Chapter 16

Calibration and Editing

16.1 Introduction

This chapter will describe the system of routines to edit and calibrate "raw" data multi-source uv data and single dish data files. This chapter is intended to be used both by programmers writing calibration or editing application software and programmers maintaining the more fundamental routines.

The basic design of this system is patterned after the now defunct VLA calibration system implemented on a Dec-10. All data for a single observing band (e.g., C band) with compatable sets of frequencies and polarizations are kept in raw form in a single file. Tables which contain calibration and editing information are manipulated until the user is satisfied with the results (or simply exhausted). The calibration and editing tables may then be applied to the data to produce single-source, calibrated and edited data files in the traditional AIPS form.

There are two parallel and overlapping calibration systems, one for interferometer (also called uv data) and the other for single dish data or other filled aperature measurments. The former system is to make amplitude and phase like corrections and the latter for amplitude and pointing corrections. These calibration systems depend heavily on the use of tables and the reader is encouraged to review the chapter on tables in this manual. Access to the raw data files is through the routines UVGET and SDGET for interferometer and single dish data. These routines can optionally calibrate and edit a selected subset of the data and return it in a specified set of Stokes' parameters.

16.2 Multi-source uv Data Files

The most useful organization of uv data for purposes of editing and calibration is to have all data which is to be calibrated together in the same file in time order. The structure of multi-source files is very similar to single-source files, both types are described in detail in the chapter on disk files. Multi-source files differ from single source files by the presence a source number random parameter, an optional FQ (frequency set[) identifier random parameter, and a number of tables extension files. Because of the similarity with single-source files, the bulk of AIPS software can process multi-source as well as single-source files if no distinction needs to be made between sources. Descriptions of the contents of the calibration and editing tables are given in Appendix C. Access to raw interferometer data is through the routine UVGET (see the detailed description at the end of this chapter). UVGET can select and optionally edit and calibrate the data.

16.2.1 Distinguishing Sources

Each logical record in a multi-source file contains a source identification number as a random parameter; the source information for the individual sources is kept in the source (SU) table. When the name, position, flux density, etc. of a given source is needed, the appropriate record of the source table should be consulted. The position of the source number random parameter is obtained by UVPGET and placed in the DUVH.INC include common as ILOCSU. See the chapters on the catalog header records and disk files for more details on determining the structure of the logical records.

16.2.2 Time Order

The most convenient order for multi-source files is time-baseline order; that is, strictly in time order. This allows indexing the data, which, in turn, allows rapid access to small portions of the file. For this reason, most of the calibration and editing software require the data to be in time order and require an index (NX) table or, at least, are much more efficient if one is available.

16.2.3 Scans

Observations generally consist of a series of "scans" or sequences of data on the same source. Since the duration of some of these observations may be quite long, the AIPS definition of a scan is more general and a long sequence on the same source may be broken into several "scans". The definition of a scan is the data described by an index (NX) table entry, which must be a contiguous sequence of data all on the same source.

16.2.4 Subarrays

Some data sets may contain observations of several sources at the same time by different members of the array; this is especially common in VLBI data, for which all array elements will not be able to see a given source over the same time range. This complication is dealt with by the concept of subarrays. A further complication is that the AIPS definition of subarrays allows partially or completely disjoint sets of antennas in different subarrays; this allows combining data from distinct synthesis arrays, while keeping track of what data came from where. It may be desirable to calibrate data from separate arrays together to insure that the phases all refer to the same place on the sky.

Each subarray in a data base will have a corresponding antenna (AN) table defining the properties of the array members and the actual observing date and frequency; this is the case even if the contents of the antenna files are identical. More details on the use of subarrays can be found in the chapter on disk files and later in this chapter.

16.2.5 Compressed Data

Data may be kept in "Compressed" form with individual correlation values packed into a single real word. The details of this packing are machine dependent and is done by the routines ZUVPAK and ZUVXPN. These packed values may be blanked by magic value blanking to indicate that they are flagged and each visibility record will have a single weight and scaling random parameters. The use of the compressed format for interferometer data is transparent when UVGETUVGET is used for access as the data is returned in expanded form.

16.2.6 Frequency Sets (FQ id)

A given set of data may have data taken with different sets of frequencies or bandwidths at different times; as for example, in rotation measure experiments or frequency switched spectral line observations. These may be kept in the same data set if they all have the same number of IFs, channels and polarizations. A given set of frequencies and bandwidths are specified by an entry in the FQ table. A random parameter in each visibility record labeled "FQID" specifies the entry in the FQ table that describes the frequencies and bandwidths of that record.

16.2.7 Tables

The manipulation of tables is at the heart of the calibration and editing software and much information about the contents of the associated uv data file is contained in them. Some of the tables, such as the source (SU), antenna (AN) and frequency (FQ) tables, are necessary to interpret values in the uv data file. Other tables, such as the flag (FG), calibration (CL and CS), and solution (SN) tables, contain information which can be used to edit and/or calibrate the data.

A short description of the major tables follows; a detailed definition of each table is given at the end of this chapter.

- AN table : this table defines the array geometry, the observing date and frequency, and some time invariant properties of each array element such as the instrumental polarization parameters.
- BL table : this table contains the gain corrections which is peculiar to pairs of antennas (baseline) and cannot be factored into antenna based components.
- BP table : this table contains antenna based gain corrections for each spectral channel in a data base.
- CL table : this table contains the calibration information necessary to calibrate the contents of an interferometer data file. Multi-source files only.
- CS table : this table contains the calibration information necessary to calibrate the contents of a single dish data file. Amplitude and pointing corrections are included.
- FG table : this table contains descriptions of data to be rejected from further consideration. Multi-source files only.
- FQ table : this table gives the frequency offset of each IF from the reference frequency. This allows IFs to have arbitrary frequency offsets. Multiple sets of frequencies and/or bandwidths are allowed with the use of a "FQID" random parameter which specifies the entry in this table. The calibration routines currently (1990) only process a single FQ id at a time.
- NX table : the index table contains an index of the uv data file. Multi-source files only.
- SN table : this table contains the solutions obtained from a calibration routine. The contents of these tables are applied to the CL table for multi-source files and to the data for single-source files.
- SU table : the source table contains the information peculiar to a given source such as the name and position.

16.3 Editing Basics

The terms editing and flagging will be used interchangeably in this chapter. The processes of editing is to remove data that is defective or otherwise unwanted. This is done in the context of the calibration software by specifying the unwanted data in terms of time range, subarray, source, antenna or baseline, etc. These descriptions of the data to be removed are kept in the flag (FG) table with a short, optional description of the reason the data is unwanted.

When these flags are applied, bad data is indicated by a non-positive weight associated with the relevant complex value. If all such values in a given logical record are marked bad, then the calibration software will not pass that logical record. Editing is invoked in UVGET or SDGET by setting DSEL.INC variable $DOFLAG =$ .TRUE. and $FGVER$ to the desired FG table version number.

16.4 Interferometric Calibration Basics

Calibration is the process of removing the instrumental and atmospheric effects from the data and referring the residual phases to the desired position on the sky. Note: the phase of the visibility data is the residual phase after the model phase at the reference position is subtracted. Most of the instrumental and all of the atmospheric effects are a function of antenna rather than baseline. Baseline dependent calibration, if done at all, is only done after the antenna based calibration. The bulk of the calibration software is for antenna based calibration.

Calibration information can be derived from external sources, such as system temperature measurements, atmospheric models, etc., or from internal methods, such as observations of sources with known flux density, position and structure (calibrator sources).

A related means of calibration is self-calibration, in which the current model of a source (e.g., CLEAN components) is iteratively used to determine the instrumental or atmospheric effects directly from the observations of the source of interest. This technique may lead to a considerable increase in the dynamic range of the final image at the cost of losing precise positional and/or flux density scale information.

The information needed to calibrate the data in a multi-source file is kept in the calibration (CL) table. In the case of external calibration, the corrections may be applied directly to the CL table. The process of internal calibration interacts with editing and other uncertainties and so frequently takes a number of iterations.

The calibration information needed is: (1) a complex gain for each correlator value (complex visibility), and, for VLBI data, (2) a residual group delay to correct the derivative of phase with frequency, and (3) a residual phase delay rate (sometimes called fringe rate) to correct the time derivative of the phase. The calibration to be applied to a given visibility measurement is interpolated from the CL table entries surrounding the visibility measurement in time.

16.4.1 Internal Calibration

Internal calibration is a two-step process; the first step is usually to determine calibration values from calibrator source observations; these solutions are kept in a solution (SN) table. The second step is to apply these solutions to the calibration (CL) table. Solutions for different calibrator sources may need to be determined independently and these solutions are put in separate SN tables. Thus, when the SN table are to be applied to the CL tables, there may be several such tables.

The reference times of the SN and CL tables entries will, in general, not be the same; the CL table entries are usually generated when the data is first read into the system. Solutions may be determined from all the data from a given scan or from shorter periods. The SN tables are applied to the CL tables by interpolating between SN table entries to the times of the CL table records and then applying the residual corrections from the SN table to the CL table.

An additional complication is that a given CL table may have been applied to the data before the determination of a set of solutions. In this case, the SN table needs to be applied to the CL table which was used to calibrate the data. In general, it is the responsibility of the user to insure that the correct version of the CL table is used.

16.4.2 Smoothing

Under some circumstances, the user will wish to smooth some aspect of the solutions (e.g., the group delay residual) before applying it to the CL table. This is done by concatenating the SN tables, adjusting them to a common reference antenna, and then smoothing the combined table before applying it to the CL table.

16.4.3 Reference Antennas

An interferometer only measures phase differences between antennas, so internal calibration cannot determine absolute phases. It is usually the practice to refer the phases derived from internal calibration to a given antenna known as the reference antenna. Phase corrections (including delay and rate) cannot be smoothed or interpolated using solutions based on different reference antennas. Solutions using different reference antennas must be re-referenced to the same antenna before smoothing or interpolation. Since all calibration in AIPS involves interpolation, the solutions must always be re-referenced to the same antenna. Routine CALREF rereferences all phase like data in a given SN table to a common antenna. Details of this routine are given at the end of this chapter.

16.5 Observing Model

The phases of the visibility data are actually the residuals from a model phase. The total phase and its cousins (i.e., phase delay, group delay, and their time derivatives) are useful values in their own right. These values can be used to determine the locations of the antennas and the sources, and the orientation of the earth, to high accuracy.

The total phase-like model values are also necessary if high accuracy earth models (precession, UT1 corrections, polar motion, solid earth tides, ocean loading, etc., etc.) are to be used to apply phase corrections to the data. The need for such corrections increases with baseline length and is essential for phase coherent interferometers with baselines of length a thousand km or more.

The total model applied to the data is kept in the calibration (CL) table. The model is expressed in terms of delays and their time derivatives. Because dispersive media are involved, both the group and phase delay must be kept. The total model is arbitrarily divided into three parts: (1) a sinusoidal term, (2) an "atmospheric" term and (3) a "clock" term. The sinusoidal term is intended to contribute the bulk of the delay, being the delay due to the positions of the array elements on the earth or of a satellite in orbit. Since the sinusoidal term is purely geometrical, there is no distinction between group and phase delay. (Note: use of a sinusoidal term may be changed to a polynomial in the future).

The distinction between the "atmospheric" and "clock" terms is, as yet, poorly defined, but both are allowed to have group and phase delay components. Whenever a new residual correction is determined, e.g., when an SN table is applied to the CL table, the "atmospheric" or "clock" total model values need to have the corresponding correction made. This procedure will result in total model values when the calibration process is complete.

16.6 Applying Calibration to Interferometer Data

This section describes the various calibration and spectral smoothing operations that can be invoked using UVGET for interferometer data. These functions are controlled by variables in the commons in DSEL.INC as described below. The values in the DSEL.INC commons are most convienently initialized using routine SELINI. Desired options may then be explicitly set.

16.6.1 Amplitude, Phase, Delay and Rate

Amplitude, phase, delay and fringe rate corrections are invoked by setting DSEL.INC variable *DOCAL*=.TRUE. and *CLUSE* to the version number of the CL (multi-source) or SN (single-source file) table to be applied. This corrections is done in DATCAL using interpolated values in tables maintained by CGASET and C-SLGET. An amplitude correction is applied to correct for amplitude loss due to time averaging if a fringe rate correction is made; this corrections is baseline dependent. To make this correction DSEL.INC variable *DXTIME* should be set to the integration time of the data in days.

The weights associated with the data may be calibrated by the amplitude correction. This is necessary if the weights are to be proper statistical weights but this is inconsistent with older useage. The question of calibration of weights is frequently left as a user option and is enabled by setting DSEL.INC variable *DOWTCL*=.TRUE.

16.6.2 Baseline Dependent Calibration

Instrumental errors may be introduced in a mannar which cannot be factored into antenna specific components. The corrections for this type of error is called baseline dependent calibration. Baseline dependent errors may be due to such effects as bandpass mismatches or defects in the correlation process. In principle, these effects can be multiplicative or additive although at present only multiplicative corrections are fully implemented. Baseline dependent corrections, both additive and multiplicative, are contained in the BL table. Baseline dependent calibrations maybe time variable, in which case the applied correction is interpolated in time. Baseline dependent corrections are applied in DATCAL using a table maintained by BLSET. Baseline dependent calibration is invoked by setting DSEL.INC variable *DOBL*=.TRUE. and *BLVER* to the version of the desired BL table.

16.6.3 Bandpass Calibration

Data in which multiple frequency samples are present in the same IF generally require a frequency dependent amplitude and/or phase correction. These are generally needed due to imperfect filters in the signal processing system in the individual antennas and thus may be factored into antenna based components. These corrections may be slowly variable with time. Amplitude and phase corrections may be determined from interferometric observations or amplitude corrections from autocorrelation measurments. These bandpass corrections are kept in the BP table and are applied by DATBND using tables maintained by BPGET, SCLOAD and SCINTP. The call sequence to these routines is described in detail at the end of this chapter. These routines have three modes of operation: 1) a constant set of corrections; 2) using the corrections closest in time; and 3) full interpolation in time of the bandpass corrections. Several scratch files may be used in this process. bandpass calibration is invoked using DSEL.INC variable *DOBAND*=1,2 or 3 for the three option described above and *BPVER* set to the desired BP table.

16.6.4 Spectral Smoothing

Spectral measurments are sometimes smoothed to increase the signal to noise ratio in features wider than the channel separation or as a filter to remove the Gibbs Phenomenon (ringing due to truncation of the lag function). In the later case, a Hanning smoothing is effective. The currently available smoothing methods are Hanning, Gaussian, boxcar and sinc. Spectral smoothing is invoked by setting DSEL.INC variable *DOSMTH* to .TRUE. and specifing the desired smoothing in array *SMOOTH*. Initialization for smoothing is done by SETSM and the smoothing is done by SMOSP.

16.6.5 Polarization Calibration

The instrumental polarization calibration constants for a single FQ id are contained in the subarray AN table. Several possible parameterizations of the instrumental polarization are possible but only a linear approximation for circular or linear (but not mixed) feeds are fully supported. More details are given in the description of the AN table given in Appendix C. If DSEL.INC variable *DOPOL*=.TRUE. then routine DATPOL will correct for the instrumental polarization and remove the effects of parallactic angle rotation for antellas with an alt-az mount. The VLA is recognized as a special case in which all antennas have the same parallactic angle. This requires that no parallactic angle corrections have been made prior to this.

At low frequencies a further corrections for ionospheris Faraday rotation will also be required. This information, if present, is kept in the CL table and is applied if polarization calibration is applied. This correction is applied by DATPOL using tables in DSEL.INC maintained by CSLGET and CGASET using values read from the CL table.

16.7 Data Selection

Data may be selected using a number of criteria specified via commons in DSEL.INC. Sources may be specified by name using character array *SOURCS*, a single qualifier *SELQUA* or calibrator code *SELCOD* (' ' means any, '* ' means any non blank calibrator code, '-CAL' means all but calibrators and anything else requires a match). Source selection is done by routine SOUFIL which indicates the selected sources by source id number in DSEL.INC variables *NSOUWD*, *DOSWNT*, *SOUWAN* and *SOUWTB*.

Selection by time is specified in array *TIMRNG* which gives the start date, hour, minute, second, end day, hour, minute and second. Day numbers are zero relative with respect to the subarray reference date. All zeroes mean all times are selected. A list of antennas whose data is to be included is specified by array *ANTENS*. If all elements of *ANTENS* are positive then selected data are to come from only those antennas. If any value is negative, then data using the antennas specified by the absolute values of the elements in *ANTENS* are to be ignored. The subarray may be selected using *SUBARR* for which zero means all. Data may be specified to be in an annulus in the uv plane by *UVRNG*. The first and second elements of this array are the inner and outer radii of the annulus in 1000's of wavelengths and zeroes mean all. The annulus is defined at the reference frequency of the subarray.

A range of spectral channels can be specified using *BCHAN* and *ECHAN* which are the lowest and highest channel numbers in each IF; zero means all. Similarly, a range of IFs may be specified with *BIF* and *EIF*. The desired Stokes' (polarization) type may be specified by *STOKES*. This will cause the Stokes' type requested to be returned if possible. The setup for this translation is done in DGINIT and the translation is done in routine DGGET. A blank value of *STOKES* means don't translate; other values are 'I', 'Q', 'U', 'V', 'IQU', 'IQUV', 'IV', 'RR', 'LL', 'RL', 'LR' 'HALF' (parallel pol.) and 'FULL' (RR,LL,RL and LR). Setting variable *DOACOR*=.TRUE. causes UVGET to return any auto correlation data found; *DOXCOR*=.TRUE. requests cross correlation data. *DOFQSL*=.TRUE. requests selection by FQ id where the desired FQID is given by *FRQSEL*. The value of *FRQSEL* may be determined directly by the user or specified by Frequency and bandwidth; this is done in routine FQMATC.

16.8 Table Access Routines

Much of the table access is built into the data access routines, but some tables (e.g., the source) may need to be accessed more generally. Most of the commonly used tables will have special access routines which are described in detail in the chapter on tables. The detailed descriptions of the contents of these tables are found in Appendix C.

16.9 Calibration Table Routines

The manipulation of the SN tables and their application to a CL table is done by a single AIPS task, CLCAL using routine CLUPDA. This routine concatenates SN tables and re-references the phases to a common reference antenna, smooths the SN table and applies the combined SN table to specified entries in an input CL table, and finally writes them to an output table. If no CL table exists in a multi-source file, one is created and the SN table copied into it. The details of the call sequence of CLUPDA are at the end of this chapter.

The detailed contents of the calibration related tables are subject to change with time. To minimize the difficulties associated with this there are several routines that check if a given table is of the current format and reformats it if it is an old version. These routines are briefly described in the following and in detail at the end of this chapter.

- BLREFM - Checks existence of BL table, changes format if necessary.
- BPREFM - Checks existence of BP table, changes format if necessary.
- CLREFM - Checks existence of CL table, changes format if necessary.
- SNREFM - Checks existence of SN table, changes format if necessary.

16.10 Data Access Routines

The principal means of accessing multi-source data is through the routine UVGET. This routine will apply a variety of selection criteria, translate polarization types, and, optionally, apply calibration and editing tables. A detailed description of UGVET is given else where in this chapter. A related routine, CALCOP, uses UVGET to create and fill a file with the desired data. UVGET also calls DGHEAD which prepares a catalog header record which describes the output data; this is left in DSEL.INC array *CATBLK*. The catalog header record for the input uv data is in array *CATUV*.

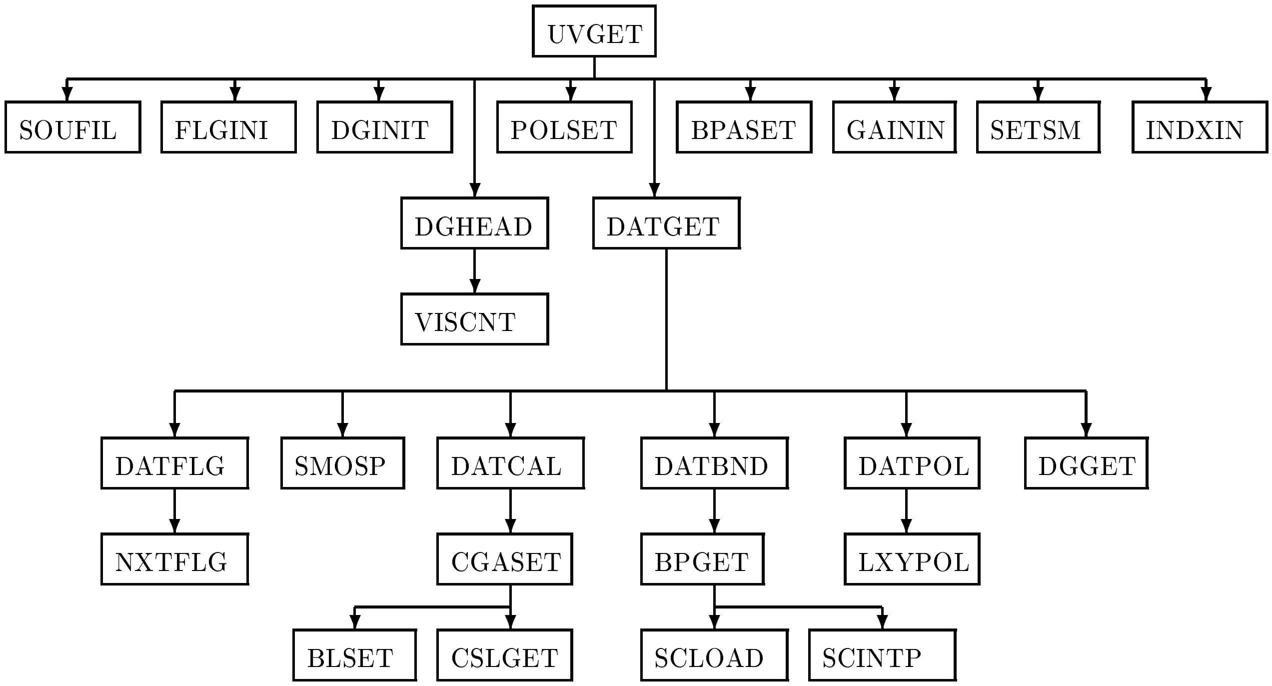
UVGET and CALCOP will work on single-source files as well as multi-source files. This allows calibration software written for multi-source files to work for single-source files whenever possible.

The communication of the selection criteria, flagging and calibration information, etc. is through a common contained in the include DSEL.INC. The commons in these includes contain most of the variables and arrays needed by the inner workings of UVGET.

16.10.1 Structure of the Interferometer Data Access System

UVGET calls a hierarchy of routines to do the necessary functions. While most programmers using UVGET will not need to be familiar with the details of the inner workings of the UVGET system, some applications may require knowledge of, or modifications to, this system.

The relations between the major routines of the UVGET system are shown in the following diagram:



Interferometer Data Calibration

16.10.2 Data Access Routines

In the following, a short description is given for each of the major routines in the data access system; detailed descriptions of the call sequences etc. are given later in this chapter.

1. UVGET : the top level data access routine. UVGET controls the initialization of the various files and arrays, calls DATGET to process the data, and closes the necessary files when done.
2. SOUFIL : this routine converts the list of sources to source numbers, which are filled into a common.
3. GAININ : initializes the gain (CL or SN) table to be applied to the data.
4. DGINIT : sets up to translate data (e.g., RR, LL to IPOL), if necessary.
5. POLSET : initializes common arrays needed for polarization calibration.
6. BPASET : initializes common arrays and possibly scratch files for bandpass calibration.
7. SETSM : initializes common variables for spectral smoothing.
8. INDXIN : initializes the index table I/O and finds the first relevant index record.
9. FLGINI : initializes reading flagging (FG) table.
10. DGHEAD : fills the catalog header record to correspond to the output data.
11. VISCNT : estimates the number of visibility records requested from the index table.

12. DATGET : reads data, applies flagging and calibration, and translates polarization.
13. DATFLG : flags data.
14. SMOSP : smooth spectra.
15. DATCAL : calibrates data.
16. DATPOL : Applies instrumental polarization corrections, parallactic angle corrections and ionospheric Faraday rotation corrections. Converts linear feed data to circular if necessary.
17. DATBND : applies bandpass correction.
18. DGGET : translates polarization.
19. NXTFLG : manages the internal arrays containing the currently active flagging criteria.
20. CGASET : interpolates gain table entries to current time.
21. CSLGET : finds gain table entries on both sides of the current time and reads the values into a common array.
22. BLSET : sets common arrays for baseline dependent calibration.
23. LXYPOL : determines time dependent polarization correction matrix for data observed with linear feeds.
24. BPGET : sets current bandpass corrections in common.
25. SCLOAD : manipulates Bandpass correction scratch files.
26. SCINTP : interpolates bandpass corrections in time.

16.11 Example Using UVGET

The following example illustrates the use of UVGET to access a multisource uv data file with calibration and editing as requested by the user through adverb values passed to the routine. In this example the user specified calibration and flagging options are applied and the selected data written to a scratch file that is created by CALCOP.

```
LOCAL INCLUDE 'PARMS.INC'
C                                       Common with task adverb values
      REAL      DISKIN, SEQIN, XQUAL, XTIME(8), XBCHAN, XECHAN, XBIF,
     *   XEIF, XDOCAL, XDOPOL, XSUBA, XFLAG, XGUSE, XBLVER, XDOBND,
     *   XBPVER, XSMOTH(3), XINT, XBAND, XFREQ, XFQID
      CHARACTER NAMEIN*12, CLAIN*6, XSOUR(30)*8, XCALCO*4, XSTOK*4
      COMMON /PARMS/ DISKIN, SEQIN, XQUAL, XTIME, XBCHAN, XECHAN, XBIF,
     *   XEIF, XDOCAL, XDOPOL, XSUBA, XFLAG, XGUSE, XBLVER, XDOBND,
     *   XBPVER, XSMOTH, XINT, XBAND, XFREQ, XFQID
      COMMON /CPARMS/ NAMEIN, CLAIN, XSOUR, XCALCO, XSTOK
LOCAL END
      SUBROUTINE CALIT (IDISK, ICNO, CATIN, SCRNO, IRET)
C-----------------------------------------------------------------------
C   Routine to calibrate and edit a user selected set of data and
C   write it to a scratch file.
C   Inputs:
C      IDISK    I       Input file disk number.
C      ICNO     I       Input file catalog slot number.
C      CATIN    I(256)  Input file catalog header record.
```

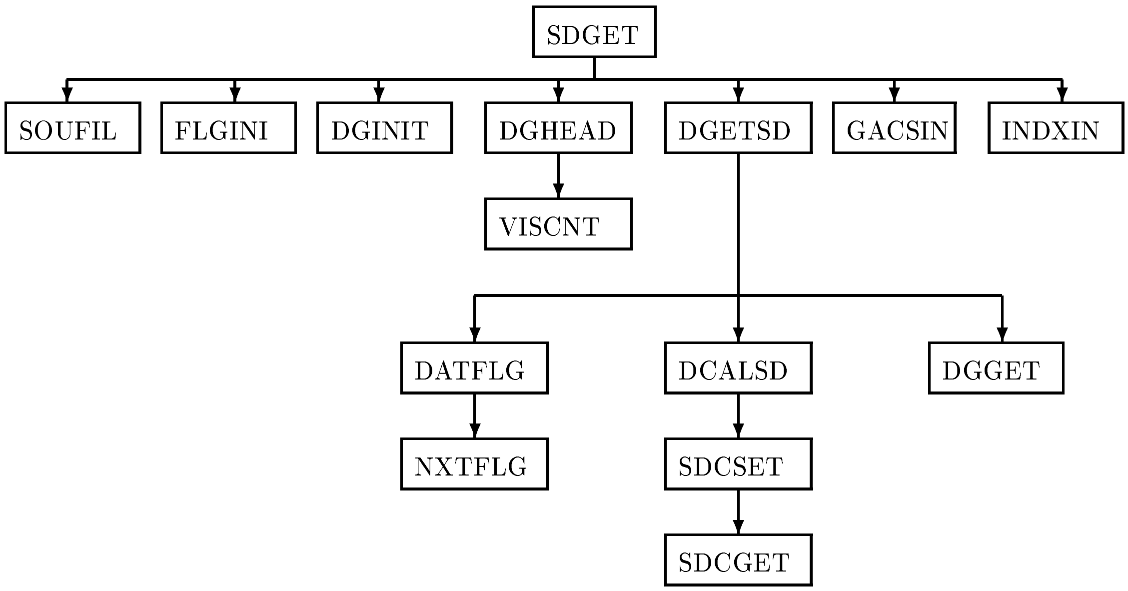
```
C   Inputs from common (PARMS.INC):
C      NAMEIN   C*12    Input uv data file name.
C      CLAIN    C*6     Input uv data file class.
C      DISKIN   R       Input uv data file disk no.
C      SEQIN    R       Input uv data file sequence no.
C      XSOUR    C(30)*8 List of source names
C      XQUAL    R       Qualifier selected
C      XCALCO   C*4     Calibrator code selected.
C      XTIME    R(8)    Time range, start d,h,m,s, end d,h,m,s
C      XSTOK    C*4     Desired Stokes parameter
C      XBCHAN   R       Lowest channel number to select.
C      XECHAN   R       Highest channel number to select.
C      XBIF     R       Lowest IF number to select.
C      XEIF     R       Highest IF number to select.
C      XDOCAL   R       If > 0 then apply calibration.
C      XDOPOL   R       If > 0 then apply polarization calibration.
C      XSUBA    R       Subarray number.
C      XFLAG    R       Flag (FG) table version to apply.
C      XGUSE    R       Calibration (CL or SN) table to apply.
C      XBLVER   R       Baseline dependent (BL) cal table to apply.
C      XDOBND   R       Bandpass calibration code
C      XBPVER   R       Bandpass (BP) table to apply
C      XSMOTH   R(3)    Spectral smoothing parameters.
C      XINT     R       Integration time of data in seconds.
C      XBAND    R       Selected bandwidth (Hz)
C      XFREQ    R       Selected frequency (Hz)
C      XFQID    R       Selected FQ id.
C   Outputs:
C      SCRNO    I       Scratch file number in DFIL.INC common.
C      IRET     I       Return code, 0=>OK else failed.
C-----------------------------------------------------------------------
      INTEGER   IDISK, ICNO, CATIN(256), SCRNO, IRET
C
      INTEGER   XBUFSZ
C                                      XBUFSZ = buffer size (words)
      PARAMETER (XBUFSZ=4096)
      INTEGER    IROUND, BUFSZ, DISK, LUN, I, INVER, OUTVER, LUNI, LUNO
      LOGICAL    MATCH
      REAL       BUFFER(XBUFSZ), RPARM(X), VIS(3,X)
      INCLUDE 'INCS:PUVD.INC'
      INCLUDE 'PARMS.INC'
      INCLUDE 'INCS:DSEL.INC'
      INCLUDE 'INCS:DMSG.INC'
      INCLUDE 'INCS:DHDR.INC'
      INCLUDE 'INCS:DUVH.INC'
C-----------------------------------------------------------------------
C                                      Initialize DSEL.INC
      CALL SELINI
C                                      Put selection criteria into
C                                      DSEL.INC
C                                      Select file
      UNAME = NAMEIN
      UCLAS = CLAIN
      UDISK = DISKIN
```

```
      USEQ = SEQIN
C                                       Select data
      DO 10 I = 1,30
         SOURCS(I) = XSOUR(I)
 10      CONTINUE
      SELQUA = IROUND (XQUAL)
      SELCOD = XCALCO
      CALL RCOPY (8, XTIME, TIMRNG)
      STOKES = XSTOK
      BCHAN = IROUND (XBCHAN)
      ECHAN = IROUND (XECHAN)
      BIF = IROUND (XBIF)
      EIF = IROUND (XEIF)
C                                       Select flagging
      FGVER = IROUND (XFLAG)
C                                       Select calibration
      DOCAL = XDOCAL .GT. 0.0
      DOPOL = XDOPOL .GT. 0.0
      SUBARR = IROUND (XSUBA)
      CLUSE = IROUND (XGUSE)
      BLVER = IROUND (XBLVER)
      DOBAND = IROUND (XDOBND)
      BPVER = IROUND (XBPVER)
C                                        Spectral smoothing
      DO 20 I = 1, 3
         SMOOTH(I) = IROUND (XSMOTH(I))
 20      CONTINUE
      DOSMTH = SMOOTH(1) .GT. 0
C
      DXTIME = XINT / 86400.0
C                                       Freq id
      IF (XBAND.GT.0.0) SELBAN = XBAND
      IF (XFREQ.GT.0.0) SELFRQ = XFREQ
      FRQSEL = IROUND (XFQID)
      IF (FRQSEL.EQ.0) FRQSEL = -1
      LUN = 28
      CALL FQMATC (IDISK, ICNO, CATIN, LUN, SELBAN, SELFRQ, MATCH,
     *   FRQSEL, IRET)
      IF (.NOT.MATCH) THEN
         IRET = 5
         MSGTXT = 'NO MATCH TO SELBAND/SELFREQ ADVERBS - CHECK INPUTS'
         GO TO 990
         END IF
      IF (IRET.GT.0) GO TO 999
C                                       Init call to UVGET, note that
C                                       RPARM and VIS are not used here.
      CALL UVGET ('INIT', RPARM, VIS, IRET)
      IF (IRET.NE.0) GO TO 999
C                                       Call CALCOP to process file.
      DISK = 0
      SCRNO = 0
      BUFSZ = XBUFSZ * 2
      CALL CALCOP (DISK, SCRNO, BUFFER, BUFSZ, IRET)
      IF (IRET.NE.0) GO TO 999
```

```
C                                         Copy FQ table
      INVER = 1
      OUTVER = 1
      LUNO = 28
      LUNI = 29
      CALL CHNCOPL, INVER, OUTVER, LUNO, LUNI, DISKIN, DISK, IUCNO,
     *     SCRNO, CATUV, CATBLK, BIF, EIF, FRQSEL, BUFFER, UBUFF, PBUFF,
     *     IREF
      GO TO 999
C                                         Error message
 990  CALL MSGWRT (8)
C
 999  RETURN
C-----------------------------------------------------------------------
      END
```

16.12 Single Dish Data

"Single Dish" data consists of sky brightness measurments or spectra made at arbitrary positions on the sky. The format of this data is very similar to that of interferometry data with sky brightness and the baseline subtracted constituting the "real" and "imaginary" parts. A detailed description of the single dish format is given in the chapter on disk I/O. Access to single dish data is through routine SDGET which can optionally calibrate amplitudes and/or positions and remove baselines. Many of the single dish calibration and editing routines are the same as used for interferometery data. This is especially true for flagging (FG table) and indexing (NX table). For purposes of selection and editing the "Feed Number" takes the place of "baseline" code in interferometer data. The discussion of editing for interferometry data also applied to single dish data. Compression of single dish data is not supported. The structure of the single dish calibration routines is shown in the following diagram:



Single Dish Data Calibration

The functions of the single dish calibration routines is briefly described in the following and detailed descriptions are given at the end of this chapter.

1. SDGET : the top level data access routine. SDGET controls the initialization of the various files and arrays, calls DGETSD to process the data, and closes the necessary files when done.
2. SOUFIL : this routine converts the list of sources to source numbers, which are filled into a common.
3. GACSIN : initializes the calibration (CS) table to be applied to the data.
4. DGINIT : sets up to translate data (e.g., RR, LL to IPOL), if necessary.
5. FLGINI : initializes reading flagging (FG) table.
6. INDXIN : initializes the index table I/O and finds the first relevant index record.
7. DGHEAD : fills the catalog header record to correspond to the output data.
8. DGETSD : reads data, applies flagging and calibration, and translates polarization.
9. VISCNT : estimates the number of visibility records requested from the index table.
10. DATFLG : flags data.
11. NXTFLG : manages the internal arrays containing the currently active flagging criteria.
12. DCALSD : calibrates data.
13. SDCSET : interpolates calibration table entries to current time.
14. SDCGET : finds calibration table entries on both sides of the current time and reads the values into a common array.
15. DGGET : translates polarization.

16.13 Text of INCLUDE files

There are several types of INCLUDE file which are distinguished by the first character of their name. Different INCLUDE file types contain different types of Fortran declaration statements as described in the following list.

- Pxxx.INC. These INCLUDE files contain declarations for parameters and the PARAMETER statements.
- Dxxx.INC. These INCLUDE files contain Fortran type (with dimension) declarations, COMMON and EQUIVALENCE statments.
- Vxxx.INC. These contain Fortran DATA statements.
- Zxxx.INC. These INCLUDE files contain declarations which may change from one computer or installation to another.

16.13.1 DSEL.INC

```
C                                                                   Include DSEL.
C                                       Commons for UVGET use
      INTEGER   XCTBSZ, XBTBSZ, XPTBSZ, XSTBSZ, XTTSZ, XBPSZ,
     *   XBPBUF
C                                       XCTBSZ=internal gain table size
      PARAMETER (XCTBSZ=2500)
C                                       XBTBSZ=baseline table size
      PARAMETER (XBTBSZ=3500)
C                                       XPTBSZ=polar. corr. table size
```

```
      PARAMETER (XPTBSZ=16384)
C                                         XSTBSZ=Source no. table size
      PARAMETER (XSTBSZ=500)
C                                         XTTSZ=Pol. trans. table size
      PARAMETER (XTTSZ=MAXIF*MAXCHA*2)
C                                         XBPSZ=max. no. BP time entries
      PARAMETER (XBPSZ=50)
C                                         XBPBUF=internal BP I/O buffer
      PARAMETER (XBPBUF=65536)
C                                         Data selection and control
      INTEGER   ANTENS(50), NANTSL, NSOUWD, SOUWAN(XSTBSZ), SOUWTN(30),
     *   NCALWD, CALWAN(XSTBSZ), CALWTN(30), SUBARR, SMOTYP, CURSOU,
     *   NXKOLS(MAXNXC), NXNUMV(MAXNXC), MVIS, JADR(2,XTTSZ), PMODE,
     *   LRECIN, UBUFSZ, BCHAN, ECHAN, BIF, EIF, NPRMIN, KLOCSU, KLOCFQ,
     *   SELQUA, SMDIV, SMOOTH(3), KLOCIF, KLOCFY, KLOCWT, KLOCSC,
     *   NDECMP, DECMP(2,MAXIF*4), BCHANS, ECHANS, FRQSEL, FSTRED,
     *   FQKOLS(MAXFQC), FQNUMV(MAXFQC)
      LOGICAL   DOSWNT, DOCWNT, DOAWNT, ALLWT, TRANSL, DOSMTH, ISCMP,
     *   DOXCOR, DOACOR, DOWTCL, DOFQSL
      INTEGER   INXRNO, NINDEX, FSTVIS, LSTVIS, IFQRNO
      REAL      TIMRNG(8),  UVRNG(2), INTPRM(3), UVRA(2), TSTART, TEND,
     *   SELFAC(2,XTTSZ), SMTAB(2500), SUPRAD, SELBAN
      CHARACTER SOURCS(30)*16, CALSOU(30)*16, STOKES*4, INTFN*4,
     *   SELCOD*4
      DOUBLE PRECISION UVFREQ, SELFRQ
C                                         Flag table info
      REAL      TMFLST, FLGTND(MAXFLG)
      INTEGER   IFGRNO
      LOGICAL   DOFLAG, FLGPOL(4,MAXFLG)
      INTEGER   FGVER, NUMFLG, FGKOLS(MAXFGC), FGNUMV(MAXFGC),
     *   KNCOR, KNCF, KNCIF, KNCS,
     *   FLGSOU(MAXFLG), FLGANT(MAXFLG), FLGBAS(MAXFLG), FLGSUB(MAXFLG),
     *   FLGBIF(MAXFLG), FLGEIF(MAXFLG), FLGBCH(MAXFLG), FLGECH(MAXFLG)
C                                         CAL table info
      REAL      GMMOD, CURCAL(XCTBSZ), LCALTM, CALTAB(XCTBSZ,2),
     *   CALTIM(3), RATFAC(MAXIF), DELFAC(MAXIF), DXTIME, DXFREQ,
     *   LAMSQ(MAXCHA, MAXIF), IFRTAB(MAXANT, 2), IFR(MAXANT)
      INTEGER   ICLRNO, NCLINR, MAXCLR, CNTREC(2,3)
      LOGICAL   DOCAL, DOAPPL
      INTEGER   CLVER, CLUSE, NUMANT, NUMPOL, NUMIF, CIDSOU(2),
     *   CLKOLS(MAXCLC), CLNUMV(MAXCLC), LCLTAB, LCUCAL, ICALP1, ICALP2,
     *   POLOFF(4,2)
C                                         Baseline table info
      REAL      LBLTM, BLTAB(XBTBSZ,2), BLFAC(XBTBSZ), BLTIM(3)
      INTEGER   IBLRNO, NBLINR
      LOGICAL   DOBL
      INTEGER   BLVER, BLKOLS(MAXBLC), BLNUMV(MAXBLC), IBLP1, IBLP2
C                                         Polarization table.
      REAL      POLCAL(2,XPTBSZ), PARAGL(2,MAXANT), PARTIM
      INTEGER   PARSOU
      LOGICAL   DOPOL
C                                         Bandpass table
      DOUBLE PRECISION BPFREQ(MAXIF)
      REAL      PBUFF(XBPBUF), TIMENT(XBPSZ), BPTIM(3), LBPTIM, CHNBND
```

```
      CHARACTER BPNAME*48
      INTEGER   IBPRNO, NBPINR, ANTPNT(2), NVISM, NVISS, NVIST
      INTEGER   BPVER, BPKOLS(MAXBPC), BPNUMV(MAXBPC), NANTBP, NPOLBP,
     *   NIFBP, NCHNBP, BCHNBP, DOBAND, ANTENT(XBPSZ,MAXANT),
     *   BPDSK, BPVOL, BPCNO, USEDAN(MAXANT), BPGOT(2),
     *   KSNCF, KSNCIF, KSNCS, MXANUM
C                                       Channel 0 stuff
      INTEGER   FSTVS3, LREC3, LSTVS3, NREAD3, FSTRD3, KLOCW3,
     *   KLOCS3, NDECM3, DECM3(2,MAXIF*4), BIND3, RECNO3, LENBU3
      LOGICAL   ISCMP3, DOUVIN
C                                       File specification.
      INTEGER   IUDISK, IUSEQ, IUCNO, IULUN, IUFIND, ICLUN, IFLUN,
     *   IXLUN, IBLUN, IPLUN, IQLUN, LUNSBP, BPFIND, CATUV(256),
     *   CATBLK(256)
      REAL      USEQ, UDISK
      CHARACTER UNAME*12, UCLAS*6, UFILE*48
C                                       I/O buffers
      INTEGER   CLBUFF(1024), FGBUFF(512), NXBUFF(512), BLBUFF(512),
     *   BPBUFF(32767), FQBUFF(512)
      REAL      UBUFF(8192)
C                                       Character common
      COMMON /SELCHR/ SOURCS, CALSOU, STOKES, INTFN, SELCOD, UNAME,
     *   UCLAS, UFILE, BPNAME
C                                       Common for UVGET use
C                                       Data selection and control
      COMMON /SELCAL/ UVFREQ, SELFRQ,
     *   USEQ, UDISK, TIMRNG, UVRNG, INTPRM, UVRA, TSTART, TEND, UBUFF,
     *   SELFAC, SMTAB, SUPRAD, SELBAN,
     *   INXRNO, NINDEX, FSTVIS, LSTVIS, IFQRNO,
     *   DOSWNT, DOCWNT, DOAWNT, ALLWT, TRANSL, DOSMTH, ISCMP, DOXCOR,
     *   DOACOR, DOWTCL, DOFQSL,
     *   CLBUFF, FGBUFF, NXBUFF, BLBUFF, BPBUFF, FQBUFF,
     *   IUDISK, IUSEQ, IUCNO, IULUN, IUFIND, ICLUN, IFLUN, IXLUN,
     *   IBLUN, IPLUN, IQLUN, LUNSBP, BPFIND, CATUV, ANTENS, NANTSL,
     *   NSOUWD, SOUWAN, SOUWTN, NCALWD, CALWAN, CALWTN,
     *   SUBARR, SMOTYP, CURSOU, NXKOLS, NXNUMV, FQKOLS, FQNUMV,
     *   MVIS, JADR, PMODE,
     *   LRECIN, UBUFSZ, BCHAN, ECHAN, BIF, EIF, NPRMIN, KLOCSU,
     *   KLOCFQ, SELQUA, SMDIV, SMOOTH, KLOCIF, KLOCFY, KLOCWT,
     *   KLOCSC, NDECMP, DECMP, BCHANS, ECHANS, FRQSEL, FSTRED
C                                       FLAG table info
      COMMON /CFMINF/ TMFLST, FLGTND, IFGRNO, DOFLAG, FLGPOL,
     *   FGVER, NUMFLG, FGKOLS, FGNUMV, KNCOR, KNCF, KNCIF, KNCS,
     *   FLGSOU, FLGANT, FLGBAS, FLGSUB, FLGBIF, FLGEIF, FLGBCH, FLGECH
C                                       CAL table info
      COMMON /CGNINF/ GMMOD, CURCAL, LCALTM, CALTAB, CALTIM, RATFAC,
     *   DELFAC, DXTIME, DXFREQ,
     *   ICLRNO, NCLINR, MAXCLR, CNTREC,
     *   DOCAL, DOAPPL,
     *   CLVER, CLUSE, NUMANT, NUMPOL, NUMIF, CIDSOU, CLKOLS, CLNUMV,
     *   LCLTAB, LCUCAL, ICALP1, ICALP2, POLOFF,
     *   LAMSQ, IFRTAB, IFR
C                                       BL table info
      COMMON /CBLINF/ LBLTM, BLTAB, BLTIM, BLFAC,
```

```
     *   IBLRNO, NBLINR,
     *   DOBL,
     *   BLVER, BLKOLS, BLNUMV, IBLP1, IBLP2
C                                      Pol. table
      COMMON /CPLINF/ POLCAL, PARAGL, PARTIM, PARSOU, DOPOL
C                                      BP table
      COMMON /CBPINF/ BPFREQ,
     *   PBUFF, TIMENT, BPTIM, LBPTIM, CHNBND,
     *   IBPRNO, NBPINR, ANTPNT, NVISM, NVISS, NVIST,
     *   BPVER, BPKOLS, BPNUMV, NANTBP, NPOLBP, NIFBP, NCHNBP, BCHNBP,
     *   DOBAND, ANTENT, BPDSK, BPVOL, BPCNO, USEDAN, BPGOT,
     *   KSNCF, KSNCIF, KSNCS, MXANUM
C                                      Channel 0 common
      COMMON /CHNZ/ FSTVS3, LREC3, LSTVS3, NREAD3, FSTRD3, KLOCW3,
     *   KLOCS3, NDECM3, DECM3, BIND3, RECNO3, LENBU3,
     *   ISCMP3, DOUVIN
C
      COMMON /MAPHDR/ CATBLK
C                                                                 End DSEL.
```

16.13.2 PUVD.INC

```
C                                                                 Include PUVD
C                                      Parameters for uv data
      INTEGER   MAXANT,  MXBASE,  MAXIF, MAXFLG, MAXFLD, MAXCHA
C                                      MAXANT = Max. no. antennas.
      PARAMETER (MAXANT=45)
C                                      MXBASE = max. no. baselines
      PARAMETER (MXBASE= ((MAXANT*(MAXANT+1))/2))
C                                      MAXIF=max. no. IFs.
      PARAMETER (MAXIF=15)
C                                      MAXFLG= max. no. flags active
      PARAMETER (MAXFLG=1000)
C                                      MAXFLD=max. no fields
      PARAMETER (MAXFLD=16)
C                                      MAXCHA=max. no. freq. channels.
      PARAMETER (MAXCHA=512)
C                                      Parameters for tables
      INTEGER MAXCLC, MAXSNC, MAXANC, MAXFGC, MAXNXC, MAXSUC,
     *   MAXBPC, MAXBLC, MAXFQC
C                                      MAXCLC=max no. cols in CL table
      PARAMETER (MAXCLC=41)
C                                      MAXSNC=max no. cols in SN table
      PARAMETER (MAXSNC=20)
C                                      MAXANC=max no. cols in AN table
      PARAMETER (MAXANC=12)
C                                      MAXFGC=max no. cols in FG table
      PARAMETER (MAXFGC=8)
C                                      MAXNXC=max no. cols in NX table
      PARAMETER (MAXNXC=7)
C                                      MAXSUC=max no. cols in SU table
      PARAMETER (MAXSUC=21)
C                                      MAXBPC=max no. cols in BP table
      PARAMETER (MAXBPC=14)
```

```
C                                         MAXBLC=max no. cols in BL table
      PARAMETER (MAXBLC=14)
C                                         MAXFQC=max no. cols in FQ table
      PARAMETER (MAXFQC=5)
C                                                             End PUVD.
```

16.14 Routines

16.14.1 BLREFM

Routine to change the format of the BL table from one containing 13/9 columns to the 14/10 columns needed by the addition of the FREQID column. NOTE: routine uses LUN 45 as a temporary logical unit number.

```
BLREFM (DISK, CNO, VER, CATBLK, LUN, IRET)
Inputs:
   DISK          I       Volume number
   CNO           I       Catalogue number
   VER           I       Version to check/modify
   CATBLK(256)   I       Catalogue header
   LUN           I       LUN to use
Output:
   IRET          I       Error, 0 => OK

Note, routine will leave no trace of its operation, i.e. BL table
will be closed on output and will have same number as one specified.
Difference will be only that number of columns has changed if that
is required.
```

16.14.2 BLSET

Fills current baseline correction table (BLFAC) by interpolation.

```
BLSET (TIME, IERR)
Inputs:
   TIME         R    Current time (of data) in days.
Inputs from DSEL.INC:
   GMMOD       R    Mean gain modulus correction, 0=>none.
   BLTAB(*,2)  R    Baseline table from BL table file.
                    The second dimension corresponds to the time
                    before and after the current data time.
                    Indexing scheme: an entry defined by ant1<ant2
                    starts in element:
      lentry * (((ant1-1)*numant-((ant1+1)*ant1)/2 + ant2) - 1) + 1
                    where lentry = 2 * NUMPOL * (EIF-BIF+1)
                    An entry contains the values in order:
                    By IF (NUMIF)
                       By Polarization (NUMPOL)
                           Real part, imaginary part.
   BLTIM(3)    R    Time of two cal. entries; third value is
                     time of current values.
   IBLP1       I    Pointer in BLTAB, BLTIM to previous time.
   IBLP2       I    Pointer in BLTAB, BLTIM to next time.
   DOBL        L    If true then use baseline correction table,
                    else initialize BLFAC.
```

```
Output:
   IERR         I    Return error code, 0=>OK else error.
Output to DSEL.INC:
   BLFAC(*)     R   Baseline dependent factors as (real, imag)
                    Includes GMMOD correction if necessary.
                    Addressing is like BLTAB.
                    Initialized to 1/GMMOD,0 if DOBL=.FALSE.
```

16.14.3 BPASET

Sets up the bandpass table array for use by DATBND.

```
BPASET (IERR)
    Inputs from DSEL.INC:
     BPBUFF(*)    I    BP table I/O TABIO buffer
     NBPINR       I    Number of BP records in file.
     NANTBP       I    Number of antennas
     NPOLBP       I    Number of IFs per group (polarizations)
     NIFBP        I    Number of IFs.
     NCHNBP       I    Number of channels
  Output:
     IERR         I    Return error code 0=>OK, else failed.
  Output to DSEL.INC:
     TIMENT(MAXBP)         R    An index array recording the times of
                                BP records in the scratch file
     ANTENT(MAXBP,MAXANT)  I    The corresponding antenna index.
     MXANUM                I    The maximum antenna number
```

16.14.4 BPGET

Gets next set of bandpass data in array PBUFF. Depending on the value of DOBAND will

1. extract data for antenna from scratch file containing averaged data.
2. extract bandpass data closest in time to visibility data
3. will do a linear interpolation in time between the entries.

If options (2) or (3) are selected the I/O rate will be very high and will slow the program down tremendously.

```
BPGET (TIME, IA1, IA2, IERR)
Inputs:
   TIME           R       Current time of data (days)
   IA1            I       First antenna to be selected
   IA2            I       Second antenna to be selected
Inputs from common /...../
   DOBAND         I       Bandpass selection option
   TIMENT
   ANTENT
Output:
   IERR           I       Return error code. 0=>OK, else error
Output to common /...../
   BPANT1         R(*)    Array containing bandpass spectrum
                          for IA1
   BPANT2         R(*)    Array containing bandpass spectrum
```

```
                              for IA2
   LBPTIM         R       Time of current calibration
   BPGOT(2)       I       Antenna numbers of present calibration.
```

16.14.5 BPREFM

Routine to change the format of the BP table from one containing 13/10 columns to the 14/11 columns needed by the addition of the FREQID column. NOTE: routine uses LUN 45 as a temporary logical unit number.

```
BPREFM (DISK, CNO, VER, CATBLK, LUN, IRET)
Inputs:
   DISK            I       Volume number
   CNO             I       Catalogue number
   VER             I       Version to check/modify
   CATBLK(256)     I       Catalogue header
   LUN             I       LUN to use
Output:
   IRET            I       Error, 0 => OK

Note, routine will leave no trace of its operation, i.e. BP table
will be closed on output and will have same number as one specified.
Difference will be only that number of columns has changed if that
is required.
```

16.14.6 CALCOP

Routine to copy selected data from one data file to another optionally applying calibration and editing information. The input file should have been opened with UVGET. Both files will be closed on return from CALCOP. Note: UVGET returns the information necessary to catalog the output file. The output file will be compressed if necessary at completion of CALCOP.

```
CALCOP (DISK, CNOSCR, BUFFER, BUFSZ, IRET)
Inputs:
   DISK     I       Disk number for catalogd output file.
                    If .LE. 0 then the output file is a /CFILES/
                    scratch file.
   BUFFER   R(*)    Work buffer for writing.
   BUFSZ    I       Size of BUFFER in bytes.
Input via common:
   LREC     I       (/UVHDR/) length of vis. record in R words.
   NRPARM   I       (/UVHDR/) number of R random parameters.
In/out:
   CNOSCR   I       Catalog slot number for if cataloged file;
                    /CFILES/ scratch file number if a scratch file,
                    IF DISK=CNOSCR=0 then the scratch is created.
                    On output = Scratch file number if created.
In/out via common:
   CATBLK   I(256)  Catalog header block from UVGET
                    on output with actual no. records
   NVIS     I       (/UVHDR/) Number of vis. records.
Output:
   IRET     I       Error code: 0 => OK,
                       > 0 => failed, abort process.
```

```
Usage notes:
(1) UVGET with OPCODE='INIT' MUST be called before CALCOP to setup
    for calibration, editing and data translation.  If an output
    cataloged file is to be created this should be done after the
    call to UVGET.
(2) Uses AIPS LUN 24
```

16.14.7 CALREF

Subroutine to adjust the reference antenna in a uv data file. The Table is first read to find all data relating ANT and REFAN. These data are then smoothed and the resulting ANT-REFAN are used in a second pass through the table to adjust data using ANT as a reference antenna to REFAN as the reference antenna. Several work arrays are passed which are used for storing, smoothing and interpolating data. The table should already be open and BUFFER should be the buffer used by TABINI (or other table opening routines).

```
CALREF (ANT, REFAN, SUB, KOLS, BUFFER, SMOTIM, MAXTIM,
  *   FREQ, WRKTIM, WORK1, WORK2, WORK3, WORK4, WORK5, IRET)
Input:
   ANT      I    Old reference antenna
   REFAN    I    New reference antenna
   SUB      I    Subarray desired
   KOLS     I(9) Array of TABIO column pointers in order:
                 antenna, ref. antenna, subarray, weight, time,
                 real, imag, delay, rate.
   BUFFER   I(*) Buffer for TABIO use; table must already be
                 open
   SMOTIM   R(3) Boxcar averaging times (days) 1=phase, 2=delay
                 3 = rate
   MAXTIM   I    Maximum number of times (dim of WRKTIM etc)
   FREQ     R    Frequency of observation (Hz)
Output:
   WRKTIM   R(*) Large work array.
   WORK1    R(*) Large work array, same size as WRKTIM
   WORK2    R(*) Large work array, same size as WRKTIM
   WORK3    R(*) Large work array, same size as WRKTIM
   WORK4    R(*) Large work array, same size as WRKTIM
   WORK5    R(*) Large work array, same size as WRKTIM
   IRET     I    Return code 0=OK, else failed.
```

16.14.8 CGASET

Gets next set of calibration data in CURCAL, does linear interpolation in time between time entries in CALTAB. Calls BLSET to fill or initialize BLFAC and apply and baseline dependent calibration then enters any rate corrections. If the preceeding or following entry is for the current source then only entries for that source is used.

```
CGASET (TIME, IERR)
Inputs:
   TIME    R        Current time (of data) in days.
Inputs from DSEL.INC:
   CALTAB  R(*,2)   Cal. table from gain table file
                    Values in order:
                    By antenna (NUMANT)
                       By IF (NUMIF)
```

```
                        By Polarization (NUMPOL)
                            Real part, imaginary part,
                            group delay, phase rate, ref. ant.
      LCLTAB  I       Number of values in CALTAB per entry (5)
      IFRTAB  R(*,2)  Ionospheric Faraday rotation from cal table,
                      listed by antenna number
      CALTIM  R(3)    Time of two cal. entries; third value is
                      time of current values.
      CIDSOU  I(2)    Previous/next source ID number using ICALPn as
                      a pointer.
      CURSOU  I       Current source ID number.
      ICALP1  I       Pointer in CALTAB, CALTIM to previous time.
      ICALP2  I       Pointer in CALTAB, CALTIM to next time.
      IBLP1   I       Pointer in BLTAB, if < 0 then BLFAC needs
                      to be initialized by BLSET.
      DOCAL   L       If true then apply antenna based calibation.
      DOBL    L       If true then apply baseline based calibation.
      RATFAC  R(*)    IF scaling factor to convert s/s to rad/day
      DELFAC  R(*)    IF scaling factor to convert s to rad/channel
      DXTIME  R       Integration time of the data in days.
   Output:
      IERR    I       Return error code, 0=>OK else error.
   Output to DSEL.INC:
      LCALTM  R       Time of current calibration.  If gain file
                      is exhausted then 1.0E20 is returned.
      CURCAL  R(*)    Current calibration information.
                      Values in order:
                      By antenna (NUMANT)
                         By IF (EIF-BIF+1)
                            By Polarization (NUMPOL)
                                Real part, imaginary part,
                                cos(delta), sin(delta), rate
                      Where delta is the phase change between
                      channels and rate is the fringe rate in
                      radians/day
      IFR     R(*)    Current ionospheric rotation measure for each
                      antenna
      BLFAC   R(*)    Baseline dependent factors.
                      Initialized to 1/GMMOD,0 if DOBL=.FALSE.
                      Indexing scheme: an entry defined by ant1<ant2
                      starts in element:
         lentry * (((ant1-1)*numant-((ant1+1)*ant1)/2 + ant2) - 1) + 1
                      where lentry = 2 * NUMPOL * (EIF-BIF+1)
                      An entry contains the values in order:
                      By IF (NUMIF)
                         By Polarization (NUMPOL)
                             Real part, imaginary part.
```

16.14.9 CLREFM

Routine to change the format of the CL table from one containing 39/24 columns to the 41/26 columns needed by the addition of the FREQID and IFR columns. NOTE: routine uses LUN 45 as a temporary logical unit number.

```
CLREFM (DISK, CNO, VER, CATBLK, LUN, IRET)
Inputs:
   DISK     I        Volume number
   CNO      I        Catalogue number
   VER      I        Version to check/modify
   CATBLK   I(256)   Catalog header
   LUN      I        LUN to use
Output:
   IRET     I        Error, 0 => OK

Note, routine will leave no trace of its operation, i.e. CL table
will be closed on output and will have same number as one specified.
Difference will be only that number of columns has changed if that
is required.
```

16.14.10 CLUPDA

Concatanates all SN tables and rereferences to the same reference antenna. Then if SNSMTH is true the SN table will be smoothed. If DOAPPL (in common) is true then the SN table is applied to the specified CL table. Leaves the output table sorted in time-antenna order.

```
CLUPDA (SINGLE, SNSMTH, REFA, IERR)
Inputs:
  SINGLE    L    If true then the uv data is a single source file
                 and only SN tables will be processed.
  SNSMTH    L    If true then smooth SN tables.
  REFA      I    The desired reference antenna, 0=most used.
Inputs from DSEL.INC
   DOAPPL      L    If true then apply SN tables to the CL table.
   CLVER       I    Input Cal (CL) file version number.
                    For single source files the input SN table.
   CLUSE       I    Cal file version number to put smoothed gains
                    into and use for calibration.
                    For single source files the output SN table.
   TSTART      R    First time to process (days) (no default)
   TEND        R    Last time to process (days) (no default)
Output:
   IERR        I    Return code, 0=>OK, otherwise failed.
Useage notes:
   1) Uses CLBUFF, BLFAC, CALTAB, BLTAB and UBUFF from /DSEL.INC/.
   2) Sorts the relevant tables.
```

16.14.11 CSLGET

Sets up for interpolation in cal (CL or SN) table, reads values from cal table. Assumes only valid, selected data in open cal table.

```
CSLGET (TIME, IERR)
Inputs:
   TIME    R         Current data time.
Inputs from DSEL.INC:
   CLBUFF  I(*)      Cal table I/O TABIO buffer
   ICLRNO  I         Current cal record number
   NCLINR  I         Number of cal records in file.
```

```
   NUMANT  I         Number of antennas
   NUMPOL  I         Number of IFs per group (polarizations)
   NUMIF   I         Number of IFs.
   GMMOD   R         Mean cal modulus
Output:
   IERR    I         Return error code 0=>OK, else failed.
Output to DSEL.INC
   CALTAB  R(*,2)    Cal. table from cal table file
                     Values in order:
                     By antenna (NUMANT)
                        By IF (NUMIF)
                           By Polarization (NUMPOL)
                               Real part, imaginary part,
                               group delay, phase rate, ref. ant.
   LCLTAB  I         Number of values in CALTAB per entry (5)
   IFRTAB  R(*, 2)   Ionospheric Faraday rotation measure from
                     cal table. Values listed by antenna.
   CALTIM  R(3)      Time of two cal. entries; third value is
                     time of current values.
   CIDSOU  I(2)      Previous/next source ID number using ICALPn as
                     a pointer.
   ICALP1  I         Pointer in CALTAB, CALTIM to previous time.
   ICALP2  I         Pointer in CALTAB, CALTIM to next time.
```

16.14.12 DATBND

Routine which applies the bandpass correction.

```
DATBND (TIME, IA1, IA2, VIS, IERR)
Inputs:
   TIME     R      Time of visibility data (in days)
   IA1      I      Antenna number 1
   IA2      I      Antenna number 2
   VIS      R(*)   Array of visibility data
Inputs from common:
   PBUFF    R(*)   Large array containing bandpass spectra for
                   several antennas
   ANTPNT   I(2)   Pointer giving the start address of the specified
                   antennas bandpass spectra within PBUFF
Outputs:
   VIS      R(*)   Array of corrected visibility data
   IERR     I      If = 0, all OK,
                   If > 0, error returned from BPGET
Output to common:
   CNTREC  I(2,3)  Record counts:
                     (1&2,1) Previously flagged (partly, fully)
                     (1&2,2) Flagged due to gains (part, full)
                     (1&2,3) Good selected (part, full)

NOTE: This routine applies the bandpass correction for formulae:

(1) Cross-power:   Scorr   =         Sobs
                               --------------------
                              SQRT (Sant_1 * Sant_2)
```

```
 (2) Total-power:   Scorr    =   (Son / Soff) - 1.0
```

16.14.13 DATCAL

Applies calibration to data.

```
   DATCAL (IA1, IA2, TIME, VIS, DROP, IERR)
   Inputs:
      IA1    I      First antenna number
      IA2    I      Second antenna number
      TIME   R      Time of record (days)
      VIS    R(*,*) Input visibility array (not yet converted to
                    output form.
   Inputs from DSEL.INC:
      DOCAL   L     If true do antenna calibration.
      DOBL    L     If true do baseline calibration.
      DOWTCL  L     If true calibrate weights.
      CURCAL  R(*)  Current calibration information.
                     Values in order:
                     By antenna (NUMANT)
                        By IF (EIF-BIF+1)
                           By Polarization (NUMPOL)
                               Real part, imaginary part,
                               cos(delta), sin(delta), rate
                     Where delta is the phase change between
                     channels and rate is the fringe rate in
                     radians/day
      LCUCAL   I     Number of values in CURCAL per entry (5)
      POLOFF   I(4,2)Offsets from the beginning of an IF entry in
                     CURCAL for a given polarization.  The first
                     dimension is the polarization pixel number and
                     the second is the antenna number of a baseline
                     (e.g. first or second = 1 or 2).
      CALTIM   R(3)  Time of two cal. entries; third value is
                     time of current values.
      LCALTM    R    Time of current calibration.
      BLFAC     R(*) Baseline dependent factors including GMMOD.
                     Indexing scheme: an entry defined by ant1<ant2
                     starts in element:
         lentry * (((ant1-1)*numant-((ant1+1)*ant1)/2 + ant2) - 1) + 1
                       where lentry = 2 * NUMPOL * (EIF-BIF+1)
                       An entry contains the values in order:
                       By IF (NUMIF)
                          By Polarization (NUMPOL)
                              Real part, imaginary part.
                        Applied only to cross corelation data.
     Output:
      DROP    L      True if data all flagged.
      IERR    I      Return code, 0=OK, else CGASET error number.
     Output to common:
      CNTREC  I(2,3) Record counts:
                     (1&2,1) Previously flagged (partly, fully)
                     (1&2,2) Flagged due to gains (part, full)
```

```
                    (1&2,3) Good selected (part, full)
```

16.14.14 DATFLG

Flags data specified in flagging table

```
DATFLG (RPARM, VIS, DROP, IERR)
Inputs:
   RPARM(*)   R    Random parameter array
   VIS(3,*)   R    Visibility array
Inputs from common /CFMINF/:
   CURSOU     I    Current source number
   NUMFLG     I    Number of flagging entries.
   TMFLST     R    Time of last visibility for which flagging
                   was checked.
   FLGSOU(*)  I    Source id numbers to flag, 0=all.
   FLGANT(*)  I    Antenna numbers to flag, 0=all.
   FLGBAS(*)  I    Baseline (A1*256+A2) numbers to flag, 0=all.
   FLGSUB(*)  I    Subarray numbers to flag, 0=all.
                   Following should have defaults filled in.
   FLGBIF(*)  I    First IF to flag.
   FLGEIF(*)  I    Highest IF to flag.
   FLGBCH(*)  I    First channel to flag.
   FLGECH(*)  I    Highest channel to flag.
   FLGPOL(4,*)L    Flags for the polarizations, should correspond
                   to selected polarization types.
Output:
   RPARM(*)   R    Random parameter array
   VIS(3,*)   R    Visibility array
   DROP       L    True if data all flagged.
   IERR       I    Return code, 0=OK, else NXTFLG error number.
```

16.14.15 DATGET

Reads next selected data record. Applies calibration and editing.

```
DATGET (RPARM, VIS, TIMLST, IERR)
Inputs from DSEL.INC:
   INXRNO       I    Current INDEX file record number.
                     If .LT. 0 then there is no index file.
   NINDEX       I    Number of entries in the index table
   NXKOLS(7)    I    Pointer array for index records.
   ANTENS(50)   I    List of antennas selected, 0=>all,
                     any negative => all except those specified
   NANTSL       I    Number of antennas selected/excluded in ANTENS
                     0 = All included.
   DOAWNT       L    If .TRUE. then antennas in ANTENS included.
                     If .FALSE. then excluded.
   SUBARR       I    Subarray number desired, 0=>any.
   NSOUWD       I    Number of sources specified.
   DOSWNT       L    If true sources specified are included
                     else excluded.
   SOUWAN(30)   I    List or source numbers from source file.
   DOACOR       L    If true, pass autocorrelations.
```

```
    DOXCOR       L    If true, pass crosscorrelations.
    TRANSL       L    If true, translate data to requested stokes.
    JADR(2,*)    I    Table to translate input data to output vis.
    TSTART       R    Start time in days.
    TEND         R    End time in days.
    UVRA(2)      R    UV range (wavelength squared)
    NPRMIN       I    No. random parameters in the input data
    NRPARM       I    No. random parameters in output data.
    LRECIN       I    Length of input record in words.
    KLOCSU       I    Source number pointer in input data.
    KLOCWT       I    Weight pointer for compressed data.
    NDECMP       I    Number of entries in DECMP
    DECMP    I(2,*)   (1,*) = number of packed correlator values
                      (2,*) = 0-rel offset in vis data.
                      (from beginning of vis data NOT ran. parms.)
    DOFQSL       L    If true, FREQSEL random parameter is present
    FRQSEL       I    FQ entry to pass.
  Input/output:
    TIMLST       R    Time of last record
  Output:
    RPARM(*)   R    Random parameter array
    VIS(3,*)   R    Visibility array
    IERR       I    Return code, 0=OK, else UVDISK error number.
 Output to common /DSEL.INC/:
    FSTVIS       I    First word pointer of current buffer.
    LSTVIS       I    Last word pointer of current buffer.
    CURSOU       I    Current source number.
```

16.14.16 DGGET

Gets requested data from visibility record, reformatting if needed. REQUIRES setup by DGINIT to set values of MVIS, JADR, SELFAC and ALLWT.

```
  DGGET (VISIN, IND, MVIS, JADR, SELFAC, ALLWT, VISOUT, DROP)
  Inputs:
    VISIN   R(IND,*)   Input visibility array
    IND     I          First dimension of VISIN (CATBLK(KINAX))
    MVIS    I          Number of visibilities in requested output
                       format.
    JADR    I(2,*)     Pointers to the first and second visibility
                       input records to be used in the output record.
                       If JADR(1,n) is negative use IABS (JADR(1,n))
                       and multiply the visibility by i (=SQRT(-1))
    SELFAC  R(2,*)     Factors to be multiplied by the first and
                       second input vis's to make the output vis.
    ALLWT   L          Flag, = .TRUE. if all visibilities must have
                       positive weight.
   Output:
    VISOUT  R(3,*)     Output visibility record
    DROP    L         .TRUE. if all data in record flaged.
```

16.14.17 DGHEAD

Corrects CATBLK in common /MAPHDR/ to correspond to UVGET output data. If only one output source is specified then the information about that source is filled in from the source file (if any).

```
DGHEAD
Inputs from include DSEL.INC:
   BCHAN   I       First channel desired.
   ECHAN   I       Last channel desired.
   BIF     I       First IF desired.
   EIF     I       Last IF desired.
   PMODE   I       Polarizarion mode (see DGINIT for codes)
                   0 => same Stokes' as in input.
   NSOUWD  I       Number of sources specified.
   DOSWNT  L       If true sources specified are included
                   else excluded.
   SOUWAN  I(30)   List or source numbers from source file.
Input/Output in common /MAPHDR/:
   CATBLK  I(256)  Uvdata catalog header record.
```

16.14.18 DGINIT

Sets up tables for selecting data from vis. record. Checks if requested data in data base. Requires catalog header record in common /MAPHDR/ and setup of common /UVHDR/ by UVPGET before call. Note: STOKES='HALF' will work if only partial information (i.e. 1 polarization) is available in the data.

```
DGINIT (STOKES, BCHAN, ECHAN, BIF, EIF, MVIS, JADR, SELFAC, ALLWT,
  *   PMODE, IERR)
Inputs:
   STOKES   C*4      Desired output data format: 'I','V','Q','U',
                     'IQU','IQUV','IV','RR','LL','RL','LR'
                     'HALF' (=parallel pol.), 'FULL' (=RR,LL,RL,LR)
   BCHAN    I        First channel desired.
   ECHAN    I        Last channel desired.
   BIF      I        First IF desired.
   EIF      I        Last IF desired.
Input from common /MAPHDR/
   CATBLK   I(256)   Catalog header record.
Output:
   MVIS     I        Number of visibilities in requested output
                     format.
   JADR     I(2,*)   Pointers to the first and second visibility
                     input records to be used in the output record.
                     If JADR(1,n) is negative use IABS (JADR(1,n))
                     and multiply the visibility by i (=SQRT(-1))
   SELFAC   R(2,*)   Factors to be multiplied by the first and
                     second input vis's to make the output vis.
   ALLWT    L        Flag, = .TRUE. if all visibilities must have
                     positive weight.
   PMODE    I        Polarization mode:
                         1 = I,    2 = V,   3 = Q
                         4 = U,    5 = IQU, 6 = IQUV
                         7 = IV,   8 = RR,  9 = LL
                        10 = RL,  11 = LR, 12 = parallel (RR,LL)
```

```
                          13 = (RR,LL,RL,LR)
      IERR     I        Error flag. 0 => ok, 1 = unrecognized stokes,
                          2 = data unavailable.
```

16.14.19 GACSIN

Single dish calibration routine - Initializes CS file, and prepares table to be applied. If there is no CS file DOCAL is set to .FALSE.

```
   GACSIN (IERR)
   Inputs from DSEL.INC:
      CLUSE    I   CS file version number to initialize.
   Output:
      IERR     I   Return code, 0 => ok, otherwise CS table exists but
                      cannot be read.
   Output to DSEL.INC:
      ICLRNO   I   Current CS record number
      NCLINR   I   Number of gain records in file.
      NUMPOL   I   Number of polarizations
      NUMIF    I   Number of IFs.
```

16.14.20 GAININ

Initializes Cal file, and prepares gain table to be applied. If there is no CL file DOCAL is set to .FALSE. For single source data files an SN table will be used rather than a CL table. Opens gain (CL or SN) and baseline (BL) tables if necessary.

```
   GAININ (IERR)
   Inputs from common /DSEL.INC/
      CLUSE        I    Cal file version number (CL or SN) to
                        initialize.
   Output:
      IERR         I    Return code, 0=>OK, otherwise CL table
                        exists but cannot be read.
   Output to common /DSEL.INC/:
      RATFAC(*)    R    IF scaling factor to convert s/s to rad/day
      DELFAC(*)    R    IF scaling factor to convert s to rad/channel
      LAMSQ(*, *)  R    Table of wavelength squared (in meters squared)
                        for each channel (first axis) and IF (second
                        axis)
      ICLRNO       I    Current cal record number
      NCLINR       I    Number of gain records in file.
      NBLINR       I    Number of BL records in file.
      NUMANT       I    Number of antennas
      NUMPOL       I    Number of polarizations
      NUMIF        I    Number of IFs.
      GMMOD        R    Mean gain modulus
```

16.14.21 LXYPOL

Fills polarization correction table from info in AN table for Linear polarization feeds (XY).

```
   LXYPOL (PANGLE, IERR)
   Inputs:
```

```
   PANGLE  R(*)      Parallactic angles of the antennas (Rad)
Output:
   IERR    I         Return error code, 0=>OK else error.
                     1=table too small, 2=multiple subarrays,
                     10 = unknown polarization parameterization,
                     otherwise GETANT error.
Output to common /DSEL.INC/:
   PARTIM  R         Time of current parallactic angles. (-1.0E10)
   PARSOU  I         Source ID for current parallactic angles. (-10)
   POLCAL  R(2,*)    Polarization correction
                     Values in order:
                     By baseline
                        By IF (EIF-BIF+1)
                           A 4x4 complex matrix to be multiplied by
                               the observed polarization vector
                               (RR,LL,RL,LR) to produce the
                               corrected data.
                    Indexing scheme: an entry defined by ant1<ant2
                    starts in element:
      (((ant1-1)*numant-((ant1+1)*ant1)/2 + ant2) - 1) + 1
```

16.14.22 NXTFLG

Updates flagging tables in common fron an FG table.

```
NXTFLG (TIME, IERR)
Inputs:
   TIME         R    Current time (days) for flag entries
Inputs from common /CFMINF/(INCLUDEs /DSEL.INC):
   NUMFLG       I    number of current FLAG entries.
   FGKOLS(8)    I    The column pointer array in order, SOURCE,
                     SUBARRAY, ANTS, TIMERANG, IFS, CHANS, PFLAGS,
                     REASON
   FGNUMV(8)    I    Element count for each column
   IFGRNO       I    Current FLAG file record.
Output to common /DSEL.INC/:
   NUMFLG     I    Number of flagging entries.
   TMFLST     R    Time of last visibility for which flagging
                   was checked.
   FLGSOU(*)  I    Source id numbers to flag, 0=all.
   FLGANT(*)  I    Antenna numbers to flag, 0=all.
   FLGBAS(*)  I    Baseline (A1*256+A2) numbers to flag, 0=all.
   FLGSUB(*)  I    Subarray numbers to flag, 0=all.
                   Following should have defaults filled in.
   FLGBIF(*)  I    First IF to flag.
   FLGEIF(*)  I    Highest IF to flag.
   FLGBCH(*)  I    First channel to flag.
   FLGECH(*)  I    Highest channel to flag.
   FLGPOL(4,*)L    Flags for the polarizations, should correspond
                   to selected polarization types.
   FLGTND(*)  R    End time of flag.
Output:
   IERR       I    Return code, 0=OK, else TABIO error number.
```

16.14.23 POLSET

Fills polarization correction table from info in AN table.

```
POLSET (IERR)
Inputs from common:
   STNEPL   R(2,*)   Feed real/elipticity (poln, IF)
   STNORI   R(2,*)   Feed imag/orientation (poln, IF)
   STNPST   C*8      Feed solution type:
                        'APPROX  ' => linear approximation
                        'ORI-ELP ' => orientation-ellipticity
                        'X-Y LIN ' => lin. approx. for lin.
                           polarized (X-Y) data.
  Output:
   IERR    I         Return error code, 0=>OK else error.
                     1=table too small, 2=multiple subarrays,
                     10 = unknown polarization parameterization,
                     otherwise GETANT error.
  Output to DSEL.INC:
   PARTIM  R         Time of current parallactic angles. (-1.0E10)
   PARSOU  I         Source ID for current parallactic angles. (-10)
   POLCAL  R(2,*)    Polarization correction
                     Values in order:
                     By baseline
                        By IF (EIF-BIF+1)
                           A 4x4 complex matrix to be multiplied by
                               the observed polarization vector
                               (RR,LL,RL,LR) to produce the
                               corrected data.
                    Indexing scheme: an entry defined by ant1<ant2
                    starts in element:
      (((ant1-1)*numant-((ant1+1)*ant1)/2 + ant2) - 1) + 1
```

16.14.24 SCLOAD

Copies part of a 'BP' scratch file to a second scratch file for more efficient I/O.

```
SCLOAD (TIME1, TIME2, LUNOP, FINDOP, CREATE, IERR)
Input:
   TIME1       R    Time label of first section of data to be
                    transferred.
   TIME2       R    Time label of second section of data to be
                    transferred. If < 0 is ignored.
   LUNOP       I    LUN of secondary scratch file.
   CREATE      L    If true must create new scratch file.
Output:
   FINDOP      I    FTAB pointer for secondary scratch file.
   IERR        I    Return error code 0=>OK, else failed.
```

16.14.25 SCINTP

When the interpolation mode of bandpass calibration is specified this routine takes the secondary scratch file created by SCLOAD and interpolates in time between the two entries and writes a scratch file containing the interpolated data. To save on I/O and interpolation time the third scratch file is updated at 0.2 of the interval between the two entries.

```
SCINTP (TIME, LUNIN, FINDIN, LUNOP, FINDOP, IERR)
Input:
   TIME     R   Current time of data (days)
   LUNIN    I   LUN of secondary scratch file.
   FINDIN   I   FTAB pointer for second scratch file.
   LUNOP    I   LUN of third (interpolated) scratch file.
Output:
   IERR     I   Return error code 0=>OK, else failed.
```

16.14.26 SDCGET

Single dish calibration routine. Sets up for interpolation in cal (CS) table, reads values from cal table. Assumes only valid, selected data in open cal table. Uses calls to TABIO directly for efficiency.

```
SDCGET (TIME, IERR)
Inputs:
   TIME         R    Current data time.
Inputs from common /DSEL.INC/:
   CLBUFF(*)    I    Cal table I/O TABIO buffer
   ICLRNO       I    Current cal record number
   NCLINR       I    Number of cal records in file.
   NUMANT       I    Number of beams
   NUMPOL       I    Number of IFs per group (polarizations)
   NUMIF        I    Number of IFs.
Output:
   IERR         I    Return error code 0=>OK, else failed.
Output to common /DSEL.INC/:
   CALTAB(*,2)  R    Cal. table from cal table file
                     Values in order:
                     By beam (NUMANT)
                        By IF (NUMIF)
                           By Polarization (NUMPOL)
                               Amplitude factor,
                               offset (before factor)
                               RA correction
                               Dec correction.
   LCLTAB       I   Number of values in CALTAB per entry (4)
   CALTIM(3)    R    Time of two cal. entries; third value is
                     time of current values.
   ICALP1       I    Pointer in CALTAB, CALTIM to previous time.
   ICALP2       I    Pointer in CALTAB, CALTIM to next time.
```

16.14.27 SDCSET

Single dish calibration routine: Gets next set of calibration data in CURCAL, does linear interpolation in time between time entries in CALTAB.

```
SDCSET (TIME, IERR)
Inputs:
   TIME     R        Current time (of data) in days.
Inputs from common /DSEL.INC/:
   CALTAB   R(*,2)   Cal. table from gain table file
                     Values in order:
                     By beam (NUMANT)
```

```
                        By IF (NUMIF)
                           By Polarization (NUMPOL)
                               Amplitude factor,
                               offset (before factor)
                               RA correction
                               Dec correction.
   LCLTAB   I        Number of values in CALTAB per entry (4)
   CALTIM   R(3)     Time of two cal. entries; third value is
                     time of current values.
   ICALP1   I        Pointer in CALTAB, CALTIM to previous time.
   ICALP2   I        Pointer in CALTAB, CALTIM to next time.
Output:
   IERR     I        Return error code, 0=>OK else error.
Output to common /DSEL.INC/:
   LCALTM   R        Time of current calibration.  If gain file
                     is exhausted then 1.0E20 is returned.
   CURCAL   R(*)     Current calibration information.
                     Values in order:
                     By beam (NUMANT)
                        By IF (EIF-BIF+1)
                           By Polarization (NUMPOL)
                               Amplitude factor,
                               offset (before factor)
                               RA correction
                               Dec correction.
```

16.14.28 SDGET

Subroutine to obtain data from a single dish data base with optional application of flaging and/or calibration and/or pointing information. Reads data with a large variety of selection criteria and will reformat the data as necessary. Does many of the startup operations, finds Single dish uv like data file etc., reads CATBLK and updates the /UVHDR/ common to reflect the output rather than input data.

```
SDGET (OPCODE, RPARM, VIS, IERR)
Input:
   OPCODE   C*4      Opcode -
                     'INIT' => Open files Initialize I/O.
                     'READ' => Read next specified record.
                     'CLOS' => Close files.
Inputs via DSEL.INC  (Include DSEL.INC)
   UNAME    C*12     AIPS name of input file.
   UCLAS    C*6      AIPS class of input file.
   UDISK    R        AIPS disk of input file.
   USEQ     R        AIPS sequence of input file.
   SOURCS   C(30)*16 Names of up to 30 sources, '*' => all
                     First character of name '-' => all except those
                     specified.
   TIMRNG   R(8)     Start day, hour, min, sec, end day, hour,
                     min, sec.   0's => all.
   UVRA     R(2)     Range of RA (1) and dec (2) in degrees about
                     the value in CATBLK at time of READ call to
                     SDGET.  0=>all.
   STOKES   C*4      Stokes types wanted.
                     'I','Q','U','V','R','L','IQU','IQUV'
```

```
                     '    '=> Leave data in same form as in input.
   BCHAN    I        First channel number selected, 1 rel. to first
                     channel in data base.    0 => all
   ECHAN    I        Last channel selected.   0 => all
   BIF      I        First IF number selected, 1 rel. to first
                     IF in data base.    0 => all
   EIF      I        Last IF selected.   0 => all
   DOCAL    L        If true apply calibration, else not.
   SUBARR   I        Subarray desired, 0 => all
   FGVER    I        FLAG file version number, if < 0 then
                     NO flagging is applied. 0 => use highest
                     numbered table.
   CLUSE    I        Cal (CS) file version number to apply.
Output:
   RPARM    R(*)     Random parameter array of datum.
   VIS      R(3,*)   Regular portion of data array.
   IERR     I        Error code: 0 => OK,
                        -1 => end of data
                        >0 => failed, abort process.
Output in common /DSEL.INC/: The default values will be filled in
if null values were specified.
   CATBLK   I(256)   Catalog header block, describes the output
                     data rather than input.
   NPRMIN   I        Number or random parameters in the input data.
   TRANSL   L        If true translate data to requested Stokes'
   CNTREC   I(2,3)   Record counts:
                     (1&2,1) Previously flagged (partly, fully)
                     (1&2,2) Flagged due to gains (part, full)
                     (1&2,3) Good selected (part, full)
Usage notes:
 1) Include DSEL.INC should be declared in the main
    program or at a level that they will not be overlaid while
    SDGET is in use (ie. between the 'INIT' and 'CLOS' calls)
 2) If no sorting is done SDGET uses AIPS luns 25, 28, 29 and 30
   (1 map, 3 non map files).  If sorting is done (usually possible)
   then 8 map and 3 non map files are used (mostly on OPCODE='INIT')
   and LUNs 16,17,18,19,20,21,22,23,24,25, 28,29,30.
 3) OPCODE = 'INIT' does the following:
   - The catalgue data file is located and the catalog header
     record is read.
   - The index file (if any) is initialized.
   - The flag file (if any) is initialized and sorted if necessary
     (Must be in time order).
   - The CS table (if any) is initialized.
   - I/O to the input file is initialized.
         The following LUNs may be used but will be closed on
     return: 16, 17, 18, 19, 20, 21, 22, 23, 24
         The following LUNs may be used but will be open on
     return: 25 (uv data), 28 (NX table), 29 (CS table),
             30 (FG table).
         NO data are returned from this call.
 4) OPCODE = 'READ' reads one  record properly selected,
    transformed (e.g. I pol.), calibrated and edited as requested
    in the call with OPCODE = 'INIT'
```

```
 5) OPCODE = 'CLOS' closes all files used by SDGET which are still
    open.  No data are returned.
 6) If DOCAL is true then the common array CNTREC will contain the
    counts of records which are good or fully or partly flagged
    both previously and due to flagged gain solutions.
```

16.14.29 SETSM

SETSM determines the type of spectral smoothing to be applied and sets up the look up table to do it. The actual smoothing is done in routine SMOSP

```
SETSM (IRET)
Inputs: (via common)
   SMOOTH  R(3)   Array containing smoothing parms
                  SMOOTH(1) = type of function
                        (2) = width of function
                              in channels
                        (3) = support of function
                              in channels
                  Type of function supported are:
                         0 => no smoothing
                         1 => hanning
                         2 => gaussian
                         3 => boxcar
                         4 => sin(x)/x
Output:
  IRET      I     Return error code, 0=>OK, otherwise abort.
```

16.14.30 SELINI

Subroutine to initialize the control values for UVGET in commons in DSEL.INC.

```
SELINI
Outputs via DSEL.INC  (Include DSEL.INC):
   UNAME    C*12      AIPS name of input file. (blank)
   UCLAS    C*6       AIPS class of input file. (blank)
   UDISK    R         AIPS disk of input file. (0.0)
   USEQ     R         AIPS sequence of input file. (0.0)
   SOURCS   C(30)*16  Names of up to 30 sources. (blank)
   SELQUA   I         Qualifier wanted (-1 => all)
   SELCOD   C*4       Cal code ('    ')
   TIMRNG   R(8)      Timerange (0s => all)
   UVRNG    R(2)      Baseline range (0s => all)
   STOKES   C*4       Stokes types wanted. (blank)
   BCHAN    I         First channel number selected, (1)
   ECHAN    I         Last channel selected. (0=>all)
   BIF      I         First IF number selected. (1)
   EIF      I         Last IF selected. (0=>all)
   DOCAL    L         If true apply calibration. (false)
   DOPOL    L         If true then correct polarization (false)
   DOACOR   L         True if autocorrelations wanted (false)
   DOXCOR   L         True if cross-correlations wanted (true)
   DOWTCL   L         True if weight calibration wanted. (false)
   DOFQSL   L         True if FREQSEL random parm present (false)
```

```
   FRQSEL   I         Default FQ table entry to select (-1)
   SELBAN   R         Bandwidth (Hz) to select (-1.0)
   SELFRQ   D         Frequency (Hz) to select (-1.0)
   DOBAND   I         >0 if bandpass calibration. (-1)
   BPNAME   C*48      Name of scratch file set up for BP's.
   DOSMTH   L         True if smoothing requested. (false)
   SMOOTH   R(3)      Smoothing parameters (0.0s)
   DXTIME   R         Integration time (days). (1 sec)
   ANTENS   I(50)     List of antennas selected. (0=>all)
   SUBARR   I         Subarray desired. (0=>all)
   FGVER    I         FLAG file version number. (0)
   CLUSE    I         Cal (CL or SN) file version number (0)
   BLVER    I         BL Table to apply (-1)
   BPVER    I         BP table to apply (-1)
```

16.14.31 SNREFM

Routine to change the format of the SN table from one containing 18/12 columns to the 20/14 columns needed by the addition of the FREQID and IFR columns. NOTE: routine uses LUN 45 as a temporary logical unit number.

```
SNREFM (DISK, CNO, VER, CATBLK, LUN, IRET)
Inputs:
   DISK     I       Volume number
   CNO      I       Catalogue number
   VER      I       Version to check/modify
   CATBLK   I(256)  Catalogue header
   LUN      I       LUN to use
Output:
   IRET     I       Error, 0 => OK

Note, routine will leave no trace of its operation, i.e. SN table
will be closed on output and will have same number as one specified.
Difference will be only that number of columns has changed if that
is required.
```

16.14.32 SOUFIL

Fills in arrays of source numbers to be included or excluded; also checks antennas to be selected.

```
SOUFIL (IERR)
Inputs from include DSEL.INC:
   SOURCS(30)   C*16 Names of up to 30 sources, *=>all
                     First character of name '-' => all except those
                     specified.
   CALSOU(30)   C*16 Names of up to 30 calibrators,
                     '*' or blank =>all, first character of name '-'
                     => all except those specified.
   SELQUA       I    Source qualifiers to be selected, -1=>any.
                     Applied to both SOURCS and CALSOU.
   SELCOD       C*4  Calibrator codes to select.
                     '    ' => any,
                     '*   ' => any non blank calibrator code.
                     '-CAL' => blank only (no calibrators)
```

```
                       anything else => matching CALcodes.
                       Applied to SOURCS or CALSOU as controlled by
                       DOAPPL
   DOAPPL      L       If true then selection of the sources in
                       CALSOU is conditioned on SELCOD else
                       selection of SOURCS is conditioned on SELCOD.
   ANTENS(50)  I       List of antennas selected, 0=>all,
                       any negative => all except those specified
Output:
   IERR        I       Return code, 0=>OK, otherwise source file
                       exists but cannot be read.
                       1=TABIO problem, 2=no sources or calibrators
Output to DSEL.INC:
   NSOUWD      I       Number of sources included or excluded; if
                       0 all sources are included.
   DOSWNT      L       If .TRUE. then sources in SOUWAN are included
                       If .FALSE. then excluded.
   SOUWAN(30)  I       The source numbers of sources included or
                       excluded.
   SOUWTB(30)  I       The SoUrce table row numbers corresponding
                       to SOUWAN.
   NCALWD      I       Number of calibrators included or excluded.
   DOCWNT      L       If .TRUE. then calibrators in CALWAN are
                       included, if .FALSE. then excluded.
   CALWAN(30)  I       The source numbers of calibrators included or
                       excluded.
   CALWTB(30)  I       The SoUrce table row numbers corresponding
                       to CALWAN.
   NANTSL      I       Number of antennas selected/excluded in ANTENS
                       0 = All included.
   DOAWNT      L       If .TRUE. then antennas in ANTENS included.
                       If .FALSE. then excluded.
Note: also uses FGBUFF and UBUFF from /SELCAL/
```

16.14.33 UVGET

Subroutine to obtain data from a data base with optional application of flaging and/or calibration information. Reads data with a large variety of selection criteria and will reformat the data as necessary. Does many of the startup operations, finds uv data file etc, reads CATBLK and updates the DUVH.INC commons to reflect the output rather than input data. Most of the input to UVGET is through the commons in DSEL.INC; the initial (default) values of these may be set using routine SELINI.

```
UVGET (OPCODE, RPARM, VIS, IERR)
Input:
   OPCODE   C*4       Opcode:
                      'INIT' => Open files Initialize I/O.
                      'READ' => Read next specified record.
                      'CLOS' => Close files.
Inputs via DSEL.INC  (Include DSEL.INC)
   UNAME    C*12      AIPS name of input file.
   UCLAS    C*6       AIPS class of input file.
   UDISK    R         AIPS disk of input file.
   USEQ     R         AIPS sequence of input file.
   SOURCS   C(30)*16  Names of up to 30 sources, *=>all
```

```
                  First character of name '-' => all except
                  those specified.
   TIMRNG   R(8)     Start day, hour, min, sec, end day, hour,
                  min, sec. 0's => all
   UVRNG    R(2)     Minimum and maximum baseline lengths in
                  1000's wavelengths. 0's => all
   STOKES   C*4      Stokes types wanted.
                  'I','Q','U','V','R','L','IQU','IQUV'
                  '    '=> Leave data in same form as in input.
   BCHAN    I        First channel number selected, 1 rel. to first
                  channel in data base. 0 => all
   ECHAN    I        Last channel selected. 0=>all
   BIF      I        First IF number selected, 1 rel. to first
                  IF in data base. 0 => all
   EIF      I        Last IF selected. 0=>all
   DOCAL    L        If true apply calibration, else not.
   DOPOL    L        If true then correct for feed polarization
                  based on antenna file info.
   DOSMTH   L        True if smoothing requested.
   DOACOR   L        True if autocorrelations are requested.
   DOWTCL   L        True if weight calibration wanted.
   DOFQSL   L        True if FREQSEL random parm present (false)
   FRQSEL   I        Default FQ table entry to select (-1)
   SELBAN   R        Bandwidth (Hz) to select (-1.0)
   SELFRQ   D        Frequency (Hz) to select (-1.0)
   DOBAND   I        >0 if bandpass calibration. (-1)
   BPNAME   C*48     Name of scratch file set up for BP's.
   DOSMTH   L        True if smoothing requested. (false)
   SMOOTH   R(3)     Smoothing parameters (0.0s)
   DXTIME   R        Integration time (days). Used when applying
                  delay corrections to correct for delay error.
   ANTENS   I(50)    List of antennas selected, 0=>all,
                  any negative => all except those specified
   SUBARR   I        Subarray desired, 0=>all
   FGVER    I        FLAG file version number, if < 0 then
                  NO flagging is applied. 0 => use highest
                  numbered table.
   CLUSE    I        Cal (CL or SN) file version number to apply.
   BLVER    I        BL Table to apply .le. 0 => none
   BPVER    I        BP table to apply .le. 0 => none
Output:
   RPARM    R(*)     Random parameter array of datum.
   VIS      R(3,*)   Regular portion of visibility data.
   IERR     I        Error code: 0 => OK,
                      -1 => end of data
                      >0 => failed, abort process.
Output in commons in DSEL.INC: The default values will be filled in
if null values were specified.
   UVFREQ   D        Frequency corresponding to u,v,w
   CATBLK   I(256)   Catalog header block, describes the output
                  data rather than input.
   NPRMIN   I        Number or random parameters in the input data.
   TRANSL   L        If true translate data to requested Stokes'
   CNTREC   I(2,3)   Record counts:
```

```
                      (1&2,1) Previously flagged (partly, fully)
                      (1&2,2) Flagged due to gains (part, full)
                      (1&2,3) Good selected (part, full)
   ISCMP    L         True if input data is compressed.
   KLOCSU   I         0-rel random parm. pointer for source in input
                      file.
   KLOCFQ   I         0-rel random parm. pointer for FQ id in input
                      file.
   KLOCIF   I         0-rel random parm. pointer for IF in input
                      file.
   KLOCFY   I         0-rel random parm. pointer for freq. in input
                      file.
   KLOCWT   I         0-rel random parm. pointer for weight in
                      input file.
   KLOCSC   I         0-rel random parm. pointer for scale in
                      input file.
Usage notes:
 1) Include DSEL.INC should be declared in the main program or at a
    level that they will not be overlaid while UVGET is in use (ie.
    between the 'INIT' and 'CLOS' calls). SELINI can be used to
    initialize the control variables in these commons.
 2) If no sorting is done UVGET uses AIPS luns 25, 28, 29 and 30
   (1 map, 3 non map files).  If sorting is done (usually possible)
   then 8 map and 3 non map files are used (mostly on OPCODE='INIT')
   and LUNs 16,17,18,19,20,21,22,23,24,25, 28,29,30,40,42,43,44,45.
 3) OPCODE = 'INIT' does the following:
   - The catalgue data file is located and the catalog header
     record is read.
   - The source file (if any) is read.
   - The index file (if any) is initialized.
   - The flag file (if any) is initialized and sorted if necessary
     (Must be in time order).
   - The gain table (if any) is initialized.
   - The bandpass table (if any) is initialized
   - The smoothing convolution table (if any) is initialized
   - I/O to the input file is initialized.
         The following LUNs may be used but will be closed on
     return: 16, 17, 18, 19, 20, 21, 22, 23, 24
         The following LUNs may be used but will be open on
     return: 25 (uv data), 28 (NX table), 29 (CL or SN table),
             30 (FG table), 40 (BL table), 41 (BP table).
         NO data are returned from this call.
 4) OPCODE = 'READ' reads one visibility record properly selected,
    transformed (e.g. I pol.), calibrated and edited as requested
    in the call with OPCODE = 'INIT'
 5) OPCODE = 'CLOS' closes all files used by UVGET which are still
    open.  No data are returned.
 6) If DOCAL is true then the common array CNTREC will contain the
    counts of records which are good or fully or partly flagged
    both previously and due to flagged gain solutions.
 7) Only one subarray can be calibrated at a time if DOPOL is true.
    This is because the polarization information for only one
    subarray is kept at a time.
```

16.14.34 VISCNT

Counts the number of visibility records for the sources selected with the given time range in DSEL.INC. The index (NX) file should already be open.

```
VISCNT (IERR)
Inputs from commons in DSEL.INC
   TSTART       R    Start time (days)
   TEND         R    End time (days)
   INXRNO       I    Current INDEX file record number.
                     If .LT. 0 then there is no index file.
   NINDEX       I    Number of entries in the index table
   NXBUFF(*)    I    TABIO buffer for INDEX table.
   NSOUWD       I    Number of sources specified.
   DOSWNT       L    If true sources specified are included
                     else excluded.
   SOUWAN(30)   I    List or source numbers from source file.
   SUBARR       I    Selected subarray, 0=>all
   FRQSEL       I    Selected freq id, 0=>all
Output:
   IERR         I    Return code, 0=>OK, 1=>no data, 2=>error.
Output to commons in DSEL.INC
   CATBLK(KIGCN)  I    Visibility count. (CATBLK)
```

THIS PAGE DELIBERATELY BLANK