOME	:	...	EDITORを終了します
: MJ *	:	...	'*' を受信するまで待ちます

## 6.9 基本入出力制御コードの設定

サブメニューのところで、「基本入出力制御」を選択すると、次のような画面が現われます。

* * * 基本入出力制御コード の 設 定 * * *			
☆通信回線漢字シフト	JIS制御コード	ACOS内部コード	
★JIS漢字制御コード 日本語漢字の指示	:S0	:H	:
JISローマ文字の指示	:S1		
★ACOS漢字制御コード 漢字インシフットコード	:Zp	:	
漢字アウトシフットコード	:Zq	:	
★プリンター制御コード 初期化機能コード	:	:	
終了機能コード	:	:	
漢字モードへの切り替え	:OK	:	
漢字モードからの復帰	:ON	:	
☆プリンターへの制御文字の処理			
改頁コード (0C)	出力させない	表示文字で出力	そのまま出力
SUB文字 (1A)	出力させない	表示文字で出力	そのまま出力
エスケープ (1B)	出力させない	表示文字で出力	そのまま出力
著作・制作 浜村 秀彦			

### (1) 通信回線漢字シフト

- 端末の属性が漢字端末のとき

接続コマンドを、\$\$\$CON, TSS,, KNJ とした場合です。通常は、この漢字端末として接続し、「JIS制御コード」を選択します。ただし、ごく一部のソフトウェアが、センターの都合上うまく動作しないことがあります。このときは他のタイプの端末として接続して下さい。

- 端末の属性が漢字端末以外のとき

他のタイプの端末として、例えば接続コマンドを\$\$\$CON, TSS,, ASC として接続する場合は、「ACOS内部コード」を選択して下さい。

### (2) JIS漢字制御コードとACOS漢字制御コード

JIS C 6228 あるいは ACOS 日本語情報処理システムの概要書を参考にして下さい。ACOS の制御コードについては変更することはないと思われますが、JIS の制御コードは、将来変更しなくてはならないかもしれません。

### (3) プリンター制御コード

プリンターにキャラクタ印字を行なうときの制御コードです。漢字モードへの切り替えはプリンターのマニュアルをよく読んで確認して下さい。

### (4) プリンターへの制御文字の処理

改頁コードやエスケープコードをプリンターへ出力するかどうかを選択します。

## 6.10 文字列を入力する上での注意

### (1) 入力可能文字

文字列に、16進数表現で  $01_{16}$  から  $7E_{16}$  までと  $A1_{16}$  から  $DF_{16}$  までのすべてのキャラクターを自由に入力することができます。文字といえば、すべての ASC II 文字とカタカナ、さらにコントロール文字が入力できます。ただし、 $7F_{16}$  は入力できません。漢字についても入力はできますが、その動作は保証していません。

### (2) コントロール文字の入力と表示

コントロール文字とは、16進で  $01_{16}$  から  $1F_{16}$  までの文字をいいます。たとえば、 $01_{16}$  は、**[CTRL]** + **[A]** と押すことによって入力します。コントロール文字は非表示（表示できない）文字のため、反転文字で表字します。具体的には、下にその対応を示していますので、参考にして下さい。

表示	16進	入力方法	表示	16進	入力方法	表示	16進	入力方法
A	0 1	[CTRL] + [A]	B	0 2	[CTRL] + [B]	C	0 3	[CTRL] + [C]
D	0 4	[CTRL] + [D]	E	0 5	[CTRL] + [E]	F	0 6	[CTRL] + [F]
G	0 7	[CTRL] + [G]	H	0 8	[INS]	I	0 9	[INS]
J	0 A	[HELP]	K	0 B	[CTRL] + [K]	L	0 C	[CTRL] + [L]
M	0 D	[Q]	N	0 E	[CTRL] + [N]	O	0 F	[CTRL] + [O]
P	1 0	[CTRL] + [P]	Q	1 1	[CTRL] + [Q]	R	1 2	[CTRL] + [R]
S	1 3	[CTRL] + [S]	T	1 4	[CTRL] + [T]	U	1 5	[CTRL] + [U]
V	1 6	[CTRL] + [V]	W	1 7	[CTRL] + [W]	X	1 8	[CTRL] + [X]
Y	1 9	[CTRL] + [Y]	Z	1 A	[CTRL] + [Z]	[ESC]	1 B	[ESC]
Z	1 C	[CTRL] + [Z]	[DEL]	1 D	[CTRL] + [DEL]		1 E	[CTRL] + [E]
	1 F	[CTRL] + [□]						

### (3) 引数のある文字列について

引数をもつ文字列は、文字列の制御の下で、書式に従って各引数を変換・編集し出力します。この制御用文字列には、2種類の文字があります。1つは普通の文字であり、そのまま出力されます。もう1つは、変換の使用を指定するもので、各変換仕様によって対応する引数が変換されて出力されます。

変換仕様は、いずれもパーセント(%)で始まり、変換コードで終わる形をしています。  
%と変換文字の間には次のような指定子があつてもかまいません。

マイナス符号：変換された文字列をフィールドの左側にそろえることを指定します。この  
指定がない場合は右側にそろえられます。

数 字 列：フィールドの最小幅を指定します。変換された文字列の長さが指定された  
フィールド幅より小さい場合、左側にパッドが入ります(マイナス符号の  
指定があれば右側にパッドが入ります)。

パッド用の文字は空白文字であります、0を最初につけてフィールドの  
幅を指定すれば0となります。

変換コードには次のようなものが用意されています。

- d 引数を10進数に変換します。
- c 文字型引数を出力します(無変換と同じ)。
- x 引数を符号なし16進数に変換します。

%のあとに続く文字が変換コードでない場合は、その文字が出力されます。したがって、%  
は%%で印字できます。次の表は、引数の値が316のときにその出力がどう変化するかを  
示したものです。まわりのコロンは、そのフィールドの範囲をわかりやすくするために付け

ています。

制御文字列	出力文字列
:%6d:	: 316:
:%06d:	:000316:
:%-6d:	:316
:%d:	:316:
:%04x:	:013C:

#### (4) 引数をもつ制御コード

引数をもつ制御コードについて、その引数と制御コード例を以下にまとめて示します。

	引 数	制御コード例
ペンアップ移動コード:	Xの座標値, Yの座標値	M%4d, %4d M J
ペンダウン移動コード:	Xの座標値, Yの座標値	D%4d, %4d D J
ペンの選択コード:	グラフィックスの描画色	J%4d M J
ドット列開始コード:	1行のドット数	[ J%04d

## 第7章 端末用コマンド

### 7.1 端末用コマンドの概説

端末用コマンドとは、端末におけるいろいろな機能を起動したり利用したりするために、端末に対して働きかける命令のことです。このコマンドを入力することにより、グラフィック画面の操作やファイルの送受信、さらに会話の記録、文字単位の会話を行なうことができます。

### 7.2 コマンドの使い方

端末用コマンドは、センターとの会話モード（「1.2 本プログラムの構成」を参照）のときに利用することができます。センターからのデータの送信が止まっていれば、任意のときに入力可能です。つまり、どんなプロンプト（入力促進記号）でもあるいはプロンプトが送られてきていないときでも、端末用コマンドを入力することができます。ただし、コマンドによっては、ある適当なサブシステムのビルドモードでアスタリスク(\*)記号が表示されていて、システムから入力が要求されていなければならぬものがあります。

端末用コマンドは必ずアクサンシルコンフレックス( ^ )記号から始まり、続けてコマンド名である英文字を入力します。今、コマンド名が ”COMMAND” のとき、

### ^ COMMAND ②

と入力します。また、大文字・小文字の区別はしていないため、次のようにすることもできます。

### ^ command ②

これで、コマンドが実行されるはずです。ファイルを指定する必要があるコマンドでは、コマンド名のあとに1個以上の空白を入れて、ファイル名を入力します。例えば、LISTコマンドでは、次のようにになります。

