

JAERI - M
86-023

汎用図形処理ユーティリティ“GPREP”と“LIBGR”

1986年3月

平山 俊雄・安積 正史

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

JAERI-Mレポートは、日本原子力研究所が不定期に公刊している研究報告書です。
入手の間合わせは、日本原子力研究所技術情報部情報資料課（〒319-11茨城県那珂郡東海村）あて、お申しこしてください。なお、このほかに財団法人原子力弘済会資料センター（〒319-11茨城県那珂郡東海村日本原子力研究所内）で複写による実費頒布をおこなっております。

JAERI-M reports are issued irregularly.

Inquiries about availability of the reports should be addressed to Information Division
Department of Technical Information, Japan Atomic Energy Research Institute, Tokai-
mura, Naka-gun, Ibaraki-ken 319-11, Japan.

©Japan Atomic Energy Research Institute, 1986

編集兼発行 日本原子力研究所
印 刷 いばらき印刷㈱

汎用図形処理ユーティリティ
"GPREP" と "LIBGR"

日本原子力研究所那珂研究所臨界プラズマ研究部
平山 俊雄 · 安積 正史

(1986年1月30日受理)

トカマクコードのライブラリ・システム化の一環として、汎用的な図形処理ユーティリティを開発した。本ユーティリティルーチンは、図形データを一定の形式でファイル出力するプリプロセッサ「GPREP」と、その図形データファイルからデータを読み取り図形出力をする「LIBGR」から構成される。「GPREP」により作成されたデータファイルは、書き込み時の情報が無くても読み出すことができるようになっており、異なる属性のデータでこん然一体となった大量のデータを処理する上で極めて有効である。

Graphic Utilities "GPREP" and "LIBGR"

Toshio HIRAYAMA and Masafumi AZUMI

Department of Large Tokamak Research
Naka Fusion Research Establishment
Japan Atomic Energy Research Institute
Naka-machi, Naka-gun, Ibaraki-ken

(Received January 30, 1986)

Plotting utilities, which are widely appreciable to problems of computational physics, have been developed as a part of a library system for a one dimensional tokamak transport code. Utility routines are composed of two procedures, i.e., the pre-processor "GPREP" registrates graphic data to a data file with a set format, and "LIBGR" processes the data to match with the required format of "ARGUS" and plotts the graph. The regular form of data by "GPREP" enables us to read out data without any information on data recording so that these utilities are very usefull for dealing with a lot of disorder data with different type.

Keywords: Library System, One Dimensional Tokamak Transport Code,
GPREP, LIBGR, ARGUS, Graphic Utility

目 次

1. はじめに	1
2. GPREP	2
2.1 GPREPの機能	2
2.2 データファイル構造	3
2.3 プロットコマンド	5
2.3.1 DEFINE ブロック	5
2.3.2 PLOT ブロック	7
2.4 一般プログラムにおけるGPREPの利用	9
コマンドプロシジャ @GPREP I	
3. LIBGR	14
3.1 LIBGRの機能	14
3.2 図形指示データ	14
3.3 CDS データ	17
3.4 LIBGRの使用例	48
4. ま と め	65
謝 辞	65
参考文献	65

Contents

1. Introduction	1
2. GPREP	2
2.1 Function of GPREP	2
2.2 Structure of data file	3
2.3 Plot command	5
2.3-1 DEFINE block	5
2.3-2 PLOT block	7
2.4 Usage of GPREP in user programs	9
Command procedure of @GPREP1	
3. LIBGR	14
3.1 Function of LIBGR	14
3.2 Graphic format data	14
3.3 CDS data	17
3.4 Examples of usage of LIBGR	48
4. Summary	65
Acknowledgements	65
References	65

1. はじめに

計算機による結果を図として表示することは、その結果について人が判断する上で大きな助けとなる。そのためには、ユーザの作り出す様々なコードの計算結果を容易に図形出力する手段が必要となるが、多種多様な計算コード毎に図形出力ルーチンを作成することは大きな負担となる。ここでは、上記の問題を解決するために、汎用的な図形処理ユーティリティを開発した。ユーティリティルーチンは、ユーザプログラムから図形出力データをファイルに出力する部分と、そのファイルから図形データを読み取り図形として出力する部分からなる。図形データをユーザプログラムから切り離し、図形データファイルとして出力するのは、ユーザプログラムから図形データをファイルに出力する形式を一定に決めておくことにより、図形データファイルを読み込み図形表示を行うプログラムをユーザプログラムと独立に作っておくことができるからである。又、作成された図形データファイルは、ユーザプログラムに関する知識なしに読み出せるので、古いデータファイルの中が読めないといった問題も解決できる。

図形データを一定の形式で出力するユーティリティルーチンを「GPREP」(Graphic Pre-processor)という。図形データファイルから、データを読み取り、図形指示データと共に汎用図形出力ルーチン「ARGUS」¹⁾にデータを受渡すユーティリティルーチンが「LIBGR」(Library Graphic Utility)である。以下の節では、「GPREP」及び「LIBGR」について述べる。

2. GPREP

2.1 GPREP の機能

GPREP は図形データファイルの作成を支援する目的で作られた、FORTRAN のプリプロセッサである。ユーザーは、自分の作成した FORTRAN プログラムの中に GPREP の提供する数種類の「プロットコマンド」を用いて、図形として出力したい変数、配列の群を指定し、又出力する時刻を指定する。「プロットコマンド」は GPREP プリプロセッサによって FORTRAN 文に変換され、図形データ出力ルーチンを構成する。図にユーザプログラムと図形処理ユーティリティの関係を示す。図形データ出力ルーチンが作成した図形データファイルは、ユーザプログラムに関する知識なしに読み出せる形式で書かれているため、ユーザーは図形データのファイル構造に関与する必要はない。

トカマクコード・ライブラリシステム²⁾では、本プリプロセッサは既存の処理プログラム「ORG」の中に組込まれているので、ライブラリシステムのユーザーは「プロットコマンド」を用いて、任意のデータを図形データとして出力することができる。ライブラリシステム以外で使用する場合は、「プロットコマンド」を FORTRAN 文に変換するマクロコマンド「@GPREP 1」が用意されている。

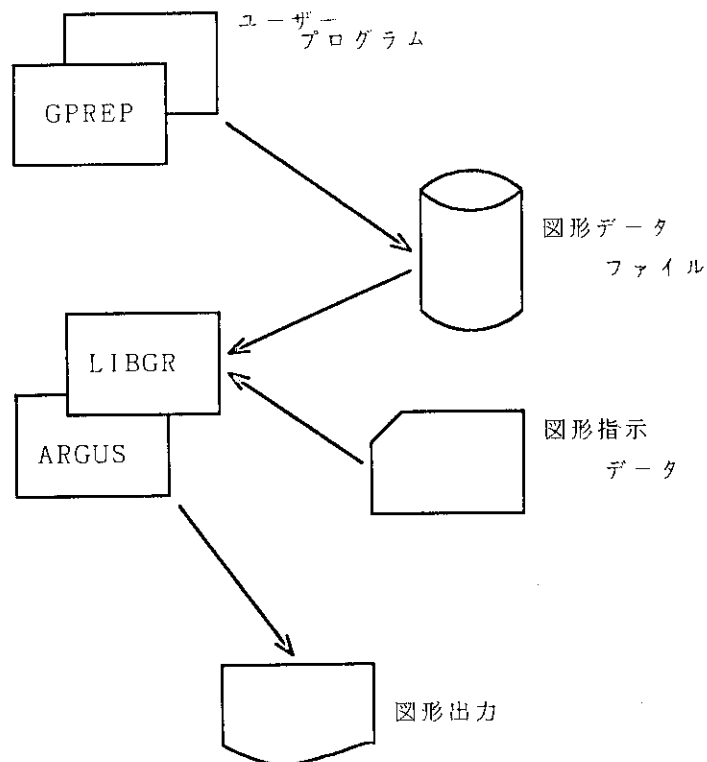


図1 ユーザプログラムと図形処理ユーティリティの関係

2.2 データファイル構造

図形データファイルは書式なしの順編成ファイルである。入出力されるデータの基本形式は、2レコードよりなるバイナリーデータである。レコードは可変長であり、

第1レコード ; 制御レコード

第2レコード ; データレコード

とする。各レコードの構成は以下のようになる。

第1レコード

識別子	属性	次元数 m	サイズ 1	-----	サイズ n
-----	----	-------	-------	-------	-------

第2レコード

データ					
(サイズ 1) * (サイズ 2) * --- * (サイズ n)		* (サイズ 1) * (サイズ 2) * --- *			
1次元		2次元			
(サイズ n) * ----- *		(属性のサイズ)			
m次元					

識別子 ; 変数

属性 ; A * 4, I * 4, R * 4, -----

図形データファイルは図2に示すような「HEAD」と「BODY」からなる階層構造を持っている。図中に枠付きで表示されている部分は、第1レコード、及び第2レコード部分を出力する具体的な要素に対応している。

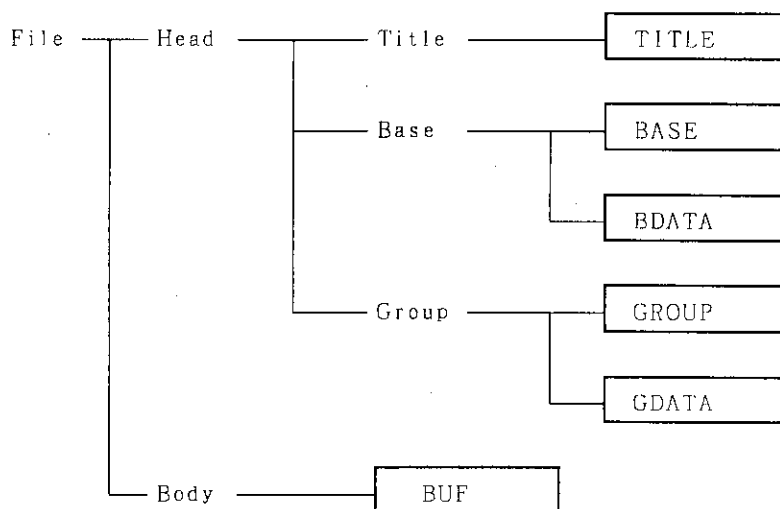


図2 図形データファイルの構造

「HEAD」はファイル名、独立変数群に関する情報、従属変数群に関する情報が書かれるヘッダー部である。ヘッダー部分の書き込みは、ユーザプログラムの初期化時に1回だけ行われる。

「BODY」はユーザプログラムで計算された値が、グループ毎にバッファリングされてファイルに出力されるボディ部分である。シミュレーションコード等では、このボディ部分が、ヘッダー部分の後にくり返し出力されることになる。

「HEAD」は「TITLE」、「BASE」そして「GROUP」の3つの部分から構成される。「TITLE」はファイルの名前に対応するメッセージであり、それ自体が任意の文字列からなる1組のレコードである。「BASE」は独立変数の群を定義する部分であって、独立変数として登録すべき変数群についての情報及び値を含む。独立変数の数については、要素 **BASE**，又独立変数のサイズ、値については、要素 **BDATA** が登録する。

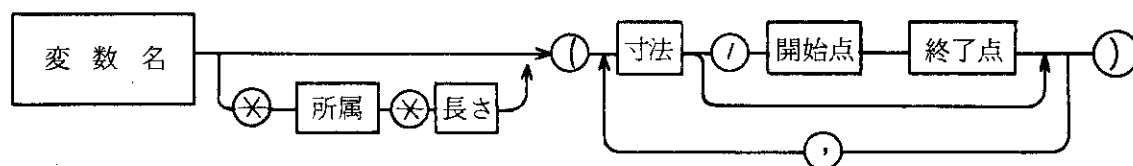
「GROUP」は従属変数の群を定義する部分で、登録するグループの数を示す要素 **GROUP** と、個別のグループに関するデータを示す要素 **GDATA** から構成される。

GDATA は、グループに属する従属変数の名前、属性、次元、そして寸法を登録する。

。 PBASE 文

`. PBASE 変数宣言子, 変数宣言子, ……`

シミュレーションコードでは、時間と独立な座標軸を取扱うことが多い。それらをあらかじめ、「BASE」という名前のグループとして定義しておくことにより、多重の出力を避けることができる。座標軸変数は一次元配列とする。変数宣言子は、独立変数（座標軸変数）として使用される一次元配列を定義するもので、最大 10 個まで許される。変数宣言子とは以下の構造を指す。



属性は、A, I, R のいずれかで、文字型、整数型、実数型を示す。

長さは、4, 8, 又は 16 のいずれかで、バイト長を示す。

例)

`. PBASE DR * R * 4 (MSH)`

DR は一次元配列 DR (MSH) である。

PBASE 文は無くても良いが、書く場合は必ず PTITLE 文の直後でなくてはならない。

。 PGROUP 文

`. PGROUP 文 グループ名, 変数宣言子, 変数宣言子`

図形データファイルに出力する変数の組を定義する。グループ名はFORTRANの変数名に準ずる。1つのプログラム単位で最大 20 個のグループを定義できる。1つのグループは最大 50 個の変数宣言子を含むことができる。変数宣言子の内容は PBASE 文のそれと同じである。

例)

```
C
.PGROUP A, DETB*R*8(CNRNETB,NTNETB),RNETB*R*8(CNRNETB),
+      TNETB*R*8(CNTNETB),
+      TETB*R*8(CNRTETB,NTTETB),RTETB*R*8(CNRTETB),
+      ITETB*R*8(CNTTETB),
+      DITB*R*8(CNRNITB,NTNITB,NB),RNITB*R*8(CNRNITB),
+      TNITB*R*8(CNTNITB),
+      TITB*R*8(CNRTITB,NTTITB,NB),RTITB*R*8(CNRTITB),
+      TTITB*R*8(CNTTITB),
+      VXTB*R*8(CNVXTB),TVXTB*R*8(CNVXTB)
C
```

例 - 1

◦ DEFEND 文

DEFEND

DEFINE ブロックの終了を示す。

以下に DEFINE ブロック全体の例を示す。

例)

```

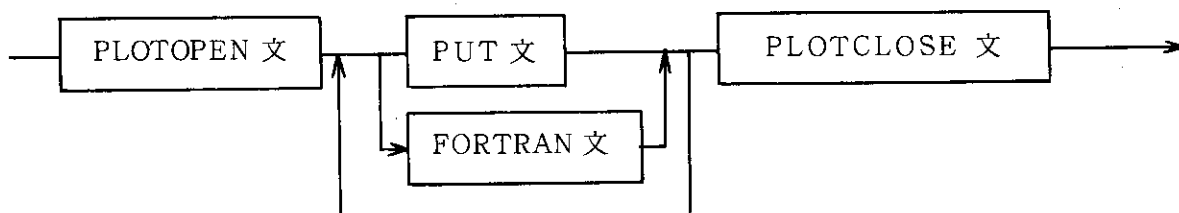
C -----
C
C      SUBROUTINE WPINIT
C -----
C
C
C .DEFINE NPLOT1
C .PTITLE '%%%%%%%%% PLOT DATA %%%%%%%%%%'
C .PBASE R**8(NP1)
C *****
C *      TIME DATA      *
C *****
C .PGROUP A,T**8,AVNE**8,AVNI**8,AVTE**8,AVTI**8,AVLNE**8(NP1),
+      ALI**8,CLI**8,RESIST**8,ZIP**8,VOLP**8
C
C .PGROUP B,T**8,TAUEL**8(NP1),TAU**8(NP1,NB0),TAUE**8,TAUX**8,
+      TAUEE**8,TAUEI**8,TAUXE**8,TAUXI**8,TPCON**8
C *****
C *      SPACE DATA      *
C *****
C .PGROUP C,T**8,DEP**8(NP1),DP**8(NP1,NB),
+      TEP**8(NP1),TP**8(NP1,NB)
C
C .PGROUP D,T**8,SCN**8(NP1,NB),GDET**8(NP1),GTET**8(NP1),
+      GDTN**8(NP1,NB),GTTN**8(NP1,NB)
C *****
C .DEFEND
C
C      RETURN

```

例 - 2

2.3.2 PLOT ブロック

PLOTOPEN 文から PLOTCLOSE 文までを、PLOT ブロックと呼び、以下に示す構成からなる



PLOT ブロックの中には、PUT 文、及びFORTRAN 文が許される。ブロック全体は1つのサブプログラム中に含まれ、全ての宣言文の後に置かれる。

◦ PLOTOPEN 文

```
.PLOTOPEN  n
```

PLOT ブロックの開始を示す。nは数字列、又は変数名であり、図形データファイルの出力機番を示す。nの値はDEFINE 文での値と等価でなくてはいけない。

◦ PUT 文

```
.PUT  グループ名
```

あるグループの変数を出力する指示である。グループ名はDEFINE ブロックで定義されていなくてはいけない。

◦ PLOTCLOSE 文

```
.PLOTCLOSE
```

PLOT ブロックの終了を示す。

以下に PLOT ブロック全体の例を示す。

例)

```

C ----- TIME DATA -----
C
C   ENTRY WPLT1
C
C   .PLOTOPEN NPLOT1
C   .PUT A
C   .PUT B
C   .PLOTCLOSE
C
C   RETURN
C
C ----- SPACE DATA -----
C
C   ENTRY WPLT2
C
C   .PLOTOPEN NPLOT1
C   .PUT C
C   .PUT D
C   .PLOTCLOSE
C
C   RETURN
C
C   END
    
```

2.4 一般プログラムにおける GPREP の利用

コマンドプロシジャ “ @ GPREP 1 ”

ライブラリ・システムのユーザーはプリプロセッサ「GPREP」をシステムの提供の下に意識せずに利用できる。ライブラリ・システムを使用しない一般ユーザーのプログラムの場合は、DEFINE ブロック、及び PLOT ブロックからなる短いサブルーチンを設けることによって、それを FORTRAN 文に変換するためのコマンドプロシジャ “ @ GPREP 1 ” を利用することができる。

図3に「GPREP」の使用例を示す。メインプログラム中において、「CALL INIT」及び「CALL OUT」がそれぞれ HEADER 部分、及び BODY 部分の書き出しを行う。サブルーチン「INIT」はプロットコマンドで記述された DEFINE ブロックからなり、そのエントリとして「OUT」があり、PLOT ブロックを構成している。サブルーチン名は FORTRAN の規約に従ったものであるが、そのメンバー名は先頭に「@GR」を付加した名前になっていなければならない。すなわち、

メンバー名 : @GRINIT

サブルーチン名 : INIT

となることが必要である。コマンドプロシジャ “ @ GPREP 1 ” をこのプログラムに対して機動させると、「@GR」のついたメンバーを FORTRAN 文に展開し、そこで呼ばれるルーチンを追加した後、「@@@@ OUT. FORT 77」という PS ファイルにプログラムソースを出力する。

使用方法と出力ファイルのリストを以下に示す。コマンドプロシジャ “ @ GPREP 1 ” を機動させると、ユーザープログラム名を尋ねてくるのでその名前を入力する。

例)

```
EX  J 2855. TSSMAC. CLIST (@ GPREP 1)
    → TEST. FORT
```

ここで、ユーザープログラム名は「TEST. FORT」となっている。

図4に FORTRAN に展開された「@@@@ OUT. FOR77」のリストを示す。図4-aは「INIT」と「MAIN」を表わし、「INIT」のプロットコマンドで書かれた DEFINE ブロックと PLOT ブロックが展開されている様子を示している。図4-b, c はプリプロセッサによって新たに追加されたサブルーチンであるが、ユーザはこれらのサブルーチンに参与する必要はない。

```

C*****
C*   TEST PROGRAM TO SHOW THE USAGE OF "GPREP"
C*
C*****
COMMON /AAA/ Y(50), X(50)
COMMON /BBB/ MESH

C
MESH = 50
C-----
CALL INIT                                } HEADERの書き出し
C-----
C
DO 100 I=1, MESH
  X(I) = FLOAT(I)/50.
  Y(I) = ( 1.0 - X(I)**2 )
100 CONTINUE
C
C-----
CALL OUT                                } BODYの書き出し
C-----
C
STOP
END

C*****
C*   SUBROUTINE FOR INITIALIZING
C*
C*****
SUBROUTINE INIT
COMMON /AAA/ Y(50), X(50)
COMMON /BBB/ MESH
INTEGER*4 F10/10/
C-----
.PLOTOPEN F10
.PTITLE ==== EXAMPLE OF USAGE OF GPREP ====
.PBASE X*R*4(50)
.PGROUP GRAPH,Y*R*4(50),MESH*I*4
.DEFEND
C-----
RETURN
END
C*****
ENTRY OUT
C*****
C
C-----
.PLOTOPEN F10
.PUT GRAPH
.PLOTCLOSE
C-----
RETURN
END

