

JAERI-M

6 6 1 9

グラフィック・ディスプレイのプログラミング

1976年7月

中 村 康 弘・小 沼 吉 男

日本原子力研究所  
Japan Atomic Energy Research Institute

この報告書は、日本原子力研究所が JAERI-M レポートとして、不定期に刊行している研究報告書です。入手、複製などのお問合せは、日本原子力研究所技術情報部（茨城県那珂郡東海村）あて、お申しこしください。

JAERI-M reports, issued irregularly, describe the results of research works carried out in JAERI. Inquiries about the availability of reports and their reproduction should be addressed to Division of Technical Information, Japan Atomic Energy Research Institute, Tokai-mura, Naka-gun, Ibaraki-ken, Japan.

グラフィック・ディスプレイのプログラミング

日本原子力研究所東海研究所計算センター

中村 康弘・小沼 吉男

(1976年6月18日受理)

グラフィック・ディスプレイは対話型グラフィック処理に有用な入出力装置であり、原研ではリフレッシュ型のグラフィック・ディスプレイ FACOM 6233 AをFACOM 230-75 システムの下で利用している。本報告では、このグラフィック・ディスプレイの概要、ならびにPGSLIBによるプログラミングとその実行の方法について述べた。

JAERI-M 6619

Graphic Display and the Programming

Yasuhiro NAKAMURA and Yoshio ONUMA

Computing Center, Tokai, JAERI

(Received June 18, 1976)

Graphic display is an input/output device for interactive computer graphics. Refresh-type graphic display FACOM 6233A is used in system FACOM 230-75, JAERI. Programming and executing methods using PGSLIB for the graphic display are described.

## 目 次

1 はじめに	1
2 グラフィック・ディスプレイ装置	2
2.1 CRT面	2
2.2 ライトペン	2
2.3 ファンクション・キーボード	4
2.4 文字キーボード	4
2.5 ハード・コピー装置	4
3 グラフィック・プログラミング	6
3.1 PGSLIBの概要	6
3.2 PGSによる図形表示のプログラミング	6
3.2.1 PGSの命令	6
3.2.2 PGSボタン	8
3.2.3 座標系と文字	8
3.3 簡易GSPによる対話処理のプログラミング	10
3.3.1 簡易GSPの命令	10
3.4 汎用GSPによるグラフィック・プログラミング	11
3.4.1 汎用GSPの命令	11
3.5 グラフィック・プログラムの例	13
4 グラフィック・プログラムの実行	16
4.1 グラフィック用ジョブ制御文	16
4.2 グラフィック・ディスプレイの使用	18
4.2.1 セミ・オープン窓	18
4.2.2 グラフィック・ジョブの実行	18
4.2.3 コメント入力	18
4.2.4 ハード・コピー	19
4.3 エラー・メッセージ	19
5 コメント入力	22
5.1 機能	22
5.2 操作	23
6 ハード・コピー	25
6.1 機能	25
6.2 操作	25
7 おわりに	32
参考文献	32
付録 PGSLIB/BASIC命令の呼出し形式	34

## 目 次

1 はじめに	1
2 グラフィック・ディスプレイ装置	2
2.1 CRT面	2
2.2 ライトペン	2
2.3 ファンクション・キーボード	4
2.4 文字キーボード	4
2.5 ハード・コピー装置	4
3 グラフィック・プログラミング	6
3.1 PGSLIBの概要	6
3.2 PGSによる図形表示のプログラミング	6
3.2.1 PGSの命令	6
3.2.2 PGSボタン	8
3.2.3 座標系と文字	8
3.3 簡易GSPによる対話処理のプログラミング	10
3.3.1 簡易GSPの命令	10
3.4 汎用GSPによるグラフィック・プログラミング	11
3.4.1 汎用GSPの命令	11
3.5 グラフィック・プログラムの例	13
4 グラフィック・プログラムの実行	16
4.1 グラフィック用ジョブ制御文	16
4.2 グラフィック・ディスプレイの使用	18
4.2.1 セミ・オープン窓	18
4.2.2 グラフィック・ジョブの実行	18
4.2.3 コメント入力	18
4.2.4 ハード・コピー	19
4.3 エラー・メッセージ	19
5 コメント入力	22
5.1 機能	22
5.2 操作	23
6 ハード・コピー	25
6.1 機能	25
6.2 操作	25
7 おわりに	32
参考文献	32
付録 PGSLIB/BASIC命令の呼出し形式	34

## 1 はじめに

グラフィック・ディスプレイ<sup>(1),(2)</sup>は、現在電子計算機に接続できる最も優れた図形入出力装置である。このグラフィック・ディスプレイによって利用者は計算機からの多量の情報をC R T (Cathode Ray Tube) 面に高速度で図形表示でき、また、ライトペン、ファンクション・キーボードまたは文字キーボードを介して計算機へ情報を直接入力できる。このような図形入出力機能により、グラフィック・ディスプレイは人間が計算機と対話をしながら多量の情報を処理し、判断を下してゆく場合に有効な道具であり、近年、計算機による情報処理の多くの分野で利用されている。

特に原子力計算の分野においては、対話形式による、ガンマ線スペクトルの解析<sup>(3)</sup>、核断面積データの処理<sup>(4)</sup>、実験データのカーブ・フィッティング<sup>(5)</sup>、および原子炉の地震シミュレーション解析<sup>(6)</sup>などのために約5年前からグラフィック・ディスプレイが利用され<sup>(8)</sup>、それが非常に有効であることが実証されている。最近では各種実験データの処理や解析<sup>(7)</sup>、核融合計算の分野での利用<sup>(8)</sup>が多い。また、プロッタやグラフィック COM (Computer Output Microfilm System) プログラムのオンライン・デッキ<sup>(9)</sup>用、および事務計算における手軽な図形出力用としても利用されている。

このようなグラフィック・ディスプレイを使用するためにはプログラムを作成しなければならないが、このプログラム作成は筆者らによって開発されたグラフィック・プログラミング用ソフトウェア PGSLIB<sup>(10)</sup> (Plotter-compatible Graphic Subroutine Library) を用いて行うことができる。PGSLIBの場合、プロッタやCOM用の図形出力プログラムがそのままグラフィック・ディスプレイの図形表示プログラムとして利用でき、かつ、この図形表示プログラムにライトペンや文字キー・ボードなどの使用に関するプログラムを追加するだけで本格的な対話型グラフィック・プログラムが作成できる。また、PGSLIBには文字キーボードによる、CRT面上に表示された図形に対する日付、利用者名、タイトルと説明文などのコメント入力の機能、およびライトペンによる、CRT面上に表示された図形のCOM、またはプロッタへのハード・コピー出力の機能があり、しかもこれらの機能が簡単なプログラミングと操作で使用できる特長がある。かくして、PGSLIBによって有効なグラフィック・プログラムがGSP<sup>(2),(11)</sup> (Graphic Subroutine Package) によるよりもずっと容易に作成できる。

以下、現在、FACOM 230-75で利用できるリフレッシュ型（再生型）のグラフィック・ディスプレイ FACOM 6233A<sup>(1)</sup>について、まずそのハードウェアの概要を述べ、つぎに、PGSLIBによるグラフィック・プログラムの作成からFACOM 230-75によるプログラムの実行までの方法について述べる。さらに、コメント入力の機能と操作、およびハード・コピーの機能と操作についても述べる。

## 2 グラフィック・ディスプレイ装置

グラフィック・ディスプレイでは、そのCRT面によって計算機からの多量のデータを高速度で図形表示でき、また、そのライトペン、ファンクション・キーボード、および文字キーボードによって計算機に指示を与えること、データを入力することができる。また、CRT面上に表示された図形はハードウェアのハード・コピー装置によって紙の上にハード・コピーできる。

ここでは、グラフィック・ディスプレイ装置FACOM 6233A<sup>(1)</sup>(Fig. 2.1)のCRT面、ライトペン、ファンクション・キーボード、および文字キーボードについてそのハードウェア機能の概要を述べる。また、FACOM 6541Bハード・コピー装置についても簡単に述べる。

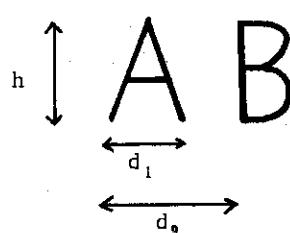
### 2.1 CRT面

グラフィック・ディスプレイ装置のCRT面は、つきのFig. 2.2に示すように直径50cmの円形であり、X、Yの両方向に各々4096の格子点に分割されている。また、このディスプレイ装置はリフレッシュ型であり、表示する図形のデータ（ディスプレイ語とよぶ）をバッファ・メモリ（8K語、1語=16ビット）に記憶しておき、毎秒約40回図形を繰り返し再生して描くことにより、ちらつきのない図形表示を可能にする。さらに、3種類の直線（実線、点線、破線）の表示、4種類の文字サイズ（Table 2.1）による英数字と特殊記号の表示、および、つきの2.2で述べるような図形単位ごとに4種類の輝度による図形表示が可能である。

Table 2.1 文字の大きさ

文字の大きさ	$d_1$	$d_2$	$h$
VSM ( very small )	2.1	2.9	2.3
SML ( small )	3.1	4.4	4.2
LAG ( large )	4.9	6.8	6.5
VLG ( very large )	7.0	9.8	9.3

（単位はmm）



### 2.2 ライトペン

ライトペンはつきのFig. 2.3のような構造であり、CRT面上に表示された図形の認識とライトペン・スイッチによる割込みの機能を有する。また、ライトペンの使い方には、表示された図形のピッキング（ポインティング）とトラッキング・シンボルによるペン・トラッキング（ポジショニング）

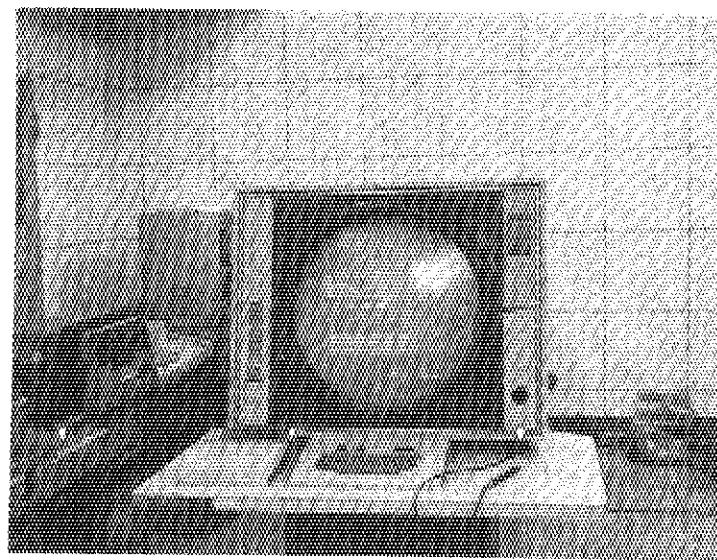


Fig. 2.1 FACOM 6233A グラフィック・ディスプレイ装置

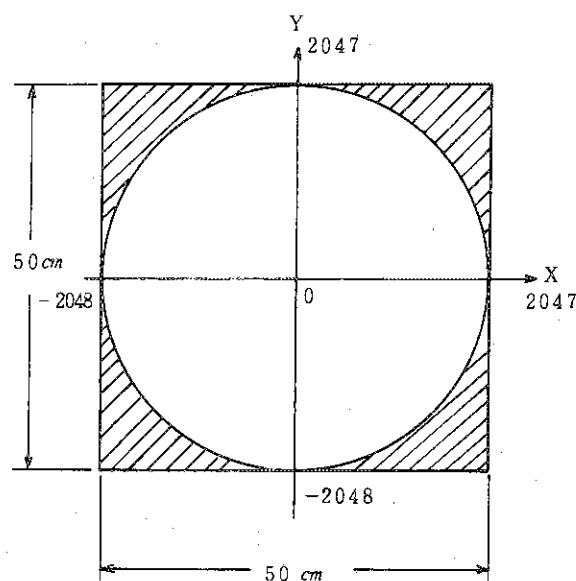


Fig. 2.2 C R T 面

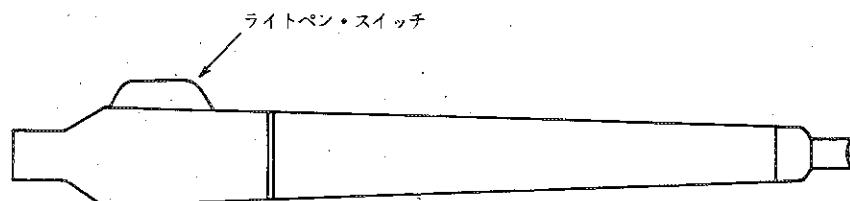


Fig. 2.3 ライトペン

がある。

ある一まとめの図形単位をエレメントといい、エレメントには名前付きとそうでないものがある。名前付きエレメントは、ライトペン・スイッチをオンにしたままその図形にライトペンを近づけるとウインクする。ウインクによって利用者はどのエレメントをピックしたかがわかる。ウインク中にライトペン・スイッチをオフにすると割込みが起り、その図形エレメントが認識されることになる。これがライトペンによるピッキングである。ピッキングの場合は選択すべきエレメントのウインクを十分確認してから、ライトペン・スイッチをオフにする必要がある。そうでないと別のエレメントを選択しかねない。

つぎに、トラッキング・シンボルは十字形のシンボル '+' であり、これはライトペン・スイッチをオンにしたまま近づけるとライトペンの動きに追従する。これがトラッキング・シンボルによるペン・トラッキングである。トラッキング・シンボルについてはその位置座標を記憶する X, Y カウンタがあり、そのカウンタの値はシンボルが動くと更新され、つねにシンボルの現在位置を記憶する。これを利用してライトペンによるトラッキング・シンボルの軌跡を計算機に入力することができる。

### 2.3 ファンクション・キーボード

ファンクション・キーボードはつきの Fig. 2.4 に示すように、32 個のランプ付きのファンクション・キーと 8 個のオーバレイ・スイッチで構成される。各キーおよびランプの機能は利用者のプログラムにより決定される。キーを押すとスイッチが閉じた時と開いた時に計算機へ割込み信号が送られる。

