



## CP/M用プロッタサブルーチンパッケージ

メタデータ	言語: jpn 出版者: 公開日: 2013-11-12 キーワード (Ja): キーワード (En): 作成者: 高橋, 参吉, 西山, 美樹夫 メールアドレス: 所属:
URL	<a href="https://doi.org/10.24729/00007993">https://doi.org/10.24729/00007993</a>

# CP/M用プロッタサブルーチンパッケージ

## Plotter Subroutine Package

### Used under the CP/M Floppy Disk Operating System

高橋 参吉\*, 西山 美樹夫\*\*  
Sankichi TAKAHASHI\*, Mikio NISHIYAMA\*\*

(昭和59年4月18日 受理)

#### あ ら す じ

パーソナルコンピュータの汎用 OS の一つである CP/M の管理下で、簡易型の X-Y プロッタ装置を動かすソフトウェア（以下、CP/M用PSP という。PSP：プロッタサブルーチンパッケージ）の開発を行なった。作成した CP/M用PSP は、パソコンと本校情報処理センター（FACOM M-150F）との互換性を高めるために、FACOM用PSP と同じ様な機能を持たせている。

本論文では、CP/M用PSP の概要を述べるとともに、CP/M FORTRAN-80 による作成プログラムならびに 2 つの作画例を示している。

#### 1. ま え が き

最近パーソナルコンピュータの進歩は著しく、現在では教育用あるいは研究用にまで使われている。また周辺装置の進歩もめざましく、本論文で取り扱っている簡易型 X-Y プロッタも、3～4 年前より低価格で良い性能の製品が市販されている。

一方プロッタのソフトウェアは、グラフ作成を中心としたものがプロッタメーカーより市販されているが、FACOM PSP<sup>1)</sup> のような汎用的なプロッタのソフトウェアは少ない。したがって、パソコンのソフトウェアを充実させれば、教育・研究用にさらに幅広い利用が考えられる。

今回作成した CP/M用PSP は、FACOM用PSP と互換性を高めたことにより次の様な利点がある。

- (1)電気工学科では情報処理入門教育で FACOM PSP も教材として取り上げており、授業や実験などでパソコンを用いる時同じ機能のものがあれば教育効果が上がる<sup>2)</sup>。
- (2)PSP を含む FACOM用FORTRAN プログラムが、若干の制約はあるがパソコンでも実行できる。

使用できるプロッタは、マイプロット<sup>3)</sup> と同様の機能を持つ機種であれば良い。ただし、機能の違いによって FACOM用PSP と完全に同じ作画はできないが実用上問題はない。

#### 2. CP/M用PSP の概要

CP/M用PSP は直接ユーザーが利用できる基本サブルーチンとその他のサブルーチンとから成っている。基本サブルーチンは FACOM用PSP と互換性を持たせるため、FACOM と同じ

---

\* 電気工学科 (Department of Electrical Engineering)

\*\* 情報処理センター (Information Processing Center)

変数名等を使用している。詳しくは文献<sup>1)</sup>に述べられており、ここでは機能の概要・引数の意味を簡単に説明する。またその他のサブルーチンは基本サブルーチンで引用されているもので、主に簡易型プロッタの出力に関するサブルーチンである。なお、作成プログラムは本論文最後に一括して載せている。

## 2.1 基本サブルーチン

CP/M用PSPの基本サブルーチンは、初期化・終了化サブルーチン (PLOTS, PLOTE), 作画データ作成サブルーチン (PLOT, SYMBOL, LINE, NUMBER, AXIS) および補助サブルーチン (PFIELD, RDSCIS, SCALE, FACTOR, SET) から成っている。これらのサブルーチンの機能を表1に示す。

表1 基本サブルーチンの機能

サブルーチン名	機能
PLOTS	プロッタの起動。座標原点, 有効作画範囲 (プロットエリア) 等の初期設定。
PLOTE	プロッタの終了。ペンをホームポジションに戻す。
PLOT	指定座標値への移動または線画。座標原点の再設定。
SYMBOL	指定座標値への移動または線画後, 文字または文字列の作画。
LINE	与えられたデータに沿って折線またはマークの作画。
NUMBER	指定座標値への移動後, 浮動小数点数を作画。
AXIS	指定座標値への移動後, 座標軸の作画。
PFIELD	プロットフィールドの定義およびシザリングモードの指定。
RDSCIS	シザリングモードの変更。
SCALE	与えられたデータ群の作画のための初期値・増分値の計算とその通知。
FACTOR	図形を拡大または縮小させるための倍率係数の指定。
SET	座標原点の再設定およびプロットフィールドの終了。

次に各サブルーチンの引数の意味等を説明する。

### (1) PLOTS (XI, YI, LD)

PSPを使用する際に一番最初に入れるサブルーチンであり, 指定座標値へ移動後その座標値を原点とする。XI, YI(実数型): ホームポジションからのX, Y座標値(単位は以後すべてcm)。LD(整数型): プロッタ装置の番号。1はマイプロット(WX4671), 2はローランド・ディージェー(DXY-101)を想定している。

### (2) PLOTE (COM)

PSPを使用する際に一番最後に入れるサブルーチンであり, コメントをディスプレイ上に表示する。COM(文字データ): 8文字以内のコメント。

### (3) PLOT (XP, YP, PM)

XP, YP(実数型): 原点からの指定X, Y座標値。PM(整数型): ペンモードで,  $\pm 2$ はペンをダウンで移動(線画),  $\pm 3$ はペンをアップで移動。負の時は移動後その座標値を原点とする。

