THEMIS / SCM

IDL procedures to calibrate waveform data

P. Robert, 2006 December 6 -- VERSION 1.0 --

SUMMARY

THEMIS / SCM
IDL procedures to calibrate waveform data

P. Robert, 2006 December 6 -- VERSION 1.0 --

I. REMINDER ON SCM CALIBRATION METHOD

A) Summary of the generic wave calibration method

This method has been used to calibrate the ULF (0.1-11.5 Hz) GEOS data (1977-1983), and then used for other mission/experiment which provides waveform data having a non linear frequency transfer function. This was the case for the search coils of STAFF-SC experiment of the CLUSTER mission, and then here for THEMIS-SCM. Coordinates systems nomenclature is here the one used by CLUSTER; correspondence with THEMIS convention are given in chapter I.B.

More details on this method can be found in ref :

[1] **"Waveform calibration method for no-linear transfer function"**, *Patrick Robert, to be written*

Application on GEOS search coils data, encountered problems and practical solutions are described with details in following document (in French) :

[2] **"Intensité et polarisation des ondes UBF détectées à bord de GEOS-1"**,"Méthode d'analyse numérique du signal et production en routine de sommaires expérimentateurs. Problèmes rencontrés et solutions pratiques." *Patrick Robert, CRPE/ETE, Note technique CRPE/NT/71, Mai 1979.*

The calibration method presently works on a series of consecutive data window, each of it having no data gap. To calibrate a waveform data window, the N points taken must be as N=2**m, with m integer, because we use a Fast Fourier Transform (FFT). This leads to get a series of calibrated data windows, but these windows cannot be merged, because of the discontinuities introduced on the edges of the windows by the FFT. This is an important restriction of this method.

Notice that a new method, allowing production of continuous calibrated waveforms, is currently under development, but not yet fully tested and applicable. A lot of work remains to do before validation.

The first classic method, applied on a series of window data of N points, is described in § 1 to 5.
Names of coordinate systems use the CETP/CLUSTER conventions; correspondence with THEMIS conventions is given in chapter I.B.

1. Production of "clean" raw waveforms in spinning Sensor Coordinate System (SCS)

This preliminary step is required because the spacecraft rotation in the local high DC magnetic field makes a high level signal (more that 100 time the useful AC field) at the spin frequency in the wave TM data. Sample Frequency Fe and spin frequency Fs are required.

Telemetry data In TM counts i=1,N	*conversion to volts and step amplificatory factors*	Raw waveforms in Volts	*subtract sinusoidal wave at spin frequency by 'desinus' soft*	Clean raw waveforms without big spin frequency in Volts
$xo(t_i)$ $yo(t_i)$ $zo(t_i)$	$\longrightarrow$	$x1(t_i)$ $y1(t_i)$ $z1(t_i)$	$\longrightarrow$	$x2(t_i)$ $y2(t_i)$ $z2(t_i)$

2. Calibration of each component in frequency domain

Calibration table (complex values, modulus and phase, in Volt/nT, versus frequency), possible sample delay (between x, y, z components) and low cut-off frequency (low significant value of the calibration tables) are required. Calibration is done window per window.

Clean raw waveforms without big spin frequency in Volts	*Transformation of time series in frequency series By FFT*	Spectrum in Volts	*(i) Correction of transfer function by *1/G(f_i) (ii) correction of possible sample delays between x,y,z (iii) cut-off at low frequency*	Calibrated spectrum, Sensor Coordinate System in nT
$x2(t_i)$ $y2(t_i)$ $z2(t_i)$	$\longrightarrow$	$\mathbf{X}2(f_i)$ $\mathbf{Y}2(f_i)$ $\mathbf{Z}2(f_i)$	$\longrightarrow$	$\mathbf{X}3(f_i)$ $\mathbf{Y}3(f_i)$ $\mathbf{Z}3(f_i)$

3. Time series calibration in spinning Sensor Coordinate System (SCS)

Return to time domain by a simple inverse FFT (so there is no continuity between successive windows)

Calibrated spectrum, sensor spinning system, in nT	*Transformation of frequency serie in time serie By FFT^{-1}*	Calibrated waveform, Sensor Coordinate System, in nT
$\mathbf{X}3(f_i)$ $\mathbf{Y}3(f_i)$ $\mathbf{Z}3(f_i)$	$\longrightarrow$	$x3(t_i)$ $y3(t_i)$ $z3(t_i)$

4. Waveform transformation from SCS to GSE System

Data are calibrated in the spinning Sensor Coordinate System, characteristic to each experiment. For a scientific use of the data, it must be converted into a known and convenient system, such as the Geocentric Solar Equatorial system (GSE). This operation requires the knowledge of a suit of matrix :

a) Sensor Coordinate System to Orthogonal Sensor System(OSS)

The original sensor system can be a non orthogonal system, the first step is to transform the data vector in an orthogonal coordinate system; Z axis being the reference of the new Orthogonal Sensor System. The corresponding matrix, close to an unit matrix, is required; values are supposed to be constant.

