


HIFI OBS upload procedure

Prepared by	: Luc Dubbeldam	18 Feb 2009	

Short Title	HIFI OBS load
Doc ID:	SRON-U/HIFI/PR/2007-007
Issue:	1.10
Date	18 Feb 2009
Filename:	SRON_U_HIFI_PR_2007_007 issue 1.10.doc

Document Change Record

Issue	Date	Changed Section	Description of Change
1.0			
1.1	10 Oct 2007	3.6	Addition of HK-switch-off during EEPROM-write (as work around)
1.2	22 Jan 2008	Annex A	Addition of parameters for OBS 5.4
		3.6	Specify EEPROM-write to different partitions
		3.6 Annex B	Move work-around to Annex B
1.3		3.6	Added instructions on corrupted pages
		C	Added program length and page usage
		C	Added EEPROM page log
		Annex A	Addition of parameters for OBS 5.5
1.4		Annex A	Addition of parameters for OBS 5.6
1.5		Annex A	Addition of parameters for OBS 5.7
1.6		Annex A	Addition of parameters for OBS 5.8
1.7		Annex A	Addition of parameters for OBS 5.8.1
1.8		Annex A	Addition of parameters for OBS 5.9
1.9		Annex A	Addition of parameters for OBS 6.0
1.10		Annex A	Addition of parameters for OBS 6.1

Content

1 SCOPE OF WORK 2

2 DOCUMENT REFERENCES 3

2.1 Applicable documents 3

2.2 Reference Documents 3

3 TEST SPECIFICATION 4

3.1 Configuration 4

3.2 Initial situation 4

3.3 Final situation 4

3.4 General remarks 4

3.5 OBS upload 5

3.6 Write image to EEPROM (partition 1 or 2) 6

 3.6.1 *Preparation* 6

 3.6.2 *Execution* 6

 3.6.3 *Note on EEPROM page-usage* 7

1 SCOPE OF WORK

This document describes the procedure to upload an OBS image. It specifies the parameters to be used for a particular OBS-version for the more general procedures as described in RD-2 sections "OBS patching" and "EEPROM write".

2 DOCUMENT REFERENCES

2.1 Applicable documents

2.2 Reference Documents

- RD-1. SRON-U/HIFI/SP/2006-003 HIFI_MIB SRN issue 11.2
- RD-2. IFSI/OBS/MA/2005-001 HIFI ICU OBS SUM issue 4.3

3 TEST SPECIFICATION

The objective of this test is load a new OBS and write the new OBS to the EEPROM

3.1 Configuration

ICU

HIFI MIB 11.2

OBS image file in format of ICD14.

The ICU is commanded by CCS. There is no instrument IEGSE required to perform the test.

3.2 Initial situation

ICU is switched on and the OBS is running.

The OBS running at startup is referred to as "old OBS".

3.3 Final situation

OBS has been uploaded and started. The uploaded OBS is referred to as "new OBS".

The new OBS has been written to EEPROM, partition 1.

3.4 General remarks

The procedure is identical for Prime and Redundant.

All commands, except the HIFI_eeprom_write can be carried out any time, without precautions.

3.5 OBS upload

- Verify that OBS is running.
Note the OBS-version and patch number.
- Copy the complete old OBS to the high memory :
Send Telecommand: HIFI_copy_mem_to_high (HC170289)
Startaddress = 0x3FFFF, number of words = 0x20000
- Patch the copy of OBS in High memory using Memory Load Service
In concreto: load the image of the new OBS to memory starting at 0x3FFFF.
- Verify the OBS checksum for the three relevant segments in high-memory

Send the command HIFI_Check_PM_memory (HC169289) command for each of the three segments.

The addresses and expected CRC-values are specified in the relevant Annex.

The command HIFI_Check_PM_memory (HC169289) is completed successfully when the CRC calculated by the OBS matches the value of HIF_check_crc. The image has been uploaded successfully when the three TCs have completed successfully.

