

ISSN 0452-2982
UDC 681.326.3:
681.327.12

航空宇宙技術研究所資料

TECHNICAL MEMORANDUM OF NATIONAL AEROSPACE LABORATORY

TM-560

ラスタスキャン型ディスプレイのための基本プログラム

磯部俊夫・末松俊二・小口慶子

1986年12月

航空宇宙技術研究所
NATIONAL AEROSPACE LABORATORY

ラスタスキャン型ディスプレイのための基本プログラム*

磯 部 俊 夫** 末 松 俊 二** 小 口 慶 子**

1. ま え が き

最近の科学技術用コンピュータ・システムでは、グラフィック・ディスプレイは必須の出力装置となっている。航空宇宙技術研究所計算センター（以後航技研計算センターと記す）のコンピュータ・システムにも種類の異なったグラフィック・ディスプレイが付属している。これらのグラフィック・ディスプレイを利用形態から分類してみると次のようになる。

1. X-Yプロッタ
2. 2次元データ表示ディスプレイ
3. 3次元データ表示ディスプレイ
4. ラスタスキャン型ディスプレイ

X-Yプロッタは紙の上にペンで線を引ながら図形を描いていく図形出力装置で、航技研計算センターでは最も古くから利用されている。作図は、ペンの現在位置からX-Yプロッタに与えられた点へ向って、直線を引くという操作を基本に行われる。

2次元データ表示ディスプレイは、X-Yプロッタの機能をCRTディスプレイ上で実現する装置としてみることができる。プログラミングの考え方はX-Yプロッタの場合と同じように考えてもさしつかえない。

3次元データ表示ディスプレイは、扱えるデータを2次元データから3次元データに拡張したものである。3次元データを効果的に表示するため座標変換等の機能をディスプレイ装置内にもっているなどの特徴はあるが、基本的には、与えられた2つの3次元座標間を線で結び、それをディスプレイ上に表示するものである。

以上のように、1.から3.までの図形出力装置では、プログラムが装置に与える2端点の座標間を

線分で結び表示することを基本としている。

これに対して、4.のラスタスキャン型ディスプレイでは、プログラムからディスプレイへ送られるデータが1.～3.のタイプのものと異なる。ラスタスキャン型ディスプレイは、CRT上の画面を構成する各画素と一対一に対応するフレッシュメモリをもつ。CRT上に表示される図形は、リフレッシュメモリの内容がそのまま投影されたものとなる。したがってラスタスキャン型ディスプレイに図形を描くためには、画面を構成する全画素のデータをコンピュータで作り、そのデータをディスプレイ内のリフレッシュメモリに転速することが必要である。

このように、ラスタスキャン型ディスプレイでは、ディスプレイ上に表示される画素一点ずつのデータをコンピュータにより指示できるため、通常われわれが見ている写真やテレビジョン画像のようなリアルな画像をも作り出すことが可能となる。また地球観測衛星等により得られる観測データ、2次元走査型温度計で代表される2次元計測器からの測定データ、ドラムスキャナなどによる画像入力装置の出力データ等を直接あるいはコンピュータ処理を施して、ディスプレイ上に表示することができる。

このたび計算センターにラスタスキャン型のディスプレイが導入され、大型コンピュータと接続された。航技研では、この型のディスプレイは既に実験設備の計測システム・コンポーネントとしていくつか導入されており、またそれらのシステムで使用されるソフトウェア¹⁾も開発されている。しかし、計算センターではこの種のディスプレイは初めての経験であり、使用目的・ソフトウェアの考え方も計測用のそれとは異なる。

今回導入したディスプレイ装置は、画素数512×480、各画素に対して赤・緑・青各8ビット計24ビットのリフレッシュメモリをもつ。すなわち各画素において1600万（ $\approx 2^{24}$ ）色のフルカラー表示が

* 昭和61年11月5日受付

** 計算センタ

できるものである。このディスプレイ用の基本的な表示プログラムを作成した。作成したプログラムは、すべてFORTRANサブルーチンの形成を採用している。各サブルーチンはできるだけ一般化し、なるべく使用ディスプレイ装置に依存しないような形でまとめた。電子機器の技術進歩が著しい現時点では、現在使用の装置より高性能のものが数年以内に導入される可能性が大きい。高性能ディスプレイへの代替えあるいは追加に対しても、ソフトウェアは最少限の手直しですむように考慮してある。

作成ルーチンは次のように分類することができる。

- 1) 装置固有のサブルーチン
- 2) 一般的なラスタスキャン用サブルーチン
- 3) 応用サブルーチン

1)のサブルーチンは、2)のサブルーチンと装置を結合するためのインタフェース・プログラムあるいは装置固有の機能を実現するためのものである。2)のサブルーチンは、リフレッシュメモリへ転送するためのデータを作成するためのものである。このデータ領域は、プログラム上では1次元配列に割り当てられている。1次元配列の大きさはリフレッシュメモリの大きさと等しい。3)のサブルーチンは、ある与えられた入力データに対し

ある特定の表現方式でディスプレイ上に表示するためのデータを生成するためのものである。このサブルーチン利用者は、使用ディスプレイのごく基本的な知識だけで、計算データをディスプレイ上に表示することができる。

本資料は、今回導入されたラスタスキャン型ディスプレイのハードウェア構成とこのディスプレイ用に作成した大型コンピュータ用サブルーチンについて述べる。また付録としてシステム全体の操作方法、各サブルーチンの利用法をまとめ、ディスプレイ利用者の便を計るものである。

2. ラスタスキャン型ディスプレイ装置

2.1 システム構成

全体のシステム構成図を図1に示す。

システムはイメージプロセッサ (nexus 6410)を中心に構成されている。イメージプロセッサは、単独では動作することができず、ホスト・コンピュータなどから与えられるコマンドにより動作する。本システムのコマンド源は次の3つである。

- (1) マイクロコンピュータ
- (2) デジタイザ
- (3) 大型コンピュータ

電源投入時、イメージプロセッサのリモートスイッチを押した時、コマンド源はマイクロコンピュ

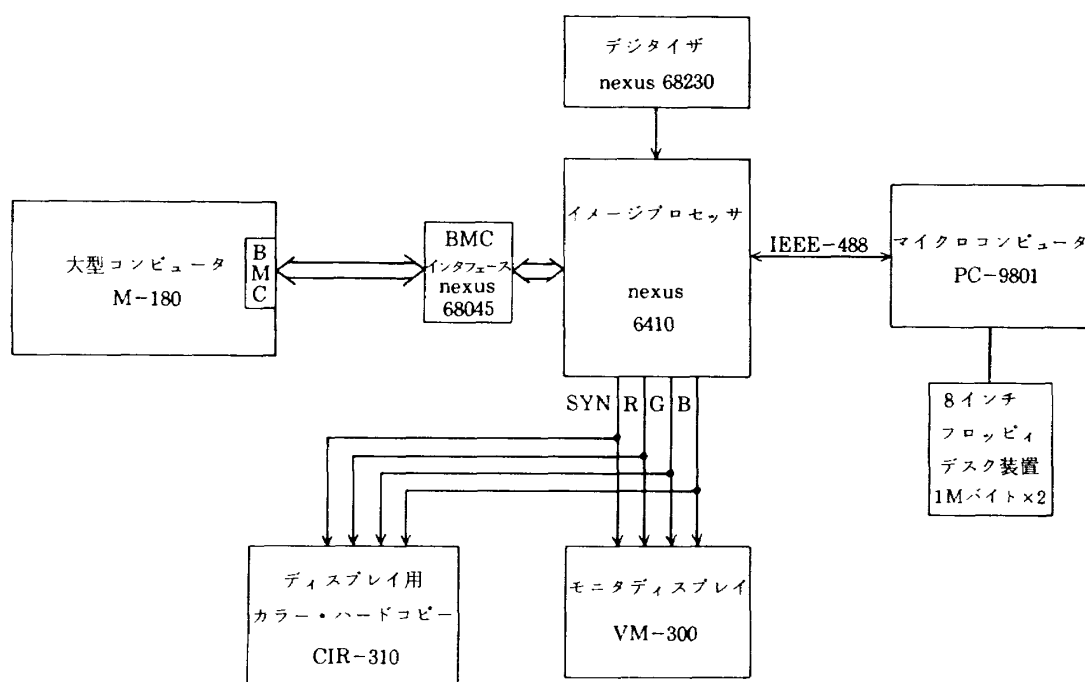


図1 システムの構成図

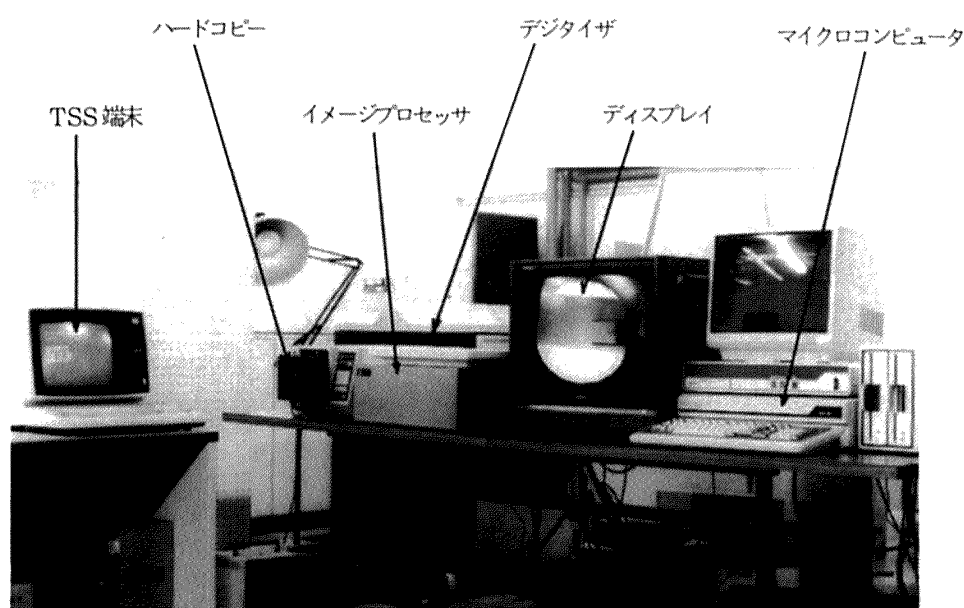


図2 ディスプレイ装置概観

ータとなる。動作中のコマンド源の変更は、コマンドにより実行される。

イメージプロセッサとマイクロコンピュータはIEEE-488インタフェースにより接続されている。また大型コンピュータとはチャンネルにより接続されている。

装置全体の写真を図2に示す。

2.2 イメージプロセッサ

イメージプロセッサ²⁾は、赤(R)・緑(G)・青(B)各色8ビット長データ、画素数512×512のリフレッシュメモリをもつプロセッサで、外部コンピュータとのインタフェース回路とR・G・Bアナログ信号および同期信号(SYN)の出力端子を備えている。主な仕様を表1に示す。

イメージプロセッサの扱う表示空間は、次の3つである。

- (1) メイン・イメージ空間(512×480)
- (2) サブ・イメージ空間(32×480)
- (3) キャラクタ空間(77字×30行)

ディスプレイ画面上におけるこれら空間の関係を図3に示す。

2.3 モニタ・ディスプレイ

イメージプロセッサからの映像出力信号、R・G

表1 イメージプロセッサの主な仕様

| 形 式 | nexus 6410 |
|---------------|---|
| 分 解 能 | 512×480ドット |
| カ ラ ー 表 示 能 力 | 1600万色同時表現可能(R, G, B, 各8ビット) |
| リフレッシュ周波数 | 水平方向 15,734 KHz 垂直方向 29.9 Hz NTSC規格 |
| リフレッシュメモリー | 512×512×8ビットのメモリー4枚 |
| 画 素 縦 横 比 | 1:1 |
| キ ャ ラ ク タ 表 示 | 77字×30行 文字数 256種 |
| ビ デ オ 出 力 端 子 | BNC端子にてRED, GREEN, BLUE, SYNC RED出力: 0.7V _{p-p} 75Ω インピーダンス 正極性 GREEN出力: " BLUE 出力: " SYNC出力: 2V _{p-p} 75Ω インピーダンス 負極性 |

・BおよびSYNCの各信号はモニタ・ディスプレイに入力され、画像化される。モニタ・ディスプレイの主な仕様を表2に示す。

2.4 カラー・ハードコピー

イメージプロセッサからの映像信号は、カラー

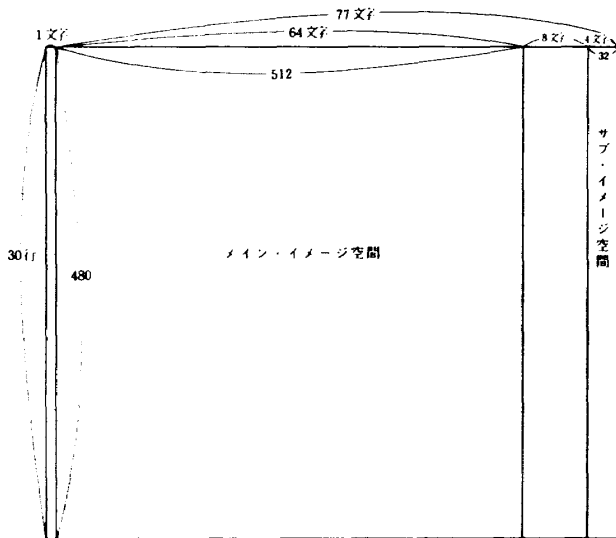


図3 イメージプロセッサの表示空間

・ハードコピー装置にも入力される。この装置はモニタ・ディスプレイ画面上に表示された画像のハードコピーを写真で得るためのものである。1画像のコピーに約15秒を要する。このハード・コピー装置で使用できるフィルムは、35mmロールフィルムとフジ・フォトラマ(6.8cm×9.1cm)である。カラー・ハードコピーの主な仕様を表3に示す。

2.5 マイクロコンピュータ

イメージプロセッサの最も優先度の高いコマンド源として、マイクロコンピュータが使用されている。マイクロコンピュータには、1Mバイトの8インチ・フロッピー・ディスク装置2台が付属している。このマイクロコンピュータはBASIC等の高級言語でプログラムを作ることができるもので、イメージプロセッサの完全なホストコンピュータとしての動作を行う。マイクロコンピュータの主な仕様を表4に示す。

2.6 デジタイザ

デジタイザ³⁾はイメージプロセッサ付属の装置で、ディスプレイ画面上の位置の指定、図形の入力などの目的に使用される。デジタイザの入力面は約30cm×30cmの大きさを持ち、モニタディスプレイ画面と一対一の対応付がされている。コマンド源として、このデジタイザが指定されると、デジタイザ付属のペンの指示するデジタイザ面上

表2 モニタ・ディスプレイの主な仕様

| | |
|-------------|--------------------------|
| 形 式 | VM-300 |
| ブラウン管の大きさ | 20型(幅40.4cm,高さ30.3cm) |
| 解 像 度 | 350本以上 |
| 信 号 入 力 | RGB信号/0.7Vp-p,75Ω 正極性 |
| 外 部 同 期 入 力 | 1~4Vp-p,75Ω,負極性 |
| 入 力 端 子 形 状 | RGB,同期ともにBNC |

表3 カラー・ハードコピーの主な仕様

| | |
|---------------|------------------------|
| 形 式 | CIR-310 |
| カ ラ ー 方 式 | R・G・B加色方式 |
| 信 号 入 力 | RGB信号/0.35~1.4Vp-p,正極性 |
| 外 部 同 期 入 力 | 0.5~4.0Vp-p,負極性 |
| 入 力 端 子 形 状 | RGB,同期ともにBNC |
| 水 平 同 期 周 波 数 | 15.75kHz,2:1インタレース |
| 垂 直 同 期 周 波 数 | 50~60Hz |
| 露 光 時 間 | 約15秒 |

表4 マイクロコンピュータの主な仕様

| | |
|---------------|--|
| 形 式 | PC-9801 |
| C P U | μPD8086 16ビット マイクロプロセッサ |
| ク ロ ッ ク | 5 MHz |
| R A M | 256Kバイト,他にVRAM 96Kバイト |
| R O M | 96Kバイト(モニタおよび N88-BASIC) |
| プ ロ グ ラ ム 言 語 | N88-BASIC |
| 表 示 機 能 | 表示文字 英数カナ80×25 漢 字40×20 グラフィック640×400 8色 |
| イ ン タ フ ェ ース | RS-232C,IEEE-488 |
| 外 部 記 憶 装 置 | 8インチ標準フロッピー 1Mバイト×2 |

の位置に相当するモニタディスプレイ画面上に、十字のカーソル・マークが表示される。

2.7 ホストコンピュータとのインタフェース

イメージプロセッサとマイクロコンピュータとは、IEEE-488-1975の規格に基づくインタフェースにより接続されている。このインタフェース⁴⁾

により、ホストコンピュータとイメージプロセッサ間でバイナリーコード画面情報を最大100K バイト / 秒で転送することができる。

イメージプロセッサ側のIEEE-488インタフェースはスレーブモードで定義されている。マイアドレス (MY ADDRESS) は1, デリミタ (ターミネータ) はCR/LFに設定されている。なおこのマイアドレス, デリミタはディップスイッチにより変更できるようになっている。

イメージプロセッサと大型コンピュータとは, BMC (ブロック・マルチプレクサ・チャネル) インタフェース⁵⁾を介して接続する。このインタフェースは中央コンピュータのBMCに直接接続されているため, 高速のデータ転送が可能である。このインタフェースの最大転送速度は1Mバイト / 秒となっている。ディスプレイ画面全体の画像データを1秒以下で転送することが可能である。

3. 基本サブルーチン

イメージプロセッサは, コマンド源から与えられたコマンドに従いリフレッシュメモリのデータを処理し, 画像信号を出力する。コマンド源がコンピュータの場合 (本システムでは大型コンピュータとマイクロコンピュータ), リフレッシュメモリへのデータ書き出しあるいはリフレッシュメモリからのデータ読み出しも行える。

ホストコンピュータからイメージプロセッサを制御し, 画像をモニタディスプレイ上に表示させるためには, ホストコンピュータとイメージプロセッサを結ぶインタフェースを介して, 次のデータを転送する。

- (1) コマンド (文字コード列)
- (2) イメージ・データ (バイナリ・データ列)

利用者の立場からすると, インタフェースを介してコマンドやイメージ・データを直接イメージプロセッサに転送あるいは受信するのはやっていないことである。実際にディスプレイを有効に使用しこなすためには, 利用者にとって使いやすいプログラムが必要である。なお, イメージプロセッサのメーカーから, 次のようなプログラムの提供を受けている。

- (1) GPIB・OBJ

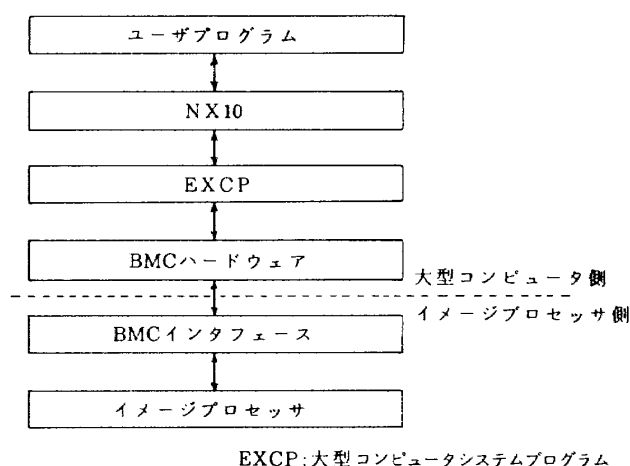


図4 ソフトウェアの構造

(2) GPIB・BAS⁵⁾

(3) NX10⁶⁾

(1)はIEEE-488インタフェースのハンドラー・プログラムで, マイクロコンピュータのマシン語で書かれている。(2)は(1)のハンドラーを用いたサンプルプログラムで, BASICで書かれている。(3)はBMCインタフェースのハンドラー・プログラムで, 大型コンピュータ内で使用される。形式はFORTRANプログラムから呼び出すことができるサブルーチン形式になっている。このNX10を用いたソフトウェア構造を図4に示す。