Calibrated waveform, Sensor Coordinate System SCS	*Constant matrix [A]* *SCS_to_OSS*	Orthogonal Sensor System OSS
$x3(t_i)$ $y3(t_i)$ $z3(t_i)$	$\longrightarrow$	$x3_A(t_i)$ $y3_A(t_i)$ $z3_A(t_i)$

b) Orthogonal Sensor System to Body Build System (BBS)

The Body Build System (BBS) is a system fixed to the geometry of the spacecraft, and is used as the spacecraft system reference for all the experiments. Generally, the Z axis is close to the maximum principal inertia axis, also called the spin axis. The Z axis of the Orthogonal Sensor System is generally close to the Z axis of the BBS axis, but the 2 others axis may be rotated by an important angle. The corresponding matrix is required; values are supposed to be constant.

Orthogonal Sensor System OSS	*Constant matrix [B]* *OSS_to_BBS*	Body Build System BBS
$x3_A(t_i)$ $y3_A(t_i)$ $z3_A(t_i)$	$\longrightarrow$	$x3_B(t_i)$ $y3_B(t_i)$ $z3_B(t_i)$

c) Body Build System to Spin Reference System (SRS)

The Spin reference system has its Z axis parallel to the spin axis ; This is a spinning system, rotating at the spin frequency. It can be exist a small angle between the Z axis of the BBS and the spin axis. This angle could be constant, but can have also small variation during operation on the spacecraft (trajectory modifications, etc.). The corresponding matrix is required.

Body Build System BBS	*Constant matrix [C]* *BBS_to_SRS*	Spin Reference System SRS
$x3_B(t_i)$ $y3_B(t_i)$ $z3_B(t_i)$	$\longrightarrow$	$x3_C(t_i)$ $y3_C(t_i)$ $z3_C(t_i)$

d) Spin Reference System to Spacecraft-SUN System (SSS)

The SSS system is derived from the SRS system by a despot operation; The spinning Spacecraft is " stopped" just at the time where the X axis is in the plane containing the Z spin axis and the direction of the Sun. The rotation angle require the Sun pulse or any other quantity to compute the spin phase angle, which can be provided at least one time per spin period. This angle, and the corresponding time measurement, is required to build the corresponding matrix. Terms of this matrix are fastly varying with time.

Spin Reference System SRS	*Fast varying matrix [D]* *SRS_to_SSS*	Spacecraft SUN System SSS
$x3_C(t_i)$ $y3_C(t_i)$ $z3_C(t_i)$	$\longrightarrow$	$x3_D(t_i)$ $y3_D(t_i)$ $z3_D(t_i)$

e) Spacecraft-SUN System to Geocentric Solar Ecliptic System (GSE)

The GSE system is a well known system, with the Z axis perpendicular to the Ecliptic plane, and the X axis toward the Sun. To do the transformation of the SSS to the GSE, the direction of the spin axis in the GSE system is required. Due to the gyroscopic effect of a spinning spacecraft, the spin axis is constant in an inertial system (but could be slightly changed during operations), and so have a yearly variation in the GSE system, excepted during spacecraft operations.

The two angles, generally right ascension and declination, and the corresponding time measurement, is required to build the corresponding matrix.

Spacecraft SUN System SSS	*Low varying matrix [E]* *SSS_to_GSE*	Geocentric Solar Ecliptic System GSE
$x3_D(t_i)$ $y3_D(t_i)$ $z3_D(t_i)$	$\longrightarrow$	$x3_E(t_i)$ $y3_E(t_i)$ $z3_E(t_i)$

5. Transformation from GSE to another Geocentric System

The GSE system being a well known system, any other transformation to Geocentric Solar Magnetospheres system (GSM), Geocentric Equatorial Inertial system (GEI), Solar Magnetic system (SM), Geographic system (GEO), Geomagnetic system (MAG), Dipole Meridian system (DM), and Vertical Dusk Horizontal system (VDH), can be done by the ROCOTLIB coordinate library transformation written for CLUSTER mission where all this systems are defined, with the corresponding matrix to pass from one to another. For details, see :

[3] **"CLUSTER SOFTWARE TOOLS, Part I, Coordinate transformation library"** *, by Patrick ROBERT, RPE/TID, DT/CRPE/1231, Centre de Recherche en Physique de l'Environnement Terrestre et Planétaire, CNET-CNRS, July 1993.*

and more recently :

[4] **"ROCOTLIB: a Coordinate Transformation Library for Solar-Terrestrial studies"** , *Version1.7, by Patrick ROBERT, Centre de Recherche en Physique de l'Environnement Terrestre et Planétaire, Internal report n° RI-CETP/02/2003.*

Complete documentation is available on http://cdpp.cesr.fr */View and recorder /libraries/the Rocotlib software/,* as the Fortran code in 1.7 version.

A new version 2.0 of November 2006, with dipole direction computed until year 2010, has been developed both in Fortran 77 as well as IDL. Codes are available on request from CETP (patrick.robert@cetp.ipsl.fr).