### (4) SYMBOL (XP, YP, HE, TEXT, ANG, NCH)

XP, YP(実数型): 原点からの指定X, Y座標値。HE(実数型): 文字の高さ。ただし0.07~1.12cmまで0.07cmきざみ。TEXT(文字データあるいは倍精度実数型): 作画したい文字(最大8文字)あるいは文字コード・マーク(倍精度実数値で与える)。ANG(実数型): 作画文字の角度(度)。ただし $\pm 90$ 度きざみ。NCH(整数型): TEXTに格納されている文字数。あるいは0の時文字コード, -1の時コードで示されるマーク, -2以下の時指定座標値まで線

画後マークをそれぞれ作画。なお文字列が8文字を越える時は、XP, YPに9999.0を指定すれば連続して作画する。

#### (5)LINE (X, Y, N, INC, LIT, INTEQ)

X, Y (実数型): X, Y座標データが格納されている一次配列。N (整数型): データ数をCOUNTとすると $N = \text{COUNT} \times |\text{INC}| + |\text{INC}| + 1$ で与えられる数で、通常は $\text{COUNT} + 2$ 。INC (整数型): 各データから取り出す座標データの間隔。LIT (整数型): 指定値をnとすると、正の時 $|n| - 1$ 点ごとに実線で各データを結びマークの作画, 0の時実線で各データを結ぶ, 負の時 $|n| - 1$ 点ごとにマークの作画。INTEQ (整数型): マークの種類。ただしマイプロットでは1~6, ローランド・ディージェーでは1~10。

#### (6)NUMBER (XP, YP, HE, NUMB, ANG, NDEC)

XP, YP, HE, ANGはSYMBOLサブルーチンの引数と同様である。NUMB (実数型): 作画したい数値で符号・小数点を含め10桁以内。ただし小数点以下は6桁以内。NDEC (整数型):  $-1$ 未満の時整数部分のみ作画。ただし $|\text{NDEC}| - 1$ を作画しない。 $-1$ の時整数部分のみ, 0の時整数部分と小数点, 正の時整数部分と小数点および小数点以下NDEC桁を作画する。

#### (7)AXIS (XP, YP, TEXT, NCH, AXLEN, ANGLE, FIRSTV, DELTAV)

XP, YP (実数型): 座標軸の作画開始座標。TEXT (文字データ): 座標軸の名前(最大8文字)。NCH (整数型): 正の時反時計方向に, 負の時時計方向に軸名を作画。AXLEN (実数型): 座標軸の長さ(cm)。ANGLE (実数型): 座標軸の角度( $\pm 90$ 度きざみ)。FIRSTV (実数型): 座標軸の最初の目盛値。DELTAV (実数型): 座標値の単位長当たりの増分値。

#### (8)PFIELD (PLLX, PLY, PURX, PURY, SCI)

有効作画範囲を限定するサブルーチンで, 指定の内側のみ作画する場合(外抜きプロットフィールド)と指定の外側のみ作画する場合(内抜きプロットフィールド)がある。PLLX, PLY, PURX, PURY (実数型): プロットフィールド境界の左下端点および右上端点のX, Y座標値。SCI (整数型): シザリングモード。1の時プロットフィールドの外側を切りとる(外抜きシザリング)。2の時内側を切りとる(内抜きシザリング)。0の時RDSCISサブルーチンで指定されるまでシザリングを行なわない。

#### (9)RDSCIS (SCI)

SCIはPFIELDサブルーチンの引数と同様である。

#### (10)SCALE (DATA, AXLEN, N, INC)

DATA (実数型): 座標データが格納されている一次元配列。AXLENはAXIS, NはLINE各サブルーチンの引数と同様である。INC (整数型): 座標データの取り出し間隔。正(負)の時増分値は正(負)となる。初期値・増分値は, DATAの $\text{COUNT} \times |\text{INC}| + 1$ と $\text{COUNT} \times |\text{INC}| + |\text{INC}| + 1$ に格納される。その決定方法は文献<sup>1)</sup>によっている。

#### (11)FACTOR (SF)

SF (実数型): 倍率係数。初期設定値は1.0である。

#### (12)SET (XI, YI)

XI, YI (実数型): ホームポジションからのX, Y座標値

## 2.2 その他のサブルーチン

その他のサブルーチンはPLOTで引用されているPLOT0, PLOT1, PLOT2(0, 1, 2はシザリングモードに対応), CROSS(プロットフィールド境界と現在値・指定値間の線分との交点を求めるサブルーチン)およびプロッタのコマンドに対応したサブルーチンから成る。

表2にプロッタのコマンドに対応したサブルーチンの機能を示す。なお、パソコンとプロッタの接続はプリンタ端子（8ビットパラレルインターフェイス）としている。

表2 プロッタのコマンドに対応したサブルーチンの機能\*1

サブルーチン名	機 能	サブルーチン名	機 能
SUBH	ホームポジションへペンを移動。	SUBQ	文字の角度の指定。
SUBD	指定座標値へペンダウンで移動。	SUBP	英数字の作画。
SUBM	指定座標値へペンアップで移動。	SUBN	マークの作画。
SUBS	文字の大きさの指定。	SUBU	ひらがな、カタカナ等*2の作画。