- Copy the patched OBS to the low memory and restart the OBS:
Send Telecommand: HIFI_copy_mem_to_low (HC171289)
Startaddress = 0x3FFFF, number of words = 0x20000

After the HIFI_copy_mem_to_low the old OBS is stopped. The Telecommand is therefore not acknowledged and the new OBS is started.

- Verify the new OBS version
Note the OBS-version and patch number.

3.6 Write image to EEPROM (partition 1 or 2)

3.6.1 Preparation

Before the image can be written to the EEPROM the following data is required:

- Are there EEPROM pages known as corrupted.
- Which image is present in partition 1. Which pages are occupied by this image?
- Which image is present in partition 2. Which pages are occupied by this image?
- How many pages are required for the new image?
- Does the new image fit in partition 1, without overwriting the image in partition 2?
- Does the new image fit in partition 2, without overwriting the image in partition 1?

When the partition is selected transform the list of corrupted pages. When the list of corrupted pages x is generated by the BSW list of pagenumbers-to-be-avoided x' in partition 1 is given by $x'=x+1$ and for partition 2: $x'=256-x$.

3.6.2 Execution

Make sure that the proper OBS version is running.

A work-around shall be used when the running OBS is version 5.3 or older. Refer to Annex A.5.

Make sure that the write-process (a few seconds) is not interrupted.

- Verify that OBS is running
- Decide to which partition the image will be written.

Partition 1: use HIF_partition= 1
Partition 2: use HIF_partiton= 2
- Issue the command HIFI_eeeprom_write
OBS_mem_end= as specified in applicable Annex.
HIF_partition_ID=1 or HIF_partition = 2
HIF_Npages_to_skip=0
(or HIF_Npages_to_skip = n, followed by the list x')

Verify successful EEPROM write:

- Cycle the power of the ICU.
- carry out the procedure to boot from EEPROM partition 1 or 2
- Verify the new OBS version
Note the OBS-version and patch number.
- The patch is successful when the version and patch number correspond to those of the new OBS.
- Fill out a new EEPROM usage form.

3.6.3 Note on EEPROM page-usage

The EEPROM consists of 256 pages. When an image is written to partition 1 the image starts at page 1 and occupies consecutive pages. It is possible to skip pages when they are known to be corrupted. partition 2 starts at page 256 and the image is written in the other direction.

At startup ALL pages of the EEPROM that have a page-header are tested. The test compares the CRC in the page-header with the calculated CRC. If the CRCs do not match the page is listed as "corrupted" in a EEPROM Memory test report.

The occurrence of an EEPROM Memory test report may or may not inhibit a successful load-boot. The corrupted page might be in the other partition. It might also not be part of the linked list of pages.

A corrupted page should be avoided during an EEPROM_write. The pagenumbers in the EEPROMemory test report runs from 0-255. The pagenumbers in the command from 1-256. The pagenumbers in the EEPROM write TC range from 1 to 256 for both partitions A and B.

If the Boot SW provided an index x for a bad page, that index x shall be transformed in the index $x' = (256 - x)$ before being used in the EEPROM write command for partition B.

When a page is corrupted and avoided during the subsequent EEPROM_Writes it stays corrupted.

When partition B is overwritten by partition A it is possible to boot successfully from partition B. The linked list of partition A probably has an inconsistent number of pages and almost certainly an incorrect CRC of the overall ASW.

The number of required pages can be derived from the start=address of the initialization segment and the end-address of the PM-code address. One page extra is needed for the interrupt-vector. Refer to C.

