地球流体電脳ライブラリ

GRPH1

(図形処理下位パッケージ)

地球流体電脳倶楽部

1995 年 6 月 9 日 (DCL-5.0)

目 次

1 概要 **1**
1.1 はじめに . . . 1
1.2 デバイスとフレーム . . . 2
1.3 座標系と変換 . . . 2
1.4 2 次元正規化変換 . . . 4
1.4.1 座標系の種類 . . . 4
1.4.2 地図投影法 . . . 5
1.4.3 正規変換パラメタ . . . 6
1.4.4 U 座標系の設定 . . . 8
1.5 3 次元正規化変換 . . . 9
1.5.1 座標系の種類 . . . 9
1.5.2 3 次元正規変換パラメタ . . . 9
1.5.3 座標系の設定 . . . 9
1.6 透視変換 . . . 10
1.6.1 変換パラメタ . . . 10
1.6.2 透視変換の設定 . . . 11
1.7 出力プリミティブ . . . 11
1.7.1 ラインインデクス . . . 12
1.7.2 ラインタイプ . . . 12
1.7.3 トーン番号 . . . 13
1.7.4 補間とクリッピング . . . 13
1.7.5 トーンのクリッピングに関する注意 . . . 14
1.8 レイアウト . . . 15

2 SGPACK : ユーザーインターフェイス **16**
2.1 概要 . . . 16
2.2 サブルーチンのリスト . . . 19
2.3 関数のリスト . . . 20
2.4 サブルーチンの説明 : コントロール . . . 21
2.4.1 SGOPN . . . 21
2.4.2 SGFRM . . . 21
2.4.3 SGCLS . . . 22
2.4.4 SG*p*GET/SG*p*SET(SG*p*STX) . . . 22
2.4.5 SGPWSN . . . 27

2.5 サブルーチンの説明：正規化変換 . . . 29
2.5.1 SGSVPT . . . 29
2.5.2 SGSWND/SGSSIM/SGSMPL . . . 29
2.5.3 SGSTRN . . . 30
2.5.4 SGSTRF . . . 31
2.6 サブルーチンの説明：ポリラインプリミティブ . . . 32
2.6.1 SGPLZU/SGPLZV/SGPLZR . . . 32
2.6.2 SGPLU/SGPLV/SGPLR . . . 32
2.6.3 SGSPLT . . . 33
2.6.4 SGSPLI . . . 34
2.6.5 SGSPLC . . . 34
2.6.6 SGSPLS . . . 35
2.6.7 SGNPLC . . . 35
2.7 サブルーチンの説明：ポリマーカープリミティブ . . . 36
2.7.1 SGPMZU/SGPMZV/SGPMZR . . . 36
2.7.2 SGPMU/SGPMV/SGPMR . . . 36
2.7.3 SGSPMT . . . 37
2.7.4 SGSPMI . . . 38
2.7.5 SGSPMS . . . 38
2.8 サブルーチンの説明：テキストプリミティブ . . . 39
2.8.1 SGTXZU/SGTXZV/SGTXZR . . . 39
2.8.2 SGTXU/SGTXV/SGTXR . . . 40
2.8.3 SGSTXS . . . 40
2.8.4 SGSTXR . . . 41
2.8.5 SGSTXC . . . 41
2.8.6 SGSTXI . . . 42
2.9 サブルーチンの説明：トーンプリミティブ . . . 43
2.9.1 SGTNZU/SGTNZV/SGTNZR . . . 43
2.9.2 SGTNU/SGTNV/SGTNR . . . 44
2.9.3 SGSTNP . . . 44
2.10 サブルーチンの説明：ラインサブプリミティブ . . . 46
2.10.1 SGLNZU/SGPLZV/SGPLZR . . . 46
2.10.2 SGLNU/SGLNV/SGLNR . . . 46
2.10.3 SGSLNI . . . 47
2.11 サブルーチンの説明：アローサブプリミティブ . . . 48
2.11.1 SGLAZU/SGLAZV/SGLAZR . . . 48
2.11.2 SGLAU/SGLAV/SGLAR . . . 48
2.11.3 SGSLAT . . . 49
2.11.4 SGSLAI . . . 49
2.12 関数の説明：DCL 文字関数 . . . 51
2.12.1 CSGI . . . 51
2.12.2 ISGC . . . 52

3 SCPACK：3 次元ユーザーインターフェイス 53
3.1 概要 . . . 53
3.2 サブルーチンのリスト . . . 54
3.3 サブルーチンの説明：3 次元正規化変換 . . . 56
3.3.1 SCSVPT/SCSWND/SCSLOG/SCSORG . . . 56
3.3.2 SCSTRN . . . 57
3.3.3 SCSTRF . . . 57
3.4 サブルーチンの説明：ポリラインプリミティブ . . . 58
3.4.1 SCPLZU/SCPLZV . . . 58
3.4.2 SCPLU/SCPLV . . . 58
3.4.3 SCSPLI . . . 59
3.5 サブルーチンの説明：ポリマーカープリミティブ . . . 60
3.5.1 SCPMZU/SCPMZV . . . 60
3.5.2 SCPMU/SCPMV . . . 60
3.5.3 SCSPMT . . . 61
3.5.4 SCSPMI . . . 61
3.5.5 SCSPMS . . . 62
3.6 サブルーチンの説明：トーンプリミティブ . . . 63
3.6.1 SCTNZU/SCTNZV . . . 63
3.6.2 SCTNU/SCTNV . . . 63
3.6.3 SCSTNP . . . 64

4 SLPACK：レイアウトルーチン 65
4.1 概要 . . . 65
4.2 サブルーチンのリスト . . . 67
4.3 主要サブルーチンの説明 . . . 67
4.3.1 SLINIT . . . 67
4.3.2 SLSIZE . . . 67
4.3.3 SLFORM . . . 68
4.3.4 SLDIV . . . 68
4.3.5 SLMGN . . . 69
4.3.6 SLRAT . . . 69
4.3.7 SLSTTL . . . 69
4.3.8 SLPVPR/SLPWWR/SLPWVR/SLPVRC/SLPWWC/SLPWVC . . . 70
4.3.9 その他のサブルーチン . . . 71

5 SZPACK：基本描画ルーチン 72
5.1 概要 . . . 72
5.2 サブルーチンのリスト . . . 73
5.2.1 インターフェイスルーチン . . . 73
5.2.2 基本描画ルーチン . . . 73

6 STPACK：変換関数 75
6.1 概要 . . . 75
6.1.1 ユーザー定義関数 . . . 75
6.2 サブルーチンのリスト . . . 78
6.3 サブルーチンの説明：座標変換 . . . 78
6.3.1 STFTRF/STITRF/STSTRF/STQTRF . . . 78
6.3.2 STFPR2/STIPR2/STSPR2 . . . 79
6.3.3 STFTR3 . . . 79
6.3.4 STFPR3/STIPR3/STSPR3 . . . 80
6.3.5 STFWTR/STWTR/STSWTR/STQWTR . . . 80
6.4 サブルーチンの説明：座標変換 (下請け) . . . 81
6.4.1 STFTRN/STITRN/STSTRN . . . 81
6.4.2 STFROT/STIROT/STSROT . . . 82
6.4.3 STFRAD/STIRAD/STSRAD . . . 82

7 SWPACK：機種依存ルーチン 83
7.1 概要 . . . 83
7.2 サブルーチンのリスト . . . 84
7.3 サブルーチンの説明：コントロールルーチン . . . 86
7.3.1 SWDOPN/SWDCLS . . . 86
7.3.2 SWPOPN/SWPCLS . . . 86
7.3.3 SWOOPN/SWOCLS . . . 87
7.4 サブルーチンの説明：描画ルーチン . . . 87
7.4.1 SWSWDI/SWSCLI . . . 87
7.4.2 SWGOPN/SWGCLS . . . 88
7.4.3 SWGMOV/SWGPLT . . . 88
7.4.4 SWGTON . . . 88
7.4.5 SWIOPN/SWIDAT/SWICLS . . . 89
7.5 サブルーチンの説明：マウス . . . 89
7.5.1 SWQPNT . . . 89
7.6 サブルーチンの説明：座標変換 . . . 90
7.6.1 SWFINT/SWIINT . . . 90
7.7 サブルーチンの説明：問い合わせルーチン . . . 90
7.7.1 SWQWDC/SWQCLC/SWQTNC/SWQIMC/SWQPTC . . . 90
7.7.2 SWQFNM . . . 91
7.7.3 SWQRCT . . . 91
7.7.4 SWSROT . . . 92
7.8 サブルーチンの説明：内部変数管理ルーチン . . . 92
7.8.1 SW*p*GET/SW*p*SET(SW*p*STX)/SWCGET/SWCSET(SWCSTX) . . . 92
7.8.2 使用例 . . . 95
7.9 各種データベース . . . 97
7.9.1 フォント . . . 97
7.9.2 カラー・トーン番号変換テーブル . . . 97
7.9.3 カラーマップ . . . 97

7.9.4 ビットマップ 98

8 付録 99

8.1 各種テーブル 99

8.1.1 ラインタイプ 100

8.1.2 フォントテーブル 1 101

8.1.3 フォントテーブル 2 102

8.1.4 トーンパターン 0 103

8.1.5 トーンパターン 1 104

8.1.6 トーンパターン 2 105

8.1.7 トーンパターン 3 106

8.1.8 トーンパターン 4 107

8.1.9 トーンパターン 5 108

8.1.10 トーンパターン 6 109

謝辞 110

第 1 章

概要

1.1 はじめに

GRPH1 (図形処理下位パッケージ) は座標変換をしたり, 線分などの基本的な図形を描画するパッケージである. このパッケージは, MATH1 と MISC1 のライブラリを使用する. GRPH1 は以下のサブパッケージからなる.

- **SGPACK:** ユーザーインターフェイス.
- **SCPACK:** 3 次元ユーザーインターフェイス.
- **SLPACK:** 複数の図の割り付けのための基本ルーチン.
- **SZPACK:** 線分やトーンなどの図形を描画する基本ルーチン.
- **STPACK:** 座標変換の基本関数.
- **SWPACK:** 実際の描画デバイスに出力する下請けルーチン.

通常, ユーザーが使うのは主として SGPACK のルーチン群で, その他のパッケージを使わなければならない場合は少ない. しかし, 実際の描画動作は SZPACK がおこなうので, 基本的には SZPACK だけでプログラムを書くことも可能である. SGPACK は各種パラメタ設定などのオーバーヘッドにかなりの時間が使われている場合があるので, 高速化を計るためには SZPACK を使えばよい.

GRPH1 の内部変数管理ルーチンは SGPACK の SGpGET/SGpSET と SWPACK の SWpGET/SWpSET だけである. ここで p は, 変数のタイプ別に R:実数; I:整数; L:論理変数; C: 文字変数 を表わす. このマニュアルでは以後, タイプ別のエントリ名を総称して SGpSET などと書く. SWpGET/SWpSET は主に機種依存パラメタを管理しているので, **GRPH1** の主要な内部変数は全て SGpGET/SGpSET によって管理されている.

なお, これらのパッケージのうち, 機種依存部分を含むのは SWPACK だけである.

1.2 デバイスとフレーム

図形を描くためには, まずディスプレイやプリンタなど デバイス の準備をする必要がある. その準備をする操作を 「オープン」という. そして, 最後に描画を終了する時の操作を 「クローズ」という. すべての図形は, デバイスをオープンしてからクローズするまでの間に描かれなければならない. 地球流体電脳ライブラリでは, 一度に 1 つのデバイスをオープンすることができる. 同時に 2 つ以上のデバイスをオープンすることはできないが, オープン時にいくつかのデバイスの中の 1 つを選択できる.

デバイスに出力される図形が複数ページにわたることがあるが, 地球流体電脳ライブラリでは, ページという言葉の代わりにフレームという言葉を使う. これは, デバイスによってはページという概念のないもの (例えば, 巻物のような長い紙に出力するようなデバイス) があったり, 後述のように物理的な 1 ページの中に複数のフレームを設定することもあるので, 物理的なページと区別するためである.

グラフィックを使うプログラムの形態は次のようになる.

```
1         CALL SGOPN(1)       ! デバイス番号 1 の装置をオープン
2         CALL SGFRM          ! フレームを用意
3          ........
4          ........           ! 1 ページめの描画プログラム
5          ........
6         CALL SGFRM          ! 改ページ
7          ........
8          ........           ! 2 ページめの描画プログラム
9          ........
10        CALL SGFRM
11            .
12            .
13            .
14            .
15            .
16            .
17        CALL SGCLS          ! デバイスのクローズ
```

1.3 座標系と変換

本ライブラリで使用する座標系には 4 つのレベルがある. それぞれのレベルには上位レベルから順に U, V, R, W という記号が付けられ, 「U 座標系」, 「V 座標系」, 「R 座標系」, 「W 座標系」というように呼ばれる (図 1.1 参照). この U, V, R, W という文字は, それぞれ次のような名前に由来するが, これらの名前が長いので, 以後は単に 「U 座標系」などの用語を使うことにする (さらに UC などとも略記する).

U	ユーザ座標系	User coordinate
V	仮想直角座標系	Virtual rectangular coordinate
R	正規直角座標系	normalized Rectangular coordinate
W	装置座標系	Workstation coordinate

レファレンスマニュアルの引数の説明でも, U 座標系の座標値は (UX,UY) というように, 座標系を表す文字をつけて記述してある. (正規直角座標系を N 座標系でなく R 座標系と呼ぶのは, 実変数に関する FORTRAN の規則による.)

最も上位の U 座標系は普通の X-Y 座標から地図投影座標まで多くの座標系を含む. その一つ下位の V 座標系には 2 次元, 3 次元のそれぞれ 1 つずつの直角座標系が定義されており, 多様な U 系の座標はすべてこの 2 種類の座標系に変換される. この変換のことを 「正規化変換」 と呼ぶ.

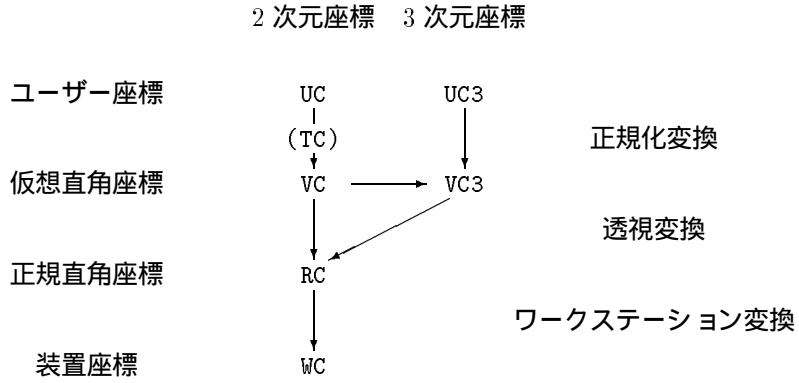


図 1.1: 座標系と変換関数の関係

一方, 最も下位の座標系である W 座標系はデバイス固有の座標系で, デバイスによって異なる. これを統一するために, その一つ上位に R 座標系を定義して, すべてこの R 座標系を経由して W 座標系に変換される. この変換を 『ワークステーション変換」と呼ぶ. R 座標系は 2 次元の直角座標系で, x 軸, y 軸 ともに定義域は $[0,1]$ である.

これらの上位 2 つと下位 2 つの座標系を結ぶ V 座標系から R 座標系への変換は 『透視変換」 と呼ばれる. 透視変換はその名の通り, 透視図法すなわち遠近法による変換で, 基本的に 3 次元の V 座標系から 2 次元の R 座標系への変換である. 2 次元の V 座標系は, 一旦, 3 次元の V 座標系に割り付けられてから透視変換される. この時 2 次元の V 座標系を割り付ける位置は, 3 次元座標の x 軸, y 軸 または z 軸に垂直な任意の平面にとることができる. これにより, 2 次元的に描いた 1 枚の図を斜めから眺めたような描写が可能になる.

初期状態では, 2 次元 V 座標系からの透視変換は恒等変換となっているため, 2 次元の座標系を普通に使う限り, 透視変換を意識する必要はないが, あとで透視変換を使いたくなることも考えて, V 座標系と R 座標系は使い分けておいた方が良い. すなわち, 座標軸などの 『図に付随した情報」は, 透視図法で図とともに変換すべき情報なので V 座標系で描画し, SLPACK の描くページ数やパラメタの数値などのように 『紙に付随した情報」は R 座標系で描画すべきである.

なお, 地図座標系の正規化変換は, 座標の 『回転」と 『投影」の 2 段階で行なわれ, その中間の座標 (U 座標系を回転した座標) を T 座標 (Terrestrial coordinate) という. T 座標は x 軸の定義域が $[-180,180]$, y 軸の定義域が $[-90,90]$ の座標系で, この座標系から各地図投影の定義式に従って変換される. 任意の経線を中央経線としたり, 横軸法や斜軸法の投影をするためには投影の前に回転操作が必要となる.

T 座標系は正規化変換の際の一時的な作業座標系であるので, これを意識しなければならないことは少いが, 地図座標のクリッピングだけは U 座標系ではなく T 座標系で行なわれるので, この座標系を意識する必要がある.

1.4 2 次元正規化変換

1.4.1 座標系の種類

2 次元の U 座標系には次のような種類があり, ユーザーはその中から関数番号で指定する (略号, または名称から関数番号を返す関数も用意されている). 各座標系は, その種類によって正規化変換に必要なパラメータが異なる.

- 直角直線座標系

番号	略号	名 称	
1	U-U	UNI-UNI	直角一様座標
2	U-L	UNI-LOG	片対数座標
3	L-U	LOG-UNI	片対数座標
4	L-L	LOG-LOG	対数座標

- 直交曲線座標系

番号	略号	名 称	
5	POL	POLAR	極座標
6	BPL	BIPOLAR	双曲線座標
7	ELP	ELLIPTIC	楕円座標 (未実装)

- 地図投影座標系

番号	略号	名 称	
10	CYL	CYLINDRICAL	正距円筒図法
11	MER	MERCATOR	メルカトール図法
12	MWD	MOLLWEIDE	モルワイデ図法
13	HMR	HAMMER	ハンメル図法
14	EK6	ECKERT 6	エッケルト第 6 図法
15	KTD	KITADA	北田楕円図法
20	CON	CONICAL	正距円錐図法
21	COA	CONICAL EQ.AREA	ランベルト正積円錐図法
22	COC	CONICAL CONFORMAL	ランベルト正角円錐図法
23	BON	BONNE	ボンヌ図法
30	OTG	ORTHOGRAPHIC	正射図法
31	PST	POLAR STEREO	ポーラーステレオ図法
32	AZM	AZIMUTHAL	正距方位図法
33	AZA	AZIMUTHAL EQ.AREA	ランベルト正積方位図法

- ユーザー定義座標系

番号	略号	名 称	
99	USR	USER	ユーザー定義関数

ユーザー定義座標系はユーザーが必要な変換関数を用意するものである. 詳しくは 6節 (STPACK) を参照のこと.

1.4.2 地図投影法

地球流体電脳ライブラリでは地図投座標も普通の U 座標系の一つとして扱われる. 地図投影法には実にさまざまな種類があり, すべてを解説するためには 1 冊の本が必要であるが, ここでは地球流体電脳ライブラリを利用するために知っておくべき基本的事柄を解説する. なお, 地図学の詳しい解説は 野村 正七 著 『地図投影法』 などを参考にされたい (ただし, この本は絶版).

地図投影法は基本的に 『球面を何らかの方法で 2 次元平面に投影する』方法である. 実際の地球は完全な球面ではなく回転楕円体に近いので, 厳密な地図投影を行なうためには回転楕円体の表面を平面に投影しなければならない. しかし, 本ライブラリでは回転楕円体からの投影ではなく, 球面からの投影のみをサポートする. なお, 投影された図形は投影法によっては 3 次元的な雰囲気を持つが, 地図投影座標はあくまでも 2 次元座標であることに注意されたい.

地図投影の多くの図法では幾何学的な方法で球面から平面に投影される. その際の平面の形状によって次のようにいくつかの図法に分類される.

円筒図法　地球儀をこれに接する円筒の中に位置付け, 何らかの方法で経緯線を円筒に投影して, 展開する図法.
円錐図法　地球儀をこれに接する直円錐の中に位置付け, 何らかの方法で経緯線を円錐に投影して, 展開する図法.
方位図法　地球儀上の一点でこれに接する平面を設定し, 何らかの方法で経緯線をこの平面に投影する図法.
便宜図法　上記以外の図法.

これらの円筒や円錐の軸, または方位図法における接点からの垂線が地球の自転軸と一致するものを正軸法, 直交するものを横軸法, 斜めに交わるものを斜軸法と呼ぶ. 一般に, 円筒図法は全球の表示に向いており, 円錐図法は中緯度の表示に向いている.

地図投影では球面を平面に投影するため, どうしてもひずみが生じてしまう. そこで, 多くの図法では 面積 または 角度 のどちらかが保存されるように工夫されている. 原理的に, この二つを同時に保存するような投影は不可能である. この様な保存性に関する性質によって次のように分類される.

正積図法　地図上のどこでも面積関係が正しく表現される図法.
正角図法　地図上のどこでも局所的な角度が正しく表現される図法.
正距図法　経線, 緯線または方位線上で, 長さが正しく表現される図法.

本ライブラリでは以下のような地図投影がサポートされる.

- 円筒図法

番号	図　　法	正積	正角	正距
10	正距円筒図法	×	×	経線
11	メルカトール図法	×	○	×
12	モルワイデ図法	○	×	×
13	ハンメル図法	○	×	×
14	エッケルト第 6 図法	○	×	×
15	北田楕円図法	○	×	×

- 円錐図法

番号	図 法	正積	正角	正距
20	正距円錐図法	×	×	経線
21	ランベルト正積円錐図法	○	×	×
22	ランベルト正角円錐図法	×	○	×
23	ボンヌ図法	○	×	×

- 方位図法

番号	図 法	正積	正角	正距
30	正射図法	×	×	×
31	ポーラーステレオ図法	×	○	×
32	正距方位図法	×	×	方位線
33	ランベルト正積方位図法	○	×	×

ハンメル図法は地図学上, ランベルト正積方位図法の変種として方位図法に分類されるが, その形状や用途はむしろ円筒図法に近いので, ここでは便宜的に円筒図法に分類する. これらの図法すべてについて, 正軸法, 横軸方, 斜軸法が可能である.

なお, 普通の地図投影法には含まれないが, 正射図法の変形として人工衛星から眺めたような投影法 (Satellite View) もサポートされている. 普通の正射図法は地球儀を無限遠から眺めたような投影になっているが, これを有限の位置から眺めたような投影に変形したものである.

1.4.3 正規変換パラメタ

上記の U 座標を 2 次元 V 座標に変換する正規化変換に必要なパラメータには以下のものがある. これらは GRPH1 の内部変数を管理するルーチン SG*p*GET/SG*p*SET により一つずつ設定/参照できるが, 同種のパラメータをまとめて設定するユーティリティーも用意されている.

ビューポート (VXMIN, VXMAX, VYMIN, VYMAX)
: ビューポートを指定するパラメタは, すべての変換に共通で必須である. ビューポートとは, 通常座標軸が描かれる矩形の枠である. クリッピングをするように指定すると, この枠からはみ出した部分は描画されない. ビューポートを指定するには, その左下と右上の角の VC における座標値を指定する.

  これらのパラメータは SGSVPT/SGQVPT によって設定/参照できる.

ウインドウ (UXMIN, UXMAX, UYMIN, UYMAX)
: ウインドウは直角直線座標系を設定するのに必要なパラメタである. これは, ビューポートに対応する座標値を UC の値で指定する.

  地図投影座標系の場合には, GRPH1 の範囲ではこの情報を必要としないが, GRPH2 のパッケージが「注目している緯度経度範囲」という意味で参照する場合がある. この場合には, 一般にビューポートとウインドウは一致しない.

  これらのパラメータは SGSWND/SGQWND によって設定/参照できる.

相似変換パラメタ (SIMFAC, VXOFF, VYOFF)
: 直交曲線座標系および地図投影変換の場合の, 相似変換 (拡大縮小と原点移動) のパラメタ. 直交曲線

座標系の関数は MATH1/CTRLIB, 地図投影座標系の関数は MATH1/MAPLIB の関数によって定義されている. GRPH1 では, これらの関数が返す値に, SIMFAC を掛けて VC の値に変換し, 原点をビューポートの中心から（VXOFF, VYOFF）平行移動した位置に設定する.

これらのパラメータは SGSSIM/SGQSIM によって設定/参照できる.

地図座標回転パラメタ（ PLX, PLY, PLROT ）

地図投影変換は常に経度範囲 $[-180°, 180°]$ 緯度範囲 $[-90°, 90°]$ に対して行なわれるので, 任意の視点からの投影を行なうためには, 地図投影を行なう前に緯度経度座標を回転させておく必要がある. (PLX, PLY) は投影座標 (TC) の極をおく経度・緯度 (UC) を指定し, PLROT は投影座標の極の回りの回転角を指定する.

これらのパラメタと, 一般に 3 次元の回転角を指定するのに使われる「オイラーの角 (θ, φ, ψ)」との関係は, $\theta = \pi/2 - \mathrm{PLY}$, $\varphi = \mathrm{PLX}$, $\psi = \mathrm{PLROT}$ である. (オイラーの角の意味については, 数学辞典などを参照のこと.)

これらのパラメータは SGSMPL/SGQMPL によって設定/参照できる.

実際の操作としては, UC と TC (投影座標) が一致している状態を初期状態として,

1. 北極を中心として TC を φ 回転させる.
   (経度 PLX の経線を中央経線とする)
2. TC の北極を中央経線の方向に θ 回転させる.
   (経度・緯度（PLX, PLY）の地点を, TC の北極とする)
3. TC の北極の回りに, ψ 回転させる.
   (経度 PLX の経線と, TC の中央経線のなす角を PLROT とする.)

となる.

TC から VC への変換は上記の相似変換パラメタによって決められる. その際地図投影のタイプによって VC 原点に投影される TC の値が異なる.

図　　法	原点に投影される TC
円筒図法	$(0, 0)$
方位図法	$(90, 0)$
円錐図法	円錐の頂点

原点に投影されるこれら TC の値は, オイラーの角を $(0, 0, 0)$ としたとき, すなわち $(\mathrm{PLX}, \mathrm{PLY}, \mathrm{PLROT}) = (0, 90, 0)$ とした場合に対応する.

通常の円筒図法や円錐図法 (正軸法) で, 経度λの経線を中央経線とするには, $(\mathrm{PLX}, \mathrm{PLY}, \mathrm{PLROT}) = (\lambda, 90., 0)$ とする.

方位図法において, 経度・緯度 (λ, ϕ) を中心とした地図を描くには, $(\mathrm{PLX}, \mathrm{PLY}) = (\lambda, \phi)$ とし, 必要に応じて PLROT を指定する $\mathrm{PLROT} = 0$ の時, 必ず UC の南極が原点の下になる.

横軸法の円筒図法などでは, (PLX,PLY) として赤道上の点を指定する. $\mathrm{PLROT} = 0$ の時, UC の南極が中央経線上に投影される. 例えば, 北極海, 大西洋, 南極環海を一つながりの海として横軸法で表すには, (PLX, PLY, PLROT) = (60., 0., -90.) とする.

標準緯度 (STLAT1, STLAT2)

円錐図法の投影では上記のパラメタ以外に, 標準緯度が必要である. ランベルト正角円錐図法は標準緯線が二つあるので, STLAT2 も指定しなければならないが, その他の円錐図法は STLAT1 の値のみを使う.

これらのパラメータは SGRGET/SGRSET によって設定/参照できる.

