1 Changelog

This document is outdated. I have moved to subversion.

1.1 Version 0.9.1 ⇒ 0.9.2-pre11 [Nov 24, 2004]

Docs added file How_to_Link.txt

Docs documentation updates and several typos fixed

libSCIA_hdf4 moved all HDF4 routines from `libSCIA` to separate directory, which makes the compiling and linking much cleaner. Modified Configuration and Makefiles, accordingly.

RPMS moved IDL routines to `/usr/share/nadc_idl`

IDL_NADC added kayword "errors" to function `get_lv1_mds_data.pro` which contains upon return the precission error on the spectrum

IDL_NADC `get_lv1_mds_time.pro`: small bugfixes added more error messages

IDL_NADC `scia_lv0_mds_trend.pro`: fixed bug: replaced REPLICATE with safer REPLICATE_INPLACE, and do memory correction also when only one MDS with coaddf ¿ 1 is given

IDL_NADC removed keyword "write" from function `scia_lv0_rd_mds_info.pro`, also modified the called C-routines, see below

IDL_NADC new function `scia_lv0_wr_mds_info.pro`

IDL_NADC added keyword `/ir_pet` to function `GET_LV0_MDS_HK` to obtain the correct PET of the Epitaxx detectors (TN-SCIA-0000DO/19,10.03.1999)

IDL_NADC function `scia_lv1_rd_mds.pro`: the values of the level 1b line-of-sight zenith angles are always larger than zero, and the azimuth angle jumps with 180 degrees while scanning through nadir. Setting the environment variable "SCIA_CORR_LOS" to one will modify these values as follows: removing the jump in the azimuth angles and returns negative zenith angles, when the original azimuth angle was larger than 180 degree. This behaviour is controlled bij the (UNIX) environment variable `SCIA_CORR_LOS`: when set and equal to one the LOS angles are modified.

IDL_NADC function `scia_lv1_rd_mds.pro`: the parameter "MDS" could contain old data when called with invalid values for the keywords: chan_id and clus_id

SCIA program `scia_nl1.c`: the values of the level 1b line-of-sight zenith angles are always larger than zero, and the azimuth angle jumps with 180 degrees while scanning through nadir. Setting the environment variable "SCIA_CORR_LOS" to one will modify these values as follows: removing the jump in the azimuth angles and returns negative zenith angles, when the original azimuth angle was larger than 180 degree. This behaviour is controlled bij the (UNIX) environment variable `SCIA_CORR_LOS`: when set and equal to one the LOS angles are modified.

libIDL program `scia_lv1_idl.c`: the values of the level 1b line-of-sight zenith angles are always larger than zero, and the azimuth angle jumps with 180 degrees while scanning through nadir. Setting the environment variable "SCIA_CORR_LOS" to one will modify these values as follows: removing the jump in the azimuth angles and returns negative zenith angles, when the original azimuth angle was larger than 180 degree. This behaviour is controlled bij the (UNIX) environment variable `SCIA_CORR_LOS`: when set and equal to one the LOS angles are modified.

libSCIA modified elements of structure `sip_scia`

libSCIA small bugfixes in `Inline/Y_interpol.inc`

libSCIA added function to calculate precision errors of the SCIA calibration `Inline/calc_error.inc`

libSCIA many bugfixes and improvements to `Inline/dark_corr.inc`

libSCIA Eval_Poly returns a double float

libSCIA added cacluation of precision error after PPG correction `ppg_etalon_corr.inc`

libSCIA added cacluation of precision error after Straylight correction `straylight_corr.inc`

libSCIA added calculation of the precission error on the spectrum (`scia_lv1_calib.c`)

libSCIA implemented dark correction for Limb states using the last limb-measurement (= limb-dark), see `Inline/dark_corr.inc`

libSCIA bugfix: index offset when ignoring PMD_4 in SIP

libSCIA implemented the function "Calc_SpectralRelError" in module `Inline/calc_error.inc`

libSCIA bugfix: range check - errors must be positive `Inline/calc_error.inc`, function "Calc_ShotNoise"

libSCIA implemented correction of the PET for the Epitaxx detectors, see TN-SCIA-0000DO/19,10.03.1999, (`scia_lv1_calib.c`). Setting the environment variable "SCIA_CORR_PET" to one will modify the PET of the Epitaxx readouts.

libSCIA added new module `scia_lv1_corr_los.c` with function "scia_lv1_corr_los" for Line-Of-Sight correction

2 Changelog

2.1 Version 0.9.0 ⇒ 0.9.1 [March 24, 2004]

general improved the generation of the RPM SPEC file

libSCIA found several bugs in the calculation of the members "pixel_type", "tang_ground_point" of the struct "geoL_scia" the modules `get_lv1c_mds_geo` and `scia_lv1_rd_mds`

IDL_NADC added module `GET_LV1_MDS_STATE`

IDL_NADC found several brown paperbag bugs in the calculation of the Limb geolocation

SCIA no longer distribute the program `scia_patch_1b`, this program is for internal (SRON) usage only

2.2 Version 0.8.6h ⇒ 0.9.0 [March 16, 2004]

general many improvements/cleanups made to the Makefiles and configuration files

Documentation many small improvements and updates: fixed typos, introduced new ones

general add new command-line parameter for applications (`-show_param`) to show the values of the parameters settings without reading the input-file

IDL-interface moved to IDL version 6.0, using several of the new (nice) features/extensions. Therefore, many routines no longer work under IDL 5.x!

IDL-interface introduced compile options" "idl2, hidden", and explicit typecasting

IDL-interface added several new routines:

- `SCIA_HL_OPEN`
- `SCIA_WR_H5_MEMCORR`
- `SCIA_RD_H5_MEMCORR`
- `SCIA_WR_H5_NLCORR`
- `SCIA_RD_H5_NLCORR`

IDL-interface modified the structs: `info_clus` and `mds1_scia`

IDL-interface bug-fixed several modules:

- `get_lv0_mds_collo` (minor)
- `get_lv0_mds_data` (major)
- `get_lv0_mds_hk` (major)
- `get_lv1_mds_data` (minor)
- `gome_lv1_rd_fcd` (major)
- `scia_lv0_mds_trend` (major + new options)
- `scia_lv0_mds_info` (major)
- `scia_lva_rd_mds` (major + new options)

C-interface improved string handling by using `strlcpy` and `strlcat`

C-interface faster floating point conversion between BE/LE (`swap_bytes.h`)

hdf5-interface wherever appropriate I will move from libHDF5+ to the NCSA HDF5 High Level API (SCIA, only)

mds0_info changed the layout and contend of this struct, to save disk space and memory, affected several C-routines and IDL-routines

SCIA new application `scia_patch_nl1`

libIDL new function and modules:

- scia_gen_idl.c added _NADC_SCIA_Calib_Mask
- scia_h5_idl.c
- scia_lv0_idl.c added _SCIA_LV0_MD_INFO

libIDL bug-fixed scia_lv1_idl.c (minor)

libNADC new functions and modules:

- allocate.c added ALLOC_D2D
- nadc_wr_ascii.c added nadc_write_long
- nadc_decodepatch.c
- nadc_encodepatch.c
- nadc_files_not_equal.c

libNADC bug-fixed several modules:

- nadc_akima.c (minor)
- nadc_decodecalib.c (major changes)
- nadc_encodecalib.c (major changes)
- nadc_error.c (minor)
- nadc_init_param.c (minor + extensions)
- nadc_show_param.c (minor + extensions)

libSCIA new functions and modules:

- get_scia_lv1c_mds.c
- get_scia_lv1c_pmd.c
- get_scia_lv1c_polV.c
- scia_lv0_h5_info.c
- scia_lv0_wr_mds_info.c
- scia_lv1_cp_ads.c
- scia_lv1_cp_hdr.c
- scia_lv1_h5_mds.c
- scia_lv1_patch_mds.c

libSCIA bug-fixed several modules:

- get_scia_lv1c_geo.c (major + extensions)
- scia_h5_structs.c (cleanups)
- scia_lv0_rd_mds.c (cleanups + more checks on data)
- scia_lv0_rw_mds_info.c (cleanups)
- scia_lv0_select.c (major)
- scia_lv1_calib.c (rewrite + many improvements)
- scia_lv1_rd_mds.c (major + cleanups)

libSCIA bug-fixed and added new extension to the SCIA calibration routines

libSCIA (bug-fixed) no longer write empty Auxiliary and PMD MDS after "end of measurement" in module `SCIA_LV0_WR_ASCII_MDS`

2.3 Version 0.8.6 ⇒ 0.8.6h [September 9, 2003]

2.3.1 Enhancements

libHDF5+ replaced the functions `H5Eget_auto` and `H5Eset_auto` by the HDF5 macro's `H5E_BEGIN_TRY` and `H5E_END_TRY`

SCIA initial release of an application `scia_update_nl1` to patch a level 1b dataset by modifying (key-)datasets. The output dataset should still be conform the PDS format. Implemented features: daily updated dark current values (starting from Janaury 2003), and improved bad/dead pixel mask. Still to be implemented features: memory correction, non-linearity correction, alternative Solar spectrum, MDS selection on state-ID, measurement category, etc.

IDL_NADC introduced the IDL statement `COMPILE_OPT` with the (RSI) recommended argument *idl2*, combined with the argument *hidden* for all such routines that are not intended to be called directly by regular users. The first argument changes the default behaviour of IDL to assume that lexical integer constants default to 32-bit type rather than 16-bit integers, therefore, the programmer is forced to do more explicit type casting.

general many improvements to the makefiles and their configuration files. The latter have now been moved to the config-directory.

libSCIA the level 0 *info*-files are now written as small HDF5-files, or binary format when no HDF5 support is included.

Docs added more documentation

general renamed the software to nadc_tools, because our software is more than a bunch of programs

2.3.2 Bug Fixes

IDL_NADC added function `gome_lv1_free_fcd` to module `gome_lv1_rd_fcd.pro`

IDL_NADC found brown paper-bag bug, introduced in 0.8.6.c

2.4 Version 0.8.5 ⇒ 0.8.6 [July 21, 2003]

2.4.1 Enhancements

libNADC added options to do Dark Signal correction to PMD data, and calculate an error estimate of the applied calibration of science data

Docs documentated the options for Dark Signal correction to PMD data, and calculate an error estimate of the applied calibration of science data

Makefiles introduced the ".PHONY" target

IDL_NADC defined structures in separate modules. These definition modules are all stored in a subdirectory `StructDefs`. This directory has to be added to the search path of IDL, for example (*Note the "+"*):

```
export IDL_PATH="<IDL_DEFAULT>:+/nfs/local/rsi/site-idl/nadc_idl"
```

IDL_NADC replaced confusing description of 'the keyword "posit" in documentation

IDL_NADC improved documentation of several modules

libNADC implmented the much cleaner string functions `strlcpy` and `strlcat` in module `nadc_string.c`

general replaced where possible `strncpy` and `strncat` by `strlcpy` and `strlcat`

Docs completely rewrote the Makefile

Makefiles added more GNU specific features

libSCIA small speedup improvements in Inline-modules `polarisation_corr.inc` and `radiance_corr.inc`

general verified that the code compiles and runs fine with RedHat 9 (gcc 3.2.2) and hdf5 version 1.6.0

libSCIA implemented the improved "Leakage" correction derived from five dark current calibration states (Scia level 0 products) take every orbit since end December 2002, derived by Q. Kleipool (SRON) [`dark_corr.inc`]

2.4.2 Bug Fixes

general I WILL NOT USE OPERATORS IN MACROS
I WILL NOT USE OPERATORS IN MACROS
I WILL NOT USE OPERATORS ...

IDL_NADC removed all redunded definitions from `defs_scia_gen.h`, and removed the obsolete files: `defs_gome_gen.h` and `defs_scia_lv1.h`

IDL_NADC fixed several bugs in function `GET_LV1_MDS_DATA`

IDL_NADC debugged function `GET_SCIA_LEVEL` which could crash when no file is opened by `SCIA_OPEN`, or the input file is not a PDS file

IDL_NADC function `SCIA_LV0_MDS_TREND` added checks on input data, and striker checks on "posit"

IDL_NADC module `scia_lv1_rd_mds.pro` documentation improvements, check number of states after calling `GET_LV1_STATES`, and debugged state selection in function `GET_LV1_STATES`

compiler code clean-ups, as suggested by gcc 3.x

libSCIA function `SCIA_RD_PDS_INFO` removed function parameter `called_by`, added checks: are we really reading ASCII data, boundary checking before copying the data to the arrays "keyword" and "keyvalue"

include module swap_bytes.h fixed a few typos in the function byte_swap_u64 (Thanks to Jochen Skupin)

libNADC fixed nasty bug in nadc_akima.c: used uninitialised elements of array "yy"

libSCIA module scia_lv1_calib.c many small bug fixes and code improvements

IDL_NADC made keyword "align" work properly in function GET_LV1_MDS_DATA

2.5 Version 0.8.4 ⇒ 0.8.5 [April 14, 2003]

libGOME made a separate module gome_lv1_wr_h5_pmd.c to grep PMD data and calculate the geolocation of each PMD pixel

IDL_NADC added routine gome_lv1_pcd_pmd.pro to obtain PMD measurements and geolocation (lat/lon, only)

libSCIA optimisation for pol/rad correction in scia_lv1_calib.c (nearly 10% faster)

libSCIA applied some speed optimisations

libGOME added glr-structure to pmd_gome structure, and calculate geometry for each PMD pixel

libGOME write glr-structure as a compound data type: added gome_h5_structs.c and incorporated this function in gome_lv1_wr_h5_pcd.c and gome_lv1_wr_h5_pmd.c

IDL_NADC the routines gome_lv1_pcd_pmd.pro and gome_lv1_rd_pcd.pro also reads the glr-structure

Makefiles small improvements

2.6 Version 0.8.3 ⇒ 0.8.4 [April 01, 2003]

Docs went through installation.tex and nadc_progs.tex adding more text, and fixing typos

libNADC minor improvements to command-line parameter handling nadc_init_params.c and nadc_show_params.c

libSCIA updated to the latest dead pixel mask (Q.K.)

libNADC minor bug fixes: wrong MDS selection logic in nadc_init_params.c

libSCIA scia_lv1_rd_psp.c and scia_lv1_rd_rsp.c: separate modules to read Limb/Occultation data sets

libSCIA scia_lv1_calib.c tried to write too many PMD and PolV MDS records...

SCIA implemented the separate modules to read Limb and Occultation polarisation and radiance sensitivity data sets in scia_nl1.c

libSCIA implemented radiance sensitivity correction of Limb and Occultation MDS data

libSCIA **ToDo** polarisation sensitivity correction of Limb and Occultation MDS data

libSCIA **ToDo** polarisation/radiance sensitivity correction of monitoring MDS data

libGOME code clean-ups, and code modification necessary for the IDL interface

libIDL **NEW** added an C-IDL layer for GOME level 1b data products

IDL_GOME **NEW** IDL routines to read GOME level 1b data products

libGOME **bug fix**: I was unaware that the detector signal of GOME could be co-added.

libSCIA reading level 0 Detector MDS should be faster now, through the more aggressive use of inline functions (also affects the IDL read module)

libSCIA added type_mds to the Nadir level 1c MDS structure (both C and IDL modules), and moved the Sun glint/Rainbow flags to the Nadir geolocation structure.

IDL_SCIA more checks on posit: $0 \leq \text{posit}[0] \leq \text{posit}[1] < \text{ArraySize}$

IDL_SCIA type error in *scia_ol2_rd_cld.pro*

IDL_SCIA Scia off-line MDS modules: added check on all array sizes for memory allocation. Fixed several minor bug fixes.

IDL_SCIA fixed minor bugs (parameter checking/boundary checking)

SCIA write software version to HDF5 file

SCIA write software version to HDF4 file

GOME write software version to HDF5 file

include modified C-header files so they can be used by C++ code

include made conversion of reals (IEEE(BE) to Intel) about 20% faster

libSCIA added HDF4 routines for level 2 Offline products

libSCIA modified function parameter-list of `SCIA_LV0_RW_MDS_INFO`

2.7 Version 0.8.2 ⇒ 0.8.3 [Februari 27, 2003]

IDL_SCIA `scia_lv0_rd_det.pro`: better check on posit, i.e. negative input is neglected

libSCIA found a little bug in the polarisation correction. Now it reproduces the SciaL1C results within 1%, i.e. I still find inconsistencies even between successive pixels, but the differences are small.

2.8 Version 0.8.1 ⇒ 0.8.2 [Februari 25, 2003]

libNADC debugged and improved Error messages routines, added 2 routines to save and restore the error status

libSCIA implemented dark correction using Q. K. database (calibration option: `1+`)

2.9 Version 0.8.0 ⇒ 0.8.1 [Februari 20, 2003]

MDS0_INFO added parameter bcps: broadcast counter for the science measurements. Affects: `scia_lv0_rd_mds.c`, `scia_lv0_rd_mds_info.pro` and `get_lv0_mds_data.pro`

IDL_SCIA `scia_lv0_rd_pmd.pro` reverted the indices for the PMD data

libHDF4+ improved the HDF4 implementation by allowing the use of named dimensions. Note that the output HDF4 file is still restricted to 5000 scientific data sets.

libSCIA all the `scia_*_wr_hdf.c` routines use named dimensions (on-the-fly I have fixed several bugs).

IDL_SCIA renamed the structures for the State DSD of level 1 and 2 to state1_scia and state2_scia, respectively. (less confusing)

libSCIA `scia_lv1c_rd_calopt.c`: rearranged elements of structure "cal_options"

libSCIA bugfix `scia_lv1c_rd_calopt.c`: did not read element "moni_mds"

libSCIA bugfix (`scia_lv1_rd_mds.pro`): did not update state-record correctly
$\Longrightarrow$ **could result in crashes or corrupted output**

IDL_SCIA bugfix (`scia_lv1_rd_mds.pro`): incorrect use of CAL_OPTIONS for level 1c product
$\Longrightarrow$ **could result in crashes or corrupted output**

IDL_SCIA `scia_lv1c_rd_calopt.pro`: rearranged elements of structure "cal_options"

libSCIA bugfix (`scia_lv1_wr_h5_ppgn.c`): typo wrote avg_wls_spec instead of sd_wls_spec

MDS0_INFO (bugfix) adding parameter bcps left the structure unalligned, added dummy parameter

libSCIA & IDL_SCIA added calibration option to calculate reflectance. Note the calibration step itself is not yet implemented

libNADC the akima routines can do the calculation in single or double precision (optional)

libNADC changed type casting parameter from a confusing character to a enumerated type definition

libNADC added double precision version of `Y_LinInterPol`

libSCIA reduced the number of parameters for the Inline calibration routines

libSCIA calculation of radiances in double precision

IDL_SCIA `get_lv1_mds_data.pro` bug-fix related to the keyword `cluster`, and added keyword `rebin`

libNADC `nadc_akima.c` fixed many little bugs, only FIT_GRID_AKIMA is exported

libSCIA implemented polarisation sensitivity correction [**not tested!**]

libSCIA added possibility to correct for non-linearity of the infrared detectors arrays (channel 6 - 8) [**experimental!**]

copyright modified copyright to include 2003

archive added version number to the base-directory in the tarfile created by the makefile (option "tarfile")

Docs added documentation about the level 2 Offline structures

libIDL added C-IDL routines to read the level 2 Offline routines

libSCIA bugfix: (level 2 Off-line) updated to the latest I/O data definition document (Issue 4/B). Added `intg_time` to `lfit_scia` as the fourth element.

libSCIA bugfix: (level 2 Off-line) several elements of the GEO DSD's are documented as (unsigned) long, however, in the released Off-line products these elements are floats?!?

SCIA `scia_ol2` now really reads and dumps the whole Sciamachy level 2 Off-line product: verification orbit 20020811_101336.

libSCIA (level 1b) added calibration option to convert radiances to reflectances (Doppler shift correction not applied!)

libNADC made parameter checking more robust (`nadc_init_param.c`)

libNADC bugfix: `nadc_encodecalib.c` DO_CALC_REFL requires DO_CORR_RAD

libNADC bugfix: `nadc_akima.c` counter out-of-range

libSCIA `radiance_norm.c` added Doppler correction to wavelength grid of SRS

libNADC changes to optional parameters (!), major rewrite of the code with improved parameter checking

libNADC added more command-line parameters, and a few bugfixes

GOME & SCIA **modified** command-line parameters of the nadc_progs: consistently switches start with "−" and parameters taking arguments start with "−−".

include modifications to param_record, see previous item.

libSCIA bug fix in `scia_lv1_rd_mds.c`: did not update state-records correctly for level 1C

libSCIA fixed `scia_lv1_wr_hdf_lcpn.c`: used uninitialised named dimension

libSCIA & IDL_SCIA use named variables for the GDF parameters: β, P_{bar} and w_0

2.10 Version 0.7.2 ⇒ 0.8.0 [November 29, 2002]

IDL_SCIA debugged `get_lv0_mds_data.pro`, major bugfixes

IDL_SCIA debugged `get_lv1_mds_data.pro`, major bugfixes

IDL_SCIA debugged `scia_lv0_rd_mds_info.pro`, did not always initialise the return value for status

libSCIA debugged `scia_lv1_wr_hdf_mds.c`, fixed memory leak

libSCIA debugged `scia_lv1_wr_h5_mds.c`, fixed memory leak

libSCIA debugged `scia_lv1_hdf_structs.c`, did not properly define the polV vgroup

libSCIA `scia_lv1_calib.c` moved the geolocation routines to `get_scia_lv1c_geo.c`

IDL_SCIA added the routine GET_LV1_MDS_TIME

libNADC & libSCIA & IDL_SCIA implemented new calibration option scheme for level 1b data, affects many routines

IDL_SCIA did not expect the number of fitting windows to be larger than 6, now 10 fitting windows are allowed

libSCIA `scia_rd_dsd.c` could hang in an end-less loop, do now proper file-status checking

libSCIA `scia_lv1_wr_hdf_mds.c` added routines to write the level 1c MDS data to a HDF-file

libHDF4+ & libSCIA the generation time of a HDF4 file was seriously affected by repeated opening and closing of the output file. Eventually the program would crash. By opening the output file only once, in CRE_HDF4_NADC_BASE, using Hopen, Vstart and SDstart, we get reasonable performance. However, the HDF4 format is limited to 5000 dimensions, which can easily be exceeded with the current implementation. **To Do**: use share dimensions and share sets of geolocations between polV and clusters within one state

libNADC added hdf4_file_id and hdf4_sd_id to structure param_record (see previous item)

libSCIA check for NaN in `radiance_corr.inc`

libNADC made Akima interpolation significantly faster for monotonic increasing or decreasing input arrays

libNADC & Docs added a Disclaimer

2.11 Version 0.7.1 ⇒ 0.7.2 [November 6, 2002]

libSCIA debugged `scia_wr_hdf_lads.c`, thanks to Jan Fokke Meirink (KNMI), RvH

LibSCIA rearranged DSD's in the output HDF 4/5 files, RvH

IDL_SCIA debugged `scia_lv1_rd_mds.pro`, some variables were not initialised

IDL_SCIA added module `get_pmd_coord.pro` to calculate geolocations of PMD data

libSCIA debugged `scia_lv1_select.c`, errors when clusters where selected

libNADC & libSCIA made code ready for the implementation of alternative calibration algorithms

libSCIA added the bad pixel mask defined at SRON (still not official)

IDL_SCIA debugged `scia_lv1_rd_mds.pro`, the size of the output arrays was not calculated correctly when clusters or channels are selected ("SCIA_LV1_RD_ONE_MDS")

libSCIA added a separate function "IS_SCIA_LV1C" to test for 1b or 1c

libSCIA & IDL_SCIA removed `is_level_1c` from struct state1_scia

SCIA scia_nl1 now uses `IS_SCIA_LV1C`

libNADC & libSCIA added (rough) selection on geolocation, based on Geolocation of States (LADS). Had to change parameter list of "SCIA_LV1_SELECT_MDS"

SCIA changed calls to "SCIA_LV1_SELECT_MDS" accordingly `scia_nl1`

libSCIA debugged "SCIA_LV1_RD_STATE", returned wrong offsets to various MDS when called more than once

GOME & SCIA the programs now also show their version, instead of only those of the libraries

SCIA scia-extactor for level 2 returns the correct data sets for the `--info` option

libNADC new options for the Sciamachy Off-line level 2 extactor

libSCIA "SCIA_LV0_RD_MDS": number of DSR is returned in case of a read error

IDL_SCIA "GET_LV0_MDS_COLLO": debugged and documentation update

IDL_SCIA "SCIA_LV0_RD_DET": modified struct chan_hdr: renamed `counter` to `bcps`

IDL_SCIA "GET_LV0_MDS_TIME": debugged, calculating time for MDS_DET (now the BCPS of the channel data is used)

IDL_SCIA "CMP_SCIA_CAL": new routine to compare the EnviView calibration with the (ATBD) NADC calibration

libNADC debugged `nadc_interpol.c`: "Y_delta" can be negative...

libIDL always return `-1` when an error has occurred

IDL_SCIA always include `defs_scia_gen.idl` and `compile_nadc.idl` at the start of an IDL-script to read Scia-data

IDL_SCIA `scia_lv1_rd_mds.pro`:

- **SCIA_LV1_RD_MDS** removed bug for option `/NoMDS`
- **SCIA_LV1_ONE_MDS** bugs for channel/cluster selection
- **SCIA_LV1C_ONE_MDS** number of MDS could exceed number of clusters
- **GET_LV1_STATES** forgot to check `state.flag_mds`
- **SCIA_LV1_FREE_MDS** bug in TAG_NAMES test
- **LV1C_RD_MDS_PMD** the external functions are modified to return `-1` in case of an error. Thus now I check the return value, because a level 1c file may contain none of these structures
- **LV1C_RD_MDS_POLV** see previous item

SCIA_LV1_RD_MDS did not check the variables returned by the reading routines to be a structure

SCIA_LV1C_RD_MDS see previous item

IDL_SCIA debugged module `get_lv1_mds_data.pro`

default.mk
- added **and modified** path to NADC IDL-programs: IDLDIR
- added path to SRON calibration parameters: DATADIR
- script did not recognised a Pentium IV

libSCIA improved `scia_lv1_calib.c`

- added the usage of alternative calibrations. Currently, these datasets are only for in-house (SRON) usage.
- debugged the calibration routines (except Polarisation and Radiance). Found several major and minor bugs using the IDL module `cmp_scia_cal.pro`
- speed up the MDS calibration (use of static variables)
- added documentation to included modules
- bugfixed extraction of fractional polarisation (geolocation)

IDL_SCIA `get_lv1_mds_data` major rewrite, requires the requested state ID as a parameter, and updated the documentation

libSCIA `scia_lv1_calib.c` bug fix: the fractional polarisation values and geolocations were not initialised correctly

IDL_SCIA `scia_lv1_rd_mds.pro` first attempt to use the IDL function `ARG_PRESENT` to check the parameter keywords

libSCIA `scia_lv1_calib.c` added radiance sensitivity correction for NADIR states

libSCIA more debugging of the geolocation interpolation routines

libSCIA `scia_lv2_rd_cld.c` implemented a work-around for file with a corrupted CLOUDS_AEROSOL MDS

libSCIA `scia_lv1_calib.c` handle nadir pixel which include the backscan correctly, GEO_NADIR_MDS still not finished

libSCIA `scia_lv1_rd_mds.c` implemented pixel_type: backscan = 0, else 1. Updated the write routines for HDF4/5 and ascii

2.12 Version 0.7.0 ⇒ 0.7.1 [Augustus 16, 2002]

libSCIA updated the routines for level 2 Off-line products, according to ENV-ID-DLR-SCI-2200-4

IDL_SCIA added a IDL script, which a user can run to compile all the NADC-IDL routines

IDL_SCIA added more options to `scia_nl1_rd_mds`

SCIA removed memory allocation bug in `scia_nl2`

SCIA updated `scia_ol2` for level 2 Off-line products, according to ENV-ID-DLR-SCI-2200-4

2.13 Version 0.6.6 ⇒ 0.7.0 [Juli 17, 2002]

libNADC added module `nadc_bits.c` to turn on and off selected bits

libNADC added module `lin_interpol.c`

libNADC added many new options to the SCIA routines

libSCIA moved all calibration functions to separate modules and put these in the directory `./Inline`. These function have to be included in the calling program, so the compiler can inline these functions

libSCIA added "WaveLength Calibration"

libSCIA debugged `scia_lv1_calib`

libSCIA debugged `scia_lv1_state`

libSCIA extract PMD and Polarisation data from level 1b product, and convert it to a level 1c PMD-MDS or polV-MDS

libHDF4/5+ added new command-line options to "History"-group

libIDL added routine to read DSD CAL_OPTIONS

libIDL added selection on channels and clusters to level 1b/1c products

SCIA (level 1) reading DSD CAL_OPTIONS

SCIA (level 1) selection on category ID

SCIA (level 1) selection on clusters

SCIA (level 1) added option `--info`

SCIA (level 2) selection on MDS (currently only: BIAS, DOAS, and CLD)

SCIA (level 2) added option `--info`

General debugged Makefile

Gereral reorganised and updated the documentation