

## **SMEI Instrument Software Patch Validation Procedure**

**M.P.Cooke**

and

**C.J.Eyles**

**University of Birmingham**

Issue 2

2 April 2002

## 1. Table of Contents

1. Introduction	3
2. Software Upload Procedure	4
A. SMEI Patch File Format And Handling	5

## **1. Introduction**

The SMEI instrument has the capability to use new flight software while on-orbit. This procedure lays out the steps required to verify that this capability works correctly when performed via the RSC facility.

The procedure is based on section 4 of the SMEI flight software update procedure, SMEI/BU/PRO/006, which was successfully performed using the Spectrum Astro EGSE in October 2001. It is designed to be an overview to the explicit detail in the 'SMEI/BU/PRO/010 : SMEI Flight Model On-Orbit Procedures – Patching Supplement, Draft 2' document.

## 2. Software Upload Procedure

The procedure for this is a three step process. First, a software image must be uploaded into the appropriate non-volatile storage page in the DHU. A patch activate command is then issued, and the software revision number in telemetry checked to ensure that the DHU is running the newly uploaded software image. Finally, the uploaded software image is disabled, and the SMEI software revision number re-checked to ensure it was returned to the original value.

The steps referred to in the following procedure are from the 'SMEI/BU/PRO/010 - SMEI Flight Model On-Orbit Procedures – Patching Supplement, Draft 2' document.

1. Switch on SMEI if necessary, and enter configuration mode by issuing the SM\_DHU\_COLD command. Ensure everything is operating correctly. [Steps 1-4]
2. Upload the patch-<n>.rsc patch files to SMEI. These will put SMEI into patch mode, and upload the new software image into data page 0x42. The file will return SMEI to configuration mode when it is complete. Verify the command count is 37 when the upload is complete. [Steps 5-13]
3. Re-enter patch mode by issuing SM\_ENBL\_PTCH and then SM\_GOTO\_PTCH, and issue the SM\_PATCH\_ACT command, with the correct checksum for the newly uploaded software image. This checksum is in the comment fields at the start of the patch file. [Steps 14-15]
4. Power down SMEI. [Step 16]
5. Switch on SMEI, and verify the software revision number in the housekeeping shows 32. This is the SW\_REL telemetry point. [Steps 17-18]
6. Issue SM\_DHU\_COLD and verify that the software revision number changes to 64. The DHU does not start running a software image in page 0x42 until coldstart is issued. This is to permit recovery in the unlikely case that a corrupted software image in page 0x42 of the data e<sup>2</sup>proms still passes the checksum test. [Steps 19-21]
7. Verify telemetry is showing nominal operation. Ensure the timestamps are increasing, analogue and digital monitors are returning appropriate values, etc. Note that this new test software image does not contain support for observing mode.
8. Enter patch mode by issuing SM\_ENBL\_PTCH and SM\_GOTO\_PTCH, and then issue SM\_PATCH\_DEA. This will disable the software image uploaded in step 2 from being executed after the next restart of SMEI. [Steps 22-23]
9. Power down SMEI. [Step 24]
10. Switch on SMEI, and verify the software revision number shows 32. [Steps 25-26]
11. Issue SM\_DHU\_COLD, and verify the software revision number remains at 32. [Steps 27-29]
12. Switch off SMEI if necessary.

## A. SMEI Patch File Format And Handling

The patch files for this update are in a simple ASCII format as specified in the draft RSC Tasking Parser User's Manual. It should be noted that these files must be real-time and as such cannot contain any delay commands or timestamps. This limitation is avoided using multiple files, and noting the required delays at appropriate points in the overall procedure.

Shown below is an extract of a patch file. Note that the command lines have wrapped in this document, but do not in the patch file. I have used three dots to indicate a skipped section of the file.

### patch-1.rsc:

```
# Automatically converted from coriolis format bulk upload files
# File chunk: 1
#
# Patch upload created for loader.i0
#
# On (Y-M-D) 2002-03-26 at 11:03:03 UCT
#
# CRC to enable the code update is: 0x95E0 (38368)
#
# Enable patch mode
SM_ENBL_PATCH("SM_CMD_SEQ#="255")
# Goto patch mode
SM_GOTO_PATCH("SM_CMD_SEQ#="255")
# Upload 0x0040 words to patch buffer offset 0000
SM_PATCH_UPL("SM_CMD_SEQ#="255",SM_CMD_CRC="0x4C33",PATCH_BUFFER_OFFSET="0x0000",PATCH_DATA_00="0x00FF",PATCH_DATA_01="0xFF00",
PATCH_DATA_02="0x0000",PATCH_DATA_03="0x0000",PATCH_DATA_04="0x0000",PATCH_DATA_05="0x0000",PATCH_DATA_06="0x0000",
PATCH_DATA_07="0x0000",PATCH_DATA_08="0x0000",PATCH_DATA_09="0x0000",PATCH_DATA_10="0x0000",PATCH_DATA_11="0x0000",
PATCH_DATA_12="0x0000",PATCH_DATA_13="0x0000",PATCH_DATA_14="0x0000",PATCH_DATA_15="0x0000",PATCH_DATA_16="0x0000",
PATCH_DATA_17="0x0000",PATCH_DATA_18="0x0000",PATCH_DATA_19="0x0000",PATCH_DATA_20="0x0000",PATCH_DATA_21="0x0000",
PATCH_DATA_22="0x0000",PATCH_DATA_23="0x0000",PATCH_DATA_24="0x0000",PATCH_DATA_25="0x0000",PATCH_DATA_26="0x0000",
PATCH_DATA_27="0x0000",PATCH_DATA_28="0x0000")
#
SM_PATCH_UPL("SM_CMD_SEQ#="0",SM_CMD_CRC="0x0620",PATCH_BUFFER_OFFSET="0x001D",PATCH_DATA_00="0x0000",PATCH_DATA_01="0x0000",
PATCH_DATA_02="0x0000",PATCH_DATA_03="0x0000",PATCH_DATA_04="0x0000",PATCH_DATA_05="0x0000",PATCH_DATA_06="0x0000",
PATCH_DATA_07="0x0000",PATCH_DATA_08="0x0000",PATCH_DATA_09="0x0000",PATCH_DATA_10="0x0000",PATCH_DATA_11="0x0000",
PATCH_DATA_12="0x0000",PATCH_DATA_13="0x0000",PATCH_DATA_14="0x0000",PATCH_DATA_15="0x0000",PATCH_DATA_16="0x0000",
PATCH_DATA_17="0x0000",PATCH_DATA_18="0x0000",PATCH_DATA_19="0x0000",PATCH_DATA_20="0x0000",PATCH_DATA_21="0x0000",
PATCH_DATA_22="0x0000",PATCH_DATA_23="0x0000",PATCH_DATA_24="0x0000",PATCH_DATA_25="0x0000",PATCH_DATA_26="0x0000",
PATCH_DATA_27="0x0000",PATCH_DATA_28="0x0000")
#
SM_PATCH_UPL("SM_CMD_SEQ#="1",SM_CMD_CRC="0xB56E",PATCH_BUFFER_OFFSET="0x003A",PATCH_DATA_00="0x0000",PATCH_DATA_01="0x0000",
PATCH_DATA_02="0x0000",PATCH_DATA_03="0x0000",PATCH_DATA_04="0x0000",PATCH_DATA_05="0x0000",PATCH_DATA_06="0x0000",
PATCH_DATA_07="0x0000",PATCH_DATA_08="0x0000",PATCH_DATA_09="0x0000",PATCH_DATA_10="0x0000",PATCH_DATA_11="0x0000",
PATCH_DATA_12="0x0000",PATCH_DATA_13="0x0000",PATCH_DATA_14="0x0000",PATCH_DATA_15="0x0000",PATCH_DATA_16="0x0000",
PATCH_DATA_17="0x0000",PATCH_DATA_18="0x0000",PATCH_DATA_19="0x0000",PATCH_DATA_20="0x0000",PATCH_DATA_21="0x0000",
PATCH_DATA_22="0x0000",PATCH_DATA_23="0x0000",PATCH_DATA_24="0x0000",PATCH_DATA_25="0x0000",PATCH_DATA_26="0x0000",
PATCH_DATA_27="0x0000",PATCH_DATA_28="0x0000")
.
.
.
# Commit 0x1000 words to bank:offset 42:0000
SM_PATCH_CMT("SM_CMD_SEQ#="255",SM_CMD_CRC="0xE79D",PATCH_BUFFER_OFFSET="0x0000",PATCH_DATA_WORDS="0x1000",
PATCH_DESTIN_PAGE="0x0042",PATCH_DESTIN_OFFSET="0x0000")
#
# INSERT A 5 SECOND DELAY BEFORE RUNNING THE NEXT CHUNK OF THE PATCH
#
```

### patch-2.rsc:

```
# Automatically converted from coriolis format bulk upload files
# File chunk: 2
#
#
# INSERT A 5 SECOND DELAY BEFORE RUNNING THIS CHUNK OF THE PATCH
#
# Commit 0x1000 words to bank:offset 42:2000
SM_PATCH_CMT("SM_CMD_SEQ#="255",SM_CMD_CRC="0x9C17",PATCH_BUFFER_OFFSET="0x0000",PATCH_DATA_WORDS="0x1000",
PATCH_DESTIN_PAGE="0x0042",PATCH_DESTIN_OFFSET="0x2000")
#
# INSERT A 5 SECOND DELAY BEFORE RUNNING THE NEXT CHUNK OF THE PATCH
#
```