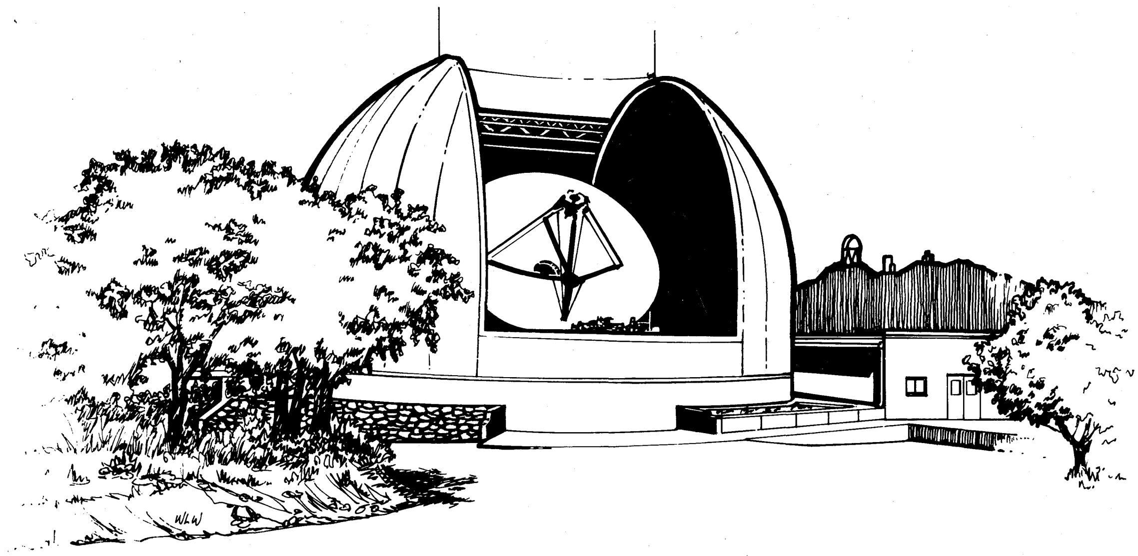


SPECTRAL LINE ANALYSIS SYSTEM

For The NRAO 12m Millimeter-Wave Telescope

KITT PEAK, ARIZONA

AUGUST 1990 EDITION

The National Radio Astronomy Observatory is operated by Associated Universities, Inc., under cooperative agreement with the National Science Foundation

NATIONAL RADIO ASTRONOMY OBSERVATORY

USER'S MANUAL

SPECTRAL LINE ANALYSIS SYSTEM
for the
12-METER TELESCOPE

Revised Edition

July 1990

Printed on Recycled Paper

TABLE OF CONTENTS

SECTION 1 - INTRODUCTION

SECTION 2 - VERB SYNOPSIS

VERBS CATAGORIZED BY FUNCTION:

SECTION 3 - VERB DICTIONARY

SECTION 4 - ADVERB SYNOPSIS

ACOUNT
ASHIFT (RCVR)
ASTACK (100)
BADPT (12)
BBASE (RCVR)
BDROP (RCVR)
BGAUSS
BMARK
BMOMENT (RCVR)
CENTER (6)
CLIPMAX (RCVR)
CLIPMIN (RCVR)
CMARK (6)
DF
EBASE (RCVR)
EDROP (RCVR)
EGAUSS
EMOMENT (RCVR)
FACT (RCVR)
FIXC
FIXHW
FMARK (6)
FR
FS
HEIGHT (6)
HWIDTH (6)
LEVS (20)
MRATIO
NBOX
NFIT
NGAUSS
NITER
NMOMENT
NREGION (RCVR,8)

NSAVE	TMARK (6)
NSIGMA	VMARK (6)
RFREQ	WEIGHT (RCVR)
RSHIFT	YINCR (RCVR)
SHIFT	YMIN (RCVR)
SLABEL	ZLINE
SMWGT (12)	

SECTION 5 - ADVERB DICTIONARY

I. INTRODUCTION

This manual describes the spectral line data reduction program currently running in the VAX 11/750 computer at the 12-m radio telescope. Similar versions of the program operate in Green Bank, West Virginia.

Tom Cram wrote the original version for the 140- and 300-ft telescopes in Green Bank in 1970. The program was based upon the POPS (People-Oriented Parsing Service) language developed by Jerry Hudson. Later Betty Stobie expanded the program and adapted it to the 36-ft (now 12-m) millimeter wave telescope in Tucson. Contributors to this documentation over the years include Tom Cram, Darrel Emerson, Mark Gordon, Kathy Hart, and Betty Stobie.

This analysis system, now 20 years old, consists of a collection of individual "procedures" --- also known as "verbs" --- which can be concatenated together on a single line. POPS parses the string and executes the "verbs" sequentially. The advantages of this system lie in its flexibility; the disadvantages, in its complexity.

This manual consists of a *Quickstart* section to get you up and running, a brief discussion of the principles underlying the system, and a summary and dictionary of the individual verbs. If you need additional help or advice, please ask the telescope operator or call any of the NRAO staff scientists in Tucson.

For this edition, special thanks go to Werner Scharlach who collated information and examples for verbs introduced since the last edition in 1981 and to Jennifer Neighbours who typed the text.

M. A. Gordon, Editor

II. QUICKSTART

Objective	Command	Explanation
Log In	`<CR>`	Type carriage return on any graphics terminal to generate USERNAME prompt on the VAX. If the terminal has already been logged in, type EXIT at the > prompt, then LO at the $ prompt to exit. Then press <CR> to log in.
	`OBS <CR>`	Selects the data reduction area on the VAX.
	`iii <CR>`	Enter your initials or those for whom the operator has established the computer area for your data reduction. This area is [OBS.iii].
	`LINE <CR>`	Selects the program for spectral analysis. You should see the > prompt on the screen. Typing TESTLINE will select the unproved "beta" version.
Plot a Scan	`nnnn F <CR>`	Plots the spectra from filterbank 1 (first 256 channels of Scan nnnn). The number nnnn is an even number.
	`nnnn S <CR>`	Plots the spectra from filterbank 2 (second 256 channels of Scan nnnn). The number nnnn is also an even number)
Plot the average of several scans	`EMPTY <CR>`	Empties STACK, which may contain the numbers of the scans to be averaged. This command prepares the STACK for loading.
	`nnnn mmmm ADD <CR>`	Adds the continguous range of scan numbers from nnnn to mmmm into the stack. (nnnn & mmmm are even numbers)
	`nnnn A; mmmm A <CR>`	Adds the single scans nnnn and mmmm into the stack. (Also, nnnn and mmmm are even numbers)

	STACK <CR>	Lists the numbers of scans in the stack.
	nnnn DELETE <CR>	Deletes the individual scan nnnn.
	C1 <CR>	Averages scans in the stack for filterbank 1 (first 256 channels) weighted inversely by the square of the RMS noise and plots the result.
	C2 <CR>	Averages scans in the stack for filterbank 2 (second 256 channels) weighted as noted above and plots the result.
	CB <CR>	Averages first and second group of channels weighted by system temperature and plots results. Both groups must have same frequency resolution.
Removing a baseline	BSET <CR>	After a spectrum has been plotted, this procedure uses the cursor to select the region to be fitted. For a single region, type 1 in answer to the question, move the crosshair to the beginning of the region and press the space bar, move the crosshair to end of the region and press the space bar. This procedure will assign values to the parameters NREGION(1,i) and NREGION(1,i+1)
	NFIT = n <CR>	Choose the degree of the polynomial to be fitted.
	BASELINE PAGE SHOW <CR>	Calculates and subtracts the polynomial baseline and replots the spectrum.
Smoothing a spectrum	HANNING <CR>	Convolves the spectrum with a Hanning filter. See also the verbs BOXCAR, SMOOTH, and CHGRES.
Fitting a Gaussian	GSET <CR>	Similar to BSET, GSET uses the cursor to select the centers and then the widths of the lines to be fit. Sets the parameters NGAUSS, CENTER(i), HEIGHT(i), and HWIDTH(i). Pressing the space bar enters the value of the cursor position.

	`GSHOW <CR>`	Fits the Gaussian, overlays the fit, and lists the fit parameters in the order height (K), width (km/s), center (km/s) with the RMS error (K) below the line. GSHOW uses the verbs GAUSS and GMODEL.
Clear graphics page	`PAGE <CR>`	Clears the graphics page of the terminal *while in line or testline*. There are also function keys on the Modgraph terminals to do this in or out of Line.
Replot scan	`XX <CR>`	Clears screen and replots whatever is in the scan buffer. XX uses the verbs PAGE and SHOW.
Change ordinate scale	`n.n m.m RANGE <CR>`	Changes the ordinate range of the plot from n.n (bottom limit) to m.m (upper limit) degrees. After doing this, type XX <CR> to replot spectra.
Reset ordinate scaling to automatic	`FREEY <CR>`	Institutes automatic scaling that, like RANGE, remains in effect until changed.
List values	`TABLE <CR>`	Lists temperatures for scan in memory in a raster format.
List header	`HEADER <CR>`	Lists "header" parameters associated with a data scan.
Prints plot	`MAKECOPY <CR>`	Prints displayed scan on the laserprinter.

III. POPS: THE HEART OF THE SYSTEM

POPS (which stands for "People-Oriented Parsing Service") is an interactive, interpretive compiler with a set of built-in functions that may be extended by the programmer before the program is put into operation and, afterwards, by the user. This language can be used as the base for various applications. In addition to supporting spectral line data reduction, POPS is also used to support the CONDAR continuum program and the AIPS image processing system.

The on-site single dish spectral line data reduction program (POPS/Spectral Line) consists of an interactive base program (POPS) and a set of application routines (Spectral Line). These routines are similar to the routines incorporated into the Charlottesville and Green Bank data reduction programs. Although use of POPS/Spectral Line does not depend on an understanding of the distinction between POPS and Spectral Line, they are described separately in the hope that this will make the program easier to understand.

When the user logs into the system, a"prompt" character is printed on the CRT screen of the interactive graphics terminal. When POPS is finished loading and initializing memory, a second prompt character is printed. Once the second prompt character has been printed, the user may enter a command by typing it on the keyboard. (It will be printed on the CRT screen.) The typed message is completed by striking the RETURN key on the keyboard. Once this key has been struck, POPS 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 POPs "mode of operation".

A. MODES OF OPERATION

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

EXECUTE mode is the most often resident mode. 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 made.

EXECUTE mode prompts with a caret (>) and, as its name implies, immediately executes the commands entered by the user. A session with POPS 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, POPS 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, POPS 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 BBASE
#1
#30 40
>
```

(**BMARK** is a single-valued variable, and **BBASE** is a two-value array; both are application variables.) In **INPUT** mode, input is treated as a value to be assigned to a variable. Note that once the argument list is satisfied, POPS 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 POPS 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 similar 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 statement 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.

B. OPERATORS, VARIABLES, CONSTANTS, AND EXPRESSIONS

POPS 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 POPS 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 procedurename. Some errors of these types are checked for by POPS, 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 POPS, 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 procedurenames (user-defined operators). All names, whether operator or variable, should be distinct. POPS 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.

POPS 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`.

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

POPS expressions are of two kinds: arithmetic and logical. Arithmetic expressions make use of the operator

`=   + -   * /   **`

(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 operation 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 POPS 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 whenever 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 POPS statements; the exceptions to this rule are `PROCEDURE` and `ARRAY` declarations, `FINISH`, `EDIT`, `LIST`, AND `ENDEDIT`. Formats for these statements are necessarily rigid.

C. COMMUNICATION WITH POPS

As mentioned in the introduction to POPS, 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 and retyping the message beginning with the last character backspaced over. To backspace strike the RUB OUT key. The response will be a \ (back slash) for each time the key is struck. To see the line typed with the corrections, strike CTRL and R keys simultaneously. A message may also be cancelled by striking the CTRL and R keys simultaneously.

POPS 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 "`RUN POPS`" is typed from the 4010, POPS/Spectral 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, POPS returns to the ready and waiting state. No "initialization" is needed to get POPS' 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 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 POPS operators.

D. OPERATOR SYNOPSIS

INFORMATION EXCHANGE

*	Begins a comment statement.
EXPLAIN	Documents the functions of all programmer defined procedures, *i.e.*, those defined permanently in the program.
HELP	Documents all verbs and lists all procedures, array variables, scalar variables and verbs. Object may be ARRAY, PROCEDURE, SCALAR, VERB or individual verb names.
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 adverb variables named from the CRT.
? <field>	An alias for PRINT.

MEMORY CONTROL

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

`RESTORE page#`	Copies the indicated disk page into the program memory space.

PROCEDURE

DEFINITION

`PROCEDURE procname`	Initiates the definition of a procedure with the given name.
`RETURN`	Is required in procedures called by other procedures.
`FINISH`	Terminates the definition of a procedure.
`INSTALL procname`	Reads procedures stored in text editor files into program memory.
`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.

LOGICAL CONSTRUCTIONS

`FOR x = a TO b BY c`	Begins a logical construction similar the PL/I iterative DO loop.
`IF (test condition)`	Begins a logical construction similar to the PL/I IF-THEN-ELSE.

`WHILE (test condition)`	Begins a logical construction similar to the PL/I `DOWHILE` loop.
`END`	Terminates the logical constructions begun by `FOR`, `IF`, and `WHILE`.

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.

IV. 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 spectral values,
(3) calculation, and
(4) display of information or spectral values.

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

A scan consists of a set of spectral 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. In the 36 foot POPS/Spectral Line Program, each filterbank is written to disk as a separate scan. Even scan numbers are the first filterbanks and the odd scan numbers are the second filterbanks. For example, scan

500 is filterbank one and scan 501 is filterbank two. A filterbank in parallel mode is considered a two-receiver scan where the first 128 channels are receiver one and the second 128 channels are receiver two. A filterbank in series mode is a one receiver scan.

A. 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 copies from the disk, where it was stored by the on-line programs, to the POPS/Spectral 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 154 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.

B. VERBS, OBJECTS AND ADVERBS

The spectral line data reduction package has a language of its own. This language is very similar to the POPS 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: 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 is CSTACK and KSCANS).

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 reference array into the HOLD array. ACCUM adds the scan in the work array to the HOLD array, multiplying the spectrum by its integration time divided by its system temperature squared (INTEGRATION/TSYS2). To use 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 a spectrum by averaging NBOX channels together.

The values or 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 functions. CLIPMIN has two values, one for each of up to two receivers. NREGION is doubly dimensioned; it has eight values for each of up to two receivers.

A single values 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, statements of the form:

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

and so on, if the adverb has more values, are used. Also for convenience a multi-valued adverb may be set in the following manner:

```
ADVERB=value1, value2, value3, value4...etc.
```

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

Spectral line also uses a few variables which are not strictly adverbs. These variables are `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`.

C. WRITING, INSTALLING, EXECUTING, AND SAVING PROCEDURES

As described above, POPS allows a sequence of its "verbs" to be bundled into "Procedures". Often it is convenient to write procedures outside of LINE and install them when needed. For example, here is a procedure prepared with the DEC editor EDT as the file INT.PRC.

```
* This program integrates the intensity distribution for 2 regions of a
* spectrum by summing channels (filters) for a range of scans from STRTSCN
* to ENDSCN.
*
* Written by M. A. Gordon, 86.10.31
*
PROCEDURE IN(STRTSCN,ENDSCN)
* Enable the edit file (See section IV-G in this manual)
GETEDIT=1
* Enter the channel (filter) ranges for the summations
STRT1=24
END1=57
STRT2=152
END2=185
* Begin the summation loop, considering only even numbered scans
FOR SCANNO=STRTSCN TO ENDSCN BY 2
        GET SCANNO
*       Integrate the first range
        LSUM+0
        FOR J=STRT1 TO END1
                INDEX=J+129
                LSUM=LSUM+CHAN(INDEX)
*         CHAN(INDEX) returns the temperature stored in filter number INDEX
        END
*    Store result in INT1
        INT1=LSUM
*    Integrate the second range
        FOR J=STRT2 TO END2
                INDEX=J+129
                LSUM=LSUM+TWH(INDEX,PTWH)
        END
*    Store the result in INT2
*    List the results on the CRT screen
        PRINT 'SCAN, INT1, INT2 ', SCANNO,INT1,INT2
END
* End procedure
FINISH
```

After entering the LINE program, you install this procedure by typing

```
INSTALL INT <CR>
```

which transfers the procedure into LINE, line by line. Comment lines are ignored. To use this procedure, type

```
nnnn mmmm INT <CR>
```

or

```
INT (nnnn,mmmm) <CR>
```

which starts the procedure operating from scan nnnn through mmmm.

Procedures can also be written within LINE, but without comment lines, as described in the section A of the chapter "The POPS Language" . Because procedures are lost when you exit from LINE, store the procedure by typing

```
STORE n <CR>
```

where n is a number between 1 and 3. To recall the procedure (and all other variables which you may have set) after re-entering LINE, type

RESTORE n <CR>

D. USING AND CHANGING HEADER VALUES

Each record contains a "header" filled with parameters describing the observing equipment and spectral values. The parameters are either numerical or character, with slightly different procedures required to manipulate each type. You can use these values in calculations, and you can change these values permanently.

Numeric Parameters

Typing

```
PRINT AZ <CR>
```

lists the commanded azimuth in degrees for the record currently in memory. Or, typing

```
PRINT AZ*3.14/180 <CR>
```

lists the azimuth in radians. To change the azimuth number permanently to 185.2, type

```
SETAZ (185.2) <CR>.
```

You may also enter data in Reverse Polish Notation (RPN). Examples are

```
185.2 SETAZ <CR>, and

(5, -2.3456) SETCHAN <CR>
```

which sets channel (filter) 5 to the value -2.3456. Note that parentheses must enclose multiple arguments and some negative numbers.

Character or String Parameters

The names of character parameters usually begin with the letter S. Typing

```
PRINT SOBSID <CR>
```

will list the 8 character string loaded into this variable containing the identity of the observer associated with the spectral record in memory. To change the contents to _MAG_DTE, type

```
SOBSID = '_MAG_DTE'; SETOBSID <CR>
```

where the number of string characters must be equal to or less than the number defined by the character definition listed in the table below.

List of Parameters and Setting Verbs

The following table[1] lists header parameters recommended by an *ad hoc* standards committee representing Cambridge University, IRAM, the NRAO, and the University of Arizona. Those yet to be implemented at the 12m telescope are marked with an asterisk. For organizational reasons the parameters are grouped into "Classes".

---

[1] Based upon Memorandum 5 by R. L. Brown and E. B. Stobie in the NRAO 12-M Computing Report Series, 21 October 1986.

Description	Parameter	Setting Verb	Parameter Type
Class 1: Basic Information			
Header Length (bytes)	HEADLEN		R*8
Data Length (bytes)	DATALEN		R*8
Scan Number	SCAN	SETSCAN	R*8
Observer Initials	SOBSID	SETOBSID	C*8
Observer Name	SOBSERVER	SETOBSERVE	2C*8
Telescope	STELESCOP	SETTELESCO	C*8
Project Identification	SPROJID	SETPROJID	C*8
Source Name	SOBJECT	SETOBJECT	2C*8
Type Data/Observing Mode	SOBSMODE	SETOBSMODE	C*8
Frontend Descriptor	SFRONTEND	SETFRONTEN	C*8
Backend Descriptor	SBACKEND	SETBACKEND	C*8
Data Precision	SPRECIS	SETPRECIS	C*8
Class 2: Pointing Parameters			
Total Az/RA Pointing (arc sec)	XPOINT	SETXPOINT	R*8
Total El/Dec Pointing (arc sec)	YPOINT	SETYPOINT	R*8
User Az/RA Pointing (arc sec)	UXPNT	SETUXPNT	R*8
User El/Dec Pointing (arc sec)	UYPNT	SETUYPNT	R*8
Pointing Constants (4)	PTCON	SETPTCON	4R*8
Orientation Angle (deg)	ORIENT	SETORIENT	R*8
Radial Focus (mm)	FOCUSR	SETFOCUSR	R*8
North-South Focus (mm)	FOCUSV	SETFOCUSV	R*8
East-West Focus (mm)	FOCUSH	SETFOCUSH	R*8
Class 3: Observing Parameters			
Universal Time date (YY.MMDD)	UTDATE	SETUTDATE	R*8
Universal Time (hours)	UT	SETUT	R*8

LST (hours)	LST	SETLST	R*8
Number of Receiver Channels	NORCHAN	SETNORCHAN	R*8
Number of Switching Variables	NOSWVAR	SETNOSWVAR	R*8
Number of Phases per Cycle	NOPHASE	SETNOPHASE	R*8
Length of Cycle (sec)	CYCLLEN	SETCYCLLEN	R*8
Length of Sample (sec)	SAMPRAT	SETSAMPRAT	R*8

Class 4: Positions

Epoch (years)	EPOCH	SETEPOCH	R*8
Commanded Source X (deg)	XSOURCE	SETXSOURCE	R*8
Commanded Source Y (deg)	YSOURCE	SETYSOURCE	R*8
Commanded Reference X (deg)	XREF	SETXREF	R*8
Commanded Reference Y (deg)	YREF	SETYREF	R*8
Epoch Right Ascension (deg)	EPOCRA	SETEPOCRA	R*8
Epoch Declination (deg)	EPOCDEC	SETEPOCDEC	R*8
Galactic Longitude (deg)	GALLONG	SETGALLONG	R*8
Galactic Latitude (deg)	GALLAT	SETGALLAT	R*8
Commanded Azimuth (deg)	AZ	SETAZ	R*8
Commanded Elevation (deg)	EL	SETEL	R*8
Indicated X Position (deg)	INDX	SETINDX	R*8
Indicated Y Position (deg)	INDY	SETINDY	R*8
Descriptive Origin (3)	DESORG	SETDESORG	3R*8
Coordinate System Code	SCOORDCD	SETCOORDCD	C*8

Class 5: Environment

Ambient Temperature (C)	TAMB	SETTAMB	R*8
Pressure (mm Hg)	PRESSURE	SETPRESSUR	R*8
Relative Humidity (%)	HUMIDITY	SETHUMIDIT	R*8
Index of Refraction	REFRAC	SETREFRAC	R*8
Dew Point (C)	DEW_PT	SETDEW_PT	R*8
Precipitable Water Vapor (mm)	MMH2O	SETMMH2O	R*8

Class 6: Map Parameters[2]

Map Scanning Angle (deg)	SCANANG	SETSCANANG	R*8
X Position at Zero (arc sec)	XZERO	SETXZERO	R*8
Y Position at Zero (arc sec)	YZERO	SETYZERO	R*8
Delta X or X Rate (arcsec or arcsec/sec)	DELTAXR	SETDELTAXR	R*8
Delta Y or Y Rate (arcsec or arcsec/sec)	DELTAYR	SETDELTAYR	R*8
Number of Grid Points	NOPTS	SETNOPTS	R*8
Number of X Grid Points	NOXPTS	SETNOXPTS	R*8
Number of Y Grid Points	NOYPTS	SETNOYPTS	R*8
Starting X Grid Cell Number	XCELLØ	SETXCELLØ	R*8
Starting Y Grid Cell Number	YCELLØ	SETYCELLØ	R*8
XY Reference Frame Code	SFRAME	SETFRAME	C*8

Class 7: Data Parameters

Beam Fullwidth at Half Maximum (arcsec)	BFWHM	SETBFWHM	R*8
Off Scan Number	OFFSCAN	SETOFFSCAN	R*8
Bad Channel value	BADCHV	SETBADCHV	R*8
Velocity Correction (km s^{-1})	RVSYS	SETRVSYS	R*8
Velocity wrt Reference (km s^{-1})	VELOCITY	SETVELOCIT	R*8
Velocity Definition & Ref	SVELDEF	SETVELDEF	C*8
Type of Calibration	STYPECAL	SETTYPECAL	C*8

Class 8: Engineering Parameters

Antenna Aperture Efficiency	APPEFF	SETAPPEFF	R*8
*Antenna Beam Efficiency	BEAMEFF	SETBEAMEFF	R*8
*Antenna Gain	ANTGAIN	SETANTGAIN	R*8
*ETAL (Rear spillover & scattering eff)	ETAL	SETETAL	R*8

---

[2] For an explanation of these parameters, consult *Continuum Mapping with the 12-m Telescope* by C. J. Salter, 1985, NRAO 12-m Computing Report Series, No. 4.

*ETAFSS (Forward spillover & scatt. eff.)	ETAFSS	SETETAFSS	R*8

Class 9: Telescope Dependent Parameters

(NRAO Green Bank)

*L1	L1	SETL1	R*8
*L1F1	L1F1	SETL1F1	R*8
*L1F2	L1F2	SETL1F2	R*8
*L2	L2	SETL2	R*8
*L2F1	L2F1	SETL2F1	R*8
*L2F2	L2F2	SETL2F2	R*8
*LA	LA	SETLA	R*8
*LB	LB	SETLB	R*8
*LC	LC	SETLC	R*8
*LD	LD	SETLD	R*8
*Level Correction	LEVCORR	SETLEVCORR	R*8
*Pointing fudge	PTFUDGE	SETPTFUDGE	R*8
*RHO	RHO	SETRHO	R*8
*THETA	THETA	SETTHETA	R*8
*Center Frequency Formula	SCFFORM	SETCFFORM	8C*8

(NRAO Kitt Peak)

Synthesizer Frequency	SYNFREQ	SETSYNFREQ	R*8
LO Factor	LOFACT	SETLOFACT	R*8
Harmonic	HARMONIC	SETHARMONI	R*8
LO IF	LOIF	SETLOIF	R*8
First IF	FIRSTIF	SETFIRSTIF	R*8
Reference Azimuth Offset	RAZOFF	SETRAZOFF	R*8
Reference Elevation Offset	RELOFF	SETRELOFF	R*8
Beam Throw	BMTHROW	SETBMTHROW	R*8
*Beam Orientation	BMORENT	SETBMORENT	R*8
Baseline Offset	BASEOFF	SETBASEOFF	R*8

Observing Tolerance	OBSTOL	SETOBSTOL	R*8
Sideband	SIDEBAND	SETSIDEBAND	R*8
Wavelength	WL	SETWL	R*8
GAIN Scan Number	GAINS	SETGAINS	R*8
+ Beam	PBEAM	SETPBEAM	R*8
- Beam	MBEAM	SETMBEAM	R*8
RA/Dec Offsets	SROFF	SETSROFF	R*8

Class 10: Data Reduction Parameters (Open)

*Open Parameter Values	SOPENPAR	SETOPENPAR	1ØC*8

Phase Block

*Variable Value	VARVAL	SETVARVAL	R*8
*Variable Descriptor	SVARDES	SETVARDES	C*8
*Phase Table	PHASTB	SETPHASTB	C*8

Class 12: Description Block for Each Receiver "Channel"

Observed Frequency (MHz)	OBSFREQ	SETOBSFREQ	R*8
Rest Frequency or TP Calib. (MHz)	RESTFREQ	SETRESTFRE	R*8
Frequency Resolution/SP Calib. (MHz)	FREQRES	SETFREQRES	R*8
Bandwidth (MHz)	BW	SETBW	R*8
Receiver Temperature (K)	TRX	SETTRX	R*8
Calibration Temperature (K)	TCAL	SETTCAL	R*8
Source System Temperature (K)	STSYS	SETSTSYS	R*8
Reference System Temperature (K)	RTSYS	SETRTSYS	R*8
Source Temperature (K)	TSOURCE	SETTSOURCE	R*8
Standard Deviation of Mean (K)	TRMS	SETTRMS	R*8
Reference Point Number	REFPT	SETREFPT	R*8
X Value at the Reference Point	XØ	SETXØ	R*8
Delta X	DELTAX	SETDELTAX	R*8

Total Integration Time (sec)	INTTIME	SETINTTIME	R*8
Number of Integrations	NOINT	SETNOINT	R*8
Starting Point Number	SPN	SETSPN	R*8
H_2O Opacity	TAUH2O	SETTAUH2O	R*8
H_2O Temperature (K)	TH2O	SETTH2O	R*8
O_2 Opacity	TAUO2	SETTAUO2	R*8
O_2 Temperature (K)	TO2	SETTO2	R*8
Polarization	SPOLARIZ	SETPOLARIZ	C*8

Class 13: Reduction Parameters

Number of Scans Stacked	NOSTAC	SETNOSTAC	R*8
First Scan in Stack	FSCAN	SETFSCAN	R*8
Last Scan in Stack	LSCAN	SETLSCAN	R*8
Line Amplitude (K)	LAMP	SETLAMP	R*8
Line Width (channels)	LWID	SETLWID	R*8
Integrated Line Intensity (K channels)	ILI	SETILI	R*8
RMS Noise (K)	DRMS	SETDRMS	R*8
Data Channel (point)	CHAN(n)	SETCHAN(n,x)	R*4

E. WORKING DIRECTLY WITH THE POPS DATA RECORDS

Basic Concept

Each POPS record --- known as a scan --- consists of a number of header words describing the equipment and the observational configuration (see previous section), followed by the individual data. You can recall and manipulate individual data points without using the verbs described in the previous section and with somewhat more control.

In memory the scan is stored in one of the 3 arrays, known as Temporary(T), Work(W), or Hold(H) arrays. The variable PTWH (pointer for TWH arrays) takes on the values 1, 2, or 3 to indicate which of the 3 arrays are active. Each element of these arrays is then defined as TWH(n,PTWH), where n is the element sequence and PTWH is the array pointer. Typing

```
6250 GET <CR>
```

loads a scan into PTWH = 2, the Work array.

The header size may vary from one type of observation to another thereby changing the address of the first data element. The verb HEADLEN gives the header size. Typing

```
PRINT HEADLEN <CR>
```

will give the header length, nnn.

To list the contents of channel (filter) 43, you type

```
PRINT TWH(nnn+43,PTWH) <CR>
```

where nnn is the content of HEADLEN indicating the data offset from the beginning of the TWH array. This command is equivalent to typing

```
PRINT CHAN (43) <CR>
```

If you have an array stored in T (PTWH=1) and if PTWH happens to equal 2, you can recall the 43th datum in the T array, by typing

```
PRINT TWH(nnn+43,1) <CR>.
```

Only numbers may be recalled by this technique. String variables need to be converted into characters as described in the previous section.

Working with the Array Elements

Suppose you want to replace a "bad" channel (filter) --- filter 43 --- in a spectrum with the average of the 2 contiguous values. Type

```
TWH(nnn+43,PTWH) = 0.5*(TWH(nnn+42,PTWH)+TWH(nnn+44,PTWH)) <CR>
```

will effect the relacement. This statement is equivalent to typing

```
SETCHAN(43,(0.5*(CHAN(42)+CHAN(44)))) <CR>
```

F. KEEPING ALTERED DATA RECORDS

Saving a Single Altered Scan

While the previous section describe making changes in the scans (data records), these changes disappear when you exit the program. They are not permanent. To keep the altered scans, you create and store a copy of the altered scan. Typing

```
RESAVE <CR>
```

copies the scan from memory into the "keep" file PKFL.iii, where the iii are the observer identification letters (usually the observer's initials). To retrieve it, you must set the parameter

GETEDIT = 1 <CR>

to cause the GET verb to look for the scan in the PKFL file before looking in the original PDFL file. Setting GETEDIT to -1 will cause the GET verb to look only at the PDFL file.

An Alternate Save Technique

Alternatively, you can save an altered scan into one of 100 save bins on the disk. Typing

NSAVE = 3 <CR>

PMW SAVE <CR>

will transfer the contents of the Work array into bin #3. To recall this scan for additional work, type

NSAVE = 3 <CR>

RECALL <CR>

This method is a "move to the edge of the desk" method of the contents of an array, whereas the RESAVE technique places altered scans into a storage system which can be recalled using the standard verbs such at GET nnn, etc.

An Even More Alternate Save Technique

Scans in the work areas can also be stored in the "keep" file, PDFL.iii, by typing

KEEP <CR>

However, if the scan is the result of a stack and averaging process, you will not be able to recall it into the online LINE program. This save method facilitates the transfer of the PKFL.iii file into a FITS format, which CAN be analyzed by other programs.

G. EDITING INDIVIDUAL RECORDS WITHIN A SCAN

A position-switched scan is usually the average of many off/on record pairs. You may edit the indivual records comprising these scans. The file IRFL.iii stores the individual records. Typing

```
6250 RGET <CR>
```

recalls the individual records of the scan 6250. nnnn RSHOW1 and nnnn RSHOW2 displays the records of filter bank 1 and 2 in the same plot so that they may be compared. RECN selects the number of records to be displayed on a single page, and SHIFT gives the temperature offset between contiguous spectra in the plot, as in

```
SHIFT = 0.1; RECN = 50; 6250 RSHOW1 <CR>
```

to plot 50 record spectra from scan 6250, each plot with a vertical separation of 0.1 K.

To delete the individual records n, m,...,o from the scan, type

```
-n, -m, ... , -o RECS <CR>
```

To select only these records to be kept, type

```
n, m, ..., o RECS <CR>
```

RECS loads the stack RSTACK with the appropriate records. Typing

```
PRINT RCOUNT <CR>
```

gives the number of records in the stack, and

```
PRINT RSTACK <CR>
```

lists the specific records. To plot the edited scan, type