### 2.4 文字キーボード

文字キーボードはつきの Fig. 2.5 に示すように、タイプライタ状のキーボードであり、一連の文字キーと制御キーより構成される。キーを押すと割込みが発生し、キーに対応する文字コードを送出する。**英数**, **英記号** のキーは、シフトの状態を変更するのみで割込みは発生しない。**英数** は英数字を入力するとき、また **英記号** は特殊記号を入力するときに前もって使用するキーである。4 個の制御キー **END**, **CANCEL**, **←**, **→** はシフトの状態に関係なく使用できる。**CANCEL** は一行全体を修正するとき、**←** と **→** は一部分を修正するときに使用するキーであり、**END** は入力の終りを示すキーである。

### 2.5 ハード・コピー装置

FACOM 6233A グラフィック・ディスプレイ装置に付属する FACOM 6541B ハード・コピー装置は、ディスプレイ装置の CRT 面に表示された図形を紙の上にハード・コピーできる。ハード・コピー装置は内蔵の記録用 CRT 面にグラフィック・ディスプレイ装置の CRT 面と同じ図形を再生し、それをエレクトロ・ファックス電子写真法により酸化亜鉛紙の上にプリントする。プリント・サイズは A4 巾で長さは約 20 cm である。ハード・コピーは **PRINT** キーを押すことによって一枚当り平均約 30 ~ 40 秒の待ち時間で得られる。

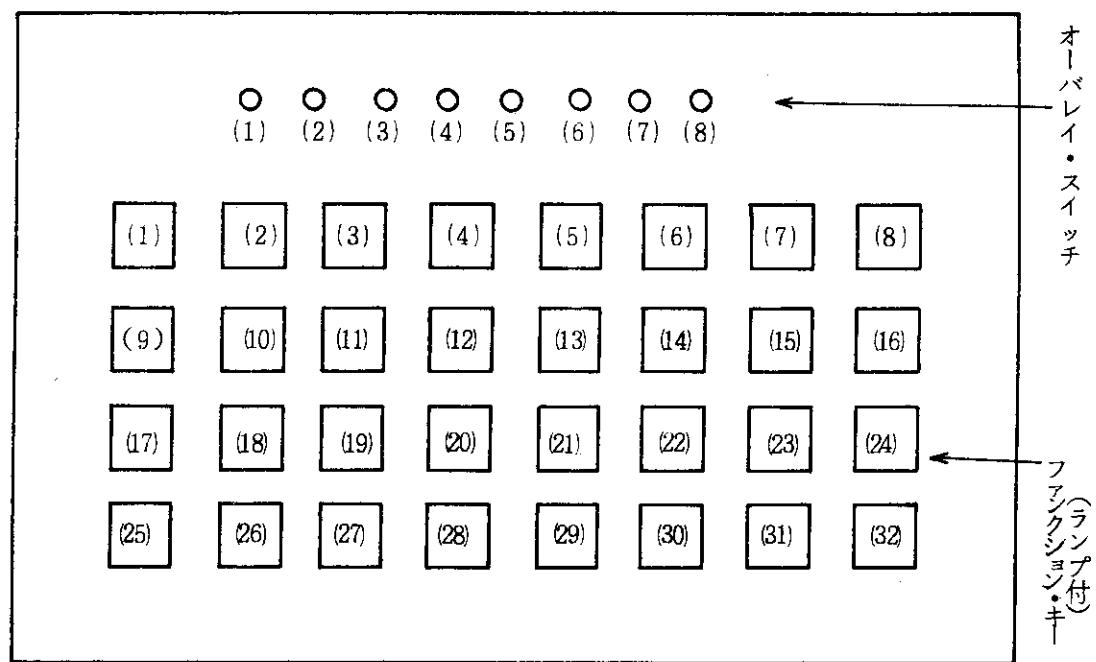


Fig. 2.4 ファンクション・キーボード

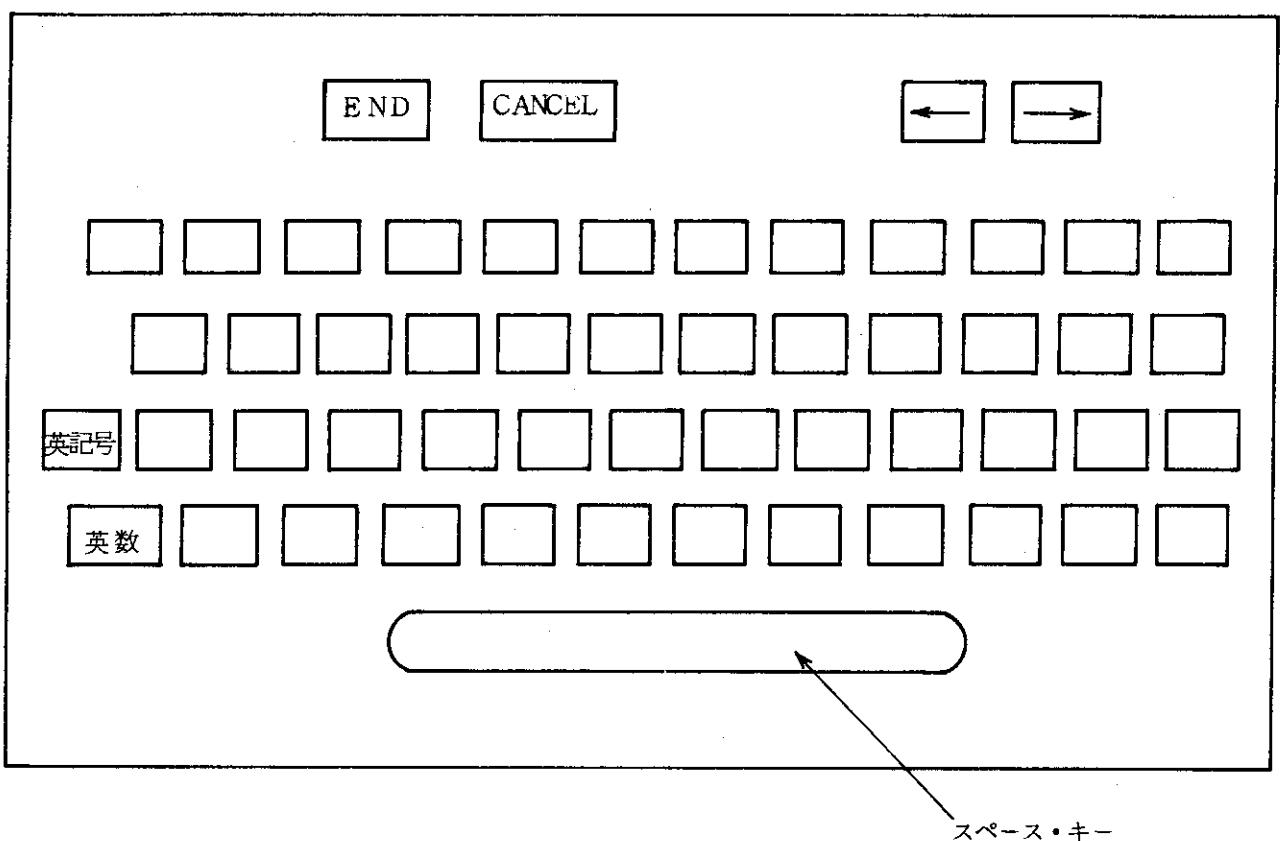


Fig. 2.5 文字キーボード

### 3 グラフィック・プログラミング

#### 3.1 PGSLIBの概要

グラフィック・ディスプレイを使用するためのプログラムは、グラフィック・プログラミング用ソフトウェアとしてのPGSLIB<sup>(10)</sup>を用いて作成することができる。このPGSLIBは約150個のグラフィック処理命令から成り、これらの命令はPGS（Plotter-compatible Graphic Subroutine Package），簡易GSP（Simplified GSP）と汎用GSP（General GSP）の3つのパッケージに大別される。

##### (1) PGS

このパッケージはCALCOMPプロッタ<sup>(12), (13)</sup>またはCOM<sup>(14)</sup>のプログラミングと同じ方法でCRT面への図形表示のプログラミングを可能にする。また、CRT面上の図形を消去したり、CRT面上の図形のハード・コピーをCOM、またはプロッタに出力できる。さらに、CRT面上の図形に対してライトペンと文字キーボードの使用による、日付、利用者名、タイトルおよび説明文などのコメント入力ができる。

##### (2) 簡易GSP

このパッケージはPGSに合わせてFACOM GSP<sup>(2)</sup>を単純化したものである。このため図形の表示や消去の処理の他に、ライトペン、ファンクション・キーボードおよび文字キーボードなどの使用による人間と計算機との対話処理に関する簡単なプログラミングを可能にする。

##### (3) 汎用GSP

このパッケージは、グラフ表示<sup>(15)</sup>、ライトペン・スムーズィング、二次元図形表示<sup>(16)</sup>、三次元図形表示<sup>(17)</sup>、テキスト処理<sup>(18)</sup>、および対話型計算処理<sup>(19)</sup>などに関する、おもにPGSと簡易GSPの命令によって書かれた、各種の汎用グラフィック処理命令からなる。したがってこのパッケージの使用によって、利用者はPGSと簡易GSPだけによるよりももっと簡単にグラフィック・プログラミングができる。

上記のPGS、簡易GSPおよび汎用GSPの各パッケージの中でグラフィック・プログラミングの面からみて基本的でかつよく使用される命令の集まりはPGSLIB/BASIC<sup>(19)</sup>と呼ばれている。通常、グラフィック・プログラムの作成はこのPGSLIB/BASICに属する命令の使用で間に合う。PGSLIBの他の命令<sup>(10)</sup>は必要に応じ選択して使用すればよい。

#### 3.2 PGSによる図形表示のプログラミング

##### 3.2.1 PGSの命令

PGSLIB/BASICに属するPGSの命令は、Table 3.1に示すようにPGS命令、接続命令、および拡張命令の3つに大別される。

##### (1) PGS命令

この命令群はプロッタまたはCOMの基本サブルーチン

PLOTS, PLOT, SYMBOL, NUMBER, SCALE, AXIS, LINE, FACTOR,  
WHERE, NEWPEN

と同じ機能をグラフィック・ディスプレイで実現する。PGS命令が実行されると、この命令のアーギュメントとして与えられた図形情報はグラフィック・ディスプレイの命令語の並びとしてのディスプレイ語列に変換され、そしてディスプレイ語列はコア上の出力領域GDOA<sup>(2)</sup> (Graphic Data Output Area) に蓄積される。GDOAの内容は後述される拡張命令の表示機能によってグラフィック・ディスプレイのバッファ・メモリに転送され、CRT面上に図形が表示される。

### (2) 接続命令

この命令群は、上記の基本サブルーチンと同じ名前をもち、自分に対応するPGS命令を呼出すだけの働きをする。たとえばPLOTS命令はGPLOTS命令を呼出す。

### (3) 拡張命令

この命令は、プロッタやCOMにはないグラフィック・ディスプレイ特有の機能を実現させるもので、

CALL GPLOT (xcoor, ycoor, ibeam)

のibeamの値によって区別されるつぎのような機能がある。

1) ibeam = 3 3 3

GDOAの初期化(消去)，またはCRT面上に表示される図形単位に対する名前付け，

2) ibeam = 4 4 4

GDOAのディスプレイ・バッファへの転送による図形の表示，およびGDOAの初期化，

3) ibeam = 6 6 6

CRT面上に表示されている全図形，または図形単位の消去，

4) ibeam = 7 7 7

GDOAのディスプレイ・バッファへの転送による図形の表示，文字キーボードからのコメント入力，およびGDOAの初期化，

5) ibeam = 8 8 8

CRT面上に表示されている図形のCOM，またはプロッタへのハード・コピー出力。

Table 3.1 PGSLIB/BASICに属するPGSの命令

命 令	機 能
GPLOTS	グラフィック・ディスプレイをオープンし，使用可能状態にする。
GPLOT	CRT面上でのビームによるベクトル表示やビームの位置決めのためのディスプレイ語列を作成し，これをGDOAに格納する。またはグラフィック・ディスプレイをクローズし，使用不能状態にする。
GSYMBL	文字列(英数字，特殊記号の一文字，またはそれらの並び)，またはプロッティング・ポイント用記号(Centered Symbol)を表示するためのディスプレイ語列を作成し，これをGDOAに格納する。
GNUMBR	実数(浮動小数点)を10進数(固定小数点)形式の文字列で表示するためのディスプレイ語列を作成し，これをGDOAに格納する。
GSCALE	データのプロッティングに先立ち，データのグラフがCRT面内におさまるようにスケーリングし，データの最小値と増分を求める。

命 令	機 能
GAXIS	線形の座標軸を表示するためのディスプレイ語列を作成し、これをGDOAに格納する。
GLINE	2つの配列に格納されているデータを順に結んでプロットするディスプレイ語列を作成し、これをGDOAに格納する。
GFACTR	図形の拡大・縮少に関する尺度因子を与える。
GWHERE	ビームの現在位置の座標と尺度因子を得る。
GNEWPN	現在は何もしない。
接続命令	上記の各PGS命令に対応して PLOTS, PLOT, SYMBOL, NUMBER, SCALE, AXIS, LINE, FACTOR, WHERE, NEWPEN の命令があり、これらの命令はそれぞれ対応するPGS命令を呼出す働きをする。
拡張命令 (GPLOT)	GDOAの初期化または図形単位の名前付け、図形の表示とGDOAの初期化、全図形または図形単位の消去、図形の表示とコメント入力とGDOAの初期化、および図形のCOMまたはプロッタへのハード・コピー出力を行う。

### 3.2.2 PGSボタン

一般に、CRT面上に表示された名前付きの図形単位について、それがライトペンによって認識されたとき、その図形単位に対応するプログラムによってある定まった働きをする図形単位をライト・ボタンと呼ぶ。