この資料では, 図4のユーザプログラムに関する部分のプログラムについて述べる。

3.1 NX10の概要†

NX10は, 次に示す10のサブルーチンから構成されている。

(1) サブルーチン NXOPEN

イメージプロセッサへの入出力を可能とするための準備を行う。イメージプロセッサを使用する最初にCALLすることが必要。

(2) サブルーチン NXCLOS

イメージプロセッサのクローズ処理を行う。

(3) サブルーチン NXRSET

イメージプロセッサをリセットする。なお本システムでは, このサブルーチンをCALLするとホ

注† 付録2参照

ストコンピュータは大型コンピュータからマイクロコンピュータに切換えられてしまう。

(4) サブルーチン NXBGET

イメージプロセッサからのバイナリ・データを読み込む。

(5) サブルーチン NXBPUT

バイナリ・データをイメージプロセッサへ出力する。

(6) サブルーチン NXCGET

イメージプロセッサからのコマンド・データを読み込む。

(7) サブルーチン NXCPUT

コマンド・データをイメージプロセッサへ出力する。

(8) サブルーチン NXEBAS

コマンド・データ出力時の文字コード変換テーブルを変換する。この文字コード変換テーブルのデフォルトは EBCDIC コード→ASCII コードである。

(9) サブルーチン NXASEB

コマンド・データ受信時の文字コード変換テーブルを変換する。この文字コード変換テーブルのデフォルトは ASCII コード→EBCDIC コードである。

(10) サブルーチン NXTOUT

入出力に要する監視タイマ値を変換する。タイマ値のデフォルト値は 3 分である。

3.2 リフレッシュメモリの構造

NX10のサブルーチンを利用して、ラスタスキャン型ディスプレイ用のプログラムを作成するのであるが、ここで作るプログラムは、ディスプレイ装置になるべく依存しないような形式とする。すなわちイメージプロセッサ nexus 6410はいろいろな描画機能をもつが、これら描画機能を使用せずに、ディスプレイ上に画像を描かせるようにする。イメージプロセッサの最も基本的な機能は、リフレッシュメモリの内容をそのままモニタディスプレイ用入力信号に変換することである。本プログラムでは、ディスプレイ内のリフレッシュメモリに相当するものを大型コンピュータ内に用意し、すべての描画を大型コンピュータで実行し、描画

が完成した時点でイメージデータをリフレッシュメモリへ転送して、ディスプレイ上に表示するという方式をとる。このような方式にすると、イメージプロセッサは単に大型コンピュータから送られてくるデータをモニタディスプレイに表示するだけのものとなる。この基本機能はどのようなラスタスキャン型のディスプレイでも備えているもので、プログラムをディスプレイ装置に依存する部分の少ないものとして行うことができる。一方イメージプロセッサのもつ描画機能を利用しないことにもなるが、ホストコンピュータの処理能力がイメージプロセッサの処理能力より高く、かつ両者間のデータ転送が高速で行われる場合には、描画処理をホストコンピュータで行う方が全体の処理能力は向上する。このことは実験を行って確かめた。

この方式にすると、ホストコンピュータ側でリフレッシュメモリに相当するイメージデータ領域を確保する必要がある。リフレッシュメモリの構造は、一般に FORTRAN でいう 2 次元配列の形式となっているが、各画素のデータ長、座標の付け方は個々のディスプレイ装置で異なる。

イメージプロセッサのリフレッシュメモリの構造を図 5 に示す。リフレッシュメモリの大きさは 512×480 で、1 画素は赤 (R)・緑 (G)・青 (B) の各 8 ビット計 24 ビット (3 バイト) となっている。(実際には R・G・B のメモリのほかに、ワーク用の $512 \times 480 \times 8$ ビットのメモリがあるが、本資料ではこのメモリは使用しないので省略する)。したがって、全画面のデータ量は

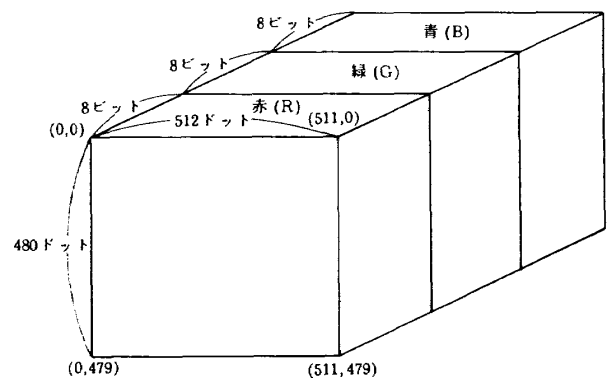


図5 リフレッシュメモリ (nexus 6410) の構造

737,280(=512×480×3)バイト=184,320 語となる。ただし、ここで1語は4バイトである。

イメージプロセッサでは、リフレッシュメモリ全体およびR,G,B単位で扱えるようになっている。各画素の位置を示す座標の原点は、モニタディスプレイ上で左上隅となっている。X方向はディスプレイ画面の左から右へ向う水平方向であり、Y方向は上から下へ向う垂直方向である。なおこの座標の定義はラスタスキャン型ディスプレイの一般的なものである。

ホストコンピュータとイメージプロセッサ間でのイメージデータ転送時のデータの並びは、X方向のデータをY座標の順に配列したものとなっている。すなわち画素の位置を(X,Y)で表示すると
(0,0), (1,0), (2,0) ---- (511,0)
(0,1), (1,1), (2,1) ---- (511,1)

(0,479), (1,479), (2,479) ----- (511,479)
の順となる。R・G・B全体を同時に転送する場合は各画素でRGBの順にデータを並べる。

ホストコンピュータ内でのイメージデータは、大きさ184,320語の1次元配列IBUFに格納する。このときのIBUF内でのイメージデータの配列は次のようになっている。

| | | | | | |
|-----------|-----------|-----------|-----------|-----------|-------|
| (0,0) | (1,0) | (2,0) | (3,0) | (9,0) | |
| RGB | RGB | RGB | RGB | RGB | |
| IBUF(1) | IBUF(2) | IBUF(3) | IBUF(4) | | |
| | | | | | |
| (507,479) | (508,479) | (509,479) | (510,479) | (511,479) | |
| RGB | RGB | RGB | RGB | RGB | |
| IBUF | IBUF | IBUF | IBUF | | |
| (184317) | (184318) | (184319) | (184320) | | |

3.3 装置固有サブルーチン

イメージプロセッサ(nexus 6410)を使用するためのサブルーチンをここでは装置固有サブルーチンと呼ぶ。ここでのサブルーチン群は、次の3.4で述べるサブルーチンなどで作られるイメージデータとディスプレイをむすぶインタフェース・サブルーチンにもなる。

ここで示すサブルーチンからディスプレイへのアクセスは、NX10サブルーチンによって行われ

る。ディスプレイ利用者は、直接NX10サブルーチンを用いてディスプレイ上にイメージデータを表示することができるが、この場合イメージプロセッサのコマンドとその機能を理解する必要がある。

ここで定義するサブルーチン群は、次の12の機能をもつ。

- (1) ラスタスキャン型ディスプレイ装置のオープン処理を行う。
- (2) ラスタスキャン型ディスプレイ装置のクローズ処理を行う。
- (3) ディスプレイ内のリフレッシュメモリをクリアする。この処理によりリフレッシュメモリの内容は0になり、ディスプレイ画面はクリアされる。なおホストコンピュータからディスプレイのリフレッシュメモリへ転送されたデータは、ホストコンピュータとディスプレイが切り離された後も、リフレッシュメモリ内にそのまま保存されている。
- (4) ホストコンピュータ内のイメージデータIBUFをディスプレイ装置のリフレッシュメモリへ転送する。イメージデータIBUFは、具体的には184,320語の1次元配列であり、ディスプレイ上のR・G・B各色のデータをもつ全画面分のデータである。
- (5) イメージデータ領域に画素単位で描画する。すなわちディスプレイ画面上の座標(IX,IY)に相当するIBUF領域へ、R・G・Bのデータを書き込む。
IX,IYの値の範囲は、
 $0 \leq IX \leq 511, 0 \leq IY \leq 479$
であり、R・G・Bのデータは、0～255の範囲の整数値で、ディスプレイ上でのR・G・Bの輝度は整数値に比例して強くなる。
- (6) ディスプレイ内のリフレッシュメモリの内容を読み出す。リフレッシュメモリの全データが184,320語のイメージデータ領域にあたる1次元配列に読み出される。
- (7) イメージデータ内の特定画素R・G・Bのデータを読み出す。画素の指定はディスプレイ画面上の座標(IX,IY)で行う。
- (8) 大きさあるいは色の限定されているイメー

ジデータをリフレッシュメモリの指定した場所に転送する。イメージデータの大きさは、パラメータNX,NYで指定する。ここでNX,NYはX方向,Y方向の画素数である。色の指定は、パラメータMRで行い、次のように定義する。

- MR=0 イメージデータの色は赤
 MR=1 イメージデータの色は緑
 MR=2 イメージデータの色は青
 MR=3 イメージデータの色は赤・緑・青のフルカラー

したがってMR ≤ 2のときイメージデータの大きさはNX×NYバイト, MR=3のときイメージデータの大きさは3×NX×NYバイトとなる。この大きさのイメージデータをディスプレイ画面上の座標(IXO,IYO)からX方向にNX画素,Y方向にNY画素の範囲に相当するリフレッシュメモリへ転送する。イメージデータとディスプレイ画面上の表示範囲の関係を図6に示す。

- (9) リフレッシュメモリの特定範囲のデータをイメージデータ領域にあたる1次元配列に読み取る。読み出すリフレッシュメモリの範囲は、パラメータIXO,IYO,NX,NY,MRで指定する。これらのパラメータの意味は機能(8)のそれと同じであり、図6に示した通りである。
- (10) ディスプレイ上のサブエリアにカラーバーを表示する。サブエリアのY座標IYOからY方向にNY画素に渡り、1次元配列で指定された値に対応する色を塗る。色指定のための1次元配列の大きさは、NYである。配列内の値と実際の色の関係は、3.4で述べる擬似カラー関数

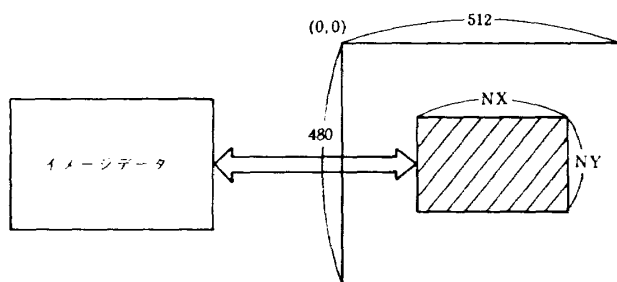


図6 イメージデータとディスプレイ画面上の表示範囲

による。また既にサブエリアに表示されているカラーバーをクリアする機能をもつ。

- (11) キャラクタ空間にキャラクタを表示するための準備を行う。具体的には表示キャラクタの色を指定する。ここで指定する色はキャラクタ空間に表示される全キャラクタに共通なもので、個々のキャラクタに対する色指定はできない。
- (12) キャラクタを表示する。この機能は、機能(11)を実行後に有効となる。キャラクタ表示のためのパラメータはCHR, IX, IY, LNである。CHRは表示キャラクタを定義する文字変数である。キャラクタ数は最大77である。IX, IYは表示キャラクタの書き出し位置を指定するキャラクタ座標(IX,IY)を示す。IX, IYの範囲は

$$0 \leq IX \leq 76, \quad 0 \leq IY \leq 29$$

である。LNは表示キャラクタ数であり、77以下の自然数である。書き終りの位置が29行のときは、キャラクタ表示後、表示キャラクタ全体が1行スクロール・アップする。また既に表示されているキャラクタすべてをクリアする機能をもつ。

以上12の機能をもつサブルーチンを用意する。

表5 装置固有サブルーチン

| 機能番号 | サブルーチン名 | 内 容 |
|------|---------|--------------------------------|
| 1 | NXOPN | オープン処理 |
| 2 | NXCLS | クローズ処理 |
| 3 | NXCLR | リフレッシュメモリのクリア |
| 4 | NXWRT | イメージデータをリフレッシュメモリへ転送 |
| 5 | NXPUT | イメージデータへの描画 |
| 6 | NXRD | リフレッシュメモリの内容をイメージデータ領域へ転送 |
| 7 | NXRDP | イメージデータからの読出し |
| 8 | NXWT2 | イメージデータをリフレッシュメモリの特定領域へ転送 |
| 9 | NXRD2 | リフレッシュメモリの特定領域の内容をイメージデータ領域へ転送 |
| 10 | NXSBOX | サブエリアにカラーバー表示 |
| 11 | NXPRO | キャラクタの色指定 |
| 12 | NXPRT | キャラクタの表示 |

各機能はそれぞれ1つのサブルーチンとしてまとめる。

表5にそれぞれの機能とそのサブルーチン名を示す。また図7は各サブルーチンの機能を図で示したものである。

3.4 汎用ラスタスキャン用サブルーチン

一般的なラスタスキャン型ディスプレイは、図8に示すように、リフレッシュメモリ、ルックアップテーブル(Look-Up-Table:LUT)、D/A変換器、CRTで構成される。フルカラーの表示できるディスプレイのリフレッシュメモリは、図5で示

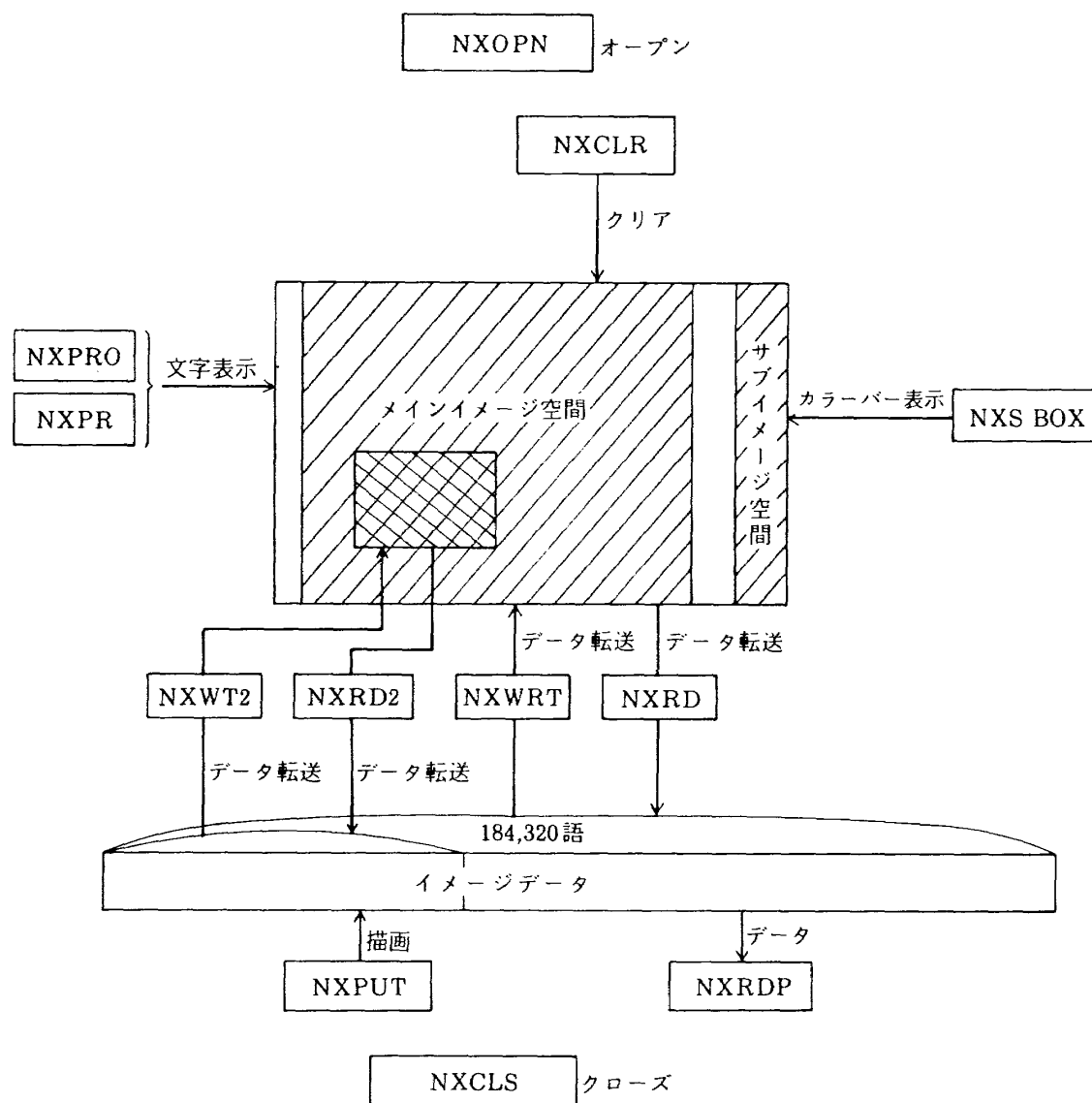


図7 サブルーチンの機能

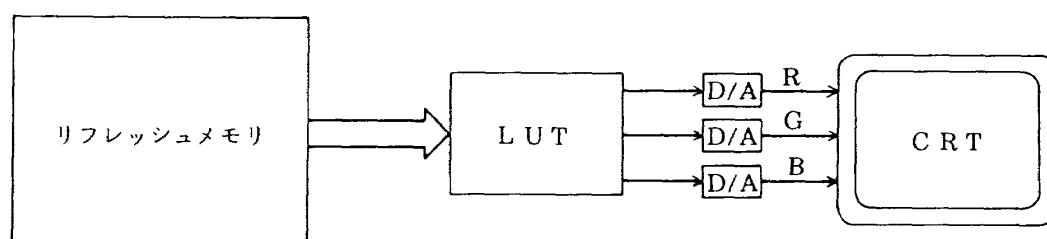


図8 ラスタスキャン型ディスプレイの構成

したように、R・G・Bの3色に対し各8ビットのメモリ構成になっているものが多いが、8ビットより少ないビット構成になっているものもある。マイクロコンピュータなどで多く使用されている簡易型のカラーディスプレイではR・G・B各1ビット構成のものもある。またカラーマップ・ディスプレイと呼ばれる型式のものでは、リフレッシュメモリに書き込まれるデータは、R・G・Bの各色の輝度ではなく、ディスプレイに表示可能な色の全種類に付けられた色番号である。ディスプレイに表示される実際の色は、リフレッシュメモリから与えられる色番号に従いルックアップテーブルから読み出されるR・G・Bの輝度データにより与えられる。この型のディスプレイではリフレッシュメモリは1画素に対して1要素のデータだけでよい。

したがって、一般化したリフレッシュメモリの構造は、次の4つのパラメータで定義できる。

- | | |
|-----------------|-----------|
| (1) 横方向の画素数 | IWD |
| (2) 縦方向の画素数 | ILN |
| (3) 1画素のデータ長 | NBIT(ビット) |
| (4) 1画素のデータの要素数 | NCMP |

ここで、NBITはNCMPの倍数で、1画素における1要素のデータ長はNBIT/NCMPとする。この構造のリフレッシュメモリの大きさは

$$IWD \cdot ILN \cdot NBIT \text{ (ビット)}$$

である。

このリフレッシュメモリに相当する領域をプログラム内に用意する。この領域は1次元配列IBUFで定義される。大きさはもちろんリフレッシュメモリと同じ大きさである。

プログラムでは、定義された1次元配列に図形データを描画する。描画は1画素ずつ行われる。

ディスプレイ上の座標とプログラム中でのデータ座標の対応付けは、図9に示すように8通りの方法が考えられる。

1画素のデータ数は要素数NCMPで与えられる。フルカラーディスプレイの場合、NCMPは3であり、R・G・Bの3つの輝度データである。数値計算の結果などを色により表現するとき、スカラー量で色が決まれば便利ことが多い[†]。ここでは

