Bell Laboratories

Cover Sheet for Technical Memorandum

The information contained herein is for the use of employees of Bell Laboratories and is not for publication. (See GEI 13.9-3)

Title- **Instructions for GRAPH , TEK , TEKSTARE and GSIP graphics packages**

Date- **June 30, 1976**

TM- **76-1527-35**

Other Keywords- **UNIX,MERT,STARE**

Author	Location	Extension	Charging Case- **39091-7**
A. R. Storm	**MH 1B-124**	**6238**	Filing Case- **39091-7**

ABSTRACT

The **graph** routine provides a grid, scales data and produces a label defining the limits of the abscissa and ordinate. When this information is piped "|" into the routine **tek** it is converted to data compatible with the 4014 storage scope terminal's internal requirements. Similarly, piping into the **gsip** routine produces properly translated and formated data to drive a GSI type terminal in the plot mode. The **tekstare (tks)** routine takes information in the 4014 format, transforms it to STARE compatible information and spawns a job on the HIS 6000 which produces STARE output.

These routines were stolen from the "research" machine with the help of M. D. McIlroy. The **graph** , **tek** and **gsip** routines were written by M. D. McIlroy and L. L. Cherry while the **tekstare (tks)** routine was written by Mike Lesk. The help of all of the above is gratefully acknowledged.

Pages Text 2	Other 7	Total 9
No. Figures 6	No. Tables 0	No. Refs. 0

Bell Laboratories

Subject: **Instructions for GRAPH , TEK , TEKSTARE and GSIP graphics packages**
Case- 39091-7 -- File- 39091-7

date: **June 30, 1976**

from: **A. R. Storm**

TM: **76-1527-35**

MEMORANDUM FOR FILE

1.0 GRAPH OPTIONS

1.0 "−a" automatic abscissas -only input the y data points-

"−d" disconnected -no lines between points-

"−s" save screen -overlay plot-

"−g" grid control ... −g = −g0 = don't draw a grid, −g1 = abscissa and ordinate with ticks, −g2 = full grid, this is the same as no −g argument at all.

1.2 The following options may all have one or more character or number arguments ... these arguments should be seperated from their option code and each other by a space.

"−l" include in the label the string that follows in quotes. If the −g or −g0 option is used no label will be produced.

"−c" plot character---use following character to plot points---don't use " * "; " . " is default for tek and " + " is default for gsip when the -d option is specified.

"−x" left and right limits of grid.

"−y" lower and upper limits of grid.

"−h" set height of plot---range is 1.0 by default and may be set down to about 0.1 (times the full sized plot grid).

"−w" set width of plot---range as in "−h" above.

"−r" offset the origin to the right---by the specified fraction of the full sized plot.

"−u" offset up the screen--- same as −r above---

2.0 EXAMPLES OF PIPING - gsip may be used anywhere tek is used. (cr) stands for carriage return, EOT is a control d.

2.1 **graph | tek(cr)**---*PLOT FROM THE TERMINAL*--- type"graph | tek(cr) |then follow with sets of x,y data interspersed with as many (cr's) as necessary followed by a final (cr) and EOT(control d). eg.
%graph | tek(cr)
0.0 0.0 1.0 1.0 2.0 4.0(cr)
3.0 9. 4. 16. 5.0 25.(cr)
6. 36.(cr)
(EOT)
(Decimal points are not necessary for integers.)
The above sequence should plot the right hand branch of a parabola.

2.2 **graph < datafile | tek(cr)**---*PLOT FROM FILE datafile*--- Takes data from a file (in the same format as in 2.1 above) and plots it on the 4014.

2.3 **prog | graph | tek(cr)(EOT)**---*PLOT THE OUTPUT OF A PROGRAM, prog*--- If prog is in bas[ic] you must use the sequence prog | graph | tek(cr)run(cr)(EOT).
These instructions will plot the output of a program when its data is in the format as described in 2.1 above.

2.4 **graph < datafile | tek | tekstare**---*PLOT FROM FILE, datafile ON STARE*---

Your HIS 6000 job and box must be included in your password file entry for tekstare to work. The telephone interfaces necessary are the same as those needed for things like **opr**. **tekstare** is documented in section VI of the UNIX PROGRAMERS MANUAL but may be too new to be available except in the online manual(man 6 tekstare); it is, therefore, included as appendix 1.

3.0 **EXAMPLES TO FOLLOW**---see figures 1 thru 6.