PGSでは、拡張命令によってコメント入力やハード・コピー出力を行うために、それぞれコメント入力用、ハード・コピー用のライト・ボタンがある。コメント入力用ボタンは

KEY, END, VLG, LAG, SML, VSM

の6個のボタンからなり、ハード・コピー用ボタンは

HARDCOPY, NOCOPY

の2個のボタンからなる。これらのボタンは総称してPGSボタンと呼ばれている。

PGSボタンはグラフィック・ディスプレイのオープン直後にFig. 3.1に示すようにCRT面上に表示される。Fig. 3.1でCRT面中央のメッセージは特にPGSメッセージと呼ばれ、利用者の図形が表示されるまでの間ウインクしながら表示し続ける。これに対してPGSボタンは利用者のプログラムが実行されている間中ほとんど表示されている。これらPGSボタンの操作、およびコメント入力やハード・コピーの機能と操作については後で詳しく述べる。

### 3.2.3 座標系と文字

PGSの座標系は、プロッタやCOMとの互換を考慮してFig. 3.2のようになっている。座標の値や長さの単位もプロッタやCOMと同じくmmである。

CRT面上に表示される文字<sup>(2)</sup>については、すでにTable 2.1に示したように4種類の大きさの文字しか描かれず、しかも角度0度のみである。文字の大きさはVSM, SML, LAG, VLGの指定か、またはmm単位での大きさの指定が可能である。mm単位で指定された場合は、その大きさによって上記4種類のいずれかにセットされる。すなわち、指定された文字の高さhによって

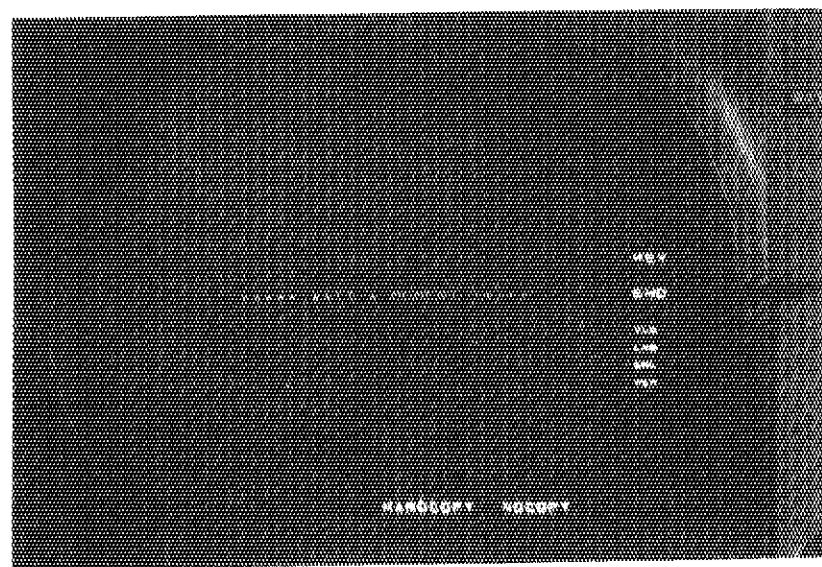
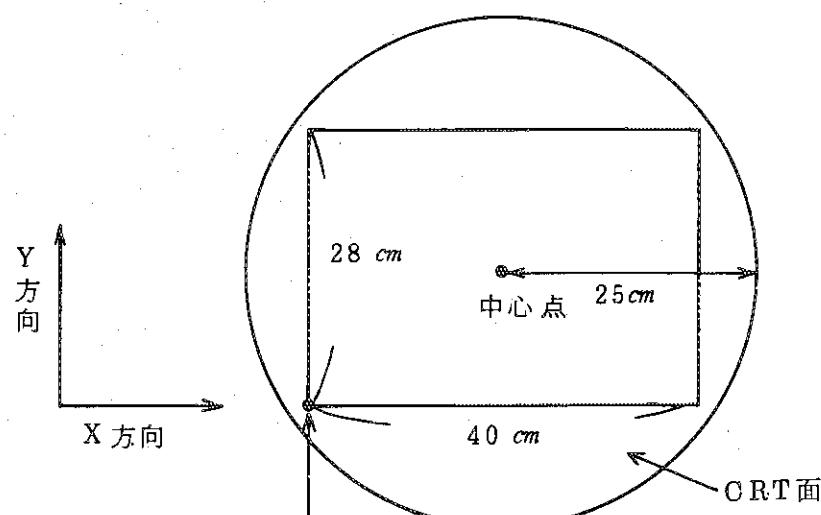


Fig. 3.1 PGS ボタンとメッセージ



PGS の最初の原点

Fig. 3.2 グラフィック・ディスプレイの座標系  
(PGSLIB の場合)

$0 \text{ mm} \leq h < 4.2 \text{ mm}$  ならば VSM,  
 $4.2 \text{ mm} \leq h < 6.5 \text{ mm}$  ならば SML,  
 $6.5 \text{ mm} \leq h < 9.3 \text{ mm}$  ならば LAG,  
 $9.3 \text{ mm} \leq h$  ならば VLG

が使用される。

### 3.3 簡易 GSPによる対話処理のプログラミング

#### 3.3.1 簡易 GSPの命令

PGSLIB/BASICに属する簡易 GSPの命令は、Table 3.2 に示すように割込み処理命令、トラッキング処理命令、およびデータ入力処理命令の3つに大別される。

##### (1) 割込み処理命令

この命令群はライトペンやファンクション・キーボードなどによって計算機に指示を与える場合に使用される。CRT面に表示された図形(名前付き)をライトペンでピックすると、計算機に対して割込みが起り、その図形単位名が割込みの識別記号として割込み処理に関する待行列に格納される(Table 3.3)。格納された識別記号はGET IDF命令によって読み出すことができる。さらにGLIST命令とGJUMP命令の対使用によって、読み出した識別記号を予め指定した識別記号のリストとチェックし、一致したならばその識別記号に対応した文番号にプログラムを分岐することができる。

##### (2) トラッキング処理命令

この命令群はCRT面上の位置座標をライトペンとトラッキング・シンボルによって計算機に入力する場合に使用される。トラッキング・シンボルは十字形をしたシンボルであるが、これはGBGTRK命令によってCRT面上に表示される。表示されたトラッキング・シンボルはライトペン・スイッチがオンの間ライトペンの動きに追従する。トラッキング・シンボルが移動するたびに計算機に対して割込みが起り、トラッキング・シンボルの中心座標はGRDTRK命令で読むことができる。トラッキング・シンボルはGENTRK命令によってCRT面から消去される。

##### (3) データ入出力処理命令

この命令群は整数型、実数型、または文字型のデータを文字キーボードによって計算機に入力したり、逆に計算機のデータを整数型、実数型、または文字型でCRT面上に表示する場合に使用される。データの入力はGREADD命令によって、またデータの出力表示はGWRITE命令によって行うことができる。入力や出力のフォーマットはFORTRANのFORMAT文で与えられる。

Table 3.2 PGSLIB/BASICに属する簡易GSPの命令

命 令		機 能
割 込 み 処 理 命 令	GETIDF	割込み処理に関する待行列に格納された識別記号を発生した順に取出す。
	GLIST	GJUMP命令で指定される文番号に対応する識別記号をリストで指定する。
	GJUMP	この命令で引用した識別記号がGLIST命令で指定された識別記号のどれかと一致すると、その識別記号に対応する、この命令で指定される文番号に分岐する。
ト 処 理 キ 命 令 グ 令	GBGTRK	CRT面上の指定された座標位置にトラッキング・シンボルを表示する。
	GRDTRK	この命令が実行された時点でのトラッキング・シンボルの位置座標を取出す。
	GENTRK	CRT面上からトラッキング・シンボルを消去する。
テ 処 理 入 命 出 力 令	GREADD	CRT面上の指定された座標位置にカーソルを表示し、文字キーボードからキー・インされた文字列を指定されたフォーマットに従って整数型、実数型または文字型の定数に変換する。
	GWRITE	整数型、実数型または文字型のデータを指定されたフォーマットに従う文字列で表示するためのディスプレイ語列を作成し、これをGDOAに格納する。

Table 3.3 割込みの種類と識別記号

割 込 み の 種 類	識 別 記 号	
	1 語	2 語
ファンクション・キー番号 1	0	1
ファンクション・キー番号 2	0	2
ファンクション・キー番号 3 2	0	3 2
ライトペンにトラッキング・シンボルが追従しているとき、 ライトペンのスイッチをオフにしたときに起る割込み	0	3 4
ライトペンにトラッキング・シンボルが追従しているとき に起る割込み（ライトペンのスイッチがオン）	0	3 5
図形単位名が (name 1, name 2) の図形をライトペンで ピックしたときに起る割込み	name 1 (≠ 0)	name 2

### 3.4 汎用GSPによるグラフィック・プログラミング

#### 3.4.1 汎用GSPの命令

汎用GSPの命令は、つぎに述べるように二次元図形表示命令、三次元図形表示命令、汎用グラフ表示命令、テキスト処理命令、対話型計算処理命令、プロッタ・タイプ・シンボル命令、およびライトペン・スムーズィング命令の7つに大別される。

## (1) 二次元図形表示命令

この命令群は二次元の基本的な図形や種々のグラフを表示するための各種の命令からなる。各命令はプロッタ用、またはCOM用のファンクショナル・サブルーチン<sup>(13)</sup>がPGSによってグラフィック化されたものであり、したがって機能的にプロッタ用、またはCOM用のものとほとんど同じである。

## (2) 三次元図形表示命令

この命令群は三次元配列として与えられた三次元図形に関するデータを透視図として表示するための各種の命令からなる。各命令はプロッタ用、またはCOM用のTHREEDサブルーチン<sup>(14)</sup>がPGSによってグラフィック化されたものであり、したがって機能的にプロッタ用、またはCOM用のものとほとんど同じである。

## (3) 汎用グラフ表示命令

この命令群は与えられたデータを自動スケールし、線形または対数の座標軸と共にグラフを表示するための数個の命令からなる。特にGPLOT 1と呼ばれる命令<sup>(15)</sup>は、プロッタ用、またはCOM用のGPLOT 1がPGSによってグラフィック化されたものであり、グラフ表示の進行がPGSボタンによって制御される点を除けば、プロッタ用、またはCOM用のものとほとんど同じである。

## (4) テキスト処理命令

この命令群は文字データの入出力処理に関してグラフィック・ディスプレイをキャラクタ・ディスプレイ的に処理しやすくするための各種の命令<sup>(16)</sup>からなる。したがって、多くの文字データに関する編集処理を対話形式で行う場合のプログラミングに有効である。

## (5) 対話型計算処理命令

この命令群はグラフ表示の動的な変更、データ・テーブルの表示、データ・テーブルでのデータの追加・修正・削除の処理、ライト・ボタンがピックされたときのプランチ処理などを行うための各種の命令<sup>(17)</sup>からなる。したがって、対話形式で数値計算やデータ処理を行う場合のグラフィック・プログラミングに有効である。

## (6) プロッタ・タイプ・シンボル命令

この命令はプロッタ、またはCOMと同じように任意の大きさ、任意の角度でシンボルを表示するための命令である。これはSYMB 4と呼ばれ、呼出し形式はGSYMBL命令（またはSYMBOL命令）と同じである。SYMB 4命令は英数字や特殊記号をソフトウェアで作成し表示する。

## (7) ライトペン・スムーズィング命令

この命令は配列で与えられたデータをライトペンでスムーズィング（平滑化）するための命令で、GSMOOTと呼ばれる。

上に述べたように汎用GSPの命令は多岐にわたっており、命令の数もかなりに達する。ここでは PGSLIB/BASICに属するGPLOT 1命令とGSMOOT命令についてTable 3.4 にそれらの機能の概要を示す。

Table 3.4 PGSLIB/BASICに属する汎用GSPの命令

命 令	機 能
G PLOT 1	自動スケーリング（リニア・リニア、リニア・ログ、ログ・リニア、ログ・ログ），同一スケールでの多重プロッティング，および多様なグラフ表示（直線プロット，点線プロット，シンボル・マーク・プロット）を行う。
G SMOOTH	2つの配列に格納されているデータについて，ライトペンによるスムージングを可能にさせ，その結果を配列に格納する。

### 3.5 グラフィック・プログラムの例

これまで述べてきたPGSLIBの命令を用いてグラフィック・プログラムを作成した具体例を，プロッタやCOMプログラムとの対比で示す。

今，二次関数

$$Y(I) = A * (X(I) - B) ** 2 + C \quad (I=1, 2, \dots, 32)$$

を計算し，そのグラフを描くプログラムを考える。

ここで，A，B，Cは1枚の入力カードから与えられるパラメータで，X(I) (I=1, 2, \dots, 32)は

$$X(1) = -15.0, X(2) = -14.0, \dots, X(31) = 15.0, X(32) = 16.0$$

のように1.0きざみで-15.0から16.0まで変化する変数とする。また，プログラムはA=0が入力された時点で終了するものとする。

さて，上記の問題に対して，つぎの4通りの方法で作成したプログラムを，それぞれTable 3.5, Table 3.6, Table 3.7, Table 3.8に示す。

#### (1) プロッタ・プログラム

1ケース，1ページとしてプロッタにグラフ出力する。ここで，1ページはプロッタ用紙上で巾30cmとする。

#### (2) COMプログラム

1ケース，1駒としてCOMにグラフ出力する。

#### (3) グラフィック・プログラム（その1）

1ケース，1画面としてCRT面上にグラフ表示する。

#### (4) グラフィック・プログラム（その2）

(3)と同じ方法でグラフ表示後，つぎのような3つの処理が選択して行えるものとする。

(a) ライトペンによる次のケースへの移行，

(b) ライトペンによる再計算の実行とその結果のグラフ表示

(c) ライトペンと文字キーボードによるパラメータの変更。

ここで，(b)，(c)についてはグラフ表示の結果に満足できるまで繰返し操作可能とする。

Table 3.5 プロッタ・プログラム

FACOM 230-75 (M7) FORTRAN-D -750715- V06-L05 76.06.03 PAGE 1

\* SOURCE STATEMENT \*

```

C
C * PLOTTER EXAMPLE *
C
1  DIMENSION X(34), Y(34),BUFF(1024)
2  CALL PLOTS ( BUFF, 1024)
3  X(1)=-15.
4  DO 10 I=2,32
5  10 X(I)=X(I-1)+1.
6  20 READ (5,21) A, B, C
7  21 FORMAT (3F10.3)
8  IF ( A.EQ.0 ) GO TO 900
9  25 DO 30 I=1,32
10 30 Y(I)= A*(X(I)-B)**2+C
11  CALL SCALE ( X, 200., 32, 1, 10. )
12  CALL SCALE ( Y, 200., 32, 1, 10. )
13  CALL AXIS ( 0., 0., 'X - V A L U E',-13,200.,0.,X(33),X(34),10. )
14  CALL AXIS ( 100.,0.,'Y - V A L U E',13,200.,90.,Y(33),Y(34),10. )
15  CALL LINE ( X, Y, 32, 1, 0, 0 )
16  CALL SYMBOL ( 60., 230., 8., 'Y=A*(X-B)**2+C',0.,14 )
17  CALL PLOT ( 300., 0., -3 )
18  GO TO 20
19 900 CALL PLOT ( 0., 0., 999 )
20  STOP
21  END