### ^ LIST A:TEXT. F77 ②

ファイル名は一般に次のような形をしています。

ドライブ記述子：ファイル名、ファイルタイプ

ここで、ドライブ記述子は英字1文字でファイルが格納されているディスクドライブを示します。

ファイル名は 8文字以内の英数字でハイフン（-）等の1部の記号を使うこともできます。

ファイルタイプは3文字以内の英数字でハイフン（-）等の1部の記号を使うこともできます。

ドライブ記述子は省略可能ですが、ファイル名、ファイルタイプは省略できません。たとえば、次のように入力することも可能です。

### ^ LIST TEXT. F77 ②

## 7.3 コマンドの一覧

用意されているコマンドの一覧を以下に示します。機能の概略の欄にファイルと書かれているコマンドは、端末のディスク上のファイル名を指定する必要があるコマンドです。

	コマンド名	機能の概略
グラフィックス	GRCLR GRCOPY	グラフィック画面を消去 グラフィック画面をプリンターへ出力
ファイルの転送	SAVE RESA LOAD DTSAVE DTRESA DTLOAD	プログラム <u>ファイル</u> をセンターから端末へ受信 プログラム <u>ファイル</u> をセンターから端末へ受信 プログラム <u>ファイル</u> を端末からセンターへ送信 データ <u>ファイル</u> をセンターから端末へ受信 データ <u>ファイル</u> をセンターから端末へ受信 データ <u>ファイル</u> を端末からセンターへ送信
ヒストリー機能	HIST HIST OFF GRHIST GRHIST OFF	文字による会話を <u>ファイル</u> に記録 HISTコマンドの終了 グラフィックデータを <u>ファイル</u> に記録 GRHISTコマンドの終了
文字単位の会話	ASC HEX	アスキーコードで会話 16進コードで会話
その他の	LIST HELP MEMORY	端末のディスク上の <u>ファイル</u> を表示 ヘルプメッセージを表示 メモリの残り容量を表示

#### 7.4 グラフィックスの操作

グラフィック画面に対する機能として、グラフィック画面の消去とグラフィック画面のハードコピーをプリンターへ出力するコマンドがあります。それぞれ、次のように入力します。

^ GRCLR ..... グラフィック画面の消去  
^ GRCOPY ..... グラフィック画面のハードコピー

グラフィック画面のハードコピーは、基本的に記憶領域上のグラフィックイメージを設定されたプリンターの制御コードに基づいて、プリンターへ出力します。ハードコピーのパラメータは、記憶領域の仮想画面に関するものとプリンターの制御コードに関するものがあります。仮想画面に関するパラメータの変更は、グラフィック描画を行なう前に行なう必要があります。変更に際しては、「6.7 グラフィック画面のコピーの設定」を参考にして下さい。

#### 7.5 ファイルの転送

センターと端末のディスクの間でテキストファイルの転送を行なうコマンドです。バイナリ型のデータが入っているファイルを転送することはできません。転送するファイルはプログラムとデータの2つに分けて考えています。これは端末のディスク上では、どちらも同一のテキストファイル

として扱いますが、センターでは、プログラムファイルの場合編集の都合上行番号を付けることが多いからです。

漢字を使用している場合、センター側では JIS 7 単位符号系、端末側ではシフト JIS を使っているものとし、自動変換しています。センター側と端末側のファイルの違いを以下にまとめます。

センターサイド		端末サイド
行番号を付加	プログラムファイル	行番号を削除
JIS 7 単位符号系	漢字の表現	シフト JIS
カレントファイル	転送するファイル	ディスク上の 1 ファイル
入力不可能 空白を付け加える	CrLf だけの行	入力可能
最大 160 文字	1 行の文字数	ファイルの容量まで

また、実際にファイルの転送を行なっているときは、次のキーだけを受け付けます。

入力可能なキー	機能
[ESC]	転送作業の中止
[STOP]	処理の中止（ブレーク）
[CTRL] + [B]	処理の中止（ブレーク）
[CTRL] + [S]	転送の 1 時停止
[CTRL] + [Q]	転送の再開

転送するファイルの種類と転送方向によって 4 つのコマンドがあります。

コマンド名	転送するファイル	転送元	転送先
SAVE	プログラムファイル	センター	端末のディスク
RESA	プログラムファイル	センター	端末のディスク
LOAD	プログラムファイル	端末のディスク	センター
DTSAVE	データファイル	センター	端末のディスク
DTRESA	データファイル	センター	端末のディスク
DTLOAD	データファイル	端末のディスク	センター

### (1) SAVE コマンド ( RESA コマンド )

センターのプログラムファイルを、端末のディスク上に保存します。対象とするセンターのプログラムファイルは、ACOS-6/MVX のカレントファイルです。ビルドモードのLISTコマンドを利用して転送するため、システムは適当なサブシステムのビルドモードにしておいて下さい。また、カレントファイルが、ライン番号つきの場合は、ライン番号が削除されてディスク上に保存されます。

このコマンドでは、ディスクに登録するファイルを指定する必要があります。SAVE コマンドは、その登録するファイル名は新しいファイル名でなければなりません。一方、RESA コマンドでは、すでにディスク上にあるファイル名を使用します。

```
*LIST②
0010 PRINT *, 'I と J を入力して下さい'
0020 READ *, I,J
0030 K = I*KJ
0040 PRINT *, 'I*KJ = ', K
0050 STOP
0060 END

*^SAVE TEXT.F77③
LIST
0010 PRINT *, 'I と J を入力して下さい'
0020 READ *, I,J
0030 K = I*KJ
0040 PRINT *, 'I*KJ = ', K
0050 STOP
0060 END

*
```

端末 OS のコマンドで確認することもできます。

```
A>TYPE TEXT.F77④
PRINT *, 'I と J を入力して下さい'
READ *, I,J
K = I*KJ
PRINT *, 'I*KJ = ', K
STOP
END

A>
```

### (2) LOAD コマンド

端末のディスクにあるプログラムファイルをセンターに転送します。ビルドモードにおけるオートマティックモードを利用して、ACOS-6/MVX のカレントファイルにプログラムファイルを作成します。そのため、システムは、適当なサブシステムのビルドモードにしておく必要があります。ファイルには、ライン番号が自動的に付加されます。さらに、CrLf

だけの行を送信する場合、空白を付加して送信します。また、接続した直後の状態では、1行あたり80文字しか送信することができません。80文字以上の行を送信したいときは、LINE 160 と入力しておいて下さい。

このコマンドでは、ディスク上の転送したいファイル名を指定しなければなりません。例えば、次のようなプログラムをセンターへ転送します。

```
A>TYPE LIST.F77②
CHARACTER FNAME*40,LINE*80
C
      PRINT *, 'ファイル名を入れてください。'
5010 READ(5, '(A)', END=5010) FNAME
      OPEN(UNIT=1, STATUS='OLD', FILE=FNAME)
      DO 1010 I = 1,9999
         READ (1,'(A)', END=1020) LINE
         WRITE(6,'(I4,: ",A")') I,LINE
1010 CONTINUE
1020 CONTINUE
      STOP
      END
```

