

航空宇宙技術研究所資料

TECHNICAL MEMORANDUM OF NATIONAL AEROSPACE LABORATORY

TM-589

2次元グラフィックディスプレイ装置用
ソフトウェアパッケージ

末松和代・磯部俊夫

1988年6月

航空宇宙技術研究所
NATIONAL AEROSPACE LABORATORY

目 次

1. はじめに	1
2. NSシステムにおける図形処理装置と2次元GDの役割	1
2.1 図形処理装置の性能	1
2.2 NSシステムにおける2次元GDの役割	4
3. 2次元GD表示プログラム	6
3.1 初期設定	6
3.2 画面の初期化	6
3.3 図形データの表示	6
3.3.1 データ変換プログラム	6
3.3.2 図形データ表示処理の流れ	10
3.3.3 図形データの生成・変換・表示処理の流れ	11
3.4 追加データの表示とインタフェースプログラムの呼び出し	14
3.5 メニュー処理	14
4. 新モニタプログラム	17
4.1 中間データファイルの形式	17
4.1.1 初期ブロック	19
4.1.2 データブロック	19
4.1.3 終了ブロック	20
4.1.4 その他	20
4.2 モニタジョブ名の決定とファイルのオープン処理	22
4.3 中間データファイルの初期化	23
4.4 GCP形式の図形データの作成	24
4.4.1 セグメントのオープン処理	24
4.4.2 図形データの変換処理	24
4.4.3 ページ処理	25
4.4.4 最終ページ処理	25
4.5 図形表示のための初期化処理	26
4.5.1 表示プログラムの初期化処理	26
4.5.2 画面の初期化およびウィンドの設定	26

4.6	図形データの生成および変換処理	26
4.7	追加データの表示とインタフェースプログラムの呼び出し	26
4.8	メニュー処理	26
4.9	中間データファイルの処置	29
5.	インタフェースプログラムと形式変換プログラム	30
5.1	インタフェースプログラム	30
5.1.1	流線インタフェースプログラム	30
5.1.2	オイルフローインタフェースプログラム	30
5.1.3	文字データインタフェースプログラム	31
5.2	図形データ形式変換プログラム	31
5.2.1	図形変換時の制御データ	32
5.2.2	2次元GD表示プログラムおよび新モニタプログラムでの処理	33
5.2.3	NLP出力用形式変換プログラム	34
5.2.4	OPR出力用形式変換プログラム	34
5.2.5	PP・XY出力用形式変換プログラム	35
6.	おわりに	36

2次元グラフィックディスプレイ装置用 ソフトウェアパッケージ*

末松和代** 磯部俊夫**

1. はじめに

昭和62年2月、航技研にはそれまでの汎用計算機システムに加えて数値シミュレーションシステムが導入され、両者を有機的に結合したNSシステムの運用が開始された。NSシステムはFACOM M-780をフロントエンドプロセッサとし、FACOM VP-400およびFACOM VP-200をバックエンドプロセッサとした疎結合方式の複合計算機システムであり、最大1GBの主記憶を使用した大規模数値シミュレーションが超高速で実行できる。

これに伴ない、ユーザプログラムから発生する数値データは膨大なものとなるため、もはや実行結果の検証を数値で追うことは不可能であり、必然的に可視化技術の開発が不可欠となった。このため、NSシステムには数多くの図形処理装置が導入され、それぞれの装置を使用するために各装置独自の図形出力ライブラリが提供されている。しかしながら、これらの装置を導入し、使用できるようにしただけでは、ユーザは新規に導入された図形処理装置用の図形出力ライブラリ機能を習得し、使用する装置毎にそれぞれ専用の作図プログラムを開発しなければならない。

NSシステムでは、ユーザに課せられたこのような負担を完全に解消し、本来の数値シミュレーションプログラムの開発効率の向上を目指して、航技研独自のライブラリであるGCPの開発・改良に取り組んでいる。現在のところ、NSシステムにおいてユーザが利用できる図形処理機能の環境は図1.1のと

おりであり、バッチ処理ではカルコンライブラリを、TSS処理ではGCPライブラリを習得するだけで、作図プログラムの開発ができるようになっている。その上、これらの図形処理装置の中でも、設置台数が特に多い2次元グラフィックディスプレイ装置（以下2次元GDと略記する）に対しては、3次元グラフィックディスプレイ装置（以下3次元GDと略記する）の代替装置とするためのソフトウェア機能、表示されている図形データを簡単に他の図形処理装置へ出力するための機能、XYプロッタ出力用の図形データを表示する機能等が用意されているために、ユーザは非常に容易にかつ効率的に新規導入の各種図形処理装置を使用できるようになっている。

図1.1のNSシステムの図形処理環境の中で、とくに2次元GD用に開発したソフトウェアパッケージは2重枠で示されているが、本稿では、これらのソフトウェアパッケージの機能およびプログラム構造について詳述する。

2. NSシステムにおける図形処理装置と 2次元GDの役割

NSシステムには機能・性能の異なる数多くの図形処理装置が設置されている。本章では、これらの装置の性能・機能を説明するとともに、特に2次元GDに要求される役割について述べる。

2.1 図形処理装置の性能

NSシステムに設置されている各種図形処理装置の性能は下記のとおりである。

(1) XYプロッター装置とプリンタプロッター装置

* 昭和63年5月26日受付

** 数理解析部

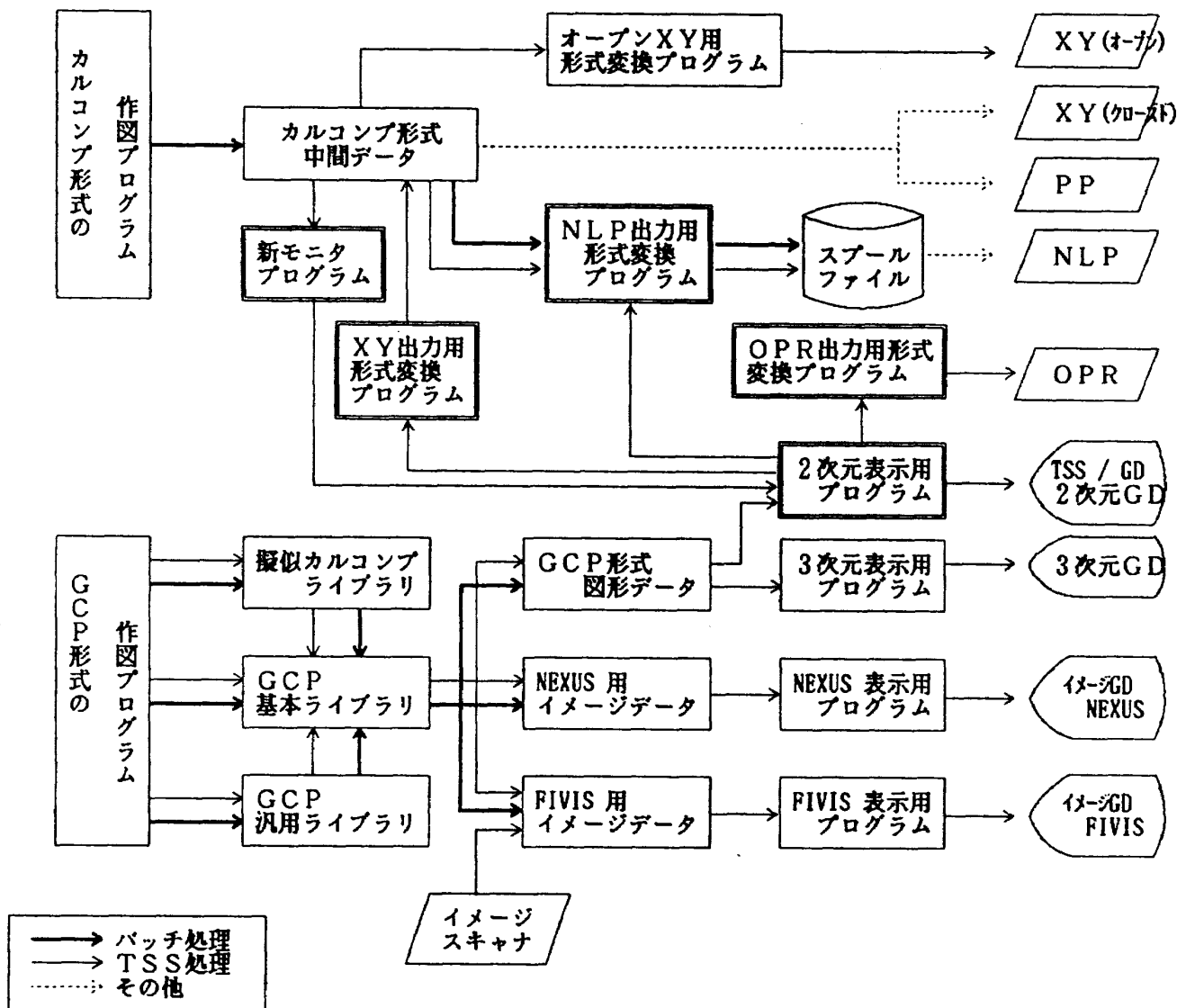


図 1.1 NSシステムにおける図形処理の流れ

表 2.1 XY、PP、NLP、OPRの仕様および性能

	クローズドXY		オープンXY	PP	NLP	OPR
	オフライン	オンライン	オンライン			
プロット幅	86 / 29 cm	86 / 29 cm	27 cm	27 cm	27 cm	19.5 cm
ステップ幅	0.05 mm	0.025 mm	0.1 mm	—	—	—
プロット速度	—	—	—	3.68cm/秒	32ページ/分	16ページ/分
使用可能色	黒、青、赤	黒、青、赤、緑	黒、青	黒	黒	黒

(注) プロット速度には図形転送時間は含まない。

XYプロッター装置（以下XYと略記する）とプリンタプロッター装置（以下PPと略記する）は前システムから継続使用されている図形処理装置であり、表 2.1 に示すハードウェア性能をもつ。これら

の装置はカルコンライブラリを介して使用できる。
 (2) 日本語ラインプリンタ装置とオフィスプリンタ装置
 日本語ラインプリンタ装置（以下NLPと略記す

表 2.2 2次元GD、3次元GD、イメージGDの仕様および性能

型名	2次元GD		3次元GD	イメージGD	
	F6683	F9434/F6242	COMTEC DS351B	F6510	NEXUS
画面	14インチ	20インチ	20インチ	20インチ	20インチ
分解能	960 x 672	1024 x 800	1280 x 1024	1280 x 1024	480 x 512
座標定義空間	960 x 672	1024 x 800	32000 x 32000	—	—
表示可能色 (同時表示)	7色	7色/16色	1670万色 (4096色)	1670万色	1670万色
接続機器	ハードコピー： カラー熱転写方式 A4、7色		ハードコピー： カラー熱転写方式 A4、4912色 カメラ： ポラロイド、35mm	カメラ： ポラロイド 35mm	

る)とオフィスプリンタ装置(以下OPRと略記する)はNSシステムで新設された図形処理装置であり、通常のラインプリンタ機能の他に、日本語の出力や図形出力が行える。これらの装置は表2.1に示すハードウェア性能をもち、PSPライブラリを介して使用できる。

(3) 2次元GD

2次元GDは図形データを2次元座標で定義して使用するものであり、線画だけでなく多角形の中塗り機能もある。使用できる色は7色であり、カラー熱転写方式のハードコピーが接続されているものもある。NSシステムには分解能、画面サイズ等の異なる3種類の2次元GDが設置されている。これらの装置は表2.2に示すハードウェア性能をもち、GSPライブラリを介して使用できる。

(4) 3次元GD

3次元GDは図形データを3次元の座標で定義して使用するものであり、図形データの階層化機能^{注1)}、表示した図のローカル処理機能^{注2)}、隠線処理・隠面処理機能、シェーディング機能^{注3)}、1670万色の中の4096色が使用できる多色表示機能等がハードウェアで用意されている。3次元GDに接続されている装置としてはポラロイドカメラ、35mmカメラ、4912色表示可能なカラー熱転写方式のハードコピーがあり、撮影機が接続されているものもある。

これらの装置は表2.2に示すハードウェア性能をもち、GRIPⅡライブラリを介して使用できる。

(5) イメージディスプレイ装置

イメージディスプレイ装置(以下イメージGDと略記する)は、1670万色の同時表示機能を用いて画面上の各ドットに適切な色づけすることによって図形データを表示する装置である。したがって、1本の線を描くときにも直線の位置を座標で定義するのではなく、その線に対応したドットに色づけを行わなければならない。しかし、ある実験結果の圧力データを図形処理する場合など、画面上のすべてのドットに対し圧力に対応した色づけを行うため、等高線図のような線画処理では見落としがちな特異点が発見できることもある。

NSシステムには、表2.2に示すように2種類のイメージGDがあり、ハードウェア性能も使用する図形出力ライブラリも異なる。また、装置によってはイメージスキャナー、ポラロイドカメラ、35mmカメラ、4912色表示可能なカラー熱転写方式のハードコピーや、撮影機が接続されているものもある。

注1) 作図プログラムを作成する際に図形データを回転、移動、拡大・縮小等の図形変換できる段階のレベルに分ける機能

注2) 画面上に表示されている図形データの回転、移動、拡大・縮小等を作図プログラムを介さずに行う機能

注3) 多角形の頂点の色を指定することにより、自動的に多角形内の色を補間してなめらかな色づけを行う機能

2.2 NSシステムにおける2次元GDの役割

NSシステムに設置されている図形処理装置には、それぞれ特有の機能があることは前項で述べた。このうち、イメージGDや3次元GDのようにすぐれたハードウェア機能をもつグラフィックディスプレイは、その機能を十分に発揮できるような環境を整えることが重要である。しかし、このように高機能なグラフィックディスプレイ装置は高価であるため、多数設置するのは困難である。このため、多数設置されている2次元GDの果たすべき役割が自ずと重要なものとなってくる。したがって、2次元GDには、本来の機能を十分に発揮できるような環境を整えるだけではなく、以下に述べるような役割を持たせることが必要不可欠となる。

(1) 3次元GDの代替装置としての役割

3次元GDは高機能、高性能であるが、前述したように設置台数が少なく設置場所も限られているため、手軽に利用できる2次元GDにその代替装置としての役割を持たせることが必要となる。それが実現できるようになれば、3次元GD用の作図プログラムも2次元GD上で実行可能となり、作図プログラム作成処理が短縮化される。このためには、2次元GDのハードウェアでは実現できない3次元GDのハードウェア機能を模擬するソフトウェアの開発が必要となってくる。

しかし、出力ライブラリが異なり、機能・性能の異なる2次元GDを3次元GDの代替装置と呼べるようにするためには、プログラム修正が不要でなければならない。このことはユーザの側からすれば、非常に重要でかつ不可欠なことである。3次元GDで作図プログラムが全く同様に動作するか否かは別として、出力装置を変更するたびにプログラムの修正が必要となるのでは、単に出力装置が複数あるというだけで、代替装置とはいえない。3次元GDの機能を意識して作成したプログラムを2次元GDで実行しようとした場合、ソフトウェアでは対処できない機能は当然あるが、その場合でもプログラム修正をしなければエラーとなるようであってはならない。

3次元GD用のプログラムを2次元GDで実行させた場合、ハードウェアの相違によりそのままでは

同様に処理ができない機能として下記の2つのレベルがある。

第一のレベルは、ソフトウェアを駆使することによりハードウェアの欠如に対処することができる機能である。たとえば、図形データの階層化機能、表示データのローカル処理機能、隠線・隠面処理機能等がこれに該当する。このうち、図形データの階層化機能やローカル処理機能は自動的に2次元の表示用プログラムに組み込む方がよいが、処理に時間を要する隠線・隠面処理機能は必要に応じて使用できるように、汎用プログラムにその機能を用意すべきである。

第二のレベルは、ハードウェアそのものに依存する機能である。これに対しては適切に処理することが必要である。たとえば、多色表示機能等がこれに該当する。2次元GDでは、3次元GDで使用できる1670万色をそのまま表示することはできないが、指定された色を2次元GDで使用できる7色の中のもっとも適切な色に変換できればよい。

なお、第一のレベルでも第二のレベルでも処理できない機能は2次元GDでは実現できない機能となるが、使用してもプログラムエラーとならないようにしなければならない。

(2) イメージGDの代替装置としての役割

イメージGDは3次元GD以上に設置台数が少く、設置場所もセンターのオープン室だけである。そのため、2次元GDにイメージGDの代替装置としての機能があればより便利であることは明白である。