衛星軌道半径 (RSAT)

正射図法は本来, 無限遠から地球を眺めたような投影法であるが, オプションとして有限の位置にいる衛星から眺めたような投影 (Satellite View) が可能である. 正射図法でこのオプションを選択するには, RSAT を 1 以上の値に設定する. RSAT は, 地球半径を 1 とした時の 「衛星の軌道半径」である.

これらのパラメータは SGRGET/SGRSET によって設定/参照できる.

1.4.4 U 座標系の設定

U 座標系で描画するためには, SGFRM を呼んだ後でかつ描画をはじめる前に変換関数を決めるパラメータを指定しておかなければならない. 指定すべきパラメータは座標系によって異なるが, 主要なパラメタはすべて SGpGET/SGpSET によって管理されている. ただし, SGpGET/SGpSET は単に掲示版の役目をしているだけなので, 設定したパラメタの値は, 変換関数を確定するルーチン SGSTRF を呼ぶことで有効になる. 変換関数のパラメタを 1 つ 1 つ設定するのは面倒なので, まとめてパラメタを設定するルーチンが用意されており, 普通はそちらを使ってパラメタの設定を行なう.

具体的に変換パラメタを設定するには SGFRM のあとで次のようにコーディングする.

直角座標系

```
1          CALL SGSVPT(VXMIN, VXMAX, VYMIN, VYMAX)
2          CALL SGSWND(UXMIN, UXMAX, UYMIN, UYMAX)
3          CALL SGSTRN(ITR)
4          CALL SGSTRF
```

直交座標系

```
1          CALL SGSVPT(VXMIN, VXMAX, VYMIN, VYMAX)
2          CALL SGSSIM(SIMFAC, XOFF, YOFF)
3          CALL SGSTRN(ITR)
4          CALL SGSTRF
```

地図座標系

```
1          CALL SGSVPT(VXMIN, VXMAX, VYMIN, VYMAX)
2          CALL SGSSIM(SIMFAC, XOFF, YOFF)
3          CALL SGSMPL(PLX, PLY, PLROT)
4          CALL SGSTRN(ITR)
5          CALL SGSTRF
```

円錐投影の場合は, これ以外に SGRSET によって標準緯線の緯度 (STLAT1, STLAT2) を与えておく必要がある.

これらのパラメタは 変換関数を確定するルーチン SGSTRF が呼ばれる前であれば, どのような順番で指定しても構わない. 変換関数名または略号を指定して, 変換関数番号を返す関数も用意されている.

1.5 3 次元正規化変換

1.5.1 座標系の種類

3 次元のユーザー座標系 (UC) は次のような種類があり, ユーザーはその中から関数番号によって 1 つを選択する.

番号	
1	直角座標
2	円筒座標
3	球座標

3 次元座標系は 2 次元座標系と異なり対数軸のしていは関数番号ではなく, 対数スイッチ (後述) によって指定する.

1.5.2 3 次元正規変換パラメタ

上記の 3 次元 U 座標を 3 次元 V 座標に変換する正規化変換に必要なパラメータには以下のものがある. これらは GRPH1 の内部変数を管理するルーチン SG*p*GET/SG*p*SET により一つずつ設定/参照できるが, 同種のパラメータをまとめて設定するユーティリティーが SCPACK の中に用意されている.

ビューポート (VXMIN3, VXMAX3, VYMIN3, VYMAX3, VZMIN3, VZMAX3)
: 直角直線座標 (ITR=1) の変換に必要なパラメタで, ウィンドのパラメタと共に指定する. 2 次元のビューポートと異なりクリッピングの範囲とは関係がない.

  これらのパラメータは SCSVPT/SCQVPT によって設定/参照できる.

ウインドウ (UXMIN3, UXMAX3, UYMIN3, UYMAX3, UZMIN3, UZMAX3)
: ITR=1 の時, ビューポートに対応する座標値を UC の値で指定する.

  これらのパラメータは SCSWND/SCQWND によって設定/参照できる.

対数スイッチ (LXLOG3, LYLOG3, LZLOG3)
: ITR=1 の時, それぞれの軸を対数にしたいときに, .TRUE. にする.

  これらのパラメータは SCSLOG/SCQLOG によって設定/参照できる.

相似変換パラメタ (SIMFAC3, VXORG3, VYORG3, VZORG3)
: 円筒座標と球座標の時に指定する相似変換 (拡大縮小と原点移動) のパラメタ. 座標原点を (VXORG3, VYORG3, VZORG3) に設定する.

  これらのパラメータは SCSORG/SCQORG によって設定/参照できる.

1.5.3 座標系の設定

3 次元の U 座標系で描画するためには, 2 次元座標系と同様 SGFRM を呼んだ後でかつ描画をはじめる前に変換関数を決めるパラメータを指定しておかなければならない. これらのパラメタの値は, 変換関数を確定するルーチン SCSTRF を呼ぶことで有効になる. 3 次元のパラメタは 2 次元のパラメタとは独立である.

主要なパラメタはすべて SGpGET/SGpSET によって管理されているが, 変換関数のパラメタを 1 つ 1 つ設定するのは面倒なので, まとめてパラメタを設定するルーチンが用意されており, 普通はそちらを使ってパラメタの設定を行なう.

具体的に変換パラメタを設定するには SGFRM のあとで次のようにコーディングする.

直角座標系

```
1          CALL SCSVPT(VXMIN3, VXMAX3, VYMIN3, VYMAX3, VZMIN3, VZMAX3)
2          CALL SCSWND(UXMIN3, UXMAX3, UYMIN3, UYMAX3, UZMIN3, UZMAX3)
3          CALL SCSTRN(ITR3)
4          CALL SCSTRF
```

円筒座標, 球座標

```
1          CALL SCSORG(SFAC3, VXORG3, VYORG3, VZORG3)
2          CALL SCSTRN(ITR3)
3          CALL SCSTRF
```

1.6 透視変換

透視変換とは, 任意の視点から 3 次元空間を眺めて遠近感を出す変換である. 電能ライブラリにおいては, 3 次元 V 座標系から 2 次元 R 座標系への変換を指す.

一般に, この変換では 3 次元空間上の平行線は投影された 2 次元平面上では平行にならない. 視点を無限遠に持っていけば, 平行線は平行線に投影されるが, そのように投影された図形はかえって不自然に感じられる.

透視変換を行なうには, 「視点」と「焦点中心」を設定しなければならない. 視点とは読んで字の通り 3 次元空間を眺める位置であり, カメラのレンズの位置であると思えば良い. これに対して, 焦点中心はカメラのフィルム上の焦点ではなく, 注目している点, すなわち視点から「見つめる点」である. この視点と焦点中心を結ぶ線が「視線」である.

透視変換によって 3 次元図形の各点は, その点と視点を結ぶ線が「投影面」と交わる点に投影される. 地球流体電脳ライブラリでは, 投影面は焦点中心を通り, 視線に垂直な面である.

2 次元座標系の場合, 2 次元の V 座標系を 3 次元の V 座標系の 1 平面に割り付けることで, 透視変換が可能になる. この際に割り付けることができる平面は, 3 次元 X, Y, Z 座標軸のいずれかに垂直な平面であり, 斜めの面に割り付けることはできない. (斜めに見るには視点を動かせば良い)

1.6.1 変換パラメタ

視点 (XEYE3, YEYE3, ZEYE3)
: 3 次元空間上での視点. この位置にカメラを構えていると思えば良い.

焦点中心 (XOBJ3, YOBJ3, ZOBJ3)
: 3 次元空間内で視線を向ける目標点. カメラを向ける被写体の位置と思えば良い. なお, 焦点中心を通り, 視線 (視点と焦点中心を結ぶ線) に垂直な面が投影面となる. すなわち, 3 次元空間上の点 (X,Y,Z) は, その点と視点を通る線が投影面と交わる点に投影される.

画角 (ANGLE3)
: 視点からこの角度で見える投影面上の長さが, 正規直角座標系の単位長さとなる.

傾き (TILT3)
: Z 軸の傾き. カメラの光軸回りの傾きに相当する. この角度が 0 の時, 3 次元空間の Z 軸は正規直角座標の Y 軸に並行になる. 視線が Z 軸と平行な場合には, Y 軸が基準になる.

オフセット (XOFF3, YOFF3)
: 正規直角座標系に投影された焦点中心の位置を, 中心からの位置で指定する. 通常この値は (0,0) で良い.

2 次元平面位置 (IXC3, IYC3, SEC3)
: 2 次元平面を 3 次元空間のどこに位置付けるか指定する. IXC3, IYC3 はそれぞれ 2 次元平面の X 座標, Y 座標に対応する 3 次元座標を (1,2,3) の数字で指定するもので, 数字はそれぞれ X 座標, Y 座標, Z 座標に対応する. これらの数値に負の値を指定すると, 正負を逆に割り当てる. SEC3 は残りの座標の座標値である.

1.6.2 透視変換の設定

透視変換のパラメタも SGFRM を呼んだ後でかつ描画をはじめる前に変換関数を決めるパラメータを指定しておく. これらのパラメタの値は, 変換関数を確定するルーチン SCSPRJ を呼ぶことで有効になる. 透視変換のパラメタは, 2 次元および 3 次元正規化変換と独立である.

主要なパラメタはすべて SG*p*GET/SG*p*SET によって管理されているが, まとめてパラメタを設定するルーチンが SCPACK に用意されており, 普通はそちらを使ってパラメタの設定を行なう.

具体的に変換パラメタを設定するには SGFRM のあとで次のようにコーディングする.

```
1           CALL SCSEYE(XEYE3, YEYE3, ZEYE3)
2           CALL SCSOBJ(XOBJ3, YOBJ3, ZOBJ3)
3           CALL SCSPRJ
```

そのほかのパラメタは必要に応じて SG*p*SET で指定する.
なお, 2 次元平面を 3 次元平面に割り付けるには, SCSPRJ の前に,

```
1           CALL SCSPLN( IXC3, IYC3, SEC3)
```

を呼ぶ. これにより, 2 次元平面上の描画ルーチンを 3 次元空間内の平面上で使うことができるようになる.

1.7 出力プリミティブ

図形を構成する基本要素を出力プリミティブという. GRPH1 の出力プリミティブには次のものがある.

- ポリライン (polyline) : 折れ線.
- ポリマーカー (polymarker) : マーカー列.

- テキスト (text) : 文字列.
- トーン (tone) : 多角形のぬりつぶし.

このほか, 補助的に次のプリミティブがある.

- ライン (line) : 線分.
- アロー (arrow) : 矢印.

これら 6 つのプリミティブのうち, トーンプリミティブ以外の 5 つはすべて線分で構成される. 線分にはラインインデクスとラインタイプ呼ばれる属性がある. また, トーンには, トーンパターン番号という属性がある. これらの属性は, 以下のように定義されている.

GRPH1 で扱う座標系には曲線座標系も含まれるので, 線分やトーンの境界を指定する座標点の間を直線でつなぐと不都合が起こる場合がある. そこで, 座標点の間をいくつかに分割して, 補間する機能がある. さらに, すべてのプリミティブはある境界でのクリッピングの対象となる.

1.7.1 ラインインデクス

GRPH1 で描かれる線分の太さと色はラインインデクスと呼ばれる 3 桁の整数 (nnm) で指定される. 線の太さと色のうちどちらか一方しか変えられないようなデバイスに出力する場合でも, ラインインデクスの異なる 2 本の線が識別できるようにするため, ラインインデクスは次のような規則にしたがって太さと色に対応づけられる.

線の太さと色が両方変えられるようなシステムでは, 上位 2 桁 (nn=0- 99) が色番号, 下位 1 桁 (m=0- 9) が線の太さを表す. 色番号は, 1 から 5 までは標準的に

1: 白または黒 (フォアグラウンド)
2: 赤
3: 緑
4: 青
5: 黄

と決められているが, それより大きな番号に関しては colormap ファイルの定義による. また m: 1(細)→9(太) となっている. 線の太さだけが変えられるような出力装置では, m=0 のときに限って nn を m として読みかえる. また, 線の色だけがかえられるような出力装置では, nn=0 のときに限って m を nn として読みかえる. したがって線の太さと色を明示したいとき以外は, 1 桁のラインインデクスを指定しておけば, とりあえず装置に固有な方法によって線分は識別可能となる.

1.7.2 ラインタイプ

ラインタイプとは, 実線, 破線などの線種である. 1 から 4 までの番号にはあらかじめ以下のタイプが決められている.

1: 実線
2: 破線
3: 点線
4: 1 点鎖線

その他の 0 以外の整数は下位 N ビット (N は内部変数'NBITS' で決まる値. 初期値は 16; 2.4.4節参照) のビットパターンを用いて線種が設定される. たとえば N=16 で ITYPE = Z'0000F0F0' (16 進定数) のとき, '4bits ON 4bits OFF 4bits ON 4bits OFF' のような破線が設定される. 1 ビット当たりの長さは内部変数 'BITLEN' が決める. ラインタイプの作画例は 8.1.1節参照.

1.7.3 トーン番号

トーンパターン番号には色の情報と, パターンの情報が両方含まれており, 下位 3 桁がパターン番号, それより上位の桁は色を指定する. 色を指定しない (3 桁の番号だけ指定) 場合は, フォアグラウンド (色番号 1) と解釈される.

パターン番号の最上位桁は, パターンの種類をあらわす.

0: ドット
1: 横線
2: 斜線 (右上がり)
3: 縦線
4: 斜線 (左上がり)
5: 格子 (縦横)
6: 格子 (斜め)

2 桁めは, ドットの大きさや斜線の太さを 0 から 5 の間で指定する. 値が大きくなるにつれてドットは大きくなり斜線は太くなる. 最下位桁はドットや斜線の密度を 0 から 5 の間で指定する. 値が大きくなるにつれてドットや斜線の密度があがる. ただし 0 のときは何も描かれない.

パターン番号として 999 を指定するとべたぬりとなる.

カラーの端末やカラーのプリンターでは, 色を指定したべた塗りによる塗りわけがしばしばおこなわれる. このプログラムをカラーがサポートされない環境で実行しても, それなりな見わけがつくようにドットによる塗りわけとして表現するようなオプションも用意されている.

これらの組合せ以外の整数値はパターンが定義されていない. トーンパターンテーブルについては 8.1.4-8.1.10節を参照のこと.

1.7.4 補間とクリッピング

補間機能には次の 2 つがある.

線形補間: ユーザ座標系で線形的に補間する. すべての座標系で指定可能であるが, 通常の直角一様座標 (ITR=1) で指定しても意味はない. 曲線座標系の座標軸や, 地図座標の緯度線を描く時などに使われる.

大円補間: 地図投影座標 (TC) で, 始点と終点を通る大円上で補間する. これは地図投影関数のみ有効.

地図投影関数の場合には, TC において特異点は必ず極になっているので, 大円補間を指定することにより, 特異点付近での線分がより適切に描けるようになる.

この機能に関する SG*p*GET/SG*p*SET のパラメタは以下の通り.

LLNINT	(L)	線形補間を行なうかどうか指定するフラグ (省略値.FALSE.).
LGCINT	(L)	大円補間を行なうかどうか指定するフラグ (省略値.TRUE.).
RDX, RDY	(R)	補間間隔. 1 セグメントの長さがこれらの値を越えないように補間される (省略値 (5., 5.)).

クリッピングは次の場所で行なわれる.

- 地図の境界 地図投影座標 (TC)
- ビューポート 仮想直角座標 (VC)
- デバイス境界 正規直角座標 (RC)

それぞれのクリッピングは, その座標系の座標軸に平行な線で囲まれる矩形領域に対して行なわれる. したがって, 地図投影の際に極を移動する回転を行なうと, TC におけるクリッピング境界はは経緯線に一致しなくなる.

この機能に関する SGpGET/SGpSET のパラメタは以下の通り.

LCLIP	(L)	ビューポートでのクリッピングを行なうかどうか指定するフラグ (省略値.FALSE.).
VXMIN, VXMAX, VYMIN, VYMAX	(R)	ビューポート. 通常 SGSVPT により設定される.
TXMIN, TXMAX, TYMIN, TYMAX	(R)	TC におけるクリッピング境界 (省略値は -180.0, +180.0, -90.0, +90.0).
IRMOD	(I)	トーンの境界線をクリッピングした時に, 右回りに接続するか, 左回りに接続するか指定するパラメタ. (1.7.5 参照) (省略値 0).

ビューポートにおけるクリッピングは 'LCLIP' によって ON/OFF できるが, 他のクリッピングは常におこなうようになっている. 正射図法で TC におけるクリッピング境界を省略値のまま使うと, 裏側の線分なども描くことになる. 表側だけを描くためには'TYMIN' を 0 にする.

1.7.5 トーンのクリッピングに関する注意

地球流体電脳ライブラリのトーンは通常 (クリッピングされなければ) 指定された境界線が交わっていても, その図形の内側が塗りつぶされる. 正確には, ソフトフィルは Even Odd Rule にしたがって塗りつぶしが行われ, ハードフィルは機種依存であるが, できるだけソフトフィルに近い規則で塗りつぶされる様になっている.

旧バージョンではハードフィルはクリッピングの対象になっていなかったが, DCL ver.5 からハードフィルもソフト的にクリッピングできるようになった. すなわち, トーンのクリッピングはハードウェアが行うのではなく, 指定された領域とクリッピング範囲が重なる部分を指定する境界点を生成して, そのデータをトーンルーチンに渡すようになっている. このクリップされた境界線を求めるアルゴリズムは, 境界線の向きが決まっていることを前提としている. ここで, 境界線の向きとは境界線を指定する点列を, 時計回りに指定するか, 反時計回りに指定するかということを言う. このアルゴリズムは旧来のトーンの塗りつぶし規則と整合性が悪いが, 地図上 (T 座標系) では境界線の向きがわからないと内側と外側を判断することができない.

具体的にはトーンをクリッピングする手順は次のようになる.

1. 境界線をクリップして, いくつかの線分に分割する.
2. 切断された線分を接続しなおし, 閉じた境界を生成する. IRMODE=0 の時, 反時計回り; IRMODE=1 の時, 時計回りに接続する.
3. 生成された境界線に対して旧来の規則にしたがって塗りつぶしを行う.

クリッピングされない領域に関しては, 旧来の規則がそのまま適用され, IRMODE は何の意味ももたない. つまり, IRMODE を適当に指定することで, 境界線の外側を塗りつぶすことができるわけではない.

したがって, 境界線が交わるような図形を塗りつぶす時に, クリッピングを行うのは危険である. クリッピングを行う時は, 安全のために

境界線の向きを決めて, 小さな単純な図形にする

様にしていただきたい.

なお, 図形の外側を塗りつぶしたい時には, その図形が確実にクリッピングされるように領域を 2 分割し, クリッピング領域 (ビューポート) を設定しなおして, 2 度トーンルーチンを呼ぶ.

1.8 レイアウト

R 座標系は x 軸, y 軸ともに 0 から 1 の値をとり, 描画範囲は 1:1 の縦横比を持つ. これに対して, 実際のデバイスの縦横比はいろいろである. 描画範囲とデバイスの縦横比が一致しない時には, 通常 SGFRM が呼ばれた段階で, デバイスの中央に描画範囲が最大内接するようにワークステーション変換が設定される.

しかし, デバイスの描画領域をいっぱいに使いたかったり, 逆に, 図の回りに適当なマージンをとりたい場合もある. そのような時には, GRPH1/SLPACK を使う. SLPACK は基本的にワークステーション変換を設定するもので, 描画範囲の縦横比を変える以外に, マージンにタイトルなどの文字列を書いたり 1 枚の紙に複数のフレームを並べたりする機能がある. この様な機能をレイアウト機能と呼ぶ.

このレイアウト機能を使うには, デバイスをオープンした後で SGFRM を呼ぶ前に, 適切な設定を行なう. 例えば,

```
1           CALL SGOPN(IWS)
2           CALL SGDIV('S', 4, 3)  ! 1ページを 4x3 に分割する.
3           CALL SGFRM
4           ..........
5
6           CALL SGFRM
7           ..........
8
9           CALL SGCLS
```

とすることで, 1 ページに 12 枚のフレームを設定することができる.

第 2 章

SGPACK : ユーザーインターフェイス

2.1 概要

SGPACK は GRPH1 の他のパッケージを統合し, ユーザーに使いやすいインターフェイスを提供するものである. SGPACK には, デバイスやページの制御などを行なうコントロールルーチン, 座標変換の設定ルーチン, 図形の基本要素 (出力プリミティブ) を作画するルーチンなどがある.

出力プリミティブとしては次のものがある.

- ポリライン (polyline) : 折れ線.
- ポリマーカー (polymarker) : マーカー列.
- テキスト (text) : 文字列.
- トーン (tone) : 多角形のぬりつぶし.

このほか, 補助的に次のプリミティブがある.

- ライン (line) : 線分.
- アロー (arrow) : 矢印.

これら 6 つのプリミティブのうち, トーンプリミティブ以外の 5 つはすべて線分で構成される. トーンプリミティブは出力装置の能力に応じて, hard fill または soft fill を切替えて多角形のぬりつぶしをおこなう; soft fill は斜線を引くことによって実現される. また, GRPH1 では各プリミティブをユーザー座標系 (U 座標系), 仮想直角座標 (V 座標系), および正規直角座標 (R 座標系) で定義できる

6 つの出力プリミティブには次のような属性がある.

1. ポリラインプリミティブについては,
   - 折れ線の線種.
   - 折れ線のラインインデクス.
2. ポリマーカープリミティブについては,
   - マーカーの種類.

- マーカーを描く線のラインインデクス.
- マーカーの大きさ.

3. テキストプリミティブについては,
   - 文字の高さ.
   - 文字列の回転角.
   - 文字列のセンタリングオプション.
   - 文字を描く線のラインインデクス.
4. トーンプリミティブについては,
   - トーンパターン.
5. ラインサブプリミティブについては,
   - 線分のラインインデクス.
6. アローサブプリミティブについては,
   - 矢印の軸部分の線種.
   - 矢印の軸部分のラインインデクス.

これらの属性は個別に指定することもできるし, すべて同時に指定することもできる (2.6, 2.7, 2.8, 2.9, 2.10 節参照).

各プリミティブを定義するサブルーチン名の命名規約は以下のようになっている.

1. 先頭の 2 文字はすべて SG で始まる.
2. 属性を設定/参照する場合, 3 文字目は S (set)/ Q (query) である.
3. 各プリミティブの名前がそれに続く. それぞれ,
   - PL : ポリラインプリミティブ.
   - PM : ポリマーカープリミティブ.
   - TX : テキストプリミティブ.
   - TN : トーンプリミティブ.
   - LN : ラインサブプリミティブ.
   - LA : アローサブプリミティブ.
4. プリミティブを定義 (出力) するルーチンについて, 属性を同時に指定する場合は Z が続く.
5. さらにプリミティブをどこに定義するかによって, U (U 座標系で定義する), V (V 座標系で定義する) または R (R 座標系で定義する) が続く.
6. プリミティブの属性を定義/参照する場合, 6 文字目はその属性を示す.
   - T : ラインタイプ, マーカータイプ.

- I : ラインインデクス.
- S : マーカーサイズ, 文字の高さ.
- R : 文字列の回転角.
- C : 文字列のセンタリングオプション.
- P : トーンパターン.

2.2 サブルーチンのリスト

コントロール

SGOPN	出力装置のオープン.
SGFRM	フレームの設定.
SGCLS	出力装置のクローズ.
SGpGET(CP,IPARA)	内部変数の参照.
SGpSET(CP,IPARA)	内部変数の設定.
SGpSTX(CP,IPARA)	内部変数の設定 (実行時オプションによる変更を許す).

正規化変換

SGSVPT(VXMIN,VXMAX,VYMIN,VYMAX)	ビューポートの設定.
SGSWND(UXMIN,UXMAX,UYMIN,UYMAX)	ウインドウの設定.
SGSSIM(SIMFAC,VXOFF,VYOFF)	相似変換の設定.
SGSMPL(PLX,PLY,PLROT)	地図投影の極の設定.
SGSTRN(ITR)	変換関数の番号による設定.
SGSTRF	変換関数の確定.

この他に, パラメタ参照ルーチンとして SGQVPT, SGQWND, SGQSIM, SGQMPL, SGQTRN がある.

ポリライン

SGPLZU(N,UPX,UPY,ITYPE,INDEX)	U 座標系で折れ線を描く.
SGPLZV(N,VPX,VPY,ITYPE,INDEX)	V 座標系で折れ線を描く.
SGPLZR(N,RPX,RPY,ITYPE,INDEX)	R 座標系で折れ線を描く.
SGPLU(N,UPX,UPY)	U 座標系で折れ線を描く.
SGPLV(N,VPX,VPY)	V 座標系で折れ線を描く.
SGPLR(N,RPX,RPY)	R 座標系で折れ線を描く.
SGSPLT(ITYPE)	ラインタイプの設定.
SGSPLI(INDEX)	ラインインデクスの設定.
SGSPLC(CHAR)	ラベルの文字列設定.
SGSPLS(RSIZE)	ラベルの文字高設定.
SGNPLC	ラベルの最後の文字番号を増やす.

この他に, パラメタ参照ルーチンとして SGQPLT, SGQPLI, SGQPLC, SGQPLS がある.

ポリマーカー

SGPMZU(N,UPX,UPY,ITYPE,INDEX,RSIZE)	U 座標系でマーカー列を描く.
SGPMZV(N,VPX,VPY,ITYPE,INDEX,RSIZE)	V 座標系でマーカー列を描く.
SGPMZR(N,RPX,RPY,ITYPE,INDEX,RSIZE)	R 座標系でマーカー列を描く.
SGPMU(N,UPX,UPY)	U 座標系でマーカー列を描く.
SGPMV(N,VPX,VPY)	V 座標系でマーカー列を描く.
SGPMR(N,RPX,RPY)	R 座標系でマーカー列を描く.
SGSPMT(ITYPE)	マーカータイプの設定.
SGSPMI(INDEX)	マーカーのラインインデクスの設定.
SGSPMS(RSIZE)	マーカーの大きさ設定.

この他に, パラメタ参照ルーチンとして SGQPMT, SGQPMI, SGQPMS がある.

テキスト

SGTXZU(UX,UY,CHARS,RSIZE,IROTA,ICENT,INDEX)	U 座標系で文字列を描く.
SGTXZV(VX,VY,CHARS,RSIZE,IROTA,ICENT,INDEX)	V 座標系で文字列を描く.
SGTXZR(RX,RY,CHARS,RSIZE,IROTA,ICENT,INDEX)	R 座標系で文字列を描く.
SGTXU(UX,UY,CHARS)	U 座標系で文字列を描く.
SGTXV(VX,VY,CHARS)	V 座標系で文字列を描く.
SGTXR(RX,RY,CHARS)	R 座標系で文字列を描く.
SGSTXS(RSIZE)	文字の高さ設定.
SGSTXR(IROTA)	文字列の角度の設定.
SGSTXI(INDEX)	文字列のラインインデクスの設定.
SGSTXC(ICENT)	文字列のセンタリングオプシ ョン設定.

この他に, パラメタ参照ルーチンとして SGQTXS, SGQTXR, SGQTXI, SGQTXC がある.

トーン

SGTNZU(N,UPX,UPY,ITPAT)	U 座標系で多角形領域の塗りつぶし.
SGTNZV(N,VPX,VPY,ITPAT)	V 座標系で多角形領域の塗りつぶし.
SGTNZR(N,RPX,RPY,ITPAT)	R 座標系で多角形領域の塗りつぶし.
SGTNU(N,UPX,UPY)	U 座標系で多角形領域の塗りつぶし.
SGTNV(N,VPX,VPY)	V 座標系で多角形領域の塗りつぶし.
SGTNR(N,RPX,RPY)	R 座標系で多角形領域の塗りつぶし.
SGSTNP(ITPAT)	トーンパターン番号設定.

この他に, パラメタ参照ルーチンとして SGQTNP がある.