A Parameter specification

A.1 Addresses for OBS 5.2

Calculated values for HIFI_OBS 5.2:

checksum from 0x0, length 0x100: 52a2
 checksum from 0x4000, length 0x149f: b0b9
 checksum from 0x6000, length 0x12a2b: a11

To be checked after upload to 0x3ffff:
 Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
 HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
 HIF_check_crc = 0x52a2

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
 HIF_check_end = 0x43fff + 0x149f -1 = 0x4549d
 HIF_check_crc = 0xb0b9

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x6000 = 0x45fff
 HIF_check_end = 0x45fff + 0x12a2b -1 = 0x58a29
 HIF_check_crc = 0xa11

The minimum number of EEPROM-pages is:
 $(0x6000 - 0x4000 + 0x12a2b) * 3 / ((0x400 - 7) * 2) = 253569 / 2034 = 125$

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0x52a2
Initialization	0x43FFF	0x4549d	0xb0b9
PM-code	0x45FFF	0x58a29	0x0a11

Parameters to be used in HIFleprom_write:

OBS_mem_end= 0x18fff

A.2 Addresses for OBS 5.3

Calculated values for HIFI_OBS 5.3:

checksum from 0x0, length 0x100: 4ab7
checksum from 0x4000, length 0x149f: 6fa4
checksum from 0x5500, length 0x12af2: d32

To be checked after upload to 0x3ffff:
Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
HIF_check_crc = 0x4ab7

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
HIF_check_end = 0x43fff + 0x149f -1 = 0x4549d
HIF_check_crc = 0x6fa4

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
HIF_check_end = 0x454ff + 0x12af2 -1 = 0x57ff0
HIF_check_crc = 0xd32

The minimum number of EEPROM-pages is:
 $(0x5500 - 0x4000 + 0x12af2) * 3 / ((0x400 - 7) * 2) = 245718 / 2034 = 121$

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0x4ab7
Initialization	0x43FFF	0x4549d	0x6fa4
PM-code	0x454FF	0x57ff0	0x0d32

Parameters to be used in HIFleprom_write:

OBS_mem_end= 0x18fff

A.3 Addresses for OBS 5.4

Calculated values for HIFI_OBS 5.4:

checksum from 0x0, length 0x100: a423
 checksum from 0x4000, length 0x1490: d29b
 checksum from 0x5500, length 0x1280c: 31a0

To be checked after upload to 0x3ffff:
 Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
 HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
 HIF_check_crc = 0xa423

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
 HIF_check_end = 0x43fff + 0x1490 -1 = 0x4548e
 HIF_check_crc = 0xd29b

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
 HIF_check_end = 0x454ff + 0x1280c -1 = 0x57d0a
 HIF_check_crc = 0x31a0

The minimum number of EEPROM-pages is:
 $(0x5500 - 0x4000 + 0x1280c) * 3 / ((0x400 - 7) * 2) = 243492 / 2034 = 120$

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0xa423
Initialization	0x43FFF	0x4548e	0xd29b
PM-code	0x454FF	0x57d0a	0x31a0

Parameters to be used in HIFleeprom_write:

OBS_mem_end= 0x18fff

A.4 Addresses for OBS 5.5

Calculated values for HIFI_OBS 5.5:

checksum from 0x0, length 0x100: 2eac
checksum from 0x4000, length 0x14b0: 1d9e
checksum from 0x5500, length 0x1280c: 2fac

To be checked after upload to 0x3ffff:
Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
HIF_check_crc = 0x2eac

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
HIF_check_end = 0x43fff + 0x14b0 -1 = 0x454ae
HIF_check_crc = 0x1d9e

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
HIF_check_end = 0x454ff + 0x1280c -1 = 0x57d0a
HIF_check_crc = 0x2fac

The minimum number of EEPROM-pages is:
 $(0x5500 - 0x4000 + 0x1280c) * 3 / ((0x400 - 7) * 2) = 243492 / 2034 = 120$
Plus one additional page for the run-time header

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0x2eac
Initialization	0x43FFF	0x454ae	0x1d9e
PM-code	0x454FF	0x57d0a	0x2fac

Parameters to be used in HIFleeprom_write:

OBS_mem_end= 0x17DCF
(0x17D0B is at page 120, 0x17DCF is end of page 120, refer to Annex C)