各要素のデータをスカラー量で定義できる関数を用意する。

ディスプレイ上に任意の大きさのビューポートが定義でき、このビューポートはプログラム上の座標系(実数値により定義)で定義されるウィンドウと対応する。

以上一般化したリフレッシュメモリの概念を図10に示す。一般化したリフレッシュメモリを操作するためのサブルーチン群は、次の8つの機能をもつ。

- (1) リフレッシュメモリの定義を行う。リフレッシュメモリのX方向、Y方向の画素数IWD, ILN, 1画素当りのデータ長NBIT(ビット数で表現)、データ要素数NCMPを定義する。

ここで定義する各パラメータは、必ずしも実際のディスプレイ中にあるリフレッシュメモリの構造と一致している必要はないが、リフレッシュメモリと意味のあるデータ転送が可能ないように定める必要がある。例えばnexus 6410の場合、サブルーチンIMGWT2, IMGRD2を用いてデータ転送を行うとするとIWD, ILNは

$$1 \leq IWD \leq 512, 1 \leq ILN \leq 480$$

の範囲で任意に選べる。またNBITは、8(このときNCMP=1)あるいは24(NCMP=3)の選択が可能である。

- (2) リフレッシュメモリに対応するイメージデータ領域の定義を行う。この領域は整数型1次元配列により確保される。描画機能により実際に描画される領域は、ここで定義された1次元配列である。
- (3) イメージメモリに描画する際の配列順序を指定する。指定はパラメータICTによりなされ、ICTの値と配列順序は図9に示した通りである。すなわち描画時の座標系は、ディスプレイ表示の座標系に対して8つの異なる座標系から選択された1つのものとなる。特にICTが奇数時、描画時のX, Y座標軸はディスプレイ表示座標系では縦方向、横方向となるので注意を要する。
- (4) イメージデータ領域にデータを描画する。描画する画素位置(IX, IY)に1画素を構成する各

注[†] スカラー量を色で表現するのを模擬カラー表示と一般に称している。

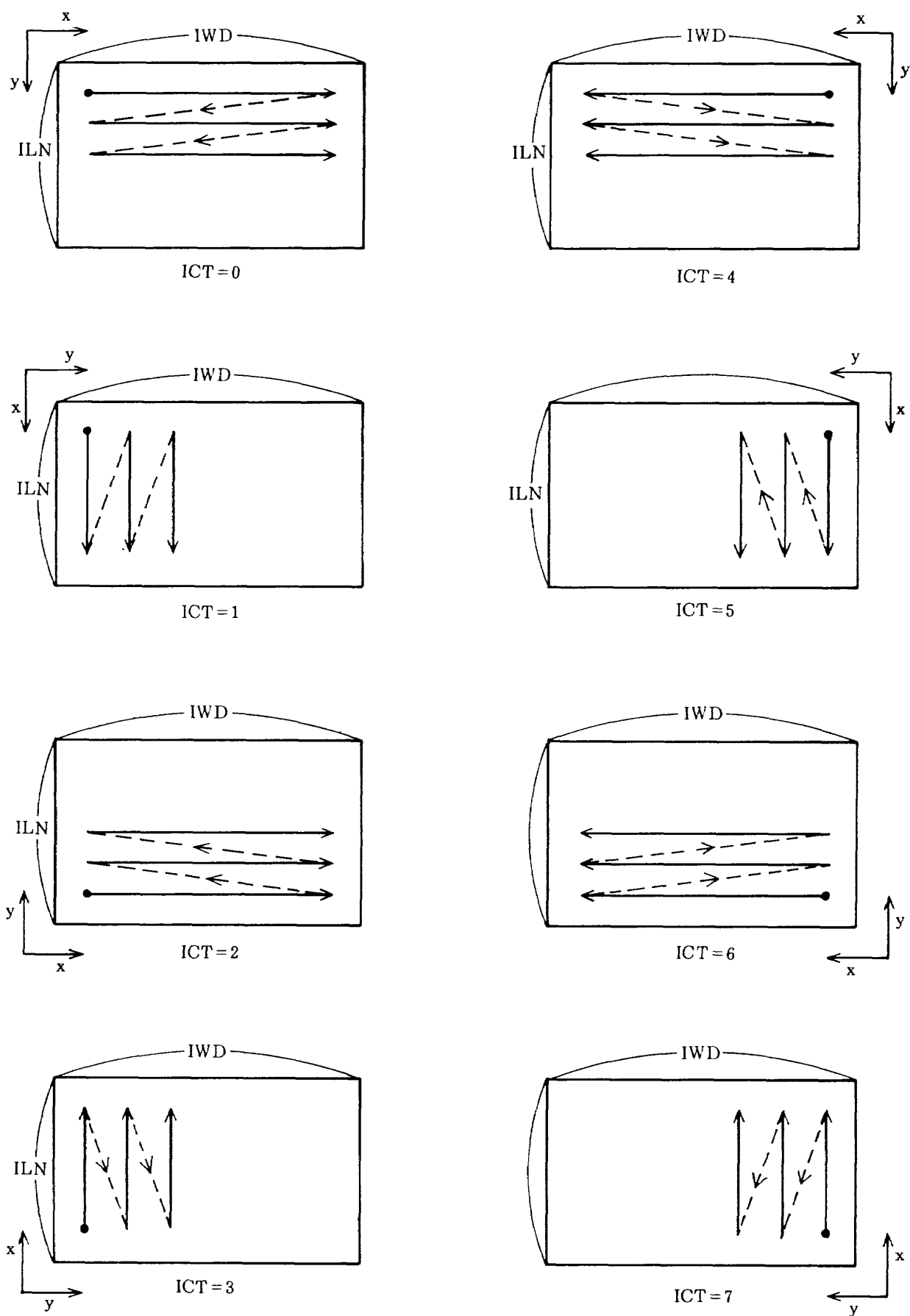


図9 プログラム中での座標と表示座標

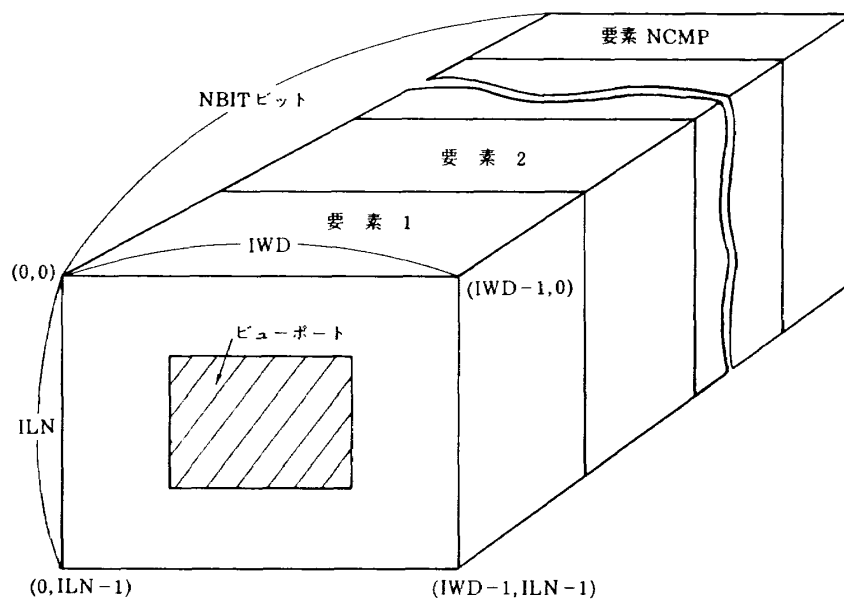


図10 一般化したリフレッシュメモリの構造

要素に指定されたデータを書き込む。各要素に書き込まれるデータMは

$$0 \leq M \leq 2 \cdot (NBIT/NCMP) - 1$$

を満たす自然数である。またIX,IYの範囲は

ICTが偶数のとき

$$0 \leq IX \leq IWD - 1, 0 \leq IY \leq ILN - 1$$

ICTが奇数のとき

$$0 \leq IX \leq ILN - 1, 0 \leq IY \leq IWD - 1$$

となる。

- (5) イメージデータ領域にスカラー量で指定されるデータを描画する。このときスカラー・データと画素構成要素のデータに変換しなければならない。この変換には、スカラー量を模擬カラーに変換する関数を用いる。画素位置指定は機能(4)と同じである。
- (6) イメージメモリ内の指定画素のデータを読み出す。画素位置の指定は機能(4)と同様である。
- (7) 機能(5)で用いる擬似カラー関数を指定する。この機能により、擬似カラーを指定する関数を任意に指定できることになる。この関数は整数型関数サブルーチンで定義され、次のような形式になっている。

$$IFNC(I,S)$$

Iは1画素中のデータ要素番号である。例えばイメージプロセッサ nexus 6410の場合は

$$I = 1 : \text{赤 (R)}$$

$$I = 2 : \text{緑 (G)}$$

$$I = 3 : \text{青 (B)}$$

のデータに対応する。SはIで指定されたデータ要素番号における関数値を決定するためのスカラー量である。あるSに対して、例えば nexus 6410ではR・G・Bのデータは、 $IFNC(1,S)$ 、 $IFNC(2,S)$ 、 $IFNC(3,S)$ で求めることができる。なおこの関数の範囲は、機能(4)で説明したMの範囲と同じである。

- (8) リフレッシュメモリ内にビューポートを定義する。ここで定義されたビューポートは、ディスプレイ画面上に表示されたとき、長方形の領域となる。ビューポートの定義は、(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)の2点を結ぶ直線を対角線とする長方形領域によりなされる。座標系は描画時のものである。リフレッシュメモリ内にビューポートが指定された場合、ビューポート外に対する描画は無視される。

以上8つの機能をもつサブルーチンを用意する。各機能はそれぞれ1つのサブルーチンとしてまとめる。

表6にそれぞれの機能とそのサブルーチン名を示す。また図11は各サブルーチンの機能を図で示したものである。

表6 汎用ラスタスキャン用サブルーチン

| 機能番号 | サブルーチン名 | 内 容 |
|------|---------|-------------------------|
| 1 | IMGDVI | リフレッシュメモリの定義 |
| 2 | IMGDVC | リフレッシュメモリに対応するイメージ領域の定義 |
| 3 | CDNTYP | 配列順序の指定 |
| 4 | DVWTSI | イメージ領域へのデータの描画 |
| 5 | DVWTSF | イメージ領域へのスカラー量データの描画 |
| 6 | DVRDI | イメージ領域中の画素データの読み出し |
| 7 | COLORF | 擬似カラー関数の指定 |
| 8 | GCPVWI | ビューポートの設定 |

4. 応用サブルーチン

3章では、画像データを1画素ずつに展開してイメージ領域へ描画するための基本サブルーチンを作成したが、本章では、画素を意識することなく、あるまとまった画像データを描画するためのサブルーチンを作成する。もちろん画像データは、応用サブルーチン内で1画素ずつに展開され、基本サブルーチンによりイメージ領域に描画される。

ここで扱う画像データは、3角形あるいは4辺形の作る面および直線である。

4.1 ウィンドウ

画像データは、汎用ラスタスキャン用サブルー

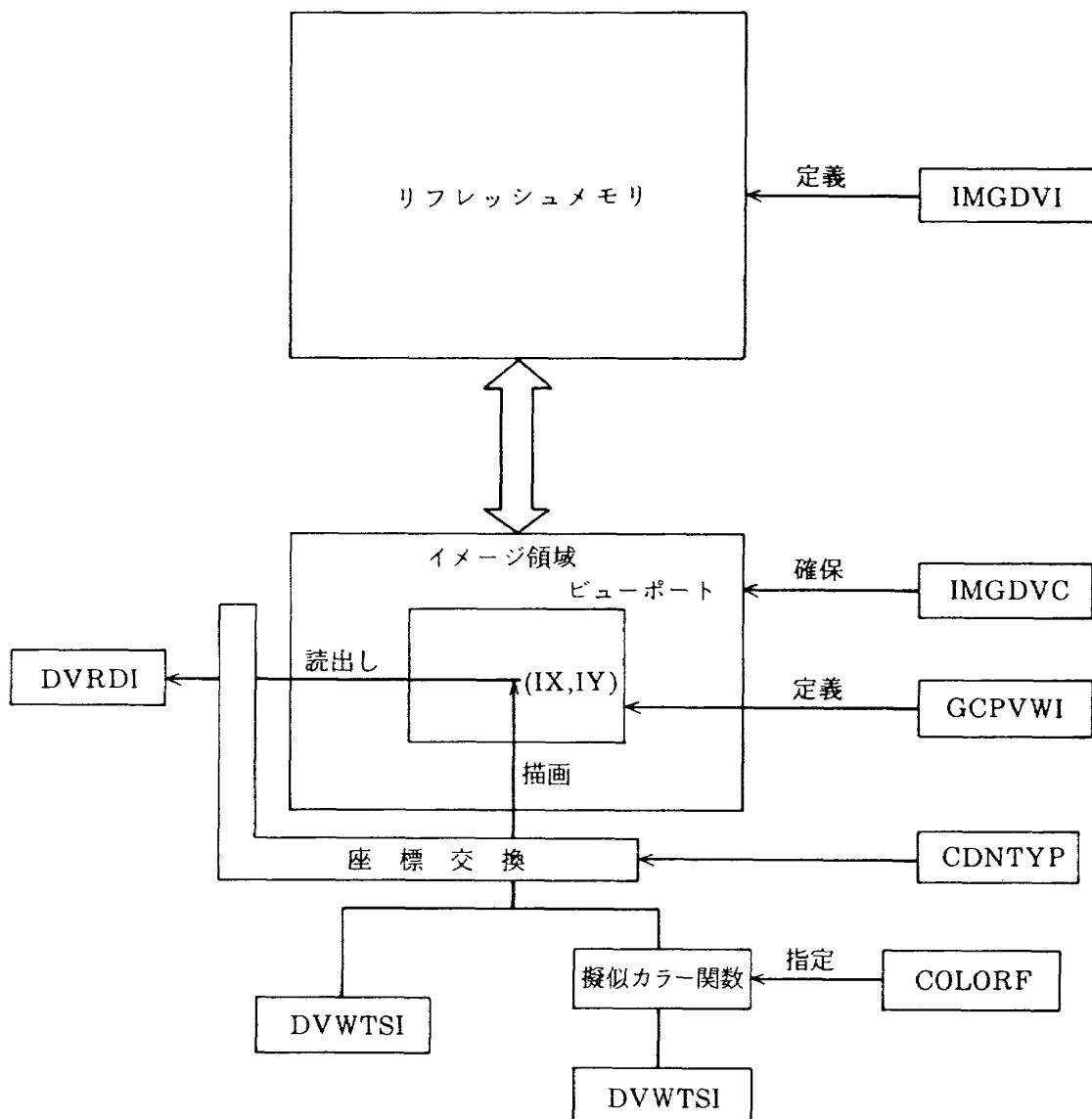


図11 サブルーチンの機能

チンのビューポート設定サブルーチンで設定されたビューポート内に描画される。従って画像データの扱われる座標空間とビューポートの対応をとる必要が生ずる。一般に画像データ座標系内で処理の対象となる領域はウィンドウと呼ばれている。

応用サブルーチンを使用するとき、まずウィンドウの設定を行い、その後、応用サブルーチンを実行することとする。ウィンドウとビューポートは1対1に対応するものとする。すなわちウィンドウ内の画像がビューポートに投影される。

ウィンドウは画像データ座標系内で長方形の領域で定義する。ウィンドウの定義は、(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)の2点を結ぶ直線を対角線とする長方形領域によりなされる。

ウィンドウ, ビューポートの設定がなされると, 画像データ座標・イメージ領域座標変換が可能になる。画像データ座標系で (x, y) の座標点は

$$IX = [k_x (x - x_0)] + IX_0$$

$$IY = [k_y (y - y_0)] + IY_0$$

で計算されるイメージ領域座標点 (IX, IY) へ変換される。

ただし, (x_0, y_0) および (IX_0, IY_0) はそれぞれウィンドウおよびビューポートの(X座標の最小値, Y座標の最小値)である。また, k_x および k_y は

$$k_x = \frac{IX_1 - IX_0 + 0.999}{x_1 - x_0}$$

$$k_y = \frac{IY_1 - IY_0 + 0.999}{y_1 - y_0}$$

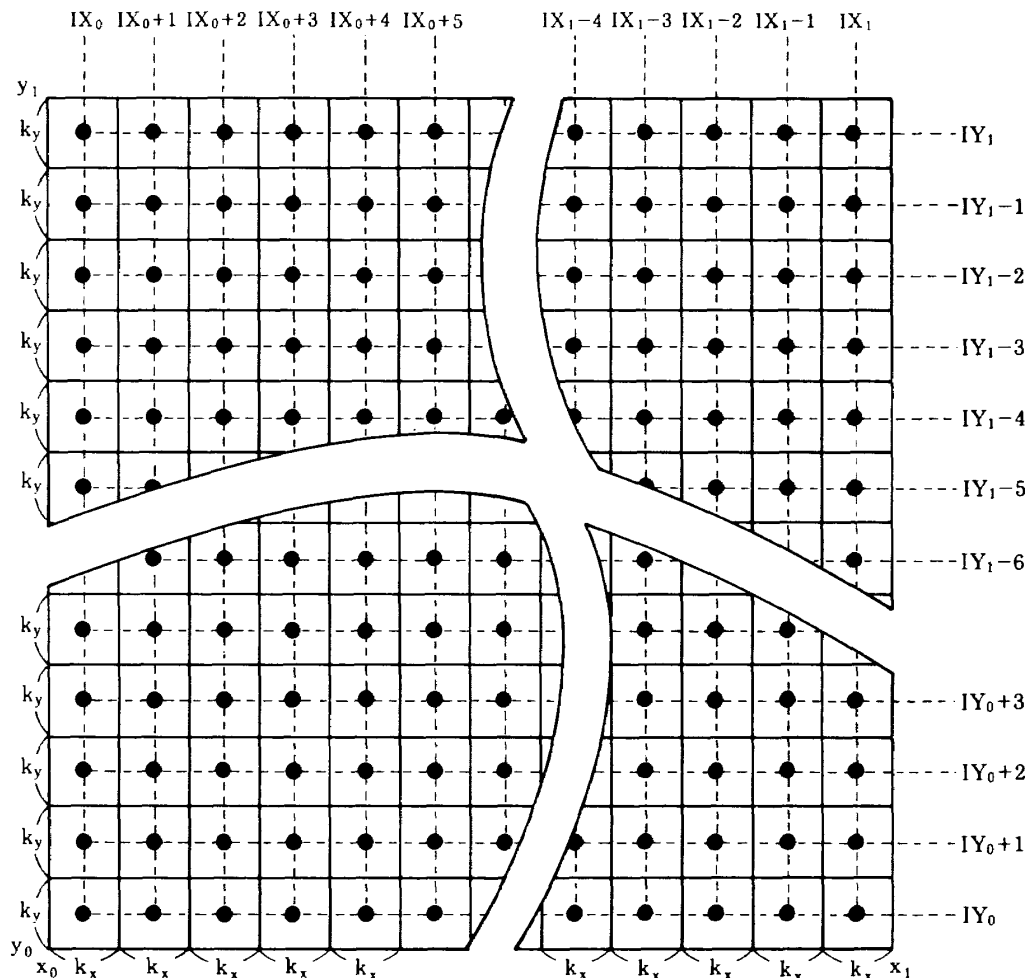


図12 ウィンドウとビューポートの関係

($x_0 \leq x < x_1, y_0 \leq y < y_1$ で示されるウィンドウ内の座標点が
 $IX_0 \leq IX \leq IX_1, IY_0 \leq IY \leq IY_1$ で定義されるビューポート
 内の画素に投影される)

であり、 (x_i, y_i) および (IX_i, IY_i) はそれぞれウインドウおよびビューポートの(X座標の最大値, Y座標の最大値)である。

図12にウインドウとビューポートの関係を示す。

4.2 2次元平面の描画

2次元平面上に置かれた3角形および4辺形の作る面を描画する。面の定義は、3角形あるいは4辺形の頂点の x, y 座標で行われる。また各頂点はそれぞれでスカラー量をもっている。このスカラー量はディスプレイ画面上に表示される時の色として反映される。

2次元平面上の面をディスプレイ上に表示することは、与えられた平面内を擬似カラーで塗りつぶすということになる。この問題は、表示しようとしている2次元平面をディスプレイ上のイメージ領域座標上に投影し、投影された平面領域内の画素における擬似カラーを決定することである。本システムでは擬似カラーとスカラー量は擬似カラー関数で定義されているので、2次元平面領域内画素に相当する位置でのスカラー量を決定することになる。

今与えられているデータは2次元平面(3角形もしくは4辺形)の各頂点座標とその点におけるスカラー量である。すなわちこれらのデータを用いて、平面内のある点におけるスカラー量を補間するという問題になる。補間法のアルゴリズムはいろいろあり、その内から適当なものを選び使用すればよいが、問題点は、対象多角形をイメージ空間に投影したときの画素上の点に対応する多角形内部でのすべての点の位置をいかに能率よく求めるかである。ここではコンピュータグラフィックスにおいて多角形内部の塗りつぶしによく用いられるアルゴリズム⁷⁾の1つである、X軸に平行なスキャン・ライン方向に多角形内部の画素点を走査する方法を採用する。

本プログラムで採用した平面内画素点位置における補間の手順を以下に示す。

- (1) 多角形(3角形あるいは4辺形)の頂点座標およびその点でのスカラー値を、多角形の1つの頂点から右(あるいは左)廻りに定義する。

- (2) (1)で定義された多角形の最大および最小Y座標値を求め、その値を y_{\max}, y_{\min} とおく。
- (3) 多角形を構成する各辺のY方向に対するX座標、スカラー値の変化率DX, DSを求める。例えば辺 $n(P_n P_{n+1})$ のDX, DSを DX_n, DS_n とすると

$$DX_n = \frac{X_{n+1} - X_n}{Y_{n+1} - Y_n} \quad DS_n = \frac{S_{n+1} - S_n}{Y_{n+1} - Y_n}$$

となる。

- (4) $Y = [Y_{\min}]^{\dagger}$ とする。
- (5) 直線 $y=Y$ と交わる多角形の辺を選びだし、その交点のX座標値 x 、スカラー量 s を次式で求める。

$$x = DX_k(y - Y_k) + X_k$$

$$s = DS_k(y - Y_k) + S_k$$

ただし、 DX_k, DS_k は辺 k のDX, DSの値、 X_k, Y_k, S_k は多角形の頂点 k におけるX座標、Y座標、スカラー量である。

このとき各辺の範囲に辺の両端を含めると、交点の数は偶数となる。すなわち、3角形のときは0あるいは2、4辺形のときは0, 2, 4のいずれかとなる。交点数を N とし、 $N = 0$ のときは(13)へ。

- (6) (5)で求めた交点のデータ (x, s) の組を x に関して昇順にソートし、ソート順にデータの組に1から N までの番号を付ける。
 - (7) $I = 1$ とする。
 - (8) データ組 I と $I+1$ のX座標値、スカラー値をそれぞれ x_a, x_b, s_a, s_b とする(2点 $(x_a, y), (x_b, y)$ を結ぶ直線が多角形の内側)。
 - (9) $X = [x_a]$ とする。
 - (10) $S = \frac{s_b - s_a}{x_b - x_a} (X - x_a) + s_a$ を計算する。
- この s を画素位置 (X, Y) のスカラー値とする。
- (11) $X = X + 1$ とする。もし $X \leq x_b$ なら(10)へ。
 - (12) $I = I + 1$ とする。もし $I < N$ なら(8)へ。
 - (13) $Y = Y + 1$ とする。もし $Y \leq y_{\max}$ なら(5)へ。
 - (14) 終了

図13は4点 P_1, P_2, P_3, P_4 から構成される凹形4辺形をイメージ空間に投影したときの図である。図中の格子点はディスプレイ画面上の画素に相当

注† ガウス記号

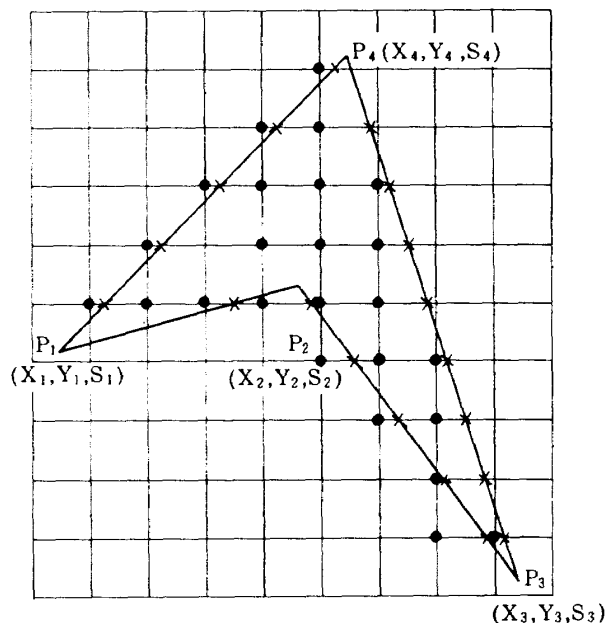


図13 4辺形の描画例 (与えられた4辺形 $P_1P_2P_3P_4$ に対して・の画素が塗りつぶされる)

する点である。上記手順に従って処理を行うと、図中の・の点でスカラー値が計算される。各辺上の×印の点が手順(5)で求められる点である。

4.3 3次元空間内の多角形描画

本節では多角形が3次元空間に置かれている場合を考える。多角形はここでも3角形あるいは4角形である。多角形の定義は前節の多角形定義に座標成分としてZ座標の値を追加したものである。多角形の頂点定義におけるZ値に制限はない。従って4辺形の場合、平面上に乗るとは限らないが4辺形を構成する4辺は直線で定義できるものとする。描画の方法は、2次元平面上への描画と基本的に同じである。すなわち3次元空間内の多角形をイメージ領域空間へ投影し、画素の位置に相当する点のスカラー量を補間し、その点の擬似カラーを表示するということになる。ただし、3次元の場合には、複数の多角形をいっしょに表示しようとするとき、隠面処理を考慮しなければならない。本プログラムではZバッファ法により隠面処理を行う。ある画素上に複数の多面体が重なった場合、ここでは最もZ座標値が大きいものを描画するようにしている。

描画の手順は2次元の場合と同じであるが、ある画素における補間値はスカラー量の他にZ座標

値も同時に補間される。Z座標値の補間方法はスカラー量補間と同じである。この補間されたZ座標値により隠面処理が施される。

隠面処理のアルゴリズムは次の通りである。

- (1) 表示画面上の画素数に対応する大きさを有するZバッファを用意し、初期値として $-\infty$ を代入しておく。
- (2) ある画素に対して描画するとき、描画対象のZ座標値を描画画素に対応するZバッファの値と比較する。もしZ座標値がZバッファ値より小さいときは、描画しようとする点は既に描画されているものにより隠されるとして描画しない。Zバッファ値より大きい場合のみ、描画が許される。このときもし既にその画素上に描画がなされているときは、それを消去した上で描画を行い同時にZバッファ上の値もZ座標値で書き直す。

以上のアルゴリズムにより3次元空間内に多角形を隠面処理を施しながら描画することができる。描画手順は、2次元の場合と同様である。

4.4 直線の描画

与えられた2点 (x_1, y_1) 、 (x_2, y_2) 間を直線で結ぶ描画は、図形処理の基本的な機能である。ラスタスキャン型ディスプレイでも当然与えられた2点間に直線を描画する機能が必要である。この問題はウィンドウ内に引かれた直線をビューポート内の画素に投影するという問題となる。筆者らはこの問題に対するサブルーチンCPLOP⁸⁾を既に作成している。このサブルーチンを利用すれば、簡単にラスタスキャン型ディスプレイ上に直線を引くことができる。

航技研計算センターでは、長い間、図形作図はXYプロッタにより行われてきた。そのためXYプロッタ用の作図サブルーチンを用いて作られた数多くの図形作図用プログラムが計算センター利用者に利用されているし、現在でもXYプロッタ用サブルーチンによりプログラムが作られている。これらXYプロッタ用サブルーチンを用いて作られているプログラムを使用してもラスタスキャン型ディスプレイ上に図形が作図できれば好都合である。

現在航技研計算センターで使用されているXYプロッタ用のプロットング・システム⁹⁾はベシック・ソフトウェア、ファンクショナル・ソフトウェア、アプリケーション・ソフトウェアと呼ばれる3つのソフトウェアから構成されている。各ソフトウェアは種々の機能をもつサブルーチン群である。これらのサブルーチンでは、次の7つの基本サブルーチンを用いて、作図を実行する。

- (1) PLOTS
- (2) PLOT
- (3) SYMBOL
- (4) NUMBER
- (5) FACTOR
- (6) WHERE
- (7) NEWPEN

従って、ラスタスキャン型ディスプレイ用のこれら7つの基本サブルーチンと同名同機能のサブルーチンを用意すれば、XYプロッタ用サブルーチンを使用したプログラムとラスタスキャン型ディスプレイとは整合を取ることができる。

4.5 応用サブルーチンの機能

応用サブルーチンとして次の機能をもつサブルーチンを作成する。

- (1) 画像データ座標系内にウィンドウを定義する。このウィンドウはXY平面上に定義される長方形の領域である。ウィンドウの定義は、(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)の2点を結ぶ直線を対角線とする長方形領域によりなされる。この長方形領域はビューポートの長方形領域に投影される。(2)以降の描画応用サブルーチンを使用する場合、ウィンドウ定義により画像データ座標系とディスプレイ上の座標系との対応を取っておく必要がある。
- (2) XY平面上の3角形, 4辺形をイメージデータ領域にその内部を塗りつぶしながら描画する。塗りつぶす際の色は、各頂点に与えられたスカラー量により補間された値に基づく擬似カラー関数値による。
- (3) 3次元空間内の3角形, 4辺形を隠面処理を施してイメージデータ領域にその内部を塗

表7 応用サブルーチン

| 機能番号 | サブルーチン名 | 内 容 |
|------|---------|---------------------------|
| 1 | GCPWNC | ウィンドウの設定 |
| 2 | IMGS2 | 3角形, 4辺形のイメージ領域への描画 |
| 3 | IMGT2 | 3次元空間内の3角形, 4辺のイメージ領域への描画 |
| 4 | IFHDN0 | 隠面処理のためのZバッファ定義 |
| 4 | IFMDN1 | ある画素における隠面判定 |
| 5 | GMOVE2 | 現在位置の移動 |
| 5 | GLINE2 | 現在位置から指定点までの直線描画 |
| 6 | PLOTS | ダミー |
| 6 | PLOT | 指定点までの直線描画 |
| 6 | SYMBOL | 英数字・記号の描画 |
| 6 | NUMBER | 数値の描画 |
| 6 | FACTOR | スケールファクタの設定 |
| 6 | WHERE | 現在位置とスケールファクタを求める |
| 6 | NEWPEN | 線描画時の色指定 |
| 7 | IMCOLR | 要素データによる線描画時の色指定 |
| 7 | GCOLOR | 模擬カラー関数による線描画時の色指定 |

りつぶしながら描画する。塗りつぶす際の色は、各頂点に与えられたスカラー量により補間された値に基づく擬似カラー関数値による。隠面処理は、(4)の機能を用い、Zバッファ法により実行される。

- (4) Zバッファ法による隠面処理バッファ定義およびある画素への描画時の隠面判定を行う。
- (5) イメージデータ領域に直線を描画する。この機能は、指定座標へ現在位置を移動する機能と現在位置から指定座標までを直線で描画した後、指定座標を新しい現在位置とする機能で実現する。
- (6) XYプロッタ用サブルーチンと整合を取るための7つの基本機能⁹⁾。
- (7) 線図形描画の際の色を指定する。色は画素を構成する各要素の値による指定とスカラー量による擬似カラー関数値での指定の2つの

方法で指定できる。

以上7つの機能をもつサブルーチンを用意する。
表7にそれぞれの機能とそのサブルーチン名を示す。

5. 使用例

5.1 ドラムスキャナ・データの表示例

写真などの画像をデジタル化し、コンピュータに入力可能なデータを生成する装置の1つにドラムスキャナ装置がある。

ここではドラムスキャナ入力データとして、文献10でのドラムスキャナ用プログラムを用いてコンピュータ内に入力されたデータを使用する。試料画像は航技研開発のSTOL機飛鳥のカラー写真を使う。この写真を図14に示すような配置でドラムスキャナにセットし、軸方向(X方向)449ライン、回転方向(Y方向)581データでデジタル化する。このデジタル化したデータを大型コンピュータに入力し、ディスプレイ上に表示する。次の3つの条件で画像化したものをディスプレイ上に一画面として表示する。

- (1) ライン番号1から449まで、回転方向データ65から575まで、それぞれ2つおきにサンプリングしたデータを描画配列順序パラメータICT=1でディスプレイ上の座標(0,0)から表示。
- (2) (1)の条件のICT=5に変え、白黒でディスプレイ上の座標(256,0)から表示。
- (3) ライン番号120から359まで、回転方向データ60から581までのデータをICT=1でディスプレイ上の座標(0,240)から表示。

なおディスプレイ上の原点(0,0)は左上隅で定義されている。

この表示結果を図15に示す。

図15を表示したドラムスキャナ入力データ表示プログラムとその使用法は付録4に示してある。

5.2 2次元平面上の4辺形の表示例

2次元平面上の多角形表示の例として、3次元空気取入口まわりの遷音速ポテンシャル流れの数値解析¹¹⁾結果のある子午面上のマッハ数を擬似カラーで表示する。この計算では、計算子午面を55×30の格子に分割し、55×30の格子点上で数値解

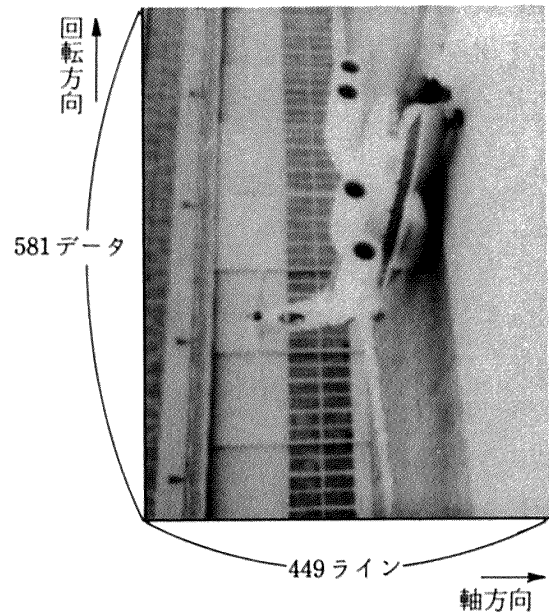


図14 ドラムスキャナ入力画像試料

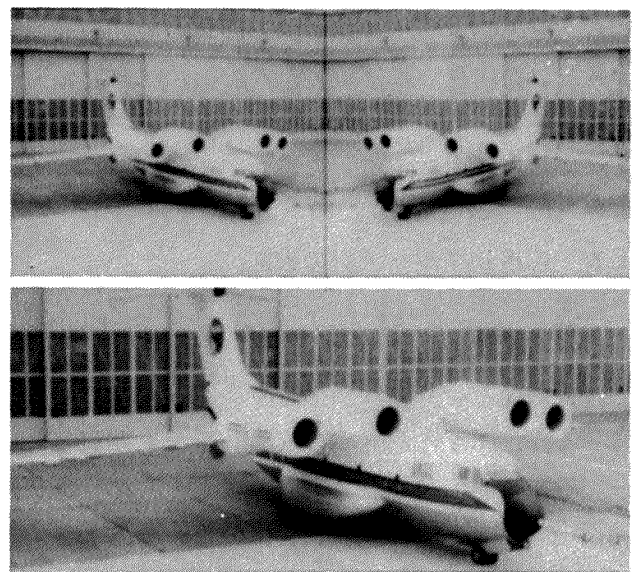


図15 ドラムスキャナ・データの表示例
(写真はカラー・ハードコピーの出力)

析が行われる。格子点の物理上のX,Yの座標値は、2次元配列XX(55,30), YY(55,30)に格納されている。またマッハ数の解析結果は2次元配列PP(55,30)に格納されている。この計算格子を構成する4辺形のセルを一つずつ取り出し、4辺形頂点のマッハ数によりセル内を擬似カラーで塗りつぶす。このようにして描画したマッハ数分布図を図16に示す。

図16を表示するためのプログラムは付録6に、またこのプログラムで使用している擬似カラー関

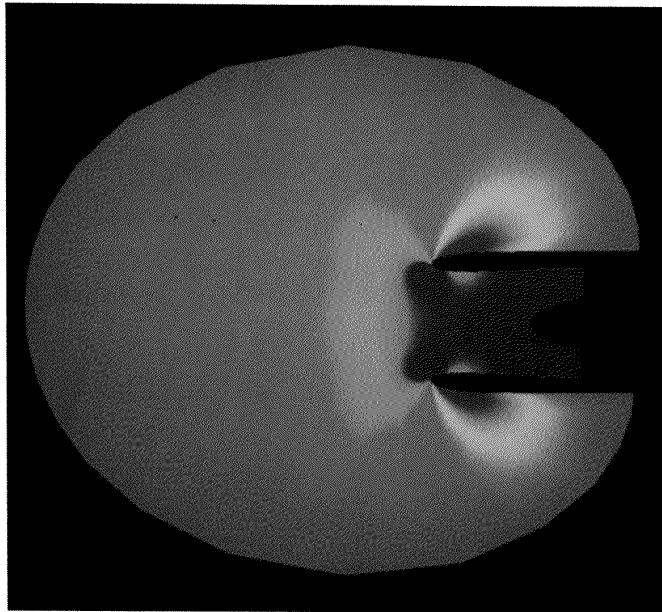


図16 2次元平面上の4辺形の表示例
(写真はポラロイドカメラでディスプレイ面を
直接撮影したもの)

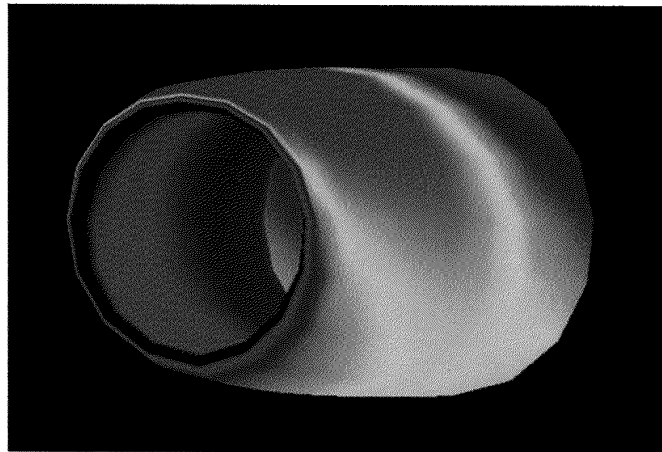


図17 3次元空間内の4辺形の表示例(隠面処理)
(写真はカラー・ハードコピーの出力)

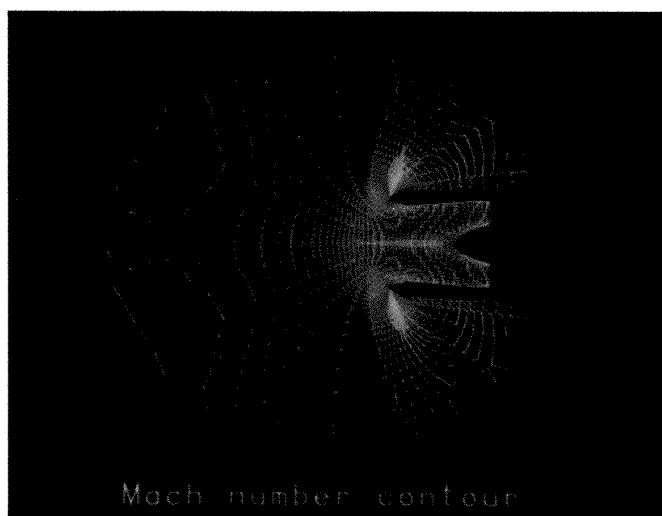


図18 線図の表示例
(写真はカラー・ハードコピーの出力)

