TABLE OF CONTENTS

1. MW_Sounder_FCDR/amsu_gps.f [Programs]

[Top] [Programs]

NAME:

```
amsu_gps.exe
```

LOCATION:

```
$SRC_DIR/Src_fortran/
```

DESCRIPTION:

```
For the specified GPSRO mission, this program loops over
the specified range of julian days and years, and reads
in all of the GPSRO profiles for each day.
Each profile is passed to an AMSU forward model routine
which calculates the brightness temperatures for 15
AMSU channels. The resulting brightness temperatures for
each day are written to an ASCII file for later use.
```

EXTERNALS:

```
AMSUfast module for AMSU Forward Model Calculations.
```

USAGE:

```
For each GPSRO mission {'champ', 'cosmic', 'grace', 'metopa'},
the user needs to first edit the RUN parameters for the program
contained in the file 'CONFIG.amsu_gps' to set the mission
name, the range of days to process and the input/output paths
for the data.

The program is then compiled to generate an executable:

     bash> make amsu_gps.exe
     bash> vi CONFIG.amsu_gps
     bash> ./amsu_gps.exe
```

PARAMETERS:

```
        gps = Name of GPSRO mission to process
   YEAR_BEG = Begining year
   JDAY_BEG = Begining julian day
   YEAR_END = Ending year
   JDAY_END = Ending julian day
     InPath = Path to interpolated GPSRO profiles
    OutPath = Path to output brightness temperature files
```

INPUTS:

```
Input ASCII files containing the {gps} profile data should be
stored with the following directory structure and file format:

      $InPath/{gps}/YYYY/Prf_YYYY.DDD.dat'
```

OUTPUTS:

```
AMSU brightness temperatures are stored in the files:

      $OutPath/{gps}/YYYY/AMSU_{gps}_atmBt_YYYY.DDD.dat
```

AUTHOR:

```
Xinjia Zhou / Ben Ho (?)
```

REVISIONS:

```
Patrick Callaghan
 *Cleaned up and Packaged the AMSU Forward model routines
   into module AMSUfast_mod.f for CDR
 *Cleaned up, added program inputs via CONFIG file, and
   Formatted main program for CDR
```

2. MW_Sounder_FCDR/AMSUfast_mod.f [Programs]

[Top] [Programs]

COPYRIGHT

3. MW_Sounder_FCDR/bin_aqua_monthlymean.pro [Programs]

[Top] [Programs]

NAME

bin_aqua_monthlymean.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

For the AQUA polar orbiter and AMSU channel and range
of months, AMSU values are averaged on a 2.5 degree grid.
Monthly gridded values are written to ASCII files for later
use.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

Daily values of AMSU are read from the files:

$InPath/aqua/YYYY/amsu_aqua_YYYY.DDD.dat

OUTPUTS:

Monthly gridded means are written to the files:

$OutPath/aqua/bin_aqua_YYYYMM_chan{chan}.dat

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

4. MW_Sounder_FCDR/bin_noaa_monthlymean.pro [Programs]

[Top] [Programs]

NAME

bin_noaa_monthlymean.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

For the specified polar orbiter and AMSU channel and range
of months, AMSU values are averaged on a 2.5 degree grid.
Monthly gridded values are written to ASCII files for later
use.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

Daily values of AMSU are read from the files:

$InPath/{noaa}/YYYY/amsu_{noaa}_YYYY.DDD_ch5-12.dat

OUTPUTS:

Monthly gridded means are written to the files:

$OutPath/{noaa}/bin_{noaa}_YYYYMM_chan{chan}.dat

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

5. MW_Sounder_FCDR/combine_amsu.pro [Programs]

[Top] [Programs]

NAME

combine_amsu.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

Read in gridded data for the specified set of polar
orbiters, combine data by a simple average of the data
from each satellite at each gridpoint, and write out
the results in an ASCII file.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

