

GSAS v3.3.2 Release Notes

Introduction

GSAS 3.3.2 is a minor release which adds additional sanity checking for GLAS_L0proc and GLAS_L1A control file specifications, fixes energies and temperatures in L1A, and provides some additional fixes in waveforms.

Release Information

The ClearCase label for this release is RELEASE_3.3.1.

The release date is February 19, 2003.

Internal version numbers have been updated to “V3.3.1 February 2003” for the following:

```
./data/anc07_001_01_0000.dat
./bin/GLAS_L0proc
./bin/GLAS_L1A
./lib/err.sl
```

This should be verified during operation by checking the version information in the appropriate ANC06 files.

All other internal version numbers remain at their previous versions.

SMDS Impact

The distribution tarfile is on glasdev.wff.nasa.gov at the following location:

```
/glasdev1/v3/dist/gsas_v3.3.2.tar.z.
```

New versions of the following anc07 data file is required:

```
./data/anc07_001_01_0000.dat
./data/anc07_001_01_0004.dat
```

GLAS_L0proc and GLAS_L1A die with a fatal error (status=3) if a sanity-check violation is detected. There are significantly more substantial checks made to ensure that the APIDs used with an ANC29 was created in GLAS_L0proc and the EXACT same APIDS used when ANC29 is read in GLAS_L1A.

All libraries and binaries should be recompiled using the top-level Makefile. The newly-created versions should be installed within the testbed (pending CCB release for operations).

Product Changes

No product sizes were changed.

The product parameters were changed.

Detailed Change Notes

CR0000177 : Work-around for EDOS-filled APID12/13/26 packets.

All 10 of the APID12/13/26 sub-packet shot counters are now checked for valid range (as opposed to just the first counter being checked). This should help us detect EDOS filled packets better.

PR0000176 : GLA01 energy computation is giving negative values. .

Modified the code to take pass the gain to C_CalcNrg directly from the APID without conversion into volts. Also fixed the problem with Engineering data on GLA01 where the code was not handling the 1/4sec engineering data rate correctly. Status still poses a problem, so the temperatures are initialized to -27.21 deg on startup. Code which uses these temperatures MUST check the appropriate APID_Av_Flg to determine if the temperatures are valid.

Also fixed logic error in C_CalcNrg where energy values would not be returned if compression was off (ie: Step=1).

PR0000172 : WFMgr crashes when called without anc09

g_havePAD is checked before calling get_anc09_degrades. Various IEEE errors were fixed by checking for non-waveforms. The minimum amplitude for saturation (in counts) was added to anc07_001_01_0004.dat and the code that reads it.

CR0000159 : Subscript out of bounds in c_calcnrg_mod

Fixed by detecting cases where there are no waveforms.

CR0000154 : modify anc08 software for efficient space allocation

Software in this module, though functional, was utilizing an inefficient data array allocation scheme. It requested a space size based upon a maximized value of the number of header elements that might contain degradation data. It was changed to mimic a similar function used in the anc09 code, which implements a preliminary step to read through the list of degradations in the header and to count them. The file is then rewound, an exact space size is allocated, and the data is read into memory.

CR0000158 : Divide by 0 in qap01_computetrdsum

The software changes in this branch/module were minimal. A divide by error check was using the wrong variable, thus nullifying the check's value. This coding error was corrected.

CR0000121 GL0P Sanity checking interface to L1A

Have implemented comparisons between the ANC29 record and the following input data, dependent upon the source APID:

```
12 GLA00_prod%AD_LgSci(1)%subpkt(1)%sp%i_shot_ctr
13 GLA00_prod%AD_SmSci(1)%subpkt(1)%sp%i_shot_ctr
```

```

14 GLA00_prod%AD_Eng(1)%subpkt%sp%i_shot_ctr
15 GLA00_prod%PC_Sci(1)%i_shot_ctr
16 GLA00_prod%PC_Eng(1)%i_shot_ctr
17 GLA00_prod%CD_Sci(1)%i_shot_ctr
18 GLA00_prod%CD_Eng(1)%i_shot_ctr
19 GLA00_prod%AN_Sci(1)%shot(1)%i_ctr
20 phdr%i_seqcnt
21 phdr%i_seqcnt
22 phdr%i_seqcnt
23 phdr%i_seqcnt
24 phdr%i_seqcnt
25 phdr%i_seqcnt
26 GLA00_prod%LPA(1)%subpkt(1)%i_shot_ctr
27 phdr%i_seqcnt
28 phdr%i_seqcnt
31 phdr%i_seqcnt
32 phdr%i_seqcnt
33 phdr%i_seqcnt
34 phdr%i_seqcnt
35 phdr%i_seqcnt
36 phdr%i_seqcnt
38 phdr%i_seqcnt
48 phdr%i_seqcnt
49 phdr%i_seqcnt
50 phdr%i_seqcnt
55 GLA00_prod%Lg_SW2(1)% iAD_Shot_ct
126 phdr%i_seqcnt
1984 phdr%i_seqcnt

```

Also implemented a scheme such that input APID CRC values are stored in the ANC29 and ANC32 headers during GLAS_L0proc execution. Upon GLAS_L1A execution these stored values are compared with CRC checksums of the L1A input APIDs to guarantee that the correct APID/ANC29 correlations have been chosen.

Changed files

```

./data/anc07_001_01_0000.dat
./data/anc07_001_01_0004.dat
./Makefile
./src/common_libs/anc_lib/anc_hdr_mod.f90
./src/common_libs/anc_lib/anc07_wf_mod.f90
./src/common_libs/anc_lib/anc08_pod_mod.f90
./src/common_libs/anc_lib/vers_anc_mod.f90
./src/common_libs/exec_lib/ReadAnc_mod.f90
./src/common_libs/file_lib
./src/common_libs/file_lib/cksum.c
./src/common_libs/platform_lib/const_wf_mod.f90
./src/common_libs/platform_lib/vers_platform_mod.f90
./src/common_libs/prod_lib/GLA00_mod.f90
./src/common_libs/prod_lib/GLA05_scal_mod.f90
./src/common_libs/prod_lib/vers_prod_mod.f90
./src/glas_alt/ElevMgr_mod.f90
./src/glas_alt/GLAS_Alt.f90
./src/glas_alt/WFMgr_mod.f90
./src/glas_l0p/GLAS_L0proc.f90
./src/glas_l0p/glop_mod.f90
./src/glas_l1a/GLAS_L1A.f90
./src/glas_l1a/L1AMgr_mod.f90

```

```
./src/11a_lib/C_CalcNrg_mod.f90
./src/11a_lib/L_Alt_mod.f90
./src/11a_lib/QAP01_mod.f90
./src/11a_lib/vers_11a_mod.f90
./src/waveforms/W_Assess/W_Assess_mod.f90
./src/waveforms/W_FunctionalFt/W_FunctionalFt_mod.f90
./src/wf_lib/vers_wf_mod.f90
```