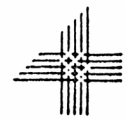


FORTH

USER GUIDE
to programs for interactive image processing
and microdensitometry on the RGO PDS.

JONATHAN MCDOWELL
KEITH TAYLOR

RGO 26/07/78

INTRODUCTION

[ø.1] Conventions:

i) User typed FORTH instructions in capitals.
ii) Terminal response in *italics*
iii) Comments and descriptions in lower case.
iv) In user typed FORTH instructions, lower case symbols such as: n, m, x, y stand for user typed numbers and are defined in accompanying comments. Numbers are to be assumed to be in 16-bit format (integers, unpunctuated) unless stated otherwise.
v) PDS coordinate directions defined by x and y.
vi) Size of 2-D data arrays on disk or tape defined by ρ (for Rows) and κ (for columns).
vii) Pixel identification within those arrays given by i, specifying row number and j, specifying pixel in row (column number).
viii) Some parameters are specified by 2 components. In this case they will be identified using vector notation in the comments. For example, $\widetilde{WN}$ is a parameter representing the dimensions of a 2-D array or scan. Its components are 2, 16-bit numbers referenced by WN (κ - direction) WN 1+ (ρ - direction).

[ø.2] Instructions:

FORTH instructions are single words, delimited by spaces. The words will be executed on typing a carriage return. (represented here by (cr)). On completion of execution, if execution has not caused any errors, output will be *OK*. Several instructions may be typed on line, separated by spaces. They will be executed in turn. FORTH uses a push-down stack for number operations. If an instruction is encountered which is not recognised, the word is echoed back followed by a question mark, and execution is interrupted.

[ø.3] Number entry:

FORTH uses single-precision (16-bit) and double precision (32-bit) signed integers. 16-bit integers are in the range $-32768 \leqslant n \leqslant 32767$. These are used for most purposes but sometimes, 32-bit numbers are needed, (for example, PDS co-ordinates). 16-bit integers are entered with no punctuation except a leading minus sign where applicable; ie the same as FORTRAN integers (but there must be no leading plus sign). 32-bit integers are entered by including punctuation either , . - or / . Any number which includes punctuation (except for a single leading minus sign which means a negative number for both 16- and 32-bit numbers) will be treated as a 32-bit number, but it will be treated as an integer, ie position of the punctuation will be ignored, hence the numbers 12.34 1.234 12,34 are all taken to be the same number, a double-precision 1234. To avoid confusion, it is probably simplest just to terminate with a point (eg 1234.)

Jonathan C. McDowell

[Ø.4] Basic operations:

Arithmetic is reverse polish; integer only.
Typing a number puts it on the stack.
Typing . prints out the top of the stack.
Typing (number) (address) ! stores (number) in (address).
Constants To put value of constant, R on stack just type;
R
n ' R! stores n in R
Variables Typing the name of a variable, DX puts address of DX on stack.

To put value of DX on stack type;
DX @
n DX ! stores n in DX
The command ? is equivalent to @ . hence it prints out value in address currently on stack; eg, DX ? prints current value of DX.
To set number base, type DECIMAL, OCTAL or HEX.
For more detailed explanation of subjects in [Ø.2] to [Ø.4], see FORTH course notes , FORTH Reference Manual.

[Ø.5] Note on FORTH words:

The FORTH compiler only stores the first three letters of each word and its length thus, FILE is treated the same way as FILL; FORTH cannot tell the difference. FILE1, FILE2, FILE3, FILA! are treated the same, but FILEAB is not (different length). This is useful when a long word is spelt incorrectly but can lead to confusion. Also note: CENTRE = CENTER, DISK = DISC etc. This applies to 'words', ie FORTH commands, but not to alphanumeric strings. These strings are usually used as identifiers or file names. They are only valid if used after a command requiring such a string; such commands include FIND, WRITE, INTER, ENTER. In this documentation, notation of the form (name) or (filename) will refer to one of these strings (eg; FIND (filename) might be FIND GALAXY). These strings should not have any spaces in them, except where mentioned in the documentation.

[Ø.6] Interrupt:

"Control C" acts as an interrupt. This may not work when tape is being used.

[Ø.7] PDS control:

For user to move PDS stage, the SELECT (blue and white) button must be on 'manual', for the computer to move it, it must be on 'auto' (eg when doing a scan). Remember to turn control over when using a command which requires stage movement, as the documentation does not mention this. Control can be turned over before or after the relevant command has been sent.

[Ø.8] Images:

The same basic instruction, PASS, is used to transfer scan data from PDS to tape, or from disk to terminal display, etc, ie from one representation to another. The format of the instruction is (unit1) SOURCE (unit2) DESTINATION PASS to transfer data from (unit1) to (unit2); the format (location, array size, scan sample interval, etc.) is determined by a set of parameters; each unit has its own set. The whole instruction may not always be typed explicitly, eg in plotting routines the word PLOT may be defined as SOURCE SCREEN DESTINATION PASS, and so DISK PLOT will transfer the data from the disk to the screen in graphic form.
Each unit name is called an 'image'. A list of 'images' follows.

Images

PLATE	The PDS stage; PLATE SOURCE (unit2) DESTINATION PASS does a scan
DISK	A file on disk. The word DISK is used implicitly in the instruction FIND (filename) which also selects a specific file.
TAPE	A file on tape
TERMINAL	The terminal; for digital output
SCREEN	The Tektronix terminal; for graphic output

Also:
INPUT)
OUTPUT) Low-level functional images used by PASS
OPERAND)

Parameters (these can be treated as 'variables')

LOCATION	Disk block number for DISK image; not used for other images. Value is 5ØØ ⩽ n ⩽ 2435.
CN	Size of scan in pixels; CN contains number of pixels in a row CN 1+ contains number of rows in the scan
DX	Sample interval in microns: DX contains sample interval in X DX 1+ contains sample interval in Y
CØ	Origin of scan in PDS co-ordinates. This is the first pixel in the array, designated pixel Ø, row Ø, (for an nxn array pixels and rows are numbered Ø through n-1.) CØ contains 2 double-precision numbers: CØ contains the X co-ordinate, CØ 2 + contains the Y co-ordinate.
WØ	A 'window' or subset of the array (on disk, tape, etc) may be defined. The origin (first pixel of first row) in pixels relative to the origin of complete array is contained in WØ. WØ contains origin in pixels, WØ 1+ contains origin row number. ie "i", "j" of origin respectively, see [Ø.1]. WØ and WØ 1+ are always positive.

WN	Size of window in pixels; WN contains number of pixels/row of window, WN 1+ contains number of rows in window.
KIND	Specified direction of scan, ie scan in X or scan in y.

Current image

The parameters listed above are not ordinary variables because each of the 8 images mentioned above has its own set of parameters. For example, typing DX will reference the DX of only one of these images, called the "current image". An image is made the current image by typing its name (or using a word which itself used an image name). For instance, TAPE DX ? will print out the current DX value for TAPE, while PLATE DX ? will print out that for PLATE, which may be different. Words which change the current image are:

FIND	makes DISK the current image
TWIDE	makes TAPE the current image
PASS	makes OUTPUT the current image
PLOT	makes OUTPUT the current image after setting OUTPUT to be equal to SCREEN
SOURCE	Copies parameters form current image to INPUT and leaves the latter as current image.
DESTINATION	copies parameters from current image to OUTPUT and leaves the latter as current image.
SECOND	copies parameters from current image to OPERAND and leaves the latter as current image.

USING FORTH

[1] Bootstrap, Loading

[1.1] Follow 'starting up' instructions in black PDS manual up to point of pressing HALT switch on console; use FORTH SYSTEM DISK 1

[1.2] Disk Bootstrap

Set up correct address on data switches, as in PDS manual , and as indicated on console.
Press console switches: HALT (ie depress HALT/ENABLE)
LOAD ADRS
ENABLE (ie raise HALT/ENABLE)
START

If — *HELLO ?* Go to [1.4]
(Otherwise — continue on to [1.3])

[1.3] Bootstrap Fails

Hit "Return" key
If — *OK* Go to [1.3.1]
Otherwise — Try reloading the disk; see 'closing down' and then 'starting up' again in black PDS manual,

[1.3.1] Block Buffers Corrupted

33350 4010 ERASE (should come back *OK*)
Return to [1.2]

[1.4] Loading

JCM LOAD *20/07/78 14:12 OK* — Response is the system date and time, these may be corrected if desired.

If you do not wish to correct date and time (this is completely optional) go to [1.5]
Otherwise, eg
5/09/78 NOW *OK* — Sets date to be 1978 Sep 5
18:07 UT *OK* — Sets time to be 18:07; GMT and BST are alternatives to UT and are identical, computer does not know about time zones.

To print out system data, time TODAY @ .DATE *5/Ø9/78 OK*

.TIME *28:Ø8 OK*

[1.5] If required, load magnetic tape as described in black PDS manual when signing log book, put 'FORTH' in comments column. To close down see PDS manual. In the following documentation, the response *OK* will be assumed after all commands.

[2] Options

[2.1] List of Options

Select an option; loading is described under the paragraph number given. Subsequently loading another option removes the programs for the first option.*

Option name	Brief description	Go to
Bigscan	Scan a plate; store data on tape, or on disk Maximum row size is 3ØØØ pixels Additional option-result may be averaged down by factor of 2^n in x and y, while scanning.	[3]
Taylor	Scan a plate, store data on disk Maximum row size is 512 pixels Align a plate, by creating coordinate transform between it and a reference plate. Graphic output; range of additional sub-options are available, eg averaging down in x and y as in Bigscan, adding, subtracting and dividing two images; contour plots, moving arrays between disk and tape, defect removal, disk file graphic display	[4]
Smoothing	Averaging digitized scan stored on disk, without shrinking array (boxcar convolution)	[5]
3D	3D plot of a scan stored on disk or tape (several versions)	[6]
Photometry	Fitting 2D Gaussians to digitized scan (on disk or tape)	[7]
Additions		
Very long scans	More than 3ØØØ pixels/row	[8.1]
Multiple scans		[8.2]
Array flipping	Reflecting arrays in x and y axes	[8.3]
Tape file record-by-record graphic display		[8.4]
Coarse file		[8.5]
Density histogram		[8.6]

*Except for options covered in [8.1], [8.2] and [8.5] which are to be loaded "on top of" BIGSCAN and [8.3], to be loaded with either BIGSCAN or TAYLOR. This is made clear in the relevant documentation.

[3] BIGSCAN

[3.1] If you wish to use the capability to average down in x and y, as opposed to simple scan onto tape, go to [3.3]

Otherwise:

[3.2] Loading BIGSCAN

κ BIGSCAN LOAD

κ is the maximum row size in pixels that you will be using. It must be ⩽ 3ØØØ. For larger scan lines see [8.1] Wait for the *OK* before proceeding.

Go to [3.4]

[3.3] Loading BIGSCAN with Averaging Capability

When averaging, a scan of ρ rows of κ pixels each will be stored in an array on tape of ρ/a rows with κ/a pixels each, where a ($= 2^n$) is the value of a constant called AV. However, working storage must be reserved for the full κ pixels.

κ BIGSCAN LOAD

κ is the maximum row size in pixels, before averaging, that you will be using. It must be ⩽ 28ØØ

435 LOAD

Loads averaging routines.