```
6250 REDIT1 <CR>, or

6250 REDIT2 <CR>
```

for filterbanks 1 or 2. To store the edited scan, type RESAVE. To recall the edited scan, set GEDTEDIT = 1 and type 6250 GET --- as decribed in an earlier section.

VERB SYNOPSIS

ACCESS The following relate to access or storage of data on disc, from within the LINE analysis program.

FETCH	Retrieves total power raw data from disk.
GET	Copies a raw spectral scan from disk to the WORK array.
GGET	Copies the current calibration array or specified calibration array from the disk to the WORK array.
KEEP	Copies the scan in the currently referenced array into the PKFL file on disk.
OFF	Copies a scan from the disk to the TEMP array. OFF requires the desired scan number as its object.
ON	Copies a scan from the disk to the WORK array.
PMT, PMW, PMH	Set the value of PTWH to point at one of the three TWH arrays.
QGET	Copies current scan from the disk to the WORK array.
RECALL	Copies the scan in the disk save bin named by the adverb NSAVE into the currently referenced array.
RGET	Retrieves all records of a given scan from the individual record file.
SAVE	Copies the scan in the currently referenced array into one of the 100 save areas on the disk called "BINS".
SORT	Sorts spectral line data in observer's raw data file and fills stack (ASTACK) with scan numbers that meet the sort criteria.
TRT, TRW, TRH	Copies the contents of the currently referenced array to the indicated array and sets PTWH to that array as well.
ZGET	Copies current zero check or specified zero check into the WORK array.

ANALYSIS Various keywords for data analysis and manipulation.

ALIGN	Calculates number of channels by which scan must be shifted to add equal velocities or frequencies and sets this value in ASHIFT(r).
BASELINE	Fits and removes polynomial frequency baseline from data in the currently referenced array.
BIAS	Adds the values of FACT(r) to the scan in the currently referenced array.
BMODEL	Evaluates the last computed coefficients for a polynomial baseline for each channel and replaces the data in the currently referenced array with the evaluated polynomial.
BOXCAR	Smoothes scan by averaging odd number of consecutive channels together.
BSET	A procedure which uses the vertical crosshair to set the values of NREGION for receiver 1. (See Procedures)
BSET2	A procedure which uses the vertical crosshair to set the values of NREGION for receiver 2. (See Procedures)
BSHAPE	Calculates coefficients of a Chebyshev polynomial for specified spectral values by a least squares fit.
BSHOW	Fits, evaluates, and displays a polynomial frequency baseline without altering data in the current array.
CHGRES	Smoothes spectral line data from its initial resolution to a lower resolution specified by NEWRES.
DCBASE	Computes average spectral value over specified range and subtracts average from all values in current array.
DELTAV	Calculates area, mean flux, high & low velocity peak fluxes, high & low peak velocities, and delta velocities at specified fractions of peak flux.
DIFF	Subtracts the scan in the TEMP array from the currently referenced array.
DIVIDE	Divides the scan in the currently referenced array by the scan in the TEMP array and leaves the result in the currently referenced array.
FFT	Performs a fast Fourier transform of the data in the currently referenced array.

GAUSS	Fits up to 6 gaussian functions over a specified interval of a scan.
GDISPLAY	Evaluates the parameters of NGAUSS gaussians and displays the sum without altering current array.
GMEASURE	This routine calculates the area, mean flux, high and low velocity peak fluxes, high and low peak delta velocities and the minimum flux and delta velocity of a specified region in the currently referenced array.
GMODEL	Evaluates the parameters of NGAUSS gaussians and constructs the sum of the gaussians.
GPARTS	Separately evaluates the parameters of NGAUSS gaussians, constructs & displays each gaussian.
HANNING	Smoothes the scan in the currently referenced array with 1/4, 1/2, 1/4 weighting.
INVERT	Reverses the order of the spectral values and changes the sign of the velocity increments.
MOMENT	Calculates the MOMENT specified by NMOMENT over a specified range of channels and stores the result in the variable SIZE(r).
PEAK	Finds the CENTER, HWIDTH, and HEIGHT of a single gaussian.
POWSPEC	Computes the power spectrum of the data in the currently referenced array.
RAP	Averages the signal and reference spectra of a frequency switched scan in the currently referenced array.
RDIFF	Subtracts the contents of the B receiver of the scan in the currently referenced array from the contents of the A receiver of the same scan.
RESIDUAL	Takes the parameters of the gaussians, (usually refined by gauss), evaluates the total for each channel & subtracts the total from the currently referenced array.
RIPPLE	Is used to remove a sinusoidal frequency baseline from the data.
RMODEL	Evaluates the last calculated sine curve fit for each channel and replaces the currently referenced array with the sine curve.
RMS	Computes sample root mean squared of the scan in the currently referenced array.

`RSET`	A procedure which uses the vertical crosshair to determine baseline regions by setting `REGION`. (See Procedures)
`RSHAPE`	Fits a sine curve to the scan in the currently referenced array, using the specified region(s) of the data & the approximate frequency specified.
`SCALE`	Multiplies the currently referenced array by the value of `FACT`(r).
`SLIDE`	Averages 2-receiver data; B is shifted by `RSHIFT` channels with respect to A.
`SMOOTH`	Smoothes the scan in the currently referenced array by the weight function `SMWGT` specified by the user.
`SOLVETIP`	(Documentation not available)
`TEMP`	Computes spectral line quotients.

BASELINE COMPUTATION AND REMOVAL

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

CONTINUUM	Some of the features of the CONDAR continuum analysis program are available with LINE - *e.g.* 5-point pointing analysis, sky tips, and focus determinations.
AVGD	Computes the mean source temperature and standard deviation of the mean of continuum digital backend data.
CF	
CGET	Retrieves continuum data from disk within the LINE program.
CHEAD	
CS1	A procedure to retrieve the first receiver of the specified continuum scan from disk.
CS2	A procedure to retrieve the second receiver of the specified continuum scan from disk.
CSORT	Sorts the continuum data in the observer's data file and fills the stack (ASTACK) with scan numbers that meet the sort requirements specified.
DIFTIP	An alternative analysis technique for sky tip data.
FOCALIZE	
SPTIP	
STIP	This procedure fits tipping scan data for zenith optical depth and for rear spillover efficiency.
SWITCHED	Computes switched power data from the raw phases in the continuum digital backend scan.
TOTALPWR	Computes the total power data from the raw phases of the continuum digital backend scan.
TPTIP	

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

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

DISK-ARRAY

CGET #	Copies the current filterbank to the WORK array. Operand = 1 or 2.
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.

DISPLAY

The following keywords relate to display of data on a Tektronix-compatible terminal (*e.g.*, the Modgraph), and general interactive work (*e.g.*, use of the cursor).

?	Identical to PRINT (q.v.).
AUTO (FREEY)	Used to restore control of the y-scaling to the program after the user has overridden it.

BOTH	Restores the default, two-receiver processing option after ONLYA or ONLYB has been used. See ONLYA and ONLYB.
CCUR	Activates the vertical crosshair and returns the channel number for its position when any key other than <RETURN> is struck.
CHAR	Prints a string of characters on the CRT at the current cursor position. Used in procedures.
COMMENT	Places the cursor at the bottom of the CRT screen, waits for the user to type in a remark no longer than a single line, then moves the cursor to the top of the screen.
CROSSHAIR	Activates horizontal and vertical crosshairs.
DEC	Sets an internal flag that causes MAPSHOW to use Declination as the y-coordinate for maps.
F	A procedure to retrieve the first filterbank of the specified scan from disk, erase the CRT page and display the scan on the CRT. (See Procedures)
FLAG	Draws a vertical flag at the specified frequency on the current display and labels the frequency.
FREEY	See AUTO
FULLGRID	Draws a rectangular grid over the current display on the CRT screen.
FV (VF)	Sets the axes labeling mode so that SHOW displays velocity for the upper x-axis labeling and frequency for the lower x-axis labeling.
G1	A procedure which retrieves and displays the current gains of the first filter bank.
G2	A procedure which retrieves and displays the current gains of the second filter bank.
GB	Sets an internal flag that causes MAPSHOW to use Galactic Latitude as the y-coordinate for maps.
GL	Sets an internal flag that causes MAPSHOW to use Galactic Longitude as the y-coordinate for maps.
GSHOW	A procedure that fits single or multiple (not greater than 6) component gaussian function to the data in the currently referenced array. (See Procedures)

HISTOGRAM	Sets plotting mode for SHOW display to HISTOGRAM.
HOLDY	Causes the SHOW verb to use the y-scaling that was last used before HOLDY was specified.
LABEL	Used to complete contour map begun by MAPSHOW. Draws and labels axes.
LIMIT	Restricts data processing to only those channels within the minimum and maximum channels specified.
LINE	Sets plotting mode for SHOW display to LINE.
MAKECOPY	Is used to make a hard copy of the CRT screen.
MAPSHOW	Draws a position vs. velocity contour map of a set of scans.
ONLYA	Changes the two-receiver processing option of the scan in the currently referenced array to the A receiver only.
ONLYB	Changes the two-receiver processing option of the scan in the currently referenced array to the B receiver only.
PAGE	Erases the CRT screen and returns a caret (>) at the top of the now blank screen.
PAUSE	Causes the program to suspend operations for ten seconds.
PLACE	Moves cursor to position commanded by IX & IY.
POINTS	Sets plotting mode for SHOW display to POINTS.
PRINT <FIELD>	Prints the <FIELD> requested on the CRT.
Q1	A procedure to retrieve and display the current quotients of the first filter bank.
Q2	A procedure to retrieve and display the current quotients of the second filter bank.
RA	Sets an internal flag that causes MAPSHOW to use Right Ascension as the y-coordinate for maps.
RANGE	Allows the user to specify minimum and maximum values for the y-scaling in SHOW.

RECSHOW	A procedure that displays all of the individual records of the scan loaded into program memory.
RESET	A procedure to reset all markers on the SHOW and RESHOW display to their initial values.
RESHOW	Plots the contents of the currently referenced array using the y-scaling last used by the SHOW verb.
RHIST	Sets plotting mode for RESHOW display to HISTOGRAM.
RLINE	Sets plotting mode for RESHOW display to LINE.
RPOINTS	Sets plotting mode for RESHOW display to POINTS.
RSHOW	Displays all the individual records of the scan currently in memory in profile form.
RSHOW1	A procedure which gets and displays all of the individual records of the first filterbank of the specified scan in profile form. (See Procedures)
RSHOW2	A procedure which gets and displays all of the individual records of the second filterbank of the specified scan in profile form. (See Procedures)
S	A procedure to retrieve the second filterbank of the specified scan from disk, erase the CRT page and display the scan on the CRT. (See Procedures)
SHOW	Plots contents of the currently referenced array on the CRT, draws and labels the axes.
SHOW1	A procedure which displays the first filter bank of the scan numbers in the stack in a HISTOGRAM profile form. (See Procedures)
SHOW2	A procedure which displays the second filter bank of the scan numbers in the stack in a HISTOGRAM profile form. (See Procedures)
SHOWS	A procedure which displays the specified filter bank of the scan numbers in the stack in profile form. (See Procedures)
SPREAD	Restricts data processing to those channels within the minimum and maximum frequency or velocity specified Units depend on the coordinate of the lower X-axis.
TABLE	Prints antenna temperatures(kelvin) for each channel of scan in the currently referenced array and labels the information with scan number and filterbank.

`TCUR`	Activates the horizontal crosshair and returns the temperature (in Kelvin) for its position when and key other than <RETURN> is struck.
`VC (CV)`	Sets the axes labeling mode so that `SHOW` displays channel number for the upper x-axis labeling and velocity for the lower x-axis labeling.
`VCTR`	Draws a line from the current cursor position to the position commanded by IX & IY.
`VCUR`	Activates the vertical crosshair and returns the velocity value for its position when any key other than <RETURN> is struck.
`XCUR`	Returns the X position of the cursor in terms of TEK points.
`XX`	A procedure to erase the CRT page and display the data in the currently referenced array.
`YCUR`	Returns the Y position of the cursor in terms of TEK points.
`Z1`	A procedure which retrieves and displays the current zero check of the first filter bank. (See Procedures)
`Z2`	A procedure which retrieves and displays the current zero check of the second filter bank.

DISK INDEX

`TELL DISK`	Lists the scans currently in the disk index.

DOCUMENT

These keywords relate to display of header information, scans available on disc, etc.

`CHECK`	Prints out the scan number (if valid) of data in the current `NSAVE` bin.
`DATALOG`	Prints on the CRT a listing of all spectral line scans in the observer's raw data file, giving the scan number, source name, velocity and rest frequency for each scan.

DOC	Writes one line of header information about the scan in the currently referenced array.
HEAD	A procedure that retrieves the specified scan and generates a listing of header information for that scan on the CRT screen. (See Procedures)
HEADER	Prints header information for the scan in the currently referenced array on the CRT.
LDISK	A procedure which prints one line of header information: scan number, date of observation, LST, position, length of scan, system temperature, filter bank resolution, frequency and source name for the specified scan range on the printer. (See Procedures)
LOGFILE	Writes two 80-byte records to file LOGFILE.DAT.
PDOC	Is used by the LF and CF pointing procedures to display the five points, the gaussian fits, updated pointing corrections and peak flux densities.
SNAP	Prints the header information for the currently referenced array on the printer.
TABLE	Lists the spectral values for the scan in the currently referenced array.
TELL	Lists information about data in disk files including PDFL(raw), IRFL(ind rec), PKFL(KEEP and edited) files.
THUMB	Prints main and reference azimuth, elevation, right ascension and declination offsets for the scan in the currently referenced array.
TITLE	Writes one line of documentation about the current scan on the CRT screen including scan number, LST, source name, rest frequency, filterbank and the sideband number.

EDITING Keywords relating to data editing.

ALL	A procedure to set all channels dropped to zero. (See Procedures)
BADCH	A procedure that determines bad data channels data is displayed with the bad channels removed and the bad channels are written across the top of the screen. (See Procedures)

CLIP	Resets any spectral value in the scan in the currently referenced array that is > CLIPMAX to CLIPMAX and any that is < CLIPMIN to CLIPMIN.
EDITING	Specifies whether or not the edited file should be searched before the raw data file for scan retrieval.
ESTACK	Lists all record numbers currently in the RSTACK.
INVERT	Reverses the order of the spectral values and changes the sign of the velocity increments.
PBAD	A procedure called by the BADCH procedure to compute bad channels for the second receiver channel of the 2 MHz filterbanks. (See Procedures)
RECS	Used to specify which records the observer wishes to retain in a given scan when record editing.
REDIT	A procedure which averages together the specified individual records of the the scan loaded into program memory. (See Procedures)
REDIT1	A procedure that loads the individual records of the first filterbank of the desired scan into memory, averages together the specified records (see RECS verb), displays the result on the CRT screen and stores the result in the LINE EDIT file. (See Procedures)
REDIT2	A procedure that loads the individual records of the second filterbank of the desired scan into memory, averages together the specified records (see RECS verb), displays the result on the CRT screen and stores the result in the LINE EDIT file. (See Procedures)
REPLACE	Replaces up to 12 bad channels (BADPT) in the currently referenced array by a parabolic interpolation over adjacent channels.
RESAVE	Stores the scan in the currently referenced array as an edited scan in the PKFL file.
RSCALE	Computes the appropriate y-scaling for the individual record in the currently referenced array.
SPIKE	Searches for bad channels according to the criteria ABSOLUTE VALUE OF CHANNEL VALUE > NSIGMA.
SPN	Puts the value of the starting point number for the selected receiver of the data in the currently referenced array on the value stack.

FOURIER TRANSFORM

FFT Computes the Fourier transform of the scan in the currently referenced array.

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 Displays the separate gaussians fit by GAUSS.

PEAK Finds the CENTER, HWIDTH, and HEIGHT of a single gaussian.

RESIDUAL Subtracts the gaussian(s) fit by GAUSS from the data in the currently referenced array.

GET HEADER

These keywords are used to retrieve header parameters from the scan in the currently referenced array.

APPEFF Puts the value of the antenna aperture efficiency (ratio of total power observed to the total power incident on the telescope) of the data in the currently referenced array on the value stack.

AZ Puts the value of the commanded azimuth of the source or position specified of the data in the currently referenced array on the value stack.

BACKEND Puts a field of eight characters that describe the backend used into the adverb SBACKEND. This adverb may then be printed.

`BADCHV, GAINS`	Puts the value of the antenna temperature to be assigned to those filterbank channels, noted defective, of the data in the currently referenced array on the value stack.
`BASEOFF`	Puts the value of the baseline offset of the data in the currently referenced array on the value stack.
`BFWHM`	Puts the value of the beam full width at half maximum of the data in the currently referenced array on the value stack.
`BMORENT`	Not currently implemented.
`BMTHROW`	Puts the value of the beam throw parameter of the data in the currently referenced array on the value stack.
`BW`	Puts the value of the bandwidth (total bandwidth of filterbank) of the selected receiver of the data in the currently referenced array on the value stack. The units are MHz.
`CHAN`	Puts the value of the specified data channel of the data in the currently referenced array on the value stack.
`COORDCD`	Puts a field of eight characters which specifies in which coordinate system the data in the currently referenced array was commanded into the adverb `SCOORDCD`. (See Adverbs)
`CYCLLEN`	Puts the value of the cycle length in seconds (time required to complete one cycle) of the data in the currently referenced array on the value stack.
`DATALEN`	Puts the value of the data length (in R*4 words) of the scan in the currently referenced array on the value stack.
`DELTAX`	Puts the value of the step-size along the x-axis (usually velocity) for the selected receiver of the data in the currently referenced array on the value stack.
`DELTAXR`	Puts the value of `DELTA X` or `X RATE` of the data in the currently referenced array on the value stack.
`DELTAYR`	Puts the value of the cell size or distance between cells on the Y axis of the data in the currently referenced array on the value stack.
`DESORG`	Puts the value of the descriptive origin (horizontal position, vertical position, position angle describing the orientation of the sky "horizontal axis") of the data in the currently referenced array on the value stack.

DSF	A value of the data scale factor is placed on the value stack if the data is continuum.
EL	Puts the value of the commanded elevation of the source or position specified of the data in the currently referenced array on the value stack.
EPOCH	Puts the value of epoch (as specified by the observer) of the data in the currently referenced array on the value stack.
EPOCRA	Puts the value of the epoch right ascension of the source or position specified of the data in the currently referenced array on the value stack.
EPODEC	Puts the value of the epoch declination of the source or position specified of the data in the currently referenced array on the value stack.
FIRSTIF	Puts the value of the FIRST-IF of the data in the currently referenced array on the value stack.
FOCUSH	Puts the value of the horizontal (East-West) focus offset of the data in the currently referenced array on the value stack.
FOCUSR	Puts the value of the radial focus position of the data in the currently referenced array on the value stack.
FOCUSV	Puts the value of the vertical (North-South) focus offset of the data in the currently referenced array on the value stack.
FRAME	Puts a field of eight characters in which the firstfour characters state whether the grid is polar (POLR)or cartesian (CART) and the second four state whether DELTAXR and DELTAYR refer to STEP sizes or SCANning rates of the data in the currently referenced array into the adverb, SFRAME. (See Adverbs)
FREQRES	Puts the value of the frequency resolution (bandwidth per channel) of the selected receiver of the spectral line data in the currently referenced array on the value stack.
FRONTEND	Puts a field of eight characters to describe the receiver used into the adverb SFRONTEND. (See Adverbs)
GALLAT	Puts the value of the commanded galactic latitude of the source or position specified of the data in the currently referenced array on the value stack.
GALLONG	Puts the value of the commanded galactic longitude of the source or position specified of the data in the currently referenced array on the value stack.

HARMONIC	Puts the value of the harmonic of the data in the currently referenced array on the value stack.
HEADLEN	Puts the value of the header length (in R*4 words) of the scan in the currently referenced array on the value stack.
INDX	Puts the value of the horizontal telescope coordinate of the position actually observed (*i.e.*, position measured by horizontal decoder) of the data in the currently referenced array on the value stack.
INDY	Puts the value of the vertical telescope coordinate of the position actually observed, *i.e.*, position measured by vertical decoder) of the data in the currently referenced array on the value stack.
INTTIME	Puts the value of the total integration time for the selected receiver of the data in the currently referenced array on the value stack.
LOFACT	Puts the value of the LO factor (fundamental, doubler or tripler) of the data in the currently referenced array on the value stack.
LOIF	Puts the value of the LO-IF of the data in the currently referenced array on the value stack.
LST	Puts the value of LST (local sidereal time at the start of the observation) of the data in the currently referenced array on the value stack.
NOINT	Puts the value of the total number of integrations for the selected receiver of the data in the currently referenced array on the value stack.
NOPHASE	Puts the value of the number of phases per cycle (different states of switched variables) of the data in the currently referenced array on the value stack.
NOPTS	Puts the value of the number of points (total number of cells in the map) of the data in the currently referenced array on the value stack.
NORCHAN	Puts the value of the number of independent polarizations or channels of this frontend of the data in the currently referenced array on the value stack.
NOSWVAR	Puts the value of the number of switching variables of the data in the currently referenced array on the value stack.
NOXPTS	Puts the value of the number of x points (map sample points along the "X-edge" of the rectangle) of the data in the currently referenced array on the value stack.

NOYPTS	Puts the value of the number of y points (map sample points along the "Y-edge" of the rectangle) of the data in the currently referenced array on the value stack.
NT	The value of NT, noise tube true or false, is place on the value stack if the data is continuum.
OBJECT	Puts a field of sixteen characters which describe the source name of the object observed into the adverb SOBJECT.
OBSERVER	Puts a field of sixteen characters containing the observer name of the data in the currently referenced array into the adverb SOBSERVER.
OBSFREQ	Puts the value of the observed frequency of the selected receiver channel of the data in the currently referenced array on the value stack.
OBSID	Puts a field of eight characters containing the observer and operator initials of the data in the currently referenced array into the adverb SOBSID. (See Adverbs)
OBSMODE	Puts a field of eight characters where four describe the type of data and four describe the observing mode into the adverb, SOBSMODE.
OBSTOL	Puts the value of the observing tolerance of the data in the currently referenced array on the value stack.
OFFSCAN	Puts the value of the off scan number (the last designated total power off scan) of the data in the currently referenced array on the value stack.
OPENPAR	Puts a field of 56 characters (the entire string is 80 characters in length but the first 24 are reserved for stacking) of the data in the currently referenced array into the adverb, SOPENPAR.
ORIENT	Puts the value of the rotation or polarization angle orientation of the receiver or reflector at the prime focus of the data in the currently referenced array on the value stack.
POLARIZ	Puts a field of eight characters which describe the type of polarization and the angle for the selected receiver of the data in the currently referenced array on the value stack.
PRECIS	Puts a field of eight characters describing the precision of the data into the adverb SPRECIS.

PROJID — Puts a field of eight characters which contains the program ID associated with the proposal as it appears on the telescope schedule of the data in the currently referenced array into the adverb **SPROJID**.

PTCON — Puts the value of the selected pointing constant ofthe data in the currently referenced array on the value stack.

RAZOFF — Puts the value of the reference azimuth offset in seconds of arc of the data in the currently referenced array on the value stack.

REFPT — Puts the value of the reference point (center channel) of the selected receiver of the data in the currently referenced array on the value stack.

RELOFF — Puts the value of the reference elevation offset in seconds of arc of the data in the currently referenced array on the value stack.

RESTFREQ — Puts the value of the rest frequency of the selected receiver channel of spectral line data in the currently referenced array on the value stack.

RTSYS — Puts the value of the reference system temperatureof the selected receiver of the data in the currently referenced array on the value stack.

RVSYS — Puts the value of the Doppler correction for theearth's motion in the source direction with respect to the velocity reference frame chosen of the data in the currently referenced array on the value stack.

SAMPRAT — Puts the value of the sample period (time required to complete a single sample) of the data in the currently referenced array on the value stack.

SCAN — Puts the value of the scan number of the data in the currently referenced array on the value stack.

SCANANG — Puts the value of the map scanning angle of the data in the currently referenced array on the value stack.

SIDEBAND — Puts the value of the sideband used for the data in the currently referenced array on the value stack.

STSYS — Puts the value of the source system temperature of the selected receiver of the data in the currently referenced array.

SYNFREQ — Puts the value of the synthesizer frequency of the data in the currently referenced array on the value stack.

TAMB — Puts the value of the ambient temperature of the data in the currently referenced array on the value stack.

TAUH2O	Puts the opacity of water as computed by a model for the selected receiver of the data in the currently referenced array on the value stack.
TAUO2	Puts the value of the opacity of oxygen as computed by a model for the selected receiver of the data in the currently referenced array on the value stack.
TCAL	Puts the value of the calibration temperature of the selected receiver of the data in the currently referenced array on the value stack.
TELESCOP	Puts a field of eight characters which contain the telescope name of the data in the currently referenced array into the adverb STELESCOP.
TH2O	Puts the value of the temperature of water as computed by a model for the selected receiver of the data in the currently referenced array on the value stack.
TO2	Puts the value of the temperature of oxygen as computed by a model for the selected receiver of the data in the currently referenced array on the value stack.
TRMS	Puts the value of the standard deviation of the mean source temperature of the selected receiver of the data in the currently referenced array on the value stack.
TRX	Puts the value of the measured receiver temperature of the selected receiver of the data in the currently referenced array on the value stack.
TSOURCE	Puts the value of the source temperature of the selected receiver of the data in the currently referenced array on the value stack.
TYPECAL	Puts a field of eight characters which describe the calibration method used for the data in the currently referenced array into the adverb, STYPECAL.
UT	Puts the value of universal time of the data in the currently referenced array on the value stack.
UTDATE	Puts the value of the universal time date of the data in the currently referenced array on the value stack.
UXPNT	Puts the value of the user Az/RA pointing correction of the data in the currently referenced array on the value stack.
UYPNT	Puts the value of the user EL/Dec pointing correction of the data in the currently referenced array on the value stack.

VELDEF	Puts a field of eight characters which describe the velocity system of the data in the currently referenced array into the adverb, SVELDEF.
VELOCTY	Puts the value of the velocity with respect to the reference of the data in the currently referenced array on the value stack.
X0	Puts the value of the velocity at the reference point for the selected receiver of the data in the currently referenced array on the value stack.
XCELL0	Puts the starting X cell (cell number) of the data in the currently referenced array on the value stack.
XPOINT	Puts the value of the total pointing correction applied in the X (horizontal - Az/RA) direction of the data in the currently referenced array on the value stack.
XREF	Puts the value of the commanded reference Xcoord (horizontal coordinate of the reference position in the system specified by the observer) of the data in the currently referenced array on the value stack.
XSOURCE	Puts the value of the commanded source Xcoord (the horizontal coordinate of source or position in the coordinate system specified by the observer) of the data in the currently referenced array on the value stack.
XZERO	Puts the value of the horizontal telescope coordinate at the map reference position of the data in the currently referenced array on the value stack.
YCELL0	Puts the value of the starting y cell (cell number) of the data in the currently referenced array on the value stack.
YPOINT	Puts the value of the total pointing correction applied in the Y (vertical - EL/Dec) direction of data in the currently referenced array on the value stack.
YREF	Puts the value of the commanded reference Ycoord (vertical coordinate of the reference position in the coordinate system specified by the observer) of the data in the currently referenced array on the value stack.
YSOURCE	Puts the value of the commanded source Ycoord (vertical coordinate of source or position in the coordinate system specified by the observer) of the data in the currently referenced array on the value stack.
YZERO	Puts the value of the Y position at Zero of the data in the currently referenced array on the value stack.

HELP — Documents all verbs, procedures, array variables and SCALAR variables.

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.
SCALE	Multiplies the scan in the currently referenced array be a constant.

SPECIAL FUNCTIONS

MOMENT	Computes the moment requested by the value of the adverb NMOMENT 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.

POINTER CHANGE

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

SMOOTHING

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

STACKING

A	Inserts one scan into the stack (ASTACK).
A1	A procedure which clears the accumulator (HOLD) array, accumulates the first filter bank of the positive scan numbers in the stack, accumulates the first filter bank of the negative scan numbers in the stack, differences the two sums, erases the CRT screen and displays the final result with the scan numbers in the stack.
A2	A procedure which clears the accumulator (HOLD) array, accumulates the second filter bank of the positive scan numbers in the stack, accumulates the second filter bank of the negative scan numbers in the stack, differences the two sums, erases the CRT screen and displays the final result with the scan numbers in the stack.
AB	A procedure which clears the accumulator (HOLD) array, accumulates the both filter banks of the positive scan numbers in the stack, accumulates both filter banks of the negative scan numbers in the stack, differences the two sums, erases the CRT screen and displays the final result with the scan numbers in the stack.
ACCUM	Sums weighted scans into HOLD array from WORK array. First use after program start, AVE or SCLEAR copies header and spectral values into HOLD array weighted by integration time/system temp**2).
ADD	Inserts a series of 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.

ALIGNF	Sets an internal flag so that the **ALIGN** verb will align frequencies.
ALIGNV	Sets an internal flag so that the **ALIGN** verb will align velocities.
AVE	Divides the accumulated scans in the **HOLD** array by the accumulated weighting factor (Sum of individual integration times/individual squares of system temperatures or sum of user supplied weighting.)
C1	A procedure which clears the accumulator (**HOLD**) array, accumulates the first filter bank of all scans in the stack aligning the data in frequency (**ALIGNF**, the default) or velocity (**ALIGNV**) and weighting the data by integration time /TSYS**2.
C2	A procedure which clears the accumulator (**HOLD**) array, accumulates the second filter bank of all scans in the stack aligning the data in frequency (**ALIGNF**, the default) or velocity (**ALIGNV**) and weighting the data by integration time /TSYS**2.
CB	A procedure which clears the accumulator (**HOLD**) array, accumulates both filter banks of all scans in the stack aligning the data in frequency (**ALIGNF**, the default) or velocity (**ALIGNV**) and weighting the data by integration time /TSYS**2.
DELETE	Deletes one scan from the stack.
EMPTY	Deletes all scans from the **STACK** and sets the stack counter to zero.
FOLD	A procedure that averages the signal and reference and the center channel is redefined by **FS**.
HALVES	A procedure that averages the data from two receivers of a filter bank in parallel configuration.
LISTSHIFT	Lists all scans that have currently been stacked including the alignment offset that has been computed by **ALIGN** on the CRT. See **ALIGN**, **ALIGNF**, **ALIGNV**.
NSTACK	A procedure which accumulates all data of the specified filter bank whose scan numbers have been entered negatively in the stack.
OSB	Sets a flag so that **ALIGN** computes the shift when aligning frequency or velocity for the other sideband by changing the sign of the shift.
PSTACK	A procedure which accumulates all data of the specified filter bank whose scan numbers have been entered positively in the stack.

RACCUM	Accumulates and averages the records specified by **RSTACK** after the records are loaded into memory by **RGET**. See **RGET**, **RSCALE**, **RESAVE**.
SCLEAR	Clears the accumulator (**HOLD**) array before stacking data. See **ACCUM**, **AVE**, **SUM**.
STACK	Lists all scans currently in the **STACK**.
SUM	Sums weighted scans into **HOLD** array from **WORK** array.
TELL CSTACK	Lists the number of scans accumulated in the **HOLD** array, as well as the scan numbers of the first 34 and the last scan that were accumulated.

TWO-RECEIVERS

BOTH	Selects processing and display of both receivers.
ONLYA	Selects processing and display of the A receiver.
ONLYB	Selects processing and display of the B receiver.
RDIFF	Subtracts receiver B of the scan in the currently referenced array from receiver A and puts the result in the receiver A channels.
SLIDE	Averages receivers A and B of the currently referenced array together, with weighting by system temperature and integration time, and puts the result in the receiver A channels.

*

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.

TWO-RECEIVER PROCESSING OPTION Does not apply.

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 = 30, EBASE = 30, NFIT = 2, SMOOTHED,
  NBOX = 3
```

A

A will insert one scan into the STACK.

PTWH	Is not used or changed.
ADVERBS	
0 ACOUNT	The STACK counter is incremented by one.
0 ASTACK(n)	The specified scan is added to the STACK array. The maximum number of entries is 200.
OPERANDS	The scan number to be inserted into the STACK.
TWO-RECEIVER OPTION	Does not apply.
RELATED VERBS	
ADD (#,#)	Inserts a series of scans into the STACK.
DELETE	Deletes one scan from the STACK.
EMPTY	Deletes all scans from the STACK and sets the stack counter to zero.
STACK	Lists all scans in the STACK.

REMARKS Positive and negative scan numbers may be entered in the STACK. The operand may precede or follow A. A negative scan number is ignored by the C1, C2 and CB procedures and causes that scan to be added to the negative sum by the A1, A2, and AB procedures.

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

EXAMPLE

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

```
650 A   or   A 650
```

To add -660 to the scans in the STACK specify:

```
A -660  or  -660 A
```

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 spectral values in the WORK array to the HOLD array. The spectral values and system temperatures are multiplied by the weight factor* of the scan. Subsequent uses of ACCUM add the spectral values and system temperatures contained in the WORK array (multiplied by the weight factor) 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, which is 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 34 and the last scan accumulated.

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

ADVERBS

0 ASHIFT (RCVR)	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 $\neq 0$.

TWO-RECEIVER PROCESSING OPTION	Does affect ACCUM. Either or both receivers may be accumulated, depending on the processing option.

RELATED VERBS

AVE	Divides the accumulated scans by the total weight factor. Sets the internal stack counter to zero. (See ** note on next page)
TELL CSTACK	Prints the first, last and number of scans stacked.
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, or its frequencies will align with the frequencies in the HOLD array.

- continued -

SCLEAR	Sets the accumulator flag and internal stack counter to zero.
SUM	Adds a scan to the HOLD array (with user specified weighting.)

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
```

* Weight factor = integration time/system temperature^2

** 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 are used by procedures C1, C2, CB, A1, A2, and AB to stack scans.

ADD

A routine which inserts a series of scans into the STACK.

PTWH	Is not used or changed.

ADVERBS

0	ACOUNT	The STACK counter is incremented by the number of scans inserted.
0	ASTACK	The scan numbers are inserted into the STACK.
2	SINCR	Increment between first and last scan numbers.

OPERANDS	Beginning and end scan number.
TWO-RECEIVER OPTION	Does not apply.

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 filterbank of scans 500 to 600 type:

```
500 600 ADD   or   ADD (500,600)
C1
```

Scans 500 to 600 will be accumulated, averaged and displayed.
C1 (C2) will accumulate, average and display the first (second) filterbank.

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 data points at equal velocities or equal frequencies will be added together. ALIGN sets the values of the adverb ASHIFT, which are used by the verbs ACCUM and SUM.

PTWH Does not affect ALIGN. Is set to the HOLD array.

ADVERBS

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

TWO-RECEIVER PROCESSING OPTION Does affect ALIGN. The value of ASHIFT will be computed for either or both receivers depending on the processing option.

RELATED VERBS

ACCUM	Adds the contents of the WORK array to the contents of the HOLD array with weighting by integration time divided by system temperature squared. Used to stack scans.
ALIGNF	Sets the option for ALIGN to align frequencies.
ALIGNV	Sets the option for ALIGN to align velocities.
AVE	Divides the accumulated scans in the HOLD array by the sum of their integration times divided by system temperature squared.
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 specified by the user.
TELL CSTACK	Prints the internal stack counter and the scan numbers of the scans accumulated.

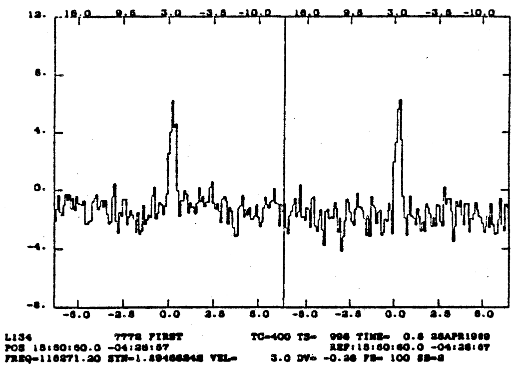
REMARKS The alignment for the first scan to be accumulated is always 0. The default alignment for ALIGN is frequency.

- continued -

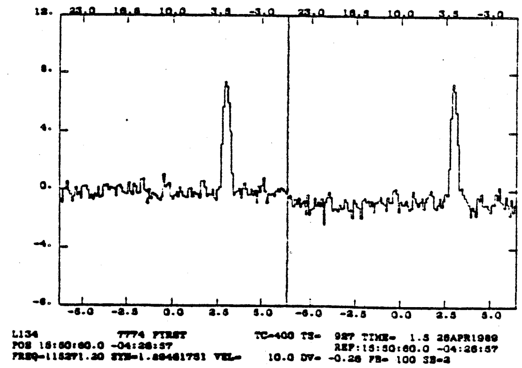
EXAMPLE

You have the following two scans:

7772



7774

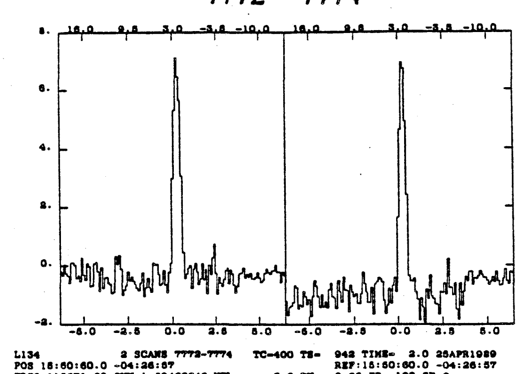


You want to average them together. Specify:

`SCLEAR`	To zero the stack counter in the header of the `HOLD` array.
`ALIGNV`	Specify alignment of velocities.
`GET 7772 ACCUM`	To copy scan 7772 to the `WORK` array weight it by its integration time/TSYS^2.
`GET 7774 ALIGN ACCUM`	To add 7774 correctly to 7772, with weighting by its integration time/TSYS^2.
`AVE PAGE SHOW`	To divide the weighted scans 7772 and 7774 by the sum of their weights and display the result.

You will see

7772 - 7774



ALIGNF ALIGNV

ALIGNF sets an internal flag so that the ALIGN verb will align frequencies.
ALIGNV sets an internal flag so that the ALIGN verb will align velocities.

Frequency is the default alignment parameter for ALIGN.

See ALIGN

PTWH	Is not used or changed.
ADVERBS	None.
TWO-RECEIVER OPTION	Does not apply.

RELATED VERBS

ALIGN	Sets the values of ASHIFT so that when the scan in the WORK array is accumulated, its frequencies or velocities will be correctly aligned with the frequencies or velocities of the HOLD array.
ACCUM	Adds the contents of the WORK array to the contents of the HOLD array with weighting by integration time divided by system temperature squared.
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 specified by the user.

EXAMPLE

See ALIGN verb.

ARRAY

This routine defines an array. It requires as its object the name of the array that is to be constructed and the dimensions of the array. The dimensions of the array may be specified in two ways:

(SCALAR) or (SCALAR TO SCALAR)
(SCALAR, LIMS) or (SCALAR TO SCALAR, LIMS).

SCALAR may be either a constant or a variable which has been assigned a value. LIMS may be a constant.

PTWH Is not used or changed by ARRAY.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not affect ARRAY.

RELATED VERBS None.

REMARKS

This verb 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 you could define the following procedure (you are using two receivers):

```
PROCEDURE RMSCALC(FSCAN,LSCAN,N)
ARRAY RESULTS (2*N)
FOR I = FSCAN TO LSCAN
        GET (I)
        RMS
        RESULTS (I) = VRMS(1)
        RESULTS (I+N) = VRMS(2)
        END
PRINT RESULTS
FINISH
```

AUTO

AUTO returns the control of the Y-scaling of its graphs to the verb SHOW. SHOW normally determines the Y-scaling of its graphs, so AUTO is used to restore control of the Y-scaling to the program after the user has overridden it by means of HOLDY or RANGE or setting the values of the adverbs YMIN and YINCR.

PTWH Is not used or changed.

ADVERBS

YMIN is set to -90000. SHOW sets its own Y-scaling if YMIN $\leq$ -9999.

0 YINCR	The increment in Kelvin degrees between the 6 tic marks on the Y-axis.

TWO-RECEIVER PROCESSING OPTION Does affect AUTO. It is possible to free the Y-scaling of either or both receivers.

RELATED VERBS

HOLDY	Causes SHOW to use the last Y-scaling it used again. Does this by setting YMIN to -9E10.
RANGE (#,#)	Sets the scaling for SHOW by using the minimum and maximum values input.
SHOW	Plots the contents of the currently referenced array on the CRT; draws and labels the axes.

EXAMPLE

See FREEY

REMARKS AUTO is an alias for FREEY.

AVE

AVE divides the accumulated scans in the HOLD array by the accumulated weighting factor (sum of individual integration times/individual squares of system temperatures). The result is left in the HOLD array. AVE also zeros the stack counter (accumulator).

PTWH Does not affect AVE. Is set to the HOLD array.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Should be BOTH when AVE is specified.

RELATED VERBS

ACCUM	Adds a scan to the HOLD array (with weighting by integration time/system temperature^2).
SCLEAR	Clears the accumulator(HOLD) array before stacking data.
SUM	Adds a scan to the HOLD array (with user specified weighting).
TELL CSTACK	Prints the stack counter and the scan numbers of the scans accumulated.

ERRORS

If the stack is empty when AVE is specified, the message AVE EMPTY? will appear. This tells the user that the stack counter is zero.

- 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
```

AVGD

Computes the mean source temperature and standard deviation of the mean of continuum digital backend data in the currently referenced array.