```

プロットデータの出力先 (FT 10)

DEFIFNEブロック

PLOTブロック

図3 プリプロセッサ「GPREP」の使用例


```

C*****
C=  SUBROUTINE FOR INITIALIZING          *
C=                                     *
C*****
      SUBROUTINE INIT
      COMMON /AAA/ Y(50), X(50)
      COMMON /BBB/ MESH
      INTEGER*4 F10/10/
C-----
CDEFINE F10
      INTEGER * 4 IQ0001(300),IQ0002,IQ0003
      IQ0002 = F10
CPTITLE ==== EXAMPLE OF USAGE OF GPREP ===
CPSASE X*R*4(50)
      CALL QOUT1('BASE          ','I*4 ','0,DUMMY,IQ0002,KER)
      IQ0003 = 1
      WRITE(IQ0002) IQ0003
      IQ0003 = 50-          1+1
      CALL QOUT1('X          ','R*4 ','1,IQ0003,IQ0002,KER)
      WRITE(IQ0002) (X      (L),L=
      * 1,          50)
CPRGROP GRAPH,Y*R*4(50),MESH*I*4
CDEFEND
      CALL QOUT1('GROUP          ','I*4 ','0,DUMMY,IQ0002,KER)
      IQ0003 = 1
      WRITE(IQ0002) IQ0003
      IQ0001(1) = 2
      IQ0001( 2)='Y '
      IQ0001( 3)=' '
      IQ0001( 4)='R*4 '
      IQ0001( 5)= 1
      IQ0001( 6)=          50-          1+1
      IQ0001( 7)='MESH'
      IQ0001( 8)=' '
      IQ0001( 9)='I*4 '
      IQ0001(10)= 0
      CALL QOUT1('GRAPH          ','I*4 ','1, 10,IQ0002,KER)
      WRITE(IQ0002) (IQ0001(J),J = 1, 10)
C-----
      RETUTN
      END
C*****
      ENTRY OUT
C*****
C-----
C PLOTOPEN F10
      IQ0002 =F10
CPUT GRAPH
      IQ0003 = +1*(50-1+1)+1
      CALL QOUT1('GRAPH          ','R*4 ','1,IQ0003,IQ0002,KER)
      WRITE(IQ0002)(Y(IDIM01),IDIM01=1,50),MESH
C PLOTCLOSE
C-----
      RETURN
      END
C*****
C*  TEST PROGRAM TO SHOW THE USAGE OF ""GPREP""          *
C*                                                         *
C*****
      COMMON /AAA/ Y(50), X(50)
      COMMON /BBB/ MESH
C
      MESH = 50
C-----
      CALL INIT
C-----
C
      DO 100 I=1, MESH
         X(I) = FLOAT(I)/50.
         Y(I) = ( 1.0 - X(I)**2 )
      100 CONTINUE
C
C-----
      CALL OUT
C-----
C
      STOP
      END

```

図 4 - a FORTRAN 文に変換された PLOT コマンド (@@@@OUT.FORT 77)

```

C -----
C
C      DATA FILE READ/WRITE UTILITY SUBROUTINES
C -----
C
C      SUBROUTINE QOUT(ISP,ITYPE,IDIM,ISZ,DATA,IUNIT,KEOF)
C
C      CHARACTER * 4 ITYPE
C      INTEGER * 4 ISP(3),IDIM,ISZ(1),IUNIT,KEOF
C      INTEGER * 4 STR(1)
C      REAL * 4 DATA(1)
C
C      CHARACTER * 4 KATTR(10)
C      INTEGER * 4 NATTR(10)
C      DATA KATTR, NATTR/
1  'I*4','R*4','R*8','R*16','L*4','C*4','C*8','C*16','A*4',' '
2,  1  ,  1  ,  2  ,  4  ,  1  ,  2  ,  4  ,  8  ,  1  ,  0 /
C
C      KEOF = 0
C      WRITE(IUNIT) ISP,ITYPE,IDIM,(ISZ(J),J=1,IDIM)
C
C      LENG = 1
C      DO 10 J=1,9
C          IF(ITYPE.EQ.KATTR(J)) LENG = NATTR(J)
10 CONTINUE
C      DO 20 J=1,IDIM
C          LENG = LENG * ISZ(J)
20 CONTINUE
C
C      WRITE(IUNIT) (DATA(J),J=1,LENG)
C      RETURN
C -----
C
C      Q O U T 1
C -----
C
C      ENTRY QOUT1(ISP,ITYPE,IDIM,ISZ,IUNIT,KEOF)
C
C      KEOF = 0
C      WRITE(IUNIT) ISP,ITYPE,IDIM,(ISZ(J),J=1,IDIM)
C
C      RETURN

```

図4-b 「GPREP」により追加されたユーティリティ・サブ
ルーチン。(@@@@OUT.FORT 77)

```

C -----
C
C           Q I N
C -----
C
C     ENTRY QIN(ISP,ITYPE,IDIM,ISZ,DATA,IUNIT,KEOF)
C
C     KEOF = 0
C     READ(IUNIT,END=800) ISP,ITYPE,IDIM,(ISZ(J),J=1,IDIM)
C     WRITE(6,7000      ) ISP,ITYPE,IDIM,(ISZ(J),J=1,IDIM)
7000 FORMAT(1H , 'ISP = ',3A4,' ITYPE = ',A4,' IDIM=',I13,' ISZ=',7I1
C
C     LENG = 1
C     DO 110 J=1,9
C       IF(ITYPE.EQ.KATTR(J)) LENG = NATTR(J)
110 CONTINUE
C     IF(IDIM.LE.0) GOTO 130
C       DO 120 J=1,IDIM
C         LENG = LENG * ISZ(J)
120 CONTINUE
130 READ(IUNIT,END=800) (DATA(J),J=1,LENG)
C
C     WRITE(6,7200) (DATA(I),I=1,8)
C7200 FORMAT(1H ,1P8E15.8)
C     WRITE(6,7300) (DATA(I),I=1,8)
C7300 FORMAT(1H ,8I15)
C     WRITE(6,7400) (DATA(I),I=1,8)
C7400 FORMAT(1H ,8A4)
C
C     RETURN
C -----
C
C           Q I N 1
C -----
C
C     ENTRY QIN1(ISP,ITYPE,IDIM,ISZ,DATA,IUNIT,KEOF)
C
C     READ(IUNIT,END=800) ISP,ITYPE,IDIM,(ISZ(J),J=1,IDIM)
C
C     RETURN
C -----
C
C           E O F      D E T E C T E D
C -----
C
C     800 KEOF = 1
C     WRITE(6,6300)
6300 FORMAT(' <<<  END OF DATA FILE  >>>')
C     RETURN
C     END

```

図4-c 同 上

3. LIBGR

3.1 LIBGR の機能

LIBGR は、GPREP を用いて作られた図形データファイル中のデータを、ARGUS を用いて図形出力させるためのユーティリティである。図 1 に使用概念図を示す。図形指示データは図形出力したい変数を指示するデータ群と、画面構成に必要な情報群（CDS データ：画面名、コメント等）、及び画面形式を決定する表示コマンドからなる。

作図機能としては、ARGUS の機能の中から使用頻度の高いもの限定して用いている。図の種類は、一次元図、斜投図、等高線図、及び鳥瞰図が可能である。

3.2 図形指示データ

図形指示データの例を以下に示す。

```
&GNAME PFL1
&GSORT 1
&BASE R
&PARM T(1),T(2),T(3)
&FUNC DEP(R),TEP(R),DP(R,1),PEIMP(R)
&CDS
*SIDE CM
*MAINCM ICM='JFT2M OXYGEN (1%) ; DA=0.4,CV=1.0,CDB=3.0,CVS=3.0&'
*SUBCM ICM='TE, NE, NI AND RADIATION&'
*XAXIS ICM='R "43" M "49"&'
*MLTSCL N1=1,A1=0.0,A2=1.0E3
*MLTSCL N1=2,A1=0.0,A2=5.0E19
C*MLTSCL N1=3,A1=0.0,A2=5.0E4
*ZAXSND N1=3
*MLTAXS N1=1,ICM='TE "43" EV "49"&'
*MLTAXS N1=2,ICM='NE,NI "43"1"51"M"48,48"3"49"&'
*MLTAXS N1=3,ICM='PR "43"W"51"M"48,48"3"49"&'
*ATTACH N1=1,ICM='2&'
*ATTACH N1=2,ICM='1,3&'
*ATTACH N1=3,ICM='4&'
*KLINE IV1=(1/1,2/2,3/3,4/4)
*APPROX IV1=(1/0,2/0,3/0,4/0)
*DATAPT IV1=(1/101,2/102,3/103,4/104),
IV2=(1/-99,2/-99,3/-99,4/-99)
*OUT ONEDIM
&&&
```

図形指示データは、「&」のついたコマンドがキーワードになっており、画面名のキーワード（「&GNAME」）からエンドマーク（「&&&」）までを一組として、任意組入力できる。「&CDS」から「&&&」までの間には、CDS コマンド及び図形表示コマンドが入力されるとその時点でのCDS コマンドに基づいて画面出力が行われる。以下に、各キーワード毎に説明する。

1) 画面名「& GNAME」

意味 画面に付けられる名前

形式 キーワード「GNAME」を1カラムから左づめに書く。1カラムの空白をあけて、画面名として英字で始まる6文字以内の英数字を空白をあけずに書く。行の継続はできない。

```
& GNAME 1 PROFIL
```

2) 画面形式「& GSORT」

意味 1次元図か2次元図かを指示する数字。

1 1次元図

2 2次元図

形式 キーワード「& GSORT」を1カラムから左づめに書く。1カラムの空白をあけて、1又は2の数字を書く。行の継続はできない。

```
& GSORT 1
```

3) 独立変数名リスト「& BASE」

意味 関数の台（定義域）となる変数名、又は配列名のリストで最大2つまでとする。図形のX-座標軸は、独立変数が一般に選ばれる。そのような独立変数については、出力の重複を避けるために「BASE」という名前のグループとして定義する。

形式 キーワード「& BASE」を1カラムから左づめで書く。1カラムの空白をあけて、変数名又は配列名、及びそれらのインデックスの範囲を明示したものを1組として、コマンドで区切って並べる。

行を継続する時はコマンドで行を区切り、次の変数名から新しい行に移る。継続行の第1カラムは空白とし、第2カラム以降から書き始める。

```
& GBASE 変数名1 ((n1:n2)) [, 変数名2]
```

ここで、 $n1 \geq 1$, $n2 \geq 1$ かつ $n2 \geq n1$ である。範囲の区切りは「:」である。配列の全てを指定する場合はインデックスの範囲指定を省略できる。記号〔……〕は、その部分が省略できることを表わす。

4) パラメータ変数名リスト「& PARM」

意味 同一の独立変数と従属変数の組に対して、別のパラメータの値を変えて複数の画面を描く時に、パラメータとする変数を指定する。

形式 キーワード「& PARM」を1カラムから左づめで書く。1カラムの空白をあけて、変数名又は配列名を書く。インデックスの範囲指定があれば「& BASE」と同様に明示する。パラメータとして指定できるのは1変数である。不要の場合はキーワードも含めて省略する。行の継続はできない。

```
& PARM 変数名 ((n1:n2))
```

5) 従属変数名リスト「& FUNC」

意味 「& BASE」で指定した独立変数上で定義されている関数の値に対応した変数名又は配列名を指定する。

形式 キーワード「& FUNC」を1カラムから左づめで書く。1カラムの空白をあけて変数名又は配列名、カッコに狭んで対応する独立変数名、及びインデックスの値の範囲をコマンドで区切って並べたものを1組として書く。行の継続は「& GBASE」と同じである。

```
& FUNC 変数名1(独立変数名) [, 変数名2(独立変数名)] [, ..... ]
& FUNC 変数名1(n1, 独立変数名) [, 変数名2(n2, 独立変数名)] [, ...
..... ]
& FUNC 変数名1(独立変数名1, 独立変数名2) [, ..... ]
& FUNC 変数名1(独立変数名, n1 : n2) [, 変数名2(n3 : n4, 独立変数
名), ..... ]
```

6) CDS データ群「& CDS」

意味 画面に書込むコメント、軸の種類等の情報。

形式 キーワード「& CDS」を1カラムから左づめで書く。CDS データは第1カラムを「*」とするキーワード形式のデータであり、規則は「&」キーワード文と同である。CDS キーワードとその内容については、3.2節で述べる。

7) エンドマーク「&&&」

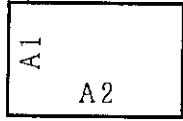

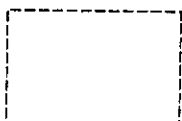
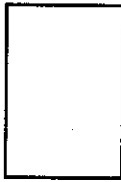
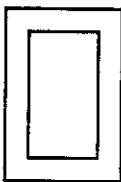

意味 1組の画面についてのデータの終りを示す。

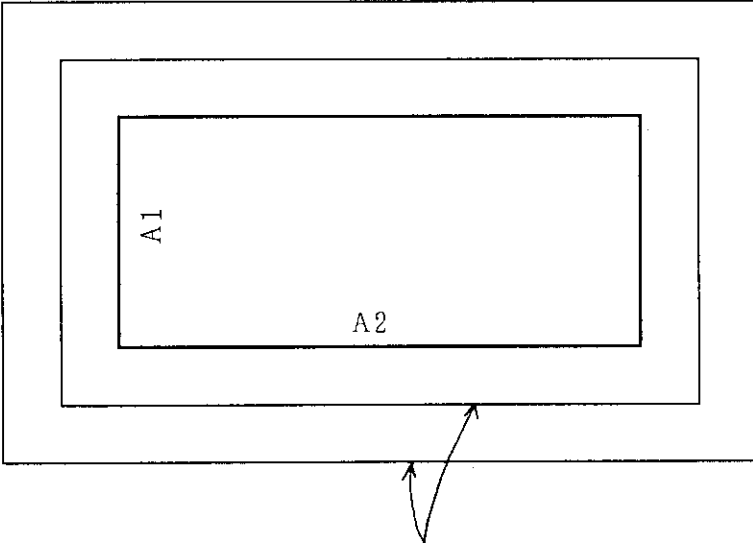
形式 キーワード「&&&」を第1カラムから左づめに書く。

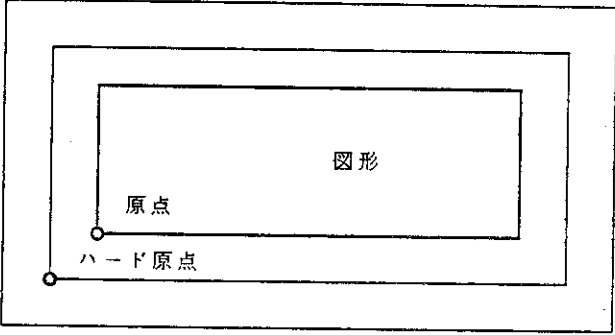
```
&&&
```

3.3 CDSデータ

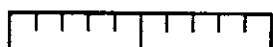
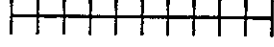
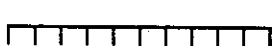
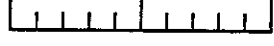
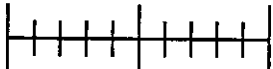
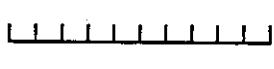
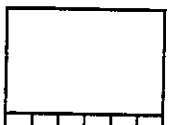
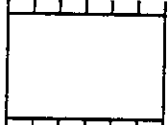

1	* MAINCM * SUBCM	概 要	主コメント内容の指定 サブコメント内容の指定
入力形式とデフォルト値			
* MAINCM ICM = ▼ &▼ 主コメント			
* SUMCM ICM = ▼ &▼ サブコメント			
パラメータの説明			
(1) ICM 出力したいコメントの内容 i) コメントの内容は、▼で括り文字形入力形式とする。 ii) コメント内容の最後は必ず「&」の識別子で終わりを示す。 iii) 表1のシンボルコード表にあるもののみ表示内容とする。 iv) ・英文字、数字は直接記述する。 ・英小文字は▼▼(ダブルコーテーション)で括り英大文字で直接記述する。 ・上記以外のシンボル、ギリシャ文字などは、コード表のコード番号を▼▼で括り記述する。			
備 考			
(1) ONEDIMとONED3D, CONTORとBRDEYEはそれぞれコメント内容を共通にしている。もし、異なる内容にしたい場合には : * MAINCM ICM = ▼000&▼ * OUT ONEDIM : の如く、表示コマンドデータ(* OUT)の前でそれぞれ指定すればよい。			
(例) ICM = ▼EXAMPLE ▼▼ABC ▼▼DEF ▼▼101 ▼▼GH 234 ▼▼121, 122 ▼▼XYZ&▼ のとき出力は次の如くなる。 EXAMPLE abc DEF ◻ GH 1234 α β XYZ			

2	* SCSIZE	概要	作画スクリーンの大きさ, 種類の指定
<p>入力形式とデフォルト値</p> <p>* SCSIZE N1 = 1, N2 = 1, A1 = 297.0, A2 = 210.0</p>			
<p>パラメータの説明</p> <p>(1) N1 スクリーンの大きさオプション</p> <p style="margin-left: 40px;">= 1 A 4判 = 2 B 5判</p> <p style="margin-left: 40px;">= 3 A 3判 = 4 B 4判</p> <p style="margin-left: 40px;">= 0, 又は上記以外 A1, A2で指定した大きさ</p> <p>(2) N2 スクリーンの種類 (縦長, 横長及び外枠, 内枠の種類)</p> <div style="display: flex; justify-content: space-around; align-items: flex-start; margin-top: 20px;"> <div style="text-align: center;"> <p>= 1</p>  <p>外枠</p> </div> <div style="text-align: center;"> <p>= 2</p>  <p>外枠, 内枠</p> </div> <div style="text-align: center;"> <p>= 3</p>  <p>枠ナシ</p> </div> </div> <div style="display: flex; justify-content: space-around; align-items: flex-start; margin-top: 20px;"> <div style="text-align: center;"> <p>= 4</p>  </div> <div style="text-align: center;"> <p>= 5</p>  </div> <div style="text-align: center;"> <p>= 6</p>  </div> </div> <p>(3) A1 スクリーンの水平方向の長さ (mm 単位)</p> <p>(4) A2 スクリーンの垂直方向の長さ (mm 単位)</p> <p>(注) A1, A2は, N1 = 1~4の時はダミーとなるので指定しても無意味</p>			

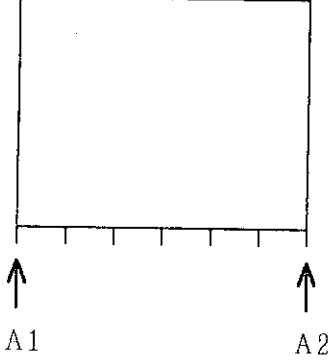
3	* GRSIZE	概要	図の大きさ, 図のフレームの指定
<p>入力形式とデフォルト値</p> <p>* GRSIZE N1 = 1, A1 = 200.0, A2 = 150.0</p>			
<p>パラメータの説明</p> <p>(1) N1 図のフレームの枠のオプション = 1 枠を書く = 0 枠は書かない</p> <p>(2) A1 図のフレームの水平方向の長さ (mm 単位)</p> <p>(3) A2 " 垂直方向の長さ (mm 単位)</p>			
<div style="text-align: center;">  <p>SCSIZEで指定</p> </div>			

4	* ORIGIN	概要	作図図形の原点を指定
<u>入力形式とデフォルト値</u>			
* ORIGIN A1 = 30.0, A2 = 10.0			
<u>パラメータの説明</u>			
(1) A1 原点のx座標 (mm単位)			
(2) A2 " y座標 (mm単位)			
			