A>②

LOAD コマンドでファイル名に LIST. F 77 と指定するだけで送信を開始します。

```
*\LOAD LIST.F77②
NEW
*AUTOX 00010,10
*00010      CHARACTER FNAME*40,LINE*80
*00020C
*00030      PRINT *, 'ファイル名を入れてください。'
*00040 5010 READ(5, '(A)', END=5010) FNAME
*00050      OPEN(UNIT=1, STATUS='OLD', FILE=FNAME)
*00060      DO 1010 I = 1,9999
*00070      READ (1,'(A)', END=1020) LINE
*00080      WRITE(6,'(I4,: ",A")') I,LINE
*00090 1010 CONTINUE
*00100 1020 CONTINUE
*00110      STOP
*00120      END
*00130

*LIST②
00010      CHARACTER FNAME*40,LINE*80
00020C
00030      PRINT *, 'ファイル名を入れてください。'
00040 5010 READ(5, '(A)', END=5010) FNAME
00050      OPEN(UNIT=1, STATUS='OLD', FILE=FNAME)
00060      DO 1010 I = 1,9999
00070      READ (1,'(A)', END=1020) LINE
00080      WRITE(6,'(I4,: ",A")') I,LINE
00090 1010 CONTINUE
00100 1020 CONTINUE
00110      STOP
00120      END

*
```

### (3) DTSAVE コマンド( DTRESA コマンド )

センターのデータファイルを、端末のディスク上に保存します。対象とするセンターのデータファイルは、ACOS-6/MVX のカレントファイルです。カレントファイルをビルドモードの LIST コマンドを利用してファイルを転送するため、データファイルをカレントファイルに呼び出して、システムをビルドモードにしておく必要があります。DTSAVE では行番号が削除されることはありません。

このコマンドでは、ディスクに登録するファイルを指定しなければなりません。その登録するファイル名は、DTSAVE コマンドでは新しいファイル名を、DTRESA コマンドではすでにあるファイル名を指定します。

あるデータファイルを端末のディスクファイルに保存します。LIST コマンドでカレントファイルの内容を確認してから、DTSAVE コマンドでファイル名を指定するだけで、転送ができます。

```
*^LIST②
1 |(p-2 . 1).1  (p-3 . 1).1      .....      .....
2 |(p-2 . 1).1  (p-3 . 1).1      .....      .....
3 |(p-2 . 1).1  (p-3 . 1).1      .....      .....
4 |(p-2 . 1).2  (p-3 . 1).1      .....      (p-2 . 2).1
5 |(p-2 . 1).2  (p-3 . 1).2      .....      (p-3 . 2).1
6 |(p-2 . 1).2  (p-3 . 1).2  (p-3 . 3).1      .....
7 |(p-2 . 1).2  (p-3 . 1).2  (p-3 . 3).1      .....

*^DTSAVE MACH.DAT②
LIST
1 |(p-2 . 1).1  (p-3 . 1).1      .....      .....
2 |(p-2 . 1).1  (p-3 . 1).1      .....      .....
3 |(p-2 . 1).1  (p-3 . 1).1      .....      .....
4 |(p-2 . 1).2  (p-3 . 1).1      .....      (p-2 . 2).1
5 |(p-2 . 1).2  (p-3 . 1).2      .....      (p-3 . 2).1
6 |(p-2 . 1).2  (p-3 . 1).2  (p-3 . 3).1      .....
7 |(p-2 . 1).2  (p-3 . 1).2  (p-3 . 3).1      .....

*
```

端末 OS のコマンドで確認できます。

```
A>TYPE MACH.DAT②
1 |(p-2 . 1).1  (p-3 . 1).1      .....      .....
2 |(p-2 . 1).1  (p-3 . 1).1      .....      .....
3 |(p-2 . 1).1  (p-3 . 1).1      .....      .....
4 |(p-2 . 1).2  (p-3 . 1).1      .....      (p-2 . 2).1
5 |(p-2 . 1).2  (p-3 . 1).2      .....      (p-3 . 2).1
6 |(p-2 . 1).2  (p-3 . 1).2  (p-3 . 3).1      .....
7 |(p-2 . 1).2  (p-3 . 1).2  (p-3 . 3).1      .....

A>
```

#### (4) DTLOAD コマンド

端末のディスクにあるデータファイルをセンターに転送します。データの送信は、EDITOR サブシステムにおけるビルドモードを利用して、ACOS-6/MVX のカレントファイルにデータファイルを作成します。EDITOR サブシステムは、ビルドモードから自動的に呼び出します。そのためシステムは適当なサブシステムのビルドモードにしておく必要があります。さらに、CrLfだけの行を送信する場合、空白を附加して送信します。一行あたり最大160 文字まで送信できますが、80 文字以上送信するときは、LINE 160 と入力して下さい。DTLOAD コマンドでは、ディスク上の転送するファイル名を指定する必要があります。

例えば、次のようなデータファイルをセンターへ転送します。

```
A>TYPE D860313.57②
860313 57
-36 -35 -35 -34 -34 -34 -32 -33 -31 -31
-31 -30 -28 -28 -26 -26 -24 -24 -22 -20
-18 -18 -16 -14 -12 -11 -9 -7 -5 1
 6 8 10 12 13 12 11 12 12 13
 13 14 14 13 14 14 14 14 15 15
 15 15 17 16 16 18 19 17 17 18
 17 18 18 19 19 18 19 19 19 19
 20 19 21 20 21 21 20 20 20 20
```

```
A>
```

DTLOAD コマンドでファイル名を指定すれば、送信することができ、LIST コマンドで確認できます。

```
*^DTLOAD D860313.57②
NEW
*EDIT
ENTER
*860313 57
* -36 -35 -35 -34 -34 -34 -32 -33 -31 -31
* -31 -30 -28 -28 -26 -26 -24 -24 -22 -20
* -18 -18 -16 -14 -12 -11 -9 -7 -5 1
* 6 8 10 12 13 12 11 12 12 13
* 13 14 14 13 14 14 14 14 15 15
* 15 15 17 16 16 18 19 17 17 18
* 17 18 18 19 19 18 19 19 19 19
* 20 19 21 20 21 21 20 20 20 20
```

```
*-
```

```
*LIST②
```

```
860313 57
-36 -35 -35 -34 -34 -34 -32 -33 -31 -31
-31 -30 -28 -28 -26 -26 -24 -24 -22 -20
-18 -18 -16 -14 -12 -11 -9 -7 -5 1
 6 8 10 12 13 12 11 12 12 13
 13 14 14 13 14 14 14 14 15 15
 15 15 17 16 16 18 19 17 17 18
 17 18 18 19 19 18 19 19 19 19
 20 19 21 20 21 21 20 20 20 20
```

```
*
```

## 7.6 ヒストリー機能

文字による会話またはグラフィックのデータを端末のディスクのファイルへ保存します。それぞれ、例題を用いて説明します。

### (1) HISTコマンド

センターとの文字による会話を指定されたファイルに記録します。記録するデータは文字データだけで、グラフィックのデータは記録しません。HISTコマンドで確認できます。HISTコマンドでは一般に次のような形をしています。

```
^HIST filename②
:
: (文字による会話)
:
^HIST OFF②
```

実際に、第4章で作成したプログラムの実行を端末のファイルに記録します。

```
*HIST A:TEXT.TSS②
LIST②
0010 PRINT *, 'IとJを入力して下さい'
0020 READ *, I,J
0030 K = I*j
0040 PRINT *, 'I*j = ', K
0050 STOP
0060 END

*RUN②
IとJを入力して下さい
I*2,3③
I*j = 6
*HIST OFF②
```

TYPEコマンドで確認すると、確かに記録できています。

```
A>TYPE TEXT.TSS②
LIST
0010 PRINT *, 'IとJを入力して下さい'
0020 READ *, I,J
0030 K = I*j
0040 PRINT *, 'I*j = ', K
0050 STOP
0060 END

*RUN
IとJを入力して下さい
I*2,3
I*j = 6
*
A>③
```

## (2) GRHIST コマンド

センターから送信されたグラフィックデータを指定されたファイルに、記録します。記録するデータはグラフィックデータだけで、文字のデータは記録しません。グラフィックデータは、例えば Mxxxx, yyyy または Dxxxx, yyyy の形でファイルに記録されます(この形式の変更方法は「6.6 グラフィックスの設定」を、変換の書式は「6.10 (3)引数をもつ制御コード」を参考にして下さい)。

