UNITED STATES DEPARTMENT OF THE INTERIOR

GEOLOGICAL SURVEY

WILDMAP, a FORTRAN computer program for the plotting of digital boundaries and mineral locations for the assessment of a given area's resources.

by

E.G. Boyce, Jr.

Open-File Report 84-651

"This report is preliminary and has not been reviewed for conformity with U.S. Geological Survey editorial standards. (Any use of trade names is for descriptive purposes only and does not imply endorsement by the USGS.)"
<table>
<thead>
<tr>
<th>CONTENTS</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td>Computer restrictions</td>
<td>1</td>
</tr>
<tr>
<td>Program limitations</td>
<td>2</td>
</tr>
<tr>
<td>Input data</td>
<td>2</td>
</tr>
<tr>
<td>Design and usage</td>
<td>4</td>
</tr>
</tbody>
</table>
## APPENDICES

<table>
<thead>
<tr>
<th>Appendix</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Appendix A</td>
<td>Links</td>
<td>7</td>
</tr>
<tr>
<td>Appendix B</td>
<td>Sample terminal session</td>
<td>8</td>
</tr>
<tr>
<td>Appendix C</td>
<td>Coastline and political files</td>
<td>9</td>
</tr>
<tr>
<td>Appendix D</td>
<td>WILDMAP FORTRAN program listing</td>
<td>10</td>
</tr>
<tr>
<td>Appendix E</td>
<td>MRDS FORTRAN program listing</td>
<td>25</td>
</tr>
<tr>
<td>Appendix F</td>
<td>Available projections</td>
<td>26</td>
</tr>
<tr>
<td>Appendix G</td>
<td>Subroutines employed</td>
<td>28</td>
</tr>
</tbody>
</table>
Introduction

This paper describes the WILDMAP program and gives the information needed to use the program effectively.

The WILDMAP program was written for the purpose of producing computer-generated maps of digital line data such as wilderness, forest, and geologic boundaries and associated mineral locations. These boundaries and locations are plotted on geographic base maps containing either political boundaries such as a given state with county boundaries or perhaps coastline boundaries of the world. The user may select a specific geographic area from which to view the data. When choosing a latitude-longitude window he may provide a title and place tick lines at a specified interval, and the program will insert them and label the axes accordingly. The program has the ability to plot multi-colored maps showing the various type lines in different colors. If mineral data are being plotted, a legend indicating which symbols are used and their significance is available. The points themselves may or may not be annotated. If so, the name, and commodities present will be shown.

These features allow the researcher to view on one map the existing mineral information and the geology which occur within a particular wilderness or forest land.

The map provides a tool to aid in the mineral assessment of wilderness or forest areas.

Computer restrictions

The WILDMAP program was written for the Honeywell Multics system and is designed for interactive use. The program incorporates several system subroutines which render it machine dependent. Refer to Appendix G for a list of these subroutines. It was written to be used in conjunction with
either a CALCOMP plotter or a Tektronix screen display and, therefore, one of these must be used in plot production. As a minimum, the user must know the "login" procedure for the Honeywell Multics System and how to link to WILDMAP and the various files required with WILDMAP for execution. Refer to Appendix A for instructions regarding the necessary links and the proper commands to obtain them.

Program limitations

WILDMAP uses software and hardware on the Honeywell Multics computer located in Reston, Virginia, which may not be available elsewhere (see Appendix A). Namely:

DISSPLA - a proprietary package employed for the purpose of plotting all coastline and political boundaries as well as the map border, title, grid, and axis labels.

Hardware used:

TEKTRONIX - The "prevuu" routines make DISSPLA compatible with the Tektronix hardware thereby allowing the user to preview the plot before generating a CALCOMP plot.

CALCOMP - giving DISSPLA the ability to interface with the CALCOMP plotter.

Input data

There are several possible input files to the program depending upon the type of map the user desires. During a single execution of the program, the user may plot one or more data files. The following formats should be used for input:
### MRDS boundary file (see figure 1)

<table>
<thead>
<tr>
<th>DATA NAME</th>
<th>POSITION</th>
<th>DESCRIPTION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>X</td>
<td>1 - 10</td>
<td>X-axis coordinate</td>
<td>longitude degrees</td>
</tr>
<tr>
<td>Y</td>
<td>11 - 20</td>
<td>Y-axis coordinate</td>
<td>latitude degrees</td>
</tr>
<tr>
<td>depo</td>
<td>22 - 58</td>
<td>deposit name</td>
<td>alpha. chars.</td>
</tr>
<tr>
<td>depotype</td>
<td>59 - 59</td>
<td>deposit type code</td>
<td>&quot;1&quot; thru &quot;5&quot;</td>
</tr>
<tr>
<td>commod</td>
<td>60 - 109</td>
<td>deposit commodity</td>
<td>alpha. chars.</td>
</tr>
<tr>
<td>irotanot</td>
<td>111 - 111</td>
<td>annotation rotation factor</td>
<td>&quot;1&quot; thru &quot;5&quot;</td>
</tr>
</tbody>
</table>