MH-1527-ARS
Attachments
Appendix 1
Figures 1-6

NAME

tekstare – convert tektronics picture to hard copy graphics

SYNOPSIS

tekstare [-d] [-j] [-m|f|s|v|c|l|p] [files]

DESCRIPTION

Tekstare is a sink which expects input intended for the Tektronics storage scope terminal (ttyt) and makes hard copy on the STARE 3 printer or microfilm. It can be piped into by such programs as kaaishu(6) or tcatsim(?).

If input files are provided, a frame feed is placed between each file. When no input files are specified, the standard input is read.

The options are as folows:
-d indicates debugging: write the GCOS job on the standard output, but don't submit it.
-j means mail job number back to you when the job is submitted.
-s means STARE output (default).
-f or m means microfilm output. Paper prints and film are returned.
-v means viewgraphs (done via microfilm).
-p means photo (done via microfilm)
-c means "cdraw" (large drawings)
-l means slides (2x2 inches)
You can only get one output medium in a job.

STARE has some hardware differences which limit the accuracy of the reproduction - the character sizes are different, resolution is only 1000 by 1000, lines which are not horizontal or vertical are drawn by steps, and the overall size of the output page is about one-half that of the Tektronix screen.

FILES

/usr/dpd/yf11? Spooling file for GRAFPAC commands. pounce/stare.r (GCOS) GRAFPAC driver.

BUGS

It doesn't know about all of extended graphics mode - only the normal modes, point plot mode, character sizes, and high resolution addressing are recognized. The GCOS submitter types nonsense on your teletype. The reliability of the pdp-8 which connects to GCOS is not all that could be desired. Instructions for rebooting it are posted on the machine itself, in room 2D501. Microfilm or any derivative product (anything but STARE) takes several days.

```
%graph -l "Parabola Test" -c + -g1 -h .4 -w .4 -u .25 -r .25|gsip(cr)
0 0 1 1 2 4(cr)
3 9 4 16 5 25 6 36(cr)
^T)
```

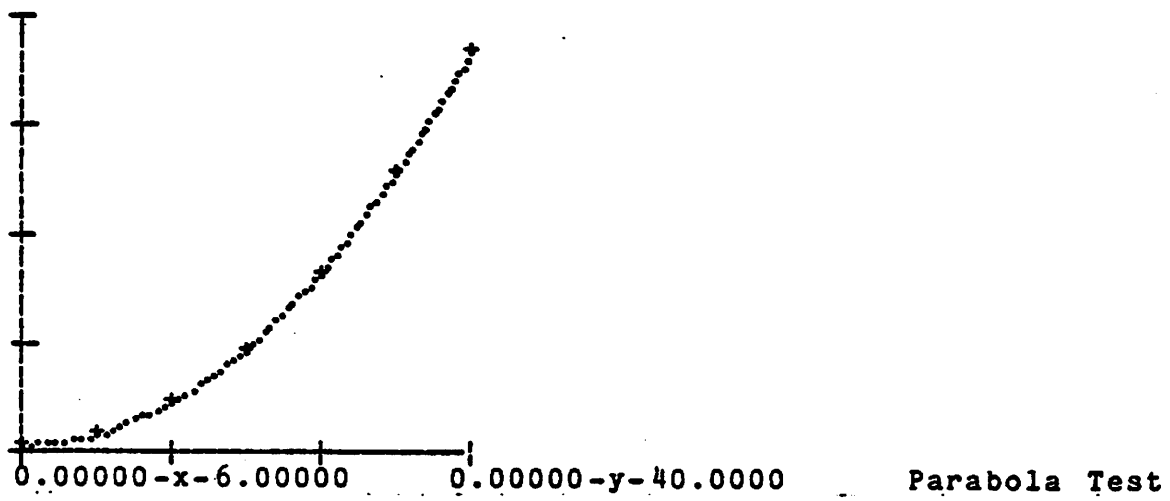

figure 1

Plotting from the keyboard on a DTC 300/S (gsi type) terminal. The -l, -c, -g, -h, -r, -u and -w options are demonstrated.

```
graph -u .15 -r .15 -d<data|gsip
```