GRHIST コマンドは、一般に次のような形式をしています。

```
^GRHIST filename②
  :
  : (グラフィックによる描画)
  :
^GRHIST OFF②
```

テストプログラムを実行して、グラフィックデータを記録してみます。

```
*LIST②
0010 RUN :L=LIB/CPLTV
0020
0030   IX = 1024/2
0040   IY = 780/2
0050   CALL INIT(120)
0060   CALL MOVABS(IX, IY+300)
0070   DO 1010 I=1,5
0080   RAD = 72*I/180.0*3.1415926
0090   CALL COLOR(MOD(I,7))
0100   JX = IX+300*SIN(RAD)
0110   JY = IY+300*COS(RAD)
0120   CALL DRWABS(JX, JY)
0130 1010 CONTINUE
0140   CALL FINIT(0,0)
0150   END

**GRHIST A:CPLT.DAT②
RUN②
**GRHIST OFF②
^GRCLR②
```

TYPE コマンドで確認します。

```
A>TYPE CPLOT, DAT②
M 0, 504
J 7
M 512, 690
J 1
D 797, 482
J 2
D 688, 147
J 3
D 335, 147
J 4
D 226, 482
J 5
D 511, 690
M 0, 0
```

```
A>②
```

## 7.7 文字単位の会話

キーボードから入力した文字をすぐにセンターへ送信するモードに移るコマンドです。このモードでは、画面編集機能のため送信することができない行端の空白や制御文字を送ることができます。このモードへ遷移するコマンドは2つあります。

^ ASC ②

^ HEX ②

ASC コマンドの場合、 ASCII コードつまり普通の文字で表示します。漢字コードも表示させることができます。一方、HEX コマンドの場合は、送受信した文字を16進数に変換して表示します。グラフィックコードなど制御文字を含んだキャラクタを調べるときに便利です。このとき、漢字を含めて一切の制御コードを解読せずに、16進数へ変換しています。

どちらの場合でも、注意すべき点がいくつかあります。

- このモードでは、スクリーンエディタやグラフィックスなどは利用できません。
- [STOP] または [CTRL]+[Q] でブレーク信号を送出します。このブレーク信号を止めることはできません。
- [ESC] が入力されると、このモードを終了します。 [ESC] を送出することはできません。

## 7.8 その他の機能

### (1) LIST コマンド

^ LIST ファイル名② の入力によって、端末 OS 上のファイルの内容を表示します。ファイルはテキストファイルでなければなりません。

## (2) HELP コマンド

・HELPが入力されると、TSS.HLPという名前のついたファイル名を表示します。

ファイルの表示は、 $13_{16}$ が見つかる度に一時停止して、HELPまたはCTRL+Cが押されるのを待っています。

## (3) MEMORY コマンド

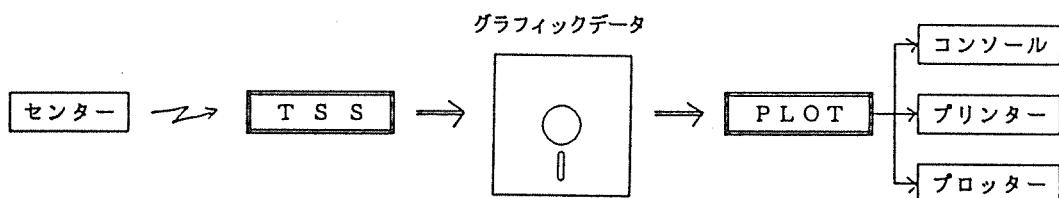
・MEMORYの入力によって、本プログラムがロードされた直後の使用可能なメモリ領域と現在のメモリ領域をKB単位で表示します。

# 第8章 より高度な利用のために

この、TSSのプログラムをさらに有効に利用するための2つのプログラムと他のシステムの端末とするための機能があります。

## 8.1 PLOT プログラム

このプログラムは、グラフィック画面、プリンター、プロッターへ図形を出力するプログラムです。TSSのプログラムとの関係は次のようにになっています。



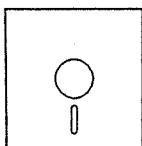
PLOTプログラムは、TSSによって作成されたグラフィックデータのファイルを適当に編集してグラフィック画面、プロッター、プリンターへ出力するものです。グラフィックデータの作成は、「5.6 グラフィックデータのファイルへの記録」を参考にして下さい。

## 8.2 INSTALL プログラム

このプログラムは、プログラムの初期値を設定・変更するものです。TSSの実行形式のファイルを直接読み込み、各種パラメータの初期値を変更するなどの編集を行なったのち、新しい実行形式のファイルとして登録します。

操作例：

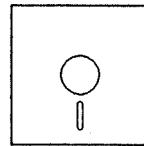
T S S . C M D



読み込む

I N S T A L L

T S S 2 . C M D



(コマンドファイル)

(新しいコマンドファイル)

初期値の設定方法は、「第6章 実行環境の設定」を参考にして下さい。

### 8.3 プリンターの遠隔操作

センターのプログラムによって、端末のプリンターを起動したり停止させたりする機能です。これは、実行結果の一部だけをプリンターに出力したいときなどに便利です。

次の制御コードをセンターから送ることによって、プリンターの動作を指示できます。

12<sub>16</sub> (DC 2, 補助装置の起動) ..... プリンターへの出力を開始する

14<sub>16</sub> (DC 4, 補助装置の停止) ..... プリンターへの出力を終了する

FORTARN 77 プログラムで、プリンターの遠隔操作を利用した例を示します。

```
*LIST②
0010 PRINT *, 'TEST PROGRAM'
0020 PRINT *, CHAR(18)          .... プリンター オン (DC 2 の送出)
0030 PRINT *, 'Printer On'
0040 PRINT *, 'プリンター 動作中'
0050 PRINT *, CHAR(20)          .... プリンター オフ (DC 4 の送出)
0060 PRINT *, 'Printer Off'
0070 PRINT *, 'プリンター 停止'
0080 STOP
0090 END

*RUN②
TEST PROGRAM

Printer On
プリンター 動作中

Printer Off
プリンター 停止
*
```

プリンターには、次のように出力されます。

O		O
O	Printer On プリンター 動作中	O
O		O
O		O

この機能自身を停止させるには、「6.4 会話環境の設定」を参考にして下さい。

#### 8.4 センター以外の端末として利用する

実行環境をホストとなるシステムに合わせて設定すると、他のシステムのターミナルとしても利用できます。ただし、注意すべき点がいくつかあります。

##### (1) RS 232-C パラメータの設定

RS 232-C のパラメータは ACOS-6 / MVX と異なることが多く、ホストとなる機種の仕様にあわせて変更して下さい。変更に際しては、「6.3 通信回線の設定」を参考にして下さい。

##### (2) 画面編集機能

画面編集機能を利用しているときは、が押されるまで送出されませんので、制御文字と行端の空白を送出することができません。制御文字は、端末用コマンドの 1 文字単位の会話 (^ASC) を利用すると入力できます。ただし、 $1B_{16}$  ( ESC ) の入力は、まずできません。

##### (3) エコー機能のオン / オフ

接続したホストに、入力された文字をエコーする機能がある場合、端末のエコー機能をオフにする必要があります。変更するときには、「6.4 会話環境の設定」を参考にして下さい。

##### (4) ブレーク機能

+ でブレーク信号が送出されます。

## 第 9 章 詳細な仕様とその実現方法

### 9.1 制御コードのうち解釈しているもの

#### (1) 漢字コード(初期値)

$1B, 24, 40$ ( ESC, \$, @ )	日本語漢字を指示する ( JIS 制御コード )
$1B, 28, 49$ ( ESC, (, I )	JIS ローマ字を指示する ( JIS 制御コード )
$1A, 60$ ( SUB, p )	漢字インシフトコード ( ACOS 内部コード )
$1A, 61$ ( SUB, q )	漢字アウトシフトコード ( ACOS 内部コード )

