CORONAL DIAGNOSTIC SPECTROMETER
SoHO

CDS SOFTWARE NOTE No. 26

Version 3.0 5 December 1995

CDS ENGINEERING TELEMETRY MONITOR

C D Pike
Rutherford Appleton Laboratories
Chilton, Didcot
Oxon OX11 0QX

cdp@astro1.bnsc.rl.ac.uk

1 Introduction

The CDS science stream telemetry contains two packets of engineering data (the so-called Engineering A and Engineering B packets). The contents of these packets are defined by the telemetry database and each element is identified by a mnemonic-type name. A reformatted and more easily readable form of the database is held in file $CDS_ENG_DATA/new_database.txt.

2 Running the monitor

EMON may be run in either interactive or non-interactive (batch) mode.

2.1 INTERACTIVE

The interactive program to monitor the engineering information works only from the telemetry files created every hour by the EGSE software. It is started by the command:

IDL> emon

If necessary, EMON can take parameters which allow the extracted data to be returned to the IDL environment for further processing (use XDOC for an explanation).

For example a typical interactive session would be started by the command:

IDL> emon, a, b, c

Here, parameter 'a' would return a vector of dates and times (CDS string format), parameter 'b' would be a 2-d array (p,N) where p is the number of telemetry parameters extracted and N is the number of data points found. Parameter 'c' is a string vector holding the names of the selected parameters.

Note that in interactive, as well as batch, mode EMON can take the keyword FNAME. If this is given a value, it is assumed to be the file name (or array of file names) from which telemetry is to be read. This saves having to select the telemetry files in EMON itself.

Within EMON the sequence of user actions should be as follows:

•Choose whether you want the engineering A or B data (they are mutually exclusive).

•Select the parameter(s) to be read from the list presented.

•Unless already given on the command line, select a telemetry file to read. Note that multiple files may be selected, the program will fall off the end of one file onto the next (they should be in chronological order though).

•Start the reading of the file. It is possible to stop the reading process at any time by clicking on the stop button.

•Select which parameter(s) to plot (from the list of those already read). The data for the chosen parameters are plotted when you hit the 'Plot' button. If a shorter time span is required then use the 'Zoom' button. After hitting that button, define the required range using two cursor clicks and

'Plot' the data again. Note that a hard copy of the plot(s) will also be produced if that mode is selected via the toggle button.

2.2 NON-INTERACTIVE (BATCH)

For non-interactive use the keyword BATCH must be given and the data file names and parameters to be selected must be supplied on the command line. It is anticipated that, used in this mode, EMON will be packaged within procedures to extract and plot pre-defined sets of parameters.

A typical call might be:

IDL> files = findfile('tm.*354*') ; find all telemetry files for day 354

IDL> parms = ['btm3','bkcfc','bsmmht8'] ; define parameters wanted

IDL> emon,a,b,c,fname=files,pname=parms,/batch

The variables a,b,c would then return the uncalibrated data for those parameters as in interactive mode. Note that, as in interactive mode, the parameters selected in a single call must all be from the "same" packet type ie. Eng-A or Eng-B.

The data may be plotted by a simple UTPLOT call eg.

IDL> utplot, str2utc(a,/dmy), b(0,*), ytitle=c(0)

for parameter BTM3.

Note that the parameter 'a' returns the date/time in string format and this must be converted for use in UTPLOT. It is probably easiest if that conversion takes place outside the call to UTPLOT. ie

IDL> a = str2utc(a,/dmy)

and then use variable 'a' as shown below.

3 Calibrating and plotting the data

When in interactive mode, the data (or most of them) are calibrated before being plotted (but note that raw data are still returned in the output variables).

The raw data returned by EMON, in either mode, may be calibrated using the procedure TM_CALIB. This function returns calibrated data and some strings for labelling any subsequent plot. Only calibration types (0,1 and 8) are catered for at present.

Thus, using the example above the data for BTM3 could be calibrated and plotted as follows:

IDL> d = tm_calib(c(0), b(0,*), words)

The returned string vector contains the plot labels. If the first element is 'title' then the parameter is a normal numeric type and the y-axis could be labelled with the second element of word (ie ytitle=word(1)). If the first element is not 'title' then the parameter is an ON/OFF type parameter and the first 2 elements of word contain the strings which could label the values 0 and 1 in the data array.

Note that the routine plot_tm_calib can be used to plot the data output by the calibration routine. That will give a 'standard' plot as in EMON itself. However, if you wish to have more control over the plotting, the plotting can be done on an individual basis. For plotting ON/OFF type parameters, for instance, it is best to set

yrange=[-1,2]

ytickname=[' ',word(0),word(1),' '] and

ytitle=word(2).

Hence a decent plot of the BTM3 parameter could be produced by:

IDL> utplot,a,d,ytitle=word(1),yrange=[?,?],psym=8,tit='Parameter '+c(0)

or for the status-type variable BSMMHT8, having run tm_calib again, then use something like:

IDL> utplot,a,d,yrange=[-1,2],ytit=word(2),ytickn=[' ',word(0),word(1),' ']

4 Packaged versions of EMON

Using the batch mode of EMON it is possible to package up the retrieval and plotting of well-defined sets of telemetry parameters. This has been done in two examples so far, one for the heater modes (which given the sampling problem may or may not be useful!) and one for the CDS operational mode.

See the documentation on the programs EMON_MAIN_HTRS, EMON_LUT and EMON_CDS_MODE for details of how to run them.

5 Maintenance of database

In order to speed the operation of EMON and TM_CALIB, the telemetry database is stored in an IDL save file ($CDS_ENG_DATA/tm_dbase.save). If the text version of the database file is ever updated (the master is kept on the EGSE machines as file tmdb_ascii.txt) then perform the following actions.

1) Copy the latest version of tmdb_ascii.txt to the file $CDS_ENG_DATA/tmdb.txt

2) Delete the file $CDS_ENG_DATA/tm_dbase.save

3) Run IDL from the $CDS_ENG_DATA directory (from an account that has write privilege) and run procedure tm_new_db. This will create a new text version of the database (file new_database.txt).

4) Run the procedure get_mask_etc with an output parameter called 'array' ie.

IDL> get_mask_etc, array

5) This output variable should then be SAVEd in a save file called $CDS_ENG_DATA/tm_dbase.save ie.

IDL> save,array,file='tm_dbase.save',/xdr

Note that if further calibration files (calibration type 6) are added to the $CDS_ENG_DATA directory which contain lists of strings with which to label plot axes, as opposed to numerical data, then the files tm_read_cal.pro and tm_calib.pro must be edited and new values given for the variable text_list.

Note also that if NEW parameter names are ever added to a revised version of the tmdb_ascii.txt file then not only will the update procedure described above be necessary but it will also be necessary to update the structure definition in ENG_STRUCT.PRO to reflect the changes.