B) THEMIS and CLUSTER coordinate systems

1. Equivalence between THEMIS and CLUSTER conventions

Themis abrev.	THEMIS name	definition	CLUSTER name	CLUSTER abrev.
SMa	Sensor Magnetic System	original tri-axis sensors	Sensor Coordinate System	SCS
SMe	Sensor Mechanical System	referenced to the sensor mounting plate	Orthogonal Sensor System	OSS
SPG	Spinning Probe Geometric system	Z along the center tube X normal to solar panel #1 Fixed to the Spacecraft geometry ; Z close to the maximum principal inertia axis. Used as reference for all experiments	Body Build System.	BBS
SSL or SLS	Spinning SunSensor L-vectorZ	Z along the spin axis X toward the Sun Sensor	Spin Reference System	SRS
DSL	Despun Sun L-vectorZ	Z along the spin axis X is in the plane defined by the spin axis and the Sun direction	Spacecraft Sun System	SSS (SR2)

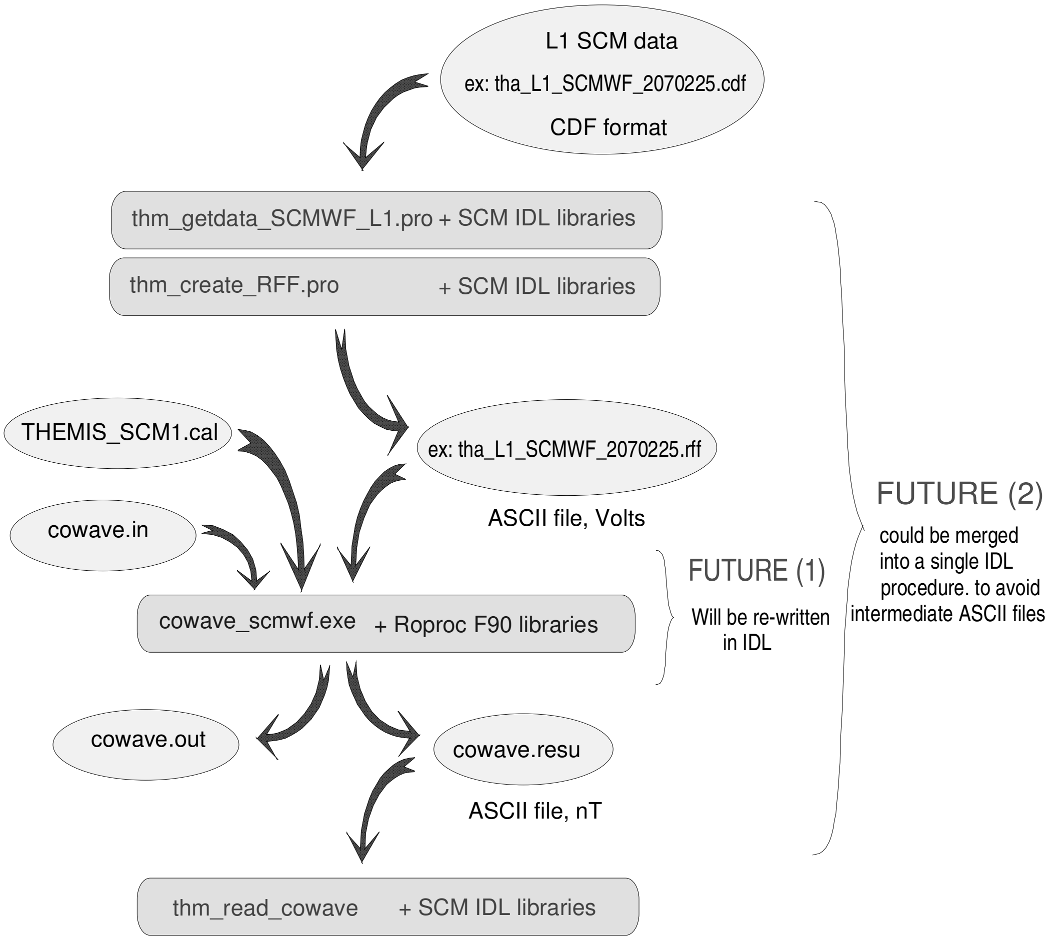
2. THEMIS transformations matrix into probe centered systems

SMa ↓ SMe	Identity Matrix (misalignment negligible)	$\begin{matrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{matrix}$
SMe ↓ SPG	12.12 deg. rotation around Z axis (see [5] **"THEMIS Science Coordinate System Definition"**, *THM-SOC-110, September 29, 2006, NAS5-02099.*	$\begin{matrix} 0{,}976 & -0{,}219 & 0 \\ 0{,}219 & 0{,}976 & 0 \\ 0 & 0 & 1 \end{matrix}$
SPG ↓ SSL	135. deg. rotation around Z axis (see [5])	$\begin{matrix} -0{,}709 & 0{,}705 & 0 \\ -0{,}705 & -0{,}709 & 0 \\ 0 & 0 & 1 \end{matrix}$
SSL ↓ DSL	rotation of the spin phase around Z axis requires Sun pulse in the STATE file and spin phase interpolation.	$\begin{matrix} \sin\phi & \cos\phi & 0 \\ \cos\phi & -\sin\phi & 0 \\ 0 & 0 & 1 \end{matrix}$

II. MAKING USE OF CALIBRATION METHOD FOR THEMIS SCM

A) Schematic diagram of SCM calibration procedure

Arrows indicate input and output files.



The calibration procedures is operating within 4 main procedures:

(1) thm_getdata_SCMWF_L1 acquires the telemetry waveform in volt, from files such as tha_L1_SCMWF_20070225.cdf

(2) thm_create_RFF create a file such as tha_L1_SCMWF_20070225.rff , with a dedicated file format used by the calibration program.