イメージGDに図を表示する場合には、まず図形データを画面上の各ドットに対する色のデータ（これをイメージデータと呼ぶ）に変換し、このデータをイメージGDに転送し表示する。したがって、2次元GDにイメージGDの機能をもたせるためには、イメージデータに従って画面上のドットに対し任意の色づけを行う機能が必要となる。

しかし、残念なことに、我々がおこなった予備テストでは、1画面分の各ドットに対する色づけ・表示に1時間以上かかることが判明した。そのため、イメージGDの代替装置としての利用は、ソフト上では可能であると考えられるが、現時点では使用に耐えられないとの結論に致った。だが、この機能は

処理の高速化が現実のものとなった時点で有効とすべき機能である。

(3) 汎用作図プログラムのインターフェース機能としての役割

汎用作図プログラムは、格子図、等高線図といった汎用的な作図を簡単に行えるようにする目的で作成されたプログラムであり、その出力はそのまま図形データの作成を行うGCP基本ライブラリの入力となる。

このうち、流線図、オイルフロー図作成用のプログラムの場合には、それらの開始点をあらかじめ汎用プログラムに対する入力データとして用意しておく必要がある。しかし、これらの開始点は試行錯誤で設定していくことが多いため、開始点変更の都度入力データを変更し実行しなおすのも煩わしく、作図の完成にも時間を要する。そこで、GCPの2次元GD表示プログラムに汎用作図プログラムのインターフェース機能をもたせ、図形データの表示後、開始点等を会話形式で指定できるようにする必要がある。この機能は、流線、オイルフローの開始点に限らず、必要に応じて充実させていく必要がある。

(4) 図形データの編集機能としての役割

作図プログラムで作成された図形データは、出力結果の確認後、図の編集、隠線・隠面処理、表示色の変更、タイトルの追加等の編集処理を必要とすることが多い。このうち、複数の図の合成・重ね合わせ等の処理や隠線・隠面処理は、2次元GD用のプログラというよりも汎用的な要素が濃いため、汎用プログラムにその機能を用意し、ユーザが適時使用できるようにすべきであると考えられる。しかし、表示色の変更やタイトルの追加機能の場合には、ユーザプログラムの手直しが不要となり、出来上がった図を見ながら指定するため、簡単でかつ適切に行えると考えられる。そこで、2次元GD使用時に利用できる機能として、表示色の変更、文字列の追加機能を用意すべきである。

(5) XY, PP出力用図形データのモニタ機能としての役割

XY, PP用に出力した図形データは、実際に出力されるまで作図処理に誤りがあるか否かの確認が

できない。そのため、作図プログラムのディバック処理には時間を要し、作図処理に誤りがあったときには結果的に意味のないものまで出力することになるため、無駄も多い。そこで、実際の出力に先立って図形データをモニタしようという機能が、2次元GDによるモニタ機能である。

航技研では、かなり以前からモニタ機能の必要性が検討され、そのためのプログラムが用意されていた。しかし、初期のモニタプログラムは単に図形データをモニタする目的であったこと、使用した2次元GDにカラー表示機能がなかったことから、図形データをそのまま表示しただけのものであり、図形変換のためのコマンドも拡大だけであった。そのため、利用者からは、図形データの縦横比が変わる、カラー表示ができないといった不満が徐々にでてきたので、その都度プログラムの改良を重ねてきた。しかし、新システムの導入を機にXY, PP用図形データのモニタ処理の効率化を検討した結果、モニタ処理を行う単位となるページという概念の導入、それに伴うページ指定機能等の追加、ローカル処理機能の充実等を盛り込んだ新モニタプログラムの作成が不可欠であるとの結論に致った。

(6) 図形データの形式変換機能としての役割

2次元GDで結果を確認した図形データをXY, PP, NLP, OPRのように、本来バッチ処理で使用する図形処理装置へ簡単に出力できると便利である。これらの装置は、2次元GDに接続されているハードコピーに比べて精度が良い上、2次元GDで作図データを確認してから転送するため、バッチ処理ですべて実行するよりも手軽にディバックができ、ターン・アラウンド・タイムも短くなる。しかし、図形出力ライブラリの異なるこれらの装置への出力をユーザレベルで行うことは、ユーザの負担が多く、効率的でない。そこで、必要となる装置間の形式変換プログラムを作成し、2次元GDへの表示後、コマンド1つでそのプログラムが起動できるようにする必要がある。なお、この変換プログラムは、作図プログラムで作成された図形データだけでなく、ローカル処理によって回転、拡大・縮小、移動した図形データや、オイルフロー、流線、文字列等を追加した図形データも取り扱え、さらに出力装置に応じた

シザリング処理^{注4)}等も行えるものでなければならない。

3. 2次元GD表示プログラム

2次元GD表示プログラムは、GCP形式の作図プログラムによって作成された図形データおよび制御テーブルを参照して、2次元GDに図形表示を行うプログラムである。このプログラムは第2章で要求された機能を実現するために、図3.1に示す流れで処理を行っている。図中の2重枠で囲まれたプログラムは、第5章で述べる3つのインターフェースプログラムと4つの図形データ形式変換プログラムを表している。

本章では、2次元GD表示プログラムの処理の流れに沿って、その機能説明を行っていく。

なお、2次元GD表示プログラムはGCPライブラリの中の1プログラムであり、先に作成された3次元GD用のソフトウェアパッケージと同じ思想のもとに作成されている。したがって、本稿ではGCPライブラリの概要等については省略するので、必要に応じて資料1を参照されたい。

3.1 初期設定

2次元GD表示プログラムでは、プログラムの初期設定として、以下に示すようにスクリーンウィンドの設定および制御変数の初期化を行っている。

(1) スクリーンウィンド省略値の設定

2次元GDに図を表示する際には、スクリーンウィンドの設定が必要となる。しかし、スクリーンウィンドとして定義できる値の範囲が装置によってそれぞれ異なるため、2次元GD表示用プログラムでは、使用されている装置の種類を調べて、その装置で定義できるスクリーンウィンドの最大範囲をスクリーンウィンド省略値として設定する。

作図プログラムでスクリーンウィンドをとくに設定していないセグメントは、この省略値を使用することによって、画面全体に表示できる。

注4) ある一定の領域からはみ出した部分のデータを取り除くための処理で、ここでは出力されたデータの中から、使用装置に許されていない範囲のデータを取り除く処理を指す。

(2) 制御変数の初期化

ここで初期化する制御変数のIST, IOIL, IMMは、流線プログラム, オイルフロープログラム, 文字データプログラムによって生成された追加データがあるか否かを0か1かで表す変数であり、それぞれのインターフェースプログラムを呼び出す際の引数ともなる。

これらの変数は初期設定時に0をセットするが、流線, オイルフロー開始点の入力処理メニューまたは文字入力処理メニューが選択されたとき、第5章で述べるインターフェースプログラムを介して、それぞれのプログラムが呼び出され、その処理によって適時変更される。

3.2 画面の初期化

画面の初期化処理では、図形データの表示に先立って画面に表示中の図形データ等があればそれらをすべて消去する。

3.3 図形データの表示

作図プログラムによって作成された図形データに対し、クラス情報, セグメント情報およびローカル処理時の回転情報を有効にするために、2次元GD表示プログラムでは図形データファイルから取り出したデータを図3.2に示す流れで処理し、表示している。

以下に、図形表示に使用する図形変換プログラムおよび2次元GD表示プログラムで行っている図形変換処理の流れについて述べる。

3.3.1 データ変換プログラム

本項では、2次元GD表示プログラムが図形データを表示する際に使用するデータ変換プログラムについて簡単に説明する。

(1) ローカル処理変換プログラム

3次元GDの場合、作図プログラムによって表示した図形は、作図プログラムとは独立に、回転, 移動, 拡大・縮小およびその初期化を行えるようなローカル処理機能がハードウェアで用意されている。この機能を2次元GDで模擬しようとした場合には、回転, 移動, 拡大・縮小およびその初期化を行うた

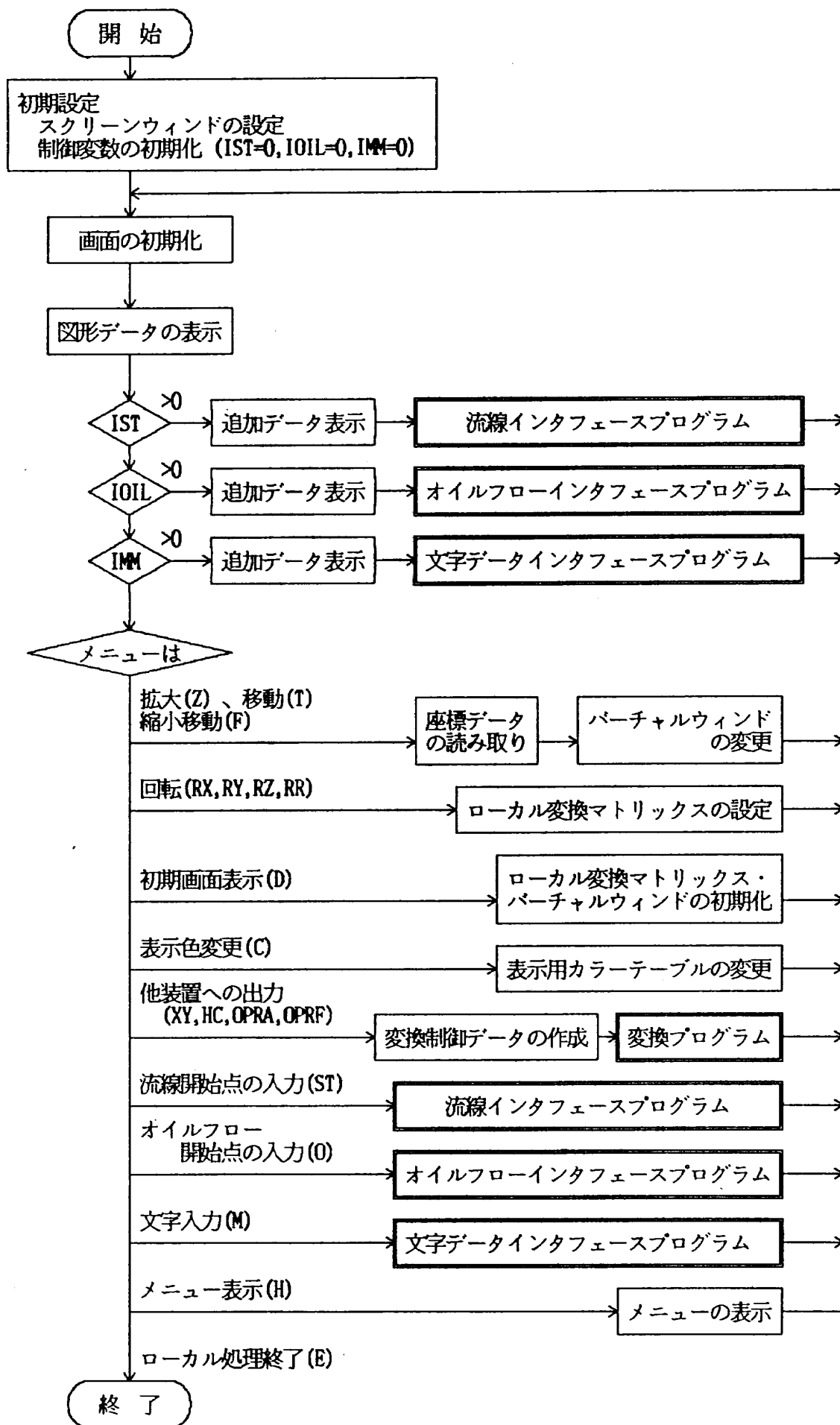


図 3.1 2次元GD表示プログラムの処理の流れ

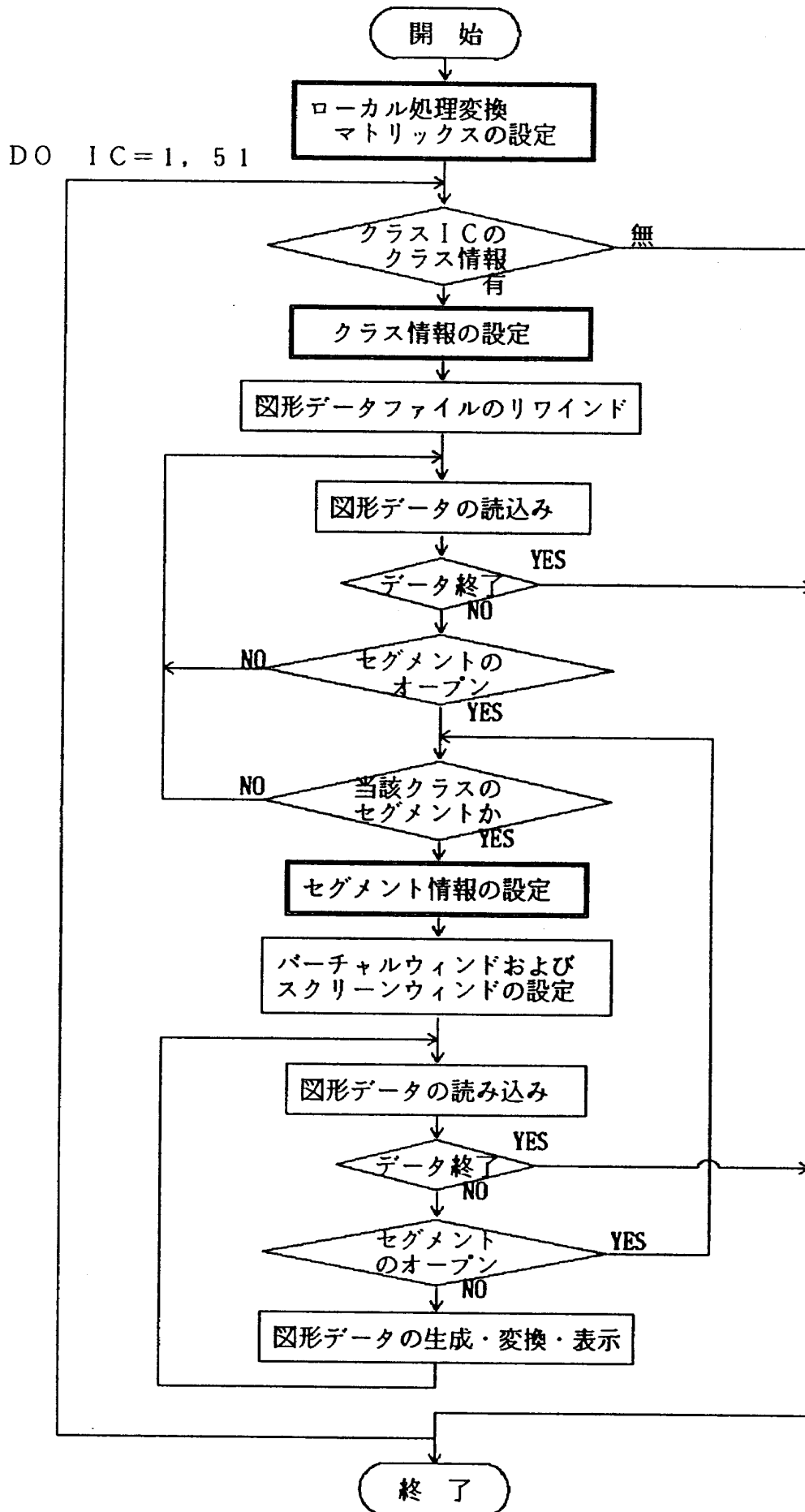


図 3.2 図形データ表示処理の流れ

めの機能をソフトウェアで実現しなければならない。

2次元GDの場合、図形データは作成した図形データのどの範囲を表示するかを示すバーチャルウィンドの指定と、画面上のどの位置に表示するかを示すスクリーンウィンドの指定に従って表示できる。そのため、ローカル処理機能の中の移動、拡大・縮小処理はバーチャルウィンドを変更するだけでよく、図形データそのものを変換する必要はない。しかし、回転の場合には指定に従って変換マトリックスを作成し、表示用に生成された図形データを変換する必要がある。この処理を行うのがローカル処理変換プログラムである。なお、図形データの回転は表示画面上の軸を中心としてではなく図形データの軸を中心に行うため、指定されたデータをすべて格納しておく必要がある。GCPではそのために、50情報が格納できる領域を用意している。

ローカル処理変換プログラムは、図形データの回転軸および回転角度が設定された場合にはローカル処理情報としてそれらを順次格納し、それにもとずきローカル処理変換マトリックスを作成する。生成された図形データの座標はこのマトリックスで図形変換することによって指定されたローカル処理を施すことができるが、マトリックスが設定されていない場合には変換されない。

