1 Introduction

The executable `spi_gain_corr` is part of the SPI scientific analysis preparation software component (SAP). It converts raw energy values of SPI event data (in PHA units) into calibrated energy values in units of keV. On input, `spi_gain_corr` needs a freshly produced Observation Group (generally using the tools `og_create`. `spi_gain_corr` will create for each science window and all SPI scientific modes data structures that hold the calibrated energy values (so called `COR` data).

`spi_gain_corr` is an ANSI C++ executable that has been developped under ISDC support platform 6.3 and requires the library `spi_toolslib` (version 4.0.0 or higher).

2 Getting started

Before installing `spi_gain_corr`, make sure that the ISDC support platform 6.3 or higher is installed on your system, and that the library `spi_toolslib` version 4.0.0 or higher are installed.

After downloading the `spi_gain_corr.tar.gz` file, step into a directory that should hold the distribution, move the `spi_gain_corr.tar.gz` file into this directory and type after the UNIX prompt `$` (don't type this prompt):

```
$ gunzip spi_gain_corr.tar.gz
$ tar xvf spi_gain_corr.tar
```

The first command uncompresses the distribution file, the second unpacks the files.

Before configuration, the distribution needs to be reset to a clean state. To do this, type

```
$ make distclean
```

Then, configure the distribution. It is assumed here that you have previosuly installed the ISDC support platform, thus you should type

```
$ ~/bin/ac_stuff/configure
```

Finally, build the distribution by typing

```
$ make global_install
```

To perform a unit test, type

```
$ make test
```

Make sure that the test data `spi_test_data-1.0.tar.gz` are available at your site (they should reside in a directory whose name is defined by the `ISDC_TEST_DATA_DIR` environment variable).

3 Parameter file

```
#############################################################################
#                                                                           #
#               Centre d'Etude Spatiale des Rayonnements                    #
#                     (in collaboration with ISDC)                          #
#                                                                           #
#                     SPI Event Energy Correction                           #
#                                                                           #
# ------------------------------------------------------------------------- #
#                                                                           #
#  File:      spi_gain_corr.par                                             #
#  Version:   1.7.0                                                         #
#  Component: cor                                                           #
#                                                                           #
#  Author:    Juergen Knoedlseder                                           #
#             knodlseder@cesr.fr                                            #
#             CESR                                                          #
#                                                                           #
#  Purpose:   Parameter file of the SPI event energy correction             #
#             executable.                                                   #
#                                                                           #
#  History:   1.7.0  11-Jan-2005  First release (Rev. 7)                    #
#                                                                           #
#############################################################################
#
# Input DOLs
#===========
prpOG,   s,ql,            "og_spi.fits",,,"Input Group DOL or filename (SWG/OG/IDX)"
coeffDOL,s,ql,"spi_gain_coeff_idx.fits",,,"Calibration DOL or filename (file/IDX)"
#
# Output DOL
#===========
outfile,s,ql,"outfile.fits",,,"Output DOL or filename (if input group is SWG)"
#
# Task parameters
#================
randomise,b,h,yes,,,"Randomise energies ?"
#
# Standard parameters
#====================
clobber,b,h,no,,,"Overwrite existing output data ?"
```

Instead of specifying complete DOLs (Data Object Locations), which are composed of a filename plus the data structure extension (HDU), `spi_gain_corr` accepts also simple filenames and adds the appropriate data structure extensions. This means that **specified data structure extensions are ignored**.

The following parameters have to be specified:

- `inDOL` specifies an input group (either a science window group, an index file, or an observation group) or level `PRP` for which gain correction should be performed.
- `coeffDOL` specifies the DOL of a gain calibration file (of HDU type `[SPI.-COEF-CAL]`) or a gain calibration index file (of HDU type `[SPI.-COEF-CAL-IDX]`) that should be used for gain correction.
- `outfile` specifies the name of the file that will hold the science window group of level `COR` and the calibrated energy data structures in case that a single science window group has been specified as input group and that no `ALL` data structure is present. In this case, no observation group has to exist prior to execution and `spi_gain_corr` will create its own data structure.
- `randomise` specifies if event energies should be randomised. If set to `yes`, `spi_gain_corr` will transform each raw energy channel into a random number that is uniformly distributed within ± 0.5 around the channel value. This means that event energies are returned as if the photons uniformly fill the raw channel spectra bins.
- `clobber` specifies if corrected data that possibly exist already should be overwritten by the task. If `clobber = no`, `spi_gain_corr` will exit if a `COR` data structure exists already. If `clobber = yes`, any existing `COR` data will be overwritten. This applies also to the case of the treatment of single science window groups.

Based on the location of each science window group in the input group (denoted by `.`), `spi_gain_corr` stores the corrected data in the file location

4 Interface definition

`spi_gain_corr` loops over the input group (which may be a single science window group but can also be an index or an observation group) and performs gain correction for all events that are found in any of the 4 science modes in the data.

In the case that the input group is an observation group (or the index of an observation group), `spi_gain_corr` stores the corrected data in the file location

`./spi/cor/spi_cor_xxxx.fits[1]`

where *xxxx* is one of `oper`, `emer`, `calb`, or `diag`, and `./` is the location of the observation group or index group file. If the corrected energy files do not exist, `spi_gain_corr` creates them and adds one HDU for each event type (such as single events, PSD events, PSD curves, multiple events, etc.). `spi_gain_corr` fills the `ENERGY` column of these HDUs. If the files exist already, and if the `clobber` parameter is set to `yes`, `spi_gain_corr` deletes all existing rows and recreates the corrected data structure (note that for PSD events, this implies that a possible post-processed PSD flag is erased).

In the case that a single science window group is processed, `spi_gain_corr` stores the science window group of level `COR` and the corrected energy data structures in the file specified by the parameter `outfile`. Note that it is assumed in this case that only a single science mode exists in the input data.

`spi_gain_corr` makes use of the `spi_toolslib` API `SPIGain` for performing gain corrections (see `spi_toolslib` User Manual for more details). In particular, possible time gaps in the calibration file (that occured due to improper calibration handling) are filled by extending the end time of a validity interval to the start time of the next validity interval. For events that occured before the first validity interval in the calibration file, the first calibration coefficients are used. For events that occured after the last validity interval in the calibration file, the last calibration coefficients are used.

5 Algorithm

Three formulae have been implemented in `spi_gain_corr` for gain correction for each of the two gain ranges of SPI (high gain range, covering $\sim 0-2$ MeV and low gain range, covering $\sim 0-8$ MeV). The formula that is employed is selected by the keywords `CAL_FCT0` for the high gain range (or low energy range) and `CAL_FCT1` for the low gain range (or high energy range) in the gain calibration file, hence the formula selection is transparent to the user of `spi_gain_corr`.

5.1 POLY

If `CAL_FCT = POLY`, the energy of each photon is evaluated using the formula

$$E = c_0 + c_1 \times \mathrm{PHA} + c_2 \times \mathrm{PHA}^2 + c_3 \times \mathrm{PHA}^3 + c_4 \times \mathrm{PHA}^4 \qquad (1)$$

where c_i are the five gain correction coefficients (found in vector elements 1-5 of the column `CHAN_KEV` in the `SPI.-COEF-CAL`), E is the corrected energy in keV,

and PHA is the was energy value in AFEE channels.

5.2 POLY_1/C

If `CAL_FCT = POLY_1/C`, the energy of each photon is evaluated using the formula

$$E = c_0/\mathrm{PHA} + c_1 + c_2 \times \mathrm{PHA} + c_3 \times \mathrm{PHA}^2 + c_4 \times \mathrm{PHA}^3 \qquad (2)$$

where c_i are the five gain correction coefficients (found in vector elements 1-5 of the column `CHAN_KEV` in the `SPI.-COEF-CAL`), E is the corrected energy in keV, and PHA is the was energy value in AFEE channels.

5.3 POLY_1/C2

If `CAL_FCT = POLY_1/C2`, the energy of each photon is evaluated using the formula

$$E = c_0/\mathrm{PHA}^2 + c_1/\mathrm{PHA} + c_2 + c_3 \times \mathrm{PHA} + c_4 \times \mathrm{PHA}^2 \qquad (3)$$

where c_i are the five gain correction coefficients (found in vector elements 1-5 of the column `CHAN_KEV` in the `SPI.-COEF-CAL`), E is the corrected energy in keV, and PHA is the was energy value in AFEE channels.

6 Error codes

The executable `spi_gain_corr` may stop with the following error codes:

```
SPI_GAIN_CORR_ERROR_MEM_ALLOC              -200000
SPI_GAIN_CORR_ERROR_EOF                    -200001
SPI_GAIN_CORR_ERROR_FILE_ERROR             -200002
SPI_GAIN_CORR_ERROR_RAW_PRP_MISMATCH       -200003
SPI_GAIN_CORR_ERROR_COR_EXISTS             -200004
SPI_GAIN_CORR_ERROR_BAD_PRP_GROUP          -200005
SPI_GAIN_CORR_ERROR_BAD_DETE_ID            -200006
SPI_GAIN_CORR_ERROR_NO_OUTPUT_FILE         -200007
SPI_GAIN_CORR_ERROR_RAW_COR_MISMATCH       -200008
```

They have the following meaning:

- `SPI_GAIN_CORR_ERROR_MEM_ALLOC` : the allocation of dynamical memory has failed. Probable your system resources are too limited to run this task.

- `SPI_GAIN_CORR_ERROR_EOF` : not used anymore.
- `SPI_GAIN_CORR_ERROR_FILE_ERROR` : not used anymore.
- `SPI_GAIN_CORR_ERROR_RAW_PRP_MISMATCH` : input `RAW` and `PRP` tables have an inconsisten number of rows. This should never happen. Probably, your input data are corrupted.
- `SPI_GAIN_CORR_ERROR_COR_EXISTS` : result `COR` structure exists already, and no overwriting (using the `clobber` parameter) has been allowed.
- `SPI_GAIN_CORR_ERROR_BAD_PRP_GROUP` : the specified input group is neither a science window group, nor an index group or an observation group.
- `SPI_GAIN_CORR_ERROR_BAD_DETE_ID` : an invalid germanium detector ID has been found in the input data. Probably your data are corrupted.
- `SPI_GAIN_CORR_ERROR_NO_OUTPUT_FILE` : no output filename has been specified in the parameter `outfile` while the task required one. Please specify a filename.
- `SPI_GAIN_CORR_ERROR_RAW_COR_MISMATCH` : a mismatch has been found between the `RAW` and `COR` table size. Probably your data are corrupted.

In addition, all errors that may occur in calls to ISDC support functions (such as for example DAL, RIL or PIL) are forwarded. Please consult the ISDC web pages for getting information about these error codes.