```

Table 3.6 COMプログラム

FACOM 230-75 (M7) FORTRAN-D -750715- V06-L05 76.06.03 PAGE 2

\* SOURCE STATEMENT \*

```

C
C * GRAPHIC COM EXAMPLE *
C
1  DIMENSION X(34), Y(34),BUFF(1024)
2  CALL PLOTS ( BUFF, 1024)
3  X(1)=-15.
4  DO 10 I=2,32
5  10 X(I)=X(I-1)+1.
6  20 READ (5,21) A, B, C
7  21 FORMAT (3F10.3)
8  IF ( A.EQ.0 ) GO TO 900
9  25 DO 30 I=1,32
10 30 Y(I)= A*(X(I)-B)**2+C
11  CALL SCALE ( X, 200., 32, 1, 10. )
12  CALL SCALE ( Y, 200., 32, 1, 10. )
13  CALL AXIS ( 0., 0., 'X - V A L U E',-13,200.,0.,X(33),X(34),10. )
14  CALL AXIS ( 100.,0.,'Y - V A L U E',13,200.,90.,Y(33),Y(34),10. )
15  CALL LINE ( X, Y, 32, 1, 0, 0 )
16  CALL SYMBOL ( 60., 230., 8., 'Y=A*(X-B)**2+C',0.,14 )
17  CALL PLOT ( 300., 0., -3 )
18  CALL PLOT ( 0., 0., 666 )
19  GO TO 20
20 900 CALL PLOT ( 0., 0., 999 )
21  STOP
22  END

```

Table 3.7 グラフィック・プログラム(その1)

FACOM 230-75 (M7) FORTRAN-D -750715- V06-L05 76.06.03 PAGE 3

\* SOURCE STATEMENT \*

```

C
C * PGS EXAMPLE *
C
1  DIMENSION X(34), Y(34),BUFF(1024)
2  CALL PLOTS ( BUFF, 1024)
3  X(1)=-15.
4  DO 10 I=2,32
5  10 X(I)=X(I-1)+1.
6  20 READ (5,21) A, B, C
7  21 FORMAT (3F10.3)
8  IF ( A.EQ.0 ) GO TO 900
9  25 DO 30 I=1,32
10 30 Y(I)= A*(X(I)-B)**2+C
11  CALL SCALF ( X, 200., 32, 1, 10. )
12  CALL SCALE ( Y, 200., 32, 1, 10. )
13  CALL AXIS ( 0., 0., 'X - V A L U E',-13,200.,0.,X(33),X(34),10. )
14  CALL AXIS ( 100.,0.,'Y - V A L U E',13,200.,90.,Y(33),Y(34),10. )
15  CALL LINE ( X, Y, 32, 1, 0, 0 )
16  CALL SYMBOL ( 60., 230., 8., 'Y=A*(X-B)**2+C',0.,14 )
17  CALL PLOT ( 300., 0., -3 )
18  CALL PLOT ( 0., 0., 777 )
19  CALL PLOT ( 0., 0., 888 )
20  CALL PLOT ( 0., 0., 666 )
21  GO TO 20
22 900 CALL PLOT ( 0., 0., 999 )
23  STOP
23  END

```

Table 3.8 グラフィック・プログラム(その2)

FACOM 230-75 (M7) FORTRAN-D -750715- V06-L05 76.06.03 PAGE 4  
 \* SOURCE STATEMENT \*

```

C
C * SIMPLE GSP EXAMPLE *
C
1   DIMENSION X(34), Y(34),BUFF(1024)
2   CALL PLOTS ( BUFF, 1024)
3   X(1)=15.
4   DO 10 I=2,32
5   10 X(I)=X(I-1)+1.
6   20 READ (5,21) A, B, C
7   21 FORMAT (3F10.3)
8   IF ( A.EQ.0 ) GO TO 900
9   25 DO 30 I=1,32
10  30 Y(I)= A*(X(I)-B)**2+C
11  CALL SCALE ( X, 200., 32, 1, 10. )
12  CALL SCALF ( Y, 200., 32, 1, 10. )
13  CALL AXIS ( 0., 0., X - V A L U E !,-13,200.,0.,X(33),X(34),10. )
14  CALL AXIS ( 00.,0.,Y - V A L U E !,13,200.,90.,Y(33),Y(34),10. )
15  CALL LINE ( X, Y, 32, 1, 0, 0 )
16  CALL SYMBOL ( 60., 230., 8., 'Y=A*(X-B)**2+C',0.,14 )
17  CALL PLOT ( 300., 0., -3 )
18  CALL PLOT ( 0., 0., 777 )
19  CALL PLOT ( 0., 0., 888 )
20  CALL SYMBOL ( 220., 180., 6.5, 'READ A NEXT CARD', 0., 16 )
21  CALL PLOT ( 'EL', 'RN', 333 )
22  CALL SYMBOL ( 220., 165., 6.5, 'CALCULATE AGAIN!', 0., 15 )
23  CALL PLOT ( 'EL', 'CH', 333 )
24  CALL SYMBOL ( 220., 150., 6.5, 'CHANGE PARAM. FMT=F10.3', 0., 23 )
25  CALL PLOT ( 'EL', 1, 333 )
26  CALL GWRITE ( 230., 135., 'LAG', A, 0., #51 )
27  51 FORMAT ( 'A',F10.3 )
28  CALL PLOT ( 'EL', 2, 333 )
29  CALL GWRITE ( 230., 120., 'LAG', B, 0., #52 )
30  52 FORMAT ( 'B',F10.3 )
31  CALL PLOT ( 'EL', 3, 333 )
32  CALL GWRITE ( 230., 105., 'LAG', C, 0., #53 )
33  53 FORMAT ( 'C',F10.3 )
34  CALL PLOT ( 0., 0., 444 )
C
35  60 CALL GETIDF ( I1, I2 )
36  CALL GJUMP ( #61, [I,I2, #70, #80, #100 ]
37  61 CALL GLIST ( 'EL', 'RN', 'EL', 'CA', 'EL', 'CH' )
38  GO TO 60
39  70 CALL PLOT ( 0., 0., 666 )
40  GO TO 20
41  80 CALL PLOT ( 0., 0., 666 )
42  GO TO 25
C
43  100 CALL GETIDF ( I1, I2 )
44  CALL GJUMP ( #101, I1, I2, #110, #120, #130 )
45  101 CALL GLIST ( 'EL', 1, 'EL', 2, 'EL', 3 )
46  GO TO 100
47  110 CALL GREADD ( 243., 128., 'LAG', A, 0., #111 )
48  GO TO 150
49  111 FORMAT (F10.3)
50  120 CALL GREADD ( 243., 113., 'LAG', B, 0., #111 )
51  GO TO 150
52  130 CALL GREADD ( 243., 98., 'LAG', C, 0., #111 )
53  150 DO 160 I=1,3

```

FACOM 230-75 (M7) FORTRAN-D -750715- V06-L05 76.06.03 PAGE 5  
 \* SOURCE STATEMENT (FTMAIN )\*

```

54  160 CALL PLOT ( 'EL', I, 666 )
55  GO TO 50
56  900 CALL PLOT ( 0., 0., 999 )
57  STOP
58  END

```

## 4 グラフィック・プログラムの実行

前章までに述べたグラフィック・プログラミングの方法によって利用者はグラフィック・プログラムをFORTRANで作成可能となった。

ここでは、作成されたグラフィック・プログラムをFACOM 230-75電子計算機とFACOM 6233Aグラフィック・ディスプレイ装置の下で実行するときの具体的な操作方法について述べる。

### 4.1 グラフィック用ジョブ制御文

グラフィック・プログラムを実行するためには実行に先立ちジョブ制御文の作成が必要である。

FACOM 230-75ではグラフィック・プログラムは、プロッタやCOMプログラムの場合と同じようにFORTRAN-C/D、またはFORTRAN-Hで実行可能である。まず、FORTRAN-C/Dの場合の代表的なジョブ制御文の構成例をFig. 4.1に示す。

```

col. 72
↓
¥NO ××××,
          C. n1 /   } (1)
          T. n2 /   }
          W. n3 /   }
          P. n4 /   }
          , /   }
          G D P /   } (2)
          C 3 5 /   }

¥GJOB ×××.....
¥DFORT
    グラフィック・プログラム (FORTRAN)      (3)
    ¥DLIEDRUN GRFD=ON, PGSLIB=CALL, GDSP=ON, GDBG=ON (4)
    ¥GCOM 3 5
    ¥DISK F 0 8
    ¥DATA
    入力データ
    ¥JEND

```

Fig. 4.1 グラフィック用ジョブ制御文の構成

Fig. 4.1で、(1)は一般ジョブと同じように使用するシステム資源の範囲を示すNO文の追加情報1であり、グラフィック・ジョブでは主記憶量、CPU時間、ライン・プリンタ出力量、カード出力量は、それぞれ256K語、30分、240ページ、0枚以下でなければならない。

(2)はNO文の追加情報2で、GDPはグラフィック・ジョブ・クラスの指定、C35はCRT面のハード・コピーをCOMに出力するときのCOM使用の指定である。グラフィック・ジョブ・クラスでは、通常1ジョブだけは他の一般ジョブ・クラスのジョブに比べ優先的な扱いを受ける。CRT面のハード・コピーをプロッタに出力するときは、C35の代りにPLTでプロッタ使用を指定する。

(3)は、FORTRANでPGSLIBの命令を用いて書かれたグラフィック・プログラムである。

(4)は、グラフィック・プログラムの結合編集および実行に関するジョブ制御文である。GRFD=ONは、結合編集時にグラフィック・サブルーチン・ライブラリとしてのPGSLIBをグラフィック・プログラムに組み込むためにそのFD文を展開することを意味する。PGSLIB=CALLは、PGSLIBを他のライブラリに優先して自動コールすることを意味する。GDSP=ONは実行時にグラフィック・ディスプレイを使用することを意味し、GDBG=ONはグラフィック・プログラムをデバッグモードで実行することを意味する。デバッグモードではプログラムのエラー・メッセージはCRT面上に表示され、かつプリンタにも出力される。GDBG=ONを省略するとエラー・メッセージは全然出力されない。

(5)は、実行段階でCRT面に表示された図形をCOMにハード・コピーするときに指定するジョブ制御文である。ハード・コピーをプロッタに出力するときはFig. 4.1で

**¥ GCOM 35**

の代りに

**¥ PLOT**

を用いる。ハード・コピーをCOMにもプロッタにも全然出力しないときは、これらのジョブ制御文はそう入する必要がない。また、このときはNO文の追加情報2としてのC35やPLTの指定も不要である。

Fig. 4.1について上で説明されなかったその他のジョブ制御文などについては、一般ジョブの場合と同じである。

つぎに、FORTRAN-Hでグラフィック・プログラムを実行する場合は、Fig. 4.1で

**¥ DFORT**

の代りに

**¥ HFORT MODE = CDEX**

また、

**¥ DLIEDRUN GRFD=ON, PGSLIB=CALL, GDSP=ON, GDBG=ON**

の代りに

¥ HLIEDRUN GLIED=LIED, GRFD=ON, PGSLIB=CALL,  
GDS P=ON, GDBG=ON

を用いればよい。現在 FORTRAN-Hによるグラフィック・プログラムの実行はCNPモード(正規化モード)で行われるが、近い内にCPモード(巨大モード)での実行が可能になる予定である。そのときは¥HFORTと¥HLIEDRUNで、それぞれMODE=CDEXとGLIED=LIEDを省略するだけでCPモードが指定されたことになる。ただし、利用者のプログラムの中で、DFILE, DREAD, DWRITE文や¥n(nは文番号)などが使用されているときは、¥HFORTでMODF=(CP, CDEX)と指定するか、または簡単にB=CDEXと指定する必要がある。

つきのTable 4.1は、3.5のグラフィック・プログラム(その2)をFORTRAN-C/Dで実行したときのジョブ制御文のリストである(ハード・コピーはCOM出力を指定)。

## 4.2 グラフィック・ディスプレイの使用

### 4.2.1 セミ・オープン室

グラフィック・ディスプレイは現在セミ・オープン室に設置され、通常はFACOM 230-75のAシステムに接続されている。このグラフィック・ディスプレイを使用するグラフィック・ジョブに関する操作はすべてセミ・オープン室で行うことができる。

グラフィック・ディスプレイを使用する場合、利用者は前もってFACOM 230-75ジョブの受付でグラフィック・ディスプレイの使用時間帯を申込んでおかなければならない。

### 4.2.2 グラフィック・ジョブの実行

グラフィック・ディスプレイの使用は、セミ・オープン室のカード・リーダにジョブ・デックを投入することから始まる。投入されたグラフィック・ジョブのFACOM 230-75による処理状況は、セミ・オープン室のコンソール・ディスプレイ上に表示される。

グラフィック・プログラムの翻訳、結合編集が終り、実行段階に入ると、PGSLIBによってコンソール・ディスプレイ上に2度続けて

\* \* \* GRAPHIC START \* \* \*

のメッセージが表示される。このメッセージが表示されたら利用者はグラフィック・ディスプレイの前にゆき、以後はプログラムの仕様にしたがってグラフィック・ディスプレイを操作すればよい。