(2) クラス変換プログラム

作図プログラムで行うクラスに対する回転、移動、拡大・縮小および当該クラスを構成するセグメント番号の指定にはCLASS 2サブルーチンまたは

CLASS 3サブルーチンが使用できる。指定されたデータはクラス毎に図3.3に示すクラス属性テーブルに格納され、図形データの生成時に当該クラスに属する全セグメントに対し有効となるように処理されなければならない。2次元GD表示プログラムにおいて、クラス化処理を行うためのプログラムがクラス変換プログラムである。

クラス情報変換プログラムは、クラス情報の設定が行われると、その定義に従って図形データがX軸まわり、Y軸まわり、Z軸まわりの順に回転し、X方向、Y方向、Z方向の順に移動し、指定された拡大率で拡大されるようなクラス変換マトリックスを作成する。

表示用に生成された図形データの座標は、当プログラムで図形変換することによって、指定されたクラス変換処理を施すことができるが、セグメント情報が設定されていない場合には変換されない。

(3) セグメント変換プログラム

作図プログラムで行うセグメントに対する回転、移動、拡大・縮小の指定には、SGMTR 2サブルーチンまたはSGMTR 3サブルーチンが使用できる。指定されたデータはセグメント毎に図3.4に示すセグメント属性テーブルに格納され、表示用図形データの生成時に指定セグメントの全データに対し有効となるように処理されなければならない。2次元GD表示プログラムにおいて、セグメント処理を行うためのプログラムがセグメント変換プログラムである。

1	X軸まわりの回転角	
2	Y軸まわりの回転角	
3	Z軸まわりの回転角	
4	X軸方向の移動量	
5	Y軸方向の移動量	
6	Z軸方向の移動量	
7	各方向の拡大率	
8	セグメント数	セグメント番号 1
9	セグメント番号 2	セグメント番号 3
⋮	⋮	⋮
⋮	セグメント番号50	(空き)

当該クラスに属するセグメント情報

図 3.3 クラス属性テーブルの形式

1	X軸まわりの回転角
2	Y軸まわりの回転角
3	Z軸まわりの回転角
4	X軸方向の移動量
5	Y軸方向の移動量
6	Z軸方向の移動量
7	X軸方向の拡大率
8	Y軸方向の拡大率
9	Z軸方向の拡大率

図 3.4 セグメント属性テーブルの形式

セグメント情報変換プログラムは、セグメント情報の設定が行われると、指定に従って図形データが X 軸まわり、Y 軸まわり、Z 軸まわりの順に回転し、X 方向、Y 方向、Z 方向の順に移動し、X 方向、Y 方向、Z 方向の拡大率にもどずき拡大されるようなセグメント変換マトリックスを作成する。

表示用に生成された図形データの座標は、当プログラムで図形変換することによって指定されたセグメント変換処理を施すことができるが、セグメント情報が設定されていない場合には変換されない。

3.3.2 図形データ表示処理の流れ

(1) ローカル処理変換マトリックスの設定

2次元GD表示時のローカル処理で、回転の指定または画面初期化の指定が行われた場合には、その指定に従ってローカル処理変換マトリックスの設定または初期化を行う。これらの処理は図形データの読み込みに先立って行っておく必要がある。

(2) クラス情報の設定

クラス番号 1 から 51 のクラスに対し、クラス情報が指定されているか否かを調べ、指定されている場合にのみクラス情報を設定し、以下の処理を行う。なお、GCP で取り扱えるクラスは 50 であるが、便宜上 51 番目のクラスを用意し、どのクラスにも属さないセグメントを処理する際に使用している。

(3) 図形データファイルのリワインド

2次元GD表示プログラムでは、ファイルから読み込んだデータを順次処理するのではなく、クラス毎にそのクラスに属するセグメントデータを読み込み、処理している。そのため、図形データファイルを最大 51 回もリワインドし、データを繰り返し読み

込むことになるが、これは 1 つのセグメントが複数のクラスに属する場合を想定しているためである。

(4) 当該セグメントのオープン情報の読み込み

図形データはセグメント単位に格納されているため、当該クラスに属するセグメントのオープン情報が見つかるまで図形データを順次読み込む。当該クラスに属するセグメントのオープン情報が見つかったならば、以下の処理を行う。また、この処理中に読み込まべき図形データがなくなれば、その時点で当該クラスに対する処理は終了する。

(5) セグメント情報の設定

セグメントデータの読み込みに先立ってセグメント属性テーブルを参照し、セグメント情報の指定が行われている場合にはセグメント変換プログラムを呼びだし、セグメント情報の設定を行う。

(6) バーチャルウィンドおよびスクリーンウィンドの設定

セグメントデータを表示する際のバーチャルウィンドおよびスクリーンウィンドを設定する。

セグメントデータを表示する際のバーチャルウィンド、スクリーンウィンドの初期値は、図 3.5 に示すセグメントウィンドテーブルの値に基づいて設定する。このテーブルは作図プログラムで SVIW2 サブルーチンまたは SVIW3 サブルーチンを使用してそれぞれのウィンドを指定したときにそのデータが格納されるが、それらの値を陽に指定していないセグメントのデータを表示する際の初期値は以下のようにして設定する。

バーチャルウィンドは図 3.6 のバーチャルウィン

1	ウィンドセットの有無
2	バーチャルウィンド左下前 X 座標
3	バーチャルウィンド左下前 Y 座標
4	バーチャルウィンド左下前 Z 座標
5	バーチャルウィンド右上後 X 座標
6	バーチャルウィンド右上後 Y 座標
7	バーチャルウィンド右上後 Z 座標
8	スクリーンウィンド左下前 X 座標
9	スクリーンウィンド左下前 Y 座標
10	スクリーンウィンド左下前 Z 座標
11	スクリーンウィンド右上後 X 座標
12	スクリーンウィンド右上後 Y 座標
13	スクリーンウィンド右上後 Z 座標

図 3.5 セグメントのウィンドテーブルの形式

1	図形データの拡大率
2	バーチャルウィンド 左下前 X座標
3	バーチャルウィンド 左下前 Y座標
4	バーチャルウィンド 左下前 Z座標
5	バーチャルウィンド 右上後 X座標
6	バーチャルウィンド 右上後 Y座標
7	バーチャルウィンド 右上後 Z座標

図 3.6 バーチャルウィンドテーブルの形式

ドテーブルを参照し、バーチャルウィンドの各座標を拡大率で割った値を設定する。このテーブルは、作図プログラムでFACTサブルーチンを使用して拡大率を指定したとき、VWIND2サブルーチンまたはVWIND3サブルーチンを使用してバーチャルウィンドを指定したときにそのデータが格納される。なお、拡大率が設定されていない場合には1.0が指定されたものとみなし、バーチャルウィンドが設定されていない場合にはその左下座標および右上座標は(0.0, 0.0)および(40.0, 30.0)が指定されたものとみなし処理をする。また、スクリーンウィンドは第3.1項で設定した値を用いる。

なお、2次元GDの場合、図を表示する際のスクリーンウィンドとバーチャルウィンドの縦横比が異なると図がゆがんでしまうため、2次元GD表示プログラムでは、GSPライブラリ²⁾に対する設定を行う前にそれぞれの縦横比が同じになるようにバーチャルウィンドの縦または横を拡張している。

(7) 図形データの生成・変換・表示処理

2次元GD表示プログラムでは、GCP図形データファイルに格納されているセグメントデータを順次読み込んで、2次元表示データの生成・変換・表示を行っている。この処理については少し複雑であるため項を改めて述べることにする。なお、この処理はセグメントのオープン情報を読み込んだときに(4)項の処理に戻り、読み込むべき図形データがなくなったときに当該クラスに対する処理を終了する。

また、現在設置されている大多数の2次元GDは、図形データを格納するためのセグメントバッファが32KBしかない。この量は線分データ2500本を格納できる程度の大きさであり、10×10×10の格子点をもつ3次元格子図でさえも格納することができない。そのため、図形データの表示を行う際には常にセグメン

トバッファに対する図形データの格納、画面への表示、セグメントバッファのクリアという過程を繰り返し行わざるを得ない。そこで、図形データ生成・変換時には常にセグメントバッファに格納されているデータ量を調べ、適時画面への表示とセグメントバッファのクリアを行っている。

3.3.3 図形データの生成・変換・表示処理の流れ

GCPでは、図形の形状を表すもっとも基本的な要素である線、多角形、文字等をプリミティブと呼び、プリミティブに対する属性をプリミティブ属性と呼ぶ。現在、2次元表示用に表3.1に示す21種類のプリミティブ出力用サブルーチンと、表3.2に示す3種類のプリミティブ属性出力用のサブルーチン、2種類のプリミティブ属性定義用サブルーチンが用意されている。表には明記されていないが、この他にカルコン形式のサブルーチンやGCPの汎用プログラムを利用することによっても、間接的にこれらのサブルーチンが呼び出される。

なお、プリミティブ出力用サブルーチンおよびプリミティブ属性出力用サブルーチンによって作成された図形データは表3.3の形式で図形データファイルに格納され、プリミティブ属性定義用サブルーチンによって定義されたカラー情報は色毎にカラーテーブルに格納される。プリミティブ属性出力用サブルーチンはそれ以降に作成されるプリミティブに対して有効となるが、そのうちの線画の色番号に対応する色はカラーテーブルに従って決定され、線番号に対応する線の種類は装置ごとに自動的に決定される。

本項では、GCP形式の図形データファイルから取り出したデータから2次元表示用の図形データが生成されるまでの過程について説明する。

(1) プリミティブ属性データの処理

図形データファイルにはプリミティブデータおよびプリミティブ属性データが格納されているが、2次元表示プログラムでは、図形データファイルから取り出した図形データがプリミティブ属性データであった場合には、以上のように処理する。

a. 線画の色の処理

線画の色番号が指定された場合には、それ以降に発生する線画に対しその色番号を有効としなければ

表 3.1 プリミティブ出力用サブルーチン

NO.	サブルーチン名	機 能
1 2 3	GTEXT GTEXT1 GTEXT2	カルコンプ形式の文字列を描く。 線画文字列を描く。 中塗り文字列を描く。
4 5 6 7 8 9 10 11 12 13 14 15 16	GARC GCIRCL GELLIP GFAN GRECT LINE2D PLYG2D PLOT2D PCIRCL PELLIP PFAN PRECT PPLY2D	円弧を描く。 円を描く。 楕円を描く。 扇形を描く。 長方形を描く。 2次元の線分列を引く。 2次元の多角形を描く。 2次元で指定位置まで移動する。 円を描き、その内側を中塗りする。 楕円を描き、その内側を中塗りする。 扇形を描き、その内側を中塗りする。 長方形を描き、その内側を中塗りする。 2次元の多角形を描き、その内側を中塗りする。
17 18 19 20 21	GCUBIC LINE3D PLYG3D PLOT3D PPLY3D	直方体を描く。 3次元の線分列を引く。 3次元の多角形を描く。 3次元で指定位置まで移動する。 3次元の多角形を描き、その内側を中塗りする。

表 3.2 プリミティブ属性出力用・設定用サブルーチン

NO.	サブルーチン名	機 能
1 2 3	NEWPEN LTYPE EEFPLN	線画を描く時の色を色番号で指定する。 線画を描く時の線種を指定する。 3次元表示時に2次元図形の出力面を定義する。
4 5	PALLET PALETX	線画および面画を描く時の色番号に対応する色を設定する。 線画および面画を描く時の色を設定する。

ならない。ここで線画の色の処理について説明する前に、カラーテーブルについて簡単に説明しておく。

作図プログラムで行うプリミティブ属性の色番号に対する色の指定は、プリミティブ属性定義用サブルーチンであるPALLETサブルーチンまたはPALETXサブルーチンで行える。色はR(赤), G(緑), B(青)の3原色の輝度を、もっとも低い0.0からもっとも高い1.0の範囲で指定できるが、指定された値はそれぞれ0~255の整数値に変換され、RGB毎に8ビットのデータとしてカラーテーブルに格納される。

ただし、GCPのカラーテーブルは1~8の色番号に限り、以下に示す省略値をもつ。

色番号1: R(1.0), G(1.0), B(1.0) [白]

色番号2: R(0.0), G(0.0), B(1.0) [青]

色番号3: R(1.0), G(0.0), B(0.0) [赤]

色番号4: R(0.0), G(1.0), B(0.0) [緑]

色番号5: R(1.0), G(1.0), B(0.0) [黄]

色番号6: R(1.0), G(0.0), B(1.0)

[赤紫]

色番号7: R(0.0), G(1.0), B(1.0)

[青紫]

色番号8: R(0.0), G(0.0), B(0.0) [黒]

3次元GD表示プログラムでは、線画の色番号の指定があった場合には、カラーテーブルの中の指定された色番号の情報からR, G, B 3原色の輝度を決定し、その色で表示されるように処理をするが、3次元GDは各色とも256段階で表示可能なため、カラーテーブルに格納されたデータは8ビットすべて有効となる。

表 3.3 図形データの格納形式

関連するサブルーチン		語数	識別番号	デ - タ
制御データ	PLOTS	2	-1	ISNO
プリミティブ データ	GARC	6	1	X, Y, R, TH1, TH2
	GCIRCL	4	2	X, Y, R
	GCUBIC	7	3	XL, YL, ZL, XR, YR, ZR
	GELLIP	5	4	X, Y, A, B, TH
	GFAN	6	5	X, Y, R, TH1, TH2
	GRECT	5	6	XL, YL, XR, YR
	LINE2D	2 N+1	7	(X(I), I=1, N), (Y(I), I=1, N)
	LINE3D	3 N+1	8	(X(I), I=1, N), (Y(I), I=1, N), (Z(I), I=1, N)
	PLYG2D	2 N+1	9	(X(I), I=1, N), (Y(I), I=1, N)
	PLYG3D	3 N+1	10	(X(I), I=1, N), (Y(I), I=1, N), (Z(I), I=1, N)
	PLOT2D	4	11	X, Y, IP
	PLOT3D	5	12	X, Y, Z, IP
	PCIRCL	7	16	X, Y, R, IPAT, ICOL, IFCOL
	PELLIP	9	17	X, Y, A, B, TH, IPAT, ICOL, IFCOL
	PFAN	9	18	X, Y, R, TH1, TH2, IPAT, ICOL, IFCOL
PRECT	8	19	XL, YL, XR, YR, IPAT, ICOL, IFCOL	
PPLY2D	2 N+4	20	(X(I), I=1, N), (Y(I), I=1, N), IPAT, ICOL, IFCOL	
PPLY3D	3 N+4	21	(X(I), I=1, N), (Y(I), I=1, N), (Z(I), I=1, N), IPAT, ICOL, IFCOL	
NUMBER	N1+6	28	X, Y, HI, (NBCD(I), I=1, N1), TH, N	
GTEXT	N1+6	29	X, Y, HI, (NBCD(I), I=1, N1), TH, N	
GTEXT1	N1+6	27	X, Y, HI, (NBCD(I), I=1, N1), TH, N	
GTEXT2	N1+9	26	X, Y, MI, (NBCD(I), I=1, N1), TH, N, IPAT, ICOL, IFCOL	
プリミティブ 属性データ 等	NEWPEN	2	1000	IPEN
	LTYPE	2	1001	LNO
	EEFPLN	7	1002	RX, RY, RZ, TX, TY, TZ

しかし、2次元GDの場合にはバックカラーを含めて8色でしか表示できないので、図3.7に示すように8ビットのうち先頭1ビットにより色を決定する。すなわち、先頭の1ビットがONの場合には1.0, OFFの場合には0.0とし、それに対応する1から8の色番号が指定されたものとして表示されるように、2次元出力ライブラリGSPの色の設定プログラムを呼び出す。なお、1から8の色番号に対応する色およびメッセージの色は、2次元GD表示プログラムのメニューにより任意に変更することができる。

b. 線種の処理

図形データの中で線種の指定が行われていた場合には、それ以降に発生する線画に対しその線番号を有効としなければならない。

3次元GDには8タイプの線があり、太さも2段階あるので、16種類の線を描くことができる。しかし、2次元GDには線の太さは2段階あるものの、4タイプの線しかないため、8種類の線しか描けない。

そこで、1～8の線番号が指定された場合には、線画は以下に示す線種で描かれるように2次元出力ライブラリGSPの線種の設定プログラムを呼び出す。また、9以上の線番号が指定された場合には、その番号を8で割ったときの剰余(0の場合には8)が指定されたときとみなして処理をし、線種が設定されていない場合には線番号1が設定されたものとみなして処理をする。

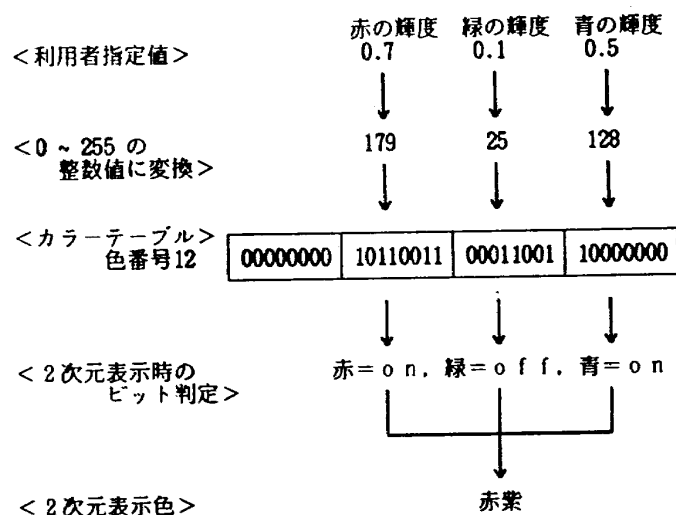


図 3.7 カラーテーブルのセットと参照例

- 線番号 1 : 実線
- 線番号 2 : 点線
- 線番号 3 : 破線
- 線番号 4 : 一点鎖線
- 線番号 5 : 実線 (太)
- 線番号 6 : 点線 (太)
- 線番号 7 : 破線 (太)
- 線番号 8 : 一点鎖線 (太)

c. 2次元データ出力面の設定

3次元ローカル処理を可能にするためには、図形データは表示直前まで3次元データとして取り扱う必要がある。したがって、2次元GD用表示プログラムでは、GCP形式で作成されている図形データが円、文字のような2次元データの場合には、すべて2次元データ出力面の指定に従って3次元化処理を行わなければならない。この処理を行うのが3次元変換プログラムである。

3次元変換プログラムは、図形データの中から2次元データ出力面を設定するプリミティブデータを読み込むたびに呼び出され、作図プログラムの指定に従って、2次元データがX軸まわり、Y軸まわり、Z軸まわりの順に回転した後、X方向、Y方向、Z方向の順に移動したデータに変換できるように、3次元変換マトリックスを作成する。生成された図形データの座標は当プログラムで図形変換することによって、3次元変換処理を施すことができるが、2次元データ出力面の設定なしに変換処理をおこなった場合には、原点を通るX-Y平面上に出力される。

(2) プリミティブデータの処理

2次元GD表示プログラムにおける図形データの生成は、STRANサブルーチンで行われる。STRANサブルーチンは、図形データファイルから取り出したプリミティブデータから線分データ、線分列データ、多角形データ、中塗り多角形データの中のいずれかの図形データを生成する。

生成された図形データが2次元データの場合には、前項の3次元変換プログラムで3次元データに変換し、すべてのデータを上記4種類の3次元データとした上で、セグメント変換、クラス変換、ローカル処理変換の順序で変換処理して、DROWサブルーチンに渡す。

DROWサブルーチンでは、上記4種類の図形データを表示するために、2次元GD用出力ライブラリGSP²⁾を適時呼び出す。したがって、出力装置を変更する場合でも、DROWサブルーチンを使用する装置用の図形出力ライブラリを使用したものに変更するだけでよい構造になっている。なお、2次元GDでは3次元データは扱えないので、GSPライブラリのサブルーチンを呼び出す時点で図形データのZ座標が捨てられる。

3.4 追加データの表示とインタフェースプログラムの呼び出し

第3.1項で述べたように、IST、IOIL、IMM変数のいずれかが1であった場合には、流線プログラム、オイルフロープログラム、または文字データプログラムのいずれかによって、GCP図形データに対する追加データが別ファイルに作成されているので、前項とまったく同様の方法で図形データの生成・表示処理を行わなければならない。なお、表示後、図3.1に示すように、関連したインターフェースプログラムが呼び出され、GCP図形データファイルへの追加データの保存や新たな追加データの作成を行うことができる(第5.1項を参照のこと)。なお、インターフェースプログラムの処理結果によって先の変数の値は適時変更される。この処理終了後、第3.2項へ戻る。

3.5 メニュー処理

2次元表示プログラムには表3.4に示す18のメニューが用意されているが、それぞれのメニューは以下のように処理される(図3.1参照)。

a. 拡大処理

メニューで拡大処理が選択された場合には、どの部分を拡大したいのかをユーザに問い合わせ、拡大部分の左下と右上、または左上と右下の2点の座標を読み取る必要がある。2次元GDには、画面上の座標を読み取るために十字カーソルが用意されている。十字カーソルが表示されているときには、矢印キーまたはジョイスティックで十字カーソルの移動が行えるので、ユーザは指定したい座標位置でコマンド終結キーを押すことによって座標をプログラム

表 3.4 2次元GD表示プログラムで利用可能なメニュー一覧(つづく)

コマンド	内容	説明
Z	図の拡大	十字カーソルが2度表示されるので、ジョイスティックまたは矢印キーを使用して拡大したい部分の左下と右上、または左上と右下の2点の座標を入力する。指定された座標値は図の縦横比が変化しないような座標値に直された上でその部分が画面全体に拡大表示される。
T	図の移動	十字カーソルが2度表示されるので、ジョイスティックまたは矢印キーを使用して2点の座標を入力する。最初に指定された座標が2番目に指定された位置へ移動するようにバーチャルウィンドが変更される。
F	特定位置への表示 (縮小移動)	十字カーソルが2度表示されるので、ジョイスティックまたは矢印キーを使用して画面に表示中の図形データの移動先位置の左下と右上、または左上と右下の2点の座標を入力する。表示中のデータは指定された位置に図の縦横比が変化しないように縮小表示される。
R X	X軸を中心とした 回転角の指定	画面に表示中の図の回転角度(右手系)を"R X 45."というように1文字分のブランクをあけて実数形式で指定する。図形データはデータのX軸を中心に指定角度だけ回転する。
R Y	Y軸を中心とした 回転角の指定	画面に表示中の図の回転角度(右手系)を"R Y 45."というように1文字分のブランクをあけて実数形式で指定する。図形データはデータのY軸を中心に指定角度だけ回転する。
R Z	Z軸を中心とした 回転角の指定	画面に表示中の図の回転角度(右手系)を"R Z 45."というように1文字分のブランクをあけて実数形式で指定する。図形データはデータのZ軸を中心に指定角度だけ回転する。
R R	回転角の複数指定	R X、R Y、R Zコマンドで行う図の回転指定をまとめて行う。Eコマンドを投入するまで繰り返しR X、R Y、R Zコマンドが投入できる。
D	初期画面の表示	Z、T、F、R X、R Y、R Z、R Rコマンドにより拡大、回転等が行われた図を初期状態に戻す。
C	表示色の変更	カラー設定画面になるのでメッセージの色、ペン番号に対応する色およびバックカラーの色を指定する。これ以降の図はこの指定に従って表示される。
X Y	X Y、P P出力の 指定	プログラム実行時の最初のX Yコマンド投入時には、バッチジョブの場合と同様にX Y出力か、P P出力か、P PおよびX Y出力かの選択およびX Y出力時の用紙等のセット番号の指定が行える。プログラム実行中にこのようにして作成した図形データは1つのファイルに格納され、開設中のセッションのジョブ番号で管理される。
H C	N L P出力の指定	表示中の図形データをN L Pに出力する。F O R T 7 7コマンド、R U Nコマンド等により作成されたN L P出力用図形データは開設中のセッションのラインプリンタ出力結果として格納される。
O P R A	O P R出力の指定	表示中の図形データを極超音速風洞建屋のO P Rへ出力する。
O P R F	O P R出力の指定	表示中の図形データを原動機1号館のO P Rへ出力する。
S T	流線の開始点の 設定	流線の色、開始点の座標を設定する。ただし、汎用プログラムの流線サブルーチンを使用している場合に限る。その後作成した流線データを図形データに追加するか否かの指定ができる。
O	オイルフロー開始 点の設定	オイルフローの色、開始点の座標を設定する。ただし、汎用プログラムのオイルフローサブルーチンを使用している場合に限る。その後作成したオイルフローデータを図形データに追加するか否かの指定ができる。

表 3.4 2次元GD表示プログラムで利用可能なメニュー一覧(つづき)

コマンド	内容	説明
M	文字列入力指定	文字列の色番号(1~8)および文字列の入力待ちとなるので、"5 ABC"のように入力する。次に十字カーソルが2度表示されるのでジョイスティックまたは矢印キーを使用して文字の左下座標と右下座標位置を指定する。文字列はこの2点上に納まるような大きさ(最大11.9)で表示される。その後文字データを図形データに追加するか否かの指定ができる。文字列を図形データに追加した場合には繰り返しこの処理が行え、文字列をブランクにした時に処理は終了する。なお、この文字データはRXコマンド等により回転することはできない。
H	コマンド種類の表示	ローカル処理時に使用できるコマンドの種類および簡単な投入例を表示する。この表示画面のときでもローカル処理用のコマンドは投入できる。
E	ローカル処理終了	ローカル処理が終了し、ユーザプログラムへ戻る。

に通知できる。

そこで、拡大処理では十字カーソルを2度表示し、ユーザから拡大部分の左下と右上または左上と右下座標が通知されるのを待つ。拡大部分が通知されたならば、その部分が画面全体に表示されるようにバーチャルウィンドを変更し、第3.2項の画面の初期化処理へ戻る。

b. 移動処理

メニューで移動処理が選択された場合には、どこからどこへ移動したいのかをユーザに問い合わせる必要がある。そこで、十字カーソルを2度表示し、ユーザから移動元座標と移動先座標が通知されるのを待つ。2点の座標が通知されたならば、1点目の位置のデータが2点目の位置へ移動するように表示データのバーチャルウィンドを変更し、第3.2項の画面の初期化処理へ戻る。

c. 縮小移動処理

縮小移動のメニューが選択された場合には、現在画面上に表示中のデータを、画面上のどの位置へ表示したいのかをユーザに問い合わせる必要がある。そこで、十字カーソルを2度表示して、ユーザから表示位置の左下と右上または左上と右下の画面上の2点の座標が通知されるのを待つ。表示位置が通知されたならば、現在画面全体に表示されている図形データがその位置へ表示されるようにバーチャルウィンドを変更して、第3.2項の処理へ戻る。

d. 回転処理

X軸回りの回転処理、Y軸回りの回転処理、Z軸

回りの回転処理のメニューが選択された場合には、ローカル処理変換プログラムを呼び出してローカル処理変換マトリックスを変更する。そして、第3.2項の処理へ戻る。また、回転の複数指定のメニューが選択された場合には、指定が終了するまで順次回転軸および回転角の情報を格納し、その後ローカル処理変換マトリックスを更新し、第3.2項の処理へ戻る。

e. 初期画面表示処理

初期画面表示処理のメニューが選択された場合には、ローカル処理変換マトリックスを初期化し、バーチャルウィンドを初期状態に戻した上で、第3.2項の処理へ戻る。

f. 表示色変更処理

表示色変更処理のメニューが選択された場合には、カラー設定画面を表示し、メッセージの色およびペン番号1から8に対応した色をそれぞれ指定できるようにする。これ以降の図形はこの指定に従って表示されるように、指定された情報を色変換テーブルに格納して、第3.2項の処理へ戻る。

g. 他装置への出力

表示中の図形データをXY, NLP, OPRに出力するメニューが選択された場合には、データ変換に必要な各種テーブル、装置に応じたシザリング情報をファイルに格納し、第5章に述べる図形データ形式変換プログラムを呼び出す。そして、第3.2項の処理へ戻る。

h. 流線、オイルフロー開始点の入力処理

流線、オイルフロー開始点の入力処理メニューが選択された場合には、それぞれのインタフェースプログラムを呼び出し、追加データの作成を行う。その後、第3.2項の処理へ戻る。

i. 文字入力処理

文字入力処理メニューが選択された場合には、文字データインタフェースプログラムを呼び出し、文字データの追加作成を行う。その後、第3.2項の処理へ戻る。

j. メニュー表示処理

メニュー表示が選択された場合には、2次元GD表示プログラムで使用できるメニューの簡単な説明および使用例を表示し、再度メニューが選択されるのを持つ。メニューが選択されたならばその処理を行うが、メニューが選択されずにENTERキーが押された場合には第3.2項の処理へ戻る。

k. ローカル処理の終了処理

ローカル処理終了メニューが選択された場合には、2次元GD表示プログラムの処理を終了させる。

4. 新モニタプログラム

新モニタプログラムは、バッチ処理等により作成されて中間データファイルに格納されているPP、XY出力用のデータを入力して、2次元GDに図形表示を行うプログラムである。初期のモニタプログラムに対するこのプログラムの最大の特徴は、ローカル処理機能の大幅改良、ページ概念の導入および中塗りデータ表示機能の組み込みにある。

一番目のローカル処理機能は、表示中の図形データに対する拡大、回転等を行うためのものであるが、2次元GD表示プログラムで使用しているものうち、新モニタプログラムでも使用できるものをすべて組み込み、さらに、任意のファクタで図形データを表示するファクタ指定機能を追加することによって大幅な機能改良を達成した。このファクタ指定機能は、モニタ処理では縦横比を変更せずにできるだけ大きくなるように図形データを表示しているために、同じファクタで表示して比較・検討すべき図でも、タイトル等を変更しただけでファクタが変わってしまうことがあるために、利用者からの要望に答えて用意した機能である。

二番目のページ概念は、X方向、Y方向共に出力範囲の制限を持つNLP、OPRが新たに設置されたことに伴って必要となった機能である。新モニタプログラムでもページ単位の処理を行えるようにし、表示するページ番号の指定、スキップするページ数の指定等、ページ制御用のメニューも種々用意している。また、第5章に述べるNLP出力用の形式変換プログラムでも、作図プログラムから出力される一連の図形データは意味のあるデータのまとまり、すなわちページを単位として1枚の図になるように作成している。

最後の中塗り機能は、利用者の要請によって組込んだ機能である。初期のモニタプログラムはXY、PP出力に先立って、バッチジョブで作成した図形データを確認するだけの目的で使用されていたが、徐々にTSSに表示するための目的で使用されることも多くなり、その結果として要請された機能である。

なお、ページ機能と中塗り機能を取り入れるためには、その情報をなんらかの形で中間データファイルに格納しなければならない。中間データの形式を変更することなく、新たな情報を格納する場合には、そのデータをXYおよびPPに出力した場合でも支障を来さないようにしなければならない。

これらの機能を盛り込んだ新モニタプログラムは図4.1に示す流れで処理を行っている。図中の2重枠で囲まれているインタフェースプログラムと図形データ形式変換プログラムは、2次元GD表示プログラムで使用しているものと同一のものである。

本項では、新モニタプログラムの入力となる中間データファイルの形式について説明し、その後図4.1の処理の流れに沿って、新モニタプログラムの機能説明を行う。

4.1 中間データファイルの形式

NSシステムの間接データファイルは、バッチ処理で作成されたXY、PP出力用の図形データを格納するためのファイルであるが、XY、PP出力時に使用するだけでなく、TSSのモニタ処理時にも使用できるようにするために、航技研独自の形式で作成されている。中間データファイルは、レコード