数は付録 5 に示してある。

5.3 3次元空間内の4辺形の表示例

例として5.2と同様に、3次元空気取入口まわりの数値解析結果を使用する。ここでは、空気取入口表面のマッハ数分布を表示する。空気取入口表面の座標データとマッハ数データは、子午面データと同様に55×17の格子点上に定義されている。ただし、この格子点は3次元空間に配置されているので、Z座標用の2次元配列が追加される。ここでも格子を構成する4辺形セルを一つずつ取り出し、4辺形頂点のマッハ数によりセル内を擬似カラーで塗りつぶす。このようにして描画した空気取入口表面のマッハ数分布図を図17に示す。

図17を表示するためのプログラムは付録6に示してある。擬似カラー関数は付録5のものである。

5.4 線図の表示例

線図の表示例として、図16を等高線表示する。まず計算格子を描き、その上に等高線を表示する。等高線は、それぞれの値に従ってそれぞれ異なる色付がなされている。色は等高線の値をパラメータとする擬似カラー関数値による。また図の説明を英字で表示する。格子図、等高線図は、図形作図用の汎用サブルーチンを用いて描く。英字はXYプロッタ用アプリケーション・ソフトウェア中のサブルーチンを使用する。これらのサブルーチンによる等高線表示結果を図18に示す。ここで用いた格子図、等高線図作図サブルーチンは、XYプロッタにもそのまま使用できるものである。

図18を表示するためのプログラムは付録6に示してある。等高線の色付けは付録5の擬似カラー関数による。

なお図16～18のためのデータは、計算センターの中村孝技官から提供されたものである。

6. ま と め

ラスタスキャン型ディスプレイを大型コンピュータに接続し、ラスタスキャン型ディスプレイのための大型コンピュータ用サブルーチンを作成した。主の結果は次の通りである。

(1) 大型コンピュータにラスタスキャン型ディ

スプレイをチャンネルを介して接続した。

(2) ラスタスキャン型ディスプレイを使用するためのプログラムをサブルーチン形式で作成した。

(3) ディスプレイ固有の機能とラスタスキャン型ディスプレイ共通機能を明確に分離し、サブルーチンの一般化を図った。

(4) XYプロッタ用サブルーチンを使って書かれているプログラムも修正することなくラスタスキャン型ディスプレイ上に表示できるようにした。

(5) 数値計算結果のデータ構造との整合性を考慮したデータ入力形式で画像の描画を可能にした。

最後に、本資料の多くのプログラム、サブルーチンは奥野隆史君・高橋美月さんの手によりコーディングされ、ディバックされたものであることを記し、感謝の意とする。

参考文献

- 1) 吉田豊明；赤外線温度計測装置(Ⅱ)ソフトウェア，航技研資料 TM-531, 1984年5月
- 2) 榊柏木研究所；nexus 6400取扱説明書RVE.01, 1983年
- 3) 榊柏木研究所；nexus 68230デジタイザ取扱説明書 REV 3.0
- 4) 榊柏木研究所；nexus 68010 GP-IB インターフェイス取扱説明書 REV.01
- 5) 榊ネクス；サンプルプログラム操作手引書 N88-BASIC 高速GP-IB/PCDM, 1985年
- 6) 榊柏木研究所；nexus 68045 BMCインターフェース仕様書 取扱説明書 REV.01
- 7) 例えば前田信一，小林健造；図形の変更や視点の移動をホストとは独立に処理するグラフィック・ディスプレイ装置，日経エレクトロニクス 1984.6.18 pp 205-232
- 8) 石塚只夫，磯部俊夫；PLOTルーチン形式による図形出力装置の使用について，航技研資料TM-437, 1981年6月
- 9) 吉沢ビジネス・マシンズ㈱；CALCOMP PLOTTERプログラミング — ベーシック・

ソフトウェア — ソフトウェア・マニュアル,
昭和52年4月

- 10) 本間幸造, 磯部俊夫; ドラムスキャナ用プログラム, 航技研資料TM-548, 1985年9月
- 11) 中村孝; 3次元空気取入口まわりの遷音速ポテンシャル流れの数値解析, 航空機計算空気力学シンポジウム講演論文集, 航技研SP-5, 1985年11月 pp61-68

付 録 1

○ ラスタスキャン型ディスプレイ装置 NEXUS 6410 の操作法

- (1) NEXUS 6410 本体電源 ON, モニタディスプレイ電源 ON, マイクロコンピュータ電源 ON
- (2) マイクロコンピュータのフロッピー・ドライブ 1 に「GPIB.BAS」のフロッピー・ディスクを入れ、マイクロコンピュータの RESET を押す（システム・プログラムが読み込まれる）。
- (3) マイクロコンピュータのディスプレイ画面上に入力促進記号“>”が表示される。

これでマイクロコンピュータと NEXUS 6410 が接続された状態になる。マイクロコンピュータのキーボードから NEXUS 6410 のコマンドを入力すれば、コマンドに従って NEXUS 6410 を動作させることができる。コマンド、操作法は文献 2), 5) を参照のこと。

大型コンピュータとオンラインで接続するためには、さらに次の操作を行う。

- (4) TSS セッションを開設、このとき TSS のワークエリアとして 2M バイト以上を確保する必要がある。
- (5) NEXUS 6410 を TSS 上で定義するために、次のコマンドを TSS 端末から入力する。

```
ALLOC, DD(NEXUS1), UNIT(500), DS('X'), O
```

ここで X は任意の文字列である。

- (6) マイクロコンピュータのキーボードから

```
HOSTIF = 3
```

を入力する。これで大型コンピュータと NEXUS 6410 がオンライン接続される。

(1)~(6) までの操作で大型コンピュータが NEXUS 6410 のホストコンピュータになる。ホストをマイクロコンピュータに戻すには

- (1) NEXUS 6410 の RESET を押す。
- (2) NX10 のサブルーチン NXRSET を CALL する。
- (3) NX10 のサブルーチン NXCPUT を用い、コマンド“HOSTIF=2”を NEXUS 6410 へ送る。

のいずれかの方法による。マイクロコンピュータにホストが移るとマイクロコンピュータのディスプレイ画面上に入力促進記号“>”が点滅する。

- (7) 大型コンピュータ内のプログラムにより NEXUS 6410 に画像を出力するには、NX10 サブルーチンもしくは本資料定義のサブルーチンを用いればよい。すなわち、これらのサブルーチンを含むライブラリとプログラムをリンケージさせ、TSS 上で実行する。

○ ホストが大型コンピュータであり、プログラム実行中に異常状態が発生したときの操作法

- (1) TSS を“READY”状態にする。
- (2) NEXUS 6410 の RESET を押す（ホストがマイクロコンピュータに移る）。
- (3) マイクロコンピュータのキーボードから

```
HOSTIF = 3
```

を入力し、ホストを再び大型コンピュータへ戻す。

- (4) プログラムを再実行する。

○ ホストがマイクロコンピュータのときの主な処理の方法

- (1) メインエリアをクリアする。

> ERASE ↵

- (2) サブエリアをクリアする。

> SERASE ↵

- (3) キャラクタをクリアする。

> HOME ↵

- (4) NEXUS 6410 の画面上の画像をフロッピーディスクにセーブする。

- (a) 「GBIB.BAS」のフロッピーディスクをドライブ 1 に、画像をセーブするディスクをドライブ 2 に入れる。

- (b) > DISK A ↵

> FORMAT ↵

ディスク ヲ ドライブ 2 ニ イレテクダサイ Y ↵

> GET ↵

- (5) フロッピーディスクのデータを NEXUS-6410 に送る。

- (a) データの入っているフロッピーディスクをドライブ 2 に入れる。

- (b) > PUT ↵

注) 下線分はマイクロコンピュータのキーボードから入力することを示している。

付 録 2

○ NX10の主なサブルーチン使用法（詳細は文献5）参照）

(1) サブルーチン NXOPEN

CALL NXOPEN(IRC)

IRC：終了情報をプログラムへ帰す。0のとき正常終了。(2)以下のサブルーチンも同様。

(2) サブルーチン NXCLOS

CALL NXCLOS(IRC)

(3) サブルーチン NXRSET

CALL NXRSET(IRC)

(4) サブルーチン NXBGET

CALL NXBGET(IBUF, NB, NRB, IRC)

IBUF：バイナリーデータを読み取るバッファ名

NB：IBUFの大きさをバイト数で指定（最大1Mバイト）

NRB：実際に読み取ったデータのバイト数

(5) サブルーチン NXBPUT

CALL NXBPUT(IBUF, NB, IRC)

IBUF：出力するバイナリーデータの格納されているバッファ名

NB：出力データのバイト数（最大1Mバイト）

(6) サブルーチン NXCGET

CALL NXCGET(CHAR, NC, NRC, IRC)

CHAR：コマンドデータを読み取るバッファ名

NC：CHARの大きさをバイト数で指定

NRC：実際に読み取ったコマンドデータのバイト数

(7) サブルーチン NXCPUT

CALL NXCPUT(CHAR, NC, IRC)

CHAR：出力するコマンドデータの格納されているバッファ名

NC：出力コマンドデータのバイト数

注) コマンドデータは EBCDIC コードで最大1023バイト。

付 録 3

○ サブルーチン使用説明

本プログラムは37のサブルーチンから構成されている。これらのサブルーチンの使用法を以下に示す。
 パラメータの型はFORTRANの暗黙の型宣言，すなわち最初の文字がI, J, K, L, M, Nのいずれかで
 あるときは4バイトの整数型，それ以外は特にことわりのない限り4バイトの実数型である。

サブルーチン 1 NXOPN

| | |
|---------|--|
| サブルーチン名 | NXOPN |
| 機能 | ディスプレイの入出力を可能とするための準備を行う。ディスプレイを使用する場合，最初にCALLしなければならない。 |
| 呼び出し形式 | CALL NXOPN |

サブルーチン 2 NXCLS

| | |
|--------|--|
| サブルーチン | NXCLS |
| 機能 | ディスプレイへの出力を終了するためのクローズ処理を行う。以後，ディスプレイへの入出力は出来ない。 |
| 呼び出し形式 | CALL NXCLS |

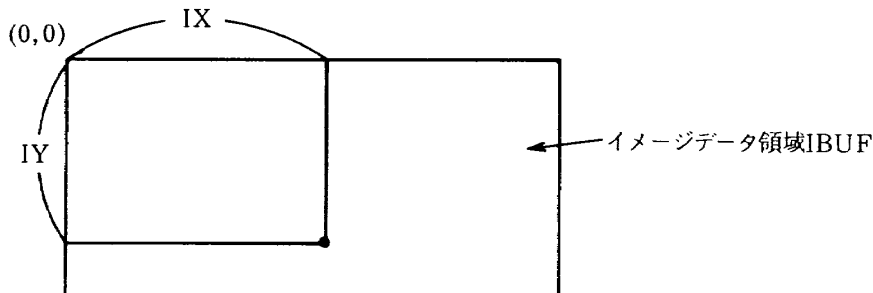
サブルーチン 3 NXCLR

| | |
|---------|--------------------------|
| サブルーチン名 | NXCLR |
| 機能 | ディスプレイ内のリフレッシュメモリをクリアする。 |
| 呼び出し形式 | CALL NXCLR |

サブルーチン 4 NXWRT

| | | |
|---------|---------------------------------|--|
| サブルーチン名 | NXWRT | |
| 機能 | イメージデータをディスプレイ内のリフレッシュメモリへ出力する。 | |
| 呼び出し形式 | CALL NXWRT(IBUF) | |
| パラメータ | 入出力 | パラメータの説明 |
| IBUF | 入力 | 1次元配列名，大きさ 184320 語 画像データ格納配列 (イメージデータ) |

サブルーチン 5 NXPUT

| サブルーチン名 | | NXPUT |
|---|-----|---|
| 機能：ディスプレイ上の座標 (IX,IY) の位置に相当するイメージデータ領域IBUFに IR(赤), IG(緑), IB(青) のデータを書き込む。 | | |
| 呼び出し方式 | | CALL NXPUT(IBUF, IX, IY, IR, IG, IB) |
| パラメータ | 入出力 | パラメータの説明 |
| IBUF | 出力 | 1次元配列名, 大きさ 184320 語 出力したい R・G・B データを格納する配列名 |
| IX | 入力 | X 軸方向の座標 |
| IY | 入力 | Y 軸方向の座標 |
| IR | 入力 | RED(赤) のデータ名 (下位 8 ビットのみ有効) |
| IG | 入力 | GREEN(緑) のデータ名 (") |
| IB | 入力 | BLUE(青) のデータ名 (") |
| 備考：IX は 0 ～ 511, IY は 0 ～ 479 までの整数 (左上が座標 (0,0) となる)。 IR, IG, IB は 0 ～ 255 までの整数で 256 以上は 256 の剰余になる。 (0 → 255 は暗 → 明に対応する) | | |
|  | | |

サブルーチン 6 NXRD

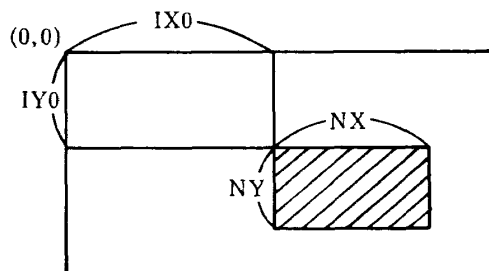
| サブルーチン名 | | NXRD |
|-------------------------------|-----|--|
| 機能：ディスプレイ内のリフレッシュメモリの内容を読み取る。 | | |
| 呼び出し形式 | | CALL NXRD(IBUF) |
| パラメータ | 入出力 | パラメータの説明 |
| IBUF | 出力 | 1次元配列名, 大きさ 184320 語 ディスプレイ上の R・G・B データが書き込まれる。 |

サブルーチン 7 NXRDP

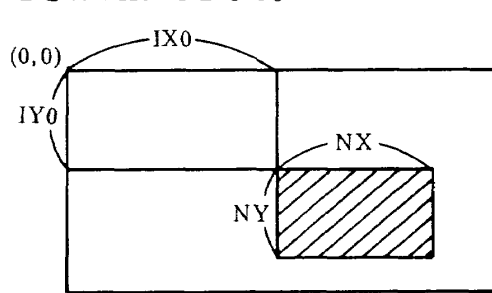
| サブルーチン名 | | NXRDP |
|---|-----|--------------------------------------|
| 機能：ディスプレイ上の座標 (IX,IY)の位置に相当するメメージデータ領域IBUFから赤，緑，青のデータを読み出す。 | | |
| 呼び出し形式 | | CALL NXRDP(IBUF, IX, IY, IR, IG, IB) |
| パラメータ | 入出力 | パラメータの説明 |
| IBUF | 入力 | 1次元配列名，大きさ 184320 語 画像データ格納配列 |
| IX | 入力 | X軸方向の座標 |
| IY | 入力 | Y軸方向の座標 |
| IR | 出力 | 赤のデータ |
| IG | 出力 | 緑のデータ |
| IB | 出力 | 青のデータ |
| 備考：IXは0～511，IYは0～479までの整数。サブルーチンNXPUTに対応するデータ読み出しサブルーチン。 | | |

サブルーチン 8 NXWT2

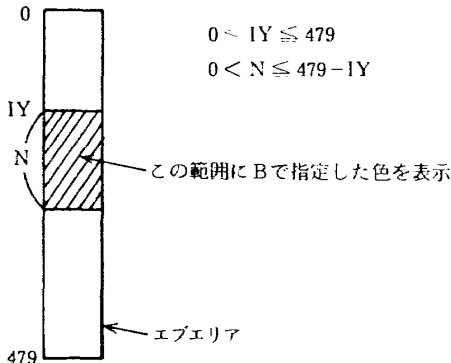
| サブルーチン名 | | NXWT2 |
|--|-----|--|
| 機能：データをディスプレイの指定されたメモリ領域に出力する。 | | |
| 呼び出し形式 | | CALL NXWT2(IBUF, IXO, IYO, NX, NY, MR) |
| パラメータ | 入出力 | パラメータの説明 |
| IBUF | 入力 | 1次元配列名，大きさ NX * NY 画像データ格納配列 |
| IXO | 入力 | ディスプレイ上の書き出し点のアドレス |
| IYO | 入力 | |
| NX | 入力 | X方向の画素数 |
| NY | 入力 | Y方向の画素数 |
| MR | 入力 | MR=0：赤のメモリ領域 MR=1：緑 " MR=2：青 " MR=3：赤，緑，青の全領域 |
| 備考：0 ≤ IXO ≤ 511, 0 ≤ IYO ≤ 479 NX > 0, NY > 0 MR=0, 1, 2, 3でないときはMR=3となる。 | | |



サブルーチン 9 NXRD2

| | | |
|---|--|--|
| サブルーチン名 | NXRD2 | |
| 機能：ディスプレイ内のリフレッシュメモリの一部のデータを読み取る。 | | |
| 呼び出し形式 | CALL NXRD2(IBUF, IXO, IYO, NX, NY, MR) | |
| パラメータ | 入出力 | パラメータの説明 |
| IBUF | 出力 | 1次元配列名, 大きさ $NX * NY$ 画像データ格納配列 } ディスプレイ上の読み出し点のアドレス X方向の画素数 Y方向の画素数 MR=0 : 赤のメモリ領域 MR=1 : 緑 " MR=2 : 青 " MR=3 : 赤, 緑, 青の全領域 |
| IXO | 入力 | |
| IYO | 入力 | |
| NX | 入力 | |
| NY | 入力 | |
| MR | 入力 | |
| 備考： $0 \leq IXO \leq 511$, $0 \leq IYO \leq 479$ $NX > 0$, $NY > 0$ MR=0, 1, 2, 3でないときはMR=3となる。 | | |
|  | | |

サブルーチン 10 NXSBOX

| | | |
|---|-----|---|
| サブルーチン名 | | NXSBOX |
| 機能：サブエリアにカラーバーを表示する。 | | |
| 呼び出し形式 | | CALL NXSBOX(B, IY, N) |
| パラメータ | 入出力 | パラメータの説明 |
| B | 入力 | 表示色を決定するスカラー値格納1次元配列, 大きさN カラーバーの表示位置 カラーバーの幅 |
| IY | 入力 | |
| N | 入力 | |
| 備考： | | |
| <div style="display: flex; align-items: center;"> <div style="flex: 1;">  </div> <div style="flex: 1; padding-left: 20px;"> $0 < IY \leq 479$ $0 < N \leq 479 - IY$ </div> </div> | | |

サブルーチン11 NXPRO

| | | |
|---|-----|--|
| サブルーチン名 | | NXPRO |
| 機能：キャラクタ文字の色を指定し、キャラクタ文字表示の準備を行う。 | | |
| 呼び出し形式 | | CALL NXPRO(IR, IG, IB) |
| パラメータ | 入出力 | パラメータの説明 |
| IR | 入力 | キャラクタ文字の色の赤成分の値 ($0 \leq IR \leq 255$) |
| IG | 入力 | キャラクタ文字の色の緑成分の値 ($0 \leq IG \leq 255$) |
| IB | 入力 | キャラクタ文字の色の青成分の値 ($0 \leq IB \leq 255$) |
| 備考：キャラクタ文字を表示する前に必ずこのサブルーチンを呼び出す必要がある。ここで指定する色はキャラクタ空間に表示される全キャラクタに共通のもので、個々のキャラクタに対する色指定はできない。 | | |

サブルーチン12 NXPRТ

| | | | | | |
|--|--------------|-----------------------------|--|------------------------------|--|
| サブルーチン名 | | NXPRТ | | | |
| 機能：キャラクタ文字をキャラクタ空間に表示する。また既に表示されているキャラクタ文字を消去。 | | | | | |
| 呼び出し形式 | | CALL NXPRТ(IX, IY, LN, CHR) | | | |
| パラメータ | 入出力 | パラメータの説明 | | | |
| IX | 入力 } 入力 } | { | | キャラクタ文字書き出しのキャラクタ座標 (IX, IY) | |
| IY | | | | IX・IY<0のとき全キャラクタ文字消去 | |
| LN | 入力 | 表示キャラクタ文字数 | | | |
| CHR | 入力 | 表示キャラクタ文字の格納されている文字数 | | | |
| 備考：キャラクタ文字表示のとき $0 \leq IX \leq 76$ $0 \leq IY \leq 29$ $0 < LN \leq 77$ 書き終りの位置が29行のときは、キャラクタ表示後、表示キャラクタ全体が1行スワロール・アップされる。なおキャラクタ文字表示のときは、あらかじめサブルーチンNXPROを呼び出しキャラクタ文字の色指定を行うこと。 | | | | | |

サブルーチン13 IMDVI

| | | |
|---|-----|-------------------|
| サブルーチン名 | | IMGDVI |
| 機能：リフレッシュメモリの定義を行う。 | | |
| 呼び出し形式 | | CALL IMGDVI |
| パラメータ | 入出力 | パラメータの説明 |
| IWD | 入力 | X方向の画素数 |
| ILN | 入力 | Y方向の画素数 |
| NBIT | 入力 | 1画素当りのデータ長(単位ビット) |
| NCMP | 入力 | 要素数 |
| 備考：NBITはNCMPの倍数 このサブルーチンで定義されるリフレッシュメモリの大きさは $IWD \cdot ILN \cdot NBIT$ (ビット) である。 | | |

サブルーチン 14 IMGDVC

| | | |
|--|---------------------------------------|-------------------------|
| サブルーチン名 | IMGDVC | |
| 機能: | リフレッシュメモリに対応するプログラム内のイメージデータ領域の定義を行う。 | |
| 呼び出し形式 | CALL IMGDVC(IBUF) | |
| パラメータ | 入出力 | パラメータの説明 |
| IBUF | 入力 | 1次元配列名, 大きさりリフレッシュメモリ相当 |
| 備考: IBUFの実際の大きさは $IWD \cdot ILN \cdot NBIT / 32$ (語) である。IWD, ILN, NBITはサブルーチンIMGDVIのパラメータ | | |

サブルーチン 15 CDNTYP

| | | |
|--|----------------------------|------------------|
| サブルーチン名 | CDNTYP | |
| 機能: | イメージデータ領域に描画する際の配列順序を指定する。 | |
| 呼び出し形式 | CALL CDNTYP(ICT) | |
| パラメータ | 入出力 | パラメータの説明 |
| ICT | 入力 | 0から7までの整数, 図9を参照 |
| 備考: このサブルーチンが呼び出されないでイメージデータ領域に描画するとICT=0の配列順序で描画が行なわれる。 | | |

サブルーチン 16 DVWTSI

| | | |
|---|-------------------------|---|
| サブルーチン名 | DVWTSI | |
| 機能: | イメージデータ領域に1画素のデータを描画する。 | |
| 呼び出し形式 | CALL DVWTSI(IX, IY, ID) | |
| パラメータ | 入出力 | パラメータの説明 |
| IX IY ID | 入力 入力 入力 | $\left. \begin{array}{l} \\ \\ \end{array} \right\}$ 描画する画素位置 (IX, IY) 描画データの格納されている1次元配列, 大きさNCMP |
| 備考: nexus用サブルーチンNXPUTに対応 IX, IYの範囲は ICTが偶数のとき $0 \leq IX \leq IWD-1, 0 \leq IY \leq ILN-1$ ICTが奇数のとき $0 \leq IX \leq ILN-1, 0 \leq IY \leq IWD-1$ IDの大きさはリフレッシュメモリの要素数, データの範囲は $2 * \{ (NBIT/NCMP) - 1 \}$ 以下の自然数 | | |

サブルーチン 17 DVWTSF

| | | | |
|--|----------------|-----------------------------|--|
| サブルーチン名 | | DVWTSF | |
| 機能：イメージデータ領域に 1 画素のスカラー値を描画する。 | | | |
| 呼び出し形式 | | CALL DVWTSF(IX, IY, F) | |
| パラメータ | 入出力 | パラメータの説明 | |
| IX IY F | 入力 入力 入力 | } 描画する画素位置 (IX,IY) 描画データ | |
| 備考：実際に描画されるデータは F をパラメータとするサブルーチン COLORF で指定した擬似カラー関数により決定される。 | | | |

サブルーチン 18 DVRDI

| | | |
|---|-----|--|
| サブルーチン名 | | DVRDI |
| 機能：イメージデータ領域内のある画素のデータ値を読み出す。 | | |
| 呼び出し形式 | | CALL DVRDI(IX, IY, ID) |
| パラメータ | 入出力 | パラメータの説明 |
| IX | 入力 | } データ読み出し画素位置 (IX,IY) データが読み出される 1 次元配列, 大きさ NCMP |
| IY | 入力 | |
| ID | 出力 | |
| 備考：nexus 用サブルーチンNXRDPに対応。IX, IY の範囲はサブルーチンDVWTSIと同じ | | |

サブルーチン 19 COLORF

| | | |
|------------------|-----|-------------------|
| サブルーチン名 | | COLORF |
| 機能：擬似カラー関数を指定する。 | | |
| 呼び出し形式 | | CALL COLORF(IFNC) |
| パラメータ | 入出力 | パラメータの説明 |
| IFNC | 入力 | 擬似カラー関数名 |

備考：パラメータに指定する関数IFNCはEXTERNAL宣言をしておくこと。すなわち次のような宣言文を用意する。

EXTERNAL IFNC

関数IFNCは次のような形式にする。

IFNC(I,S)

Iは画素のデータ要素番号，SはIで指定されたデータ要素番号における関数値を決定するためのスカラー量。

擬似カラー関数のプログラム例を付録5に示す。なおこのサブルーチンによる擬似カラー関数の指定がなされないときは，この付録5の関数が用いられる。

サブルーチン 20 CWIDTH

| | | |
|--|-----|-------------------------|
| サブルーチン名 | | CWIDTH |
| 機能：擬似カラー関数の入力パラメータの範囲を設定する。 | | |
| 呼び出し形式 | | CALL CWIDTH(PMAX, PMIN) |
| パラメータ | 入出力 | パラメータの説明 |
| PMAX | 入力 | 入力パラメータの最大値 |
| PMIN | 入力 | 入力パラメータの最小値 |
| <p>備考：このサブルーチンは、本プログラム全体で共通に取られているコモン領域に PMAX, PMIN 等のデータを設定するだけである。このコモンは 128 語の領域で GPCPMN の名前が付けられている。このサブルーチンでは、この領域の</p> <p>22 番目→ 1, 82 番目→ PMAX, 83 番目→ PMIN, 84 番目→ PMAX-PMIN を設定する。擬似カラー関数を新しく作るとき、これらのコモン値を利用することができる。もちろん利用しなくてもよい。</p> <p>このサブルーチン内で擬似カラー関数の最初のパラメータを 0 にして（すなわち、IRFNC (0,X)）、関数を呼び出している（付録 5 参照）。</p> <p>なお、擬似カラー関数をサブルーチン COLORF により指定するときは、CALL COLORF の後にこのサブルーチンを CALL すること。</p> | | |

サブルーチン 21 GCPVWI

| | | |
|--|-----|-----------------------------------|
| サブルーチン名 | | GCPVWI |
| 機能：リフレッシュメモリ内にビューポートを定義する。 | | |
| 呼び出し形式 | | CALL GCPVWI(IVXO, IVYO, IVX, IVY) |
| パラメータ | 入出力 | パラメータの説明 |
| IVXO | 入力 | X座標の最小値 |
| IVYO | 入力 | Y座標の最小値 |
| IVX | 入力 | X座標の最大値 |
| IVY | 入力 | Y座標の最大値 |
| <p>備考：ビューポート外に対する描画は無視される。</p> <p>ビューポートが設定されないときは、フレッシュメモリ全体に描画される。</p> | | |

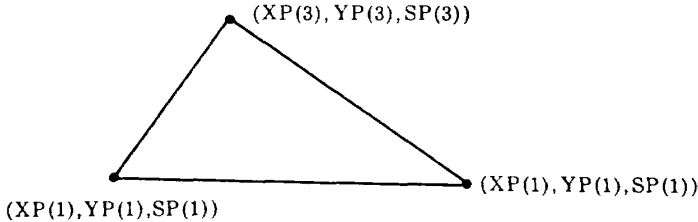
サブルーチン 22 GCPWNC

| | | |
|---|-----|-------------------------------|
| サブルーチン名 | | GCPWNC |
| 機能：画像データ座標系内にウィンドウを定義する。 | | |
| 呼び出し形式 | | CALL GCPWNC(WXO, WYO, WX, WY) |
| パラメータ | 入出力 | パラメータの説明 |
| WXO | 入力 | X座標の最小値 |
| WYO | 入力 | Y座標の最小値 |
| WX | 入力 | X座標の最大値 |
| WY | 入力 | Y座標の最大値 |
| 備考：このサブルーチンで設定されたウィンドウがリフレッシュメモリ内のビューポートに投影される。 | | |

サブルーチン 23 IMGS2

| | | |
|---------------------------|-----|---------------------------|
| サブルーチン名 | | IMGS2 |
| 機能：2次元多角形をイメージデータ領域に描画する。 | | |
| 呼び出し形式 | | CALL IMGS2(XP, YP, SP, N) |
| パラメータ | 入出力 | パラメータの説明 |
| XP | 入力 | 頂点のX座標格納1次元配列, 大きさ N |
| YP | 入力 | 頂点のY座標格納1次元配列, 大きさ N |
| SP | 入力 | 頂点におけるスカラー量格納1次元配列, 大きさ N |
| N | 入力 | 3あるいは4, 3のとき3角形, 4のとき4辺形 |

備考：XP, YP, SPの定義は図のように各配列にデータを格納して行う。

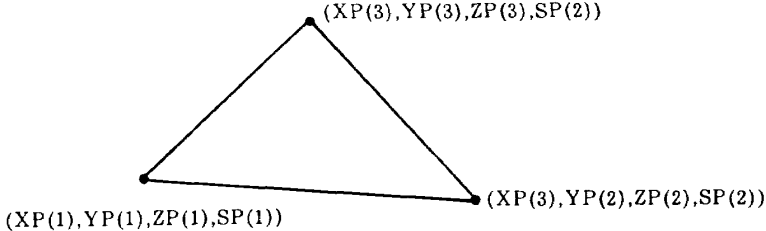


描画対象の多角形のある空間をサブルーチンGCPWNCで定義しておく必要がある。
多角形内の色は, SPをパラメータとする擬似カラー関数により決定される。この関数はサブルーチンCOLORFで定義できる。

サブルーチン 24 IMGT2

| | | |
|---------------------------------------|-----|-------------------------------|
| サブルーチン名 | | IMGT2 |
| 機能：3次元空間内の多角形を隠面処理を施してイメージデータ領域に描画する。 | | |
| 呼び出し形式 | | CALL IMGT2(XP, YP, ZP, SP, N) |
| パラメータ | 入出力 | パラメータの説明 |
| XP | 入力 | 頂点のX座標格納1次元配列, 大きさ N |
| YP | 入力 | 頂点のY座標格納1次元配列, 大きさ N |
| ZP | 入力 | 頂点のZ座標格納1次元配列, 大きさ N |
| SP | 入力 | 頂点におけるスカラー量格納1次元配列, 大きさ N |
| N | 入力 | 3あるいは4, 3のとき3角形, 4のとき4辺形 |

備考：XP, YP, ZP, SPの定義は図のように右あるいは左廻りの順で各配列にデータを格納して行う。



このサブルーチンを使用する前に, サブルーチン関数IFHDNOによりZバッファの定義を行っておくこと。
また描画対象の多角形のある空間をサブルーチンGCPWNCで定義しておく必要がある。
多角形内の色は, SPをパラメータとする擬似カラー関数により決定される。この関数はサブルーチンCOLORFで定義できる。

サブルーチン 25 IFHDN0

| | | |
|--|-----|------------------------------|
| サブルーチン名 | | IFHDN0 (関数サブルーチン) |
| 機能：隠面処理のためのZバッファを定義する。 | | |
| 呼び出し形式 | | IFHDN0(ZBUF, NI, NJ) |
| パラメータ | 入出力 | パラメータの説明 |
| ZBUF | 入力 | Zバッファのための1次元配列名, 大きさ NI * NJ |
| NI | 入力 | ZバッファのX方向の画素数 |
| NJ | 入力 | ZバッファのY方向の画素数 |
| 備考：Zバッファの大きさはリフレッシュメモリの大きさと等しくすること。サブルーチンIMGT2を使用する前に必ずこの関数サブルーチンによりZバッファの領域を確保すること。 このサブルーチンは形式的に関数サブルーチンにしてあるだけで、 IFHDN0 = 0 でRETURNする。 | | |

サブルーチン 26 IFHDN1

| | | |
|---|----------------|------------------------------------|
| サブルーチン名 | | IFHDN1 (関数サブルーチン) |
| 機能：画素 (IX,IY) のZ座標Zの点が見えるかどうかをZバッファ法により判定する。 見えるとき IFHDN1=1 見えないとき IFHDN1=0 | | |
| 呼び出し形式 | | IFHDN1(IX, IY, Z) |
| パラメータ | 入出力 | パラメータの説明 |
| IX IY Z | 入力 入力 入力 | } 判定画素のX,Y座標位置 (IX, IY) の判定Z座標値 |
| 備考：このサブルーチンを使用する前に関数サブルーチンIFHDN0でZバッファを定義しておくこと。 | | |

サブルーチン 27 GMOVE2

| | | |
|--|----------|--------------------------------|
| サブルーチン名 | | GMOVE2 |
| 機能：現在位置を(X,Y)へ移す。描画はしない。 | | |
| 呼び出し形式 | | CALL GMOVE2(X, Y) |
| パラメータ | 入出力 | パラメータの説明 |
| X Y | 入力 入力 | 新らしい現在位置のX座標値 新らしい現在位置のY座標値 |
| 備考：従来のXYプロッタ用サブルーチンPLOTの CALL PLOT(X, Y, 3) に相当する。 | | |

サブルーチン 28 GLINE2

| | | |
|---|--|---------------------------|
| サブルーチン名 | GLINE2 | |
| 機能 | 現在位置から (X, Y) までを直線で結ぶ。そして (X, Y) を新しい現在位置とする。 | |
| 呼び出し形式 | CALL GLINE2(X, Y) | |
| パラメータ | 入出力 | パラメータの説明 |
| X Y | 入力 入力 | } 現在位置から引かれる線の終点座標 (X, Y) |
| 備考：従来の XYプロッタ用サブルーチン PLOT の CALL PLOT(X, Y, 2) に相当する。 | | |

サブルーチン 29 PLOTS

| | |
|---------|--|
| サブルーチン名 | PLOTS |
| 機能 | XYプロッタ用システムとの整合性を取るためのダミーサブルーチン。何もせずに RETURN する。 |
| 呼び出し形式 | CALL PLOTS |

サブルーチン 30 PLOT

| | | |
|----------------|--|---|
| サブルーチン名 | PLOT | |
| 機能 | XYプロッタ用サブルーチン PLOT に準じイメージデータ領域内で現在位置を指定位置 (X, Y) へ移す。 | |
| 呼び出し形式 | CALL PLOT(X, Y, IPEN) | |
| パラメータ | 入出力 | パラメータの説明 |
| X Y IPEN | 入力 入力 入力 | } 移動先の座標 (X, Y) IPEN = 2 現在位置から (X, Y) まで線を引く。 IPEN = 999 何もせずに RETURN IPEN ≠ -2, 2, 999 線を引かず (X, Y) まで移動 IPEN < 0 のとき (X, Y) を新しい原点 (0.0, 0.0) とする。 |

サブルーチン 31 SYMBOL

| | | |
|---|--|---------------------------|
| サブルーチン名 | SYMBOL | |
| 機能： | XYプロッタ用サブルーチンSYMBOLに準じイメージデータ領域内に英数字を描画する。 | |
| 呼び出し形式 | CALL SYMBOL(X,Y,HIGHT,NBCD,TH,N) | |
| パラメータ | 入出力 | パラメータの説明 |
| X | 入力 | } 最初の文字の左下隅の座標 (X,Y) |
| Y | 入力 | |
| HIGHT | 入力 | 文字の高さ |
| NBCD | 入力 | 描画文字の格納されている文字変数 |
| TH | 入力 | 文字列と X 軸のなす角度。反時計方向が正。単位度 |
| N | 入力 | 文字数。N > 0 |
| 備考：N < 0 のとき NBCD は XY プロッタ用サブルーチン SYMBOL のシンボル表の整数コード番号を指定すること。このとき N キー 1 のときは現在位置から (X,Y) まで直線を引き、N = -1 のときは直線を引かず (X,Y) に指定された 1 文字を描画する。また NBCD ≤ 15 のときは (X,Y) は描画されるシンボルの中心位置となる。 | | |

サブルーチン 32 NUMBER

| | | |
|---------|--|--|
| サブルーチン名 | NUMBER | |
| 機能： | XYプロッタ用サブルーチンNUMBERに準じ、イメージデータ領域内に数値を描画する。 | |
| 呼び出し形式 | CALL NUMBER(X, Y, HIGHT, FPN, TH, N) | |
| パラメータ | 入出力 | パラメータの説明 |
| X | 入力 | } 最初の文字の左下隅の座標 (X,Y) |
| Y | 入力 | |
| HIGHT | 入力 | 文字の高さ |
| FPN | 入力 | 描画する数値 |
| TH | 入力 | 文字列と X 軸のなす角度。反時計方向が正。単位度 |
| N | 入力 | 描画数値の桁数 > 0 のとき、小数点以上、小数点、小数点以下 N 桁を描画 = 0 のとき、小数点以上と小数点を描画 < 0 のとき、小数点以上 N 桁より大きい部分を描画 |

サブルーチン 33 FACTOR

| | | |
|---|---|----------|
| サブルーチン名 | FACTOR | |
| 機能： | XYプロッタ用サブルーチンFACTORに準じ、図全体のスケールファクタを設定する。 | |
| 呼び出し形式 | CALL FACTOR(S) | |
| パラメータ | 入出力 | パラメータの説明 |
| S | 入力 | スケールファクタ |
| 備考：このルーチンを使用しないときは S = 1.0 となる。 PLOT ルーチン内では $X = S \cdot X$ $Y = S \cdot Y$ の変換された X, Y を用いて図が描画される。 | | |

サブルーチン 34 WHERE

| | | | |
|---|-----|---------------------|-------------|
| サブルーチン名 | | WHERE | |
| 機能：XYプロッタ用サブルーチンWHEREに準じ，現在位置とスケールファクタを求める。 | | | |
| 呼び出し形式 | | CALL WHERE(X, Y, S) | |
| パラメータ | 入出力 | パラメータの説明 | |
| X | 出力 | } 求まる現在位置 | 求まるスケールファクタ |
| Y | 出力 | | |
| S | 出力 | | |

サブルーチン 35 NEWPEN

| | | | |
|--|-----|-------------------|--|
| サブルーチン名 | | NEWPEN | |
| 機能：XYプロッタ用サブルーチンNEWPENとの整合性を取るためサブルーチンで線画の色を指定する。 | | | |
| 呼び出し形色 | | CALL NEWPEN(NPEN) | |
| パラメータ | 入出力 | パラメータの説明 | |
| NPEN | 入力 | 色番号 | |
| 備考：実際の色は、 $S=0.1 \cdot \text{FLOAT}(\text{NPEN})$ の値をもつパラメータで擬似カラー関数により求まるものである。 | | | |

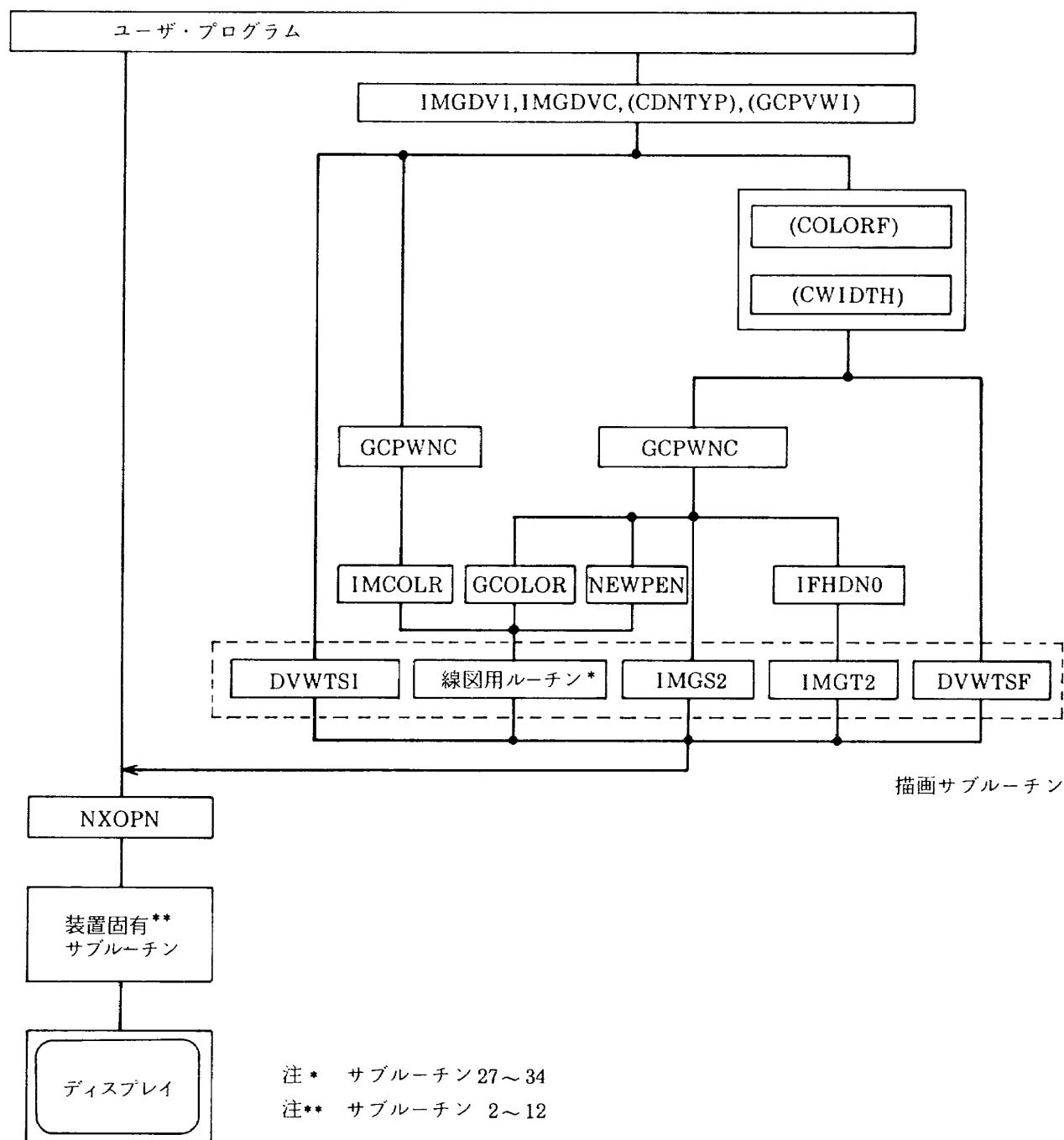
サブルーチン 36 IMCOLR

| | | | |
|---|-----|--------------------------------|--|
| サブルーチン名 | | IMCOLR | |
| 機能：イメージデータ領域内に線画を描画する際の色指定を行う。 | | | |
| 呼び出し形式 | | CALL IMCOLR(ICD) | |
| パラメータ | 入出力 | パラメータの説明 | |
| ICD | 入力 | 各画素要素データを格納されている1次元配列，大きさNCMP. | |
| 備考：ICDの配列の大きさはサブルーチンIMGDVIの要素数パラメータNCMPである。 | | | |

サブルーチン 37 GCOLOR

| | | | |
|---|-----|-----------------|--|
| サブルーチン名 | | GCOLOR | |
| 機能：イメージデータ領域内に線画を描画する際の色指定を行う。色は擬似カラー関数により決定される。 | | | |
| 呼び出し形式 | | CALL GCOLOR (S) | |
| パラメータ | 入出力 | パラメータの説明 | |
| S | 入力 | 色決定パラメータ | |
| 備考：このサブルーチン呼び出し後に描画される線の色は、Sをパラメータとする擬似カラー関数により決定される。 | | | |

以上で示した37のサブルーチンは、利用者のプログラムの中から自由に呼び出すことができるが、描画ルーチンを利用する前に利用環境設定のためのいくつかのサブルーチンをあらかじめ呼び出しておく必要がある。これらサブルーチンの関係を下に示す。



なお、これらのサブルーチンは現在、プログラム・ライブラリーとして次の2つのファイルに格納してある。

GCP. GCPNXL (サブルーチン 1～12および27～35)

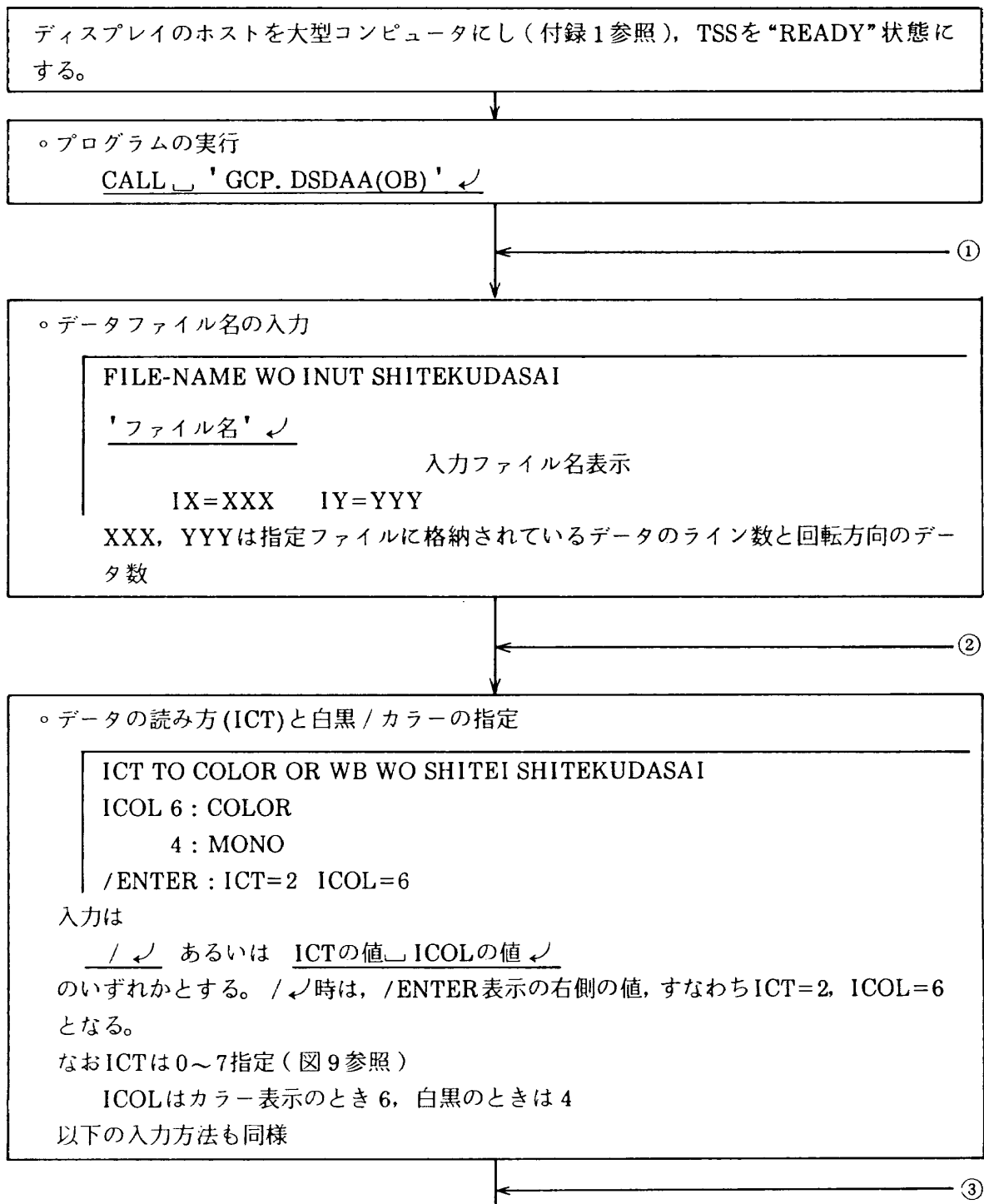
GCP. GCPLIB (サブルーチン 13～26, 36, 37)

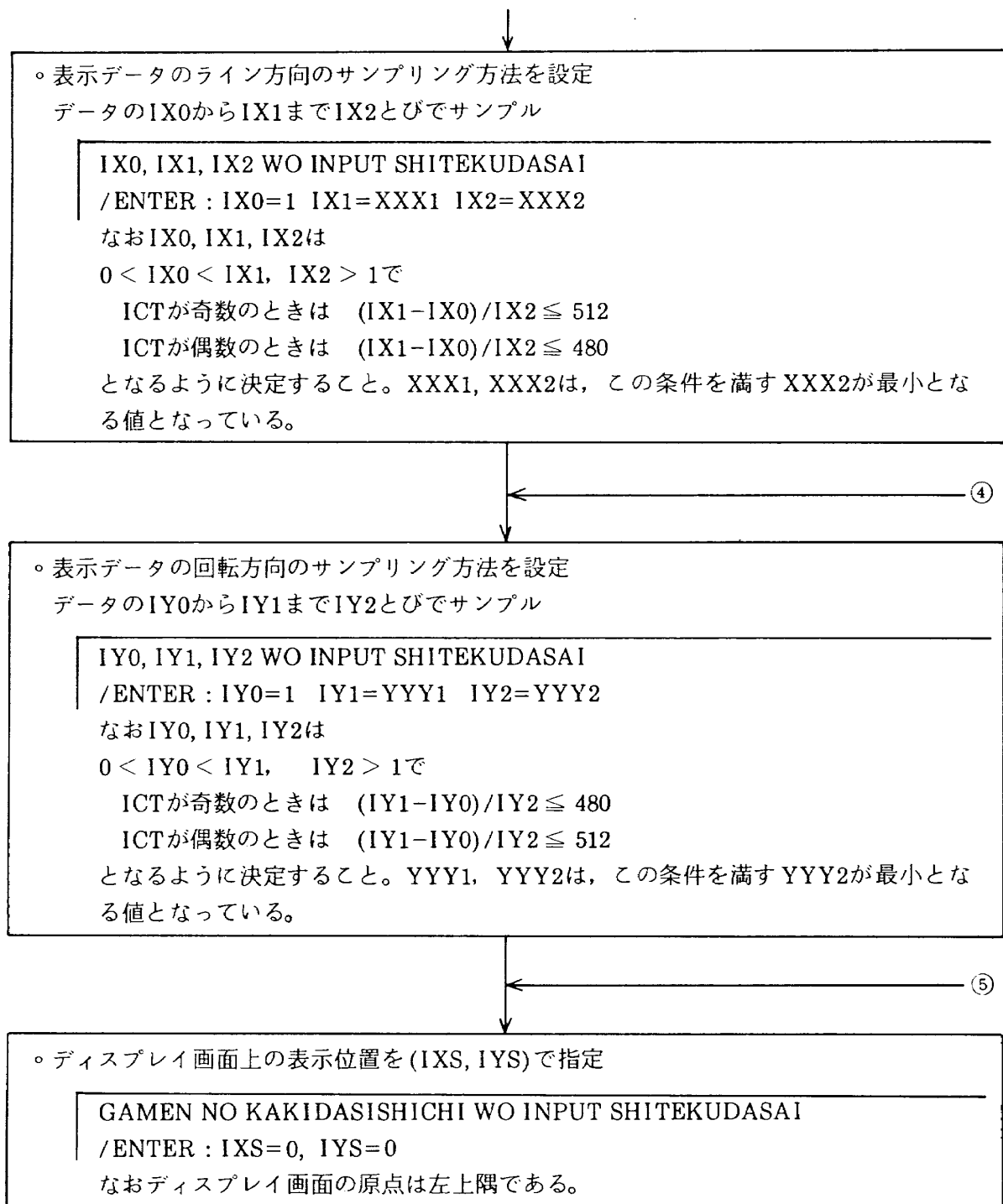
付 録 4

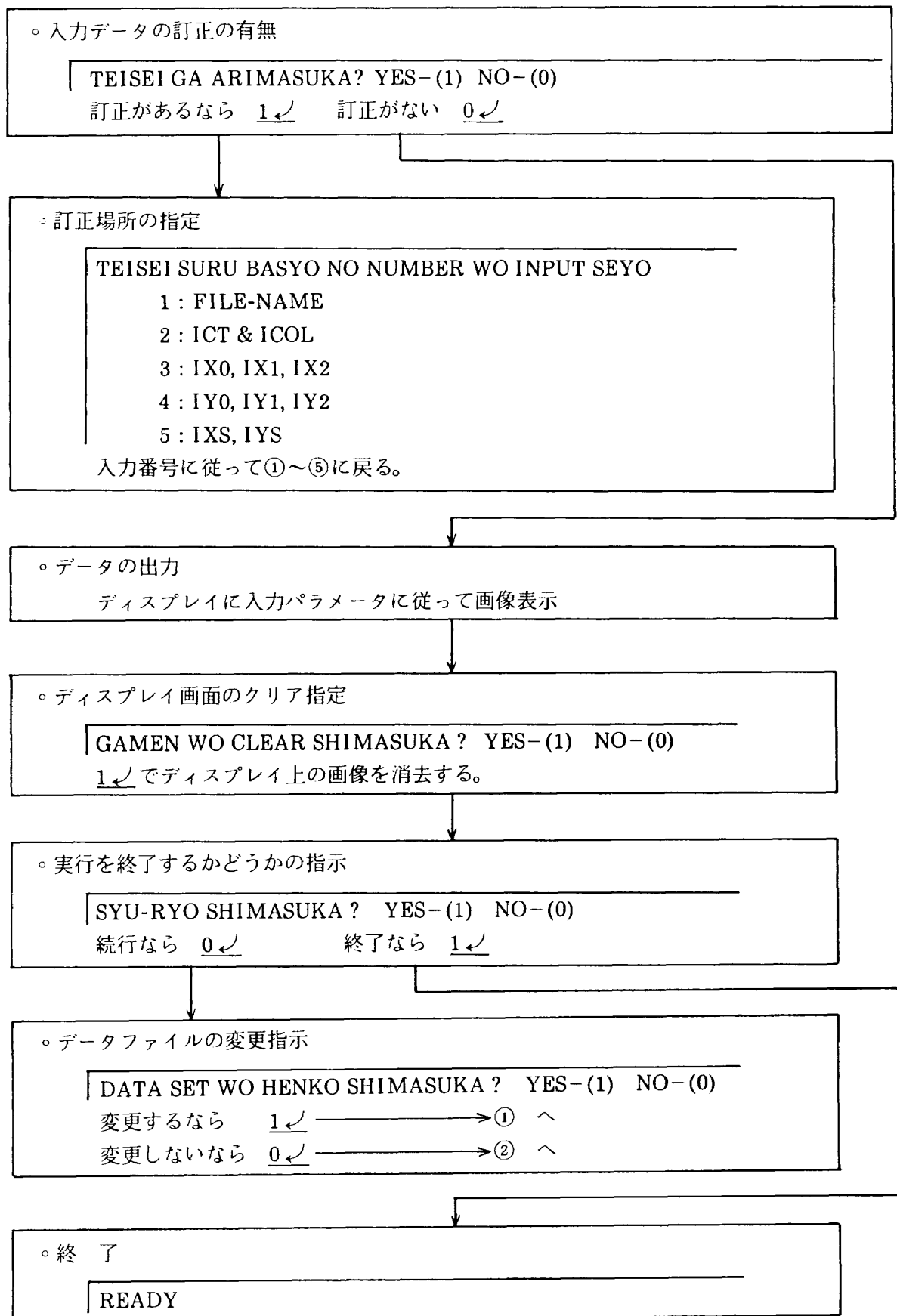
ドラムスキャナによりデジタル化されたデータの表示用プログラム GCP. DSDAA の使用法を述べる。
表示しようとするデータはあらかじめ大型コンピュータのディスクファイル内に格納しておく必要がある。
利用者はこのプログラムに

1. データ格納ファイル名
2. データの読み方（描画配列順序）と白黒 / カラーの別
3. データの表示範囲
4. 表示位置

の情報をキーボードから入力する。このための手順を以下に示す。下線部が入力操作を示す。







このプログラム“GCP, DSDAA”のリストを以下に示す。

```

DIMENSION IBUF(184320),IP(1024)
DIMENSION IXYD(3)
CHARACTER*40 FNAME
CHARACTER*77 ALLO
CHARACTER*1 CZZ
DATA CZZ/200/
COMMON/DSCMND/IDSC(274)
443   ICT=2
      IX0=1
      IX2=1
      IX5=0
      IY0=1
      IY2=1
      IY5=0
      ICL=1
      IEND=0
      IDA=0
C
C *****
C *   FILE-NAME INPUT   *
C *****
444   WRITE(6, '(1H1, ''FILE-NAME WU INPUT SHITEKUDASAI '' )')
      READ(5, '(A40)')FNAME
      ALLO(1:26)='ALLOC DD(FT20F001) DLD DA('
      J=1
      DO 40 I=1,40
        IF (FNAME(I:1).EQ.CZZ .OR. FNAME(I:1).EQ.' ') GO TO 40
        FNAME(J:J)=FNAME(I:1)
        J=J+1
40    CONTINUE
      J=J-1
      L1=26+J
      ALLO(27:L1)=FNAME(1:J)
      ALLO(L1+1:L1+11)=' ' REUSE SHR'
      L=L1+11
      WRITE (6, '(1H ,A40)')ALLO(27:L1)
      CALL IPFCMD(IRC1,IRC2,ALLO,L)
      OPEN(20,FILE=FNAME(2:J-1),STATUS='OLD',ACCESS='SEQUENTIAL'
+       ,FORM='UNFORMATTED')
      IO=20
777 CALL DSHD(IO,IX,IY)
      WRITE(6,*) 'IX=',IX,'IY=',IY
C *****
C *   ICT TO COLOR OR MONO INPUT   *
C *****
      ICOL=IDSC(7)
      IF (ICOL.NE.4 .AND. ICOL.NE.6)ICOL=4
      IF (ICOL.EQ.4)GO TO 222
      WRITE(6, '(1H1, ''ICT TO COLOR OR WB WU SHITEI SHITEKUDASAI '' )')
      WRITE(6, '(1H ,'' ICOL  6 : COLOR '' )')
      WRITE(6, '(1H ,''      4 : MONO  '' )')
      WRITE(6,*) '/ ENTER : ICT=',ICT,' ICOL=',ICOL
      READ(5,*)ICT,ICOL
      IF (ICT.LT.0 .OR. ICT.GT.7) GO TO 222
      WRITE(6,*) '      ICT=',ICT,' ICOL=',ICOL
      GO TO 149
222  WRITE(6, '(1H1, ''ICT WU INPUT SHITEKUDASAI '' )')
      WRITE(6,*) '/ ENTER : ICT=',ICT
      READ(5,*)ICT
      IF (ICT.LT.0 .OR. ICT.GT.7) GO TO 222
      WRITE(6,*) '      ICT=',ICT

```

```

149  IX1=IX
    IY1=IY
C *****
C *   IX2 & IY2 NO SYU-RYAKUCHI NO KEISAN *
C *****
    IA=512
    IB=480
    IF(MOD(ICT,2).EQ.0)GO TO 150
    IA=480
    IB=512
150  IF(IX1.LE.IA)GO TO 151
    IM=MOD(IX1,IA)
    IF(IM.EQ.0)THEN
        IX2=IX1/IA
    ELSE
        IX2=IX1/IA+1
    ENDIF
151  IF(IY1.LE.IB)GO TO 111
    IM=MOD(IY1,IB)
    IF(IM.EQ.0)THEN
        IY2=IY1/IB
    ELSE
        IY2=IY1/IB+1
    ENDIF
C *****
C *   IX0,IX1,IX2 INPUT *
C *****
111  WRITE(6, '(1H1, ''IX0,IX1,IX2 WO INPUT SHITEKUDASAI''))
    WRITE(6,*)'/ ENTER : IX0=',IX0,' IX1=',IX1,' IX2=',IX2
    READ(5,*)IX0,IX1,IX2
    WRITE(6,*)' IX0=',IX0,' IX1=',IX1,' IX2=',IX2
C *****
C *   IY0,IY1,IY2 INPUT *
C *****
555  WRITE(6, '(1H1, ''IY0,IY1,IY2 WO INPUT SHITEKUDASAI''))
    WRITE(6,*)'/ ENTER : IY0=',IY0,' IY1=',IY1,' IY2=',IY2
    READ(5,*)IY0,IY1,IY2
    WRITE(6,*)' IY0=',IY0,' IY1=',IY1,' IY2=',IY2
C *****
C *   IXS, IYS, INPUT *
C *****
333  WRITE(6, '(1H1, ''GAMEN NO KAKIDASHIICHI WO INPUT SHITEKUDASAI''))
    WRITE(6, '(1H , ''/ ENTER : IXS=',IXS,' IYS=',IYS,')')IXS,IYS
    READ(5,*)IXS,IYS
    WRITE(6, '(1H , '' IXS=',IXS,' IYS=',IYS,')')IXS,IYS
C *****
C *   TEISEI ? *
C *****
    WRITE(6, '(1H , ''TEISEI GA ARIMASUKA ? YES - (1) ,NO - (0)''))
    READ(5,*)IG0
    IF(IG0.EQ.0)GO TO 500
    WRITE(6, '(1H , ''TEISEI SURU BASYO NO NUMBER WO INPUT SHITEKUDASA
+1''))
    WRITE(6, '(1H , '' 1 : FILE-NAME''))
    WRITE(6, '(1H , '' 2 : ICT & ICUL''))
    WRITE(6, '(1H , '' 3 : IX0, IX1, IX2, '))
    WRITE(6, '(1H , '' 4 : IY0, IY1, IY2, '))
    WRITE(6, '(1H , '' 5 : IXS, IYS, '))
    READ(5,*)IG01
    IF(IG01.EQ.1)THEN
        CLOSE(20)
        GO TO 444

```

```

ENDIF
IF(IGU1.EV.2)GO TO 777
IF(IGU1.EV.3)GO TO 111
IF(IGU1.EV.4)GO TO 555
IF(IGU1.EV.5)GO TO 333
C
500 WRITE(6,'(1H ,A40)')ALLO(27:L1)
WRITE(6,'(1H ,''ICT='',I1,''' ICOL='',I1)'')ICT,ICOL
WRITE(6,'(1H ,''IX0='',I3,''' IX1='',I3,''' IX2='',I3)'')
+ IX0,IX1,IX2
WRITE(6,'(1H ,''IY0='',I3,''' IY1='',I3,''' IY2='',I3)'')
+ IY0,IY1,IY2
WRITE(6,'(1H ,''IXS='',I3,''' IYS='',I3)'')IXS,IYS
IX=(IX1-IX0+1)/IX2
IY=(IY1-IY0+1)/IY2
CALL RDSIT(IX0,IY0,IX2,IY2,IX,IY)
CALL NXUPN
C
CALL IMGDV0(IBUF)
CALL IMGDV1(IBUF)
IXX=IX
IYY=IY
IF(MOD(ICT,2).NE.1)GO TO 511
IXX=IY
IYY=IX
511 IF(IXX.GT.512)IXX=512
IF(IYY.GT.480)IYY=480
IF(IXS.GE.512-IXX)IXS=512-IXX
IF(IYS.GE.480-IYY)IYS=480-IYY
WRITE(6,'(1H ,''IMGDV1 : '',4I5)'')IXX,IYY,IXS,IYS
CALL IMGDV1(IXX,IYY,24,3)
CALL CDNTYP(ICT)
DO 10 I=1,IX
CALL DSDATA(IP,NM)
DO 20 J=1,IY
CALL DBITE(IP(J),ICC,IBB,IAA,I1)
IF(ICOL.NE.6)GO TO 888
IAA=255-IAA
IBB=255-IBB
ICC=255-ICC
IXYD(1)=IAA
IXYD(2)=IBB
IXYD(3)=ICC
GO TO 89
888 DO 88 KK=1,3
88 IXYD(KK)=255-I1
89 CALL DVWTSI(I-1,J-1,IXYD)
C
CALL NXPUT(IBUF,J,1,255-IAA,255-IBB,255-ICC)
20 CONTINUE
10 CONTINUE
C
CALL NXWRT(IBUF)
CALL NXWT2(IBUF,IXS,IYS,IXX,IYY,3)
C *****
C * GAMEN CLEAR ? *
C *****
WRITE(6,*)'GAMEN WD CLEAR SHIMASUKA ? YES - (1) NO - (0)'
READ(5,*)ICL
IF(ICL.EQ.1)GO TO 160
GO TO 250
160 CALL NXCLR

```

```
C *****
C *   SYU-RYU- ?   *
C *****
  250 WRITE(6, '(1H1, ''SYU-RYU- SHIMASUKA ? YES - (1) NO - (0)'''))
      READ(6, *) IEND
      IF (IEND.EQ.1) GO TO 999
C *****
C *   DATA SET HENKU- ?   *
C *****
      WRITE(6, '(1H1, ''DATA SET WD HENKO- SHIMASUKA ? YES - (1) NO - (0)
+ '''))
      READ(6, *) IDA
      IF (IDA.EQ.1) THEN
        CLOSE(20)
        GO TO 443
      ELSE
        REWIND 20
        GO TO 777
      ENDIF
999 CALL NXCLS
      CLOSE(20)
      STOP
      END
```

リスト 1 (4/4)

付 録 5

擬似カラー関数サブルーチンの例を示す。

擬似カラー関数は整数型関数で、I, Sの2つのパラメータをもつ。すなわちIFNC(I,S)の型式をとる。

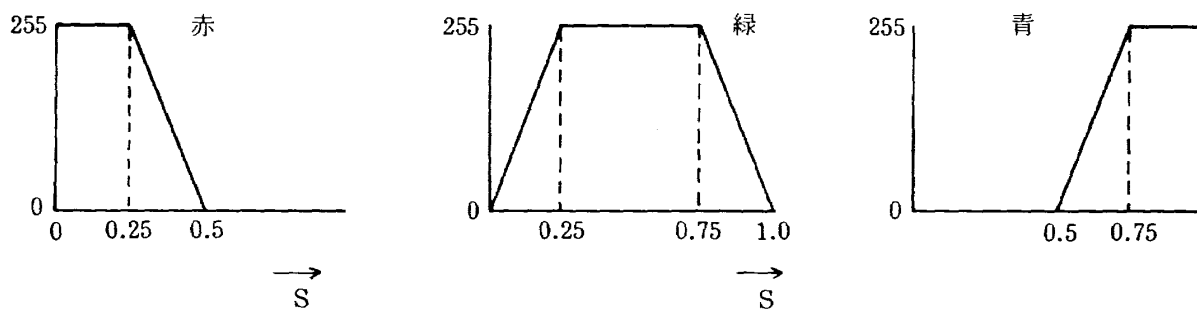
I: 画素のデータ要素番号, I=0のとき初期化

S: データ要素I番の関数値を決定するスカラー量

ここでNEXUS 6410用の擬似カラー関数を作る。

今、仮にSの範囲を $0 \leq S \leq 1.0$ とする。

関数の赤要素 (I=1), 緑要素 (I=2), 青要素 (I=3) を下図のようにする。



また、I=0のとき、パラメータSの範囲をサブルーチンCWIDTHで設定された範囲になるように、関数を初期化する。

この擬似カラー関数サブルーチンのコーディング例は次のようになる。

```

C
C *** FUNCTION IRFNC(I,XX) ***
C
      FUNCTION IRFNC(I,XX)
      COMMON/GCPCMN/IGC1(12),NPC,IGC2(8),ICMAX,IGC3(59),CF,CFMIN,CFBN
      +           ,IGC4(44)
      DATA C10,C20,C30,C40,C50/0.00,0.25,0.50,0.75,1.0/
      IND=MOD(I,3)
      IF(IND.NE.1)GO TO 20
      IF(XX.LE.C2)THEN
        IRFNC=N255
      ELSE IF(XX.LE.C3)THEN
        IRFNC=F255*(C3-XX)*C32
      ELSE
        IRFNC=0
      ENDIF
      RETURN
20    IF(IND.NE.2)GO TO 30
      IF(XX.LE.C1)THEN
        IRFNC=0
      ELSE IF(XX.LE.C2)THEN
        IRFNC=F255*(XX-C1)*C21
      ELSE IF(XX.LE.C4)THEN
        IRFNC=N255
      ELSE IF(XX.LE.C5)THEN
        IRFNC=F255*(C5-XX)*C54
      ELSE
        IRFNC=0
      
```



```
ENDIF
RETURN
30 IF(I.EQ.0)GO TO 40
IF(XX.LE.C3)THEN
  IRFNC=0
  ELSE IF(XX.LE.C4)THEN
    IRFNC=F255*(XX-C3)*C43
  ELSE
    IRFNC=1.255
ENDIF
RETURN
40 IF(NPC.LE.0)THEN
  N255=255
  ELSE
    N255=NPC
ENDIF
F255=N255
IF(ICMAX.EQ.1)THEN
  C1=CFMIN
  C2=CFBN*(C20-C10)/(C50-C10)+C1
  C3=CFBN*(C30-C10)/(C50-C10)+C1
  C4=CFBN*(C40-C10)/(C50-C10)+C1
  C5=CFBN+C1
  ELSE
    C1=C10
    C2=C20
    C3=C30
    C4=C40
    C5=C50
ENDIF
C21=1.0/(C2-C1)
C32=1.0/(C3-C2)
C43=1.0/(C4-C3)
C54=1.0/(C5-C4)
RETURN
END
```

リスト 2 (2/2)

なお、擬似カラー関数サブルーチン名は任意のものでよいが、このときサブルーチンCOLORFにより関数名を定義する必要がある。サブルーチンCOLORFの定義がないときは、上で示したリスト2の関数が擬似関数となる。

付 録 6

(1) 図16のためのプログラム

```

PARAMETER (II=55, KK=30)
DIMENSION PP(II, KK), XX(II, KK), YY(II, KK)
DIMENSION XSAN(4), YSAN(4), ZSAN(4)
DIMENSION IBUF(184320)
EXTERNAL IRFNC
READ(10) NI, NK, ((XX(I, K), I=1, NI), K=1, NK)
READ(10) NI, NK, ((YY(I, K), I=1, NI), K=1, NK)
READ(10) NI, NK, ((PP(I, K), I=1, NI), K=1, NK)

```

C

```

CALL IMGDVC(IBUF)
CALL IMGDAI(512, 480, 24, 3)
CALL GCPWND(-3.0, -3.0, 2.5, 3.0)
CALL GCPVWI(0, 0, 511, 479)
CALL CDNTYP(2)
CALL MAXIN2(PP, 1, 1, NI, NK, FMAX, FMIN)
CALL COLORF(IRFNC)
CALL CWIDTH(FMAX, FMIN)
DO 602 I=1, NI-1
DO 603 JJ=1, NK-1
XSAN(1)=XX(I, JJ)
XSAN(2)=XX(I+1, JJ)
XSAN(3)=XX(I+1, JJ+1)
XSAN(4)=XX(I, JJ+1)
YSAN(1)=YY(I, JJ)
YSAN(2)=YY(I+1, JJ)
YSAN(3)=YY(I+1, JJ+1)
YSAN(4)=YY(I, JJ+1)
ZSAN(1)=PP(I, JJ)
ZSAN(2)=PP(I+1, JJ)
ZSAN(3)=PP(I+1, JJ+1)
ZSAN(4)=PP(I, JJ+1)
IND=4
CALL IMGS2(XSAN, YSAN, ZSAN, IND)
603 CONTINUE
602 CONTINUE
CALL NXDPN
CALL NXWRT(IBUF)
CALL NXCLS
STOP
END

```

リスト 3

図16は55×30の格子上で計算されたデータを表示したものである。格子点で定義されているデータは、X座標値、Y座標値、マッハ数である。これらのデータセット識別番号10のファイルにX座標、Y座標、マッハ数の順に格納されている。

プログラムの流れは次の通りである。

- (1) 各配列の定義
- (2) 各データを配列XX, YY, PPに読み取る。
- (3) イメージデータ領域を配列IBUFに定義。
- (4) リフレッシュメモリ、ウィンドウ、ビューポート、描画時の配列順序の定義。
- (5) データPPの最大値、最小値を求める[†]。

注[†] この処理は2次元配列内の最大値、最小値を求めるサブルーチンMAXIN2により行っている。

- (6) 擬似カラー関数, パラメータ範囲の定義。
- (7) 格子を構成する各セルを1つつ取り出し, イメージデータ領域へ描画。
- (8) イメージデータをディスプレイへ転送。

(2) 図17のためのプログラム

```

PARAMETER (II=55, KK=17)
DIMENSION XDM(II, KK), YDM(II, KK), ZDM(II, KK), PDM(II, KK)
DIMENSION XSAN(4), YSAN(4), ZSAN(4), FSAN(4)
DIMENSION DMP(512*480), IBUF(184320)
EXTERNAL IRFNC
READ(11) NI, NJ, ((XDM(I, K), I=1, NI), K=1, NJ)
READ(11) NI, NJ, ((YDM(I, K), I=1, NI), K=1, NJ)
READ(11) NI, NJ, ((ZDM(I, K), I=1, NI), K=1, NJ)
READ(11) NI, NJ, ((PDM(I, K), I=1, NI), K=1, NJ)

C
CALL IMGDV(IBUF)
CALL IMGDVI(512, 480, 24, 3)
CALL GCPWND(-0.5, -1.0, 1.4, -0.5 + (1.4 + 0.5) * 480. / 512.0)
CALL GCPVWI(0, 0, 511, 479)
CALL CDNTYP(2)
THX=0.0
THY=45.0
THZ=0.0
CALL KAITEN(XDM, YDM, ZDM, NI, NJ, NI, NJ, THX, THY, THZ, 1, 0, 1, 0)
CALL MAXIN2(PDM, 1, 1, NI, NJ, FMAX, FMIN)
CALL COLDRF(IRFNC)
CALL CWIDTH(FMAX, FMIN)
I=IFHDN0(DMP, 512, 480)
DO 602 I=1, NI-1
DO 603 JJ=1, NJ-1
XSAN(1)=XDM(I, JJ)
XSAN(2)=XDM(I+1, JJ)
XSAN(3)=XDM(I+1, JJ+1)
XSAN(4)=XDM(I, JJ+1)
YSAN(1)=YDM(I, JJ)
YSAN(2)=YDM(I+1, JJ)
YSAN(3)=YDM(I+1, JJ+1)
YSAN(4)=YDM(I, JJ+1)
ZSAN(1)=ZDM(I, JJ)
ZSAN(2)=ZDM(I+1, JJ)
ZSAN(3)=ZDM(I+1, JJ+1)
ZSAN(4)=ZDM(I, JJ+1)
FSAN(1)=PDM(I, JJ)
FSAN(2)=PDM(I+1, JJ)
FSAN(3)=PDM(I+1, JJ+1)
FSAN(4)=PDM(I, JJ+1)
IND=4
CALL IMGT2(XSAN, YSAN, ZSAN, FSAN, IND)
603 CONTINUE
602 CONTINUE
CALL NXDPN
CALL NXWRT(IRUF)
CALL NXCLS
STOP
END

```

リスト 4

図17は表示データが3次元である。プログラムの構造, 流れは, Zバッファの定義と表示データをY軸に45°回転させる処理を追加した以外はリスト3と同じである。データの回転処理はサブルーチン KAITEN で行っている。

(3) 図18のためのプログラム

```

PARAMETER (II=55, KK=30)
DIMENSION IBUF(184320), IXYD(3)
DIMENSION PP(II, KK), XX(II, KK), YY(II, KK)
DIMENSION JBCD(1)
EXTERNAL IRFNC
READ(10) NI, NK, ((XX(I, K), I=1, NI), K=1, NK)
READ(10) NI, NK, ((YY(I, K), I=1, NI), K=1, NK)
READ(10) NI, NK, ((PP(I, K), I=1, NI), K=1, NK)

C
CALL IMGDVC(IBUF)
CALL IMGDVI(512, 480, 24, 3)
CALL GCPWND(-3.5, -3.5, 2.5, 3.0)
CALL GCPVWI(0, 0, 511, 479)
CALL CDNTYP(2)
CALL CULURF(IRFNC)
CALL MAXIN2(PP, 1, 1, NI, NK, PMAX, PMIN)
CALL CWIDTH(PMAX, PMIN)
IXYD(1)=150
IXYD(2)=150
IXYD(3)=150
CALL IMCOLR(IXYD)
CALL DGRID2(XX, YY, 55, 30, NI, NK)
NNN=50
W=(PMAX-PMIN)/FLOAT(NNN-1)
CALL CLDATA(PMAX, W, PMIN)
CALL CONTO2(XX, YY, PP, 55, 30, NI, NK)
IXYD(1)=255
IXYD(2)=255
IXYD(3)=255
CALL IMCOLR(IXYD)
JBCD(1)=999
CALL LETTER(0.80, 0.0, -999.0, JBCD, 1.0, -1)
CALL LETTER(-2.5, -3.0, 0.25, 'MACH NUMBER CONTOUR', 0.0, 19)
CALL NXOPN
CALL NXWRT(IBUF)
CALL NXCLS
STOP
END
    
```

リスト 5

図18は、図16を等高線表示し、さらに格子の作図、説明文を追加したものである。リスト2の描画部分を格子作図、文字作図のサブルーチンに置き替えてある。格子はサブルーチンDGRID2で、等高線はサブルーチンCONTO2で、文字はXYプロッタ用アプリケーション・サブルーチンLETTERで描画している。格子はR・G・Bそれぞれ150、文字はR・G・Bそれぞれ255の色指定を行っている。またサブルーチンCLDATAは等高線の値を指定するためのものである。

概 刊 資 料

| | | | |
|--------|---|-----------|--|
| TM-550 | 宇宙ステーションの利用について | 1985年11月 | 宇宙ステーション ワーキンググループ |
| TM-551 | FJR710/600S エンジン第一次空中試験概要報告 | 1985年12月 | STOL プロジェクト推進 本部エンジン技術開発室 エンジン空中試験チーム |
| TM-552 | ファンジェット STOL 実験機自動飛行制御システム (SCAS) 用センサの認定試験結果 | 1986年 5 月 | STOL プロジェクト推進 本部実験機開発室操縦システム技術開発チーム |
| TM-553 | 低騒音 STOL 実験機の全機地上振動試験 (第 2 報: 二次試験) | 1986年 8 月 | 小松 敬治, 佐野 政明 甲斐 高志, 安藤 泰勝 峯岸 正勝, 森田 甫之 |
| TM-554 | 低騒音 STOL 実験機 (飛鳥) 用 HUD (ヘッド・アップ・ディスプレイ) の第一次シミュレーション評価試験 | 1986年10月 | STOL プロジェクト推進 本部操縦システム開発チーム 飛行解析チーム |
| TM-555 | 有限要素法を用いた静磁場解析プログラム MA FIA の開発 | 1986年11月 | 早川 幸男 |
| TM-556 | カプセル型回収体の極超音速動安定特性試験 | 1986年11月 | 山本 行光, 渡辺 光則 野村 茂昭, 小山 忠勇 穂積 弘一, 吉沢 昭 伊藤 哲一, 高塚 均 |
| TM-557 | カプセル型回収体の極超音速オイル・フロウ試験 | 1986年11月 | 山本 行光, 渡辺 光則 野村 茂昭, 小山 忠勇 穂積 弘一, 吉沢 昭 |
| TM-558 | STOL 実験機用フライトディレクタ・システムの基礎設計 | 1986年12月 | 田中 敬司 |
| TM-559 | 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 | 1986年12月 | 末松 和代, 磯部 俊夫 |

航空宇宙技術研究所資料 560 号

昭和 61 年 12 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 丁目 44 番地 1
電話武蔵野三鷹(0422)47-5911(大代表)〒182
印刷所 株式会社 三興印刷
東京都新宿区信濃町 12 三河ビル