\*1 サブルーチン名の SUB を省略した名がコマンド名    \*2 マイプロットでは使用不可

### 3. 作画例

#### 3.1 プロットフィールドによるシザリング（作画例1）

PFIELDサブルーチンを用いて、内抜き・外抜きシザリングを行なった作画例を図1に、そのメインプログラムを図2に示す。

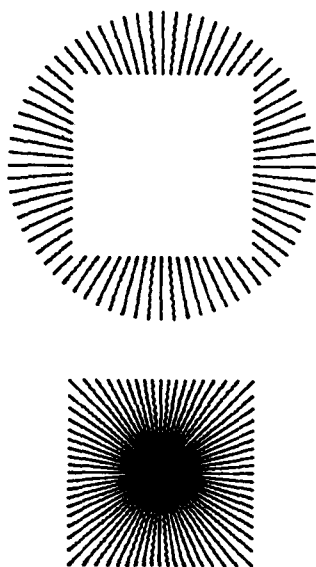


図1 作画例1

```

C**** MAIN PROGRAM
DIMENSION CR(36),SR(36)
RAD=3.14159/180.0
DO 10 I=1,36
WS=RAD*(I-1)*5.0
CR(I)=2.5*COS(WS)
SR(I)=2.5*SIN(WS)
10 CONTINUE
CALL PLOTS(10.0,5.0,2)
DO 20 I=1,2
CALL PFIELD(-1.5,-1.5,1.5,1.5,I)
DO 30 J=1,36
CALL PLOT(CR(J),SR(J),3)
CALL PLOT(-CR(J),-SR(J),2)
30 CONTINUE
CALL SET(10.0,10.0)
20 CONTINUE
CALL PLOTE('CP/M PSP')
STOP
END

```

図2 メインプログラム

#### 3.2 SIN, COS およびその合成波形（作画例2）

SCALE, AXIS, LINE, SYMBOL サブルーチンを使った作画例を図3に、そのメインプログラム図4に示す。

### 4. むすび

今回作成した PSP では、特に文字に制限（大きさ、角度）があった。これはプロッタの機能によるものである。今後、BASIC 言語による書き換え、PSL（ライブラリー）の作成を検討したい。

プログラム作成には、NEC PC8801, dux PDS-V を使用した。PDS-V の使用に関しては、

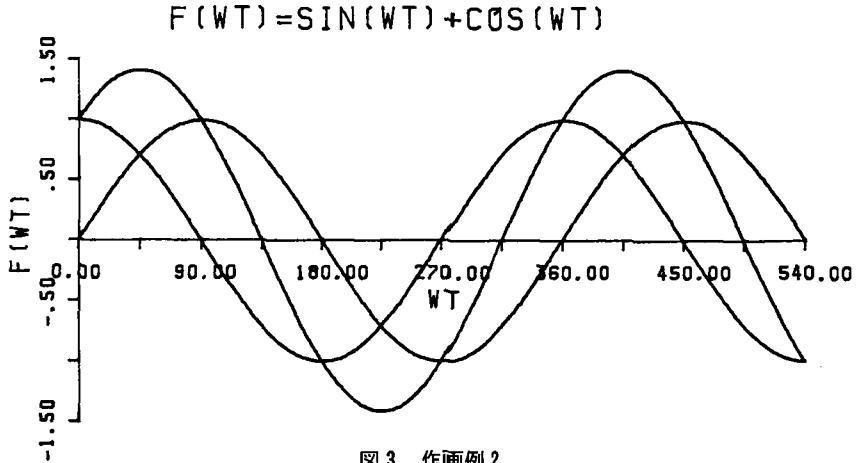


図3 作画例2

```

C**** MAIN PROGRAM
DIMENSION X(543),YS(543),YC(543),Y(543)
RAD=3.14159/180.0
DO 10 IWT=1,541
X(IWT)=FLOAT(IWT-1)
YS(IWT)=SIN(X(IWT)*RAD)
YC(IWT)=COS(X(IWT)*RAD)
Y(IWT)=YS(IWT)+YC(IWT)
10 CONTINUE
CALL PLOTS(5.0,5.0,1)
X(542)=0.0
X(543)=45.0
CALL SCALE(Y,6.0,543,1)
YO=-Y(542)/Y(543)
CALL PLOT(0.0,YO,3)
CALL AXIS(0.0,YO,'WT',-2,12.0,0.0,X(542),X(543))
CALL AXIS(0.0,0.0,'F(WT)',5,6.0,90.0,Y(542),Y(543))
YS(542)=Y(542)
YS(543)=Y(543)
YC(542)=Y(542)
YC(543)=Y(543)
CALL LINE(X,YS,543,1,0,1)
CALL LINE(X,YC,543,1,0,1)
CALL LINE(X,Y,543,1,0,1)
CALL SYMBOL(1.5,6.5,0.35,'F(WT)=SI',0.0,8)
CALL SYMBOL(9999.0,9999.0,0.35,'N(WT)+CO',0.0,8)
CALL SYMBOL(9999.0,9999.0,0.35,'S(WT)',0.0,5)
CALL PLOTE('CP/M PSP')
STOP
END

```

図4 メインプログラム

電気工学科原田教授に、プロッタの使用に関しては、下村教授、須崎助教授にご協力いただき感謝する。

なお本研究は延原善廣君（現フジテック）が昭和58年度の卒業研究として行なったものであり、同君の協力に感謝する。

#### 参考文献

- 1) 富士通：FACOM OSIV/×8 PSP 使用手引書，PSP 文法書（昭53）
- 2) 高橋，中村：マイコンを用いた電力系のシミュレーション，昭和58年度高専情報処理教育研究発表会
- 3) 中村一治監修：プロッタ入門，ラジオ技術社（昭55）
- 4) 高橋，西山：CP/M 用プロッタサブルーチンについて，昭和59年度高専情報処理教育研究発表会