### GEOLOGIC boundary file

<table>
<thead>
<tr>
<th>DATA NAME</th>
<th>POSITION</th>
<th>DESCRIPTION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>header</td>
<td>iden</td>
<td>identification number</td>
<td>integer</td>
</tr>
<tr>
<td>record</td>
<td>leftid</td>
<td>annotation left of line</td>
<td>alpha. chars.</td>
</tr>
<tr>
<td></td>
<td>righid</td>
<td>annotation right of line</td>
<td>alpha. chars.</td>
</tr>
<tr>
<td></td>
<td>npoints</td>
<td>number of points in line segment</td>
<td>integer</td>
</tr>
</tbody>
</table>

### FOREST boundary file

<table>
<thead>
<tr>
<th>DATA NAME</th>
<th>POSITION</th>
<th>DESCRIPTION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>header</td>
<td>type</td>
<td>forest code</td>
<td>integer</td>
</tr>
<tr>
<td>record</td>
<td>numbr</td>
<td>number of points in line segment</td>
<td>integer</td>
</tr>
</tbody>
</table>

### WILDERNESS boundary file (see figure 2)

<table>
<thead>
<tr>
<th>DATA NAME</th>
<th>POSITION</th>
<th>DESCRIPTION</th>
<th>UNITS</th>
</tr>
</thead>
<tbody>
<tr>
<td>header</td>
<td>kount</td>
<td>number points in the segment</td>
<td>integer</td>
</tr>
<tr>
<td></td>
<td>attragay</td>
<td>agency code</td>
<td>integer</td>
</tr>
<tr>
<td></td>
<td>attrilude</td>
<td>type code</td>
<td>integer</td>
</tr>
</tbody>
</table>

| detail    | X(h)     | X-axis coordinate | real number |
| record    | Y(h)     | Y-axis coordinate | real number |

| thru       | X(n+1)   | X-axis coordinate | real number |
|           | Y(n+1)   | Y-axis coordinate | real number |
Data are obtained by performing a retrieval from the Mineral Resource Data System (MRDS-GIPSY manual (Orris and Stoltz, 1982) data base. Refer to the MRDS manual for instructions on how to perform the retrieval. The current method of operation is to transfer the output to a SUPERBRAIN microcomputer and then to the MULTICS computer.

Once the data have been transferred to the MULTICS computer, they should be prepared to be read by the WILDMAP program. The latitude and longitude coordinates must be converted to decimal figures and negative signs inserted for southern latitudes and western longitudes. An additional program (see Appendix E) is available to perform this task unless the user would like to use an alternative means such as text editor. MRDS FORTRAN is a program to aid in the conversion of MRDS data from the AMDAHL computer to a format acceptable to the WILDMAP program. It reads the file derived from a MRDS retrieval and converts latitude and longitude figures to decimal format. A negative sign will be used for points of west longitude and south latitude. It will also place a three in the deposit type column as well as the annotation rotation factor column. Refer to Appendix E for format statements and a listing of the program. WILDMAP also uses a deposit type code and an annotation rotation factor which must be placed in each line. The annotation rotation factor determines the angle of the annotation text. A "1" indicates 90 degrees rotation to the north, "2" is 45 degrees north, "3" is no rotation, "4" is 45 degrees rotation to the south and "5" indicates 90 degrees to the south.

The digitized forest boundary file is obtained from the U.S. Forest Service. Geologic boundaries come from the Johns Hopkins Applied Physics Lab and the Wilderness area boundaries are provided by the National Mapping Division. For input format statements see program listing in Appendix D.
Design and usage

After linking to WILDMAP as described in Appendix A, enter "WILDMAP" to initiate execution. The WILDMAP program will then request the transmission rate of the terminal currently being used. You will then be asked whether the output is to be directed to a Tektronix screen, a CALCOMP plotter, or saved in a Multics segment entitled "pfile". If a CALCOMP plot is desired, the tape should have already been taken to the computer room; otherwise a message to this effect is issued and the execution is terminated.

Once these preliminaries are established, you are prompted for a projection type. The various projections available are listed in Appendix F along with their appropriate explanations. For some of the more distorted projections, such as Mollweide, Sinusoidal or Gnomic, a calculation to cause the projection center to be the map center may be required. This feature is accomplished by answering "yes" when so prompted. The default is the Greenwich-equator intersection. Next you will be requested to supply a title for the map. The response may contain as many as 48 characters and will appear in upper case letters on the map, regardless of the form in which it was entered. You will then be asked whether a state map with county boundaries is desired. If so, enter the postal abbreviation (first letter upper case, second letter lower case) of the state to be mapped. You should have already linked to the proper state boundary file (see Appendix A) at this point. In the current version of WILDMAP only one state may be mapped at a time. An additional prompt is issued to enable you to select a geographic window of your own choice. In this case, you must enter the range of longitude, from the minimum to the maximum, as well as
the interval number of degrees between grid lines. This information is then required for the range of latitude and grid intervals. Coordinates of south latitude and west longitude should be preceded with negative signs. You are then prompted to specify which, if any, coastline and/or political files are to be used in the generation of your map. Refer to Appendix C for a list of available files. The program will then inquire as to whether you have a file of wilderness areas you would like to plot, and if so will ask if you would like to select areas within agency and wilderness type. A prompt for the name of the file is then given. Next a line which asks if a MRDS point file is to be plotted will appear and subsequently, if appropriate, whether annotations are desired or not. The MRDS data file name is then requested. The following prompt is for a forest boundary file and its corresponding name. The last file to be entered is the geologic data file, if one is to be plotted. At this point the necessary data are present and the program gives the option of restarting should there have been errors in entering information.
Figure 1.--Sample MRDS data input.
Figure 2.—Sample Wilderness data input.
Figure 3.--Sample WILDMAP output Wilderness boundaries and MRDS locations.
Appendix A

- Links -
Enter "link" followed by:
>udd>ORA>library>WILDMAP
>udd>ORA>library>assoc
>udd>ORA>library>closer
>udd>ORA>library>statwndo
>udd>ORA>library>grasp
>udd>ORA>library>Ak
through
>udd>ORA>library>Wy

- Search Rules -
Enter "asr" (assign search rules) followed by:
>mr82_fortran -be >system_library_unbundled
Appendix B

- Sample Terminal Session -

[WILDMAP] (Brackets indicate user response)

Enter terminal speed ('9600' '1200' '300' etc.): [9600]

Choose either a calcomp plot 'cal'
or a tektronix screen display 'tek' [tek]

Enter desired projection: [merca]

Do you want Projection Center at the Map Center?: [no]

Enter appropriate title: [las cruces wilderness & mrds data]

Do you want a STATE MAP with COUNTY BOUNDARIES?: [no]

Enter longitude min, max, and step size: [-107.25,-106.25,1]

Enter latitude min, max, and step size: [32.25,32.75,.5]

Do you wish a coastline file? ('yes' or 'no'): [no]

Do you wish a Political File? ('yes' or 'no'): [no]

Do you have a Wilderness Area File to plot? ('yes' or 'no'): [no]

Would you like to select areas within agency? ('yes' or 'no'): [no]

Would you like to select areas within wilderness type? ('yes' or 'no'): [no]

Enter name of file with Wilderness Area coordinates: [nmexwild]

Do you have a MRDS point file to be plotted? ('yes' or 'no'): [yes]

Would you like the points annotated? ('yes' or 'no'): [yes]

Enter name of file with MRDS data points: [cribdata]

Do you have a file of forest boundaries to be plotted? ('yes' or 'no'): [no]

Do you have a file of geologic data to be plotted? ('yes' or 'no'): [no]

Do you wish to START OVER AGAIN? ('yes' or 'no'): [no]
Appendix C
- Coastline and Political Files -

Coastline Files: 'MAPDTA'
 'HERSHEY'
 'AFRICA'
 'ANTARCTIC'
 'ASIA'
 'AUSTRALASIA'
 'EUROPE'
 'NORTH AMERICA'
 'SOUTH AMERICA'
 'COASTLINES' (of the world)

Political Files: 'PAFRICA'
 'PASIA'
 'PAUSTRALASIA'
 'PNORTH AMERICA'
 'PSOUTH AMERICA'
 'POLITICAL' (of the world)
 'USTATESx' (x = 0 thru 9,
   0 = best resolution)

All files are described in the DISSPLA manual.
Appendix D

(WILDMAP)

Program Listing

WILDMAP.fortran  mod. 05-29-84

A basic mapping program using DISSPLA to
generate geographic map plots of data files. Output is to TEKTRONIX SCREEN or as
a CALCOMP plot tape or plot file on Multics.

OPTIONS:

Plot

- CALCOMP
- Plot tape
- Multics plot file

- TEKTRONIX
- screen display

Map

- Projection (12 different, + user defined )
  - Projection center at map center, if desired.
- State map with county boundaries
  - Geographic window, select or default.
- Geographic window and Lat/Long Grid.
- Geographic Boundaries
- Political Boundaries
- Wilderness Boundaries
- Forest Boundaries
- MRDS mineral locations
  with or without annotations

INPUT FORMAT: ___1234.567___1234.56 (where "_"= space)
(2f10.3)

***** 
***** Multics and DISSPLA routines 
*****
external ioa_$nnl(descriptors), setup_tektronix_tcs(descriptors), asr(descriptors), dsr(descriptors)
external close_file(descriptors), project(descriptors), setup_calcomp(descriptors), assoc(descriptors)
external lines(descriptors)
external linest(descriptors)
external legend(descriptors)
external ioa_(descriptors)
external rlmess(descriptors)
external reset(descriptors)
external mylegn(descriptors)
external title(descriptors)
external mapfil(descriptors)
external newpen(descriptors)
dimension ititl(48)
dimension x2(1500), y2(1500)
dimension x(1000), y(1000), t1(1000), t2(1000), mark(1000), pscale(1000), recnum(1000)
dimension state(50), xxmin(50), xxmax(50), yymin(50), yymax(50), xo(300), yo(300)
dimension xxx(1000), yyy(1000) ough "info segments" such as this one.
dimension depo(1000)
dimension commod(1000)  e name of a topic and the system will print th
dimension irotanot(1000)
dimension xfor(1000), yfor(1000)
dimension xgeo(1000), ygeo(1000)
dimension depotype(1000)
dimension depoleg(3)
dimension xlegenray(100)
dimension xleg(3)
dimension yleg(3)
data iblnk, idol/" ", ", ", ", ", ", "$"/
data i 10/0/
c ***** variable type definitions *****
c ***** variable type definitions *****
integer irotanot
integer wait
integer wait2
integer first
integer maxarray
integer type
integer numbr
integer titl(12)
integer attragcy
integer attrwildcde
integer agcynum
integer wildnum
integer depotype
integer depoleg
integer prevtype
integer j
real degrot
character titlans
character lineone
character linetwo
character linethree
character*20 turner,cfile,pfile,line
character*8 fildegs
character*8 cribfil
character*20 xlegenray
character*37 depo
character*50 commod
character*3 restart
character*3 resp
character*3 resp2
character*3 skip
character*3 skip1
character*3 skip2
character*3 skip3
character*3 endfile
character*3 dsplq
character*6 calcq
character*3 tapemt
character*3 maptyp
character*2 mapnam
character*8 forstfil
character*8 geolfile
character*8 fname
character*8 statwndo
character*3 window
character*2 state
character*3 forstans
character*3 geoans
character*3 selcde
character*3 cribans
character*3 wildcde
character*3 anotans

c ***** prompts user to determine physical characteristics *****
c ******
101 call ioa_$nnl( A /Enter terminal speed (°9600° 1200° 300° etc.):")
   read(5,1T5)ispeed
   ispeed=ispeed/10
   call ioa_$nnl( A/Choose either a calcomp plot °cal°)
   call ioa_$nnl( A/or a tektronix screen display °tek°)
200 read(5,1T5)dsplq
   if(dsplq.eq."cal")go to 210
if(dsplq.eq."tek") go to 220
call ioa_$nnl("A/enter only "tek" or "cal")
go to 200

210 call ioa_$nnl("A/choose a plot tape with a 6 digit name: ______")
call ioa_$nnl("A/or an output file named: pfile")
read(5,115)calcq
if(calcq.eq."pfile") go to 240
call ioa_$nnl("A/Is the tape upstairs...AND have you asked the operator to locate it?")
read(5,115)tapemt
if(tapemt.eq."yes") go to 230
call ioa_$nnl("A/Go tend to it, and THEN come back")
go to 300

230 call asr(">iml>calcomp")
call setup_calcomp("-un","16","-tp","-nm",calcq)
go to 245

240 call asr(">iml>calcomp")
call setup_calcomp("-un","16","-fl","-nm","pfile")

245 call asr(">iml>disspla")
call calcomp(16)
call inch30
call page(30.,30.)
call nobrdr
go to 5

220 call setup_tektronix_tcs
4 call asr(">iml>calcomp")
call asr(">iml>disspla")
call nhance(ispeed)
call page(10.,8.)
go to 5

105 call setup_calcomp ("-next_file")
c ***** prompts user to set map parameters *****
c *****
5 call ioa_$nnl("A/Enter desired projection:")
read(5,100)turner
call nochek
call grace(0.0)
call ioa_$nnl("A/Do you want Projection Center at the Map Center?:")
read(5,115)skip3

100 format(a20)
skip="yes"
skip3="yes"
c ***** read title (up to 48 characters) *****
c *****
call ioa_$nnl("A/Enter appropriate title:")
read(5,110)titl
110 format(48a1)
1=48
113 l=l+1
ititl(1)=idol
c encode(titl,110)ititl
c
c 109 continue

c
--------** SELECT STATE MAP WITH COUNTY BOUNDARIES **--
c
111 call ioa $nnl("A/Do you want a STATE MAP with COUNTY BOUNDRIES ?:")
   read(5,115)maptyp
   if(maptyp.ne."yes")go to 114
   do 112 i=1,48
       if(ititl(i).ne.iblnk)go to 113
   112 i=i-1
   print,"A single state at a time can be mapped; Select the state using the "
   print,"TWO LETTER postal abbreviation, with the First Letter CAPITALIZED"
   8 call ioa $nnl("A/Enter Abreviation of the State desired ,(2 char.),(i.e. Az = Arizona):")
   c
   c *** SELECT GEOGRAPHIC WINDOW FROM STATE FILE ***--
c
call assoc(15,"statwndo","si ")
do 9 i=1,48
6 read(15,7,end=99)state(i),xxmin(i),xxmax(i),yymin(i),yymax(i)
7 format(a2,2f4.0,2f3.0)
   if(mapnam.eq.state(i))go to 10
9 continue
   call closer(15)
go to 99
10 continue
   call closer(15)
   print,"State= ",state(i)," window= ",xxmin(i)," ",xxmax(i)," ",yymin(i)," ",yymax(i)
   call ioa $nnl("A/Do you wish to select your own Geographic Window?:")
   read(5,115)window
   if(window.eq."yes")go to 114
   skip="no"
skip1="no"
xstp=2
ystp=2

   --------** Assign Default Window ****--
c
   xmin=xxmin(i)
   xmax=xxmax(i)
   ymin=yymin(i)
   ymax=yymax(i)
   go to 65
99 print,"ABBREVIATION ERROR"
   print,"Use POSTAL ABBREVIATION"
   go to 8

*****

***** user selected geographic window *****
c
*****
114 call ioa_$nnl("A/Enter longitude min, max, and step size:")
        read(5,115)xmin,xmax,xstp
115 format(v)
        call ioa_$nnl("A/Enter latitude min, max, and step size:")
        read(5,115)ymin,ymax,ystp

C ***** select data files for program input *****
C *****
C ***** coastline and political files taken *****
C ***** from DISSPLA *****
C *****
50 call ioa_$nnl("A/DO you wish a coastline file ? ('yes' or 'no'):")
        read(5,115)skip
        if(skip.eq."no")go to 55
        call ioa_$nnl("A/Enter coastline file desired:")
        read(5,100)cfile
55 call ioa_$nnl("A/DO you wish a Political File ? ('yes' or 'no'):")
        read(5,115)skip1
        if(skipl.eq."no")go to 65
60 call ioa_$nnl("A/Enter political file desired:")
        read(5,100)pfile
65 call ioa_$nnl("A/DO you have a Wilderness Area File to plot ? ('yes' or 'no'):")
        read(5,115)skip2
        if(skip2.eq."no")go to 600

C ***** prompts user for restriction by wilderness agency *****
C *****
call ioa_$nnl("A/Would you like to select areas within agency ? ('yes' or 'no'):")
        read(5,115)selcde
        if(selcde.eq."no")go to 309
        call ioa_("1 = Bureau of Land Management"")
        call ioa_("2 = U.S. Forest Service"")
        call ioa_("3 = National Park Service"")
        call ioa_("4 = Fish and Wildlife Service"")
        call ioa_$nnl("A/Enter the number of the agency desired :")
        read(5,115)agcynum

C ***** prompts user for restriction by wilderness type *****
call ioa_$nnl("A/Would you like to select areas within wilderness type ? ('yes' or 'no'):")
        read(5,115)wildcde
        if(wildcde.eq."no")go to 600
        call ioa_("1 = Designated wilderness"")
        call ioa_("2 = Administratively endorsed as suitable (prior to 1981)"")
        call ioa_("3 = Further planning or study area (WSA)"")
        call ioa_("4 = BLM wilderness inventory not completed"")
        call ioa_("5 = BLM lands under appeal (WSA's)"")
        call ioa_("6 = USFS rare II under litigation (California)"")
CALL IOA_$NNL("A/Enter the number of the type desired :")
READ(5,115)WILDNUM
600 IF(SKIP2.EQ."NO")GO TO 601
C *****
C ***** prompt for wilderness boundary file name *****
C *****
CALL IOA_$NNL("A/Enter name of file with Wilderness Area coordinates:")
READ(5,120)FILDEGS
C *****
C ***** prompt for MRDS point file name *****
C *****
601 CALL IOA_$NNL("A/Do you have a MRDS point file to be plotted ? ('yes' or 'no'):")
READ(5,115)CRIBANS
IF(CRIBANS.EQ."NO")GO TO 207
CALL IOA_$NNL("A/Would you like the points annotated ? ('yes' or 'no'):")
READ(5,115)ANOTANS
CALL IOA_$NNL("A/Enter name of file with MRDS data points:")
READ(5,120)CRIBFIL
C *****
C ***** prompt for forest file name *****
C *****
207 CALL IOA_$NNL("A/Do you have a file of forest boundaries to be plotted ? ('yes' or 'no'):")
READ(5,115)FORSTANS
IF(FORSTANS.EQ."NO")GO TO 209
CALL IOA_$NNL("A/Enter name of forest boundary file :")
READ(5,115)FORSTFIL
C *****
C ***** prompt for geologic file name *****
C *****
209 CALL IOA_$NNL("A/Do you have a file of geologic data to be plotted ? ('yes' or 'no'):")
READ(5,115)GEOANS
IF(GEOANS.EQ."NO")GO TO 211
CALL IOA_$NNL("A/Enter name of geologic data file :")
READ(5,115)GEOFIL
211 CALL IOA_$NNL("A/Do You Wish To START OVER AGAIN? ('yes' or 'no'):")
READ(5,115)RESTART
IF(RESTART.EQ."YES")GO TO 5
120 format(a8)
    first=1
    endfile="no"
20 if(first.eq.1)go to 136
19 call assoc(15,fildegs,"si ")
21 first = 2
  c ***** read wilderness boundary file *****
  c *****
    read(15,126,end=133)kount,attragcy,attrwldcde
126 format(15,25x,i1,i1)
  c *****
  c ***** fill x & y arrays with single line segment *****
  c *****
134 do 136 n=1,kount,8
    read(15,127,end=133)x(n),y(n),x(n+1),y(n+1),x(n+2),y(n+2),x(n+3),y(n+3),x(n+4),y(n+4),x(n+5),y(n+5),x(n+6),y(n+6),x(n+7)
127 format(16f12.2)
136 continue
  c *****
  c ***** check agency & wilderness type codes *****
  c *****
  if(first.eq.1)go to 280
    if(selcde.eq."no".and.wildcde.eq."yes")go to 131
    if(selcde.eq."no".and.skip2.eq."yes")go to 323
    if(agcynum.eq.attragcy)go to 131
    go to 21
131 if(wildcde.ne."yes".and.agcynum.eq.attragcy)go to 323
    if(wildnum.eq.attrwldcde)go to 323
    go to 21
133 call closer(15)
    endfile = "yes"
    if(first.ne.1.and.cribans.eq."yes")go to 999
    go to 71
280 continue
  c *****
  c ***** DISSPLA calls to establish projection type *****
  c *****
  axis labels and axis length *****
  c *****
  go to 285
call projct(turner)
call title(titl,-100,"longitude",9,"latitude",8,8.0,6.0)
goto 284

284 call projct(turner)

---*** STATE BOUNDARY MAP***---

284 continue

287 call nochek
call grace(0.0)
if(skip3.ne."yes")go to 286

**** DISSPLA calls to set parameters for ****
 **** STATE boundary map ****

xpole=(xmin+xmax)/2.
ypole=(ymin+ymax)/2.
call mapole(xpole,ypole)

286 call mapgr(xmin,xstp,xmax,ymin,ystp,ymax)
if(skip.eq."no")go to 70
call mapfil(cfile)
70 if(skip1.eq."no")go to 73
call mapfil(pfile)
73 call grid(i,l)
    if(maptyp.ne."yes")go to 71
    fName=mapnam
call assoc(i,l,fName,"sqi")

74 n=300

reads & plots STATE boundary map

read(i,l,end=71)n,(xo(ii),ii=1,n),(yo(ii),ii=1,n)
call curve(xo,yo,n,0)
go to 74

888 call closer(i,l)
71 if(skip2.eq."no".and.cribans.eq."yes")go to 999

92 format(2x,a3)
if(first.eq.1)go to 19
if(skip2.eq."yes".and.maptyp.eq."no")go to 21
if(skip2.eq."no")go to 1001
if(first.eq.1)go to 19
322 if(first.eq.2)go to 21
323 call newpen(4)

plot wilderness boundary by line segment

read(i,l,end=71)n,(xo(ii),ii=1,n),(yo(ii),ii=1,n)
call curve(xo,yo,n,0)
go to 74

888 call closer(i,l)
71 if(skip2.eq."no".and.cribans.eq."yes")go to 999

92 format(2x,a3)
if(first.eq.1)go to 19
if(skip2.eq."yes".and.maptyp.eq."no")go to 21
if(skip2.eq."no")go to 1001
if(first.eq.1)go to 19
322 if(first.eq.2)go to 21
323 call newpen(4)
call curve(x,y,kount,0)

verification of input file name

324 if(endfile.ne."yes")go to 21
999 if(cribans.ne."yes")go to 1001
endfile="yes"
c write(6,92)endfile
call marker(3)
go to 335
c 333 call height(.05)
333 call sclpic(.4)
call marker(4)
335 continue
c ******
   **** plot symbol ******
c ******
call curve(xxx, yyy, j-1, -1)
c xxx(l) = xxx(j)
c yyy(l) = yyy(j)

depotype(1) = depotype(j)
if (done.eq.1) go to 1000
go to 334
c ******
c ****** plot legend with labels stored in xlegenray ******
c ******
call legend(xlegenray, 3, 0, .5)
325 continue
   if(anotans.eq."no") go to 1000
c ******
c ****** maxarray limit hard coded ******
c ******
maxarray = 1500
n = 1
344 continue
c ******
c ****** set switch for deposit type ******
c ******
prevtype = depotype(n)
n = n + 1

c ******
c ****** read & plot MRDS information ******
c ****** for annotations ******
c ******
read(18, 326, end=489) xxx(n), yyy(n), depo(n), depotype(n), commod(n), irotanot(n)
if(depotype(n).eq.prevtype and j.lt.maxarray) go to 344
326 format(2f10.3, 1x, a37, 1l, a50, 1x, i1)
go to 346
489 done = 1
346 continue
   if (depotype(n).ne.depotype(n-1)) go to 327
   n = n + 1
327 continue
do 345 j = 1, n
c ******
c ****** select symbol type & height ******
c ******
go to(341, 342, 343) depotype(n-1)
341 call height(.10)
call marker(5)
go to 345
342 call height(.07)
call marker(3)
go to 345
343 call height(.05)
call marker(4)
345 continue
c ******          ******
c ****** plot symbol ******
c ******          ******
call curve(xxx,yyy,n-1,-1)
c xxx(1) = xxx(n)
c yyy(1) = yyy(n)
depotype(1) = depotype(n)
if(done.eq.1)go to 289
continue
if(anotans.ne."yes")go to 1000
call newpen(3)
c ******          ******
c ****** determine angle of rotation ******
c ******          ******
go to (361,362,363,364,365,366),irotanot(k)
361 degrot=90.
go to 366
362 degrot=45.
go to 366
363 degrot=0.
go to 366
364 degrot=-45.
go to 366
365 degrot=-90.
go to 366
366 continue
c ******          ******
c ****** set shift factor for annotation ******
c ******          ******
call angle(degrot)
if (degrot.eq.90.) call mshift(-.05, .05)
if (degrot.eq.45.) call mshift(.025, .09)
if (degrot.eq.0.) call mshift(.05, .05)
if (degrot.eq.-45.) call mshift(.09, -.025)
if (degrot.eq.-90.) call mshift(.05, -.05)
call rlmess(depo(k),37,xxx(k),yyy(k))
call reset("mshift")
if (degrot.eq.90.) call mshift(.05, .05)
if (degrot.eq.45.) call mshift(.09, .03)
if (degrot.eq.0.) call mshift(.05, -.05)
if (degrot.eq.-45.) call mshift(.03, -.09)
if (degrot.eq.-90.) call mshift(-.05, -.05)
call rlmess(commod(k),50,xxx(k),yyy(k))
call reset("mshift")
call reset("angle")
go to 344
c ****** plot legend ******
c ****** ******
c call legend(xlegenray,3,0,.5)
1000 continue
call closer(18)
1001 continue
    if(forstans.eq."no") go to 288
    c ****** ******
c ****** read forest file ******
c ****** ******
call assoc(19,forstfil,"si ")
367 read(19,370,end=1003)type,numbr
370 format(1x,i3,16x,i5)
    c ****** ******
c ****** fill xfor & yfor arrays by line segment ******
c ****** ******
do 1002 l=1,numbr
    read(19,375,end=1003)xfor(l),yfor(l)
375 format(2f!0.5)
1002 continue
call newpen(l)
    c ****** ******
c ****** plot forest boundaries ******
c ****** ******
call curve(xfor,yfor,numbr,0)
go to 367
1003 continue
call closer(19)
288 if(geoans.eq."no") go to 289
    c ****** ******
call assoc(20,geolfile,"si ")
1004 read (20, 291, end=1006)iden,leftid,rightid,npoints
291 format(i5,a5,a5,i5)
    c ****** ******
c ****** fill xgeo & ygeo arrays by line segment ******
c ****** ******
do 1005 m=1,npoints,3
    read (20,294,end=1006)xgeo(m),ygeo(m),xgeo(m+1),ygeo(m+1),xgeo(m+2),ygeo(m+2)
294 format(6f12.9)
    c ****** ******
c ****** convert from radians to degrees ******
c ****** ******
xgeo(m)=xgeo(m)*-57.29578
xgeo(m+1)=xgeo(m+1)*-57.29578
xgeo(m+2)=xgeo(m+2)*-57.29578
ygeo(m)=ygeo(m)*57.29578
ygeo(m+1)=ygeo(m+1)*57.29578
ygeo(m+2)=ygeo(m+2)*57.29578
1005 continue
xgeo(l)=xgeo(2)
ygeo(1)=ygeo(2)
call newpen(1)
c
*****

***** set line type for plotting

*****
if(iden.eq.11)call dash
if(iden.eq.20)call thicrv(2)
if(iden.ne.21)go to 1007
call dash
call thicrv(2)
1007 continue
if(iden.eq.30)go to 1008
call newpen(3)
call thicrv(2)
1008 continue
if(iden.eq.31)go to 1009
call newpen(3)
call dash
call thicrv(2)
1009 continue
if(iden.eq.80)go to 1010
call newpen(4)
call thicrv(2)
1010 continue
if(iden.eq.81)go to 1011
call newpen(4)
call dash
call thicrv(2)
1011 continue

c
*****

***** plot geology features

*****
c
call curve(xgeo,ygeo,npoints,0)
call reset('dash')
call reset('thicrv')
go to 1004
1006 continue
call closer(20)
289 if(cribans.ne."yes")go to 290
call endpl(0)
call height(.35)
call mylegn("mrds occurrences $",100)
call reset("height")
call height(.25)
call lines("porphry deposit$",xlegenray,1)
call lines("lode$",xlegenray,2)
call lines("all else$",xlegenray,3)
call leglin
call reset("height")
if(cribans.eq."no")go to 290
call projct(turner)
call title(titl,-100," ",0," ",0,8.0,6.0)
call mapgr(xmin,xstp,xmax,ymin,ystp,ymax)
do 302  j=1,3
    call assoc(21,"legnsymb","si ")
    read(21,303,end=302)xleg(j),yleg(j),depoleg(j)
 303  format(2f10.3,38x,i1)
302  continue
    do 301  i=1,3
      go to(296,297,298)depoleg(i)
 296  call sclpic(3.)
      c 296  call height(.35)
      call marker(5)
      go to 299
 297  call sclpic(1.)
      c 297  call height(.10)
      call marker(3)
      go to 299
 298  call sclpic(2.)
      c 298  call height(.30)
      call marker(4)
 299  continue
      call curve(xleg,yleg,l,-1)
301  continue
    call linesp(2.0)
    call leglin
    call legend(xlegenray,3,0,3)
 300  call close_file(“-all"
430  continue
    call closer(15)
431  stop
eend
Appendix E