<u>注意事項</u>			
(1) 指定する座標は、オリジナル原点 (COM, プロッタ, テクトロのハード原点) からの移動距離である。			

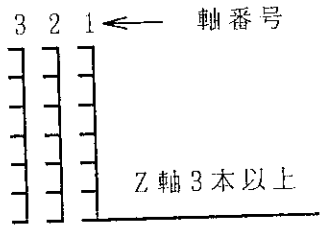
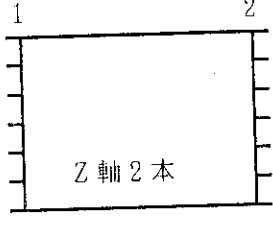
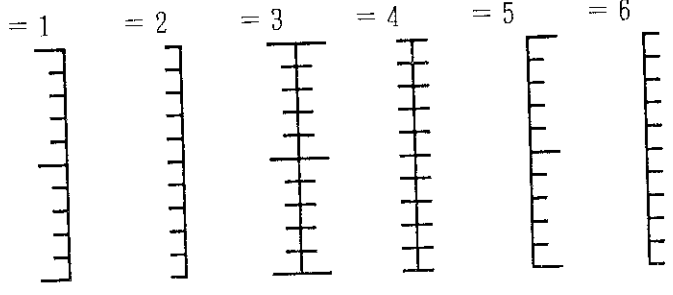
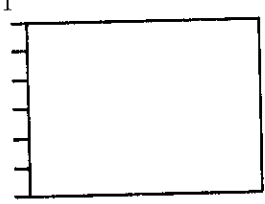
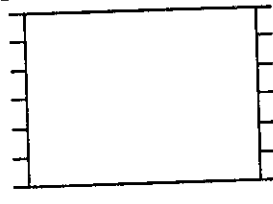
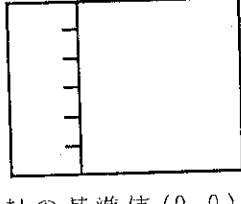
5	* SIDECDM	概要	空間データ出力時に時間情報を出力
<u>入力形式</u> * SIDECDM			
<ul style="list-style-type: none"> ◦ 図形指示データの& PARMで定義される変数について、空間データ出力時に変数名と数値が表示される。 ◦ 図形指示データの& FUNCで定義される変数について、空間データ出力時に変数名が表示される。さらにCDSデータの「* DATAPT」でセンタードシンボルの作画を指示した場合、変数名に対応したシンボルも表示される。 ◦ 上記の表示を行う場合、図形指示データの「& CDS」行から「* OUT」行までの間で、キーワード「* SIDECDM」を指定する。 			

6	* XAXIS	概要	x軸の作画特性の指定	1/2	
ONEDIM, ONED3D, CONTOR					
入力形式とデフォルト値					
* XAXIS N1 = 1, N2 = 1, N3 = 1, A1 = 3.0, ICM = X-AXIS&					
パラメータの説明					
(1) N1 目盛種類					
= 1			= 4		
= 2			= 5		
= 3			= 6		
(2) N2 データ変換オプション					
= 1 線形変換 = 2 LOG変換					
(3) N3 軸目盛をつける位置フラグ					
= 1		= 2		= 3	
下側のみ		上下		Z軸の基準値	
(注) 「ONED3D」の時は、N3はダミーで指定しなくてよい。常にN3 = 3の出力となる。					
(4) A1 目盛数値の大きさ (mm単位)					
>0.0 指定した大きさで作画					
=0.0 デフォールト値で作画 (3.0)					
<0.0 作画しない					
(5) ICM x軸タイトルの内容					
ICM = 'OOO&' の形式で指定する。					
詳細は、*MAINCM, *SUBCMを参照のこと。					
BRDEYE					
入力形式とデフォルト値					
* XAXIS N2 = 1, A1 = 3.0, ICM = 'X-AXIS&'					
パラメータの説明					
(1) N2 データ変換オプション					
= 1 線形変換 = 2 LOG変換					

<p>(2) A1 目盛数値, 軸タイトルの大きさ (≥ 0.0) (透視変換されるので, 実際の大きさではない)</p> <p>(3) ICM x軸タイトルの内容 ICM = '000&' の形式で指定する。 詳細は *MAINCM, *SUBCM を参照のこと。</p>	2 / 2
<p><u>備 考</u></p> <ul style="list-style-type: none">◦ ICM の入力方法は, *MAINCM, *SUBCM の場合と同じである。◦ 出力位置は, システムで自動決定される。	

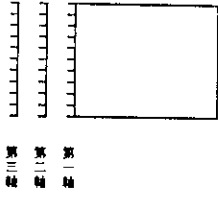
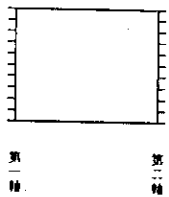
7	*XSCALE	概要	X軸のマニュアルスケール指定
<p>入力形式とデフォルト値</p> <p>*XSCALE A1 = 0.0 , A2 = 10.0</p>			
<p>パラメータの説明</p> <p>(1) A1 Xデータの最小値又は、X軸の最小目盛数値</p> <p>(2) A2 Xデータの最大値又は、X軸の最大目盛数値</p>			
			
<p>○ スケールのデフォルト状態は、自動スケールになっている。</p>			
<div style="border: 1px solid black; display: inline-block; padding: 2px;">ONEDIM , ONED3D</div> のみ使用可能			

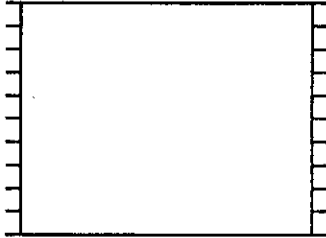
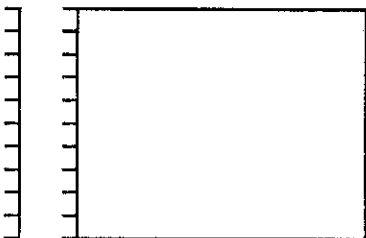
8	* XAUTO	概要	X軸のマニュアルスケールの解除
<u>入力形式 とデフォルト値</u> * XAUTO			
<ul style="list-style-type: none"> ◦ CDS : * XSCALEで指定したマニュアル・スケール指定を解除し, 自動スケールとする。 ◦ パラメータ入力なし <div style="border: 1px solid black; display: inline-block; padding: 2px 10px; margin: 5px 0;">ONEDIM, ONED3D</div> のみ使用可能			

9	*MLTAXS	概要 Z軸の作画特性の指定	1 / 3
ONEDIM			
入力形式とデフォルト値 *MLTAXS N1 = 1, N2 = 1, N3 = 1, N4 = 1, A1 = 3.0, ICM = 'Z-AXIS&'			
パラメータの説明			
(1) N1 目盛の軸番号 ($1 \leq N1 \leq 9$) <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <p>3 2 1 ← 軸番号</p>  <p>Z軸3本以上</p> </div> <div style="text-align: center;"> <p>1 2</p>  <p>Z軸2本</p> </div> </div>			
(2) N2 目盛種特 <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;">= 1</div> <div style="text-align: center;">= 2</div> <div style="text-align: center;">= 3</div> <div style="text-align: center;">= 4</div> <div style="text-align: center;">= 5</div> <div style="text-align: center;">= 6</div> </div> 			
(3) N3 データ変換オプション = 1 線形変換 = 2 LOG変換			
(4) N4 軸目盛をつける位置フラグ <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <p>= 1</p>  <p>左側</p> </div> <div style="text-align: center;"> <p>= 2</p>  <p>両側</p> </div> <div style="text-align: center;"> <p>= 3</p>  <p>X軸の基準値(0.0)</p> </div> </div>			
(5) A1 目盛数値の大きさ (mm単位) > 0.0 指定した大きさを作画 = 0.0 デフォルト値で作画 (3.0) < 0.0 作画しない			
(6) ICM Z軸タイトルの内容 ICM = 'OOO&' の形式で指定する。 詳細は、*MAINCM, *SUBCMを参照のこと			

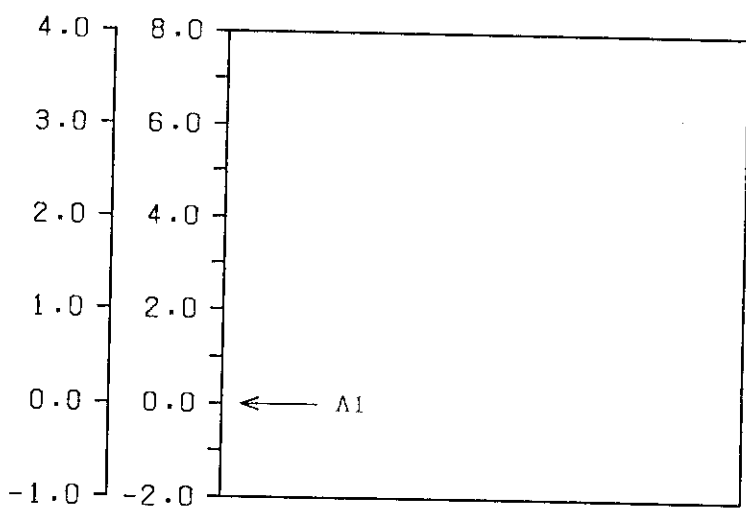
ONED 3 D	2 / 3
<u>入力形式とデフォルト値</u> *MLTAXS N2 = 1, N3 = 1, A1 = 3.0, ICM = 'Z - AXIS&'	
<u>パラメータの説明</u> (1) N2 目盛の種類 (= 1 ~ 6) (「ONEDIM」と同じ) (2) N3 データ変換オプション = 1 線形変換 = 2 LOG変換 (3) A1 目盛数値の大きさ (mm単位) (「ONEDIM」と同じ) (4) ICM Z軸タイトルの内容 (「ONEDIM」と同じ)	
CONTOR, BRDEYE <u>入力形式とデフォルト値</u> *MLTAXS N3 = 1, A1 = 3.0, ICM = 'Z - AXIS&'	
<u>パラメータの説明</u> (1) N3 データ変換オプション = 1 線形変換 = 2 LOG変換 (2) A1 目盛数値, Z軸タイトルの大きさ (≥ 0.0) (透視変換されるので, 実際の大きさではない) (注) 「CONTOR」のとき指定する必要はない。 (4) ICM Z軸タイトルの内容 ICM = 'OOO&'の形式で指定する。 詳細は, *MAINCM, *SUBCMを参照のこと。 (注) 「CONTOR」のとき指定する必要はない。	
<u>備考</u> ◦ ICMの入力方法は, *MAINCM, *SUBCMの場合と同じである。 ◦ 出力位置は, システムで自動決定される。	

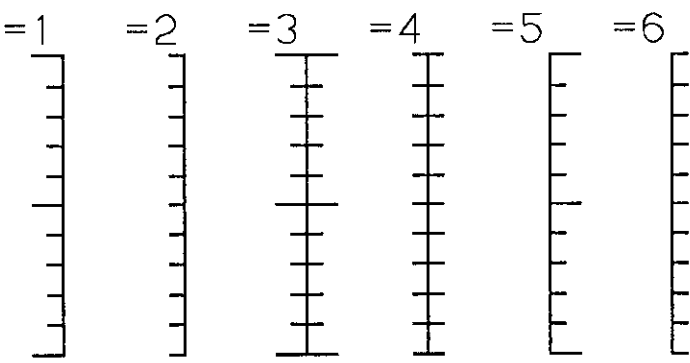
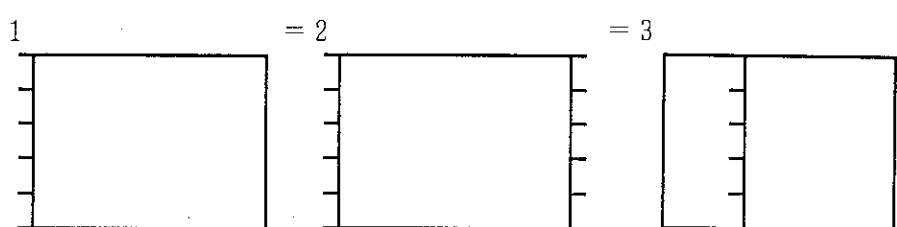
10	*MLTSCL	概要	Z軸のマニュアルスケール指定
ONEDIM, ONED3D			
<p>入力形式とデフォルト値</p> <p>*MLTSCL N1 = 1, A1 = 0.0, A2 = 10.0</p>			
<p>パラメータの説明</p> <p>(1) N1 マニュアルスケールしたいZ軸の軸番号 ($1 \leq N1 \leq 9$) (注) 「ONED3D」のとき指定しなくてよい。</p> <p>(2) A1 Zデータの最小値又は、Z軸の最小目盛数値</p> <p>(3) A2 Zデータの最大値又は、Z軸の最大目盛数値</p> <div data-bbox="478 793 829 986" style="text-align: center;"> </div>			
CONTOR, BRDEYE			
<p>入力形式とデフォルト値</p> <p>*MLTSCL A1 = 0.0, A2 = 10.0</p>			
<p>パラメータの説明</p> <p>(1) A1 Zデータの最小値又は、Z軸の最小目盛数値</p> <p>(2) A2 Zデータの最大値又は、Z軸の最大目盛数値</p> <div data-bbox="989 1270 1308 1542" style="text-align: center;"> </div>			
<p>○ スケールのデフォルト状態は、自動スケールである。</p>			

11	*MLTAUT	概要	Z軸のマニュアル・スケールの解除
ONEDIM, ONED3D			
入力形式とデフォルト値 *MLTAUT N1 = 1			
パラメータの説明 (1) N1 マニュアルスケールを解除したいZ軸の軸番号 ($1 \leq N1 \leq 9$) (注) 「ONED3D」のときは指定しなくてよい。 <div style="display: flex; justify-content: space-around; align-items: center;"> <div style="text-align: center;"> <p>N1 = 3</p>  </div> <div style="text-align: center;"> <p>N1 = 2</p>  </div> </div>			
CONTOR, BRDEYE			
入力形式とデフォルト値 *MLTSCL			
パラメータの説明 パラメータ入力なし。			
○ CDS : *MLTSCLで指定したマニュアル・スケール指定を解除し, 自動スケールとする。			

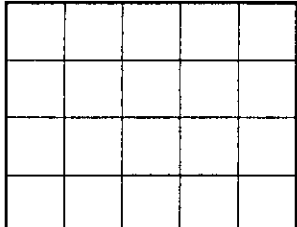
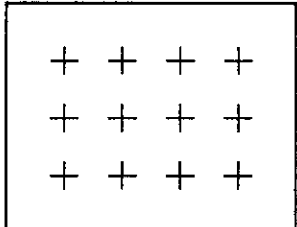
12	* ZAXSNO	概要	Z 軸の本数指定
<p><u>入力形式とデフォルト値</u> * ZAXSNO N1 = 1</p>			
<p><u>パラメータの説明</u> (1) N1 Z 軸の本数 ($1 \leq N1 \leq 9$)</p>			
<p><u>備考</u> (1) Z 軸の本数が 2 本以上の場合には「* ATTACH」により曲線と軸の関連づけを必ずしなければならない。 (2) ONEDIM のみ使用可能</p> <p><u>注意事項</u></p> <div style="display: flex; justify-content: space-around; align-items: flex-start;"> <div style="text-align: center;"> <p>N1 = 2</p>  <p>第一軸 第二軸</p> </div> <div style="text-align: center;"> <p>N1 = -2</p>  <p>第二軸 第一軸</p> </div> </div>			

13	* ATTACH	概要	Z 軸と曲線との関連づけを指定
<u>入力形式とデフォルト値</u> * ATTACH N1 = 1, ICM = ∇ 1 - 10 & ∇			
<u>パラメータの説明</u> (1) N1 Z 軸の軸番号 ($1 \leq N1 \leq 9$) (2) ICM 軸番号 N1 に割り当てる曲線の番号 (≤ 10) 文字形式で指定し, ICM = ∇ 1 - 3, 5 & ∇ の如く ∇ で括り & で終わること。			
<u>注意事項</u> (1) ONEDIM のみ使用可能 使用例 Z 軸 3 本 } 曲線 10 本 } であるとする。 Z 軸番号 1 \leftrightarrow 曲線 1, 4, 5, 7, 8, 9 } 2 \leftrightarrow 2, 3 } の如く割り当てる 3 \leftrightarrow 6, 10 }			
<u>以下の如く指定する</u> $\left\{ \begin{array}{l} * \text{ATTACH} \quad N1 = 1, \quad \text{ICM} = \nabla 1, 4 - 5, 7 - 9 \nabla \\ * \text{ATTACH} \quad N1 = 2, \quad \text{ICM} = \nabla 2 - 3 \nabla \\ * \text{ATTACH} \quad N1 = 3, \quad \text{ICM} = \nabla 6, 10 \nabla \end{array} \right.$			

14	* ZBASE	概要	Z 軸の基準値の指定
<p><u>入力形式とデフォルト値</u></p> <p>* ZBASE N1 = 0, A1 = 0.0</p>			
<p><u>パラメータの説明</u></p> <p>(1) N1 基準値の指定又は, その解除フラグ = 1 基準値の指定 = 0 解除</p> <p>(2) A1 基準値 (N1 = 0 のときはダミー)</p>			
<p><u>注意事項</u></p> <p>(1) Z 軸の最小値, 最大値の範囲に A1 の値がない場合, データ線が図形範囲からはずれ, 図が出力されない場合がある。</p> <p>(2) 基準値 A1 の目盛位置は, 第 1 Z 軸に合せる。</p> <p>(3) ONEDIM, ONED3D のみ使用可能</p>			
<p><u>使用例</u></p> <p>* ZBASE N1 = 1, A1 = 0.0</p> <div style="text-align: center;">  </div>			

15	*YAXIS	概要	y軸の作画特性の指定	1 / 2
<div style="border: 1px solid black; padding: 2px; display: inline-block;">ONED 3 D</div>				
<u>入力形式とデフォルト直</u> *YAXIS A1 = 2.5, ICM = 'Y-AXIS&'				
<u>パラメータの説明</u> (1) A1 目盛数値の大きさ (mm単位) > 0.0 指定した大きさで作画 = 0.0 デフォルト値で作画 < 0.0 作画しない (2) ICM y軸のタイトル内容 ICM = 'OOO&'の形式で指定する。 詳細は、*MAINCM, *SUBCMを参照のこと				
<div style="border: 1px solid black; padding: 2px; display: inline-block;">CONTOR, BRDEYE</div>				
<u>入力形式とデフォルト直</u> *YAXIS N1 = 1, N2 = 1, N3 = 1, A1 = 3.0, ICM = 'Y-AXIS&'				
<u>パラメータの説明</u> (1) N1 目盛種類 (BRDEYEの時指定する必要はない) =1 =2 =3 =4 =5 =6  (2) N2 データ変換オプション = 1 線形変換 = 2 LOG変換 (3) N3 軸目盛をつける位置フラグ = 1 = 2 = 3  <div style="display: flex; justify-content: space-around; margin-top: 5px;"> 左側 両側 X軸の基準値(0.0) </div> (注) 「BRDEYE」の場合指定する必要ない。				

<p>(4) A1 目盛数値の大きさ (mm単位)</p> <table border="0" style="width: 100%;"> <tr> <td style="text-align: center; border-bottom: 1px solid black;"><u>CONTOR</u></td> <td style="text-align: center; border-bottom: 1px solid black;"><u>BRDEYE</u></td> </tr> <tr> <td style="vertical-align: top;"> $\left\{ \begin{array}{l} > 0.0 \text{ 指定した大きさで作画} \\ = 0.0 \text{ デフォルト値で作画 (3.0)} \\ < 0.0 \text{ 作画しない} \end{array} \right.$ </td> <td style="vertical-align: top;"> <ul style="list-style-type: none"> ◦ 目盛数値, 軸タイトルの大きさ ◦ 透視変換されるので実際の大きさでない </td> </tr> </table> <p>(5) ICM y軸タイトルの内容</p> <p style="margin-left: 40px;">ICM = 'OOO&' の形式で指定する。</p> <p style="margin-left: 40px;">詳細は, *MAINCM, *SUBCMを参照のこと</p>	<u>CONTOR</u>	<u>BRDEYE</u>	$\left\{ \begin{array}{l} > 0.0 \text{ 指定した大きさで作画} \\ = 0.0 \text{ デフォルト値で作画 (3.0)} \\ < 0.0 \text{ 作画しない} \end{array} \right.$	<ul style="list-style-type: none"> ◦ 目盛数値, 軸タイトルの大きさ ◦ 透視変換されるので実際の大きさでない 	<p>2 / 2</p>
<u>CONTOR</u>	<u>BRDEYE</u>				
$\left\{ \begin{array}{l} > 0.0 \text{ 指定した大きさで作画} \\ = 0.0 \text{ デフォルト値で作画 (3.0)} \\ < 0.0 \text{ 作画しない} \end{array} \right.$	<ul style="list-style-type: none"> ◦ 目盛数値, 軸タイトルの大きさ ◦ 透視変換されるので実際の大きさでない 				
<p><u>備 考</u></p>					
<p>(1) パラメータ N1, N3 については, 「BRDEYE」の場合には指定する必要ない。</p> <ul style="list-style-type: none"> ◦ ICMの入力方法は, *MAINCM, *SUBCMの場合と同じである。 ◦ 出力位置はシステムで自動決定される。 					