作成プログラム

```

C *****
C *
C *      CP/M  PLOTTER  SUBROUTINE  PACKAGE  *
C *
C *****
C Labeled COMMON BLOCK USED IN THIS PROGRAM
C COMMON /BLK1/XO,YO
C COMMON /BLK2/X1,Y1
C COMMON /BLK3/FACT
C COMMON /BLK4/SM
C COMMON /BLK5/XMIN,YMIN,XMAX,YMAX
C COMMON /BLK6/XLOW,YLOW,XUPP,YUPP
C COMMON /BLK7/LN,MN
C**** PLOTS (PLOT START) SUBROUTINE
SUBROUTINE PLOTS(XI,YI,LD)
COMMON /BLK3/FACT/BLK5/XMIN,YMIN,XMAX,YMAX/BLK7/LN,MN
XMIN=0.0
YMIN=0.0
LN=LD
IF(LN.EQ.2) GOTO 10
XMAX=36.0
YMAX=26.0
MN=6
GOTO 20
10 XMAX=37.0
YMAX=26.0
MN=10
20 WRITE(1,1)
1 FORMAT(1H,'START OF PLOTTER SUBROUTINE')
FACT=1.0
CALL SET(XI,YI)
RETURN
END
C**** PLOTE (PLOT END) SUBROUTINE
SUBROUTINE PLOTE(COM)
REAL*8 COM
WRITE(1,1) COM
1 FORMAT(1H,'END OF PLOTTER SUBROUTINE '/1H,AB)
CALL SUBH
RETURN
END
C**** PLOT (PLOTING) SUBROUTINE
SUBROUTINE PLOT(XP,YP,PM)
INTEGER SM,PM,PMW
COMMON /BLK1/XO,YO/BLK2/X1,Y1/BLK3/FACT/BLK4/SM
*BLK5/XMIN,YMIN,XMAX,YMAX/BLK6/XLOW,YLOW,XUPP,YUPP
PMW=PM
X2=XO+FACT*XP

```

```

Y2=YO+FACT*YP
IF(X2.GT.XMIN.AND.X2.LT.XMAX.AND.Y2.GT.YMIN.AND.Y2.LT.YMAX)
*GOTO 10
WRITE(1,1)
1 FORMAT(1H,'SPECIFIED POINT OVER PLOT AREA')
RETURN
10 IPM=IABS(PMW)
IF(IPM.EQ.2.OR.IPM.EQ.3) GOTO 20
WRITE(1,2)
2 FORMAT(1H,'INVALID PENMODE PARAMETER IN PLOT'/
*1H,'3 WAS ASSUMED')
PMW=3
20 IF(PMW.GT.0) GOTO 30
SM=0
XLOW=XMIN
YLOW=YMIN
XUPP=XMAX
YUPP=YMAX
CALL PLOT0(X2,Y2,IPM)
XO=X2
YO=Y2
GOTO 70
30 IF(SM-1) 40,50,60
40 CALL PLOT0(X2,Y2,IPM)
GOTO 70
50 CALL PLOT1(X2,Y2,IPM)
GOTO 70
60 CALL PLOT2(X2,Y2,IPM)
70 X1=X2
Y1=Y2
RETURN
END
C**** PLOT0 (SCISSORING MODE 0) SUBROUTINE
SUBROUTINE PLOT0(X2,Y2,IPM)
IF(IPM.EQ.2) GOTO 10
CALL SUBM(X2,Y2)
RETURN
10 CALL SUBD(X2,Y2)
RETURN
END
C**** PLOT1 (SCISSORING MODE 1) SUBROUTINE
SUBROUTINE PLOT1(X2,Y2,IPM)
LOGICAL LX1,LX2,IPM2
DIMENSION XC(2),YC(2)
COMMON /BLK2/X1,Y1/BLK6/XLOW,YLOW,XUPP,YUPP
IPM2=IPM.EQ.2
LX1=X1.GT.XLOW.AND.X1.LT.XUPP.AND.Y1.GT.YLOW.AND.Y1.LT.YUPP
LX2=X2.GT.XLOW.AND.X2.LT.XUPP.AND.Y2.GT.YLOW.AND.Y2.LT.YUPP
IF(LX1.AND.LX2) GOTO 70
CALL CROSS(X2,Y2,XC,YC,MODE)
IF(MODE-1) 90,10,40