A.5 Addresses for OBS 5.6

Calculated values for HIFI_OBS 5.6:

checksum from 0x0, length 0x100: f52c
 checksum from 0x4000, length 0x14eb: a6e7
 checksum from 0x5500, length 0x12d88: cfc5

To be checked after upload to 0x3ffff:
 Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
 HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
 HIF_check_crc = 0xf52c

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
 HIF_check_end = 0x43fff + 0x14eb -1 = 0x454e9
 HIF_check_crc = 0xa6e7

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
 HIF_check_end = 0x454ff + 0x12d88 -1 = 0x58286
 HIF_check_crc = 0xcfc5

The minimum number of EEPROM-pages is:
 $(0x5500 - 0x4000 + 0x12d88) * 3 / ((0x400 - 7) * 2) = 247704 / 2034 = 122$
 Plus one additional page for the run-time header

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0xf52c
Initialization	0x43FFF	0x454e9	0xa6e7
PM-code	0x454FF	0x58286	0xcfc5

Parameters to be used in HIFleeprom_write:

OBS_mem_end= 0x1831B
 (0x18287 is at page 122, 0x1831B is end of page 122, refer to Annex C)

A.6 Addresses for OBS 5.7

Calculated values for HIFI_OBS 5.7:

checksum from 0x0, length 0x100: f952
 checksum from 0x4000, length 0x14ec: b3fb
 checksum from 0x5500, length 0x12dea: 34a9

To be checked after upload to 0x3ffff:
 Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
 HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
 HIF_check_crc = 0xf952

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
 HIF_check_end = 0x43fff + 0x14ec -1 = 0x454ea
 HIF_check_crc = 0xb3fb

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
 HIF_check_end = 0x454ff + 0x12dea -1 = 0x582e8
 HIF_check_crc = 0x34a9

The minimum number of EEPROM-pages is:
 $(0x5500 - 0x4000 + 0x12dea) * 3 / ((0x400 - 7) * 2) = 247998 / 2034 = 122$
 Plus one additional page for the run-time header

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0xf952
Initialization	0x43FFF	0x454ea	0xb3fb
PM-code	0x454FF	0x582e8	0x34a9

Parameters to be used in HIFleeprom_write:

OBS_mem_end= 0x1831B
 (0x182e9 is at page 122, 0x1831B is end of page 122, refer to Annex C)

A.7 Addresses for OBS 5.8

Calculated values for HIFI_OBS 5.8:

checksum from 0x0, length 0x100: 6eea
checksum from 0x4000, length 0x14ec: fd86
checksum from 0x5500, length 0x12e00: e6fe

To be checked after upload to 0x3ffff:
Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
HIF_check_crc = 0x6eea

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
HIF_check_end = 0x43fff + 0x14ec -1 = 0x454ea
HIF_check_crc = 0xfd86

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
HIF_check_end = 0x454ff + 0x12e00 -1 = 0x582fe
HIF_check_crc = 0xe6fe

The minimum number of EEPROM-pages is:
 $(0x5500 - 0x4000 + 0x12e00) * 3 / ((0x400 - 7) * 2) = 248064 / 2034 = 122$
Plus one additional page for the run-time header

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0x6eea
Initialization	0x43FFF	0x454ea	0xfd86
PM-code	0x454FF	0x582fe	0xe6fe

Parameters to be used in HIFleprom_write:

OBS_mem_end= 0x1831B
(0x182ff is at page 122, 0x1831B is end of page 122, refer to Annex C)

A.8 Addresses for OBS 5.8.1

Calculated values for HIFI_OBS 5.8.1:

checksum from 0x0, length 0x100: 5026
 checksum from 0x4000, length 0x14ec: 9ecd
 checksum from 0x5500, length 0x12e54: a6b4

To be checked after upload to 0x3ffff:
 Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
 HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
 HIF_check_crc = 0x5026

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
 HIF_check_end = 0x43fff + 0x14ec -1 = 0x454ea
 HIF_check_crc = 0x9ecd