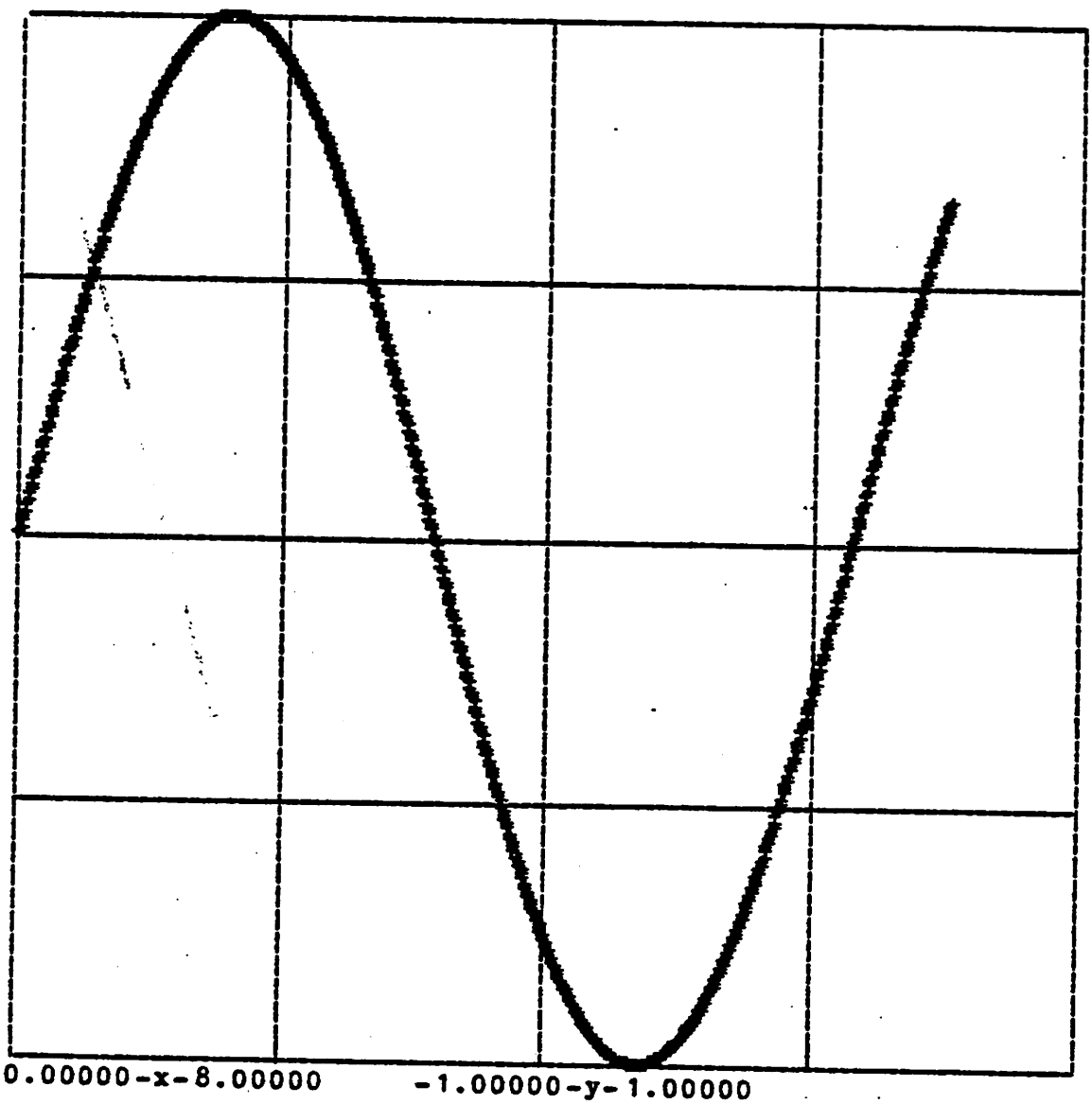

figure 2

Plotting from a file *data*. The -d option invokes the "+" sign as the ploting character with the gsip driver and with the closely spaced data points makes a very heavy line.

```
graph -g -d -c . -u .3 -r .2<data|gsip(or)
```