```

```

10 IF(LX1) GOTO 20
   CALL SUBM(XC(1),YC(1))
   GOTO 70
20 IF(IPM2) GOTO 30
   CALL SUBM(XC(1),YC(1))
   RETURN
30 CALL SUBD(XC(1),YC(1))
   RETURN
40 D1=(X1-XC(1))**2+(Y1-YC(1))**2
   D2=(X1-XC(2))**2+(Y1-YC(2))**2
   IF(D1.LE.D2) GOTO 50
   XW=XC(1)
   YW=YC(1)
   XC(1)=XC(2)
   YC(1)=YC(2)
   XC(2)=XW
   YC(2)=YW
50 CALL SUBM(XC(1),YC(1))
   IF(IPM2) GOTO 60
   CALL SUBM(XC(2),YC(2))
   RETURN
60 CALL SUBD(XC(2),YC(2))
   RETURN
70 IF(IPM2) GOTO 80
   CALL SUBM(X2,Y2)
   RETURN
80 CALL SUBD(X2,Y2)
90 RETURN
END
C**** PLOT2 (SCISSORING MODE 2) SUBROUTINE
SUBROUTINE PLOT2(X2,Y2,IPM)
LOGICAL LX1,LX2,IPM2
DIMENSION XC(2),YC(2)
COMMON /BLK2/X1,Y1/BLK6/XLOW,YLOW,XUPP,YUPP
IPM2=IPM.EQ.2
LX1=X1.GT.XLOW.AND.X1.LT.XUPP.AND.Y1.GT.YLOW.AND.Y1.LT.YUPP
LX2=X2.GT.XLOW.AND.X2.LT.XUPP.AND.Y2.GT.YLOW.AND.Y2.LT.YUPP
IF(LX1.AND.LX2) RETURN
CALL CROSS(X2,Y2,XC,YC,MODE)
IF(MODE-1) 40,10,60
10 IF(LX1) GOTO 30
   IF(IPM2) GOTO 20
   CALL SUBM(XC(1),YC(1))
   RETURN
20 CALL SUBD(XC(1),YC(1))
   RETURN
30 CALL SUBM(XC(1),YC(1))
40 IF(IPM2) GOTO 50
   CALL SUBM(X2,Y2)
   RETURN
50 CALL SUBD(X2,Y2)
   RETURN
60 D1=(X1-XC(1))**2+(Y1-YC(1))**2
   D2=(X1-XC(2))**2+(Y1-YC(2))**2
   IF(D1.LE.D2) GOTO 70
   XW=XC(1)
   YW=YC(1)
   XC(1)=XC(2)
   YC(1)=YC(2)
   XC(2)=XW
   YC(2)=YW
70 IF(IPM2) GOTO 80
   CALL SUBM(XC(1),YC(1))
   CALL SUBM(XC(2),YC(2))
   CALL SUBM(X2,Y2)
   RETURN
80 CALL SUBD(XC(1),YC(1))
   CALL SUBD(XC(2),YC(2))
   CALL SUBD(X2,Y2)
   RETURN
END
C**** CROSS SUBROUTINE
SUBROUTINE CROSS(X2,Y2,XC,YC,MODE)
DIMENSION XC(2),YC(2)
COMMON /BLK2/X1,Y1/BLK6/XLOW,YLOW,XUPP,YUPP
IF(ABS(X1-X2).LT.1.0E-6) GOTO 50
A=(Y1-Y2)/(X1-X2)
B=Y1-A*X1
IF(A.GE.0.0) GOTO 10
BLOW=YLOW-A*XLOW
BUPP=YUPP-A*XUPP
GOTO 20
10 BLOW=YLOW-A*XUPP
   BUPP=YUPP-A*XLOW
20 IF(B.GT.BLOW.AND.B.LT.BUPP) GOTO 30
   MODE=0
   RETURN
30 MODE=2
   K=0
   X=XLOW
   DO 40 I=1,2
   Y=A*X+B
   IF(Y.LE.YLOW.OR.Y.GE.YUPP) GOTO 40
   K=K+1
   XC(K)=X
   YC(K)=Y
40 X=XUPP
   IF(K.EQ.2) GOTO 70
50 Y=YLOW
   DO 60 I=1,2
   X=(Y-B)/A
   IF(X.LT.XLOW.OR.X.GT.XUPP) GOTO 60
   K=K+1
   XC(K)=X

```



```

        YC(K)=Y
60  Y=YUPP
    IF(K.EQ.2) GOTO 70
    MODE=0
    RETURN
70  XXLOW=X1
    XXUPP=X2
    IF(X1.LT.X2) GOTO 80
    XW=X1
    XXLOW=X2
    XXUPP=XW
80  K=0
    DO 90 I=1,2
    IF(XC(I).LT.XXLOW.OR.XC(I).GT.XXUPP) GOTO 90
    K=K+1
    XC(K)=XC(I)
    YC(K)=YC(I)
90  CONTINUE
    MODE=K
    RETURN
    END
C**** SYMBOL (PLOT SYMBOL) SUBROUTINE
SUBROUTINE SYMBOL(XP,YP,HE,TEXT,ANG,NCH)
REAL*8 TEXT
LOGICAL LX,LY
COMMON /BLK1/X0,Y0/BLK2/X1,Y1/BLK3/FACT/BLK7/LN,MN
NCHW=NCH
IF(NCHW.GT.8) NCHW=8
H=FACT*HE
IF(H.GT.0.0) GOTO 10
WRITE(1,1)
1  FORMAT(1H,'INVALID HEIGHT PARAMETER IN SYMBOL'
*/1H,'0.28 WAS ASSUMED')
H=0.28
10  CALL SUBS(H)
    CALL SUBQ(ANG)
    XPW=XP
    YPW=YP
    LX=XPW.GE.9999.0
    LY=YPW.GE.9999.0
    IF(LX) XPW=X1-X0
    IF(LY) YPW=Y1-Y0
    IF(LX.AND.LY) GOTO 20
    IPM=3
    IF(NCHW.LE.-2) IPM=2
    CALL PLOT(XPW,YPW,IPM)
20  IF(NCHW) 30,40,50
30  IT=TEXT+0.5
    IF(IT.LT.0.OR.IT.GT.MN) WRITE(1,2)
2  FORMAT(1H,'INVALID TEXT PARAMETER IN SYMBOL')
    CALL SUBN(IT)
    RETURN

```

```

40  IF(LN.NE.2) RETURN
    IT=TEXT+0.5
    CALL SUBU(IT)
    RETURN
50  CALL SUBP(TEXT,0.0,0,NCHW,8,1)
    RETURN
    END
C**** LINE (PLOT LINE) SUBROUTINE
SUBROUTINE LINE(X,Y,N,INC,LIT,INTEQ)
INTEGER COUNT
DIMENSION X(N),Y(N)
COMMON /BLK3/FACT
N2=N
INCW=INC
IF(INCW.GT.0) GOTO 10
WRITE(1,1)
1  FORMAT(1H,'INVALID INC PARAMETER IN LINE'
*/1H,'1 WAS ASSUMED')
INCW=1
10  INCW1=INCW+1
    NO=N2-INCW1
    COUNT=NO/INCW
    IF(COUNT.GT.0) GOTO 20
    COUNT=10
    N2=COUNT*INCW+INCW1
    WRITE(1,2) N2
2  FORMAT(1H,'INVALID N PARAMETER IN LINE'
*/1H,'12 WAS ASSUMED')
20  N1=NO+1
    XFIRST=X(N1)
    XDELTA=X(N2)
    IF(ABS(XDELTA).LT.1.0E-6) XDELTA=1.0
    YFIRST=Y(N1)
    YDELTA=Y(N2)
    IF(ABS(YDELTA).LT.1.0E-6) YDELTA=1.0
    XS=(X(1)-XFIRST)/XDELTA
    YS=(Y(1)-YFIRST)/YDELTA
    CALL PLOT(XS,YS,3)
    IF(LIT.NE.0) GOTO 40
    DO 30 I=INCW1,NO,INCW
    XS=(X(I)-XFIRST)/XDELTA
    YS=(Y(I)-YFIRST)/YDELTA
30  CALL PLOT(XS,YS,2)
    RETURN
40  H=0.21*FACT
    CALL SUBS(H)
    CALL SUBN(INTEQ)
    INCLT=INCW*IABS(LIT)
    DO 50 I=INCW1,NO,INCW
    XS=(X(I)-XFIRST)/XDELTA
    YS=(Y(I)-YFIRST)/YDELTA
    IF(LIT.LT.0) GOTO 60