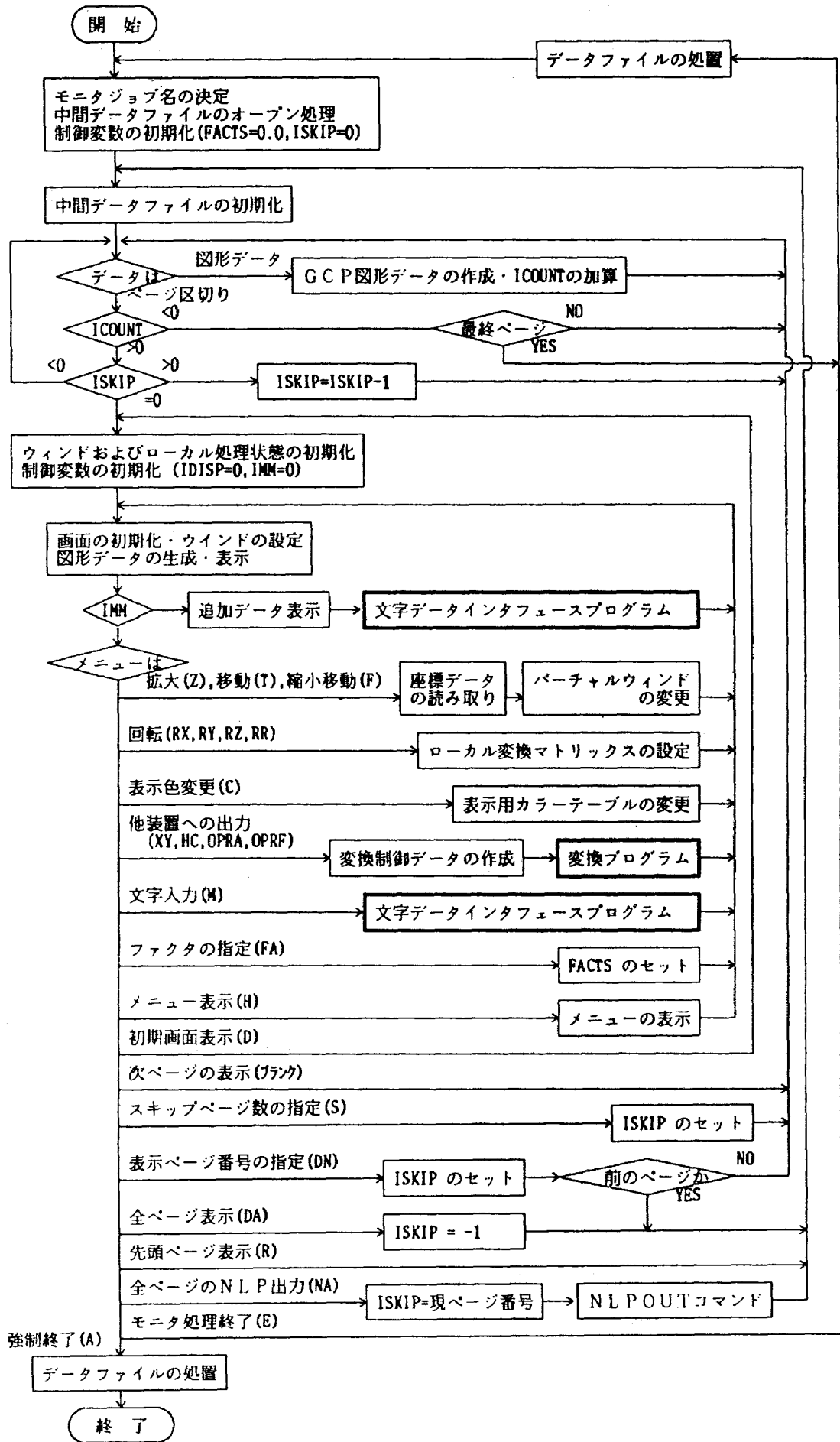


図 4.1 モニタ処理の流れ

長480バイトの固定長ファイルであるが、ブロック長は使用しているディスクの使用効率を考慮して、現システムでは23040バイトで作成している。このデータファイルには初期ブロック、データブロック終了ブロックと呼ばれる3種類のデータが格納されるが、読み込んだデータがどの種類のものであるかは、各レコードの先頭1語に格納されているブロック識別子で判定することができる。

以下に、これらのブロックに格納されるデータの種類および内容について説明する。

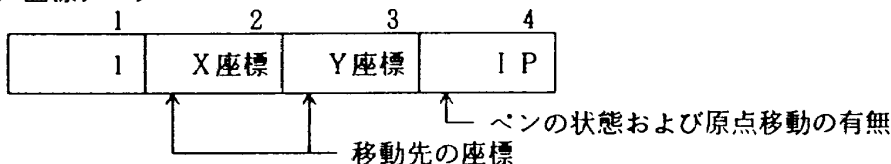
4.1.1 初期ブロック

初期ブロックのブロック識別子は ' & P L S ' であり、このブロックには図形データファイル作成時の情報として、図形データの作成日付等が格納される。

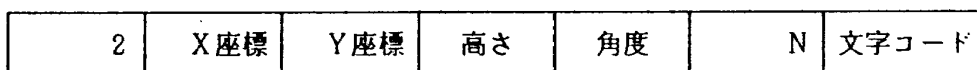
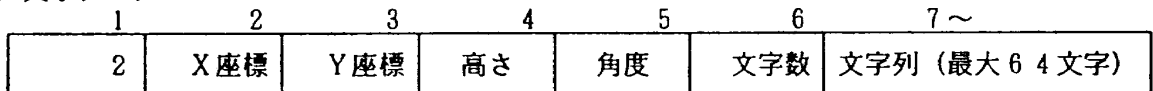
初期ブロックだけの図形データファイルは、ファイルのオープン処理は正常に終了しているが、プログラム実行時に図形データが出力されなかったことを意味する。

4.1.2 データブロック

(1) 直線データ



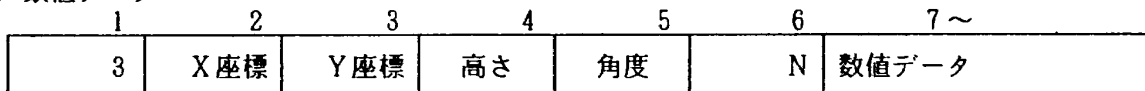
(2) 文字データ



文字左下の座標

- 1 : 開始位置までペンアップで移動
- 2 : 開始位置までペンダウンで移動

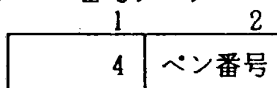
(3) 数値データ



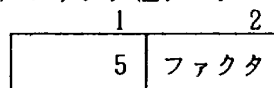
数値データ左下の座標

語数+1000

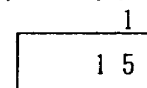
(4) ペン番号データ



(5) ファクタ値データ



(6) ブロックエンドデータ



カルコンプライブラリで作成した図形データは、すべてベーシックソフトウェアを介して直線データ、文字データ、数値データ、ペン番号データ、ファクタ値データとなって、データブロックに格納される。

データブロックのブロック識別子は ' & D T A ' であり、2語目以降に図4.2に示す7種類のデータが連続して格納される。それぞれのデータの種類の、データの先頭1語に整数値で格納されているデータ識別子により識別することができる。中間データファイルに格納されているデータブロックの数は、作図プログラムで出力するデータ量に応じて異なる。

(1) 直線データ

データの1語目に格納されているデータ識別子が1のデータは、カルコンプライブラリのPLOTサブルーチンで作成された直線データである。この場合には、サブルーチンの引数で指定された移動先のX座標およびY座標、移動時のペンの状態（ペンアップかペンダウンか）の情報が2語目から4語目に格納される。

(2) 文字データ

データの1語目に格納されているデータ識別子が2のデータは、カルコンプライブラリのSYMBOL

図4.2 中間データの格納形式(一般データ)

サブルーチンで作成された文字データである。この場合には、2語目から6語目には引数で指定された文字データ左下のX座標およびY座標、文字の高さ、文字の角度、文字数が、7語目から最大22語目には文字列が格納される。なお、SYMBOLサブルーチンで64文字以上の文字列が指定された場合には、複数の文字データに分割して格納される。

また、SYMBOLサブルーチンで文字コードが指定された場合には、開始座標までペンアップで移動するかペンダウンで移動するかによって、文字数には-1または-2が格納され、7語目に文字コードが格納される。

(3) 数値データ

データの1語目に格納されているデータ識別子が3のデータは、カルコンプライブラリのNUMBERサブルーチンで作成された数値データである。この場合には、2語目から6語目には引数で指定された数値データ左下のX座標およびY座標、文字の高さ、文字の角度、数値データ語数に1000を加えた数が、7語目以降には数値データがEBCDICの内部コードで格納される。

(4) ペン番号データ

データの1語目に格納されているデータ識別子が4のデータは、カルコンプライブラリのNEWPENサブルーチンで作成されたペン番号データである。この場合には、2語目にペン番号が格納される。

(5) ファクタ値データ

データの1語目に格納されているデータ識別子が5のデータは、カルコンプライブラリのFACTORサブルーチンで作成されたファクタ値データである。この場合には、2語目にファクタ値が格納される。

(6) ブロックエンドデータ

1語目に格納されているデータ識別子が15のデータは、処理中のデータブロック内の作図データの終了を示すブロックエンドデータである。

(7) エラー情報

エラーの情報は各データブロックの113語目以降に格納される。

4.1.3. 終了ブロック

終了ブロックの識別子は '&PLE' であり、こ

のブロックには図形データの終了日付等が格納される。

終了ブロックがない図形データファイルはファイルのクローズド処理が行われなかったことを意味する。

4.1.4 その他

本項では、NSシステムになって追加した中塗り情報およびページ情報の格納形式について説明する。

中間データファイルでは中塗りの円、扇形、多角形情報およびページ情報の格納には、主として図形表示を伴わないペン番号データを使用することにした。そして、通常のペン番号データと区別するために、ペン番号データの先頭8ビットを識別情報として使用している。図4.3に追加データの格納形式を示す。

(1) 中塗りの円情報

中塗りの円を作成するためのPCIRCLサブルーチンでは、中心のX座標とY座標、半径、中塗りパターン、中塗りの色、枠の色が指定できる。PCIRCLサブルーチンが呼ばれたときには、以下のようにして図形データを中間データファイルに格納している。

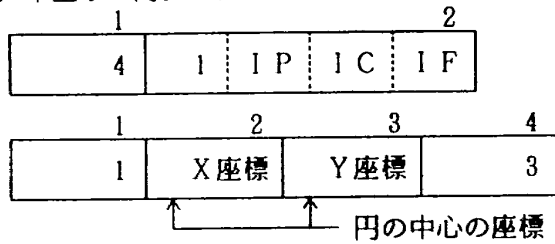
中間データファイルでは、中塗りの円情報を示す識別情報を1としている。そこで、中塗りの円情報を示す識別情報および指定された引数のうちの中塗りパターン、中塗りの色、枠の色の情報を、ペン番号データの1語をそれぞれ8ビットづつに分割して格納している。また、中心のX座標とY座標および半径は、ペンアップで移動する2つの直線データとして格納している。

この処理により、XY、PP出力時のペンの色に変更されてはならないので、最後にそれまで使用されていたペン番号で通常のペン番号データを発行している。

(2) 中塗りの扇形情報

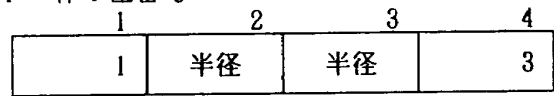
中塗りの扇形を作成するためのPFANサブルーチンでは、中心のX座標とY座標、半径、扇形の開始角と終了角、中塗りパターン、中塗りの色、枠の色が指定できる。PFANサブルーチンが呼ばれたときには以下のようにして図形データを中間データ

(1) 中塗りの円データ

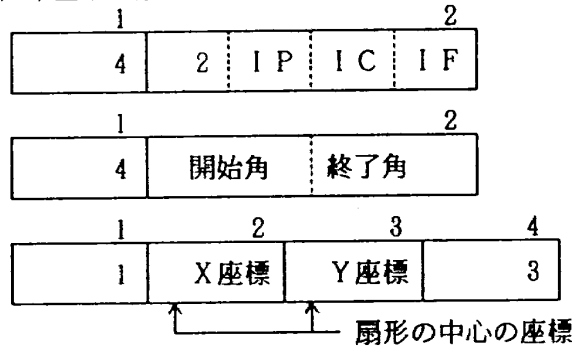


2語目は8ビット毎に分割使用

IP: 中塗りパターン
IC: 中塗りの色番号
IF: 枠の色番号



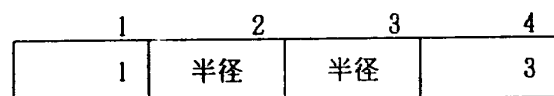
(2) 中塗りの扇形データ



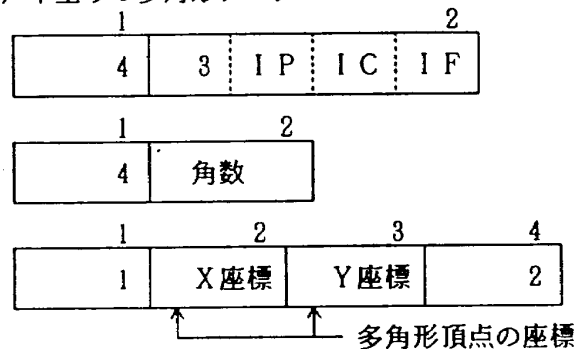
2語目は8ビット毎に分割使用

IP: 中塗りパターン
IC: 中塗りの色番号
IF: 枠の色番号

2語目は16ビット毎に分割使用



(3) 中塗りの多角形データ

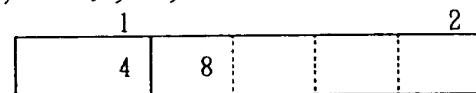


2語目は8ビット毎に分割使用

IP: 中塗りパターン
IC: 中塗りの色番号
IF: 枠の色番号

このデータは多角形の角数分使用する。

(4) ページデータ



2語目は8ビット毎に分割使用

図4.3 中間データファイルの格納形式(追加データ)

ファイルに格納している。

中間データファイルでは、中塗りの扇形情報を示す識別情報を2としている。そこで、この識別情報および指定された引数のうちの中塗りパターン、中塗りの色、枠の色の情報は、(1)の場合と同様にペン番号データに格納している。扇形の開始角と終了角もペン番号データの1語を16ビットづつに分割して格納している。また、中心のX座標とY座標および半径はペンアップで移動する2つの直線データとして格納し、最後にそれまで使用されていたペン番号で通常のペン番号データを発行している。

(3) 中塗りの多角形情報

中塗りの多角形を作成するためのPPLY2Dサ

ブルーチンでは、多角形のX座標とY座標、角数、中塗りパターン、中塗りの色、枠の色が指定できる。PPLY2Dサブブルーチンが呼ばれたときには、以下のようにして図形データを中間データファイルに格納している。

中間データファイルでは、中塗りの多角形情報を示す識別情報を3としている。そこで、この識別情報と指定された引数のうちの中塗りパターン、中塗りの色、枠の色の情報は、(1)の場合と同様にペン番号データに格納している。また、多角形の角数をペン番号データとして、さらにX座標とY座標は、XYまたはPP出力時には線画として表示されるように、角数に応じた直線データとして格納している。

そして、最後にそれまで使用されていたペン番号で通常のペン番号データを発行している。

(4) ページ情報

ページ情報を指定するPAGEサブルーチンは引数をもたない。PAGEサブルーチンが呼ばれたときには、ペン番号データの先頭8ビットに識別情報を格納している。

中間データファイルでは、ページ情報を示す識別情報を8として、通常のペン番号データと区別している。この場合にも、最後にそれまで使用されていたペン番号で通常のペン番号データを発行し、XY, PP出力時のペンの色を変更されないようにしている。

4.2 モニタジョブ名の決定とファイルのオープン処理

新モニタプログラムでは、以下の手順でモニタジョブ名を決定し、そのジョブで作成された図形データファイルのオープン処理を行う。

(1) モニタ対象ファイル名およびユーザ名のセット
新モニタプログラムでは、最初にシステムに存在

するモニタ対象の全図形ファイル名とセッション開設ユーザ名を調べる。そしてジョブ名検索時のキーとするユーザ名に、セッション開設ユーザ名をセットする。

なお、モニタ対象とし得る図形データには、PP出力待ち状態のものとXY出力待ち状態のものがあるが、モニタ処理中の図形データをXY出力しようとするるとXY出力処理が異常終了となるため、PP出力待ち状態のもののみを対象としている。

(2) モニタ対象ジョブ名の表示

ユーザ名をキーにしてモニタすべき図形データがあるか否かを調べ、図4.4の形式の初期画面を表示する。図はユーザ名X01のモニタ対象図形データが4つある場合の例である。

(3) 処理内容の問い合わせ

ここでは、どのジョブで作成した図形データをどのように処理したいのかを利用者に問い合わせる。利用者は、処理しようとするジョブ名の前の選択子記述欄に、画面右下のSELECT OPTIONSに示されている選択子のいずれかを指定することにより図形データのモニタ、消去、出力装置の変更、NL

```

-----< MONITORX >-----
COMMAND=>
SELECT   JOB          --- FILE NAME ---
=>    1  X01S001      ==>
=>    2  X01T004
=>    3  X01T005      --- BACK COLOR ---
=>    4  X01S011      ==> 8 (1-WHITE 2-BLUE 3-RED 4-GREEN 5-YELLOW
                          6-MAGENTA 7-SYANIC 8-BLACK )

                                SELECT OPTIONS
                                S - SELECT
                                D - DELETE
                                X - OUTPUT DEVICE CHANGE
                                  PP  --> XY
                                  PPXY --> XY
                                N - NLPOUT
                                A - NLPOUT ( AUTO SCALE )

                                PFK3 - MONITORX END
  
```

図4.4 新モニタプログラムの初期画面

P出力を行うことができる。モニタ処理の場合にはジョブ名またはジョブ名を簡略化するために付けられている番号をコマンド欄で指定するか、あるいはファイル名をファイル名指定欄で指定してもよい。また、初期画面表示中にPF3キーを押すとモニタプログラムは終了する。