16	*GRID	概要	グリッドの種類指定
<p>入力形式とデフォルト値</p>			
<p>*GRID N1 = 0, N2 = 0, N3 = 0</p>			
<p>パラメータの説明</p>			
<p>(1) N1 グリッドの種類</p>			
<p>= 0 描かない</p>			
<p>= 1</p>			
		<p>= 2</p>	
			
<p>(2) N2 X軸のグリッド本数</p>			
<p>(3) N3 Z軸のグリッド本数</p>			
<p>N2, N3 < 0 描かない</p>			
<p>= 0 システムで自動決定</p>			
<p>> 0 N2本, N3本描く (両端含まない)</p>			
<p>注意事項</p>			
<p>(1) ONEDIM, CONTOR のみ使用可能</p>			

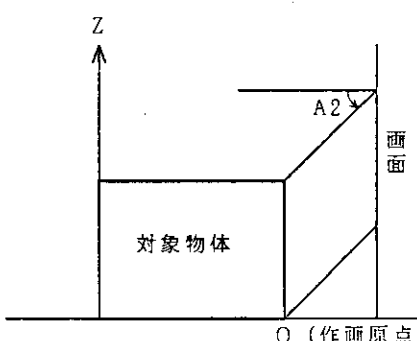
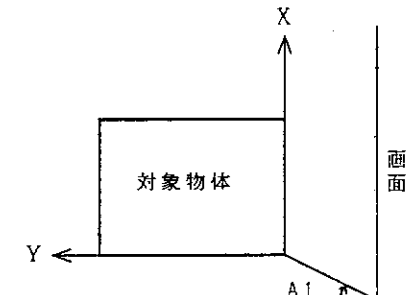
17	*KLINE	概要	曲線の線種類の指定
<p>入力形式とデフォルト値</p> <p>*KLINE IV1 = (1/1, 2/1, 3/1, 4/1, ……………, 9/1, 10/1)</p>			
<p>パラメータの説明</p> <p>(1) IV1 線種類</p> <p style="margin-left: 40px;">IV1 = (n₁/m₁, n₂/m₂, ……………, n_ℓ/m_ℓ) ℓ ; 曲線の本数</p> <p style="margin-left: 80px;"> $\left\{ \begin{array}{l} n_i ; \text{曲線の番号} (i = 1 \sim \ell, n_i \leq 10) \\ m_i ; \text{曲線} n_i \text{の線種類} (0 \leq m_i \leq 5) \end{array} \right.$ </p> <p style="margin-left: 40px;">m_i = 0 作画しない</p> <p style="margin-left: 80px;">= 1 実線 —————</p> <p style="margin-left: 80px;">= 2 太線 —————</p> <p style="margin-left: 80px;">= 3 点線 - - - - -</p> <p style="margin-left: 80px;">= 4 破線 - - - - -</p> <p style="margin-left: 80px;">= 5 一点鎖線 - - - - -</p>			
<p>注意事項</p> <p>(1) n₁, n₂, …………… は昇順に指定すること</p> <p>(2) ONEDIM のみ使用可能</p>			

18	* APPROX	概要	補間種類の指定
<u>入力形式とデフォルト値</u> * APPROX IV1 = (1/0, 2/0, 3/0, …, 9/0, 10/0)			
<u>パラメータの説明</u> (1) IV1 補間の種類 $IV1 = (n_1/m_1, n_2/m_2, \dots, n_\ell/m_\ell) \quad \ell ; \text{曲線の本数}$ $\begin{cases} n_i ; \text{曲線の番号} (i = 1 \sim \ell, n_i \leq 10, \ell \leq 10) \\ m_i ; \text{曲線} n_i \text{の補間種類} (= 0 \text{ 又は } 1) \end{cases}$ $m_i = 0 \quad \text{線形補間}$ $= 1 \quad \text{3次自然スプライン補間}$			
<u>注意事項</u> (1) <u>CONTOR</u> の場合は $IV1 = (1/0)$ 又は $IV1 = (1/1)$ とする。 (2) n_1, n_2, \dots は昇順とする。 (3) ONEDIM, CONTOR のみ使用可能			

19	*DATAPT	概要	センタードシンボル作画の指定	1 / 2
<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">ONEDIM</div> <p><u>入力形式とデフォルト値</u></p> <p>*DATAPT IV1 = (1/0, 2/0, 3/0, …… , 9/0, 10/0), IV2 = (1/-99, 2/-99, …… , 9/-99, 10/-99), IV3 = (1/1, 2/1, 3/1, …… , 9/1, 10/1), V1 = (1/2.5, 2/2.5, 3/2.5, …… , 9/2.5, 10/2.5)</p>				
<p><u>パラメータの説明</u></p> <p>(1) IV1 センタードシンボルコード番号</p> <p>IV1 = (n₁/m₁ , n₂/m₂ , …… , n_ℓ/m_ℓ) (ℓ ; 曲線の本数) n_i : 曲線の番号 (i = 1 ~ ℓ , n_i ≤ 10 , ℓ ≤ 10) m_i : シンボルコード又は, オプション m_i = 0 作画しない = 101 ~ 120 下記のシンボル作画</p> <p style="text-align: center;"> □ ○ △ + × ◇ ↑ ↖ ▽ Υ ⊞ ※ ☆ △ ◎ ⊗ ⊕ ⊖ ↔ </p> <p style="text-align: center;">101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120</p> <p>= 999 システムで自動決定</p> <p>(注) IV1 = (1/999) と指定するだけでよい)</p> <p>(2) IV2 シンボル作画開始位置</p> <p>IV2 = (n₁/K₁ , n₂/K₂ , …… , n_ℓ/K_ℓ) K_i = Nx_j 点 (X(Nx_j), Z(Nx_j)) より作画 1 ≤ Nx_j ≤ NX = -99 点 (X(1), Z(1)) にのみ作画 = -999 点 (X(NX), Z(NX)) にのみ作画</p> <p>(3) IV3 間引間隔</p> <p>IV3 = (n₁/i₁ , n₂/i₂ , n₃/i₃ , …… , n_ℓ/i_ℓ i_ℓ ; 間引間隔数 (1 ≤ i_i ≤ NX)</p> <p>(4) V1 センターシンボルの大きさ</p> <p>V1 = (n₁/s₁ , n₂/s₂ , n₃/s₃ , …… , n_ℓ/s_ℓ) s_i ; シンボルの大きさ (mm単位)</p>				
<div style="border: 1px solid black; padding: 2px; display: inline-block; margin-bottom: 5px;">CONTOR</div> <p><u>入力形式とデフォルト値</u></p> <p>*DATAPT N1 = 0, A1 = 2.5</p>				
<p><u>パラメータの説明</u></p> <p>(1) N1 センターシンボルコード番号</p> <p>= 0 作画しない。 = 101 ~ 120 下記を作画</p>				

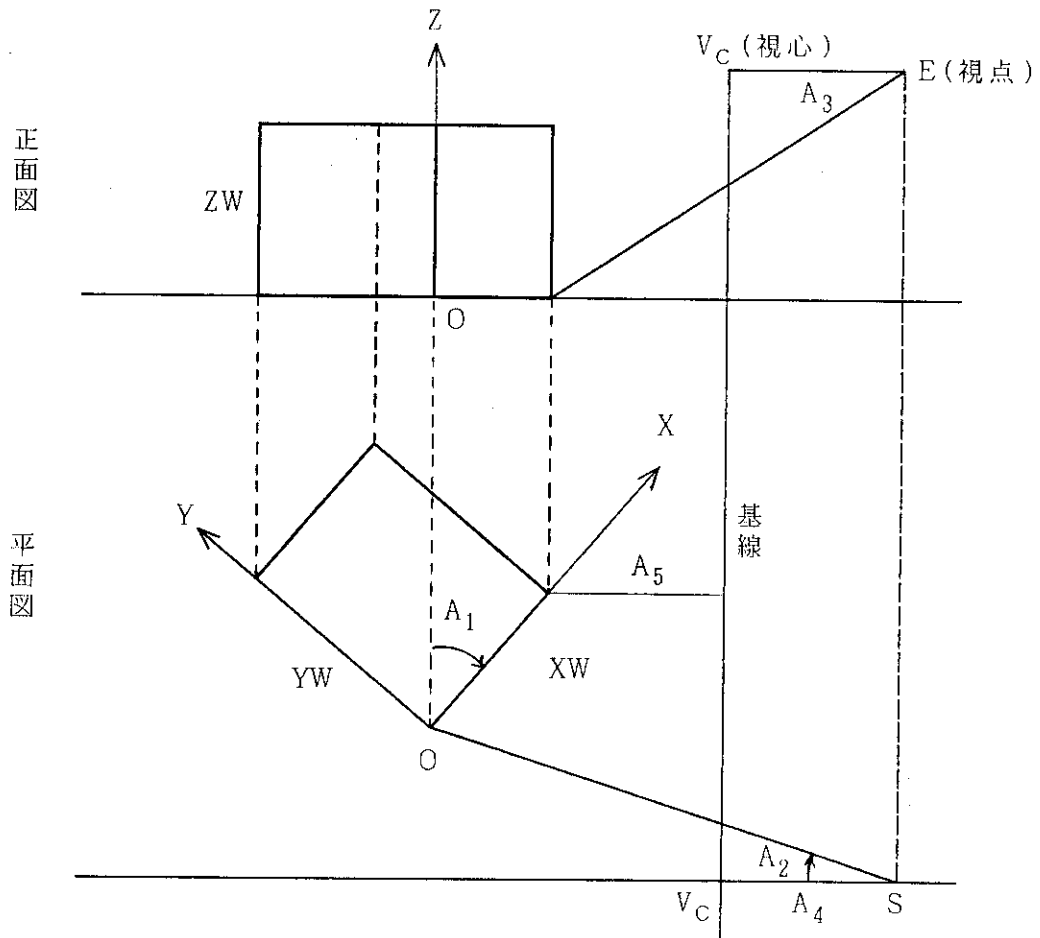
<p>□ ○ △ + × ◇ ↑ ✕ ▽ Υ 𠃊 ✕ ☆ △ ⊙ Σ ⊕ ⊖ ⇄</p> <p>101 102 103 104 105 106 107 108 109 110 111 112 113 114 115 116 117 118 119 120</p>	<p>2 / 2</p>
<p>(2) A1 センターシンボルの大きさ (mm単位)</p>	
<p><u>備考</u></p> <p>(1) シンボルを作画するとき、作画は、データ点全部に作画する。</p>	
<ul style="list-style-type: none"> ○ センタシンボルの作画座標は、シンボルの中心座標に対してである。 ○ センタシンボルに対応する番号については、表-1を参照 	

20	*OBJECT	概要	立体を収める箱の大きさ指定
<p>入力形式とデフォルト値</p> <p>*OBJECT A1 = 200.0, A2 = 120.0, A3 = 150.0 (ONED3D) (150.0) (150.0) (120.0) (BRDEYE)</p>			
<p>パラメータの説明</p> <p>(1) A1 X軸の大きさ (≥ 10.0) (2) A2 Y軸の大きさ (≥ 10.0) (3) A3 Z軸の大きさ (≥ 10.0)</p> <div data-bbox="767 598 1230 970" style="text-align: center;"> </div>			
<p>注意事項</p> <p>(1) A1, A2, A3 > 0 であること。あまり小さな値にしないこと。 (2) A1, A2, A3 は表示する時の x, y, z の比率を指定するものである。*GRSIZE で指定された A1, A2 の値をもつ長方形の領域に収まるように作画される。 (3) ONED3D, BRDEYE のみ使用可能</p>			




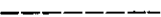

21	* VIEWP	概要 物体, 画面, 視点の位置関係を与える	1 / 2
<p>ONED3D</p>			
<p>入力形式とデフォルト値 * VIEWP N1 = 1, A1 = + 30.0, A2 = - 40.0</p>			
<p>パラメータの説明</p> <p>(1) N1 隠線処理オプション = 1 隠線処理する = 0 隠線処理しない</p> <p>(2) A1 視線とY軸のなす角 (度, 下図では正) ($-90^\circ < A1 < 90^\circ$)</p> <p>(3) A2 視点と原点Oを見上げる角 (度) ($-90^\circ < A2 < 0^\circ$)</p> <div style="text-align: center;">  <p>正面図</p> </div> <div style="text-align: center;">  <p>平面図</p> </div>			
<p>BRDEYE</p>			
<p>入力形式とデフォルト値 * VIEWP N1 = 1, A1 = 30.0, A2 = 0.0, A3 = - 30.0, A4 = 600.0, A5 = 200.0</p>			
<p>パラメータの説明</p> <p>(1) N1 隠線処理オプション = 1 隠線処理する。 = 0 隠線処理しない。</p> <p>(2) A1 x軸が基線となす角 (度, 下図では負) ($A1 < 180^\circ$)</p> <p>(3) A2 線分 OS と線分 $V_c S$ となす角 (度, 下図では正) ($A2 < 90^\circ$)</p> <p>(4) A3 基線に最も近い底面の頂点を視点から仰ぐ角 (度, 下図では負) ($A3 < 90^\circ$)</p>			

(5) A4 視距離 (mm 単位)

(6) A5 基線に最も近い底面の頂点と基線との距離 (mm 単位)



22	*XWYWGH	概要	x : y の実際の形の比を保存するか否かを指定する (実形表示オプション)
<u>入力形式とデフォルト値</u> *XWYWGH N1 = 0			
<u>パラメータ説明</u> (1) N1 実形表示オプション = 0 実際形で作画する = 1 CONTOR : *GRSIZE で指定した A1, A2 の大きさで作画。 BRDEYE : *OBJECT で指定した A1, A2 の大きさで作画。			
<u>注意事項</u> (1) CONTOR, BRDEYE のみ使用可能			

24	*Hkind	概要	等高線の線種類の指定
<p>入力形式とデフォルト値</p> <p>*Hkind N1 = 5, IV1 = (1/1, 2/2, 3/3, 4/4, 5/5)</p>			
<p>パラメータの説明</p> <p>(1) N1 線種類を指定する数 ($0 \leq N1 \leq 10$) = 0 ならば, 指定の解除</p> <p>(2) IV1 線種類</p> <p>IV1 = ($n_1/m_1, n_2/m_2, \dots, n_{N1}/m_{N1}$) n_i; 等高線番号 ($1 \leq n_1 < n_2 < \dots < n_{N1} \leq 10$) m_i; n_i の線種類</p> <p>$m_i = 0$ 書かない</p> <p>= 1 実線 </p> <p>= 2 太線 </p> <p>= 3 点線 </p> <p>= 4 破線 </p> <p>= 5 一点鎖線 </p>			
<p>注意事項</p> <p>(1) N1 が等高値の数 (*HVALUE で指定) より小さいときには, IV1 で指定したものが順にくり返し指定されたものとしてシステムで自動的に残り分をセットする。</p> <p>(2) N1 = 0 のとき IV1 は指定しなくてよい。</p> <p>(3) CONTOR のみ使用可能</p>			

25	*CVKIND	概要	等高値記入方法の指定																						
<p>入力形式とデフォルト値 *CVKIND N1 = 0, N2 = 1, N3 = 1</p>																									
<p>パラメータの説明</p> <p>(1) N1 等高値記入オプション = 0 等高線に等高値の値を記入 = 1 等高線に番号を記入 図の右側に等高値表を作画</p> <p>(2) N2 等高値を記入する最初の等高値の番号 N1 = 1 のときは指定しなくてよい。</p> <p>(3) N3 等高値を記入するときの等高値間隔</p>																									
<table border="1"> <thead> <tr> <th>等高線番号</th> <th>等高値 $h(i)$</th> </tr> </thead> <tbody> <tr><td>○ 1</td><td>$h(1) = 10.0$</td></tr> <tr><td>2</td><td>$h(2) = 20.0$</td></tr> <tr><td>3</td><td>$h(3) = 30.0$</td></tr> <tr><td>4</td><td>$h(4) = 40.0$</td></tr> <tr><td>○ 5</td><td>$h(5) = 50.0$</td></tr> <tr><td>6</td><td>$h(6) = 60.0$</td></tr> <tr><td>7</td><td>$h(7) = 70.0$</td></tr> <tr><td>8</td><td>$h(8) = 80.0$</td></tr> <tr><td>9</td><td>$h(9) = 90.0$</td></tr> <tr><td>○ 10</td><td>$h(10) = 100.0$</td></tr> </tbody> </table>		等高線番号	等高値 $h(i)$	○ 1	$h(1) = 10.0$	2	$h(2) = 20.0$	3	$h(3) = 30.0$	4	$h(4) = 40.0$	○ 5	$h(5) = 50.0$	6	$h(6) = 60.0$	7	$h(7) = 70.0$	8	$h(8) = 80.0$	9	$h(9) = 90.0$	○ 10	$h(10) = 100.0$	<p>図の例</p>	
等高線番号	等高値 $h(i)$																								
○ 1	$h(1) = 10.0$																								
2	$h(2) = 20.0$																								
3	$h(3) = 30.0$																								
4	$h(4) = 40.0$																								
○ 5	$h(5) = 50.0$																								
6	$h(6) = 60.0$																								
7	$h(7) = 70.0$																								
8	$h(8) = 80.0$																								
9	$h(9) = 90.0$																								
○ 10	$h(10) = 100.0$																								
<p>上表の等高値であるとする時 *CVKIND N1 = 0, N2 = 1, N3 = 5 と指定すると表中の ○印のついた等高線に等高値が表示される。</p>																									
<p>注意事項</p> <p style="border: 1px solid black; display: inline-block; padding: 2px;">CONTOR</p> のみ使用可能																									

*** CHARACTER TABLE ***

1	11	A	26	P	41	.	56	?	71	Q	86	p	101	□	115	⊙	121	κ	136	π	145	Γ
2	12	B	27	Q	42	<	57	:	72	h	87	q	102	○	117	⊗	122	β	137	ρ	146	Λ
3	13	C	28	R	43	(58		73	C	88	r	103	△	118	⊥	123	γ	138	σ	147	β
4	14	D	29	S	44	+	59	⊙	74	d	89	s	104	⊥	119	↑	124	δ	139	τ	148	Λ
5	15	E	30	T	45		60	'	75	e	90	t	105	×	120	↔	125	ε	140	υ	149	Σ
6	16	F	31	U	46		61	≡	76	f	91	u	106	◇			126	ζ	141	φ	150	
7	17	G	32	V	47	¥	62	"	77	g	92	v	107	⤴			127	η	142	χ	151	Σ
8	18	H	33	W	48	×			78	h	93	w	108	⊗			128	θ	143	ψ	152	Υ
9	19	I	34	X	49)			79	i	94	x	109	▽			129	ι	144	ω	153	φ
10	20	J			50	;			80	j			110	Y			130	κ			154	ψ
	21	K			51	/			81	k			111	⊗			131	λ			155	Ω
	22	L			52	,			82	l			112	✖			132	μ			156	.
	23	M			53	%			83	m			113				133	ν			157	Θ
	24	N			54	-			84	n			114	⊗			134	ξ			158	×
	25	O			55	>			85	o			115	△			135	ο				

表1 ARGUS-V4 ID=CHARACTR

3.4 LIBGR の使用例

1) LIBG を機動するために必要な JCL は、ライブラリシステムのユーザーであれば、マクロコマンド「LIBJCL」を用いてキーパラメータ「LIBGR」を入力すれば、以下の JCL が得られる。

例)

```
.LIBJCL LIBGR
KEY = LIBGR
### PLOT DATA FILE (INPUT) = 'J3051.PLOTX.DATA' ###
### LIBGR INPUT DATA FILE = 'J3051.LIBGR.DATA(XXX)' ###

00100 //JCLG JOB
00200 // EXEC JCLG
00300 //SYSIN DD DATA,DLM='++'
00400 // JUSER 04573051,TQ.HIRAYAMA,0019.06
00500 T.4 W.2 C.3 P.0 I.5 SRP GRP
00600 OPTP PASSWORD= ,MSGLEVEL=(1,0),NOTIFY=J3051
00700 //*****
00800 //***** CONSULTING TEL. 5-3111 ( 427 ) *****
00900 //*****
01000 ++
01100 // DD DSN=J2855.LIBJCL66.CNTL(LIBGR),DISP=SHR
01200 // DD DATA,DLM='%%'
01300 //LIBGR EXEC LIBGR,PNM=TEMPNAME,
01400 // LM='J2855.LIBGR',Q=' .LOAD',
01500 // LPLOT='J3051.PLOTX.DATA', ← 図形データファイル
01600 // LGR='J3051.LIBGR.DATA(XXX)', ← 図形指示データファイル
01700 // GOUT=H,
01800 // SYSOUT='*'
01900 %%
02000 //
```

2) 以下に、図形指示データと図の例を示す。

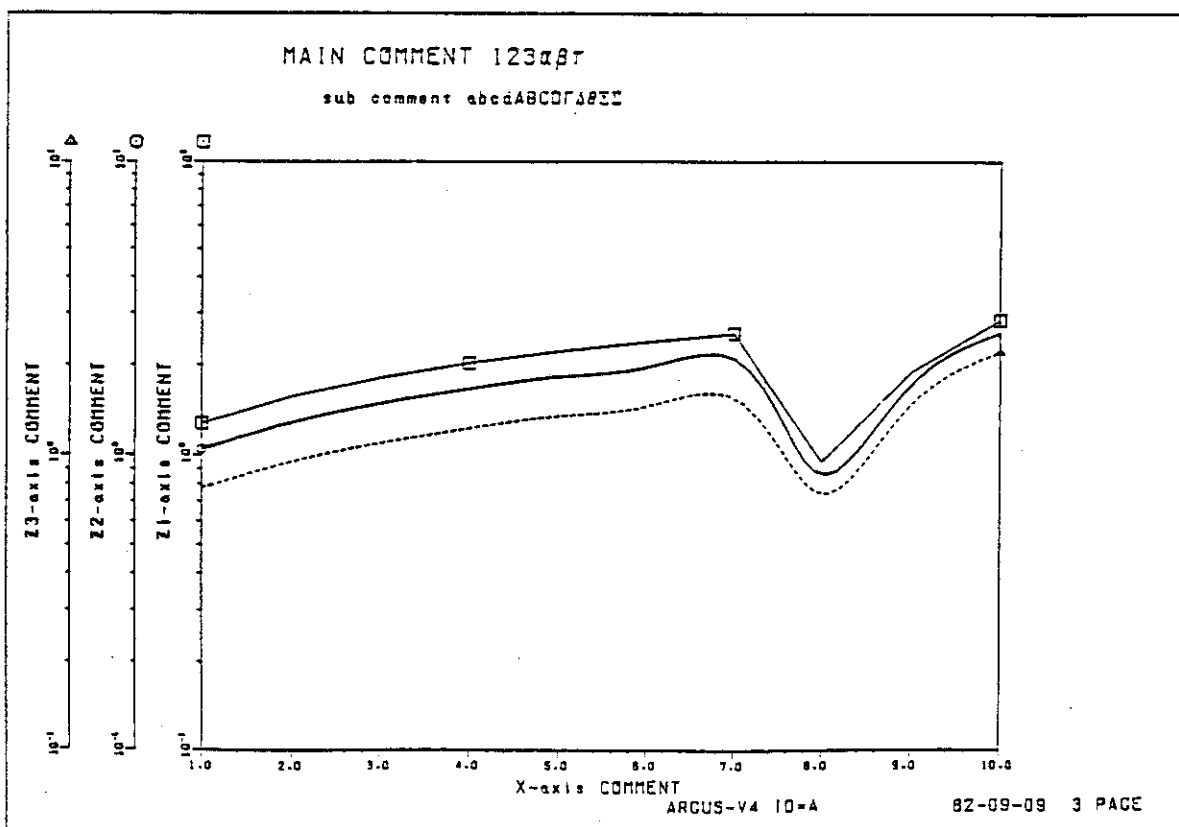
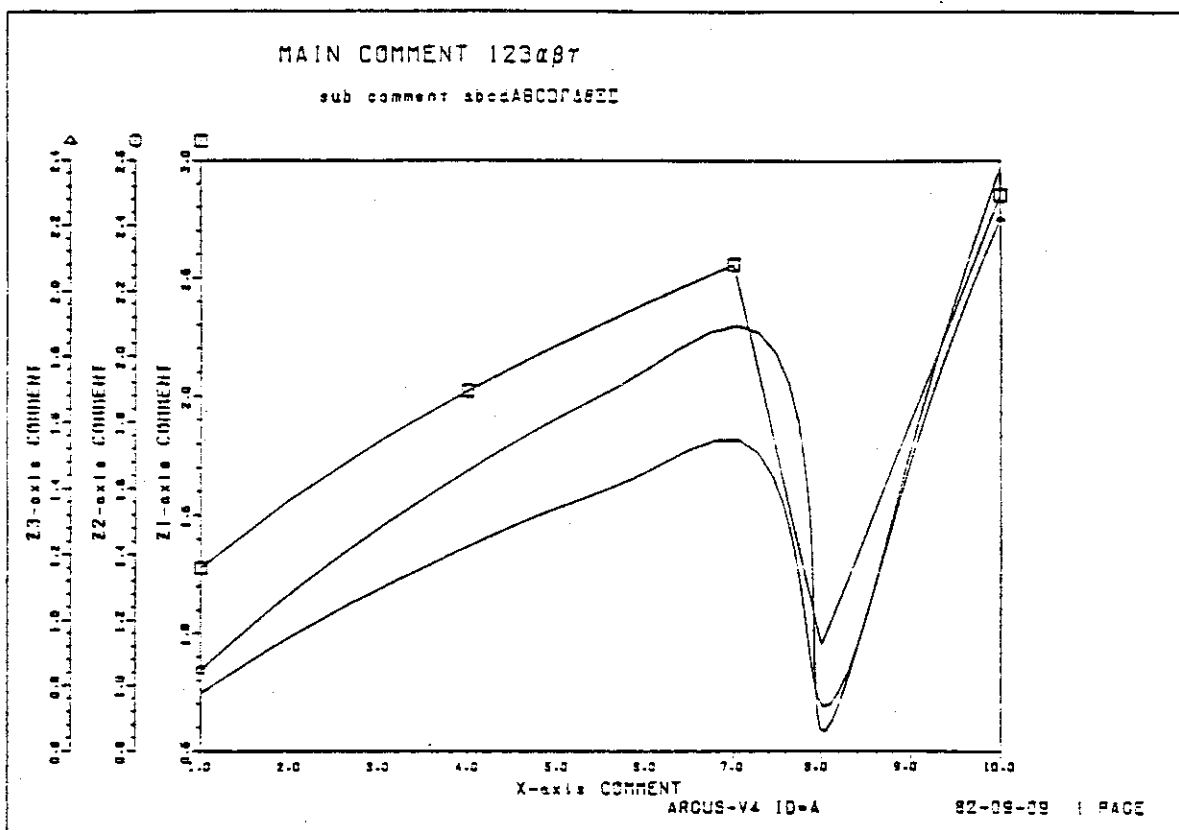
A. 一次元図, 斜投図 (1次元)

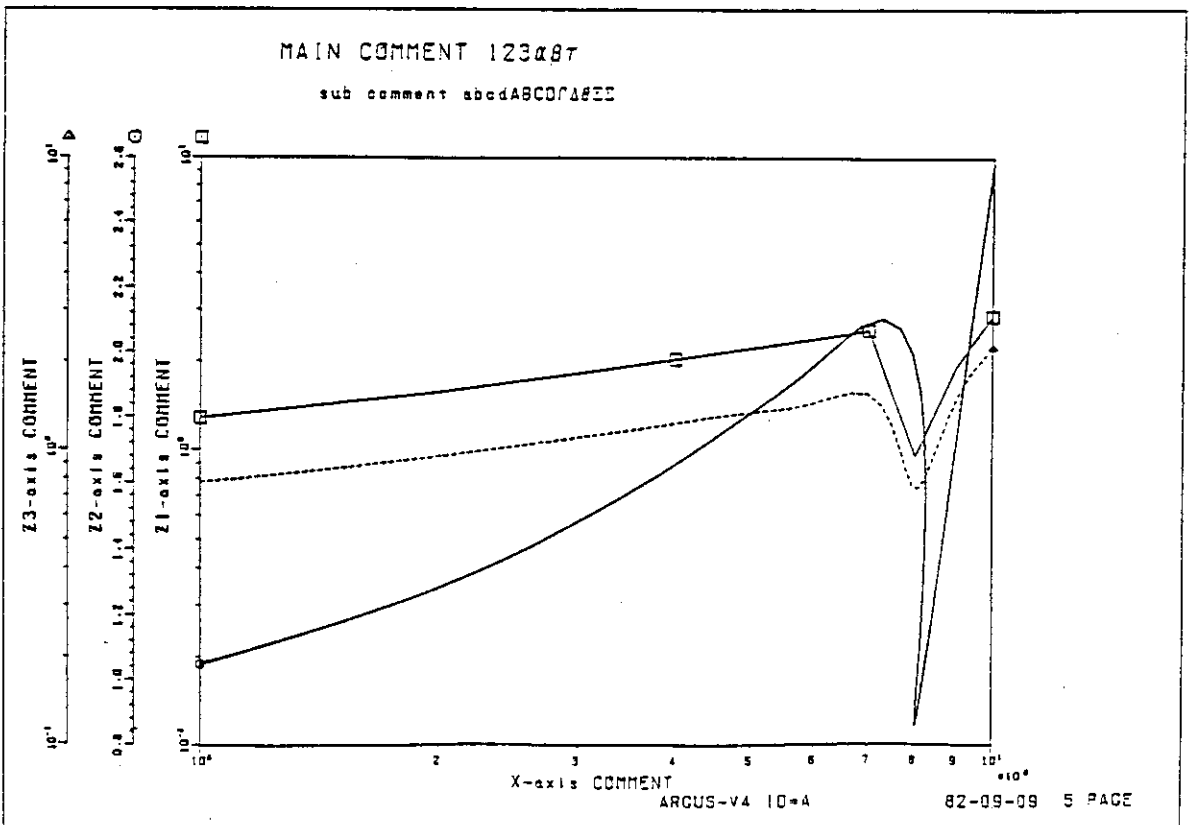
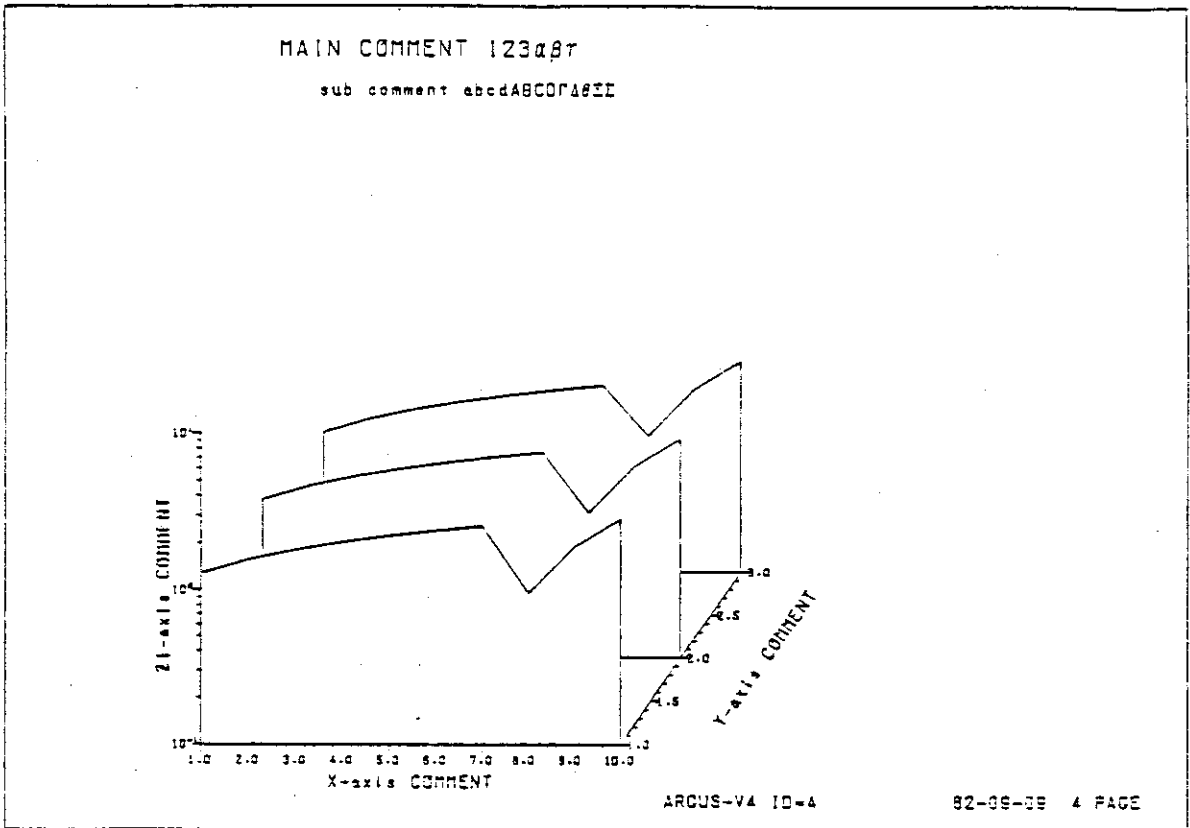
1. 図形「A」の例

```

&GNAME A
&GSCRT 1
&BASE R
&PARM T(1:3)
&FUNC VI(R)
&CDS
*MAINCM ICM='MAIN COMMENT 123'121,122,123'3'
*SUBCM ICM='SUB3' 'COMMENT' 'ABCD'ABCD'145,146,147,149,151'3'
*APPROX IV1=(1/0,2/1,3/1)
*XAXIS N1=1,A1=2.0,N2=1,N3=1,ICM='X-"AXIS" COMMENT&'
*MLTAXS N1=1,N2=1,N3=1,N4=1,A1=2.0,ICM='Z1-"AXIS" COMMENT&'
*MLTAXS N1=2,N2=1,N3=1,N4=1,A1=2.0,ICM='Z2-"AXIS" COMMENT&'
*MLTAXS N1=3,N2=1,N3=1,N4=1,A1=2.0,ICM='Z3-"AXIS" COMMENT&'
*ZAXSNO N1=3
*ATTACH N1=1,ICM='1&'
*ATTACH N1=2,ICM='2&'
*ATTACH N1=3,ICM='3&'
*OATAPT IV1=(1/101,2/102,3/103),IV2=(1/1,2/-99,3/-999),
IV3=(1/3,2/0,3/0),V1=(1/3,0,2/2.5,3/2.0)
*OUT ONEDIM
&COS
*MLTAXS N1=1,N2=1,N3=2,N4=1,A1=2.0,ICM='Z1-"AXIS" COMMENT&'
*MLTAXS N1=2,N2=1,N3=2,N4=1,A1=2.0,ICM='Z2-"AXIS" COMMENT&'
*MLTAXS N1=3,N2=1,N3=2,N4=1,A1=2.0,ICM='Z3-"AXIS" COMMENT&'
*KLINE IV1=(1/1,2/2,3/3)
*VIEWP A1=-30.0,A2=-40.0,N1=1
*YAXIS N1=1,A1=2.0,N2=1,ICM='Y-"AXIS" COMMENT&'
*OUT ONEDIM
*OUT ONED3D
&CDS
*XAXIS N1=1,A1=2.0,N2=2,N3=1,ICM='X-"AXIS" COMMENT&'
*MLTAXS N1=1,N2=1,N3=2,N4=1,A1=2.0,ICM='Z1-"AXIS" COMMENT&'
*MLTAXS N1=2,N2=1,N3=1,N4=1,A1=2.0,ICM='Z2-"AXIS" COMMENT&'
*MLTAXS N1=3,N2=1,N3=2,N4=1,A1=2.0,ICM='Z3-"AXIS" COMMENT&'
*OUT ONEDIM
&&&

```



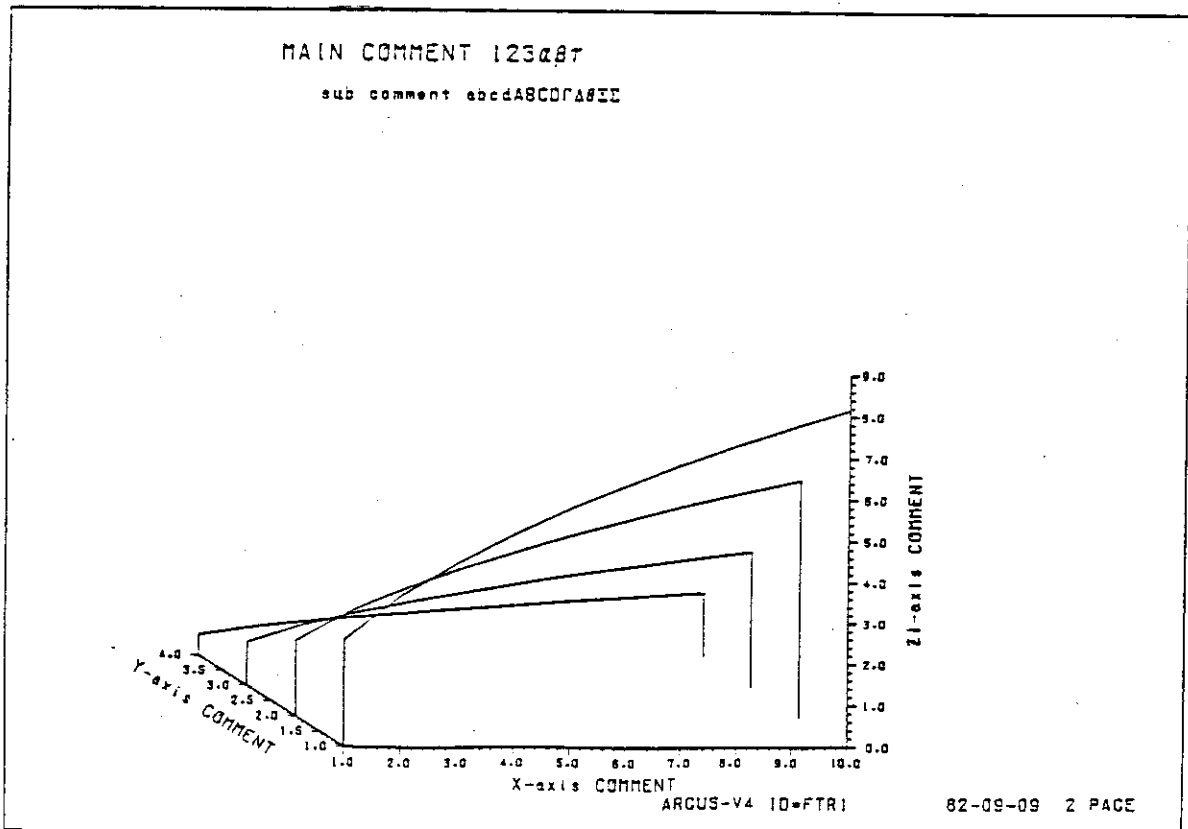
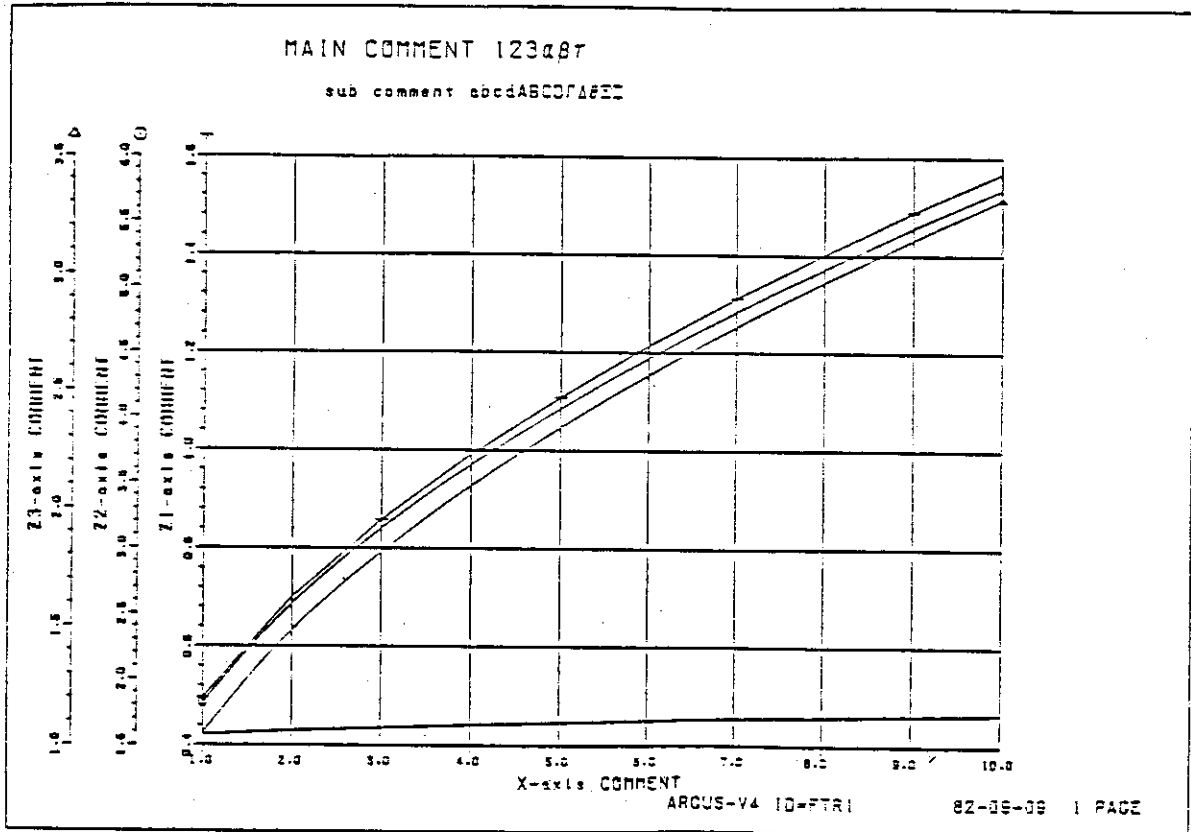


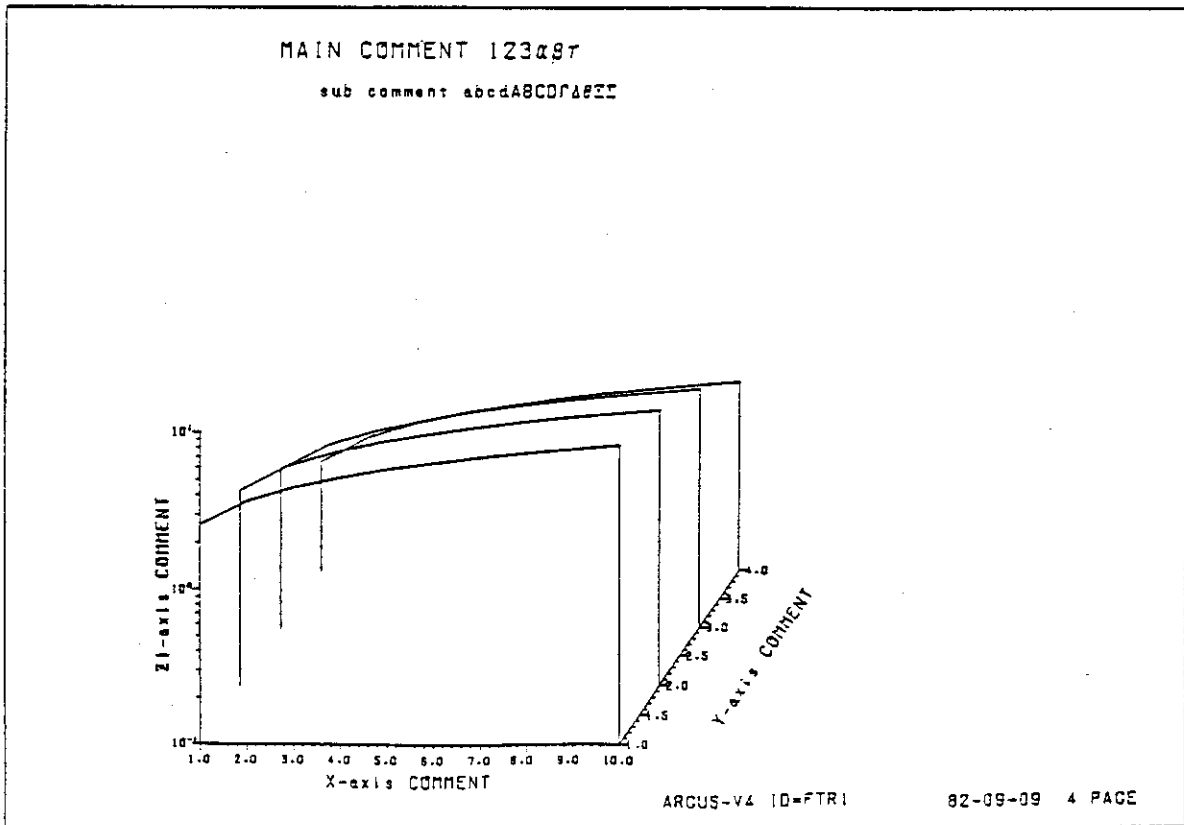
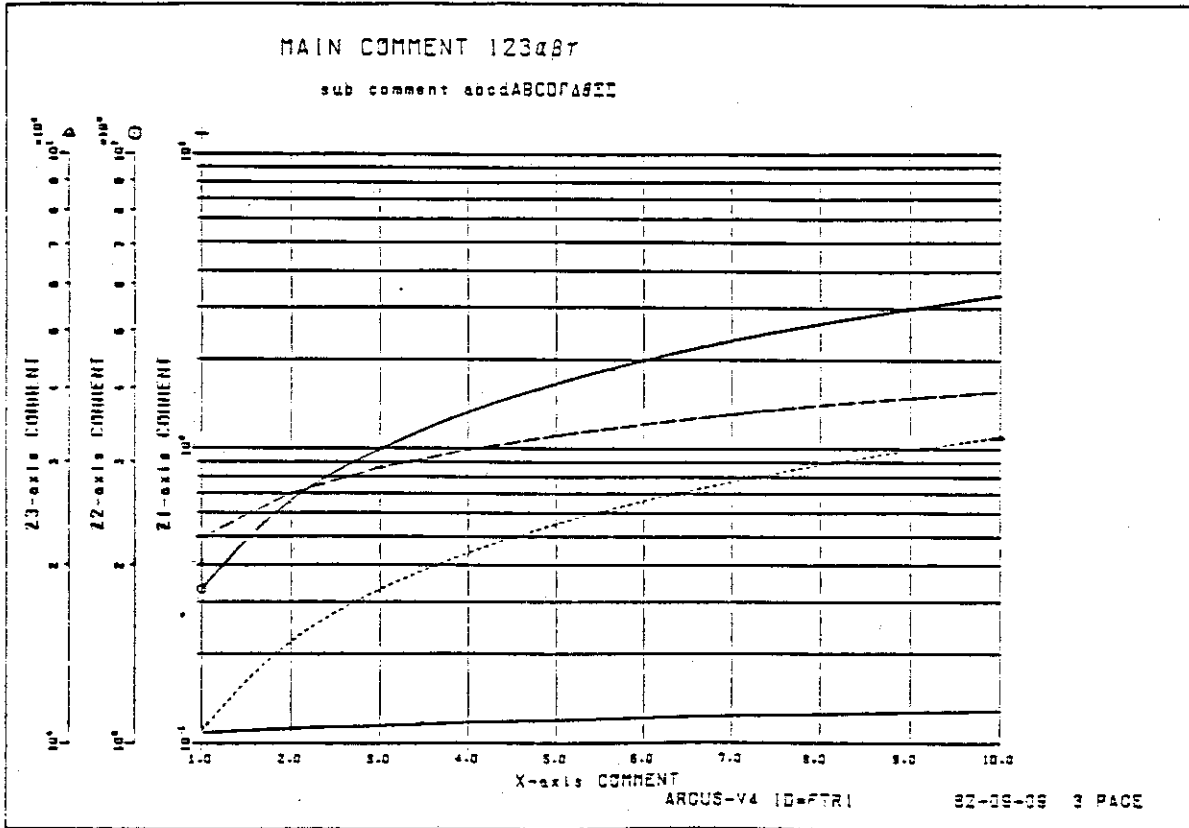
2. 図形「FTR1」の例

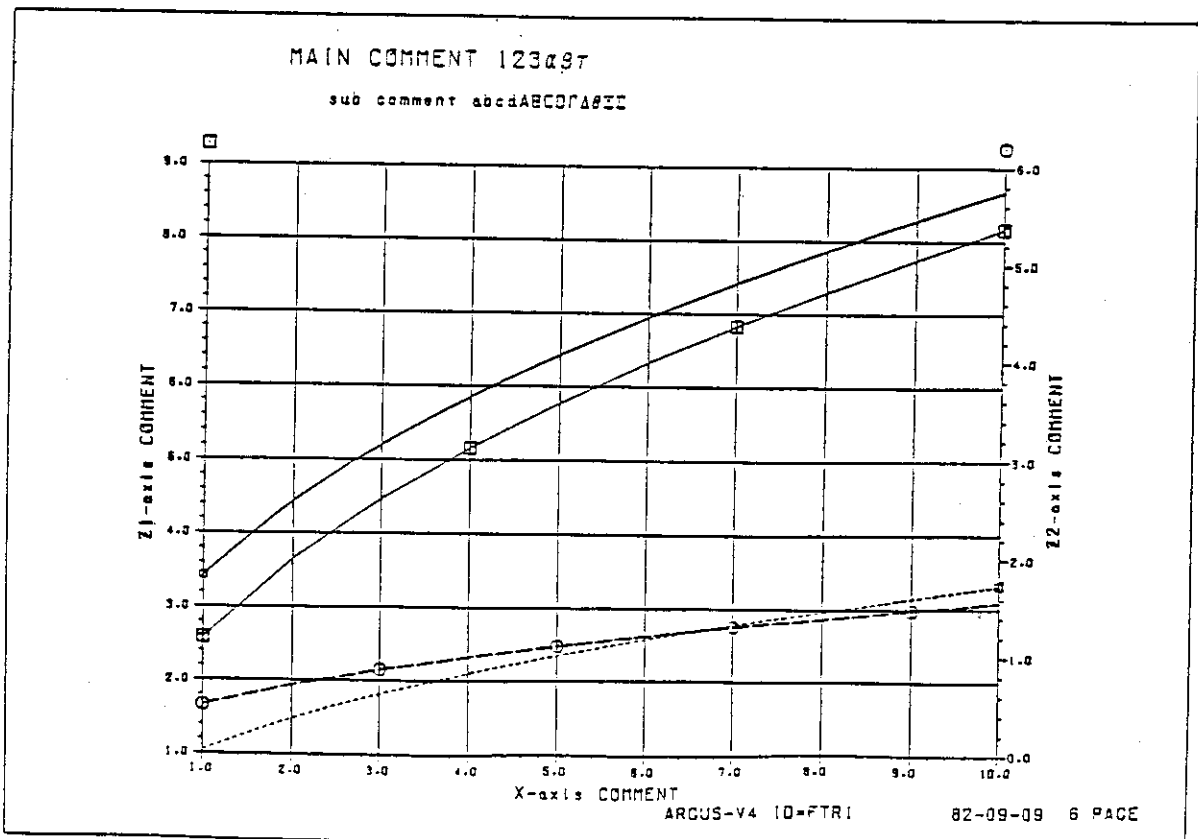
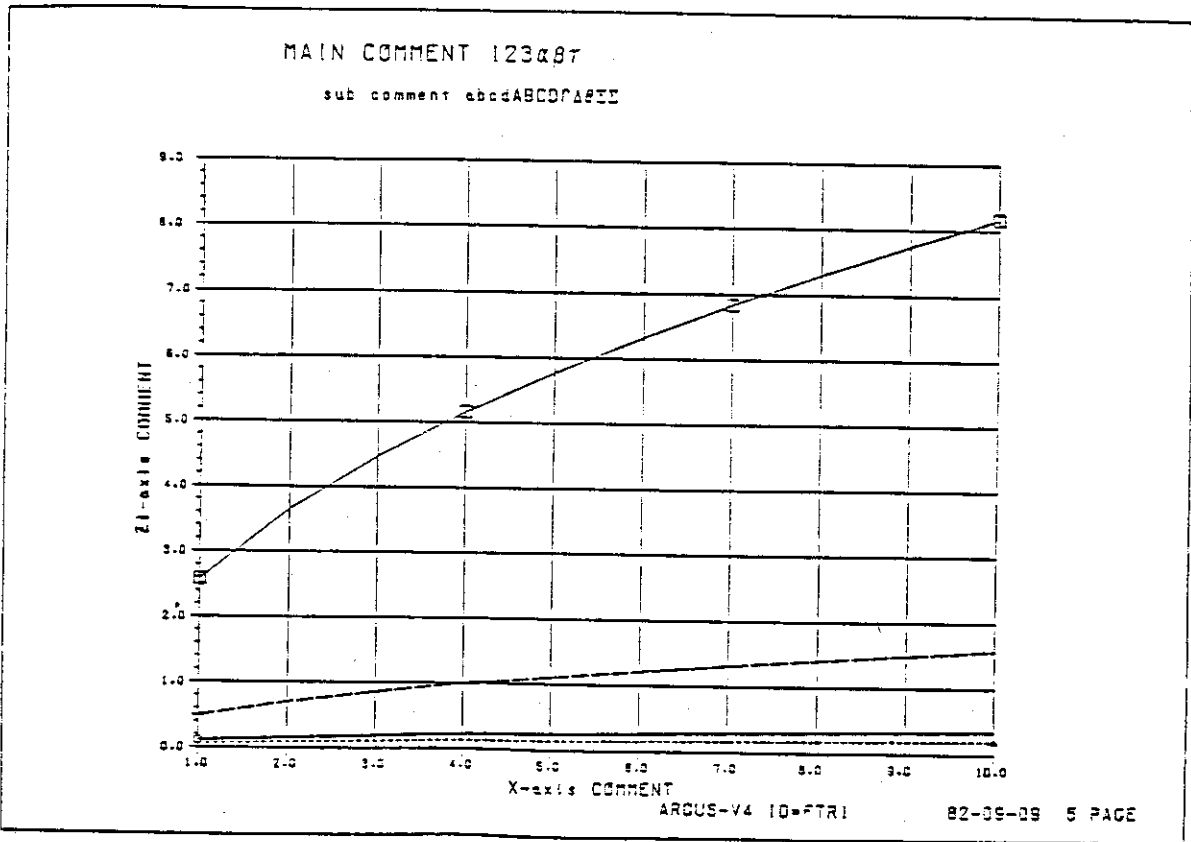
```

&GNAME FTR1
&GSCRT 1
&BASE R
&PARAM T(2),T(+),T(5),T(8)
&FUNC V2(3,R)
&CDS
*MAINCM ICM='MAIN COMMENT 123'121,122,123'3'
*SUBCM ICM='SUBE' COMMENT 'ABCD'ABCD'145,146,147,149,151'3'
*APPROX IV1=(1/0,2/1,3/1,4/0)
*XAXIS N1=1,A1=2.0,N2=1,N3=1,ICM='X-'AXIS' COMMENT&'
*YAXIS A1=2.0,ICM='Y-'AXIS' COMMENT&'
*MLTAXS N1=1,N2=1,A1=2.0,N3=1,N4=1,ICM='Z1-'AXIS' COMMENT&'
*MLTAXS N1=2,N2=1,A1=2.0,N3=1,N4=1,ICM='Z2-'AXIS' COMMENT&'
*MLTAXS N1=3,N2=1,A1=2.0,N3=1,N4=1,ICM='Z3-'AXIS' COMMENT&'
*ZAXSNO N1=3
*ATTACH N1=1,ICM='1&'
*ATTACH N1=2,ICM='2&'
*ATTACH N1=3,ICM='3&'
*ATTACH N1=+,ICM='1&'
*DATAPT IV1=(1/101,2/102,3/103,+/104),IV2=(1/1,2/-99,3/-999,4/1),
IV3=(1/3,2/0,3/0,4/2),V1=(1/3.0,2/2.5,3/2.0,4/3.0)
*VIEWP A1=30.0,A2=-0.0,N1=0
*GRID N1=1,N2=0,N3=0
*OUT QNEDIM
*OUT QNED3D
&CDS
*MLTAXS N1=1,N2=1,A1=2.0,N3=2,N4=1,ICM='Z1-'AXIS' COMMENT&'
*MLTAXS N1=2,N2=1,A1=2.0,N3=2,N4=1,ICM='Z2-'AXIS' COMMENT&'
*MLTAXS N1=3,N2=1,A1=2.0,N3=2,N4=1,ICM='Z3-'AXIS' COMMENT&'
*KLINE IV1=(1/1,2/2,3/3,4/4)
*VIEWP A1=30.0,A2=-20.0,N1=0
*OUT QNEDIM
*OUT QNED3D
&CDS
*MLTAXS N1=1,N2=1,A1=2.0,N3=1,N4=1,ICM='Z1-'AXIS' COMMENT&'
*MLTAXS N1=2,N2=1,A1=2.0,N3=1,N4=1,ICM='Z2-'AXIS' COMMENT&'
*MLTAXS N1=3,N2=1,A1=2.0,N3=1,N4=1,ICM='Z3-'AXIS' COMMENT&'
*ZAXSNO N1=1
*ATTACH N1=1,ICM='1-+&'
*OUT QNEDIM
&CDS
*ZAXSNO N1=2
*ATTACH N1=1,ICM='1,3&'
*ATTACH N1=2,ICM='2,+&'
*OUT QNEDIM
&&&

```







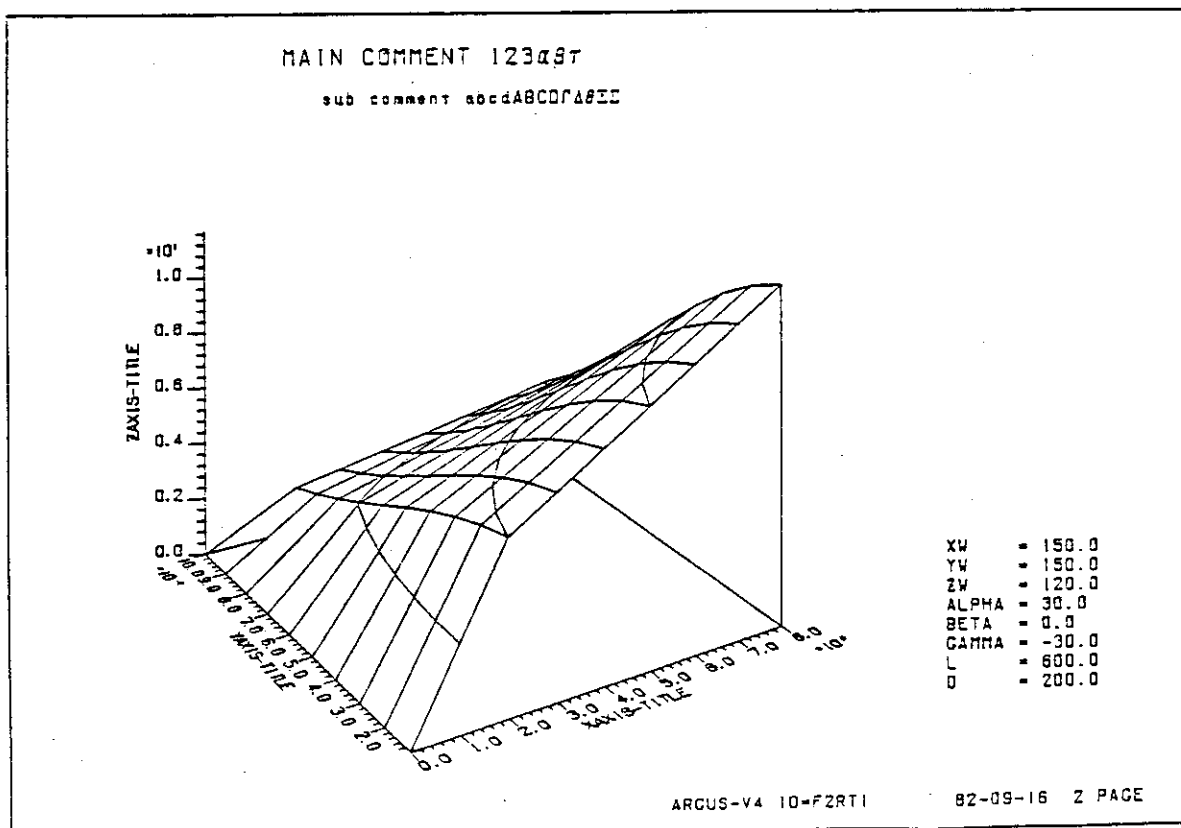
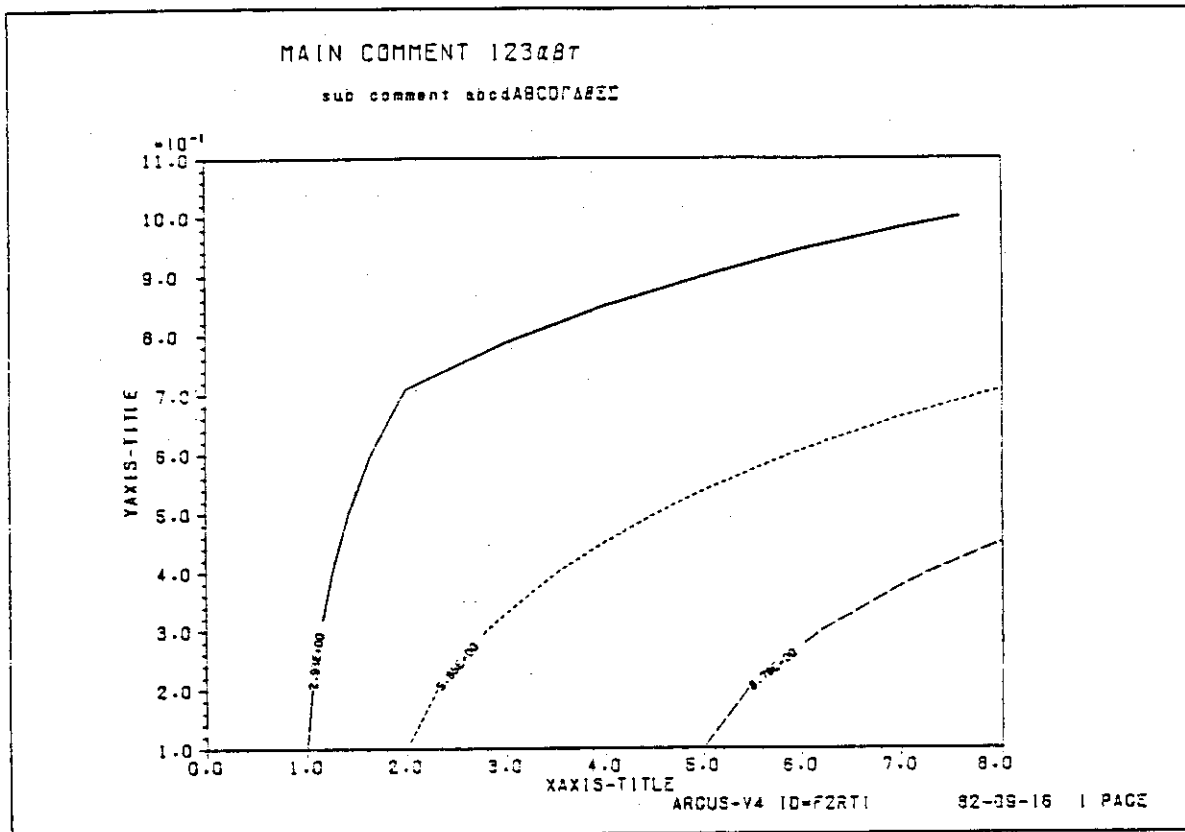
B. 等高線図, 鳥瞰図 (2次元)

1. 図形「F2RT1」の例

```

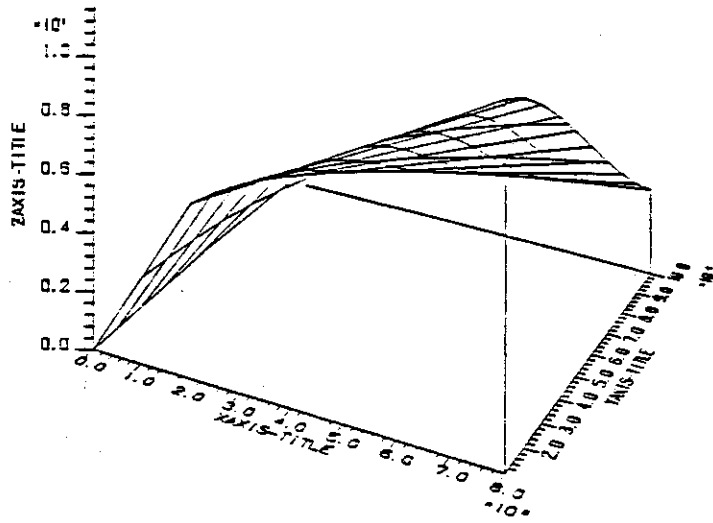
&GNAME F2RT1
&GSORT 2
&BASE R(2:9),T
&FUNC FXY(R,5)
&CDS
*MAINCM ICM='MAIN COMMENT 123 121,122,123 1'
*SUBCM ICM='SUB COMMENT ABCD ABCD 145,146,147,149,151 3'
*XAXIS N3=1,A1=3.0,ICM='XAXIS-TITLE3'
*YAXIS N3=1,A1=3.0,ICM='YAXIS-TITLE3'
*MLTAXS N3=1,A1=3.0,ICM='ZAXIS-TITLE3'
*VIEWP A1=30.0,A2=0.0,A3=-30.0,A4=600.0,A5=200.0,N1=1
*HVALUE N1=-5
*HKIND N1=5,IV1=(1/1,2/2,3/3,+4/4,5/5)
*XWYWGHI N1=1
*OUT CONTOUR
*OUT BROEYE
&CDS
*VIEWP A1=-30.0,A2=0.0,A3=-30.0,A4=600.0,A5=200.0,N1=1
*OUT BROEYE
&333

```



MAIN COMMENT 123aBT

sub comment abcdABCDPABEE



XW	=	150.0
YW	=	150.0
ZW	=	120.0
ALPHA	=	-30.0
BETA	=	0.0
GAMMA	=	-30.0
L	=	500.0
D	=	200.0

ARGUS-V4 10-F2RTI

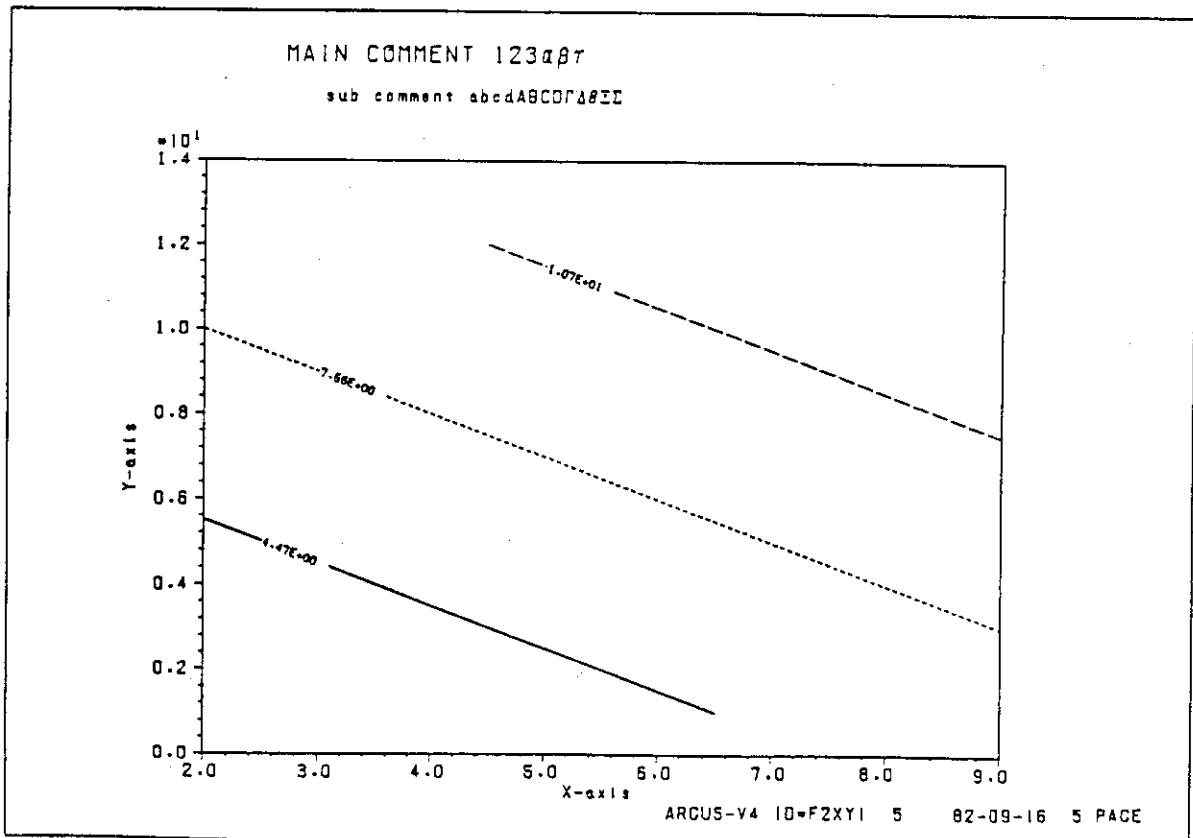
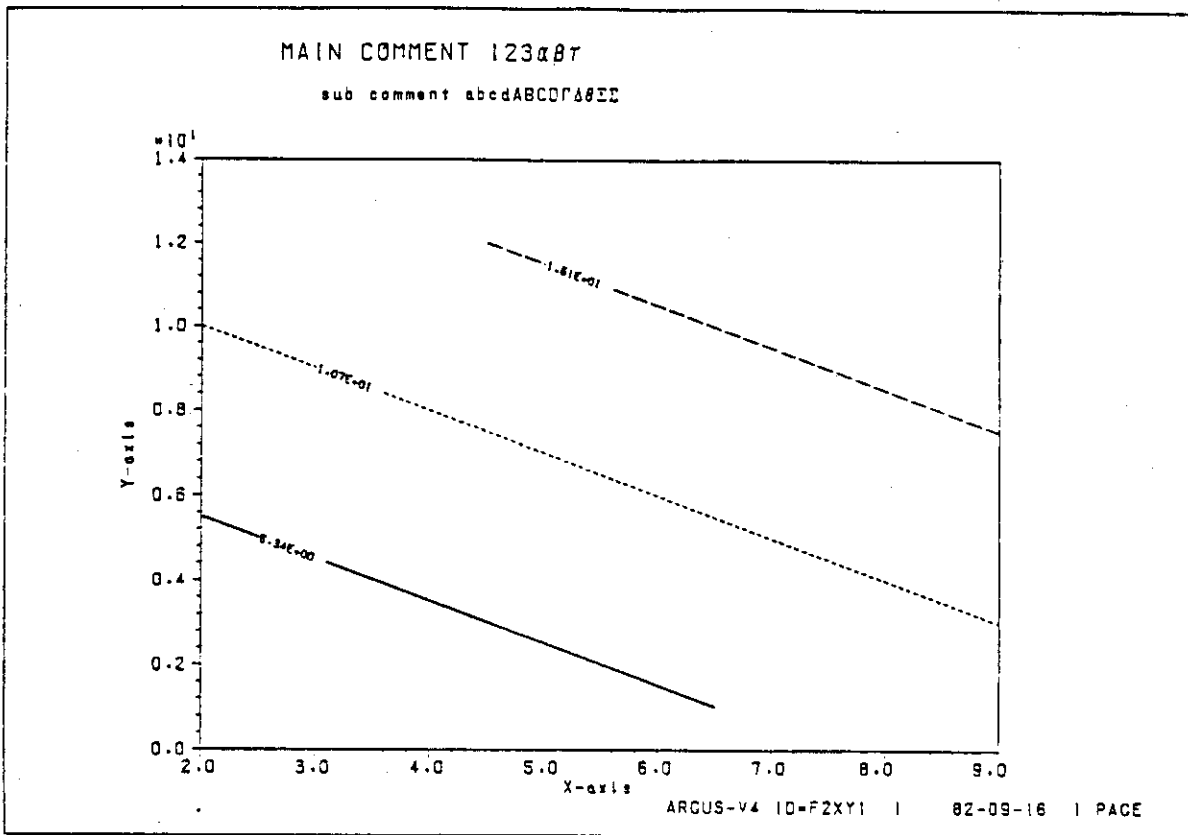
82-09-16 3 PAGE

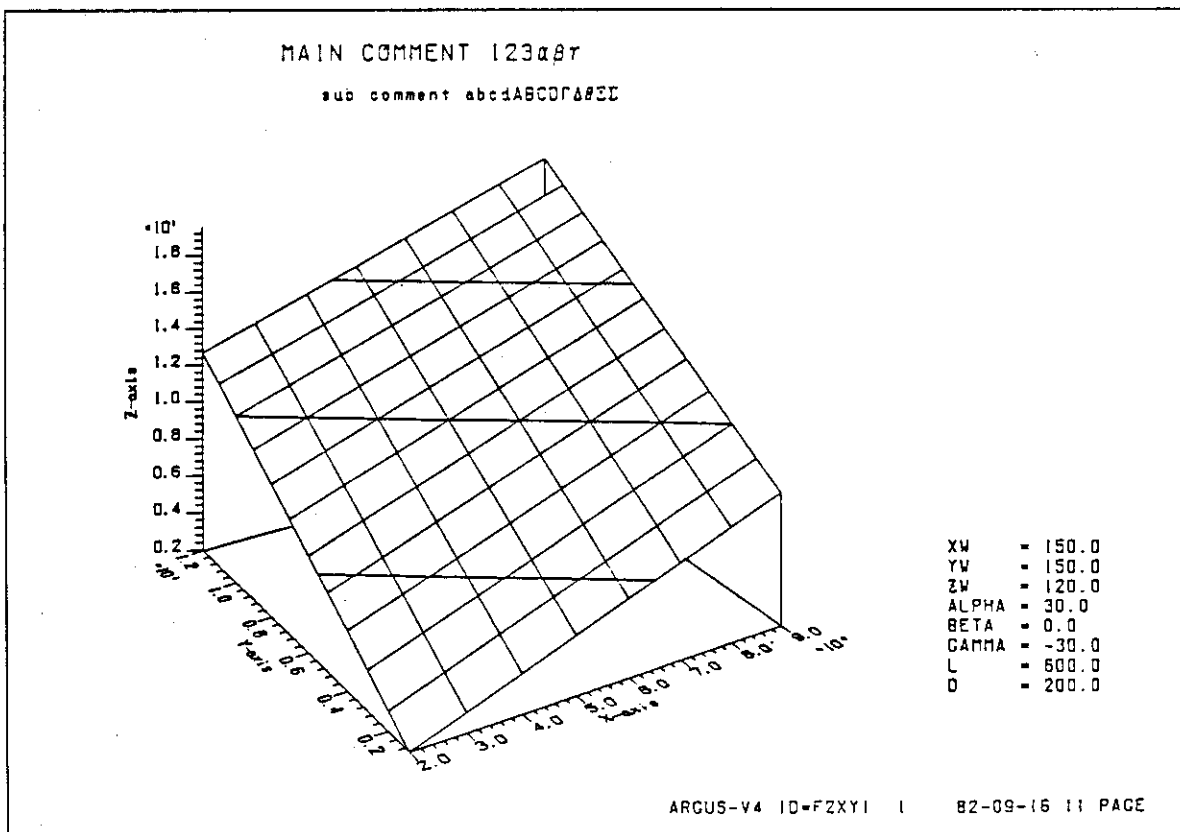
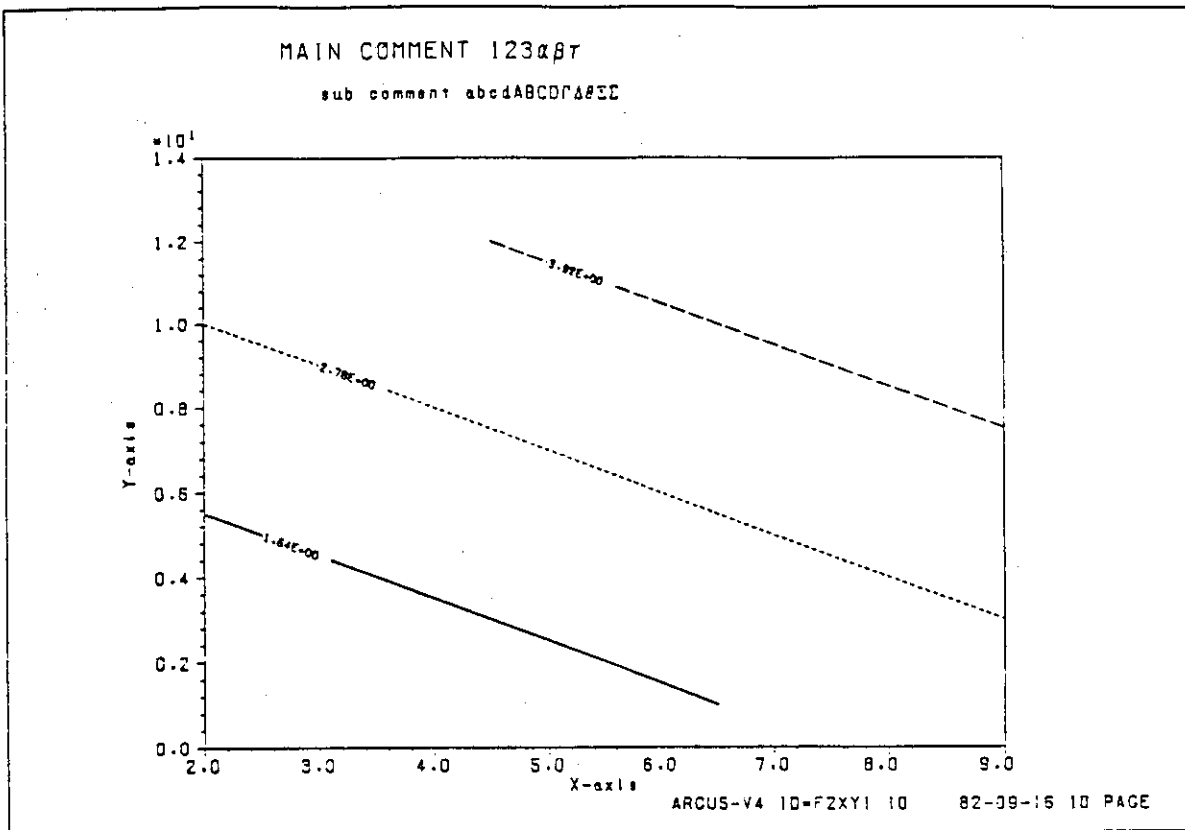
2. 図形「F2XYL」の例

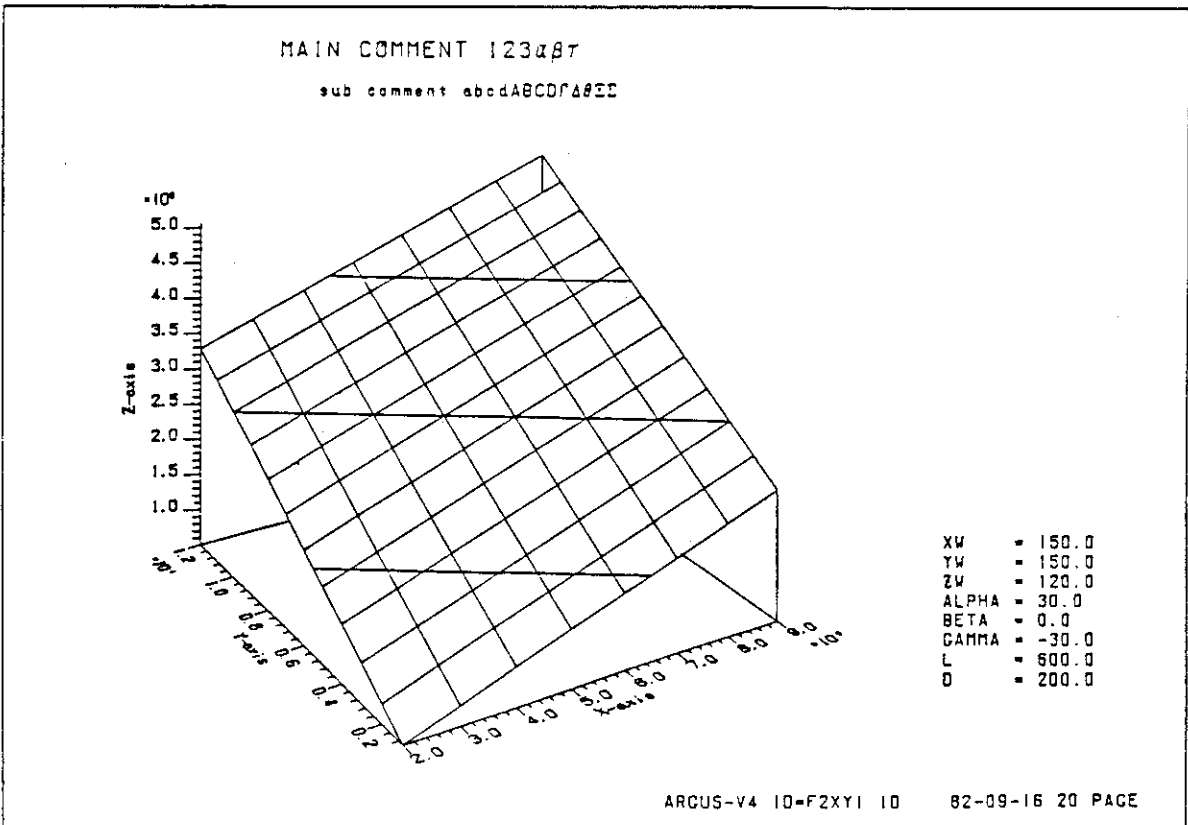
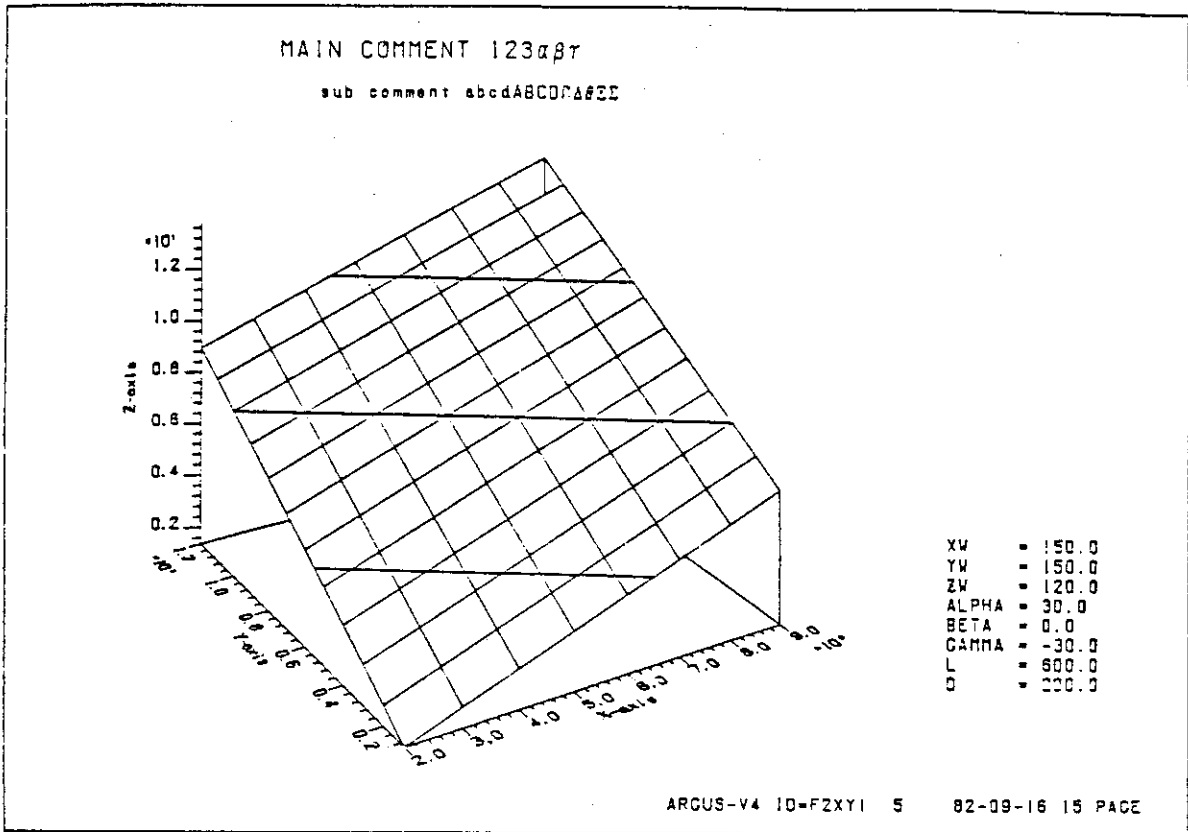
```

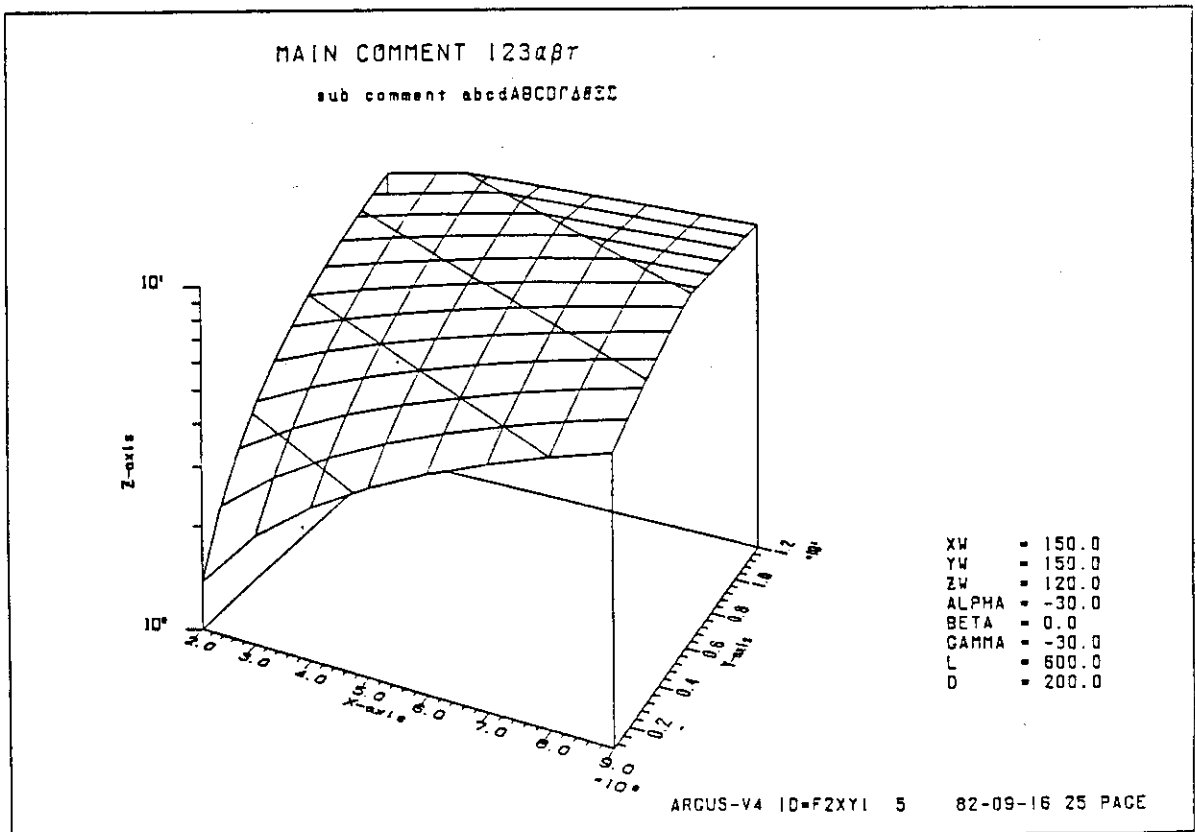
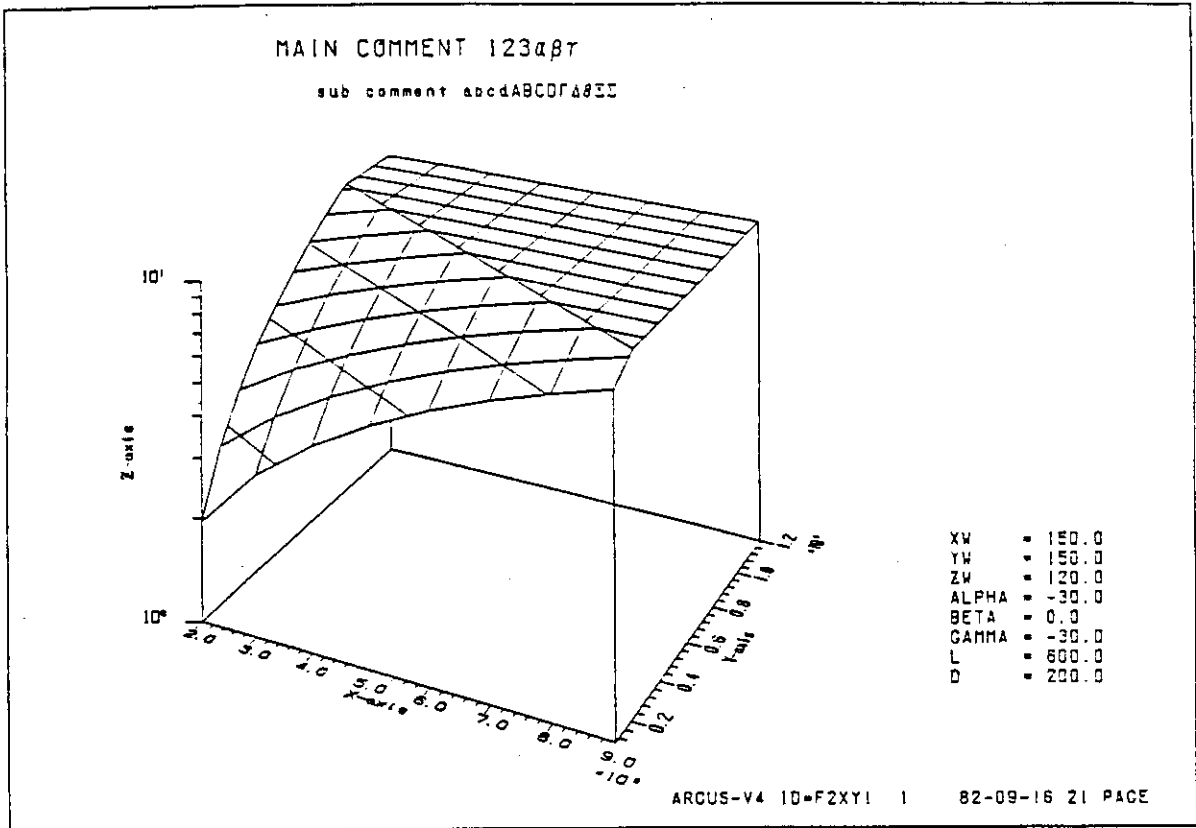
&GNAME F2XYL
&GSCRT 2
&BASE R(2:9),RY(3:11)
&FUNC FXY(R,RY)
&CDS
*MAINCM ICM='MAIN COMMENT 123*121,122,123*3'
*SUBCM ICM='SUB* COMMENT' "ABCD"ABCD*145,146,147,149,151*3'
*VIEWP A1=30.0,A2=0.0,A3=-30.0,A4=600.0,A5=200.0,N1=1
*HVALUE N1=-5
*MKIND N1=5,IV1=(1/1,2/2,3/3,4/4,5/5)
*XAXIS N3=1,A1=3.0,ICM='X-"AXIS"3'
*YAXIS N3=1,A1=3.0,ICM='Y-"AXIS"3'
*MLTAXS N3=1,A1=3.0,ICM='Z-"AXIS"3'
*XWYWGHT N1=1
*CUT CONTOUR
*OUT BRDEYE
&CDS
*VIEWP A1=-30.0,A2=0.0,A3=-30.0,A4=600.0,A5=200.0,N1=1
*XAXIS N3=1,A1=3.0,ICM='X-"AXIS"3'
*YAXIS N3=1,A1=3.0,ICM='Y-"AXIS"3'
*MLTAXS N3=2,A1=3.0,ICM='Z-"AXIS"3'
*MLTSCL N1=1,A1=1.0,A2=10.0
*OUT BRDEYE
&44

```



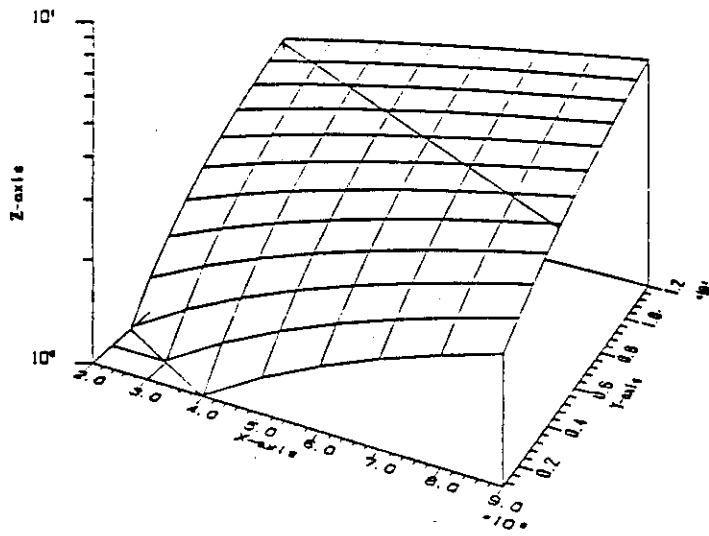






MAIN COMMENT 123aβr

sub comment abcdABCDΓΔΘΣ



XW	=	150.0
YW	=	150.0
ZW	=	120.0
ALPHA	=	-30.0
BETA	=	0.0
GAMMA	=	-30.0
L	=	500.0
D	=	200.0

ARCUS-V4 ID=F2XY1 10 82-09-16 30 PAGE

4. ま と め

トカマクコードのライブラリ・システム化の一環として汎用図形処理ユーティリティ GPREP / LIBGR の開発を行った。これらユーティリティルーチンを利用することにより、ライブラリシステムから作り出された多種多様なユーザプログラムに対応する図形出力機能が得られた。本報告書は、GPREP / LIBGR を利用する際の使用手引書となるものである。

GPREP / LIBGR はライブラリシステムのコードのみならず、一般のコードにおいても簡単に利用することができる。GPREP で定まるデータ形式で出力された図形データファイルについては、複数のファイルの結果を一枚の図に編集することのできるユーティリティも利用することも可能である。又、GPREP の機能は単に図形データを出力するだけではなく、汎用的なデータベース作成機能をも持っている。今後、GPREP のデータベース機能を一層充実する予定である。

謝 辞

GPREP / LIBGR の開発に際しては松下 樹氏 (数理技研) の努力に負う所が大きく、ここに記して感謝に代えます。コマンドプロシジャの整備については、木原和久氏 (日本ソフトウェア開発株式会社) に担当して頂き、ここに感謝します。トカマクコード・ライブラリ・システム整備計画全般を通じての、臨界プラズマ研究部吉川允二部長の御激励に感謝します。

参考文献

- 1) T. Takeda, T. Tsunematsu and S. Tokuda, " Integrative graphic subroutine package ARGUS - V4 ", Computer Phys. Commu. **34**, 15 (1984)
- 2) 平山俊雄, 他, "一次元トカマク輸送コード・ライブラリシステム LIBJT60", JAERI - M 82 - 204 (1982)

4. ま と め

トカマクコードのライブラリ・システム化の一環として汎用図形処理ユーティリティ GPREP / LIBGR の開発を行った。これらユーティリティルーチンを利用することにより、ライブラリシステムから作り出された多種多様なユーザプログラムに対応する図形出力機能が得られた。本報告書は、GPREP / LIBGR を利用する際の使用手引書となるものである。

GPREP / LIBGR はライブラリシステムのコードのみならず、一般のコードにおいても簡単に利用することができる。GPREP で定まるデータ形式で出力された図形データファイルについては、複数のファイルの結果を一枚の図に編集することのできるユーティリティも利用することも可能である。又、GPREP の機能は単に図形データを出力するだけではなく、汎用的なデータベース作成機能をも持っている。今後、GPREP のデータベース機能を一層充実する予定である。

謝 辞

GPREP / LIBGR の開発に際しては松下 樹氏 (数理技研) の努力に負う所が大きく、ここに記して感謝に代えます。コマンドプロシジャの整備については、木原和久氏 (日本ソフトウェア開発株式会社) に担当して頂き、ここに感謝します。トカマクコード・ライブラリ・システム整備計画全般を通じての、臨界プラズマ研究部吉川允二部長の御激励に感謝します。

参考文献

- 1) T. Takeda, T. Tsunematsu and S. Tokuda, " Integrative graphic subroutine package ARGUS - V4 ", Computer Phys. Commu. **34**, 15 (1984)
- 2) 平山俊雄, 他, " 一次元トカマク輸送コード・ライブラリシステム LIBJT60 ", JAERI - M 82 - 204 (1982)

4. ま と め

トカマクコードのライブラリ・システム化の一環として汎用図形処理ユーティリティ GPREP / LIBGR の開発を行った。これらユーティリティルーチンを利用することにより、ライブラリシステムから作り出された多種多様なユーザプログラムに対応する図形出力機能が得られた。本報告書は、GPREP / LIBGR を利用する際の使用手引書となるものである。

GPREP / LIBGR はライブラリシステムのコードのみならず、一般のコードにおいても簡単に利用することができる。GPREP で定まるデータ形式で出力された図形データファイルについては、複数のファイルの結果を一枚の図に編集することのできるユーティリティも利用することも可能である。又、GPREP の機能は単に図形データを出力するだけではなく、汎用的なデータベース作成機能をも持っている。今後、GPREP のデータベース機能を一層充実する予定である。

謝 辞

GPREP / LIBGR の開発に際しては松下 樹氏 (数理技研) の努力に負う所が大きく、ここに記して感謝に代えます。コマンドプロシジャの整備については、木原和久氏 (日本ソフトウェア開発株式会社) に担当して頂き、ここに感謝します。トカマクコード・ライブラリ・システム整備計画全般を通じての、臨界プラズマ研究部吉川允二部長の御激励に感謝します。

参考文献

- 1) T. Takeda, T. Tsunematsu and S. Tokuda, " Integrative graphic subroutine package ARGUS - V4 ", Computer Phys. Commu. **34**, 15 (1984)
- 2) 平山俊雄, 他, " 一次元トカマク輸送コード・ライブラリシステム LIBJT60 ", JAERI - M 82 - 204 (1982)