Continue

[3.4] Scanning

Set up over desired scan

Scan may be specified by center or any corner.

If you wish to zero PDS coordinates at this point,

ZILCH

If you wish to scan in x no special command is needed, this is default.
However if you scan in y and later wish to scan in x, you must then use

HORIZONTAL

If you wish to scan in y VERTICAL

```
 HELLO ? JCM LOAD 26/07/78 33:52 OK
19:48 UT OK
1024 BIGSCAN LOAD OK
ZILCH OK
3 MM/S  OK
0 ' ODD !  OK
PLATE 54 64 WIDE 10 10 DXDY! OK
CENTER OK
REWIND OK
NTP OK
WRITE JCM TEST SCAN 1970JUL26 OK
```

Example of a simple scan using BIGSCAN

```
 VERTICAL OK
5 VEL ! 1 ' ODD !  OK
PLATE 32 20 WIDE 12 5 DXDY! OK
LL CORNER OK
WRITE VERTICAL SCAN, ODD=1 OK
```

Example of a further scan using non-standard parameters

To set PDS velocity

Either	n MM/S	Sets velocity in mm s^{-1}
or	m VEL !	Sets velocity in PDS units (m = 255/60n)

To set scan mode:

Either	Ø ' ODD !	Default. Scans in 1 direction only (increasing X, or Y if VERTICAL)
or	1 ' ODD !	Scans back-and-forth, and flips rows in working storage so final result is same

Then set up scan parameters as follows:

	PLATE makes PLATE the current image (see [0.8])
	ρ κ WIDE Size of scan; ρ rows with κ pixels in each row.
	Sets parameters $\widetilde{WN}(=\widetilde{CN})$
	δx δy DXDY! Sets sampling interval in microns dx is interval in x-coordinate dy is interval in Y-coordinate (sets parameters DX)
If you wish to check the above parameters,	STATUS Ignore output column marked 'ORIGIN"

If you wish to define the scan by its center:

	Position over center
	CENTER Stores origin (ie coordinates of most -ve Corner) by setting parameters $C\widetilde{Ø}$ using current PDS position (x,y) and $\widetilde{WN}$ and $\widetilde{DX}$, such that $x_o = x - \frac{1}{2} w_x \delta x$; $y_o = y - \frac{1}{2} W_y \delta y$ where (W_x, W_y) are (κ,ρ) or (ρ,κ) depending on scan direction.

Go to [3.5]

Otherwise define the appropriate corner

	Position over corner	
Either	UR	Upper right corner; default
or	UL	Upper left corner

or	LR	Lower right corner
or	LL	Lower left corner
		These define corner of scan, as seen on PDS viewing screen, not on plate itself. They modify the signs of PLATE parameters DX
Then	CORNER	Stores current PDS coordinates in PLATE parameters CØ. This position will be the first pixel (κ = Ø; ρ = Ø)of the scan.

[3.5] Scanning

For scanning to disk, see the description under TAYLOR LOAD, section [4.7.2]

If tape is a new tape

REWIND	Rewinds tape, initializes variables,
NTP	Writes 1 record and 2 filemarks, then backspaces twice. The record is for compatibility with RT tapes which have no filemark before the first file

Otherwise the tape may be positioned anywhere before the final double filemark.

Then: If averaging option is loaded and you wish to use it on this scan, go to [3.6]

Otherwise	WRITE (identifier)	(identifier) is a string of ⩽ 40 characters (including spaces). Follow this by carriage return.
		(i) Tape will run forward until a double filemark is found, and will then back-space so that the second filemark is overwritten.
		(ii) The scan will be performed using the PLATE parameters set previously. The data will be written on to tape in Scansalot format.
		(iii) After the scan is completed, a double filemark will be written at the end of the file, and the tape will then be back spaced twice, so that a second WRITE wil not require any repositioning of the tap

(For an example of user commands for a scan using BIGSCAN, see p)

Go to [3.7]

[3.6] Averaging to tape

To average on κ x ρ scan into an array of size INT (κ/a x ρ/a)

```
a ' AV !
R-UP TURN-OFF
```

Stores a (= 2^n) in constant AV.
Prints out warning message on other terminal (VDU or DECWRITER if either is switched on. The other terminal cannot be used while AVWRITE is in use since this uses the other terminals' program area as working storage.

```
AVWRITE (identifier)
```

As WRITE (see [3.5]) but averages.
eg If there are 18 pixels/row and AV=4, 1st 4 rows of 18 pixels each will be averaged down to one row of 4 pixels and then written on to tape, similarly for subsequent sets of 4 rows, until the 16th row, κ, ρ $\geqslant$ 17 are ignored.

Continue

3.7] Further scans

You may return to section [3.4]; remember to type PLATE so that any parameters modified will be those of PLATE. Reset any parameters by using MM/s, WIDE, DXDY! VERTICAL, ODD. You may leave any of these unchanged. Then specify origin using CENTER or UR, UL, LL, LR CORNER

3.8] To Read Files Created by BIGSCAN

```
R-DOWN
```

R-DOWN lowers location of working storage to program area and 2nd terminal, necessary due to a hardware fault causing crashes when reading to high memory. Before using AVWRITE after this, use R-UP which raises working storage again.

Go to [4.12] (Reading tape files)

[4]

[4.1] Loading

	TAYLOR LOAD	Loads PDS scanning routines, Tektronix graphics routines, least squares fit routines. STARS file routines, and disk file index. Max no of pixels/scan row is 512.

[4.2] Sub-options

There are a number of additional facilities which may be loaded as well as the programs in TAYLOR LOAD. Only one of these options may be loaded at a time, subsequently loading another one will remove the first.

Either (i)	PROCESSING LOAD		Loads programs to aver in i,j and to add, subtract, divide and normalize two disk files
	Go to [4.3]		
or (ii) Either	REMOVAL LOAD	) see [4.9] before	Loads programs to display data from a file line by line, and to remove defects by linear interpolation.
or	SLICING OPT LOAD	) loading	
	Go to [4.9]		
or (iii)	CONTOUR LOAD		Loads programs for contour plot of disk file.
or (iv)	SCANSALOT LOAD		Facility to move disk files to tape, and vice versa, read tape files etc. Also scans direct to tape
	Go to [4.12]		

4.3] If you wish to align a plate before a scan, so that corresponding arrays may be obtained from different plates for the purpose of adding two plates of same object, or for the normalization of electronographs using a photocathode map go to [4.4]

Otherwise If you just wish to scan to disk without using the plate alignment facility, go to [4.7]

4.4] ALIGN

Description of method: 3 or more points (fiducial marks or stars) are entered into a file, called the STARS file, followed by the scan origin coordinates. A scan is done. Corresponding points are then entered in file for a second plate. Least-squares fitting is used to determine angle of rotation needed for the plates to be aligned (the translative component of the coordinate transformation is taken care of by the software but the rotation must be made zero by physically rotating the stage.) After stage is rotated, re-enter points and iterate until angle is acceptably small. (Usually 2 or 3 iterations are enough.) Then the corresponding scan is done.

(NOTE: the stars file is stored on disk so the data will not be lost until they are overwritten by the user.)

Continue

[4.5] LINK-CLEAR

Clears out stars file.
If there is a set of data already in it (eg you are aligning different sets of plates) omit this instruction, and place the new data later in the file

Position over the first fiducial mark (or star)

If it is a star, you may wish to fit gaussians to it to determine the center point.
The word 'mark' will be used (for this section and the next) to mean fiducial mark or star.

If gaussians to be fitted:

1 ENTER (name)

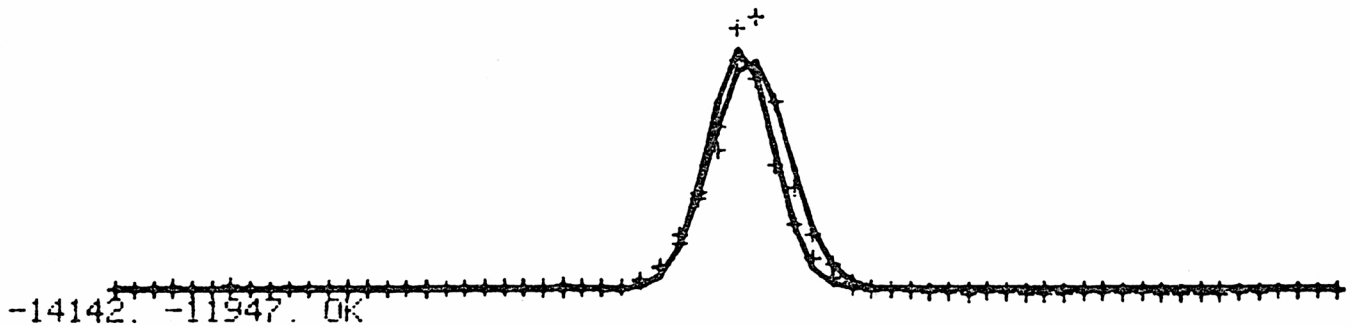
(name) is ⩽ 6 characters long and is for user identification only, not used in programs. This will so a scan 640 μ long (64 pixels) in Y (center of scan line is the current PDS position), fit gaussian to the scan line data, plot the data points and the fitted curve on the Tektronix screen. Do similar scan in X at Y value determined by first fit, fit gaussian, type out co'ords of fitted center and enter them and (name) into record 1 of STARS file. If you are keeping old data in records 1 through n of stars file, you can start at record n + 1 thus N+1 ENTER (name)

Otherwise

Center over mark
1 ØENTER (name)

(name) is ⩽ 6 characters long and is not used in programs. Enters (name) and curren PDS coordinates

```
TAYLOR LOAD OK
PROCESSING LOAD OK
LINK-CLEAR OK
1 ENTER SMALL
```



Example of ENTER

ROYAL GREENWICH OBSERVATORY PAGE 1 12/06/78

STARS FILE

#	NAME	X	Y	X	Y	X	Y
1	FLU						
2	FRU	-24778	2	-24767	-11		
3	FRL	-24634	-24620	-24627	-24604		
4	GALAXY	-9478	-11933	-9547	-11918		
5	STAR1	1426	-4209	1356	-4182		
6	STAR2	-22561	-5337	-22626	-5310		
7	STAR3	-21128	-18525	-21180	-18498		
8	HII#1	-10825	-11999	-10891	-11983		

OK

↑ ↑ ↑ ↑ y-coord These columns not used

Example of STARS ALL

into record 1 of STARS file. If you are keeping other data in the first n records of STARS file, you can start at record n+1, thus n+1 $ENTER (name)

Then
If you wish to check contents of STARS file at any time — STARS ALL — Follow by two carriage returns. Page will automatically be cleared when return key is pressed for a second time.

Proceed to next mark

If gaussian centering required — 2 ENTER (name) — (or n+2 ENTER (name) if you started at n+1)
Otherwise — 2 $ENTER (name) — (or n+2 ENTER (name) if you started at n+1)

Then carry on, using ENTER, or $ENTER to enter consecutive records of the STARS file, until all marks have been entered. Some marks may be entered using ENTER and others using $ENTER.

After all marks have been entered, move the PDS to the position of the scan (or the first scan, if more than one) that you are going to do, and enter the position in the STARS file in the next record, using $ENTER (eg 4 SENTER SCAN). Enter all the scans that you wish to re-scan on the second plate in consecutive records of the STARS file using $ENTER. The point that defines a scan can be its center or its center or any of its corners.

To do the scans go to [4.7] or [4.12]
for tape and then after scanning go to [4.6].

[4.6.] Aligining second plate

You wish to do a scan or scans which will correspond pixel for pixel to a scan done on another plate, whose fiducial or stars have been entered in the STARS file. The new plate must be already roughly aligned by eye with the first one, within a few degrees, (by aligning fiducial marks or stars it should be possible to rotate the plate to within about 5^{o} or the other: this is quite adequate.) The program uses the approximation sin $\theta \approx \theta$.

[4.6.1] Position over first mark ENTERed on first plate

If gaussians to be fitted 1 FIRST — Initializes rotation variables. Scans in Y and X as in ENTER, displays results as in ENTER, but stores final coordinates in second set of X, Y columns in STARS file. If for first mark on first plate you used a record other than record 1 (ie n+1 ENTER (name)) use the same record here (ie n+1 FIRST).

Otherwise — 1 $FIRST — Initializes rotation variables Enters current coordinates in second set of

X,Y columns in STARS file.
The record number used should be the same as for the corresponding mark on the first plate, ie if (n+1) ENTER (name) or n+1 $ENTER (name) was used, n+1 $FIRST should be used here.

NOTE: The records in the file used for the alignment must be consecutive, but if the first mark cannot be found, find another and use m FIRST where m is the appropriate record number, then type Ø R# ! (or n R# ! if first mark is record n+1) before using NEXT (see below).

Then [4.6.2] NEXT

PDS will move to a position corresponding to next record in the STARS file, plus an offset of the coordinate origin calculated from the 2 sets of coordinates in the record for which FIRST or $FIRST was used; ie where the next mark ought to be if the angle of rotation were zero.

Move the PDS to be correctly over the mark

In the first iteration the error may be quite large but in later iterations the position reached after NEXT should be close enough to require only a little 'tweaking up'

If gaussian fit required

OK

Scans, fits, displays, stores in STARS file (second set of columns). Record number was incremented by NEXT.

Otherwise $OK

Stores current coordinates in STARS file (second set of columns). Record number was incremented by NEXT, so will be correct automatically

Then loop back to [4.6.2] until all marks have been stored.

Then n m ROTATE

STARS file records n through m inclusive are used to fit a transformation between the two sets of coordinates. If first record was record 1 and 3 marks were entered,
1 3 ROTATE

ROTATION
X2 Y2 ANGLE
23.4 -14.2 3.760
12.0 15.0 0.218
137

Output from ROTATE
The number immediately under ANGLE represents the angle of rotation in degrees X2, Y2 are offsets and second, third rows are residuals. The only number the user needs is the angle.
+ve angle ≡ rotate clockwise (platten)
If angle is large, accuracy in rotation is not important due to the approximation in the transformation

Rotate PDS stage through the appropriate angle

Iterate by returning to [4.6.1], repeat until angle is sufficiently small.

Then NEXT

This will take the stage to the appropriate scan position (ie the position corresponding to the scan location entered for the first plate). The second scan may now be done into a different disk or tape file. (see [4.7] or [4.12]). If several scan positions were stored in STARS file for the first plate, repeating NEXT will move stage to the corresponding positions, so all the scans may be done in turn.

[4.6.3] Additional ALIGN commands

y x GO — Where y, x are 32-bit PDS coordinates; moves stage to these coordinates.

n 1GO — moves stage to coordinates given by record n of STARS file (1st set of coordinates).

n 2GO — moves stage to coordinates given by record n of STARES file (2nd set of coordinates).

Graphics mode: The data points for the gaussian fit in ENTER, FIRST, OK are plotted out as crosses. ("POINTS mode") Alternatives are listed below, using the word sets the plotting mode until another of the words is used or the graphics are re-loaded.

POINTS

FOLLOW

HISTOGRAM

The commands AXES and BIG GRID will plot axes and a grid on the screen.

[4.7] Scanning

[4.7.1] Position PDS over point defining first scan (if ALIGN has been used, this will already have been done.) This point may be the center or any corner.

If you wish to zero the PDS coordinates at any time, this can be done by the command ZILCH or by pressing the appropriate buttons on the console displaying the coordinates.

If you wish to scan in Y

VERTICAL	Sets Y-scan flag

To return to scanning in X later,

HORIZONTAL	Clears Y-scan flag

Otherwise: Scanning in X is default; it is not necessary to type HORIZONTAL unless VERTICAL has been used earlier.

To set PDS velocity

Either	n MM/S	Sets velocity in mm s^{-1}
or	m VEL !	Sets velocity in PDS units (m = 255/60 n)

To set scan mode:

Either	0 ' ODD !	Default. Scans in 1 direction only (increasing X, or Y if VERTICAL)
or	1 ' ODD !	Scans back-and forth, and flips rows in working storage so final result is same.

Then set up scan parameters as follows:

PLATE	Makes PLATE the current image (see [0.8]).
ρ κ WIDE	Size of scan; ρ rows with κ pixels in each row. Sets parameters $\widetilde{WN}$, and sets $\widetilde{CN}$ equal to $\widetilde{WN}$.
δx δy DXDY!	Sets sampling interval in microns δx is interval in x-coordinate, δy is interval in y-coordinate (sets parameters $\widetilde{DX}$)

If you wish to check the above parameters

STATUS

Ignore output column marked 'ORIGIN'

If you wish to define the scan by its center

Position over center
CENTER

Stores origin (ie coordinates of most -ve corner) by setting parameters CØ, using current PDS position (x,y) and $\widetilde{WN}$ and $\widetilde{DX}$, Such that $x_o = x - \frac{1}{2} w_x \delta x$

$y_o = y - \frac{1}{2} w_y \delta y$

where (w_x, w_y) are (κ,ρ) or (ρ,κ) depending on scan direction.

[4.7.2]

Otherwise define the appropriate corner.

	Position over corner	
Either	UR	Upper right corner: default
or	UL	Upper left corner
or	LR	Lower left corner
or	LL	Lower left corner

These define corner of scan, as seen on PDS viewing screen, not on plate itself. They modify the signs of PLATE parameters DX.

Then CORNER

Stores current PDS coordinates in PLATE parameters $\widetilde{C\emptyset}$. This position will be the first pixel (i = Ø,j = Ø) of the scan

If you are scanning to tape using SCANSALOT LOAD, return to [4.12]

Otherwise go to [4.7.2]

[4.7.2] Scanning to disk

SOURCE

Specifies PLATE (the current image) as the source for the pass (scan operation). The parameters set up in PLATE will be copies to INPUT. SOURCE leaves INPUT as the current

ROYAL GREENWICH OBSERVATORY PAGE 1 12/06/78

IMAGES INDEX

#	NAME	LOC'N	ORIGIN:X	Y	DIR.	CORN.	SIZE:X	Y	+X	+Y
1	CART2	485	0.	0.	H	UR	128	128	10	10
		516	640.	640.						
2	CART3	520	-1920.	-1920.	H	UR	64	64	60	60
		527	0.	0.						
3	COPY	470	0.	0.	V	LL	64	64	10	10
		477	-320.	-320.						
4	JCM	400	-2562.	-2560.	H	UR	128	128	40	40
		431	-2.	0.						
5	LONG	600	0.	0.	V	UR	512	4	10	10
		603	20.	2560.						
6	MGD01	457	-1000.	1000	H	UL	32	32	10	10
		458	-1160.	1160.						
7	N1068#1	480	-480.	-501.	H	UR	50	50	20	20
		484	14.	-1.						
8	PK	461	0.	0.	H	LR	64	64	10	10
		468	320.	-320						
9	TEST	459	-1000.	1000.	H	UL	32	32	10	10
		460	-1160.	1160.						
10	TWIN	432	0.	0.	V	UR	64	64	10	10
		439	320.	320.						

OK

(Name) | Starting block / Finishing block | x origin / x center | y origin / y center | Horizontal or Vertical scan? | Which corner? (CENTER>UR) | X pixels (K) | Y pixels (P) | δx microns | δy microns

Example of LISTFILE

image (ie use of WIDE etc will modify INPUT parameters, not those of PLATE.

LISTFILE (cr) (cr)

LISTFILE must be followed by two carriage returns, it automatically clears the page (on the Tektronix) It prints out index of dat for current disk files (see example attached) If the index takes more than 1 page, press any key except PAGE to clear page.

Find highest block number used

This is highest number in column marked LOC'N

Store your image starting in a block higher than this (usually the next block)
Allocate the file by:-
n INTER (filename)

Make an attempt to fill up empty disk areas. 1 Block takes 512 pixels (32 x 32 array ≡ 2 Blocks.)
n is starting block number ($500 \leqslant n \leqslant 2435$)
(filename) is a string of $\leqslant 8$ characters.
This creates a file accessed by the command FIND (filename)

If there are old files in the index that are no longer used,

REMOVE (filename)

Removes entry from index; Files should be removed as soon as they are finished with. This command can be used at any time.

If you have done a PROCESSING LOAD (or do one at this point) and wish to use the averaging capability for this scan, go to [4.7.3]

Otherwise

FIND (filename) EQUAL DESTINATION

FIND (filename) selects the file (the one just created, usually) EQUAL copies INPUT's parameters to DISK and DESTINATION specifies DISK as the OUTPUT image, copying parameters form DISK to OUTPUT.

PASS

Passes INPUT to OUTPUT. Since INPUT is PLATE and OUTPUTis DISK, PASS will do a scan according to the parameters set up, and store the data on disk, Remember to select AUTO on PDS

If you are using ALIGN and now wish to align second plate, go to [4.6].

If you wish to do further scans, go to [4.7]

For further capabilities in PROCESSING LOAD see [4.8]

[4.7.3] Averaging

FIND (filename)
a ' AV !

Sects disk file
Stores a (number to be averaged by, $a = 2^n$) in AV

AVBY DESTINATION

AVBY takes INPUT parameters, multiplies or divides by AV where appropriate, and puts them in DISK's parameters. DESTINATION specified DISK as the OUTPUT (copies DISK's parameters to OUTPUT)

AVERAGE

Modified PASS - averages in i and j scans a line, puts this in working storage. Takes first AV numbers, adds each into first location of "sum buffer"; next AV numbers into second location, etc. Before being added in, numbers are divided by AV^2 2nd, 3rd... AVth scan line added in and then sum buffer stores as 1st row of output array, and sum buffer zeroed. Repeated for every group of AV scan lines, so final array will have each number being the average of AV^2 numbers; eg a scan 64 64 WIDE with 4 in AV will result in an output array of size 16 by 16 with sample size multiplied by 4 in each direction (ie 1Ø 1Ø DXDY! Will have an output dx, dy of 4Ø, 4Ø).

If you are using ALIGN and now wish to align second plate, go to [4.6]

If you wish to do further scans, go to [4.7]

For further capabilities in PROCESSING LOAD see [4.8]

4.8] Further capabilities in PROCESSING LOAD - combination of disk images

For two disk files (file 1) and (file 2) to be arithmetically combined to form a third file (file 3) use the following instructions

FIND (file 1) SOURCE
FIND (file 3) DESTINATION

(file 1) is identified with INPUT
(file 3) is identified with OUTPUT

```
                                  (  ADD
FIND (file 2)                     (  SUBTRACT                  (file 2) is identified with OPERAND by
                                  (  DIVIDE                    using one of the words.

PASS                                                           PASS combines data in INPUT with data in
                                                               OPERAND and places the result in (file 3)
```

The commands defining the OPERAND are

```
ADD         giving (file 3)  =  (file 1) + (file 2)
SUBTRACT    giving (file 3)  =  (file 1) - (file 2)
DIVIDE      giving (file 3)= NUM x (file 1)/(file 3)
```

where NUM is a variable (default 1000) which may be set by n NUM !

Each element in the INPUT array is combined with the corresponding element in the OPERAND array row by row in working storage. The INPUT and OPERAND files are unaltered.

If you wish to overwirte (file 1) with the result then

```
FIND (file 1) SOURCE DESTINATION
              ) ADD
FIND (file 2)) SUBTRACT  PASS
              ) DIVIDE
```

If , however you wish to overwrite (file 2) with the result, then

```
FIND (file 1)  SOURCE
FIND (file 2) DESTINATION
ADD     )
SUBTRACT) PASS
DIVIDE  )
```

[4.8.2] Sum and average of an array

SBAR will claculate the average value of an array and both print it out and store it in the variable NUM. The sum of the array is stored in the double precision (32-bit) variable XSUM, to see value: XSUM 2@ D. To subtract a constant background value n throughout the calculation (while not altering the array itself) use the command

```
n BKG !
```

before using SBAR; default BKG is zero, after changing it, it will stay at current value until altered again or until PROCESSING is reloaded.

```
FIND (filename) SBAR AVERAGE VALUE = 1025          Calculates sum and average
```

[4.8.3] Normalization of arrays

For example, for removing variations due to photocathode sensitivity on an electronograph, divide scan array by scan photocathode map and multiply through by average of latter scan.

Suppose first scan is in a disk file called STAR, the photocathode map in a file called SKY, and you wish to place the result in a file called FINAL, which you first need to create with size equal to those of STAR and SKY, starting at block n.

LISTFILE	See [4.7.2] for details
FIND STAR	Makes disk current image with STAR's parameters.
n INTER FINAL	Enters FINAL in index file with parameters equal to STAR.
FIND SKY SBAR *AVERAGE VALUE = 952*	Stores average value of SKY by NUM.
FIND STAR SOURCE FIND FINAL DESTINATION	Divides STAR by SKY
FIND SKY DIVIDE PASS	multiplying through by NUM and putting the result in FINAL.

If you wish to do further DIVIDE's with NUM at the default value

1ØØØ NUM !

[4.8.4]

NOTES (i) If you wish to use only part of the image, ie a subset of window, see notes on windowing in [4.11]

(ii) Words like ADD, DIVIDE, SUBTRACT do not do anything themselves except assign an operation called PROCESS which is executed for each row in PASS. At the end of a PASS, a default operation which does nothing is re-assigned so that subsequent PASSes will not perform any operation. This default operation is called NOTHING; using it immediately after ADD, etc will cancel the effect of the latter.

These words also assign the current image to be the OPERAND (they include the word SECOND).

[4.9] REMOVAL LOAD/SLICING LOAD

SLICING OPT LOAD loads basic display and defect removal routines. (disk files only)
REMOVAL LOAD loads the above, plus routines to

(1) Makes PDS move to appropriate location for direct visual examination of defects
(2) Make available a file of defect coordinates
(3) Fit gaussians to scan lines

If you only want the basic SLICING routines

SLICING OPT LOAD Go to [4.9.1]	Loads limited options

Otherwise, for REMOVAL

REMOVAL LOAD Go to [4.9.1], continuing through to [4.9.2]	Loads complete package

[4.9.1] SLICING

To view an array line by line:-

FIND (filename)	Select appropriate disk file.
FIRST	Displays first line, sets graphics scalings and row sets no R to Ø.
N	Displays next line. Repeat N for consecutive lines clearing pages as required.

[If response is *OK* with no lines plotted, file is complete]

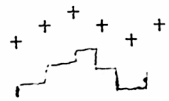
Subsidiary Display words:-

PLOT	Replots current line. Useful after clearing page.
n ' R + !	Skips n lines
(n-1) ' R ! N	Plots line n
m n X SIZE	Resets x-scale so that pixels m ⩽ i ⩽ n are plotted
m n Y SIZE	Resets Y-scale so that density values m through n are plotted. (Default to 0.4000)

POINTS — Sets Plotting mode

HISTOGRAM — Sets Plotting mode

FOLLOW — Sets Plotting mode (Default.)



To removed defect, plot line and

[4.9.1.1]

FIX — Puts up cursors
Position cursor on L.H.S. of defect
Tap any key (except RETURN)
Position cursor on R.H.S. of defect — Program does linear interpolation between 2 points.
Tap key

[New interpolated line is stored in the disk file and replotted on screen. Old line is lost].

These are all the available words in SLICING.
If you have loaded REMOVAL continue to [4.9.2]

[4.9.2] REMOVAL

This assumes that source of disk file is still on PDS and co-ordinate system has not been changed.

[4.9.2.1]

Plot lines until possible defect appears — If desired clear page and PLOT

If you wish to enter removed-defects co-ordinate into DEFECTS file to initialize that file then

?DEFECTS (or) (or) — Displays cursor and contents of DEFECTS

LINK-CLEAR — Initializes file

Otherwise

NOX — Puts up cursor
Centre cursor over apparent defect — This position is not used in any subsequent defect removal and hence position of cursor is not critical.

Tap any key (except RETURN)

PDS will move to co-ordinate corresponding to that point and the user may decide from visual appearance of area whether removal of feature is desired.

If defect to be removed go to [4.9.2.2]
Otherwise go back to [4.9.2.1] and locate another defect candidate

[4.9.2.2] If defect to be entered into DEFECTS file go to [4.9.2.3]
Otherwise use FIX as described before in [4.9.1.1]

[4.9.2.3] NIX

Identical to FIX except, stores current PDS co-ordinates in DEFECTS file. (Do not move PDS between NOX and NIX otherwise wrong co-ordinates will be entered.)

To view DEFECTS file

?DEFECTS

Defects are entered sequentially numbered after last defect in file.

To remove further defects go to [4.9.2.1]

If a second plate is then set up using ALIGN so that the co-ordinate translation has been calculated using ROTATE

n @DEFECT

This will move PDS to position corresponding to coordinates in record n of defects file, with the translation added.

In this way the user can see if the same defects are present on different plates.

If no ROTATE has been performed since loading TAYLOR then this instruction will take PDS to n^{th} defect of existing plate.

[4.9.2.4] Fitting a Gaussian to a Scan Line

Plot Line by using (N or PLOT)

APPROX FIT

Calculates lst approximation and does least squares fit

CURVE

Plots fitted curve

RESULT

Outputs values of parameters XØ

CENTERING

CENTER	*PEAK*	*BASE*	*HW*
I.52	*3ØØØ*	*257*	*3.42*
Ø.ØØ	*Ø*	*Ø*	*Ø.ØØ*
Ø			

(value in column CENTER) is distance from center of curve in units of $\frac{1}{10}$ μ (i.e. if result is x, pixel number in line is $\frac{x}{10\ \delta x} + \frac{1}{2}K$ where K is pixel length of row)

PEAK Is height of the peak above the base

BASE Is the base. Units for these are the same as the scan data

($Ø \leqslant$ density $\leqslant 4095$)

HW Halfwidth, units $\frac{1}{10}$ μ. Second line gives residuals

Repeat FIT, CURVE, RESULT as required, for example:

FIT CURVE FIT FIT CURVE FIT RESULT CURVE etc

[4.1Ø] CONTOUR LOAD

[4.10.1] FIND (filename)

Selects disk image to be plotted

If only a part (window) of the file is to be plotted, see [4.11] to do this.

```
SCREEN n WIDTH
```

sets up the screen to be n wide by $\frac{3}{4}$ n high (in pixels)

So, for an m x m image-
$\frac{4}{3}$ WIDTH Will set the scale so that all the file will be plotted, and the right hand quarter of the screen will be blank.

m WIDTH would fill the whole screen but only plot the first $\frac{3}{4}$ m rows

```
n SPACE !
```

Optional: default 1ØØ
Sets contour interval to be n (i.e. difference of n in the numbers in the array.)

```
low high CLIP
```

Optional: default -32768 32767 CLIP
"low" and "high" are numbers defining cutoff points, for example 2ØØ 4ØØØ CLIP will treat all numbers <2ØØ as 2ØØ, and all >4ØØØ as 4ØØØ

```
DISK PLOT
```

Plots contour map

[4.10.2] Contour plotting directly from tape

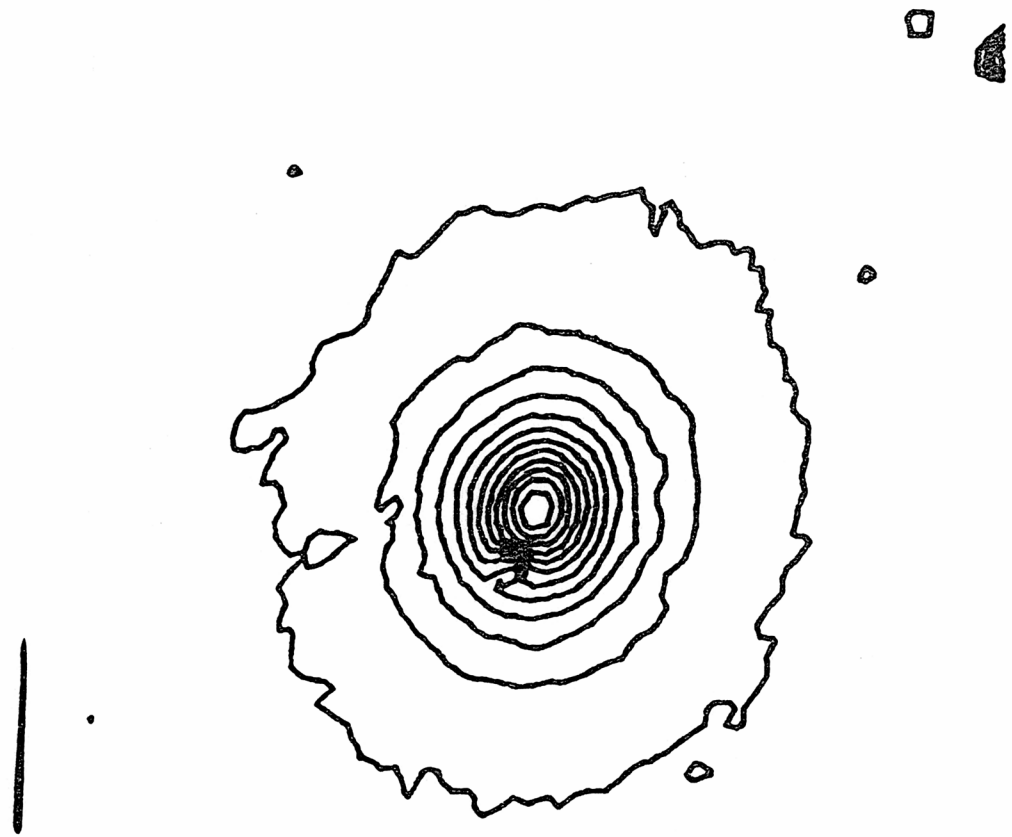
```
512 MAXIMUM SCANSALOT LOAD CONTOUR LOAD
```

This unloads TAYLOR so if you subsequently wish to use programs in TAYLOR you must repeat TAYLOR LOAD.

```
n SKIPS TWIDE
```

Select tape file; described in [4.12]

```
0 1200 CLIP 100 SPACE ! DISK PLOT
```



```
OK
```

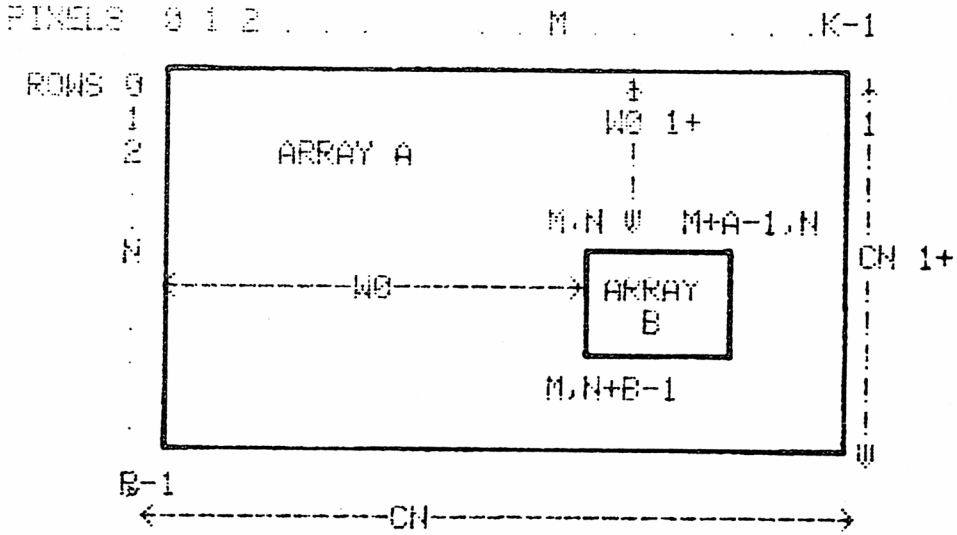
Example of a PLOT using CONTOUR

If windowing is desired, see [4.11]

SCREEN n WIDTH	Described in [4.10.1]
n SPACE ! low high CLIP	" " "
TAPE PLOT	Plots contour map.

[4.11] Disk file manipulation

[4.11.1] Windowing



Given a 2D array Aij of pixel dimension ρ x n (such that $0 \leqslant i < \kappa$ and $0 \leqslant j < \rho$) a subset Bij of pixel dimension a x b (such that $m \leqslant i < (m + a)$ and $n \leqslant j < (n + b)$) may be accessed, provided m and $n \geqslant 0$, $(m + a) < \kappa$ and $(n + b) < \rho$.

This is used for plotting part of a large array, combining part of an array with a smaller array fitting a function to part of an array, etc.

Windowing words are:

n m ORIG	Sets $\overrightarrow{W0}$: defines window origin at row n, pixel m

	59	60	61	62	63	64	65	66	67	68
50	541	554	554	555	554	551	541	551	564	540
51	554	567	562	562	569	570	575	568	565	559
52	567	576	577	591	592	586	587	588	580	584
53	586	592	605	647	635	611	609	607	602	592
54	595	620	642	655	665	655	654	647	632	620
55	634	656	684	691	718	717	717	695	676	665
56	672	700	726	742	765	775	776	764	738	701
57	721	755	789	823	845	836	839	830	794	744
58	758	810	868	917	946	945	933	908	860	797
59	833	884	956	1030	1077	1100	1047	995	937	858
60	888	976	1077	1177	1245	1265	1204	1104	1015	914
61	945	1062	1185	1310	1430	1440	1363	1220	1082	960
62	983	1116	1290	1470	1591	1581	1475	1300	1133	988
63	1008	1152	1346	1532	1686	1669	1516	1326	1154	996
64	993	1162	1333	1500	1649	1614	1473	1295	1118	980
65	965	1109	1260	1397	1486	1458	1356	1211	1064	939
66	932	1037	1035	1048	1242	1276	1203	1093	979	877
67	856	935	891	835	1043	1099	1053	996	914	820
68	799	856	910	955	979	962	927	894	831	760
69	746	791	929	853	859	851	830	795	759	710
70	700	735	756	769	784	770	755	729	702	671
71	665	692	694	706	726	720	699	681	660	644
72	627	647	653	666	676	668	651	644	629	612
73	603	611	617	621	623	625	616	611	596	582
74	582	583	588	595	594	585	587	579	570	549
75	562	564	559	567	567	560	563	565	557	547
76	550	549	546	548	549	548	542	549	546	536
77	537	541	537	541	540	528	527	527	532	532
78	525	529	528	531	528	518	516	517	516	519
79	511	509	516	518	521	513	513	511	509	505

OK

Example of VCILA

b a WW	Sets $\overrightarrow{WN}$, defines size of window to be b rows of a pixels each.
FULL	Sets $\overrightarrow{W\emptyset}$ to (0.0) and $\overrightarrow{WN} = \overrightarrow{CN}$ ie; makes the window the default value, the whole array.

[4.11.2] Digital display

FIND (filename) m n VOILA	Clears page and displays window of file with origin at pixel n, row m. Resets $\overrightarrow{W\emptyset}$, $\overrightarrow{WN}$ to default at end.

[4.11.3] Examples of Copying Files

FIND (file 1) SOURCE	
FIND (file 2) DESTINATION PASS	Copies (file 1) to (file 2)
FIND (file 1) 32 32 WW SOURCE	Copies "top left" (1st 32 pixels of 1st 32 rows of file 1) to corresponding area of (file 2)
FIND (file 2) 32 32 WW DESTINATION PASS	

[4.12] SCANSALOT LOAD

Taping by SCANSALOT LOAD, uses the second terminal's working storage area so the second terminal may not be used while you are using the tape.

[4.12.1] To set up parameters for a scan to tape, go to [4.7.1]

If tape is a new tape (no files on it to be kept)

REWIND	Rewinds tape Initialises tape variables
NTP	Writes 1 dummy record and 2 filemarks then backspaces twice. The record is for compatability with RTSCAN tapes which have no filemark before the first file.

Otherwise ensure tape is positioned at some point before the final double filemark. Program will look for a double filemark before starting to write.

Then WRITE (identifier) (cr)

(Identifier) is a string of ⩽ 40 characters (including spaces). Tape will run forward until a double filemark is found, and will then backspace so that the second filemark will be overwritten. The scan will be performed using the PLATE parameters set previously. The data will be written onto tape in Scansalot format.

After the scan is completed, a double filemark will be written at the end of the file, and the tape will then be backspaced twice, so that a second WRITE will not require any repositioning of the tape.

The rest of this section [4.12] can be used without doing a TAYLOR LOAD, by doing a 512 MAXIMUM SCANSALOT LOAD instead.

[4.12.2] Disk-to tape pass

FIND (filename)
WRITE (identifier)

Selects appropriate disk file. Looks for end of tape (double filemark) backspaces once and copies disk file onto tape. Finishes by writing a new double filemark and backspaces twice.

(Identifier) is ⩽ 40 character string including spaces terminated by carriage return.

[4.12.3] Tape positioning

REWIND		Rewinds tape, initialises tape variables EOFS and REC.
n SKIPS	n > Ø	Passes forward over n filemarks. Remains positioned immediately after the last (i.e. at the beginning of a file.)
	n < Ø	Goes backward over n filemarks then forward 1 space, so positioning tape at the beginning of a file
	-1 SKIPS	Returns to beginning of current file.
TWIDE		Returns to beginning of current file, goes through file checking number of records and coordinates in header, uses these to set TAPE parameters $\overrightarrow{WN}$, $\overrightarrow{CN}$, $\overrightarrow{C\emptyset}$, $\overrightarrow{DX}$. Types first identifier, returns to beginning of the file. n SKIPS TWIDE is the taping equivalent of FIND (filename) for disk files.
T-INDEX		Prints directory of files on tape; rewinds tape and goes forward until double filemark found, then backspaces twice.
n FILE		Positions at end of first record of file n, n is absolute file number with dummy header file being zero. However in some circumstances when interrogating individual records in a file, one goes past the end of file. Subsequent use of FILE may give the wrong result.

```
T-INDEX
 0 DUMMY FILE ;  BEGINNING OF TAPE          2   1   0   0      0.      0.
 1 JCM TEST SCAN 1978JUL26                 64  64  10  10-26860. -3398.
 2 VERTICAL SCAN, ODD=1                    20  32 -12  -5-14010.-11834.
 3 ANOTHER VERTICAL SCAN                   20  32 -12  -5-14010.-11834.
 4 SIMPLE SCAN                             64  64  10  10-26860. -3398.
 5 AVWRITE TEST                            64  64  10  10      0.      0.

 END OF FILES OK
```

ωN ωNΔT DX DXΔT CΦ CΦ ΔT

Example of T-INDEX

	LABEL	Reads and prints next records identifier.
	n RECS	Moves forward n-1 records and reads next record 1 RECS reads next record 2 RECS reads next record but 1 0 RECS reads record just read -1 RECS reads record before one just read
	SHOW	Types identifier and dumps scan data currently in working storage. To be used after RECS.

[4.12.4] Tape to disk pass

	n SKIPS TWIDE	Selects file, prints identifier, sets TAPE parameters
	SOURCE	Copies TAPE parameters to INPUT
If disk array not yet allocated	LISTFILE	See [4.7.2] for explanation, lists disk fil
	n INTER (filename)	See [4.7.2] for explanation, enters disk fi
	FIND (filename)	Selects disk file
	DESTINATION	Copies DISK parameters to OUTPUT
Otherwise	FIND (filename)	Selects disk file
	EQUAL DESTINATION	Sets DISK and OUTPUT parameters
Then	PASS	Copies tape file to disk file

[5] SMOOTHING (boxcar convolution)

[5.1] Averaging in i and j by constant factor a; disk arrays only.
Description: let contents of pixel i, row j in array A be Aij

A is source array, B is smoothed array
Each array consits of ρ rows of K pixels each
$[\chi]$ means integer part of χ

Action of program:

Bij = Aij for (i) $\quad \emptyset \leqslant j < [\frac{1}{2}a],\ \emptyset \leqslant i < \kappa$

(ii) $\quad \emptyset \leqslant i < [\frac{1}{2}a],\ \emptyset \leqslant j < \rho$

(iii) $\quad K - \left[\frac{a+1}{2}\right] + 1 \leqslant i < K,\ \emptyset \leqslant j < \rho$

Partially smoothed s.t. $Bij = \dfrac{1}{a(\rho - j + [\frac{a}{2}])} \displaystyle\sum_{l=j-[\frac{a}{2}]}^{\rho-1} \sum_{k=i-[\frac{a}{2}]}^{i+[\frac{a+1}{2}]-1} A_{kl}$

for $[\frac{1}{2}a] \leqslant i < K \left[\frac{a+1}{2}\right] + 1,\ \rho - \left[\frac{a+1}{2}\right] + 1 \leqslant j < \rho$

Smoothed s.t. $Bij = \dfrac{1}{a^2} \displaystyle\sum_{\rho=j}^{j+\left[\frac{a+1}{2}\right]-1} \sum_{k=i-\left[\frac{a}{2}\right]}^{i+\left[\frac{a+1}{2}\right]-1} A_{kl}$

for $[\frac{1}{2}a] \leqslant i < K - \left[\frac{a+1}{2}\right] + 1,\ [\frac{1}{2}a] \leqslant j < \rho - \left[\frac{a+1}{2}\right] + 1$

Inside fully smoothed region:

For odd values of a - Bij is the average of the a x a square of points in array A with center (i, j)

For even values of a - Bij is the average of the a x a square of points in array A with center (i - $\frac{1}{2}$, j - $\frac{1}{2}$) so for even a the image is translated by $\frac{1}{2}$ pixel in i and j. If an array is to be repeatedly smoothed by an even factor it is recommended that the array be 'flipped' each time before smoothing so that Aij → A(K-i-1) (ρ-j-1). Thus the successive translations will cancel instead of adding. How to achieve this is described below.

[5.2] Tape files

This program only works on disk files, to smooth a tape file first copy it to disk, either 512 MAXIMUM SCANSALOT LOAD or TAYLOR LOAD SCANSALOT LOAD, and then go to [4.12.4] for instructions on tape-to-disk pass.

[5.3] Loading

a SMOOTHING LOAD *5I2*

a is the full width of the boxcar in p and k (a ⩾ 2).
Output is maximum row size allowed. This is 512 or less depending on value of a. For a > 8 the buffer sizes must be < 512 to fit in **core**.

[5.4] Disk files

Allocate all disk files you will require using

LISTFILE

Lists images index, for explanation see [4.7.2]

FIND (file 1)

Sets current image parameters to those of (file 1). (file 1) is the source file i.e. the one you wish to smooth.

	n INTER (filename)	n is block number, select as described in [4.7.2]. INTER allocates a file called (filename) the same size as (file 1). Use INTER to allocate all the files you will need.

[5.5] Smoothing

(i) Single smoothing	FIND (file 1) SOURCE	Defines (file 1) as the INPUT file, i.e.: file to be smoothed
	FIND (file 2) CONVOLUTE	Copies (file 1) to (file 2) and smooths (file 2). CONVOLUTE is equivalent to DESTINATION PASS $CONVOLVE (see below).
(ii) Smoothing into same file	FIND (file 1) $CONVOLVE	Smooths (file 1), overwriting original data. If you wish to save original data, copy it into another file or use CONVOLUTE as in (i) and then FIND (file 2) $CONVOLVE.
	FIND (file 1) $CONVOLVE $CONVOLVE $CONVOLVE	Repeatedly smooths (file 1).
(iii) Flipping and then smoothing		
This is for use with repeated smoothing by even values etc.		
	FIND (file 1) SOURCE	(file 1) will be overwritten later, to save original first copy it.
	FIND (file 2) RCONVOLUTE	Copies (file 1) to (file 2) reflecting it on the way so that pixel (i,j) of (file 1) is put in pixel (k-i-1,ℓ-j-1) of (file 2), i.e. reflected in the center point of the array. Then smooths (file 2).
	SOURCE FIND (file 1) RCONVOLUTE	Flips (file 2) into (file 1) and smooths (file 1).

SOURCE FIND (file 2) RCONVOLUTE
SOURCE FIND (file 1) RCONVOLUTE
SOURCE FIND (file 2) RCONVOLUTE

Repeatedly smooths array, with the ½-pixel translation caused by each smoothing cancelling.

If the final file is (file 1), the array will be the right way round, but if it is (file 2), one more flip must be done to correct it.

SOURCE FIND (file 1) RC-FLIP PASS

Takes current file, flips into (file 1)

6] 3D plot

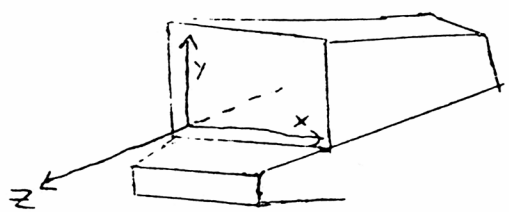
There are 3 variants loaded by:

Either 3D LOAD — Basic FORTH 3D plot
or 3D/J LOAD — Hidden line 3D plot
or 3DSLCW LCAD — Alternative hidden line plot

Then if tape files are to be plotted: SCANSALOT LOAD — Loads taping routines

6.1] Form of plot

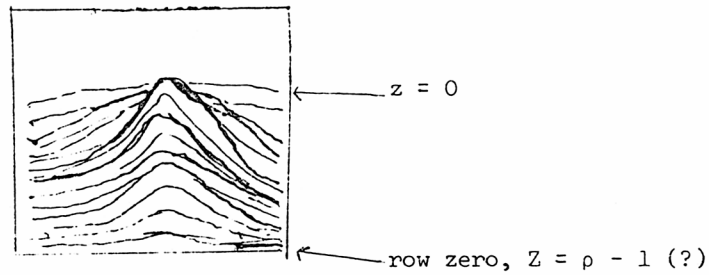
Orientation of axes



X represents pixels/row

Y represents density units (numbers in array)

Z is direction of decreasing row number i.e. later rows are plotted towards the more negative Z.



n m X SIZE sets the X scale, n being the abscissa of the left hand side of the screen, m being the abscissa of the right hand side.

Thus Ø WN @ X SIZE will cause a plotted row to fill the screen from left to right. (WN @ gives the value of K)

n m Y SIZE sets the Y scale

n m Z SIZE controls the spacing between the rows.

Default values: Ø 32 X SIZE
Ø 8192 Y SIZE
- 64 64 Z SIZE

The pitch and yaw of the image can also be altered.

Pitch inclines the Z axis from the horizontal
Yaw inclines the Y axis from the vertical, in the plane normal to Z.
Default pitch is 6Ø°, yaw is zero.

[6.2] Commands

Commands for the three variants are identical except in the case of PITCH and YAW.

SCREEN ρ K WIDE

No more than ρ rows and no more than K pixels/row will be plotted.

For a disk file:	FIND (filename)
For a tape file:	n SKIPS TWIDE

Selects disk file to be plotted.

n is relative file number;
see [4.12] for explanation

Then [6.2.1]

If only a part of the file is to be plotted:

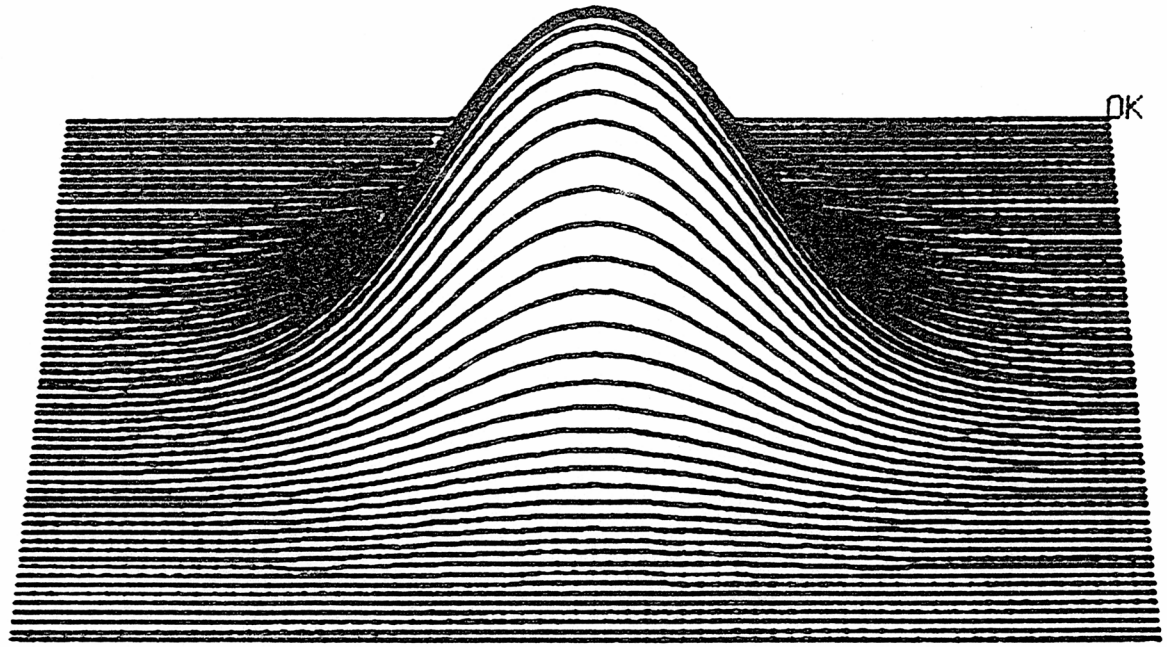
n m ORIGIN	see [4.11] for explanation
b a WW	see [4.11] for explanation
n m X SIZE	Sets up X axis co-ordinates n = abscissa of L.H.S. m = abscissa of R.H.S. (Default 0 32 X SIZE)
n m Y SIZE	(Default 0 8192 Y SIZE)
n m Z SIZE	(Default -64 64 Z SIZE) For explanation see [6.1]

If 3D variant being used

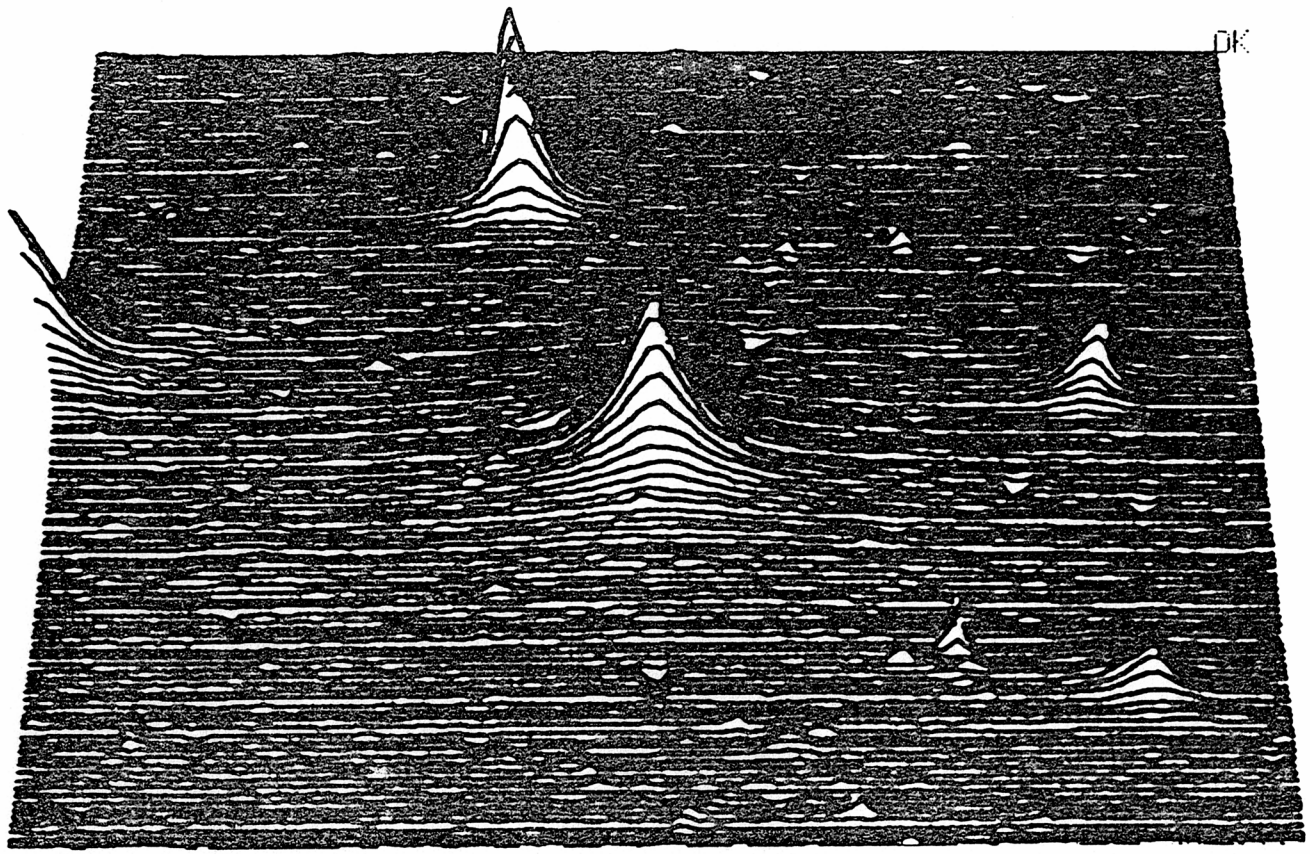
S c PITCH	Default 866 500 PITCH
S c YAW	Default 0 1000 YAW S = 1000 sin θ c = 1000 cos θ

Otherwise (ie 3D/J or 3DSLOW variants)

nn . nn PITCH	Default 60.00 PITCH
nn . nn YAW	Default 00.00 YAW nn.nn is angle in degrees. There must be exactly 2 decimal places - do NOT omit trailing zeros.



a PLOT using 3DSLOW



@ PLOT using 3D/5

Then For a disk file: DISK PLOT — Clears page, plots file

For a tape file: TAPE PLOT — Clears page, plots file

Then to replot using different window, scales, pitch or yaw: go to [6.2.1]

[7] Photometry

Description: Least squares fitting of a 2-D gaussian to scan data from a disk file. The file may be windowed (see [4.11]).

Consider a 2-D array of data Aij where $0 \leqslant i < k$ and $0 \leqslant j < \rho$.

Fitting Parameters		Parameter definition
XØ	$= 6400\ i/_K$	) (XØ, YØ) is centre of gaussian
YØ	$= 6400\ j/_K\ \delta j/_{\delta i}$	) $\delta_{j/\delta i}$ = ratio of sample intervals in microns of the j and i directions.

PEAK Height of peak above base (units as in Aij)

BASE Base (units as in Aij)

HX Half width at half height in i direction (units as in XØ)

HY Half width at half height in i direction (units as in YØ)

SR Correlation. Stored as integer. Value multiplied by 1000.

Function:

$$f(i,j) = (BASE) + (PEAK)\ 2^{\xi} \leqslant C$$

$$\text{where } \xi = \frac{(X-X\text{Ø})2}{(Hx)} + \frac{SR(X-X\text{Ø})(Y-Y\text{Ø})}{HX.HY} + \frac{(Y-Y\text{Ø})^2}{(HY)}$$

$$X = \frac{6400i}{k} \qquad Y = \frac{6400j}{k} \qquad \frac{\delta j}{\delta i}$$

C is a cut-off value (Default 4095)

[7.1] PHOTOMETRY LOAD — loads relevant programs

[7.2] n CUTOFF ! — Sets cutoff value, default 4Ø95

FIND (filename) — Selects disk file

If windowing required, see [4.11]

APPROX — Calculates a first approximation

[7.2.1] FIT — Performs the fit

RESULT — Outputs results.

PHOTOMETRY

XØ	*YØ*	*PEAK*	*BASE*	*HX*	*HY*	*R*
I6.ØØ	*I6.ØØ*	*3ØØØ*	*2ØØ*	*8.ØØ*	*8.ØØ*	*Ø.ØØØ*
Ø.ØI	*Ø.Ø3*	*I4*	*2*	*Ø.23*	*Ø.Ø4*	*Ø.ØØ7*
48						

XØ, YØ, HX, HY output in pixel units
residuals (related to covanance matrix $\sqrt{S^2}$ where S^2 is sum of squared deviations $\sqrt{\rho k-7}$
of <u>previous</u> fit

Loop to [7.2.1] until residuals are sufficiently small.

4 FITS are usually enough; a single FIT takes about 3 mins. for a 32 x 32 file, 9 mins. for a 64 x 64 file.

[7.3] <u>Creating 'Faked' Gaussians</u>

To create a file containing a digitized gaussian which has been defined by a FIT, immediately after FIT or RESULT.

If cutoff value to be changed

```
c CUTOFF !
```

32767 CUTOFF !gives c its maximum value.

If you wish to overwrite the fitted data in a file by its fitted gaussian

```
FAKE
```

For every point in the file the function is calculated and the value is stored in that point of the array.

Otherwise create a new file by:-

```
LISTFILE
```

Explained in [4.7.2]

```
n INTER (file 2)
```

" " "

```
FIND (file 2) FAKE
```

Puts gaussian in (file 2)

[7.3.1] Test gaussians

These can be created by setting up the appropriate parameter values as follows:

```
FIND (filename)
```

Either

```
x_o RSU XØ !
y_o UM YØ !
h_x WØ @ + RSU HX !
h_y WØ 1 + @ + RSU  HY !
```

) xo, yo, hx, hy in pixels

or	xo XO! yo YØ!	xo, yo, hx, hy in scaled units described at beginning of [7]
	hx HX! hy HY!	
Then	p PEAK! b BASE! r SR!	Sets PEAK, BASE, SR
If cutoff to be reset:	c CUTOFF!	where r = correlation x 1000
To check the above entries	RESULT	Ignore 2nd and 3rd rows of output

If you wish to add normally distributed noise:

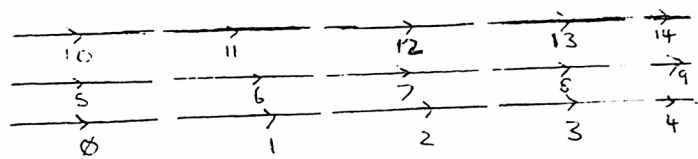
	s SD !	Multiples each value to be stored in the array by $(1 + \frac{\delta(s)}{1000})$ where $\delta(s)$ is normally distributed between $\pm$ 3s with mean zero.
Then	FAKE	Creates gaussian, stores in disk file.

8] Additional Facilities

8.1] Very long scans

For scans with row length too long to fit in the buffer (working storage) the rows must be spilt up into several sections, each section being scanned and stored as a single record on tape (or disk)

Scans as follows:



numbers are record (section) numbers.

Records Ø-3, 5-8, 10-13 are of size equal to the buffer size. Records 4, 9 and 14 are shorter since the total row length is in this case, not an integral multiple of the buffer size. (This example might be: 1Ø24 RLENGTH !3 45ØØ LWIDE). For scans to tape the scansalot header for records other than record Ø will be incorrect.

```
ℓ BIGSCAN LOAD
  427 LOAD
```

ℓ (<2....) is maximum buffer size, loads very-long-scan programs

```
K RLENGTH  !
```

Buffer size to be used, i.e. standard section length, $K \leqslant \ell$. For scans to disk $K \leqslant 512$.

Sets up parameters as in [3.4] except replace WIDE by LWIDF such that

```
ρ K LWIDE
```

ρ is actual number of rows to be scanned, of K pixels each. LWIDE will automatically divide each row into sections.

For scanning to tape: Scan as in [3.5] except replace WRITE by LWRITE such that

```
LWRITE (identifier)
```

scans onto tape as described above

For scanning to disk: Scan as in [4.7.2] except

Replace: FIND (filename) EQUAL DESTINATION PASS

by: FIND (filename) LEQUAL DESTINATION LPASS

[8.2] Multiple Scans

Capability to set up for a number of scans to tape as in BIGSCAN, but instead of doing each scan before setting up for the next, store the parameters of each scan in a file called SCANS file, then a single instruction will cause all the scans to be done in turn.

The following parameters may be different for each scan:

The entire "image parameter field": $\widetilde{\vec{CN}}$, $\widetilde{\vec{DX}}$, $\widetilde{\vec{C\emptyset}}$, $\widetilde{\vec{W\emptyset}}$, $\widetilde{\vec{WN}}$, KIND

i.e. (i) size of scan, set by WIDE

(ii) sample intervals, set by DXDY!

(iii) origin, set by either CENTER or c CORNER where c is UR, UL, LL or LR. Scans may be identified by different corners.

(iv) direction of scan, set by VERTICAL or HORIZONTAL

and also (v) The Scansalot 4Ø-character identifier

[8.2.1]	K BIGSCAN LOAD 428 LOAD	K is buffer size
	CLEAR	Clears scans file
	Set up as in [3.4] for first scan	
[8.2.2]	n ENTER (identifer)	n is scan no. (1 for first scan) Enters parameters and identifer in file
	Set up PDS for n^{th} scan:	
Either	HORIZONTAL	
or	VERTICAL	
or nothing if previous scan was the same direction		
Then	ρ K WIDE	omit if same as previous scan
	δx δy DXDY!	" " " " " "
Either	CENTER	After positioning PDS over appropriate point
or either	UR or UL or LL or LR CORNER	" " " " " "

Then Loop to [8.2.2] incrementing n by 1 each time until all scans ($\leqslant 16$) have been entered.

If tape is a new tape

	REWIND	
	NTP	see [3.5] for explanation
Then	SCAN	Does each scan in turn, writing onto tape, with the appropriate parameters; with a filemark between each scan and two at the end.
	CLEAR	Clears scans file for next user.

[8.3] Array flipping

Capability to flip data in an array on disk
in i : flips each row end for end
in j : flips each column end for end
or in both i and j

[8.3.1]	TAYLOR LOAD	
	431 OPT LOAD	Loads flipping option
	FIND (file 1) SOURCE FIND (file 2) DESTINATION	Allocate file 2 by INTER as in [4.7.2]
Either	ROW-FLIP	Sets up for flip in i
or	COL-FLIP	" " " " " j
or	RC-FLIP	" " " " " i and j

Then KEEP PASS

Updates index file and copies array from (file 1) to (file 2) flipping on the way.

To flip an image scanned with origin at one corner to be as if it was scanned from another, use RECTIFY:

FIND (file 1) SOURCE FIND (file 2) UR DESTINATION RECTIFY

Copies (file 1) to (file 2) flipping it approximately to make the scan as if it was scanned from the UR corner. Using UL, LL or LR instead of UR would cause the final file to have the specified corner as its origin.

To transform an array scanned in Y to one scanned in X:

FIND (file 1) SOURCE FIND (file 2)
DESTINATION TURN

If (file 1) is A and (file 2) is B, the effect of the command TURN is

Aij becomes Bji where subscripts represent row and column numbers.
Due to the simplicity of the method, the operation is prohibitively slow for arrays >(64 by 64)

8.4] Tape file graphic record display

This utility combines SCANSALOT with the display words in SLICING (see [4.9], [4.12]).

454 LOAD

Selects tape file, see [4.12]

n SKIPS TWIDE

FIRST

Sets scales, plots 1st line. See [4.9]

The following commands are available, and are described in [4.9]

N	Reads and plots next line
PLOT	Plots current line
AXES	Define graphic options
BIG GRID	
POINTS	
FOLLOW	
HISTOGRAM	
n m X SIZE	
n m Y SIZE	

There is one additional command:

SAMPLE	Plots every N^{th} point of current line n is set by n SPL ! (default 5). Like PLOT, SAMPLE does not actually read a record, N must be used for that first.

[8.5] Coarse file

Description: Used most often to compare scans from an electronograph with the corresponding area on a cathode map. Disadvantage is that it is not as accurate as ALIGN, but advantage is that cathode map is only scanned once, and file is reused for different high-resolution scans. ALIGN is generally used to eliminate fine variations within a single scan area while the coarse-file method is used to eliminate grosser variations across a larger area of photocathode.

The whole or a large part of the photocathode map is scanned at a coarse spatial resolution and stored on tape or disk.

A second scan at a higher spatial resolution is stored on disk or tape. The program creates a third file, on disk, containing photocathode map points corresponding to those of second scan, the numbers being the weighted average of the 4 nearest-neighbour points in the coarse file.

8.5.1] Creating the coarse file

Do a normal scan as in [3.4], [3.5] or [4.7.1], [4.7.2] except: ZILCH over a standard mark common to all electronographs to be scanned, (for example, upper left fiducial mark). In addition electronographs must be approximately aligned with one another using the fiducial marks or photocathode defects. The scan must be in X and must be specified by either the CENTER or the UR CORNER.

8.5.2] Creating the nearest-neighbour file from the parameters of a scan (called "source scan") done from a corresponding electronograph.

The source scan is done either onto tape or disk. The sample intervals must be less than those of the coarse file. The source scan must be in X and the origin must be defined by either CENTER or UR CORNER. ZILCH must be done at the same standard point as was done for the coarse file.

The nearest neighbour file may now be created using the data from the coarse file and the parameters of the source scan.

	n BIGSCAN LOAD	n is buffer size
	457 LOAD	loads coarse file routines
If source scan on dish:	FIND (filename) SOURCE	Selects source scan file
Otherwise, i.e. on tape:	n SKIPS TWIDE SOURCE	Selects source scan file see [4.12] for explanation
If destination file not allocated in index. use	LISTFILE n INTER (file 2)	See [4.7.2] for explanation
	FIND (file 2) DESTINATION	Selects destination file

If coarse file on disk: FIND (coarse file) SECOND

Sets up coarse file parameters

If coarse file on tape: n SKIPS TWIDE SECOND

Sets up coarse file parameters

Then NEAREST

Calculates and stores nearest neighbour file in (file 2)

To use (file 2) created by NEAREST to take out effects of photocathode response from (file 1) do TAYLOR LOAD PROCESSING LOAD and see section [4.8]

[8.6] Density histogram

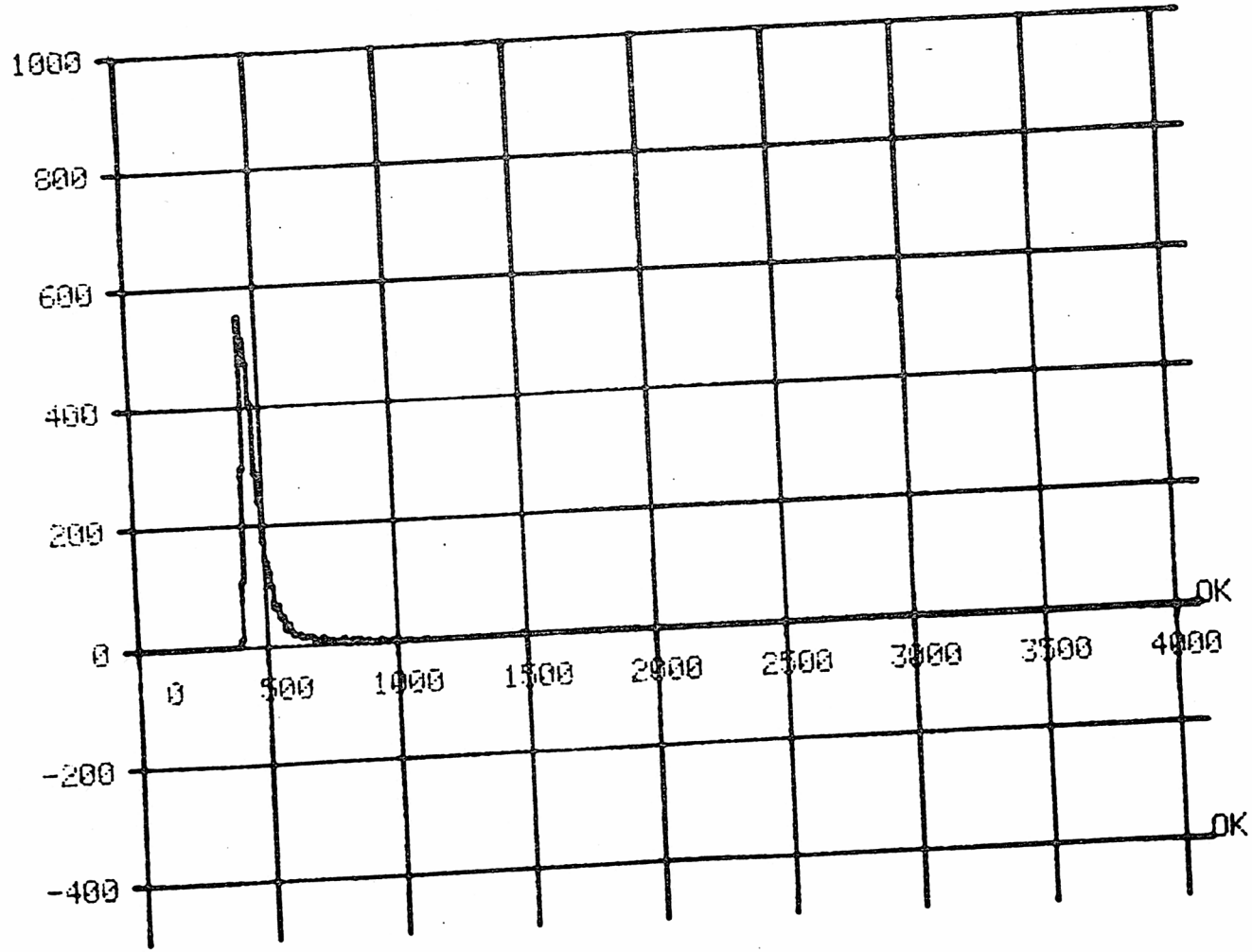
(Disk files only)

DHIST LOAD

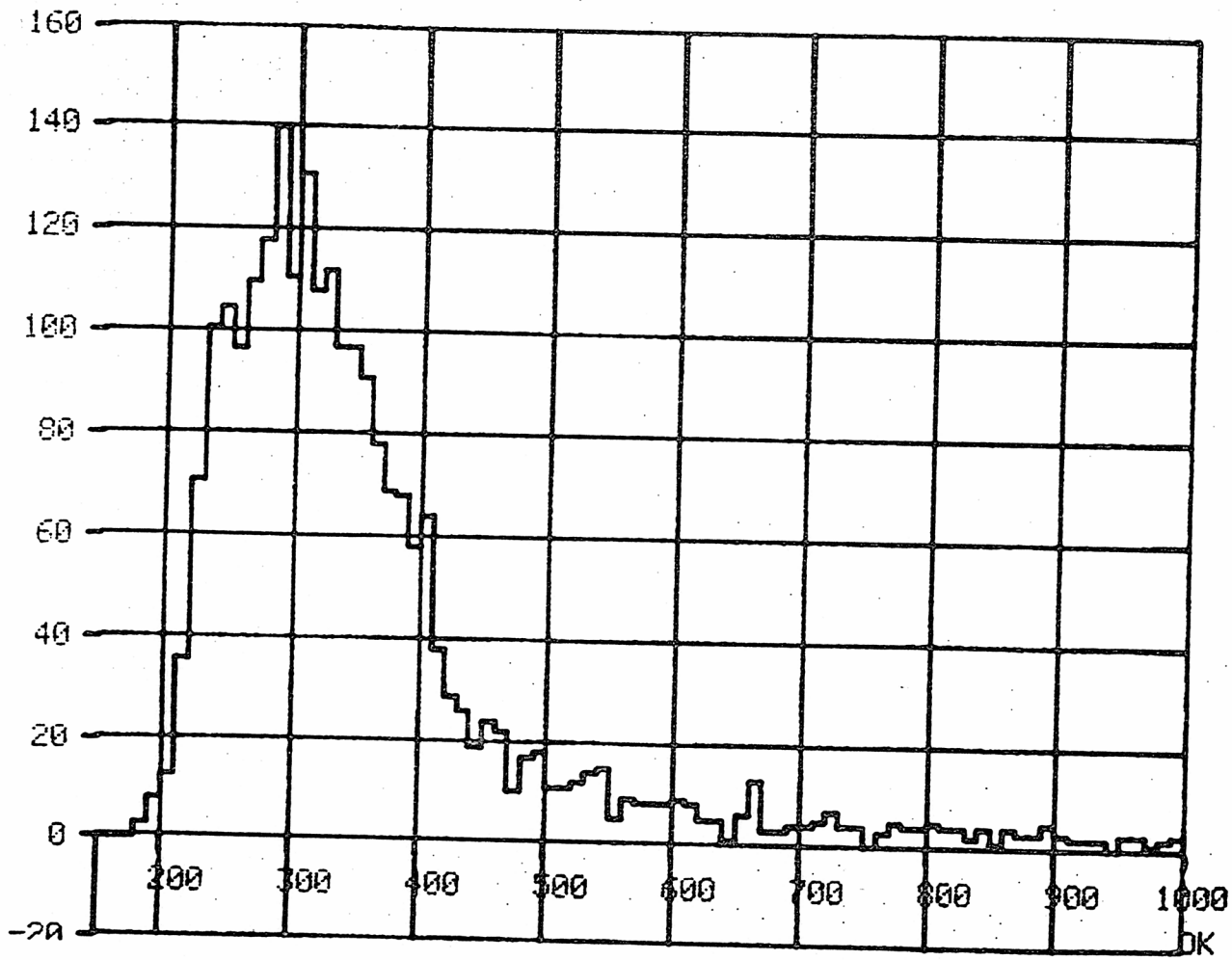
FIND (filename)

DENSITY

Sets up array of 'buckets', goes through file counting number of points in file with densities in range of each bucket, and plots results as a histogram of number against density. The default bucket size is 10 (minimum value is 8, set by n ' QUAN !)
i.e. an ordinate of 200 means that there are 200 points between the abscissa density value and that value +10 (or + the value of QUAN)



Example of DENSITY



Example of PLOT

Data may be replotted as follows:-

a	b	ROI	PLOT
a	b	SEE	PLOT
min	max	Y	SIZE

ENTIRE PLOT

Plots density values a to b only, without change of scale.

Plots density values a to b, expanding scale to fill screen.

Sets Y scale

Equivalent to Ø 4Ø95 SEE PLOT DENSITY plots all values from Ø to 4Ø95