#### (2) 画面関係の制御

$07$ ( BEL )	ベルを鳴らす
$0A$ ( LF )	改行

0 C ( FF )	画面消去
1 2 ( DC 2 )	プリンター起動
1 4 ( DC 4 )	プリンター停止
(3) グラフィック関係	
1 6 ( SYN )	同期信号(無視している)
1 C ( FS )	ポイントプロットモードにする
1 D ( GS )	グラフィックモードにする
1 E ( RS )	インクリメンタルプロットモードにする
1 F ( US )	アルファモードにする
1 B, 0 C ( ESC, FF )	グラフィック画面消去
1 B, 1 6 ( ESC, SYN )	同期信号(無視している)
1 B, 1 7 ( ESC, ETB )	ハードコピー出力
1 B, 1 A ( ESC, SUB )	グラフィックインプットモードにする (グラフィックカーソル起動)
1 B, 3 0 ( ESC, 0 )	色の指定を行なう(黒)
1 B, 3 1 ( ESC, 1 )	色の指定を行なう(赤)
1 B, 3 2 ( ESC, 2 )	色の指定を行なう(緑)
1 B, 3 3 ( ESC, 3 )	色の指定を行なう(黄)
1 B, 3 4 ( ESC, 4 )	色の指定を行なう(青)
1 B, 3 5 ( ESC, 5 )	色の指定を行なう(紫)
1 B, 3 6 ( ESC, 6 )	色の指定を行なう(水色)
1 B, 3 7 ( ESC, 7 )	色の指定を行なう(白)
1 B, 3 8 ( ESC, 8 )	文字サイズ極大を選択(無視している)
1 B, 3 9 ( ESC, 9 )	文字サイズ大を選択(無視している)
1 B, 6 0 ( ESC, 9 6 <sub>10</sub> )	ノーマルZ軸・輝度(無視している)
1 B, 6 8 ( ESC, 1 0 4 <sub>10</sub> )	文字サイズ大を選択(無視している)
1 B, 7 F ( ESC, DEL )	抹消(無視している)

## 9.2 スクリーンエディタとその実現について

### (1) スクリーンエディタ

このスクリーンエディタは、行内編集しかできないターミナルを使用しているときにテキストの編集が簡単にできることを目的として設計されています。したがって、単独のマシン

上で走っているスクリーンエディタ（WM, WS, PMATE, SEDIT等）とは、その動作に違いがあります。例えば、②を押されるまでは、センターへテキストを送信することはありませんので、テキストの内容が変更されることはありません。また、②を押しますと、送信されたテキストがプロンプトのところに再び表示されます（「4.1 基本的なテキストの編集」を参考にして下さい）。センターから受信した文字は緑色、入力した文字は水色、送信した文字は黄色（標準）と色分けもしています。

### (2) プロンプトの処理

ただ、”②を押した行を送信する”という方法でスクリーンエディタを実現してもプロンプトの文字は削除して送信する必要があります。このアルゴリズムを説明します。

SYSTEM ?FRT7 ②

と入力した場合を考えます。まず、2つのバッファを用意しておきます。1つはプロンプトバッファで、

プロンプトバッファ： SYSTEM ?

の8文字が入っています。このプロンプトバッファには $0A_{16}$ (LF)のあとで送られてきた最新の文字列が入っています。一方ラインバッファには、②が押された行から後ろ側の空白を削除した文字列を格納します。すると、次のようにラインバッファに入っています。

ラインバッファ： SYSTEM ?FRT7

この2つのバッファを比較して、ラインバッファ中の前の8文字がプロンプトであると判断して残ったFRT7だけをセンターへ送信します。プロンプトバッファとラインバッファの比較をする際、数字どうしは一致していないくとも、その桁が一致しているだけでプロンプトであるとします。これは、AUTOコマンドを使って入力しているときの便宜を図っているためです。ただし、パスワードを入力しているときに問題が生じます。このとき、ラインバッファとプロンプトバッファは次のようになっています。

プロンプトバッファ： TEWPJFIEPVR

ラインバッファ：PASWORDEPVR

ここで、前述したアルゴリズムを用いますと、先頭の1文字目から異なりますのでラインバッファをすべて送信してしまいます。そこで、本プログラムでは先頭文字からのチェックに加えて、後ろから一致する文字を検出しています。この場合 EPVR の4文字が一致しているため、パスワード入力のためのプロンプトであると判断して最終的にPASSWORDの4文字を送信します。

### (3) テキストバッファの管理

このスクリーンエディタはテキストを画面のサイズを越えて記憶しています。テキストを

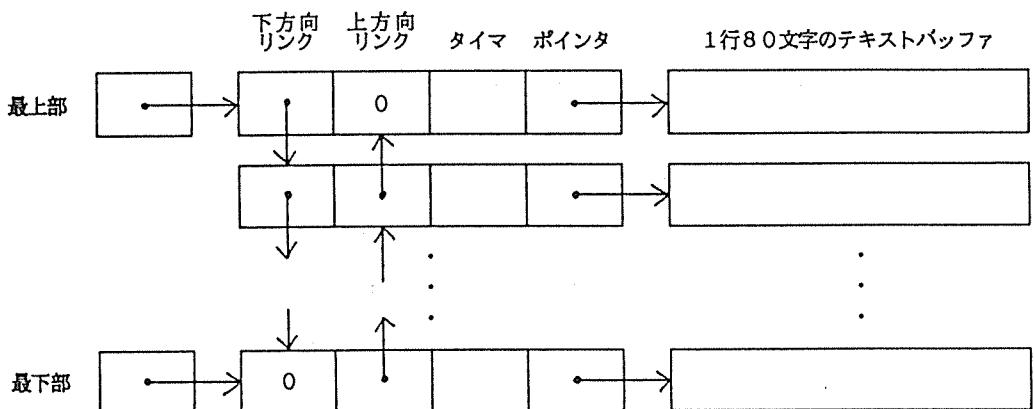
800行(実際は50行~2600行の間で可変)記憶していく、現在の画面の位置から1行あるいは1頁単位で移動して見ることができます。

ところが、9600ポードで文字が送られてきている最中に、改行のたびに800行をスクロールすることは不可能であり、双方向リンクリストをフルに活用して処理の高速化を図っています。以下に、データ構造といくつかの手続きについて説明します。

テキストバッファは大きく、

- ・画面前方バッファ (画面の上から消えたテキスト)
- ・画面中央バッファ (現在画面に表示しているテキスト)
- ・画面後方バッファ (画面の下から消えたテキスト)

の3つに分けて、それぞれを次のようなリンクリストによって管理します。



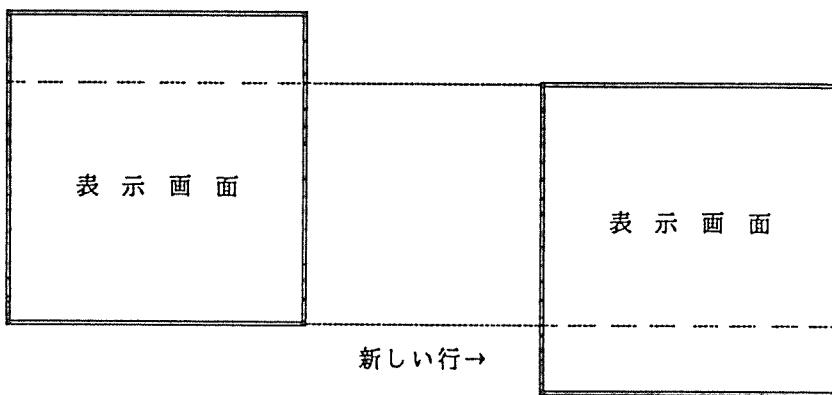
このように、データ構造は、テキストバッファとそのテキストバッファへのポインタで構成されており、ポインタのテーブルは双方向のリンクリストで接続されています。このテキストバッファに対して外部から行なう操作として、4つの手続きが用意されています。

