University of Michigan Space Physics Research Laboratory		
RETRIEVE Design and Maintenance	CAGE No.	0TK63
	Drawing No.	055-4278
	Project	TIDI
	Contract No.	NASW-5-5049
	Page	1 of 10

REVISION RECORD

Rev	Description	Date	
	Initial Release	22 Nov 2003	M Cooper

Contents

1. References

1) Gell, D. A., "TIDI Line of Sight File Format", TIDI File 055-4191.

2) Gell, D. A., "RETRIEVE Requirements Specification", TIDI File 055-3533.

3) Burek, M., "API for a C based library for accessing TIDI Level 0 Data", TIDI File 005-3696.

4) Burek, M., "TIDI Packet 0 Spreadsheet", TIDI File 055-3693.

2. Introduction

The purpose of this document is to describe the RETRIEVE program, so that a maintenance programmer can more easily modify the program. A secondary goal is to provide a detailed description of the algorithm used in RETRIEVE.

3. Program Structure

3.1 Overview

3.2 Functional Calling Tree

Here are the details of which routines/functions call/are called by other routines/functions.

Main calls:

```
READ_L0_FILE – opens and reads L0 file
OPEN_CPF
BUILD_LOS_FILENAME
OPEN_LOS_FILE
OPEN_PVAT_FILE
READ_L0_FILE
READ_SCAN_TABLE
READ_BINNING_TABLE
STORE_L0_DATA
VALIDATE_SPECTRUM
OPEN_PVAT_FILE
if ( .NOT. SPEC_CORRECTIONS ) then
   REMOVE_BACKGROUND
   NORMALIZE_SPECTRUM
   CORRECT_SENSITIVITY
endif
GET_TP_SC_POS
GET_TRACK_ANGLE
VIEWING_GEOMETRY – for spacecraft position
VIEWING_GEOMETRY – for tangent point positions
WRITE_LOS
CLOSE_PVAT_FILE
CLOSE_LOS_FILE
CLOSE_PVAT_FILE
CLOSE_CPF
CLOSE_L0
```

```
READ_L0_FILE calls:
   various level 0 access routines

OPEN_CPF calls:
   NF_OPEN (netCDF access routine)

OPEN_LOS_FILE calls:
   DEFINE_LOS
   NF_OPEN (netCDF access routine)

OPEN_PVAT_FILE calls:
   NF_OPEN (netCDF access routine)

READ_L0_FILE calls:
   various level 0 access routines

READ_SCAN_TABLE calls:
   NEXT_VALUE

READ_BINNING_TABLE calls:
   READ_CPF

VALIDATE_SPECTRUM calls:
   READ_CPF

REMOVE_BIAS calls:
   READ_CPF
   INTERP

REMOVE_DARK_COUNTS calls:
   READ_CPF
   INTERP

REMOVE_BACKGROUND calls:
   READ_CPF
   READ_L0_FILE
   VALIDATE_SPECTRUM
   REMOVE_BIAS
   REMOVE_DARK_COUNTS

NORMALIZE_SPECTRUM calls:
   READ_CPF
   INTERP

CORRECT_SENSITIVITY calls:
   READ_CPF
   INTERP

GET_TP_SC_POS calls:
   READ_CPF
   PVAT_INTERP
   ADT_EVAL_XFORM (UARS routine)
   OA_LIMB_CALC (UARS routine)
   PLEPH (ephemeris access routine)
   OA_SZA_LST (UARS routine)

GET_TRACK_ANGLE calls:
   READ_CPF
```

```
   READ_ORBIT_FILE

VIEWING_GEOMETRY calls:
   PLEPH (ephemeris access routine)

PVAT_INTERP calls:
   READ_CPF
   READ_NETCDF_VARIABLE
   CLOSE_PVAT_FILE
   ADD_ONE_DAY
   OPEN_PVAT_FILE
   READ_PVAT_FILE

WRITE_LOS calls:
   Various netCDF access routines

CLOSE_CPF calls:
   NF_CLOSE (netCDF access routine)

CLOSE_LOS_FILE calls:
   NF_CLOSE (netCDF access routine)

CLOSE_PVAT_FILE calls:
   NF_CLOSE (netCDF access routine)

READ_CPF calls:
   READ_NETCDF_VARIABLE

READ_PVAT_FILE calls:
   READ_NETCDF_VARIABLE

READ_NETCDF_VARIABLE calls:
   various netCDF access routines

DEFINE_LOS calls:
   various netCDF access routines
```

3.3 Program Modules.

The program is contained in the following files: (the directory is /tidi/tidi_software/retrieve/ unless otherwise given)

`retrieve.f` - calls nearly all the routines, except as detailed below

`TIDI_L0_SCIENCE.INC` - defines the level 0 science structure

`TIDI_L0_TEMPS.INC` - defines the level 0 temperature structure

`TIDI_LOS.INC` - defines the LOS file's record structure

`l0_file.f` – read in the L0 file and store in a L0 structure

`store_l0_data.f` – stores the 'pass through' data from the L0 record. These are variables that are not calculated from any values, but are simply passed on to the LOS record.

