NATIONAL RADIO ASTRONOMY OBSERVATORY

EDGEMONT ROAD CHARLOTTESVILLE, VIRGINIA 22903-2475
TELEPHONE 804 296-0211 TWX 910 997-0174

To: All Spectral Line Manual Users June 14, 1985

From: Lorrie Morgan

Re: Replacement pages for Spectral Line manual

Please replace the following pages in your copy of Spectral Analysis for the NRAO Single Dish Telescopes. The replacement pages are supplied. (The first number shown in each line is the front side, the second number shown is the back side.)

Pages 5-7, 5-8
5-15, 5-16
5-17, 5-18
6-11
7-5
verbs-65, verbs-66
verbs-97, verbs-98
verbs-103, verbs-104
verbs-145, verbs-146
adverbs-17, adverbs-18

Spectral Analysis

for the

NRAO Single Dish Telescopes

Edition 1.0
May, 1985

Table of Contents

11. Utility Programs

Appendices

Index

Introduction

This manual describes the spectral line data reduction program running in Charlottesville on the IBM 4341 and the MODCOMP Classic. Similar versions are run at the NRAO sites in Green Bank, West Virginia and Tucson, Arizona.

This POPS language was originally written by T. R. Cram and J. A. Hudson and its first installation was at the 140-foot telescope in 1975. The data reduction applications have been expanded considerably since then by several people.

The object of the program is to furnish the observer with a complete data reduction package. Standard reduction procedures are built in to the program for the convenience of the observer. Also, the tools are provided for the observer to create his own procedures to tailor the data reduction to his own needs. I am open to suggestions and am happy to provide any assistance to the observer for the most effective use of the program.

Standard reduction routines have been collected together in the chapter titled, "Quick Reference." These routines consist of standard vocabulary that are described in further detail in other chapters of the manual.

This manual is partly a collection of what I thought to be the best features of all of the manuals that have been previously published, namely: <u>On-Site Spectral Line Data Reduction</u> by Thomas R. Cram and Kathy L. Harper; <u>Spectral Analysis for the 36-Foot Telescope</u> by Elizabeth B. Stobie; and <u>CONDAR: Continuum Analysis System for the NRAO Single Dish Telescopes</u> by Lorrie Morgan, Harry Payne, Chris Salter, Betty Stobie, and Bob Vance.

The current edition of this manual is designed in such a way that it will be very easy to update. There will probably be many changes in the program in the near future, so please check with me if there is any doubt about the latest developments and the associated documentation.

Many thanks to Lori Hiner and Bonnie LaVault for their infinite patience while I invaded their offices in order to use the word processor and printer.

I would also like to thank Dr. Morton S. Roberts for his permission to use the contour map of M31 on the front cover.*

Lorrie Morgan
May, 1985

* from Cram, T. R., M. S. Roberts, R. N. Whitehurst. "A Complete High-Sensitivity 21-cm Hydrogen Line Survey of M31." <u>Astronomy and Astrophysics Supplement Series,</u> May 1980. p.p. 215-248.

Chapter 2

POPS: The Interactive Base

POPS (which stands for "People-Oriented Parsing Service") is an interactive, interpretive compiler with various sets of built-in functions that may be extended by the programmer before the program is put into operation, and by the user, afterwards. In addition to supporting spectral line data reduction, POPS is also used to support continuum and off-line VLA data reduction at most NRAO sites.

The single disk spectral line data reduction program (POPS/LINE) consists of an interactive base program and a set of spectral line application routines. Although the compiler portion of the spectal line data reduction program is POPS, the program will hereafter be referred to as LINE, its program name.

When the user types "LINE", the operating system prints a "prompt" character on the CRT screen of the interactive graphics terminal after LINE has finished loading and initializing memory. The user may enter a command by typing it on the keyboard (it will be echoed on the CRT screen). The typed message is completed by striking the RETURN key on the keyboard. Once this key has been struck, LINE examines the typed line for errors. If an error is found, a message is printed and a new prompt character is printed. If no errors are found, the line is processed. How the line is processed depends on the resident LINE "mode of operation".

2.1 Modes of Operation

LINE has three basic modes of operation: EXECUTE, INPUT and COMPILE. These modes are distinguished by how a line of input is treated and by different prompt characters.

EXECUTE mode is the resident mode most often. The program enters this mode when loaded. Entry to each of the other modes is from this mode. Each of the other modes exits to this mode, and an error in a statement in any mode returns the program to this mode.

EXECUTE mode prompts with a caret (>) and as its name implies, immediately executes the commands entered by the user. A session with LINE in EXECUTE mode might look like this:

```
>PRINT 'RADIO ASTRONOMY'
 RADIO ASTRONOMY
>PRINT SQRT (2)
 1.4142
>
```

INPUT mode is a special purpose mode. When in this mode LINE prompts with a number sign (#). This mode is entered by means of the READ command, which has the form READ list , list is a list of variable names. Once this command is entered, LINE prompts with the number sign until the user has input enough values to satisfy the list. For example, a session might look like this:

```
>READ BMARK
#1
>
```

(BMARK is a single-valued application variable.) In INPUT mode, input is treated as a value to be assigned to a variable. Note that once the argument list is satisfied, LINE automatically returns to EXECUTE mode. The reader is referred to the Verb and Operator Dictionary entry for READ.

COMPILE mode is really two modes -- DEFINE and EDIT. Both of these use the colon (:) as the prompt character. DEFINE mode is entered by means of the PROCEDURE statement and is used to create a new user-defined operator. The word PROCEDURE must be followed by an unused name, which becomes the new operator. The program responds to the PROCEDURE statement by prompting with a colon. All statements entered by the user after a colon, constitute the definition of the new operator. The definition is terminated by the command FINISH (entered immediately after a colon) which also returns the program to EXECUTE mode. An additional flexibility of procedures is that their definitions may include logical constructions which cannot be used in EXECUTE mode. These constructions are the FOR, IF THEN ELSE, and WHILE constructions.

A session with LINE using the DEFINE mode might look like:

```
>PROCEDURE EXAMPLE
:FOR I = 1 TO 10
:    PRINT I, SQRT(I)
:    END
:FINISH
>
```

Note that END completes the FOR construction; FINISH completes the procedure definition. Input entered in the DEFINE mode is treated as part of a definition of a new operator. Note that this code is not executed until the user returns to EXECUTE mode and invokes the name of the procedure he has defined. The reader is referred to the Verb and Operator Dictionary entries for PROCEDURE, FOR, IF and WHILE.

EDIT mode is similiar in some respects to DEFINE mode. Statements entered in EDIT mode are treated as part of the definition of a procedure--however, in this case the procedure already exists. EDIT mode is used to change the definition of a user-defined operator. A session using EDIT mode might look like:

```
>EDIT EXAMPLE 3
:     PRINT SQRT(I), I, I*I
:ENDEDIT
>
```

Note that 3 is the line in the procedure EXAMPLE that is to be changed. The first statement entered will replace statement 3; if further statements were entered, they would be inserted between the new statement 3 and 4. ENDEDIT signals the end of the corrections at statement 3 and returns the program to EXECUTE mode. Note also that the editing is of entire lines. To change anything in a line, you must replace the entire line. The reader is referred to the Verb and Operator Dictionary entry for EDIT.

There is also a second edit word, MODIFY. With MODIFY, the user can selectively edit within a procedure line.

2.2 Operators, Constants, Variables, and Expressions

LINE recognizes two basic types of symbols. These are operators and variables. Operators stand for an operation. The program decodes these as instructions to do something. Operators may be arithmetic (+ - * / ** SQRT), inclusion (), logical (= < > ¬ | &), pseudo (IF THEN ELSE FOR WHILE END), or symbolic (PRINT PROCEDURE EDIT LIST STORE).

Symbolic operators may also be application operators, or user-defined operators. User-defined operators are defined in terms of LINE and application operators.

Since an operator is a command to do something, various operators have restrictions on the context or format in which they may be properly used. For example, "=" doesn't mean much without something on both sides of it, and PROCEDURE needs an "Operand", which is a procedure name. Some errors of these types are checked for by LINE, but the user should understand how an operator is properly used.

A variable is an entity that has a value. Variables, like symbolic operators, may also be LINE, application, or user-defined. User variables may be defined only in COMPILE mode. Variables may be either scalars or arrays.

Variables are maintained as single-precision floating point numbers (32 bits). When needed for an array index, the floating point number is rounded down to the next integer value.

Names of variables may be up to 10 alphanumeric characters. The leading character must not be numeric. No imbedded blanks are permitted. No imbedded non-alphanumeric characters are permitted (A*B will be three symbols.) These rules also apply to procedure names (user-defined operators). All names, whether operator or variable, should be distinct. LINE or application names cannot be redefined by the user.

User scalars are defined by a statement of the form A=B, appearing in a procedure. B may be an already defined symbol or an expression consisting of constants and already defined variables and operators. User arrays are defined by a statement of the form ARRAY A(N), appearing in a procedure. The reader is referred to the Dictionary entry for ARRAY for further information.

LINE recognizes two types of constants; decimal numbers written with or without a decimal and fraction, and Hollerith strings, enclosed in single quotes. Decimal constants may appear in expressions, and as operands of operators. Hollerith constants are used as operands of the verb PRINT.

LINE expressions are combinations of a few operators: symbolic, arithmetic, logical and inclusion and variables of any type.

LINE expressions are of two kinds: arithmetic and logical. Arithmetic expressions make use of the operators:

= + - * / **

(shown in order of ascending precedence; those of equal precedence are grouped together). Function calls and subscript evaluation are of the highest precedence (and of the same precedence). The logical operators are:

& %]   < = >

(again grouped according to precedence where & is AND, % is OR and] is NOT. The operations carried out by the = symbol varies with the context in which it is found. It is used as a "store" operator when it appears the first time in any executable LINE statement, except when that statement begins with the 'IF' keyword. The second instance of = in an arithmetic expression, or any instance within an IF statement is taken to be a comparison operation. The group, = , is on the same precedence level as the = (store) operation.

The same rules of precedence apply as in common mathematical notation wherever symbols of inclusion ( ) are encountered.

Operands of arithmetic or logical operators may be real scalars, constant or variable, and also elements of arrays. Arithmetic on Hollerith constants or entire (unsubscripted) arrays is not defined.

The special values "TRUE" and "FALSE" are defined in the language for assigning Boolean states to variables. (Their values are +1.0 and -1.0, respectively.) Any variable may be used as a logical variable; no special declaration is required.

Examples of arithmetic and logical expressions:

```
Y(I)+3*X+ALPHA(M,N)
LOGICAL=(P > Q1BETA(2*J))
IF 3*X< 40 THEN . . .
```

Generally, an expression may be substituted for a variable or constant in LINE statements; the exceptions to this rule are PROCEDURE and ARRAY declarations, FINISH, EDIT, LIST, MODIFY and ENDEDIT. Formats for these statements are necessarily rigid.

2.3 Communication with LINE

As mentioned in the introduction to LINE, commands are entered to the program by typing a message immediately after a prompt character and striking the RETURN key. A message must consist of recognizable symbols to be accepted. (A message containing an unrecognizable symbol will generate the message SYMBOL?, but no other action.) A message may be corrected by backspacing over the error with the BACKSPACE key and retyping the message.

LINE is a linear communicator. It accepts one message at a time. Each time the RETURN key is hit, the input is examined and acted upon. While the program is processing the message, it cannot accept another message. Another message may be entered when the program prints another prompt character. A message is processed in the exact order given. SHOW BASELINE is not equivalent to BASELINE SHOW.

Once "LINE" is typed from the TEKTRONIX, LINE is loaded and running ("ready and waiting"). The program will remain ready and waiting until a command is input. Once the command has been completed, LINE returns to the ready and waiting state. No "initialization" is needed to get LINE's attention, unless the CRT screen is blank, in which case it is a good idea to hit the return key. If the program is loaded and running, it will print a caret (>) on the screen.

A message (one line of input begun after a prompt character and terminated by striking the RETURN key) to the program may contain two basic types of statements. Assignment statements are of the form A=EXPRESSION. Examples are BMARK=1 and ZETA=ALPHA*BETA**2. Command statements are of the form OPERATOR operand(s) ; the operator may or may not require one or more operands. Examples are RESTART, LIST EXAMPLE and EDIT EXAMPLE 3.

In general, each statement (or complete thought) of either type must be entered entirely on one line. (The exceptions to this rule are the logical constructions.)

More than one statement may be entered on one line if the following rules are followed:

(a) An assignment statement must be separated from any other statement by a semicolon (;).

(b) A command statement whose operator is listed in the operator synopsis must be separated from any other statement by a semicolon.

(c) The following operators should always appear immediately after a prompt character: PROCEDURE, EDIT, ENDEDIT, RESTART, STORE and RESTORE.

Examples are:

```
BMARK=1; PRINT BMARK
BMARK=1; ZLINE=0
PROCEDURE EG(A); PRINT SQRT(A); FINISH
```

We note here that application operators (verbs) need not be separated by semicolons, but user-defined operators should be treated as LINE operators.

Chapter 3

Spectral Line: Applications

The spectral line data reduction package is designed to manipulate scans. It therefore provides the user with various operators which do things to scans. These operators are the spectral line verbs, each of which is associated with a distinct data reduction routine. The routines perform four types of operations:

(1) movement of scans,
(2) modification of data values,
(3) calculation and
(4) display of information or data values.

Some verbs only perform one of these functions; others combine two or three.

A scan consists of a set of data values and the identifying information associated with those values. It is this entity, or parts of it, which is manipulated by most of the spectral line routines. All of the autocorrelator channels, regardless of the number of independent spectra being accumulated, are part of the same scan, filed under the same scan number. You may choose to process all the the receivers (spectra) in a scan in parallel, or you may process any one of them at a time. Your choice is referred to as the receiver processing option, and is made by invoking one of a number of verbs.

3.1 Basic Algorithm / Program Structure

The program structure, like the spectral line language (discussed below), reflects its function as a manipulator of scans. In order for a scan to be displayed, used or modified, it must first be copied from the disk, where it was stored by the on-line link program, to the LINE program memory. This memory contains three locations where scans may be kept. These three locations are the WORK, TEMP and HOLD arrays.

Once a scan has been put into one of these three arrays, it can be displayed, changed, moved to another array, or stored in one of the save bins on disk. These subsequent operations are influenced by the array pointer, PTWH. This pointer always has the value 1, 2 or 3, indicating that the "currently referenced array" is the TEMP, WORK, or HOLD array. Since most verbs are defined as operating on the scan in the currently referenced array, it is sometimes necessary and often useful to know which array is current. However, for many simple data reduction schemes, the user will find that the pointer takes care of itself. Briefly, those operations which move data change the pointer to the new location of the data. Most other operations use, but do not change, the pointer.

The user who is designing a data reduction scheme will be most successful if he keeps track of the following: the contents of each program array, the value of the array pointer and the definitions of the verbs. Careful attention to these points will prevent accidental destruction of much-reduced data and enable the user to utilize the program most effectively.

3.2 Verbs, Objects, and Adverbs

The spectral line data reduction package has a language of its own. This language is very similiar to the LINE command language; most of what has been said about the latter applies to the former. We add here additional qualifying remarks.

Verbs are operators; when encountered by the program, the action which is the definition of the verb is performed. Verbs may be said to be of two kinds: those that require an object, and those which have implied objects. Examples of verbs of the first type are

GET scan#
TELL DISK

Each of these verbs requires an object which further specifies the action the verb is to perform. In the first case, scan# specifies which scan is to be copied. In the second case, DISK specifies that a listing of scans in the disk index is desired (the only other permitted objects of TELL are CSTACK, KSCANS, DATA and GPARMS).

Most verbs are of the second type. Examples are

BASELINE
MOMENT
TRH
ACCUM

The definition of verbs of the second type specify upon what the verbs act. BASELINE calculates a baseline for and removes it from the scan in the "currently referenced array" (see "Basic Algorithm/Program Structure"). MOMENT calculates a moment for the scan in the currently referenced array. TRH copies the scan in the currently referenced array into the HOLD array. ACCUM adds the scan in the WORK array to the HOLD array. To use the verbs of the second type successfully, the user should understand what are the implied objects of the verbs. This information is given in the Verb Dictionary. For verbs of the first type, the Verb Dictionary entries describe the required object.

An adverb is simply a variable that is assigned a value which is used by a verb. Examples are the adverbs ZLINE and NBOX. If ZLINE = 1, the verb SHOW will draw a horizontal line at zero Kelvins. BOXCAR smoothes the data by averaging NBOX data points together.

The values of adverbs are set by the program when it is restarted. However, many are set to zero, and the user must specify a meaningful value before using the verb which uses that adverb. New adverb values may be specified at any time and remain the same until the program is restarted or the user changes them again.

Many adverbs are vectors. Examples are CENTER, CLIPMIN and NREGION. CENTER has six values, one for each of up to six gaussian components. CLIPMIN has two values in Tucson and eight in Green Bank, one for each channel. NREGION is doubly dimensioned; it has eight values for each receiver channels.

A single valued adverb is set by a statement of the form:

ADVERB=value

Similarly, all values of a multi-valued adverb may be set equal by a statement of the same form. To set the values of a multi-valued adverb separately, use statements of the form

ADVERB(1)=value1
ADVERB(2)=value2

and so on. Also for convenience, a multi-valued adverb may be set in the following manner:

ADVERB=value,value2,value3,value4,. . . etc.

All subsequent values of the adverb array will be set to zero.

LINE also uses a few variables which are not strictly adverbs, such as SIZE and VRMS; they are used by the verbs MOMENT and RMS to store the results of calculations. They are most often used as operands of the operator PRINT.

3.3 The STACK

The STACK is an array of scan numbers, called ASTACK, that is used by the resident procedures CB, SHOWS, SHOW1 and SHOW2. The STACK can also be used by the observer in his own procedures. The adverb, ACOUNT, is used with ASTACK to denote the number of entries currently in the STACK. The STACK has a limit of 100 scans.

Several verbs alter the STACK. They are listed below.

To empty the STACK, type

    EMPTY

To list the scan numbers currently in the STACK, type

    STACK

To add a scan to the STACK, type

    592 A   or   A 592

To add a contiguous group of scans to the STACK, type

    606 672 ADD   or   ADD (606,672)

To delete an individual scan from the STACK, type

    648 DELETE   or   DELETE 648

The STACK is not used by any verbs except those listed above which define its contents. The purpose of the STACK is to give the observer a means to process several scans once or more in a procedure with minimal effort.

The observer can use the STACK for many purposes other than the current resident procedures. For example, to display the last twenty scans observed, one would define the following procedure:

```
PROCEDURE DISPLAY
     FOR I = 1 TO ACOUNT
          GET ASTACK(I) PAGE SHOW END; RETURN
     FINISH
```

All that is now needed is to define the STACK.

```
6000  6190 ADD
```

When DISPLAY is invoked the specified scans will be displayed on the CRT screen one at a time.

Chapter 4

Synopsis

4.1. Operator and Verb Synopsis

INFORMATION EXCHANGE

*	Begins a comment statement.
HELP	Lists the LINE and application operators, and/or their descriptions. HELP ARRAY lists all arrays, and HELP PROC lists all procedures defined within the program. HELP with no operator lists all the verbs.
PRINT field	Prints the requested field on the CRT. Field may be a literal, or an expression or adverb whose value is then printed.
READ field	Reads the values of the adverbs or variables named from the CRT.
TELL	TELL DISK lists the scans in the disk index. TELL CSTACK prints the stack counter and the number of scans that have been accumulated in the HOLD array. TELL GPARMS prints the current gaussian fit parameters. TELL KSCANS lists the scans in the keep index.
? field	An alias for PRINT.

MEMORY CONTROL

RESTART	Returns the program memory to its original, or default, condition.
RESTORE page#	Copies the indicated disk page into the program memory space.
STORE page#	Stores a copy of the current state of the program memory in the indicated disk page save area.

PROCEDURE

DEFINITION

FINISH	Terminates the definition of a procedure.
PROCEDURE procname	Initiates the definition of a procedure with the given name.
RETURN	Is required in procedures called by other procedures.
SCRATCH procname	Deletes the specified procedure from the program, but does not return that procedure space.

LISTING

HELP PROCEDURE	Lists the names of the defined procedures.
LIST procname	Lists the entire named procedure.

EDITING

EDIT procname line#	Initiates editing of the named procedure at the specified line.
ENDEDIT	Terminates editing of a procedure.
MODIFY procname line#	Modifies the specified line of the named procedure.

LOGICAL CONSTRUCTIONS

FOR x = a TO B BY c	Begins a logical construction like the PL/I interactive DO loop.
IF (test condition)	Begins a logical construction similiar to the PL/I IF-THEN-ELSE.
WHILE (test condition)	Begins a logical construction similiar to the PL/I DO WHILE loop.
END	Terminates the logical constructions begun by FOR, IF and WHILE.

PROCEDURE

SPACE ALLOCATION

ARRAY arrayname( )	Allocates the requested space for an array with the specified name.
CORE	Lists the beginning and last used locations of the procedure and variable definition spaces. Verb Synopsis

BASELINE COMPUTATION AND REMOVAL

BASELINE	Computes and subtracts a polynomial baseline from the currently referenced array.
BMODEL	Constructs a model of the previously computed polynomial baseline in the currently referenced array.
BSHAPE	Computes a polynomial baseline for the currently referenced array.
DCBASE	Computes and subtracts an average data value from the currently referenced array.
RIPPLE	Computes and subtracts a sinusoidal baseline from the currently referenced array.
RMODEL	Constructs a model of the previously computed sinusoidal baseline in the currently referenced array.
RSHAPE	Computes a sinusoidal baseline from the currently referenced array.

DATA TRANSFER

ARRAY-ARRAY

TRH	Copies the contents of the currently referenced array into the HOLD array. (Transfer to HOLD array.)
TRT	Copies the contents of the currently referenced array into the TEMP array. (Transfer to TEMP array.)
TRW	Copies the contents of the currently referenced array into the WORK array (Transfer to WORK array.)

ARRAY-DISK

KEEP	Copies the currently referenced array into the KEEP file to be transferred to tape later.
SAVE	Copies the currently referenced array into a save bin on the disk.

DISK-ARRAY

CFETCH	Copies the most recently completed scan into the WORK array and the associated off scan into the TEMP array.
CGET rcvr#	Retrieves the specified receiver (feed) of the most recently completed scan into the work array.
FETCH scan#	Retrieves the specified scan number into the WORK array and retrieves the associated off scan into the TEMP array.
GET scan#	Copies the indicated scan to the WORK array.
OFF scan#	Copies the indicated scan into the TEMP array.
ON scan#	Copies the indicated scan into the WORK array.
RECALL	Copies the contents of a disk save bin into the currently referenced array.

DATA TRANSFER

TAPE-DISK

LOAD	Reads scans from a 9-track tape and writes them to the raw data file for processing.

DISK-TAPE

MAP	Writes data points to tape to be processed on the IBM to generate position vs. velocity coutour plots.
PIX	Writes data points to tape to be processed on the IBM to make a DICOMED PIX TAPE of a velocity vs. position contour plot on the DICOMED.
PLOT	Writes data points to tape to be processed on the IBM to make CALCOMP plots of spectra.
PROFILE	"Stacks" several scans on a 10 by 25 CALCOMP plot.
REPLOT	Plots a scan on top of the last CALCOMP plot

DISPLAY OF DATA

AUTO	Alias for FREEY.
CCUR	Returns the point value at the current x crosshair position.
CROSSHAIR	Prints the point number and temperature at the current crosshair positions.
DEC	Sets the y-coordinate of maps to declination and x-coordinate to right ascension.
FIX	Alias for HOLDY.
FLAG #	Flags the specified velocity or frequency on the screen.
FREEY	Returns determination of the y-scaling to SHOW.
FV or VF	Sets the x-axis to velocity on the top and frequency on the bottom.
GB	Sets the y-coordinate of velocity-position maps to galactic latitude.
GL	Sets the y-coordinate of velocity-position maps to galactic longitude.
GRID	Draws a rectangular grid on the screen.
HISTOGRAM	Sets the SHOW display mode to histogram.
HOLDY	Fixes the scaling used by SHOW.
LABEL	Labels a contour plot.
LIMIT (#,#)	Limits the SHOW display to only those points between the minimum and maximum values specified.
LINE	Sets the SHOW display mode to a continuous line.
MAPSHOW	Creates a position vs. position contour map of a series of scans.
PAGE	Erases the CRT.

DISPLAY OF DATA

POINTS	Sets the SHOW display mode to points.
RA	Sets the y-coordinate of maps to right ascension and the x-coordinate to declination.
RANGE(#,#)	Sets the y scaling for SHOW according to MIN and MAX values entered with it.
RESHOW	Graphs the currently referenced array on the CRT with no labels or borders, using the scaling last used by SHOW.
RHIST	Sets the RESHOW display mode to histogram.
RLINE	Sets the RESHOW display mode to a continuous line.
RPOINTS	Sets the RESHOW display mode to points.
SHOW	Graphs the currently referenced array on CRT, with labels and borders.
SPREAD (#,#)	Limits the SHOW display to only those channels between the specified values.
TCUR	Returns the temperature value at the current y-crosshair position.
VC or CV	Sets the x-axes to channel number on the top and velocity on the bottom.
VCUR	Returns the velocity value at the current vertical crosshair position.

DISK INDEX

SUMMARY	Produces a listing of the scans on disk belonging to the specified user.
TELL DISK	Lists the scans currently in the disk index.
WIPE	Lists all numbers of users who have data in raw data file and prompts observer for the user number of the observer whose data is is to be wiped from disk.

DOCUMENTATION

HEADER	Lists the header information for the scan in the currently referenced array on the CRT.
STITLE	Prints one line of header documentation on the CRT.
TABLE	Lists the data values for the scan in the currently referenced array on the CRT.

DATA EDITING

CLIP	Limits the data values of the scan in the currently referenced array to within the range CLIPMIN to CLIPMAX.
INVERT	Flips the scan end for end in the currently referenced array.
OVERLAP	Undoes Dual-Dicke frequency switching.
RAPL RAPR	Averages a frequency-switched scan by shifting the data DF MHz, subtracting the shifted data from the unshifted data, and dividing by 2. RAPL shifts to the left, RAPR shifts to the right.

RECEIVER SELECTION

ALL	Selects processing for all of the receivers, and display for receiver A.
ONLYA	Selects processing and display of the A receiver.
ONLYB	Selects processing and display of the B receiver.
ONLYC	Selects processing and display of the C receiver.
ONLYD	Selects processing and display of the D receiver.
PAIR (#,#)	Selects processing for the two specified receivers, and display for the first receiver.
SELECT	Used to select processing for one or more receivers.
SLIDE	Averages two receivers selected by PAIR of the currently referenced array, with weighting by system temperature and integration time. Result is put into the first-named receiver.

GAUSSIAN FITTING

GAUSS	Fits the requested number of gaussians (1 to 6) to the data in the currently referenced array.
GMODEL	Constructs a model of the (sum of the) gaussian(s) fit by GAUSS in the currently referenced array.
GPARTS	Evaluates the parameters and separately displays NGAUSS components of a gaussian.
PEAK	Finds the CENTER, HWIDTH, and HEIGHT of a single gaussian.
REGAUSS	Fits the requested number of gaussians without resetting the first guess of the HEIGHTS to the value at the CENTERS.
RESIDUAL	Subtracts the gaussian(s) fit by GAUSS from the data in the currently referenced array.

MATHEMATICAL

BASIC FUNCTIONS

BIAS	Adds a constant to the scan in the currently referenced array.
DIFF	Subtracts the scan in the TEMP array from the scan in the currently referenced array.
DIVIDE	Divides the scan in the currently referenced array by the scan in the TEMP array.
EQTOGAL	Converts equatorial coordinates (1950 RA and Dec) into galactic coordinates and prints the results on the Tektronix screen.
GALTOEQ	Converts galactic coordinates into equatorial coordinates and prints the results on the Tektronix screen.
RDIFF	Subtracts the second reveiver selected by PAIR in the currently referenced array from the first receiver selected by PAIR and puts the results in the first receiver.
SCALE	Multiplies the scan in the currently referenced array by the constant, FACT.
SLIDE	Averages the two receivers selected by PAIR of the currently referenced array, with weighting by sytem temperature and intergration time, and puts the results in the first named receiver.

SPECIAL FUNCTIONS

DELTAV	Computes velocities at specified fractions of the peak flux.
FFT	Replaces the data in the currently referenced array with its Fast Fourier Transform.
GMEASURE	Calculates the area, high and low velocity peak fluxes within a specified region of spectral values.
MOMENT	Computes the moment requested by the value of the adverb NMOMENT over the points specified by BMOMENT and EMOMENT of the scan in the currently referenced array.
RMS	Computes the root-mean-squared noise over the regions specified for the scan in the currently referenced array.
TEMP	Computes the temperature spectrum from the on scan in the WORK array and the off scan in the TEMP array by the formula (ON-OFF) / OFF * Tsys.

SMOOTHING

BOXCAR	Smoothes the scan in the currently referenced array by averaging NBOX points together and placing the result in the central point.
HANNING	Smoothes the scan in the currently referenced array by averaging three points together, giving the central point twice the weight of either flanking point.
SMOOTH	Smoothes the scan in the currently referenced array with the SMWGT weighting function.

POINTER CHANGE

PMH	Sets the array pointer to the HOLD array. Points to the HOLD array.
PMT	Sets the array pointer to the TEMP array. Points to the TEMP array.
PMW	Sets the array pointer to the WORK array. Points to the WORK array.

STACKING SCAN NUMBERS

A #	Inserts a single scan into the STACK.
ACCUM	Adds the scan in the WORK array to the contents of the HOLD array with no weighting.
ADD (#,#)	Inserts the specified scans into the STACK.
ALIGN	Computes the number of channels necessary to shift the currently referenced array so that its velocities or frequencies will align with those of the HOLD array.
AVE	Divides the HOLD array by the number of scans that have been accumulated.
DELETE #	Deletes the specified scan from the STACK.
EMPTY	Empties the STACK.
SCLEAR	Clears the accumulator (HOLD) array before stacking data.
STACK	Lists all the scans entered in the STACK.
TELL CSTACK	Lists the number of scans accumulated in the HOLD array, as well as the scan numbers of the first 24 and the last scan that were accumulated.

4.2. Adverb Synopsis

ACOUNT	The STACK counter.
ASHIFT (CH)	The number of channels by which a scan must be shifted so that its velocities align with the velocities in the HOLD array. Used by ACCUM, set by ALIGN.
ASTACK(100)	The 100 element STACK array.
BBASE(CH)	The number of channels at the left side of the data which will be used to compute a baseline or RMS.
BDROP(CH)	The number of channels at the left side of the data which will not be used.
BGAUSS	The channel at which GAUSS is to begin to fit.
BMARK	If =1, SHOW will draw boxes to indicate the regions last used to compute a baseline or RMS.
BMOMENT (CH)	The data point at which MOMENT will begin to compute.
BRANGE	Minimum 1950 Right Ascension (in hours) of scans to be loaded from a KEEP tape or telescope tape.
BSCAN	Minimum scan number of scans to be loaded from a KEEP tape or telescope tape.
CENTER(6)	The points at which up to six gaussians have their centers.
CLIPMAX(CH)	The maximum data value desired.
CLIPMIN(CH)	The minimum data value desired.
CMARK(6)	The channels at which up to six vertical lines are to be drawn by SHOW.

DF	The frequency in MHz data is to be shifted by RAPR or RAPL.
EBASE(CH)	The number of data points at the right end of the data which are to be used to compute a baseline or RMS.
EDROP(CH)	The number of data points at the right end of the data which are not to be used.
EGAUSS	The point at which GAUSS is to stop fitting.
EMOMENT(CH)	The data point at which MOMENT is to stop computing.
ERANGE	Maximum 1950 Right Ascenscion (in hours) of scans to be loaded from a KEEP tape or telescope tape.
ESCAN	Maximum scan number of scans to be loaded from a KEEP tape or telescope tape.
FACT(CH)	A number of Kelvins to be added to a scan, or a numerical factor to be multiplied with a scan.
FIXC	If $\geq$ 1, the values of CENTER will not be iterated by GAUSS.
FIXHW	If $\geq$ 1, the values of HWIDTH will not be iterated by GAUSS.
GREGION (rcvr,8)	The channels, in sequential order, at which up to four regions are to be considered by GAUSS.
HEIGHT(6)	The height, in Kelvins, of up to six gaussians.
HWIDTH(6)	The half-width, in number of points, of up to six gaussians.
LEVS(20)	Up to twenty levels at which contours may be drawn by MAPSHOW.
MRATIO	The ratio between the horizontal and vertical scaling of a contour map.
NBOX	The number of data points which are to be averaged together by BOXCAR.
NFIT	The order of a polynomial which is to be fit by BASELINE or BSHAPE.

NGAUSS	The number of gaussians which are to be fit by GAUSS.
NITER	The number of iterations which GAUSS will execute trying to compute a fit, before giving up.
NMOMENT	Denotes the type of moment to be calculated. Currently only the zero order (area) is supported.
NREGION(CH,8)	The data points, in sequential order, at which up to four regions per channel are to begin and end.
NSAVE	Names a disk save bin.
OSHIFT	The number of channels by which part of a Dual-Dicke switched spectrum has been shifted by the verb OVERLAP to undo the effect.
RFREQ	The approximate frequency, in MHz, of the sinusoidal baseline to computed by RIPPLE or RSHAPE.
RSHIFT	The number of channels by which receiver B must be shifted relative to receiver A to align their velocities.
SET1/2/3	Used with SELECT to select processing of one or more receivers.
SINCR	The increment between scans.
SIZE(rcvr)	Set to the results of the verb MOMENT.
SMWGT(12)	Smoothing function used by SMOOTH.
TMARK(6)	The temperatures at which up to six horizontal lines are to be drawn by SHOW.
VMARK(6)	The velocites at which up to six vertical lines are to be drawn by SHOW.
VRMS(rcvr)	The array containing the results of the verb RMS.
WEIGHT(rcvr)	The weight used by SUM in accumulating data.

YINCR(CH)	The distance between Y tic marks that is to be used by SHOW.
YMIN(CH)	The value of Y which is to be the minimum value to appear on a graph by SHOW.
ZLINE	If =1, SHOW will draw a horizontal line at zero Kelvins.

Chapter 5

Getting Started

The current version of the POPS / Spectral Line program is available on both the MODCOMP Classic and under the CMS environment of the IBM 4341. This is essentially the same program as the on-line analysis programs used at the telescopes in Green Bank and Tucson. The purpose of this chapter is to explain how to start the program on either of the computers. Much of the information is the same for both computers but is repeated so that the reader only has to refer to one section at a time.

5.1. Using POPS/LINE on the IBM

Any terminal which talks to the IBM can be used to run the LINE program, but of course graphic displays can only be done at the Tektronics 4010 terminal located on the first floor in Room 106. To use any terminal, that terminal must be put into the CMS system. If you are using a non-graphics terminal which is part of the PANDORA system, please see the document Using the Interactive Systems of the IBM, available from the Computer Division Secretary.

For graphics display, you will be using the Tektronics terminal, so go to the machine room and make sure that LINE 023 is enabled for CMS. If it is not, then have the operator enable that line for CMS. If the operator is new or for some reason does not know how to do this, show him Appendix A.3 in this manual. When the terminal is ready, you will see the message VM/370 ONLINE printed on the terminal. These screens have an auto-erase feature, and it is possible that this message may disappear before you sit down at the terminal. If you suspect this, try pressing the carriage return key. If the terminal is active, the computer will prompt you with a period.

5.1.1. Starting LINE

Starting the Spectral Line program is like logging on to any other CMS "machine." (Again, please see the document on the interactive systems of the IBM for a detailed description.) First press the RETURN key and get the system's prompting period. When you are prompted, type

```
LOGON POPS MACHINE
```

If you make a mistake, the computer will print the message RESTART. Try again. The backspace key does not work until after the logon procedure is completed.

It is possible that the disk pack that contains all the files necessary for the LINE program is not mounted on the system. If this is the case, the following message will appear on the screen:

```
****************************************************************
*                                                              *
*  The disk that contains the LINE programs and data is not    *
*  currently mounted on the system.  A message has been sent   *
*  to the computer operator, and he should send you a message  *
*  informing you when the disk is mounted.  The name of this   *
*  disk is STORE.  Please be patient, it may take a minute or  *
*  two.  When the operator notifies you that the disk is       *
*  mounted, of if you think you have waited long enough, type  *
*  the command ...    LINKPOPS                                 *
*                                                              *
****************************************************************
```

If, after you type LINKPOPS the STORE disk is still not mounted, the following message will appear:

```
*
*   The STORE disk has not yet been mounted.  Please wait a bit
*   longer.  The operator should send you a message.  When he does,
*   type the command ... LINKPOPS    again.
*
```

If you get impatient, you may want to call the operator at extention 250. Sometimes the operator will not see the message because it appeared in the middle of some other output and got rolled off of the operator's screen.

With the STORE disk mounted, the logon will be accepted and the following will appear on the screen:

This is the CMS machine for the POPS/Spectral Line program and its auxiliary programs. The following commands are available:

.... LINE	....	To start the data analysis program.
.... LOADTAPE	....	To load data to the disk.
.... KTAPE	....	To dump the KEEP file to tape.
.... PLOTTAPE	....	To dump the PLOT file to tape (for the OS procedure MCPLOT)
.... CNTLCARD	....	Dumps the position-position control card file to tape.

THE COMMAND ... INFO ... WILL REPEAT THIS LIST ON THE SCREEN.

After this is complete, you will be prompted again with a period. There is no "read ahead" in the CMS system, so never try to type anything until you are prompted to do so. When you are prompted, you are ready to start the LINE program.

You start the Spectral Line program by typing the command...

LINE

As with logging on, you will get several lines of messages. When LINE is ready for input, you will get the standard LINE prompt, a right caret (>) and the CMS prompting period (.) on the same line. NEVER type a line until you see this period. If you DO type a command before the period is printed, the computer will probably type the prompt CP on the screen. If this happens, you can get back into LINE by typing the CP command B (for Begin).

5.1.2. Using Tapes

When you first log on to POPS, there is no tape drive assigned to you. Before you can load data to the disk or dump data from the disk you will have to get the operator to assign a tape drive to you. One way is to call him at ext. 250. Another way is to send a message from the terminal. This is done with the CMS command MESSAGE (MSG). For example,

```
MSG OP PLEASE ASSIGN A TAPE DRIVE FOR POPS AT ADDRESS 181
MSG OP PLEASE MOUNT TAPE 2496 ON THE DRIVE
```

When the operator has assigned a drive and/or mounted the tape, you will get a message at your terminal (you may have to press the RETURN key to get it printed). If things are busy in the computer room, your message may be hidden on the operator's terminal, or may roll off before it is seen. If after a few minutes nothing has happened, type the command ... QUERY TAPE ... to see if you have been assigned a drive. If your tape has not been mounted, you might want to send the message again.

When you finish loading or dumping and will not need the drive again within a short time, you should release the drive back to the system. You can do this with the CP command ...

```
DET 181
```

This will unload your tape and DETach the drive from POPS.

With regard to the number 181 above, in CMS terminology 181 is the "virtual address" to which the tape drive was assigned to POPS. This is one of four addresses, 181 through 184. The LINE commands assume that 181 was used, so if the tape gets assigned to a different address, the commands will not work.

5.1.2.1. Loading User or Keep Tapes

The command LOADTAPE is used to load data from a tape to the LINE data file. The data format on the tape can be the LINE KEEP format, the IBM 2496 byte format (usually produced when a 300-foot telescope tape is processed into a User tape), or the IBM 4980 byte format (the User tape produced from a 140-foot telescope tape). The program is able to determine which format is being used.

When the program starts, is asks you whether you want the data file initialized. If you do, it will start loading your data from the top of the file, overwriting any other data that may have existed. Otherwise, it will start loading your data after the last scan that is currently on the disk.

The program will also prompt you for windowing parameters. The program can currently window by right ascension and/or scan number.

Your data will remain on disk until someone else writes over your data using the LOADTAPE program. Because of the large amount of disk space needed for the data file, it is not feasible at the moment to have private data files. However, the next section talks about dumping your data file to tape and restoring it back to the disk.

The LINE interactive LOAD verb (as the one used on the MODCOMP) was removed because it was undesirable for the POPS logon to hold a tape drive for extended periods of time and thus precluding its use by others.

5.1.2.2. Private Data Files

If you wish to dump your data file to tape, you can use the CMS TAPE command to do so. After EXITing the LINE program, type in ...

```
TAPE DUMP DK1 LINEPOPS G
```

This will dump (here dump is the word to use) the data file to tape. When you want to restore the tape to disk, type ...

```
TAPE LOAD DK1 LINEPOPS G (REPLACE
```

This will destroy the existing data file, and replace it with the tape's data. This will usually work much faster than executing the LOADTAPE program again to re-load your data. Be warned, however, that the tape created by the TAPE DUMP command can only be read by another TAPE command.

If you get a message telling you that DK1 LINEPOPS is not accessible, try typing ...

```
LINK LINEPOPS 199 199 W
ACCESS 199 G
```

... and after getting the CMS prompt (.), type the desired TAPE command again.

5.1.2.3. Writing Keep Tapes

KTAPE is the command used to write the keep file (created by the LINE verb KEEP) to a LINE format KEEP tape. KLAB is command used to put an initial label on the tape so that KTAPE can write on the tape.

When the command KTAPE begins execution, it will ask you if the tape has been labeled. If you reply no, KTAPE will execute the KLAB command for you.

When the KTAPE program is running, it will ask you for the user number under which the scans were kept. It will also ask you if you want to add a constant to the scan number. All scans written to the tape are added after the previous last scan on the tape. When all scans are written to the tape, they are removed from the keep file.

5.1.2.4. Writing a Plot Tape

The command PLOTTAPE writes the plot file produced by the LINE program's plotting verbs to tape in the format used by the OS procedure MCPLOT. This procedure is almost useless for any other purpose. The disk file is an indexed file similar to the keep file. When a plot is written to tape, it is removed from the plot file.

THE PLOTTAPE COMMAND ALWAYS REWINDS THE TAPE BEFORE STARTING.

See the chapter on Off-data Display for instructions on executing the OS procedure MCPLOT.

5.1.2.5. Writing GPCP Control Card Files

The command CNTLCARD writes the General Purpose Contouring Program (GPCP) control card file produced by the LINE verb MAPRD verb to tape. This command does not "MOD" the tape but does ask if you want to skip any files on the tape before writing a new file. NOTE WELL!! If you tell the program to skip more files than there are on the tape, it could spin the tape off the reel. This is certain to happen if you are using a tape which has never been written on in the space after the last file -- such as a new tape.

5.1.3. Accessing Data on Disk

There are site-dependent differences in the data-taking programs which are reflected in differences in the way scans are numbered, and therefore the way scans are stored on disk.

For spectral line data there is a fundamental difference between the Tucson and Green Bank telescopes. In Tucson, each signal consists of the output of a single filter bank, and each signal is recorded with an individual scan number. Up to two signals can be recorded, in which case every other scan number corresponds to the output of the first filter bank. In Green Bank, autocorrelation spectrometers are used, and all of the recorded spectra are contained side-by-side within a single scan. To stack data from the first receiver, one need only stop sequentially through scan numbers, but additional information must be given to the program to specify what part of the data within the scan is to be used. This additional information is referred to as the receiver processing option, and is described briefly later in this section.

5.1.3.1. What Data is Stored

In addition to the data points and scan number, information pertaining to the observing program, the source, timing, receiver setup, observing frequencies, calibration values, telescope positioning, and environmental parameters is included with each scan. All of this information accompanies the data points at each step in the data reduction process.

5.1.3.2. Finding Out what is on Disk

The scan table is a directory consisting of the scan numbers of the data on disk, along with the addresses used for direct access to each scan. The simplest way of finding out what scans are on disk is to list the scan numbers in the scan table. This is done with the command TELL DISK. No scans need be read from disk, so the list is quickly generated.

A summary which produces information about the scans on disk must read in each scan referred to in the scan table. This can be accomplished with the verb SUMMARY, used in the form

SUMMARY user#

where user# is the NRAO user number assigned by the Computer Division. This will print a single line of information for each scan. Scans are read into the TEMP scan register, whose contents are lost. To speed up the summary, the range of scan numbers actually read in can be limited by setting adverbs BSCAN and ESCAN to beginning and ending scan numbers. However, BSCAN and ESCAN will be reset to their default values after SUMMARY is invoked.

To direct a SUMMARY or TELL DISK to the hard copy device, first use the verb PRINTER. Use the verb CRT to route subsequent print ouput back to the CRT.

To speed access to data on disk, it is possible to remove references to other observers' data from the scan table. The verb WIPE will first cause the entire disk area to be read while a list of NRAO user numbers found is compiled. You may then specify the user number(s) you wish to delete. Note that this procedure does not actually free up any disk space. Sometimes, a previous observer will have observed a scan with the same scan number as one of yours. If that other scan appears earlier in the scan table than yours, a request for your scan will bring in the other instead. Using WIPE should cure this problem.

5.1.3.3. Loading Spectral Line Data from Disk

To bring a single scan from disk into the WORK array, use the command GET in the form

```
GET scan#
```

The pointer will be set to the WORK array so that SHOW will display the newly loaded data.

For frequency-switched data, individual scans can be loaded from disk and manipulated, but for total power scans, two scans must be loaded. The ratio spectrum

$$T_i = T_{sys} * (ON_i - OFF_i) / OFF_i \qquad (1)$$

must be formed before processing can continue. An unambiguous way of loading ON and OFF scans is with verbs ON and OFF in the form

```
ON onscan#
OFF offscan#
```

The ON scan goes into the WORK array, and the OFF scan goes into the TEMP array. LINE knows which scans were called ON scans when they were observed and which were OFF scans. This allows a shorthand procedure, provided that the OFF was observed before the ON. Using the command FETCH in the form

FETCH onscan#

will load the specified scan into the WORK array as the ON scan (regardless of whether it was called an ON or OFF when observed), and the most recently observed scan flagged as an OFF will be loaded into the TEMP array.

If a scan number given explicitly in a command or implicitly in a search for a preceding OFF is not found in the scan table, then the message

SCAN ?

will be generated, and the program will exit to the EXECUTE communications mode.

After ON and OFF scans are loaded, use the verb TEMP to form the ratio of equation (1). For example, if scan 7238 is an ON scan and 7237 is an OFF scan, then the difference spectrum can be formed with these commands:

FETCH 7238 TEMP

However, if the associated OFF scan were 7239, later than the ON, then you must use

ON 7238 OFF 7239 TEMP

5.1.3.4 The Receiver Processing Option

The basic unit of spectral line data is the spectrum. Filter bank data from Tucson maintains a one spectrum - one scan number correspondence. This is not true of autocorrelator data from Green Bank, where a single scan number may be associated with up to four spectra. When loading spectral line data from disk, and in a number of other operations, all receivers recorded in a scan are processed in parallel. For display purposes, however, the basic unit of data is the spectrum of a single receiver, and because the information on the independent receivers may be quite different, it is often useful to process them separately. The receiver processing option gives you control over the way the separate receivers are processed.

The current Green Bank observing systems allow up to four receivers to be recorded in a scan. They are labelled A for the first receiver, and if present, B for the second, and C and D for the third and fourth. You have the choice of processing all receivers present or of processing only a single receiver. The verb ALL specifies that all receivers will be processed in parallel. The verbs ONLYA, ONLYB, ONLYC, and ONLYD specify that only receiver A, B, C, or D, respectively will be processed. The default processing option is ALL, and the option is reset to ALL every time a scan is loaded from disk. If receivers have been processed independently then it will probably be necessary to reset the processing option to ALL before averaging with another scan.

5.1.4. Summary of Analysis Verbs

Autocorrelation spectrometer data differs from filter bank data in scan numbering scheme, since all spectrometer channels are filed under the same scan number, regardless of the configuration of the autocorrelator. The frequency resolution is 1.21 times the channel spacing, and 2.0 times the channel spacing after Hanning smoothing.

The system temperature recorded with your data includes sky, atmosphere and source contributions, a receiver contribution, and a contribution equal to half the calibration noise source, since it is on half the time. The system temperature calculation depends linearly on the assumed temperature contribution of the calibration noise source.

To remove the instrumental profile, two spectra are observed - signal and reference - and the ratio (signal-reference)/reference is formed. In total power observing the signal and reference spectra have different scan numbers and must be differenced in the analysis computer, using the verb TEMP. In frequency switched observing, signal and reference spectra are accumulated simultaneously, and only the differenced spectrum is passed to the analysis computer.

The receiver processing option controls the processing of scans containing more than one spectrum. Use ALL to process all receivers in parallel, where appropriate, and to display the first receiver. Use ONLYA, ONLYB, ONLYC, or ONLYD to process and display only the first, second, third, or fourth receiver. Loading a scan from the raw data disk resets the receiver processing option to ALL. However, scans reloaded from a KEEP tape or a SAVE/RECALL bin retain the receiver processing option in effect when the data were KEEP'ed or SAVE'ed.

Use the verb SLIDE to average two receivers in the same scan. If more than two receivers are present, first use the verb PAIR to specify which receivers are to be averaged. Use the verb RDIFF with PAIR to subtract one receiver from another in the same scan.

Use the verb OVERLAP to average the two portions of the scan covering a common span of velocities when Dual-Dicke, or overlapped frequency switching is used. The number of channels by which the reference portion of the spectrum is shifted is computed automatically, but the channels not in common are not dropped from the display.

Polynomial baseline fitting has several options for displaying intermediate steps. You must first specify NFIT, the order of the polynomial which is to be fit. Use BASELINE to compute and subtract the baseline. Use BSHAPE to compute the baseline, and BMODEL to construct a model of the baseline. If the model is constructed in the TEMP array, then the verb DIFF can be used to remove the baseline from the data.

Sinusoidal baseline fitting has the same options for displaying intermediate steps as polynomial fitting. You must first specify an initial guess for RFREQ, the frequency of the sinusoid in MHz. Use RIPPLE to compute and remove a baseline. Use RSHAPE to compute the parameters of the baseline, and RMODEL to replace the data with a model of the sinusoid. If the model is constructed in the TEMP array, then the verb DIFF can be used to remove the baseline from the data.

When averaging scans, use the verb ALIGN before accumulating to insure that the current scan is shifted to be on the same velocity scale as the accumulated data in the HOLD array. It can do no harm to invoke ALIGN every time before accumulating a scan.

Extensive facilities exist for directing displays of spectra to tape for plotting on the CALCOMP plotter driven by the IBM computer. Many of the options are similar to those of the verb SHOW. On the MODCOMP, the verbs MAP and PIX also exist for directing contour maps to tape for display on the CALCOMP or the DICOMED display.

GMEASURE is a specialized routine for reducing 21 cm galaxy profiles.

A number of procedures in the POPS language are built-in to the LINE system:

XX	A shorthand way to produce a display, equivalent to PAGE SHOW.
GETSCAN	A procedure which will load a specified scan, and if there is an associated total power off scan, then it will load it also and form the difference spectrum.
LOOK	Same as GETSCAN except that a display is also produced.
SHOWS	Same as LOOK except that all of the scans in the scan number stack are displayed on the same page.
R1, R2, R3, R4	Select and display the associated receiver.

-continued-

CB — Average all of the scans listed in the scan number stack, using default weighting, and display the result.

NRSET — Use the crosshairs to set baseline fitting regions or rms regions.

ADVSET — Return off reference lines on a display except for the zero line and baseline window boxes.

RESET — Same as ADVSET except that BDROP and EDROP are also set to 10.

RESET36 — Same as ADVSET except that BDROP and EDROP are also set to zero.

5.1.5. Logging Off (Gracefully or Otherwise)

Gracefully:

After you have EXITed from LINE, you should log off from POPS. One of the quirks of CMS is that only one person will be able to use the program at a time. Therefore, if you do not log off, no one else will be able to run the program.

The command to log off is simply LOGOFF, or more simply, LOG.

Otherwise:

If, for some reason, you need to abort LINE, press the BREAK key. When the computer prompts you with a period, type the command ...

HX

... for Halt eXecution. This should abort the LINE program and you will be back in the CMS environment logged on under POPS.

If even this does not work, you can ask the computer operator to "force POPS off the machine." This will result in the VM/370 ONLINE message. You will no longer be logged on under POPS.

5.2. Using POPS/LINE on the MODCOMP

In Charlottesville, the terminals and tape drives are located in Room 215, and the computer - a MODCOMP Classic - and disk drives, are in Room 216. This building is known as Stone Hall to trivia buffs and University of Virginia officials. In Room 215 there are two user terminals, two tape drives, and a Varian printer/plotter associated with the MODCOMP. The terminals are on the right hand side of the room and are identified with tags. The other terminals and the Versatek printer/plotter belong to the VAX system. The MODCOMP Classic is a multi-user system and you should always find it running. However, POPS/LINE will seldom be running.

5.2.1. Starting LINE

While LINE can be run from either of the Tektronix 4012 terminals, you should normally use the one designated as the B batch processing terminal. The other 4012 is used by the AIPS VLA post-processing system.

Before starting LINE, determine which tape drive you will be using to load data or to write plots or maps. The default tape drive is drive MT2, which is the one on the left as you face them. MT1 can be used by setting a parameter when you start LINE. Load the tape by following the diagram on the drive. If the button marked PE/NZRI is not lit, push it to select 1600 BPI. Now press the LOAD button. The tape will advance to its load point and the ON-LINE and LOAD lights will be lit.

Now go to the terminal. To prolong the lives of the 4012 terminals, they are often turned off. The on/off rocker switch is located on the upper right had corner of the base pedestal. After it is turned on, the terminal's screen will become bright; push the PAGE ERASE button. It usually will take one or two minutes warm-up time before you will be able to see what you type on the screen. You should also check that the switch on the left side of the base is set for the MODCOMP. After the terminal is ready to use, type

```
JOB
```

The computer should respond with a $ (dollar sign) prompt. If it does not, you will have to activate a background processor for the terminal. This procedure will be explained below.

When you have the $ prompt, you are ready to start the LINE program. If you will be using tape drive 2, type

```
LINE      or      DO LINE
```

If you will be using tape drive 1, type

LINE MT1 or DO LINE MT1

The computer will now respond with

ASS LO NO

The carat indicates that POPS is ready to accept commands.

5.2.1.1. How to Get a $ Prompt

If you typed JOB and did not get the $ prompt, go to the operator's terminal. Currently, this is the Tektronix 4025 next to the Varian printer. Type "Control A" by holding down the key marked CTRL and striking the "A" key. The computer should respond with

/IV/....../

where what appears between the second and third slashes depends on previous commands. Next type:

/B/E,,BM

If the computer accepts the commands, the first forward slash of each command will be replaced by a backward slash. If you will be using the other 4012 terminal (designated C), replace the /B/E with /C/E. A $ prompt should now appear on your terminal. If there is not one, try typing a SPACE followed by a RETURN. Now try the start-up procedure given above. If nothing happens, you will have to IPL the computer.

5.2.1.2. IPL'ing the Computer

IPL'ing the computer should seldom be necessary. Try checking with someone before you do it. To IPL go to Room 216. The MODCOMP Classic is in the first equipment rack on the left. The MODCOMP and its peripherals in Room 216 are always left on, so if any of the equipment is turned off, DO NOT turn it on - it is off for a reason. To run LINE, the two small Diablo disks must be on. The three large Ampex drives should also be on.

Use the following procedure to start the IPL:

1. Set the RUN/HALT switch to HALT.
2. In the register switches above the RUN/HALT switch, set switch 15 up and the others to the center position. Note that there is a momentary contact position below the center position.
3. Press MASTER CLEAR.
4. Press FILL.
5. Press the RUN/HALT switch to RUN.

Now go to the operator's console in Room 215. After about a minute, the computer will prompt you to enter the date and then the time. If you begin to get error messages before the date prompt, go back to Room 216 and set the RUN/HALT switch to HALT and then try to find Lorrie Morgan, Eric Griesen, or Dave Brown.

After you enter the time, the system will give you a $ prompt. Go through the process of activating a terminal and start LINE as described above. If one of the Ampex disk drives is down, you may get a series of error messages and no prompt. These you can ignore. Just go through the procedure of activating a terminal. If you need any help, look for Lorrie or one of the other programmers.

5.2.2. Using Tapes

5.1.2.1 Loading User or Keep Tapes

Loading data onto the MODCOMP data disk from tape is done via the LINE verb LOAD. See the Verb and Operator Dictionary.

5.1.2.2. Writing Keep Tapes

The KEEP file is a disk file for storing reduced data. It exists so that its contents may be copied to tape and permanently stored. Here we describe the process for copying the KEEP file to a KEEP tape.

A new tape must be labelled as a KEEP tape before data can be written on it. To do so:

1. Turn on POWER to a tape drive and mount the tape.
2. Check that 1600bpi or high density is selected.
3. EXIT the data reduction program if it is running.
4. Initiate the KEEP tape labelling program by entering the commands:

```
$JOB
$DO KLAB MTn
```

where n is either 1 or 2, depending on which drive your tape is mounted. The tape label contains room for your user number. The first time you write data to the tape, your user number will be written into the tape label. For all subsequent tape writes, the user number in the tape label is compared to the user number recorded with your data, so that only one user's data may be recorded on the same KEEP tape.

To copy the KEEP file on disk to a KEEP tape:

1. Turn on POWER to a tape drive and mount the tape.
2. Check that 1600bpi or high density is selected.
3. EXIT the data reduction program if it is running.
4. Initiate the KEEP tape writing program by entering the commands:

   ```
   $JOB
   $KEEP MTn
   ```

   where n is either 1 or 2, depending on which drive your tape is mounted.
5. The program will ask you to enter your user number.
6. The program will advance the tape to the end-of-file mark at the end of the tape before writing.

As scans are copied to the tape, they are simultaneously erased from the disk file, freeing up space to KEEP more scans from the analysis programs. To maintain two copies of a KEEP tape, however, you will have to repeat the copying to the KEEP file as well as the tape writing since the disk file is cleared as it is copied.

5.2.2.3. Writing Plot Tapes

The verbs PLOT, REPLOT, PROFILE, MAP and PIX in the LINE program will write your plots directly to the tape. The tape drive used in determined when you started the program. The default drive is MT2. You then take the tape downstairs onto the IBM and use the OS procedure MCPLOT to produce your CALCOMP plots. See the chapter on off-line data display for an explanation of the use of MCPLOT.

5.2.3. Accessing Data on Disk

There are site-dependent differences in the data-taking programs which are reflected in differences in the way scans are numbered, and therefore the way scans are stored on disk.

For spectral line data there is a fundamental difference between the Tucson and Green Bank telescopes. In Tucson, each signal consists of the output of a single filter bank, and each signal is recorded with an individual scan number. Up to two signals can be recorded, in which case every other scan number corresponds to the output of the first filter bank. In Green Bank, autocorrelation spectrometers are used, and all of the recorded spectra are contained side-by-side within a single scan. To stack data from the first receiver, one need only stop sequentially through scan numbers, but additional information must be given to the program to specify what part of the data within the scan is to be used. This additional information is referred to as the receiver processing option, and is described briefly later in this section.

5.2.3.1. What Data is Stored

In addition to the data points and scan number, information pertaining to the observing program, the source, timing, receiver setup, observing frequencies, calibration values, telescope positioning, and environmental parameters is included with each scan. All of this information accompanies the data points at each step in the data reduction process.

5.2.3.2. Finding Out what is on Disk

The scan table is a directory consisting of the scan numbers of the data on disk, along with the addresses used for direct access to each scan. The simplest way of finding out what scans are on disk is to list the scan numbers in the scan table. This is done with the command TELL DISK. No scans need be read from disk, so the list is quickly generated.

A summary which produces information about the scans on disk must read in each scan referred to in the scan table. This can be accomplished with the verb SUMMARY, used in the form

```
SUMMARY user#
```

where user# is the NRAO user number assigned by the Computer Division. This will print a single line of information for each scan. Scans are read into the TEMP scan register, whose contents are lost. To speed up the summary, the range of scan numbers actually read in can be limited by setting adverbs BSCAN and ESCAN to beginning and ending scan numbers. However, BSCAN and ESCAN will be reset to their default values after SUMMARY is invoked.

To direct a SUMMARY or TELL DISK to the hard copy device, first use the verb PRINTER. Use the verb CRT to route subsequent print ouput back to the CRT.

To speed access to data on disk, it is possible to remove references to other observers' data from the scan table. The verb WIPE will first cause the entire disk area to be read while a list of NRAO user numbers found is compiled. You may then specify the user number(s) you wish to delete. Note that this procedure does not actually free up any disk space. Sometimes, a previous observer will have observed a scan with the same scan number as one of yours. If that other scan appears earlier in the scan table than yours, a request for your scan will bring in the other instead. Using WIPE should cure this problem.

5.2.3.3. Loading Spectral Line Data from Disk

To bring a single scan from disk into the WORK array, use the command GET in the form

```
GET scan#
```

To bring in the scan currently being accumulated, and not yet filed on disk, use CGET with no scan number. In either case, the pointer will be set to the WORK array so that SHOW will display the newly loaded data.

For frequency-switched data, individual scans can be loaded from disk and manipulated, but for total power scans, two scans must be loaded. The ratio spectrum

$$T_i = T_{sys} * (ON_i - OFF_i) / OFF_i \qquad (1)$$

must be formed before processing can continue. An unambiguous way of loading ON and OFF scans is with verbs ON and OFF in the form

```
ON onscan#
OFF offscan#
```

The ON scan goes into the WORK array, and the OFF scan goes into the TEMP array. LINE knows which scans were called ON scans when they were observed and which were OFF scans. This allows a shorthand procedure, provided that the OFF was observed before the ON. Using the command FETCH in the form

FETCH onscan#

will load the specified scan into the WORK array as the ON scan (regardless of whether it was called an ON or OFF when observed), and the most recently observed scan flagged as an OFF will be loaded into the TEMP array. The command CFETCH, used without a scan number, will load the scan currently being accumulated as the ON scan, and will also load the most recently observed OFF scan.

If a scan number given explicitly in a command or implicitly in a search for a preceding OFF is not found in the scan table, then the message

SCAN ?

will be generated, and the program will exit to the EXECUTE communications mode.

After ON and OFF scans are loaded, use the verb TEMP to form the ratio of equation (1). For example, if scan 7238 is an ON scan and 7237 is an OFF scan, then the difference spectrum can be formed with these commands:

FETCH 7238 TEMP

However, if the associated OFF scan were 7239, later than the ON, then you must use

ON 7238 OFF 7239 TEMP

5.2.3.4 The Receiver Processing Option

The basic unit of spectral line data is the spectrum. Filter bank data from Tucson maintains a one spectrum - one scan number correspondence. This is not true of autocorrelator data from Green Bank, where a single scan number may be associated with up to four spectra. When loading spectral line data from disk, and in a number of other operations, all receivers recorded in a scan are processed in parallel. For display purposes, however, the basic unit of data is the spectrum of a single receiver, and because the information on the independent receivers may be quite different, it is often useful to process them separately. The receiver processing option gives you control over the way the separate receivers are processed.

The current Green Bank observing systems allow up to four receivers to be recorded in a scan. They are labelled A for the first receiver, and if present, B for the second, and C and D for the third and fourth. You have the choice of processing all receivers present or of processing only a single receiver. The verb ALL specifies that all receivers will be processed in parallel. The verbs ONLYA, ONLYB, ONLYC, and ONLYD specify that only receiver A, B, C, or D, respectively will be processed. The default processing option is ALL, and the option is reset to ALL every time a scan is loaded from disk. If receivers have been processed independently then it will probably be necessary to reset the processing option to ALL before averaging with another scan.

5.2.4. Summary of Analysis Verbs

Autocorrelation spectrometer data differs from filter bank data in scan numbering scheme, since all spectrometer channels are filed under the same scan number, regardless of the configuration of the autocorrelator. The frequency resolution is 1.21 times the channel spacing, and 2.0 times the channel spacing after Hanning smoothing.

The system temperature recorded with your data includes sky, atmosphere and source contributions, a receiver contribution, and a contribution equal to half the calibration noise source, since it is on half the time. The system temperature calculation depends linearly on the assumed temperature contribution of the calibration noise source.

To remove the instrumental profile, two spectra are observed - signal and reference - and the ratio (signal-reference)/reference is formed. In total power observing the signal and reference spectra have different scan numbers and must be differenced in the analysis computer, using the verb TEMP. In frequency switched observing, signal and reference spectra are accumulated simultaneously, and only the differenced spectrum is passed to the analysis computer.

The receiver processing option controls the processing of scans containing more than one spectrum. Use ALL to process all receivers in parallel, where appropriate, and to display the first receiver. Use ONLYA, ONLYB, ONLYC, or ONLYD to process and display only the first, second, third, or fourth receiver. Loading a scan from the raw data disk resets the receiver processing option to ALL. However, scans reloaded from a KEEP tape or a SAVE/RECALL bin retain the receiver processing option in effect when the data were KEEP'ed or SAVE'ed.

Use the verb SLIDE to average two receivers in the same scan. If more than two receivers are present, first use the verb PAIR to specify which receivers are to be averaged. Use the verb RDIFF with PAIR to subtract one receiver from another in the same scan.

Use the verb OVERLAP to average the two portions of the scan covering a common span of velocities when Dual-Dicke, or overlapped frequency switching is used. The number of channels by which the reference portion of the spectrum is shifted is computed automatically, but the channels not in common are not dropped from the display.

Polynomial baseline fitting has several options for displaying intermediate steps. You must first specify NFIT, the order of the polynomial which is to be fit. Use BASELINE to compute and subtract the baseline. Use BSHAPE to compute the baseline, and BMODEL to construct a model of the baseline. If the model is constructed in the TEMP array, then the verb DIFF can be used to remove the baseline from the data.

Sinusoidal baseline fitting has the same options for displaying intermediate steps as polynomial fitting. You must first specify an initial guess for RFREQ, the frequency of the sinusoid in MHz. Use RIPPLE to compute and remove a baseline. Use RSHAPE to compute the parameters of the baseline, and RMODEL to replace the data with a model of the sinusoid. If the model is constructed in the TEMP array, then the verb DIFF can be used to remove the baseline from the data.

When averaging scans, use the verb ALIGN before accumulating to insure that the current scan is shifted to be on the same velocity scale as the accumulated data in the HOLD array. It can do no harm to invoke ALIGN every time before accumulating a scan.

Extensive facilities exist for directing displays of spectra to tape for plotting on the CALCOMP plotter driven by the IBM computer. Many of the options are similar to those of the verb SHOW. On the MODCOMP, the verbs MAP and PIX also exist for directing contour maps to tape for display on the CALCOMP or the DICOMED display.

GMEASURE is a specialized routine for reducing 21 cm galaxy profiles.

A number of procedures in the POPS language are built-in to the LINE system:

XX	A shorthand way to produce a display, equivalent to PAGE SHOW.
GETSCAN	A procedure which will load a specified scan, and if there is an associated total power off scan, then it will load it also and form the difference spectrum.
LOOK	Same as GETSCAN except that a display is also produced.
SHOWS	Same as LOOK except that all of the scans in the scan number stack are displayed on the same page.
R1, R2, R3, R4	Select and display the associated receiver.

-continued-

CB	Average all of the scans listed in the scan number stack, using default weighting, and display the result.
NRSET	Use the crosshairs to set baseline fitting regions or rms regions.
ADVSET	Return off reference lines on a display except for the zero line and baseline window boxes.
RESET	Same as ADVSET except that BDROP and EDROP are also set to 10.
RESET36	Same as ADVSET except that BDROP and EDROP are also set to zero.

5.2.5. Logging off (Gracefully or Otherwise)

Gracefully:

After EXITing from the LINE program, you will get the $ prompt. Simply type in ...

JOB

... any you will be finished.

Otherwise:

If, for some reason you must abort the LINE program, go to the operator's terminal and type "CONTROL A" by holding down the key marked CTRL and striking the "A" key. The computer should respond with

/IV/...../

where what appears between the second and third slashes depends on previous commands. Next, type:

/B/A if you were at the B graphics terminal

or

/C/A if you were at the C graphics terminal.

You will then get a $ on the appropriate graphics terminal, and you can either start the LINE program again, if you knew what happened, or go find Lorrie Morgan, if you didn't know what happened.

Chapter 6

Quick Reference

6.1. Display Examples

Display Embellishments

Embellishment	To add it, specify	To subtract it, specify
Baseline boxes.	BMARK=1 PAGE SHOW	BMARK=0 PAGE SHOW
Horizontal line at zero kelvins.	ZLINE=1 PAGE SHOW	ZLINE=0 PAGE SHOW
Vertical lines at channels c1 and c2.	VMARK(1)=c1 VMARK(2)=c2 PAGE SHOW	VMARK=0 PAGE SHOW
Horizontal lines at t1 and t2 kelvins.	TMARK(1)=t1 TMARK(2)=t2 PAGE SHOW	TMARK=0 PAGE SHOW

Setting the scaling for a graph

SHOW normally computes its own y-scaling. If the y-scaling computed by SHOW is not satisfactory, you can set your own by setting the values of YMIN and YINCR, where

YINCR = the distance between tic marks
YMIN = lowest y value to appear on the graph.

Once you have set YINCR and YMIN, SHOW will use them until

(a) You use the verb FREEY, which returns control of the scaling to SHOW, or
(b) You set YMIN = -9999

If the y-scaling computed by SHOW is so satisfactory that you want to continue to use that scaling on all plots, you may force it to do that by using the verb HOLDY. To return to automatic scaling, follow steps (a) or (b) above.

Note that YMIN and YINCR may be given different values for each receiver.

YINCR and YMIN must be used together. Using only YMIN can produce erroneous results.

Simple Display Procedures

For looking at a number of total power scans:

```
PROCEDURE LOOK (FSCAN,LSCAN)
FOR S = FSCAN TO LSCAN BY 2
     FETCH (S) TEMP PAGE SHOW
     READ N
     END
RETURN
FINISH
```

For looking at a number of frequency switched scans:

```
PROCEDURE SEE(FSCAN,LSCAN)
FOR S = FSCAN TO LSCAN
     GET (S) PAGE SHOW
     READ N
     END
RETURN
FINISH
```

The above procedures will display the scans starting with scan number FSCAN and ending with scan number LSCAN. After each scan is displayed, the program will wait for the user to type a character before drawing the next scan.

6.2. Frequency Baseline Examples

<u>Basic polynomial baseline removal</u>

```
GET 7238 PAGE SHOW
BDROP=20; EDROP=20
BBASE=50; EBASE=50
NFIT=2
BASELINE
PAGE SHOW
```

<u>Basic sinusoidal baseline removal</u>

```
GET 7238 PAGE SHOW
BDROP=20; EDROP=20
BBASE=50; EBASE=50
RFREQ=5
RIPPLE
PAGE SHOW
```

<u>Display of polynomial baseline before removal</u>

```
GET 7238 PAGE SHOW
NREGION(1,1)=25
NREGION(1,2)=40
NREGION(1,3)=100
NREGION(1,4)=180
NFIT=2
TRT
BSHAPE BMODEL RESHOW
PMW DIFF RESHOW
```

<u>Display of sinusoidal baseline before removal</u>

```
GET 7238 PAGE SHOW
NREGION(1,1)=25
NREGION(1,2)=40
NREGION(1,3)=100
NREGION(1,4)=180
RFREQ=5
TRT
RSHAPE RMODEL RESHOW
PMW DIFF RESHOW
```

Subtracting an average spectral value

```
GET 7238 PAGE SHOW
BDROP=15; EDROP=15
BBASE=45; EBASE=75
DCBASE
PAGE SHOW
```

A procedure for computation and display of polynomial baselines

```
PROCEDURE BSLN(ISCAN)
GET (ISCAN) PAGE SHOW
READ BBASE EBASE
NFIT=4
TRT BSHAPE BMODEL RESHOW
PMW DIFF RESHOW
RETURN
FINISH
```

To execute the procedure, type

```
BSLN(7238)
```

6.3. Gaussian Fitting Examples

Fit one gaussian to receiver A; display model

```
GET 7238 PAGE SHOW
BASELINE
NGAUSS=1
BGAUSS=10; EGAUSS=380
HWIDTH=5
CENTER=85
GAUSS
GMODEL
RESHOW
```

Fit one gaussian to receiver A; find residual

```
GET 7238 PAGE SHOW
BASELINE
NGAUSS=1
BGAUSS=10; EGAUSS=380
HWIDTH=5
CENTER=85
GAUSS
RESIDUAL
RESHOW
```

Fit one gaussian to receiver A; display model; find residual

```
GET 7238 PAGE SHOW
BASELINE
NGAUSS=1
BGAUSS=10; EGAUSS=380
HWIDTH=5
CENTER=85
GAUSS
TRT GMODEL RESHOW
PMW DIFF RESHOW
```

Create a model of a particular gaussian

```
NGAUSS=1
HEIGHT=5
HWIDTH=25
CENTER=96;
BGAUSS=1; EGAUSS=192
GMODEL
PAGE SHOW
```

Fit three gaussians to receiver A; display total model

```
GET 7238 PAGE SHOW
BASELINE
NGAUSS=3
BGAUSS=10; EGAUSS=380;
CENTER(1)=85
CENTER(2)=88
CENTER(3)=76
HWIDTH(1)=5
HWIDTH(2)=2
HWIDTH(3)=8
GAUSS
GMODEL
RESHOW
```

Fit three gaussians to receiver A; display separate fits

```
GET 7238 PAGE SHOW
BASELINE
NGAUSS=3
BGAUSS=10; EGAUSS=380;
CENTER(1)=85
CENTER(2)=88
CENTER(3)=76
HWIDTH(1)=5
HWIDTH(2)=2
HWIDTH(3)=8
GAUSS
GPARTS
```

Fit three gaussians to receiver A; find residual

```
GET 7238 PAGE SHOW
BASELINE
NGAUSS=3
BGAUSS=10; EGAUSS=380;
CENTER(1)=85
CENTER(2)=88
CENTER(3)=76
HWIDTH(1)=5
HWIDTH(2)=2
HWIDTH(3)=8
GAUSS
RESIDUAL RESHOW
```

Fit three gaussians to receiver A; do it all

```
GET 7238 PAGE SHOW
BASELINE
NGAUSS=3
BGAUSS=10; EGAUSS=380;
CENTER(1)=85
CENTER(2)=88
CENTER(3)=76
HWIDTH(1)=5
HWIDTH(2)=2
HWIDTH(3)=8
GAUSS
TRT GMODEL RESHOW
GPARTS
PMW RESIDUAL RESHOW
```

Procedure to fit NGAUSS gaussians

```
PRODEDURE GFITN(ISCAN)
GET (ISCAN) BASELINE PAGE SHOW
PRINT 'NGAUSS' READ NGAUSS
FOR N=1 TO NGAUSS
     PRINT 'CENTER' READ CENTER(N)
     PRINT 'HWIDTH' READ HWIDTH(N)
     END
GAUSS
RETURN
FINISH
```

To execute this procedure, type

```
GFITN (7238)
```

Procedure to fit one gaussian, display model, find and display residual

```
PROCEDURE GFIT1(ISCAN)
GET (ISCAN) BASELINE PAGE SHOW
PRINT 'CENTER' READ CENTER(1)
PRINT 'HWIDTH' READ HWIDTH(1)
NGAUSS=1
GAUSS
PMT GMODEL RESHOW
DIFF RESHOW
RETURN
FINISH
```

To execute this procedure, type

```
GFIT1 (7238)
```

6.4. Stacking and Averaging Examples

Stack four scans with same LO; display average

```
AVE
GET 7238 ACCUM
GET 7239 ACCUM
GET 7241 ACCUM
GET 7240 ACCUM
AVE PAGE SHOW
```

Stack four scans with different LO settings; display average

```
AVE
GET 7238 ACCUM
GET 7239 ACCUM
GET 7244
ALIGN
ACCUM
GET 7245 ACCUM
AVE PAGE SHOW
```

Average four scans, but only one receiver of two scans

```
AVE
GET 7238 ACCUM
GET 7239 ACCUM
GET 7244 ONLYB ACCUM
GET 7445 ONLYA ACCUM
AVE PAGE SHOW
```

Procedure to stack and average scans (same LO)

```
PROCEDURE SSTACK (FSCAN, LSCAN)
FOR S = FSCAN TO LSCAN
     GET (S) ACCUM
     END
AVE PAGE SHOW
RETURN; FINISH

PROCEDURE TSTACK (FSCAN, LSCAN)
FOR S = FSCAN TO LSCAN
     FETCH (S) TEMP ACCUM
     END
AVE PAGE SHOW
RETURN; FINISH
```

Procedure to stack scans; stack two groups of scans

```
PROCEDURE STAKSCAN (FSCAN, LSCAN)
FOR S = FSCAN TO LSCAN
     GET (S)
     ACCUM
     END
RETURN; FINISH
```

For Total Power data:

```
PROCEDURE STAKSCAN (FSCAN, LSCAN)
FOR S = FSCAN TO LSCAN BY 2
     FETCH (S)
     TEMP ACCUM
     END
RETURN; FINISH
```

You can now use this procedure to stack more than one group of scans:

```
AVE
STAKSCAN (7200,7222)
STAKSCAN (7240,7260)
AVE PAGE SHOW
```

Special TPOWER example

The purpose of this example is to average many onscans, average many offscans, and subtract the average of the offscans from the average of the onscans. First define the following procedure:

```
PROCEDURE TSCACK (FSCAN,LSCAN)
FOR S = FSCAN TO LSCAN BY 2
     ON (S) ACCUM
     END
AVE
RETURN; FINISH
```

Then, follow the example below:

```
AVE
TSTACK (6101,6149)
TRT
TSTACK (6100,6150)
TRW
DIFF
PAGE SHOW
```

Average a MODE 1 scan with receiver A of a MODE 2 scan

```
AVE
GET 4444
ACCUM
GET 4400
ASHIFT(1) = 92
ACCUM
AVE PAGE SHOW
```

This example will work only if the MODE 2 scan (4444) is accumulated first, and only if the MODE 2 scan has a bandwidth equal to half the bandwidth of the MODE 1 scan.

For a TPOWER scan, replace GET SCAN# with FETCH SCAN# TEMP

Set the value of ASHIFT(1) so that when the MODE 1 scan is added to receiver A of the MODE 2 scan, the velocities match. The actual number will depend on the data.

Average daily averages of scans

First, define the following procedure:

```
PROCEDIRE PILE (FSCAN, LSCAN)
FOR S = FSCAN TO LSCAN
     GET (S)
     ACCUM
     END
AVE PAGE SHOW
RETURN; FINISH
```

Then, follow the example below:

First day:

```
AVE
PILE (2122,2130)
NSAVE=25; SAVE
```

Second day:

```
AVE
PILE (2144,2152)
TRW ACCUM
NSAVE=25; PMW RECALL
ACCUM
AVE PAGE SHOW
```

6.5. Multi-Receiver Examples

Display two receivers

```
GET 7238
PAGE SHOW
ONLYB RESHOW
```

Remove a baseline from all receivers, but display C

```
GET 7238
ONLYC PAGE SHOW
ALL
TRT BSHAPE BMODEL
ONLYC RESHOW
ALL
PMW DIFF
ONLYC RESHOW
```

Averaging the output of receivers A and B

```
GET 7238
1 2 PAIR
SLIDE PAGE SHOW
```

Differencing receivers A and C

```
GET 7238
1 3 PAIR
RDIFF PAGE SHOW
```

Procedure to smooth receivers A and B and display both results

```
PROCEDURE SMOOTH1(SCAN)
GET (SCAN) PAGE SHOW
ONLYB RESHOW
BOTH HANNING PAGE SHOW
ONLYB RESHOW
RETURN; FINISH
```

To execute this procedure, type

```
SMOOTH1(7238)
```

Chapter 7

Fitting Options

7.1. Baseline

POPS/Spectral Line provides routines which allow the user to fit either a Chebyshev polynomial or a sinusoidal curve to user-specified regions of the data, to construct a model of the curve whose fit has been computed and to subtract the curve from the data.

The frequency baseline routines are:

FUNCTION	POLYNOMIAL VERB	SINUSOIDAL VERB
Calculate the parameters of the baseline.*	BSHAPE	RSHAPE
Evaluate the parameters and replace the data with the baseline model.	BMODEL	RMODEL
Calculate and evaluate the parameters of the baseline and subtract the baseline from the data.	BASELINE	RIPPLE

One other baseline verb is available:

DCBASE — Computes the average spectral value over the regions specified by the user and subtracts that value from the data.

All baseline verbs operate on the currently referenced array only. Thus, BSHAPE computes a fit using the data in the currently referenced array with a model of the last calculated polynomial baseline and BASELINE computes a curve using the data in the currently referenced array and subtracts the curve from that same data.

* Where "calculate parameters of the baseline" means find the coefficients of the Chebyshev polynomial of the desired order if a polynomial is used; or find the phase, amplitude and frequency of the sinusoid if a sinusoid routine is used.

To use the frequency baseline routines, the user must tell LINE which region(s) of the scan are to be used to fit the polynomial or sinusoid. These regions should be defined so as to exclude known or suspected spectral features. The baseline fitting regions are set by specifying the values of the adverbs

BDROP EDROP BBASE EBASE or NREGION

ADVERB	DESCRIPTION	INITIAL VALUE
BDROP(rcvr)	The number of channels on the extreme left side of each receiver which will be ignored by the fitting routines.	0
BBASE(rcvr)	The number of channels of the left side of the receiver (starting after the channels dropped by BDROP) which will be used to fit the baseline if NREGION(rcvr,1)=0.	50
EBASE(rcvr)	The number of channels on the right side of each receiver (up to the channels dropped by EDROP) which will be used to fit the baseline if NREGION(rcvr,1)=0.	50
EDROP(rcvr)	The number of channels of the extreme right right side of each receiver which will be ignored by the fitting routines.	0
NREGION(rcvr,N)	The channel numbers, for each receiver, at which up to four baseline fitting regions are to begin and end. If N is odd, NREGION(rcvr,N) = the channel number at which a baseline fitting region is to begin. If N is even, NREGION(rcvr,N) = the channel number at which a baseline fitting region is to end. If a region is not to be used, its starting channel should = 0. The only constraint on the channel numbers is that the end channel of a region must be larger than the start channel.	0

One more adverb, used by the verb SHOW, is of interest:

BMARK	If BMARK = 1, then SHOW will draw boxes to indicate the baseline fitting regions last used to compute a baseline.	0

Once the baseline fitting regions have been set, either a polynomial or sinusoidal baseline verb may be used. However, if a polynomial verb is to be used, the user must tell the program what order should be computed. This is done by specifying the value of the adverb NFIT.

NFIT	The order of the polynomial which is to be fit by a polynomial baseline routine. NFIT may be assigned any value between 1 and 12.	1

If, instead, a sinusoidal routine is to be used, the user must input the value of the adverb RFREQ:

RFREQ	The approximate frequency, in MHz, of the sinusoid to be computed.	23

7.2. Gaussian

When analyzing spectral line data, it is sometimes useful to model the obserbed spectrum with gaussian functions. A gaussian function is characterized by its height, its halfwidth, and the location of its center in the spectrum.

LINE provides a routine which iterates the user's estimates of the halfwidths and center locations of up to six gaussians until a fit is obtained. Once the characteristic values have been refined, the user may construct a model of the sum of the gaussian or subtract the sum from the data. A special routine allows the user to display each of the component gaussians.

The gaussian fitting routines are:

VERB	FUNCTION
GAUSS	Refine the user's estimates of the halfwidth(s) and center location(s) of the requested number of gaussians; calculates the heights of the gaussians.
GMODEL	Evaluate the refined values and replace the data with a model of the sum of the gaussians.
GPARTS	Using the refined values, construct and display each of the component gaussians.
PEAK	Finds the peak temperature in the spectrum and sets the center, halfwidth, height, BGAUSS, and EGAUSS for one gaussian component.
RESIDUAL	Evaluate the refined values and subtract the sum of the gaussians from the data.

The gaussian verbs operate on the currently referenced array only. GAUSS fits to the data in that array; GMODEL puts its model in that array; PEAK sets the gaussian parameters from that array and RESIDUAL subtracts the functions from that array. GPARTS, however, does not use or change the contents of any array.

The gaussian verbs work on one receiver's data at a time. Normally, the A receiver is used, but any of the other receivers may be used by specifying ONLYB, ONLYC or ONLYD (so that the SHOW verb will display the appropriate receiver) and assigning the appropriate adverbs' channel numbers.

To use the gaussian routines, the user must specify some initial information. This is done by assigning the appropriate values to the following adverbs:

ADVERB	DESCRIPTION	INITIAL VALUE
BGAUSS	The channel at which the fit begins if GREGION = 0.	1
CENTER(N)	The channel at which gaussian N has its center.	0
EGAUSS	The channel at which the fit is to end if GREGION = 0.	256
GREGION(rcvr,N)	The channel numbers, for each receiver, at which up to for gaussians are to begin and end if BGAUSS = 0. If N is odd, then GREGION(rcvr,N) = the starting channel number of a region. If N is even, GREGION(rcvr,N) = the ending channel number of a region.	0
HEIGHT(N)	The height in kelvins of gaussian N.	0
HWIDTH(N)	The width in number of channels of gaussian N at half its peak height.	0
NGAUSS	The number of gaussians to be fit. Must be a number between 1 and 6.	1

The user may also exert some control over GAUSS by changing the following adverbs:

FIXC	If ≥ 1, GAUSS will not refine the center locations of the gaussians.	1
FIXHW	If ≥ 1, GAUSS will not refine the values of the halfwidths of the gaussians.	1
NITER	The number of iterations GAUSS will perform trying to compute a fit with an RMS error less than 1.0E-05. If this condition is satisfied before NITER iterations have been made, GAUSS stops iterating.	8

Chapter 8

Mapping Options

POPS/Spectral line provides a routine which draws a position vs velocity contour map of a set of scans. Other routines specify the coordinate of the Y-axis and label the axes.

The mapping routines are:

VERB	FUNCTION
DEC	Sets the Y-coordinate of the map to declination.
GB	Sets the Y-coordinate of the map to galactic latitude.
GL	Sets the Y-coordinate of the map to galactic longitude.
LABEL	Labels the X and Y axes of the map.
MAPSHOW	Contours the specified scans with the specified levels.
RA	Sets the Y-coordinate of the map to right ascension.

MAPSHOW works on the currently referenced array and uses the HOLD array as intermediate storage (two scans must be in memory simultaneously to draw the contours). MAPSHOW can only work on one receiver at a time. The default is to map to reveiver A. To map the other receivers, specify ONLYB, ONLYC, or ONLYD before mapping.

To create a map the user must input some initial information. This is done by assigning the appropiate values to the following adverbs:

ADVERB	DESCRIPTION	INITIAL VALUE
BDROP	The number of channels to be dropped from the beginning of the data.	0
EDROP	The number of channels to be dropped from the end of the data.	0
LEVS	Up to 20 contour levels may be specified to be contoured.	-9999
MRATIO	The aspect ratio of the Y to the X axis.	1

The most convenient way to display a map is by using a procedure. One method would be to remove previously a baseline from the scans and save the data in NSAVE bins 1 through N on the disk. A procedure to map this data with right ascension as the Y-coordinate would be:

```
PROCEDURE MAPS(N)
PAGE RA
FOR I = 1 TO N
     NSAVE = I; RECALL MAPSHOW
     END
LABEL
RETURN
FINISH
```

The mapping levels and map ratio must be specified before invoking the procedure.

If standard baseline regions are to be used, no prior processing is necessary. The easiest method would be to load the desired scans into the stack and use a procedure as described below:

```
PROCEDURE MAPS2
PAGE GL
FOR I = 1 TO ACOUNT
     GET ASTACK(I) BASELINE MAPSHOW
     END
LABEL
RETURN
FINISH
```

Baseline regions and mapping levels must be specified before invoking the procedure. In this example, galactic longitude will be the Y-coordinate of the map.

There exists no provision for automatic scaling to set the contour levels. The array LEVS, of length twenty, must contain a list of each level to be plotted. If fewer than twenty levels are specified, they must go into the beginning of the array, and the first location after the desired levels should be set to a number less than -999. All the levels must be set with assignment statements. Defining a procedure to make the assignments may simplify this operation. For example, the followint procedure will set levels at a specified number of logarithmic intervals (db) down from a specified maximum.

```
PROCEDURE SETLEV (MAX,NLEVS)
NN = NLEVS + 1
FOR I = 1 TO NN
     J = NN - 1; LEVS(J) = MAX/10.**(I/10.);end
LEVS(NN) = -99999
RETURN
FINISH
```

Chapter 9

Resident Procedures

Most of the procedures built into the LINE program were copied from the Tucson system, which relies on the use of built-in procedures to a greater extent than has historically been the case in Green Bank or Charlottesville. Each procedure contains a RETURN statement, and may therefore be used in procedures you write.

Shorthand for PAGE SHOW: XX

The procedure XX is simply a shorthand for PAGE SHOW:

```
PROCEDURE XX
PAGE SHOW
RETURN
FINISH
```

Receiver Selection with Display: R1, R2, R3, and R4

Four separate procedures are built in to allow each receiver within a scan to be examined in a shorthand way, after the scan has been retrieved from disk. Each of the four procedures R1, R2, R3, and R4 sets the receiver processing option to the specified receiver and makes a display. Only R1 is listed below:

```
PROCEDURE R1
RCVR = 1; ONLYA PAGE SHOW
RETURN
FINISH
```

Getting Scans from Disk: GETSCAN and LOOK, F and S

The built-in procedure GETSCAN has the virtue of working for both frequency switched and total power scans. It takes as its object a single scan number, XSCAN. That scan, if found, is placed into the WORK array. The procedure then looks at the header data associated with scan XSCAN to see if there is an associated off source scan, as there would be if XSCAN were a total power scan. If there is, then the associated scan is placed into the TEMP array and the difference spectrum is formed by the TEMP verb:

```
PROCEDURE GETSCAN(XSCAN)
ON XSCAN; OSCAN = TWH(2,2)
IF OSCAN > 0 THEN OFF OSCAN; TEMP; END
RETURN
FINISH
```

To get a spectrum from disk and also produce a display, use the procedure LOOK(XSCAN), which calls GETSCAN:

```
PROCEDURE LOOK(XSCAN)
GETSCAN XSCAN
XX RCVR=1
RETURN
FINISH
```

F and S are procedures which retreive and display the first receiver of the on and off scans, respectively. They each take one argument, the desired scan number (notice that the procedures multiply the specified scan numbers by 10, and that the procedure S also adds 10 to the entered scan number), which may be entered immediately before or after F or S. Both F and S call the procedure GETSCAN and R1:

```
PROCEDURE F (XSCAN)
GETSCAN (XSCAN * 10) R1
RETURN
FINISH

PROCEDURE S (XSCAN)
GETSCAN (XSCAN * 10 + 10) R1
RETURN
FINISH
```

Multiple Scan Display: SHOWS, SHOW1, and SHOW2

To produce a display of each scan in the scan number stack, fitting them all on a single plot, first set the adverb SHIFT to the desired y-axis interval between successive scans. Then invoke procedure SHOWS, which invokes GETSCAN for each scan in the stack, and applies a BIAS equal to the appropriate multiple of SHIFT:

```
PROCEDURE SHOWS
XSCAN = ABS( ASTACK(1) ); GETSCAN XSCAN XX
FOR I=2 TO ACOUNT
   XSCAN = ABS( ASTACK(I) ); GETSCAN XSCAN
   FACT = ( I-1 )*SHIFT; BIAS RESHOW
   END
RETURN
FINISH
```

SHOW1 and SHOW2 are similar to SHOWS, except that instead of using GETSCAN to retreive the data, SHOW1 retrieves only the onscan and SHOW2 retrieves only the offscan of the onscan numbers in the stack. Notice again the multiplication/adding by ten of the stack's onscan number:

```
PROCEDURE SHOW1
XSCAN = ABS( ASTACK(I) ); GET (XSCAN*10) XX
FOR I=2 TO ACOUNT
   XSCAN = ABS( ASTACK(I) ); GET (XSCAN*10)
   FACT = ( I-1 )*SHIFT; BIAS RESHOW
   END
RETURN
FINISH

PROCEDURE SHOW2
XSCAN = ABS( ASTACK(1) ); GET (XSCAN*10+10) XX
FOR I=2 TO ACOUNT
   XSCAN = ABS( ASTACK(I) ); GET (XSCAN*10+10)
   FACT = ( I-1 )*SHIFT; BIAS RESHOW
   END
RETURN
FINISH
```

Averaging Scans in the Scan Number Stack: CB

Use the procedure CB to average together, using default weighting, all of the scans listed in the scan number stack and display the result:

```
PROCEDURE CB
ALL
FOR I = 1 TO ACOUNT
   GETSCAN( ASTACK(I) ) ALIGN ACCUM
   END
AVE TRW XX
RETURN
FINISH
```

Setting Baseline Fitting Windows: NRSET(NUM)

If you are unhappy with the baselining produced with a single fitting region at each end of the spectrum and want to set values for NREGION(RCVR,I) instead, use the procedure NRSET. The single argument for procedure NRSET is NUM, the number of baseline fitting windows you wish to specify. NUM may not exceed four. NRSET will set windows only for the receiver currently specified - that is, for the receiver displayed when you invoke SHOW. If the variable RCVR is zero, then the procedure will set RCVR to 1. When you invoke NRSET, the crosshairs are activated. Simply use the thumbwheel to position the vertical crosshair to the edges of the windows you wish to specify, and strike the SPACE bar. The prompt character will re-appear automatically after you have entered the specified number of windows. Remember that each pair of channels specifying a window must be specified so that the first channel number is less than the second channel number.

```
PROC NRSET(NUM)
IF RCVR < 1 THEN RCVR = 1;END;
J = 0
FOR I = 1 TO NUM
   J = J+1
   NREGION(RCVR,J) = CCUR
   J = J+1
   NREGION(RCVR,J) = CCUR
   END
RETURN
FINISH
```

Restoring Adverb Defaults: ADVSET, RESET, and RESET36

Three procedures have been built in to allow you to return most system defined adverbs to their default values without taking the drastic step of invoking a RESTART, which also deletes you procedure definitions. The procedure ADVSET restores the defaults for many of the reference line options that may appear on the plot:

```
PROC ADVSET
VMARK = -99999; FMARK = -99999
TMARK = -99999; CMARK = 0
RETURN
FINISH
```

To invoke ADVSET and return the number of points dropped from each end of the spectrum to values appropriate for autocorrelator data, use RESET:

```
PROC RESET
BDROP = 10; EDROP = 10; ADVSET
RETURN
FINISH
```

To invoke ADVSET and return the number of points dropped from each end of the spectrum to values appropriate for filter bank data, use RESET36:

```
PROC RESET36
BDROP = 0; EDROP = 0; ADVSET
RETURN
FINISH
```

Chapter 10

Off-Line Data Display

The POPS / Spectral Line program has several verbs to aid the user in making Calcomp plots and/or pictures with the Dicomed image recording device. These verbs are PLOT, PROFILE, and REPLOT to make Calcomp spectra plots, MAP and LABEL to make Calcomp velocity vs. position contour maps, and PIX and LABEL to make DICOMED velocity vs. position contour map pictures. (In Charlottesville, the verbs MAP and PIX are available only on the MODCOMP).

All of these verbs were implemented in such a way as to interface with the plotting routines of the older IBM TPOWER/SPOWER package of programs. However, because at present none of the LINE programs can communicate directly with the IBM OS/MFT system, where the Calcomp plotting package resides, the LINE verbs write data and other control information to a magnetic tape. This tape must then be read by a program which runs under OS/MFT to initiate and pass data to the appropriate job step of the TPOWER/SPOWER Calcomp package.

At Charlottesville, a small supply of plot tapes are available and can be found in Room 215 or on top of the large metal shelves in the first floor hallway near the door to the computer room. These tapes are small tapes and are labeled PLT01, PLT02, etc.

11.1. The Plotting Verbs on the IBM

On the IBM, the plotting verbs write to a disk file, and after exiting from the LINE program, you must execute the PLOTTAPE program to write the disk file to tape. (See the IBM section of Chapter 5 on how to write plot tapes.)

11.2. The Plotting Verbs on the MODCOMP

On the MODCOMP, using the various LINE verbs require that you have mounted a tape with a write-ring on a tape drive. In Room 215, there are two tape drives available, MT1 and MT2. The default drive for the program is MT2. (See the section in Chapter 5 about starting the LINE program on the MODCOMP if you wish to change the default.)

If working on the MODCOMP, thread the tape according to the diagram on the drive, and press the LOAD button. The tape will then advance to its load point. These tape drives are dual-density drives. The light on the button labeled "PE/NRZI" on these drives should be on. If not, push the button. This selects 1600 bpi recording. If you have been using the plot verbs and then notice that the light is off (indicating 800 bpi), DO NOT at this point change the density. The IBM tape drives, on which this tape will later be read, test for the density of tapes and select the density needed to read the tapes. A tape recorded with mixed densities is unreadable.

The plotting verbs assume there are no previous plots on the tapes. If there are and you do not wish to overwrite them, then you must position the tape yourself. To do this, after loading the tape but BEFORE starting the LINE program, do the following:

```
$JOB
$ASS TAP MTn
$AVF TAP n
```

where MTn is either MT1 or MT2, depending on what tape drive you wish to use. Remember, the default drive for the LINE program is MT2. The "n" in the AVF (AdVance File) command is the number of files that you wish to skip.

The next step depends on whether you want to begin a new file or append new plots to the current file. If you wish to begin a new file, do nothing.

If you wish to append plots to the current file, type

```
$BKF TAP
```

Now start the LINE program. After you have made all of the plots, maps and/or pixs you need, terminate the program by typing EXIT. Before stopping, the program will write two end-of-file marks on the tape and rewind it. To take the tape off of the drive press the RESET button and then the REWIND button. The tape will unload.

Should the program abort before you type EXIT, and you decide not to continue and restart the program, you MUST write end-of-file marks on the tape yourself if you wish to process it. To do this, type:

```
$JOB
$ASS TAP MTn
$WEOF TAP
$WEOF TAP
$REW TAP
```

Now you are ready to process the tapes and make plots, using the procedure MCPLOT.

11.3. The Procedure MCPLOT

Before you submit your job, you must give your tape to the IBM Computer Operator.

All of the JCL to read the tape and run the various job steps has been collected in one procedure called MCPLOT. The only required parameter for the procedure's EXEC card is TAPE=name where "name" is the one-to-six alphanumeric label on the outside of the tape. If you expect a Dicomed pix tape as part of the output, you must also code PIXTAPE=name2 on the EXEC card, where "name2" can be name.

The job may be submitted to the computer by either cards or thourgh the PANDORA terminal system. (Visitors who do not have a PANDORA logon may use VISITOR, OVERFLO1, or OVERFLO2.) A sample job looks like

```
//PLOTS JOB (159,140),BROWN,CLASS=B,MSGLEVEL=1
/*SETUP        PLT01
/*MESSAGE      NINE TRACK MINI TAPE, NO LABEL, NO RING
//     EXEC MCPLOT,TAPE=PLT01
```

Notes:

1. All lines (or cards) must start in column 1.
2. There must be at lease one space between the keyword JOB and the left parenthisis.
3. The keyword EXEC must begin between columns 4 and 16.
4. The tape number or name on the SETUP card must start in column 16.
5. If submitted from cards, the SETUP and MESSAGE cards can be omitted.
6. If this job is started from Green Bank via PANDORA, the line

   ```
   /*ROUTE  PRINT REMOTE1
   ```

   should be placed between the JOB and SETUP lines. Otherwise the printout for the job will be printed in Charlottesville.

7. If a PIXTAPE were expected, the EXEC card would be coded

   ```
   //     EXEC MCPLOT,TAPE=PLT01,PIXTAPE=PIX
   ```

   If you leave out the PIXTAPE parameter, the operator will cancel your job.

Other parameters that can be coded on the EXEC card are

FILE=n	where n is the file on your plot tape that you wish to plot
PLTSIZE=0	if you want the spectra plots drawn on Oversize paper
MAPOUT=0	if you want or need to have your maps drawn on the Oversize paper.

The CLASS=B parameter and the two minute time limit should be sufficient for most jobs generating only spectra plots or for a job making one or two small Dicomed pictures or for a job making a small map. Jobs which make pictures using a large number of data points or with large values for IROW and ICOL will have to run a CLASS=C or CLASS=D.

If a job makes a map where the number of scans times the number of channels contoured is greater than 5000, it will have to be run as a CLASS=L job and MAPSIZE=L must be coded on the EXEC card. Notice that the TIME parameter on the JOB card must also be coded, such as

... CLASS=L,TIME=4, ...

Chapter 11

Utility Programs

Several OS/MFT utility programs are supported for the convenience of the user. ACPROG will list, dump, or copy a user tape, TIDY will copy a user tape removing duplicate copies of scans, CONVERT2/CONVERT3 converts MODEL II and MODEL III autocorrelator user tapes from the 140/300-foot telescope into the current format, and ATUCSON, BTUCSON, and FTUCSON convert ascii, binary and FITS keep tapes, respectively, from Tucson's 12-meter telescope. Each of these programs has a catalogued procedure and is simple to execute.

11.1. Listing, Copying, and Dumping User Tapes

If a user needed a complete listing of his Model IV autocorrelator user tape, he would use ACPROG in the following way:

```
//LIST    JOB (209,140),USERTAPE,MSGLEVEL=1,CLASS=C
/*SETUP   5555
/*MESSAGE 5555   NO RING,LABELED
// EXEC     ACPROG,NAME=BURTON,TAPE=5555,AC=1024
//SYSIN      DD *
   OPERATION='LIST USER TAPE';
```

If a user wished to copy his user tape, he would also use ACPROG:

To copy one 384 channel tape to another (standard format):

```
//COPYTAPE  JOB (209,140),DATA384,CLASS=B
/*SETUP   4142,3456
/*MESSAGE 4142   NO RING,  LABELED
/*MESSAGE 3456   RING IN,  LABELED
//   EXEC ACPROG,NAME=BURTON,TAPE=4142
//TAPEOUT  DD DSN=TPOWER.BURTON,UNIT=TAPE,VOL=SER=3456,
//        DISP=NEW,DCB=RECFM=F
//SYSIN      DD *
  OPERATION='COPY USER TAPE';
  BEGIN=2360 END=4580 SCAN#_ADJUSTMENT=3000;
```

To copy 1024 channel data to a "FORTRAN readable" tape:

```
//COPYTAPE  JOB (209,140),DATA1024,CLASS=B
/*SETUP      2421,1748
/*MESSAGE    2421      NO RING, LABELED
/*MESSAGE    1748      RING IN, LABELED
//   EXEC ACPROG,NAME=TURNER,TAPE=2421,AC=1024,OUTFMT=FORTRAN
//TAPEOUT  DD DSN=TPOWER.TURNER,UNIT=TAPE,VOL=SER=1748,
//        DISP=NEW,DCB=RECFM=VS
//SYSIN      DD *
  OPERATION='COPY USER TAPE';
  BEGIN=2360 END=4580 SCAN#_ADJUSTMENT=3000;
```

The term "FORTRAN readable" means that the tape can be read using unformatted read statements in an IBM FORTRAN program. Setting the DCB sub-parameter RECFM to the option VS causes the system to add certain control information to the data records so that a FORTRAN unformatted read statement can correctly process the records. Other than 8 bytes of control information at the beginning of each data block on the tape (there is one record per block), the data records are the same for the "standard" tape output and the "FORTRAN" tape output.

When using computers other than the IBM, it may not be possible to use the FORTRAN unformatted read statement. In any event, it will be necessary to convert the IBM floating point numbers to whatever format is used by another computer. It is also possible that the EBCDIC character strings in the data records will have to be converted. For some computers -- especially CDC's -- it will be necessary to convert the intergers also.

11.2. Sorting User Tapes

It may be necessary to reorder the scans on a user tape by a given parameter. This may be accomplished by using the TSORT program. For example, user tapes are generally ordered chronologically by scan number. The following procedure will produce a listing of the scans on tape 5555 in ascending order of source name and within each source name the scan numbers will appear in ascending order:

```
//SORT    JOB (209,140),USERTAPE,MSGLEVEL=1,CLASS=B
/*SETUP    5555
/*MESSAGE  5555   NO RING, LABELED
//      EXEC     TSORT,NAME=BURTON,TAPE=5555,AC=1024
//SYSIN     DD *
  A    SOURCE  SCAN_NUMBER
```

where A refers to ascending order ("D" would result in descending order)

SOURCE refers to the first sorting parameter and

SCAN_NUMBER is the second sorting parameter.

Most of the parameters in the header may be sorted and are listed below:

SCAN_NUMBER	OFF_SCAN	OBS_NAME	SOURCE
MONTH	DAY	YEAR	TYPE_OBS
OBS_NUMBER	SAMPLE_RATE	TELESCOPE	POS_CODE
ZERO	TYPE_SCAN	AMOUNT	PAD1
SNI	VREF	VDEF	Z_DIST
RA_RATE	DEC_RATE	EPOCH	ORIENT
FOCUS	RA_IND	DEC_IND	LST
EST	JND	REST_FREQ(1)	REST_FREQ(2)
REST_FREQ(3)	REST_FREQ(4)	CNTR_FREQ(1)	CNTR_FREQ(2)
CNTR_FREQ(3)	CNTR_FREQ(4)	TSYS(1)	TSYS(2)
TSYS(3)	TSYS(4)	NOISE_TUBE(1)	NOISE_TUBE(2)
NOISE_TUBE(3)	NOISE_TUBE(4)	INTEGRATION	CNTR_VEL(1)
CNTR_VEL(2)	CNTR_VEL(3)	CNTR_VEL(4)	DEL_VEL(1)
DEL_VEL(2)	DEL_VEL(3)	DEL-VEL(4)	RVSYS
RVSUN	RA_APP	DEC_APP	RA_1950
DEC_1950	L_GAL	B_GAL	BW(1)
BW(2)	BW(3)	BW(4)	OBS_PSN(1)
OBS_PSN(2)	MODE	LM(1)	LM(2)
LU(1)	LU(2)	LL(1)	LL(2)
MF(1)	MF(2)	NF(1)	NF(2)

When sorting multiple parameters, the program sorts the data in the order that the parameters are given, but no more than five parameters may be used in any given job. If edited tapes (i.e., the same scan number appears more than once) are sorted, the order of the duplicate scans is reversed. See the description of TIDY to avoid this problem.

To generate a new sorted user tape, the user should submit a job similar to that given below:

```
//SORT    JOB (209,140),NEWTAPE,MSGLEVEL=1,CLASS=B
/*SETUP     5555,3456
/*MESSAGE   5555  NO RING, LABELED
/*MESSAGE   3456  RING IN, LABELED
//   EXEC   TSORT,NAME=BURTON,TAPE=5555,AC=1024
//SORTOUT   DD DSN=TPOWER.BURTON,UNIT=TAPE,DISP=NEW,
//    VOL=SER=3456
//SYSIN    DD *
  A TAPE SOURCE SCAN_NUMBER
```

When sorting a very large number of scans, the program may abort due to insufficient disk space allocated to the program. This space can be increased by coding SPACE=n (where n is some number greater than 6) on the EXEC card. For example:

```
//   EXEC   TSORT,NAME=BURTON,TAPE=5555,AC=1024,SPACE=8
```

A user should keep n as small as possible. If n becomes too large, the system may not be able to run the job.

11.3. Removing Duplicate Scans from User Tapes

TIDY is a convenient utility program for copying user tapes and removing duplicate scan numbers (the most recent version is copied). This program is especially useful for those wishing to sort their user tapes. The SORT program will reverse the order of the duplicate scans of the user tape, making tapes updated by editing of rewriting (KEEP) unusable.

TIDY will handle a maximum of 2300 scans and requires approximately 20 seconds of cpu time per 1000 scans. To execute TIDY, submit a job similar to the following:

```
//SHRINK  JOB (13,140),TURNER,MSGLEVEL=1,CLASS=B
/*SETUP    1957,1874
/*MESSAGE  1957     NO RING, LABELED
/*MESSAGE  1874     RING IN, LABELED
//  EXEC  TIDY,NAME=OHTURNER,INTAPE=1957,OUTTAPE=1874
```

11.4. Converting Old User Tapes

In October of 1974 the format of data from the 140-foot telescope changed from TPMOD2 and SPMOD2 to TPOWER and SPOWER. Also, in June of 1975, the 300-foot TPMOD3 and SPMOD3 formats were updated to TPOWER and SPOWER. In order to reduce TPMOD2, SPMOD2, TPMOD3, and SPMOD3 data with the current program, the data must be updated to the proper format with the programs CONVERT2 for the 140-foot data and CONVERT3 for the 300-foot data.

For example, to convert a pre-October 1974 TPMOD2 or SPMOD2 user tape to the TPOWER-SPOWER format, submit the following:

```
//CONVERT  JOB  (165,140),OLD140,MSGLEVEL=1,CLASS=B
/*SETUP    3524,4578
/*MESSAGE  3524     NO RING, LABELED
/*MESSAGE  4578     RING IN, LABELED
//   EXEC  CONVERT2
//STEPLIB  DD DSN=AUTOCORR.NEWLIB,DISP=SHR
//OLD      DD DSN=TPOWER.GORDON,VOL=SER=3524,UNIT=TAPE,DISP=OLD
//NEW      DD DSN=TPOWER.GORDON,VEL=SER=4578,UNIT=TAPE,DISP=NEW,
//        DCB=(RECFM=F,BLKSIZE=2496)
```

or to convert a 300-foot TPMOD3 or SPMOD3 user tape to the TPOWER-SPOWER format, use the following:

```
//CONVERT  JOB  (165,300),OLD300,MSGLEVEL=1,CLASS=B
/*SETUP    5698,9873
/*MESSAGE  5698      NO RING, LABELED
/*MESSAGE  9873      RING IN, LABELED
//   EXEC  CONVERT3
//STEPLIB  DD DSN=AUTOCORR.NEWLIB,DISP=SHR
//OLD      DD DSN=TPOWER.GORDON,VOL=SER=5698,UNIT=TAPE,DISP=OLD
//NEW      DD DSN=TPOWER.GORDON,VEL=SER=9873,UNIT=TAPE,DISP=NEW,
//        DCB=(RECFM=F,BLKSIZE=2496)
```

11.5. Converting Tucson's 12-Meter Tapes

There are several programs to convert the various types of tapes from the 12-Meter telescope in Tucson into the user tape format. The program TUCSON converts an ASCII tape; the program BTUCSON converts a binary tape; and the program FTUCSON converts a FITS keep tape.

11.5.1. ASCII Tape

To convert a 12-meter ASCII tape, submit the following job:

```
//ATUCSON  JOB(559,COMP),MORGAN,CLASS=B,MSGLEVEL=(1,1)
/*SETUP        TUCSON,1234
/*MESSAGE      TUCSON   NO RING, NO LABEL
/*MESSAGE      1234     RING IN, LABELED
// EXEC TUCSON,TELTAPE=TUCSON,USETAPE=1234,NAME=MORGN,STATUS=NEW
//LATEST     DD DCB=BUFNO=1
//SYSIN   DD *
   OBS#=319  START=200  STOP=1000;
```

11.5.2. Binary Tape

To convert a 12-meter binary tape, submit the following job:

```
//BTUCSON   JOB (459,COMP),WOOTTEN,CLASS=B,MSGLEVEL=(1,1)
/*SETUP     TUCSON,3888
/*MESSAGE   TUCSON      NO RING, NO LABEL
/*MESSAGE   3888        RING IN, LABELED
//   EXEC BTUCSON,TELTAPE=TUCSON,USETAPE=3888,NAME=WOOTN,STATUS=NEW
//SYSIN  DD *
  OBS#=459;
```

11.5.3. FITS Keep Tape

To convert a 12-meter keep tape, submit the following job:

```
//FTUCSON   JOB (459,COMP),WOOTTEN,CLASS=B,MSGLEVEL=(1,1)
/*SETUP     TUCSON,3908
/*MESSAGE   TUCSON      NO RING, NO LABEL
/*MESSAGE   3908        RING IN, LABELED
//   EXEC FTUCSON,TELTAPE=TUCSON,USETAPE=3908,NAME=WOOTN,STATUS=NEW
//SYSIN  DD *
  OBS#=459;
```

Appendix A.1

V E R B

A N D

O P E R A T O R

D I C T I O N A R Y

A

A will insert one scan into the STACK.

PTWH	Is not used or changed.

ADVERBS

ACOUNT	The STACK counter is incremented by one.
ASTACK	The specified scan is added to the STACK array.

OPERANDS

scan#	The scan number to be inserted into the STACK.

RELATED VERBS

ADD (#,#)	Inserts a group of scans into the STACK.
DELETE #	Deletes a scan from the STACK.
EMPTY	Empties the STACK.
STACK	Lists all scans in the STACK.

REMARKS

Positive scan numbers may be entered in the STACK. The operand may precede or follow A.

Multiple references to A on a single line must be separated by semicolons.

EXAMPLES

To add 650 to the scans already in the STACK specify:

```
650 A    or    A 650
```

ABS

A built-in absolute value function. Functions the same as the Fortran ABS.

PTWH Neither used nor changed.

ADVERBS None.

OBJECT

argument The argument may be a constant, a variable, or an arithmetic expression, enclosed in parentheses.

EXAMPLE

```
PRINT ABS( -4 )
 4.0000
```

ACCUM

A routine that is used to stack scans in the HOLD array. The first use of ACCUM after an AVE, SCLEAR or a program restart copies the header information and the data values in the WORK array to the HOLD array. The spectral values are multiplied by the integration time of the scan. Subsequent uses of ACCUM add the data values contained in the WORK array (weighted by the integration time divided by the system temperature squared) to the contents of the HOLD array. Subsequent uses of ACCUM do not change the header stored in the HOLD array, except for the integration time and the system temperature, which are summed each time a scan is accumulated. ACCUM increments the internal stack counter each time a scan is accumulated, and stores the numbers of the first 39 and the last scan accumulated.

PTWH — Must be pointing to WORK array.
Is set to the HOLD array.

ADVERBS

ASHIFT — The number of channels by which a scan is shifted by subsequent calls to ACCUM. On the first call to ACCUM, the scan is not shifted even if ASHIFT is not equal to 0.

RELATED VERBS

ALIGN — Sets the values of ASHIFT, using the header information in the WORK array, so that whin the scan in the WORK array is accumulated, its velocities will be correctly aligned with the velocities in the HOLD array.

AVE — Divides the accumulated scans by the number of scans accumulated. Sets the internal stack counter to zero. (See ** on the next page.)

SCLEAR — Sets the accumulator flag and internal stack counter to zero.

SUM — Accumulates scans with user supplied weighting.

TELL CSTACK — Prints the internal stack counter and the numbers of scans accumulated.

-continued-

REMARKS

The system temperatures are summed with the same weights as the spectra. A procedure using the STACK may facilitate the use of ACCUM.

EXAMPLE

To accumulate scans 602, 604, and 606 first specify:

```
SCLEAR
```

to empty the stack. (This is not necessary if no scans have been accumulated since the last AVE.) Then, to accumulate the scans, specify:

```
GET 602 ACCUM
GET 604 ACCUM
GET 606 ACCUM
```

If you now specify TELL CSTACK you will get:

```
3
602  604  606
```

To average the accumulated scans, specify:

```
AVE
```

** The internal stack and counter are used by ACCUM and AVE for bookkeeping purposes and are stored in the header of the accumulated scan. They are not to be confused with ASTACK and ACOUNT, the POPS adverbs which are defined by the observer and can be used to stack scans.

ADD

A routine which inserts a series of scan numbers into the STACK.

PTWH — Is not used or changed.

ADVERBS

ACOUNT	The STACK counter is incremented by the number of scan numbers inserted.
ASTACK	The scan numbers are inserted into the adverb array ASTACK.
SINCR	The increment between scan numbers.

OPERANDS — Beginning and end scan number.

RELATED VERBS

A #	Inserts one scan into the STACK.
DELETE #	Deletes one scan from the STACK.
EMPTY	Empties the STACK.
STACK	Lists the STACK.

REMARKS

The operands may precede or follow ADD. If they follow ADD, they must be enclosed in parentheses. Multiple references to ADD on a single line must be separated by semicolons.

EXAMPLE

To accumulate the first channel of scans between 500 and 600 type:

```
500 600 ADD      or      ADD (500,600)
```

ALIGN

ALIGN calculates the number of channels by which the scan in the WORK array must be shifted (when it is accumulated into the HOLD array) so that the data points at equal velocites will be added together. ALIGN sets the values of the adverb ASHIFT, which is used by the verbs ACCUM and SUM.

PTWH	Is set to the HOLD array.

ADVERBS

ASHIFT (rcvr)	The number of channels by which the velocities in the WORK array are out of alignment with the velocities in the HOLD array. ASHIFT is set by ALIGN.

RELATED VERBS

ACCUM	Adds the contents of the WORK array to the contents of the HOLD array with weighting by one over Tsys squared. Used to accumulate scans.
AVE	Divides the accumulated scans in the HOLD array by the sum of their integration times. Zeros the number of scans accumulated.
SCLEAR	Clears the accumulator (HOLD) array before stacking data.
SUM	Adds the contents of the WORK array to the contents of the HOLD array with weighting by a user-specified weight.
TELL CSTACK	Prints the stack counter and the scan numbers of the scans accumulated.

REMARKS

ALIGN will set ASHIFT = 0, if no scans have been accumulated since the last AVE. ALIGN can be used to accumulate scans taken with different LO settings.

The default alignment for ALIGN is frequency.

-continued-

EXAMPLE

Suppose you want to average two scans, but the velocity (frequency) scales are slightly different. The ALIGN routine looks at the local oscillator frequencies in the header of the scan in the currently referenced array, compares those with the frequencies in the header of the averaged data being accumulated in the HOLD array, and calculates the number of channels by which each receiver should be shifted so that its velocity scale matches that of the first scan accumulated into the HOLD array. The results ar to put into the adverb array ASHIFT(rcvr). If the scan number stack is empty, then ALIGN will result in ASHIFT = 0. Therefore, it can do no harm to invoke ALIGN every time before accumulating another scan into the average.

For example, consider the illustration given above when discussing the receiver processing option: You want to average four scans, but only one receiver (C) of one of them, and only three veceivers (A, B, and D) of another. We could alter that example by inserting the verb ALIGN before each ACCUM to insure that the result would be on a consistent velocity scale:

```
 AVE
AVE EMPTY?
 GET 7238 ACCUM
 GET 7239 ONLYC ALIGN ACCUM
 GET 7240 ONLYA ALIGN ACCUM
 ONLYB ALIGN ACCUM ONLYD ALIGN ACCUM
 GET 7241 ALIGN ACCUM AVE XX
```

If all of the scans were already on the same velocity scale, then no harm has been done, and if some of the scans were on a slightly different velocity scale, then the correct result is produced.

ALL

ALL sets the receiver processint option to all of the receivers of a scan in the currently referenced array. ALL is normally the default for a scan loaded from disk. (Scans reloaded from a KEEP tape or recalled from a SAVE/RECALL bin have the option which was in effect before the KEEP or SAVE.)

PTWH Determines which scans's processing option is changed. It is not changed.

ADVERBS None.

RELATED VERBS

ONLYA Sets the receiver processing option to the first receiver.

ONLYB Sets the receiver processing option to the second receiver.

ONLYC Sets the receiver processing option to the third receiver.

ONLYD Sets the receiver processing option to the fourth receiver

PAIR(m,n) Sets the receiver processing option to receivers m and n.

REMARKS

ALL is a rename of the old verb BOTH. BOTH is no longer a verb.

ALL should not be used for one-receiver data.

ALL is equivalent to SET1 = 1; SET2 = #rcvrs; SET3 = 1; SELECT.

ARRAY

This routine is used in a procedure to declare an array. It requires as its object the name of the array that is to be constructed and the dimensions of the array. An array may have any number of dimensions, limited only by the amount of space available for variable declaration. The size of the array in any dimension may be specified in two ways:

(SCALAR) or (SCALAR TO SCALAR)

SCALAR may be either a constant or a variable which has been assigned a value, but may not be an expression.

PTWH Is not used or changed by ARRAY.

ADVERBS None.

RELATED VERBS None

REMARKS

ARRAY can be used to set up an array in which results of operations can be stored. It can only be used in procedures. Arrays defined in procedures cannot be edited.

EXAMPLES

(1) valid definitions:

```
ARRAY ALPHA(N)
ARRAY BETA(-1 TO +1), GAMMA(3,5)
ARRAY DELTA(6 TO LEVEL, 8 TO 10)
```

(2) You want to store a number of results and print them all out at one time. For RMS calculations, for example:

```
PROCEDURE RMSCALC(FSCAN, LSCAN,N)
ARRAY RESULTS (N)
J = LSCAN - FSCAN + 1
FOR I = 1 TO J
   GET FSCAN + I - 1; RMS
   RESULTS (I) = VRMS(1)
END
PRINT RESULTS
FINISH
```

ATAN

A built-in arctangent function. Functions the same as the Fortran ATAN.

<u>PTWH</u> Neither used nor changed.

<u>ADVERBS</u> None.

<u>OBJECT</u>

argument The argument of ATAN is expected to be in radians. The argument may be a constant, a variable, or an arithmetic expression, enclosed in parentheses.

<u>RELATED VERBS</u>

see COS, SIN, and TAN

<u>EXAMPLE</u>

```
PRINT ATAN( 1.0 )
 0.7854
```

AUTO

AUTO returns the control of the y-axis scaling to the verb SHOW. Automatic scaling is the default, so AUTO is used to restore automatic y-axis scaling after the user has overridden it by means of the verb HOLDY, or by setting the values of the adverbs YMIN and YINCR by using the verb RANGE or with assignment statements.

PTWH — Is not used or changed.

ADVERBS

YMIN — The minimum temperature to appear on the graph. AUTO sets YMIN to -99999.9. SHOW sets its own y-axis scaling if YMIN is less than -9999 but greater than -9.E9.

RELATED VERBS

FIX — Alias for HOLDY (below).

FREEY — Same as AUTO.

HOLDY — Causes SHOW to retain the last determined y-axis scaling.

RANGE — Defines the values of YMIN and YINCR according to the specified max and min values.

AVE

AVE divides the accumulated scans in the HOLD array by the accumulated weight. The result is left in the HOLD array. The integration time is set to the sum of the integration times, and the system temperature to the weighted mean. AVE also zeros the stack counter.

PTWH	Current value is not used. Is set to the HOLD array.
ADVERBS	None.

RELATED VERBS

ACCUM	Adds a scan to the HOLD array (with weighting by the integration time divided by the sytem temperature squared).
SUM	Adds a acan to the HOLD array (with weighting by integration time times a user specified weight).
TELL CSTACK	Prints the stack counter and the scan numbers of the scans accumulated.

ERRORS

If the HOLD (accumulator) array is empty when AVE is specified, the message AVE EMPTY? or UNAVAILABLE! will appear. This tells the user that there is nothing to average.

-continued-

EXAMPLES

(1) see example with ACCUM

(2) to stack a number of scans, or two groups of scans, you might define the following procedure:

```
PROCEDURE STACKSCANS (FSCAN, LSCAN)
FOR I = FSCAN TO LSCAN BY 2
   GET (I) ACCUM
END
RETURN
FINISH
```

Then you could use the procedure to stack a few scans in the HOLD array:

```
STACKSCANS (700, 720)
STACKSCANS (760, 782)
```

When you have stacked all the scans you want, you can obtain the average and display it by

```
AVE PAGE SHOW
```

BASELINE

This routine computes by a least squares fit the coefficients of a Chebyshev polynomial of a specified order for a specified region or regions of the scan in the currently referenced array. The polynomial is then evaluated for each point and subtracted from the scan.

<u>PTWH</u>	Determines which scan has a baseline removed. It is not changed.

<u>ADVERBS</u>

NFIT	The order of the polynomial baseline to be removed. NFIT must be between 1 and 12.
BDROP(CH) EDROP(CH)	The number of points at each end of each receiver channel that will be ignored by the routine if NREGION(CH,1) = 0.
NREGION(CH,N)	Specifies the edges of a region or regions of the scan to be used to compute the baseline coefficients. Up to four regions may be specified for each receiver channel. N is an integer between 1 and 8.
BBASE(CH) EBASE(CH)	The number of points at each end of each receiver channel (not including the points dropped by BDROP and EDROP) which will be used to compute the baseline coefficients if NREGION(CH,1) = 0.

<u>RELATED VERBS</u>

BSHAPE	Computes the coefficients of the Chebyshev polynomial.
BMODEL	Computes the point by point values of the baseline using the last computed coefficients of the Chebyshev polynomial.
DCBASE	Computes the average value of the data for a specified range of points and subtracts this from all points in the scan.
MDBASE	Computes the median values over a set of data points centered on each point in a scan, and subtracts the medians from the data.
PCBASE	Computes and subtracts from the data that constant which makes a specified percentage of the data points negative.

-continued-

REMARKS

The regions used to compute the baseline should not include a known or suspected source structure.

ERRORS

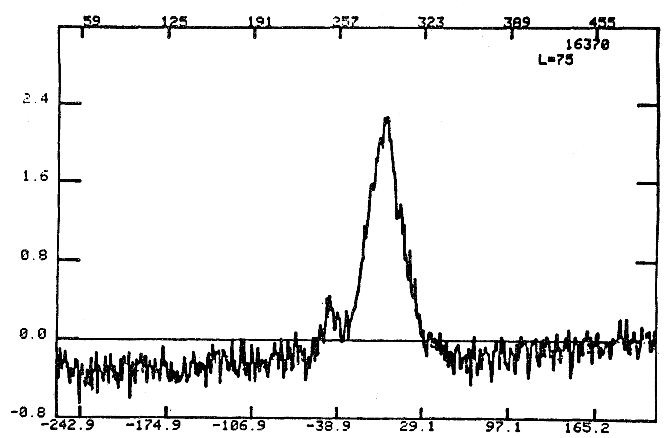
If NREGION(CH,N), where N is even, is less than NREGION(CH,N-1), the message

NREGION?

will appear. (The endpoint of a baseline fitting region must be greater than the startpoint.)

EXAMPLE

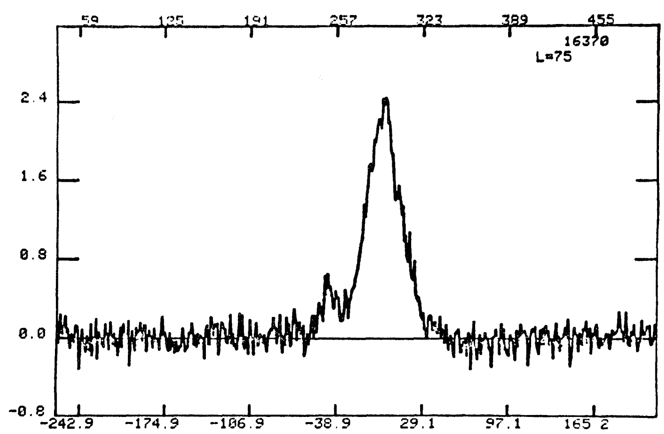
Your data looks like:



You want to remove a fifth order baseline. Specify:

```
NREGION (1,1) = 50
NREGION (1,2) = 230
NREGION (1,3) = 330
NREGION (1,4) = 495
NFIT = 5
BASELINE PAGE SHOW
```

You will get:



BIAS

BIAS adds the values of FACT(CH) to the scan in the currently referenced array. The array index is the receiver number.

<u>PTWH</u> Determines to which scan FACT is added. It is not changed.

ADVERBS

FACT(CH) The number of degrees Kelvin which is to be added to a scan. FACT is also used by the verb SCALE.

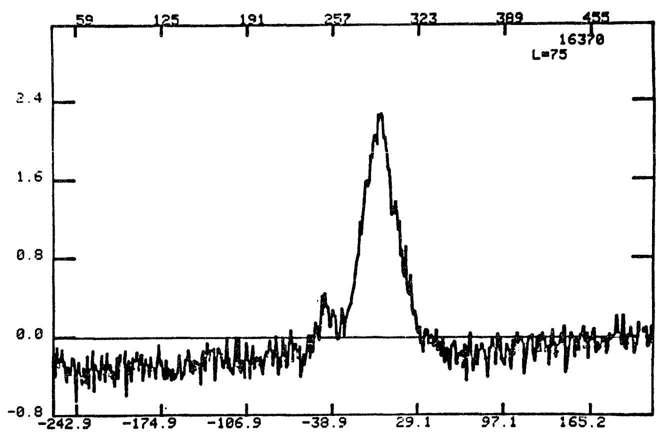
RELATED VERBS

SCALE Also uses the adverb FACT. SCALE multiplies the data in the currently referenced array by the appropriate FACT(CH).

EXAMPLE

BIAS can be used to plot more than one scan on the same graph, separated for visibility. Specify

GET 16370 PAGE SHOW



-continued-

Set the Y-scaling so that the first scan will be plotted at the bottom of the graph. For the graph below,

```
-1 6 RANGE
```

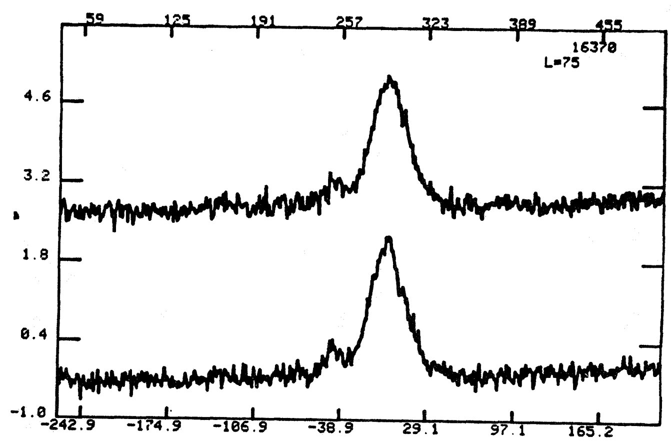
was used. Replot. Then set FACT to separate the scans:

```
PAGE SHOW
GET 16380
FACT = 3.0
```

and specify

```
BIAS RESHOW
```

The results look like:



BMODEL

BMODEL evaluates the last computed coefficients for a polynomial baseline for each point and replaces the currently referenced array with the evaluated polynomial.

<u>PTWH</u> Determines which array is replaced by the baseline model. Is not changed.

<u>ADVERBS</u> None used directly -- BMODEL uses the results of the previous BASELINE or BSHAPE.

<u>RELATED VERBS</u>

BASELINE Computes the coefficients for the polynomial, evaluates the polynomial and subtracts it from the currently referenced array.

BSHAPE Computes the coefficients for the polynomial using the currently referenced array.

<u>REMARKS</u>

Most often used in conjunction with BSHAPE to compute a baseline without subtracting it from the data. It is often useful to plot the baseline model on top of the data for comparison purposes.

-continued-

EXAMPLE

You have twenty scans of the same source and you want to remove exactly the same baseline from each of them. You can do this by

```
GET 1710 TRT BSHAPE BMODEL PMW DIFF SHOW
GET 1712 DIFF PAGE SHOW
GET 1714 DIFF PAGE SHOW
      .
      .
      .
GET 1748 DIFF PAGE SHOW
```

The first set of commands above create a model of the baseline in the TEMP array and subtract the result from the scan in the WORK array. The result of the subtraction is left in the WORK array and the TEMP array is not changed. Thus the model can be subtracted from scan after scan.

BOXCAR

This routine smoothes a scan by averaging an odd number of consecutive points together and placing the result in the center point.

PTWH — Determines which scan is smoothed. Is not changed.

ADVERBS

BDROP(CH)
EDROP(CH) — The number of points at each end of each receiver channel which will be ignored by the routine.

NBOX — The number of points which will be averaged together to smooth the scan.

RELATED VERBS

HANNING — A smoothing routine which averages three points together with the center point getting twice as much weight as either side point.

SMOOTH — A smoothing routine which averages data according to user specified weighting.

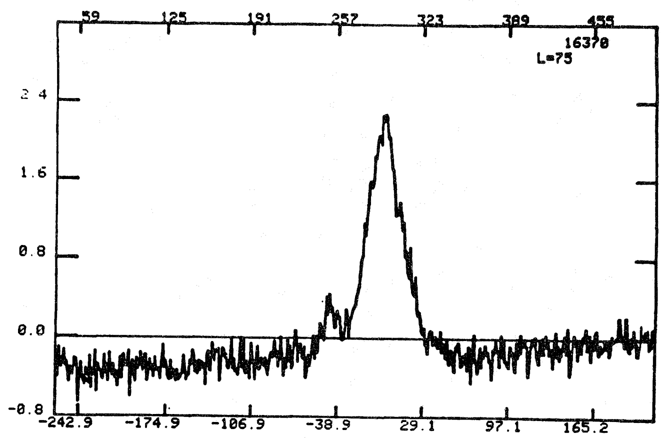
ERRORS

If NBOX is not odd, the error message "NBOX ODD?" is printed, and the BOXCAR operation is not performed.

-continued-

EXAMPLE

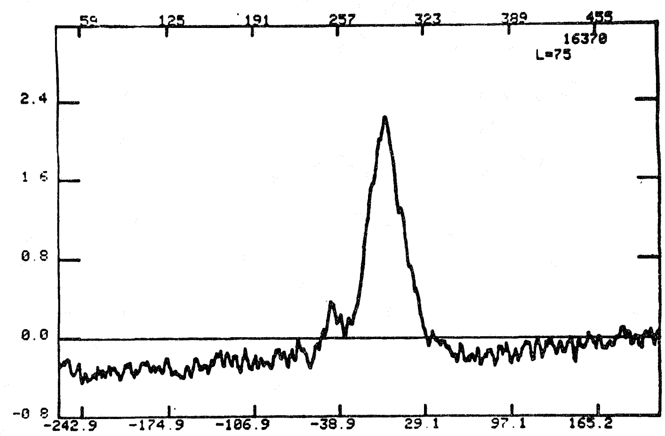
Your data looks like the scan on
the right and you want to smooth it.



If you specify:

NBOX = 3
BOXCAR PAGE SHOW

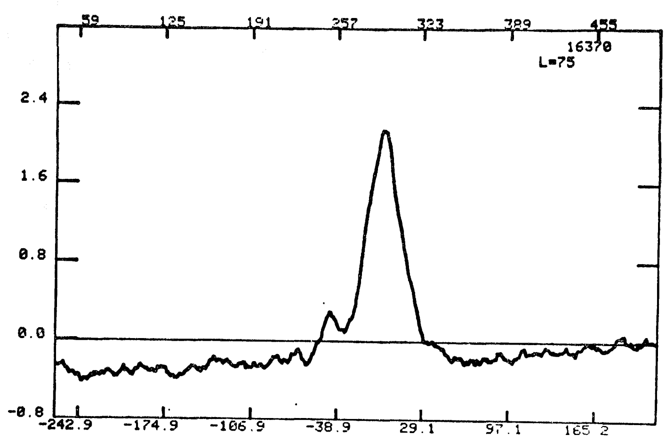
you will get the display to the
right.



If instead you specify:

NBOX = 7
BOXCAR PAGE SHOW

you will get the display to the
right.



BSHAPE

BSHAPE calculates the coefficients of a Chebyshev polynomial of a given order by a least squares fit for a specified region or region of spectral values. BSHAPE does NOT evaluate the polynomial. (BASELINE and BMODEL do.)

PTWH	Determines the scan to which the baseline is fit. It is not changed.

ADVERBS

NFIT	The order of the polynomial baseline to be removed. NFIT must be between 1 and 12.
BDROP(CH) EDROP(CH)	The number of points at each end of each receiver channel that will be ignored by the routine if NREGION(CH,1) = 0.
NREGION(CH,N)	Specifies the region or regions of the scan to be used to compute the baseline coefficients. N is an integer between 1 and 8.
BBASE(CH) EBASE(CH)	The number of points at each end of each receiver channel (not including the points dropped by BDROP and EDROP) which will be used to compute the baseline coefficients if NREGION(CH,1) = 0.

RELATED VERBS

BASELINE	Not only computes the coefficients but evaluates them for each point and subtracts the evaluated polynomial from the data.
BMODEL	Evaluates a polynomial using the last computed coefficients and replaces the data with the evaluated polynomial.

-continued-

REMARKS

BSHAPE is most frequently used in conjunction with BMODEL to construct a model of a particular baseline. The set of commands

TRT BSHAPE BMODEL PMW DIFF

is equivalent to BASELINE.

EXAMPLE

You want to remove a parabolic baseline from scan 1718, which is in the WORK array, but before you subtract the parabola from the scan, you want to see how well it fits. To copy the scan to the TEMP array, compute the polynomial coefficients, and replace your copy of the scan with the computed polynomial, specify:

TRT BSHAPE BMODEL

To plot the baseline model over the scan already plotted on the screen:

RESHOW (result is in Figure 1)

To subtract the model from the data in the WORK array and plot the results:

PMW DIFF PAGE SHOW

When you are done, you will have something like Figure 2:

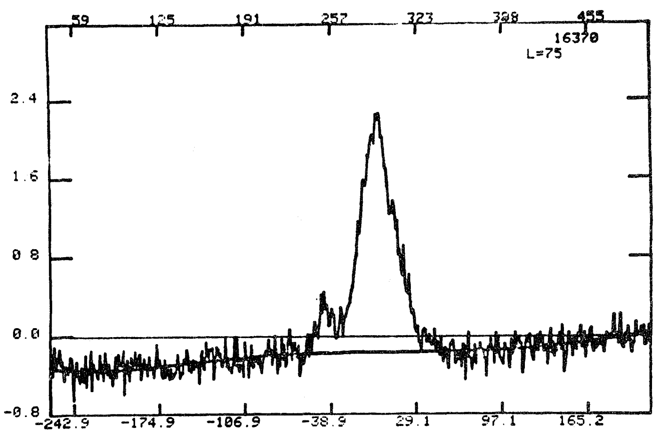


Figure 1

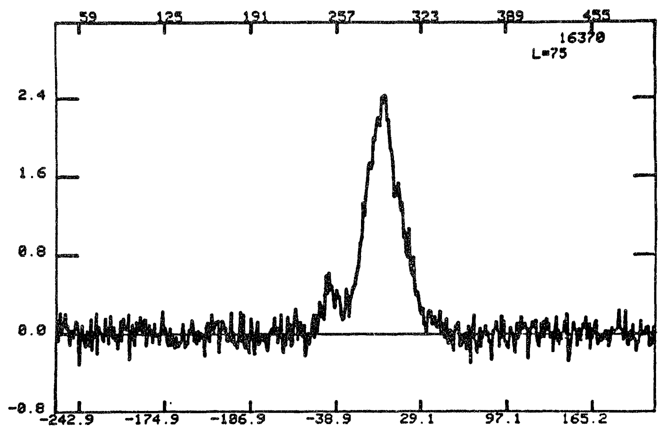


Figure 2

CCUR

This routine activates the vertical crosshair and returns the sample number for that crosshair position on the current display when any key except RETURN is struck.

PTWH — Is not used or changed by CCUR.

ADVERBS — None.

RELATED VERBS

CROSSHAIR — Activates the horizontal and vertical crosshairs and prints the point number, position, and temperature values at crosshair positions.

TCUR — Returns the temperature value at the current horizontal crosshair position.

VCUR — Returns the velocity at the current vertical crosshair position.

REMARKS

CCUR is best used in a procedure, generally for defining baseline regions or for entering initial guesses for fitting gaussians.

CCUR returns a zero value if the RETURN key is struck.

EXAMPLE

A procedure for setting gaussian parameters with the crosshairs is defined below:

```
PROCEDURE GSET
PRINT 'ENTER # OF GAUSSIANS'
READ NGAUSS
PRINT 'SET CENTERS'
FOR I = 1 TO NGAUSS
    X1 = CCUR;  CENTER(I) = X1;  END
PRINT 'SET HWIDTHS'
FOR I = 1 TO NGAUSS
    X1 = CCUR;  X2 = CCUR;  HWIDTH (I) = X2 - X1;  END;
PRINT     'SET BGAUSS & EGAUSS'
X1 = CCUR;  BGAUSS = X1;  X1 = CCUR;  EGAUSS = X1
RETURN
FINISH
```

CFETCH

CFETCH is used to retrieve from disk the most recently completed scan along with its associated offscan. CFETCH copies the onscan from disk into the WORK array, and its associated offscan into the TEMP array.

PTWH	Is set to the WORK array.
ADVERBS	None

RELATED VERBS

CGET rcvr #	Copies the specified receiver of themost recently completed scan into the WORK array.
FETCH scan #	Copies the specified scan into the WORK array and the associated offscan into the TEMP array.
GET scan #	retrieves the specified scan from disk and places it in the WORK array.
ON scan #	retrieves the specified scan from disk and places it in the WORK array.
OFF scan #	retrieves the specified scan from disk and places it in the TEMP array.

ERRORS

The receiver number must be in the range 1 CH 4 or the message UNKNOWN? will be generated.

CGET

CGET is used to retrieve from disk the indicated receiver of the most recently completed scan. It requires the receiver number as its object. CGET copies the scan from disk into the WORK array.

PTWH	Not used but is set to the WORK array.
ADVERBS	None

OBJECT

Receiver #	The most recently completed scan for this receiver is copied into the WORK array.

RELATED VERBS

CFETCH	Copies the most recently completed scan into the WORK array and its associated offscan into the TEMP array.
FETCH scan #	Copies the first feed of the specified scan into the WORK array and the second feed into the TEMP array.
GET scan #	retrieves the specified scan from disk and places it in the WORK array.
ON scan #	retrieves the specified scan from disk and places it in the WORK array.
OFF scan #	retrieves the specified scan from disk and places it in the TEMP array.

ERRORS

The receiver number must be in the range $1 \leq CH \leq 4$ or the message UNKNOWN? will be generated.

EXAMPLE

You wish to retrieve from disk the most recently completed scan from the second receiver and display it on the screen; specify:

```
CGET 2 PAGE SHOW
```

CLIP

CLIP resets any data points in the currently referenced array with values greater than CLIPMAX to CLIPMAX and any data points with values less than CLIPMIN to CLIPMIN.

PTWH — Determines which scan is clipped. Is not changed.

ADVERBS

CLIPMAX — The maximum data value that will be found in a scan after it is clipped.

CLIPMIN — The minimum data value that will be found in a scan after it is clipped.

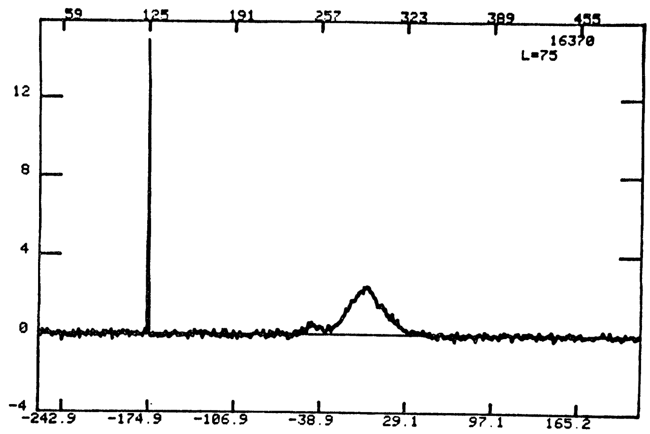
REMARKS

CLIP is commonly used to limit the range of data values in a scan.

-continued-

EXAMPLE

Your data look like the display to the right.

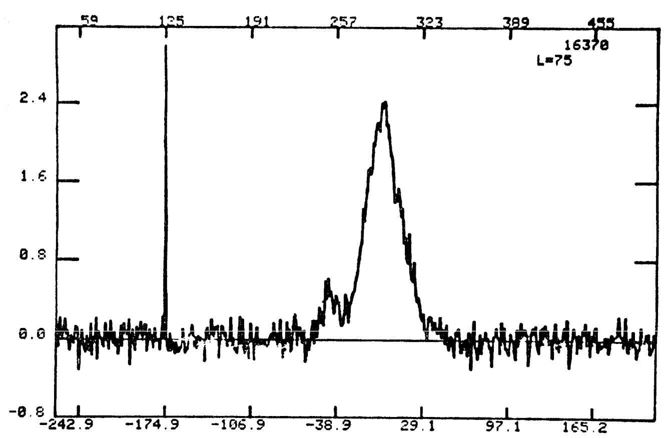


You are interested in the structure between the arrows.

Specify:

```
CLIPMIN = -1.0
CLIPMAX = 3.0
CLIP PAGE SHOW
```

and you will get the display to the right.



CORE

CORE prints the amount of memory available for procedure and variable definition.

PTWH — Is not used or changed by CORE.

ADVERBS — None.

RELATED VERBS

RESTART	Empties the procedure and variable space.
STORE page #	Stores the current procedures in disk page page#. page# must be between 1 and 6.
RESTORE page#	Retrieves disk page page#.

REMARKS

CORE is purely an informational verb. The amounts of space allocated for procedure source code, for compiled procedure code, and for variable definition are fixed. Filling up any of these spaces while you are defining a new procedure will give the error message BLEW CORE.

If you see the BLEW CORE error message, then you are stuck. You may continue with the procedures that you have defined already, but you will not be able to edit them. The only cure is to do a RESTART, which will delete all of the procedures that you have defined, and then start typing in your procedures again. Since editing a procedure uses additional memory space, the procedures may all fit if they can be typed in without editing, and if they were edited the first time around. Alternatively, you may decide which procedures may be grouped into two or more memory pages, stored separately on disk.

You can avoid getting the message BLEW CORE by not trying to define a procedure when there is very little space left in memory. You can protect yourself from the BLEW CORE condition to a limited extent by STOREing the procedure definition space after each new procedure is defined. If you then get the BLEW CORE message while defining a new procedure, you may RESTORE all your other procedures or RESTART and then put your new procedure on a new memory page.

-continued-

EXAMPLE

You have defined three procedures and you want to know if there is room for any more. You specify

```
CORE
```

and get the output

```
                SPACE AVAILABLE FOR
PROGRAM              VARIABLES           SOURCE
3387 4131           11277 11741         815 2048
```

This tells you that you have 744 words of program space, 464 words of variable space and 1233 words of procedure source space left.

COS

A built-in cosine function. Functions the same as the Fortran COS.

PTWH — Neither used nor changed.

ADVERBS — None.

OBJECT

argument — The argument of COS is assumed to be in radians. The argument may be a constant, a variable, or an arithmetic expression, enclosed in parentheses.

RELATED VERBS

see SIN, TAN, and ATAN

EXAMPLE

```
PRINT COS(0.7854)
 0.7071
```

CROSSHAIR

This routine activates the horizontal and vertical crosshairs. The horizontal and vertical lines can be positioned with the white thumb wheels at the lower right of the terminal keyboard. When the lines are in position, striking any key but the RETURN key causes the program to mark the intersection of the lines and print the channel number, velocity, and temperature at that point. The crosshairs are then reactivated and can be moved to a different position. To terminate the routine, strike the RETURN key.

<u>PTWH</u> Is not used or changed by CROSSHAIR.

<u>ADVERBS</u> None.

<u>RELATED VERBS</u>

CCUR	Returns the point value at the current vertical crosshair crosshair position.
TCUR	Returns the temperature value at the current horizontal crosshair position.
VCUR	Returns the velocity at the current vertical crosshair position.

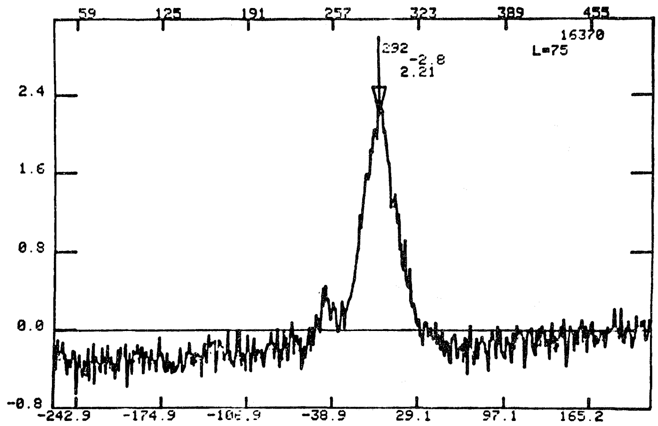
<u>REMARKS</u>

CROSSHAIR prints information concerning the crosshair positions on the CRT screen. CCUR and TCUR and VCUR derive a value from the appropriate crosshair and stores the value in the program to be used in setting adverb values.

-continued-

EXAMPLE

An example of the output from CROSSHAIR is shown below.



CRT

THIS VERB IS NOT IN EFFECT ON THE IBM

Used to direct the output of certain print routines to the CRT scree. Those verbs affected are PRINT, SUMMARY, TABLE, and TELL DISK.

PTWH   Neither used nor changed.

ADVERBS   None.

RELATED VERBS

PRINTER   Directs print output to the printer

REMARKS

CRT is the default print option. Using PRINTER may result in more readable copy than printing on the CRT screen and making a copy of the screen.

Once the output option is set to the CRT screen, it remains in effect until changed by the verb PRINTER.

DCBASE

DCBASE computes the average data value over a specified range of data values and subtracts the average from all the data values in the currently referenced array.

PTWH	Determines which scan has a baseline removed. Is not changed.

ADVERBS

BDROP(CH) EDROP(CH)	The number of points at each end of each receiver channel which are ignored by the routine if NREGION(CH,1) = 0.
NREGION(CH,N)	Specifies the edges of a region or regions of the data which are used to compute the average value. Up to four regions may be specified for each receiver channel. N is an integer between 1 and 8.
BBASE(CH) EBASE(CH)	Specifies the number of points at each end of each receiver channel (not including the points dropped by EDROP and BDROP) that will be used to compute the average data value if NREGION(CH,1) = 0.

RELATED VERBS

See BASELINE
BIAS

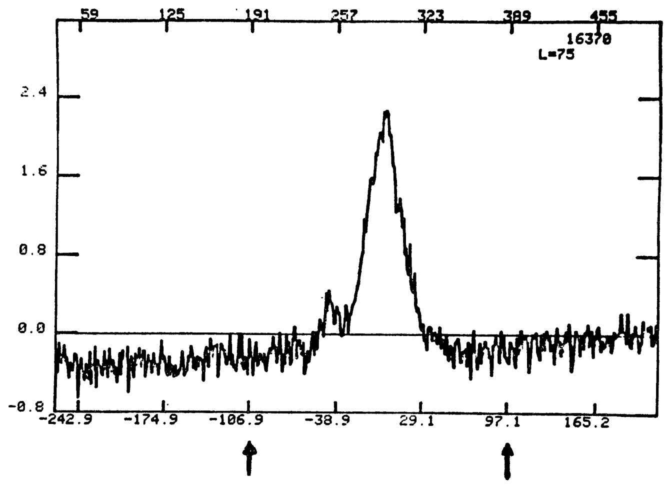
REMARKS

A conservative means of baseline removal since baseline slope is retained.

-continued-

EXAMPLE

Your data looks like

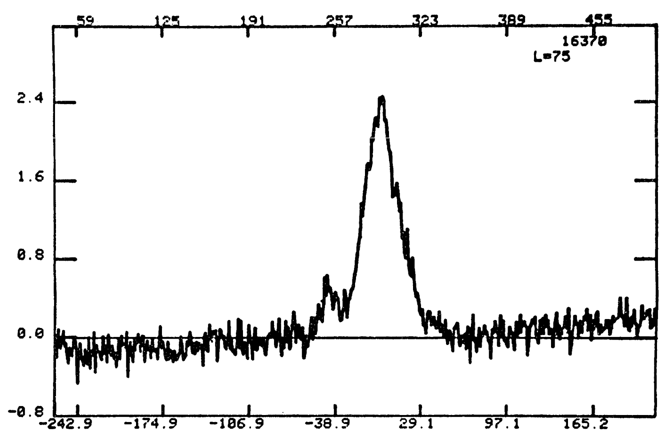


You want to remove an average computed over these regions

So you specify

NREGION (1,1) = 0
BBASE = 191
EBASE = 123
DCBASE
PAGE SHOW

And you will get



DEC

DEC sets an internal flag that causes LABEL to identify declination as the Y-coordinate and velocity as the X-coordinate for maps. This is the default condition on entry to the program. Would be used for drift scans or if the telescope were scanned along lines of constant declination.

PTWH Is not used or changed.

ADVERBS None.

RELATED VERBS

GB	Sets the y-coordinate label as galactic latitude.
GL	Sets the y-coordinate label as galactic longitude.
LABEL	Labels the map drawn by MAPSHOW.
MAPSHOW	Draws a map with the levels specified by LEVS.
RA	Identifies right ascension as the Y-coordinate and velocity as the X-coordinate for maps.

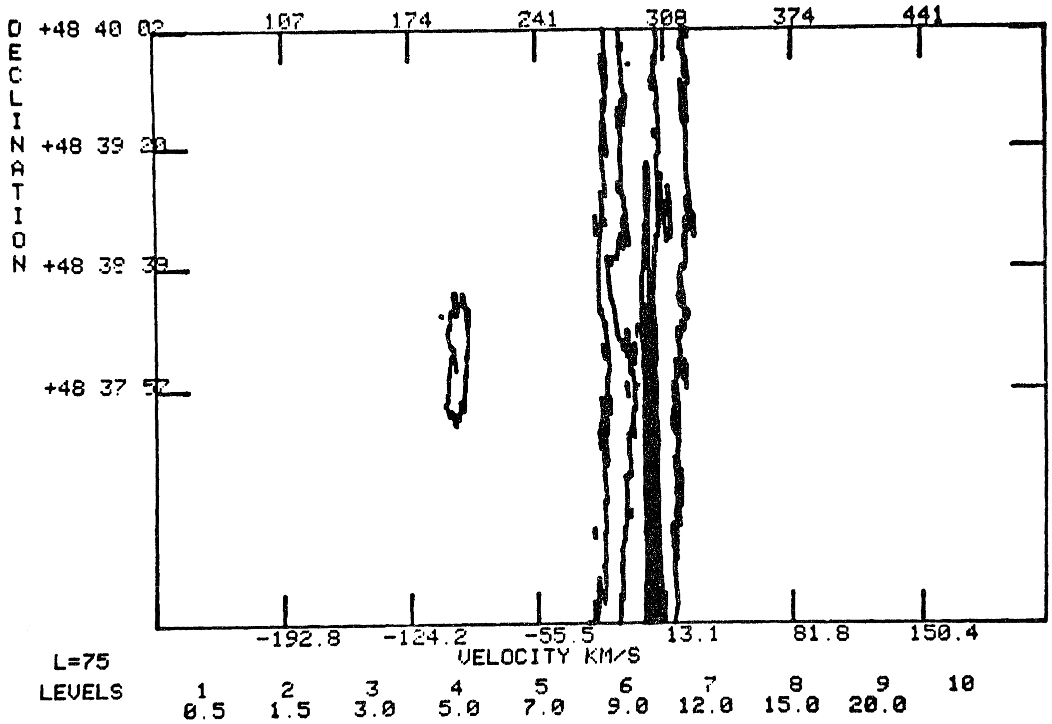
REMARKS

DEC does not affect the scaling of the y-axis. MAPSHOW and LABEL assume that the scans have been correctly spaced in the coordinate of interest.

-continued-

EXAMPLE

An example of a display is shown below:



DELETE

A routine which deletes one scan number from the STACK.

PTWH — Is not changed or used.

ADVERBS

ACOUNT	The STACK counter, which is decremented by one.
ASTACK	The specified scan number is deleted from the STACK array contained in the ASTACK array.

OPERANDS

scan#	The number of the scan to be deleted from the STACK.

RELATED VERBS

A scan#	Inserts one scan number into the STACK.
ADD (#,#)	Inserts a series of scan numbers into the STACK.
EMPTY	Empties the STACK.
STACK	Lists the STACK.

REMARKS

The scan number may be typed immediately proceeding or following the verb DELETE.

EXAMPLE

The STACK currently has scans 512-536 in it, but you want to delete scan 524. Type

```
524 DELETE
```

DELTAV

This routine calculates the area, mean flux, high and low velocity peak fluxes, high and low peak velocities, and the low, center, high, and delta velocities at specified fractions of the peak flux.

PTWH	Determines which array is used to compute the values. It is not changed.

ADVERBS

DFRACT	The increment between fractions for which velocites are calculated.
NFRACT	The number of fractions desired.
FRACTION	The lowest fraction for which to calculate the velocities.
BMOMENT(rcvr) EMOMENT(rcvr)	The first and last channels which define the window used to compute the area, mean flux, and velocities.

RELATED VERBS

MOMENT	Computes either the area or the centroid of the fearure.
GMEASURE	Calculates area, mean flux, high and low velocity peak fluxes, high and low peak delta velocities and the minimum flux and delta velocity for a specified region.

REMARKS

DELTAV is intended for galaxy profiles.

DELTAV will only process one receiver at a time; if the option ALL is used, DELTAV will process the A receiver.

ERRORS

See GMEASURE.

EXAMPLE

If you wish to calculate DELTAV's quanities at 20%, 45%, 70%, and 95% of the peak flux, specify,

```
NFRACT = 4; FRACTION = 20; DFRACT = 25; DELTAV
```

DIFF

This routine subtracts the scan in the TEMP array from the currently referenced array.

PTWH	Determines from which array the TEMP array is subtracted. It is not changed.
ADVERBS	None.

RELATED VERBS

DIVIDE	Computes Array / TEMP
RDIFF	Subtracts two receivers within the same scan.
TEMP	Computes ((WORK - TEMP) / TEMP) * Tsys.

REMARKS

This verb can be used to remove baselines by subtracting an off-source scan.

EXAMPLE

To difference a scan 4588 and an off-source scan 4575, and display the result

```
OFF 4575 ON 4588 DIFF PAGE SHOW
```

The most convenient way to add two scans is to put the negative of one of them into the TEMP array (by using the verb OFF, for example), and then use DIFF. For example, to add scans 4588 and 4575:

```
OFF 4575; FACT = -1; SCALE
ON 4588 DIFF PAGE SHOW
```

The same example using GET rather than ON and OFF is:

```
GET 4575; FACT = -1; SCALE TRT
GET 4588 DIFF PAGE SHOW
```

DIVIDE

This routine divides the scan in the currently referenced array by the scan in the TEMP array. The result is left in the currently referenced array.

PTWH — Determines which array is divided by the TEMP array. It is not changed.

ADVERBS — None.

EXAMPLE

To divide scan 548 by scan 550:

put the dividend scan in the WORK array:	ON 548
put the divisor scan in the TEMP array:	OFF 550
point to the dividend scan:	PMW
form the quotient:	DIVIDE
display the result:	PAGE SHOW

The same example using GET rather than ON and OFF is:

```
GET 550 TRT
GET 548 DIVIDE PAGE SHOW
```

EDIT

EDIT is used to change an already defined procedure. EDIT puts the program into EDIT mode and signals this to the user by prompting with a colon instead of a caret. EDIT requires as its object the name of the procedure to be edited and the number of the line in the procedure which is to be changed. If the specified line number matches a line in the procedure, then that line is replaced by the first line entered in the EDIT mode and subsequent lines are inserted at that point. If the line number is an interpolation between two existing lines, then no line in the procedure is replaced, and lines entered in the EDIT mode are inserted at that point.

PTWH — Is not used or changed by EDIT.

ADVERBS — None.

OBJECTS

Procedurename — The name of the procedure to be edited.

Line# — The line number of the line in the procedure which is to be changed, if an integer. Subsequent lines inserted after that line. If line# is not an integer, lines entered are inserted between the lines implied. See the third example.

RELATED VERBS

LIST procedurename — Lists the procedure with the given name, and supplies line numbers.

ENDEDIT — Gets the program out of EDIT mode.

MODIFY — Changes characters within a procedure line.

See also PROCEDURE.

-continued-

ERRORS

A fixed amount of space is available for defining procedures. This space is also used up by editing procedures. When the space has been completely used up, the program will give the message

BLEW CORE!

At this point you have three options:

(1) You can do without the procedure you were defining or editing when the message appeared. All other procedures will still be usable, but not able to be edited.
(2) You can wipe out all of your procedures (and adverb changes) by using RESTART. Then type in your correct procedures, hoping that if you don't do any editing, they will all fit.
(3) You can store your current set of procedures on the disk, using the STORE command. Then RESTART. Then define any additional procedures you need. Since RESTART kills your adverb values and array contents, if you have two "pages" of procedures (one on disk, and one in memory) which use each other's results, you will have to store intermediate results in scan save bins. Intermediate variable results (for example, SIZE) will be lost.

EXAMPLE

To edit the already defined procedure PRCDR, use LIST to see line numbers:

```
>LIST PRCDR
 1  PROCEDURE PRCDR(N)
 2  GET(N)
 3  BASELINE
 4  HANNING
 5  PAGE SHOW
 6  ACCUM
 7  RETURN
 8  FINISH
```

Suppose you want to change HANNING to BOXCAR:

```
>EDIT PRCDR 4
:BOXCAR
:ENDEDIT
```

-continued-

Now suppose you wanted to add a line to your procedure:

```
>EDIT PRCDR 1.5
:NBOX=3
:ENDEDIT
>LIST PRCDR
1  PROCEDURE PRCDR(N)
2  NBOX=3
3  GET(N)
4  BASELINE
5  BOXCAR
6  PAGE SHOW
7  ACCUM
8  RETURN
9  FINISH
```

You may also delete a line with:

```
>EDIT PRCDR 7     or      >EDIT PRCDR 7
:*                        :ENDEDIT
:ENDEDIT
```

EMPTY

This routine will delete all scannumbers from the STACK, ASTACK, and set the STACK counter, ACOUNT, to zero.

PTWH — Is not used or changed.

ADVERBS

ACOUNT — The STACK counter, which is set to zero by EMPTY.

ASTACK — The STACK array, which is zeroed by EMPTY.

RELATED VERBS

A scan# — Inserts a single scan number into the STACK.

ADD (#,#) — Inserts a group of scan numbers into the STACK.

DELETE scan# — Deletes a scan number from the STACK.

STACK — Lists all scan numbers in the STACK.

EXAMPLE

You have just stacked a group of scans and now wish to clear the STACK. Type:

```
EMPTY
```

END

END is used to end a logical construction. It is used in conjunction with the verbs IF, FOR, and WHILE.

PTWH It is not used or changed.

ADVERBS None.

RELATED VERBS

See FOR
IF
WHILE

ENDEDIT

ENDEDIT is used in conjunction with EDIT. EDIT gets the program into EDIT mode; ENDEDIT returns the program to EXECUTE mode. The program signals that it is in EXECUTE mode by prompting with a caret.

PTWH Is not used or changed.

ADVERBS None.

RELATED VERBS

See EDIT

EQTOGAL

EQTOGAL will convert equatorial coordinates (epoch 1950.0) into galactic coordinates and print the results on the CRT screen.

PWTH	Is neither used nor changed.
ADVERBS	None.

OPERANDS

right ascension	Epoch 1950.0 right ascension in format HHMMSS.S.
declination	Epoch 1950.0 declination in format ±DDMMSS.S.

RELATED VERBS

GALTOEQ	Converts galactic coordinates to equatorial coordinates (epoch 1950.0) and prints the results on the screen.

REMARKS

The operands should follow EQTOGAL and be enclosed in parentheses, as in

EQTOGAL (HHMMSS.S, ±DDMMSS.S).

EXAMPLE

To convert the position of SS433, RA = $19^h09^m21.^s3$, DEC = $4^o53'53.1$, from equatorial to galactic coordinates, specify:

EQTOGAL (190921.3,45353.1)

the galactic coordinates are printed as,

39.694 -2.245

EXIT

EXIT is the routine that terminates the program.

<u>PTWH</u> Is not used or changed.

<u>ADVERBS</u> None.

<u>RELATED VERBS</u> None.

<u>REMARKS</u>

Before you EXIT, take inventory of the items that will be lost when you EXIT. The contents of the three scan registers and the procedure definition space will be lost. To preserve any of the scan registers, save their contents in SAVE/RECALL bins. To preserve your procedure definitions and adverb values, use the STORE verb to make a copy of the procedure definition space in one of the six disk pages.

It is necessary to use EXIT whin writing a plot tape to be used offline. EXIT writes and end-of-file on the tape and rewinds it. Also, it may be necessary to exit from the program so the the full KEEP file may be dumped to tape.

<u>EXAMPLE</u>

You have just finished your observing run and wish to take home a KEEP tape of reduced data. You must exit CONDAR in order to be able to run the KEEP program. To do so simply type:

EXIT

EXP

A built-in exponential function. Functions the same as the Fortran EXP.

<u>PTWH</u> Neither used nor changed.

<u>ADVERBS</u> None.

<u>OBJECT</u>

argument The argument of EXP may be a constant, a variable, or an arithmetic expression, enclosed in parentheses.

<u>RELATED VERB</u>

LN A built-in logarithm function.

<u>EXAMPLE</u>

```
PRINT EXP( 1.0 )
2.7183
```

FETCH

FETCH is used to retrieve from disk a specified scan and place it into the WORK array, while also retrieving the associated off-scan, placing it into the TEMP array. FETCH requires one object, which is the scan number of the first scan to be retrieved.

PTWH	Not used, but is set to the TEMP array on exit.
ADVERBS	None.

OBJECT

Scan #	The number of the scan to be copied into the WORK array, where the associated off-scan will be copied into the TEMP array.

RELATED VERBS

CFETCH	Copies the most recently completed scan into the WORK array and the associated off-scan into the TEMP array.
CGET feed #	Gets the specifed feed of the most recently completed scan and places it in the WORK array.
GET scan #	Gets the specified scan and places it in the WORK array.
ON scan #	Gets the specified scan and places it in the WORK array.
OFF scan #	Gets the specified scan and places it in the TEMP array.

EXAMPLE

To get scan 29301 into the WORK array and 29302 into the TEMP array and display them consecutively, you specify,

```
FETCH 29301 PMW PAGE SHOW          displays  29301
PMT  PAGE  SHOW                    displays  29302
```

FFT

This routine replaces the currently referenced array with its fast fourier transform.

PTWH	Determines which array is transformed. It is not changed.

ADVERBS

BDROP(rcvr) EDROP(rcvr)	The number of channels at each end of the receiver which are ignored by the routine.

REMARKS

Any or all of the receivers may be processed, depending on the receiver processing option.

FINISH

This verb is used to end the definition of a procedure. PROCEDURE puts the program into the COMPILE mode; FINISH puts the program back into EXECUTE mode. FINISH signals that the program has returned to EXECUTE mode by prompting with a caret.

PTWH Is not used or changed.

ADVERBS None.

RELATED VERBS

PROCEDURE Begins the definition of a procedure.

FIX

FIX causes the verb SHOW to use the y-axis scaling that was last used before FIX was specified. Once FIX has been used, the y-axis scaling will remain fixed until FREEY or AUTO is specified.

PTWH — Is neither used nor changed.

ADVERBS

YMIN — Is set to -9.E10 by FIX. SHOW does not update its y-axis scaling if YMIN is less than -9.E09.

RELATED VERBS

AUTO — Alias for FREEY

FREEY — Sets YMIN to -99999.9. SHOW will automatically calculate the y-axis scaling if YMIN is between -9.E9 and -9999.

HOLDY — Alias for FIX.

RANGE — Sets YMIN and YINCR to span a range of y-coordinates specified as the arguments of RANGE.

EXAMPLE

See HOLDY.

FLAG

FLAG is a routine which draws a vertical line on the current display at the specified velocity or frequency. FLAG then labels the line.

<u>PTWH</u> Expects the current array to be the same as the last SHOW.

<u>ADVERBS</u> None.

<u>OBJECT</u> The velocity in km/sec for frequency in MHz for kHz to be flagged.

RELATED VERBS

GRID Draws a rectangular grid on the CRT at each horizontal and vertical tic mark and half tic mark point.

REMARKS

FLAG assumes the valus if its object has the units of the lower horizontal axis. The object may preceed or follow FLAG. Multiple uses if FLAG on the same line should be separated by semicolons.

FLAG only marks the current display. The vertical line will not be present on the next PAGE SHOW. "Permanent" flags can be placed on the screen by setting the adverbs VMARK(n) or FMARK(n).

EXAMPLE

You wish to see if a feature in the display is between -40 km/sec and -20 km/sec and the current values of the tic marks make guessing uncertain. Specifying

```
-40 FLAG; -20 FLAG
```

will plot vertical lines on the screen at the two velocities.

FOR

FOR begins a logical construction of the form

```
FOR (VARIABLE) = (ALPHA) TO (BETA) BY (GAMMA)
    (statements to be repeated for each value of (VARIABLE))
END
```

FOR can be used only in procedures. The FOR logical construction is a looping device similar to the iterative DO found in such languages as Fortran and PL/I.

(VARIABLE) can be any unused variable; it functions as an index for the loop. (ALPHA), (BETA), and (GAMMA) can be constants, variables, or arithmetic expressions. END is required to complete the construction. FOR loops may be nested and arranged on one or more lines; if arranged on one line, a semicolon must separate the statements to be iterated from the logical statements. BY (GAMMA) need not be included if (GAMMA) is to equal 1.

<u>PTWH</u> Is not used or changed.

<u>ADVERBS</u> None.

<u>RELATED VERBS</u>

END Completes a logical construction.

WHILE Begins a logical construction of the form:

```
WHILE (test condition)
    (statements to be iterated)
END
```

IF Begins a logical construction of the form:

```
IF (test condition)
    THEN (do this)
    ELSE (do this)
END
```

See PROCEDURE

-continued-

EXAMPLES

(1) FOR construction on several lines in a procedure

```
PROCEDURE EG
READ N
FOR I=2 TO N
     PRINT SQRT(N)
END
FINISH
```

(2) FOR construction on one line in a procedure

```
PROCEDURE EGALT
READ N
FOR I=2 TO N; PRINT SQRT(N); END
FINISH
```

(3) FOR constructions nested in a procedure

```
PROCEDURE EG2
READ N, M
SUM=0
FOR I=1 TO N
    FOR J=1 TO M
        SUM=SUM+J
        END
    END
FINISH
```

FREEY

FREEY returns the control of the y-axis scaling to the verb SHOW. Automatic scaling is the default, so FREEY is used to restore automatic y-axis scaling after the user has overridden it by means of the verb HOLDY, or by setting the values of the adverbs YMIN and YINCR with the verb RANGE or with assignment statements.

PTWH — Is not used or changed.

ADVERBS

YMIN — The minimum temperature to appear on the graph. FREEY sets YMIN to -90000. SHOW sets its own y-axis scaling if YMIN is less than -9999 but greater than -9.E9.

RELATED VERBS

AUTO — Alias for FREEY.

FIX — Alias for HOLDY.

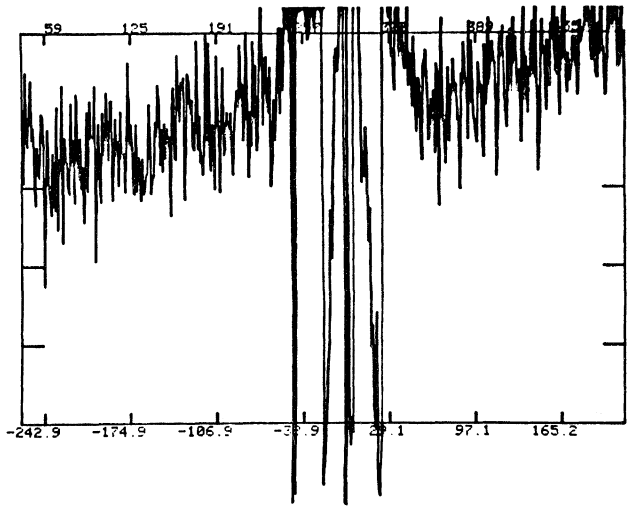
HOLDY — Causes SHOW to retain the last determined y-axis scaling.

RANGE — Defines the values of YMIN and YINCR according to the specified max and min values.

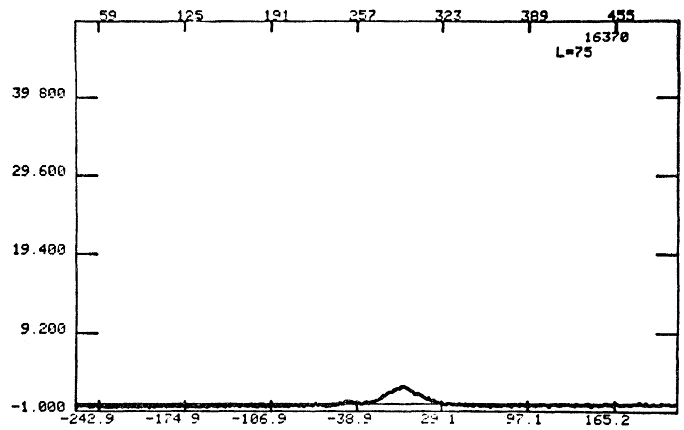
-continued-

EXAMPLE

Your last graph came out like this



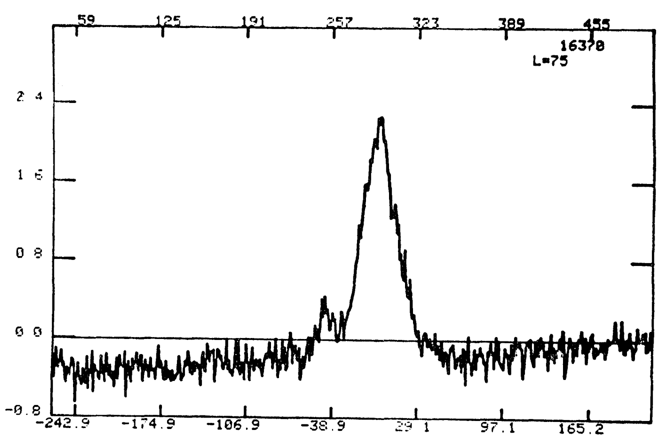
or this



To get a better graph, specify

FREEY PAGE SHOW

You will get



FV VF

A routine which sets the axis labeling mode so that SHOW labels the upper x-axis in velocity and the lower x-axis in frequency.

PTWH — Is not used nor changed.

ADVERBS — None.

RELATED VERBS

VC or CV — Sety the axis labeling mode for channel numers on the x-axis and velocity on the lower x-axis.

REMARKS

The frequency axis is given in terms of delta frequency (MHz or kHz, depending on bandwidth) from the center channel.

The default axis labeling for SHOW is channel number for the upper axis and velocity for the lower axis. Once FV or VF is specified, that mode remains in effect until changed by VC or CV, or the program RESTART operator is used.

GALTOEQ

GALTOEQ will convert galactic coordinates into equatorial coordinates (epoch 1950.0) and print the results on the TEKTRONIX screen.

PTWH Is neither used nor changed.

ADVERBS None.

OPERANDS

longitude Galactic longitude in decimal degrees.

latitude Galactic latitude in decimal degrees.

RELATED VERBS

EQTOGAL Converts equatorial coordinates (epoch 1950.0) to galactic coordinates and prints the results on the CRT screen.

REMARKS

The operands should follow GALTOEQ and be enclosed in parentheses, as in:

```
GALTOEQ(longitude,latitude)
```

EXAMPLE

To convert the position of SS433, l = 39°694, b = -2°245 from galactic to equatorial coordinates, specify

```
GALTOEQ (39.694, -2.245)
```

the equatorial coordinates are printed as

```
19  9  21.4    4 53 53.0
```

GAUSS

This routine fits up to six gaussian functions of the form

f(x) = HEIGHT * EXP (-2.772 * (x-CENTER)**2 / HWIDTH**2)

over a specified interval of a scan. The routine requires initial guesses of the center (explicit channel number) and width (in channels) at half the peak height of NGAUSS gaussians. The routine calculates initial guesses of the heights (in Kelvins) of the gaussians. Then the routine iterates the heights, center locations, and half-widths of the gaussians until a satisfactory fit is obtained. If the fit is successful, a caret is printed. If the fit is unsuccessful, the message FIT FAILED is printed.

PTWH — Determines for which array the fit is calculated. Is not changed.

ADVERBS

GREGION(r,N) — The intervals of channel numbers over which to fit for the specified receiver. This is ONLY used if BGAUSS = 0. Maximum number of 4 intervals for each of a maximum of 4 receivers.

NGAUSS — The number of gaussians to be fit. NGAUSS must be between 1 and 6.

BGAUSS
EGAUSS — The explicit channel numbers where the fit is to begin and end. These are used if GREGION = 0.

CENTER(N) — The channel number where gaussian N has its center. Iterated by the routine.

HEIGHT(N) — The height in degrees Kelvin of gaussian N. First guess is set, and then iterated by the routine.

HWIDTH(N) — The width in number of channels of gaussian N at half its peak height. Iterated by the routine.

NITER — The number of iterations which will be made to try to find a fit. The default value of NITER is 8.

FIXC — If FIXC is less than zero, GAUSS will iterate CENTER.

FIXHW — If FIXHW is less than zero, GAUSS will iterate HWIDTH.

-continued-

RELATED VERBS

GMODEL	Prints the gaussian parameters and the RMS error of the fit. Replaces the data in the currently referenced array with a model of the specified the gaussian(s).
GPARTS	Display each specified gaussian superimposed on the current plot.
RESIDUAL	Subtracts the sum of the specified gaussian(s) from the data in the currently referenced array.
PEAK	Finds a peak value and sets all the parameters for a single gaussian in preparation for GAUSS.

REMARKS

Any baseline offset should be removed before GAUSS is called. GAUSS is used to fit gaussians to data. The other gaussian verbs do not refine the parameters of the gaussian(s). GAUSS does not change the contents of any of the three scan arrays.

ERRORS

If the fit is unsuccessful after NITER iterations, the message FIT FAILED is printed.

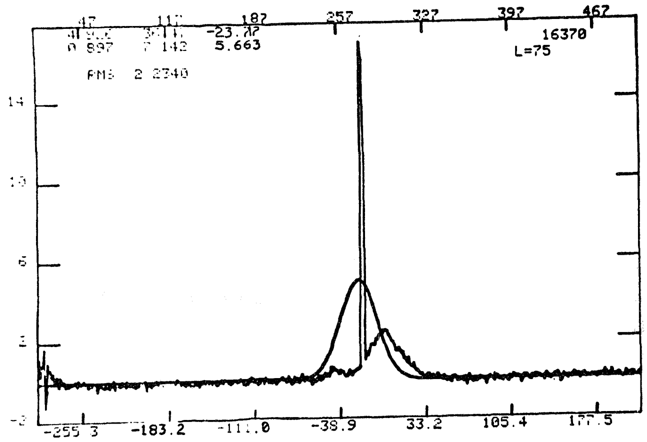
-continued-

EXAMPLE

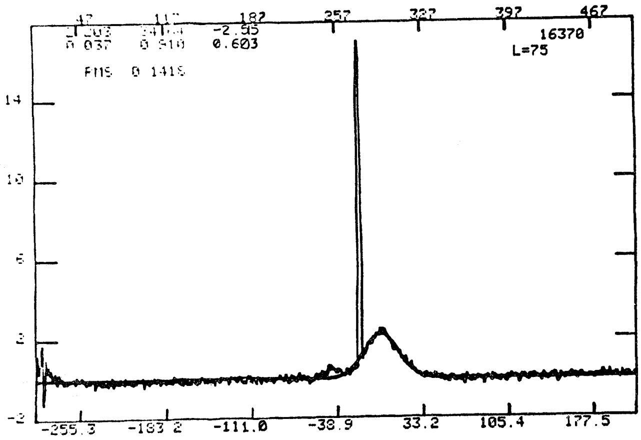
Your data has some interference.
You specify:

NGAUSS=1; BGAUSS=250; EGAUSS=300;
CENTER=294; HWIDTH=33; GREGION=0;
GAUSS GMODEL RESHOW

The results are the unsatisfactory display to the right.
To discount the bad channels, recall the original data and specify:



BGAUSS = 0;
GREGION(1,1)=250
GREGION(1,2)=270
GREGION(1,3)=280
GREGION(1,4)=300
GAUSS GMODEL RESHOW



This will result in the routine fitting channels 250-270 and 280-300, skipping the bad data, as in the display to the right.

GB

GB sets an internal flag which causes MAPSHOW to label the Y-axis of a velocity vs. position map as galactic latitude.

PTWH Is not used nor changed.

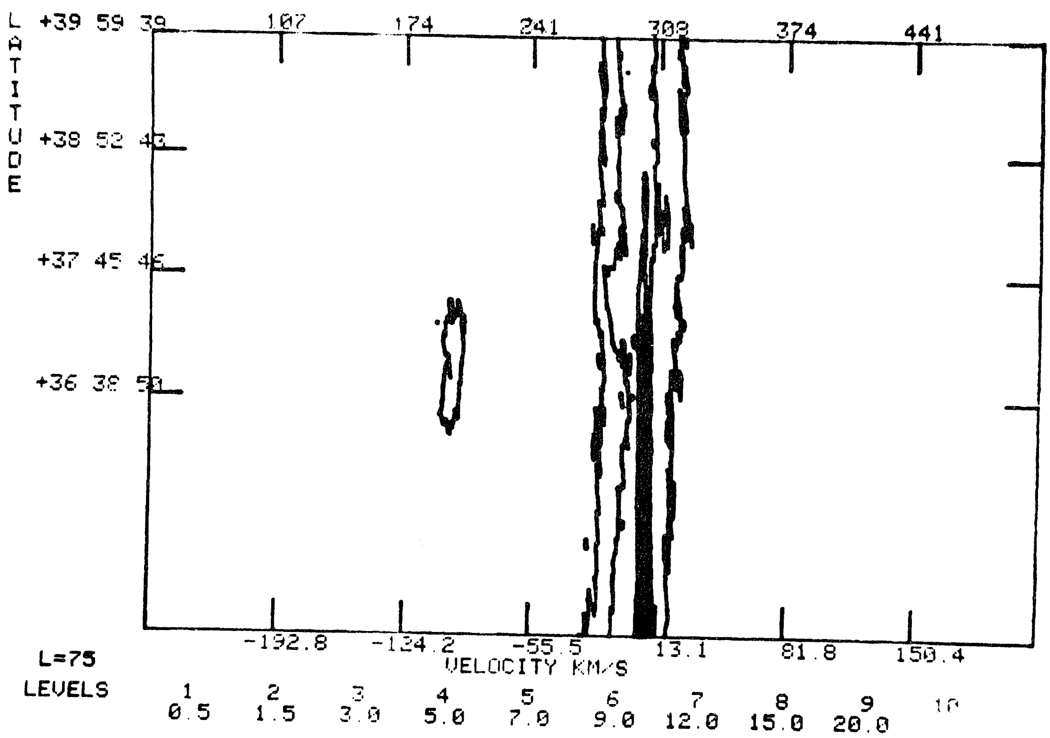
ADVERBS None.

RELATED VERBS

DEC	Sets the y-coordinate label as declination.
GL	Sets the y-coordinate label as galactic longitude.
RA	Sets the y-coordinate label as right ascension.

REMARKS

GB does not affect the scaling of the y-axis. MAPSHOW and LABEL assume that the scans have been correctly spaced in the coordinate of interest.



GET

GET is used to retrieve data from disk. It requires as its object a scan number. GET copies the indicated scan from the disk to the WORK array.

PTWH	Is not used, but is set to the WORK array.
ADVERBS	None.

OBJECT

Scan #	The number of the scan to be copied from disk.

RELATED VERBS

CGET rcvr#	Copies the most recently completed scan for the specified receiver.
ON SCAN#	Copies the requested scan into the WORK array.
OFF SCAN#	Copies the requested scan into the TEMP array.
TELL DISK	Lists the numbers of the scans in the disk index.

ERRORS

If scan # does not appear in the disk index, the message SCAN ? will be generated.

EXAMPLE

To bring a scan from the disk and display it on the screen, specify

```
GET 550 PAGE SHOW
```

where 550 is the number of the scan you want to see.

GL

GL sets an internal flag which causes MAPSHOW to label the y-axis of a velocity vs. position map as galactic longitude.

PTWH Is not used nor changed.

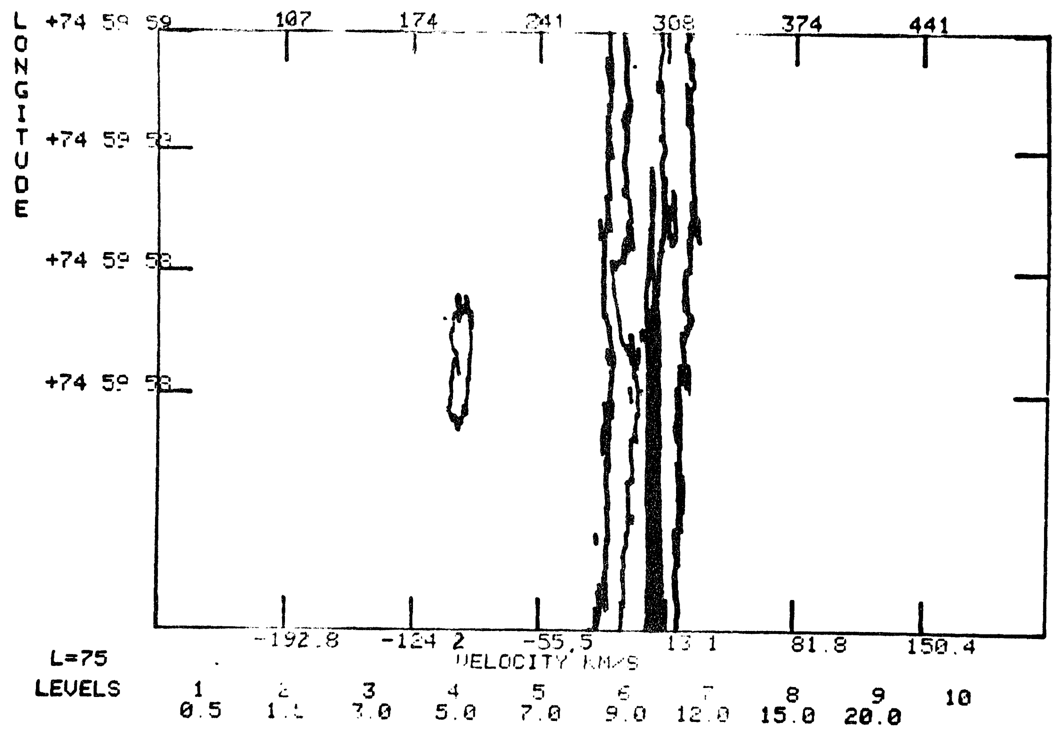
ADVERBS None.

RELATED VERBS

DEC	Sets the y-coordinate label as declination.
GB	Sets the y-coordinate label as galactic latitude.
RA	Sets the y-coordinate label as right-ascension.

REMARKS

GL does not affect the scaling of the y-axis. MAPSHOW and LABEL assume that the scans have been correctly spaced in the coordinate of interest.



GMEASURE

This routine calculates the area, mean flux, high and low velocity peak fluxes, high and low peik delta velocities and the minimum flux and delta velocity of a specified region in the currently referenced array.

PTWH	Determines which array is to be used to compute the values. It is not changed.

ADVERBS

BMOMENT(rcvr) EMOMENT(rcvr)	The first and last channels which define the window used to compute the area and mean flux.
FRACTION	The percentage of mean flux which is to be used to window the search for the high and low velocity peaks.

RELATED VERBS

MOMENT	Computes either the area or the centroid of a feature.
DELTAV	Computes velocities at specified fraction of the peak flux.

REMARKS

GMEASURE is primarily intended for galaxy profiles.

GMEASURE will only process one receiver at a time; if the option ALL is used, GMEASURE will process the A receiver.

ERRORS

If the window speicfied by FRACTION is outside of BMOMENT or EMOMENT, then the error message, 'B OR EMOMENT?' will appear.

If a new window less than or equal to the BMOMENT or EMOMENT cannot be found, the message 'FRACTION TOO SMALL' will appear.

GMODEL

This routine evaluates the parameters of NGAUSS gaussians, and constructs the sum of the gaussians. The sum replaces the data in the currently referenced array. GMODEL also prints the value of the RMS error of the fit.

PTWH	Determines which array is replaced with the model. It is not changed.

ADVERBS

NGAUSS	The number of gaussians.
CENTER(N)	The center points of the gaussians.
HEIGHT(N)	The heights of the gaussians.
HWIDTH(N)	The half widths of the gaussians.

RELATED VERBS

GAUSS	Refines the user's first guesses of CENTER(N) and HWIDTH(N) for N gaussians. Determines HEIGHT for each gaussian.
GPARTS	Display each specified gaussian superimposed on the current plot.
RESIDUAL	Subtracts the sum of the specified gaussian(s) from the data in the currently referenced array.
PEAK	Finds a peak value and sets all the parameters for a single gaussian in preparation for GAUSS.

-continued-

REMARKS

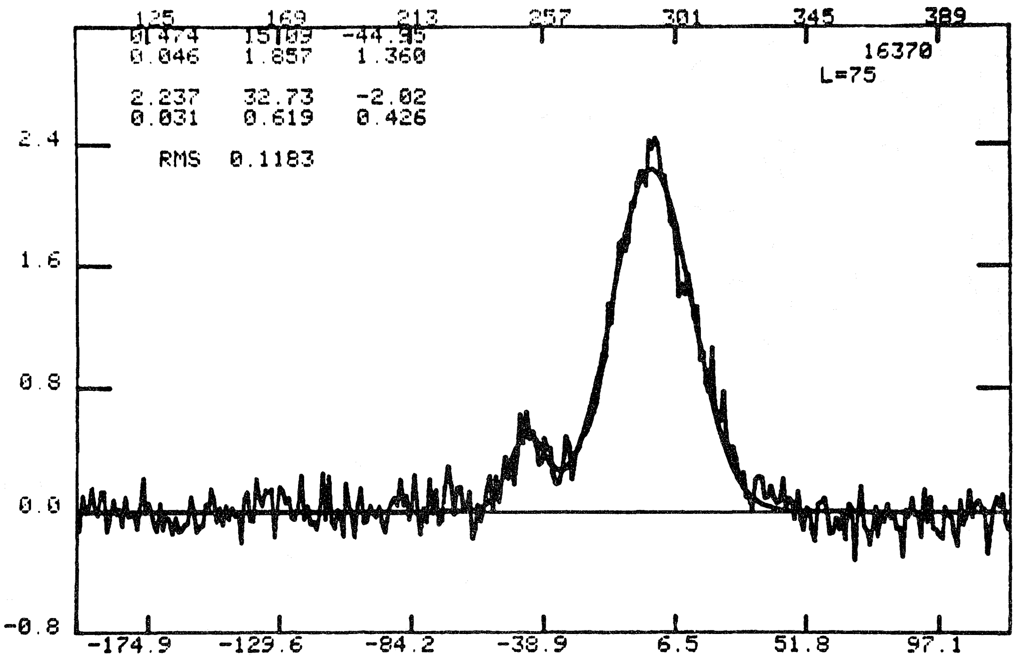
GMODEL is used after GAUSS to create a model of the fit, or by itself to create a particular gaussian or set of gaussians.

EXAMPLE

You fit a gaussian to your data. To display the model on top of the already displayed data, specify

TRT GMODEL RESHOW PMW

You will get



0.1183 is the RMS error of the fit in Kelvins

GPARTS

GPARTS evaluates the parameters of NGAUSS gaussians, and constructs and displays them separately. The data in the three main arrays are neither used nor affected.

<u>PTWH</u>	Is neither used nor changed.

<u>ADVERBS</u>

NGAUSS	The number of gaussians.
BGAUSS EGAUSS	The first and last points in the region within which the gaussians will be constructed and displayed.
CENTER(N)	The center points of the gaussians.
HEIGHT(N)	The heights of the gaussians.
HWIDTH(N)	The half widths of the gaussians.

<u>RELATED VERBS</u>

GAUSS	Refines the user's first guesses of CENTER(N) and HWIDTH(N) for N gaussians. Determines HEIGHT for each gaussian.
GMODEL	Evaluates the parameters of the gaussian(s), constructs a model of the gaussian (or sum of the gaussians), and replaces the data in the currently referenced array with the model. Also prints the parameters of the gaussians and the RMS error of the fit.
GPARTS	Display each specified gaussian superimposed on the current plot.
RESIDUAL	Subtracts the sum of the specified gaussian(s) from the data in the currently referenced array.
PEAK	Finds a peak value and sets all the parameters for a single gaussian in preparation for GAUSS.

-continued-

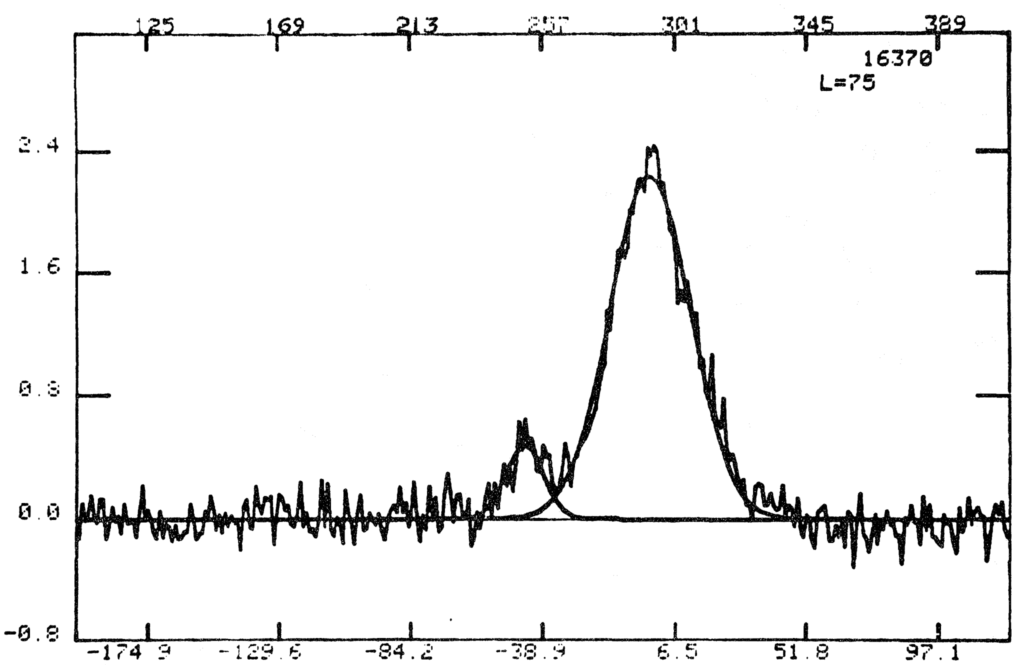
REMARKS

GPARTS is either used after GAUSS to create and display the fitted gaussians separately or by itself to create and display a particular gaussian or set of gaussians.

EXAMPLE

You have fitted a souble gaussian to scan 16370. To display the gaussians within the area over which it was fitted specify,

GPARTS



GRID

This routine draws a rectangular grid over the current display on the CRT screen.

PTWH — Is not used or changed by GRID.

ADVERBS — None.

RELATED VERBS

FLAG — Draws a vertical line on the current display at the specified velocity or frequency.

REMARKS

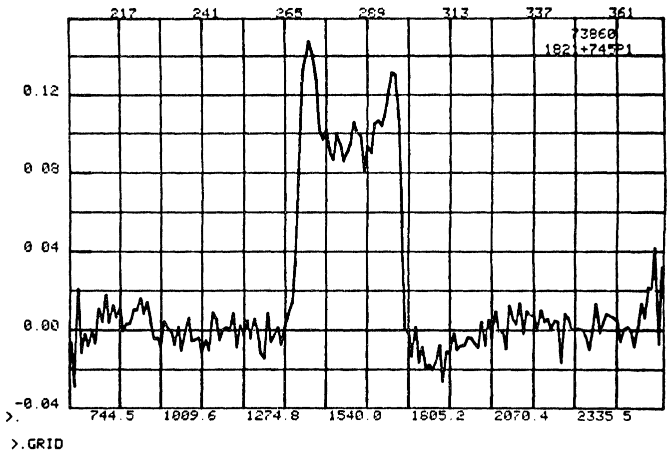
The grid is only for the current display and will not be present on the nexe PAGE SHOW.

EXAMPLE

In order to draw a rectangular grid on your current display specify

GRID

and you will get



HANNING

HANNING smoothes the scan in the currently referenced array by the algorithm

$$x_i = \frac{1}{4} x_{i-1} + \frac{1}{2} x_i + \frac{1}{4} x_{i+1}$$

PTWH	Determines which scan is smoothed. Is not changed.

ADVERBS

BDROP(CH) EDROP(CH)	The number of points at each end of each receiver channel which will be ignored by the routine.

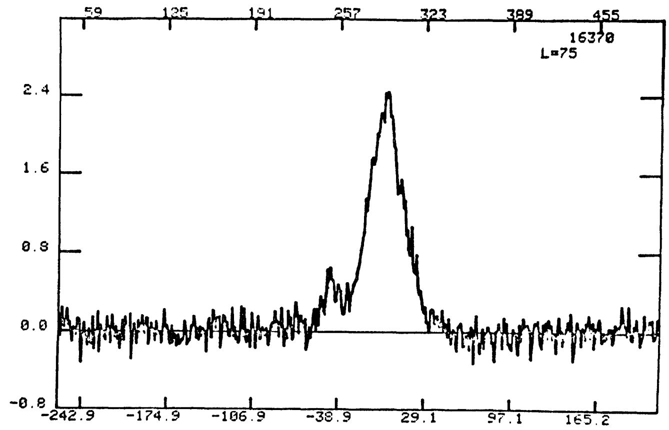
RELATED VERBS

BOXCAR	Smoothes a scan by averaging a specified number of points together.
SMOOTH	Smoothes a scan by convolution with a user specified weighting function.

-continued-

EXAMPLE

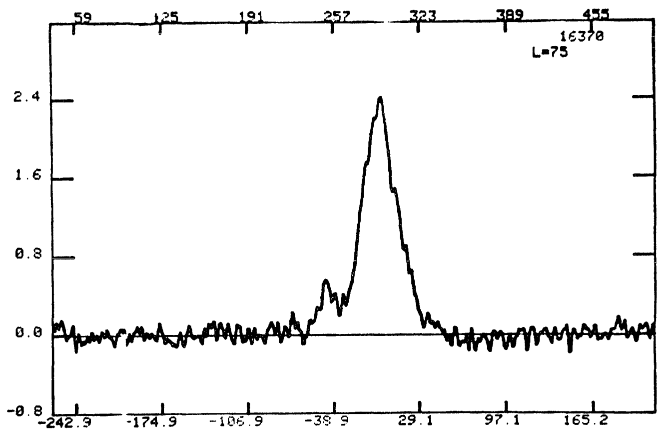
Your data looks like



To smooth it, specify

HANNING PAGE SHOW

You will get



HEADER

HEADER is an informational verb which prints the header information for the scan in the currently referenced array.

PTWH — Determines which header is printed. It is not changed.

ADVERBS — None.

RELATED VERBS

TABLE — Prints the data values for the currently referenced array.

EXAMPLE

To print the header information of a scan, specify,

```
GET scan#
PAGE HEADER
```

You will then see something like

```
                           NRAO SPECTRAL LINE ANALYSIS SYSTEM            ON
--------------   ----------------------------------------------   ----------
/ SCAN  16370/   / 198     LOCKMAN/KMJL188C    L=75             /   / MODE 2 /
--------------   ----------------------------------------------   ----------
DATE          EST          LST          IND POSITION         ROTAT FOCUS   OFF
 4 11 84    02 51 25.3  15 50 57.0  16 48 11.3 +48 36 07  90.5  454.      0

RECV CNTR FREQ   REST FREQ    BW     CNTR VEL   DEL VEL    RVSYS   INTG  ZEN

        MHZ        MHZ        MHZ       RA         RA        LR    TIME  DST

                                       KM/S       KM/S     KM/S    MIN   DE

 A   1420.7040  1420.4058   2.500    -38.89     1.031    -24.04    0.3  14.

 B   1420.7040  1420.4058   2.500    -38.89     1.031    -24.04    0.3  14.

          APPARENT                  1950               TSYS    TSYS    NOISE
      RA          DEC          RA          DEC         SIG     REF     TUBE
 A   16 48 05.4  +48 36 12   16 47 08.7  +48 40 02   22.13   21.94    3.71
 B   16 48 05.4  +48 36 12   16 47 08.7  +48 40 02   22.71   22.66    3.89
```

HELP

HELP is an informational verb with a number of uses:

1) when used by itself, it lists the various verbs and symbols recognized by the program;
2) when used with the object VERB, it lists the verbs and symbols recognized by the program (adverbs not included);
3) when used with the object PROCEDURE, or PROC, it lists the names of the procedures currently defined and in memory;
4) when used with the object ARRAY, it lists the arrays used by the program;

PTWH	Is neither used or changed.
ADVERBS	None.

OBJECTS

ARRAY	When used with HELP, generates a list of the arrays (most are adverbs) used by the program.
PROCEDURE or PROC	When used with HELP, generates a list of names of defined procedures.

RELATED VERBS

LIST procedurename	Lists the procedure whose name is specified.
PRINT adverbname	Lists the value of the adverb whose name is specified. If the adverb does not exist, the message SYMBOL ? will appear.
? adverbname or adverbname ?	Alias for PRINT.

-continued-

REMARKS

HELP generates lists of the symbols reserved by the program. These symbols cannot be used as procedure or variable names.

EXAMPLES

If you type HELP, the following output will appear on the screen:

```
>.HELP
,          (          )          =          +          -
*          /          **         >          <          ^
TO         BY                    &          ;          FOR
END        READ       PRINT      ?          RETURN     GETS
RUN        EXIT       RESTART    USING      PLACE      VCTR
CHAP       FLSH       SIN        COS        TAN        ATAN
SQRT       DUMP       XCUR       YCUR       LN         EXP
ABS        IFIX       PMT        PMW        PMH        TRT
TRW        TRH        ONLYA      ONLYB      ONLYC      ONLYD
ALL        GET        FETCH      ON         OFF        RECALL
SAVE       SELECT     TEMP       WIPE       TELL       BIAS
CRT        PRINTER    CGET       CFETCH     PAIR       KEEP
SUMMARY    CARD       ACCUM      SUM        AVE        BOXCAR
HANNING    SCALE      SLIDE      CLIP       DIFF       MOMENT
SMOOTH     INVERT     ALIGN      FREEY      AUTO       HOLDY
FIX        RANGE      DIVIDE     RDIFF      KACCUM     KAVE
SCLEAR     BASELINE   BMODEL     BSHAPE     DCBASE     RMS
KTEMP      CSMOOTH    SHOW       RESHOW     CROSSHAIR  FLAG
PAGE       CCUR       VCUR       TCUR       GRID       MAPSHOW
LABEL      RA         DEC        GL         GB         MAP
PIX        GAUSS      GMODEL     REGAUSS    PEAK       GPARTS
RESIDUAL   GMEASURE   DELTAV     FFT        IFFT       CGAUSS
CGMODEL    CREGAUSS   CGPARTS    CRESIDUAL  HEADER     PLOT
REPLOT     PROFILE    TABLE      STITLE     LOAD       SPREAD
FV         VF         VC         CV         LINE       HISTOGRAM
POINTS     MAPRD      LIMIT      RLINE      RHIST      RPOINTS
OVERLAP    RAPL       RAPR       A          ADD        DELETE
EMPTY      STACK      KDELETE    RIPPLE     RMODEL     RSHAPE
PROCEDURE  PROC       ARRAY      ELSE       THEN       FINISH
DEBUG      IF         WHILE      EDIT       ENDEDIT    HELP
MODIFY     STORE      RESTORE    LIST       CORE       SCRATCH
COMPRESS
>.
```

If you type HELP PROCEDURE before you have defined any of your own procedures, the following information will appear on the screen:

```
>.HELP PROCEDURE
R1         R2         R3         R4         GSCAN      F
S          LOOK       XX         SHOW1      SHOW2      SHOWS
ADVSET     RESET36    RESET      NRSET      CB
```

The output from HELP ARRAY is:

```
>.HELP ARRAY
BDROP      EDROP      FACT       YMIN       YINCR      CLIPMIN
CLIPMAX    SIZE       VRMS       BMOMENT    EMOMENT    CCHNL
VMARK      ASHIFT     NREGION    BBASE      EBASE      BREG
LEVS       CENTER     HEIGHT     HWIDTH     CMARK      SMWGT
WEIGHT     XMIN       XINCR      ASTACK     FMARK      TMARK
BADPT      GREGION    DVM        TWH        XDATA      WORK
>.
```

HISTOGRAM

A routine which sets the plotting mode for the SHOW display to histogram.

PTWH Is not changed or used.

ADVERBS None.

RELATED VERBS

LINE Causes SHOW to display data as a continuous line.

POINTS Sets the SHOW display mode to points.

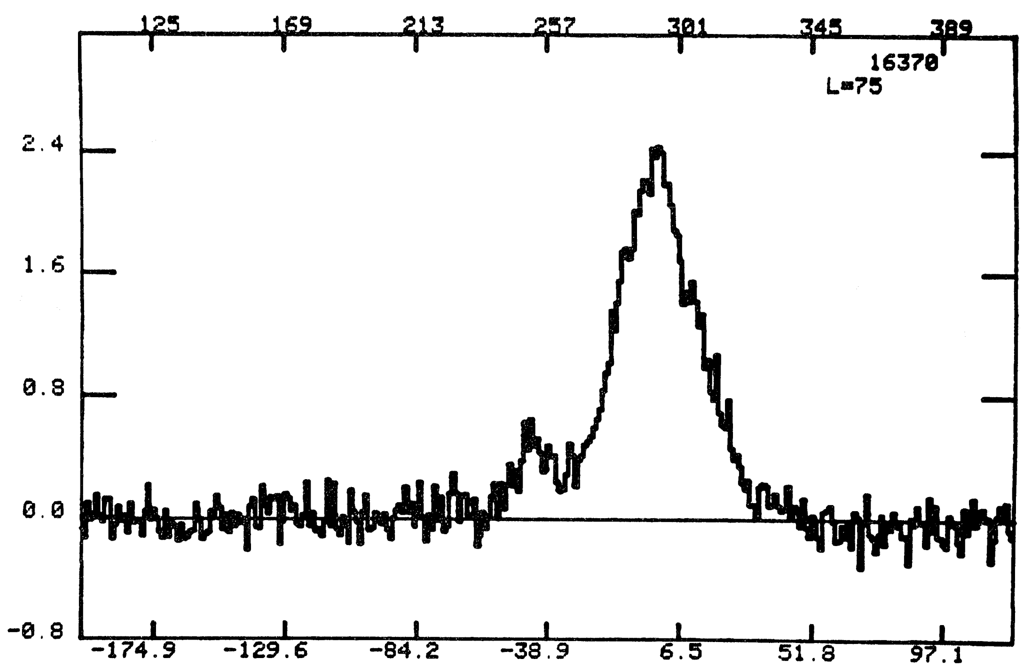
SHOW Plots the contents of the currently referenced array on the CRT screen.

EXAMPLE

Your last display was in LINE mode. In order to redisplay the data in HISTOGRAM mode, specify

```
HISTOGRAM PAGE SHOW
```

and you will get



HOLDY

HOLDY causes the verb SHOW to use the y-axis scaling that was last used before HOLDY was specified. Once HOLDY has been used, the y-axis scaling will remain fixed until FREEY or AUTO is specified or the adverb YMIN is set to a number less than or equal to -9999, which restores automatic scaling, or until a new scaling is specified with the verb RANGE or by setting YMIN and YINCR with assignment statements.

<u>PTWH</u> Is not used or changed.

<u>ADVERBS</u>

YMIN The minimum temperature to appear on the graph. HOLDY sets YMIN to -9.E10. SHOW sets its own y-axis scaling if YMIN is less than -9999 but greater than -9.E9.

<u>RELATED VERBS</u>

FREEY Sets YMIN to -99999.9. SHOW sets its own y-axis scaling
if

YMIN is less than -9999 but greater than -9.E9.

AUTO Alias for FREEY.

RANGE Defines the values of YMIN and YINCR according to the specified max and min values.

<u>EXAMPLE</u>

You want to compare four scans of different sources. To do this most easily, you want to get all the scans plotted to the same scale. You can accomplish this by

```
GET 520 PAGE SHOW
HOLDY
GET 536 PAGE SHOW
GET 552 PAGE SHOW
GET 510 PAGE SHOW
FREEY
```

IF

IF begins a logical construction of the form

```
IF (test condition)
     THEN (do this)
     ELSE (do this)
END
```

IF can be used only in procedures. The IF logical construction is a conditional device similar to the IF-THEN-ELSE construction found in PL/I.

(test condition) can be any expression which has a single true or false result. The (do this) parts may each consist of many statements. The first (do this) will be executed if the test condition is true; the second (do this) will be executed if the test condition is false. END is required to complete the construction. IF statements may be nested and arranged on one or more lines. If they appear on the same line, the IF, THEN, ELSE, and END clauses must be separated by semicolons. ELSE (do this) need not be included when no action is desired if (test condition) is false.

PTWH — Is not used or changed.

ADVERBS — None.

RELATED VERBS

END — Completes a logical construction.

FOR — Begins a logical construction of the form:

```
FOR (VARIABLE) = (ALPHA) TO (BETA) BY (GAMMA)
     (statements to be repeated)
END
```

WHILE — Begins a logical construction of the form:

```
WHILE (test condition)
     (statements to be iterated)
END
```

see PROCEDURE

-continued-

EXAMPLES

(1) IF statement on several lines in a procedure

```
PROCEDURE EG
READ X1,X2
IF X1  X2
   THEN PRINT X1
   ELSE PRINT X2
   END
FINISH
```

(2) IF statement on one line in a procedure

```
PROCEDURE EGALT
READ X1,X2
IF X1  X2; THEN PRINTX1; ELSE PRINT X2; END
FINISH
```

(3) IF statements nested in a procedure

```
PROCEDURE EG2
READ X1, X2, X3
IF X1  X2; THEN PRINT X1
        ELSE IF X2  X3; THEN PRINT X2
                         ELSE PRINT X3
             END
        END
FINISH
```

INVERT

INVERT flips the data end-for-end in the currently referenced array. The data on the disk is not flipped. It may be necessary to use INVERT if you have scans across the source in a "negative" direction (i.e., scan rate negative). If you are mapping, you take some scans in the opposite direction, so you will have to use INVERT to flip the negative direction scans before using MAPSHOW (not in Green Bank, however).

PTWH Determines the scan to be flipped end-for-end. It is not changed.

ADVERBS None.

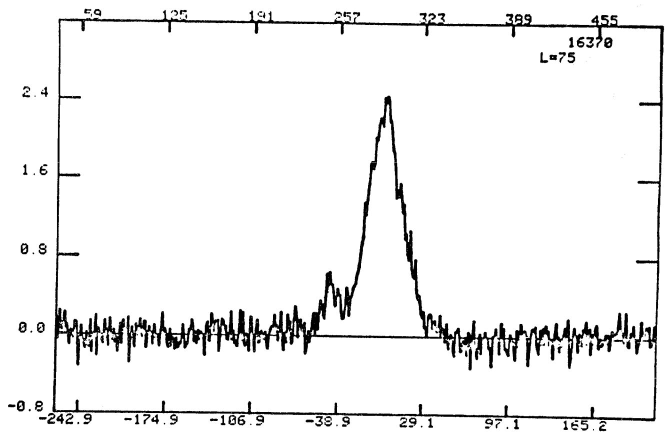
RELATED VERBS None.

REMARKS

If you decide INVERT should not have been invoked, a second INVERT will restore the status quo!

EXAMPLE

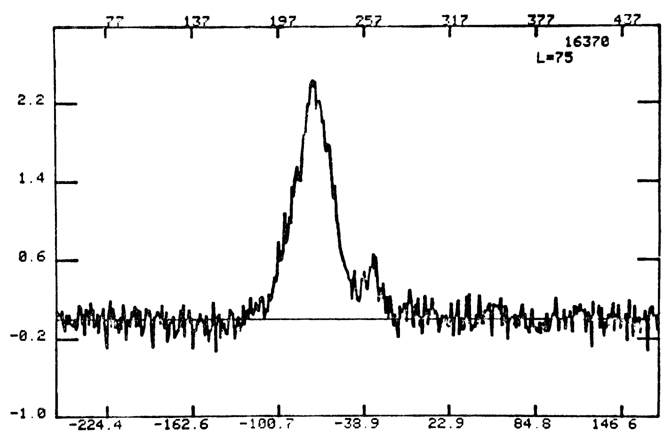
The scan in the currently referenced array looks like the display to the right. You have decided that the scan crossed the source the "wrong" way, and want to see how the scan would look if the scanning direction were reversed.



Specify:

INVERT PAGE SHOW

to produce the display to the right.



KEEP

KEEP copies the scan in the currently referenced array into a special file on disk. A separate program, external to the program, will then copy data in this file to tape. This is useful in taking home reduced data. The external program in Charlottesville is called KTAPE on both the MODCOMP and the IBM computers.

PTWH	Determines which array will be copied to disk. Is not changed.
ADVERBS	None.

RELATED VERBS

RECALL	Copies the scan in the bin NSAVE into the currently referenced array.
SAVE	Copies the currently referenced array into the disk bin NSAVE.

REMARKS

See Appendix A.4 for information on the data tape format.

In the current KEEP file, there is room for 143 scans. When the file is full, it must be dumped to tape before any more scans can be kept. If another user has scans in the KEEP file and has left, see Lorrie Morgan for help in removing those scans.

If the scans which were saved in the KEEP file are reloaded to the data disk file, they will retain, as the default option, the receiver processing option they had when the KEEP verb was used.

See the appropirate sections in Chapter 5 to write your keep file to tape.

-continued-

ERRORS

If the KEEP file is full when you use KEEP, the program will give the error message

NO ROOM!

EXAMPLE

You have stacked sixty scans and wish to carry home the result to do further processing at your home institution. Type

KEEP

At the end of your terminal session, EXIT the program and run the KTAPE program to produce your KEEP tape. The standard format of KEEP tapes is the same as the analysis disk format, in MODCOMP binary representation.

LABEL

LABEL is used to complete a contour map begun by MAP, MAPSHOW, or PIX. If the map was made by MAPSHOW, LABEL draws and labels the axis of the contour map and labels the contour levels. If the map was mady by MAP or PIX, LABEL writes additional information to the tape for the post-processing programs.

PTWH	Determines which array's contents are mapped. It is not changed.

ADVERBS

The following adverb is used for may type of map:

TITLE	A twelve character string which will be used to label the map if it is non-blank.

The following adverbs are used when the map was made by MAPSHOW:

BDROP(CH) EDROP(CH)	The number of points at each end of each receiver channel which are to be ignored by the routine.
LEVS(M)	The levels at which the contours were drawn by MAPSHOW. Up to 20 levels may be specified. The value of LEVS(n+1), if n less than 20, must be set to -9999.
MRATIO	is used to vary the relative X-Y scaling of the map made by MAPSHOW.

The following adverbs are used when the map was made by MAP:

DLEVS	Distance between the successive labeling of levels on the CALCOMP contour map.
IPEN	The pen used for drawing the map contours.
LEVS(M)	The levels at which the contours were drawn. Up to 20 levels may be specified. The value of LEVS(n+1), if n less than 20, must be set to -9999.
LPEN	The pen used to label the map.
MRATIO	Varies the relative X-Y scaling of the map.
NBOLD	The modulo number of contours to be drawn heavy.

-continued-

PUNIT — If non-blank, will cause the map to be labeled with the beginning and ending y-coordinates and the associated constant coordinate. The string assigned to PUNIT must be consistent with the one assigned to YUNITS.

REMARK — A 64-character string printed along with the map to provide additional documentation.

XMINCR — The units (km/s) per inch for the x-coordinate.

YMINCR — The number of units (as specified by the second character of YUNITS) per tic mark on the y-axis.

The following adverbs are used when the map was made by PIX:

ICOL
IROW — The number of additional points between each channel and each scan to be interpolated when making the picture.

MRATIO — Varies the relative X-Y scaling of the map.

XMINCR — The units (km/s) per tic for the x-coordinate.

YMINCR — The number of units (as specified by the second character of YUNITS) per tic mark for the y-coordinate.

RELATED VERBS

DEC — Identifies declination as the Y-coordinate for the map.

EXIT — Writes and end-of-file on the output tape when terminating the program. Must be used when MAP or PIX has been used.

GB — Causes LABEL to label the y-coordinate as galactic latitude when MAPSHOW is used.

GL — Causes LABEL to label the y-coordinate as galactic longitude when MAPSHOW is used.

RA — Causes LABEL to label the y-coordinate as right ascension when MAPSHOW is used.

-continued-

REMARKS

LABEL must always be used to complete a map. If you try to begin a new map with a different verb and have not used LABEL, the program will prompt you to use LABEL.

If the adverb TITLE is blank, LABEL will use the source name of the last scan contoured to label the map.

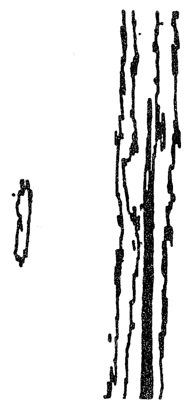
ERRORS

If there is no map to complete and LABEL is used, LABEL gives the message:

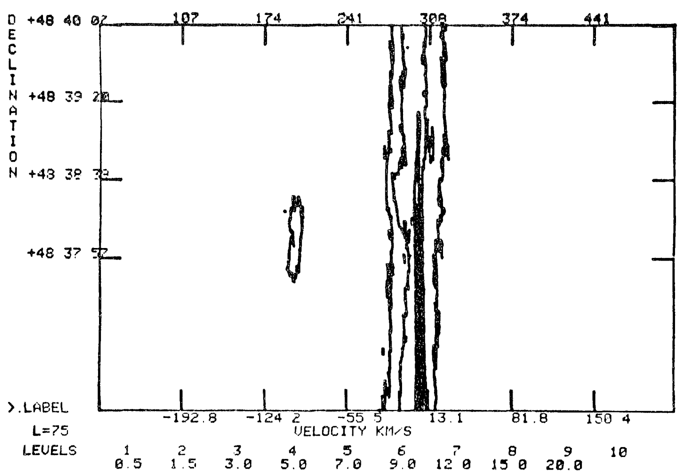
NO MAP!

EXAMPLE

Contour map before labeling (see MAPSHOW):



Same map after LABEL has been specified:



LIMIT

This routine will restrict data processing to only those channels within the minimum and maximum channels specified.

PTWH — Is not used or changed by LIMIT.

ADVERBS — LIMIT sets BDROP(rcvr) and EDROP(rcvr).

OBJECTS — The minimum and maximum point numbers desired.

RELATED VERBS

SPREAD — Limits the data processing and display to the minimum and maximum frequencies or velocity specified.

REMARKS

LIMIT restricts the processing of data by setting the appropriate values of BDROP and EDROP. To return to the original limits, reset BDROP and EDROP to zero, or use the RESET procedure.

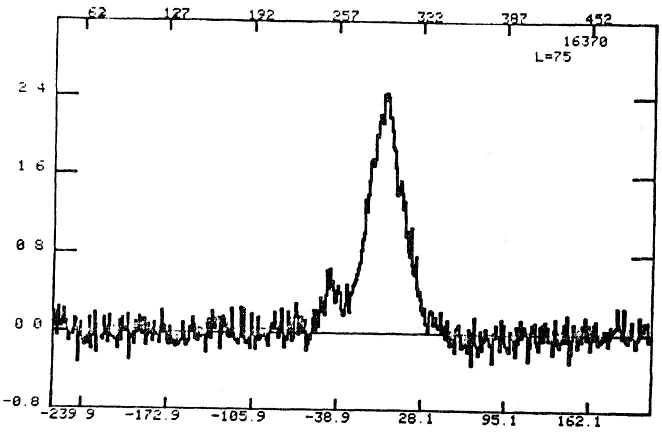
ERRORS

If LIMIT is specified without minimum and maximum values, the message "ARG LIST?" will be printed.

-continued-

EXAMPLE

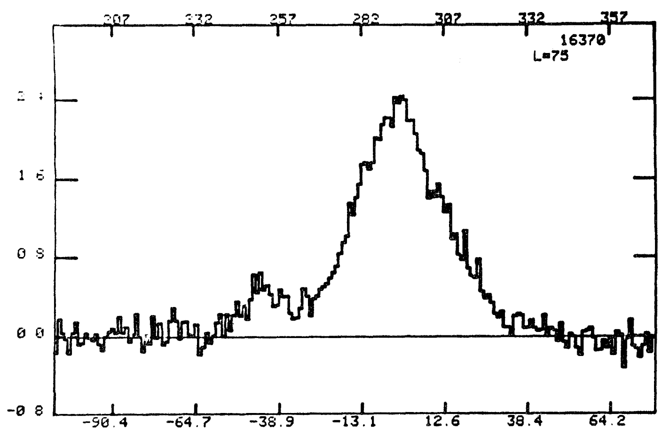
Your current display looks like the scan to the right. You would like to expand the horizontal scale by looking only at channels 190 through 370.



Use LIMIT to restrict the plotting to the desired channels:

```
190 370 LIMIT
PAGE SHOW
```

to produce the display to the right.



LINE

A routine which sets the plotting mode for the SHOW display to a continuous line.

PTWH Is not changed or used.

ADVERBS None.

RELATED VERBS

HISTOGRAM Sets the SHOW display mode to a histogram.

POINTS Sets the SHOW display mode to points.

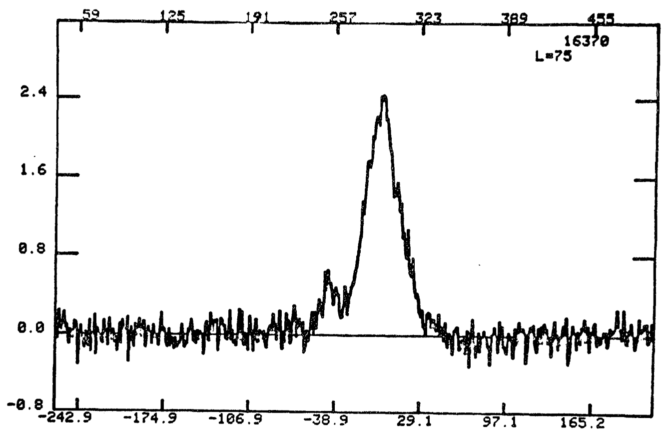
SHOW Plots the contents of the currently referenced array on the CRT screen.

EXAMPLE

Your last display was a HISTOGRAM. You want to change it to LINE. Specify

LINE PAGE SHOW

and you will get



LIST

LIST is an informational verb used to produce a listing of a defined procedure. LIST requires as its object the name of the procedure to be listed.

PTWH — Is not used or changed.

ADVERB — None.

OBJECTS

Procedurename	The name of the procedure to be listed.

RELATED VERBS

HELP PROCEDURE	Lists the names of the defined procedures.
PROCEDURE	Defines a procedure.

REMARKS

LIST numbers the lines of the procedures it lists. These line numbers should be used when editing the procedures.

EXAMPLE

You have defined the procedure DUMMY. To list it, specify

```
LIST DUMMY
```

You will get

```
1    PROCEDURE DUMMY
2    READ AA
3    PRINT AA
4    FINISH
```

LN

A built-in logarithmic function. Functions the same as the FORTRAN LOG.

PTWH Neither used nor changed.

ADVERBS None.

OBJECT

argument The argument of LN may be a constant, a variable, or an arithmetic expression, enclosed in parentheses.

RELATED VERB

EXP A built-in exponential function.

EXAMPLE

```
PRINT LN( 2.7183 )
1.0000
```

LOAD

On the Charlottesville MODCOMP,this routine will read scans from a 9-track tape and write them to the analysis disk file for later processing.

On the Charlottesville IBM, you may load data from a user tape or a KEEP tape by executing the program LOADTAPE <u>before</u> starting the program. After logging on to POPS MACHINE, have the computer operator assign a tape drive to POPS at virtual address 181. When he does, mount the tape on the drive. At the terminal, type

LOADTAPE

to start the loading program. You will be prompted for all of the adverbs listed below. The adverb TFMT is not needed.

<u>PTWH</u>	It is not used or changed.

<u>ADVERBS</u>

BRANGE ERANGE	The minimum and maximum right ascension in hours (1950) of scans to be loaded.
BSCAN ESCAN	The minimum and maximum scan numbers to be loaded.
TFMT	Tape format indicator. (MODCOMP only) 1024 = MODCOMP produced KEEP tape 360 = IBM produced tape.

<u>RELATED VERBS</u>

KEEP	Writes a scan to a disk keep file which can be copied to tape.
WIPE	Removes all of the scans belonging to a specified user from the disk index.

-continued-

REMARKS

The verb LOAD and the program LOADTAPE search the entire tape for scans in the range of scan numbers from BSCAN to ESCAN, and the range of right ascensions from BRANGE to ERANGE. BRANGE must be less than ERANGE, so to load sources in the range 20 hours to 04 hours, first LOAD 20 to 24 hours and then LOAD 0 to 4 hours.

The verb LOAD and the program LOADTAPE begin loading scans onto disk at a point immediately following the last scan previously written to disk. If the scan table or the disk file becomes full, LOAD and LOADTAPE will begin over-writing scans from the beginning of the scan table until the last scan on the tape has been loaded.

EXAMPLE

You have made a MODCOMP keep tape of partially reduced scans and would like to process them further on the Charlottesville MODCOMP. After starting the program and threading the tape on the tape drive, type:

```
TFMT=1024;LOAD
```

and all of the scans on the tape will be loaded.

An alternative format allows the Charlottesville MODCOMP to read tapes produced by the IBM program:

```
TFMT=360;LOAD
```

You have a user tape or a KEEP tape from any of the telescopes and you wish to load the data onto the Charlottesville IBM. After logging on, type

```
LOADTAPE
```

and the program will prompt you for the various windowing parameters. When the program has finished, you may start the program by typing

```
LINE
```

MAP

THIS ROUTINE IS ONLY AVAILABLE ON THE MODCOMP

MAP is a routine which writes data points on a tape to be processed on the Charlottesville IBM to generate position vs. velocity contour plots. The user must order the scans from "top to bottom" and use MAP for each scan. Only one receiver may be contoured at one time. The verb LABEL must be used after the last scan to complete the map.

PTWH — Determines which array is to be contoured. Is not changed.

ADVERBS

The following adverbs are required by the verb MAP. They must be set before a map is started and cannot be changed before LABEL is used to complete the map. Otherwise, the results will be unpredictable.

BDROP(rcvr)
EDROP(rcvr) — The number of channels at the beginning and end of the receiver which are to be ignored.

YUNITS — Indicated the coordinate and the units of the Y-axis. The first character specifies the corrdinate (R, D, L, B) and the second specifies the units (H, D, M, S). The default is "DD".

The following adverbs are required by the verb LABEL to complete the map. They may be set either before beginning a map or any time before LABEL is used.

DLEVS — Distance between successive labeling levels on the CALCOMP contour map. The default is 0.5.

IPEN — The pen used for drawing the map contours. The default is 1.

LEVS(20) — Values (and number) of the contour levels to be plotted. The value of LEVS(n+1), if n is less than 20, mus to set to -9999.

LPEN — The pen used to label the map. The default is 1.

MRATIO — Varies the relative X-Y scaling of the map. The default is 1.

NBOLD — The modulo number of contours to be drawn heavy. default is 2.

-continued-

PUNIT	If non-blank, will cause the map to be labeled with beginning and ending Y-map coordinates and its associated constant coordinate. The string assigned to PUNIT must be consistent with YUNITS. The default is the blank string (' '). Permissable values are 'RD','DR','LB','BL'.
REMARK	A string of 1 to 64 characters used for additional documentation of the map. The default is a blank string.
TITLE	A user-specified alternative name to lable the map. TITLE may contain from 1 to 12 characters. If TITLE contains all blanks, then the source name of the last scan used in the map will be used to label the map. The default for TITLE is a blank string.
XMINCR	The units (km/s) per inch for the x-coordinate. The default is 20.
YMINCR	The units (as specified by the second character of YUNITS) per tick for the y-coordinate.

RELATED VERBS

EXIT	Writes an end-of-file on the plot tape.
LABEL	Completes the map.
MAPSHOW	Produces a CRT contour map.
PIX	Used instead of MAP to generate a DICOMED picture.
PLOT	Writes a spectrum on the plot tape for the CALCOMP plotter.

REMARKS

The receiver processing options ALL or ONLYA will cause the first receiver to be mapped. To map the other recievers, use ONLYB, ONLYC, ONLYD, or the verb SELECT.

The verb LABEL must be used to complete each map. The routine assumes that there is a tape with a write-ring on the correct drive. (See the separate section on off-line plotting.) At the end of a terminal session, the verb EXIT is needed to write an end-of-file on the tape.

-continued-

The size of the map affects the post-processing on the Charlottesville IBM. If the number of points to be contoured (# points per scan times # scans) is greater than 5000 for any map, then the post-processing step must be run as a class L job.

MAP is most conveniently used within a procedure. If the scans have been preprocessed and reloaded from a KEEP tape, or if a baseline can be fitted to all the scans with common parameters, then the STACK is an easy way to order the scan numbers to be contoured.

A tape must be mounted on the tape drive to use MAP.

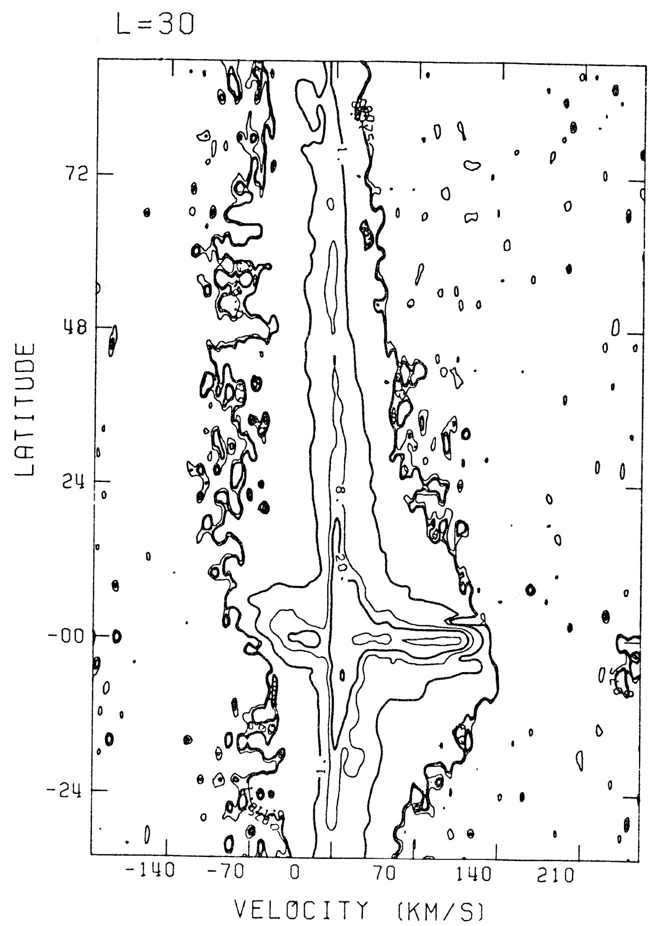
EXAMPLE

The following procedure will make a longitude vs. velocity map with simple baseline removal of a series of discontinuous scan numbers. Will map at 30 minute intervals in galactic longitude, and velocity at 10 km/s.

```
*DEFINE THE PROCEDURE
PROC CALCOMP
:    FOR I = 1 to ACOUNT
:         GET ASTACK(I); BASELINE; MAP; END;
:    LABEL
:    RETURN
:    FINISH

*SET MAP ADVERBS
PUNIT='LB'; YNITS='LM'; YMINCR+30; XMINCR=10;
LEVS = 0.1, 0.2, 0.4, 0.6, 0.8, 1.0, 5, 10, -9999
REMARK = 'TEST FOR A DICOMED PICTURE'
DLEVS=999; NBOLD=5

*SET BASELINE PARAMETERS
BDROP=20; EDROP=20; BBASE=50; EBASE=50; NFIT=1;

*PUT SCAN NUMBERS ON THE STACK
EMPTY
A 527 A 559 A 519 A 593 A 521 A 551 A 40 A 563 A 41 A 549
A 420 A 591 A 43 A 547 A 44 A 589 A 45 A 631 A 46 A 587 A 47

*MAKE THE MAP
CALCOMP
```

-continued-

An example of a Calcomp map is shown below:



MAPSHOW

This routine draws a position vs. velocity contour map of a set of scans (more than one). The first use of MAPSHOW after a LABEL (or program restart) copies the scan in the currently referenced array into the HOLD array. Subsequent MAPSHOW's compare the scan in the currently referenced array to the scan in the HOLD array, draw the contours, and move the scan in the currently referenced array into the HOLD array. After the last MAPSHOW, LABEL must be used to complete the map.

PTWH Determines which scan is contoured.
Is not changed.

ADVERBS

The following adverbs are required by the verb MAPSHOW. They must be set prior the starting a map and cannot be changed before the verb LABEL is used to complete the map; otherwise, the results will be unpredictable.

BDROP(CH)
EDROP(CH) The number of points at each end of each receiver channel that are ignored by the routine.

LEVS(N) The levels at which a contour line is to be drawn. Up to 20 levels may be specified. If less than twenty levels are specified, the first level beyond the desired levels should be set equal to -9999.

MRATIO Is used to vary the relative X-Y scaling of the map made by MAPSHOW.

The following adverb is optionally used by the verb LABEL to complete the map. It may be set any time before LABEL is used.

TITLE(12) A 1 - 12 character string to be used to label the map. If TITLE is a blank string, LABEL will use the source name of the last scan contoured to label the map.

-continued-

RELATED VERBS

DEC	Selects declination as the Y-coordinate of the map.
GB	Causes the y-axis of the map to be labeled in galactic latitude.
GL	Causes the y-axis of the map to be labeled in galactic longitude.
LABEL	Draws and labels the axes and labels the contours the map made by MAPSHOW. Used after all the scans have been mapped.
MAP	Writes the data to a magnetic tape which is used to make a position vs. velocity map on the CALCOMP plotter.
PIX	Writes the data to a magnetic tape which is used to make a position vs. velocity DICOMED picture.
RA	Selects right ascension as the Y-coordinate of the map.

REMARKS

MAPSHOW contours only one scan at a time, so it is often used in a procedure. Note that you cannot map by recalling data into the HOLD array!

-continued-

EXAMPLE

We need to produce a contour map of scans 16370 - 16700. We wish to draw contour levels at 0.5, 1.5, 3.0, 5.0, 7.0, 9.0, 12.0, 15.0, and 20.0 Kelvin, specify

```
LEVS = .5, 1.5, 3., 5., 7., 9., 12., 15., 20., -9999
```

to set the levels, and

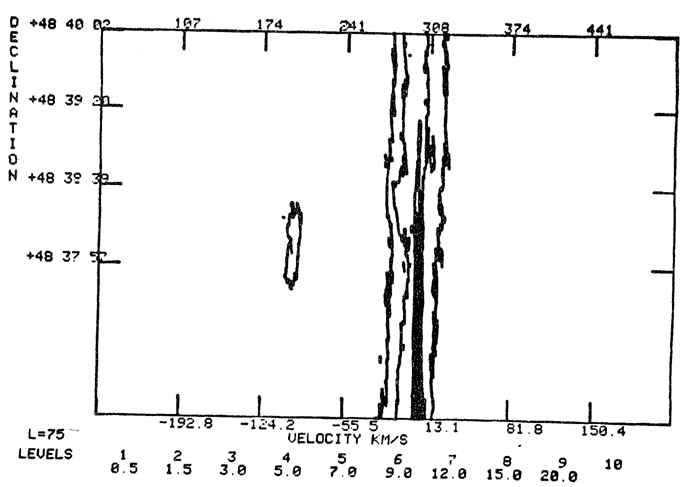
```
MRATIO = 10
```

to expand the graph in the vertical and horizontal directions.

Then enter

```
>PROCEDURE MAPS (IST,ISP)
:PAGE SHOW
:FOR I = IST TO ISP BY 10
:GET I; MAPSHOW
:LABEL
:END
:FINISH
>PAGE MAP (17)
```

and you will have something like this:



MAPRD

THIS VERB EXISTS ON THE IBM ONLY.

MAPRD is a routine which writes "CNTL" cards for the Calcomp General Purpose Contouring Program (GPCP). The output parameters are right ascension and declination, and the veriable, SIZE. The order of the scans is insignificant, since "CNTL" cards are for randomly spaced data. The output is to a disk file, which can be written to tape using the external program CNTLCARD.

<u>PTWH</u> Determines which array is processed. It is not changed.

<u>ADVERBS</u>

SIZE(n) Current value. SIZE is initialized by the MOMENT verb.

<u>RELATED VERBS</u> None.

<u>REMARKS</u>

After EXITing LINE, use the program CNTLCARD to wuite the output of MAPRD to tape. See the section in Chapter 5 on writing GPCP control cards to tape.

Using MAPRD/CNTLCARD is not a necessity if using the GPCP. The GPCP is actually smart enough to realize what is to be contoured by the data it encounters. See the chapter on off-line data display for instructions on using the procedure MCPLOT to produce the contour map.

MODIFY

MODIFY is an editing routine which allows you to change a line of a procedure by typing underneath the characters you want to change on the line, the new characters you want. The cursor is positioned with the space bar and the backspace key. MODIFY prompts you for input with a question mark rather than the colon of the EDIT command.

OBJECTS

procedurename	The name of the procedure to be modified.
linenumber	The number of the line to be change.

RELATED COMMANDS

EDIT procedurename linenumber	Used to change lines, insert lines, and delete lines in a procedure.
LIST procedurename	Used to list a procedure.

REMARKS

Modify prints the line you wish to change preceded by a prompt character. The first character position to the right of the prompt character is the first character of the line. This could be a blank character. Next MODIFY does a carriage return, line feed, and prints the prompt character again. The cursor is left under the first character of the line above.

The spacebar and the backspace key do not affect the line being modified. They are only used to position the cursor underneath a character to be changed. When you backspace to a character, any changes you may have made to the right of the cursor are deleted. You will have to re-type them.

MODIFY uses two special characters. The @ symbol is used to replace the character above it with a space. The $ used to delete the character above it.

You can "escape" from MODIFY by typing a blank line. A blank line is a line with at least one space or blank and no other type of character on the line, including the special characters. If you type a null line (nothing on it), you will be prompted again.

-continued-

You cannot use MODIFY to insert a new line into a procedure. If you use a line number with a fractional part, the fraction will be truncated. You should not use MODIFY to delete a line.

Most errors detected by MODIFY will cause you to be re-prompted to change the line again.

ERRORS

A fixed amount of space is available for defining procedures. This space is also used up by editing procedures. When the space has been completely used up, the program will give the message

BLEW CORE!

At this point you have three options:

(1) You can do without the procedure you were defining or editing when the message appeared. All other procedures will still be usable, but not able to be edited.
(2) You can wipe out all of your procedures (and adverb changes) by using RESTART. Then type in your correct procedures, hoping that if you don't do any editing, they will all fit.
(3) You can store your current set of procedures on the disk, using the STORE command. Then RESTART. Then define any additional procedures you need. Since RESTART kills your adverb values and array contents, if you have two "pages" of procedures (one on disk, and one in memory) which use each other's results, you will have to store intermediate results in scan save bins. Intermediate variable results (for example, SIZE) will be lost.

-continued-

EXAMPLE

To change a line of a procedure, first list it

```
                        >LIST PRCDR
                        1    PROCEDURE PRCDR(S)
                        2    GET(S) BASELINE NBOX=3; BOXCAR
                        3    PAGE SHOW ACCUM
                        4    RETURN
                        5    FINISH
```

to change NBOX=3 to NBOX=5, type

```
                        >MODIFY PRCDR 2
MODIFY types            ?GET(S) BASELINE NBOX=3; BOXCAR
you type (after "?")    ?                      5
MODIFY then types        GET(S) BASELINE NBOX=5; BOXCAR
```

and changes the procedure in the procedure definition space.

To change BOXCAR to HANNING

```
                        >MODIFY PRCDR 2
MODIFY types            ?GET(S) BASELINE NBOX=5; BOXCAR
you type (after "?")    ?                HANNING$$$$$$$
MODIFY then types        GET(S) BASELINE HANNING
```

If you try to delete a line with MODIFY:

```
                        >MODIFY PRCDR 4
MODIFY types            ?RETURN
you type (after "?")    ?$$$$$$
MODIFY then types     (blank line)
```

```
you end up with         >LIST PRCDR
                        1    PRCDR(S)
                        2    GET(S) BASELINE HANNING
                        3    PAGE SHOW ACCUM
                        4
                        5    FINISH
```

MOMENT

A routine which calculates the moment specified by the adverb NMOMENT over a specified range of channels and stores the result in the variable SIZE. Currently the verb will only compute the answer for NMOMENT = 0.

PTWH	Determines for which scan the moment is calculated. Is not changed.

ADVERBS

NMOMENT	The order of the moment to be calculated, as follows: 0 gives AREA (Kelvins)
BMOMENT(CH) EMOMENT(CH)	The first and last channels of the region overwhich the moment is to be calculated.

REMARKS

The moment calculated by MOMENT is stored in the variable SIZE, which can be printed by the command

```
PRINT SIZE
```

One value is printed for each receiver channel.

EXAMPLE

You want to calculate the area under your data.

You specify:

```
NMOMENT = 0
BMOMENT = 238
EMOMENT = 329
MOMENT
PRINT SIZE(1)
```

OFF

OFF copies the indicated scan from the disk to the TEMP array. OFF requires a scan number as its object.

PTWH — Is not used, but is set to the TEMP array.

ADVERBS — None.

OBJECTS

scan # — The number of the scan to be copied into the TEMP array.

RELATED VERBS

ON scan # — Copies the indicated scan into the WORK array.

REMARKS

OFF is commonly used in conjunction with ON to pair up off and on scans so that they can be differenced to remove baselines.

ERRORS

If the scan # indicated is not in the disk index, the message SCAN ? will be generated.

EXAMPLE

To difference 4588 and offscan 4575 and display the result, specify

```
OFF 4575; ON 4588;  DIFF PAGE SHOW
```

ON

ON copies the indicated scan from the disk to the WORK array. ON requires a scan number as its object.

PTWH Is not used, but is set to the WORK array.

ADVERBS None.

OBJECTS

scan# The number of the scan to be copied into the WORK array.

RELATED VERBS

OFF scan# Copies the indicated scan into the TEMP array.

REMARKS

ON is commonly used in conjunction with OFF to pair up on and off scans so that they can be differenced to remove a baseline.

ERRORS

If the scan # indicated is not in the disk index, the message SCAN ? will be generated.

EXAMPLE

To difference onscan 4588 and offscan 4575 and display the result, specify

```
OFF 4575; ON 4588; DIFF PAGE SHOW
```

ONLYA ONLYB ONLYC ONLYD

These verbs change the receiver processing option to the indicated receiver for the currently referenced array. Verbs that can process more than one receiver will only process the A, B, C, or D receiver.

PTWH — Determines which array's processing option is changed. It is not changed.

ADVERBS — None.

RELATED VERBS

ALL — Sets the receiver processing option to all the the receivers in the currently referenced array.

PAIR(m,n) or m n PAIR — Sets the receiver processing option to receivers m and n.

SELECT — Used with the adverbs SET1, SET2, and SET3 to select one or more receivers for processing.

REMARKS

You should not use ONLY_ to select a receiver which does not exist. It is important to realize that when ONLYA, ONLYB, ... , or SELECT is used, the processing option is only changed for the scan in the currently referenced array. It does not affect other scans (or the same scan) in other arrays. When a scan is loaded from the disk using GET, FETCH, CGET, or CFETCH, the default option is ALL.

However, when the verb SAVE (or KEEP) is used to write a scan to disk, its current processing option is stored with it. Thus, using RECALL to retrieve the saved scan causes the processing option to be set to the option the scan was saved with, not the default of ALL.

-continued-

EXAMPLE

To remove a baseline only from the A receiver of the curretnly referenced array, type

    ONLYA BASELINE PAGE SHOW

To remove a baseline only from the D receiver, specify

    ONLYD BASELINE PAGE SHOW

If after removing the baseline from D, you saved the results using

    NSAVE=21; RECALL PAGE SHOW

would cause the D receiver to be displayed. The other receivers could then be displayed or processed by using ONLYA, etc.

OVERLAP

This routine combines the right hand half of the A receiver with the left hand half of the A reciever. The routine eill look at the frequencies in the header information and calculate the number of channels to shift the spectrum (OSHIFT) before averaging. It will make a copy of the spectrum in the currently referenced array, negate it, and then wrap the spectrum by the calculated number of channels, meaning that channels shifted off the end of the spectrum wrap around and enter at the other end, and then average. The number of channels to be dropped att each end of the spectrum is not changed. You must decide which half of the spectrum is the signal half and drop the other half by resetting BDROP and EDROP.

<u>PTWH</u> Determines which scan is to be overlapped. Is not changed.

<u>ADVERBS</u>

OSHIFT Set by OVERLAP. The number of channels necessary to align the left and right hand sides.

<u>RELATED VERBS</u> None.

<u>REMARKS</u>

OVERLAP works only on the referenced receiver. If the processing option is ALL, only the A receiver is processed.

OVERLAP is used on data taken with the observing technique known as Dual Dicke switching or overlapped frequency switching.

-continued-

EXAMPLE

You want to combine the signal and reference spectra, so you specify

    OVERLAP

Set EDROP before SHOWing to display the average combined half.
OSHIFT will be set to the number of channels needed to shift the feature.

PAGE

PAGE erases the CRT screen and returns a caret at the top of the now blank screen. PAGE is equivalent to striking the PAGE ERASE button except that the PAGE ERASE button does not respond with a caret.

PTWH Is not used or changed.

ADVERBS None.

REMARKS

If you hit the PAGE ERASE button instead of entering the command PAGE, you will not get a caret. To be certain that the program is ready to accept a command, you can strike RETURN to get a caret.

N.B. The verb PAGE is preferred over striking the PAGE ERASE button because PAGE also resets the relative origin of the CRT screen.

PAIR

PAIR selects any two recievers to be processed. It requires two objects, the numbers of the two receivers to be processed.

PTWH — Determines which scan's processing option is to be changed. PTWH is not changed.

ADVERBS — None.

OBJECTS

m — The number of the first receiver to be processed.

n — The number of the second reciever to be processed.

RELATED VERBS

ONLYA — Sets the receiver processing option to the first reciever.

ONLYB — Sets the receiver processing option to the second receiver.

ONLYC — Sets the receiver processing option to the third receiver.

ONLYD — Sets the receiver processing option to the fourth receiver.

SELECT — Changed the processing option of the scan in the currently referenced array.

REMARKS

PAIR is primarily intended to be used with the verb RDIFF and SLIDE when a scan has more than two receivers. The object m is normally less than n but may be greater than or equal to n.

PAIR (1,3) is equivalent to SET1=1; SET2=3; SET3=2; SELECT

PAIR (m,n) and m n PAIR are both acceptable syntax.

-continued-

ERRORS

If either m or n is less than 1 or greater than the number of receivers, then the error message "ARG LIST" is given.

If, after executing the verb SLIDE, your results look no different than what you started with, then make sure that you have the receiver processing option ALL in effect before using PAIR.

EXAMPLE

You have four receiver data and wish to average the second and fourth receivers. Type:

```
2 4 PAIR
SLIDE
```

The results will be left in the second receiver. If you wish the results to be left in the fourth receiver, then type:

```
4 2 PAIR
SLIDE
```

PEAK

PEAK finds the CENTER, HWIDTH and HEIGHT of a single gaussian. Prints CENTER point, HWIDTH and PEAK temperature. Sets BGAUSS and EGAUSS for the GAUSS verb.

PTWH	Determines which array is to be used.
ADVERBS	Sets CENTER(1), HWIDTH(1), HEIGHT(1), BGAUSS and EGAUSS.
BGAUSS	is set to (CENTER(1) - HWIDTH(1))
EGAUSS	is set to (CENTER(1) + HWIDTH(1))
BDROP(rcvr) EDROP(rcvr)	Beginning and ending channel numbers between which to search for the peak temperature,.

RELATED VERBS

GAUSS	Refines the first guesses of CENTER(N), HWIDTH(N), and HEIGHT(N) for N gaussians.
GMODEL	Evaluates the parameters of the gaussian(s), constructs a model of the gaussian (or the sum of the gaussians), and replaces the data in the currently referenced array with the model. Also prints the value of the RMS error of the fit.

REMARKS

PEAK searches for the largest channel value greater than zero. PEAK can find smaller peaks if BDROP and EDROP are used to window the region.

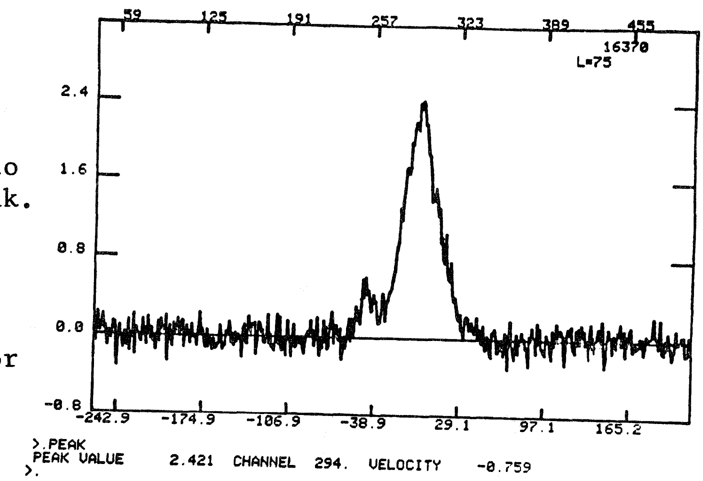
-continued-

EXAMPLE

You have just removed a baseline from scan 16370 and produced the display at right. Now you want to fit a gaussian to the primary peak. Type:

PEAK

to set all the initial guesses for the gaussian fit. The output is seen on the display at right.



You are now ready to attempt a gaussian fitting by specifying

GAUSS

PIX

THIS ROUTINE IS ONLY AVAILABLE ON THE MODCOMP

PIX is a routine which writes data points on a tape to be processed on the Charlottesville IBM to generate a DICOMED PIX TAPE of a velocity vs. position contour plot on the Dicomed. PIX processes one scan at a time and the user must order the scans from "top to bottom" and use PIX for each scan. The verb LABEL must be used after the last scan to complete the plot.

PTWH	Determines which array is to be contoured. Is not changed.

ADVERBS

The following adverbs are required by the verb PIX. They must be set before a map is started and cannot be changed before LABEL is used to complete the map. Otherwise, the results will be unpredictable.

BDROP(rcvr) EDROP(rcvr)	The number of channels at the beginning and end of the receiver which are to be ignored.
YUNITS	Indicated the coordinate and the units of the Y-axis. The first character specifies the corrdinate (R, D, L, B) and the second specifies the units (H, D, M, S). The default is "DD".

The following adverbs are required by the verb LABEL to complete the map. They may be set either before beginning a map or any time before LABEL is used.

IROW ICOL	The number of additional points between each channel and each scan to be interpolated by the program.
MRATIO	Varies the relative X-Y scaling of the map. The default is 1.
TITLE	A user-specified alternative name to lable the map. TITLE may contain from 1 to 12 characters. If TITLE contains all blanks, then the source name of the last scan used in the map will be used to label the map. The default for TITLE is a blank string.
XMINCR	The units (km/s) per inch for the x-coordinate. The default is 20.
YMINCR	The units (as specified by the second character of YUNITS) per tick for the y-coordinate.

-continued-

RELATED VERBS

EXIT	Writes an end-of-file on the plot tape.
LABEL	Completes the map.
MAP	Used instead on PIX to make a CALCOMP contour map.
MAPSHOW	Produces a CRT contour map.
PLOT	Writes a spectrum on the plot tape for the CALCOMP plotter.

REMARKS

See the chapter on off-line data display for more information.

The receiver processing options ALL or ONLYA will cause the first receiver to be mapped. To map the other recievers, use ONLYB, ONLYC, ONLYD, or the verb SELECT.

The verb LABEL must be used to complete each map. The routine assumes that there is a tape with a write-ring on the correct drive. (See the separate section on off-line plotting.) At the end of a terminal session, the verb EXIT is needed to write an end-of-file on the tape.

Documentation on the use of the DICOMED can be found in the manual:

The NRAO Image Recording System

by Thomas R. Cram and Eric W. Griesen.

The size of the map affects the post-processing on the Charlottesville IBM. If the number of points to be contoured (# points per scan times # scans) is greater than 5000 for any map, then the post-processing step must be run as a class L job.

PIX is most conveniently used within a procedure. If the scans have been preprocessed and reloaded from a KEEP tape, or if a baseline can be fitted to all the scans with common parameters, then the STACK is an easy way to order the scan numbers to be contoured.

-continued-

ERRORS

If LABEL has not been used after last PIX, the program gives the error message:

USE LABEL!

If a tape is not mounted on the drive when PIX is used, the program gives the error message:

MOUNT TAPE!

If there is no write ring in the tape on the drive, the program gives the error message:

WRITE RING?

Any other tape-related error causes the program to give the error message:

TAPE ERROR!

PLOT

PLOT is a routine which writes data on a tape to be processed on the IBM in order to make CALCOMP plots of the spectra. See the appropriate sections in Chapter 5 on writing PLOT tapes. The off-line plotting will scale the data or the user can control the scaling and annotation by setting the adverbs used by PLOT.

<u>PTWH</u>	Determines which array is plotted.

<u>ADVERBS</u>

BDROP(rcvr) EDROP(rcvr)	The number of channels at the beginning and end of the receiver which are to be ignored.
BMARK	If =1, the regions last used to compute a baseline or RMS will be indicated by boxes. Set BMARK=0 to supress these boxes.
BORDER	Controls the type of border to be drawn around the plotting region. BORDER may take as values the character strings 'WHOLE', 'HALF', or 'NONE', indicating a border on all four sides, just the left and bottom sides, or no border, respectively.
CMARK(6)	Channels to be marked with vertical lines on the plot. Set CMARK = 0 to supress these lines.
NPLOT	Controls the scaling of the plot when the plot is on small paper, or the scaling of the plot and the number of plots per page when the plot is on large paper.
PUNIT	Causes the specified coordinates of the plot to be printed in the lower right hand corner of the plot. ('RD','LB')
REMARK(64)	A string of 1 to 64 characters to be printed at the left-hand corner of the plot.
TMARK(6)	Temperatures to be marked with horizontal lines on the plot. Set TMARK=0 to supress these lines.
XMIN(rcvr) YMIN(rcvr)	The minimum velocity and temperature to be plotted. If XMIN or YMIN is less than -99999, then the plotting program will scale the x or y-coordinate, respectively.

-continued-

XINCR(rcvr) YINCR(rcvr)	The increment per inch for the x-axis (velocity) and the y-axis (temperature) if XMIN or YMIN is greater than -99999, respectively.
XTITLE(20) YTITLE(20)	A string of 1 to 20 characters used to label the x- and y-axes.
ZLINE	If =1, a horizontal line will be drawn at zero kelvin. Set zmark=0 to supress this line.

RELATED VERBS

PROFILE	"Stacks" several scans on a 10 x 25 inch CALCOMP plot.
REPLOT	Plots a scan on top of the last CALCOMP plot.
SHOW	Plots a spectrum on the CRT screen.

REMARKS

Any or all of the receivers may be plotted, depending on the receiver processing option.

on the MODCOMP, the plots are written directly to tape, so a tape with a write-ring needs to be mounted on the tape drive. On the IBM, the plots are written to a disk file, and after you have written all of your plots to the file, you need to EXIT the program and run the external program PLOTTAPE to write the plots from the disk file to tape. For more information, see the appropriate sections in Chapter 5 on writing plot tapes.

Also, see the chapter on off-line data display.

The base size of a CALCOMP plot is five inches by ten inches. The plotting program uses the adverb NPLOT to scale the size of the plot by a factor of

4 / NPLOT

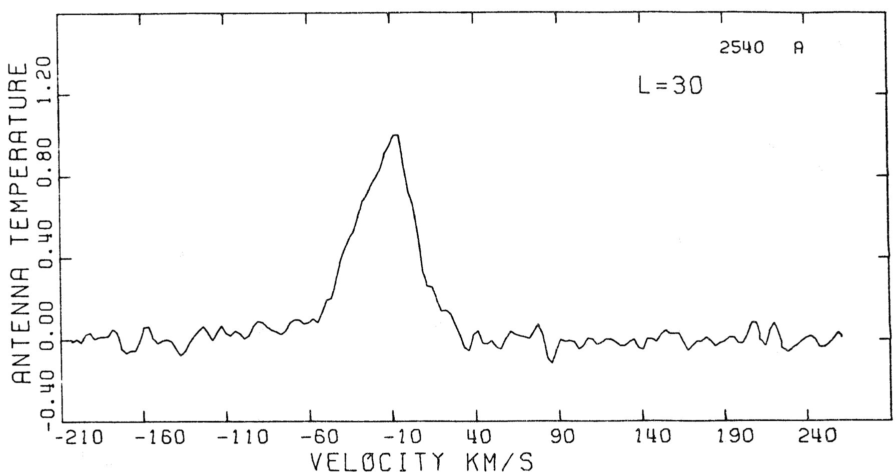
so that NPLOT = 4 will produce plots 5" x 10", NPLOT = 8 will produce plots 2.5" x 5". If plots are drawn on the large size paper (36 inches wide), the NPLOT is also the number of plots drawn before the plotter is shifted in the x-direction to make a "new page." NPLOT should be set to a number between 2 and 20.

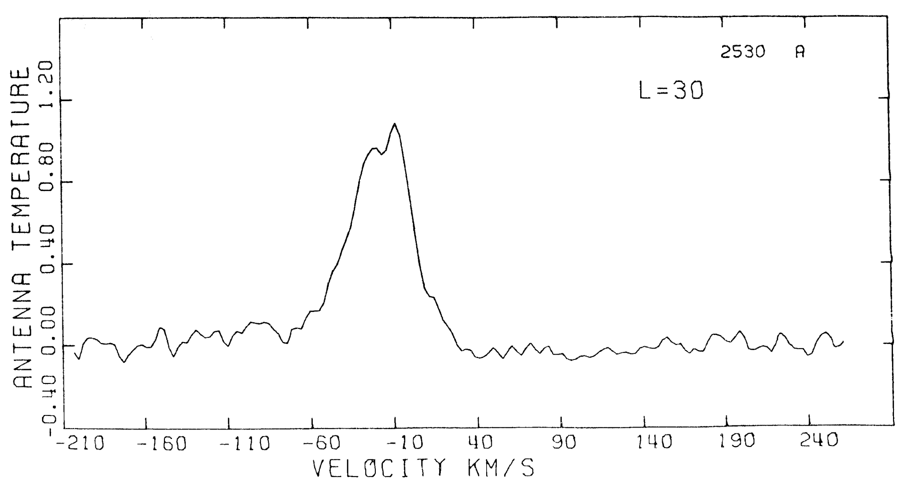
-continued-

Velocity and range control: The plotting program uses the adverbs XMIN(rcvr) and XINCR(rcvr) to control the minimum velocity and increment per inch for each receiver. If XMIN = -99999, then the velocity scale will be chosen by the plotting program. If you wish to set the velocity scale, both XMIN and XINCR must be set to control the scaling. If you are specifying the velocity range on a plot to cover only a portion of the spectrum, then you must take care to drop the channels outside the plotted region by setting BDROP(rcvr) and EDROP(rcvr).

Temperature and range control: The plotting program usus adverbs YMIN(rcvr) and YINCR(rcvr) to control the minimum temperature and temperature increment per inch for each receiver. If YMIN = -99999, then the temperature scale will be chosen by the plotting program. If yousish to set the temperature scale, both YMIN and YINCR must be set to control the scaling. In this case, it is advisable to use the verb CLIP to limit the spectral values to the desired range.

Examples of the PLOT verb are shown below:





PMH PMT PMW

These verbs set the value of PTWH so that it points at one of the three arrays.

PMT sets PTWH = 1 (which means TEMP array)

PMW sets PTWH = 2 (which means WORK array)

PMH sets PTWH = 3 (which means HOLD array)

PTWH Is set as described above.

ADVERBS None.

RELATED VERBS None.

REMARKS

These three verbs are used to change the pointer to the desired array prior to performing operations which work on "the currently referenced array".

EXAMPLE

To display the contents of all three arrays on the same graph:

```
PMW PAGE SHOW
PMT RESHOW
PMH RESHOW
```

POINTS

A routine which sets the plotting mode for the SHOW display to points.

<u>PTWH</u> Is not used or changed by POINTS.

<u>ADVERBS</u> None.

<u>RELATED VERBS</u>

HISTOGRAM Sets the SHOW display mode to a histogram.

LINE Sets the SHOW display mode to a continuous line.

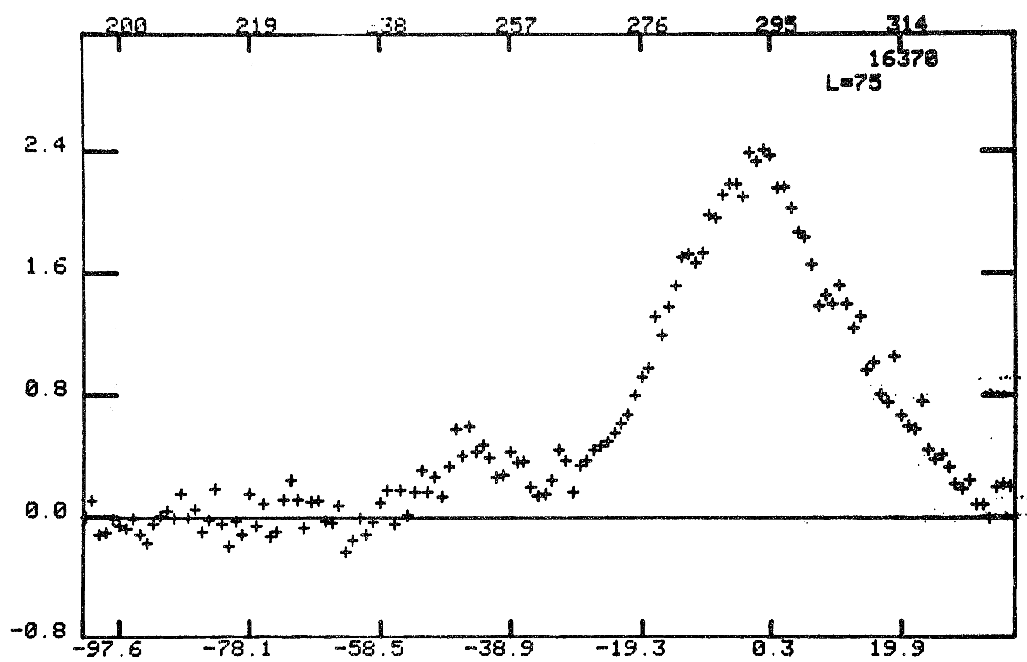
SHOW Displays a scan with user specified or default scaling.

<u>EXAMPLE</u>

You wish to display the current scan in points mode. Specify:

POINTS PAGE SHOW

and you will get



PRINT

PRINT field prints the field requested on the CRT screen. field may be one or more variable or adverb names, literals, or arithmetic expressions. If field is an adverb or variable (a variable is an array or scalar defined in a procedure) name, the value(s) of the adverb or variable will be printed. Arrays of more than one dimension are printed with the first index varying most rapidly, as in Fortran. If field is a literal, it must be enclosed in single quotation marks, and it will be printed without the quotation marks. If field is an arithmetic expression, the result of the expression will be printed.

PTWH Is not used or changed.

ADVERBS None.

OBJECTS

field As described above.

RELATED VERBS

READ field Reads the values requested by field from the CRT. In this case field may be one or more variable or adverb names, and may not include expressions or literals.

? field
field ? Alias for PRINT.

PRINTER Directs the PRINT output to the hard copy device.

CRT Directs the PRINT output to the CRT screen. CRT is the default assignment, but must be invoked after PRINTER to return output to the CRT.

REMARKS

PRINT can be used to monitor the values of adverbs, PTWH, SIZE and VRMS.

-continued-

EXAMPLES

You type	PRINT CENTER
Response is	25.0000 33.0000 78.0000 0.0000 0.0000 0.0000
You type	PRINT 'YOU SHOULD USE THE 300-FOOT'
Response is	YOU SHOULD USE THE 300-FOOT
You type	PRINT SQRT(144)
Response is	12.0000
You type	PRINT HWIDTH(2)
Response is	5.0000
You type	X = 1.25 PRINT 'x = 'X
Response is	x = 1.2500
You type	PRINT BMARK ZLINE YMIN
Response is	1.0000 0.0000 -9999.9999 -9999.9999

PRINTER

Used to direct the output of certain print routines to the Versatek printer. Those verbs affected are PRINT, SUMMARY, LIST, and TELL DISK.

PTWH — Neither used nor changed.

ADVERBS — None.

RELATED VERBS

CRT — Directs print output to the CRT screen.

REMARKS

CRT is the default print output direction. Using PRINTER may result in more readable copy than printing on the CRT screen and making a copy of the screen.

PROCEDURE

PROCEDURE is used to define a routine which can then be executed by entering its name on the terminal. (Procedurenames are essentially user-defined verbs.) The PROCEDURE statement is of the form

PROCEDURE procedurename (optional arguments)

and begins the definition of the procedure. The complete definition of a procedure has the form:

PROCEDURE procedurename (optional arguments)
statements the procedure is to execute
FINISH

The PROCEDURE statement puts the program into the COMPILE mode. When in the COMPILE mode, the program prompts with a colon (:) instead of a caret (>). The FINISH statement is required to complete the definition of the procedure and return the program to EXECUTE mode.

<u>PTWH</u> Is not used or changed.

<u>ADVERBS</u> None.

<u>OBJECTS</u>

procedurename The name of the procedure which is being defined. A procedurename may be any alpha-numeric name up to 10 characters long. The first character of the name must be an alphabetic character. Procedure names are defined globally, and so must be unique.

arguments Variables which are to be passed to the procedure when it is called by the user. Arguments may be any alphanumeric name up to 10 characters long. The first character must be an alphabetic character. New names are implicitly declared. There is no restriction on the number of arguments permitted, but they must be encased in parentheses and separated by commas.

-continued-

RELATED VERBS

FINISH	Completes the definition of a procedure.
EDIT procedurename line #	Initiates editing of a procedure.
ENDEDIT	Terminates editing of a procedure.
RETURN	Is used in a procedure that is called by another procedure.
HELP PROCEDURE	Lists the names of the already defined procedures.
LIST procedurename	Lists the requested procedure.
CORE	Tells the user how much core is left for definition of procedures.
RESTART	Empties core of all user defined procedures (and returns adverbs to their default values).

ERRORS

A fixed amount of space is available for defining procedures. This space is also used up by editing procedures. When the space has been completely used up, the program will give the message

BLEW CORE!

At this point you have three options:

(1) You can do without the procedure you were defining or editing when the message appeared. All other procedures will still be usable, but not able to be edited.
(2) You can wipe out all of your procedures (and adverb changes) by using RESTART. Then type in your correct procedures, hoping that if you don't do any editing, they will all fit.
(3) You can store your current set of procedures on the disk, using the STORE command. Then RESTART. Then define any additional procedures you need. Since RESTART kills your adverb values and array contents, if you have two "pages" of procedures (one on disk, and one in memory) which use each other's results, you will have to store intermediate results in scan save bins. Intermediate variable results (for example, SIZE) will be lost.

-continued-

EXAMPLE

To define a procedure to add two scans together:

```
PROCEDURE ADDSCANS (SCAN1,SCAN2)
GET (SCAN1)
FACT=-1;SCALE TRT
GET (SCAN2)
DIFF
PAGE SHOW
FINISH
```

Now to use ADDSCANS, you specify

```
ADDSCANS (710,712)
```

PROFILE

PROFILE plots manu apectral profiles on a 10 inch by 25 inch CALCOMP plot. Each call to PROFILE after the first plots a new profile, displaces by the last profile by the adverb YMINCR. If the next profile cannot fit within the given 25 inches, then a new plot is started.

PTWH	Is not changed.

ADVERBS

BDROP(rcvr) EDROP(rcvr)	The number of channels at the beginning and end of the receiver which are to be ignored.
XINCR(rcvr) YINCR(rcvr)	The increment per inch for the x-axis (velocity) and the y-axis (temperature) if XMIN or YMIN is greater than -99999, respectively.
XMIN(rcvr) YMIN(rcvr)	The minimum velocity and temperature to be plotted. If XMIN or YMIN is less than -99999, then the plotting program will scale the x or y-coordinate, respectively.
YMINCR	Separation in inches between profiles.
XTITLE(20) YTITLE(20)	A string of 1 to 20 characters used to label the x- and y-axes.
ZLINE	If =1, a horizontal line will be drawn at zero kelvin. Set ZLINE=0 to supress this line.

RELATED VERBS

PLOT	Plots a scan on the CALCOMP plotter.
REPLOT	Plots a scan on top of the last CALCOMP plot.
SHOW	Plots a spectrum on the CRT screen.

-continued-

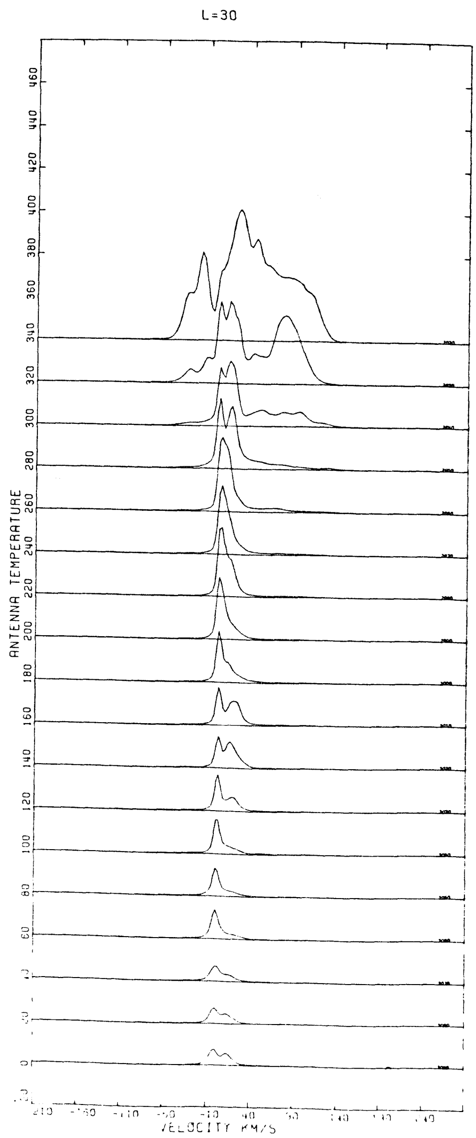
REMARKS

PLTSIZE = 0 should be specified on the MCPLOT procedure's EXEC card when using PROFILE. If the small paper default is taken, then the plotting routine will reduce the y-axis of teh plot from 25 inches to 12 inches.

See the chapter on off-line data display for information on producing your plots from the plot tape.

EXAMPLE

An example of a PROFILE is shown below.



RA

RA sets an internal flag which causes MAPSHOW to label the Y-axis of a velocity vs. position map as right ascension.

PTWH Is not used nor changed.

ADVERBS None.

RELATED VERBS

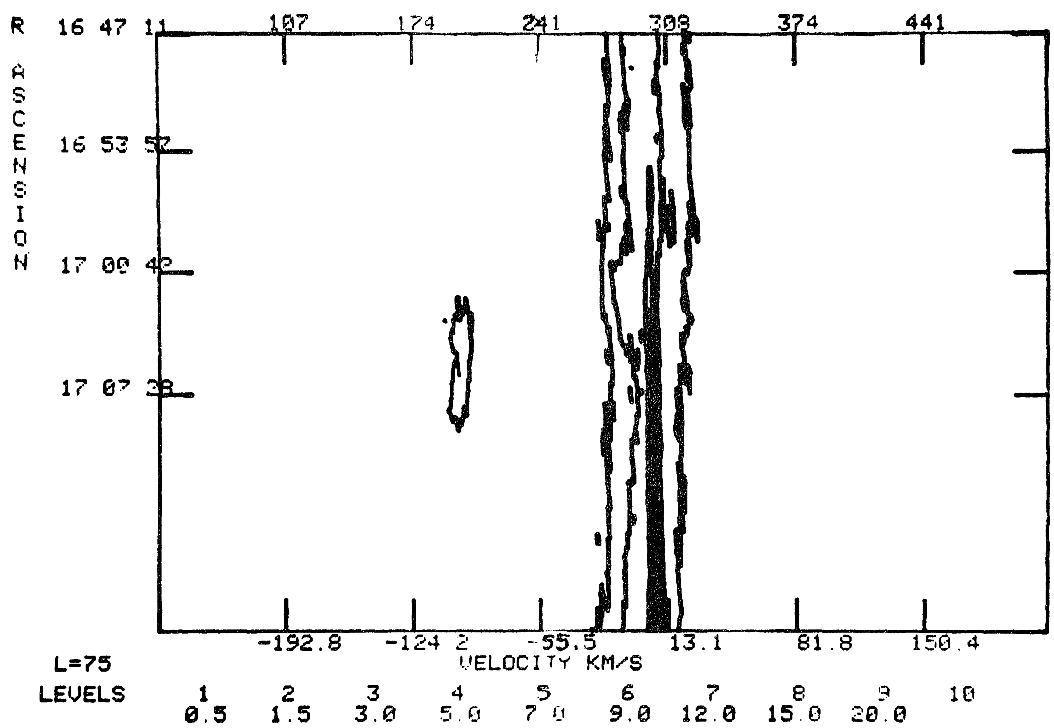
DEC Sets the y-coordinate label as declination.

GB Sets the y-coordinate label as galactic latitude.

GL Sets the y-coordinate label as galactic longitude.

REMARKS

GB does not affect the scaling of the y-axis. MAPSHOW and LABEL assume that the scans have been correctly spaced in the coordinate of interest.



RANGE

RANGE allows the user to specify minimum and maximum values for the y-scaling in SHOW. RANGE then sets the values of YMIN and YINCR.

PTWH Is not used or changed.

ADVERBS

YMIN(rcvr) Set by RANGE to ymin.

YINCR(rcvr) Is set by RANGE to (ymax - ymin) / 5

OPERANDS

ymin The minimum y-coordinate to be plotted.

ymax The maximum y-coordinate to be plotted.

RELATED VERBS

HOLDY Causes SHOW to reuse the previous y-scaling.
Done by setting YMIN to -9E10.

FREEY Cause SHOW to do automatic scaling for the y-axis.
AUTO

SPREAD(v1,v2) Sets BDROP and EDROP for v1 and v2.

REMARKS

The operands may precede or follow RANGE. If they follow RANGE, they must be enclosed in parentheses.
The scaling set by RANGE remains in effect until the verb FREEY is used or until the adverb YMIN(rcvr) is set to -9999.
RANGE is useful when stacking several scans or profiles on the CRT.

EXAMPLE

You want to display 10 scans with the same y-axis scaling (-2° to 8°).
You specify:

```
-2  8 RANGE
```

or

```
RANGE (-2,8)
```

RAPL RAPR

RAPL and RAPR are routines for averaging a frequency-switched scan by shifting the data DF MHz, subtracting the shifted data from the unshifted data, and dividing by 2. RAPL shifts the data DF MHz to the left, and RAPR shifts the data DF MHz to the right.

PTWH Determines which array is used. It is not changed.

ADVERBS

DF The frequency in MHz the data is to be shifted.

RELATED VERBS

OVERLAP Undoes Dual Dicke switching.

EXAMPLE

You want to combine the signal and reference spectra, so after setting DF, you specify

```
RAPL PAGE SHOW
```

RDIFF

RDIFF subtracts the contents of the second receiver of the scan in the currently referenced array from the contents of the first receiver of the scan in the currently referenced array. The result is left in the first receiver's channels.

PTWH — Determines for which scan the receivers are differenced. Is not changed.

ADVERBS — None.

RELATED VERBS

DIFF — Subtracts two scans.

PAIR (m,n) — Selects the two receivers to be subtracted by RDIFF.

SLIDE — Averages two receivers of the the currently referenced scan with weighting by integration time and temperature.

REMARKS

The processing option should be set by PAIR when RDIFF if used.

EXAMPLE

You are doing a polarization experiment. Receiver A is measuring right circularly polarized flux, and receiver B is measuring left circularly polarized flux. You want to difference the two recievers. You specify

PAGE SHOW	To display the A receiver
ONLYC RESHOW	To display the C receiver
1 2 PAIR	To select the two receivers
RDIFF	To difference the receivers
RESHOW	To display the difference.

READ

READ field reads from the CRT the values of the variables named in field . field may be one or more adverb or variable names (a variable is an array or scalar defined in a procedure). READ will respond with '#' until the user has entered enough values to satisfy field , at which time a caret (>) is returned.

PTWH	Is not used or changed.
ADVERBS	None.

OBJECTS

field	One or more adverb or variable names.

RELATED VERBS

PRINT field	Prints the values, or literals, requested by field on the CRT.
? field	Alias for PRINT

REMARKS

READ can be used to diminish the typing required to input the values of adverb arrays.

-continued-

EXAMPLE

To input the values of CENTER, which has six elements, you can specify

```
READ CENTER
```

the program will respond with

```
#
```

and then you input the values of CENTER -- either all at once:

```
#25 70 65 0 0 0
```

or one at a time:

```
#25
#70
#65
#0
#0
#0
```

When you have input six values, the program will give you a caret again.

The READ command can be used as a pause command in a procedure. For example, the procedure

```
PROCEDURE WAIT(FSCAN,LSCAN)
FOR I = FSCAN TO LSCAN
    GET (I) PAGE SHOW
    READ N
    END
FINISH
```

will graph FSCAN and then wait for the user to input something before it goes on to draw the next graph. The user can input either a number or a letter.

RECALL

RECALL copies the scan in the disk save bin named by the adverb NSAVE into the currently referenced array. Both the scan and its header information are copied from the disk bin into the array.

PTWH — Determines into which array the scan is copied. Is not changed.

ADVERBS

NSAVE — The number of the disk bin whose contents are copied into the currently referenced array.

RELATED VERBS

SAVE — Copies the currently referenced array into the disk bin NSAVE.

ERRORS

If NSAVE is greater than 80, the message

```
NSAVE ?
```

will be generated.

EXAMPLE

To recall two scans in two different arrays, you can retrieve them by

```
NSAVE = 65;  PMW RECALL
NSAVE = 66;  PMT RECALL
```

REGAUSS

REGAUSS is used to fit up to six gaussian functions. REGAUSS is identical to GAUSS except that REGAUSS does not calculate first guesses for the gaussian heights. REGAUSS can be used to continue iterating the results of a previous GAUSS, or in the case where you can guess good first approximations to the gaussian heights.

<u>PTWH</u>	Determines for which array the fit is calculated. Is not changed.

<u>ADVERBS</u>

NGAUSS	The number of gaussians to be fit. NGAUSS must be between 1 and 6.
BGAUSS EGAUSS	The explicit sample numbers where the fit is to begin and end.
CENTER(N)	The sample number where gaussian N has its center. Iterated by the routine.
HEIGHT(N)	The height in degrees Kelvin of gaussian N. Iterated by the routine. Must be set before entering REGAUSS if GAUSS has not been previously executed as REGAUSS will <u>not</u> set a first-guess itself.
HWIDTH(N)	The width in number of points of gaussian N at half its peak width. Iterated by the routine.
NITER	The number of iterations which will be made to try to find a fit. Initial value is 8.
FIXC	If FIXC is ≥ 1, REGAUSS will not iterate CENTER.
FIXHW	If FIXHW is ≥ 1, REGAUSS will not iterate HWIDTH.

-continued-

RELATED VERBS

GAUSS	Refines the user's first guesses of CENTER(N) and HWIDTH(N) for N Gaussians. Determines HEIGHT for each gaussian.
GMODEL	Evaluates the parameters of the gaussian(s), constructs a model of the gaussian (or sum of the gaussians), and replaces the data in the currently referenced array with the model. Also prints the parameters of the gaussians and the RMS error of the fit.
RESIDUAL	Evaluates the parameters of the gaussian(s) for each point and subtracts the total from the correctly referenced array.
GPARTS	Evaluates the parameters of, constructs and displays each gaussian separately.

REMARKS

REGAUSS is identical to GAUSS except that the routine assumes that the values in HEIGHT(N) are meaningful. REGAUSS does not change the contents of any of the three arrays.

ERRORS

If the fit is unsuccessful after NITER iterations, the message FIT FAILED is printed.

EXAMPLE

After making the fit in the example given in GAUSS you feel that convergence was not reached and you would like to iterate further. Specify

```
REGAUSS
```

To see the improved parameters, use TELL GPARMS or specify

```
PRINT CENTER (1) HEIGHT (1) HWIDTH (1)
```

REPLOT

A routine that plots the spectral values (usually modified by GMODEL) on top of the last CALCOMP plot.

If using the MODCOMP, this verbs writes directly to the plot tape.

If using the IBM, this verb writes to the plot disk file, and after exiting the program, you must execute the program PLOTTAPE to write the disk file to tape.

See the chapter on off-line data display for more information on producing CALCOMP plots.

<u>PTWH</u> Is not changed.

<u>ADVERBS</u> None.

<u>REMARKS</u>

Often used to plot a gaussian function fitted to a profile.

RESHOW

This routine plots the contents of the currently referenced array using the Y-scaling used by the last SHOW. RESHOW does not draw or label the axis.

<u>PTWH</u>	Determines which array is plotted. Is not changed.

ADVERBS

BDROP(CH) EDROP(CH)	The number of points at each end of each receiver channel which will not be plotted.

RELATED VERBS

RHIST	Sets the plotting mode for RESHOW to HISTOGRAM.
RLINE	Sets the plotting mode for RESHOW to LINE.
RPOINTS	Sets the plotting mode for RESHOW to POINTS.

REMARKS

RESHOW is often used to plot a polynomial baseline model, or gaussian model on top of the data for which the model was generated. RESHOW can also be used to plot the reduced data on top of the original.

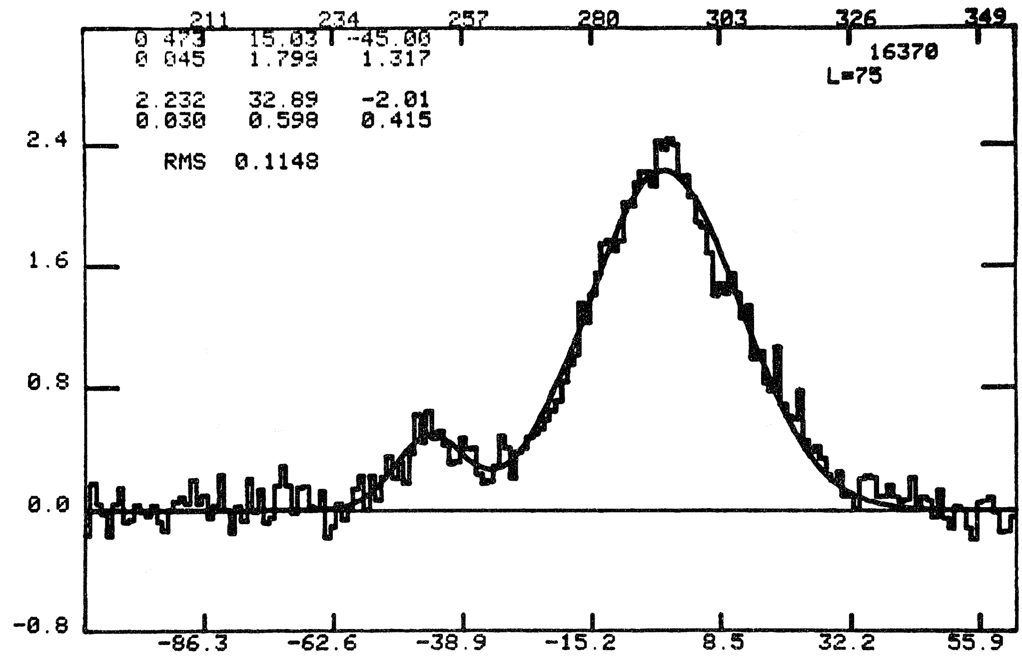
-continued-

EXAMPLE

After plotting scan 16370 using as a histogram, you wish to plot the gaussian model on top of the data. Specify:

LINE GAUSS GMODEL RESHOW

to produce the display below.



RESIDUAL

This routine takes the parameters of specified gaussians (usually refined by GAUSS), and subtracts the sum of the gaussians from the currently referenced array.

PTWH	Determines from which array the total model is subtracted. Is not changed.

ADVERBS

NGAUSS	The number of gaussians.
BGAUSS EGAUSS	The first and last points of the region over which the gaussian(s) are subtracted.
CENTER(N)	The center points of the gaussians.
HEIGHT(N)	The heights of the gaussians.
HWIDTH(N)	The half widths of the gaussians.

RELATED VERBS

GAUSS	Refines the user's first guesses of CENTER(N) and HWIDTH(N) for N gaussians. Determines HEIGHT for each gaussian.
GMODEL	Constructs the sum of the N gaussians using the parameters CENTER, HWIDTH, and HEIGHT, and replaces the currently referenced array with the sum.
PEAK	Finds the CENTER, HWIDTH and HEIGHT of a single gaussian. Also sets the BGAUSS and EGAUSS.

REMARKS

GAUSS may or may not have been called before RESIDUAL.

-continued-

EXAMPLE

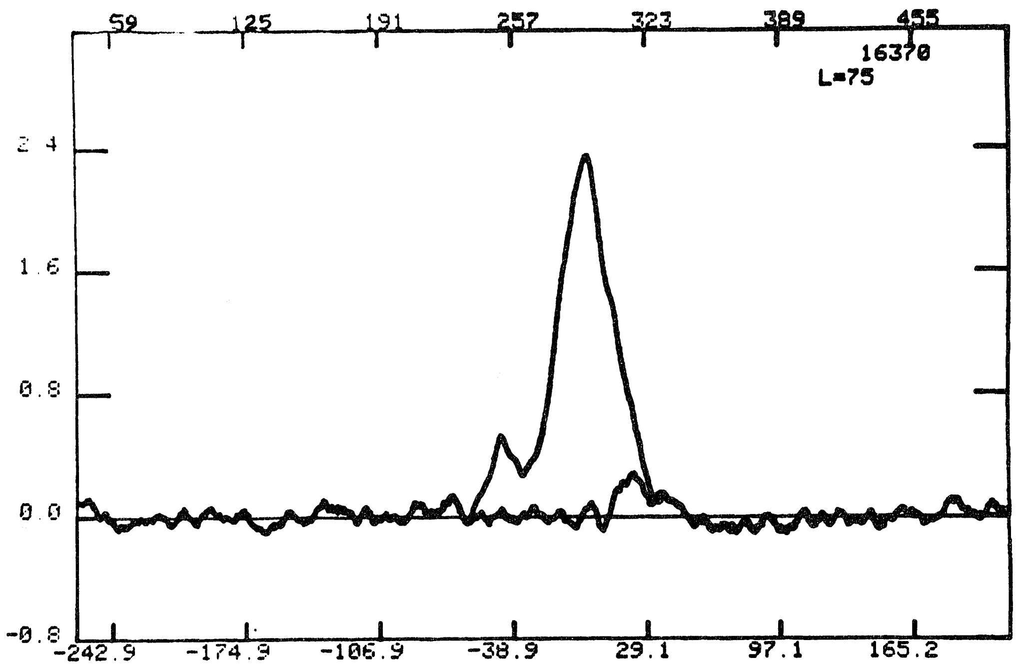
You had

GET 16370 BASELINE PAGE GAUSS SHOW

Now you want to remove the gaussians and see what the residual looks like. Specify

TRH RESIDUAL RESHOW

You will have



RESTART

RESTART re-initializes the program memory. It zeros the three arrays (WORK, TEMP, and HOLD), empties the procedure definition space, and sets the values of all the adverbs to their initial values. RESTART puts the program into the same condition it has immediately after loading the program.

<u>PTWH</u>	Is not used, but is set to the WORK array.

ADVERBS

All adverbs	Are set to their default values.

RELATED VERBS

STORE page#	Copies the procedure definition space and adverb values into a disk storage area called "page" page#.
RESTORE page#	Copies disk page page# into the program memory.

REMARKS

STORE, RESTORE, and RESTART are useful for control of the program memory space. RESTART is used to get a fresh page on which to write new procedures and to return the adverbs to their initial values.

EXAMPLE

You were trying to define a procedure and got the message

```
BLEW CORE
```

which tells you that you are out of procedure space. To empty the procedure space, enter

```
RESTART
```

but realize that once you have done this, the three arrays will be empty, your adverbs will return to their initial values, and all of your procedure definitions will be <u>gone</u>.

RESTORE

RESTORE page# restores to memory the procedure definition space stored on disk by the command STORE page#. Both procedure definitions and adverb values are restored.

PTWH	Is not used, but is reset to the value it had before the STORE page# command.

ADVERBS

All Adverbs	Are reset to the values they had before the STORE page# command.

OBJECTS

Page#	The disk page which is copied into memory. Page# must be an integer between 1 and 6.

RELATED VERBS

STORE page#	Creates a copy of the current state of memory in the specified disk page#.
RESTART	Restores the program memory to its original or default state.

REMARKS

STORE, RESTORE, and RESTART are used to control the program memory space. RESTORE restores the program to a prior condition saved for that purpose.

EXAMPLE

Remember that copy of memory you saved yesterday with all the procedures and adverb values just the way you wanted them? (See STORE example.) Well, you can get back to where you were before by specifying

```
RESTORE 5
```

RETURN

RETURN is required in procedures which are called by other procedures. The RETURN statement should immediately precede the FINISH statement when it is used.

PTWH Is not used or changed.

ADVERBS None.

RELATED VERBS

See PROCEDURE

REMARKS

A procedure which is called by another procedure must be defined before the procedure which calls it. That is, procedures must be defined in a "bottom up" fashion.

ERRORS

If the RETURN statement is omitted from a called procedure, the program exits that procedure to the EXECUTE communications mode rather than to the calling procedure.

If a procedure which is called by another procedure is not defined before the calling procedure, the message SYMBOL ? will be generated (and the program will return to EXECUTE mode) when the name of the not-yet-defined procedure is used while defining the calling procedure.

-continued-

EXAMPLE

To define a procedure which can be called by another:

```
>PROCEDURE BUSY(T)
:R = T * T
:PRINT T R
:RETURN
:FINISH
>
```

Now define the calling procedure:

```
>PROCEDURE EG
:T = 12
:BUSY(T)
:PRINT 'THATS ALL FOLKS'
:FINISH
>
```

Now you can execute BUSY through EG by

```
 EG
```

and you will get

```
 12.0000  144.0000
THATS ALL FOLKS
```

RHIST RLINE RPOINTS

These routines control the type of plot made by the RESHOW verb. RHIST sets the RESHOW display mode to histogram. RLINE sets the RESHOW display mode to a continuous line and RPOINTS sets the RESHOW display mode to points.

PTWH Is not used or changed by RHIST, RLINE or RPOINTS.

ADVERBS None.

RELATED VERBS

HISTOGRAM Sets the SHOW display mode to a histogram.

LINE Sets the SHOW display mode to a continuous line.

POINTS Sets the SHOW display mode to discrete points marked by crosses.

RESHOW Graphs the currently referenced array on the CRT, no labels or borders, using the scaling last used by SHOW.

SHOW Graphs the currently referenced array on the CRT with labels and borders.

EXAMPLE

You have just fit a double gaussian to scan 520 and modeled the gaussian over the data. Now you would like to also see the residual by specifying

```
PMH RESIDUAL RPOINTS RESHOW
```

RIPPLE

RIPPLE is used to remove a sinusoidal frequency baseline from the data. RIPPLE fits a sine curve to a specified region or regions of a scan, given an approximate frequency (in MHz). The sine curve is then evaluated over all channels and subtracted from the scan.

PTWH	Determines for which scan the sine curve is fitted and subtracted. Is not changed.

ADVERBS

RFREQ	The approximate frequency (in MHz) of the sinusoidal baseline to be removed. RFREQ is iterated by the routine.
BDROP(rcvr) EDROP(rcvr)	The number of channels at each end of the receiver which will be ignored by the routine.
NREGION(rcvr,N)	Specifies the region or regions of the scan to be used to fit the sine curve. N is an integer between 1 and 8.
BBASE(rcvr) EBASE(rcvr)	The number of channels at each end of each receiver (not including BDROP and EDROP) which will be used compute the sine curve, if NREGION(rcvr,1) = 0.

RELATED VERBS

RSHAPE	Computes the sinusoidal fit for the indicated scan.
RMODEL	Evaluates the last computed fit for each channel and replaces the currently referenced array with the model.
BASELINE	Computes and subtracts a polynomial baseline from the currently referenced array.
BMODEL	Constructs a model of the previously computed polynomial baseline in the currently referenced array.
BSHAPE	Computes a polynomial baseline for the currently referenced array.

-continued-

REMARKS

The regions used to compute the fit should not contain a known or suspected spectral feature.

The sinusoidal baseline may be removed from one or more receivers, depending on the receiver processing option.

ERRORS

If NREGION(rcvr,N), where N is even, is less than NREGION(rcvr,N-1), the message

    NREGION ?

will appear. The endchannel of a fitting region must be greater than the start channel.

If the routine cannot fit a sine curve with the requested frequency, the message

    FIT FAILED

is generated.

EXAMPLE

```
GET 7238 PAGE SHOW
BDROP=25;EDROP=25
RFREQ=10
RIPPLE PAGE SHOW
```

To display the baseline before subtracting, use RSHAPE and RMODEL. In the follwing example we define a procedure:

```
>PROCEDURE RPL
:TRT RSHAPE RMODEL RESHOW
:READ N
:PMW DIFF PAGE SHOW
:RETURN
:FINISH

 GET 7238 PAGE SHOW
 BDROP=25;EDROP=25;RFREQ=10
 RPL
```

In the procedure, the data is copied to the TEMP register where it is eventually replaced by the baseline. By moving the pointer to the WORK array with the verb PMW, the verb DIFF causes the baseline to be subtracted from the data. Finally, the corrected data are plotted on a new page. The READ statement simply serves as a pause.

RMODEL

RMODEL evaluates the last calculated sine curve fit for each channel and replaces the currently referenced array with the sine curve.

PTWH	Determines which array is replaced by the sine curve. Is not changed.
ADVERBS	None used directly -- RMODEL uses the results of the previous RIPPLE or RSHAPE.

RELATED VERBS

RIPPLE	Calculates the sine curve, evaluates it for each channel, and subtracts the result from the currently referenced array.
RSHAPE	Calculates the sine curve for the currently referenced array.
BASELINE	Computes and subtracts a polynomial baseline from the currently referenced array.
BMODEL	Constructs a model of the previously computed polynomial baseline in the currently referenced array.
BSHAPE	Computes a polynomial baseline for the currently referenced array.

REMARKS

Most often used in conjunction with RSHAPE to compute a sine curve without subtracting it from the data. It is often useful to plot the sine curve on top of the data for comparison purposes.

One or more receivers may be replaced by a sine curve, depending on the processing option.

-continued-

EXAMPLE

You have twenty scans of the same source and you want to remove the same sine curve from each of them. You can do this by

For TPOWER data:

```
FETCH 6106 TEMP TRT RSHAPE
RMODEL PMW DIFF SHOW
FETCH 6102 TEMP DIFF PAGE SHOW
FETCH 6103 TEMP DIFF PAGE SHOW
        .
        .
        .
FETCH 6126 TEMP DIFF PAGE SHOW
```

For SPOWER data:

```
GET 7238 TRT RSHAPE RMODEL PMW DIFF SHOW
GET 7239 DIFF PAGE SHOW
GET 7240 DIFF PAGE SHOW
GET 7241 DIFF PAGE SHOW
      .
      .
      .
GET 7258 DIFF PAGE SHOW
```

For another example, see the verb RIPPLE.

RMS

This routine computes the sample root mean squared of the scan in the currently referenced array, using the specified region or regions. The result is stored in the variable VRMS(CH).

PTWH	Determines for which scan the RMS is computed. Is not changed.

ADVERBS

BDROP(CH) EDROP(CH)	The number of points at each end of each receiver channel which will be ignored by the routine.
NREGION(CH)	Specifies the region or regions of the scan to be used to compute the RMS. N is an integer between 1 and 8.
BBASE(CH) EBASE(CH)	The number of points at each end of each receiver channel (not including the points dropped by BDROP and EDROP) which will be used to compute the RMS if NREGION(CH,1)=0.

RELATED VERBS

None.

REMARKS

The RMS computed is stored in the variable VRMS.

-continued-

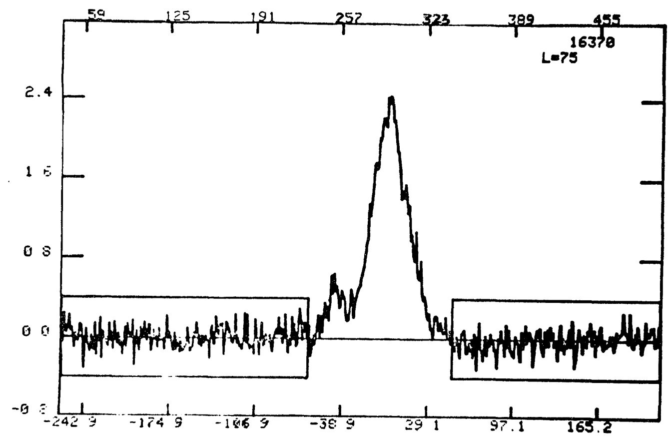
EXAMPLES

To compute the RMS of the scan in the currently referenced array, specify

```
BMARK=1;
PAGE SHOW
```

The boxes indicate the regions over which the RMS will be computed. To compute and print the RMS, specify

```
RMS
```




```
> RMS
RMS FOR RCUR 1: 0.1065 2: 0.1109
```

RSHAPE

RSHAPE fits a sine curve to the scan in the currently referenced array, using the specified region or regions of the data and the approximate frequency specified. RSHAPE does not evaluate the fit for each channel.

PTWH	Determines for which scan the sine curve is fitted and subtracted. Is not changed.

ADVERBS

RFREQ	The approximate frequency (in MHz) of the sinusoidal baseline to be removed. RFREQ is iterated by the routine.
BDROP(rcvr) EDROP(rcvr)	The number of channels at each end of the receiver which will be ignored by the routine.
NREGION(rcvr,N)	Specifies the region or regions of the scan to be used to fit the sine curve. N is an integer between 1 and 8.
BBASE(rcvr) EBASE(rcvr)	The number of channels at each end of each receiver (not including BDROP and EDROP) which will be used compute the sine curve, if NREGION(rcvr,1) = 0.

RELATED VERBS

RIPPLE	Calculates the sine curve fit, evaluates it for each channel, and subtracts the model from the currently referenced array.
RMODEL	Evaluates the last computed fit for each channel and replaces the currently referenced array with the model.
BASELINE	Computes and subtracts a polynomial baseline from the currently referenced array.
BMODEL	Constructs a model of the previously computed polynomial baseline in the currently referenced array.
BSHAPE	Computes a polynomial baseline for the currently referenced array.

-continued-

REMARKS

RSHAPE is most frequently used in conjuction with RMODEL to construct a model of a particular sine curve. The set of commands

TRH RSHAPE RMODEL PMW DIFF

is equivalent to RIPPLE.

RIPPLE can calculate the fit for one or more receivers, depending on the processing option.

ERRORS

If a sine curve of the requested frequency cannot be fit, the message

FIT FAILED

is generated.

EXAMPLE

See RIPPLE, RMODEL.

SAVE

SAVE copies the scan in the currently referenced array into one of the save areas on the disk called "bins". Both the scan and its header information are copied into the save bin named by the adverb NSAVE.

<u>PTWH</u> Determines which scan is saved. Is not changed.

ADVERBS

NSAVE The number of the disk bins into which the scan in the currently referenced array is put. NSAVE must be an integer between 1 and 80.

RELATED VERBS

RECALL Copies the scan in bin NSAVE into the currently referenced array.

ERRORS

If NSAVE is greater than 80, the message

NSAVE ?

is generated.

EXAMPLE

You are observing between noon and six o'clock. It is now 5:59 and the next observer is anxious to get on the computer. You have a partly reduced scan in the WORK array and an incomplete stack of scans in the HOLD array. To save both of these until you can get back on the computer, specify

```
NSAVE = 65; PMW SAVE
NSAVE = 66; PMH SAVE
```

SCALE

SCALE multiplies the data in the currently referenced array by the values of the adverb FACT(CH). FACT may be specified independently for each receiver channel, and may have any real value.

PTWH	Determines which scan is scaled. Is not changed.

ADVERBS

FACT(CH)	The value by which each receiver channel is to be multiplied.

RELATED VERBS

BIAS	Also uses the adverb FACT. BIAS adds FACT(CH) to the data in each receiver channel.

EXAMPLE

SCALE is often useful when the values of the y-coordinate are so large that the tic mark labels overflow the formats used by SHOW. If the numbers are too large, set FACT to a small number and SCALE the data:

```
FACT = 1/1000
SCALE PAGE SHOW
```

SCALE is also useful for negating a scan. The trick for adding two scans is to negate one:

```
FACT = -1; SCALE
```

and then to use the verb DIFF to subtract them. You may need to negate the result again.

SCLEAR

SCLEAR is a routine which clears the accumulator (HOLD) array before stacking data.

<u>PTWH</u> Is not used or changed by SCLEAR.

<u>ADVERBS</u> None.

<u>RELATED VERBS</u>

ACCUM	Adds the scan in the WORK array to the contents of the HOLD array.
AVE	Divides the HOLD array by the number of the accumulated scans.

<u>EXAMPLE</u>

See ACCUM.

SCRATCH

SCRATCH deletes the specified procedure from the program, but does not return the procedure space used by the procedure to the program.

PTWH	Is not used or changed by SCRATCH
ADVERBS	None.
OBJECT	The name of the procedure to be deleted.

RELATED VERBS

See PROCEDURE

SELECT

SELECT is used to change to processing option of the scan in the currently referenced array. The processing option desired is specidied by setting the values of the adverbs SET1, SET2, and SET3.

PTWH — Determines which scan's processing option is changed. PTWH is not changed.

ADVERBS

SET1
SET2
SET3 — The values of these adverbs are used together to indicate which processing option is desired for the scan in the currently referenced array.

Option Desired	SET1	SET2	SET3
Only A	1	1	1
Only B	2	2	1
Only C	3	3	1
Both A & B	1	2	1
Both A & C	1	3	2

RELATED VERBS

ONLYA — Is equivalent to SET1=1; SET2=1; SET3=1; SELECT

ONLYB — Is equivalent to SET1=2; SET2=2; SET3=1; SELECT

ALL — Is equivalent to SET1=1; SET2=#rcvrs; SET3=1; SELECT

PAIR(m,n) — Is equivalent to SET1=m; SET2=n; SET3=ABS(m-n); SELECT

-continued-

EXAMPLE

(1) TPOWER example

FETCH 6106 TEMP PAGE SHOW	Processing option is ALL, so receiver A is shown.
SET1=2; SET2=2; SELECT RESHOW	Processing option is changed to only the B receiver, so the B receiver is shown.
SET1=1; BASELINE PAGE SHOW	Processing option has been changed back to ALL, so the baseline is removed from both, and receiver A is shown.
SET1=2; RSHOW	Processing option has been changed to only the B receiver, so receiver B is shown.

(2) SPOWER example

same as above except that

FETCH 6106 should be replaced by GET 7238.

(3) Make a CRT map of any one receiver (SPOWER, using the stack)

```
PROCEDURE MAPPING
     PAGE
     FOR I = 1 TO ACOUNT
          GET ASTACK(I); SELECT BASELINE MAPSHOW
          END
     LABEL
     RETURN; FINISH
```

Set the necessary adverbs, load the stack:

For receiver A,

SET1=1; SET2=1; SET3=1; MAPPING

For receiver C,

SET1=3; SET2=3; SET3=1; MAPPING

SHOW

SHOW plots the contents of the currently referenced array on the CRT screen. The routine also draws and labels the axis with the default as follows:

y axis	antenna temperature	Kelvins
x axis	sample number and position	Points

SHOW will automatically compute the scaling needed to display the scan in the currently referenced array: however, the user may control the scaling by means of the adverbs BDROP, EDROP, YMIN and YINCR and the verbs FREEY, HOLDY, and RANGE.

<u>PTWH</u> Determines which array is plotted. Is not changed.

<u>ADVERBS</u>

BDROP(CH)
EDROP(CH) The number of points at each end of each receiver channel which will not be plotted.

BMARK If = 1, the regions last used to compute a baseline or RMS will be indicated by boxes.

CMARK(N) N is an integer between 1 and 6. If CMARK(N) is greater than 0, a vertical line will be drawn at point CMARK(N).

TMARK(N) N is an integer between 1 and 6. If TMARK(N) is greater than -999 a horizontal line will be drawn at temperature TMARK(N) in degrees K.

YMIN(CH) The minimum temperature that is to appear on the graph.

YINCR(CH) The user may specify how many degrees are to be between each y-axis tic mark. If YMIN is greater than -999, SHOW will not compute the y-axis scaling, but will use the values specified by YMIN and YINCR.

ZLINE If set to 1, a horizontal line will be drawn at zero K.

-continued-

RELATED VERBS

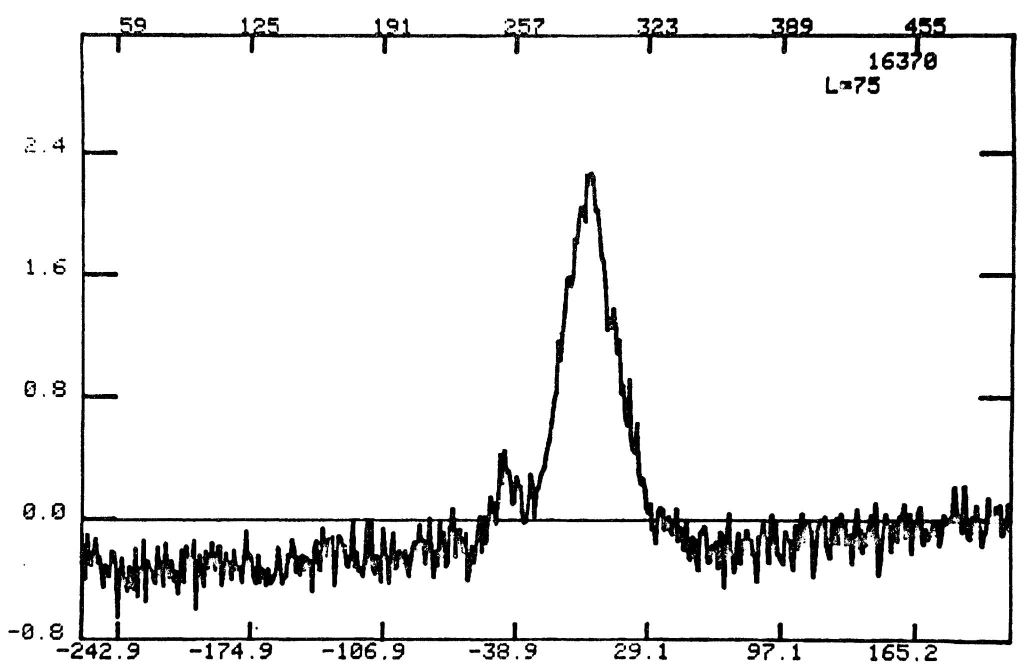
AUTO	Alias for FREEY (below).
FREEY	Sets YMIN to -90000. This causes SHOW to automatically scale the y-axis.
HISTOGRAM	Sets the SHOW display mode to a histogram.
HOLDY	Causes SHOW to retain the last determined y-axis scaling.
LINE	Sets the SHOW display mode to a continuous line.
POINTS	Sets the SHOW display mode to points marked by crosses.
RANGE	Defines the values of YMIN and YINCR according to the specified max and min values.

EXAMPLE

After retrieving scan 16370 from disk, you specify

PAGE SHOW

and your display will look like



SIN

A built-in sine function. Functions the same as the Fortran SIN.

PTWH — Neither used nor changed.

ADVERBS — None.

OBJECT

argument — An argument assumed to be in radians. The argument may be a constant, a variable, or an arithmetic expression, enclosed in parentheses.

RELATED VERBS

see COS, TAN, ATAN

EXAMPLE

```
PRINT SIN(0.7854)
0.7071
```

SLIDE

SLIDE is used to average two receivers in a scan. This routine averages the two receivers together with weighting by integration time divided by the system temperature squared. The resulting weighted average is put into the channels of one of the receivers. The system temperatures are also averaged with the same weighting factor as the data. The effective integration time of the averaged receivers is the sum of the integration times of the two receivers.

PTWH Determines which scan is slid. Is not changed.

ADVERBS

RSHIFT The number of channels by which the second will be shifted when averaged with the first receiver to properly align the spectral features.

RELATED VERBS

PAIR(m,n) Selects which two receivers are to be averaged. The results will be left in the first receiver specified.

RDIFF Subtracts receiver B from receiver A, without weighting, and puts the result in the receiver A channels.

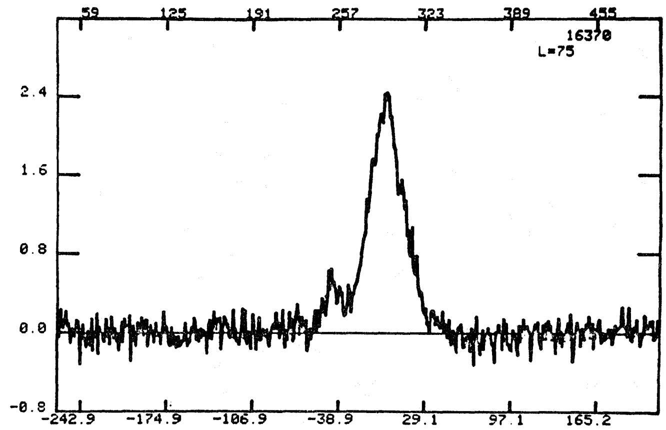
REMRRKS

The receivers to be processed should be set with the verb PAIR. It may be set with ALL if there are only two receivers.

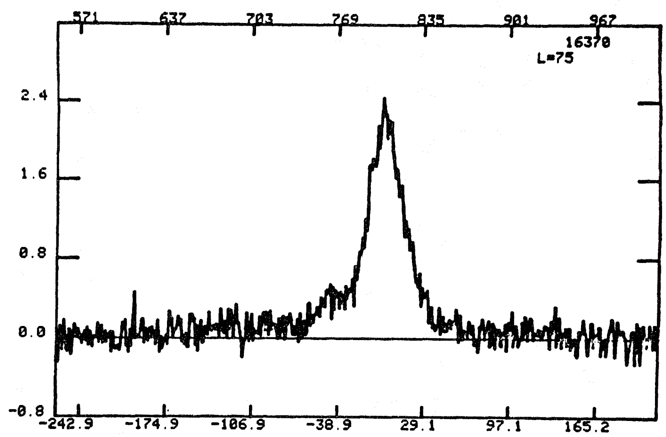
-continued-

EXAMPLE

You have a two-receiver scan in the currently referenced array.
Receiver A looks like



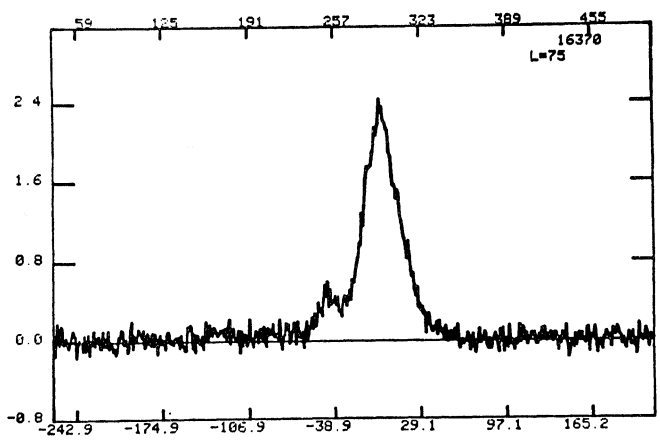
Receiver B looks like



You specify

1 2 PAIR
SLIDE
PAGE SHOW

Receiver A now looks like



SMOOTH

SMOOTH smooths the scan in the currently referenced array by convolution with the weight function SMWGT specified by the user.

PTWH — Determines which scan is to be smoothed.

ADVERBS

BDROP(CH) EDROP(CH)	The number of points at each end of each receiver channel which will be ignored.
SMWGT (12)	The smoothing function array. The first element of the array specifies the number of elements in the function - not greater than 11. Other elements specify the weighting of each point. The total of all weights should equal unity.

RELATED VERBS

BOXCAR	Smooths a scan by averaging a specified number of points together.
HANNING	A smoothing routine which averages three points together with the center point getting twice as much weight as either side point.

EXAMPLE

To mimick commonly used smoothing functions,

```
SMWGT = 3, 0.25, 0.50, 0.25; SMOOTH
```

is the same as HANNING and

```
SMWGT = 5, 0.2, 0.2, 0.2, 0.2, 0.2: SMOOTH
```

is the same as BOXCAR with NBOX=5. An example of a five point non-distorting smoothing function is

```
SMWGT = 5, -0.073427, 0.293706, 0.559441, 0.293706,
           -0.073427
```

SPREAD

This routine will restrict data processing and display to only those channels within the minimum and maximum velocities specified.

PTWH — Is neither used nor changed.

ADVERBS — Sets BDROP(rcvr) and EDROP(rcvr).

OBJECTS — The minimum and maximum velocities for which processing is to be limited.

RELATED VERBS

LIMIT — Limits the processing and display of the data to the minimum and maximum channel numbers specified.

REMARKS

SPREAD requires that there be at least 10 channels between the minumum and maximum velocities.

To return to the original limits, set BDROP(rcvr) and EDROP(rcvr) to zero, or use the RESET procedure.

SPREAD affects only the current receiver. If the option is ALL, then the processing limit for the A receiver is set.

ERRORS

If the first operand is greater than the second, if either operand is outside the velocity range of the receiver, or if the two velocities span fewer than 10 channels, SPREAD will give the error message

```
ARG LIST?
```

EXAMPLE

You wish to spread out the region between -150 km/sec and -50 km/sec. Specify,

```
-150, -50 SPREAD    or    SPREAD(-150,-50)
```

SQRT

A built-in square root function. Functions the same as the Fortran SQRT.

PTWH Neither used nor changed.

ADVERBS None.

OBJECT

argument The argument of SQRT may be a constant, a variable, or an arithmetic expression, enclosed in parentheses.

EXAMPLE

```
PRINT SQRT(144)
 12.000

NPOINTS = 144
PRINT SQRT(NPOINTS)
 12.000
```

STACK

A routine which lists all scan numbers currently in the STACK.

<u>PTWH</u>	Is not used or changed.

ADVERBS

ACOUNT	A pointer whose value is the number of scan numbers in the stack.
ASTACK	An array containing the stack of scan numbers.

RELATED VERBS

A #	Inserts one scan number into the STACK.
ADD (#,#)	Inserts a series of scan numbers into the STACK.
DELETE #	Deletes one scan number from the STACK.
EMPTY	Empties the STACK.

EXAMPLE

After defining the STACK by

    16370 A; 16380 A; 16390 A; 16400 A;

STACK will produce the following output:

    16370 16380 16390 16400

STITLE

TITLE writes one line of documentation about the current scan on the CRT screen including scan number, off-scan, date, source name, and rest frequency.

PTWH Determines which array is used by STITLE.

ADVERBS None.

RELATED VERBS

HEADER Prints header information about the currently referenced array on the CRT.

EXAMPLE

You want to compute the area under the curve of several profiles and document the results. You might define the following procedure:

```
PROCEDURE AREA (BS,ES)
FOR I = BS TO ES BY 2
    GET I BASELINE MOMENT STITLE PRINT SIZE (1)
    END
FINISH
```

STORE

STORE page# copies the procedure definition space into a disk storage space called a "page". It does not change the present state of memory; it merely makes a copy of memory for later retrieval. Both procedure definitions and adverb and variable values are stored.

PTWH	Is not used or changed.
ADVERBS	No adverbs are changed, but all adverb values are stored.

OBJECTS

Page#	The disk page into which the copy of memory is to be put. Page# must be an integer between 1 and 6.

RELATED VERBS

RESTORE page#	Copies disk page# back into memory.
RESTART	Wipes out the current state of memory by replacing it with the default or original state.

REMARKS

STORE, RESTORE and RESTART are used to control the program memory space. STORE saves the current state of memory for later restoration and use.

EXAMPLE

You are one of three observers presently using the telescope. You have gone to a great deal of trouble to set up your adverb values and procedures and now it is time to get off the telescope. You are not thrilled at the prospect of having to redefine your procedures and reset your adverbs tomorrow. Pick a number between 1 and 6, say 5, and specify

```
STORE 5
```

Now in disk page 5 there is a copy of the procedure definition space exactly like that in program memory.

SUM

A routine that is used to stack scans in the HOLD array with user supplied weighting. SUM after an AVE, SCLEAR or a program restart copies the header information and the data values in the WORK array to the HOLD array. Subsequent uses of SUM add the data values contained in the WORK array to the contents of the HOLD array. Subsequent uses of SUM do not change the header stored in the HOLD array, except for the integration time, which is incremented, and the system temperature, which is averaged with the specified weights. SUM increments the internal stack each time a scan is accumulated, and stores the numbers of the first 39 and the last scan accumulated.

PTWH — Must be pointing to WORK array.
Is set to the HOLD array.

ADVERBS

ASHIFT(rcvr) — The number of channels by which a scan is shifted by subsequenc calls to SUM. On the first call to SUM, the scan is not shifted even if ASHIFT is not equal to 0.

WEIGHT(CH) — The weighting used in accumulating the current scan.

RELATED VERBS

ACCUM — Same as SUM except the weighting is by the integration time divided by the system temperature squared.

ALIGN — Sets the values of ASHIFT, using the header information in the WORK array, so that when the scan in the WORK array is accumulated, its velocities will be correctly aligned with the velocities in the HOLD array.

AVE — Divides the accumulated scans by the number of scans accumulated. Sets the internal stack counter to zero.

SCLEAR — Sets the accumulator flag and internal stack counter to zero.

TELL CSTACK — Prints the internal stack counter and the numbers of scans accumulated.

-continued-

EXAMPLE

To stack scans weighted inversely by an RMS, define the following procedure:

```
PROCEDURE ADDS
FOR I = 1 TO ACOUNT
   GET ASTACK (I); RMS WEIGHT (1) = 1./VRMS (1); SUM END
AVE PAGE SHOW
FINISH
```

Load the appropriate scans in the STACK and execute ADDS.

SUMMARY

SUMMARY produces a listing of information about the scans on the analysis disk under a specified user number. A SUMMARY contains more information than a TELL DISK output since the actual data, and not just the scan number table in memory, are referrred to. Besides the scan number, a SUMMARY lists the source name, type of scan, right ascension and declination (epoch 1950.0), scanning rate, scanning direction, number of samples in the scan, and the calibration factor.

PTWH Not used but set to the TEMP array.

ADVERBS None.

OBJECT

user# NRAO Computer Division assigned user number, associated with your data through your setup or source cards.

RELATED VERBS

CRT Directs the print output of certain routines, including SUMMARY, to the CRT screen.

PRINTER Directs the print output of certain routines, including SUMMARY, to the printer.

REMARKS

SUMMARY destroys the contents of the TEMP array.

If you have no idea what data are on disk, but need to know more than the scan numbers, first use WIPE to get a list of the user numbers represented on disk. Then do a SUMMARY of each user number.

TABLE

TABLE is an informational verb. Each point is labelled with its channel number.

PTWH — Determines for which scan the table is printed. Is not changed.

ADVERBS — None.

OBJECTS — None.

RELATED VERBS

HEADER — Prints the header information for the scan in the currently referenced array.

EXAMPLE

To list the data values of scan 16370, Specify:

```
GET 16370 PAGE TABLE
```

A portion of the output will look like

CHNL	VELOCITY	TEMP	VELOCITY	TEMP	VELOCITY	TEMP	VELOCITY	TEMP
1	-302.72	1.211	-301.69	1.111	-300.66	0.819	-299.62	2.208
5	-298.59	4.076	-297.56	0.359	-296.53	1.817	-295.50	-0.525
9	-294.47	1.195	-293.44	0.634	-292.41	1.101	-291.38	1.324
13	-290.35	0.520	-289.32	1.028	-288.29	1.038	-287.26	1.828
17	-286.23	-1.064	-285.20	-0.171	-284.17	1.144	-283.14	0.534
21	-282.10	0.817	-281.07	0.443	-280.04	0.646	-279.01	0.238
25	-277.98	0.301	-276.95	0.468	-275.92	0.160	-274.89	-0.029
29	-273.86	0.416	-272.83	0.288	-271.80	0.082	-270.77	0.231
33	-269.74	-0.064	-268.71	-0.013	-267.68	0.077	-266.65	-0.050
37	-265.62	0.066	-264.59	0.069	-263.55	0.101	-262.52	0.090
41	-261.49	0.184	-260.46	0.047	-259.43	-0.038	-258.40	0.241
45	-257.37	0.069	-256.34	0.050	-255.31	0.129	-254.28	0.218
49	-253.25	0.065	-252.22	0.093	-251.19	0.003	-250.16	-0.099
53	-249.13	0.041	-248.10	-0.017	-247.07	-0.033	-246.04	0.124
57	-245.00	0.112	-243.97	0.036	-242.94	-0.308	-241.91	-0.051
61	-240.88	0.082	-239.85	-0.051	-238.82	-0.092	-237.79	-0.133
65	-236.76	0.017	-235.73	-0.109	-234.70	-0.122	-233.67	0.136
69	-232.64	-0.173	-231.61	0.001	-230.58	-0.044	-229.55	0.189
73	-228.52	-0.209	-227.48	0.052	-226.45	-0.008	-225.42	-0.013
77	-224.39	-0.069	-223.36	-0.089	-222.33	0.137	-221.30	-0.033
81	-220.27	0.041	-219.24	-0.114	-218.21	0.159	-217.18	0.003
85	-216.15	-0.057	-215.12	0.042	-214.09	-0.081	-213.06	-0.031
89	-212.03	-0.168	-211.00	-0.050	-209.97	0.107	-208.93	-0.092
93	-207.90	0.135	-206.87	0.042	-205.84	0.019	-204.81	0.022
97	-203.78	0.171	-202.75	-0.274	-201.72	-0.015	-200.69	0.069
101	-199.66	-0.066	-198.63	-0.125	-197.60	0.097	-196.57	-0.008
105	-195.54	0.026	-194.51	0.148	-193.48	0.015	-192.45	-0.025
109	-191.41	0.123	-190.38	0.124	-189.35	-0.128	-188.32	-0.041
113	-187.29	0.074	-186.26	-0.019	-185.23	0.016	-184.20	-0.095
117	-183.17	0.091	-182.14	-0.028	-181.11	-0.025	-180.08	-0.019
121	-179.05	-0.114	-178.02	0.050	-176.99	0.213	-175.96	-0.009
125	-174.93	0.052	-173.90	-0.006	-172.86	-0.084	-171.83	-0.123

TAN

A built-in tangent function. Functions the same as the Fortran TAN.

<u>PTWH</u> Neither used nor changed.

<u>ADVERBS</u> None.

<u>OBJECT</u>

argument The argument of TAN is expected to be in radians. The argument may be a constant, a variable, or an arithmetic expression, enclosed in parentheses.

<u>RELATED VERBS</u>

see COS, SIN, and ATAN

<u>EXAMPLE</u>

```
PRINT TAN(0.7854)
 1.000
```

TCUR

This routine activates the horizontal crosshair and returns the temperature value for the crosshair position on the current display when any key except RETURN is struck.

PTWH — Is not used or changed by TCUR.

ADVERBS — None.

RELATED VERBS

CCUR — Returns the sample number of the current vertical crosshair position.

CROSSHAIR — Activates the horizontal and vertical crosshairs and prints the point number and temperature values at the crosshair position.

VCUR — Returns ths velocity of the current vertical crosshair position.

EXAMPLE

You want to expand the y-axis of the SHOW display between two temperatures designated by the horizontal crosshairs. A procedure for doing so is defined below:

```
PROCEDURE YSET
X1 = TCUR; X2 = TCUR;  X1  X2  RANGE
RETURN
FINISH
```

TELL

TELL is an informational verb requiring an object. The four permitted objects of TELL are DISK, CSTACK, GPARMS and KSCANS. TELL DISK lists the scans in the disk index; TELL CSTACK prints the stack counter and the numbers of the scans that have been accumulated in the HOLD array; TELL GPARMS prints the current gaussian fit parameters; TELL KSCANS lists the scans in the KEEP file.

PTWH — Is not used or changed.

ADVERBS — None.

OBJECTS

DISK — Is used to request a listing of the scan in the disk index.

CSTACK — Is used to request a listing of the scans that have been accumulated. Only the numbers of the first thirty-four and the last scans that were accumulated will be printed.

GPARMS — Is used to print the current gaussian fit parameters in the format used by GMODEL. Used as an alternative to PRINT CENTER, etc. where the output is the sample number.

KSCANS — Is used to request a listing of the scans in the KEEP file.

RELATED VERBS

CRT — Used to direct the output of TELL DISK to the terminal screen.

PRINTER — Used to direct the output of TELL DISK to the hard copy device.

-continued-

The output of TELL DISK looks like:

```
TELL DISK
16370  16380  16390  16400  16410  16420  16430  16440  16450  16460
16470  16480  16490  16500  16510  16520  16530  16540  16550  16560
16570  16580  16590  16600  16610  16620  16630  16640  16650  16660
16670  16680  16690  16700  16710  16720  16730  16740  16750  16760
16770  16780  16790  16800  16810  16820  16830  16840  16850  16860
16870  16880  16890  16900  16910  16920  16930  16940  16950  16960
16970  16980  16990  17000
```

After KEEPing 3 scans to disk, the output of TELL KSCANS is:

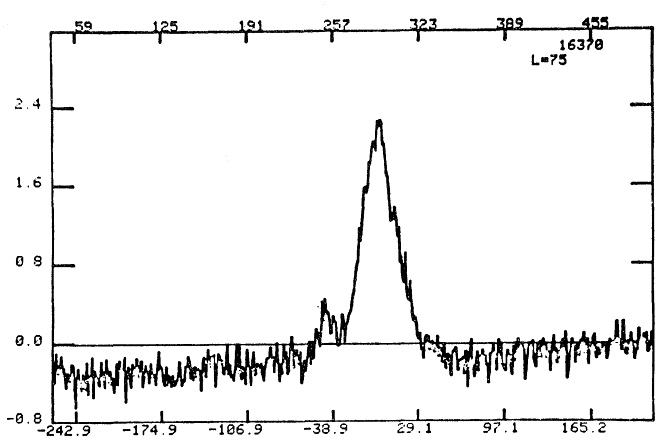
```
> TELL KSCANS
 16370  16620  16830
```

After accumulating 6 scans, the output of TELL CSTACK is:

```
> TELL CSTACK

  6 SCAN(S) HAVE BEEN ACCUMULATED
 16370  16380  16390  16400  16410  16420
```

After using GAUSS to fit a gaussian to the scan shown below, the output of TELL GPARMS is:

```
  TELL GPARMS
       0.322    9.15  -44.22
       2.143   27.77   -2.44
ESTIMATED RMS OF FIT:      0.125
>.
```



TEMP

TEMP computes ((ON - OFF) / OFF) * T_{sys}

where ON = TPOWER onscan stored in the WORK array
OFF = TPOWER offscan stored in the TEMP array.

and puts the result in the WORK array. TEMP is used to combine the bandpasses obtained by total power observing methods into a temperature spectrum.

PTWH Is not used, but is set to the WORK array.

ADVERBS None.

RELATED VERBS

CFETCH	Puts the current scan in the WORK array and the associated offscan in the TEMP array.
FETCH scan#	Puts the indicated scan in the WORK array and the associated offscan in the TEMP array.
ON scan#	Puts the indicated scan in the WORK array.
OFF scan#	Puts the indicated scan in the TEMP array.

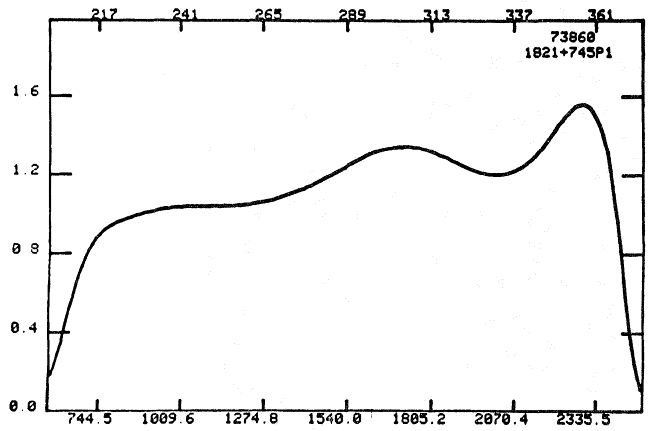
REMARKS

The temperature spectrum will be computed for any or all receivers, depending on the processing option of the onscan.

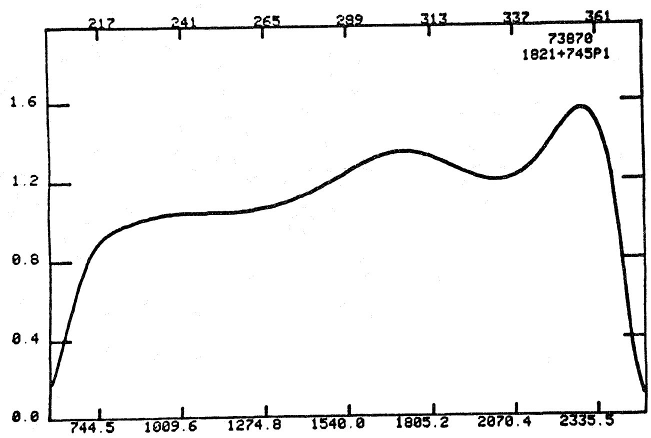
-continued-

EXAMPLE

You have a TPOWER onscan in the WORK array:



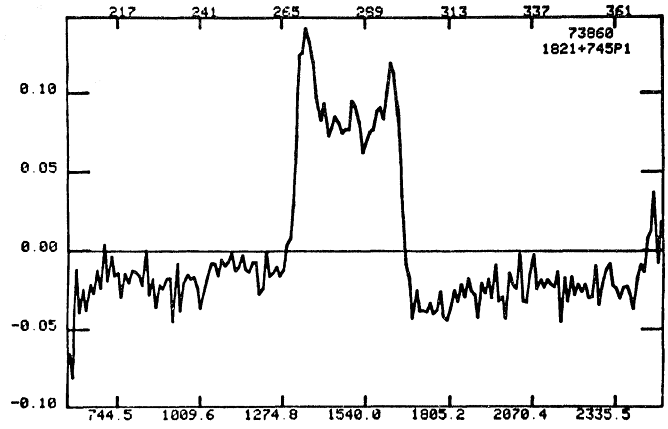
And a TPOWER offscan in the TEMP array:



You specify

TEMP

and get a temperature spectrum in the WORK array:



TRH TRT TRW

These routines copy the contents of the currently referenced array to the indicated array:

TRH copies to the HOLD array

TRT copies to the TEMP array

TRW copies to the WORK array

PTWH — Determines which scan is transferred. Is changed to the indicated array:

TRH sets PTWH to the HOLD array

TRT sets PTWH to the TEMP array

TRW sets PTWH to the WORK array

ADVERBS — None.

OBJECTS — None.

RELATED VERBS — None.

VC CV

A routine which sets the axis labeling mode so that SHOW labels the upper x-axis in channel numbers and the lower x-axis in velocity.

PTWH	Is not used nor changed.
ADVERBS	None.

RELATED VERBS

FV or VF	Sets the axis labeling mode for the velocity on the upper x-axis and frequency on the lower x-axis.

REMARKS

The default axis labeling for SHOW is channel number for the upper axis and velocity for the lower axis. This mode remains in effect until changed by FV or VF.

VCUR

This routine activates the vertical crosshair and retyrns the velocity for that crosshair position of the current display when any key except RETURN is struck.

PTWH Is neither used nor changed.

ADVERBS None.

RELATED VERBS

CCUR Returns the sample number at the current vertical crosshair position.

CROSSHAIR Activates the horizontal and vertical crosshairs and prints the channel number, temperature, and velocity values at the crosshair position.

TCUR Returns the temperature value at the current horizontal crosshair position.

EXAMPLE

You have displayed a scan with velocity on the bottom axis. You wish to spread the display for a certain velocity range specified by the vertical crosshairs. A procedure for doing so is defined below:

```
PROCEDURE XSET
X1 = VCUR; X2 = VCUR; X1 X2 SPREAD
RETURN
FINISH
```

WHILE

WHILE begins a logical construction of the form

```
WHILE (test condition)
     (statements to be iterated)
END
```

WHILE can be used only in procedures. The WHILE logical construction is a conditional looping device similar to the DO WHILE looping facility in PL/I.

(test condition) can be any expression which has a single true or false result. As long as (test conditions) is true, the (statements to be iterated) will be repeated. The (statements to be iterated) may be any group of statements. END is required to complete the construction. WHILE loops may be nested and arranged on one or more lines; if arranged on one line, the statements to be iterated must be separated from the logical statements by semicolons.

<u>PTWH</u> Is not used or changed.

<u>ADVERBS</u> None.

<u>RELATED VERBS</u>

END Completes a logical construction.

FOR Begins a logical construction of the form:

```
FOR(VARIABLE) = (ALPHA) TO (BETA) BY (GAMMA)
     (statements to be repeated)
END
```

IF Begins a logical construction of the form:

```
IF(test condition)
     THEN(do this)
     ELSE(do this)
END
```

See PROCEDURE

-continued-

EXAMPLE

(1) WHILE construction on several lines in a procedure

```
PROCEDURE EG(X,Y)
WHILE X Y
      Y=X*Y/2+1
      PRINT X,Y
      END
FINISH
```

(2) WHILE construction on one line in a procedure

```
PROCEDURE EGALT
WHILE X Y; Y=X*Y/2+1; PRINT X,Y; END
FINISH
```

(3) WHILE constructions nested in a procedure

```
PROCEDURE EG2(X,Y)
WHILE X Y
      WHILE Y  SQRT(2)
      X=X+SQRT(2)
      END
END
FINISH
```

WIPE

This routine is provided for wiping out scans on the analysis disk that are filed under a specified user number. References to scans having the specified user number are deleted from the scan directory but no disk space is actually freed. After invoking WIPE, the data disk is searched to provide a list of user numbers represented on disk. You are then asked to specify, one at a time, the user numbers of the scans to be wiped.

PTWH Not used or changed.

ADVERBS None.

RELATED VERBS None.

REMARKS

WIPE is useful for speeding up scan access if the scan directory is quite long.

WIPE is especially useful in cases where another observer has scans with the same scan numbers as yours, preventing you from seeing your data.

EXAMPLE

A sample use of WIPE is illustrated:

```
>WIPE

  .THE FOLLOWING USER NUMBERS APPEAR
   IN THE DISK INDEX.
 915
 533
 198
 999
  YOU MAY DELETE ONE USER NUMBER AT A TIME
  FROM THE INDEX  ENTER THE VICTIM'S FULL
  THREE-DIGIT USER NUMBER. WITH LEADING
  ZEROS IF NECESSARY. TO EXIT FROM THIS
  ROUTINE. ENTER THE NUMBER 0.
999

ANOTHER USER NUMBER? (ENTER 0 IF FINISHED)
0
>WIPE

   THE FOLLOWING USER NUMBERS APPEAR
   IN THE DISK INDEX.
 915
 533
 198
  YOU MAY DELETE ONE USER NUMBER AT A TIME
  FROM THE INDEX. ENTER THE VICTIM'S FULL
  THREE-DIGIT USER NUMBER. WITH LEADING
  ZEROS IF NECESSARY. TO EXIT FROM THIS
  ROUTINE. ENTER THE NUMBER 0.
0
>
```

*

This operator, when typed immediately after a caret (>) or colon (:) indicates to the program that the current line is a comment. The program does not interpret the line.

PTWH Is not used or changed.

ADVERBS None

OBJECT

comment field The comment field may contain any remarks or notes desired, but may not exceed the length of the line.

RELATED VERBS None

ERRORS

If the asterisk appears in any column except the first, it is interpreted as a multiplication operator.

EXAMPLES

To include a comment on a display of reduced data, use the *:

```
* STACK OF SCANS  500-520,  BBASE = 12,  EBASE = 12,  NFIT = 2
```

?

An alias for PRINT. ? field prints the field requested on the CRT screen. field may be one or more variable or adverb names, literals, or arithmetic expressions. If field is an adverb or variable (a variable is an array or scalar defined in a procedure) name, the value(s) of the adverb or variable will be printed. Arrays of more than one dimension are printed with the first index varying most rapidly, as in Fortran. If field is a literal, it must be enclosed in quotation marks, and it will be printed without the quotation marks. If field is an arithmetic expression, the result of the expression will be printed.

PTWH — Is not used or changed.

ADVERBS — None.

OBJECTS

field — As described above.

RELATED VERBS

READ field — Reads the values requested by field from the CRT. In this case field may be one or more variable or adverb names, and may not include expressions or literals.

PRINT field
field ? — Same as ? field .

PRINTER — Directs the PRINT output to the hard copy device.

CRT — Directs the PRINT output to the CRT screen. CRT is the default assignment, but must be invoked after PRINTER to return output to the CRT.

EXAMPLE

See PRINT

Appendix A.2

A D V E R B

D I C T I O N A R Y

ACOUNT

A counter for the ASTACK adverb. Determines how many scans are in the STACK. Is used by STACK verb, and in procedures.

<u>ACOUNT is modified by</u>

A ADD DELETE EMPTY

<u>The initial value of ACOUNT is 0.</u>

<u>REMARKS</u>

When ACOUNT = 0, the stack is empty.

ASHIFT

The number of channels by which a spectrum must be shifted sto that when it is accumulated, the points at the same velocities or frequencies are added together.

ASHIFT is used by

ACCUM SUM

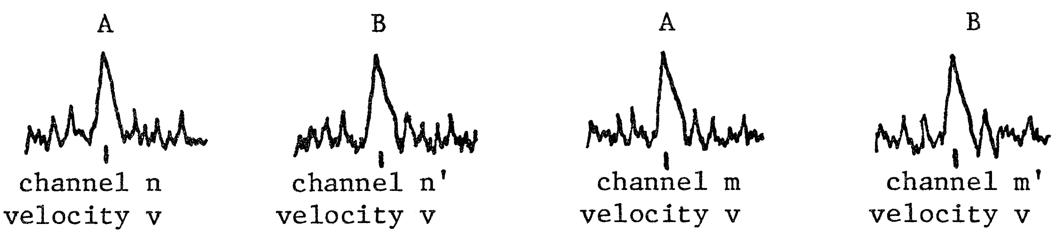
The initial values of ASHIFT(rcvr) are 0.

The user may set the value of ASHIFT as required. ASHIFT may be set by using the verb ALIGN. ASHIFT may be specified separately for each receiver.

The values of ASHIFT are determined by

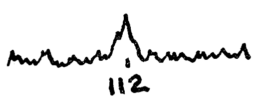
WORK array HOLD array

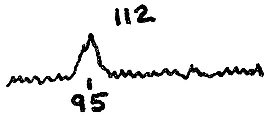
ASHIFT(1)=n-m
ASHIFT(2)=n'-m'



EXAMPLES

Your data was taken at two different LO settings.

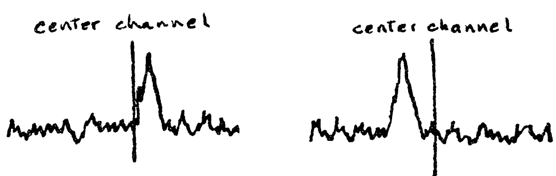
In the HOLD array you have: 

In the WORK array you have: 

To add the contents of the WORK array to the contents of the HOLD array, with velocities matching, specify

ASHIFT = 17
ACCUM

You want to accumulate these two scans:
To accumulate them with the spectral feature centered in the receiver channels, specify



ALIGN

before each ACCUM.

ASTACK

The STACK that is listed by STACK and used in procedures to operate on up to 100 scans.

ASTACK is modified by

A ADD DELETE EMPTY

The initial values of ASTACK are 0.

REMARKS

When ASTACK = 0, the stack is empty.

EXAMPLE

EMPTY	Assures that the stack is empty.
600 614 ADD	Inserts scans into the STACK.

STACK will produce the following output on the terminal:

```
 15
600.  601.  602.  603.  604.  605.  606.  607.  608.  609.
610.  611.  612.  613.  614.
```

BBASE EBASE

The number of points at the beginning and end of the data to be used for computation of a baseline. BBASE and EBASE are arrays, and may be set independently for each receiver. BBASE is the number of points at the beginning of the data; EBASE is the number of points at the end of the data. BBASE(CH) and EBASE(CH) will not be used if NREGION(CH,1) is different from zero.

BBASE and EBASE are used by

BASELINE BMODEL BSHAPE DCBASE RMS

The initial values of BBASE(CH) and EBASE(CH) are 50.

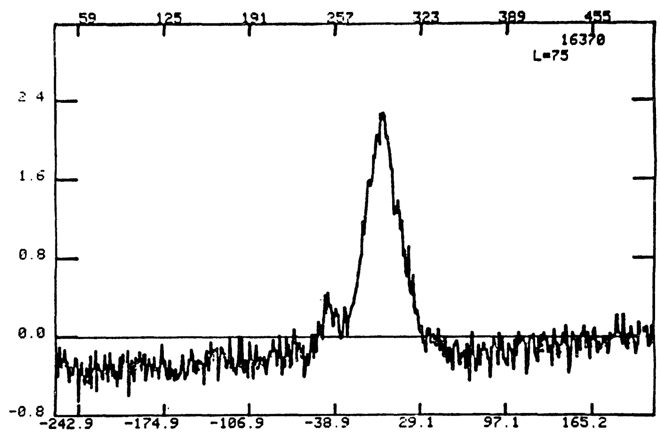
REMARKS

The values can be changed by the user as needed. No verbs change the values of BBASE and EBASE. BBASE and EBASE can be specified separately for each receiver.

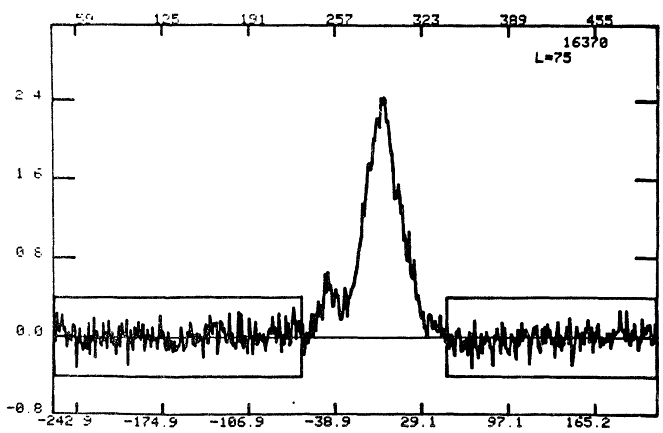
EXAMPLE

Your data look like the display to the right, and you want to fit a baseline in the windows from channels 40 to 190 and from 342 to 502. Specify:

```
BDROP = 40; EDROP=10;
BBASE = 190; EBASE = 160
BMARK = 1;
BASELINE PAGE SHOW
```



to remove a baseline fitted to the desired regions.



BDROP EDROP

The number of points at the beginning and end of the data which will be ignored by the routines listed below. BDROP and EDROP are arrays, and may be set independently for each receiver.

BDROP and EDROP are used by

SHOW	RESHOW	MAPSHOW	LABEL	GMODEL	CROSSFCN
BMODEL	BASELINE	BSHAPE	DCBASE	INVERT	GAINCRV
BOXCAR	HANNING	SMOOTH	RESIDUAL	BMSWITCH	

The initial values of BDROP(CH) and EDROP(CH) are 10.

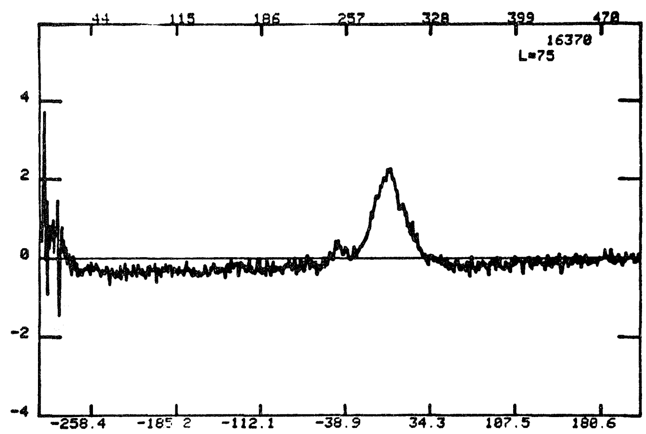
REMARKS

The values can be changed by the user as needed. No verbs change the values of BDROP and EDROP. BDROP and EDROP can be specified separately for the different receiver channels.

EXAMPLE

Your data look like the display to the right. To get rid of the spikes at the left of the scan, drop some channels from the beginning of the scan:

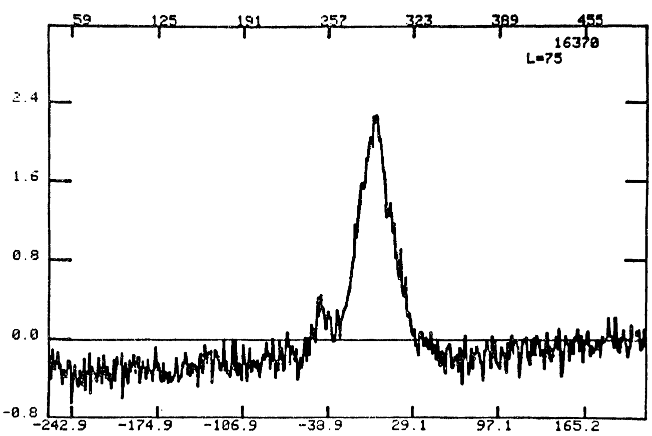
BDROP = 40



A new plot, produced with:

PAGE SHOW

will then show the scan without the spikes at the beginning.



BGAUSS EGAUSS

The explicit sample numbers between which a gaussian is fit by GAUSS.

BGAUSS and EGAUSS are used by

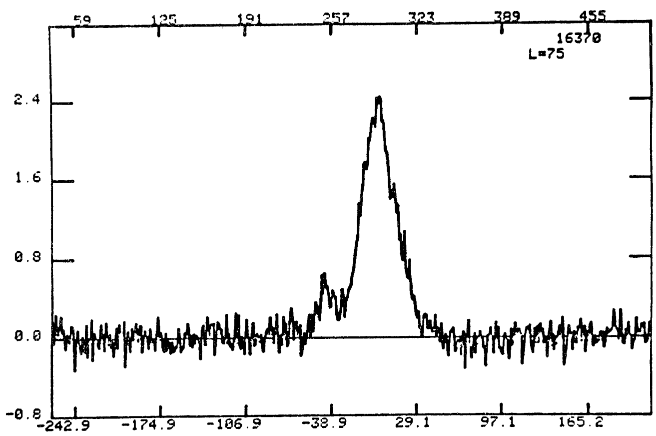
GAUSS GPARTS RESIDUAL

The initial values of BGAUSS and EGAUSS are 1 and 256.

REMARKS

PEAK will set the values of BGAUSS and EGAUSS (see PEAK) or the user may specify their values before using GAUSS or GMODEL.

EXAMPLE



Your data look like the display on the right, and you want to fit a gaussian between channels 230 and 325. Specify:

```
NGAUSS = 2;
BGAUSS = 230; EGAUSS = 325
CENTER(1) = 251
CENTER(2) = 294
HWIDTH(1) = 14
HWIDTH(2) = 33
GAUSS
```

If the fit were successful you get a caret prompt. To print the parameters of the fit, use TELL GPARMS

```
>.TELL GPARMS
      0.474    15.09   -44.95
      2.237    32.73   -2.02
ESTIMATED RMS OF FIT:        0.118
>.
```

in channel numbers and Kelvins.

BMARK

This adverb acts as a switch. If BMARK = 1, the verb SHOW will draw boxes indicating the baseline fitting regions last used or the regions in which an RMS was last computed. If BMARK = 0, the boxes will not be displayed.

BMARK is used by

SHOW

The initial value of BMARK is 0.

REMARKS

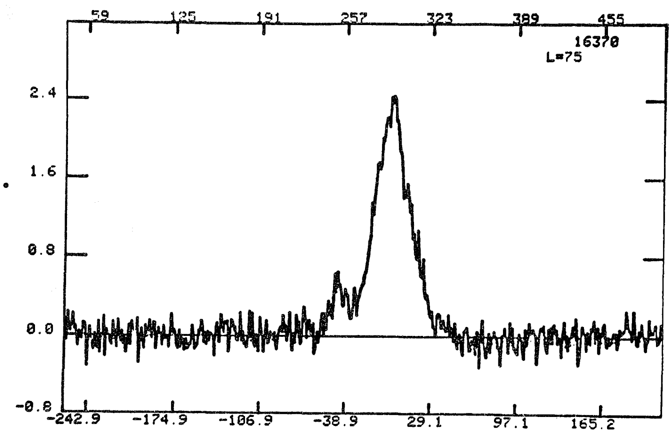
The user can change BMARK as desired. No verbs alter the value of BMARK. BMARK cannot be specified separately for each receiver.

EXAMPLE

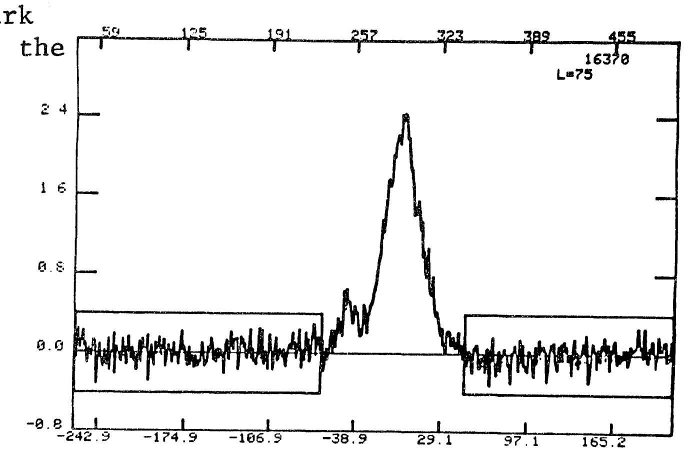
If BMARK = 0, then:

BASELINE PAGE SHOW

produces the display at right.



If BMARK = 1, however, the result of the same commands is shown at the right, where the boxes mark the regions used to calculate the baseline (or RMS).



BMOMENT EMOMENT

These adverbs specify the region of a scan which is used to compute a moment. BMOMENT is the first point of the region and EMOMENT is the last point of the region. BMOMENT and EMOMENT are arrays, and may be set independently for each receiver.

BMOMENT and EMOMENT are used by

MOMENT

The initial values of BMOMENT(CH) and EMOMENT(CH) are 0.

REMARKS

The user must specify values for BMOMENT and EMOMENT before using MOMENT. These adverbs are not changed by any verbs and may be specified separately for the different receiver channels.

BRANGE ERANGE

These are the minimum and maximum right ascensions (Epoch 1950.0, in hours) of scans to be loaded from a KEEP tape.

BRANGE and ERANGE are used by

LOAD

The initial values of BRANGE and ERANGE are -1.

REMARKS

If BRANGE or ERANGE is set less than zero, then no check is made on the corresponding RA limit. BRANGE must be less than ERANGE. To LOAD scans in the RA range from 20 hours to 4 hours, first LOAD 20 to 24 hours and then LOAD zero to 4 hours.

EXAMPLE

On the Charlottesville MODCOMP, you wish to read from your keep-tape all scans between a RA of 09h and 15h 30m. Specify:

```
BRANGE = 9.
ERANGE = 15.5
LOAD
```

On the IBM, you are prompted for these parameters by the LOADTAPE program.

BSCAN ESCAN

These are the mimimum and maximum scan numbers to be loaded from a KEEP tape.

BSCAN and ESCAN are used by

LOAD

The initial values of BSCAN and ESCAN are -1.

REMARKS

If BSCAN or ESCAN are set less than zero, then no check is made on the corresponding scan numbers.

EXAMPLE

On the Charlottesville MODCOMP, you wish to read from your KEEP tape all scans between number 29300 and 29900. Specify:

```
BSCAN = 29300.
ESCAN = 29900.
LOAD
```

On the IBM, you are prompted for these parameters by the LOADTAPE program.

CENTER

An array of length six containing the sample numbers of the centers of up to six gaussians.

CENTER is used by

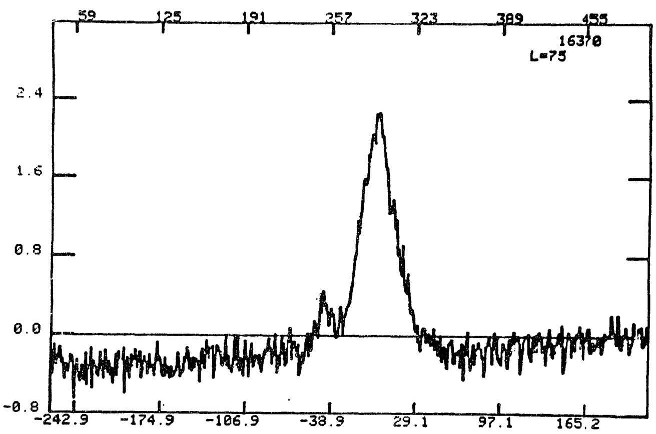
GAUSS GMODEL GPARTS RESIDUAL REGAUSS

The initial values of CENTER(N) are 0.

REMARKS

The user must specify NGAUSS values of CENTER before calling GAUSS. These values will be iterated by the routine GAUSS unless the adverb FIXC is less than zero (see FIXC). The values of CENTER (usually refined by GAUSS) are used by GMODEL, GPARTS, and RESIDUAL. PEAK will find the approximate center of a single gaussian.

EXAMPLE



Your data look like the display on the right. Using CROSSHAIR, you have determined that the center of the desired gaussian is near channel 294. Specify:

NGAUSS = 1
CENTER = 294

Before using GAUSS, however, you must also specify HWIDTH, and may have to change BGAUSS and EGAUSS.

CLIPMAX CLIPMIN

These adverbs are the maximum and minimum temperatures (in Kelvin) that will be found in the data after it is CLIPped. CLIPMIN and CLIPMAX are arrays, and may be set independently for each receiver.

CLIPMAX and CLIPMIN are used by

CLIP

The initial values of CLIPMAX(CH) and CLIPMIN(CH) are 99999.9 and -99999.9 respectively.

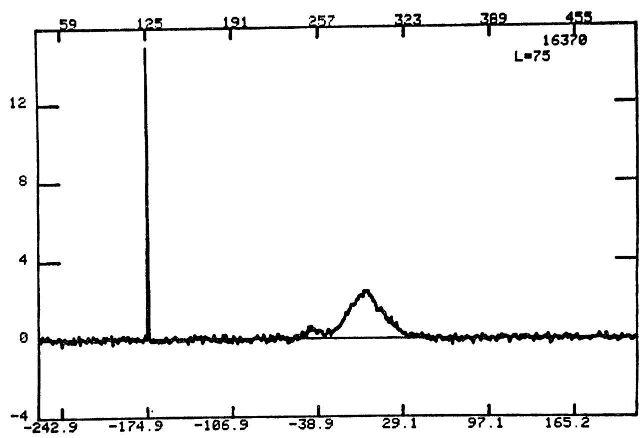
REMARKS

The user may specify any values for CLIPMAX and CLIPMIN. No verbs change the values of CLIPMAX and CLIPMIN. CLIPMAX and CLIPMIN may be specified separately for the different receiver channels.

EXAMPLE

Your data look like the display to the right. The interference spike at the left has overwhelmed the y-axis scaling chosen by SHOW. To eliminate this problem, specify:

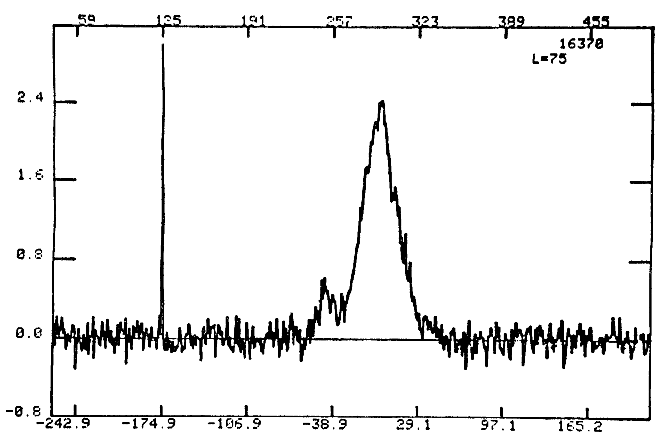
```
CLIPMIN = -1; CLIPMAX = 3
CLIP
```



After CLIPping, specify:

```
PAGE SHOW
```

to produce the display at right.



CMARK

An array of length six containing the sample numbers at which to draw a vertical line when a scan is displayed. Up to six different points may be specified.

CMARK is used by

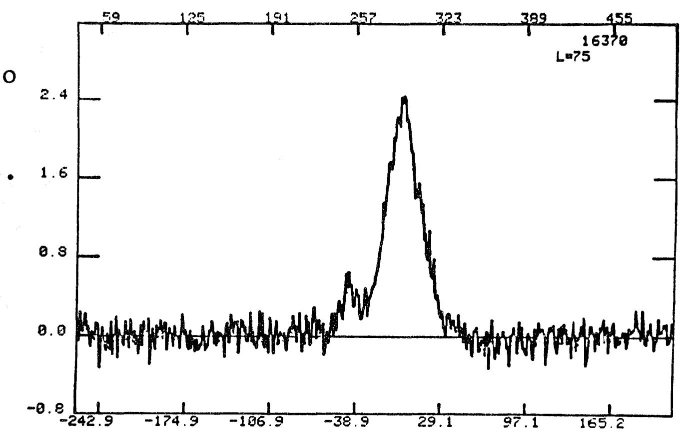
SHOW

The initial values of CMARK(N) are 0.

REMARKS

If CMARK(N) is less than or equal to zero, no vertical line will be drawn. The user may change the values of CMARK. No verbs change CMARK.

EXAMPLE



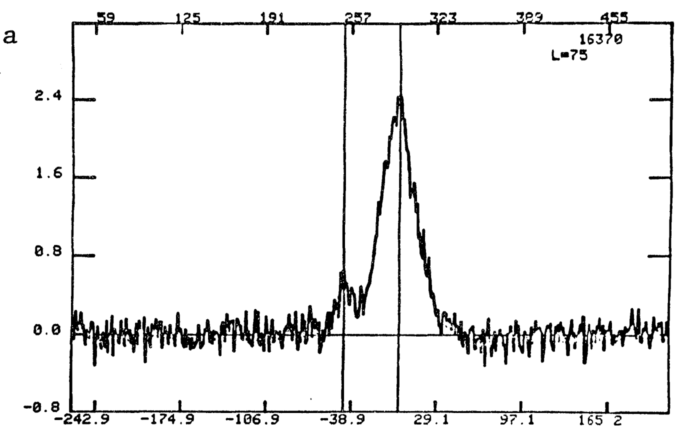
Your data look like the display to the right. You want to draw a vertical line down the peaks, which are at channels 251 and 294. Specify

```
CMARK(1) = 294
CMARK(2) = 251
```

or

```
CMARK = 294, 251
```

(the comma is necessary). Then a subsequent SHOW will produce the display to the right.



DF

The amount of shift between the signal and reference spectra in MHz for frequency switched data.

DF is used by

RAPL RAPR

The initial value of DF is 0.

EXAMPLE

See RAPL and RAPR

FACT

An array of numbers supplied by the user which may be used:

(1) by SCALE to multiply the data points for each receiver by a constant, or

(2) by BIAS to add a constant to the data points for each receiver.

FACT is used by

SCALE BIAS

The initial values of FACT(CH) are 1.0.

REMARKS

FACT may be set by the user. It is not changed by any verbs. It may be specified separately for the different receiver channels.

EXAMPLES

(1) You want to multiply a scan by .001
so you specify:

```
FACT = .001
SCALE
```

(2) You want to subtract ½°K from a scan
so you specify:

```
FACT = -0.5
BIAS
```

FIXC FIXHW

These adverbs are used as switches. If FIXC or FIXHW is greater than or equal to 1, then the gaussian routines will not iterate the user's guesses for the centers or the halfwidths of the gaussians, respectively.

FIXC and FIXHW are used by

GAUSS REGAUSS

The initial values of FIXC and FIXHW are 1.

REMARKS

For these default values, GAUSS will not iterate the values of CENTER and HWIDTH specified by the user. For values of FIXC or FIXHW less than zero, GAUSS will iterate the CENTER or HWIDTH values. FIXC and FIXHW may be changed by the user and are not set by any verb.

EXAMPLE

If you want to fit a gaussian with center at channel 294, specify:

```
CENTER = 294
FIXC   = -1
```

(FIXC = -1 will refine all the center values if you are fitting more than one gaussian.)

GREGION

GREGION is a two-dimensional array containing pairs of beginning and ending channel numbers for up to four data windows for each receiver. The first array index is the receiver number for the scan in the currently referenced array. The second array index labels the list of beginning and ending window channels, and runs from 1 to 8. These data windows are used for gaussian fitting.

GREGION is used by

GAUSS REGAUSS

The initial values of GREGION(rcvr,I) are 0.

GREGION has two dimensions so that gaussian fitting windows may be specified independently for each receiver. GREGION is not set by any verbs.

Beginning and ending channel numbers must be assigned consecutively along the second dimension of the array. If the second index to GREGION is odd, the value of that element is a beginning channel of a region. If the second index is even, the value of that element is an ending channel of a region, which must be larger than the beginning channel. If the begining channel of the first pair for a given receiver is zero, then the routines mentioned above will look at BGAUSS and EGAUSS, rather than GREGION, to define gaussian fitting windows.

EXAMPLE

See GAUSS.

HEIGHT

An array of length six containing the heights (in Kelvins) of up to six gaussians. First guesses for the values of HEIGHT are determined by GAUSS. Therefore, HEIGHT need not be specified before GAUSS is called. The values of HEIGHT are used by GMODEL, GPARTS, and RESIDUAL.

HEIGHT is used by

GMODEL GPARTS RESIDUAL REGAUSS

The initial values of HEIGHT(N) are 0.

REMARKS

The values of HEIGHT are set each time GAUSS is called. If GAUSS is not called before GMODEL, GPARTS or RESIDUAL, NGAUSS values of HEIGHT should be set by the user. PEAK will set HEIGHT to the highest data value in the scan, excluding points dropped from the ends.

EXAMPLE

To fit one gaussian, you may specify:

```
CENTER = 294
HWIDTH = 33
NGAUSS =  1
```

The value of HEIGHT will be calculated by GAUSS. After GAUSS, the calculated value of HEIGHT may be printed by

```
PRINT HEIGHT(1)
```

or with TELL GPARMS.

HWIDTH

An array of length six containing the widths at half the peak height (in number of points) of up to six gaussians.

HWIDTH is used by

GAUSS GMODEL GPARTS RESIDUAL REGAUSS

The initial values of HWIDTH(N) are 0.

REMARKS

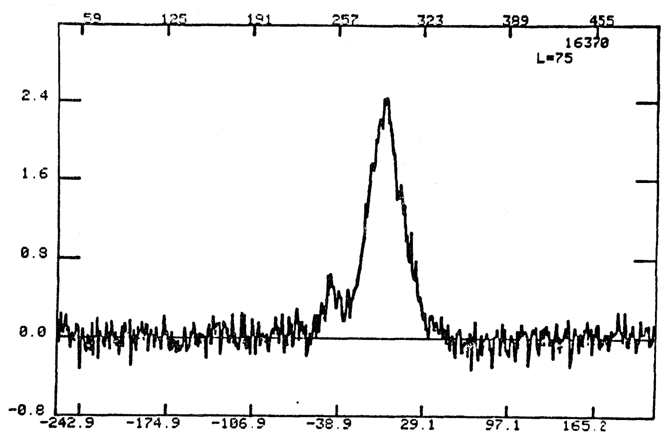
The user must specify the values of HWIDTH before calling GAUSS. The values of HWIDTH will be iterated by GAUSS unless the value of the adverb FIXHW is less than zero. (See FIXC FIXHW.) The values of HWIDTH (usually refined by GAUSS) are used by GMODEL, GPARTS, and RESIDUAL. PEAK will set the approximate HWIDTH of a single gaussian.

EXAMPLE

Your data look like the display to the right. From the display, you determine that the half-width of the larger peak is near 33 channels. Specify:

```
HWIDTH(1) = 33
```

Also specify the CENTER before using GAUSS to refine your guesses.



LEVS

The temperatures (in degrees Kelvin) at which a contour is to be plotted by MAPSHOW. Up to twenty contour levels (in monotonically increasing order) may be specified.

LEVS is used by

MAPSHOW LABEL

The initial values of LEVS (M) are -1 E+5

REMARKS

The user must specify the levels he wants drawn before he invokes MAPSHOW. No verb changes the values of LEVS. The first contour level beyond the highest level you wish to plot, should be less than -999.

EXAMPLE

You want to get a contour map with contours at every half degree from 1 to 3,

You specify

```
LEVS(1) = 1
LEVS(2) = 1.5
LEVS(3) = 2
LEVS(4) = 2.5
LEVS(5) = 3
LEVS(6) = -1000
```

or more conveniently specify

```
LEVS = 1, 1.5, 2, 2.5, 3, -1000
```

(the commas are necessary).

MRATIO

The ratio between x and y scaling of contour maps.

MRATIO is used by

MAPSHOW LABEL

The initial value of MRATIO is 1.

REMARKS

The size of a contour map along the x-axis is fixed. The size in the y-axis direction is determined with respect to the x-axis. If your map is too large in the y-axis direction, make MRATIO smaller. No verbs change MRATIO.

NBOX

An odd number of points to be averaged together by the BOXCAR baseline removal routine, or sorted by the MDBASE baseline removal routine (Green Bank only).

NBOX is used by

BOXCAR MDBASE

The initial value of NBOX is 3.

REMARKS

The user may change NBOX to any other positive odd integer. Attempting to use BOXCAR or MDBASE when NBOX is even will result in an error message. No verbs set NBOX and it cannot be specified separately for each receiver.

EXAMPLE

You want to average seven points together for the purpose of smoothing your data. Before you BOXCAR, specify:

```
NBOX = 7.
```

NFIT

The order of the polynomial baseline to be computed.

NFIT is used by

BASELINE BMODEL BSHAPE

The initial value of NFIT is 1.

REMARKS

The user may specify any other value for NFIT between 1 and 12. NFIT is not changed by any verbs and cannot be specified separately for each receiver.

EXAMPLE

You want to remove a cubic baseline from your data; before you use a polynomial baseline routine, specify:

```
NFIT = 3.
```

NGAUSS

The number of gaussians to be fit by GAUSS or evaluated by GMODEL, GPARTS, or RESIDUAL.

NGAUSS is used by

GAUSS GMODEL GPARTS RESIDUAL REGAUSS

The initial value of NGAUSS is 1.

REMARKS

The user may specify a value of NGAUSS between 1 and 6. No verbs change NGAUSS. If GAUSS is to be called next, NGAUSS values of CENTER and HWIDTH should be specified. If GMODEL or RESIDUAL is to be called next, NGAUSS values of CENTER, HWIDTH and HEIGHT, should be specified.

EXAMPLE

To fit or model three gaussians, specify:

```
NGAUSS = 3.
```

NITER

The number of iterations which a gaussian fitting routine will attempt before giving up.

NITER is used by

GAUSS REGAUSS

The initial value of NITER is 8.

REMARKS

The user may change NITER if desired. NITER is not changed by any verbs.

EXAMPLE

Your attempt to fit a gaussian failed although you were certain your guesses for HEIGHT, CENTER and HWIDTH were accurate. Specify:

```
NITER = 12
```

and try again. Use REGAUSS if the values of HEIGHT are reasonable; otherwise, use GAUSS to recompute initial guesses for the heights.

NMOMENT

The order of the moment to be calculated by the verb MOMENT. If NMOMENT = 0, the area under the curve (between BMOMENT and EMOMENT) will be calculated. Higher moments are not currently implemented.

NMOMENT is used by

MOMENT

The initial value of NMOMENT is 0.

REMARKS

No other values of NMOMENT are allowed at this time. No verbs change NMOMENT and it may not be specified separately for each receiver.

EXAMPLE

See MOMENT

NREGION

NREGION is a two-dimensional array containing pairs of beginning and ending sample numbers for up to four data windows for each receiver. The first array index is the receiver number for the scan in the currently referenced array. The second array index labels the list of beginning and ending window samples, and runs from 1 to 8. These data windows are used primarily for baseline fitting.

NREGION is used by

BASELINE	BMODEL	BSHAPE	DCBASE
RMS			

The initial values of NREGION (CH,I) are 0.

REMARKS

NREGION has two dimensions so that baseline fitting windows may be specified independently for each receiver. NREGION is not set by any verbs.

Beginning and ending channel numbers must be assigned consecutively along the second dimension of the array. If the second index to NREGION is odd, the value of that element is a beginning point of a region. If the second index is even, the value of that element is an ending point of a region, which must be larger than the beginning point. If the begining point of the first pair for a given receiver is zero, then the routines mentioned above will look at BBASE and EBASE, rather than NREGION, to define baseline fitting windows.

NSAVE

The disk bin into which a scan is to be SAVEd or from which it is to be RECALLed. NSAVE is the index to a disk file containing copies of the scan registers in memory.

NSAVE is used by

SAVE RECALL

The initial value of NSAVE is 0.

The user must select a save bin between 1 and 80 before using SAVE or RECALL. NSAVE is not changed by any verb.

EXAMPLE

You have reduced scan which you wish to save so that you can show it to your colleague when he returns from coffee. Thirty four is your favorite number. Specify:

```
NSAVE = 34
SAVE
```

When your colleague arrives, you can recall the scan as follows,

```
NSAVE = 34            (in case you had reset its value)
RECALL
```

OSHIFT

The number of channels by which the left and right sides of dual Dicke or overlapped frequency switching data must be adjusted to be correctly combined by the OVERLAP verb.

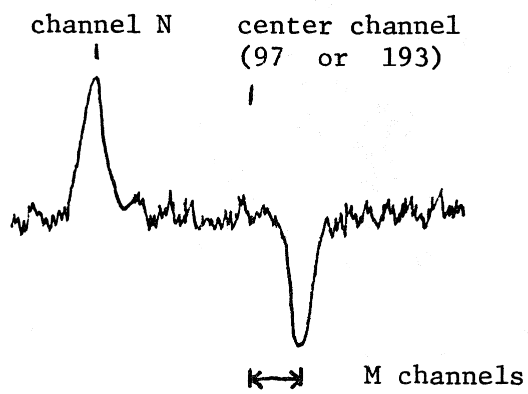
OSHIFT is used by

OVERLAP

The initial value of OSHIFT is 0.

The user may change the value of OSHIFT. If the user does not change the value from 0, then the OVERLAP routine will calculate OSHIFT. OSHIFT may not be specified separately for each receiver (OVERLAP works only on the A receiver, so the value of OSHIFT will be used to OVERLAP the A receiver).

Unless specified, the value of OSHIFT is determined by



OSHIFT = N - M

EXAMPLE

See OVERLAP.

PTWH

The adverb used by the program as a pointer to the "currently referenced array:"

PTWH = 1 points to the TEMP array,
PTWH = 2 points to the WORK array, and
PTWH = 3 points to the HOLD array.

The pointer may be changed by an assignment statement (e.g., PTWH=2), or by using the verbs PMT, PMW, and PMH, which are acronyms for "pointer moves to (array name)."

PTWH is used by

PMT PMW PMH,

and all verbs which use or modify the contents of the currently referenced array. See the verb dictionary for the effect of a verb in question.

The initial value of PTWH is 2.

EXAMPLES

To find out which array is the currently referenced array:

```
PRINT PTWH
2.000
```

In this case the result indicates that the WORK array is the currently referenced array.

RFREQ

User's first guess to frequency of sinusoidal baseline (in MHz).

RFREQ is used by

RIPPLE RMODEL RSHAPE

The initial value of RFREQ is -1.

The user must pick a positive value for RFREQ before using one of the verbs above. The program will try to fit a sinusoidal baseline with a frequency near RSHIFT. It it cannot do it, the error message FIT FAILED will be generated. RFREQ cannot be specified separately for each receiver.

EXAMPLE

See RIPPLE.

RSHIFT

The number of channels by which receiver B is shifted when added to receiver A.

RSHIFT is used by

SLIDE

The initial value of RSHIFT is 0.

The user may change RSHIFT as needed. RSHIFT is not changed by any verbs.

EXAMPLE

If the peak in receiver B was to the right of the peak in receiver a by N channel, you would specify

```
RSHIFT = -N
SLIDE
```

If the peak in receiver B was to the left of the peak in receiver A by N channels, then you would specify

```
RSHIFT = N
SLIDE
```

SET1 SET2 SET3

The values of these adverbs are used in conjunction with the verb SELECT to select processing the receivers.

SET1, SET2, and SET3 are used by

SELECT

The initial valus of SET1, SET2, and SET3 are 1.

The user may set SET1 and/or SET2 and/or SET3 as needed. See the SELECT verb for a description of how the SELECT facility works.

EXAMPLE

See SELECT.

SINCR

SINCR is the increment between scan numbers being put on the stack by the verb ADD. It is meant to be set to the scan number increment between scans from the same receiver. In Green Bank, this increment is always ten.

<u>SINCR is used by</u>

ADD

<u>The initial value of SINCR is 10.0</u>

SIZE

SIZE is an array used by the verb MOMENT to return its results. The array index is the receiver in use.

SIZE is used by

MOMENT

The initial value of SIZE is 0.

REMARKS

After using the verb MOMENT, you must

PRINT SIZE

to see the result of the calculation.

EXAMPLE

See MOMENT.

SMWGT

An array of length twelve containing a template of weights for data smoothing. The verb SMOOTH smooths a scan by convolving the template with the data. The first element of SMWGT is the number of points in the weighting function (not greater than 11). The following elements define the template of weights. The sum of the weights should be one.

SMWGT is used by

SMOOTH

The initial values of SMWGT are 0.

EXAMPLE

To mimick HANNING smoothing, where the template spans three data points, specify:

```
SMWGT( 1 ) = 3
SMWGT( 2 ) = 0.25
SMWGT( 3 ) = 0.50
SMWGT( 4 ) = 0.25
```

or more compactly:

```
SMWGT = 3, 0.25, 0.50, 0.25
```

(where the commas are necessary).

For other examples of smoothing templates, see the verb SMOOTH. To see the effect of one kind of smoothing, see the verb BOXCAR.

TMARK

An array of length six containing the temperatures at which to draw a horizontal line when a scan is displayed. Up to six different temperatures may be specified.

TMARK is used by

SHOW

The initial values of TMARK(N) are -9999

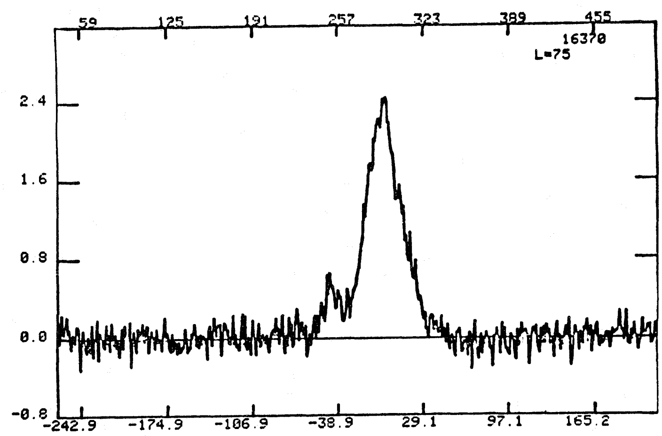
REMARKS

When TMARK is less than zero, no horizontal lines will be drawn. The user may change the values of TMARK. No verbs change TMARK.

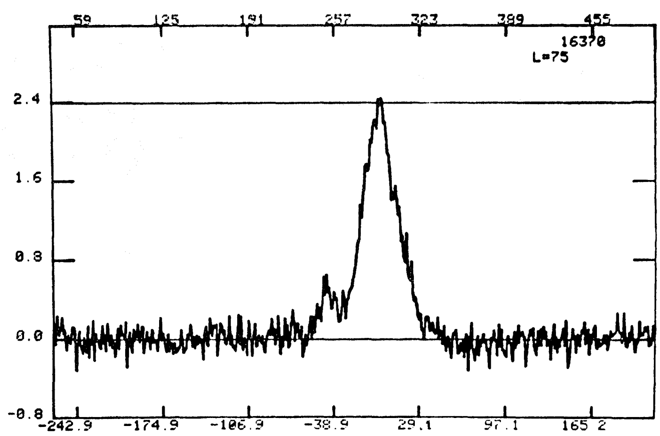
EXAMPLE

Your data look like those to the right. You want to verify the line temperature associated with a given tick mark on the display. Specify:

```
TMARK(1) = 2.4
SHOW
```



Your display will then look like the one to the right.



TWH

TWH is a two dimensional array which constitutes the three scan registers or arrays in memory. The second index runs from one to three, and is the scan pointer. The program uses the pointer PTWH to refer to the currently referenced array. The first index is the real word in the scan. The content of each word in the scan is given in this manual under the heading Disk Format of Continuum Scans. Only the real words in these formats can be accessed; integer and character fields cannot be changed and can be read only to the extent provided for through special routines, like HEADER and SLENGTH.

TWH is used by

All routines that refer to data in scans.

The initial values of TWH are 0.

Attempting to perform an operation requiring data in TWH before any data have been loaded results in the error message:

```
LOAD DATA!
```

REMARKS

Except for the restriction to real words in the disk formats, direct access to header and data values in the TWH array gives you the ability to perform almost any data manipulation.

EXAMPLE

Suppose you are in Green Bank, observing on the 140-foot telescope. You want to scale up your data for atmospheric attenuation, which depends on zenith distance z as

```
exp( tau / cos(z) )
```

where tau is the optical depth at zenith. You find that the zenith distance is in word 39, so the scale factor is:

```
FACT = EXP( TAU/ COS( TWH( 39,PTWH ) ) )
```

VMARK

Velocities at which to draw a vertical line when a scan is displayed. Up to six different velocities may be specified.

VMARK is used by

SHOW

The initial values of VMARK(N) are -9999

REMARKS

When VMARK is less than zero, no vertical lines will be drawn. The user may change the values of VMARK. No verbs change VMARK.

VRMS

VRMS is an array used by the verb RMS to return the rms values. The array index is the receiver in use.

VRMS is used by

RMS

The initial value of VRMS is 0.

REMARKS

The rms results are printed by RMS, but are available for calculations through VRMS.

EXAMPLE

Suppose you are observing with two receivers, and scans 151 and 152 are on receivers one and two. You decide to average the two scans, weighted by the inverse of the square of the rms noise in each scan. First set windows for RMS using NREGION or BBASE and EBASE. The windows may be set independently for the two receivers. Use the verb SUM and the array WEIGHT to take a weighted average:

```
GET 151 RMS; WEIGHT = 1/VRMS( 1 )**2; SUM
GET 152 RMS: WEIGHT = 1/VRMS( 2 )**2; SUM AVE
```

WEIGHT

WEIGHT is an array of weighting values used to accumulate scans using non-standard weighting. The value of WEIGHT multiplies the default weighting of scans, but the CONDAR program default is to weight all scans equally.

WEIGHT is used by

SUM

The initial values of WEIGHT(N) are 0.

EXAMPLE

See SUM.

YINCR YMIN

Are arrays used to set the Y scaling of the data graphs. YINCR equals the number of Kelvins that will appear between each of six y-axis tick mark. YMIN equals the lowest number of Kelvins that will appear on the graph. The array index is the receiver number.

YINCR and YMIN are used by

SHOW RESHOW

The initial values of YINCR (CH) are 0.
The initial values of YMIN (CH) are -99999.9.

REMARKS

If YMIN is less than or equal to -9999, SHOW will automatically set the y-axis scaling. If you set YMIN to greater than -9999, SHOW will use your values of YMIN and YINCR. So if you set YMIN you *must* also set YINCR.

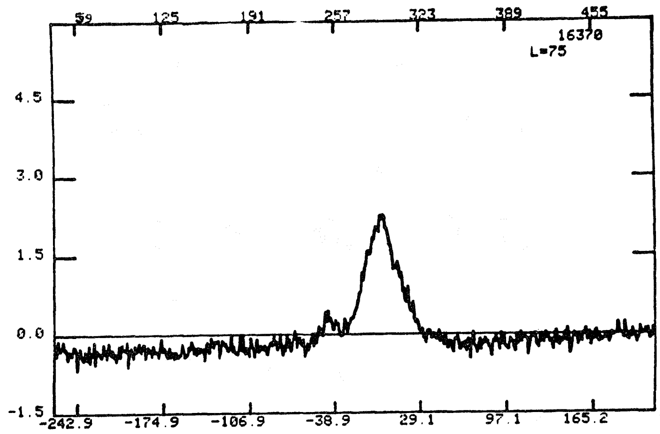
Four verbs set the value of YMIN. FREEY and AUTO set YMIN to greater than -9999, so that SHOW and RESHOW *will* set the y-axis scaling. HOLDY sets YMIN to a large negative number (-9E10) so that SHOW *will not* update its y-axis scaling. RANGE sets the values of YMIN and YINCR according to specified minimum and *maximum* temperatures to be plotted.

If set, you must set *both* YMIN and YINCR. YINCR and YMIN may be specified separately for each receiver.

-continued-

<u>EXAMPLES</u>

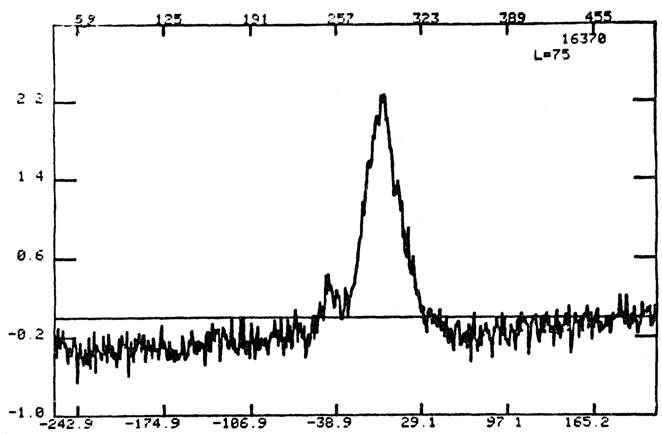
Suppose SHOW produced the top display to the right.



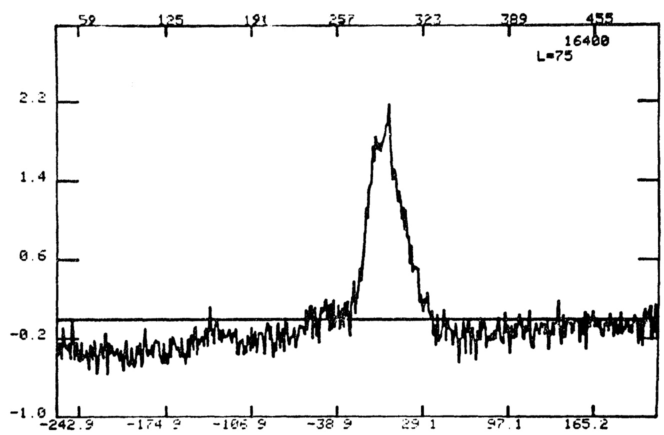
To expand the vertical scale, specify:

    YMIN = -1.0
    YINCR = .8
    PAGE SHOW

to produce the middle display on the right.



Now you want to compare two different scans but you are frustrated because SHOW plots them on different scales. To get the second graph to the same scale, specify HOLDY before you specify SHOW for the second scan to produce the bottom display on the right.



To get back to automatic scaling, specify FREEY or AUTO.

ZLINE

This adverb is used as a switch. If ZLINE = 1, a horizontal line is drawn at zero Kelvins. If ZLINE = 0, the line is not drawn.

ZLINE is used by

SHOW

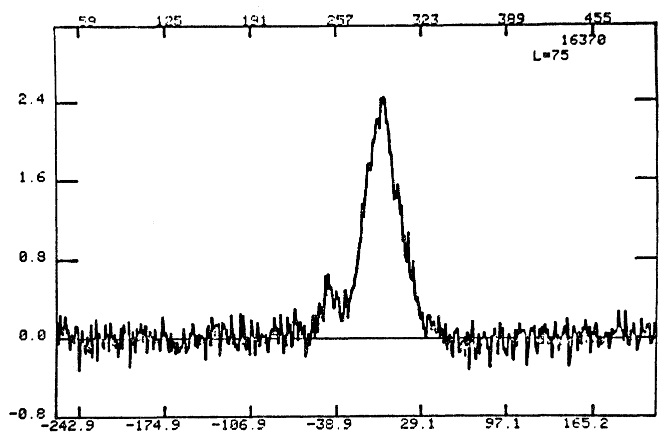
The initial value of ZLINE is 0.

REMARKS

The user may change ZLINE as desired. No verbs change ZLINE.

EXAMPLE

If ZLINE = 1, SHOW will draw the zero line as shown:



Appendix A.3

Enabling the Tektronics 4010 for Graphics

Under our current setup, the graphics teminal (line 023, "Green Screen", room 106) is the only one that requires operator intervention to switch between CMS and OS/MFT. Following are all the possible states that the terminal could be in, and what should be done to get it to where its wanted. First, query the line (QUERY 023 or Q 023) to find out what its STATUS is, and then enter the appropriate COMMAND(s) as shown below.

For CMS Graphics (POPS)

STATUS	COMMAND
LINE 023 ENABLED	None needed.
LINE 023 DISABLED	ENABLE 023 CMS
LINE 023 ATTACHED TO MFTVIRT	ENABLE 023 OK
LINE 023 OFFLINE	VARY ONLINE 023 ENABLE 023 CMS
LOGON023 -023	CMS023

For OS / MFT Graphics

STATUS	COMMAND
LINE 023 ATTACHED TO MFTVIRT	None needed.
LINE 023 DISABLED	DISABLE 023 OK
LINE 023 ENABLED	DISABLE 023 OK
LINE 023 OFFLINE	VARY ONLINE 023 DISABLE 023 OK
LOGON023 -023	MFT023

Appendix A.4

Format of Spectral Line Scans on Disk

Integer ITWH	Real TWH	Double DTWH	Type	Contents	Units
1-2	1		R	Scan Number	
3-4	2		R	Off Scan Number	
5-10			C	Source Name (12)	
11			I	Observer's Number	
12-21			C	Observer's Name	
22-24		8	DR	Julian Date	
25			I	Solar Day of Year	
26			I	Month	
27			I	Day	
28			I	Year	
29-30	15		R	Local Sidereal Time	rad
31-32	16		R	Eastern Standard Time	rad
33			I	Telescope	
34			I	Observing Program	= 1 or 2
35			I	Scan type Code	
36			I	MODE	
37			I	IREC	
38			I	LREC	
39			I	INCR	
40-43			I	ISTART (4)	
44-47			I	ISTOP (4)	
48			I	VREF	
49			I	VDEF	
50			I	Position Code	
51			I	Scan Direction (H-V)	
52			I	Unused	
53-54	27		R	Scan Duration	s
55			I	Status Word 1	
56			I	Status Word 2	
57-58	29		R	Integration Time	s
59-62	30-31		R	Unused	
63-64	32		R	Epoch of Observation	years
65-66	33		R	H Coordinate Rate	arcmin/min
67-68	34		R	V Coordinate Rate	arcmin/min
69-70	35		R	RA, Indicated	rad
71-72	36		R	Dec, Indicated	rad
73-74	37		R	Focus	mm
75-76	38		R	Orientation	rad
77-78	39		R	Zenith Distance	rad
79-84	40-42		R	Descriptive Origin (3)	
85-86	43		R	RA Pointing Correction	rad
87-88	44		R	Dec Pointing Correction	rad

Integer ITWH	Real TWH	Double DTWH	Type	Contents		Units
89-90	45		R	PVLS - P1		rad
91-92	46		R	PVLS - P2		rad
93-94	47		R	PVLS - P3		rad
95-98	48-49		R	Unused		
99-106			I	Environmental Values	(8)	
107-114	54-57		R	RA, Apparent	(4)	rad
115-122	58-61		R	Dec, Apparent	(4)	rad
123-130	62-65		R	RA, Epoch	(4)	rad
131-138	66-69		R	Dec, Epoch	(4)	rad
139-146	70-73		R	Galactic Longitude	(4)	rad
147-154	74-77		R	Galactic Latitude	(4)	rad
155-162	78-81		R	Observed H	(4)	rad
163-170	82-85		R	Observed V	(4)	rad
171-178	86-89		R	RHO, Feed Offset	(4)	rad
179-186	90-93		R	THETA, Feed Offset	(4)	rad
187-194	94-97		R	Noise Tube	(4)	K
195-202	98-101		R	System Temperature	(4)	K
203-210	102-105		R	Center velocity	(4)	km/s
211-218	106-109		R	Velocity spacing	(4)	km/s
219-226	110-113		R	RVSYS	(4)	km/s
227-234	114-117		R	Integration time	(4)	s
235-242	118-121		R	Bandwidth	(4)	MHz
243-250	122-125		R	Theoretical RMS	(4)	K
251-252			R	Unused		
253-264		85-88	DR	Center Frequency	(4)	MHz
265-276		89-92	DR	Rest Frequency	(4)	MHz
277-312			C	Center Freq Formula	(4x18)	
313-315		105	DR	L1		MHz
316-318		106	DR	L1F1		MHz
319-321		107	DR	L1F2		MHz
322-324		108	DR	L2		MHz
325-327		109	DR	L2F1		MHz
328-330		110	DR	L2F2		MHz
331-332	166		R	LA		MHz
333-334	167		R	LB		MHz
335-336	168		R	LC		MHz
337-338	169		R	LD		MHz
339-360			I	A/C Words, RESERVED	(50)	
361-464			I	RESERVED		
465-472	233-236		R	Ref System Temperature	(4)	K
473-504	237-252		R	Power Counters	(16)	
505-512	253-256		R	Channel zero values	(4)	
513-1280	257-640		R	Spectral values	(384, 300-foot)	
513-2560	257-1280		R	Spectral values	(1024, 140-foot)	

INDEX

R

S