なお、モニタする図形データはセッション開設ユーザ以外が作成したものでよい。このような場合、コマンド指定欄で“ID ユーザ名”の形式でユーザ名を指定すると、ジョブ名検索時のキーとするユーザ名を変更することができる。ただし、NSシステムではファイルの機密保護を行っているため、他ユーザのジョブ結果をモニタする場合にはそのファイルに対するアクセス権がなければならない。

(4) 処理内容のチェックと処置

ユーザの指定した処理内容に応じて以下の処置を行う。なお、該当する項目がない場合には(3)の処理に戻る。

- a. モニタ処理指定の場合には次項の処理へ移る。
- b. ジョブ名検索時のキーとするユーザ名変更指定であれば、キーとするユーザ名に指定ユーザ名をセットし、(2)項の処理に戻る。
- c. 図形データの消去指定の場合には、消去処理を行った後に(3)項の処理へ戻る。
- d. 出力装置変更指定の場合には変更手続きを行い、(3)項の処理へ戻る。
- e. NLP出力指定の場合には第5.3項に述べるプログラムによりNLP用の図形データを作成して、(3)項の処理へ戻る。このプログラムの処理はバッチ処理時にNLP文を使用した場合と同じである。
- f. PF3キーが押された場合には、モニタ処理を終了させる。

(5) 指定ジョブ名の確認

ジョブ名を簡略化するための番号が指定された場合には、それに対応するジョブ名が指定されたものとみなし、次項の処理へ移る。また、ジョブ名が指定された場合には、モニタ対象ジョブ名に該当するものがあるか否かを調べ、あれば次項の処理へ移り、ない場合にはその旨のメッセージを表示し、(2)項の処理へ戻る。

(6) ファイルのオープン処理

指定されたジョブで作成された中間データファイルを、ファイル識別番号1のファイルとしてオープンする。このとき、中間データファイルにデータブロックがあるか否かを調べ、データブロックがないファイルはシステムファイルの有効利用を図るために消去し、その旨のメッセージを表示して、(2)項の処理へ戻る。また、データブロックがある場合には、次項の処理へ移る。

(7) 制御変数の初期化

モニタ処理時に必要となるFACTSおよびISKIP変数を初期化する。

このうち、FACTS変数は表示時のファクタ値が設定されているか否かを示す変数で、初期化時には設定されていない旨の0.0をセットする。この値はファクタ指定のメニューが選択された場合に更新される。

また、ISKIP変数はスキップすべきページ数を格納する変数で、初期化時には0をセットする。コマンドでページのスキップが指定された場合にはこの変数に格納され、図形データを順次読み込みページの区切になるたびに1ずつ減算し、0になったときに表示を行う。なお、全ページ表示のメニューが選択されたときのように、ページの区切りを無視したい場合にはISKIP変数を負にすることになっている。

4.3 中間データファイルの初期化

中間データファイルの初期化処理では、入力データを初期の状態に戻すために、以下の処理を行う。

(1) 中間データファイルの先頭データの読み込み

中間データの格納されている図形データファイルをリワインドし、中間データファイルの先頭に格納してあるデータブロックを入力し、データ入力位置を2語目にセットする。

(2) 疑似カルコンプライブラリの諸変数の初期化

GCPライブラリには、カルコンプ形式の作図プログラムを入力できるようにするための疑似カルコンプライブラリが用意されている。このライブラリにはカルコンプの出力に合わせて、PLOTサブルーチン、SYMBOLサブルーチン、NUMBER

サブルーチン、NEWPENサブルーチン、FACTORサブルーチンがあるが、GCPには原点移動や一時的なファクタを指定する機能を用意していないため、指定された座標データはファクタを掛けた絶対座標データに変換した後に、GCPの図形データファイルに格納している。

ここでいう諸変数の初期化とは、これら変換用の変数の初期化を意味する。

(3) 表示図形データ量の初期化処理

作成した図形データは主にPAGEサブルーチンによるページの区切りで表示されるが、表示処理時に表示すべき図形データがないとエラーとなる。そこで、表示すべきデータがあるか否かの判定に使用するICOUNT変数に0をセットして初期化する。

4.4 GCP形式の図形データの作成

中間データファイルには第4.1項に述べた図形データおよび制御データが格納されている。新モニタープログラムではこれらのデータからGCP形式の図形データを作成するために、以下のような処理を行っている。

4.4.1 セグメントのオープン処理

新モニタープログラムでは、表示する図形データをセグメント番号1のデータとして作成する。そのため、データの作成に先立ちセグメント番号1のセグメントのオープン処理を行う。

4.4.2 図形データの変換処理

データブロックに格納されている図形データの先頭にはデータ識別子があり、それに応じてGCP形式の図形データに変換するために以下の処理を行う。ここでの処理は、特に明記しない限り、繰り返し行うものとする。また、表示すべきデータがあるか否かを示すICOUNT変数は、データに応じて適時加算され、図形データの表示後に初期化する。

a. ページ情報

データの1語目に格納されているデータ識別子がペン番号データを示す4で、かつ2語目のペン番号データの先頭8ビットにページ情報を示す8が格納されている場合には、第4.4.3項のページ処理へ移

る。

b. 中塗りの円情報

データの1語目に格納されているデータ識別子がペン番号データを示す4で、かつ2語目のペン番号データの先頭8ビットに中塗りの円情報を示す1が格納されている場合には、以下の処理をする。

まず、ペン番号データの1語を8ビットずつに分割し、作図プログラムで指定された中塗りパターン、中塗りの色、枠の色の情報を復元する。つづいて、次のデータを読み込むためにデータ入力位置を2語移動し、直線データとして格納されているX座標、Y座標を読み込む。さらに、データ入力位置を4語移動し、直線データとして格納されている半径を読み込む。

これらのデータから、GCPのPCIRCLサブルーチンによって中塗り円データを作成し、次のデータを読み込むためにデータ入力位置を4語移動する。

c. 中塗りの扇形情報

データの1語目に格納されているデータ識別子がペン番号データを示す4で、かつ2語目のペン番号データの先頭8ビットに中塗りの扇形情報を示す2が格納されている場合には、以下の処理をする。

まず、ペン番号データの1語を8ビットずつに分割し、作図プログラムで指定された中塗りパターン、中塗りの色、枠の色の情報を復元する。つづいて、次のデータを読み込むためにデータ入力位置を2語移動してペン番号データを読み込み、1語を16ビットずつに分割して扇形の開始角と終了角データを復元する。さらに、データ入力位置を2語移動して直線データとして格納されているX座標、Y座標を読み込み、同様にデータ入力位置を4語移動し、直線データとして格納されている半径を読み込む。

これらのデータから、GCPのPFANサブルーチンによって中塗り円データを作成し、次のデータを読み込むためにデータ入力位置を4語移動する。

d. 中塗りの多角形情報

データの1語目に格納されているデータ識別子がペン番号データを示す4で、かつ2語目のペン番号データの先頭8ビットに中塗り多角形の情報を示す3が格納されている場合には、以下の処理をする。

まず、ペン番号データの1語を8ビットずつに分

割し、作図プログラムで指定された中塗りパターン、中塗りの色、枠の色の情報を復元する。つづいて、次のデータを読み込むためにデータ入力位置を2語移動し、ペン番号データとして格納されている多角形の角数を復元し、データ入力位置を2語移動する。そして、直線データとして格納されているX座標、Y座標の読み込みおよびデータ入力位置の4語移動の処理を角数に応じて行う。最後に、これらのデータからGCPのPPLY3Dサブルーチンによって中塗り多角形データを作成する。

e. 直線データ

データの1語目に格納されているデータ識別子が直線データを示す1の場合には、2語目から4語目に格納されているデータを使用してGCP形式の図形データを作成する。この処理には疑似カルコンプライブラリのPLOTサブルーチンを使用する。このとき、直線データが原点移動を伴うものである場合には相対座標を絶対座標に変換するための変数を更新する。その後、次の図形データを読み込むために、データ入力位置を4語移動する。

f. 文字データ

データの1語目に格納されているデータ識別子が文字データを示す2の場合には、2語目以降に格納されているデータを使用してGCP形式の図形データを作成する。この処理には疑似カルコンプライブラリのSYMBOLサブルーチンを使用する。その後、次の図形データを読み込むために、データ入力位置を7語から最大22語移動する。

g. 数値データ

データの1語目に格納されているデータ識別子が数値データを示す3の場合には、2語目以降に格納されているデータを使用してGCP形式の図形データを作成する。この処理には疑似カルコンプライブラリのNUMBERサブルーチンを使用する。そして、次の図形データを読み込むために、データ入力位置をデータ数に応じて移動する。

h. ペン番号データ

データの1語目に格納されているデータ識別子が4で、かつ2語目のペン番号データの先頭8ビットにデータが格納されていない通常のペン番号データの場合には、疑似カルコンプライブラリのNEWP

ENサブルーチンを使用してGCP形式の図形データを作成する。その後、次の図形データを読み込むために、データ入力位置を2語移動する。

i. ファクタ値データ

データの1語目に格納されているデータ識別子がファクタ値データを示す5の場合には、疑似カルコンプライブラリのFACTORサブルーチンを使用して疑似カルコンプライブラリのファクタ変数を変更する。その後、次の図形データを読み込むために、データ入力位置を2語移動する。

j. ブロックエンドデータ

1語目に格納されているデータ識別子がブロックエンドデータを示す15の場合には、中間データファイルから後続のレコードを読み込む。そして、そのレコードが図形データを格納してあるデータブロックであれば、データ入力位置を2語目にセットし、第4.4.2項の処理を続行する。また、読み込みレコードが図形データの終了を示す終了ブロックであった場合や、読み込むべきレコードがなくなった場合には、第4.4.4項の最後ページ処理へ移る。

4.4.3 ページ処理

ここでは、中間データファイルの中にページ情報があった場合の処理を行う。ただし、 $ISKIP < 0$ の場合にはページ情報を無視し、第4.4.1項へ戻る。

まず、図形データがあるか否かをICOUNT変数で調べ、表示すべきデータがない場合には、後続のページデータを作成するために第4.4.1項に戻る。また、表示すべきデータがある場合にはページ番号に1加算するが、それ以降の処理はスキップ処理中であるか否かにより異なる。すなわち、 $ISKIP \geq 1$ の場合にはISKIP変数から1減算し、第4.4.1項の処理に戻る。また、それ以外の場合には第4.5項の表示のための処理に移る。

4.4.4. 最終ページ処理

ここでは、中間データファイルの中に入力すべき図形データがなくなった場合の処理を行う。まず、図形データがあるか否かをICOUNT変数で調べ、表示すべきデータがない場合には、第4.9項の中間ファイルの処置へ移る。また、表示すべきデータが

ある場合にはページ処理と同様に行う。

4.5 図形表示のための初期化処理

4.5.1 表示プログラムの初期化処理

(1) スクリーンウィンドおよびバーチャルウィンドの初期化

新モニタプログラムでは、図形データは縦横比を変えずにできるだけ大きく表示することにしている。

そこで、スクリーンウィンドは2次元GD表示プログラムの場合と同様に、装置の種類の応じてそれぞれの適正値をセットするが、バーチャルウィンドは表示すべき図形データをすべて調べそれらがぴったり収まるような値を、そのページを表示する際の初期値としてセットしている。

(2) ローカル処理変数の初期化

図形データ生成時にはローカル処理変換プログラムで変換されるが、初期表示時には変換処理が有効とならないように、ローカル処理変数を初期化する。

(3) 制御変数の初期化

ここでは、IMM、IDISP変数を初期化する。このうち、IMM変数は文字データプログラムによって生成された追加データがあるか否かを0か1かで表す変数であり、文字データインタフェースプログラムを呼び出す際の引数ともなる。この変数は初期設定時に0をセットするが、その後はインタフェースプログラムで適時変更される。

また、IDISP変数はFACTS変数の指定を有効にするか否かを示す変数であり、0の場合のみ有効とする。この変数は初期設定時には0をセットするが、拡大、移動、縮小移動のメニューが選択されたときには1または2がセットされ、初期画面表示メニューが選択されたときおよびファクタ値指定メニューが選択されたときに初期化される。

4.5.2 画面の初期化およびウィンドの設定

新モニタプログラムでは2次元GD表示プログラムと同様の理由により、図の表示に先立ち画面を常に初期化している。そして、セットされているバーチャルウィンドの値にもとずき、表示時のバーチャルウィンドを設定する。ただし、IDISP=2であればバーチャルウィンドの移動のみを行い、F

ACTS>0.0でIDISP=0であったならば、FACTSに格納されているファクタ値で表示されるように、バーチャルウィンドを設定する。

なお、設定したバーチャルウィンドの値によって図の縦横比が変わってしまうときにはバーチャルウィンド縦または横の値を広げている。

なお、モニタ表示画面の右上には、表示データのファクタとページ番号を表示している。

4.6 図形データの生成および変換処理

新モニタプログラムでは、2次元GD表示プログラムの場合と同様の方法で、表示用図形データの生成を行っている。しかし、クラス情報、セグメント情報は使用しないので、GCP形式の図形データから順次表示用の図形データを生成し、ローカル処理変換のみをおこなっている。なお、図形データの生成およびローカル処理変換で使用するプログラムは、2次元GD表示プログラムで使用したものと同一のプログラムである。

4.7 追加データの表示とインタフェースプログラムの呼び出し

IMM変数が1であった場合には、文字データプログラムによってGCP図形データに対する追加データが別ファイルに作成されているので、前項と同様の方法で図形データの生成・表示処理を行う。表示後、文字データインタフェースプログラムが呼び出され、第5章で述べるGCP図形データファイルへの追加データ保存や、新たな追加データの作成を行うことができる。ここでの処理結果により、先の変数の値は適時変更される。インタフェースプログラムでの処理後、第4.5.2項へ戻る。

4.8 メニュー処理

新モニタプログラムには、表4.1に示す24のメニューが用意されている。これらのメニューは以下のように処理される。

a. 拡大処理

メニューで拡大処理が選択された場合には、どの部分を拡大したいのかを問い合わせる必要がある。そこで、十字カーソルを2度表示し、ユーザから拡

表 4.1 モニタ処理時に利用可能なメニュー一覧(つづく)

コマンド	内容	説明
Z	図の拡大	十字カーソルが2度表示されるので、ジョイスティックまたは矢印キーを使用して拡大したい部分の左下と右上、または左上と右下の2点の座標を入力する。指定された座標値は図の縦横比が変化しないような座標値に直された上でその部分が画面全体に拡大表示される。
T	図の移動	十字カーソルが2度表示されるので、ジョイスティックまたは矢印キーを使用して2点の座標を入力する。最初に指定された座標が2番目に指定された位置へ移動するようにバーチャルウィンドが変更される。
F	特定位置への表示 (縮小移動)	十字カーソルが2度表示されるので、ジョイスティックまたは矢印キーを使用して画面に表示中の図形データの移動先位置の左下と右上、または左上と右下の2点の座標を入力する。表示中のデータは指定された位置に図の縦横比が変化しないように縮小表示される。
R X	X軸を中心とした 回転角の指定	画面に表示中の図の回転角度(右手系)を"R X 45."というように1文字分のブランクをあけて実数形式で指定する。図形データはデータのX軸を中心に指定角度だけ回転する。
R Y	Y軸を中心とした 回転角の指定	画面に表示中の図の回転角度(右手系)を"R Y 45."というように1文字分のブランクをあけて実数形式で指定する。図形データはデータのY軸を中心に指定角度だけ回転する。
R Z	Z軸を中心とした 回転角の指定	画面に表示中の図の回転角度(右手系)を"R Z 45."というように1文字分のブランクをあけて実数形式で指定する。図形データはデータのZ軸を中心に指定角度だけ回転する。
R R	回転角の複数指定	R X、R Y、R Zコマンドで行う図の回転指定をまとめて行う。Eコマンドを投入するまで繰り返しR X、R Y、R Zコマンドが投入できる。
C	表示色の変更	カラー設定画面になるのでメッセージの色、ペン番号に対応する色およびバックカラーの色を指定する。これ以降の図はこの指定に従って表示される。
X Y	X Y、P P出力の 指定	モニタコマンド実行時の最初のX Yコマンド投入時には、パッチジョブの場合と同様にX Y出力か、P P出力か、P PおよびX Y出力かの選択およびX Y出力時の用紙等のセット番号の指定が行える。モニタ処理中にこのようにして作成した図形データは1つのファイルに格納され、開設中のセッションのジョブ番号で管理される。
H C	N L P出力の指定	表示中の図形データをN L Pに出力する。モニタ処理中に作成されたN L P出力用図形データは開設中のセッションのラインプリンタ出力結果として格納される。
O P R A	O P R出力の指定	表示中の図形データを極超音速風洞建屋のO P Rへ出力する。
O P R F	O P R出力の指定	表示中の図形データを原動機1号館のO P Rへ出力する。
M	文字列入力指定	文字列の色番号(1~8)および文字列の入力待ちとなるので、"5 ABC"のように入力する。次に十字カーソルが2度表示されるのでジョイスティックまたは矢印キーを使用して文字の左下座標と右下座標位置を指定する。文字列はこの2点上に納まるような大きさ(最大11.9)で表示される。その後文字データを図形データに追加するか否かの指定ができる。文字列を図形データに追加した場合には繰り返しこの処理が行え、文字列をブランクにした時に処理は終了する。なお、この文字データはモニタ中の図形データには有効とはならないがH C、X Y、N L P、O P R A、O P R Fコマンドには有効となる。また、この文字データはR Xコマンド等により回転することはできない。
F A	ファクタの指定	図を表示する際のファクタを1文字分のブランクをあけて実数形式で指定する。ファクタ値を指定しなかった場合には初期状態のファクタ値がセットされる。この値はZコマンド、Fコマンド、D Aコマンドの場合には有効とならない。

表 4.1 モニタ処理時に利用可能なメニュー一覧(つづき)

コマンド	内容	説明
H	コマンド種類の表示	ローカル処理時に使用できるコマンドの種類および簡単な投入例を表示する。この表示画面のときでもローカル処理用のコマンドは投入できる。
D	初期画面の表示	Z、T、F、RX、RY、RZ、RRコマンドにより拡大、回転等が行われた図を初期状態に戻す。
(ファンク)	次ページの表示	モニタ中の図形データの次ページが表示される。
S	スキップページ数の指定	スキップするページ数を" S 15" のように1文字文のブランクをあけて1桁または2桁で入力する。3ページ目を表示中のときに" S 3" と入力すると6ページ目が表示される。
DN	表示ページ番号の指定	表示するページ番号を" S 3" のように1文字文のブランクをあけて1桁または2桁で入力する。
DA	全ページの表示	モニタ中の全ページのデータが一度に表示される。
R	先頭ページの表示	モニタ中の図形データの先頭のページが表示される。
NA	図のNLP出力の指定	モニタ中の図形データをTSSのNLP OUTコマンドと同様に処理する。
E	モニタの終了	プロットデータ処置画面となるのでモニタ処理していた図形データを保存するか否かの指定をする。その後、再びモニタジョブ名の指定が行える。
A	モニタプログラムの強制終了	プロットデータ処置画面となるのでモニタ処理していた図形データを保存するか否かの指定をする。その後、モニタプログラムは終了し、READY状態となる。

大部分の左下と右上、または左上と右下の2点の座標を通知されるのを待つ。拡大部分が通知されたならば、その部分が画面全体に表示されるように表示データのバーチャルウィンドを変更し、IDISP変数に1をセットして、第4.5.2項の処理へ戻る。

b. 移動処理

メニューで移動処理が選択された場合には、どこからどこへ移動したいのかをユーザに問い合わせる必要がある。そこで、十字カーソルを2度表示し、ユーザから移動元座標と移動先座標が通知されるのを待つ。2点の座標が通知されたならば、1点目の位置のデータが2点目の位置に移動するように表示データのバーチャルウィンドを変更し、IDISP変数に2をセットして、第4.5.2項の処理へ戻る。

c. 縮小移動処理

縮小移動のメニューが選択された場合には、現在画面上に表示中の図形データを画面上のどの位置へ表示したいのかをユーザに問い合わせる必要がある。そこで、十字カーソルを2度表示し、ユーザから表示位置の左下と右上、または左上と右下の2点の座

標を通知されるのを待つ。表示位置が通知されたならば、現在画面全体に表示中のデータがその位置に表示されるようにバーチャルウィンドを変更し、IDISP変数に1をセットして、第4.5.2項の処理へ戻る。

d. 回転処理

X軸回りの回転処理、Y軸回りの回転処理、Z軸回りの回転処理のメニューが選択された場合には、ローカル処理変換プログラムを呼び出して、ローカル処理変換マトリックスを変更する。そして、第4.5.2項の処理へ戻る。また、回転の複数指定のメニューが選択された場合には、指定が終了するまで順次回転軸および回転角の情報を格納し、その後ローカル処理変換マトリックスを更新し、第4.5.2項の処理へ戻る。

e. 表示色変更処理

表示色変更処理のメニューが選択された場合には、カラー設定画面を表示し、メッセージおよびペン番号1から8のそれぞれに対応した色を指定できるようにする。これ以降の図形表示では指定に従って表

示されるように、指定された情報を色変換テーブルに格納して、第4.5.2項の処理へ戻る。

f. 他装置への出力

表示中の図形データをXY, NLP, OPRに出力するメニューが選択された場合には、データ変換に必要な各種テーブル、装置に応じたシザリング情報をファイルに格納し、第5章に述べる図形データ形式変換プログラムを呼び出す。そして、第4.5.2項の処理へ戻る。

g. 文字入力処理

文字入力処理メニューが選択された場合には、文字データインタフェースプログラムを呼び出し、文字データの追加作成を行う。その後、第4.5.2項の処理へ戻る。

h. ファクタ処理

メニューで表示時のファクタ処理が選択された場合には、FACTS変数に指定されたファクタ値をセットして、第4.5.2項の処理へ戻る。ただし、ファクタ値が指定されなかった場合には、初期表示時のファクタ値で表示されるように、FACTS変数に0.0をセットする。

i. メニュー表示処理

メニュー表示が選択された場合には、新モニタープログラムで使用できるメニューの簡単な説明および使用例を表示し、メニューが選択されるのを待ち、選択されたならばその処理を行う。なお、メニューが選択されずにENTERキーが押された場合には、第4.5.2項の処理へ戻る。

j. 初期画面表示処理

初期画面表示処理のメニューが選択された場合には、バーチャルウィンドを初期状態に戻し、ローカル処理変換マトリックスを初期化するため、第4.5.1項の処理へ戻る。

k. 次ページの表示処理

メニューで次ページの表示が選択された場合には、第4.4項の処理へ戻る。

l. スキップページ数指定処理

メニューでスキップページ数指定が選択された場合には、ISKIP変数に指定された値をセットし、第4.4項の処理へ戻る。

m. 表示ページ番号指定処理

メニューで表示ページ番号指定が選択された場合には、現在のページ番号と比較し、後のページであれば現在のページ番号と指定ページ番号の差をISKIP変数にセットし、第4.4項の処理へ戻る。また、現在のページよりも前のページであれば、ISKIP変数に指定された値をセットし、第4.3項の処理へ戻る。

n. 全ページ表示処理

メニューで全ページ表示が選択された場合には、ISKIP変数に-1をセットして、第4.3項の処理へ戻る。

o. 先頭ページの表示処理

メニューで先頭ページの表示が選択された場合には、第4.3項の処理へ戻る。

p. 全ページのNLP出力の指定

中間データファイルに格納されている全データを表示中の大きさとは関係なく、作成されたときの大きさまたはNLP用紙に収まる大きさでNLPに出力することができる。このメニューが選択された場合には、ISKIP変数に現在のページ番号を保存し、システムで用意されているNLPOUTコマンドを実行する。このコマンドは、PP, XY出力用の図形データをNLP用に変換するプログラムを呼び出すためのコマンドである。この処理後、処理前の図を再表示するために、第4.3項の処理へ戻る。

q. モニタの終了および強制終了処理

メニューでモニタの終了または強制終了が選択された場合には、中間データファイルの処置を行うために第4.9項の処理へ移る。

4.9 中間データファイルの処置

ここでの処理は、オープン中の中間データファイルに対するモニタ処理が終了した場合に行われる。

まず、オープン中の中間データファイルをそのまま保存するか、消去するかを利用者に問い合わせ、消去すると指定された場合には、中間データファイルを消去する。その後、第4.2(2)項へ戻る。

ただし、モニタ処理中の中間データファイルがバッチジョブ実行時にPPXY文で出力されたものである場合には、中間データファイルを保存するか消去するだけでなく、PP出力をキャンセルしてX

Y出力に変更するか否かの選択も行えるようにしている。

中間データファイルの処置が終了したならば、次のモニタ処理を行えるようにするために、第4.2(2)項へ戻る。ただし、強制終了メニューによってこの処理が行われた場合には、新モニタプログラムの処理を終了させる。

5. インタフェースプログラム と形式変換プログラム

GCPライブラリには、2次元GD表示プログラムとは別に、2次元GDに表示された図形データに流線、オイルフロー、文字データを追加するためのインタフェースプログラムおよび表示されている図形データをXY, NLP, OPRへ出力するための形式変換プログラムが用意されている。

本章では、これらのプログラムについて説明する。

5.1 インタフェースプログラム

インタフェースプログラムには、流線インタフェースプログラム、オイルフローインタフェースプログラム、文字データインタフェースプログラムがあり、呼び出し時の引数が0の場合には流線、オイルフロー、文字列データを追加データとして作成する。また、引数が1以上の場合には、追加データをGCP図形データファイルへ保存したり、新たな追加データの作成処理を行うものである。なお、処理の結果により、先の引数の値を適時変更する。以下に、これらのインタフェースプログラムについて説明する。

5.1.1 流線インタフェースプログラム

一般的に、流線データは格子図等に重ねて描くことが多い。したがって、格子図等がセグメント情報、クラス情報等によって変換されている場合、流線データもそれらの情報によって変換しないと表示がずれることになる。そこで、インタフェースプログラムでは、流線データは一旦識別番号90のワークファイルに格納し、作成した流線データを保存する場合には、最終セグメントに対する追加データとして図形データファイルの最後にコピーすることにして

いる。したがって、流線データと重ねて描く必要のある格子データ等は、図形データファイルの最後のセグメントとして作成しておく必要がある。

なお、流線インタフェースプログラムは、引数の値が1以上であれば以下に示す(1)項の処理を、0であれば(2)項の処理を行う。

(1) 追加データの処置

追加データがある場合には、そのデータをGCP図形データファイルに保存するか否かを利用者に問い合わせる。保存以外の返答がなされた場合には、引数に0をセットし処理を終了する。

また、保存すると返答された場合には、作図プログラムで作成された図形データファイルの最後に追加データをコピーする。そして、次項の処理へ移る。

(2) 追加データの作成

流線インタフェースプログラムでは、流線データの作成にあたり、流線の色、開始点の格子位置を利用者に問い合わせる。これに対する入力データが投入されたならば、その値の妥当性を調べ、妥当であれば流線プログラムを呼び出し、識別番号90のファイルにデータを生成し、引数に1をセットして処理を終了する。もし、入力データが妥当なものでなければ、エラーメッセージを発行し、引数に0をセットして処理を終了する。

なお、問い合わせに対し何も指定せずにコマンド終結キーを押した場合には、流線処理の終了とみなし、引数に0をセットして処理を終了する。

5.1.2 オイルフローインタフェースプログラム

オイルフローインタフェースプログラムは、引数の値が1であれば以下に示す(1)項の処理を、0であれば(2)項の処理を行う。なお、インタフェースプログラムで作成するオイルフローデータも一旦識別番号90のファイルに作成し、保存する場合には流線データと同様の理由から、最終セグメントのデータとして図形データファイルの最後にコピーすることになっている。

(1) 追加データの処置

追加データがある場合には、そのデータをGCP図形データファイルに保存するか否かを利用者に問い合わせる。保存以外の返答がなされた場合には、

引数に0をセットし処理を終了する。

また、保存すると返答された場合には、作図プログラムで作成された図形データファイルの最後に追加データをコピーする。その後、(2)項の処理を行う。

(2) 追加データの作成

オイルフローインタフェースプログラムでは、オイルフローデータの作成にあたり、オイルフローの色、開始点の格子位置を利用者に問い合わせる。これに対する入力データが投入されたならば、その値の妥当性を調べ、妥当であればオイルフロープログラムを呼び出しデータを生成し、引数に1をセットして処理を終了する。もし、入力データが妥当なものでなければエラーメッセージを発行し、引数に0をセットして処理を終了する。

なお、問い合わせに対し何も指定せずにコマンド終結キーを押した場合には、処理終了とみなし、引数に0をセットして処理を終了する。

5.1.3 文字データインタフェースプログラム

このプログラムは、表示されている図形データを見ながら任意の位置へ任意の大きさで文字を重ねて描くためのプログラムである。このプログラムを介して作成される図形データは、セグメント情報、クラス情報等によって変換された後のデータに重ね書きするだけでよい。そこで、2次元GD表示プログラムでは、インタフェースプログラムを介して作成される文字データは、一般に使用されることが少ないセグメント番号50の図形データとして保存し、セグメント情報やクラス情報、ローカル処理情報による変換処理は一切行わないものとしている。なお、文字データインタフェースプログラムは引数の値が1であれば以下に示す(1)項の処理を、0であれば(2)項の処理を行う。

(1) 追加データの処置

追加データがある場合には、そのデータをGCP図形データファイルに保存するか否かを利用者に問い合わせる。保存以外の返答がなされた場合には、引数に0をセットし処理を終了する。

また、保存すると返答された場合には、作図プログラムで作成された図形データファイルの最後に追加データをコピーする。この後で(2)項の処理を行う。

(2) 追加データの作成

文字データインタフェースプログラムでは、文字データの作成にあたり、最初に文字の色と文字列を利用者に問い合わせる。これに対する入力データが投入されたならば、その値の妥当性を調べ、妥当であれば続いて文字データの開始点および終了点の座標を問い合わせる。そして、開始点から終了点までの位置に指定された文字データが描かれるように、文字の高さ、角度等を求め、文字データ生成プログラムを呼び出す。そして、引数に1をセットして処理を終了する。

もし、妥当なものでなければ、引数に0をセットして処理を終了する。

なお、問い合わせに対し何も指定せずにコマンド終結キーを押した場合には、処理終了とみなし、引数に0をセットして処理を終了する。

5.2 図形データ形式変換プログラム

既存の図形データを他の装置に出力するためには、出力装置に応じて図形データを形式変換することが必要となる。これを実現する上で1番重要なことは、ライブラリの結合方法である。

図形データのある装置に出力しようとした場合、必ずその装置にサポートされている図形出力ライブラリを介して行わなくてはならない。1つのプログラムで複数の図形処理装置用のライブラリを使用する場合、呼び出し名がすべて異なる場合には特に支障はないが、NSシステムの図形処理装置のように個々の図形出力ライブラリの呼び出し名に同じものがある場合には、最初に結合されているプログラムが呼び出されるため、プログラム上で任意の図形出力ライブラリのプログラムを呼び出すことはできない。

そこでGCPでは、出力装置に応じた図形変換はそれぞれ独立した実行形式のプログラムとして用意し、プログラム起動という形で使用することにした。そのため、サブルーチン呼び出しで使用する場合には引数を使用して図形変換のための制御データを受け渡すことができないので、図形データ以外の情報はすべて制御データファイルを介して行うことにしている。なお、このファイルはファイル

識別番号98のワークファイルを使用している。

本項では、形式変換プログラムで使用するの制御データの内容と、それに対する2次元GD表示プログラム等での設定、それぞれの形式変換プログラムでの処理内容について説明する。

5.2.1 図形変換時の制御データ

2次元GDに表示されている図形データを、そのままのイメージでXY, PP, NLP, OPRへ出力しようとした場合、GCP形式の図形データの他に、各種変換テーブルと表示時のバーチャルウィンドが必要となる。そこで、GCPではこれらのデータを制御データファイルに格納し、図形データファイルとともに図形データ形式変換プログラムに渡す方式を採っている。

以下に、図形変換時に使用される制御データの内容について説明する。

なお、この制御データはファイル識別情報98のデータとして入力できるように、図形データの変換に先立ち、読み込み可能な状態としておくものとする。また、入出力は書式なし入出力文を使用するものとする。

(1) 図形データファイル識別番号および図形データファイル名

制御データファイルの1番目の情報は、図形データファイル識別番号および図形データファイル名である。

図形変換時に入力データとして使用するGCP形式の図形データは、通常ファイル識別番号91のワークファイルに格納している。しかし、形式変換プログラムで使用するファイルが91以外の識別番号でアロケートされている場合でも、またアロケート処理が行われていない一般ファイルの場合でも取り扱えるようにする必要がある。そのために、これらの情報を用意している。

ただし、識別番号は図形変換時に使用しない1～96のいずれかとし、ファイル名は先頭に空白を入れない40文字以内の英数字とする。したがって、1～96以外の識別番号が指定された場合や、先頭の1バイトが空白のファイル名が指定された場合には、それぞれ指定がなかったものとして処理をする。

なお、形式変換プログラムではこれらの指定によって、以下のように処理している。

a) 識別番号のみが指定された場合

図形データファイルのアロケート処理が行われているものとみなし、図形データの読み込みは指定された識別番号のファイルに対して行われるように処置する。

b) ファイル名のみが指定された場合

ファイル識別番号91で指定ファイルのアロケートし、図形データの読み込みが識別番号91のファイルに対して行われるように処置する。

c) 識別番号およびファイルが指定された場合

指定された識別番号で指定ファイルのアロケートし、図形データの読み込みが指定された識別番号に対して行われるように処置する。

(2) 図形変換情報およびシザリング情報

制御データファイルの2番目の情報は、図形変換情報およびシザリング情報である。図形変換情報には図形変換時のファクタ、X方向の移動量、Y方向の移動量がある。また、シザリング情報にはX座標の最小値、Y座標の最小値、X座標の最大値、Y座標の最大値がある。

図形データ形式変換プログラムでは、これらの指定に従い以下のように処理している。

たとえば、ファクタ、X方向の移動量、Y方向の移動量がそれぞれFACT, X0, Y0であり、Xデータの最小値、最大値、Yデータの最小値、最大値がそれぞれXMIN, XMAX, YMIN, YMAXであったとする。その場合、入力データのすべてのX座標、Y座標の値X, Yは

$$X = X * FACT + X0$$

$$Y = Y * FACT + Y0$$

のように置き直された上で、 $XMIN \leq X \leq XMAX$, $YMIN \leq Y \leq YMAX$ の範囲以外のデータをシザリングする。

(3) 図形データの回転情報

制御データファイルの3番目の情報は、ローカル処理時の回転情報である。

2次元GDで表示中のデータに対しローカル処理で図形データを回転した場合には、ローカル処理マトリックスを作成し、それを利用して入力データを

回転している。形式変換プログラムでも同様の処理を行う必要があるため、変換の必要の有無と変換マトリックスを指定する。

(4) 色変換情報

制御データファイルの4番目の情報は、図形出力時の色変換情報である。この情報は色番号1～8の色に対して何色で表示するかを指定するもので、PP・XY出力用の形式変換プログラムの場合のみ指定する。

2次元GDは8色しか使用できないため、すべての色は1から8の色番号の色として表示されるが、メニュー処理で色番号に対応する表示色が変更された場合には、プログラム内の色変換テーブルに格納し、それに従って表示している。形式変換プログラムの色変換情報は、このような場合に有効となる。

(5) セグメント情報およびクラス情報

制御データファイルの5番目の情報は、図形データに対するセグメント情報およびクラス情報である。

2次元GDに表示されている図形データの形状をそのままXY等に出力するために、(3)項の回転情報とともにこれらの情報が必要となる。なお、この情報に限り指定しなくてもよいものとするが、その場合にはセグメント番号1のデータのみが図形データファイルにあるとみなして処理をする。

5.2.2 2次元GD表示プログラムおよび新モニタープログラムでの処理

2次元GD表示プログラムおよび新モニタープログラムでは、形式変換プログラムを呼び出す前に、以下の処理を行っている。

(1) 図形データファイル識別番号、図形データファイル名の指定

GCP形式の図形データファイルをアロケートしているファイル識別番号のみをセットし、ファイル名は指定しない。

ただし、PP・XY用形式変換プログラムを使用するときで8以上の色番号を使用している図形データの場合のみ、(4)項で述べるように識別番号92をセットする。

(2) 図形変換時のファクタ、移動量、シザリング情報の指定

2次元表示プログラムでは、大多数の2次元GDの画面の幅が約25cmであるため、表示中の図形データがほぼ同じ大きさで出力されるように、バーチャルウィンドの左下の座標が(0.0, 0.0)、右上の座標が(25.0, 0.0)となるようにファクタおよび移動量を設定する。また、シザリングの指定は $0.0 \leq X \leq 26.0$ 、 $0.0 \leq Y \leq 27.0$ の範囲でデータが出力されるように設定する。

(3) 図形データの回転情報の指定

2次元GDの表示時に使用しているローカル処理変換マトリックスの値と設定の有無を指定する。

(4) 色変換情報

PP・XY出力用の形式変換プログラムを使用する場合に限り、2次元表示プログラムで使用している色変換情報をそのまま指定する。

XYで使用できる色は最大4色であるが、カラーテーブルを使用して表示している場合でも、出力した図形データをモニター処理したときには、そのままの色で再現されることが望ましい。そこで、8以上の色番号を使用している場合にのみ、図形データファイルに格納されているペン番号を表示時のペン番号に置き換えて、ファイル識別番号92のワークファイルに格納している。したがって、この場合のみ(1)項で設定するファイル識別番号を92としている。

(5) セグメント情報およびクラス情報

2次元GD表示プログラムの場合には、2次元GDに表示されている図形データの形状をそのままNLP等に出力するために、表示時に使用しているセグメント情報およびクラス情報をそのまま指定する。

なお、新モニタープログラムではこれらの情報を使用していないため、指定は行わない。

(6) その他の処理

PP・XY出力用形式変換を行う場合、バッチ処理時と同様に、実行ジョブ番号で取り扱える中間データファイルを作成し、このファイルへ格納しなければならない。しかし、XYコマンドを投入される都度ファイルの作成を行うようにすると、不要にファイル数が増える恐れがあり、混乱を招きかねない。そこで、GCPではファイルの作成処理と形式変換処理を分離させている。そのため、2次元GD表示プログラムおよび新モニタープログラムでは、1回目

のXYコマンドが投入された場合にのみ、形式変換プログラム処理に先立って以下の手順で中間データファイルの作成処理を行っている。

まず、出力装置をPPにするか、XYにするか、PPおよびXYにするか、またXY出力がある場合には出力時の用紙等の種類は何にするかを利用者にお問い合わせる。この指定はバッチジョブでPP・XY出力を行う場合に、PP文を使用するか、XY文を使用するか、PPXY文を使用するか等の指定およびXY使用時のSETNOパラメータの指定と同等である。2次元GD表示プログラムでは識別番号99でPP・XY出力用のファイルを作成し、指示にしたがって図形出力できるように処置する。

5.2.3 NLP出力用形式変換プログラム

NLP出力用形式変換プログラムは、GCP形式の図形データファイルとNLP変換用の制御データファイルを入力して、NLP出力用のデータに変換するためのプログラムである。

2次元GDに表示中の図形データは、PSPライブラリだけで処理することもできるが、線種を生かすために、PSPライブラリで一般ファイルに図形データを作成した後、EGRET/DRAWライブラリで変換し、スプールファイルへ出力するという方法を採用した。

このプログラムは以下の順序で処理を行っている。

(1) 制御データの読み込みと処置

制御データファイルの1番目の情報を読み込み、図形データファイルのファイル名指定があれば、アロケート処理をおこなう。そして、形式変換プログラムで使用される図形データが指定された識別番号のファイルから入力されるように処置する。

つづいて、制御データファイルの2番目以降の情報を読み込み、生成された図形データが指定されたセグメント情報、クラス情報、回転情報等によって変換され、シザリング処理されるように情報をセットする。

なお、NLPの作図領域は $0.0 \leq X \leq 26.0$ 、 $0.0 \leq Y \leq 27.0$ であるが、EGRET/DRAWでの処理の都合上、図形データの範囲は $0.0 \leq X \leq 26.0$ 、 $0.0 \leq Y \leq 19.5$ でなければならないため、指定され

たシザリング情報がこの領域外のデータを生成するようなものであれば修正される。

(2) PSP出力用ファイルの作成とアロケート

図形データの変換処理に先立って、まずPSP出力用ファイルの作成とアロケート処理を行う。GCPでは、NLP出力は識別番号97を使用することになっているので、識別番号97でPSP出力用ファイルを作成する。なお、このファイルは(5)項の処理終了後、自動消去される。

(3) 図形データの生成・データ変換処理

NLP出力用の形式変換プログラムでは、図形データファイルを使用して2次元GD表示プログラムの処理とまったく同様の方法で図形データを生成し、制御情報ファイルから入力されたセグメント情報、クラス情報、ローカル処理変換マトリックスでデータ変換している(第3.3項参照のこと)。そして第5.2.1項に述べた方法で、NLP用のデータ変換およびシザリング処理を行っている。このデータはEGRET/DRAWライブラリの入力となる。

なお、NLPで使用できる色は黒1色であるため、NEWPENサブルーチンで指定した色番号はすべて無視しているが、EGRET/DRAWライブラリではペン番号毎に8種類の線種のいずれで出力するかを指定できるため、GCPのLTYPEサブルーチンで指定された線種は、ペン番号データに置き換えて出力している。

(4) スプールファイルのアロケート

EGRET/DRAWからの出力がシステムのスプールファイルに格納されるように、スプールファイルを識別番号95でアロケートする。

(5) NLP用変換処理

(2)項で作成された図形データを入力し、1~8のペン番号の線画が8種類の線種で描かれ、さらに表示中の図形データがほぼ同じ大きさで出力されるように、線種、用紙、ファクタの設定等を行い、EGRET/DRAWで処理する。

(6) スプールファイルの解放

識別番号95のスプールファイルを解放する。

5.2.4 OPR出力用形式変換プログラム

OPR出力用形式変換プログラムは、GCP形式

の図形データファイルと変換用の制御データファイルを入力して、OPR出力用のデータに変換するためのプログラムであり、コマンドに応じて出力先を極超音速風洞建屋または原動機1号館のOPRに定義する。

なお、2次元GDに表示中の図形データをOPRへ出力するための手段はいくつかあるが、OPR用紙全面に表示データと同様のサイズで図形データを出力するために、PSPライブラリで一般ファイルに図形データを作成した後、EGRET/DRAWライブラリでOPR用の大きさに変換し、DSPRINTライブラリでOPRへ出力するという方法を採用した。

また、この方法を採用することにより、NLPの場合と同様に線種の指定を生かすことができた。

このプログラムは以下の順序で処理を行っている。

(1) 制御データの読み込みと処置

制御データファイルの1番目の情報を読み込み、図形データファイルのファイル名指定があればアロケート処理をおこなう。そして、形式変換プログラムで使用する図形データが指定された識別番号のファイルから入力されるように処置する。

つづいて、制御データファイルの2番目以降の情報を読み込み、生成された図形データが指定されたセグメント情報、クラス情報、回転情報等によって変換され、シザリング処理されるように情報をセットする。

なお、EGRET/DRAWでの処理の都合上、図形データの範囲は $0.0 \leq X \leq 26.0$ 、 $0.0 \leq Y \leq 19.5$ でなければならないため、指定されたシザリング情報がこの領域外のデータを生成するようなものであれば修正される。

(2) PSP出力用ファイルの作成とアロケート

図形データの変換処理に先立って、まずPSP出力用ファイルの作成とアロケート処理を行う。GCPでは、OPR出力は識別番号97を使用することになっているので、識別番号97でPSP出力用ファイルを作成する。なお、このファイルは(5)項処理終了後、自動消去される。

(3) 図形データの生成・データ変換処理

OPR用の形式変換プログラムでは、図形データ

ファイルと制御情報ファイルを使用して、NLPの場合とまったく同様の方法で、図形データの生成からシザリングまでの処理を行っている。このデータはEGRET/DRAWライブラリの入力となる。

なお、OPRで使用できる色は黒1色であるため、NEWPENサブルーチンで指定した色番号はすべて無視しているが、NLP出力の場合と同様に、GCPのLTYPEサブルーチンで指定された線種はペン番号データに置き換えて出力している。

(4) EGRET/DRAW出力ファイルの作成とアロケート

EGRET/DRAW出力用ファイルを、識別番号95で作成する。このファイルは(6)項の処理終了後、自動消去される。

(5) OPR用変換処理

(2)項で作成された図形データを入力し、1~8のペン番号の線画が8種類の線種で描かれ、さらに表示中の図形データとほぼ同様の大きさに出力されるように、線種、用紙、ファクタの設定等を行い、EGRET/DRAWで処理する。

(6) OPR装置への出力

(4)項で作成された図形データを、DSPRINTライブラリを使用してOPRへ出力する。このとき、コマンドに応じて極超音速風洞建屋または原動機1号館のOPRへ出力されるような処置をする。

5.2.5 PP・XY出力用形式変換プログラム

PP・XY出力用形式変換プログラムは、GCP形式の図形データファイルと変換用の制御データファイルを入力して、PP・XY出力用のデータに変換するためのプログラムであり、カルコンプライブラリで処理をしている。なお、この処理に先立ち、PP・XY出力用のファイルがファイル識別番号99でアロケートされているものとする。

このプログラムは以下の順序で処理を行っている。

(1) 制御データの読み込みと処置

制御データファイルの1番目の情報を読み込み、図形データファイルのファイル名指定があれば、アロケート処理をおこなう。そして、形式変換プログラムで使用する図形データが指定された識別番号のファイルから入力されるように処置する。

つづいて、制御データファイルの2番目以降の情報を読み込み、生成された図形データが指定されたセグメント情報、クラス情報、回転情報等によって変換され、シザリング処理されるように情報をセットする。

なお、PP・XY出力時の用紙は、もっとも幅の狭いオープンのXYを想定した場合にはY方向のみ $0.0 \leq Y \leq 27.0$ としなければならないため、指定されたシザリング情報がこの領域外のデータを生成するようなものであれば修正される。また、当プログラムで作成されたPP・XY用の図形データをモニタ処理した場合でも、2次元GDでの表示色で表示できるように、色変換情報をセットする。

(2) 図形データの生成・データ変換処理

PP・XY出力用の形式変換プログラムでは、図形データファイルを使用して、NLPの場合と同様の方法で図形データの生成からシザリングまでの処理を行っている。このデータをPP・XYに出力するために、XY出力用のカルコンプライブラリを使用している。

なお、図形データファイルの中に色番号の情報が格納されていた場合には、(1)項でセットした色変換情報にもとずいて色番号を変換する。

(3) 原点移動と改ページ処理

PP・XY出力用の形式変換プログラムでは、画面に表示中の図形データの変換処理がすべて終了したときに、図1枚分として、X方向に30.0cmの原点移動を行う。これは、この後出力される図形データが重ね書きされないための処理である。さらに、XYコマンドで出力された図毎にモニタ処理できるように、PAGEサブルーチン呼び出す。

6. おわりに

NSシステムの導入から約1年が経過し、導入と同時に開始したGCPライブラリの組込み作業および機能追加の作業もようやく一段落できるような状況になった。

NSシステムでは、スーパーコンピュータの有効利用とともに、計算結果の検討・確認を行うための

図形処理装置の有効利用は非常に重要な課題であるが、XYプロッタやプリンタプロッターしか使用していなかった大多数のユーザが、新システムのグラフィックディスプレイ装置にうまく移行できるか否かはひとえにどのようなライブラリを用意するかにかかっていた。

NSシステム導入時には、GCPライブラリはユーザが困らない最低限度の機能しかなかったが、疑似カルコンプライブラリが用意されており、さらに出力装置に依存しないで作図プログラムを作成できるようにもなっていたので、大きなトラブルもなく移行できた。

その後、順次ライブラリ機能の追加、装置間のデータ形式変換プログラムの作成およびユーザインタフェース部分の改良を重ねて今に至っている。

今後、NSシステムの各種の図形処理装置をより使いやすくし、NSジョブの前処理(格子形成等)および後処理(計算結果の可視化による検証技術の確立等)において、真に活用されうるようにするためには、GCPに汎用プログラム機能の追加を行い、TSS処理だけでなくバッチ処理でも容易に利用可能なものにしていかなければならない。

また、大半が大型計算機システムで処理されている図形処理も、近い将来には高速グラフィックス機能を有するワークステーションによる分散処理の方向へ進んでいくものと予想されるため、ワークステーションによる図形処理とその中央計算機との相互接続方式の研究が今後の重要な課題になると考える。

おわりに、ライブラリのインタフェース部分の改良においては多くのユーザの方から貴重な御意見をいただいた。末筆ながら、この場を借りて感謝の意を表する。

参 考 資 料

- 1) 末松和代, 磯部俊夫; 3次元グラフィックディスプレイ装置用ソフトウェアパッケージ, 航技研資料 TM-559, 1986年12月
- 2) 富士通㈱; FACOM OSN GSP文法書

航空宇宙技術研究所資料589号

昭和63年6月発行

発行所 航空宇宙技術研究所
東京都調布市深大寺東町7-44-1
電話 三鷹(0422)47-5911(大代表) 〒182
印刷所 株式会社実業公報社
東京都千代田区九段南4-2-12