グラフィック・プログラムが実行に入ったとき、グラフィック・ディスプレイのCRT面にはすぐにPGSボタンとウインクするメッセージが表示される(Fig. 3.1参照)。PGSボタンについては実行中ほとんど表示されているが、メッセージは利用者の図形が表示される直前にCRT面から消去される。

グラフィック・ジョブの実行が終ると、計算結果などはセミ・オープン室に設置されている2台のプリンタのどちらかに出力される。

### 4.2.3 コメント入力

CRT面上に図形が表示された段階で利用者はライトペンと文字キーボードを用いて日付、利用者

名、タイトル、および説明文などを図形に対して表示することができる。このコメント入力の機能と操作については第5章で詳しく述べる。

#### 4.2.4 ハード・コピー

グラフィック・ディスプレイに表示された図形のハード・コピーはハード・コピー装置、COM、またはプロッタに出力できる。ハード・コピー装置の場合はプログラムに関係なくいつでもPRINTキーを押すことによってハード・コピーできる。つきのFig. 4.2の図はハード・コピー装置によって出力されたハード・コピーである。COM、またはプロッタへのハード・コピー出力の機能と操作については第6章で詳しく述べる。

#### 4.2.5 ロール・イン／ロール・アウト

FACOM 230-75でのジョブ処理状況が混雑しているとき、グラフィック・プログラムが待ち状態になっているときは、プログラムが主記憶からロール・アウトされる場合がある。<sup>(20)</sup>このときは、その旨を利用者に知らせるための下記のメッセージがCRT面の左下に表示される。

GSP NO JOB WA ROLL OUT TYU

しかし、利用者がライトペンや文字キーボードを使い始めるとプログラムがすぐロール・インされるような仕組みになっている。すなわち、利用者が考えているときはロール・アウトされがちであるが、使い始めるとすぐロール・インされる。したがって主記憶の有効利用が可能となっている。

### 4.3 エラー・メッセージ

翻訳、結合編集の段階ではグラフィック・ジョブ特有のエラー・メッセージは出力されない。しかし、実行段階に入ると、一般ジョブでは出力されないようなグラフィック・ジョブ特有のエラー・メッセージが出力される場合がある。

グラフィック・プログラムをデバッキング・モードで実行すると、プログラムに関するエラー・メッセージはCRT面の左上につぎのように表示される。<sup>(20)</sup>

GS 0010 ee GSP サブルーチン名 KILL CONT

ここで、eeはエラー・コードであり、KILLがライトペンでピックされると、プログラムは強制終了となる。エラーを無視してプログラムを先に進めたい場合は、CONTをライトペンでピックすればよい。CRT面上に表示されたエラー・メッセージはプリンタにも出力される。したがって、実行終了後でもエラーの原因を探ることができる。エラーの程度が軽いものについてはCRT面上には表示されず、プリンタのみに出力されるものもある。

デバッキング・モードを指定しなかったときは、エラーが起きてもエラー・メッセージは出力されず、プログラムはエラーを無視して先へ進む。

つきのTable 4.2にグラフィック・プログラムで実行時に起りがちなエラーのエラー・コード一覧表<sup>(20)</sup>を示す。

Table 4.1 グラフィック用ジョブ制御文の例

```

FACOM 230-60/75      MONITOR6/7      SYSTEM=V04/L17      BATCH=V04/L01      JOB CONTROL LANGUAGE LIST 76.06.03 PAGE 1
.....1.....2.....3.....4.....5.....6.....7.....8

1 *NO 1803*
               /
W+0/PAGE 40
P+0/PCH 0
   /
GDP/
C35/
2 VM *GJOB    1411803,NAKAMURA,Y+341.02          /GRAPHICS
3 VM *DFORT
4 VM *DL1EDRUN GRFD=ON,PGSLIB=CALL,GDSP=ON,GDBG=ON
5 VM *GCOM35
6 VM *DISK F08
7 VM *DATA
8 *JEND

```

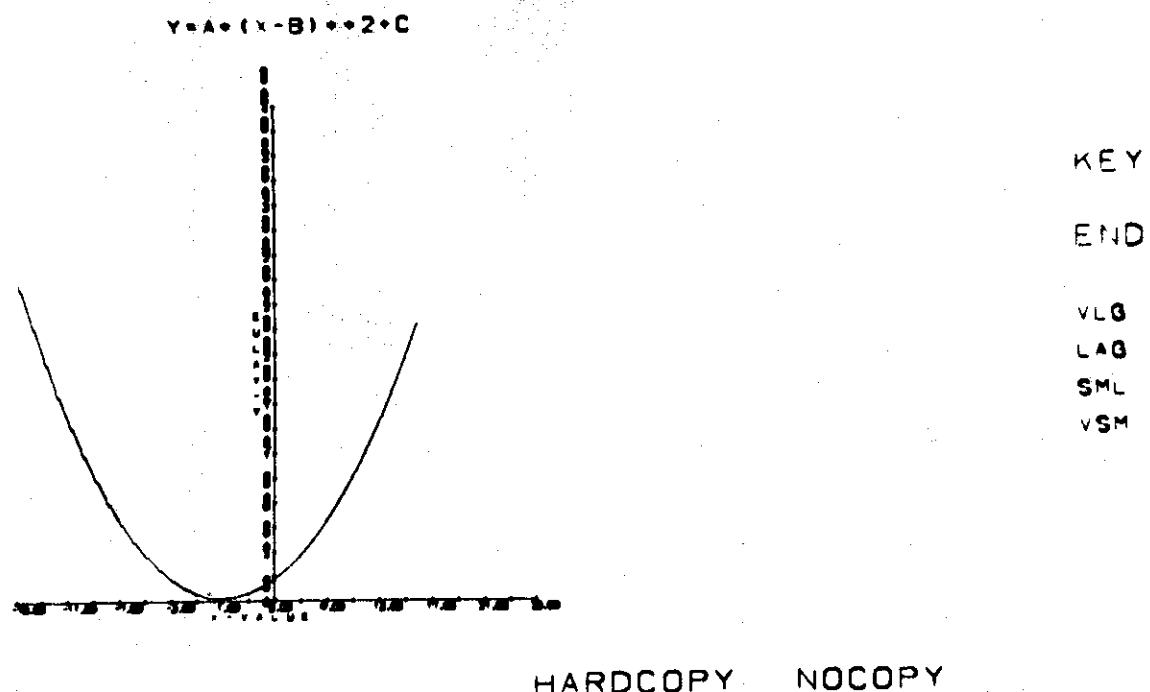


Fig. 4.2 ハード・コピー装置によって出力されたハード・コピー

Table 4.2 エラー・コード一覧表

エラー・コード	エラーの内容	命令
6	許されないFORMAT文が指定された。	GREADD, GWRITE
11	指定されたエレメント名について、第1エレメント名が0である。	拡張命令 (GPLOT)
15	GLISTのパラメータがない。	GJUMP
23	グラフィック・ディスプレイがオープンされていない。	拡張命令 (GPLOT)
25	ディスプレイ・バッファがオーバ・フローした。	拡張命令 (GPLOT)
26	トッキング・シンボルまたはカーソルがCRT面の外に出た。	GBGTRK, GREADD
27	キー・インされた文字が80個を越えた。	GREADD
29	許されない文字が指定された (INVALID CHARACTER)。	GREADD, GWRITE, 拡張命令 (GPLOT)
41	GDOAがオーバ・フローした。	GPLOT (拡張命令も含む), GSYMBL, GWRITE
43	指定されたエレメント名がない。	拡張命令 (GPLOT)
61	指定された座標値の絶対値が非常に大きいため、命令実行に伴う座標計算中に演算オーバ・フローが起きた。	GPLOT, GSYMBL, GREADD, GWRITE
63	GSPモニタ、またはGSP命令自体のシステム・エラー	

## 5 コメント入力

### 5.1 機能

すでに 3.2.1(3)の拡張命令で述べたように、利用者は

```
CALL GPLOT (0.0, 0.0, 777)
```

によって、CRT面上に図形が表示でき、しかも表示された図形に対して、日付、利用者名、タイトル、および説明文などのコメントをつけることができる。

このコメント入力は、ライトペンと文字キーボードを用いて、CRT面上の任意の位置に、何行でも行える。各行について入力・表示される最初の文字の位置は、ライトペンとトラッキング・シンボルによって指定される。コメントは1行当たり最大80文字まで入力・表示できる。文字の大きさはTable 2.1に示される4種類が可能である。

つきのFig. 5.1は、3.5のグラフィック・プログラム（その1）によってグラフが表示された段階で、コメント入力を用いて表示されたコメントと共にグラフをCOMにハード・コピーしたものである。

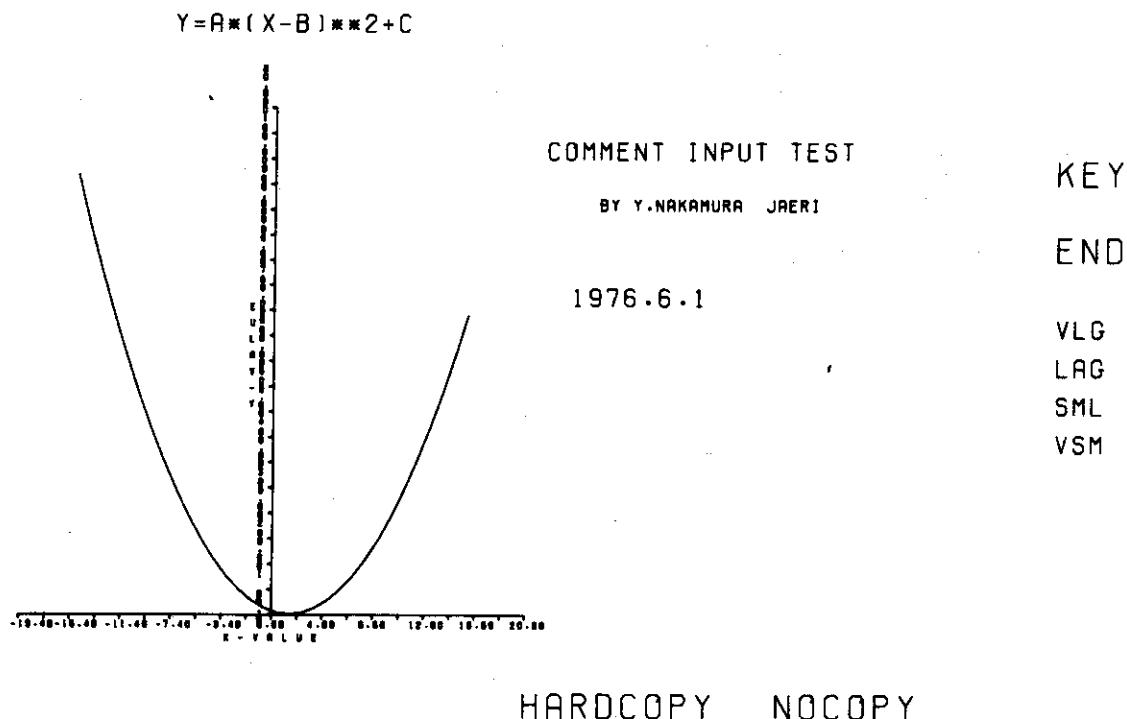


Fig. 5.1 コメント入力例のCOMへのハード・コピー  
(PGSボタン付き、わく無し)

## 5.2 操作

コメント入力はコメント入力用ボタン

KEY, END, VLG, LAG, SML, VSM

を用いて行われる。実際、これはつぎのような手順で行われる。

### (1) KEYをピックする

これは、文字キーボードによるコメント入力を開始するためにあり、ピッキングによってトラッキング・シンボルがCRT面の左上に表示される。このトラッキング・シンボルの位置は、それをライトペンに追従させることにより任意の位置に移動できる。移動後のトラッキング・シンボルの位置が、キー・インされた最初の文字が表示される出発点となる。

### (2) VLG, LAG, SML, VSMのどれかをピックする

出発位置が決ったら、上の4つのボタンの内1つを選び、文字の大きさを指定する。これらのボタンはTable 2.1に示されている文字の大きさに対応している。ボタンをピックするとカーソルが表示されるので、それを確認してからキー・イン開始する。

### (3) 文字キーボードによって文字を入力する

キー・インは1行につき最大80文字まで可能である。キー・インされた文字は左から右に向って表示されてゆき、それに伴いカーソルも移動する。キー・インの際、英数字の場合は [英数] キーを押してから、また特殊記号の場合は [英記号] キーを押してから、文字キーを押す。キー・インの途中で打ち誤りがあったときは、修正文字のところまで ← キーや → キーを使ってカーソルを移し、打ち直す。1行全体を修正する場合は [CANCEL] キーを使う。1行に対する入力が完了したら、 [END] キーを押すとカーソルが消える。

さらに別の行にキー・インしたい場合は、トラッキング・シンボルをライトペンに追従させ、つぎの出発点に移動させる。そして上の(2)から繰り返せばよい。

### (4) ENDをピックする。

キー・インが全部完了したらENDボタンをピックする。ピッキングによってトラッキング・シンボルはCRT面から消去される。キー・インを全然しない場合は、最初にENDをピックしてもよい。ENDがピックされるまで利用者のプログラムは待ち状態となるので注意されたい。

つぎのFig. 5.2は、コメント入力の操作の概要をフロー・チャートで示したものである。

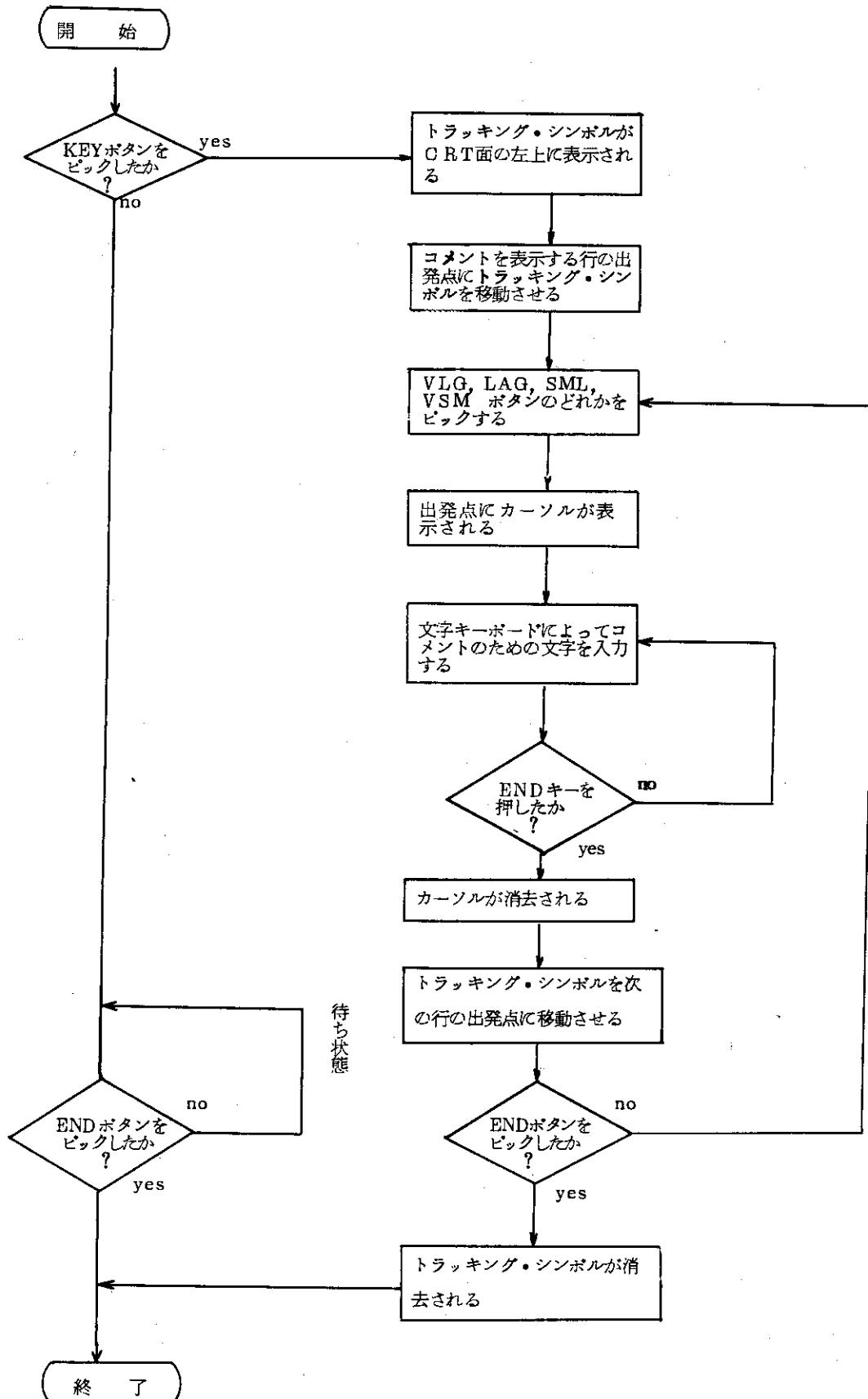


Fig. 5.2 コメント入力の操作に関するフロー・チャート

## 6 ハード・コピー

### 6.1 機能

すでに3.2.1(3)の拡張命令で述べたように、利用者は

```
CALL GPLT(0.0, 0.0, 888)
```

によって、それまでにCRT面上に表示されていた図形を、COM、またはプロッタにハード・コピーできる。

ハード・コピーはライトペン、またはファンクション・キーボードの指示により行われる。ライトペンのときはPGSボタンを除く全图形が、またファンクション・キーボードのときはPGSボタンも含めた全图形がハード・コピーされる。ハード・コピーはCOM、またはプロッタ上で、CRT面上に作成された图形の順序で描かれる。

実際、ハード・コピーを行うハード・コピー・ルーチンは、まずバッファ・メモリの内容をスクラッチ用の磁気ディスクに書き込む。つぎに、磁気ディスクの内容を1レコード単位で読み出してCOM、またはプロッタのコマンドに変換する。作成されたコマンドはシステム出力用の磁気ディスクに格納される。

このようにハード・コピー・ルーチンは、COM、またはプロッタのコマンドを磁気ディスクに出力するまでの働きをする。コマンドを磁気テープに出力し、磁気テープをCOM、またはプロッタにかけて图形を出力するのは利用者のプログラムとは別の時点で行われる。したがって、利用者がプログラムの実行段階でハード・コピーに要する時間はわずかである(FACOM230-75の場合、一回のハード・コピーに要する平均待ち時間は約10秒程度である)。

磁気ディスクに出力されたコマンドはシステム出力制御プログラムによって、複数ジョブ分のCOM、およびプロッタ・コマンド混在で磁気テープに出力される。この磁気テープの内容は振分けユーティリティによって、COM専用テープ、またはプロッタ専用テープに振分けられる。これらの専用テープを、それぞれCOM、またはプロッタにかけるとハード・コピーされた图形が出力される。出力された图形は後で利用者に配布される。

CRT面上の图形のハード・コピーは実物の1/2の大きさで描かれる。つぎのFig. 6.1はハード・コピーをCOMのフィルム上に出力したときの概略図である。Fig. 6.2はハード・コピーをプロッタに出力したときの概略図である。各图形は25cm(COMの場合は、10.1倍でプリントした場合)正方形のわく付き、または無しで描くことができる。また、ハード・コピーの图形の前に、利用者を区別するジョブ・ヘッダ用の駒(COMの場合)、またはタイトル(プロッタの場合)が描かれる。

つぎのFig. 6.3, Fig. 6.4は、ハード・コピーの実例である。Fig. 6.3はCOMのフィルムに出力されたものをプリントした例であり、Fig. 6.4はプロッタへの出力例である。

### 6.2 操作

ハード・コピーをCOMに出力するか、またはプロッタに出力するかの指定、およびハード・コピーにわくを付けて出力するか、またはわく無しで出力するかの指定は、利用者のプログラムの実行前にASW(Alteration Switch)によって行われる。このハード・コピーに関するASWのセッ

## 6 ハード・コピー

### 6.1 機能

すでに3.2.1(3)の拡張命令で述べたように、利用者は

```
CALL GPLOT(0.0, 0.0, 888)
```

によって、それまでにCRT面上に表示されていた図形を、COM、またはプロッタにハード・コピーできる。

ハード・コピーはライトペン、またはファンクション・キーボードの指示により行われる。ライトペンのときはPGSボタンを除く全図形が、またファンクション・キーボードのときはPGSボタンも含めた全図形がハード・コピーされる。ハード・コピーはCOM、またはプロッタ上で、CRT面上に作成された図形の順序で描かれる。

実際、ハード・コピーを行うハード・コピー・ルーチンは、まずバッファ・メモリの内容をスクランチ用の磁気ディスクに書き込む。つぎに、磁気ディスクの内容を1レコード単位で読み出してCOM、またはプロッタのコマンドに変換する。作成されたコマンドはシステム出力用の磁気ディスクに格納される。

このようにハード・コピー・ルーチンは、COM、またはプロッタのコマンドを磁気ディスクに出力するまでの働きをする。コマンドを磁気テープに出力し、磁気テープをCOM、またはプロッタにかけて図形を出力するのは利用者のプログラムとは別の時点に行われる。したがって、利用者がプログラムの実行段階でハード・コピーに要する時間はわずかである(FACOM230-75の場合、一回のハード・コピーに要する平均待ち時間は約10秒程度である)。

磁気ディスクに出力されたコマンドはシステム出力制御プログラムによって、複数ジョブ分のCOM、およびプロッタ・コマンド混在で磁気テープに出力される。この磁気テープの内容は振分けユーティリティによって、COM専用テープ、またはプロッタ専用テープに振分けられる。これらの専用テープを、それぞれCOM、またはプロッタにかけるとハード・コピーされた図形が出力される。出力された図形は後で利用者に配布される。

CRT面上の図形のハード・コピーは実物の1/2の大きさで描かれる。つぎのFig. 6.1はハード・コピーをCOMのフィルム上に出力したときの概略図である。Fig. 6.2はハード・コピーをプロッタに出力したときの概略図である。各図形は25cm(COMの場合は、10.1倍でプリントした場合)正方形のわく付き、または無しで描くことができる。また、ハード・コピーの図形の前に、利用者を区別するジョブ・ヘッダ用の駒(COMの場合)、またはタイトル(プロッタの場合)が描かれる。

つぎのFig. 6.3, Fig. 6.4は、ハード・コピーの実例である。Fig. 6.3はCOMのフィルムに出力されたものをプリントした例であり、Fig. 6.4はプロッタへの出力例である。

### 6.2 操作

ハード・コピーをCOMに出力するか、またはプロッタに出力するかの指定、およびハード・コピーにわくを付けて出力するか、またはわく無しで出力するかの指定は、利用者のプログラムの実行前にASW(Alteration Switch)によって行われる。このハード・コピーに関するASWのセッ

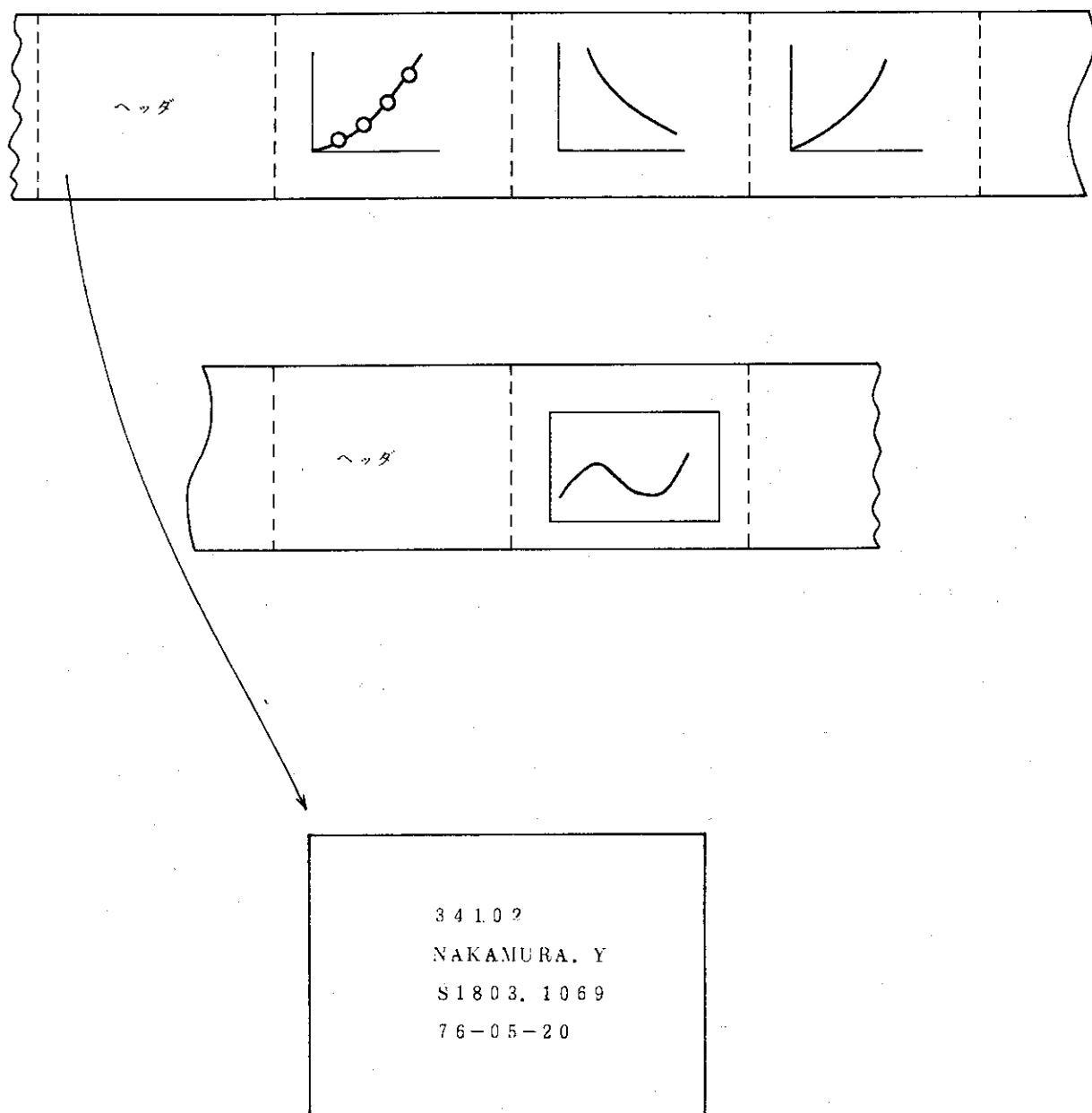


Fig. 6.1 COMへの出力形式  
(わく無しの場合)

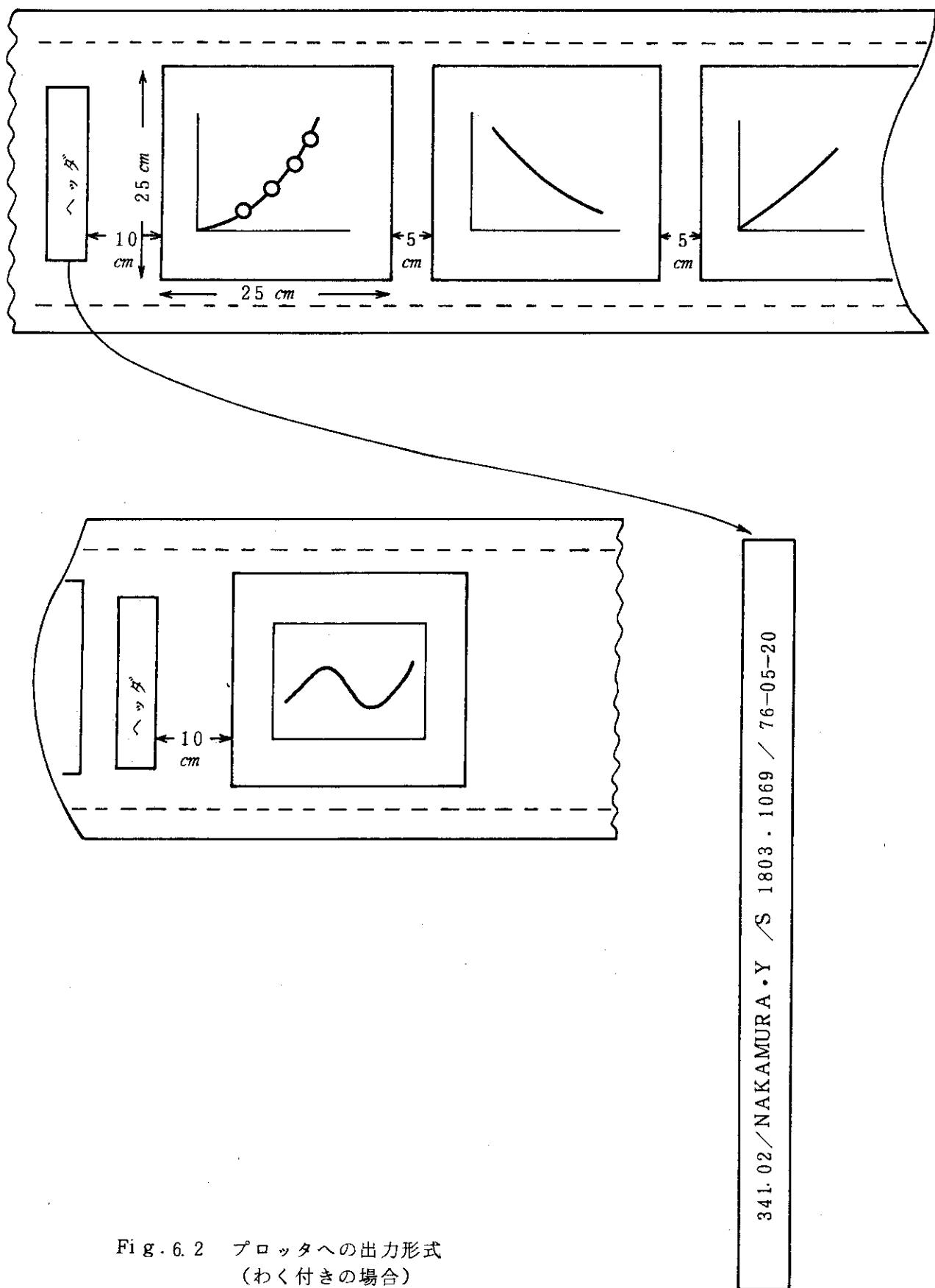


Fig. 6.2 プロッタへの出力形式  
(わく付きの場合)

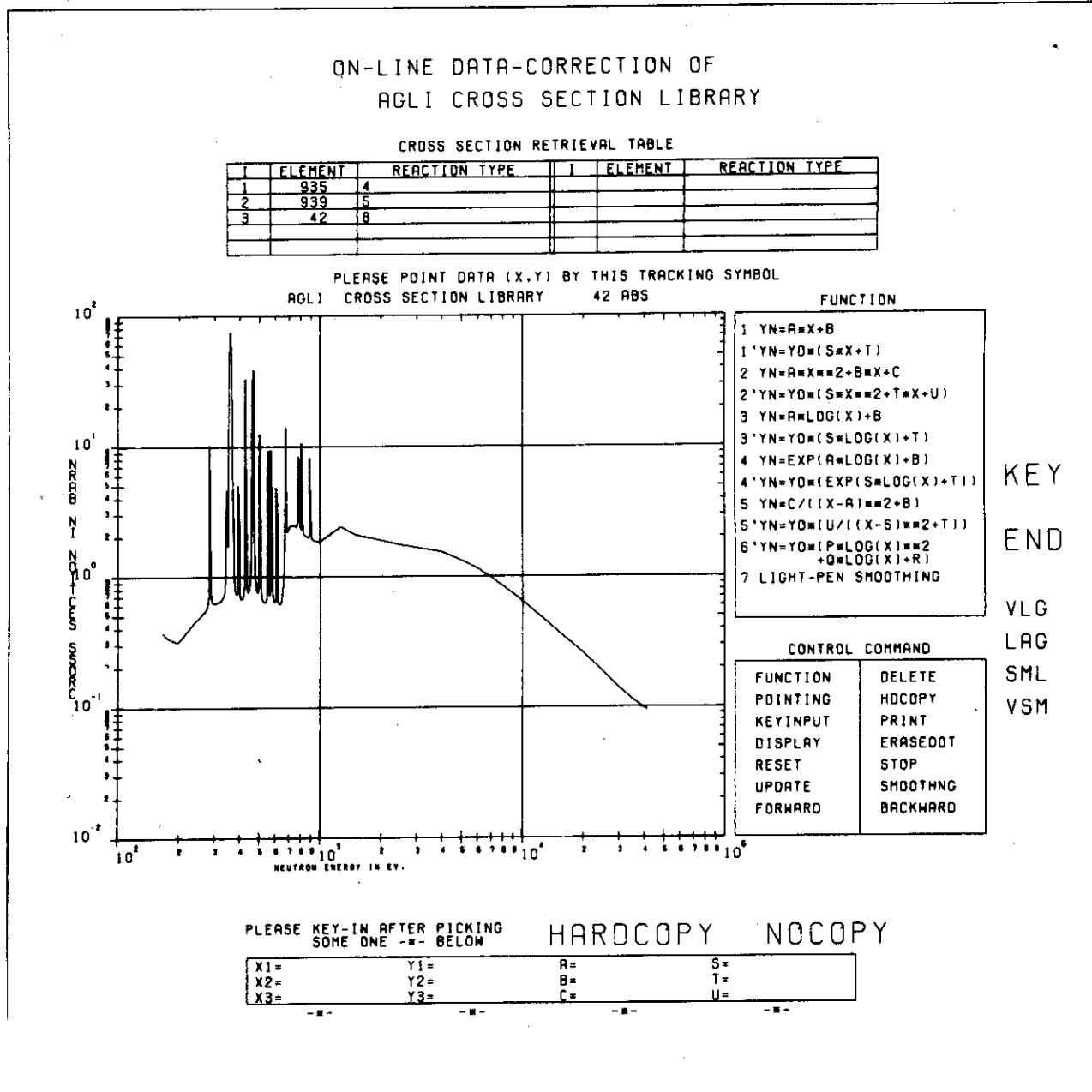


Fig. 6.3 COMへのハード・コピー出力例  
(PGS ボタン付き、わく付き)

ON-LINE DATA-FITTING  
BY ORTHOGONAL POLYNOMIALS

DEGREE OF POLYNOMIAL = 5  
 NUMBER OF RAW DATA = 15  
 NUMBER OF CONSTRAINTS = 2  
 WEIGHT IS UNIT  
 ERROR NORM  $\approx 0.16547E-01$  (L-TWO)

DATA FITTING BY ORTHOGONAL POLYNOMIALS

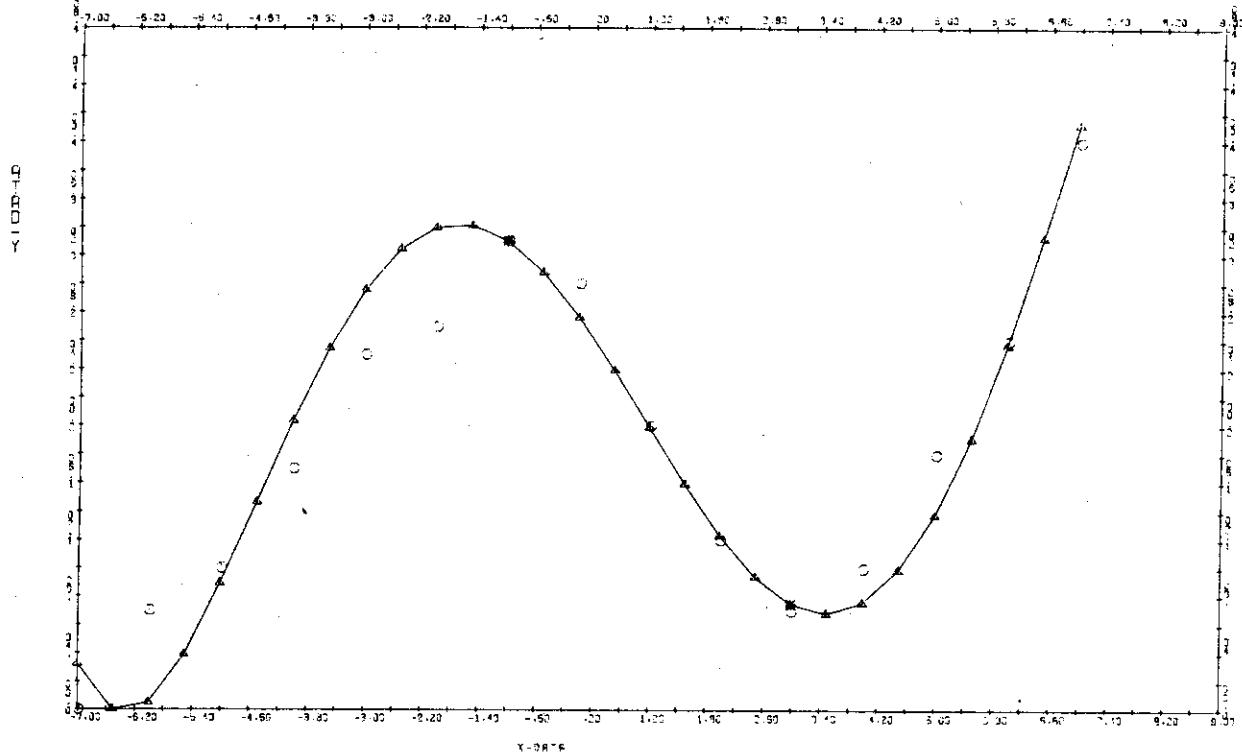


Fig. 6.4 プロッタへのハード・コピー出力例  
(PGS ボタン無し, わく無し)

トの仕方はTable 6.1に示す。

Table 6.1 ASW(8, 7, 6)のセットの仕方

ASW の番号	ASWの 状態	ON	OFF
8		ハード・コピーをする	ハード・コピーをしない
7		プロッタへハード・コピーする	COMへハード・コピーする
6		わく付きでハード・コピーする	わく無しでハード・コピーする

(注) ASW(8, 7, 6)はグラフィック・プログラムの実行前にセットされなければならない。

利用者のプログラムが実行に入り

CALL GPLOT(0.0, 0.0, 888)

が実行されると、プログラムは待ち状態に入る。ここで利用者はハード・コピー用ボタン

HARDCOPY, NOCOPY

を用いてハード・コピーを実行できる。実際、ハード・コピーはつぎの手順で行われる。

#### (1) HARDCOPYをピックする

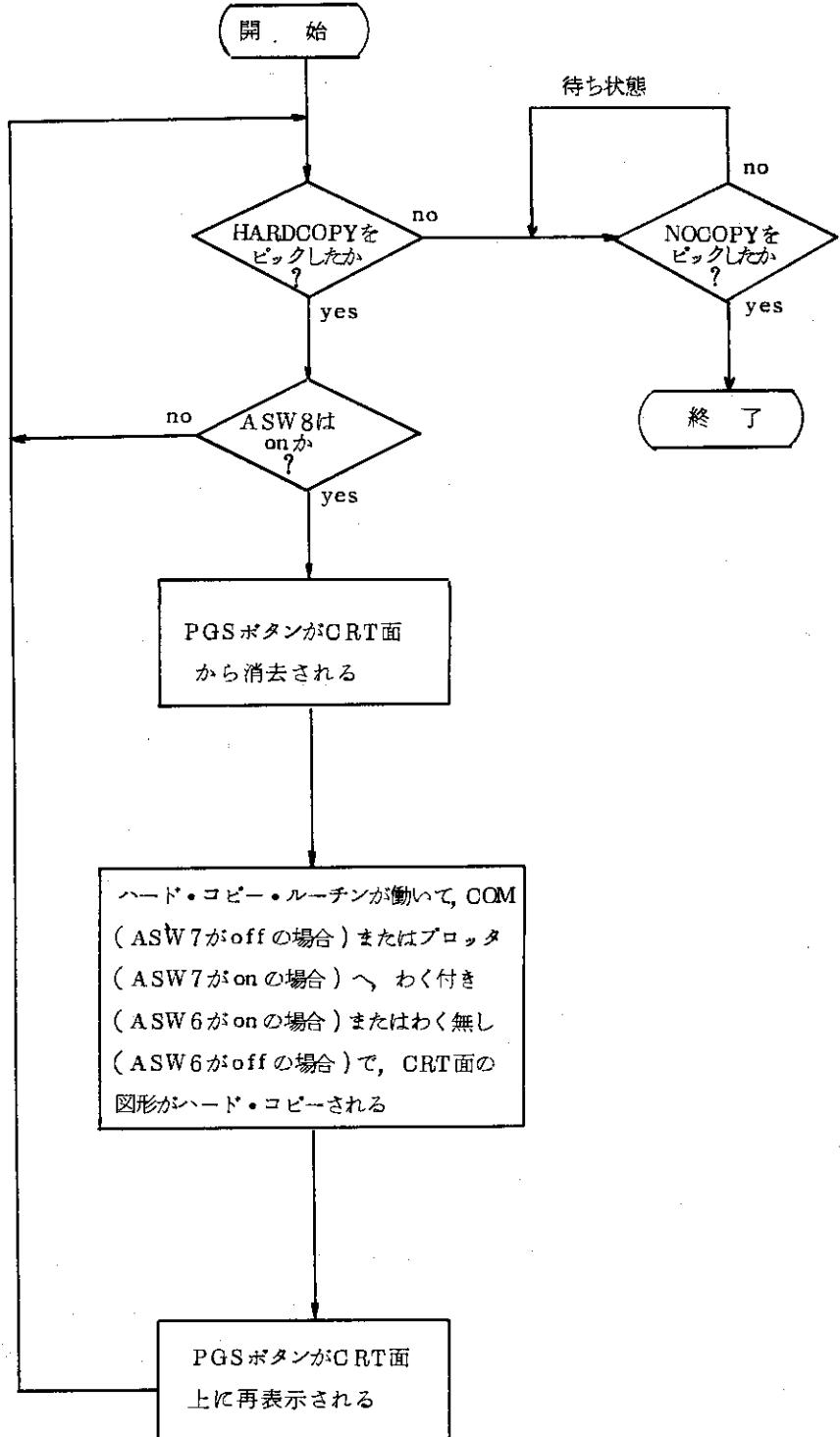
プログラムが待ち状態になったところでHARDCOPYボタンをピックすると、PGSボタンが消去され、PGSボタン以外の全図形がハード・コピーされる。ハード・コピーが終るとPGSボタンが再表示される。つぎにまたHARDCOPYをピックすると同じ図形のハード・コピーが得られる。このように何回でもHARDCOPYをピックできる。

#### (2) NOCOPYをピックする

ハード・コピーを終りにしたいときはNOCOPYボタンをピックする。ハード・コピーを全然知らない場合は、最初からNOCOPYをピックすればよい。NOCOPYがピックされるまで利用者のプログラムは待ち状態となるので注意されたい。

HARDCOPY, NOCOPYと同じ機能は、それぞれファンクション・キーの8番, 16番によっても行うことができる。ただし、ファンクション・キーの8番の場合は、PGSボタンも含めた全図形がハード・コピーされる。また、8番, 16番キーには通常ランプがついているが、ハード・コピー中は消去されている。このランプによってPGSボタンの場合と同じように、ハード・コピーの開始と終了を簡単に知ることができる。

つぎのFig. 6.5は、ハード・コピーの操作の概要をフロー・チャートで示したものである。



(注) {  
 (1) ファンクション・キーの場合は  
 PGSボタンも含めた全図形がハ  
 ード・コピーされる。  
 (2) ファンクション・キーによるハ  
 ード・コピーの間, 8番と16番  
 キーのランプが消去される。

Fig. 6.5 ハード・コピーの操作に関するフロー・チャート

## 7 おわりに

以上、FACOM 6233Aグラフィック・ディスプレイについて、PGSL1Bによるグラフィック・プログラムの作成からFACOM230-75によるプログラムの実行までの具体的な方法について述べてきた。

PGSL1Bでは、図形表示のプログラミングはプロッタ、またはCOMと同じ方法で行われ、対話処理のプログラミングは数個の命令で可能となっている。かくして、PGSL1Bによるグラフィック・プログラミングは、GSPによる場合に比べ数段簡単であると言える。

このFACOM6233Aグラフィック・ディスプレイは、バッチ処理用であり、リフレッシュ型であるため、図形表示が速いこと、ライトペンが使用できることなどの利点をもっている。しかし、遠方からの使用、または図形表示量が多い場合には不向きである。

このような場合は、TSS処理用であり、ストレージ型（蓄積型）のグラフィック・ディスプレイであるTektronix 4014が有効である。このTSS用グラフィック・ディスプレイに対しては、現在PGSL1Bと同じ考え方で、そのプログラミング用ソフトウェアPTSL1B<sup>(21)</sup>（Plotter-compatible Tektronix Graphic Subroutine Library）を開発中であり、近い内に使用可能になる予定である。

なお、PGSL1Bによるグラフィック・ディスプレイの使い方で不明な点は計算センター（中村、小沼—Tel 5369、居室127）まで御相談下さい。

### 参考文献

- (1) 富士通(株)：FACOMディスプレイ装置III(2)グラフィック・ディスプレイ装置FACOM6233A, 1972
- (2) 富士通(株)：FACOM GSP 文法編, 1973
- (3) 中村康弘, 小沼吉男, 小林健介, 鈴木忠和：グラフィック・ディスプレイの使い方と各種グラフィック・プログラムの概要, JAERI-M 5659, 1974
- (4) 中村康弘, 小沼吉男, 黒井英雄：ARCADIA-I：グラフィック・ディスプレイによる核断面積データのオンライン修正システム（内そうによる修正）, JAERI-M 5658, 1974
- (5) 中村康弘, 小沼吉男, 小林健介, 鈴木忠和：GORVFIT：直交多項式を用いた最少二乗法による対話型カーブ・フィッティング・システム, JAERI-M 5596, 1974
- (6) 幾島毅, 小沼吉男, 中村康弘：GTOROTO-3, 4：グラフィック・ディスプレイによるブロック状炉心の地震シミュレーション・プログラム, JAERI-M 5981, 1975
- (7) 中村康弘, 小沼吉男, 生田目健, 鈴木紀男：GROSA-II：グラフィック・ディスプレイを用いたROSA-IIのための対話型データ解析システム, JAERI-M 6237, 1975
- (8) 亀有昭久, 二宮博正, 相川裕史, 鈴木康夫：グラフィック・ディスプレイを用いたポロイダルコイル設計プログラム（臨界プラズマ試験装置設計報告, XI), JAERI-M 6324, 1975
- (9) 中村康弘, 小沼吉男：グラフィック・ディスプレイによるプロッタ・プログラムのオンライン・デバッキング, JAERI-M 6391, 1976

## 7 おわりに

以上、FACOM 6233Aグラフィック・ディスプレイについて、PGSL1Bによるグラフィック・プログラムの作成からFACOM230-75によるプログラムの実行までの具体的な方法について述べてきた。

PGSL1Bでは、図形表示のプログラミングはプロッタ、またはCOMと同じ方法で行われ、対話処理のプログラミングは数個の命令で可能となっている。かくして、PGSL1Bによるグラフィック・プログラミングは、GSPによる場合に比べ数段簡単であると言える。

このFACOM6233Aグラフィック・ディスプレイは、バッチ処理用であり、リフレッシュ型であるため、図形表示が速いこと、ライトペンが使用できることなどの利点をもっている。しかし、遠方からの使用、または図形表示量が多い場合には不向きである。

このような場合は、TSS処理用であり、ストレージ型（蓄積型）のグラフィック・ディスプレイであるTektronix 4014が有効である。このTSS用グラフィック・ディスプレイに対しては、現在PGSL1Bと同じ考え方で、そのプログラミング用ソフトウェアPTSL1B<sup>(21)</sup>（Plotter-compatible Tektronix Graphic Subroutine Library）を開発中であり、近い内に使用可能になる予定である。

なお、PGSL1Bによるグラフィック・ディスプレイの使い方で不明な点は計算センター（中村、小沼 Tel 5369、居室127）まで御相談下さい。

### 参考文献

- (1) 富士通株：FACOMディスプレイ装置III(2)グラフィック・ディスプレイ装置FACOM6233A, 1972
- (2) 富士通(株)：FACOM GSP 文法編, 1973
- (3) 中村康弘, 小沼吉男, 小林健介, 鈴木忠和：グラフィック・ディスプレイの使い方と各種グラフィック・プログラムの概要, JAERI-M 5659, 1974
- (4) 中村康弘, 小沼吉男, 黒井英雄：ARCADIA-I：グラフィック・ディスプレイによる核断面積データのオンライン修正システム（内そうによる修正）, JAERI-M 5658, 1974
- (5) 中村康弘, 小沼吉男, 小林健介, 鈴木忠和：GCRVFIT：直交多項式を用いた最少二乗法による対話型カーブ・フィッティング・システム, JAERI-M 5596, 1974
- (6) 幾島毅, 小沼吉男, 中村康弘：GTOROTO-3, 4：グラフィック・ディスプレイによるブロック状炉心の地震シミュレーション・プログラム, JAERI-M 5981, 1975
- (7) 中村康弘, 小沼吉男, 生田目健, 鈴木紀男：GROSA-II：グラフィック・ディスプレイを用いたROSA-IIのための対話型データ解析システム, JAERI-M 6237, 1975
- (8) 亀有昭久, 二宮博正, 相川裕史, 鈴木康夫：グラフィック・ディスプレイを用いたポロイダルコイル設計プログラム（臨界プラズマ試験装置設計報告, XI), JAERI-M 6324, 1975
- (9) 中村康弘, 小沼吉男：グラフィック・ディスプレイによるプロッタ・プログラムのオンライン・デバッキング, JAERI-M 6391, 1976

- (10) 中村康弘, 小沼吉男: PGSLIB-A Plotter-compatible Graphic Subroutine Library, JAERI-M (to be published)
- (11) 日本IBM(株): 2250映像表示装置/GSP, 1971
- (12) 吉沢ビジネス・マシンズ(株): CALCOMP プログラミング・マニュアルー I, 1969
- (13) 吉沢ビジネス・マシンズ(株): CALCOMP プログラミング・マニュアルー II, 1969
- (14) 中村康弘, 小沼吉男, 青木 裕: グラフィックCOMのプログラミング, JAERI-M (to be published)
- (15) 長谷川 明: 汎用グラフ作成サブルーチンGPLOT1, 自動グラフ作成コードGPLOTCの開発, JAERI-memo 4255 (公開), 1970
- (16) CALCOMP, INC: THREED: A Perspective Drawing Software System, 1969
- (17) 小沼吉男, 中村康弘: テキスト・モード・グラフィック・サブルーチン・パッケージ, JAERI-M (to be published)
- (18) 小沼吉男, 中村康弘: 対話型計算処理用グラフィック・サブルーチン・パッケージ, JAERI-M (to be published)
- (19) 中村康弘, 小沼吉男: PGSLIB/BASIC: グラフィック・プログラミングのための基本ソフトウェア, JAERI-M 6023, 1975
- (20) 富士通(株): FACOM 230 M-VL/VII GSP602/GSP603 使用手引書, 1974
- (21) 小沼吉男, 中村康弘: TSS用グラフィック・ディスプレイのプログラミング, JAERI-M (to be published)

## 付録 PCSILB/BASIC 命令の呼び出し形式

## 1. PGS の命令

## 1.1 PGS 命令

- (1) GPLOTS 命令 (Open Graphic Display)  
CALL GPLOTS (gdoa, nwords)
- (2) GPLOT 命令 (Plot Vector and Close Graphic Display)  
CALL GPLOT (xcoor, ycoor, ibeam)
- (3) GSYMBL 命令 (Plot Symbols)
  - (a) CALL GSYMBL (xcoor, ycoor, height, string, angle, nchars)
  - (b) CALL GSYMBL (xcoor, ycoor, height, icode, angle, nbeam)
- (4) GNUMBR 命令 (Plot Numbers)  
CALL GNUMBR (xcoor, ycoor, height, fltpn, angle, ndecs)
- (5) GSSCALE 命令 (Scale Data)  
CALL GSSCALE (data, width, ndata, incrmt, div)
- (6) GAXIS 命令 (Plot Axis)  
CALL GAXIS (xcoor, ycoor, title, nchars, width, angle, start, delta, div)
- (7) GLINE 命令 (Plot Line)  
CALL GLINE (xdata, ydata, ndata, incrmt, lintyp, icode)
- (8) GFACTR 命令 (Set Scale Factor)  
CALL GFACTR (factor)
- (9) GWWHERE 命令 (Where is the beam?)  
CALL GWWHERE (X, Y, FACTOR)
- (10) GNEWPN 命令 (Set Pen Number)  
CALL GNEWPN (ipen)

## 1.2 接続命令

- (1) PLOTS 命令  
CALL PLOTS (gdoa, nwords)
- (2) PLOT 命令  
CALL PLOT (xcoor, ycoor, ibeam)
- (3) SYMBOL 命令  
CALL SYMBOL (xcoor, ycoor, height, ibcd, angle, nchars)
- (4) NUMBER 命令  
CALL NUMBER (xcoor, ycoor, height, fltpn, angle, ndecs)
- (5) SCALE 命令  
CALL SCALE (data, width, ndata, incrmt, div)

- (6) AXIS 命令  
CALL AXIS (xcoor, ycoor, title, nchars, width, angle, start, delta, div)
- (7) LINE 命令  
CALL LINE (xdata, ydata, ndata, incrmt, lintyp, icode)
- (8) FACTOR 命令  
CALL FACTOR (fctr)
- (9) WHERE 命令  
CALL WHERE (X, Y, FCTR)
- (10) NEWPEN 命令  
CALL NEWPEN (ipen)

### 1.3 拡張命令

- (1) GPLOT 命令 (Control Graphic Display)  
CALL GPLOT (xcoor, ycoor, ibeam)

## 2. 簡易 GSP の命令

### 2.1 割込み処理命令

- (1) GETIDF 命令 (Get Attention Identifier)  
CALL GETIDF (NAME1, NAME2)
- (2) GLIST 命令 (Specify Name List)  
CALL GLIST (name2-1, name2-2 [, name2-1, name2-2] ....)
- (3) GJUMP 命令 (Jump to Statement Number)  
CALL GJUMP (Ystno, id-1, id-2, Ystno 1 [, Ystno2] ....)

### 2.2 トランкиング処理命令

- (1) GBGTRK 命令 (Begin Lightpen Tracking)  
CALL GBGTRK (xcoor, ycoor, distan)
- (2) GRDTRK 命令 (Read the Current Location of the Tracking Symbol)  
CALL GRDTRK (XCOOR, YCOOR)
- (3) GENTRK 命令 (End Lightpen Tracking)  
CALL GENTRK

### 2.3 データ入出力処理命令

- (1) GREADD 命令 (Read Data from Alphanumeric Keyboard)  
CALL GREADD (xcoor, ycoor, height, DATA, angle, Yfmtno)

(2) GWRITE 命令 (Write Data on the CRT)

```
CALL GWRITE (xcoor, ycoor, height, data, angle, Yfmtno)
```

3. 汎用 GSP の命令

3.1 汎用グラフ表示命令

(1) GPLOT1 命令 (Plot Data)

```
CALL GPLOT1 (iplt, imax3, x, y, withx, withy, ip, np, ist, nlogx,  
nlogy, xwide, ywide, ixmin, iymin, axl, ax2, ayl, ay2,  
mscale, ratiox, ratioy)
```

3.2 ライトペン・スムーズィング命令

(1) GSMOOTH 命令 (Smooth Data)

```
CALL GSMOOTH (xdata, ydata, YSMOOT, ndata, ISMTH, namlsc, nam2sc,  
nam1lb, nam2lb)
```