- SCROLL\_UP 現在の画面を1行上へ移動する
- SCROLL\_DOWN 現在の画面を1行下へ移動する
- NEW\_LINE 新しい行を獲得して画面の最下部につけ加える  
画面上の(x, y)で示された位置に対応する  
テキストバッファのアドレスを与える。
- CHK\_ADDR

この中から、NEW\_LINEについて説明します。表示されている画面の最下部で、センターからの改行コードを受信すると画面の文字を1行上にあげて最下部へ空白で埋められた行を挿入しなければなりません。

改行コードが送られてくる前

改行コードが送られてきた後



このとき、リンクテーブルに対する操作は次のようにになります。

①新しい行を獲得する

画面前方最上部と画面後方最下部のうち先に登録された行を新しい行とします。

登録された時間に関する情報はリンクテーブルのタイマにあります。画面前方最上部を新しい行とするときは、そのリンクテーブルを画面前方のリンクリストからはずします。

②新しい行を現在の画面に付け加える

新しい行のテキストバッファを空白でクリアしてから、新しい行のリンクテーブルを画面中央リンクリストの最下部に付け加えます。

③現在の画面の1番上の行をとり出す

画面中央最上部のリンクテーブルをリンクリストからはずします。

④取り出した行をバッファに付け加える

取り出したリンクテーブルを画面前方のリンクリストの最下部に付け加えます。

このとき、リンクテーブルの中のタイマを更新します。つまりリンクテーブルをリンクリストに付け加えるたびにリンクリストを管理しているシステムがもっているクロックを更新し、クロックの値をリンクテーブルの中のタイマに登録します。

以上の4つの段階で、NEW\_LINEが実現されています。手順は、一見複雑にみえますが、メモリに対するアクセスはわずかであり、テキストを実際に動かすことは全く行なっていません。

### 9.3 グラフィックコードの解釈について

このプログラムでは、テクトロニクス4010シリーズのエミュレート機能を備えており、グラフィックコードがセンターから送信されると、グラフィック表示を行ないます。

本エミュレータでサポートするモードは、以下の5種類です。

- アルファモード
- グラフィックモード
- グラフィックインプットモード(GIN)
- ポイントプロットモード
- インクリメンタルプロットモード

#### 9.3.1 アルファモード

アルファモードは文字を表示するモードであり、グラフィック以外の制御コードを受信すると本モードへ移行します。アルファモード時は、通常のキャラクタ端末であり、例えばページフル機能や文字サイズの指定等のエミュレートは、行いません。

#### 9.3.2 グラフィックモード

グラフィックモードは、線画を行なうモードです。GSコードを受信することにより、本モードに移行します。

##### (1) アドレスデータ

アドレスデータは、4010型(4012型)のとき HIY, LOY, HIX, LOX の4バイトで、4014型の場合は HIY, EXT-B, LOY, HIX, LOX の5バイトで指定します。

アドレ ス データバイト名	7ビット ASCII キャラクタ						
	TAGビット		アドレスビット				
	6	5	4	3	2	1	0
HIY	0	1	Yアドレス 上位 5ビット				
EXT-B	1	1	0	Y <sub>2</sub>	Y <sub>1</sub>	X <sub>2</sub>	X <sub>1</sub>
LOY	1	1	Yアドレス 下位 5ビット				
HIX	0	1	Xアドレス 上位 5ビット				
LOX	1	0	Xアドレス 下位 5ビット				

TAGビットはバイトデータの順序をチェックするためのものであります。EXT-Bは、他のビットデータに対して下位の2ビットを補います。すなわち、4010型・4014型では、それぞれ次のように解釈します。

4010型:  $\begin{array}{ccccccccc} X_0 & X_8 & X_7 & X_6 & X_5 & X_4 & X_3 & X_2 & X_1 \\ \hline H & I & X & (5ビット) & L & O & X & (5ビット) & E & X & T-B \end{array}$

$\begin{array}{ccccccccc} Y_0 & Y_8 & Y_7 & Y_6 & Y_5 & Y_4 & Y_3 & Y_2 & Y_1 & Y_0 & Y_{-1} & Y_{-2} \\ \hline H & I & Y & (5ビット) & L & O & Y & (5ビット) & E & X & T-B \end{array}$

4014型:  $\begin{array}{ccccccccc} X_{14} & X_{10} & X_9 & X_8 & X_7 & X_6 & X_5 & X_4 & X_3 & X_2 & X_1 & X_0 \\ \hline H & I & X & (5ビット) & L & O & X & (5ビット) & E & X & T-B \end{array}$

$\begin{array}{ccccccccc} Y_{14} & Y_{10} & Y_9 & Y_8 & Y_7 & Y_6 & Y_5 & Y_4 & Y_3 & Y_2 & Y_1 & Y_0 \\ \hline H & I & Y & (5ビット) & L & O & Y & (5ビット) & E & X & T-B \end{array}$

内部では、整数で処理しているため4010型ではEXT-Bを無視しています。その結果、X・Yアドレスとともに、4010型のとき0～1023の数値が表現可能であり、4014型の場合0～4095の数値を表現することができます。

常に4バイト(EGMの場合5バイト)を送っていたのではデータ量が多くなります。短い線を描く場合など、伝送効率を上げるために前回アドレスと同じデータバイトが省略されることがあります。下図の●は必ず送信されなければならないデータを示しており、空白は省略されると前回のアドレスのデータが充当されます。EXT-Bは省略されると、Y<sub>2</sub>Y<sub>1</sub>X<sub>2</sub>X<sub>1</sub>ビットがオール0とみなされます。

変化するデータ	送られるべきデータ				転送バイト数
	H I Y	L O Y	H I X	L O X	
H I Y	●			●	2バイト
L O Y		●	●	●	2バイト
H I X			●	●	3バイト
L O Y				●	1バイト

### (3) 画面上の動き

GSコードの直後に送られてきたアドレスデータで指定された位置に描画座標が移動します。引き続いてアドレスデータがくると、その位置まで直線が描かれ描画座標も再び移動します。

### 9.3.3 グラフィックインプットモード

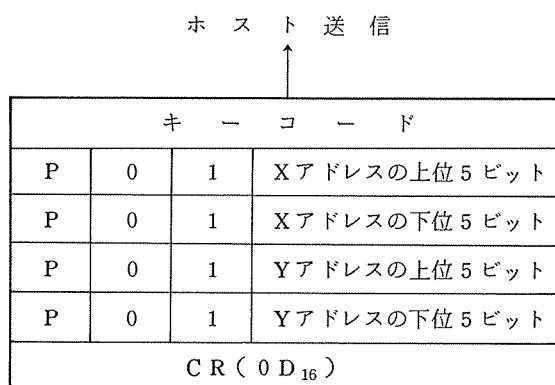
端末からセンターへカーソル位置を送信するモードです。ESC, SUBを受信することにより本モードに移行します。

#### (1) 画面上の動き

本モードに移行するとクロスヘアカーソルが表示され、キーボードのカーソル移動キーによって自由にその位置を変えることができます。

#### (2) アドレス送信

カーソルアドレスは、キーボードのキーが押されたときに送信されます。アドレスデータの送信形式は下記のとおりです。



### 9.3.4 ポイントプロットモード

ポイントプロットモードは、グラフィックモードで直線を描く代わりに指定座標位置に点を書き込むモードです。FSコードを受信することにより本モードに移行します。アドレスの指定方法はグラフィックモード時と全く同様です。