See SWITCHED, TOTALPWR

EXAMPLE

```
CGET 500 SWITCHED AVGD PAGE SHOW
```

AZ

Puts the value of the commanded azimuth of the source or position specified of the data in the currently referenced array on the value stack.

BACKEND

Puts a field of eight characters that describes the backend used into the adverb SBACKEND. This adverb may then be printed.

EXAMPLE: To query the backend used in an observation type

```
BACKEND;  PRINT SBACKEND
```

BADCHV, GAINS

BADCHV	Puts the value of the antenna temperature to be assigned to those filterbank channels, noted defective, of the data in the currently referenced array on the value stack.
GAINS	For total power data, GAINS in the same header location refers to the scan number of the calibration data to be applied to the scan in the currently referenced array.

BASELINE

BASELINE is used to fit and remove polynomial frequency baselines from the data. This routine computes the coefficients of a Chebyshev polynomial of a specified order by a least squares fit for a specified region or regions of the scan in the currently referenced array. The polynomial is then evaluated for each channel and subtracted from the scan.

PTWH

BASELINE computes and subtracts a polynomial baseline from the scan in the array at which PTWH is pointing. BASELINE does not change PTWH.

ADVERBS

1	NFIT	The order of the polynomial baseline to be removed. NFIT must be between 1 and 12. Example: 3 NFIT
0 0	BDROP(RCVR) EDROP(RCVR)	The number of channels at each end of each receiver which will be ignored by the routine.
0	NREGION(RCVR,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.
50 50	BBASE(RCVR) EBASE(RCVR)	The number of channels at each end of each receiver (not including the channels dropped by BDROP and EDROP) which will be used to compute the baseline coefficients if NREGION(RCVR,1) = 0.

TWO-RECEIVER PROCESSING OPTION Does affect BASELINE. A baseline may be removed from either or both receivers.

RELATED VERBS

BSHAPE	Computes the coefficients of the Chebyshev polynomial for specified spectral values by a least squares fit.
BMODEL	Computes the channel by channel values of the baseline using the last computed coefficients of the Chebyshev polynomial.

See also

DCBASE RIPPLE RMODEL RSHAPE

- continued -

REMARKS

The regions used to compute the baseline should not include a known or suspected spectral feature.

ERRORS

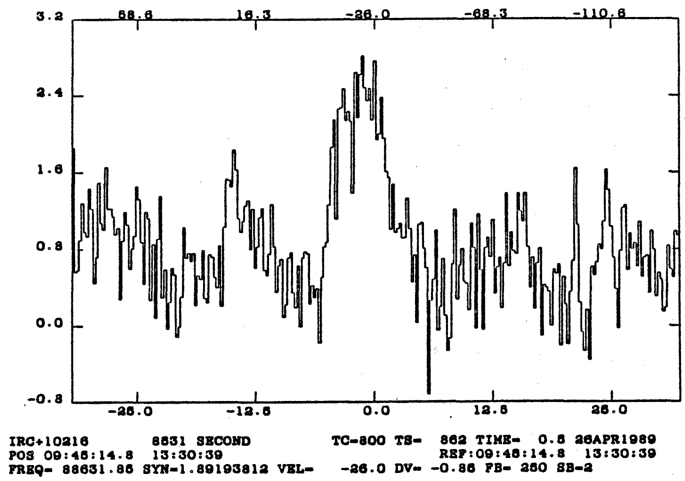
If `NREGION(RCVR,N)`, where N is even, is less than `NREGION(RCVR, N-1)`, the message

NREGION ?

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

EXAMPLES

(1) Your one-receiver data looks like

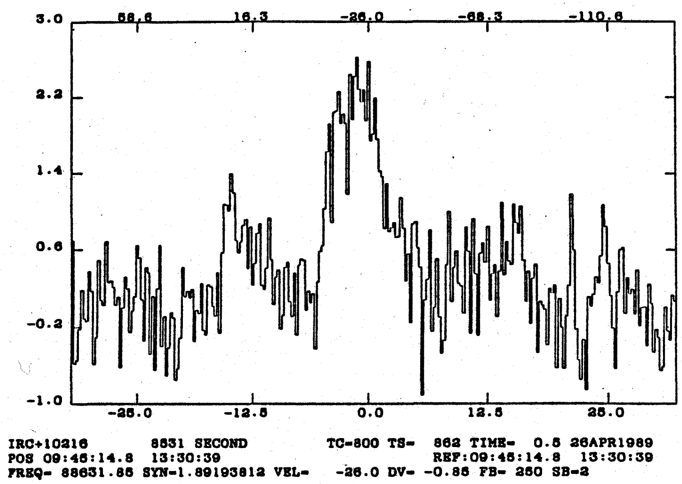


You want to remove a second order baseline. Specify:

```
NREGION (1,1) = 0
BBASE = 50
EBASE = 50
NFIT = 2
BASELINE PAGE SHOW
```

- continued -

You will get



(2) If you had two-receiver data, but both receivers were essentially identical, you would use the same procedure.

BASEOFF

Puts the value of the baseline offset of the data in the currently referenced array on the value stack.

BFWHM

Puts the value of the beam full width at half maximum of the data in the currently referenced array on the value stack.

BIAS

BIAS adds the values of FACT(RCVR) to the scan in the currently referenced array.

PTWH — Determines to which scan FACT is added. Is not changed.

ADVERBS

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

TWO-RECEIVER PROCESSING OPTION — Does affect BIAS. FACT may be added to either or both receivers. (If both are biased, they may be biased by different values.)

RELATED VERBS

SCALE — Also uses the adverb FACT. SCALE multiplies each receiver of the currently referenced array by FACT(RCVR).

- continued -

EXAMPLE

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

```
GET 8530 PAGE SHOW
```

Set YMIN and YINCR so that the first scan will be plotted on the top of the graph. For the graph below,

```
-2      10 RANGE
```

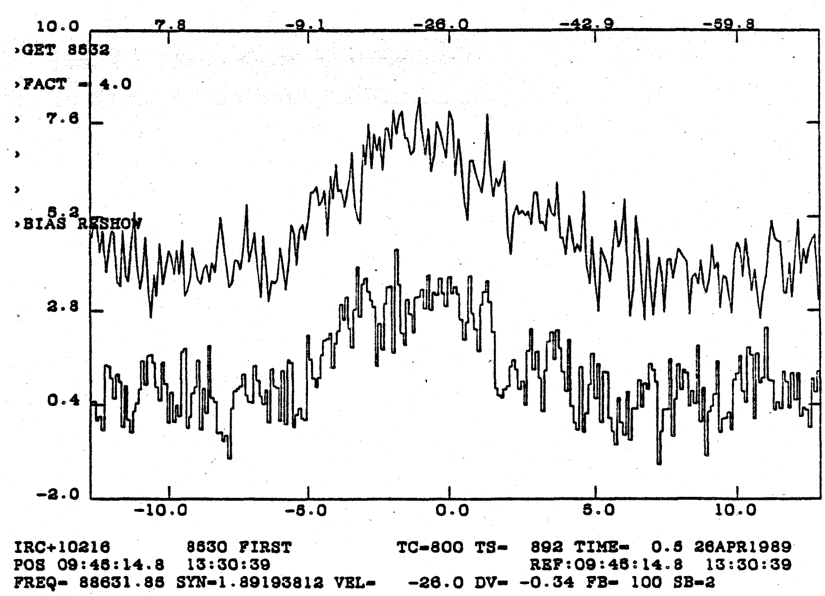
was used. Replot. Then set FACT to separate the scans

```
PAGE SHOW
GET 8532
FACT = 4.0
```

and specify

```
BIAS RESHOW
```

The results look like



BMODEL

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

PTWH — Determines which array is replaced by the baseline model. Is not changed.

ADVERBS — None used directly -- BMODEL uses the results of the previous BASELINE or BSHAPE.

TWO-RECEIVER PROCESSING OPTION — Does affect BMODEL. Either or both receivers may be replaced by a baseline model depending on the processing option.

RELATED VERBS

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.

REMARKS

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 710 TRT BSHAPE BMODEL PMW DIFF SHOW
GET 712 DIFF PAGE SHOW
GET 714 DIFF PAGE SHOW
.
.
.
GET 748 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.

BMORENT

Not currently implemented

BMTHROW

Puts the value of the beam throw parameter of the data in the currently referenced array on the value stack. This is not implemented as yet. Instead the wavelength (focus step size) used in a continuum `FOCALIZE` is stored in this location and may be retrieved by the verb `WL`.

BOTH

BOTH sets the two-receiver processing option of the scan in the currently referenced array to both receivers. (Both receivers is the default processing option, so BOTH is used to restore the default after ONLYA or ONLYB has been used.)

PTWH Determines which scan's processing default is changed. Is not changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Is set by BOTH to both receivers.

RELATED VERBS

ONLYA Sets the two-receiver processing option to only the A receiver.

ONLYB Sets the two-receiver processing option to only the B receiver.

REMARKS BOTH should not be used on one-receiver data.

EXAMPLE

You have displayed the A receiver by

```
GET 710 ONLYA PAGE SHOW
```

And now you want to specify both receivers. To process both receivers, specify

```
BOTH PAGE SHOW
```

BOXCAR

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

PTWH	Determines which scan is smoothed. Is not changed.

ADVERBS

0	BDROP(RCVR)	The number of channels at each end of each receiver which
0	EDROP(RCVR)	will be ignored by the routine.
0	NBOX	The number of channels which will be averaged together to smooth the scan.

TWO-RECEIVER PROCESSING OPTION Does affect BOXCAR. Either or both receivers may be smoothed. If both are smoothed, they are smoothed in the same way, i. e., NBOX cannot be specified separately for each receiver.

RELATED VERBS

HANNING	A smoothing routine which averages three channels together with the center channel getting twice as much weight as either side channel.
SMOOTH	A smoothing routine which averages channels according to user specified weighting.

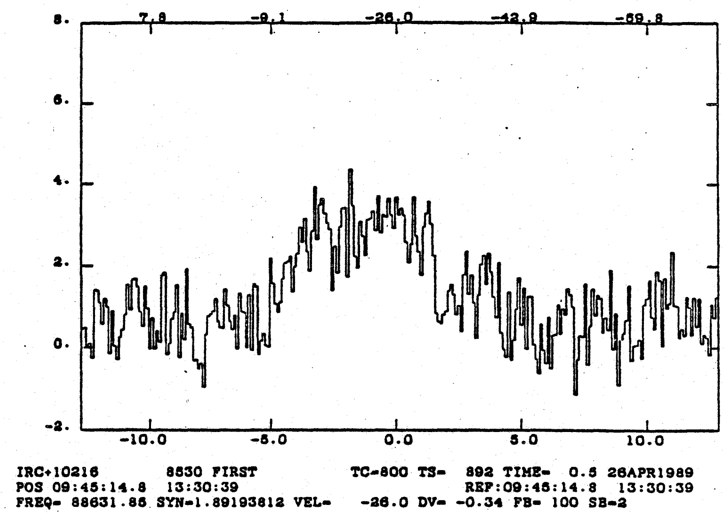
ERRORS

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

- continued -

EXAMPLE

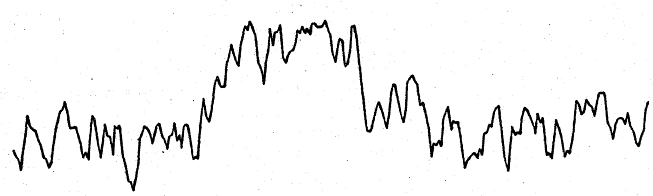
Your data looks like



And you want to smooth it. If you specify

```
NBOX = 3
BOXCAR PAGE RESHOW
```

You will get

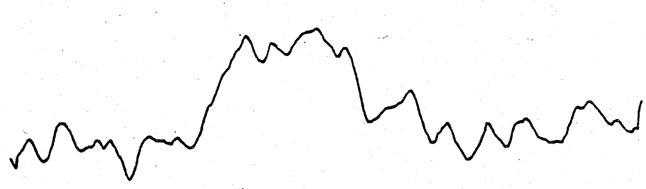


(**RESHOW** does not label the axes.)

If you specify

```
NBOX = 7
BOXCAR PAGE RESHOW
```

You will get



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	BSHAPE computes the Chebyshev coefficients for the scan in the array at which PTWH is pointing. BSHAPE does not change PTWH.

ADVERBS

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

TWO-RECEIVER PROCESSING OPTION Does affect BSHAPE. The coefficients will be computed for either or both receivers depending on the processing option.

RELATED VERBS

BASELINE	Not only computes the coefficients but evaluates them for each channel 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 712, which is in the WORK array, but before you subtract the parabola from the scan, you want to see how well it fits. You specify

TRT BSHAPE BMODELTo copy the scan to the TEMP array, compute the polynomial coefficients, and replace your copy of the scan with the computed polynomial.

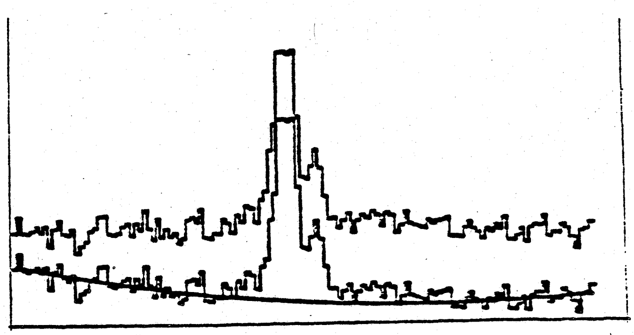
RESHOW To plot the baseline model over the scan already plotted on the screen.

PMW DIFF RESHOWTo subtract the model from the data in the WORK array and plot the results over the original scan and the baseline model.

When finished, you will have something like this:

Example:

```
-2  10 RANGE
1186 S
TRT BSHAPE BMODEL
RESHOW
PMW DIFF RESHOW
```



BSHOW

Fits, evaluates and displays a polynomial frequency baseline without altering data in the current array.

Other relevant parameters or adverbs, with initial value (if any):

1	`NFIT`	Order of polynomial baseline to be removed.
0	`BDROP(r)`	# channels at beginning of data to be ignored.
0	`EDROP(r)`	# channels at end of data to be ignored.
0	`NREGION(r,n)`	Region(s) of data where baseline is computed.
50	`BBASE(r)`	# channels at each end of data excluding BDROP
50	`EBASE(r)`	&EDROP where baseline is fit if NREGION(r,1)=0

BUGREPORT

To report a problem with the analysis system, type `BUGREPORT` (or just `BUG`), and you will then be able to use the standard VMS EDT editor to describe the problem. Please give as much useful information as possible, including specific information (scan numbers, observing mode etc.)

When you have finished typing your report, enter `CONTROL-Z` and `EX` to exit from the editor and return to the analysis program.

BW

Puts the value of the bandwidth (total bandwidth of filterbank) of the selected receiver of the data in the currently referenced array on the value stack. The units are MHz.

BY

Used in procedures in `FOR`, `TO`, `BY` logical construction.

Example: `FOR X = 1 TO 3 BY 2; ...; END;`

CCUR

This routine activates the vertical crosshair and returns the channel 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.

TWO-RECEIVER PROCESSING OPTION Will compute proper channel for either receiver A, receiver B or both when both are displayed together.

RELATED VERBS

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

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

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

REMARKS CCUR is best used in a procedure, generally for defining baseline regions, bad points to be removed by REPLACE or initial guesses for fitting gaussians.

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
```

CGET

CGET retrieves the continuum data from disk within the LINE program. Continuum FIVE points, TIPs, and FOCALIZEs processed in the LINE program use CGET to retrieve the raw data. An error occurs if GET is used to retrieve a continuum scan or if CGET is used to retrieve a LINE scan. CGET takes one object, the desired scan number.

PTWH Is not used by CGET, but is set to the WORK array.

ADVERBS None.

OPERANDS Specify 1 or 2 for first or second filterbank.

TWO-RECEIVER PROCESSING OPTION Does not affect CGET. After GET, the scan in the WORK array will have the default processing option.

RELATED VERBS

See also GET

REMARKS

CGET does not require the scan number as its object. Its object is 1 or 2.

ERRORS

If there is no current scan, or if the current scan has not finished one pair, the message

SCAN ?

will be generated.

EXAMPLE

To bring the first filterbank of the current scan into the WORK array and display it, you specify

```
CGET 1 PAGE SHOW or Q1
```

CHAN

Puts the value of the specified data channel of the data in the currently referenced array on the value stack.

EXAMPLE PRINT CHAN 31

CHAR

Prints a string of characters on the CRT at the current cursor position. Used in procedures. Requires two objects.

Other relevant parameters or adverbs, with initial value (if any):

	STRING	Statement to be printed. Typed within single quotes.
	#CHARACTERS	# of characters in string.

EXAMPLE CHAR ('HELP',4)

CHECK

Prints out the scan number (if valid) of data in the current NSAVE bin.

Other relevant parameters or adverbs, with initial value (if any):

0	NSAVE	The number of the disk bin whose data are checked.

CHGRES

Smoothes spectral line data from its initial resolution to a lower resolution specified by NEWRES. The routine steps a gaussian of appropriate FWHM through the existing spectrum and performs a convolution at each position. The scan frequency resolution and delta velocity are updated.

Other relevant parameters or adverbs, with initial value (if any):

0	NEWRES	The desired frequency resolution.

CLIP

CLIP resets any spectral value in the scan in the currently referenced array that is greater than CLIPMAX to CLIPMAX and any spectral value that is less than CLIPMIN to CLIPMIN.

PTWH Determines which scan is clipped. Is not changed.

ADVERBS

CLIPMAX	The maximum spectral value that will be found in a scan after it is clipped.
CLIPMIN	The minimum spectral value that will be found in a scan after it is clipped.

TWO-RECEIVER PROCESSING OPTION Does affect CLIP. Either or both receivers may be clipped depending on the processing option.

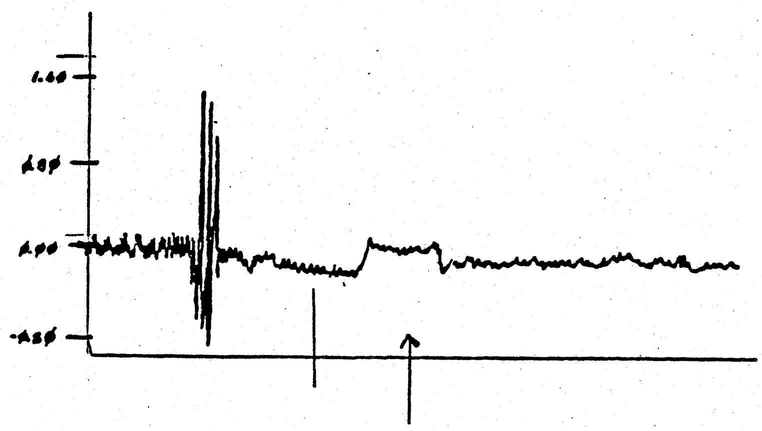
REMARKS

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

- continued -

EXAMPLE

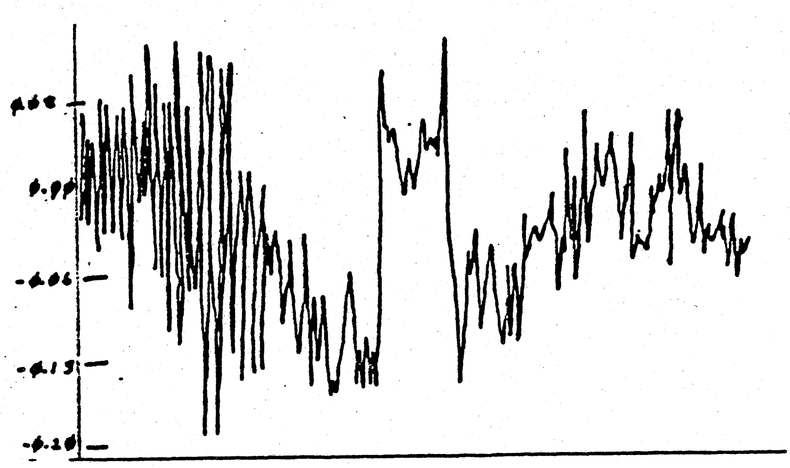
Your data will look like



You are interested in this structure. Specify

```
CLIPMIN = 0.18;  CLIPMAX = 0.14;  CLIP PAGE SHOW
```

You will get



COMMENT

Places the cursor at the bottom of the CRT screen, waits for the user to type in a remark no longer than a single line, then moves the cursor to the top of the screen. The verb COMMENT does not appear on the hardcopy plot.

COORDCD

Puts a field of eight characters which specifies in which coordinate system the data in the currently referenced array was commanded into the adverb SCOORDCD. This adverb may then be printed.

CORE

CORE prints the amount of memory available for programs, variables and source listings.

PTWH Is not used or changed by CORE.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not affect CORE.

RELATED VERBS

RESTART	Empties the procedure and variable space.
STORE page #	Stores the current procedures in disk page bin page# ($1 \leq$ page# ≤ 6).
RESTORE page#	Retrieves disk page bin N.

REMARKS

CORE is purely an informational verb. You can avoid getting the message BLEW CORE by not trying to define a procedure when there is very little space left in memory.

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

PROGRAM	VARIABLES	SOURCE
2997 4003	3729 4189	1225 2048

This tells you that you have 1006 words of program space, 460 words of variable space and 823 words of procedure source space left.

CROSSHAIR

This routine activates the horizontal and vertical crosshairs. Striking any key except the RETURN keys causes the intersection of the crosshairs to be marked with a '+' and the channel number, velocity and temperature at that point to be labeled. After the information is printed, the crosshairs are prompted again. Striking the '=' key will deactivate the crosshairs. Do not strike the RETURN key when using `CROSSHAIR`.

PTWH Is not used or changed by `CROSSHAIR`.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does apply. Will only compute the proper velocity for one receiver at a time.

RELATED VERBS

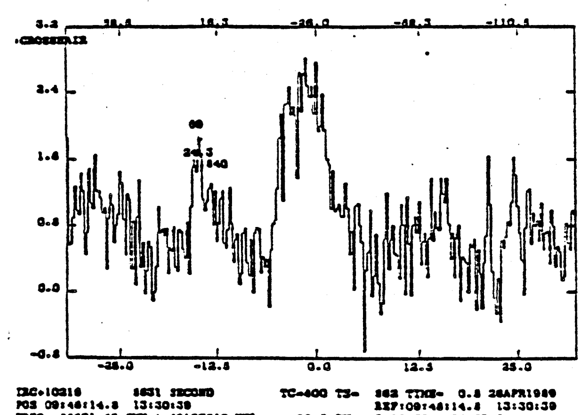
- `CCUR` Returns the channel value at the current vertical crosshair position.
- `TCUR` Returns the temperature value at the current horizontal crosshair position.
- `VCUR` Returns the velocity value at the current vertical crosshair position.

REMARKS

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

EXAMPLE

An example of the output from `CROSSHAIR` is shown below.



CRT

Directs all non-graphic output from the printer back to the CRT.

See PRINTER

EXAMPLE `CRT`

CSORT

Sorts the continuum data in the observer's data file and fills the stack (ASTACK) with scan numbers that meet the sort requirements specified. One or more of the adverbs listed below may be used.

Other relevant parameters or adverbs, with initial value (if any):

BSCAN	Beginning scan number for the sort.
ESCAN	Ending scan number for the sort.
SOURCE	Source name used for the sort.
FREQ(2)	Beginning and ending rest frequency of sort.

CYCLLEN

Puts the value of the cycle length in seconds (time required to complete one cycle) of the data in the currently referenced array on the value stack.

DATALEN

Puts the value of the data length (in R*4 words) of the scan in the currently referenced array on the value stack. This value may be printed or used in calculations. To find the total number of data points in the scan in the currently referenced array type:

```
PRINT DATALEN
```

DATALOG

Prints on the CRT a listing of all spectral line scans in the observer's raw data file, giving the scan number, source name, velocity and rest frequency for each scan. A listing of all continuum scans follows with each scan number, source name and rest frequency.

DCBASE

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

PTWH	Determines which scan gets the average computed and subtracted. Is not changed.

ADVERBS

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

TWO-RECEIVER PROCESSING OPTION	Does affect DCBASE. The average will be computed and subtracted from either or both receivers depending on the processing option.

RELATED VERBS

See BASELINE
BIAS

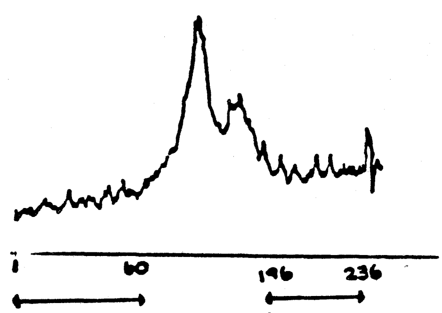
- continued -

REMARKS

A conservative means of frequency baseline removal.

EXAMPLE

Your data looks like



You want to remove an average computed over these regions

So you specify

```
EDROP = 20
NREGION (1,1) = 0
BBASE = 60
EBASE = 40
DCBASE
PAGE SHOW
```

or (for receiver A only)

```
NREGION (1,1) = 11
NREGION (1,2) = 70
NREGION (1,3) =133
NREGION (1,4) =172
NREGION (1,5) =  0
DCBASE PAGE SHOW
```

DCL

Temporarily leaves the program, putting you into the Vax DCL environment. You may then give any DCL commands, in the normal way. To return to the analysis program, at the point you left it, type LO (to logoff from the spawned task).

DEBUG

When followed by the object, TRUE, causes the LINE program to print diagnostic information as commands are executed. To turn off the diagnostics, type DEBUG FALSE.

DEC

DEC sets an internal flag that causes MAPSHOW to use declination as the Y-coordinate for maps.

PTWH	Is not used or changed.
ADVERBS	None.
TWO-RECEIVER OPTION	Does not apply.

RELATED VERBS

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

DELETE

A routine which deletes one scan from the STACK.

PTWH	Is not changed or used.
ADVERBS	
ACOUNT	The STACK counter is decremented by one.
ASTACK	The specified scan is deleted from the STACK array.
OPERANDS	The number of the scan to be deleted from the STACK.
TWO-RECEIVER OPTION	Does not apply.
RELATED VERBS	
A #	Inserts one scan into the STACK.
ADD (#,#)	Inserts a series of scans 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. The scan number is specified positive regardless of the sign of the scan number in the STACK.

EXAMPLE

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

```
524     DELETE
```

DELTAV

Calculates area, mean flux, high & low velocity peak fluxes, high & low peak velocities and delta velocities at specified fractions of peak flux.

Other relevant parameters or adverbs, with initial value (if any):

0	DFRACT	Increment between fractions for velocity.
0	NFRACT	Number of fractions desired.
0	FRACT	Lowest fraction for velocity calculation.
0	BMOMENT(r)	Beginning channel of velocity window.
0	EMOMENT(r)	Ending channel of velocity window.

DELTAX

Puts the value of the step-size along the x-axis (usually velocity) for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

DELTAXR

Puts the value of DELTA X or X RATE of the data in the currently referenced array on the value stack. The cell size or distance (in seconds of arc) between cells on the X axis. DELTAXR is used to specify the horizontal grid cell size in the mapping routines. See Mapping and GRID.

DELTAYR

Puts the value of the cell size or distance between cells on the Y axis of the data in the currently referenced array on the value stack. DELTAYR is used to specify the vertical grid cell size in the mapping routines. See Mapping and GRID.

DESORG

Puts the value of the descriptive origin (horizontal position, vertical position, position angle describing the orientation of the sky "horizontal axis") of the data in the currently referenced array on the value stack.

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. Is not changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does affect DIFF. Either or both receivers may be differenced, depending on the value of the processing option.

REMARKS

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

EXAMPLES

To difference a scan 710 and an off-the-source scan 712, and display the result

```
OFF 712 ON 710 DIFF PAGE SHOW
```

DIR

Invokes the VMS DCL command DIRECTORY which causes the contents of the current directory to be listed on the screen.

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. The currently referenced array must be in the TEMP array.

PTWH Determines which array is divided by the `TEMP` array. Is not changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does affect `DIVIDE`. Either or both receivers may be divided, depending on the processing option.

RELATED VERBS

`OFF` Copies a scan from the disk to the `TEMP` array.

`ON` Copies a scan from the disk to the `WORK` array.

EXAMPLE

To divide one scan by another:

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`
and	`DIVIDE` `PAGE SHOW`

to produce and display the result.

DOC

Writes one line of header information about the scan in the currently referenced array. Scan number, date, LST, position, length of scan, TS, filterbank, frequency and source name are printed. See `TITLE`, `HEADER`, `SNAP`.

EDIT

EDIT is used to change an already defined procedure. 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. EDIT puts the program into EDIT mode and signals this to the user by prompting with a colon instead of a carat.

PTWH Is not used or changed by EDIT.

ADVERBS None.

OBJECTS

Procedurename	The name of the procedure to be edited.
Linenumber	The line number of the line in the procedure which is to be changed.

TWO-RECEIVER PROCESSING OPTION Does not affect EDIT.

RELATED VERBS

LIST procedurename	Lists the procedure with the given name.
ENDEDIT	Gets the program out of EDIT mode.

See also PROCEDURE.

EXAMPLE

To edit the already defined procedure PRCDE

first list it

```
>LIST PRCDR

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

- continued -

Suppose you want to change HANNING to BOXCAR

```
EDIT PRCDE 4
:BOXCAR
:ENDEDIT
>
```

Now suppose you wanted to add a line to your procedure

```
EDIT PRCDR 2.5
:NBOX=3
:ENDEDIT
>
```

The corrected procedure looks like

```
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 by

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

EDITING

Specifies whether or not the edited file should be searched before the raw data file for scan retrieval.

`EDITING` takes one object

`TRUE` or `FALSE`

EXAMPLE `EDITING TRUE`

EDT

Invokes the VMS editor. Do not use any arguments with `EDT`. The editor will prompt for the name of the file you wish to edit. Upon exiting from the editor, control will be passed back to the LINE program. `EDT` is very useful for creating procedures in .PRC files outside the LINE program.

EL

Puts the value of the commanded elevation of the source or position specified of the data in the currently referenced array on the value stack.

ELSE

Used in procedures in `IF`, `THEN`, `ELSE` logical constructions. `ELSE` (do this) need not be included when no action is desired if the test is false.

See `IF`, `END`, `FOR`, `WHILE`

EXAMPLE `IF X = 2 THEN; Y = 3; ELSE Y = 1; END`

EMPTY

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

PTWH	Is not used or changed.

ADVERBS

ACOUNT	The STACK counter is set to zero.
ASTACK	The STACK array is zeroed.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

A #	Inserts a single scan into the STACK.
ADD (#,#)	Inserts a group of scans into the STACK.
DELETE #	Deletes a scan from the STACK.
STACK	Lists all scans 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 Is not used or changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not affect END.

RELATED VERBS

See FOR
IF
WHILE

EXAMPLE

```
FOR I = 1 TO 3
...
...
END
```

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 carat.

PTWH Is not used or changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not affect ENDEDIT.

RELATED VERBS

See EDIT	Used to change an already defined procedure.
LIST	Produces listing of the requested procedure.

EXAMPLE

```
EDIT XMPL 3
:...
:...
:ENDEDIT
>
```

EPOCH

Puts the value of epoch (as specified by the observer) of the data in the currently referenced array on the value stack.

EPOCRA

Puts the value of the epoch right ascension of the source or position specified of the data in the currently referenced array on the value stack.

EPODEC

Puts the value of the epoch declination of the source or position specified of the data in the currently referenced array on the value stack.

ESTACK

Lists all record numbers currently in the `RSTACK`.

Other relevant parameters or adverbs, with initial value (if any):

0	`RSTACK(n)`	Array dimensioned to 20 for record numbers.
0	`RCOUNT`	Counter of number of records in `RSTACK`.

EXEDIR

Invokes the VMS DCL command `DIRECTORY *.EXE` which lists all files in the main directory with the extension `.EXE`.

EXIT

EXIT is a routine that terminates the LINE program and returns control to VMS.

PTWH	Is not used or changed.
ADVERBS	None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

STORE	Copies the program memory into a disk storage space called a "page" for later retrieval.
RESTORE	Restores the program memory previously saved on disk by the store command.

EXAMPLE

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

```
EXIT
```

The terminal will respond with a message

```
FORTRAN STOP
```

EXPLAIN

Documents the functions of all NRAO defined procedures permanently residing in the program.

EXAMPLE

`EXPLAIN HALVES`

(Replaced by HELP PROCEDURES)

FETCH

Retrieves total power raw data from disk. A scan number is required as its object. `FETCH` copies the desired `ON` scan from disk to the `WORK` array, then copies the appropriate `OFF` scan into the `TEMP` array, the `ZERO` levels into the `HOLD` array and the `GAINS` into the `SPECT` array. `TEMP` may calculate a calibrated quotient after these data are loaded.

FFT

Performs a fast Fourier transform of the data in the currently referenced array.

PTWH	Determines which array is transformed. Is not changed.

ADVERBS

0 BDROP(RCVR) 0 EDROP(RCVR)	The number of channels at each end of each receiver which are ignored by the routine.
0 INVFFT	Flag that denotes if an inverse Fourier transform is to be done. If so, INVFFT = 1.

TWO-RECEIVER PROCESSING OPTION	Does affect FFT. Either or both may be processed, depending on the processing option.
RELATED VERBS	None.

FINISH

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

PTWH Is not used or changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not affect FINISH.

RELATED VERBS

PROCEDURE Begins the definition of a procedure.

REMARKS

FINISH must appear on a line by itself.

EXAMPLE

```
PROC XMPL
XX
FINISH
```

FIRSTIF

Puts the value of the FIRST-IF of the data in the currently referenced array on the value stack.

FLAG

This routine will draw a vertical flag perpendicular to the horizontal frequency axis at the specified frequency on the current display and labels the frequency.

PTWH — Is not used or changed by `FLAG`.

ADVERBS

`FMARK` — Frequencies at which to draw a vertical line when a scan is displayed. Used by `SHOW`.

`OBJECT` — The frequency desired to be flagged specified in MHz.

TWO-RECEIVER PROCESSING OPTION — `FLAG` will work on either or both receivers simultaneously.

RELATED VERBS

`FULLGRID` — Draws a rectangular grid on the CRT at each horizontal and vertical tic mark.

`SHOW` — Plots contents of the currently referenced array on the CRT; draws and labels the axes.

REMARKS — `FLAG` will mark the specified frequency once. If the adverb, `FMARK`, is set by the user, each call to `SHOW` will flag the specified frequencies with a vertical line.

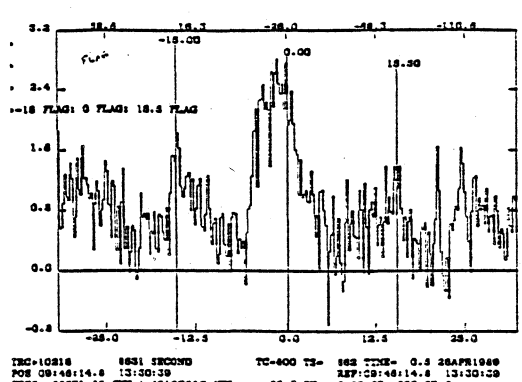
ERRORS — If `FLAG` is typed without specifying a frequency the error message "ARG LIST?" is printed.

EXAMPLE

To flag and label -10, -5, 5, and 10 MHz frequencies on the current display, specify

