

NATIONAL RADIO ASTRONOMY OBSERVATORY
Post Office Box "O"
Socorro, New Mexico 87801

VLA Computer Memorandum No. 133

DESCRIPTION OF THE OBSERVATION-REQUEST-EDITING PROGRAM OBSERV

A USER'S MANUAL

Ernst Raimond
September 1976

I. INTRODUCTION

The program described in this memorandum has been designed to facilitate the preparation of observation request cards. The format of these cards has been fully described by Barry Clark in VLA Computer Memo 131, and can be summarized as follows:

1. Observations are defined by the parameters provided on the observation (source) request cards and by a set of general and default parameters residing in the files: ARRAY, SUB1 through SUB5, ANTENNAS, and in the IF-parameter and subreflector files. All these files are catalogued in the directoried partition DVP.
2. For the parameters not given in these files the synchronous program assumes default values (e.g., cable wrap, planetary motion, etc.).
3. It is possible to override the default values by specifying some or all of the parameters on the source request card or on cards following it: LO for LO-frequencies, etc., AN for cable wrap, LI for line observation specification, and PM for planetary motion. These cards are identified by "/" in columns 1 and 2 and "LO", "AN", "LI", or "PM" in columns 3 and 4. The values given are valid for one observation only.

The procedure OBSERV allows the user to make up an observation request list in the required format by entering, editing and modifying the parameters in free format and in the order he specifies. It also has capabilities of

looking up sources in a large master source list and in a smaller table of sources that he enters himself. It checks many of the parameters for validity as they are entered, and it is able to check the feasibility and the consistency of a completed observation request list.

Finally it will catalog the list under a name specified by the user.

The user's manual consists of three parts:

1. A few general remarks about entering the program and about some general rules (section 2).
2. A description of the commands and their formats (section 3 and table 1)
3. A table of the identifiers used, their data formats, and some remarks with regard to a few of them (section 4 and table 2).

Some notes for programmers concerning the structure of the procedure OBSERV and the embedded program OBS are given in an appendix.

2. THE PROCEDURE OBSERV

In order to start executing the observation request editing program (procedure OBSERV) one must be at the Job-Control level of operating the background of one of the ModComp computers. It is a good habit to initialize things by typing on the terminal.

\$JOB, and, if no error messages appear, one proceeds with:

\$DO OBSERV, name, or \$OBSERV, name; where "name" is an arbitrary name (≤ 8 characters) that will ultimately be used to catalogue the observation request list.

This starts the OBSERV procedure. To have to give the cataloging name at this stage rather than in the ./CAT command itself (cf section 3.6) is a slight inconvenience but it was sufficiently attractive from a programming point of view to justify it. It allowed us to use the facilities of the ModComp Source Editor (SEDIT) for this particular function. If the user forgets to give the name in the call, the request list will be catalogued under a default name (STUPID) unless he corrects this omission in time (cf. /EXIT command, section 3.6).

The procedure OBSERV will automatically call a program OBS which does most of the work. In this manual this program will hardly be mentioned any more since its use is entirely transparent to the user. He only has to start the procedure OBSERV.

Once the procedure OBSERV has been started it expects commands and values of parameters from the user. It is not a conversational program, although it will, of course, complain when the user tries to do things it does not like.

Normally it is assumed that the user enters commands and data by means of a terminal. However, for special cases, this can be changed by assigning AI to the device one wants to read from (a card reader for instance) before starting OBSERV (e.g.: \$ASS AI=CR assigns AI to the card reader).

While the user is working on an observation request list it is compiled on a disc file called BO (binary output). When he is done with it (even temporarily) he must catalogue it either in the disc-partition that contains all the observational parameters (DVP) or in one of the other directoried partitions, (DSV, DSC, DSH, DSR or DSS). A partly complete list should never be left on BO since all compilers, editors and many other programs use BO as well.

Throughout the execution of the editing program an internal pointer indicates the current card. In general the current card is the last one entered. When this is not the case it is specifically mentioned in the manual. These are special commands to move the pointer around to other cards (cf section 3.5).

The general philosophy of OBSERV is to set as many parameters as possible to a fixed default value at the beginning of the program and to set up convenient input and/or output orders, etc., so that the actual compilation of the observation request list can be done with a minimum of effort and in a way adapted to the requirements of the user rather than those of the synchronous computer program. The commands to set things up initially are described in section 3.1.

Some default values are loaded with the program (e.g., default DECLINATION = -90°) so that observation requests for which the coordinates are not specified will be caught by the checking procedure.

The way to force OBSERV to use default values for parameters one would normally enter manually is to leave the value blank. This, of course, does not always work when setting the defaults themselves. In that case blanks will be stored as the default value of the parameter if that is an allowed option for the particular parameter.

3. THE COMMANDS

All commands start with "./,". The first three characters thereafter determine the function. Any additional characters will be ignored. In the command formats the following convention is adhered to:

Upper case symbols must be typed literally (at least the underlined part) while lower case symbols must be replaced by actual quantities or names. Commas and equal signs may be replaced by spaces or other delimiters (,/=). Optional parameters have been put between square brackets.

The commands are listed below in the order one is likely to use them when setting up an observation request list. They have been grouped in six sections:

- 3.1. Initialization.
- 3.2. Entering new observation request cards.
- 3.3. Displaying the observation request list.
- 3.4 Making modifications.
- 3.5 Utility commands.
- 3.6 Getting the list ready for observing.

Table 1 contains an alphabetically ordered summary of the commands.

3.1 INITIALIZATION

./ID initials, number

When entering OBSERV this command must be used first. It writes a card at the current location in the BO-file (at location 1 when used at the beginning of the program) containing the observer's initials and a positive number (<32768) identifying the observing program. The first two characters of this card will be /.

./SET identifier = value [identifier = values,]

This assigns default values to the identifiers mentioned. The number of identifier-value pairs is limited by the length of the line only. When the user tries to assign an invalid value to an identifier an error message is issued. Only the incorrect parameter needs to be reset. Some default values are loaded with the program (e.g., DEC=-90°) but these can be overwritten with the ./SET command.

./INFORMAT [identifier [, identifier [, identifier.....]]]

If used with parameters this command determines the order in which identifiers are going to be entered by means of the ./ADD command. The default input order (loaded with the program OBSERV) is: NAME, STOPTIME, RA, DEC, EPOCH, BANDS (L, C, U or K), OBSERVING MODE, GAIN FACTOR, BANDWIDTH CODES; i.e., the regular order of the observation request card without the feature velocity.

After entering the input order OBSERV produces the default values of all parameters not mentioned in the ./INF command. The command may also be used without parameters to perform only the function just mentioned: provide information on the default values. In this case the input order is left unchanged.

The user should be aware of the fact that the location of NAME with respect to, for instance, RA and DEC has some influence on the way the data from master source list and the user's source table are copied onto the request cards (cf. section 4.1).

./OUTFORMAT identifier [, identifier [, ...:]]

This command establishes the order in which parameters will be listed with the ./LIST command. It is limited to observation request parameters. The cards //LO, //AM, //LI, //PM will always be listed as card images. Consequently identifiers belonging to those cards will be ignored when ./LIST processes the outputformat.

./USERSSOURCETABLE [,device
 ,partition]

By means of this command the user can enter a table of sources into the program. This table will be scanned whenever a source name is entered by means of other commands (./ADD, ./CHANGE, ./MODIFY or ./SET). The coordinates and the gainfactor will then automatically be copied from the sourcelist.

The user's source table is normally entered from the terminal (SBI), which is the default if no parameter is given in the command. In this case the table must be terminated by typing an End-of-File (\$\$ in columns 1, 2). The parameter, if specified, will cause the card images to be read from the partition or device, the table being terminated by an EOF. The order of the parameters is fixed and is always prompted on the terminal:

NAME, RA, DEC, GAIN FACTORS for L, C, U, K BANDS.

The parameters are read in free format.

The number of sources in the user's source table is determined by the size of the partition in which the program OBS is running, since the table is stored in the space remaining above the program. At present, with the smallest position in general use, this will allow about 400 sources. If this limit is exceeded the message "NO MORE SPACE AVAILABLE, END OF TABLE".

./RECOVER [filename [, partition]]

a) Without parameters:

This version of the command must be used when the program OBSERV is entered for a second time while the observation request

list produced the previous time is still on the file B0. ./REC will then just read this file and adjust an internal variable that indicates the length of the list and it will leave the pointer at the last card of the existing list. In addition the command ./REC restores the default parameters used the previous time, the user's source table, and the calibrator list.

b) With parameters:

In this case the entire file with the specified name will be copied from the specified partition (or from partition DVP if not specified) into the observation request list. Copying starts the current card. If after the current card was not the last one the tail end of the list will be restored after the copied file. The pointer is left at the last card of the copied file.

./CALIBRATORLIST name1 [,name [...]]

This command is used to define the list of sources which are to be used as calibrators during the current observing request list. When the name of a new source is typed in under the ./ADD command, this list is scanned and the calibrator indicator is automatically inserted if the source is present. If name1 is "LIST", then the calibrator list will be listed on the terminal. If name1 is "DELETE", then the following names will be deleted from the calibrator list.

3.2 ENTERING NEW OBSERVATION REQUEST CARDS, ETC.

./ADD [,LO [,AM [,LI [,PM]]]]

a) Without parameters

This is the command by which new observation request cards are entered normally. When the user types ./ADD cards are added to the observation request list after the current card and the remainder of a possibly already existing file is saved (this may

take a few seconds). The input format is prompted on the screen of the terminal to remind the user of what he specified. After this the user is expected to enter values of parameters in the order of the input format. An observation request card and (if the appropriate indentifiers occur in the input format) one or more of the special cards (//LO, etc.) will be written onto BO following the current card for every line entered on the terminal. The last card entered becomes the current card. A blank value (or a value not given by typing a comma instead or by omitting it at the end of a line) indicates that the default value of the identifier is to be used.

If the user enters an erroneous value for one of the parameters an error message is issued and the entire line must be retyped. The command is terminated by any other command (anything starting with ./) as long as it is typed at the beginning of a line. Before this new command is allowed to be executed the part of the observation request file that was saved at the beginning of the ./ADD command is restored after the current card. However, the last card "added" remains the current card.

b) With parameter(s)

With this command it is possible to insert an incidental //LO-card or any of the other special cards into an observation request list. The parameter(s) specifies the card(s). At the end of the parameter list the program returns to the ./ADD-without-parameters mode and prompts the input format, etc.

./DUPLICATE [,n[,m]]

By means of this command cards n through m are duplicated after the current card. The cards already present after the current one are "pushed down" to after the duplicated ones. As a result of this the command may take a few seconds.

The parameter m defaults to n if not given. The parameter n defaults to the current card number. Therefore: ./DUP (without

parameters) duplicates the current card, `./DUP n`, just duplicates card `n` after the current one and `./DUP,,m` duplicates the current card through card `m`.

Caution: If any of the cards to be duplicated occupy places that will be occupied by the duplicated cards (e.g., if `n = current + 1`) the resulting file will be different from what is expected since the duplicated cards are gradually replacing the cards to be duplicated. One may end up with a set of identical cards in this case.

The command `./LINE` (cf section 3.5) may be used to find current line number.

./DELETE [`,n[,m]`]

Deletes cards `n` through `m` from the observation request list. `m` defaults to `n` if not given and `m` defaults to the current card number, in exactly the same way as illustrated under `./DUPLICATE`.

3.3 DISPLAYING THE OBSERVATION REQUEST LIST

./PRINT [`,n`]

Prints `n` cards starting with the current card on the terminal. `n` defaults to 1 when not given all cards are printed as card images.

./LIST [`,DEVICE`]

Lists the entire observation request file using the specified outputformat to determine the order of parameters on the line. Cards starting with `/.` (the ID card) and with `//` (LO, AM, LI and PM cards) are printed as card images. If the parameter device is present then the listing will appear on that device (e.g., TYC will produce a listing on the lineprinter).

3.4 MAKING MODIFICATIONS IN AN ALREADY EXISTING LIST

./MODIFY [`,identifier=value[,identifier=value.....]`]

To modify one or more parameters to the specified values on the current card only. After execution of the command the modified line

is printed on the terminal. Upon entry of erroneous values for any of the parameters an error message will be issued. Only the incorrect parameter needs to be remodified.

./CHANGE [,identifier[,identifier....]][,n]

This command allows modifications of the specified identifiers on a number of subsequent cards.

The identifiers mentioned in a CHANGE command must all belong to the same card type (all observation request, all LO, all AM, etc.). If they are not, an error message is issued. Cards of other types will automatically be skipped in the change process.

a) Without the numerical parameter n

In this case the sequence of events is:

1. the unmodified current card is printed on the terminal,
2. the user enters the values he wants to modify the identifiers to for this card (in the order given in the command).
3. the modified card is printed,
4. the pointer advances one so that the next card becomes the current one and cycle 1 through 4 starts over again.

b) With the numerical parameter n

In this case n indicates the number of cards to which the modification of the specified identifiers will be applied. All card types are counted, so that in a "mixed" observation request list fewer cards may actually be modified than is indicated by n.

The sequence of events is in this case:

1. the unmodified current card is printed,
2. the user enters the values into which the identifiers on the current and the next (n-1) cards must be changed,
3. n (or less) modified cards are printed.

3.5 UTILITY COMMANDS

These commands move the pointer from one card to another, assign files, write end-of-file marks, etc.

./LINE

This command returns the sequence number of the current card (line) on the terminal screen. This is especially useful for duplicating or deleting.

./GOTO n

Makes card n the current card (sets the pointer to card n). If n is outside the existing observation request list the message "END OF QUEUE" will be printed and the pointer will be left at the last existing card.

./BACKWARD n

Decrements the pointer by n so that the current card will be n cards before the previous current card. The pointer never becomes < 1.

./FORWARD n

Increments the pointer by n unless if this would advance the pointer beyond the last card, the "END OF QUEUE" message is printed and the last existing card becomes the current one.

./REWIND

Sets the pointer to 1 so that the 1st card becomes the current one.

./WEOF

Writes an EOF on the observation request list after the present card. This has the property of deleting all subsequent cards. It is not necessary to do this before exiting from the program since this is done automatically by all commands that exit (cf section 4.6).

4.6 GETTING THE REQUEST LIST READY FOR OBSERVING

./START [,begintime]

If durations have been used in making up the observation request list, this command will replace all of them by absolute stoptimes by adding the duration to the stoptime of the preceding observation or to the specified begin time for the first observation of the list. If no begintime is specified 00h 00m 00s LST is assumed by default.

It is not required to perform this operation on a request list since the synchronous program will accept durations.

./CHECK [,device]

After writing an end-of-file mark on BO after the last obs-request card and saving all default values, the user's source table, and the calibrator list in a second file of BO, this command terminates OBS and calls a separate program OBSCHECK (load module OCH) written in FORTRAN that checks the consistency and the feasibility of the entire observation request list. It prints the list on the specified device [or on the one assigned to file 6 (normally the line printer) if no device is given] with messages describing the errors found in the card printed just above the messages. This program at present checks for sequence errors, elevation limit errors and some other obvious errors, but many more checks can easily be built in as experience grows.

At the end of the checking operation the main OBSERV program is re-entered and a ./RECOVER command is automatically executed (not shown on the screen) so that any errors found by the ./CHECK command may be corrected right away. ./REC recovers all information that was saved in the first two files of BO just before entering OBSCHECK.

./CATALOGUE [,partition]

Once the user is satisfied with his observation request list or when he wants to save a partly finished list for completion at a later time he must use the ./CAT command to get it stored in such a way that the synchronous program (or OBSERV) can use it. The list will be catalogued in the disc partition specified, or in DVP (the observation file) if no

partition is specified. In any case the partition must be a directoried one (DSV, DSC, DSH, DSR, DSS or DVP). The list will be catalogued under the name given as a parameter in the call of the program OBSERV (\$OBSERV, name). If no name was given in the call, the name "STUPID" will be used. In order to get back to the same state at a later instance one must use the ./REC, name, partition command after entering OBSERV again.

The cataloguing operation is performed through the ModComp SOURCE EDITOR (SEdit) after termination of OBS in the fashion described under ./CHECK. However, this switch of programs is of no concern to the user.

./EXIT

This command simply terminates OBSERV having previously written an EOF on BO and saved the default values, the user's source table, and the calibrator list as in ./CHECK and ./CAT. Therefore it is possible to get back to the state in which one was before exiting as long as BO is not disturbed by another program. It is obvious that in general it is safer to exit through ./CAT if one expects to have to operate on this observation request list again. However, ./EXIT is useful at least in the case the user forgot to provide a catalog name in the original call of OBSERV. The way to correct that is by typing:

```
./EXIT
$OBSERV, name (catalog name)
./ID, observer, number
./REC
```

Now we can execute ./CAT to catalog the list under the name just given.

4. IDENTIFIERS

All identifiers (observing parameters) are defined by the first three characters for parameters belonging to the observing request cards themselves and by three characters preceded by the appropriate prefix (LO, AN, LI or PM) for the parameters belonging to the special cardtypes //LO, //AN, //LI and //PM respectively.

Table 2 lists all identifiers together with the format of their values and the values permitted by OBSERV upon entry by the user. Sometimes a more stringent check is made by the ./CHECK command. Most identifiers need no further explanation. However, a few, the source name and the stoptime, will be given some extra attention in sections 4.1 and 4.2.

Blanks (spaces) often have a special meaning when entering values. Even for those parameters for which blanks are a permitted value, blanks are used to indicate the user's wish to enter default values at least in the ./ADD and the ./CHANGE commands. In the ./SET and ./MODIFY commands blanks are accepted as values if they are permitted. If they are not they result in an error message. In some cases where it may be desirable to be able to enter blanks as a value even in ./ADD or ./CHA a question mark (?) may be used to indicate a character to be replaced by a space.

Except in the cases where the values are character strings with a special significance to their positions rather than numeric values free formats may be used everywhere, even to the extent that minutes and seconds need not be typed in times and coordinates. Of course, if a time or a coordinate occurs in the middle of a string of values the place of the minutes and seconds must be indicated by commas (e.g., 23,,= 23 00 00) unless the next value is a non-numeric parameter. For the sake of brevity formats are indicated by their FORTRAN notation although the limitations of the FORTRAN formatter apply nowhere (e.g., in the OBSERV program "9" is the same as "009" in an I3-format).

4.1 NAME, RA, DEC and GAIN and CALIBRATOR INDICATION

Whenever a sourcename is entered the program scans two source tables in an attempt to find the coordinates of the source and the GAIN code. First a large master source list is scanned. If an identical name is found in that list the right ascension and declination are copied onto the current observation request card as well as a gaincode. The latter is one of four codes (one for each frequency band) selected on the basis of the current BAND indication. This may be the one just entered or an ./ADD command if BAND precedes NAME in the command sentence or otherwise it is the default BAND indicator. After the standard source list the much shorter user's source table is scanned. From this list RA, DEC, and the appropriate GAIN code given in that list are copied onto the current observation request card.

From this description it follows that, if the source name occurs both in the master source list and in the user's source table, the latter one has priority.

If NAME, RA, DEC, and GAIN appear in the input format there is a subtle difference in the treatment of the source tables depending on where NAME appears with respect to the other identifiers:

If NAME occurs in front all or some of the other relevant identifiers the user must leave the values of the later occurring identifiers blank if he wants to use the data in the source tables. If he types values for those identifiers anyhow these values will be the ones that are written on the request card. On the other hand, if he leaves values blank for sources that do not occur in any of the tables, the default values will be used. For this reason a default declination of -90° is loaded with the program. The ./CHECK operation will catch a mistake of this type in this case.

If, however, NAME occurs behind all or some of the other relevant identifiers the data from the source tables will always have priority over whatever is typed by the user for the earlier occurring identifiers.

Finally, one should realize that the coupling of several identifiers to NAME does not only occur in the ./ADD command but also in the ./MOD and ./CHANGE commands. (Also in ./SET but that is hardly relevant.) Whenever the sourcename is modified into one that occurs in one of the source tables, the coordinates and the gaincode and sometimes also the calibrator indicator get changed at the same time.

4.2 STOPTIMES AND DURATIONS

Normally the stoptime of an observation is indicated by an absolute local sidereal time (LST). However, the synchronous program also accepts durations, indicated by a \$-prefix in front of the time. The OBSERV program provides both options: one can enter a stoptime (without prefix) or a duration being a stoptime with a prefix. In addition there are some other possibilities of producing durations.

1. By specifying a DURATION rather than a STOPTIME, in which case the program rather than the user provides the \$-prefix.
2. By typing an "O" or a "C" instead of either a stoptime or a duration. In these cases the default normal Observations duration or the default Calibrator observation duration will be used. These default durations may be set by means of the command:

```
./SET ODUR=value, CDUR=value
```

If things are properly prepared one is able to enter stoptimes and durations five different ways:

1. by typing a stoptime,
2. by typing a duration,
3. by leaving the stoptime blank, resulting in the use of the general default duration (or stoptime),
4. by typing an O so that the default ODURATION will be used, and finally,
5. by typing a C so that the default CDURATION will be used.

Table 1

ALPHABETICAL LIST OF COMMANDS

COMMAND AND FORMAT	DEFAULT	SECTION	FUNCTION
./ADD[,LO[,AN[,LI[,PM]]]]		3.2	<p>a. adds observation request cards under the current one (without parameters)</p> <p>b. adds //LO, //AN, //LI and/or //PM-cards under the current one [with the appropriate parameter(s)]</p>
./BAC[,n]	n=1	3.5	moves the pointer n places backwards in the list (does not become ≤ 1)
./CAT[,partition]	part.=DVP	3.6	catalogs the observation request list in the specified disc partition after exiting in the same way as through ./EXIT.
./CHA,identifier[,identifier,...][,n]		3.4	<p>a. changes specified parameters to values entered line by line until command is entered (without numeric parameter)</p> <p>b. changes specified parameters on n cards to values entered on first line (with numeric parameter)</p>
./CHE[,device]	device= line printer	3.6	checks consistency and feasibility of observation request list. Prints list and errors on specified device.
./DEL[,n[,m]]	m=n n=current	3.2	deletes card n through m from list.
./DUP[,n[,m]]	m=n n=current	3.2	duplicates cards n through m under the current card.

Table 1 (continued)

COMMAND AND FORMAT	DEFAULT	SECTION	FUNCTION
./EXI		3.6	writes EOF at end of observation request list, saves default values of parameters and the user's source table and terminates the program.
./FOR[,n]	n=1	3.5	moves pointer n places forward in the list (not beyond end of list)
./GOTO n		3.5	moves pointer to card n
./ID,initials,programnumber		3.1	produces card with observer's initials and his programnumber and /. in columns 1 and 2. Sets switch to accept other commands hereafter.
./INF[,identifier[,identifier...]]	preceding input format or standard obs-requ format	3.1	<ol style="list-style-type: none"> 1. specifies order in which values of parameters will be entered by means of ./ADD command. 2. lists all the default values of the parameters not mentioned in ./INF....
./LIN		3.5	prompts the number of the current card (or line).
./LIST[,device]	device=terminal	3.3	lists obs requ cards in the order specified by ./OUT and the other cards as card images.

COMMAND AND FORMAT	DEFAULT	SECTION	FUNCTION
./MOD identifier=value[,ident.=value..]		3.4	modifies values of specified parameters on current card only.
./OUT identifier[,identifier...]	standard obs.requ. format	3.1	specifies order in which parameters will be listed by ./LIST command (only those on obs requ cards).
./PRI[,n]	n=1	3.3	prints the n cards starting with the current one. Leaves pointer at current line.
./REC[,filename[,partition]]	part=DVP	3.1	a. recovers administration, default values, user's source table and calibrator list after a second entry of OBSERV (without parameters). b. inserts the specified file from the specified partition after the current card (with parameters).
./SET identifier=value[,ident.=value...]		3.1	sets default values of parameters.
./START[,time]	time= 00 ^h 00 ^m 00 ^s	3.6	replaces all observation durations by stoptimes, specified time is the start time of the first observation.
./REW		3.5	sets pointer at card 1 of obs requ list.

Table 1 (continued)

COMMAND AND FORMAT	DEFAULT	SECTION	FUNCTION
./USE[,device]	device= terminal	3.1	initializes the entering of a special list of often used sources.
./WEOF		3.5	writes EOF after current cards and termates deletes all subsequent cards.
./CAL,name1 [,name[,...]]		3.1	builds a calibrator list: <ul style="list-style-type: none"> a. if name1 is LIST, lists the table b. if name1 is DEL, deletes all subsequent names from list c. Else, adds all subsequent names to list.

IDENTIFIERS GROUPED BY CARD TYPE

IDENTIFIER	UNITS	FORMAT	PARAMETER	PERMITTED VALUES
<u>OBSERVATION REQUEST CARD</u>				
<u>NAME</u>		A8	source name (cf section 4.1)	any character string
<u>QUALIFIER</u>		I5	source qualifier	$0 \leq \text{QUA} < 32768$
<u>STOPTIME</u>	h,m,s	[\$]I2[1X,I2 [,1X, I2]]	stoptime (local sidereal time) with \$-prefix: duration	< 24 h or < 24 h
<u>DURATION</u>	h,m,s	I2[,1X,I2 [,1X, I2]]	duration of observation	< 24 h
<u>ODURATION</u>	h,m,s	I2[1X, I2 [,1X,I2]]	default duration of standard observation (can only be used with ./SET command)	< 24 h
<u>CDURATION</u>	h,m,s	I2[,1X,I2 [,1X,I2]]	default duration of calibrator observation (can only be used with ./SET command)	< 24 h
<u>RA</u>	h,m,s	I2,[1X,I2 [,1X,F7.4]]	right ascension of field center	< 24 h
<u>DECLINATION</u>	o ' "	I3,[1X,I2 [1X,F6.3]]	declination of field center	$-90^\circ < \text{DEC} < 90^\circ$
<u>EPOCH</u>	char[,yrs.]	A1, I4	epoch of RA and DEC	blank or ?: equinox of 1950.0 C: equinox of 2000.0 D: apparent position of date Y: followed by yyyy: equinox of yyyy.0

Table 2 (continued)

IDENTIFIER	UNITS	FORMAT	PARAMETER	PERMITTED VALUES
<u>BANDS</u>		2A1	frequency band indications for AB and CD paramp	LL, CC, UU, KK, CU, UC (L = 21-18 cm, C = 6 cm, U = 2 cm, K = 1.2 cm) First letter for AB I.F., second for CD I.F.)
<u>MODE</u>		A2	observing mode	Blank or ?: normal interferometer PA, PB, PC, PD: single dish pointing. IA, IB, IC, ID: interferometer pointing. D: delay center determination
<u>CALIBRATOR</u>		A1	calibrator indication	Blank or ?: no calibrator C: calibrator
<u>GAINCODE</u>		I1	$2 \log(\text{gain factor})$	$0 \leq \text{GAI} \leq 9$
<u>WIDTH</u>		4I1	Bandwidth codes for A, B, C and D IFs	0-9, C, D, E $n = 50/2^n$ MHz C = 4 Hz, D = 1.5 MHz, E = 0.5 MHz.
<u>FEATUREVEL</u>	km/sec	F12.X	feature velocity for line observations	
<u>//LO - CARD</u> <u>.....</u>				
<u>LOTUNE</u>		2I1	tuning digits for paramps AB and CD	blank, 0, 1, 2, 3,
<u>LO1AB</u>	GHz	F7.1	first LO frequency for AB	
<u>LO1CD</u>			and for CD	

Table 2 (continued)

IDENTIFIER	UNITS	FORMAT	PARAMETER	PERMITTED VALUES
<u>LOSYA</u>	MHz	I4	synthesizer frequency IF A	$2710 \leq \text{LOSYA} \leq 3590$
<u>LOSYB</u>			synthesizer frequency IF B	" " B "
<u>LOSYC</u>			synthesizer frequency IF C	" " C "
<u>LOSYD</u>			synthesizer frequency IF D	" " D "
<u>LOFIAC</u>	MHz	F10.4	fine tuning synthesizer setting	
<u>LOFIBD</u>			for AC and BD respectively	
<u>LORECEIVER</u>		A8	name of file with receiver parameters (IF parameters)	
<u>LOSUBREFL</u>		A8	name of file with subreflector and related data	
<u>//AN-card (cable wrap card)</u>				
<u>AN01</u>		2A1	characters controlling the motion of antennas 1 through 27	<u>1st character (elevation):</u> U: use elevations > 90° D: don't use elevations > 90° Blank or ?: don't care <u>2nd character (azimuth):</u> L: request counter clockwise rotation in Az R: request clockwise rotation in Az Blank or ?: don't care
<u>AN26</u>				
<u>AN27</u>				

Table 2 (continued)

IDENTIFIER	UNITS	FORMAT	PARAMETER	PERMITTED VALUES
<u>ANARM</u>		3(3A1)	characters controlling the motion of the antennas on the three arms	<u>1st character of three:</u> N: North arm S: Southeast arm W: Southwest arm A: entire array
<u>//LI-card (line L0-card)</u>				2nd character=elevation character 3rd character=azimuth character cf. AN01 ... AN27
<u>LIWIDTH</u>		4I1	default bandwidth codes for A, B, C and D IFs	cf. WIDTH
<u>LILC</u>		A1	line/continuum indicator	L: Line, C: continuum
<u>LIVSYSTEM</u>		A1	velocity-system-indicator	T: topocentric, H: heliocentric L: Local standard of rest
<u>LIUNITS</u>		A1	velocity units indicator	Blank or V: km/sec F: kHz
<u>LIMODE</u>		A3	correlator-mode-code	

Table 2 (continued)

IDENTIFIER	UNITS	FORMAT	PARAMETER	PERMITTED VALUES
<u>LITRANSITION</u>		A4	line-transistron-code	
<u>LIRESTFREQ</u>	MHz	F16.X	line-rest frequency	
<u>LIEXTERNAL</u>	MHz	F20.X	external L0-frequency	
<u>//PM-card (planetary motion card)</u>				
<u>PMDRA</u>	s/d	F10.3	dRA/dt	
<u>PMDDEC</u>	arcsec/d	F10.3	dDEC/dt	
<u>PMEPOCH</u>	h,m,s	I2[,1X,I2 [,1X,I2]]	epoch of the coordinates given in observation request card in UT; must be on same UT date!	