```

```

      CALL PLOT(XS,YS,2)
      GOTO 70
60 CALL PLOT(XS,YS,3)
70 II=I-1
   IF(MOD(II,INCLT).NE.0) GOTO 50
   CALL SUBN(INTEQ)
50 CONTINUE
   RETURN
   END
C**** NUMBER (PLOT NUMBER) SUBROUTINE
SUBROUTINE NUMBER(XP,YP,HE,NUMB,ANG,NDEC)
  REAL NUMB
  LOGICAL LX,LY
  COMMON /BLK1/XO,YO/BLK2/X1,Y1/BLK3/FACT
  H=FACT*HE
  IF(H.GT.0.0) GOTO 10
  WRITE(1,1)
1  FORMAT(1H,'INVALID HEIGHT PARAMETER IN NUMBER'/
  *1H,'0.28 WAS ASSUMED')
  H=0.28
10 CALL SUBS(H)
   CALL SUBG(ANG)
   XPW=XP
   YPW=YP
   ND=NDEC
   LX=XPW.GE.9999.0
   LY=YPW.GE.9999.0
   IF(LX) XPW=X1-X0
   IF(LY) YPW=Y1-Y0
   IF(LX.AND.LY) GOTO 20
   CALL PLOT(XPW,YPW,3)
20 IF(ND.LT.0) GOTO 50
   IF(ND.GT.6) ND=6
   NUM=NUMB
   NI=8-ND
   DO 30 N=1,NI
     NUM=NUM/10
   IF(IABS(NUM).EQ.0) GOTO 40
30 CONTINUE
   N=NI
40 NW=N+ND+2
   ND1=ND+1
   CALL SUBP(0.0,NUMB,0,NW,ND1,2)
   RETURN
50 NUMB=NUMB
   IF(ND.EQ.-1) GOTO 60
   NUMB=NUMB/10.0**(-ND-1)
60 NUM=NUMB
   DO 70 N=1,6
     NUM=NUM/10
   IF(IABS(NUM).EQ.0) GOTO 80
70 CONTINUE

```

```

      N=6
80 N1=N+1
   CALL SUBP(0.0,0.0,NUMB,N1,8,3)
   RETURN
   END
C**** AXIS (PLOT AXIS) SUBROUTINE
SUBROUTINE AXIS(XP,YP,TEXT,NCH,AXLEN,ANGLE,FIRSTV,DELTA)
  REAL*8 TEXT
  REAL NUMB
  NCHW=NCH
  AXL=AXLEN
  DELTA=DELTA
  IF(NCHW.NE.0) GOTO 10
  WRITE(1,1)
1  FORMAT(1H,'INVALID NCH PARAMETER IN AXIS'/
  *1H,'1 WAS ASSUMED')
  NCHW=1
10 IF(AXL.GT.0.0) GOTO 20
  WRITE(1,2)
2  FORMAT(1H,'INVALID AXISLENGTH PARAMETER IN AXIS'/
  *1H,'10 WAS ASSUMED')
  AXL=10.0
20 IF(ABS(DELTA).GT.1.0E-6) GOTO 30
  WRITE(1,3)
3  FORMAT(1H,'INVALID DELTAV PARAMETER IN AXIS'/
  *1H,'1 WAS ASSUMED')
  DELTA=1.0
30 CALL PLOT(XP,YP,3)
   NCHWA=IABS(NCHW)
   IAXL=AXL
   IAXL1=IAXL+1
   FIRST=FIRSTV
   ANG=ANGLE/90.0
40 IF(ANG.GE.-0.5.AND.ANG.LT.1.5) GOTO 50
   ANG=ANG-SIGN(2.0,ANG)
   GOTO 40
50 I=0
   IW=I
60 DA=ABS(DELTA)
   IF(DA.GT.0.01.AND.DA.LT.100) GOTO 70
   I=I+1
   IW=I
   IF(DELTA.GT.100.0) IW=-IW
   FIRST=FIRST*10.0**IW
   DELTA=DELTA*10.0**IW
   GOTO 60
70 N=I
   IF(DELTAV.LE.0.01) N=-N
   IF(ANGLE.GE.0.5) GOTO 100
   YPO=YP-0.15
   CALL PLOT(XP,YPO,2)
   CALL PLOT(XP,YP,3)

```

```

DO 80 I=2, IAXL1
XP1=XP+I-1
YP1=YP
CALL PLOT(XP1,YP1,2)
XP2=XP1
YP2=YP1-0.15
CALL PLOT(XP2,YP2,2)
80 CALL PLOT(XP1,YP1,3)
IAXL1=IAXL/2+1
DO 90 I=1, IAXL1
XP1=2.*(I-1)+XP-0.63
YP1=YP-0.63
NUMB=FIRST+DELTA*(I-1)*2
90 CALL NUMBER(XP1,YP1,0.21,NUMB,0.0,2)
YP1=YP-1.12
IF(NCHW.GE.0) YP1=YP+0.84
XP1=XP+(AXL-NCHWA*0.21)/2.0
IF(N.NE.0) XP1=XP1-6*0.21/2.0
CALL SYMBOL(XP1,YP1,0.28,TEXT,0.0,NCHWA)
IF(N.EQ.0) RETURN
XP1=XP1+0.28*NCHWA
CALL SYMBOL(XP1,YP1,0.28,' *10',0.0,4)
XP1=XP1+0.28*4
YP1=YP1+0.21
NUMB=N
CALL NUMBER(XP1,YP1,0.14,NUMB,0.0,-1)
RETURN
100 XP0=XP-0.15
CALL PLOT(XP0,YP,2)
CALL PLOT(XP,YP,3)
DO 110 I=2, IAXL1
XP1=XP
YP1=YP+I-1
CALL PLOT(XP1,YP1,2)
XP2=XP1-0.15
YP2=YP1
CALL PLOT(XP2,YP2,2)
110 CALL PLOT(XP1,YP1,3)
IAXL1=IAXL/2+1
DO 120 I=1, IAXL1
XP1=XP-0.42
YP1=2.0*(I-1)+YP-0.63
NUMB=FIRST+DELTA*(I-1)*2
120 CALL NUMBER(XP1,YP1,0.21,NUMB,90.0,2)
XP1=XP-0.84
IF(NCHW.LT.0) XP1=XP+1.12
YP1=YP+(AXL-NCHWA*0.21)/2.0
IF(N.NE.0) YP1=YP+(AXL-(NCHWA+6)*0.21)/2.0
CALL SYMBOL(XP1,YP1,0.28,TEXT,90.0,NCHWA)
IF(N.EQ.0) RETURN
YP1=YP1+0.28*NCHWA
CALL SYMBOL(XP1,YP1,0.28,' *10',90.0,4)

```

```

XP1=XP1-0.21
YP1=YP1+0.28*4.0
NUMB=N
CALL NUMBER(XP1,YP1,0.14,NUMB,90.0,-1)
RETURN
END
C**** PFIELD (PLOT FIELD DEFINITION) SUBROUTINE
SUBROUTINE PFIELD(PLLX,PLLY,PURX,PURY,SCI)
INTEGER SM,SCI
COMMON /BLK1/XO,YO/BLK4/SM
*/BLK5/XMIN,YMIN,XMAX,YMAX/BLK6/XLOW,YLOW,XUPP,YUPP
IF(SCI.GE.0.AND.SCI.LE.2) GOTO 10
WRITE(1,1)
1 FORMAT(1H,'INVALID SCISSORMODE PARAMETER IN PFIELD'/
*1H,'THIS SUBROUTINE IS IGNORED')
RETURN
10 IF(PLLX.LT.PURX.AND.PLLY.LT.PURY) GOTO 20
WRITE(1,2)
2 FORMAT(1H,'INCORRECT SETTING OF PLOT FIELD'/
*1H,'PFIELD SUBROUTINE IS IGNORED')
RETURN
20 SM=SCI
XLOW=XO+PLLX
YLOW=YO+PLLY
XUPP=XO+PURX
YUPP=YO+PURY
IF(XLOW.LT.XMIN) XLOW=XMIN
IF(YLOW.LT.YMIN) YLOW=YMIN
IF(XUPP.GT.XMAX) XUPP=XMAX
IF(YUPP.GT.YMAX) YUPP=YMAX
RETURN
END
C**** RDSCIS (REDEFINE SCISSORING MODE) SUBROUTINE
SUBROUTINE RDSCIS(SCI)
INTEGER SM,SCI
COMMON /BLK4/SM
IF(SCI.GE.0.AND.SCI.LE.2) GOTO 10
WRITE(1,1)
1 FORMAT(1H,'INVALID SCISSORMODE PARAMETER IN RDSCIS'/
*1H,'THIS SUBROUTINE IS IGNORED')
RETURN
10 SM=SCI
RETURN
END
C**** SCALE (DATA SCALING) SUBROUTINE
SUBROUTINE SCALE(DATA,AXLEN,N,INC)
INTEGER COUNT
DIMENSION DATA(N)
N2=N
INCW=INC
AXL=AXLEN
IF(AXL.GT.0.0) GOTO 10

```

```

WRITE(1,1)
1 FORMAT(1H,'INVALID AXISLENGTH PARAMETER IN SCALE'/
*1H,'10 WAS ASSUMED')
AXL=10.0
10 IF(INCW.NE.0) GOTO 20
WRITE(1,2)
2 FORMAT(1H,'INVALID INC PARAMETER IN SCALE'/
*1H,'1 WAS ASSUMED')
INCW=1
20 IINCW=IABS(INCW)
INCW1=IINCW+1
NO=N2-INCW1
COUNT=NO/IINCW
IF(COUNT.GT.0) GOTO 30
COUNT=10
N2=COUNT*IINCW+INCW1
WRITE(1,3)N2
3 FORMAT(1H,'INVALID N PARAMETER LINE'/
*1H,'I2,' WAS ASSUMED')
30 N1=NO+1
DMAX=DATA(1)
DMIN=DMIN
DO 40 I=INCW1,NO,IINCW
DATAW=DATA(I)
IF(DMAX.LT.DATAW) DMAX=DATAW
IF(DMIN.GT.DATAW) DMIN=DATAW
40 CONTINUE
J=0
DEL1=(DMAX-DMIN)/AXL
IF(DEL1.LT.1.0E-6) DEL1=1.0
50 IF(DEL1.GE.1.0) GOTO 60
DEL1=DEL1*10.0
J=J+1
GOTO 50
60 IF(DEL1.LT.10.0) GOTO 70
DEL1=DEL1/10.0
J=J+1
GOTO 60
70 DO 80 I=1,3
IW=I
DEL2=2.0**IW
IF(DEL1.LT.DEL2) GOTO 90
80 CONTINUE
IW=9
DEL2=10.0
90 IF(IW.NE.3) GOTO 110
IF(DEL1.LT.5.0) GOTO 100
IW=4
GOTO 110
100 DEL2=5.0
110 IF(INC.LT.0) DEL2=-DEL2
DELTA=DEL2*10.0**J