Gridded polar orbiter data values are read from file with
the following directory structure and file format:

$InPath/bin_{noaa[*]}_YYYYMM_{chan}.dat

OUTPUTS:

```
Gridded combined polar orbiter data values are written to an
ASCII file:

  $OutPath/bin_YYYYMM_{chan}_noaqua.dat
```

AUTHOR:

```
Xinjia Zhou / Ben Ho (?)
```

REVISIONS:

```
Patrick Callaghan (Formatted for CDR)
```

COPYRIGHT

6. MW_Sounder_FCDR/convert_amsu_bygps.pro [Programs]

[Top] [Programs]

NAME

convert_amsu_bygps.pro

LOCATION

```
$SRC_DIR/Src_IDL/
```

DESCRIPTION:

```
For the specified AMSU dataset, Read in GPSRO correction
coeffcients from the specfied file, read in gridded AMSU
data, apply the correction, and write the converted data
to the given output file.
```

EXTERNALS:

```
none.
```

USAGE:

```
The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.
```

INPUTS:

```
Monthly linear fit coefs between NOAA and GPSRO data
are read from:
  $InPath_coef/timeseries_offset_slope_{noaa}_gpsro.dat

Gridded NOAA AMSU values are read from:
  $InPath_noaa/bin_{noaa}_YYYYMM_chan9.dat
```

OUTPUTS:

```
Converted NOAA AMSU values are written to the
ASCCI file:
  $OutPath_noaa/bin_{noaa}_YYYYMM_chan9_converted.dat
```

AUTHOR:

```
Xinjia Zhou / Ben Ho (?)
```

REVISIONS:

```
Patrick Callaghan (Formatted for CDR)
```

COPYRIGHT

7. MW_Sounder_FCDR/extract_aqua_amsu.pro [Programs]

[Top] [Programs]

NAME

extract_aqua_amsu.pro

LOCATION

```
$SRC_DIR/Src_IDL/
```

DESCRIPTION:

```
For the AQUA polar orbiter, this program loops over the
specified range of months and years, and reads the L1B
HDF files from the specified input directory.
(downloaded from NASA website:                          )
( http://disc.sci.gsfc.nasa.gov/AIRS/data-holdings/     )
(                    by-data-product/amsuL1B_Rad.shtml  )
Values of brightness temperature and related data are stored
for each julian day in ASCII files for later use.
```

EXTERNALS:

```
none.
```

USAGE:

```
The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.
```

INPUTS:

```
The L1B HDF files should be stored with the following directory
structure and file format:

 $InPath/aqua/YYYYMM/AIRS.YYYY.MM.DD.SSS.L1B.AMSU_Rad.v5.*.hdf
```

OUTPUTS:

```
Output ASCII files are stored with the following directory
structure and file format:

 $OutPath/aqua/YYYY/amsu_aqua_YYYY.DDD.dat
```

AUTHOR:

```
Xinjia Zhou / Ben Ho (?)
```

REVISIONS:

```
Patrick Callaghan (Formatted for CDR)
```

COPYRIGHT

```
Â© University Corporation for Atmospheric Research (UCAR) 2011.
All rights reserved. Please refer to the UCAR copyright
statement in the main directory
```

8. MW_Sounder_FCDR/extract_gpsro_profiles.pro [Programs]

[Top] [Programs]

NAME

extract_gpsro_profiles.pro

LOCATION

```
$SRC_DIR/Src_IDL/
```

DESCRIPTION:

```
For the specified GPSRO mission, this program loops over
the specified range of julian days and years, and reads
in all of the WET/ATM profiles for each day.

The daily profiles DRY temperatures from 'atmPrf' files and
profiles of water vapor from 'wetPrf' files are interpolated
```

```
to the 100 fixed pressure levels used by the AMSU forward
model. Missing data values are replaced using Standard
atmosphere values.

The resulting interpolated profiles are related values for
each Julian day are stored for later use.
```

EXTERNALS:

```
readstd.pro
```

USAGE:

```
The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.
```

INPUTS:

```
GPSRO level2 profile files should be stored with the following
directory structure and file format:

  $InPath/{gps}/level2/atmPrf/YYYY.DDD/atmPrf_MMMM.YYYY.DDD.*_nc
  $InPath/{gps}/level2/wetPrf/YYYY.DDD/wetPrf_MMMM.YYYY.DDD.*_nc

 ** This program requires that matched 'atm' and 'wet' profiles
    are used. To achieve this, the set of ATM files are read
    from the $InPath directory. Then the 2 occurances of 'atm'
    in each full file name are replaced with 'wet'. The resulting
    WET profile must exist to be included. Because of this,
    the user must be careful not to include 'atm' anywhere in the
    $InPath directory name or the program will fail.
```

OUTPUTS:

```
Output ASCII files are stored with the following directory
structure and file format:

  $OutPath/{gps}/YYYY/Prf_YYYY.DDD.dat'
```

AUTHOR:

```
Xinjia Zhou / Ben Ho (?)
```

REVISIONS:

```
Patrick Callaghan (Formatted for CDR)
```

COPYRIGHT

9. MW_Sounder_FCDR/extract_noaa_amsu.pro [Programs]

[Top] [Programs]

NAME

extract_noaa_amsu.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

For the Specified NOAA polar orbiter 'noaa1', this program loops over the specified range of months and years, and reads the L1B binary files from the specified input directory. (downloaded from NOAA-CLASS TOVS from from KLM 8.3.1.6.1) Values of brightness temperature and related data are stored for each julian day in ASCII files for later use.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set directly below this header. The user needs to edit these values to process the desired values and then execute the program from within IDL.

INPUTS:

L1B files should be stored with the following directory structure and file format:

$InPath/{noaa1}/YYYYMM/NSS.AMAX.SS.Dyyddd.Snnnn.Ennnn.CC

OUTPUTS:

Output ASCII files are stored with the following directory structure and file format:

$OutPath/{noaa1}/YYYY/amsu_{noaa1}_YYYY.DDD_ch5-12.dat

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

10. MW_Sounder_FCDR/gen_netCDF_anomaly.f [Programs]

[Top] [Programs]

NAME

gen_netCDF_anomaly.f

LOCATION

$SRC_DIR/Src_fortran/

DESCRIPTION:

Read in gridded ASCII data for the specified months,
write out desired values to netCDF files. The original
IDL program had to be rewritten in FORTRAN to produce
Version 4 netCDF files. The cumbersome usage methodology
is the same as for the IDL routines for consistency.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

Gridded combined polar orbiter data values are read from an
ASCII file:

  $InPath/tls_anamoly_YYYY_1.0

OUTPUTS:

netCDF files are written with the following directory
structure and file format:

  $OutPath/AMSU_{proc}_{chan}_V1.0_YYYYMM.nc

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

11. MW_Sounder_FCDR/gen_netCDF_climatology.f [Programs]

[Top] [Programs]

NAME

gen_netCDF_climatology.f

LOCATION

$SRC_DIR/Src_fortran/

DESCRIPTION:

Read in gridded ASCII data for the specified months,
write out desired values to netCDF files.The original
IDL program had to be rewritten in FORTRAN to produce
Version 4 netCDF files. The cumbersome usage methodology
is the same as for the IDL routines for consistency.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

Gridded combined polar orbiter data values are read from an
ASCII file:

$InPath/tls_climatology_1.0

OUTPUTS:

netCDF files are written with the following directory
structure and file format:

$OutPath/AMSU_{proc}_{chan}_V1.0_MM.nc

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

12. MW_Sounder_FCDR/gen_netCDF_monthly.f [Programs]

[Top] [Programs]

NAME

gen_netCDF_monthly.f

LOCATION

$SRC_DIR/Src_fortran/

DESCRIPTION:

Read in gridded ASCII data for the specified months,
write out desired values to netCDF files. The original
IDL program had to be rewritten in FORTRAN to produde
Version 4 netCDF files. The cumbersome usage methodology
is the same as for the IDL routines for consistency.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

Gridded combined polar orbiter data values are read from an
ASCII file:

  $InPath/bin_YYYYMM_{chan}.dat

OUTPUTS:

netCDF files are written with the following directory
structure and file format:

  $OutPath/AMSU_{proc}_{chan}_V1.0_YYYYMM.nc

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

13. MW_Sounder_FCDR/generate_climatology_anomaly.pro [Programs]

[Top] [Programs]

NAME

generate_climatology_anomaly.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

For the given range of months/years, generate a mean monthly climatology and save the results. For an atlerante range of months, read in gridded ASCII data for the specified months, subtract the climatological mean for the data and write out anomaly values.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set directly below this header. The user needs to edit these values to process the desired values and then execute the program from within IDL.

INPUTS:

Gridded calibrated polar orbiter data values are read from ASCII files:

$InPath/bin_YYYYMM_{chan}.dat

OUTPUTS:

Climatological mean values are written to an ASCII file with the structure and file format:

$CoutPath/$ClimatologyName

Monthly Anomaly values are written to ACSII files with the structure and file format:

$AoutPath/tls_anamoly_YYYY_1.0

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

14. MW_Sounder_FCDR/match_gps_aqua.pro [Programs]

[Top] [Programs]

NAME

match_gps_aqua.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

For the given range of Julian days, read in AMSU data extracted from the AQUA dataset and simulated AMSU brightness temperatures from profiles of the specified GPSRO mission.

Find colocated measurements satisfying the matchup criteria and then save the matched values in an ASCCI file for later use.

EXTERNALS:

none.

USAGE:

```
The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.
```

INPUTS:

```
AQUA AMSU data should be stored with the following directory
structure and file format:

  $InPath_noaa/aqua/YYYY/amsu_aqua_YYYY.DDD.dat

GPSRO simulated AMSU data should be stored in the format:

  $InPath_gps/{gps}/YYYY/AMSU_{gps}_atmBt_YYYY.DDD.dat
```

OUTPUTS:

```
Matched AMSU values are stored in ASCII files with the following
directry structure and file format:

  $OutPath/aqua_{gps}/YYYY/match_{gps}_aqua_YYYY.DDD.dat
```

AUTHOR:

```
Xinjia Zhou / Ben Ho (?)
```

REVISIONS:

```
Patrick Callaghan (Formatted for CDR)
```

COPYRIGHT

15. MW_Sounder_FCDR/match_gps_noaa.pro [Programs]

[Top] [Programs]

NAME

match_gps_noaa.pro

LOCATION

```
$SRC_DIR/Src_IDL/
```

DESCRIPTION:

```
For the given range of Julian days, read in AMSU data
extracted from the specified NOAA dataset and simulated
AMSU brightness temperatures from profiles of the
specified GPSRO mission.
```

```
Find colocated measurements satisfying the matchup
criteria and then save the matched values in an ASCCI
file for later use.
```

EXTERNALS:

```
none.
```

USAGE:

```
The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.
```

INPUTS:

```
NOAA AMSU data should be stored with the following directory
structure and file format:

  $InPath_noaa/{noaa}/YYYY/amsu_{noaa}_YYYY.DDD_ch5-12.dat

GPSRO simulated AMSU data should be stored in the format:

  $InPath_gps/{gps}/YYYY/AMSU_{gps}_atmBt_YYYY.DDD.dat
```

OUTPUTS:

```
Matched AMSU values are stored in ASCII files with the following
directry structure and file format:

  $OutPath/{noaa}_{gps}/YYYY/match_{gps}_{noaa}_YYYY.DDD.dat
```

AUTHOR:

```
Xinjia Zhou / Ben Ho (?)
```

REVISIONS:

```
Patrick Callaghan (Formatted for CDR)
```

COPYRIGHT

16. MW_Sounder_FCDR/offset_slope_multigps_aqua_month.pro [Programs]

[Top] [Programs]

NAME

offset_slope_multigps_aqua_month.pro

LOCATION

```
$SRC_DIR/Src_IDL/
```

DESCRIPTION:

```
For each month in the specified range, read in the matched
AMSU temperatures between the AQUA polar orbiter and
for all specified GPSRO missions. Calcualte the linear fit
coefficients for each month. Save the resulting monthly
values in an ASCII file for later use.
```

EXTERNALS:

```
none.
```

USAGE:

```
The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.
```

INPUTS:

```
Matched AMSU values are read from the ASCII files with the
following directory structure and file format:

  $InPath/aqua_{gps[*]}/YYYY/match_{gps[*]}_aqua_YYYY.DDD.dat
```

OUTPUTS:

```
Linear fit coefficients are written to the ASCII file:

  $OutPath/timeseries_offset_slope_aqua_gpsro.dat
```

AUTHOR:

```
Xinjia Zhou / Ben Ho (?)
```

REVISIONS:

```
Patrick Callaghan (Formatted for CDR)
```

COPYRIGHT

17. MW_Sounder_FCDR/offset_slope_multigps_noaa_month.pro [Programs]

[Top] [Programs]

NAME

offset_slope_multigps_noaa_month.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

For each month in the specified range, read in the matched AMSU temperatures between the specified polar orbiter and for all specified GPSRO missions. Calcualte the linear fit coefficients for each month. Save the resulting monthly values in an ASCII file for later use.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set directly below this header. The user needs to edit these values to process the desired values and then execute the program from within IDL.

INPUTS:

Matched AMSU values are read from the ASCII files with the following directory structure and file format:

$InPath/{noaa}_{gps[*]}/YYYY/match_{gps[*]}_{noaa}_YYYY.DDD.dat

OUTPUTS:

Linear fit coefficients are written to the ASCII file:

$OutPath/timeseries_offset_slope_{noaa}_gpsro.dat

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

18. MW_Sounder_FCDR/read_amsu_anomaly.pro [Programs]

[Top] [Programs]

NAME

read_amsu_anomaly.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

This program provides a template to be used as a guide
for reading in data from netCDF dataset generated by
the calibration algorithm.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

netCDF files are written with the following directory
structure and file format:

  $InPath/AMSU_{proc}_{chan}_V1.0_YYYYMM.nc

OUTPUTS:

NONE.... this is a template.

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

19. MW_Sounder_FCDR/read_amsu_climatology.pro [Programs]

[Top] [Programs]

NAME

read_amsu_climatology.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

This program provides a template to be used as a guide
for reading in data from netCDF dataset generated by
the calibration algorithm.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

netCDF files are written with the following directory
structure and file format:

  $InPath/AMSU_{proc}_{chan}_V1.0_MM.nc

OUTPUTS:

NONE.... this is a template.

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

20. MW_Sounder_FCDR/read_amsu_monthly.pro [Programs]

[Top] [Programs]

NAME

read_amsu_monthly.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

This program provides a template to be used as a guide
for reading in data from netCDF dataset generated by
the calibration algorithm.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

netCDF files are written with the following directory
structure and file format:

  $InPath/AMSU_{proc}_{chan}_V1.0_YYYYMM.nc

OUTPUTS:

NONE.... this is a template.

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

21. MW_Sounder_FCDR/readstd.pro [Programs]

[Top] [Programs]

NAME

readstd.pro

LOCATION

$SRC_DIR/Src_IDL/

DESCRIPTION:

IDL routine to open the given file containing Standard
Atmosphere data, read in the contents, interpolate values
to the given pressure levels, and return the results.

EXTERNALS:

none.

USAGE:

The RUN parameters specifying the data to process are set
directly below this header. The user needs to edit these
values to process the desired values and then execute the
program from within IDL.

INPUTS:

StdFile = Filename containing Std. Atmosphere data
  Ps = Pressure levels for interpolation

OUTPUTS:

Tps = Interpolated values

AUTHOR:

Xinjia Zhou / Ben Ho (?)

REVISIONS:

Patrick Callaghan (Formatted for CDR)

COPYRIGHT

22. MW_Sounder_FCDR/aqua_amsu_tran [Subroutines]

[Top] [Subroutines]

PURPOSE

```
* AQUA AMSU+HSB 101-level PLOD-based fast transmittance model
.... version of 28.10.04

* Fast (regression) model based on line-by-line transmittances
  calculated with Liebe''s model.
* Input temperatures and water-vapor mixing ratios must be
  defined at the pressure levels in array 'pstd'
  (see 'block data stdatmos').

* Logical units 51 & 52 are used for coefficient files.
```

INPUT:

```
          temp = temperature profile (deg K)
          wvmr = water-vapor profile (g/kg)
         theta = local (spacecraft) zenith angle (deg)
         kanal = channel number:  1-15 for AMSU-A
                                 16-19 for HSB
```

OUTPUT:

```
          taut = total transmittance profile
     in common/taucom/
          taud = dry (O2)  transmittance profile
          tauw = wet (H2O) transmittance profile
```

23. MW_Sounder_FCDR/calpmw [Subroutines]

[Top] [Subroutines]

PURPOSE:

```
    Routine to calculate the predictors for the dry
    (temperature), and wet components of a fast transmittance
     model for a scanning satellite based microwave instrument.
  REFERENCES:
    AIRS FTC package science notes and software, S. Hannon and
      L. Strow, Uni. of Maryland, Baltimore County (UMBC)
  CREATED:
    19-Sep-1996 HMW
```

ARGUMENTS:

```
      Input
    -----------
   t_avg_ref  -REAL reference layer average temperature array (K)
  amt_wet_ref -REAL reference water vapour amount array (k.mol)/cm^2
```

```
      t_avg     -REAL layer average temperature array (K)
     amt_wet    -REAL water vapour amount array (k.mol)/cm^2
      p_avg     -REAL layer average pressure array (mb)
    sec_theta   -REAL secant of the zenith angle array
     n_layers   -INT Number of atmospheric layers
    n_dry_pred  -INT number of dry (temperature) predictors
    n_wet_pred  -INT number of water vapour predictors

        Output
      -----------
      pred_dry  -REAL dry gas (temperature) predictor matrix
      pred_wet  -REAL water vapour predictor matrix

    COMMENTS:
      Levels or Layers?
      -----------------
        Profile data is input at a number of *LAYERS*.
      Layer Numbering pt. A
      ---------------------
        Layer 1   => Atmosphere between LEVELs 1 & 2
        Layer 2   => Atmosphere between LEVELs 2 & 3
        Layer L-1 => Atmosphere between LEVELs L-1 & L

      Layer Numbering pt. B
      ---------------------
        For the HIS instrument, Layer 1 is at the top of the
        atmosphere and Layer L-1 is at the surface.

      Layer Numbering pt. C
      ---------------------
        In this routine the number of *LAYERS* is passed in the
          argument list, _not_ the number of LEVELS.  This was
          done to improve the readability of this code,
        i.e. loop from 1->L(ayers) rather than from 1->L(evels)-1.
```

24. MW_Sounder_FCDR/conpmw [Subroutines]

[Top] [Subroutines]

PURPOSE:

```
      Function to convert atmospheric water vapour (g/kg) profile
        specified at n_levels layer BOUNDARIES to n_levels-1
        integrated layer amounts of units (k.moles)/cm^2 for use in
           microwave fast transmittance models. The average LAYER
        pressure and temperature are also returned.
    REFERENCES:
      AIRS LAYERS package science notes, S. Hannon and L. Strow,
        Uni. of Maryland, Baltimore County (UMBC)
    CREATED:
      19-Sep-1996 HMW
```

ARGUMENTS:

```
       Input
      --------
         p     - REAL*4 pressure array (mb)
         t     - REAL*4 temperature profile array (K)
         w     - REAL*4 water vapour profile array (g/kg)
      n_levels - INT*4 number of elements used in passed arrays
       i_dir   - INT*4 direction of increasing layer number
              i_dir = +1, Level(1) == p(top)         } satellite/AC
                          Level(n_levels) == p(sfc)  }    case
              i_dir = -1, Level(1) == p(sfc)         } ground-based
                          Level(n_levels) == p(top)  }    case
       Output
      --------
       p_avg   - REAL average LAYER pressure array (mb)
       t_avg   - REAL average LAYER temperature (K)
       w_amt   - REAL integrated LAYER water vapour amount
```

```
                array (k.moles)/cm^2

    ROUTINES:
      Subroutines:
      ------------
        gphite  - calculates geopotential height given profile data.
      Functions:
      ----------
        NONE
    COMMENTS:
      Levels or Layers?
      -----------------
        Profile data is input at a number of *LEVELS*.
          Number densitites are calculated for *LAYERS* that are
          bounded by these levels. So, for L levels there are
          L-1 layers.
      Layer Numbering
      ---------------
        Layer 1   => Atmosphere between LEVELs 1 & 2
        Layer 2   => Atmosphere between LEVELs 2 & 3
        Layer L-1 => Atmosphere between LEVELs L-1 & L
```

25. MW_Sounder_FCDR/gphite [Subroutines]

[Top] [Subroutines]

PURPOSE:

```
    Routine to compute geopotential height given the atmospheric
      state. Includes virtual temperature adjustment.
  CREATED:
   19-Sep-1996 Received from Hal Woolf, recoded by Paul van Delst
   18-May-2000 Logic error related to z_sfc corrected by Hal Woolf
```

ARGUMENTS:

```
      Input
      --------
        p     - REAL*4 pressure array (mb)
        t     - REAL*4 temperature profile array (K)
        w     - REAL*4 water vapour profile array (g/kg)
      z_sfc   - REAL*4 surface height (m).  0.0 if not known.
     n_levels - INT*4 number of elements used in passed arrays
      i_dir   - INT*4 direction of increasing layer number
            i_dir = +1, Level(1) == p(top)         } satellite/AC
                        Level(n_levels) == p(sfc)  }    case
            i_dir = -1, Level(1) == p(sfc)         } ground-based
                        Level(n_levels) == p(top)  }    case
      Output
      --------
        z     - REAL*4 pressure level height array (m)
  COMMENTS:
    Dimension of height array may not not be the same as that
      of the input profile data.
```

26. MW_Sounder_FCDR/intrte_mws [Subroutines]

[Top] [Subroutines]

DESCRIPTION

```
  * Integrate RTE for a microwave sounder with surface reflection
  .... version of 28.10.04
```

```
Inputs:
     tran = profile of transmittance (~1.0 at top)
     temp = profile of temperature   (deg K)
     tsfc = surface temperature      (deg K)
     esfc = surface emissivity       (0.0 --> 1.0)
     lsfc = surface level number     (101 = bottom [1100mb])
Outputs:
     tbbd = direct (upwelling) component
     tbbr = reflected component
     tbbs = surface component
     tbbt = total brightness temperature
```

27. MW_Sounder_FCDR/scan2zen [Functions]

[Top] [Functions]

FUNCTION

```
* Convert scan (nadir) angle to local zenith angle
  for the given altitude
  .... version of 21.05.03

* Inputs:
    altkm = altitude in kilometers
    scang = scan angle in degrees

* Output (function name):
          = local zenith angle in degrees
```