2.3 関数のリスト

文字関数

CSGI	文字番号に対応する文字を返す.
ISGC	文字に対応する文字番号を返す.

2.4 サブルーチンの説明： コントロール

GRPH1 において図形出力の初期化処理 ･終了処理 ･作画領域の設定 ･内部変数の管理をおこなうサブルーチン群. 以下, 特に断らない限り, 単に 「内部変数」というのは, SG*p*GET/SG*p*SET(2.4.4節参照) の管理する内部変数のことを指す.

2.4.1 SGOPN

1. 機能

   図形出力装置をオープンする (初期化処理をおこなう).

2. 呼び出し方法

```
CALL SGOPN(IWS)
```

3. パラメーターの説明

   IWS (I) ワークステーション番号を指定する.
   IWS>0 のときは画面を横長に使い図形出力は正立,
   IWS<0 のときは画面を縦長に使い図形出力は 90 度回転して表示される.

4. 備考

   (a) すべての GRPH1 サブルーチンコールに先だって 1 回だけ呼ぶこと.

   (b) 実際の初期化作業は SGINIT がおこなっている. SGINIT は内部変数 'IWS' を見て各装置に対する初期化をおこなう. SGOPN は 内部変数 'IWS' を設定して SGINIT を実行している.

   (c) このルーチンはレイアウトサブルーチンパッケージ (SLPACK) を初期化して, 第 1 レベルのワークステーションビューポートを設定する (第 4章参照). したがって, SLPACK の各ルーチンは SGOPN の後に呼ばなければならない.

   (d) 指定できるワークステーション番号とワークステーション名のリストは SGPWSN によってプリントできる (2.4.5節参照). 標準ライブラリにおいては, 1:X (X サーバー), 2:PS (ポストスクリプト), 3:Tek (テクトロ端末) となっている.

2.4.2 SGFRM

1. 機能

   新しい作画領域を設定する.

2. 呼び出し方法

```
CALL SGFRM
```

3. パラメーターの説明

   なし.

4. 備考

(a) 内部変数'LFULL' が.TRUE. のとき, 作画領域 (SLPACK が設定するワークステーションビューポート) をいっぱいに使うようにワークステーション変換が設定される. このとき R 座標系におけるワークステーションウインドウはワークステーションビューポートの縦・横比に応じて, 短辺が変更される. 'LFULL' が.FALSE. のとき, ワークステーションウインドウは $[0,1] \times [0,1]$ と設定され, ワークステーションビューポートもこの縦・横比を保ちながら作画領域に最大内接するように再設定される. このとき'WORKSTATION VIEWPORT WAS MODIFIED.' というメッセージが出力されるが, 実際の作画に影響はない. しかし, SLPACK を用いてマージン部分にタイトルを書こうとしているときは, タイトルの文字高が意図したようにならないことがあるので, あらかじめ SLRAT (4.3.6節参照) などを用いて縦横比を 1:1 に設定しておくこと. 'LFULL' の初期値は.FALSE. (2.4.4節参照) である.

(b) このルーチンによって正規化変換が初期化され, 恒等変換を設定する. したがって正規化変換は SGFMR を呼んだあと毎回再設定すること.

2.4.3 SGCLS

1. 機能

   図形出力装置をクローズする (終了処理をおこなう).

2. 呼び出し方法

   ```
   CALL SGCLS
   ```

3. パラメーターの説明

   なし.

4. 備考

   (a) すべての GRPH1 サブルーチンコールのあとで 1 回だけ呼ぶこと.

2.4.4 SGpGET/SGpSET(SGpSTX)

1. 機能

   GRPH1 で使用する内部変数を参照/変更する. (SGpSTX は実行時オプションによる変更を許す.) p は R/I/L で, それぞれ実数, 整数, 論理変数用.

2. 呼び出し方法

   ```
   CALL SGpGET(CP,IPARA)
   CALL SGpSET(CP,IPARA)
   CALL SGpSTX(CP,IPARA)
   ```

3. パラメーターの説明

CP	(C*8)	内部変数の名前.
IPARA	(I,R,L)	内部変数の値.

   以下に CP として指定できる名前のリストを記す.

[全般的な制御に関するパラメーター]

'IWS' (I) オープンする装置番号 (初期値は 1). この番号は SGOPN が設定する. GRPH1 の範囲ではこの値を実行時オプションによって変更することはできない (GRPH2 の GRPACK を用いれば変更可能である).

'LFULL' (L) SGFRM (2.4.2節参照) で設定するワークステーションビューポートを作画可能な範囲いっぱいにとるかどうかを指定する. .TRUE. のときいっぱいにとる; .FALSE. のとき最大内接する正方形にとる (初期値は.FALSE.).

[ポリラインプリミティブに関するパラメーター]

'NBITS' (R) ポリラインプリミティブの描く折れ線のラインタイプをビットパターンで設定するとき, 下位何ビットを使うか指定する (初期値は 16).

'BITLEN' (R) ポリラインプリミティブの描く折れ線のラインタイプをビットパターンで設定するとき, 1 ビットが VC 上でどれだけの長さに相当するかを指定する (初期値は 0.003).

'MOVE' (I) SZPACK の下位ルーチンが破線等のパターンで線を引くとき pen-up move の後のパターンのサイクルについて指定する (初期値は 1).

1 : pen-up の間も作図を続けたものと見なす.

0 : 最後に pen-down で移動した状態を保つ.

−1 : pen-up で移動した後, 新しいサイクルを始める.

'LCHAR' (L) ポリラインプリミティブで折れ線を描くとき, ラベルつき折れ線とするかどうかを指定する.TRUE. のときはラベルつき折れ線とする; .FALSE. のときはラベルなし折れ線とする (初期値は.FALSE.). 以下はこのラベルつき線分に関するパラメーターである.

'LROT' (L) ラベルとして付ける文字列を一定の回転角で描くか, あるいは線分に沿って描くかを指定する. .TRUE. のとき一定の回転角で描く; .FALSE. のとき線分に沿って描く (初期値は.FALSE.).

'IROT' (I) ラベルとしてつける文字列を一定の回転角で描くとき ('LROT' が.TRUE. のとき) の回転角を指定する (単位は度; 初期値は 0).

'FWC' (R) 文字列を描くための空白域の幅を全文字幅を単位としてどれだけとるかを指定する係数 (初期値は 1.25).

'CWL' (R) 1 サイクルあたり, 線分部分の長さを文字高を単位として何文字分とるかを指定する係数 (初期値は 30.0).

'FFCT' (R) ラベル付き折れ線を描くにあったて, 線分部分を 1 単位としてどれだけ進んだところから描き始めるかを指定する (初期値は 0.5).

'INDEXC' (I) ラベルとして描く文字列のラインインデクスを指定する (初期値は 0). 値が 0 のとき, 線分部分を描くために現在設定されているラインインデクスと同じ値をとる.

'LBUFF' (L) 折れ線の終りが文字列を描くべき空白部分であるときそこを線分で結ぶためのバッファリングをするかどうかを指定する. .TRUE. のときバッファリングをする; .FALSE. のときバッファリングしない (初期値は.TRUE.).

'NBUFF' (I) バッファリングをするときその長さを指定する. (初期値は 20).
'RBUFF' (R) バッファリングした内容を書き出した後, ラベルつき折れ線のサイクルの位置を指定する. 0 と 1 の間の値を指定すること. (初期値は 0.99).
'LCURV' (L) 折れ線のなす角が大きなところなどではラベルを書かないことを指定する. 正確には, 文字を書くべきスペースが次のパラメータ'RCURV' の係数以下のときラベルを書かない. .TRUE. のときラベルを書かない; .FALSE. のときラベルを書く (初期値は.TRUE.).
'RCURV' (R) 'LCURV' が.TRUE. で, ラベルを書くべきスペースがこの係数以下のときラベルを書かない. (初期値は 1.0).

[ポリマーカープリミティブに関するパラメーター]

'PMFACT' (R) ポリマーカープリミティブが描くマーカーの大きさを調整する係数 (初期値は 2.0). ポリマーカープリミティブはテキストプリミティブを用いてマーカーを描く. このとき, 描くマーカーは標準的な文字高に比べて小さい (フォントテーブル 8.1.2, 8.1.3節参照). この内部変数はマーカーが標準的な文字高を持つようにするため使われる拡大係数である.
'NPMSKIP'(I) ポリマーカープリミティブでマーカーを描くとき, 何個に 1 個の割合で描くかを指定する (初期値は 1; すべてマーカーを描く. SGPMZU がバッファリングする最大個数 50 以下でなければならない).

[テキストプリミティブに関するパラメーター]

'IFONT' (I) フォント番号 (初期値は 1).
'LCNTL' (L) テキストプリミティブで文字を描くとき, 制御文字 ('ISUP', 'ISUB', 'IRST' 参照) を有効とするかどうかを指定する.TRUE. のときは有効とする; .FALSE. のときは有効としない (初期値は.FALSE.).
'SMALL' (R) 上付および下付添え字の標準文字高に対する比率 (初期値は 0.7).
'SHIFT' (R) 上付および下付添え字の中心を標準文字の中心からどれだけはなすかを標準文字高に対する比率で指定する (初期値は 0.3).
'ISUP' (I) 上付添え字のモードの始まりをしめす制御文字の文字番号. 初期値は 124; これは'|' に対応する.
'ISUB' (I) 下付添え字のモードの始まりをしめす制御文字の文字番号. 初期値は 95; これは'_' に対応する.
'IRST' (I) 上付および下付添え字のモードの終わりをしめす制御文字の文字番号. 初期値は 34; これは'"' に対応する.

[トーンプリミティブに関するパラメーター]

'LSOFTF' (L) トーンプリミティブで塗りつぶしをおこなうとき soft fill をおこなうか hard fill をおこなうかを指定する. .TRUE. のときは soft fill をおこなう; .FALSE. のときは hard fill をおこなう (初期値は.FALSE.). ただし, hard fill を指定してもその能力がない場合は soft fill となる.

'LCL2TN' (L) 色を用いたべた塗りによる塗りわけをおこなうとき, 色番号をドットなどのパターン番号として読みかえるかどうかを指定する. .TRUE. のときは読みかえる; .FALSE. のときは読みかえない (初期値は.FALSE.). ただし, 色を用いた塗りわけができないような装置では, .FALSE. と設定されていてもパターン番号による読みかえをおこなおうとする.

'TNBLEN' (R) トーンプリミティブでドットパターンによる soft fill をおこなう場合, フィルエリアサブプリミティブの下位ルーチンを用い点線の集合として表現する. このとき点線のラインタイプはビットパターンで設定されるが, その 1 ビットが RC 上でどれだけの長さに相当するかを指定する (初期値は 0.001).

'IRMODE' (I) トーンプリミティブで指定された閉領域がクリッピング境界にかかる時に, 境界線の向きを指定するパラメタ. 0 のとき反時計回り; 1 のとき時計回り. 1.7.5 節参照のこと. (初期値は 0).

[アローサブプリミティブに関するパラメーター]

'LPROP' (L) 矢じり部分の線分を本体部分の線分の長さに比例させて描くかどうかを指定する. .TRUE. なら比例させて描く;.FALSE. なら内部変数'CONST' で指定してある一定値で, 地図投影のときは内部変数'CONSTM' で指定してある一定値で描く (初期値は.TRUE.).

'LUARW' (L) 地図投影のとき, 矢じり部分の線分も地図にへばりついたように描くかどうかを指定する. .TRUE. なら地図上で描く; .FALSE. なら VC で描く (初期値は.TRUE.).

'AFACT' (R) 'LPROP' が.TRUE.(比例させて描く) のときに用いる比例定数 (初期値は 0.33).

'CONST' (R) 'LPROP' が.FALSE.(一定値で描く) のときに用いる一定値 (単位は VC; 初期値は 0.01).

'CONSTM' (R) 'LPROP' が.FALSE.(一定値で描く) でかつ地図投影のときに用いる一定値 (単位は度; 初期値は 5.0).

'ANGLE' (R) 矢じり部分の線分と本体部分の線分のなす角 (単位は度; 初期値は 20.0).

'LATONE' (L) 矢じり部分を定義する三角形の領域を塗りつぶすかどうかを指定する. .TRUE. なら塗りつぶす; .FALSE. なら塗りつぶさない (初期値は.FALSE.).

'IATONE' (L) 'LATONE' が.TRUE. (矢じり部分を定義する三角形の領域を塗りつぶす) のときのトーンパターン (初期値は 999). 3 桁以下のとき, カラーインデクスは軸の部分について指定されているものを用いる.

[変換関数に関するパラメタ] - これらのパラメタの意味については, 1.4節, 1.6節 を参照のこと.

'ITR'	(I)	変換関数番号 (初期値は 1).
'VXMIN','VXMAX','VYMIN','VYMAX'	(R)	ビューポート (初期値は 0.0, 1.0, 0.0, 1.0).
'UXMIN','UXMAX','UYMIN','UYMAX'	(R)	ウインドウ (初期値は 0.0, 1.0, 0.0, 1.0).
'SIMFAC','VXOFF','VYOFF'	(R)	相似変換パラメタ (初期値は 1.0, 0.0, 0.0).
'PLX','PLY','PLROT'	(R)	地図座標回転パラメタ (初期値は 0.0, 90.0, 0.0).
'STLAT1','STLAT2'	(R)	標準緯度 (初期値は 0.0, 0.0).
'RSAT'	(R)	衛星軌道半径 (初期値は 0.0).
'LDEG'	(L)	単位系の指定. 'LDEG' が .TRUE. なら度を単位とする; .FALSE. ならラジアンを単位とする (初期値は .TRUE.).
'ITR3'	(I)	変換関数番号 (初期値は 1).
'VXMIN3','VXMAX3','VYMIN3','VYMAX3'		
'VZMIN3','VZMAX3'	(R)	3 次元ビューポート (初期値は 0., 1., 0., 1., 0., 1.).
'UXMIN3','UXMAX3','UYMIN3','UYMAX3'		
'UZMIN3','UZMAX3'	(R)	3 次元ウィンドウ (初期値は 0., 1., 0., 1., 0., 1.).
'SIMFAC3'	(R)	円筒座標と球座標のスケーリングファクター (初期値は 1.).
'XORG3', 'YORG3', 'ZORG3'	(R)	円筒座標と球座標の原点位置 (初期値は 0., 0., 0.).
'XEYE3','YEYE3','ZEYE3'	(R)	視点 (初期値は-0.8, -1.5, 1.5).
'XOBJ3','YOBJ3','ZOBJ3'	(R)	焦点中心 (初期値は 0.5, 0.5, 0.0).
'ANGLE3'	(R)	画角 (初期値は 30.0).
'TILT3'	(R)	傾き (初期値は 0.0).
'XOFF3','YOFF3'	(R)	オフセット (初期値は 0.0, 0.0).
'IXC3','IYC3','SEC3'	(I,R)	2 次元平面位置 (初期値は 1, 2, 0.0).
'L2TO3'	(L)	2 次元透視変換の指定. 'L2TO3' が .TRUE. なら 2 次元平面を 3 次元空間に割り付ける; .FALSE. なら V 座標と R 座標を同一のものとして扱う. (初期値は .FALSE.).

[補間およびクリッピングに関するパラメタ] - これらのパラメタの意味については, 1.7節 を参照のこと.

'LLNINT','LGCINT'	(L)	線形補間, 大円補間のフラグ (初期値は .FALSE., .TRUE.).
'RDX','RDY'	(R)	補間間隔 (初期値は 5.0, 5.0; 単位は度).

'LCLIP' (L) ビューポートでのクリッピングのフラグ (初期値は.FALSE.).

'TXMIN','TXMAX','TYMIN','TYMAX' (R) T 座標系でのクリッピング境界 (初期値は-180.0, 180.0, -90.0, 90.0; 単位は度).

[レイアウトに関するパラメーター] - dcl-ver.4 まで SLPGET/SLPSET が管理していたパラメーター.

'INDEX' (I) 第 1 レベル目のマージンに描く文字列とコーナーマークのラインインデクス (初期値は 1).

'LCORNER' (L) コーナーマークを描くかどうかを指定する. .TRUE. ならコーナーマークを描く; .FALSE. なら描かない (初期値は.TRUE.).

'CORNER' (R) コーナーマークの長さ (初期値は 0.01). 単位は第 1 レベル目の長辺を 1 とする.

'LWIDE' (L) 文字列 (タイトル) を描く第 1 レベル目のマージンの定義をする. .TRUE. のときマージンの幅を第 1 レベル目のフレーム幅いっぱいにとる; .FALSE. のとき横方向のマージンを取った残りの部分をマージンの幅とする (初期値は.FALSE.).

'XPAD' (R) 第 1 レベル目のマージンに描く文字列の左右にとるすきまの和を文字高を単位として指定する (初期値は 1.0).

'YPAD' (R) 第 1 レベル目のマージンに描く文字列の上下にとるすきまの和を文字高を単位として指定する (初期値は 1.0).

4. 備考

(a) 内部変数を管理するための下請けルーチンとして以下のものがある.

SGPQNP(NCP)	内部変数の総数 NCP を求める.
SGPQID(CP,IDX)	内部変数 CP の位置 IDX を求める.
SGPQCP(IDX,CP)	IDX の位置にある内部変数の名前 CP を参照する.
SGPQVL(IDX,IPARA)	IDX の位置にある内部変数の値 IPARA を参照する.
SGPSVL(IDX,IPARA)	IDX の位置にある内部変数の値 IPARA を変更する.

(b) SGpGET は上に述べた SGPQID を呼んで内部変数の位置を求め, SGPQVL によって値を参照する; SGpSET は SGPQID を呼んで内部変数の位置を求め, SGPSVL によって値を設定する. したがって指定した内部変数名が見つからないとき, エラーメッセージは SGPQID が出力する.

(c) IPARA としては適切な型の定数または変数を指定すること.

(d) このほかに, soft fill によって斜線を描こうとしたとき斜線の回転角に加える微小値を指定する 'RFAROT' という内部変数や, レイアウトに関する'NFRAME', 'NPAGE', 'NLEVEL', 'FACTOR' などの内部変数があるがユーザーは指定してはならない.

2.4.5 SGPWSN

1. 機能

利用可能なワークステーション名のリストを書き出す.

2. 呼び出し方法

```
CALL SGPWSN
```

3. パラメーターの説明

なし

4. 備考

(a) リストを書き出す装置番号は GL*p*GET/GL*p*SET の管理する内部変数'IOUNIT' が決める.

2.5 サブルーチンの説明： 正規化変換

正規化変換の設定に関するサブルーチン群. 実際の正規化変換関数は STPACK に含まれる.

以下のルーチンが設定する正規化変換に関するパラメータは, GRPH1 の内部変数を管理するルーチン SG*p*GET/SG*p*SET により一つずつ参照/変更することもできる. 以下のルーチンは, 内部的に SG*p*SET を用いて, 内部変数を設定している. したがって, これらのルーチンを使って設定するパラメータは, 実行時オプションによる変更ができない. 実行時オプションによる変更をしたいときは GRPH2 の GRPACK を用いればよい.

2.5.1 SGSVPT

1. 機能

   正規化変換のためのビューポートを設定する.

2. 呼び出し方法

```
CALL SGSVPT(VXMIN,VXMAX,VYMIN,VYMAX)
```

3. パラメーターの説明

VXMIN,VXMAX,VYMIN,VYMAX	(R)	ビューポート. 1.4節参照.

4. 備考

   (a) 現在設定されているビューポートは

```
CALL SGQVPT(VXMIN,VXMAX,VYMIN,VYMAX)
```

   によって参照できる.

   (b) これらのパラメータは SGSTRF が呼ばれた時点で有効になる.

2.5.2 SGSWND/SGSSIM/SGSMPL

1. 機能

   正規化変換のためのパラメータを設定する.

2. 呼び出し方法

```
CALL SGSWND(UXMIN,UXMAX,UYMIN,UYMAX)
CALL SGSSIM(SIMFAC,VXOFF,VYOFF)
CALL SGSMPL(PLX,PLY,PLROT)
```

3. パラメーターの説明

UXMIN,UXMAX,UYMIN,UYMAX	(R)	ウインドウ. 1.4節参照.
SIMFAC	(R)	相似変換のスケーリングファクター. 1.4節参照.
VXOFF,VYOFF	(R)	原点のオフセット. VXOFF,VYOFF が 0 の時, 原点はビューポートの中心に設定される. 1.4節参照.
PLX,PLY,PLROT	(R)	地図投影の際の回転角. 1.4節参照.

4. 備考

(a) 現在設定されているパラメータは

```
CALL SGQWND(UXMIN,UXMAX,UYMIN,UYMAX)
CALL SGQSIM(SIMFAC,VXOFF,VYOFF)
CALL SGQMPL(PLX,PLY,PLROT)
```

によって参照できる.

(b) UXMIN≠UXMAX, UYMIN≠UYMAX さらに対数変換の場合は正の領域で定義しなければならない.

(c) 円錐図法の標準緯線は SG*p*GET/SG*p*SET が管理するパラメーター'STLAT1', 'STLAT2' で設定する.

(d) これらのパラメータは SGSTRF が呼ばれた時点で有効になる.

2.5.3 SGSTRN

1. 機能

正規化変換の変換関数番号を指定する.

2. 呼び出し方法

```
CALL SGSTRN(ITR)
```

3. パラメーターの説明

ITR (I) 変換関数番号 (1.4節参照).

4. 備考

(a) 現在設定されている変換関数番号は

```
CALL SGQTRN(ITR)
```

によって参照できる.

(b) 変換関数番号, 名称, 略称の間の対応関係などを与える以下のような関数 ・サブルーチンがある (ISGTRC が関数でそれ以外はサブルーチンである).

ISGTRC(CTR)	名称あるいは略称を与えて変換関数番号を返す整数型の関数.
SGTRQF(NTX,LTR)	変換関数番号 NTX が定義されているかどうかを調べる. 定義されていれば LTR として .TRUE. が返される; 定義されていなければ .FALSE. が返される.
SGTRSL(CTS,CTL)	略称から名称を求める.
SGTRSN(CTS,NTX)	略称から変換関数番号を求める.
SGTRLS(CTL,CTS)	名称から略称を求める.
SGTRLN(CTL,NTX)	名称から変換関数番号を求める.
SGTRNS(NTX,CTS)	変換関数番号から略称を求める.
SGTRNL(NTX,CTL)	変換関数番号から名称を求める.

2.5.4 SGSTRF

1. 機能

   正規化変換を確定させる.

2. 呼び出し方法

   ```
   CALL SGSTRF
   ```

3. パラメーターの説明

   なし.

4. 備考

   (a) このルーチンは SG*p*GET/SG*p*SET によって管理されている変換パラメタに基づいて, STPACK のルーチン群の設定を行なう.

   (b) このルーチンを呼ぶ前に, 必要なパラメタを設定しておかなければならない.

2.6 サブルーチンの説明 : ポリラインプリミティブ

折れ線の図形出力 (polyline primitive) に関するサブルーチン群.

GL*p*GET/GL*p*SET が管理する内部変数'LMISS' を.TRUE. にすると欠損値処理をおこなう. つまり欠損値の前後は線で結ばない.

内部変数'LCHAR' を.TRUE. にするとラベル付きの折れ線を描く. ここでいうラベル付き折れ線とは, 描くべき線分のある長さを 1 サイクルとして, その一部分を空白域としその空白部分に指定した文字列を描くものである. ラベルとしてつける文字列は SGSPLC で指定する (2.6.5節参照); その文字列の高さは SGSPLS で指定する (2.6.6節参照). サイクルの定義などに関しては SG*p*GET/SG*p*SET が管理する内部変数を参照のこと (2.4.4節).

なお, 内部変数'LCHAR' を.TRUE. とした効果は線分描画ルーチンすべてに及ぶ. したがって, 'LCHAR' を.TRUE. としてラベルつき線分を描いたあとは必ず.FALSE. に戻しておかなくてはいけない.

2.6.1 SGPLZU/SGPLZV/SGPLZR

1. 機能

   U/V/R 座標系で折れ線を描く. 属性も同時に指定する.

2. 呼び出し方法

```
CALL SGPLZU(N,UPX,UPY,ITYPE,INDEX)
CALL SGPLZV(N,VPX,VPY,ITYPE,INDEX)
CALL SGPLZR(N,RPX,RPY,ITYPE,INDEX)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY などの長さ.
UPX, UPY	(R)	長さ N の配列. 折れ線を結ぶ点の U 座標系における (X, Y) 座標の値を与える.
VPX, VPY	(R)	長さ N の配列. 折れ線を結ぶ点の V 座標系における (X, Y) 座標の値を与える.
RPX, RPY	(R)	長さ N の配列. 折れ線を結ぶ点の R 座標系における (X, Y) 座標の値を与える.
ITYPE	(I)	ラインタイプ. SGSPLT 参照.
INDEX	(I)	ラインインデクス. SGSPLI 参照.

4. 備考

   (a) N は 2 以上でなければならない.

   (b) ここで指定した ITYPE, INDEX は, SGPLU, SGPLV, SGPLR が参照する ITYPE, INDEX を変更しない.

2.6.2 SGPLU/SGPLV/SGPLR

1. 機能

   U/V/R 座標系で折れ線を描く.

2. 呼び出し方法

```
CALL SGPLU(N,UPX,UPY)
CALL SGPLV(N,VPX,VPY)
CALL SGPLR(N,RPX,RPY)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY などの長さ.
UPX, UPY	(R)	長さ N の配列. 折れ線を結ぶ点の U 座標系における (X, Y) 座標の値を与える.
VPX, VPY	(R)	長さ N の配列. 折れ線を結ぶ点の V 座標系における (X, Y) 座標の値を与える.
RPX, RPY	(R)	長さ N の配列. 折れ線を結ぶ点の R 座標系における (X, Y) 座標の値を与える.

4. 備考

(a) N は 2 以上でなければならない.

(b) 描く折れ線の属性は SGSPLT, SGSPLI で設定する.

2.6.3 SGSPLT

1. 機能

SGPLU, SGPLV, SGPLR で描く折れ線のラインタイプを設定する.

2. 呼び出し方法

```
CALL SGSPLT(ITYPE)
```

3. パラメーターの説明

ITYPE	(I)	折れ線の線種 (初期値は 1). ITYPE が 1 のとき実線, 2 のとき破線, 3 のとき点線, 4 のとき 1 点鎖線となる. その他の 0 以外の整数は下位 N ビット (N は内部変数'NBITS' で決まる値. 初期値は 16; 2.4.4節参照) のビットパターンを用いて線種が設定される. たとえば N=16 で ITYPE = Z'0000F0F0' (16 進定数) のとき, '4bits ON 4bits OFF 4bits ON 4bits OFF' のような破線が設定される. 1 ビット当たりの長さは内部変数'BITLEN' が決める. ラインタイプの作画例は 8.1.1節参照.

4. 備考

(a) ITYPE は 0 以外でなければならない. ただし ITYPE が 0 のとき, メッセージは出力されるがなにもしない.

(b) ITYPE の妥当性はこのルーチンでチェックされず, SGPLU, SGPLV, SGPLR を呼んだときにチェックされる.

(c) 現在設定されているラインタイプは

```
CALL SGQPLT(ITYPE)
```

によって参照できる.

2.6.4 SGSPLI

1. 機能

   SGPLU, SGPLV, SGPLR で描く折れ線のラインインデクスを設定する.

2. 呼び出し方法

   ```
   CALL SGSPLI(INDEX)
   ```

3. パラメーターの説明

   INDEX (I) 折れ線のラインインデクス (1.7節参照; 初期値は 1).

4. 備考

   (a) INDEX は 1 以上でなければならない. ただし INDEX が 0 のとき, メッセージは出力されるがなにもしない.

   (b) INDEX の妥当性はこのルーチンでチェックされず, SGPLU, SGPLV, SGPLR を呼んだときにチェックされる.