`adt_eval_xform.f` - ADT_Eval_Xform

`oa_limb_calc.f` - OA_LIMB_CALC

`interp.f` – interpolation routine used in the spectral corrections subroutines

`validate_spectrum.f` - VALIDATE_SPECTRUM

`remove_bias.f` - REMOVE_BIAS

`remove_dark_counts.f` - REMOVE_DARK_COUNTS

`remove_background.f` – removes the background from the spectrum

`normalize_spectrum.f` - NORMALIZE_SPECTRUM

`correct_sensitivity.f` - CORRECT_SENSITIVITY

`get_tp_sc_pos.f` – gets the spacecraft positions and tangent points

`get_track_angle.f` – calculates the track angle of the spacecraft and the tangent points

`los_define.f` – defines the NetCDF LOS file and contains the subroutine that writes to the NetCDF file

`los_file.f` – contains subroutines that invents the NetCDF LOS filename, and opens and closes the NetCDF LOS file. Also contains the subroutine check_err which is mainly used in los_define.f. These are the subroutines associated with the LOS file that are not automatically generated using the ncgen command on the tidi_los.cdl file.

`pvat.f` – contains subroutines that open, read, and close the PVAT file

`pvat_interp.f` – interpolates the PVAT file data based on a given mission time

`read_orbit_file.f` – reads in the orbit file – essential in the calculation of track angle

`viewing_geometry.f` - viewing conditions (solar zenith angle, etc.) for the current tangent point are calculated

`/tidi/tidi_software/cpf/cpf_access.f` - CPF access routines

`/tidi/tidi_software/ephem/ephem.f` – contains the ephemeris access routines, used to get the solar and lunar positions

4. Theory of Operation

The RETRIEVE program reads TIDI level 0 files, processing the science data contained in them into a series of line-of-sight measurements of the wind speed, temperature, and other quantities. These results, along with the collection conditions for the measurements, are stored in a TIDI level 1B file, the LOS output file (one file for each run of the program). The level 1B files or LOS files are in the NetCDF format, in which a description of the file's format is included in the file, which makes them more easily transportable and readable. There are several commercially available packages, include IDL (which is available on the TIDI system), which have easy to use interfaces to NetCDF files. The operation of the program is summarized in the following pseudo-code:

```
get the filename of data to process
open required files
while (not end of current file and start and end time of chosen period
is still with range, and all else is ok)
  read next record from level 0 files
  if (scan table has changed) read new scan table
```

```
  if (binning table has changed) read new binning table
  for each field of view in the sample (there can be five per sample)
     validate spectrum and collection conditions (and initialize
       output structure)
     Correct spectrum for instrument effects
       remove background
       normalize spectrum
       correct sensitivity
     Calculate tangent point
     Calculate the viewing conditions
     Recover line of sight quantities
     Write out LOS record
  end for !for each of the 4 spectra
end while !time of current sample is within range and all else is ok
close files
```

The above outline the entire program, with each of the 'action' lines being described in more detail below, in section 3.1, Program Structure.

4.1 Program Structure.

The following sections provide a more detailed expansion of the pseudo-code above.

4.1.1 get the filename of Level 0 data to process

The user must enter a filename to process, using the command switch '-i'. All of the science data for this day processed into a level 1B file, LOS file.

4.1.2 open required files

The file specified on the command line is opened. Three additional files are also opened: the constant parameter file (CPF) (which contains the constants which describe the instrument's various states and operations), the position-velocity-attitude-time file (PVAT), which includes describes the satellite's position and orientation, and the level 1B file (or line-of-sight (LOS)) file, which is the result of the processing.

4.1.3 read next record from level 0 files

The next level 0 record is then read (this include both the next science packet, and the housekeeping packet which is closest in time to the science packet - see reference #3 and #4).

4.1.4 if (scan table has changed) read new scan table

Each packet contains data from 1 to 5 fields of view of the instrument, as well as the values which are required to describe the data. One of these descriptive values is the 'scan table id', which denotes which 'scan table' was in operation when the data sample was taken. Each scan table can include a number of steps (at various filter wheel settings and telescope elevations) at which samples are taken. One can think of the scan table as the 'program' for the instrument. The whole of this program needs to be read and understood by the RETRIEVE software, in order to correctly process the samples. Here, if the current sample has a different scan table id than the prior sample, the new scan table is read.

4.1.5 if (binning table has changed) read new binning table

Each science packet is made up of from 1 to 255 'bins', which are individual measurements of the electrons in a number of pixels on the TIDI CCD. Each of the four telescopes on TIDI and the 'calibration input' can be used to focus photons onto separate regions (or 'fields of view') of the CCD. The number and location of the pixels summed into each bin, and the gain with which the bin is sampled, are detailed in a 'binning table'. The binning table is stored in a file, and associated with a 'binning table id', which is one of the values which describe each data packet. Here, if the current sample has a different binning table id than the prior sample, the new binning table is read in, since the RETRIEVE program needs to know the binning table in order to process the sample correctly.

4.1.6 validate spectrum and collection conditions (and initialize output structure)

Each science packet can contain bins from one or more field of view. The bins from a single field of view (together called a spectrum), are tested here to flag spectra which are possibly contaminated in one of several ways (incorrect instrument operation or mechanical fault, cosmic ray hits on the CCD, overfilling of the pixels or bins).

4.1.7 remove bias

Each spectrum can be characterized as a 'real' signal (which contains the information from the atmosphere being sampled) plus a series of signals which are the result of instrumental effects, which are not directly related to the atmosphere being viewed. One of the instrumental signals is the electronic bias, which is a result of the sampling process, and can be thought of as a signal level added to each of the samples. Here the bias, which is modeled as a function of the binning table and gain (among other things), is subtracted from the measured signal. The modelled bias values are read and/or interpolated from the CPF.

4.1.8 remove dark counts

Another instrumental-effect signal is the dark level (or dark counts). Here this signal, which is characterized as a function of the CCD temperature (and other indicators of the instrument's status), is subtracted from the bias-corrected signal. Again, this characterization is read from the CPF.

4.1.9 remove background

Once a spectrum is corrected for bias and dark counts, the next 'signal' to be removed is the background, which is a result of electrons (and other charged particles) which strike the detector. This signal varies widely with the position of the spacecraft (latitude/longitude/altitude) and also changes from day to day. As a result this signal cannot be modelled easily, and here we rely on measurement of the background, made with the telescope pointed at a high altitude and filters 'crossed' (non-overlapping bandwidths). The background, which is measured every scan (or so) is saved, and subtracted from the corrected signal here.

4.1.10 normalize spectrum

With the instrumental and background signals removed from the measurements, the spectrum is 'normalized' here, which adjusts for the bin to bin differences in sensitivity of the instrument. The normalization values, read from the CPF and interpolated for the current conditions, are numbers from 1.0 to 0.0, and are divided into the corrected spectrum.

4.1.11 correct sensitivity

The normalized spectrum is divided by the overall sensitivity of the instrument (for the current collection conditions), to change the spectrum from counts to Rayleighs per inverse centimeter (R/cm-1). This sensitivity again is read from the CPF.

4.1.12 calculate tangent point

The PVAT file is read and interpolated for the current sample time, giving the spacecraft's position, velocity, and attitude. These are used, along with the telescope elevation and azimuth angles, to determine the direction of each line of sight, and then the position of the point at which this line of sight is closest to the surface of the earth, also called the tangent point.

4.1.13 calculate viewing conditions

The viewing conditions for the current tangent point (solar zenith angle, etc.) are calculated here.

4.1.14 recover line of sight quantities

The line of sight wind, temperature, and other values are recovered here, using the corrected spectrum, collection conditions, and a model of the instrument and atmosphere.

4.1.15 write out LOS record

The LOS record, which holds the current spectrum, collection conditions, and line of sight quantities, is written out to the LOS NetCDF file.

4.1.16 close files

The CPF, PVAT, LOS, and level 0 files are closed here.

4.2 Running RETRIEVE.

The RETRIEVE program can be run by setting default (cd) to the directory `/tidi/tidi_software/retrieve` and issuing the command 'retrieve'. The level 0 filename must be specified using the '-i' command line parameter. Here is a list of the command line parameters that can be used:

'-i <filename>'	Level 0 input filename
'-o <filename>'	forcing output LOS filename
'-s <missiontime>'	start time of the data to process
'-e <missiontime>'	end time of the data to process
-'c'	do not do any spectral corrections (REMOVE_BIAS, REMOVE_DARK_COUNTS, NORMALIZE_SPECTRUM, and CORRECT_SENSITIVITY)
'-k'	overwrite the output file without asking, is exists
'-D'	use default debug directory for output file, /tidi/tidi_software/retrieve/debug/YYYY/filename

'-debug'	printout debug messages (warning - can be lots of output)
'-verbose'	printout messages in verbose mode (lots of output – beware)

4.3 Building RETRIEVE.

The RETRIEVE program can be built by setting default (cd) to the directory `/tidi/tidi_software/retrieve` and issuing the command 'make'.

5. Maintenance