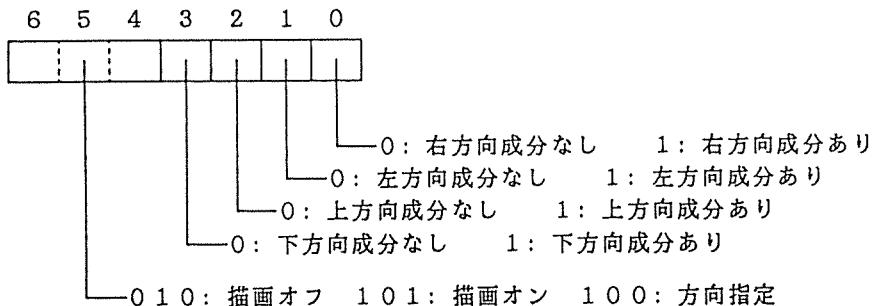
### 9.3.5 インクリメンタルプロットモード

インクリメンタルプロットモードは、現在の描画座標から指定された方向に、点を描画するモードです。RSコードを受信することにより本モードに移行します。RSの次にくるキャラクタで、点のオン/オフを決め、第三バイト目からをプロット方向のデータと見なし、点をプロットします。

#### (1) 方向指定データ

方向指定データは、現在の描画座標からどの方向へ進むかを指定するデータです。データの機能を次の表に示します。

データ	ビット							指定動作
	6	5	4	3	2	1	0	
空白 (20 <sub>16</sub> )	0	1	0	0	0	0	0	描画オン
P (50 <sub>16</sub> )	1	0	1	0	0	0	0	描画オフ
D (44 <sub>16</sub> )	1	0	0	0	1	0	0	上
E (45 <sub>16</sub> )	1	0	0	0	1	0	1	右上
A (41 <sub>16</sub> )	1	0	0	0	0	0	1	右
I (49 <sub>16</sub> )	1	0	0	1	0	0	1	右下
H (48 <sub>16</sub> )	1	0	0	1	0	0	0	下
J (4A <sub>16</sub> )	1	0	0	1	0	1	0	左下
B (42 <sub>16</sub> )	1	0	0	0	0	1	0	左
F (46 <sub>16</sub> )	1	0	0	0	1	1	0	左上



## (2) 画面上の動き

現在の描画座標から方向指定データにそって 1 ドットずつ移動します。

### 9.4 オペレーティング・システムの利用について

このプログラムでは、いくつかの周辺機器との入出を行なっていますが、入出力の速度が問題になる部分は、アセンブラーによって最適なマシンコードを生成しています。ただし、以下の 4 つの機能についてはオペレーティング・システムのシステムコールを用いて実現しています。

- キーボードからの入力
- コンソールへの出力
- ディスクとの入出力
- メモリエリアの管理

その他一部に拡張機能呼び出しや BIOS コマンド (ROM 内ルーチン) の中で、使っているものがあります。そこで、実際に利用しているサブルーチンの一覧を示します。

(1) ai-M16 (CP/M-86)

基本ディスク・オペレーティング・システムの呼び出し(割込コード E0H)

機能コード 0 :	システム・リセット
機能コード 5 :	リストへの出力
機能コード 15 :	ファイルのオープン
機能コード 16 :	ファイルのクローズ
機能コード 20 :	シーケンシャルな読出し
機能コード 21 :	シーケンシャルな書込み
機能コード 22 :	ファイルの作成
機能コード 24 :	ログイン・ベクトルの取出し
機能コード 37 :	ドライブのリセット
機能コード 53 :	利用可能最大メモリの獲得
機能コード 54 :	メモリ領域の獲得
機能コード 57 :	メモリ領域の解放

(2) PC-9801 (CP/M-86)

基本ディスク・オペレーティング・システムの呼び出し(割込コード E0H)

機能コード 0 :	システム・リセット
機能コード 6 :	コンソールによる直接入出力
機能コード 15 :	ファイルのオープン
機能コード 16 :	ファイルのクローズ
機能コード 20 :	シーケンシャルな読出し
機能コード 21 :	シーケンシャルな書込み
機能コード 22 :	ファイルの作成
機能コード 24 :	ログイン・ベクトルの取出し
機能コード 37 :	ドライブのリセット
機能コード 53 :	利用可能最大メモリの獲得
機能コード 54 :	メモリ領域の獲得
機能コード 57 :	メモリ領域の解放

拡張機能の呼び出し(割込コード DCH)

機能コード 12 :	ファンクションキーやカーソル移動キーなどの取得
機能コード 13 :	ファンクションキーやカーソル移動キーなどの設定

### **B IOS コマンドの呼び出し(割込コード 1AH)**

識別コード 11H : セントロニクス仕様プリンタへ1バイトを出力

#### **(3) PC-9801 (MS-DOS)**

##### **ファンクションリクエスト(割込コード 21H)**

ファンクション 06H : 直接コンソール I/O

ファンクション 0DH : リセットディスク

ファンクション 33H : <Ctrl-C>検査のセット/リセット

ファンクション 3CH : ハンドルの作成

ファンクション 3DH : ハンドルのオープン

ファンクション 3EH : ハンドルのクローズ

ファンクション 3FH : リードハンドル

ファンクション 40H : ライトハンドル

ファンクション 48H : メモリの割り当て

ファンクション 4AH : 割り当てられたメモリブロックの変更

ファンクション 4CH : プロセスの終了

##### **拡張機能の呼び出し(割込コード DCH)**

機能コード 12 : ファンクションキー/カーソル移動キーなどの取得

機能コード 13 : ファンクションキー/カーソル移動キーなどの設定

### **B IOS コマンドの呼び出し(割込コード 1AH)**

識別コード 11H : セントロニクス仕様プリンタへ1バイトを出力

直接制御している周辺デバイスやメモリ領域をまとめておきます。

#### **(1) ai-M16**

##### **RS-232C 関連**

STC Am 9513 : システム・タイマ・コントローラ

PIC 8259A-2 : プログラマブル・インターラプト・コントローラ

SIO<sub>μ</sub>PD7201 : シリアル入出力コントローラ

##### **割込みポインタテーブル**

84～87H : 割込みタイプ32 (SIO割込)

#### **(2) PC-9801 (CP/M-86)**

##### **RS-232C 関連**

<sub>μ</sub>PD8259A : 割込コントローラ

<sub>μ</sub>PD8251A : RS-232C インターフェース

$\mu$ PD8255A-5 : システムポート  
 $\mu$ PD8253-5 : タイマコントローラ  
割込みベクタテーブル  
18~1BH : 割込みベクタ6 (STOP割込)  
30~33H : 割込みベクタ12 (RS-232C割込)

#### キャラクタ表示関連

A0000~A0F9EH : テキスト用VRAM表示エリア  
A2000~A2F9EH : テキスト用VRAMアトリビュートエリア

#### グラフィック表示関連

$\mu$ PD7220 : グラフィック・ディスプレイ・コントローラ  
A8000~AFCFFH : グラフィック用VRAM (ブルー)  
B0000~B7CFFH : グラフィック用VRAM (レッド)  
B8000~BFCFFH : グラフィック用VRAM (グリーン)

### (3) PC-9801 (MS-DOS)

#### RS-232C 関連

$\mu$ PD8259A : 割込コントローラ  
 $\mu$ PD8251A : RS-232C インターフェース  
 $\mu$ PD8255A-5 : システムポート  
 $\mu$ PD8253-5 : タイマコントローラ

#### 割込みベクタテーブル

18~1BH : 割込みベクタ6 (STOP割込)  
30~33H : 割込みベクタ12 (RS-232C割込)  
90~93H : 割込みベクタ36 (INT 24H)