   (c) 現在設定されているラインインデクスは

   ```
   CALL SGQPLI(INDEX)
   ```

   によって参照できる.

2.6.5 SGSPLC

1. 機能

   ラベルつき折れ線を描くとき, その文字列を指定する.

2. 呼び出し方法

   ```
   CALL SGSPLC(CHARX)
   ```

3. パラメーターの説明

   CHARX (R) 描く文字列 (初期値は'A').

4. 備考

   (a) 設定する文字列の長さは 32 文字以下でなければならない.

   (b) 現在設定されている文字列は

   ```
   CALL SGQPLC(CHARX)
   ```

   によって参照できる.

2.6.6 SGSPLS

1. 機能

   ラベルつき折れ線を描くとき, 文字列の高さを指定する.

2. 呼び出し方法

   ```
   CALL SGSPLS(RSIZE)
   ```

3. パラメーターの説明

   RSIZE (R) 文字列の高さを V 座標系の単位で指定する (初期値は 0.02).

4. 備考

   (a) RSIZE は 0 より大きくなければならない. ただし RSIZE が 0 のとき, メッセージは出力されるがなにもしない.

   (b) 現在設定されている文字列の高さは

   ```
   CALL SGQPLS(RSIZE)
   ```

   によって参照できる.

2.6.7 SGNPLC

1. 機能

   設定されている文字列の最後の文字の文字番号を 1 つ増やす.

2. 呼び出し方法

   ```
   CALL SGNPLC
   ```

3. パラメーターの説明

   なし.

4. 備考

   (a) このルーチンを呼ぶと, 設定されている文字列の最後の文字の文字番号が 1 つふやされる. つまり文字列として'X=1' が指定されていたとすると, SGNPLC を呼ぶことによって設定されている文字列が'X=2' となる.

2.7 サブルーチンの説明 : ポリマーカープリミティブ

マーカー列の図形出力 (polymarker primitive) に関するサブルーチン群.

GL*p*GET/GL*p*SET が管理する内部変数'LMISS' を.TRUE. にすると欠損値処理をおこなう. つまり欠損値の点にはマーカーを打たない.

また, 内部変数'NPMSKIP'(2.4.4節参照) を変更することによって何点かに 1 個の割合でマーカーを描くこともできる.

2.7.1 SGPMZU/SGPMZV/SGPMZR

1. 機能

   U/V/R 座標系でマーカー列を描く. 属性も同時に指定する.

2. 呼び出し方法

```
CALL SGPMZU(N,UPX,UPY,ITYPE,INDEX,RSIZE)
CALL SGPMZV(N,VPX,VPY,ITYPE,INDEX,RSIZE)
CALL SGPMZR(N,RPX,RPY,ITYPE,INDEX,RSIZE)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY のなど長さ.
UPX, UPY	(R)	長さ N の配列. マーカーを打つ点の U 座標系における (X, Y) 座標の値を与える.
VPX, VPY	(R)	長さ N の配列. マーカーを打つ点の V 座標系における (X, Y) 座標の値を与える.
RPX, RPY	(R)	長さ N の配列. マーカーを打つ点の R 座標系における (X, Y) 座標の値を与える.
ITYPE	(I)	マーカータイプ. SGSPMT 参照.
INDEX	(I)	マーカーのラインインデクス. SGSPMI 参照.
RSIZE	(R)	マーカーのサイズ. SGSPMS 参照.

4. 備考

   (a) N は 1 以上でなければならない.

   (b) ここで指定した ITYPE, INDEX, RSIZE は, SGPMU, SGPMV, SGPMR が参照する ITYPE, INDEX, RSIZE を変更しない.

2.7.2 SGPMU/SGPMV/SGPMR

1. 機能

   U/V/R 座標系でマーカー列を描く.

2. 呼び出し方法

```
CALL SGPMU(N,UPX,UPY)
CALL SGPMV(N,VPX,VPY)
CALL SGPMR(N,RPX,RPY)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY のなど長さ.
UPX, UPY	(R)	長さ N の配列. マーカーを打つ点の U 座標系における (X, Y) 座標の値を与える.
VPX, VPY	(R)	長さ N の配列. マーカーを打つ点の V 座標系における (X, Y) 座標の値を与える.
RPX, RPY	(R)	長さ N の配列. マーカーを打つ点の R 座標系における (X, Y) 座標の値を与える.

4. 備考

   (a) N は 1 以上でなければならない.

   (b) 描くマーカーの属性は SGSPMT, SGSPMI, SGSPMS で設定する.

2.7.3 SGSPMT

1. 機能

   SGPMU, SGPMV, SGPMR で描くマーカータイプを設定する.

2. 呼び出し方法

```
CALL SGSPMT(ITYPE)
```

3. パラメーターの説明

ITYPE	(I)	マーカータイプ (初期値は 1). ITYPE が 1 のとき'·', 2 のとき'+', 3 のとき'∗', 4 のとき'∘' のように, DCL 文字番号 (2.12節および 8.1.2, 8.1.3節のフォントテーブル参照) に対応する文字を描く. たとえば ITYPE = 152 のとき, 'α' を描く.

4. 備考

   (a) ITYPE は 0 以外でなければならない. ただし ITYPE が 0 のとき, メッセージは出力されるがなにもしない.

   (b) ITYPE の妥当性はこのルーチンでチェックされず, SGPMU, SGPMV, SGPMR を呼んだときにチェックされる.

   (c) 現在設定されているマーカータイプは

```
CALL SGQPMT(ITYPE)
```

   によって参照できる.

2.7.4 SGSPMI

1. 機能

   SGPMU, SGPMV, SGPMR で描くマーカーのラインインデクスを設定する.

2. 呼び出し方法

   ```
   CALL SGSPMI(INDEX)
   ```

3. パラメーターの説明

   INDEX (I) マーカーを描く線のラインインデクス (1.7節参照; 初期値は 1).

4. 備考

   (a) INDEX は 1 以上でなければならない. ただし INDEX が 0 のとき, メッセージは出力されるがなにもしない.

   (b) INDEX の妥当性はこのルーチンでチェックされず, SGPMU, SGPMV, SGPMR を呼んだときにチェックされる.

   (c) 現在設定されているマーカーのラインインデクスは

   ```
   CALL SGQPMI(INDEX)
   ```

   によって参照できる.

2.7.5 SGSPMS

1. 機能

   SGPMU, SGPMV, SGPMR で描くマーカーの大きさを設定する.

2. 呼び出し方法

   ```
   CALL SGSPMS(RSIZE)
   ```

3. パラメーターの説明

   RSIZE (R) マーカーの大きさを V-座標系における単位で指定する (初期値は 0.01).

4. 備考

   (a) RSIZE は 0 より大きくなければならない. ただし RSIZE が 0 のとき, メッセージは出力されるがなにもしない.

   (b) RSIZE の妥当性はこのルーチンでチェックされず, SGPMU, SGPMV, SGPMR を呼んだときにチェックされる.

   (c) 現在設定されているマーカーの大きさは

   ```
   CALL SGQPMS(RSIZE)
   ```

   によって参照できる.

2.8 サブルーチンの説明： テキストプリミティブ

文字列の図形出力 (text primitive) に関するサブルーチン群.

GRPH1 では 2 種類のフォントが使用できる. フォント番号は内部変数'IFONT' を指定することによって選択できる (初期値は 1). 2.4.4節, およびフォントテーブル (8.1.2, 8.1.3節) 参照.

普通, キーボードにある文字はそのまま表示される. キーボードにない文字, あるいはキーボードにあっても正しく表現されない文字は, 関数副プログラム CSGI (2.12.1節参照) の引数としてフォントテーブルに示してある番号 (DCL 文字番号) を与えることによって描くことができる. 詳しくは, 2.12節を参照のこと.

また内部変数'LCNTL' が .TRUE. なら, 制御文字を有効として上付および下付添え字を描くことができる (2.4.4節参照). たとえば, 制御文字の文字番号が 2.4.4節で設定してある値のとき,

$(X_i)^2$

と描くためには

```
'(X_i")|2"'
```

と指定すればよい (いちばん最後の上付および下付添え字のモードの終わりをしめす制御文字は省略できない). 制御文字の指定方法が妥当かどうかは, 文字の長さと高さを求める下位ルーチン SZQTXW がチェックする. したがって, 指定方法が妥当でないときエラーメッセージは SZQTXW から出力される.

文字列の有効な長さは MATH1 (下位基本数学処理ライブラリ) の中の関数 LENC (CHRLIB 参照) を用いて決定される. したがって, 与えた文字列の後方にある NULL 文字あるいは空白文字は無視される.

文字は基本的に線分の集まりとして表現されるのでクリッピング (1.7節参照) の対象となる.

2.8.1 SGTXZU/SGTXZV/SGTXZR

1. 機能

   U/V/R 座標系で文字列を描く. 属性も同時に指定する.

2. 呼び出し方法

```
CALL SGTXZU(UX,UY,CHARS,RSIZE,IROTA,ICENT,INDEX)
CALL SGTXZV(VX,VY,CHARS,RSIZE,IROTA,ICENT,INDEX)
CALL SGTXZR(RX,RY,CHARS,RSIZE,IROTA,ICENT,INDEX)
```

3. パラメーターの説明

UX, UY	(R)	U 座標系における文字の位置の (X, Y) 座標値を与える.
VX, VY	(R)	V 座標系における文字の位置の (X, Y) 座標値を与える.
RX, RY	(R)	R 座標系における文字の位置の (X, Y) 座標値を与える.
CHARS	(C*(*))	描く文字列.
RSIZE	(R)	文字の高さ. SGSTXS 参照.
IROTA	(I)	文字列の角度. SGSTXR 参照.
ICENT	(I)	センタリングオプション. SGSTXC 参照.
INDEX	(I)	文字のラインインデクス. SGSTXI 参照.

4. 備考

   (a) ここで指定した RSIZE, IROTA, ICENT, INDEX は, SGTXU, SGTXV, SGTXR が参照する RSIZE, IROTA, ICENT, INDEX を変更しない.

2.8.2 SGTXU/SGTXV/SGTXR

1. 機能

   U/V/R 座標系で文字列を描く.

2. 呼び出し方法

```
CALL SGTXU(UX,UY,CHARS)
CALL SGTXV(VX,VY,CHARS)
CALL SGTXR(RX,RY,CHARS)
```

3. パラメーターの説明

UX, UY	(R)	U 座標系における文字の位置の (X, Y) 座標値を与える.
VX, VY	(R)	V 座標系における文字の位置の (X, Y) 座標値を与える.
RX, RY	(R)	R 座標系における文字の位置の (X, Y) 座標値を与える.
CHARS	(C*(*))	描く文字列.

4. 備考

   (a) 描く文字列の属性は SGSTXS, SGSTXR, SGSTXC, SGSTXI で設定する.

2.8.3 SGSTXS

1. 機能

   SGTXU, SGTXV, SGTXR で描く文字の高さを設定する.

2. 呼び出し方法

```
CALL SGSTXS(RSIZE)
```

3. パラメーターの説明

RSIZE	(R)	文字の高さを V 座標系における単位で指定する (初期値は 0.05).

4. 備考

   (a) RSIZE は 0 より大きくなければならない. ただし RSIZE が 0 のとき, メッセージは出力されるがなにもしない.

   (b) RSIZE の妥当性はこのルーチンでチェックされず, SGTXU, SGTXV, SGTXR を呼んだときにチェックされる.

   (c) 現在設定されている文字の高さは

```
CALL SGQTXS(RSIZE)
```

   によって参照できる.

2.8.4 SGSTXR

1. 機能

   SGTXU, SGTXV, SGTXR で描く文字列の回転角を設定する.

2. 呼び出し方法

   ```
   CALL SGSTXR(IROTA)
   ```

3. パラメーターの説明

   IROTA (I) 文字列の傾きを度の単位で与える (初期値は 0).

4. 備考

   (a) 現在設定されている回転角は

   ```
   CALL SGQTXR(IROTA)
   ```

   によって参照できる.

2.8.5 SGSTXC

1. 機能

   SGTXU, SGTXV, SGTXR で描く文字列のセンタリングオプション を設定する.

2. 呼び出し方法

   ```
   CALL SGSTXC(ICENT)
   ```

3. パラメーターの説明

   ICENT (I) センタリングオプション (初期値は 0). $-1, 0, +1$ のいずれかを指定する.

   -1 : 左あわせ
   0 : 中央あわせ
   $+1$: 右あわせ

4. 備考

   (a) ICENT の妥当性はこのルーチンでチェックされず, SGTXU, SGTXV, SGTXR を呼んだときにチェックされる.

   (b) 現在設定されているセンタリングオプション は

   ```
   CALL SGQTXC(ICENT)
   ```

   によって参照できる.

2.8.6 SGSTXI

1. 機能

   SGTXU, SGTXV, SGTXR で描く文字のラインインデクスを設定する.

2. 呼び出し方法

   ```
   CALL SGSTXI(INDEX)
   ```

3. パラメーターの説明

   INDEX (I) 文字を描く線のラインインデクス (1.7節参照; 初期値は 1).

4. 備考

   (a) INDEX は 1 以上でなければならない. ただし INDEX が 0 のとき, メッセージは出力されるがなにもしない.

   (b) INDEX の妥当性はこのルーチンでチェックされず, SGTXU, SGTXV, SGTXR を呼んだときにチェックされる.

   (c) 現在設定されている文字のラインインデクスは

   ```
   CALL SGQTXI(INDEX)
   ```

   によって参照できる.

2.9 サブルーチンの説明： トーンプリミティブ

多角形閉領域のぬりつぶし図形出力 (tone primitive) に関するサブルーチン群.

出力装置の能力に応じて, hard fill または soft fill を切替える. 内部変数'LSOFTF' (2.4.4節参照) が.TRUE. のときは soft fill をおこなう; .FALSE. のときは hard fill をおこなう (初期値は.FALSE.). ただし, hard fill を指定してもその能力がない場合は soft fill となる.

また, 色を用いたべた塗り (下位 3 桁が 999 であるようなトーンパターン番号) による塗りわけをおこなうとき, 色番号をドットなどのパターン番号として読みかえることができる. 内部変数'LCL2TN' (2.4.4節参照) が.TRUE. のときは読みかえる; .FALSE. のときは読みかえない (初期値は.FALSE.). ただし, 色を用いた塗りわけができないような装置では, 'LCL2TN' が.FALSE. と設定されていても, パターン番号による読みかえをおこなおうとする.

すでに描かれた図形の上に soft fill をおこなってもトーンパターンが上書きされるだけで, それまでに描かれた図形が (べた塗りをしない限り) 消えてしまうことはない. しかしながら, 出力装置によっては hard fill によるぬりわけによって先に描かれた図形が消えてしまうことがあるので, ほかの図形出力と一緒にトーンプリミティブを用いるときは描く順番に注意すること.

以下のサブルーチンの説明で, 多角形を定義する頂点の座標 (PX(i), PY(i)) ($i = 1$,N) (これらは U/V/R 座標系で指定できる) に関しては, 次のように解釈される. (PX(1), PY(1)) と (PX(N), PY(N)) は同一な点である必要はない. つまり, (PX(i), PY(i)) と (PX($i+1$), PY($i+1$)) ($i = 1$,N-1) および (PX(N), PY(N)) と (PX(1), PY(1)) を結んだ閉領域をぬりつぶす.

現在サポートされているサブルーチン群は, 内部的に確保している作業領域の制約から座標値を与える配列の長さ N は 256 以下でなければならない.

2.9.1 SGTNZU/SGTNZV/SGTNZR

1. 機能

   U/V/R 座標系で多角形閉領域をぬりつぶす. トーンパターン番号も同時に指定する.

2. 呼び出し方法

```
CALL SGTNZU(N,UPX,UPY,ITPAT)
CALL SGTNZV(N,VPX,VPY,ITPAT)
CALL SGTNZR(N,RPX,RPY,ITPAT)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY の長さ.
UPX, UPY	(R)	長さ N の配列. 多角形を定義する頂点の U 座標系における (X, Y) 座標の値を与える.
VPX, VPY	(R)	長さ N の配列. 多角形を定義する頂点の U 座標系における (X, Y) 座標の値を与える.
RPX, RPY	(R)	長さ N の配列. 多角形を定義する頂点の U 座標系における (X, Y) 座標の値を与える.
ITPAT	(I)	トーンパターン番号. SGSTNP 参照.

4. 備考

   (a) N は 3 以上でなければならない.

   (b) ここで指定した ITPAT は, SGTNU, SGTNV, SGTNR が参照する ITPAT を変更しない.

2.9.2 SGTNU/SGTNV/SGTNR

1. 機能

   U/V/R 座標系で多角形閉領域をぬりつぶす.

2. 呼び出し方法

```
CALL SGTNU(N,UPX,UPY)
CALL SGTNV(N,VPX,VPY)
CALL SGTNR(N,RPX,RPY)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY の長さ.
UPX, UPY	(R)	長さ N の配列. 多角形を定義する頂点の U 座標系における (X, Y) 座標の値を与える.
VPX, VPY	(R)	長さ N の配列. 多角形を定義する頂点の V 座標系における (X, Y) 座標の値を与える.
RPX, RPY	(R)	長さ N の配列. 多角形を定義する頂点の R 座標系における (X, Y) 座標の値を与える.

4. 備考

   (a) N は 3 以上でなければならない.

   (b) 領域をぬりつぶすトーンパターン番号は SGSTNP で設定する.

2.9.3 SGSTNP

1. 機能

   SGTNU, SGTNV, SGTNR で用いるトーンパターン番号を設定する.

2. 呼び出し方法

```
CALL SGSTNP(ITPAT)
```

3. パラメーターの説明

ITPAT	(I)	トーンパターン番号 (初期値は 1). 8.1.4- 8.1.10節参照.

4. 備考

   (a) ITPAT は 1 以上でなければならない. ただし ITPAT が 0 のとき, メッセージは出力されるがなにもしない.

   (b) ITPAT の妥当性はこのルーチンでチェックされず, SGTNU, SGTNV, SGTNR を呼んだときにチェックされる.

(c) 現在設定されているトーンパターン番号は

```
CALL SGQTNP(ITPAT)
```

によって参照できる.

2.10 サブルーチンの説明 : ラインサブプリミティブ

線分の図形出力 (line sub-primitive) に関するサブルーチン群.

なお, このラインサブプリミティブは, 次節に述べるアローサブプリミティブの特殊な場合 (矢羽部分がないもの) と考えられるので, 将来的には削除される可能性がある.

2.10.1 SGLNZU/SGPLZV/SGPLZR

1. 機能

   U/V/R 座標系で線分を描く. 属性も同時に指定する.

2. 呼び出し方法

```
CALL SGLNZU(UX1,UY1,UX2,UY2,INDEX)
CALL SGLNZV(VX1,VY1,VX2,VY2,INDEX)
CALL SGLNZR(RX1,RY1,RX2,RY2,INDEX)
```

3. パラメーターの説明

UX1,UY1	(R)	U 座標系における線分の始点の座標.
UX2,UY2	(R)	U 座標系における線分の終点の座標.
VX1,VY1	(R)	V 座標系における線分の始点の座標.
VX2,VY2	(R)	V 座標系における線分の終点の座標.
RX1,RY1	(R)	R 座標系における線分の始点の座標.
RX2,RY2	(R)	R 座標系における線分の終点の座標.
INDEX	(I)	SGSLNI 参照.

4. 備考

   (a) ここで指定した INDEX は, SGLNU, SGLNV, SGLNR が参照する INDEX を変更しない.

2.10.2 SGLNU/SGLNV/SGLNR

1. 機能

   U/V/R 座標系で線分を描く.

2. 呼び出し方法

```
CALL SGLNU(UX1,UY1,UX2,UY2)
CALL SGLNV(VX1,VY1,VX2,VY2)
CALL SGLNR(RX1,RY1,RX2,RY2)
```

3. パラメーターの説明

UX1,UY1	(R)	U 座標系における線分の始点の座標.
UX2,UY2	(R)	U 座標系における線分の終点の座標.
VX1,VY1	(R)	V 座標系における線分の始点の座標.
VX2,VY2	(R)	V 座標系における線分の終点の座標.
RX1,RY1	(R)	R 座標系における線分の始点の座標.
RX2,RY2	(R)	R 座標系における線分の終点の座標.

4. 備考

   (a) 描く線分の属性は SGSLNI で設定する.

2.10.3 SGSLNI

1. 機能

   SGLNU, SGLNV, SGLNZR で描く線分のラインインデクスを設定する.

2. 呼び出し方法

```
CALL SGSLNI(INDEX)
```

3. パラメーターの説明

   INDEX (I) 線分のラインインデクス (1.7節参照; 初期値は 1).

4. 備考

   (a) INDEX は 1 以上でなければならない. ただし INDEX が 0 のとき, メッセージは出力されるがなにもしない.

   (b) INDEX の妥当性はこのルーチンでチェックされず, SGLNU, SGLNV, SGLNR を呼んだときにチェックされる.

   (c) 現在設定されているラインインデクスは

```
CALL SGQLNI(INDEX)
```

   によって参照できる.

2.11 サブルーチンの説明 : アローサブプリミティブ

矢印付き線分の図形出力 (arrow sub-primitive) に関するサブルーチン群.

矢印をどのように表現するか (その長さやなす角など) は SG*p*GET/SG*p*SET の管理する内部変数できまる (2.4.4節参照).

2.11.1 SGLAZU/SGLAZV/SGLAZR

1. 機能

   U/V/R 座標系で矢印付き線分を描く. 属性も同時に指定する.

2. 呼び出し方法

```
CALL SGLAZU(UX1,UY1,UX2,UY2,ITYPE,INDEX)
CALL SGLAZV(VX1,VY1,VX2,VY2,ITYPE,INDEX)
CALL SGLAZR(RX1,RY1,RX2,RY2,ITYPE,INDEX)
```

3. パラメーターの説明

UX1,UY1	(R)	U 座標系における線分の始点の座標.
UX2,UY2	(R)	U 座標系における線分の終点の座標.
VX1,VY1	(R)	V 座標系における線分の始点の座標.
VX2,VY2	(R)	V 座標系における線分の終点の座標.
RX1,RY1	(R)	R 座標系における線分の始点の座標.
RX2,RY2	(R)	R 座標系における線分の終点の座標.
ITYPE	(I)	ラインタイプ. SGSLAT 参照.
INDEX	(I)	ラインインデクス. SGSLAI 参照.

4. 備考

   (a) ここで指定した ITYPE, INDEX は, SGLAU, SGLAV, SGLAR が参照する ITYPE, INDEX を変更しない.

2.11.2 SGLAU/SGLAV/SGLAR

1. 機能

   U/V/R 座標系で矢じり付き線分を描く.

2. 呼び出し方法

```
CALL SGLAU(UX1,UY1,UX2,UY2)
CALL SGLAV(VX1,VY1,VX2,VY2)
CALL SGLAR(RX1,RY1,RX2,RY2)
```

3. パラメーターの説明

UX1,UY1 (R) U 座標系における線分の始点の座標.
UX2,UY2 (R) U 座標系における線分の終点の座標.
VX1,VY1 (R) V 座標系における線分の始点の座標.
VX2,VY2 (R) V 座標系における線分の終点の座標.
RX1,RY1 (R) R 座標系における線分の始点の座標.
RX2,RY2 (R) R 座標系における線分の終点の座標.

4. 備考

(a) 描く線分の属性は SGSLAT, SGSLAI で設定する.

2.11.3 SGSLAT

1. 機能

SGLAU, SGLAV, SGLAR で描く線分のラインタイプを設定する.

2. 呼び出し方法

```
CALL SGSLAT(ITYPE)
```

3. パラメーターの説明

ITYPE (I) 線分のラインタイプ (1.7節参照; 初期値は 1).

4. 備考

(a) ITYPE は 0 以外でなければならない. ただし ITYPE が 0 のとき, メッセージは出力されるがなにもしない.

(b) ITYPE の妥当性はこのルーチンでチェックされず, SGLAU, SGLAV, SGLAR を呼んだときにチェックされる.

(c) 現在設定されているラインタイプは

```
CALL SGQLAT(ITYPE)
```

によって参照できる.

2.11.4 SGSLAI

1. 機能

SGLAU, SGLAV, SGLAR で描く線分のラインインデクスを設定する.

2. 呼び出し方法

```
CALL SGSLAI(INDEX)
```

3. パラメーターの説明

INDEX (I) 線分のラインインデクス (1.7節参照; 初期値は 1).

4. 備考

(a) INDEX は 1 以上でなければならない. ただし INDEX が 0 のとき, メッセージは出力されるがなにもしない.

(b) INDEX の妥当性はこのルーチンでチェックされず, SGLAU, SGLAV, SGLAR を呼んだときにチェックされる.

(c) 現在設定されているラインインデクスは

```
CALL SGQLAI(INDEX)
```

によって参照できる.

2.12 関数の説明： DCL 文字関数

GRPH1 で用いる文字コードの取り扱いに関する関数群.

GRPH1 では, フォントテーブル (8.1.2,8.1.3節) を見ると分かるように, 普通の ASCII 文字ばかりでなく, ギリシャ文字や特殊な記号も使用できる. テキストプリミティブで文字列を指定するには, キーボードにある文字はそのまま文字列として書き込めばよい. キーボードにない文字, あるいはキーボードにあっても正しく表現されない文字は, 関数副プログラム CSGI の引数としてフォントテーブルに示してある番号 (以下では DCL 文字番号という) を与えることによって指定する.

この文字関数 CSGI は ASCII コードを使用する計算機システムでは FORTRAN の CHAR 関数と同値である. しかし, たとえば EBCDIC コードを使用する計算機システムでは CHAR 関数で正しい文字が得られない. どのようなコード系を使用していても一意に文字を指定するための関数が CSGI である. なおこの逆関数として, ある文字の DCL 文字番号を返す ISGC がある.

CSGI の使用例としては, たとえば,

$\cos\alpha x$

と書きたいとき

```
CHARACTER CSGI*1
```

と宣言した上で

```
'cos'//CSGI(152)//'x'
```

のように文字列を生成してテキストプリミティブにわたせばよい.

なおここに記されている関数群はシステムに依存しているので, ソースコードは sgpack の下にはなく csgi という独立したディレクトリに収められている.

2.12.1 CSGI

1. 機能

   DCL 文字番号に対応する文字を返す.

2. 呼び出し方法

   ```
   CSGI(IC)
   ```

3. パラメーターの説明

IC	(I)	与える DCL 文字番号. 0 から 255 の整数値で指定する.
CSGI	(C*1)	DCL 文字番号に対応する文字.

4. 備考

   (a) この関数を用いるためには

   ```
   CHARACTER CSGI*1
   ```

   と宣言してなければならない.

   (b) ASCII コード系において CSGI は CHAR と同値である.

2.12.2 ISGC

1. 機能

   文字に対応する DCL 文字番号を返す.

2. 呼び出し方法

```
ISGC(CH)
```

3. パラメーターの説明

CH	(C*1)	与えるの文字 (文字長は 1).
ISGC	(I)	文字に対応する DCL 文字番号.

4. 備考

   (a) ASCII コード系において ISGC は ICHAR と同値である.

第 3 章

SCPACK : 3 次元ユーザーインターフェイス

3.1 概要

SCPACK は 3 次元関係のインターフェイスを集めたものである. ただし, 3 次元関係の部分でも変換関数のパラメタなどは基本的に SGPACK が管理しており, SCPACK にはまとめて指定するインターフェイスしかない.
SCPACK には, 3 次元正規化変換の設定ルーチン, 透視変換の設定ルーチン, 3 次元図形の基本要素 (出力プリミティブ) を作画するルーチンなどがある.

