1983

Archaeological applications of the PL1022 Data Base Management System and GRAFIX mapping program

Dori M. Passmann
The University of Montana

ARCHAEOLOGICAL APPLICATIONS OF THE PL1022 DATA BASE MANAGEMENT SYSTEM AND GRAFIX MAPPING PROGRAM

By

Dori M. Passmann

B.A., Southern Illinois University, 1980

Presented in partial fulfillment of the requirements for the degree of

Master of Arts

UNIVERSITY OF MONTANA

1983

Approved by:

Chair, Board of Examiners

Dean, Graduate School

11-7-83

Date

ProQuest LLC.
789 East Eisenhower Parkway
P.O. Box 1346
Ann Arbor, MI 48106 - 1346

INTRODUCTION

In this paper I will discuss the adaptation of two existing computer programs to the management of archaeological information. Computers are ubiquitous in our society and are designed to handle large volumes of data with ease. Archaeology, now more than ever, generates large amounts of data that, in some form or other, need to be made accessible to people involved in or with the field. It seems logical to enlist the computer in providing services for easily accessing and distributing needed information.

The PL1022 Data Base Management System (PL1022) is used here to store and retrieve bibliographic information on archaeological surveys done in Montana. The data are retrieved via various keyed attributes thus making it a very useful tool. This file is still largely incomplete since seven counties were selected as a test and not all the available data have been entered into the computer at this time.

GRAFIX is a mapping program capable of producing maps of the entire state, of any county, or of selected townships and ranges. This program enables us to plot archaeological sites and other information in a variety of formats. Many of the capabilities of this complex program remain to be explored.

General guidelines for the output format were established in consultation with the Montana State Historic Preservation Office, after evaluating statewide archaeological needs.

GRAFIX and PL1022, if used in conjunction with the Montana Statewide Site Files, should be able to supply most of the information requested by people who work with or around archaeological data.

II. THE PL1022 and THE BIBLIOGRAPHY

The PL1022 manages a data base consisting of an annotated bibliography of archaeological surveys and the relevant legal locations for each survey. Each survey is a separate entry. There is no limit on the size of the surveys entered. They range from small well-pad inventories to large multi-county projects. The bibliography utilizes two data files and two programs.

There are two excellent published manuals: the PL1022 Data Base Management System Primer (1982) and the User's Reference Manual (1981). Both are distributed by Software House. The Primer is ideally suited for those unsophisticated in the use of computers and is all a user needs to manipulate the bibliography files.

In the following paragraphs I will be discussing each of the files, their setup, problems, and how the files interact with each other.

The first data file is called LEGALS. It contains the legal locations for the surveys. Each entry is formatted as follows:

Survey ID (SID)
Township (TOWN)
Range (RANGE)
Section (SECT)

The format has a high cost in space because each entry can contain only one SID. Thus if one township, range, and section had had six surveys, this would require six different entries. The file was formatted this way to make it easy to add new records, to change the existing records, and to simplify programming.

The second data file is SURVEYS; it is the actual bibliography. The entry format is as follows:

Survey ID (SID)
Author/Editor Lastname (LN)
Author/Editor Firstname (FN)
Survey Report Date (RDATE)
Survey Report Title (TITLE)
Publisher (PUB)
Place of Publication (PUBL)
Annotation (ANNO)
Agency Sponsor (SP)
Agency Local Office (LOCAL)
Survey Date (SDATE) -- year only
Pages (PAGES)
Volumes (VOL)
Survey Class (CLASS) -- BLM survey classes
Principal Investigator (PI)
Field Supervisor (FS)

The standard annotated bibliograpy format was expanded to provide some idea of survey characteristics. For instance, knowing CLASS or SDATE may indirectly provide information on the quality of work, or knowing FS may lend some credibility to the work.

The survey classes (CLASS) are supposed to be based on BLM designations. Since no one seems to have the same definition of these three classes I have tried to find the core meaning. In this paper Class I refers to a literary search only. Class II is any partial survey; systematic, random, or otherwise. Class III is 100% surveys, test excavations, etc. As different people update the bibliography, their personal definitions of the classes will alter the above definitions.

The two programs discussed below must be used within the 1022 environment. Thus after logging in, the user must access the 1022. These programs were designed to minimize work for the user and to reduce the risk of error.

The first program is called SBIB (see Appendix I). This called program requires the most work from the user because the user must establish the selection group to be printed. For the sample SBIB output (see Appendix II), the selection group consisted of all survey reports written or edited by Beckes in 1982. After the selection group has

been established, then the command *USE SBIB will produce the output seen in Appendix II.

Example showing login procedure and how to establish the selection group for Appendix II.

```
@LOGIN ANTH.PASSMANN-D  (password)
@1022:1022               - gets user into the 1022
                           environment
*OPEN SURVEY             - makes the file available to the
                           user
*FIND ALL                - establishes a selection group of
                           all records
 477 RECORDS.            - the system informs user as to the
                            number of records in the group
*FIND LN EQ BECKES AND RDATE EQ 1982
                         - establishes the selection group
                           for Appendix II
 4 RECORDS FOUND.        - informs user of how many records
                           are in the group
*USE SBIB                - formats and prints the
                           information as seen in Appendix
                           II.
```

Selection groups can be established using any of the attributes in the SURVEYS file. For example, a command to find all Class 2 surveys will create a group with Class 2 surveys only, or a user can make a selection group based on surveys done between 1973 and 1976, inclusive. This program sorts the output first by last name, then by report date, last by title, and then formats it for printing. The output is not seen on the terminal but is sent to an .LST file, a disk file found in the user's directory.

The second program is named LEGSUR (see Appendix III). The user could not ask for a simpler program. Upon entering 1022, issue the *USE LEGSUR command. The program asks for

the townships, ranges, and sections showing an example of how the input is to be formatted. The program then goes through the LEGALS file matching the townships, ranges, and sections. As it finds the matching records it checks the SIDs for repeats, any repeated SIDs are dropped in order to eliminate multiple listings of the same bibliographic entry. After deleting redundant entries, the program goes into the SURVEY file and matches SIDs from the LEGALS file. The information is formatted and then printed. Sample output from LEGSUR is seen in Appendix IV. Note that first the requested townships, ranges, and sections are listed with their corresponding SIDs and with the redundant entries left in. The bibliographic entries are then listed minus any possible repeats. Since the program processes the records one at a time, sorting the output file is not possible. Maybe in the future a feature will be created in the software to permit sorting under these circumstances. If sorting is imperative the user can use the Dec20 system SORT command outside of the 1022 environment. The program prints the output into an .LST file.

APPENDIX I Program SBIB

```
SORT LN RDATE TITLE.
INIT 5 SBIB.LST.
REPORT START.
  SECTION INITIAL.
    HEADING ON 5 PRINT
    'REFERENCES TO THE SURVEYS'
    FORMAT 10X A // END.
  SECTION GETREC.
  SECTION HEADING.
  SECTION PRINT.
    PRINT ON 5 LN FN SID
    FORMAT 10X A ',' X A 9X I4 END.
    PRINT ON 5 RDATE TITLE '.  ' PUB ', ' PUBL
    FORMAT 13X I4 2X ST60 SAS SAT SAS SAT / END.
    PRINT ON 5 ANNO
    FORMAT 19X ST60 / END.
      PRINT ON 5 SDATE
      FORMAT 19X 'SURVEY YEAR: ' I4 END.
      PRINT ON 5 PI FS
      FORMAT 19X 'PRINCIPLE INVESTIGATOR:  ' A / 19X 'FIELD
       SUP: ' A END.
      PRINT ON 5 SP ', ' LOCAL
      FORMAT 19X 'SPONSOR: ' ST47 SAS SAT END.
      PRINT ON 5 CLASS PAGES VOLUMES
      FORMAT 19X 'CLASS: ' I 5X 'PAGES: ' I5 5X 'VOL: ' I
       // END.
  SECTION TOTALS.
  SECTION FINAL.
REPORT END.
RELEASE.
```

APPENDIX II Sample SBIB Output

REFERENCES TO THE SURVEYS

BECKES, MICHAEL 144
1982 CLINE 1, STAGVILLE 1, ASPEN 82-1, PARKS 1, AND OVERSTORY 4 TIMBER SALES. CUSTER NATIONAL FOREST, BILLINGS, MT

AN INTENSIVE CULTURAL RESOURCE RECONNAISSANCE OF SIX SMALL TIMBER SALE PLOTS IN THE BELL TOWER DIVIDE AREA OF THE LONG PINES UNIT AND IN THE EKALAKA HILLS LOCATED NO CULTURAL RESOURCES.

SURVEY YEAR: 1982
PRINCIPLE INVESTIGATOR: MICHAEL BECKES
FIELD SUP: MICHAEL BECKES
SPONSOR: CUSTER NATIONAL FOREST, SIOUX RD, BILLINGS, MT
CLASS: 3 PAGES: 3 VOL: 1

BECKES, MICHAEL 145
1982 CROOKED CREEK TIMBER SALES/CR SURVEYS. CUSTER NATIONAL FOREST, BILLINGS, MT

AN INTENSIVE PEDESTRIAN RECONNAISSANCE OF A PROPOSED TIMBER SALE AREA IN THE EAST FLANK OF BIG PRYOR MOUNTAIN WAS CONDUCTED. NO CULTURAL RESOURCES WERE LOCATED.

SURVEY YEAR: 1982
PRINCIPLE INVESTIGATOR: MICHAEL BECKES
FIELD SUP: MICHAEL BECKES
SPONSOR: CUSTER NATIONAL FOREST, BEARTOOTH RD, BILLINGS, MT
CLASS: 3 PAGES: 4 VOL: 1

BECKES, MICHAEL 143
1982 CULTURAL RESOURCE SERVICE VISITS: KING, SHEEP, BAINBRIDGE, LOYNING, BIG, ANGUS, AND UPPER RANGER STATION SPRINGS, LOYNING RESERVOIR AND EAST PRYOR ELECTRONIC SITE. CUSTER NATIONAL FOREST, BILLINGS, MT

A FIELD EXAMINATION OF NINE PROPOSED STOCK WATERING FACILITIES AND ONE ELECTRONIC SITE IN THE PRYOR MOUNTAIN UNIT WAS CONDUCTED. THREE OF THE PROPOSED UNITS CONTAINED LITHIC MATERIALS; THESE DID NOT WARRANT FURTHER MEASURES.

SURVEY YEAR: 1982

PRINCIPLE INVESTIGATOR: MICHAEL BECKES
FIELD SUP: MICHAEL BECKES
SPONSOR: CUSTER NATIONAL FOREST, BEARTOOTH RD, BILLINGS, MT
CLASS: 3 PAGES: 5 VOL:1

BECKES, MICHAEL 147
1982 CULTURAL RESOURCE SURVEY: STEVENS LAND EXCHANGE. CUSTER NATIONAL FOREST, BILLINGS, MT

AN INTENSIVE FIELD EXAMINATION OF SIX PARCELS OF FEDERAL LAND BEING CONSIDERED IN THE PROPOSED STEVENS EXCHANGE WAS CONDUCTED. PARCELS 1,2,3, AND 6 CONTAINED SITES WHICH WERE RECORDED.

SURVEY YEAR: 1982
PRINCIPLE INVESTIGATOR: MICHAEL BECKES
FIELD SUP: MICHAEL BECKES
SPONSOR: CUSTER NATIONAL FOREST, ASHLAND RD, BILLINGS, MT
CLASS: 3 PAGES: 27 VOL: 1

APPENDIX III Program LEGSUR

```
INIT 5 LEGSUR.LST.
OPEN LEGALS SURVEY.
FIND ALL.
PL1022 START.
DEFINE TEXT 3 T(10) R(10) TOWN RANGE.
DEFINE TEXT 2 S(10) SECT.
DEFINE INTEGER X I SNO(50) Y Z SID.
PRINT "ENTER TOWNSHIPS, RANGES, AND SECTIONS (01N,23E,02).
END WITH 3 X'S, XXX."
FORMAT 3X A / END.
REPEAT
  LET I EQ I+1.
  ACCEPT T(I) R(I) S(I).
UNTIL Y(I) EQ XXX.
PRINT ON 5 ' ' FORMAT A / END.
REPEAT
  DBSET LEGALS.
  LET X EQ X+1.
  FIND TOWN EQ T(X) AND RANGE EQ R(X) AND SECT EQ S(X).
  PRINT ON 5 TOWN RANGE SECT SID
  FORMAT 5X "T" A3 " R" A3 " SECT." A2 I6 END.
UNTIL X EQ I-1.
LET X EQ 0.
REPEAT
  LET X EQ X+1.
  FIND TOWN EQ T(X) AND RANGE EQ R(X) AND SECT EQ S(X).
  REPEAT
    GETREC LEAVE.
    LET Y EQ Y+1.
    LET SNO(Y) EQ SID.
      IF Y GT 1 THEN
        WHILE Z LT Y-1 DO
          LET Z EQ Z+1.
          IF SID EQ SNO(Z) THEN
            DROP.
            LET Y EQ Y-1.
            GOTO NEXT.
          ENDIF.
        ENDWHILE.
      ENDIF.
    MAP TO SURVEY VIA SID TO SID.
    CALL RT1.
    NEXT:  LET Z EQ 0.
  UNTIL 1 EQ 0.
UNTIL X EQ I-1.
CLOSE.  CLOSE.
RELEASE.
PL1022 STOP.
  RT1:  PRINT ON 5 ' ' FORMAT A END.
```

```
  PRINT ON 5 LN FN SID
  FORMAT 10X A ',' X A 9X I4 END.
  PRINT ON 5 RDATE TITLE '.  ' PUB ', ' PUBPL
  FORMAT 13X I4 2X ST60 SAS SAT SAS SAT / END.
  PRINT ON 5 ANNO
  FORMAT 19X ST60 / END.
    PRINT ON 5 SDATE
    FORMAT 19X 'SURVEY YEAR:  ' I4 END.
    PRINT ON 5 PI FS
    FORMAT 19X 'PRINCILPE INVESTIGATOR:  ' A / 19X 'FIELD
      SUP: ' A END.
    PRINT ON 5 SP ', ' LOCAL
    FORMAT 19X 'SPONSOR: ' ST47 SAS SAT END.
    PRINT ON 5 CLASS PAGES VOLUMES
    FORMAT 19X 'CLASS: ' I 5X 'PAGES: ' I5 5X 'VOL: ' I
      END.
  RETURN.
PL1022 END.
```

APPENDIX IV Sample Output from LEGSUR

T09S R22E SECT.03 54
T09S R22E SECT.18 54
T07S R57E SECT.24 426

LOENDORF, LAWRENCE 54
1967 A PRELIMINARY ARCHAEOLOGICAL SURVEY OF THE CLARK FORK RIVER, CARBON COUNTY, MONTANA. UNIVERSITY OF MONTANA, MISSOULA, MT

THE AREA UNDER STUDY FOR MY MASTER'S THESIS WAS THE CLARK FORK OF THE YELLOWSTONE RIVER VALLEY. GIVES A REPORT OF THE LOCATION AND GEOGRAPHICAL FEATURES AND DEVELOPS A DISTRIBUTION OF THE SITES.

SURVEY YEAR: 1964
PRINCIPLE INVESTIGATOR: LAWRENCE LOENDORF
FIELD SUP: LAWRENCE LOENDORF
SPONSOR: UNIVERSITY OF MONTANA, MISSOULA, MONTANA
CLASS: 3 PAGES: 156 VOL: 1

CLARK, JERRY 426
1980 CATTAIL PIT, JP'S PIT, AND PETER'S PIT. BLM, MILES CITY

ALL AREAS TO BE DISTURBED BY CONSTRUCTION AND USE OF THE PITS EXAMINED. NO CULTURAL RESOURCES WERE LOCATED.

SURVEY YEAR: 1980
PRINCIPLE INVESTIGATOR: JERRY CLARK
FIELD SUP: JERRY CLARK
SPONSOR: BLM, MILES CITY
CLASS: 3 PAGES: 5 VOL:1

III. The GRAFIX mapping program

The GRAFIX program came from the Montana State Department of Administration, Computer Services in Helena, Montana. The program is composed of many subroutines to facilitate producing maps of the state or parts of the state in several formats. Three files accompanied the program:

(1) FEATURE.DAT is the digitized coordinates for county outlines, the state outline, and the outlines for the national forests.

(2) TOWNSHIP.DAT is the digitized coordinates of all the townships and ranges in the state.

(3) PLACES.DAT has single latitude/longitude coordinates of the center of each incorporated and unincorporated place as shown on the USGS 1:2,500.000 map series.

The manual for GRAFIX, retyped and with changes made for use with the Dec20 and Calcomp plotter, is appended to this paper.

My task with GRAFIX was to adapt it to the University of Montana's Decsystem20 and to provide directions for plotting point data on maps. GRAFIX was written in FORTRAN for an IBM system, so getting it to run on our system was not easy. The program was given to the University of Montana with no information on the logic of how it ran and no explanation on what parts of the subroutines did. The

manual which accompanied the program was of no help with understanding the program code or its logic. The following discussions and explanations of GRAFIX are illustrated by an accompanying series of maps and log listings. Thus each map and listing will be followed by a discussion on how to interpret the control data file, the selection and testing parameters, size considerations, where I ran into problems, etc. In the end I hope it will be easy for others to produce maps.

Please note the following:

FOR01.DAT is the file for the control data set.

FORPLT.OUT is the file you send to the plotter. It will appear in your directory after executing GRAFIX.

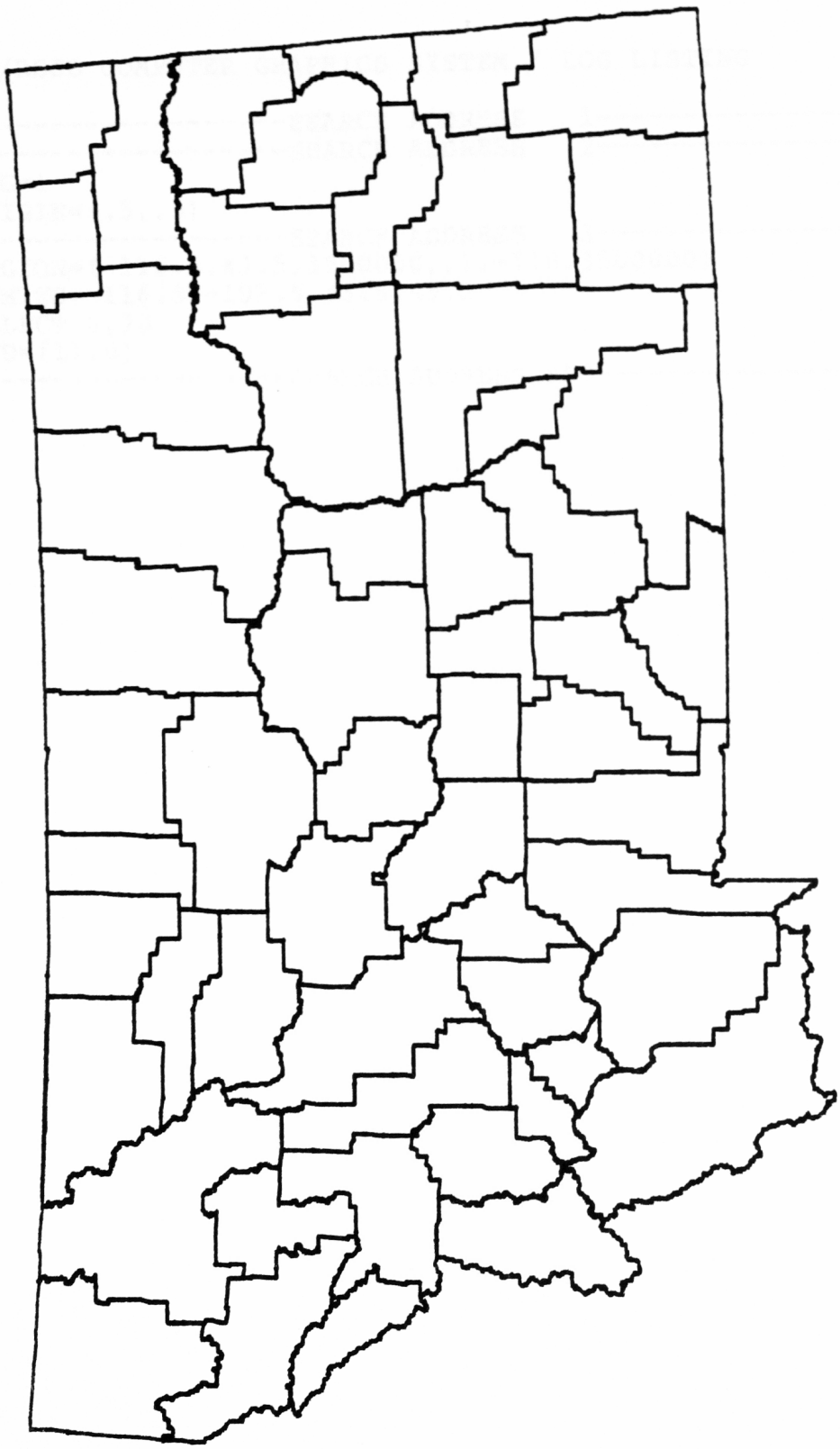
FOR22-24.DAT will hold the data you wish to input. These files must conform to one of six file types discussed under *SELECT in the manual.

20 is the reference number to the FEATURE.DAT file.

21 is the reference number to the TOWNSHIP.DAT file.

****I highly recommend that readers familiarize themselves with the GRAFIX manual attached to the end of this paper before reading the following map discussions.

MAP I



```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1--------------------
--------------------SEARCH ADDRESS   2--------------------
*LOG=1
*ORIGIN=(.5,.5)
--------------------SEARCH ADDRESS   3--------------------
*REGION=(-116.5,43.5,3500000,,1,-110,3500000)
*LIMITS=-116.5,-103.5,43.5,49.5
*SELECT 6,20
*END=(11,0)
--------------------SEARCH ADDRESS 999--------------------
```

```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1--------------------
--------------------SEARCH ADDRESS   2--------------------
*LOG=2
*ORIGIN=(.5,.5)
--------------------SEARCH ADDRESS   3--------------------
*REGION=(-116.5,43.5,2000000,,1,-110,2000000)
*LIMITS=-116.5,-103.5,43.5,49.5
*SELECT 6,20
(F2='COUN':1)
*END=(11,0)
--------------------SEARCH ADDRESS 999--------------------
```

```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1--------------------
--------------------SEARCH ADDRESS   2--------------------
*LOG=3
*ORIGIN=(5,5)
--------------------SEARCH ADDRESS   3--------------------
*REGION=(-116.5,43.5,1000000,,1,-110,1000000)
*LIMITS=-116.5,-103.5,43.5,49.5
*SELECT 6,20
(F2='COUN':1)
  L:    1.0  N=1754  (X  8.62, 16.11/Y  3.09, 10.11/1  8.62,  9.09/N  8.62,  9.09)     COUNTY  BEAVERHEAD
  L:    2.0  N= 435  (X 12.38, 16.55/Y  4.52,  9.58/1 12.45,  8.62/N 12.45,  8.62)     COUNTY  MADISON
  L:    3.0  N= 609  (X 15.17, 18.31/Y  3.54, 10.99/1 15.17,  9.30/N 15.17,  9.30)     COUNTY  GALLATIN
  L:    4.0  N=1116  (X  6.67,  9.96/Y  8.12, 13.37/1  7.18, 11.28/N  7.18, 11.28)     COUNTY  RAVALLI
  L:    5.0  N= 452  (X  9.95, 12.93/Y  9.26, 11.46/1  9.96, 10.10/N  9.96, 10.10)     COUNTY  DEER LODGE
  L:    6.0  N= 544  (X 11.26, 13.99/Y  8.62, 11.06/1 11.77, 10.29/N 11.77, 10.29)     COUNTY  SILVER BOW
  L:    7.0  N= 672  (X 12.82, 15.64/Y  9.12, 12.70/1 12.86, 11.04/N 12.86, 11.04)     COUNTY  JEFFERSON
  L:    8.0  N= 710  (X 15.26, 17.50/Y  9.45, 13.59/1 15.30, 12.68/N 15.30, 12.68)     COUNTY  BROADWATER
  L:    9.0  N= 251  (X 17.46, 21.31/Y  5.76, 10.98/1 19.84, 10.94/N 19.84, 10.94)     COUNTY  PARK
  L:   10.0  N=  72  (X 19.80, 22.45/Y  6.52, 11.12/1 20.69, 11.10/N 20.69, 11.10)     COUNTY  SWEET GRASS
  L:   11.0  N= 174  (X 20.50, 24.24/Y  6.52, 10.72/1 20.69,  7.29/N 20.69,  7.29)     COUNTY  STILLWATER
  L:   13.0  N= 379  (X 21.31, 26.57/Y  5.77,  8.58/1 21.31,  6.52/N 21.31,  6.52)     COUNTY  CARBON
  L:   14.0  N= 279  (X 23.96, 28.41/Y  7.81, 12.39/1 24.24,  8.45/N 24.24,  8.45)     COUNTY  YELLOWSTONE
  L:   15.0  N= 275  (X 24.68, 32.29/Y  5.81, 10.44/1 24.87,  7.82/N 24.87,  7.82)     COUNTY  BIG HORN
  L:   16.0  N= 172  (X 27.53, 30.04/Y  9.72, 12.38/1 30.02, 10.31/N 30.02, 10.31)     COUNTY  TREASURE
  L:   17.0  N= 444  (X 27.00, 32.44/Y  6.74, 14.12/1 32.44, 14.12/N 32.44, 14.12)     COUNTY  ROSEBUD
  L:   18.0  N=  63  (X 32.16, 36.15/Y  6.02,  9.68/1 32.30,  6.02/N 32.30,  6.02)     COUNTY  POWDER RIVER
  L:   19.0  N=  72  (X 32.31, 36.60/Y  9.49, 14.22/1 32.33, 10.41/N 32.33, 10.41)     COUNTY  CUSTER
  L:   20.0  N=  62  (X 35.99, 39.18/Y  6.23, 11.30/1 36.08,  6.23/N 36.08,  6.23)     COUNTY  CARTER
  L:   21.0  N=  35  (X 36.08, 38.88/Y 10.26, 13.71/1 36.25, 11.23/N 36.25, 11.23)     COUNTY  FALLON
  L:   22.0  N= 450  (X  6.40, 10.84/Y 13.26, 17.43/1  6.62, 13.77/N  6.62, 13.77)     COUNTY  MISSOULA
  L:   23.0  N= 474  (X  3.60,  7.47/Y 13.60, 17.27/1  3.98, 17.19/N  3.98, 17.19)     COUNTY  MINERAL
  L:   24.0  N= 576  (X  9.01, 11.54/Y 10.10, 13.99/1 11.49, 11.46/N 11.49, 11.46)     COUNTY  GRANITE
  L:   25.0  N= 370  (X 10.35, 13.76/Y 11.43, 17.36/1 10.80, 14.70/N 10.80, 14.70)     COUNTY  POWELL
  L:   26.0  N=1109  (X 11.35, 16.20/Y 12.04, 18.84/1 15.33, 14.56/N 15.33, 14.56)     COUNTY  LEWIS & CLARI
  L:   27.0  N= 456  (X 14.59, 18.79/Y 13.72, 17.60/1 14.63, 16.84/N 14.63, 16.84)     COUNTY  CASCADE
  L:   28.0  N= 829  (X 15.72, 19.87/Y 10.94, 14.90/1 17.47, 10.99/N 17.47, 10.99)     COUNTY  MEAGHER
  L:   29.0  N= 322  (X 18.34, 21.46/Y 13.09, 16.33/1 18.73, 13.73/N 18.73, 13.73)     COUNTY  JUDITH BASIN
  L:   30.0  N= 160  (X 19.84, 22.54/Y 11.09, 13.41/1 19.87, 13.23/N 19.87, 13.23)     COUNTY  WHEATLAND
  L:   31.0  N= 809  (X 20.06, 25.70/Y 13.16, 18.03/1 20.69, 16.58/N 20.69, 16.58)     COUNTY  FERGUS
  L:   32.0  N=  48  (X 21.74, 24.40/Y 10.33, 13.43/1 23.67, 13.43/N 23.67, 13.43)     COUNTY  GOLDEN VALLEY
  L:   33.0  N= 272  (X 23.67, 27.41/Y 10.73, 13.53/1 24.81, 13.44/N 24.81, 13.44)     COUNTY  MUSSELSHELL
  L:   34.0  N=1095  (X 24.46, 27.23/Y 13.44, 17.19/1 24.85, 14.53/N 24.85, 14.53)     COUNTY  PETROLEUM
  L:   35.0  N=1288  (X 26.74, 32.44/Y 13.94, 19.05/1 27.04, 14.58/N 27.04, 14.58)     COUNTY  GARFIELD
  L:   36.0  N=  57  (X 32.16, 36.94/Y 12.93, 15.71/1 32.40, 14.79/N 32.40, 14.79)     COUNTY  PRAIRIE
  L:   37.0  N=  88  (X 34.30, 37.72/Y 14.36, 18.39/1 34.40, 15.70/N 34.40, 15.70)     COUNTY  DAWSON
  L:   38.0  N=  78  (X 36.89, 38.63/Y 13.54, 16.92/1 36.89, 14.44/N 36.89, 14.44)     COUNTY  WIBAUX
  L:   39.0  N= 801  (X  2.92,  8.20/Y 15.37, 20.66/1  3.00, 20.51/N  3.00, 20.51)     COUNTY  SANDERS
  L:   40.0  N= 582  (X  3.00,  7.36/Y 18.93, 23.94/1  7.08, 23.67/N  7.08, 23.67)     COUNTY  LINCOLN
  L:   41.0  N=1114  (X  5.98, 12.23/Y 17.31, 23.67/1  7.08, 23.67/N  7.08, 23.67)     COUNTY  FLATHEAD
  L:   42.0  N= 441  (X  7.10, 10.06/Y 15.41, 19.41/1  8.80, 19.18/N  8.80, 19.18)     COUNTY  LAKE
  L:   43.0  N= 595  (X 11.91, 16.56/Y 16.76, 19.61/1 14.82, 19.14/N 14.82, 19.14)     COUNTY  TETON
  L:   44.0  N= 275  (X 10.92, 16.57/Y 18.85, 21.12/1 14.34, 21.06/N 14.34, 21.06)     COUNTY  PONDERA
  L:   45.0  N= 529  (X 16.53, 22.07/Y 16.33, 20.22/1 16.57, 19.49/N 16.57, 19.49)     COUNTY  CHOTEAU
  L:   46.0  N= 337  (X 22.01, 25.77/Y 17.66, 23.31/1 25.76, 23.31/N 25.76, 23.31)     COUNTY  BLAINE
  L:   47.0  N= 649  (X 23.97, 28.85/Y 16.56, 23.40/1 28.79, 23.40/N 28.79, 23.40)     COUNTY  PHILLIPS
  L:   48.0  N= 584  (X 28.28, 32.97/Y 17.55, 23.54/1 31.89, 23.54/N 31.89, 23.54)     COUNTY  VALLEY
  L:   49.0  N= 439  (X 31.16, 34.83/Y 15.22, 19.70/1 31.20, 18.94/N 31.20, 18.94)     COUNTY  MCCONE
  L:   50.0  N= 436  (X 34.72, 38.39/Y 16.63, 20.05/1 34.74, 18.39/N 34.74, 18.39)     COUNTY  RICHLAND

  L:   51.0  N= 415  (X  8.98, 14.38/Y 20.35, 23.55/1 13.88, 23.35/N 13.88, 23.35)     COUNTY  GLACIER
  L:   52.0  N= 123  (X 14.33, 17.03/Y 19.87, 23.33/1 14.93, 23.32/N 14.93, 23.32)     COUNTY  TOOLE
  L:   53.0  N=  37  (X 16.57, 18.55/Y 19.47, 23.27/1 18.55, 23.26/N 18.55, 23.26)     COUNTY  LIBERTY
  L:   54.0  N=  56  (X 18.48, 22.25/Y 19.46, 23.26/1 20.69, 23.24/N 20.69, 23.24)     COUNTY  HILL
  L:   55.0  N=  35  (X 31.79, 35.29/Y 21.64, 23.71/1 32.21, 23.55/N 32.21, 23.55)     COUNTY  DANIELS
  L:   56.0  N= 626  (X 32.85, 38.16/Y 19.25, 21.86/1 32.88, 21.68/N 32.88, 21.68)     COUNTY  ROOSEVELT
  L:   57.0  N=  30  (X 34.92, 38.04/Y 21.13, 23.92/1 37.82, 23.92/N 37.82, 23.92)     COUNTY  SHERIDAN
*END=(11,0)
--------------------SEARCH ADDRESS 999--------------------
```

MAP I Discussion.

The three preceding log listings are produced by the *LOG function. They show what is printed for *LOG=1,2, and 3. For a discussion on the differences between the various log listings, see *LOG in the manual. If you omit *LOG the system defaults to the type "2" listing.

The *ORIGIN moves the map one half inch from the X and Y axes. We are currently having a problem with maps produced by GRAFIX and how they are drawn on the Calcomp plotter. This problem is manifested by the same map being cut in half on one run and coming out perfect on the next. The Computer Center is working on a solution for GRAFIX users.

In the *REGION function I set the logical coordinates (-116.5,43.5) to coincide with the physical origin set with *ORIGIN. The scaling factor is what produces changes in map sizes. A scale factor of:

1,000,000 produced a map approximately 35" x 20"

2,000,000 produced a map approximately 17.5" x 10"

5,000,000 produced the map seen at the beginning of this section.

The 1 (..,,1,..) signals that the projection is Transverse Mercator and (-110) signifies that the reference meridian is 110°W.

The *LIMITS function sets the boundaries for the acceptable X and Y values. If there is an error message stating the plot exceeds plotter limits, the user should first check the parameters of *LIMITS. If the error still occurs, check the parameters of *REGION.

The *SELECT 6,20 indicates that the format of the data file is Type 6 (see *SELECT Type 6 in the manual) and that the reference number is 20. I have set 20 as the reference number to the FEATURE.DAT file in the GRAFIX program with a FORTRAN Open statement.

(F2='COUN':1) establishes the selection and action parameters. As the program goes through the FEATURE file it looks in field 2 (F2) for the occurrence of 'COUN' (see Figure 2). The :1 indicates that when a 'COUN' is encountered it is to be drawn. Thus as each county is found, it gets plotted to produce the map shown.

*END indicates that this is the end of the control data set (which is located in FOR01.DAT). However, when it finishes plotting the program moves the pen off to the right of my finished plot.

Figure 2. A listing showing the numeric codes for the counties, the state, and the national forests. This listing will help with selection testing when using the *SELECT function. Note that the fields are broken into groups of 4 characters each. Thus field 1 (F1) is the ID number, field 2 (F2) will be 'coun', 'fede', 'stat', or 'n fo', field 4 (F4) will be the first four characters of the counties, etc. All of the following are found in the FEATURE.DAT file.

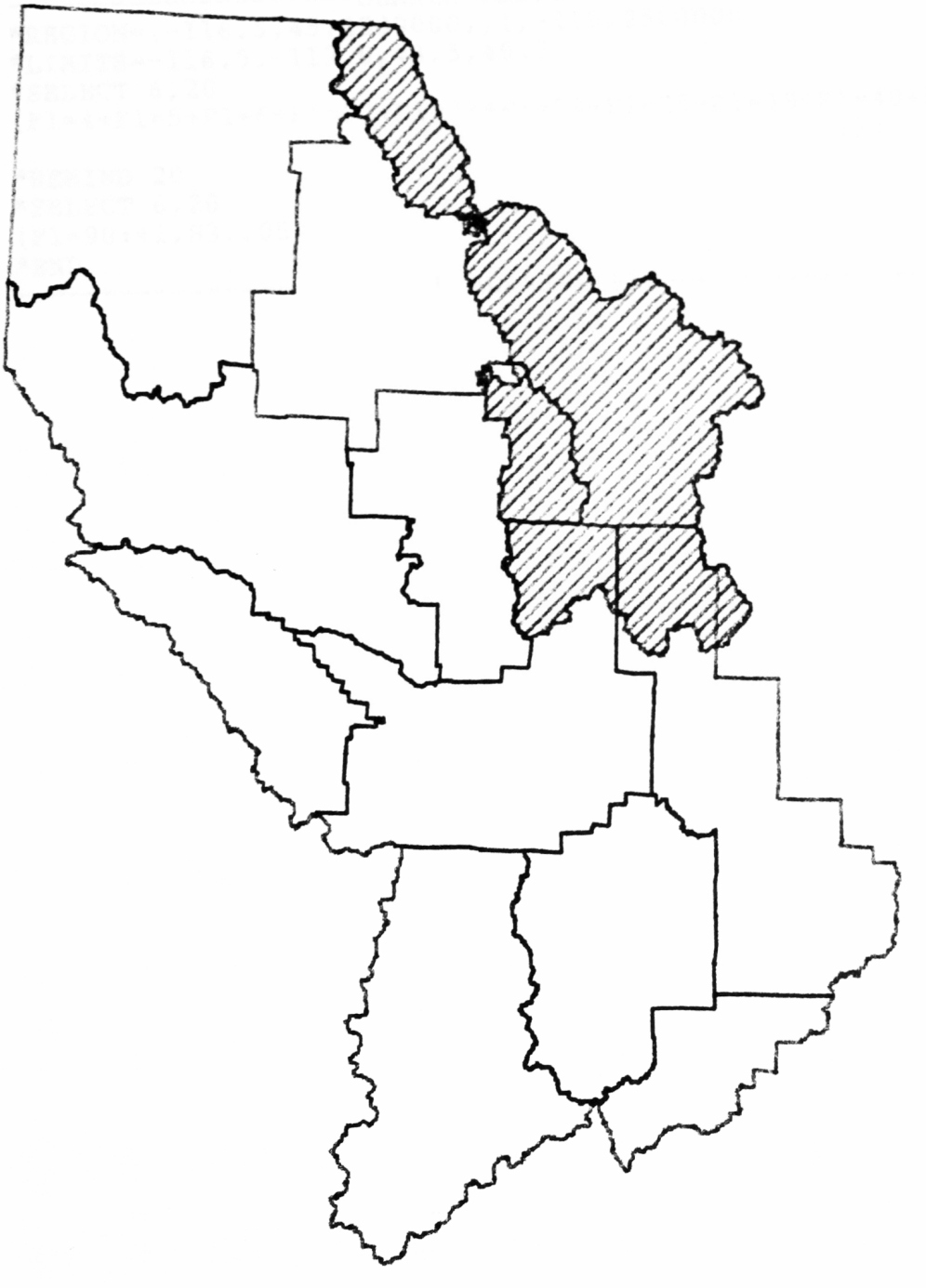
ID	HEADER	
1	COUNTY	BEAVERHEAD
2	COUNTY	MADISON
3	COUNTY	GALLATIN
4	COUNTY	RAVALLI
5	COUNTY	DEER LODGE
6	COUNTY	SILVER BOW
7	COUNTY	JEFFERSON
8	COUNTY	BROADWATER
9	COUNTY	PARK
10	COUNTY	SWEET GRASS
11	COUNTY	STILLWATER
12	FEDERAL	NAT PARKYELLOWST
13	COUNTY	CARBON
14	COUNTY	YELLOWSTONE
15	COUNTY	BIG HORN
16	COUNTY	TREASURE
17	COUNTY	ROSEBUD
18	COUNTY	POWDER RIVER
19	COUNTY	CUSTER
20	COUNTY	CARTER
21	COUNTY	FALLON
22	COUNTY	MISSOULA
23	COUNTY	MINERAL
24	COUNTY	GRANITE
25	COUNTY	POWELL
26	COUNTY	LEWIS & CLARK
27	COUNTY	CASCADE
28	COUNTY	MEAGHER
29	COUNTY	JUDITH BASIN
30	COUNTY	WHEATLAND
31	COUNTY	FERGUS
32	COUNTY	GOLDEN VALLEY
33	COUNTY	MUSSELSHELL
34	COUNTY	PETROLEUM
35	COUNTY	GARFIELD
36	COUNTY	PRAIRIE
37	COUNTY	DAWSON
38	COUNTY	WIBAUX
39	COUNTY	SANDERS
40	COUNTY	LINCOLN

41 COUNTY FLATHEAD
42 COUNTY LAKE
43 COUNTY TETON
44 COUNTY PONDERA
45 COUNTY CHOTEAU
46 COUNTY BLAINE
47 COUNTY PHILLIPS
48 COUNTY VALLEY
49 COUNTY MCCONE
50 COUNTY RICHLAND
51 COUNTY GLACIER
52 COUNTY TOOLE
53 COUNTY LIBERTY
54 COUNTY HILL
55 COUNTY DANIELS
56 COUNTY ROOSEVELT
57 COUNTY SHERIDAN
60 FEDERAL N FORESTBITTERRO
61 FEDERAL N FORESTCUSTER
62 FEDERAL N FORESTDEERLODG
63 FEDERAL N FORESTCUSTER
64 FEDERAL N FORESTBEAVERHE
65 FEDERAL N FORESTDEERLODG
66 FEDERAL N FORESTLOLO
67 FEDERAL N FORESTLOLO
68 FEDERAL N FORESTDEERLODG
69 FEDERAL N FORESTHELENA
70 FEDERAL N FORESTDEERLODG
71 FEDERAL N FORESTHELENA
72 FEDERAL N FORESTBEAVERHE
73 FEDERAL N FORESTBEAVERHE
74 FEDERAL N FORESTBEAVERHE
75 FEDERAL N FORESTLOLO
76 FEDERAL N FORESTLOLO
77 FEDERAL N FORESTDEERLODG
78 FEDERAL N FORESTDEERLODG
79 FEDERAL N FORESTBEAVERHE
83 FEDERAL N FORESTBEAVERHE
84 FEDERAL N FORESTBEAVERHE
85 FEDERAL N FORESTGALLATIN
86 FEDERAL N FORESTCUSTER
87 FEDERAL N FORESTGALLATIN
88 FEDERAL N FORESTLEWIS &C
89 FEDERAL N FORESTGALLITAN
90 FEDERAL N FORESTFLATHEAD
91 FEDERAL N FORESTFLATHEAD
92 FEDERAL N FORESTKOOTENAI
93 FEDERAL N FORESTKANIKSU
94 FEDERAL N FORESTKANIKSU
95 FEDERAL N FORESTKANIKSU
96 FEDERAL N FORESTLOLO

```
 97  FEDERAL N FORESTLOLO
 98  FEDERAL N FORESTLOLO
 99  FEDERAL N FORESTLOLO
100  FEDERAL N FORESTKOOTENAI
101  FEDERAL N FORESTLEWIS &C
102  FEDERAL N FORESTKANIKSU
104  STATE   MONTANA
106  FEDERAL N FORESTCUSTER N
298  FEDERAL N FORESTFLATHEAD
299  FEDERAL N FORESTLOLO NAT
300  FEDERAL N FORESTLOLO NAT
301  FEDERAL N FORESTLOLO NAT
302  FEDERAL N FORESTLOLO NAT
303  FEDERAL N FORESTLOLO NAT
304  FEDERAL N FORESTLOLO NAT
305  FEDERAL N FORESTLOLO NAT
306  FEDERAL N FORESTLOLO NAT
310  FEDERAL N FORESTLOLO NAT
311  FEDERAL N FORESTLOLO
312  FEDERAL N FORESTLOLO
320  FEDERAL N FORESTGALLATIN
410  N FORESTBITTERROOT NATIO
415  FEDERAL N FORESTHELENA N
416  FEDERAL N FORESTGALLITAN
417  FEDERAL N FORESTHELENA N
418  FEDERAL N FORESTLEWIS AN
419  FEDERAL N FORESTLEWIS AN
420  FEDERAL N FORESTLEWIS AN
485  FEDERAL N FORESTBEAVERHE
494  N FORESTHELENA NATIONA
495  N FORESTHELENA NATIONA
526  FEDERAL S FORESTSWAN RIV
527  FEDERAL N FORESTHELENA N
528  FEDERAL N FORESTLOLO NAT
529  FEDERAL N FORESTLOLO NAT
530  FEDERAL N FORESTKOOTENAI
531  FEDERAL N FORESTKOOTENAI
532  FEDERAL N FORESTKOOTENAI
533  FEDERAL N FORESTKOOTENAI
534  FEDERAL N FORESTKOOTENAI
535  FEDERAL N FORESTKOOTENAI
536  FEDERAL N FORESTKOOTENAI
537  FEDERAL N FORESTKOOTENAI
538  FEDERAL N FORESTKOOTENAI
539  FEDERAL N FORESTKOOTENAI
540  FEDERAL N FORESTKOOTENAI
541  FEDERAL N FORESTKOOTENAI
542  FEDERAL N FORESTKOOTENAI
543  FEDERAL N FORESTKOOTENAI
544  FEDERAL N FORESTKOOTENAI
545  FEDERAL N FORESTKOOTENAI
```

546	FEDERAL	N	FORESTKOOTENAI
547	FEDERAL	N	FORESTKOOTENAI
548	FEDERAL	N	FORESTKOOTENAI
549	FEDERAL	N	FORESTKOOTENAI
550	FEDERAL	N	FORESTFLATHEAD
551	FEDERAL	N	FORESTFLATHEAD
552	FEDERAL	N	FORESTFLATHEAD
553	FEDERAL	N	FORESTFLATHEAD
554	FEDERAL	N	FORESTFLATHEAD
555	FEDERAL	N	FORESTFLATHEAD
826	FEDERAL	NAT	FORSBITTERRT
827	FEDERAL	NAT	FORSBITTERRT
828	FEDERAL	NAT	FORSBITTERRT
829	FEDERAL	NAT	FORSBITTERRT
832	FEDERAL	NAT	FORSCUSTER
833	FEDERAL	NAT	FORSCUSTER
876	FEDERAL	N	FORESTLEWIS
877	FEDERAL	N	FORESTLEWIS
878	FEDERAL	N	FORESTLEWIS
918	FEDERAL	N	FORESTBEAVERHD
919	FEDERAL	N	FORESTBEAVERHD
920	FEDERAL	N	FORESTBEAVERHD
921	FEDERAL	N	FORESTBEAVERHD
1004	FEDERAL	N	FOREST HELENA
1005	FEDERAL	N	FOREST HELENA
1006	FEDERAL	N	FOREST HELENA
1083	FEDERAL	N	FORESTLEWIS AN
1084	FEDERAL	N	FORESTLEWIS AN
1085	FEDERAL	N	FORESTLEWIS AN
1113	FEDERAL	N	FORESTKOOTENAI
1122	FEDERAL	N	FORESTLOLO
1123	FEDERAL	N	FORESTKANIKSU

MAP II



```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1--------------------
--------------------SEARCH ADDRESS   2--------------------
*LOG=2
*ORIGIN=(0,0)
--------------------SEARCH ADDRESS   3--------------------
*REGION=(-116.5,45,2500000,,1,-110,2500000
*LIMITS=-116.5,-112.0,43.5,49.5
*SELECT 6,20
(Fl=4+Fl=5+Fl=6+Fl=22+Fl=23+Fl=24+Fl=25+Fl=39+Fl=40+Fl=42
                                                    :1)
*REWIND 20
*SELECT 6,20
(Fl=90:+1,S3,.05)
*END
-------------------SEARCH ADDRESS 999-------------------
```

```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1--------------------
--------------------SEARCH ADDRESS   2--------------------
*LOG=2
*ORIGIN=(.5,.5)
--------------------SEARCH ADDRESS   3--------------------
*REGION=(-116.5,45,2500000,,1,-110,2500000)
*LIMITS=-116.5,-112.0,43.5,49.5
*SELECT 6,20
(F1=4+F1=5+F1=22+F1=23+F1=24+F1=25+F1=39+F1=40+F1=41+
                                        F1=42:1)
*GRID 1
(36,37,-41,-19)
(33,35,-42,-17)
(25,32,-34,-13)
(17,24,-33,-12)
(13,15,-27,-10)
(12,12,-27,-8)
(11,11,-24,-7)
(10,10,-21,-6)
(3,6,-24,-11)
(2,1,-23,-14)
(-1,-1,-25,-19)
(-3,-2,-25,-20)
*END
--------------------SEARCH ADDRESS 999--------------------
```

MAP II Discussion

The *LOG, *ORIGIN, *REGION, and *LIMITS specifications are similar to those presented for Map I (see the appropriate section in the manual). Note that scale of 2500000 (1:2500000) produced the map seen at the beginning of this section.

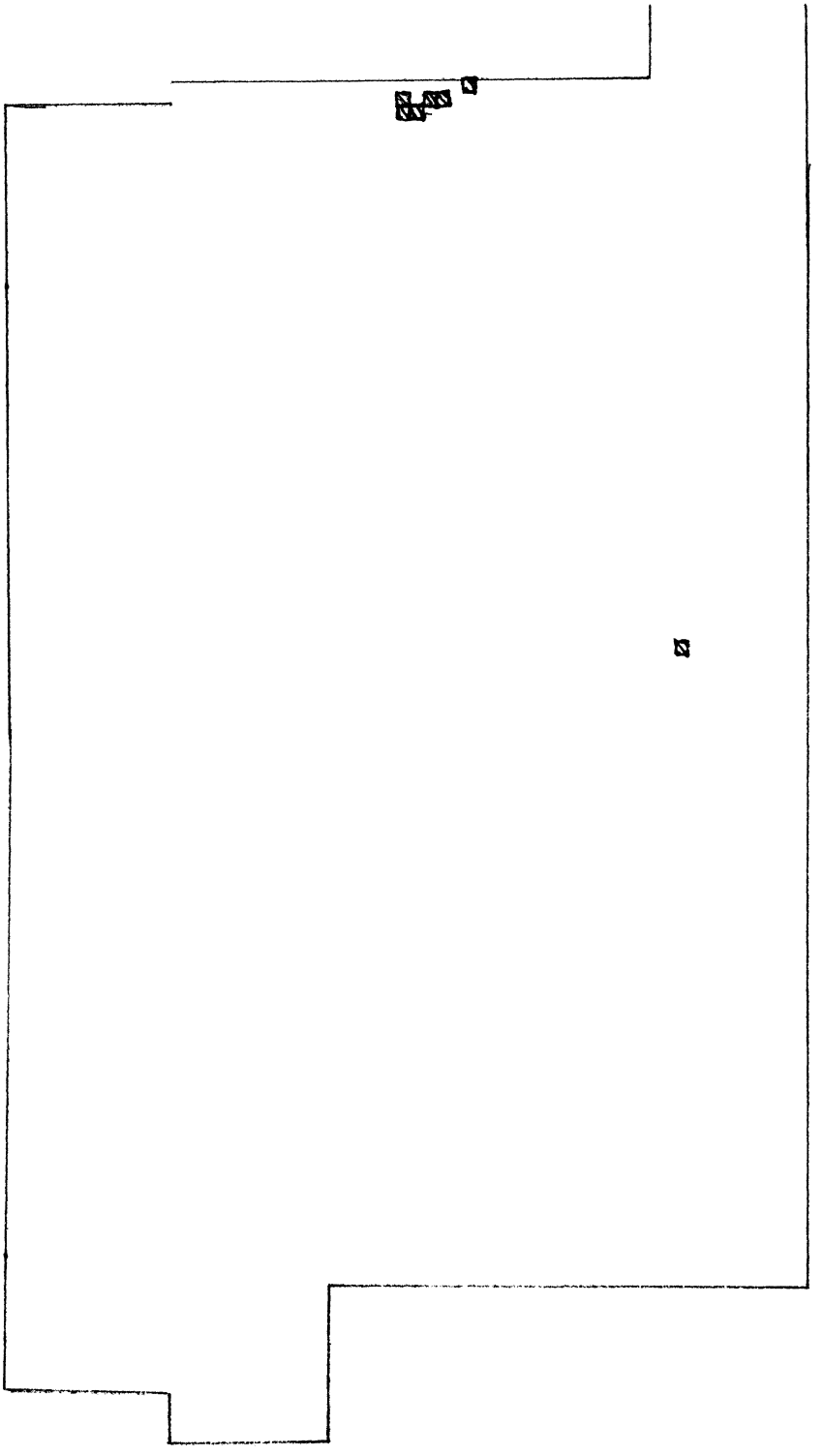
I type *SELECT 6,20 once again indicating that I want to use the FEATURE file. Note the set up for the selection parameters (F1=4+F1=5...). Since the FEATURE file is not random access, it makes one pass through the file. Thus on that one pass I have the program looking in field 1 (F1) for the numbers indicated (see Figure 2). All of the selection parameters must be set up in one line as opposed to the way *GRID is set up in the second log listing. When I first ran this program I had each of the F1=? on separate lines, each in its own set of parentheses. This arrangement resulted in error messages which took me about half an hour to unravel. The + means "and", so the program is looking for F1=4 and 5 and 6 etc. When the program finds the desired counties in the FEATURE.DAT file, the 1 behind the colon indicates that the county boundaries are to be drawn.

*REWIND 20 resets the FEATURE file so I can use it again. To use *REWIND for each of the selections wanted is inefficient in terms of the computer time used.

On the second run through the FEATURE file (F1=90:+1,S3,.05), I look in F1 for 90, which is the Flathead National Forest. In addition to having the outline plotted (+1), I want it shaded. S3 indicates that the cross-hatches are to be at 45 degree angles and .05 inches apart (see *SHADE in the manual for the S? options).

On the second log listing there is no shading or forest; this map (not shown) would have the township-range grid drawn on it as well as the county boundaries. The purpose of including this listing was to show the difference between the *GRID and *SELECT parameter set up.

MAP III



```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1--------------------
--------------------SEARCH ADDRESS   2--------------------
*LOG=2
*ORIGIN=(.5,.5)
--------------------SEARCH ADDRESS   3--------------------
*REGION=(-106.1532,48.5624,400000,,1,-110,500000
*LIMITS=-106.16,-104.98,48.56,49.0
*SELECT 6,20
(F4='DANI':1)
*SELECT 4,23
(F8=55:2,S3,.03)
*END
--------------------SEARCH ADDRESS 999--------------------
```

```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1--------------------
--------------------SEARCH ADDRESS   2--------------------
*LOG=2
*ORIGIN=(.5,.5)
--------------------SEARCH ADDRESS   3--------------------
*REGION=(-106.1532,48.5624,500000,,1,-110,500000
*LIMITS=-106.16,-104.98,48.56,49.0
*SELECT 6,20
(F4='DANI':1)
*SELECT 4,23
(F8=55:2,205,.03)
*END
--------------------SEARCH ADDRESS 999--------------------
```

MAP III Discussion

This discussion, once again, centers on selection and test parameters for the *SELECT function.

When going through the FEATURE file (20) I am looking in field 4 (F4) for the occurrence of 'DANI' (see Figure 2), which signifies Daniels county, when 'DANI' is found I want to plot it. This gives me the outline for the county as seen on the third map.

The second *SELECT directs the program to a data file of my own creation. The file is FOR23.DAT and is formatted in Type 4 (see *SELECT Type 4 in the manual). FOR23.DAT contains all of the archaeological sites in Daniels county and looks like the following (all entries begin in column 6):

```
     133N47E012550051XXXXA
     235N51E016550056XXXXC
     335N51E017550056XXXXD
     435N51E020550055XXXXA
     535N51E021550054XXXXC
     635N51E028550052XXXXB
     735N51E033550053XXXXA
```

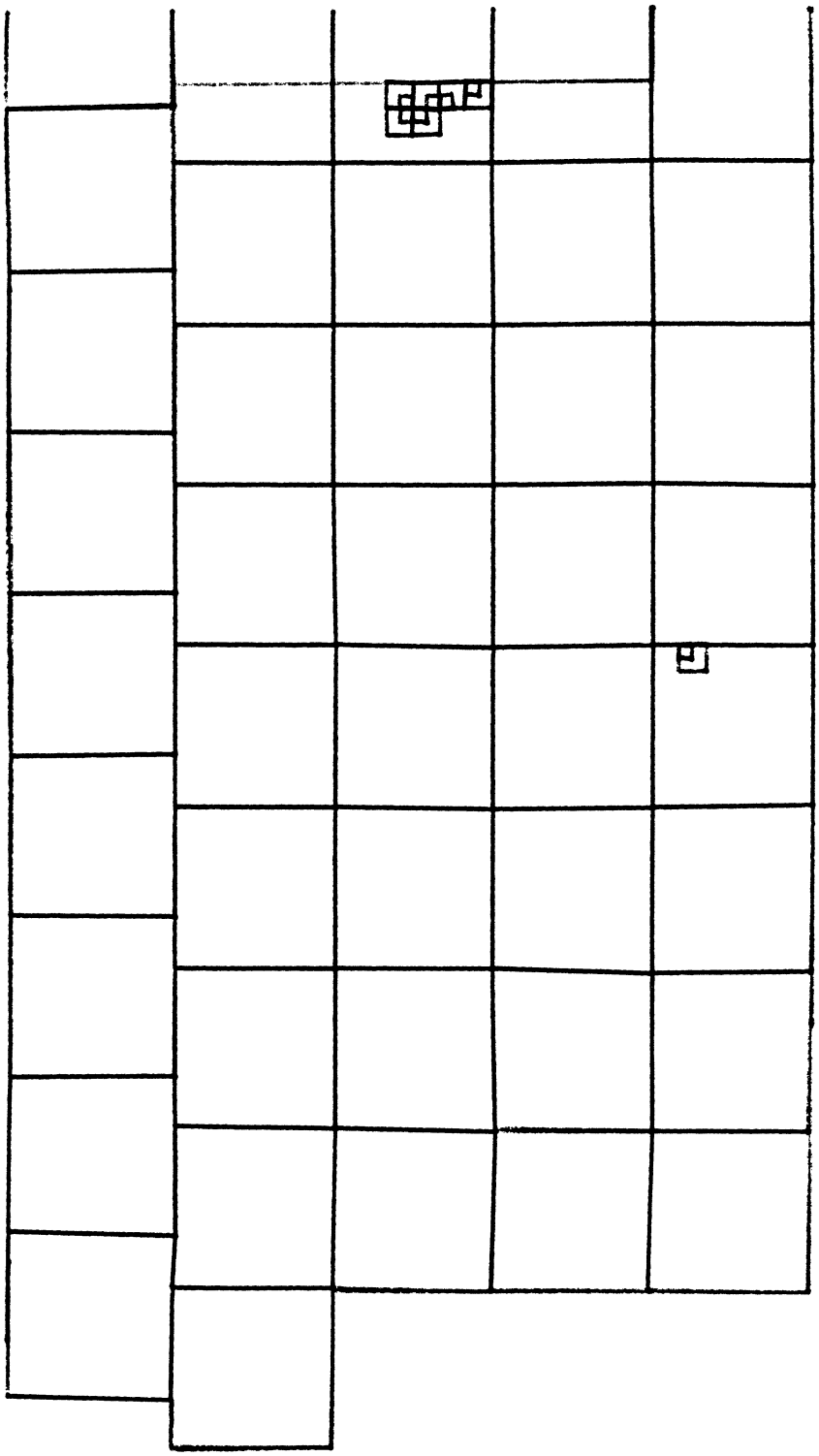
Thus the coding for the first site entry corresponds to:

```
     F1   seq=1
     F2   twn=33
     F3   n/s=N
     F4   rng=47
     F5   e/w=E
     F6   F=0  -  not a fractional township
     F7   se=12
     F8   co=55  -  see Figure 2
     F9   Id=0051  -  I am using site numbers
     F10  text=XXXX  -  filler
     F11  tract=A  -  northeast, see *SELECT Type 4
```

The selection parameter (F8=55...) indicates that as GRAFIX goes through FOR23.DAT, it is to look in field 8 (F8) for the number 55, the numeric code for Daniels county. When selection is true, then the 2 means that only the parcel is to be processed; the parcel in this case is the tract (see *SELECT Type 4 in the manual). The processing consists of shading the parcel (tract) with cross-hatches at 90 degrees and .03 inches apart.

On the second listing (F8=55:2,205,.03) indicates that when a 55 is found in F8, then the tract is to have a circle (symbol number=205, see Fig. A-1 of the GRAFIX manual) drawn in it. The circles are to be .03 inches high.

MAP IV



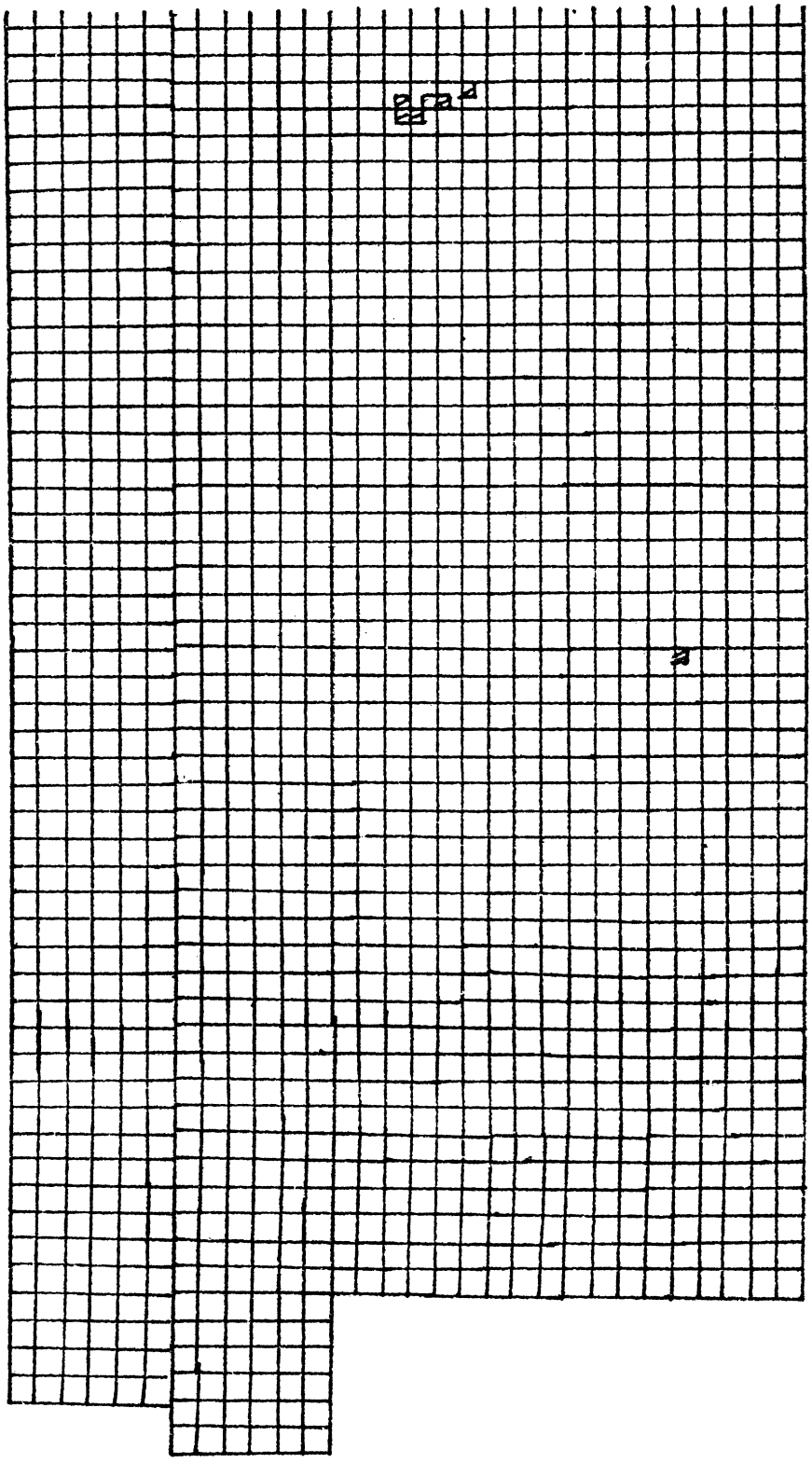
```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1--------------------
--------------------SEARCH ADDRESS   2--------------------
*LOG=2
*ORIGIN=(.5,.5)
--------------------SEARCH ADDRESS   3--------------------
*REGION=(-106.1532,48.5624,4500000,,1,-110,4500000
*LIMITS=-106.16,-104.98,48.56,49.0
*SELECT 6,20
(F4='DANI':1)
*SELECT 4,23
(F8=55:3)
*GRID 1
(36,37,48,51,90,90)
(33,35,44,51,90,90)
*END
--------------------SEARCH ADDRESS 999--------------------
```

MAP IV Discussion.

This map differs from MAP III in two ways. First, note the action parameters in the second *SELECT statement (F8=55:3). When 55 is found in F8 of FOR23.DAT, then the program processes the tract and the section. As you can see from MAP IV, this resulted in the section being drawn with the appropriate tract drawn within. I did not specify any shading. If I had then both the section and the tract would have been shaded.

I also had a township-range grid drawn within the county boundaries (see *GRID in the manual for detailed discussion). Note that there are problems with labelling the townships and ranges.

MAP V



```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1-------------------
--------------------SEARCH ADDRESS   2-------------------
*LOG=2
*ORIGIN=(.5,.5)
--------------------SEARCH ADDRESS   3-------------------
*REGION=(-106.1532,48.5624,400000,,1,-110,400000
*LIMITS=-106.16,-104.98,48.56,49.0
*SELECT 6,20
(F4='DANI':1)
*SELECT 4,23
(F8=55:2,S6,.03)
*GRID 3
(36,37,43,51,89,89)
(33,35,44,51,89,89)
*END
--------------------SEARCH ADDRESS 999-------------------
```

MAP V Discussion.

Another Daniels county map. In the second *SELECT, shading cross-hatches are at 112.5 degrees (S6) and are .03 inches apart. The *GRID 3 indicates that a township-range-section grid is to be drawn.

MAP VI

```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS   1--------------------
--------------------SEARCH ADDRESS   2--------------------
*LOG=2
*ORIGIN=(.5,.5)
--------------------SEARCH ADDRESS   3--------------------
*REGION=(-116.5,43.5,4000000,,1,-110,4000000)
*LIMITS=-116.5,-103.5,43.5,49.5
*SELECT 6,20
(F1=104:1)
*SELECT 4,24
(F1=1,8:1,S5,.04)
*END
--------------------SEARCH ADDRESS 999--------------------
```

MAP VI Discussion.

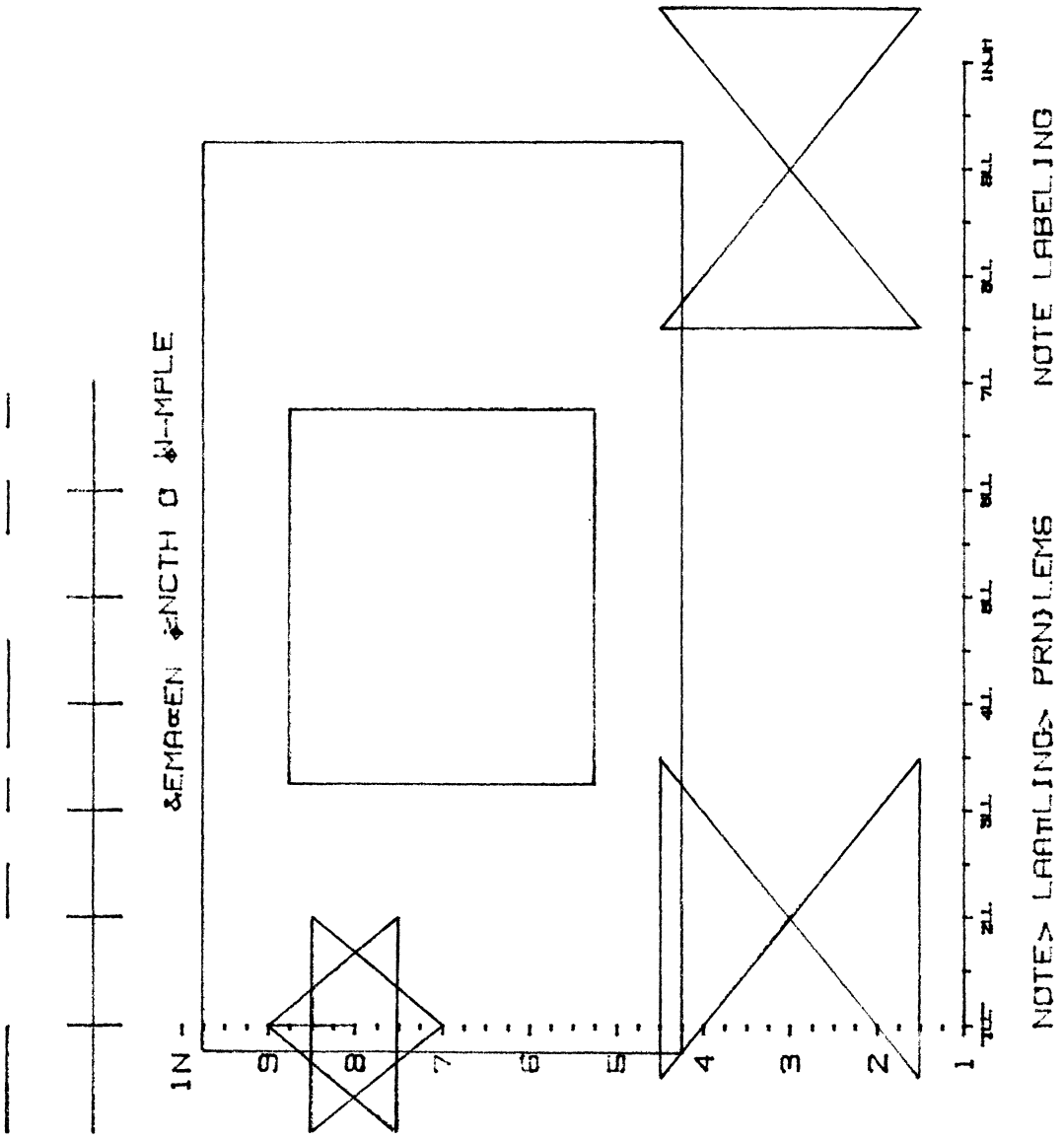
In the first *SELECT, the program goes through the FEATURE file looking in F1 for 104 (see Figure 2), when found the state outline is plotted.

The information for plotting the sites is in FOR24.DAT and is a in Type 4 format (see *SELECT Type 4). Following is FOR24.DAT (all entries begin in column 6):

```
105S28E020150797XXXXC
205S28E020150797XXXXA
337N21E011460331XXXXD
433N13E013540032XXXXC
509S05W018020075XXXXC
626N32E024471118XXXXD
735N32E019471922XXXXD
820N05W039430067XXXXB
```

The selection test (F1=1,8:..) indicates that if field 1 (F1) contains an integer 1 through 8, inclusive, then process the entry. For this map I have switched to processing sections because tracts would not be visible on a state map this size. To use tracts and have them appear clearly would require a map about three times the size of the one shown here.

SAMPLE VII



```
ISD/RSSB COMPUTER GRAPHICS SYSTEM / LOG LISTING

--------------------SEARCH ADDRESS  1--------------------
--------------------SEARCH ADDRESS  2--------------------
*LOG=2
*REGION=(0,0)
*AXIJ
 1,1,(1,10,1,9,2),1
 2,1,(1,10,1,9,4),-1,.2
*FORMLINE=(1,1,.5,.5,.3,.3)
*LINE
 7,(0,12)
*FORMLINE=(1)
*LINE
 7,(0,11)
*FORMLINE
*CENTER=(5,7)
*SYMBOL
 14,2,(1,8)
*BORDER
 0,.75
 0,.5
*ROTATE 90
*SYMBOL
 12,3,(3,-9)
*ROTATE 0
*SYMBOL
 12,3,(2,3)
*TEXT
 .2,(1,0),'NOTE  LABELING  PROBLEMS'
 .2,(7,0),'NOTE'
 .2,(8,0),'LABE'
 .2,(8.8,0),'LING'
*EMBLEM
 38,.2,(3,10),.2,' EMBLEM FUNCTION EXAMPLE'
*END
--------------------SEARCH ADDRESS 999--------------------
```

SAMPLE VII Discussion

*REGION=(0,0) establishes an origin point for the future use of *ROTATE.

*AXIJ 1,1,(1,10,1,9,2),1 defines an x-axis to be drawn one inch from the origin. The minimum value is one (1) and the maximum is ten (10). Nine (9) intervals will be drawn with each interval equal to one (1). There will be two subintervals per interval (one minor tic per interval). The numeric labels are to be in decimal format (note that decimals do not come out as expected).

*AXIJ 2,1,(1,10,1,9,4),-1,.2 defines an y-axis to be drawn one inch from the origin, the values go from 1 to 10, there are 9 intervals each with a value of one. Each interval will have 4 subintervals. The numeric labels are integers .2 inches high.

*FORMLINE=(1,1,.5,.5,.3,.3) defines a dashed line such that the first dash and blank are one inch long, the second are half an inch, and the third are one-third inch long.

*LINE 7,(0,12) will draw a seven inch line with the above line specifications. The sequence will repeat itself until the seven inches are completed. The line origin is at 0,12.

*FORMLINE=(1) defines a solid line with cross-hatches (the railroad tracks mentioned under *FORMLINE in the manual).

*LINE 7,(0,11) will draw a line 7 inches long in the above *FORMLINE format.

*FORMLINE sets the line style back to solid. Without this function all future lines would conform to the above cross-hatch line format.

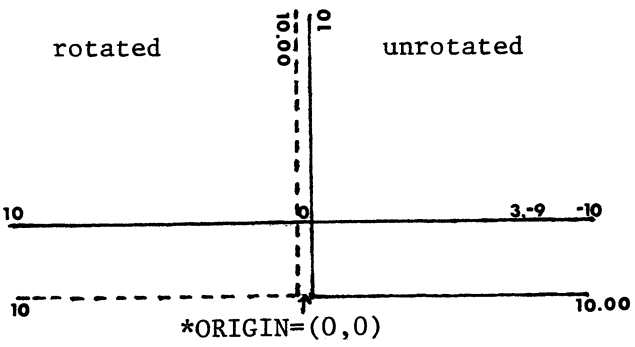
*CENTER=(5,7) causes all future features, lines, symbols, etc. to be centered at 5,7.

*SYMBOL 14,2,(1,8) cause a two inch star (14) to be drawn at coordinates 1,8. The coordinates established in this function and any following functions override those set in *CENTER.

*BORDER 0,.75
*BORDER 0,.5 causes a double border to be drawn. Since I did not specify cooridinates,It reverts back to the *CENTER; thus a borderIs drawn around the point at 5,7.

*ROTATE 90 will rotate all following lines, symbols, etc. 90 degrees to the right.

*SYMBOL 12,3,(3,-9) draws a three inch hourglass (12) figure rotated ninety degrees at 3,-9. See drawing below for coordinate explanation.



*ROTATE 0 returns the plot to its original position.

*SYMBOL 12,3,(2,3) draws a three inch hourglass (12) type figure at 2,3. This symbol was drawn to show that the rotate function does work.

*TEXT .2,(1,0),'NOTE LABELING PROBLEMS' print an alphanumeric string .2 inches high with an origin at 1,0. Notice that what the user inputs is not what the user gets. I think the problem is caused by the program having been written for four bit words and our system uses five bit words. The end result is that most textual (alphanumeric) data does not come out as expected.

```
.2,(7,0),'NOTE'
.2,(8,0),'LABE'
.2,(8.8,0),'LING'
```

The three above lines illustrate a way of handling string data for text, emblem, etc. functions. This method will take considerable work for long strings.

*EMBLEM 38,.2,(3,10),.2,'EMBLEM FUNCTION EXAMPLE' will produce a symbol (38) .2 inches high followed by the text, also .2 inches high. The text output has the same problems

discussed in *TEXT above.

THE

GRAFIX

MANUAL

1. Introduction

GRAFIX is a complex of computer programs designed to facilitate production of maps, drawings, or charts on a pen plotter. GRAFIX is written in FORTRAN IV for use on the Decsystem-20 and the Calcomp 563 30" drum plotter. This document provides detailed instructions on the use of GRAFIX.

I wish to thank Randy Holm from the Montana State Department of Administration, Computer Services for permission to change the wording in the manual and alter the program to make them compatible with the Dec20 and Calcomp plotter.
Dori Passmann

2. Spaces

There are two spaces or regions relevant to users of the GRAFIX system: physical space and logical space.

2.1 Physical Space

Physical space is determined by the particular plotter and paper being used. All dimensions are expressed in inches. The drum plotter (Calcomp 563) currently in use accepts rolls of plotting paper 30 inches in width and up to 6 feet in length. The GRAFIX system allows a plot to be rotated on the plotting paper; thus a user need only insure that the smallest dimension of his/her plot (either width or height) is less than or equal to 28 inches. The largest dimension is restricted only by the six foot limit or by the length of paper remaining on the plotter when the plot is generated.

2.2 Logical Space

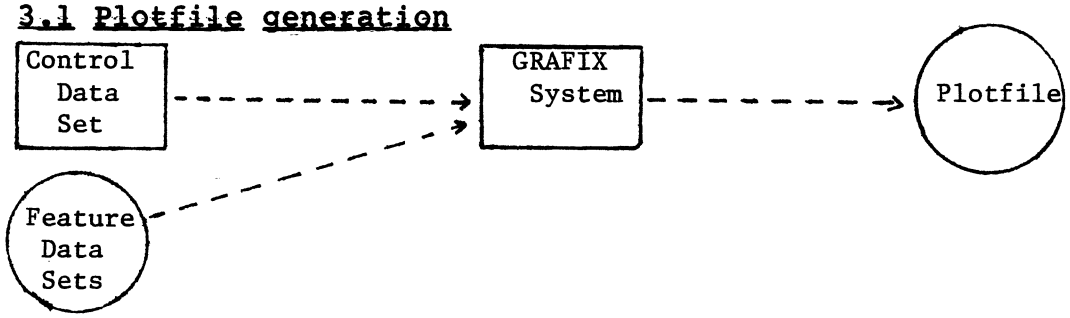
Logical space is the mathematical or theoretical space being portrayed by the plot. The units of dimensions are whatever the user desires (e.g. feet, miles, degrees of latitude/longitude, etc.). The range of X and Y values is unbounded (as long as an appropriate scale is chosen so that the plot fits in the physical space available). A regular mathematical coordinate system is used where X values increase to the right and Y values increase to the top of the plot. Appropriate commands are available to allow the user to relate logical space to physical space.

3. The Plotting Process

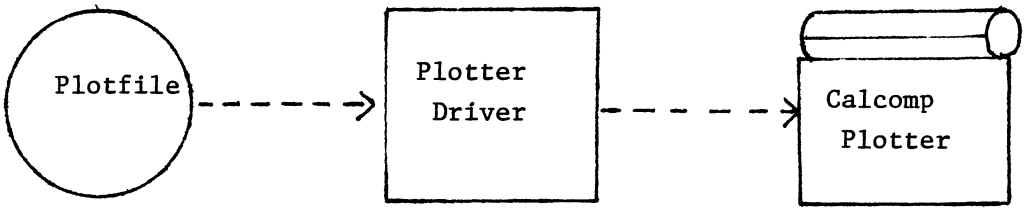
The plotting process consists of two phases:

(1) generation of the plotfile
(2) driving plotter with plotfile

3.1 Plotfile generation



3.2 Plotting



4. Data Sets

A data set is a set of related records existing as punched cards, or on a magnetic tape, or on a magnetic disk.

4.1 Designation

Data sets are referenced in GRAFIX by a number - an integer between 22 and 24 (the standard Fortran unit designations; 1 is reserved for the control file, 20 is designated the FEATURE file, and 21 is for the TOWNSHIP file).

4.2 Input data sets

Data sets input to GRAFIX are used to control processing and to supply required data.

4.21 Conrol data set

The control data set consists of the instructions created by the user to control processing.

4.211 Function records

Function records initiate one of the options or functions available in GRAFIX. An asterisk ("*") must be present in location (column or field) 1 of a function record. Locations 2 through 4 define the particular function being initiated while the rest of the record is available for commentary or required parameters. The following table provides a summary of the functions currently available in GRAFIX. Note that although the function name is spelled out (e.g. *AREA) only the first 3 letters are required (*ARE). These functions are ordered generally by increasing complexity of use.

*bbb - commentary record, ignored by GRAFIX (b denotes a blank)
*END - denotes end of control data set; specifies location of pen at termination of plot
*INPUT/UNIT - specifies the data set reference number for remainder of control data set
*RECORD/LENGTH - specifies length of records in remainder of control data set
*LOG - specifies the extent of output listings desired
*NEWPEN - specifies one of 4 pens (1 through 4)
*ORIGIN - establishes the physical origin on plotter page
*REGION - specifies the relationship of logical space to physical space
*ROTATE - causes further plotting to be rotated on plotter page
*CENTER - forces following plotted feature(s) to be centered about some point
*CHECK - insures that plot will fit in physical space available
*ERR/SET - overrides specifications in FORTRAN error handling routines
*SYMBOL - draws a symbol
*TEXT - "draws" an alphanumeric textual string
*EMBLEM - "draws" a text centered within a symbol
*BORDER - draws a border about some area
*AXIJ - draws an axis together with tic marks and labels
*TIC/LENGTH - specifies the length of tic marks drawn under *AXIS function
*ANNOTATE - provides versatile labelling capabilities for axes, grids, etc.
*GRID - draws and labels rectangular grid or township grid
*OUTLINE - provides a rectangular outline to a projected latitude-longitude grid
*PLOT - reads records containing X-Y series of points from a feature data set and draws line connecting points
*REWIND - repositions a data set at its beginning
*INSERT - provides the capability to insert labels within a line
*MOVE - provides the capability to selectively move labels created by the *INSERT function
*AREA - causes areas to be computed for all following features
*FORMLINE - specifies line format (solid, dashed, etc.)
*SHADE - provides shading (cross-hatching) in features
*ASSIGN - saves location data for subsequent use
*MEASURE - provides capability to obtain extent of X-Y values used in plotting created by a series of function instructions

*SAVE/UNIT - saves a series of X-Y values for subsequent use
*DATA/FORMAT - specifies the format of X-Y data to be read under the *PLOT function
*LIMITS - specifies the minimum and maximum values to be accepted for any feature
*FACTOR - enlarges or reduces an entire plot
*CALL EXIT - provides an exit from GRAFIX to user coded routine
*SELECT - reads point or property description data, plots and labels it
*LABEL - draws a label (numeric)
*TRANSLATE TO - specifies a temporary translation of a physical origin

4.212 Selection records

Many function records require selection records to follow. The selection records specify selection criteria to be applied to the function in question. Consider the following sequence:

```
*BORDER        (a)
1,(1,1,10,10)  (b)
```

Line (a) provides the function record *BORDER which directs GRAFIX to draw a border about an area. Line (b) provides a selection record which specifies the minimum X-Y value: (1,1) and the maximum X-Y value (10,10) of the border to be drawn.

4.213 Documentation notation

Section 5 provides detailed information concerning function and selection records. The notation used will conform to the following rules:

(1) Asterisks (*), apostrophes ('), upper-case letters and paraentheses must appear in control records as indicated.
(2) Parameter names are written in lowercase letters.
(3) Groups of parameters that are delimited by brackets ({,}) are optional. The brackets do not appear in the control record.
(4) Within a group of parameters, as delimited by parentheses and/or brackets, required parameters are underlined. Optional parameters are not underlined.
(5) A slash (/) separates parameters or groups of

parameters to denote that either one parameter (group) or the other parameter (group) is present

For example:

*BORDER
ntrval, {xsw,ysw,xne,yne,} delta

indicates that the parameter ntrval must appear in the control record while the series xsw, ysw, xne, yne is optional. However, if the series does appear then all 4 parameters in the series must be present because they are underlined. The delta parameter is an optional entry.

4.214 Free-form fields

All parameters specified in the control data set occur in free-form fields. That is, the user need not be concerned with the starting location or the length of a field. Further GRAFIX ignores any characters except the 10 digits (0-9), minus signs (-), and decimal points when interpreting parameter entries.
Thus:

```
 1,(1,1,10,10)        and
ntrval=1,(xsw=1,ysw=1,xne=10,yne=10)
```

would be interpreted identically by GRAFIX.

4.215 Symbolic parameters

Generally a user of GRAFIX would supply numbers for the parameters required by the various control records. However, in some cases it may be desirable to enter symbolic parameters, that is, variables. Symbolic parameters are denoted by the "at" sign (@). The following table defines the symbolic parameters currently available:

parameter	value
@XL, @YL	minimum X,Y since start (or occurrence of last *MEASURE 1)
@XH, @YH	maximum X,Y since start (or occurrence of last *MEASURE 1)
@XM, @YM	mid-range X,Y since start (or occurrence of last *MEASURE 1)
@X-, @Y-	minimum X,Y in the last feature
@X+, @Y+	maximum X,Y in the last feature
@XC, @YC	mid-range X,Y in the last feature
@X.f, @Y.f	.f part of X,Y range in the last feature
@XN, @YN	last X,Y point in the last feature

@Xi, @Yi	ith point in last feature (i is an integer number)
@α	denotes a bearing distance measurement. (α is a number, real or integer, specifying an angle in degrees) Occurs as Y element in X-Y pair. For example, .. 123.7, @23.5,.. indicates a point 123.7 units distant from last point and at an angle of 23.5 degrees.

4.216 Label format (ndec) specifications

The format of the various labels produced by GRAFIX is controlled by the parameter, ndec. In the following discussion let x be the 10's digit and y be the units digit.

(1) Real number with decimal point: ndec=10*x+y>=0. The number will be plotted with y digits to the right of the decimal point. If x#0 only x least significant digits to the left (including leading zeros) of the decimal point will be plotted. If x=0 all digits to the left (excluding leading zeros) of the decimal point will be plotted.
(2) Integer number (no decimal point): ndec=-(10*x+1). If x#0, x least significant digits (including leading zeros) will be plotted. If x=0 all digits (excluding leading zeros) will be plotted.
(3) Character data: ndec=-(10*x+2). If x#0, character string to be plotted starts with the xth word in the string (one word = five characters). If x=0 the full string will be plotted.
(4) Degrees-minutes-seconds: ndec=-(10*x+3). The format of label is defined by the following table:

	Plot:		
x	deg DDD	min MM'	sec SS"
0	x		
1			x
2		x	
3		x	x
4	x		
5	x		x
6	x	x	
7	x	x	x

4.217 Location/orientation parameters

Certain of the function-selection records appear in the form:

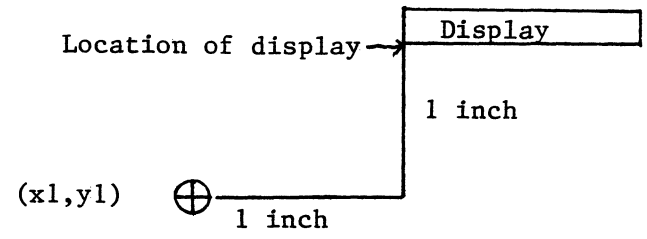
```
*......rform,.......
.....{pos,disp,} {xdel,ydel,} {(x1,y1,x2/angle,y2)}
```

These parameters all have to do with the location and orientation of a straight line, a symbol, or textual data (display). The rform parameter indicates the presence of pos,disp and xdel,ydel according to the following table:

rform	pos,disp	xdel,ydel
0		
1		x
2	x	
3	x	x

(x1,y1,x2/angle,y2) specifies a starting location and/or angle for a display. The following: (x1,y1), (x1,y1,x2,y2), (x1,y1,angle), or (angle) are all acceptable forms for this series of parameters. (x1,y1) = the starting location. If (x2,y2) is specified, the angle is determined as the angle of the vector from (x1,y1) to (x2,y2). If neither (x2,y2) nor the angle is specified the angle is taken as 0. If (x1,y1) is not specified, then (x1,y1) is taken as the last (x1,y1) to have been entered. The pos,disp and xdel,ydel parameters are used to specify the location and orientation of a display relative to the starting point (x1,y1).

The xdel,ydel are offsets, in the x and y directions, of the location of the display from (x1,y1). The xdel and ydel are specified in the units of logical space as defined by the *REGION function. The location of a display is the lower left-hand corner of that display. For example, if the units of xdel,ydel are miles and scaling is 1:63360, then a specification of xdel=ydel=1 would result in the following displacement on the plot:



The displacement is 1 inch in X and Y since 1 mile = 63360 inches but the scaling is 1:63360.

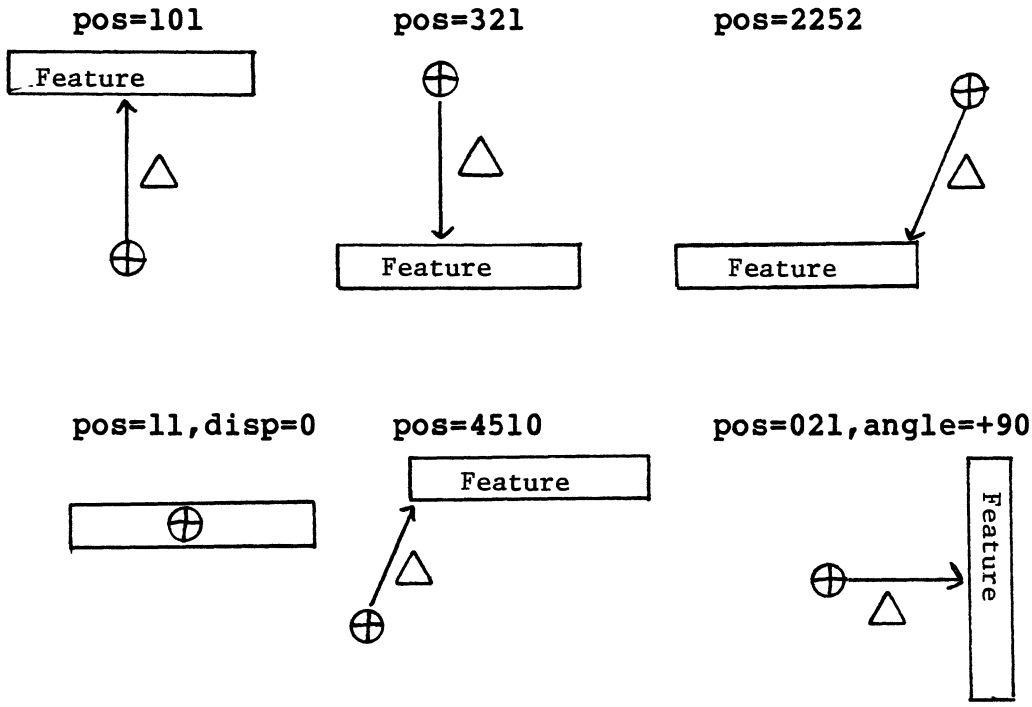
The pos,disp parameters define a further translation of the location of a display. The disp is a radial displacement expressed in inches. Let pos = 100*A + 10*V + H, where A, V, and H are defined as follows:

A = angle in degrees for A>=4. For convenience A<=3 is defined: 0=>0 degrees, 1=>90 degrees, 2=>180 degrees, and 3=>270 degrees. A, together with disp, defines a polar coordinate displacement of the location.

V = positioning along a line perpendicular to angle, i.e. @ angle + 90 degrees: consider this a vertical adjustment; then V=(0=> display "above" location, 1=> display centered vertically, and 2=> display "below" location).

H = positioning along line at "angle" degrees: consider this a horizontal adjustment; then H=(0=> display to right of location, 1=> display centered horizontally, and 2=> display to left of location).

In the following examples let Δ=disp and let ⊕ denote the starting location after xdel,ydel (if any) translation.



4.218 Selection testing

Certain of the function records require selection test records of the following form:

(f1 ⊕ l1,h1 □ f2 ⊕ l2,h2 □ ... □ fn ⊕ ln,hn {:---}),...
where the fi are field numbers, the
li are low values acceptable, the
hi are high values acceptable, the
⊕ are relationship indicators (= for equals and # for not equals), and the
□ are connectors (& for "and" and + for "or")

Each set of fi⊕li,hi is called a test and a series of tests delimited by paraentheses is called a group. A test may also be indicated in the form: fi⊕vi, where the vi is a single value. The tests are logically associated such that "and"ing takes place before "or"ing. The Entry in brackets {:---} denotes the following:

(a) If a colon is present but no numeric data follows the colon indicates that this is a terminating group. If the testing results in a true condition no further records are to be read from the data set being accessed.
(b) If no colon is present the group is called a key group. Results of testing in this group are to be "and"ed with tests in all following groups. (Normally groups are independent)
(c) A colon together with numeric entries indicates a regular group. The numeric entries specify some action that is to be taken if the testing results in a true condition.

Generally the fi are field numbers, that is, indices to the fields that are to be tested. More specifically:

fi	meaning
>0	fi is the index of field to be tested. The test results in a true condition if the test condition is satisified
=0	testing is performed with the number of records accessed as the argument (in place of a field)
<0	two possibilities: (1) If |fi|<=100: the number of records thus far accepted by group |fi| is the argument (2) If |fi|>100: |fi|-100 is a group

number and vi=0 or 1 to denote
that testing was unsuccessful or
successful in group number |fi|-100.

Examples

(f1=10,100:action 1)
specifies that action 1 is to be performed if field 1 is: 10 <= field 1 <= 100.

f3#0), (group 2), -----
specifies that testing in groups 2, 3, ---- can result in a true condition only if field 3 is not zero.

(n=100:) or (f0=100:stop)
each specify that reading data set is to be discontinued when 100 records have been read.

(---group 1---),(---group 2---),---,(-2=1,10;action n)
specifies that action n is to be performed if the number of records accepted in group 2 testing is: 1 <= nrecs accepted <= 10.

(---group 1---),(-101=0:stop)
spefifies that reading the data set is to be discontinued as soon as testing in group 1 is unsuccessful.

4.22 Feature data sets

In most applications data generated by another program or digitized in some fashion must be introduced to GRAFIX for processing. These auxiliary inputs are called feature data sets and are accessed via various GRAFIX functions. Feature data sets may be divided into two types: line segment and point or parcel.

4.221 Line segment, X-Y series

Data sets consisting of polygonal boundaries for geographic areas or sets consisting of X-Y data for charts would fall in this classification. These sets are accessed by the *SELECT and *PLOT functions.

4.222 Point, property description

Data sets consisting of point data (eg: cities, well locations, individual X-Y pairs) or legal descriptions for property fall in this classification and are accessed by the

*SELECT function.

4.3 Output data sets

There are three types of outputs from GRAFIX: a log listing, a plotfile, and subfiles.

4.31 Log listing

The log listing is generally a printed output listing of the run and diagnostic information. The extent of output received is controlled by the *LOG function.

4.32 Plotfile

The plotfile consists of the instructions required to drive the plotter. It is generally created on a magnetic tape so that it can be used in an ensuing plot job.

4.33 Subfiles

Under control of some functions GRAFIX must write a data set which will be used later in the same run. These data sets would probably be temporary sets created on a magnetic tape. The functions *LIMITS and *SAVE/UNIT may require a subfile to be written. The subfiles written are in a *SELECT Type 5 format.

5.0 Symbols available

Symbols are special characters that may be created by GRAFIX. Two types of symbols are available: regular symbols and special symbols.

5.1 Regular symbols

The following figure (Fig. A-1) designates the regular symbols available in GRAFIX. These symbols are specified by a symbol number of 0 through 255.

5.2 Special symbols

The special symbols defined by the following table are available in GRAFIX:

symbol number	symbol created
205	circle
1xxx	polygon with xxx sides. Note that the symbol number is an integer greater than 1000.

6.0 References

The following pages discuss in detail the functions available in GRAFIX.

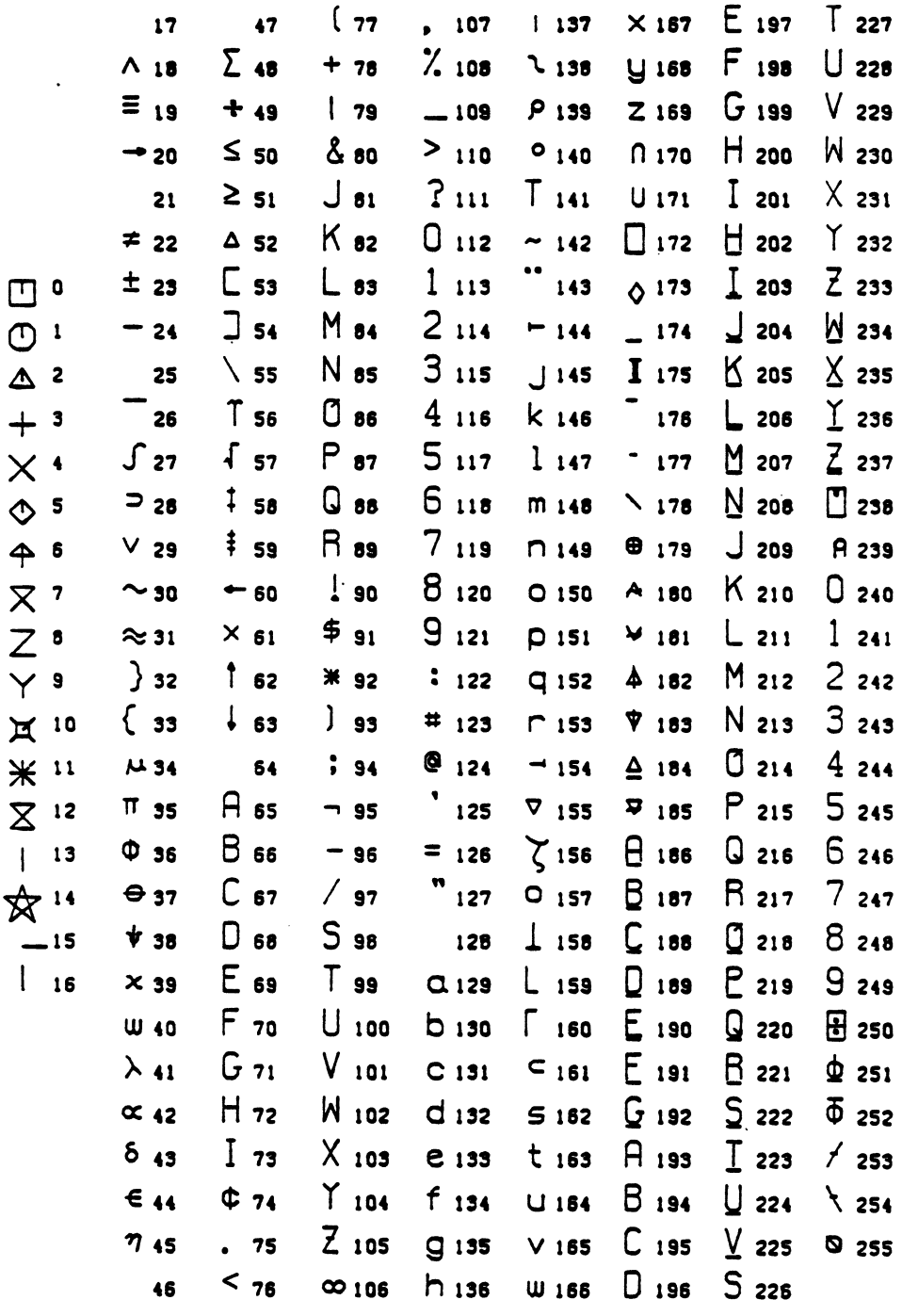


FIGURE A-1 EBCDIC HOST CHARACTER ORDER

*bbb <---------- optional ---------->

Function records that are blank in locations (columns, fields) 2-4 are ignored by GRAFIX. These records may then be used to introduce any commentary desired to the control data set. Any entry desired may be made starting with location 5.

*AREA iconv, ndeca, unita

Indicates to GRAFIX that areas are to be computed and listed for all features until encountering an *AREA entry with iconv=0.

iconv = units in which an area is to be expressed
- 0 - no area computations to be performed (default)
- 1 - acres
- 2 - square miles
- 3 - squre feet
- 4 - square inches
- 5 - hectares
- 6 - square kilometers
- 7 - square meters

ndeca = number of decimal places desired in the listing (default is 0)

unita: data set reference number for the data set in which areas are to be listed.

*ANNOTATE
idir,offset,ndec,height,{(min,max,delta,nlabel,pos,disp,
angle,modlbl,},{vall,vald,nchar,'text'}

Labels an axis line.

idir = index to axis (1=>x-axis, 2=>y-axis). Note that idir can also be -1 or -2. (See min, max, delta).
offset = displacement of the labels from origin (x-axis labels offset in +y direction, y-axis offset in +x direction)
ndec = number of decimal places desired in labels
height = height (inches) of labels - if specified as 0, height will be set to .105
min = the minimum value to be labeled
max = the maximum value to be labeled
delta = value between labels
If min, max, delta not specIfied, values input with an *AXIJ option will be used. If idir<0 the min, max,
and delta specified here will denote the number of intervals from the origin of the axis to min and max and the number of intervals in delta. If idir<0 and min, max, and delta are unspecified then min=1, max=ntrval specified with *AXIJ, and delta=1 interval.
nlabel = the number of labels to be printed
pos,disp = see Location parameters 4.217
pos: (default is 321 for x-axis and 212 for y-axis)
disp: (default is lticl+height/2 - see *TIC/LENGTH)
angle = angle (degrees from x-axis) at which label is to be plotted (default is 0).
modlbl = index of intervals to be labeled/not labeled.
If modlbl=0 (default) all intervals will be labeled.
If modlbl<0 and modlbl=-n then intervals 0,n,2n,... will not be labeled. If modlbl>0 and modlbl=n then only intervals 0,n,2n,... will be labeled.
vall = the value of first label (default is min).
vald = the incremental value between labels (default is delta)
Note: vall, vald may be specified only if ndec not equal to -3.
nchar = the number of characters in each label, labels obtained from character string in 'text'
text = a character string of label text
Note that nchar, text must be specified if and only if ndec = -3.

Examples of *ANNOTATE

*ANNOTATE

```
1,,-1,.105,(1,10,1)                                    (a)
1,,0,.105,(1,10,1,,321,.1,90,2),10,10                  (b)
1,,-2,.105,(1,4,1),3,'sunmontuewed'                    (c)
```

(a): 1 2 3 4 5 6 7 8 9 10

(b):

10.
30.
50.
70.
90.

(c) sun mon tue wed

*ASSIGN vl,ll,v2,----,vn,ln

Assigns values to specific locations in the X or Y arrays.

vi = a value to be assigned to location li
li = a location to receive value vi

Example: *ASSIGN 100,X101,@X-,Y1000
Places the value 100 at the 101st location in the X array and the value of @X- at the 1000th location in the Y array. See the discussion on symbolic parameters in section 4.215. Note that the X and Y arrays are each 4096 locations in length.

*AXIJ
idir,offset,(min,max,delta,ntrval,nsubnt),ndec,height

Draws a line with tic marks and labels.

idir = index to the axis desired (1=>x-axis, 2=>y-axis)
offset = the displacement of axis from origin (x-axis is displaced in +Y direction, y-axis is displaced in +X direction).
min = the value at start of axis line
max = the value at end of axis line
delta = a value of one intera val on axis
ntrval = the number of intervals desired
nsubnt = the number of subintervals per interval (default= 1)
ndec = the number of decimal places desired in labels. See section 4.216. If ndec is not specified labels will not be produced.
height = the height of labels in inches (default = .105 in).

Note that delta and ntrval are not independent and thus both need not be specified. The following table applies:

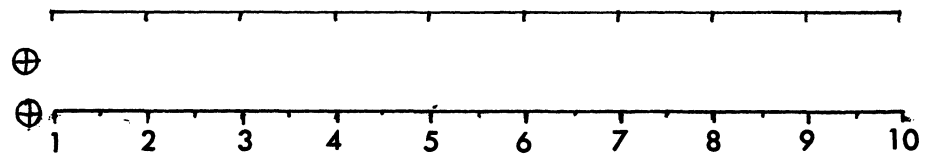
If	Then
delta=0,ntrval=0	ntrval=10,delta=(max-min)/10
delta=0,ntrval#0	delta=(max-min)/ntrval
delta#0,ntrval=0	ntrval=smallest integer greater than or equal to (max-min)/delta
delta#0,ntrval#0	max=min + ntrval * delta

If lticl=0 tic marks will not be produced (see *TIC/LENGTH)

Examples: The following records in the control data set:

```
*AXIJ
1,,(1,10,1)
1,,(1,10,,9,2),-1,.1
```

would cause the following two axes to be produced:



⊕represents the logical origin as established by the user and it is not actually drawn. The axis drawn (or extension of the axis) will pass through the logical origin. The length

of the tic marks is established by *TIC/LENGTH.

*BORDER ntrval,{xsw,ysw,xne,yne,} delta

Draws a border around an area.

ntrval = the number of intervals of X-Y coordinates to be created along border lines parallel with the x-axis. Note that ntrval need be greater than 1 only if a border line following a line of latitude is desired. In that case a curved line conforming to the projection requested is created. (See *REGION). Also, see below for special uses of ntrval.
xsw,ysw = coordinates of border corner with the minimum X and Y values
xne,yne = coordinates of border corner with the maximum X and Y values
delta = displacement (inches) of actual border drawn from extremes as defined by xsw,ysw and xne,yne (default =0)

If ntrval <= 0 xsw,ysw and xne,yne are not specified and a border is drawn "delta" inches outside the rectangle defined as follows:

- ntrval=0 - the rectangle encloses a feature previously processed
- ntrval<0 - rectangule encloses all features processed since the last occurrence of *MEASURE

*CALL EXIT xparm1,xparm2,-----,xparmn

Provides exits to user coded subroutine(s). EXIT would be written to provide capabilities not available through any of the * options.

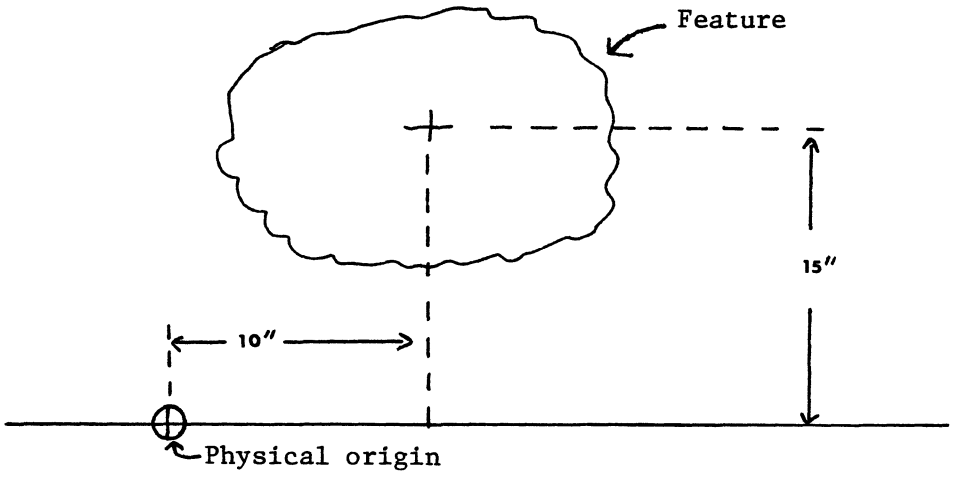
xparm: are parameters provided to EXIT.

*CENTER= xctr,yctr

Causes features to be plotted centered about the physical location (xctr,yctr).

xctr,yctr: coordinates (inches, relative to the physical origin) for the center (mid-range of X and Y extremes) of all subsequent features plotted. Note that this is the physical location on the plotter page before rotation (if rotation has been specified - see *ROTATE). Centering is performed on all features until a *CENTER record without the xctr,yctr parameters is encountered.

Example. The entry:
*CENTER=(10,15)
would cause a feature to be placed as in the following:



*CHECK= ncheck,xpghi

Causes testing for erronous plotter instructions.

ncheck: indicator of checking to be performed, default is 0
=0 - no checking is to be performed
=n (n>=1) - n points outside the plotter limits will be corrected and plotted. When n points have been so corrected execution will terminate.
xpghi: the maximum value (inches) to be accepted in the x-direction, default is 60.

The acceptable ranges of X and Y values (inches, physical) are:
minimum x=-10, maximum x=xpghi
minimum y=0, maximum y=27.5

If ncheck>=1 each X-Y value computed by GRAFIX will be compared to these thresholds. If a value exceeds a limit the value will be set to the limit and processing will continue. As sooon as ncheck "bad" points are encountered processing will terminate. Whenever a value is corrected a listing of the original and corrected values will appear on the log listing.

Example. The entry
*CHECK=(1,20)
specifies that execution is to terminate after the first "bad" point is encountered. It also specifies that only 20 inches are to be allowed in the +x direction.

*DATA/FORMAT = (datype,count,delim,invald),'dfmt'

Specifies the type and format of input X-Y data.

datype = index to type of data (1 => X-Y data; 2 => positional data - only y value is specified with x values assumed to be 1, 2,...; default is 1)
count = number of data items per logical record (0 => number is variable and data is read in a free form with commas separating data items and no comma after last data item, dfmt is not used. If #0 => data items read using dfmt, default is 0)
delim = value of data item to be interpreted as delimiter of data set - applicable only when count # 0 (default is 999)
ivald = the value of data item to be ignored when reading positional data - applicable when datype = 2 (default is 998)
dfmt = FORTRAN format to be used for reading X-Y data when count # 0.

Examples:

*DATA/FORMAT=(1,0) specifies X-Y data in free form

*DATA/FORMAT=(1,10,0),'(10F5.0)' specifies 10 items of X-Y data per logical record are to be read with the FORTRAN format F5.0. The occurrence of an x value of 0 terminates the data set.

```
*EMBLEM  {rform,ovrrd}
  isymbl,height,{pos,disp},{xdel,ydel},{(x1,y1,x2/angle,y2)}
  height',{pos',disp',}{xdel',ydel',}{(x1',y1',x2'/angle',
          y2'),} 'text'
```

Plots symbol together with textual data.

rform = indicator to presence of fields in following data records (default is 0)
10's digit of rform provides rform for symbol parameters
units digit of rform provides rform for text parameters (parameters with ')

ovrrd = height (inches) of symbol and text. If specified this height overrides height and height'.

isymbl = code number of the symbol to be plotted. See Figure A-1 and section 5.2.

height = the height (inches) of symbol

{pos,....,y2),} = location parameters for symbol and text see section 4.217

height' = height (inches) of text

text = string of textual data to be plotted. This string must be enclosed by apostrophes ('). If it is desired to plot an apostrophe, code a double apostrophe. For example:

To plot: don't Code: 'don''t'

The set of acceptable characters is found in the symbol listing. No more than 160 characters may be contained in one string.

*END =(xorg2,yorg2)

Indicates the end of control data set. Moves the pen to a specified location and terminates execution.

xorg2,yorg2: coordinates (inches) relative to the current physical origin to which the pen is to be moved before execution is terminated (default = max plot extent in x direction +10 inches, min of page in y direction)

This is the last record in the control data set.

Example. Assume a user created a plot 20"x10" with a physical origin in the lower left hand corner. The function record

```
*END=(30,0)
```

would move the pen 10" to the right of the finished plot.

*ERR.SET = errno,nterm,nmsg,itrace,iusadr,irange

Overrides specification in FORTRAN execution-time error handling routine.

errno: the number of the error
nterm: number of occurrences of error required to terminate execution; nterm>255 => no limit
nmsg: the number of times error message is to be printed
<0 - all messages suppressed
=0 - existing limit is not to be altered
=n - (1<=n<=255) - message printed for first n occurrences
=256 - no limit
itrace: traceback specification (default is 0)
0 - the existing specification is not to be altered
1 - no traceback is to be printed
2 - the traceback is to be printed
iusdr: (should be set to 0, default is 0)
irange: specifies upper end of range of errors for which specifications apply (default is 0)

Example. The entry
*ERR.SET=215,200,100,1
specifies that error condition 215 may occur 200 times before the run is to be terminated. Further a message will be printed for the first 100 occurrences and no traceback is to be printed.

*FACTOR =factr

Causes all plotting from this point to the occurrence of another *FACTOR entry to be scaled by factr.

factr = the scale factor (0<factr<∞ ; default= 1). All dimensions from this point forward will be multipilied by this factr.

Note that use of *FACTOR changes all dimensions in a plot including the height of lettering, symbols, etc. If it is desired to modify only the relative position of items (scale change only), *REGION should be used.

*FORMLINE=(form1,form2,...,formn),lsymbl,lsymblh

Specifies the format of line which is to be drawn.

form1 = the length (inches) of line drawn with pen down
form2 = length (inches) of line skipped with pen lifted
form3 = length (inches) of line drawn with pen down
.
.
lsymbl = code number of symbol to be drawn at each X-Y coordinate - if lsymbl=0 or is not specified no symbol will be drawn (0 is default). See the symbol table, Figure A-1.
lsymbh = the height (inches) of symbol (default= the.105)

If no formi are specified a solid line is drawn. This is the default condition. If only one formi is specified:

form1 <0 no line is drawn
form1 =0 solid line drawn
form1 #0 => a solid line is drawn but a cross-hatching (railroad tracks) is also drawn. Cross-hatches are form1 inches apart and .6*form1 inches long.

*GRID typpgrd
  (xl,x2,xd,yl,y2,yd,modlin),ndec,height,modlbl (typpgrd=0)
   (twnl,twn2,rngl,rng2,twnl,rngl,ltlbl,lrlbl),height,
      htslbl   (typgrd>0)

Draws a rectangular (typgrd=0) or township-range-section (typgrd>0) grid.

typgrd: specifies type of grid to be drawn (default is 0)
     0 - rectangular grid
     1 - township-range grid
     2 - section grid (township boundary not plotted)
     3 - township-range-section grid

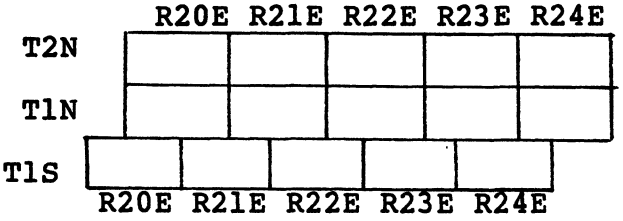
Township-Range-section Grid

twn1: minimum (southernmost) township to be plotted
twn2: maximum (northernmost) township to be plotted
Note: twn1,twn2 <0 => south, twn1,twn2 >0 => north
rng1: minimum (westernmost) range to be plotted
rng2: maximum (easternmost) range to be plotted
Note: rng1,rng2 <0 => west, rng1,rng2 >0 => east
twn1: index to township labelling
<=90 - number of township to contain range numbers
=91 - townships are to be labelled outside of grid on the western boundary of grid
=92 - townships labelled on eastern boundary
=93 - townships are to be labeled on east and west
Note: if twn1 not specified or if specified as 0 no labels will be produced
rng1: index to range labelling
<=90 - number of range to contain township number
=91 - ranges are to be labeled on south
=92 - ranges are to be labeled on north
=93 - ranges are to be labeled on north and south
Note: If rng1 not specified or if specified as 0 no labels will be produced
**Note: If twn1 or rng1 <=90 the township or range labelling will consist of the township or range number only, i.e. township 3 south would appear : 3. If twn1 or rng1 >90 the same township would be labelled: T3S.
ltlbl: location of range numbers in township (twn1<=90)
Range numbers will be placed:  as ltlbl=1,2, 3, or 4 (default is 2)

lrlbl: location of township numbers in township (twn1<=90)
Township numbers will be placed: a lrlbl=1,2, 3, or 4. (default is 2)

height: the height (inches) of township/range labels (default is .105)
htslbl: the height (inches) of section labels (typgrd>=2). If htslbl=0 sections will not be labelled. If htslbl>0 sections 1,6,31, and 36 will be labelled in the corners of the township.

Example. The entry
*GRID 1
(-1,2,20,24,91,93)
would produce:



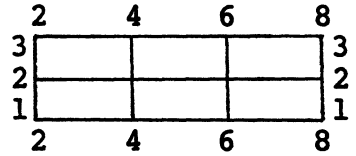
Rectangular Grid

xl: minimum x-value of lines parallel with y-axis
x2: maximum x-value of lines parallel with y-axis
xd: value of x-interval between grid lines parallel to the y-axis. If xd=0 no lines parallel to the y-axis will be drawn
yl: minimum y-value of lines parallel with x-axis
y2: maximum y-value of lines parallel woth x-axis
yd: value of y-interval between grid lines parallel to the x-axis. If yd=0 no lines parallel to the x-axis will be drawn
modlin: index of grid lines to be omitted, i.e. if 0<modlin=n grid lines 0,n,2n,... will be omitted where the grid line m is located at xl+m*xd or yl+m*yd (default is 0)
ndec: label format specification, see section 4.216
height: height (inches) of the label (default is .105)
modlbl: index of grid lines to be labelled/not labelled If modlbl=0 (default) and if ndec has been specified, then all grid lines will be labelled. If modlbl=-n then grid lines 0,n,2n,.. will not be labelled. if modlbl=+n then only grid lines 0,n,2n,.. will be labelled.

Example. The entry

```
*GRID
 (2,8,2,1,3,1),-1
```

would cause the following grid/labels to be drawn.



*INPUT/UNIT=uniti

Specifies a new refernece number for the control data set.

uniti: control data set refernece number

This entry is used to specify an alternate data set for input of further control records. Initially control records are read from data set 1 (FOR01.DAT). However some of the control records may appear in another data set. For instance when control records are generated by a program.

Example. Assume that control records have been generated in data set 10. The entry

```
     *INPUT/UNIT=10
```

appearing in data set 1 would cause further input of control records to be done from data set 10. The last record in data set 10 should be another *INPUT/UNIT entry of an *END entry.

*INSERT {valmod,space,ndec,height}

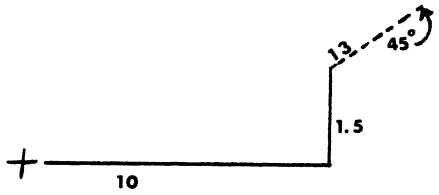
Causes labels to be printed in line data.

valmod = an index to value of lines to be labelled. If valmod=v, lines with value 0,v,2v,...will be labelled. If valmod=0 all lines will be labelled.
space = the intervals (inches) at which labels are to be inserted - space<1 => space is fractional part of line segment
ndec = the number of decimal places desired in label, see section 4.216
height = the height (inches) of labels (default is .105)

The "value" of a line is generally assigned outside of GRAFIX and is an identification number associated with the line. See discussions of *SELECT and *PLOT. Labels will be inserted in a line along the slope of the line. When labels are created an identification number is assigned to the label and sufficient information concerning the label is written to the output data set to enable a user of GRAFIX to uniquely identify each label on a finished plot. Using the *MOVE option a user can then alter the position of any inserted label which conflicts with other features plotted. If *INSERT is present without arguments the "insert" option is "turned off".

```
*LABEL rform,ovrrd
   height,{pos,disp,}{xdel,ydel,}{(x1,y1,x2/angle,
      y2)}ndec,value

Draws label(s) (numeric).

rform = indicates the format of following selection
        record(s) (default is 0), see 4.217
ovrrd = height (inches) of labels for all following labels,
        if specified.
height = the height (inches) of label(s)
{pos,...,y2)} = location parameters, see 4.217
ndec = the format desired for label, see 4.216
value = value to be plotted (can be symbolic parameter,
        see 4.215)

Example.
     *LABEL
      .1,(10,1.5.45),-1,@n2
```

specifies that the number of records selected in group 2 of the previous *SELECT operation (@n2) should be plotted at x=10, y=1.5 at a 45 degree angle. The label should be .1 inch high and should contain no decimal point (ndec=-1). Assume @n2=13:



"13" would be plotted as shown

*LIMITS ={xmin,xmax,ymin,ymax,unitt}

Constrains the finished plot to within specified ranges of X and Y values.

xmin = the minimum x values
xmax = the maximum x values
ymin = the minimum y values
ymax = the maximum y values
unitt = the identification number of data set to which line segments exceeding the limits are written. Data set will be written in unformatted mode (*SELECT Type 5 format) acceptable for subsequent retrieval by a *SELECT option.

The limits are specified in units applicable before projecting or scaling have been applied. If *LIMIT appears with no arguments all limits are removed. This is the default condition.

```
*LINE rform,ovrrd
    length,{pos,disp,}{xdel,ydel,}{(xl,yl,x2/angle,y2)}
```

Draws a straight line.

rform = indicates the form of following data records (default is 0). See location parameters, 4.217
ovrrd = the length (inches) of line. If specified, ovrrd overrides lengths specified in following data records.
length = the length (inches) of the line to be drawn
{pos,...,y2)} = location parameters, see 4.217

Line is drawn from xl,yl (as modified by pos,disp and xdel,ydel) in the direction specified by x2/angle,y2. If the length is specified as 0, the line will be drawn to x2,y2.

*LOG=log,incr

Specifies the extent to which the control data set and diagnostic information are to be listed on the log listing.

log: the index to extent of listing desired, default is 2
- 0 - no listing
- 1 - list function records (records with * in col 1)
- 2 - list function records and selection records, that is, the entire control data set
- 3 - list control data set, identifying information, and range of X-Y values for evey feature or display plotted
- 4 - everything in 3 above, plus listing of coordinates of all points in features plotted

incr: incremental index to X-Y coordinates to be plotted under log=4; that is, if log=4,i the values x1,y1, x+i,y+i, x+2i,y+2i,... will be listed. Default is 1

Example. The entry

```
*LOG=3
```

specifies that a user wishes his/her entire control data set listed. Further he/she wants diagnostic information such as the minimum and maximum X-Y values (inches, physical space) listed.

*MEASURE meas

Initiates measuring process.

GRAFIX maintains a record of the maximum extents of X-Y values which have been provided to the plotfile. These maximum extents are available as symbolic parameters (see 4.215). *MEASURE allows the user to re-start this measuring process at any time.

meas: indicator of action to be performed (default is 0)
- <0 - initializes all measures
- =0 - restores measures stored for meas>0
- >0 - temporarily stores existing measures and initiates new measuring.

A *MEASURE 1 would be issued to obtain the extents of a subset of a plot for use as symbolic parameters. After the subset extents had been used, a *MEASURE 0 would restore the extents of the entire plot. A *MEASURE -1 would initialize all measuring and would probably be used only when the physical origin had been moved.

*MOVE id1,l1,id2,...ln

Allows user to "move" a label which has been automatically inserted.

idi = identification number of the ith label to be moved
li = the length (inches) that the ith label is to be moved

Labels are automatically inserted in lines through use of the *INSERT option. In some cases a plot will be made and it is found that a label conflicts with some other plotted data. If log>=3 (see *LOG) sufficient information will have been provided to the output listing to uniquely identify each label. Through the use of *MOVE the offending label can be moved in a regeneration of the plotfile. If li>0, movement is to the right. If li<0, movement is to the left. If li is large enough to move the label off the line segment the label will not be produced at all.

*NEWPEN ipen

Creates a special record in plotfile so that plotting may be interrupted for the changing of pens. This option also primes the new pen.

ipen = pen to be used in plotting where 1<=ipen<=4 (default is first pen)

When *NEWPEN is encountered in the control data set a search address record is written to the plotfile. Each search address record is given a unique identifier number. A comment of the following form is written to the log listing:

----------SEARCH ADDRESS XXX----------

where XXXX is the search address number. If log>=1 (see *LOG) the particular search address number corresponding to each *NEWPEN entry should be readily ascertained. Note that locations 5 through n in the *NEWPEN record are available for any commentary desired (n is the length of the control record). The computer operator is able to selectively skip or plot portions of the plotfile by referring to the search addresses available. That is, the operater issues commands of the following form: locate f1 XXXX, pause f1 XXXX (where f1 refers to the plotfile on the tape) to the computer program processing the plotfile. Thus search addresses may be used for changing pens (new widths/colors) or to allow selective plotting from a plotfile.

** Currently the UM Calcomp plotter does not have any pen options.

*ORIGIN=(xorg2,yorg2)

Specifies a new physical origin.

xorg2,yorg2: coordinates (inches) relative to the current origin at which a new physical origin is to be established.

At the start of a GRAFIX run a physical origin will be established at the lower left-hand corner of the plotting paper. A user would generally issue an *ORIGIN entry to move the origin symbolically upwards and/or to the right, off the bottom left of the page.

Example. The entry

```
*ORIGIN=(1,5)
```

would establish a new physical origin (X=Y=0) one inch to the right and five inches above the current origin.

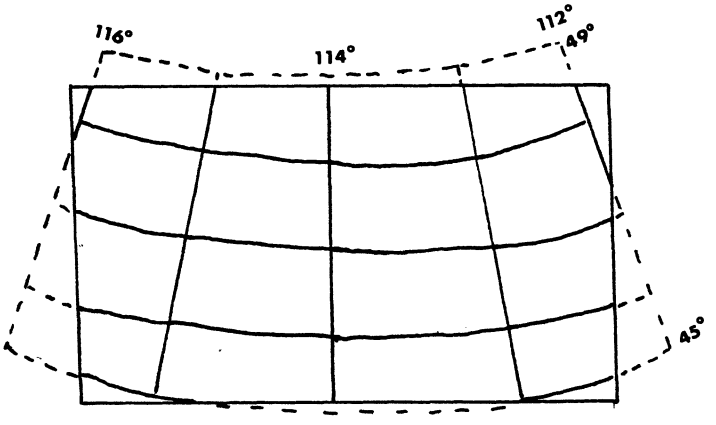
*OUTLINE=xnw,xne,yn,ys,outdel

Sets the limits for *GRID when a latitude-longitude grid is desired. Rectangular limits to the grid are also required. A border will be drawn if so specified.

xnw = the x-value of the northwest corner limit
xne = the x-value of the northeast corner limit
yn = the y-value of the north limit
ys = the y-value of the south limit
outdel = the displacement (inches) of plotted rectangle from the limits specified - if outdel is not specified, the rectangle is not plotted. Note that outdel can be specified as 0.

Example:

```
*OUTLINE=-116.1,-111.9,48.8,44.8,0.
*GRID
 (-116,-112,1,45,49,1)
```



Only the solid lines will be plotted.

*PLOT unitm,nfeat

Reads the coordinates of a line and draws the line

unitm = logical unit number of a data set containing coordinate data
nfeat = (was not explained in original manual) may indicate the number of features to be plotted

If unitm is not specified, data elements follow this *PLOT record in the current input data set. In this case data elements are separated by commas and are in free form. Data records will be read until encountering a data element not delimited by a comma.

If unitm is specified data elements will be read from the data set specified by unitm in one of two modes depending upon the value of count (see *DATA/FORMAT):

(1) count=0: Data is read in free form and data elements must be separated by commas. The last data element is not delimited by a comma.
(2) count#0: Data are read according to the FORTRAN format, dfmt, as specified by *DATA/FORMAT.

In either case data elements will be read as:
x1,y1,x2,y2,....,yn
unless datype=2 as set by *DATA/FORMAT. In this case the data elements are positional and read as:
y1,y2,...,yn while x1 is set to 1, x2 is set to 2,...

*RECORD/LENGTH=(lrecl1,lrecl2)

Defines the length (characters) of control data set records.

lrecl1: the length of the data area, initially set at 72
lrecl2: total length of each record, initially set at 80

This entry is used to specify the length of the control data set records if the default lengths of (72,80) are not applicable. The maximum lrecl2 that may be specified is 120. lrecl1<=lrecl2.

Example. The entry

```
*RECORD/LENGTH=(80,88)
```

specifies that control records are 88 characters long. However, control data are available in only the first 80 characters. Locations 81-88 may be used for sequencing, for identification, or for any other use desired.

*REGION=(xorgo,yorgo,xscale,yscale,proj,ipmu/ref/zone,corr)

Specifies the characteristics of logical space.

xorgo,yorgo: coordinates of point in logical space which is to coincide with physical origin, see *ORIGIN, default is (0,0)

xscale,yscale: scaling factors in the X and Y directions. If xscale is specified and yscale is not specified or if yscale is set to 0, yscale will be set equal to xscale, default is 1,1

proj: index to projection desired, default is 0
- 0 - no projection
- 1 - Transverse Mercator
- 2 - Montana state plane (Lambert conformal)
- 3 - Lambert conformal @330 and 450
- 4 - polygonic

impu: for proj=0, conversion factor - inches per mapping unit (converts logical space units to physical space units), default is 1

ref: for proj=1,3, or 4 longitude (degrees, signed) of reference meridian, must be specified if proj=1, 2, or 3 as there is no default value

zone: for proj=2, zone number for state plane projection (1=north, 2=central, 3=south), must be specified if proj=2 as there is no default value

corr: correction factor, inches per units of xdel,ydel, see section 4.217. The default depends upon proj (0=>impu, 1=>39.370432, and 2,3,4=>12)

Example. The entry
*REGION=(-116,44.4,1000000,,1,-110,1000000)
would specify that:
(1) the point where longitude=116°W and latitude= 44.4°N would coincide with the physical origin
(2) the scale in the x direction is 1:1,000,000. Since the scale in the y direction is specified as 0 (no entry before closing comma) y scale is also set to 1:1,000,000.
(3) the projection desired is Transverse Mercator and the reference meridian is 110°W
(4) each unit in xdel,ydel (see section 4.217) is 1,000,000 inches in length - since the scale is 1:1,000,000 this specification allows a user to express xdel,ydel data in inches

Example. The entry
*REGION=(0,0,1000,10000)
would specify that:
(1) the logical origin and the physical origin are coincident

(2) the scale in the x direction is 1:1,000 and
the scale in the y direction is 1:10,000
(3) since proj is not specified it is set
equal to zero and ipmu defaults to 1. That is,
one unit in logical space is one inch (since
the x scale is 1:1,000 one unit in logical
space is .001' on the plot. Ditto for
y)

*REWIND unitm

Causes the data set to be positioned such that the next record read will be the first record in the data set. The data set is rewound to its beginning record.

unitm = data set reference number (logical unit number) of the data set in question

*ROTATE=rotate

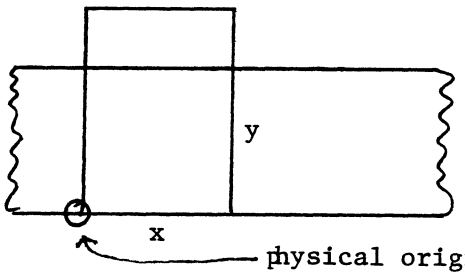
Specifies rotation of the entire physical plot.

rotate: The angle in degrees (rotated counter clockwise from the x-axis) desired, default is 0.

Currently there are only two acceptable values 0 and 90.

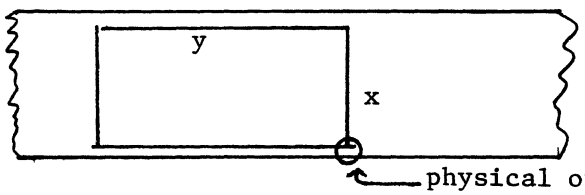
This entry is used to process a plot with range of y-values greater than 27.5 inches but with a range of x-values <=27.5 inches. The *ROTATE function can be turned "on" or "off" (rotate=0 or unspecified) at will. Thus this function can be used to rotate a plot inside another plot. Note that if *ROTATE=90 is specified a physical origin should be established sufficiently far in the positive x-direction that the rotated plot will be properly placed. This function is independent of all other functions. Thus a user should continue to think of a vertical y-axis and a horizontal x-axis in creating a plotfile.

Example.



unrotated plot

With *ROTATE=90
and *ORIGIN=(y,1):



rotated plot

*SAVE/UNIT=units,idsav

Specifies that X-Y coordinates (physical space) of plotted features are to be written to a data set.

units = reference number of data set to which identifying number and X-Y coordinate is to be written. Data set written will be in *SELECT Type 5 format. If units=0 (default) or if units is not specified, these data will not be written.

idsav = the starting identification number for feature(s). Applicable only if no identification number is available for feature as would be true if feature were obtained with *PLOT option.

```
*SELECT typfil,unitm
     (<------selection test parameters--->{:action}),----
```

This causes the reading of a data set consisting of point, property, or polygon data, tests records according to user specified criteria and performs user specified action when records are accepted.

typfil: type of file to be read. Currently 6 types of files are recognized and are discussed in detail on the following pages.
unitm: The data set reference number of the data set in question

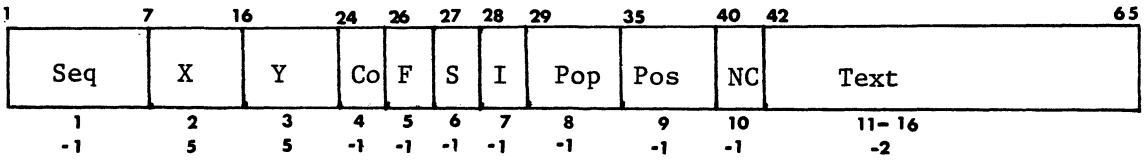
The selection test parameters are discussed in section 4.218.

Action parameters vary according to the type file and are discussed in the following pages.

On the following detailed discussions of fiile types these conventions will apply:

(1) the starting location in record is written above the field
(2) field number for test purposes is below the field
(3) field format below the field number
  -2 = character format (test fields enclosed in ')
  -1 = integer data
  >=0 = number of deciaml places (real)

*SELECT Type 1



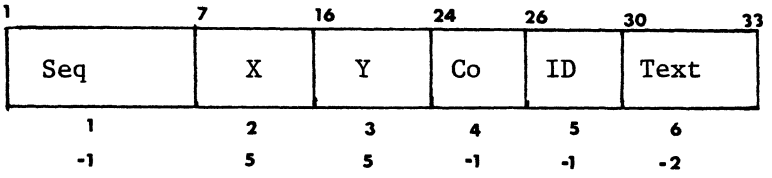
Fixed length records, formatted file. Primarily used for city/town location data.

seq = sequence number (file not required to be in sequence)
x = x-coordinate
y = y-coordinate
co = county
F = 0/1 if digitized boundary available(0)/not available(1)
S = 0/1 if is not(0)/is county seat(1)
I = 0/1 if is not(0)/is census place(1)
pop = population
pos = orientation of text relative to X,Y. See 4.217.
nc = the number of characters in text
text = a text string (<= 24 characters)

Note that only the X and Y fields are required to have the meaning given above.

Action parameters {:isymbl,hts,htt}
isymbl: number of symbol to be drawn at X,Y (see fig.A-1)
hts: height of symbol (if hts=0 or is omitted, no symbol will be drawn
htt: height of text string and displacement of text from X,Y (if htt=0 or is omitted no text will be plotted)

*SELECT Type 2



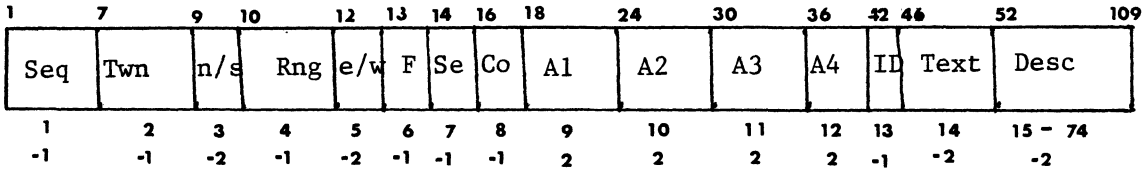
Fixed length records, formatted file.

seq = sequence number (file not required to be in sequence)
x = x-coordinate
y = y-coordinate
co = county
ID = identification number
text = any text

Note that only X and Y are required to have the meaning given above.

Action parameters {:label,height}
label: labelling desired (centered at X,Y)
=n (n<0) field n is to be plotted as text
=0 no labelling
=n (n>0) symbol number n
='text' literal to be plotted (<= 4 characters)
height: height of label (default is .105)

*SELECT Type 3



Fixed length records, formatted file. Legal description of property.

seq = sequence number (file not required to be in sequence)
twn = township
n/s = direction of township (north/south)
rng = range
e/w = direction of range (east/west)
F = 0/1 if township is not/is fractional township
se = section
co = county
Ai = arbitrary. This might be used for area of a parcel
ID = identification number
text = text
desc = legal description. An abbreviated form of the legal description is coded. (1/2, 1/4 can be written as 2 or 4 or can be omitted entirely). Parcels are separated by commas or semi-colons. Thus the following legal descriptions might appear:

legal description	as coded
all of section 4	ALL
N1/2 of SE1/4 and SE1/4SE1/4	N2SE4;SW4SE4 or NSE;SWSE

Note that only twn, n/s, rng, e/w, and se are required to have the meaning given above.

Action parameters {:level<label,height}
level: geographic area (<0 => boundary not to be plotted)
- ±1 section only
- ±2 parcel(s) only (if parcel description is not able to be interpreted, an estimation of the parcel will be based on Ai/640 if |level| = 10*i+2)
- ±3 process section and parcel(s)

label: labelling desired (at center of section/parcel)
- =n (n<0) field n is to be plotted as text
- =0 no labelling
- =n (n>0) symbol number n (see Fig. A-1)
- ='text' literal to be plotted (<=4 characters)

height: the height of the label (default is .105)

An alternate action of the form {:level,slopes,step} is available if shading of parcels is desired, see Type 5.

*SELECT Type 4

1	7	9	10	12	13	14	16	18	22	26	29
Seq	Twn	N/S	Rng	E/W	F	Se	Co	ID	Text	Tract	
1 -1	2 -1	3 -2	4 -1	5 -2	6 -1	7 -1	8 -1	9 -1	10 -2	11 - 14 -2	

Fixed length records, formatted file. USGS quarter-quarter tract description.

seq = sequence number (file not required to be in sequence)
twn = township
n/s = direction of township (north/south)
rng = range
e/w = direction of range (east/west)
F = 0/1 if township(1) is not(0)/is fractional township(1)
se = section
co = county
ID = identification number
text = text
tract = USGS description of tract. Quarters are denoted by A, B, C, and D:

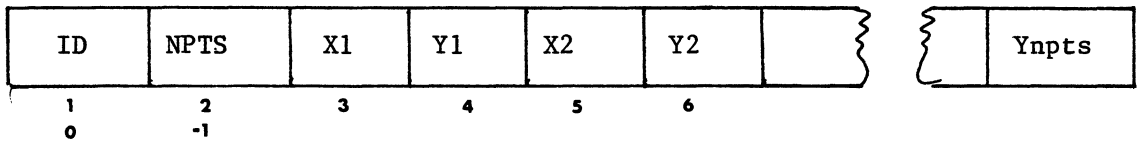
B	A
C	D

The SW1/4 of the NE1/4 would appear : AC. Note that the smallest tract that can be described is 1/256 of a section.

Note that only twn, n/s, rng, e/w, and se are required to have the meaning above.

Action parameters are the same as for Type 3.

*SELECT Type 5



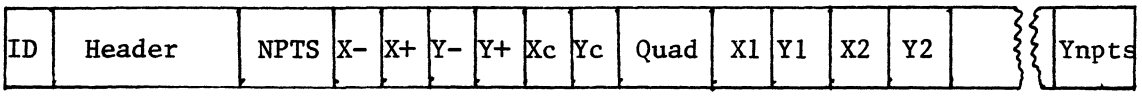
Variable length records, unformatted FORTRAN file.
Line and polygon data.

ID = arbitrary identification number (note that this is a real number)
npts = number of X-Y pairs following
Xi = x-coordinate i
Yi = y-coordinate i

Action parameters {:±1,label,height}
±1 - +1 line is to be plotted
-1 line is not to be plotted
label = labelling desired (at center of feature)
=n (n<0) field n is to be plotted as text
=0 no label (default)
=n (n>0) symbol number (see Fig.A-1)
='text' literal to be plotted (<=4 characters)
height = height of label (default is .105)

Alternate action parameters {:±1,slopes,step}
where slopes is of the form Sn. This form specifies that shading (cross-hatching) of the feature selected is to be performed. The integer n denotes ishade, i.e. the angle(s) of the shading lines. See the discussion of *SHADE for further information. If an *SHADE entry is present while *SELECT with shading is in effect both shadings specified will be performed.

*SELECT Type 6



Variable length records, unformatted FORTRAN file.
Line and polygon data.

ID = identification number
header = a text description of the feature
npts = number of X-Y pairs of coordinates
X- = minimum x extent of the feature
X+ = maximum x extent of the feature
Y- = minimum y extent of the feature
Y+ = maximum y extent of the feature
Xc = the mid-range of x extents
Yc = the mid-range of y extents
quad = logical variable dimensioned: LOGICAL*1 (29)
Xi = x coordinate i
Yi = y coordinate i

Note that only npts, Xi, and Yi must have the meaning above.

Action parameters are same as for Type 5.

*SHADE=ishade,step

Causes shading (cross-hatching) to occur for all poygons plotted until an occurrence of *SHADE with ishade=0 or *SHADE with no arguments.

ishade: denotes the angle(s) of the lines used for shading. ishade consists of one or more of the digits 1, 2, ...,8 where:
1 => 0 degrees
2 => 22.5 degrees
3 => 45 degrees
4 => 67.5 degrees
5 => 90 degrees
6 => 112.5 degrees
7 => 135 degrees
8 => 157.5 degrees
Thus ishade=37 specifies that shading lines at 45 degrees and at 135 degrees are to be drawn. All eight angles may be specified if desired.

step: denotes the spacing (inches) between shading lines.

Example. The entry

```
*SHADE=5,.1
```

specifies that vertical (angle=90 degrees) shading lines spaced .1 inch apart are to be drawn inside all further features.

*SYMBOL rform,ovrrd
isymbl,height,{pos,disp,}{xdel,ydel,}{(x1,y1,x2/angle,y2)}

Draws a symbol.

rform = indicator of format of data record(s) to follow (default is 0). See location parameters in section 4.217.
ovrrd = height (inches) of symbol. If specified, ovrrd, overrides heights specified in following data records
isymbl = code number of symbol to be drawn. See Fig. A-1.
height = the height (inches) of symbol
{pos,.....,y2)} = location parameters. See section 4.217.

```
*TEXT rform,ovrrd
  height,{pos,disp,}{xdel,ydel'}{x1,y1,x2/angle,y2)}'text'

Plots textual data.

rform = indicator of format of following data record(s)
        (default is 0).  See location parameters, section
        4.217.
ovrrd = height (inches) of textual data.  If specified,
        ovrrd, overrides heights specified in following
        data records.
{pos,....,y2)} = location paramters, see 4.217
text = string of textual data to be plotted.  This string
        must be enclosed by apostrophes (').  If it is
        desired to plot an apostrophe, code it as a
        double apostrophe, eg:
          To plot:  don't        Code:'don''t'
        The set of acceptable characters are found in the
        symbol table, Fig. A-1. No more than 160 characters
        may be contained in one string.
height = the height (inches) of textual data
```

*TIC/LENGTH=(lticl,ltic2)

Specifies length of tic mark lines.

lticl = length (inches) of major tics (interval tics)
default is .1
ltic2 = length (inches) of minor tics (subinterval tics)
default is .05

*TRANSLATE TO (xorg2t,yorg2t)

Specifies temporary translation of physical origin.

xorg2t,yorg2t = coordinates (inches) of physical origin. All furhter plotting commands will be referenced to xorg2t,yorg2t until the occurrence of another *TRANSLATE or the occurrence of an *ORIGIN which sets xorg2t,yorg2t=(0,0). Default for xorg2t, yorg2t is (0,0).

This function would normally be used when more than one *REGION is required in a single plot.

Example. The entry

```
*ORIGIN=(2,2)
*REGION=(0,0)
*AXIJ
  1,,(0,10,1),-1,.1
*TRANSLATE TO (10,0)
*REGION=(0,50,10)
*TIC/LENGTH=(-.1)
*AXIJ
 2,,(50,100,10),-1,.1
```

would yield: