UNITED STATES DEPARTMENT OF INTERIOR
GEOLOGICAL SURVEY

GEOPLOT

An Advanced Graphics Package for Minicomputers

Part 1: User's Guide

Peter L. Ward

Open-File Report 83-807 — A

Office of Earthquake Studies
345 Middlefield Road
Menlo Park, Ca. 94025

Table of Contents

Introduction

Geoplot is an advanced graphics package that runs interactively, from command files, or as a subroutine package on 16 bit microcomputers. Geoplot contains many features usually found only in graphics packages on larger computers. A geoplot user can draw lines, symbols, and letters of any pattern, size, thickness, and orientation, using 29 different transformations, including 16 map transformations. Great circle and rhumbline paths are easily drawn. Two sets of world coastline data are available in files containing 5,000 and 80,000 points. Map data can be digitized using an X-Y table-top digitizer. Map coordinates can be rotated and shifted about any arbitrary sequence of poles. Complex plots can be generated simply using high-level routines or the user may access low-level calls. Plots may be stored in disk files and output to a wide variety of vector or raster graphics devices.

Development of Geoplot began in 1978 but was severly curtailed in 1981 for economic and political reasons. The version described in this report consists of all improvements as of September 14, 1983. Several advanced features such as multiple fonts, shading of arbitrary areas, and placing rectangular blank areas within plots, were planned but never completed. Most of the existing code has been tested, but not extensively. This version runs as a child process under the UNIX Operating System on a Digital Equipment Corp PDP-11/70 using an overlay scheme available in the Berkeley 2.9 version of Unix. The overlays are only required for the transformations and the package could be made significantly smaller by omitting some transformations. The full power of geoplot is best utilized through the interactive command language Geolab developed simultaneously by James Herriot.

The author does not intend to develop this package further or to provide anything but minimal support for the existing package. This report, therefore, provides a final report on this development effort as a stepping stone for future interest by others.

The author wishes to thanks Barbara Bekins for carefully reviewing this report.

A Very Preliminary Tutorial for Geoplot

Peter L. Ward

U.S. Geological Survey
Menlo Park, California

1. Introduction

Geoplot is an advanced graphics package that is easily used interactively within Geolab or may be called from compiled languages. Geoplot can also be used from a shell script of geolab commands. Geoplot is designed with three basic principles in mind:

a. The beginning user should be able to create complex plots with very few standard commands while the advanced user should have "hands-on" control of every parameter necessary for customizing plots as well as all basic low-level plot commands.

b. Plotting must be fully device independent but if a device supports hardware lettering, line dashing, etc., the device hardware should be used automatically when appropriate to increase speed. Interactive response time is important.

c. An advanced graphics package should be made available on minicomputers without usurping much of the users limited address space.

Most users needs for plots can be met with a few simple geoplot commands. For example if you have two arrays in Geolab (x and y), they can be plotted on a Tektronix 4014 display with the following commands:

ploton tek plot x y

To plot x versus y as a bar graph in log-log coordinates on a new page (clear screen) type:

5 = trans page 7$plot x y

To plot y as a function of its index beginning at 1 type:

plot 1 y

To put the same plot on the Versatec type:

ploton vers plot 1 y

To plot a map of the world in mercator projection type:

14=trans world frame "world.co" map

Yet below this simple facade, the user can control most options to make lines dashed, vary the width of lines, change the size, font, aspect, and spacing of letters, vary the length of tic marks or the angle of labels on grids, add arrow tips or other symbols to the end of each line, etc.

2. Scope

This tutorial assumes sufficient knowledge of UNIX and Geolab to be able to log on to the machine, enter Geolab, and have a basic knowledge of how Geolab works. Geoplot may be called from compiled languages like Fortran. The

easiest way to learn the features of Geoplot, however, is to use it interactively. Thus this tutorial uses only Geolab calls to Geoplot. All detailed descriptions of Geoplot commands in section GA2 of the manuals show both the Geolab command and the Fortran call. A geolab script and an identical fortran sample program are included at the end of this tutorial. The reader should refer regularly to section GA2 of the manuals for details of commands discussed.

3. Getting Started

Assuming you have logged into your system and have entered Geolab, just type **ploton tek** if you are using a Tektronix 4014 display with enhanced graphics. Otherwise refer to *ploton*(GA2) for other device names. Geoplot creates a file in your home directory called **geoplot.w**. This file remembers the state of geoplot when you used it last time. To reset geoplot to the default options either remove geoplot or add a capitol R to the device name (e.g. **ploton tekR**). You may plot into or from a file using **plotfrom** or **plotinto** (*ploton*(GA2)).

4. The Two Major Plot Operators

The easiest way to plot in geoplot is to use the commands **plot** and **plotlet** which scale the plot, draw and label a frame and plot the data. Lets define two arrays and plot them:

```
real x:10          real y:10
count:10 = x    **  2 = y
plot x y
```

Either one of the arrays may be a scalar. In this case the values for the axis are the counting numbers beginning at the scalar and continuing for the length of the other array. Thus:

```
page  plot 1 y
```

Does the same thing. Page erases the screen. Then:

```
page  plot -5 y
```

does it a little differently.

Many transformations are available (see *trans*(GA2)).

Thus to plot in log-log coordinates:

```
5 = trans  page plot 1 y
```

There are many options for the type of plot to be drawn. See *plt*(GA2) and try:

```
1 = trans  8 do (page i$plot 1 y)
```

Letters and strings may be plotted at each point.

To plot an x centered on each point rotated 20 degrees type

```
page  plotlet 1 y 'x' 20 1
```

See *plotlet*(GA2) for the options. The string, angle, and position values may be arrays or scalars.

5. Plot spaces and page units

A primary concept in geoplot is the idea of page units. The narrowest dimension of a plot device output is considered to be one page unit and the other dimension is measured in these units. The maximum page units of the device you are using in x and y directions is given by typing in Geolab

fullsp is

Fullspace on the tektronix 4014 is 1.31 in the x or horizontal direction and 1.0 in the y or vertical direction. Using page units we may then refer to parts of the plot surface.

Picture space (**pictsp**) is defined as the part of the plot surface into which the whole plot will be put including grid, labeling, and data. Lettering is typically clipped to fit within picture space. Several picture spaces might be defined successively at different parts of the same plot surface.

Grid space (**gridsp**) is the part of the picture space around which the grid or frame will be drawn.

Subject space (**subjsp**) is the part of the grid space within which the data will be plotted. Subject space is typically equal to grid space, but may not be equal when multiple data sets are plotted within the same grid. Data is typically clipped to fit within subject space.

Data space (**datasp**) is the value of the data units to be mapped into subject space.

The value of the plot spaces can be displayed by typing **pspaces**.

We can change picture space, for example, by typing

[0.3 0.9 0.3 0.7] = pictsp pspaces

and we note that grid and subject space also changed. Grid space is linked to picture space through all the default values of the tic lengths, letter sizes, angle of lettering, etc. Label space (**lablsp**) is also used to set the number of letter widths to be saved between picture and grid space when the lettering is outside of gridspace and the letters are written at an angle to the grid line.

You may set gridspace to some other value after setting picture space. Subject space is linked to gridspace but it may also be set separately after gridspace is set.

Picture space may be set up easily into parts of the display area using *partx* and *party*(GA2):

page partx 1 2 party 2 3

Grid space may be divided up using *propor*(GA2).

6. Transformations

A wide variety of coordinate transformations exist in **geoplot** (*trans*(GA2)) and new ones can be added easily.

7. Basic plot ops

The basic operators are:

ploton	initialize a plot device
movdrw	move or draw a line in data, page, or relative units
letter	same as movdrw but put a string at the end of the line
plt	plot arrays
pltltr	plot arrays and strings
scale	scale data space
plot	plot arrays with frame and scaling
plotlet	plot arrays and strings with frame and scaling
map	draw a map
page	erase screen or move to next page
repro	make a copy on the hard copy device if it exists

transform	change this point to or from page or data units
trans	change transformation
axis	draw and label an axis on the edge of gridspace
label	put a title on an axis
grid	draw and label a gridded line in any direction
frame	draw four axes around grid space
fare	pick a point using the cursor
pick	pick points in a plot with multiple data sets

Operators for handling plot spaces are:

partx	subdivide pictsp
party	subdivide pictsp in y direction
pspaces	list plot spaces
world	set datasp for the whole world

Programs for making or modifying map files are:

mapdig	digitize a map
maputil	edit, convert, rotate or translate a map file

8. Gpcom, The Nerve Center

Most of the options on Geoplot may be controlled through *gpcom*(GA2). Try these options to find out how they work.

For example:

```
123456 = lintype
0.002 = linwidth
1.0 = symshape
plot  1  y
```

HAVE FUN!!!

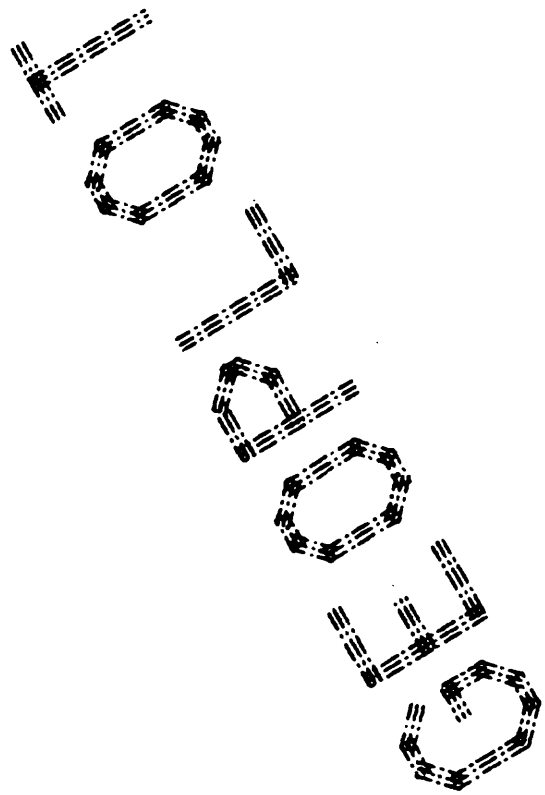
```
% SAMPLE geoplot program using geolab commands

% Initialize variables, etc.
!real x:10 !real y:10 !real angles:10
!char names:10:5 "AB+CD"=names:1 "$"=names:2 "hi+  "=names:3
"abcde"=names:4 "1+2"=names:5 "\A\B"=names:6 "bop"=names:7
"+\b0"=names:8 "\5\_2"=names:9 "<-+ ->"=names:10
count 10 **3=x count 10 **2=y count 10 * 33=angles
!op sleep(e "sleep 10")

ploton tekR
0=gprint  1=letheight 0.1=letheight 0.007=linwidth:2 4=lintype:2
0.005=linspace:2
page 0.5 0.5 1 "GEOPLOT" 30 1 letter
% geolab flushes the plot buffer before each line of input
% thus sleep should be on a new line
sleep

0.0016=linwidth:2 0.0003=linspace:2 0.05=letheight [3 1 1]=lintype
page "This is an example of drawing vectors." is
0 0 1 movdrw 0.3 0.3 2 movdrw 0.1 0.2 4 movdrw
0.7 0.7 2 "AB+CD" 305 1 letter -0.1 -0.4 4 "\A\B" -10 1 letter
sleep

1=letheight:1 0=linwidth:2 1=lintype:1
[0.1 fullsp:1 0.1 (fullsp:2 - 0.2)]=pictsp
page "This is an example of plotting tickmarks for x vs y." is
3$plot x y  label -1 "This is the x-axis" label 3 "This is the y-axis."
sleep

page "This is an example of plotting y vs its index as a bar graph" is
"and x vs its index as triangles about the y axis." is
scale 10 y 7 plt scale x 10 6 plt
sleep

"Now plot x vs y using arrow tipped vectors." is 0.5=symshape
scale x y 1 plt
sleep

page "This is an example of plotting x vs y vs a character array." is
0=symshape x y 1 names angles 1 pltltr
sleep

page "Now plot a map in orthographic azimuthal projection." is
21=trans [ -140 -50 0 65]=datasp frame "world.co" map
% Note space after [ and before minus sign. Geolab gives incorrect
%  results if this space is left out due to parsing problems and
%  confusion in differentiating unary and binary minus.
sleep

3=letheight:1 1=trans page q
```

```
c--*--PLTFOR---SAMPLE PLOT PROGRAM IN FORTRAN USING THE GEOPLOT PACKAGE--
c
      program pltfor
c
c-----Programmer: PLWard,USGS, Menlo Park, California 94025  12/13/79
c
c-----Compile: f77 pltfor.f -lgeop
c          or: f77 -I4 pltfor.f -lgeop4
c
      integer i
      integer length
      integer*2 idelay
      real x(10),y(10),angle(10),space(4)
      character*5 name(10)
c
      include "/usr/include/GPCOM.h"
      real pltcom
      external pltcom
c
      data name /'AB+CD','$\0','hi+  ','abcde','1+2\0','\\A\\B\0',
     1    'bop\0','+\\b0\0','\\5\\_2','<-+->'/
c
      idelay=10
c
      length=10
      do 100 i=1,length
      x(i)=i**3
      y(i)=i**2
100   angle(i)=33.*i
c
c-----Initialize the tektronix 4014 to begin geoplot process
c     The R resets common to default values
      call ploton(1,"tekR",0.0,-8,99999.0)
      call putcom(GPRINT,1,0.0)
c
c-----Set the character size to the largest hardware size.
      call putcom(LETHEI,1,1.0)
c
c---Set the software character size
      call putcom(LETHEI,1,0.1)
c
c-----Set the line width for letters
      call putcom(LINWID+1,1,0.007)
c
c-----Set the line type for letters
      call putcom(LINTYP+1,1,4.0)
c
c-----Set line spacing for letters
      call putcom(LINSPA+1,1,0.005)
c
c-----Erase screen
      call page
c
c-----Print title of geoplot,dump buffer and sleep
      call letter(0.5,0.5,1,'GEOPLOT',30.0,1)
      call clrbuf
      call sleep(idelay)
c
c-----Set the line width and spacing for letters
      call putcom(LINWID+1,1,0.0016)
      call putcom(LINSPA+1,1,0.0003)
c
c-----Set the software character size.
      call putcom(LETHEI,1,0.05)
```

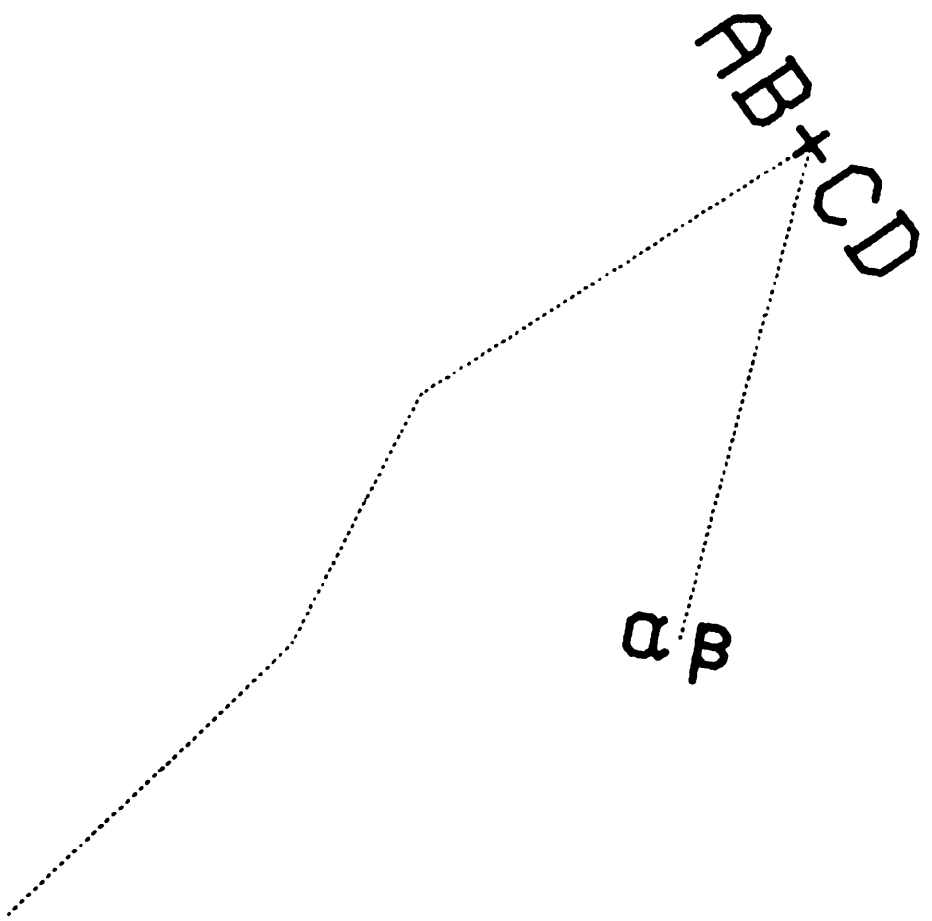
```
c
c-----Set the line type
      call putcom(LINTYP,1,3.0)
      call putcom(LINTYP+1,1,1.0)
c
c-----Erase the screen and title new plot
      call page
      write(6,1)
  1   format("This is an example of drawing vectors.")
c
      call movdrw(0.0,0.0,1)
      call movdrw(0.3,0.3,2)
      call movdrw(0.1,0.2,4)
      call letter(0.7,0.7,2,name(1),305.0,1)
      call letter(-0.1,-0.4,4,name(6),-10.0,1)
      call clrbuf
      call sleep(idelay)
      call page
c
c-----Set software character size.
      call putcom(LETHEI,1,1.0)
      call putcom(LINWID+1,1,0.0)
c
c-----Scale plot spaces
      space(1)=0.1
      space(2)=pltcom(FULLSP)
      space(3)=0.1
      space(4)=pltcom(FULLSP+1)-0.2
      call putcom(PICTSP,4,space)
      call scale(x,length,y,length)
c
c-----Set solid line type
      call putcom(LINTYP,1,1.0)
c
c-----Plot x data versus y data
      write(6,2)
  2   format("This is an example of plotting ticmarks for x vs y.")
      call frame
      call label(-1,"This is the x-axis")
      call label(3,"This is the y-axis")
      call plt(x,length,y,length,3)
      call clrbuf
      call sleep(idelay)
      call page
c
c-----Plot y vs its index plus 10.
987   continue
      write(6,3)
  3   format("This is an example of plotting y vs its index as a bar gra
     1ph\nand x vs its index as triangles about the y axis.")
      call scale(10.0,0,y,length)
      call plt(10.0,0,y,length,7)
      call scale(x,length,10.0,0)
      call putcom(LINWID+1,1,0.0)
      call plt(x,length,10.0,0,6)
      call clrbuf
      call sleep(idelay)
c
c-----Turn on symbols to draw arrows.
      call putcom(SYMSHA,1,0.5)
c
      write(6,4)
  4   format(/,/,/,/,"Now plot x vs y using arrow tipped vectors.")
      call scale(x,length,y,length)
```

```
      call plt(x,length,y,length,1)
      call clrbuf
      call sleep(idelay)
      call page
c
c-----Turn off the arrows
      call putcom(SYMSHA,1,0.0)
c
c-----Use pltltr
      write(6,5)
   5  format("This is an example of plotting x vs y vs a character array
     1.")
      call putcom(LINWID+1,1,0.0)
      call pltltr(x,length,y,length,1,name,5,length,angle,length,1,0)
      call clrbuf
      call sleep(idelay)
      call page
c
c-----Use map
      write(6,6)
   6  format("Now plot a map in orthographic azimuthal projection.")
      call putcom(TRANS,1,21.0)
      space(1)=-140.
      space(2)=-50.
      space(3)=0.
      space(4)=65.
      call putcom(DATASP,4,space)
      call frame
      call map("world.co")
      call clrbuf
      call sleep(idelay)
      call page
c
c-----Set hardware letter size back to next to smallest
      call putcom(LETHEI,1,3.0)
c
c-----Terminate plotting
      call gpfin
      end
```

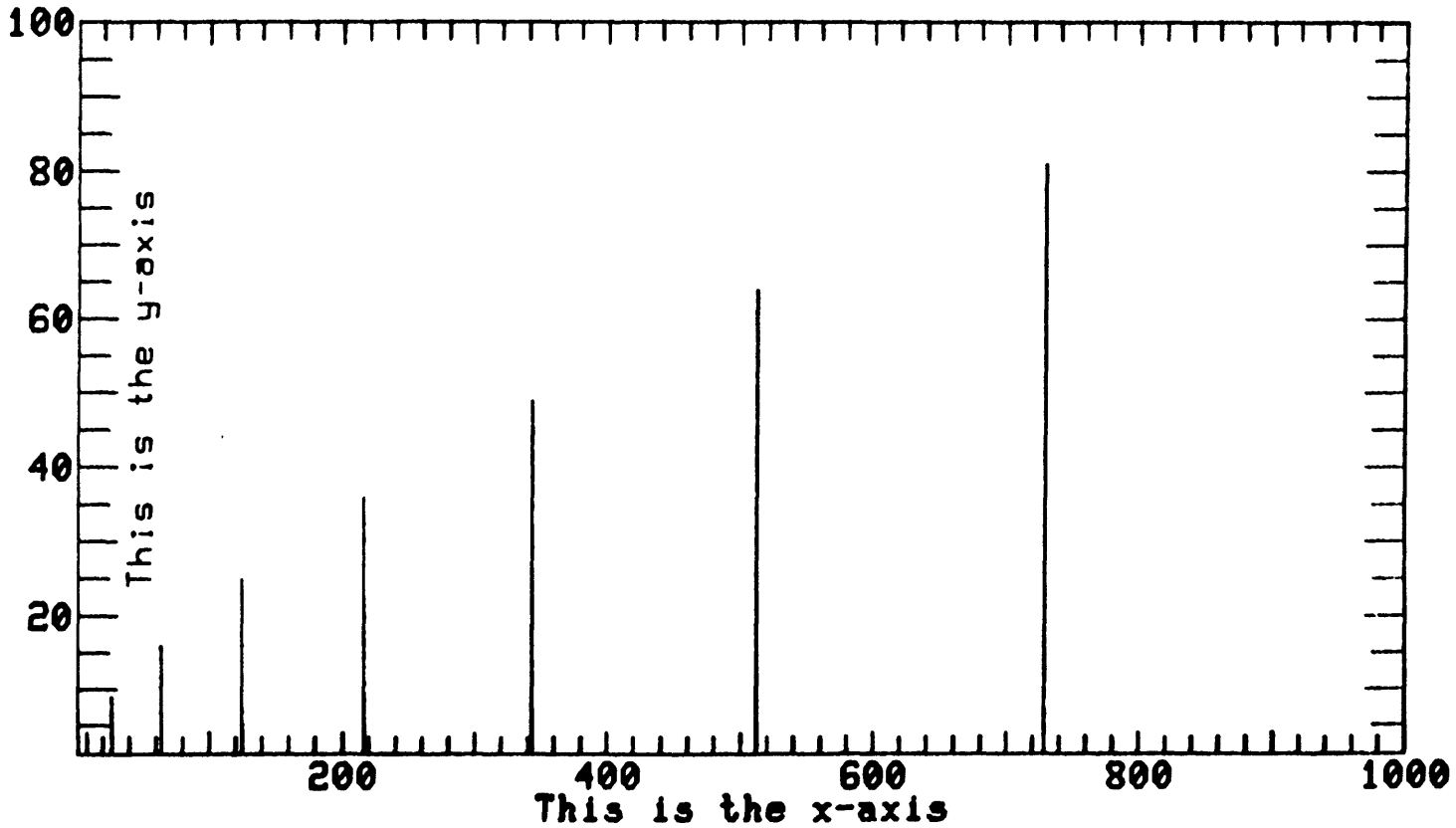
SAMPLE OUTPUT FROM PLTFOR



This is an example of drawing vectors.

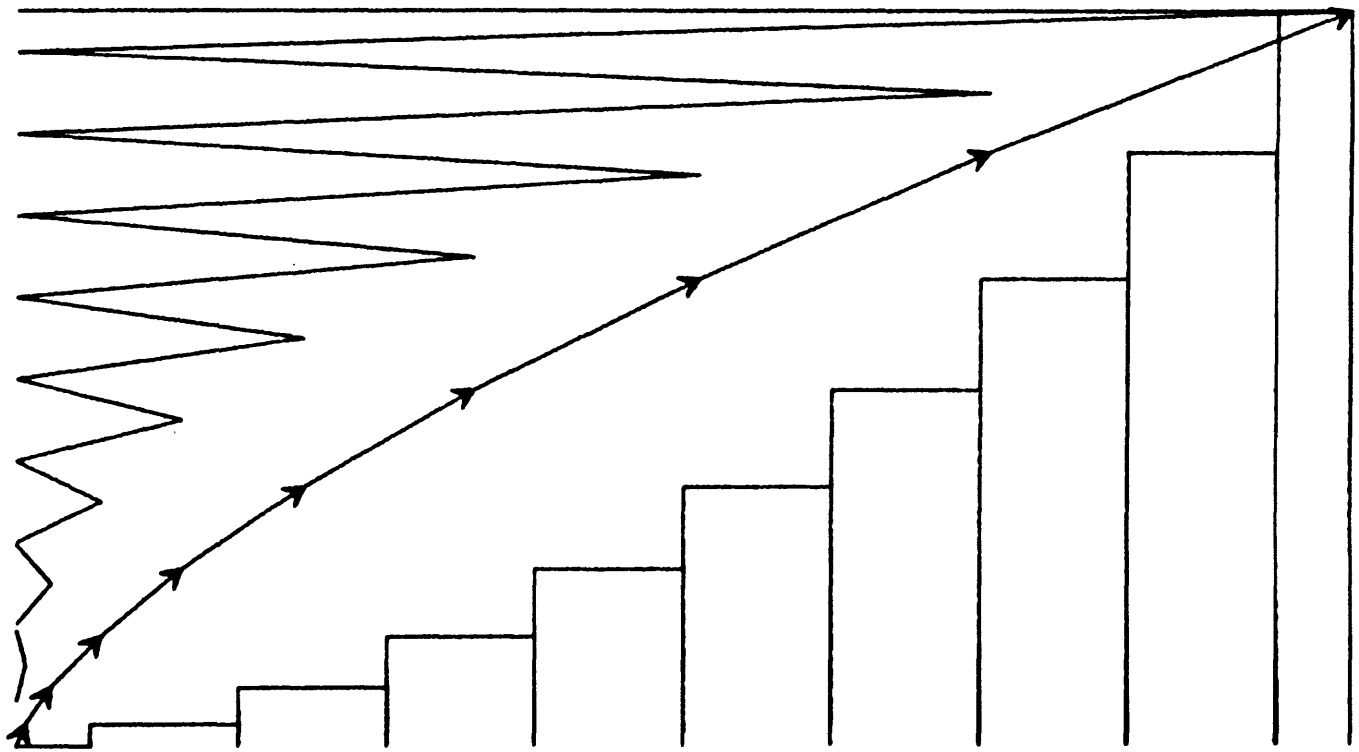


This is an example of plotting ticmarks for x vs y.

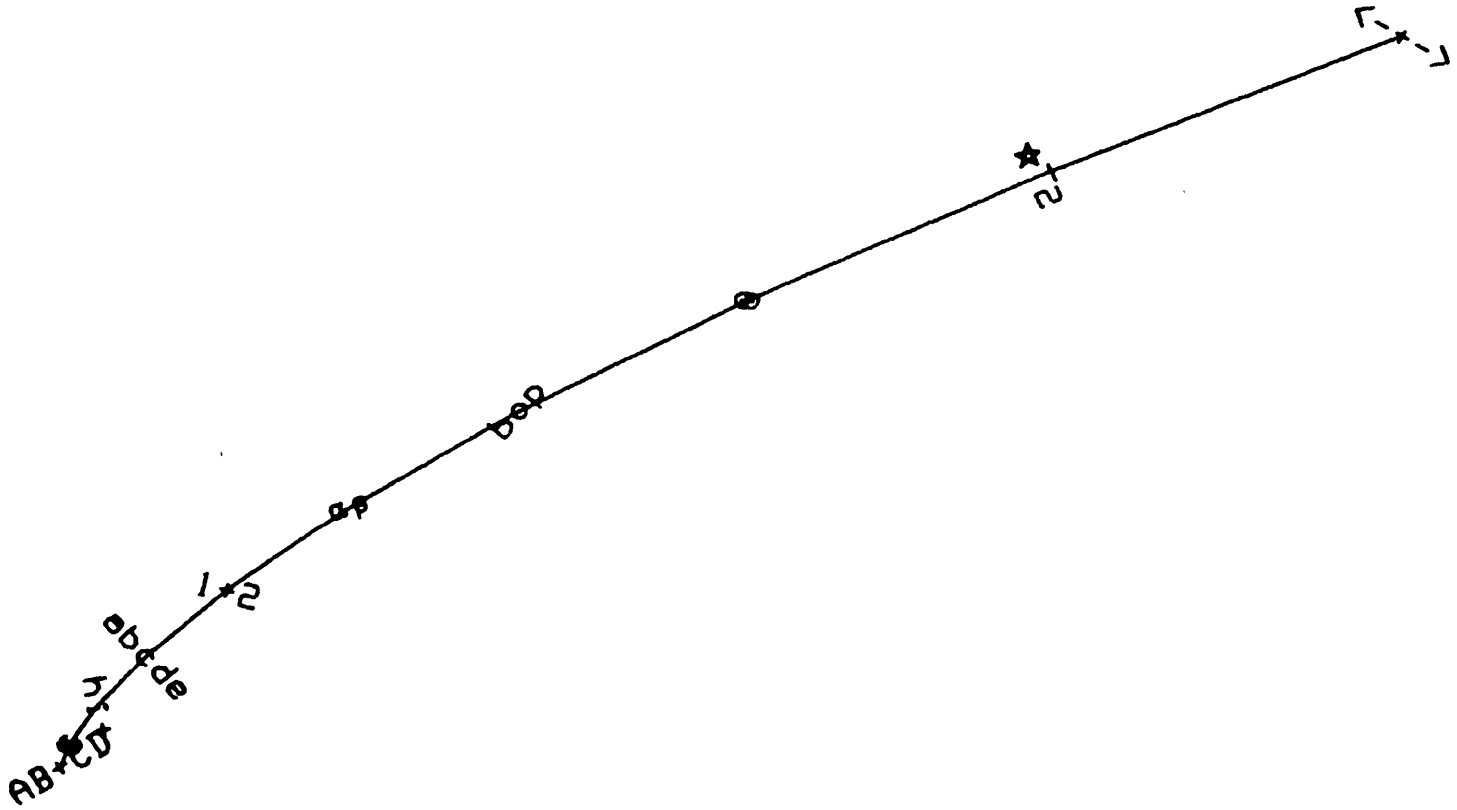


This is an example of plotting y vs its index as a bar graph and x vs its index as triangles about the y axis.

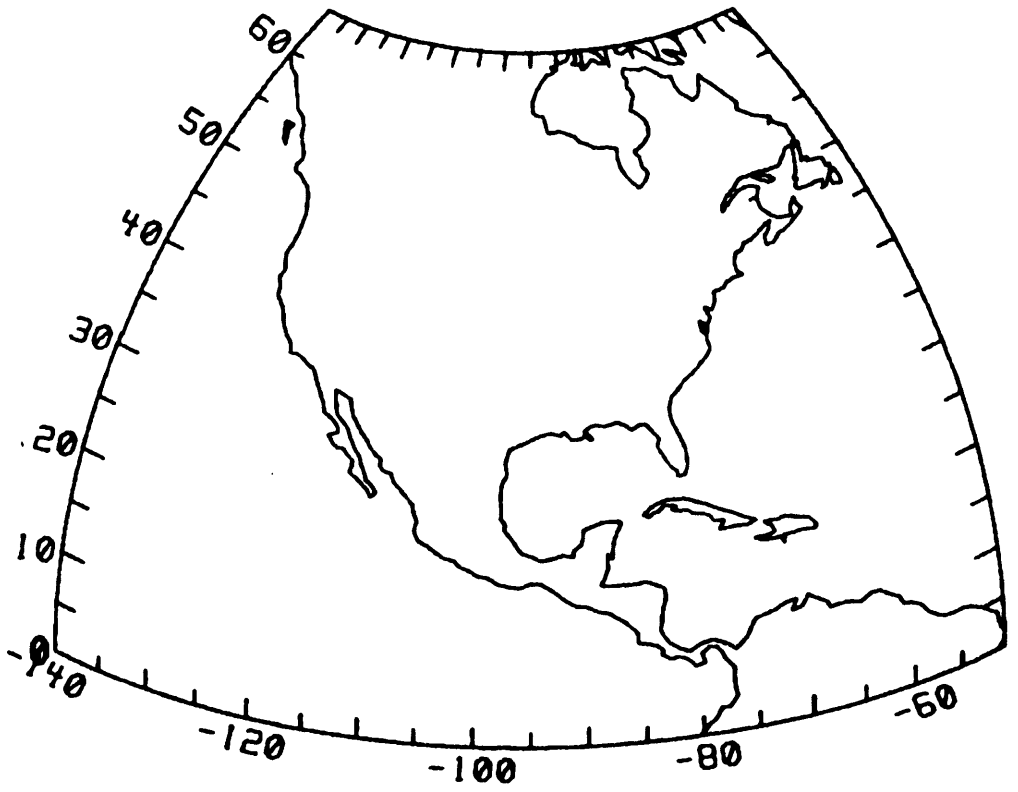
Now plot x vs y using arrow tipped vectors.



This is an example of plotting x vs y vs a character array.



Now plot a map in orthographic azimuthal projection.
Change to orthographic map transformation.



NAME

axis – plot and label an axis

SYNTAX

G: **axis** which

F: **call axis**(which)

DESCRIPTION

axis draws one of the axis along the edge of grid space. *which* may be:

1 lower x axis
2 upper x axis
3 lower y axis
4 upper y axis

The properties of the axis are set by variables in *gpcom*(GA2).

SEE ALSO

gpcom(GA2), *Geoplot Tutorial*

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

disazm – find distance in km, azimuth and back azimuth between two geographic points

SYNTAX

C: lona lata lonb latb type **disazm**

F: call **disazm** (lona, lata, lonb, latb, type)

real lona, lata, lonb, latb
integer type

DESCRIPTION

If type = 1, calculate distance, azimuth, and back azimuth between two points assuming an elliptical earth. Return three values to stack in geolab or as first three arguments in fortran.

If type = 2, same as 1 except assume spherical earth.

If type = 3, input longitude and latitude of a point and distance and azimuth to a second point. Return longitude and latitude of second point to stack in geolab or as first two arguments in fortran. Assume elliptical earth.

If type = 4, same as 3 except assume spherical earth.

Earth values assumed: equatorial radius 6378.163 km, polar radius 6356.177 km, mean earth radius 6371. km.

SEE ALSO

mapscale(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, Californa, based on distance azimuth routines by Bruce Julian, U.S. Geological Survey, Menlo Park, California.

NAME

fare – find crosshair position

SYNTAX

G: [!] **fare**

F: mode = +_7; call **movdrw**(x,y,mode)

real x,y
integer mode

XOP

op fare (0.0 0.0 if 7 –7 movdrw)

DESCRIPTION

Turns on crosshairs on appropriate plot device. A point is picked by pressing any key. The xy coordinates of the point are returned in page units (G: *!fare*; F: *7*) or in data units (G: ***fare***; F: *-7*) and the ascii integer value of the key pushed is returned as mode. In Geolab the values left on the stack are mode, y, x (top to bottom).

BUGS

The plot device must be strapped to send a carriage return after the cursor position code. On a Tektronix there are three options for the termination characters. Put the strap in the center position.

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

frame – draw and label plot grids.

DESCRIPTION

frame is used to draw axes, tic marks, and auto-label axes. It simply calls axis four times.

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

gpcom,getcom,putcom — access elements in geoplot common block

SYNTAX

G: value=(**gpcom** part index n)

F: call **getcom**(index,n,array)
call **putcom**(index,n,array)

integer index,n
real array(n)

XOP

Hard op but equivalent, for example, to:

op pictsp(gpcom part 41 4)

DESCRIPTION

Put in *array* or get from *array* the *n* values in geoplot common beginning at element index. Geoplot is set up with many defaults that make it easy to use. An experienced user, however, can control through *gpcom* a wide variety of options as described below.

OPTIONS

Geoplot common contains the following variables. In Geolab each variable may be called by the name given in boldfaced type and treated as an array or scalar as appropriate. The column **n** designates the dimension of the variable. From Geolab if a scalar is set equal to an n-dimensional array, all values of the array are set equal to the value of the scalar.

		index	n	default
Plot spaces (low x, hi x, low y, hi y)				
pictsp	picture space in page units	21	4	device dependent
gridsp	grid space in page units	27	4	device dependent
subjsp	subject space in page units	33	4	device dependent
datasp	data space in data units	39	4	device dependent
lablsp	label space, number of character widths or heights between picture and grid space	191	4	[7 2 7 2]
omitsp1 **omitsp2** **omitsp3** **omitsp4**	areas of plot to be omitted	45	16	0.0
fullsp	user adjustable maximum plot width. Must be < = devmax	161	2	
pspaces	display all plot spaces	272	1	
clip	clipping factor used in propor	163	3	[1.0 1.0 1.0]
Transformations				
trans	transformation type	190	1	1
transcon	constants used in map transformations	61	9	usym
curinc	page units between interpolated points on curves	90	1	0.01

pole	pole of projection for map transformations	97	2	usym
mapscale	scale of map projection (see *mapscale*(GA2))	133	1	usym
Line Properties (where n=3, elements 1,2,3 refer to lines,letters,symbols respectively)				
lintype	0 point 1 line 2 line of points 3-11 hardware dash patterns >11 software dash patterns made up of a concatenation of: 1 Draw 0.005 page units 2 Move 0.005 page units 3 Draw 0.010 page units 4 Move 0.010 page units 5 Draw 0.020 page units 6 Move 0.020 page units 7 Draw 0.040 page units 8 Move 0.040 page units (e.g. 1476 means draw 0.005, move 0.010, draw 0.040, move 0.020, draw 0.005 etc.)	181	3	[1 1 1]
linwidth	line width in page units	84	3	rasmin
linspace	spacing in page units between parallel lines used to make width	81	3	rasmin
linside	side of line stippling drawn 1=both, 2=left, 3=right	184	3	[1 1 1]
linshade	line intensity or shade or color	87	3	device dependent
linclip	clip space for lines: 1=pictsp 2=gridsp, 3=subjsp, 4=datasp	188	1	3
cliplines	=1 draw lines on the boundary when data goes out of space and returns =2 do not draw lines	189	1	1
Lettering properties				
letheight	height of letters in page units. Integer values of 1 and above input hardware letter sizes for a given device with 1 being the largest size. A tek 4014, for example, has 4 sizes. The input value is converted to page units.	91	1	0.01053
leterror	error allowed between *letheight* and hardware lettering size to still use hardware lettering	96		0.0002
letclip	clipspace for lettering: 1=pictsp 2=gridsp, 3=subjsp, 4=datasp	187	1	1
letaspect	ratio of letter width to height	92	1	0.6
letangle	angle upper part of letters rotated from lower part	93	1	0.0

letspace	width of spaces between letters as a proportion of the letter width	94	1	0.725
letspacev	height of spaces between lines as a proportion of the letter height	95	1	0.61
up	up position on the plot	206	1	2

Symbol properties

symshape	symbol type: under 2.=arrows, over 2.0= n sided polygons	76	1	0.0
symlength	length of symbols in page units	77	1	0.02
symspace	spacing between concentric symbols in page units. If negative lines connecting symbols are eliminated	78	1	0.001
symangle	angle of symbol to line	79	1	31.57404
symfactor	factor multiplied times vector length to get symbol length	80	1	0.0

Frame, Axis, and Grid properties

ticxin	Internal tic lengths as proportion of gridsp for x axis for different levels of tics.	107	5	[0.3 0.024 0.018 0.012 0.006]
ticyin	Internal tic lengths as proportion of gridsp for y axis for different levels of tics.	112	5	[0.03 0.24 0.018 0.012 0.006]
ticxout	External tic lengths as proportion of gridsp.	117	5	0.0
ticyout	for x and y axis for different levels of tics.	122	5	0.0
ticfac	Factor tic lengths multiplied by for given axis.	101	4	[1.0 1.0 1.0 1.0]
ticmin	Minimum tic spacing in page units for given axis.	127	4	[0.02 0.02 0.02 0.02]
labmin	Minimum label spacing in page units for given axis.	142	4	[0.04 0.04 0.03 0.03]
labdist	Distance of label to axis in page units. If 0, do not plot label. Negative means outside the plot.	136	4	[-1.0 0 -1.0 0]
labfactor	amount to be factored out of the label	148	4	1.0
labangle	angle of label with the axis	154	4	[0.0 0.0 -90.0 -90.0]
maxlevel	maximum number of tic levels. Must be <=5	216	4	5.0
minlabel	minimum number of characters before labels are drawn in E format	222	4	8.0
datatype	type of data. =0 normal, =1 date, =2 degrees and minutes =3 degrees, minutes, and seconds for each axis (not implemented)	228	4	0.0
noaxis	If =1 do not draw axis line	207	4	0.0

ticpage	If =1 for tics to be drawn in page units only	213	1	0.0

Plot positioning and scaling on device

poffset	x and y offset of origin in page units	70	2	
pangle	angle in degrees plot to be rotated in degrees	72	1	
pgain	factor by which to multiply plot	73	2	[1.0 1.0]

Beam positioning and output characters

beamtype	for devices where plotting is done intermixed with input commands from same device, after plotting: 1 restore beam to next input line 2 restore beam to top of the page 3 leave beam where it is 4 do nothing -1 to -4 same as above but output is integers rather than plot characters	197	1	device dependent

Miscellaneous constants maintained from Geolab

gpusym	missing data symbol	160	1	usym
julian	if 1, dates are julian	203	1	0.0
gpzone	time zone for dates	204	1	-8.0
gprint	prints intermediate output values from 0 to 10 where 0 means no extra output and 10 means a lot of extra output	198	1	0.0
timebase	base time for date variables	180	1	0.0
ticbase	baseline for ticmarks & bar graphs (plt or pltltr type 3 to 8) on x or y axis as appropriate in data units	75	1	0.0
lineid	line identification number and in pick	234	1	-9999
numfields	number of fields stored for use by *pick*(GA2)	235	1	0.0
picktype	type of pick to use. See *pick*(GA2)	237	1	2.0

READ ONLY VALUES IN GEOPLOT COMMON

pagemax	maximum page units in x and y directions.	1	2	device dependent
devunit	device units per page unit in x and y directions	3	2	
home	home for cursor in x and y page units	5	2	
inchscreen	inches per page unit on display	7	1	
inchrepro	inches per page unit on hard copy device.	8	1	
rasmin	page units per minimum device unit i.e. raster width in page units.	9	1	
nchar	number hardware character types	264	1	

kchar	present hardware character type	265	1	
numdev	device number	253	3	[1 1 0]
pointp	present pen position in software	16	2	
pfilename	Display name of present plot file	271	1	

BUGS

omitsp not working yet

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

gpfin – flush buffers and terminate geoplot process

SYNTAX

C: **gpfin**

F: call **gpfin**

DESCRIPTION

Flush plot buffers and terminate the geoplot child process.

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California.

NAME

grid – draw and label a gridded line in any direction

SYNTAX

C: x y mode string datalo datahi transformation **grid**

F: call **grid** (x,y,mode,string,datalo,datahi,transf)

```
real x,y,datalo,datahi
integer mode, tranof
character*? string
```

DESCRIPTION

grid draws a gridded line from the present pen position to x y where mode may be ± 1,2,3,4 as defined in ***movdrw***(GA2). The line is always straight, even in curvilinear transformations. The tics on the line are scaled from ***datalo*** at the beginning to ***datahi*** at the end. Transformations may be:

1 linear
2 log base 10
3 natural log

The string is placed in the center of the grid. Options for tic length, lettering position, etc are the same as in **axis** (***gpcom***(GA2)) and the values for ticmin, labmin, etc are those for the low x axis, i.e. ticmin:1, labmin:1, etc.

BUGS

grid is not fully debugged.

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

label – label the x or y axis

SYNTAX

G: **label** position string

F: call **label** (position, string)

integer position
character*? string

DESCRIPTION

Prints string centered along x or y axis on inside if position is positive and outside if position is negative. Position is:

1 lower x axis
2 upper x axis
3 lower y axis
4 upper y axis

SEE ALSO

axis(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

letter – plot string of characters

SYNTAX

G: **x y mode string angle position letter**

F: call **letter** (x,y,mode,string,angle,position)

real x,y,angle
integer mode,position,slength
character*slength string
slength is passed by Fortran automatically.
From C add it as a long passed by value as the last argument.

DESCRIPTION

Plot the *string* at an *angle* (degrees counterclockwise from the x axis) positioned according to *position* about point x,y.

OPTIONS

Mode may be positive for page coordinates and negative for data coordinates:

1 Move to x,y
2 Draw a line to x,y
3 Move to point which is at x,y relative to present point
4 Draw to point which is at x,y relative to present point

Lines drawn by *letter* in modes 2 and 4 have all the features available for lines drawn by *movdrw* modes 2. and 4

Position may be:

1 Center string around point
2 Center string around right-most character
3 Center string around left most character
4 Center string to right of the point
5 Center string to left of the point
6 Center string above the point
7 Center string below the point
8 Put string above and to right of point

If position 4 to 7 are negative, one half character width will be left between point and closest part of string.

FONTS may be selected by including in the string **!n** where **n** is 0 thru 9. Font 0 is the default for each string. Once a font is selected it is kept within that string until changed. An exclamation point may be included in the string by using **!!**.

SPECIAL CHARACTERS can be selected by including in the string a backslash followed by one of the following letters:

Positioning:

d subscript (following characters are half size)
u superscript (following characters are half size)
b backspace
r carriage return
l line feed
v reverse line feed
n new line (carriage return and line feed)

Symbols:

1 arrow
2 diamond
3 triangle
4 square
5 star

Mathematical symbols:

= not equal
~ approximately equal
S subset
C cents
— overline
I integral
R square root
S sum (capital sigma)
i infinity

Greek letters (lower case):

A alpha
B beta
D delta
E epsilon
F phi
G gamma
L lambda
M mu
N eta
O omega
P pi
T theta
Y psi

SEE ALSO

movdrw(GA2) mode 16

BUGS

Only one font is implemented. Selection of any font gives font 0.

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

map – plot map data

SYNTAX

G: file **map**

F: call **map** (file)

character*? file

DESCRIPTION

Map plots data from the designated file. Map first looks in the current working directory for the file and then in **/usr/maps**. Map files are binary lists of geographic coordinates in degrees given as pairs of floating point numbers (longitude or x, latitude or y). North and east are positive, south and west are negative. If the latitude is ≥ 990.0, the coordinate is interpreted as a command to move the plotting pen to the next coordinate.

Large map files may be broken up so that all points with 10 degree dbands of latitude occur in one file. If the map file name given ends in a **.m**, map will look for files **filename.m.n** where n is the integer latitude of the lower edge of the band (-90 ≤ n = 80) and search only those files needed as defined by the users datasp (see *gpcom*(GA2)).

FILES

/usr/maps/world.co	world coastlines in 5200 points
/usr/maps/world.co.m.*	world coastlines in 80,700 points

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

mapdig – digitize graphs and maps

SYNTAX

mapdig [-cn num -e -o file -i -n -s yscale -t trans]

DESCRIPTION

mapdig takes x-y coordinates from a tabletop digitizer and converts them to orthogonal x-y page units using the inverse transformations given in *trans*(GA2). It is set up for maps but can be used for any charts. The program normally is used interactively and asks for input:

1. westmost longitude, eastmost longitude, southmost latitude, northmost latitude
2. digitize four corners in order listed by program

Program now calculates several scale factors and angles of rotation and prints an error message if these differ by less than 0.001 in scale or 0.0175 radians in angle. Normally if you get an error message, start over again.

OPTIONS

cn the next argument is the value of **transcon**(n). See *trans*(GA2), *gpcom*(GA2).

e also output coordinates on standard error. This allows you to pipe standard out to a file and still see coordinates on terminal.

o next argument is name of output file. If none is given, data are put in file named *mapdig.out*

i set to non-interactive mode

n do not put data in *mapdig.out* or any output file

s next argument is a factor to multiply each y coordinate by when read in from digitizer. This is useful for digitizers with different scale factors in x and y.

t next argument is transformation number (see *trans*(GA2). 14 or mercator is the default.

All these options may be upper or lower case.

Digitizing input options:

(Digitizer input is expected as numbers such as +12345+23456 representing coordinates in thousandths of inches and is thus divided by 1000. Numbers may be up to 5 digits long and separated by +, -, 0 space. They may include a decimal point and the input stream may include card numbers which are ignored but are assumed to be a / followed by three digits.) Other input characters interpreted are:

c repeat first coordinate since last move. Useful when digitizing closed forms such as an island to be sure coastline closes.

d delete last point digitized

e or *q* exit, quit

f close current output file and open a new file given by the following name

m move. Put out move command in map file.

All other characters will be ignored and an error message given.

USING THE DIGITIZER

The tabletop digitizer at 275 Middlefield Road, Menlo Park, Ca., building 8 room 8246 along the left wall can be easily used in an interactive mode.

1. Turn on the ADM terminal over the digitizing table. The switch is under the rear-left edge.
2. Turn on the digitizer. Switch is near wall on top of lower black box to left of the digitizing table.
3. Be sure the digitizer is connected to the terminal and not the card punch. On rear of the top black box, the switch nearest the center of the room should be in the up position.
4. Connect terminal to UNIX. Tan and cream box on top of digitizer electronics should have switch turned to "UNIX".
5. Log in to UNIX.
6. Type the mapdig command.
7. Connect digitizer to UNIX. Turn switch at left of terminal keyboard to "DIGITIZER". Now output from the digitizer and the digitizer keyboard go to UNIX and the terminal keyboard is disconnected. To use terminal keyboard, turn this switch back to "TERMINAL".
8. Zero digitizer by placing cursor in bracket on left side of table, putting slot around pin and rotating slightly clockwise. Press the leftmost button.
9. To digitize slide cursor to desired point and push the second button from the left. Terminal will beep after the program has received the data for a point, processed the data, and is ready for another point. You can digitize several points ahead of the computer. If you move the cursor outside of the rectangular area defined by the four black dots on the table or if you pick the cursor up, you must re-zero as in number 8 above.
10. When done, be sure to leave the terminal-digitizer switch on "TERMINAL" and shut off the power you turned on in items 1 and 2 above.

SEE ALSO

gpcom(GA2), map(GA2), maputil(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

mapscale – find and set scale of a map

SYNTAX

G: **mapscale** is or value = **mapscale**

F: call **getcom** (133, 1, value)

real value

call putcom (133, 1, value)

DESCRIPTION

When **mapscale** is requested by *getcom*, it is calculated based on the current transformation in effect. The distance (see *disazm*(GA2)) is calculated along the line of mid latitude (transcon:2) from one side of the map (westmost longitude) to the other side (eastmost longitude). This distance is then divided by the physical distance on whatever device is currently selected using inchscreen (see *gpcom*(GA2) to convert page units to inches. If the tektronix 4014 is selected, for example, the scale will then be true for the screen but not for the hard copy unit. Multiply the scale by inchscreen/inchrepro to get scale on some hard copy devices. However, there are three different types of hardcopy devices in use and you may need to calculate a fudge factor.