3.2 サブルーチンのリスト

3 次元正規変換

SCSVPT(VXMIN3,VXMAX3,VYMIN3,VYMAX3,VZMIN3,VZMAX3)	ビューポートの設定.
SCSWND(UXMIN3,UXMAX3,UYMIN3,UYMAX3,UZMIN3,UZMAX3)	ウィンドウの設定.
SCSLOG(LXLOG3,LYLOG3,LZLOG3)	対数軸の設定.
SCSORG(SIMFAC,VXORG3,VYORG3,VZORG3)	スケーリングファクターと原点の設定.
SCSTRN(ITR3)	変換関数番号の設定.
SCSTRF	変換関数の確定.

この他に, パラメタ参照ルーチンとして SCQVPT, SCQWND, SCQORG, SCQLOG, SCQTRN がある.

透視変換

SCSEYE(XEYE3, YEYE3, ZEYE3)	視点の設定.
SCSOBJ(XOBJ3, YOBJ3, ZOBJ3)	焦点の設定.
SCSPLN(IXC3, IYC3, SEC3)	2 次元平面の割付.
SCSPRJ	透視変換の確定.

この他に, パラメタ参照ルーチンとして SCQEYE, SCQOBJ, SCQPLN がある.

ポリライン

SCPLZU(N,UPX,UPY,UPZ,INDEX)	U 座標系で折れ線を描く.
SCPLZV(N,VPX,VPY,VPZ,INDEX)	V 座標系で折れ線を描く.
SCPLU(N,UPX,UPY,UPZ)	U 座標系で折れ線を描く.
SCPLV(N,VPX,VPY,VPZ)	V 座標系で折れ線を描く.
SCSPLI(INDEX)	ラインインデクスの設定.

この他に, パラメタ参照ルーチンとして SCQPLI がある.

ポリマーカー

SCPMZU(N,UPX,UPY,UPZ,ITYPE,INDEX,RSIZE)	U 座標系でマーカー列を描く.
SCPMZV(N,VPX,VPY,VPZ,ITYPE,INDEX,RSIZE)	V 座標系でマーカー列を描く.
SCPMU(N,UPX,UPY,UPZ)	U 座標系でマーカー列を描く.
SCPMV(N,VPX,VPY,VPZ)	V 座標系でマーカー列を描く.
SCSPMT(ITYPE)	マーカータイプの設定.
SCSPMI(INDEX)	マーカーのラインインデクスの設定.
SCSPMS(RSIZE)	マーカーの大きさ設定.

この他に, パラメタ参照ルーチンとして SCQPMT, SCQPMI, SCQPMS がある.

トーン

SCTNZU(UPX,UPY,UPZ,ITPAT1,ITPAT2)	U 座標系で多角形領域の塗りつぶし.
SCTNZV(VPX,VPY,VPZ,ITPAT1,ITPAT2)	V 座標系で多角形領域の塗りつぶし.
SCTNU(UPX,UPY,UPZ)	U 座標系で多角形領域の塗りつぶし.
SCTNV(VPX,VPY,VPZ)	V 座標系で多角形領域の塗りつぶし.
SCSTNP(ITPAT1,ITPAT2)	トーンパターン番号設定.

この他に, パラメタ参照ルーチンとして SCQTNP がある.

3.3 サブルーチンの説明： 3 次元正規化変換

正規化変換の設定に関するサブルーチン群. 実際の正規化変換関数は STPACK に含まれる.

以下のルーチンが設定する正規化変換に関するパラメータは, GRPH1 の内部変数を管理するルーチン SGpGET/SGpSET により一つずつ参照/変更することもできる. 以下のルーチンは, 内部的に SGpSET を用いて, 内部変数を設定している. したがって, これらのルーチンを使って設定するパラメータは, 実行時オプションによる変更ができない.

3.3.1 SCSVPT/SCSWND/SCSLOG/SCSORG

1. 機能

   3 次元正規化変換のためのパラメータを設定する.

2. 呼び出し方法

```
CALL SCSVPT(VXMIN3,VXMAX3,VYMIN3,VYMAX3,VZMIN3,VZMAX3)
CALL SCSWND(UXMIN3,UXMAX3,UYMIN3,UYMAX3,UZMIN3,UZMAX3)
CALL SCSLOG(LXLOG3,LYLOG3,LZLOG3)
CALL SCSORG(SIMFAC,VXORG3,VYORG3,VZORG3)
```

3. パラメーターの説明

VXMIN3,VXMAX3,VYMIN3, VYMAX3,VZMIN3,VZMAX3	(R)	直角直線座標のビューポート. 1.5節参照.
UXMIN3,UXMAX3,UYMIN3, UYMAX3,UZMIN3,UZMAX3	(R)	直角直線座標のウインドウ. 1.5節参照.
LXLOG3,LYLOG3,LZLOG3	(R)	直角直線座標の対数軸指定. 1.5節参照.
SIMFAC	(R)	円筒座標, 球座標のスケーリングファクター. 1.5節参照.
VXORG3,VYORG3,VZORG3	(R)	円筒座標, 球座標の原点の位置. 1.5節参照.

4. 備考

   (a) 現在設定されているパラメータは

```
CALL SCQVPT(VXMIN3,VXMAX3,VYMIN3,VYMAX3,VZMIN3,VZMAX3)
CALL SCQWND(UXMIN3,UXMAX3,UYMIN3,UYMAX3,VZMIN3,VZMAX3)
CALL SCSLOG(LXLOG3,LYLOG3,LZLOG3)
CALL SCQORG(SIMFAC,VXORG3,VYORG3,VZORG3)
```

   によって参照できる.

   (b) UXMIN3≠UXMAX3, UYMIN3≠UYMAX3, UZMIN3≠UZMAX3 さらに対数変換の場合は正の領域で定義しなければならない.

   (c) これらのパラメータは SCSTRF が呼ばれた時点で有効になる.

3.3.2 SCSTRN

1. 機能

   正規化変換の変換関数番号を指定する.

2. 呼び出し方法

```
CALL SCSTRN(ITR3)
```

3. パラメーターの説明

   ITR3 (I) 変換関数番号 (1.5節参照).

4. 備考

   (a) 現在設定されている変換関数番号は

```
CALL SCQTRN(ITR3)
```

   によって参照できる.

3.3.3 SCSTRF

1. 機能

   3 次元正規化変換を確定させる.

2. 呼び出し方法

```
CALL SCSTRF
```

3. パラメーターの説明

   なし.

4. 備考

   (a) このルーチンは SGpGET/SGpSET によって管理されている変換パラメタに基づいて, STPACK のルーチン群の設定を行なう.

   (b) このルーチンを呼ぶ前に, 必要なパラメタを設定しておかなければならない.

3.4 サブルーチンの説明： ポリラインプリミティブ

折れ線の図形出力 (polyline primitive) に関するサブルーチン群.

GL*p*GET/GL*p*SET が管理する内部変数'LMISS' を.TRUE. にすると欠損値処理をおこなう. つまり欠損値の前後は線で結ばない.

2 次元のポリラインと異なり, 実線しか描けない. また, ラベルをつけることもできない.

3.4.1 SCPLZU/SCPLZV

1. 機能

   U/V 座標系で折れ線を描く. 属性も同時に指定する.

2. 呼び出し方法

```
CALL SCPLZU(N,UPX,UPY,UPZ,ITYPE)
CALL SCPLZV(N,VPX,VPY,VPZ,ITYPE)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY, UPZ の長さ.
UPX, UPY, UPZ	(R)	長さ N の配列. 折れ線を結ぶ点の U 座標系における (X, Y, Z) 座標の値を与える.
VPX, VPY, VPZ	(R)	長さ N の配列. 折れ線を結ぶ点の V 座標系における (X, Y, Z) 座標の値を与える.
INDEX	(I)	ラインインデクス. SCSPLI 参照.

4. 備考

   (a) N は 2 以上でなければならない.

   (b) ここで指定した ITYPE, INDEX は, SCPLU, SCPLV が参照する INDEX を変更しない.

3.4.2 SCPLU/SCPLV

1. 機能

   U/V 座標系で折れ線を描く.

2. 呼び出し方法

```
CALL SCPLU(N,UPX,UPY,UPZ)
CALL SCPLV(N,VPX,VPY,VPZ)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY, UPZ の長さ.
UPX, UPY, UPZ	(R)	長さ N の配列. 折れ線を結ぶ点の U 座標系における (X, Y, Z) 座標の値を与える.
VPX, VPY, VPZ	(R)	長さ N の配列. 折れ線を結ぶ点の V 座標系における (X, Y, Z) 座標の値を与える.

4. 備考

   (a) N は 2 以上でなければならない.

   (b) 描く折れ線の属性は SCSPLI で設定する.

3.4.3 SCSPLI

1. 機能

   SCPLU, SCPLV で描く折れ線のラインインデクスを設定する.

2. 呼び出し方法

```
CALL SCSPLI(INDEX)
```

3. パラメーターの説明

INDEX	(I)	折れ線のラインインデクス (1.7節参照; 初期値は 1).

4. 備考

   (a) INDEX は 1 以上でなければならない. ただし INDEX が 0 のとき, メッセージは出力されるがなにもしない.

   (b) INDEX の妥当性はこのルーチンでチェックされず, SCPLU, SCPLV を呼んだときにチェックされる.

   (c) 現在設定されているラインインデクスは

```
CALL SCQPLI(INDEX)
```

   によって参照できる.

3.5 サブルーチンの説明： ポリマーカープリミティブ

マーカー列の図形出力 (polymarker primitive) に関するサブルーチン群.

GL*p*GET/GL*p*SET が管理する内部変数'LMISS' を.TRUE. にすると欠損値処理をおこなう. つまり欠損値の点にはマーカーを打たない.

また, 内部変数'NPMSKIP'(2.4.4節参照) を変更することによって何点かに 1 個の割合でマーカーを描くこともできる.

3.5.1 SCPMZU/SCPMZV

1. 機能

   U/V 座標系でマーカー列を描く. 属性も同時に指定する.

2. 呼び出し方法

```
CALL SCPMZU(N,UPX,UPY,UPZ,ITYPE,INDEX,RSIZE)
CALL SCPMZV(N,VPX,VPY,VPZ,ITYPE,INDEX,RSIZE)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY, UPZ の長さ.
UPX, UPY, UPZ	(R)	長さ N の配列. マーカーを打つ点の U 座標系における (X, Y, Z) 座標の値を与える.
VPX, VPY, VPZ	(R)	長さ N の配列. マーカーを打つ点の V 座標系における (X, Y, Z) 座標の値を与える.
ITYPE	(I)	マーカータイプ. SCSPMT 参照.
INDEX	(I)	マーカーのラインインデクス. SCSPMI 参照.
RSIZE	(R)	マーカーのサイズ. SCSPMS 参照.

4. 備考

   (a) N は 1 以上でなければならない.

   (b) ここで指定した ITYPE, INDEX, RSIZE は, SCPMU, SCPMV が参照する ITYPE, INDEX, RSIZE を変更しない.

3.5.2 SCPMU/SCPMV

1. 機能

   U/V 座標系でマーカー列を描く.

2. 呼び出し方法

```
CALL SCPMU(N,UPX,UPY,UPZ)
CALL SCPMV(N,VPX,VPY,VPZ)
```

3. パラメーターの説明

N	(I)	配列 UPX, UPY, UPZ の長さ.
UPX, UPY, UPZ	(R)	長さ N の配列. マーカーを打つ点の U 座標系における (X, Y, Z) 座標の値を与える.
VPX, VPY, VPZ	(R)	長さ N の配列. マーカーを打つ点の V 座標系における (X, Y, Z) 座標の値を与える.

4. 備考

(a) N は 1 以上でなければならない.

(b) 描くマーカーの属性は SCSPMT, SCSPMI, SCSPMS で設定する.

3.5.3 SCSPMT

1. 機能

SCPMU, SCPMV で描くマーカータイプを設定する.

2. 呼び出し方法

```
CALL SCSPMT(ITYPE)
```

3. パラメーターの説明

ITYPE	(I)	マーカータイプ (初期値は 1). ITYPE が 1 のとき'·', 2 のとき'+', 3 のとき'∗', 4 のとき'∘' のように, DCL 文字番号 (2.12節および 8.1.2, 8.1.3節のフォントテーブル参照) に対応する文字を描く. たとえば ITYPE = 152 のとき, 'α' を描く.

4. 備考

(a) ITYPE は 0 以外でなければならない. ただし ITYPE が 0 のとき, メッセージは出力されるがなにもしない.

(b) ITYPE の妥当性はこのルーチンでチェックされず, SCPMU, SCPMV を呼んだときにチェックされる.

(c) 現在設定されているマーカータイプは

```
CALL SCQPMT(ITYPE)
```

によって参照できる.

3.5.4 SCSPMI

1. 機能

SCPMU, SCPMV で描くマーカーのラインインデクスを設定する.

2. 呼び出し方法

```
CALL SCSPMI(INDEX)
```

3. パラメーターの説明

INDEX (I) マーカーを描く線のラインインデクス (1.7節参照; 初期値は 1).

4. 備考

   (a) INDEX は 1 以上でなければならない. ただし INDEX が 0 のとき, メッセージは出力されるがなにもしない.

   (b) INDEX の妥当性はこのルーチンでチェックされず, SCPMU, SCPMV を呼んだときにチェックされる.

   (c) 現在設定されているマーカーのラインインデクスは

   ```
   CALL SCQPMI(INDEX)
   ```

   によって参照できる.

3.5.5 SCSPMS

1. 機能

   SCPMU, SCPMV で描くマーカーの大きさを設定する.

2. 呼び出し方法

   ```
   CALL SCSPMS(RSIZE)
   ```

3. パラメーターの説明

   RSIZE (R) マーカーの大きさを R-座標系における単位で指定する (初期値は 0.01).

4. 備考

   (a) RSIZE は 0 より大きくなければならない. ただし RSIZE が 0 のとき, メッセージは出力されるがなにもしない.

   (b) RSIZE の妥当性はこのルーチンでチェックされず, SCPMU, SCPMV を呼んだときにチェックされる.

   (c) 現在設定されているマーカーの大きさは

   ```
   CALL SCQPMS(RSIZE)
   ```

   によって参照できる.

3.6 サブルーチンの説明： トーンプリミティブ

三形閉領域のぬりつぶし図形出力 (tone primitive) に関するサブルーチン群.

3 次元のトーンプリミティブは 2 次元のプリミティブと異なり, 三角形の塗りつぶしのみ可能である. これは, 基本的にトーンプリミティブはある 2 次元平面内の領域を塗りつぶすものであり, 3 次元空間内でこれを行なうには, 2 次元平面を特定しなければならないからである.

また, 面の表側が見えている場合と, 裏側が見えている場合があるので, トーンパターンは 2 種類指定する. ここで, 表側とは頂点を反時計回りに見る側である.

そのほかの点に関しては 2 次元ルーチンと同様である.

3.6.1 SCTNZU/SCTNZV

1. 機能

   U/V 座標系で三角形閉領域をぬりつぶす. トーンパターン番号も同時に指定する.

2. 呼び出し方法

```
CALL SCTNZU(UPX,UPY,UPZ,ITPAT1,ITPAT2)
CALL SCTNZV(VPX,VPY,VPZ,ITPAT1,ITPAT2)
```

3. パラメーターの説明

UPX, UPY, UPZ	(R)	長さ 3 の配列. 三角形を定義する頂点の U 座標系における (X, Y, Z) 座標の値を与える.
VPX, VPY, VPZ	(R)	長さ 3 の配列. 三角形を定義する頂点の U 座標系における (X, Y, Z) 座標の値を与える.
ITPAT1, ITPAT2	(I)	表と裏のトーンパターン番号. SCSTNP 参照.

4. 備考

   (a) ここで指定した ITPATn は, SCTNU, SCTNV が参照する ITPATn を変更しない.

3.6.2 SCTNU/SCTNV

1. 機能

   U/V 座標系で三角形閉領域をぬりつぶす.

2. 呼び出し方法

```
CALL SCTNU(UPX,UPY,UPZ)
CALL SCTNV(VPX,VPY,VPZ)
```

3. パラメーターの説明

UPX, UPY, UPZ (R) 長さ 3 の配列. 多角形を定義する頂点の U 座標系における (X, Y, Z) 座標の値を与える.

VPX, VPY, VPZ (R) 長さ 3 の配列. 多角形を定義する頂点の V 座標系における (X, Y, Z) 座標の値を与える.

4. 備考

(a) 領域をぬりつぶすトーンパターン番号は SCSTNP で設定する.

3.6.3 SCSTNP

1. 機能

SCTNU, SCTNV で用いるトーンパターン番号を設定する.

2. 呼び出し方法

```
CALL SCSTNP(ITPAT1,ITPAT2)
```

3. パラメーターの説明

ITPAT1, ITPAT2 (I) 表と裏のトーンパターン番号 (初期値は 1). 8.1.4- 8.1.10節参照.

4. 備考

(a) 表側とは, UPX, UPY, UPZ を反時計周りに見る側である.

(b) ITPATn は 1 以上でなければならない. ただし ITPATn が 0 のとき, メッセージは出力されるがなにもしない.

(c) ITPATn の妥当性はこのルーチンでチェックされず, SCTNU, SCTNV を呼んだときにチェックされる.

(d) 現在設定されているトーンパターン番号は

```
CALL SCQTNP(ITPAT1, ITPAT2)
```

によって参照できる.

第 4 章

SLPACK : レイアウトルーチン

4.1 概要

これは, 装置に依存した図形出力表示面を分割して, 1 画面のうちに互いに重なり合わない複数のワークステーションビューポートを設定できるようにしたサブルーチンパッケージである.

本パッケージは, 次のような主要サブルーチンからなる.

SLINIT ··· 初期化をおこなう.

SLSIZE ··· A4, B4 などの紙の大きさで作画範囲を設定する.

SLFORM ··· 縦横の実長で作画範囲を設定する.

SLDIV ··· 画面の分割をおこなう.

SLMGN ··· マージンをとる.

SLRAT ··· 縦横比を指定してマージンをとる.

SLSTTL ··· タイトルを設定する.

まず, SLINIT を呼ぶことにより最大作画範囲を設定する. ただし SLINIT は SGOPN (2.4.1参照) の中で呼ばれるので, ユーザーが直接呼ぶ必要はない. この作画可能な最大領域を第 1 レベル目のフレームという. SLPACK のほかのルーチンを呼ばなければ, このフレームがワークステーションビューポートとなる. さらに, SLSIZE, SLFORM によってこの第 1 レベルのフレームを具体的な実長を持った大きさで再設定することもできる.

次に, SLMGN でこのフレームにマージン (余白部分) をとることができる. マージンをとった内側が新しいフレームとなる. SLRAT によって, フレームの縦横比を指定してマージンをとることもできる. 第 1 レベルのフレームに限って, このマージン部分に文字列 (タイトル) を描くことができる. この文字列の設定は, SLSTTL によっておこなう.

1 ページの中に 2 つ以上のワークステーションビューポートを設定する場合, サブルーチン SLDIV を呼ぶことにより第 1 レベル目のフレームを分割する. このときワークステーションビューポートを設定してゆく順序および分割数を指定する. ここで, 分割された各部分を第 2 レベルのフレームという. 再び SLMGN または

SLRAT を呼ぶことによって, 第 2 レベルのフレームすべてについてマージンをとることができる. SLDIV による分割は 2 回まで可能である; すなわち第 3 レベルのフレームまで設定することができる. SGFRM (2.4.2 節参照) はこうやって設定したフレームをワークステーションビューポートと考えて, つぎつぎと作画領域を定義してゆき必要に応じて改ページの動作もおこなう.

基本的にこれらのルーチンは SGOPN と最初に呼ばれる SGFRM の間で呼ばなければならない. ただし SLSTTL は SGFRM のあとでも呼ぶことができて, これは次の改ページのタイミングで有効となる.

その他に, GRPH1 の管理する情報にしたがって作画境界を実線枠あるいはコーナーマークを使って描くサブルーチンがある.

SLPVPR ··· ビューポートの境界を描く.

SLPWWR ··· ワークステーションウインドウの境界を描く.

SLPWVR ··· 最大作画領域の境界を描く.

SLPVPC ··· ビューポートの 4 隅を描く.

SLPWWC ··· ワークステーションウインドウの 4 隅を描く.

SLPWVC ··· 最大作画領域の 4 隅を描く.

なお, SLPACK が用いる内部変数は SGpGET/SGpSET が管理している.

内部的には, SLBLK1 という名前の共通ブロックを使用している. ユーザーは同じ名前の外部手続き名・共通ブロック名を用いてはならない.

4.2 サブルーチンのリスト

主要ルーチン

SLINIT(WXMAX,WYMAX,FACT)	初期化.
SLSIZE(CSIZE)	第 1 フレームの再設定 (A4, B5 等).
SLFORM(DXA,DYA)	第 1 フレームの再設定 (実長).
SLDIV(CFORM,IX,IY)	フレームの分割.
SLMGN(XL,XR,YB,YT)	マージンの設定.
SLRAT(RX,RY)	縦横比の設定.
SLSTTL(CTTL,CSIDE,PX,PY,HT,NT)	タイトルの設定.

作画境界

SLPVPR(INDEX)	ビューポートの枠を描く.
SLPWWR(INDEX)	ワークステーションウインドウの枠を描く.
SLPWVR(INDEX)	最大作画領域の枠を描く.
SLPVPC(INDEX,RC)	ビューポートのコーナーマークを描く.
SLPWWC(INDEX,RC)	ワークステーションウインドウのコーナーマークを描く.
SLPWVC(INDEX,RC)	最大作画領域のコーナーマークを描く.

4.3 主要サブルーチンの説明

4.3.1 SLINIT

1. 機能

   レイアウトサブルーチンパッケージを初期化する. GRPH1 のもとでは SGOPN (実際には SGINIT; 2.4.1節参照) 内で呼ばれるので, ユーザーが直接呼ぶ必要はない.

2. 呼び出し方法

```
CALL SLINIT(WXMAX,WYMAX,FACT)
```

3. パラメーターの説明

WXMAX,WYMAX	(R)	最大作画範囲 (W 座標系の単位で与える).
FACT	(R)	W 座標系の単位長と実長との比を与える.

4. 備考

   (a) WXMAX>0, WYMAX>0, FACT>0 でなければならない.

   (b) このルーチンを呼ぶと第 1 レベル目のフレームとして最大作画範囲が割りあてられる. 第 1 レベル目のフレームは, SLSIZE, SLFORM によって変更できる.

4.3.2 SLSIZE

1. 機能

   A 系列および B 系列の紙の大きさで, 第 1 レベル目のフレームを再設定する.

2. 呼び出し方法

```
CALL SLSIZE(CSIZE)
```

3. パラメーターの説明

CSIZE	(C*3)	フレームの大きさと向きを表す. 最初の 2 文字が大きさ, 3 文字目が向きを表す. フレームの大きさは A 系列または B 系列 (A4,B5 など) で指定する. フレームの向きは, 縦長に割り付けるなら'T'ate または'P'ortrait, 横長に割り付けるなら'Y'oko または'L'andscape とする. また'A'uto とすると SLINIT で指定された最大作図範囲の縦・横比を見て, 横長なら'Y'oko, 縦長なら'T'ate として割り付ける. たとえば A4 の大きさで横長のフレームを設定するには'A4Y' と指定する.

4. 備考

(a) なし.

4.3.3 SLFORM

1. 機能

縦横の長さを実長で指定して, 第 1 レベル目のフレームを再設定する.

2. 呼び出し方法

```
CALL SLFORM(DXA,DYA)
```

3. パラメーターの説明

DXA, DYA	(R)	フレームの横および縦の実長. 単位は cm.

4. 備考

(a) なし.

4.3.4 SLDIV

1. 機能

フレームの分割をおこない, 次のレベルのフレームを定義する.

2. 呼び出し方法

```
CALL SLDIV(CFORM,IX,IY)
```

3. パラメーターの説明

CFORM	(C*1)	フレームを割り付ける方向を指定する. 縦方向に割り付けるなら, 'T'ate または'L'engthways, 横方向に割り付けるなら, 'Y'oko または'S'ideways, とする.
IX, IY	(I)	X 方向, Y 方向の分割数.

4. 備考

   (a) SLDIV は 2 回まで呼ぶことができる. すなわち最大のレベル数は 3 である.

   (b) 各レベルのフレームの総計は 40 である.

4.3.5 SLMGN

1. 機能

   今のレベルのフレームすべてについて指定したマージンをとる.

2. 呼び出し方法

```
CALL SLMGN(XL,XR,YB,YT)
```

3. パラメーターの説明

   XL,XR,YB,YT (R) それぞれ, 左辺, 右辺, 下辺, 上辺のマージン. 左辺, 右辺のマージンについては横方向の幅を 1 とする比率で, 下辺, 上辺のマージンについては縦方向の幅を 1 とする比率で比率で指定する.

4. 備考

   (a) $0 \leq$XL, XR, YB, YT≤ 1 かつ, XL+XR<1, YB+YT<1 でなければならない.

4.3.6 SLRAT

1. 機能

   今のレベルのフレームすべてについて, 縦横比を指定してフレームが最大内接するようにマージンをとる.

2. 呼び出し方法

```
CALL SLRAT(RX,RY)
```

3. パラメーターの説明

   RX, RY (R) フレームの縦横比を与える. たとえば RX=1.0, RY=1.0 とすると, 正方形のフレームが最大内接するようにマージンを取る.

4. 備考

   (a) RX, RY>0 でなければならない.

4.3.7 SLSTTL

1. 機能

   第 1 レベル目のトップマージンまたはボトムマージンに描く文字列を設定する.

2. 呼び出し方法

```
CALL SLSTTL(CTTL,CSIDE,PX,PY,HT,NT)
```

3. パラメーターの説明

CTTL (C*(*)) 描く文字列. 次の予約変数があってそのつど相当する文字列に置き換えられる.
'#PAGE' : ページ数. アラビア数字で 2 文字分のスペースをとる.
'#DATE' : プログラムを実行したときの日付. YY/MM/DD (YY:年, MM:月, DD:日) のフォーマットで表現する.
'#TIME' : プログラムを実行したときの時刻. HH:MM:SS (HH:時, MM:分, SS:秒) のフォーマットで表現する.

CSIDE (C*1) 文字列を描く場所. 'T'op (トップマージン) または'B'ottom(ボトムマージン) を指定する.

PX, PY (R) マージン内における文字列の位置を-1.0 から$+1.0$ までの実数値で指定する.
PX については,
-1.0 : 左よせ
0.0 : 中央あわせ
$+1.0$: 右よせ
PY については,
-1.0 : 下よせ
0.0 : 中央あわせ
$+1.0$: 上よせ

HT (R) 文字の高さ. 単位は R 座標系.

NT (I) 何番目の文字列かを指定する.

4. 備考

(a) このルーチンは文字列を設定するだけで, 実際に文字列を描くのは SLPTTL である. SLPTTL は SGFRM (2.4.2節参照) のなかで呼ばれるので, ユーザーは SLPTTL を直接呼ぶ必要はない. またこのとき同時に SLPCNR が最大作画領域の 4 隅にコーナーマークをつける. なお SLSTTL はほかの SGPACK のルーチンと違って, SGFRM のあとでも呼べる. こうすることによってページごとに異なるタイトルを書くことができる. SGpGET/SGpSET の管理する内部変数'LCORNER' を.FALSE. とするとコーナーマークを描かない (初期値は.TRUE.); また'LTITLE' を.FALSE. とするとタイトルを描かない (初期値は.TRUE.).

(b) 設定できる文字列の最大数は 5 である.

(c) 設定した文字列を無効にしたいときは

```
CALL SLDTTL(NT)
```

とすると第 NT 番目の設定が無効になる.

4.3.8 SLPVPR/SLPWWR/SLPWVR/SLPVRC/SLPWWC/SLPWVC

1. 機能

作画境界を実線枠あるいはコーナーマークを使って描く
SLPVPR : ビューポートの枠を描く.

SLPWWR : ワークステーションウインドウの枠を描く.
SLPWVR : 最大作画領域の枠を描く.
SLPVPC : ビューポートのコーナーマークを描く.
SLPWWC : ワークステーションウインドウのコーナーマークを描く.
SLPWVC : 最大作画領域のコーナーマークを描く.

2. 呼び出し方法

```
CALL SLPVPR(INDEX)
CALL SLPWWR(INDEX)
CALL SLPWVR(INDEX)
CALL SLPVPC(INDEX,RC)
CALL SLPWWC(INDEX,RC)
CALL SLPWVC(INDEX,RC)
```

3. パラメーターの説明

INDEX	(I)	線分のラインインデクス (1.7節参照; 初期値は 1).
RC	(R)	コーナーマークの長さ (単位は R 座標系).

4. 備考

(a) INDEX は 1 以上でなければならない. ただし INDEX が 0 のとき, メッセージは出力されるがなにもしない.

(b) RC は 0 より大きくなければならない. ただし RC が 0 のとき, メッセージは出力されるがなにもしない.

4.3.9 その他のサブルーチン

このパッケージにはこのほかに以下の下位ルーチンがある. ここではその名前をあげるにとどめる.