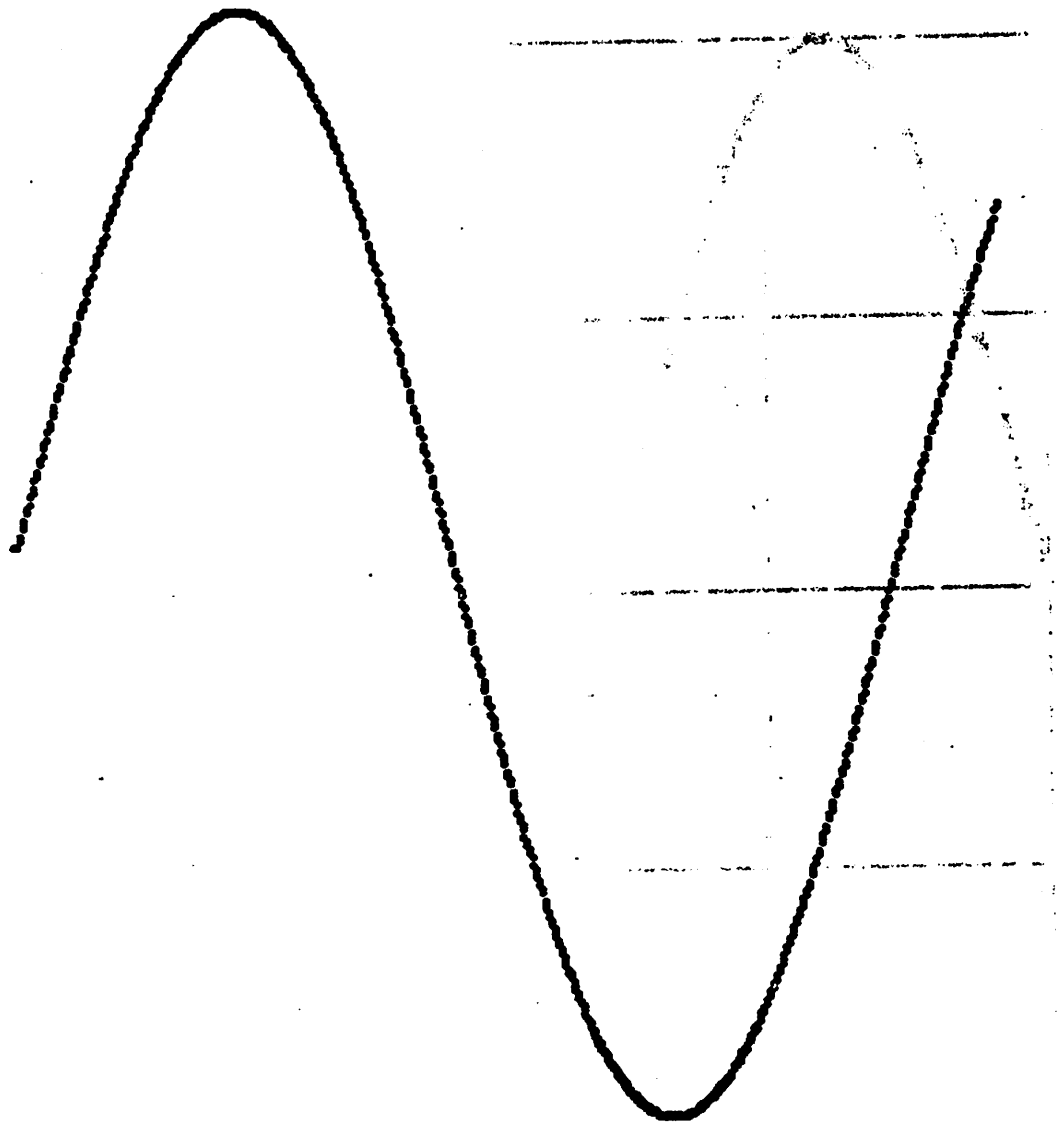

figure 3

Plotting from the file *data* again, but with the -c argument setting the plot character to "." . The -g option eliminates the grid.

```
% cat parastring
10 for i=1 100
20 a=i-101
30 y1=a^2
40 y2=i^2
50 print a
60 print y1
70 print i
80 print y2
90 next i
% bas parastring|graph -h .5 -w .5 -u .5 -r .25|gsip(cr)
run(cr)
(EOT)
```

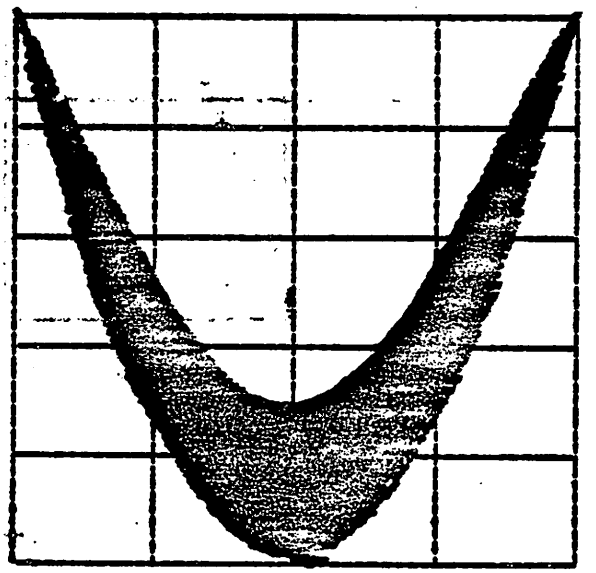

figure 4

Plotting the output of a program. Note that since the -d option is not used successive data points are

```
% bas  parastring|graph -d -u .2 -l "note the -d option"|gsip(cr)
run(cr)
(EOY#T)
```