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
 HIF_check_end = 0x454ff + 0x12e54 -1 = 0x58352
 HIF_check_crc = 0xa6b4

The minimum number of EEPROM-pages is:
 $(0x5500 - 0x4000 + 0x12e54) * 3 / ((0x400 - 7) * 2) = 248064 / 2034 = 123$
 Plus one additional page for the run-time header

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0x5026
Initialization	0x43FFF	0x454ea	0x9ecd
PM-code	0x454FF	0x58352	0xa6b4

Parameters to be used in HIFleeprom_write:

OBS_mem_end= 0x18353
 (0x18353 is at page 123, 0x185C1 is end of page 123, refer to Annex C)

A.9 Addresses for OBS 5.9

Calculated values for HIFI_OBS 5.9:

checksum from 0x0, length 0x100: ee3d
checksum from 0x4000, length 0x14f2: ded8
checksum from 0x5500, length 0x12f36: 7b2e

To be checked after upload to 0x3ffff:
Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
HIF_check_crc = 0xee3d

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
HIF_check_end = 0x43fff + 0x14f2 -1 = 0x454f0
HIF_check_crc = 0xded8

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
HIF_check_end = 0x454ff + 0x12f36 -1 = 0x58434
HIF_check_crc = 0x7b2e

The minimum number of EEPROM-pages is:

$(0x5500 - 0x4000 + 0x12f36) * 3 / ((0x400 - 7) * 2) = 248994 / 2034 = 123$

Plus one additional page for the run-time header

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0xee3d
Initialization	0x43FFF	0x454f0	0xded8
PM-code	0x454FF	0x58434	0x7b2e

Parameters to be used in HIFleeprom_write:

OBS_mem_end= 0x185c1

(0x18435 is at page 123, 0x185C1 is end of page 123, refer to Annex C)

A.10 Addresses for OBS 6.0

Calculated values for HIFI_OBS 6.0:

checksum from 0x0, length 0x100: 2629
 checksum from 0x4000, length 0x14ed: 756d
 checksum from 0x5500, length 0x1301f: 727f

To be checked after upload to 0x3ffff:
 Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
 HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
 HIF_check_crc = 0x2629

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
 HIF_check_end = 0x43fff + 0x14ed -1 = 0x454eb
 HIF_check_crc = 0x756d

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
 HIF_check_end = 0x454ff + 0x1301f -1 = 0x5851d
 HIF_check_crc = 0x727f

The minimum number of EEPROM-pages is:
 $(0x5500 - 0x4000 + 0x1301f) * 3 / ((0x400 - 7) * 2) = 249693 / 2034 = 123$
 Plus one additional page for the run-time header

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0x2629
Initialization	0x43FFF	0x454eb	0x756d
PM-code	0x454FF	0x5851d	0x727f

Parameters to be used in HIFleeprom_write:

OBS_mem_end= 0x185c1
 (0x1851e is at page 123, 0x185C1 is end of page 123, refer to Annex C)

A.11 Addresses for OBS 6.1

Calculated values for HIFI_OBS 6.1:

checksum from 0x0, length 0x100: 2629
checksum from 0x4000, length 0x14ed: 3f89
checksum from 0x5500, length 0x1301f: 8e6d

To be checked after upload to 0x3ffff:
Use HIFI_check_PM_memory,

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x0 = 0x3ffff
HIF_check_end = 0x3ffff + 0x100 -1 = 0x400fe
HIF_check_crc = 0x2629

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x4000 = 0x43fff
HIF_check_end = 0x43fff + 0x14ed -1 = 0x454eb
HIF_check_crc = 0x3f89

Segment in High-memory:

HIF_check_start = 0x3ffff + 0x5500 = 0x454ff
HIF_check_end = 0x454ff + 0x1301f -1 = 0x5851d
HIF_check_crc = 0x8e6d

The minimum number of EEPROM-pages is:
 $(0x5500 - 0x4000 + 0x1301f) * 3 / ((0x400 - 7) * 2) = 249693 / 2034 = 123$
Plus one additional page for the run-time header

Parameters to be used in HIFI_Check_PM_memory (HC169289):

Segment	HIF_check_start	HIF_check_end	HIF_check_crc
Runtime header	0x3FFFF	0x400fe	0xee3d
Initialization	0x43FFF	0x454eb	0x3f89
PM-code	0x454FF	0x5851d	0x8e6d

Parameters to be used in HIFleeprom_write:

OBS_mem_end= 0x185c1
(0x1851e is at page 123, 0x185C1 is end of page 123, refer to Annex C)

B Write image to EEPROM (partition 1 or 2) for OBS 5.3 and earlier

Make sure that the proper OBS version is running.

Make sure that the write-process (a few seconds) is not interrupted.

- Verify that OBS is running
- As work-around for SPR 1402, send the command HIFI_Housekeeping_on:

HIF_FCU_S	ON
HIF_LCU_S	ON
HIF_WBSV_S	OFF
HIF_WBSH_S	OFF
HIF_HRSV_S	OFF
HIF_HRSH_S	OFF

This work around applies to OBS up to 5.3. The Spectrometer HK will stop automatically in OBS 5.4 e.v.

- Issue the command HIFI_eeeprom_write
OBS_mem_end= as specified in applicable Annex.
HIF_partition_ID=1 or HIF_partition=2
HIF_Npages_to_skip=0
- Restore original HK-collection by sending the command HIFI_Housekeeping_on.

Verify successful EEPROM write:

- Cycle the power of the ICU.
- carry out the procedure to boot from EEPROM partition 1
- Verify the new OBS version
Note the OBS-version and patch number.
- The patch is successful when the version and patch number correspond to those of the new OBS.

C Required number of pages

The EEPROM write command writes the interrupt-vector in one page, followed by the indicated memory-block, paginated into a number of pages.

The start address of the block is typical 0x4000 (start of init-segment) and the end-address is the end of the PM-code segment.

The number of pages is: $1 + (\text{endAddress} - \text{startAddress} + 1) * 3 / ((1024 - 7) * 2)$.

start address	end address	Number of pages
0x4000	0x17DCF	1 + 120
0x4000	0x18075	1 + 121
0x4000	0x1831B	1 + 122
0x4000	0x185C1	1 + 123
0x4000	0x18867	1 + 124
0x4000	0x18B0D	1 + 125
0x4000	0x18DB3	1 + 126
0x4000	0x19059	1 + 127
0x4000	0x192FF	1 + 128
0x4000	0x195A5	1 + 129
0x4000	0x1984B	1 + 130
0x4000	0x19AF1	1 + 131
0x4000	0x19D97	1 + 132
0x4000	0x1A03D	1 + 133
0x4000	0x1A2E3	1 + 134
0x4000	0x1A589	1 + 135
0x4000	0x1A82F	1 + 136
0x4000	0x1AAD5	1 + 137
0x4000	0x1AD7B	1 + 138
0x4000	0x1B021	1 + 139
0x4000	0x1B2C7	1 + 140

D EEPROM usage log

Date:	
partition 1:	
partition 2:	
avoided pages	

Purpose:	
Operator:	