```
SLDIVZ  SLMGNZ  SLPAGE  SLPCNR  SLQRCT  SLQSIZ
SLRATZ  SLPTTL  SLTLCV  SLZTTL
```

第 5 章

SZPACK : 基本描画ルーチン

5.1 概要

SZPACK ではポリラインやテキストといった各図形要素をすべて折れ線とトーンに分解する. したがって SZPACK には, 折れ線とトーン描画の 2 系統の基本描画ルーチンがあり, その上位に SGPACK の各プリミティブに対応するインターフェイスルーチンがある.

折れ線とトーンの基本描画ルーチンは両方とも, オープンとクローズの間で必要な座標点を指定する構造になっており, 最上位の U 座標系のルーチンから最下位の R 座標系のルーチンまで, 座標変換やクリッピングの処理単位ごとのルーチンが用意されている.

上位のインターフェイスルーチンも, オープンとクローズのルーチンがあって, その間で呼ばれる描画ルーチン (基本描画ルーチンのオープンからクローズまでをひとまとめにしたもので, SGPACK の各プリミティブに対応する.) が用意されている. この上位インターフェイスのオープンのタイミングで, 各プリミティブの属性設定などの初期化が行なわれ, SWPACK の SWOOPN (オブジェクトのオープン) が呼ばれる. クローズのタイミングでは SWOCLS (オブジェクトのクローズ) が呼ばれる.

SGPACK の各ルーチンは, 基本的に 1 つのプリミティブ毎にオープンとクローズを行なうが, 同じ属性のプリミティブを多数描画する際には, このオープンとクローズの処理にかかる時間が無視できなくなる場合がある. そのような場合には, SZPACK の上位インターフェイスルーチンを使って, 1 回のオープンとクローズの間に, 複数のプリミティブを描画するようにすることで高速化が図れる.

5.2 サブルーチンのリスト

5.2.1 インターフェイスルーチン

インターフェイスルーチンは各プリミティブをオープン, 描画, クローズの 3 つの手続きに分割したものである. 複数の点の座標などを指定するポリラインなどは, 座標値を配列で指定する.

SZPLOP(ITYPE,INDEX)	ポリラインプリミティブの初期化.
SZPLZc(N,PX,PY)	ポリラインプリミティブの描画.
SZPLCL	ポリラインプリミティブの終了.
SZPMOP(ITYPE,INDEX,RSIZE)	ポリマーカープリミティブの初期化.
SZPMZc(N,PX,PY)	ポリマーカープリミティブの描画.
SZPMCL	ポリマーカープリミティブの終了.
SZTNOP(ITPAT)	トーンプリミティブの初期化.
SZTNZc(N,PX,PY)	トーンプリミティブの描画.
SZTNCL	トーンプリミティブの終了.
SZTXOP(RSIZE,IROTA,ICENT,INDEX)	テキストプリミティブの初期化.
SZTXZc(X,Y,CHARS)	テキストプリミティブの描画.
SZTXCL	テキストプリミティブの終了.
SZLNOP(INDEX)	ラインサブプリミティブの初期化.
SZLNZc(X1,Y1,X2,Y2)	ラインサブプリミティブの描画.
SZLNCL	ラインサブプリミティブの終了.
SZLAOP(ITYPE,INDEX)	アローサブプリミティブの初期化.
SZLAZc(X1,Y1,X2,Y2)	アローサブプリミティブの描画.
SZLACL	アローサブプリミティブの終了.

ここで c は描く座標系に応じて U: UC, V: VC, R: RC の 3 種類がある.

5.2.2 基本描画ルーチン

基本描画ルーチンには折れ線ルーチンとトーンルーチンがあり, それぞれ座標値を配列ではなく 1 点ずつ指定するようになっている. 折れ線やトーンの属性に関してはあらかじめ設定ルーチンで設定しておかなければならない.

折れ線ルーチン

	入力座標	処理内容
SZxxLU	U	回転/正規化変換, 線形補間
SZxxLT	T	地図投影, 大円補間, 地図境界でのクリッピング
SZxxLV	V	View Port でのクリッピング
SZxxLC	V	ラベルつき線分への展開
SZxxLD	V	点線の展開
SZxxLP	V	透視変換
SZxxLR	R	WS View Port でのクリッピング
SZxxLZ	R	WS 変換, 直線描画

ここで xx は動作に応じて OP: OPEN, MV: MOVE, PL: PLOT, CL: CLOSE の 4 種類がある.

これらのルーチンは, インターフェイスルーチンの SZPLZc(N,PX,PY) を展開したものと思えばよい. 基本ルーチンを使って折れ線を描くプログラムは次のようになる.

```
1         CALL SZPLOP(ITYPE,INDEX)
2           CALL SZOPLc
3             CALL SZMVLc(UXX,UYY)
4             CALL SZPLLc(UXX,UYY)
5               ........
6           CALL SZCLLc
7         CALL SZPLCL
```

SZPLOP ではラインタイプやラインインデクスの設定の他, 必要な情報を SGRGET などによって取得するので, 頻繁に呼ぶと動作速度に影響する場合がある.

トーンルーチン

	入力座標	処理内容
SZxxTU	U	回転/正規化変換, 線形補間
SZxxTT	T	地図投影, 大円補間, 地図境界でのクリッピング
SZxxTV	V	View Port でのクリッピング
SZxxTP	V	投影変換, soft/hard fill の switching
SZxxTR	R	WS View Port でのクリッピング
SZxxTZ	R	WS 変換, バッファリング
SZxxTS	V	バッファリング (ソフトフィル)
SZTNSV	V	ソフトフィル

これらのルーチンは, インターフェイスルーチンの SZTNZc(N,PX,PY) を展開したものと思えばよい. 基本ルーチンを使ってトーンをかけるプログラムは次のようになる.

```
1         CALL SZTNOP(ITPAT)
2           CALL SZOPTc
3             CALL SZSTTc(UXX,UYY)
4               ........
5           CALL SZCLTc
6         CALL SZTNCL
```

SZTNOP ではトーンパターンの設定, および境界線の向きに関する処理などを行う他, 必要な情報を SGRGET などによって取得するので, 頻繁に呼ぶと動作速度に影響する場合がある.

第 6 章

STPACK : 変換関数

6.1 概要

STPACK は座標系の間の変換を受け持つ関数を集めたものである. 座標系の変換は,

- 正規化変換
- 透視変換
- ワークステーシ ョン変換

の 3 段階に分けられる. 正規化変換はさらに次の変換を含む.

- 角度の単位変換 (ラジアンと度)
- 地図座標の回転
- 基本変換

ここで基本変換とは, 正規化変換の本質的な部分で, 直角座標系では正規化変換そのものであり, 地図投影座標では座標系回転後の変換 (math1/maplib による変換) である. また, 基本変換では角度は常にラジアンで扱われる.

6.1.1 ユーザー定義関数

以下のエントリをユーザー自身が用意することで, 独自の変換関数を定義することができる.

`STFUSR(UX,UY,XX,YY)`	正変換
`STIUSR(XX,XY,UX,UY)`	逆変換
`STSUSR`	関数初期化

この正変換と逆変換のエントリは, `STFTRF` の下請けルーチンである `STFTRN` から呼ばれる. `STFUSR` に渡ってくる `UX`, `UY` の値は, `STFRAD` と `STFROT` による処理が終った後の値である. 拡大, 並行移動は `STFTRN` が行なうので, これらの関数は, それ以外の基本的な変換関数を定義するだけで良い. 変換関数には正変換と逆変換があるが, 逆変換を使わない場合には, 逆変換はエントリー文だけで良い.

STSUSR は, 変換関数を確定する SGSTRF から呼ばれる. このエントリは, 通常 SGSTRF が行なっている STPACK の関数を初期化しなければならない. 初期化すべき関数は, STSTRF, STSTRN, STSRAD, STSROT (地図投影の時) である.

次のプログラムは常用対数の log-log 座標を定義するものである.

```
*-----------------------------------------------------------------------
*     USER SUPPLIED FUNCTION
*-----------------------------------------------------------------------
      SUBROUTINE STFUSR(UX, UY, XX, YY)

      XX = LOG(UX)
      YY = LOG(UY)

      RETURN
*-----------------------------------------------------------------------
      ENTRY STIUSR(XX, YY, UX, UY)

      UX = EXP(XX)
      UY = EXP(YY)

      RETURN
      END
*-----------------------------------------------------------------------
      SUBROUTINE STSUSR

      CALL SGQVPT(VXMIN, VXMAX, VYMIN, VYMAX)
      CALL SGQWND(UXMIN, UXMAX, UYMIN, UYMAX)

      CX = (VXMAX-VXMIN)/LOG(UXMAX/UXMIN)
      CY = (VYMAX-VYMIN)/LOG(UYMAX/UYMIN)

      VX0 = VXMIN - CX*LOG(UXMIN)
      VY0 = VYMIN - CY*LOG(UYMIN)

      CALL STSTRF(.FALSE.)
      CALL STSRAD(.FALSE., .FALSE.)
      CALL STSTRN(99, CX, CY, VX0, VY0)

      END

```

次のプログラムは MATH1/MAPLIB の MPFMWL/MPIMWL を使ってモルワイデもどきの地図投影関数を定義するものである.

```
*-----------------------------------------------------------------------
*     USER SUPPLIED FUNCTION
*-----------------------------------------------------------------------
      SUBROUTINE STFUSR(UX, UY, XX, YY)

      CALL MPFMWL(UX, UY, XX, YY)

      RETURN
*-----------------------------------------------------------------------
      ENTRY STIUSR(XX, YY, UX, UY)

      CALL MPIMWL(XX, YY, UX, UY)

      RETURN
      END
*-----------------------------------------------------------------------
      SUBROUTINE STSUSR
      LOGICAL LDEG

      CALL SGQVPT(VXMIN, VXMAX, VYMIN, VYMAX)
      CALL SGQSIM(SIMFAC, XOFF, YOFF)
      CALL SGQMPL(PLX, PLY, PLROT)
      CALL SGLGET('LDEG', LDEG)

      IF(LDEG) THEN
        CP = RFPI()/180
      ELSE
        CP = 1
      ENDIF
```

```
30
31          CALL SGRGET('TXMIN', TXMIN)
32          CALL SGRGET('TXMAX', TXMAX)
33          CALL SGRGET('TYMIN', TYMIN)
34          CALL SGRGET('TYMAX', TYMAX)
35          CALL SZSCLX(CP*TXMIN, CP*TXMAX)
36          CALL SZSCLY(CP*TYMIN, CP*TYMAX)
37
38          VX0 = (VXMAX+VXMIN)/2 + XOFF
39          VY0 = (VYMAX+VYMIN)/2 + YOFF
40
41          CALL STSTRF(.TRUE.)
42          CALL STSRAD(LDEG, LDEG)
43          CALL STSROT(RFPI()/2-CP*PLY, CP*PLX, CP*PLROT)
44          CALL STSTRN(99, SIMFAC, SIMFAC, VX0, VY0)
45
46          END
```

6.2 サブルーチンのリスト

座標変換

STFTRF(UX,UY,VX,VY)	正規化変換
STFTR3(UX,UY,UZ,VX,VY,VZ)	3 次元正規化変換
STFPR2(VX,VY,RX,RY)	二次元透視変換.
STFPR3(VX,VY,VZ,RX,RY)	三次元透視変換.
STFWTR(RX,RY,WX,WY)	ワークステーション変換.
STSTRF(LMAP)	正規化変換パラメタ設定.
STSPR2(IX,IY,SECT)	二次元透視変換パラメタ設定.
STSPR3(XFC,YFC,ZFC,THETA,PHI,PSI,FAC,ZVIEW,RXOFF,RYOFF)	三次元透視変換パラメタ設定.
STSWTR (RXMIN,RXMAX,RYMIN,RYMAX,WXMIN,WXMAX,WYMIN,WYMAX,IWTRF)	ワークステーション変換パラメタ設定.

この他に, ルーチン名の 3 文字めが I の逆変換, Q のパラメタ問い合わせルーチンがある.

下請けルーチン

STFTRN(UX,UY,VX,VY)	基本変換.
STSTRN(ITR,CXA,CYA,VX0,VY0)	基本変換パラメタ設定.
STFROT(UX,UY,TX,TY)	地図座標の回転.
STSROT(THETA,PHI,PSI)	地図座標の回転パラメタ設定.
STFRAD(UX1,UY1,UX2,UY2)	角度の単位変換.
STSRAD(LXDEG,LYDEG)	角度の単位変換パラメタ設定.

この他に, ルーチン名の 3 文字めが I の逆変換, Q のパラメタ問い合わせルーチンがある.

6.3 サブルーチンの説明 : 座標変換

6.3.1 STFTRF/STITRF/STSTRF/STQTRF

1. 機能

   正規化変換 (UC と VC の変換) を行なう.

2. 呼び出し方法

```
CALL STFTRF(UX,UY,VX,VY)
CALL STITRF(VX,VY,UX,UY)
CALL STSTRF(LMAP)
CALL STQTRF(LMAP)
```

3. 引数の説明

UX,UY (R) UC の 座標
VX,VY (R) VC の 座標
LMAP (L) 地図投影関数であることを示すフラグ.

4. 備考

(a) この関数は, その中で STFRAD, STFROT, STFTRN を呼ぶ.
(b) LFLAG=.TRUE. の時, STFROT で座標回転を行なってから, STFTRN を呼ぶ.
(c) 地図投影の際に特異点などで変換不能の場合は, GLpGET/GLpSET が管理する内部変数 RUNDEF の示す値 (初期値は -999.0) を返す.

6.3.2 STFPR2/STIPR2/STSPR2

1. 機能

二次元透視変換を行なう.

2. 呼び出し方法

```
CALL STFPR2(VX,VY,RX,RY)
CALL STIPR2(RX,RY,VX,VY)
CALL STSPR2(IX,IY,SECT)
```

3. 引数の説明

VX,VY (R) VC の座標
RX,RY (R) RC の座標
IX,IY (I) 2 次元の X 軸と Y 軸を, 3 次元のどの軸に対応させるか, 番号で指定する. (1:X 軸, 2:Y 軸, 3:Z 軸) 負の値を指定すると, 逆向きに対応させる.
SECT (R) 2 次元平面と交わる座標軸の交点座標.

4. 備考

(a) 透視変換のパラメタは三次元透視変換と共通である.

6.3.3 STFTR3

1. 機能

3 次元正規化変換 (UC3 と VC3 の変換) を行なう.

2. 呼び出し方法

```
CALL STFTR3(UX,UY,UZ,VX,VY,VZ)
```

3. 引数の説明

UX,UY,UZ (R) UC の 座標
VX,VY,VZ (R) VC の 座標

4. 備考

なし.

6.3.4 STFPR3/STIPR3/STSPR3

1. 機能

三次元透視変換を行なう.

2. 呼び出し方法

```
CALL STFPR3(UX,UY,UZ,RX,RY)
CALL STSPR3(XFC,YFC,ZFC,THETA,PHI,PSI,FAC,ZVIEW,RXOFF,RYOFF)
```

3. 引数の説明

UX,UY,UZ	(R)	UC の座標
RX,RY	(R)	RC の座標
ITR3	(I)	変換関数番号
XFC,YFC,ZFC	(R)	焦点中心の座標
THETA,PHI,PSI	(R)	回転角
FAC	(R)	拡大縮小ファクター
ZVIEW	(R)	視点の Z 座標
RXOFF,RYOFF	(R)	オフセット

4. 備考

(a) この関数は以下の操作をする.

- 焦点中心を原点に並行移動する.
- 原点を中心に (THETA,PHI,PSI) 回転する.
- 座標値を FAC 倍する.
- (0,0,ZVIEW) を視点として, X–Y 平面に投影する.
- 投影された座標値に (RXOFF,RYOFF) を加える.

6.3.5 STFWTR/STWTR/STSWTR/STQWTR

1. 機能

ワークステーション変換を行なう.

2. 呼び出し方法

```
CALL STFWTR(RX,RY,WX,WY)
CALL STIWTR(WX,WY,RX,RY)
CALL STSWTR(RXMIN,RXMAX,RYMIN,RYMAX,WXMIN,WXMAX,WYMIN,WYMAX,IWTRF)
CALL STQWTR(RXMIN,RXMAX,RYMIN,RYMAX,WXMIN,WXMAX,WYMIN,WYMAX,IWTRF)
```

3. 引数の説明

RX,RY	(R)	RC の座標
WX,WY	(R)	WC の座標
RXMIN,RXMAX,RYMIN,RYMAX	(R)	ワークステーションウインドウ.
WXMIN,WXMAX,WYMIN,WYMAX	(R)	ワークステーションビューポート.
IWTRF	(I)	変換番号.

4. 備考

   (a) なし.

6.4 サブルーチンの説明 : 座標変換 (下請け)

以下の関数は, STFTRF/STITRF から呼ばれるルーチンである. これらの関数の初期化は, 通常 SGSTRF によって行なわれる. ユーザー定義関数を使う場合は, ユーザー自身がこれらのルーチンを初期化する必要がある.

6.4.1 STFTRN/STITRN/STSTRN

1. 機能

   正規化変換 (UC と VC の変換) または地図投影 (TC と VC の変換) を行なう.

2. 呼び出し方法

```
CALL STFTRN(UX,UY,VX,VY)
CALL STITRN(VX,VY,UX,UY)
CALL STSTRN(ITR,CXA,CYA,VXO,VYO)
```

3. 引数の説明

UX,UY	(R)	UC/TC の 座標
VX,VY	(R)	VC の 座標
ITR	(I)	変換関数番号
CXA,CYA	(R)	座標値にかけるファクター
VXO,VYO	(R)	VC における原点の位置

4. 備考

   (a) 地図投影の際に特異点などで変換不能の場合は, GLpGET/GLpSET が管理する内部変数 RUNDEF の示す値 (初期値は -999.0) を返す.

   (b) この関数は以下の操作をする.

   - 変換数番号にしたがって, (UX,UY) を基本的な変換をして (XX,YY) とする.
   - さらに次式により, 拡大, 移動を行なう.

```
VX = CXA*XX + VXO
VY = CYA*XX + VYO
```

6.4.2 STFROT/STIROT/STSROT

1. 機能

   地図座標の回転を行なう.

2. 呼び出し方法

```
CALL STFROT(UX,UY,TX,TY)
CALL STIROT(TX,TY,UX,UY)
CALL STSROT(THETA,PHI,PSI)
```

3. 引数の説明

UX,UY	(R)	UC の座標
TX,TY	(R)	TC の座標
THETA,PHI,PSI	(R)	回転角

4. 備考

   (a) なし.

6.4.3 STFRAD/STIRAD/STSRAD

1. 機能

   角度の単位変換を行なう.

2. 呼び出し方法

```
CALL STFRAD(UX1,UY1,UX2,UY2)
CALL STIRAD(UX2,UY2,UX1,UY1)
CALL STSRAD(LXDEG,LYDEG)
```

3. 引数の説明

UX1,UY1	(R)	UC の座標
UX2,UY2	(R)	UC の座標 (常にラジアン)
LXDEG,LYDEG	(L)	UX1,UY1 の単位が, 度」であることを示すフラグ.

4. 備考

   (a) なし.

第 7 章

SWPACK : 機種依存ルーチン

7.1 概要

SWPACK は GRPH1 のライブラリの中の機械依存ルーチンをまとめたもので, 最下位に位置するものである. SWPACK では, 上位インターフェイスの規格のみ定められており, 実際の動作や内部の構造は各種のデバイスによって異なる.

SWPACK の機能には 「基本機能」と 「拡張機能」があり, 基本機能はすべてのデバイスにおいて実現される (と仮定している) が, 拡張機能はデバイスによって実現できないものもある.

- 基本機能
  - 線画機能 (線の太さか色のどちらかが変えられること)
- 拡張機能
  - ハードフィル機能 (ハードウェアによる網かけ)
  - イメージ表示機能 (ドット単位の描画)
  - ポインティング機能 (マウスなど)

上位ルーチンは拡張機能が使えない場合があることを想定して, コーディングされなければならない. これらの機能が使えるかどうかは, SWPACK の問い合わせルーチンを通して取得できる.

GRPH1 のレベルでは, 線の太さと色はデバイスの能力によって切替えられる. また, ハードフィルのできないデバイスでは, ソフトフィルによる網かけに自動的に変更する.

イメージ表示機能とポインティング機能に関しては代替処置がないので, これらの機能を使う時には注意が必要である.

なお, SWPACK では機種依存内部変数を SWpGET/SWpSET(整数・実数・論理型) および SWCGET/SWCSET (文字型) によって参照/変更することができる. ただし, SWpGET/SWpSET および SWCGET/SWCSET の管理する内部変数は, デバイスのオープン時にすべて読み込まれて, その後は参照されないので, SWPSET および SWCSET は SGOPN の前に呼ばなければならない.

7.8節には, 電脳ライブラリが標準的に提供する図形出力装置 (X サーバー, ポストスクリプト, テクトロ端末) に対する内部変数に関する説明がある. どのような内部変数が定義されているかはシステムに依存する.

7.2 サブルーチンのリスト

コントロールルーチン

SWDOPN	デバイスのオープン.
SWDCLS	デバイスのクローズ.
SWPOPN	ページのオープン.
SWPCLS	ページのクローズ.
SWOOPN(COBJ,COM)	オブジェクトのオープン.
SWOCLS(COBJ)	オブジェクトのクローズ.
SWpGET(CP,IPARA)	内部変数参照.
SWpSET(CP,IPARA)	内部変数設定.
SWpSTX(CP,IPARA)	内部変数の設定 (実行時オプションによる変更を許す).
SWCGET(CP,CPARA)	文字型内部変数参照.
SWCSET(CP,CPARA)	文字型内部変数設定.
SWCSTX(CP,CPARA)	文字型内部変数の設定 (実行時オプションによる変更を許す).

描画ルーチン

SWSWDI(IWDIDX)	線幅指定.
SWSCLI(ICLIDX)	線色指定.
SWGOPN	プリミティブのオープン.
SWGMOV(WX,WY)	ペンアップ MOVE.
SWGPLT(WX,WY)	ペンダウン MOVE.
SWGCLS	プリミティブのクローズ.
SWGTON(NP,WPX,WPY,ITPAT)	ハードフィル.
SWIOPN(IWX,IWY,IMW,IMH)	イメージオープン.
SWIDAT(IMAGE,NLEN)	イメージデータ.
SWICLS	イメージクローズ.

マウス

SWQPNT(WX,WY,MB)	マウスポイント取得.

座標変換

SWFINT(WX,WY,IWX,IWY)	ワークステーション座標からイメージ座標への変換.
SWIINT(IWX,IWY,WX,WY)	イメージ座標からワークステーション座標への変換.

問い合わせなど

SWQWDC(LWD)	線幅可変能力の取得.
SWQCLC(LCL)	線色可変能力の取得.
SWQTNC(LTN)	HARD FILL 能力の取得.
SWQIMC(LIM)	イメージ描画能力の取得.
SWQPTC(LPT)	ポインティング能力の取得.
SWQFNM(CPARA,CFNAME)	各種データベースファイル名の取得.
SWQRCT(XMIN,XMAX,YMIN,YMAX,UNIT)	最大作画領域取得.
SWSROT(IWTROT)	画面の回転指定.

7.3 サブルーチンの説明 : コントロールルーチン

7.3.1 SWDOPN/SWDCLS

1. 機能

   デバイスをオープン/クローズする.

2. 呼び出し方法

```
CALL SWDOPN
CALL SWDCLS
```

3. 引数の説明

   なし.

4. 備考

   (a) SW*p*SET(SW*p*STX), SWCSET(SWCSTX) をのぞくすべての SWPACK ルーチンはこの 2 つのルーチンの間で呼ばれる.

   (b) 座標系の回転は STPACK が行う.

   (c) SWDOPN を呼ぶ前に SW*p*GET/SW*p*SET が管理する内部変数'IWS' を設定しておかなければならない. この設定はふつう SGOPN の中でおこなわれる.

7.3.2 SWPOPN/SWPCLS

1. 機能

   ページををオープン/クローズする.

2. 呼び出し方法

```
CALL SWPOPN
CALL SWPCLS
```

3. 引数の説明

   なし.

4. 備考

   (a) CALCOMP 系プロットルーチンの様にページの概念のない処理系では, 原点移動などにより仮想的にページの概念が導入される.

   (b) ページのクローズとオープンが"改ページ"として常にセットで行われる様な処理系では, どちらかのルーチンが DUMMY になる場合もある.

7.3.3 SWOOPN/SWOCLS

1. 機能

   オブジェクトをオープン/クローズする.

2. 呼び出し方法

```
CALL SWOOPN(COBJ,COMM)
CALL SWOCLS(COBJ)
```

3. 引数の説明

COBJ	(C*(*))	オブジェクト名.
COMM	(C*(*))	コメント.

4. 備考

   (a) オブジェクトとは, 同じ属性を持つ一まとまりの図形要素である.

   (b) これらのルーチンは, SZPACK の上位インターフェイスルーチンのオープン/クローズのタイミングで呼ばれる.

   (c) オブジェクトの概念のないデバイスではダミールーチンとなる.

7.4 サブルーチンの説明： 描画ルーチン

7.4.1 SWSWDI/SWSCLI

1. 機能

   描く線分の属性 (線幅と色) を指定する.

2. 呼び出し方法

```
CALL SWSWDI(IWDIDX)
CALL SWSCLI(ICLIDX)
```

3. 引数の説明

IWDIDX	(I)	線分の太さに関するインデクス. SGPACK (1.7.1節) 参照.
ICLIDX	(I)	線分の色に関するインデクス. SGPACK (1.7.1節) 参照.

4. 備考

   (a) これらのルーチンは実際に描画するルーチンの前に呼ばなければならない.

   (b) IWDIDX, ICLIDX が 0 の時は, 1 と見なして処理する.

   (c) GRPH1 で指定できる線幅は 0 から 9 までであるが, 色が指定できないデバイスの場合には, 線色のインデクス ICLIDX を線幅のインデクス IWDIDX に読み変えて SWPACK に渡す場合があるので, このようなシステムにおいては IWDIDX が 99 までの値をとる.

   (d) 実際に線がどのような色になるか, また, どの程度の太さになるかは, システムに依存する.

7.4.2 SWGOPN/SWGCLS

1. 機能

   線分をオープン/クローズする.

2. 呼び出し方法

```
CALL SWGOPN
CALL SWGCLS
```

3. 引数の説明

   なし.

4. 備考

   (a) CALCOMP 系の処理系では, どちらも DUMMY ルーチンとなる.

7.4.3 SWGMOV/SWGPLT

1. 機能

   ペンアップ/ペンダウンで移動する.

2. 呼び出し方法

