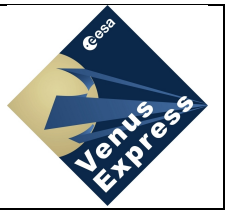


SPICAM/SPICAV project
Spectroscopy for Investigation of Characteristics of the Atmosphere of Mars
Spectroscopy for Investigation of Characteristics of the Atmosphere of Venus

Using SPICAM and SPICAV IR 1A level data

LATMOS-SPI-UD-002

Release 2

05 November 2014

Prepared by:	Approved by:
Nicolas Chapron (LATMOS)	Anna Fedorova (IKI) Gaetan Lacombe (LATMOS)

Release history

Release	Date	Comments
1	31/08/2014	First release
1.1	22/10/2014	Correction of text, functional parameters were added
1.2	24/10/2014	Block diagram updates FORTRAN and C added in section 4
2	05/11/2014	Reformatted

Table of contents

1. Introduction

1.1 Purpose

This document provides a description of the cleaned SPICAM and SPICAV IR data, which is named the level 1A data. The purpose is to give the users, essential information about the format and the content of the data.

1.2 Reference documents

Ref N°	Title	Author	Date
1	SPICAM IR acousto-optic spectrometer experiment on Mars Express, Journal of Geophysical Research, Volume 111, Issue E9, CiteID E09S03	Korablev et al.	2006
2	SPICAV IR acousto-optical spectrometer experiment on Venus-Express, Planetary and Space Science, Volume 65, Issue 1, p. 38-57	Korablev et al.	2012
3	Exploration of Mars in SPICAM-IR experiment onboard the Mars-Express spacecraft: 1. Acousto-optic spectrometer SPICAM-IR, Cosmic Research, Volume 44, Issue 4, pp.278-293	Korablev et al.	2006
4	SPICAM and SPICAV data level 1A IR. Process description	Nicolas Chapron Anna Fedorova	2014

1.3 Abbreviations

ADU	Analog to Digital Unit
AOTF	Acousto-Optical Tunable Filter
DC	Dark Current
FITS	Flexible Image Transport System
FOV	Field Of View
HDU	Header Data Unit
IR	Infrared
LOS	Line Of Sight
MEX	Mars Express spacecraft
NaN	Not a Number
SPICAM	Spectroscopy for Investigation of Characteristics of the Atmosphere of Mars
SPICAV	Spectroscopy for Investigation of Characteristics of the Atmosphere of Venus
UV	Ultraviolet
VEX	Venus Express spacecraft

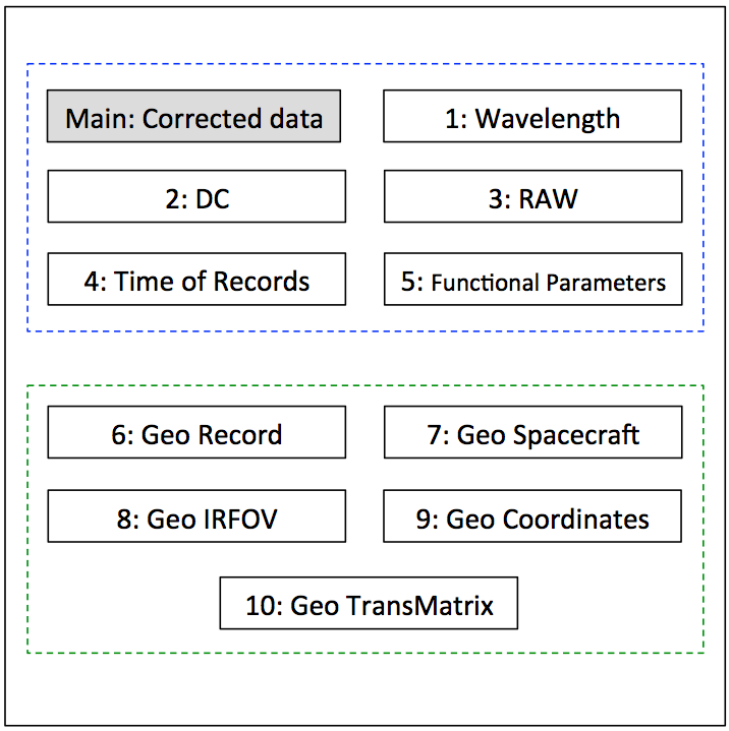
2. Brief description of the data

The SPICAM and SPICAV 1A IR data files are built from the 0B level files. The 1A level provides calibrated data (corrected from the dark charge) with associated wavelengths, geometry data and parameters of the instrument.
The 1A level files are under FITS (Flexible Image Transport System) format and conform to the requirements of the FITS standard. For a general introduction of FITS format please follow the link: http://fits.gsfc.nasa.gov/fits_documentation.html