(3) cowave_scmwf.exe is an executable file, build from a Fortran 90 code, and launched by a spawn command under IDL. This is the code which do the calibration. It takes level 1 data from tha_L1_SCMWF_20070225.rff file, and requires the calibration files such as THEMIS_SCM1.cal, and calibration parameters contained in the cowave.in file. It product the cowave.resu file, which contain the level 2 calibrated data.

(4) thm_read_cowave read the level 2 data contained in the cowave.resu file.

B) IDL procedures description

1. Get SCMWF data:

The following procedure read a CDF file of SCM telemetry waveform data, and get the 3 waveform in Volt, given in the spinning sensor coordinate system. It requires also auxiliary data to extract the sampling frequency and the spin frequency.

```
PRO thm_getdata_SCMWF_L1     DataFile, datim1,datim2, SCnum, mode, times, x,y,z,
                             Fsamp, Fspin
```

input:

DataFile	:	Path of data file (ex: ./ tha_L1_SCMWF_20070225.cdf)
datim1, datim2	:	date time as 2007-02-25/10:47:52.1234 for the asked period
Scnum	:	Spacecraft number (1-5, or a-e)
mode	:	SCMWF mode (444, 448, 44c)

output:

times	:	Double precision array of time tags, representing sec from THEMIS origin (Jan 1.0, 2001)
x,y,z	:	Single precision floating point array of X,Y,Z components.
Fsamp	:	Sample Frequency, double precision, Hz, depending of used mode
Fspin	:	Spin frequency, double precision, Hz

2. Create RFF L1 data file

The following procedure creates an ascii file containing the L1 SCM data, in a dedicated format used by Search Coils generic calibration program. This format is called Roproc_File_Format, and the corresponding file extension is ".rff". Full description of this format can be found in document :

[6] **"Roproc Command Language, a set of linked commands for spatial data processing**" *, Patrick Robert, CNRS/CETP, version 2.0, November 2004.*

The RFF file is created by:

```
PRO thm_create_RFF,    DataFile,  SCnum, mode, times, x,y,z, Fsamp, Fspin
```

All arguments are **input** arguments, obtained from the previous thm_getdata_SCMWF_L1 procedure, excepted DataFile which is the RFF output file name. The DataFile name can be anything; if the DataFile name has no suffix, suffix .rff is added ; if the name is the same as previously, for instance "tha_L1_SCMWF_20070225.cdf", suffix .cdf is changed to .rff.

3. Calibration of SCMWF data

a) Program description

The calibration program is presently a Fortran executable file. It is run by IDL, under Unix system, by a spawn command as:

SPAWN, 'cowave_scmwf.exe < cowave.in > cowave.out'

5 files are used :

tha_L1_SCMWF_20070225.rff
cowave_scmwf.exe
cowave.in
cowave.out
cowave.resu

The tha_L1_SCMWF_20070225.rff file contain the SCMWF L1 data (see 2).
The cowave_scmwf.exe program read the input parameters from the cowave.in file, and an execution report is put in the cowave.out file. If execution of cowave_scmwf.exe run correctly, the following message is displayed at the end of the run:

"STOP: COWAVE_SCMWF   NORMAL TERMINATION "

At the same time, a cowave.resu file is created, and contain the calibrated waveform.

In case of abnormal execution, an error message is displayed, and the cowave.resu file is not created. In this case, the user can look at the cowave.out file, which contains the execution report, and see what is the encountered problem. In any case, this .out file may contains interesting information or warning, and can be seen at anytime; it can be useful for the users who have not a good knowledge of the SCM experiment, or of the calibration method used to process SCM data.

The cowave.in file contains the input parameters required by the corresponding .exe program.

Example is given below:

```
./tha_L1_SCMWF_20070225.rff     # Path of RFF L1 data file
./THM_SCMWF_CALFILES            # Path of directory containing calibration files
2007 02 25                      # Year-month-day of asked period
10 48 20.1234                   # Hour-min-sec   of start period (real double sec)
 4.5                            # Duration of the time period (minute)
512                             # Number of point of the calibration window (256-65536)
0.                              # Detrend frequency, 0.=classic despin
0.1                             # Frequency cut-off for calibration
0. 0.                           # Min/Max frequencies for filtering
3                               # Processing step (1-6)
444                             # Mode, ex: 444, 448 or 44c
```

A short description of calibration parameters is given in next chapter.

Full description of the calibration method and detailed explanation of the calibration parameter are given in document :

[7] "**THEMIS SCM waveform calibration method"** *, P. Robert, December 2006* (to be written).

b) Short description of calibration parameters

Year-month-day of requested period
Hour-min-sec of start period (real double sec)

This corresponds to the date time epoch of the first point of the waveform window to be processed. All are integers, except the parameter "second", which is a real double precision value.

Duration of the time period (minute)

This is the total duration of the sampled period to be calibrated. If its duration is longer than the duration corresponding to the number of point of the calibration window, several successive windows will have to be processed.

Number of point of the calibration window (256-65536)

This must be a power of 2, as N=2**m. The corresponding duration will be N/Fsamp.

Detrend frequency, 0.=classic despin

There are two ways to remove the large amplitude signal at the spin frequency:
(i) the classic despin, which run well on a window of several spin periods. The classic despin assume that the DC field does not vary too much during the window, so it is adapted to a short duration window (at least 1 period, up to a few, depending on the stability of the DC field). This option allows the computation of the X-Y components of the DC field in the spin plane.