MRDS.FORTRAN

external close_file(descriptors), assoc(descriptors)
character*37 depo
integer depotype
integer irotanot
character*50 commod
character*1 longew
character*1 latns
1 read(15,100,end=99) longdeg, longmin, longsec, longew, latdeg, latmin, latsec, latns, depo, commod
100 format(i4,1x,i2,1x,i2,a1,1x,i2,1x,i2,1x,i2,1x,al,1x,a37,a50)
along=longdeg+longmin/60.+longsec/3600.
alat=latdeg+latmin/60.+latsec/3600.
if(longew.eq."W")along = along * -1
if(latns.eq."S")alat = alat * -1
depotype = 3
irotanot = 3
write(16,200) along, alat, depo, depotype, commod, irotanot
200 format(2f10.3,a37,i1,a50,1x,i1)
go to 1
99 continue
end
Appendix F
- Available Projections -

Cylindrical Projections ***

a) Cylindrical Equidistant (CYLIN) - does not project data but rather displays coordinates as they are. It checks limits for validity and is useful if limits are calculated and there is potential error in specification.

b) Mercator (MERCA) - At all points the scale is the same in all directions but does not express the variance of the separation of lines of latitude and longitude thereby giving the illusion of enlarged areas toward the poles.

c) Exact Cylindrical Equidistant (EXACT) - Similar to (A) except that coordinates are fully corrected for ellipcity of the earth and are scaled at a constant factor based on latitude and longitude at the map center.

d) Corrected Mercator (CORRE) - Similar to (B) except that whereas longitude lines are still separated by a constant factor, the local latitude scale is corrected at all points for ellipticity of the globe.

Elliptical Projections ***

a) Mollweide (MOLLW) An equal area map with parallel lines of constant latitude while the meridians (longitude) appear as ellipses equally spaced at the equator. Distortion is great near the poles.

b) Aitoff (Hammers) Projection (AITOF) - an equal area projection but is not as badly distorted near the poles as the Mollweide. Latitude lines are neither parallel nor straight, equally spaced only at the map center while meridians are equally spaced at the equator.

c) Sanson (Flamsteed) Sinusoidal (SANSO) - an equal area projection with straight, parallel and equally spaced lines of constant latitude while longitude lines are equally spaced at the equator. Distortion is severe near the edges and poles.

d) Simple Elliptical (ELLIP) - resembles Mollweide very closely but is not equal area and has the feature that latitude and longitude obey a simple equation it is thus possible to relate coordinates from the projected data with accuracy without a fine mesh grid.
Conical Projections ***

a) Bi-parallel Conformal Conic (CONFO) - a conformal projection using two reference parallels for which the meridians are straight lines, intersecting at either the north or south pole when extrapolated and parallels appear as segments of concentric circles whose centers are the poles. The separation of parallels and meridians preserves the local scale.

b) Bi-parallel Equal Area Conic (Alberts Equal Area Conic) (ALBER) - Similar to (A) except that the spacing of meridians, radii and circle segments is such as to preserve local area on the globe. A desirable feature is that although local scale is not preserved, the angle is, making it useful for maps of limited area.

c) True Polyconic Projection on an Infinitesimal Graticule (POLYC) - a conformal projection which is an extension of the simple conic projection, however, it is more accurate due to the use of a series of tangent cones as opposed to a single cone.

Azimuthal Projections ***

a) Gnomic Projection (GNOMO) - a projection which is neither conformal nor equal-area with heavy distortion near map corners making useful area limited to area near the map center. All straight lines on the map represent the shortest path between two points on the globe surface.

b) Orthographic Projection (ORTHO) - a true perspective view of the globe which is neither conformal nor equal area. The view is that which would be seen by an astronaut as the projection pole lies at infinity.

c) Stereographic (STERE) - a conformal projection with projection pole on the surface diametrically opposite the map pole. Due to its conformal property it is used for maps of large areas (entire hemispheres).

d) Azimuthal Equi-Distant (AZIMU) - a non-conformal non-equal area projection for which lengths on the map plane correspond to the lengths on the sphere surface. It is often used for polar projections in which the map pole coincides with the north or south pole.

e) Azimuthal (Lambart) Equal-Area (LAMBE) - Similar to (D) except the projection is equal-area but scales are not the same in all directions. It is often used for large maps as the bearings taken from the map poles are true.
Appendix G

- Subroutines Employed -

A. setup_calcomp and setup_tektronix_tcs
   Used to prepare the program to interface with standard CALCOMP and
   Tektronix-supplied software.

B. ioa_$nnl
   Is an entry point to the ioa_subroutine and is used to format the data
   string entered at the terminal and write the resulting string to the
   user output switch.

C. add_search_rules(asr) and delete_search_rules(dsr)
   Used to modify the current "search rules" established by the users
   start_up.ec or those acquired by default as well as to reset them upon
   termination.

D. close_file
   Closes specified Fortran files opened by the program.
Bibliography