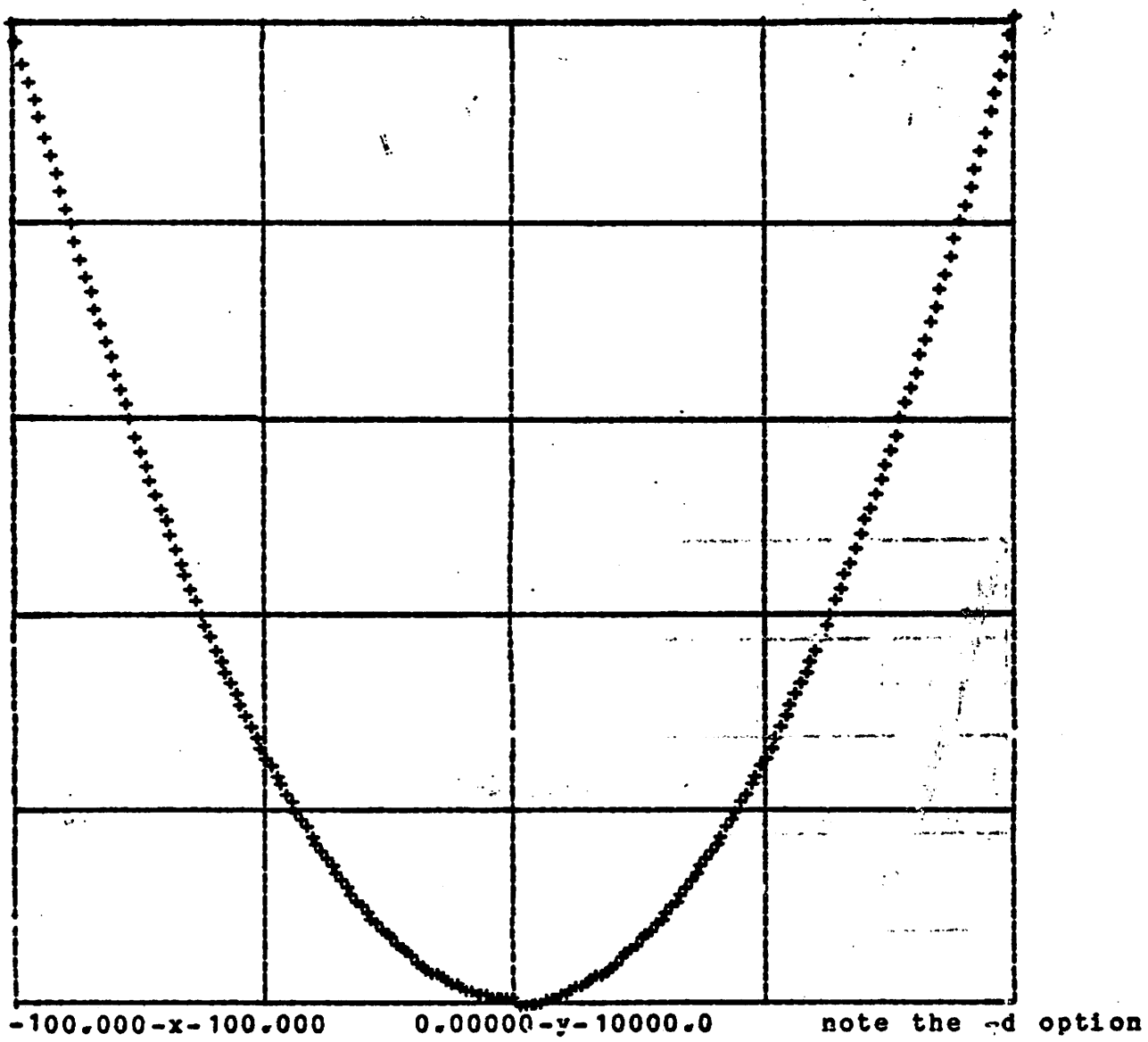

figure 5

The same program as in fig. 4 but the -d option is used This program generates a very ineffecient plot on the "gsi" type terminal.

```
% graph -d -c + -y 0 3e-07<data3 | tek|tekstare(cr)
```