```
CALL SWGMOV(WX,WY)
CALL SWGPLT(WX,WY)
```

3. 引数の説明

   WX,WY (R) ワークステーション座標. (i)

4. 備考

   (a) プリミティブの中にこの 2 つのルーチンは任意の回数出現するが, プリミティブがオープンされた直後に呼ばれるルーチンは SWGMOV でなければならない. SWGOPN の直後に SWGPLT を呼んだ場合の動作は保証されない.

7.4.4 SWGTON

1. 機能

   ハードフィルを行う.

2. 呼び出し方法

```
CALL SWGTON(NP,WPX,WPY,ITPAT)
```

3. 引数の説明

NP	(I)	配列 WPX,WPY の大きさ. (i)
WPX	(R)	塗りつぶす領域の境界の X 座標. (i)
WPY	(R)	塗りつぶす領域の境界の Y 座標. (i)
IPAT	(I)	塗りつぶしパターン番号. (i)

4. 備考

(a) パターン番号は 5 桁の番号で, 上位 2 桁が色, 3 桁目がパターンの種類 (点または線), 4 桁目が点または線の太さ, 最下位桁が点または線の密度に対応する.

7.4.5 SWIOPN/SWIDAT/SWICLS

1. 機能

イメージデータを描画する.

2. 呼び出し方法

```
CALL SWIOPN(IWX,IWY,IMW,IMH)
CALL SWIDAT(IMAGE,NLEN)
CALL SWICLS
```

3. 引数の説明

IWX,IWY	(I)	イメージの左上の座標. (i)
IMW,IMH	(I)	イメージの大きさ. (i)
IMAGE	(I)	イメージデータ. (i)
NLEN	(I)	イメージデータの長さ. (i)

4. 備考

(a) イメージの位置, および大きさはイメージ座標で指定する.

(b) イメージデータは色番号 (パレット番号) を 4 バイトの整数で指定する.

(c) SWIOPN と SWICLS の間で, SWIDAT を複数回呼ぶことができる. すなわち, データを分割して指定することができる.

(d) 一回に指定するデータの長さは任意であるが, SWICLS が呼ばれるまでに SWIOPN で指定したイメージの大きさ分だけのデータを指定しなければならない. それより, 多いか少い場合の動作は不定である.

7.5 サブルーチンの説明： マウス

7.5.1 SWQPNT

1. 機能

マウスポイントの位置を返す.

2. 呼び出し方法

```
CALL SWQPNT(WX,WY,MB)
```

3. 引数の説明

WX,WY	(R)	マウスポイントの位置. (o)
MB	(I)	ボタンの種類. (o)

4. 備考

(a) X ウインドウシステムにおいて MB としては次のような値を返す. Button1 が押されたら 1, Button2 が押されたら 2, Button3 が押されたら 3, その他の場合は 0 を返す.

7.6 サブルーチンの説明 : 座標変換

7.6.1 SWFINT/SWIINT

1. 機能

ワークステーション座標とイメージ座標の変換をする.

2. 呼び出し方法

```
CALL SWFINT(WX,WY,IWX,IWY)
CALL SWIINT(IWX,IWY,WX,WY)
```

3. 引数の説明

WX,WY	(R)	ワークステーション座標. (i/o)
IWX,IWY	(I)	イメージ座標. (i/o)

4. 備考

(a) なし.

7.7 サブルーチンの説明 : 問い合わせルーチン

7.7.1 SWQWDC/SWQCLC/SWQTNC/SWQIMC/SWQPTC

1. 機能

SWPACK のルーチンが持つ能力を問い合わせる.

2. 呼び出し方法

```
CALL SWQWDC(LWD)
CALL SWQCLC(LCL)
CALL SWQTNC(LTN)
CALL SWQIMC(LIM)
CALL SWQPTC(LPT)
```

3. 引数の説明

LWD (L) 線幅を変える能力がある時.TRUE. (o)
LCL (L) 線色を変える能力がある時.TRUE. (o)
LTN (L) ハードフィルを行う能力がある時.TRUE. (o)
LIM (L) イメージ描画を行う能力がある時.TRUE. (o)
LIM (L) マウスポイントの位置を調べる能力がある時.TRUE. (o)

4. 備考

(a) 線幅または線色が変化できない機種では, 特に指定されない限り上位ルーチンが, 線分属性の情報をその機種で区別可能な属性に変換する.

(b) ハードフィルができない機種では, 上位ルーチンがソフトフィルを行う.

7.7.2 SWQFNM

1. 機能

各種データベースファイル名を得る.

2. 呼び出し方法

```
CALL SWQFNM(CPARA,CFNAME)
```

3. 引数の説明

CPARA (C*(*)) データベース名を示す内部変数の名前 (7.8節参照).
CFNAME (C*(*)) データベースファイル名.

4. 備考

(a) どのようなルールでファイル名を探すかはシステムに依存する.

(b) 標準ライブラリではまずカレントディレクトリを探す. 次に GLCGET/GLCSET が管理する内部変数'DUPATH' のさすディレクトリを探す. 最後に内部変数'DSPATH' のさすディレクトリを探す.

7.7.3 SWQRCT

1. 機能

最大作画領域を通知する.

2. 呼び出し方法

```
CALL SWQRCT(WXMIN,WXMAX,WYMIN,WYMAX,FACT)
```

3. 引数の説明

WXMIN (R) 作画領域の左下の X 座標. (o)
WXMAX (R) 作画領域の右上の X 座標. (o)
WYMIN (R) 作画領域の左下の Y 座標. (o)
WYMAX (R) 作画領域の右上の Y 座標. (o)
FACT (R) 座標系の 1 単位の長さ (cm). (o)

4. 備考

   (a) ディスプレイの様な大きさのはっきりしないデバイスに対しては, 描画領域の大きさがほぼ A4 程度の紙の大きさであると考えて FACT を設定する.

7.7.4 SWSROT

1. 機能

   画面の回転方向を指定する.

2. 呼び出し方法

```
CALL SWSROT(IWTROT)
```

3. 引数の説明

   IWTROT (I) 1 のとき正立; 2 のとき 90 度回転していることをあらわす. (i)

4. 備考

   (a) 通常 SWPACK は上位ルーチンがどのような向きに図形を描画しているか関知しないが, ハードフィルの際には図形の向きとデバイスの向きの関係が問題になるので, それを通知するためのルーチンである. したがって, この情報はハードフィルルーチンのみが使用する.

7.8 サブルーチンの説明 : 内部変数管理ルーチン

7.8.1 SWpGET/SWpSET(SWpSTX)/SWCGET/SWCSET(SWCSTX)

1. 機能

   SWpGET/SWpSET(SWpSTX) は整数 実数 論理型の内部変数について, SWCGET/SWCSET(SWCSTX) は文字型の内部変数について参照/変更する. (SWpSTX/SWCSTX は実行時オプションによる変更を許す.)

2. 呼び出し方法