```

```

FIRSTV=DELTA
120 IF(FIRSTV.LE.DMIN) GOTO 130
FIRSTV=FIRSTV-ABS(DELTA)
GOTO 120
130 FLASTV=FIRSTV+AXL*ABS(DELTA)
IF(FLASTV.GE.DMAX) GOTO 150
IW=IW+1
IF(IW.EQ.3) GOTO 140
DEL2=2.0*IW
GOTO 110
140 DEL2=5.0
GOTO 110
150 IF(INC.LT.0) FIRSTV=FLASTV
DATA(N1)=FIRSTV
DATA(N2)=DELTA
RETURN
END
C**** FACTOR (SET SCALING FACTOR) SUBROUTINE
SUBROUTINE FACTOR(SF)
COMMON /BLK3/FACT
FACT=SF
IF(FACT.LE.0.0) FACT=1.0
RETURN
END
C**** SET (SET ORIGIN) SUBROUTINE
SUBROUTINE SET(XI,YI)
INTEGER SM
COMMON /BLK1/X0,Y0/BLK2/X1,Y1/BLK3/FACT/BLK4/SM
*/BLK5/XMIN,YMIN,XMAX,YMAX/BLK6/XLOW,YLOW,XUPP,YUPP
X=FACT*XI
Y=FACT*YI
IF(X.GE.XMIN.AND.Y.GE.YMIN.AND.X.LE.XMAX.AND.Y.LE.YMAX)
*GOTO 10
WRITE(1,1)
1 FORMAT(1H,'ORIGIN OVER PLOT AREA')
IF(X.LT.XMIN) X=XMIN
IF(Y.LT.YMIN) Y=YMIN
IF(X.GT.XMAX) X=XMAX
IF(Y.GT.YMAX) Y=YMAX
10 CALL SUBM(X,Y)
X0=X
Y0=Y
X1=X
Y1=Y
SM=0
XLOW=XMIN
YLOW=YMIN
XUPP=XMAX
YUPP=YMAX
RETURN
END

```

```

C**** COMMAND H(HOME) SUBROUTINE
SUBROUTINE SUBH
WRITE(2,1)
1 FORMAT(1H ,1HH)
RETURN
END
C**** COMMAND D(DRAW) SUBROUTINE
SUBROUTINE SUBD(X,Y)
IX=100.0*X
IY=100.0*Y
WRITE(2,1) IX,IY
1 FORMAT(1H ,1HD,I5,1H,,I5)
RETURN
END
C**** COMMAND M(MOVE) SUBROUTINE
SUBROUTINE SUBM(X,Y)
IX=100.0*X
IY=100.0*Y
WRITE(2,1) IX,IY
1 FORMAT(1H ,1HM,I5,1H,,I5)
RETURN
END
C**** COMMAND S(ALPHA SCALE) SUBROUTINE
SUBROUTINE SUBS(H)
N=H/0.07-0.5
IF(N.GE.15) N=15
WRITE(2,1)N
1 FORMAT(1H ,1HS,I2)
RETURN
END
C**** COMMAND Q(ALPHA ROTATE) SUBROUTINE
SUBROUTINE SUBQ(S)
SA=ABS(S)
IF(S.GE.0.0) GOTO 10
N=-SA-0.5
SA=360-(N-N/360*360)
10 N=SA/90.0+0.5
N=N-N/4*4
WRITE(2,1) N
1 FORMAT(1H ,1HQ,I1)
RETURN
END
C**** COMMAND P(PRINT) SUBROUTINE
SUBROUTINE SUBP(CHARAB,CHARA4,CHARA2,WN,DN,TYPE)
REAL*8 CHARAB
INTEGER CHARA2,WN,DN,TYPE,FORM(9),CHAA,CHAF,CHAI,W(10),D(8)
DATA FORM/'( ' ,1H',' ' ,1H','P',' ' ' ' ' ' ' ' ' ' /
*,CHAA/' A' ,CHAF/' F' ,CHAI/' I' /
*,W/' 1' ,2' ,3' ,4' ,5' ,6' ,7' ,8' ,9' ,10' /
*,D/' .0' ,.1' ,.2' ,.3' ,.4' ,.5' ,.6' , ' /
GOTO (10,20,30),TYPE
10 FORM(6)=CHAA
FORM(7)=W(WN)
FORM(8)=D(8)
WRITE(2,FORM) CHARAB
RETURN
20 FORM(6)=CHAF
FORM(7)=W(WN)
FORM(8)=D(DN)
WRITE(2,FORM) CHARA4
RETURN
30 FORM(6)=CHAI
FORM(7)=W(WN)
FORM(8)=D(8)
WRITE(2,FORM) CHARA2
RETURN
END
C**** COMMAND N(MARK) SUBROUTINE
SUBROUTINE SUBN(N)
COMMON /BLK7/LN,MN
NN=N
IF(NN.LT.1.OR.NN.GT.MN) NN=1
WRITE(2,1)NN
1 FORMAT(1H ,1HN,I2)
RETURN
END
C**** COMMAND U(USER) SUBROUTINE
SUBROUTINE SUBU(IT)
M=IT/1000
N=IT-M*1000
IF(M.GT.2.OR.N.GT.255) RETURN
IF(M.LE.0) M=1
WRITE(2,1) M,N
1 FORMAT(1H ,1HU,I1/I1H ,1HP,A1)
RETURN
END

```