```
-15 FLAG;  0 FLAG;  15.5 FLAG
```

and you will get



FOCUSH

Puts the value of the horizontal (East-West) focus offset of the data in the currently referenced array on the value stack.

FOCUSR

Puts the value of the radial focus position of the data in the currently referenced array on the value stack.

FOCUSV

Puts the value of the vertical (North-South) focus offset of the data in the currently referenced array on the value stack.

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 completely 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 variables or arithmetic expressions or explicit numerical values. 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.

PTWH Is not used or changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

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
```

- continued -

See PROCEDURE

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
```

FRAME

Puts a field of eight characters in which the first four characters state whether the grid is polar (POLR) or cartesian (CART) and the second four state whether DELTAXR and DELTAYR refer to STEP sizes or SCANning rates of the data in the currently referenced array into the adverb, SFRAME. This adverb may then be printed.

FREEY

FREEY returns the control of the Y-scaling of its graphs to the verb SHOW. SHOW normally determines the Y-scaling of its graphs, so FREEY is used to restore control of the Y-scaling to the program after the user has overridden it by means of the verb HOLDY or setting the values of the adverbs YMIN and YINCR by the verb RANGE.

PTWH Is not used or changed.

ADVERBS

YMIN is set to -90000 by FREEY. SHOW sets its own Y-scaling if YMIN $\leq$ -9999.

TWO-RECEIVER PROCESSING OPTION Does affect FREEY. It is possible to free the Y-scaling of either or both receivers.

RELATED VERBS

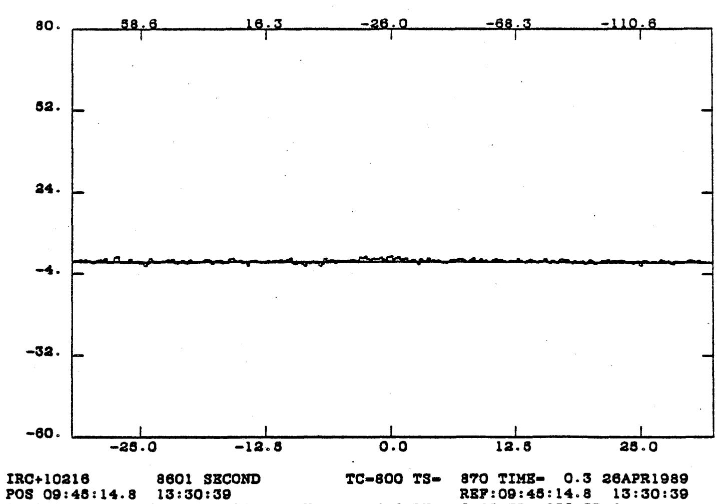
HOLDY Causes SHOW to use the last Y-scaling it used again. Does this by setting YMIN to -90000000000.

AUTO Alias for FREEY.

RANGE Sets YMIN and YINCR.

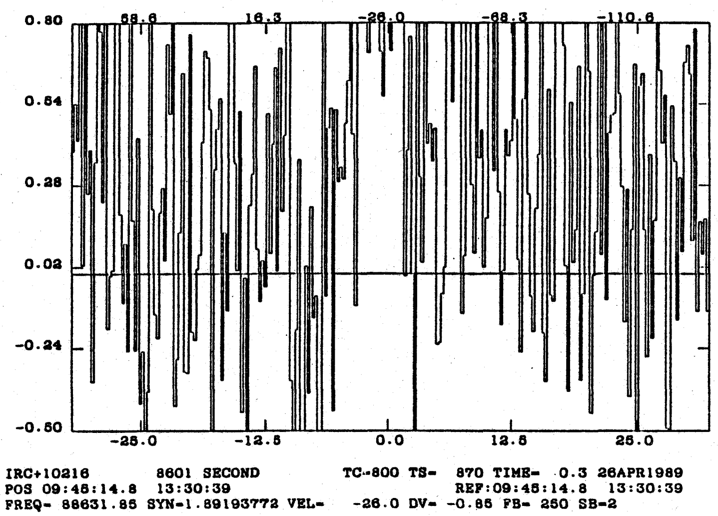
EXAMPLE

Your last graph looked like this



- continued -

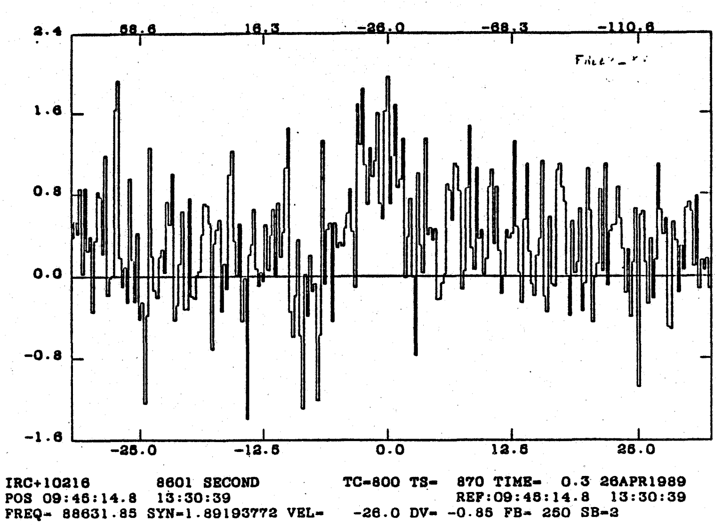
or this



To get a better graph, specify

FREEY PAGE SHOW
or
AUTO XX

You will get



FREQRES

Puts the value of the frequency resolution NT (bandwidth per channel) of the selected receiver of the spectral line data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

If the data is continuum, the value of NT, noise tube true or false, is placed on the value stack.

FRONTEND

Puts a field of eight characters to describe the receiver used into the adverb SFRONTEND. This adverb may then be printed.

EXAMPLE

To query the frontend used for an observation, type

```
FRONTEND;  PRINT SFRONTEND
```

FULLGRID

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

PTWH Is not used or changed by FULLGRID.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION A rectangular grid will be drawn for either receiver A, or receiver B or both when displayed together.

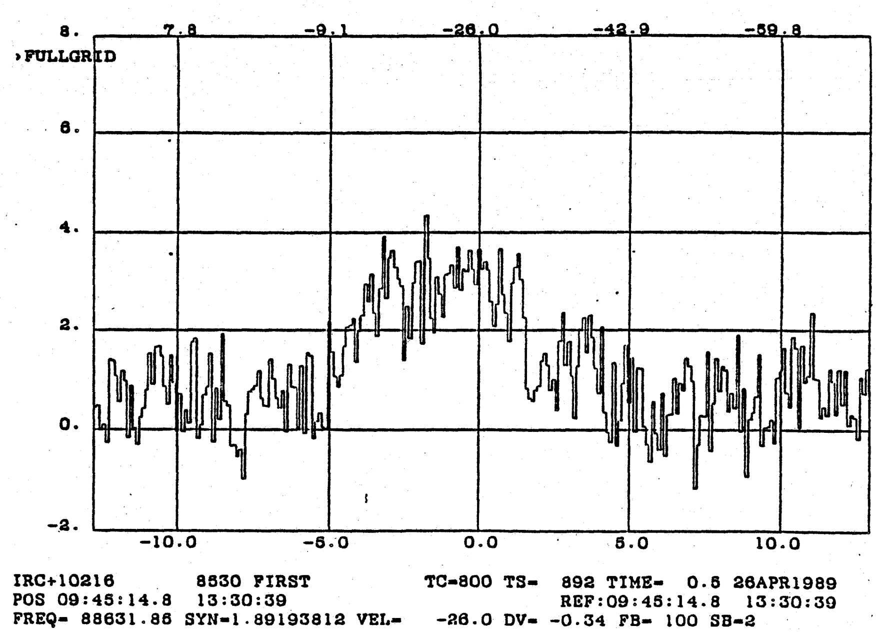
RELATED VERBS

FLAG Draws a vertical flag perpendicular to the horizontal frequency axis at the specified frequency on the current display.

SHOW Plots the contents of the currently referenced array on the CRT; draws and labels the axes.

EXAMPLE

In order to draw a rectangular grid on your current display, specify FULLGRID and you will get



FV or VF

A routine which sets the axis labeling mode so the SHOW displays velocity for the upper x-axis labeling and frequency for the lower x-axis labeling.

PTWH Is not used or changed.

ADVERBS None.

TWO-RECEIVER OPTION Does not apply.

RELATED VERBS

`VC` or `CV` Sets the axis labeling mode so that `SHOW` displays channel numbers for the upper x-axis labeling and velocity for the lower x-axis labeling.

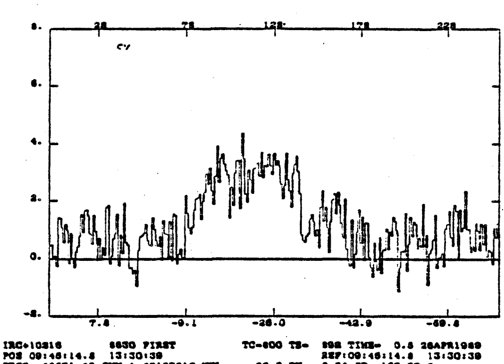
`SHOW` Plots contents of the currently referenced array on the CRT; draws and labels the axes.

REMARKS

The default labeling in `SHOW` for the lower x-axis is frequency and for the upper x-axis is velocity.

EXAMPLE

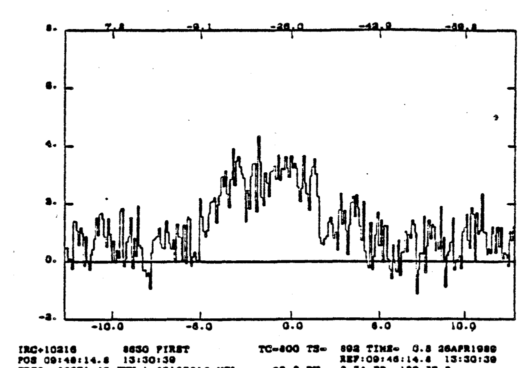
Your current display looks like



and you want to change your x-axes to frequency and velocity. Type:

FV XX

and you will get



GALLAT

Puts the value of the commanded galactic latitude of the source or position specified of the data in the currently referenced array on the value stack.

GALLONG

Puts the value of the commanded galactic longitude of the source or position specified of the data in the currently referenced array on the value stack.

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 carat 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

1	NGAUSS	The number of gaussians to be fit. NGAUSS = N, where $1 \leq N \leq 6$.
0 0	BGAUSS EGAUSS	The explicit channel numbers where the fit is to begin and end.
0	CENTER(N)	The channel number where gaussian N has its center. Iterated by the routine.
0	HEIGHT(N)	The height in degrees Kelvin of gaussian N. Determined by the routine.
0	HWIDTH(N)	The width in number of channels of gaussian N at half its peak height. Iterated by the routine.
8	NITER	The number of iterations which will be made to try to find a fit. The initial value of NITER is 8.
1	FIXC	If FIXC is less than zero, GAUSS will not iterate CENTER.
1	FIXHW	If FIXHW is less than zero, GAUSS will not iterate HWIDTH.

TWO-RECEIVER PROCESSING OPTION

GAUSS may only be used to fit gaussians in one receiver at a time. To fit lines in the A receiver, specify BGAUSS, EGAUSS, and all CENTER values equal to channels in the A receiver.

To fit lines in the B receiver, specify BGAUSS, EGAUSS, and all CENTER values equal to channels in the B receiver.

- continued -

RELATED VERBS

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 values of **HEIGHT** (in Kelvins), **CENTER** (in km/sec), and **HWIDTH** (in km/sec), plus the RMS error of the fit.

RESIDUAL Evaluates the parameters of the gaussian(s) for each channel and subtracts the total from the currently referenced array.

GPARTS Evaluates the parameters of each gaussian separately, constructs each gaussian, and displays it on the screen.

PEAK Finds the peak value and sets all the gaussian parameters accordingly for **GAUSS**.

REMARKS

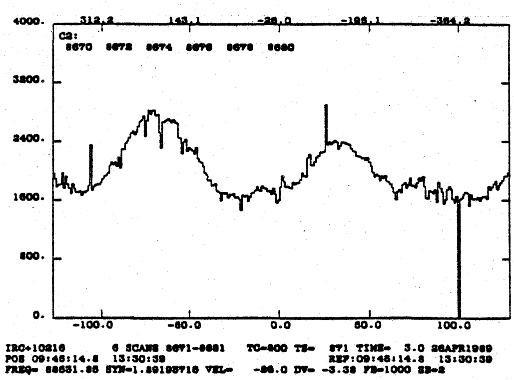
The frequency baseline should be removed before **GAUSS** is called. **GAUSS** is used to fit gaussians to data. The other gaussian verbs do not refine the parameters of the gaussian(s). GAUSS does not change the contents of any of the three arrays.

ERRORS

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

EXAMPLE

Your A receiver data looks like



You specify

NGAUSS=1; BGAUSS-105; EGAUSS=152; CENTER=129; HWIDTH=5
GAUSS

- continued -

All you get is a carat. RESHOW or SHOW will show you an unaltered spectrum. If you specify

PRINT CENTER (1) HEIGHT (1) HWIDTH (1)

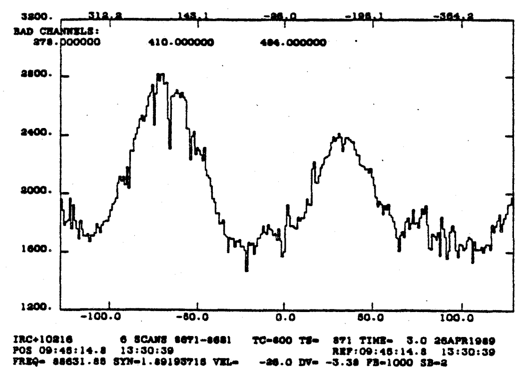
You will get

151.7371 28.6771 4.7104

(These refined values are in channels and Kelvins.)

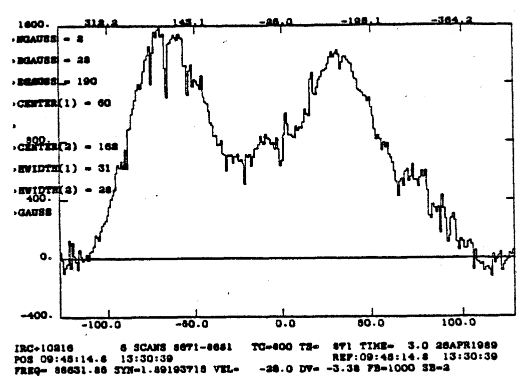
1. C2

   which gives stack: 8670 8672 8674 8676 8678 8680



2. BAD CHANNELS (BADCH)

   which gives: 278.000000 410.000000 484.000000

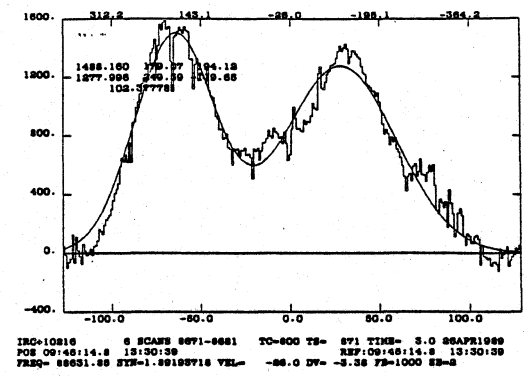


- continued -

3.

```
ZLINE=1
NREGION(1,1)=0
BBASE=10
EBASE=10
NFIT=2
BASELINE XX
```

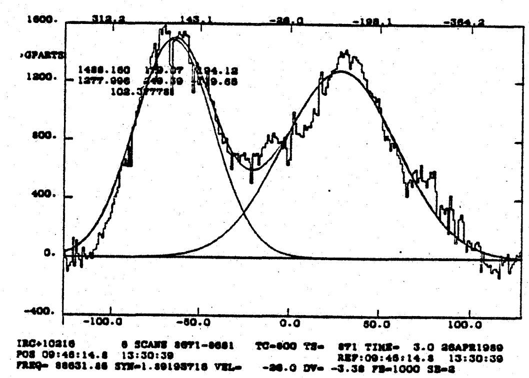
(which gives 4)



4.

```
NGAUSS=2
BGAUSS=28
EGAUSS=190
CENTER(1)=60
CENTER(2)=162
HWIDTH(1)=31
HWIDTH(2)=28
→ GAUSS
XX GSHOW
```

(which gives 5)



5. Shows result of 4

6.

PRINT CENTER(1)	HEIGHT(1)	HWIDTH(1)
63....	1488....	52....
PRINT CENTER(2)	HEIGHT(2)	HWIDTH(2)
156....	1277....	73....

GB

GB sets an internal flag that causes MAPSHOW to use galactic latitude as the Y-coordinate for maps.

PTWH Is not changed or used.

ADVERBS None.

TWO-RECEIVER OPTION Does not apply.

RELATED VERBS

DEC	Sets declination as the Y-coordinate for maps.
GL	Sets galactic longitude as the Y-coordinate for maps.
LABEL	Labels the map drawn by MAPSHOW.
MAPSHOW	Draws a map with the levels specified by LEVS.
RA	Sets right ascension as the Y-coordinate for maps.

GDISPLAY

Evaluates the parameters of NGAUSS gaussians and displays the sum without altering current array.

Other relevant parameters or adverbs, with initial value (if any):

1	NGAUSS	# of gaussian components.
0	BGAUSS	Beginning channel for gaussian evaluation.
0	EGAUSS	Ending channel for gaussian evaluation.
0	CENTER(n)	Center channel of each gaussian.
0	HEIGHT(n)	Heights of each gaussian.
0	HWIDTH(n)	Halfwidths of each gaussian.

GET

GET copies a raw spectral scan from disk to the WORK array. A scan number is required for its object.

See QGET, ON, OFF, TELL DISK

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.
TWO-RECEIVER PROCESSING OPTION	Does not affect GET. After GET, the scan in the WORK array will have the default processing option.

RELATED VERBS

CGET #	Copies the current scan into the WORK array.
ON #	Copies the requested scan into the WORK array.
OFF #	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.

GGET

Copies the current calibration array or specified calibration array from the disk to the WORK array. Specify 1 or 2 if the current calibration is desired or if a previous calibration is desired specify its scan number.

EXAMPLE `GGET 1`

GL

GL sets an internal flag that causes MAPSHOW to use galactic longitude as the Y-coordinate for maps.

PTWH Is not changed or used.

ADVERBS None.

TWO-RECEIVER OPTION Does not apply.

RELATED VERBS

DEC	Sets declination as the Y-coordinate for maps.
GB	Sets galactic latitude as the Y-coordinate for maps.
LABEL	Labels the map drawn by MAPSHOW.
MAPSHOW	Draws a map with the levels specified by LEVS.
RA	Sets right ascension as the Y-coordinate for maps.

Other relevant parameters or adverbs, with initial value (if any):

0	BMOMENT(r)	Beginning channel of window.
0	EMOMENT(r)	Ending channel of window.
0	GMFRAC	% mean flux to window search for high/low peak.

GMEASURE

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

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 values for HEIGHT(N) (in degrees Kelvin), CENTER(N) (in km/sec), HWIDTH(N) (in km/sec) and the value of the RMS error of the fit.

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

ADVERBS

1	NGAUSS	The number of gaussians.
0	BGAUSS	The first and last channels of the region which was
0	EGAUSS	used to fit the gaussian(s).
0	CENTER(N)	The center channels of the gaussians.
0	HEIGHT(N)	The heights of the gaussians.
0	HWIDTH(N)	The half widths of the gaussians.

TWO-RECEIVER PROCESSING OPTION Does affect GMODEL. If GMODEL is used to model line(s) in the A receiver, either BOTH or ONLYA will be satisfactory process-options. If GMODEL is used to model line(s) in the B receiver, ONLYB should be specified before GMODEL.

RELATED VERBS

GAUSS Refines the user's first guesses of CENTER(N) and HWIDTH(N) for N gaussians. Determines HEIGHT for each gaussian.

GPARTS Evaluates the parameters of, constructs and displays each gaussian separately.

RESIDUAL Evaluates the parameters of each gaussian, constructs the sum of the NGAUSS gaussians and subtracts the sum from the currently referenced array.

PEAK Finds the peak value and sets all the gaussian parameters according for GAUSS.

- continued -

REMARKS

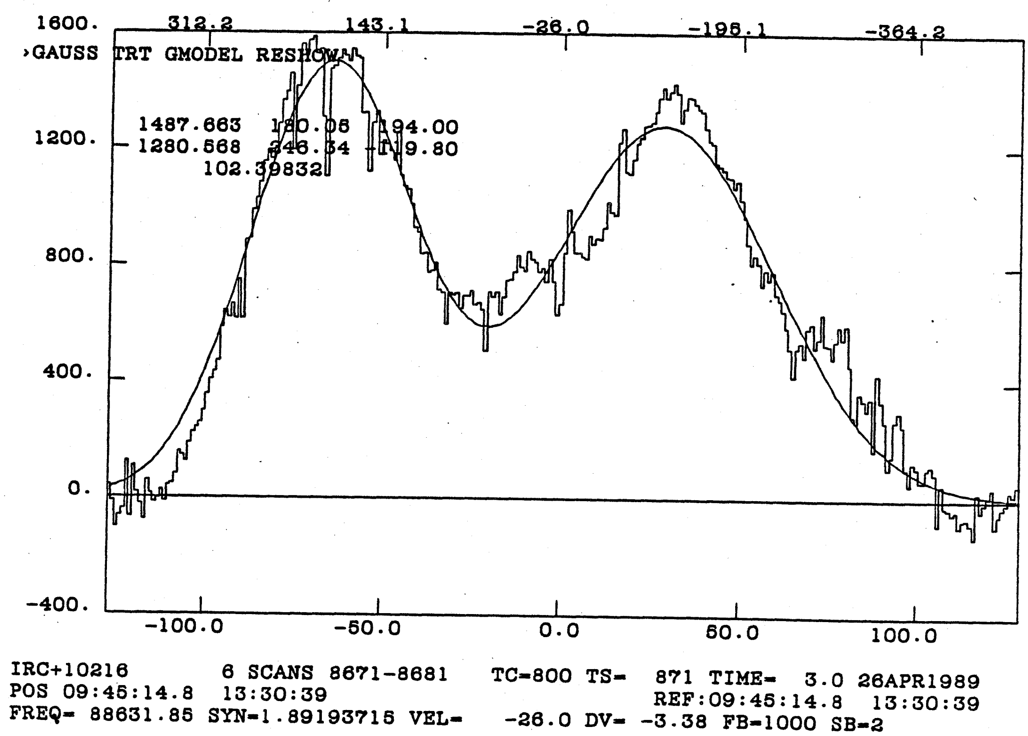
GMODEL is used after GAUSS to print out the parameters of the gaussian(s) and create a model of the fit, or by itself to create a particular gaussian or set of gaussians.

EXAMPLE

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

```
GMODEL RESHOW
```

You will get



where	1487.663 and 1280.568 are the heights of the gaussians in Kelvins, 180.05 and 246.34 are the widths of the gaussians in KM/SEC, 194.00 and -119.80 are the centers of the gaussians in KM/SEC,
and	102.39832 is the RMS error of the fit in Kelvins.

GPARTS

GPARTS separately evaluates the parameters of NGAUSS gaussians, constructs each gaussian, and displays each gaussian on the CRT screen.

PTWH Is not used or changed.

ADVERBS

1	NGAUSS	The number of gaussians.
0 0	BGAUSS EGAUSS	The first and last channels of the region over which the parameters of each gaussian will be evaluated.
0	CENTER(N)	The center channels of the gaussians.
0	HEIGHT(N)	The heights of the gaussians.
0	HWIDTH(N)	The half widths of the gaussians.

TWO-RECEIVER PROCESSING OPTION To model gaussians in the A receiver, either BOTH or ONLYA should be in effect when GPARTS is used.

To model gaussians in the B receiver, ONLYB should be specified before GPARTS is used.

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 their parameters and replaces the currently referenced array with the sum.
RESIDUAL	Same as GMODEL, except that the sum is subtracted from the currently referenced array.
PEAK	Finds the peak value and sets all the gaussian parameters accordingly for GAUSS.

- continued -

REMARKS

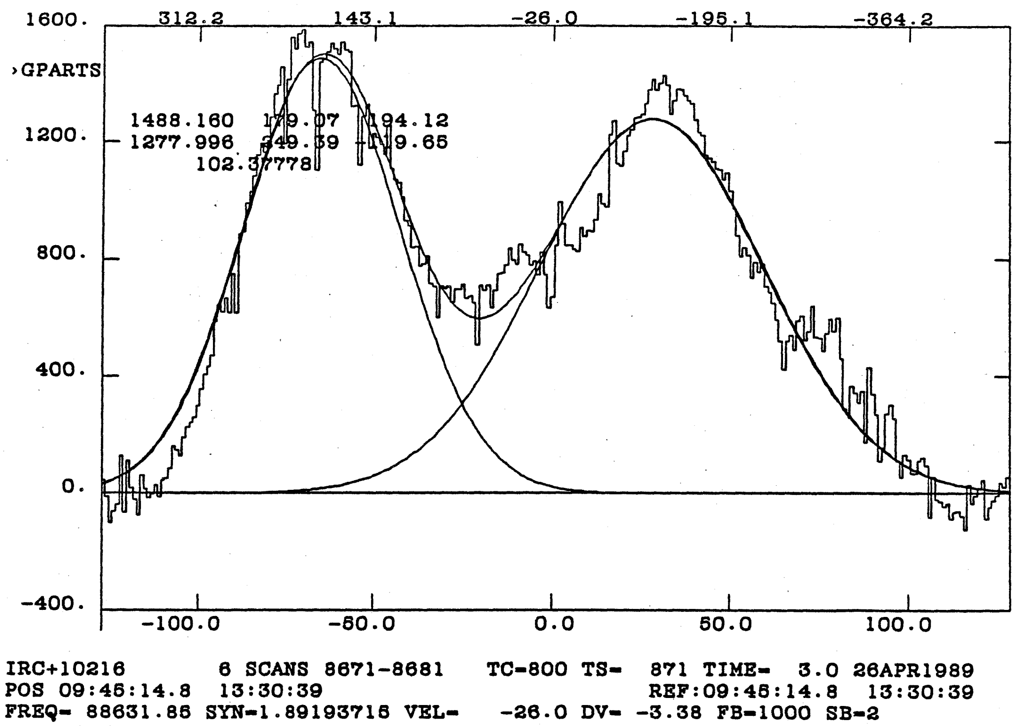
GPARTS does not change the contents of any of the three arrays.

EXAMPLE

You have already fit three gaussians to your data. To see the separate gaussians, specify

GPARTS

You will get



HANNING

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

$$xi = 0.25X_{i-1} + 0.5\ X_i + 0.25X_{i+1}$$

PTWH Determines which scan is smoothed. Is not changed.

ADVERBS

0	`BDROP(RCVR)`	The number of channels at each end of each
0	`EDROP(RCVR)`	receiver which will be ignored by the routine.

TWO-RECEIVER PROCESSING OPTION Does affect `HANNING`. Either or both receivers may be smoothed, depending on the processing option.

RELATED VERBS

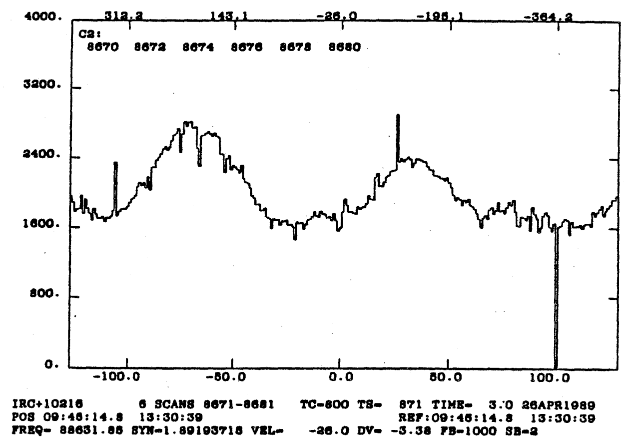
`BOXCAR` Smoothes a scan by averaging a specified number of channels together.

`SMOOTH` Smoothes a scan by averaging channels according to a user specified function.

- continued -

EXAMPLE

Your data looks like



To smooth it, specify

HANNING PAGE RESHOW

You will get

HARMONIC

Puts the value of the harmonic of the data in the currently referenced array on the value stack.

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. Is not changed.
ADVERBS	None

TWO-RECEIVER PROCESSING OPTION Does not affect HEADER.

RELATED VERBS

TABLE	Prints the channel by channel values for the currently referenced array.
SNAP	Prints the header information for the scan in the currently referenced array on the line printer.

ERRORS

If there is no scan in the currently referenced array when HEADER is specified, the message LOAD SCAN will be generated.

- continued -

EXAMPLE

To print the header information for scan 8648, specify

8648 HEADER

You will get

```
>8648 HEADER

---------------------------------------------------------------------------
 SCAN  8649            NRAO 12M            IRC+10216                   WWS
---------------------------------------------------------------------------
MM/DD/YY                 LST                UTC             MODE     OPR
04/26/89              12:28:43.1        05:38:17.5          PS       TWF

                  RA          DEC                         OFFSETS
                                               POINTING    +BEAM      REF
1950       09:45:14.8    13:30:39     AZ:     00:02       02:00      30:00
1950 REF   09:45:14.8    13:30:39     EL:     -1:07       00:00      00:00

GALACTIC     221.4467     45.0598     RA:     00:00                  00:00
AZ/EL        253.5205     48.7068    DEC:     00:00                  00:00

 TSYS     TC     RCVRS    BW/CHAN    REST FREQ    DOP FREQ      IFS    SB

  871.   800.0     1    F  1000.0    88631.850    88629.117     100.   2.
                        S  1000.0   1.89193734                 1500.

 TIME    NS     SEC        VEL     VLSR/VR    VEL/CHAN    TOL      FO

 15.0    30     30.0      -26.00    35.240     -3.382     10.    47.6
```

HEADLEN

Puts the value of the header length (in R*4 words) of the scan in the currently referenced array on the value stack. This value may be printed or used in calculations. The index of the first spectral value is

HEADLEN+1

EXAMPLE

to zero the first spectral value

LEN=HEADLEN; TWH(LEN+1,PTWH)=0.

HELP

HELP is an informational verb with two uses: (1) when used by itself, lists the verbs and symbols recognized by the program (adverbs not included); (2) when used with the object PROCEDURE, lists the names of the procedures currently defined and in memory; (3) when used with the objects ARRAY and SCALAR, lists the arrays and scalars (adverbs) used by the program.

PTWH	Is not 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	When used with HELP, generates a list of names of defined procedures.
SCALAR	When used with HELP, generates a list of the scalars (adverbs and variables) used by the program.

TWO-RECEIVER PROCESSING OPTION Does not apply.

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`	Alias for PRINT.

REMARKS

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

- continued -

EXAMPLE

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

```
HELP

   The  HELP  command  invokes  the  VAX/VMS  HELP  Facility  to  display
   information about a VMS command or topic.  In response to the "Topic?"
   prompt, you can:

      o Type  the  name of the command  or topic for which you need help.

      o Type  INSTRUCTIONS  for more detailed  instructions on how to use
        HELP.

      o Type  HINTS  if you are not  sure of the  name  of the command or
        topic for which you need help.

      o Type a question mark (?) to redisplay the most recently requested
        text.

      o Press the RETURN key one or more times to exit from HELP.


Press RETURN to continue ...
```

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

```
You  can  abbreviate any  topic name, although ambiguous abbreviations
result in all matches being displayed.

  Additional information available:

  :=         =          @          ACCOUNTING ALLOCATE   ANALYZE    APPEND
  ASSIGN     ATTACH     BACKUP     CALL       CANCEL     CC         CLOSE
  CONNECT    CONTINUE   CONVERT    COPY       CREATE     DAU_Graph
  DEALLOCATE DEASSIGN   DEBUG      DECK       DEFINE     DELETE     DEPOSIT
  DIFFERENCES           DIRECTORY  DISCONNECT Diskeeper  Disk_Analysis
  DISMOUNT   DUMP       EDIT       ENCRYPT    EOD        EOJ        Errors
  EXAMINE    EXCHANGE   EXIT       FDL        FINGER     FORTRAN    FTP
  GOSUB      GOTO       HELP       Hints      IF         INITIALIZE INQUIRE
  Instructions          JOB        Lexicals   LIBRARY    Line_editing
  LINK       LOGIN      LOGOUT     MACRO      MAIL       MERGE      MESSAGE
  MIDAS      MONITOR    MOUNT      MULTINET   NewFeatures_V44
  NFSDISMOUNT           NFSMOUNT   ON         OPEN       PASSWORD   PATCH
  PHONE      PRINT      PURGE      Queues     RCP        READ       RECALL
  RECOVER    RENAME     REPLY      REQUEST    RETURN     RLOGIN     RMS
  RSHELL     RTL_Routines          RUN        RUNOFF     RUSERS     SEARCH
  SET        SHOW       SORT       SPAWN      Specify    START      STOP
  SUBMIT     Symbol_Assign         SYNCHRONIZE           System_Services
  TALK       TELNET     TFTP       TYPE       UNLOCK     VAXsim     WAIT
  WHOIS      WPCORP     WRITE

  Additional help libraries available (type @name for topics):

  OPER12M  PGPLOT  NRAO  TOOLS  UATOOLS

Topic?
```

HISTOGRAM

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

PTWH Is not changed or used.

ADVERBS None.

TWO-RECEIVER OPTION Does not apply.

RELATED VERBS

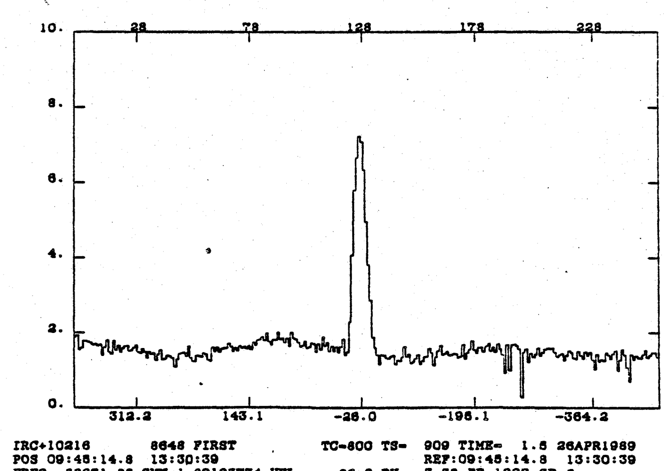
LINE	Causes SHOW to display spectra as a continuous line.
POINTS	Sets the SHOW display mode to points.
RLINE RPOINTS RHIST	Sets plotting mode for RESHOW display tp LINE, POINTS, HISTOGRAM, respectively.
RESHOW	Plots the contents of the currently referenced array.
SHOW	Plots the contents of the currently referenced array on the CRT screen; draws and labels the axes.

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-scaling that was last used before HOLDY was specified. Once HOLDY has been used, the Y-scaling will remain fixed until FREEY or AUTO is specified or the adverb YINCR is set to greater than -9999.

PTWH Is not used or changed.

ADVERBS

-9999 YMIN(R) Is set to -9E10 by HOLDY. The verb SHOW does not update its Y-scaling if YMIN $\leq$ -9E09.

TWO-RECEIVER PROCESSING OPTION Does affect HOLDY. The Y-scaling will be fixed for either or both receivers depending on the processing option when HOLDY was specified.

RELATED VERBS

FREEY Sets YMIN to -9E04. The verb SHOW will calculate the Y-scaling to be used if 9E09 < YMIN $\leq$ -9999.

AUTO Alias for FREEY.

RANGE Sets YMIN and YINCR.

EXAMPLE

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 710 PAGE SHOW
HOLDY
GET 736 PAGE SHOW
GET 752 PAGE SHOW
GET 784 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 completely 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.

TWO-RECEIVER PROCESSING OPTION Does not apply.

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 A,B
IF A>B
   THEN PRINT A
   ELSE PRINT B
   END
FINISH
```

(2) IF statement on one line in a procedure

```
PROCEDURE EGALT
READ A,B
IF A>B;  THEN PRINT A;  ELSE PRINT B;  END
FINISH
```

(3) IF statements nested in a procedure

```
PROCEDURE EG2
READ A,B,C
IF A>B;  THEN PRINT A
         ELSE IF B>C; THEN PRINT B
                      ELSE PRINT C
              END
       END
FINISH
```

INDX

Puts the value of the horizontal telescope coordinate of the position actually observed (i.e., position measured by horizontal decoder) of the data in the currently referenced array on the value stack.