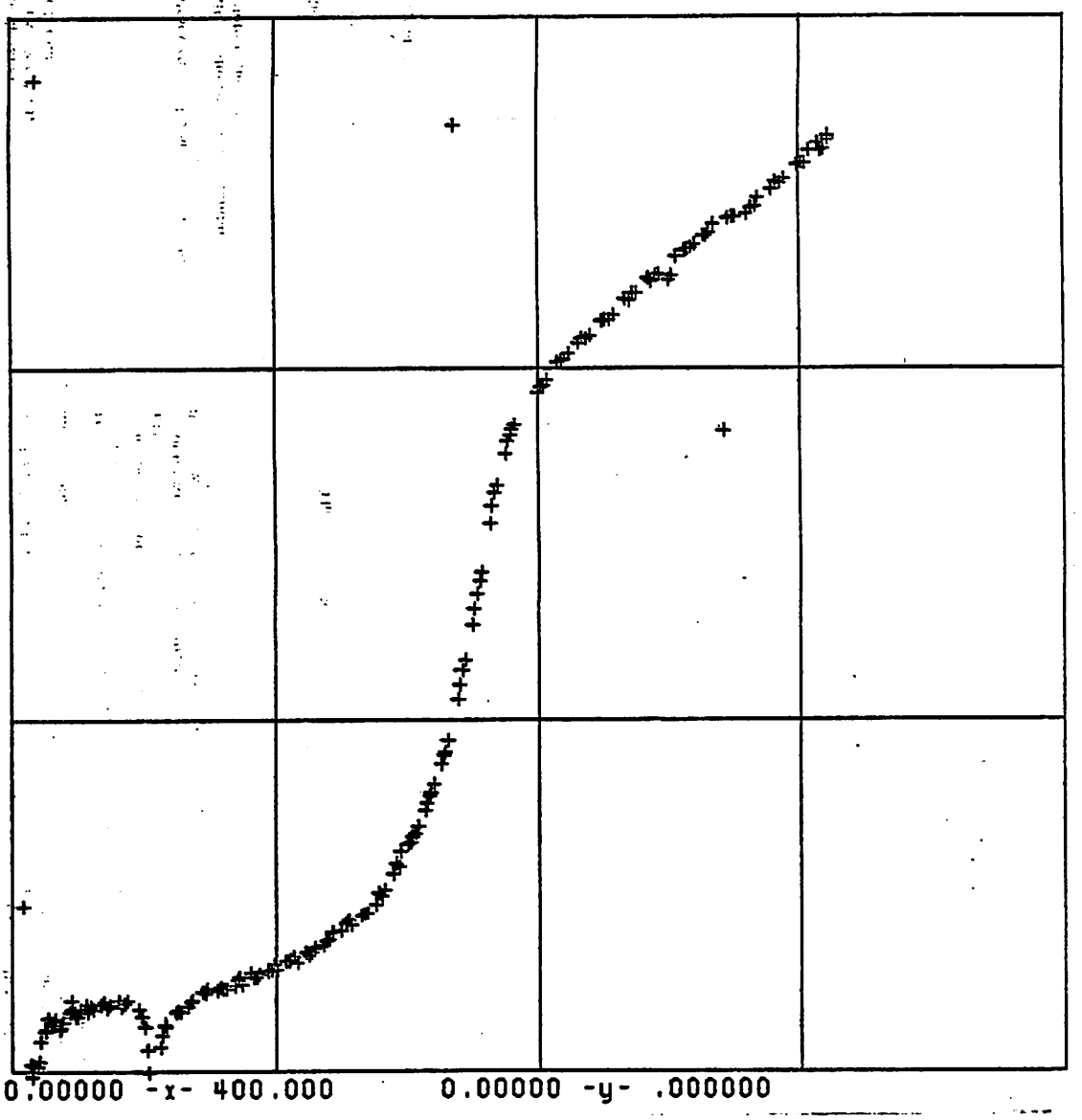

figure 6

A STARE plot of some experimental data. The upper "y" value input was 3e-07(.0000003) but the graph program only displays six digits in these feields. Note the -d option used also.