(ii) the second option is the detrend option which means that a smoothed signal is computed from the data, and then removed. This option is useful for the calibration of long windows, or when the DC field is fastly varying. Nevertheless, all frequencies under the detrend frequency are removed, and the DC field cannot be computed in the X-Y plane.

Frequency cut-off for calibration

The transfer function of the search coils being equal to zero at zero frequency, calibration is impossible at this frequency. The Fc value correspond to the minimum frequency where the calibration can still be meaningful. A value of 0.1 Hz is a minimum value. Once this value of this minimum frequency is set by the user, a high pass filter is applied to the data, in the spinning sensor coordinate. Thus this frequency is not the same in the despun system, due to the Doppler shift. (right mode will be shifted by –Fspin, while left mode is shifted by +Fspin).

Min/Max frequencies for filtering

Possible filtering can be applied when the data are transformed into the Despun system; this is taken into account only for processing steps >4. A value of 0. for Fmax corresponds to the highest possible value, according to Nyquist frequency: Fsample/2.

Processing step (1-6)

The calibration operates on 5 successive steps :

step 1 : nothing is done, only format conversion, thus the cowave.resu will contain telemetry data in spinning sensor coordinates.

step 2 : a sine signal at spin frequency is removed from the data (despin classic), or the detrend method is applied (Fdet not equal to zero).

step 3 : the signal is calibrated in spinning sensor coordinates, by FFT, complex transfer function correction, filtering, FFT-1 (see chapter I.A.2).

step 4 : The sensor misalignment matrix is applied, and signal is computed in Spinning Sun-sensor L system (SSL).

step 5 : the signal is computed in Despun Sun L system (DSL) ; this step requires the Spin phase or Sun pulse data (see section d).

step 6 : The X-Y B DC components are added; it can be useful for comparisons with FGM. The Despin classic method must then be used.

step 7: data from step 5 (in DSL) are transformed into the GSE system; this step requires spin axis direction (right ascension and declination) in inertial system (GEI), at least one time each day (see chapter II.B.4).

Mode, ex: 444, 448 or 44c

This is the selected mode we want. If the L1 data file does not contain data in this mode, during the selected period, an error is printed in the cowave.out file.

c) Getting SCM calibrated waveform from IDL

This done by the following procedure, which read the cowave.resu file:

```
PRO thm_read_cowave    SCnum, mode, times, x,y,z, Fsamp, Fspin, , nbp, nb_win, step_num,
                       Fdet, Fcut, F1, F2,title, subtitle, anomaly
```

where all arguments are output arguments:

output:

Scnum	:	Spacecraft number (1-5, or a-e)
mode	:	SCMWF mode (444, 448, 44C)
times	:	2-Dimension Double precision array of time tags, representing sec since THEMIS origin (Jan 1.0, 2001), second dimension being the window number
x,y,z	:	2-Dimension Single precision floating point array of X,Y,Z components, second dimension being the window number.
Fsamp	:	Sample Frequency, Hz, depending on mode
Fspin	:	Spin frequency, Hz
nbp	:	Number of points in a calibration window (see cowave.in file).
nb_win	:	Number of windows of nbp points containing in the given period (2^{nd} dimension of x,y,z)
step_num	:	Step number of the calibration process
Fdet	:	Detrend frequency (Hz)
Fcut	:	Cut-off frequency (Hz)
F1,F2	:	Optional filtering range in the Despun system
title	:	string as “THEMIS / SCM-444 / Fast Survey #1”
subtitle	:	string as “step 3: Calibrated data in spinning system [nT] without DC”
anomaly	:	Integer, 0 if no anomaly, or else a number from 1 to 21

Anomaly code:

1 : Saturation of waveform X
2 : Saturation of waveform Y
3 : Saturation of waveform Z
4 : Saturation of waveforms X et Y
5 : Saturation of waveforms Y et Z
6 : Saturation of waveforms Z et X
7 : Saturation of waveforms X,Y,Z

8 : Null in waveform X
9 : Null in waveform Y
10 : Null in waveform Z
11 : Nulls in waveforms X et Y
12 : Nulls in waveforms Y et Z
13 : Nulls in waveforms Z et X
14 : Nulls in waveforms X,Y,Z

15 : Change of date between 2 vectors
16 : Time is going back up
17 : Time discontinuity between 2 vectors

18 : END INDEXED_DATA encountered on RFF file
19 : END DATA encountered on RFF file
20 : END ROPROC_FORMAT_FILE encountered on RFF file
21 : EOF encountered on L1 file

d) Notes on calibration steps 5 to 7

Calibration steps 1 to 4 can be done from the procedure defined in chapter I as :

```
PRO thm_getdata_SCMWF_L1    DataFile, datim1,datim2, SCnum, mode, times, x,y,z,
                            Fsamp, Fspin
```

Nevertheless, to include step 5, which is the transformation from Spinning Sun-sensor L system (SSL) to the Despun Sun L system (DSL), one needs the sun pulse or spin phase information. So, this step is available if we can add to this procedure a new output which could be an array S_phase(n), where each element of the array is the spin phase computed at the same time as the times used for the array.

A second option would be to add 2 new output arrays, S_time(m) and S_Phase(m), where we can get the spin phase and the time where the spin phase is computed. In this case, the sun pulse time resolution could be less than in the previous option. Spin phase can then be interpolated and aligned with the time vectors by the calibration program.

In the same spirit, step 7 (Despun to GSE) will be available if we add to this procedure a new output which could be the spin axis direction (right ascension and declination) in the Geocentric Equatorial Inertial system (GEI), at least once per day.

If we remain with the initial thm_getdata_SCMWF_L1 procedure defined in chapter I, that means that level 5 to 7 will not be available in the calibration program, and so SCM (or FGM, or others) should provide a generic program to compute SSL to DSL transformation, and DSL to GSE and GSM transformations.

These programs will then be applied, for SCM, to the cowave.resu file containing step 4 calibrated data (in SSL system), and will act as a filter (cowave.resu file format will be the input and output of such a filter). This can be envisaged, since such filters already exist for CLUSTER, and can be used without any change for SCM on THEMIS. Filters process a cowave.resu file, and replace its content by a new one. For example, it exists wave_to_GSE, wave_to_GSM, wave_to_MAG, wave_to_MFA, wave_to_MVA, wave_filter, etc. (MFA= Magnetic Field Aligned, using FGM data; MVA= Minimum Variance Analysis). Conversion from F90 to IDL could be envisaged.

All this options should be discussed soon.

III. USEFUL PROCEDURES FOR ACCURATE TIME CONVERSION

A) Note on time accuracy with ASCII expression

THEMIS time is given in Julian second from epoch Jan 1.0, 2001 (Real double precision). With IDL on Unix machine, the real double precision variable have at least 15 digits accuracy, and 16 at maximum. Then, THEMIS time should be written as:

182285472.7170868 for 2006-10-11 18:51:12.7170868

and so, we must have 7 digits after decimal point.

This is confirmed by the use of IDL with successive operations, such as differences between two consecutive points, for instance:

180473085.499282837		
180473085.499404907	8192.000000000 Hz	*computed from time difference with the previous point*
180473085.499526978	8192.000000000 Hz	*idem*
180473085.499649048	8192.000000000 Hz	*idem*
180473085.499771118	8192.000000000 Hz	*idem*
180473085.499893188	8192.000000000 Hz	*idem*

So, the conversion of a real double precision time variable must be expressed into an ASCII field with 7 digits after the point, then with a time accuracy of 100 nano second. Test are given in chapter III.C.1 hereafter.

B) Note on date-time field with ISO convention

ISO date-time epoch is a string as '2006-02-27T07:39:27.1234567Z', used in RFF files, where T is the date-time separator and Z is the end of the string. In ISO convention, seconds can be given with any precision, and hour, minute, second can be omitted if not useful. In case of THEMIS, to keep 100 nano second time accuracy, no less, no more, the time field must be written with 7 digits after the decimal point of the field "second".

C) Best format for accurate ISO date-time field

1. Test on ISO format to keep time accuracy

As previously sawn, the theoretical real double precision variable have at least 15 digits accuracy, and 16 at maximum. This has been checked on time issued from real THEMIS data, as shown in the following listing:

```
----------------------------------------------
test time accuracy in THEMIS SCM waveform data
----------------------------------------------

Reminder: theoretical real double precision is 15 significant digits, 16 max.

Read CDF data file splitvc3_44c.cdf
Done... Number of vectors data=      524288

1) Time of 10 first points of the waveform
   ---------------------------------------

 180473085.499282837
 180473085.499404907      8192.000000000 Hz
 180473085.499526978      8192.000000000 Hz
 180473085.499649048      8192.000000000 Hz
 180473085.499771118      8192.000000000 Hz
 180473085.499893188      8192.000000000 Hz
 180473085.500015259      8192.000000000 Hz
 180473085.500137329      8192.000000000 Hz
 180473085.500259399      8192.000000000 Hz
 180473085.500381470      8192.000000000 Hz

2) Themis time and Julian 1970 time:
   ---------------------------------

Themis origin in Jul_sec_1970: 978307200.000000000
Themis time of the 1st point : 180473085.499282837
Julian time (sec since 1970) :1158780285.499282837

3) Classic computation of ISO date-time with secdate and sectime
   -------------------------------------------------------------

ISO date-time= 2006-09-20T19:24:45.499Z --> Standard procedures: only mil.sec.

4) Best accuracy with secofday_to_strtime (7 digits after point)
   --------------------------------------

ISO date-time= 2006-09-20T19:24:45.4992828Z  --> Accuracy kept

5) Inverse : computation of Themis time from ISO date-time
   -------------------------------------------------------

hour, min, secdbl= 19 24        45.499282800

sec of the day  =     69885.499282800
Jul_sec 1970    =1158780285.499282837
Themis time     = 180473085.499282837

Initial thm time= 180473085.499282837
Difference      =         0.000000000    -->  No difference

6) Unnecessary over precision (ex: 10 digits):
   -------------------------------------------

ISO date-time= 2006-09-20T19:24:45.4992828369Z

sec of the day  =     69885.49928283690
Jul_sec 1970    =1158780285.49928283691
Themis time     = 180473085.49928283691

Initial thm time= 180473085.49928283691
Difference      =         0.00000000000    -->  No difference
```

```
Go and back with Julsec_to_isodatim and isodatim_to_Julsec :
-----------------------------------------------------------


7) Time of 10 first points after go and back, Ndigit= 6
   ----------------------------------------------------

 180473085.499282837
 180473085.499404907 2006-09-20T19:24:45.499405Z      8208.031311155  Hz
 180473085.499526978 2006-09-20T19:24:45.499527Z      8192.000000000  Hz
 180473085.499649048 2006-09-20T19:24:45.499649Z      8192.000000000  Hz
 180473085.499771118 2006-09-20T19:24:45.499771Z      8192.000000000  Hz
 180473085.499892950 2006-09-20T19:24:45.499893Z      8208.031311155  Hz
 180473085.500015020 2006-09-20T19:24:45.500015Z      8192.000000000  Hz
 180473085.500137091 2006-09-20T19:24:45.500137Z      8192.000000000  Hz
 180473085.500258923 2006-09-20T19:24:45.500259Z      8208.031311155  Hz
 180473085.500380993 2006-09-20T19:24:45.500381Z      8192.000000000  Hz

8) Time of 10 first points after go and back, Ndigit= 7
   ----------------------------------------------------

 180473085.499282837
 180473085.499404907 2006-09-20T19:24:45.4994049Z     8192.000000000  Hz
 180473085.499526978 2006-09-20T19:24:45.4995270Z     8192.000000000  Hz
 180473085.499649048 2006-09-20T19:24:45.4996490Z     8192.000000000  Hz
 180473085.499771118 2006-09-20T19:24:45.4997711Z     8192.000000000  Hz
 180473085.499893188 2006-09-20T19:24:45.4998932Z     8192.000000000  Hz
 180473085.500015259 2006-09-20T19:24:45.5000153Z     8192.000000000  Hz
 180473085.500137329 2006-09-20T19:24:45.5001373Z     8192.000000000  Hz
 180473085.500259399 2006-09-20T19:24:45.5002594Z     8192.000000000  Hz
 180473085.500381470 2006-09-20T19:24:45.5003815Z     8192.000000000  Hz

9) Time of 10 first points after go and back, Ndigit=10
   ----------------------------------------------------

 180473085.499282837
 180473085.499404907 2006-09-20T19:24:45.4994049072Z  8192.000000000  Hz
 180473085.499526978 2006-09-20T19:24:45.4995269775Z  8192.000000000  Hz
 180473085.499649048 2006-09-20T19:24:45.4996490479Z  8192.000000000  Hz
 180473085.499771118 2006-09-20T19:24:45.4997711182Z  8192.000000000  Hz
 180473085.499893188 2006-09-20T19:24:45.4998931885Z  8192.000000000  Hz
 180473085.500015259 2006-09-20T19:24:45.5000152588Z  8192.000000000  Hz
 180473085.500137329 2006-09-20T19:24:45.5001373291Z  8192.000000000  Hz
 180473085.500259399 2006-09-20T19:24:45.5002593994Z  8192.000000000  Hz
 180473085.500381470 2006-09-20T19:24:45.5003814697Z  8192.000000000  Hz

-------------------------------------------------------------
 test_time_accuracy : Normal termination
-------------------------------------------------------------
```

2. Best ISO format for THEMIS time

Test shown above, as theoretical values deduced in chapter III.B, leads to compute the date time string in ISO format with 7 digits after the decimal point (see § 8). Less than 7 digits leads to an imprecise time (see § 7), and more than 7 is usefulness (see § 9).

D) Useful accurate date and time procedures

1. To convert THEMIS time to an ISO date time string:

a) Compute decimal Julian second since 1970-01-01

Since THEMIS time is given in Julian second from epoch Jan 1.0, 2001, origin of this date must be converted in Julian 1970 second by (see III.D.3) :

```
ori_t2001=datesec('2001-01-01')
```

ori_t2001 is a double precision real value, and is equal to 978307200.

So Julian 1970 decimal second is given by :

```
Jul_sec= thm_time + ori_t2001
```

for instance 1160592672.71708679 with thm_time= 182285472.71708679, corresponding to date time epoch 2006-10-11T18:51:12.71708679Z

b) Compute ISO date-time epoch

The string date field is given by (see III.D.3) :

```
str_date= secdate(Jul_sec ,sec_day, FMT=0)
```

While the string time field is given by:

```
str_time= secofday_to_strtime(sec_day,8)
```

The second arguments of `secofday_to_strtime` correspond to the number of digits taken to write the `str_time` value. As time accuracy is 10 nsec, it leads to have 8 digits after decimal point.

Finally, ISO date time string is given by:

```
isodatim= str_date +'T' +str_time +'Z'
```

c) Julian_sec_to_isodatim procedure

The previous commands can be linked in a single procedure such as :

```
isodatim= Julsec_to_isodatim(Jul_sec,Ndigits)
```

which give the isodatim string from the 1970 Julian second, with `Ndigits` digits after decimal point.

d) thm_time_to_isodatim procedure

One can also provides the computation of the `isodatim` string directly from the THEMIS time with:

```
isodatim = thm_time_to_isodatim(thm_time)
```

In this case, `Ndigits` is set to 7.

2. To convert an ISO date time string in THEMIS time:

a) Decode ISO date time string

This can be done by :

```
ih= FIX(STRMID(isodatim,11,2))
im= FIX(STRMID(isodatim,14,2))
nZ= STRPOS(isodatim,'Z')
ncar= nZ-17
sd= DOUBLE(STRMID(isodatim,17,ncar))
```

b) Compute Julian second since 1970

The second of the day is first computed by :

```
secofday= ih*3600.D +im*60.D +sd
```

Then the Julian second since 1970 is computed by :

```
Jul_sec = datesec(STRMID(isodatim,0,10)) + secofday
```

c) Compute THEMIS time

THEMIS time is simply computed by :

with ori_t2001= 978307200.

d) isodatim_to_Julsec procedure

The previous commands can be linked in a single procedure, as :

```
Jul_sec=isodatim_to_Julsec(isodatim,secofday)
```

which gives the Julian second from 1970 from the `isodatim` string; the variable `secofday` is an output argument which give the second of the day, and can be useful.

e) isodatim_to_thm_time procedure

THEMIS time can be directly given by:

```
thm_time = isodatim_to_thm_time(isodatim)
```

3. Summary of useful procedures:

IDL FUNCTIONS & PROCEDURES	***INPUT***	***VALUE***

Existing Berkeley functions

Jul_sec = **datesec**(str_date)	str_date as '2006-02-27'	give DBL Julian second since 1970-01-01
str_date = **secdate**(Jul_sec ,sec_day, FMT=0)	Jul_sec since 1970-01-01	give string date as '2006-02-27' give also sec of the day in DBL
str_datim = **time_to_str**(Jul_sec, FMT=0)	Jul_sec since 1970-01-01	give string date & time as '2006-02-27/07:39:27.123'
str_time = **sectime**(sec_day)	DBL sec of the day	give string time as '07:39:27.123'

New CETP functions

str_time = **secofday_to_strtime**(sec_day,Ndg)	DBL sec of the day Ndg: digits after decimal point.	give string time, second with Ndg digit after decimal point
Jul_sec = **isodatim_to_Julsec**(isodatim, sec_day)	isodatim as '2006-02-27T07:39:27.1234567Z'	give DBL Julian second since 1970-01-01 + DBL sec of the day
isodatim = **Julsec_to_isodatim**(Jul_sec,Ndg)	Jul_sec: Julian sec. since 1970-01-01 Ndg: digits after decimal point.	give string isodatim as '2006-02-27T07:39:27.1234567Z'
isodatim = **thm_time_to_isodatim**(thm_time)	Themis Julian sec. since 2001-01-01	give string isodatim as '2006-02-27T07:39:27.1234567Z'
thm_time = **isodatim_to_thm_time**(isodatim)	isodatim as '2006-02-27T07:39:27.1234567Z'	give Themis Julian sec. from 2001-01-01
isodatim = str_to_isodatim(str_datim)	string date & time as '2006-02-27/07:39:27.123'	give string isodatim as '2006-02-27T07:39:27.123Z'
str_datim = isodatim_to_str(isodatim)	string isodatim as '2006-02-27T07:39:27.1234567Z'	give string date & time as '2006-02-27/07:39:27.1234567'

New CETP procedures

decode_isodatim(isodatim, year, month, day, hour, min,sec)	isodatim as '2006-02-27T07:39:27.1234567Z'	give year, month, day, hour, min as an integer, and sec as a real double precision.
encode_isodatim(year, month, day, hour, min, sec, isodatim)	year,month,day,hour,min as an integer, sec is DBL Julian second since 1970-01-01	give isodatim as '2006-02-27T07:39:27.1234567Z'

BIBLIOGRAPHY

[1] **“Waveform calibration method for no-linear transfer function”**, *Patrick Robert, to be written*

[2] **“Intensité et polarisation des ondes UBF détectées à bord de GEOS-1”**,”Méthode d’analyse numérique du signal et production en routine de sommaires expérimentateurs. Problèmes rencontrés et solutions pratiques.” *Patrick Robert, CRPE/ETE, Note technique CRPE/NT/71, Mai 1979.*

[3] **“CLUSTER SOFTWARE TOOLS, Part I, Coordinate transformation library”** *, by Patrick ROBERT, RPE/TID, DT/CRPE/1231, Centre de Recherche en Physique de l’Environnement Terrestre et Planétaire, CNET-CNRS, July 1993.*

[4] **“ROCOTLIB: a Coordinate Transformation Library for Solar-Terrestrial studies”** *, Version1.7, by Patrick ROBERT, Centre de Recherche en Physique de l’Environnement Terrestre et Planétaire, Internal report n° RI-CETP/02/2003.*

Complete documentation is available on http://cdpp.cesr.fr /View and recorder /libraries/the Rocotlib software, as the Fortran code in 1.7 version.
Last version 2.0, November 2006, with dipole direction computed until year 2010, code in Fortran 77 as well as IDL, is available on request from CETP (patrick.robert@cetp.ipsl.fr).

[5] **“THEMIS Science Coordinate System Definition”**, *THM-SOC-110, September 29, 2006, NAS5-02099.*

[6] **“Roproc Command Language, a set of linked commands for spatial data processing”** *, Patrick Robert, CNRS/CETP, version 2.0, November 2004.*

[7] “**THEMIS SCM waveform calibration method”** *, P. Robert, December 2006* (to be written).