INDY

Puts the value of the vertical telescope coordinate of the position actually observed (i.e., position measured by vertical decoder) of the data in the currently referenced array on the value stack.

INSTALL

Causes procedures to be read from editor files outside the program. Allows user to create ASCII files with the VAX text editor which can remain on disk indefinitely to be used on future observing runs. See EDT, PRCDIR

EXAMPLE INSTALL XMPL

INTTIME

Puts the value of the total integration time for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

INVERT

A routine which reverses the order of the spectral values and changes the sign of the delta velocity.

PTWH Determines which array is reversed. Is not changed.

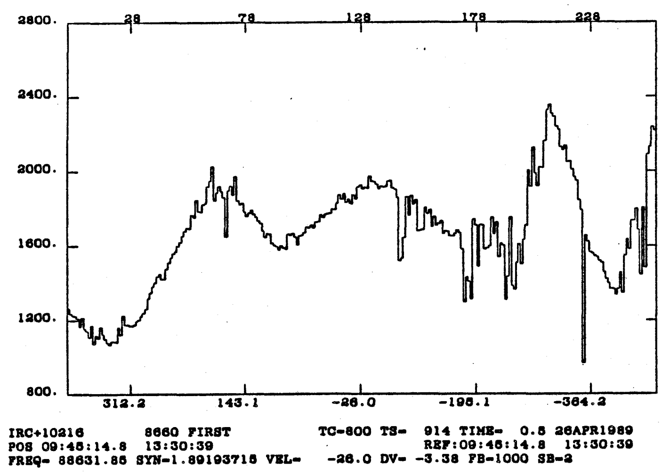
ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does affect INVERT. Either or both receivers may be inverted, depending on the processing option.

RELATED VERBS None.

EXAMPLE

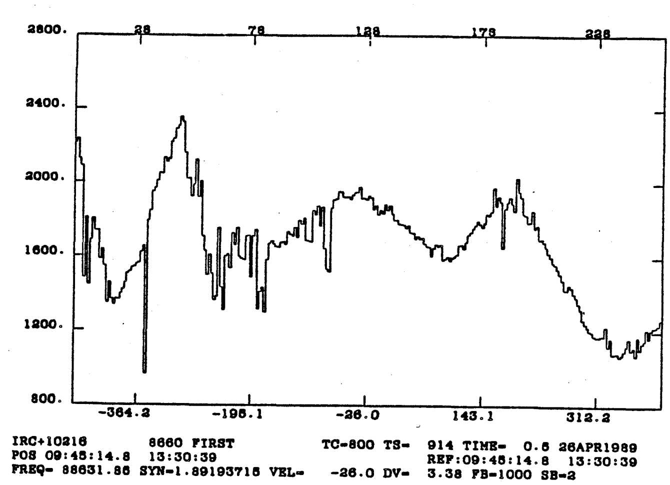
Your data looks like



but you want to reverse (invert) the display,

so you specify

    INVERT PAGE SHOW



KEEP

KEEP copies the scan in the currently referenced array into the PKFL file on disk. Scans from this file can be exported to the observer in the ASCII FITS format. Observers can take home reduced data as well as the raw data. Data that is kept (KEEP) CANNOT be retrieved by the LINE program, but EDITed and SAVE data may be retrieved.

PTWH Determines which array will be copied to disk. Is not changed.

ADVERBS None.

TWO-RECEIVER OPTION Does not apply.

RELATED VERBS

- **RECALL** Copies the scan in the bin NSAVE into the currently referenced array.
- **RESAVE** Stores the scan in the currently referenced array as an edited scan in the PKFL file.
- **SAVE** Copies the currently referenced array into the disk bin NSAVE.

REMARKS

Please see the programmer for information on the data tape format and how to read the tape.

EXAMPLE

You have stacked sixty scans and wish to carry home the result to do further processing at your home institution. Type

```
KEEP
```

and at the end of your run exit LINE and fill out the "Data Tape Request Form." Give this to the operator or to the secretary at the Tucson office. Your files will be processed and sent to you as per your request.

LABEL

LABEL is used to complete a contour map begun by MAPSHOW. LABEL draws and labels the axis of the contour map and labels the contour levels.

PTWH	Determines which array's contents are mapped. Is not changed.

ADVERBS

BDROP(RCVR) EDROP(RCVR)	The number of channels at each end of each receiver 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.
MRATIO	Is used to vary the relative X-Y scaling of the map made by MAPSHOW.

TWO-RECEIVER PROCESSING OPTION Does apply. If BOTH or ONLYA is in effect when MAPSHOW and LABEL are used, the contour map for receiver A is drawn and labeled. If ONLYB is in effect, the contour map for receiver B is drawn and labeled.

RELATED VERBS

MAPSHOW	Draws a contour map with the labels specified by LEVS.
RA	Selects right ascension as the Y-coordinate for the map.
DEC	Selects declination as the Y-coordinate for the map.
GL	Selects galactic longitude as the Y-coordinate for the map.
GB	Selects galactic latitude as the Y-coordinate for the map.

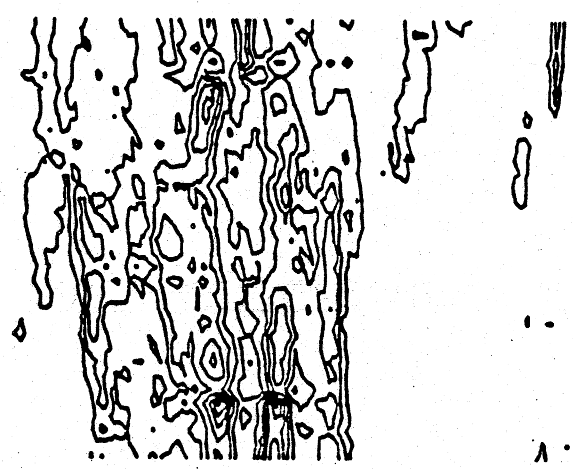
- continued -

REMARKS

LABEL should always be called after MAPSHOW.

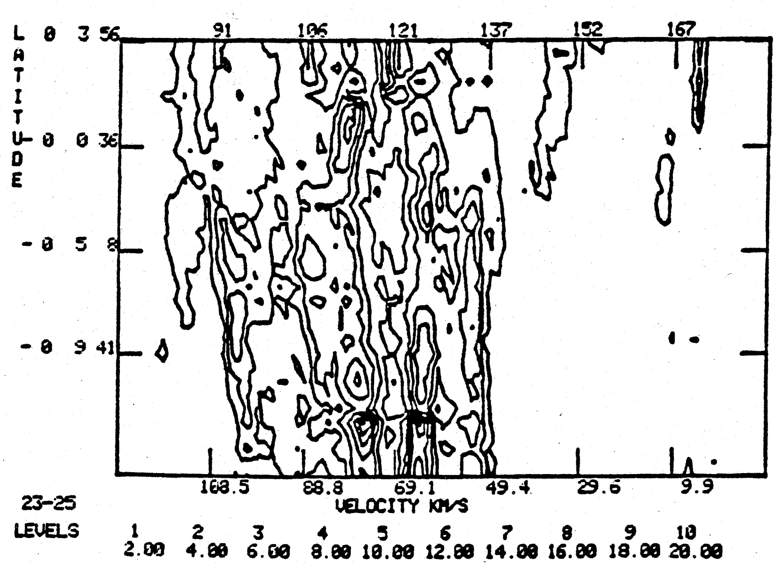
EXAMPLE

Contour map before labeling (see MAPSHOW):



Same map after LABEL has been specified:

(channel numbers)



LASER

Invokes the VMS DCL command PRINT/QUE=QMS which allows the user to print ASCII files on the laser printer from the LINE program.

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 None.

OBJECTS The minimum and maximum channel numbers desired.

TWO-RECEIVER PROCESSING OPTION Does affect LIMIT. Only one receiver may be processed at a time.

RELATED VERBS

SPREAD Limits the data processing to only those channels within the minimum and maximum frequency 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 ALL procedure.

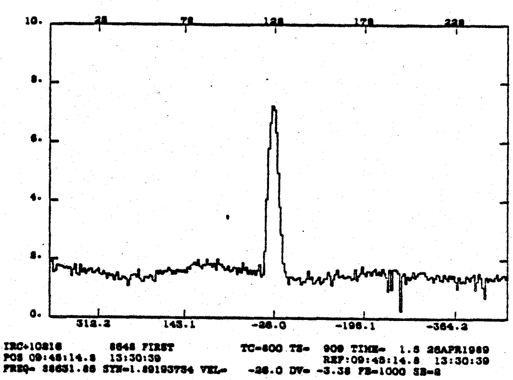
- continued -

ERRORS

If LIMIT is specified without minimum and maximum values, the message "ARG LIST?" will be printed.

EXAMPLE

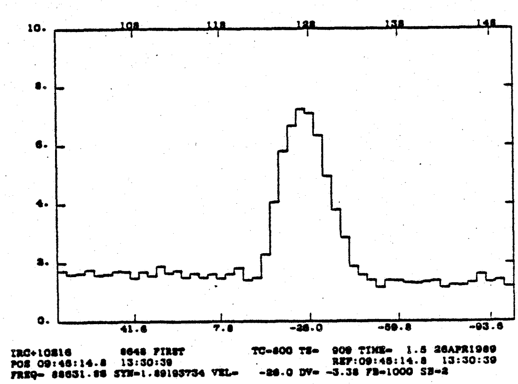
Your current display looks like



and you would like to limit your display to channels 100 thru 150.
Specify

```
100  150  LIMIT PAGE SHOW
```

and you will get



LINE

A routine which sets the plotting mode for the `SHOW` display to a continuous line.

PTWH Is not changed or used.

ADVERBS None.

TWO-RECEIVER OPTION Does not apply.

RELATED VERBS

`HISTOGRAM`	Causes `SHOW` to display spectra as a histogram.
`POINTS`	Sets the `SHOW` display mode to `POINT`.
`RESHOW`	Plots the contents of the currently referenced array using the y-scaling last used by the verb `SHOW`.

`RLINE`
`RPOINTS`
`RHIST`
	Sets plotting mode for `RESHOW` display to `LINE`, `POINTS`, `HISTOGRAM`, respectively.

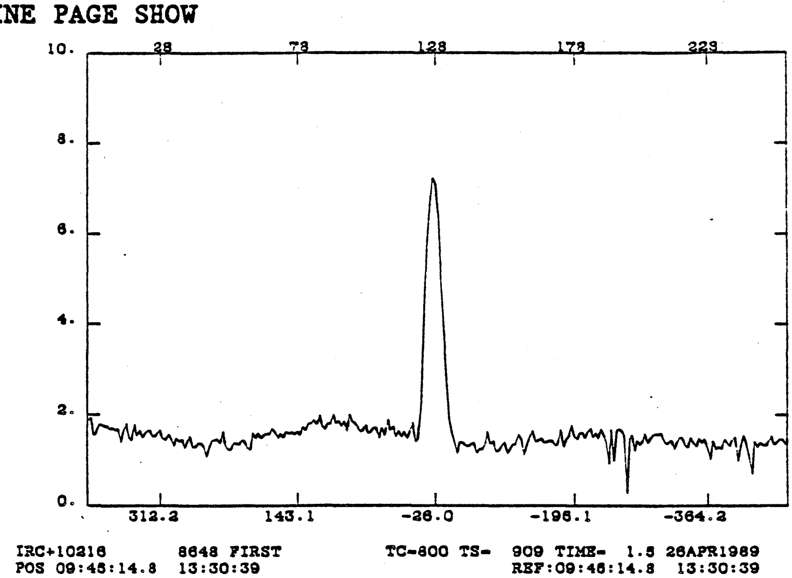
`SHOW`	Plots the contents of the currently referenced array on the CRT screen.

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.

TWO-RECEIVER PROCESSING OPTION Does not apply.

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 A
3    PRINT A
4    FINISH
```

LISTSHIFT

This routine lists all scans that have currently been stacked including the alignment offset that has been computed by `ALIGN` on the CRT.

PTWH Is not used or changed by `LISTSHIFT`.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

`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.

`ALIGNF` Sets the alignment mode to align frequencies.

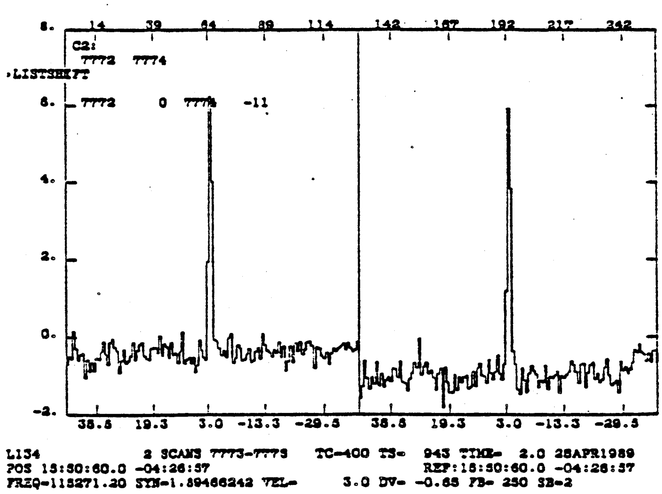
`ALIGNV` Sets the alignment mode to align velocities.

EXAMPLE

The user types the following commands:

```
7772 7774 ADD
SCLEAR
ALIGNV
GET 7772 ACCUM
GET 7774 ALIGN ACCUM
AVE PAGE SHOW
C2
LISTSHIFT
```

and gets



LOFACT

Puts the value of the LO factor (fundamental, doubler or tripler) of the data in the currently referenced array on the value stack.

LOGFILE

Writes two 80-byte records to file LOGFILE.DAT. If file already exists, records are added to it, otherwise a file is created. Contents of 1st rec: scan, source, RA, Dec, LII, BII, Az, El, and date. Contents of 2nd rec: LST, UT, TC, TAU, center freq and velocity, sideband, # rcvrs, # points for each receiver, filterbank resolution, delta velocity, integration time and 2 system temperatures.

LOIF

Puts the value of the LO-IF of the data in the currently referenced array on the value stack.

LST

Puts the value of LST (local sidereal time at the start of the observation) of the data in the currently referenced array on the value stack.

MAKECOPY

This command causes a hard copy to be made from the CRT screen. All input and output to the CRT screen is saved in a hardcopy file which is sent to the QMS laser printer when the MAKECOPY verb is invoked. The hardcopy file is not reinitialized until the PAGE verb is invoked, regardless of the number of hard copies made.

PTWH Is not used or changed by MAKECOPY.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not affect MAKECOPY.

RELATED VERBS None.

REMARKS This verb can be used to make automatic hardcopies within a procedure.

EXAMPLE

You would like a hard copy of the spectrum on the screen. You can type MAKECOPY or strike the MAKECOPY button on the terminal.

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 a program restart) copies the scan in the currently referenced array into the HOLD array. Subsequent MAPSHOWs 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.

PTWH Determines which scan is contoured. Is not changed.

ADVERBS

0	BDROP(RCVR)	The number of channels at each end of each receiver
0	EDROP(RCVR)	that are ignored by the routine.
-1000	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 -1000.
1	MRATIO	Is used to vary the relative X-Y scaling of the map made by MAPSHOW.
-1	SQMAP	(no info yet)

TWO-RECEIVER PROCESSING OPTION Does apply. If the processing option is BOTH or ONLYA, a contour map for the A receiver will be generated. If the option is ONLYB, a contour map for the B receiver will be generated.

RELATED VERBS

LABEL	Draws and labels the axis and labels the contours of the map made by MAPSHOW. Used after all the scans have been mapped.
RA	Selects right ascension as the Y-coordinate on the map.
DEC	Selects declination as the Y-coordinate of the map.

- continued -

GL	Selects galactic longitude as the Y-coordinate of the map.
GB	Selects galactic latitude as the Y-coordinate of the map.

REMARKS

MAPSHOW contours only one scan at a time, so it is often used in a procedure.

EXAMPLE

To produce a contour map of the scans that already have had baselines removed in the NSAVE area with contour levels at 2, 4, 6, 8, 10, 12, 14, 16, 18, and 20 Kelvins, specify

```
LEVS = 2., 4., 6., 8., 10., 12., 14., 16., 18., 20., -9999.
```

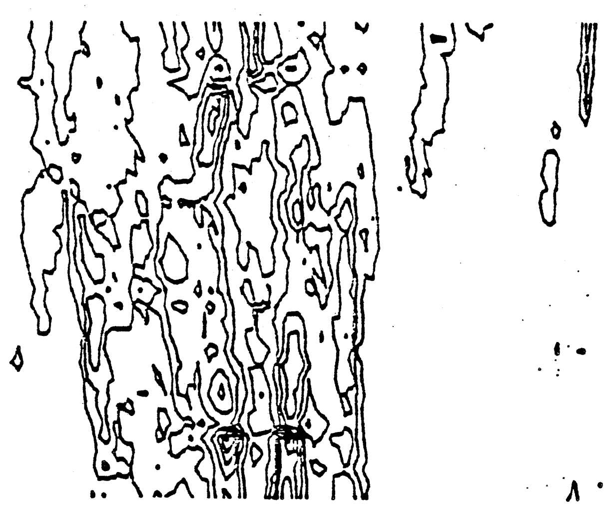
to set the levels, and

```
MRATIO = 2
```

to expand the graph in the vertical direction. Then enter

```
PROCEDURE MAP (NN)
FOR I = 1 to NN
      NSAVE = I;   RECALL MAPSHOW
      END
FINISH
PAGE MAP (33)
```

and you'll have something like



(see LABEL verb to label the completed contour map.)

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.

PTWH — Determines for which scan the moment is calculated. Is not changed.

ADVERBS

0	NMOMENT	The order of the moment to be calculated, as follows: 0 = AREA (Kelvins/km/sec) 1 = CENTROID (km/sec)
0 0	BMOMENT(RCVR) EMOMENT(RCVR)	The first and last channels of the region over which the moment is to be calculated.

TWO-RECEIVER PROCESSING OPTION — Does apply. The specified moment may be calculated for either or both receivers, depending on the processing option.

REMARKS

The moment calculated by MOMENT is stored in the variable SIZE, which can be printed by the command

```
PRINT   SIZE
```

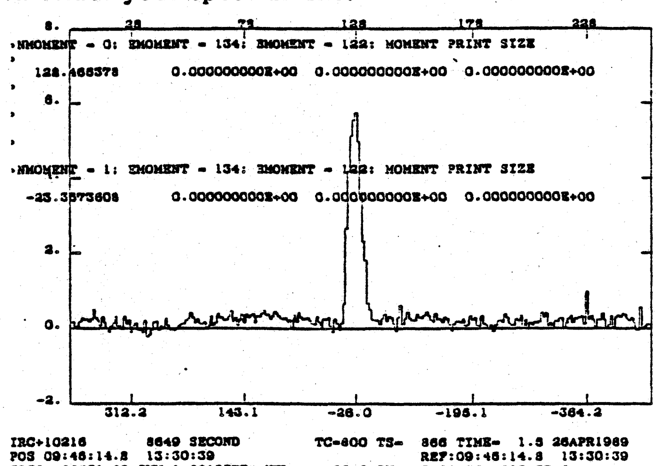
One value is printed for each receiver.

EXAMPLE

You want to calculate the area under your spectral line.

You specify

```
NMOMENT = 1
EMOMENT = 134
BMOMENT = 122
MOMENT
PRINT SIZE
```



NOINT

Puts the value of the total number of integrations for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

NOPHASE

Puts the value of the number of phases per cycle (different states of switched variables) of the data in the currently referenced array on the value stack.

NOPTS

Puts the value of the number of points (total number of cells in the map) of the data in the currently referenced array on the value stack.

NORCHAN

Puts the value of the number of independent polarizations or channels of this frontend of the data in the currently referenced array on the value stack. When receiver channels refer to different positions in the sky, they are expressed as separate scans.

NOSWVAR

Puts the value of the number of switching variables of the data in the currently referenced array on the value stack.

NOXPTS

Puts the value of the number of x points (map sample points along the "X-edge" of the rectangle) of the data in the currently referenced array on the value stack.

NOYPTS

Puts the value of the number of y points (map sample points along the "Y-edge" of the rectangle) of the data in the currently referenced array on the value stack.

OBJECT

Puts a field of sixteen characters which describe the source name of the object observed into the adverb SOBJECT.

EXAMPLE to query the source name

```
OBJECT; PRINT SOBJECT
```

OBSERVER

Puts a field of sixteen characters containing the observer name of the data in the currently referenced array into the adverb SOBSERVER. SOBSERVER may then be printed.

EXAMPLE

```
OBSERVER; PRINT SOBSERVER
```

OBSFREQ

Puts the value of the observed frequency of the selected receiver channel of the data in the currently referenced array on the value stack. Only the most significant 32-bits are stored on the stack. The receiver channel is selected by the value of the adverb RCVR.

EXAMPLE

```
RCVR=2; PRINT OBSFREQ
```

OBSID

Puts a field of eight characters containing the observer and operator initials of the data in the currently referenced array into the adverb SOBSID. This adverb may then be printed.

EXAMPLE To query the observer initials

```
OBSID; PRINT SOBSID
```

OBSMODE

Puts a field of eight characters where four describe the type of data and four describe the observing mode into the adverb, SOBSMODE. This adverb may then be printed.

EXAMPLE To query the observing mode and type of data, type

```
OBSMODE;  PRINT SOBSMODE
```

OBSTOL

Puts the value of the observing tolerance of the data in the currently referenced array on the value stack.

OFF

OFF requires a scan number as its object. OFF copies the indicated scan from the disk to the TEMP array.

PTWH Is not used, but is set to the TEMP array.

ADVERBS None.

OBJECTS

ON scan # — The number of the scan to be copied into the TEMP array.

TWO-RECEIVER PROCESSING OPTION — Does not affect OFF. After OFF, the scan in the TEMP array will have the default processing option.

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 onscans 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 710 and offscan 708 and display the result, specify

```
ON 710;  OFF 708;  DIFF PAGE SHOW
```

OFFSCAN

Puts the value of the off scan number (the last designated total power off scan) of the data in the currently referenced array on the value stack.

ON

It requires a scan number as its object. `ON` copies the indicated scan from the disk to the `WORK` array.

PTWH Is not used, but is set to the `WORK` array.

ADVERBS None.

OBJECTS

`OFF scan #` The number of the scan to be copied into the `WORK` array.

TWO-RECEIVER PROCESSING OPTION Does not affect `ON`. After `ON`, the scan in the `WORK` array will have the default processing option.

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 offscans 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 710 and offscan 708 and display the result, specify

```
ON 710;  OFF 708;  DIFF PAGE SHOW
```

ONLYA ONLYB

ONLYA changes the two-receiver processing option of the scan in the currently referenced array to only the A receiver.

ONLYB changes the two-receiver processing option of the scan in the currently referenced array to only the B receiver.

PTWH Determines which scan's processing option is changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Is changed for the scan in the currently referenced array.

RELATED VERBS

BOTH Changes the processing option of the currently referenced scan to both receivers (the default option).

REMARKS

ONLYA and ONLYB should not be used on one-receiver data. When ONLYA, ONLYB or BOTH are used, the processing option is changed only for that copy of the scan in the currently referenced array. Whenever a scan is copied from the disk, it will have the default processing option until it is changed by the user.

EXAMPLE

To remove a baseline only from A receiver of the currently referenced array, specify

```
ONLYA BASELINE PAGE SHOW
```

To remove a baseline only from the B receiver, specify

```
ONLYB BASELINE PAGE SHOW
```

To display both receivers together again, specify

```
BOTH PAGE SHOW
```

OPENPAR

Puts a field of 56 characters (the entire string is 80 characters in length but the first 24 are reserved for stacking) of the data in the currently referenced array into the adverb, SOPENPAR. This adverb may then be printed. These 56 characters are available to the observer for description of his data processing to be stored in the header of the data.

ORIENT

Puts the value of the rotation or polarization angle orientation of the receiver or reflector at the prime focus of the data in the currently referenced array on the value stack.

OSB

Sets a flag so that ALIGN computes the shift when aligning frequency or velocity for the other sideband by changing the sign of the shift. Typing OSB once sets it, typing it again clears it.

PAGE

PAGE erases the CRT screen and returns a carat (>) at the top of the now blank screen. PAGE is equivalent to striking the PAGE ERASE button (ERASE SCREEN) except that the PAGE ERASE button does not respond with a carat.

PTWH Is not used or changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

REMARKS

If you hit the PAGE ERASE button instead of entering the command PAGE, you will not get a carat. To be certain that the program is ready to accept a command, you can strike RETURN to get a carat.

N. B. The verb PAGE is preferred over striking the PAGE ERASE button because PAGE also resets the relative origin of the CRT screen.

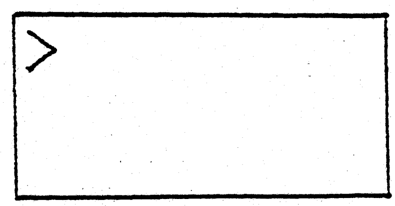
EXAMPLE

You have a full screen on the CRT.

You want to erase it, so you specify

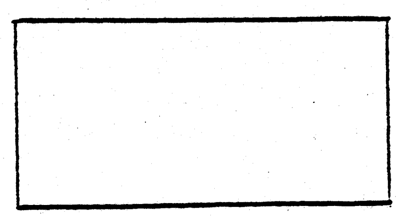
PAGE

and you will get



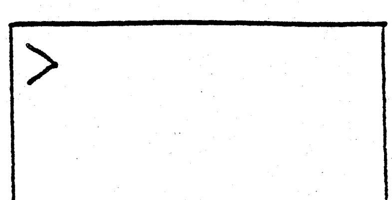
If instead you hit PAGE ERASE,

you will get



If you then hit RETURN,

you will get



PAUSE

Causes the program to suspend operations for ten seconds. Used to build in a delay in large loops of data displays.

PDOC

Is used by the LF and CF pointing procedures to display the five points, the gaussian fits, updated pointing corrections and peak flux densities. PDOC will fit spectral line and continuum five point data.

PEAK

PEAK finds the CENTER, HWIDTH and HEIGHT of a single gaussian. Prints CENTER channel and velocity and PEAK temperatures. Sets BGAUSS and EGAUSS for GAUSS verb.

PTWH Determines which array is to be used.

ADVERBS Sets CENTER(1), HWIDTH(1), HEIGHT(1), BGAUSS and EGAUSS.

TWO-RECEIVER PROCESSING OPTION

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 values of HEIGHT (in Kelvins), CENTER (in km/sec) and HWIDTH (in km/sec), plus the RMS error of the fit.

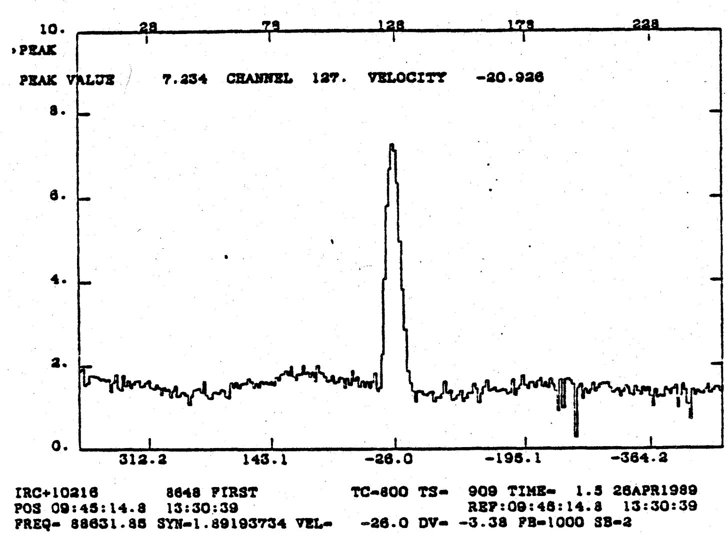
EXAMPLE

You have just removed a baseline from scan 8648. Now you want to fit a gaussian to the primary peak. Type

PEAK

to set all the initial guesses for the gaussian fit.

You will get



PLACE

Moves cursor to position commanded by IX & IY.

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.

TWO-RECEIVER PROCESSING OPTION

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.

PTWH Is not used or changed by `POINTS`.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION

RELATED VERBS

`HISTOGRAM`	Causes `SHOW` to display spectra in the form of a histogram.
`LINE`	Causes `SHOW` to display spectra in the form of a continuous line.
`RESHOW`	Plots the contents of the currently referenced array using the y-scaling last used by the verb `SHOW`.

`RHIST`
`RLINE`
`RPOINTS`
	Sets the plotting mode for `RESHOW` display to `HISTOGRAM`, `LINE`, `POINTS`, respectively.

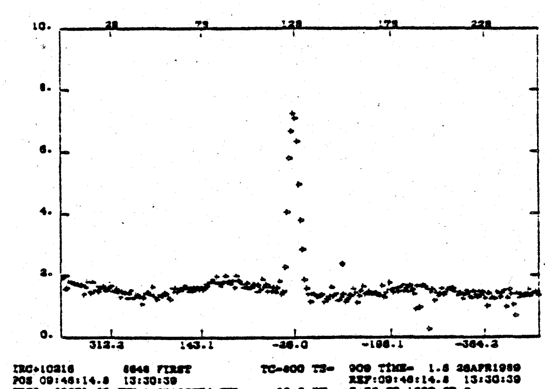
`SHOW`	Displays a scan with user specified or default scaling and user specified or default labeling.

EXAMPLE

You wish to display the current scan in points mode. Specify

```
POINTS PAGE SHOW
```

and you will get



POLARIZ

Puts a field of eight characters which describe the type of polarization and the angle for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

POWSPEC

Computes the power spectrum of the data in the currently referenced array.

PRCDIR

Invokes the VMS DCL command DIRECTORY *.PRC which lists all files in the current directory which have the extension PRC.

PRECIS

Puts a field of eight characters describing the precision of the data into the adverb SPRECIS. This adverb may then be printed.

EXAMPLE

To query the precision of the data

```
PRECIS;  PRINT SPRECIS
```

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 scaler defined in a procedure) name, the value(s) of the adverb or variable will be printed. 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.

TWO-RECEIVER PROCESSING OPTION

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`	Alias for PRINT.

REMARKS

PRINT can be used to monitor the values of adverbs, PTWH, SIZE and VRMS.

- continued -

EXAMPLES

```
> PRINT CENTER
     25.0000    33.0000    78.0000    0.0000    0.0000    0.0000
> PRINT 'YOU SHOULD USE THE 300-FOOT'
YOU SHOULD USE THE 300-FOOT
> PRINT SQRT(144)
12.0000
> PRINT HWIDTH(2)
5.0000
PRINT 'x =  ' X
x =  1.2500
> PRINT BMARK Z_LINE YMIN
1.0000        0.0000  -9999.9999      -9999.9999
```

PRINTER

`Printer` directs all non-graphic output of the program to the printer instead of the CRT screen. The default is the CRT.

See CRT

EXAMPLE `PRINTER`

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 DEFINITION mode. When in DEFINITION mode, the program prompts with a colon (:) instead of a carat (>). The FINISH statement is required to complete the definition of the procedure and return the program to EXECUTE mode.

PTWH Is not used or changed.

ADVERBS None.

OBJECTS

procedurename — The name of the procedure which is being defined. A procedurename may be any not already defined alphanumeric name up to 10 characters long. The first character of the name must be an alphabetic character.

arguments — Variables which are to be passed to the procedure when it is called by the user. Arguments may be any unused alphanumeric name up to 10 characters long. The first character must be an alphabetic character. There is no restriction on the number of arguments permitted, but they must be encased in parentheses and separated by commas.

TWO-RECEIVER PROCESSING OPTION Does not apply.

- 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 procedures (and returns adverbs to their default values).

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)
```

PROJID

Puts a field of eight characters which contains the program ID associated with the proposal as it appears on the telescope schedule of the data in the currently referenced array into the adverb SPROJID.

EXAMPLE

to query the project identification

```
PROJID; PRINT SPROJID
```

PTCON

Puts the value of the selected pointing constant of the data in the currently referenced array on the value stack. There are four pointing constants. Individual constants are selected by setting the value of the adverb, RCVR, to a number between one and four before trying PTCON.

EXAMPLE

to print the third pointing constant type

```
RCVR=3; PRINT PTCON
```

PURGE

Invokes the VMS DCL command PURGE *.PRC which purges all but the latest version of the user defined procedure files.

QGET

Copies current scan from the disk to the WORK array. Specify 1 or 2 for the first or second filterbank.

EXAMPLE `QGET 1`

RA

RA sets an internal flag that causes MAPSHOW to use right ascension as the Y-coordinate for maps.

PTWH	Is not used or changed.
ADVERBS	None.
TWO-RECEIVER OPTION	Does not apply.

RELATED VERBS

DEC	Sets declination as the Y-coordinate for maps.
GB	Sets galactic latitude as the Y-coordinate for maps.
GL	Sets galactic longitude as the Y-coordinate for maps.
LABEL	Labels the map drawn by MAPSHOW.
MAPSHOW	Draws a map with the levels specified by LEVS.

RACCUM

Accumulates and averages the records specified by RSTACK after the records are loaded into memory by RGET.

See RGET, RSCALE, RESAVE

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

0 YMIN	Minimum temperature to be plotted. Set by RANGE.
-9999 YINCR	Increment between tic marks. Set by RANGE.

OPERANDS Two, minimum and maximum values.

TWO-RECEIVER PROCESSING OPTION The Y-scaling is set for both receivers.

RELATED VERBS

HOLDY	Causes SHOW to use the last Y-scaling it used again. Does this by setting YMIN to -9E10.
FREEY AUTO	Causes SHOW to do auto scaling for the y-axis.

REMARKS

The operands may proceed or follow RANGE. If they follow RANGE, they must be enclosed in parentheses.

EXAMPLE

You want to display 10 scans with the same Y scaling (-2~ to 7~). You specify:

```
-2    7 RANGE  or  RANGE (-2,7)
```

and then display the data.

RAP

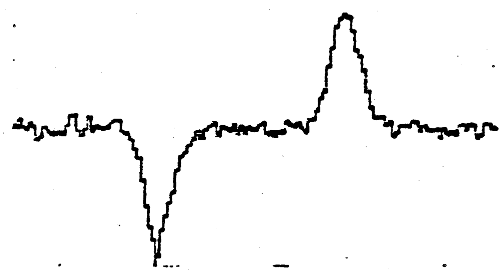
Averages the signal and reference spectra of a frequency switched scan in the currently referenced array. The signal (FS) and reference (FR) frequency adverbs and the sideband number determine the direction and the amount of the shift.

Other relevant parameters or adverbs, with initial value (if any):

0	FS	Signal Frequency
0	FR	Reference Frequency

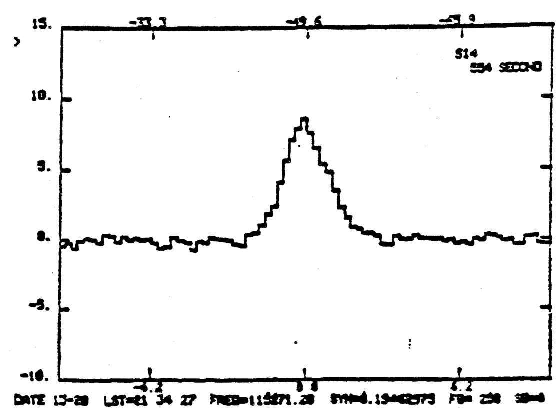
EXAMPLE

Your spectra look like



You want to combine the signal and reference spectra, so you specify

```
DF = 12;   RAP PAGE SHOW
```



RAZOFF

Puts the value of the reference azimuth offset in seconds of arc of the data in the currently referenced array on the value stack.

RDIFF

RDIFF subtracts the contents of the B receiver of the scan in the currently referenced array from the contents of the A receiver of the scan in the currently referenced array. The result is left in the receiver A channels of the currently referenced array.

PTWH Determines for which scan the receivers are differenced. Is not changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION The processing option should be BOTH when RDIFF is used.

RELATED VERBS

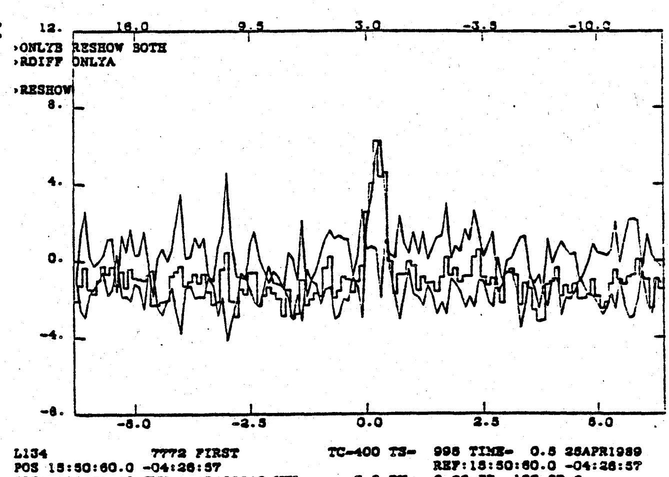
SLIDE Averages receivers A and B of the currently referenced scan, with weighting by integration time and system temperature.

EXAMPLE

You are doing a polarization experiment. Receiver A is measuring right circularly polarized flux, and receiver B is measuring left circulatory polarized flux. You want to difference the two receivers. You specify

ONLYA PAGE SHOW	to display the A receiver.
ONLYB RESHOW BOTH	to display the B receiver and reselect the A receiver.
RDIFF ONLYA	to difference the receivers.
RESHOW	to display the difference

On the CRT, you will have something like:



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>` as described above.

TWO-RECEIVER PROCESSING OPTION Does not apply.

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.

EXAMPLE

(1) 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
```

- continued -

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.

(2) The READ command can be used in a procedure to allow the user to signal the program when to continue. 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 and draws 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

0 NSAVE	The number of the disk bin whose contents are copied into the currently referenced array. NSAVE may range from 1 to 100.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

SAVE	Copies the currently referenced array into the disk bin NSAVE.

ERRORS

If NSAVE is greater than 154, 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
```

RECS

Used to specify which records the observer wishes to retain in a given scan when record editing. RECS takes multiple arguments where negative numbers are to be omitted from the sum and positive numbers are to be included in the sum. The end result (the records to be averaged) are placed in RSTACK to be used by RACCUM.

EXAMPLE

```
(-2, -4) RECS
```

REFPT

Puts the value of the reference point (center channel) of the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

RELOFF

Puts the value of the reference elevation offset in seconds of arc of the data in the currently referenced array on the value stack.

REPLACE

A routine which replaces up to 12 bad channels (BADPT) in the currently referenced array by a parabolic interpolation over adjacent channels. The array BADPT is set by SPIKE or can be set by the user.

PTWH	Determines which array is to be used.

ADVERBS

0 BADPT (12)	Array of bad channels. Is set by SPIKE or can be set by user.

TWO-RECEIVER PROCESSING OPTION Does affect REPLACE. Bad channels can be removed from one or both receivers depending on the processing option.

RELATED VERBS

SPIKE	Sets BADPT to channels whose absolute value is greater than NSIGMA.

EXAMPLE

See BADPT.

RESAVE

Stores the scan in the currently referenced array as an edited scan in the PKFL file. To retrieve this version instead of the original raw data, set GETEDIT = 1 before GETting the scan. To retrieve the original data set GETEDIT = -1 before using GET.

See KEEP, SAVE, RECALL

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 axes.

PTWH	Determines which array is plotted. Is not changed.

ADVERBS

BDROP(RCVR) EDROP(RCVR)	The number of channels at each end of each receiver which will not be plotted.

TWO-RECEIVER PROCESSING OPTION	Does apply. If the option is ONLYA, receiver A will be shown. If the option is ONLYB, receiver B will be shown. If the option is BOTH, the two receivers will be shown side by side.

RELATED VERBS

See SHOW.

REMARKS

RESHOW is often used to plot a polynomial baseline model, sine curve, or gaussian model on top of the spectrum for which the model was generated. RESHOW can also be used to plot the reduced spectrum on top of the original.

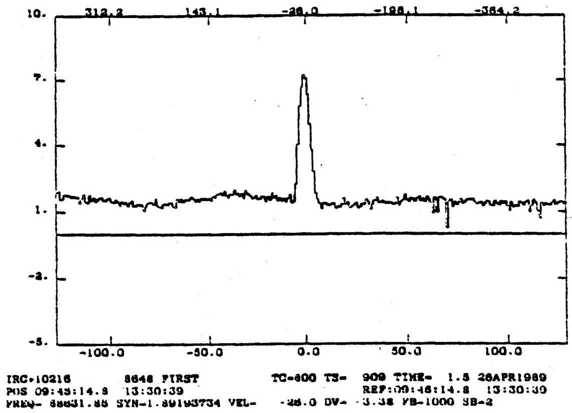
- continued -

EXAMPLE

To plot scan 8648, specify

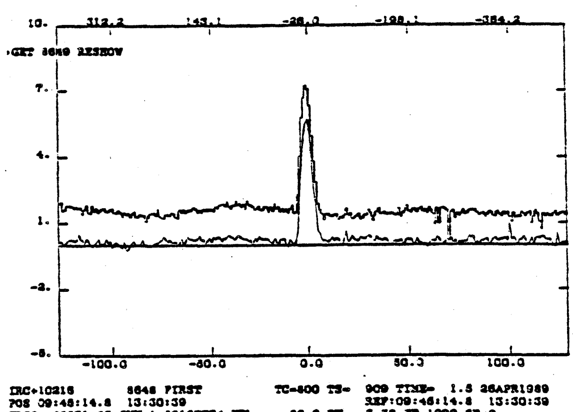
```
GET 8648 PAGE SHOW
```

to see the first filterbank



To see the second filterbank plotted on top of the first filterbank

```
GET 8649 RESHOW
```



RESIDUAL

This routine takes the parameters of the gaussians (usually refined by GAUSS), evaluates the total for each channel and subtracts the total from the currently referenced array.

PTWH	Determines from which array the total model is subtracted. Is not changed.

ADVERBS

1	NGAUSS	The number of gaussians.
0	BGAUSS	The first and last channels of the region over
0	EGAUSS	which the gaussian(s) are subtracted.
0	CENTER(N)	The center channels of the gaussians.
0	HEIGHT(N)	The heights of the gaussians.
0	HWIDTH(N)	The half widths of the gaussians.

TWO-RECEIVER PROCESSING OPTION To subtract one or more gaussians from the A receiver, the option should be ONLYA or BOTH. To subtract one or more gaussians from the B receiver, specify ONLYB.

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.
GPARTS	Constructs each gaussian separately from the parameters and displays each on the screen.
PEAK	Finds the CENTER, HWIDTH, and HEIGHT of a single gaussian. Also sets BGAUSS and EGAUSS.

- continued -

REMARKS

GAUSS may or may not be called before RESIDUAL.

EXAMPLE

You had

    GET 8671 BASELINE PAGE SHOW

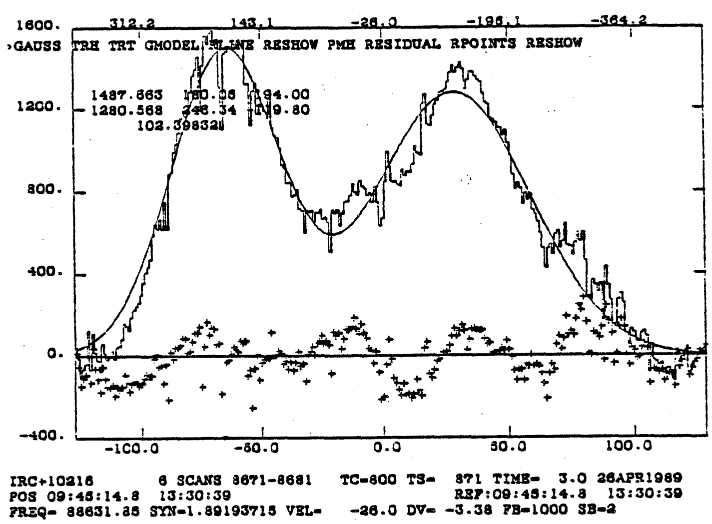
and you have calculated the parameters for two gaussians and displayed the model over the data

    GAUSS TRH TRT GMODEL RLINE RESHOW

Now you want to remove the gaussians and see what the residual looks like. Specify

    PMH RESIDUAL RPOINTS RESHOW

You will have



RESTART

RESTART reinitializes 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 the LINE command which loads the program is executed.

PTWH Is not used, but is set to the WORK array.

ADVERBS All adverbs are set to their default values.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

STORE page# Copies the program memory 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 (WARNING) realize that once you have done this, the three arrays will be empty and your adverbs will return to their initial values.

RESTFREQ

Puts the value of the rest frequency of the selected receiver channel of spectral line data in the currently referenced array on the value stack. Only the most significant 32-bits are stored on the stack. The receiver channel is selected by the value of the adverb RCVR. If the data is continuum, the value of the data scale factor is placed on the value stack.

RESTORE

RESTORE page# restores the program memory to the state it was in prior to the last STORE command, by copying the indicated disk page into the program memory. All adverbs and procedures defined in this memory page will be available in the program.

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 3.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

STORE page# — Creates a copy of the current state of memory in disk page 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 3
```

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. Any procedure which is called by another procedure must be defined before the procedure which calls it.

PTWH	Is not used or changed.
ADVERBS	None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

See PROCEDURE

REMARKS

A procedure which is called by another procedure must be defined before the procedure which calls it.

ERRORS

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.

EXAMPLE

To define a procedure which can be called by another

```
>PROCEDURE BUSY(T)
:R = T * T
:PRINT T R
:RETURN
:FINISH
>
```

- continued -

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'll get

```
12.0000    144.0000
THATS ALL FOLKS
>
```

RGET

Retrieves all records of a given scan from the individual record file. The maximum number of records to be retrieved is 14. A scan number is required as the object of RGET.

EXAMPLE

```
RGET 500
```

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.

TWO-RECEIVER PROCESSING OPTIONAL Does not apply.

RELATED VERBS

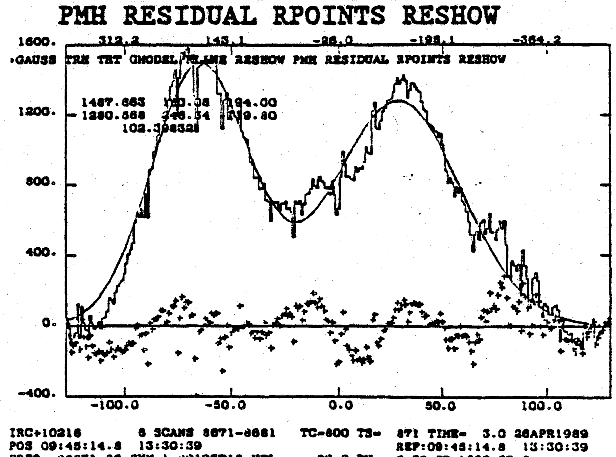
`HISTOGRAM`	Sets the `SHOW` display mode to histogram.
`LINE`	Sets the `SHOW` display mode to a continuous line.
`POINTS`	Sets the `SHOW` display mode to points.
`RESHOW`	Graphs one or both receivers of the currently referenced array on the CRT, no labels or borders, using the scaling last used by `SHOW`.
`SHOW`	Graphs one or both receivers of the currently referenced array on the CRT with labels and borders.

EXAMPLE

You have just fit a double gaussian to scans 8671-8681 and modeled the gaussian over the data. Now you would like to also see the residual by specifying

```
PMH RESIDUAL RPOINTS RESHOW
```

and you get



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

0 `RFREQ`	The approximate frequency (in MHz) of the sinusoidal baseline to be removed. `RFREQ` is iterated by the routine.
0 `BDROP(RCVR)` 0 `EDROP(RCVR)`	The number of channels at each end of each receiver which will be ignored by the routine.
0 `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.
0 `BBASE(RCVR)` 0 `EBASE(RCVR)`	The number of channels at each end of each receiver (not including the channels dropped by `BDROP` and `EDROP`) which will be used to compute the sine curve, if `NREGION(RCVR,1) = 0`.

TWO-RECEIVER PROCESSING OPTION Does affect `RIPPLE`. The sinusoidal baseline may be removed from either or both receivers, depending on the processing option.

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.

see also

`BASELINE BSHAPE BMODEL`

REMARKS

The regions used to compute the fit should not contain a known or suspected spectral feature.

- continued -

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 startchannel.

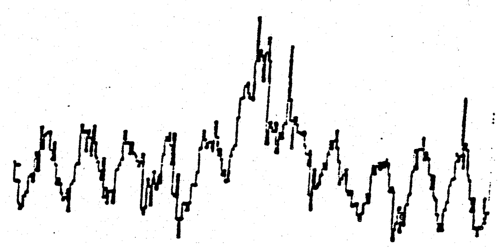
If the routine cannot fit a sine curve with the requested frequency, the message

FIT FAILED

is generated.

EXAMPLE

Your A receiver data looks like



You want to remove a sinusoidal baseline
using these regions to calculate the fit

So you specify

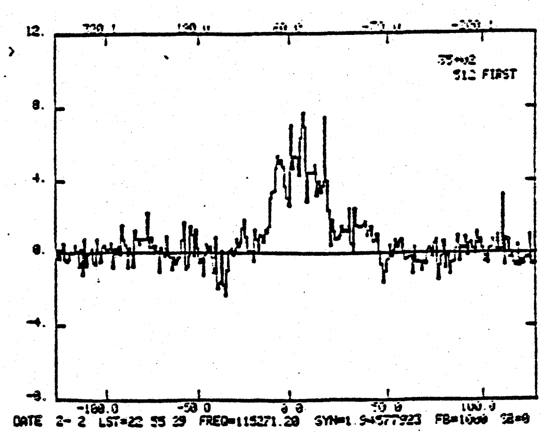
```
NREGION (1,1) = 10;   NREGION (1,2) = 80
NREGION (1,3) = 160;  NREGION (1,4) = 250
NREGION (1,5) = 0
```

and

```
RFREQ = 23
RIPPLE PAGE SHOW
```

(23 is the default value)

to get and display the result



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.

TWO-RECEIVER PROCESSING OPTION Does affect RMODEL. Either or both receivers may be replaced by a sine curve, depending on the processing option.

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.

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.

EXAMPLE

You have twenty scan of the same source and you want to remove the same sine curve from each of them. You can do this by

```
GET 710 TRT RSHAPE RMODEL PMW DIFF SHOW
GET 712 DIFF PAGE SHOW
GET 714 DIFF PAGE SHOW
GET 716 DIFF PAGE SHOW
        .
        .
        .
GET 748 DIFF PAGE SHOW
```

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(RCVR).

PTWH		Determines for which scan the RMS is computed. Is not changed.

ADVERBS

0	BDROP(RCVR)	The number of channels at each end of each
0	EDROP(RCVR)	receiver which will be ignored by the routine.
0	NREGION(RCVR,N)	Specifies the region or regions of the scan to be used to compute the RMS. N is an integer between 1 and 8.
0	BBASE(RCVR)	The number of channels at each end of each
0	EBASE(RCVR)	receiver (not including the channels dropped BDROP and EDROP) which will be used to compute the RMS if NREGION(RCVR,1)=0.

TWO-RECEIVER PROCESSING OPTION Does affect RMS. The RMS may be computed for either or both receivers, depending on the processing option.

RELATED VERBS None.

REMARKS

The RMS computed is stored in the variable VRMS.

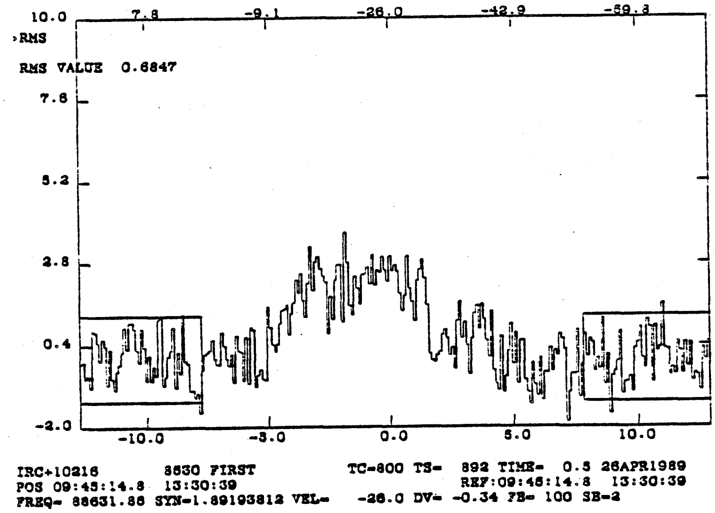
EXAMPLE

To compute the RMS of the scan in the currently referenced array, specify

```
BMARK=1;  PAGE SHOW
```

- continued -

You will see something like



where the boxes indicate the regions over which the RMS will be computed. To compute and print the RMS, specify

RMS

In this case, the result is

RMS VALUE 0.6847

(One value is computed for each receiver).

RSCALE

Computes the appropriate y-scaling for the individual record in the currently referenced array.

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 fit is calculated. Is not changed.

ADVERBS

0 RFREQ	The approximate frequency of the sine curve to be fit. Iterated by the routine.
0 BDROP(RCVR) 0 EDROP(RCVR)	The number of channels at each end of each receiver which will be ignored by the routine.
0 NREGION(RCVR,N)	Specifies the region or regions of the scan to be used to compute the fit. N is an integer between 1 and 8.
0 BBASE(RCVR) 0 EBASE(RCVR)	The number of channels at each end of each receiver (not including the channels dropped by BDROP and EDROP) which will be used to compute the sine curve.

TWO-RECEIVER PROCESSING OPTION Does affect RSHAPE. The fit will be calculated for either or both receivers, depending on the processing option.

RELATED VERBS

RMODEL	Takes the calculated sine curve fit, evaluates it for each channel, and replaces the currently referenced array with the model of the sine curve.
RIPPLE	Calculates the sine curve fit, evaluates it for each channel, and subtracts the model from the currently referenced array.

- continued -

REMARKS

RSHAPE is most frequently used in conjunction with RMODEL to construct a model of a particular sine curve. The set of commands

```
TRT    RSHAPE    RMODEL    PMW    DIFF
```

is equivalent to RIPPLE.

ERRORS

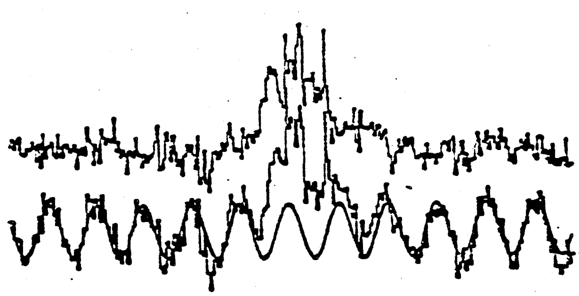
If a sine curve of the requested frequency cannot be fit, the message FIT FAILED is generated.

EXAMPLE

You have scan 1520 in the WORK array and displayed on the screen. You would like to remove a sinusoidal baseline from the data, but you also want to see the sine curve before it is removed. You specify

`TRT RSHAPE RMODEL`	to copy your scan to the TEMP array, compute the sine curve fit for the scan, and replace the copy in the TEMP array with a model of the sine curve.
`RESHOW`	to plot the model on top of the data.
`PMW DIFF RESHOW`	to subtract the model from the data and plot the results on top of the model and the data.

You will have something like



RSHOW

Displays all the individual records of the scan currently in memory in profile form. The records are retrieved by the verb RGET.

Other relevant parameters or adverbs, with initial value (if any):

0	RECN	Number of records in this scan. Usually set by RGET.
0	SHIFT	Bias added to all subsequent records after the first to make the profile plot. The bias added equals (rec no - 1)*SHIFT.

RTSYS

Puts the value of the reference system temperature of the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

RVSYS

Puts the value of the Doppler correction for the earth's motion in the source direction with respect to the velocity reference frame chosen of the data in the currently referenced array on the value stack.

SAMPRAT

Puts the value of the sample period (time required to complete a single sample) of the data in the currently referenced array on the value stack. A sample may be composed of multiple cycles. Also known as sample rate.

SAVE

SAVE copies the scan in the currently referenced array into one of 100 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.

PTWH	Determines which scan is saved. Is not changed.

ADVERBS

0 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 100.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

RECALL	Copies the scan in bin NSAVE into the currently referenced array.

ERRORS

If NSAVE is greater than 100, 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
```

(You may use any numbers between 1 and 100.)

SCALE

SCALE multiplies the currently referenced array by the values of the adverb FACT(RCVR). FACT may be specified separately for each receiver, and may have any real value.

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

ADVERBS

0 FACT(RCVR)	The value by which each receiver is to be multiplied.

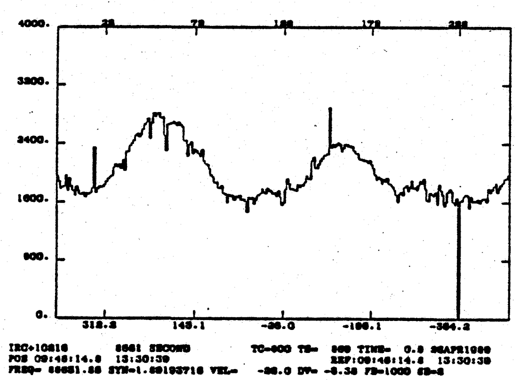
TWO-RECEIVER PROCESSING OPTION Does apply. Either or both receivers may be scaled depending on the processing option. (If both are scaled, they may be scaled by different factors.)

RELATED VERBS

BIAS	Also uses the adverb FACT. BIAS adds FACT(RCVR) to each receiver of the currently referenced array.

EXAMPLE

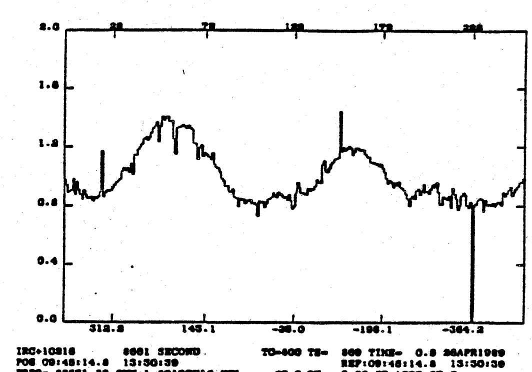
SCALE is often useful when the y-scale is so large that the numbers exceed the formats used by SHOW. If the numbers are too large, you will get



To get the scale down to numbers which fit in the format, specify

```
FACT = 1/2000
SCALE PAGE SHOW
```

and you will get



SCAN

Puts the value of the scan number of the data in the currently referenced array on the value stack. This value may be printed or used in calculations.

EXAMPLE

to verify the scan number of data in the currently referenced array

```
PRINT SCAN
```

SCANANG

Puts the value of the map scanning angle of the data in the currently referenced array on the value stack. Orientation on the sky in the reference frame specified by "XY Reference Frame Code" of the rectangle to be mapped. It is not the angle through the rectangle that the telescope is driven. This provides an alternate capability to that of using descriptive coordinates.

SCLEAR

A routine which clears the accumulator (HOLD) array before stacking data.

PTWH Is not used or changed by SCLEAR.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

ACCUM	Adds the scan in the WORK array to the contents of the hold array with weighting by integration time/system temperature^2.
AVE	Divides the HOLD array by the accumulated weight of the accumulated scans.
SUM	Adds the scan in the WORK array to the contents of the HOLD array with user specified weighting.

EXAMPLE

See ACCUM.

SCRATCH

SCRATCH deletes the specified user defined procedure from the program symbol table, but does not return the procedure or source space.

PTWH Is not used or changed by SCRATCH.

ADVERBS None.

OBJECT The name of the procedure to be deleted.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

See PROCEDURE
RESTART

SETAPPEF

Stores a number on the value stack in the header location for the antenna aperture efficiency (ratio of total power observed to the total power incident on the telescope) of the data in the currently referenced array.

SETAZ

Stores a number on the value stack in the header location for the commanded azimuth of the source or position specified of the data in the currently referenced array.

SETBACKEND

Stores the adverb **SBACKEND** in the header location that describes the backend used for the data in the currently referenced array. **SBACKEND** is an 8-byte character string.

SETBADCHV

Stores a number on the value stack in the header location for the antenna temperature to be assigned to those filterbank channels noted defective of the data in the currently referenced array.

SETBASEOFF

Stores a number on the value stack in the header location for the baseline offset of the data in the currently referenced array.

SETBFWHM

Stores a number on the value stack in the header location for beam full width at half maximum of the data in the currently referenced array.

SETBMORENT

(Not currently implemented)

SETBMTHROW

Stores a number on the value stack in the header location for the beam throw parameter of the data in the currently referenced array.

SETBW

Stores a number on the value stack in the header location for the bandwidth (total bandwidth of the filterbank) of the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETCHAN

Stores the number on the value stack at the specified channel number of the data in the currently referenced array. SETCHAN takes two arguments, the desired channel and the desired data in that order.

EXAMPLE

to replace channel 15 with 0.0

```
15 0.0 SETCHAN
```

SETCOORDCD

Stores the adverb SCOORDCD in the header location which specifies in which coordinate system the data in the currently referenced array was commanded. SCOORDCD is an 8-byte character string.

SETCYCLLEN

Stores a number on the value stack in the header location for the cycle length in seconds (time required to complete one cycle) of the data in the currently referenced array.

SETDELTAX

Stores a number on the value stack in the header location for the step-size along the x-axis (usually velocity) for the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETDELTAXR

Stores a number on the value stack in the header location for DELTA X or X RATE of the data in the currently referenced array. DELTAXR is used to specify the grid cell size in the mapping routines.

See GRID.

SETDELTAYR

Stores a number on the value stack in the header location for the cell size or distance between cells on the Y axis of the data in the currently referenced array. DELTAYR is used to specify the grid cell size in the mapping routines.

See GRID.

SETDESORG

Stores a number on the value stack in the header location for the descriptive origin (horizontal position, vertical position, position angle describing the orientation of the sky "horizontal axis") of the data in the currently referenced array.

SETEL

Stores a number on the value stack in the header location for the commanded elevation of the source or position specified of the data in the currently referenced array.

SETEPOCDEC

Stores a number on the value stack in the header location for the epoch declination of the source or position specified of the data in the currently referenced array.

SETEPOCH

Stores a number on the value stack in the header location for epoch (as specified by the observer) of the data in the currently referenced array.

SETEPOCRA

Stores a number on the value stack in the header location for the epoch of right ascension of the source or position specified of the data in the currently referenced array.

SETFIRSTIF

Stores a number on the value stack in the header location for the FIRST-IF of the data in the currently referenced array.

SETFOCUSH

Stores a number on the value stack in the header location for the horizontal (East-West) focus offset of the data in the currently referenced array.

SETFOCUSR

Stores a number on the value stack in the header location for the radial focus position of the data in the currently referenced array.

SETFOCUSV

Stores a number on the value stack in the header location for the vertical (North-South) focus offset of the data in the currently referenced array.

SETFRAME

Stores the adverb SFRAME in the header location in which the first four characters state whether the grid is polar (POLR) or cartesian (CART) and the second four state whether DELTAXR and DELTAYR refer to STEP sizes or SCANning rates of the data in the currently referenced array. SFRAME is an eight-byte character string.

SETFREQRES

Stores a number on the value stack in the header location for the frequency resolution (bandwidth per channel) of the slected receiver of the spectral line data in the currently referenced array. The receiver is selected by the value of adverb RCVR. If the data is continuum, the value of NT is stored in this header location.

SETFRONTEN

Stores the adverb SFRONTEND in the header location that describes the receiver used for the data in the currently referenced array. SFRONTEND is an 8-byte character string.

SETGAINS

Stores a number on the value stack in the header location indicated above to be the scan number of the calibration data for the data in the currently referenced array.

SETGALLAT

Stores a number on the value stack in the header location for the commanded galactic latitude of the source or position specified of the data in the currently referenced array.

SETGALLONG

Stores a number on the value stack in the header location for the commanded galactic longitude of the source or position specified of the data in the currently referenced array.

SETHARMONI

Stores a number on the value stack in the header location for the harmonic of the data in the currently referenced array.

SETINDX

Stores a number on the value stack in the header location for the horizontal telescope coordinate of the position actually observed (i.e., position measured by horizontal decoder) of the data in the currently referenced array.

SETINDY

Stores a number on the value stack in the header location for the vertical telescope coordinate of the position actually observed (i.e., position measured by vertical decoder) of the data in the currently referenced array.

SETINTTIME

Stores a number on the value stack in the header location for the total integration time for the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETLOFACT

Stores a number on the value stack in the header location for the LO factor (fundamental, doubler or tripler) of the data in the currently referenced array.

SETLOIF

Stores a number on the value stack in the header location for the LO-IF of the data in the currently referenced array.

SETLST

Stores a number on the value stack in the header location for LST (local sidereal time at the start of the observation) of the data in the currently referenced array.

SETNOINT

Stores a number on the value stack in the header location for the total number of integrations for the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETNOPHASE

Stores a number on the value stack in the header location for the number of phases per cycle (different states of switched variables) of the data in the currently referenced array.

SETNOPTS

Stores a number on the value stack in the header location for the number of points (total number of cells in the map) of the data in the currently referenced array.

SETNORCHAN

Stores a number on the value stack in the header location for the number of independent polarizations or channels of this frontend of the data in the currently referenced array.

SETNOSWVAR

Stores a number on the value stack in the header location for the number of beam switching variables of the data in the currently referenced array.

SETNOXPTS

Stores a number on the value stack in the header location for the number of X points (map sample points along the "X-edge" of the rectangle) of the data in the currently referenced array.

SETNOYPTS

Stores a number on the value stack in the header location for the number of Y points (map sample points along the "Y-edge" of the rectangle) of the data in the currently referenced array.

SETOBJECT

Stores the adverb `SOBJECT` in the header location for the source name of the object of the data in the currently referenced array. `SOBJECT` is a 16-byte character string.

SETOBSERVE

Stores the adverb SOBSERVER in the header location for observer name of the data in the currently referenced array. SOBSERVER is a 16-byte character string.

EXAMPLE

```
SOBSERVER = 'JOHN DOE'; SETOBSERVE
```

SETOBSFREQ

Stores a number on the value stack in the header location for the observed frequency of the selected receiver channel of the data in the currently referenced array. The receiver is selected by the value of the RCVR adverb.

SETOBSID

Stores the adverb SOBSID in the header location for observer and operator initials of the data in the currently referenced array. SOBSID is an eight-byte character string.

EXAMPLE

```
SOBSID = '_PRJ_PJR'; SETOBSID
```

SETOBSMODE

Stores the adverb SOBSMODE in the header location for the type of data (4 bytes) and the observing mode (4 bytes) of the data in the currently referenced array. SOBSMODE is a character string of 8 bytes.

SETOBSTOL

Stores a number on the value stack in the header location for the observing tolerance of the data in the currently referenced array.

SETOFFSCAN

Stores a number on the value stack in the header location for the off scan number (the last designated total power off scan) of the data in the currently referenced array.

SETOPENPAR

Stores the adverb SOPENPAR in the header location for open parameters in the currently referenced array. SOPENPAR is a 56-byte character string that the observer may use to describe how he has processed his data.

SETORIENT

Stores a number on the value stack in the header location for the rotation or polarization angle orientation of the receiver or reflector at the prime focus of the data in the currently referenced array.

SETPOLARIZ

Stores the contents of the adverb SPOLARIZ in the header location which describes the type of polarization and the angle for the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETPRECIS

Stores the adverb **SPRECIS** in the header location that describes the precision of the data for the data in the currently referenced array. **SPRECIS** is an 8-byte character string.

SETPROJID

Stores the adverb **SPROJID** in the header location for the program ID as it appears on the telescope schedule of the data in the currently referenced array. **SPROJID** is an eight byte character string.

SETPTCON

Stores a number on the value stack in the selected header location for one of the four pointing constants of the data in the currently referenced array. Individual constants are selected by the value of the adverb **RCVR**.

EXAMPLE

to set the third pointing constant to 15

```
RCVR = 3;    15  SETPTCON
```

SETRAZOFF

Stores a number on the value stack in the header location for the reference azimuth offset in seconds of arc of the data in the currently referenced array.

SETREFPT

Stores a number on the value stack in the header location for the reference point (center channel) of the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETRELOFF

Stores a number on the value stack in the header location for the reference azimuth offset in seconds of arc of the data in the currently referenced array.

SETRESTFRE

Stores a number on the value stack in the header location for the rest frequency of the selected receiver channel of spectral line data in the currently referenced array. The receiver channel is selected by the adverb RCVR. If the data is continuum, the value of the data scale factor is replaced by the number on the value stack.

SETRTSYS

Stores a number on the value stack in the header location for the reference system temperature of the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETRVSYS

Stores a number on the value stack in the header location for the Doppler correction for the earth's motion in the source direction with respect to the velocity reference frame chosen of the data in the currently referenced array.

SETRX

Stores a number on the value stack in the header location for the measured receiver temperature of the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETSAMPRAT

Stores a number on the value stack in the header location for the sample period (time required to complete a single sample) of the data in the currently referenced array.

SETSCAN

Takes a number from the value stack and stores it in the header location for scan number of the data in the currently referenced array.

EXAMPLE

to change the scan number to 2070

```
2070 SETSCAN
```

SETSCANANG

Stores a number on the value stack in the header location for the map scanning angle of the data in the currently referenced array.

SETSIDEBAN

Stores a number on the value stack in the header location for the sideband used for the data in the currently referenced array.

SETSPN

Stores a number on the value stack in the header location for the starting point number for the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETSTSYS

Stores a number on the value stack in the header location for the source system temperature of the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETSYNFREQ

Stores a number on the value stack in the header location for the synthesizer frequency of the data in the currently referenced array.

SETTAMB

Stores a number on the value stack in the header location for the ambient temperature of the data in the currently referenced array.

SETTAUH2O

Stores a number on the value stack in the header location for the opacity of water as computed by a model for the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETTAUO2

Stores a number on the value stack in the header location for the opacity of oxygen as computed by a model for the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETTCAL

Stores a number on the value stack in the header location for the calibration temperature of the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETTELESCO

Stores the adverb STELESCOP in the header location for the telescope name of the data in currently referenced array. STELESCOP is an eight-byte character string.

SETTH2O

Stores a number on the value stack in the header location for the temperature of water as computed by a model for the selected receiver of the data in the currently referenced array. The receiver is selecte by the value of the adverb RCVR.

SETTO2

Stores a number on the value stack in the header location for the temperature of oxygen as computed by a model for the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETTRMS

Stores a number on the value stack in the header location for the standard deviation of the mean source temperature of the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETTSOURCE

Stores a number on the value stack in the header location for the source temperature of the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETTYPECAL

Stores the adverb STYPECAL in the header location that describes the calibration method used for the data in the currently referenced array. STYPECAL is an eight-byte character string.

SETUT

Stores a number on the value stack in the header location for universal time of the data in the currently referenced array.

SETUTDATE

Stores a number on the value stack in the header location for universal time date of the data in the currently referenced array.

SETUXPNT

Stores a number on the value stack in the header location for the user supplied AZ/RA pointing correction of the data in the currently referenced array.

SETUYPNT

Stores a number on the value stack in the header location for the user supplied El/Dec pointing correction of the data in the currently referenced array.

SETVELDEF

Stores the adverb `SVELDEF` in the header location which describes the velocity system of the data in the currently referenced array. `SVELDEF` is an eight-byte character string where the first four characters denote the velocity definition (RADI, OPTL or RELV) and the last four denote the velocity reference (LSR, HELO, EART, BARI or OBS defined).

SETVELOCIT

Stores a number on the value stack in the header location for the velocity with respect to the reference of the data in the currently referenced array.

SETWL

Stores a number on the value stack in the header location noted above for the wavelength used in a continuum focalize scan. `WL` replaces the `BMTHROW` value for continuum data.

SETX0

Stores a number on the value stack in the header location for the velocity at the reference point for the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SETXCELL0

Stores a number on the value stack in the header location for the starting X cell (cell number) of the data in the currently referenced array.

SETXPOINT

Stores the number of the value stack in the header location for the total pointing correction applied in the X (horizontal - AZ/RA) direction of the data in the currently referenced array.

SETXREF

Stores a number on the value stack in the header location for the commanded reference Xcoord (horizontal coordinate of the reference position in the system specified by the observer) of the data in the currently referenced array.

SETXSOURCE

Stores a number on the value stack in the header location for the commanded source Xcoord (the horizontal coordinate of source or position in the coordinate system specified by the observer) of the data in the currently referenced array.

SETXZERO

Stores a number on the value stack in the header location for the horizontal telescope coordinate at the map reference position of the data in the currently referenced array.

SETYCELL0

Stores a number on the value stack in the header location for the starting Y cell (cell number) of the data in the currently referenced array.

SETYPOINT

Stores a number on the value stack in the header location for the total pointing correction applied in the Y (vertical - El/Dec) direction of the data in the currently referenced array.

SETYREF

Stores a number on the value stack in the header location for the commanded reference Ycoord (vertical coordinate of the reference position in the coordinate system specified by the observer) of the data in the currently referenced array.

SETYSOURCE

Stores a number on the value stack in the header location for the commanded source Ycoord (vertical coordinate of source or position in the coordinate system specified by the observer) of the data in the currently referenced array.

SETYZERO

Stores a number on the value stack in the header location for the Y position at Zero of the data in the currently referenced array.

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	Temperature spectrum	Kelvins
x axis	Top of graph Bottom of graph	Velocity (km/sec) Frequency (MHz)

SHOW will compute automatically 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.

PTWH	Determines which array is plotted. Is not changed.

ADVERBS

0 BDROP(RCVR) 0 EDROP(RCVR)	The number of channels at each end of each receiver which will not be plotted.
0 BMARK	If = 1, the regions last used to compute a baseline or RMS will be indicated by boxes.
0 CMARK(N)	N is an integer between 1 and 6. If CMARK(N) > 0, a vertical line will be drawn at channel CMARK(N).
-9999 FMARK(N)	N is an integer between 1 and 6. If FMARK(N) > -9999, a vertical line will be drawn at frequency FMARK(N) IN MHz.
0 SLABEL	Determines where scan number and source name are printed on the display.
-9999 TMARK(N)	N is an integer between 1 and 6. If TMARK(N) > -9999, a horizontal line will be drawn at temperature TMARK(N) in degrees K.
-9999 VMARK(N)	N is an integer between 1 and 6. If VMARK(N) > -9999, a vertical line will be drawn at velocity VMARK(N) in km/sec.

- continued -

0 `YINCR(RCVR)`	The user may specify how many degrees are to be between each y tic mark. If YMIN > -9999, SHOW will not compute the y-scaling, but will use the values specified by `YMIN` and `YINCR`.
0 `YMIN(RCVR)`	The minimum temperature that is to appear on the graph. See note under `YINCR`.
0 `ZLINE`	If = 1, a horizontal line will be drawn at zero degrees K.

TWO-RECEIVER PROCESSING OPTION Does affect `SHOW`. If the processing option is `BOTH`, both receivers will be displayed ignoring the values of `BDROP` and `EDROP`. If the option is `ONLYA`, receiver A will be shown taking into account the values of `BDROP` and `EDROP`. If the option is `ONLYB`, receiver B will be shown taking into account the values of `BDROP` and `EDROP`.

RELATED VERBS

`CV` or `VC`	Sets the upper x-axis coordinate to channels and the lower x-axis coordinate to velocity.
`FLAG`	Draws vertical lines at specified frequency in MHz.
`FV` or `VF`	Sets the upper x-axis coordinate to velocity and the lower x-axis coordinate to frequency.
`HISTOGRAM`	Causes a `HISTOGRAM` plot to be drawn by `SHOW`.
`LINE`	Causes a `LINE` plot to be drawn by `SHOW`.
`RANGE`	Defines the values of `YMIN` and `YINCR` according to the specified max and min values.

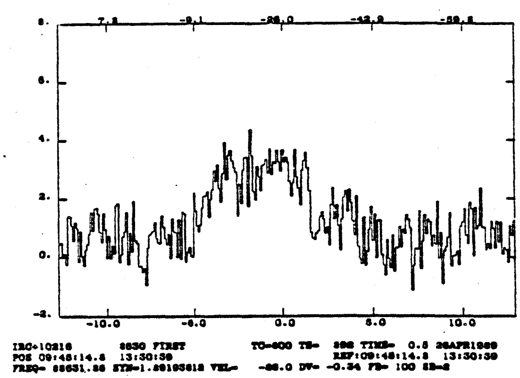
- continued -

EXAMPLE

After retrieving scan 8530 from disk, one may display the scan by

```
PAGE SHOW
```

and one will get



SIDEBAND

Puts the value of the sideband used for the data in the currently referenced array on the value stack.

SLIDE

`SLIDE` is used on two-receiver data. B is shifted by `RSHIFT` channels with respect to A, each receiver weighted by integration time divided by system temperature squared. The resulting weighted average is in A receiver.

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

ADVERBS

0 `RSHIFT` — The number of channels by which receiver B will be shifted when averaged with receiver A to properly align the spectral features.

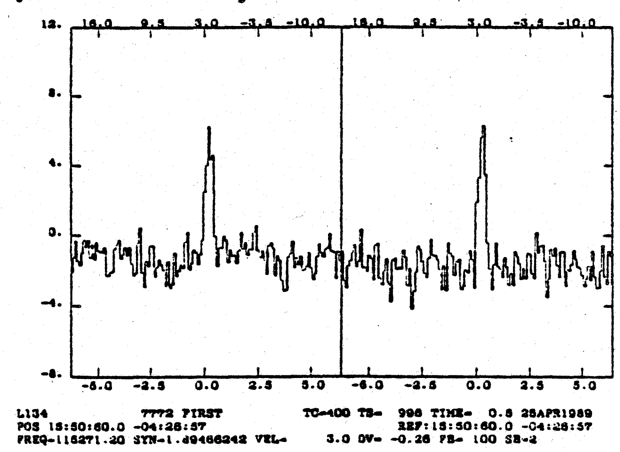
TWO-RECEIVER PROCESSING OPTION — The processing option should be `BOTH` when `SLIDE` is used.

RELATED VERBS

`RDIFF` — Subtracts receiver B from receiver A, without weighting, and puts the result in the receiver A channels.

EXAMPLE

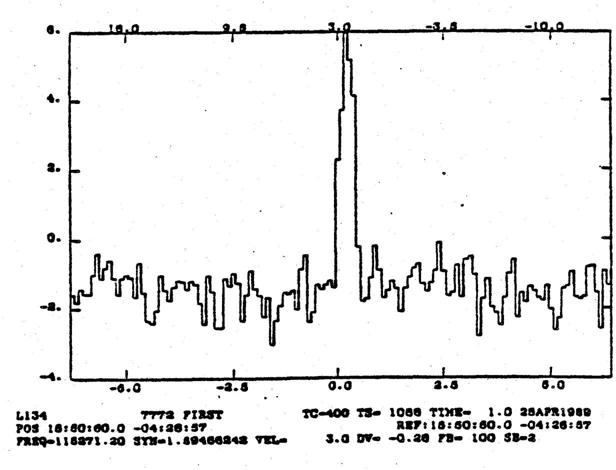
You have a two-receiver scan in the currently referenced array. Receiver A and B look like



you specify

```
SLIDE ONLYA
PAGE SHOW
```

Receiver A now looks like



SMOOTH

SMOOTH smoothes the scan in the currently referenced array by the weight function SMWGT specified by the user.

PTWH	Determines which scan is to be smoothed.

ADVERBS

0 BDROP(RCVR) 0 EDROP(RCVR)	The number of channels at each end of each receiver which will be ignored.
0 SMWGT(12)	The smoothing function array. The first element of the array specifies the number of elements in the function - not greater than 11.

TWO-RECEIVER PROCESSING OPTION Does affect SMOOTH. Either or both receivers may be smoothed, depending on the processing option.

RELATED VERBS

BOXCAR	Smoothes a scan by averaging a specified number of channels together.
HANNING	A smoothing routine which averages three channels together with the center channel getting twice as much weight as either side channel.

EXAMPLE

```
SMWGT(1) = 2
SMWGT(2) = .5
SMWGT(3) = .5
SMOOTH
```

See SMWGT

SNAP

SNAP is an informational verb which prints the header information for the currently referenced array on the printer.

PTWH	Determines which header is printed. Is not changed.
ADVERBS	None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

HEADER	Prints the same documentation on the CRT.
TABLE	Prints the channel by channel values for the currently referenced array.

ERRORS

If there is no scan in the currently referenced array when SNAP is specified, the message "LOAD SCAN" will be generated.

EXAMPLE

See HEADER

SOLVETIP

(Documentation not available)

SORT

Sorts spectral line data in observer's raw data file and fills stack (ASTACK) with scan numbers that meet the sort criteria. Sort by one or more of the below:

Other relevant parameters or adverbs, with initial value (if any):

BSCAN	Beginning scan number for the sort.
ESCAN	Ending scan number for the sort.
SOURCE	Source name used for the sort.
FREQ(2)	Beginning and ending rest frequency of the sort.
VEL(2)	Beginning and ending velocity of the sort.

SPIKE

A routine which searches for bad channels according to the criteria, absolute value of channel value > NSIGMA. SPIKE will find up to 12 bad channels and print them on the CRT.

PTWH Determines which array is to be used.

ADVERBS

0 BADPT The bad channels. Set by SPIKE.

0 NSIGMA The maximum value for determining bad channels.

TWO-RECEIVER PROCESSING OPTION Does affect SPIKE. Bad channels may be found in one or both receivers depending on the processing option.

RELATED VERBS

REPLACE	Replaces bad channels by a parabolic interpolation over adjacent channels.
RMS	Computes the RMS over the specified regions in the currently referenced array and stores the result in VRMS(RCVR).

EXAMPLE

See BADPT adverb.

SPN

Puts the value of the starting point number for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

SPREAD

This routine will restrict data processing to only those channels within the minimum and maximum frequency or velocity specified. The units depend on the coordinate of the lower x-axis. If the lower x-axis is velocity, the units are KM/S and if the coordinate of the lower x-axis is frequency, the units are MHz.

PTWH Is not used or changed by SPREAD.

ADVERBS

0 BDROP(R)	The number of channelsat each end of the spectrum
0 EDROP(R)	which will be ignored.

OBJECTS The minimum and maximum frequencies or velocities that limit the SPREAD.

TWO-RECEIVER PROCESSING OPTION Does affect SPREAD. Either or both receivers will be spread depending on the processing option.

RELATED VERBS

LIMIT — Limits the processing of data to only those channels between the minimum and maximum values specified.

REMARKS

SPREAD 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 ALL procedure.

ERRORS

If SPREAD is specified without minimum and maximum values, the message

"ARG LIST ?"

will be printed.

EXAMPLE

```
PROCEDURE XSET
X1 = VCUR
X2 = VCUR
X1 X2 SPREAD
FINISH
```

See VCUR.

STACK

A routine which lists all scans currently in the STACK.

PTWH Is not used or changed.

ADVERBS Uses ASTACK and ACOUNT to list the scans in the STACK.

TWO-RECEIVER OPTION Does not apply.

RELATED VERBS

A#	Inserts one scan into the STACK.
ADD (#,#)	Inserts a series of scans into the STACK.
DELETE #	Delets one scan from the STACK.
EMPTY	Empties the STACK.

EXAMPLE

After defining the STACK by

```
500 A;  506 A
512  520  ADD
```

STACK will produce the following output:

```
500.  506.  512.  514.  516.  518.  520.
```

STORE

STORE page# copies the program memory 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.

PTWH Is not used or changed.

ADVERBS (No adverbs are changed)

OBJECTS

Page# The disk page into which the copy of memory is to be put. Page# must be an integer between 1 and 3.

TWO-RECEIVER PROCESSING OPTION Does not apply.

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. So you pick a number between 1 and 3, say 3, and specify

```
STORE 3
```

Now in disk page 3 there is a copy of memory exactly like the present copy.

STSYS

Puts the value of the source system temperature of the selected receiver of the data in the currently referenced array. The receiver is selected by the value of the adverb RCVR.

SUM

A routine that is used to stack scans in the HOLD array with user specified weighting. The first use of SUM after an AVE, SCLEAR, or program restart copies the header information and the spectral values in the WORK array to the HOLD array. The spectral values are multiplied by WEIGHT(RCVR) * INTEGRATION(RCVR). Subsequent uses of SUM add the spectral values contained in the WORK array (multiplied by WEIGHT(RCVR)* INTEGRATION(RCVR) 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 summed each time a scan is accumulated. SUM increments the internal stack counter each time a scan is accumulated, and stores the numbers of the first 34 and the last scan accumulated.

PTWH Does not affect SUM. Is set to the HOLD array.

ADVERBS

1 ASHIFT(RCVR) The number of channels by which a scan is shifted by subsequent calls to SUM. On the first call to SUM, the scan is not shifted even if ASHIFT $\neq$ 0.

1 WEIGHT(RCVR) Weighting to be applied to each scan.

TWO-RECEIVER PROCESSING OPTION Does affect SUM. Either or both receivers may be summed, depending on the processing option.

RELATED VERBS

ACCUM Adds the scan in the WORK array to the contents of the HOLD array, with weighting by integration time/system temperature^2.

- continued -

ALIGN	Sets the values of ASHIFT, using the header information in the WORK array, so that when the scan in the WORK array is summed, its velocities or frequencies will be correctly aligned with the velocities or frequencies in the HOLD array.
ALIGNF	Sets the alignment mode to frequency.
ALIGNV	Sets the alignment mode to velocity.
SCLEAR	Clears the accumulator (HOLD) array before stacking data.
TELL CSTACK	Prints the internal stack counter and the numbers of the scans accumulated.

EXAMPLE

Suppose you want to stack several scans and weight by $e^{-\tau}$. You might define the following procedure:

```
PROCEDURE STACKER
FOR I = 1 TO ACOUNT
      XSCAN = ABS(ASTACK(I)):  GET XSCAN
      WEIGHT = EXP(-(1/COS(90.-TWH(31,PTWH))*TWH(34,PTWH)))
      ALIGN SUM END
AVE PAGE SHOW
RETURN
FINISH
```

where TWH(31,PTWH) is the elevation angle of the scan and TWH(34,PTWH) is the atmospheric attenuation.

SWITCHED

Computes switched power data from the raw phases in the continuum digital backend scan in the currently referenced array.

See `TOTALPWR`, `AVGD`

EXAMPLE

```
CGET 500 SWITCHED AVGD PAGE SHOW
```

SYNFREQ

Puts the value of the synthesizer frequency of the data in the currently referenced array on the value stack.

SYSHELP

Invokes the VMS DCL command `HELP` which makes all of the system help files available to the user.

TABLE

TABLE is an informational verb. TABLE prints the antenna temperature (Kelvins) for each channel (from 1 to 256) and labels the information with scan number and filterbank.

PTWH Determines for which scan the table is printed. Is not changed.

ADVERBS None.

OBJECTS None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

HEADER Prints the header information for the scan in the currently referenced array.

EXAMPLE

The user wants a list of the spectral values of scan 8661. He specifies

```
GET 8661 PAGE TABLE
```

and gets

```
>TABLE
8661 SECOND FB 1000
```

0.9800	0.9426	0.8908	0.9029	0.9073	0.9833	0.8799	0.9601
0.9072	0.8759	0.8452	0.9063	0.8767	0.8551	0.8841	0.8618
0.8351	0.8570	0.8541	0.8640	0.8960	1.1730	0.8666	0.8913
0.9039	0.9064	0.9072	0.9217	0.9433	0.9631	0.9703	0.9826
1.0350	1.0567	1.0405	1.0582	1.0274	1.0891	1.0157	1.1460
1.1442	1.1865	1.2027	1.2232	1.2536	1.2682	1.2479	1.2639
1.2990	1.3332	1.3447	1.3709	1.2348	1.3408	1.4092	1.3842
1.4063	1.4094	1.3740	1.3797	1.2563	1.1519	1.3318	1.3333
1.3433	1.3533	1.3403	1.3264	1.3438	1.3270	1.3198	1.2231
1.2218	1.1159	1.1955	1.2117	1.1340	1.1577	1.1479	1.1323
1.1125	1.1576	1.0823	1.0622	1.0536	1.0269	0.9841	0.9723
0.9338	0.9340	0.8971	0.8990	0.9102	0.8606	0.8535	0.7978
0.8504	0.8457	0.8459	0.8180	0.8382	0.8373	0.8440	0.8182
0.8204	0.8075	0.7343	0.8347	0.8258	0.8319	0.7932	0.8216
0.8480	0.8503	0.8625	0.8900	0.8685	0.8617	0.8944	0.8802
0.8756	0.8686	0.8643	0.8411	0.8793	0.8395	0.7842	0.8019
0.8911	0.9630	0.9267	0.8903	0.8910	0.8849	0.8822	0.9218
0.9078	0.9194	0.9184	0.9379	0.9826	0.9654	0.9618	1.0861
1.1096	1.0380	1.0361	1.0621	1.0894	1.1014	1.1226	1.1264
1.1318	1.4499	1.1764	1.1968	1.1808	1.1934	1.2077	1.1946
1.1467	1.1715	1.1938	1.1938	1.1863	1.1802	1.1782	1.1515
1.1334	1.1303	1.0994	1.0965	1.0928	1.0834	1.0846	1.0712
1.0860	1.0568	1.0303	0.9886	0.9556	0.9745	0.9641	0.9298
0.9692	0.9465	0.9636	0.9286	0.9218	0.9070	0.8835	0.8504
0.8057	0.8582	0.8721	0.8619	0.9059	0.9179	0.8800	0.9002
0.9472	0.8980	0.9014	0.8862	0.9188	0.9494	0.9315	0.9603
0.8957	0.8154	0.8102	0.8634	0.8566	0.8454	0.8709	0.7881
0.9217	0.8822	0.8624	0.7784	0.8053	0.8706	0.8916	0.8737
0.8066	0.7842	0.8286	0.0070	0.8088	0.8148	0.8420	0.8538
0.8467	0.7668	0.8392	0.8083	0.8104	0.8100	0.8203	0.8005
0.8191	0.8280	0.8271	0.8098	0.8941	0.8693	0.8893	0.9103

TAMB

Puts the value of the ambient temperature of the data in the currently referenced array on the value stack.

TAUH2O

Puts the opacity of water as computed by a model for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

TAUO2

Puts the value of the opacity of oxygen as computed by a model for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

TCAL

Puts the value of the calibration temperature of the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

TCUR

This routine activates the horizontal crosshair and returns the temperature value for that crosshair position on the current display when any key except RETURN is struck.

PTWH Is not used or changed by TCUR.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

CCUR	Returns the channel value 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.
VCUR	Returns the velocity value at 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
```

TELESCOP

Puts a field of eight characters which contain the telescope name of the data in the currently referenced array into the adverb STELESCOP.

EXAMPLE

to query the telescope name

```
TELESCOP;  PRINT STELESCOP
```

TELL

TELL is an informational verb requiring an object. The six permitted objects of TELL are DATA, DISK, CSTACK, ESCANS, KSCANS, and RECORDS. Lists information about data in disk files including PDFL(raw), IRFL(ind rec), PKFL (KEEP and edited) files.

PTWH	Is not used or changed.
ADVERBS	None.

OBJECTS

DATA	Prints number of scans and remaining space PDFL.
DISK	Lists LINe then CONTINUUMscan numbers in PDFL file.
CSTACK	Is used to request a listing of the scans that have been accumulated. Only the numbers of the first and the last scans that were accumulated and number of spaces stacked will be printed.
ESCANS	Lists EDITED scans in the PKFL file.
KSCANS	Lists KEEP scans in the PKFL file.
RECORDS	Printsnumber of scans and remining space in IRFL.

- continued -

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS None.

The output of TELL DISK looks like:

```
›TELL DISK

 LIST OF SPECTRAL LINE SCANS ON DISK FOR  WWS
   596    598    600    602    604    606    608   5858   1672   1674
  1676   1678   1680   1682   1684   1686   1688   1690   1692   1694
  1696   1698   1700   1702   1704   1706   1708   1710   1712   1714
  1716   1718   1720   1722   1724   1726   1728   1730   1734   1736
  1738   1740   1742   1744   1746   1748   1750   1752   1754   1756
  1758   1760   1762   1764   1766   1768   1770   1772   1774   1776
  1778   1780   1792   1794   1796   1798   1800   1802   1814   1826
  1860   1862   1864   1866   4696   4698   4700   4702   4704   4708
  4710   4712   4714   4716   4718   4720   4722   4724   4726   4728
  4730   4732   4734   4736   4738   4740   4742   4744   4748   4750
  4752   4754   4756   4758   4760   4762   4764   4766   4768   4770
  4772   4774   4776   4778   4780   4782   4784   4786   4788   4790
  4792   4794   4796   4798   4800    686
 LIST OF CONTINUUM SCANS ON DISK FOR      WWS
   536    537    538    539    540    541    542    543    544    545
   546    547    548    549    550    551    552    553    554    555
   556    557    558    559    560    561    562    563    564    565
   566    567    568    569    570    571    572    573    574    575
   576    577    578    579    580    581    582    583    584    585
   586    587    588    589    590    591    592    593    594    595
  3868   3869   3870   3871   3872   3873   3874   3875   3876   3877
  3878   3879   3880   3881   3882   3883   3884   3885   3886   3887
  3888   3889   3890   3891   3892   3893   3894   3895   3896   3897
  3898   3899   3900   3901   3902   3903   3904   3905   3906   3907
  3908   3909   3910   3911   3912   3913   3914   3915   3916   3917
  3918   3919   3920   3921   3922   3923   3924   3925   3926   3927
  3928   3929   3930   3931   3932   3933   3934   3935   3936   3937
  3938   3939   3940   3941   3942   3943   3944   3945   3946   3947
  3948   3949   3950   3951   3952   3953   3954   3955   3956   3957
  3958   3959   3960   3961   3962   3963   3964   3965   3966   3967
  3968   3969   3970   3971   3972   3973   3974   3975   3976   3977
```

If you have accumulated 29 scans (1072 - 1128), the output of TELL CSTACK would be:

```
TELL STACK
29
1700 1756
```

TEMP

Computes spectral line quotients:

```
((Sig(i)-Ref(i))/(Ref(i)-Zeros(i)))*Gains(i)
```

from the total power data in the WORK and TEMP arrays, and the calibration data. The data must be loaded as follows:

```
WORK array = SIGNAL (ON)
TEMP array = REFERENCE (OFF)
HOLD array = ZERO levels
SPECT array = GAINS
```

TH20

Puts the value of the temperature of water as computed by a model for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

THEN

Used in procedures in IF, THEN, ELSE logical constructions.

See IF, ELSE, END, FOR, WHILE

EXAMPLE

```
IF X = 2 THEN; Y = 3; END
```

THUMB

Prints main and reference azimuth, elevation, right ascension and declination offsets for the scan in the currently referenced array.

TITLE

TITLE writes one line of documentation about the current scan on the CRT screen including scan number, LST, source name, rest frequency, filterbank and sideband number.

PTWH Determines which array is used by TITLE.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not apply

RELATED VERBS

DOC — Writes one line of header information about the scan in the currently referenced array.

HEADER — Prints header information about the currently referenced array on the CRT.

SNAP — Prints header information about the currently referenced array on the printer.

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 TITLE PRINT SIZE
FINISH
```

TO

Used in procedures in FOR, TO, BY logical construction.

EXAMPLE

```
FOR X = 1 TO 3 BY 2; ... ; END
```

TO2

Puts the value of the temperature of oxygen as computed by a model for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

TOTALPWR

Computes the total power data from the raw phases of the continuum digital backend scan in the currently referenced array.

See SWITCHED, AVGD

EXAMPLE

```
CGET 500 TOTALPWR AVGD PAGE SHOW
```

TRMS

Puts the value of the standard deviation of the mean source temperature of the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

TRT TRW TRH

These routines copy the contents of the currently referenced array to the indicated array ans sets PTWH to that array as well.

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.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS None.

EXAMPLE

The most common use of TRT is to create a copy of a scan to be replaced by a baseline or gaussian model, leaving one copy of the scan untouched in the WORK array. For example:

```
GET 710 PAGE SHOW
TRT
BSHAPE BMODEL RESHOW
```

TRX

Puts the value of the measured receiver temperature of the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

TSOURCE

Puts the value of the source temperature of the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

TYPE

Invokes the VMS DCL command TYPE. ASCII files outside of the LINE program may be listed on the CRT. TYPE will prompt for the name of the file to be typed.

TYPECAL

Puts a field of eight characters which describe the calibration method used for the data in the currently referenced array into the adverb STYPECAL. This adverb may then be printed.

UT

Puts the value of universal time of the data in the currently referenced array on the value stack.

UTDATE

Puts the value of the universal time date of the data in the currently referenced array on the value stack.

UXPNT

Puts the value of the user Az/RA pointing correction of the data in the currently referenced array on the value stack.

UYPNT

Puts the value of the user EL/Dec pointing correction of the data in the currently referenced array on the value stack.

VC or CV

A routine which sets an internal flag so that `SHOW` displays channel numbers for the upper x-axis labeling and velocity for the lower x-axis labeling.

PTWH Is not used or changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

RELATED VERBS

`VF` or `FV` — Sets an internal flag so that `SHOW` displays velocity for the upper x-axis labeling and frequency for the lower x-axis labeling.

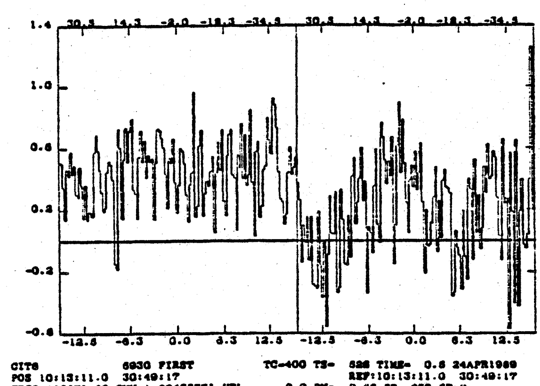
`SHOW` — Displays a scan with user specified or default scaling and user specified or default labeling.

REMARKS

The default labeling in `SHOW` for the lower x-axis is frequency and for the upper x-axis is velocity.

EXAMPLE

Your current display looks like

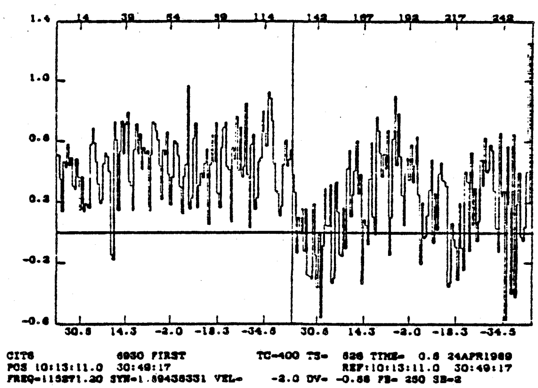


- continued -

and you want to change your x-axis to velocity and channel numbers. Type:

```
VC XX
```

and you will get



VCTR

Draws a line from the current cursor position to the position commanded by IX & IY.

Other relevant parameters or adverbs, with initial value (if any):

IX	One of 1024 horizontal points on the CRT screen.
IY	One of 780 vertical points on the CRT screen. (Used in procedures)

See PLACE, CHAR

EXAMPLE

```
VCTR (400,200)
```

VCUR

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

PTWH Is not used or changed by VCUR.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does apply. Will only compute the proper velocity for one receiver at a time.

RELATED VERBS

CCUR	Returns the channel value 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 positions.
TCUR	Returns the temperature value at the current horizontal crosshair position.

EXAMPLE

You have displayed a scan with velocity as 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
```

VELDEF

Puts a field of eight characters which describe the velocity system of the data in the currently referenced array into the adverb, SVELDEF. The first four characters denote the velocity definition (RADI, OPTL or RELV) and the last four denote the velocity reference (LSR, HELO, EART, BARI or DBS defined). The adverb, SVELDEF, may then be printed.

VELOCITY

Puts the value of the velocity with respect to the reference of the data in the currently referenced array on the value stack. The source velocity specified by the observer relative to the velocity reference frame.

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 completely similar to the DO WHILE looping facility in PL/I.

The (test condition) can be any expression which has a single true or false result. As long as (test condition) 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.

PTWH Is not used or changed.

ADVERBS None.

TWO-RECEIVER PROCESSING OPTION Does not apply.

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
```

IF Begins a logical construction of the form

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

- continued -

See PROCEDURE

EXAMPLES

(1) WHILE construction on severallines 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 linein 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
```

X0

Puts the value of the velocity at the reference point for the selected receiver of the data in the currently referenced array on the value stack. The receiver is selected by the value of the adverb RCVR.

XCELL0

Puts the starting X cell (cell number) of the data in the currently referenced array on the value stack. May be positive, negative, or zero. It is used to define the position of the rectangle to be mapped with respect to the reference position which, by definition, is (x,y) = (0.0).

XCUR

Returns the X position of the cursor in terms of TEK points. Used in procedures.

See PLACE, VCTR, CHAR, YCUR

EXAMPLE

```
IX=XCUR
```

XPOINT

Puts the value of the total pointing correction applied in the X (horizontal - Az/RA) direction of the data in the currently referenced array on the value stack.

XREF

Puts the value of the commanded reference Xcoord (horizontal coordinate of the reference position in the system specified by the observer) of the data in the currently referenced array on the value stack.

XSOURCE

Puts the value of the commanded source Xcoord (the horizontal coordinate of source or position in the coordinate system specified by the observer) of the data in the currently referenced array on the value stack.

XZERO

Puts the value of the horizontal telescope coordinate at the map reference position of the data in the currently referenced array on the value stack. Together with "Y position at Zero", this defines the cell (X,Y) = (0,0) at the map reference position.

YCELL0

Puts the value of the starting y cell (cell number) of the data in the currently referenced array on the value stack.

YCUR

Returns the Y position of the cursor in terms of TEK points. Used in procedures.

See **PLACE, VCTR, CHAR, XCUR**

EXAMPLE

```
IY=YCUR
```

YPOINT

Puts the value of the total pointing correction applied in the Y (vertical - EL/Dec) direction of data in the currently referenced array on the value stack.

YREF

Puts the value of the commanded reference Ycoord (vertical coordinate of the reference position in the coordinate system specified by the observer) of the data in the currently referenced array on the value stack.

YSOURCE

Puts the value of the commanded source Ycoord (vertical coordinate of source or position in the coordinate system specified by the observer) of the data in the currently referenced array on the value stack.

YZERO

Puts the value of the Y position at Zero of the data in the currently referenced array on the value stack.

ZGET

Copies current zero check or specified zero check into the WORK array. Specify 1 or 2 for the first or second filterbank if the current zero check is desired. If a previous zero check is desired, specify its scan number.

EXAMPLE

```
ZGET 1
```

ADVERB SYNOPSIS

ACOUNT	The stack counter.
ASHIFT (RCVR)	The number of channels by which a scan must be shifted so that its velocities or frequencies align with the velocities or frequencies in the HOLD array. Used by ACCUM or SUM; set by ALIGN.
ASTACK (100)	The 100 element STACK array.
BADPT (12)	The bad channels found by SPIKE and altered by REPLACE.
BBASE (RCVR)	The number of channels at the left side of the spectrum which will be used to compute a baseline or RMS.
BDROP (RCVR)	The number of channels at the left side of the spectrum 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 (RCVR)	The channel at which MOMENT will begin to compute.
CENTER (6)	The channels at which up to six gaussians have their centers.
CLIPMAX (RCVR)	The maximum spectral value desired.
CLIPMIN (RCVR)	The minimum spectral value desired.
CMARK (6)	The channels at which up to six vertical lines are to be drawn by SHOW.
DF	The amount of shift between the signal reference spectra in MHz for frequency switched data.
EBASE (RCVR)	The number of channels at the right end of the spectrum which are to be used to compute a baseline or RMS.
EDROP (RCVR)	The number of channels at the right end of the spectrum which are not to be used.
EGAUSS	The channel at which GAUSS is to stop fitting.

EMOMENT (RCVR)	The channel at which MOMENT is to stop computing.
FACT (RCVR)	A number of Kelvins to be added to a scan, or a numerical factor to be multiplied with a scan.
FIXC	If >=1, the values of CENTER will not be iterated by GAUSS.
FIXHW	If >=1, the values of HWIDTH will not be iterated by GAUSS.
FMARK (6)	The frequencies at which up to six vertical lines are to be drawn by SHOW.
FR	Reference frequency of a frequency switched scan.
FS	Signal frequency of a frequency switched scan.
HEIGHT (6)	The height, in Kelvins, of up to six gaussians.
HWIDTH (6)	The half-width, in number of channels, of up to six gaussians.
LEVS (20)	Up to twenty levels at which a contour is to be drawn.
MRATIO	The ratio between the horizontal and vertical lengths of a contour map.
NBOX	The number of channels 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	The order of the moment which is to be computed by MOMENT.
NREGION(RCVR,8)	The channels, in sequential order, at which up to four regions per receiver are to begin and end.
NSAVE	Names a disk save bin.
NSIGMA	The value used to find BADPT's in SPIKE.
RFREQ	The approximate frequency, in MHz, of the sinusoid to be 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.
SHIFT	The amount of shift between scans shown in profile form by the SHOW1 and

	SHOW2 procedures.
SLABEL	A flag to determine where SHOW writes certain documentation information on the display.
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 velocities at which up to six vertical lines are to be drawn by SHOW.
WEIGHT (RCVR)	The weight used by SUM in accumulating data.
YINCR (RCVR)	The distance between Y ticmarks that is to be used by SHOW.
YMIN (RCVR)	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.

ADVERBS

ACOUNT

A counter for the **ASTACK** adverb. Determines how many scans are in the STACK. Is used by **STACK** verb, and in procedures.

ACOUNT is modified by

A **ADD** **DELETE** **EMPTY**

The initial value of **ACOUNT** is 0, which means that the stack is empty.

EXAMPLE

```
EMPTY
500    508    ADD
```

The value of **ACOUNT** will be 5. The scans inserted into the stack are incremented by 2.

ASHIFT

The number of channels by which a spectrum must be shifted so that when it is accumulated, the points at the same velocities or frequencies are added together.

ASHIFT is used by

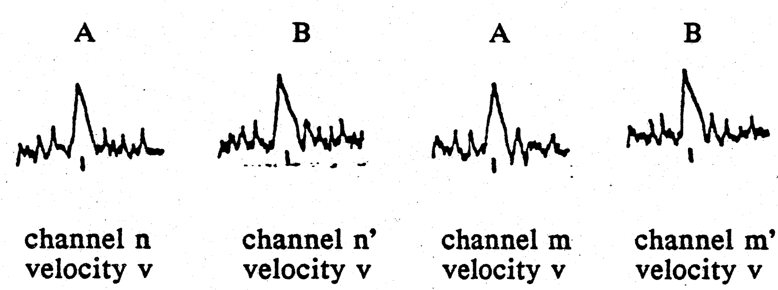
    ACCUM SUM

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 value of ASHIFT is determined by

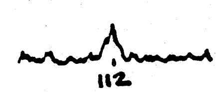
WORK array	HOLD array	ASHIFT(1)=n-m ASHIFT(2)=n'-m'



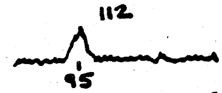
EXAMPLES

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
```

ASTACK

The STACK that is listed by STACK and used in procedures to operate on up to 100 scans.

STACK is modified by

A ADD DELETE EMPTY

The initial values of ASTACK are 0.