A FITS file is made of several data blocks, which are named as Header Data Units (HDUs). The file contains a primary data block and several extensions, which can be an Image Extension (multi-dimensional array of pixels, similar to a primary data block) or a Binary Table Extension (tabular information in a binary representation which contain multi-dimensional arrays).

The 1A level files contain the following data blocks:

- ***Primary***: 3D image. It corresponds to the corrected data.
- ***Wavelength***: 3D image extension. It corresponds to wavelengths of the data.
- ***DC***: 3D image extension. It corresponds to the DC signal used to calibrate the data.
- ***Raw***: 3D image extension, it corresponds to the raw data.
- ***Time of records***: Binary table extension
- ***Functional_Parameters***: Binary table extension. This extension contains functional parameters information and temperatures.
- ***Geo_****: Binary table extensions. Each of these extensions contains the geometrical parameters record associated.



SPICAM/V IR 1A FITS files block diagram

The development team provides a reading program, based on the IDL, FORTRAN and C languages. It can be found in the directories:

/net/nfs/spicam/data/ARCHIVE/SOFTWARE/1A/
/net/nfs/spicav/data/ARCHIVE/SOFTWARE/1A/

NB: those distributions are identical; the reading program is able to read both SPICAM and SPICAV 1A IR files as they share the same format.

The section 4 explains how to use the program. Note that any other common languages can read the FITS files.

3. Data structure description

1.4 Primary block: CLEANDATA

This field represents the primary block of the FITS data and it contains the "clear" data. The name given in the FITS file is *CLEANDATA*. This is a 3-dimension float array containing the raw data corrected from the Dark Current and saturation in ADU [1, 4]. The dimensions of the array are: ***[NB_SPECT, NB_POINT, NB_CHANN]***

The header of the primary block is the main header of the data; it contains general information about the observation data. All the parameters are described in the following table:

PARAMETERS	*TYPE*	*DESCRIPTION*
INSTRU:	STRING	Name of the instrument
ORBIT:	INTEGER	Orbit number
SEQ_NB:	INTEGER	Sequence number
STATUS:	STRING	Status of the FITS file (see below)
NB_SPECT:	INTEGER	Number of spectra
NB_POINT:	INTEGER	Number of spectral points
NB_CHANN:	INTEGER	Number of channels
NB_BLOCK:	INTEGER	Number of blocks for one spectrum
SIZE_BL:	INTEGER	Size of a block (see below)
OBSTYPE:	STRING	Type of observation
BEGINS:	STRING	UTC time of the first record of the observation, format="YYYY-MM-DDThh:mm:ss.ms"
ENDS:	STRING	UTC time of the last record of the observation format="YYYY-MM-DDThh:mm:ss.ms"

Note about the STATUS keyword:
The STATUS keyword defines the status of the geometry inside the FITS file. Two values are possible, value "P" means that the geometry is in preliminary status (the geometry is predicted), and value "F" means that the geometry is in final status (the geometry is reconstructed).

Note about the SIZE_BL keyword:
The SIZE_BL correspond to the size of a block (on board packed data transmitted from the spacecraft) in spectral points, 332 spectral points for standard case if two detectors were activated and 664 points if one detector was activated [see 1, 2].

1.5 Extension 1: WAVELENGTH

The first extension contains the wavelengths in nm corresponding to each spectral point in *CLEANDATA*. The extension is a 3-dimension float array with the same dimension as the *CLEANDATA* block. The wavelengths are obtained based on RF frequency-wavelength calibration described in [1] for SPICAM IR and [2] for SPICAV IR.

1.6 Extension 2: DARK CURRENT

The second extension contains the Dark Current data in ADU. The calibration curves and temperature dependence of DC are presented in [1,3] for SPICAM IR and [2] for SPICAV.
The name given in the FITS file is *DC*. The extension is a 3-dimension float array with the same dimension as the *CLEANDATA* block.

1.7 Extension 3: RAW DATA.

The third extension contains the raw data in ADU corrected for the saturation [4].
The name given in the FITS file is *RAW*. The extension is a 3-dimension float array with the same dimension as the *CLEANDATA* block.

1.8 Extension 4: TIME OF RECORDS

The fourth extension is a Binary Table and defines the time of records (beginning of a spectrum record). The name given in the FITS file is *TIME_OF_RECORDS*. It contains the time parameters of each data records in a structured way. The size of the table is equal to *NB_SPECT*.

PARAMETER	*TYPE*
YEAR	INTEGER
MONTH	INTEGER
DAY	INTEGER
HOUR	INTEGER
MINUTE	INTEGER
SECOND	INTEGER
MSECOND	DOUBLE

1.9 Extension 5: FUNCTIONAL PARAMETERS

The fifth extension is a Binary Table and contains vectors corresponding to the functional parameters of the instrument during the observation. The name given in the FITS file is *FUNCTIONAL_PARAMETERS*. The following table describe each parameters of the table:

PARAMETER	*TYPE*	*DESCRIPTION*	*UNIT*
FREQUENCY	FLOAT ARRAY	Vector of AOTF frequencies - size:[NB_POINTS]	kHz
T_D0	FLOAT ARRAY	Vector of Detector 0 temperature for the beginning of each spectrum - size:[NB_SPECT]	Volts
T_D1	FLOAT ARRAY	Vector of Detector 1 temperature for the beginning of each spectrum - size:[NB_SPECT]	Volts
T_AOTF	FLOAT ARRAY	Vector of AOTF temperature for the beginning of each spectrum - size:[NB_SPECT] (not valid for SPICAV)	Kelvin
T_BASE	FLOAT ARRAY	Vector of base plate temperature for the beginning of each spectrum - size:[NB_SPECT] (not valid for SPICAV)	Kelvin
POWER_RF	FLOAT ARRAY	Vector of RF power at 110MHz (middle of AOTF frequency range), DAC Value is determined by command - size:[NB_SPECT]	Volts
PVS	FLOAT ARRAY	Vector of the supply voltage control - size:[NB_SPECT]	Volts

The header of this extension contains scalar numbers corresponding to the functional parameters of the instruments programmed for the observation:

PARAMETER	*TYPE*	*DESCRIPTION*	*UNIT*
CODEOP	BOOLEAN	UV operation code	
UVON	BOOLEAN	UV channel indicator – always 1	
SOIR_ON	INTEGER	SOIR channel indicator - always 0 for SPICAM	
EXIT	BOOLEAN	Flight (1) or lab (0) mode	
DETS	INTEGER	Detectors used for spectrum measurement: 0-detector 1 only; 1- detector 2 only; 2 – both detectors; 3 – AOTF RF power at both DC channels	
SOURCE	BOOLEAN	Host command is (0) or not (1) executed	
TIME	FLOAT	AOTF chopping period	MS
GAIN	INTEGER	Amplifiers gain factor	
GAINBST	FLOAT	The gain of amplification stage (only for SPICAV)	
DAC	INTEGER	AOTF RF power control	
PELTIER	BOOLEAN	Peltier code indicator	
DOTS	INTEGER	Number of predefined set of spectrum dots. The set recorded after windows	
NDOTS	INTEGER	Total number of spectral points within DOTS	

1.10 Extension 6-10: GEOMETRY

The geometry data is defined in 6 different Binary Table extensions. All the information are directly extracted from the 0B level geometry file and rearranged in a structured way.

1.10.1 Extension 6: GEO_RECORDS

This extension is an array containing the time of a beginning of block record with fixed time interval of delivery. The maximum size of the array is equal to the number of spectrum multiply by the number of blocks needed to build one spectrum (*NB_SPECT*NB_BLOCKS)*. To find the geo information for single spectral point, users have to interpolate from the time of spectra records taking into account the integration time of one spectral point and time between blocks' delivery. Example is presented in [4]
The two table parameters are:

PARAMETER	*TYPE*	*DESCRIPTION*
NUMBER	INTEGER	Record number
TIME	STRING	UTC time of the record, format="YYYY-MM-DDThh:mm:ss.ms"

The header of this extension contains geometrical parameters of the instrument which do not vary during observation:

PARAMETER	*TYPE*	*DESCRIPTION*	*UNIT*
TARGET	STRING	Pointed target (For stars, see SPICAM/V star catalog)	
SUNLAT	FLOAT	Sub-solar point latitude in planet body-fixed frame, at first record	Deg.
SUNLONG	FLOAT	Sub-solar point longitude in planet body-fixed frame, at first record	Deg.
SUNDIST	FLOAT	Distance between planet and sun, at first record	U.A.
SUNLS	FLOAT	Planet position seen from the Sun in Ecliptic J2000 frame	Deg.
SUNRA	FLOAT	Sun Right Ascension in J2000 frame, at first record	Deg.
SUNDEC	FLOAT	Sun Declination in J2000 frame, at first record	Deg.
IRFOVPOS	FLOAT ARRAY	IR FOV position vector (theta, phi) in the spacecraft referential	Deg.
SLIT_C	FLOAT ARRAY	Slit center position vector (theta, phi) in the spacecraft referential	Deg.
IRSHIFT	FLOAT	IR/UV delta	Pixels

Note about the IRFOV_POS and SLIT_CENTER parameters:
For the specific SUN observing mode, the values of the *SLIT_CENTER* parameter are extracted from the (theta,phi) line in the 0B archive geometry files, and the values of the *IRFOV_POS* parameter is equal to: [theta:30.235°, phi:-0.097°]
For the other observing modes, the values of both parameters *IRFOV_POS* and *SLIT_CENTER* are equal to the (theta,phi) line from the 0B archive geometry files.

1.10.2 Extension 7: GEO_SPACECRAFT

This extension is a table containing some geometrical information relative to the Mars Express or Venus Express spacecraft:

PARAMETER	*TYPE*	*DESCRIPTION*	*UNIT*
LAT	FLOAT	Latitude of the sub-spacecraft point (in degrees, body-fixed frame)	Deg.
LONG	FLOAT	Longitude of the sub-spacecraft point (in degrees, increases toward East from 0 to 360, body-fixed frame)	Deg.
ALT	FLOAT	Altitude of the spacecraft above planet ellipsoid (in km, body-fixed frame)	Deg.
SZA	FLOAT	Solar Zenith Angle at sub-spacecraft point (in degrees, body-fixed frame)	Deg.

1.10.3 Extension 8: GEO_IRFOV

This extension is a table containing all the geometrical information relative to the IR channel field of view:

PARAMETER	*DESCRIPTION*	*UNIT*
LAT	Latitude of nearest point on MARS/VENUS ellipsoid to LOS emanating from the IR FOV center (IAU_MARS/IAU_VENUS)	Deg.
LONG	Longitude of nearest point on MARS/VENUS ellipsoid to LOS emanating from the IR FOV center (IAU_MARS/IAU_VENUS)	Deg.
ALT	Altitude above the nearest point on MARS/VENUS ellipsoid of LOS emanating from the IR FOV center (IAU_MARS/IAU_VENUS, < 0 IF intersection)	Km.
SZA	Solar zenith angle at nearest point on MARS/VENUS ellipsoid to LOS emanating from the IR FOV center (IAU_MARS/IAU_VENUS)	Deg.
DISTTOPLANETNP	Distance from MEX/VEX to nearest point on MARS/VENUS ellipsoid to LOS emanating from the IR FOV center (IAU_MARS/IAU_VENUS, < 0 if behind SC)	Km.
PIXELSIZE	Pixel (0.01 deg) size at nearest point on MARS/VENUS ellipsoid to LOS emanating from the IR FOV center (IAU_MARS/IAU_VENUS)	Km.
ANGLELOSSUN	Angle between S/C-SUN vector and LOS emanating from the IR FOV center (IAU_MARS/IAU_VENUS)	Deg.
PHASELOSSUN	Phase angle between the SUN and the Observer measured at the nearest point on MARS/VENUS ellipsoid to LOS emanating from the IR FOV center	Deg.
SOLARINCIDENCE	Solar incidence angle between SUN and Normal at nearest point on MARS/VENUS ellipsoid to LOS emanating from the IR FOV center	Deg.
SOLARLOCALTIME	Solar local time at nearest point on MARS/VENUS ellipsoid to LOS emanating from the IR FOV center	Deg.
ANGLENORMALOBSERVER	Emission angle between Normal and Observer at nearest point on MARS/VENUS ellipsoid to LOS emanating from the IR FOV center	Deg.
SUNAZIMUTH	Azimut of SUN in MEX/VEX axes (relative to X)	Deg.

DISTLOSPLANETCENTER	Distance between the nearest point on LOS emanating from the IR FOV center and center of MARS/VENUS	Km.
DISTSCPLANETCENTER	Distance between MEX/VEX position and center of MARS/VENUS	Km.

All parameters are floating numbers.

1.10.4 Extension 9: GEO_COORDINATES

This extension is a table containing all the Mars Express or Venus Express spacecraft coordinates in different referential:

PARAMETER	*DESCRIPTION*
XSC_X	X component of the unit inertial pointing vector (1,0,0) in SC coordinates relative to MARSIAU (SPICAM) or IAU_VENUS (SPICAV) frame
XSC_Y	Y component of the unit inertial pointing vector (1,0,0) in SC coordinates relative to MARSIAU (SPICAM) or IAU_VENUS (SPICAV) frame
XSC_Z	Z component of the unit inertial pointing vector (1,0,0) in SC coordinates relative to MARSIAU (SPICAM) or IAU_VENUS (SPICAV) frame
YSC_X	X component of the unit inertial pointing vector (0,1,0) in SC coordinates relative to MARSIAU (SPICAM) or IAU_VENUS (SPICAV) frame
YSC_Y	Y component of the unit inertial pointing vector (0,1,0) in SC coordinates relative to MARSIAU (SPICAM) or IAU_VENUS (SPICAV) frame
YSC_Z	Z component of the unit inertial pointing vector (0,1,0) in SC coordinates relative to MARSIAU (SPICAM) or IAU_VENUS (SPICAV) frame
ZSC_X	X component of the unit inertial pointing vector (0,0,1) in SC coordinates relative to MARSIAU (SPICAM) or IAU_VENUS (SPICAV) frame
ZSC_Y	Y component of the unit inertial pointing vector (0,0,1) in SC coordinates relative to MARSIAU (SPICAM) or IAU_VENUS (SPICAV) frame
ZSC_Z	Z component of the unit inertial pointing vector (0,0,1) in SC coordinates relative to MARSIAU (SPICAM) or IAU_VENUS (SPICAV) frame
ANGLE_XSC_LOCALVERT	Angle between the S/C X axis and the local vertical at the planet Nearest Point (in degrees)
ANGLE_XSC_PROJ	Angle between the projection plane U axis and the S/C X axis projected on this plane (in degrees)
P1_u	P1 u horizon point component in Planproj
P1_v	P1 v horizon point component in Planproj
P2_u	P2 u horizon point component in Planproj
P2_v	P2 v horizon point component in Planproj
P3_u	P3 u horizon point component in Planproj
P3_v	P3 v horizon point component in Planproj
P4_u	P4 u horizon point component in Planproj
P4_v	P4 v horizon point component in Planproj
P5_u	P5 u horizon point component in Planproj
P5_v	P5 v horizon point component in Planproj

All parameters are floating numbers.

1.10.5 Extension 10: GEO_TRANSMATRIX

This extension is a table containing the transformation matrix coordinates from the local referential to EMEJ2000:

PARAMETER	*DESCRIPTION*
X_Ra	X Ra component of transformation matrix from MARSIAU (SPICAM) or IAU_VENUS (SPICAV) to EMEJ2000
X_Dec	X Dec component of transformation matrix from MARSIAU (SPICAM) or IAU_VENUS (SPICAV) to EMEJ2000
Y_Ra	Y Ra component of transformation matrix from MARSIAU (SPICAM) or IAU_VENUS (SPICAV) to EMEJ2000
Y_Dec	Y Dec component of transformation matrix from MARSIAU (SPICAM) or IAU_VENUS (SPICAV) to EMEJ2000
Z_Ra	Z Ra component of transformation matrix from MARSIAU (SPICAM) or IAU_VENUS (SPICAV) to EMEJ2000
Z_Dec	Z Dec component of transformation matrix from MARSIAU (SPICAM) or IAU_VENUS (SPICAV) to EMEJ2000

All parameters are floating numbers.

4. Proposed reading software

1.11 IDL

1.11.1 Installation and execution

The IDL software package is located in the directories:
/net/nfs/spicam/data/ARCHIVE/SOFTWARE/1A/IDL/
/net/nfs/spicav/data/ARCHIVE/SOFTWARE/1A/IDL/

Download and package in your working directory. Compile the procedures. The function SPICA_Read_1A takes for argument the filename of the 1A file to read and returns a structure variable. The SILENT keyword can be added to avoid printing the reading confirmation.

1.11.2 Returned Structure

The IDL structure returned by the software contains several variables. Each variables match to one or several specific FITS blocks, except for the *Infos* sub-structure, which contains the parameters from the main data header. The *Geo* structure contains all the content from extension 6 to 11.

VARIABLE	*TYPE*
CLEANDATA	FLOAT ARRAY [NB_SPECT, NB_POINT, NB_CHANN]
WAVELENGTH	FLOAT ARRAY [NB_SPECT, NB_POINT, NB_CHANN]
DC	FLOAT ARRAY [NB_SPECT, NB_POINT, NB_CHANN]
RAW	FLOAT ARRAY [NB_SPECT, NB_POINT, NB_CHANN]
INFO	STRUCT
TIME_OF_RECORDS	STRUCT
FUNCTIONAL_PARAMETERS	STRUCT
GEO	STRUCT

1.11.3 Example of use

- Read a SPICAM 1A file located in /1Apath into *obs* variable:

```
IDL> obs=SPICA_Read_1AIR ('/1Apath/SPIM_1AR_00687A01_E_01.FITS')
The file SPIM_1AR_00687A01_E_01.FITS was read succesfully!
```

- The FITS data header is stored in the *info* sub-strcuture :

```
IDL> help,obs.info
  NAXIS1          LONG            131
  NAXIS2          LONG            664
  NAXIS3          LONG              2
  INSTRUMENT      STRING    'SPICAM  '
  ORBIT           LONG            687
  SEQUENCE        LONG              1
  STATUS          STRING    '     0'
  OBSTYPE         STRING    'E'
  BEGINTIME       STRING    '2004-08-03T02:44:45.680'
  ENDTIME         STRING    '2004-08-03T02:53:25.680'
  NB_SPECTRUM     LONG            131
  NB_POINTS       LONG            664
  NB_CHANNELS     LONG              2
  NB_BLOCKS       LONG              2
  SIZE_BLOCK      LONG            332
```

1.12 FORTRAN

1.12.1 Building the CFITSIO library

The 1A reading software provided uses the ***cfitsio*** library for reading FITS format data files. This library package can be found at:
http://heasarc.gsfc.nasa.gov/docs/software/fitsio/fitsio.html

To build the library, the user should:

- Download and uncompress the cfitsio package for your system.
- Build the *cfitsio* library in local, by launching the *makefile* (see README file in the cfitsio directory for building information).
- The result of building the library is the creation of the file: ***libcfitsio.a*** in the *cfitsio* directory.

For Unix-like systems, the simple launch of the *makefile* permits to build easily the library. For Windows systems, a unix-like emulator system such as *Cygwin* can be used to launch this *makefile*.

1.12.2 Installation and execution

The FORTRAN software package is located in the directories:
/net/nfs/spicam/data/ARCHIVE/SOFTWARE/1A/FORTRAN/
/net/nfs/spicav/data/ARCHIVE/SOFTWARE/1A/FORTRAN/

The package contains several FORTRAN source files, needed to read SPICAM or SPICAV FITS data file:

- *spica_read_1a_mod.f*: This module contains the function spica_read_1a to read the SPICAM and SPICAV UV/IR data in level 1A. The function returns a FORTRAN derived data type (struc_1AFile) corresponding to the FITS file
- *spica_struct_1a_mod_type.f*: This module contains the declarations of the different derived data types specific to the SPICAM/SPICAV UV/IR data level 1A structure
- *spica_struct_1a_mod_func.f*: This module contains the functions used to read all the FITS extension data and header of the SPICAM/SPICAV UV/IR data level 1A
- *test_uv.f*: program to test the spica_read_1a_mod module for UV data
- *test_ir.f*: program to test the spica_read_1a_mod module for IR data

To build a executable test program, the user should follows these steps:
(Example for gfortran compiler)
- From the root directory the following commands, compile each source file and the main test program:
gfortran -ffree-form -c -o "spica_read_1a_mod.o" "spica_read_1a_mod.f"
gfortran -ffree-form -c -o "spica_struct_1a_mod_type.o" "spica_struct_1a_mod_type.f"
gfortran -ffree-form -c -o "spica_struct_1a_mod_func.o" "spica_struct_1a_mod_func.f"
gfortran -ffree-form -c -o "test_uv.o" " test_uv.f"
gfortran -ffree-form -c -o "test_ir.o" " test_ir.f"

- link each compiled file with the cfitsio library **libcfitsio.a** (example for test_uv):
gfortran -o "test_uv" "spica_read_1a_mod.o" "spica_struct_1a_mod_type.o" "spica_struct_1a_mod_func.o" "spica_struct_1a_mod_func.o" "test_uv.o –lcfitsio

1.12.3 Returned Structure

The function spica_read_1a from the module *spica_read_1a_mod.f* return a specific FORTRAN derived type:

VARIABLE	*TYPE*
CLEANDATA	REAL ARRAY [NB_SPECT, NB_POINT, NB_CHANN]
WAVELENGTH	REAL ARRAY [NB_SPECT, NB_POINT, NB_CHANN]
DC	REAL ARRAY [NB_SPECT, NB_POINT, NB_CHANN]
RAW	REAL ARRAY [NB_SPECT, NB_POINT, NB_CHANN]
INFO	TYPE STRUCT_INFO
TIME_OF_RECORDS	TYPE STRUCT_TIME_OF_RECORDS
FUNCTIONAL_PARAMETERS	TYPE STRUCT_PARAMETERS
GEOINFO	TYPE STRUCT_GEOINFO
GEO	TYPE STRUCT_GEO

1.13 C

1.13.1 Installation and execution

Before using the C reading software, the user should build its own *cfitsio* library by following the 4.2.1 section.

The C software package is located in the directories:
/net/nfs/spicam/data/ARCHIVE/SOFTWARE/1A/C/
/net/nfs/spicav/data/ARCHIVE/SOFTWARE/1A/C/

The package contains several C source and header files, needed to read SPICAM or SPICAV FITS data file:

- *include/spica_read_1a.h*: This header contains the declarations of the main functions used to read the data
- *include/spica_struct_1a.h:* This header contains the declarations of the different typedef specific to the SPICAM/SPICAV UV/IR data level 1A structure
- *include/spica_read_1a_cmd.h:* This header contains the declarations of the functions used to read the data in command line
- *src/spica_read_1a.c:* This source file contains the function which reads the SPICAM and SPICAV UV/IR data in level 1A. The function returns the C typedef (spica_1Afile) corresponding to the FITS file
- *src/spica_struct_1a.c:* This source file contains the functions used to read all the FITS extension data and header of the SPICAM/SPICAV UV/IR data level 1A
- *src/spica_read_1a_cmd.c:* This source file contains the functions to reads the data in command line files, and permits to display some data sample, parameters and information

As an example of using the spica_read_1a library, the user can build the spica_read_1a_cmd program to read the SPICAM/SPICAV 1A data in command line (provided as an example for testing the reading process). To build the project on UNIX system, the user should type first the command *./configure* in the console and then the command *make*. Those 2 commands create the executable file in the "*src*" directory.

1.13.2 Returned Structure

The function spica_read_1a from the source file *spica_read_1a.c* return a specific C typedef:

VARIABLE	*TYPE*
CLEANDATA	FLOAT * ([NB_SPECT, NB_POINT, NB_CHANN])
WAVELENGTH	FLOAT * ([NB_SPECT, NB_POINT, NB_CHANN])
DC	FLOAT * ([NB_SPECT, NB_POINT, NB_CHANN])
RAW	FLOAT * ([NB_SPECT, NB_POINT, NB_CHANN])
INFO	TYPEDEF INFO
TIME_OF_RECORDS	TYPEDEF TIME_OF_RECORDS
FUNCTIONAL_PARAMETERS	TYPEDEF PARAMETERS
GEOINFO	TYPEDEF GEOINFO
GEO	TYPEDEF GEO