When a value is assigned to **mapscale** the plot spaces are adjusted to set the scale. If the resulting plot will fit within the currently assigned gridspace, gridspace and subject space are reduced to set the correct scale. If the resulting plot will not fit within gridspace then dataspace is reduced until the scale is correct. These adjustments are done iteratively and are only done immediately after **mapscale** is set. Anytime thereafter that the plotspaces or transformation is changed, the scale will not be reset until specifically requested.

Use **mapscale** with care!

SEE ALSO

gpcom(GA2), trans(GA2), map(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

maputil – utility program for handling map files

SYNTAX

maputil [-ai -ao -c confile -e -f 10 -F -l 1000 -m -o outfile -p lon lat -x num -y num] files

DESCRIPTION

Map files are binary lists of geographic coordinates in degrees given as pairs of floating point numbers (longitude or x, and latitude or y). If the latitude is ≥ 990.0, the coordinate is interpreted as a command to move the plotting pen to the next coordinate. Map files are often kept in **/usr/maps**.

maputil provides a means to convert ascii files to mapfiles, convert mapfiles to ascii, edit the binary or ascii files, and rotate the coordinates around a list of arbitrary poles to, for example, drift continents.

OPTIONS

ai ascii input files

ao ascii output files

c next argument is name of a control file that lists in ascii format groups of four numbers representing the longitude and latitude (in degrees) of a pole of rotation, and the scalar amounts to add to longitude and latitude respectively after rotation. Up to 20 poles may be given. Each coordinate is sequentially rotated about each pole in the list in order and then output.

e edit map files before output. Do not redirect standard if you use this option since you must use editor interactively. If no output file is specified, last input file read is overwritten by output.

f next argument is number of first map coordinate to be read in from file

F if output file exists, force overwriting it

l next argument is number of last map coordinate to be read in from file.

m find minimum and maximum values of longitude and latitude and print them on standard output.

o next argument is name of output file

p next two arguments are longitude and latitude of a pole of rotation in degrees

r next four arguments are region data restricted to: low lon, high lon, low lat, high lat. All data outside of this range will be ignored. A move command is inserted each time the data first goes out of the permitted range.

x scalar amount in degrees to add to longitude after rotating coordinate

y scalar amount in degrees to add to latitude after rotating coordinate

SEE ALSO

map(GA2), mapdig(GA2), gpcom(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

movdrw – primary plot subroutine

SYNTAX

G: **x**,y mode **movdrw**

F: call **movdrw**(x,y,mode)

real x,y integer mode

DESCRIPTION

movdrw is the primary subroutine in geoplot.

OPTIONS

Mode is as follows:
If mode is positive **x** and y are in page space.
If mode is negative x and y are in data space.

1 Move to the point x,y.
2 Draw a vector to the point x,y.
3 Move to a new point where x,y are relative to present point.
4 Draw a vector to a new point where x,y are relative to present point.
5 Transform x, y to data space
-5 Transform x, y to page space
-9 Draw a great circle to new point (for map projection)
-10 Draw a rhumbline to new point (for map projection)

SEE ALSO

gpcom(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

page – erase screen or move to new page.

SYNTAX

G: **page**

F: call **page**

XOP

op page(0.0 0.0 13 movdrw)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

partx, party – divide plot space into parts

SYNTAX

G: **partx** part total
party part total

F: call **part**(axis,part,total)

DESCRIPTION

partx and ***party*** divide the total available plot space on the device into parts. Total plot space is defined by **fullsp** (*gpcom*(GA2)) which is equal to **devmax** (*gpcom*(GA2)) except for those devices where **devmax** is greater than 2. If you wish to have **fullsp** set larger or small, do so before calling ***partx*** or ***party***.

total is the total number of parts and ***part*** is which part you wish to choose numbered from left to right or top to bottom.

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

pick,setpick,resetpick – find cross hair position from many superimposed plots

SYNTAX

G: [!] **pick**
resetpick
setpick lineid

F: call **pick**(x,y,letter,lineid)
call **putcom**(235, 1, 0.0) (is the same as *resetpick*)
call **putcom**(234, 1, lineid) setpick (is the same as *setpick*)
real x,y
integer letter, lineid

XOP

op pick(f if lineid 0.0 900.0 7 movdrw lineid)
op resetpick (0 = numfields, 0 = ksubj,)
op setpick (^ = lineid, 1 = ksubj,)

DESCRIPTION

pick is used to pick data points off of a plot device when several different sets of data are plotted at the same time. Each set of data may have its own data space, subject space, and transformation type. A table of these values is stored at the time the data are plotted along with the first point plotted by any routine except axis, grid, or frame after subject space has changed. *pick* turns on the crosshairs. After adjusting the crosshairs the operator strokes one key, thus picking a point. To determine which line the pick is on the table is searched for the shortest distance in page units between the first point and the crosshair. This distance is measured according to how **picktype** (*gpcom*(GA2)) is set.

1 x distance only
2 y distance only
3 distance in x and y
4 same as 3 except now that the line is identified, the crosshair is turned on again to make the actual pick

If '!' is used in Geolab or the x value of *movdrw* in Fortran is not equal to 0.0, then the integer value of x is considered to be line identifier (**lineid**).

Values returned to the stack then are from top to bottom.

lineid
letter
Y in data units
X in data units

To use *pick* you must first use *resetpick* before any lines are plotted on the device. *page*(GA2) and *ploton*(GA2) also reset pick. Then before each line is plotted, *setpick* must be used to associate a line identifier **lineid** in *gpcom*(GA2) with the line.

SEE ALSO

gpcom(lineid,numfields,picktype)(GA2)
fare(GA2)

BUGS

Table of plot spaces arbitrarily limited to 32.

EXAMPLE

```
resetpick
10 do(propor i 10  -1.0  1.0  setpick i  !!plot 1 Y)
pick
```

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

plot – plot arrays including scaling and frame

SYNTAX

G: [!!][type$] **plot** xarray yarray

F: call **scale** (see options under *scale*(GA2))
call **frame**
call **plt** (see options under *plt*(GA2))

XOP

op plot(f<2 ifthen scale ~~ f ifelse frame g p)

DESCRIPTION

plot is the main operator for making x-y graphs. When preceded by two exclamation points it behaves just like *plt*(GA2) except the x and y values follow *plot*. When proceded by one exclamation point the plot is scaled using *scale*(GA2). By itself plot x y means scale the plot, draw a frame, and then plot the x y values.

Values of type preceding *$* may be from 1 to 8 as described in *plt*(GA2).

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

plot10 – imitation Tektronix plot10 graphics package

SYNTAX

f77 program.f -lplot10

DESCRIPTION

Contains simulations for most of the routines in the Tektronix *plot10* graphics package. Calls to Geoplot routines are used to perform the desired tasks. Names of entry points and argument lists are the same as documented in the **Tektronix Plot10** manual except as noted below. We recommend that *plot10* be used only to run existing programs. Geoplot should be used directly for new programs because it is much more efficient.

BUGS

The following routines are not implemented. However, the package does contain entry points to satisfy the loader:

svstat	rsttab	toutst	ainstr
restat	tabhor	toutpt	setbuf
recovr	tabver	tinstr	seebuf
ttblsz	setmrg	tinput	leftio
settab	tcslev	a1in	

In addition the values of the following arguments in the named routines are ignored or not returned:

ibaud in initt
ispeed in seetrm
iscale in term
mode in seemod
rsuprs in poltrn

EXAMPLE

```
c-----------------------------------
c  Draws a line from raster coordinates
c  (0,0) to (100,100) on the screen
c-----------------------------------
      call term (3, idum)
      call initt (idum)
      call movabs (0,0)
      call drwrel (100,100)
      call finitt (0,0)
      end
```

SEE ALSO

A Very Preliminary Tutorial for Geoplot by Peter L. Ward
TEKTRONIX PLOT10 Terminal Control System Users Manual, 1976, Tektronix Inc.

AUTHOR

Barbara A. Bekins, Electronic Data Systems, Corp., for U.S. Geological Survey, Menlo Park, California

NAME

plotlet – plot arrays with character strings including scaling and frame

SYNTAX

G: [!!][type$] **plotlet** x array yarray string angle position

F: call **scale** (see options under *scale*(GA2))
call **frame**
call **pltltr** (see options under *scale*(GA2))

XOP

op plotlet (f < 2 ifthen scale ~ ~ f ifelse frame g ~ ~ ~ pltltr)

DESCRIPTION

plotlet is the main operator for making x y graphs with character strings. When preceded by two exclamation points it behaves just like *pltltr*(GA2) except the x, y string, angle and position values follow *plotlet*. When preceded by one exclamation point the plot is scaled using *scale*(GA2). By itself, plot x y means scale the plot, draw a frame, and then plot x y values with the character string or strings at each x y point.

Values of type preceding *$* may be from 0 to 8 as described in *pltltr*(GA2).

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

ploton,plotinto,plotfrom – select plotting devices and files

SYNTAX

G: **ploton** device
plotinto "path"
plotfrom "path"

F: call **ploton**(mode,string,timebase,timezone,usym)
integer mode,timezone,
character*? string
real timebase, usym

XOP

op **ploton** (lit fa 1$ plotdev)
op **plotinto** (^ fa 2$plotdev)
op **plotfrom** (^ fa 3$plotdev)

DESCRIPTION

ploton, *plotinto*, or *plotfrom* begin the geoplot child process, if not already begun, and initialize which device the plot will be output on or which file name a plot will be put in or plotted from. *ploton* is followed by one string of letters and numbers that designate the name and model of the plot device. This string does not need to be enclosed in quotes. *plotinto* and *plotfrom* must be followed by the pathname of a file given as a string enclosed in quotes or by an expression that resolves to a string. In Fortran *mode* 1 is *ploton*, *mode* 2 is *plotinto*, and *mode* 3 is *plotfrom*. When *mode* 1 is used the timebase, timezone, and missing data symbol (*usym*(GA1)) must be specified (good default values are 0.0, 0, 99999.0).

OPTIONS

Devices presently supported by geoplot are:

Manufacturer	Model	Numdev(*gpcom*(GA2)) numdev:1	numdev:2
Tektronix	4014 Enhanced graphics	1	1
	4014	1	2
	4012	1	3
	4010	1	4
	4011	1	5
	4025	1	6
Calcomp	1051	2	1
	1012	2	1
Houston Instruments		3	1
Anderson-Jacobson	832	4	1
Hewlett Packard	7221a	5	1
Versatec	1200a	21	1
Printronix	600	22	1
Florida Data	bny78	23	1

4011 is Retrographics emulation of a Tektronix 4010. Any unique set of letters and numbers may be used to specify a *device* after *ploton*. If an unrecognizable manufacturer or model is specified for *device*, then defaults are **silently** used. The default manufacturer is Tektronix and the default

model is the first one listed for each manufacturer. If *plotinto* or *plotfrom* is given and no device has previously been specified at any time since a new plot_com was created, the default device is chosen.

Some capital letters included at the end of a device name have special meanings:

R Reset common to default values.

B Turn off beam positioning. Useful when running plots from a shellfile.

T Test mode. Plot vectors will be printed as integers rather than sent to the plot device.

L On raster devices the plot is normally positioned so that the x axis is parallel to the hardware raster line. Thus a plot typically fills one page of fanfold paper. To rotate the plot so that the x axis may go down the page or perpendicular to the hardware raster line, follow the model by L for long.

Q Raster plots of large data arrays moving down the page (options L) can be done very quickly by designating Q after the model number. The restriction is that these may be only 132 lines or traces. The plot vectors are put into a file. The file is then scanned to be sure that the x values typically increase, and decrease no more than 132 times. These lines or traces are then multiplexed and output directly to the raster device. Lettering must thus be used very sparingly and the axis grid, and frame routines should not be used.

BUGS

Only Tektronix 4014 enhanced and Versatec available at this time.

If you execute device for a Tektronix plot in Geolab and continue typing before the Geolab prompt returns, you may lose your input and sometimes may be an error message. The reason is that **geoplot** asks the Tektronix where its beam position is and may confuse your input with the input sent by the Tektronix.

EXAMPLES

ploton	t	use Tektronix 4014 with enhanced graphics
ploton	tektronix	same
ploton	tek25R	use Tektronix 4025, reset common
ploton	versL	use Versatec 1200a, long option
ploton	t0T	use Tektronix 4010, test option

AUTHOR

Peter L. Ward, U.S.Geological Survey, Menlo Park, California

NAME

plots – imitation Versatec graphics package

SYNTAX

f77 program.f -lvtec

DESCRIPTION

Contains simulations for most of the routines in the Versatec graphics package. Calls to geoplot routines are used to perform the desired tasks. Names of entry points and argument lists are the same as documented in the Versatec Plot Manual. We recommend that *plots* be used only to run existing programs. Geoplot should be used directly for new programs because it is much more efficient. Subroutines implemented are axis, cursor, factor, line, newpen, nplots, number, plot, plots, scale, and symbol.

SEE ALSO

A Very Preliminary Tutorial for Geoplot by Peter L. Ward

AUTHOR

Dave Oppenheimer, U.S. Geological Survey, Menlo Park, California

NAME

plt – plot arrays

SYNTAX

G: xarray yarray type **plt**

F: call **plt**(xarray,xlength,yarray,ylength,type)

real xarray(xlength),yarray(ylength)
integer xlength,ylength,type

OPTIONS

plt plots x vs y from the input data arrays. The data is normally in data coordinates and **datasp** must be set equal the the min and max values of the arrays (see *scale*(GA2)). The data is then plotted in subject space.

Type may be:

0 plot ends of lines only
1 plot lines
2 plot segmented lines (alternate draw and move)
3 plot tic marks along the x axis
4 plot tic marks along the y axis
5 plot triangles along the x axis
6 plot triangles along the y axis
7 plot bar graph along the x axis
8 plot bar graph along the y axis

Tic marks, triangles, and bar graphs are plotted relative to **ticbase**(see *getcom*(GA2) which is normally set to 0.0.

The arrays are normally in absolute data coordinates. Other options may be selected by making:

type = type + 100*itype

where itype is:

0 absolute coordinates in data space
1 relative coordinates in data space
2 absolute coordinates in page space
3 relative coordinates in page space

In Geolab the ***xarray*** or ***yarray*** may be a scalar. In this case the values of that "array" then become the counting numbers beginning with the value of the scalar.

In Fortran ***xarray*** or ***yarray*** may be a scalar if the appropriate length is set <= 1. If ***xlength*** and ***ylength*** are both > 1, then the number of points plotted will be whichever length is least.

SEE ALSO

scale(GA2), ticbase in gpcom(GA2), pltltr(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

pltltr – plot arrays with character strings

SYNTAX

G: xarray yarray type string angle position **pltltr**

F: call **pltltr**(xarray,xlength,yarray,ylength,type,string,
slength,swidth,angle,alength,position,plength)

real xarray(xlength),yarray(ylength),angle(alength)
integer xlength,ylength,type,alength,position(plength)
integer plength,slength,swidth
character*swidth string(slength)

OPTIONS

pltltr plots **x** vs **y** from the input data arrays. The data is normally in data coordinates and ***datasp*** should be set equal the min and max values of the arrays (see *scale*(GA2)). The data is then plotted in subject space.

At each point a string is drawn at a given angle and position relative to the point as in ***letter***(GA2). The string, angle, and position may each independently be the same at each point (***alength***, ***plength***, or ***slength*** <=1) or may be arrays of the same length as the x and y arrays (e.g. ***alength***, ***plength***, or ***slength*** =***xlength***). If the lengths vary, then the number of points plotted will be the minimum of all lengths > 1.

Position may be:

1 Center string around point
2 Center string around right-most character
3 Center string around left most character
4 Center string to right of the point
5 Center string to left of the point
6 Center string above the point
7 Center string below the point
8 Put string above and to right of point

If position 4 to 7 are negative, one half character width will be left between point and closest part of string.

Type may be:

0 plot no lines between points
1 plot lines
2 plot segmented lines (alternate draw and move)
3 plot tic marks along the x axis
4 plot tic marks along the y axis
5 plot triangles along the x axis
6 plot triangles along the y axis
7 plot bar graph along the x axis
8 plot bar graph along the y axis

Tic marks, triangles, and bar graphs are plotted relative to **ticbase**(see *gpcom*(GA2) which is normally set to 0.0.

The arrays are normally in absolute data coordinates. Other options may be selected by making:

type = type + 100*itype

where itype is:

0 absolute coordinates in data space
1 relative coordinates in data space
2 absolute coordinates in page space
3 relative coordinates in page space

In Geolab ***xarray*** or ***yarray*** may be a scalar. In this case the values of that "array" then become the counting numbers beginning with the value of the scalar.

In Fortran if any length is <= 1 the associated value is considered to be a scalar. The number of points plotted is the minimum value of all lengths > 1.

SEE ALSO

scale(GA2), ticbase in gpcom(GA2), plt(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

proporx,propory – proportion grid space in x or y direction

SYNTAX

G: **proporx** center total lodata hidata
propory center total lodata hidata

F: call **propor**(center,total,lodata,hidata,axis)
real center, total,lodata,hidata
integer axis

XOP

op proporx (~ ~ ~ ~ 1 propor_)
op propory (~ ~ ~ ~ 2 propor_)

DESCRIPTION

proporx and *propory* set subject space equal to a part of grid space and scale dataspace accordingly. *total* is the number of equal parts that the x or y direction will be divided into. In Fortran axis = 1 for x direction, = 2 for y direction. The width of one part is the width of grid space divided by total.

center designates which part is to be used from left to right or top to bottom. If *center* is less than 1.0, it represents the center point of the part in normalized units where the width of grid space is 1 and 0 is at the bottom of grid space. *center* may not be larger than total. *datalo* and *datahi* are the data units at the lower and upper boundaries of the part chosen.

The data plotted is normally clipped at the boundary of subject space. The global variable **clip** (*gpcom*(GA2)) may be used, however, to change the clipping values. **clip** is used in *propor* for setting subject and data space. The effect of **clip** is that while the data will be scaled the same, data and subject space will be set so that the effective width of the plot for clipping purposes is the width of subject space times **clip**. The default value of **clip** is 1.0. A value of 2.0 would allow the plots in each part to be twice the width of the part, provided they fall within grid space.

EXAMPLES

To plot the same sin wave in array y versus its subscript four times in horizontal strips one below the next, type in Geolab:

```
4 do (propory i 4 -1  1 !!plot 1 y)
```

To plot a seismic record section, to make each seismic trace fill 1/4 of the grid space in the y direction, and where the distances to the stations range from distlo to disthi, then for each trace do:

```
!max trace =y
propory 1 - ((disthi - distance)/(disthi - distlo)) 4
        -y y !!plot 1 trace
```

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

pspaces – print values of plotting spaces

SYNTAX

G: **pspaces**

F: call **getcom**(272,1,0.0)

XOP

op pspaces(gpcom:272 ,)

DESCRIPTION

List on the terminal the values of pictsp, gridsp, subjsp, datasp, lablsp, and omitsp.

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

repro – make a print of a plot on the hard copy device

SYNTAX

G: **repro**

F: call **repro**

XOP

op repro(0.0 0.0 14 movdrw)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

scale – autoscaling for x vs y plots

SYNTAX

G: **scale** xarray yarray

F: call **scale**(xarray,xlength,yarray,ylength)
real xarray(xlength,yarray(ylength) integer xlength,ylength

XOP

op scale=([] 2 do (^ - > h len_ = lenspa:i, xchg | [(!min h) (!max h)]) = datasp, 2 do (lenspa:i <=1 ifthen (lenspa:(3-i)+ datasp: (2*i-1)-1 = datasp:(2*i),)))

DESCRIPTION

scale sets datasp for the min and max values of ***xarray*** and ***yarray***. In Geolab if either ***xarray*** or ***yarray*** is a scalar or in Fortran if either ***xlength*** or ***ylength*** is <=1, then the corresponding axis is scaled from the value of the scalar to the value of the scalar plus the length of the other array minus 1.

SEE ALSO

plt(GA2),mapscale(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

shade – shade in an area of a plot

DESCRIPTION

Not implemented yet.

NAME

trans – set transformation type for plotting

SYNTAX

G: n = **trans**

F: call **putcom**(190,1,n)

real n

DESCRIPTION

Whenever plotting is done in data space, *gptran* is called to transform data to page space. Inverse transformations are used when picking points off of the plot device. All inverse transformations are done by the same iterative approximation technique for efficiency in coding.

OPTIONS

1 x - linear, y - linear
2 x - linear, y - log
3 x - linear, y - ln
4 x - log, y - linear
5 x - log, y - log
6 x - log, y - ln
7 x - ln, y - linear
8 x - ln, y - log
9 x - ln, y - ln
10 polar: x - degrees, y - distance, counterclockwise from east
11 polar: x - degrees, y - distance, clockwise from north
12 polar: x - degrees, y - dip, clockwise from north, equal angle, Wulff net
13 polar: x - degrees, y - dip, clockwise from north, equal area, Schmidt net
14 Mercator Cylindrical
15 Miller Cylindrical
16 Transverse Mercator Cylindrial
17 Universal Transverse Mercator
18 Equal Areal Azimuthal
19 Equal Distance Azimuthal
20 Gnomonicphic Azimuthal
21 Orthographic Azimuthal
22 Perspective Azimuthal
23 Stereographic Azimuthal
24 Lambert Conformal Conic
25 Ptolemy Equal Interval Conic
26 Kavraiskiy IV Equal Interval Conic
27 Albers Equal Area Conic
28 Polyconic
29 Sinuisoidal
30 Find lon and lat of point around new pole (see pole in *gpcom*(GA2))

MAP PROJECTIONS

It is impossible to project the surface of a spheroid onto a plane without distortion. Therefore, a wide variety of map projections have been devised to minimize the distortion for particular cases. Maps are typically projected onto an idealized planar, cylindrical, or conical surface that is either tangent to the earth's surface or secant to the earth's surface (i.e., cutting into and then back out of the earth). A map projection is said to be conformal if all angles and shapes within a small area are true. Equidistant means all distances in any direction from a given point are in proportion. Equal-area means all areas are in proportion. Azimuthal means that all azimuths from the center of the map are true and that straight lines through the center are great circles.

Map projections available in geoplot are as follows:

1: **Equirectangular** where lines of latitude and longitude are spaced equally. Ideal for maps of small areas such as for a city or local area. Easy to hand plot data on.

14: **Mercator** conformal cylindrical where the spacing between lines of longitude increases with latitude. A straight line is a compass course or rhumbline and thus this projection is used widely for navigation. There is serious exaggeration of distances and areas at latitudes greater than about 40°. Cannot be used at the poles. Program limits the absolute value of latitude to less than 89.9°.

15: **Miller** cylindrical is a compromise between the area compression of the equirectangular plot and the extreme scale expansion of the Mercator plot. Most useful for maps of very large areas (scale > 1:1,000,000).

16: **Transverse** conformal **mercator** cylindrical is ideal for a 15 to 20 degree band centered around a central meridian (line of longitude). Straight lines are not rhumblines as in the mercator projection. Often used for topographic maps.

17: **Universal tranverse mercator** is a worldwide grid system in meters based on the transverse mercator projection. Central meridians are constructed every 6° from 80°S. Zones extending 3° to either side of the central meridian are overlaid by a uniform rectangular grid. The central meridian whether input or calculated is rounded to the nearest standard meridian (3°, 9°, 15° etc.). If you use this transformation to calculate grid coordinates add 500,000 to the longitude and 10,000,000 to the latitude. Be careful about precision, while all calculations are done in double precision, input and output through geoplot is limited to single precision (7 + decimal places). Caution, the UTM grid uses 10 different values for the major and minor axis of the earth depending on location. These are not handled automatically in this program and are not important except for high precision in very small areas.

18: **Azimuthal** equal area invented by Lambert is best suited for maps of continental areas or hemispheres.

19: **Azimuthal equal-distance** is ideal for radio and seismic work since all distances and azimuths are accurate from the center of the map.

20: **Gnomonic** has the unique property that any straight line is a great circle or the shortest distance between two points. It is best suited for areas where the longitude differs by less than 40° from the central meridian.

21: **Orthographic** azimuthal is a perspective projection where the viewer is a very long way away from the earth. It is most useful for representing hemispheres centered on any point.

22: **Perspective** azimuthal is useful for viewing slightly less than a hemisphere from a spaceship at any arbitrary elevation above the point on earth that is the center of the map. The projection is often used for orientation. Elevation of viewer is given as a proportion of earth's radius. Thus 1.0 means 1 earth radius above earth surface.

23: **Stereographic** conformal azimuthal is the only projection on which any circle on the earth is shown as a circle on the map. It is useful for showing ranges from various points and is used heavily in polar areas.

24: **Lambert** conformal conic uses two standard parallels and adjusts such that at every latitude the north-south and east-west scales are equal. It is used for large and small scale mapping and is especially accurate at mid latitudes. Great circle lines are approximately straight.

25: **Ptolemy** equal interval conic is a compromise projection that uses one standard parallel and is easy to construct and plot on manually.

26: **Kavraiskiy IV** equal interval conic uses two standard parallels to give the least mean square distortion for the Soviet Union. Excellent for display of areas with large longitudinal extent.

27: **Albers** equal-area conic uses two standard parallels and minimizes scale error. Ideal for maps with large longitudinal extent.

28: **Polyconic** is a compromise projection where each parallel is a standard parallel causing the map to be approximately equivalent and conformal only in small areas bisected by the central meridan. Used by USGS for topographic maps.

29: **Sinusoidal** or Mercator equal area reduces the distortion of cylindrical equal area maps and is ideal for maps with significant latitudinal extent. Convenient for manual plotting of data.

Two lines with special properties often used with maps are the great circle, which is the shortest distance between two points, and the rhumbline, which is a line that intersects all meridians at a constant azimuth. The great circle is a straight line on the gnomonic azimuthal projection whereas the rhumbline is a straight line on a mercator cylindrical projection. These lines can be drawn using **movdrw**(UA2) with mode of -9 for great circle and -10 for rhumbline.

Map transformations use constants that may be changed using **gpcom**(UA2) element transcon. The items used by each projection are:

		TRANSFORMATION																
	Default value	1	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29
1 Mid-longitude or central meridian	(max long + min lon)/2		x	x	x	x	x	x	x	x		x	x	x	x	x	x	x
2 Mid-latitude	(max lat + min lat)/2						x	x	x	x		x		x				
3 Lower standard parallel	1/6 way up map												x		x	x		
4 Upper standard parallel	5/6 way up map												x		x	x		
5 Major axis of earth	6378388.0		x			x						x	x			x	x	
6 Minor axis of earth	6356911.9462		x			x						x	x			x	x	
7 1 for north polar projection, -1 for south polar projection	1.0												x					
8 Nature of elevation of observer above earth to radius of earth	1.0										x							
9 Earth's radius in nautical miles	3437.9768										x			x	x			

REFERENCES

CAM, Cartographic Automatic Mapping Program Documentation, 5th edition, Central Intelligence Agency GC77-10126, 121 p., June 1977. Source of most equations for map projections used in **gptran**. Gives sample maps. Caution some equations have errors!

Alpha, Tau Rho, and John P. Snyder, *The properties and uses of selected map projections*, U.S. Geological Survey Misc. Investigations Series MAP I-1402, 1982. Table and maps showing different projections.

Birdsey, C. H., *Formulas and Tables for the Construction of Polyconic Projections*, United States Geological Survey Bulletin 809, Washington, D.C., 126 p., 1929.

Deetz, C. H., and Adams, O.S., *Elements of Map Projections with Applications to Map and Chart Construction*, United States Coast and Geodetic Survey Special Publication 68, Washington, D.C., 1944.

Maling, D. H., FIA Review of Some Russian Map Projections, Empire Survey Review 15, pp. 203-215, 255-266, 294-303, 1960.

Maling, D. H., *Coordinate Systems and Map Projections*, George Philip and son, London, 255 p., 1973.

Richardus, Peter, and Ron K. Adler, *Map Projections*, North-Holland Publishing Co., 174 p., 1972.

Robinson, A. H. and Sale, R. D., *Elements of Cartography*, 3rd Edition, John Wiley & Sons, New York, 415 p., 1969.

Thomas, P. D., *Conformal Projections in Geodesy and Cartography*, Coast and Geodetic Survey special Publication 251, Washington, D. C., 142 p., 1968.

Tobler, W. R., *A Classification of Map Projections*, Association of American Geographers Annals, Vol. 52, No. 2, pp. 167-175, June 1962.

SEE ALSO

map(GA2), mapdig(GA2), gpcom(GA2)

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

transform – transform data to page space or visa versa.

SYNTAX

G: x, y [!] **transform**

F: call **movdrw**(x, y, ±5)

XOP

op transform(f if 5 -5 movdrw,)

OPTIONS

transform(-5): transform x, y to page units and leave new values on the stack, y on top, x next down.

!transform; transform x, y to data units.

AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

NAME

world – set map scale to full world

SYNTAX

G: **world**

XOP

op world ([-180 180 -89 89] = datasp,)

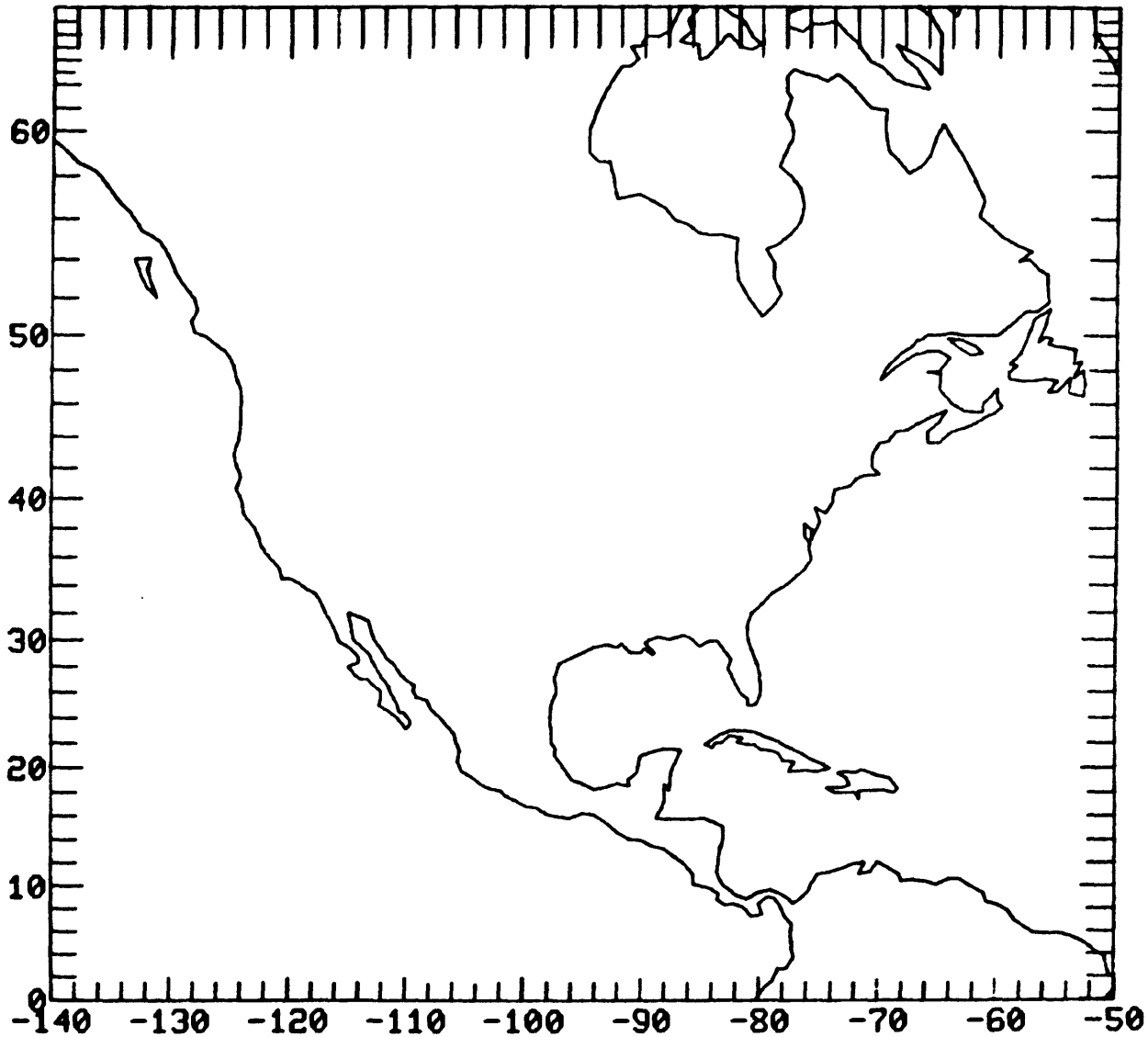
DESCRIPTION

world is shorthand to set dataspace for plotting maps to full scale. 89 is used for latitude to avoid problems with mercator projection.

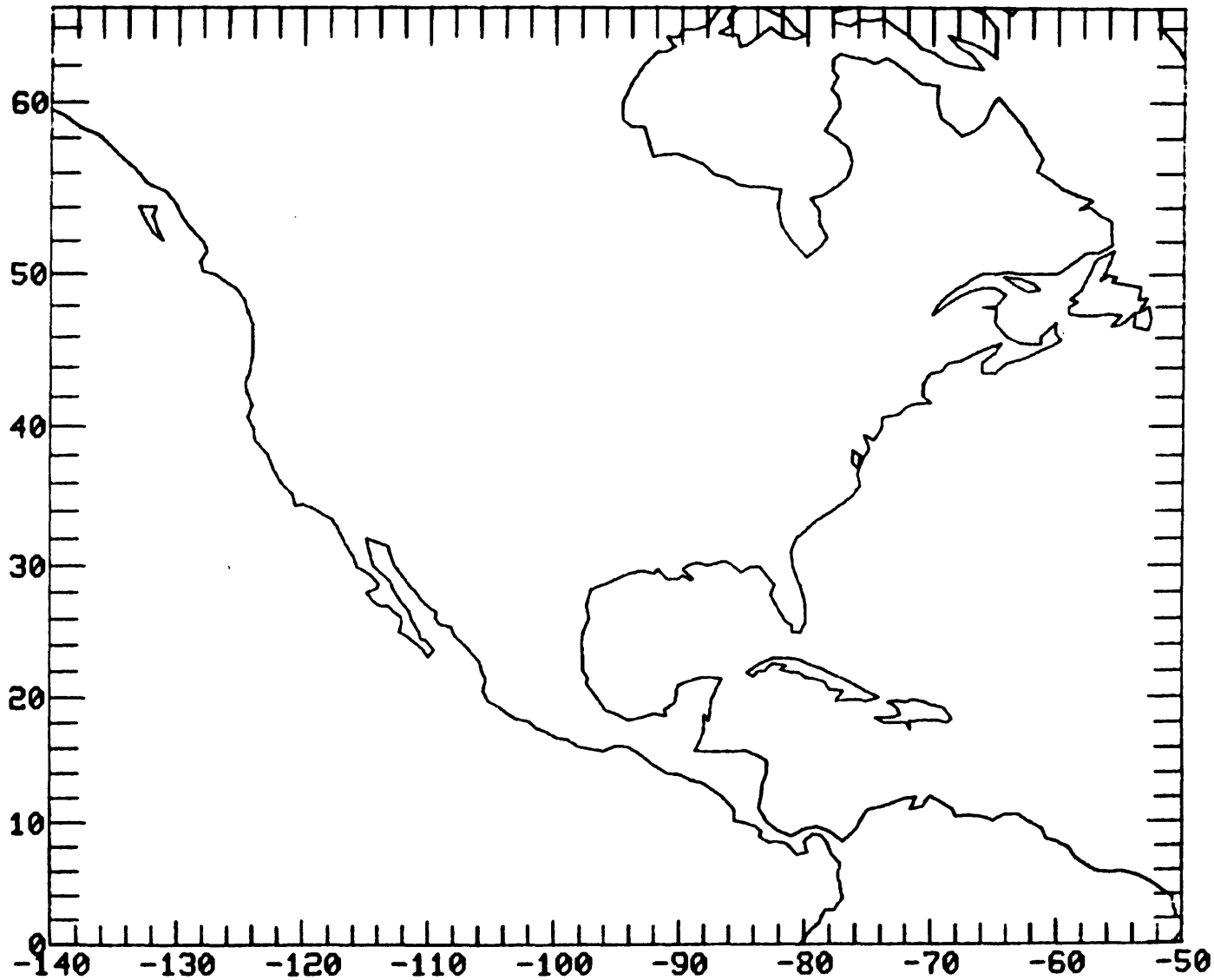
AUTHOR

Peter L. Ward, U.S. Geological Survey, Menlo Park, California

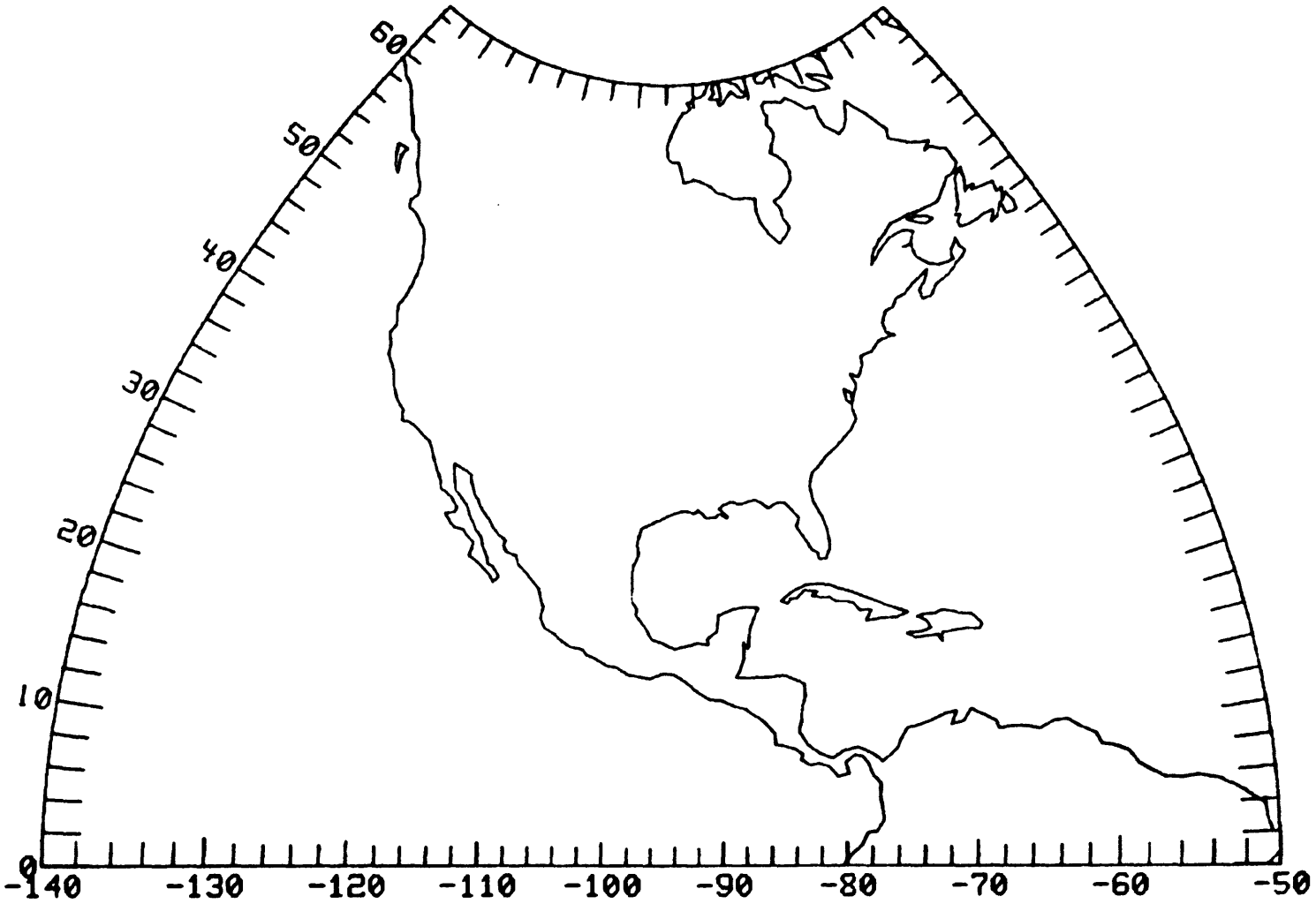
Change to mercator map transformation.



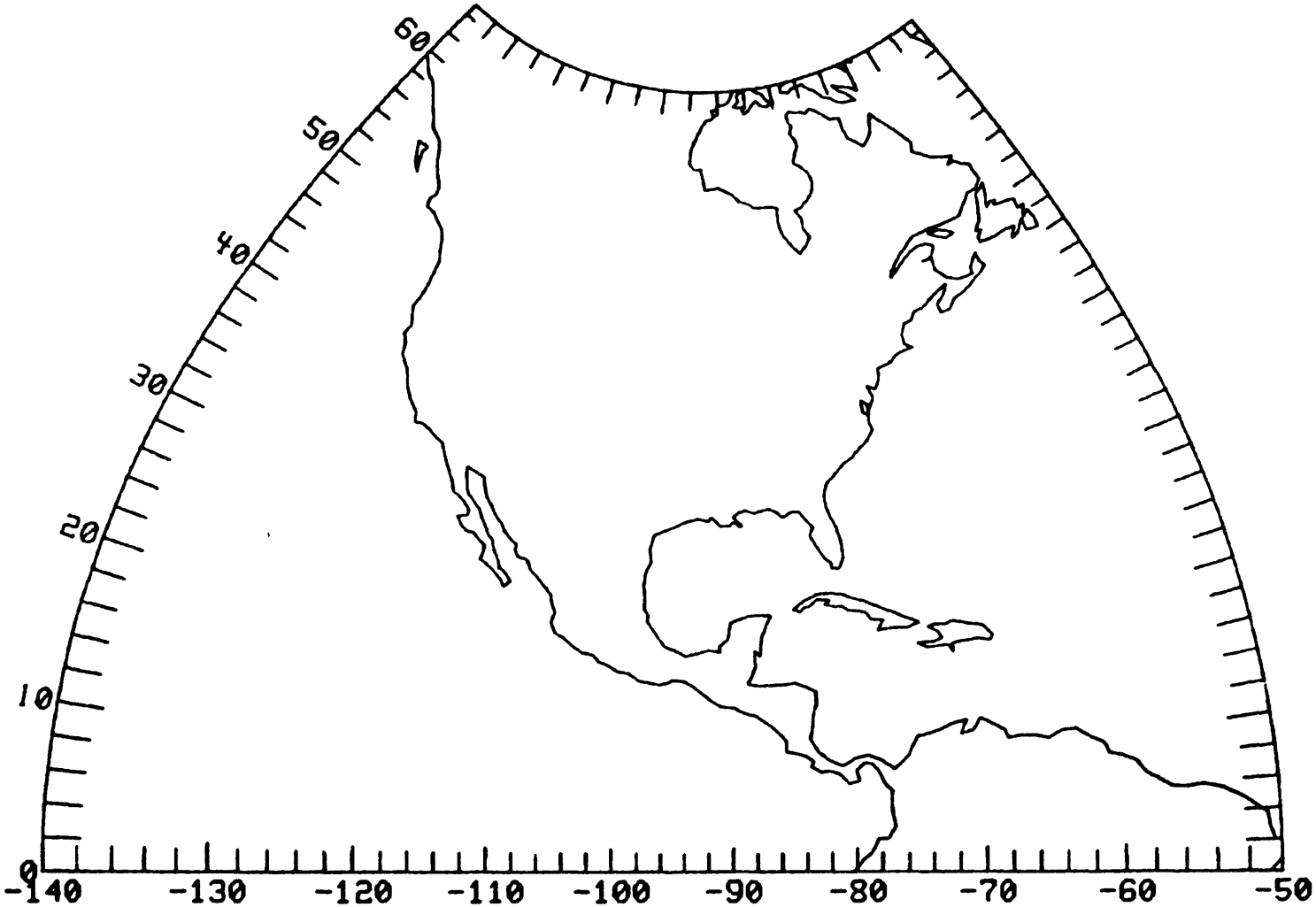
Change to miller map transformation.



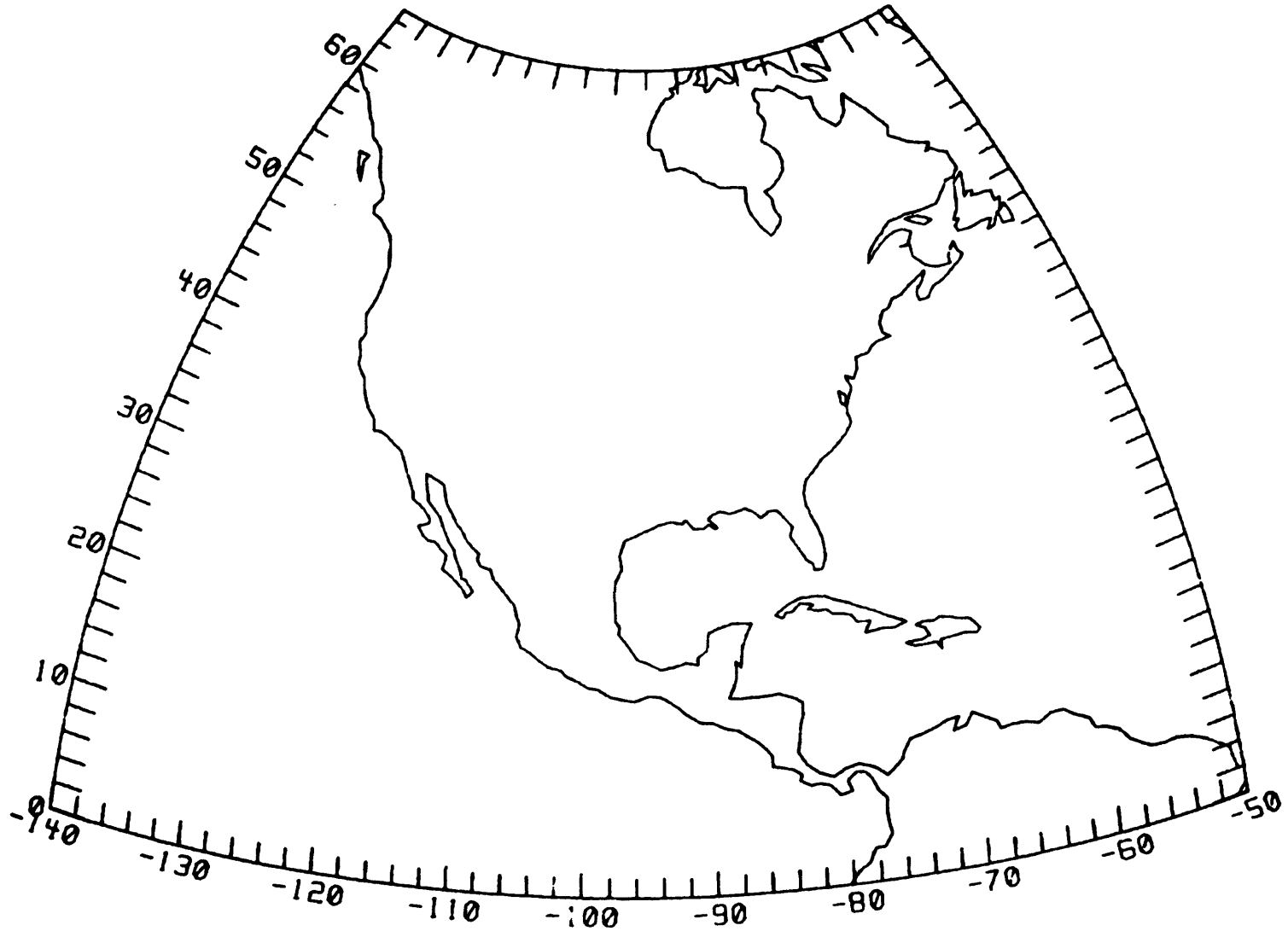
Change to transverse map transformation.



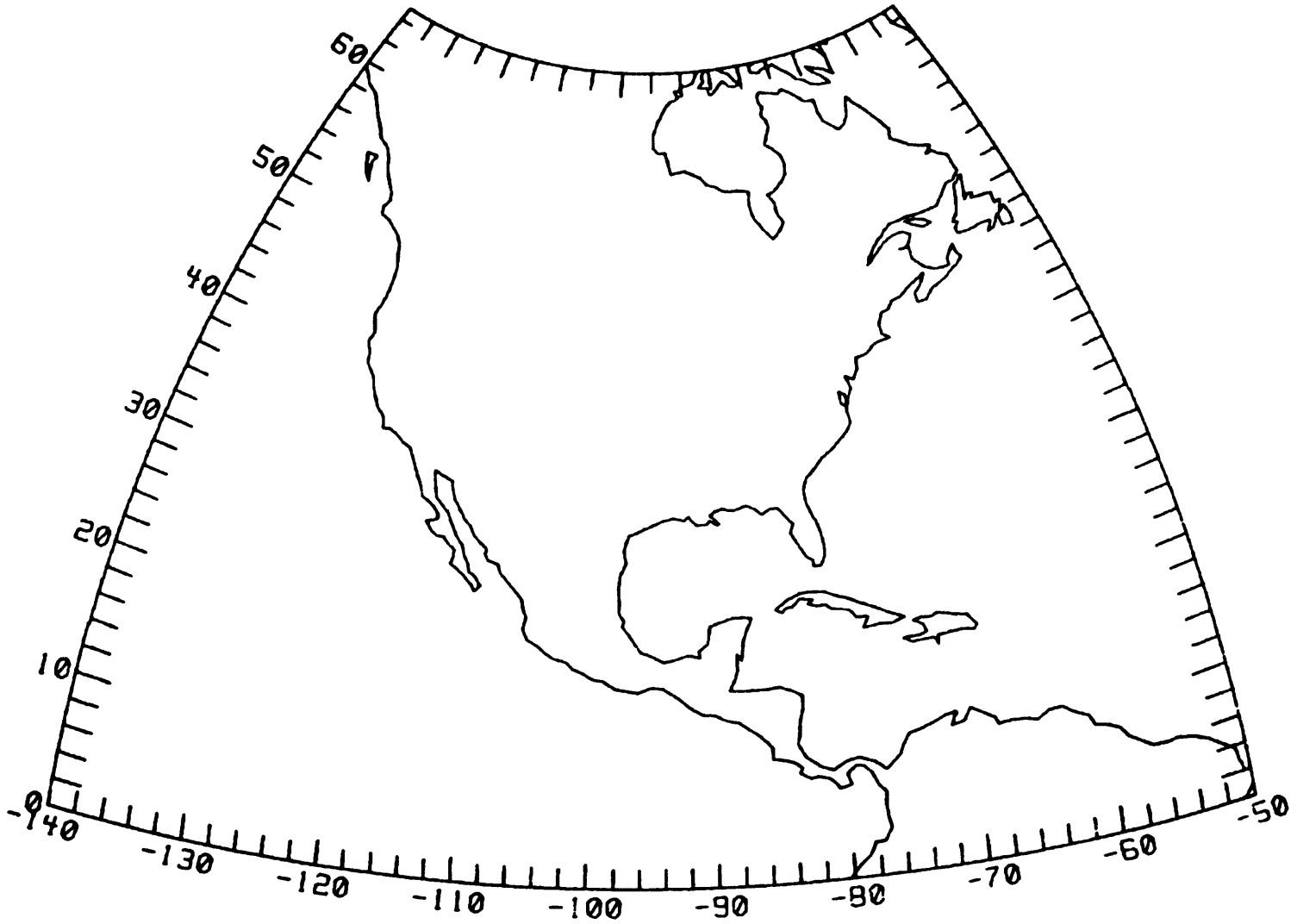
Change to utm map transformation.



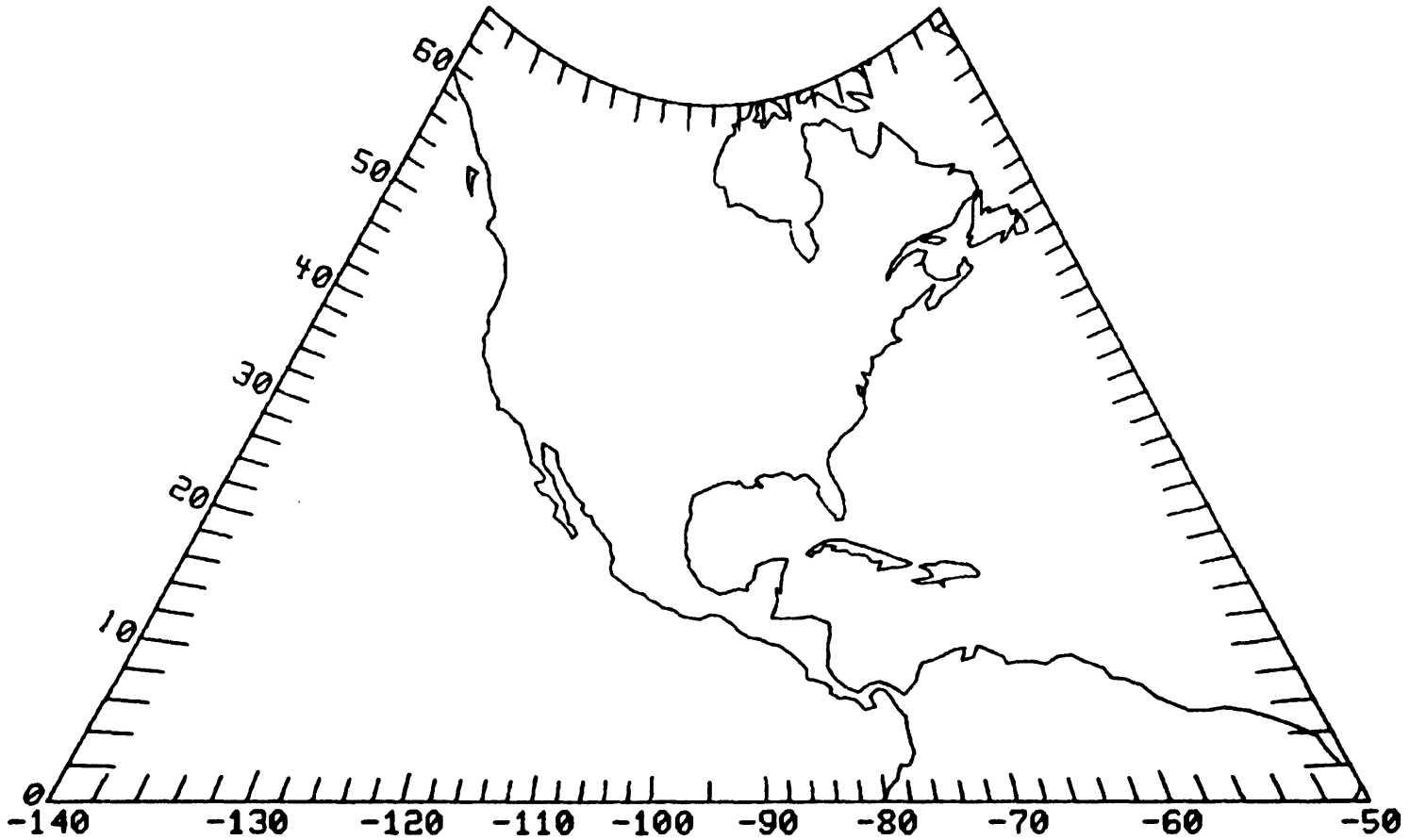
Change to areaequal map transformation.



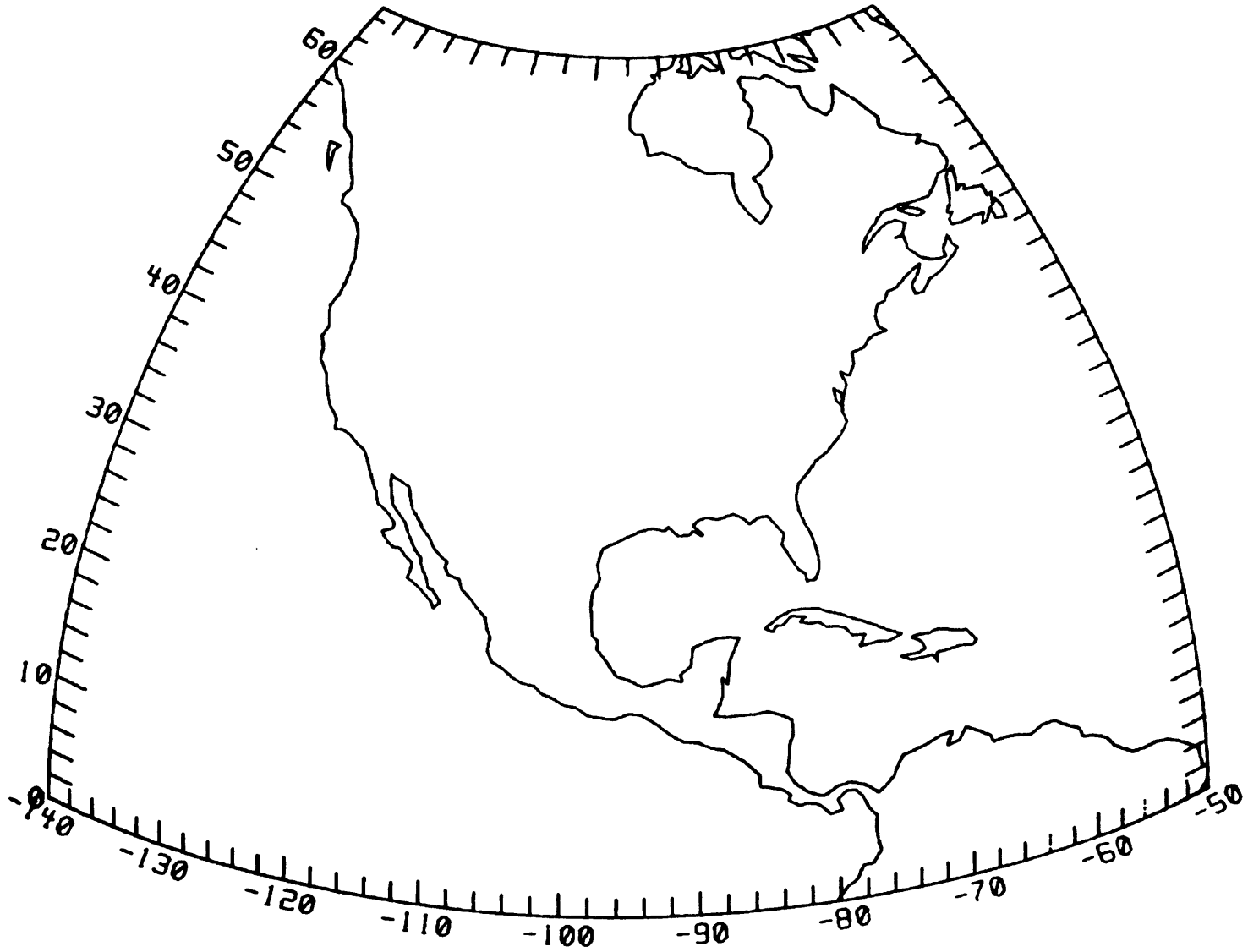
Change to distequal map transformation.



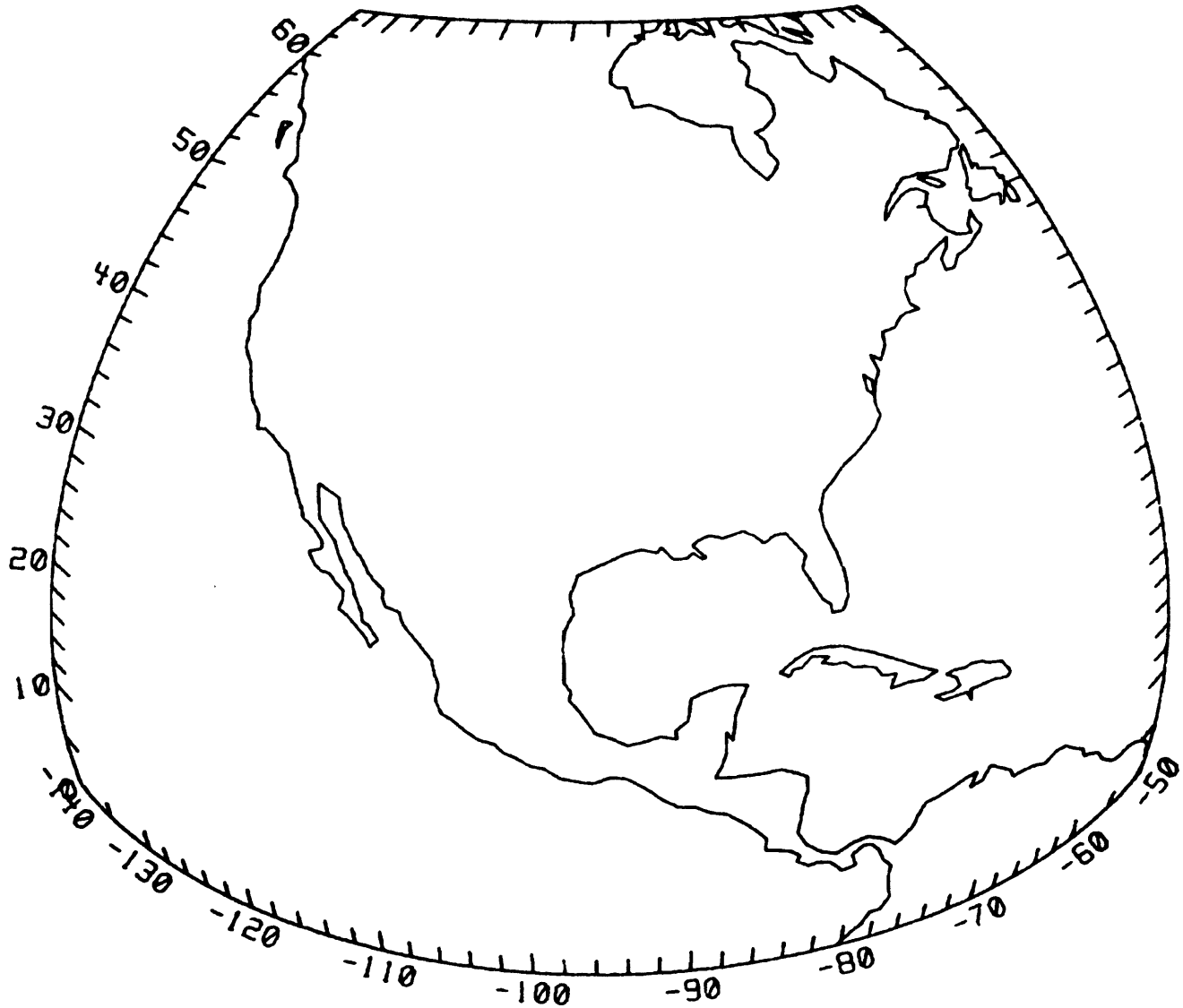
Change to gnomonic map transformation.



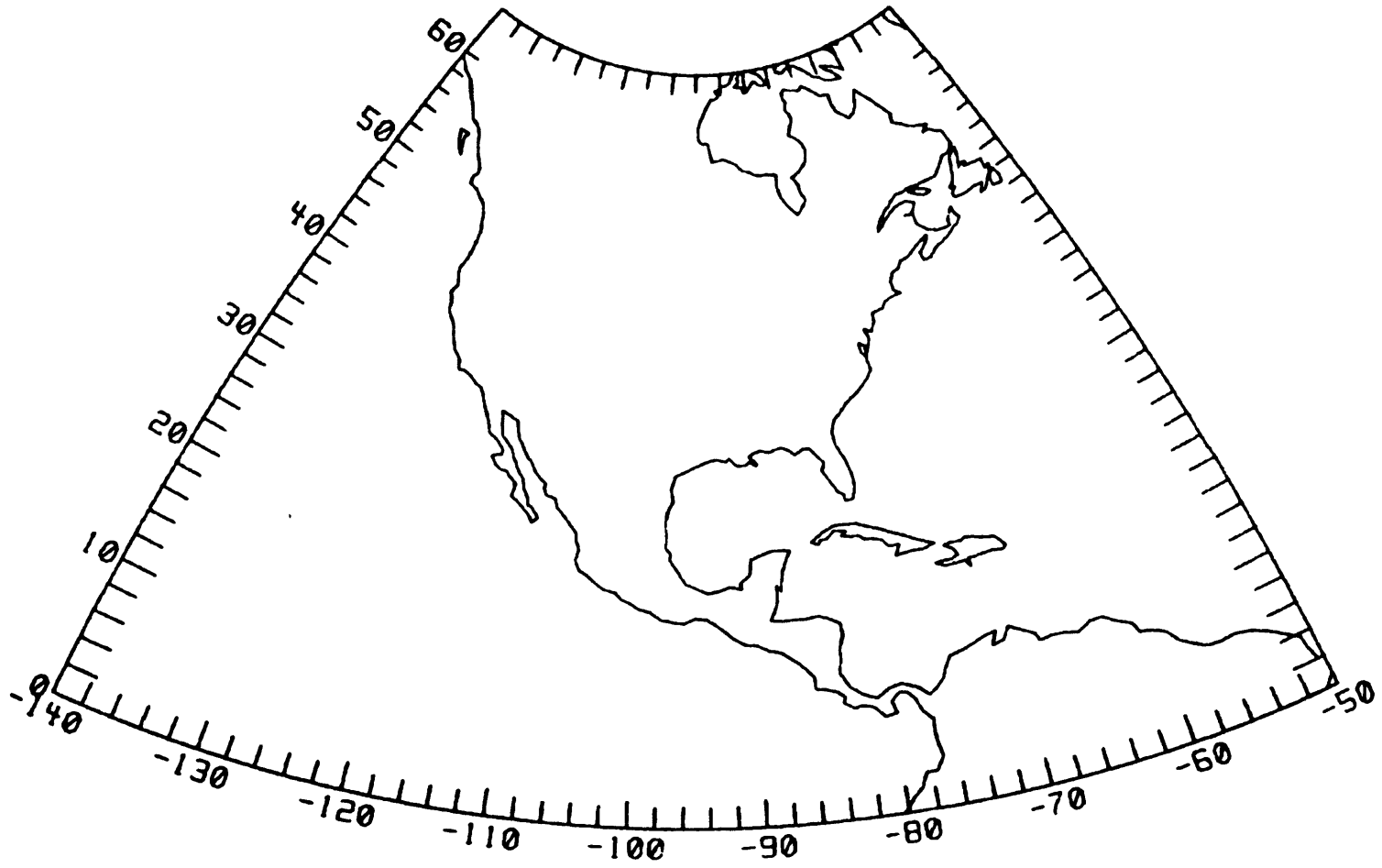
Change to orthographic map transformation.



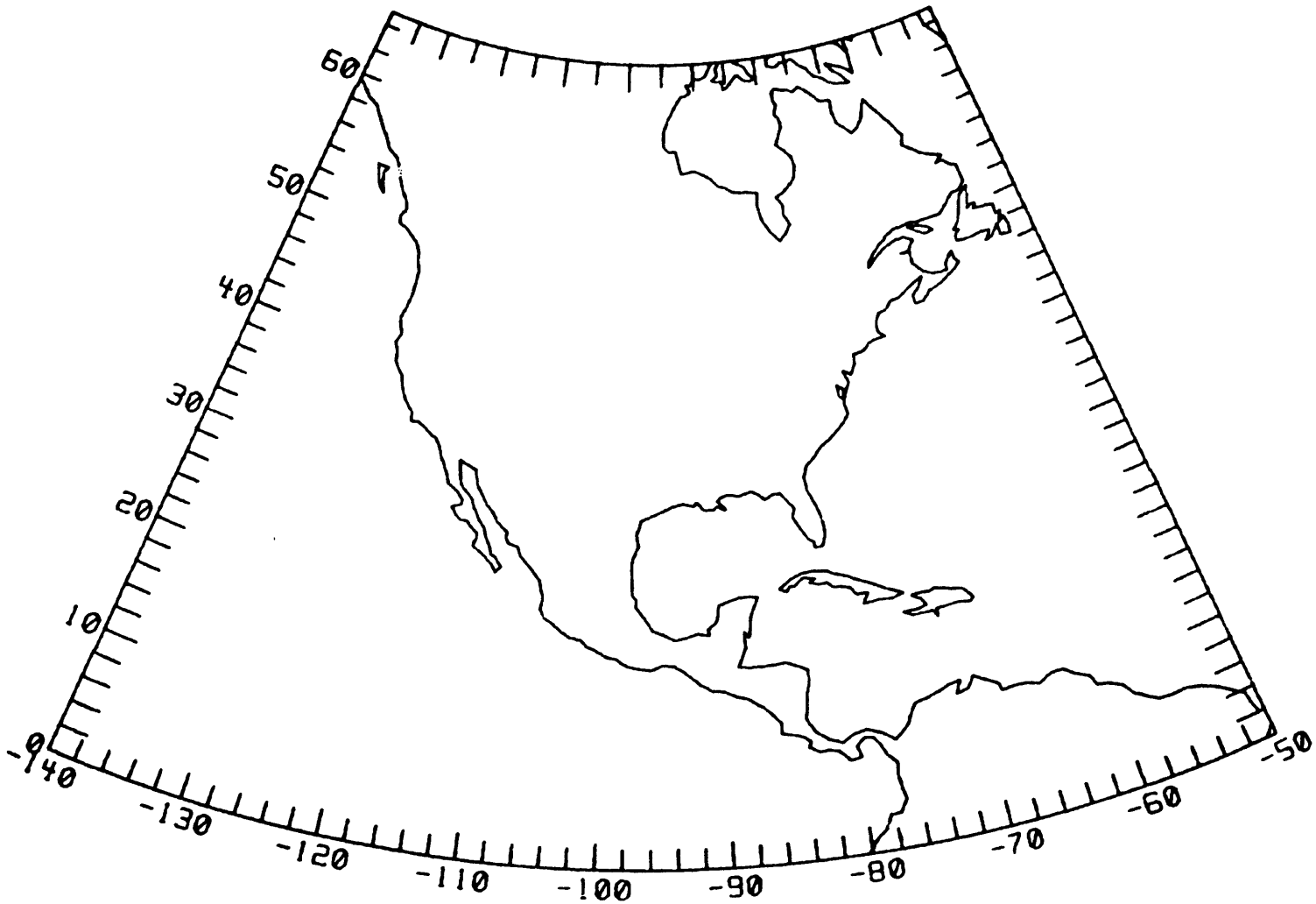
Change to perspective map transformation.



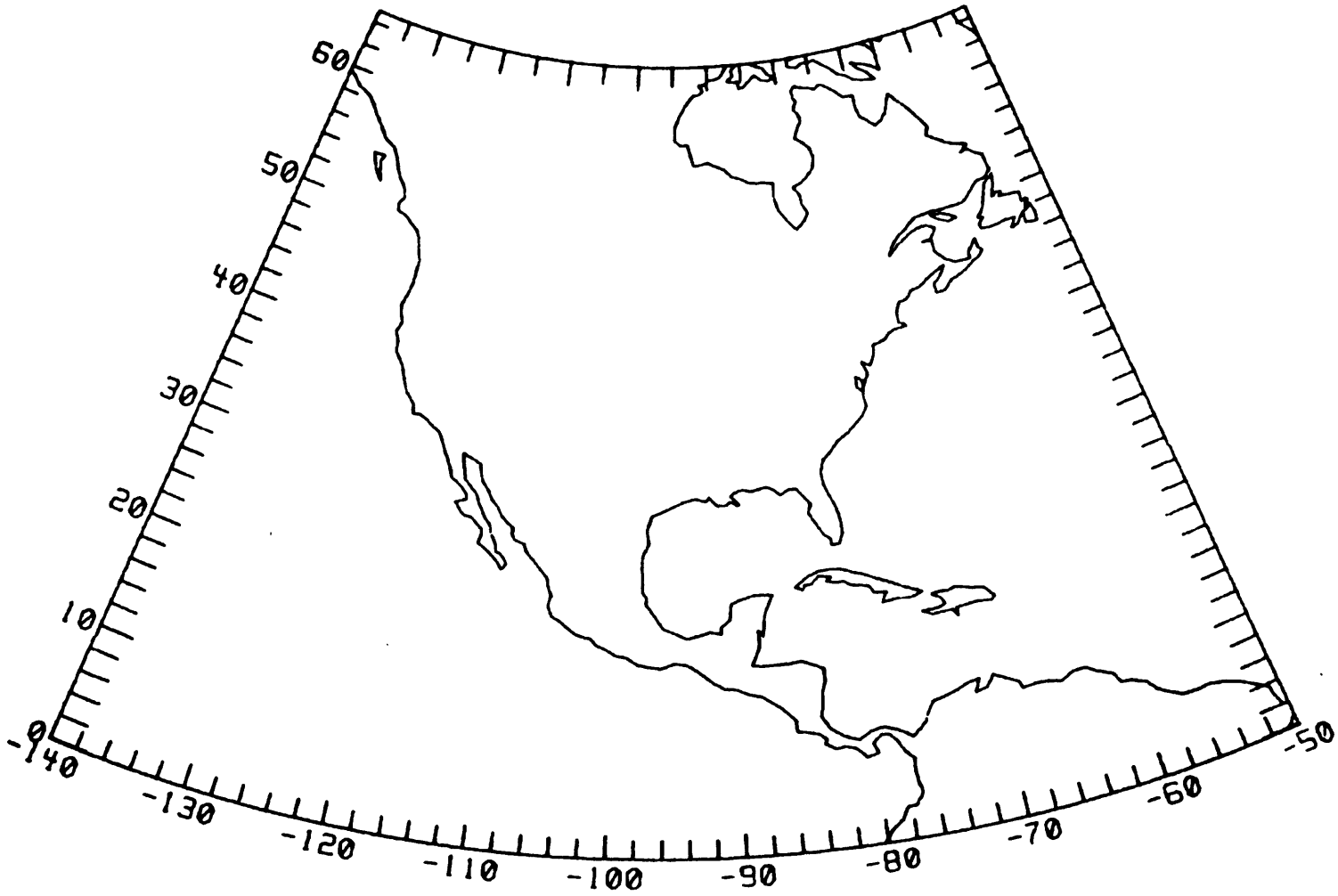
Change to stereographic map transformation.



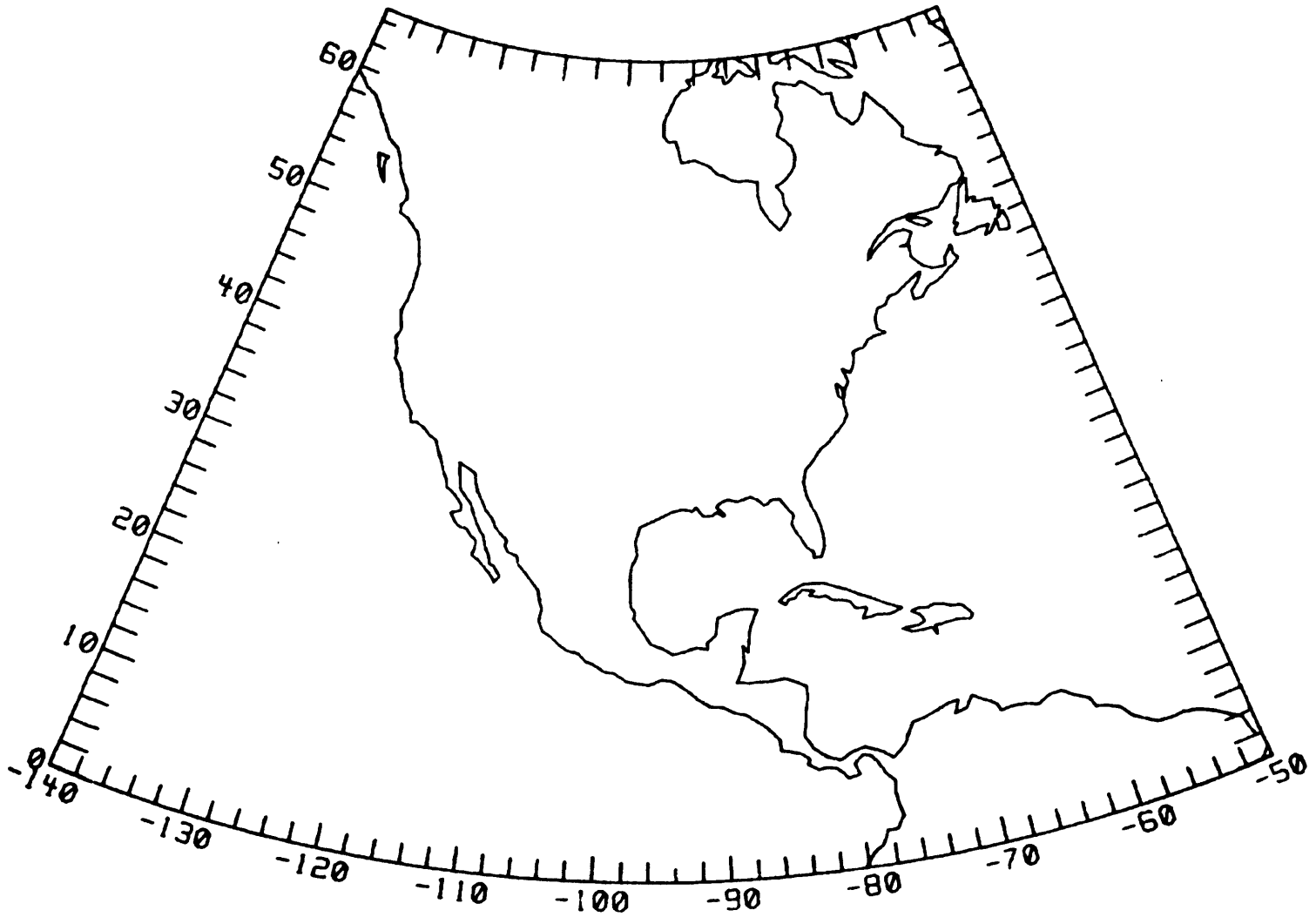
Change to lambert map transformation.



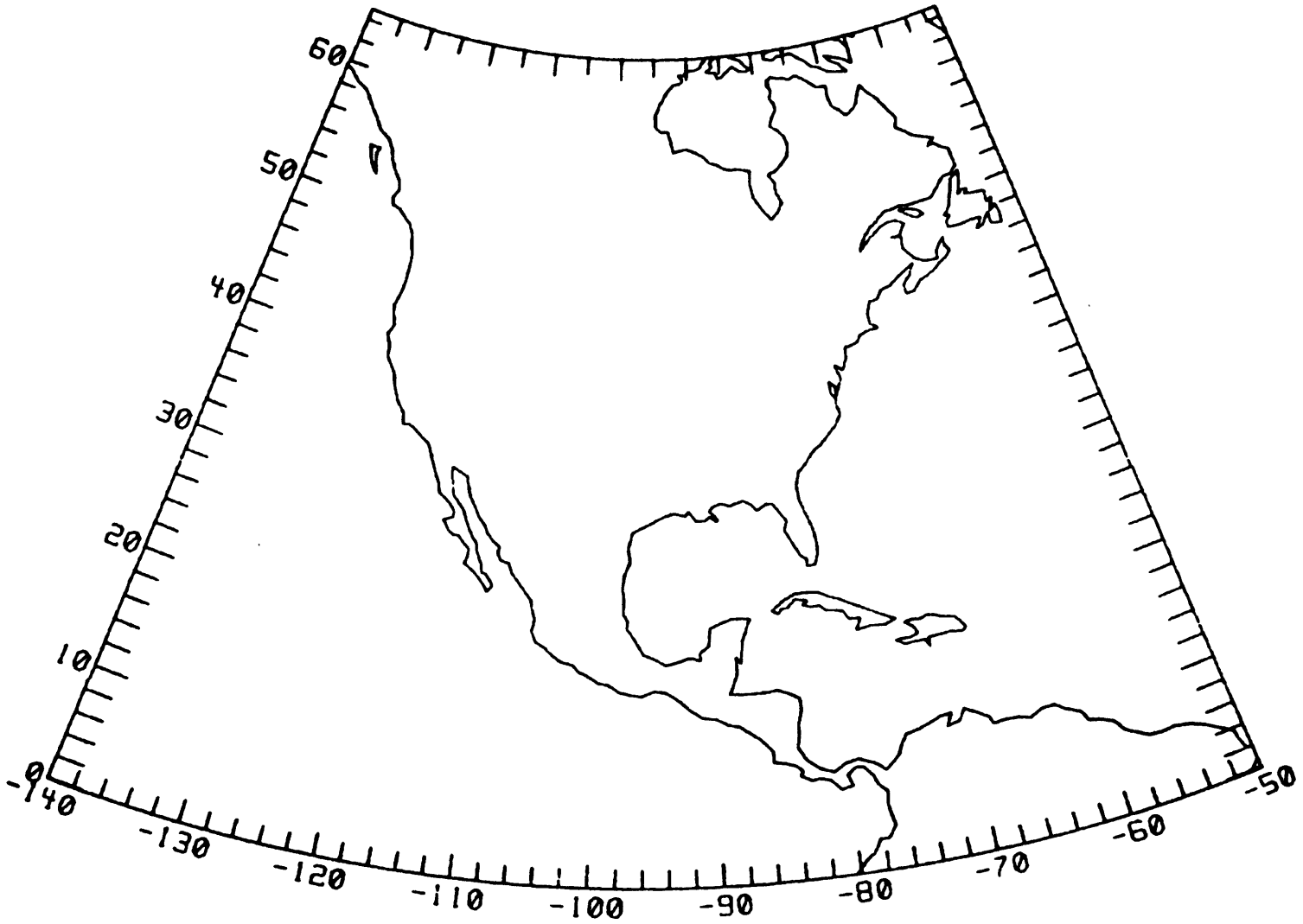
Change to ptolemy map transformation.



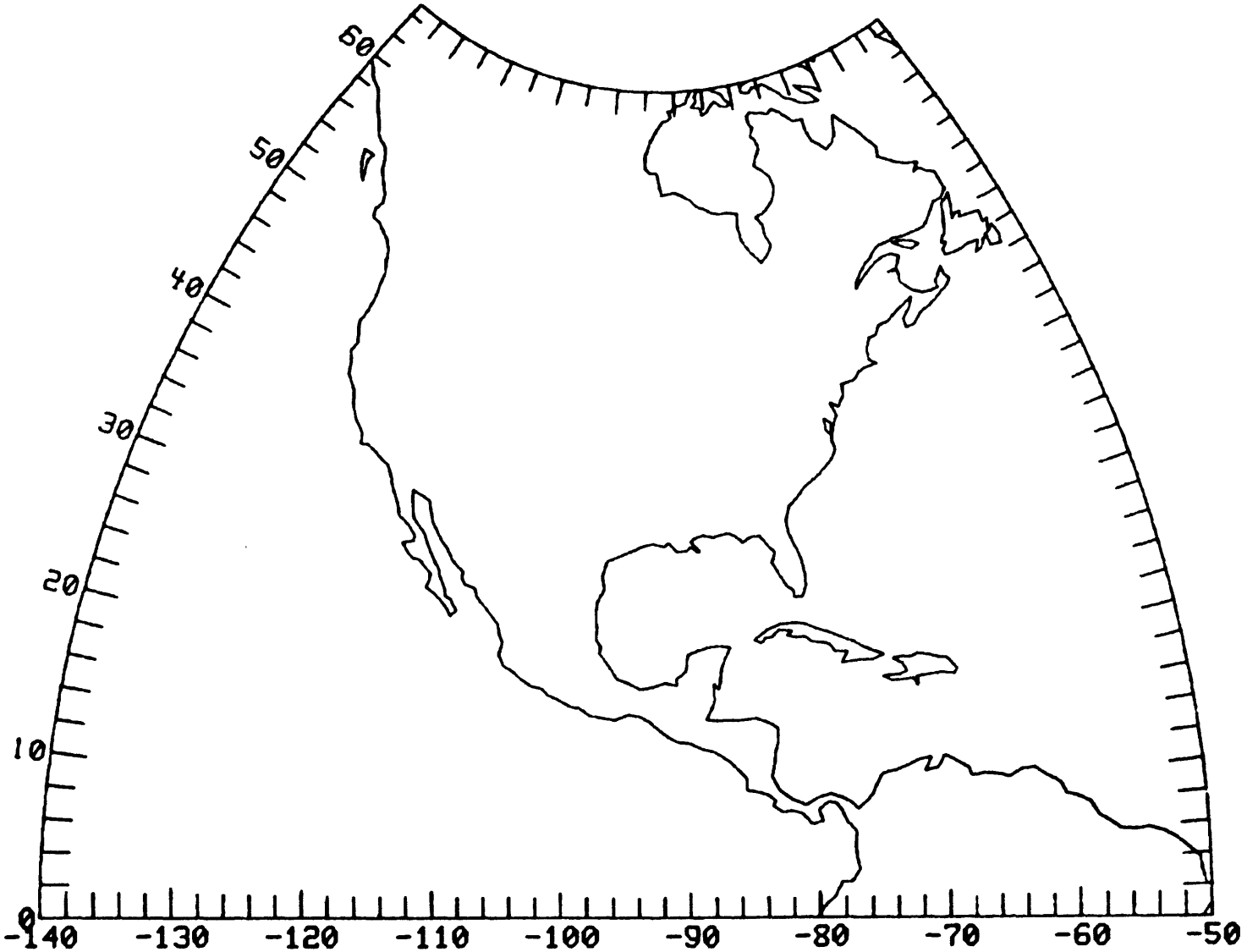
Change to kavraisky map transformation.



Change to albers map transformation.



Change to polyconic map transformation.



Change to sinusoidal map transformation.