XMM Optical Monitor

DEPARTMENT OF PHYSICS
UNIVERSITY OF CALIFORNIA, SANTA BARBARA   Author: Jamie Kennea (UCSB), Cheng Ho (LANL), Tim Sasseen (UCSB)

Description of Changes to DPU Software August-December 2000 (Release 12)

DOCUMENT: XMM-OM/UCSB/TC/0053 Version 1.0   12-December-00

Contents

This document describes the changes made for release “12”, released to ESA December 2000. It summarises the changes and fixes reported in the ECRs and NCRs below. The description of the revision versions indicates changes since the last delivery of August 2000. For previous changes to the DPU flight software, see documentation is available on the UCSB web site (`http://xmmom.physics.ucsb.edu/`). Test reports specific to the changes described in this document are found in XMMOM/UCSB/RP/0008.

List of NCRs/ECRS Described in this Document

NCR/ECR	Description	Status this Release
NCR 186	Fast mode data contain internal inconsistency seen as more slices sampled than commanded in some tracking frame.	open
NCR 206	FAQ failed, why the access alerts came out of a field acq	open
NCR 210	Synchronization of two fast mode areas	closed
NCR 212	Ordering issue with DP_FAQ packet	closed

1 NCR 186 - Fast mode data contain internal inconsistency seen as more slices sampled than commanded in some tracking frame.

Explanation:

This problem is caused by the fact that fast mode data buffer is in 16 bit small words. This problem occurs under the condition when a large fast mode window in a faint field (or narrow band filter) with small sampling time and long frame time. For example, let's say the frame time is 20 sec, fast mode slice time is 64 DPU msec, and the fast mode area is 512 pixels. Then the full dynamic range required for this case (under thw worse case scenario when no photon was detected in the fast window during this frame) is 20*1024*512/64 = 0x28000 which cannot be expressed in 16-bit data record. In the current fast mode code, when this happens, the information above 16 bit will be lost and this leads to an unrecoverable ambiguity.

FAST24 is a new fast mode implementation that using 24 bit words instead, and fixes the problem described above. However as this code change would change the format for output fast mode data, FAST24 mode is not currently compiled (as controlled by the FAST24 compile flag), and therefore the format of fast mode is unchanged from Release 11.

Affected codes:
- blue/src/blue.asm
- white/src/cc/whitedsp.c

```
blue/src/blue.asm
----------------------------
revision 1.36
date: 2000/12/06 21:50:25;  author: ho;  state: Exp;  lines: +21 -28
CH20001206: 2000aug implementation of the FAST24 configuration.
----------------------------
revision 1.35
date: 2000/04/18 18:15:28;  author: ho;  state: Exp;  lines: +58 -1
CH20000418: NCR186. Preliminary codes to implement the FAST24 scheme.
This scheme will get around the 16-bit dynamic range overflow problem
in the fast mode. Consult with ESA/MSSL requirements for the
appropriate time to turn on the FAST24 flag and ddliver after thorough
testing.
----------------------------

white/src/cc/whitedsp.c
----------------------------
revision 1.76
date: 2000/12/06 21:52:46;  author: ho;  state: Exp;  lines: +18 -5
CH20001206: 1. Split up the GET_BFAST so it does not SCI broadcast,
try to load BFAST simultaneously and eat up bus cycles.
2. Change code to do DD_FST24 data downlink.
----------------------------
```

2 NCR 206 - Field Acquisition fails often.

Explanation:

Field Acquisition (FAQ) was seen to fail often. Examination of the DP_FAQ data obtained over many orbits revealed that although FAQ works with a reasonably high success rate (3/22) for the first observation of the, the subsequent FAQs in an orbit all fail. This was found to be due to variables not being correctly reset after the first FAQ. The value of "allowed_to_miss", was also being incorrectly interpreted by the software such that FAQ fails if 3 and not 4 (as defined) stars are defined. A new FAQ success criterion was also defined in that a minimum of 4 matches are necessary to do FAQ.

For further debugging a DD_REF packet is now generated as part of the FAQ process.

Affected codes:
- include/white_default.h
- include/white_shared_variables.h
- include/white_static_variables.h
- white/src/cc/su_initialize.c
- white/src/cc/su_deliverdata.c
- white/src/cc/su_makeags.c
- white/src/cc/whitedsp.c

```
include/white_default.h
--------------------------------
revision 1.26
date: 2000/12/07 21:22:35;  author: jak;  state: Exp;  lines: +4 -1
NCR206: Define NEED_TO_FIND = 4, the minimum number of matches required for
FAQ to be successful.
--------------------------------

include/white_shared_variables.h
--------------------------------
revision 1.20
date: 2000/12/07 21:21:55;  author: jak;  state: Exp;  lines: +2 -1
NCR 206: Define need_to_find.
--------------------------------

include/white_static_variables.h
--------------------------------
revision 1.21
date: 2000/12/07 21:40:50;  author: jak;  state: Exp;  lines: +3 -2
NCR206: Small fix.
--------------------------------
revision 1.20
date: 2000/12/07 21:21:48;  author: jak;  state: Exp;  lines: +2 -2
NCR 206: Define need_to_find.
--------------------------------

white/src/cc/su_initialize.c
--------------------------------
revision 1.44
date: 2000/12/07 21:31:09;  author: jak;  state: Exp;  lines: +2 -2
NCR206: Fix initialisation of need_to_find.
--------------------------------
revision 1.43
date: 2000/12/07 21:20:28;  author: jak;  state: Exp;  lines: +2 -1
```

```
NCR206: Included "need_to_find" FAQ sucess criterion. I.e. if less than 4 stars
are matched then FAQ is considered a failure. This is to catch the possibility
that if 4 stars are uploaded and 0 are found, this can be considered to be
successful.
--------------------------------

white/src/cc/su_deliverdata.c
--------------------------------
revision 1.44
date: 2000/12/06 19:22:55;  author: xmmom;  state: Exp;  lines: +14 -1
R206: Generate DD_REF for FAQ debugging.
--------------------------------

white/src/cc/su_makeags.c
--------------------------------
revision 1.20
date: 2000/12/07 21:20:58;  author: jak;  state: Exp;  lines: +2 -2
NCR206: Included "need_to_find" FAQ sucess criterion. I.e. if less than 4 stars
are matched then FAQ is considered a failure. This is to catch the possibility
that if 4 stars are uploaded and 0 are found, this can be considered to be
successful.
--------------------------------
revision 1.19
date: 2000/12/06 19:25:12;  author: xmmom;  state: Exp;  lines: +2 -2
NCR206: Minor change to FAQ failure criterion.
--------------------------------

white/src/cc/whitedsp.c
--------------------------------
revision 1.75
date: 2000/12/06 19:22:05;  author: xmmom;  state: Exp;  lines: +6 -1
NCR206: Generate DD_REF for FAQ debugging.
--------------------------------
revision 1.74
date: 2000/11/30 20:19:41;  author: jak;  state: Exp;  lines: +5 -1
NCR206: Added initialisation of acquistion_iter and box at the start of
acquire_field_task() as having these variables as non-zero when starting
the FAQ algorithm causes the DPU to misbehave.
--------------------------------
```

3 NCR 210 - Synchronization of two fast mode areas

Explanation:

If two fast mode areas are requested a synchronization error occurs. The is to split the command to get the fast mode configuration into a separate command for each blue DSP with a a 20-30 ms gap in between.

Affected codes:

- white/src/cc/whitedsp.c

```
white/src/cc/whitedsp.c
----------------------------
revision 1.76
date: 2000/12/06 21:52:46;  author: ho;  state: Exp;  lines: +18 -5
CH20001206: 1. Split up the GET_BFAST so it does not SCI broadcast,
try to load BFAST simultaneously and eat up bus cycles.
2. Change code to do DD_FST24 data downlink.
----------------------------
```

4 NCR 212 - Ordering issue with DP_FAQ packet

Explanation:

The output DP_FAQ contains both coordinates of the uploaded star list and the matches stars within the acquisition frame. In the case that not all uploaded stars are matched, the matched list will be shorter than uploaded list, with padding at the end of the list. The structure of the DP_FAQ packet was changed so that in the case where a reference star is not found, it's coordinates in the matched star list are given as 0xFFFFFF, making automatic matching of uploaded and matched star lists easier.

Affected codes:
• white/src/cc/su_deliverdata.c • white/src/cc/su_makeags.c

```
white/src/cc/su_deliverdata.c
----------------------------
revision 1.43
date: 2000/10/13 20:50:04;  author: jak;  state: Exp;  lines: +15 -7
NCR 212: When delivering a DP_FAQ, the list of found guide star positions
now has dummy values when a star is not found, making the reference star
list and the found star list easier to compare. Whenever a star is not
found it's x and y coordinates are set to 0xFFFFFF.
----------------------------

white/src/cc/su_makeags.c
----------------------------
revision 1.18
date: 2000/10/13 20:43:24;  author: jak;  state: Exp;  lines: +11 -1
NCR 212: When a uploaded FAQ reference star is not found, a dummy index is
placed into the ref_star_index instead of doing nothing. This dummy index
is then trapped in su_deliverdata and the coordinates of the reference
star that was not found are set to 0xFFFFFF in the DP_FAQ alert. This
makes it easier to match up found stars with the uploaded reference star
list which previously had to be done by eye.
----------------------------
```