#### キャラクタ表示関連

A0000~A0F9EH : テキスト用VRAM表示エリア  
A2000~A2F9EH : テキスト用VRAMアトリビュートエリア

#### グラフィック表示関連

$\mu$ PD7220 : グラフィック・ディスプレイ・コントローラ  
A8000~AFCFFH : グラフィック用VRAM (ブルー)  
B0000~B7CFFH : グラフィック用VRAM (レッド)  
B8000~BFCFFH : グラフィック用VRAM (グリーン)

## 9.5 周辺装置との入出力について

### (1) RS-232C の割込み処理 ( ai-M16 )

SIO7201 と PIC8259 を直接コントロールすることによって、入出力とも割込みで処理しています。6つの SIO が同一の割込みタイプであるため、コンソールとの文字の入出力も作成した割込みルーチンで処理をしています。そのため、CTRL+SHIFT+'0' による強制割込みは使用できません。さらに、コンソールとのインターフェースは、転送レートが 19200 ボーでもできますが、逆に低いボーレートではファンクションキーが正常に動作しないことがあります。

### (2) RS-232C の割込み処理 ( PC-9801 )

送信バッファレディと受信バッファエンプティの割込み信号線が、同一の割込み線を利用しているため、確実に 2 つの割込みを区別することができません。したがって、受信には割込みを使っていますが、送信にはポーリングを用いています。このプログラムに使っている割込み処理ルーチンは、徹底的に最適化を図っていますので、OS の中にあるルーチンよりも高速になっているはずです。

### (3) RS-232C のフロー制御

受信バッファは 256 バイトの容量をもっています。受信フロー制御のスイッチがオンになつていれば、受信バッファに 40 文字以上たまつたときに X-OFF キャラクタ (DC3 コード) を送出し、10 文字以下に減つたときに X-ON キャラクタ (DC1 コード) を送出しています。

### (4) ディスク

ファイルごとに 8 K バイトの容量をもつバッファをダイナミックに作成して、入出力を行なっています。さらに、RS-232C とディスクのアクセスに関する割込みが衝突するのを避けるため、ディスクへアクセスする前に X-OFF を送信し、キャラクタが受信されなくなつたのを確認してから、ディスクとの間でデータの転送を行なっています。そして、ディスクのアクセスが終了すれば、X-ON を送出しています。

ディスクとの入出力をこのようにしたのは、データを受信中しながらディスクへ書き込もうとしたとき (ダウンロード時) に、ディスクの入出力の割込みと RS-232C の割込みの衝突が頻繁に発生し、ディスクのほうが優先されてしまうからです。そのうえ、センターのフロー制御は、センターに対して X-OFF と X-ON のキャラクタを連続して送信すると、回線そのものが何故か切断されてしまいます (この現象は、センター側でも確認済)。そこで、8 K バイトのバッファを使って、X-OFF と X-ON のコードの送信回数を減らし、送信間隔をあけています。

## (5) 漢字コードの入出力

RS-232Cを通した漢字の入出力は、JISの漢字コードが漢字インシフトコードと漢字アウトシフトコードではさまれたキャラクタ列によって行なっています。シフトJIS漢字コードは、RS-232Cを通して送受信することはできません。それは、送受信のための制御コードとシフトJISの一部のキャラクタが重なっているからです。

センターは、端末を漢字属性としたときだけ、JIS制御文字を用いて送受信をするとなっています。しかし、ACOS-6/MVXは現在(1986年11月)、JIS規格のC6226-1978の改正前の版の符号拡張制御文字を採用しています。したがって、JIS C 6266-1978やJIS C 6266-1983のいずれの規格にも適合していません(ACOSで使用している制御文字は、ISOの規格によるとスウェーデンの図形文字集合を選択しています)。この新しい漢字の制御文字は、将来追加される予定があり、新しい制御文字にも対応できるように、このプログラムではユーザが制御文字を簡単に変更できるようになっています。

また、他の属性の端末として接続した場合は、ACOSの内部コードがそのまま端末へ送信されますが、設定によりこのときも対応できるようにしています。

## 参考文献

### 1. デバイス関連

μCOM-86/87 ユーザーズ・マニュアル	(NEC IEM-729D)
μPD8251AFの使い方	(NEC IEP-647A)
μPD8253-2の使い方	(NEC IEP-681A)
μPD8255A-2の使い方	(NEC IEP-673A)
μPD7201 MPSC ユーザーズ・マニュアル	(NEC IEP-723B)
μPD7220 GDC ユーザーズ・マニュアル	(NEC IEP-734I)
Am9513 Applications Manual	(Advanced Micro Device)
iAPX86 ファミリ・ユーザーズ・マニュアル	(インテル・ジャパン)

### 2. ターミナル関連

PC-9800 シリーズテクニカルマニュアル	(日本電気情報処理OAシステム事業部)
ai-M16 16ビット・マイクロ・コンピュータ・ハードウェア解説書	(アイ電子測器)
PC-9801E USER'S MANUAL	(日本電気)
PC-9801VF2/VM0/VM2 ユーザーズ・マニュアル	(日本電気)

インストラクション・マニュアル 4010型

(ソニー・テクトロニクス)

D-SCAN システム解説書 グラフィックディスプレイ (セイコー電子 S407-30079-D)

### 3. OS 関連

CP/M-86 システムガイド (デジタル・リサーチ)

CP/M-86<sup>TM</sup> 1.1 Operating System USER'S MANUAL (日本電気)

CP/M-86<sup>TM</sup> 1.1 Operating System PROGRAMMER'S MANUAL (日本電気)

MS-DOS<sup>TM</sup> 3.1 ユーザーズ マニュアル (日本電気)

MS-DOS<sup>TM</sup> 3.1 プログラマーズリファレンス マニュアル (日本電気)

### 4. 周辺機器

ミニエース漢字ライタ(24W) USER'S MANUAL (NEC DOI-E04601-1)

PC-PR 201H2 日本語シリアルプリンタ ユーザーズ・マニュアル  
(NEC PC-PR 201H2-UM)

グラフィック・ビデオ・ディスプレイ・ユニット VDU-140 ユーザーズ・マニュアル  
(アイ電子測器)

プリンター M82W 取扱説明書

パソコンプロッタ FP5301 取扱説明書 (グラフティック FP5301-UM-101)

### 5. オンライン関連(大阪大学大型計算機センターニュース)

小田晃一：「マイコンによるTSSグラフィック端末」 Vol.12 №2 (1983) 79-98.

藤井 博：「パソコンコンピューターを大型計算機センターの端末に」  
Vol.12 №2 (1983) 104-120.

北本昇一, 大西正一, 中島聖勝：「漢字端末について」 Vol.12 №3 (1983) 97-103.

国本雅夫：「PC-8801を用いたTSSグラフィック端末プログラム」  
Vol.13 №4 (1984) 99-112.

国本雅夫：「PC-9801Fを用いたグラフィック&漢字ターミナルプログラム」  
Vol.14 №4 (1985) 75-101.

### 6. センターソフトウェア関連

图形処理の手引 第2版 (大阪大学大型計算機センター)

图形処理基本システム説明書<GDSP-2/4/6 PLOT> (日本電気 FXG01-4)

TSS-AF システム説明書 (日本電気 FXG01-4)

### 7. 関連規格

JIS C 6220-1976 情報交換符号

JIS C 6226-1978 情報交換用漢字符号系

JIS C 6226-1983	情報交換用漢字符号系
JIS C 6228-1984	情報交換用符号の拡張法
JIS C 6225-1979	情報交換用漢字符号系のための制御文字符号

TSS インテリジェント ターミナル ソフトウェア  
操作 説 明 書

1986年8月 初版 発行

1986年10月 改訂版 発行

著者 濱村秀彦  
発行 大阪大学 工学部 精密工学科  
牧之内研究室  
大阪府吹田市山田丘2-1  
TEL (06) 877-5111  
内線 4628

