Chapter 7

High Level Utility Routines

7.1 Overview

There are a number of high level AIPS utility routines which merit special attention. Many of these routines do complex, but common, operations on data or image files, such as gridding uv data or doing 2-D FFTs. Since many of the routines do a great deal of computation, most use the array processor.

Many of these routines make heavy use of commons or the values in catalog header records for control and internal communication. A number of these routines will create scratch and/or output files if necessary. Several general and somewhat overlapping categories of routines are discussed below.

7.2 File Specification

The routines described in this chapter use several methods to specify the input, output, and scratch files. For cataloged files the file is usually specified by a disk number and a catalog slot number. For scratch files an index in arrays SCRVOL and SCRCNO in the common from include DFIL.INC is passed. The indicated values from SCRVOL and SCRCNO are the disk and catalog slot numbers of the scratch files. These values are filled in by SCREAT when the files are created.

A common convention for the routines described in this chapter is that a disk and "catalog slot" number are passed as call arguments and if the disk number is zero and the "catalog slot" number is positive then the file is a scratch file and the "catalog slot" number is the index in SCRVOL and SCRCNO. Several of the routines in this chapter also allow optional creation of output and/or scratch files.

7.3 Data Calibration and Reformatting Routines

The variety of different uv data formats, especially different polarization types, allowed in AIPS uv data bases complicates handling of uv data. In addition, uncalibrated multi-source uv data files need to have calibration, editing and selection criteria applied. A pair of routines allows simplified read access to either single- or multi-source uv data files. A short description is given here and the details of the subroutine calls are given at the end of this chapter. These routines do not use the array processor.

- UVGET sets up, selects, reformats, calibrates, edits either single- or multi-source data files.
- CALCOP. After set up by UVGET, CALCOP can be used to process the entire selected contents of a file to another file.

7.4 Operations on Images

These operations are those performed on entire image files. A short description is given here and the details of the subroutine calls and interface COMMONs are given at the end of this chapter.

- DSKFFT is a disk-based, two dimensional FFT.
- GRDCOR normalizes and corrects an image for the gridding convolution used to grid the image. Used in conjunction with UVGRID and DSKFFT.
- APCONV convolves two images.

7.5 UV Model Calculations

A system of routines is available to compute the Fourier transform of a model, given as either CLEAN or Gaussian components or an image, at the u,v and w locations of the data in a uv data file and to either subtract the model values from the observed values or divide the model values into the observed values. These routines make heavy use of COMMONs. A short description is given here and the details of the subroutine calls and interface COMMONs are given at the end of this chapter.

- UVMDIV divides model visibilities derived from CLEAN or Gaussian components or images into a uv data set.
- UVMSUB subtracts model visibilities derived from CLEAN or Gaussian components or images from a uv data set.

7.6 Image Formation

Routine MAKMAP makes an image or a dirty beam given a uv data set. The data may be either calibrated or uncalibrated (raw) data and calibration and various selection criteria may be (optionally) applied. MAKMAP makes heavy use of COMMONs and the array processor. The details of the call sequence and interface COMMONs are given at the end of this chapter.

7.7 INCLUDEs

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.

7.7.1 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.
```

7.7.2 DFIL.INC

```
C                                                          Include DFIL.
C                                       AIPS system catalog and scratch
      INTEGER   NSCR, SCRVOL(128), SCRCNO(128), IBAD(10), LUNS(10),
     *   NCFILE, FVOL(128), FCNO(128), FRW(128), CCNO
      LOGICAL   RQUICK
      COMMON /CFILES/ RQUICK, NSCR, SCRVOL, SCRCNO, NCFILE, FVOL, FCNO,
     *   FRW, CCNO, IBAD, LUNS
C                                                          End DFIL.
```

7.7.3 DGDS.INC

```
C                                                          Include DGDS.
C                                       include for uv modeling
      INTEGER   SCRBLK(256), KLNBLK(256), MFIELD, FLDSZ(2,MAXFLD),
     *   CCDISK(MAXFLD), CCCNO(MAXFLD), CCVER(MAXFLD), CNOBEM,
     *   BEMVOL, KSTOK, SCTYPE, VOFF, NSTOK, NCHANG
      LOGICAL   DOFFT, NONEG, DOPTMD, NGRDAT
      INTEGER   NSUBG(MAXFLD), NCLNG(MAXFLD)
      REAL      CELLSG(2), FLUXG(MAXFLD), TFLUXG, SSROT, CCROT,
```

```
     *   XPOFF(MAXFLD), YPOFF(MAXFLD), SCLUG(MAXFLD), SCLVG(MAXFLD),
     *   SCLWG(MAXFLD), SCLUM, SCLVM, FACGRD, DXCG(MAXFLD),
     *   DYCG(MAXFLD), DZCG(MAXFLD), OSFX, OSFY,
     *   PTFLX, PTRAOF, PTDCOF, PARMOD(6)
      DOUBLE PRECISION FREQG(MAXCHA)
      COMMON /MAPDES/  FREQG, SCRBLK, KLNBLK,
     *   CELLSG, SCLUG, SCLVG, SCLWG, SCLUM, SCLVM,
     *   DXCG, DYCG, DZCG, FLUXG, TFLUXG, XPOFF, YPOFF, SSROT, CCROT,
     *   FACGRD, OSFX, OSFY, PTFLX, PTRAOF, PTDCOF, PARMOD,
     *   NCLNG, NSUBG, DOFFT, NONEG, DOPTMD, NGRDAT,
     *   MFIELD, FLDSZ, CCDISK, CCCNO, CCVER, CNOBEM, BEMVOL,
     *   KSTOK, SCTYPE, VOFF, NSTOK, NCHANG
C                                                            End DGDS.
```

7.7.4 DMPR.INC

```
C                                                            Include DMPR.
C                                       include for gridding
C                                       and correction routines.
C                                       NOTE: uses PARAMETER in DGDS.INC
      INTEGER   NXBEM, NYBEM, NXUNF, NYUNF, NXMAX, NYMAX,
     *   ICNTRX(MAXFLD), ICNTRY(MAXFLD), CTYPX, CTYPY, NUVCH, CHUV1,
     *   NCHAVG, UNFBOX, TVFLD, BORES(MAXFLD), BOBEM, MDISKN, MSEQ
      LOGICAL   DOZERO, DOTAPE, DOUNIF
      REAL      XFLD(MAXFLD), YFLD(MAXFLD), XPARM(10), YPARM(10),
     *   TAPERU, TAPERV, ZEROSP(5), BMMAX, BMMIN,
     *   FLDMAX(MAXFLD), FLDMIN(MAXFLD), BEMMAX,
     *   XSHIFT(MAXFLD), YSHIFT(MAXFLD), BLMAX, BLMIN
      CHARACTER MNAME*12, MCLASS*6
      DOUBLE PRECISION FREQUV
      COMMON /GRDCOM/ FREQUV,
     *   XFLD, YFLD, XPARM, YPARM, TAPERU, TAPERV, ZEROSP,
     *   BMMAX, BMMIN, FLDMAX, FLDMIN,
     *   BEMMAX, XSHIFT, YSHIFT, BLMAX, BLMIN,
     *   DOZERO, DOTAPE, DOUNIF,
     *   NXBEM, NYBEM, NXUNF, NYUNF, NXMAX, NYMAX, ICNTRX, ICNTRY,
     *   CTYPX, CTYPY, NUVCH, CHUV1, NCHAVG, UNFBOX,
     *   TVFLD, BORES, BOBEM, MDISKN, MSEQ
      COMMON /GRDCHR/ MNAME, MCLASS
C                                                            End DMPR.
```

7.7.5 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.
```

7.7.6 DUVH.INC

```
C                                                           Include DUVH.
C                                       If you change this include you
C                                       must also change common
C                                       /CATHDR/ in DBCON
C                                       Include for uv header info
      INTEGER   NVIS
      INTEGER   ILOCU, ILOCV, ILOCW, ILOCT, ILOCB, ILOCSU, ILOCFQ,
     *   JLOCC, JLOCS, JLOCF, JLOCR, JLOCD, JLOCIF, NRPARM, LREC,
     *   NCOR, INCS, INCF, INCIF, ICOR0, TYPUVD
      CHARACTER   SOURCE*8, ISORT*2
      DOUBLE PRECISION FREQ, RA, DEC
      COMMON /UVHDR/ FREQ, RA, DEC, NVIS, ILOCU, ILOCV, ILOCW, ILOCT,
     *   ILOCB, ILOCSU, ILOCFQ, JLOCC, JLOCS, JLOCF, JLOCR, JLOCD,
     *   JLOCIF, INCS, INCF, INCIF, ICOR0, NRPARM, LREC, NCOR, TYPUVD
      COMMON /UVHCHR/ SOURCE, ISORT
C                                                           End DUVH.
```

7.8 Routines

7.8.1 APCONV

APCONV is a disk based, two dimensional convolution routine. The image to be convolved and the FFT of the convolving function are passed to APCONV along with two scratch files. All are specified as pointers to the arrays in the common (/CFILES/) from INCLUDE DFIL.INC. NOTE: Uses AIPS LUNs 18, 23, 24, 25.

```
APCONV (NX, NY, LI, LW1, LW2, LO, LC, FACTOR, JBUFSZ, BUFF1, BUFF2,
```

```
  *   BUFF3, SMAX, SMIN, IERR)
Inputs:
   NX       I     The number of columns in the input image (must be
                  a power of 2).
   NY       I     The number of rows in the input image.
   LI       I     File number in /CFILES/ of input.
   LW1      I     File number in /CFILES/ of work file no. 1
                     size = (4*NX x NY+2).
   LW2      I     File number in /CFILES/ of work file no. 1
                     size = (4*NX x NY+2).
   LO       I     File number in /CFILES/ of output.
   LC       I     File number in /CFILES/ of FFT of convolving fn.
                     size = (4*NX x NY+2).
   FACTOR   R     Normalization factor for convolving function; i.e.
                  is multiplied by the transform of the convolving
                  function
   JBUFSZ   I     Size of BUFF1,2,3 in AIPS bytes.  Should be
                  large, at least 8192 words.
Output:
   BUFF1    R(*)  Working buffer
   BUFF2    R(*)  Working buffer
   BUFF3    R(*)  Working buffer
   SMAX     R     Maximum value in the output file.
   SMIN     R     Minimum value in the output file.
   IERR     I     Return error code, 0 => OK, otherwise error.
```

7.8.2 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 reduced in size if necessary at completion of CALCOP. Makes heavy use of common /CFILES/ from INCLUDE DFIL.INC.

```
CALCOP (DISK, CNOSCR, BUFFER, BUFSZ, IRET)
Inputs:
   DISK     I       Disk number for cataloged 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: (DUVH.INC)
   LREC     I       length of vis. record in R words.
   NRPARM   I       number of R random parameters.
In/out:
   CNOSCR   I       Catalog slot number for if cataloged file;
                    (DFIL.INC) 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       (DUVH.INC) 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
```

7.8.3 DSKFFT

DSKFFT is a disk based, two dimensional FFT. If the FFT all fits in AP memory then the intermediate result is not written to disk. Input or output images in the sky plane are in the usual form (i.e. center at the center, X the first axis). Input or output images in the uv plane are transposed (v the first axis) and the center-at-the-edges convention with the first element of the array the center pixel. NOTE: Uses AIPS LUNs 23, 24, 25. Makes use of commons in INCLUDE DFIL.INC.

```
DSKFFT (NR, NC, IDIR, HERM, LI, LW, LO, JBUFSZ, BUFF1,
  *   BUFF2, SMAX, SMIN, IERR)
Inputs:
   NR      I     The number of rows in input array (# columns in
                 output).  When HERM is TRUE and IDIR=-1, NR is
                 twice the number of complex rows in the input file
   NC      I     The number of columns in input array (# rows in
                 output).
   IDIR    I     1 for forward (+i) transform, -1 for inverse (-i)
                 transform.
                 If HERM = .TRUE. the follwing are recognized:
                    IDIR=1 keep real part only.
                    IDIR=2 keep amplitudes only.
                    IDIR=3 keep full complex (half plane)
   HERM    L     When HERM = .FALSE., this routine does a complex to
                 complex transform.
                 When HERM = .TRUE. and IDIR = -1, it does a
                 complex to real transform.  When HERM = .TRUE. and
                 IDIR = 1, it does real to complex.
   LI      I     File number in (DFIL.INC) of input.
   LW      I     File number in (DFIL.INC) of work file (may equal LI)
   LO      I     File number in (DFIL.INC) of output.
   JBUFSZ  I     Size of BUFF1, BUFF2 in bytes.  Should be large
                 at least 4096 R   words.
Output:
   BUFF1   R(*)  Working buffer
   BUFF2   R(*)  Working buffer
   SMAX    R     For HERM=.TRUE. the maximum value in output file.
   SMIN    R     For HERM=.TRUE. the minimum value in output file.
   IERR    I     Return error code, 0 => okay, otherwise error.
```

7.8.4 GRDCOR

GRDCOR normalizes and corrects for the gridding convolution function used in gridding uv data to make the image. Uses AIPS LUNs 18 and 19

```
GRDCOR (IFIELD, DOGCOR, DISKI, CNOSCI, DISKO, CNOSCO,
  *   MAPMAX, MAPMIN, JBUFSZ, BUFF1, BUFF2, BUFF3, IRET)
Input:
   IFIELD      I     The subfield number, if = 1 the histogram is
                     zero filled first.
                     If IFIELD = 0 the input is assumed to be
                     a beam.
   DOGCOR      L     If TRUE, do gridding convolution correction.
   DISKI       I     Input file disk number for catalogd files,
                     .LE. 0 => /CFILES/ scratch file.
   CNOSCI      I     Input file catalog slot number or /CFILES/
                     scratch file number.
   DISKO       I     Output file disk number for catalogd files,
                     .LE. 0 => /CFILES/ scratch file.
   CNOSCO      I     Output file catalog slot number or /CFILES/
                     scratch file number.
   JBUFSZ      I     Size in bytes of buffers. Dimension of
                     BUFF1,2,3  must be at least 4096 words.
From commons: (Includes DGDS, DMPR, DUVH)
   BEMMAX       R    Sum of the weights used in gridding, used
                     to normalize images.
   CTYPX,CTYPY  I    Convolving function types for RA and Dec
   XPARM(10)    R    Convolving function parameters for RA
                     XPARM(1) = support half width.
   YPARM(10)    R    Convolving function parameters for Dec.
   BORES(16)    I    Block offset desired in output file for
                     an image, 1 per field. (1 rel.)
   BOBEM        I    Block offset desired in output file for
                     an beam. (1 rel.)
   NGRDAT       L    If FALSE get map size, scaling etc. parms
                     from the model map cat. header. If TRUE
                     then the values filled in by GRDAT must
                     already be filled into the common.
The following must be provided if NGRDAT is .TRUE.
   FLDSZ(2,*)        I   Dimension of map in RA, Dec (cells)
   ICNTRX,ICNTRY(*) I   The center pixel in X and Y for each
                        field.
Output:
   MAPMAX      R     The maximum value in the resultant image.
   MAPMIN      R     The minimum value in the resultant image.
   BUFF1       R     Working buffer
   BUFF2       R     Working buffer
   BUFF3       R     Working buffer
   IRET        I     Return error code. 0=>OK, error otherwise.
```

7.8.5 MAKMAP

MAKMAP makes a image or a dirty beam given a uv data set. The data may either calibrated or uncalibrated (raw) data and calibration and various selection criteria may be (optionally) applied. Data in an arbitrary sort order can be processed although only "TB" ordered data can be calibrated or edited.

The weights of the data may (optionally) have the uniform weighting correction made.

The visibilities are convolved onto the grid using the convolving function specified by CTYPX, CTYPY, XPARM, YPARM. The defaults for these values are filled in by a call to GRDFLT. The gridded data is phase rotated so that the map center comes out at location ICNTRX, ICNTRY. If requested, a uv taper is applied to the visibility weights before gridding. If necessary, a three dimension phase reference position shift is done.

Multiple channels may be gridded onto the same grid; a technique calles bandwidth synthesis. This bandwidth synthesis (BS) process may use the SCRWRK file. For bandwidth synthesis both the CNOSCO and SCRWRK files should be big enough for an extra m rows, where m is the half width of the X convolving function. Zero spacing flux densities are gridded if provided.

The final image will be normalized and (optionally) corrected for the effects of the gridding convolution function.

The input and output files are specified by either disk number and catalog number or as pointers in the /CFILES/ common from INCLUDE DFIL.INC. Input uv data file in UV file CNOSCI, DISKI. Output image file in image file CNOSCO, DISKO and may optionally be created as a scratch file.

Communication is through commons in INCLUDES DSEL.INC, DGDS.INC and DMPR.INC.

Uses buffer UBUFF from the UVGET commons (include DSEL.INC)

```
MAKMAP (IFIELD, DISKI, CNOSCI, DISKO, CNOSCO, SCRGRD, SCRWRK,
  *   CHANUV, CHANIM, DOCREA, DOINIT, DOBEAM, DOSEL, DOGCOR,
  *   JBUFSZ, BUFFER, IRET)
Inputs:
   IFIELD      I     Field number to map, if 0 then make a beam.
   DISKI       I     Input file disk number for cataloged files,
                     .LE. 0 => /CFILES/ scratch file.
   CNOSCI      I     Input file catalog slot number or /CFILES/
                     scratch file number.
   DISKO       I     Output file disk number for cataloged files,
                     .LE. 0 => /CFILES/ scratch file.
   CNOSCO      I     Output file catalog slot number or /CFILES/
                     scratch file number.  If DOCREA is FALSE and
                     DISKO=0 and CNOSCO=0 a scratch file is created.
   SCRGRD      I     Grid scratch file number, will be set if the
                     file is created, (DOINIT=TRUE)
   SCRWRK      I     Work scratch file number, will be set if the
                     file is created, (DOINIT=TRUE)
   CHANUV      I     Channel number to grid.  If DOSEL=TRUE
                     then this is 1-rel wrt the selected data.
   CHANIM      I     Channel number of output image.
   DOCREA      L     If TRUE, Create/catalog output image file.
   DOINIT      L     If TRUE, initialize scratch files, set defaults
                     for convolving functions.  Should
                     be TRUE on first call, and FALSE there after.
   DOBEAM      L     If TRUE a grid the beam before gridding the
                     field.  See useage notes.
   DOSEL       L     If true, data need to be reformatted to a
                     single Stokes' type.  If TRUE, the cataloged
                     file NAME, CLASS etc should be filled into
                     UNAME, UCLAS, UDISK, USEQ in common /SELCAL/
   DOGCOR      L     If TRUE, correct image for gridding
                     convolution correction function.
                     (Normally .TRUE.)
   JBUFSZ      I     Size in bytes of buffers. Dimension of
                     BUFFER  must be at least 4096 R.
From commons: (Includes DGDS and DMPR)
   MFIELD       I    The number of fields which are going to
                     to be imaged (excluding any beam).
                     MUST be filled in.
   FLDSZ(2,*)   I    Dimension of map in RA, Dec (cells) of each
```

```
                     field.  MUST be completely filled in before the
                     DOINIT=TRUE call if the output file (either
                     image or scratch) is to be created or zeroed
                     if the files already exist.
   DOUNIF       L    If TRUE, apply Uniform weighting. Should be
                     TRUE on only the first call, otherwise it will
                     be applied again.
   NCHAVG       I    Number of channels to grid together for
                     bandwidth synthesis.
   UNFBOX       I    Half width of unif. wt. counting box size.
   CTYPX,CTYPY  I    Convolving function types for RA and Dec
   XPARM(10)    R    Convolving function parameters for RA
                     XPARM(1) = support half width.
   YPARM(10)    R    Convolving function parameters for Dec.
   UVRNG(2)     R    Minimum and maximum baseline lengths in
                     1000's wavelengths. 0's => all
   XSHIFT(16)   R    Shift in X (after rotation) in asec.
                     in projected coordinates. 1 per field.
   YSHIFT(16)   R    Shift in Y (after rotation) in asec.
                     in projected coordinates. 1 per field.
   STOKES       C*4  Stokes types wanted.
                     'I','Q','U','V','R','L'
   DOZERO       L    If true then do zero spacing flux.
   ZEROSP(5)    R    Zero spacing flux, 1=>flux density (Jy)
                     5 => weight to use.
                     polarization.
   TFLUXG       R    The total flux density removed from the data,
                     this will be subtracted from the zero spacing
                     flux before gridding.
   DOTAPE       L    True if taper requested.
   TAPERU,TAPERV R   TAPER ( to 30%) in u and v (kilolamda)
   NXUNF,NYUNF   I   Dimension (cells) of the map in RA and Dec
                     to be used to set uniform weighting.
                     (should be min. of FLDSZ)
The following must be provided if DOSEL is .FALSE.:
  CATBLK(256)   I    Catalog header for uv data input file.
                     (only used on DOINIT=TRUE call)
The following must be provided if DOCREA is .TRUE. (includes DMPR,
 DGDS)
   MNAME           C*12 Output image name.
   MCLASS          C*6  Output image class.
                        (If more than 1 field the last 2 char
                        are used to encode the field number)
   MDISK           I    Desired image file output disk
   MSEQ            I    Desired image file output sequence no.
The following must be provided if the output file is to be created;
either by setting DOCREA=TRUE or DISKO=CNOSCO=0.
   FLDSZ(2,*)      I    Dimension of map in RA, Dec (cells)
   NXBEM,NYBEM     I    Dimension (cells) of beam.
   CELLSG(2)       R    The cell spacing in X and Y in arcseconds.
   XSHIFT(16)      R    Shift in X (after rotation) in asec.
                        in projected coordinates. 1 per field.
   YSHIFT(16)      R    Shift in Y (after rotation) in asec.
                        in projected coordinates. 1 per field.
```

```
   ICNTRX,ICNTRY(*) I   The center pixel in X and Y for each
                        field. 0 values cause the default.
The following must be provided if DOCREA is FALSE and output
files already exist. (Includes DGDS).
   CCDISK(16)   I    Disk numbers of the output images.
                     (Must be zeroed if not filled in.)
   CCCNO(16)    I    Catalog slot numbers of output images.
                     (Must be zeroed if not filled in.)
The following must be provided if DOSEL is .TRUE.
(Includes 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.
   FGVER        I    FLAG file version number, if .le. 0 then
                     NO flagging is applied.
   SOURCS(1)   C*16  Name of desired source.
   TIMRNG(8)    R    Start day, hour, min, sec, end day, hour,
                     min,sec. 0's => all
   STOKES       C*4  Stokes types wanted.
                     'I','Q','U','V','R','L'
   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.
The following must be provided if DOCAL is TRUE.
   ANTENS(50)   I    List of antennas selected, 0=>all,
                     any negative => all except those specified
   GAUSE        I    GAIN (CL or SN) file version number to use.
Output:
   DISKI       I     UV data file disk if data reformatted.
   CNOSCI      I     Reformatted uv data scratch file number
                     to be used in subsequent calls.
   DISKO       I     Output image file disk number if output file.
                     created and/or cataloged (DOCREA=TRUE
                     or input DISKO=0 and CNOSCO=0).
   CNOSCO      I     Output image file catalog slot number
                     or scratch file number if output file created.
   SCRGRD      I     Grid scratch file number, will be set if the
                     file is created, (DOINIT=TRUE)
   SCRWRK      I     Work scratch file number, will be set if the
                     file is created, (DOINIT=TRUE)
   DOSEL       L     Set to FALSE if data reformatted.
   DOBEAM      L     Set to FALSE.
   DOINIT      L     Set to FALSE.
   BUFFER(*)   R     Working buffer
   IRET        I     Return error code. 0=>OK, error otherwise.
Output in Common:
   DOUNIF      L     Set to FALSE if uniform weighting applied.
   UBUFSZ      I     Buffer size for UBUFF (UVGET buffer)
   MNAME       C*12  Output image name. (defaults applied)
```

```
   MCLASS      C*6   Output image class (defaults applied)
   MDISK       I     Desired image file output disk
                     (defaults applied)
   MSEQ        I     Desired image file output sequence no.
                     (defaults applied)
   FLDMAX(*)   R     Maximum pixel value in field.
   FLDMIN(*)   R     Minimum pixel value in field.
The following are filled in if a output file is created:
   CCDISK(16)  I     Disk numbers of the output images.
   CCCNO(16)   I     Catalog slot numbers of output images.
Useage Notes:
 1) The input uvdata file is, with one exception, assumed to be
  accurately described by the contents of CATR and the common
  /UVHDR/ (include DUVH).  The exception is that the u, v and
  w may refer to a different frequency.  The reference frequency for
  the u,v and w terms is taken from the input CATBLK in the DOINIT
  TRUE call unless the data is reformatted (DOSEL=TRUE).
  In this latter case this frequency is obtained from UVGET call.
  If DOSEL = TRUE the input value of CATBLK is ignored.
 2) Information about the output image is obtained from the
  catalog header for the relevant file.  If MAKMAP makes the
  output file this information is filled in.  If MAKMAP does not
  make the output image file then this information must be filled
  in before hand.  Routine IMCREA will help do this.  Note: even
  scratch files are cataloged and thus have a catalog header.
  If MAKMAP does not create the output files, CCDISK(IFIELD) and
  CCCNO(IFIELD) should give their disk and catalog slot number
  before the call to MAKMAP.
 3) only one polarization can be processed and the input data
  to the gridding routine is assumed to be in the desired Stokes'
  type (i.e. I, Q, U, V etc.).
     If DOSEL = TRUE the input data will be selected, calibrated
  and reformatted as specified in common (include DSEL).
  Only Stokes' types I,Q,U,V,R,L should be used.
     Multiple channels may be gridded together a la bandwidth
  synthesis by specifying NCHAVG > 1. One channel of several
  channels may be gridded specified by CHANUV.
 4) If DOSEL=FALSE on the first call (i.e. the data is not
  reformatted),  the random parameters in the data should include,
  in order, u, v, w, weight (optional), time (optional) and baseline
  (optional).  While the last are optional and not used, the last
  words of random parameters are used as work space and, if they
  are missing, u, v, and w may be clobbered.  The weights are
  required but may be passed either as random parameters or as
  part of the regular data array, CATR should tell which.
  If DOSEL=TRUE is used these conditions will be satisfied.
 5) The necessary image normalization constant for proper
  normalization of the FFTed image is produced only by gridding the
  beam.  If a beam is to be made, it should be done first; in this
  case DOBEAM should be FALSE in all calls.  If a beam is not
  desired then the first call to MAKMAP should have DOBEAM TRUE and
  FALSE on subsequent calls.  Note MAKMAP sets DOBEAM to FALSE.
 6) Much of the control information used by MAKMAP is passed to and
  stored in commons.  The calling routine should have the following
```

```
includes:
DHDR.INC, DUVH.INC, DFIL.INC, DMPR.INC, DGDS.INC, DSEL.INC
NOTE: care should be taken that the contents of these commons
not be clobbered by overlaying.
7) If calibration is applied then up to 8 map and 3 non map files
will be open at once; this should be reflected in the call to
ZDCHIN and the dimension of FTAB in the main routine of the
calling program.  MAKMAP may use AIPS LUNs 16, 17, 18, 19,
20, 21, 22, 23, 24, 25, 28, 29, 30.
```

7.8.6 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 common /SELCAL/  (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.
```

7.8.7 UVMDIV

UVMDIV divides model visibilities derived from CLEAN or Gaussian components or images into a uv data set. The weights of the data returned will be the input values multiplied by the model amplitude.

A variety of model computation methods are available; if a single pass through VISDFT, the DFT routine, is not sufficient then the data is copied to a scratch file which has space for a second copy of the data, the model values are computed and summed in these locations and finally then model is divided into the data and written to the output file.

Extensive use is made of commons to communicate with UVMDIV, in particular /MAPDES/ (include DGDS.INC) contains most of the critical information about the model components files or images to be used. Common /UVHDR/ (DUVH.INC filled in by UVPGET) is presumed to describe the uv data files.

If the data is not sorted 'X*' and MODEL=1 then UVMSUB will use the DFT irregardless of the value of METHOD.

Also fills in frequency table (NCHANG, FREQG) in INCLUDE DGDS.INC

```
UVMDIV (DISKI, CNOSCI, DISKO, CNOSCO, MODEL, METHOD, DOMSG, CHANEL,
  *   NCHAN, CATBLK, JBUFSZ, FREQID, BUFF1, BUFF2, BUFF3, IRET)
 Inputs:
   DISKI      I   Input disk number. if .LE. 0 then input is a
                  scratch file.
   CNOSCI     I   Input file catalog slot number or /CFILES/
                  scratch file number.
   DISKO      I   Output disk number. if .LE. 0 then output is a
                  scratch file.
```

```
   CNOSCO      I   Output file catalog slot number or /CFILES/
                   scratch file number.  If .LE. 0 then one of the
                   internal scratch files will be used.
   MODEL       I   1=> clean components, 2=>image.
   METHOD      I   1=>gridded, -1=>DFT, 0=>chose.
   DOMSG       L   If true give percent done messages for DFT.
   CHANEL      I   First uv data channel to subtract.
   NCHAN       I   Number of frequency channels to subtract.
   CATBLK(256)I   Uv data catalog header record.
   JBUFSZ      I   Size of BUFF1,2,3 in bytes, must be at least 4096
                   words.
   FREQID      I   Freq ID number, if it exists.
   BUFF1,2,3  R   Work buffers.
Inputs from COMMON /MAPDES/:
   MFIELD      I     Number of fields
   NSUBG(*)    I     Number of components already sub.
   NCLNG(*)    I     Number of components per field.
   CCDISK(*)   I     Disk numbers for CC files
   CCCNO(*)    I     Catalog slot numbers for CC files.
   CCVER(*)    I     CC file version number for each field.
   FACGRD      R     Value to multiply clean component fluxes
                     by before subtraction (negative for sum).
   SCTYPE      C*2   Scratch file type to create. (eg. 'SC')
   NONEG       L    Stop reading comps. from a file past the first
                    negative component. (DFT modeling ONLY)
   DOPTMD      L    Use the point model specified by PTFLX, PTRAOF,
                    PTDCOF (DFT modeling ONLY)
   PTFLX       R    Point model flux density (Jy) (I pol. only)
   PTRAOF      R    Point model RA offset from uv phase center
                    (asec)
   PTDCOF      R    Point model Dec. offset from uv phase center
Input from COMMON /UVHDR/:
   LREC        I     Length of visibility record.
   NVIS        I     Number of visibility records.
   NRPARM      I     "Random" parameters before data, can be used
                     to skip observed values when computing model.
Output:
   CNOSCO      I   Output file catalog slot number or /CFILES/
                   scratch file number.  Value returned if not
                   specified in call.
   IRET        I   Return error code. 0=>OK, otherwise failed.
```

7.8.8 UVMSUB

UVMSUB subtracts a CLEAN or Gaussian model or an image from a set of uv data. Extensive use is made of commons to communicate with UVMSUB, in particular /MAPDES/ (include DGDS.INC) contains most of the critical information about the model components files or images to be subtracted. Common /UVHDR/ (filled in by UVPGET) is presumed to describe the uv data files.

If the data is not sorted 'X*' and MODEL=1 then UVMSUB will use the DFT irregardless of the value of METHOD.

Also fills in frequency table (NCHANG, FREQG) in INCLUDE DGDS.INC

```
UVMSUB (DISKI, CNOSCI, DISKO, CNOSCO, MODEL, METHOD, CHANEL, NCHAN,
  *   DOSUM, DOMSG, CATBLK, JBUFSZ, FREQID, BUFF1, BUFF2, BUFF3,
```

```
 *   IRET)
 Inputs:
   DISKI      I   Input disk number. if .LE. 0 then input is a
                  scratch file.
   CNOSCI     I   Input file catalog slot number or /CFILES/
                  scratch file number.
   DISKO      I   Output disk number. if .LE. 0 then output is a
                  scratch file.
   CNOSCO     I   Output file catalog slot number or /CFILES/
                  scratch file number.
   MODEL      I   1=> clean components, 2=>image.
   METHOD     I   1=>gridded, -1=>DFT, 0=>chose.
   CHANEL     I   First uv data channel to subtract.
   NCHAN      I   Number of frequency channels to subtract.
   DOSUM      L   If true then sum component fluxes in FLUXG,
                  TFLUXG.
   DOMSG      L    If true give percent done messages for DFT.
   CATBLK(256)I   Uv data catalog header record.
   JBUFSZ     I   Size of BUFF1,2,3 in bytes, must be at least 4096
                  words.
   FREQID     I   Freq ID number, if it exists.
Inputs from COMMON /MAPDES/:
   MFIELD      I     Number of fields
   NSUBG(*)    I     Number of components already sub.
   NCLNG(*)    I     Number of components per field.
   CCDISK(*)   I     Disk numbers for CC files
   CCCNO(*)    I     Catalog slot numbers for CC files.
   CCVER(*)    I     CC file version number for each field.
   FACGRD      R     Value to multiply clean component fluxes
                     by before subtraction (negative for sum).
   NONEG       L    Stop reading comps. from a file past the first
                    negative component. (DFT modeling ONLY)
   DOPTMD      L    Use the point model specified by PTFLX, PTRAOF,
                    PTDCOF (DFT modeling ONLY)
   PTFLX       R    Point model flux density (Jy) (I pol. only)
   PTRAOF      R    Point model RA offset from uv phase center
                    (asec)
   PTDCOF      R    Point model Dec. offset from uv phase center
Input from COMMON /UVHDR/ (DUVH.INC):
   LREC        I     Length of visibility record.
   NVIS        I     Number of visibility records.
   NRPARM      I     "Random" parameters before data, can be used
                     to skip observed values when computing model.
   BUFF1,2,3  R   Work buffers.
Output:
   IRET       I   Return error code. 0=>OK, otherwise failed.
```

THIS PAGE DELIBERATELY BLANK