P1	P2		P1	P2		P1	P2		P1	P2		P1	P2		P1	P2		P1	P2			
1	256		33	224		65	192		97	160		129	128		161	96		193	64		225	32
2	255		34	223		66	191		98	159		130	127		162	95		194	63		226	31
3	254		35	222		67	190		99	158		131	126		163	94		195	62		227	30
4	253		36	221		68	189		100	157		132	125		164	93		196	61		228	29
5	252		37	220		69	188		101	156		133	124		165	92		197	60		229	28
6	251		38	219		70	187		102	155		134	123		166	91		198	59		230	27
7	250		39	218		71	186		103	154		135	122		167	90		199	58		231	26
8	249		40	217		72	185		104	153		136	121		168	89		200	57		232	25
9	248		41	216		73	184		105	152		137	120		169	88		201	56		233	24
10	247		42	215		74	183		106	151		138	119		170	87		202	55		234	23
11	246		43	214		75	182		107	150		139	118		171	86		203	54		235	22
12	245		44	213		76	181		108	149		140	117		172	85		204	53		236	21
13	244		45	212		77	180		109	148		141	116		173	84		205	52		237	20
14	243		46	211		78	179		110	147		142	115		174	83		206	51		238	19
15	242		47	210		79	178		111	146		143	114		175	82		207	50		239	18
16	241		48	209		80	177		112	145		144	113		176	81		208	49		240	17
17	240		49	208		81	176		113	144		145	112		177	80		209	48		241	16
18	239		50	207		82	175		114	143		146	111		178	79		210	47		242	15
19	238		51	206		83	174		115	142		147	110		179	78		211	46		243	14
20	237		52	205		84	173		116	141		148	109		180	77		212	45		244	13
21	236		53	204		85	172		117	140		149	108		181	76		213	44		245	12
22	235		54	203		86	171		118	139		150	107		182	75		214	43		246	11
23	234		55	202		87	170		119	138		151	106		183	74		215	42		247	10
24	233		56	201		88	169		120	137		152	105		184	73		216	41		248	9
25	232		57	200		89	168		121	136		153	104		185	72		217	40		249	8
26	231		58	199		90	167		122	135		154	103		186	71		218	39		250	7
27	230		59	198		91	166		123	134		155	102		187	70		219	38		251	6
28	229		60	197		92	165		124	133		156	101		188	69		220	37		252	5
29	228		61	196		93	164		125	132		157	100		189	68		221	36		253	4
30	227		62	195		94	163		126	131		158	99		190	67		222	35		254	3
31	226		63	194		95	162		127	130		159	98		191	66		223	34		255	2
32	225		64	193		96	161		128	129		160	97		192	65		224	33		256	1

EEPROM usage log for 5.4 in P1 and 5.4 in P2

Date:	
partition 1:	5.4
partition 2:	5.4
avoided pages	none

Purpose:	
Operator:	

P1	P2		P1	P2		P1	P2		P1	P2		P1	P2		P1	P2		P1	P2			
1	256		33	224		65	192		97	160		129	128		161	96		193	64		225	32
2	255		34	223		66	191		98	159		130	127		162	95		194	63		226	31
3	254		35	222		67	190		99	158		131	126		163	94		195	62		227	30
4	253		36	221		68	189		100	157		132	125		164	93		196	61		228	29
5	252		37	220		69	188		101	156		133	124		165	92		197	60		229	28
6	251		38	219		70	187		102	155		134	123		166	91		198	59		230	27
7	250		39	218		71	186		103	154		135	122		167	90		199	58		231	26
8	249		40	217		72	185		104	153		136	121		168	89		200	57		232	25
9	248		41	216		73	184		105	152		137	120		169	88		201	56		233	24
10	247		42	215		74	183		106	151		138	119		170	87		202	55		234	23
11	246		43	214		75	182		107	150		139	118		171	86		203	54		235	22
12	245		44	213		76	181		108	149		140	117		172	85		204	53		236	21
13	244		45	212		77	180		109	148		141	116		173	84		205	52		237	20
14	243		46	211		78	179		110	147		142	115		174	83		206	51		238	19
15	242		47	210		79	178		111	146		143	114		175	82		207	50		239	18
16	241		48	209		80	177		112	145		144	113		176	81		208	49		240	17
17	240		49	208		81	176		113	144		145	112		177	80		209	48		241	16
18	239		50	207		82	175		114	143		146	111		178	79		210	47		242	15
19	238		51	206		83	174		115	142		147	110		179	78		211	46		243	14
20	237		52	205		84	173		116	141		148	109		180	77		212	45		244	13
21	236		53	204		85	172		117	140		149	108		181	76		213	44		245	12
22	235		54	203		86	171		118	139		150	107		182	75		214	43		246	11
23	234		55	202		87	170		119	138		151	106		183	74		215	42		247	10
24	233		56	201		88	169		120	137		152	105		184	73		216	41		248	9
25	232		57	200		89	168		121	136		153	104		185	72		217	40		249	8
26	231		58	199		90	167		122	135		154	103		186	71		218	39		250	7
27	230		59	198		91	166		123	134		155	102		187	70		219	38		251	6
28	229		60	197		92	165		124	133		156	101		188	69		220	37		252	5
29	228		61	196		93	164		125	132		157	100		189	68		221	36		253	4
30	227		62	195		94	163		126	131		158	99		190	67		222	35		254	3
31	226		63	194		95	162		127	130		159	98		191	66		223	34		255	2
32	225		64	193		96	161		128	129		160	97		192	65		224	33		256	1

EEPROM usage log for 5.6 in P1 and 5.6 in P2

Date:	
partition 1:	5.6
partition 2:	5.6
avoided pages	none

Purpose:	
Operator:	

P1	P2		P1	P2		P1	P2		P1	P2		P1	P2		P1	P2		P1	P2			
1	256		33	224		65	192		97	160		129	128		161	96		193	64		225	32
2	255		34	223		66	191		98	159		130	127		162	95		194	63		226	31
3	254		35	222		67	190		99	158		131	126		163	94		195	62		227	30
4	253		36	221		68	189		100	157		132	125		164	93		196	61		228	29
5	252		37	220		69	188		101	156		133	124		165	92		197	60		229	28
6	251		38	219		70	187		102	155		134	123		166	91		198	59		230	27
7	250		39	218		71	186		103	154		135	122		167	90		199	58		231	26
8	249		40	217		72	185		104	153		136	121		168	89		200	57		232	25
9	248		41	216		73	184		105	152		137	120		169	88		201	56		233	24
10	247		42	215		74	183		106	151		138	119		170	87		202	55		234	23
11	246		43	214		75	182		107	150		139	118		171	86		203	54		235	22
12	245		44	213		76	181		108	149		140	117		172	85		204	53		236	21
13	244		45	212		77	180		109	148		141	116		173	84		205	52		237	20
14	243		46	211		78	179		110	147		142	115		174	83		206	51		238	19
15	242		47	210		79	178		111	146		143	114		175	82		207	50		239	18
16	241		48	209		80	177		112	145		144	113		176	81		208	49		240	17
17	240		49	208		81	176		113	144		145	112		177	80		209	48		241	16
18	239		50	207		82	175		114	143		146	111		178	79		210	47		242	15
19	238		51	206		83	174		115	142		147	110		179	78		211	46		243	14
20	237		52	205		84	173		116	141		148	109		180	77		212	45		244	13
21	236		53	204		85	172		117	140		149	108		181	76		213	44		245	12
22	235		54	203		86	171		118	139		150	107		182	75		214	43		246	11
23	234		55	202		87	170		119	138		151	106		183	74		215	42		247	10
24	233		56	201		88	169		120	137		152	105		184	73		216	41		248	9
25	232		57	200		89	168		121	136		153	104		185	72		217	40		249	8
26	231		58	199		90	167		122	135		154	103		186	71		218	39		250	7
27	230		59	198		91	166		123	134		155	102		187	70		219	38		251	6
28	229		60	197		92	165		124	133		156	101		188	69		220	37		252	5
29	228		61	196		93	164		125	132		157	100		189	68		221	36		253	4
30	227		62	195		94	163		126	131		158	99		190	67		222	35		254	3
31	226		63	194		95	162		127	130		159	98		191	66		223	34		255	2
32	225		64	193		96	161		128	129		160	97		192	65		224	33		256	1