```
CALL SWpGET(CP,IPARA)
CALL SWpSET(CP,IPARA)
CALL SWpSTX(CP,IPARA)
CALL SWCGET(CP,CPARA)
CALL SWCSET(CP,CPARA)
CALL SWCSTX(CP,CPARA)
```

3. 引数の説明

   CP (C*8) 内部変数の名前.
   IPARA (I,R,L) (整数 ・実数 ・論理型の) 内部変数の値.
   CPARA (C*(*)) (文字型の) 内部変数の値.

以下に, 電脳ライブラリが標準的に提供する図形出力装置 (X サーバー, ポストスクリプト, テクトロ端末) に対し, CP として指定できる名前のリストを記す. 値が整数 実数 論理型のものは SWpGET/SWpSET(SWpSTX) で指定する; 値が文字型のものは SWCGET/SWCSET(SWCSTX) で指定する.

'FONT1', 'CL2TN' などデータベースファイル名を示す内部変数は, その内部変数の値がファイル名としてそのまま利用されるとは限らない. 実際に利用されるファイル名は SWQFNM が決める. その時のヒントとしてこれらのパラメータ名は定義されていなければならない. 標準ライブラリではまずカレントディレクトリを探す. 次に GLCGET/GLCSET が管理する内部変数'DUPATH'(ユーザーの指定するパス名) のさすディレクトリを探す. 最後に内部変数 'DSPATH'(システムがあらかじめ用意するパス名) のさすディレクトリを探す.

また, 出力装置に依存して.x11, .psx などのサフィックスのついたものを優先的に探したりする. たとえば, 標準ライブラリにおいて 'CLRMAP' の示すファイル名を探すときに, 出力装置として X サーバーを選んでいれば, まず .x11 のサフィックスのついたものが探される; 次にサフィックスなしのものが探される; 最後に .psx のサフィックスのついたものが探される. いっぽう, 出力装置として ポストスクリプトを選んでいれば, まず .psx のサフィックスのついたものが探される; 次にサフィックスなしのものが探される; 最後に .x11 のサフィックスのついたものが探される.

[すべての出力装置に共通なパラメーター] - SWPACK がどのような出力装置を実現していても, これらの内部変数名は定義されていなければならない.

'MAXWNU' (I) SWPACK が実現している出力装置の数. 標準ライブラリにおける値は 3 (X, PS, Tek) である.

'IWS' (I) オープンする装置番号. 1 から'MAXWNU' の間の値を指定する (初期値は 1). この番号は SGOPN が設定する.

'WSNAMEnn'(C) ワークステーション名 ($1 \leq nn \leq$'MAXWNU'). 標準ライブラリにおける値は, 'WSNAME01'='X' (X サーバー), 'WSNAME02'='PS' (ポストスクリプト), 'WSNAME03'='Tek' (テクトロ端末) となっている.

'FONTn' (C) フォントファイル名 ($n = 1, 2$). 標準ライブラリにおける値は, 'FONT1'='font1u', 'FONT2'='font2u',

'CL2TN' (C) カラー番号をトーンパターン番号に読みかえるためのファイル名. 標準ライブラリにおける値は'cl2tnmap'.

[X サーバー (X) およびポストスクリプト (PS) に共通なパラメーター]

'IWIDTH' (I) イメージウインドウの横方向のサイズ (省略値は 900).

'IHEIGHT' (I) イメージウインドウの縦方向のサイズ (省略値は 650). X では, 'IWIDTH', × 'IHEIGHT' のウインドウがオープンされる. PS では, 縦横比が 'IHEIGHT':'IWIDTH' の描画領域が最大描画領域に最大内接するように用意される.

'FNAME' (C) 出力ファイル名 (省略値は'dcl'). この値が'*' あるいは空白のとき, 現在実行中のプログラム名が用いられる. X のダンプファイル名, PS の複数ページ出力のためのファイル名として用いられるとき, この名前に'_nnn' (nnn はページ番号) を付けたものとなる. さらに X のダンプファイル名の場合は'.xwd' のサフィックスがつく; PS ファイル名の場合は'.ps' のサフィックスがつく.

'TITLE' (C) タイトル名 (省略値は'*'). この値が'*' あるいは空白のとき, DCLVNM (MISC1/MISCLIB 参照) が返すバージョン名となる. X の場合, このタイトル名はウインドウのタイトル名として用いられる; PS の場合, このタイトル名はコメント部分の Creator 部分に書かれる.

'CLRMAP' (C) カラーマップを収めたファイル名. 標準ライブラリにおける値は 'colormap'.

[X サーバー (X) に関するパラメーター]

'IPOSX' (I) ウインドウの位置を示す X 座標. (省略値は-999).

'IPOSY' (I) ウインドウの位置を示す Y 座標. (省略値は-999). (IPOSX,IPOSY)=$(-999, -999)$ のときユーザーがマウスで指定する.

'LWAIT' (L) 改ページのタイミングで一時停止するかどうかを指定する. .TRUE. なら一時停止する; .FALSE. なら一時停止しない (省略値は .TRUE.).

'LWAIT0' (L) デバイスをオープンするタイミングで一時停止するかどうかを指定する. .TRUE. なら一時停止する; .FALSE. なら一時停止しない (省略値は .FALSE.).

'LWAIT1' (L) デバイスをクローズするタイミングで一時停止するかどうかを指定する. .TRUE. なら一時停止する; .FALSE. なら一時停止しない (省略値は .TRUE.).

'LDUMP' (L) X のダンプファイル (xwd) を作成するかどうか指定する. .TRUE. なら作成する; .FALSE. なら作成しない (省略値は .FALSE.).

'LALT' (L) 裏画面で描画するかどうかを指定する. .TRUE. なら裏画面で描画する; .FALSE. なら裏画面で描画しない (省略値は .FALSE.).

'LKEY' (L) 一時停止しているときに, キーボードからの指定を有効にするかどうかを指定する. .TRUE. なら有効にする; .FALSE. なら有効にしない (省略値は .TRUE.).

SP, RETURN : 次ページに進む.
s : 'LWAIT' を.FALSE. にする.
w : 'LWAIT' を.TRUE. にする.
d : ダンプファイルを作成する.
q : 実行を中止する.

'BITMAP' (C) ビットマップを収めたファイル名. 標準ライブラリにおける値は'bitmap'.

[ポストスクリプト (PS) に関するパラメーター]

'LCOROL' (L) カラー対応した PS ファイルを出力するかどうかを指定する. .TRUE. ならカラー用の PS ファイルを出力する; .FALSE. ならモノクロ用の PS ファイルを出力する (省略値は .TRUE.).

'LSEP' (L) 出力ファイルを 1 ページごとに分割して出力するかどうかを指定する. .TRUE. なら分割する; .FALSE. なら分割しない (省略値は .FALSE.).

'LPRINT' (L) 作成された出力ファイルをプリンターに直接出力するかどうかを指定する. .TRUE. ならプリンターに直接出力する; .FALSE. なら出力しない (省略値は .FALSE.).

'LPR' (C) 'LPRINT' を.TRUE. としたときに用いられるプリントコマンドを指定する (省略値は 'lpr')).

[テクトロ端末 (Tek) に関するパラメーター]

'LWAIT' (L) 改ページのタイミングで一時停止するかどうかを指定する. .TRUE. なら一時停止する; .FALSE. なら一時停止しない (省略値は .TRUE.).

'NLNSIZE' (I) テクトロ画面のうちコンソール部分の 1 行の長さを指定する (省略値は 21).

4. 備考

(a) 内部変数を管理するための下請けルーチンとして以下のものがある. ここではサブルーチン名をあげるにとどめる.

SWPQNP SWPQID SWPQCP SWPQVL SWPSVL
SWCQNP SWCQID SWCQCP SWCQVL SWCSVL

(b) SWpGET は上に述べた SWPQID を呼んで内部変数の位置を求め, SWPQVL によって値を参照する; SWpSET は SWPQID を呼んで内部変数の位置を求め, SWPSVL によって値を設定する. したがって指定した内部変数名が見つからないとき, エラーメッセージは SWPQID が出力する.

(c) SWCGET は上に述べた SWCQID を呼んで内部変数の位置を求め, SWCQVL によって値を参照する; SWCSET は SWCQID を呼んで内部変数の位置を求め, SWCSVL によって値を設定する. したがって指定した内部変数名が見つからないとき, エラーメッセージは SWCQID が出力する.

(d) IPARA としては適切な型の定数または変数を指定すること.

7.8.2 使用例

以下では, 電脳ライブラリが標準的に提供する図形出力装置 (X サーバー, ポストスクリプト, テクトロ端末) に対する使用例をいくつか紹介する.

- **X でマウスを使わずにコントロールする:**

  'LKEY'=.TRUE. を指定し, さらに, kterm などのターミナルエミュレータと描画ウインドウの一部が重なるように 'IPOSX', 'IPOSY' を指定して, そこにあらかじめマウスカーソルを移動させておくと, キーボードから手を離さずに操作できる.

- **アニメーション作成 (その 1):**

  1 ページの画面が比較的短時間で用意できる場合は, 'LALT'=.TRUE., 'LWAIT'=.FALSE. とすることで, アニメーション表示が可能になる.

- アニメーション作成 (その 2):

  上記の方法では遅い場合, 'LDUMP'=.TRUE. として xwd ファイルを作成し, animate やビデオ作成ユーティリティーなどを使ってアニメーションを作成する.

  'LWAIT', 'LWAIT0', 'LWAIT1' をすべて .FALSE. にしておけば, 自動的にファイルを作成することが可能であるが, X サーバーを使っているため画面表示をせずにこれを行なうことはできない.

- 大量プリント:

  大量のプリントをしたいとき, 'LPRINT'=.TURE. にしておくと, 1 ページごとにプリンタに送るので, ディスクが足りなくなることが少くなる.

  また, プリントサーバーが別になっているシステムでは, 実行中のプログラムとプリントジョブが並列処理されることになるので, 処理全体が高速化される.

- 長時間ジョブのモニター:

  数値実験などの長時間ジョブのモニターとして, X サーバーを使うことも可能だが, プリンタに出力する方が, 常時監視する必要もなく, 前後の比較などもできて便利である.

  そのような場合も, 'LPRINT'=.TURE. にしておけば, 改ページのタイミングで出力されるので, 途中経過をプログラム実行中にモニターすることができる.

7.9 各種データベース

GRPH1 では, 図形描画のためにいくつかのデータベースファイルを必要とする (オリジナルデータはすべてアスキー形式で記述されている). 標準ライブラリでは, これらのオリジナルデータベースファイルは src/env1 の下におかれており, 必要に応じて変換された後, 内部変数 'DSPATH' の示すディレクトリに移される.

以下に示すデータベースファイルのうち 「フォントファイル」, 「カラー ・トーン変換テーブルファイル」の 2 つは必ず用意されていなければならない.

ファイルの記述方法などの詳細に関しては, env1 の下の各ディレクトリにあるメモを参照していただきたい.

なお, env1 の下には, 以下で述べるデータベースファイルのほかに, GRPH2 が用いる地図情報データも収められている.

7.9.1 フォント

'FONT1', 'FONT2' の示す 2 種類のフォントファイルがある (8.1.2, 8.1.3節参照). インストール時に, コンバージョンプログラム (CVFONT) によって, アスキー形式のデータファイルを書式なしのデータファイルに変換しなければならない. これは, 書式なしのデータファイルの構造がコンパイラによって異なるためである.

このデータファイルは SWPACK の中では直接使用されず, SZPACK の中の SZFONT からアクセスされる. ファイル名は SWQFNM によって検索される. なお, このデータは文字描画ルーチンが最初に呼ばれたときに読み込まれ, フォント番号を変えない限りは再びアクセスされない.

7.9.2 カラー ・トーン番号変換テーブル

色を用いたべた塗り (下位 3 桁が 999 であるようなトーンパターン番号) による塗りわけをおこなうとき, 色が使えないような出力装置において (あるいは SGpGET/SGpSET の管理する内部変数'LCL2TN' を .TRUE. としたとき), 色番号をドットなどのパターン番号として読みかえるための変換テーブル. 内部変数 'CL2TN' が示すファイルとして必ず用意されていなければならない.

このデータファイルは SWPACK の中では直接使用されず, SZPACK の中の SZCLTN からアクセスされる. ファイル名は SWQFNM によって検索される. なお, このデータはトーンプリミティブが最初に呼ばれたときに読み込まれ, それ以降はアクセスされない.

7.9.3 カラーマップ

色番号と実際の色を対応づけるデータファイルであって, 内部変数 'CLRMAP' が示すファイルである. ファイル名には機種を区別する拡張子を付けることができる. たとえば, 標準ライブラリにおいては, X 用として .x11, PS 用として .psx のサフィックスをつけると, それらが優先的に検索される.

色番号 0 はバックグラウンドカラーを指定するもので, それ以外は ICLIDX (7.4.1参照) に対応する. 色は RGB の強さで指定されており, 値の範囲は 0 から 65535 (16bit) である. 機種によってはこれだけの階調が

表現できない場合もあるが, SWPACK がこの形式のファイルを読んで適当な階調に読み変える. したがって, 異なるシステムでもこのファイルはそのまま有効になる (ように SWPACK を書かなければならない).

色番号と実際の色はシステムによって異なるが, 色番号 1 から 5 までは, 標準的に次の様な対応になっている.

色番号	色
1	白または黒
2	赤
3	緑
4	青
5	黄

(色番号 2 から 4 までは RGB の順番である.)

7.9.4 ビットマップ

'BITMAP' が示す Hard Fill 用の標準データファイル. Hard Fill のパターンを指定するデータ形式は機種によって様々なので, あらかじめ, このデータをもとにそれぞれの機種にあったデータ形式に変換しておく必要がある.

なお, このデータは 1 サイクル 2–3mm の長さになるようなデバイスを想定して用意されている.

第 8 章

付録

8.1 各種テーブル

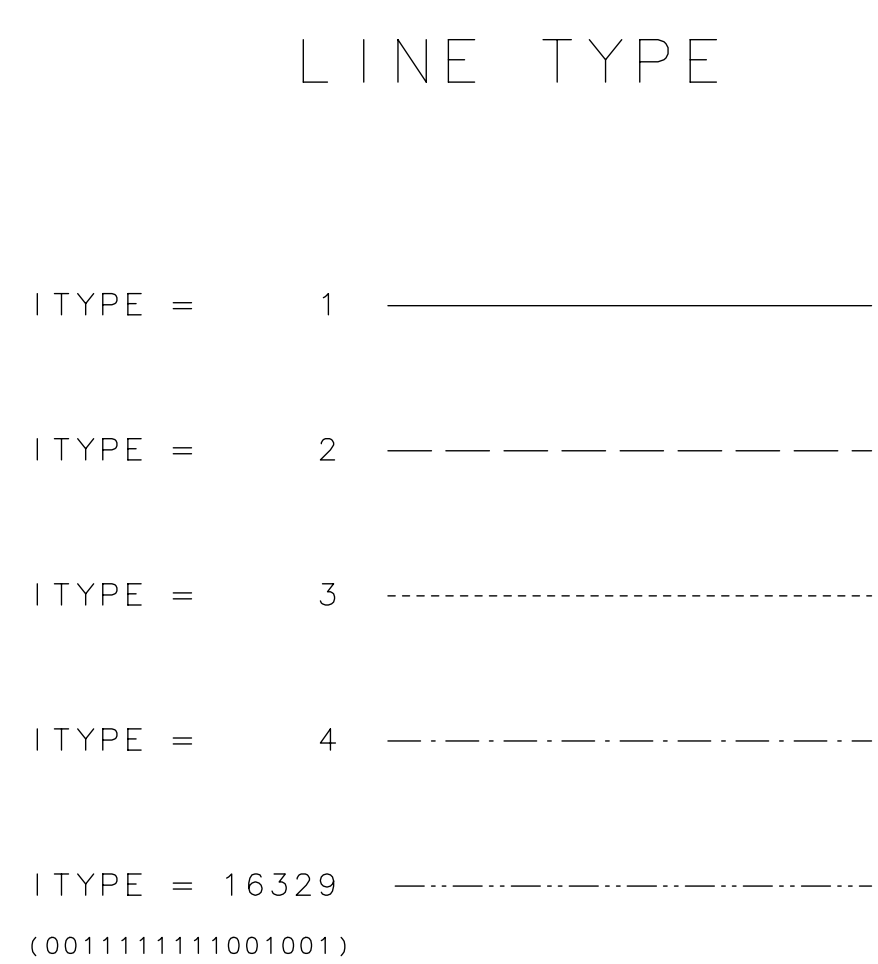
以下では, ラインタイプ, フォント, トーンパターンに関する各種テーブルを示す. これらのテーブルを生成するプログラムは dcl-x.x/demo/grph1/sgpack の中にある.

- ラインタイプについては sgltyp.f,
- フォントについては sgfont.f,
- トーンパターンについては sgtone.f,

によって対応するテーブルを生成できる.

8.1.1 ラインタイプ

ラインタイプとして指定する折れ線の線種を以下に示す. 最後の線種は, ビットパターンでを用いて設定した例である. プログラムは dcl-x.x/demo/grph1/sgpack に sgltyp.f として収めてあるので参考にされたい.

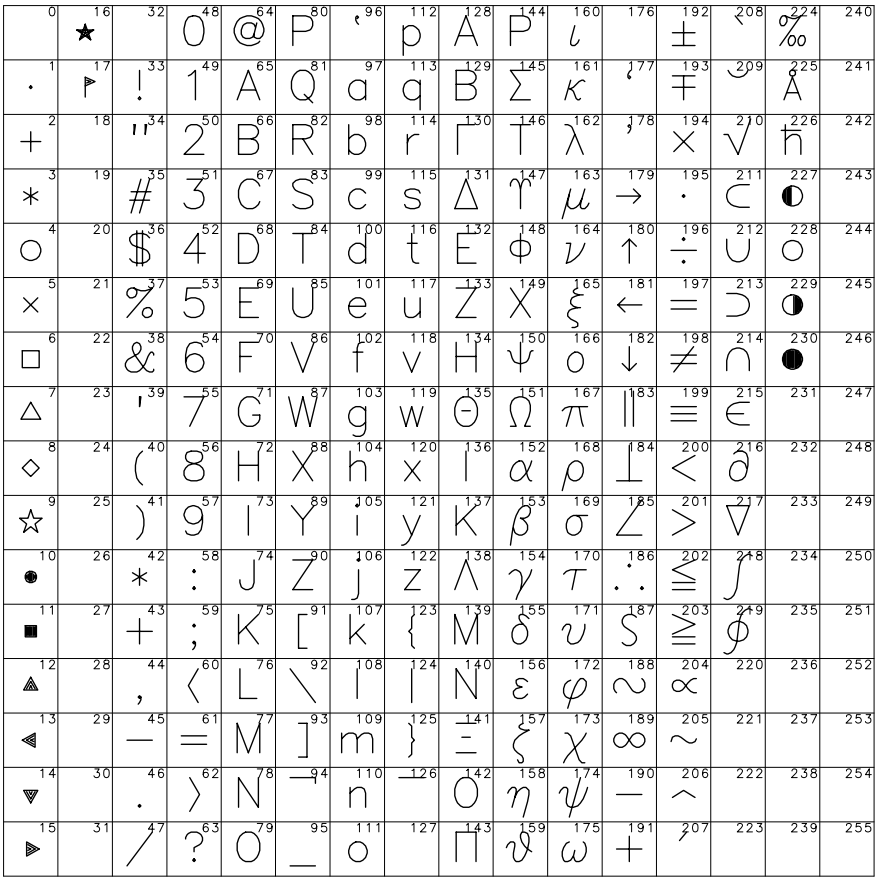


sgltype.f:page1

8.1.2 フォントテーブル 1

次のテーブルは、フォント番号 1 のフォントテーブルである. プログラムは dcl-x.x/demo/grph1/sgpack に sgfont.f として収めてあるので参考にされたい.

FONT NO. = 1

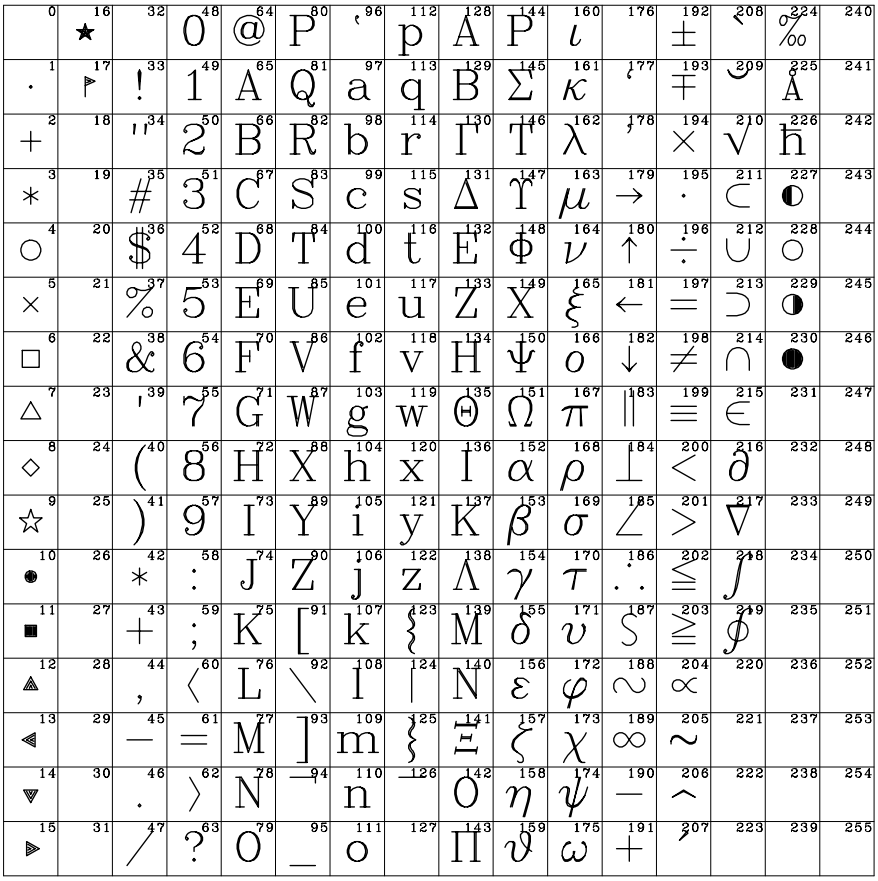


sgfont.f:page1

8.1.3 フォントテーブル 2

次のテーブルは、フォント番号 2 のフォントテーブルである.

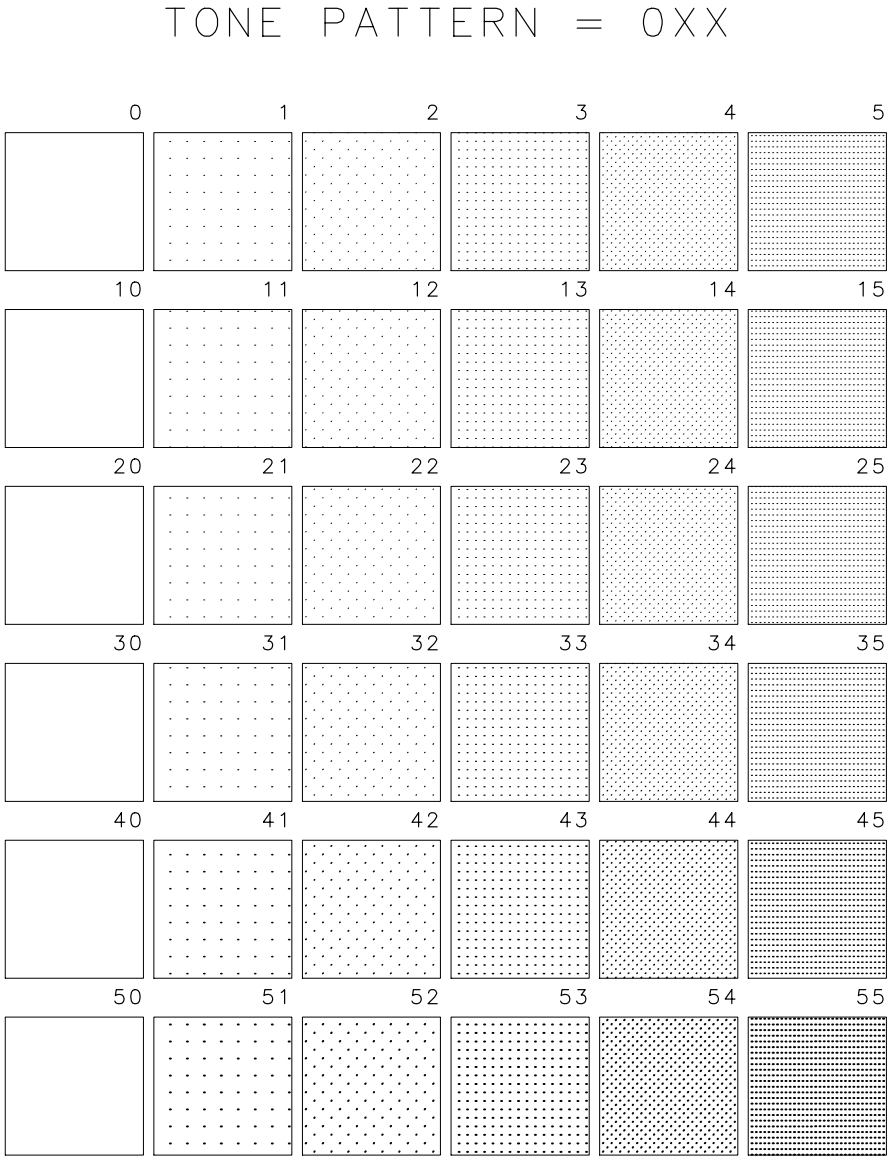
FONT NO. = 2



sgfont.f:page2

8.1.4 トーンパターン 0

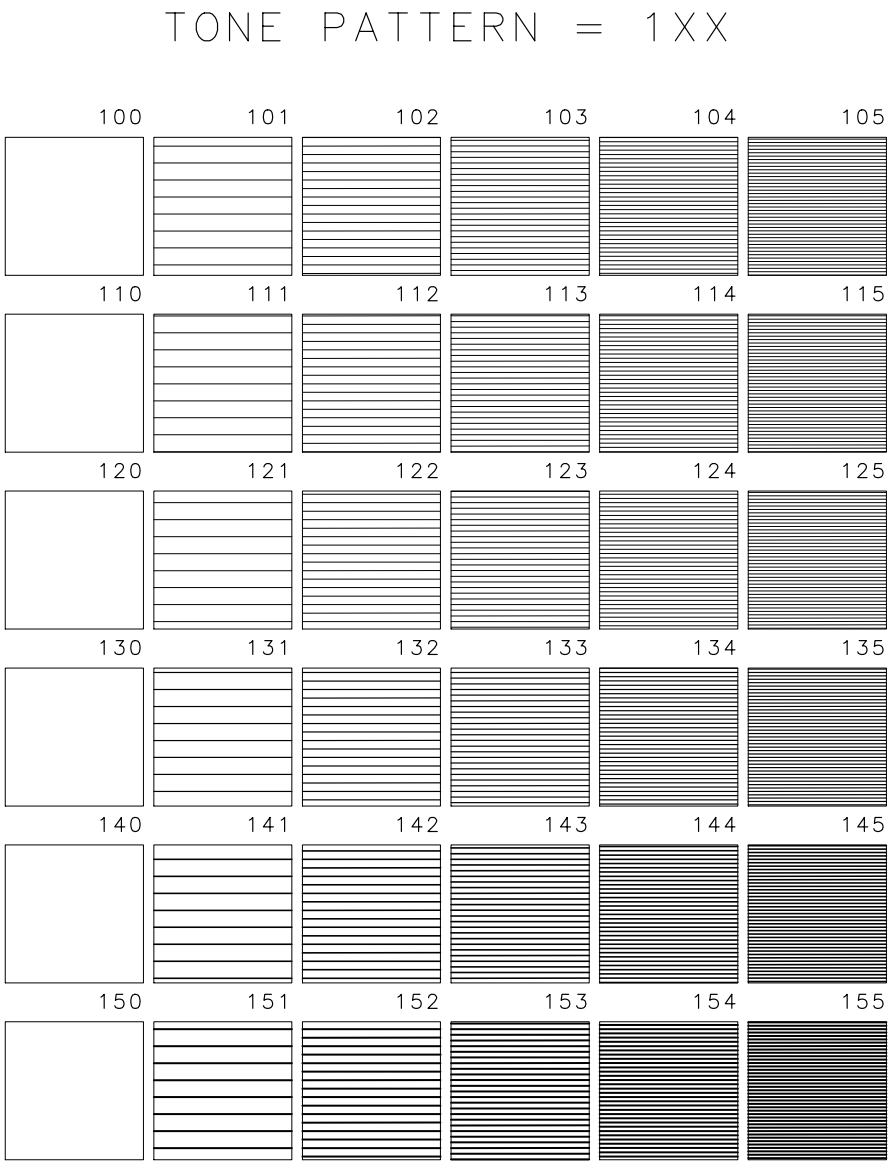
次のテーブルは、パターン番号の百の位が 0 のトーンパターンテーブルである. 以下のトーンパターンのプログラムは dcl-x.x/demo/grph1/sgpack に sgtone.f として収めてあるので参考にされたい.



sgtone.f:page1

8.1.5 トーンパターン 1

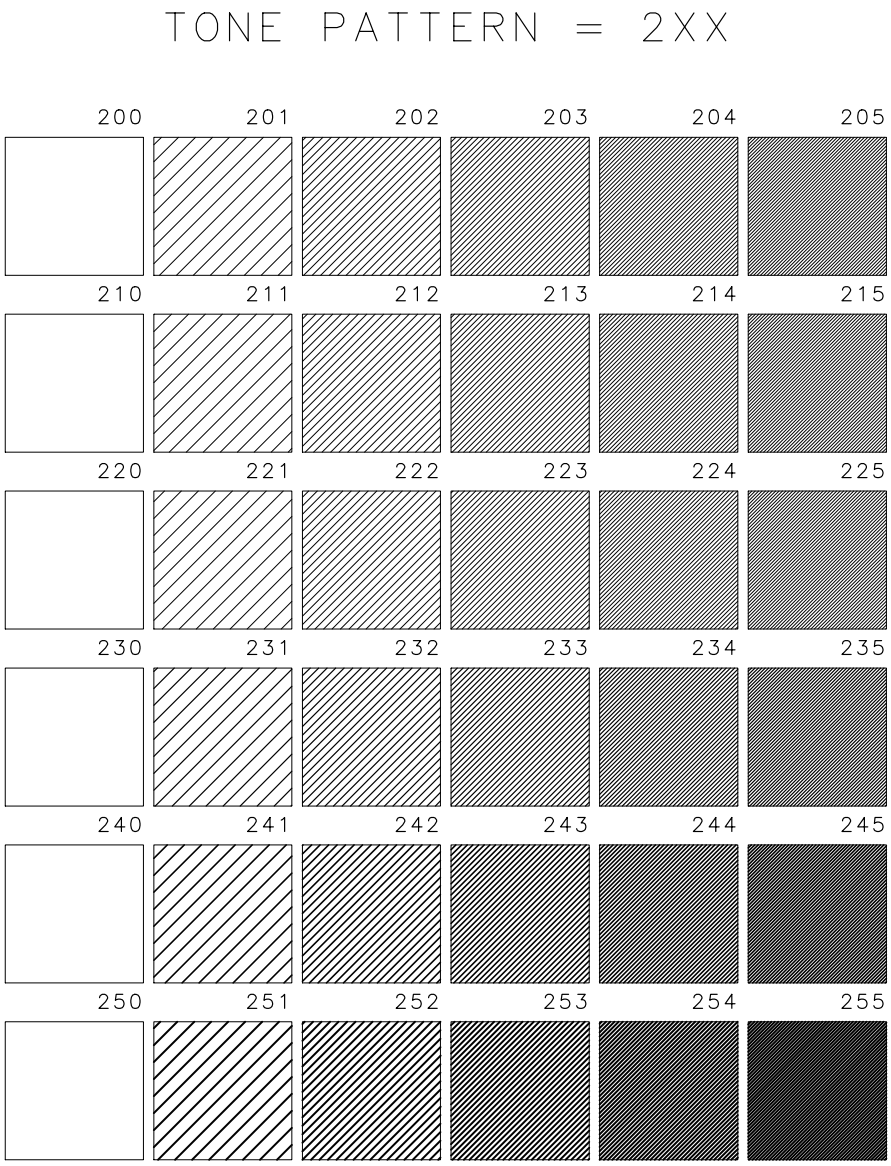
次のテーブルは、パターン番号の百の位が 1 のトーンパターンテーブルである.



sgtone.f:page2

8.1.6 トーンパターン 2

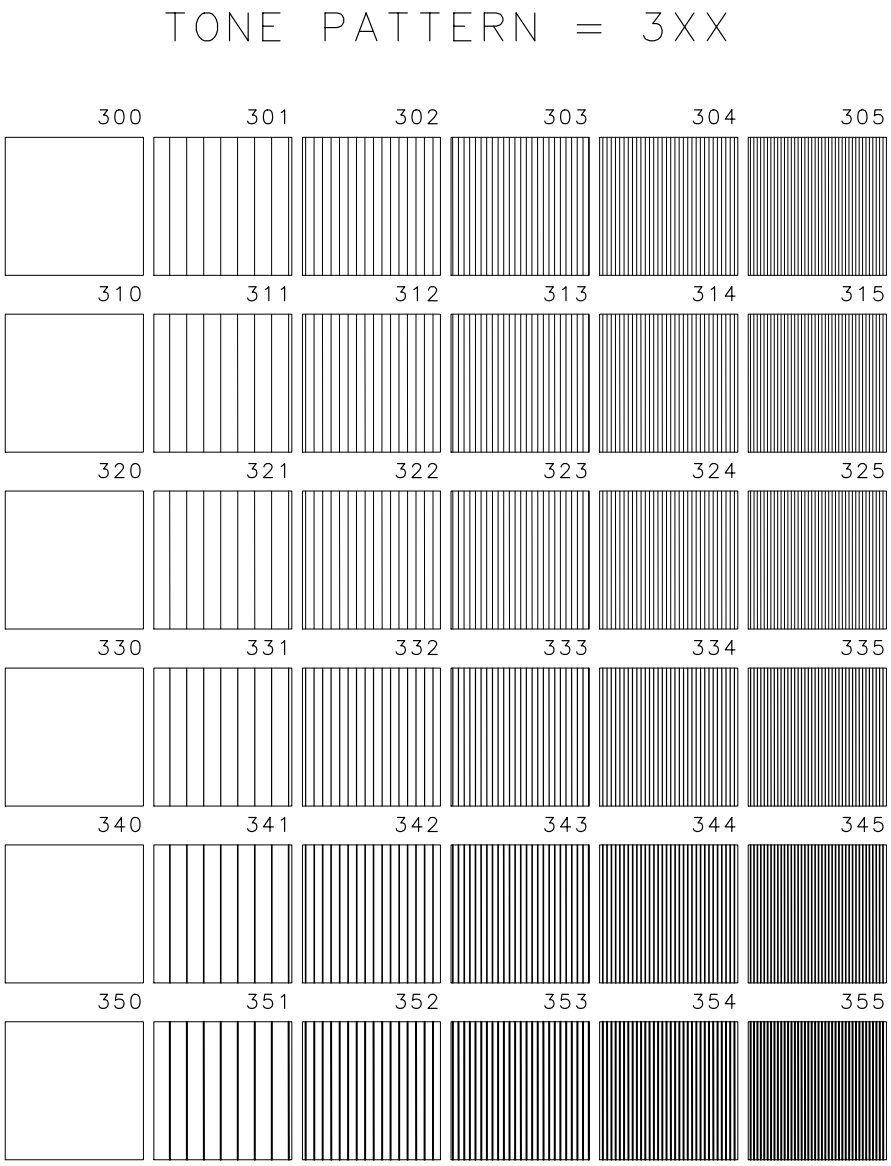
次のテーブルは、パターン番号の百の位が 2 のトーンパターンテーブルである.



sgtone.f:page3

8.1.7 トーンパターン 3

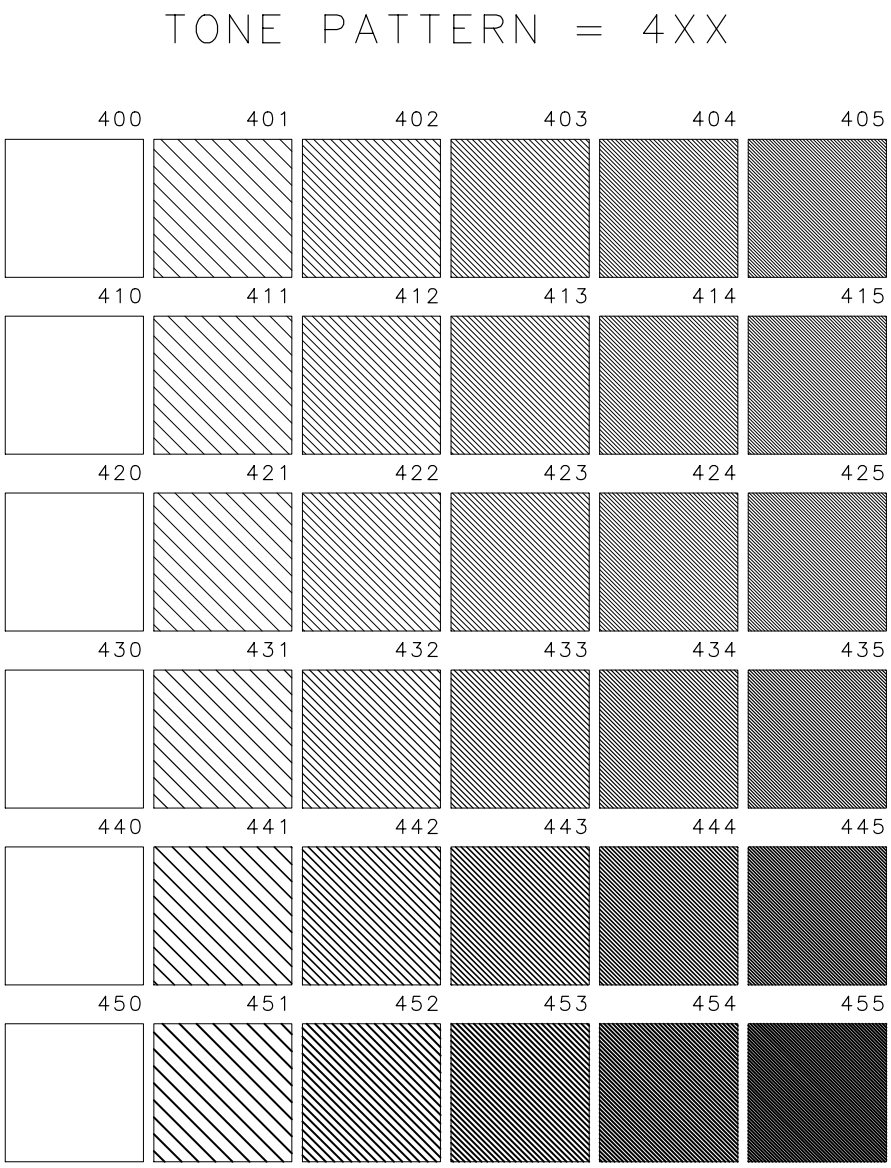
次のテーブルは、パターン番号の百の位が 3 のトーンパターンテーブルである.



sgtone.f:page4

8.1.8 トーンパターン 4

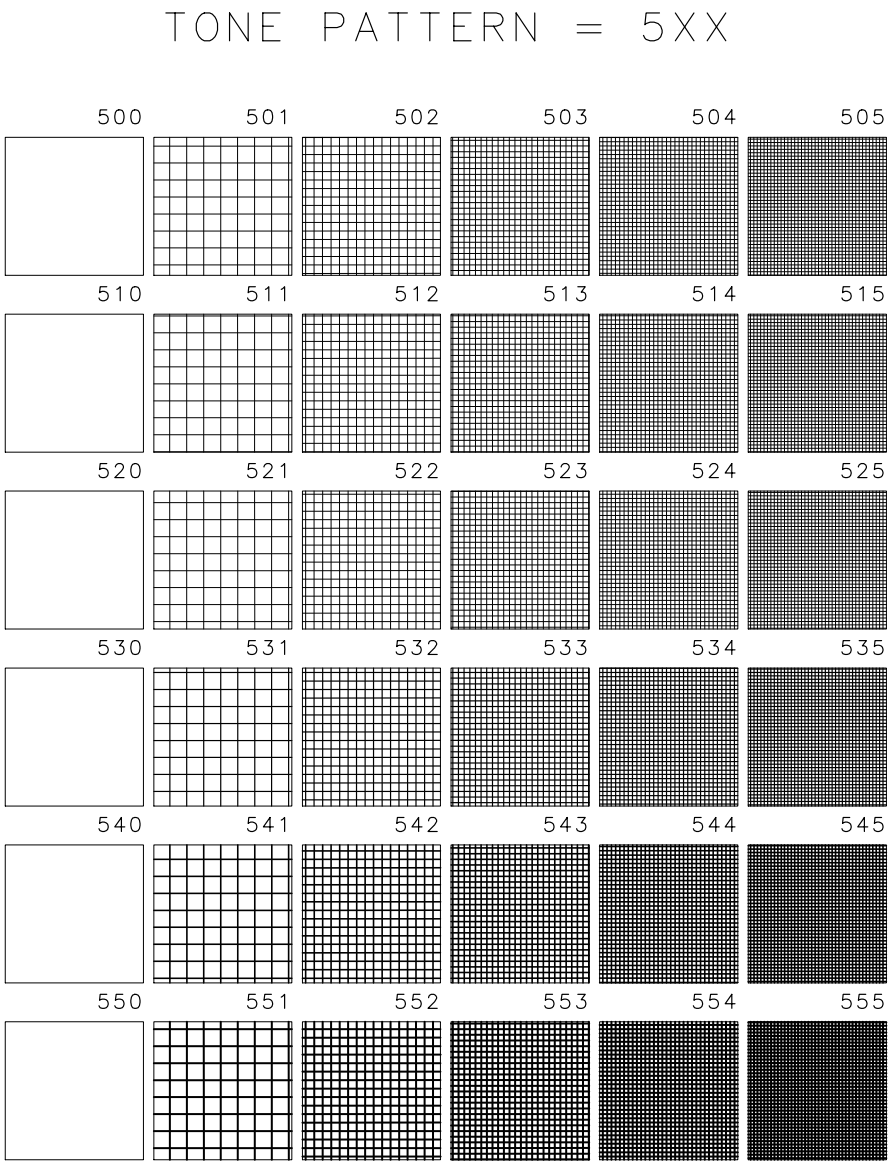
次のテーブルは、パターン番号の百の位が 4 のトーンパターンテーブルである.



sgtone.f:page5

8.1.9 トーンパターン 5

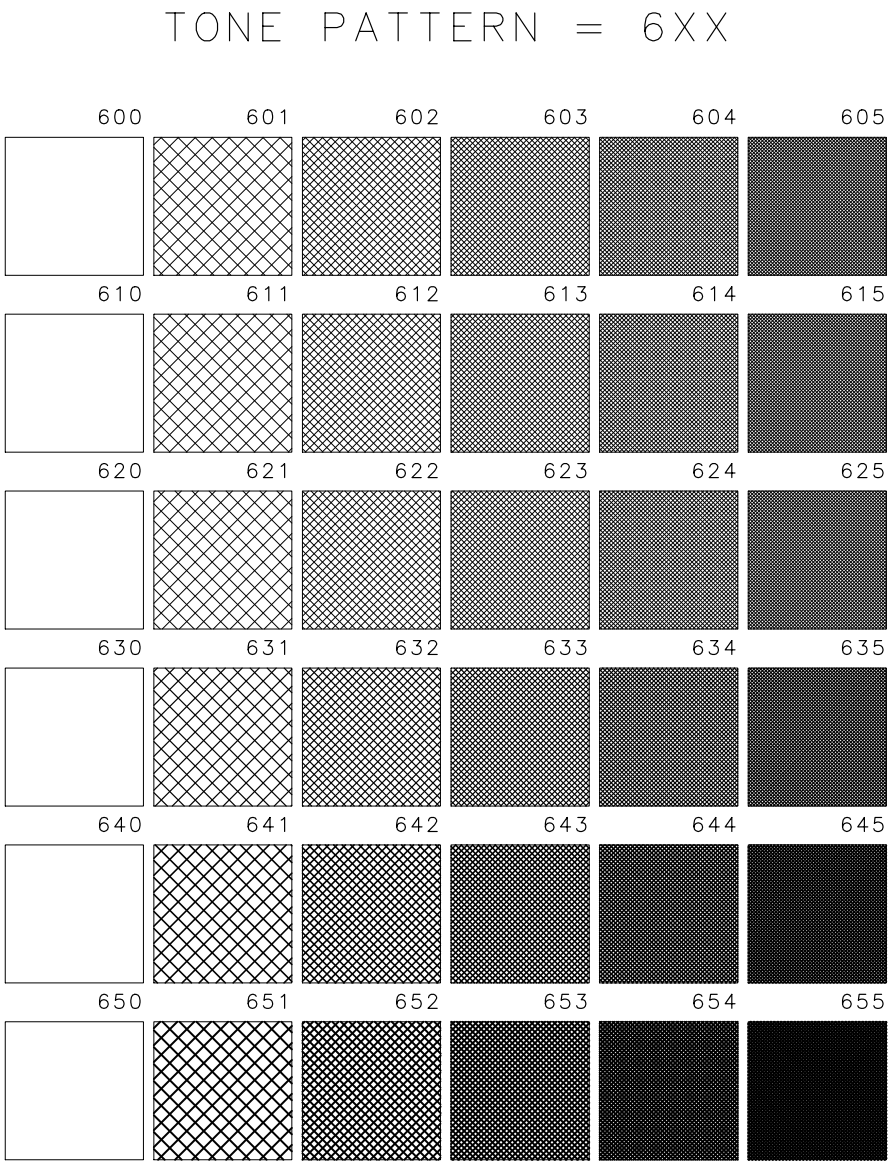
次のテーブルは、パターン番号の百の位が 5 のトーンパターンテーブルである.



sgtone.f:page6

8.1.10 トーンパターン 6

次のテーブルは、パターン番号の百の位が 6 のトーンパターンテーブルである.



sgtone.f:page7

謝辞

資源	**dcl-5**
プログラム製作	石岡圭一, 酒井敏, 佐藤薫, 塩谷雅人, 沼口敦
プログラム作成協力	石渡正樹, 竹広真一, 中島健介, 西憲敬, 林祥介, 保坂征宏, 余田成男
デバッグ協力	石田明生, 石渡正樹, 市川香, 沖大幹, 沖理子, 桜井隆博, 佐藤正樹, 竹広真一, 中島健介, 沼口敦, 根田昌典, 野村真佐子, 藤尾伸三, 保坂征宏
文章	酒井敏, 塩谷雅人, 余田成男
文章協力	石渡正樹, 佐藤薫, 佐藤正樹, 竹広真一, 中島健介, 西憲敬, 沼口敦, 林祥介, 保坂征宏
データソース	NCAR [フォント情報, 地図情報]
参考資源	Layout [二木徹], XYGRAPH [佐藤亨], Numerical Recipes [Willam H. Press et al.], NCARG [NCAR], FFTPACK [Paul Swarztrauber]

版権

本資源の版権は地球流体電脳倶楽部に属する. 資源の利用にあたっては地球流体電脳倶楽部の定める規定にしたがっていただきたい. 原則として, 教育的目的の場合には自由に使用・改変して良いものとしている. 地球流体電脳倶楽部が定める規定の詳細は地球流体電脳倶楽部サーバー

```
dennou.gaia.h.kyoto-u.ac.jp
```

上の

```
~ftp/saloon/dennou/To.Users
```

というファイルに収められているので参照されたい. (このファイルは電脳ライブラリを収めたファイルシステムのトップディレクトリにも置かれている.) 上記のアクセスができない場合は, 電子メールにて地球流体電脳倶楽部管理グループ

```
dennou_admin@gaia.h.kyoto-u.ac.jp
```

へ連絡されたい.

引用について

「地球流体電脳ライブラリ」の著作者は「地球流体電脳倶楽部」, その英語名は 'GFD-DENNOU Library', 'GFD-DENNOU Club' である. 地球流体電脳ライブラリを用いて作成された著作物等には, 英語では例えば "The figures were produced by GFD-DENNOU Library." 日本語では例えば「図の作成には地球流体電脳ライブラリをもちいた.」のように引用していただければ幸いである.