Which means 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
```

600. 602. 604. 606. 608. 610. 612. 614.

BADPT

Channels in the filterbank that are considered bad. The channels are set by `SPIKE` and removed by `REPLACE`.

`BADPT` is used by

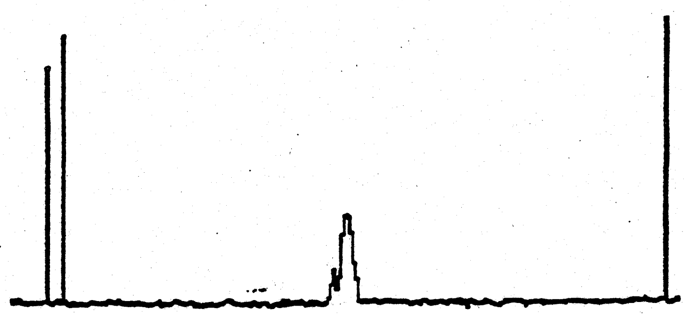
`SPIKE` `REPLACE`

The initial values of `BADPT` (12) are 0.

Which means that no bad channels have been found or will be replaced.

EXAMPLE

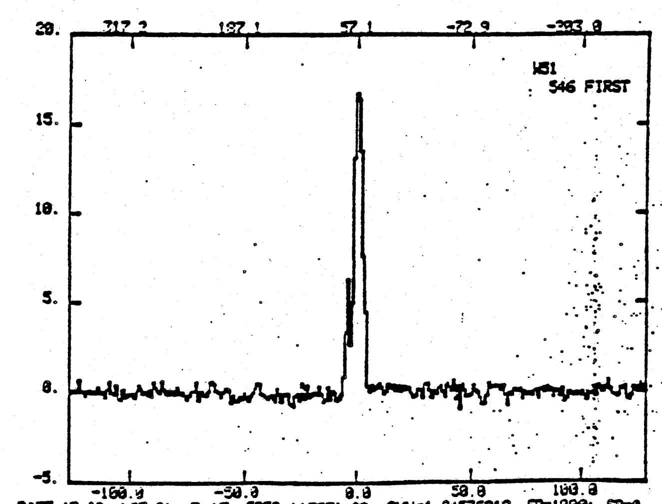
Your data looks like



and you wish to remove the spikes. You specify

```
NSIGMA = 20; SPIKE REPLACE PAGE SHOW
```

Your display will then look like



BBASE EBASE

The number of channels at each end of the spectrum to be used to compute a baseline. BBASE is the number of channels at the beginning of the spectrum; EBASE is the number of channels at the end of the spectrum.

BBASE and EBASE are used by

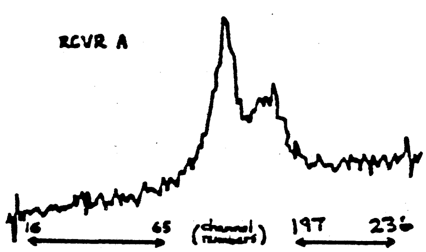
BASELINE	BMODEL	BSHAPE	DCBASE
RIPPLE	RMODEL	RSHAPE	RMS

The initial values of BBASE (RCVR) and EBASE (RCVR) are 50.

The default 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 spectrum looks like (you are using two receivers)



You want to use these regions to fit the baseline(you want to use the corresponding regions or the B receiver).

so you specify

```
BDROP = 15
BBASE = 50
EBASE = 40
EDROP = 20
```

before calling a baseline verb.

BDROP EDROP

The number of channels at each end of the spectrum which will be ignored by the following routines:

BDROP and EDROP are used by

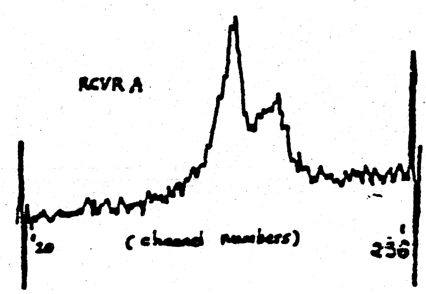
SHOW	RESHOW	MAPSHOW	LABEL
BMODEL	BASELINE	BSHAPE	DCBASE
RIPPLE	RMODEL	RSHAPE	
BOXCAR	HANNING		

The initial values of BDROP (RCVR) and EDROP (RCVR) are 0.

The default 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 A and B receivers.

EXAMPLE

Your spectrum looks like



To get rid of the glitches, you specify

```
BDROP = 20
EDROP = 20
```

This drops 20 channels at each end of each receiver.

```
PAGE SHOW
```

Will then plot the spectrum without the glitches.

BGAUSS EGAUSS

The explicit channel numbers where a gaussian fit is to begin and to end.

BGAUSS and EGAUSS are used by

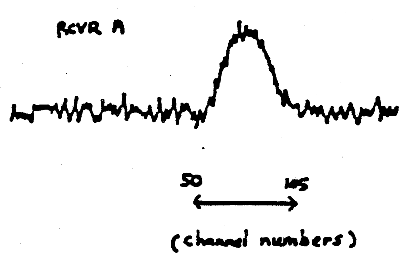
GAUSS GMODEL GPARTS

The initial values of BGAUSS and EGAUSS and 1 and 256.

PEAK will set the values of BGAUSS and EGAUSS or the user may specify their values before using GAUSS, GMODEL or GPARTS. BGAUSS and EGAUSS may be specified for only one receiver.

EXAMPLE

Your spectrum looks like



and you want to fit a gaussian over this region,

so you specify

BGAUSS = 50

EGAUSS = 105

BMARK

This adverb acts as a switch. If BMARK = 1, the verb SHOW will draw boxes indicating the baseline fitting regions last used. If BMARK = 0, the boxes will not be displayed.

BMARK is used by

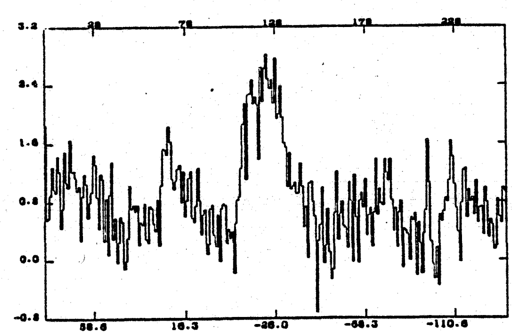
SHOW

The initial value of BMARK is 0.

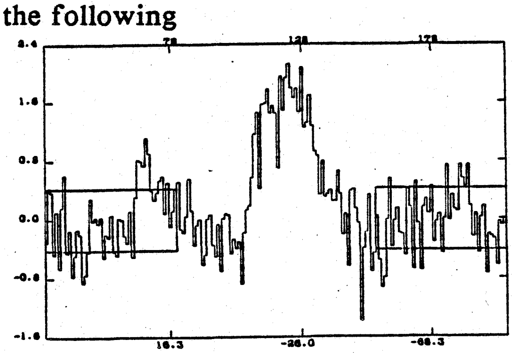
The user can change BMARK as he desires. No verbs change BMARK. BMARK cannot be specified separately for each receiver.

EXAMPLE

If BMARK = 0, BASELINE PAGE SHOW produces



but is BMARK = 1 in the same situation, you will get the following



where the boxes delineate the parts of the spectrum used to calculate the baseline (or RMS).

BMOMENT EMOMENT

These adverbs specify the region of the spectrum which is used to compute a moment. BMOMENT is the first channel of the region and EMOMENT is the last channel of the region.

BMOMENT and EMOMENT are used by

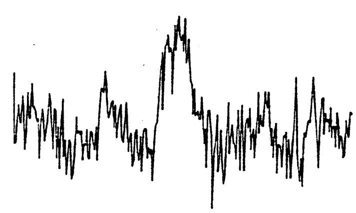
MOMENT

The initial values of BMOMENT (RCVR) and EMOMENT (RCVR) are 0.

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 A and B receivers.

EXAMPLE

Your spectrum looks like



and you want to compute a moment for this region,

so you specify

```
BMOMENT(1) = 102
EMOMENT(1) = 146
```

CENTER

User's first guess of the channel(s) at which the center(s) of up to six gaussians lie.

CENTER is used by

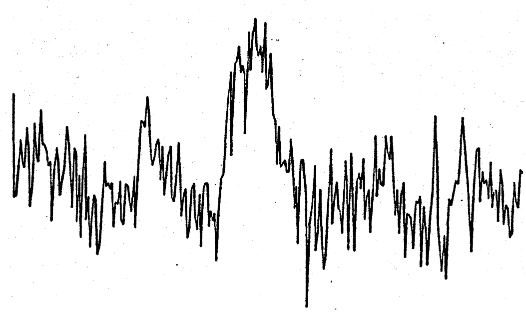
GAUSS GMODEL GPARTS RESIDUAL

The initial values of CENTER(N) are 0.

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). CENTER values must all be channels in the same receiver. The values of CENTER (usually refined by GAUSS) are used by GMODEL, GPARTS, and RESIDUAL. PEAK will find the center of a single gaussian.

EXAMPLE

Your spectra looks like



and you want to fit gaussians with centers as shown. You specify

```
CENTER(1) =  69
CENTER(2) = 123
NGAUSS    =   2
```

CLIPMAX CLIPMIN

These adverbs are the maximum and minimum temperatures (in Kelvin) that will be found in a spectrum after it is CLIPped.

CLIPMAX and CLIPMIN are used by

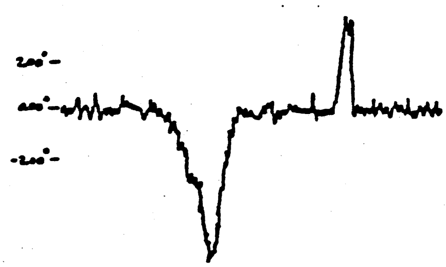
CLIP

The initial values of CLIPMAX (RCVR) and CLIPMIN (RCVR) are 99999.9 and -99999.9, respectively.

The user may specify other values of CLIP_MAX and CLIP_MIN. No verbs change the values of CLIPMAX and CLIPMIN. CLIPMAX and CLIPMIN may be specified separately for the A and B receivers.

EXAMPLE

Your spectra looks like



The upper spike is interference.
To chop it down to noise level,

specify

```
CLIPMAX = 0.5
CLIP
```

CMARK

Channels at which to draw a vertical line when a scan is displayed. Up to six different channels may be specified.

CMARK is used by

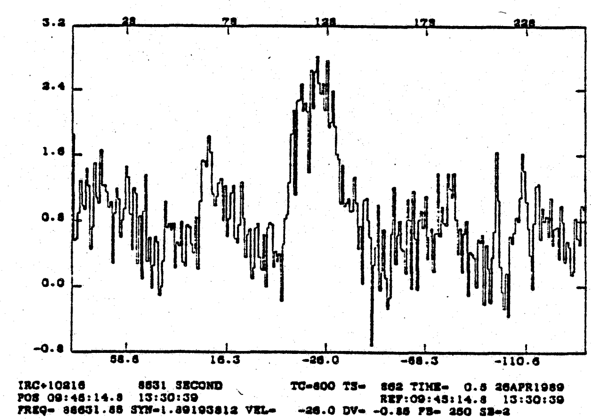
SHOW

The initial values of CMARK(N) are 0.

Which means that no vertical lines will be drawn. The user may change the values of CMARK. No verbs change CMARK.

EXAMPLE

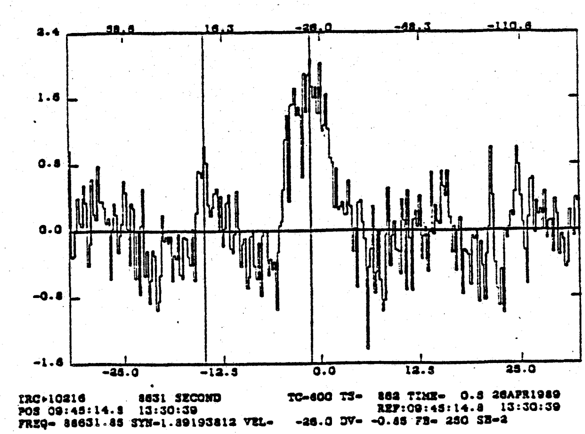
Your spectrum looks like



and you want to draw a vertical line down the center of your line, so you specify

```
CMARK(1) = 69;   CMARK(2) = 123;   SHOW
```

Your display will then look like



DF

The amount of shift between the signal and reference spectra in MHz for frequency switched data.

DF is used by

RAP

The initial value of DF is 0.

The user must specify the value of DF before using RAP or the FOLD procedure.

EXAMPLE

See RAP.

FACT

(1) A numerical constant supplied by the user, by which a spectrum may be multiplied, or

(2) A number of degrees Kelvin which may be added to the spectrum.

FACT is used by

SCALE BIAS

The initial values of FACT (RCVR) are 1.0.

FACT may be set by the user. It is not changed by any verbs. It may be specified separately for the A and B receivers.

EXAMPLE

(1) You want to multiply a spectrum by .001, so you specify

```
FACT = .001
SCALE
```

(2) You want to subtract 0.5K from a spectrum, so you specify

```
FACT = -0.5
BIAS
```

FIXC FIXHW

These adverbs are used as switches. If they are <0, then the gaussian routines will not iterate the user's guesses for the centers of the gaussians or the halfwidths of the gaussians.

FIXC and FIXHW are used by

GAUSS

The initial values of FIXC and FIXHW are 1.

For these default values, GAUSS will iterate on the values of CENTER and HWIDTH specified by the user. For values of FIXC and FIXHW less than zero, GAUSS will not iterate on the CENTER and 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 79, you specify

```
CENTER = 79
FIXC   = -1
```

(FIXC = -1 will fix all the center values if you are fitting more than one gaussian.)

FMARK

Frequencies at which to draw a vertical line when a scan is displayed. Up to six different frequencies may be specified.

FMARK is used by

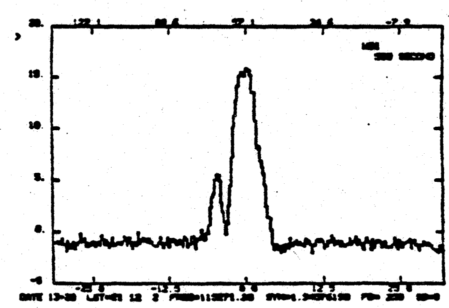
SHOW

The initial values of FMARK(N) are -9999.

Which means that no vertical lines will be drawn. The user may change the values of FMARK. No verbs change FMARK.

EXAMPLE

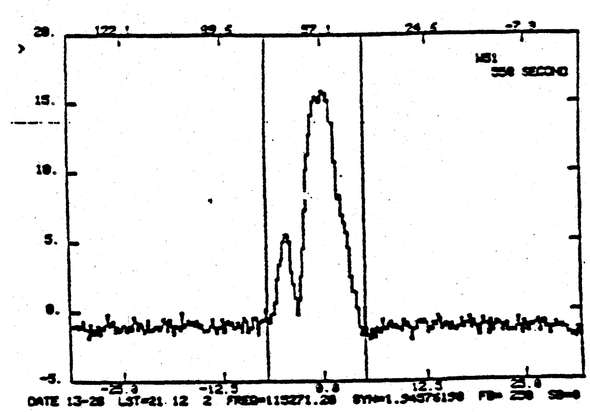
Your spectrum will look like



and you want to bracket your line, so you specify

```
FMARK(1) = -7.5;   FMARK(2) = 5.;   SHOW
```

Your display will look like



HEIGHT

The height (in Kelvins) of up to six gaussians. 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

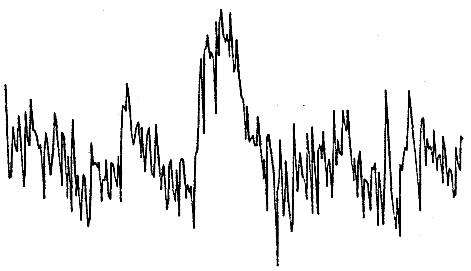
GMODEL GPARTS RESIDUAL

The initial values of HEIGHT(N) are 0.

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 the HEIGHT of a single gaussian.

EXAMPLE

To fit one gaussian, you may specify



```
CENTER = 79
HWIDTH = 12
NGAUSS =  1
```

The value of HEIGHT will be ignored. After GAUSS, the calculated value of HEIGHT may be printed by PRINT HEIGHT(1).

HWIDTH

User's first guess of the width at half the peak height (in number of channels) of up to six gaussians.

`HWIDTH` is used by

`GAUSS` `GMODEL` `GPARTS` `RESIDUAL`

The initial values of `HWIDTH(N)` are 0.

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 `HWIDTH` of a single gaussian.

EXAMPLE

Your data looks like

and you want to fit two gaussians.

For `HWIDTH` guesses, you specify

```
HWIDTH(1) =  7
HWIDTH(2) = 27
```

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 -10 E + 4

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
```

MRATIO

A ratio between x and y scaling of contour maps.

MRATIO is used by

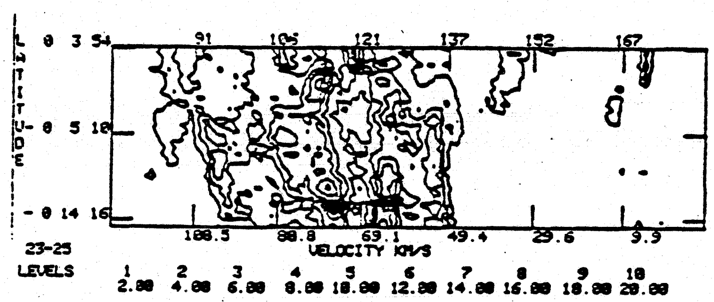
MAPSHOW   LABEL

The initial value of MRATIO is 1.

The user may change MRATIO to get a larger or smaller contour map. No verbs change MRATIO.

EXAMPLE

Your map looks like

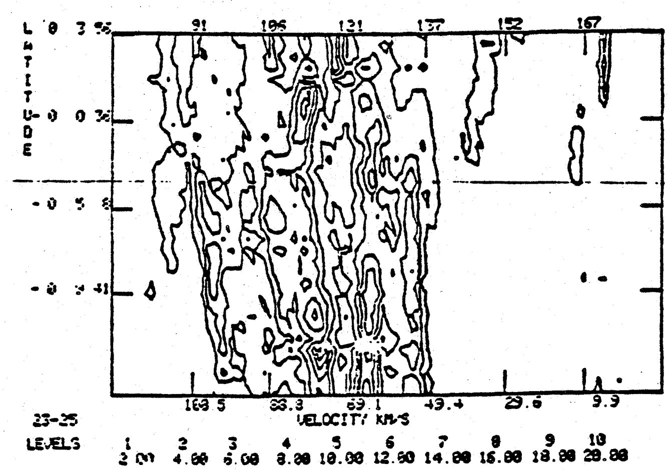


You want it to fill up the screen,

so you specify

MRATIO = 2

for your next try, and get this:



NBOX

An odd number of channels to be averaged together by BOXCAR.

NBOX is used by

BOXCAR

The initial value of NBOX is 3.

The user may change NBOX to any other positive odd integer. No verbs set NBOX and it cannot be specified separately for each receiver.

EXAMPLE

You want to average seven channels 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.

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, you 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

The initial value of NGAUSS is 1.

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, GPARTS, 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 routine will go through trying to fit a gaussian before giving up.

`NITER` is used by

    GAUSS

The user may change `NITER` if he desires. `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,

so you specify

    NITER = 12

and try again.

NMOMENT

The order of the moment to be calculated by the verb MOMENT. If MOMENT = 0, the area under the curve (between BMOMENT and EMOMENT) will be calculated. If NMOMENT = 1, the centroid will be calculated.

NMOMENT is used by

MOMENT

The initial value of NMOMENT is 1.

The user may specify any of the other values above as desired. No verbs change NMOMENT and it may not be specified separately for each receiver.

EXAMPLE

You want to calculate the area under a spectral feature, so you specify

```
BMOMENT =  80
EMOMENT = 112
NMOMENT =   0
```

NREGION

The beginning and ending channels of up to four regions for each receiver. These regions are used by following verbs.

`NREGION` is used by

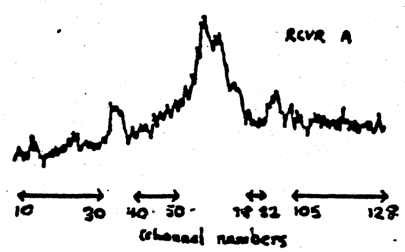
BASELINE	BMODEL	BSHAPE	DCBASE
RIPPLE	RMODEL	RSHAPE	
RMS			

The initial values of `NREGION` (RCVR, I) are 0.

`NREGION` has two dimensions: the first is the receiver and the second is a number between 1 and 8. If the second dimension is odd, the value of that element is a beginning channel of a region. If the second dimension is even, the value of that element is an ending channel of a region. `NREGION` is not set by any verbs.

EXAMPLE

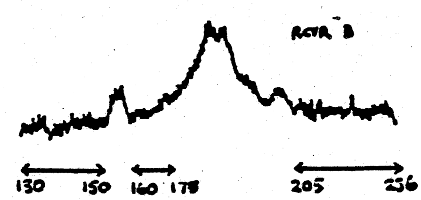
Your data looks like



and you want to use these regions to compute a baseline so you specify

```
NREGION (1,1) =   10;  NREGION (1,2)  =   30
NREGION (1,3) =   40;  NREGION (1,4)  =   50
NREGION (1,5) =   78;  NREGION (1,6)  =   82
NREGION (1,7) =  105;  NREGION (1,8)  =  128
```

but for the B receiver you have



and you want to use these regions, so you specify

```
NREGION (2,1) = 130;  NREGION (2,2) =  150
NREGION (2,3) = 160;  NREGION (2,4) =  175
NREGION (2,5) = 205;  NREGION (2,6) =  256
NREGION (2,7) =   0.
```

NOTES: Set the beginning channel of the first region and you don't want to use equal to zero. `NREGION` will supercede `BBASE` and `EBASE` (for a particular receiver) unless `NREGION` (RCVR, I) = 0.

NSAVE

The disk bin into which a scan is to be loaded or from which it is to be retrieved.

NSAVE is used by

```
SAVE      RECALL
```

The initial value of NSAVE is 0.

The user must select a save bin between 1 and 154 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.

So you specify

```
NSAVE = 34
SAVE
```

NSIGMA

An adverb that specifies the absolute maximum value to be used as the criteria for determining bad channels.

NSIGMA is used by

SPIKE

The initial value of NSIGMA is 3.

EXAMPLE

See BADPT

RFREQ

User's first guess to frequency of sinusoidal baseline (in Megahertz).

RFREQ is used by

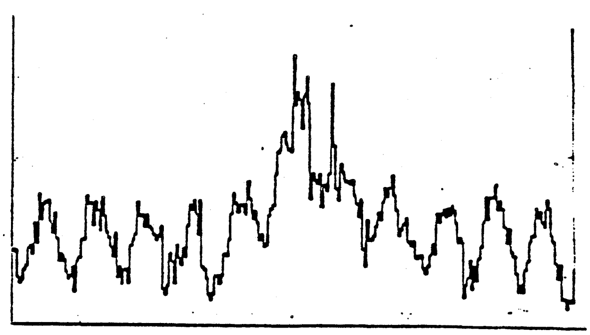
RIPPLE RMODEL RSHAPE

The initial value of RFREQ is 23.

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 RFREQ. If it can't do it, it will give the message "FIT FAILED". RFREQ cannot be specified separately for each receiver.

EXAMPLE

Your data looks like



and you want to remove a sinusoidal baseline so you guess its frequency, and specify

```
REFREQ = 23
```

RSHIFT

The number of channels by which receiver B is shifted when added to receiver A.

RSHIFT is used by

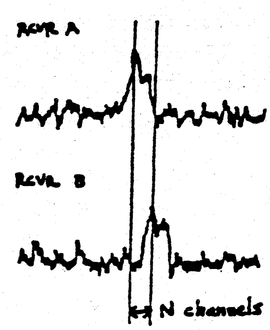
    SLIDE

The initial value of RSHIFT is 0.

The user may change RSHIFT as needed. RSHIFT is not changed by any verbs.

EXAMPLE

Your data looks like



You specify

```
RSHIFT = -N
SLIDE
```

If the peak in receiver B had been to the left of the peak in receiver A by N channels, you would have specified

```
RSHIFT = N
SLIDE
```

SHIFT

The amount of shift between scans shown in profile form by the SHOW1 and SHOW2 procedures.

SHIFT is used by

SHOW1   SHOW2

The initial value of SHIFT is 1.

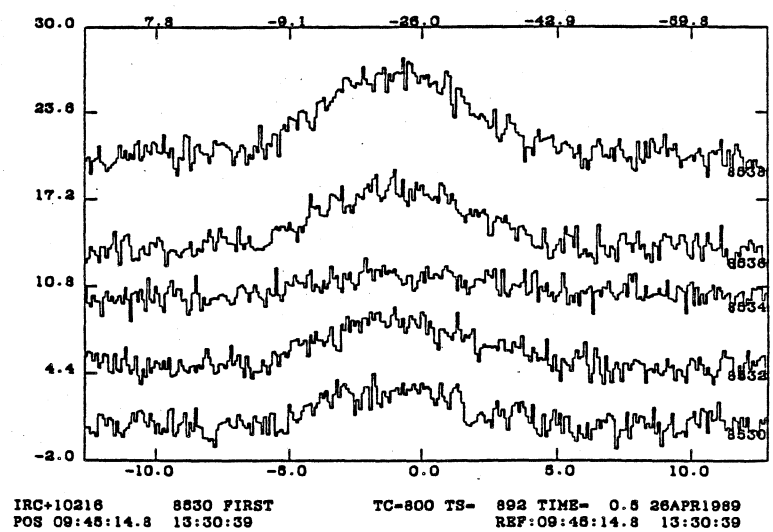
The user must specify the value of SHIFT before using the SHOW1 or SHOW2 procedures.

EXAMPLE

You want to make a profile of the scans you wish to map. You specify

```
8530 8538 ADD
-2 30 RANGE
SHIFT = 5
SHOW1
```

and you will get



SLABEL

The value of SLABEL determines the placement of the scan number and source name on the SHOW display. If SLABEL = 0, the information is written in the upper right corner of the SHOW display. If SLABEL = 1, the information is printed at the bottom of the SHOW display. SLABEL is set to 2 for profile displays where each scan is labeled with its scan number at the rightmost edge of its spectrum.

SLABEL is used by

SHOW

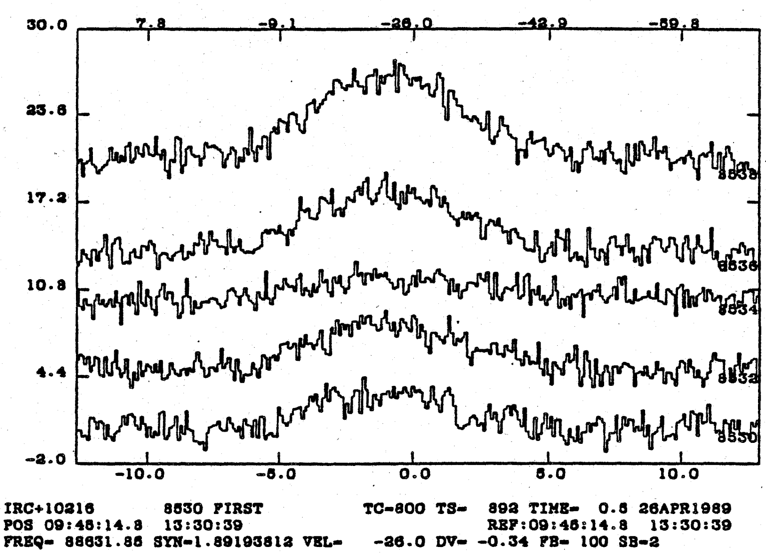
The initial value of SLABEL is 0.

The C1, C2, CB, A1, A2 and AB procedures set SLABEL = 1 so that when the STACK is printed at the top of the screen, it does not overprint the scan number and source name. These procedures then reset SLABEL to the default value. The SHOW1 and SHOW2 procedures set SLABEL = 2 so that each scan will be labeled with its scan number. SHOW1 and SHOW2 then reset SLABEL to the default.

EXAMPLE

A typical SHOW2 display with SLABEL = 2 is shown below.

```
8530 8538 ADD
-2 30 RANGE
SHIFT = 5
SHOW1
```



SMWGT

An array to describe a weighting function for smoothing data. The first element reflects the number of points in the weighting function (not greater than 11).

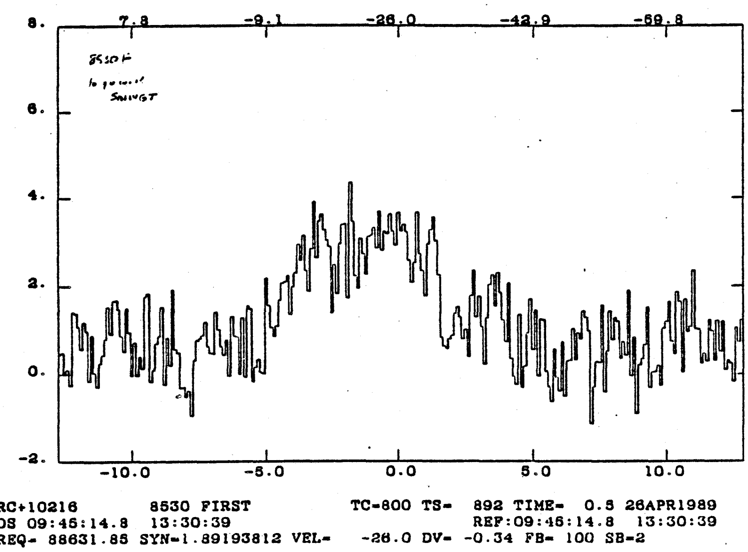
SMWGT is used by

SMOOTH

The initial values of SMWGT are 0.

EXAMPLE

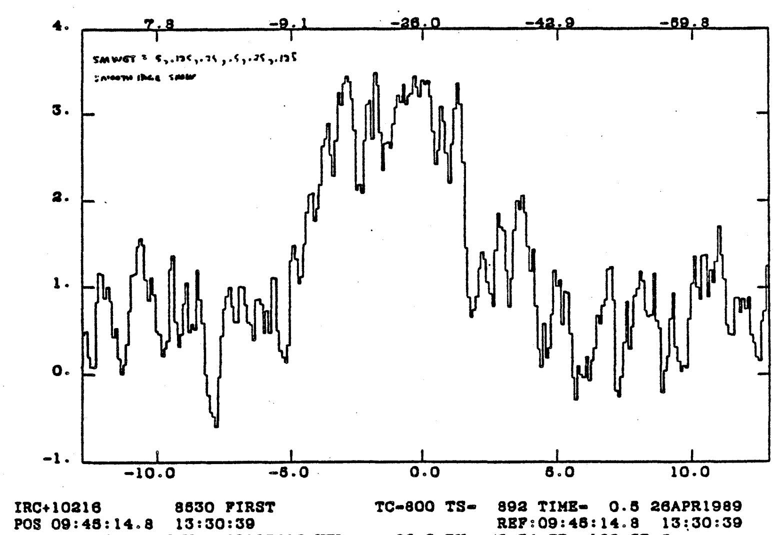
Your data looks like



so you specify

```
SMWGT = 5,  .125,  .25,  .5,  .25,  .125
SMOOTH PAGE SHOW
```

Your data then looks like



TMARK

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

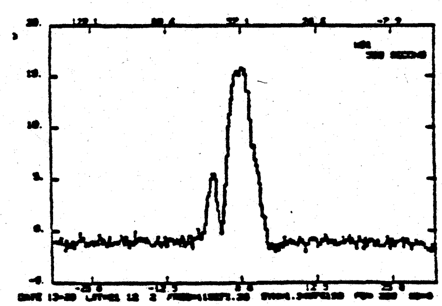
    SHOW

The initial values of TMARK(N) are -9999.

Which means that no horizontal lines will be drawn. The user may change the values of TMARK. No verbs change TMARK.

EXAMPLE

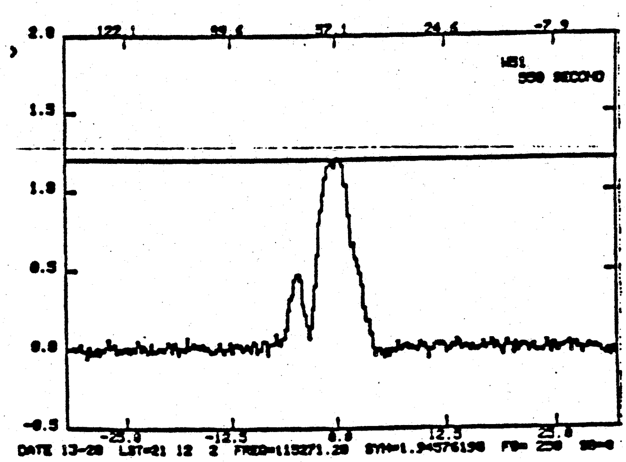
Your spectrum looks like



and you want to verify the line temperature, so you specify

    TMARK(1) = 1.2;   SHOW

Your display looks like



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

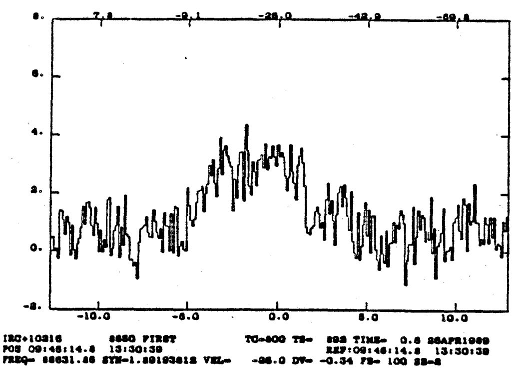
The initial values of VMARK(N) are -9999.

Which means no vertical lines will be drawn.

The user may change the values of VMARK. No verbs change VMARK.

EXAMPLE

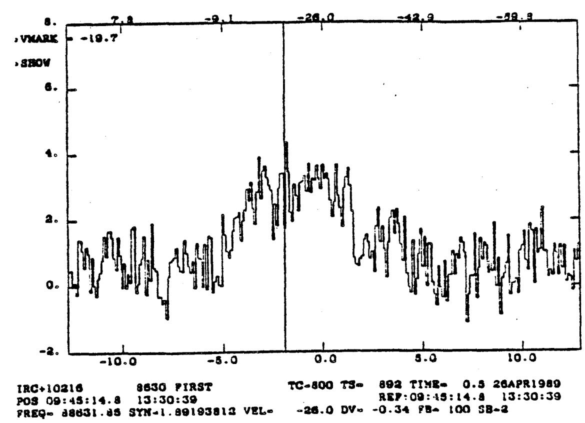
Your spectrum looks like



and you want to draw a vertical line down the center of your line, so you specify

```
VMARK = -19.7
SHOW
```

Note this sets VMARK(1) through VMARK(6) to -19.7. If you now change VMARK(1) to -9, the next time you SHOW, you will get lines at -9 and -19.7.



WEIGHT

The weighting that is applied to scans when stacked by `SUM`.

`WEIGHT` is used by

`SUM`

The initial value of `WEIGHT` is 1.

The user must set the desired value of `WEIGHT` before using `SUM`. If left unchanged, the data will be weighted by the integration time.

EXAMPLE

See `SUM`.

YINCR YMIN

Are used to set the Y scaling of the spectrum graphs. YINCR equals the number of Kelvins that will appear between each Y tick mark. YMIN equals the lowest number of Kelvins that will appear on the graph.

YINCR and YMIN are used by

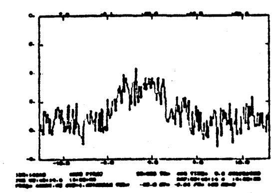
SHOW RESHOW

The initial values of YINCR (RCVR) are 0.
The initial values of YMIN (RCVR) are -99999.9.

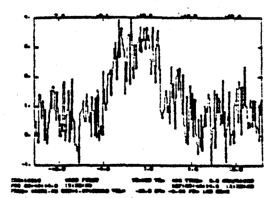
If YMIN is less than or equal to -9999, SHOW and RESHOW will set the Y scaling automatically. If YMIN is set to greater than -9999 by the user, SHOW and RESHOW will use the values of YMIN and YINCR specified by the user. So if the user sets YMIN he should also set YINCR. Four verbs set the value of YMIN. FREEY and AUTO set YMIN to greater than -9999, so that SHOW and RESHOW will set the Y scaling. HOLDY sets YMIN to a very negative number (-9E10) so that SHOW will not update its Y scaling. Both YINCR and YMIN may be specified separately for each receiver.

EXAMPLES

(1) SHOW gives you this graph



To expand the scale to get this

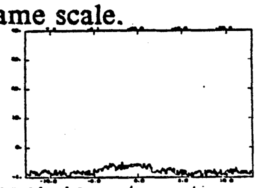
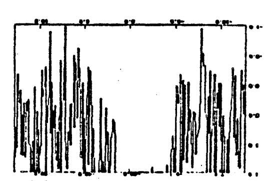


specify

```
YMIN = -1
YINCR = 1.00
```

(2) You want to compare two different spectra but you are frustrated because SHOW plots them to different scales. To get the second graph to the same scale you specify HOLDY, before you specify SHOW.

(3) You followed the directions in example (2) and now everything is coming out to the same scale.

You get  or this 

To get back to automatic scaling, specify FREEY or AUTO.

ZLINE

This adverb is used as a switch. If Z_LINE = 1, a horizontal line is drawn at zero Kelvins. If Z_LINE = 0, the line is not drawn.

ZLINE is used by

SHOW

The initial value of Z_LINE is 0.

The user may change ZLINE as desired. No verbs change ZLINE. It is not specified separately for each receiver.

EXAMPLE

If ZLINE = 1, SHOW will draw the zero line as shown: